

SINUMERIK

SINUMERIK 840D sl Basesoftware and operating software

Commissioning Manual

Preface

Fundamental safety
instructions

1

SINUMERIK Operate (IM9)

2

Easy XML

3

SINUMERIK Integrate
Access MyMachine / OPC UA

4

SINUMERIK Integrate
Run MyScreens (BE2)

5

NCU Operating System (IM7)

6

PCU base software (IM8)

7

PCU base software (IM10)

8

Appendix

A

Valid for

Control:
SINUMERIK 840D sl/840DE sl

Software:
CNC software V4.7 SP1
SINUMERIK Operate V4.7 SP1

01/2015

6FC5397-1DP40-5BA2

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

DANGER

indicates that death or severe personal injury will result if proper precautions are not taken.

WARNING

indicates that death or severe personal injury may result if proper precautions are not taken.

CAUTION

indicates that minor personal injury can result if proper precautions are not taken.

NOTICE

indicates that property damage can result if proper precautions are not taken.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

WARNING

Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Preface

SINUMERIK documentation

The SINUMERIK documentation is organized in the following categories:

- General documentation
- User documentation
- Manufacturer/service documentation

Additional information

You can find information on the following topics at www.siemens.com/motioncontrol/docu:

- Ordering documentation/overview of documentation
- Additional links to download documents
- Using documentation online (find and search in manuals/information)

Please send any questions about the technical documentation (e.g. suggestions for improvement, corrections) to the following address:

docu.motioncontrol@siemens.com

My Documentation Manager (MDM)

Under the following link you will find information to individually compile OEM-specific machine documentation based on the Siemens content:

www.siemens.com/mdm

Training

For information about the range of training courses, refer under:

- www.siemens.com/sitrain
SITRAIN - Siemens training for products, systems and solutions in automation technology
- www.siemens.com/sinutrain
SinuTrain - training software for SINUMERIK

FAQs

You can find Frequently Asked Questions in the Service&Support pages under Product Support. <http://support.automation.siemens.com>

SINUMERIK

You can find information on SINUMERIK under the following link:

www.siemens.com/sinumerik

Target group

This documentation is intended for commissioning personnel.

The plant or system is readily assembled and wired. For the following steps, e.g. configuring the individual components, the Commissioning Manual contains all necessary information or at least references.

Benefits

The intended target group can use the Commissioning Manual to test and commission the system or the plant correctly and safely.

Utilization phase: Setup and commissioning phase

Standard scope

This documentation only describes the functionality of the standard version. Additions or revisions made by the machine manufacturer are documented by the machine manufacturer.

Other functions not described in this documentation might be executable in the control. This does not, however, represent an obligation to supply such functions with a new control or when servicing.

For the sake of simplicity, this documentation does not contain all detailed information about all types of the product and cannot cover every conceivable case of installation, operation, or maintenance.

Technical Support

You will find telephone numbers for other countries for technical support in the Internet under <http://www.siemens.com/automation/service&support>

CompactFlash Cards

- Do not remove the memory card while it is being accessed. This can lead to damage of the memory card and the SINUMERIK as well as the data on the memory card.
- Insert the memory card carefully and the right way round into the memory card slot (observe indicators such as arrow or similar). This way you avoid mechanical damage to the memory card or the device.
- Only use memory cards that have been approved by Siemens for use with SINUMERIK. Even though SINUMERIK follows general industry standards for memory cards, it is possible that memory cards from some manufacturers will not function perfectly in this device or are not completely compatible with it (you can obtain information on compatibility from the memory card manufacturer or supplier).
- For further information on handling CompactFlash cards, please refer to the NCU manuals.

Table of contents

	Preface	3
1	Fundamental safety instructions	9
1.1	General safety instructions	9
1.2	Handling electrostatic sensitive devices (ESD)	12
1.3	Industrial security	13
1.4	Residual risks of power drive systems.....	13
2	SINUMERIK Operate (IM9)	17
3	Easy XML	631
4	SINUMERIK Integrate Access MyMachine / OPC UA.....	761
5	SINUMERIK Integrate Run MyScreens (BE2)	791
6	NCU Operating System (IM7).....	1027
7	PCU base software (IM8)	1125
8	PCU base software (IM10)	1263
A	Appendix.....	1373
A.1	Overview	1373

Fundamental safety instructions

1.1 General safety instructions



DANGER

Danger to life due to live parts and other energy sources

Death or serious injury can result when live parts are touched.

- Only work on electrical devices when you are qualified for this job.
- Always observe the country-specific safety rules.

Generally, six steps apply when establishing safety:

1. Prepare for shutdown and notify all those who will be affected by the procedure.
2. Disconnect the machine from the supply.
 - Switch off the machine.
 - Wait until the discharge time specified on the warning labels has elapsed.
 - Check that it really is in a no-voltage condition, from phase conductor to phase conductor and phase conductor to protective conductor.
 - Check whether the existing auxiliary supply circuits are de-energized.
 - Ensure that the motors cannot move.
3. Identify all other dangerous energy sources, e.g. compressed air, hydraulic systems, or water.
4. Isolate or neutralize all hazardous energy sources by closing switches, grounding or short-circuiting or closing valves, for example.
5. Secure the energy sources against switching on again.
6. Ensure that the correct machine is completely interlocked.

After you have completed the work, restore the operational readiness in the inverse sequence.



WARNING

Danger to life through a hazardous voltage when connecting an unsuitable power supply

Touching live components can result in death or severe injury.

- Only use power supplies that provide SELV (Safety Extra Low Voltage) or PELV- (Protective Extra Low Voltage) output voltages for all connections and terminals of the electronics modules.



! WARNING

Danger to life when live parts are touched on damaged devices

Improper handling of devices can cause damage.

For damaged devices, hazardous voltages can be present at the enclosure or at exposed components; if touched, this can result in death or severe injury.

- Ensure compliance with the limit values specified in the technical data during transport, storage and operation.
- Do not use any damaged devices.



! WARNING

Danger to life through electric shock due to unconnected cable shields

Hazardous touch voltages can occur through capacitive cross-coupling due to unconnected cable shields.

- As a minimum, connect cable shields and the cores of cables that are not used at one end at the grounded housing potential.



! WARNING

Danger to life due to electric shock when not grounded

For missing or incorrectly implemented protective conductor connection for devices with protection class I, high voltages can be present at open, exposed parts, which when touched, can result in death or severe injury.

- Ground the device in compliance with the applicable regulations.

! WARNING

Danger to life due to fire spreading if housing is inadequate

Fire and smoke development can cause severe personal injury or material damage.

- Install devices without a protective housing in a metal control cabinet (or protect the device by another equivalent measure) in such a way that contact with fire is prevented.
- Ensure that smoke can only escape via controlled and monitored paths.

 WARNING**Danger to life through unexpected movement of machines when using mobile wireless devices or mobile phones**

Using mobile wireless devices or mobile phones with a transmit power > 1 W closer than approx. 2 m to the components may cause the devices to malfunction, influence the functional safety of machines therefore putting people at risk or causing material damage.

- Switch the wireless devices or mobile phones off in the immediate vicinity of the components.

 WARNING**Danger to life due to fire if overheating occurs because of insufficient ventilation clearances**

Inadequate ventilation clearances can cause overheating of components with subsequent fire and smoke. This can cause severe injury or even death. This can also result in increased downtime and reduced service lives for devices/systems.

- Ensure compliance with the specified minimum clearance as ventilation clearance for the respective component.

 WARNING**Danger to life when safety functions are inactive**

Safety functions that are inactive or that have not been adjusted accordingly can cause operational faults on machines that could lead to serious injury or death.

- Observe the information in the appropriate product documentation before commissioning.
- Carry out a safety inspection for functions relevant to safety on the entire system, including all safety-related components.
- Ensure that the safety functions used in your drives and automation tasks are adjusted and activated through appropriate parameterizing.
- Perform a function test.
- Only put your plant into live operation once you have guaranteed that the functions relevant to safety are running correctly.

Note**Important safety notices for Safety Integrated functions**

If you want to use Safety Integrated functions, you must observe the safety notices in the Safety Integrated manuals.



WARNING

Danger to life or malfunctions of the machine as a result of incorrect or changed parameterization

As a result of incorrect or changed parameterization, machines can malfunction, which in turn can lead to injuries or death.

- Protect the parameterization (parameter assignments) against unauthorized access.
- Respond to possible malfunctions by applying suitable measures (e.g. EMERGENCY STOP or EMERGENCY OFF).

1.2 Handling electrostatic sensitive devices (ESD)

Electrostatic sensitive devices (ESD) are individual components, integrated circuits, modules or devices that may be damaged by either electric fields or electrostatic discharge.



NOTICE

Damage through electric fields or electrostatic discharge

Electric fields or electrostatic discharge can cause malfunctions through damaged individual components, integrated circuits, modules or devices.

- Only pack, store, transport and send electronic components, modules or devices in their original packaging or in other suitable materials, e.g. conductive foam rubber or aluminum foil.
- Only touch components, modules and devices when you are grounded by one of the following methods:
 - Wearing an ESD wrist strap
 - Wearing ESD shoes or ESD grounding straps in ESD areas with conductive flooring
- Only place electronic components, modules or devices on conductive surfaces (table with ESD surface, conductive ESD foam, ESD packaging, ESD transport container).

1.3 Industrial security

Note

Industrial security

Siemens provides products and solutions with industrial security functions that support the secure operation of plants, solutions, machines, equipment and/or networks. They are important components in a holistic industrial security concept. With this in mind, Siemens' products and solutions undergo continuous development. Siemens recommends strongly that you regularly check for product updates.

For the secure operation of Siemens products and solutions, it is necessary to take suitable preventive action (e.g. cell protection concept) and integrate each component into a holistic, state-of-the-art industrial security concept. Third-party products that may be in use should also be considered. For more information about industrial security, visit Hotspot-Text (<http://www.siemens.com/industrialsecurity>).

To stay informed about product updates as they occur, sign up for a product-specific newsletter. For more information, visit Hotspot-Text (<http://support.automation.siemens.com>).



WARNING

Danger as a result of unsafe operating states resulting from software manipulation

Software manipulation (e.g. by viruses, Trojan horses, malware, worms) can cause unsafe operating states to develop in your installation which can result in death, severe injuries and/or material damage.

- Keep the software up to date.

You will find relevant information and newsletters at this address
(<http://support.automation.siemens.com>).

- Incorporate the automation and drive components into a holistic, state-of-the-art industrial security concept for the installation or machine.

You will find further information at this address
(<http://www.siemens.com/industrialsecurity>).

- Make sure that you include all installed products into the holistic industrial security concept.

1.4 Residual risks of power drive systems

The control and drive components of a drive system are approved for industrial and commercial use in industrial line supplies. Their use in public line supplies requires a different configuration and/or additional measures.

These components may only be operated in closed housings or in higher-level control cabinets with protective covers that are closed, and when all of the protective devices are used.

These components may only be handled by qualified and trained technical personnel who are knowledgeable and observe all of the safety instructions on the components and in the associated technical user documentation.

When assessing the machine's risk in accordance with the respective local regulations (e.g., EC Machinery Directive), the machine manufacturer must take into account the following residual risks emanating from the control and drive components of a drive system:

1. Unintentional movements of driven machine components during commissioning, operation, maintenance, and repairs caused by, for example,
 - Hardware and/or software errors in the sensors, control system, actuators, and cables and connections
 - Response times of the control system and of the drive
 - Operation and/or environmental conditions outside the specification
 - Condensation/conductive contamination
 - Parameterization, programming, cabling, and installation errors
 - Use of wireless devices/mobile phones in the immediate vicinity of the control system
 - External influences/damage
2. In the event of a fault, exceptionally high temperatures, including an open fire, as well as emissions of light, noise, particles, gases, etc. can occur inside and outside the inverter, e.g.:
 - Component failure
 - Software errors
 - Operation and/or environmental conditions outside the specification
 - External influences/damage

Inverters of the Open Type/IP20 degree of protection must be installed in a metal control cabinet (or protected by another equivalent measure) such that contact with fire inside and outside the inverter is not possible.

3. Hazardous shock voltages caused by, for example,
 - Component failure
 - Influence during electrostatic charging
 - Induction of voltages in moving motors
 - Operation and/or environmental conditions outside the specification
 - Condensation/conductive contamination
 - External influences/damage

4. Electrical, magnetic and electromagnetic fields generated in operation that can pose a risk to people with a pacemaker, implants or metal replacement joints, etc., if they are too close
5. Release of environmental pollutants or emissions as a result of improper operation of the system and/or failure to dispose of components safely and correctly

Note

The components must be protected against conductive contamination (e.g. by installing them in a control cabinet with degree of protection IP54 according to IEC 60529 or NEMA 12).

Assuming that conductive contamination at the installation site can definitely be excluded, a lower degree of cabinet protection may be permitted.

For more information about residual risks of the components in a drive system, see the relevant sections in the technical user documentation.

SINUMERIK

SINUMERIK 840D sl SINUMERIK Operate (IM9)

Commissioning Manual

Valid for

Control:
SINUMERIK 840D sl/840DE sl

Software:
CNC software V4.7 SP1
SINUMERIK Operate V4.7 SP1

<u>System overview</u>	1
<u>Principles of start-up</u>	2
<u>General settings</u>	3
<u>Licensing</u>	4
<u>Creating drive connections</u>	5
<u>Channel menu</u>	6
<u>Machine and setting data</u>	7
<u>Customizing the "Machine" operating area</u>	8
<u>Simulation and simultaneous recording</u>	9
<u>Spindle functions</u>	10
<u>Drive system</u>	11
<u>Tool management</u>	12
<u>Configuring alarms</u>	13
<u>Collision avoidance</u>	14
<u>Data backup</u>	15
<u>Configuring the network</u>	16

SINUMERIK 840D sl SINUMERIK Operate (IM9)

Commissioning Manual

Continuation

Service and diagnostics

17

OEM-specific online help

18

SINUMERIK Integrate -
AMB, AMC, AMM

19

Safety Integrated

20

Configuring cycles

21

Additional language

22

SINUMERIK Operate on PC/
PCU

23

HT 8

24

List of abbreviations/
acronyms

A

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

DANGER

indicates that death or severe personal injury will result if proper precautions are not taken.

WARNING

indicates that death or severe personal injury may result if proper precautions are not taken.

CAUTION

indicates that minor personal injury can result if proper precautions are not taken.

NOTICE

indicates that property damage can result if proper precautions are not taken.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

WARNING

Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Table of contents

1	System overview.....	15
1.1	System overview.....	15
1.2	System requirements.....	16
2	Principles of start-up.....	17
2.1	Data structure of SINUMERIK Operate.....	17
2.2	Editing files.....	20
2.2.1	Editing the file at the controller.....	20
2.2.2	Editing a file externally.....	22
3	General settings.....	25
3.1	Changing the language.....	25
3.2	Set date/time.....	26
3.3	Configuring Caps Lock.....	28
3.4	Activating/deactivating the virtual keyboard.....	28
3.5	Defining the access rights for the "HMI restart" softkey.....	29
3.6	Setting the screensaver.....	29
3.7	Inserting a user-specific run up screen.....	30
3.8	Access levels	30
3.8.1	Definition of access levels.....	30
3.8.2	Modifying the access levels password.....	31
3.8.3	Access levels for programs.....	32
3.8.4	Access rights for files.....	34
3.8.5	Access levels for softkeys.....	35
3.8.6	This is how you define new access levels for softkeys.....	36
3.9	Program selection.....	37
3.9.1	Displaying the softkey.....	37
4	Licensing.....	41
4.1	Determining the license requirement.....	42
4.2	Activating the test license.....	44
4.3	Exporting the license requirement.....	45
4.4	Reading in the license key.....	45
5	Creating drive connections.....	47
5.1	Setting up drives.....	47
5.2	Setting EXTCALL.....	48
5.3	Setting the EES.....	49

5.4	Creating a global part program memory.....	50
6	Channel menu.....	51
6.1	Configuring a channel menu.....	51
6.2	Constraints.....	52
6.3	General structure of the channel menu.....	52
6.4	Defining channel groups in the "netnames.ini" file.....	53
6.5	Configuring operating stations in the "config.ini" file.....	55
6.6	Example: Settings for 1:N.....	57
6.7	Distribution via job lists in the case of 1:N.....	59
7	Machine and setting data.....	61
7.1	Machine and setting data.....	61
7.2	Displaying/editing machine data.....	61
7.3	Displaying/editing display machine data.....	64
7.4	Displaying/editing setting data.....	64
7.5	Displaying/editing drive parameters.....	66
7.6	Machine data display options.....	68
7.7	Editing machine data and drive parameters	70
7.7.1	Editing hexadecimal values	70
7.7.2	Editing BICO values.....	71
7.7.3	Editing enum values.....	71
7.8	Searching for data.....	72
7.9	Managing data.....	74
7.10	User views.....	76
7.10.1	Creating a user view.....	77
7.10.2	Editing the user view.....	78
7.10.3	Deleting a user view.....	80
7.11	Plain texts for machine and setting data.....	80
8	Customizing the "Machine" operating area.....	83
8.1	Setting the font size of the actual value display.....	83
8.2	Inserting a user-specific logo.....	83
8.3	Configuring the display of the G-code groups.....	84
8.4	Configuring the channel operational message.....	85
8.5	Deactivating program test.....	89
8.6	Activating the "Teach In" function.....	89
8.7	Block search.....	91
8.7.1	Activating block search mode.....	91
8.7.2	Accelerated block search for execution from external.....	92

8.8	Multi-channel support.....	92
8.9	Manual machine.....	94
8.10	User status display (OEM).....	96
9	Simulation and simultaneous recording.....	101
9.1	Simulation overview.....	101
9.2	Setting the technology for simulation.....	103
9.3	Simultaneous recording overview.....	105
9.4	Clamping a blank.....	106
10	Spindle functions.....	111
10.1	Spindle control.....	111
10.2	Analog spindle.....	112
10.3	Leadscrew error compensation.....	113
10.4	Spindle diagnostics.....	114
10.4.1	Evaluation of the status signals.....	114
10.4.2	Fetch temperatures.....	116
10.4.3	Fetch speed, torque and power.....	116
10.4.4	Checking the clamping system	117
10.4.5	Fetching the logistics data.....	118
11	Drive system.....	119
11.1	Commissioning of drives	119
12	Tool management.....	121
12.1	Machine data for tool management.....	121
12.1.1	Settings with/without magazine management.....	121
12.1.2	Configuring the access levels of the tool management.....	122
12.1.3	Additional settings.....	124
12.2	Configuring the user interface.....	127
12.2.1	General settings.....	129
12.2.2	Configure the tool lists.....	134
12.2.2.1	Tool parameter identifiers.....	138
12.2.2.2	Cutting parameter identifiers.....	140
12.2.2.3	Monitoring parameter identifiers.....	142
12.2.2.4	Grinding parameter identifiers.....	143
12.2.2.5	Magazine location parameter identifiers.....	144
12.2.2.6	Magazine location adapter parameter identifiers.....	145
12.2.2.7	Multitool parameter identifiers.....	145
12.2.2.8	Multitool location parameter identifiers.....	146
12.2.3	Configure the list parameters.....	147
12.2.4	List of tool types.....	151
12.2.5	Configure tool types.....	153
12.2.6	Configuring the "More data" window.....	154
12.2.7	Configure the "New tool - favorites" window.....	156
12.2.8	Configuring the "New Tool" window.....	156
12.2.9	Configuring default values for new tools.....	157

12.2.10	Configuring the "Details" window.....	161
12.2.11	Assigning a name for magazines and magazine locations.....	162
12.2.12	Assigning a name for magazine location types.....	164
12.2.13	Assignment of magazines to channels.....	165
12.2.14	Coolant and tool-specific functions.....	167
12.2.15	Reason for the tool change when reactivating.....	170
12.2.16	Configure the code carrier connection.....	174
12.3	Creating OEM texts.....	176
12.3.1	Identifiers of the standard texts.....	177
12.3.2	Examples of OEM texts.....	180
12.4	Examples.....	182
12.4.1	Example: Configuring an OEM tool list.....	182
12.4.1.1	Adapting the configuration file.....	182
12.4.1.2	Adaptation of the customer text file.....	184
12.4.2	Example: Configuring magazine location types with names.....	186
12.4.2.1	Adapting the configuration file.....	186
12.4.2.2	Adaptation of the customer text file.....	187
12.5	Working with two toolholders.....	190
12.6	Editor for the tool and magazine management.....	192
12.6.1	Overview.....	192
12.6.2	Calling the tool management.....	193
12.6.3	Editing the tool management.....	194
12.6.4	Parameterizing the tool management.....	196
12.6.4.1	Create tool unit.....	196
12.6.4.2	Define buffer locations.....	197
12.6.4.3	Define loading location.....	198
12.6.4.4	Create magazine configuration.....	198
12.6.4.5	Link assignments.....	200
13	Configuring alarms.....	203
13.1	Creating alarm and message texts via the user interface.....	203
13.2	Configuring alarm and message texts via alarm text files.....	205
13.2.1	Creating in-house alarm texts.....	206
13.2.2	Creating texts for indexed alarm parameters.....	208
13.2.3	Creating part program message texts.....	210
13.2.4	Changing alarm attributes.....	214
13.2.5	Replacing standard alarm texts.....	218
13.2.6	Range of alarms.....	222
13.2.7	Parameter specifications in alarm texts.....	223
13.2.8	Opening error file.....	224
13.3	Configuring an alarm log.....	224
13.3.1	Setting alarm logs from the user interface.....	225
13.3.2	Loading alarm log via configuration file.....	226
13.4	PLC alarms with parameters	230
13.4.1	Definition of a parameter of the octet string data type.....	230
13.4.2	Definition of the language-dependent formatting.....	233
13.5	Deactivating a warning.....	234

14	Collision avoidance.....	237
14.1	Overview.....	237
14.2	General sequence.....	238
14.3	Set collision avoidance.....	239
14.4	Graphic machine model editor.....	240
14.5	Editing the machine model.....	241
14.6	Changing and adapting the machine model view.....	243
14.6.1	Enlarging or reducing the machine graphic.....	243
14.6.2	Changing the section of the machine graphic.....	243
14.6.3	Rotating and shifting the machine graphics.....	244
14.6.4	Various machine model views.....	245
14.6.5	Displaying protection area elements for collision avoidance.....	246
14.7	Creating a kinematic structure.....	247
14.7.1	Kinematic elements.....	247
14.7.1.1	Creating a kinematic chain element.....	247
14.7.1.2	Kinematic rotation and parallel kinematic rotation.....	248
14.7.1.3	Linear axis and parallel linear axis.....	249
14.7.1.4	Rotary axis and parallel rotary axis.....	249
14.7.1.5	Offset and parallel offset.....	250
14.7.2	Protection areas.....	251
14.7.2.1	Creating protection areas.....	251
14.7.2.2	Tool protection area	252
14.7.2.3	Machine protection area.....	253
14.7.3	Protection area elements.....	254
14.7.3.1	Creating protection area elements.....	254
14.7.3.2	Frame and parallel frame protection area element.....	255
14.7.3.3	Box and parallel box protection area elements.....	257
14.7.3.4	Sphere and parallel sphere protection area elements.....	259
14.7.3.5	Cylinder and parallel cylinder protection area element.....	260
14.7.3.6	File and parallel 3D file protection area element.....	262
14.7.4	Collision pair.....	264
14.7.4.1	Creating a collision pair.....	264
14.7.4.2	Collision pair.....	265
14.7.5	Supplementary condition.....	266
14.8	Collision avoidance example.....	266
14.8.1	Fundamentals.....	266
14.8.2	Create machine model example.....	269
14.8.2.1	Creating kinematic elements.....	269
14.8.2.2	Creating machine protection areas.....	271
14.8.2.3	Creating a tool protection area.....	274
14.8.2.4	Creating collision pairs.....	275
15	Data backup.....	277
15.1	Creating a start-up archive.....	278
15.2	Reading-in a start-up archive.....	280
15.3	Backing up the hardware configuration.....	281

15.4	Creating an archive with original data.....	282
15.5	Reading in an archive with original data.....	284
15.6	Generating the complete archive.....	285
15.7	Serial interface (V24 / RS232).....	285
15.7.1	Reading-in and reading-out archives via a serial interface.....	285
15.7.2	Setting interface parameters.....	287
15.8	Backing up setup data.....	289
15.9	Network settings.....	289
16	Configuring the network.....	293
16.1	Displaying the network overview.....	293
16.2	Settings of the system network.....	294
16.3	Settings of the factory network.....	297
16.4	Saving network settings.....	299
16.5	Configuring operator panels.....	300
16.5.1	Displaying the available operator panels.....	300
16.5.2	Configuring VNC connections.....	301
16.5.3	T:M:N configuration.....	302
16.6	Station-related network diagnostics.....	303
16.6.1	Displaying network adapters.....	304
16.6.2	Error analysis.....	305
16.6.2.1	Performing error diagnostics.....	305
16.6.2.2	MCP/EKS as network participant cannot be accessed.....	306
16.6.2.3	HMI on the PCU cannot establish a network connection to the NC	307
16.6.2.4	TCU cannot establish a network connection to the HMI.....	311
16.6.2.5	The TCU cannot establish a network connection to the assigned PCU.....	313
16.6.3	Configuring TCP/IP diagnostics.....	314
16.6.4	Ethernet network diagnostics.....	318
16.6.4.1	Diagnostics of the network and DHCP server.....	318
16.6.4.2	Messages.....	319
16.6.4.3	Network trace.....	319
16.6.4.4	Accessible nodes.....	323
17	Service and diagnostics.....	327
17.1	NC/PLC variables.....	327
17.1.1	Displaying and editing PLC and NC variables.....	327
17.1.2	Saving and loading screen forms.....	330
17.2	Displaying the Service overview.....	331
17.2.1	Selecting axes and drives.....	333
17.2.2	Axis diagnostics.....	335
17.2.3	Service drive.....	339
17.3	System utilization.....	347
17.4	Creating screenshots.....	347
17.5	Machine identity.....	348
17.5.1	Entering machine-specific information.....	349

17.5.2	Creating a selection file.....	350
17.5.3	Reading in a selection file.....	352
17.5.4	Save information.....	354
17.5.5	Adding hardware components.....	355
17.5.6	Configuration data.....	357
17.6	Logbook.....	357
17.6.1	1. Documenting startup.....	358
17.6.2	2. Defining start-up.....	358
17.6.3	Making a logbook entry.....	359
17.7	Action log.....	360
17.7.1	Setting the action log.....	360
17.7.2	Displaying the log file.....	362
17.7.3	Searching in the log files.....	363
17.7.4	Storing a log.....	363
17.7.5	Structure of a log file.....	364
17.7.6	Advanced settings.....	366
17.8	HMI trace.....	367
17.9	PROFIBUS diagnosis.....	368
17.9.1	Displaying details for DP slaves.....	370
17.10	Drive system.....	371
17.10.1	Displaying drive states.....	371
17.10.2	Displaying details of the drive objects.....	372
17.11	Remote diagnostics.....	374
17.11.1	Adapting remote diagnostics.....	374
17.11.2	Remote diagnostics via Teleservice adapter IE at X127.....	375
17.11.3	PLC control for remote access.....	378
17.12	Trace.....	379
17.12.1	General procedure.....	380
17.12.2	Trace session.....	381
17.12.2.1	Creating a session file.....	381
17.12.2.2	Saving the trace file.....	381
17.12.2.3	Load trace session file.....	382
17.12.3	Variable for Trace.....	383
17.12.3.1	Variables filter/search.....	383
17.12.3.2	Selecting attributes of a variable.....	385
17.12.3.3	Displaying details of a variable.....	388
17.12.4	Trace settings.....	389
17.12.4.1	Trace settings (PLC, NC, servo).....	389
17.12.4.2	Trace settings (drive).....	389
17.12.4.3	Trace options.....	391
17.12.4.4	Starting the trace.....	391
17.12.5	Evaluate a trace.....	392
17.12.5.1	Setting trace views.....	392
17.12.5.2	Selecting a variable.....	393
17.12.5.3	Scaling the display.....	394
17.12.5.4	Zooming the display.....	395
17.12.5.5	Position the cursor.....	396
17.12.5.6	Acquiring measured values.....	397

17.13	PROFIBUS/PROFINET and AS-i bus diagnostics.....	398
17.13.1	PROFIBUS/PROFINET.....	398
17.13.2	Displaying PROFIBUS/PROFINET diagnostics.....	398
17.13.3	Configuring the AS-i bus.....	400
17.13.4	Displaying AS-i bus diagnostics.....	402
18	OEM-specific online help.....	405
18.1	Overview.....	405
18.2	Generating HTML files.....	406
18.3	Generating the help book.....	409
18.4	Integrating the online help in SINUMERIK Operate.....	411
18.5	Saving help files.....	413
18.6	Generating online help for user alarms and machine data.....	413
18.7	Example: This is how you create an online help for NC/PLC variables.....	416
18.8	Example: This is how you create a programming online help.....	419
18.9	Help files in PDF format.....	421
19	SINUMERIK Integrate - AMB, AMC, AMM.....	423
20	Safety Integrated.....	425
20.1	Displaying machine data and parameters.....	425
20.1.1	View of the axes.....	425
20.1.2	General machine data for Safety Integrated.....	426
20.1.3	Axis-specific machine data in Safety Integrated.....	427
20.1.4	Drive machine data in Safety Integrated.....	428
20.1.5	Copying and confirming Safety Integrated data.....	429
20.1.6	Activating/deactivating start-up mode.....	430
20.1.7	Displaying the Safety-Integrated settings.....	431
20.2	Safety Integrated diagnostics.....	432
20.2.1	Status display for NCK Safety Integrated.....	432
20.2.2	Display SGE/SGA signals.....	433
20.2.3	Display SPL signals.....	434
20.2.4	Displaying Safety Integrated checksums.....	435
20.2.5	Safety Integrated Basic Functions (Drive).....	436
20.2.6	Safety Integrated fail-safe I/O.....	437
20.2.7	Displaying cam signals.....	438
20.2.8	Displaying SI communication data.....	439
20.2.9	SI communication send connections.....	439
20.2.10	SI communication receive connections.....	441
21	Configuring cycles.....	443
21.1	Activating turning/milling/drilling/grinding technologies.....	443
21.2	Technology cycles for drilling.....	448
21.3	Manufacturer cycles.....	450
21.3.1	Manufacturer cycles.....	450
21.3.2	Prerequisites for the simulation mode.....	451
21.3.3	Extending the PROG_EVENT standard cycle.....	452

21.3.4	Manufacturer cycle for tool change CUST_T and CUST_M6.....	452
21.3.5	CUST_TECHCYC.SPF manufacturer cycle.....	454
21.3.6	CUST_MULTICHAN user cycle.....	456
21.4	Milling.....	457
21.4.1	General.....	457
21.4.2	Technology cycles for milling.....	457
21.4.3	Setting-up ShopMill cycles for milling.....	458
21.4.4	Cylinder surface transformation (TRACYL).....	460
21.4.5	Example: Milling machine with the XYZ-AC axis configuration.....	461
21.4.6	ShopMill cycles for multiple clamping.....	465
21.5	Turning.....	467
21.5.1	General.....	467
21.5.2	Setting up cycles for turning.....	467
21.5.3	Setting-up ShopTurn cycles for turning.....	471
21.5.4	Setting up a counterspindle.....	473
21.5.5	Setting up the counterspindle under ShopTurn.....	475
21.5.6	Technology cycles for turning.....	477
21.5.7	Axis configuration of a lathe.....	481
21.5.8	Cylinder surface transformation (TRACYL).....	482
21.5.9	End face machining (TRANSMIT).....	485
21.5.10	Inclined Y axis (TRAANG).....	488
21.6	grinding.....	491
21.7	Swiveling.....	493
21.7.1	Technology cycles for swiveling.....	493
21.7.2	CYCLE800 checklist for the identification of the machine kinematics.....	501
21.7.3	Commissioning swivel data (kinematics chain).....	502
21.7.4	Examples of machine kinematics for the commissioning of the Swivel function.....	512
21.7.5	Manufacturer cycle CUST_800.SPF.....	521
21.7.6	CYCLE996 measure workpiece kinematics.....	527
21.8	Turning on milling machines.....	529
21.8.1	Setting up machine data and setting data.....	529
21.8.2	Setting up a kinematics transformation.....	534
21.8.3	Behavior at reset and power on.....	536
21.8.4	Setting the tool clamping angle.....	536
21.8.5	Setting up the Hirth joint.....	537
21.8.6	Different coordinate systems for milling and turning.....	538
21.8.7	Adaptations for CUST_800.....	539
21.8.8	Rotary axis positioning during block search.....	542
21.9	High-speed machining.....	542
21.9.1	High speed settings: Configuring CYCLE832.....	542
21.9.2	Manufacturer cycle CUST_832.SPF.....	545
21.10	Measuring cycles and functions.....	546
21.10.1	Measuring cycles and measurement functions, general.....	546
21.10.2	Manufacturer and user cycle CUST_MEACYC.SPF.....	549
21.10.3	Measuring in JOG mode.....	552
21.10.3.1	Measuring workpieces at the milling machines.....	554
21.10.3.2	Measuring tools at the milling machines.....	556
21.10.3.3	Measuring tools at the turning machines.....	561
21.10.3.4	Measuring without electronic probe in JOG.....	562

21.10.4	Measuring in AUTOMATIC mode.....	562
21.10.4.1	Measuring workpieces, general	564
21.10.4.2	Measuring workpieces at the milling machines.....	567
21.10.4.3	Measuring tools at the milling machines.....	568
21.10.4.4	Measuring workpieces at the turning machines.....	574
21.10.4.5	Measuring tools at the turning machines.....	575
21.10.5	Logging.....	576
21.10.5.1	Logging the measuring, general.....	576
21.10.5.2	Logging while measuring in the JOG mode.....	577
21.10.5.3	Logging while measuring in the AUTOMATIC mode.....	577
21.10.5.4	Manufacturer and user cycle CUST_MEAPROT.SPF.....	578
21.11	Compare cycles version.....	579
21.11.1	Display cycles version.....	579
21.11.2	Specify cycles version.....	581
22	Additional language.....	583
22.1	Installing additional languages.....	583
22.2	Uninstalling languages.....	584
22.3	Supported languages.....	584
23	SINUMERIK Operate on PC/PCU.....	587
23.1	Link OEMFrame application.....	587
23.2	Using interactive or silent installation versions.....	591
23.3	Setting the IP address of the NCU.....	593
23.4	SINUMERIK Operate exiting.....	594
24	HT 8.....	595
24.1	Configuring the traversing keys.....	595
24.2	Configuring user-specific key labeling.....	599
24.3	Configuring the function display at user-specific keys (U keys).....	602
A	List of abbreviations/acronyms.....	605
	Index.....	607

System overview

1.1 System overview

NCU

The SINUMERIK Operate V4.x operating software is an integral component of the CNC software V4.x and can therefore run on an NCU 7x0.3. The NCU's CompactFlash card is the standard data carrier of the CNC software.

The SINUMERIK Operate operating software is started automatically when the NCU software boots and in the delivery state offers its standard functional scope, which depends on the actual NC and PLC configuration.

PCU

The SINUMERIK Operate operating software is supplied on DVD and must then be installed on the hard disk of the PCU. The operating software is automatically started when the controller boots.

PC

The SINUMERIK Operate operating software is supplied on DVD and can be installed in a freely selectable directory.

The following sections describe the expansions and adaptations that the machine manufacturer can make using the SINUMERIK Operate operating software.

References

For a description of the commissioning procedure for the SINUMERIK 840D sl controller, please refer to the following documentation:

Commissioning Manual "IBN CNC: NCK, PLC, drive Commissioning Manual".

1.2 System requirements

Overview

It is a requirement that all system components be fully mounted, installed and configured. Commissioning of the NC and PLC must have been completed. The PLC is in the "Run" condition.

- USB keyboard: Handles text input with upper- and lower-case letters.
- Storage medium, e.g. USB FlashDrive and/or network connection: Handles data exchange with an external PC/PG.

Note

When you load user data with a USB FlashDrive, the data size must not be larger than 4 GB!

SINUMERIK Operate under Windows

You require the following software if you use SINUMERIK Operate under Windows:

- PCU 50.5: WinXP PCU basic software as of V1.2
- PC: Windows XP SP3 / Windows 7 operating system

Additional programs

- TextPad text editor: Used for the external creation and editing of XML and INI files in Windows. The program is available on the Internet at: <http://www.textpad.com/>
- WinSCP: Enables protected data transfer between different computers (NCU - PC). The program is available on the Internet at: <http://winscp.net/eng/docs/lang:de>
- RCS Commander: Permits the remote maintenance for machines with the SINUMERIK Operate operating software from a standard Windows PC.
- As of V4.5, the RCS Commander provides a special conversion function to transfer alarm texts of HMI Advanced or HMI Embedded to SINUMERIK Operate.
- System Network Center (SNC): Sets up the Ethernet connection of one or more TCUs to NCU/PCU.

References

A detailed description of the structure and commissioning of the system network with controller and operator components is provided in the following description:

SINUMERIK 840D sl/840Di sl Operator Components and Networking

Principles of start-up

2.1 Data structure of SINUMERIK Operate

For SINUMERIK Operate, all files are saved on the CompactFlash card (NCU) or on the hard disk (PCU) or DVD (PC).

The file system is at the uppermost level:

- On the NCU in the "/System CF-Card/" directory
- On the PCU in the "F:\hmis\\" directory
- On the PC, the drive can be freely selected, e.g. "C:\Programs\siemens\sinumerik"

The directory structure comprises the following directories:

- addon
- compat (only PC/PCU)
- oem
- siemens
- user

These directories have essentially an identical structure.

Note

The files in the "siemens" directory represent the original status and cannot be modified!

Files, whose contents can be supplemented or modified, are available as templates (specimen files) in the following directories:

Files	Directory
Configuration files	/siemens/sinumerik/hmi/template/cfg
Text files	/siemens/sinumerik/hmi/template/lng

Before making any changes/supplements, these files should be copied into the /cfg or /lng in /**oem**/sinumerik/hmi/, /**addon**/sinumerik/hmi/ or /**user**/sinumerik/hmi/ directories.

Structure

The section of the directory structure relevant for SINUMERIK Operate is shown below:

siemens directory		
/siemens/sinumerik		
	/hmi	
	/appl	// applications (operating areas)
	/base	// basis system components

siemens directory		
	/cfg	// all configuration files
	/data	// version data
	/hlp	// online help files
	/hlps	// online help files, zipped and version files
	/ico	// symbol files
	/lco640	// icons in resolution 640x480
	/lco800	// icons in resolution 800x600
	/lco1024	// icons in resolution 1024x768
	/lco1280	// icons in resolution 1280x1024
	/lco1600	// icons in resolution 1600x1240
	/lng	// text files
	/lngs	// text files zipped and versions files
	/osal	
	/ace	// ACE/TAO
	/qt	// Qt
	/proj	// EasyScreen configuring
	/template	// various templates
	/cfg	// templates for configuration files
	/ing	// templates for text files
	/tmpp	// storage, temporary data
	/sys_cache/hmi	// various log files

compat directory		
/compat		// files for integrating the OEMFrame applications
	/add_on	
	/hmi_adv	
	/mmc2	
	/oem	
	/user	

addon directory		
/addon/sinumerik		
	/hmi	
	/appl	// applications (operating areas)
	/cfg	// configuration files
	/data	// version data
	/hlp	// online help files, zipped and version files
	/ico	// symbol files
	/lco640	// icons in resolution 640x480
	/lco800	// icons in resolution 800x600
	/lco1024	// icons in resolution 1024x768

addon directory		
	/ico1280	// icons in resolution 1280x1024
	/ico1600	// icons in resolution 1600x1240
	/lng	// text files
	/lngs	// text files zipped and versions files
	/proj	// EasyScreen configuring
	/template	// various templates

oem directory		
/oem/sinumerik		
	/data	// version data
	/archive	// manufacturer archive
	/hmi	
	/appl	// applications (operating areas)
	/cfg	// configuration files
	/data	// version data
	/hlp	// online help files
	/hlps	// online help files, zipped and version files
	/ico	// symbol files
	/ico640	// icons in resolution 640x480
	/ico800	// icons in resolution 800x600
	/ico1024	// icons in resolution 1024x768
	/ico1280	// icons in resolution 1280x1024
	/ico1600	// icons in resolution 1600x1240
	/lng	// text files
	/lngs	// text files zipped and versions files
	/proj	// EasyScreen configuring
	/template	// various templates

user directory		
/user/sinumerik		
	/data	// version data
	/archive	// user-specific archive
	/prog	// user-specific programs
	/hmi	
	/cfg	// configuration files
	/data	// version data
	/hlp	// online help files
	/ico	// symbol files
	/ico640	// icons in resolution 640x480
	/ico800	// icons in resolution 800x600
	/ico1024	// icons in resolution 1024x768

user directory		
	/ico1280	// icons in resolution 1280x1024
	/ico1600	// icons in resolution 1600x1240
	/lng	// text files
	/proj	// EasyScreen configuring
	/log	// log files
	/md	// machine data views
	/proj	// EasyScreen configuring

2.2 Editing files

2.2.1 Editing the file at the controller

You can use a copy of the sample file in order to make specific adjustments. Template files are available in the following directories:

`/siemens/sinumerik/hmi/template/cfg` and `/siemens/sinumerik/hmi/template/lng`

Save the sample file in the "user" or "oem" directory - in the appropriate folder.

Note

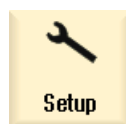
As soon as the file is in the user-specific directory, the entries of this file have priority over the Siemens file. If an entry is missing in a user-specific file, the corresponding entry from the Siemens file is used instead.

You need to restart the operating software for the settings made in the file to take effect.

Precondition

- If you copy files, you require access level 1 (manufacturer).
- When assigning a new file name, ensure that only files with a maximum name length of 49 characters can be managed.

Copying / pasting / opening a file



1. Select the "Start-up" operating area.
2. Press the "System data" softkey.
The data tree is displayed.
3. For instance, on the NCU directory, "System CF card", under "siemens" open the required directory (e.g. `/sinumerik/hmi/template/cfg`).



4. Position the cursor on the desired file.

5. Press the "Copy" softkey.



6. For instance, on the NCU in the "System CF card" directory, under "oem" or "user" open the required directory (e.g. /sinumerik/hmi/cfg), in which the copied file should be saved.

7. Press the "Paste" softkey.

If a file of the same name already exists, you receive a message. You can overwrite or rename the file.



8. Press the "OK" softkey.



9. You can open the selected file in the editor by pressing the "Open" softkey.

- OR -

Press the <INPUT> key.



- OR -

Double-click the highlighted file.

Rename file



1. Select the required file.

2. Press the ">>" and "Properties" softkeys.

The "Properties of..." window opens.

The following data is displayed:

- Path:
- Name:
- Creation date and time:
- Date of change and time:



3. To change the file name, place the cursor in the "Name" entry field and overwrite the name. Date of change and time are simultaneously updated.



4. Press the "OK" softkey to save the new name.

Cut-out/delete file



1. Select the required file.

2. Press the "Cut" softkey.

The file is copied into the buffer and at the same time deleted from the previous directory.

As the file is located in the buffer, you can also insert the file into another directory.



- OR -
2. Press the ">>" and "Delete" softkeys.
You receive an alarm and you can delete the file by pressing the "OK" softkey.
Press the "Cancel" softkey to cancel the delete operation.

Displaying the file in the preview window



1. Select the required file.
2. Press the ">>" and "Preview window" softkeys.
The preview window is displayed in the lower area of the window together with the file contents.

Press the "Preview window" softkey again to close the window.

2.2.2 Editing a file externally

To create or edit an XML file on an external PC with Windows, use a text editor that supports the required "UTF-8" coding, e.g. "TextPad".

This means that you can enter or directly insert all characters of the unicode system without rewriting them using the keyboard.

If files with a different coding are saved, the special characters are not correctly displayed in the editor. In SINUMERIK Operate, there is no automatic conversion into the UTF-8 coding!!

Saving the XML file in UTF-8 coding

1. Select the "Save As" dialog box.
2. Set the character set to "UTF-8".

Entering comments in an XML file

If you are entering comments to explain a program, you must keep the following in mind:

- A comment always begins with the sequence: <!--
- A comment ends with the character string: -->

Example

```
<!-- Work offset: -->
```

Note

In the comment itself, you can never use two minus signs one right after the other!

Special characters

If you use characters in the text where the syntax of XML is of significance, then these must specifically formatted:

Special characters	Notation
&	&
'	'
<	<
>	>
"	"

Comments in an ini file

If you enter a comment in an ini file, start the comment line with a semicolon.

Copy file

Using the "WinSCP" or "RCS Commander" program, copy the file from an external computer to the NCU via a network connection – or in the opposite direction from the NCU to an external PC.

For a detailed description, see the following sub-book:

References

NCU Operating System (IM7): Backing up and restoring data and WinSCP and PuTTY service tools

See also

Configuring alarm and message texts via alarm text files (Page 205)

General settings

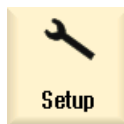
3.1 Changing the language

The operating software is available as standard in several languages.

Selecting a language

You can define the language selection mode using the following machine data.

MD9100 \$MM_CHANGE_LANGUAGE_MODE	Language selection mode
= 1	The user-interface language is specified via the "Language Selection" window (default value).



1. Select the "Start-up" operating area.



2. Press the "Change language" softkey.
The "Language Selection" window opens. The language most recently set is highlighted.
3. Switch the cursor to the required language.
4. Press the <INPUT> key.



- OR -

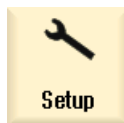


Press the "OK" softkey.

The operating software switches to the language selected.

Defining two languages

MD9100 \$MM_CHANGE_LANGUAGE_MODE	Language selection mode
= 2	Selection of two languages. During operation, the "Change language" softkey can be used to toggle between these languages.



1. Select the "Start-up" operating area.



2. Press the "HMI" softkey.



3. Press the ">>" softkey.

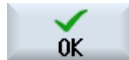


4. Press the "Language selection" softkey.
The "Language Selection" window opens.
All the installed languages are displayed in the fields "First language" and "Second language".



5. Select a language in each of these.
6. Press the <INPUT> key.

- OR -



Press the "OK" softkey.



During operation, you can switch between the two languages by pressing the "Change language" softkey.

3.2 Set date/time

You have the possibility of manually setting the date and time for the NCU (PLC and Linux) or PCU/PC (Windows). In so doing, you can select from various display formats.

Formats of the date display

Format	Example
d.M.yy	3.1.09
dd.MM.yy	03.01.09
dd.MM.yyyy	03.01.2009
d/M/yy	3/01/09
dd/MM/yy	03/01/09
dd/MM/yyyy	03/01/2009
M/d/yy	1/3/09
MM/dd/yy	01/03/09
MM/dd/yyyy	01/03/2009
yyyy/MM/dd	2009/01/03

The following applies:

Date	Description of the entry
d	Single or double digit day of the month (1 - 31).
dd	Double digit day of the month, with leading zero (01 - 31).
M	Single or double digit month of the year (1 - 12).
MM	Double digit month of the year, with leading zero (01 - 12).
yy	Double digit year (00 - 99).
yyyy	Four-digit year (e.g. 2009).

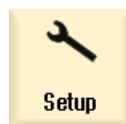
Time display formats

Format	Example	Hour system
H:m:s AP	1:2:4 PM	24 hour system, with display of "AM" / "PM"
h:m:s ap	1:2:8 pm	12 hour system, with display of "am" / "pm"
HH:mm:ss AP	13:02:08 PM	24 hour system, with display of "AM" / "PM"
hh:mm:ss ap	01:02:09 pm	12 hour system, with display of "am" / "pm"
H:mm:ss AP	1:02:09 PM	24 hour system, with display of "AM" / "PM"
h:mm:ss ap	1:02:09 pm	12 hour system, with display of "am" / "pm"
h:mm:ss	13:02:09	24 hour system
hh:mm:ss	13:12:04	24 hour system

The following applies:

Time symbol:	Description of the entry
h	Single or double digit hour (0 - 23 or 1 - 12 for am/pm).
hh	Double digit hour, with leading zero (00 - 23 or 01 - 12 for am/pm).
H	Single or double digit hour (0 - 23 for AM/PM).
HH	Double digit hour, with leading zero (00 - 23 for AM/PM).
m	Single or double digit minute (0 - 59).
mm	Double digit minute, with leading zero (00 - 59).
s	Single or double digit second (0 - 59).
ss	Double digit second, with leading zero (00 - 59).
AP	Display of the time of day: AM or am = before midday or PM or pm = after midday

Procedure



1. Select the "Start-up" operating area.



2. Press the "HMI" softkey.



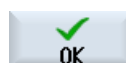
3. Press the "Date/Time" softkey.
The "Set Date and Time" window opens.



4. Select the required formats for the date and time with the <SELECT> key in the "Format" field.

5. For SINUMERIK Operate in the NCU: In the entry fields after "new" enter the actual date and time.

For SINUMERIK Operate on PCU/PC: In the entry fields after "new NCU + PCU" enter the actual date and time.



6. Confirm the entry with the "OK" softkey.

The new date and time details are accepted and output in the fields "actual NCU" and "actual PCU".

Note

You require administrator rights to set the date and time on the PCU/PC.

3.3 Configuring Caps Lock

The "CAPSLOCK on" function ensures that text entered via an external keyboard is always entered in upper case and not in lower case.

The key behavior is set using the following display machine data.

Setting

MD9009 \$MM_KEYBOARD_STATE	
= 0	CAPSLOCK off (default value)
= 2	CAPSLOCK on

If the machine data is changed, the system must be rebooted.

3.4 Activating/deactivating the virtual keyboard

The virtual keyboard is configured in the file "slguiconfig.ini".

Procedure

1. Copy the "slguiconfig.ini" file from the folder
/siemens/sinumerik/hmi/template/cfg.
2. Paste the copy of the file into the following directory:
/oem/sinumerik/hmi/cfg or /user/sinumerik/hmi/cfg.
3. Open the file in the editor.
4. To activate or deactivate the keyboard, make the following settings:
 - Activating the virtual keyboard:
In the section [TouchPanel], EnableTouch = true
In the section [Keyboard], EnableVirtualKeyBoard = true
 - Deactivating the virtual keyboard:
In the section [TouchPanel], EnableTouch = true
In the section [Keyboard], EnableVirtualKeyBoard = false
5. Double click in the input field to display the keyboard.

3.5 Defining the access rights for the "HMI restart" softkey

Displaying the machine data

Using the following display machine data, set the access rights from which access level the "HMI restart" softkey is displayed.

For SINUMERIK Operate on PC or PCU, the "EXIT" softkey is displayed.

MD9110_\$MM_ACCESS_HMI_EXIT	Access level of the "EXIT" softkey
= 1	Access level 1 (manufacturer), default value

3.6 Setting the screensaver

Defining the time until the screen is darkened (screensaver)

The duration (in minutes), after which the screen is automatically darkened if no keyboard key is pressed, can be specified using the display machine data MD9006 \$MM_DISPLAY_SWITCH_OFF_INTERVAL.

NOTICE
Influence of screen darkening on the service life
To extend its service life, the recommended duration for darkening a screen is 60 minutes. If you significantly reduce the duration, i.e. the screen is switched dark too often, this tends to shorten the screen service life.

Interface signal

The screen is darkened by setting an interface signal.

DB 19 byte	Signals to the operator panel front (PLC → HMI)	
DBB0	Bit 1	Bit 0
	Darken screen	Brighten screen

The following applies when actively darkening the screen using the interface signal:

- It is no longer possible to light up the screen again from the keyboard.
- The first keystroke on the operator panel already triggers an operator action.

Note

In order to prevent accidental operator actions via the interface signal when the screen is darkened, we recommend disabling the keyboard at the same time.

3.7 Inserting a user-specific run up screen

OEM-specific run up screen

You can replace the Siemens run up screen by your own run up screen.

Procedure:

1. Call your own run up screen "splash.png".

Note

The file name must not contain any lower case letters.

2. Save your own run up screen in one of the following directories:
/user/sinumerik/hmi/ico/ico640 or
/oem/sinumerik/hmi/ico/ico640 or
/addon/sinumerik/hmi/ico/ico640

Note

The size of the screen is of no significance. A resolution of 640x480 pixels is always set.

3.8 Access levels

3.8.1 Definition of access levels

Access to programs, data and functions is protected in a user-oriented hierarchical system of seven access levels. These are divided into

- Three password levels for manufacturer, service and users
- Four keyswitch positions for users

There are access levels 1 to 7 (see table below); where

- 1 is the highest and
- 7 is the lowest level.

Table 3-1 Access level concept

Access level	Protected by	Area
1	Password: SUNRISE (default value)	Manufacturer
2	Password: EVENING (default value)	Service
3	Password: CUSTOMER (default value)	User
4	Keyswitch 3	Programmer, machine setter
5	Keyswitch 2	Qualified operator
6	Keyswitch 1	Trained operator
7	Keyswitch 0	Semi-skilled operator

3.8.2 Modifying the access levels password

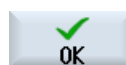
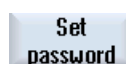
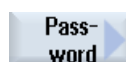
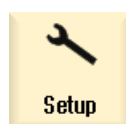
Overview

You can activate the access levels protected with passwords from the user interface.

The following options are available for editing the passwords:

- Set new password
- Change password
- Delete password

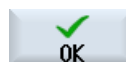
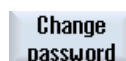
Setting the password



1. Select the "Start-up" operating area.
2. Press the "Password" softkey.
3. Press the "Set password" softkey.
The "Set Password" window opens.
4. Enter one of the possible default passwords and press the "OK" softkey to confirm the entry.
A permissible password is set and the valid access level is displayed.
Invalid passwords will be rejected.

Change password

You must change the standard passwords to obtain a secure access protection.



1. Press the "Change password" softkey.
The "Change Password" window opens.
2. Select the area (e.g. user) for which you want to assign a new password.
3. Enter the new password in the input fields "New password" and "Repeat password".
4. Press the "OK" softkey to confirm your entry.
A new, valid password is only accepted if the two terms entered are identical.

Delete password



Press the "Delete password" softkey. The access authorization is reset.

Note

Access authorization is not automatically deleted at POWER ON.

When a general NC reset is performed, the passwords are reset to the default passwords.

3.8.3 Access levels for programs

Objective

Access levels can be allocated to protect information; they can also be used to prevent unauthorized operation or deliberate damage to the control system or machine.

With access authorization levels to files and directories, it can be defined as to which users are authorized to execute which operations on a directory or a file in a file system. Operations in this sense include:

- Reading the contents of a file (READ), copying the file or placing the file in an archive.
- Changing the contents of a file (WRITE) or deleting the file.
- Executing a file (EXECUTE).
- Listing and displaying a file in a directory (SHOW).
- Creating a file in a directory.
- Displaying or reading the contents of a directory.
- Creating or deleting a directory.

Description of functions

The **individual allocation of access rights** acts, in the passive NC file system, on the following directories:

- Part programs
- Subprograms
- Manufacturer cycle directory
- User cycle directory
- Workpieces directory
- Workpiece directories

Further, the function acts on the files of the main program type (*.mpf) and subprograms or cycle (*.spf).

In addition, the technique can be expanded to include the following directories on the local drive:

- Part programs
- Subprograms

- Workpieces directory
- Workpiece directories

Note**Working via remote access**

When working via remote access (e.g. WinSCP or RCS Commander) the same access rights apply for editing or saving files and directories as directly at the controller.

Access rights via machine data

The following machine data is still available to **globally allocate access rights**:
MD51050 \$MNS_ACCESS_WRITE_PROGRAM

Default setting = 4 (access level: User)

Note**Effect of access rights**

The more restrictive setting is always effective both when individual access rights have been allocated and when settings have been made via
MD51050 \$MNS_ACCESS_WRITE_PROGRAM.

Settings in the slfsfileattributes.ini file

The slfsfileattributes.ini file is available as a template in the following directory:

../siemens/sinumerik/hmi/template/cfg

The values from 0 ... 7 correspond to the access levels that, for example, also apply to the password. The 4-digit numbers designate the sequence according to the following rights:

Significance from left to right: R W X S		
R	READ	Reading files and directories
W	WRITE	Changing, deleting or creating files and directories
X	EXECUTE	Executes program: For files, for directories, always 7
S	SHOW	Outputs a list of files and directories

Copy the slfsfileattributes.ini file into one of the following directories:

../oem/sinumerik/hmi/cfg or ../user/sinumerik/hmi/cfg

The operating software must be restarted in order that the changes become effective.

Example

The paths are composed of the data type identifiers. In the following example, "\\wks.dir*.wpd *.mpf" comprises the data type for workpiece directories "wks.dir" the workpieces "*.wpd" and the part programs "*.mpf".

For all workpieces, 7777 is defined as default setting for new part programs.

All entries are in the section [ACCESSMASKS] and have the following form:

```
[ACCESSMASKS]

\wks.dir\*.wpd\*.mpf = 7777
\mpf.dir\*.mpf = 6577
\cus.dir\*.spf = 6577
```

See also

Definition of access levels (Page 30)

3.8.4 Access rights for files

Changing access rights

Access rights for execution, writing, listing and reading are displayed in the "Properties" window.

- Execute: is used for the selection for execution.
- Write: controls the changing and deletion of a file or a directory.

For NC files, you have the option to set the access rights from keyswitch 0 to the current access level, to be set separately for each file.

If an access level is higher than the current access level, it cannot be changed.

For external files (e.g. on a local drive), the access rights are displayed to you only if settings have been executed for these files by the machine manufacturer. They cannot be changed via the "Properties" window.

Note

Settings for the access rights to directories and files

Via the configuration file and MD51050, access rights of the directories and file types of the NC memory and user memory (local drive) can be changed and pre-assigned.

Configuration file slfsfileattributes.ini

The access rights defined in the configuration file slfsfileattributes.ini and in MD51050 become valid in the NC when creating new files and can then be changed individually in the Program Manager with the "Properties" function. The access rights are an individual attribute of each file in the NC.

The access rights defined in the configuration file slfsfileattributes.ini and MD51050 become valid for external data on the CompactFlash Card of the NCU, or on the hard disk of the PCU/PC immediately for all the specified directories and file types, and cannot be changed through operation. The access rights can only be set globally for external data.

3.8.5 Access levels for softkeys

Use

The display and operation of softkeys can be suppressed by both the OEM as well as the user. This allows the operating software to be specifically adapted to the required functional scope and therefore be configured as transparently as possible.

This means that the functional scope of the system is restricted, to prevent access to functions in the operating software, or to restrict the possibility of operator errors.

Access hierarchy

A new access level can only be allocated for those softkeys that are also visible and can be used in the actual access level.

Example: If a softkey is only accessible with the "Manufacturer" access level, then the access level cannot be changed by an operator with access level "Key-operated switch setting 3".

Restrictions

In order to guarantee that the system remains stable and operable, certain softkeys are excluded from the access level change, e.g. the "Customize softkeys" softkey itself, in order that the undo path is not cut-off or the softkey "<< Back" to the vertical softkey bar.

The following softkeys cannot be hidden or the access level changed:

- All "OK" softkeys
- All "Cancel" softkeys
- All "Accept" softkeys
- All "<<" Back and ">>" Continue softkeys

The access level for softkeys in the cycle dialog boxes can also be changed; this does not apply to softkeys in Easy Screen dialog boxes.

Note

Software upgrade

As all changes of the access levels for softkeys are described using the softkey ID, they still remain valid after a software upgrade, which may have resulted in softkeys being shifted.

Softkeys, where access is already protected by machine data - for example MD51045 \$MNS_ACCESS_TEACH_IN - are only protected by the allocated access level after a new one has been allocated. The machine data is no longer evaluated.

All of the softkeys hidden using this mechanism can be temporarily displayed again using the access rights specified by the system. This is done using the "Show all softkeys" softkey. The softkey is deactivated as long as no softkeys have been allocated new access levels: This setting remains effective until the system is rebooted.

3.8.6 This is how you define new access levels for softkeys

General sequence

In order to assign a new access level to a softkey, proceed as follows:

- Activate the adaptation mode.
- Change the access level of a softkey.
- Accept the changes and confirm.

Preconditions

Access to the "Customize softkeys" is protected using a general MD51073 \$MNS_ACCESS_SET_SOFTKEY_ACCESS. This machine data is pre-assigned access level 3, i.e. the function is available from "Service" password and higher. In order that an operator with "User" access level can use this function, this machine data should be set to a value > 3.

An external mouse is required in order to make changes.

Changing the access level of a softkey

Procedure:

1. In the "Start-up" operating area, select the "HMI" menu.
2. On the expansion bar ">>" select the "Customize softkeys" softkey.
You can obtain precise information about the procedure in the following dialog.
1. Confirm with "OK", in order to activate the "Customize the softkey access levels active" mode. A table is displayed.
2. Select an operating area.
3. Place the mouse cursor on a softkey and using the right-hand mouse key call the context menu "Define access level", in order to define a new access level for this softkey.
 - The displayed access level is the default setting.
 - A softkey is practically made invisible with "Always hide".
 - You can reset the additionally allocated access level using "Original setting". This selection is only listed after an individual access level was set.
4. In order to accept changes, return to the "Start-up" operating area and there, press the "Close customizing" softkey. This means that you exit the "Customize the softkey access levels active" mode.
If you acknowledge the following query with "OK", then the changes are accepted and become effective once the system has rebooted.
5. If the changes to a softkey are not to be applied, select the line and press the "Original setting" softkey. This means that in the "New access level" column, the "Original setting" identifier is entered, which functions in the same way as deletion. This entry is no longer available following a system restart. "

Example

The following access levels were changed:

Softkey	New access level
SISuMainScreenMenuHu::0::3	Manufacturer
SIMaJogMillMenuHU::jogHuMore::6	User
SISuPaUserDataMenu:::4	Service

Once selected, the identifier and position of the softkey are transferred to the "Softkey" column.

Result

When the changes have been accepted, a configuration file with the "slsoftkeyaccess.xml" name is created and is saved in the file system under the following path: **../user/sinumerik/hmi/cfg**. Files can also be saved in the **/oem** path, which are evaluated when the system boots. The following priority applies: oem < user.

Note

Editing the file "slsoftkeyaccess.xml"

In order to avoid errors in the xml syntax and for the softkey identifier the "slsoftkeyaccess.xml" file may only be edited via the operating software as described above, and not directly in the file system under "System data". It is possible that the complete file or individual entries for softkeys can no longer be interpreted.

3.9 Program selection

3.9.1 Displaying the softkey

The softkeys on the user interface for calling the program lists are activated via general configuration machine data.

Configuring a softkey

MD51041 \$MNS_ENABLE_PROGLIST_USER	
0	No softkey
1	The "Prog. list" softkey is displayed

MD51043 \$MNS_ENABLE_PROGLIST_MANUFACT	
0	No softkey
1	The "MANUFACTURER" softkey is displayed

Precondition

To create program lists, you require the following authorization:

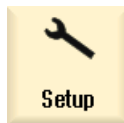
- "Program manager" operating area: Access level 3 (password: User) or
- "Start-up" operating area: Access level 1 (password: Manufacturer).

Program lists

You can create a program list by combining desired programs from the NC data, e.g. workpieces, part programs and subprograms. These programs can then be selected by the PLC for execution.

Program lists	Directory
plc_proglst_user.ppl	/user/sinumerik/hmi/plc/programlist
plc_proglst_manufacturer.ppl	/oem/sinumerik/hmi/plc/programlist

Procedure



1. Select the "Start-up" operating area if you have the authorization for access level 1 (password: Manufacturer).

- OR -



Select the "Program manager" operating area if you have the authorization for access level 3 (password: User).



2. Press the menu forward key and the "Prog. list" softkey.
The "Program List" window opens.



3. Press the "Manufacturer" softkey if you want to create a program list that can be used for all machines of a certain series.



4. Place the cursor in the desired line (program number).

5. Press the "Select program" softkey.

The "Programs" window opens. The data tree of the NC memory with the directories "Workpieces", "Part programs" and "Subprograms" is displayed.



6. Place the cursor on the desired program and press the "OK" softkey.
The selected program is inserted in the first line of the list together with its path.
- OR -
Enter the program name directly in the list.
If you are making entries manually, check that the path is correct (e.g. // NC/WKS.DIR/MEINPROGRAMM.WPD/MEINPROGRAMM.MPF).
If required, the extension (.MPF) is added.

Note:

A syntax check is not performed.

With workpieces, make sure that there is a main program or a job list with the same name in the workpiece itself.



7. To remove a program from the list, place the cursor on the appropriate line and press the "Delete" softkey.

- OR -



To delete all programs from the program list, press the "Delete all" softkey.

Licensing

Licenses are required for the activated options. When you purchase licenses you receive a "license key" that contains all options requiring a license and which is only valid for your CompactFlash card.

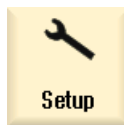
Requirement

If you wish to set or reset the options for a machine, you require at least access level 3 (user).
You can only license options for a machine from this access level and higher.

General information

Serial number of the Com-Permanently defined serial number of the CompactFlash card	
Hardware type	Data of the control
Machine name/No.	Displays the data that is saved in machine data MD17400 \$MN_OEM_GLOBAL_INFO
License key	Only with access level 1 (manufacturer) can a display field be edited Information about options that require a license

Procedure:



1. Select the "Start-up" operating area.



2. Press the menu forward key.



3. Press the "Licenses" softkey.
The "Licensing" window appears and you can select the following with the vertical softkeys:
 - Determining the license requirement
 - Overview
 - All options
 - Missing licenses
 - Activating the test license
 - Exporting licensing requirements
 - Entering or reading in the "License key"
 - Entering the lock file

4.1 Determining the license requirement

License database

The license is purchased from the Web License Manager in the Internet, administered by Siemens Industry.

For a detailed description, please refer to the following documentation:

References

CNC Commissioning Manual: NCK, PLC, drive: Licensing

Deactivate options

When importing old archives from other HMI systems, it can occur that options are set which are no longer required in the actual system – however, these are subject to the license check:

Order number	Option
6FC5800-0AP04-0YB0	Machining step programming
6FC5800-0AP11-0YB0	Manual machine
6FC5800-0AP15-0YB0	ShopTurn HMI/ShopMill HMI for SINUMERIK 840Di sl incl. HMI-Advanced
6FC5800-0AP20-0YB0	3D simulation of the finished part
6FC5800-0AP21-0YB0	Simulation, milling (2D dynamic; 3D static)
6FC5800-0AP23-0YB0	Simultaneous recording, ShopMill (real-time simulation)
6FC5800-0AP24-0YB0	Simultaneous recording, ShopTurn (real-time simulation)

4.1 Determining the license requirement

Procedure



1. Press the "All options" softkey to list all the options that can be selected for this controller.

- OR -



1. Press the "Missing lic./opt." softkey.

You receive an overview of all options that

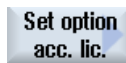
- are not adequately licensed
- are licensed, but which were not set

2. Activate or deactivate the required options in the "Set" column:

- Activate checkbox or
- Enter the number of options

Options shown red are activated, however, not yet licensed or not licensed in adequate number.

- OR -



Press the "Set option according to license" softkey to activate all of the options contained in the license key.

You obtain a confirmation prompt that you must confirm with "OK".



3. To activate new selected options, press the "Reset (po)" softkey. A safety prompt appears.

For several options, you will have to also restart SINUMERIK Operate. Corresponding prompts will appear in the user response line.



4. Press the "OK" softkey to trigger a warm restart.

- OR -



Press the "Cancel" softkey to cancel the process.

Searching for options

You have the possibility of finding options using the search function.

1. The window "Licensing: All Options" or "Licensing: Non-licensed Options" is open.



2. Press the "Search" softkey and enter the required option in the search form.

The cursor is positioned on the first entry that corresponds to the search term.



3. Press the "Continue search" softkey if the term that was found is not the option that you were looking for.

Additional search option



Press the "Go to beginning" softkey to start the search at the first option.



Press the "Go to end" softkey to start the search at the last option.

Note

Use of non-licensed options

Before licensing options, you can also activate them temporarily without the license keys and use them for test purposes. In this case, the control displays alarm 8080/8081 periodically, indicating that the option used has not yet been assigned the required license. Further, in this state, the NC cannot be started.

4.2 Activating the test license

If you wish to use options without any restrictions – and without purchasing a license – you can activate these for a limited time using the test license. Six test license periods are available. The first test license period has a longer operating time. The operating time depends on the NCU being used. Generally, the additional test license periods are 150 operating hours of the control system.

Duration of the first test license period

NCU	Operating time duration
710.3	1000 h
720.3	2000 h
730.3	3000 h

Requirement

If you wish to activate the test license, you require access level 2 (service).

Procedure



1. Press the "Activate test license" softkey.

A message informs you of the remaining number of test license periods.



2. Press the "OK" softkey.

The test license period is now active.

The active test license periods and the remaining operating time are now displayed.

Note

In the message line, the control displays the operating time still available for the current test license period every hour.

Note

Interruption of the test license

A test license period is interrupted in the following cases:

- If no axis is operated actively, e.g. during simulation mode.
- If all non-licensed options are reset.
The test license continues when a non-licensed option is set again.

Note**End of the test license**

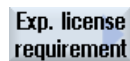
A test license period is terminated in the following cases:

- If the time has expired.
- If a new valid license key is entered.

No further test license can be activated on the NCU if all test license periods have expired.

4.3 Exporting the license requirement

After you have determined the license requirement, you can export the data via the set-up external storage locations.

Procedure

1. Press the "Export license requirement" softkey.
A small window opens listing the set-up storage locations.
2. Select "USB" for example, if you export the licenses on a USB FlashDrive.
3. Press the "OK" softkey.
The set options, the options which are not set but are licensed options, the license requirement and the license key are stored in an Alm file. In addition, an Internet connection is established for this Web License Manager.

A detailed description of the license management is provided in the following documentation:

References

CNC Commissioning Manual: NCK, PLC, drive: Licensing

4.4 Reading in the license key

The newly purchased license key can be loaded to the controller as follows:

- Enter license key manually
- Read in license key electronically

Procedure

1. Press the "Licenses" softkey.

- OR -

4.4 Reading in the license key



Press the "Overview" softkey.

The "Licensing" window opens.



2. If you obtain the license key via the Web License Manager, enter the license key manually in the field "You can enter a new license key here".
3. Press the <INPUT> key.

If the license key is valid, the message "License key set" is displayed.

- OR -



If you have the appropriate license key in the form of an alm file, then press the "Read-in license key" softkey.

...

Open the appropriate storage location (e.g. USB) with "OK".



The matching .alm file is detected automatically and the license key is read in.

Note:

If several matching .alm files are in the directory, they will be offered to you in a subsequent selection dialog box.

Select the desired file.

Creating drive connections

Up to 21 connections to so-called logical drives (data carriers) can be configured. These drives can be accessed in the "Program manager" and "Startup" operating areas.

The following logical drives can be set up:

- USB interface
- CompactFlash card of the NCU, only for SINUMERIK Operate in the NCU
- Network drives
- Local hard disk of the PCU, only for SINUMERIK Operate on the PCU



Software option

In order to use the CompactFlash card as data carrier, you require the "Additional HMI user memory on CF card of the NCU" option (not for SINUMERIK Operate on PCU or PC).

Note

The USB interfaces of the NCU are not available for SINUMERIK Operate and therefore cannot be configured.

5.1 Setting up drives

The "Set up drives" window is available in the "Start-up" operating area for configuring the softkeys in the Program Manager.

Information on the procedure and configuration can be found in the following documentation:

References

Operating Manual, Universal/Turning/Milling/Grinding, Chapter "Managing programs" > "Setting up drives"

File

The created configuration data is stored in the "logdrive.ini" file. This file is located in the /user/sinumerik/hmi/cfg directory.

General information

Note

We do not recommend that a USB front interface is permitted for part program execution using EES if a system has several local USB front interfaces (several TCUs, the use of a HT8, etc.). The connection to the USB FlashDrive can be interrupted as a result of the TCU switchover. This setting should be made using the checkbox "Use this drive for program execution with EES".

Note

If a global USB front interface should be used via PCU, then the PCU must be configured as server via the VNC connection, see Chapter "Configuring VNC connections (Page 301)".

5.2 Setting EXTCALL

Using the EXTCALL command, a program on an external program memory can be called from a part program.

EXTCALL must be switched off, when in certain configurations two SINUMERIK Operate are connected to one NCU.

Activating the function

Processing EXTCALL calls is switched-on or switched-off using the following display machine data.

MD9106 \$MM_SERVE_EXTCALL_PROGRAMS	
= 0	HMI ignores EXTCALL instructions and selection via PLC
= 1	HMI processes EXTCALL instructions and selection via PLC (default value)
= 2	HMI processes EXTCALL instructions and ignores selection via PLC
= 3	HMI ignores EXTCALL instructions and processes selection via PLC

Note

A main program selected from an external program memory is selected again automatically after a power-on if the same program memory is still available and execution of EXTCALL calls has been activated in MD9106.

5.3 Setting the EES

The EES (Execution from External Storage) function allows part programs for drives enabled for EES to be processed directly without reload buffer. The behavior is the same as that for processing from the NC part program memory without the restrictions that apply to "processing external".



Software option

In order to use this function, you require the "Execution from External Storage (EES)" software option.

Note

The EES function makes EXTCALL calls superfluous. If required, existing EXTCALL calls can be converted. EXTCALL calls without conversion are also executed for an active EES, although still as EXTCALL.

Diagnostics

The EES mode is displayed using the following values to check the current EES setting:

MD18045 \$MN_EES_MODE_INFO		Mode, in which the EES function operates.
Bit 0 = 1	Local EES active (LOCAL_DRIVE of the NCU available as the only EES drive) ^{*)}	
Bit 1 = 1	Global EES active (all available drives (USB, incl. local EES, when present))	
Bit 2 = 1	A global part program memory has been created in an external memory. With the search path, the search is first performed for programs in the file system of the NC and then in the corresponding directories in an external memory.	

^{*)} If you also possess the "Additional HMI user storage on the CF card of the NCU" option, you can use the EES for the user storage of the NCU. MD10123 EES_LOCAL_ENABLE is required for activation.

A detailed description of the EES function is provided in the following documentation:

References

Function Manual, Basic Functions: Mode group, channel, program operation, reset response (K1)

Chapter "Execution from External Storage (EES)"

5.4 Creating a global part program memory

With the global part program memory (GDIR), a structure is provided analog to the passive file system of the NC. It is integrated in the search path of the NC for subprogram calls. If you create a GDIR, all participants in the system have access to the same logical drive. The participants can directly execute part programs from the common program memory. The GDIR so replaces or extends the part program memory of the NC.

Note

The global part program memory plays a role only for the EES function. The creation of a GDIR is, however, not essential for the EES operation.

Possible procedure

1. Declare the global part program memory in the "Creating drives" window
2. Shift the content of the local program memory with SPF, MPF and WCS directory to an external drive, e.g. to a USB-FlashDrive.
3. Copy the directories that have been backed up to the global part program memory.

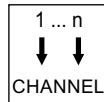
Additional information about managing the program memory is available in the following documentation:

References

Work Preparation Programming Manual: File and Program Management
Chapter "Program memory"

Channel menu

6.1 Configuring a channel menu



The channel menu is activated for display via the channel switchover key.

The channel menu is used to switchover the NC channel displayed in the SINUMERIK Operate or more precisely at the operator station. If a channel menu has been configured, when the channel switchover key is pressed, the channel menu for the channel switchover is displayed to make the appropriate operator action. If there is no channel menu, then the channel switchover key results in the HMI display advancing to the next NC channel.

Configuring

The configuration is made in the "netnames.ini" file for the operating software, as well as in the particular "config.ini" files for the individual operator stations.

To configure the data for the operator stations (entries in the "config.ini" files), instead of directly editing in the "config.ini" file, we recommend that you use the "System Network Center (SNC)" program.

The "netnames.ini" file must be generated using an editor.

Storage

The "netnames.ini" file is stored for SINUMERIK Operate in the following directory:
/oem/sinumerik/hmi/cfg or /user/sinumerik/hmi/cfg

The "config.ini" file is located in the relevant directory:

- On the NCU:
/user/common/tcu/<TCU name>/common/tcu/config.ini
- On the PCU:
F:\user_base\common\tcu\<TCU name>\common\tcu\config.ini

For <TCU name>, the TCU name of the operator station should be used that was assigned at the system when a TCU powered-up for the first time. If you work with the "System Network Center (SNC)" tool, then these file paths are of no significance.

Using the channel menu

- For a system comprising several operator stations and several NCUs within the framework of operator unit management, if the NC channels of several NCKs are operated from one operator station.
- For a simple system with many NC channels, as an alternative to the standard mechanism, the channel switchover can be made using the channel switchover key.

6.3 General structure of the channel menu

The basic philosophy of the channel menu is to provide - in a selection menu - the operator with a view of the NC channels of all NCUs that are to be operated with the channel menu. In conjunction with this, a logical grouping/structuring is introduced using the so-called channel groups. The desired NC channel can be directly selected using the selection menu. Without a channel menu, when pressing the channel switchover key, the NC channels must be scrolled through consecutively until the desired NC channel has been selected. An operator station can only be switched over across NCUs using the channel menu.

6.2 Constraints

- Only one SINUMERIK Operate on a PCU/ PC may be permanently connected to one NCK. It is not possible to switch over to another NCK.
- SINUMERIK Operate of an NCU can only operate with its own NCK and cannot be switched over to another NCU.

6.3 General structure of the channel menu

Overview

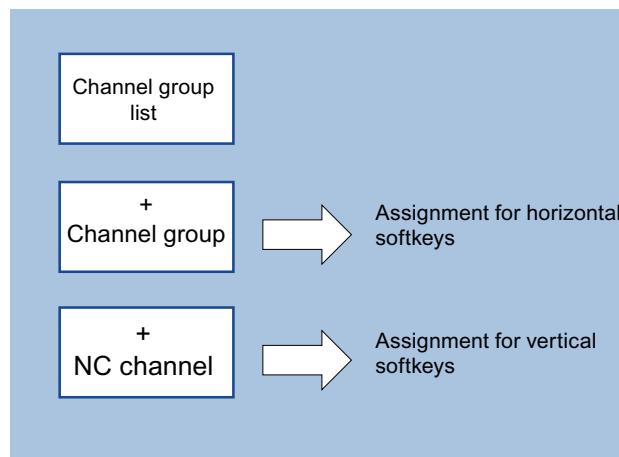
For the channel menu, a logical view is defined across all of the channels of all NCKs involved; this logical view is structured in channel groups and this in turn in specific NC channels.

A channel group list is defined for the channel menu in order to be able to switch over operation. A channel group list comprises one or several channel groups. In turn, a channel group comprises one or several NC channels.

In the channel menu, the channel groups can be selected using the horizontal softkeys.

The channels of a selected channel group can be selected with the vertical softkeys.

A certain NC channel of a specific NCU is always a switchover target that can be selected. A switchover to another channel triggered via the channel menu can implicitly mean switchover to another NCU.

**Note**

In a system, for example, a channel group can correspond to a machining station, the channels contained in it, machining units.

Up to 32 channel groups each with up to 8 channels can be configured in a channel menu.

Configuring the total structure

For a system, the total structure of the channel menu is configured in the "netnames.ini" file. The configuring in the "netnames.ini" file includes all switchover targets that are used at the system. This configuration must be saved or distributed in a standard form across all SINUMERIK Operate systems that provide the channel menu.

An individual channel menu for the operating station is configured for each operating station in the particular "config.ini" file.

Two-stage procedure

1. In the first stage, all channel groups that are used in the system are defined in the "netnames.ini" file, irrespective of the actual operating stations.
2. In the second stage, the version of the channel menu for the specific operating station is defined in the "config.ini" files belonging to the station.

6.4 Defining channel groups in the "netnames.ini" file**Defining channel groups**

All channel groups that are used at the system are defined in the "netnames.ini" file. Independent of the specific operator stations.

A separate section [CH_GRP_x] ($1 \leq x \leq 32$) should be created for every channel group.

6.4 Defining channel groups in the "netnames.ini" file

The entries required for a channel group are as follows:

- The text to be displayed in the channel for the channel group at the corresponding horizontal softkey.
- The list of jump destinations summarized in this channel group, i.e. NC channels. The jump destinations are defined in a consecutively numbered list with the following entries:
 - Member y .Target: Jump destination
 - Member y .Text: Softkey text to be displayed in the channel menu ($1 \leq y \leq 8$).

"netnames.ini" file

A channel group, e.g. channel group 1 is therefore specified as follows:

```
[CH_GRP_1]                ; Name of the channel group; the notation
                           ; must follow this scheme. The groups are
                           ; consecutively numbered at the last
                           ; position.

Text = Process1            ; Softkey text for the channel group for
                           ; display in the channel menu (horizontal
                           ; softkey) switchover designation 1 of the
                           ; channel group.

Member1.Target =           ; Computer name (host name) or the IP
[<HMI>.]<NC>.#<channel    ; address of the computer on which the HMI
number>                   ; to be selected runs (NCU or PCU):
                           ; <HMI> = <IP address or name of the HMI
                           ; host (PC/PCU)>
                           ; <HMI> can be omitted if the HMI runs
                           ; locally on the NCU
                           ; <NC> = <IP address or name of the NC>
                           ; Alternatively, the machine name can also
                           ; be specified from the mmc.ini.
                           ; Using ".#Channel number", optionally the
                           ; NC channel can be specified that should
                           ; be taken when switching over.
                           ;

Member<No>.Text = NCU730_K1 ; Softkey text for the display in the
                           ; channel menu (vertical softkey). If, for
                           ; "Target" a channel number is specified,
                           ; then instead of a text, using the fixed
                           ; value "$MC_CHAN_NAME", it can also be
                           ; defined that the text to be displayed
                           ; should be the channel name defined in the
                           ; NCK machine data.
                           ;

Member2.Target =           ;
Member2.Text              ; Same as for the vertical softkey 2
```

6.5 Configuring operating stations in the "config.ini" file

The version of the channel menu is individually defined for the operator stations in the "config.ini" files belonging to the station. The data defined in the "netnames.ini" file is used. Here, the channel groups are assigned a softkey in the channel menu.

"config.ini" file

- The data is defined in section [T2M2N].
- The references CH_GRP_xx of the softkeys refer to the channel groups which are defined in the "netnames.ini" file. This notation is binding.
- SKx in this case, designates the xth horizontal softkey in the channel menu.

Example

```
...  
[T2M2N]  
SK1=CH_GRP_1  
SK2=CH_GRP_3  
SK8=CH_GRP_15
```

"System Network Center" tool (SNC)

The operator stations can be configured in a transparent way using various windows with the aid of the "SNC" tool.

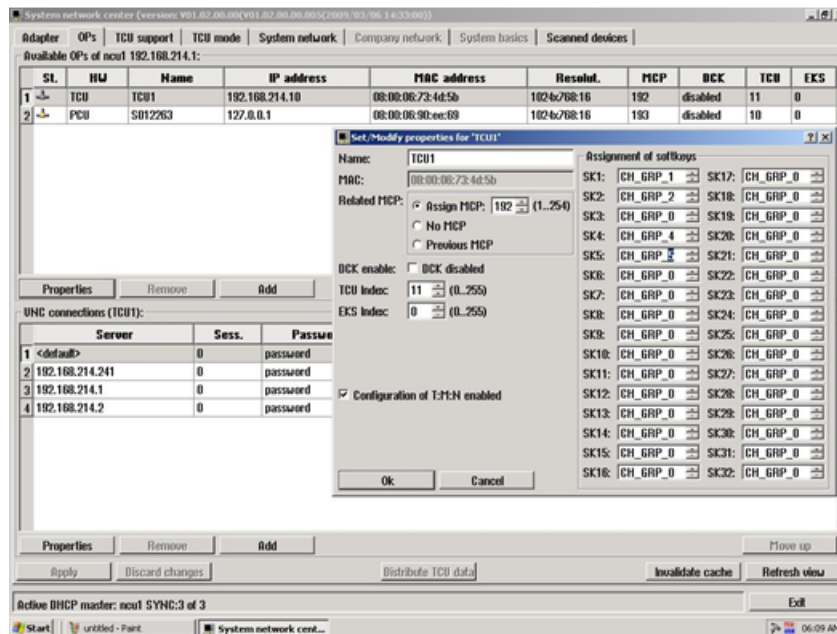
Calling SNC

- For the NCU, the program is also included on the CompactFlash card and is called with "sc start snc" using the service command.
- For the PCU, the program is on the hard disk and can be started from the user interface using an icon.

6.5 Configuring operating stations in the "config.ini" file

Configuring with SNC

1. The operator stations are listed via the "OPs" tab.
2. The "Set/Modify properties for "TCU"" is opened after selecting an OP and pressing the "Properties" button.
3. The softkey assignment is displayed after selecting the checkbox "Configuration of T:M:N enabled". The desired channel group can now be assigned to the softkey for this operator station.



Note

The PCU must not change the NCK with which it is connected during the switchover. With the switchover, it can only initiate a channel switchover on this NCU.

Note

The section [T2M2N] can also be created in the "netnames.ini" file instead of in the "config.ini" file. This setting then applies for all TCUs that are not configured via config.ini. For SINUMERIK Operate on PC, this procedure is required because it is not possible to access the "config.ini" file.

6.6 Example: Settings for 1:N

1:N configuration

For a 1:N configuration the following requirements apply:

- A PCU with SINUMERIK Operate can be connected to a maximum of four NCUs.
- SINUMERIK Operate must be switched off on all NCUs.
- The following configuration files must be adapted:
 siemens/sinumerik/hmi/template/cfg/**netnames.ini**
 siemens/sinumerik/hmi/template/cfg/**mmc.ini**

This is illustrated in the following sections using an example with 1 PCU and 2 NCUs.

Example mmc.ini

```
[Global]
NcddeMachineNames=net, NCU840D_1, NCU840D_2
NcddeDefaultMachineName=NCU840D_1

[NCU840D_1]
ADDRESS0=192.168.214.1,LINE=14,NAME=/NC,SAP=040d,PROFILE=CLT1__CP_L4_INT
ADDRESS1=192.168.214.1,LINE=14,NAME=/PLC,SAP=0202,PROFILE=CLT1__CP_L4_INT
ADDRESS2=192.168.214.1,LINE=14,NAME=/CP,SAP=0502,PROFILE=CLT1__CP_L4_INT
ADDRESS10=192.168.214.1,LINE=14,NAME=/DRIVE_00_000,SAP=0201,
SUBNET=0046-00000000:000,PROFILE=CLT1__CP_L4_INT
ADDRESS11=192.168.214.1,LINE=14,NAME=/DRIVE_03_003,SAP=0900,
PROFILE=CLT1__CP_L4_INT

[NCU840D_2]
ADDRESS0=192.168.214.2,LINE=14,NAME=/NC,SAP=040d,
PROFILE=CLT1__CP_L4_INT
ADDRESS1=192.168.214.2,LINE=14,NAME=/PLC,SAP=0202,
PROFILE=CLT1__CP_L4_INT
ADDRESS2=192.168.214.2,LINE=14,NAME=/CP,SAP=0502,
PROFILE=CLT1__CP_L4_INT
ADDRESS10=192.168.214.2,LINE=14,NAME=/DRIVE_00_000,SAP=0201,
SUBNET=0000-00000000:000,PROFILE=CLT1__CP_L4_INT
ADDRESS11=192.168.214.2,LINE=14,NAME=/DRIVE_03_003,SAP=0900,
PROFILE=CLT1__CP_L4_INT
```

Example netnames.ini

```
;*****
;      netnames.ini      - EXAMPLE -
;
; This is an example of a netnames.ini for 1:N NCU-switching in
; a configuration of 3 NCU and a PCU50.
```

6.6 Example: Settings for 1:N

```

;*****
[own]
owner = HMI

;*****
; ChanMenu
;*****
[chan HMI]
ShowChanMenu = true

;*****
;Channel-groups
;Correlation from channel-group to horizontal softkeys
; -> see config.ini of the TCU
;The section [T2M2N] can also be created in the netnames.ini file
;instead of config.ini file. This configuration applies to all TCU,
;which are not configured by their config.ini.
;*****
[T2M2N]
SK1 = CH_GRP_1
SK2 = CH_GRP_2

[CH_GRP_1]

Text = 1. NCU
Member1.Target = 192.168.214.241.192.168.214.1.#1
Member1.Text = NCU1%nKanal 1

Member2.Target = 192.168.214.241.192.168.214.1.#2
Member2.Text = NCU1%nKanal 2

[CH_GRP_2]
Text = 2. NCU
Member1.Target = 192.168.214.241.192.168.214.2.#1
Member1.Text = NCU2%nKanal 1

Member2.Target = 192.168.214.241.192.168.214.2.#2
Member2.Text = NCU2%nKanal 2

```

Note

For switching over an MCP with the operating software, the PLC block FB9 must not be used (FB9 must not be active). If necessary, the MCP must be switched over in the PLC user program.

Solution

In the OB100 (call for FB1), set MCP1BusAdr to 255 and MCP1Stop to "true".

In the OB1, scan the "MMCBTSSready" (DB10.DBX108.3) signal. If this is set, write the MCP address from DB19.DBB123 (MCP index) to the DB7.DBB27 before removing the MCP stop bit DB7.DBX62.1. The MCP is now active.

If the "MMCBTSSready" (DB10.DBX108.3) is removed again (operating software switches to another NCU), the MCP stop bit DB7.DBX62.1 will have to be reset.

6.7 Distribution via job lists in the case of 1:N

Simplified selection of programs

The function is only available in a group of several NCUs and must be enabled in the "systemconfiguration.ini" configuration file:

```
[miscellaneous]
distributeJoblist=true
```

The sources are located on the local drive of the PCU or the PC.

The distribution of the programs is done by selecting a job list which is located in a workpiece. The job list may contain COPY and SELECT instructions:

```
; Copy A.MPF to NCU1:
COPY A.MPF NC=NCU1

; Copy B.MPF to NCU2:
COPY B.MPF NC=NCU2

; Copy all subprograms to all NCUs:
COPY *.SPF *

; Select A.MPF on NCU1 in channel 1.
SELECT A.MPF ch=NCU1,1

; Select B.MPF on NCU2 in channel 2.
SELECT B.MPF ch=NCU2,2
```

Sequence:

1. If a distributed job list is active during the selection, all distributed programs on the NCU are deleted without confirmation. If the workpiece on the NCUs is empty, it will be deleted there.
2. The programs are copied to the corresponding workpieces of the relevant NCU.
3. Existing programs are overwritten without any message.
4. Following a change in a distributed program on the local drive, the modified programs using the Editor are copied to the corresponding NCU.

Sample structure

/user/sinumerik/data/ prog/ JOBA.WPD	NCU1	NCU2
JOBA.JOB	WKS.DIR	WKS.DIR
A.MPF	WKS.DIR/JOBA.WPD	WKS.DIR/JOBA.WPD
B.MPF	WKS.DIR/JOBA.WPD/B.MPF	WKS.DIR/JOBA.WPD/B.MPF
UP1.SPF	JOBA.WPD/UP1.SPF	JOBA.WPD/UP1.SPF
UP2.SPF	WKS.DIR/JOBA.WPD/UP2.SPF	WKS.DIR/JOBA.WPD/UP2.SPF

/user/sinumerik/data/ prog/ wks.dir/JOBA.WPD	NCU1	NCU2
JOBA.JOB	WKS.DIR	WKS.DIR
A.MPF	WKS.DIR/JOBA.WPD	WKS.DIR/JOBA.WPD
B.MPF	WKS.DIR/JOBA.WPD/B.MPF	WKS.DIR/JOBA.WPD/B.MPF
UP1.SPF	JOBA.WPD/UP1.SPF	JOBA.WPD/UP1.SPF
UP2.SPF	WKS.DIR/JOBA.WPD/UP2.SPF	WKS.DIR/JOBA.WPD/UP2.SPF

Supplementary conditions

Special behavior:

- Distributed programs are retained during a restart of the operating software.
- On the local drive, distributed programs can neither be deleted nor renamed nor overwritten using copy/move.
- Changes of distributed programs on the NCU are lost, if they are not manually saved.
- The program correction in the machine is not offered.
- When creating a commissioning archive, all data of the local drive is also automatically saved.

Restrictions:

- In the event of changes of distributed programs on the local drive using the Editor, it should be noted that only the channel states are monitored on the NCU currently connected. If the program is still interrupted or active on another NCU, it cannot be correctly distributed after the change.
- The display of the active block, the marking of the block search as well as the read-only handling of the program opened in the Editor are executed only for the active programs of the currently connected NCU.
- In the Editor, programs are always opened in the context of the currently connected NCU. This can lead to a faulty step recognition with distributed programs.
- In the current block display in the "Machine" operating area, you can display only the active/ distributed programs of the currently connected NCU.
- The simulation of job lists via several machines is not possible.
- If `distributeJoblist=true` is configured in the `systemconfiguration.ini` file, then part or subprograms can no longer be selected via the "Program Manager" operating area.

Machine and setting data

7.1 Machine and setting data

Overview

Prefix	Number ranges	Machine/setting data	Storage in range (softkey)
\$MM	9000 - 9999	Display machine data	Display MD
\$MNS	51000 - 51299	General configuration machine data	General MD
	51300 - 51999	General cycle machine data	
\$MCS	52000 - 52299	Channel-specific configuration machine data	Channel MD
	52300 - 52999	Channel-specific cycle machine data	
\$MAS	53000 - 53299	Axis-specific configuration machine data	Axis MD
	53300 - 53999	Axis-specific cycle machine data	
\$SNS	54000 - 54299	General configuration setting data	General SD
	54300 - 54999	General cycle setting data	
\$SCS	55000 - 55299	Channel-specific configuration setting data	Channel SD
	55300 - 55999	Channel-specific cycle setting data	
\$SAS	56000 - 56299	Axis-specific configuration setting data	Axis SD
	56300 - 56999	Axis-specific cycle setting data	

References

You can find a detailed description of the machine and setting data in the following documentation:

- List Manual (book 1)

7.2 Displaying/editing machine data

You can access the following machine data in the "Start-up" operating area.

- General machine data (\$MN) and general configuration machine data (\$MNS)
- Channel-specific machine data (\$MC) and channel-specific configuration machine data (\$MCS)
- Axis-specific machine data (\$MA) and axis-specific configuration machine data (\$MAS)
- Drive-specific machine data (\$M_)
- Machine data control unit: Drive parameters
- Machine data feed-in: Drive parameters

- I/O component parameters: Drive parameters
- Communication parameters (CULINK, DMC): Drive parameters

Note

The display of the machine data / drive parameters depends on the configuration of your control.

Access authorization to the machine data operating area can be controlled by keyswitch position or password.

Read-access to machine data is possible as of access level 4 (keyswitch 3).

Machine data can be changed with access level 1 (manufacturer).



WARNING

Incorrect parameterization

Changes in the machine data have a considerable influence on the machine. Incorrect configuration of the parameters can endanger human life and cause damage to the machine.

Information about machine data

The following information is displayed from left to right:

- Machine data number, with array index if applicable
- Machine data name
- Machine data value
- Machine data unit
- Effective

Note

If the machine data does not use units, the corresponding column is empty. If the data is not available, the hash symbol "#" is displayed instead of the value. If the value ends in an "H", it is a hexadecimal value.

The physical units of machine data are displayed on the right-hand side of the input field.

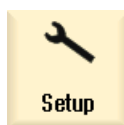
For each machine data item, an activation type can be read in the column on the right.

so Immediately active: No action required
cf Configuration: "Set MD to active (cf)" softkey
re Reset: <Reset> key on the machine control panel
po POWER ON: "Reset(po)" softkey

Additional Information

In the footer you are given a brief explanation of the selected machine or setting data item.

Procedure



1. Select the "Start-up" operating area.



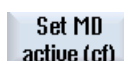
2. Press the "Mach. data" softkey. Additional softkeys for the individual machine data areas are displayed.



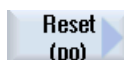
3. Press the "General MD", "Channel MD", or "Axis MD" softkey. The window opens and the selected machine data is displayed.



4. Place the cursor on a machine data item.
5. Position the cursor on the entry to be modified and enter the required value.
6. Depending on the activation type, there are two ways of activating the settings:
Press the "Activate MD (cf)" softkey. The value is applied.

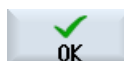


- OR -



Press the "Reset (po)" softkey.

A safety prompt appears.



7. Press the "OK" softkey to trigger a warm restart.

- OR -



Press the "Cancel" softkey if you do not want to apply the settings.

Axis selection

If several components are available for the area selected (e.g. "Axis MD"), you can choose from the following options:



1. Press the "Axis +" or "Axis -" softkey. The values of the next (+) and the previous (-) axes are displayed.



- OR -



Press the "Direct selection..." softkey.

The "Axis Direct Selection" window opens.

2. Select the required axis directly from those available in the drop-down list box.



3. Press the "OK" softkey to save the setting.

- OR -

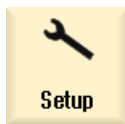


- Press the "Cancel" softkey to reject the settings.

7.3 Displaying/editing display machine data

You can access the display machine data via the user interface in the "Start-up" operating area.

Procedure



1. Select the "Start-up" operating area.



2. Press the "Mach. data" softkey.



3. Press the menu forward key.
Further softkeys are displayed.

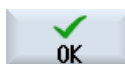


4. Press the "Display MD" softkey.
The display machine data appears.

5. Position the cursor on the entry to be modified and enter the required value.



6. Press the "Reset (po)" softkey.
A safety prompt appears.



7. Press the "OK" softkey to trigger a warm restart.

- OR -



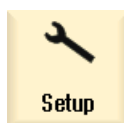
- Press the "Cancel" softkey if you do not want to apply the settings.

7.4 Displaying/editing setting data

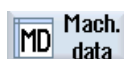
You can access the following setting data via the user interface in the "Startup" operating area.

- General setting data
- Channel-specific setting data
- Axis-specific setting data

Procedure



1. Select the "Start-up" operating area.



2. Press the "Mach. data" softkey.



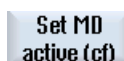
3. Press the menu forward key.
The softkeys "General SD", "Channel SD", and "Axis SD" are displayed.



4. Press the relevant softkey to display a particular setting data area.
The window opens and the selected setting data is displayed.



5. Place the cursor on the desired setting data and position the cursor on the entry to be changed. Enter the desired value.
6. Depending on the activation type, there are two ways of activating the settings:

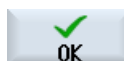


Press the "Activate MD (cf)" softkey.

- OR -



Press the "Reset (po)" softkey.
A safety prompt appears.



7. Press the "OK" softkey to trigger a warm restart.

- OR -



Press the "Cancel" softkey if you do not want to apply the settings.

Axis/channel selection

If several components are available for the area selected (e.g. "Axis SD" or "Channel SD"), you can choose from the following options:



1. Press the "Axis +" or "Axis -" or "Channel +" or "Channel -" softkey.
The values of the next (+) and the previous axis/channel (-) are displayed.



- OR -
Press the "Direct selection..." softkey.
The "Direct Selection" window opens.
2. Select the required axis directly from those available in the drop-down list box.
3. Press the "OK" softkey to save the setting.



- OR -
Press the "Cancel" softkey to reject the setting.

**7.5 Displaying/editing drive parameters**

The "Control Unit MD", "Infeed MD" and "Drive MD" softkeys as well as the "I/O comp. MD" and "Comm MD" on the menu forward key are available to display and edit drive parameters.

A separate list display is provided for each of these areas in which you can view and edit drive data.

Precondition

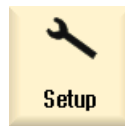
Settings for drive parameters are protected with access level 2 (service).
The lists of the drive parameters are visible with access level 4 (keyswitch 3).

Please refer to the following documentation for information on commissioning the drive:

References

CNC Commissioning Manual: NCK, PLC, drives

Procedure



1. Select the "Start-up" operating area.



2. Press the "Mach. data" softkey.
The "Control unit parameters", "Supply MD" and "Drive MD" softkeys are displayed.

3. Press the softkey of an area to show its parameters.

4. Place the cursor on the desired parameter.

5. Position the cursor on the entry to be modified and enter the required value.



6. In order to activate the settings, you have the option of saving or carrying out a reset.

Press the "Save/reset" softkey.



7. Press the "Save" softkey.

You are prompted "What do you wish to save?"

You can make the following selection per softkey:

- Actual drive object
- Drive unit
- Drive system

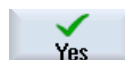
- OR -



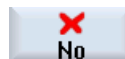
Press the "Reset (po)" softkey.

You are prompted as to whether you wish to initiate a warm restart for the NCK and the complete drive system.

Confirm the prompt with the "Yes" softkey.



Reject the warm restart with the "No" softkey.



Extended reset

If you wish to carry out a warm restart only for specific objects or specific areas, then additional softkeys are available for this purpose.



Press the "Extended reset" softkey.

You are prompted to enter for which objects you wish to initiate a warm restart.

You can make a selection with the following softkeys:



Only the NCK is restarted.

NCK and drive sys.	All of the control units and the NCK in the system are restarted. This softkey corresponds to the "Reset (po)" softkey if you confirm the prompt with "Yes".
NCK, HMI, drive sys.	All of the control units and the NCK in the system as well as the operator software are restarted.

Hiding the expanded softkey bar

The "Extended reset" softkey is defined in the "slsuconfig.ini" file. If the softkey is not to be displayed, proceed as follows:

1. Copy the "slsuconfig.ini" file from the **/siemens/sinumerik/hmi/template/cfg/** directory.
2. Insert the file into the following directory:
/user/sinumerik/hmi/cfg/
3. Open the file and in the section
[Softkeys] and enter
AdvancedReset = 0

7.6 Machine data display options

Through the use of the machine data display filter, it is possible to reduce the number of displayed machine data relating to a certain area, e.g. general machine data, for special purposes.

Machine data areas

Display filters are available for the following machine data areas:

- General machine data
- Channel-specific machine data
- Axis-specific machine data
- Drive machine data
- Display machine data

Filter options

It is possible to limit the machine data display using the following filtering methods:

- Only display axes that are used
- According to indices
- According to display groups
- Display expert parameters

Only display axes that are used

If this checkbox is activated, axes that are not used are hidden when scrolling.

Indices from ... to

The index filter refers to the machine data arrays. On the display, this machine data can be identified by the field index attached to the machine data string.

Example: 10000[index]AXCONF_MACHAX_NAME_TAB

If the index filter is activated, machine data arrays are only displayed in the specified index area.

Display groups

A display group contains machine data within a machine data area that belongs to the same topic.

Note

To find out which display group a machine data item belongs to, refer to the "Display filter" parameter associated with the description of the machine data element in question.

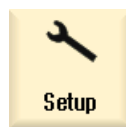
References: List Manual (book 1)

Select/deselect display groups to increase or decrease the number of machine data items displayed for the current machine data area.

Display expert parameters

If the "Display expert parameters" filter is disabled, only the machine data in a machine data area that is required for the basic functionality of the NC is displayed.

Procedure



1. Select the "Start-up" operating area.



2. Press the "Mach. data" softkey.



3. Press the softkey of a particular data area, e.g. "General MD".
The list of general machine data is displayed.



4. Press the ">>" and "Display options" softkeys.



The window "Display options: ..." for the selected machine data opens.



5. Select the required filter options (according to display groups, indices or expert parameters) by activating the checkboxes.

6. Press the "Select all" softkey to activate all display groups.

- OR -

Press the "Deselect all" softkey.

All checkmarks are removed and you can select individual display groups.

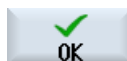
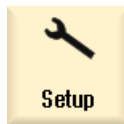
Press the "OK" softkey to save the settings.

7.7 Editing machine data and drive parameters

7.7.1 Editing hexadecimal values

You can use a bit editor to enter the required values for machine data in hexadecimal format.

Procedure



1. Select the "Start-up" operating area.

2. Press the "Mach. data" softkey and select the required machine data area (e.g. general machine data).

3. Position the cursor on the entry to be edited.

4. Press the <SELECT> or <INSERT> key.

The "Bit Editor" window opens.

All bits of the machine data item that have a description text are displayed.

5. If you activate the "Display all bits" checkbox, all bits regardless of their data type (8, 16 or 32 bits) are displayed. Activate the relevant bit.

6. Press the "OK" softkey. The setting is checked and an error message may be output.

The window closes and you return to the machine data overview.

The new value is displayed in hexadecimal format.

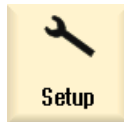
7.7.2 Editing BICO values

You can use the BICO editor to enter BICO values for drive parameters.

Note

BICO values can also be entered by copying and inserting or also directly entered the field.

Procedure



1. Select the "Start-up" operating area.



2. Press the "Mach. data" softkey.



3. Press the "Control unit MD" softkey and select a BICO parameter, e.g. "p738".



4. Position the cursor on the entry to be changed, e.g. BICO 63.2091.0

5. Press the <SELECT> or <INSERT> key.

The "BICO Editor" window opens.



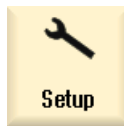
6. Complete the "Drive object number", "Parameter number", "Bit/Index" and "Parameter value (hex)" fields.

Note

The BICO editor also opens if you enter a value other than 0 or 1 for the value to be changed.

7.7.3 Editing enum values

For drive parameters, you have the option of editing enum elements (e.g. r2, p3, p4, p97 etc.) using a selection list.

Procedure

1. Select the "Start-up" operating area.
2. Press the "Mach. data" softkey.
3. Select the desired drive parameter area, e.g. "Control unit MD".
4. Select the desired parameter, e.g. "p3" and position the cursor at the entry to be changed, e.g. "[1] Display, Si...".
5. Press the <SELECT> or <INSERT> key.
A window containing a list of all defined enum values and their descriptions opens.
6. You can select a value from the list.
- OR
You can also directly enter the value as decimal number into the brackets [].

7.8 Searching for data

You can search for specific machine, setting and display machine data.

Search strategies

- When a digit or a digit with an index is entered, an exact search for this parameter is performed.
Example: If "9" is entered > p9 will be found (but not p99)
- When text is entered, a full-text search is performed: i.e. the term is sought in the relevant table and in the explanations in the footers.

Note**Search with place holders**

When searching for machine data, you have the option of using place holders:

- "*": Replaces any character string
 - "?": Replaces any character
-

Searched fields

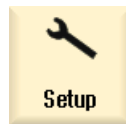
A search is made in the following fields of the machine data for the character string that has been entered if the "Additionally search in the explanation texts" option is not set:

- Machine data number
- Symbolic name
- Unit

The following fields are additionally searched if the "Additionally search in the explanation texts" option is set:

- Descriptive text
- Bit texts
- Enum texts (only for drives)

Procedure



1. Select the "Start-up" operating area.



2. Press the "Mach. data" softkey.



3. Select the required area by pressing the relevant softkey.
4. Press the "Search..." softkey and enter the required text or digit in the search screen.
5. If you wish to restrict the search only to the texts in the tables that are displayed, deactivate the "Search explanation texts too" checkbox.



6. Select the search direction required (forwards, backwards) in the "Direction" field.

The cursor is positioned on the first entry that corresponds to the search term.

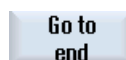


7. Press the "Continue search" softkey if this machine, setting or display machine data item is not the one you are looking for.

Additional search option



Press the "Go to start" softkey to start the search at the first data.



Press the "Go to end" softkey to start the search at the last data.

7.9 Managing data

Overview

The "Manage data" function is used to simplify the commissioning and permits the transfer, backup, loading and comparing of machine, setting, compensation and drive data at the file level. The files are saved in the .tea ASCII format.

The "Manage data" softkey is visible as of access level "Service" (= 2).

The following option fields are available in the "Transfer Machine Data" window:

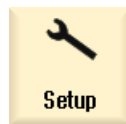
- Transfer data within the controller
Only those controller objects are listed that are currently available in the controller.
- Save data to a file
Additional files with the extensions .acx and .log are created for the SINAMICS parameters:
 - Files with the extension .acx contain the parameter descriptions for the selected drive object.
 - The log file may also contain the warnings and errors that occurred during the conversion from .acx to .tea.
- Load data from a file

Note

An NCK and drive reset is only triggered automatically for drive objects after loading. A reset must be triggered manually for all other controller objects.

- Compare data
Several files (.tea) and/or controller objects (axis, channel, servo, infeed, etc.) can be compared at the same time. An online comparison of constantly changing data is not possible because the values are saved temporarily at the time of the comparison. Up to eight elements can be taken into the list of the data to be compared.

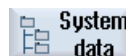
Procedure



1. Select the "Start-up" operating area.



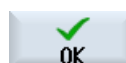
2. Press the "Mach. data" softkey.
The "Manage data" softkey is available in each machine data area, except in "User views".
- OR -



Press the "System data" softkey and select the HMI data/Data backups/IB data directory.



3. Press the "Manage data" softkey.
The "Transfer Machine Data" window opens.



4. Activate the required option filed and press "OK".

Transfer data within the controller

5. Position the cursor on a controller object and confirm with "OK".



6. In the window that opens, select the target controller object and confirm with "OK".

Save data to a file

5. Position the cursor on a controller object and confirm with "OK".

6. The "Save Under: Select Archive" window opens and the appropriate storage location is preselected.



If required, press the "New directory" softkey to generate a new subdirectory. Enter the required name in the "New Directory" window and confirm with "OK".



7. In the "New File" window, enter the required name for the file to be saved and confirm with "OK".

The stored files can be found in the data tree of the "System data" in the HMI data/Data backups/IB data directory.

Load data from a file

5. The "Load Trace: Select File" window opens.
Position the cursor on a .tea file that is located either in the IB data directory or on a USB-FlashDrive and confirm with "OK".

Please note: If the source and target drive object numbers differ when loading drive data, you must adapt individual parameters. To do this, press the "Adapt" softkey and edit the parameter values in the window that appears.

Compare data

5. In the "Compare Data - Controller" window, select an element to be compared from the tree structure of the control objects.

- OR -



Press the "File" softkey to select a .tea file to be compared from the tree structure of the files.



6. Press the "Add to list" softkey.

The list of the data to be compared is displayed in the lower part of the window and the selected element is added.



7. Press the "Delete from list" softkey to remove individual elements from the comparison list.

- OR -

Deactivate the checkbox of the selected element in the list of the data to be compared.



8. When at least two elements are listed, the "Compare" softkey is enabled. Press this softkey to perform the comparison.

7.10 User views

Use

In the "User Views" window you can create and adapt all machine data relevant to a particular functionality. Machine data is individually collated to simplify the user's task.

You can include additional comments when creating and/or editing user views.

Note

Importing user views (compatibility)

If you have already created user views in HMI-Advanced, copy the *.klb files to `/user/sinumerik/hmi/template/user_views` to use them in SINUMERIK Operate.

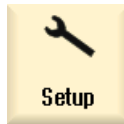
Display in the the "Parameter" operating area

Proceed as follows to show user views also in the "Parameters" operating area:

1. Select the "Start-up" operating area.
2. Press the "Machine data" softkey.
3. Press the "User views" softkey.
4. Press the "Manage view" softkey.
5. Press the "New view" softkey.
6. Activate the checkbox "Display parameter / use setting data", if you also wish to display the new user view in the "Parameter" → "Setting data" → "Data lists".
7. Enter a name for the user view.
8. Confirm with "OK".

7.10.1 Creating a user view

Procedure



1. Select the "Start-up" operating area and press the "Mach. data" softkey.



2. Press the softkeys "User views", "Manage view" and

"New view".

The "New View" window opens.

3. Enter the desired view name.

Press the "OK" softkey.

If a user view already exists with the same name, you receive a safety prompt as to whether the existing view should be overwritten.



4. Press the "Enter data" softkey.

The "Insert data" selection box opens.



5. Open the selection list with the <INSERT> key, select a machine data area and press the <INPUT> key.



Select a machine data item with the cursor keys or using the "Find" softkey.



6. Press the "Insert before line" or "Insert after line" softkey to insert the selected machine data at the required position in the user view.

Note:

You can navigate in the open user view with the mouse and select a line without closing the list box.

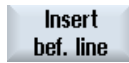


7. Press the "Back" softkey to close the list box and return to the user view you want to edit.

Your entries are automatically stored.



8. Press the "Insert text" softkey to insert any text in the open user view. The "Insert Text" input window opens.



9. Enter a text and explanatory description for the text and press the "Insert before line" or "Insert after line" softkey to accept the text.



10. Press the "Back" softkey to save your entries and return to the user view you are editing.

Selecting a particular machine data item



You can use the Search dialog box to look for a particular data item.

7.10.2 Editing the user view

Depending on the row you have selected, you can use the "Properties" softkey to change comments and, in the case of machine data, the data source (channel, axis, drive unit) and view the description and, if required, change.

- Machine data

Depending on the data type you can select either a fixed or a variable data source for machine data.

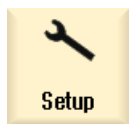
The entry "variable(*)" means that when you can select another data source (channel, axis, drive unit) with vertical softkeys 1- 3 (+, - and direct selection) the value displayed changes to refer to the data source currently selected.

Alternately, you can select a fixed data source so that the displayed value does not change when you select a different data source.

- Comment texts

Text and description

Procedure



1. Select the "Start-up" operating area and press the "Mach. data" softkey.



2. Press the "User views" softkey. The "User Views" window opens.

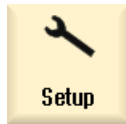
	3. Press the "Edit view" softkey. Additional softkeys to edit the user view are listed.
	4. Press the "Insert data" softkey if you wish to enter another data item.
	5. Select a machine data item with the cursor keys.
	
	6. Press the "Up" or "Down" softkeys to move the selected row.
	
	- OR - Press the "Delete line" softkey to remove the selected line from the view. The data item is removed without a prompt.
	- OR - Press the "Properties" softkey to view the texts for comments and, if required, change. The "Properties" window is opened. For machine data, depending on the setting, the description or the machine data is displayed in the lower left-hand window.
	7. Press the "Back" softkey to save your changes.

Selecting a particular machine data item

	You can use the Search dialog box to look for a particular data item.
	Press the "Go to start" softkey to start the search at the first entry.
	Press the "Go to end" softkey to start the search at the last entry.
	Press the "Continue search" softkey if the data found during the search does not match up with what you are looking for.

7.10.3 Deleting a user view

Procedure



1. Select the "Start-up" operating area.



2. press the "Machine data" softkey.



3. Press the "User views" softkey.



4. Press the "Manage view" softkey.



5. Select the desired view from the selection list.

6. Press the "Delete view" softkey.



With the final confirmation prompt you will be requested to confirm with either "Yes" or cancel with "No".

or



The "User views" dialog is then redisplayed.

Just like before, as an alternative, it is possible to delete a user view under "System data" under the following path: `../user/sinumerik/hmi/template/user_views`

7.11 Plain texts for machine and setting data

Application-specific and language-dependent plain texts can be configured for machine and setting data depending on the version [Index].

Displaying the plain texts

You set the display of the plain texts using the following display machine data:

MD9900 \$MM_MD_TEXT_SWITCH	Plain texts instead of MD identifier
= 1	Plain text is displayed in the table
= 0	Plain text is displayed in the dialog line

Files

File	Machine/setting data
nctea_XXX.txt	General machine/setting data
chtea_XXX.txt	Channel-specific machine/setting data
axtea_XXX.txt	Axis-specific machine/setting data
hmitea_XXX.txt	Display machine data

For "XXX" the specified language code must be used, e.g. "deu" for German, "eng" for English, etc.

Supported languages (Page 584)

Ensure that the file names are written in lower case letters.

Procedure

1. Depending on the machine or setting data for which you wish to change the plain text, create the corresponding file, e.g. "nctea_XXX.txt".
2. Save the file with the UTF-8 coding in order that umlauts and special characters are correctly displayed in SINUMERIK Operate.
3. Place the file in SINUMERIK Operate in the following directory:
System CF card/**user**/sinumerik/hmi/Ing or
System CF card/**oem**/sinumerik/hmi/Ing or
System CF card/**addon**/sinumerik/hmi/Ing

Note

The files are searched through in the sequence of the directories "user", "oem" and "addon". The first file that is found is evaluated. If, e.g. text files are available in the "user" and "oem" directories, the files are evaluated in the "user" directory.

Note

If a machine data item has different indices and you do not specify an index for the plain text, the same plain text appears for all the indices of the machine data item.

The parameters and indices must be sorted in an ascending order.

Example of a file with German texts

```
14510 Mein Integer-Text für PLC_MD 14510
14510[1] Mein Integer-Text für MD 14510[1]
14512[1] Mein Text für HEX-Datum 14512[1]
```

Result in the display

The plain text is displayed in the table or in the diagnostics line in the operating area "Startup", "General MD".

Machine data	Text
14510	My integer text for PLC_MD 14510
14510[1]	My integer text for MD 14510[1]
14512[1]	My text in HEX data 14512[1]

Customizing the "Machine" operating area

Automatic changeover

The circumstances under which an automatic changeover is made into the "Machine" operating area are set in MD51040 \$MNS_SWITCH_TO_MACHINE_MASK.

MD51040 \$MNS_SWITCH_TO_MACHINE_MASK		0 (default setting)
Bit 0 = 1	For program selection in the Program Manager, an automatic change is not made into the Machine operating area.	
Bit 1 = 1	For program selection in the Program Manager, an automatic change is not made into the Machine operating area. When changing over the operating mode via the machine control panel, an automatic change is not made into the Machine operating area.	
Bit 2 = 1	For program selection in the Program operating area, an automatic change is not made into the Machine operating area.	
Bit 3 = 1	When selecting a program/executing a program in the Program operating area, the block search is not automatically started.	

8.1 Setting the font size of the actual value display

Setting font size

The actual values for the machine coordinate system or workpiece coordinate system can be displayed in two different font sizes in the "Machine" operating area.

Using a channel machine data item, you can set how many axes should be displayed in a larger font.

MD52010 \$MCS_DISP_NUM_AXIS_BIG_FONT		Number of actual values with a large font
= 3	Number of actual values with a larger font (default setting)	

8.2 Inserting a user-specific logo

OEM-specific logo

The Siemens logo is visible in the header of the "Machine" operating area.

8.3 Configuring the display of the G-code groups

You can replace the Siemens logo with your own logo.

Procedure:

1. Name your own logo "logo.png".

Note

The file name must not contain any lower case letters.

2. Save your own logo depending on the bitmap size, in one of the directories: **/oem/sinumerik/hmi/ico** or **/user/sinumerik/hmi/ico**

Directory	Bitmap size
/lco640	122 x 19
/lco800	160 x 24
/lco1024	199 x 30

8.3 Configuring the display of the G-code groups

The "G-functions" pane is visible in the "Machine" operating area.

Up to a maximum of 16 G-code groups are displayed in this pane. Configure the "slmagcodeconfig.ini" file accordingly to determine which G-code groups should be displayed, and where.

Configuring G-code groups

Section	Meaning
Selected G groups	This area is intended for operation of the controller in Siemens mode.
Selected G groups ISO mode	This area is intended for operation of the controller in ISO mode.

Enter the G-code group and position as follows:

SelectedGGroupN = M	
N	Position where G-code group M will be displayed. If you want to leave a position empty, omit the number (M), e.g. "SelectedGGroup1 = "
M	The number of the G-code group to be displayed at position N

Structure of the file

```
[Selected G-Groups]
SelectedGGroup1 = 1
SelectedGGroup2 = 2
SelectedGGroup3 = 3
```



```
SelectedGGroup4 = 4
SelectedGGroup5 = 5
SelectedGGroup6 = 6
SelectedGGroup7 = 7
SelectedGGroup8 = 8
SelectedGGroup9 = 9
SelectedGGroup10 = 11
SelectedGGroup12 = 12
SelectedGGroup13 = 13
SelectedGGroup14 = 14
SelectedGGroup15 = 15
SelectedGGroup16 = 16
[Selected G-Groups ISO-Model]
... (as with section for Siemens)
```

Procedure

1. Copy the "slmagcodeconfig.ini" file from the folder
/siemens/sinumerik/hmi/template/cfg.
2. Place the file in the following directory: /oem/sinumerik/hmi/cfg or /user/sinumerik/hmi/cfg.
As soon as the file is located in the user-specific directory, the entries for this file take precedence over the Siemens file. If an entry is missing in a user-specific file, the corresponding entry from the Siemens file is used instead.
3. Open the file and enter the number of the G code in question at the required position
"SelectedGGroup1 = 5".
I.e. the 5th G code group will be displayed at the 1st position.

8.4 Configuring the channel operational message

Overview

The channel operational messages are displayed in the third line of the header in the "Machine" operating area. There are two types of channel operational messages:

1. Channel operational messages that require an operator action to continue a program, e.g. feed enable missing, M01 active.
The channel operational messages are identified by exclamation marks.
2. Channel operational messages that do not require an operator action to continue a program. A stop condition is active for a length of time (>1 s) and is interrupted by the program processing, e.g. dwell time active, waiting for tool change.
The channel operational message is identified by a clock icon and disappears after a certain time.

File

The settings for this are specified in the "slmahdconfig.ini" file.

Configuring the channel operational message

Section	Meaning
CondStopIcons	Specification of an icon that is to be displayed with a certain channel operational message. If no icon is specified for a certain channel operational message, then the icon specified for "DefaultIcon" is displayed.
CondDelayTime	Specification of how long (in milliseconds) a certain message must be present before it is displayed. - If no time is specified for a certain channel operational message, then the time specified at "CondDelayTime" is used. - If the channel operational message is to be displayed immediately, "0" must be specified as time.

Procedure

1. Copy the "slmahdconfig.ini" file from the `/siemens/sinumerik/hmi/template/cfg` directory.
2. Place the file in the following directory: `/oem/sinumerik/hmi/cfg` or `/user/sinumerik/hmi/cfg`.
3. Open the file and make the required settings.

"slmahdconfig.ini" file

Standard configuration

```
[CondStopIcons]
Default icon = condwait.png
1 = condstop.png      ; No NC ready
2 = condstop.png      ; No mode group ready
3 = condstop.png      ; EMERGENCY STOP active
4 = condstop.png      ; Alarm with stop active
5 = condstop.png      ; M0/M1 active
6 = condstop.png      ; Block in SBL mode terminated
7 = condstop.png      ; NC stop active
8 = condstop.png      ; Read-in enable missing
9 = condstop.png      ; Feed enable missing
10 = condwait.png     ; Dwell time active
11 = condwait.png     ; Aux. function acknowledgment missing
12 = condstop.png     ; Axis enable missing
13 = condwait.png     ; Exact stop not reached
```

14 = condwait.png	; Waiting for positioning axis
15 = condwait.png	; Waiting for spindle
15 = condwait.png	; Waiting for other channel
17 = condstop.png	; Waiting for feedrate override
18 = condstop.png	; Error in NC block
19 = condstop.png	; Waiting for external NC blocks
20 = condwait.png	; Waiting for synchronized action
21 = condwait.png	; Block search active
22 = condstop.png	; Spindle enable missing
23 = condstop.png	; Axis feedrate override
24 = condwait.png	; Waiting for tool change acknowledgment
25 = condwait.png	; Gear step change
26 = condwait.png	; Waiting for position control
27 = condwait.png	; Waiting for thread cut
28 = condwait.png	; Reserved
29 = condwait.png	; Waiting for punching
30 = condwait.png	; Waiting for safe operation
31 = condwait.png	; No channel ready
32 = condstop.png	; Oscillation active
33 = condwait.png	; Axis replacement active
34 = condwait.png	; Axis container rotation
35 = condwait.png	; AXCT: Following axis active
36 = condwait.png	; AXCT: Leading axis active
37 = condwait.png	; AXCT: Follow-up active
38 = condwait.png	; AXCT: Internal status change
39 = condwait.png	; AXCT: Axis/spindle disable
40 = condwait.png	; AXCT: Corr. motion active
41 = condwait.png	; AXCT: Axis replacement active
42 = condwait.png	; AXCT: Axis interpolator active
43 = condwait.png	; Waiting for compile cycle
44 = condwait.png	; Access to system variable
45 = condstop.png	; Search target found
46 = condwait.png	; Rapid retraction started
47 = condwait.png	; AXCT: Wait for spindle stop
48 = condwait.png	; Machine data match
49 = condwait.png	; Axis replacement: Axis coupled
50 = condwait.png	; Axis replacement: Liftfast active
51 = condwait.png	; Axis replacement: New config active
52 = condwait.png	; Axis replacement: AXCTSW active
53 = condwait.png	; Axis replacement: Waitp active
54 = condwait.png	; Axis in another channel
55 = condwait.png	; Axis replacement: Axis is PLC axis

8.4 Configuring the channel operational message

```
56 = condwait.png      ; Axis replacement: Axis is oscillating axis
57 = condwait.png      ; Axis replacement: Axis is Jog axis
58 = condwait.png      ; Axis replacement: Command axis
58 = condwait.png      ; Axis replacement: Axis is OEM axis
60 = condwait.png      ; Coupled following axis
61 = condwait.png      ; Coupled-motion following axis
62 = condwait.png      ; Coupled slave axis
63 = condstop.png      ; Stop at cycle end M0
64 = condstop.png      ; Stop at cycle end M1
65 = condwait.png      ; Wait: Axis is at fixed stop
66 = condwait.png      ; Master-slave switchover active
67 = condwait.png      ; Axis replacement: Axis is single axis
68 = condstop.png      ; Stop: Target reached after block search
69 = condwait.png      ; Synchronism: Synchronous spindle
70 = condwait.png      ; Deactivation position synch. spindle
71 = condwait.png      ; Waiting for transformer axis enable
72 = condstop.png      ; Waiting because of possible collision
73 = condstop.png      ; Jog: Position reached
74 = condstop.png      ; Jog: Direction blocked
75 = condwait.png      ; Brake request
76 = condwait.png      ; Wait: G4 still: U
77 = condstop.png      ; Axial feed disable from PLC
78 = condstop.png      ; Waiting for axial feed enable
79 = condwait.png      ; Axial feed disable from SYNACT
80 = condwait.png      ; Waiting for master spindle speed
81 = condwait.png      ; Waiting for parameter set change
82 = condwait.png      ; Waiting for end of motion before
                        transformation change

83 = condwait.png
84 = condwait.png
85 = condwait.png
```

[CondDelayTime]

Default delay time = 1000

```
1 = 0      ; No NC ready
2 = 0      ; No mode group ready
3 = 0      ; EMERGENCY STOP active
4 = 0      ; Alarm with stop active
5 = 0      ; M0/M1 active
6 = 0      ; Block in SBL mode terminated
7 = 0      ; NC stop active
8 = 0      ; Read-in enable missing
```

```

9 = 0           ; Feed enable missing
10 = 0          ; Dwell time active
12 = 0          ; Axis enable missing
16 = 0          ; Waiting for other channel
17 = 0          ; Waiting for feedrate override
18 = 0          ; Error in NC block
19 = 0          ; Waiting for external NC blocks
22 = 0          ; Spindle enable missing
23 = 0          ; Axis feedrate override
32 = 0          ; Oscillation active
45 = 0          ; Search target found
76 = 0          ; Wait: G64 still U

```

8.5 Deactivating program test

There are several program test options to test and run-in a part program that are executed using the "Program Test" function.

This function is active as standard. You can select the program test in the "Machine" operating area, in the "Program control" window and using the "PRT" check box.

Deactivating the program test

You can deactivate the program test using the following configuration machine data. "PRT" is no longer displayed in the status line.

MD51039 \$MNS_PROGRAM_CONTROL_MODE_MASK		Options for machine - program control
Bit 0 = 0	Function program test not available	

The test option is described in the following documentation:

References

Function Manual, Basic Functions, Mode Group, Channel, Program Operation, Reset Behavior (K1)

8.6 Activating the "Teach In" function

Activate TEACH IN

The TEACH IN function is activated in part via the general configuration machine data.

Settings

MD51034 \$MNS_TEACH_MODE	
Bit 0 = 1	The "Teach progr." softkey is displayed. The approached position is accepted into the program with the "Accept" softkey, default value.
Bit 1 = 1	The acceptance of the teach-in block can be blocked by the PLC. DB19.DBX13.0 = 0 Block is accepted DB19.DBX13.0 = 1 Block is not accepted

Access to the function still depends on access rights.

MD51045 \$MNS_ACCESS_TEACH_IN		Protection level TEACH IN
= 7	The protection level for the execution of traversing movements to MDA buffer / part program for TEACH IN is specified, default value.	

MD51050 \$MNS_ACCESS_WRITE_PROGRAM		Protection level write part program
= 7	The protection level for writing part programs is specified, default value.	

Set the channel-specific machine data:

MD20700 \$MC_REFP_NC_START_LOCK		NC-Start disable without reference point
= 0		

Activating TEACH IN

1. Set the machine data and initiate a reset.
2. Now change into the "AUTO" and "MDA" operating mode.
3. Select the TEACH IN machine function.
4. Press the "Teach progr." softkey.

TEACH IN not active

If the "Prog. teach in" softkey is not active, the reasons can be as follows:

- The program is not in the stop, reset or interrupt state.
- Not all of the axes that require referencing are referenced.

Accept STAT and TU

If you wish to accept the joint position of the machine and the rotary axes in the traversing block, then activate the "STAT" and "TU" checkboxes.

In order that the checkboxes are displayed, set the following general configuration machine data:

MD51032 \$MNS_STAT_DISPLAY_BASE		Numerical basis for display of the joint position STAT
= 1	Display, joint position STAT	

MD51033 \$MNS_TU_DISPLAY_BASE	Numerical basis, display of the rotary axes TU position
= 1	Display of the rotary axes position TU

8.7 Block search

8.7.1 Activating block search mode

Block search mode settings

The following machine data can be used to set the possible variants for the block search with G-code programming:

MD51028 \$MNS_BLOCK_SEARCH_MODE_MASK	Bit mask for the available search modes
Bit 0 = 1	Block search with calculation, without approach
Bit 1 = 1	Block search with calculation, with approach
Bit 4 = 1	Block search without calculation
Bit 5 = 1	Block search with test run

The following machine data can be used to set the possible variants for the block search in a JobShop program (ShopMill, ShopTurn):

MD51024 \$MNS_BLOCK_SEARCH_MODE_MASK_JS	Bit mask for the available search modes (ShopMill, ShopTurn, single-channel)
Bit 0 = 1	Block search with calculation, without approach (default setting)
Bit 1 = 1	Block search with calculation, with approach Notice: In a ShopMill/ShopTurn program, the search is performed only on G code-blocks.
Bit 3 = 1	Skip EXTCALL programs
Bit 5 = 1	Block search with test run

Note

The "Block search mode" softkey is only displayed, if, in the machine data MD51024 or MD51028, more than one version for the block search has been defined; for example "Block search with calculation without approach" and "Block search with calculation with approach".

When so doing, take into account that the search versions set active in the "Block search mode" window are also referred to the program.

8.7.2 Accelerated block search for execution from external

The accelerated block search function allows you to run the part program forward until you reach the block you require.

Settings

If you activate the "Skip EXCALL" function, then you must set the following general configuration machine data:

MD51028 \$MNS_BLOCK_SEARCH_MODE_MASK	Bit mask for the available search modes
Bit 3 = 1 (33H to 3BH)	

For an external main program, a block search without calculation is always a fast block search. Only the blocks from the search position are loaded.

8.8 Multi-channel support

In the "Machine" operating area, a dual channel view is possible for any two channels (2 from n) of a mode group.

Using the multi-channel editor, for a multi-channel machine, you can open, change or generate programs simultaneously that are independent of one another from a time perspective. In this case, the programs involved are displayed next to one another in the editor.

Further, you have the option of simulating the programs before they are actually processed at your control.

In addition, you can align the programs according to synchronization points.

Precondition

- Multi-channel data only involve machines where the first technology is turning.
- For multi-channel machine, the following data must be the same for all channels:
 - Measurement unit
 - Work offset (e.g. G54)
 - Z value of the work offset (optional)
 - Blank
 - Speed limitation
- Further, the following JobShop-specific data must be the same for all channels:
 - Retraction plane (one each for the main and counterspindle)
 - Tailstock (one each for the main and counterspindle)
 - Tool change point (one each for the main and counterspindle)
 - Safety clearance
 - Climbing/ and conventional

Creating a multi-channel program

For multi-channel machining, you require a job list with the following data:

- Program names
- Program type
- G code
- ShopTurn program
- Channel assignment



Software option

You require the "ShopMill/ShopTurn" option to generate and edit ShopTurn/ShopMill programs.

After you have created a job list, in the "multi-channel data" parameter screen, you can combine the data from the G code and JobShop, which is then saved as cycle call in the job list. The "multi-channel data" parameter screen is displayed as soon as at least one ShopTurn program is contained in a job list.

Only programming G code programs

If the job list only contains G code programs, then you can deactivate the "multi-channel data" function. To do this, set the following machine data:

MD51228 \$MNS_FUNCTION_MASK_Tech		Cross-technology function mask
Bit 0	G code programming without multi-channel data	
= 1	Multi-channel data are not listed for job lists that only contain G code programs. The "multi-channel data" softkey is not displayed.	

8.9 Manual machine

General information

"Manual machine" allows you to work with large cycles in manual mode without having to write programs especially for this purpose.



Software option

You require the following software option in order to use this function:
"ShopMill/ShopTurn"

Activation

MD52212 \$MCS_FUNCTION_MASK_Tech		Cross-technology function mask
Bit 8	Manual machine (ShopMill/ShopTurn)	

Enables

MD52214 \$MCS_FUNCTION_MASK_MILL		Milling function screen
Bit 5	Enable spindle control, tool spindle	
= 0	via the NC/PLC interface	
= 1	via the user interface	
Bit 6	Enable spindle control, rotating spindle	
= 0	via the NC/PLC interface	
= 1	via the user interface	

MD52218 \$MCS_FUNCTION_MASK_TURN		Turning function screen
Bit 3	Enable spindle control, main spindle	
= 0	via the NC/PLC interface	
= 1	via the user interface	
Bit 4	Enable spindle control, counterspindle	
= 0	via the NC/PLC interface	
= 1	via the user interface	

Bit 5	Enable spindle control, tool spindle
= 0	via the NC/PLC interface
= 1	via the user interface

SD43200 \$SA_SPIND_S	Speed for spindle start using NC/PLC interface signal; value outputs using the operating software
SD43202 \$SA_SPIND_CONSTCUT_S	Cutting speed for spindle start using NC/PLC interface signal; value outputs using the operating software
SD43206 \$SA_SPIND_SPEED_TYPE	Spindle speed pipe for spindle start using NC/PLC interface signal; value outputs using the operating software

Additional parameter assignments

We recommend the following additional parameter assignments:

Handwheel	
MD11346 \$MN_HANDWH_TRUE_DISTANCE = 3	Handwheel path or velocity specification
MD11602 \$MN_ASUP_START_MASK bit 0, 3 = 1	Ignore stop conditions for ASUB
MD20150 \$MC_G_CODE_RESET_VALUE [7] = 2	Delete position of the G groups; run-up G54 active
MD20624 \$MC_HANDWH_CHAN_STOP_COND = 0xFFFF	Definition of the behavior of traveling with handwheel, channel-specific
MD32084 HANDWH_STOP_COND = 0x7FF	Behavior, handwheel travel
MD52212 FUNCTION_MASK_TECH Bit 9 = 1	Function screen form across technologies; work offset via softkey

Note

If, in machine data MD52212 FUNCTION_MASK_TECH bit 9 is set, the "WO selection" softkey is displayed in the "Work Offset – G54 ... G599" window.

Working area limits	
MD28600 \$MC_MM_NUM_WORKAREA_CS_GROUPS > 0	Number of coordinate system-specific working area limits
MD20150 \$MC_GCODE_RESET_VALUES[59] = 2	Delete position of the G groups; activate working area limits of the group

Note

If \$MC_GCODE_RESET_VALUES[59] = 2 is set in the MD20150 machine data, the "Limit stops" softkey is displayed in the "Manual Machine" window. This requires that \$MC_MM_NUM_WORKAREA_CS_GROUPS also has the value 1 for MD28600.

Feedrate	
SD42600 \$SC_JOG_FEED_PER_REV_SOURCE = -3	Revolutional feedrate control in JOG mode
SD43300 \$SA_ASSIGN_FEED_PER_REV_SOURCE [X axis] = -3	Revolutional feedrate for positioning axis X
SD43300 \$SA_ASSIGN_FEED_PER_REV_SOURCE [Z axis] = -3	Revolutional feedrate for positioning axis Z

Note

Ensure that the axis-specific setting data SD43300 SA_ASSIGN_FEED_PER_REV_SOURCE is set the same.

References

- Operating:
Operating Manual, Turning/Milling, Chapter "Working with a manual machine"
- Programming:
Programming Manual, Fundamentals; Chapter "Working area limit"

8.10 User status display (OEM)

Function

User icons can be displayed in the Machine operating area in the second line in the header. In this case, the program name is displayed in the right-hand field of the third line in the header. Active hold conditions hide the program name.

The display of the user icons is controlled via PLC bits. The icons are provided in the form of PNG files and stored resolution-specifically in the file system at: `/user/sinumerik/hmi/ico/ico640`, etc. **Maximum 16 positions are available in the header** to display the user icons.

Configuration

The user icons are configured in the `slmahdconfig.ini` file in Section [UserIcons]. The following data is required:

- The start address of the PLC double word with the bits to control the display of the user icons.
- For each bit used, the file name of the icon and the position at which the icon should be displayed.

Alternatively, you can use the following parameters for the definition:

- **USER_ICON_BASE**
This means that 32 bits are available with UI_0 (highest bit) to UI_31 (lowest bit).
Example: USER_ICON_BASE=DB80.DBB0
⇒ UI_0 = DB80.DBX3.0 and UI_31 = DB80.DBX0.7
- **USER_ICON_BASE256**
This means that 256 bits are available with UI_0 (highest bit) to UI_255 (lowest bit).
Example: USER_ICON_BASE256=DB19.DBB80
⇒ UI_0 = DB19.DBX111.0 and UI_255 = DB19.DBX80.7

Example

```
[UserIcons]
USER_ICON_BASE=DB19.DBB80
; set the start byte of the PLC double word that displays the icons (DB19.DBD80)
UI_0 = icon1.png,4
; show icon1.png on position 4 if bit 0 (DB19.DBX83.0) is set
UI_1 = icon2.png,6
; show icon2.png on position 6 if bit 1 (DB19.DBX83.1) is set
```

Superimposition

With a **non-unique** selection (bits for multiple identifiers whose image is to be displayed at the same position):

The image with the lower identifier number is superimposed with the image with the higher identifier number for the same position. Non-unique selection can also be used with less than 16 active selection bits. The PLC can deliberately use this superimposition to overlay less important displays (with lower identifier numbers) with important displays (high identifier numbers).

Note

If two HMI applications are assigned to one NC/PLC unit (M:N), the PLC must first determine from the interface which of the HMI interfaces is active. Icons can be requested by the PLC for the active HMI application only.

Possible applications

With a **unique selection**, a maximum of 16 bits are set simultaneously in the four selection bytes for each of the different positions.

- 32 alternative images for a position, each selected with a different bit (currently only one active selection bit)
- Two alternative images for each of the permissible 16 positions, each position requires two of the 32 selection bits (maximum 16 active selection bits at any one time)

8.10 User status display (OEM)

- Combination thereof with a maximum total of 32 images for the maximum 16 positions (maximum of 16 active selection bits at any one time)
- More than 16 selection bits.

Configuring OEM texts

Machine manufacturers can display their own texts in the program directory and program name lines. These texts cannot include % parameters as is possible in user texts. The lines are divided into <OEM_NUMBER_TEXTFIELD> areas according to the entry in HEADER.INI.

The data is provided in a user data block. The number of the data block and the exact offset are declared in the HEADER.INI file. The following entries are available in the [OEMTextInfo] section for this purpose:

```
[OEMTextInfo]
```

```
OEM_NUMBER_TEXTFIELD = 1 ; maximum of 3 text fields per line
```

```
OEM_TEXT_BASE = DBx.DBBx
```

The interface in the user DB has the following format:

Start-DBB byte or Start-DBB +18 byte	Bit0=1: there is something to do All bits=1: job is acknowledged by HMI All bits=0: PLC can write to interface again.
Start-DBB +1 word or Start-DBB +19 word	Offset for text from alpu_xx.com to be displayed; 700 000 is the base address and is always summed internally. All bits=0: Delete line
Start-DBB + 3 string or Start DBB+20 double Word	String containing a maximum of 12 characters

The first parameter contains an offset for a text number with a base of 700,000 (user PLC texts). If this parameter = 0, the corresponding line is deleted. The second parameter contains a maximum 12-character string. Unless empty, this text is output instead of the user PLC text.

The parameters must be written from "back to front" in the PLC, i.e. all parameters must be initialized before the first byte is set. The data for the next line is located exactly 18 bytes after the start byte. The lines are output from left to right; a maximum of three output fields per line are provided (i.e. a total of six fields are possible).

A hotlink to the first byte for each line is set up in the operating software. If a change occurs in that byte, the operating software responds. If the operating software displayed the data, it writes a -1 to the first variable. The operating software is informed about this change by means of the hotlink, whereupon it writes a 0 to this variable.

The HMI itself does not have to react to this change. The PLC can only write data to the interface again if the variable is set to 0. This mechanism ensures that the operating software knows about all parameter changes.

Configuring OEM icons

OEM icons indicating the machine state can be output in the field provided for the program control display (e.g. SBL, M01, etc.). If OEM icons are defined, the elements for the program control display are hidden.

The icon names are identified in the HEADER.INI configuration file, similar to user icons:

```
[OemIcons]
OI_0= <name.png>, <position>
...
OI_31= <name.png>, <position>
```

Here, <name> is the file name of the bit map and <Position> is the output position (from 1 to 16) in the display line. Multiple bit maps can be output at the same position. If more than one bit map is active simultaneously at the same position, the bit map with the highest screen number is displayed.

The output is controlled by means of a PLC double word. This double word is declared in the next section in the HEADER.INI file:

```
[OemIcons]
Oem_ICON_BASE = DBx.DBBy
```

Each bit in this double word represents exactly one OEM icon corresponding to the screen number. Thus, if bit 0 is set, bit map OI_0 is displayed. If a bit is reset, the associated bit map is deleted.

A maximum of 16 icons can be displayed, enabling a total of 16 display positions. Empty positions need not be specified.

Simulation and simultaneous recording

9.1 Simulation overview



Software option

You require the "3D simulation 1 (finished part)" option for the 3D representation of the simulation.

Simulation is supported with its own program interpreter (SNCK) and a separate simulation-data environment in SINUMERIK Operate. The SNCK considers the complete syntax of the SINUMERIK control family, including the possibility of incorporating special user options on the machine by comparing data with the NC environment. The simulation data can be matched statically as required with the NC environment (initialization data, macros, user data, tool data, machining cycles) or also dynamically when tool data or machining cycles are changed.

Machining simulations, with emphasis on the drilling, milling, turning and grinding technologies, can be performed in the workpiece coordinate system for certain machine kinematics on the user interface of the controller:

- The simulation of the finished part is performed with the real NC data.
- The NC data is automatically compared at each change of the part program.
- The simulation allows a dynamic representation of the machining, even for active 5-axis transformation (TRAORI) and swiveled planes.
- Turning on milling machines.
- Simulation of a machining channel for conventional milling machines.
- Simulation of up to four machining channels for turning machines with B axis.
- Intelligent determination of the block times and the program execution time.
- Very fast graphical representation through the continuous refinement of the workpiece.
- Optimum resolution for each selected picture area.
- Any sections are possible.
- In parallel to the machining of one workpiece, the machining of another workpiece can be simulated (as of NCU 720.x).
- 3D representation of simulation (option).

General conditions for the simulation

- Very fast simulation result, this means, however, low dynamic resolution.
- For complex representations, fine recording is too slow or does not run for very complex parts.
- "Quickview" for mold-making part programs.
- The simulation cannot be used for compile cycles with special kinematics.

9.1 Simulation overview

- Compile cycles are supported only to a limited extent.
- Blank CAD data cannot be imported (for example, for castings).
- Couplings cannot be simulated (coupled motion, axial master-value coupling, electronic gearbox):
 - Coupled motion: TRAILON(<following axis>, <leading axis>, <coupling factor>)
 - Axial master-value coupling: LEADON(Faxis, LAxis, curve table n) actual value and setpoint coupling
 - Electronic gearbox: EGDEF(FA, LA1, coupl.1, LA2, coupl.2, ... LA5, coupl.5), EGON, EGONSYN, EGONSYNE
- Reference point approach G74 is not operative during simulation
- The following is not supported:
 - Axis container
 - Dependencies on part programs and PLC
 - Multiple clamping
 - CNC function Synchronized actions for safe programmable logic (SPL) in the NC
- Different technologies in different channels with different axis coordinate systems are not fully supported.
- A 3D simulation is not possible for the grinding technology.
- The software limit switches are not evaluated:

In a manufacturer cycle for the tool replacement, the "Software limit switch + the Z axis" machine data is used as replacement position:
G0 Z=\$MA_POS_LIMIT_PLUS[Z]-0.1

The position, however, is not used in the simulation, but rather the initial setting of the machine data = 1e8 = 100,000,000. This specified position is traversed for the simulation. This makes the simulation very slow and causes the machining time to explode, e.g. 555 hours rather than 2 minutes.
- Program start in the simulation:

If MD22622 \$MCS_DISABLE_PLC_START is set in a channel, then the channel is not automatically started in the simulation. It can then only be started from another channel using the part program command "START". This function is not available for the SINUMERIK 828D.
- Alarm display in the simulation:

Alarms are output in the simulation exactly the same way as alarms in the normal program execution, but no more than five alarms are displayed (above the graphic, to distinguish them from the machine alarms).

If a different error signaling behavior (e.g. syntax error) occurs between simultaneous recording and simulation, then, for example, the reason can be due to the hardware access operations in the programming. The simulation does not access the hardware, however, internally it reads the input with 0 and is then exited. On the other hand, for simultaneous recording, an alarm is output.

9.2 Setting the technology for simulation

Turning technology

Examples of machine kinematics:

- Conventional turning with two geometry axes
- Three spindles: Main spindle, counterspindle, tool spindle
- Counterspindle slides, tailstock as NC axis
- B axis: Aligning turning tools in the tool spindle
- Milling with geometry axes: TRANSMIT, TRACYL, TRAANG

Machine data for the turning technology:

MD52200 \$MCS_TECHNOLOGY = 1	Turning technology
MD52000 \$MCS_DISP_COORDINATE_SYSTEM = 34	Position of the coordinate system (example)

Milling technology

Examples of machine kinematics:

- Milling with five axes: Swivel/TRAORI
- Swivel head change

Machine data for the milling technology:

MD52200 \$MCS_TECHNOLOGY = 2	Milling technology
MD52000 \$MCS_DISP_COORDINATE_SYSTEM = 0	Position of the coordinate system (example)

Grinding technology

Machine data for the grinding technology:

MD52200 \$MCS_TECHNOLOGY = 3	Cylindrical grinding technology
MD52200 \$MCS_TECHNOLOGY = 4	Surface grinding technology
MD52000 \$MCS_DISP_COORDINATE_SYSTEM = 0	Cylindrical grinding and surface grinding with moving column (example)
MD52000 \$MCS_DISP_COORDINATE_SYSTEM = 6	Surface grinding with table machine (example)

Meaning of the axes

MD52206 \$MCS_AXIS_USAGE[n]	Meaning of the axes in the channel [n] channel axis number
= 0	No special meaning
= 1	Tool spindle (driven tool)

= 2	Auxiliary spindle (driven tool)
= 3	Main spindle (turning)
= 4	C axis of the main spindle (turning)
= 5	Counterspindle (turning)
= 6	C axis of the counterspindle (turning)
= 7	Linear axis of the counterspindle (turning)
= 8	Tailstock (turning)
= 9	Back rest (turning)

Enter the direction of rotation for the rotary axes that are not configured in a toolholder or a 5-axis transformation via the following channel-specific machine data.

MD52207 \$MCS_AXIS_USAGE_ATTRIB[n]		Attributes of the axes [n] channel axis number
Bit 0	Rotation around 1st geometry axis (for rotary axes)	
Bit 1	Rotation around 2nd geometry axis (for rotary axes)	
Bit 2	Rotation around 3rd geometry axis (for rotary axes)	
Bit 3	Reversal of direction of rotation (for rotary axes)	

The MD52290 \$MCS_SIM_DISPLAY_CONFIG machine data acts only on the OP019. It has no significance for SINUMERIK 828D.

MD52290 \$MCS_SIM_DISPLAY_CONFIG		Position of the status display of the channel in the simulation.
Bit 0	Upper left-hand corner	
Bit 1	Upper right-hand corner	
Bit 2	Lower left-hand corner	
Bit 3	Lower right-hand corner	

In the channel-specific machine data, for at least one geometry axis per channel, enter a value other than 0.0, e.g. 0.001. If the value is 0.0, the system assumes that this parameter has still not been set.

MD53230 \$MCS_SIM_START_POSITION		Axis position when starting the simulation
Simulation is only possible if a value not equal to 0 has been set for at least one geometry axis.		

Deactivating simulation

The following machine data must be set to deactivate the simulation:

MD51226 \$MNS_FUNCTION_MASK_SIM		Simulation function screen
Bit 1 = 1	Deactivate simulation	
Bit 10 = 1	Hide zero point symbol	

9.3 Simultaneous recording overview

**Software option**

You require the option "Simultaneous recording (real-time simulation)" for the "Simultaneous recording" function.

**Software option**

You require the "ShopMill/ShopTurn" option to display the tailstock

**Software option**

You require the "3D simulation 1 (finished part)" option for the 3D representation of the simultaneous recording function.

During machining, the tool paths can be simultaneously recorded on the display of the controller in a three-side view or 3D view. Workpiece graphics and views correspond to the graphic simulation.

Note

To prevent an incomplete representation, you must activate the simultaneous recording prior to the NC start.

Example: Activating the tailstock representation in the spindle chuck data

Procedure:

1. Make the tailstock visible by making the selection in the program.
2. Specify the tailstock dimensions:
Select in the operating area "Parameters" → "Setting data" → "Spindle chuck data", and enter the "ZR" and "XR" parameters for the tailstock length and tailstock diameter.
The tailstock dimensions are written into the following machine data:

MD52218 \$MCS_FUNCTION_MASK_TURN	Turning function screen
Bit 2 = 1	Tailstock enable: The tailstock becomes visible if selected in the "tailstock" program.

MD53240 \$MAS_SPINDLE_PARAMETER[i]	Spindle chuck data in mm
[0]: Chuck dimension [1]: Stop dimension [2]: Jaw dimensions	

MD53241 \$MAS_SPINDLE_CHUCK_TYPE	Spindle jaw type
0 = clamp from the outside 1 = clamp from the inside	

MD53242 \$MAS_TAILSTOCK_PARAMETER[i]	Tailstock data in mm
[0]: Tailstock dimension [1]: Tailstock length	

Activating the machining time display

Timers are provided as system variables in the Program runtime function. While the NC-specific timers are always activated (for time measurements since the last controller power-up), the channel-specific timers must be started via the following channel-specific machine data.

MD27860 \$MC_PROCESS_TIMERMODE	Activate the program runtime measurement.
Bit 0 = 1	The measurement of the total runtime for all part programs is active (\$AC_OPERATING_TIME).
Bit 1 = 1	The measurement of the actual program runtime is active (\$AC_CYCLE_TIME).
Bit 4 = 1	Measurement also during active dry run feedrate.
Bit 5 = 1	Measurement also during program test.

9.4 Clamping a blank

The blank clamping function allows you to select and specify in more detail the clamping location for the blank.

You require this function if your machine satisfies one or more of the following requirements:

Requirements

- The machine has several clamping options available for selection.
- The machine has a clamping option that does not point in the Z direction as standard (e.g. a boring mill).

Under the following circumstances, you can specify the position and orientation of the rotary axes in the machine data MD53220 and MD52207:

- The position and orientation of the rotary axis is offset against the MCS zero point, but not specified by a toolcarrier or a TRAORI.
- In a turning machine, the reference points of the main spindle or counterspindle are offset against the MCS zero point in X or Z.
- The counterspindle cannot be traversed with a saddle, but rather is at a fixed position.

In order that for simulation / simultaneous recording, the rotary axes are displayed at the correct location, their positions in the machine coordinate system must be specified using the following machine data:

MD53220 \$MAS_AXIS_MCS_POSITION[n]	Position of the axis in the MCS [n] 0 = X, 1 = Y, 2 = Z

For further information about the transformation of a rotary axis, see Chapter "Configuring cycles (Page 443)".

Function

Set the following channel-specific machine data:

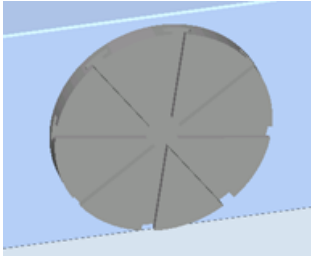
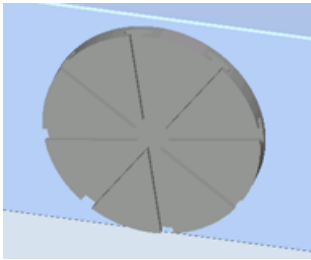
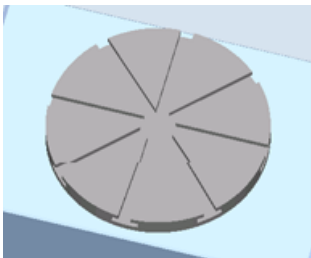
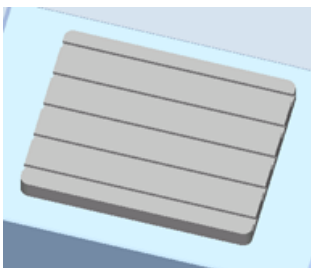
MD52207 \$MCS_USAGE_ATTRIB[n]	Attribute of the axes in the channel [n] channel axis number
Bit 0	Rotation around 1st geometry axis (for rotary axes)
Bit 1	Rotation around 2nd geometry axis (for rotary axes)
Bit 2	Rotation around 3rd geometry axis (for rotary axes)
Bit 8	Offer rotary axis for blank clamping (on milling machines)

MD52214 \$MCS_FUNCTION_MASK_MILL	Milling function screen
Bit 1	Offer fixed table for blank clamping (on milling machines)

Take the machine configuration for MD20080 from the example.

Function options

In the program header, a blank can be optionally set to A, B, C and table clamping:

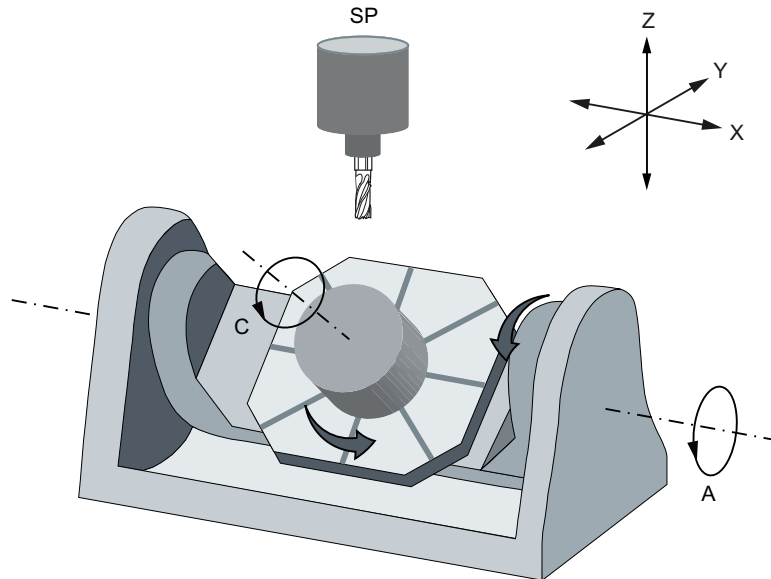
Clamping	View	Machine data
A axis		MD52207 \$MCS_USAGE_ATTRIB bit 0 and bit 8 = 1
B axis		MD52207 \$MCS_USAGE_ATTRIB bit 1 and bit 8 = 1
C axis		MD52207 \$MCS_USAGE_ATTRIB bit 2 and bit 8 = 1
Table		MD52214 \$MCS_FUNCTION_MASK_MILL bit 1 = 1

Note

To obtain the selection option in the program header, two rotary axes must be defined for the raw part clamping.

Example for the machine configuration

Example for a machine configuration when the A axis should rotate at the X axis:



MD20080 \$MC_AXCONF_CHANAX_NAME_TAB[n]	Channel axis name in channel [n]
[0]	X
[1]	Y
[2]	Z
[3]	Spindle
[4]	A
[5]	C

MD52207 \$MCS_USAGE_ATTRIB[4]	Attribute of rotary axis A in channel 4
= 181H	
Bit 0 = 1	Rotation around 1st geometry axis (for rotary axes)
Bit 8 = 1	Offer rotary axis for blank clamping (on milling machines)

References

Operating Manual, Milling

- Chapter "Create ShopMill program" > "Program header"
- Chapter "Create ShopMill program" > "Change program settings"

Spindle functions

10.1 Spindle control

You can select from the following spindle control settings:

End of program

The following M functions are used for the end of program:

M function from MD10714 \$MN_M_NO_FCT_EOP	Main program end	Spindle continues to run
M2, M30	End of main program and return jump to beginning of program	Spindle stops
M17	End of subprogram and return jump into the main program	Spindle continues to run

If machine data MD10714 \$MN_M_NO_FCT_EOP is not equal to zero, then a distinction is made between M2 / M30 (program end of an NC program) and the M function set in MD10714 \$MN_M_NO_FCT_EOP.

Example

MD10714 \$MN_M_NO_FCT_EOP = 32 means "M32" for the program end of a program generated in the "JOG" or "MDA" operating mode.

Among other things, this functionality is required in order to continuously start the spindle in manual operation (e.g. for scratching).

Configuring keys

If you implement a manual control using the keys on the machine control panel, then this is realized via the following interface signals in the spindle data block:

DB3x.DBX30.0	Spindle stop
DB3x.DBX30.1	Spindle start, clockwise rotation
DB3x.DBX30.2	Spindle start, counter-clockwise rotation

The spindle can be started and stopped in the following state:

DB21.DBX35.7 = 1	Channel in reset state
DB21.DBX35.6 = 1	Channel state interrupted
DB21.DBX35.3 = 1	Program state interrupted

Note

If the spindle must be stopped when the program is running, then set the interface signal DB3x.DBX4.3, "Feed stop/spindle stop" in the user PLC.

Additional settings:

MD11450 \$MN_SEARCH_RUN_MODE	Search run parameterization
Bit 1 = 1	Automatic ASUB start after output of the action blocks (see also MD11620 \$MN_PROG_EVENT_NAME). Alarm 10208 is not output until the ASUB is completed.

References

For a detailed description how to configure the spindle, see
Function Manual, Basic Functions: Spindles (S1)

Direction of rotation (only for ShopTurn)

In the ShopTurn user interface, ensure that the direction of rotation of the spindle and C axis is correctly displayed and when programming ShopTurn functions, that the correct direction of rotation is executed. You must base these settings on the actual direction of rotation of the spindle / C axis on the machine.

- If you use the functions cylinder surface transformation and face machining, then a precondition is that the function was correctly commissioned.
See Cylinder surface transformation (TRACYL) (Page 482)
- The direction of rotation of the spindle / C axis for the M function M3 that is displayed in the ShopTurn interface is defined in part using machine data:
See: Defining the direction of rotation in Section: Setting-up ShopTurn cycles for turning (Page 471)
- The spindle direction of rotation (M3/ M4) is assigned to the positive direction of rotation of the C axis via the interface signal DB3n.DBX17.6. The bit defines whether M3 and C+ rotate in the same direction (= 0) or in opposite directions (= 1). The corresponding setting options can be found in the following sections:
 - Direction of rotation of counter-spindle Setting up the counterspindle under ShopTurn (Page 475)
 - Direction of rotation of main spindle Setting-up ShopTurn cycles for turning (Page 471)

10.2 Analog spindle**Machine axis index for analog spindle**

For a spindle which is not assigned to any drive (FD, MSD, SLM), the power display can be controlled via the PLC.

In order that the operating software identifies the spindle as analog spindle, enter the axis index of the analog spindle in the following machine data:

MD51068 \$MNS_SPIND_DRIVELOAD_FROM_PLC1	Machine axis index spindle 1 Utilization display from the PLC
	Machine axis index of a spindle (analog) that draws the data for the utilization display in the T,F,S window from the PLC (DB19.DBB6).

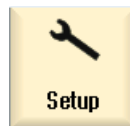
MD51069 \$MNS_SPIND_DRIVELOAD_FROM_PLC2	Machine axis index spindle 2 Utilization display from the PLC
	Machine axis index of a spindle (analog) that draws the data for the utilization display in the T,F,S window from the PLC (DB19.DBB7).

10.3 Leadscrew error compensation

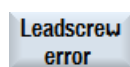
Instead of transferring the compensation data via the part program or INI file to the NC, you can enter and check the compensation data for all axes or change the configuration in an input screen.

Finally, the changed data is activated in the NC in one step. The machine data required when activating is set automatically.

Procedure



1. Select the "Start-up" operating area and press the "NC" softkey.



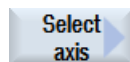
2. Press the "Leadscrew error" softkey.
The "Values of the Compensation Tables" window opens and displays the default compensation tables for an axis or, if no compensation values are available, the measuring system used.



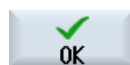
3. Press the "Axis +" or "Axis -" softkey to select the desired axis.



- OR -



Press the "Select axis" softkey.



The "Axis Direct Selection" window opens. Position the cursor on the desired axis and confirm with "OK".



4. Press the "Configuration" softkey to edit the compensation values for a selected axis.
The "Configuration of the Compensation Tables" window opens.
5. Enter the following:
 - Type of compensation: EEC or CEC.
 - Start position, end position and interpolation point spacing.
 - Only for EEC:
Used measuring system - if several measuring systems are available, select the desired system from the selection menu.
 - Only for CEC:
 - Compensation table (-): Index of the table in the negative direction.
 - Compensation table (+): Index of the table in the positive direction.



6. Confirm your entries with the "Activate" softkey.
A dialog box to carry out a NC warm restart is then displayed to activate the compensation. Confirm with "OK".



The "Values of the Compensation Tables" window opens.

7. Enter the desired compensation values for each interpolation point.



8. Press the "Activate" softkey again to transfer the compensation values to the NC.

References

Extended Functions (K3) Function Manual

10.4 Spindle diagnostics

10.4.1 Evaluation of the status signals



Software option

For the "Spindle diagnostics" you require the "S-Monitor" option.

Main spindle drives are monitored with various status signals. The DRIVE-CLiQ connection at the spindle can be used to evaluate these signals in the drive. For a more effective diagnosis of the spindle operating states, the status signals are evaluated and the following information displayed on the user interface:

- Operating hours
- Temperatures
- Speed/power
- Logistics data

Preconditions

- A spindle has an SMI24:
r0459, bit 14 = 1.
- Drive telegram 139 is configured for the spindle.
- The spindle functionality for the machine axis is present, when:
MD35000 \$MA_SPIND_ASSIGN_TO_MACHAX > 0
The value corresponds to the spindle number.

References

For further information, refer to the following documentation:
Basic Functions (S1), Function Manual; Section "Spindle with SMI24"

Procedure



1. Select the "Diagnostics" operating area.



2. Press the menu forward key twice.



3. Press the "Spindle diag." softkey to obtain a data overview.
The "S1 Spindle Diagnostics" window opens.

Note

If several spindles are in operation, select the desired spindle using the "Spindle +" or "Spindle -" softkey.

Data overview

The "S1 Spindle Diagnostics" window shows the following information:

Parameter	Value
Operating hours	h
• Spindle under control • Spindle under speed	
Number of clamping cycles (tool change)	
Motor temperature	°C
Clamping system (sensor 1)	V
Piston free (sensor 4)	
Shaft in the change position (sensor 5)	
Load classification *)	%

Parameter	Value
Nominal residual use ^{*)}	h
Maximum speed	rpm

- ^{*)} If at least one collision was detected, "Load classification" and "Nominal residual use" will be grayed-out because they no longer display any reliable values.

10.4.2 Fetch temperatures

To ensure a full temperature protection, a diagnosis of the temperatures can also be performed at low speeds or standstill. Which temperatures are displayed depends on the number and the mounting location (e.g. S6) of the used sensors, for example:

- Motor temperature (KTY84)
- Winding, full motor protection (PTC)
- S6, e.g. bearing temperature (KTY)

Procedure

1. The "S1 Spindle Diagnostics" window is open.
2. Press the "Temperatures" softkey.
The "S1 Temperatures" window opens and displays the acquired data.
3. Press the "Histogram" softkey to track the temperature change of a sensor over a certain period.
- OR -
Press the "Back" softkey to return to the data overview.



10.4.3 Fetch speed, torque and power

To check the loads on the spindle, the speed, torque, and power are acquired in a histogram as function of the operating hours. The color coding indicates the following states:

- Light green: low load
- Green: rated load
- Yellow: high load
- Orange: critical load

Procedure



1. The "S1 spindle diagnostics" window is open.
2. Press the "Speed/power" softkey.
The "S1 Speed/Power" window opens and displays in a histogram the acquired data.
3. Press the "Back" softkey to return to the data overview.

Possible conclusions

- The speed histogram is an indication of the load on the bearings and rotary gland caused by the speed.
- The speed histogram is an indication of the load on the bearings caused by the stock removal forces. The relationship between torque and radial force changes, however, depending on the tool diameter (lever arm), the milling arc (superimposition of the cutting forces of the individual cutting edges) and the cutting force coefficients (the stock removal consists of cutting force force, infeed force and passive force - only the cutting force is acquired via the torque).
- The power histogram provides an indication of the potential for a process optimization. The power histogram also allows an estimate whether the loads caused by speed and torque are superimposed over time and thus accumulated.

10.4.4 Checking the clamping system

The diagnosis of the clamping system offers the following capabilities:

- Fast and robust detection of the clamped states
- Monitoring of the clamping task and the clamping system

The sensors in the motor provide information about the clamping device and the orientation of the tool in the spindle:

Sensor	Meaning	
S1	Clamped state	Released
		Clamped with tool
		Clamped without tool
S4	Piston free	
S5	Shaft in the change position	

Procedure



1. The "S1 spindle diagnostics" window is open.
2. Press the "Clamping system" softkey.
The "S1 Clamping System" window opens and displays the acquired data. The clamping state is displayed as a histogram.

Speed limitation



3. Press the "Speed limitation" softkey to change the limit values for the speeds of the clamped states.

The "S1 Speed Limitations" window opens.



4. Press the "Change" softkey to enter the desired speed limitations depending on the clamped state and confirm with "OK".



- OR -



3. Press the "Cycle counter diagnostics" softkey to fetch the clamping times and perform a diagnosis of the clamping system.

The "S1 Cycle Counter and Diagnostics" window opens and displays the acquired data.

- OR -



3. Press the "Back" softkey to return to the data overview.

10.4.5 Fetching the logistics data

The following manufacturer characteristics of the spindle can be fetched via DRIVE-CLiQ:

- Manufacturer identification
- Drawing number
- Serial number
- Date of manufacture
- Service date and service information (1...4):
Can be fetched only with "Service" access level.

Procedure



1. The "S1 spindle diagnostics" window is open.
2. Press the "Logistics data" softkey.
The "S1 Logistics Data" window opens and displays the acquired data.



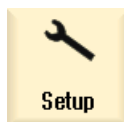
3. Press the "Back" softkey to return to the data overview.

Drive system

11.1 Commissioning of drives

Procedure

Once commissioning of the PLC has been completed, you can carry out commissioning of the SINAMICS drives via the user interface of SINUMERIK Operate.



1. Select the "Start-up" operating area.



2. Press the "Drive system" softkey.
The "Drive system overview" window opens.
A message is output whether the drive system has already been commissioned or not.

References

The procedure for commissioning the SINAMICS drives is described in the following documentation:

CNC Commissioning Manual: NCK, PLC, drive

Tool management

12.1 Machine data for tool management

12.1.1 Settings with/without magazine management

Setting without magazine management

Make the following settings if you are commissioning the tool management **without** NC magazine management:

MD18080 \$MN_MM_TOOL_MANAGEMENT_MASK	Memory for the tool management is reserved step-by-step (SRAM).
= 02H	
Bit 1	Provide monitoring data.

MD20310 \$MC_TOOL_MANAGEMENT_MASK	Activating the tool management functions.
= 02H	
Bit 1	Monitoring functions of tool management active.

MD17530 \$MN_TOOL_DATA_CHANGE_COUNTER	Marking the changed tool data for HMI.
= 1FH	
Bit 0	Tool status change.
Bit 1	Tool remaining unit quantity change.
Bit 2	Tool data change service.
Bit 3	Magazine data in the change service.
Bit 4	ISO data in the change service.

MD28450 \$MC_MM_TOOL_DATA_CHG_BUFF_SIZE	Buffer for changing tool data (DRAM).
= 100	Number of entries in the buffer for the OPI change service for tool data. Dynamic memory is used. This buffer is only created if either bit 2 or bit 3 is set in MD17530 \$MN_TOOL_DATA_CHANGE_COUNTER.

MD19320 \$ON_TECHNO_FUNCTION_MASK	
=2000000H	
Bit 25	Enables the tool monitoring function without magazine management.

Setting with magazine management

Make the following settings if you are commissioning the tool management **with** NC magazine management:

MD18080 \$MN_MM_TOOL_MANAGEMENT_MASK	Memory for the tool management is reserved step-by-step (SRAM).
= 03H	
Bit 0	Provide tool management data.
Bit 1	Provide monitoring data.

MD20310 \$MC_TOOL_MANAGEMENT_MASK	Activating the tool management functions.
= 03H	
Bit 0	Magazine management active.
Bit 1	Monitoring functions of tool management active.

MD17530 \$MN_TOOL_DATA_CHANGE_COUNTER	Marking the changed tool data for HMI.
= 1FH	
Bit 0	Tool status change.
Bit 1	Tool remaining unit quantity change.
Bit 2	Tool data change service.
Bit 3	Magazine data in the change service.
Bit 4	ISO data in the change service.

MD28450 \$MC_MM_TOOL_DATA_CHG_BUFF_SIZE	Buffer for changing tool data (DRAM).
= 100	Number of entries in the buffer for the OPI change service for tool data. Dynamic memory is used. This buffer is only created if either bit 2 or bit 3 is set in MD17530 \$MN_TOOL_DATA_CHANGE_COUNTER.

MD19320 \$ON_TECHNO_FUNCTION_MASK	
= 10H	
Bit 4	Enables the tool monitoring function with magazine management.

12.1.2 Configuring the access levels of the tool management

Access levels for selected parameters

MD51199 \$MNS_ACCESS_WRITE_TM_GRIND	Access level, write TM grinding data.
MD51200 \$MNS_ACCESS_WRITE_TM_GEO	Access level, write TM geometry data.
MD51201 \$MNS_ACCESS_WRITE_TM_WEAR	Access level, write TM wear data.
MD51202 \$MNS_ACCESS_WRITE_TM_WEAR_DELTA	Access level, restricted writing of tool wear values.

MD51203 \$MNS_ACCESS_WRITE_TM_SC	Access level, write TM additive offsets.
MD51204 \$MNS_ACCESS_WRITE_TM_EC	Access level, write TM operating off-sets.
MD51205 \$MNS_ACCESS_WRITE_TM_SUPVIS	Access level, write TM monitoring data.
MD51206 \$MNS_ACCESS_WRITE_TM_ASSDNO	Access level, write TM unique D number.
MD51207 \$MNS_ACCESS_WRITE_TM_WGROUP	Access level, write TM wear groups (magazine location / magazine).
MD51208 \$MNS_ACCESS_WRITE_TM_ADAPT	Access level, write TM tool adapter geometry data.
MD51209 \$MNS_ACCESS_WRITE_TM_NAME	Access level, write TM tool name and duplo.
MD51210 \$MNS_ACCESS_WRITE_TM_TYPE	Access level, write TM tool type.

The access level of individual parameters can be overlaid through settings in the <PARAMETERCONFIGURATION> tag, "Access level" entry (see Section Configure the list parameters (Page 147)).

Access levels for the "Further Details" window

MD51198 \$MNS_ACCESS_READ_TM_ALL_PARAM	Protection level, TM details - read all parameters. Depending on the machine data, the "Further details" softkey is either hidden or displayed.
MD51215 \$MNS_ACCESS_WRITE_TM_ALL_PARAM	Access level, TM details - write all parameters.

Access levels for certain functions of the tool management

MD51216 \$MNS_ACCESS_TM_TOOL_CREATE	Access level, TM, create tool.
MD51217 \$MNS_ACCESS_TM_TOOL_DELETE	Access level, TM, delete tool.
MD51218 \$MNS_ACCESS_TM_TOOL_LOAD	Access level, TM, load tool.
MD51219 \$MNS_ACCESS_TM_TOOL_UNLOAD	Access level, TM, unload tool.
MD51220 \$MNS_ACCESS_TM_TOOL_MOVE	Access level, TM, move tool.
MD51221 \$MNS_ACCESS_TM_TOOL_REACTIVATE	Access level, TM, reactivate tool.
MD51222 \$MNS_ACCESS_TM_TOOL_MEASURE	Access level, TM, measure tool.
MD51223 \$MNS_ACCESS_TM_TOOLEEDGE_CREATE	Access level, TM, create tool cutting edge.
MD51224 \$MNS_ACCESS_TM_TOOLEEDGE_DELETE	Access level, TM, delete tool cutting edge.
MD51225 \$MNS_ACCESS_TM_MAGAZINE_POS	Access level, TM, position magazine.

12.1.3 Additional settings

Additional settings

You can enable further functions on the user interface via the following machine data / setting data:

MD52270 \$MCS_TM_FUNCTION_MASK	
Bit 0	Creating a tool at the magazine location is not permitted.
Bit 1	Load/unload lock if the machine is not in the reset state.
Bit 2	Load/unload lock for EMERGENCY OFF.
Bit 3	Loading/unloading or relocating tool in/out of spindle locked.
Bit 4	Tool loaded directly into spindle.
Bit 5	Using the grinding configuration file.
Bit 6	Relocating a tool in/out of spindle is permitted despite lock (see bit 3).
Bit 7	Tool is created using the number.
Bit 8	Hide tool relocation.
Bit 9	Hide magazine positioning.
Bit 10	Reactivate tool with magazine positioning.
Bit 11	Reactivate tool in all monitoring types.
Bit 12	Hide reactivating tool.

SD54215 \$SNS_TM_FUNCTION_MASK_SET	
Bit 0	Diameter display for rotating tools. Cutting edge parameters 6 (tool nose radius) and 15 (wear radius) are not displayed as a radius value, but as a diameter value for the following tool types: Types 100 to 299, 580, 710, 711, 712, 713, 714 and 725. Cutting edge parameter 7 (outside radius) is not displayed as a radius value, but as a diameter value for the following tool types: Types 140 and 714.
Bit 1	M4 is the standard direction of rotation for all turning tools. When creating turning tools, the direction of rotation is pre-assigned with M4.
Bit 2	A name is not suggested when creating a tool.
Bit 3	Input lock, tool name and tool type for loaded tools. For loaded tools, the tool name and the tool type can no longer be changed.
Bit 4	Input lock for loaded tools if the channel is not in the reset state.
Bit 5	Tool wear entries are added: Wear data is entered in addition to the already existing wear value.
Bit 6	Only numerical entries are permitted in the "Tool identifier".
Bit 7	Hide tool monitoring parameters. The tool monitoring parameters are hidden in the user interface.
Bit 8	Diameter display for face axis - geometry. The geometry value of the transverse axis is displayed as a diameter value, if a transverse axis is defined in MD20100 \$DIAMETER_AX_DEF and SD42940 \$TOOL_LENGTH_CONST is set to 18 and SD42950 \$TOOL_LENGTH_TYPE is set to 2.

Bit 9	Diameter display for face axis - wear. The wear value of the transverse axis is displayed as a diameter value, if a transverse axis is defined in MD20100 \$DIAMETER_AX_DEF and SD42940 \$TOOL_LENGTH_CONST is set to 18 and SD42950 \$TOOL_LENGTH_TYPE is set to 2.
Bit 10	Enable tool load/relocate to buffer storage locations. The magazine number can be entered into the load dialog box. It is therefore possible to access the buffer storage via magazine number 9998.
Bit 11	Creating new tools at gripper locations is blocked.
Bit 12	Measuring tools are not unloaded with the "Unload all" function.

Dependencies

The setting data SD54215 \$TM_FUNCTION_MASK_SET and the machine data MD20360 \$TOOL_PARAMETER_DEF_MASK have the following dependencies:

Dependency			Description	Recommendation
SD54215	⇔	MD20360		
Bit 9	⇔	Bit 0	If bit 0 is set in MD20360, then SD54215 bit 9 has no effect because the NC already returns the linear wear value of the transverse axis as a diameter value.	It is recommended that the setting MD20360 bit 0 and bit 1 not be used. Instead, use SD4215 bit 9 and bit 8. MD20360 bit 0 and bit 1 are effective only for turning and grinding tools. This setting therefore cannot be used on a turning/milling machine because the linear component of the transverse axis applies for turning tools in the diameter and for milling tools in the radius. When the setting data is used, only the display goes to diameter, the internal value in the NC remains at radius.
Bit 8	⇔	Bit 1	If bit 1 is set in MD20360, then SD54215 bit 8 has no effect because the NC already returns the linear geometry value of the transverse axis as a diameter value.	
Bit 0	⇔	Bit 11	If bit 11 is set in MD20360, then SD54215 bit 0 has no effect with regard to cutting edge parameter 6 (tool nose radius) because the NC already returns the tool nose radius as a diameter value.	
Bit 0	⇔	Bit 12	If bit 12 is set in MD20360, then SD54215 bit 0 has no effect with regard to cutting edge parameter 15 (wear radius) because the NC already returns the wear radius as a diameter value.	

Enabling the graphic tool and magazine display function

Make the following settings when the tools and the magazine should be represented graphically:

MD52271 \$MCS_TM_MAG_PLACE_DISTANCE		Distance between individual magazine locations in the unit of measurement of the basic NC system.
> 0	The tools and magazine locations are displayed graphically.	
= 0	The tools and magazine locations are not displayed graphically.	

You must make the following settings on the user interface:

1. Press the "Settings" softkey in the "Tool list" window.
The "Settings" window opens.
2. Activate the "Switch on graphical magazine display" checkbox.
This checkbox is offered only when the MD52271 \$MCS_TM_MAG_PLACE_DISTANCE machine data has a value > 0.

Supplementary conditions:

- MD52200 \$MCS_TECHNOLOGY = 2 (milling)
- MD52201 \$MCS_TECHNOLOGY_EXTENSION = 0

Default settings for unloading/loading and relocating

MD52272 \$MCS_TM_TOOL_LOAD_DEFAULT_MAG	Default magazine for loading tools.
= 0	No default magazine, default value

MD52273 \$MCS_TM_TOOL_MOVE_DEFAULT_MAG	Default magazine for relocating tools.
= 0	No default magazine, default value

MD52274 \$MCS_TM_TOOL_LOAD_STATION	Number of loading station.
= 0	All configured stations are considered, default value

Input limits for wear input

MD51212 \$MNS_TM_WRITE_WEAR_ABS_LIMIT	Maximum value of the tool wear.
MD51213 \$MNS_TM_WRITE_WEAR_DELTA_LIMIT	Maximum differential value of restricted tool wear input.
MD51214 \$MNS_TM_WRITE_LIMIT_MASK	Range of validity of the restricted tool wear input.

Setting the manual tool function

MD22562 \$MC_TOOL_CHANGE_ERROR_MODE	Response when errors occur at tool change.
Bit 1 = 1	Manual tools can be loaded.
Bit 2 = 1	Active D No. > 0 and active T No. = 0 generate an alarm message. Active D No. > 0 and active D No. = 0 generate an alarm message.

Note

If, in machine data MD22562 \$MC_TOOL_CHANGE_ERROR_MODE bit 2 = 1, then you have the option, using a correction selection (D No.) in the program, to stop the machine until the PLC signals that the tool change has been completed. To do this, machine data MD20270 \$MC_CUTTING_EDGE_DEFAULT must be set to = 0.

12.2 Configuring the user interface

Configurable settings

The following settings can be configured:

- General settings
- Configure the tool lists
- Configure the list parameters
- Configure the tool types
- Configure the "Additional Data" window
- Configure the "New Tool - Favorites" window
- Configure the "New Tool" window
- Configure the default values for new tools
- Configure the "Details" window
- Assign names for magazines and magazine locations
- Assign names for magazine location types
- Assign magazines to channels
- Coolant and tool-specific functions
- Reason for the tool change when reactivating
- Configure the code carrier connection

The following sections explain how you configure these adaptations.

Requirement for grinding technology

Set the "Universal" technology and set:
MD52270 \$MN_TM_FUNCTION_MASK bit 5 = 1.

Only the grinding tool types can be then selected on the user interface.

Procedure

1. Open the directory: **/siemens/sinumerik/hmi/template/cfg**
2. Copy the "oem_sltmlistconfig.xml" template file

Note

Other template for the grinding technology

The "sltmaddgrindinglistconfig.xml" template expands a turning configuration by adding grinding tools. Copy the file from the directory: **/siemens/sinumerik/hmi/template/cfg**

3. Copy the file to the directory **/oem/sinumerik/hmi/cfg**
OR: **/user/sinumerik/hmi/cfg**
4. Change the file name according to the set technology:

File name	Technology
sltmlistconfig.xml	Milling technology
sltmturninglistconfig.xml	Turning technology
sltmplclistconfig.xml	PLC tool management "TRANSLINE 2000"
sltmcirculargrindinglistconfig.xml	Cylindrical grinding technology
sltmsurfacegrindinglistconfig.xml	Surface grinding technology

5. Remove the comments of the example in the template to create your specific configuration.

Configuration examples

Examples for all possible configurations are provided in the template oem_sltmlistconfig.xml, which are described individually in the following sections:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
- <CONFIGURATION>
<!-- *****-->
<!-- enter your configuration behind this comment -->
<!-- ***** -->
```

Configuration examples

```
<!-- ***** -->
<!-- enter your configuration ahead this comment -->
<!-- ***** -->
</CONFIGURATION>
```

Enter your configuration using the following xml identifiers:

Identifier <tag>	for the following setting:
<SETTINGS>	General settings (Page 129)
<LISTCONFIGURATION>	Configure the tool lists (Page 134)
<PARAMETERCONFIGURATION>	Configure the list parameters (Page 147)
<TOOLTYPECONFIGURATION>	Configure tool types (Page 153)

Identifier <tag>	for the following setting:
<MOREDATACONFIGURATION>	Configuring the "More data" window (Page 154)
<NEWTOOLFAVORITECONFIGURATION>	Configure the "New tool - favorites" window (Page 156)
<NEWTOOLCONFIGURATION>	Configuring the "New Tool" window (Page 156)
<NEWTOOLDEFAULTVALUECONFIGURATION>	Configuring default values for new tools (Page 157)
<DETAILSCONFIGURATION>	Configuring the "Details" window (Page 161)
<MAGAZINEPLACENAMECONFIGURATION>	Assigning a name for magazines and magazine locations (Page 162)
<PLACETYPECONFIGURATION>	Assigning a name for magazine location types (Page 164)
<CHANNELMAGAZINEASSIGNMENT>	Assignment of magazines to channels (Page 165)
<TOOLCHANGEREASON>	Reason for the tool change when reactivating (Page 170)
<IDENTCONNECTIONCONFIGURATION>	Configure the code carrier connection (Page 174)

12.2.1 General settings

<SETTINGS> tag

The general settings for the tool management user interface are stored in the <SETTINGS> tag.

Entries	Meaning
ReactivateWithMagPos	True - reactivate with positioning of the tool to the tool magazine loading point. False - default setting, reactivate without positioning the tool magazine.
ReactivateAllMonitorModes	True - the actual values of all the monitoring types set in the NC are reset. False - default setting, actual value of the active monitoring type is reset.
ReactivateEnabled	True - default setting, reactivation is enabled. False - reactivation is disabled.
CreateNewToolDialog	True - the "New Tool" window is displayed. In this window, you can specify the tool name and the size of the tool (oversized tool) before actually creating the tool. Other input options can be defined by configuring the "New tool" window. False - default setting, the "New Tool" window is not displayed. Once the name has been entered, the tool will be created directly in the tool list.

Entries	Meaning
CreateNewMultiToolDialog	True - default setting, the "New Multitool" window is displayed. In this window, you can configure the following before actually creating the multitool: • Multitool names • Number of multitool locations • Type of clearance input • Value of the location clearance for each multitool location False - the "New Multitool" window is not displayed. Once the name has been entered, the multitool will be created directly in the tool list.
MagazineSelectionDialog	True - a dialog box displaying the buffer, tool magazine and NC memory is displayed. A mark can be added to or removed from each of these elements to indicate whether that element appears in the list display. The "Go to" softkey can be used to select an element directly in the list display. False - no dialog box for magazine selection. The system switches between the buffer, the tool magazines and the NC memory. If there is no MagazineSelectionDialog entry in the <code><Settings></code> tag, the dialog box is automatically displayed with more than one tool magazine (excluding the system magazine) when magazine configuration is taking place. If there is only one tool magazine, the switchover function is used.
NewToolFavoritesOnly	Refers to the "New tool" function. True - only those tool types that are configured as favorites are listed. False - default setting, all tool types are listed.
SortationInFirstEtcLevel	True - the "Sorting" function is placed at the first softkey level. The "Continue" function is omitted. False - default setting, sorting at all softkey levels.
ToolBufferOnceOnTop	True - default setting, buffer is displayed at the start of the list. False - buffer is displayed for each magazine. Only those locations are displayed that are assigned to the magazine.
UnloadToolFromLoadPlaceEnabled	True - a tool at the load position can be removed from the load position using the "Unload" softkey. False - default setting. The "Unload" softkey is not displayed for a tool at the load position.
AutoHNumberDisplayEnabled	Column of the cutting parameter "IsoHNoDPH - ISO H number" depending on the machine configuration. True - default setting, the column is automatically displayed. False - the column is not automatically displayed.
AutoUniqueDNumberDisplayEnabled	Column of the cutting edge parameter "UniqueDNo" depending on the machine configuration. True - default setting, the column is automatically displayed. False - the column is not automatically displayed.

Entries	Meaning
AccessLevelDeleteAllDNos	The following entry specifies the access level as of which the "Delete all D numbers delete" function is offered. Default setting: Access level 3 (user) - relevant only when the "Unique D number" setting has been made.
MagazineNumberOfDCheckAndActivate	Magazine number to which the D check (check for unique D number) and Activate (activation of a wear group) are to be applied. Default setting: -2 Significance: All magazines that are connected to a toolholder or a spindle.
AutoFixedPlaceDisplayEnabled	Column of the tool parameter "ToolStateFixed" - tool status - tool is fixed-location coded and acts dependent on the magazine configuration. True - default setting, the column is automatically hidden, if all of the configured magazines are fixed location coded. False - the column is not automatically hidden.
MagPlaceAdapterDisplayEnabled	Columns of the magazine location adapter parameters are displayed in the "Magazine" window. False - default setting, parameters are not displayed.
AdaptTransDataViewEnabled	The magazine location adapter transformed display of the cutting edge parameters is enabled. It can be switched on and off in the settings dialog of the tool lists. False - default setting, the magazine location adapter transformed display of the cutting edge parameters is not enabled.
ActiveCompensationsEnabled	The "Working offsets" function is enabled. This function depends on the machine configuration. Unique D number, magazine location adapter and magazine management must be set. False - default setting, the "Working offsets" function is not enabled.
M2NLockEnabled	Locking of an M to N switchover while a function of the tool management is being executed. True - default setting, switchover is locked.
AccessLevelWriteDetailsAllParameter	The following entry specifies the access level as of which write access applies in the "Details - All Parameters" window. Default setting: Access level 7 (keyswitch position 0) - can be overlaid through the setting in MD51215.
AccessLevelReadDetailsAllParameter	The following entry specifies the access level as of which the "Further details" function is offered in the "Details" dialog. Default setting: Access level 7 (keyswitch position 0) - can be overlaid through the setting in MD51198.
FixNumberOfMultiToolPlaces	Multitools are created with a fixed number of multitool locations. There is no prompt when you create the multitool.
FixKindOfDistInMultiTools	Multitools are created with a fixed type of clearance input of the multitool locations. There is no prompt when you create the multitool. Possible input value: 3 - clearance of the multitool locations as an angle.

Entries	Meaning
DIDisplayMode	• DisplayDistributed - default setting, the setup offsets EC are displayed in the tool list and the additive offsets SC are displayed in the tool wear list. • EclnWearList - the setup offsets EC are displayed in the tool wear list and the additive offsets SC are not displayed.
CombinedToolMonitoring	Specifies the combined monitoring types offered in the toggle field: • Time/Count - tool life and workpiece count • Time/Wear - tool life and wear monitoring • Count/Wear - workpiece count and wear monitoring • Time/Count/Wear - tool life, workpiece count and wear monitoring
FiltersCombinedWithOr	Defines the filter criteria, which are linked in the filter window using a logical OR. The default setting for all the filter criteria is a logical AND. The possible entries are: • "StandBy" - for tools that are ready for use • "Active" - for tools with active identifier • "Locked" - for locked tools • "Prewarn" - for the prewarning limit • "RemainingPieces" - for the remaining unit quantity • "RemainingTime" - for the remaining tool time • "MarkedToUnload" - for the unload identifier • "MarkedToLoad" - for the load identifier
MagazineMoveMessage	True - "Magazine movement running" or "Magazine movement ended" message is displayed. False - default setting, message is not displayed.
ToolLoadErrorMessage	Refers to the output of the fault message due to a negative PLC. True - default setting, the fault message is displayed. False - the fault message is not displayed.

The following settings refer to markings in the lists:

True - marking is displayed.

False - marking is not displayed (default setting).

If several markings apply to one location, the most important information is displayed. For instance, if a tool is locked, and at the same time, the "Tool position" and the "Reference

direction holder angle" for this tool do not match, then a red cross is displayed for locked. The yellow triangle for the conflict cannot be seen.

Entries	Meaning
ShowSymbolActivePlace	Marking of the magazine location at the machining position. True - default setting applies for the "Revolver" magazine type. False - setting for all other magazine types.
ShowSymbolActiveMultiToolPlace	Marking of the multitool location at the machining position. Is only displayed when the multitool is located in the spindle or the toolholder. True - default setting.
ShowSymbolLoadPlace	Selection of the magazine location at the loading position. True - default setting applies for all magazine types (except Revolver is true). False - setting for the "Revolver" magazine type.
ShowSymbolChangeInSpindlePlace	Marking of the magazine location at the change position. True - default setting applies for all magazine types (except Revolver is true). False - setting for the "Revolver" magazine type.
ShowSymbolActiveTool	Marking of the active tool. False - default setting.
ShowSymbolProgrammedTool	Marking of the follow-on tool. True - default setting.
ShowSymbolActiveDNo	Marking of the active cutting edge. False - default setting.
ShowSymbolActiveDNoInTool	Marking of the active tool by marking the active cutting edge. True - default setting for configuration without magazine management. False - setting for configuration with magazine management.
ShowSymbolActiveDIno	Marking of the active DL number. True - default setting.
ShowConflictPositionAndDirection	Marking of a conflict; tool position and cutting direction do not match. True - default setting.
ShowConflictActiveToolWarning	Marking of a conflict; revolver magazine type and the active tool are not at the the machining position. True - default setting.
ShowSymbolActiveIsoHNoL1	Marking of the active H number for length 1 in the ISO tool list. true: Default setting.
ShowSymbolActiveIsoHNoL2	Marking of the active H number for length 2 in the ISO tool list. True - default setting.
ShowSymbolActiveIsoHNoL3	Marking of the active H number for length 3 in the ISO tool list. True - default setting.

Entries	Meaning
ShowSymbolActiveIsoDNo	Marking of the active D number in the ISO tool list. True - default setting.
ShowSymbolActiveIsoHDNo	Marking of the active HD number in the ISO tool list. True - default setting.

Example

```

<SETTINGS>
    <ReactivateWithMagPos value="false" type="bool" />
    <MagazineMoveMessage value="false" type="bool" />
    <CreateNewToolDialog value="true" type="bool" />
    <MagazineSelectionDialog value="false" type="bool" />
    <AccessLevelWriteDetailsAllParameter value="4" type="int" />
</SETTINGS>

```

12.2.2 Configure the tool lists

<LISTCONFIGURATION> tag

The entries used for configuring the following list views are located in the <LISTCONFIGURATION> tag:

- "Tool List" window, identifier <SlTmToollistForm>, see the 1st example.
- "Tool Wear" window, <SlTmToolwearForm> identifier
- "Magazine" window, <SlTmToolmagazinForm> identifier
- "Tool Data OEM" window, <SlTmTooloemForm> identifier
This window is hidden in the default configuration. The window is only displayed when you enable it with <Enabled>, see the 3rd example.
- "Tool Selection" window, <SlTmToolselectionForm> identifier: In order to select a tool this window can be displayed in the program input screen.

Entry	Meaning
Enabled	True - window is displayed. False - window is hidden.
CAPTION	Definition of the list name. See 1st example: "Tool list" appears on the user interface for the "TM_CAPTION_TOOLLIST" identifier.
COLUMNX	Column selection: X stands for the column number.
TOOLTYPE_ALL	The column setting applies to all tool types.
TOOLTYPE_XXX	The column setting applies to a particular tool type. XXX stands for the number of the tool type.

Entry	Meaning
EMPTY_MAG_PLACE	The column setting applies to an empty magazine location. Empty magazine locations appear in a list sorted according to magazine.
Item	Identifier of the list parameter. See example: The column in which the magazine number and magazine location are displayed is created for the "ToolInMagInPlace" identifier. Tool parameter identifiers (Page 138) Cutting parameter identifiers (Page 140) Monitoring parameter identifiers (Page 142) Magazine location parameter identifiers (Page 144)
DELETE_COLUMN	Delete column.
INSERT_COLUMN	Insert column.

Note

Independent of their programming, the entries are effective in the following sequence:

1. COLUMNX
2. DELETE_COLUMN
3. INSERT_COLUMN

Note

The Enabled entry does not apply for the "Tool Selection" window. The other entries apply.

Example 1: Configuring the magazine list - magazine location and tool location type**Note**

The columns of the list configuration cannot be taken from the display in the user interface in all cases because some columns are automatically inserted and therefore not included in the count of the configuration.

Variant 1

Column 10 and column 11 are overwritten. If column 10 and column 11 do not exist, the new columns will be added to the existing columns.

```
<LISTCONFIGURATION>
  <!-- **** Magazine list *** -->
  <SlTmToolmagazinForm>
    <!-- **** Column 10 *** -->
    <COLUMN10>
      <!-- **** For all tool types *** -->
      <TOOLTYPE_ALL>
```

```

        <!-- **** Display magazine location type *** -->
        <Item value="MagPlaceType" type="QString" />
    </TOOLTYPE_ALL>
</COLUMN10>
<!-- **** Column 11 *** -->
<COLUMN11>
    <!-- **** For all tool types *** -->
    <TOOLTYPE_ALL>
        <!-- **** Display tool location type *** -->
        <Item value="ToolPlaceSpec" type="QString" />
    </TOOLTYPE_ALL>
    <!-- **** For all multitools *** -->
    <TOOLTYPE_9997>
        <!-- **** Display for all multitool location types *** -->
        <Item value="MultiToolPlaceSpec" type="QString" />
    </TOOLTYPE_9997>
</COLUMN11>
</SlTmToolmagazinForm>
</LISTCONFIGURATION>

```

Variant 2

The new columns are inserted in the standard configuration between the "Magazine location locked" and "Oversized tool" columns.

```

<LISTCONFIGURATION>
    <!-- **** Magazine list *** -->
    <SlTmToolmagazinForm>
        <!-- **** Insert column *** -->
        <INSERT_COLUMN>
            <!-- **** Column 7 *** -->
            <COLUMN7>
                <!-- **** For all tool types *** -->
                <TOOLTYPE_ALL>
                    <!-- **** Display magazine location type *** -->
                    <Item value="MagPlaceType" type="QString" />
                </TOOLTYPE_ALL>
            </COLUMN7>
            <!-- **** Column 8 *** -->
            <COLUMN8>
                <!-- **** For all tool types *** -->
                <TOOLTYPE_ALL>
                    <!-- **** Display tool location type *** -->
                    <Item value="ToolPlaceSpec" type="QString" />
                </TOOLTYPE_ALL>
            </COLUMN8>
        </INSERT_COLUMN>
    </SlTmToolmagazinForm>
</LISTCONFIGURATION>

```

```

<!-- **** For all multitools *** -->
<TOOLTYPE_9997>
  <!-- **** Display for all multitool location types *** -->
  <Item value="MultiToolPlaceSpec" type="QString" />
</TOOLTYPE_9997>
</COLUMN8>
</INSERT_COLUMN>
</SlTmToolmagazinForm>
</LISTCONFIGURATION>

```

Example 2: Configuring the magazine list - "Oversized tool" column is deleted

Note

The columns of the list configuration cannot be taken from the display in the user interface in all cases because some columns are automatically inserted and therefore not included in the count of the configuration.

```

<LISTCONFIGURATION>
  <!-- **** Magazine list *** -->
  <SlTmToolmagazinForm>
    <!-- **** Delete column 7 *** -->
    <DELETE_COLUMN>
      <COLUMN7 />
    </DELETE_COLUMN>
  </SlTmToolmagazinForm>
</LISTCONFIGURATION>

```

Example 3: Enabling the OEM tool data list

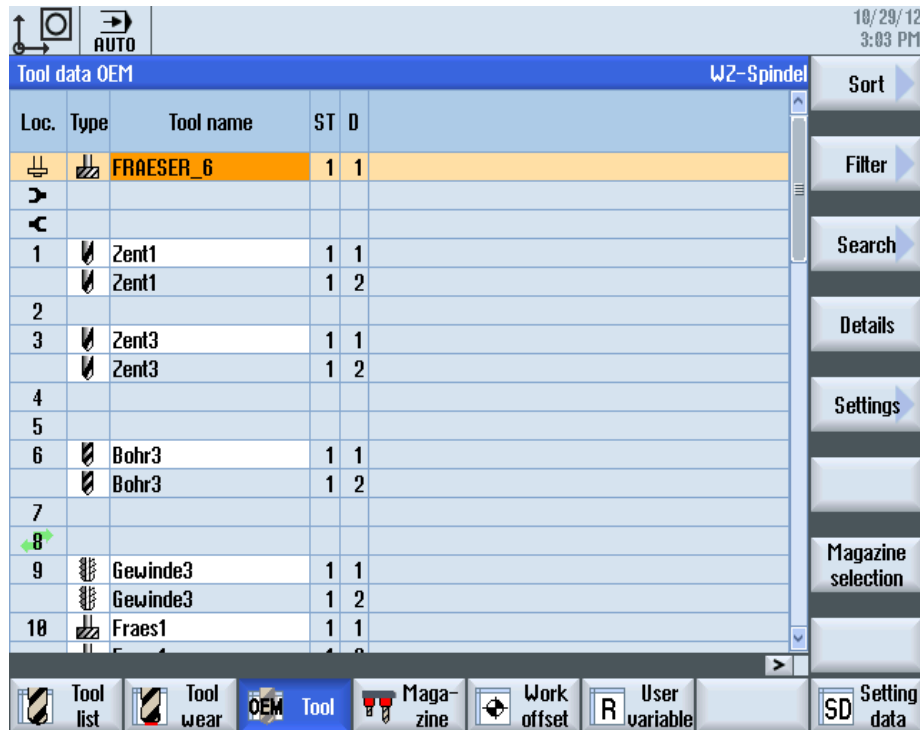
```

<SlTmTooloemForm>
  <Enabled value="true" type="bool" />
</SlTmTooloemForm>

```

Result

- The "OEM Tool Data" window is displayed. It automatically displays the columns shown in the figure.
- The new "OEM tools" softkey appears and can be operated.



12.2.2.1 Tool parameter identifiers

Tool parameters

For the tool parameters, the following identifiers are used in the configuration file:

Identifier	Parameter or property	System variable *)
Empty	Empty field	-
ToolNo	Number of the tool	-
AdaptNo	Adapter number assignment	\$TC_MPP8
DuploNo	Duplo number	\$TC_TP1
DuploNoRW	Duplo number, read and write access	\$TC_TP1
NumCuttEdges	Number of cutting edges of tool	\$P_TOOLND
ToolIdent	Tool name	\$TC_TP2
ToolIdentRO	Tool name, read only access	\$TC_TP2
ToolInfo	Tool information	\$TC_TP11
ToolInMag	Magazine in which the tool is located	\$A_TOOLMN
ToolInPlace	Magazine location containing the tool	\$A_TOOLMLN

Identifier	Parameter or property	System variable ^{*)}
ToolInMagInPlace	Magazine number/Magazine location	\$A_TOOLMN / \$A_TOOL_MLN
ToolMonTCW	Type of tool monitoring, tool life, workpiece count and wear. If wear monitoring has not been enabled via the machine data, this parameter is handled like ToolMonTC.	\$TC_TP9
ToolMonTC	Type of tool monitoring, tool life and workpiece count	\$TC_TP9
ToolPlaceSpec	Location type	\$TC_TP7
ToolPlaceSpecIdent	Location type, if it is to be displayed as text. The precondition is that the text for the location type was configured.	\$TC_TP7
ToolSearch	Tool search type for replacement tools	\$TC_TP10
ToolMyMag	Owner magazine of the tool	\$A_MYMN
ToolMyPlace	Owner magazine location of the tool	\$A_MYMLN
ToolSizeLeft	Tool size left in half locations	\$TC_TP3
ToolSizeRight	Tool size right in half locations	\$TC_TP4
ToolSizeUpper	Tool size upper in half locations	\$TC_TP5
ToolSizeDown	Tool size lower in half locations	\$TC_TP6
ToolOverSize	Tool size as fixed setting - two half locations left, two half locations right, one half location top, one half location bottom	\$TC_TP3 - \$TC_TP6
ToolState	Tool status in the hexadecimal notation	\$TC_TP8
ToolStateActiv	Tool status - tool active	\$TC_TP8, Bit0
ToolStateEnabled	Tool status - tool released	\$TC_TP8, Bit1
ToolStateLocked	Tool status - tool locked	\$TC_TP8, Bit2
ToolStateLockedRO	Tool status - tool locked, only read access	\$TC_TP8, Bit2
ToolStateMeasured	Tool status - tool measured	\$TC_TP8, Bit3
ToolStatePrewarn	Tool status - tool has reached prewarning limit	\$TC_TP8, Bit4
ToolStatePrewarnRO	Tool status - tool has reached prewarning limit, only read access	\$TC_TP8, Bit4
ToolStateInChange	Tool status - tool is being changed	\$TC_TP8, Bit5
ToolStateFixed	Tool status - tool fixed place coded	\$TC_TP8, Bit6
ToolStateUsed	Tool status - tool was in use	\$TC_TP8, Bit7
ToolStateAutoReturn	Tool status - automatic return	\$TC_TP8
ToolStateIgnoreLocked	Tool status - do not observe locked	\$TC_TP8, Bit9
ToolStateMarkedToUnload	Tool status - tool is marked to unload	\$TC_TP8, Bit10
ToolStateMarkedToLoad	Tool status - tool is marked to load	\$TC_TP8, Bit11
ToolStatePermanent	Tool status - tool is permanent	\$TC_TP8, Bit12
ToolState1To1Exchange	Tool status - one to one replacement	\$TC_TP8, Bit14
ToolStateHandTool	Tool status - manual tool	\$TC_TP8, Bit15
ToolProtAreaFile	File name of the tool description which is required to define the protection zone	\$TC_TP_PROTA
ToolMaxVelo	Maximum speed	\$TC_TP_MAX_VELO

Identifier	Parameter or property	System variable ^{*)}
ToolMaxAcc	Maximum acceleration	\$TC_TP_MAX_ACC
ToolInMultiTool	Multitool in which the tool is located	\$A_TOOLMTN
ToolInMultiToolPlace	Multitool location containing the tool	\$A_TOOLMTLN
ToolMyMultiTool	Owner multitool of the tool	\$A_MYMTN
ToolMyMultiToolPlace	Owner multitool location of the tool	\$A_MYMTLN
ToolAlarmsExtended	PLC tool status - function "extended alarm" active	-
ToolAlarmLimit	PLC tool status - alarm limit reached	-
ToolExtAlarmLimit	PLC tool status - limit "extended alarm" reached	-
ToolUser_1, ... ToolUser_10	OEM tool parameters 1 to 10	\$TC_TPC1, ... \$TC_TPC10
ToolAppl_1, ... ToolAppl_10	Siemens application tool parameters 1 to 10	\$TC_TPCS1, ... \$TC_TPCS10

^{*)} Parameter corresponds to the system variable or the system variable is the basis for this parameter.

Note

Tool name "ToolIdent"

The maximum length of tool names is 31 ASCII characters.

The number of characters is reduced for Asian characters or Unicode characters.

The following special characters are not permitted: | # "

12.2.2.2 Cutting parameter identifiers

For the cutting parameters, the following identifiers are used in the configuration file:

Identifier	Parameter or property	System variable ^{*)}
EdgeNo	D number	-
EdgeNoIdx	Cutting edge number	-
ToolType	Tool type, cutting parameter 1	\$TC_DP1
CuttEdgePos	Cutting edge position, cutting parameter 2	\$TC_DP2
GeoLength1	Length 1, cutting parameter 3	\$TC_DP3
GeoLengthGeoAx1	Length of geometry axis 1, cutting parameter 3, e.g. length X	\$TC_DP3
GeoLength	Length, cutting parameter 3	\$TC_DP3
GeoLength2	Length 2, cutting parameter 4	\$TC_DP4
GeoLengthGeoAx3	Length of geometry axis 3, cutting parameter 4, e.g. length Z	\$TC_DP4
GeoLength3	Length 3, cutting parameter 5	\$TC_DP5
GeoLengthGeoAx2	Length of geometry axis 2, cutting parameter 5, e.g. length Y	\$TC_DP5
GeoRadius	Radius, cutting parameter 6	\$TC_DP6
GeoCornerRadius	Corner radius, cutting parameter 7	\$TC_DP7

Identifier	Parameter or property	System variable *)
GeoOutsideRadius	Outside radius, cutting parameter 7	\$TC_DP7
GeoBoomLength	Cantilever length, cutting parameter 7	\$TC_DP7
GeoLength4	Length 4, cutting parameter 8	\$TC_DP8
PlateLength	Cutting tip length, cutting parameter 8	\$TC_DP8
GeoLength5	Length 5, cutting parameter 9	\$TC_DP9
GeoWidth	Width, cutting parameter 9	\$TC_DP9
PlateWidth	Cutting tip width, cutting parameter 9	\$TC_DP9
GeoPitch	Thread pitch, cutting parameter 9	\$TC_DP9
BoreRadius	Hole radius, cutting parameter 9	\$TC_DP9
GeoAngle1	Angle 1, cutting parameter 10	\$TC_DP10
HolderAngle	Holder angle, cutting parameter 10	\$TC_DP10
CorrectionAngle	Correction angle to align the probe, cutting parameter 10	\$TC_DP10
GeoAngle2	Angle 2, cutting parameter 11	\$TC_DP11
AngleConicalMillTool	Angle conical mill tools, cutting parameter 11	\$TC_DP11
CuttDirection	Reference direction holder angle, cutting parameter 11	\$TC_DP11
WearLength1	Wear length 1, cutting parameter 12	\$TC_DP12
WearLengthGeoAx1	Wear length of geometry axis 1, cutting parameter 12, e.g. Δ length X	\$TC_DP12
WearLength	Wear length, cutting parameter 12	\$TC_DP12
WearLength2	Wear length 2, cutting parameter 13	\$TC_DP13
WearLengthGeoAx3	Wear length of geometry axis 3, cutting parameter 13, e.g. Δ length Z	\$TC_DP13
WearLength3	Wear length 3, cutting parameter 14	\$TC_DP14
WearLengthGeoAx2	Wear length of geometry axis 2, cutting parameter 14, e.g. Δ length Y	\$TC_DP14
WearRadius	Wear radius, cutting parameter 15	\$TC_DP15
WearCornerRadius	Wear corner radius, cutting parameter 16	\$TC_DP16
WearLength4	Wear length 4, cutting parameter 17	\$TC_DP17
WearLength5	Wear length 5, cutting parameter 18	\$TC_DP18
WearAngle1	Wear angle 1, cutting parameter 19	\$TC_DP19
WearAngle2	Wear angle 2, cutting parameter 20	\$TC_DP20
AdaptLength1	Adapter length 1, cutting parameter 21	\$TC_DP21
AdaptLengthGeoAx1	Adapter length of geometry axis 1, cutting parameter 21, e.g. adapter length X	\$TC_DP21
AdaptLength	Adapter length, cutting parameter 21	\$TC_DP21
AdaptLength2	Adapter length 2, cutting parameter 22	\$TC_DP22
AdaptLengthGeoAx3	Adapter length of geometry axis 3, cutting parameter 22, e.g. adapter length Z	\$TC_DP22
AdaptLength3	Adapter length 3, cutting parameter 23	\$TC_DP23
AdaptLengthGeoAx2	Adapter length of geometry axis 2, cutting parameter 23, e.g. adapter length Y	\$TC_DP23
ReliefAngle	Relief angle, cutting parameter 24	\$TC_DP24

Identifier	Parameter or property	System variable ^{*)}
PlateAngle	Cutting tip angle, combination of cutting parameter 24 and cutting parameter 10	\$TC_DP24 and \$TC_DP10
NoseAngle	Nose angle, cutting parameter 24	\$TC_DP24
CuttRate	Cutting rate, cutting parameter 25	\$TC_DP25
SpindleDirection	For rotating tools, direction of rotation of the tool spindle. For turning tools, direction of rotation of the main spindle	\$TC_DP25 bit8 and bit9
Coolant1	Coolant 1	\$TC_DP25 Bit10
Coolant2	Coolant 2	\$TC_DP25 Bit11
MFunction1	M function 1	\$TC_DP25 Bit0
MFunction2	M function 2	\$TC_DP25 Bit1
MFunction3	M function 3	\$TC_DP25 Bit2
MFunction4	M function 4	\$TC_DP25 Bit3
IsoHNoDPH	ISO H number	\$TC_DPH
OrientNo	Cutting orientation	\$TC_DPV
OrientV1	Cutting orientation vector 1	\$TC_DPV3
OrientV2	Cutting orientation vector 2	\$TC_DPV4
OrientV3	Cutting orientation vector 3	\$TC_DPV5
OrientVGeoAx1	Cutting orientation vector of geometry axis 1, cutting orientation parameter 3, e.g. vector X	\$TC_DPVN3
OrientVGeoAx3	Cutting orientation vector of geometry axis 3, cutting orientation parameter 4, e.g. vector Z	\$TC_DPVN4
OrientVGeoAx2	Cutting orientation vector of geometry axis 2, cutting orientation parameter 5, e.g. vector Y	\$TC_DPVN5
TeethCount	Number of teeth, cutting parameter 34	\$TC_DPNT
UniqueDNo	Unique D number	\$TC_DPCE
EdgeUser_1, ..., Edge-User_10,	OEM cutting parameters 1 to 10	\$TC_DPC1, ... \$TC_DPC10
EdgeAppl_1, ..., Edge-Appl_10x	Siemens application cutting parameters 1 to 10	\$TC_DPCS1, ... \$TC_DPCS10

^{*)} Parameter corresponds to the system variable or the system variable is the basis for this parameter.

12.2.2.3 Monitoring parameter identifiers

For the monitoring parameters, the following identifiers are used in the configuration file:

Identifier	Parameter or property	System variable ^{*)}
SupWarning	Warning limits, taking the current monitoring type into consideration	-
SupRemaining	Actual value, taking the current monitoring type into consideration	-
SupDesired	Setpoint, taking the current monitoring type into consideration	-
SupWarningTime	Tool life warning limit	\$TC_MOP1
SupRemainingTime	Tool life actual value	\$TC_MOP2

Identifier	Parameter or property	System variable ^{*)}
SupWarningPieces	Prewarning limit for workpiece count	\$TC_MOP3
SupRemainingPieces	Actual value for workpiece count	\$TC_MOP4
SupDesiredTime	Setpoint for tool life	\$TC_MOP11
SupDesiredPieces	Setpoint for workpiece count	\$TC_MOP13
SupWarningWear	Prewarning limit for wear	\$TC_MOP5
SupRemainingWear	Actual value for wear	\$TC_MOP6
SupDesiredWear	Setpoint for wear	\$TC_MOP15
SupExtendedAlarm	PLC tool management: Extended alarm	-
SupWarningPiecesIncremental	PLC tool management: Prewarning limit for workpiece count	\$TC_MOP13 - \$TC_MOP3
SupActualPieces	PLC tool management: Actual value for workpiece count	\$TC_MOP13 - \$TC_MOP4
EdgeSupUser_1, ... EdgeSupUser_10	OEM cutting edge monitoring parameters 1 to 10	\$TC_MOPC1, ... \$TC_MOPC10
EdgeSupAppl_1, ... Edge- SupAppl_10	Siemens application cutting edge monitoring parameters 1 to 10	\$TC_MOPCS1, ... \$TC_MOPCS10

^{*)} Parameter corresponds to the system variable or the system variable is the basis for this parameter.

12.2.2.4 Grinding parameter identifiers

For grinding parameters, the following identifiers are used in the configuration file:

Identifier/parameter	Meaning	System variable ^{*)}
GrindingSpindleNo	Spindle number	\$TC_TPG1
GrindingConnectionRule	Chaining rule	\$TC_TPG2
MinimalDiscRadius	Minimum grinding wheel radius	\$TC_TPG3
ActualDiscRadius	Actual grinding wheel radius (read access only)	-
MinimalDiscWidth	Minimum grinding wheel width	\$TC_TPG4
ActualDiscWidth	Actual grinding wheel width	\$TC_TPG5
MaximalDiscSpeed	Maximum grinding wheel speed	\$TC_TPG6
MaximalDiscPeripheralVelocity	Maximum grinding wheel peripheral velocity	\$TC_TPG7
AngleBevelDisc	Angle of inclined wheel	\$TC_TPG8
ParamForRadiusCalculation	Parameters to calculate the radius	\$TC_TPG9
	Directory of the dressing program	\$TC_TPG_DRSPATH
	Dressing program name	\$TC_TPG_DRSPROG

^{*)} Parameter corresponds to the system variable or the system variable is the basis for this parameter.

The following parameters are assigned to the cutting edges of the grinding tool:

Identifier/parameter	Meaning
GrindingMonitoring	Grinding-specific tool monitoring
GrindingUseBaseLength	Taking into account the basis dimension when calculating the grinding wheel radius

12.2.2.5 Magazine location parameter identifiers

For the magazine location parameters, the following identifiers are used in the configuration file:

Identifier	Parameter or property	System variable ^{*)}
MagPlaceKind	Location kind	\$TC_MPP1
MagPlaceType	Location type	\$TC_MPP2
MagPlaceTypeIdent	Location type as text	\$TC_MPP2
MagPlaceTNo	Tool number of the tool in this location	\$TC_MPP6
MagPlaceWatchNeighbour	Consider adjacent location	\$TC_MPP3
MagPlaceStateLocked	Magazine location locked	\$TC_MPP4, Bit0
MagPlaceStateEmpty	Magazine location free	\$TC_MPP4, Bit1
MagPlaceStateResInterMag	Reserved for tool in buffer	\$TC_MPP4, Bit2
MagPlaceStateResLoadTool	Reserved for tool to be loaded	\$TC_MPP4, Bit3
MagPlaceStateOccupiedLeft	Magazine location occupied, left half location	\$TC_MPP4, Bit4
MagPlaceStateOccupiedRight	Magazine location occupied, right half location	\$TC_MPP4, Bit5
MagPlaceStateOccupiedUpper	Magazine location occupied, upper half location	\$TC_MPP4, Bit6
MagPlaceStateOccupiedDown	Magazine location occupied, lower half location	\$TC_MPP4, Bit7
MagPlaceStateReservedLeft	Magazine location reserved, left half location	\$TC_MPP4, Bit8
MagPlaceStateReservedRight	Magazine location reserved, right half location	\$TC_MPP4, Bit9
MagPlaceStateReservedUpper	Magazine location reserved, upper half location	\$TC_MPP4, Bit10
MagPlaceStateReservedDown	Magazine location reserved, lower half location	\$TC_MPP4, Bit11
MagPlaceMagazineNo	Magazine number	-
MagPlaceTypeIdx	Type index	\$TC_MPP5
MagPlaceWearGroup	Wear group number	\$TC_MPP5
MagPlaceAdaptNo	Adapter number	\$TC_MPP7
MagPlaceNo	Magazine location number	-
MagNoMagPlaceNo	Magazine number/Magazine location number	-
MagPlaceUser_1, ... MagPlaceUser_10	OEM magazine location parameters 1 to 10	\$TC_MPPC1, ... \$TC_MPPC10
MagPlaceAppl_1, ... MagPlaceAppl_10	Siemens magazine location parameters 1 to 10	\$TC_MPPCS1, ... \$TC_MPPCS10

^{*)} Parameter corresponds to the system variable or the system variable is the basis for this parameter.

12.2.2.6 Magazine location adapter parameter identifiers

The following identifiers are used in the configuration file for the magazine location adapter parameters:

Identifier	Parameter or property	System variable ^{*)}
MagPlaceAdapterLength1	Magazine location adapter length 1	\$TC_ADPT1
MagPlaceAdapterLengthGeoAx1	Magazine location adapter geometry axis 1	\$TC_ADPT1
MagPlaceAdapterLength2	Magazine location adapter length 2	\$TC_ADPT2
MagPlaceAdapterLengthGeoAx3	Magazine location adapter geometry axis 3	\$TC_ADPT2
MagPlaceAdapterLength3	Magazine location adapter length 3	\$TC_ADPT3
MagPlaceAdapterLengthGeoAx2	Magazine location adapter geometry axis 2	\$TC_ADPT3
MagPlaceAdapterTrafoNo	Magazine location adapter transformation number	\$TC_ADPT4

^{*)} Parameter corresponds to the system variable or the system variable is the basis for this parameter.

12.2.2.7 Multitool parameter identifiers

For the multitool parameters, the following identifiers are used in the configuration file:

Identifier	Parameter or property	System variable ^{*)}
MultiToolNo	Number of the multitool	-
MultiToolNumberOfPlaces	Number of multitool locations	\$TC_MTPN
MultiToolNumberOfPlacesRO	Number of multitool locations, only read access	\$TC_MTPN
MultiToolIdent	Name of the multitool	\$TC_MTP2
MultiToolInMag	Magazine in which the multitool is located	\$A_TOOLMN
MultiToolInPlace	Magazine location containing the multitool	\$A_TOOLMLN
MultiToolInMagInPlace	Magazine number/Magazine location	\$A_TOOLMN / \$A_TOOLMLN
MultiToolMyMag	Owner magazine of the multitool	\$A_MYMN
MultiToolMyPlace	Owner magazine location of the multitool	\$A_MYMLN
MultiToolPlaceSpec	Location type	\$TC_MTP7
MultiToolPlaceSpecIdent	Location type as text	\$TC_MTP7
MultiToolSizeLeft	Multitool size to the left in half locations	\$TC_MTP3
MultiToolSizeRight	Multitool size to the right in half locations	\$TC_MTP4
MultiToolSizeUpper	Multitool size to the top in half locations	\$TC_MTP5
MultiToolSizeDown	Multitool size to the bottom in half locations	\$TC_MTP6
MultiToolOverSize	Multitool size as fixed setting - two half locations left, two half locations right, one half location top, one half location bottom	\$TC_MTP3 - \$TC_MTP6
MultiToolPosition	Multitool position	\$TC_MTP_POS
MultiToolProtAreaFile	File name of the multitool description which is required to determine the protection zone	\$TC_MTP_PROTA
MultiToolKindOfDist	Type of distance input of the multitool locations	\$TC_MTP_KD

Identifier	Parameter or property	System variable ^{*)}
MultiToolKindOfDistRO	Type of distance input of the multitool locations, only read access	\$TC_MTP_KD
MultiToolDistAngle	Type of distance input of the multitool locations as an angle	\$TC_MTP_KD
MultiToolState	Multitool status in the hexadecimal notation	\$TC_MTP8
MultiToolStateEnabled	Multitool status - multitool released	\$TC_MTP8, Bit1
MultiToolStateLocked	Multitool status - multitool locked	\$TC_MTP8, Bit2
MultiToolStateLockedRO	Multitool status - multitool locked, only read access	\$TC_MTP8, Bit2
MultiToolStateInChange	Multitool status - multitool is being changed	\$TC_MTP8, Bit5
MultiToolStateFixed	Multitool status - multitool fixed location coded	\$TC_MTP8, Bit6
MultiToolStateUsed	Multitool status - multitool was in use	\$TC_MTP8, Bit7
MultiToolStateAutoReturn	Multitool status - automatic return	\$TC_MTP8, Bit8
MultiToolStateIgnoreLocked	Multitool status - do not observe locked status	\$TC_MTP8, Bit9
MultiToolStateMarkedToUnload	Multitool status - multitool is selected to be unloaded	\$TC_MTP8, Bit10
MultiToolStateMarkedToLoad	Multitool status - multitool is selected to be loaded	\$TC_MTP8, Bit11
MultiToolStatePermanent	Multitool status - multitool is a master tool	\$TC_MTP8, Bit12
MultiToolState1To1Exchange	Multitool status - one to one replacement	\$TC_MTP8, Bit14
MultiToolStateHandTool	Multitool status - manual tool	\$TC_MTP8, Bit15
MultiToolStateLockMtlIfToolLock	Multitool status - multitool is locked if a tool is locked in multitool	\$TC_MTP8, Bit16
MultiToolUser_1, ... MultiToolUser_10	OEM multitool parameters 1 to 10	\$TC_MTPC1, ... \$TC_MTPC10
MultiToolAppl_1, ... MultiToolAppl_10	Siemens application multitool parameters 1 to 10	\$TC_MTPCS1, ... \$TC_MTPCS10

^{*)} Parameter corresponds to the system variable or the system variable is the basis for this parameter.

12.2.2.8 Multitool location parameter identifiers

For the multitool location parameters, the following identifiers are used in the configuration file:

Identifier	Parameter or property	System variable ^{*)}
MultiToolDist	Multitool location distance taking into account the actual type of distance input of the multitool	-
MultiToolDistLength	Multitool location distance, length	\$TC_MTPPL
MultiToolDistAngle	Multitool location distance angle, angle	\$TC_MTPPA
MultiToolPlaceType	Location type	\$TC_MTPP2
MultiToolPlaceTypeIdent	Location type as text	\$TC_MTPP2
MultiToolPlaceState	Multitool status in the hexadecimal notation	\$TC_MTPP4
MultiToolPlaceStateLocked	Multitool location locked	\$TC_MTPP4, Bit0

Identifier	Parameter or property	System variable *)
MultiToolPlaceStateEmpty	Multitool location empty	\$TC_MTPP4, Bit1
MultiToolPlaceTNo	Tool number of the tool in this multitool location	\$TC_MTPP6
MultiToolPlaceAdaptNo	Adapter number	\$TC_MTPP7
MultiToolPlaceNo	Multitool location number	-
MultiToolPlaceUser_1, ... MultiToolPlaceUser_10	OEM multitool location parameters 1 to 10	\$TC_MTPPC1, ... \$TC_MTPPC10
MultiToolPlaceAppl_1, ... MultiToolPlaceAppl_10	Siemens application multitool location parameters 1 to 10	\$TC_MTPPCS1, ... \$TC_MTPPCS10

*) Parameter corresponds to the system variable or the system variable is the basis for this parameter.

12.2.3 Configure the list parameters

<PARAMETERCONFIGURATION> tag

In the <PARAMETERCONFIGURATION> tag, you have the following options:

1. Changing list parameters.
2. Creating new list parameters based on an existing parameter.

Changing list parameters

You can change all the parameters included in the system apart from the ISO parameters.

Tool parameter identifiers (Page 138)

Cutting parameter identifiers (Page 140)

Monitoring parameter identifiers (Page 142)

Grinding parameter identifiers (Page 143)

Magazine location parameter identifiers (Page 144)

Magazine location adapter parameter identifiers (Page 145)

Multitool parameter identifiers (Page 145)

Multitool location parameter identifiers (Page 146)

Entries	Meaning
HeadLine	Column header. Text entered is displayed as a header. Example: Text "Radius" is displayed as a header (default).
Tooltip	Text that is displayed in the tooltip. Example: Text "Geometry radius" is displayed in the tooltip (default).
ShortText	Text if the parameter is displayed in the "More Data" window. Example: Text "Rad." is displayed in the "More Data" window (default).
DetailsText	Text, if the parameter is displayed in the "Details" window. Example: Text "Radius" is displayed in the "Details" window (default).

Entries	Meaning	
Width	Column width in pixels in relation to a 640 x 480 resolution. See example below: The default column width is changed to 53 pixels.	
DisplayMode	Values that the parameter is displayed with. See example below: The default setting is changed to "DoubleMode". The following values can also be accepted:	
	AnyMode	All characters
	IntegerMode	Integers
	UnsignedIntegerMode	Unsigned integers
	DoubleMode	Values with decimal places The number of decimal places is specified in "DecimalPlaces"
	UnsignedDoubleMode	Unsigned values with decimal places The number of decimal places is specified in "DecimalPlaces"
	Length	Specified length
	Angle	Specified angle
	LinearFeedPerTime	Linear feed in mm/minute
	LinearFeedPerRevolution	Linear feed in mm/revolution
	LinearFeedPerTooth	Linear feed in mm/tooth
	RevolutionSpeed	Speed
	ConstantCuttingSpeed	Constant cutting rate
DecimalPlaces	Number of decimal places if "DoubleMode" or "UnsignedDoubleMode" is selected in the DisplayMode. See example: Two decimal places are entered.	
ItemType	Field types in which a value is displayed.	
	TextField	Input/output field for values and texts
	TextFieldReadOnly	Output field for values and texts
	CheckBox	Input/output field for states
	CheckBoxReadOnly	Output field for states
BitMask	Bit mask for the display of a bit from a value. The bit mask is entered as integer value. Bit 0 → 1, Bit 1 → 2, Bit 2 → 4	
AccessLevel	Value for the access level that is used for these parameters in the user interface. Value range 1 to 7	
UpperLimit	Upper input limit for these parameters in the user interface. Only applies for numerically entered parameters.	
LowerLimit	Lower input limit for these parameters in the user interface. Only applies for numerically entered parameters.	

Example

In the following example, the <GeoRadius> parameter is used.

Only enter the modified data.

```
<PARAMETERCONFIGURATION>
  <GeoRadius>
```



```

        <Width value="53" type="int" />
        <DisplayMode value="DoubleMode" type="QString" />
        <DecimalPlaces value="2" type="int" />
    </GeoRadius>
</PARAMETERCONFIGURATION>

```

Configuring new list parameters based on an existing list parameter.

Assign a new parameter name and enter only the changed data:

Entries	Meaning
	Assign new parameter name See 1st example: Text "NewGeoRadius"
Base	Name of the parameter on which the new parameter is based. See 1st example: "GeoRadius" parameter is used as a template.
	Also enter only the changed data here. All other data is taken from the existing parameter. See 1st example: Column width changed to 46 pixels. Number of decimal places changed to 1.

1. Example

```

<PARAMETERCONFIGURATION>
    <NewGeoRadius>
        <Base value="GeoRadius" type="QString" />
        <Width value="46" type="int" />
        <DisplayMode value="DoubleMode" type="QString" />
        <DecimalPlaces value="1" type="int" />
    </NewGeoRadius>
</PARAMETERCONFIGURATION>

```

2. Example

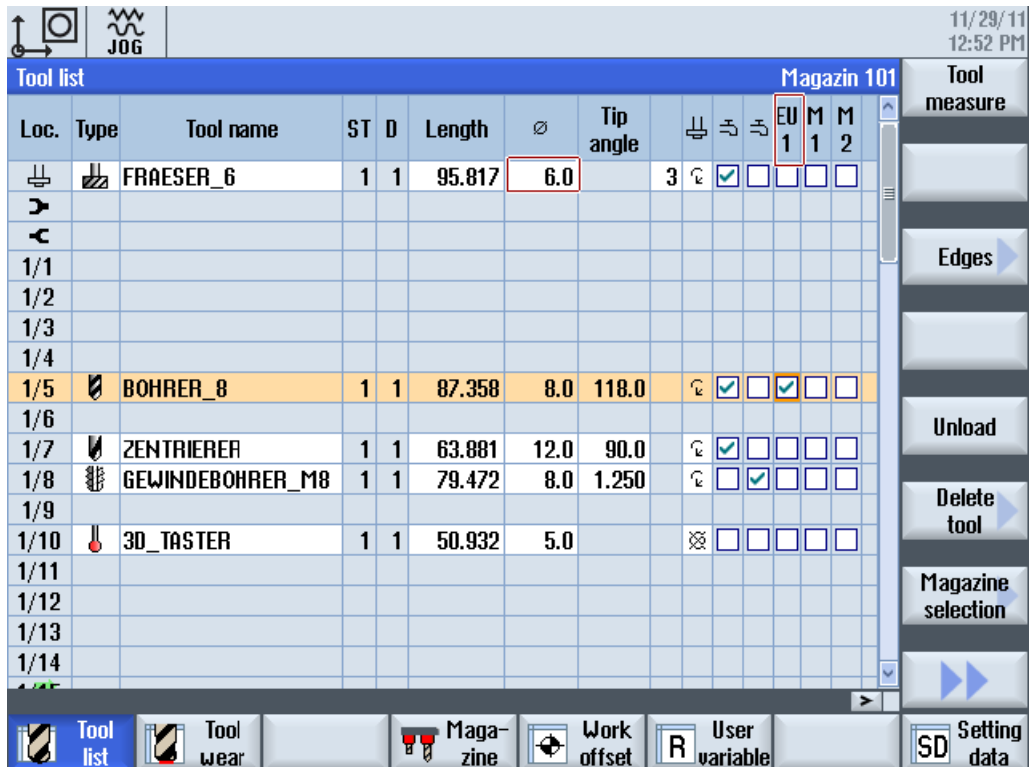
Entries	Meaning
	Assign new cutting edge parameter name The name is "EdgeUser_1_Bit0" in the following example.
Base	Name of the parameter on which the new parameter is based. In the following example, the "EdgeUser_1" parameter is used as template.

Entries	Meaning
	Also enter only the changed data here. All other data is taken from the existing cutting edge parameter.
	The following entries are changed: Field type: Input/output field for states Value output: All characters Bit mask: Bit 0 Column width changed to 17 pixels. Header changed to "TM_HL_EDGE_USER_1_Bit0" Text in the tooltip changed to "TM_TT_EDGE_USER_1_Bit0"

```

<PARAMETERCONFIGURATION>
  <EdgeUser_1_Bit0>
    <Base value="EdgeUser_1" type="QString" />
    <ItemType value="CheckBox" type="QString" />
    <DisplayMode value="AnyMode" type="QString" />
    <BitMask value="1" type="int" />
    <Width value="17" type="int" />
    <HeadLine value="TM_HL_EDGE_USER_1_Bit0" type="QString" />
    <ToolTip value="TM_TT_EDGE_USER_1_Bit0" type="QString" />
  </EdgeUser_1_Bit0>
</PARAMETERCONFIGURATION>

```



12.2.4 List of tool types

Coding of tool types for milling tools

Group with type 1xy (milling tool):

100	Milling tool according to CLDATA (DIN 66215)
110	Ballhead cutter (cylindrical die milling tool)
111	Ballhead cutter (tapered die milling tool)
120	End mill (without corner rounding)
121	End mill (with corner rounding)
130	Angle head cutter (without corner rounding)
131	Angle head cutter (with corner rounding)
140	Facing cutter
145	Thread cutter
150	Side mill
151	Saw
155	Bevel cutter (without corner rounding)
156	Bevel cutter (with corner rounding)
157	Tapered die-sinking cutter
160	Drill and thread milling cutter

Coding of tool types for drills

Group type 2xy (drills):

200	Twist drill
205	Solid drill
210	Boring bar
220	Center drill
230	Countersink
231	Counterbore
240	Tap regular thread
241	Tap fine thread
242	Tap Whitworth thread
250	Reamer

Coding of tool types for grinding tools

Group type 4xy (grinding tools):

400	Grinding wheel circumference
410	Face grinding wheel
490	Dresser
494	Dressing roll

495	Dressing roll with auxiliary axis
496	Dressing wheel
497	Dressing wheel with swivel axis

Grinding tool types 401, 402, 403 as well as 411, 412, 413 are not listed as tool types that can be selected. The grinding-specific tool monitoring - taking into account the basis dimension when calculating the grinding wheel radius - is set using dedicated parameters. These parameters act on the tool types in the usual form.

Coding of tool types for turning tools

Group type 5xy (turning tools):

500	Roughing tool
510	Finishing tool
520	Plunge cutter
530	Parting tool
540	Threading tool
550	Steel profile
560	Rotary drill (ECOCUT)
580	Orientated probe
585	Calibration tool

Coding of tool types for special tools

Group type 7xy (special tools):

700	Slotting saw
710	3D probe
711	Edge probe
712	Mono probe
713	L probe
714	Star probe
725	Calibration tool
730	Stop
731	Spindle sleeves
732	End support
900	Auxiliary tools

12.2.5 Configure tool types

<TOOLTYPECONFIGURATION> tag

The entries used for configuring the tool types are located in the <TOOLTYPECONFIGURATION> tag:

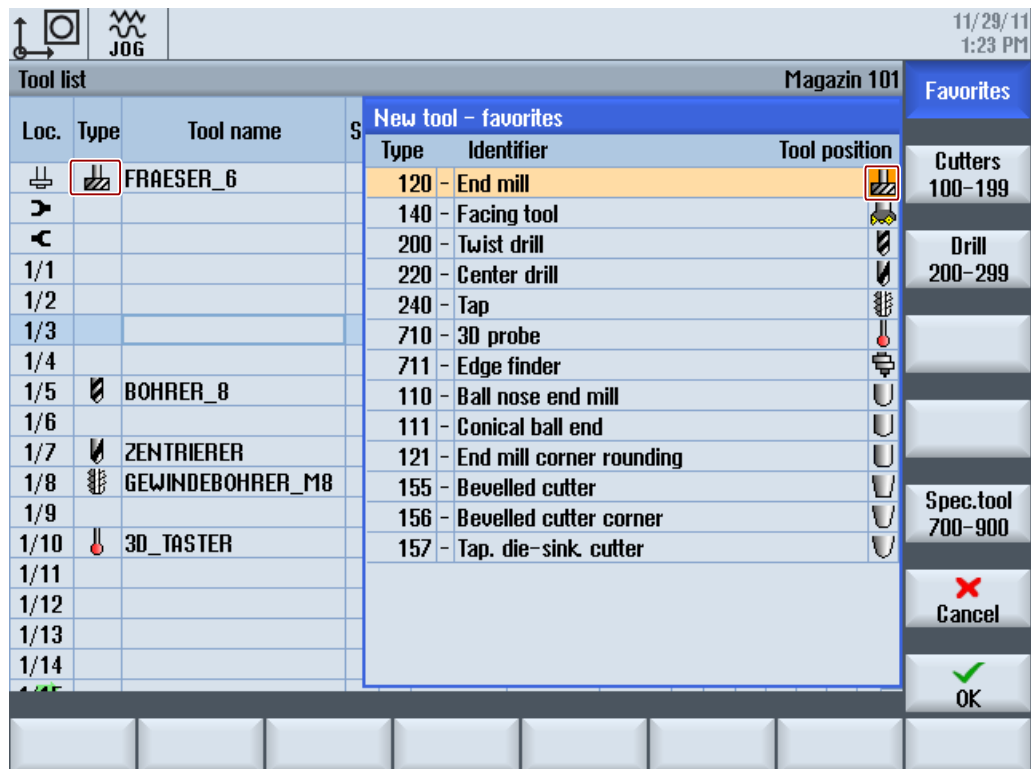
Entries	Meaning
TOOLTYPE_XXX	XXX stands for the number of the tool type. For the assignment of the tool type and number, please refer to Section "List of tool types (Page 151)".
Tooltype	Number (XXX) of the tool type
Name	Text identifier of the tool name. The tool name is displayed in the following windows: "New Tool - Favorites" "New Tool - Milling Cutter 100 - 199" "New Tool - Drill 200 - 299" "New Tool - Grinding Tool 400 - 499" "New Tool - Special and Auxiliary Tools 700 - 900" See example: For the text identifier "TM_PAR_SHANK_END_CUTTER", the name "End mill" appears on the user interface.
Shortname	Text identifier of the tool name. Tool name is displayed in the tool list. See example: For the text identifier "TM_PPTT_SHANK_END_CUTTER", the name "MILLING CUTTER" appears on the user interface.
Tooltip	Text identifier of the tool name. This tool name is displayed in the tooltip. See example: For the text identifier "TM_TTTT_SHANK_END_CUTTER", the name "End mill" appears on the user interface.
Icon9 - for milling technology	Icon representing the tool type. The icons are displayed in the following windows: "Tool List" in the "Type" column "New Tool - Favorites" in the "Tool position" column "New Tool - Milling Cutter 100 - 199" in the "Tool position" column "New Tool - Drill 200 - 299" in the "Tool position" column "New Tool - Special Tool 700 - 900" in the "Tool position" column. The icons are in ".png" format. They are stored according to image resolution in the following directories: /oem/sinumerik/hmi/ico/ico640, or ico800, or ico1024 /user/sinumerik/hmi/ico/ico640, or ico800, or ico1024
IconX - for the turning and grinding technologies	Icon representing the tool type (see Icon9 - for milling technology) Special feature - tool positions are supported in the turning and grinding technologies. X is for the tool position represented by the icon. Position 9 is an undefined position represented by a cross.
Iconorder	Order in which the tool positions, represented by icons, are switched.

Example

```

<TOOLTYPECONFIGURATION>
  <TOOLTYPE_120>
    <Tooltype value="120" type="uint" />
    <Name value="TM_PAR_SHANK_END_CUTTER" type="QString" />
    <Shortname value="TM_PPTT_SHANK_END_CUTTER" type="QString" />
    <Tooltip value="TM_TTTT_SHANK_END_CUTTER" type="QString" />
    <Icon9 value="to_poly_shank_end_cutter_down.png" type="QString" />
  </TOOLTYPE_120>
</TOOLTYPECONFIGURATION>

```



12.2.6 Configuring the "More data" window

<MOREDATACONFIGURATION> tag

The entries used for configuring the "Additional Data" window are located in the <MOREDATACONFIGURATION> tag. Various data items can be displayed for each tool type in a number of rows and columns. Each data item to be displayed is specified by entering a list parameter:

Tool parameter identifiers (Page 138)

Cutting parameter identifiers (Page 140)

Monitoring parameter identifiers (Page 142)

A brief description of the parameter then appears (ShortText), with the value itself beside it. Configure the list parameters (Page 147).

If another description text is required in the "More Data" window, a text can also be entered in a row or column. In the window, the width of the text is made to conform to that of the brief description and value of a parameter.

Any number of rows and columns can be created. A scrollbar appears when the window reaches a certain size.

Entries	Meaning
TOOLTYPE_XXX	Number of the tool type. See example: Tool type 111 = Ball end mill (tapered die-sinking cutter).
ROWX_COLY	Particular rowX_columnY. If there is no entry after rowX and columnY, this field remains blank. See example: In the "More Data" window, the first row is blank.
Item	The following entries are possible: • Parameter identifier • Text See example: The header "Corner radius" appears in the second row of the window. The text "Radius" and an input field for the value appear in the third row of the window. The fourth row is blank.

Example

```

<MOREDATACONFIGURATION>
  <TOOLTYPE_111>
    <ROW1_COL1>
    </ROW1_COL1>
    <ROW2_COL1>
      <Item value="TM_DGL_CORNER_RADIUS" type="QString" />
    </ROW2_COL1>
    <ROW3_COL1>
      <Item value="GeoCornerRadius" type="QString" />
    </ROW3_COL1>
    <ROW4_COL2>
    </ROW4_COL2>
  </TOOLTYPE_111>
</MOREDATACONFIGURATION>

```

12.2.7 Configure the "New tool - favorites" window

<NEWTOOLFAVORITECONFIGURATION> tag

In the tag <NEWTOOLFAVORITECONFIGURATION> you define the tool types that are most frequently used. The specified tool types are displayed in the "New Tool - Favorites" window.

Entries	Meaning
StaticTooltypes	Numbers of tool types Enter the individual numbers, separated by blanks. The following section describes the assignment of the tool type and number: List of tool types (Page 151)

Example

```
<NEWTOOLFAVORITECONFIGURATION>
  <StaticTooltypes value="120 140 200 220 710 711" type="QString"/>
</NEWTOOLFAVORITECONFIGURATION>
```

12.2.8 Configuring the "New Tool" window

<NEWTOOLCONFIGURATION> tag

You use the "New Tool" dialog to supply a new tool with specific tool data before creating the tool or before loading the tool to a magazine location. These are typically the tool size and/or the tool location type because these parameters can no longer be changed when the tool is loaded. It is also useful to display the tool type and the tool names in this window.

Specify the parameters in the <NEWTOOLCONFIGURATION> tag that are to be displayed in the "New Tool" dialog.

Note

The number of parameters is limited by the size of the window.

Entries	Meaning
ItemX	Entry of the identifier of the tool parameter. "X" specifies the order in which the parameters are displayed in the window. The identifier of the tool parameter can be found in Chapter: Tool parameter identifiers (Page 138)

Example

```
<NEWTOOLCONFIGURATION>
  <!-- The standard field "oversized tool" is removed and replaced by the fields -->
```



```

<!-- "tool location type", "tool size left" and "tool size right" -->
<!-- -->
<Item2 value="ToolPlaceSpec" type="QString"/>
<Item3 value="ToolSizeLeft" type="QString"/>
<Item4 value="ToolSizeRight" type="QString"/>
</NEWTOLCONFIGURATION>

```

12.2.9 Configuring default values for new tools

<NEWTOLDEFAULTVALUECONFIGURATION> tag

In the <NEWTOLDEFAULTVALUECONFIGURATION> tag, you can specify default values for tools that you create via the user interface. These values can be configured depending on the tool type and, if required, also depending on the cutting edge position:

Entries	Meaning
TOOLTYPE_ALL	The parameter setting applies to all tool types.
TOOLTYPE_XXX	The parameter setting applies to a specific tool type. XXX stands for the number of the tool type.
CUTTEDGEPOS_XXX	The parameter setting applies to a specific cutting edge position. XXX stands for the number of the cutting edge position.
Default	Default value of the parameter.

Example

```

<NEWTOLDEFAULTVALUECONFIGURATION>
  <!-- Initialize geometry length 1 for all tool types to 70.0 -->
  <TOOLTYPE_ALL>
    <GeoLengthGeoAx1>
      <Default value="70.0" type="double" />
    </GeoLengthGeoAx1>
  </TOOLTYPE_ALL>
  <!-- Initialize tip angle of the twist drill to 120.0 -->
  <TOOLTYPE_200>
    <NoseAngle>
      <Default value="120.0" type="double" />
    </NoseAngle>
  </TOOLTYPE_200>
  <TOOLTYPE_500>
    <!-- Initialize plate angle of the roughing tool to 82.0 -->
    <PlateAngle>
      <Default value="82.0" type="double" />
    </PlateAngle>
  </TOOLTYPE_500>
</NEWTOLDEFAULTVALUECONFIGURATION>

```

```

<!-- Initialize holder angle of the roughing tool depending on the cutting edge position
-->
<HolderAngle>
  <CUTTEDGEPOS_1>
    <Default value="93.0" type="double" />
  </CUTTEDGEPOS_1>
  <CUTTEDGEPOS_2>
    <Default value="93.0" type="double" />
  </CUTTEDGEPOS_2>
  <CUTTEDGEPOS_3>
    <Default value="93.0" type="double" />
  </CUTTEDGEPOS_3>
  <CUTTEDGEPOS_4>
    <Default value="93.0" type="double" />
  </CUTTEDGEPOS_4>
  <CUTTEDGEPOS_5>
    <Default value="49.0" type="double" />
  </CUTTEDGEPOS_5>
  <CUTTEDGEPOS_6>
    <Default value="49.0" type="double" />
  </CUTTEDGEPOS_6>
  <CUTTEDGEPOS_7>
    <Default value="49.0" type="double" />
  </CUTTEDGEPOS_7>
  <CUTTEDGEPOS_8>
    <Default value="49.0" type="double" />
  </CUTTEDGEPOS_8>
</HolderAngle>
</TOOLTYPE_500>
<!-- Initialize coolant 2 of the plunge cutter to switched on -->
<TOOLTYPE_520>
  <Coolant2>
    <Default value="true" type="bool" />
  </Coolant2>
</TOOLTYPE_520>
</NEWTOLDEFAULTVALUECONFIGURATION>

```

Default values when creating new tools

System variables		Default value
Tool size		
Size on left	\$TC_TP3	1
Size on right	\$TC_TP4	1
Size above	\$TC_TP5	1
Size below	\$TC_TP6	1

System variables		Default value
Magazine location type of the tool		
Created outside of the magazine	\$TC_TP7	1
Created in a magazine location	\$TC_TP7	Location type of the magazine location
Tool status		
released	\$TC_TP8	2

The following combinations between the cutting-edge position and the cutting direction result dependent on the tool type. When creating the new tool, the operator selects a combination in the "New Tool - Favorites" window.

Tool type \$TC_DP1	Default values	
	Cutting edge \$TC_DP2	Cutting direction \$TC_DP11
Type 500 roughing tool	1	4
Type 510 finishing tool	2	3
Type 560 rotary drill	3	3
	4	4
	5	1
	6	3
	7	1
	8	3
Type 520 plunge cutter	1	2
Type 530 cut-off tool	1	4
Grinding tool types	4	4
	3	1
	4	1
	2	3
	3	3
	2	2
Type 540 threading tool	5	2
	5	1
	8	3
	8	4
	6	3
	6	4
	7	2
	7	1

If the ShopMill/ShopTurn option is set, the **direction of spindle rotation** is preassigned:

Tool type \$TC_DP1	Default value \$TC_DP25	Direction of spindle rotation
Measuring tools		
Type 580/585/710/712/713/725	Bit 8 = 0 Bit 9 = 0	Spindle stop
Turning tools dependent on SD54215 TM_FUNCTION_MASK_SET bit 1		
SD54215 bit 1 = 0	Bit 8 = 1 Bit 9 = 0	Spindle right
SD54215 bit 1 = 1	Bit 8 = 0 Bit 9 = 1	Spindle left
Special tools		
Type 730/731/732	Bit 8 = 0 Bit 9 = 0	Spindle stop
All other tool types		
-	Bit 8 = 1 Bit 9 = 0	Spindle right

Tool type \$TC_DP1	Default value \$TC_DP25 for switching direction
Type 712 mono probe	Bit 17 = 1 Bit 20 = 1
Type 713 L probe	Bit 17 = 1 Bit 21 = 1
Type 714 star probe	Bit 16 = 1 Bit 17 = 1 Bit 18 = 1 Bit 19 = 1

Tool type \$TC_DP1	Default value \$TC_DP24 for tip angle
Type 200 twist drill	118
Type 220 centering tool	90

The plate angle rather than the clearance angle is displayed on the screen:
plate angle = 180 - holder angle - clearance angle

Tool type \$TC_DP1	Default values		
	Cutting edge position \$TC_DP2	Holder angle \$TC_DP10	Clearance angle \$TC_DP24
Type 500 roughing tool	1 to 4	95	5
	5 to 8	50	50
Type 510 finishing tool	1 to 4	93	32
	5 to 8	62.5	62.5
Type 560 rotary drill	-	90	2

Tool type \$TC_DP1	Unit of measurement	Default value \$TC_DP8 for plate length
Type 500 roughing tool	metric	11
Type 510 finishing tool	inch	0.4

For all grinding tools, the parameter for **calculating the radius**:
 \$TC_TPG9 = 3 (corresponds to length 1) applies

12.2.10 Configuring the "Details" window

<DETAILSCONFIGURATION> tag

In the tag <DETAILSCONFIGURATION> you can make the settings at the "Detail" window:

Entries	Meaning
ShowToolNumber	The tool number is displayed in the upper right field of the "Detail" window.
UseAxisNameInLength	The tool lengths are displayed with axis identifiers, e.g. length X instead of length 1. If no data is specified, then the setting is taken from the tool lists.
ShowBaseLength	The basis length is displayed in the "Detail" window (AdaptLength1, AdaptLength2 and AdaptLength3). If no data is specified, then the setting is taken from the tool lists.
ShowYAxis	The Y axis is displayed in the "Detail" window. If no data is specified, then the setting is taken from the tool lists.

Example

```
<DETAILSCONFIGURATION>
  <ShowToolNumber value="true" type="bool" />
</DETAILSCONFIGURATION>
```

12.2.11 Assigning a name for magazines and magazine locations

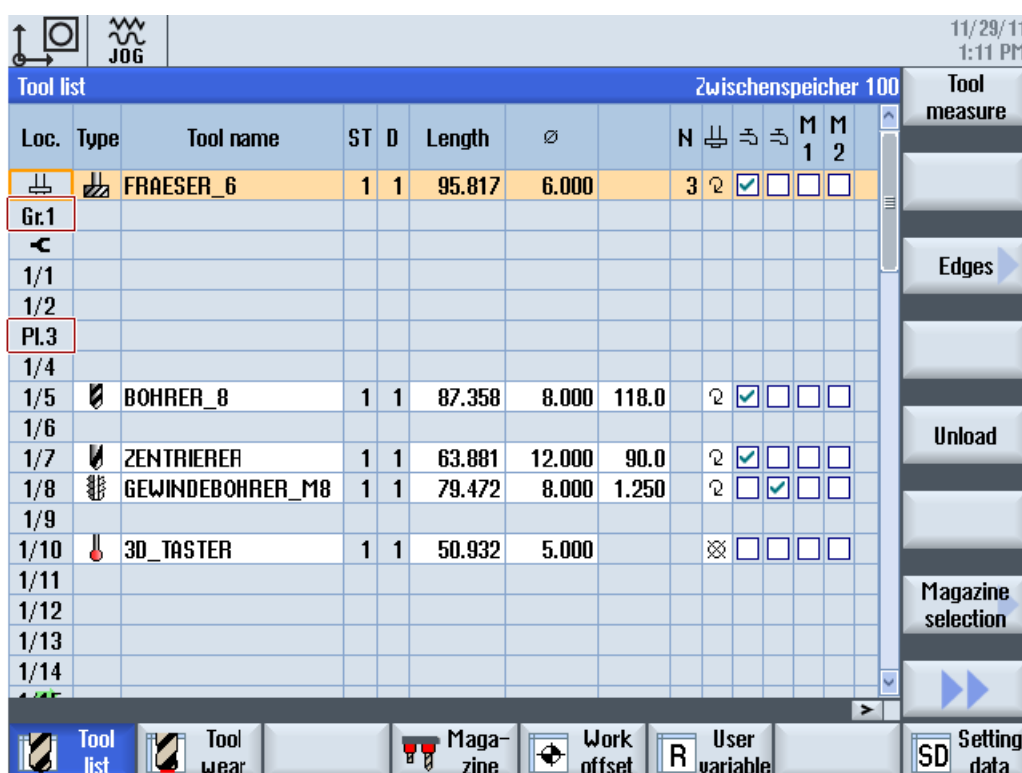
<MAGAZINEPLACENAMECONFIGURATION> tag

The entries to assign dedicated magazine location names are located in the <MAGAZINEPLACENAMECONFIGURATION> tag. Magazine locations can be displayed with texts. Instead of the identifier, e.g. 1/5 for the first magazine, location five, the configured text is then output in the tool lists.

Entries	Meaning
MAGAZINE_XXX	XXX stands for the magazine number.
PLACE_XXX	XXX stands for the location number.
Name	Text identifier of the magazine location.

Example

```
<MAGAZINEPLACENAMECONFIGURATION>
  <MAGAZINE_9998>
    <PLACE_2>
      <Name value="TM_SPECIAL_MAG_PLACE_1" type="QString" />
    </PLACE_2>
  </MAGAZINE_9998>
  <MAGAZINE_1>
    <PLACE_3>
      <Name value="TM_SPECIAL_MAG_PLACE_2" type="QString" />
    </PLACE_3>
  </MAGAZINE_1>
</MAGAZINEPLACENAMECONFIGURATION>
```



The screenshot shows the SINUMERIK tool management interface. At the top right, the date and time are 11/29/11 1:11 PM. The main area is a table titled 'Tool list' with columns: Loc., Type, Tool name, ST, D, Length, Ø, N, and two columns for M (M1, M2). The table contains several tool entries, including 'FRAESER_6', 'BOHRER_8', 'ZENTRIERER', 'GEWINDEBOHRER_M8', and '3D_TASTER'. Some cells are highlighted with red boxes, such as 'Gr.1', 'Pl.3', and the first row. To the right of the table is a vertical toolbar with buttons: 'Tool measure', 'Edges', 'Unload', 'Magazine selection', and 'Setting data'. At the bottom, there is a horizontal toolbar with buttons: 'Tool list', 'Tool wear', 'Magazine', 'Work offset', 'User variable', and 'SD'.

Loc.	Type	Tool name	ST	D	Length	Ø	N	M1	M2
Gr.1		FRAESER_6	1	1	95.817	6.000	3		
1/1									
1/2									
Pl.3									
1/4									
1/5		BOHRER_8	1	1	87.358	8.000	118.0		
1/6									
1/7		ZENTRIERER	1	1	63.881	12.000	90.0		
1/8		GEWINDEBOHRER_M8	1	1	79.472	8.000	1.250		
1/9									
1/10		3D_TASTER	1	1	50.932	5.000			
1/11									
1/12									
1/13									
1/14									

Name for a magazine

Dedicated texts can also be allocated for magazines. The text or the text identifier is entered into the magazine configuration file (ini file) of the NC. The system variable for the magazine name is \$TC_MAP2[magNo]. The magazine name is displayed at the top right in the tool lists.

Example

Magazine 1 should have the name "main magazine".

```
$TC_MAP2[1]="main magazine"
```

If the "main magazine" text is to be output language-dependent, then the language-dependent texts for the text identifier "main magazine" must be created.

See also

Please refer to the following section for the procedure to create language-dependent text:
Creating OEM texts (Page 176)

12.2.12 Assigning a name for magazine location types

<PLACETYPECONFIGURATION> tag

You can find entries to allocate your own location type identifiers in the tag <PLACETYPECONFIGURATION>. Magazines and magazine location types can be displayed with texts in the tool lists. Instead of identifier 2, e.g. for location type 2, the configured text is then output in the tool lists.

Entries	Meaning
PLACETYPEXXX	XXX stands for the number of the location type
Text	Text identifier of the location type
Tooltip	Tooltip text identifier of the location type

Example

```
<PLACETYPECONFIGURATION>
  <PLACETYPE1>
    <Text value="TM_PLACETYPE_1" type="QString" />
    <Tooltip value="TM_TT_PLACETYPE_1" type="QString" />
  </PLACETYPE1>
  <PLACETYPE2>
    <Text value="TM_PLACETYPE_2" type="QString" />
    <Tooltip value="TM_TT_PLACETYPE_2" type="QString" />
  </PLACETYPE2>
  <PLACETYPE3>
    <Text value="TM_PLACETYPE_3" type="QString" />
    <Tooltip value="TM_TT_PLACETYPE_3" type="QString" />
  </PLACETYPE3>
</PLACETYPECONFIGURATION>
```


Magazine										Magazin 101	
Loc.	Type	Tool name	ST	D	D	Z	L	Magazine loc. typ	Tool loc. typ		
1/2					☐			normal			
1/3					☐			normal			
1/4					☐			normal			
1/5		BOHRER_8	1	1	☐	☐	☐	normal	normal		
1/6					☐			normal			
1/7		ZENTRIERER	1	1	☐	☐	☐	normal	normal		
1/8		GEWINDEBOHRER_M8	1	1	☐	☐	☐	normal	normal		
1/9					☐			normal			
1/10		3D_TASTER	1	1	☐	☐	☐	normal	normal		
1/11					☐			normal			
1/12		PLANFRAESER_120	1	1	☐	☐	☐	schwer	schwer		
1/13		BOHRSTANGE_720	1	1	☐	☐	☐	schwer	schwer		
1/14		WINKELKOPF	1	1	☐	☐	☐	hoch	hoch		
1/15		3D_TASTER_32	1	1	☐	☐	☐	hoch	hoch		
1/16					☐			normal			
1/17					☐			normal			
1/18					☐			normal			

See also

Please refer to the following section for the procedure to create language-dependent text:
Creating OEM texts (Page 176)

12.2.13 Assignment of magazines to channels

Requirement

Turning technology has been set: Name of the associated configuration file
slmtturninglistconfig.xml.

Assignment between magazine and channel

With this setting, magazines can be assigned to channels. This makes sense if the machine is configured so that several channels are assigned to one tool area (TOA area). If each magazine is then only used in one channel, here it is possible to specify this assignment between the magazine and channel.

As a consequence of this configuration, the tool icons are output with reference to their position depending on the channel. Every channel can be assigned a position using the channel machine data MD52000 MCS_DISP_COORDINATE_SYSTEM. If a tool is at a magazine

location which is assigned a channel via this configuration, then the tool icon is output depending on this position that has been set.

This setting option is only available for turning. This can be especially helpful if, on a multi-channel machine, a magazine is used before the turning center and another magazine after the turning center. This means that the tools can be entered into the tool list in a position visible for the operator.

<CHANNELMAGAZINEASSIGNMENT> tag

Entries to assign magazines to channels are in the <CHANNELMAGAZINEASSIGNMENT> tag.

Entries	Meaning
CHANNELNO_XXX	XXX stands for the channel number.
MAGAZINES	Number of the magazines that are assigned to the channel.

Example

The following assignment between magazine and channel should apply:

Magazine 1-3	Channel 1
Magazine 4	Channel 2
Magazine 5	Channel 3
Magazine 6	Channel 4

```
<CHANNELMAGAZINEASSIGNMENT>
  <CHANNELNO_1>
    <Magazines value="1 2 3" type="QString" />
  </CHANNELNO_1>
  <CHANNELNO_2>
    <Magazines value="4" type="QString" />
  </CHANNELNO_2>
  <CHANNELNO_3>
    <Magazines value="5" type="QString" />
  </CHANNELNO_3>
  <CHANNELNO_4>
    <Magazines value="6" type="QString" />
  </CHANNELNO_4>
</CHANNELMAGAZINEASSIGNMENT>
```

12.2.14 Coolant and tool-specific functions

Assigning coolant (ShopMill/ShopTurn)

In the tool management of ShopMill/ShopTurn, you can assign coolant and tool-specific functions to each tool. These are activated when the tool is changed. You assign the coolant to the corresponding machine functions (M functions) using the following machine data:

MD52230 \$MCS_M_CODE_ALL_COOLANTS_OFF	M code for all coolants OFF
MD52231 \$MCS_M_CODE_COOLANT_1_ON	M code for coolant 1 ON
MD52232 \$MCS_M_CODE_COOLANT_2_ON	M code for coolant 2 ON
MD52233 \$MCS_M_CODE_COOLANT_1_AND_2_ON	M code for both coolants ON

Tool-specific functions (ShopMill)

Tool-specific functions 1 ... 4 are intended for other M functions that can be activated for a tool, for example, third coolant, speed monitoring, tool breakage, etc.

Up to eight M functions can be programmed for the activation or deactivation of the tool-specific functions. Definition is realized using the following machine data:

MD52281 \$MCS_TOOL_MCODE_FUNC_ON[0...3]	M code for tool-specific function ON
= - 1	

MD52282 \$MCS_TOOL_MCODE_FUNC_OFF[0...3]	M code for tool-specific function OFF
= - 1	

The M functions for the tool-specific functions are executed by the ShopMill cycles at the tool change. If the checkbox is activated, the corresponding M code is generated from MD52281. If the checkbox is not activated, the corresponding M code is generated from MD52282.

If you do not want to automatically execute additional M commands at the tool change, then set the associated machine data to -1 for the M commands of this tool-specific function. This also hides the associated fields in the tool list.

The default value of the machine data is -1 so that no M values are output for the corresponding functions.

Technical procedure

The M functions for the tool-specific functions are generated by the ShopMill cycles in a set (up to four M commands are possible).

The number and also the sequence of the M commands are dependent on the settings of the machine data (MD52281 to MD52282) and the programming.

If M functions less than 100 are used, the number and sequence of the M commands has no significance. They are located in the user PLC in the decoded area (DB21.DBB194 to DB21.DBB206), independent of the number and the sequence when programming.

However, if M functions from 100 and higher are used, then you must decode these yourself in the user PLC. In this case, you must take into consideration the number and sequence (M function 1 to M function 4).

JOG mode

In the JOG mode, no coolant and tool-specific functions are output to the interface. For a tool change in the JOG mode, these functions can be initiated by the operator using the keys on the machine control panel (implementation via the PLC user program).

Example: Machine data settings

MD52281 \$MCS_TOOL_MCODE_FUNC_ON[0]	= 90
MD52281 \$MCS_TOOL_MCODE_FUNC_ON[1]	= 92
MD52281 \$MCS_TOOL_MCODE_FUNC_ON[2]	= 94
MD52281 \$MCS_TOOL_MCODE_FUNC_ON[3]	= 96

MD52282 \$MCS_TOOL_MCODE_FUNC_OFF[0]	= 91
MD52282 \$MCS_TOOL_MCODE_FUNC_OFF[1]	= - 1
MD52282 \$MCS_TOOL_MCODE_FUNC_OFF[2]	= 95
MD52282 \$MCS_TOOL_MCODE_FUNC_OFF[3]	= 97

As a result, fields 1 to 4 of the tool-specific functions are displayed in the tool list.

The following fields are activated:

Tool-specific function 1:	On
Tool-specific function 2:	Off
Tool-specific function 3:	Off
Tool-specific function 4:	Off

The M functions M90, M95 and M97 are generated by the ShopMill cycles at the tool change.

You can change the texts for "Coolant 1/2 on/off" and "Tool-spec. fct. 1, ..." (see Section Identifiers of the standard texts (Page 177)). So that the texts between the tool list and program area machine functions match, please note the following section.

Creating user-specific texts for the "Machine Functions" window.

If you require an additional possibility of switching on or switching off coolant and tool-specific functions, then you can program this in the "Program" operating area using the input window "Machine Functions". You can create the texts and the tooltips for the tool-specific functions which are used in this window.

Define the texts for the M functions in the file "slstepforms_xxx.ts".

If you require the texts in additional languages, then you must create a separate file for each language. The names of the files only differ by the language code "xxx" (Supported languages (Page 584)). Use only the specified language codes.

File "slstepforms_XXX.ts "

Tag	Meaning
source	Text identifier for label: SIStepLabels A maximum of up to four M functions can be defined: T_LAB_USER_MACHINE_FUNC_1 T_LAB_USER_MACHINE_FUNC_2 T_LAB_USER_MACHINE_FUNC_3 T_LAB_USER_MACHINE_FUNC_4 Text identifier for tooltip: SIStepToolTip There are 3 tooltips for every machine function x (1 - 4): T_TT_USER_MACHINE_FUNC_x: Standard tooltip if the machine function was not selected. T_TT_USER_MACHINE_FUNC_x_ON: Tooltip when the machine function is activated T_TT_USER_MACHINE_FUNC_x_OFF: Tooltip for deactivated machine function
translation	Text, which is displayed at the user interface. Label texts: You can only enter texts on one line. The "lines" entry is not applicable. Tooltip: You can enter multi-line texts. A line break is set using the "%n" character.
chars	Text length The maximum text length is 30 characters.
lines	Number of lines The maximum number of lines is three.
remark	Own comment - entry is not displayed at the user interface.

Procedure

1. You can copy the sample file "oem_slstepforms_deu.ts" from the following directory: / **siemens**/sinumerik/hmi/template/lng
2. Save or create the file in the /**oem**/sinumerik/hmi/lng or /**user**/sinumerik/hmi/lng directory.
3. Change the file name to "slstepforms_deu.ts".
If you require texts for additional languages, then a separate file must be created for each language. Save the file with the appropriate language code in the file name.
4. Open the file and define the texts in the <message> and </message> areas.
5. Restart the system.
The file must still be converted into a binary format in order that the texts are displayed during the program runtime. This data is only converted during power up.

Example of a file "slstepforms_deu.ts"

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE TS><TS>
<context>
  <name>SIStepLabels</name>
  <message>
```

```

        <source>T_LAB_USER_MACHINE_FUNC_1</source>
        <translation>tool-specific function 1</translation>
        <chars>20</chars>
    </message>
</context>
<name>SlStepToolTip</name>
<!-- tooltips of user machine function 1-->
<message>
    <source>T_TT_USER_MACHINE_FUNC_1</source>
    <translation>User%nMachine functions 1</translation>
    <chars>25</chars>
    <lines>3</lines>
</message>
</context>
</TS>

```

12.2.15 Reason for the tool change when reactivating

Precondition



Software option

You require the "MC Information System TDI Statistic" option for this function

Function

When reactivating a tool, you have the option of specifying the reason for the tool change, e.g. to use this information for statistical purposes. The "Reason for Tool Change" selection window is opened after pressing the "Reactivate" softkey. In a checkbox, you can select the reason for the tool change. The tool change is entered in a tool parameter to be configured or in a local variable of the SICap service.

Tool change reasons

You can adapt or expand the tool change reasons as required. The operating software provides the following pre-defined texts for the configuration:

Text identifier	Text
TM_DGL_DESIRED_PIECES_ELAPSED	Unit quantity setpoint (desired pieces) elapsed
TM_DGL_TOOL_BREAKAGE	Tool breakage
TM_DGL_CUTTING_EDGE_DAMAGE	Cutting edge damage
TM_DGL_PREMATURE_WEAR	Premature wear
TM_DGL_TYPE_REEQUIPPING	Type reequipping

Text identifier	Text
TM_DGL_TOOL_TRIAL	Tool trial
TM_DGL_PREVENTIVE_CHANGE	Preventive change

<TOOLCHANGEREASON> tag

You configure the function in the <TOOLCHANGEREASON> tag:

Parameter	Designation
Enabled	true - "Reason for tool change" selection window is displayed when re-activating.
	false - default value, selection window is not displayed.
OnlyOneReasonAllowed	It is only permissible to select one tool change reason.
UseCapLocalParameter	The reason for the tool change is entered into a local variable of the SiCap service.
REASONXXX	XXX stands for an internal number. The numbers for the tool change reason must be clear and unique.
Text	Text identifier of the tool change reason.
Parameter	Parameter identifier of the parameter in which it is entered whether the tool change reason was selected.
BitMask	Bit mask to enter the reason for the change in the local variable of the SiCap service. The bit mask is entered as integer value. Bit 0 → 1, bit 1 → 2, bit 2 → 4

Local variable of the SiCap service

The identifier of the local variables is called "/Hmi/ToolEdgeReplacementReason". The variable is the string type.

The table below lists the most important arguments and their significance:

Argument	Meaning
datetime	Data and time of the event
containerNo	Number of the management area
toaNo	Number of the To area
toolNo	Tool number
edgeNo	Cutting edge number
toolIdent	Tool name
toolDuplo	Replacement tool number
ncuName	Name of the NCU
reactivationReason	Reason for tool change
monitoringMode	Monitoring type
remainingBefore	Remaining unit quantity/tool lifetime before reactivation

Example 1

The following example uses the texts for the tool change reason, already prepared in the operating software. In this example, the tool change reason is stored in parameters that still have to be configured, e.g. the OEM tool data 1, bit 1 and following.

```
<TOOLCHANGEREASON>
  <Enabled value="true" type="bool" />
  <REASON1>
    <Text value="TM_DGL_DESIRED_PIECES_ELAPSED" type="QString" />
    <Parameter value="ToolUser_1_Bit1" type="QString" />
  </REASON1>
  <REASON2>
    <Text value="TM_DGL_TOOL_BREAKAGE" type="QString" />
    <Parameter value="ToolUser_1_Bit2" type="QString" />
  </REASON2>
  <REASON3>
    <Text value="TM_DGL_CUTTING_EDGE_DAMAGE" type="QString" />
    <Parameter value="ToolUser_1_Bit3" type="QString" />
  </REASON3>
  <REASON4>
    <Text value="TM_DGL_PREMATURE_WEAR" type="QString" />
    <Parameter value="ToolUser_1_Bit4" type="QString" />
  </REASON4>
  <REASON5>
    <Text value="TM_DGL_TYPE_REEQUIPPING" type="QString" />
    <Parameter value="ToolUser_1_Bit5" type="QString" />
  </REASON5>
  <REASON6>
    <Text value="TM_DGL_TOOL_TRIAL" type="QString" />
    <Parameter value="ToolUser_1_Bit6" type="QString" />
  </REASON6>
  <REASON7>
    <Text value="TM_DGL_PREVENTIVE_CHANGE" type="QString" />
    <Parameter value="ToolUser_1_Bit7" type="QString" />
  </REASON7>
</TOOLCHANGEREASON>
```

Example 2

Also in this example, the prepared texts for the reason for the change are used. It is only permitted to enter one reason for the change as a valid entry. The reason for the tool change is saved in the local variable of the SiCap service.

```
<TOOLCHANGEREASON>
  <Enabled value="true" type="bool" />
```



```

<OnlyOneReasonAllowed value="true" type="bool" />
<UseCapLocalParameter value="true" type="bool" />
<REASON1>
  <Text value="TM_DGL_DESIRE_PIECES_ELAPSED" type="QString" />
  <BitMask value="1" type="int" />
</REASON1>
<REASON2>
  <Text value="TM_DGL_TOOL_BREAKAGE" type="QString" />
  <BitMask value="2" type="int" />
</REASON2>
<REASON3>
  <Text value="TM_DGL_CUTTING_EDGE_DAMAGE" type="QString" />
  <BitMask value="4" type="int" />
</REASON3>
<REASON4>
  <Text value="TM_DGL_PREMATURE_WEAR" type="QString" />
  <BitMask value="8" type="int" />
</REASON4>
<REASON5>
  <Text value="TM_DGL_TYPE_REEQUIPPING" type="QString" />
  <BitMask value="16" type="int" />
</REASON5>
<REASON6>
  <Text value="TM_DGL_TOOL_TRIAL" type="QString" />
  <BitMask value="32" type="int" />
</REASON6>
<REASON7>
  <Text value="TM_DGL_PREVENTIVE_CHANGE" type="QString" />
  <BitMask value="64" type="int" />
</REASON7>
</TOOLCHANGEREASON>

```

See also

For configuring new parameters, see [Configure the list parameters \(Page 147\)](#).

12.2.16 Configure the code carrier connection

Requirement



Software option

You require the "Access MyTool ID" option for this function (Tool Ident Connection).

<IDENTCONNECTIONCONFIGURATION> tag

The entries used for configuring the code carrier connection in the tool list of the user interface are located in the <IDENTCONNECTIONCONFIGURATION> tag. The code carrier connection is implemented via TDI Ident Connection.

The following functions in the tool list are available:

1. Creating tools for the code carrier:

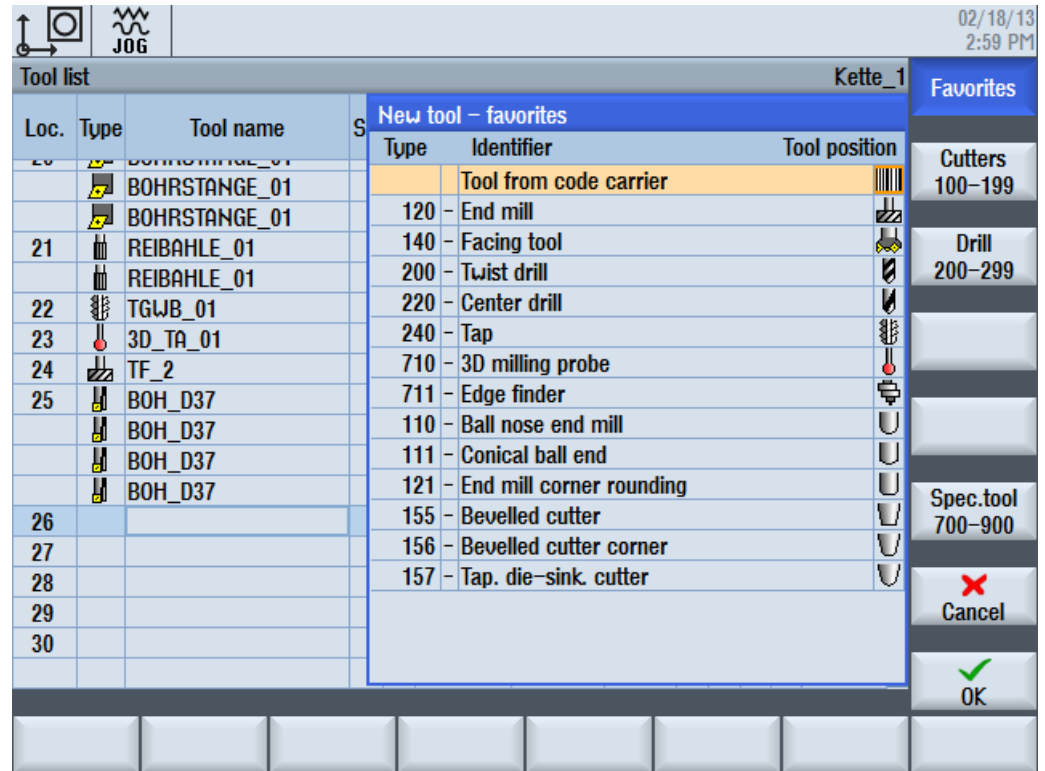


Figure 12-1 New tool for the code carrier

The data for the code carrier tool is read and displayed in the "New Tool" window. Click OK to create the tool in the NC and to initialize it with the code carrier data.

2. Unloading tools on the code carrier:
The tool is unloaded and the data of the tool is written to the code carrier. The tool can also be deleted using a setting (see below).
3. Deleting the tool on the code carrier:
The tool is unloaded and the data of the tool is written to the code carrier. The tool is then deleted from the NC. Using a setting (see below), the "On code carrier" softkey can also be hidden for deleting a tool.

The following settings can be made:

Entries	Meaning
Enabled	true - code carrier connection is offered. false - code carrier connection is not offered.
ShowInternalMessage	true - internal messages for error analysis are also displayed. false - default setting, no internal messages are displayed.
UnloadWithDelete	true - the tool is deleted when unloaded to the code carrier. The "On code carrier" function is not offered for "Delete tool". false - default setting, the tool is only unloaded for "unload to code carrier" and remains in the NC memory.

Entries	Meaning
OnlyInLoadStation	true - default setting, on the one hand, only loaded tools are stored "on code carrier", on the other hand, new tools "from code carrier" can be created only directly in the magazine. false - tools can be created independently of the "from code carrier" magazine slot or deleted "on code carrier".
CreateNewToolWithDialog	true - default setting, before creating the tool for code carrier, the tool data is displayed in the "New Tool" window. false - the tool is created directly in the tool list.

Example

```
<IDENTCONNECTIONCONFIGURATION>
  <Enabled value="true" type="bool" />
  <ShowInternalMessage value="false" type="bool" />
  <UnloadWithDelete value="false" type="bool" />
  <OnlyInLoadStation value="true" type="bool" />
  <CreateNewToolWithDialog value="true" type="bool" />
</IDENTCONNECTIONCONFIGURATION>
```

References

For further information about tool management, please refer to:

Function Manual, SINUMERIK Integrate Access MyTool Ident Connection
(MCIS TDI Ident Connection)

12.3 Creating OEM texts

Overview

This section describes how you create OEM texts in a particular language.

The texts are created in the "sltmlistdialog_XXX.ts" file.

A dedicated file must be created for each language. The file names differ by the language code (Supported languages (Page 584)). For "XXX", set the appropriate language code.

Creating texts

All OEM texts from the previous sections, such as the tooltip, can be made language-dependent by using the OEM text as text identifier under the "source" tag. The translation of the text in the particular language is entered under the "translation" tag.

Entries	Meaning
source	Text identifier See 1st example: Single-line text with new text identifier "MY_NEW_TEXT". See 2nd example: Double-line text with new text identifier "MY_NEW_TEXT_2_LINES".
translation	Text, which is displayed on the user interface. You can enter single and multi-line texts. "%n" is used to indicate a line break. See 1st example: Single-line text "My new text" is displayed. See 2nd example: Double-line text "My new - text" is displayed.
chars	Text length See 1st example: Text length of 30 characters is specified. See 2nd example: Text length of 10 characters per line is specified.
lines	Number of lines See 2nd example: The number of rows is 2.
remark	Own comments - are not displayed.

Procedure

1. You can copy the sample file "oem_sltmlistdialog_eng.ts" from the following directory: /**siemens**/sinumerik/hmi/template/lng
2. Save the file in the /**oem**/sinumerik/hmi/lng or /**user**/sinumerik/hmi/lng directory.
3. Change the file name to "sltmlistdialog_eng.ts".
If you wish to create texts for additional languages, then a separate file must be created for each language. Save the file with the appropriate language code in the file name.
4. Open the file and define the texts in the <message> and </message> areas.
5. Restart the system.
The file must still be converted into a binary format in order that the texts are displayed during the program runtime. This data is only converted during power up.

12.3.1 Identifiers of the standard texts

Identifiers of the standard texts

With the help of the identifiers, you can change the standard texts of OEM parameters, coolants and tool-specific functions. This applies to the column header in the tool lists, the tooltip and, if available, also the designation in the detail displays.

A list of the identifiers is provided below:

Identifiers of OEM tool parameters

"ToolUser_1" to "ToolUser_10"

"ToolAppl_1" to "ToolAppl_10"

Column heading	Tooltip
TM_HL_TOOL_USER_1 to ...10	TM_TT_TOOL_USER_1 to ...10
TM_HL_TOOL_APPL_1 to ...10	TM_TT_TOOL_APPL_1 to ...10

Tool details	Tool details - all parameters
TM_HL_TOOL_USER_1_DETAILS to ...10_DETAILS	TM_HL_TOOL_USER_1_LONG to ...10_LONG
TM_HL_TOOL_APPL_1_DETAILS to ...10_DETAILS	TM_HL_TOOL_APPL_1_LONG to ...10_LONG

Identifiers of the OEM cutting edge parameters

"EdgeUser_1" to "EdgeUser_10"

"EdgeAppl_1" to "EdgeAppl_10"

Column heading	Tooltip
TM_HL_EDGE_USER_1 to ...10	TM_TT_EDGE_USER_1 to ...10
TM_HL_EDGE_APPL_1 to ...10	TM_TT_EDGE_APPL_1 to ...10

Tool details	Tool details - all parameters
TM_HL_EDGE_USER_1_DETAILS to ...10_DETAILS	TM_HL_EDGE_USER_1_LONG to ...10_LONG
TM_HL_EDGE_APPL_1_DETAILS to ...10_DETAILS	TM_HL_EDGE_APPL_1_LONG to ...10_LONG

Identifiers of the OEM monitoring parameters

"EdgeSupUser_1" to "EdgeSupUser_10"

"EdgeSupAppl_1" to "EdgeSupAppl_10"

Column heading	Tooltip
TM_HL_EDGE_SUPUSER_1 to ...10	TM_TT_EDGE_SUPUSER_1 to ...10
TM_HL_EDGE_SUPAPPL_1 to ...10	TM_TT_EDGE_SUPAPPL_1 to ...10

Tool details	Tool details - all parameters
TM_HL_EDGE_SUPUSER_1_DETAILS to ...10_DETAILS	TM_HL_EDGE_SUPUSER_1_LONG to ...10_LONG
TM_HL_EDGE_SUPAPPL_1_DETAILS to ...10_DETAILS	TM_HL_EDGE_SUPAPPL_1_LONG to ...10_LONG

Identifiers of OEM multitool parameters

"MultiToolUser_1" to "MultiToolUser_10"

"MultiToolAppl_1" to "MultiToolAppl_10"

Column heading	Tooltip
TM_HL_MULTITOOl_USER_1 to ...10	TM_TT_MULTITOOl_USER_1 to ...10
TM_HL_MULTITOOl_APPL_1 to ...10	TM_TT_MULTITOOl_APPL_1 to ...10

Tool details	Tool details - all parameters
TM_HL_MULTITOOl_USER_1_DETAILS to ... 10_DETAILS	TM_HL_MULTITOOl_USER_1_LONG to ...10_LONG
TM_HL_MULTITOOl_APPL_1_DETAILS to ... 10_DETAILS	TM_HL_MULTITOOl_APPL_1_LONG to ...10_LONG

Identifiers of the OEM magazine location parameters

"MagPlaceUser_1" to "MagPlaceUser_10"

"MagPlaceAppl_1" to "MagPlaceAppl_10"

Column heading	Tooltip
TM_HL_MAGPLACE_USER_1 to ...10	TM_TT_MAGPLACE_USER_1 to ...10
TM_HL_MAGPLACE_APPL_1 to ...10	TM_TT_MAGPLACE_APPL_1 to ...10

Identifiers of OEM multitool location parameters

"MultiToolPlaceUser_1" to "MultiToolPlaceUser_10"

"MultiToolPlaceAppl_1" to "MultiToolPlaceAppl_10"

Column heading	Tooltip
TM_HL_MTPLACE_USER_1 to ...10	TM_TT_MTPLACE_USER_1 to ...10
TM_HL_MTPLACE_APPL_1 to ...10	TM_TT_MTPLACE_APPL_1 to ...10

Identifiers of the coolant and tool-specific functions

"Coolant1" and "Coolant2"

"MFunction1" to "MFunction4"

Column heading	Tooltip
No column	TM_TT_STATE_COOL_1 and ...2
TM_HL_MFCT1 to ...4	TM_TT_MFCT1 to ...4

12.3.2 Examples of OEM texts

Example 1 with a single-line and double-line text

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE TS><TS>

<!-- ***** -->
<!-- Definition of OEM texts -->
<!-- ***** -->

<context>
  <name>SlTmListForm</name>

  <!-- ***** -->
  <!-- enter your text behind this comment --> -->
  <!-- ***** -->

  <!-- ***** -->
  <!-- 1st example of a single-line text --> -->
  <!-- ***** -->

  <!-- <message>
    <source>MY_NEW_TEXT</source>
    <translation>My new text</translation>
    <chars>30</chars>
  </message>-->

  <!-- ***** -->
  <!-- 2nd example of a double-spaced text --> -->
  <!-- ***** -->

  <!-- <message>
    <source>MY_NEW_TEXT_2_LINES</source>
    <translation>My new%n text</translation>
    <remark>My comment to the text</remark>
    <chars>10</chars>
    <lines>2</lines>
  </message>-->

  <!-- ***** -->
  <!-- enter your text ahead of this comment --> -->
  <!-- ***** -->

</context>
</TS>
```

Example 2

A new parameter is configured in the following example from Section Configure the list parameters (Page 147):

```
<PARAMETERCONFIGURATION>
  <EdgeUser_1_Bit0>
    <Base value="EdgeUser_1" type="QString" />
```



```

        <ItemType value="CheckBox" type="QString" />
        <DisplayMode value="AnyMode" type="QString" />
        <BitMask value="1" type="int" />
        <Width value="17" type="int" />
        <HeadLine value="TM_HL_EDGE_USER_1_Bit0" type="QString" />
        <ToolTip value="TM_TT_EDGE_USER_1_Bit0" type="QString" />
    </EdgeUser_1_Bit0>
</PARAMETERCONFIGURATION>

```

The two texts under HeadLine and ToolTip should be output depending on the language. To do this, the two text identifiers must be included in the sltmlistdialog_xxx.ts file:

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE TS><TS>

<!-- ***** -->
<!-- Definition of OEM texts -->
<!-- ***** -->

<context>
    <name>SlTmListForm</name>

    <!-- ***** -->
    <!-- enter your text behind this comment --> -->
    <!-- ***** -->

    <!-- ***** -->
    <!-- Translate the text for the headline --> -->
    <!-- ***** -->

    <message>
        <source>TM_HL_EDGE_USER_1_Bit0</source>
        <translation>heavy %n tool</translation>
        <chars>10</chars>
        <lines>2</lines>
    </message>

    <!-- ***** -->
    <!-- Translate the text for the tooltip --> -->
    <!-- ***** -->

    <message>
        <source>TM_TT_EDGE_USER_1_Bit0</source>
        <translation>heavy tool</translation>
        <chars>20</chars>
    </message>

    <!-- ***** -->
    <!-- enter your text ahead of this comment --> -->
    <!-- ***** -->

</context>
</TS>

```

12.4 Examples

12.4.1 Example: Configuring an OEM tool list

12.4.1.1 Adapting the configuration file

The following excerpt shows the contents of the configuration file:

- Milling technology: slttmlistconfig.xml
- Turning technology: sltmturninglistconfig.xml

The file in this example is stored in the "user/sinumerik/hmi/cfg" directory.

This example describes a configuration of the OEM tool list.

- OEM tool parameter 1 is displayed in column 6.
- A new "Coolant 3" parameter is inserted in column 7 that displays bit 0 from OEM tool parameter 2.
- OEM cutting edge parameter 1 is displayed in column 8.

Example

"slttmlistconfig.xml" or "sltmturninglistconfig.xml" file:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<CONFIGURATION>
  <!-- **** Identifier for the configuration of the lists **** -->
  <LISTCONFIGURATION>
    <!-- **** OEM tool list **** -->
    <SlTmTooloemForm>
      <!-- **** Enabling the OEM tool list **** -->
      <Enabled value="true" type="bool" />
      <!-- **** Columns 1 to 5 have already been assigned default values. **** -->
      <!-- **** Column 6 **** -->
      <COLUMN6>
        <!-- **** For all tool types **** -->
        <TOOLTYPE_ALL>
          <!-- **** Display OEM tool parameter 1 **** -->
          <Item value="ToolUser_1" type="QString" />
        </TOOLTYPE_ALL>
      </COLUMN6>
      <!-- **** Column 7 **** -->
      <COLUMN7>
        <!-- **** For all tool types **** -->
        <TOOLTYPE_ALL>
```

```

        <!-- **** Priority tool, own parameter *** -->
        <!-- **** Display OEM tool parameter 2 bit 0 *** -->
        <Item value="PriorTool" type="QString" />
    </TOOLTYPE_ALL>
</COLUMN7>

<!-- **** Column 8 *** -->
<COLUMN8>
    <!-- **** For all tool types *** -->
    <TOOLTYPE_ALL>
        <!-- **** Display OEM cutting edge parameter 1 *** -->
        <Item value="EdgeUser_1" type="QString" />
    </TOOLTYPE_ALL>
</COLUMN8>
</SlTmTooloemForm>
</LISTCONFIGURATION>

<!-- **** Identifier for the configuration of own parameter *** -->
<PARAMETERCONFIGURATION>
    <!-- **** Change OEM tool parameter 1 *** -->
    <ToolUser_1>
        <!-- **** Column width *** -->
        <Width value="36" type="int" />
        <!-- **** Type change to unsigned integer *** -->
        <DisplayMode value="UnsignedIntegerMode" type="QString" />
    </ToolUser_1>
    <!-- **** Identifier of own parameter *** -->
    <!-- **** Priority tool *** -->
    <PriorTool>
        <!-- ** New parameter is based on OEM tool parameter 2**-->
        <Base value="ToolUser_2" type="QString" />
        <!-- **** Bit 0 is evaluated *** -->
        <BitMask value="1" type="int" />
        <!-- **** Text identifier of the headline *** -->
        <HeadLine value="TMO_HL_PRIOR_TOOL" type="QString" />
        <!-- **** Text identifier of the tooltip *** -->
        <ToolTip value="TMO_TT_PRIOR_TOOL" type="QString" />
        <!-- **** Column width *** -->
        <!-- **** Recommended widths 640x480 - 19 pixels *** -->
        <!-- **** 800x600 - 21, 1024x768 - 17 *** -->
        <Width value="19" type="int" />
        <!-- **** Display item of the parameter *** -->
        <ItemType value="CheckBox" type="QString" />
    </PriorTool>
</PARAMETERCONFIGURATION>
</CONFIGURATION>

```

12.4.1.2 Adaptation of the customer text file

The following excerpt shows the contents of the "sltmlistdialog_eng.ts" customer text file.

The file in this example is stored in the "user/sinumerik/hmi/lng" directory.

Example

"sltmlistdialog_eng.ts" file:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE TS><TS>

<!-- ***** -->
<!-- Definition of OEM texts -->
<!-- ***** -->
<context>
    <name>SlTmListForm</name>

<!-- ***** -->
<!-- Enter your text behind this comment -->
<!-- ***** -->

    <!-- **** OEM tool parameter 1 headline *** -->
    <!-- **** Weight *** -->
    <message>
        <!-- **** Text identifier *** -->
        <source>TM_HL_TOOL_USER_1</source>
        <!-- **** Text that is output *** -->
        <translation>%nweight</translation>
        <!-- **** Comment only for the translator *** -->
        <remark>priority tool</remark>
        <!-- **** Maximum line length in characters *** -->
        <!-- **** only for the translator *** -->
        <chars>7</chars>
        <!-- **** Number of possible lines *** -->
        <!-- **** only for the translator *** -->
        <lines>2</lines>
    </message>
    <!-- **** OEM tool parameter 1 tooltip *** -->
    <!-- **** Weight *** -->
    <message>
        <source>TM_TT_TOOL_USER_1</source>
        <translation>weight of the tool</translation>
        <chars>35</chars>
        <lines>2</lines>
```

```

</message>
<!-- **** Headline for own parameter *** -->
<!-- **** Priority tool *** -->
<message>
  <source>TMO_HL_PRIOR_TOOL</source>
  <translation>V</translation>
  <remark>priority tool</remark>
  <chars>1</chars>
  <lines>2</lines>
</message>
<!-- **** Tooltip for own parameter *** -->
<!-- **** Priority tool *** -->
<message>
  <source>TMO_TT_PRIOR_TOOL</source>
  <translation>priority tool, %nuse tool preferably< /translation>
  <chars>35</chars>
  <lines>3</lines>
</message>
<!-- **** OEM cutting edge parameter 1 headline *** -->
<!-- **** Maximum tool width *** -->
<message>
  <source>TM_HL_EDGE_USER_1</source>
  <translation>maximum%nwidth</translation>
  <chars>8</chars>
  <lines>2</lines>
</message>
<!-- **** OEM cutting edge parameter 1 tooltip *** -->
<!-- **** Maximum tool width *** -->
<message>
  <source>TM_TT_EDGE_USER_1</source>
  <translation>maximum width%n of the tool%noverall</translation>
  <chars>35</chars>
  <lines>2</lines>
</message>

<!-- ***** -->
<!-- Enter your text ahead of this comment -->
<!-- ***** -->

</context>
</TS>

```

12.4.2 Example: Configuring magazine location types with names

12.4.2.1 Adapting the configuration file

The following excerpt shows the contents of the configuration file:

- Milling technology: sltmlistconfig.xml
- Turning technology: sltmturninglistconfig.xml

The file in this example is stored in the "user/sinumerik/hmi/cfg" directory.

This example describes a configuration of the location types in the magazine list.

In this example, the magazine location type and the tool location type are to be displayed as a toggle field with plain texts.

The assignment of the parameters to the columns of the list is made in the "LISTCONFIGURATION" area. The magazine location type is displayed in column 10. The tool location type is displayed in column 11.

The texts to the location types are assigned in the "PLACETYPECONFIGURATION" area. The number of the location type is the number at the end of the "PLACETYPE" identifier. A text identifier for a toggle box entry and a text identifier for a tooltip can be stored for each location type.

Example

"sltmlistconfig.xml" or "sltmturninglistconfig.xml" file:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<CONFIGURATION>
  <!-- **** Configuration of the lists **** -->
  <LISTCONFIGURATION>
    <!-- **** Magazine list **** -->
    <SlTmToolmagazinForm>
      <!-- **** Column 10 **** -->
      <COLUMN10>
        <!-- **** For all tool types **** -->
        <TOOLTYPE_ALL>
          <!-- **** Display magazine location type **** -->
          <Item value="MagPlaceTypeIdent" type="QString" />
        </TOOLTYPE_ALL>
      </COLUMN10>
      <!-- **** Column 11 **** -->
      <COLUMN11>
        <!-- **** For all tool types **** -->
        <TOOLTYPE_ALL>
          <!-- **** Display tool location type **** -->
          <Item value="ToolPlaceSpecIdent" type="QString" />
        </TOOLTYPE_ALL>
      </COLUMN11>
    </SlTmToolmagazinForm>
  </LISTCONFIGURATION>
</CONFIGURATION>
```

```

<!-- **** For all multitools *** -->
<TOOLTYPE_9997>
  <!-- **** Display for all multitool location types *** -->
  <Item value="MultiToolPlaceSpecIdent" type="QString" />
</TOOLTYPE_9997>
</COLUMN11>
</SlTmToolmagazinForm>
</LISTCONFIGURATION>
<!-- **** Configuration of the magazine location types *** -->
<PLACETYPECONFIGURATION>
  <!-- **** Magazine location type 1 *** -->
  <PLACETYPE1>
    <!-- **** Text identifier of the toggle box entry *** -->
    <Text value="TMMP_ITEM_NORMAL_TOOL" type="QString" />
    <!-- **** Text identifier of the tooltip *** -->
    <Tooltip value="TMMP_TT_NORMAL_TOOL" type="QString" />
  </PLACETYPE1>
  <!-- **** Magazine location type 2 *** -->
  <PLACETYPE2>
    <Text value="TMMP_ITEM_HEAVY_TOOL" type="QString" />
    <Tooltip value="TMMP_TT_HEAVY_TOOL" type="QString" />
  </PLACETYPE2>
  <!-- **** Magazine location type 3 *** -->
  <PLACETYPE3>
    <Text value="TMMP_ITEM_BIG_TOOL" type="QString" />
    <Tooltip value="TMMP_TT_BIG_TOOL" type="QString" />
  </PLACETYPE3>
  <!-- **** Magazine location type 4 *** -->
  <PLACETYPE4>
    <Text value="TMMP_ITEM_SENSITIVE_TOOL" type="QString" />
    <Tooltip value="TMMP_TT_SENSITIVE_TOOL" type="QString" />
  </PLACETYPE4>
</PLACETYPECONFIGURATION>
</CONFIGURATION>

```

12.4.2.2 Adaptation of the customer text file

The following excerpt shows the contents of the "sltmlistdialog_eng.ts" customer text file. The file in this example is stored in the "user/sinumerik/hmi/lng" directory.

Example

"sltmlistdialog_eng.ts" file:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE TS><TS>

<!-- ***** -->
<!-- Definition of OEM texts -->
<!-- ***** -->
<context>
    <name>SlTmListForm</name>

<!-- ***** -->
<!-- Enter your text behind this comment -->
<!-- ***** -->
    <!-- **** Magazine location type 1 item **** -->
    <!-- **** Normal tool **** -->
    <message>
        <!-- **** Text identifier **** -->
        <source>TMMP_ITEM_NORMAL_TOOL</source>
        <!-- **** Text that is output **** -->
        <translation>normal</translation>
        <!-- **** Maximum line length in characters **** -->
        <!-- **** only for the translator **** -->
        <chars>9</chars>
    </message>
    <!-- **** Magazine location type 1 tooltip **** -->
    <!-- **** Normal tool **** -->
    <message>
        <source>TMMP_TT_NORMAL_TOOL</source>
        <translation>normal tool,%nno restrictions</translation>
        <chars>35</chars>
        <lines>2</lines>
    </message>
    <!-- **** Magazine location type 2 item **** -->
    <!-- **** Heavy tool **** -->
    <message>
        <source>TMMP_ITEM_HEAVY_TOOL</source>
        <translation>heavy</translation>
        <chars>9</chars>
    </message>
    <!-- **** Magazine location type 2 tooltip **** -->
    <!-- **** Heavy tool **** -->
    <message>
```



```

        <source>TMMP_TT_HEAVY_TOOL</source>
        <translation>heavy tool</translation>
        <chars>35</chars>
        <lines>2</lines>
    </message>
    <!-- **** Magazine location type 3 item *** -->
    <!-- **** Big tool *** -->
    <message>
        <source>TMMP_TT_BIG_TOOL</source>
        <translation>big</translation>
        <chars>9</chars>
    </message>
    <!-- **** Magazine location type 3 tooltip *** -->
    <!-- **** Big tool *** -->
    <message>
        <source>TMMP_TT_BIG_TOOL</source>
        <translation>big tool</translation>
        <chars>35</chars>
        <lines>2</lines>
    </message>
    <!-- **** Magazine location type 4 item *** -->
    <!-- **** Sensitive tool *** -->
    <message>
        <source>TMMP_ITEM_SENSITIVE_TOOL</source>
        <translation>sensitive</translation>
        <chars>9</chars>
    </message>
    <!-- **** Magazine location type 4 tooltip *** -->
    <!-- **** Sensitive tool *** -->
    <message>
        <source>TMMP_TT_SENSITIVE_TOOL</source>
        <translation>Notice: Tool can%be destroyed easily</translation>
        <chars>35</chars>
        <lines>2</lines>
    </message>

    <!-- ***** -->
    <!-- Enter your text ahead of this comment -->
    <!-- ***** -->

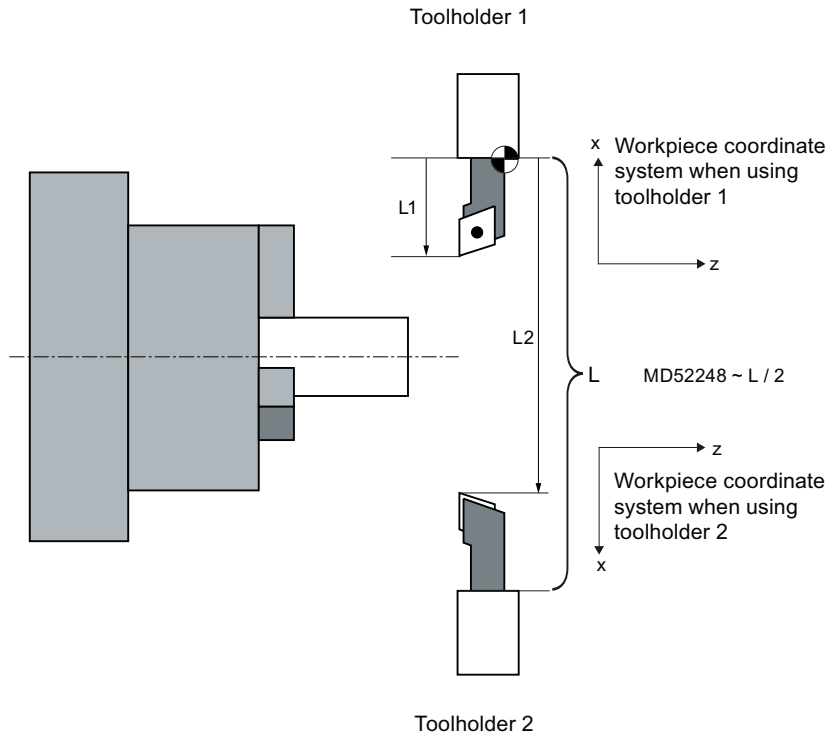
</context>
</TS>

```

12.5 Working with two toolholders

It is possible to work on a single turning machine with two toolholders, both of which are mounted on an X axis. The toolholders may be revolvers, multifix, or a combination of both. As both tool holders are mounted on the same axis it is only possible to machine with one tool. The workpiece is located between the two tool adapters.

All tools have the same reference point (on toolholder 1) irrespective of which toolholder they are on. Consequently, the tool lengths of the second toolholder are always longer than the tool lengths of the first toolholder. The main machining is performed in the negative X axis direction.



Toolholder definition

The MD52248 REV_2_BORDER_TOOL_LENGTH defines a tool length limit X so that the tools can be assigned to the two toolholders automatically:

Tool length $X < \text{MD52248}$: Tool belongs to the 1st toolholder

Tool length $X \geq \text{MD52248}$: Tool belongs to the 2nd toolholder

For the mirror X to become active for the activation of the tools of the second toolholder, the following T/TCA replacement must be set:

MD10717 \$MN_T_NO_FCT_CYCLE_NAME="F_T_REV2"

MD15710 \$MN_TCA_CYCLE_NAME="F_T_REV2"**Note**

If there are a large number of old G code programs that were programmed without mirroring of the X axis, you should consider setting mirroring for G code programs.

If the required mirroring is also to be automatically activated when programming the TRANSMIT command, the TRANSMIT command must be replaced by a subprogram (e.g. in the manufacturer directory):

```
MD10712 $MN_NC_USER_CODE_CONF_NAME_TAB[0]="TRANSMIT"
```

```
MD10712 $MN_NC_USER_CODE_CONF_NAME_TAB[1]="TRANSM"
```

TRANSMIT.SPF subprogram:

```
PROC TRANSMIT(INT_NR)
```

```
F_T_REV2
```

```
TRANSM(_NR)
```

```
F_T_REV2
```

```
RET
```

Note

In the status display, TRANSM is now displayed instead of TRANSMIT.

Additional parameter assignments

For the used mirrorings of the X and Y axes and the offsets of the C axis, the following machine data has to be configured:

MD24006 \$MC_CHSFRAME_RESET_MASK	Active system frames after RESET
Bit 4 = 1	System frame for workpiece reference remains active after RESET

MD24007 \$MC_CHSFRAME_RESET_CLEAR_MASK	Clear system frames after RESET
Bit 4 = 0	Do not clear system frame workpiece reference after RESET

MD28082 \$MC_MM_SYSTEM_FRAME_MASK	Setting-up system frames (SRAM)
Bit 4 = 1	Workpiece reference (\$P_WPFRAME)

MD28083 \$MC_MM_SYSTEM_DATAFRAME_MASK	Setting-up data management, system frames (SRAM)
Bit 4 = 1	Workpiece reference (\$P_WPFRAME)

Due to the coordinate system being mirrored, the following machine/setting data must be set:

MD21202 \$MC_LIFTFAST_WITH_MIRROR	Lift fast with mirror
= 1	For the determination of the lifting direction, it is also mirrored for active mirroring. The mirroring of the lifting direction only refers to the direction components perpendicular to the tool direction.

SD42910 \$SC_MIRROR_TOOL_LENGTH	Sign change, tool wear when mirroring
= 1	Inverting the sign mirrors the wear values of the tool length components whose associated axes are mirrored.

General conditions

- The taper turning is not enabled for "working with two toolholders".
- The B-axis function (kinematics) is also not enabled for "working with two toolholders".

12.6 Editor for the tool and magazine management

12.6.1 Overview

The tool management offers the possibility to define all real tool magazines and internal magazines with the appropriate properties from the operating software and link them channel-specific with the tool unit. You can configure the relationships between tool magazine, gripper and spindle via a tree view.

General sequence

The following steps are required to create a tool management:

- Create tool unit
- Define buffer locations
- Define loading location
- Create magazine configuration
- Link assignments

Note

When you create a new tool unit, a buffer location with a spindle location, a loading magazine with a loading location and a real magazine with five locations are created by default.

Meaning of the symbols

Symbol	Description
	Folder for the actively used tool units
	Folder with access to templates
	Tool spindle / tool holder
	Tool gripper 1
	Tool gripper 2
	Shows the changes in the the "Active data" folder that has not been activated yet

Meaning of the tree view

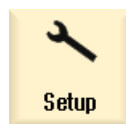
The following folders are available in the tree view:

- Active data: Configure and activate the magazine management.
- Examples: Copy templates into the "Active data" folder.
- Network/USB: Copy the new magazine managements into the "Active data" folder or use existing ones for other systems.

12.6.2 Calling the tool management

You open the tool management window as follows:

Procedure



1. Select the "Start-up" operating area.



2. Press the "NC" and "Tool management" softkeys.



The "Tool management" window opens.

12.6.3 Editing the tool management

For the tree view of the tool management, functions are available with which you can parameterize magazine data items and link them with each other.

Note

If you position the cursor on an element, only those functions possible for this element are active. All other functions are inactive which is indicated with grayed-out softkeys.

Procedure



The "Tool management" window is open.

Adding a new tool unit



1. Position the cursor on the "Active data" folder element.

2. Press the "New element" softkey.

A folder element with the standard designation <tool unit> is inserted and the "TO unit" input window opens.

3. You can make changes in the input window of the folder element.



4. Press the "OK" softkey to insert the folder element with the settings.

Inserting a new magazine



1. Position the cursor on the "Active data" folder element.

2. Navigate with the <Cursor right> and <Cursor down> keys to the folder element of the tool unit.



3. Press the "New element" softkey.

A folder element with the standard designation <magazine> is inserted and the "Magazine" input window opens.

4. You can make changes in the input window of the folder element.



5. Press the "OK" softkey to insert the folder element with the settings.

Editing an element

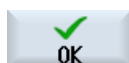
1. Position the cursor on the "Active data" folder element.



2. Navigate with the <Cursor right> and <Cursor down> keys to the element that you want to edit.



3. Press the <Cursor right> or <INPUT> key.
The input window for the element settings opens.



4. Enter the desired changes and press the "OK" softkey to confirm the entries.

Note

You define the number of buffer and magazine locations in the input window of the higher-level <Magazine> folder element with the "Locations" parameter.

Activating Data



1. Press the "Activate data" softkey.
All changed parameterizations are transferred to the NC and activated.
All of the error messages are displayed again, and you have the option of specifically resolving these by modifying the data.

Note

Once you have made changes to the tool management, the data is marked with "***". This data is not yet in the NC and so does not have any effect.

If you want to create a tool management that is similar to an already existing element, then you can save time if you copy the existing elements and insert them at a different position. You have the option, for example of also copying inactive subtrees into the active tree.

Copying/cutting and pasting elements

1. Position the cursor on the desired element.
2. Press the "Copy" softkey.



- OR -



Press the "Cut" softkey.



3. Position the cursor at the required position and press the "Paste" softkey. If the element exists already at the desired location, you receive further insertion options.



4. Press the "OK" softkey if you wish to attach the element to the tree view as the next element.
The element is inserted.

Note:

You have the option of changing the name of the element in the associated input window.

Deleting elements



1. Place the cursor on the desired element and press the "Delete element" softkey.



2. Press the "OK" softkey to confirm the deletion.
The selected object is removed.

12.6.4 Parameterizing the tool management

12.6.4.1 Create tool unit

A tool unit is the grouping of tools and magazines to form a unit. A tool unit can be assigned one or more channels. The tools and magazines must be unique within a tool unit. For a tool unit, specify the search parameters and the channel in the "TO unit" window.

Directory values

You are currently located under the "Active data" → <tool unit> directory in the "Commissioning" → "NC" → "Tool management" operating area.

You can set the following parameters in the "TO unit" window from the tool unit directory:

Parameters	Description
Name	Designation that is displayed in the tree structure.
Channel number	The channel to which the tool unit is assigned.

Parameters	Description
Tool search	Search strategy: • Without search strategy • Active tool • Shortest way
Empty location search	Search strategy: • Without search strategy • First location forwards • Current location forwards • Current location backwards • Last location backwards • Symmetric current location • 1:1 exchange

12.6.4.2 Define buffer locations

Buffer locations are spindles, grippers, loaders and transfer locations. All buffer locations are managed in an internal buffer magazine with the number 9998. The number of buffer locations is defined from the "Magazine" window. The buffer location type is defined from the "Location" window.

Directory values

You are currently located under the "Active data" → <tool unit> → <buffer> directory in the "Commissioning" → "NC" → "Tool management" operating area.

You can define the following parameters in the "Magazine" window from the buffer directory:

Parameters	Description
Name	Designation of the buffer magazine The designation is displayed in the tree structure.
Magazine number	9998 (cannot be changed)
Locations	Number of the buffer locations

Element values

You can change the buffer location type with the parameters in the "Location" window:

Parameters	Description
(Selection menu)	Select a buffer type • Tool spindle / tool holder • Loader • Gripper • Transfer location
Index	The index counts the locations of a type.

12.6.4.3 Define loading location

Loading location are loading points or loading stations. All loading locations are stored in an internal loading magazine with the number 9999. The number of loading locations is defined from the "Magazine" window. The loading location type is defined from the "Location" window.

Directory values

You are currently located under the "Active data" → <tool unit> → <loading magazine> directory in the "Commissioning" → "NC" → "Tool management" operating area.

You can define the following parameters in the "Magazine" window from the loading magazine directory:

Parameters	Description
Name	Designation of the loading magazine The designation is displayed in the tree structure.
Magazine number	9999 (cannot be changed)
Locations	Number of loading locations

Element values

You can change the loading location type with the parameters in the "Location" window:

Parameters	Description
(Selection menu)	Select a loading location type: • Load location Load locations are areas on the machine at which the direct loading in the magazine is possible. The tool can be inserted directly into the magazine at this location. The magazine location to be loaded is moved to the loading point. For example, chain magazines have loading points. • Loading station A loading station is a location outside the magazine onto which the tool to be loaded is placed. The tool is then transported from that location to the magazine via a transport mechanism. Loading stations are generally used for box or chain magazines.
Index	The index counts the locations of a type.

Note

Loading point 1 is always set automatically as loading/unloading point for spindle 1 (manual loading point).

12.6.4.4 Create magazine configuration

You can define several magazines in a tool unit. You define the magazine configuration via the "Magazine" window. The magazine location type is defined from the "Location" window.

Directory values

You are currently located under the "Active data" → <tool unit> → <magazine> directory in the "Commissioning" → "NC" → "Tool management" operating area.

You can define the following parameters in the "Magazine" window from the magazine directory:

Parameters	Description
Magazine type	Selection menu: • Chain magazine • Box magazine • Revolver
Name	Designation of the magazine. The designation is displayed in the tree structure.
Magazine number	Specifies the number of the magazine
Locations	Number of magazine locations
Tool search	Search strategy: • Without search strategy • Take from the TOA • Active tool • Shortest way
Empty location search	Search strategy: • Without search strategy • Take from the TOA • First location forwards • Current location forwards • Current location backwards • Last location backwards • Symmetric current location • 1:1 exchange
Location coding	• Variable • Fixed-location coded
Location type	Identification for the magazine location type (Changes the value globally for all locations in the magazine)
Consider adjacent location	• Yes • No (Changes the value globally for all locations in the magazine)

Note

If one of the Location type or Consider adjacent location parameters is changed individually and the value is not identical for all locations, the "Variable" definition appears in the display.

Element values

You can change the magazine location type with the parameters in the "Location" window:

Parameters	Description
Location type	Identification for the magazine location type
Consider adjacent location	• Yes • No

12.6.4.5 Link assignments

You can establish the mechanical relationship between the individual components of the tool unit in the "Assignment" window.

Element values

You are currently located under the "Active data" → <tool unit> directory in the "Commissioning" → "NC" → "Tool management" operating area.

You can link the following parameters with each other from the selection list via the Assignments element:

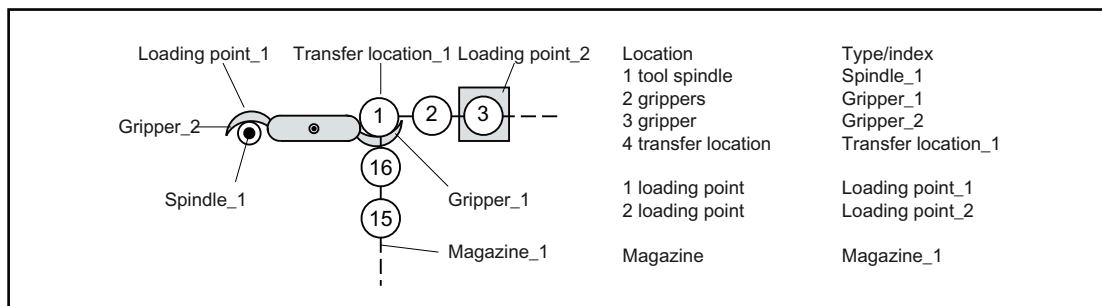
- Spindle/gripper/loader for magazine
- Gripper/loader for spindle
- Loading point for magazine

Note

The separation as number of locations from the magazine zero point of the affected magazine to the internal magazine location must be defined for each magazine assignment.

Example

If, for example, tools on a spindle must be replaced from a magazine, the following assignment can result:



Parameters	Assignment
Spindle/gripper/loader for magazine	• Tool spindle location 1 for magazine (separation 0) • Gripper location 2 for magazine (separation 0) • Gripper location 3 for magazine (separation 0) • Transfer location 4 for magazine (separation 0)
Gripper/loader for spindle	• Tool spindle location 1 for gripper 2 • Tool spindle location 1 for gripper 3
Loading point for magazine	• Loading point location 1 for magazine (separation 0) • Loading point location 2 for magazine (separation 3)

Configuring alarms

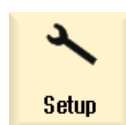
13.1 Creating alarm and message texts via the user interface

From the user interface, you can create and edit your alarm and message texts from the part program. In turn, the alarm and message texts are saved corresponding to their number in various text files.

Storage

The text files are created in the directory:
/oem/sinumerik/hmi/lng

Procedure



1. Select the "Start-up" operating area.

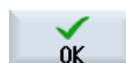


2. Press the "HMI" softkey.



3. Press the "Alarm texts" softkey.

The "Select File" window opens and offers the user-specific text files.



4. Select the desired file and press the "OK" softkey.

The window, e.g. "Edit Alarm Texts (User PLC Alarm Texts) German" opens.

5. Enter the desired alarm number in the "Number" column.

Enter the desired alarm text in the "Alarm text" column.

Select the desired font color in the "Color" column.

In the column "PopUp" with "yes" or "no", select whether the alarm text should be displayed in a window that requires acknowledgment - or not.

- AND / OR -



Press the "Insert line" softkey.

A new line is inserted above your cursor.

- OR



Press the "Delete line" softkey to delete the selected line.



6. Press the "Search" softkey.
The "Search" window opens.
Enter the desired alarm text or the alarm number in the "Text" field.
Activate the "Case sensitive" checkbox if a distinction must be made between upper- and lower-case letters when entering text.

- OR



- Press the "Search and replace" softkey.
The "Search and Replace" window opens.
Enter the search term in the "Text" field. Enter the replacing term in the "Replace with" field.



7. Position the cursor in the "Direction" field. Using the <SELECT> key, select the search direction (forwards, backwards).



8. Press the "OK" softkey to start search or search and replace.



Press the "Cancel" softkey if you wish to interrupt the action.



9. If you have entered the alarms with the help texts, press the "OK" softkey.
You will receive the message "The alarm texts have been saved and converted."

Restart SINUMERIK Operate

You must restart SINUMERIK Operate in order that the "color" and "PopUp" attributes take effect. Modified alarm texts, however, take effect immediately.

Further search options



The cursor jumps to the first entry of the selected alarm text file.



The cursor jumps to the last entry of the selected alarm text file.

Creating search directories

In order that a search can be made for alarm files in subdirectories other than the existing /Ing directory, you have the following option:

1. Copy the "systemconfiguration.ini" configuration file from the following directory:
/siemens/sinumerik/hmi/cfg
2. Store the file in the following directory:
/oem/sinumerik/hmi/cfg
3. Create any subdirectory under /hmi, e.g. /oem_dir.

4. Create the following section in the configuration file "systemconfiguration.ini":
`[oem_dirs]`
`OEM_3=oem_dir`
5. Restart SINUMERIK Operate.
When searching, the subdirectory is now also taking into account.

Foreign language alarm and message texts

If you create foreign language alarm and message texts, change over into the desired language. All text files are automatically allocated the language code corresponding to the user interface language that has been selected. For instance, if you create alarm texts in the English user interface, then the text file is assigned the "eng" language code.

See also

Range of alarms (Page 222)

13.2 Configuring alarm and message texts via alarm text files

You have the option of creating customized alarm and message texts using alarm text files and adapting these.

The alarm texts are set up in standard format (".ts" format) by SINUMERIK Operate. The ".ts" format is XML-based.

You can create and edit these files not only with SINUMERIK Operate, but also externally on a PC.

Note

If you wish to edit alarm text files on a PC, use an editor supporting UTF-8 coding.

Creating and editing alarm texts

- Creating your own alarm texts
- Creating texts for indexed alarm parameters
- Creating part program message texts
- Changing alarm attributes
- Replacing standard alarm texts
- Converting alarm texts

13.2.1 Creating in-house alarm texts

Creating an alarm text file

1. You can copy a sample file "oem_alarms_eng.ts" from the following directory: **/siemens/sinumerik/hmi/template/lng**
2. Save or create the file in the **/oem/sinumerik/hmi/lng** or **/user/sinumerik/hmi/lng** directory. Create a separate file for each country language that is supported.
3. Assign a unique name to the file, e.g. "my_alarms_eng.ts". You can freely select the file names with the exception of the language code and the extension. The file name always ends with the language code of the particular language and always has the ".ts" extension. For example "my_alarms_deu.ts" for German or "my_alarms_eng.ts" for English.

Creating alarm texts

1. Open the "my_alarms_eng.ts" file.
2. For each alarm text, you must insert your own area in the file that is marked using the `<message>` tag.
3. The `<name>` tag contains the context names. Normally, you should not change the specified name "slaeconv".
4. The `<source>` tag contains the alarm number and the name of the alarm source (source URL). In the example "700000" is the alarm number and "/PLC/PMC" is the name of the alarm source. The possible alarm sources can be found in the tables in Section Range of alarms (Page 222).
5. The `<translation>` tag contains the alarm text.

Example:

```
<!DOCTYPE TS>
<TS>
  <context>
    <name>slaeconv</name>
    <message>
      <source>700000/PLC/PMC</source>
      <translation>First OEM alarm text</translation>
    </message>
    <message>
      <source>700001/PLC/PMC</source>
      <translation>Second OEM alarm text</translation>
    </message>
  </context>
</TS>
```

Registering the alarm text file

You still have to register your alarm text file so that the system knows your alarm texts during the program runtime:

1. You can copy a sample configuration file "oem_slaesvcadapconf.xml" from the following directory: **/siemens/sinumerik/hmi/template/cfg**
2. Save the file in the directory **/oem/sinumerik/hmi/cfg** or **/user/sinumerik/hmi/cfg**
3. Assign the name "slaesvcadapconf.xml" to the file.
4. Open the file and in the <BaseNames> tag, enter the file name of the alarm text file without language code and file extension, e.g. "my_alarms", as "value".
The registration of your alarm text file starts, for example, with the name "OEM_BaseName_01".

Note

The names of the <BaseNames> tag can be freely selected. They must be unique in the system and must not clash with names used by Siemens.

The "Siemens_BaseName_01" name is reserved for Siemens.

Several alarm text files can also be registered. To do this, use names such as "OEM_BaseName_02", "OEM_BaseName_03", etc.

Example:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!-- Configuration of the Solutionline Alarm & Event Service Adapter
-->
<CONFIGURATION>
  <AlarmTexts>
    <BaseNames>
      <OEM_BaseName_01 type="QString" value="my_alarms"/>
    </BaseNames>
  </AlarmTexts>
  ...
</CONFIGURATION>
```

Restart SINUMERIK Operate

The files must still be converted into a binary format so that the alarm texts can be displayed during the program runtime. This data is only converted during power up.

To do this, restart SINUMERIK Operate. In the same directory where the .ts files are located, files with the same name are created with the ".qm" file extension.

The result of the conversion is written to the "alarmtext_conversion.log" or "oem_text_conversion.log" file. Errors that occur during the conversion, such as syntax errors in a parameter file, are also written to the file.

This file is located in the "/user/sinumerik/hmi/log" directory.

Note

The conversion is only made if the .ts file is newer than the associated .qm file.

13.2.2 Creating texts for indexed alarm parameters

If a string, the so-called "Index identifier", is specified in the alarm text in pointed brackets after a space holder (e.g. "%1<OEM>"), the parameter itself is not inserted in the alarm text, but a text from another alarm text file, a so-called index text file. In this case, the alarm parameter value serves as index in order to select the text in the index file. The index text can, in turn, include parameters, also index parameters.

Note

Explanation for the "Index identifier" string

- "<" ≙ opening pointed bracket "<"
 - ">" ≙ closing pointed bracket ">"
-

Creating an index text file

1. You can copy a sample index text file "oem_indexparams_eng.ts" from the following directory: /siemens/sinumerik/hmi/template/lng
2. Save or create the file in the /oem/sinumerik/hmi/lng or /user/sinumerik/hmi/lng directory. Create a separate file for each country language that is supported.
3. Assign a unique name to the file, e.g. "my_indextexts_eng.ts". You can freely select the file names with the exception of the language code and the extension.
The file name always ends with the language code of the particular language and always has the ".ts" extension. For instance, "my_indextexts_deu.ts" for German or "my_indextexts_eng.ts" for English.

Creating index texts

1. Open the "my_indextexts_eng.ts" file.
2. In the <name> tag, enter a name that you have selected, e.g. "my_context".
3. For each index text, a separate area must be inserted between the <message> and </message> tags.
4. The <source> tag contains the alarm parameter value, e.g. the values "1" and "2".
5. The <translation> tag includes the index text that is displayed if the alarm parameter involved has the value specified between <source> and </source>.

Example

```

<!DOCTYPE TS>
<TS>
  <context>
    <name>my_context</name>
    <message>
      <source>1</source>
      <translation>First OEM parameter text</translation>
    </message>
    <message>
      <source>2</source>
      <translation>Second OEM parameter text</translation>
    </message>
  </context>
</TS>

```

Registering an index text file

You still have to register your index text file so that the system knows your index texts during the program runtime. The registration is made in the "slaesvcadapconf.xml" file.

1. Open the file "slaesvcadapconf.xml" already created for the alarm texts **/oem/sinumerik/hmi/cfg** or **/user/sinumerik/hmi/cfg** directory.
2. Remove the lines "<!-- Start of comment" and "End of comment -->".
3. Enter the so-called identifier, e.g. <Identifier type="QString" value="OEM"/>. The index identifier is always specified in pointed brackets next to the parameter specification in the alarm text, e.g. "%1<&OEM>".
4. Enter the BaseName of the index text file, e.g.
<BaseName type="QString" value="my_indexttexts"/>.
5. Enter the context name that you selected, e.g.
<ContextName type="QString" value="my_context"/>.
6. You can also use several different indices. Then, in this case, a dedicated section must be created for every index between the <IndexTexts> and </IndexTexts> tags. The tags for these sections are called, e.g.
<OEM_IndexText_01>, <OEM_IndexText_02>, <OEM_IndexText_03>, etc.
The tags <IndexText_01> to <IndexText_99> are reserved for Siemens.

Example

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!-- Configuration of the Solutionline Alarm & Event Service Adapter
-->
<CONFIGURATION>
...

```

```
<!-- The following narrated part of the configuration is used
only when 'Indexparameters' (e.g. '%1&lt;OEM&gt;') are used
within OEM alarm texts. In this case the value of the parameter
is used as an index into an additional text list to reference
another text which is placed into the alarm text instead of the
original parameter value. -->
<IndexTexts>
  <OEM_IndexText_01>
    <Identifier type="QString" value="OEM"/>
    <BaseName type="QString" value="my_indexttexts"/>
    <ContextName type="QString" value="my_context"/>
    <MetaTextID type="QString" value="%ParamValue%"/>
  </OEM_IndexText_01>
</IndexTexts>
</CONFIGURATION>
```

Restart SINUMERIK Operate

The files must still be converted into a binary format so that the alarm texts can be displayed during the program runtime. This data is only converted during power up.

To do this, restart SINUMERIK Operate. In the same directory where the .ts files are located, files with the same name are created with the ".qm" file extension.

The result of the conversion is written to the "alarmtext_conversion.log" or "oem_text_conversion.log" file. Errors that occur during the conversion, such as syntax errors in a parameter file, are also written to the file.

This file is saved in the /user/sinumerik/hmi/log directory.

Note

The conversion is only made if the .ts file is newer than the associated .qm file.

13.2.3 Creating part program message texts

Message texts from the part program

In order to output message texts from the part program, use the MSG instruction in the part program. The message texts are referenced via the number after the "\$ character, e.g. MSG("\$4711"). You can configure channel-specific message texts for each NC channel.

Creating message texts

Procedure:

1. You can copy a sample message text file "oem_pp_messages_eng.ts" from the following directory: **/siemens/sinumerik/hmi/template/lng**
2. Save or create the file in the **/oem/sinumerik/hmi/lng** or **/user/sinumerik/hmi/lng** directory. Create a separate file for each country language that is supported.
3. Assign a unique name to the file, e.g. "my_msgs_eng.ts". You can freely select the file names with the exception of the language code and the extension.
The file name always ends with the language code of the particular language and always has the ".ts" extension. For instance, "my_msgs_deu.ts" for German or "my_msgs_eng.ts" for English.
4. Open the "oem_msgs_eng.ts" file and in the <name> tag enter the "partprogmsg01" character string. This is the default for the part program message texts from all channels.
5. For each message text, a separate area must be inserted between the <message> and </message> tags.
6. The <source> tag contains the number from the MSG command in the part program.
7. The <translation> tag contains the message text.

Example

```
<!DOCTYPE TS>
<TS>
  <context>
    <name>partprogmsg01</name>
    <message>
      <source>4711</source>
      <translation>part program message No. 4711</translation>
    </message>
  </context>
</TS>
```

Registering the message text file

You still have to declare your message text file so that the system knows your message texts during the program runtime.

1. You can copy a sample configuration file "oem_slaesvcadapconf.xml" from the following directory: **/siemens/sinumerik/hmi/template/cfg**
2. Save or create the file in the **/oem/sinumerik/hmi/cfg** or **/user/sinumerik/hmi/cfg** directory.

3. Change the name to "slaesvcadapconf.xml".
4. Open the file and in the <BaseNames> tag enter the file name of the message text file without language code and file extension, e.g. "my_msgs", as "value". The registration of your message text file starts, for example, with the name "OEM_BaseName_01".

Note

The names of the <BaseNames> tag can be freely selected. They must be unique in the system and must not clash with names used by Siemens.

The "Siemens_BaseName_01" name is reserved for Siemens.

Several message text files can also be registered. To do this, use names such as "OEM_BaseName_02", "OEM_BaseName_03", etc.

Example

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!-- Configuration of the Solutionline Alarm & Event Service Adapter
-->
<CONFIGURATION>
  <AlarmTexts>
    <BaseNames>
      <OEM_BaseName_01 type="QString" value="my_msgs"/>
    </BaseNames>
  </AlarmTexts>
  ...
</CONFIGURATION>
```

Message texts for several channels

It is possible to assign different message texts the same message number, e.g. the number 4711 in different NC channels. This means, different message texts are output depending on the channel in which the part program is executed.

1. Open the "oem_msgs_eng.ts" file in the editor.
2. Change the text between the two <name> and </name> tags from "partprogmsg01" into "partprogmsgXY", whereby "XY" should be replaced by the particular channel number, e.g. "partprogmsg02" for channel 2.
3. Now open the "slaesvcconf.xml" configuration file.
4. Enter the respective NC channel in the <Connections> tag. Only use the XML tags from the following table to specify the NC channel.

NC channel	XML tag	ContextName
1	PartprogramMessageChannel_01	partprogmsg01
2	PartprogramMessageChannel_02	partprogmsg02

NC channel	XML tag	ContextName
3	PartprogramMessageChannel_03	partprogmsg03
4	PartprogramMessageChannel_04	partprogmsg04
5	PartprogramMessageChannel_05	partprogmsg05
6	PartprogramMessageChannel_06	partprogmsg06
7	PartprogramMessageChannel_07	partprogmsg07
8	PartprogramMessageChannel_08	partprogmsg08
9	PartprogramMessageChannel_09	partprogmsg09
10	PartprogramMessageChannel_10	Partprogmsg10

Example

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!-- Configuration of the Solutionline Alarm & Event Service Adapter
-->
<CONFIGURATION>
  <Connections>
    <PartprogramMessageChannel_02>
      <ContextName type="QString" value="partprogmsg02"/>
    </PartprogramMessageChannel_02>
  </Connections>
</CONFIGURATION>
```

Restart SINUMERIK Operate

The files must still be converted into a binary format so that the alarm texts can be displayed during the program runtime. This data is only converted during power up.

To do this, restart SINUMERIK Operate. In the same directory where the .ts files are located, files with the same name are created with the ".qm" file extension.

The result of the conversion is written to the "alarmtext_conversion.log" or "oem_text_conversion.log" file. Errors that occur during the conversion, such as syntax errors in a parameter file, are also written to the file.

This file is saved in the /user/sinumerik/hmi/log directory.

Note

The conversion is only made if the .ts file is newer than the associated .qm file.

13.2.4 Changing alarm attributes

Color types of the alarms

You can individually change the colors of the alarms and messages displayed in the message line.

You can change the following for each alarm number and alarm source:

- Font color of the alarm/message text
- Background color of the alarm/message text
- Font color of the alarm number
- Background color of the alarm number

Create alarm attribute file

Procedure:

1. You can copy a sample alarm attribute file "oem_slaedatabase.xml" from the directory: / **siemens**/sinumerik/hmi/template/cfg/
2. Save the file in the directory /**oem**/sinumerik/hmi/cfg/ or /**user**/sinumerik/hmi/cfg/
3. Assign a unique name to the file, e.g. "muster_slaedatabase.xml".
The file name can be freely selected; however, it must only contain lower-case letters.

Defining alarm colors

Procedure:

1. Open the file "muster_slaedatabase.xml" in the editor.
2. First, enter which alarm colors you generally wish to change.
To do this, create a separate section <Attribute> for each alarm color that you wish to change in the <Attributes> area.
3. Enter the attribute name of the alarm color to be changed in the <Attribute AttrName=...> tag. You can change the following colors:
 - Font color of the alarm/message text = TEXTCOLOR
 - Background color of the alarm/message text = TEXTBACKGROUNDCOLOR
 - Font color of the alarm number = NUMBERCOLOR
 - Background color of the alarm number = NUMBERBACKGROUNDCOLOR

4. Alarm sources are defined in the <Sources> area for whose alarms the alarm colors are to be changed, e.g. the alarm sources "/HMI" and "/PLC/PMC".

Note

If you insert further alarm sources, make sure that the alarm number is always assigned to the correct alarm source.

The possible values for source ID and URL can be found in the table in Section Range of alarms (Page 222).

5. In the <Alarms> tag, create a separate sub-area for each alarm or for an alarm number range.
6. Enter the alarm number in the <Alarm AlarmID= "... "> tag, or enter the alarm number range in the <Range FromAlarmID= "... " ToAlarmID= "... "> tag.

7. Enter the required color values in the following tags:

```
<TEXTCOLOR>
<TEXTBACKGROUNDCOLOR>
<NUMBERCOLOR>
<NUMBERBACKGROUNDCOLOR>
```

The attributes of the color values are defined by specifying an RGB value:

- An RGB value always starts with the "#" character.
- The individual color values R, G and B are represented by double-digit hexadecimal numbers. Syntax: "#RRGGBB", e.g. "#FF9A00".

Example

Note the order in the XML configuration file for the assignment of the colors to alarm number ranges:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE SlAeAlarmAttributes>
<SlAeAlarmAttributes Version="01.00.00.00">
<Types>
  <Type TypeName="Condition" TypeID="32">
    <Category Version="1.0" CatID="1">
      <CatDescr>Alarms of the SINUMERIK 840D sl.</CatDescr>
      <Attributes>
        <Attribute AttrName="TEXTCOLOR" AttrID="5003"
          AttrDataType="10">
          <AttrDescr>
            Text color of an alarm used when displayed within the
            header panel.
          </AttrDescr>
        </Attribute>
        <Attribute AttrName="TEXTBACKGROUNDCOLOR" AttrID="5004"
          AttrDataType="10">
```

```

        <AttrDescr>
            Background color of an alarm used when displayed within
            the header panel.
        </AttrDescr>
    </Attribute>
    <Attribute AttrName="NUMBERCOLOR" AttrID="5005"
    AttrDataType="10">
        <AttrDescr>
            Text color of an alarm used when displayed within the
            header panel.
        </AttrDescr>
    </Attribute>
    <Attribute AttrName=" NUMBERBACKGROUNDCOLOR " AttrID="5006"
    AttrDataType="10">
        <AttrDescr> Background color of an alarm used when
        displayed within the header panel.
        </AttrDescr>
    </Attribute>
</Attributes>
</Category>
</Type>
</Types>
<Sources>
    <Source> CatLink="1" SourceID="10000" SourceURL="/HMI">
        <Alarms>
            <Alarm AlarmID="130000">
                <TEXTCOLOR>#000000</TEXTCOLOR>
                <TEXTBACKGROUNDCOLOR>#FFFFFF</TEXTBACKGROUNDCOLOR>
                <NUMBERCOLOR>#FFFFFF</NUMBERCOLOR>
                <NUMBERBACKGROUNDCOLOR>#000000</NUMBERBACKGROUNDCOLOR>
            </Alarm>
        </Alarms>
    </Source>
    <Source> CatLink="1" SourceID="51" SourceURL="/PLC/PMC">
        <Alarms>
            <Range FromAlarmID="700100" ToAlarmID="700199">
                <TEXTCOLOR>#000000</TEXTCOLOR>
                <NUMBERCOLOR>#00FF00</NUMBERCOLOR>
            </Range>
            <Alarm AlarmID="700000">
                <TEXTCOLOR>#000000</TEXTCOLOR>
                <TEXTBACKGROUNDCOLOR>#FFFFFF</TEXTBACKGROUNDCOLOR>
                <NUMBERCOLOR>#FFFFFF</NUMBERCOLOR>
            </Alarm>
        </Alarms>
    </Source>
</Sources>

```

```
<NUMBERBACKGROUNDCOLOR>#000000</NUMBERBACKGROUNDCOLOR>
</Alarm>
</Alarms>
</Source>
</Sources>
</SlAeAlarmAttributes>
```

Registering the alarm attribute file

The alarm attribute file with the color definitions must be registered in order that the alarm colors can be assigned to the alarms during the program runtime:

1. You can copy a sample configuration file "oem_slaesvcconf.xml" from the following directory: **/siemens/sinumerik/hmi/template/cfg**
2. Save or create the file in the **/oem/sinumerik/hmi/cfg** or **/user/sinumerik/hmi/cfg** directory.
3. Name the file "slaesvcconf.xml".
4. Open the file and in the <DataBases> tag enter, for example, the file name of the alarm attribute file without language code and file extension, e.g. "muster_slaedatabase", as "value". The registration of your alarm attribute file starts, for example, with the name "OEM_DataBase_01".

Note

The names of the <DataBases> tag can be freely selected. They must be unique in the system and must not clash with names used by Siemens.

The "Siemens_DataBase_01" name is reserved for Siemens.

If you register additional alarm attribute files, i.e. you wish to include them in the configuration file, use names such as "OEM_DataBase_02", "OEM_DataBase_03", etc.

Restart SINUMERIK Operate

The files still have to be converted into a binary format in order that the alarm colors become effective during the program runtime. This data is only converted during power up.

To do this, restart SINUMERIK Operate. A file with the same name is created with the ".hmi" file extension, e.g. "sample_slaedatabase.hmi" in the same directory in which the alarm attribute file is also located.

The result of the conversion is written to the file "oem_ae_database_conversion.log". This file is located in the "/user/sinumerik/hmi/log" directory.

Note

The conversion is only carried out if the XML file is newer than the associated hmi file.

13.2.5 Replacing standard alarm texts

You can replace alarm texts of the standard SINUMERIK Operate with own alarm texts.

Creating an alarm text file

1. You can copy a sample alarm text file "oem_alarms_eng.ts" from the following directory: /**siemens**/sinumerik/hmi/template/lng
2. Save or create the file in the /**oem**/sinumerik/hmi/lng or /**user**/sinumerik/hmi/lng directory. Create a separate file for each country language that is supported.
3. Assign the file a user-specific name, e.g. "my_nck_alarms_eng.ts". You can freely select the file names with the exception of the language code and the extension. The file name always ends with the language code of the particular language and always has the ".ts" extension. For example "my_nck_alarms_deu.ts" for German or "my_nck_alarms_eng.ts" for English.

Creating alarm texts

1. Open the "my_nck_alarms_eng.ts" file.
2. A separate area must be inserted for each alarm text, marked by the <message> tag.
3. The <name> tag contains the context names. Change the name and ensure that the name is unique in the system, e.g. "myNckAlarms".

Note

The name for the <name> tag can always be freely selected. However, the name must not clash with the "slaeconv" name used by Siemens.

4. The <source> tag contains the alarm number of the standard alarm, e.g. 10000.
5. The <translation> tag contains the actual alarm text, e.g. "OEM alarm text example for NCK alarm 10.000".

Example

```
<!DOCTYPE TS>
<TS>
  <context>
    <name>myNckAlarms</name>
    <message>
      <source>10000/NCK</source>
      <translation> OEM alarm text example for NCK alarm 10.000</translation>
    </message>
  </context>
</TS>
```

Registering the alarm text file

You still have to register your alarm text file so that the system knows your alarm texts during the program runtime:

1. You can copy a sample configuration file "oem_slaesvcadapconf.xml" from the following directory: **/siemens/sinumerik/hmi/template/cfg**
2. Save or create the file in the **/oem/sinumerik/hmi/cfg** or **/user/sinumerik/hmi/cfg** directory.
3. Assign the name "slaesvcadapconf.xml" to the file.
4. Open the file and in the <BaseNames> tag enter the file name of your alarm text file without language code and file extension, e.g. "my_nck_alarms", as "value". The registration of your alarm text file starts, for example, with the name "Siemens_BaseName_01".

Note

The names of the <BaseNames> tag can be freely selected. They must be unique in the system and must not clash with names used by Siemens.

The "Siemens_BaseName_01" name is reserved for Siemens.

Several alarm text files can also be registered. To do this, use names such as "OEM_BaseName_02", "OEM_BaseName_03", etc.

Example:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!-- Configuration of the Solutionline Alarm & Event Service Adapter
-->
<CONFIGURATION>
  <AlarmTexts>
    <BaseNames>
      <OEM_BaseName_01 type="QString" value="my_nck_alarms"/>
    </BaseNames>
  </AlarmTexts>
</CONFIGURATION>
```

Replacing standard alarm texts

You still have to define which standard alarm texts should be replaced by your alarm texts so that during the program runtime, the original alarm text is replaced by your alarm text.

1. Create a new alarm attribute file (see Changing alarm attributes (Page 214)) in the **/oem/sinumerik/hmi/cfg** or **/user/sinumerik/hmi/cfg** directory. The file name can be freely selected, e.g. "my_nck_alarms_db.xml".
2. Open the file.
3. The file always comprises the two areas <Types> and <Sources>.
4. The <sources> tag includes the reference, assigned to an alarm source, e.g. "NCK", to the associated alarm text.

5. Link the contents of the <context> <name> tags with the contents of the <message> <source> tags from your alarm text file.
 - <context> <name> = the context name that you assigned
 - <message> <source> = the number of the alarm whose text is to be replaced.
 E.g. enter the following link in the alarm attribute file: <MSGTEXT>myNckAlarms I 10000</MSGTEXT>
 Make sure that the context name and the alarm number are separated by the pipe character "|".
6. If you include an additional NCK alarm, copy the range from <Alarms> to </Alarms> and adapt the alarm number.
7. If you include an additional alarm from another number range (e.g. PLC alarm), copy the range <Sources> to </Sources> and then adapt the SourceID and SourceURL as well as the alarm number.
 You can take the corresponding SourceID and SourceURL from the table in Section Range of alarms (Page 222).

Note

A dedicated link must be created for each standard alarm for which the text is to be replaced.

Example

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE SIaAlarmAttributes>
<SIaAlarmAttributes Version="01.00.00.00">
  <Types>
    <Type TypeName="Condition" TypeID="32">
      <Category Version="1.0" CatID="1">
        <CatDescr>Alarms of the Sinumerik 810/840 D(i).</CatDescr>
        <Attributes>
        </Attributes>
      </Category>
    </Type>
  </Types>
  <Sources>
    <Source CatLink="1" SourceID="0" SourceURL="/NCK">
      <Alarms>
        <Alarm AlarmID="10000">
          <MSGTEXT>myNckAlarms|10000/NCK</MSGTEXT>
        </Alarm>
      </Alarms>
    </Source>
```



```
</Sources>  
</SlAeAlarmAttributes>
```

Registering the alarm attribute file

The alarm attribute file still has to be registered in order that the standard alarm texts are replaced during the program runtime:

1. You can copy a sample configuration file "oem_slaesvcconf.xml" from the following directory: **/siemens/sinumerik/hmi/template/cfg**
2. Save or create the file in the **/oem/sinumerik/hmi/cfg** or **/user/sinumerik/hmi/cfg** directory.
3. Change the name to "slaesvcconf.xml".
4. Open the file and in the **<DataBases>** tag enter the name of the alarm attribute file, e.g. **<OEM_DataBase_01 type="QString" value="my_nck_alarms_db">**

Note

The names of the **<DataBases>** tag can be freely selected. They must be unique in the system and must not clash with names used by Siemens.

The **<Siemens_DataBase_01>** tag name is reserved for Siemens.

If you register additional alarm attribute files, i.e. you wish to include them in the configuration file, use the names **<OEM_DataBase_02>**, **<OEM_DataBase_03>**, etc.

Restart SINUMERIK Operate

The files with the alarm texts and the attribute file still have to be converted into a binary format in order that the alarm texts can be displayed during the program runtime. This data is only converted during power up.

To do this, restart SINUMERIK Operate. In the same directory where the .ts files or the alarm attribute file are located, files with the same name are created with the ".qm" and ".hmi" file extension.

The result of the conversion is written to the "alarmtext_conversion.log" or "oem_text_conversion.log" file. Errors that occur during the conversion, such as syntax errors in a parameter file, are also written to the file.

This file is located in the **"/user/sinumerik/hmi/log"** directory.

Note

The conversion is only made if the .ts file is newer than the associated .qm file.

13.2.6 Range of alarms

Number ranges of alarms

Numerical range	Description		SourceID	SourceURL
000.000 – 009.999	General alarms		0 (standard NCU)	/NCK //configuration-spec. designation>//NCK
010.000 – 019.999	Channel alarms		100 (1st NCU)	
020.000 – 029.999	Axis/spindle alarms		...	
030.000 – 039.999	Functional alarms	General	9999 (99th NCU)	
040.000 – 059.999		Reserved		
060.000 – 064.999		Siemens cycle alarms		
065.000 – 069.999		Cycle alarms user		
070.000 – 079.999		Compile cycles Manufacturer and OEM		
080.000 – 084.999		Siemens cycles message texts		
085.000 – 089.999		User cycles message texts		
090.000 – 099.999		Reserved		
100.000 – 129.999	System		10.000	/HMI
130.000 – 139.999	OEM			
140.000 – 199.999	Reserved			
200.000 – 299.999	SINAMICS drive		0	/NCK
300.000 – 399.999	Drive and I/O alarms			
400.000 – 499.999	General alarms		51	/PLC/PMC
500.000 – 599.999	Channel alarms			
600.000 – 699.000	Axis/spindle alarms			
700.000 – 799.999	User area			
800.000 – 899.999	Sequencers/graphs			
810.000 – 810.009	System error messages		50 150 (1st NCU)	/PLC/DiagBuffer //<configuration-spec. Designation>/PLC/DiagBuffer
900.001 – 965.999	HMI PRO sl Runtime		0	
966.000 – 999.999	Reserved		0	/NCK

SourceIDs 1 ... 10

The SourceIDs 1 ... 10 have the following relationship:

SourceID	SourceURL
1	/NCK/Channel#1/Partprogram
2	/NCK/Channel#2/Partprogram
3	/NCK/Channel#3/Partprogram
4	/NCK/Channel#4/Partprogram

SourceID	SourceURL
5	/NCK/Channel#5/Partprogram
6	/NCK/Channel#6/Partprogram
7	/NCK/Channel#7/Partprogram
8	/NCK/Channel#8/Partprogram
9	/NCK/Channel#9/Partprogram
10	/NCK/Channel#10/Partprogram

13.2.7 Parameter specifications in alarm texts

Alarm texts can contain alarm parameters (accompanying values) that specify the cause of an alarm in more detail. These parameters are usually numerical values that are transmitted from an alarm source when the alarm is signaled along with the other alarm data.

How the parameters are incorporated into the alarm text is specified via place holders (parameter specifications) in the language-specific alarm texts, e.g. "Channel %1 Axis %2 in machine data %3 defined for several channels".

When an alarm occurs, the text is replaced by the corresponding parameters, e.g. "Channel 5 Axis A3 in machine data 4711 defined for several channels".

Standard parameter specifications

The following table shows the standard parameter specifications:

Table 13-1 Standard parameter specifications

Parameter specifier	Description
%1	First parameter from the alarm data of the alarm source.
%2	Second parameter from the alarm data of the alarm source.
%3	Third parameter from the alarm data of the alarm source.
%4	Fourth parameter from the alarm data of the alarm source.
%5	Fifth parameter from the alarm data of the alarm source. Only for NCK alarms: First part (up to the separator) of the fourth parameter from the alarm data of the alarm source.
%7	Seventh parameter from the alarm data of the alarm source Only for NCK alarms: Third part (between the 2nd and 3rd separator) of the fourth parameter from the alarm data of the alarm source.
%8	Eighth parameter from the alarm data of the alarm source Only for NCK alarms: Fourth part (between the 3rd and 4th separator) of the fourth parameter from the alarm data of the alarm source.
%9	Ninth parameter from the alarm data of the alarm source.
%0	Tenth parameter from the alarm data of the alarm source.

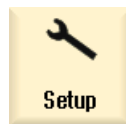
Parameter specifier	Description
%Z	As for %1, in the case of S7-HiGraph alarms the step number of the graph is shown.
%K	Specially for PLC alarms: Second digit of the decimal alarm ID: 123456, corresponds to channel number (0 = channel 10).
%A	Specially for PLC alarms: Third and fourth digit of the decimal alarm ID: 123456, corresponds to the axis number.
%N	Specially for PLC alarms: Fifth and sixth digit of the decimal alarm ID: 123456, corresponds to the signal number.

13.2.8 Opening error file

The errors that occur during the conversion are written to the "alarmtext_conversion.log" or "oem_text_conversion.log" file.

File loc.: /user/sinumerik/hmi/log

Opening the error file



1. Select the "Start-up" operating area.
 2. Press the "System data" softkey.
 3. Open the "System CF card" folder.
 4. Open the storage directory and select the required file.
 5. To open the error file, press the "Open" softkey.
- OR -
- Press the <INPUT> key.

13.3 Configuring an alarm log

Default

Per default, the alarm log contains all alarms and messages - with their incoming and outgoing time stamps - in chronological order since the last boot. The exceptions are messages from the NC part program (msg command).

Contrary to the "Alarm List" or "Messages" window, all of the alarms or messages that are no longer active when the log is displayed are also displayed (historical alarm events).

Specify the number of results

The alarm log is organized as a ring buffer: When the maximum size is exceeded, the oldest entries are overwritten by the new alarm events. Each incoming or outgoing time stamp represents a separate event.

Persistent storage (permanent storage)

If required, the alarm log can be set up as a persistent log that also contains alarm events from before the last time it was switched on. This means that the log is then either time controlled or for each alarm event, is backed-up in an internal binary format in the file system.

Depending on the hardware being used, for the NCU, the alarm log is saved on the CompactFlash card or for the PC/PCU, on the hard disk.

NOTICE

Shorter CF card service life

If the alarm log is continuously written to the CompactFlash card, then only a limited number of write cycles is possible. Therefore, ensure that the storage is only performed when there is a justifiable need! Undo the setting "at every event" if you no longer require that the alarm log is saved.

The CompactFlash card has a service life of over 10 years for an average written data volume of approx. 60 MB/day.

The alarm log is not saved in the default configuration.

Configuring an alarm log

- Via the operator software in the "Diagnostics" operating area
- Via the "slaesvcconf.xml" configuration file

13.3.1 Setting alarm logs from the user interface

Procedure



1. Select the "Diagnostics" operating area.



2. Press the "Alarm log" and "Settings" softkeys.



3. Enter a number in the "Number of entries" field to change the maximum number of administered incoming and outgoing events.
The alarm log is organized as a ring buffer: When the maximum size is exceeded, the oldest entries are overwritten by the new alarm events. Each incoming or outgoing time stamp represents a separate event. You can specify a value between 0 and 32000. Default value is 500.
4. In the "Write mode file" field, under the following entry, select:
 - "off", the changes are not logged (default setting)
 - "for each event", each alarm event immediately initiates that the alarm log is backed-up (persistent save).
 - "time controlled" if the alarm log is to be backed-up again after a certain time. An additional input field called "Time interval" appears in which you can specify a time in seconds.You receive a warning if you only save to a CompactFlash card.

Restart SINUMERIK Operate

The made settings take effect only after a restart of the operating software. To do this, restart SINUMERIK Operate.

13.3.2 Loading alarm log via configuration file

Creating the configuration file

Adjustments are made in file "slaesvcconf.xml". Copy a sample configuration file "oem_alarmprot_slaesvcconf.xml" from the following directory: **/siemens/sinumerik/hmi/template/cfg**

Procedure:

1. Save or create the file in the **/oem/sinumerik/hmi/cfg** or **/user/sinumerik/hmi/cfg** directory.
2. Assign the name "slaesvcconf.xml" to the file.
3. Open the file and enter the number of events to be output in the tag `<Records type .../>`. The preset value is 500. The maximum number depends on the storage medium (hard disk or CompactFlash card).

4. Enter the backup mode in the `<DiskCare type="int" value="-1"/>` tag. The following values are possible:
 - 1: There is no backup of the alarm log (default setting).
 - 0: Each alarm event triggers an immediate backup of the alarm log (persistency backup).
 - >0: Rhythm of the log saving in seconds: When there is a change, the log is time-triggered saved every $n > 0$ seconds (persistency backup).
5. You adapt the filter for the entry type in the `<Filter>` tag. Please observe the following:
 - An alarm event is only entered in the log when it satisfies the filter criterion.
 - If several filters are specified in succession, they are linked by a logical OR.
 - Several filters must be combined in series with the keyword AND for an AND operation.

Note

Each incoming or outgoing event of an alarm or message requires a separate entry, even when they belong to the same alarm or message.

Acknowledgement events are also contained in the alarm log. You require such entries even when these are presently not recognizable in the alarm log.

Filter properties

A filter consists of the following three elements `<identification>` `<relation>` `<value>`.

Entry <code><identification></code>	Description
AlarmID	Alarm number
SourceID	Default values for SourceID and SourceURL can be found in the table in Section Range of alarms (Page 222).
SourceURL	
<code><Attribute name></code>	Arbitrary alarm attribute from the "slaedatabase.xml" file, such as "SEVERITY" or "CLEARINFO".

Entry <code><relation></code>	Description
EQUAL	Equal to
NOT	Not equal to
LOWER	Less than
HIGHER	Greater than

Entry <code><value></code>	Description
Numbers	-
Character strings	-

Cancel criteria

Cancel criterion (ClearInfo)	Alarm source (Source)	Description
0	/HMI	Alarms that are canceled by the HMI.
1	/NCK	Alarms that are canceled by power-on of the NCU.
2		Conditions are canceled by a hardware reset of the NCU.
3		Conditions are deleted by a "CANCEL" command to the NCU.
4		Conditions are canceled by the NCK itself.
5		Conditions are canceled by an "NC Start" command on the NCU.
6		Conditions are canceled by a reset of the mode group (BAG).
7		Conditions are canceled by an "NC Reset" command on the NCU.
8	/PLC	PLC messages of the FB15 (basic program).
9		PLC alarms of the FB15 (basic program).
10		Dialog alarms of the HMI that are canceled by the "Recall" key [^].
11		Reserved
12		S7-PDiag, S7-Graph, S7-HiGraph or other Alarm_S(Q) alarms of the PLC (SFC17/18) with alarm state "not acknowledged"
13		S7-PDiag, S7-Graph, S7-HiGraph or other Alarm_S(Q) alarms of the PLC (SFC17/18) with alarm state "acknowledged".
14	/NCK	Drive alarms via NCK.
15	/NCK	Part program messages.

Examples

Log all alarms with a ClearInfo not equal to 15, i.e. no part program messages:

```
<CONFIGURATION>
  <Protocol>
    <Filters>
      <Siemens_Filter_01 type="QString" value="CLEARINFO NOT 15" />
    </Filters>
  </Protocol>
</CONFIGURATION>
```

Log all alarms with the SourceURL "/NCK" or "/HMI":

```
<CONFIGURATION>
  <Protocol>
    <Filters>
      <Filter_01 type="QString" value="SourceURL EQUAL /NCK" />
      <Filter_02 type="QString" value="SourceURL EQUAL /HMI" />
    </Filters>
  </Protocol>
</CONFIGURATION>
```


In the <FilePath> tag, adapt the path and file name for the file in which the alarm log is stored persistently:

```
<CONFIGURATION>
  <Protocol>
    <Filters>
      <FilePath type="QString" value="$(HMI_INSTALL_DIR)user/
        sinumerik/hmi/ log/alarm_log/slaep_" />
    </Filters>
  </Protocol>
</CONFIGURATION>
```

- **Path**
Environment variables can also be incorporated in the path, e.g. for the installation directory: \$(HMI_INSTALL_DIR).
- **File name:**
A 3-digit number and the file extension "hmi" are automatically added to the specified file name during operation, e.g. "slaep_123.xml". The number is automatically increased when:
 - The file has reached the maximum size (DiskCare = 0), or
 - The specified period has elapsed (DiskCare > 0).
Older files are deleted.

Example: Switching off existing filters

To switch off the existing Siemens filter - without overwriting the filter - proceed as follows:

1. An "empty" filter switches off **all** previously set filters.
Knowledge of the filter designation is not required.
2. For the effect of the filters, the order in which they are set is important:
siemens → addon → oem → user

Example of the filtering of alarms 700000 to 700200:

```
<CONFIGURATION>
  <Protocol>
    <Filters>
      <Filter_00 type="QString" value="" />
      <Filter_01 type="QString" value="CLEARINFO NOT 15 AND
        AlarmID LOWER 700000"/>
      <Filter_02 type="QString" value="CLEARINFO NOT 15 AND
        AlarmID HIGHER 700200"/>
    </Filters>
  </Protocol>
</CONFIGURATION>
```

Result

For the changes to the protocol settings to take effect, restart SINUMERIK Operate.

13.4 PLC alarms with parameters

Introduction

For alarms that are triggered via the PLC blocks SFC17 and SFC18, a parameter can be transferred each time the block is called. If the "Octet string" data type is selected, up to 12 bytes can be transferred.

With an appropriate configuration of the operating software, a maximum of 12 bytes can be interpreted, e.g. also as an array of bytes or as a structure with any order of simple data types. In this way, several parameters can be displayed in the alarm text.

A parameter description must be stored for each alarm in the operating software. Two aspects must be taken into account with this parameter description:

- How the parameter has to be interpreted correctly (data type and, if necessary, length)
- How the parameter has to be prepared for display (string or number, decimal, hexadecimal, binary, etc.)

Point 1 is irrespective of the language selected for the display, point 2 can be different depending on the national language.

The possible parameter descriptions are therefore divided into a language-independent parameter statement and a language-dependent format statement and stored in the following directories:

- Parameter statement: /oem/sinumerik/hmi/cfg/ or /user/sinumerik/hmi/cfg/
- Format statement: /oem/sinumerik/hmi/lng or /user/sinumerik/hmi/lng

The names of the files can be freely selected. The parameter statement is added to the configured data of the relevant alarm in the database in the form of the alarm attribute "HMIPROPARAMDESCR2". Because of the language dependency, the format statement is part of the alarm text.

13.4.1 Definition of a parameter of the octet string data type

Parameter statement

The parameter statement describes how one or more parameters can be defined with the maximum 12-bytes long octet string supplied with the alarm PDU.

Syntax

The following syntax applies for the parameter statement:

<Parameter directive ext>	::=	[% <Parameter description>]
<Parameter description ext>	::=	<No. of the associated value><Element type><Offset>
<No. of the associated value>	::=	Always 1 (reserved for future applications)
<Offset>	::=	Data type-dependent offset of the parameter within the associated value: • BOOL data type (B): Bit offset within the associated value • All except for BOOL data type (Y, W, X, I, D, C, R): Byte offset within the associated value

Element type	Data type
Y	BYTE (8-bit, unsigned)
W	WORD (16-bit, unsigned)
X	DWORD (32-bit, unsigned)
I	INTEGER (16-bit, signed)
D	INTEGER (32-bit, signed)
B	BOOL (1-bit)
C	CHAR (8-bit)
R	REAL (32-bit)

Examples

%1W0%1W2	→	1st parameter: 1st associated value, WORD at byte offset 0 2nd parameter: 1st associated value, WORD at byte offset 2
%1Y0%1Y1%1W2	→	1st parameter: 1st associated value, BYTE at byte offset 0 2nd parameter: 1st associated value, BYTE at byte offset 1 3rd parameter: 1st associated value, WORD at byte offset 2
%1B0%1B1%1B2%1B3%1I1	→	1st parameter: 1st associated value, BOOL at bit offset 0 2nd parameter: 1st associated value, BOOL at bit offset 1 3rd parameter: 1st associated value, BOOL at bit offset 2

4th parameter: 1st associated value, BOOL at bit offset 3

5th parameter: 1st associated value, INTEGER (16-bit, signed) at byte offset 1

Example of an alarm attribute file (how to create an alarm attribute file, see Changing alarm attributes (Page 214)):

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<SlAeAlarmAttributes Version="01.00.00.00">
<Types>
  <Type TypeID="32" TypeName="Condition">
    <Category CatID="1" Version="1.0">
      <CatDescr>Alarms of the SINUMERIK 840D sl.</CatDescr>
      <Attributes>
        <Attribute AttrName="MSGTEXT" AttrID="-1" AttrDataType="10"/>
        <Attribute AttrName="HMIPROPARAMDESCR2" AttrID="5012" AttrDataType="10"/>
      </Attributes>
    </Category>
  </Type>
</Types>
<Sources>
  <Source CatLink="1" SourceID="51" SourceURL="/PLC/PMC">
    <Alarms>
      <!-- Alarm 700000 with three BYTE parameters -->
      <Alarm AlarmID="700000">
        <MSGTEXT>my_oem_plc_alarms_context|700000/PLC/PMC</MSGTEXT>
        <HMIPROPARAMDESCR2>%IY0%IY1%IY2</HMIPROPARAMDESCR2>
      </Alarm>
      <!-- Alarm 700001 with three WORD parameters -->
      <Alarm AlarmID="700001">
        <MSGTEXT>my_oem_plc_alarms_context|700001/PLC/PMC</MSGTEXT>
        <HMIPROPARAMDESCR2>%IW0%IW2%IW4</HMIPROPARAMDESCR2>
      </Alarm>
      <!-- Alarm 700002 with three 16-bit INTEGER parameters -->
      <Alarm AlarmID="700002">
        <MSGTEXT>my_oem_plc_alarms_context|700002/PLC/PMC</MSGTEXT>
        <HMIPROPARAMDESCR2>%II0%II2%IY4</HMIPROPARAMDESCR2>
      </Alarm>
      <!-- Alarm 700003 with four parameters: BYTE -> WORD -> 32-bit INTEGER -> BYTE -->
      <Alarm AlarmID="700003">
        <MSGTEXT>my_oem_plc_alarms_context|700003/PLC/PMC</MSGTEXT>
        <HMIPROPARAMDESCR2>%IY0%IW1%ID3%IY7</HMIPROPARAMDESCR2>
      </Alarm>
```

```

    </Alarms>
  </Source>
</Sources>
</S1AeAlarmAttributes>

```

13.4.2 Definition of the language-dependent formatting

Format statement

The format statement is part of the alarm text and is used as a placeholder for a parameters to be displayed.

Syntax

The following syntax applies for the format statement:

<Format directive>	::=	@ <No. of the parameter description><Format specification> @
%[i]x	::=	Hexadecimal number with i digits
%[i]u	::=	Decimal number without sign with i digits
%[i]d	::=	Decimal number with sign with i digits (incl. sign)
%[i]b	::=	Binary number with i digits
%[i][.y]f	::=	Fixed-point number: Signed value of the form [-]xxx.yyyy, where i specifies the total number of digits including sign and decimal point and y specifies the number of digits after the decimal point. Missing digits both before and after the decimal point are filled with 0. If there are more places after the decimal point than permitted by y, this is rounded off.

If the optional digit specification i is not specified, then only the number of digits is used required to completely display the number. The same applies when the number of digits specified by i is not sufficient to completely display the number including sign, i.e. i is then ignored.

The number contained in the format statement does not relate to the position of the associated value in the alarm PDU, but to the position of the parameter description in the parameter statement. Any number of format statements can be present in an alarm text.

Examples

- @1%b@ → Value from the first parameter description to be displayed as a binary number, e.g. "1011011"
- @2%8X@ → Value from the second parameter description to be displayed as a hexadecimal number with eight digits, missing digits are filled with 0, e.g. "00AF37FE"
- @1%7.2f@ → Value from the first parameter description to be displayed as a fixed point number with seven digits (incl. sign and decimal point) and two digits after the decimal point, missing digits are filled with 0, e.g. "-012.34"

Example of an alarm text file (how you create alarm texts, see Creating in-house alarm texts (Page 206)):

```
<!DOCTYPE TS>
<TS>
<context>
  <name>my_oem_plc_alarms_context</name>
  <message>
    <source>700000/PLC/PMC</source>
    <translation>Text with three BYTE parameters: @1%X@, @2%u@, @3%b@</translation>
  </message>
  <message>
    <source>700001/PLC/PMC</source>
    <translation>Text with three WORD parameters: @1%X@, @2%u@, @3%b@</translation>
  </message>
  <message>
    <source>700002/PLC/PMC</source>
    <translation>Text with three 16-bit INTEGER parameters: @1%X@, @2%u@, @3%b@</translation>
  </message>
  <message>
    <source>700003/PLC/PMC</source>
    <translation>Text with four parameters: @1%X@, @2%u@, @3%r@, @4%b@</translation>
  </message>
</context>
</TS>
```

13.5 Deactivating a warning

Generally, alarms/messages are not used without alarm texts. If the alarm text is missing, the warning "No text available" is output.

To deactivate the warning, add the <ControlFlags> area to your "slaesvcadapconf.xml" configuration file.

You can copy this area from the original file. The "slaesvcadapconf.xml" file is in the following directory: **/siemens/sinumerik/hmi/base**

Setting

In the `<MissingTextWarning type="bool" value="FALSE"/>` tag, deactivate/activate the warning.

Entry	Meaning
TRUE	The warning is displayed.
FALSE	The warning is deactivated.

Example

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<CONFIGURATION>
  <ControlFlags>
    <MissingTextWarning type="bool" value="FALSE"/>
  </ControlFlags>
</CONFIGURATION>
```


Collision avoidance

14.1 Overview

With the aid of collision avoidance, you can avoid collisions and therefore major damage during the machining of a workpiece or when creating programs.



Software option

You require the "Collision avoidance (machine, working area)" software option in order to use this function.



Machine manufacturer

Please refer to the machine manufacturer's specifications.

Collision avoidance is based on a machine model. The kinematics of the machine are described as a kinematic chain. For machine parts, which are to be protected, protection areas are attached to these chains. The geometry of the protection areas is defined using protection area elements. The control then knows how they move in the machine coordinate system depending on the position of the machine axes. You then subsequently define the collision pairs, i.e. two protection areas, which are monitored with respect to one another.

The "Collision avoidance" function regularly calculates the clearance from these protection areas. When two protection areas approach one another, and a specific safety clearance is reached, an alarm is displayed and before the corresponding traversing block, the program is stopped and/or the traversing motion is stopped.

Data of the machine model

The machine model data is entered into the system variables.

Note

Start-up of a machine model

You require access level 1 (manufacturer) for creating and editing a machine model.

Note

The collision monitoring is only valid for single-channel machines.

Note

Referenced axes

The positions of the axes in the machine area must be known so that the protection areas can be monitored. For this reason, the collision avoidance is only active after the referencing.

NOTICE**No complete machine protection**

Incomplete models, (e.g. machine parts, workpieces that have not been modelled or new objects in the working area), are not monitored and can therefore cause collisions.

Collision detection

The detection of a risk of collision is controlled via the following criteria:

- Collision tolerance
The collision tolerance defines the accuracy of the tool modeling. You must note that an increase in the accuracy reduces the performance.
- Safety clearance
The safety clearance sets the minimum clearance that must exist.

For better visualization, you can assign different colors to the various model parts. If there is a risk of a collision, at least one collision pair is highlighted in color.

During simultaneous recording, the active machine model is shown in addition to the simultaneous recording graphic.

References

More detailed explanations of collision avoidance can be found in the following reference:

Function Manual, Special Functions (FB3):

- Section: "Kinematic chain (K7)"
- Section: "Geometric machine modeling (K8)"
- Section: "Collision avoidance (K9)"
- Section: "NC/PLC interface signals (Z3)" > "Collision avoidance (K9)"

14.2 General sequence

The following steps are required to create a machine model:

- Activating the option
- Setting machine data
- Create kinematic structure (with kinematic elements)
- Creating protection areas
- Creating protection area elements
- Define collision pairs

14.3 Set collision avoidance

Enable function

Set the following machine data item in order to enable the function:

MD19830 \$ON_COLLISION_MASK	Functional scope of the collision avoidance
Bit 0	Collision avoidance (machine, tool)

Parameter assignment

No.	Identifier	Meaning
MD10619	\$MN_COLLISION_TOLERANCE	Collision tolerance
MD10622	\$MN_COLLISION_SAFETY_DIST	Safety clearance
MD18896	\$MN_MM_MAXNUM_3D_COLLISION	Memory location for the collision avoidance
MD18880	\$MM_MAXNUM_KIN_CHAIN_ELEM	Maximum number of elements for kinematic chains
MD16800	\$MN_ROOT_KIN_ELEM_NAME	Name of the first element in the kinematic chain
MD18890	\$MN_MM_MAXNUM_3D_PROT_AREAS	Maximum number of protection areas
MD18892	\$MN_MM_MAXNUM_3D_PROT_AREA_ELEM	Maximum number of protection area elements
MD18893	\$MN_MM_MAXNUM_3D_T_PROT_ELEM	Maximum number of tool protection area elements
MD18897	\$MN_MM_MAXNUM_3D_INTERFACE_IN	Maximum number of NC/PLC interface signals for the preactivation of protection areas
MD18895	\$MN_MM_MAXNUM_3D_FACETS	Maximum number of triangles for protection areas
MD18894	\$MN_MM_MAXNUM_3D_FACETS_INTERN	Maximum number of triangles for automatic tool protection areas
MD18899	\$MN_PROT_AREA_TOOL_MASK	Creation mode for automatic tool protection areas
MD51160	\$MNS_ACCESS_WRITE_CA_MACH_JOG	Write protection level, collision monitoring machine jog
MD51161	\$MNS_ACCESS_WRITE_CA_MACH_AUTO	Write protection level, collision monitoring machine automatic
MD51162	\$MNS_ACCESS_WRITE_CA_TOOL	Write protection level collision monitoring tool

Note

If a specific safety clearance has been set for a collision pair via the system variable \$NP_SAFETY_DIST, this has priority over the NC-specific safety clearance set in the MD10622 \$MN_COLLISION_SAFETY_DIST machine data.

NC/PLC interface signals

Signal	Meaning
DB10.DBX58.0 - 7	Collision avoidance: Deactivate protection area group
DB10.DBX234.0 - DBX241.7	Collision avoidance: Activate protection area

Note

The \$NP_BIT_NO protection area can be assigned to an arbitrary bit number of the DB10.DBX234.0 - DBX241.7 interface signals.

The following setting is precondition: \$NP_INIT_STAT == "P" (preactivated or PLC-controlled)

References

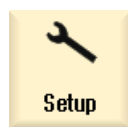
Detailed machine data:

List Manual (book 1), Section "Detailed description of the machine and setting data"

14.4 Graphic machine model editor

This is how you create the machine model from the user interface in a graphic editor that has 2 areas:

- Tree view
In the "Tree" window, model the machine model based on the kinematic chain and allocated protection areas.
- Graphic view
You also have the option of activating a graphic view.
In the "Graphic" window, you can view and check the result of the machine modeling in the various views, and, for example, simultaneously correct in the tree view.
Elements that are selected in the tree are highlighted in color in the graphic view.

Procedure

1. Select the "Start-up" operating area.



2. Press the softkeys "NC" and "Machine model".
The "Tree" window opens.



The "Tree" window opens.
The following folders are available in the tree view:

- Active data
- Examples
- Network/USB



3. Press the "Graphic" softkey in order to use the graphic view.



The "Graphic" window opens in the right half of the editor.

The machine model created in the tree view is displayed in three dimensions.

4. By pressing the active softkey "Graphic" or "Tree" again, you display the graphic view or the tree view.

14.5 Editing the machine model

Functions are available in the tree view of the graphic editor, similar to those in the program editor; these simplify creating and editing a machine model.

Procedure

Copying/cutting/pasting elements

If you want to create a kinematic or protection area element that is similar to an already existing element, then you can save time if you copy the element and insert it at a different position. You have the option, for example of also copying inactive subtrees into the active tree.

Copying/cutting and pasting elements

1. Position the cursor on the desired element.
2. Press the "Copy" softkey.



- OR -



Press the "Cut" softkey.



3. Position the cursor at the required position and press the "Paste" softkey.



4. Press the "OK" softkey if you wish to attach the element to the chain as the next element.

The element is inserted and the name is automatically extended by a long underscore and incremented.

Note:

You have the option of changing the name of the element in the associated input window.

- OR -



Press the "Parallel" softkey if you wish to insert the element parallel to the selected element.

Note:

The "Parallel" softkey is not available for protection areas.

Note**Inserting the same element**

If you insert the same element, then this is automatically inserted in front of the selected element.

Deleting elements

1. Place the cursor on the desired element and press the "Delete element" softkey.



2. Press the "OK" softkey to confirm the deletion.

The selected object is removed.

All subsequent elements are moved into the "Free elements" folder.

- OR -



If the selected element is linked with additional elements, which are also to be deleted, then press the "With successors" softkey.

Note**Deleting a machine model**

If you select the "Active data" folder in the tree view, and press the "Delete elements" softkey, and confirm the subsequent prompt with "OK", the machine model is deleted with all elements and protection areas.

If you do not wish to remove the STL files, then cancel delete with "No" to the prompt "3D files".

Activating the machine model

Each time that an element of the collision model is entered or changed, then the collision model is automatically activated. If the collision model is incorrect, an error message is output and the incorrect element has a pink background. The "Activate data" softkeys becomes active, and you can use this to again initiate possibly acknowledged alarms for the collision model.



1. Press the "Activate data" softkey.

All of the error messages are displayed again, and you have the option of specifically resolving these by modifying the data.

14.6 Changing and adapting the machine model view

14.6.1 Enlarging or reducing the machine graphic

Precondition

- You are in the graphic machine model editor.
- The "Graphic" and "Autozoom" softkeys are active.

Procedure

	1. Position the cursor in the graphic. 2. Press the "Zoom +" softkey if you wish to enlarge the size of the section.
	- OR -
	Press the "Zoom -" softkey if you wish to reduce the size of the section.
	- OR -
	Press the "Auto zoom" softkeys if you wish to automatically adapt the segment to the size of the window. The automatic scaling function takes into account the largest expansion of the model.

Note

Selected section

The selected sections and size changes are kept as long as the program is selected.

Note

Mouse operation

Using the mouse wheel, you have the option of shifting the model in the graphic view.

14.6.2 Changing the section of the machine graphic

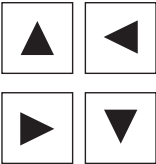
Use the magnifying glass if you would like to move, enlarge or reduce the size of the section when viewing the machine model, e.g. to view details and to subsequently display the complete model.

Using the magnifying glass, you can define your own section and then enlarge or reduce its size.

Requirement

- You are in the graphic machine model editor.
- The "Graphic" and "Autozoom" softkeys are active.

Procedure

	1. Position the cursor in the graphic.
	2. Press the "Magnifying glass" softkey. A magnifying glass in the shape of a rectangular frame appears.
	3. Press the "Magnify +" or <+> softkey to enlarge the frame. - OR - Press the "Magnify -" or <-> softkey to reduce the frame. - OR - Press one of the cursor keys to move the frame up, down, left or right.
	
	4. Press the "Accept" softkey to accept the selected section.

14.6.3 Rotating and shifting the machine graphics

You have the option of rotating and shifting the position of the machine graphics so that you can view the model from all sides.

Requirement

- You are in the graphic machine model editor.
- The "Graphic" and "Autozoom" softkeys are active.

Procedure

	1. Position the cursor in the graphic.
	2. Press the "Rotate view" softkey.



3. Press the "Arrow right", "Arrow left", "Arrow up", "Arrow down", "Arrow clockwise" or "Arrow counter-clockwise" softkeys to change the position of the machine model.

...



...



- OR -



Press the >Cursor right>, <Cursor left>, <Cursor top> or <Cursor bottom> keys to shift the machine model.



- OR -

Keep the <Shift> key pressed and rotate the machine model in the desired direction using the appropriate cursor keys.

Note

Working with a mouse

Using a mouse, you have the option of rotating and shifting the model in the graphic view.

- Move the model with the left-hand mouse key pressed to shift the model.
 - Move the model with the right-hand mouse key pressed to rotate the model.
-

14.6.4 Various machine model views

For the graphic display of the machine model, you can choose between various views. This allows you to always optimally view the actual machining situation.

The following views are available:

- From the front
View parallel to the Y axis
- Side view
View parallel to the X axis
- Top view
View parallel to the Z axis

Precondition

- You are in the graphic machine model editor.
- The "Graphic" and "Autozoom" softkeys are active.

Procedure

	1. Position the cursor in the graphic
	2. Press the ">>" and "Views" softkeys.
	2. Press the "From front" softkey if you wish to view the machine model from the front.
	- OR - Press the "Side view" softkey if you wish to view the machine model from the side.
	- OR - Press the "Top view" softkey if you wish to view the machine model from the top.

14.6.5 Displaying protection area elements for collision avoidance

In the graphic display, you can select between the following display types:

- **Display model**
In the graphic display of the machine model, all elements are displayed where the "Display" or "Display+monitoring" setting was selected for element value "Use".
- **Monitoring model**
In the graphic display of the machine model, all elements are displayed where the "Monitoring" or "Display+monitoring" setting was selected for element value "Use".

Both display versions can be simultaneously selected.

Precondition

- You are in the graphic machine model editor.
- The "Graphic" and "Autozoom" softkeys are active.

Procedure

	1. Position the cursor in the graphic
	2. Press the ">>" softkey.
	The machine model with all display and monitoring elements is displayed in the graphics window.
	3. By deselecting, i.e. by pressing the corresponding softkey again, you can display and hide the display elements and monitoring elements.

Setting the detail level

If, for protection elements, different detail levels are specified, then soft-keys are available that you can use to change the detail level.



1. Press the "Detail level +" softkey to display the next higher detail level.

- OR -



Press the "Detail level -" softkey to display the next lower detail level.

Note

One of the two display versions is always active!

14.7 Creating a kinematic structure

14.7.1 Kinematic elements

14.7.1.1 Creating a kinematic chain element

An exact kinematic description of the machine is required in order to be able to detect collisions.

To do this, you describe the kinematic sequence of the linear and rotary axes as well as the position and direction in relationship to the world coordinate system (ROOT). This informs the control, for example, whether the Y axis is moved by the X axis or vice versa.

Kinematic elements

The following elements are available to create the kinematic chain:

- Rotation and parallel rotation
- Linear axis and parallel linear axis
- Rotary axis and parallel rotary axis
- Offset and parallel offset

World coordinate system

To uniquely describe the kinematic structure of a machine, all elements of the kinematic chain are referred to the world coordinate system.

You have the option of freely selecting the origin and orientation of the world coordinate system. However, it is recommended that you orientate the world coordinate system so that the coordinate axes are arranged in the positive traversing direction of the machine linear main axes.

Precondition

- The option is set.
- The machine data are set.

Procedure**Defining a ROOT element**

1. Machine data 16800 \$MN_ROOT_KIN_ELEM_NAME contains the name of the ROOT element. This defines the start of the kinematic chain.
All subsequent, parameterized elements and subchains that are connected with this element are part of the currently effective kinematic chain.
2. Switch to the graphic machine model editor.
3. Position the cursor to the "Active data" folder and press the "New element" and "Kinematic element" softkeys.
The "New kinematic element" selection window opens.
4. For example, select the "Offset" element and press the "OK" softkey.
The "Offset" window is opened to define the shift vector.
The first element of the kinematic chain can be an offset, a rotation, a linear axis or also a rotation axis.
5. Press the "OK" softkey to insert the element.

**Attaching additional kinematic elements**

Depending on your machine, you can attach any number of elements or branch off subchains.

See also

Kinematic rotation and parallel kinematic rotation (Page 248)

Linear axis and parallel linear axis (Page 249)

Rotary axis and parallel rotary axis (Page 249)

Offset and parallel offset (Page 250)

14.7.1.2 Kinematic rotation and parallel kinematic rotation

You can define fixed rotation of a subsequent element in the "Rotation" window.

Display in the tree topology

Kinematic rotations are identified by this symbol in the tree topology.

Element values

Parameter	Meaning
Name	Designation, which is displayed in the tree structure.
Rotary axis (X)	X component of the vector, around which the system rotates.
Rotary axis (Y)	Y component of the vector, around which the system rotates.
Rotary axis (Z)	Z component of the vector, around which the system rotates.
Angle of rotation	Angle, around which the system rotates in the direction of the rotation vector.

See also

Creating a kinematic chain element (Page 247)

14.7.1.3 Linear axis and parallel linear axis

You define the position and orientation of the axes, whose traversing motion describes a straight line, in the "Linear axis" window.

Display in the tree topology



Dynamic linear axes are identified by this symbol in the tree topology.

Element values

Parameter	Meaning
Name	Designation, which is displayed in the tree structure.
Axis (X)	X component of the direction vector
Axis (Y)	Y component of the direction vector
Axis (Z)	Z component of the direction vector
Axis name	Specification of the axis identifier (see MD 10000)
Axis offset	Additive offset between the actual position of the axis, and the value used by the collision monitoring.

See also

Creating a kinematic chain element (Page 247)

14.7.1.4 Rotary axis and parallel rotary axis

In the "Rotation axis" window, you define a rotary axis, whose traversing motion describes a circle or arc. This means that you change the orientation of the subsequent elements.

Display in the tree topology

Dynamic rotation axes are identified by this symbol in the tree topology.

Element values

Parameter	Meaning
Name	Designation, which is displayed in the tree structure.
Rotary axis (X)	X component of the vector, around which the system rotates.
Rotary axis (Y)	Y component of the vector, around which the system rotates.
Rotary axis (Z)	Z component of the vector, around which the system rotates.
Axis name	Specification of the axis identifier (see MD 10000)
Axis offset	Additive offset between the actual position of the axis, and the value used by the collision monitoring.

See also

Creating a kinematic chain element (Page 247)

14.7.1.5 Offset and parallel offset

In the "Offset" window you define fixed linear offsets of the subsequent elements, which are linked with this element.

Display in the tree topology

Static offset vectors are designated by this symbol in the tree topology.

Element values

Parameter	Meaning
Name	Designation, which is displayed in the tree structure.
Offset (X)	X component of the direction vector.
Offset (Y)	Y component of the direction vector.
Offset (Z)	Z component of the direction vector.

See also

Creating a kinematic chain element (Page 247)

14.7.2 Protection areas

14.7.2.1 Creating protection areas

- At locations where there are machine components, e.g. table or spindle in the machining area, machine protection areas are defined.
- At the positions where tools can be in the collision space, tool protection areas (e.g. for a turning tool of the cutting part, i.e. the cutting edge) are created and defined.

Note

Tool carriers are not modeled

Tool carriers, clamping or cutting edge holders are not emulated in the machine modeling and are not checked for collision.

Precondition

- The option is set
- Machine data is set
- A kinematic chain has been created

Procedure

	1.	Position the cursor at the kinematic chain element to which you wish to attach a protection area.
	2.	Press the "New element" softkey.
	3.	Press the "Collision element" softkey. The "New Collision Element" selection window opens.
	4.	Select "Machine protection area" if you wish to model a machine part. - OR - Select "Tool protection area" if you wish to model a tool part.
	5.	Press the "OK" softkey to insert the element. The "Machine Protection Area" or "Tool Protection Area" window opens.
	6.	Enter the desired values and press the "OK" softkey to confirm your entries.

Attach additional protection areas

You have the option of attaching as many protection areas as you require, corresponding to your machine.

See also

Tool protection area (Page 252)

Machine protection area (Page 253)

14.7.2.2 Tool protection area

Parameterize model use in the "Tool Protection Area" window.

You define from where the control retrieves data to generate the machine model.

Further, you define the parameters to monitor and display the element.

For a protection area, the protection area element is automatically taken from the tool management data. For instance, if you define a tool protection area for a tool in the spindle, then for a tool change (M6), a new tool protection area element is created with the data of the new tool.

Note

For the automatic tool protection area, a spindle must exist in the channel.

Note**Attaching a tool protection area**

"Frame" type tool protection area elements may be attached to tool protection areas.

Display in the tree topology

Tool protection areas are identified by this symbol in the tree topology.

Element values

Parameter	Meaning
Name	Designation that is displayed in the tree structure.
Color 	Selection of the color
Detail level	Defines from which detail level the protection area or the protection area elements are displayed on the user interface. • Lowest detail level: 0 • Highest detail level: 3

Parameter	Meaning
PLC bit	Only for pre-activation - 1 Not activated 0 - 63 Activation of the collision avoidance is controlled using the bit entered here. The maximum number of PLC bits depends on MD18897. It can be a maximum of 64.
Activation 	- Activates the collision avoidance for the protection area. - Deactivated Deactivates the collision avoidance for the protection area. - Preactivated Activation of the collision avoidance is controlled using the bit entered into the "PLC bit".
TO unit	Specifies the TO area in which the magazine and tool data are saved.
Magazine	When magazine management is active: Specifies the magazine number in which the tool or magazine location is located (e.g. for spindle 9998).
Location	- Without magazine management: 1 - For revolvers: Number of the magazine location

See also

Creating protection areas (Page 251)

14.7.2.3 Machine protection area

In the "Machine protection area" window, you model the machine parts, for example, a table. You define the parameters to monitor and display the element.

Display in the tree topology



Machine protection areas are identified by this symbol in the tree topology.

Element values

Parameter	Meaning
Name	Designation, which is displayed in the tree structure.
Color 	Selection of the color

Parameter	Meaning
Detail level	Defines from which detail level the protection area or the protection area elements are displayed on the user interface. • Lowest detail level: 0 • Highest detail level: 3
PLC bit	Only for pre-activation • - 1 Not activated • 0 - 63 Activation of the collision avoidance is controlled using the bit entered here. The maximum number of PLC bits depends on MD18897. It can be a maximum of 64.
Activation 	• Activated Activates the collision avoidance for the protection area. • Deactivated Deactivates the collision avoidance for the protection area. • Preactivated Activation of the collision avoidance is controlled using the bit entered into the "PLC bit".

See also

Creating a kinematic chain element (Page 247)

Creating protection areas (Page 251)

14.7.3 Protection area elements

14.7.3.1 Creating protection area elements

To protect machine parts and tools, model a protection area with the following basic bodies:

- Frame or parallel frame
- Box or parallel box
- Sphere or parallel sphere
- Cylinder or parallel cylinder
- File or parallel 3D file

Note**Parallel protection area elements**

You have the possibility of creating two protection area elements in parallel in a hierarchic level of the tree structure. This means that parallel elements have the previous element as the same origin.

To create two elements in parallel with one another at one level, use a "parallel" element.

If you create an additional protection area element, then this is shifted into the next level.

Preconditions

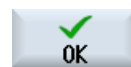
- A kinematic chain has been created
- The cursor is on a protection area (machine or tool protection area) or at a protection area element.

Procedure

1. Press the "New element" softkey.



2. The "New Collision Element" selection window opens.



3. Select the required basic form (e.g. box to model a table) and press the "OK" softkey.

The corresponding window to enter the element values opens.



4. Define the values and press the "OK" softkey to confirm the entries.

Attaching additional protection area elements

You have the option of attaching as many protection area elements as you require, corresponding to your machine.

See also

Frame and parallel frame protection area element (Page 255)

Box and parallel box protection area elements (Page 257)

Sphere and parallel sphere protection area elements (Page 259)

File and parallel 3D file protection area element (Page 262)

Cylinder and parallel cylinder protection area element (Page 260)

14.7.3.2 Frame and parallel frame protection area element

You define a coordinate system transformation that will take effect for the following protection area elements in the "Frame" window.

With elements of this type, you define, for example, additional work offsets or rotations between protection area elements.

Note

A work offset or coordinate rotation affects all the following protection area elements.

Display in the tree topology

Coordinate system transformations are identified by this symbol in the tree topology.

Element values

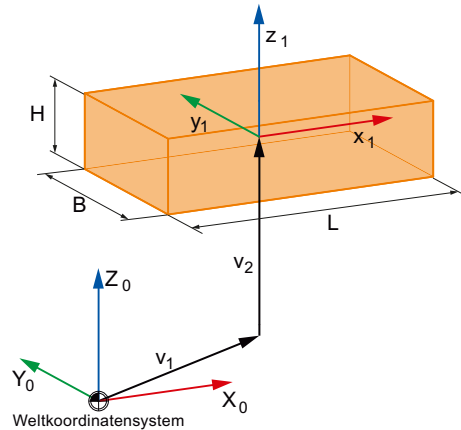
Parameter	Meaning
Name	Designation that is displayed in the tree structure.
Offset (X)	X component of the direction vector.
Offset (Y)	Y component of the direction vector.
Offset (Z)	Z component of the direction vector.
Rotary axis (X)	X component of the rotation vector.
Rotary axis (Y)	Y component of the rotation vector.
Rotary axis (Z)	Z component of the rotation vector.
Angle of rotation	The value by which the system rotates around the rotation vector.

See also

Creating protection area elements (Page 254)

14.7.3.3 Box and parallel box protection area elements

You can specify the values for the display or collision monitoring of a box-shaped protection area element in the "Box" window.



V1 Sum of the offset and rotation between this element and the root element.

V2 Offsets and rotation from the parameters of this element.

Display in the tree topology



Box-shaped protection area elements are identified by this symbol in the tree topology.

Element values

Parameter	Meaning
Name	Designation that is displayed in the tree structure.
Color 	Selection of the color for the protection area element. • Activate the checkbox "From protection area", if the color of the associated tool and/or machine protection area should be applied. Deactivate the checkbox "From protection area" in order to select any color. • Select the required color from the selection list.
Detail level	Defines from which detail level the protection area or the protection area elements are displayed on the user interface. • Activate the checkbox "From protection area", if the detail level of the associated tool and/or machine protection area should be applied. Deactivate the checkbox "From protection area", if you wish to define your own detail level for the element. • Enter the desired level of detail: – Lowest detail level: 0 – Highest detail level: 3

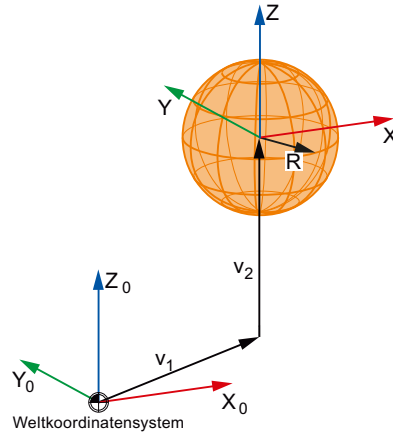
Parameter	Meaning
Use 	• Display The protection area element is shown in the graphic view. The element is only visible if the "Display model" softkey is active. The element is not monitored for collision. • Monitoring The protection area element is used for the collision avoidance. The element is only visible if the "Monit. model" softkey is active. • Display and monitoring The protection area element is used for display in the graphic view and for collision avoidance. The element is visible in both display versions.
Length (X)	Length of the cube in the X direction.
Width (Y)	Width of the cube in the Y direction.
Height (Z)	Height of the cube in the Z direction.
Offset (X)	X component of the direction vector.
Offset (Y)	Y component of the direction vector.
Offset (Z)	Z component of the direction vector.
Rotary axis (X)	X component of the rotation vector.
Rotary axis (Y)	Y component of the rotation vector.
Rotary axis (Z)	Z component of the rotation vector.
Angle of rotation	The value by which the system rotates around the rotation vector.

See also

Creating protection area elements (Page 254)

14.7.3.4 Sphere and parallel sphere protection area elements

You can specify the dimensions and position as well as values for the display and collision monitoring of a sphere-shaped protection area element in the "Sphere" window.



V1 Sum of the offset and rotation between this element and the root element.

V2 Offsets and rotation from the parameters of this element.

Display in the tree topology



Sphere-shaped protection area elements are identified by this symbol in the tree topology.

Element values

Parameter	Meaning
Name	Designation, which is displayed in the tree structure.
Color 	Selection of the color for the protection area element. • Activate the checkbox "from protection area", if the color of the associated tool and/or machine protection area should be applied. Deactivate the checkbox "from protection area" in order to select any color. • Select the required color from the selection list.
Detail level	Defines from which detail level the protection area or the protection area elements are displayed on the user interface. • Activate the checkbox "from protection area", if the detail level of the associated tool and/or machine protection area should be applied. Deactivate the check box "from protection area", if you wish to define your own detail level for the element. • Enter the desired level of detail: – Lowest detail level: 0 – Highest detail level: 3

Parameter	Meaning
Use 	• Display The protection area element is shown in the graphic view. The element is only visible if the "Display model" softkey is active. The element is not monitored for collision. • Monitoring The protection area element is used for the collision avoidance. The element is only visible if the "Monit. model" softkey is active. • Display + monitoring The protection area element is used for display in the graphic view and for collision avoidance. The element is visible in both display versions.
Radius	Specification of the radius
Offset (X)	X component of the direction vector.
Offset (Y)	Y component of the direction vector.
Offset (Z)	Z component of the direction vector.
Rotary axis (X)	X component of the rotation vector.
Rotary axis (Y)	Y component of the rotation vector.
Rotary axis (Z)	Z component of the rotation vector.
Angle of rotation	The value by which the system rotates around the rotation vector.

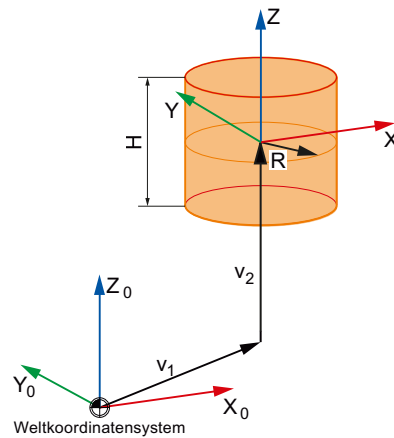
See also

Creating protection area elements (Page 254)

File and parallel 3D file protection area element (Page 262)

14.7.3.5 Cylinder and parallel cylinder protection area element

You can specify the dimensions and position as well as values for the display and collision monitoring of a cylindrical protection area element in the "Cylinder" window.



V1 Sum of the offset and rotation between this element and the root element.

V2 Offsets and rotation from the parameters of this element.

Display in the tree topology



Cylinder-shaped protection area elements are identified by this symbol in the tree topology.

Element values

Parameter	Meaning
Name	Designation, which is displayed in the tree structure.
Color 	Selection of the color for the protection area element. • Activate the checkbox "from protection area", if the color of the associated tool and/or machine protection area should be applied. Deactivate the checkbox "from protection area" in order to select any color. • Select the required color from the selection list.
Detail level	Defines from which detail level the protection area or the protection area elements are displayed on the user interface. • Activate the checkbox "from protection area", if the detail level of the associated tool and/or machine protection area should be applied. Deactivate the check box "from protection area", if you wish to define your own detail level for the element. • Enter the desired level of detail: – Lowest detail level: 0 – Highest detail level: 3

Parameter	Meaning
Use 	• Display The protection area element is shown in the graphic view. The element is only visible if the "Display model" softkey is active. The element is not monitored for collision. • Monitoring The protection area element is used for the collision avoidance. The element is only visible if the "Monit. model" softkey is active. • Display + monitoring The protection area element is used for display in the graphic view and for collision avoidance. The element is visible in both display versions.
Height (Z)	Height of the cylinder in the Z direction.
Radius	Specification of the radius
Offset (X)	X component of the direction vector.
Offset (Y)	Y component of the direction vector.
Offset (Z)	Z component of the direction vector.
Rotary axis (X)	X component of the rotation vector.
Rotary axis (Y)	Y component of the rotation vector.
Rotary axis (Z)	Z component of the rotation vector.
Angle of rotation	The value by which the system rotates around the rotation vector.

See also

Creating protection area elements (Page 254)

14.7.3.6 File and parallel 3D file protection area element

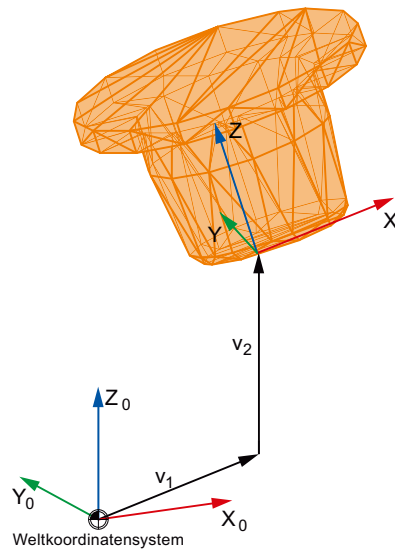
You can specify the dimensions and position as well as values for the display and collision monitoring of a sphere-shaped protection area element in the "Sphere" window.

The file used here contains a grid of triangular areas in the STL format, with which you can display bodies with complicated shapes.

Note**Coordinate origin**

The coordinate origin of the file does not have to match the coordinate system of the machine.

You align the coordinate system using offsets and rotations.



V1 Sum of the offset and rotation between this element and the root element.

V2 Offsets and rotation from the parameters of this element.

Display in the tree topology



STL files are identified by this symbol in the tree topology.

Element values

Parameter	Meaning
Name	Designation, which is displayed in the tree structure.
Color 	Selection of the color for the protection area element. • Activate the checkbox "from protection area", if the color of the associated tool and/or machine protection area should be applied. Deactivate the checkbox "from protection area" in order to select any color. • Select the required color from the selection list.
Detail level	Defines from which detail level the protection area or the protection area elements are displayed on the user interface. • Activate the checkbox "from protection area", if the detail level of the associated tool and/or machine protection area should be applied. Deactivate the check box "from protection area", if you wish to define your own detail level for the element. • Enter the desired level of detail: – Lowest detail level: 0 – Highest detail level: 3

Parameter	Meaning
Use 	• Display The protection area element is shown in the graphic view. The element is only visible if the "Display model" softkey is active. The element is not monitored for collision. • Monitoring The protection area element is used for the collision avoidance. The element is only visible if the "Monit. model" softkey is active. • Display + monitoring The protection area element is used for display in the graphic view and for collision avoidance. The element is visible in both display versions.
File name	Name of the file with the description of the protection area element in STL format.
Offset (X)	X component of the direction vector.
Offset (Y)	Y component of the direction vector.
Offset (Z)	Z component of the direction vector.
Rotary axis (X)	X component of the rotation vector.
Rotary axis (Y)	Y component of the rotation vector.
Rotary axis (Z)	Z component of the rotation vector.
Angle of rotation	The value by which the system rotates around the rotation vector.

See also

Creating protection area elements (Page 254)

Sphere and parallel sphere protection area elements (Page 259)

14.7.4 Collision pair**14.7.4.1 Creating a collision pair**

Using collision pairs, you define the two protection areas, which should mutually monitor themselves regarding collision.

Precondition

A kinematic chain has been created and includes valid protection areas.

Procedure



1. Position the cursor in the tree view on the "Collision groups" folder. The "New collision element" selection window opens.
2. Select "Collision pair" and press the "OK" softkey.
3. Enter the values and press the "OK" softkey to confirm the entries.

See also

Collision pair (Page 265)

14.7.4.2 Collision pair

In the "Collision pair" window you define the protection areas, which should mutually monitor themselves regarding collision. Further, you select a safety clearance here.

Display in the tree topology



Collision pairs are identified by this symbol in the tree topology.

Element values

Parameter	Meaning
1. Area	Designation of the first protection area that is to be monitored.
2. Area	Designation of the second protection area that is to be monitored.
Safety clearance	Defining the safety clearance: • 0 For the safety clearance, the value that has been defined in MD10622 is taken. • >0 The value that you entered here is used for the safety clearance.

Note

In order that the defined protection areas are checked for collision, they must be activated and the collision avoidance activated.

See also

Creating a collision pair (Page 264)

14.7.5 Supplementary condition

Tool changes

If, in a machine model, tools change as a result of being replaced/exchanged or wear without programming a tool offset selection (D number), then the recalculation of the machine model must be explicitly requested. You will find more precise information in:

Function Manual Special Functions (FB3); collision avoidance (K9),
Chapter: "Function description" > "Tools"

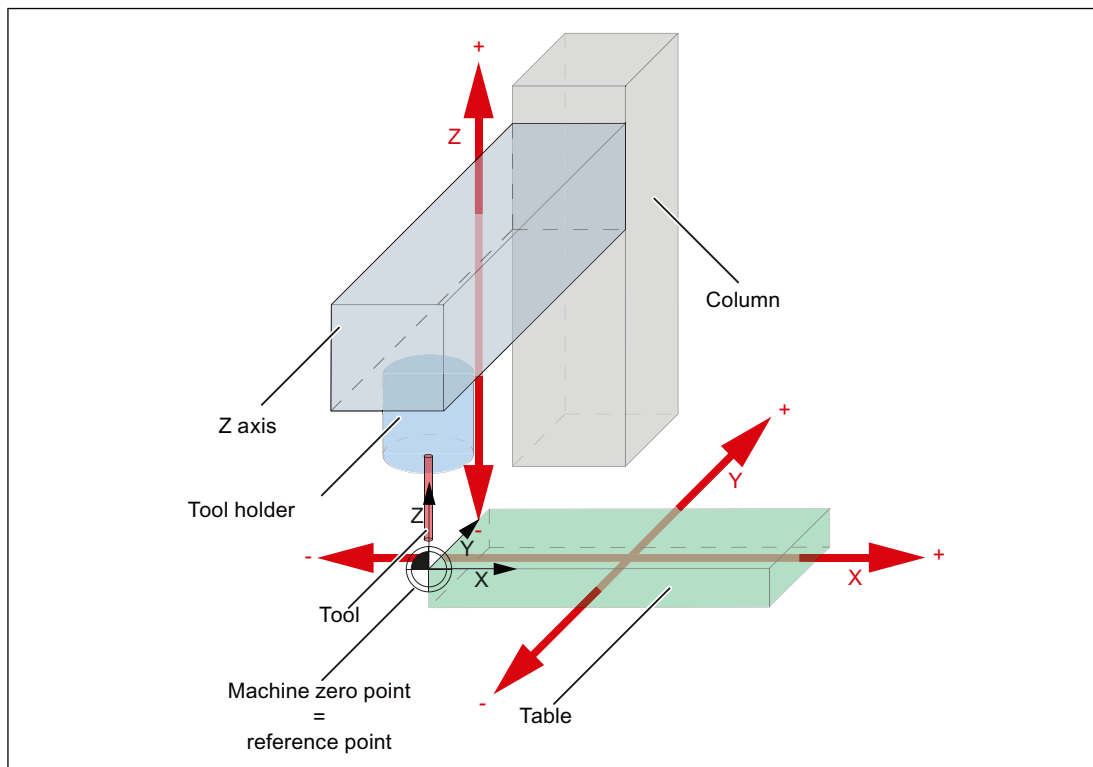
14.8 Collision avoidance example

14.8.1 Fundamentals

General Information

The following example contains the machine model elements of a simple 3-axis milling machine for a simplified kinematic chain.

Machine model elements

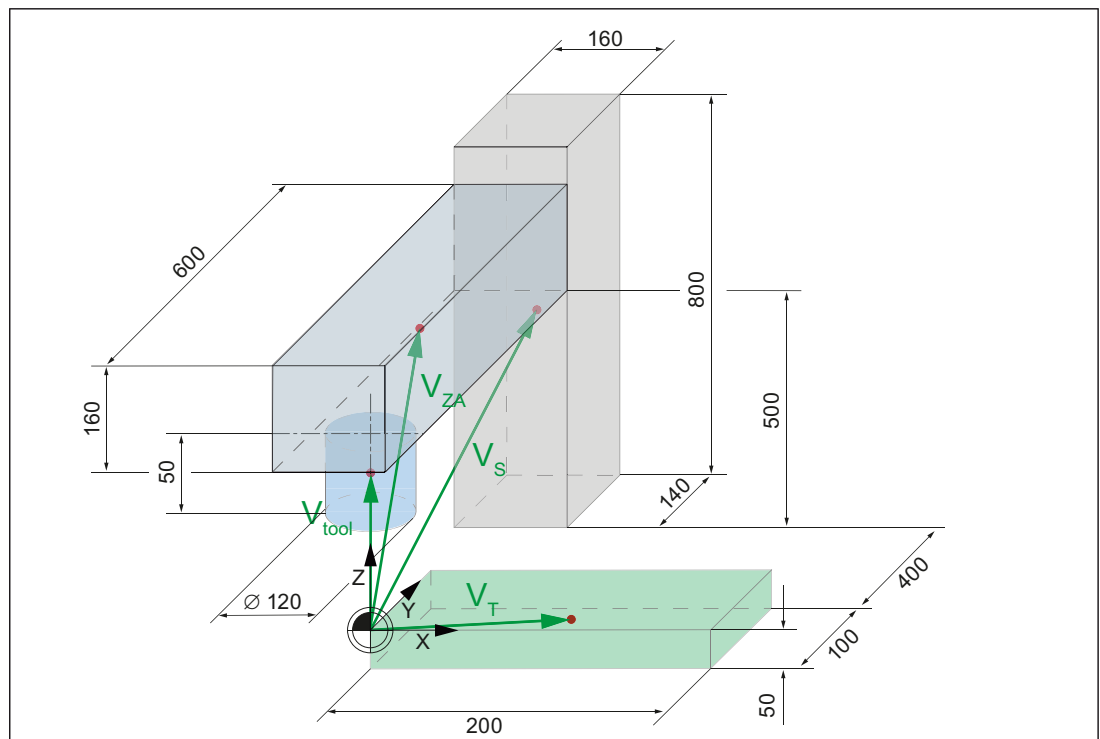


The following elements are defined for the example of the 3-axis milling machine:

Element	Color	Direction of movement
Table	Green	X, Y
Z axis	Gray	Z
Column	Gray	None
Tool holder	Blue	Z
Tool (when used)	red	Z
Machine zero	None	None

Geometry

Each machine-model element has a defined geometry and an offset vector to the element center starting with the machine zero:

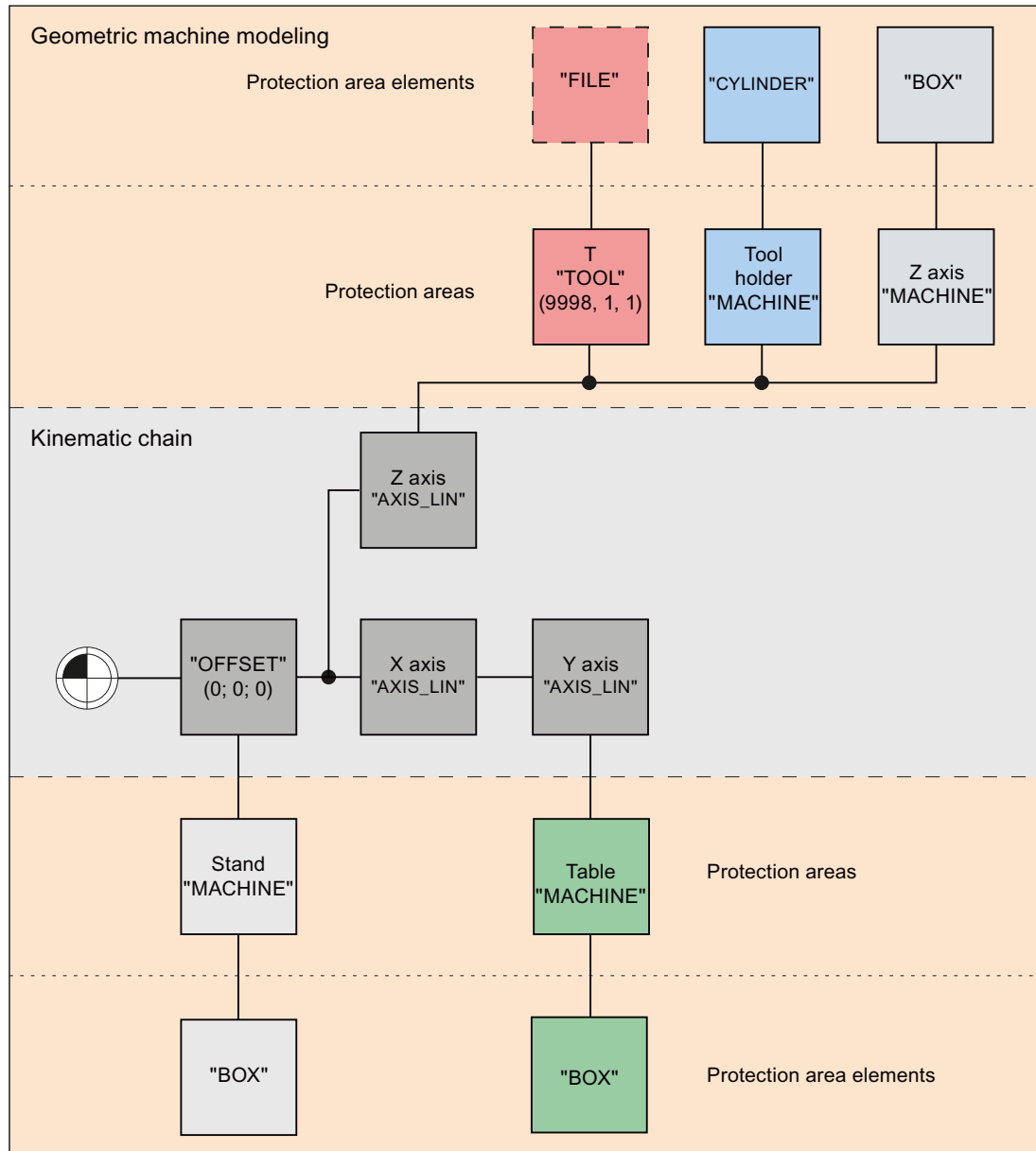


The following offset vectors are defined for the example of the 3-axis milling machine:

Element	Offset vector
Tool holder	$V_{\text{tool}} = (0; 0; 25)$
Z axis	$V_{\text{ZA}} = (0; 200; 130)$
Column	$V_{\text{S}} = (0; 570; 350)$
Table	$V_{\text{T}} = (100; 50; -25)$

Structure of the geometric machine modeling

The kinematic chain begins with the first offset kinematic element that represents the machine zero. The other kinematic elements form the linear axes X, Y and parallel Z. The protection area is used to define the basic bodies of the machine and tools. The movable protection areas are linked with the associated linear axis; the unmovable protection area is linked with the offset.



14.8.2 Create machine model example

14.8.2.1 Creating kinematic elements

Offset



1. Mark the "Active data" folder and press the "New element" softkey.
A selection window with kinematic elements opens.
2. Select the "Offset" kinematic element and press the "OK" softkey.
The appropriate input window for the element values opens in the lower part of the tree view.
3. Enter the required values:
Name: ROOT
Offset (X): 0
Offset (Y): 0
Offset (Z): 0
4. Press the "OK" softkey to save the values.

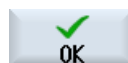


- OR -

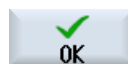


Press the "Cancel" softkey to close the input window.

Linear axis X under offset



1. Mark the "ROOT" kinematic element and press the "New element" softkey.
A selection window with kinematic elements opens.
2. Select the "Linear axis" kinematic element and press the "OK" softkey.
The appropriate input window for the element values opens in the lower part of the tree view.
3. Enter the required values:
Name: X axis
Axis (X): 1
Axis (Y): 0
Axis (Z): 0
Axis name: X1
Axis offset: 0
4. Press the "OK" softkey to save the values.



- OR -



Press the "Cancel" softkey to close the input window.

Linear axis Y under linear axis X



1. Mark the "X axis" kinematic element and press the "New element" softkey. A selection window with kinematic elements opens.
2. Select the "Linear axis" kinematic element and press the "OK" softkey. The appropriate input window for the element values opens in the lower part of the tree view.
3. Enter the required values:
 Name: Y axis
 Axis (X): 0
 Axis (Y): 1
 Axis (Z): 0
 Axis name: Y1
 Axis offset: 0
4. Press the "OK" softkey to save the values.



- OR -



Press the "Cancel" softkey to close the input window.

Linear axis Z parallel to linear axis X



1. Mark the "X axis" kinematic element and press the "New element" softkey. A selection window with kinematic elements opens.
2. Select the "Parallel linear axis" kinematic element and press the "OK" softkey. The appropriate input window for the element values opens in the lower part of the tree view.
3. Enter the required values:
 Name: Z axis
 Axis (X): 0
 Axis (Y): 0
 Axis (Z): 1
 Axis name: Z1
 Axis offset: 0
4. Press the "OK" softkey to save the values.



- OR -



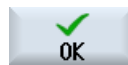
Press the "Cancel" softkey to close the input window.

14.8.2.2 Creating machine protection areas

Stand under offset



1. Mark the "ROOT" kinematic element and press the "New element" softkey.
2. Press the "Collision element" softkey.
3. In the "New collision element" window, select the collision element "Machine protection area" and press the "OK" softkey.
4. Enter the following values:
 Name: Stand
 Color: Gray
 Detail level: 0
 PLC bit: -1
 Activation: Activated
5. Press the "OK" softkey to save the values.



Box element under stand



1. Mark the "Stand" machine protection area and press the "New element" softkey.
2. Select the "Box" protection area element and press the "OK" softkey.
3. Enter the following values:
 Name: SBE stands
 Color: From protection area
 Detail level: From protection area
 Application: Display
 Length (X): 160
 Width (Y): 140
 Height (Z): 800
 Offset (X): 0
 Offset (Y): 570
 Offset (Z): 350
4. Press the "OK" softkey to save the values.

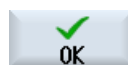


Table under linear axis Y



1. Mark the "Y axis" kinematic element and press the "New element" softkey.
2. Press the "Collision element" softkey.
3. Select the "Machine protection area" collision element and press the "OK" softkey.
4. Enter the following values:
 Name: Table
 Color: Green
 PLC bit: -1
 Detail level: 0
 Activation: Activated
5. Press the "OK" softkey to save the values.

Box element under table



1. Mark the "Table" machine protection area and press the "New element" softkey.
2. Select the "Box" protection area element and press the "OK" softkey.
3. Enter the following values:
 Name: SBE table
 Color: From protection area
 Detail level: From protection area
 Application: Display and monitoring
 Length (X): 200
 Width (Y): 100
 Height (Z): 50
 Offset (X): 100
 Offset (Y): 50
 Offset (Z): -25
4. Press the "OK" softkey to save the values.

Z axis under linear axis Z



1. Mark the "Z axis" kinematic element and press the "New element" softkey.
2. Press the "Collision element" softkey.



3. Select the "Machine protection area" collision element and press the "OK" softkey.

4. Enter the following values:

Name: Z axis

Color: Gray

Detail level: 0

PLC bit: -1

Activation: Activated



5. Press the "OK" softkey to save the values.

Box element under Z axis



1. Select the "Z axis" machine protection area and press the "New element" softkey.



2. Select the "Box" protection area element and press the "OK" softkey.

3. Enter the following values:

Name: SBE-Z axis

Color: From protection area

Detail level: From protection area

Application: Display and monitoring

Length (X): 160

Width (Y): 600

Height (Z): 160

Offset (X): 0

Offset (Y): 200

Offset (Z): 130



4. Press the "OK" softkey to save the values.

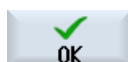
Tool holder under linear axis Z



1. Mark the "Z axis" kinematic element and press the "New element" softkey.



2. Press the "Collision element" softkey.



3. Select the "Machine protection area" collision element and press the "OK" softkey.

4. Enter the following values:

Name: Tool holder

Color: Blue

Detail level: 0



PLC bit: -1

Activation: Activated

5. Press the "OK" softkey to save the values.

Cylinder element under tool holder



1. Mark the "Tool holder" machine protection area and press the "New element" softkey.



2. Select the "Cylinder" protection area element and press the "OK" softkey.

3. Enter the following values:

Name: SBE tool holder

Color: From protection area

Detail level: From protection area

Application: Display and monitoring

Height (Z): 50

Radius: 60

Offset (X): 0

Offset (Y): 0

Offset (Z): 25



4. Press the "OK" softkey to save the values.

14.8.2.3 Creating a tool protection area

Tool under linear axis Z



1. Mark the "Z axis" kinematic element and press the "New element" softkey.



2. Press the "Collision element" softkey.



3. In the selection box, select "Tool protection area" collision element and press the "OK" softkey.

4. Enter the following values:

Name: T

Color: red

Detail level: 0

PLC bit: -1

Activation: Activated

TO unit: 1

Magazine: 9998.



Location: 1

5. Press the "OK" softkey to save the values.

14.8.2.4 Creating collision pairs

Tool/table collision pair



1. Mark the "Collision groups" folder and press the "New element" softkey. A selection window for the collision element opens.



2. Select the "Collision pair" collision element and press the "OK" softkey. The appropriate input window for the element values opens in the lower part of the tree view.

3. Enter the required values:

1st range: T

2nd range: Table



4. Press the "OK" softkey to save the values.

- OR -



Press the "Cancel" softkey to close the input window.

Tool holder / table collision pair



1. Mark the "Collision groups" folder and press the "New element" softkey. A selection window for the collision element opens.



2. Select the "Collision pair" collision element and press the "OK" softkey. The appropriate input window for the element values opens in the lower part of the tree view.

3. Enter the required values:

1st range: Tool holder

2nd range: Table



4. Press the "OK" softkey to save the values.

- OR -



Press the "Cancel" softkey to close the input window.

Data backup

Data backup

The following times are recommended for performing a data backup:

- After a start-up
- After changing machine-specific settings
- After replacing a hardware component
- For a software upgrade
- Before the activation of memory-configuring machine data

Note

A start-up archive is saved as a file of the type ".arc" (archive).

There are various ways of creating and reloading archives via the SINUMERIK Operate user interface.

- Standard archive: Data can be selected specifically in the data tree and backed up by pressing the "System data" softkey.
- Start-up archive: The "Start-up archive" softkey offers the following selection:
 - Create and read-in a start-up archive
 - Create PLC hardware upgrade archive (only SDBs)
 - Create and read in archive of original status

Storage locations

Archives can be stored in the following directories:

- CompactFlash card at: System CF-Card/**user**/sinumerik/data/archive or System CF-Card/**oem**/sinumerik/data/archive
- PCU: F:\hmis**user**\sinumerik\data\archive or \oem\sinnerik\data\archive
- All configured logical drives (USB, network drives)

Note

USB FlashDrive

USB-flash drives are not suitable as persistent memory media.

15.1 Creating a start-up archive

Overview

A start-up archive can be generated, for example, after the control has been commissioned. To do this, control components can be saved individually or jointly. It is also possible to transfer the backed-up data to other controls so that they can be updated to the same status.

Note

When generating the start-up archive, programs that have been shifted outside the system (network drive) are not taken into account. Only the programs from the NC memory, global part program memories that have been possibly created as well as the local drive are archived.

Requirement

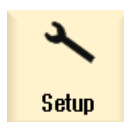
You require at least access level 2 (service).

Data of the control components

Control components		Data
NC data		Machine data Setting data Option data Global (GUD) and local (LUD) user data Tool and magazine data Protection zone data R parameters Zero offsets Workpieces, global part programs and subprograms Standard and user cycles Definitions and macros
	With compensation data	• QEC - quadrant error compensation • CEC - operation/angularity compensation • EEC - leadscrew pitch / encoder error compensation It only makes sense to archive machine-specific compensation data if the start-up file is reloaded into the same control.
	with compile cycles	Compile cycles (*.elf) are displayed, if compile cycles are available.
PLC data (including CP)		OB (organization blocks) FB (function blocks) SFB (system function blocks) FC (functions) SFC (system functions) DB (data blocks) SDB (system data blocks)

Control components		Data
Drive data		Archiving the drive data, either in binary or ASCII format.
HMI data, all		
HMI data, selection:		If data is available, you must set the tick in order to activate the components.
	Texts	User texts, alarm texts
	Templates	Individual templates, workpiece templates
	Applications	HMI applications, OEM applications
	Engineering data	Engineering data
	Configuration	Configurations, incl. display of machine data
	Help	Help files
	Version data	Version data
	Logs	Error logs
	Program lists	Program lists
	Dictionaries	Dictionaries
	Data backups	Files located in the data tree of the "System data" in the "HMI data / data backups" directory.
	Programs on local drive	Programs contained in the user memory area of the CompactFlash card.
Comment		Entry field for possible comments regarding the start-up archive.
Created by		Entry field for the creator as well as creation date of the start-up archive.

Procedure



1. Select the "Start-up" operating area.

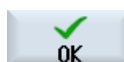


2. Press the menu forward key.



3. Press the "Start-up archive" softkey.

The "Start-up" window opens.

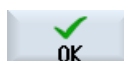


4. Activate "Create a start-up archive" and press the "OK" softkey.

The "Create Start-up Archive" window opens.

5. Select the desired control components.

6. When required, enter a comment as well as your name and the date of creation.

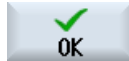


7. Press the "OK" softkey.

The "Generate Archive: Select Storage Location" window opens.



8. Position the cursor at the required storage location, press the "Search" softkey and enter the required search term in the search dialog. Then press the "OK" softkey if you wish to search for a certain directory or subdirectory.



Note: The place holders "*" (replaces any character string) and "?" (replaces any character) make it easier for you to perform a search.

- OR -



Select the required storage location, press the "New directory" softkey, enter the required name in the "New Directory" window and press the "OK" softkey to create a directory.



9. Press the "OK" softkey.

The "Generate Archive: Name" window opens.



10. Enter the required name and press the "OK" softkey.

An archive file with the ARC format type is stored in the selected directory.

15.2 Reading-in a start-up archive

NOTICE

Loss of additional languages during software upgrade

All the system languages are overwritten during software upgrades. If there are more languages in the system than in the upgrade archive, these additional languages are lost when the archive is read in. It is not possible to back up the languages before software upgrade.

Note

The reading in of a start-up archive with drive configurations results in the overwriting of the drives in the logdrive.ini file. The system creates a backup copy with the name logdrive_ini.sav or logdrive.oldstyle. Before the archive is read in, the new drive configurations are activated.

Note

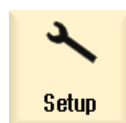
To prevent the NCU from shutting down unintentionally, you must disable all energy saving profiles before you read in a start-up archive. For more detailed information about the energy saving profiles topic, see:

Operating Manuals for SINUMERIK Operate, Chapter "Ctrl Energy".

Requirement

You require at least access level 2 (service).

Procedure



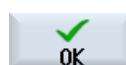
1. Select the "Start-up" operating area.



Press the menu forward key.



2. Press the "Start-up archive" softkey.



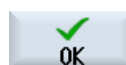
3. Activate "Read-in a start-up archive" and press the "OK" softkey.
The "Select Start-up Archive" window opens.

4. Select a storage location and position the cursor at the required start-up archive (ARC).

- OR -



Select the required storage location, press the "Search" softkey and enter the required search term. Then press the "OK" softkey if you wish to search for a certain directory, subdirectory or start-up archive.



5. Press the "OK" softkey.

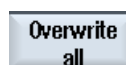
The "Read-in Start-up Archive" window opens.

You obtain an overview with path details, version information, name of the archive, etc., as well as a list of the archived components.

Remove the tick of a checkbox to exclude a component from the read-in process.

6. Press the "OK" softkey to download the selected archive file.

The "Read-in Archive" window opens and a progress message box appears for the read-in process.



7. Press the "Overwrite all" softkey if you wish to overwrite the existing files.

- OR -



Press the "No overwriting" softkey if you wish to keep existing files.

- OR -



Press the "Skip" softkey if you only wish to overwrite certain files.

You will then obtain a "Read error log for archive" in which the skipped or overwritten files are listed.

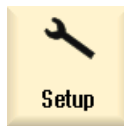


8. Press the "Cancel" softkey to cancel the read-in process.

15.3 Backing up the hardware configuration

The hardware is configured via the system data blocks (SDB) and only these are stored in the archive.

Procedure



1. Select the "Start-up" operating area.



Press the menu forward key.



2. Press the "Start-up archive" softkey.

The "Start-up" window opens.



3. Activate "Create PLC hardware upgrade archive (SDBs only)" and press the "OK" softkey.

The "Create PLC hardware upgrade archive (SDBs only): Select Storage Location" window opens.

4. Select the desired storage location in the displayed data tree.

- OR -



Press the "New directory" softkey to create a separate directory.

The "New Directory" window opens.



5. Enter the required name and press the "OK" softkey.

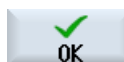
The directory is created below the created folder.



6. Press the "OK" softkey.

The "Generate Archive: Name" window opens.

The file type of the archive is ARC and is displayed in the window. Enter the desired name and press the "OK" softkey.



The archive is generated and stored in the selected directory.

15.4 Creating an archive with original data

You can select this type of start-up in order to save the original status of the control. The files are archived in the "original.arc" file.

The file is stored on the CompactFlash card in the Archives/Manufacturer directory.

If there is no "original.arc" file in the directory, it is recommended that an archive be generated from the data of the original control state as delivered.

As for "Create start-up", control components can be saved individually or jointly.

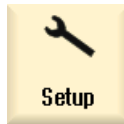
Requirement

In order to save NC data, you require the access level 2 (service).

To back up programs/workpieces, you need access level 6 (keyswitch position 1).

Data of the control components

Control components		Data
NC data		Machine data Setting data Option data Global (GUD) and local (LUD) user data Tool and magazine data Protection zone data R parameters Zero offsets Compensation data Workpieces, global part programs and subprograms Standard and user cycles Definitions and macros
	With compensation data	• QEC - quadrant error compensation • CEC - operation/angularity compensation • EEC - leadscrew pitch / encoder error compensation • It only makes sense to archive machine-specific compensation data if the series start-up file is reloaded into the same control.
	With compile cycles	Compile cycles (*.elf) are displayed, if compile cycles are available.
PLC data (including CP)		OB (organization blocks) FB (function blocks) SFB (system function blocks) FC (functions) SFC (system functions) DB (data blocks) SDB (system data blocks)
HMI data, all		
HMI data, selection:		
	Cycle storage	Definitions and cycles
	Texts	User texts, alarm texts
	Templates	Individual templates, workpiece templates
	Applications	HMI applications, OEM applications
	Engineering data	Engineering data
	Configuration	Configurations, incl. display machine data
	Help	Help files
	Version data	Version data
	Logs	Error logs
	User views	Individually compiled tables with selected machine and setting data.
	Dictionaries	Dictionaries
	Programs on local drive	Programs contained in the user memory area of the CompactFlash card.

Procedure

1. Select the "Start-up" operating area.



2. Press the menu forward key.



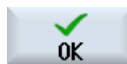
3. Press the "Start-up archive" softkey.

The "Start-up" window opens.



4. Activate the "Create archive of original status" checkbox and press the "OK" softkey.

The "Create Start-up for Original Status" window opens.



5. Select the desired control components.

6. When required, enter a comment into the field as well as your name and the date of creation - and then press the "OK" softkey.

A warning is output if an "original.arc" archive file already exists.



7. Press the "OK" softkey to overwrite the file.

A new archive file is generated and stored in the Archives/Manufacturer directory.

- OR -



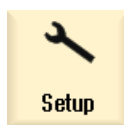
Press the "Cancel" softkey so that the existing file is not overwritten.

15.5 Reading in an archive with original data

The data of the original controller state is stored in the "original.arc" file. If you want to restore the controller to the as-delivered state, you can read in the original data archive.

Precondition

You require access level 2 (service).

Procedure

1. Select the "Start-up" operating area.



2. Press the menu forward key.



3. Press the "Start-up archive" softkey.

The "Start-up" window opens.



4. Activate the "Read in archive of original status" checkbox and press the "OK" softkey.

SINUMERIK Operate automatically accesses the "original.arc" file.

A window opens prompting whether you want to perform a start-up.



5. Press the "OK" softkey.

The read-in process is started and a progress message box is displayed in the window.

Any errors that occur are then displayed in the "Write Error Log for Archive" window.

15.6 Generating the complete archive

You have the option of generating an archive of the complete data of your control on an external storage medium that you can provide to Technical Support for diagnostic purposes.

Procedure

1. Insert the storage medium in the USB port.
2. Press the key combination <Ctrl> + <Alt> + S.
All data required for diagnostics are compiled in an archive. Operator control is not possible while the archive is being generated.
3. You will receive a message once archiving has been completed.
The system assigns an archive name and this is: CompletArchiv<Date>_<Time>.arc.
The dialog box is closed and you can re-operate the control.

15.7 Serial interface (V24 / RS232)

15.7.1 Reading-in and reading-out archives via a serial interface

Availability of the V24 serial interface

You have the option of reading-out and reading-in archives in the "Program manager" operating area as well as in the "Startup" operating area via the V24 serial interface.

- SINUMERIK Operate on the NCU:
The softkeys for the V24 interface are available as soon as an option module is connected and the slot is occupied.
- SINUMERIK Operate on the PCU:
The softkeys for the V24 interface are always available.

Reading-out archives

The files to be sent (directories or individual files) are zipped in an archive (*.arc). If you send an archive (*.arc), this is sent directly without being additionally zipped. If you have selected an archive (*.arc) together with an additional file (e.g. directory), then these are zipped into a new archive and are then sent.

Reading-in archives

Only archives can be read-in via the V24 interface. These are transferred and then subsequently unzipped.

Note

Start-up archive

If you read in a start-up archive via the V24 interface, then this is immediately activated.

Externally processing the punched tape format

If you wish to externally process an archive, then generate this in the punched tape format.

Procedure



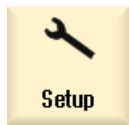
1. Select the "Program manager" operating area, and press the "NC" or "Local drive" softkey.



...



- OR -



- Select the "Startup" operating area and press the "System data" softkey.



Reading-out archives

2. Select the directories or the files that you wish to send via V24.
3. Press the ">>" and "Archive" softkeys.





4. Press the "V24 send" softkey.

- OR -

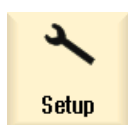
Reading in an archive



Press the "V24 receive" softkey if you wish to read-in files via V24.

15.7.2 Setting interface parameters

Procedure



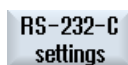
1. Select the "Start-up" operating area.



2. Press the "System data" softkey.



3. Press the ">>" and "Archive" softkeys.



4. Select the "RS-232-C settings" softkey.
The "Interface: RS-232-C" window is opened and the interface settings are displayed.



5. Press the "Details" softkey if you wish to view and process additional settings for the interface.

RS-232-C settings

Parameter	Meaning
Protocol	The following protocol is supported for transfer via the RS-232-C: RTS/CTS
Transfer	It is also possible to use a secure protocol for data transfer (ZMODEM protocol). - Normal (default setting) - Secure For the selected interface, secure data transfer is set in conjunction with handshake RTS/CTS.

Parameter	Meaning
Baud rate	Transfer rate: Transfer rates of up to 115 kbaud can be selected. The baud rate that can be used depends on the connected device, the cable length and the general electrical conditions. • 110 • • 19200 (default setting) • ... • 115200
Archive format	• Punched tape • Binary format (PC format)
RS-232-C settings (details)	
Interface	• COM1 • COM2 only relevant for SINUMERIK Operate on PC
Parity	Parity bits are used for error detection: The parity bits are added to the coded characters to make the number of positions set to "1" an uneven number (uneven parity) or to an even number (even parity). • None (default setting) • Odd • Even
Stop bits	Number of stop bits for asynchronous data transfer. • 1 (default setting) • 2
Data bits	Number of data bits for asynchronous data transfer. • 5 bits • ... • 8 bits (default setting))
XON (hex)	Only for punched tape format
XOFF (hex)	Only for punched tape format
End of data transfer (hex)	Only for punched tape format Stop with end of data transfer character The default setting for the end of data transfer character is (HEX) 1A.
Time monitoring (sec)	Time monitoring For data transfer problems or at the end of data transfer (without end of data transfer character) data transfer is interrupted after the specified number of seconds. The time monitoring is controlled by a time generator (clock) that is started with the first character and is reset with each transferred character. The time monitoring can be set (seconds).

15.8 Backing up setup data

Setting for "Backup setup data"

The "Backup setup data" function should only be used if MD11280 \$MN_WPD_INI_MODE is set to 1 (default setting). Otherwise, when the NC is started for the first time after selecting the part program, a file with the same name with the ".ini" extension would be automatically executed.

Machine data:

MD11280 \$MN_WPD_INI_MODE		
Processing mode of ini files in the workpiece directory		
= 1	When the NC is started for the first time, after the workpiece selection, ini files with the name of the selected part program and the following extensions are executed:	
	CEC	Sag compensation (Cross Error Compensation)
	GUD	User data (Global User Data)
	PRO	Protection zones
	RPA	R parameters
	SEA	Value assignments (Setting Data Active)
	TMA	Magazine data (Tool Magazine Active)
	TOA	Tool offsets (Tool Offset Active)
	UFR	Work offsets (User Frame)

15.9 Network settings

Activate

By default, the network settings cannot be backed up. The archiving of the network settings must be enabled:

1. Copy the "slpmconfig.ini" file from the /siemens/sinumerik/hmi/template/cfg directory.
2. Store the file in the /oem/sinumerik/hmi/cfg or /user/sinumerik/hmi/cfg directory.
3. Open the file and enter the following for [SeriesSetup]:
SupportNetworkSettings=true
4. Restart SINUMERIK Operate.

Note

This setting also affects the creation of a complete standard archive using Ctrl+Alt+S or Ctrl+Alt+C.

Restrictions

- The leases files are not saved:
 - /system/etc/udhcpd-eth0.leases
 - /system/etc/udhcpd-ibn0.leases
- A configuration in the system network must be based on the DNS name rather than IP addresses.

Basic procedure

- Backup of network settings
After activation, the network settings can also be backed up when a commissioning archive is created when a check mark is set, see Creating a start-up archive (Page 278).
- Importing network settings
The saved archives can be imported in the same manner as a commissioning archive, see Reading-in a start-up archive (Page 280).

Note

Importing network settings for replacement components

Disconnect the replacement components from the system network before you import the saved network settings.

NCU

Only the NCU settings are saved with the HMI on the NCU. The following data is backed up:

- /user/system/etc/basesys.ini
- /user/common/tcu (complete directory)

The network settings of an NCU can be backed up and read only from an HMI on the PC/PG. This means the NC address in mmc.ini on the PC/PG is always 192.168.215.1.

PCU

PCU and, if applicable, NCU settings saved with the HMI on the PCU. The following data is backed up:

- PCU
 - /user/system/etc/basesys.ini
 - mmc.ini for SINUMERIK Operate
- NCU (when present as master)
 - /user/system/etc/basesys.ini
 - /user/common/tcu (complete directory)

The following archives are created: Archivename_pcu.arc and Archivename_ncu.arc

Importing an archive for a configuration with NCU as master (DHCP) and PCU (on_low):

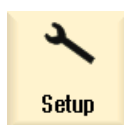
1. Import the archive for Archivename_pcu.arc on a PCU with SINUMERIK Operate.
2. Depending on the replaced parts of the NCU, note the following:
 - If the NCU was not replaced, the system restarts.
 - The NCU has been replaced, Archivename_ncu.arc must be imported via the X127 interface with a separate HMI.

Configuring the network

16.1 Displaying the network overview

All the available interfaces and their network settings are listed in the **Overview of the Network Settings** window. For example, the IP addresses, subnet masks and MAC addresses for the company and system network are displayed here. You have the option of editing the settings.

Procedure



1. Select the "Start-up" operating area.



2. Press the "Menu forward" key.



3. Press the "Network" softkey.



The "Overview of the Network Settings" window opens.



4. Press the "Change" softkey to edit settings.

See also

Settings of the system network (Page 294)

Settings of the factory network (Page 297)

Saving network settings (Page 299)

16.2 Settings of the system network

The configuration for the DHCP server, the routing, active protocols, for DNS servers and other properties of the system network is performed in the **Settings of the System Network** window.

Setting	Description
DHCP activated	Activates or deactivates the DHCP mode for the system network. This automatically assigns IP addresses to the nodes in the system network.
DHCP server synchronization mode	Activates or deactivates the DHCP server synchronization mode. If this setting is active, then the DHCP servers synchronize themselves in the system network (X120) so that only one of them actively assigns addresses. This makes it possible to operate multiple NCUs or PCUs concurrently without having to adjust the network settings. The DHCP servers that are not active go into "standby" mode; in this mode they regularly get the current address data and TCU data from the active server so that if the active server fails a standby server can take over the active role without data loss. • The Master priority setting influences the synchronization so that the server with the "Master" setting is always the active server (it must be active in the network). Deterministically the same control is then always the DHCP server and the current address and TCU data can be found there. Master priority must only be set for a single DHCP server in the system network. • The settings Low priority and High priority can be used to specify which DHCP server takes precedence as the active server. A DHCP server with low priority is then only used as active server when a server with high priority cannot be found in the network. Default: High priority
DHCP address range	Specifies the range for DHCP addresses. The range starts at the address specified at Start and ends at the address entered at End. The resulting range of numbers has the following default setting: Start address = first address in the system network End address = start address + 10 or + 2 (if less than 16 addresses are available)

Setting	Description
Timeout when waiting for master	Specifies a maximum waiting time (in seconds) for the response of the master server (see DHCP server synchronization mode). After this time elapses, the server itself becomes the active server. This additional pause makes it possible for the intended DHCP master to become the active DHCP server without displacement even if it is shortly switched on thereafter, or takes longer to boot than other controls. Default: 120 seconds
Host name	Assigns the DHCP server a fixed host name. In this way, the server can also be addressed via the host name and not only via the IP address. Because the host name is also used as DNS name, it must satisfy the corresponding requirements of the RFC: • ASCII letters (a-z, A-Z), numbers (0-9) and '-' • maximum 63 characters
DNS domain	This is used to specify the Top Level Domain (TLD) name used in the system network. The DNS server of the NCU assigns names to the devices in the system network in this zone. Name requests for all other zones are forwarded to an external name server in the company network. The "local" default setting recommended by the RFC1035 is used for local networks, to avoid conflicts with globally defined domain names. As a rule, this default setting is sufficient.
PN station name	A separate machine name can be defined here for PROFINET purposes. This is used at the ERTEC interface (when available), transferred to the ERTEC system program and also used by DCP at this network interface.
Station name	This string is used for the SINUMERIK-specific OID mcSinumerikMIB.mcSinumerikMiscStation. The value is the name of a station to which the device belongs. Devices with the same station name can therefore be identified as belonging together. This is for information purposes only.
Location	This string is used for the standard OID SNMPv2-MIB::sysLocation. If required, a location can be specified here which can be called with an SNMP client. This is for information purposes only.
Contact person	This string is used for the standard OID SNMPv2-MIB::sysContact. If required, a contact address can be specified here which can be called with an SNMP client. This is for information purposes only.

Setting	Description
Function	This string is used for the Siemens Industry-specific OID automationSystemIdent.automation-FunctionTag. If required, a function designation can be specified here which can be called with an SNMP client. This is for information purposes only.
Fixed domain	Defines an additional DNS domain name (suffix for DNS queries). This is of especial interest when the DHCP server is switched off at the X120 because this cannot set the "local" domain, but it is required for some DNS queries (e.g. for machine control panels).
Fixed DNS servers	Specifies up to three DNS servers that are to be used by the DHCP server. This is of especial interest when the DHCP server is switched off at the X120 because this is also a DNS server or determines DNS servers in the network.
Routing	
Routing X120/X127 -> X130	Specifies whether the routing of packets from the system network (X120) and/or service connection X127 to the company network (X130) is active: - Enabled: Activates the routing for X120 and X127 - Disabled: Deactivates the routing for X120 and X127 - X127: Activates only the routing from X127 to the company network - X120: Activates only the routing from the system network to the company network
Routing X127 -> X120	Activates or deactivates the NAT routing of packets from the service connection X127 to the system network (X120).
Routing X120 -> X127	Activates or deactivates the routing of packets from the system network (X120) to the service connection X127. This is normally prohibited by the firewall. Note that there is no NAT and the sender must ensure that packets to the PG or service PC at X127 with address 192.168.215.x actually reach the NCU to which the device is connected.
Active protocols	Specifies which protocols are to be used in the system network. DCP (Discovery and Basic Configuration Protocol) and LLDP (Link Layer Discovery Protocol) are available for selection. The appropriate active protocol is also used for the service connection X127 at the NCU.

Procedure

	1. The network overview is open.
	2. Press the "System network" softkey. The "Settings of the System Network" window opens.
	3. Press the "Change" softkey to edit settings.
	4. Press the "Reset DHCP server" softkey to return the current settings of the DHCP server to the original settings.
	5. Press the "Distribute DHCP data" softkey to synchronize the settings with all participants in the system network.

Note

You can only use the "Reset DHCP server" and "Distribute DHCP data" softkeys with a connection to the active DHCP server.

16.3 Settings of the factory network

The configuration for the network interface, active protocols and firewall exceptions of the company network are performed in the **Settings of the Factory Network** window.

Setting	Description
Gateway	If this value is not empty, the host specified there is used as the default gateway, which means that all IP packets that cannot be directly assigned are sent here for routing.
DNS server	If DNS name servers are specified here (maximum three), then they must be used to resolve symbolic host names, i.e. at most of the points where an IP address is expected, a computer name can also be used instead. The name server setting is also sent to your DHCP client (TCU, PG) via the DHCP server of the NCU, so that this can also work with symbolic names.
Time server	Here you can specify up to three NTP servers (UDP/123) that can be used by the NTPD on the NCU for time synchronization.

Setting	Description
Host name	Here you can define a name for the local host. This manually assigned name takes priority over all other names. The host name is determined in the following order: • This entry (identical to the host name from the basesys.ini file) • A name received from the DHCP server (reverse DNS lookup, i.e. which name belongs to the received IP address) • A default name ("NONAME_...") Because the host name is also used as DNS name, it must satisfy the corresponding requirements of the RFC: • ASCII letters (a-z, A-Z), numbers (0-9) and '-' • maximum 63 characters
Active protocols	Specifies which protocols are to be used in the company network. DCP (Discovery and Basic Configuration Protocol) and LLDP (Link Layer Discovery Protocol) are available for selection.
Firewall exceptions	You can specify ports here that are to be enabled in the external network interface. Port TCP/22 (SSH protocol) is always enabled.
• S7 communication (TCP/102)	Port for the S7 communication
• VNC access (TCP/5900)	Port for the VNC access
• Additional ports	Any further ports can be enabled here. The entry starts with the protocol name (TCP or UDP), followed by a slash '/' and the port number of the service to be enabled. Several entries are separated by a space. Example (enabling of the VNC servers): TCP/5900 TCP/5904 TCP/5905

NOTICE**Security risk**

Note that opening the ports in the firewall can represent a security risk. Only enable the ports that you actually require.

Note that every accessible service can have security gaps.

Procedure

	1. The network overview is open.
	2. Press the "Company network" softkey. The "Settings of the Company Network" window opens.
	3. Press the "Change" softkey to edit settings.

16.4 Saving network settings

The settings made in the NCU can be saved as a file.

A target folder in which the network configuration is to be saved, or an existing file that is to be overwritten with the new settings, can be selected in the **Save Network Settings: Select Archive** window.

Procedure

	1. The network overview is open and you have made network settings.
	2. Press the "Save NCU settings" softkey. The "Save Network Settings: Select Archive" window opens.
	3. Select the folder in which the network configuration is to be stored. If a file is already available, it will be overwritten. - OR - Press the "New directory" softkey to create a new directory. - OR - Press the "Find" softkey, for example, to find the storage location of the network configuration file.
	
	4. Press the "Find next" softkey if the file found during the search does not correspond to the required configuration file.
	5. Press the "Cancel" softkey to reject the settings. - OR - Press the "OK" softkey to save the network configuration.
	

16.5 Configuring operator panels

16.5.1 Displaying the available operator panels

A list with all the available operator panels is displayed in the upper part of the **Available Operator Panels ...** window. One line corresponds to one operator panel and contains the following information:

Column	Description
	Consecutive number to identify the operator panel
Status	Status of the operator panel:  : The operator panel is available on the bus.  : The operator panel is not available on the bus.
HW	Information on the hardware of the operator panel Possible values are unknown (no information available), NCU, PCU, PPU, TCU (all TCU including HT 8, etc.). Default value: Unknown
Name	Name of the operator panel
MCP	DIP switch position for the machine control panel The values can be freely selected from 1 to 254 (0 means that the last active machine control panel is retained, 255 means that a machine control panel is not being used). Default value: 192
DCK	Availability of direct control keys Active: The TCU has direct control keys Inactive: The TCU has no direct control keys
TCU	Index for the TCU (Thin Client Unit) The value is freely configurable from 0 to 255 and is used for identification in the PLC.
EKS	DIP switch position for the chip reader for access levels (Electronic Key System) Possible values range from 1 to 255. The value 0 means that an EKS is not used.
Virtual key-board	Use of a virtual keyboard Possible values are "Never", "Always" and "Auto", whereby "Auto" is set as default value.

Note

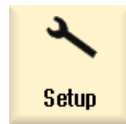
The information of the operator panels is read-only and cannot be changed in the **OPs** window.

Update of the overview

When switching to the **OPs** window, a search for available operator controls is performed automatically. This is also possible later via the "New display" or "Reload OP data" softkeys. The latter action results in a completely new scan during which already available operator controls are deleted from the local cache. New operator controls and data are sought during a reload.

The active DHCP server that contains the data for the operator panels can be a different NCU, but also the local NCU itself. Irrespective of this, a copy of the data is stored and used on the local NCU.

Procedure



1. Select the "Start-up" operating area.



2. Press the "OPs" softkey.



The "OPs" window is opened.

All the presently available operating elements are displayed.



3. Press the "New display" softkey to refresh the overview.
A search is performed for new operator controls and data.



4. Press the "Reload OP data" softkey to start a completely new search.
Already available operator controls are deleted from the buffer during this search.



5. Press the "Distribute TCU data" softkey to synchronize the available controls with all participants in the system network.

Note

You can only use the "Distribute TCU data" softkey with a connection to the active DHCP server.

16.5.2 Configuring VNC connections

A list of the available VNC connections is displayed in the lower part of the "Available Operator Panels ..." window. It contains the following information:

Column	Description
	Consecutive number to identify the VNC connection
VNC server	IP address or name of the NC
Displacement	Specifies whether it is possible that the TCU will be displaced by other nodes.
/Priority	Displacement priority: Specifies the priority which influences whether the TCU will be displaced by other nodes. The possible values lie between 0 and 10. The smaller the value, the faster the TCU will be displaced. If the displacement is blocked, the priority will be ignored.
Start	Specifies whether the TCU is to be started active, i.e. as operator panel.
/Priority	Start-up priority: Specifies the priority of the TCU connection to be used. Any value greater than 0 is permitted. The larger the value, the lower the connection priority.

Column	Description
Display area	Specifies which HMI screen is displayed when activating the TCU.
Channel	Specifies which channel is displayed when activating the TCU.

Editing the list of VNC connections

	1. The "Available Operator Panels ..." window is open.
	2. Press the "Change" softkey. The information on the VNC connections can be edited.
	- OR -
	3. Press the "Add" softkey to establish a new VNC connection.
	- OR -
	4. Press the "Remove" softkey to remove a selected VNC connection from the list.
	- OR -
	5. Press the "Move up" or "Move down" softkey to move a VNC connection within the list.
	
	6. Make the required change and press the "OK" softkey.
	- OR -
	Press the "Cancel" softkey to cancel the changes.

See also

Displaying the available operator panels (Page 300)

16.5.3 T:M:N configuration

By T:M:N configuration we mean a grouping of t operating stations (TCUs with MCP), m HMI systems and n NCK systems, in which the operating stations can be freely switched to the various HMI systems in order to operate. In a two-stage process, the switchover targets for each operating station are individually configured in a channel menu.

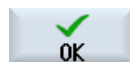
From the point of view of the HMI, T:M:N operation is active if a channel menu is configured. Available switchover targets are defined directly via the address of the HMI and optionally via an NC channel, which must be assumed on switchover from the HMI. These switchover targets can be combined into so-called channel groups. Channel groups are displayed on horizontal

softkeys in the channel menu, the switchover targets themselves (HMI, NC channel) are displayed on the vertical softkeys assigned to a channel group.

Linking of softkeys to channel groups

A list for the linking of the HMI softkeys to configured channel groups is displayed in the lower part of the window. The channel groups can in turn contain switchover targets that execute functions of the operator panel, for example. Up to 32 softkeys (SK_1 to SK_32) can be linked to a specific channel group (e.g. CH_GRP_0).

Procedure



1. The "Available Operator Panels ..." window is open.
 2. Select the required operator panel and press the "T:M:N" softkey
A list for the linking of the HMI softkeys to configured channel groups is displayed in the lower part of the window.
 3. Press the "Change" softkey.
The list can be edited.
 4. Make the required change and press the "OK" softkey.
- OR -
- Press the "Cancel" softkey to cancel the changes.

See also

Displaying the available operator panels (Page 300)

16.6 Station-related network diagnostics

Using the station-related network diagnostics, you have the possibility of identifying faulty or incorrectly parameterized components or those components that have not been connected.

You obtain an overview of all of the components belonging to the system network:

- NCU
- PCU
- TCU
- MCP
- EKS

If components are either faulty or absent (or unavailable), they will be marked as such in this overview.

Error analysis

You have the option of initiating a fault analysis for the components involved. You obtain the result of a such a diagnostics in the form of a fault message. The possible fault states and causes are described in this fault message together with how to resolve the fault.

16.6.1 Displaying network adapters

The actual network adapters (system network X120, company network X130 or system network ETH2 and company network ETH1) and their availability are displayed in a tree-like structure in the "TCP/IP diagnostics" window.

Displaying the components

The following components are displayed in the window:

- The control is displayed at the topmost level.
- System network (X120 / ETH2) with configured operator stations with IP address
 - Operator panels with IP address
 - Machine control panels
 - EKS
- Company network (X130 / ETH1) with
 - Operator panels with IP address
 - Machine control panels
 - EKS

Components that cannot be accessed

 If a component cannot be accessed, then it is marked using this symbol.

Input rights

Operator panels, which have input rights, have a green background.

Detailed view

You have the option to display the following information about a selected component.

TCU

- IP address
- SW version
- MCP index from the configuration
- TCU index from the configuration
- DNS name
- Resolution

MCP

- IP address
- DNS name
- MCP index from the configuration
- MCP index requested from HMI
- MCP index requested from PLC

Procedure

1. Select the "Diagnostics" operating area.



2. Press the menu forward key and the "Bus TCP/IP" softkey.



3. Press the "TCP/IP diagnostics" softkey.
The "TCP/IP Diagnostics" window opens and displays the current availability of the network connections.
4. Position the cursor on the component for which you wish to display the detailed information.



5. Press the "Details" softkey to display all available parameters of the configured network connections.



6. Press the "Details" softkey to hide the detailed display.



7. Press the "Return" softkey in order to exit the "TCP/IP diagnostics" window and return to the "PROFIBUS diagnostics" window.

16.6.2 Error analysis**16.6.2.1 Performing error diagnostics****Precondition**

Error analysis only functions for a PCU.

Procedure



1. The "TCP/IP Diagnostics" window opens.
2. Position the cursor on the component that is marked as having an error (NCU).
3. Press the "Error analysis" softkey.
The corresponding error message is displayed.

16.6.2.2 MCP/EKS as network participant cannot be accessed

Error	Possible error causes and remedies
The MCP/EKS cannot be accessed as network participant.	
	Error cause 1
	There is no physical network connection. Notice: The display only refers to the circumstance that the direct connection from MCP/EKS to the next connection nodes has not been established, i.e. • The Ethernet cable is not directly inserted at the MCP/EKS, • The Ethernet cable is not inserted at the direct connection partner or the partner is not available (e.g. MCP), • The Ethernet cable connected to the MCP/EKS is defective.
	Remedy
	Establish the physical network connection (check the plug-in connections, replace the cable, check the connection partner).
	Error cause 2
	The physical network connection does not exist after the next connection node to the system network. There is no physical connection from the MCP/EKS to the computer with the active DHCP server. The cables are either defective, have not been inserted or switches have not been closed. Switches can also be located on the MCP or distribution box.
	Remedy
	Establish the physical network connection (check the plug-in connections, replace the cable, check the connection partner). Once the connection has been established, the TCU continues to boot.
The MCP/EKS cannot be accessed as network participant. There is a physical network connection to the system network.	
	Cause of the error
	No active DHCP server is found in the system network. Possible causes: • Only a single computer has been configured in the system network with active DHCP server - and this computer is not available. The system network is not operated with synchronizing DHCP servers. • No computer has been configured in the system network with active DHCP server. DHCP operation is disabled on all potential computers (NCU, PCU). This is a configuration error. • In the system network, there is no NCU or PCU computer available that has DHCP operation activated on it and could be a server.
	Remedy
	In the system network, configure a computer that is available as DHCP server.

16.6.2.3 HMI on the PCU cannot establish a network connection to the NC

Error	Error cause 1	Remedy
HMI displays alarm 120202 "Wait for connection to NC/ PLC" and/or no values (only '#').	The IP address of the target NCK/PLC set on the PCU is incorrect.	From the HMI, correct the target address for NCK/PLC and restart the PCU.

Error	Error cause 2.1	Remedy
HMI displays alarm 120202 "Wait for connection to NC/ PLC" and/or no values (only '#').	The PCU has an IP address that does not belong to the network or has an invalid IP address. The TCU support is deactivated on the PCU (= DHCP is disabled) and the PCU either operates with a permanently set IP address that does not belong to the network, or with the standard Windows DHCP client to dynamically retrieve addresses. This is a configuration error.	Correctly set the network settings for the system network in the Windows service mode (set the fixed specified IP address and subnet mask for the system network, use the addressing mode, fixed IP address). Execute the following actions using the "System Network Center" program: • Connect the DHCP service to the interface for the system network • Set the DHCP mode (TCU support, DHCP SyncMode, address range of the addresses that can be assigned) • Initiate DHCP clear

Error	Error cause 2.2	Remedy
HMI displays alarm 120202 "Wait for connection to NC/ PLC" and/or no values (only '#').	The PCU has an IP address that does not belong to the network or has an invalid IP address. The TCU support is activated on the PCU (= DHCP is enabled) and the PCU is active DHCP-/DNS server (either as a result of the DHCP SyncMode ON_MASTER or as a result of the Sync priority - just the same as the other server candidates and random selection) - or the PCU is a DHCP client. As a result of a software error or the actual network configuration, the PCU cannot obtain a valid IP address (e.g. '0 address').	Correctly set the network settings for the system network in the Windows service mode (set the fixed specified IP address and subnet mask for the system network, use the addressing mode, fixed IP address). Execute the following actions using the "System Network Center" program: • Connect the DHCP service to the interface for the system network • Set the DHCP mode (TCU support, DHCP SyncMode, address range of the addresses that can be assigned) • Initiate DHCP clear In addition, at the station with DHCP master (DHCP SyncMode = ON_MASTER) a DHCP clear should be initiated.

Error	Error cause 2.3	Remedy
HMI displays alarm 120202 "Wait for connection to NC/PLC" and/or no values (only '#').	The PCU has an IP address that does not belong to the network or has an invalid IP address. The TCU support is activated on the PCU (= DHCP is enabled) and the PCU is active DHCP-/DNS server (either as a result of the DHCP SyncMode ON_MASTER or as a result of the Sync priority - just the same as the other server candidates and random selection). The PCU is set with an incorrect IP address and an incorrect IP range for the IP addresses that can be dynamically allocated. The PCU possibly changed from DHCP standby to active server as the computer with DHCP MASTER has failed or been switched-off.	Correctly set the network settings for the system network in the Windows service mode (set the fixed specified IP address and subnet mask for the system network, use the addressing mode, fixed IP address). Execute the following actions using the "System Network Center" program: • Connect the DHCP service to the interface for the system network • Set the DHCP mode (TCU support, DHCP SyncMode, address range of the addresses that can be assigned) • Initiate DHCP clear In addition, at the station with DHCP master (DHCP SyncMode = ON_MASTER) a DHCP clear should be initiated.

Error	Error cause 2.4	Remedy
HMI displays alarm 120202 "Wait for connection to NC/PLC" and/or no values (only '#').	The PCU has an IP address that does not belong to the network or has an invalid IP address. TCU boot support is activated on the PCU (= DHCP is enabled) and the PCU is the standby DHCP/DNS server (i.e. in the role of the DHCP client). An additional active DHCP server was switched into the system network while operational; this active DHCP server also has an incorrect address range. This is a configuration error.	Correctly set the network settings for the system network in the Windows service mode (set the fixed specified IP address and subnet mask for the system network, use the addressing mode, fixed IP address). Execute the following actions using the "System Network Center" program: • Connect the DHCP service to the interface for the system network • Set the DHCP mode (TCU support, DHCP SyncMode, address range of the addresses that can be assigned) • Initiate DHCP clear In addition, at the station with DHCP master (DHCP SyncMode = ON_MASTER) a DHCP clear should be initiated. Switch the complete system off and on again.

Error	Possible error causes and remedies
HMI displays alarm 120202 "Wait for connection to NC/PLC" and/or no values (only '#').	
	Error cause 3.1
	The NCU has an IP address that does not match the network or has an invalid IP address. The NCU does not participate in DHCP operation and runs with a permanently set IP address that is incorrect and does not match the system network.
	Remedy 3.1

Error	Possible error causes and remedies
	Connect the service PG to the NCU via X127. Log into Linux via WinSCP or Putty. Using the service command 'sc show ip -eth0', display the actual settings - and check these. If the cause of the error is confirmed, then the NCU must be reparameterized. Perform the service commands and settings: • sc enable DHCPSPvr -X120 or the equivalent sc enable DHCPSPvr -eth0 to activate DHCP operation • sc enable DHCPSSync -X120 -LOW HIGH MASTER or equivalent sc enable DHCPSSync -eth0 -LOW HIGH MASTER to set the DHCP synchronization mode of this NCU Enter the following entries correctly in the /user/system/etc/basesys.ini file <pre>:[InternalInterface] ; With InternalIP and InternalNetMask (both must be set together), ; you can change the address on the internal/TCU/automation/system net. InternalIP=192.168.214.1 InternalNetMask=255.255.255.0 ... ; This is the range of dynamic IPs given out by the DHCP server ; Defaults should be sensible InternalDynRangeStart=192.168.214.10 InternalDynRangeEnd=192.168.214.239</pre> The values in green should be set corresponding to the overall system. Then restart the NCU.
	Error cause 3.2
	The NCU has an IP address that does not match the network or has an invalid IP address. The NCU participates in DHCP operation and the NCU is an active DHCP/DNS server (either as a result of the DHCP SyncMode ON_MASTER or as a result of the Sync priority - just the same as the other server candidates and random selection). The NCU is set with an incorrect IP address and an incorrect IP range for the IP addresses that can be dynamically allocated. The NCU possibly changed from DHCP standby to active server as the computer with DHCP MASTER has failed/has been switched-off.
	Remedy 3.2
	The same as for remedy 3.1
	Error cause 3.3
	The NCU has an IP address that does not match the network or has an invalid IP address. The NCU participates in DHCP operation and operates with the incorrect IP address. After the NCU booted, an additional component that is already operational (NCU or PCU) with active DHCP server was connected to the network. An additional active DHCP server was switched into the system network while operational; this active DHCP server has an incorrect address range. This is a configuration error.
	Remedy 3.3
	The same as for remedy 3.1
HMI displays alarm 120202 "Wait for connection to NC/PLC" and/or no values (only '#').	
	Error cause 4.1
	The NCU has a valid IP address in the system network, however, not the required IP address. The NCU does not participate in DHCP operation and operates with a permanently set target address that is different from that saved in MMC.INI in the system network - however, this is a valid address. This is a configuration error.
	Remedy 4.1

Error	Possible error causes and remedies
	Connect the service PG to the NCU via X127. Log into Linux via WinSCP or Putty. If the set NCU mode (no DHCP) is to be kept, the fixed IP address of the NCU must be changed. Correctly enter the following entries into the file /user/system/etc/basesys.ini <pre>[InternalInterface] ; With InternalIP and InternalNetMask (both must be set together), ; you can change the address on the internal/TCU/automation/system net. InternalIP=192.168.214.1 InternalNetMask=255.255.255.0</pre> Then restart the NCU.
	Error cause 4.2
	The NCU has a valid IP address in the system network, however, not the required IP address. The NCU does not participate in DHCP operation and operates with a permanently set target address that is different from that saved in MMC.INI in the system network - however, this is a valid address. This is a configuration error.
	Remedy 4.2
	The same as for remedy 4.1
	Error cause 4.3
	The NCU has a valid IP address in the system network, however, not the required IP address. The NCU participates in DHCP operation and was not able to obtain the corresponding required IP with the target address saved in the MMC.INI; this is the reason that it was dynamically allocated an address. A reason for this could be that the required IP was configured twice in the system network, or the required IP is not in the reserved range, but is located in the dynamic address range. This is a configuration error.
	Remedy 4.3
	The same as for remedy 4.1
	Error cause 4.4
	The NCU has a valid IP address in the system network, however, not the required IP address. The NCU participates in DHCP operation as DHCP client and has a required IP that has been set but which does not match the system network. For instance, this situation occurs if a network address was used in the system network other than the default network address, but it was forgotten to change over the required IP of the NCU to this address range. The NCU is allocated an IP address that matches the system network from the dynamically allocated range, but which is different from its 'old' required IP that was not changed over. Up to and including SW 2.4.1, in this situation, the NCU cannot be addressed in the system network. The reason for this is that it has internally established its firewall regarding the required IP that was entered for it, which does not match the system network - which is why it does not respond at X120.
	Remedy 4.4
	The same as for remedy 4.1
HMI displays alarm 120202 "Wait for connection to NC/PLC" and/or no values (only '#').	
	Error cause 5.1

Error	Possible error causes and remedies
	There is no physical connection. Possible error causes include: • A switch is not closed/has failed; this can also be the switch of an MCP • The cable connection has become loose (poor contact) • An incorrect cable was used (crossover/not crossover) • A cable being used as data link is defective
	Remedy 5.1
	Establish a physical connection.

16.6.2.4 TCU cannot establish a network connection to the HMI

Error	Possible error causes and remedies
When booting, TCU signals 'ERROR: Network connection not established'.	
	Cause of the error
	There is no physical connection. Notice: The display only refers to the situation where the direct connection of the TCU to the next connection nodes has not been established, i.e. • The Ethernet cable is not directly inserted at the TCU, • The Ethernet cable is not inserted at the direct connection partner or the partner is not available (e.g. MCP), • The Ethernet cable at the TCU is defective.
	Remedy
	Establish the physical network connection (check the plug-in connections, replace the cable, check the connection partner). Once the connection has been established, the TCU continues to boot.
When the TCU boots, it signals 'connection established – no DHCP Server available'.	
	Cause of the error
	The physical network connection does not exist after the next connection node to the system network. There is no physical connection from the TCU to the computer with the active DHCP server. The cables are either defective, have not been inserted or switches have not been closed. Switches can also be located on the MCP or distribution box.
	Remedy
	Establish the physical network connection (check the plug-in connections, replace the cable, check the connection partner). Once the connection has been established, the TCU continues to boot.
When the TCU boots, it signals 'connection established – no DHCP Server available'. There is a physical network connection to the system network.	
	Cause of the error

Error	Possible error causes and remedies
	No active DHCP server is found in the system network. Possible error causes include: • Only a single computer has been configured in the system network with active DHCP server - and this computer is not available. The system network is not operated with synchronizing DHCP servers. • No computer has been configured in the system network with active DHCP server. DHCP operation is disabled on all potential computers (NCU, PCU). This is a configuration error. • In the system network, there is no NCU or PCU computer available that has DHCP operation activated on it and could be a server.
	Remedy
	In the system network, configure a computer that is available as DHCP server.
After it has booted, the TCU displays 'Waiting for HMI'. (The TCU cannot establish its connection to the HMI.)	
	Error cause 1
	The connection to the HMI has not been established - neither the connection configured in config.ini nor the default connection. An incorrect address for the HMI connection after booting has been set in the config.ini file.
	Remedy
	Correct the configuration using the "System Network Center" program. The TCU automatically reboots.
	Error cause 2
	The connection to the HMI has not been established - neither the connection configured in config.ini nor the default connection. The computer, to which the TCU should connect after booting according to the configuration in config.ini, cannot be reached because the computer is not switched on/is not available.
	Remedy
	Switch off the computer (NCU or PCU) and switch on again.
	Error cause 3
	The connection to the HMI has not been established - neither the connection configured in config.ini nor the default connection. The HMI on the computer, to which the TCU should connect after booting according to the configuration in config.ini, cannot be reached because the HMI is deactivated.
	Remedy
	Activate the HMI on the NCU using the service command <code>sc enable HMI</code> .
	Error cause 4
	The connection to the HMI has not been established - neither the connection configured in config.ini nor the default connection. The HMI on the computer, to which the TCU should connect after booting according to the configuration in config.ini, cannot be reached because the HMI has either not start or has crashed due to a software error.
	Remedy
	Switch off the computer (NCU or PCU) and switch on again.

16.6.2.5 The TCU cannot establish a network connection to the assigned PCU

Error	Possible error causes and remedies
The PCU does not have the IP address specified in the configuration, but an address from the IP range of the system network that can be dynamically allocated. In this error case, the TCU cannot reach its configured PCU for display.	
	Error cause 1
	IP address assigned twice The PCU has a default IP address which is also configured for another NCU/PCU. This means that an IP address has been assigned twice and therefore there is a system configuration error. The PCU is not an active DHCP server (otherwise, it would have the required IP), however, it is operated in the synchronizing DHCP mode. As the address requested from the PCU has already been assigned, the PCU is allocated an address from the dynamic address range via DHCP.
	Remedy
	A decision has to be made which computer should keep the IP address that has been assigned twice. If it is the PCU, then • on the PCU, under network settings, this address should be again set as fixed IP address in the system network, • at the NCU/PCU which runs in the DHCP synchronization mode ON_MASTER, a 'Clear DHCP' should be executed (for an NCU sc command <code>sc clear dhcp -X120</code>, for a PCU using the "System Network Center" program). • at the computer, which uses the IP address assigned twice as second computer, the IP address should be re-configured. Notice: If this is a PCU, then it should be linked to this new address using the "System Network Center" program (this is the first screen in the program). If the PCU is to change its address, then on the PCU • under network settings, the new address should be set as fixed IP address in the system network, • linked to this new address using the "System Network Center" program (first screen in the program) • at the NCU/PCU, which runs in the DHCP synchronization mode ON_MASTER, a 'Clear DHCP' should be executed (for an NCU using the sc command, for a PCU, using the "System Network Center" program). Restart the system.
	Error cause 2
	This was preceded by replacing the PCU hardware. As a result of the hardware replacement, in spite of the same settings of the new PCU, the requested (set) IP address is still internally reserved for the previous hardware as this is an inherent feature of the system. The PCU is either active or non-active DHCP server, however, is operated in the synchronizing DHCP mode. As the address requested from the PCU is still reserved, the PCU is allocated an address from the dynamic address range via DHCP.
	Remedy

Error	Possible error causes and remedies
	Perform the following steps: • Under network settings, on the PCU, set the required address again as fixed IP address in the system network. • At the NCU/PCU, which runs in the DHCP synchronization mode ON_MASTER or is active DHCP server, execute a 'Clear DHCP' (for an NCU using the sc command <code>sc clear dhcp – X120</code>, for a PCU, using the "System Network Center" program). • Restart the system.
	Error cause 3
	An address that does not belong to the network has been configured for the PCU. An address has been configured for the PCU, which does not match the network address in the system network, and is therefore dynamically assigned an IP address from the active server via DHCP. The PCU is not an active DHCP server, but is operated in the synchronizing DHCP mode.
	Remedy
	Perform the following steps: • Under network settings, on the PCU, set the required address again as fixed IP address in the system network. • On the PCU, using the "System Network Center" program, link the DHCP service to this new address (first screen in the "System Network Center" program), if required, adapt the other DHCP settings (dynamic address range). • At the NCU/PCU, which runs in the DHCP synchronization mode ON_MASTER or is active DHCP server, execute a 'Clear DHCP' (for an NCU using the sc command <code>sc clear dhcp – X120</code>, for a PCU, using the "System Network Center" program). • Restart the system.

16.6.3 Configuring TCP/IP diagnostics

The configuration of the network adapter is displayed in the "TCP/IP Configuration" window.

System network

On the system network, process data communication and image transmission of the components are executed with the operating software for the display units (TCUs).

A TCU and NCU are connected to the system network via Ethernet interface X120.

A PCU is connected via the ETH2 Ethernet adapter.

Company network

The company network is used, for example, to access the network drives.

An NCU is connected to the company network via Ethernet interface X130.

A PCU is connected via the ETH1 Ethernet adapter.

Availability of the network connections

Network adapter connection



White

Network cable inserted



Red

Network cable not inserted

Availability

The availability describes the percentage of faulty packages compared to all the sent and received packages.

Problems in the company network (e.g. logical drives that cannot be accessed, double IP address, etc.) as well as settling time during power up can result in fluctuations in availability.



Green

Greater than 95%



Yellow

50 - 95%



Red

Less than 50%

Details of the network connections

- **Computer name**
- **DNS name**
Computer name of the controller as stored at the DNS server (**D**omain **N**ame **S**ervice).
- **MAC address**
Physical address of the network adapter

- **Address type**

Information about the configuration of the network adapter in the configuration file "basesys.ini":

- **DHCP:** DHCP is active for this network adapter.
The information behind the hyphen "-" specifies the DHCP mode of operation:
off - DHCP is deactivated for this adapter
Client - a DHCP client that receives an IP address and further data from a server, is running on the interface
Server - the NCU provides a DHCP server on this interface that supplies clients on this network with IPs
Synced Server - a protocol is active with which several NCUs synchronize with each other. This ensures that always only one of these NCUs functions as DHCP server and, for example, no double IP addresses are assigned.
If a network adapter operates as "Synced Server", the "State of synch. DHCP server" line displays additional information.
If no changes have been made for the network adapters in the configuration file, then "Default" is also output.
- **Manual**
The settings IP address, subnet mask, DNS server 1, DNS server 2 and gateway are configured in the "basesys.ini".
Note:
In the "Change" mode, either "Manual" or "DHCP" can be selected (only for company network, X130).

- **IP address assigned**

Current IP address of the network adapter.

- **Subnet mask assigned**

- **DHCP server**

IP address of the network adapter (for address type "DHCP").

- **State, DHCP server**

State of the synch. DHCP server:

- Active (actual NCU is active in the group of all NCUs in the "Synced Server" mode and transfers the IP addresses);
- Standby (NCU is not active, if the active server fails, this NCU could take over the task of the server).

- **Mode, DHCP synchron.**

Priority of the synch. DHCP server:

Low, High, Master

- **DNS server 1, DNS server 2**

Domain Name Server

- **Gateway**

- **Note**

All information that is not available is marked in the relevant table line with a hyphen "-".

Modifiable parameters

The following parameters of the company network adapter (X130) can be set by selecting the "Address type":

- Address type
- IP address
- Subnet mask
- DNS server 1 and 2
- Gateway (applies for system and company network)

Address types

- DHCP
The values for the settable parameters IP address, subnet mask, DNS server 1, DNS server 2 and gateway are defined automatically by the DHCP server, in accordance with the entry in the "basesys.ini".
- Manual
The values for the settable parameters IP address, subnet mask, DNS server 1, DNS server 2 and gateway can be specified manually; these are then stored in the "basesys.ini".

References

For additional information about network configuration, please refer to the following documentation:

SINUMERIK 840D sl, Operator Components and Networking (IM5)

Procedure

1. Select the "Diagnostics" operating area.



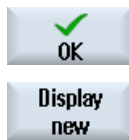
2. Press the menu forward key and the "Bus TCP/IP" softkey.



3. Press the "TCP/IP diagnostics" and TCP/IP Config." softkeys. The "TCP/IP Configuration" window opens.



4. Press the "Modify" softkey to change specific parameters. The fields that can be edited have a white background.



5. Enter the desired changes and press the "OK" softkey to confirm the entries.
6. Press the "New display" softkey to refresh the display.

16.6.4 Ethernet network diagnostics

16.6.4.1 Diagnostics of the network and DHCP server

In order to more effectively influence the commissioning, press the "Network diagnostics" softkey to easily represent the data and availability of each network node on the Ethernet. The results display of this scan process delivers, among other things, the following information:

- Component identification of the hardware
- TCP/IP address / MAC address
- DHCP server detection
- Status of the nodes

This gives you the option to test whether the current topology matches the reference topology.

Procedure



1. Select the "Diagnostics" operating area.



2. Press the menu forward key and the "Bus TCP/IP" softkey.



3. Press the "TCP/IP diagnostics" softkey.
The "TCP/IP Diagnostics" window opens and displays the current availability of the network connections.



4. Press the "Network diagnostics" softkey to browse your networks.
The "Network Status" window opens and displays the addresses of the HMI (X120) system network, the PN (X150) system network and the (X130) company network.
The corresponding fields in the window remain empty for non-accessible networks.



5. Press the "New display" softkey to refresh the display.

- OR -



Press the "DHCP server" softkey to fetch information and query the status of the DHCP server.

The "DHCP Server" window opens.

For each active DHCP server, the status, the role (priority), the MAC address, the TCU version and the lease version are displayed in tabular form in the upper part of the window. The required IP with the associated address area appears in the lower part of the window.

Any differing data is displayed in red.



6. Press the "Address list" softkey to acquire all information about the active DHCP server.

The "Address list" window is opened.

The IP address, the client ID in the form of the MAC address or the CF card number, the lease version, the device number, the DNS name and additional information are displayed for each node in a tabular form.

Inactive nodes are grayed out.

16.6.4.2 Messages

All the system-relevant state and error messages from the "syslog" log file are displayed via the "Messages" softkey.

Procedure



1. The "Network Status" window opens.



2. Press the "Messages" softkey.
The "Messages" window opens.



3. Press the "Show all", "DHCP", "MCP" or "TCU" softkey to filter the display.



16.6.4.3 Network trace

If network problems occur, you can record the data traffic as a log file for analysis and evaluation via the "Network Trace" window.

General information

- The function is protected with the standard password NETWORK

Note

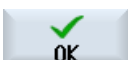
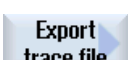
Change the standard password to prevent improper use.

- The log is recorded via the tcpdump program (Linux) or the windump program (Windows). Further information can be found at www.tcpdump.org

Requirement

If you want to change the password, you require access level 1 (manufacturer).

Procedure

	1. The "Network Status" window is open.
	2. Press the "Network trace" softkey. The "Set Password" window opens.
	3. Enter the password and press the "OK" softkey to confirm the entry. The "Network Trace" window opens.
	4. If you have stored parameters in a file, press the "From file" softkey in order, for example, to read-in the file from a USB. The "Select File" window opens.
	5. Select the file from a directory and confirm with "OK".
	- OR -
	4. If required, parameterize the recording conditions in the "Parameters (tcpdump)" input field. 5. Press the "Start" softkey to record the log. Note: The recording is stopped automatically after 20 minutes.
	6. Press the "Stop" softkey to complete the recording. The "Export Trace File" window opens.
	7. Select a storage location, assign a file name and confirm with "OK". The trace file is moved to the storage location. With "Cancel", the log remains in the following directory: HMI data/Logs/Network trace/tcpdump.pcap or windump.pcap
	8. If the log already exists, press the "Export trace file" softkey to save the current log file, for example, on a USB. The "Select Storage Location" window opens.
	9. Select a storage location, assign a file name and confirm with "OK".

Change password



1. The "Network Trace" window is open.
2. Press the "Change password" softkey.
The "Change Password" window opens.
Enter a new password and confirm with "OK".

Trace parameters

You can parameterize the conditions for the log recording via the "Parameters (tcpdump)" input field in the Network Trace window.

Syntax

<switch> <option>

<switch> parameters

Value	Meaning
-D	Displays a list of all network interfaces. All other <Switch> parameters are ignored.
-i <interface>	Defines which network interface has to be checked.
-n	Host names are not resolved.
-c <number>	The recording is stopped after a defined number of packets.
-C <file size>	Limits the file size. When the file size is reached, a new file with name and consecutive number, starting with 1, is created. The unit of the file size corresponds to million bytes.
-e	Outputs the Ethernet header.
-q	Creates an abbreviated log version.
-vv	Creates more detailed packet information.
-S	Changes the packet sequence number from absolute to relative.
-A	Outputs the contents of a packet as ASCII.
-xx	Outputs the contents of a packet as Hex.
-XX	Outputs the contents of a packet as Hex and as ASCII.
-s <bytes>	Specifies the packet size in bytes that has to be recorded per packet. The default is 68 bytes.

Note

The value `-w` is always implicitly contained in the call for the creation of the standard log file:
HMI data/Logs/Network trace/tcpdump.pcap or windump.pcap

Parameter <option>

Option	Value	Meaning
host	IP address or host name	Only those packets are recorded that have the specified IP address or host name as source or as destination.
net	Network range	Only those packets are recorded that have an IP address or host name from the specified network range as source or as destination.
port	A number from 0 to 65535	Records packets that have the specified port, e.g. 5900, on the source or on the destination side.
portrange	A range between 0 and 65535	Records packets that have a port in the specified range on the source or on the destination side.
src	host, net, port or portrange	Only those packets are recorded that have the specified value as source.
dst	host, net, port or portrange	Only those packets are recorded that have the specified value as destination.
proto	ether, fddi, tr, wlan, ip, ip6, arp, rarp, dec-net, tcp and udp	Network protocol in which packets can be recorded.

You can link several options using the following expressions:

and, or, not

You can limit extended expressions using the following characters:

() and '

Reference

Further parameter assignments can be found at www.tcpdump.org

Examples

```
-n port 22 ; Records whether packets arrive at port 22 of the local system.

-i eth0 port 4080 ; Records all data for port 4080 arriving at the eth0 interface.

-i 2 ; Records all data arriving at the interface with adapter number 2.

-n host 10.113.20.0 ; Records the network traffic from system 10.113.20.0, whereby the IP address is not changed to the host name.
```

```

192.168.214.4 and not net 10.113.20.0 ; Records the entire network
                                     traffic of system
192.168.214.4 which does not
run via system 10.113.20.0.

-i 10.113.20.0 -q '(tcp port 80) or   ; Records HTTP and also HTTPS
(tcp port 443) '                     packets of the network
                                     interface from system
                                     10.113.20.0.

```

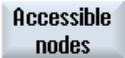
16.6.4.4 Accessible nodes

By actuating the "Accessible nodes" softkey in the network, the list of accessible nodes on X120 and, if necessary, on X150, is created and displayed. The list content can be compared as current topology with a reference topology.

Note

Although handheld terminals and EKS identification systems are not determined for the list of accessible nodes, they can be added as additional nodes to the topology.

Procedure

	1. The "Network Status" window opens.
	2. Press the "Accessible nodes" softkey. The "Accessible Nodes" window opens. The IP address, HW-Id/type, status and location are displayed in tabular form for each node.
	3. Press the "Sort" softkey.
	4. Press the "By name", "By IP", "By type" or "By location" softkeys to sort the display.
	
	
	

- OR -



3. Press the "Topology" softkey if you require an overview of all of the networked nodes, e.g. to check if there are connection problems.

The "Network Topology" window opens.

The interface, port and the actual connection are displayed for each node using a tree topology.

Note: The station name must be defined for all nodes so that the topology is correctly displayed.



4. Press the "Expand all" or "Close all" softkey to maximize or minimize the tree topology.



5. Press the "Follow connection" softkey to follow nodes in the structure tree using the marked line.

- OR -



3. If further information is provided for a selected node, press the "Details" softkey to open the Web user interface.

Note: The installed operating software must be located in the same network as the device to be displayed. If necessary, this is achieved by the assignment of a second IP address (alias).

The "Details" window opens.



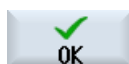
4. Press the "Zoom +" or "Zoom -" softkey until the desired display size is reached.



Additional nodes



3. Press the "Additional nodes" softkey to search for nodes that are not listed under "Accessible nodes".



The "Search for Additional Nodes" window opens. If necessary, change the address range and confirm with "OK".

The found nodes are appended in the table; the HW-Id/type, Location, Contact and Description fields can then be edited and, by activating the checkbox, added to the first column, and, with a subsequent save, added to the reference topology.

Comparing



3. Press the "Compare" softkey to compare the current list of nodes either with the previously saved manufacturer topology, a reference topology or a saved topology in the .xml format.



The "Compare with" window opens. Position the cursor on the desired option box and confirm with "OK".

Deviations during the comparison are identified by being color-highlighted:

- Red: missing or non-accessible components compared with the reference topology.
- Pink: components that deviate in their attribute values compared with the reference topology.
- Blue: new components not contained in the reference topology.

Saving



3. Press the "Save" softkey to save the current topology.
The "Save as" window opens.



4. Position the cursor on the desired option box, specify a file name and possibly a comment, and confirm with "OK".

Note

Manufacturer topology and reference topology can be created with access level 3 (user) or higher and are always saved on the active DHCP server.

Default storage locations:

- Current topology (.xml or .csv):
The following storage locations are offered depending on the configuration:
 - Network topology (storage location: HMI-data / data backups)
 - Local drive
 - Network drives
 - USB

In CSV format, ";" is preset as a separator; it can be changed in sldgconfig.xml

- Reference topology:
/user(_base)/common/tcu/\$soll
- Manufacturer topology:
/oem(_base)/common/tcu/\$soll

Service and diagnostics

17.1 NC/PLC variables

17.1.1 Displaying and editing PLC and NC variables

The "NC/PLC Variables" window allows NC system variables and PLC variables to be monitored and changed.

You receive the following list in which you can enter the desired NC/PLC variables in order to display the actual values.

- Variable
Address for NC/PLC variable.
Incorrect variables have a red background and are displayed with a # character in the value column.
- Comment
Any comment on the variable.
The columns can be displayed and hidden.
- Format
Specify the format in which the variable is to be displayed.
The format can be specified (e.g. floating point).
- Value
Displays the actual value of the NC/PLC variables.

PLC variables	
Inputs	Input bit (Ex), input byte (EBx), input word (EWx), input double word (EDx)
Outputs	Output bit (Ax), output byte (ABx), output word (AWx), output double word (ADx)
Bit memory	Memory bit (Mx), memory byte (MBx), memory word (MWx), memory double word (MDx)
Times	Time (Tx)
Counters	Counter (Cx)
Data	Data block (DBx): Data bit (DBXx), data byte (DBBx), data word (DBWx), data double word (DBDx)

Formats	
B	Binary
H	Hexadecimal
D	Decimal without sign
+/-D	Decimal with sign

Formats	
F	Floating point (for double words)
A	ASCII character

Notation for variables

- PLC variables
EB2
A1.2
DB2.DBW2
- NC variables
 - NC system variables - notation
\$AA_IM[1]
 - User variables/GUDs - notation
GUD/MyVariable[1,3]
 - OPI - notation
/CHANNEL/PARAMETER/R[u1,2]

Note

NC system variables and PLC variables

- System variables can be dependent on the channel. When the channel is switched over, the values from the corresponding channel are displayed.
 - For user variables (GUDs) it is not necessary to make a specification according to global or channel-specific GUDs. The indices of GUD arrays are, just like NC variables in the system variable syntax, 0-based; this means that the first element starts with the index 0.
 - Using the tooltip, for NC system variables, you can display the OPI notation (with the exception of GUDs).
-

Changing PLC variables

Changes can only be made to the PLC variables with the appropriate password.



WARNING

Incorrect parameterization

Changes in the states of NC/PLC variables have a major impact on the machine. Incorrect configuration of the parameters can endanger human life and cause damage to the machine.

Changing and deleting values



1. Select the "Diagnostics" operating area.



2. Press the "NC/PLC variables" softkey.

The "NC/PLC Variables" window opens.

3. Position the cursor in the "Variable" column and enter the required variable.



4. Press the <INPUT> key.

The operand is displayed with the value.



5. Press the "Details" softkey.

The "NC/PLC Variables: Details" window opens. The information for "Variable", "Comment" and "Value" is displayed in full length.



6. Position the cursor in the "Format" field and choose the required format with <SELECT>.



7. Press the "Display comments" softkey.

The "Comments" column is displayed. You have the option of creating comments or editing existing comments.



Press the "Display comments" softkey once again to hide the column.



8. Press the "Change" softkey if you would like to edit the value.

The "Value" column can be edited.



9. Press the "Insert variable" softkey if you wish to select a variable from a list of all existing variables and insert this.

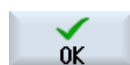
The "Select Variable" window opens.



10. Press the "Filter/search" softkey to restrict the display of variables (e.g. to mode group variables) using the "Filter" selection box and/or select the desired variable using the "Search" input box.



Press the "Delete all" softkey if you would like to delete the entries for the operands.



11. Press the "OK" softkey to confirm the changes or the deletion.

- OR -



Press the "Cancel" softkey to cancel the changes.

Note**"Filter/Search" when inserting variables**

The start value for "Filter/Search" of variables differs.

For example, to insert the variable \$R[0], set "Filter/Search":

- The start value is 0, if you filter according to "System variables".
 - The start value is 1, if you filter according to "All (no filter)". In this case, all signals are displayed and shown in the OPI notation.
-

Changing operands

Depending on the type of operand, you can increment or decrement the address by 1 place at a time using the "Operand +" and "Operand -" softkeys.

Note**Axis names as index**

For axis names, the softkeys "Operand +" and "Operand -" do not act as index, e.g. for \$AA_IM[X1].

**Examples**

DB97.DBX2.5

Result: DB97.DBX2.6

\$AA_IM[1]

Result: \$AA_IM[2]



MB201

Result: MB200

/Channel/Parameter/R[u1,3]

Result: /Channel/Parameter/R[u1,2]

17.1.2 Saving and loading screen forms

You have the option of saving the configurations of the variables made in the "NC/PLC variables" window in a screen form that you reload again when required.

Editing screen forms

If you change a screen form that has been loaded, then this is marked using with * after the screen form name.

The name of a screen form is kept in the display after switching-off.

Procedure



1. You have entered values for the desired variables in the "NC/PLC variables" window.
2. Press the ">>" softkey.
3. Press the "Save screen" softkey.
The "Save screen: Select archiving" window opens.
4. Position the cursor on the template folder for variable screen forms in which your actual screen form should be saved and press the "OK" softkey.
The "Save screen: Name" window opens.
5. Enter the name for the file and press the "OK" softkey.
A message in the status line informs you that the screen form was saved in the specified folder.
If a file with the same name already exists, they you will receive a prompt.
6. Press the "Load screen" softkey.
The "Load screen" window opens and displays the sample folder for the variable screen forms.
7. Select the desired file and press the "OK" softkey.
You return to the variable view. The list of all of the predefined NC and PLC variables is displayed.

17.2 Displaying the Service overview

The operating state of machine axes and drives is displayed in the "Service Overview" window. You can configure the display via the "Extended selection >" softkey or directly via the drop-down list according to the following criteria:

- All NC axes and drives (= default)
- NC axes
- Drives without NC axis assignment

Enable displays

- On/Off1
- Off2
- Off3
- From the drive: Operation enabled
- From the infeed: Enable operation
- Enable pulses
- Speed controller enable NC

- Enable pulses
- Drive ready
- Heat sink temperature
- Power section in i²t limiting
- Motor temperature
- **Measuring system 1 active / measuring system 2 active**
The states for measuring system 1/2 have the following meaning:

Symbol	Meaning
	The position measuring system is active.
	The position measuring system is parked. - OR - The position measuring system is not configured.
	The position measuring system is passive.

Procedure



1. Select the "Diagnostics" operating area.



2. Press the menu forward key.
New horizontal softkeys are displayed.



3. Press the "Axis diagnostics" softkey.
The "Service Overview" window opens.

4. Select an overview in the combo box "Select" combo box.
- OR -



- Press the "Go to selection" softkey to search for an overview.

17.2.1 Selecting axes and drives

To display certain enable signals and statuses of machine axes, you can arrange a selection of all the axes and drives that are available in any sequence you wish.

Note

The following selection configurations have been previously defined, and cannot be changed or deleted:

- All NC axes and drives
- NC axes
- Drives without NC axis assignment

Selection overview

The following information is provided in the selection overview.

Column	Description
Bus No.	Bus number for drives without NC assignment. No entry for NC axes.
Slave address	Slave address for drives without NC assignment. No entry for NC axes.
Device No.	Device number for drives without NC assignment. No entry for NC axes.
Drive object No.	Drive object number for drives without NC assignment. No entry for NC axes.
Name/identifier	For NC axes contains the standard machine axis identifier and the user-defined machine axis names. For drives without NC assignment, contains the drive object names.
Assignment	NC is displayed for an internal object assignment. PLC is displayed for an external object assignment.

Note

The selection overview differs in the information provided regarding the slave address and device number between PROFIBUS and PROFINET configuration.

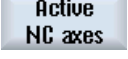
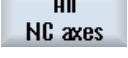
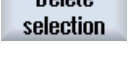
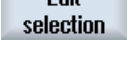
Procedure



1. Select the "Diagnostics" operating area.



2. Press the menu forward key.
New horizontal softkeys are displayed.

	3. Press the "Axis diagn." softkey. The "Service Overview" window opens.
	4. Press the "Extended selection" softkey. The window "Extended selection: Axes and drives" opens.
	5. Press the "New selection" softkey. The "New selection" dialog is opened.
	6. Enter a display and a file name in the entry fields. 7. Confirm the entry with the "OK" softkey. The "Edit selection" window is opened for the newly created selection.
	8. Set checkmarks in the check boxes for the axes and drives that you wish to display. - OR - Press one of the following softkeys to obtain a certain selection: All axes/drives are selected.
	
	No axis/drive is selected.
	Only internal drives are selected.
	Only external drives are selected.
	Those axes are selected that are assigned to a real drive.
	All axes are selected, which are defined in a minimum of one channel.
	9. Press the "OK" softkey to confirm your selection.
	10. Press the "Delete selection" softkey to delete the actual selection.
	11. Press the "Edit selection" softkey to edit the actual selection. The "Edit selection" window opens.
	12. If you click the "Back" softkey, you will return to the "Service Overview" main screen.

17.2.2 Axis diagnostics

Application

The following information is displayed in the "Service Overview" window:

- Check of the setpoint branch (e.g. programmed position setpoint, speed setpoint, spindle speed setpoint)
- Check of the actual value branch (e.g. actual position value, measuring system 1 or 2, actual speed value)
- Optimization of the position control loop of the axis (e.g. following error, control deviation, servo gain factor)
- Check of the entire control loop of the axis (e.g. through position setpoint/actual-value comparison and speed setpoint/actual-value comparison)
- Check of hardware errors (e.g. by checking the encoder: If the axis is moved mechanically, the actual position value must change)
- Setting and check of the axis monitoring functions.

Procedure



1. Select the "Diagnostics" operating area.



2. Press the menu forward key.
New horizontal softkeys are displayed.



3. Press the "Axis diagnostics" softkey.
The "Service Overview" window opens.



4. Press the "Service axis" softkey.
The "Service Axis/Spindle" window opens.



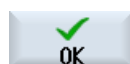
5. Press the "Axis +" or "Axis -" softkey to scroll the axes forward or backward.



- OR -



Press the "Axis selection" softkey.
The "Axis Direct Selection" window opens.
Select the required axis directly from those available in the drop-down list box.



6. Confirm the selection with "OK".
The values of the axis are displayed.

Display data

Display data	Meaning
Following error	The difference between the position setpoint and the actual position value of the active measuring system 1 or 2. Unit: mm, inch or degrees
System deviation	The difference between the position setpoint at the position controller input and the actual position value of the active measuring system 1 or 2. Unit: mm, inch or degrees
Contour deviation (axial)	The actual contour deviation is displayed with this value (variations of the following error caused by equalization operations on the speed controller due to load changes). The contour deviation results from the difference between an actual position pre-calculated from the position setpoint and the actual position value of active measuring system 1 or 2. Unit: mm, inch or degrees
Servo gain factor (calculated)	The servo gain factor in the display is calculated by the NC according to the following equation: $K_v - \text{factor} = \frac{\text{Speed setpoint}}{\text{Following error}} ;$ $\text{Unit (for default setting): } \frac{[\text{m/min}]}{[\text{mm}]} ;$ Velocity setpoint = setpoint currently being output to the axis/spindle References: Basic Functions Function Manual; Velocities, Setpoint / Actual Value Systems, Closed-Loop Control (G2)
Active measuring system	0: No measuring system active. 1: Measuring system 1 active. 2: Measuring system 2 active.
State of measuring system 1 State of measuring system 2	The state of measuring system 1 or 2 is output here: Active - Parked - Passive
Actual position value measuring system 1	The actual position of the axis measured via measuring system 1 or 2. The position is displayed in the machine coordinate system (no work offsets or tool offsets taken into account). Unit: mm, inch or degrees
Actual position value measuring system 2	
Position setpoint	Specified position transferred from the interpolator to the position control Unit: mm, inch or degrees
Absolute compensation value measuring system 1	Display of the absolute compensation value for measuring system 1 or 2.
Absolute compensation value measuring system 2	The compensation value consists of the sum of backlash and leadscrew error compensation for the actual axis position. Unit: mm, inch or degrees

Display data	Meaning
Compensation, sag + temperature	Display of the compensation value calculated for the current axis position based on the total of the sag and temperature compensations. Unit: mm, inch or degrees
Actual speed value, active encoder	The pulses supplied by the encoder are evaluated by the NC and displayed. Unit: % 100% means maximum speed.
Speed setpoint, drive	Speed setpoint transferred to the drive (= speed setpoint from position controller and feedforward control). Unit: % 100% means maximum speed setpoint.
Programmed spindle speed setpoint	Speed setpoint programmed by the user. Unit: rpm For example: Input: S1000; display: 1000 rpm Display applies to spindles only.
Spindle speed setpoint current	Current active speed setpoint with correct sign, including calculated compensation value and any active speed limitation (specified by setting or machine data) Unit: rpm Display applies to spindles only.
Position offset to the leading axis / spindle actual value	The currently valid position offset value is displayed here (relative to the actual value) if a position offset (angular offset between the following and leading spindle) has been programmed within the synchronous spindle functionality. Unit: mm, inches, degrees References: Function Manual, Extended Functions; Synchronous Spindle (S3)
Position offset to the leading axis / spindle setpoint	The currently valid position offset value is displayed here (relative to the setpoint) if a position offset (angular offset between the following and leading spindle) has been programmed within the synchronous spindle functionality. Unit: mm, inches, degrees References: Function Manual, Extended Functions; Synchronous Spindle (S3)
Override	The effective correction factor of the feed or spindle correction switch is displayed. Unit: %
Current gear stage	Display of the current actual gear stage With axes, this is only displayed if a spindle is assigned to the axis. The display corresponds to the NC/PLC interface signal: DB31, ... DBX16.0-2 (actual gear stage) References: Function Manual, Basic Functions; Spindles (S1)

Display data	Meaning
Parameter set (axis)	Displays which of the six parameter sets of the position controller is active. References: Basic Functions Function Manual; Velocities, Setpoint / Actual Value Systems, Closed-Loop Control (G2)
Controller mode	Display of the current controller state: • Position control • Speed control • Hold • Parking • Follow-up • Braking References: Function Manual, Basic Functions; Various NC/PLC interface signals and functions (A2)
Feedforward control mode	Indicates whether and if so, which mode of the dynamic feedforward control for the axis is active: • Inactive • Velocity The velocity-dependent speed feedforward control is active. • Torque The acceleration-dependent torque feedforward control is active (in combination with the speed feedforward control) References: Extended Functions Function Manual; Compensations (K3)
"Referenced" state	Status display for reference point approach (axis): • Active MS need not be referenced • Active MS referenced • Active MS must be referenced The display depends on the settings in the machine data: • MD34110 \$MA_REFP_CYCLE_NR • MD20700 \$MC_REFP_NC_START_LOCK The display corresponds to the NC/PLC interface signal: DB31, ... DBX60.4 and 60.5 (referenced/synchronized 1 or 2) References: Function Manual Basic Functions; Reference Point Travel (R1)

Display data	Meaning
QEC state	Indicates whether and if so, which method of the quadrant error compensation (QEC) is active for the axis: • Inactive • Neural QEC learning active • Conventional QEC active • Conventional QEC w. adapt. comp. value act. (conventional QEC with adaptation of the compensation value active) • Neural QEC active • Neural QEC w. ad. meas. time val. act. (neural QEC active with adaptation of the measuring time active) • Neural QEC w. ad. dec. time comp. val. act. (neural QEC with adaptation of decay time of compensation value active) • Neural QEC w. ad. meas. time + dec. time comp. val. act. (neural QEC active with adaptation of measuring time and decay time of compensation value active) References: Extended Functions Function Manual; Compensations (K3)
"Travel to fixed stop" state	Indicates whether or not the axis has fulfilled the conditions for "Fixed stop reached" when the "Travel to fixed stop" function is active (IS DB31, ... DBX62.5): • Normal control ("Travel to fixed stop" function not activated) • Fixed stop reached • Failed References: Function Manual, Basic Functions; Travel to Fixed Stop (F1)
Torque limitation value	Indicates the value programmed via FXST[x] or SD43510 \$SA_FIXED_STOP_TORQUE or the value defined via MD37010 \$MA_FIXED_STOP_TORQUE_DEF for the clamping torque for "Travel to fixed stop". Unit: % of maximum torque References: Function Manual, Basic Functions; Travel to Fixed Stop (F1)

17.2.3 Service drive

Display drive and motor data

The "Service Drive" window displays important information about the status of the motors and drive modules, such as motor temperature and DC-link voltage.

Procedure



1. Select the "Diagnostics" operating area.



2. Press the menu forward key.
A switch is made to a new softkey bar.



3. Press the "Axis diagnostics" softkey.
The "Service Overview" window opens.



4. Press the "Service drive" softkey.
The "Service Drive" window opens.



5. Press the "Drive +" or "Drive -" softkey to scroll forward or backward.



- OR -



Press the "Drive selection" softkey.
The "Drive Direct Selection:" window opens.
Select the desired drive directly from the selection list.



6. Confirm the selection with "OK."
The drive data is displayed.

Overview

The individual status displays, warnings, messages, etc. that are displayed in the "Service Drive" window are explained in the following.

Further information about the interface signals can be found in the following documentation:
Function Manual, Basic Functions (A2, Z1)

Further information about the drive parameters can be found in the following documentation:
List Manual, SINAMICS S120/S150 (Servo)

PLC pulse enable

The display, whether the pulse enable from the PLC is available for the drive corresponds to the interface signal:
DB31, ... DBX21.7 "Pulse enable".

Status	Meaning	Display
1	Pulse enable for this drive is activated by the PLC.	Yes
0	The pulses for the drive module are disabled from the PLC.	No

Speed controller enable NC

The display, whether the speed controller enable from the NC is available for the drive corresponds to the interface signal:

DB31, ... DBX61.6 "Speed controller active".

Status	Meaning	Display
1	Speed controller enable from the NC present.	Yes
0	Speed controller enable from the NC not present.	No

Rampup function generator quick stop

The status display for the ramp-function generator quick stop corresponds to the interface signal:

DB31, ... DBX92.1 "Ramp-function generator disable active".

Status	Meaning	Display
1	Ramp-up function generator quick stop is active. The drive is stopped without a ramp function with speed setpoint = 0 and without pulse suppression.	Yes
0	Ramp-up function generator quick stop is not active for the drive.	No

Pulses enabled

The message whether the pulses have been enabled for the drive corresponds to the interface signal:

DB31, ... DBX93.7 "Pulses enabled".

Status	Meaning	Display
1	The drive module pulses are enabled. The axis/spindle can now be traversed.	Yes
0	The drive module pulses are suppressed. The axis/spindle can therefore not be traversed.	No

Drive ready

The display of the current status of the selected drive corresponds to the interface signal:

DB31, ... DBX93.5 "Drive ready".

Status	Meaning	Display
1	The drive is ready.	Yes
0	The drive is not ready.	No

Ramp-up phase

The display of the current ramp-up phase of the selected drive corresponds to the drive parameter:

r0002 "Drive operating display".

Number of faulty signs-of-life

Display of communications errors detected in hardware between NC and drive.

Note

If the display shows a value other than "0", please contact your Siemens regional office!

SC1 drive alarm message

Displays (yes/no) whether messages of status class 1 are active. Status class 1 messages are alarms with the following properties:

- They lead to internal responses (e.g. regenerative braking, immediate pulse suppression)
- They are modal.

This is a group message. For details on the actually pending drive alarms, please refer to the "Drive System Diagnostics" window, see Service drive.

DC-link voltage, smoothed

The display of the smoothed actual value of the DC-link voltage of the selected drive corresponds to the drive parameter:
r0026 "DC-link voltage smoothed".

Unit: Volts

Speed setpoint, smoothed

The display of the smoothed speed setpoint - measured according to the setpoint limitation for the P component of the speed controller - corresponds to the drive parameter:
r1438 "Speed controller speed setpoint".

Unit: rpm

Actual speed value

The display of the smoothed actual value of the motor speed corresponds to the drive parameter:
r0021 "Actual speed value smoothed".

Unit: rpm

Actual current value, smoothed

The display of the smoothed actual current value corresponds to the drive parameter:
r0078[1] "Actual current value, torque-generating".

Unit: A

Motor temperature

The display of the current temperature in the motor corresponds to the drive parameter:
r0035 "Motor temperature"
Unit: °C

Integrator disabling

The display as to whether the integrator of the speed controller is active, corresponds to the interface signal:
DB31, ... DBX93.6 "n-controller integrator disabled".

Status	Meaning	Display
1	The requested shutdown of the speed controller integrator is active in the drive. The speed controller has been switched over from the PI to P control response.	Yes
0	The integrator of the speed controller is enabled. The speed controller functions as a PI controller.	No

Parking axis

Displays (yes/no) whether it is a parking axis/spindle.

Note

With parking axes/spindles, all encoder-specific monitoring and evaluation functions are switched off. This allows the encoder to be withdrawn without initiating an alarm.

Specified drive data set

The display (default: DDS0) of which of the eight drive parameter sets is to be activated by the PLC corresponds to the interface signal:
DB31, ... DBX21.0 to 21.2 "Drive parameter set selection A, B, C".

More detailed information on working with the drive and motor data sets can be found in:
Commissioning Manual, IBN CNC: NCK, PLC, Drive

Actual drive data set

The display (default: DDS0) of which of the eight drive parameter sets is currently active corresponds to the interface signal:
DB31, ... DBX93.0 to 93.2 "Active drive parameter set A, B, C".

Specified motor data set

The display (MDS0...3) of which motor data set is to be activated by the PLC corresponds to the interface signal:
DB31, ... DBX21.3 to 21.4 "Motor selection A, B".

The following assignment applies:

Motor data set	Coding	
MDS0	0	0
MDS1	0	1
MDS2	1	0
MDS3	1	1

Actual motor data set

The display (MDS0...3) of which motor data set is currently active corresponds to the interface signal:

DB31, ... DBX93.3 to 93.4 "Active motor A, B".

Operating mode

The display of the control type of a drive corresponds to the drive parameter: p1300[0...n] "Open-loop/closed-loop control mode".

Depending on "n", the following "values" are displayed:

n	Display
20	Speed control (without encoder)
21	Speed control (with encoder)
23	Torque control (with encoder)

Position actual value measuring system 1/2

The actual position of the axis as measured via measuring system 1/2. The position is displayed in the machine coordinate system (no zero offsets or tool offsets included).

Unit: mm, inches or degrees

Heat sink temperature

The display of whether the heat sink temperature is OK corresponds to the interface signal: DB31, ... DBX94.1 "Heat sink temperature prewarning".

Status	Meaning	Display
1	The drive signals a "Heat sink temperature prewarning" to the PLC.	Overtemperature
0	The drive module heat sink temperature pre-warning has not responded.	OK

Motor temperature

The display of whether the motor temperature is OK corresponds to the interface signal: DB31, ... DBX94.0 "Motor temperature prewarning".

Status	Meaning	Display
1	The motor temperature has exceeded the warning threshold configured in the drive.	Overtemperature
0	The motor temperature is below the warning threshold.	OK

Ramp-up function completed

The status display of the drive as to whether the ramp-up has been completed corresponds to the interface signal: DB31, ... DBX94.2 "Ramp-up completed".

Status	Meaning	Display
1	The actual speed value has reached the speed tolerance band specified via p2164 after a new speed setpoint specification and has not left the band for the duration of p2166.	Yes
0	The ramp-up procedure is still active after the speed setpoint has been changed.	No

Torque lower than threshold setting

The status display of the drive as to whether the threshold torque has been undershot corresponds to the interface signal: DB31, ... DBX94.3 " $|M_d| < M_{dx}$ ".

Status	Meaning	Display
1	The current torque utilization lies below the torque utilization threshold set in p2194. The drive signals the PLC that the torque setpoint $ M_d $ does not exceed the threshold torque M_{dx} .	Yes
0	The torque setpoint $ M_d $ is larger than the threshold torque M_{dx} . This signal can be used to determine whether the motor is overloaded.	No

Speed lower than minimum setting

The status display of the drive as to whether the minimum speed has been undershot corresponds to the interface signal: DB31, ... DBX94.4 " $|n_{act}| < n_{min}$ ".

Status	Meaning	Display
1	The actual speed value $ n_{act} $ is less than the threshold minimum speed n_{min} set in p2161.	Yes
0	The actual speed value is greater than the threshold minimum speed.	No

Speed lower than threshold setting

The status display of the drive as to whether the threshold speed has been undershot corresponds to the interface signal:

DB31, ... DBX94.5 " $|n_{act}| < n_x$ ".

Status	Meaning	Display
1	The actual speed value $ n_{act} $ is less than the threshold speed n_x set in p2155.	Yes
0	The actual speed value is greater than the set threshold speed.	No

Actual speed = set speed

The status display of the drive as to whether the actual speed value matches the speed setpoint corresponds to the interface signal:

DB31, ... DBX94.6 " $|n_{act}| < n_{set}$ ".

Status	Meaning	Display
1	The speed deviation between the setpoint and actual value is within the tolerance specified in p2163.	Yes
0	The speed deviation between setpoint and actual value is outside the specified tolerance.	No

Diagnostics for alarms

This information is also provided as a diagnostic tool for diagnosing the causes of alarms such as:

- Drive fault:
 ⇒ **SC1 drive alarm message** is set.
 ⇒ Check **specified drive data set, specified motor data set, DC-link voltage**.
- Alarm 25040 "Standstill monitoring",
 Alarm 25050 "Contour monitoring",
 Alarm 25060 "Speed setpoint limitation",
 Alarm 25080 "Positioning monitoring"
 ⇒ the drive enable may be missing (**PLC pulse enable** not present); this results in the display **Pulses enabled** = No.
- **Motor temperature** = exceeded
 ⇒ check the current motor temperature.

For information on the behavior of the NC controller in response to individual alarms and the remedy, please refer to:

References:

Diagnostic Manual

See also

Displaying drive states (Page 371)

17.3 System utilization

For the NC areas you can display the system resources (utilization display) currently being used:

- Runtime for the position controller, interpolator and pre-processing are displayed
- Time required for synchronized actions



Machine manufacturer

Please observe the machine manufacturer's data for the display of the time required for synchronized actions.

- Position controller and interpolator NC load
- Interpolator buffer level

Procedure



1. Select the "Diagnostics" operating area.



2. Press the menu forward key.
New horizontal softkeys are displayed.



3. Press the "System utilization" softkey.
The "System Utilization" window appears.
You can track the dynamic utilization display.



4. Press the "Stop" softkey to stop the display update.



5. Press the "Start" softkey to refresh the values.

17.4 Creating screenshots

You can create screenshots of the current user interface.

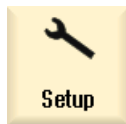
Each screenshot is saved as a file and stored in the following folder:

`/user/sinumerik/hmi/log/screenshot`

Procedure

- Ctrl + P Press the <Ctrl+P> key combination.
- A screenshot of the current user interface is created in .png format.
- The file names assigned by the system are in ascending order from "SCR_SAVE_0001.png" to "SCR_SAVE_9999.png". You can create up to 9,999 screenshots.

Copy file



1. Select the "Start-up" operating area.



2. Press the "System data" softkey.



3. Open the folder specified above, and select the required screenshots.
4. Press the "Copy" softkey.

- OR -



Press the "Cut" softkey.



5. Open the required archive directory, e.g. on a USB flash drive and press the "Paste" softkey.

Note

You can also copy the screenshots using "WinSCP" to a Windows PC.

Note

If you wish to view the screenshots, then you can open the files in SINUMERIK Operate. On a Windows PC, you can open the data using a graphic program, e.g. "Office Picture Manager".

17.5 Machine identity

You can store important information about the machine electronically that until now has only been available on paper for the end user notification administration (EUNA). If service is required, these entries can be viewed using remote diagnostics, or can be directly transferred to EUNA via the drives that have been set up.

You can store the following data for each machine:

- Machine type
- Machine number
- Manufacturer information

- Dealer information
- End user information
- Component list of the control technology that has been installed
- Logbook entries for changes to the machine

You enter the machine information and address information in the "identSNAPSHOT" window. The information can be either manually entered or using selection files.

Machine information

Entries	Meaning
Machine No.	The unique machine number (serial number of the CompactFlash card) is displayed in the header. This number serves as information only and cannot be changed.
Machine name	The manufacturer assigns a unique machine number in this field which is stored in machine data MD17400 \$MN_OEM_GLOBAL_INFO. Note: The machine name is a mandatory field. No data can be stored until this field is completed.
Machine type	Machine type.

Address information

Under Addresses you can enter the following data for the manufacturer/manufacturer's regional office, dealer and end customer:

- Customer number
- Manufacturer name and, if required, Branch name / Dealer name / End user name
- Street, Postal code/ZIP code, Location, Country (can be selected from a selection list), Region/State
- Address of contact: Name, Phone, Fax, E-mail, URL

17.5.1 Entering machine-specific information

Preconditions

You must have the following access rights to be able to enter or change machine-specific information.

Manufact.	Access level 1 (manufacturer)
Dealer	Access level 2 (service)
End user	Access level 3 (user)

Procedure



1. Select the "Diagnostics" operating area.



2. Press the "Version" softkey.
It takes some time to call the version display. While the version data is being determined a progress message box and the appropriate text are displayed in the dialog line.



3. Press the "Logbook" and "Change" softkeys.
The "IdentSNAPSHOT" window opens.



4. Press the "Manufacturer" softkey if you want to acquire and store manufacturer-specific data.
If you press the "Manufacturer" softkey again, it changes to "Manufacturer branch".
- OR / AND -



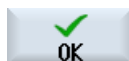
Press the "Manufacturer branch" softkey if you want to acquire and store address data of the manufacturer branch.
Pressing the softkey once again changes it back to "Manufacturer".
- OR / AND -



Press the "Dealer" softkey if you want to acquire and store dealer-specific data.
- OR / AND -



5. Press the "End user" softkey if you want to acquire and store user-specific data.



Enter the address data in the following text fields.
- OR -
If selection files exist, all unique information will automatically be set to valid for the system when you press the "OK" softkey. The entries are stored in the operating software.

See also

1. Documenting startup (Page 358)
 2. Defining start-up (Page 358)
- Making a logbook entry (Page 359)

17.5.2 Creating a selection file

Using the selection files, you can prepare the machine information and address information and transfer it into the controller. This means that it does not have to be manually entered.

Selection files

You can store the following selection files in the controller:

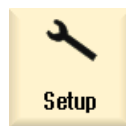
- "dealer.xml" for dealer data
- "ma_types.xml" for machine types
- "oem.xml" for manufacturer data
- "oemsubs.xml" for manufacturer data of a regional office
- "user.xml" for end user data

Templates

Templates for selection files are available under the directory /HMI-data/templates/examples/machine identity.

In future, EUNA data will also be available, which you can store in the controller. This therefore ensures that the address information is clear.

Procedure



1. Select the "Startup" operating area.



2. Press the "System data" softkey.



3. Copy the files from the folder /HMI-data/templates/examples/machine identity to a data carrier.
4. Copy the files to your PC to edit them.
5. After processing, copy the data to a data storage medium so that these can then be transferred to the control system.



6. Insert the copies into the folder HMI-data/Templates/Manufacturer/Machine identity or use the "Import data" softkey.

The addresses or machine types are displayed in drop-down list boxes. When you select a list, the data stored for that address is automatically entered in the input fields.

Entering dealer data

You can acquire the address data of any number of dealers in file "dealer.xml". If you wish to enter additional dealers, for each dealer, copy the area <Dealer> to <Dealer>.

```
<?xml version="1.0" encoding="utf-8" ?>
<Addresses>
  <Dealer>
```

```

    <Customer-Id></Customer-Id>
    <Name></Name>
    <Street></Street>
    <ZIP code></ZIP code>
    <Location></Location>
    <Country></Country>
    <State></State>
    <Contact></Contact>
    <Phone></Phone>
    <Fax></Fax>
    <E-mail></E-mail>
    <URL></URL>
  </Dealer>
</Addresses>

```

You can list the address data of any number of manufacturers in the "oem.xml" file. If you wish to enter additional manufacturers (OEMS), for each dealer, copy the area `<Manufacturer>` to `</Manufacturer>`.

You can list the address data of any number of OEM regional offices in the "oemsubs.xml" file. If you wish to enter additional regional offices, copy the area `<Manufacturer-Subsidiary>` to `</Manufacturer-Subsidiary>`.

You can list the address data of any number of users in the "user.xml" file. If you wish to enter additional users, copy the area `<User>` to `</User>`.

Example of file "ma_types.xml"

You can acquire the available machine types in file "ma_types.xml".

```

<?xml version="1.0" encoding="utf-8" ?>
<Machinetypes>
  <Type>LC 80</Type>
  <Type>LC 82</Type>
  <Type>LC 120</Type>
  <Type>LC 122</Type>
  <Type>LC 150</Type>
</Machinetypes>

```

17.5.3 Reading in a selection file

You can directly read-in your selection files into the controller. Depending on the configuration, files can be selected and imported from a local drive, network drive or USB. The data are then automatically copied into the following folder: `/oem/sinumerik/hmi/template/identity`.

Selection files

This procedure applies for the following selection files:

File	Information
ma_types.xml	Machine types
dealer.xml	Dealer's data
oem.xml	Manufacturer's data
oemsubs.xml	Manufacturer's data of a regional office
user.xml	End customer data

Procedure



1. Select the "Diagnostics" operating area.



2. Press the "Version" softkey.
It takes some time to call the version display. The progress of the data determination is displayed in the dialog line.

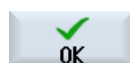


3. Press the "Logbook" and "Change" softkeys.



4. Press the "Import data" softkey.
The "Read-in Selection Files: Select File(s)" window opens.

5. Select the appropriate drive and the required file(s).



6. Press the "OK" softkey.
The file is automatically copied to the correct location in the system.

Note

Effectiveness of the selection file

A file selection (e.g. oem.xml) only becomes effective if you have still not saved any files in this area (e.g. manufacturer). Use the "Import data" softkey to make a new selection. You can then select a different address. The selection list disappears again when you save. You also have the option to delete all of the fields of the area that have been filled, and confirm this using the "OK" softkey.

See also

Creating a selection file (Page 350)

17.5.4 Save information

All the machine-specific information of the control is combined in a configuration via the user interface. You can save machine-specific information on the drives that have been set-up.

Procedure



1. Select the "Diagnostics" operating area.



2. Press the "Version" softkey.

It takes some time to call the version display. While the version data is being determined a progress message box and the appropriate text are displayed in the dialog line.



3. Press the "Save" softkey.

The "Save Version Information: Select Archive" window opens. The following storage locations are offered depending on the configuration:

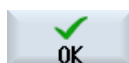
- Local drive
- Network drives
- USB
- Version data (archive: Data tree in the "HMI data" directory)



4. Then press the "New directory" softkey if you wish to create your own directory.



5. Press the "OK" softkey. The directory is created.



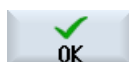
6. Press the "OK" softkey again to confirm the storage location.

The "Save Version Information: Name" window opens. The following options are available:

- In the "Name:" text field, the file name is pre-assigned with <Machine name/no.>+<CF-card number>. "_config.xml" or "_version.txt" is automatically attached to the file names.
- In text field "Comment", you can add a comment that is stored with the configuration data.

Select the following data via a checkbox:

- Version data (.TXT): Output of pure version data in text format.
- Configuration data (.XML): Output of configuration data in XML format. The configuration file contains the data you entered under Machine identity, the license requirements, the version information and the logbook entries.



7. Press the "OK" softkey to start the data transfer.

17.5.5 Adding hardware components

Most of the hardware components supplied by Siemens are already electronically acquired. You can add further hardware components.

You can view and supplement the hardware components in the version screen at the user interface.

All of the hardware components are listed in the version display and in the configuration data.

Precondition

You require access level 2 (service) to add more hardware components.

Entering additional components

If new hardware components are added, they must be assigned to the appropriate categories. The "Select component" drop-down list box offers the following categories in the "Add HW Comp." window:

Category
NCU/PLC
Operator Panel
PLC-peripheral devices
Drive/Motor
Cable
Accessories/Miscellaneous

You record the following data in the "Enter Additional Components (Configuration Data)" window.

Entries	Meaning
Name	Hardware designation
Version	Version of the program
Order No.[MLFB]	Order number
Serial number	Serial number
Number	Number of components

Procedure



1. Select the "Diagnostics" operating area.



2. Press the "Version" softkey.

It takes some time to call the version display. While the version data is being determined a progress message box and the appropriate text are displayed in the dialog line.



3. Select the "Hardware" area and press the "Details" softkey. The "Version Data / Hardware" window opens.

The "Add HW comp." softkey appears.



4. Press the "Add HW comp." softkey.

The "Select component type" drop-down list box opens.

5. Select a category under which you would like to store the hardware information.

The "Enter Additional Hardware Components" input window opens.

6. Enter additional hardware components via the keyboard.

- OR -

If you are using a CSV file (CSV = Comma Separated Values), for example a EUNA parts list, you can also read in the data.

Select the storage location where the CSV file is located.

Select the file and press "OK".



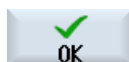
The data is transferred from the CSV file into the table. It is read-in according to the CSV format of the EUNA parts list. For each component, quantity, MLFB, name and serial number are entered. The version is not saved and is not transferred.

Example:

QUANTITY;MLFB;Z_OPTION;SERIAL

12;6AV7812-0BB11-2AC0;;

1;6FC52030AB110AA2;;T-0815



7. Press the "OK" softkey.

The data is written to the "versions.xml" configuration file and is therefore electronically acquired.

All hardware components are displayed in the "Version Data / Hardware" window.

The manually entered components are assigned a "+", e.g. Cable+.

17.5.6 Configuration data

The machine-specific information is stored in the configuration data. This file saved in the XML format provides the basis for further processing in EUNA.

- The data can be read by remote diagnostics.
- The data can also be transferred directly to EUNA, e.g. via network or USB FlashDrive.

Configuration data

The configuration data includes:

- Machine-specific data that is stored in the "Machine identity" dialog box
- Hardware/software versions that are stored in the "Versions" dialog box
- Options requiring licenses, which are stored in the "Licensing" dialog box
- Logbook, whose entries are stored in the "Logbook" dialog box

Storage path

Depending on how the drives have been configured, the configuration data can be stored in the available directories.

See also

Machine identity (Page 348)

Adding hardware components (Page 355)

Licensing (Page 41)

Logbook (Page 357)

17.6 Logbook

The logbook provides an electronic machine history.

The time and date of commissioning is recorded in the logbook, and if the machine is serviced, this can also be logged electronically. This helps to optimize the service.

Stored entries cannot be changed or deleted.

17.6.1 1. Documenting startup

Precondition

As a minimum, the machine name/No., the customer No. and the country where the manufacturer is located must be entered into the machine identity.

Procedure



1. The machine-specific information data are entered.



2. Start-up (commission) the machine.
3. Press the "1st startup completed" softkey.
The "1.SETUP" entry is displayed in the "Machine logbook" window together with the time and date.

Note

2. Perform commissioning

On completion of first commissioning, you will be prompted to perform the 2nd commissioning.

If the 2nd commissioning is not performed, then you will be prompted at regular intervals to complete the 2nd commissioning.

See also

Entering machine-specific information (Page 349)

17.6.2 2. Defining start-up

Precondition

As a minimum, the country of the end customer must be entered into the machine identity.

Procedure



- The machine-specific information data are entered.



2. Start-up (commission) the machine.
3. Press the "2nd startup completed" softkey.
The "2.SETUP" entry is displayed in the "Machine logbook" window together with the time and date.

Note**Registering the machine identity in EUNA**

After completion of the 2nd commissioning, you are prompted to send the machine identity to the EUNA database via the Internet.

See also

Entering machine-specific information (Page 349)

17.6.3 Making a logbook entry

Using the "New logbook entry" window to make a new entry into the logbook.

Enter your name, company and department and a brief description of the measure taken or a description of the fault.

Note

If you wish to make line breaks in the "fault diagnostics/measure" field, use the key combination <ALT> + <INPUT>.

The date and entry number are automatically added.

Sorting the entries

The logbook entries are displayed numbered in the "machine logbook" window.

More recent entries are always added at the top in the display.

Procedure

1. The logbook is opened.
2. Press the "New entry" softkey.
The "New logbook entry" window opens.
3. Enter the required data and press the "OK" softkey.
You return to the "Machine logbook" window and the entry is displayed below the machine identity data.

Note**Deleting logbook entries**

Up to the end of the 2nd commissioning, you have the option to delete the logbook entries up to the time of the first commissioning using the "Clean up" softkey.

Searching for a logbook entry

You have the option for searching for specific entries using the search function.

- 
1. The "Machine logbook" window is opened.

2. Press the "Search..." softkey and enter the desired term in the search form. You can make a search according to date/time, company name/department or according to fault diagnostics/measure.
The cursor is positioned on the first entry that corresponds to the search term.

3. Press the "Continue search" softkey if the entry found is not the one that you are looking for.

Additional search option

- 
- Press the "Go to Beginning" softkey to start the search at the latest entry.

Press the "Go to End" softkey to start the search at the oldest entry.

17.7 Action log

17.7.1 Setting the action log

In the "Action Log Settings" window you specify whether the operating sequences are to be logged.

Here, you select which control actions should be logged, so that operator actions and sequences can be tracked and understood at a later date.

When the action log is active, you can display recorded traces in the system data using the following files:

- "actual_actionlog.com", actual action log report
- "actual_crashlog.com", crashlog backup

Data that can be logged

Logging on	Logging is activated or deactivated.
Alarm status change	Incoming and outgoing alarms are logged.
Keyboard actions	All actions on the operator panel front and on an external keyboard are logged.
Channel status change	NC/PLC states are logged via the information channel status. If this information can be obtained in a timely manner, in many cases these states can be used to verify the operation of the machine control panel.

Window switchover	The form and dialog names (assigned by the programmer) are logged each time a window is opened or closed.
Writing NCK/PLC data	The writing of NCK and PLC variables is logged.
File access	Copying to the NC is logged.
Function calls in the NCK (PI service)	Some program calls, for example, ASUB, are logged.
Curr. program status	Additional information for particular events is activated. For important alarms that require NC stop, NC start, or NC reset, actual values and the current block are recorded for example. The program ascertains which channel and which spindle are used for the additional information.
Write interval file	Settings for updating files: • "automatic": The action log saves the information to an internal buffer. If the buffer is full the entries are written to the CF card. The entries may be lost when the controller is switched off. • "after every event": All entries are backed up directly, this prevents data loss due to a power failure or similar. Notice: As CF cards only permit a limited number of write accesses, this setting is not recommended for normal operation. • "time-controlled": New entries are backed-up for a particular length of time. An additional input field appears in which you can specify a time in seconds.
Save log as file for alarm(s)	The alarm numbers for which a "crash log" is generated are specified. The alarms are entered separated by a comma.

Generating a crashlog file

The "actual_crashlog.com" is a backup file of the actual log when a critical event occurs (for example, emergency stop alarm).

The file is then no longer stored in the ring buffer of the action log and can no longer be overwritten with new entries.

The entries in the crash log file are only overwritten when yet another critical events occurs.

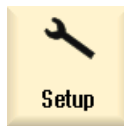
At the beginning, the file has no entries and is only written to after the following events:

- Interface signal DB19.DBX0.6 "Save teleprinter log" changes from 0 to 1.
- The alarm entered in the "Save log as file for alarm(s)" entry field is output.

Interface signal

DB19 (PLC → HMI)								
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB00		Save action log						

Procedure



1. Select the "Start-up" operating area.



2. Press the "HMI", "Diagnostics", and "Action log" softkeys
The "Action Log Settings" window opens.



3. Activate the "Logging on" checkbox to activate the action log.
4. Check the relevant checkbox if you want to acquire particular events in the action log.

See also

Displaying the log file (Page 362)

17.7.2 Displaying the log file

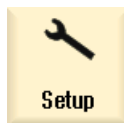
Loggable data that was logged when the action log function was active is displayed in the "Action Log File" window.

Log files

The actual data can be displayed using the following files:

- "actual_actionlog.com"
- "actual_crashlog.com"

Procedure



1. Select the "Start-up" operating area.



2. Press the "System data" softkey.
The data tree is displayed.
You will find the saved files in the HMI data folder under /logs/action log.



3. Select the desired com file and press the "Open" softkey or double-click the file.
4. Press the "Display new" softkey to update the display of events in the log (actual_actionlog.com).
The entries made since the log file was called up are displayed.

17.7.3 Searching in the log files

You can search for specific events in the log.

Precondition

The relevant log file is open.

Procedure



1. Press the "Search" softkey.
A new vertical softkey bar appears. The "Search" window opens at the same time.
 2. Enter the desired search term in the "Text" field.
 3. Position the cursor in the "Direction" field and choose the search direction (forward, backward) with the <SELECT> key.
 4. Press the "OK" softkey to start the search.
If the text you are searching for is found, the corresponding line is highlighted.
Press the "Continue search" softkey to continue the search, as necessary.
- OR -
- Press the "Cancel" softkey when you want to cancel the search.

Additional search option



1. Press the "Go to End" softkey to scroll to the end of a large log. You get to the oldest entry of the recording.
2. Press the "Go to Beginning" software to return quickly to the most recent log entry.

17.7.4 Storing a log

You can store the displayed log in the "HMI data/Logs/Alarm logs" directory. It is stored as a binary file as well as ASCII file. You can read an ASCII file with any editor.

The following logs are generated:

- action.com (binary file)
- action.log (ASCII file)
- crash.com (binary file)
- crash.log (ASCII file)

Requirement

The relevant log file is open.

Procedure



1. Press the "Save log" softkey.
The log is stored in the "HMI data/Logs/Alarm logs" directory.

Note

The "Save log" softkey is only available for files that have not yet been saved.

17.7.5 Structure of a log file

Content of the log file

The following data is recorded for every logged event:

HMI version		NC version	
System	Entry keyword		Date/time
	Entry text		
	DETAILS		

Entry		Description
HMI and NC version output		
System:		
	HMI	Keystrokes, window change
	NC	Write variables, PI services
	MSG	Alarm entries
	USR	Entries generated by the machine manufacturer for the user
	ERR	Action log error, e.g. log file not found
Entry keyword:		
	HMI_START	Entry of an HMI boot
	HMI_EXIT	Entry of an HMI shutdown

Entry		Description	
	PLC_CRASH	Entry of a PLC crash	
	KEY_PRESSED	Entry of a keystroke	
	KEY_HOLD	Entry of a key hold	
	KEY_RELEASED	Entry of a key released	
	ALARM_ON	Entry of an incoming alarm event	See below "Alarm display"
	ALARM_OFF	Entry of an alarm going event	
	ALARM_ACK	Entry of an alarm acknowledgement event	
	OPEN_WINDOW	Entry of a window opening	
	CHN_STATE_CHANGED	Entry of a channel status change	
	OPMODE_CHANGED	Entry of an operating mode change	
	TOOL_CHANGED	Entry of a tool change	
	OVERRIDE_CHANGED	Entry of override changes	
	DOM_CMD	Entry of a download to the NC	
	PI_CMD	Entry of a PI service	
	WRITE_VAR	Entry of a write to NCK/PLC variable	
	AREA_CHANGED	Entry of an area changeover	
	NC_CONNECTION	Entry of an coming/going NC connection	
	USER	User entry via the OEM interface	
	ACTIVATED	Action log was activated	
	DEACTIVATED	Action log was deactivated	
	INTERNAL	Internal action log entry	
Date/time		Date and time of the event For HMI_Start, PLC_Crash or date change since the last entry, the intermediate period between the entry keyword and date is designated with an "-".	
Entry text		Event data is written in plain text.	
DETAILS		If additional information regarding an entry is known, e.g. a recorded IPO trace.	

Example

Note

DB19.DBW24 interface signal

The DB19.DBW24 interface signal used in the example does not apply to the SINUMERIK 840D sl. It is DB1900.DBW4 for the SINUMERIK 828D.

```

NCK    WRITE_VAR                                18.02.2009  09:40:12
        ncui.local wrote variable: DB19.DBW24 = 0
HMI    KEY_PRESSED                              18.02.2009  09:40:12
        Key up: "Return" (1000004/ 0)
HMI    HMI_START -----                      18.02.2009  09:34:15
        HMI started.
```

Alarm display

```
MSG      ALARM                                     Date/time
      No.      Deletion criterion  Text
      DETAILS
```

Entry		Description
MSG		
ALARM		Type of alarm: ALARM_ON, ALARM_OFF, ALARM_ACK
Date/time		Data and time of the event
No.		Alarm number
Deletion criterion:		Information as to how the alarm was acknowledged
	AUTOMATIC	Automatic acknowledgement
	POWER ON	Acknowledgement using PowerOn
	RESET	Acknowledgement using NCK reset
	NC START	Acknowledgement using NC start
	NC RESET	Acknowledgement using NC reset
	ALARM CANCEL	Acknowledgement using Alarm Cancel key
	RECALL	Acknowledgement using the Recall key
	HMI	Acknowledgement using HMI
	PLC	Acknowledgement using PLC
Text		Display of the English alarm text
DETAILS		If additional information regarding an entry is known.

Example

```
MSG      ALARM_ON                                     20.02.2009   14:25:37
      8020   POWER ON   : Option 'activation of more than 1 channels' not
      set
      DETAILS:
      Mode: JOG Program: canceled Channel: interrupted
      Program-Level information:
      Level Program running:                               Invoc      Offset
      1      /_N_MPF0                                     0          0
```

17.7.6 Advanced settings

You define the size of the log file of the action log in the "actlog.ini" configuration file.

Procedure

1. You can copy a sample configuration file "actlog.ini" from the following directory: /siemens/sinumerik/hmi/template/cfg.
2. Save or create the file in the /oem/sinumerik/hmi/cfg or /user/sinumerik/hmi/cfg directory.
3. Open the file and after "; ActionLogSize=", enter the file size (in bytes).

"actlog.ini" configuration file

```
Template for the configuration of Action Log
;
; To activate the settings remove the
; commentary ';' at the beginning of the line
[ActionLogSettings]
;To change the path of the internal action log file use the following
setting
;ActionLogPath=/user/sinumerik/hmi/ac_log/action.com
;To change the size of the internal action log file use the following
setting
;ActionLogSize=5000000
;To change the path of the internal crash log file use the following
setting
;CrashLogPath=/user/sinumerik/hmi/ac_log/crash.com
```

17.8 HMI trace

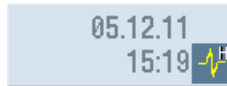
Creating an HMI trace

The "HMI trace" function is available for extended diagnostics of the operating software. The prerequisite is at least access level 3 (user).

Procedure to create an HMI trace file:

1. An expert of the Service & Support Center provides a sltrc_keybitset.ini file. Copy the sltrc_keybitset.ini file to a USB memory medium and load the file on the controller or operator panel.
2. In the operating area, select "Start-up" → "HMI" → "Diagnostics >" → "HMI trace".
3. Select the INI file on the USB memory medium with the "Load configuration" softkey.

4. Start the trace by activating the checkbox or from any place on the user interface with the following key combination:
<Alt> + <T> on the operator panel
<Alt> + <Shift> + <T> on an external keyboard
The following icon is displayed at the top right in the header to show that a trace is active:



5. On the user interface, repeat the operating sequence whose cause is to be clarified.
6. Stop the trace by deactivating the checkbox or from any place on the user interface with the following key combination:
<Ctrl> + <T> on the operator panel
<Ctrl> + <Shift> + <T> on an external keyboard
7. A sltrc.out output file is generated which you can copy to the storage medium with the "Export data" softkey.
8. Send the sltrc.out file for analysis back to the Service & Support Center.

Note

On completion of the analysis, it is recommended to delete all files under the following path: ../user/sinumerik/hmi/log/sltrc

Note

"Extended" softkey

The other options in this dialog support suitably qualified service engineers in the analysis of the system and evaluation of the HMI trace.

17.9 PROFIBUS diagnosis

Display the PROFIBUS status for diagnostic purposes during the configuration or when errors occur. This diagnostics window is only intended for information purposes. You cannot modify them.

PROFIBUS connections

- DP1 X126
- DP2 X136
- DP integrated

Display	Meaning/information
Status	
Bus status	POWER_ON (0): State after the control has been switched on OFFLINE (1): Basic initialization has been performed STOP (2): Start in accordance with the hardware configuration (SDB) CLEAR (3): PROFIBUS slaves have been parameterized and configured in accordance with the hardware configuration (SDB) and taken into the cyclic data exchange with zero output data OPERATE (4): Cyclic data exchange with the PROFIBUS slaves running ERROR (9): A fatal error had been detected (e.g. invalid or faulty SDB) (<n>) corresponds to the OPI variable value for the bus state.
Bus configuration	
S7 Subnet ID	S7 Subnet ID of the PROFIBUS subnet
Baud rate	Transmission rate in Mbd
Cycle time	Configured bus cycle time in msec; also defines the position controller cycle at the same time
Sync. Component (TDX)	Configured time interval in msec for the cyclic data exchange within a PROFIBUS DP cycle
PROFIBUS diagnostics/slaves	
Slave address	Configured number of the DP slave
Assignment	Information whether the DP slave is assigned to the NC or to the PLC. NC: e.g. one or more drives controlled by the NC PLC: e.g. I/O peripherals or an axis controlled by the PLC NC/PLC (for DP integrated)
Communication state	Information whether the DP slave is identified on the bus via the communication state.  Green: DP slave has been detected on the PROFIBUS DP and the data exchange is working error-free with the assigned component (NC and/or PLC)  Red: Incorrect response / no communication
Synchr. with NC	Information whether the DP slave is running synchronously to NC on the bus.  Green: DP slave runs synchronously to NC on the PROFIBUS DP, i.e. there is an isochronous data exchange  Red: Incorrect response / no communication  Gray: DP slave is not assigned to the NC, but to the PLC
Number of slots	Number of configured slots within the DP slave

References

Further information about the configuration of the properties of the network interface for PROFIBUS can be found in the following:

CNC Commissioning Manual: NCK, PLC, Drives, SINUMERIK 840D sl, SINAMICS S120

Procedure



1. Select the "Diagnostics" operating area.

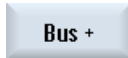


2. Press the "Menu forward" key and the "Bus TCP/IP" softkey.
The "PROFIBUS Diagnostics" window is opened.



3. If several PROFIBUS connections have been configured, press the "Bus -" or "Bus +" softkey to select the desired configuration.

...



- OR -



Press the "Select bus" softkey and in the list that is displayed, select the required configuration.



Press the "OK" softkey.

17.9.1 Displaying details for DP slaves

Further information about the slots of a selected DP slave is displayed here.

Display	Meaning/information
Slave	
Address	The DP slave selected in the PROFIBUS diagnostics / slaves pane
Communication state	Data regarding the NC and/or PLC assignment of the DP slave
In synchronism with NC	Data regarding NC synchronous operation of the DP slave If a synchronous value is not available for the DP slave, then the display remains hidden.
Slots	
No.	Slot number within the DP slave
I/O addr.	I/O address in the I/O address space of the PLC, assigned to this slot For NC axes, the setpoint and the actual value must always be configured on the same I/O address.
Type	Data as to whether the slot is input, output or diagnostic slot If the slot is assigned to an NC axis, then the output is always denoted as setpoint value and the input always as actual value.
Length (bytes)	Length of the I/O area reserved for the slot in the STEP7 I/O address space

Display	Meaning/information
Communication state	Actual communication state of the slot If a value is not available for the NC axes, then the display remains hidden.  Green: Slot used by the NC, communication active  Red: Slot used by the NC, communication currently not active  Gray: No NC axis
Machine axis	Display of the name defined in the machine data for this slot If the slot is not assigned to any NC axis, then the display remains hidden.
Telegr. type (NC)	Displays the assigned telegram type If a telegram type is not assigned in the NC machine data, then the display remains hidden.

Procedure



1. Select the "Diagnostics" operating area.



2. Press the "Menu forward" key and the "Bus TCP/IP" softkey.
The "PROFIBUS Diagnostics" window is opened.



3. Select the PROFIBUS configuration for which you want to display details.



4. Press the "Details" softkey.
The "PROFIBUS Diagnostics ... - Details" window is opened.

17.10 Drive system

17.10.1 Displaying drive states

Status display

The "Drive System Diagnostics" window displays the status information about the drive units and the associated drive objects. The status symbols have the following meaning:

Display icon		Meaning
	Green	The drive object is in (cyclic) operation, without any detected problems.
	Yellow	The drive object has detected a less serious problem, i.e. a warning is pending, for example, or enables are missing.
	Red	This drive object has detected a serious problem, for example, an alarm is pending.

Display icon		Meaning
	Gray	The drive status could not be determined for this drive object.
#	Special characters	Error when reading the data.

Procedure



1. Select the "Diagnostics" operating area.



2. Press the menu forward key and the "Drive system" softkey.



3. The "Drive System Diagnostics" window opens.
The name of the selected drive object is displayed in the window title.
Press the "Drive object +" or "Drive object -" softkey.
The next (+) or previous (-) drive object is selected.



- OR -



4. Press the "Select drive unit" softkey.
The "Select Drive Unit" window opens.
Select the desired drive unit via the drop-down list box and press "OK" to confirm the selection.

See also

Displaying details of the drive objects (Page 372)

17.10.2 Displaying details of the drive objects

Drive system diagnostics - details

The following information is displayed for the drive objects:

- Status display (r0002)
 - The status display (parameter r0002) of the drive object is displayed.
- Missing enable signals (r0046)
 - The enable signals are diagnosed except for the drive objects that do not have enable signals (e.g. Control Unit). Missing enables are shown in a pane below the displays.

- Commissioning parameter filter
 - (p0009) – control unit
The value of the "Device commissioning parameter filter" control unit parameter (p0009) is displayed.
 - (p0010) – further drive object
The value of the "Drive object commissioning parameter filter" parameter (p0010) is displayed.
- Current fault
The alarm number of the actual drive object fault is displayed or "No fault present", when no drive object fault is pending.
You obtain an overview of the pending faults with the following data in an additional window:
 - Alarm number
 - Time: Date and time
 - Alarm text
- Current warning
The number of the current drive object warning is displayed or "No warning present", when no drive object warning is pending.
You obtain an overview of the pending alarms with the following data in an additional window:
 - Alarm number
 - Time: Date and time
 - Alarm text

Meaning of the symbols

Display icon		Meaning
	Green	The relevant value of the drive object signals no problem detected.
	Yellow	The relevant value of the drive object signals a less serious problem, i.e. a warning is pending, for example, or enables are missing.
	Red	The relevant value of the drive object signals a serious problem, for example, an alarm is pending.
	Gray	The drive status could not be determined for this drive object.
#	Special characters	Error when reading the data.

Procedure



1. Select the "Diagnostics" operating area.



2. Press the menu forward key and the "Drive system" softkey.



The "Drive System Diagnostics" window opens.



3. Select the drive object for which you want to display further details.
4. Press the "Details" softkey.

The "Drive System Diagnostics - Details " window opens.

The name of the drive object is displayed in the window header.



Press the "Faults" softkey.

The "Drive Unit Faults" window opens that shows an overview of the faults that have been output.



Press the "Alarms" softkey.

The "Drive Unit Alarms" window opens that shows an overview of the alarms that have been output.



You can display the pending SI messages if Safety Integrated is integrated in your controller.

17.11 Remote diagnostics

17.11.1 Adapting remote diagnostics

Depending on the operating mode, the initiative for remote diagnostics comes from the service provider or from the controller.

If the initiative comes from the controller, the "Request remote diagnostics" softkey is set-up.

The display of the "Request remote diagnostics" softkey is defined in the "tcu.ini" file under the [PINGSERVICE] section.

- Variable PingServiceMode
0 = disable
1 = enable

In additional variables of this section, the data for the request telegram for service should be set.

- Variable PingServerIP
IP address of the service provider to which the request telegram should be sent, e.g.
PingServerIP = 128.45.234.19
- Variable PingTransmissionData
A text that should be sent out as data content in the request telegram,
e.g. PingTransmissionData = help at machine 35
- Variable PingTransmissionPeriod
The length in minutes that the telegram should be sent,
e.g. PingTransmissionPeriod = 5

- Variable PingTransmissionInterval
Duration in seconds after which send is repeated,
e.g. PingTransmissionInterval = 15
- Variable PingServerPort
Port number of the Ping service with the manufacturer,
e.g. PingServerPort=6201

17.11.2 Remote diagnostics via Teleservice adapter IE at X127

The following settings must be made for remote diagnostics via modem using a Teleservice adapters IE (TS Adapter) that is connected at X127:

Settings in the TS Adapter required

- IP address of the adapter: 192.168.215.30, subnet mask 255.255.255.224
- IP address of the caller: 192.168.215.29
These two addresses are also stored as standard in the "tcu.ini" file. If other addresses are used, the entries "X127ModemIP" and "X127RemoteAccessIP" must be appropriately changed in the "tcu.ini" file.
- Enable the ports for the services for S7 communication, http, HTTPS, SSH as well as VNC (port 5900) in the firewall.

Connecting the modem for remote diagnostics

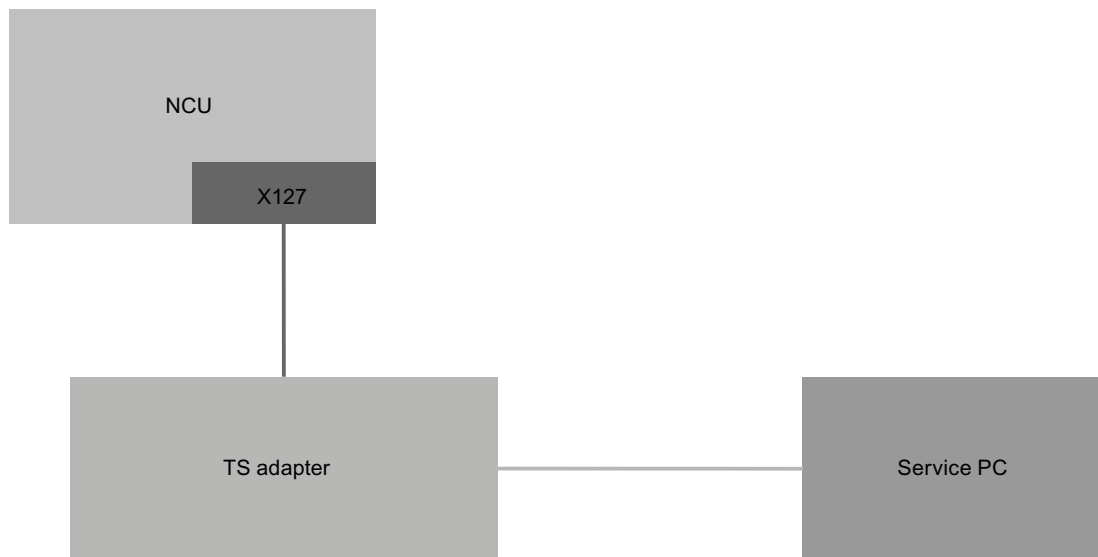


Figure 17-1 Basic system with SINUMERIK Operate on NCU (and operating station with TCU)

With the previously mentioned settings and the default values, in this configuration, remote diagnostics via TS Adapter is possible without any additional changes having to be made.

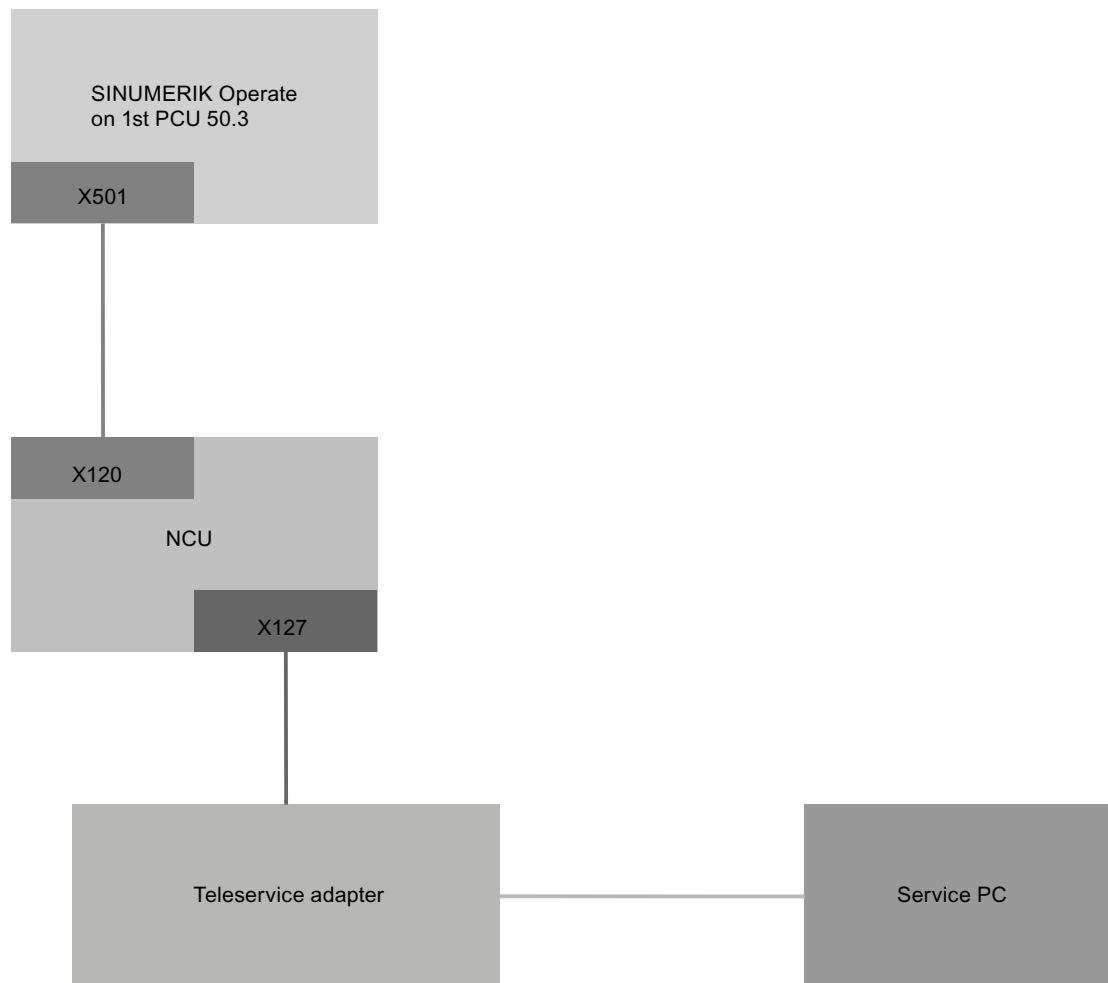


Figure 17-2 Basic system with SINUMERIK Operate on PCU

With the previously mentioned settings and the default values, in this configuration, remote diagnostics via TS Adapter is possible without any additional changes having to be made. When operating a SINUMERIK Operate on a PCU, the SINUMERIK Operate in the NCU must be switched off.

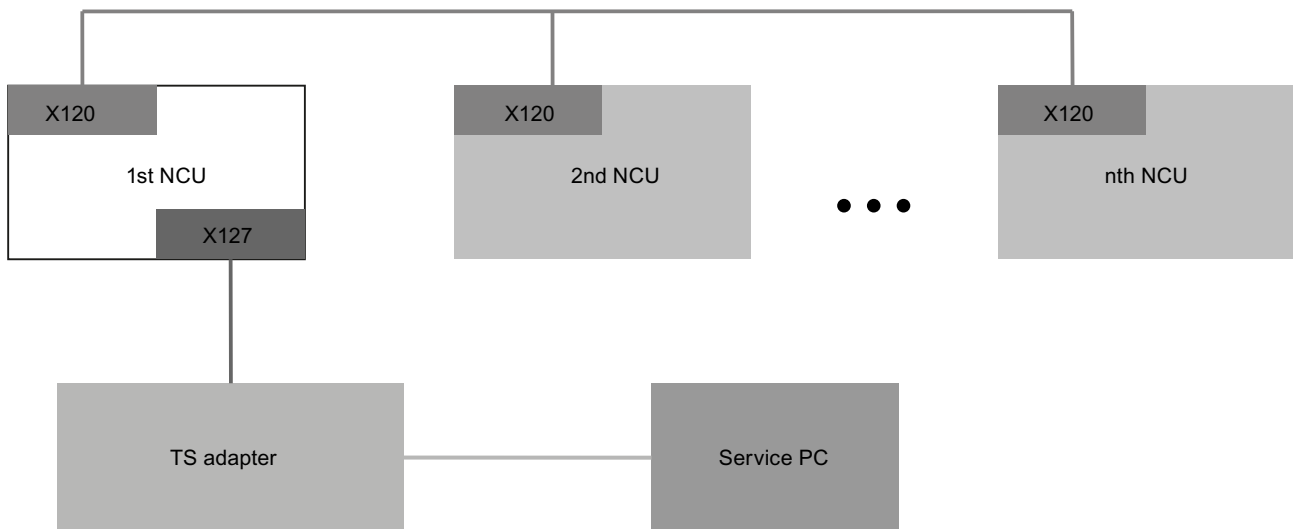


Figure 17-3 System with several NCUs and a fixed modem

The "MC Information System RCS Host" option only has to be set for the NCU to which the TS Adapter is connected at the X127 interface. Modem access can only be permitted by making the appropriate entries at the SINUMERIK Operate in this NCU and the "Request remote control" function executed. If modem access is permitted, then the service PC implicitly has access to all other NCUs and their operating software. On the other hand, the operating screens to control the remote access can be used and set at each station (SINUMERIK Operate in the NCU) individually for the station in conjunction with the relevant local PLC.

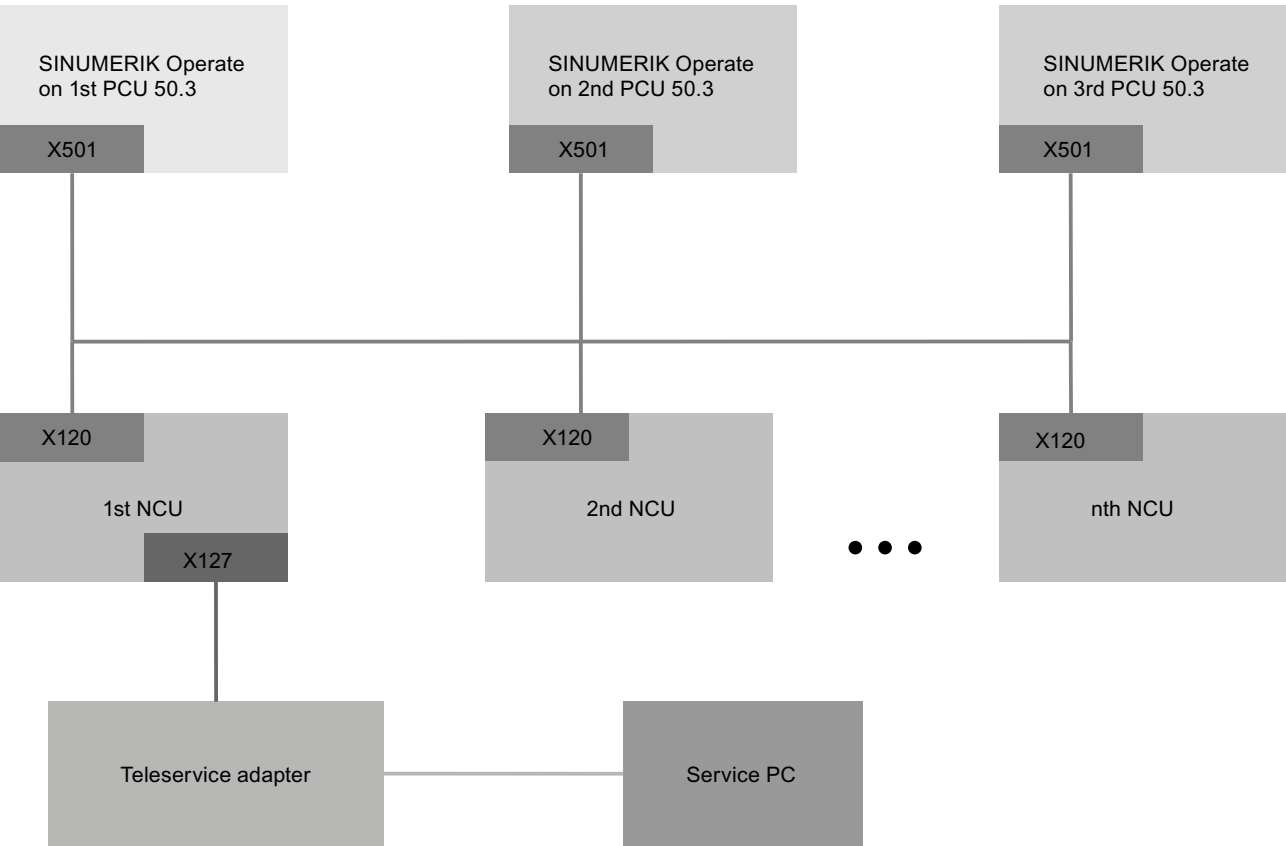


Figure 17-4 System with several NCUs, several PCUs and a fixed modem

The "MC Information System RCS Host" option only has to be set for the NCU to which the TS Adapter is connected at the X127 interface. Modem access can only be permitted by making the appropriate entries at the SINUMERIK Operate on the PCU that is assigned to this NCU and the "Request remote control" function executed. If modem access is permitted, then the service PC implicitly has access via the plant or system network to all other NCUs and PCUs and their operating software. The operating screens to control the remote access can be individually used and set at each station (SINUMERIK Operate in the NCU) for the station in conjunction with the relevant local PLC.

17.11.3 PLC control for remote access

In the PLC user program, you can control external viewers using the following settings.

DB19.DBX1.0	PLCExtViewerReject		
Value	= 0	= 1	
Meaning	Permit remote control	No remote control	
DB19.DBX1.1	PLCExtViewerMode		
Value	= 0	= 1	ignored

Meaning	Operating rights for remote control	Only monitoring for remote control	-
Result	Operating rights for remote control	Only monitoring for remote control	No remote control

17.12 Trace

The trace provides you with an oscilloscope function that allows the following data (signals) to be graphically displayed:

- General NC data
- PLC data

Trace session

The function serves for troubleshooting and resolving faults as well as to analyze the machine and process performance. In a session, data (signals) are recorded briefly before and after an event.

Graphic display

The recorded data is optically displayed in the form of curves, the so-called characteristics graphs.

You have the option of individually setting the recording type.

Saving and downloading data

You can store the recorded data in an XML file. The saved file can then be re-loaded so that it can be evaluated at a later time.

Recording fundamentals: Signals (data)

Signals (data) specify what is to be recorded:

- NC variables (axis data, drive data, channel data, etc.)
- PLC data

Cyclic events



Interpolation clock cycle (IPO clock cycle)

PLC cycle OB1



Position control clock cycle (servo clock cycle)

Non-cyclic events



Starting a session

- Recording using a trigger
- Manually starting and stopping a trace

Analyzing the result

The following functions are available to analyze the result:

- Scaling characteristic graphs
- Zooming characteristic graphs
- Defining a characteristic range using markers

17.12.1 General procedure

Sequence

- Create a trace session file as follows:
 - Select the variable and define its representation type.
 - Set the graphics window, and the save mode
 - Set the trigger property to define the start and end of recording
- Save the recording and its associated properties that have been set
- Analyze the result by making the settings for the display in order to view additional details (e.g. zoom in, scroll, select section, etc.)

See also

Variable for Trace (Page 383)

Trace settings (Page 389)

Trace options (Page 391)

Saving the trace file (Page 381)

Evaluate a trace (Page 392)

17.12.2 Trace session

17.12.2.1 Creating a session file

Procedure



1. Select the "Diagnostics" operating area.



2. Press the menu forward key.



3. Press the "Trace" softkey.



4. Press the "Load trace" and "New" softkeys.



The "New Session" window opens in which the "Session" name is recommended. The file format is automatically specified as ".xml".



5. Supplement the file name or enter a new file name.
6. Select a variable type and press the "OK" softkey.
You return to the "Select Variables for Trace:" window.
The name of the session that has been newly created is displayed in the header.

See also

Variables filter/search (Page 383)

17.12.2.2 Saving the trace file

There are two options of saving a trace session:

- Variables and settings: Only the variables and the settings are saved.
- Variables, settings and recorded values: The variables, settings and the recorded values are saved.

Procedure



1. You have edited an active trace session.



2. Press the "Store trace" softkey.
The "Save Trace: Session ..." window opens and offers the following storage options.



3. Activate the corresponding checkbox and press the "OK" softkey.
The "Store Trace" window opens and specifies the file name in the "Name of trace file:" entry field. The file format is automatically specified as ".xml" and cannot be changed.



4. Press the "OK" softkey to save the session file.
If the session file already exists, then you obtain a corresponding confirmation prompt.

5. Press the "OK" softkey if you wish to overwrite the existing session files.
- OR -



Press the "Save as" softkey if you do not wish to overwrite the existing session file.

A new name is suggested but you can specify a name yourself.

- OR -



Press the "Change directory" softkey if you wish to store the session file in a directory other than in the standard "Trace" directory.

The data tree opens and you can select between the configured drives.



Then press the "New directory" softkey if you wish to create a new directory.

The "New Directory" window opens in which you can assign a directory name.



Press the "OK" softkey.

The "Store Trace" window opens in which the names of the target directory and the session file are displayed.

You can change the name of the session file.



6. To save the session file, press the "OK" softkey one more time.

- OR -

If the session file with this name already exists, then you obtain a corresponding confirmation prompt.

17.12.2.3 Load trace session file

You have the option of loading already saved session data, to restart them or to display the trace that has been recorded.

Settings	Meaning
Variables, settings	You can restart a trace session that has already been created.
Variables, settings and recorded values	You can view the result of a trace session, change the variable and if required, restart.

Precondition

A session file of recorded traces is available.

Procedure



1. You have selected the "Trace" function.



2. Press the "Load trace" softkey.

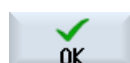
The "Load Session File: Please Select File" window opens.



3. Select the desired session file from the appropriate directory, e.g. "Trace", and press the "Display trace" softkey.

The "Load trace: session..." is opened.

4. Select the checkbox "Variables, settings", if you wish to immediately start the trace session.



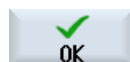
5. Press the "OK" softkey and the "Start trace" softkey to restart the trace.



- OR -

Select the checkbox "Variables, settings and recorded values", if you wish to display the trace session.

Press the "OK" softkey.



The data of the selected session are loaded displayed in the "Trace" window.

17.12.3 Variable for Trace

17.12.3.1 Variables filter/search

Selection of variables

To limit the selection of variables, set a filter and/or enter an initial value for the search. The following filters are available for selection:

Filter	
Drive parameters	Displaying the drive parameters
System variables	Displays the system variables
NC	Display of NC, axis and channel signals as well as signals of the mode group in the OPI notation.
Axis	
Channel	
Mode groups	
Servo	Displays the servo-trace signals
PLC	Displays the PLC signals
PLC alarms	Displays the manufacturer-specific PLC alarm signals
All (no filter)	Displays all signals without filter: NC, axis and channel setting data as well as signals of the mode group in the OPI notation.

Note**"Filter/Search" when inserting variables**

The start value for "Filter/Search" of variables differs.

For example, to insert the variable \$R[0], set "Filter/Search":

- The start value is 0, if you filter according to "System variables".
 - The start value is 1, if you filter according to "All (no filter)". In this case, all signals are displayed and shown in the OPI notation.
-

Procedure

For a trace session, in the window "Selected Variables for Trace: Session ..." set the variables whose signals are to be traced.



1. In the "Select variables for trace: Session..." window, a trace session is displayed.

2. Position the cursor in the "Variable" column in an empty line.

- OR -

Position the cursor in the "Variable" column on a variable that you wish to replace by another.



3. Press the "Insert variable" softkey.
The "Select Variable" window opens.

Note:

When selecting a "Drive parameter", firstly select the drive object (DO) and then insert the desired parameter from the list of corresponding parameters.



4. Press the "Filter/search" softkey to start a specific search.

In the "Filter" selection box, select a variable area.

- OR -

Directly enter the variable into the "Search" entry field.

The cursor jumps to the term that is found.

Press the "Delete search" softkey if you wish to terminate the search.



5. Press the "Add" softkey to enter the required variable into a trace session.
The "Variable Attributes" window opens and you can select the appropriate value (e.g. channel data, drive bus no.) from a combobox.

- OR -



Press the "Replace" softkey to replace an already selected version by another one.



6. Press the "OK" softkey to add or replace the variable.
Each variable is automatically allocated a reference number, which is displayed in the upper section of the window. This number is opened automatically allocated. If a version is removed, then the following variables are re-numbered.
7. Now, select the desired settings for display, such as color, pen, qty. event etc.

Editing variables of a trace session

Function	Meaning
Ctrl + X or right-click, shortcut menu "Cut"	Press the key combination or open the shortcut menu in order to cut out a variable.
Ctrl + C or right-click, shortcut menu "Copy"	Press the key combination or open the shortcut menu in order to copy a variable.
Ctrl + V or right-click, shortcut menu "Paste"	Press the key combination or open the shortcut menu in order to insert a variable.

17.12.3.2 Selecting attributes of a variable

Attributes

For a trace session, select the variables and assign the corresponding attributes:

Column	Meaning
Variable	Address of the variables

Attributes	Meaning
Comment	- Name of the selected variables - A general description for the variables.
Color	Color selection, the representation of the characteristic graphs. When creating a variable, the next color from the available palette is automatically used.

Attributes	Meaning
Pen	Line type selection to display the characteristic graphs. • No line • Solid line • Dashed line • Dash-dot line • Dash-dot-dot line • Dotted line • Stepped line XY • Stepped line YX • "*" line" (line made up of * symbols) • + line • X line
Qty.	Defines whether the characteristic of the variables is displayed in the "Trace" window. If the checkbox is not selected, then no characteristic graphs are displayed.
Events	Event selection which triggers that the signal is traced: • Interpolation clock cycle (IPO clock cycle) • Interpolation cycle (IPO2) • Position control clock cycle (servo clock cycle) • PLC cycle OB1 • Drive trace rate • Start geometry axis/direction change • Stop geometry axis • Machine axis start / direction change • Machine axis stop • NC start (program runs) • NC start (program ran) • Start of the data recording

Attributes	Meaning
	• Block start, type 1 or block end (all program levels, without intermediate blocks) • Block start, type 2 or block end (all program levels, with intermediate blocks) • Block start, type 3 or block end (all main programs, without intermediate blocks) • Block start or block end, block search (all program levels, without intermediate blocks) • Geo axis start / direction change (2nd event) • Geo axis stop (2nd event) • Block start, type 2 (2nd event: all program levels, without intermediate blocks) • Block end, type 2 (2nd event: all program levels, with intermediate blocks) • Block start, block advance, type 1 (2nd event) • OEM test event 1 (non-cyclic, block advance) • OEM test event 2 (non-cyclic, main run)
	• Activate/deactivate synchronized action • Synchronized action initiated (condition fulfilled) • Signaled alarm • Cancel softkey pressed • Program level change • Block end (Interpreter) • Trace end (last event!) • Start trace
	• WRTPR part program command • WRTPR part program command (block search) • Tool change • Cutting edge change • Tool change (block search) • Cutting edge (block search) • Start trigger initiated • Stop trigger initiated • Block end (block advance) • Block end (2nd event: Block search) • WRTPR part program command (block advance) • Tool change (block advance) • Cutting edge (block advance)
Event channel	This means that the channel can be defined in SINUMERIK in which a specific event occurs. For events which are not channel-specific, the field remains empty.

Attributes	Meaning
Bit mask	In the case of an integer variable, a bit mask can be specified in this field. For a bit mask, the particular signal value is AND'ed before being displayed. After the mask has been applied, all of the selected bits are moved to the right so that it looks as if the bit or the bits all start with bit zero. This means, if all bits with the exception of bit 7 have been masked, then the integer number that would have been obtained would either have a value of 0 or 1, however, not 0 or 128. If all of the bits with the exception of bit 7 and 0 have been masked, the resulting integer number would either have the value 0, 1, 2 or 3, however, not 0, 1, 128 or 129.
Decimal places	This setting is used to define how many places to the right of the decimal point are displayed at the axis identifiers.
Coord. axis	The coordinate axes are displayed to the left or right in the graphic window or there is no display.
Display Y	Value input or 0
Scale factor	Defines the scale.
Units	Displays the measurement unit, e.g. mm/min. The system specifies this and it cannot be changed.

17.12.3.3 Displaying details of a variable

You can display the most important information and settings of the display at a glance in the detail window, i.e. you do not have to use the scroll bar for example to get to the "Event" column.

Displayed information

- Variable address
- Comment with a description of the variables
- Events
- Channel, axis, access level, mode group, etc.
- Smaller graphic display with the event; when this event occurs the characteristic graphs are recorded as well as their settings, such as color, line type, etc.

Procedure



1. You are in the "Select Variables for Trace: ..." window. ...".
2. Position the cursor to the required variable.
3. Press the "Details" softkey.
A subscreen with the most important information as well as the graphic display is shown in the lower half of the window.
Press the "Details" softkey again to display the subscreen.

17.12.4 Trace settings

17.12.4.1 Trace settings (PLC, NC, servo)

In the "Trace Settings (PLC, NC, Servo)" window, you define the conditions for starting and stopping the trace session.

Condition for Start trace	
With "Start trace" softkey	The recording is started immediately after pressing the "Start trace" softkey.
If variable	The session is activated by a variable. The desired variable is selected via the "Insert variable" or "Quick list" softkey.
Condition and value	After the session has been activated, this defines at which value the trace recording is started. The following values are offered: =, ≠, >, >=, <, <=, ascending, descending, changed.
Data acquisition [s]	
Specifies where the recorded data is saved: - With "On NCK", the recorded data is saved in the NC work memory until the recording is stopped. Only then is the session file saved on the CompactFlash card. - With "CF card" or "On hard disk", the data is saved during the recording on the CompactFlash card or hard disk (e.g. of the PCU). As a consequence, more signals can be processed.	
Condition for Stop trace	
With "Stop trace" softkey	The recording is stopped after pressing the "Stop trace" softkey.
If the time has expired	Defines the period which after expiry, the trace recording is exited.
If the memory is full	The trace session is terminated when the specified data volume (KB) is reached.
If variable	The recording is stopped by a variable. The desired variable is selected via the "Insert variable" or "Quick list" softkey.
Condition and value	After the session has been activated, this defines at which value the trace recording is exited. The following values are offered: =, ≠, >, >=, <, <=, ascending, descending, changed.
Delay time [s]	This defines how long the trace still runs after the session has been exited.
Overwrite at memory limit (KB)	Defines from which memory limit the data is overwritten (ring buffer).

17.12.4.2 Trace settings (drive)

In the "Trace Settings (Drive)" window, you define the conditions for starting and stopping the trace session:

Condition for Start trace	
With "Start trace" softkey	The recording is started immediately after pressing the "Start trace" softkey.

If variable	The recording is activated after pressing the "Start trace" softkey when the selected variable clearly satisfies the condition and the specified value. The following conditions are offered: • Positive edge • Negative edge • With bit mask The "Bit mask" softkey is displayed when the "Bit mask" condition is selected in order to select the desired bits. Irrespective of the maximum value (hexadecimal) which is possible through the bit mask, a "Value" (decimal) can be entered at which the trace recording starts precisely. Example: The trace recording is started when the following bits are set: Bit 0 = 0; bit 1 = 1; bit 2 = 0; bit 3 = 1 <table><tr><th>Bit mask = fH</th><th>Condition</th><th>Value (decimal)</th></tr><tr><td>Variable: CU/ r722/3.3.1</td><td>With bit mask ☒ Bit 0 ☒ Bit 1 ☒ Bit 2 ☒ Bit 3</td><td>10</td></tr></table>			Bit mask = fH	Condition	Value (decimal)	Variable: CU/ r722/3.3.1	With bit mask ☒ Bit 0 ☒ Bit 1 ☒ Bit 2 ☒ Bit 3	10
Bit mask = fH	Condition	Value (decimal)							
Variable: CU/ r722/3.3.1	With bit mask ☒ Bit 0 ☒ Bit 1 ☒ Bit 2 ☒ Bit 3	10							
Preprocessing time [s]:	The input of a preprocessing time corresponds to the functional parameter p4722 "Trace trigger delay": • A positive preprocessing time starts the recording before the trigger event. • A negative preprocessing time starts the recording after the trigger event. When entering a preprocessing time, the entire duration of the recording must be taken into account: • If the preprocessing time is greater than the entire duration of the recording, the entry is rejected. • If too large a preprocessing time is entered, the trigger event may not be recorded.								
Data storage on CU									
Drive trace rate [μs]:	Enter a factor with the "Sampling rates" softkey to set the effective sampling rate. With a higher sampling rate, the recording duration is also longer.								
Condition for Stop trace									
If the time has expired	Defines the period which after expiry, the recording is exited.								
If the memory is full	The recording is terminated when the specified data volume (KB) is reached.								

17.12.4.3 Trace options

You define the following properties of a trace session in the "Trace - Options" window.

- Setting the graphic window
 - Displaying main grid lines: You define whether a horizontal grid is displayed and in which color.
 - Displaying grid sub-lines: You define whether a vertical grid should be displayed and in which color.
 - Background: You select the background color of the graphic window.
- Defining storage options
 - Activate the "Confirm file overwrite" checkbox in order to receive a confirmation prompt each time that a trace session is to be overwritten. If the checkbox is deactivated, the trace session is overwritten without a confirmation prompt.
 - Activate the "Export data in csv file" checkbox in order to also save the data as a csv file in addition to the xml session file. This file only serves to export the data and cannot be read in again.

Procedure



1. You are in the "Select Variables for Trace: ..." window. ... and a trace session has been selected.
2. Press the "Options" softkey.
The "Trace - Options" window opens.
3. Select the desired settings and press the "OK" softkey in order to accept the settings.

17.12.4.4 Starting the trace

In the "Trace" window, you obtain the curve with the data of the variables selected for a trace session.

The diagram comprises two Cartesian coordinates:

- Time axis
The horizontal x axis supplies the time values.
- Value axis
The vertical y axis displays the resolution fineness (amplitude).

Recording the trace session

- Manually starting and stopping a trace
- Starting and stopping recording using a trigger

Procedure



1. You have listed the desired variables for a trace section.
2. Press the "Display trace" softkey.
The "Trace" window is opened. The recording is loaded and the procedure is displayed in progress window.
3. Press the "Start trace" softkey if you wish to immediately start recording.

- OR -
The recording is triggered by the trigger condition.
4. If you wish to manually end the recording, press the "Stop trace" softkey.

- OR -
The recording is stopped by the trigger condition.

17.12.5 Evaluate a trace

17.12.5.1 Setting trace views

In order to permit a precise evaluation of the recorded signals, you have the option to edit the view as well as the display of the characteristic graphs:

- Selection/legend
 - Show legend: In addition to the graphic display, the variable data are displayed as comment, numbered and in color.
 - Make a selection: You define which variables are displayed in the graphic window.
- Scale
 - Specify scaling factors for minimum and maximum X and Y value
- Zoom
 - Increasing or decreasing the section
- Cursors: The cursor display provides functions that allow one, two or both curves to be precisely measured.
 - Cursor A
 - Cursor B
 - Both cursors
 - Snap to curve: The cursor precisely moves along the curve
 - Point mode: Cursor jumps directly to various values.
 - High point mode: The cursor jumps from one peak value to the next.
 - Low point mode: Cursor jumps to the lowest values.

Precondition

The "Trace" graphic window is opened and the trace has been recorded.

Editing curves



You jump from <TAB> one variable to the next using the <TAB> key.



You jump back using the <Ctrl> and <TAB> keys.



With the cursor keys, you select marks, move to individual values and can define a zoom range.



17.12.5.2 Selecting a variable

You have the option of displaying the selected variables as legend and to make a selection using a checkbox.

Legend
Numbering the variables
Axis identifier
Comment of the variables

Displaying a legend and selecting a variable



1. A recording is opened in the "Trace" graphics window.



2. Press the "Legend / selection" softkey.
The graphic window is halved. The variables are displayed as comment and with checkbox in the left-hand section of the window.
3. Activate or deactivate the required variable using the checkbox to display it as curve.



4. Press the softkey again to hide the legend area again.

17.12.5.3 Scaling the display

The scaling allows you to make a precise measurement of selected variables from the X time axis as well as from the Y value axis. You have the option of saving the larger or smaller display in a user-defined way

Setting	Meaning	
X minimum or X maximum	Highest or lowest value of the X time axis. The values are used for recording when the trace is started again.	
	Fixed scale, X axis	Values of the X time axis are kept.
	Strip chart recorder mode	The values of the X time axis are used while recording.
Y minimum or Y maximum	Defining the highest or lowest value of the Y value axis (amplitude).	
Offset or resolution	Defining the rms values (mean values) of the amplitudes. The scale is changed by a factor of 10.	

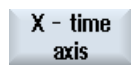
Scaling the time and value axis



1. A recording is opened in the "Trace" graphics window.



2. Press the "Scale" softkey in order to scale the display.



The "Select X-Y Scaling" input window opens and the values of the X time axis are displayed.



3. Press the "X time axis" softkey again if you wish to hide the values.

- OR -



Press the "Y selected curves" softkey to display the values of the Y value axis.



Press the softkey again if you wish to hide the values.

- OR -

You have selected both axes.



4. Press the "Scale +" or "Scale -" softkey until the desired scaling is reached.



- OR -

Enter the scaling values directly into the input window and select the appropriate property.



5. Press the "Scale" softkey to save the settings.

- OR -



Press the "Adapt selected" softkey to display the selected characteristic graphs in the full graphic window.



Press the "Fit one" softkey to display the selected characteristic graphs in the full graphic window.

17.12.5.4 Zooming the display

Using the zoom function, you have the option of viewing selected sections of curves in detail. After you have defined an area or a section, you can increase or decrease the size of it (zoom in or zoom out).

Procedure



1. A recording is opened in the "Trace" graphics window.



2. Press the "Zoom" softkey if you wish to zoom-in or zoom-out on the characteristic graphs.



3. The "X time axis" or "Y value axis" softkey is selected.



4. Press the "Adapt individually" or "Adapt all" softkey if you wish to display all selected characteristic graphs in a separate area positioned one above the other.

or



- OR -



Press the "Adapt together" softkey if you wish to adapt all of the characteristic graphs to the graphic window.

As all of the curves fill the complete window area, they are displayed superimposed on one another.

- OR -



Press the "Individually adapt X" softkey if you wish to normalize the X time axis only in the graphic window.

- OR -



Press the "Individually adapt Y" or "Adapt all" softkey if you only wish to scale the Y value axis in the graphic window.

or



- OR -



Press the "Adapt Y together" softkey if you wish to adapt the value axis of all characteristic graphs to the graphic window.
As all curves fill the complete window area, they are displayed superimposed.



5. Press the "Zoom +" or "Zoom -" softkey until the desired display size is reached.



Zoom in/zoom out area



Press the "Zoom area" softkey to define a specific section.
A frame is displayed.



Press the "Zoom +" or "Zoom -" softkey to change the frame step-by-step.



Press the "OK" softkey to display the zoom area.

17.12.5.5 Position the cursor

Using the cursor, you have the option of determining the precise values from a defined section.
To do this, the cursor is positioned directly on the curve.

Procedure



1. A recording is opened in the "Trace" graphics window.



2. Press the "Cursor A" softkey.
The reference number, color of the variables and the color of the coordinate crossing point are displayed on the softkey.



3. Position "(A)" at the position on the curve where you wish to start the measurement.

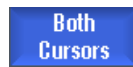
The coordinates are specified in a tooltip and in the status line.



4. Press the "Cursor B" softkey.
The reference number, color of the variables as well as the color of the coordinate crossing point are also displayed on this softkey.

Position "(B)" at the position on the curve where you wish to end the measurement.





5. Press the "Both cursors" softkey to display the area between A and B. The coordinates of A and B as well as the difference " Δ " are displayed in the status line.



Press the active "Cursor A" and "Cursor B" softkeys again to deselect the position.



When re-selected, the positions are reset.



Press the "Crop to screen" softkey to display the area, defined by "Cursor A" and "Cursor B", in the complete graphic window.



Press the "Snap to waveform" softkey to move the cursor along the curve step-by-step.

17.12.5.6 Acquiring measured values

You have the option of directly determining the following values of a curve:

- Peak-to-peak value
- Peak-to-valley value

Procedure



A recording is opened in the "Trace" graphics window.



Press the "Point Mode" softkey to continuously move the cursor along the curve.



Press the "Peak Mode" softkey to position the cursor at the highest value.



Press the "Valley Mode" softkey to position the cursor at the lowest value.



If you press the "Peaks" softkey, the cursor only goes to the peak-peak values.



If you press the "Minimum" softkey, the cursor only goes to the peak-valley values.

17.13 PROFIBUS/PROFINET and AS-i bus diagnostics

17.13.1 PROFIBUS/PROFINET

With the PROFIBUS/PROFINET diagnostics, you obtain a quick overview from which, when required, you can call status and detailed information for external DP master systems and PROFINET IO systems.

Precondition for detailed diagnostics

When using HMI PRO RT, the diagnostics can be extended by additional information (e.g. module and Order No.). This data involves offline data, which can be extracted from the associated Step7 project.

- You can create offline data using the HMIPRO CS program.
- You transfer data to the control using the HMIPRO program.



Software option

For faulted modules, in order that in addition to the status information, detailed information can also be displayed, you now require the following option:
"Operating software SW HMI PRO sl RT".

In order that detailed information can be displayed for PROFINET-IO systems, the PLC block FB_SL_COM from the HMI PRO block library must be used. Use DB449 as the associated instance DB.

Note

Detailed information for DP master systems is also available without using FB_SL_COM.

Diagnostics information from the DP master systems

You can configure the following versions:

1. Software option not available and HMIPRO offline data not available
2. Software option available and HMIPRO offline data not available
3. Software option available and HMIPRO offline data available

17.13.2 Displaying PROFIBUS/PROFINET diagnostics

Diagnostics information from the DP master systems

From the user interface, you can display the fieldbus diagnostics known from HMIPRO. The module state is displayed using diagnostic symbols.

The following diagnostics information is displayed depending on the particular version that you have configured:

Information for all versions		
Status		Module has a fault
		Module has failed
		Module is OK
		Module is suppressed (this is only possible with HMI PRO RT)
PROFIBUS address		PROFIBUS address data

Additional information for the second version	
PROFIBUS/PROFINET module or channel diagnostics	
Device diagnostics	For Siemens DP/DP coupler and diagnostics repeater

Additional diagnostic information for the third version	
Name	Module designation from the HMI offline data
Name	Module name from the HMI offline data
Order no.	Order number of the module from the HMI offline data
IP address	IP address of the module from HMI offline data (only for PROFINET)
Graphic display of the complete hardware configuration in a dedicated window	

Procedure



1. Select the "Diagnostics" operating area.



2. Press the "Menu forward" key twice and the "PB/PN diag." softkey. The "PROFIBUS/PROFINET Diagnostics" window opens.



3. Select the required DP master system.
The complete diagnostics information is displayed.



4. Press the "Available devices" softkey to display the diagnostics data of the available modules.

- OR -



Press the "Suppressed devices" softkey to display the diagnostics data of the suppressed modules.

- OR -


 Only failed

Press the "Failed devices" softkey to display the diagnostics data of the failed modules.

- OR -


 Only faulted

Press the "Faulty devices" softkey to display the diagnostics data of the faulty modules.

17.13.3 Configuring the AS-i bus

You obtain a fast overview of the existing AS-Interface network using the AS-Interface (AS-i) diagnostics. Permanent monitoring and an error search function provide you with the following information:

- Status flags acc. to the AS-i specification
- States of the AS-i slaves

Precondition

- For AS-i diagnostics, it is absolutely necessary to use the FB_SL_COM PLC block from the HMI PRO block library.
- The configuration file "sltIprodiaghmi.ini" must be available.

AS-i components

The following AS-i components are supported:

CP142-2	(Type ID 1)
CP343-2	(Type ID 2)
DP/AS-i link	(Type ID 3)
DP/AS-i link 20E	(Type ID 4)
DP-AS-i link Advanced	(Type ID 6)

Diagnostic information

All of the AS-i components to be diagnosed must be parameterized in the "sltIprodiaghmi.ini" configuration file.

Entry	Description
[SKO]	
ComboCount=x	Number of entries (x) in the selection list
[SKO_Index0]	1. Entry
DiagType= 6	Type ID of the AS-i component (refer above)
TypeSection=DP-AS-i-Link-Advanced 1	Designation of the AS-i component, which should be output in the selection list

Entry	Description
Logical_Address=43	E address of the AS-i component from the hardware configuration
RequestDB=449	Instance DB of the function block "FB_SL_COM" being used

Example of the "sltprodiaghmi.ini" with 5 components

```
[SK0]
ComboCount=5

[SK0_Index0]
DiagType= 1
TypeSection=CP142-2
Logical_Address=43
RequestDB=449

[SK0_Index1]
DiagType= 2
TypeSection=CP343-2
Logical_Address=75
RequestDB=449

[SK0_Index2]
DiagType= 3
TypeSection=DP-AS-i-Link
Logical_Address=75
RequestDB=449

[SK0_Index3]
DiagType= 4
TypeSection=DP-AS-i-Link-20E
Logical_Address=75
RequestDB=449

[SK0_Index4]
DiagType= 6
```

```
TypeSection=DP-AS-i-Link-Advanced
```

```
Logical_Address=75
```

```
RequestDB=449
```

Procedure

1. Copy the "sltprodiaghmi.ini" file from the `/siemens/sinumerik/hmi/template/cfg` directory.
2. Place the file in the directory: `/oem/sinumerik/hmi/cfg` or `/user/sinumerik/hmi/cfg`.
If the file is already available in the directory, then just supplement this by the corresponding entries.
3. Open the file and enter the corresponding data of the AS-i components.
4. Save and close the file.
The diagnostics information is displayed on the user interface.

17.13.4 Displaying AS-i bus diagnostics

Diagnostics information from AS-i components

Using the user interface, you have the option of displaying information about the networking of binary sensors and actuators (AS-i).



Software option

Additional error counter can be displayed for the component DP-AS-i Link Advanced if the option "Operating SW HMI PRO sl RT" is set.

Diagnostic information

Flags acc. to the AS-i specification	Description
Offline	Offline operating state
Internal	Internal operating state
EEPROM o.k.	Internal EEPROM okay
Auto-Addr.progr. enabled	Automatic address assignment enabled
I/O fault	Peripheral fault present
Master Power On - Start	Starting after Master Power On
Offline ready	Offline phase completed
AS-i cable voltage too low	AS-i cable voltage too low
CP in normal operation	CP in normal operation
Operating mode	Operating mode state
Auto.-Addr.progr. executed	Automatic address assignment executed
Auto-Addr.progr. possible	Automatic address assignment possible

Flags acc. to the AS-i specification	Description
AS-i slave with addr.=0 available	Slave with address 0 available
Reference config. = actual config.	Target configuration equal to actual configuration

State	Marking
Slave OK	Green
Slave fault	Red
Slave is available, but not configured	Yellow

Procedure



1. Select the "Diagnostics" operating area.



2. Press the menu forward key twice and the "AS-i diag." softkey. The "Diagnostics AS-Interface" window is opened.



3. Select the required DP AS-interface.
The complete diagnostics information is displayed.
The values in italics indicate an error counter > 0.

OEM-specific online help

18.1 Overview

In addition to the existing extensive online help, you also have the option of generating a manufacturer-specific online help and then linking this into SINUMERIK Operate.

This online help is generated in the HTML format, i.e. it comprises HTML documents that are linked with one another. The subject being searched for is called in a separate window from a contents or index directory. Similar to a document browser (e.g. Windows Explorer), a list of possible selections is displayed in the left-hand half of the window and when you click on the required subject, the explanation is displayed in the right hand half of the window.

Context sensitive selection of online help pages is not possible.

Procedure

1. Generating HTML files
2. Generating a help book
3. Integrating the online help in SINUMERIK Operate
4. Saving help files

Other application cases

Online help for the following OEM-specific expansions can be created and used to supplement the SINUMERIK Operate online help system:

- Online help for **cycles and/or M functions** of the machine manufacturer which extend the programming options for SINUMERIK control systems. This online help is called in just the same way as the SINUMERIK Operate online help "Programming".
- Online help for OEM-specific **variables** of the machine manufacturer. This online help is called from the variable view of SINUMERIK Operate.

Programming online help

You can use the "SINUMERIK HMI programming package sl" for additional options for configuring the online help. Using this programming package, it is possible to develop high-level language applications in the C++ programming language for SINUMERIK Operate on the NCU 7x0.

Note

The "SINUMERIK HMI programming package sl" must be ordered separately as a software option. The associated documentation is provided together with the programming package.

18.2 Generating HTML files

Generating help files in the HTML format. It is possible to save all information in a single HTML file or to distribute the information over several HTML files.

You can assign the file names yourself, however, you must observe the following:

- References within HTML files should always be specified with relative paths. Only then can it be ensured that the references function in precisely the same way on both the development computer as well as on the target system.
- If jumps are to be made to certain points within an HTML file per link, then so-called anchors must be defined for this purpose.

Example of an HTML anchor:

```
<a name="myAnchor">This is an anchor</a>
```

- The contents of HTML documents must be saved with the UTF-8 coding. Only then is it guaranteed that the HTML documents are correctly displayed in all of the country languages supported by SINUMERIK Operate.
- The following sub-sets of the HTML functional scope are supported:

HTML tags

Tag	Description	Comment
a	Anchor or link	Supported attributes: href and name
address	Address	
b	Bold	
big	Larger font	
blockquote	Indented paragraph	
body	Document body	Supported attributes: bgcolor (#RRGGBB)
br	Line break	
center	Centered paragraph	
cite	Inline citation	Same effect as tag i
code	Code	Same effect as tag tt
dd	Definition data	
dfn	Definition	Same effect as tag i
div	Document division	The standard block attributes are supported
dl	Definition list	The standard block attributes are supported
dt	Definition term	The standard block attributes are supported
em	Emphasized	Same effect as tag i
font	Font size, family, color	Supported attributes: size, face, and color (#RRGGBB)
h1	Level 1 heading	The standard block attributes are supported
h2	Level 2 heading	The standard block attributes are supported
h3	Level 3 heading	The standard block attributes are supported
h4	Level 4 heading	The standard block attributes are supported
h5	Level 5 heading	The standard block attributes are supported
h6	Level 6 heading	The standard block attributes are supported
head	Document header	

Tag	Description	Comment
hr	Horizontal line	Supported attributes: width (can be specified as absolute or relative value)
html	HTML document	
i	Italic	
img	Image	Supported attributes: src, width, height
kbd	User-entered text	
meta	Meta-information	
li	List item	
nobr	Non-breakable text	
ol	Ordered list	The standard attributes for lists are supported
p	Paragraph	The standard block attributes are supported (default setting: left-aligned)
pre	Preformatted text	
s	Strikethrough	
samp	Sample code	Same effect as tag tt
small	Small font	
span	Grouped elements	
strong	Strong	Same effect as tag b
sub	Subscript	
sup	Superscript	
table	Table	Supported attributes: border, bgcolor (#RRGGBB), cellpadding, cellspacing, width (absolute or relative), height
tbody	Table body	No effect
td	Table data cell	The standard attributes for table cells are supported
tfoot	Table footer	No effect
th	Table header cell	The standard attributes for table cells are supported
thead	Table header	This is used to print tables that extend over several pages
title	Document title	
tr	Table row	Supported attributes: bgcolor (#RRGGBB)
tt	Typewrite font	
u	Underlined	
ul	Unordered list	The standard attributes for lists are supported
var	Variable	Same effect as tag tt

Block attributes

The following attributes are supported by the tags div, dl, dt, h1, h2, h3, h4, h5, h6, p:

- align (left, right, center, justify)
- dir (ltr, rtl)

Standard attributes for lists

The following attributes are supported by tags ol and ul:

- type (1, a, A, square, disc, circle)

Standard attributes for tables

The following attributes are supported by tags td and th:

- width (absolute, relative, no-value)
- bgcolor (#RRGGBB)
- colspan
- rowspan
- align (left, right, center, justify)
- valign (top, middle, bottom)

CSS properties

The following table includes the supported CSS functional scope:

Property	Values	Description
background-color	<color>	Background color for elements
background-image	<uri>	Background image for elements
color	<color>	Foreground color for text
text-indent	<length>px	Indent the first line of a paragraph in pixels
white-space	normal pre nowrap pre-wrap	Defines how whitespace characters are handled in HTML documents
margin-top	<length>px	Width of the upper edge of the paragraph in pixels
margin-bottom	<length>px	Width of the lower edge of the paragraph in pixels
margin-left	<length>px	Length of the left hand edge of the paragraph in pixels
margin-right	<length>px	Width of the righthand edge of the paragraph in pixels
vertical-align	baseline sub super middle top bottom	Vertical alignment for text (in tables, only the values middle, top and bottom are supported)
border-color	<color>	Border color for text tables
border-style	none dotted dashed dot-dash dot-dot-dash solid double groove ridge inset outset	Border style for text tables
background	[<'background-color'> <'background-image'>]	Short notation for background property
page-break-before	[auto always]	Page break before a paragraph/table
page-break-after	[auto always]	Page break after a paragraph/table
background-image	<uri>	Background image for elements

Supported CSS selectors

All CSS 2.1 selector classes are supported with the exception of so-called pseudo selector classes such as :first-child, :visited and :hover.

18.3 Generating the help book

The help book is an XML file in which the structure of the online help is defined. In this file, you define:

- HTML documents
- Contents and subject index

Syntax for the help book

Tag	Number	Meaning						
HMI_SL_HELP	1	Root element of the XML document						
I-BOOK 	+	Identifies a help book. The name can be freely selected under the constraint that no name predefined by the system is used (such as sinumerik_alarm_plc_pmc). In the example, the name of the help book is "hmi_myhelp" Attributes: <table><tr><td>ref</td><td>Identifies the HTML document that is displayed as the entry page for the help book.</td></tr><tr><td>title</td><td>Title of the help book that is displayed in the table of contents.</td></tr><tr><td>helpdir</td><td>Directory that contains the online help of the help book.</td></tr></table>	ref	Identifies the HTML document that is displayed as the entry page for the help book.	title	Title of the help book that is displayed in the table of contents.	helpdir	Directory that contains the online help of the help book.
ref	Identifies the HTML document that is displayed as the entry page for the help book.							
title	Title of the help book that is displayed in the table of contents.							
helpdir	Directory that contains the online help of the help book.							
I-ENTRY 	*	Section of the online help Attributes: <table><tr><td>ref</td><td>Identifies the HTML document that is displayed as entry page for the section.</td></tr><tr><td>title</td><td>Title of the section that is displayed in the table of contents.</td></tr></table>	ref	Identifies the HTML document that is displayed as entry page for the section.	title	Title of the section that is displayed in the table of contents.		
ref	Identifies the HTML document that is displayed as entry page for the section.							
title	Title of the section that is displayed in the table of contents.							
II-INDEX_ENTRY 	*	Subject (keyword) to be displayed Attributes: <table><tr><td>ref</td><td>Identifies the HTML document that is jumped to for this subject index entry.</td></tr><tr><td>title</td><td>Title of the subject that is displayed in the subject index.</td></tr></table>	ref	Identifies the HTML document that is jumped to for this subject index entry.	title	Title of the subject that is displayed in the subject index.		
ref	Identifies the HTML document that is jumped to for this subject index entry.							
title	Title of the subject that is displayed in the subject index.							

The following applies for the "Number" column:

* means 0 or more

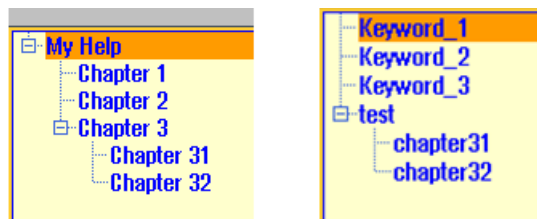
+ means 1 or more

Example for a help book

In the following example, the structure of a help book with the "My Help" name is described. Further, it forms the basis for the table of contents and subject index.

```
<?xml version="1.0" encoding="utf-8"?>
<HMI_SL_HELP language="en-US">
  <BOOK ref="index.html" title="My Help" helpdir="hmi_myhelp">
    <ENTRY ref="section_1.html" title="Section 1">
      <INDEX_ENTRY ref="section_1.html#Keyword_1" title="Keyword_1"/>
      <INDEX_ENTRY ref="section_1.html#Keyword_2" title="Keyword_2"/>
    </ENTRY>
    <ENTRY ref="section_2.html" title="Section 2">
      <INDEX_ENTRY ref="section_2.html#Keyword_3" title="Keyword_3"/>
    </ENTRY>
    <ENTRY ref="section_3.html" title="Section 3">
      <ENTRY ref="section_31.html" title="Section 31">
        <INDEX_ENTRY ref="section_31.html#test" title="test;section31"/>
      </ENTRY>
      <ENTRY ref="section_32.html" title="Section 32">
        <INDEX_ENTRY ref="section_32.html#test" title="test;section32"/>
      </ENTRY>
    </ENTRY>
  </BOOK>
</HMI_SL_HELP>
```

The book comprises three sections, whereby the third section has two subsections. The various subject words (keywords) are defined within the section.



You have the following three options to format the subject index:

1. Single entry:

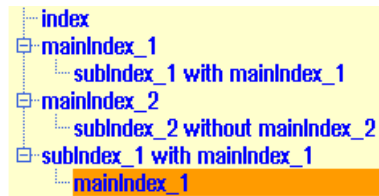
```
<INDEX_ENTRY ...title="index"/>
```

2. Two two-stage entry, whereby each title has a main and a subentry. Separate the entries from one another using a comma.

```
<INDEX_ENTRY ...title="mainIndex_1,subIndex_1 with mainIndex_1"/>
```

3. Two-stage entry, whereby the first title is the main entry and the second title is the subentry. Separate the entries from one another using a semicolon.

```
<INDEX_ENTRY ...title="mainIndex_2;subIndex_2 without  
mainIndex_1"/>
```



18.4 Integrating the online help in SINUMERIK Operate

If you wish to integrate the generated help book into the online help system of SINUMERIK Operate, then you require the "slhlp.xml" file.

Format description of the "slhlp.xml"

Tag	Number	Meaning
CONFIGURATION	1	Root element of the XML document. Designates that this involves a configuration file.
I-OnlineHelpFiles	1	Introduces the section about the online help books.
II-<help_book>	*	Introduces the section of a help book.
III-EntriesFile	1	File name of the help book with the list of contents and subject (keyword) entries.
III		Attributes:
III		value Name of the XML file
III		type Data type of the value (QString)
III-Technology	0, 1	Specifies the technology that applies to the help book. "All" applies to all technologies.
III		If the help book applies to several technologies, then the technologies are listed separated by comma.
III		Possible values: All, Universal, Milling, Turning, Grinding, Stroking, Punching
III		Attributes:
III		value Technology data
III		type Data type of the value (QString)
IIII		

Tag	Number	Meaning
III -DisableSearch III III III III	0, 1	Disable the subject (keyword) search for the help book. Attributes: value true, false type type, data type of the value (bool)
III-DisableFullTextSearch III III III III	0, 1	Disable the full text search for the help book. Attributes: value true, false type type, data type of the value (bool)
III-DisableIndex III III III III	0, 1	Disable the subject index for the help book. Attributes: value true, false type type, data type of the value (bool)
III-DisableContent III III III III	0, 1	Disable the table of contents for the help book. Attributes: value true, false type type, data type of the value (bool)
III-DefaultLanguage III III III III III	0, 1	Abbreviation for the language that should be displayed if the actual country language is available for the help book. Attributes: value chs, deu, eng, esp, fra, ita, ... type Data type of the value (QString)

The following applies for the "Number" column:

* means 0 or more

Example of a file "slhlp.xml"

The help book "hmi_myhelp.xml" is made known to SINUMERIK Operate in the following example.

The subject index has not been activated for the help book.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!DOCTYPE CONFIGURATION>
<CONFIGURATION>
  <OnlineHelpFiles>
    <hmi_myHelp>
      <EntriesFile value="hmi_myhelp.xml" type="QString"/>
      <DisableIndex value="frue" type="bool"/>
    </hmi_myHelp>
  </OnlineHelpFiles>
</CONFIGURATION>
```


18.5 Saving help files

Saving help files in the target system

1. Open the `/oem/sinumerik/him/hlp` directory and create a new folder for the required language. For this purpose, use the specified language code.
It is mandatory that the folder names are written in lower-case letters.
For instance, if you are integrating a help function for German and English, then create the "deu" and "eng" folders.
2. Place the help book, e.g. "hmi_myhelp.xml" in the "deu" and "eng" folders.
3. Copy the help files into the directories, e.g. `/oem/sinumerik/him/hlp/deu/hmi_myhelp` for German and `/oem/sinumerik/him/hlp/eng/hmi_myhelp` for English help files.
4. Place the configuration file "slhlp.xml" into the directory `/oem/sinumerik/hmi/cfg`.
5. Restart the HMI.

Note

When displaying the list of contents and subject index of a help book, the help files are saved in the binary format (slhlp_<Hilfe-Buch_*.hmi) under the directory `/siemens/sinumerik/sys_cache/hmi/hlp` for faster use. If you change the help book, you must always delete these files.

See also

Supported languages (Page 584)

18.6 Generating online help for user alarms and machine data

You have the option to create a dedicated online help for user-specific PLC alarms. These can be opened context-sensitive from the alarm list when the alarms occur.

The help texts for the user-specific PLC alarms are realized in the HTML file "sinumerik_alarm_oem_plc_pmc.html".

Creating an HTML file

1. Create the HTML file in the following directory: `/oem/sinumerik/hmi/hlp/<lng>/sinumerik_alarm_plc_pmc/` or `/user/sinumerik/hmi/hlp/<lng>/sinumerik_alarm_plc_pmc/`.
<lng> stands for the language code, e.g. deu, eng, etc. If you create a language directory, use the language code from the table in Section Supported languages (Page 584)
2. Assign the "sinumerik_alarm_oem_plc_pmc.html" name to the file. If you create help texts for several languages, then create the corresponding number of HTML files with precisely these names. This file name must not be changed and is applicable for all languages.

3. Place the files in the corresponding language directories. For instance, place the file for the German help text in the following directory: `/oem/sinumerik/hmi/hlp/deu/sinumerik_alarm_plc_pmc/sinumerik_alarm_oem_plc_pmc.html`
4. Delete the file `"slhlp_sinumerik_alarm_plc_pmc_*.hmi"` in the `/siemens/sinumerik/sys_cache/hmi/hlp` directory and restart the operating software.

"sinumerik_alarm_oem_plc_pmc.html" file

Entry	Meaning
<code>AlarmNr</code>	Hyperlink to the alarm number
<code></code>	Help text for the corresponding alarm
<code><td width="85%">.....</td></code>	Text that is displayed after the "Explanation" or "Remedy" field.

Example

The alarm number is used as HTML anchor.

```
<?xml version="1.0" encoding="UTF-8"?><!DOCTYPE html PUBLIC "-//W3C//DTD
HTML 4.0 Transitional//EN" >
- <html>
  <head><title></title></head>
- <body>
- <table>
- <tr>
- <td width="15%">
  <b><a name="510000">510000</a></b>
  </td>
- <td width="85%">
  <b>This is the help for user alarm 510000</b>
  </td>
- </tr>
- <tr>
- <td valign="top" width="15%">
  <b>Explanation</b>
  </td>
  <td width="85%"> This is the explanation of user alarm 510000.</td>
- </tr>
- <tr>
  <td valign="top" width="15%"><b>Remedy:</b></td>
  <td width="85%">Eliminate reason for alarm. </td>
- </tr>
- </table>

  </body>
</html>
```

Distribute the online help for user alarms over several HTML files

You have the option of distributing the online help for user alarms over several HTML files.

1. Create an XML file with the name "sinumerik_alarm_plc_pmc.xml" in the following directory: `/oem/sinumerik/hmi/hlp/<lng>/` or `/user/sinumerik/hmi/hlp/<lng>/`
<lng> stands for the language code, e.g. deu, eng, etc. When you create a language directory, use the language code from the table in Section Supported languages (Page 584)
2. Open the XML file and in the tag <NUM_AREAS> enter the number ranges of the alarms and the corresponding HTML files.
3. Store the HTML files, specified in the XML file, in the following directory:
`/oem/sinumerik/hmi/hlp/<lng>/sinumerik_alarm_plc_pmc` or
`/user/sinumerik/hmi/hlp/<lng>/sinumerik_alarm_plc_pmc` file

Example: Creating an XML file

Entry	Meaning
<BOOK>	Help book
<NUM_AREAS>	Number ranges of the alarms and the reference to the particular HTML file

The following example includes the online helps for the alarms from 500000 to 599999 in the HTML file "sinumerik_alarm_oem_1.html", the alarms from 600000 to 699999 in the HTML file "sinumerik_alarm_oem_2.html", the alarms from 700000 to 799999 in the HTML file "sinumerik_alarm_oem_3.html", the alarms from 800000 to 899999 in the HTML file "sinumerik_alarm_oem_4.html".

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<!DOCTYPE HMI_SL_HELP>
<HMI_SL_HELP>
<BOOK>
  <NUM_AREAS>
    <NUM_AREA from="500000" to="599999" ref="sinumerik_alarm_oem_1.html" />
    <NUM_AREA from="600000" to="699999" ref="sinumerik_alarm_oem_2.html" />
    <NUM_AREA from="700000" to="799999" ref="sinumerik_alarm_oem_3.html" />
    <NUM_AREA from="800000" to="899999" ref="sinumerik_alarm_oem_4.html" />
  </NUM_AREAS>
</BOOK>
</HMI_SL_HELP>
```

Replacing the standard online help

You have the option of generating your own online help for the HMI, NC and PLC alarms as well as machine data. This means that you can replace the standard online help by your own online help.

You can call your own online help directly at the control using the INFO key. Instead of the standard online help, your own online help is displayed.

Directory for HTML files

Area	Directory
PLC alarms	/oem/sinumerik/hmi/hlp/<Ing>/sinumerik_alarm_plc_pmc/ or /user/sinumerik/hmi/hlp/<Ing>/sinumerik_alarm_plc_pmc/
HMI alarms	/oem/sinumerik/hmi/hlp/<Ing>/sinumerik_alarm_hmi/ or /user/sinumerik/hmi/hlp/<Ing>/sinumerik_alarm_hmi/
NC alarms	/oem/sinumerik/hmi/hlp/<Ing>/sinumerik_alarm_nck/ or /user/sinumerik/hmi/hlp/<Ing>/sinumerik_alarm_nck/
NC MD	/oem/sinumerik/hmi/hlp/<Ing>/sinumerik_md_nck/
Channel MD	/oem/sinumerik/hmi/hlp/<Ing>/sinumerik_md_chan/
Axis MD	/oem/sinumerik/hmi/hlp/<Ing>/sinumerik_md_axis/

<Ing> stands for the language code, e.g. deu, eng, etc. Only use the language codes from the table in Section Supported languages (Page 584)

Procedure:

1. Create the corresponding directory, refer to the table "Directory for HTML files".
2. Generate the help file and as file names, use the alarm/machine data number as well as the extension ".html" for HTML help files and ".pdf" for PDF help files. If you create help texts for several languages, then create the corresponding number of help files with precisely these names and then create the file in the corresponding language directory.

Examples:

- Your own German online help for the PLC Alarm 510000:
/oem/sinumerik/hmi/hlp/deu/sinumerik_alarm_plc_pmc/510000.html
/oem/sinumerik/hmi/hlp/deu/sinumerik_alarm_plc_pmc/510000.pdf
- Your own German online help for the NC machine data 14510 \$MN_USER_DATA_INT:
/oem/sinumerik/hmi/hlp/deu/sinumerik_md_nck/14510.html

18.7 Example: This is how you create an online help for NC/PLC variables

Overview

In order to create context-sensitive online help for NC/PLC variables or system variables as shown in the following example, the descriptive texts are managed in language-dependent html files.

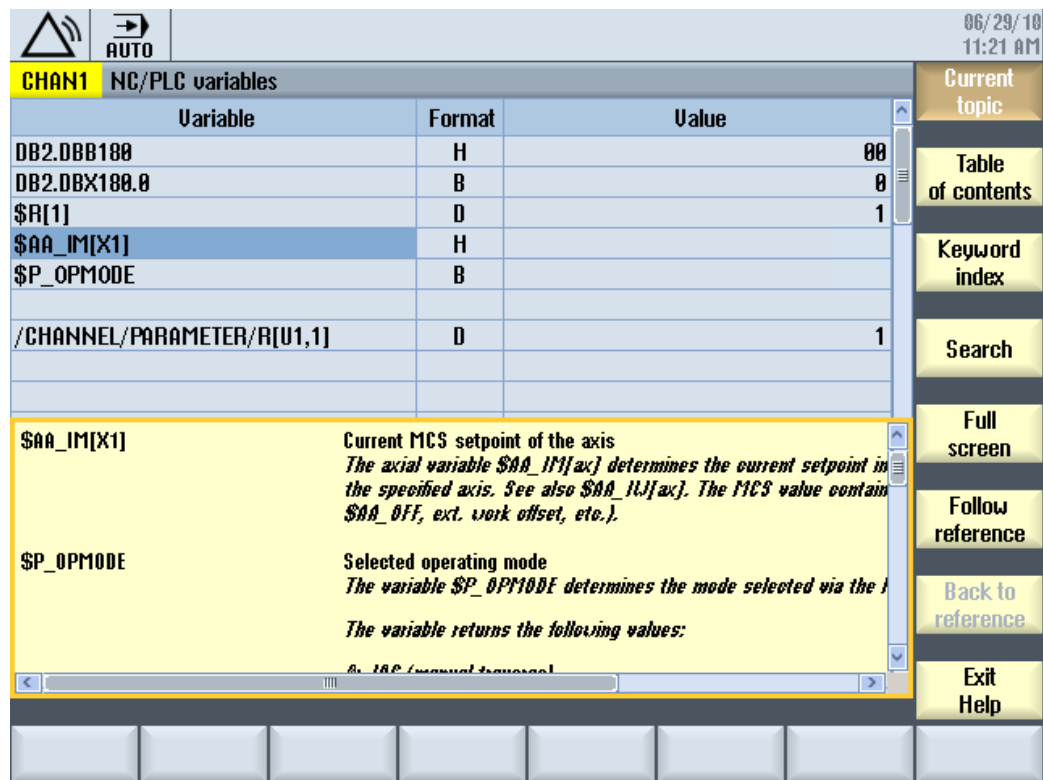


Figure 18-1 Example: Online help for user variables

Structure of the online help

The following files are required for the online help:

File	Meaning
sldgvarviewhelp.ini	Configuration file to manage an html file or several html files
<lng>/<name>1.html	The contents of all html files of the online help are language-dependent and are saved in the relevant language directory <lng>.
<lng>/<name>2.html	
. . .	
<lng>/<name>n.html	

Structure of the configuration file

The file is independent of any language and is structured as follows:

```
sldgvarviewhelp.ini
[HelpBindings]
```

18.7 Example: This is how you create an online help for NC/PLC variables

sldgvarviewhelp.ini

```

/BAG/STATE/OPMODE = var1_help.html#var1
$AA_IM[X1] = var1_help.html
$R[1] = var1_help.html#var2
/Channel/Parameter/R[u1,1] = var2_help.html#var2
DB2.DBX180.0 = var2_help.html#var1
GUD/MyVar[2] = var2_help.html

```

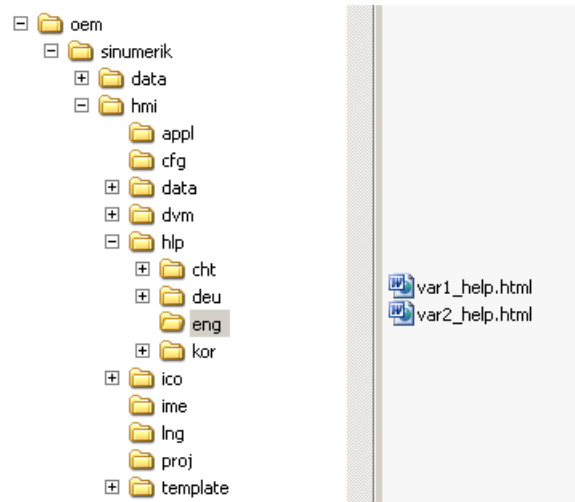
Note

The html files can be generated with any html editor. A definition is provided in the configuration file as to which html files belong to the online help.

The description can comprise one or several html files: For example, one html file for each variable or several identical variables in one file.

Procedure:

1. Copy the configuration file to the following directory:
/oem/sinumerik/hmi/cfg/sldgvarviewhelp.ini
2. Copy the html files to one of the following directories:
/oem/sinumerik/hmi/hlp/<lng>/
/user/sinumerik/hmi/hlp/<lng>/
<lng> stands for the language code.



3. Delete all files in the directory:
/siemens/sinumerik/sys_cache/hmi//hlp

The settings become effective only after restarting the system.

18.8 Example: This is how you create a programming online help

Preconditions

Create the following files:

- Configuration file "prog_help.ini"

```
[milling]
CYCLE1=cycle1_help.html
CYCLE2=cycle2_help.html#TextAnchor1
CYCLE3=cycle3_help.html
CYCLE4=cycle4_help.html

[turning]
CYCLE3=cycle2_help.html
CYCLE4=cycle3_help.html
```

- Configuration file for the help book "shlp.xml" (optional)

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!DOCTYPE CONFIGURATION>
<CONFIGURATION>
  <OnlineHelpFiles>
    <hmi_prog_help>
      <EntriesFile value="hmi_prog_help.xml" type="QString"/>
      <DisableIndex value="true" type="bool"/>
    </hmi_prog_help>
  </OnlineHelpFiles>
</CONFIGURATION>
```

- Configuration file for the help book "hmi_prog_help.xml" (optional)

```
<?xml version="1.0" encoding="UTF-8"?>
<HMI_SL_HELP language="en-US">
```

```

<BOOK ref="index.html" title="OEM_CYCLES" helpdir="hmi_prog_help">
  <ENTRY ref="cycle1_help.html" title="Cycle1"></ENTRY>
  <ENTRY ref="cycle2_help.html" title="Cycle2"></ENTRY>
  <ENTRY ref="cycle3_help.html" title="Cycle3"></ENTRY>
  <ENTRY ref="cycle4_help.html" title="Cycle4"></ENTRY>
  <ENTRY ref="cycle_help.html" title="OEM_Cycles"></ENTRY>
</BOOK>
</HMI_SL_HELP>

```

- Language-dependent file "<prog_help_eng>.ts": this filename is permanently specified.

```

<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<!DOCTYPE TS>
<TS>
  <context>
    <name>oem_cycles</name>
    <message>
      <source>CYCLE1</source>
      <translation>short description for cycle 1</translation>
      <chars>*</chars>
    </message>
    <message>
      <source>CYCLE2</source>
      <translation>short description for cycle 2</translation>
      <chars>*</chars>
    </message>
    <message>
      <source>CYCLE3</source>
      <translation>short description for cycle 3</translation>
      <chars>*</chars>
    </message>
  </context>
</TS>

```

Saving help files in the target system

Procedure:

1. Copy the configuration file "prog.ini" to the following directory:
/oem/sinumerik/hmi/cfg
2. Copy the file "slhlp.xml" to the help book in the following directory:
/oem/sinumerik/hmi/cfg

3. Create a directory for the desired language of the online help under the following path: `/oem/sinumerik/hmi/hlp/<lng>` and copy the `hmi_prog_help.xml` file there. Use the specified language code from the list of language codes for file names. The directory names must be written in lower case.
 4. Copy the language-dependent `prog_help_<lng>.ts` file for the Product Brief to the following path: `/oem/sinumerik/hmi/<lng>/prog_help_<lng>.ts`
 5. Copy the html files with the description of the OEM cycles to the following directory: `/oem/sinumerik/him/hlp/<lng>/hmi_prog_help/cycle<n>_help.html`
- The settings become effective only after restarting the system.

18.9 Help files in PDF format

In addition to help files in HTML format, you can also include PDF-format information in the operating software. Individual PDF helps can be opened with links from the table of contents or index, or directly from HTML files.

Storing PDF helps

Copy the PDF helps to one of the following directories:

```
/oem/sinumerik/him/hlp/<lng>/<hmi_myhelp>  
/user/sinumerik/him/hlp/<lng>/<hmi_myhelp>
```

Including PDF helps

Bind in dialog configurations or configurations of table of contents and indexes with the extension "pdf" in the same manner as for the "html" extension:

```
<ENTRY ref="myFile.pdf" title="Help 1">
```

Create a link from HTML files to the PDF help:

```
<a href="myFile.pdf">My Help File</a>
```

Note

It is not possible to select context-sensitive jump labels in the PDF help, select jump labels or make jumps to other HTML or PDF files.

The search function is possible only within a PDF file. A higher-level search over multiple PDF helps is not supported.

SINUMERIK Integrate - AMB, AMC, AMM

Precondition

To start the "SINUMERIK Integrate - AMB, AMC, AMM" software, configure a softkey using a display machine data. To do this you require, as a minimum, authorization for access level 3 (user).

Configuring a softkey

MD9108 \$MM_SINUMERIK_INTEGRATE	
= 1	The "SINUMERIK integrate" softkey is displayed on the extended user interface bar.

References

Information on the software can be found in the following documentation:

Function Manual SINUMERIK Integrate - AMB, AMC, AMM

Safety Integrated

20.1 Displaying machine data and parameters

20.1.1 View of the axes

You can display the Safety Integrated axes in the "Machine Configuration" window.

Copying and confirming Safety Integrated data

You can copy and confirm Safety Integrated data.

Activating and deactivating the start-up of the Safety Integrated drives

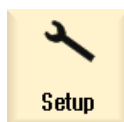
You have the possibility of activating and deactivating Safety Integrated drives.

The Safety commissioning mode in the drives is activated.

References

Further information is available in the Safety Integrated 840D sl Function Manual

Procedure



1. Select the "Start-up" operating area.



2. Press the menu forward key and the "Safety" softkey.
The "Machine Configuration Safety Axes" window opens.



All of the safety axes are displayed.



3. Press the "All axes" softkey if you wish to display all of the axes.

The "All Axes Machine Configuration" window opens and the "All axes" softkey changes into "Safety axes".



4. Press the "Safety axes" softkey in order to return to the view of the safety axes.

See also

Copying and confirming Safety Integrated data (Page 429)

Activating/deactivating start-up mode (Page 430)

Displaying the Safety-Integrated settings (Page 431)

20.1.2 General machine data for Safety Integrated

The general machine data is listed in the "Selection of General MD" window and can be edited, copied and confirmed.

Note

Changes as well as copying and confirming are effective for all axes and the general machine data.

Screen view

The machine data display is split into two halves:

- In the upper part of the screen view, the machine data that are important for commissioning are displayed; however, they do not involve Safety Integrated machine data.
- Safety Integrated machine data are displayed in the lower part of the screen view.

Changing the screen view



1. Press the "MD selection" to change into the full screen view.
The selection of general machine data is displayed over the complete screen.



2. Press the "SI-MD" softkey.
The general safety integrated machine data are displayed over the full screen.



3. Press the "SI + MD selection" softkey.
You return to the common view split in two of the general machine data and the Safety Integrated machine data.

Searching for machine data

	1. Press the "Search" softkey, enter the machine data number being searched for in the window that opens or a character string that is being searched for and press the "OK" softkey.
	
	2. Press the "Continue search" softkey if the text position found does not correspond to the required machine data.
	3. Press the "Go to start" or "Go to end" softkey to set the cursor to the beginning or to the end of the displayed data.
	
	4. Press the "Search" softkey if you wish to change the search term.

20.1.3 Axis-specific machine data in Safety Integrated

The axis machine data is listed in the "Axis MD" window and can be changed.

Copying and confirming Safety Integrated data for certain axes

Select the desired axis using the "Axis +" or "Axis -" softkey for which the Safety Integrated data is to be copied and confirmed.

Screen view

The machine data display is split into two halves:

- In the upper part of the screen view, the machine data that are important for commissioning are displayed; however, they do not involve Safety Integrated machine data.
- Safety Integrated machine data are displayed in the lower part of the screen view.

Changing the screen view

	1. Press the "MD selection" to change into the full screen view. The selection of Axis MD is displayed over the complete screen.
	2. Press the "SI-MD" softkey. The Safety Integrated Axis MD are displayed over the complete screen.
	3. Press the "SI + MD selection" softkey. You return to the common view split in two of the general machine data and the Safety Integrated machine data.

Searching for machine data

	1. Press the "Search" softkey, enter the machine data number being searched for in the window that opens or a character string that is being searched for and press the "OK" softkey.
	
	2. Press the "Continue search" softkey if the text position found does not correspond to the required machine data.
	3. Press the "Go to start" or "Go to end" softkey to set the cursor to the beginning or to the end of the displayed data.
	
	4. Press the "Search" softkey if you wish to change the search term.

20.1.4 Drive machine data in Safety Integrated

The drive machine data is listed in the "Selection of Drive MD" window and can be changed.

Activating or deactivating the startup mode

Use the "Drive +" or "Drive -" softkey to select the drive that you want to switch to startup mode.

Searching for machine data

	1. Press the "Search" softkey, enter the machine data number being searched for in the window that opens or a character string that is being searched for and press the "OK" softkey.
	
	2. Press the "Continue search" softkey if the text position found does not correspond to the required machine data.
	3. Press the "Go to start" or "Go to end" softkey to set the cursor to the beginning or to the end of the displayed data.
	
	4. Press the "Search" softkey if you wish to change the search term.

20.1.5 Copying and confirming Safety Integrated data

All the NC machine data relevant for the SI functions is transferred to the corresponding drive parameters.

The SI machine data or drive parameters to define the encoder mounting arrangement must be separately entered for the NCK and drive. These are not copied

Note

Certain NCK machine data is not copied into the drive parameters.

When confirming the data, the actual checksum is saved in the reference checksum.

Note

The data is saved for all safety axes.

Requirement



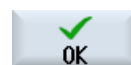
The "Machine configuration" window is open.

Procedure

Copying Safety Integrated data



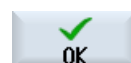
1. Press the "Copy SI data" and "OK" softkeys. Safety-oriented NCK machine data is copied into the drive parameters.



2. Press the "Axis MD" softkey and select an axis using the "Axis +" or "Axis -" softkey if you wish to copy axis-specific machine data and safety data.



3. Press the "Reset (po)" and "OK" softkeys.
The control and the drives restart (warm restart).



Confirming Safety Integrated data



4. Press the "Confirm SI data" and "OK" softkeys.
The correctness of the values is confirmed and the current checksum saved.
The drive data is saved automatically



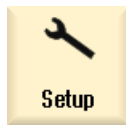
20.1.6 Activating/deactivating start-up mode

For the start-up of the Safety Integrated drives, the drive parameter p0010 is switched to start-up mode (p0010=95).

Value "0" is entered in drive parameter p0010 to exit the start-up mode. Beforehand, the CRC in p9798 and p9898 are confirmed. The start-up state can be exited with or without saving.

At the same time, you can set the drive PROFIsafe address.

Procedure



1. Select the "Start-up" operating area.



2. Press the menu forward key and the "Safety" softkey.



3. Press the "Drive MD" softkey and select the desired drive using the "Drive +" or "Drive -" softkey.



- OR -

...



Press the "View axes" softkey.



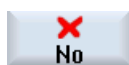
4. Press the "Activate dr. start-up" softkey.
A prompt is displayed.



5. Press the "OK" softkey to continue with the start-up.
A prompt is displayed whether you want to set the PROFISafe address for the selected drive.

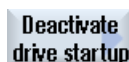


6. Press the "Yes" softkey.
Already existing settings of parameter p9810 are overwritten and the drive parameter p0010 is switched to the start-up mode.
- OR -



Press the "No" softkey.

The existing settings of parameter p9810 are retained and the drive parameter p0010 is switched to the start-up mode.



7. Press the "Deactivate dr. start-up" softkey.

20.1.7 Displaying the Safety-Integrated settings

The following configured values can be displayed in the "Safety Integrated Settings" window:

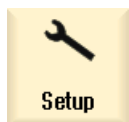
- Safe operating stop and safely limited speed (SBH/SG)
- Safe software end position for software limit switches (SE)
- Safe cams (SN)



You can use the softkeys to scroll between the Safety Integrated axes.



Procedure



1. Select the "Startup" operating area.



2. Press the menu advance key and the "Safety" softkey.



3. Press the "View settings" softkey.
The "Safety Integrated Settings" window opens.



The configured limits for safe operating stop (SBH) and safely limited speed (SG) are displayed.



4. Press the "Show safe limit pos." softkey if you wish to display the configured directions and limits of the software limit switches (SE).



5. Press the "Show Safe Cams" softkey if you wish to display the positions and assignments of the safe cams (SN).

See also

View of the axes (Page 425)

20.2 Safety Integrated diagnostics

20.2.1 Status display for NCK Safety Integrated

The status display shows signals or values with NCK and drive information for a selected axis.

Available signals

- Safe actual position
- Position deviation NCK/drive
- "Safe operating stop" monitoring active
- "Safe velocity" monitoring active
- Active SV step
- Active SV correction factor
- Safe actual velocity limit
- Set velocity limit
- Current velocity difference
- Maximum velocity difference
- Active safe software limit switch
- Active gear ratio (step)
- Active stop
- Currently requested external stop
- Stop F code value
- Pulses enabled
- Traversing inhibit, stop in other axis

Procedure



1. Select the "Diagnostics" operating area.



2. Press the menu forward key and the "Safety" softkey.



3. Press the "SI status" softkeys.
The "Safety Integrated Status (NCK)" window appears.



...



4. Press the "Axis +" or "Axis -" softkey.
The values of the next or the previous axis are displayed.
- OR -

Press the "Axis selection" softkey and select the desired axis directly from the displayed list of available axes.

20.2.2 Display SGE/SGA signals

Safety-related inputs and outputs

SGE

Safe input signals, NCK Bit15...0

Safe input signals, drive Bit15...0

Safe input signals, NCK, Bit 16...31

Safe input signals, drive, Bit 16...31

SGA

Safe output signals, NCK Bit15...0

Safe output signals, drive Bit15...0

Safe output signals, NCK, Bit 16...31

Safe output signals, drive, Bit 16...31

Procedure



1. Select the "Diagnostics" operating area.
2. Press the menu forward key and the "Safety" softkey.
3. Press the "SGE/SGA" softkey to display the safety-related input and output signals.
The "Safety Integrated SGE/SGA" window appears.



4. Press the "Axis +" or "Axis -" softkey.
The values of the next or the previous axis are displayed.



- OR -



Press the "Axis selection" softkey and select the desired axis directly from the displayed list of available axes.

20.2.3 Display SPL signals

Safe programmable logic

Variables

\$A_INSE (P)	\$A_INSE (P) - corresponds to simultaneous selection of: \$A_INSE (upper line, origin of the NCK) and \$A_INSEP (lower line, origin of the PLC)
\$A_OUTSE (P)	comp. \$A_INSE (P)
\$A_INSI (P)	comp. \$A_INSE (P)
\$A_OUTSI (P)	comp. \$A_INSE (P)
\$A_MARKERSI (P)	comp. \$A_INSE (P)
\$A_PLCSIIN	comp. \$A_INSE (P)
\$A_PLCSIOU	comp. \$A_INSE (P)

Bit Selection of an 8-bit area of the selected signal.

Available signals/values

DCC fill level
DCC status
DCC control word
SPL booting state
SPL started up
SPL interfaces have been parameterized
SPL program file SAFE.SPL loaded
NCK and PLC state
Interrupt for PLC start should be assigned
Interrupt has been assigned for PLC start
Interrupt processing for SPL start called
Interrupt processing for SPL start terminated
SPL start implemented using PROG_EVENT mechanism
SPL started via AUTO start
SPL processing completed, end of program reached.

NCK data cross-checking (DCC) has been started
PLC data cross-checking has been started
Cyclic SPL checksum test active
All SPL protective mechanisms active

Procedure



1. Select the "Diagnostics" operating area.



2. Press the menu forward key and the "Safety" softkey.



3. Press the "SPL" softkey to display the safe programmable logic signals.
The "Safety Integrated SPL" window is opened

20.2.4 Displaying Safety Integrated checksums

You obtain the following information areas in the "SI Configuration" window:

- Overview of the settings of the Safety Integrated option
- Tables indicate the checksum for the actual Safety Integrated configuration.

Safety Integrated options

The overview of the Safety Integrated options includes the number of the Safety Integrated axes configured as well as an overview of the active Safety Integrated options.

Safety Integrated checksum area

The following checksums are displayed for the overview of the Safety Integrated checksum area:

- Checksum for the "SAFE.SPF" program
- General checksums
- Checksum for NCK and drive for each axis and the corresponding drive, for which the Safety Integrated has been enabled.

You receive information about the time of the last change of the checksum in the NCK.

Procedure



1. Select the "Diagnostics" operating area.



2. Press the menu forward key and the "Safety" softkey.



3. Press the "SI checksum" softkey.
The "SI Configuration" window opens.



4. Press the "Axis +" or "Axis -" softkey.
The values of the next or the previous axis are displayed.

...



- OR -



Press the "Axis selection" softkey and select the desired axis directly from the displayed list of available axes.

20.2.5 Safety Integrated Basic Functions (Drive)

Supported functions

The following Safety Integrated functions (Safety Integrated Basic Functions) are available:
These functions are available in the standard drive version.

- **Safe Torque Off (STO)**
STO is a safety function that prevents the drive from restarting unexpectedly in accordance with EN 60204-1.
- **Safe Stop 1 (SS1, time controlled)**
The SS1 function is based on the "Safe Torque Off" function. This means that a Category 1 stop in accordance with EN 60204-1 can be implemented.
- **Safe Brake Control (SBC)**
The SBC function permits the safe control of a holding brake.

Note

Safety Integrated Basic Functions

When a drive object that has Safety Integrated functions released is switched to "Parking" state, the Safety Integrated software responds by activating STO without generating a separate message.

References

The functions are described in: Function Manual, Safety Integrated

Procedure

	1. Select the "Diagnostics" operating area.
	2. Press the menu forward key and the "Safety" softkey.
	
	3. Press the "SI status" softkey. The "SI status" window is opened.
	4. Press the "Display drive status" softkey to display the status of safe drives.
	5. Press the "Drive +" or "Drive -" softkey. The parameters are displayed for the next or previous drive object.
...	
	- OR -
	6. Press the "Drive selection" softkey and in the drop-down list that appears, directly select the desired drive object.

20.2.6 Safety Integrated fail-safe I/O

Fail-safe I/O modules ("fail-safe modules")

The "SI I/O" display provides a summary of the configuration and status of the PROFIsafe I/O. The following signals are displayed:

- Fail-safe master address
- Parameterized PROFIsafe cycle time
- Maximum occurring PROFIsafe cycle time
- Modular PROFIsafe I/O interface
- Number of active I/O modules
- Number of passive I/O modules

Procedure



1. Select the "Diagnostics" operating area.



2. Press the menu forward key and the "Safety" softkey.



3. Press the "SI I/O" softkey.
The "SI I/O" window is opened and the status of the signals is displayed. Press the "Fail-safe modules" softkey to display the status of the fail-safe modules and their PROFIsafe address.



4. Press the "Details" softkey to display further information on this fail-safe module.

20.2.7 Displaying cam signals

The signals of the cam SGA are displayed in the "Cam SGA" window.

Cam synchronization is executed between NCK and PLC at the cam SGA.

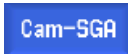
Procedure



1. Select the "Diagnostics" operating area.



2. Press the "Menu forward" key and then the "Safety" softkey.



3. Press the "Output cam SGA" softkey.
The "Cam SGA" window opens.



4. Press the "Axis +" or "Axis -" softkey.
The values of the next or the previous axis are displayed.



- OR -



Press the "Axis selection" softkey and select the desired axis directly from the displayed list of available axes.

20.2.8 Displaying SI communication data

Signals and values for the following communication data are displayed in the "SI Communication (General)" window:

- Parameterized CPU-CPU communication clock cycle status
- Actual CPU-CPU communication clock cycle status
- Maximum CPU-CPU communication clock cycle
- Number of active send data connections (F_SENDDP)
- Number of active receive data connections (F_RECVDP)

Procedure



1. Select the "Diagnostics" operating area.



2. Press the "Menu forward" key and then the "Safety" softkey.



3. Press the "SI communication" softkey.



The "SI Communication (General)" window opens.

20.2.9 SI communication send connections

The configuration as well as the status of F_SENDDP are displayed in the SI communication (send)" window.

Displayed signals

The following signals of the selected connection are displayed:

- CPU-CPU communication ID
- Logical base address
- Connection number
- Maximum parameterized communication time
- Actual communication time
- Maximum communication time
- Error reaction
Error reactions that can be set:
 - [0] "Alarm 27350 + STOP D/E"
 - [1] "Alarm 27350"
 - [2] "Alarm 27351 (self clearing)"
 - [3] "No reaction"
- Error
- Diagnostics error code
- Substitute values active
- Actual communication data
- Driver state
Status display:
 - [0] "Not initialized"
 - [1] "Establish communication after booting"
 - [2] "Establish communication after error"
 - [3] "Wait for checkback signals SN=1"
 - [4] "Receiver waits for user acknowledgement"
 - [5] "Normal operation"

Procedure

General Send connect. Connect. + Connect. -	1. The "SI communication (general)" window with the communication data is opened. 2. Press the "Send connection" softkey. The "SI communication (send)" window is opened. 3. Press the "Connection +" or "Connection -" softkey to select the connection whose data you wish to display.



4. Press the "Display SPL connection" softkey in order to display additional details, e.g. connection data that has been set.

The softkey is only available if send connections have been configured.



5. Press the "Display connection" softkey to return to the window with the send connections of the selected connection.

20.2.10 SI communication receive connections

The configuration as well as the status of F_ RECVDP are displayed in the SI communication (receive)" window.

Displayed signals

The following signals of the selected connection are displayed:

- CPU-CPU communication ID
- Logical base address
- Connection number
- Maximum parameterized communication time
- Actual communication time
- Maximum communication time
- Error reaction
 - Error reactions that can be set:
 - [0] "Alarm 27350 + STOP D/E"
 - [1] "Alarm 27350"
 - [2] "Alarm 27351 (self clearing)"
 - [3] "No reaction"
- Error
- Diagnostics error code
- Substitute values active
- Substitute values
- Actual communication data
- Driver state
 - Status display:
 - [0] "Not initialized"
 - [1] "Establish communication after booting"
 - [2] "Establish communication after error"
 - [3] "Wait for checkback signals SN=1"
 - [4] "Receiver waits for user acknowledgement"
 - [5] "Normal operation"

- Request user acknowledgement
- Sender in the deactivated safety mode

Procedure

	1.	The "SI communication (general)" window with the communication data is opened.
	2.	Press the "Receive connection" softkey. The "SI communication (send)" window is opened.
	3.	Press the "Connection +" or "Connection -" softkey to select the connection whose data you wish to display.
		
	4.	Press the "Display SPL connection " softkey to display additional details, e.g. connection data that has been set. The softkey is only available if send connections have been configured.
	5.	Press the "Display connection" softkey to return to the window with the receive connections of the selected connection.

Configuring cycles

21.1 Activating turning/milling/drilling/grinding technologies

The cycles are automatically loaded when the controller boots (Power On).

The configuration is realized using cycle machine data and cycle setting data. In so doing, a differentiation is made between the following data:

- General cycle machine data
- Channel-specific cycle machine data
- Axis-specific cycle machine data
- General cycle setting data
- Channel-specific cycle setting data
- Axis-specific cycle setting data

Requirement for memory configuration

To ensure that the cycles run correctly, the following standard setting must be checked:

MD28082 \$MC_MM_SYSTEM_FRAME_MASK	System frames (SRAM)
Bit 5	System frame for cycles
= 1	Default setting

MD52201 \$MCS_TECHNOLOGY_EXTENSION	Technology extension for combined machines with several technologies
= 1	Additional settings for turning
= 2	Additional settings for turning, e.g. lathe with milling technology MD52200 \$MCS_TECHNOLOGY = 1 MD52201 \$MCS_TECHNOLOGY_EXTENSION = 2
= 3	Additional settings for the cylindrical grinding technology
= 4	Additional settings for the surface grinding technology

Adapting the user interface

The softkey display for the selection of the cycle support in the program editor is set using the following channel-specific configuration machine data:

MD52200 \$MCS_TECHNOLOGY	Technology
= 1	Setting for turning
= 2	Setting for milling

= 3	Setting for the cylindrical grinding technology
= 4	Setting for the surface grinding technology

References: Loops

For further information about the grinding technology, please refer to:

- Chapter "grinding (Page 491)"
- Operating Manual, Grinding
- Programming Manual, Production Planning; Chapter "Programming cycles externally":
 - CYCLE495 - form-truing
 - CYCLE4071 - CYCLE4074: Longitudinal grinding cycles
 - CYCLE4075 - CYCLE4079: Surface grinding cycles
- Function Manual, Extended Functions; Chapter "Compensations (K3)" > "Interpolatory compensation" > "Cylinder error compensation"

Configuring the screen display

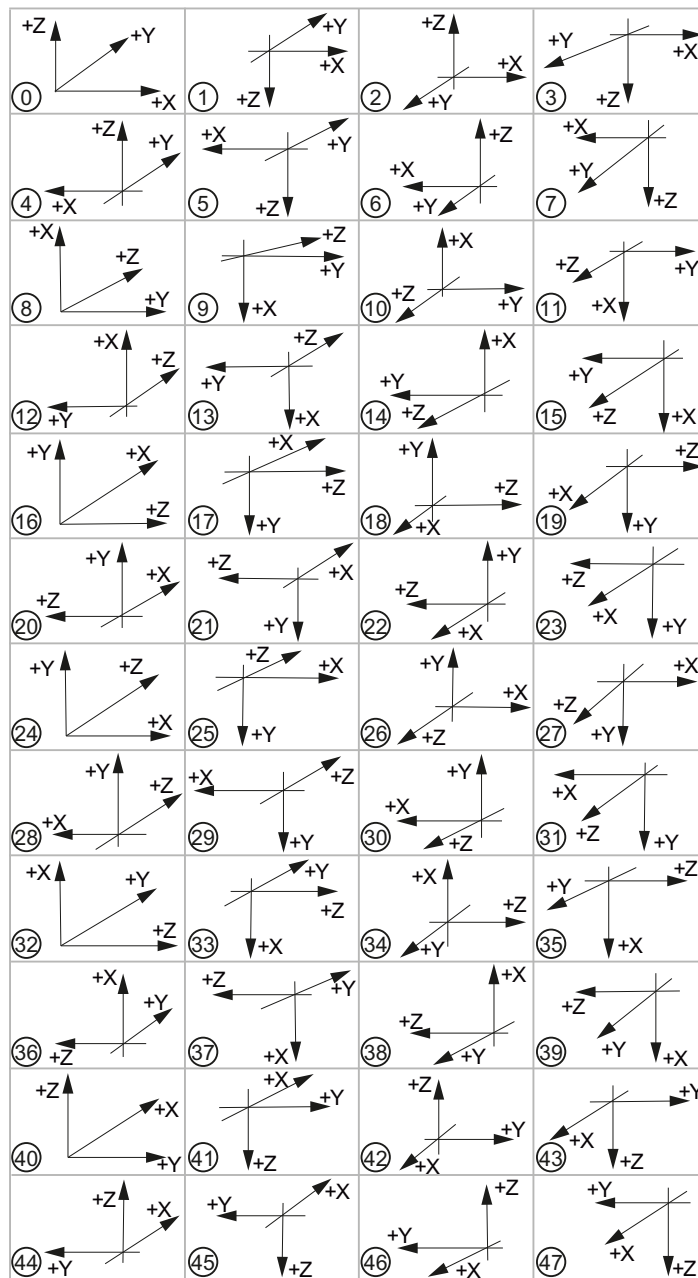
The screen display can be configured using the following channel-specific machine data:

MD52210 \$MCS_FUNCTION_MASK_DISP		Function screen display
Bit 0	Dimension system for programs always in the basic system	
Bit 1	Front view when turning in the school coordinate system	
Bit 2	Hide "T,S,M" softkey in the JOG area	
Bit 3	Generate end of program automatically in MDI (with "Delete blocks" softkey)	
Bit 4	Display follow-on tool in the T,F,S window	
Bit 5	Hide "MCS " softkey	
Bit 6	Hide tool radius/diameter in the T,F,S window	
Bit 7	Hide tool lengths in the T,F,S window	
Bit 8	Hide tool icon in the T,F,S window	
Bit 9	Display "Easy" input selection, see Section ""Easy" input in cycle support"	
Bit 10	Do not offer channel in job lists	
Bit 11	For WCS, activate the display of the logical spindles	
Bit 12	Hide mold making view for G code	

Setting the coordinate system

MD52000 \$MCS_DISP_COORDINATE_SYSTEM		Position of the coordinate system
= 0	Default setting	
With this machine data you adapt the position of the coordinate system of the user interface to the coordinate system of the machine. In the user interface, all help screens, sequence graphics, simulations, and input fields with circle direction indication change automatically according to the selected position.		

The following settings can be selected for the coordinate system:



Meaning of the axes

MD52206 \$MCS_AXIS_USAGE[n]		Meaning of the axes in the channel [n] channel axis number
= 0	No special meaning	
= 1	Tool spindle (driven tool)	
= 2	Auxiliary spindle (driven tool)	
= 3	Main spindle (turning)	
= 4	Separate C axis of the main spindle (turning)	

= 5	Counterspindle (turning)
= 6	Separate C axis of the counterspindle (turning)
= 7	Linear axis of the counterspindle (turning)
= 8	Tailstock (turning)
= 9	Back rest (turning)
= 10	B axis (turning)
= 12	B axis in counterspindle (turning)
= 13	Transverse travel X of the counterspindle (turning)

Defining the direction of rotation

Enter the direction of rotation for the rotary axes that are not configured in a toolholder or a 5-axis transformation via the following channel-specific machine data.

MD52207 \$MCS_USAGE_ATTRIB[n]		Attribute of the axes in the channel [n] channel axis number
Bit 0	Rotation around the 1st geometry axis (for rotary axes)	
Bit 1	Rotation around the 2nd geometry axis (for rotary axes)	
Bit 2	Rotation around the 3rd geometry axis (for rotary axes)	
Bit 3	Direction of rotation is counter-clockwise (for rotary axis / C axis).	
Bit 4	Displayed direction of rotation of the spindle/C axis for the M function M3 is counter-clockwise	
Bit 5	Inverts M3 / M4 (for spindles) This bit must be set analog to PLC bit DBnn.DBX17.6! (nn = 31 + Machine axis index)	
Bit 6	Display rotary axis as offset target for measurement	
Bit 7	List rotary axis in the position pattern	
Bit 8	List rotary axis to accept blank (on milling machines)	
Bit 9	Spindle is not SPOS-capable	
Bit 10	Rotary axis rotates around the 1st geometry axis (only for position pattern)	
Bit 11	Rotary axis rotates around the 2nd geometry axis (only for position pattern)	
Bit 12	Rotary axis rotates around the 3rd geometry axis (only for position pattern)	

"Easy" input in cycle support

You can influence the number of visible input parameters via MD52210
\$MCS_FUNCTION_MASK_DISP with bit 9:

- All input parameters are displayed with the default setting bit 9 = 0 for "Complete".
- With the setting bit 9 = 1, you can select "Complete" or "Easy" input parameters in the cycle support.

If you select the "Easy" input screen for input of the parameters, the values from the following setting data are used for the invisible parameters (can be adapted machine-specifically):

SD55300 \$SCS_EASY_SAFETY_CLEARANCE		Easy input: Safety clearance
= 1	mm	

21.1 Activating turning/milling/drilling/grinding technologies

SD55301 \$SCS_EASY_DWELL_TIME	Easy input: Dwell time
= 0.6	s

SD55305 \$SCS_EASY_DRILL_DEEP_FD1	Easy input: Percentage of 1st deep-hole drilling feedrate
= 90	%

SD55306 \$SCS_EASY_DRILL_DEEP_DF	Easy input: Percentage of deep-hole drilling infeed
= 90	%

SD55307 \$SCS_EASY_DRILL_DEEP_V1	Easy input: Min. deep-hole drilling depth infeed
= 1.2	mm

SD55308 \$SCS_EASY_DRILL_DEEP_V2	Easy input: Deep-hole drilling retraction distance
= 1.4	mm

SD55309 \$SCS_EASY_THREAD_RETURN_DIST	Easy input: Thread turning return distance
= 2	mm

Note

The specified default values are valid for the metric system and are converted by the NC automatically on the machine with the inch setting.

Additional settings

MD52005 \$MCS_DISP_PLANE_MILL	Plane selection G17, G18, G19
= 0	Milling: Plane selection in the cycle support when programming under "programGUIDE G code"
= 17	G17 plane (default value)
= 18	G18 plane
= 19	G19 plane

MD52006 \$MCS_DISP_PLANE_TURN	Plane selection G17, G18, G19
= 18	G18 plane (permanently set)

MD52212 \$MCS_FUNCTION_MASK_TECH		Cross-technology function screen
Bit 0	Enable swivel	
= 0	Swivel plane, swivel tool not enabled	
= 1	Swivel plane, swivel tool enabled	
Bit 1	No optimized travel along software limit switches	
= 0	No optimized travel along software limit switches	
= 1	Optimized travel along software limit switches	
Bit 5	Call block search function for SERUPRO.	
= 0	CYCLE207 is not called in the block search cycle PROG_EVENT.SPF.	
= 1	The cycle for SERUPRO (CYCLE207) is called in the block search cycle PROG_EVENT.SPF	

MD52240 \$MCS_NAME_TOOL_CHANGE_PROG		Tool change program for G code steps
= Program name	The associated program is called for tool change.	

See also

Technology cycles for swiveling (Page 493)

21.2 Technology cycles for drilling

Drilling technology

You can set drilling technology using the following channel-specific configuration machine data and channel-specific cycle setting data.

MD52216 \$MCS_FUNCTION_MASK_DRILL		Drilling function screen form
Bit 0	Tapping cycle CYCLE84, technology input fields	
= 0	Hide input fields	
= 1	Display input fields	
Bit 1	Tapping cycle CYCLE840, technology input fields	
= 0	Hide input fields	
= 1	Display input fields	

SD55216 \$SCS_FUNCTION_MASK_DRILL_SET		Drilling function screen form
Bit 1	Boring CYCLE86: Take into account the rotation of the tool plane when positioning the spindle Note: Spindle direction of rotation M3/ M4 and direction of rotation of the rotary axes must be set according to DIN. For SPOS=0, the tool cutting edge points in the + direction of the 1st axis of the plane (for G17, to X+).	
= 0	Do not take into account the rotation of the tool plane when positioning the spindle (SPOS).	

= 1	Take into account the rotation of the tool plane when positioning the spindle (SPOS).
Bit 2	Boring CYCLE86: When positioning the spindle, take into account swiveled table kinematics (tool carrier)
= 0	When positioning the spindle, do not take into account the components of the swiveled table kinematics (swiveling with CYCLE800, tool carrier). Note: Only the rotary axis of the machine kinematics that rotates the workpiece is taken into account. In the initial state of the kinematics, this rotary axis must be orientated parallel to the tool axis. e.g. rotary axis C rotation around Z (for G17) rotary axis vector of the swivel data set V2xyz = 0, 0, -1.
= 1	When positioning the spindle, take into account the components of the swiveled table kinematics (swiveling with CYCLE800, tool carrier).
Bit 3	Tapping CYCLE84: Monitoring of the machine data MD31050 \$MA_DRIVE_AX_RATIO_DENOM and MD31060 \$MA_DRIVE_AX_RATIO_NUMERA of the spindle
Bit 4	Tapping CYCLE84, set monitoring of the machine data MD31050 \$MA_DRIVE_AX_RATIO_DENOM and MD31060 \$MA_DRIVE_AX_RATIO_NUMERA of the spindle
= 0	No monitoring
= 1	Monitoring
Bit 6	Boring CYCLE86: Adapt the spindle position to the tool orientation Tool orientation can be realized using TOOLCARR or TRAORI before the cycle is called.
= 1	Bits 1 and 2 are no longer effective.
Bit 7	Boring CYCLE86: Adapt the direction of retraction travel in the plane to the active mirroring
= 0	Retraction in the plane is realized as programmed.
= 1	When mirroring is active, the direction when retracting in the plane is adapted in the cycle to the active mirroring.

Tapping (CYCLE84 and CYCLE840)

If the technology screens are hidden using the channel-specific configuration machine data MD52216 \$MCS_FUNCTION_MASK_DRILL, then the settings in the following channel-specific cycle setting data are effective:

SD55481 \$SCS_DRILL_TAPPING_SET_GG12[0]	Setting tapping G group 12: Block change behavior for exact stop
= 0	Exact stop response as before the cycle call (default value).
= 1	G601
= 2	G602
= 3	G603

SD55482 \$SCS_DRILL_TAPPING_SET_GG21[0]	Setting tapping G group 21: Acceleration behavior
= 0	Acceleration behavior as before the cycle call (default value).
= 1	SOFT
= 2	BRISK
= 3	DRIVE

SD55483 \$SCS_DRILL_TAPPING_SET_GG24[0]		Setting tapping G group 24: Precontrol
= 0	Precontrol, the same as before the cycle call (default value).	
= 1	FFWON	
= 2	FFWOF	

Tapping (CYCLE84)

SD55484 \$SCS_DRILL_TAPPING_SET_MC[0]		Spindle operation for MCALL
= 0	For MCALL, reactivate spindle operation (default value).	
= 1	For MCALL, remain in position controlled spindle operation.	

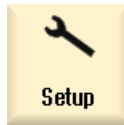
21.3 Manufacturer cycles**21.3.1 Manufacturer cycles**

The following cycles are also available in the cycle package for individual adaptation:

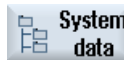
CUST_TECHCYC.SPF	Manufacturer cycle for the adaptation of technology cycles.
CUST_800.SPF	Manufacturer cycle for the adaptation of the swivel plane and the swivel tool functions (CYCLE800), see Chapter "Manufacturer cycle CUST_800.SPF (Page 521)".
CUST_832.SPF	Manufacturer cycle for the adaptation of the high-speed settings function (CYCLE832)", see Chapter Manufacturer cycle CUST_832.SPF (Page 545)".
CUST_MEACYC.SPF	Manufacturer cycle for the adaptation of the measuring functions, see Chapter "Manufacturer and user cycle CUST_MEACYC.SPF (Page 549)".
CUST_MEAPROT.SPF	Manufacturer cycle and user cycle for the adaptation of the log form and content, see Chapter "Manufacturer and user cycle CUST_MEAP-ROT.SPF (Page 578)".
PROG_EVENT.SPF	Standard cycle to support the following functions: • Block search when milling or turning is activated • Block search and swivel plane • Block search and align / advance tool Observe MD52212 \$MCS_FUNCTION_MASK_Tech bit 3 and bit 5.
CUST_T	Cycle is used to track the T preparation after SERUPRO.
CUST_M6	Cycle is used to track the tool change after SERUPRO.
CUST_MULTICHAN	Manufacturer cycle for lathes with several channels
CUST_CLAMP	Manufacturer's cycle for clamping elements

Copying manufacturer cycles

Copy the cycle from the NC data/Cycles/Standard cycles directory to the NC data/Cycles/User cycles directory before making any changes. Changes for the cycle take effect only in this directory.



1. Select the "Start-up" operating area.



2. Press the "System data" softkey.

3. Open the directory NC data/Cycles/Standard cycles and select the manufacturer cycles.



4. Press the "Copy" softkey and open the NC data/Cycles/Manufacturer cycles directory.



5. Press the "Paste" softkey.
The cycles are available for you to individually adapt them.

Note

When upgrading the software, please refer to the upgrade instructions.

21.3.2 Prerequisites for the simulation mode

Configuration

If you use feedback signals regarding the machine state, e.g. in CYCPE1MA, for tool change cycles, measuring cycles or CUST_800, then in the NC program you must link in the following system variables in order to avoid PLC access. Otherwise, PLC access prevents data alignment, results in alarms or the simulation being canceled:

System variable	Meaning
\$P_SIM	NCK simulation active

Example

In the following example, the simulation mode in PROG_EVENT or CYCPE1MA.SPF is stopped by the PLC access:

```
N100 REPEAT
N110 UNTIL PLC_VAR <> 0
```

If you use the system variable \$P_SIM, then PLC access in the simulation mode is prevented:

```
N099 IF NOT $P_SIM
N100 REPEAT
```

```
N110 UNTIL PLC_VAR <> 0
N111 ENDIF
```

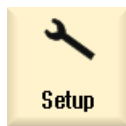
21.3.3 Extending the PROG_EVENT standard cycle

If you wish to extend the functionality of the PROG_EVENT, then you must not copy this into the manufacturer cycles directory, but you must create two new manufacturer cycles. Use the names CYCPE1MA.SPF or CYCPE_MA.SPF names for the manufacturer cycles.

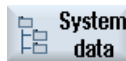
For the manufacturer cycles CYCPE1MA.SPF or CYCPE_MA.SPF, the corresponding jump markers are prepared at the beginning and at the end of PROG_EVENT.SPF.

If the CYCPE1MA.SPF and CYCPE_MA.SPF manufacturer cycles are created in the NC, the program correspondingly branches from PROG_EVENT to the manufacturer cycles.

Procedure



1. Select the "Start-up" operating area.



2. Press the "System data" softkey.



3. Select the directory /NC data/Cycles/Manufacturer cycles

4. Press the "New" softkey.
The "New G Code Program" window opens.

5. Assign the name CYCPE1MA or CYCPE_MA.
Only use these names.



6. Press the "OK" softkey.
The cycle has been created and you can now adapt it.

21.3.4 Manufacturer cycle for tool change CUST_T and CUST_M6

Function

The two cycles are used to track the T preparation (CUST_T) and the tool change (CUST_M6) according to SERUPRO. In so doing, depending on the specific toolholder, the following is output:

- Last programmed change
- Last programmed preparation

Precondition

The cycles can only be used when tool management is active.

Sequence

The two cycles are called in the "SERUPRO-END-ASUB", which also provides the two transfer parameters.

Parameter	Meaning
_THnr	Toolholder or master spindle number that was programmed for the change or the preparation.
_MTHnr	Number of the active toolholder or master spindle at the time of programming.

The cycles themselves do not contain and "machine logic". Pure data operations are performed.

The active toolholder or the master spindle is saved.

Using the transfer parameter, the situation at the time of the T or M6 programming is determined. This situation is established, T and/or M06 output or a branch is made into the manufacturer cycles. After the output of T and M06 or after returning from the manufacturer cycles, the toolholder, saved at the beginning, is reactivated.

A change cycle can then only be called without any restriction, if the corresponding queries such as program test, block search,...exist.

T replacement cycle

If a T replacement cycle is being used, then the following note must be observed:

Note

The CUST_T cycle uses the language command TCA (ToolChangeAbsolut) for the T call and therefore also has the possibility of inserting tools that have been disabled. However, this language command cannot be substituted.

The following must be observed if working at a machine with T replacement:

- The T preparation cycle must be explicitly called at the specified position.
 - If the call is realized using the CUST_T cycle, then the parameter evaluation (\$C_TS_PROG, ...) is skipped.
Alternatively, the original TCA language command can be reprogrammed.
-

References

For more information, please refer to the Function Manual, Basic Functions, Chapter: SERUPRO-End-ASUB

21.3.5 CUST_TECHCYC.SPF manufacturer cycle

Function

For technological cycles, the CUST_TECHCYC cycle is called from ShopMill and ShopTurn cycles, if the machine builder must make certain adaptations to a specific machine (e.g. activate specific M commands). For instance, this may be necessary in order to retract or extend part catchers when cutting off parts on lathes or to position a spindle for clamping (only in JobShop).

The CUST_TECHCYC.SPF cycle can be used to program and execute the necessary sequences.

Adapt manufacturer cycle CUST_TECHCYC.SPF

You can execute the following actions with the aid of the _MODE parameter:

Marker	Action	
_M1	Main spindle:	Changeover to C axis mode
_M2		Changeover to spindle mode
_M3		Clamp C axis
_M4		Release C axis clamping
_M5		Rinse chuck
_M6		Close chuck
_M7		Open chuck when spindle is stationary
_M8		Open chuck when spindle is rotating
_M9		Flush chuck
_M11	Tool spindle:	Changeover to C axis mode
_M12		Changeover to spindle mode
_M13		Clamp C axis
_M14		Release C axis clamping
_M21	Counterspindle:	Changeover to C axis mode
_M22		Changeover to spindle mode
_M23		Clamp C axis
_M24		Release C axis clamping
_M25		Rinse chuck
_M26		Close chuck
_M27		Open chuck when spindle is stationary
_M28		Open chuck when spindle is rotating
_M29		Rinse chuck off
_M30	4. Position axis after block search (e.g. counterspindle, tailstock, back rest)	
M41	Engage driven tool	

Marker	Action	
_M42	Disengage driven tool	
_M61	Changeover to turning	
_M62	Changeover to peripheral surface machining C	
_M63	Changeover to face machining C	
_M64	Changeover to peripheral surface machining Y	
_M65	Changeover to face machining Y	
_M66	Changeover to face machining B (swivel)	
_M67	Deselection of the face machining B (swivel)	
_M68	After block search to ST block before swivel	
_M100	Drawer:	Position before cut-off
_M101		Open during cut-off
_M102		Close after cut-off
_M103	Cut-off completed	
_M110	Tool change	
_M111	After end of tool change cycle (with tool change)	
_M112	Before tool change on retraction plane	
_M113	Before cutting edge reselection	
_M114	After end of tool change cycle (without tool change)	
_M115	After tool preselection (ShopTurn)	
_M116	At tool change point (ShopTurn)	
_M117	Before tool change T command	
_M120	Definition: Coupling of counterspindle / main spindle	
_M121	After COUPOF switch off position control	
_M122	Definition: Coupling, main/counterspindle	
_M123	After COUPOF switch off position control	
_M125	Changeover to main spindle	
_M126	Changeover to counterspindle	
_M131	Start of program (ShopTurn program)	
_M132	End of program header (ShopTurn program)	
_M135	End of program loop (ShopTurn program)	
_M136	End of program (ShopTurn program)	
_M140	Block search	Before tool change
_M141		After tool change
_M142		After tool change (new tool from ShopTurn)

Marker	Action	
_M150	2-channel stock removal:	Leading channel before Balance Cutting
_M151		Leading channel after Balance Cutting
_M152		Following channel before Balance Cutting
_M153		Following channel after Balance Cutting
_M154		Start machining in the leading channel
_M155		End machining in the leading channel
_M156		Start machining in the following channel
_M157		Start machining in the following channel
_M160	Start multichannel programming	
_M170	Start block with spindle	
_M171	End block with spindle	
_M231	Start of program (ShopMill program)	
_M232	End of program header (ShopMill program)	
_M235	End of program loop (ShopMill program)	
_M236	End of program (ShopMill program)	
_M300	Boring CYCLE86 Adapt the spindle to the tool position (SD55216 \$SCS_FUNCTION_MASK_DRILL_SET bit 6 = 1)	

21.3.6 CUST_MULTICHAN user cycle

Function

The CUST_MULTICHAN cycle is used for multi-channel programming at lathes. It is called at the start of the block when programming with blocks.

In the cycle, e.g. the master spindle is set to the spindle specified at the start of the block. The return value (= 1) can be used to control whether the complete block is skipped.

CUST_MULTICHAN

Parameter	Meaning
_S_NR	Spindle number to which the master spindle is set.
_RET	Return value
= 0	The block is executed corresponding to the run-in mode.
= 1	The entire block is skipped.

21.4 Milling

21.4.1 General

All settings for machine and setting data of the standard cycles are also applicable for milling under ShopMill and turning under ShopTurn.

21.4.2 Technology cycles for milling

Milling function screen

SD52214 \$SCS_function_MASK_MILL		Milling function screen
Bit 0	Enable cylinder surface transformation (ShopMill)	
Bit 1	List table to accept blank (on milling machines) (This bit only has to be set, if the machine has a rotary axis and a fixed table for accepting the blank.)	
Bit 2	reserved	
Bit 3	Enable machining inside/outside	
Bit 4	Enable spindle clamping (C axis)	
Bit 5	Enable spindle control of the tool spindle via user interface	
Bit 6	Enable spindle control of the lathe spindle via user interface	

SD55214 \$SCS_FUNCTION_MASK_MILL_SET		Milling function screen
Bit 0	Basic setting, milling in climbing.	
Bit 2	Depth calculation of the milling cycles, with or without safety clearance.	
= 0	Depth calculation of the milling cycles is performed between the reference plane + safety clearance and the depth.	
= 1	Depth calculation is performed without including the safety clearance. Bit 2 is effective in the following milling cycles: CYCLE61, CYCLE71, CYCLE76, CYCLE77, CYCLE79, CYCLE899, LONGHOLE, SLOT1, SLOT2, POCKET3, POCKET4.	

Contour milling (CYCLE63)

SD55460 \$SCS_MILL_CONT_INITIAL_RAD_FIN		Finishing approach circle radius The radius of the approach circle during the finishing of contour pockets is affected.
= 0	The radius is selected so that at the starting point the safety clearance to the finishing allowance is maintained (default value).	
> 0	The radius is selected so that at the starting point the value of this channel-specific setting data to the finishing allowance is maintained.	

Multiple edge (CYCLE79), circular position pattern (HOLES2), circumferential groove (SLOT2)

SD55230 \$SCS_CIRCLE_RAPID_FEED	Rapid feed in mm/min for positioning on a circular path between the circumferential grooves or the contour elements.
= 10000	

21.4.3 Setting-up ShopMill cycles for milling**Load ShopMill cycles for milling**

When the control boots, the ShopMill cycles are automatically loaded into the following directory: ../siemens/sinumerik/cycles/sm/prog

Setting the coordinate system

MD52000 \$MCS_DISP_COORDINATE_SYSTEM	Position of the coordinate system
= 0	

Examples:

=0 for vertical milling machines

=16 for horizontal milling machines, boring mill

A detailed description of the MD52000 is provided in Section:

Activating turning/milling/drilling/grinding technologies (Page 443)

For plane selection when milling, appropriately set the following machine data:

MD52005 \$MCS_DISP_PLANE_MILL	Plane selection for milling
= 17	
= 0	In the cycle support
= 17	G17 (default value)
= 18	G18
= 19	G19

Simulation and simultaneous recording

In order that simulation and simultaneous recording are displayed without any errors, set the machine data as described in the following section:

Setting the technology for simulation (Page 103)

Defining the direction of rotation

In order that the direction of rotation is correctly displayed in the ShopMill user interface, and when programming ShopMill functions, the correct direction of rotation is executed, you must make some settings that are coordinated with one another. You must align these settings to the actual direction of rotation of the axis at the machine.

MD52207 \$MCS_USAGE_ATTRIB[n]	Attribute of the axes in the channel [n] channel axis number
Bit 4	Displayed direction of rotation for M3 is counter-clockwise (for spindles)
Bit 5	Direction of rotation M3 corresponds to rotary axis minus (for spindles) This bit must be set analog to PLC bit DBnn.DBX17.6!

Additional channel-specific machine data

MD52216 \$MCS_FUNCTION_MASK_DRILL	Drilling function screen form
Bit 2	Tapping also without encoder The setting depends on whether at the machine the tool spindle is equipped with an encoder.
= 0	Tapping only possible with encoder
= 1	Tapping also possible without encoder
Bit 3	Constant cutting speed referred to the diameter of the centering
= 0	Constant cutting speed referred to the diameter of the tool
= 1	Constant cutting speed referred to the diameter of the centering

MD52229 \$MCS_ENABLE_QUICK_M_CODES	Enable fast M commands
= 0	
Bit 0	Coolant OFF (default value)
Bit 1	Coolant 1 ON
Bit 2	Coolant 2 ON
Bit 3	Coolants 1 and 2 ON

MD52230 \$MCS_M_CODE_ALL_COOLANTS_OFF	M code for all coolants Off
= 9	You define the M function to switch off the coolant that is output when the tool is changed.

MD52231 \$MCS_M_CODE_COOLANT_1_ON	M code for coolant 1 On
= 8	You define the M function for coolant 1 that is output when the tool is changed.

MD52232 \$MCS_M_CODE_COOLANT_2_ON	M code for coolant 2 On
= 7	You define the M function for coolant 2 that is output when the tool is changed.

MD52233 \$MCS_M_CODE_COOLANT_1_AND_2_ON	M code for both coolants On
= -1	You define the M function for coolant 1 and 2 that is output when the tool is changed.

MD52281 \$MCS_TOOL_MCODE_FUNC_ON[]		M code for tool-specific function ON
= -1	M function is not output. If both M commands of a function "= -1", the corresponding field is not displayed on the interface.	
[0]	M code for tool-specific function 1 ON	
[1]	M code for tool-specific function 2 ON	
[2]	M code for tool-specific function 3 ON	
[3]	M code for tool-specific function 4 ON	

MD52282 \$MCS_TOOL_MCODE_FUNC_OFF[]		M code for tool-specific function OFF
= -1	M function is output. If both M commands of a function "= -1", the corresponding field is displayed on the interface.	
[0]	M code for tool-specific function 1 OFF	
[1]	M code for tool-specific function 2 OFF	
[2]	M code for tool-specific function 3 OFF	
[3]	M code for tool-specific function 4 OFF	

Channel-specific cycle setting data

SD55212 \$SCS_FUNCTION_MASK_TECH_SET		Cross-technology function screen
= 6		
Bit 0	Tool preselection active. The next tool is prepared directly after a tool change. Note: For a revolver, the setting data must be set to "0".	
Bit 1	Automatic calculation of the thread depth for metric threads.	
Bit 2	Take the thread diameter and thread depth from the table.	

21.4.4 Cylinder surface transformation (TRACYL)



Software option

You require the following software option in order to use this function:
"Transmit and peripheral surface transformation".

Precondition

- There must be at least one rotary axis at the machine.
- The milling tool must be radially oriented to the cylinder to be machined.

Function

The following groove machining operations can be performed with the Cylinder surface transformation functions:

- Longitudinal grooves on cylindrical bodies
- Transverse grooves on cylindrical objects
- Grooves with any path on cylindrical bodies

The path of the grooves is programmed with reference to the unwrapped, level surface of the cylinder. Programming can be realized using straight line/circle, drilling or milling cycles or contour milling (free contour programming).

There are two variants of cylinder surface transformation, i.e.

1. with groove side offset (ON)
2. without groove side offset (OFF)

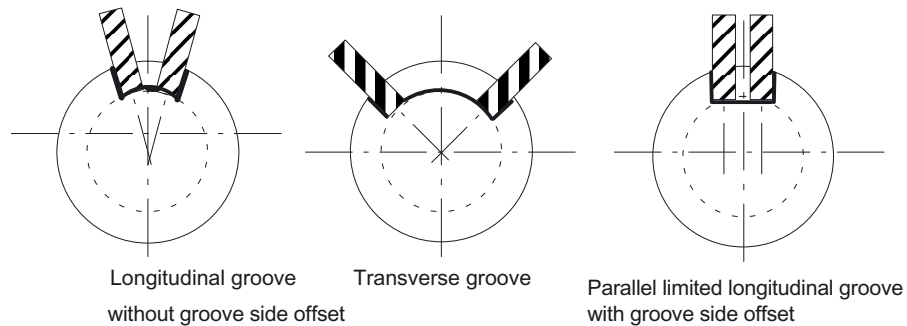


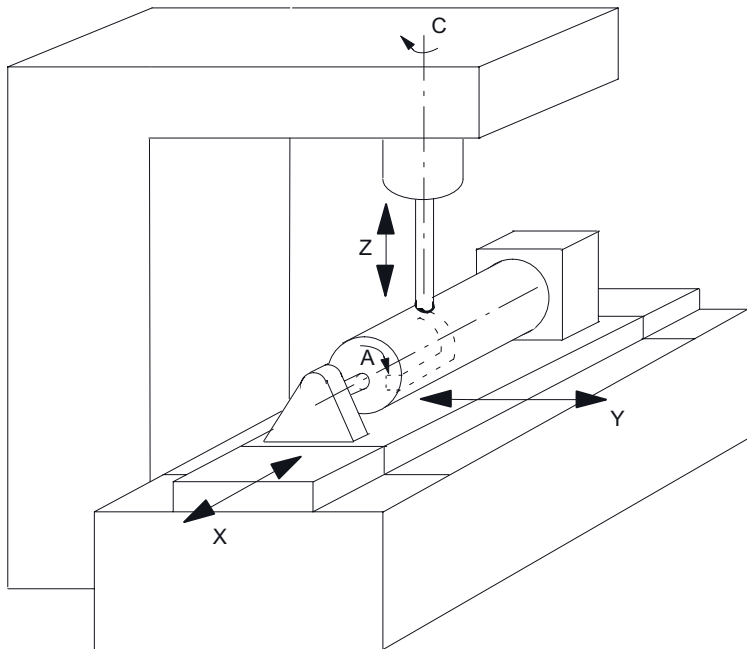
Figure 21-1 Grooves with and without groove side offset

21.4.5 Example: Milling machine with the XYZ-AC axis configuration

Example: Milling machine with the XYZ-AC axis configuration

Example of an axis configuration on the following machine:

Axis configuration



- X 1. axis of the machining plane parallel to the rotary axis
- Y 2. axis of the machining plane
- Z Infeed axis (tool axis) perpendicular (radial) to the rotary axis
- A Rotary axis
- C Working spindle

Figure 21-2 Machining slots on a cylinder surface with X-A-Z kinematics

You must configure two data records with the following machine data for the machine illustrated above:

MD20070 \$MC_AXCONF_MACHAX_USED[4]	Machine axis number valid in channel
= 5	Number of channel axes

MD20080 \$MC_AXCONF_CHANAX_NAME_TAB[]	Name of channel axis in the channel
[0] = XC	Channel axis XC, corresponds to channel axis 1
[1] = YC	Channel axis YC, corresponds to channel axis 2
[2] = ZC	Channel axis ZC, corresponds to channel axis 3
[3] = A	Channel axis A, corresponds to channel axis 4
[4] = C	Channel axis C, corresponds to channel axis 5

MD28082 \$MC_MM_SYSTEMFRAME_MASK	System frames (SRAM)
= 21H	Default setting
Bit 0 = 1	System frame for actual value setting and scratching

Bit 5 = 1	System frame for cycles
Bit 6 = 1	System frame for transformations

General settings for the transformation

MD10602 \$MN_FRAME_GEOAX_CHANGE_MODE	Frames when switching over geometry screens
= 1	The actual total frame (zero offsets) is recalculated when switching over geometry axes (selecting - deselecting TRACYL).

Data set for the 1st transformation in the channel

MD24100 \$MC_TRAFO_TYPE_1	Definition of transformation 1 in the channel.
= 512	Transformer type (512 = cylinder surface transformation without groove side offset)

MD24110 \$MC_TRAFO_AXES_IN_1		Axis assignment for the 1st transformation in the channel
[0] = 3	Channel axis: Infeed axis (tool axis) perpendicular (radial) to the rotary axis	Z
[1] = 4	Channel axis: Rotary axis	A
[2] = 1	Channel axis: 1st axis of the machining plane parallel to the rotary axis	X
[3] = 2	Channel axis: 2nd axis of the machining plane	Y

MD24120 \$MC_TRAFO_GEOAX_ASSIGN_TAB_1		Assignment of the geometry axes to channel axes for transformation 1.
[0] = 1	Channel axis: 1st geometry axis	X
[1] = 4	Channel axis: 2nd geometry axis	A
[2] = 3	Channel axis: 3rd geometry axis	Z

MD24800 \$MC_TRACYL_ROT_AX_OFFSET_1	
= 0	Offset of rotary axis for the 1st TRACYL transformation.

MD24805 \$MC_TRACYL_ROT_AX_FRAME_1	
= 1	Axial offset of rotary axis for the 1st TRACYL transformation.

MD24810 \$MC_TRACYL_ROT_SIGN_IS_PLUS_1	
= 1	Sign of the rotary axis for the 1st TRACYL transformation.

MD24820 \$MC_TRACYL_BASE_TOOL_1[]	
[0] = 0	Vector of basis tool for the 1st TRACYL transformation in XYZ

[1] = 0	
[2] = 0	

Data set for the 2nd transformation in the channel

MD24200 \$MC_TRAFO_TYPE_2	
= 513	Transformer type (513 = cylinder surface transformation with groove side offset)

MD24210 \$MC_TRAFO_AXES_IN_2		
[0] = 3	Channel axis: Infeed axis perpendicular (radial) to rotary axis	Z
[1] = 4	Channel axis: Rotary axis	A
[2] = 1	Channel axis: 1st axis of the machining plane parallel to the rotary axis	X
[3] = 2	Channel axis: 2nd axis of the machining plane	Y

MD24220 \$MC_TRAFO_GEOAX_ASSIGN_TAB_2		
[0] = 1	Channel axis: 1st geometry axis	X
[1] = 4	Channel axis: 2nd geometry axis	A
[2] = 3	Channel axis: 3rd geometry axis	Z

MD24850 \$MC_TRACYL_ROT_AX_OFFSET_2	
= 0	Offset of rotary axis for the 2nd TRACYL transformation

MD24855 \$MC_TRACYL_ROT_SIGN_IS_FRAME_2	
= 1	Offset of rotary axis for the 2nd TRACYL transformation

MD24860 \$MC_TRACYL_ROT_SIGN_IS_PLUS_2	
= 1	Sign of rotary axis for the 2nd TRACYL transformation

MD24870 \$MC_TRACYL_BASE_TOOL[]	
[0] = 0	Vector of basis tool for the 2nd TRACYL transformation in XYZ
[1] = 0	
[2] = 0	

Note**Data records for transformations**

For both data records, you can use any transformations from all available transformations (MD24100 \$MC_TRAFO_TYPE_1, MD24200 \$MC_TRAFO_TYPE_2, etc.). The two data records need not be directly next to each other.

However, the 1st data record must always be used for "Cylinder surface transformation **without** groove side offset" (= 512) and the 2nd data record for "Cylinder surface transformation **with** groove side offset" (=513).

21.4.6 ShopMill cycles for multiple clamping

**Software options**

You require the following software option in order to use this function: "ShopMill/ShopTurn"

Function

The Multiple clamping function provides optimized tool change over several workpiece clampings. On the one hand, this reduces downtimes and, on the other, eliminates tool change times because a tool performs as many machining operations as possible in all clampings before the next tool change is initiated.

You can either execute the same program several times for the clampings or you can select different programs. If you are using large-area fixture plates on your machine, you do not have to set up anything else.

In the case of rotating clamping devices, on the other hand, you must adapt a cycle to match the features of the clamping device so as to ensure that the next workpiece can be turned to the machining position after machining of the previous one (or for multiple clamping devices even while the current workpiece is being machined).

Procedure for setting up

1. Change the CLAMP.SPF cycle which is stored in the cycle/sc/prog/templates_deu or cycle/sc/prog/templates_eng directory.
2. Copy the cycle into the user or manufacturer directory.

Example

A rotating clamping device (reversible clamping device) is used with four clampings. The respective clampings can be machined by positioning rotary axis A:

Clamping 1:	A = 0°
Clamping 2:	A = 90°

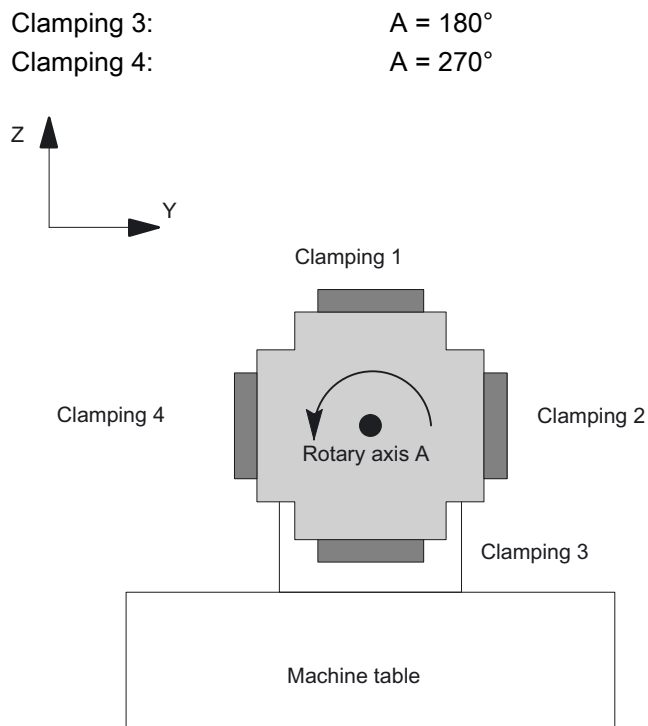


Figure 21-3 Rotating clamping device (reversible clamping device)

The cycle CLAMP.SPF must be adapted as follows:

```
...
DEF INT _NV ; Auxiliary variable
;-----
;Adaptation
;
IF _ACT==1
G0 A=DC (0)
ENDIF
;
IF _ACT==2
G0 A=DC (90)
ENDIF
;
IF _ACT==3
G0 A=DC (180)
ENDIF
;
```

```

IF _ACT==4
G0 A=DC (270)
ENDIF

;

;-----
_NV=_NPV+_ACT ; Calculate current zero offset
N10 G[8]=_NV ; There must not be a calculation here
RET

```

21.5 Turning

21.5.1 General

All settings for machine and setting data of the standard cycles are also applicable for milling under ShopMill and turning under ShopTurn.

21.5.2 Setting up cycles for turning

Setting the coordinate system

MD52000 \$MCS_DISP_COORDINATE_SYSTEM	Position of the coordinate system
= 0	Default setting

Examples:

0: Vertical lathe (carousel-type lathe)

19: Horizontal lathe, machining in front of the turning center

34: Horizontal lathe, machining behind the turning center (inclined bed lathe)

A detailed description of the MD52000 is provided in Section: "Activating turning/milling/drilling/grinding technologies (Page 443)".

Turning function screen

Enable various functions under the Turning function screen in the following channel-specific configuration machine data.

MD52218 \$MCS_FUNCTION_MASK_TURN	Turning function screen
Bit 0	Enable zoom under manual for tool measurement
Bit 1	Enable part catcher for cut-off

Bit 2	Enable tailstock
Bit 3	Enable spindle control of main spindle via user interface
Bit 4	Enable spindle control of the counterspindle control via user interface
Bit 5	Enable spindle control of the tool spindle via user interface
Bit 6	Enable balance cutting for two-channel stock removal
Bit 7	Retraction when cutting along the contour with G1
Bit 8	Enter the spindle chuck data in the program
Bit 9	Additional input of the tailstock data in the program
Bit 10	Enable crowned thread
Bit 12	Deactivate thread synchronization
Bit 13	Cutting along the contour with CYCLE95 (828D programGUIDE without Advanced Technology

Meaning of the axes

You set the meaning of the axes in machine data MD52206 \$MCS_AXIS_USAGE[n].

MD52206 \$MCS_AXIS_USAGE[n]	Meaning of the axes in the channel [n] channel axis number

A detailed description of MD52206 is provided in Chapter "Activating turning/milling/drilling/grinding technologies (Page 443)

Traversing direction

In order that the user interface and cycles know in which direction the spindles and rotary axes rotate, the following settings must be made.

You set the direction of rotation of an NC rotary axis via MD32100 \$MA_AX_MOTION_DIR.

MD32100 \$MA_AX_MOTION_DIR	Traversing direction
-1	Direction reversal
0, 1	No direction reversal

Defining the direction of rotation

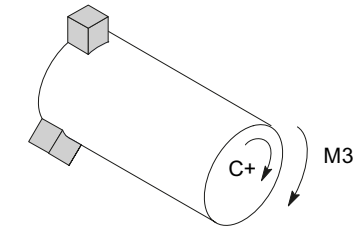
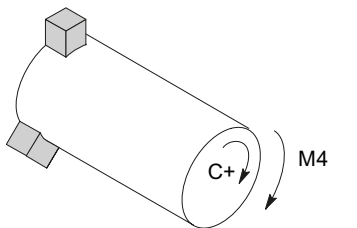
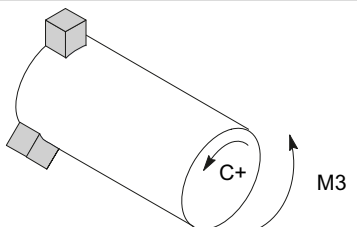
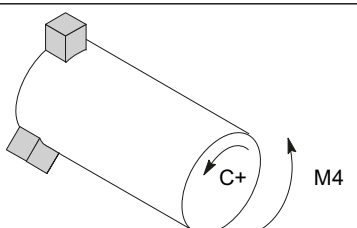
In order to ensure that the direction of rotation of the spindle and C axis is correctly displayed on the user interface and when programming, the correct direction of rotation is executed, you must make several settings that are coordinated with one another. You must base these settings on the actual direction of rotation of the spindle/C axis on the machine.

You specify the settings in the following machine data:

MD52207 \$MCS_AXIS_USAGE_ATTRIB[n]	Attributes of the axes [n] channel axis number
Bit 3	Displayed positive direction of rotation is counter-clockwise (for rotary axes) The view must be from the inside of the machine (operator's view) to the axis of rotation.

Bit 4	Displayed direction of rotation for M3 is counter-clockwise (for spindles) This bit can be set from the operator's view or according to DIN 66025.
Bit 5	Direction of rotation for M3 corresponds to minus rotary axis (for spindles) This bit must be set analogous to PLC bit DBn.DBX17.6!

As a result, the following setting options are obtained for the main spindle:

Main spindle direction of rotation	52207[n] bit 3 =	52207[n] bit 4 = *)	52207[n] bit 5 =	DB3n. DBX17.6
	0	0	0	0
	0	1	1	1
	1	1	0	0
	1	0	1	1

Note

MD52207[n] bit 4 = *)

Bit 4 can be set from the operator's view (values as specified) or according to DIN ("0" and "1" are then interchanged).

Dimensions

To define the reference point for moving the counterspindle, you must first communicate the dimensions of the counterspindle. You can either enter the dimensions in the following axis-specific cycle machine data or in the menu "Parameter" -> Setting data -> Spindle chuck data". Changes to the machine data are automatically accepted in the menu and vice versa.

MD53240 \$MAS_SPINDLE_PARAMETER[]	Spindle chuck data
[0]	Chuck dimensions
[1]	Stop dimensions
[2]	Jaw dimensions

MD53241 \$MAS_SPINDLE_CHUCK_TYPE[]	Spindle jaw type
= 0	Clamping, outer
= 1	Clamping, inner

Additional settings

The M code, e.g. M34 or M1 = 34 for the spindle chuck is defined in the following machine data. The manufacturer cycle CUST_TCHCYC.SPF also takes the M functions from the following machine data:

MD52250 \$MCS_M_CODE_CHUCK_OPEN[]	M code for open chuck with stationary spindle
[0]	Main spindle
[1]	Counterspindle

MD52251 \$MCS_M_CODE_CHUCK_CLOSE_OPEN_ROT[]	M code for open chuck with spindle rotating
[0]	Main spindle
[1]	Counterspindle

MD52252 \$MCS_M_CODE_CHUCK_CLOSE[]	M code for close chuck
[0]	Main spindle
[1]	Counterspindle

MD52253 \$MCS_M_CODE_TAILSTOCK_FORWARD[]	M code for quill forward
[0]	Quill opposite the main spindle
[1]	

MD52254 \$MCS_M_TAILSTOCK_BACKWARD[]	M code for quill back
[0]	Quill opposite the main spindle
[1]	Quill opposite the main spindle

Rounding the contour

SD55582 \$SCS_TURN_CONT_TRACE_ANGLE		Contour turning: Minimum angle for rounding the contour
= 5	Specifies the angle between the cutting edge and contour, above which for contour turning, the contour is rounded in order to remove residual material (default value).	

21.5.3 Setting-up ShopTurn cycles for turning

MD52210 \$MCS_FUNCTION_MASK_DISP		Function screen display
Bit 0 = 1	Dimension system for programs always in the basic system (default setting)	
Bit 1 = 1	Face view when turning (default setting)	
Bit 4 = 1	Display follow-on tool in the T,F,S window	

MD52212 \$MCS_FUNCTION_MASK_TECH		Cross-technology function screen
Bit 2	Approach logic for stepped drill	
Bit 3	Block search cycle for ShopMill / ShopTurn	
= 0	The E_ASUP or F_S_ASUP cycles are not called in the block search cycle PROG_EVENT.SPF	
= 1	The E_ASUP (under ShopMill) or F_S_ASUP (under ShopTurn) cycles are called in the block search cycle PROG_EVENT.SPF	
Bit 4	Approach logic using the cycle	
Bit 5	The cycle for SERUPRO (CYCLE207) is called in the block search cycle PROG_EVENT.SPF. Bit 3 has no effect when bit 5 is set.	
Bit 6	Work offset value WO cannot be entered	
Bit 7	Identification of the end of tool lives in the program (ShopMill / ShopTurn)	
Bit 8	Manual machine (ShopMill/ShopTurn)	
Bit 9	Selection/deselection of the work offset using softkey	
Bit 11	Deactivate position check for drilling and milling tools	

MD52214 \$MCS_FUNCTION_MASK_MILL		Milling function screen
Bit 3	"Inner/rear" machining is enabled in the ShopTurn screens which themselves define the machining plane.	
Bit 4	If you have implemented the "Clamp/release spindle" function using the manufacturer cycle CUST_TECHCYC.SPF, then using this machine data, you can activate the "Clamp/release spindle" parameter in the drilling and milling screens.	
= 0	The "Clamp/release spindle" parameter is not displayed in the drilling and milling screens. ShopTurn automatically clamps the spindle if it makes sense for the particular machining operation.	
= 1	The "Clamp/release spindle" parameter is displayed in the drilling and milling screens. You decide for which machining operation the spindle should be clamped.	

MD52216 \$MCS_FUNCTION_MASK_DRILL		Drilling function screen
Bit 3	Constant cutting speed referred to the diameter of the centering	
= 0	Constant cutting speed referred to the diameter of the tool	
= 1	Constant cutting speed referred to the diameter of the centering	

MD52218 \$MCS_FUNCTION_MASK_TURN		Turning function screen
Bit 2	Enable tailstock	
Bit 5	Enable spindle control of the tool spindle via user interface	
Bit 6	Enable Balance Cutting for two-channel stock removal	

MD52229 \$MCS_ENABLE_QUICK_M_CODES		Enable fast M commands
= 0		
Bit 0	Coolant OFF	
Bit 1	Coolant 1 ON	
Bit 2	Coolant 2 ON	
Bit 3	Coolants 1 and 2 ON	

MD52230 \$MCS_M_CODE_ALL_COOLANTS_OFF		M code for all coolants off
= 9	This machine data is used to define the M function for switching off the coolant; which is output when the tool is changed.	

MD52231 \$MCS_M_CODE_COOLANT_1_ON		M code for coolant 1 on
= 8	This machine data is used to define the M function for coolant 1, which is output when the tool is changed.	

MD52232 \$MCS_M_CODE_COOLANT_2_ON		M code for coolant 2 on
= 7	This machine data is used to define the M function for coolant 2, which is output when the tool is changed.	

MD52233 \$MCS_M_CODE_COOLANT_1_AND_2_ON		M code for both coolants on
= -1	This machine data is used to define the M function for coolant 1 and 2, which is output when the tool is changed.	

SD55505 \$SCS_TURN_ROUGH_O_RELEASE_DIST		Retraction distance for stock removal during external machining
= 1	Specifies the distance, by which the tool is retracted from the contour when removing stock from an outer corner. This does not apply to stock removal at a contour (default value).	
= -1	The distance is internally defined.	

SD55506 \$SCS_TURN_ROUGH_'I'_RELEASE_DIST	Retraction distance for stock removal during internal machining
= 0.5	Specifies the distance, by which the tool is retracted from the contour when removing stock from an internal corner. This does not apply to stock removal at a contour (default value).
= -1	The distance is internally defined.

21.5.4 Setting up a counterspindle



Software option

You require the following software option in order to use the counterspindle on your machine:

"Generic coupling 'CP-STATIC'" or higher

Function

If your turning machine has a counterspindle, you can machine workpieces using turning, drilling and milling functions on the front and rear faces without reclamping the workpiece manually.

Before machining the rear face, the counterspindle must grip the workpiece, pull it out of the main spindle, and position it at the new machining position.

Meaning of the axes

MD52206 \$MCS_AXIS_USAGE[n]	Meaning of the axes in the channel [n] channel axis number

A detailed description of MD52006 is provided in Chapter "Activating turning/milling/drilling/grinding technologies (Page 443)".

Traversing direction

In order that the user interface and cycles know in which direction the spindles and rotary axes rotate, the following settings must be made.

You set the direction of rotation of an NC rotary axis via MD32100 \$MA_AX_MOTION_DIR.

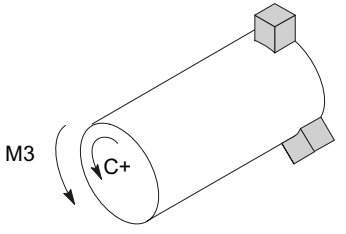
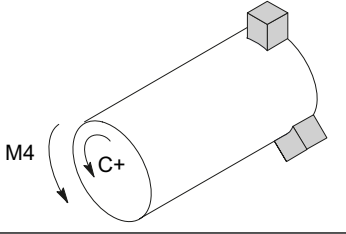
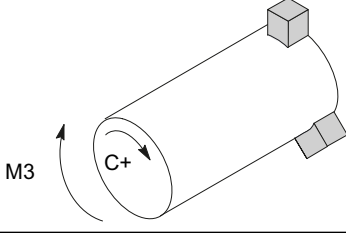
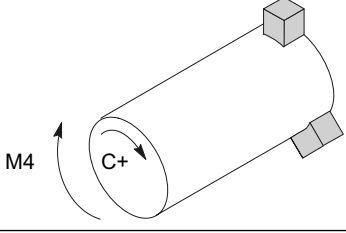
MD32100 \$MA_AX_MOTION_DIR	Traversing direction (not control direction)
-1	Direction reversal
0, 1	No direction reversal

Defining the direction of rotation

You set the directions of rotation for the user interface in the following machine data:

MD52207 \$MCS_AXIS_USAGE_ATTRIB[n]		Attributes of the axes in the channel [n] channel axis number
Bit 3	Displayed positive direction of rotation is counter-clockwise (for rotary axes) The view must be from the inside of the machine (operator's view) to the axis of rotation.	
Bit 4	Displayed direction of rotation for M3 is counter-clockwise (for spindles) This bit can be set from the operator's view or according to DIN.	
Bit 5	Direction of rotation for M3 corresponds to minus rotary axis (for spindles) This bit must be set analogous to PLC bit DBn.DBX17.6!	

As a result, the following setting options are obtained for the counterspindle:

Counterspindle direction of rotation	52207[n] bit 3 =	52207[n] bit 4 = *)	52207[n] bit 5 =	DB3n. DBX17.6
	1	1	0	0
	1	0	1	1
	0	0	0	0
	0	1	1	1

Note**MD52207[n] bit 4 = *)**

Bit 4 can be set from the operator's view (values as specified) or according to DIN ("0" and "1" are then interchanged).

The display must be the same for the main spindle and the counterspindle ("from operator's view" or "according to DIN").

Dimensions

To define the reference point for moving the counterspindle, you must first announce the dimensions of the counterspindle. You can either enter the dimensions in the following axis-specific cycle machine data or in the menu "Parameter". → "Setting data" → "Spindle chuck data". Changes to the machine data are automatically accepted in the menu and vice versa.

MD53240 \$MAS_SPINDLE_PARAMETER[]	Spindle chuck data
[0]	Chuck dimensions
[1]	Stop dimensions
[2]	Jaw dimensions

MD52241 \$MCS_SPINDLE_CHUCK_TYPE	Spindle jaw type
= 0	Clamping, outer
= 1	Clamping, inner

21.5.5 Setting up the counterspindle under ShopTurn**Software option**

You require the following software option in order to use the counterspindle on your machine:

"Generic coupling 'CP-STATIC'" or higher

The position to which the counterspindle travels when the program starts is defined in the following channel-specific cycle setting data:

SD55232 \$SCS_SUB_SPINDLE_REL_POS	Retraction position Z for counterspindle

The following channel-specific setting data becomes effective when traveling to the fixed stop:

SD55550 \$SCS_TURN_FIXED_STOP_DIST	Distance for travel to fixed stop.
SD55551 \$SCS_TURN_FIXED_STOP_FEED	Feedrate for travel to fixed stop.
SD55552 \$SCS_TURN_FIXED_STOP_FORCE	Force for travel to fixed stop in %.

Between traveling to the fixed stop and gripping, the counterspindle can retract a short distance to counteract compressive stress in the workpiece.

SD55553 \$SCS_TURN_FIXED_STOP_RETRACTION	Retraction distance before clamping after fixed stop.

After gripping you can cut off the workpiece. Before doing so, the counterspindle can retract a short distance with the workpiece to exert tensile stress on the workpiece. This relieves pressure on the tool when cutting off.

SD55543 \$SCS_TURN_PART_OFF_RETRACTION	Retraction distance before cut-off.

After cut-off, you can carry out a cut-off check and for turning, use the "Travel to fixed stop" function. You can activate/deactivate the cut-off check using the channel-specific cycle setting data:

SD55540 \$SCS_TURN_PART_OFF_CTRL_DIST	Distance for cut-off check.
SD55541 \$SCS_TURN_PART_OFF_CTRL_FEED	Feedrate for cut-off check.
SD55542 \$SCS_TURN_PART_OFF_CTRL_FORCE	Force for cut-off check in %.

The cut-off is successful when travel to fixed stop fails. The following alarms are output:

Alarm	Alarm text
20091	Axis %1 has not reached the fixed stop.
20094	Axis %1 end stop has been aborted.

You can switch off the alarm display using the following machine data:

MD37050 \$MA_FIXED_STOP_ALARM_MASK	Enabling the fixed stop alarms.
= 2	Suppressing alarms 20091 and 20094.

You can set this machine data axis-specifically in the "Machine Data" window in the "Tool zero" operating area.

If, however, the specified force is reached during the cut-off check (i.e. travel to fixed stop is successful), alarm 61255 "Error during cut-off: Tool break?" is issued.

Note

The "Travel to fixed stop" function can also be used when gripping the spindle (see above). If travel to fixed stop does not succeed when gripping, an alarm will of course still be issued. Instead of alarms 20091 and 20094, the alarm 61254 "Error during travel to fixed stop" will be issued.

Note

MD37050 can also contain values not equal to 2, e.g. $10 = 2 + 8$. Make sure that the value 2 is always contained logically so that the monitoring for travel to fixed stop can be hidden. If MD37050 does not contain the value 2, the cut-off check is not active.

Machine manufacturer cycle

If you perform one of the following actions, you must adapt the machine manufacturer cycle CUST_TECHCYC.SPF.

- Switch between spindle and C axis mode for the main spindle or counterspindle.
- Open, close, flush chuck (main spindle / counterspindle).
- Change default settings for the coupling of the main spindle and counterspindle.

Please refer to the following section for the procedure:

Setting-up ShopTurn cycles for turning (Page 471)

21.5.6 Technology cycles for turning

Thread-cutting (CYCLE99)

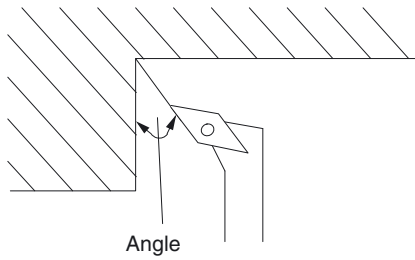
To correctly machine the thread, it is necessary to set the machine data MD52207 \$MCS_AXIS_USAGE_ATTRIB[n] with bit 3, see Chapter: Setting up a counterspindle (Page 473).

Contour grooving CYCLE930, contour turning CYCLE950, stock removal at corner CYCLE951, contour turning CYCLE952

SD55500 \$SCS_TURN_FIN_FEED_PERCENT	Enter the finishing feedrate for complete machining, roughing and finishing. The percentage of the value corresponds to that entered under parameter F (feedrate).
= 100	100% finishing feedrate

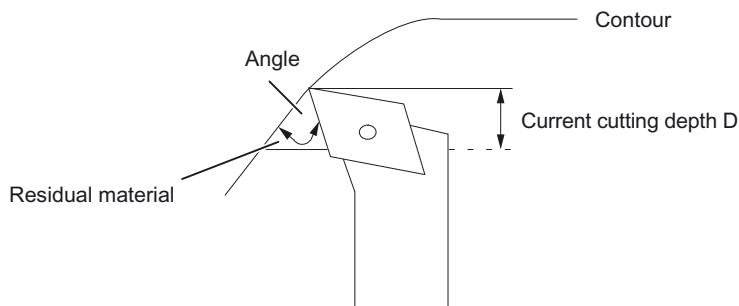
SD55510 \$SCS_TURN_GROOVE_DWELL_TIME	Dwell time, which is necessary between grooving and retracting for grooving technology. Tool clearance time during grooving at the base.
= > 0	Dwell time in seconds
= < 0	Dwell time in spindle revolutions

SD55580 \$SCS_TURN_CONT_RELEASE_ANGLE	Angle, through which the tool is lifted from the contour for contour turning, roughing.
= 45	Retraction angle of 45 degrees



SD55581 \$SCS_TURN_CONT_RELEASE_DIST		Amount, by which the tool is lifted when roughing a contour taking into account the retraction angle SD55580 \$SCS_TURN_CONT_RELEASE_ANGLE. This also applies to stock removal, grooving and plunge turning.
= 1	1 mm or 1 inch retraction distance	

SD55582 \$SCS_TURN_CONT_TRACE_ANGLE		The angle between cutting edge and contour as of which rounding is performed on the contour during contour turning in order to remove residual material. If the angle of the residual material is greater than that specified in the setting data, the tool will round the contour.
= 5	5 degree angle	



SD55583 \$SCS_TURN_CONT_VARIABLE_DEPTH		Percentage of the changing cutting depth when turning a contour You can select changing the cutting depth for stock removal and for removing residual material.
= 20	20% variable cutting depth	

SD55584 \$SCS_TURN_CONT_BLANK_OFFSET		Safety clearance when avoiding obstacles, to avoid collisions during approach or retraction from the machining. This also applies to stock removal, grooving and plunge turning.
= 1	1 mm or 1 inch blank allowance	

SD55585 \$SCS_TURN_CONT_INTERRUPT_TIME	Time for the feedrate interruption for contour turning. This also applies to stock removal, grooving and plunge turning.
= > 0	Interruption time in seconds
= < 0	Interruption time in revolutions
= 0	No interruption

Note

The channel-specific cycle setting data SD55585 is only evaluated if SD55586 \$SCS_TURN_CONT_INTER_RETRACTION = 0.

SD55586 \$SCS_TURN_CONT_INTER_RETRACTION	Retraction distance for contour turning for feedrate interruption. This also applies to stock removal, grooving and plunge turning.
= > 0	Retraction distance at feed interruption SD55585 \$SCS_TURN_CONT_INTERRUPT_TIME has no effect.
= 0	No retraction distance

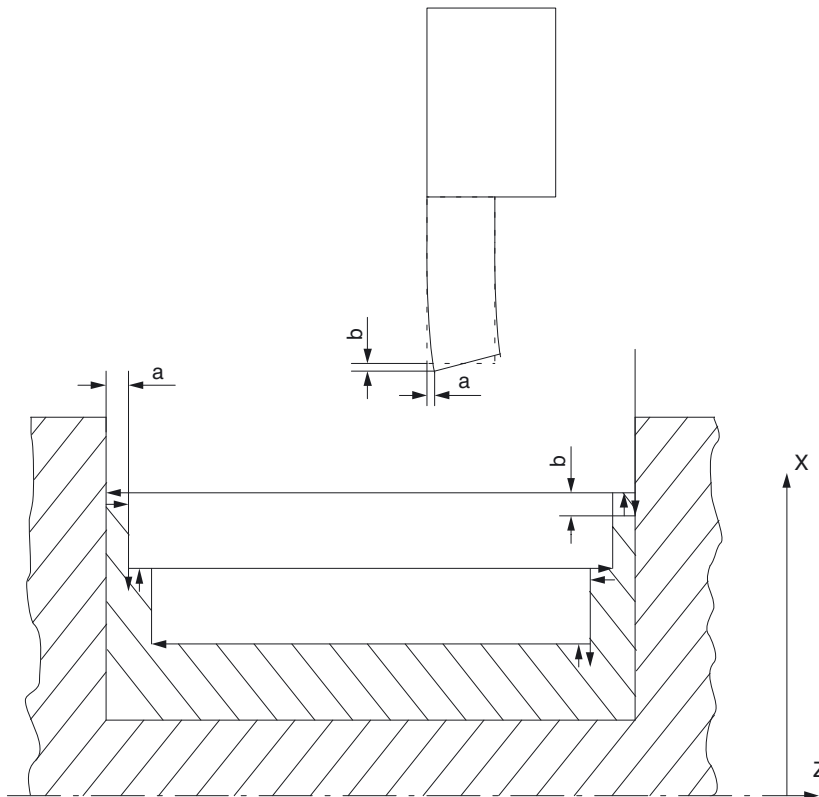
SD55587 \$SCS_TURN_CONT_MIN_REST_MAT_AX1	Limit value during contour turning for removal of residual material in the direction of axis 1 (for G18 Z). This also applies to residual machining – stock removal, grooving and plunge turning.
50	The roughing step for residual machining is not executed, if the material to be removed is less than 50% of the finishing allowance. The material is only removed when finishing.

Example: If the SD is set to 50% and the final machining allowance is 0.5 mm, any residual material less than 0.25 mm is not removed with the residual machining – but is removed during finishing. If during a machining step, less residual material is present than defined in the SD, the error message "No material present" is issued. This means that this residual material roughing step can be omitted because no machining is performed.

SD55588 \$SCS_TURN_CONT_MIN_REST_MAT_AX2	Limit value during contour turning for removal of residual material in the direction of axis 2 (for G18 X). This also applies to residual machining – stock removal, grooving and plunge turning.
50	The roughing step for residual machining is not executed, if the material to be removed is less than 50% of the finishing allowance. The material is only removed when finishing.

As the tool bends during plunge turning, the tool cannot travel right up to the contour during stock removal. The lateral distance to the last cut by which the next cut is shortened is specified in the following channel-specific cycle setting data.

SD55595 \$SCS_TURN_CONT_TOOL_BEND_RETR		Retraction distance because of tool bending
0.1	0.1 mm or 0.1 inch retraction distance	



a Distance to the last cut, SD55595

b Retraction between plunge-cutting and stock removal, SD55596

As the tool bends during plunge turning, the tool would make an excessively deep cut during stock removal. The retraction distance of the tool between plunge-cutting and stock removal is specified in the following channel-specific cycle setting data:

SD55596 \$SCS_TURN_CONT_TURN_RETRACTION		Retraction depth before turning
= 0.1	0.1 mm or 0.1 inch retraction depth	

21.5.7 Axis configuration of a lathe

General configuration

If driven milling tools are available on a lathe, then the following functions can also be set-up on this machine:

- Cylinder surface transformation (TRACYL) (Page 482)
- End face machining (TRANSMIT) (Page 485)

Lathe with X and Z axes, main and tool spindle

For example, for a lathe with X and Z axes, main spindle (C1) and tool spindle (WZ), you can configure the following channel-specific machine data:

MD20080 \$MC_AXCONF_CHANAX_NAME_TAB[]		Channel axis name in the channel
[0] = XC	Channel axis XC	
[1] = ZC	Channel axis ZC	
[2] = C1	Main spindle C1	
[3] = WZ	Tool spindle WZ	

Lathe with X and Z axes, main and tool spindle and counterspindle

For example, for a lathe with X and Z axes, main spindle (C1), tool spindle (WZ) and counterspindle (C2), you can configure the following machine data:

MD20080 \$MC_AXCONF_CHANAX_NAME_TAB[]		Channel axis name in the channel
[0] = XC	Channel axis XC	
[1] = ZC	Channel axis ZC	
[2] = C1	Main spindle C1	
[3] = WZ	Tool spindle WZ	
[5] = C2	Counterspindle C2	

Lathe with X and Z axes, main and tool spindle and Y axis

For example, for a lathe with X, Z and Y axes, main spindle (C1) and tool spindle (WZ), you can configure the following machine data:

MD20080 \$MC_AXCONF_CHANAX_NAME_TAB[]		Channel axis name in the channel
[0] = XC	Channel axis XC	
[1] = ZC	Channel axis ZC	
[2] = C1	Main spindle C1	
[3] = WZ	Tool spindle WZ	
[5] = YC	Channel axis YC	

21.5.8 Cylinder surface transformation (TRACYL)



Software option

You require the following software option in order to use this function:

"Transmit and peripheral surface transformation"

Function

Using the cylinder surface transformation function (TRACYL), you can machine the peripheral surface of a turned part.

Setting up

General settings for cylinder transformation

MD10602 \$MN_FRAME_GEOAX_CHANGE_MODE	Frames when switching over geometry axes.
= 1	The actual total frame (work offsets) and when switching over geometry axes (selecting - deselecting TRACYL) is recalculated.

MD24040 \$MC_FRAME_ADAPT_MODE	Adapting the active frames.
Bit 0 = 1	Rotations in active frames that rotate coordinate axes for which there are no geometry axes, are deleted from the active frames.
Bit 1 = 1	Shearing angles in active frames are made orthogonal.
Bit 2 = 1	Scaling factors of all geometry axes in the active frames are set to 1.

MD28082 \$MC_MM_SYSTEM_FRAME_MASK	
Bit 6 = 1	Configuration of the transformation frame for the compensation.

When setting up the functions, you can take the following channel-specific machine data into account:

MD24300 \$MC_TRAFO_TYPE_3	Cylinder surface transformation, main spindle: Transformation 3
= 512	Without groove side offset (without Y axis)
= 513	With groove side offset (with Y axis)
= 514	With or without groove side offset and Y offset

MD24400 \$MC_TRAFO_TYPE_4	Cylinder surface transformation, counter-spindle: Transformation 4
= 512	Without groove side offset (without Y axis)
= 513	With groove side offset (with Y axis)
= 514	With or without groove side offset and Y offset

Note

You must also set up other machine data for each of the individual transformations.

Cylinder surface transformation without groove side offset

MD24300 \$MC_TRAFO_TYPE_3	Definition of the 3rd transformation in the channel
= 512	TRACYL main spindle.

MD24310 \$MC_TRAFO_AXES_IN_3[]	Axis assignment for transformation 3.
[0] = 1	Perpendicular to the rotary axis XC
[1] = 3	Rotary axis (main spindle) C1
[2] = 2	Parallel to the rotary axis ZC

MD24320 \$MC_TRAFO_GEOAX_ASSIGN_TAB_3[]	Assignment of geometry axes to channel axes for transformation 3.
[0] = 1	1st channel axis X
[1] = 3	2nd channel axis Y
[2] = 2	3rd channel axis Z

MD24800 \$MC_TRACYL_ROT_AX_OFFSET_1	Offset of the rotary axis for the 1st TRACYL transformation.
= 0	

MD24805 \$MC_TRACYL_ROT_AX_FRAME_1	Axial offset of the rotary axis is taken into account during TRACYL.
= 2	

MD24810 \$MC_TRACYL_ROT_SIGN_IS_PLUS_1	Sign of the rotary axis for the 1st TRACYL transformation.
= 1	

MD24820 \$MC_TRACYL_BASE_TOOL_1[]	Vector of the basic tool for the 1st TRACYL transformation.
[0] = 0	
[1] = 0	
[2] = 0	

Setting up cylinder surface transformation under ShopTurn

In addition, set the following channel-specific configuration machine data:

MD52214 \$MCS_FUNCTION_MASK_MILL		Milling function screens
Bit 3	Enable "internal / rear" machining in ShopTurn screens that define the machining plane themselves.	
Bit 4	If you have implemented the "Clamp/release spindle" function using the machine manufacturer cycle CUST_TECHCYC.SPF, then using this machine data, you can activate the "Clamp/release spindle" parameter in the drilling and milling screens.	
= 0	The "Clamp/release spindle" parameter is not displayed in the drilling and milling screens. ShopTurn automatically clamps the spindle if it makes sense for the particular machining operation.	
= 1	The "Clamp/release spindle" parameter is displayed in the drilling and milling screens. The operator decides for which machining operation the spindle should be clamped.	

Cylinder surface transformation with groove side offset

MD24300 \$MC_TRAFO_TYPE_3		Definition of the 3rd transformation in the channel
= 513	TRACYL main spindle	

MD24310 \$MC_TRAFO_AXES_IN_3[]		Axis assignment for transformation 3.
[0] = 1	Perpendicular to the rotary axis XC	
[1] = 3	Rotary axis (main spindle) C1	
[2] = 2	Parallel to the rotary axis ZC	
[3] = 6	Parallel to the cylinder surface and perpendicular to the rotary axis ZC	

MD24320 \$MC_TRAFO_GEOAX_ASSIGN_TAB_3[]		Assignment of geometry axes to channel axes for transformation 3.
[0] = 1	1st channel axis X	
[1] = 3	2nd channel axis Y	
[2] = 2	3rd channel axis Z	

MD24800 \$MC_TRACYL_ROT_AX_OFFSET_1		Offset of the rotary axis for the 1st TRACYL transformation.
= 0		

MD24805 \$MC_TRACYL_ROT_AX_FRAME_1		Axial offset of the rotary axis is taken into account during TRACYL.
= 2		

MD24810 \$MC_TRACYL_ROT_SIGN_IS_PLUS_1	Sign of the rotary axis for the 1st TRACYL transformation.
= 1	

MD24820 \$MC_TRACYL_BASE_TOOL_1[]	Vector of the basic tool for the 1st TRACYL transformation.
[0] = 0	
[1] = 0	
[2] = 0	

21.5.9 End face machining (TRANSMIT)



Software option

You require the following software option in order to use this function:

"Transmit and peripheral surface transformation"

Function

Using the end face machining function (TRANSMIT), you can machine the end face of a turned part.

Setting up

For general settings for transformations, please refer to Chapter "Cylinder surface transformation (TRACYL) (Page 482)".

You can make additional settings in the following channel-specific machine data:

MD24100 \$MC_TRAFO_TYPE_1	End face machining, main spindle: Transformation 1
= 256	Machining without Y axis
= 257	Machining with Y axis

MD24200 \$MC_TRAFO_TYPE_2	End face machining, counterspindle: Transformation 2
= 256	Machining without Y axis
= 257	Machining with Y axis

MD24110 \$MC_TRAFO_AXES_IN_1[]	Axis assignment for the 1st transformation in the channel
[0] = 1	Perpendicular to the rotary axis XC

[1] = 3	Rotary axis (main spindle) C1
[2] = 2	Parallel to the rotary axis ZC

MD24120 \$MC_TRAFO_GEOAX_ASSIGN_TAB_1[]	Assignment of the geometry axes to channel axes for transformation 1
[0] = 1	1st channel axis X
[1] = 3	2nd channel axis Y
[2] = 2	3rd channel axis Z

MD24900 \$MC_TRANSMIT_ROT_AX_OFFSET_1	
= 0	Offset of the rotary axis for the 1st TRANSMIT transformation.

MD24905 \$MC_TRANSMIT_ROT_AX_FRAME_1	
= 2	Axial offset of the rotary axis is taken into account during TRANSMIT 1.

MD24910 \$MC_TRANSMIT_ROT_SIGN_IS_PLUS_1	
= 0	Sign of the rotary axis for the 1st TRANSMIT transformation.

MD24911 \$MC_TRANSMIT_POLE_SIDE_FIX_1	
= 1	Limitation of working range in front of / behind the pole, 1st TRANSMIT.

MD24920 \$MC_TRANSMIT_BASE_TOOL_1[]	Vector of the basic tool for the 1st TRANSMIT transformation.
[0] = 0	
[1] = 0	
[2] = 0	

TRANSMIT with real Y axis

MD24100 \$MC_TRAFO_TYPE_1	
= 257	Definition of transformation 1 in the channel: TRANSMIT main spindle.

MD24110 \$MC_TRAFO_AXES_IN_1[]	Axis assignment for the 1st transformation in the channel.
[0] = 1	Perpendicular to the rotary axis XC
[1] = 3	Rotary axis C1
[2] = 2	Parallel to the rotary axis ZC

MD24120 \$MC_TRAFO_GEOAX_ASSIGN_TAB_1[]		Assignment of the geometry axes to channel axes for transformation 1.
[0] = 1	1st channel axis X	
[1] = 3	2nd channel axis Y	
[2] = 2	3rd channel axis Z	

MD24900 \$MC_TRANSMIT_ROT_AX_OFFSET_1		
= 0	Offset of the rotary axis for the 1st TRANSMIT transformation.	

MD24905 \$MC_TRANSMIT_ROT_AX_FRAME_1		
= 2	Axial offset of the rotary axis is taken into account during TRANSMIT 1.	

MD24910 \$MC_TRANSMIT_ROT_SIGN_IS_PLUS_1		
= 0	Sign of the rotary axis for the 1st TRANSMIT transformation.	

MD24911 \$MC_TRANSMIT_POLE_SIDE_FIX_1		
= 1	Limitation of working range in front of / behind the pole, 1st TRANSMIT.	

MD24920 \$MC_TRANSMIT_BASE_TOOL_1[]		Vector of the basic tool for the 1st TRANSMIT transformation.
[0] = 0		
[1] = 0		
[2] = 0		

Additional settings for rotary axes

If you want to integrate rotary axes, in addition to the following machine data you also require MD28082 with bit 6, see Chapter "Cylinder surface transformation (TRACYL) (Page 482)".

MD24906 \$MC_TRANSMIT_BASE_TOOL_COMP_1		Compensation of the basic offset in the 1st transformation frame.
Bit 0	1st geometry axis	
Bit 1	2nd geometry axis	
Bit 2	3rd geometry axis	

MD24956 \$MC_TRANSMIT_BASE_TOOL_COMP_2		Compensation of the basic offset in the 2nd transformation frame.
Bit 0	1st geometry axis	
Bit 1	2nd geometry axis	
Bit 2	3rd geometry axis	

For further information, see also Chapter "Clamping a blank (Page 106)".

Face machining under ShopTurn

If you wish to use the face machining function under ShopTurn function, set the following additional channel-specific configuration machine data:

MD52214 \$MCS_FUNCTION_MASK_MILL		Function screens, ShopTurn.
Bit 3	Enable "internal / rear" machining in ShopTurn screens that define the machining plane themselves.	
Bit 4	If you have implemented the "Clamp/release spindle" function using the machine manufacturer cycle CUST_TECHCYC.SPF, then using this machine data, you can activate the "Clamp/release spindle" parameter in the drilling and milling screens.	
= 0	The "Clamp/release spindle" parameter is not displayed in the drilling and milling screens. ShopTurn automatically clamps the spindle if it makes sense for the particular machining operation.	
= 1	The "Clamp/release spindle" parameter is displayed in the drilling and milling screens. The operator decides for which machining operation the spindle should be clamped.	

Note

The face end machining is automatically integrated in the cycles, with the exception of the straight line and circle.

You can select the functions for these two cycles in the "Program" operating area at "Straight line" and "Circle".

References

Additional information on face end machining is provided in:

Function Manual, Extended Functions; Kinematics Transformation (M1): TRANSMIT

21.5.10 Inclined Y axis (TRAANG)



Software option

You require the following software option in order to use the inclined Y axis function on your machine:

"Inclined axis"

Function

If your lathe has an inclined Y axis (i.e. this axis is not perpendicular to axes X and Z), you can still completely program machining operations in Cartesian coordinates. The control uses the inclined axis function (TRAANG) to transform the Cartesian coordinates to the motion of the inclined axis.

Setting up

You still have to set up the inclined axis function (TRAANG) via machine data.

Inclined axis under ShopTurn

Please also note the following:

Note

In the user interface, the inclined axis function is automatically integrated in the cycles after setting up. For machining with inclined axis, in the screen forms of the machining plane, you can select "Face Y" or "Peripheral surface Y" and enter the Cartesian coordinates.

You will find additional information on the inclined Y axis function in:

References

- Function Manual, Extended Functions; Kinematic Transformations (M1): TRAANG
- 840D sl Programming Manual Job Planning: Inclined axis (TRAANG)

Example

For example, for a turning machine with X and Z axes and inclined Y axis, main spindle (C) and tool spindle (WZ), you must configure the following machine data:

MD20050 \$MC_AXCONF_GEOAX_ASSIGN_TAB[]		Assignment of geometry axis to channel axis.
[0] = 1	1. real geometry axis X axis.	
[1] = 0	2. real geometry axis, Y axis not available.	
[2] = 2	3. real geometry axis, Z axis.	

MD20110 \$MC_RESET_MODE_MASK	
Bit 0 = 1	TRAANG is retained after ramp-up.
Bit 7 = 0	

MD20112 \$MC_START_MODE_MASK	
Bit 7 = 1	TRAANG is retained after "Cycle start".

MD20118 \$MC_GEOAX_CHANGE_RESET	
= 1	Allow automatic geometry axis change.

MD20140 \$MC_TRAFO_RESET_VALUE	
= 5	TRAANG always active after reset.

MD20144 \$MC_TRAFO_MODE_MASK	
Bit 0 = 1	TRAANG runs in the background (persistent) and is not shown on the user interface.

MD20070 \$MC_AXCONF_MACHAX_USED[4]	
= 5	Channel axis YC = 5th machine axis.

MD20080 \$MC_AXCONF_CHANAX_NAME_TAB[]		Name of channel axis in the channel.
[0] = XC	1. channel axis XC	
[1] = ZC	2. channel axis ZC	
[2] = C	3. channel axis C	
[3] = WZ	4. channel axis WZ	
[4] = YC	5. channel axis YC	

Data set for inclined axis:

MD24430 \$MC_TRAFO_TYPE_5	
= 1024	Transformation 5: TRAANG

MD24432 \$MC_TRAFO_AXES_IN_5[]		Axis assignment for transformation 5.
[0] = 5	1. transformation axis = channel axis, YC	
[1] = 1	2. transformation axis = channel axis, XC	
[2] = 2	3. transformation axis = channel axis, ZC	

MD24434 \$MC_TRAFO_GEOAX_ASSIGN_TAB_5[]		Assignment of geometry axes to channel axes for transformation 5.
[0] = 1	1. axis = channel axis, XC	
[1] = 5	2. axis = channel axis, YC	
[2] = 2	3. axis = channel axis, ZC	

MD24436 \$MC_TRAFO_INCLUDES_TOOL_5	
= 0	Tool handling with active transformation 5.

MD24700 \$MC_TRAANG_ANGLE_1	
= 55	Angle between 1st and 2nd transformer axis. Data record for linking (TRACON) of end face machining on main spindle (TRANSMIT) and inclined axis (TRAANG).

Data record for linking (TRACON) of end face machining on main spindle (TRANSMIT) and inclined axis (TRAANG):

MD24440 \$MC_TRAFO_TYPE_6	
= 8192	Type of transformation that is available as sixth in the channel.

MD24444 \$MC_TRAFO_GEOAX_ASSIGN_TAB_6[]		Assignment of geometry axes to channel axes for transformation 6.
[0] = 1	1. axis = channel axis, XC	

[1] = 3	2. axis = channel axis, YC
[2] = 2	3. axis = channel axis, ZC

MD24995 \$MC_TRACON_CHAIN_1[]	Transformation linking
[0] = 1	Number of the TRANSMIT transformation (main spindle) for linking.
[1] = 5	Number of the TRAANG transformation for linking. Data set for linking (TRACON) of cylinder surface transformation on main spindle (TRACYL) and inclined axis (TRAANG).

Data record for linking (TRACON) of cylinder surface transformation on main spindle (TRACYL) and inclined axis (TRAANG):

MD24450 \$MC_TRAFO_TYPE_7	
= 8192	Type of transformation 7 in the TRACON channel.

MD24454 \$MC_TRAFO_GEOAX_ASSIGN_TAB_7[]	Assignment of geometry axes to channel axes for transformation 7.
[0] = 1	1. axis = channel axis, XC
[1] = 3	2. axis = channel axis, YC
[2] = 2	3. axis = channel axis, ZC

MD24996 \$MC_TRACON_CHAIN_2[]	Transformation linking
[0] = 3	Number of the TRACYL transformation (main spindle) for linking.
[1] = 5	Number of the TRAANG transformation for linking.

21.6 grinding

General

Grinding tools are identified by tool type 400, 410; dressing tools are identified by tool type 490. The "New tool" function offers these types for selection. The tool list shows the "Grinding disk" and "Dressing tool" tool types listed according to their tool type. To generate the grinding disk correction, a tool setter creates a dressing program. Depending on the complexity, the tool setter makes use of external or internal code generators. This program will be assigned later to a grinding disk.

Base variant

The dressing is performed in the JOG and AUTOMATIC operating modes, where in automatic operation, the dressing is called from a part program. To parameterize the dressing, a diagnostic screen form deployed in manual mode and in the program editor is provided.

The base package offers predefined write contours. The dressing is supported with supplied screen forms and system cycles. These system cycles are available independent of and separate from the machining cycles package.

Oscillation cycles

These grinding cycles permit the machining of workpieces with grinding machines using two geometry axes.

The machines type does not exist because the cycles create only an oscillating movement that can be used as required on every machine.

This requires a SINUMERIK control as well as fast inputs/outputs for the program processing.

The cycles package provides the following cycles:

- CYCLE4071: Longitudinal grinding with infeed at the reversal point
- CYCLE4072: Longitudinal grinding with infeed at the reversal point and measurement control
- CYCLE4073: Longitudinal grinding with continuous infeed
- CYCLE4074: Longitudinal grinding with continuous infeed and measurement control
- CYCLE4075: Surface grinding with infeed at the reversal point
- CYCLE4077: Surface grinding, plunge grinding with measurement control
- CYCLE4078: Surface grinding with continuous infeed
- CYCLE4079: Surface grinding with intermittent infeed

References

For further information about oscillating cycles, please refer to:

Programming Manual, Production Planning; Chapter "Programming cycles externally" > "Technological cycles"

Grinding function screen

MD51840 \$MNS_GRIND_FUNCTION_MASK		Grinding cycles function screen
Bit 0	Reference point for measuring the dresser/wheel is a work offset	
= 0	Deselection of the work offset as reference point	
= 1	Selection of the work offset as reference point	

MD52840 \$MCS_GRIND_FUNCTION_MASK		Grinding cycles function screen
Bit 7	Retraction when form-truing with G1	
= 0	Retraction when form-truing with rapid traverse (G0)	
= 1	Retraction when form-truing with feedrate (G1)	

SD54840 \$SNS_GRIND_FUNCTION_MASK	Grinding cycles function screen
SD55840 \$SCS_GRIND_FUNCTION_MASK	Grinding cycles function screen

Form-truing (CYCLE495)

SD55880 \$SCS_GRIND_CONT_RELEASE_ANGLE	Angle of retraction
	This setting data is used to specify the angle of retraction from the contour during form-truing.

SD55881 \$SCS_GRIND_CONT_RELEASE_DIST	Retraction distance
	This setting data is used to specify the distance retracted in both axes during form-truing.

SD55884 \$SCS_GRIND_CONT_BLANK_OFFSET	Blank allowance
	This setting data is used to specify the distance to the grinding wheel as of which the switch-over is made from G0 to G1 during form-truing.

21.7 Swiveling

21.7.1 Technology cycles for swiveling

Requirement

The commissioning of the kinematic chain of the machine is a mandatory requirement for correct swiveling functionality (CYCLE800). The kinematic chain is stored in the tool parameters \$TC_CARR1 to \$TC_CARR65.

Note

The vectors of the kinematic chain can be determined with the "Measure kinematics" measuring function (CYCLE996).

To activate the swivel function, there must be at least one toolholder that can be oriented (swivel data set) in the NCK, and the workpiece, tool and rotary table reference system frames must be activated:

MD18088 \$MN_MM_NUM_TOOL_CARRIER	Maximum number of definable toolholders
> 0	

MD28082 \$MC_MM_SYSTEM_FRAME_MASK	System frames (SRAM)
= 3DH	
Bit 2 = 1	TCARR and PAROT

Bit 3 = 1	TAROT and TOFRAME
Bit 4 = 1	Workpiece reference points
Bit 5 = 1	System frame for cycles

Note

Changing machine data MD18088 and MD28082 causes the buffered memory to be reorganized.

After changing the machine data, a series start-up file must be generated and downloaded, otherwise, it can be assumed that data will be lost.

To change machine data, you require the following authorization: Access level 1 (manufacturer).

References

Current information:

- "siemensd.txt" file in the supplied software (standard cycles) or
- Description of Functions Basic Functions; Tool Offset (W1): Inclined surface machining with 3 + 2 axes
- Programming Manual, measuring cycles: CYCLE996

Activating the swivel function

The swivel function is enabled on the user interface via the following channel-machine data:

MD52212 \$MCS_FUNCTION_MASK_TECH	Cross-technology function screen
Bit 0 = 1	Enable swivel

Configuring input screens

You can configure the input screen for swiveling using the following channel-specific cycle setting data. The setting data is effective for all of the declared swivel data sets.

SD55221 \$SCS_FUNKTION_MASK_SWIVEL_SET	Function screen, swiveling CYCLE800
Bit 0	Input field "No swiveling"
= 0	Hide
= 1	Display
Bit 1	Select text when retracting "Z, ZY" or "Fixed position 1/2"
= 0	Display text Z = "Z", display text Z, XY = "Z,XY"
= 1	Display text Z = "Fixed point 1", Display text Z, XY = "Fixed point 2". If you wish to modify the retraction version "Z" or "Z, XY" via the manufacturer cycle CUST_800.SPF, the neutral text "Fixed point 1" and "Fixed point 2" can be displayed.
Bit 2	Select, permit "deselection" of the swivel data set
= 0	If deselection is not permitted, the "Swivel data set" (TC) selection field is not displayed in the "Swivel" input screen.

= 1	Deselection permitted see also swivel data set parameter \$TC_CARR37 HUNDRED MILLIONS position
Bit 3	Displays the active swivel plane under swivel in JOG. The setting in the swivel function screen acts on all swivel data sets.
Bit 4	Positioning in the basic position (pole position) of the kinematics
= 0	Evaluation of the input values in pole position of the machine kinematics
= 1	Compatibility
Bit 5	Tool alignment absolute or via frame calculation
= 0	Tool alignment via frame calculation (TCOFRY for G18)
= 1	Tool alignment absolute (TCOABS)
Bit 6	Do not list swivel mode "direct" under swivel in JOG
Bit 7	Swivel plane: Direction selection in basic position of the kinematic
= 0	With direction selection + or -, both calculated solutions for the rotary axes are approached
= 1	With direction selection + or -, only one of the calculated solutions for the rotary axes is approached
Bit 8	Traverse the zero offset for the rotary axes of the swivel data set as offset
= 0	Only take the zero offset for the rotary axes of the swivel data set into account in the WCS (compatibility)
= 1	Traverse the zero offset for the rotary axes of the swivel data set as offset Bit 8 is set in conjunction with MD21186.

Additional settings

For the swivel function, set the following machine data as a minimum to the following - although this deviates from the default value:

MD10602 \$MN_FRAME_GEOAX_CHANGE_MODE	
= 1	The actual total frame (zero offsets) is recalculated when switching over geometry axes (selecting/deselecting TRAORI).

MD10760 \$MN_G53_TOOLCORR	
Method of operation with G53, G153 and SUPA	
Bit 0 = 1	If the machine axis/axes are to be retracted before swiveling, see Chapter "Manufacturer cycle CUST_800.SPF (Page 521)"

MD11450 \$MN_SEARCH_RUN_MODE	
Settings, block search	
Bit 1 = 1	Activate PROG_EVENT.SPF after block search. This means that for a block search, the rotary axes of the active swivel data set are pre-positioned.

Declare swivel data sets

For the swivel function, swivel data sets must be created in the NCK and declared in the following machine data.

MD18088 \$MN_MM_NUM_TOOL_CARRIER	Maximum number of definable toolholders that can be oriented. Number of swivel data sets

If several channels are declared in the NCU, the number of swivel data sets is split up, taking MD28085 \$MN_MM_LINK_TOA_UNIT into account.

Example 1:

Machine has two channels with different TO units.

Channel 1: MD28085 \$MN_MM_LINK_TOA_UNIT= 1

Channel 2: MD28085 \$MN_MM_LINK_TOA_UNIT= 2

Three swivel data sets (SDS) are required for each TO unit.

$MD18088 = (\text{number of TO units}) \times (\text{number of SDS per TO unit}) = 2 \times 3 = 6$

Example 2:

Machine has three channels with two different TO units.

Channel 1: MD28085 \$MN_MM_LINK_TOA_UNIT= 1

Channel 2: MD28085 \$MN_MM_LINK_TOA_UNIT= 2

Channel 3: MD28085 \$MN_MM_LINK_TOA_UNIT= 1

One swivel data set (SDS) is required for each TO unit.

$MD18088 = (\text{number of TO units}) \times (\text{number of SDS per TO unit}) = 2 \times 1 = 2$

Using angular tools

Angular tools are created and managed in the NC with tool type 130. The tool lengths are entered in the tool of tool type 130. If an angular tool is also used on a swiveled machining plane (CYCLE800), the basic tool orientation must be enabled with MD18114 \$MM_ENABLE_TOOL_ORIENT = 2. This enables a direction vector to be transferred to the tool in the tool parameters \$TC_DPV3[n] to \$TC_DPV5[n]. The TOROT (G17), TOROTY (G18), and TOROTX (G19) commands must be programmed for the tool change so that the basic tool orientation is updated.

n = internal tool number

MD18114 \$MM_ENABLE_TOOL_ORIENT	Assign orientation to cutting edges. Basic tool orientation
= 0	No basic tool orientation active
= 2	Basic tool orientation active Used for angular tools and swiveling

MD20110 \$MC_RESET_MODE_MASK	Initial control setting for RESET and end of part program
Bit 14 = 1	Calculation of the basis and system frames, example: 4041H.

MD20112 \$MC_START_MODE_MASK	Basic control setting for START and part program.
400H	

MD20126 \$MC_TOOL_CARRIER_RESET_VALUE	Active toolholder at RESET
= 0	No active toolholder at RESET
= > 0 (n)	Toolholder with number n active at RESET MD20126 is written to in CYCLE800 CYCLE800() corresponds to deselect toolholder (MD20126 = 0)

MD20150 \$MC_GCODE_RESET_VALUES[]	Initial setting of G group.
[41] = 1	Initial setting of G group 42 on TCOABS
[51] = 2	Initial setting of G group 52 on PAROT
[52] = 1	Initial setting of G group 53 on TOROTOF
[52] = > 1	Initial setting of G group 53 on TOROT, TOROTY or TOROTX Used for machine kinematics, types "T" and "M" See parameter \$TC_CARR34

Note

If, after a RESET from the NC, a frame must be calculated in the tool direction, then MD20150 \$MC_GCODE_RESET_VALUES[52] can be set to a value > 1.

Applications:

- Machine kinematics with Hirth joint
- Angular tool with basic tool orientation

Note on kinematics (swivel head / mixed kinematics) with Hirth gearing:

Depending on the active plane (G17,G18,G19), the TOROT command (or TOROTX, TOROTY) is programmed in the NCU (G group 53) to calculate the compensating frame for the Hirth joint in CYCLE800. If the Hirth joint causes the programmed rotation to deviate from the possible positions of the rotary axes, a \$P_TOOLFRAME compensating frame is created for swivel head and mixed kinematics (see HMI active ZO / details tool reference).

If the compensating frame must be retained after RESET or end of part program, enter the following value in the channel-specific machine data:

MD20150 \$MC_GCODE_RESET_VALUES[52]	Reset behavior of G groups
= 2	For G17 (TOROT)

= 3	For G18 (TOROTY)
= 4	For G19 (TOROTX)

MD20196 \$MC_TOCARR_ROTAX_MODE	Rotary axis mode for toolholders with orientation capability
Bit 0 = 1	Swivel data set with a rotary axis used for a rotary table with C axis
Bit 1 = 1	Swivel data set with two rotary axes used for standard swiveling

MD20360 \$MC_TOOL_PARAMETER_DEF_MASK	Tool parameter setting
Bit 10 = 1	The orientation vector remains at T0 or D0 (no tool) Used for machine kinematics, types "T" and "M" See parameter \$TC_CARR34
Bit 19 = 1	The changed orientation vector remains at T0 or D0 (no tool) Bit 19 acts only with bit 10 = 1

MD21186 \$MC_TOCARR_ROT_OFFSET_FROM_FR	Offset of the rotary axes for a tool holder with orientation capability from the zero offset of the rotary axis.
= 0	In CYCLE800, the WCS is recalculated when there is a value in the zero offset (ZO) of the rotary axes. If values are entered in the zero offset of the rotary axes of the active swivel data set, the basic position of the swivel data set and therefore also the basic position of the WCS can change.
= 1	A value in the ZO of the rotary axes acts as offset of the tool holder that has orientation capability. The Work remains unchanged.

MD21186 may not be rewritten in a program with call CYCLE800.

The setting MD21186 = 0 is only suitable for a swivel data set in which one of the available rotary axes rotates exactly around the tool axis. In this way, a useful rotation of the WCS can be calculated in CYCLE800 when values are entered in the zero offset of the rotary axes. For example, for a swivel table with rotary axis A around X and rotary axis C around Z for a basic position of the kinematics in G17.

If several swivel data sets are declared per channel, and if machine functions need to be activated on changeover between swivel heads or tables, an M command can be issued in the PLC program on switchover to another swivel data set.

MD22530 \$MC_TOCARR_CHANGE_M_CODE	M code for swivel data set change
= 0	No swivel data set change
< 0	M code + number of the swivel data set for the swivel data set change

Example

Number of swivel data sets in channel 1	= 2
MD22530 \$MC_TOCARR_CHANGE_M_CODE	= -800
Programming swivel data set 1 (TCARR=1)	= M801
Programming swivel data set 2 (TCARR=2)	= M802

With the output of the M commands, the PLC can limit or invert the spindle speed or clamp or release the rotary axes, for example.

Setting the workpiece, tool and rotary table reference

Using the following machine data, you can set the workpiece, tool and rotary table reference system frames or you can influence the behavior of the system frames.

Application: System frames can be active after Reset or Power On, in order to retract a drill from a swiveled position without causing a collision, for example.

MD24006 \$MC_CHSFRAME_RESET_MASK	Active system frames after RESET
Bit 4	System frame workpiece reference
= 0	Not active
= 1	Remains active

MD24007 \$MC_CHSFRAME_RESET_CLEAR_MASK	Clear system frames after RESET
Bit 4	System frame workpiece reference
= 0	Do not delete
= 1	Delete

When used for measuring or swiveling in JOG, the workpiece reference must be active at RESET and not cleared (cascaded measuring).

MD24006 \$MC_CHSFRAME_RESET_MASK	Active system frames after RESET
Bit 4 = 1	System frame for workpiece reference remains active after RESET

MD24007 \$MC_CHSFRAME_RESET_CLEAR_MASK	Clear system frames after RESET
Bit 4 = 0	Do not clear system frame workpiece reference after RESET

MD24008 \$MC_CHSFRAME_POWERON_MASK	Reset system frames after Power On.
Bit 2	System frame rotary table reference (PAROT)
= 0	Do not reset
= 1	Reset
Bit 3	System frame tool reference (TOROT,...)
= 0	Do not reset
= 1	Reset
Bit 4	System frame workpiece reference

= 0	Do not reset
= 1	Reset

MD24080 \$MC_USER_FRAME_POWERON_MASK	Settings for settable frames.
Bit 0	
= 0	Settable zero offset via Power On not active.
= 1	Last active settable zero offset remains active after power on if MD20152 \$MC_GCODE_RESET_MODE[7] = 1

Application: Zero offset G5xx, including all rotations, should remain active after Power On.

MD28082 \$MC_MM_SYSTEM_FRAME_MASK	Setting-up system frames (SRAM)
Bit 2 = 1	Rotary table reference (\$P_PARTFRAME)
Bit 3 = 1	Tool reference (\$P_TOOLFRAME)
Bit 4 = 1	Workpiece reference (\$P_WPFRAME)

MD28083 \$MC_MM_SYSTEM_DATAFRAME_MASK	Setting-up data management, system frames (SRAM)
Bit 2 = 1	Rotary table reference (\$P_PARTFRAME)
Bit 3 = 1	Tool reference (\$P_TOOLFRAME)
Bit 4 = 1	Workpiece reference (\$P_WPFRAME)

MD28085 \$MC_MM_LINK_TOA_UNIT	Assigning a TO unit to a channel (SRAM)
If several channels are set-up, then the following can be set in the machine data: Assignment of the TO units (tools and orientable toolholders) to the NC channel. One TO unit can be assigned to several channels. - OR - Each NC channel is assigned one TO unit.	

Axial machine data for the modulo rotary axes of the swivel data set

MD30455 \$MA_MISC_FUNCTION_MASK	Axis functions
Bit 0	Modulo rotary axis programming
= 0	No modulo rotary axis programming (e.g. 0 to 359.999 degrees)
= 1	Modulo rotary axis programming (e.g. -180 to 180 degrees)
Bit 2	Positioning, rotary axis
= 0	As programmed
= 1	Along the shortest path
Application: With the setting bit 2=1, the rotary axis C travels along the shortest path for G90 with DC. For further information, refer to Chapter "Manufacturer cycle CUST_800.SPF (Page 521)".	

MD32010 \$MA_JOG_VELO_RAPID[AX] AX = axis name	Rapid traverse in JOG Rotary and machine axes that should travel for swiveling in JOG.
= 10000	Rapid traverse in JOG mode for swivelling in JOG

SD42980 \$SC_TOFRAME_MODE	Setting, frame definition for TOROT, PA-ROT
= 2000	Swivel (default value)

SD42974 \$SC_TOCARR_FINE_CORRECTION	Fine offset TCARR (swivel data set)
= 0	No fine offset of the swivel data set vectors
= 1	Fine offset of the swivel data set vectors The parameters of the swivel data set from \$TC_CARR41[n] n...number of the swivel data set and higher apply

Swiveling in the JOG mode

Cycle alarms 62186 and 62187 can be hidden or displayed using the following cycle machine data:

MD55410 \$MC_MILL_SWIVEL_ALARM_MASK	Activate fault evaluation CYCLE800
Bit 0	Activate fault 61186
= 0	Hide fault 61186 "Active zero offset G%4 and base (base reference) contains rotations" (default setting).
= 1	Display fault 61186
Bit 1	Activate fault 61187
= 0	Hide fault 61187 "Active base and base reference (G500) contain rotations" (default setting).
= 1	Display fault 61187

21.7.2 CYCLE800 checklist for the identification of the machine kinematics

Note

Identification of the machine kinematics (kinematic chain) according to DIN 66217 or ISO 841-2001

This checklist does not claim to be complete.

- **Do the three linear axes of the machine that are active for the transformation form an orthogonal coordinate system?** Geometry axes XYZ
- **How many swivel kinematics does the machine have?**
Combinations of two (or one) rotary axis and the three linear axes are always formed.

- **Which kinematics type is it?**
Swivel head, swivel table or mixed kinematics of swivel head and swivel table.
- **What are the names of the rotary axes of the kinematics?**
Manual rotary axes are permitted and do not have to be declared in the NC.
- **What is the 1st or 2nd rotary axis of a swivel data set?**
Rule: Rotary axis 2 is based on rotary axis 1. With mixed kinematics, rotary axis 1 is always the axis for the tool orientation.
- **Is the traversing direction of the linear axes and the rotary axes correct?** Right-hand rule
Rule: If the linear axis or the rotary axis moves the workpiece, the direction of motion of the axis and also the sign of the rotary axis vector change.
- **What is the initial setting of the kinematics?**
This defines the tool orientation and the plane G17, G18, G19.
- **Which rotary axis rotates around which axis of the coordinate system or the machine axis (axes)?**
This defines the rotary axis vectors of the kinematics.
Example 1:
Head kinematics. Rotary axis 2 rotates around axis Y → rotary axis vector V2xyz = 0,1,0
Example 2:
Table kinematics. Rotary axis 1 rotates around axis X → rotary axis vector V1xyz = -1,0,0

21.7.3 Commissioning swivel data (kinematics chain)

Defining a swivel data set

You must create a swivel data set for every swivel head, swivel table or each combination of both.

Swivel data sets can be declared in several channels.

A swivel data set comprises the parameters \$TC_CARR1[n] to \$TC_CARR65[n]
n = number of the swivel data set.

The parameters of the swivel data set (\$TC_CARR1[n] to \$TC_CARR65[n]) can be read-in and read-out in the startup operating area. Programming with appropriate value assignment is also possible in an NC program (manufacturer cycle). The parameters of the swivel data set are immediately effective after the program has started.

The following softkeys are assigned to the Swivel function in the "Program" → "Various" operating section:

Milling technology	Turning technology	
"Swivel plane"	"Swivel plane"	
"Advance milling tool"	"Swivel tool"	→ "Align turning tool"
		→ "Align milling tool"
		→ "Advance milling tool"

The "Align turning tool" and "Align milling tool" softkeys are only displayed if the "B-axis kinematics" function has been activated via \$TC_CARR37[n].

Offset vectors I1 to I4

The vectors always contain three components which represent the reference to the machine axes (X, Y, Z). The positions in the kinematic chain are measured by the machine manufacturer; they are always relevant with respect to a swivel head / swivel table (swivel data set). Offset vectors I1 to I4 refer to the non-swiveled state of the rotary axes (machine kinematics basic setting).

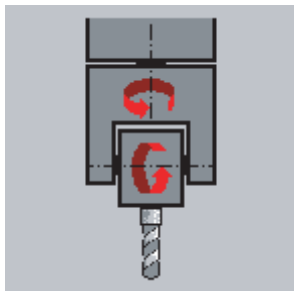
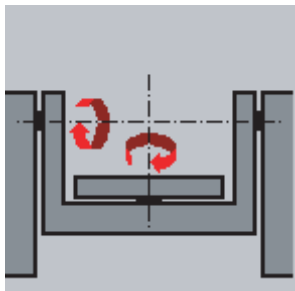
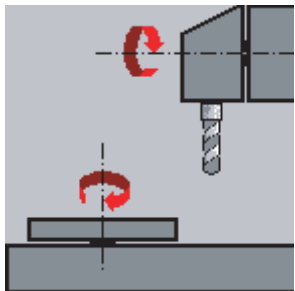
The machine kinematics used do not need to be fully implemented. However, be aware that the traversing range in the swivel planes may be restricted. If machine kinematics are to be implemented with just one rotary axis, this must always be declared as the 1st rotary axis.

\$TC_CARR1[n],	\$TC_CARR2[n],	\$TC_CARR3[n]	Offset vector I1xyz
\$TC_CARR4[n],	\$TC_CARR5[n],	\$TC_CARR6[n]	Offset vector I2xyz
\$TC_CARR15[n],	\$TC_CARR16[n],	\$TC_CARR17[n]	Offset vector I3xyz
\$TC_CARR18[n],	\$TC_CARR19[n],	\$TC_CARR20[n]	Offset vector I4xyz

Rotary axis vectors V1 and V2

\$TC_CARR7[n],	\$TC_CARR8[n],	\$TC_CARR9[n]	Rotary axis vector V1xyz
\$TC_CARR10[n],	\$TC_CARR11[n],	\$TC_CARR12[n]	Rotary axis vector V2xyz

Kinematic types \$TC_CARR23[n]

Swivel head (type T)	Swivel table (type P)	Swivel head + swivel table (type M)
		
Offset vector I1	Offset vector I2	Offset vector I1
Rotary axis vector V1	Rotary axis vector V1	Rotary axis vector V1
Offset vector I2	Offset vector I3	Offset vector I2
Rotary axis vector V2	Rotary axis vector V2	Offset vector I3
Offset vector I3	Offset vector I4	Rotary axis vector V2
		Offset vector I4

References

You can find additional information in:

- Function Manual Basic Functions; Tool Offset (W1)
- Function Manual Special Functions; Multiple Transformations (F2)

Name of the swivel data set

Kinematic channel1		Name of swivel data record	
Name	TC1	Kinematic	Swivel table
Enable	Yes	Init. setting	No. 1
Retract	Z	Working plane	
Retract position	X	Y	Z
Offset vector I2	-300.030000	-200.040000	-500.000000 [mm]
Rotary axis vector U1	-1.000000	0.000000	0.000000
Offset vector I3	120.030000	0.040000	0.000000 [mm]
Rotary axis vector U2	0.000000	0.000000	-1.000000
Offset vector I4	100.000000	200.000000	500.000000 [mm]
Swivel mode	Axis by axis		
Rotary axes direct	No	Track tool	No
Projection angle	No	B axis kinematics	No
Solid angle	No		
Direction refer.	Rotary axis 1, + direction selected		
JobShop functions	Automatic swivel data record change		

Figure 21-4 Dialog to input parameters for the swivel data set

If several swivel data sets are declared in each NC channel, then a name is assigned to each swivel data set. \$TC_CARR34[n]

No name needs to be specified if the swivel-mounted tool carrier is not exchangeable (i.e. one swivel data set per channel).

The name of the swivel data set may only contain characters that are permissible for NC programming: A...Z, 0...9 and _ !

If several NC channels have been set up, note the settings in MD28085, Chapter "Technology cycles for swiveling (Page 493)". If a TO unit is assigned to several NC channels, the channel display in the dialog corresponds to the TO unit.

Offset vectors I1 to I4 \$TC_CARR1[n] ... \$TC_CARR20[n]

The vectors always contain three components which represent the reference to the machine axes (X, Y, Z). The positions in the kinematic chain are measured by the machine manufacturer; they are always relevant with respect to a swivel head / swivel table (swivel data set).

Offset vectors I1 to I4 refer to the non-swiveled state of the rotary axes (basic setting: machine kinematics). However, be aware that the traversing range in the swivel planes may be

restricted. If machine kinematics are to be implemented with just one rotary axis, this must always be declared as the 1st rotary axis.

Manually adjustable rotary axes (manual mode) are possible with and without measuring systems and can be used with "simple machines".

Rotary axes that are positioned by an additional mechanical system, can be declared in the "Semiautomatic" mode.

Example

In a boring mill, the swivel head is positioned by coupling the spindle. In this case, the rotary axes of the swivel head must be declared as "Semiautomatic". The appropriate machine-specific cycles are called in CUST_800.SPF.



Machine manufacturer

Please refer to the machine manufacturer's specifications.

Swivel head

- I3 distance from the tool adapter to the pivot point/intersection of the 2nd rotary axis
- I2 distance from the pivot point/intersection of the 2nd rotary axis to the pivot point/intersection of the 1st rotary axis
- I1 closure of the $I1 = -(I2 + I3)$ vector chain, if the swivel head cannot be changed

Swivel table

- I2 distance from the machine reference point to the pivot point/intersection of the 1st rotary axis
- I3 distance from the pivot point/intersection of the 1st rotary axis to the pivot point/intersection of the 2nd rotary axis (or to the reference point of the tool adapter)
- I4 closure of the $I4 = -(I2 + I3)$ vector chain, if the swivel table cannot be changed

Swivel head/swivel table (mixed kinematics)

- I2 distance from the tool adapter to the pivot point/intersection of the 1st rotary axis
- I1 closure of the $I1 = -I2$ vector chain, if the swivel head cannot be changed
- I3 distance from the machine reference point to the pivot point/intersection of the 2nd rotary axis (or to the reference point of the tool adapter)
- I4 closure of the $I4 = -I3$ vector chain, if the swivel table cannot be changed

The offset vectors do not have to point to the pivot point of the rotary axes. The important thing is that they point to a point on the direction of rotation (intersection).

The sign of the offset vectors (I1 to I4) and the rotary axis vectors (V1, V2) result from the specifications of the axis directions according to **ISO 841-2001** or **DIN 66217** (right-hand rule).

In the case of machine kinematics that move the workpiece (rotary table), the axis direction is reversed.

Note

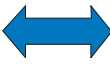
Interrelation of TOOLCARRIER $\leftrightarrow$ 5-axis transformation (transformer type 24, 40, 56):

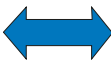
For 5-axis transformation, transformer type 72 in MD24100: \$MC_TRAFO_TYPE_1 can be used.

In the case of transformer type 72, the vectors of the TOOLCARRIER in MD24582: \$MC_TRAFO5_TCARR_NO_1 are used.

Swivel head (swivel-mounted tool)				
\$TC_CARR23[1]="T"			MD24100: \$MC_TRAFO_TYPE_1=24	
I1	\$TC_CARR1...3[n]		MD24500: \$MC_TRAFO5_PART_OFFSET_1 [0...2]	
I2	\$TC_CARR4...6[n]		MD24560: \$MC_TRAFO5_JOINT_OFFSET_1 [0...2]	
I3	\$TC_CARR15...17[n]		MD24550: \$MC_TRAFO5_BASE_TOOL_1 [0...2]	
Closure of the I1=-(I2+I3) vector chain; for fixed-mounted machine kinematics				

Swivel table (swivel-mounted workpiece)			
\$TC_CARR23[1]="P"			MD24100: \$MC_TRAFO_TYPE_1=40
I2	\$TC_CARR4...6[n]		MD24550: \$MC_TRAFO5_BASE_TOOL_1 [0..2]
I3	\$TC_CARR15...17[n]		MD24558: \$MC_TRAFO5_JOINT_OFFSET_1 [0...2]
I4	\$TC_CARR18...20[n]		MD24500: \$MC_TRAFO5_PART_OFFSET_1 [0...2]
Closure of the $I4=-(I2+I3)$ vector chain; for fixed-mounted machine kinematics			

Swivel head + swivel table (swivel-mounted tool + workpiece)			
\$TC_CARR23[1]="M"			MD24100: \$MC_TRAFO_TYPE_1=56
I1	\$TC_CARR1...3[n]		MD24560: \$MC_TRAFO5_JOINT_OFFSET_1 [0...2]
I2	\$TC_CARR4...6[n]		MD24550: \$MC_TRAFO5_BASE_TOOL_1 [0...2]
I3	\$TC_CARR15...17[n]		MD24558: \$MC_TRAFO5_JOINT_OFFSET_PART_1[0...2]
I4	\$TC_CARR18...20[n]		MD24500: \$MC_TRAFO5_PART_OFFSET_1 [0...2]
Closure of the I1=-I2 I4=-I3 vector chain; for fixed-mounted machine kinematics			

Rotary axis vectors V1, V2		
V1 \$TC_CARR7..9[n]		MD24570: \$MC_TRAFO5_AXIS1_1[0..2]
V2 \$TC_CARR10..12[n]		MD24572: \$MC_TRAFO5_AXIS2_1[0..2]

Fine offsets of offset vectors

\$TC_CARR41[n]	to	\$TC_CARR60[n]

Assignment of base vectors to the fine offset vectors:

I1 \$TC_CARR1..3[n]	to	\$TC_CARR41..43[n]
I2 \$TC_CARR4..6[n]	to	\$TC_CARR44..46[n]
I3 \$TC_CARR15..17[n]	to	\$TC_CARR55..57[n]
I4 \$TC_CARR18..20[n]	to	\$TC_CARR58..60[n]

The fine offsets are activated by the following setting data:

SD42974 \$SC_TOCARR_FINE_CORRECTION = 1	Fine offset TCARR on/off
= 0	When activating a tool carrier that can be oriented, the fine offset values are not taken into account.
= 1	When activating a tool carrier that can be orientated, the fine offset values are taken into account.

The fine offsets act in addition to the corresponding base vectors when the Swivel function CYCLE800 or the NC function TCARR=n is called.

Name of rotary axes

Rotary axis channel1

Name: WSP Kinematics Inclinable head No. 1

Rot. axis 1 Identifier **A1** Mode automatically

Angle area 0.000 - 0.000

OffsetKinematics 0.000

Hirth teeth No

Rot. axis 2 Identifier Mode automatically

Angle area 0.000 - 0.000

OffsetKinematics 0.000

Hirth teeth No

Kinematic

Figure 21-5 Dialog to input parameters for the rotary axis

The following identifiers should be preferably chosen for the names of the rotary axes:

- Rotary axis rotates around machine axis X → A
- Rotary axis rotates around machine axis X → B
- Rotary axis rotates around machine axis Z → C

For automatic rotary axes, the channel names of the corresponding NC rotary axes must be entered (see \$TC_CARR37[n] TENS and HUNDREDS position: Automatic mode). For manual (manually adjustable) and semiautomatic rotary axes, you can use any axis identifier (up to six letters or digits).

Manual and semiautomatic rotary axes are not defined in the NC. With manual rotary axes, the value for the angular range is entered in the dialog box. Semiautomatic rotary axes are moved mechanically on the machine, e.g. through a coupled spindle. The adaptations for this can be made in the CUST_800 manufacturer cycle. The value for the angular range must not be specified, the rotary axis positioning is performed automatically.



Machine manufacturer

Please refer to the machine manufacturer's specifications.

\$TC_CARR35[n]	Name of rotary axis 1
\$TC_CARR36[n]	Name of rotary axis 2

System variable \$TC_CARR37[n]

In order that a value can be displayed in the input/selection boxes of the dialog for swiveling, the following display versions can be set:

Meaning of the decimal positions

\$TC_CARR37[n]	Display variants of the input screens for CYCLE800	
Decimal place	Meaning	
ONES	Selects the swivel mode	
	0 =	Axis-by-axis
	1 =	Axis-by-axis + projection angle
	2 =	Axis-by-axis + projection angle + solid angle
	3 =	Axis-by-axis + direct
	4 =	Axis-by-axis + projection angle + direct
	5 =	Axis-by-axis + projection angle + solid angle + direct
TENS	Rotary axis 1	
	0 =	Automatic
	1 =	Manual
	2 =	Semiautomatic
HUNDREDS	Rotary axis 2	
	0 =	Automatic
	1 =	Manual
	2 =	Semiautomatic

\$TC_CARR37[n]	Display variants of the input screens for CYCLE800	
Decimal place	Meaning	
THOUSANDS	Selection field direction: Direction selection of the rotary axes	
	0 =	No display of the direction reference for kinematics that only have one solution. Direction selection (_DIR) Minus is generated in the cycle call CYCLE800.
	3 =	Direction reference, rotary axis 1, direction selection Minus in the basic setting of the kinematics.
	4 =	Direction reference, rotary axis 2, direction selection Minus in the basic setting of the kinematics.
	5 =	No display of the direction reference for kinematics that only have one solution. Direction selection (_DIR) Plus is generated in the cycle call CYCLE800.
	8 =	Direction reference, rotary axis 1, direction selection Plus in the basic setting of the kinematics.
	9 =	Direction reference, rotary axis 2, direction selection Plus in the basic setting of the kinematics.
		The values 1, 2, 6 and 7 are not permitted.
TEN THOUSANDS	Selection field, correction of the tool tip or B axis kinematics	
	0 =	No display of the correction of the tool tip input field.
	1 =	Correction of tool tip by means of TRAORI.
	2 =	No correction of tool tip + B axis kinematics turning technology.
	3 =	Correction of tool tip + B axis kinematics turning technology. The Correction of tool tip function requires the "5-axis transformation (TRAORI)" option.
HUNDRED THOUSAND	Basic position of the kinematics ¹⁾	
	0 =	Corresponding to the working plane G17, G18 or G19 (compatibility)
	1 =	Tool axis in the Z direction
	3 =	Tool axis in the X direction
ONE MILLION TEN MILLION	Selection field, retraction	
	00 =	No retraction
	01 =	Retraction Z
	02 =	Retraction Z, XY
	03 =	Retraction Z or Z, XY
	04 =	Maximum retraction in tool direction
	...	
	08 =	Incremental retraction in tool direction
	...	
	15 =	Retraction Z or Z, XY or in maximum tool direction or in incremental tool direction
	\$TC_CARR38[n]	Retraction position X
	\$TC_CARR39[n]	Retraction position Y
	\$TC_CARR40[n]	Retraction position Z

\$TC_CARR37[n]	Display variants of the input screens for CYCLE800	
Decimal place	Meaning	
HUNDRED MILLION	Swivel data set enabled Setting swivel data set change Automatic or manual tool change is only necessary under ShopMill or Shop-Turn (→ see also: CUST_800.spf, markers _M2 to _M13). A swivel data set must be "enabled" in every case (value ≥ 4).	
	0 =	Swivel data set not enabled
	4 =	Swivel data set enabled automatic swivel data set and tool change
	5 =	Swivel data set enabled automatic swivel data set change and manual tool change
	6 =	Swivel data set enabled manual swivel data set change and automatic tool change
	7 =	Swivel data set enabled manual swivel data set and tool change

Notes on the setting Basic position of the kinematics ¹⁾

For turning machines (1st technology is turning) with a B axis, you can set whether the tool is oriented toward "-Z" or toward "-X" in the basic position of the kinematics. This setting is taken into account for "Align tool" and for "Swivel plane".

Align tool

- For "Align tool", the programming of beta and gamma is machine-independent.
- With the basic position setting "Working plane" or "-Z", beta = 0 also corresponds to this basic position (compatibility).
- With the basic position setting "-X", the B axis is re-oriented through 90° for beta = 0.

Swivel plane (new swivel plane)

- With the basic position setting "Working plane" or "-Z" and active working plane G17, there is no new tool orientation for swivel plane to basic position (compatibility).
- With the basic position setting "-X" and active working plane G17, the tool orientation is rotated 90° around Y for swivel plane to basic position.

Therefore, with the basic position setting "-X", milling is possible in a swiveled plane in different working planes (G17 or G19).

The following settings and programming is not permitted and results in error messages:

- Swivel plane, basic position -Z and G18 or G19 active
- Swivel plane, basic position -X and G18 active

Retracting the geometry axes before swiveling

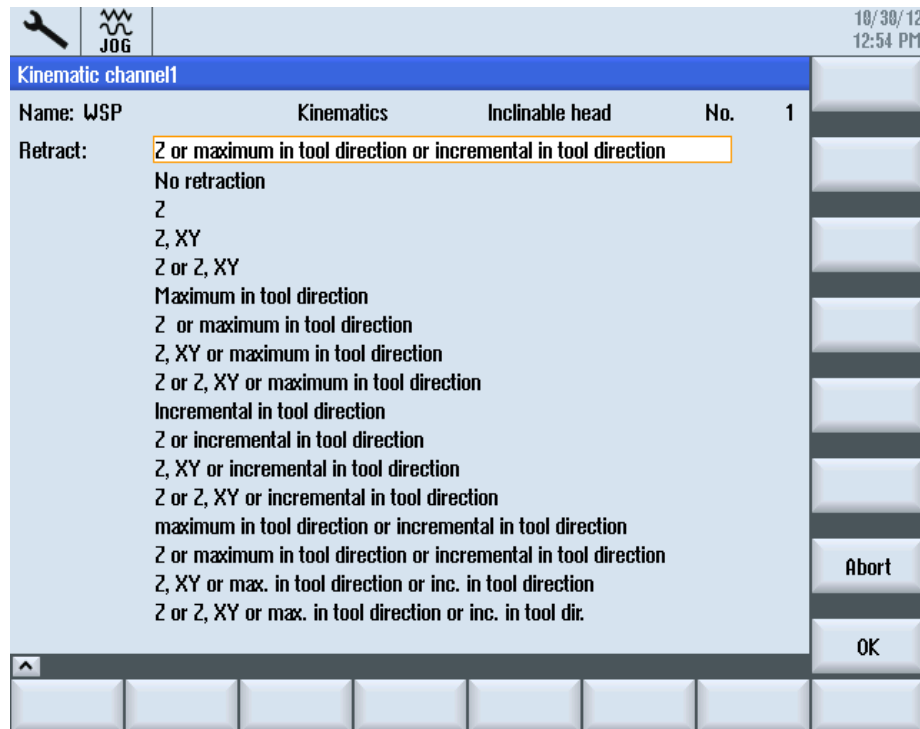


Figure 21-6 Dialog to select the type of retraction

Select the retraction variant in the input field: This is defined in the ONE MILLION and TEN MILLION position of the system variable `$TC_CARR37[n]`:

- Retract axis Z
- Retract axes Z, XY
- Retract in the tool direction, maximum or incremental

NOTICE**Collision avoidance**

Make sure that the tool and the workpiece cannot collide during swivelling when moving the tool axes.

Retracting axis Z or retracting axes Z, XY is realized as an absolute machine position at the values of parameters `$TC_CARR38[n]` to `$TC_CARR40[n]`.

<code>\$TC_CARR38[n]</code>	Retraction position X
<code>\$TC_CARR39[n]</code>	Retraction position Y
<code>\$TC_CARR40[n]</code>	Retraction position Z

The type of retraction is modified in the Manufacturer cycle CUST_800.SPF (Page 521).

Note

Retraction in the tool direction is useful for swivel data sets in which the tool is re-oriented (swivel head or mixed kinematics). This applies particularly for turning machines with a B axis. During retraction in "maximum tool direction", traversing is performed until one of the relevant linear axes reaches the software limit switch.

21.7.4 Examples of machine kinematics for the commissioning of the Swivel function

Example 1: Swivel head 1 "HEAD_1"

- Rotary axis 1(C) (manual) around Z
- Rotary axis 2(A) (manual) around X
- Manually adjustable changeable swivel head (manual)

Vectors relate to the kinematics basic setting (drawing not true-to-scale)

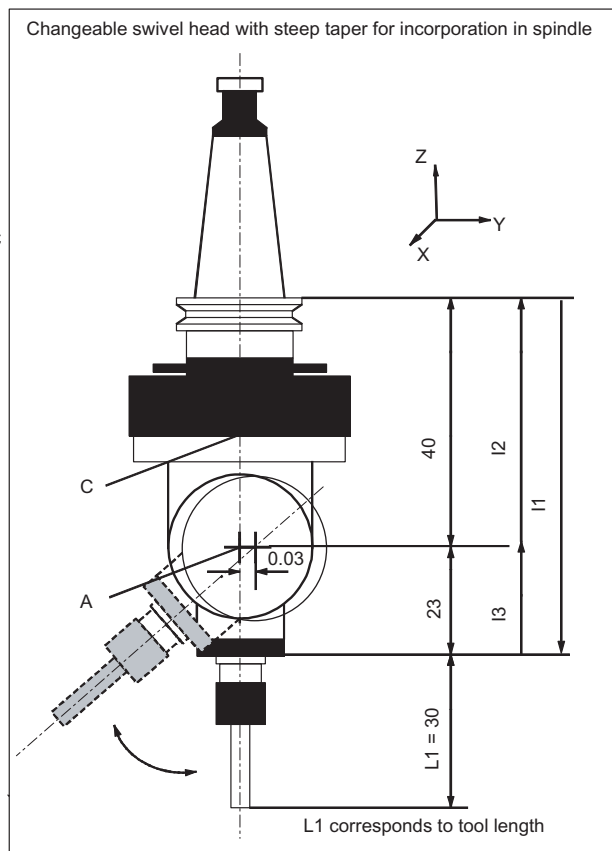


Table 21-1 Commissioning softkey "Swivel", Kinematics (Example 1)

Kinematics	Swivel head		HEAD_1
Retract	Z		
	X	Y	Z

Kinematics	Swivel head		HEAD_1
			200.000
Offset vector I1	0.000	0.030	-63.000
Rotary axis vector V1	0.000	0.000	1.000
Offset vector I2	0.000	0.000	40.000
Rotary axis vector V2	1.000	0.000	0.000
Offset vector I3	0.000	-0.030	23.000
Display version			
Swivel mode	Axis-by-axis		
Direction reference	Rotary axis 2		
Correct tool	No		
Rotary axes			
Rotary axis 1	C	Mode	Manual
Angular range	0.000		360.000
Rotary axis 2	A	Mode	Manual
Angular range	-15.000		100.000

Example 2: Swivel head 2 "HEAD_2"

Rotary axis vector V1: Rotary axis B rotates around Y
 Rotary axis vector V2: Rotary axis C rotates around Y **and** around Z
 Offset vector I1: Closure of vector chain with fixed-mounted swivel head
 $I1 = -(I2 + I3)$
 Offset vector I2: Distance between pivot point of **rotary axis 1** and pivot point of **rotary axis 2**
 Offset vector I3: Distance between reference point of tool and pivot point of **rotary axis 2**
 Cardanic swivel head (manually adjustable) with Hirth joint
Vectors relate to the basic setting of the kinematics
 If the swivel head is fixed-mounted, the vector chain is closed (see I1)

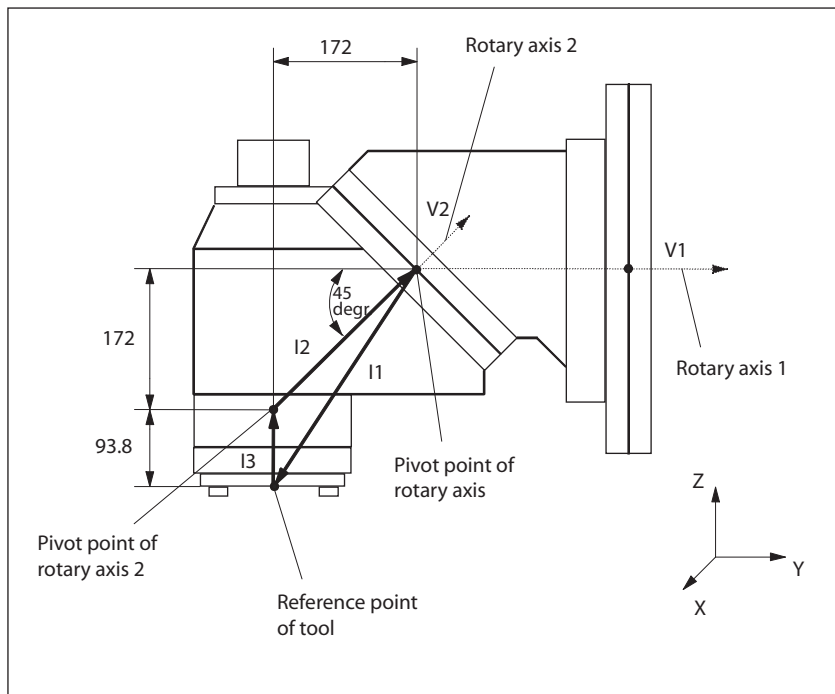


Table 21-2 Commissioning softkey "Swivel", Kinematics (Example 2)

Kinematics	Swivel head		HEAD_2
Retract	Z	Tool direction	Max.+inc.
	X	Y	Z
			200.000
Offset vector I1	0.000	-172.000	-265.800
Rotary axis vector V1	0.000	1.000	0.000
Offset vector I2	0.000	172.000	172.000
Rotary axis vector V2	0.000	1.000 ¹⁾	1.000 ¹⁾
Offset vector I3	0.000	0.000	93.800
Display version			
Swivel mode	Axis-by-axis		
Direction reference	Rotary axis 2		
Rotary axes			
Rotary axis 1	B	Mode	Manual
Angular range	0.000		360.000
Kinematics offset	0.000		
Hirth joint	Yes	Angular grid	1.000
Rotary axis 2	C	Mode	Manual
Angular range	0.000		180.000
Kinematics offset	0.000		

The reference point of the pivot point of rotary axes 1 and 2 can be offset on the line of rotation and does not have to coincide with the mechanical pivot point.

- 1) Calculation of rotary axis vector V2: 45 degree angle
 $V2Y = \sin(45) = 0.7071$
 $V2z = \cos(45) = 0.7071$
 $V2Y$ and $V2z$ can be normalized to 1.

Example 3: Cardanic table "TABLE_45"

Vectors relate to the basic setting of the kinematics

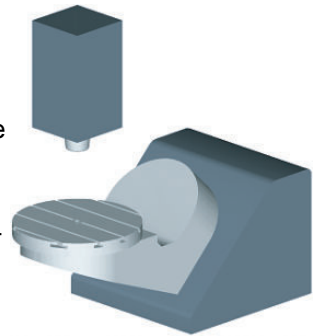
Rotary axis vector V1: Rotary axis B rotates around Y **and** around Z.

Rotary axis vector V2 Rotary axis C rotates around Z.

Offset vector I2: Distance from the reference point of the machine to the pivot point/intersection of **rotary axis 1**

Offset vector I3: Distance between pivot point/intersection of **rotary axis 1** and pivot point/intersection of **rotary axis 2**

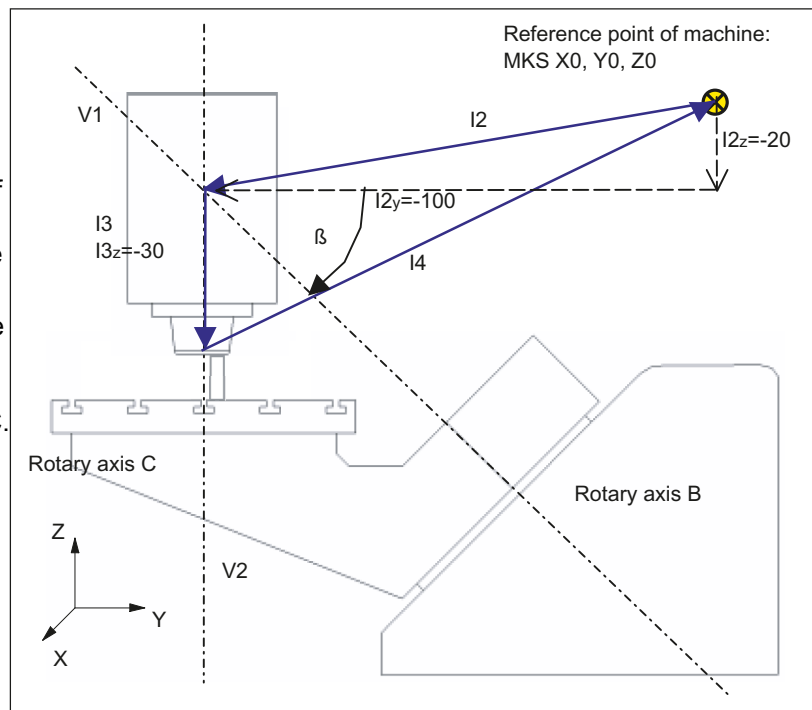
Offset vector I4: Closure of vector chain $I4 = -(I2 + I3)$



Side view of the machine

Spindle (tool adapter) is positioned on a block dimension above the top edge of the table (rotary axis C) or the center of the table.

A measuring rod in the spindle is used to determine the turning center of rotary axis C.



Kinematics	Swivel table		TABLE_45
	X	Y	Z
Offset vector I2	0.000	-100.000	-20.000
Rotary axis vector V1	0.000	-1.000 ¹⁾	1.000 ¹⁾
Offset vector I3	0.000	0.000	-30.000
Rotary axis vector V2	0.000	0.000	-1.000
Offset vector I4	0.000	100.000	50.000
Display version			
Swivel mode	Axis-by-axis		
Direction reference	Rotary axis 2		
Correct tool	No		
Rotary axes			
Rotary axis 1	B	Mode	Auto
Angular range	0.000		180.000
Rotary axis 2	C	Mode	Auto
Angular range	0.000		360.000

- 1) Calculation of rotary axis vector V1: $\beta = -45$ degrees
 $V1Y = \sin(-45) = -0.7071$
 $V1z = \cos(-45) = 0.7071$
 $V1Y$ and $V1z$ can be normalized to -1 and 1.

Example 4: Swivel head/rotary table "MIXED_45"

Vectors relate to the basic setting of the kinematics

Rotary axis vector V1: Rotary axis B rotates around Y **and** around Z.

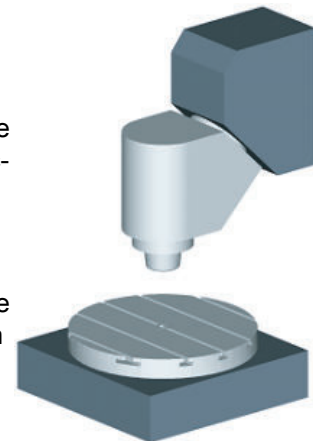
Rotary axis vector V2: Rotary axis C rotates around Z.

Offset vector I2: Distance from the reference point of the tool adapter to the pivot point/intersection of
rotary axis 1

Offset vector I1: Closure of vector chain I1=-I2

Offset vector I3: Distance from the reference point of the machine to the pivot point/intersection of
rotary axis 2

Offset vector I4: Closure of vector chain I4=-I3



The diagram illustrates a 3-axis machine tool with a 3D coordinate system (X, Y, Z) and a reference point (marked with a yellow circle). The tool is composed of several components, including a rotary axis B and a rotary axis C. The tool's position is defined by offsets I_1, I_2, I_3, I_4 and I_{2z}, I_{3y} . The angle β is shown between the tool's axis and the Y-axis. The reference point is labeled "Reference point of machine: MKS X0, Y0, Z0".

Spindle (tool adapter) is positioned on a block dimension above the top edge of the table (rotary axis C) or the center of the table. A measuring rod in the spindle is used to determine the turning center of rotary axis C.

Table 21-3 Commissioning softkey "Swivel", Kinematics (Example 4)

Kinematics	Mixed kinematics		MIXED_45
	X	Y	Z
Offset vector I1	0.000	0.000	-30.600
Rotary axis vector V1	0.000	1.000 ¹⁾	1.000 ¹⁾
Offset vector I2	0.000	0.000	30.600
Offset vector I3	300.000	150.000	0.000
Rotary axis vector V2	0.000	0.000	-1.000
Offset vector I4	-300.000	-150.000	0.000
Display version			
Swivel mode	Axis-by-axis		
Direction	Rotary axis 1		
Correct tool	Yes		
Rotary axes			
Rotary axis 1	B	Mode	Auto
Angular range	0.000		180.000

Kinematics	Mixed kinematics		MIXED_45
Rotary axis 2	C	Mode	Auto
Angular range	0.000		360.000

- 1) Calculation of rotary axis vector V1: $\beta = 45$ degrees
 $V1Y = \sin(45) = -0.7071$
 $V1Z = \cos(45) = 0.7071$
 $V1Y$ and $V1Z$ can be normalized to 1.

Example 5: Swivel table "TABLE_5"

Vectors relate to the basic setting of the kinematics

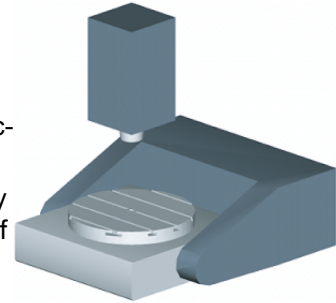
Rotary axis vector V1: Rotary axis A rotates around X.

Rotary axis vector V2: Rotary axis C rotates around Z.

Offset vector I2: Distance from the reference point of the machine to the pivot point/intersection of **rotary axis 1**

Offset vector I3: Distance from the pivot point of rotary axis 1 to the pivot point/intersection of **rotary axis 2**

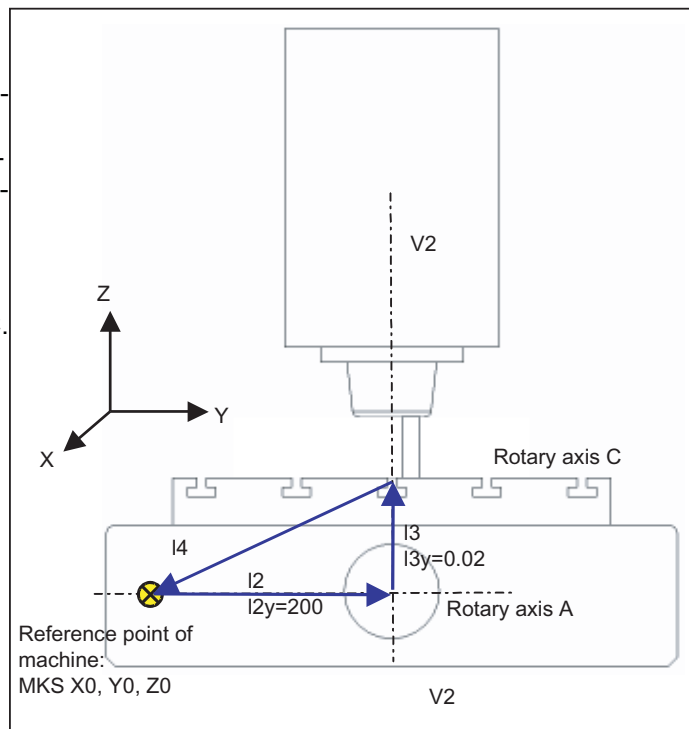
Offset vector I4: Closure of vector chain $I4 = -(I2 + I3)$



Side view of the machine from the X direction

Spindle (tool adapter) is positioned on a block dimension above the top edge of the table (rotary axis C) or the center of the table.

A measuring rod in the spindle is used to determine the turning center of rotary axis C.



Side view of the machine from the Y direction

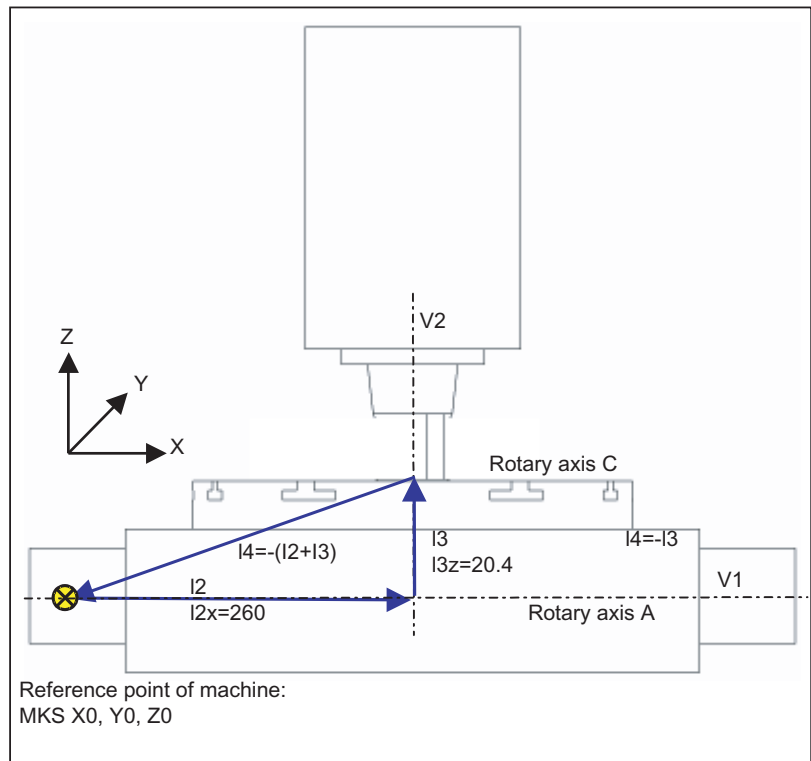


Table 21-4 Commissioning softkey "Swivel", Kinematics (Example 5)

Kinematics	Swivel table		TABLE_5
	X	Y	Z
Offset vector I2	260.000	200.000	0.000
Rotary axis vector V1	-1.000	0.000	0.000
Offset vector I3	0.000	0.020	20.400
Rotary axis vector V2	0.000	0.000	-1.000
Offset vector I4	-260.000	-200.020	-20.400
Display version			
Swivel mode	Axis-by-axis		
Direction	Rotary axis 1		
Correct tool	No		
Rotary axes			
Rotary axis 1	A	Mode	Auto
Angular range	-90.000		90.000
Rotary axis 2	C	Mode	Auto
Angular range	0.000		360.000

Example 6: Setting-up a lathe with B axis

For lathes with an additional rotary axis B, the tool can be aligned or swiveled on the X/Z plane. If your machine has a counterspindle, you can, for example, machine with a tool alternating between the main spindle and counterspindle.

You require the following swivel data sets to use the "Align tool" and "Swivel plane" functions:

- Swivel data set 1 for aligning and swiveling tools on the main spindle and counterspindle (turning, face Y/C, surface Y/C).
- Swivel data set 2 for machining inclined surfaces when milling on the main spindle (face B).
- Swivel data set 3 for machining inclined surfaces when milling on the counterspindle (face B).

When commissioning the "B-axis kinematics turning technology" in parameter \$TC_CARR37[n], the TEN THOUSANDS position must be set to 2 or 3 in swivel data set 1. This is the reason why in the program editor "Swivel tool → align turning or milling tool" is listed for swivel data set 1.

See also Section "Commissioning swivel data (kinematics chain) (Page 502)"

Example 7: Configuring a milling machine with B axis for turning.

You can configure a milling machine with a B axis (rotation around Y) so that turning is possible.

The following example refers to a milling machine with a rotary axis around Y (B), a rotary axis around Z (C) and a main spindle (SP) in the tool direction Z (G17).

You require the following swivel data sets to use the "Align tool" and "Swivel plane" functions:

- Swivel data set 1 for "Swiveling in the plane" when milling.
Mixed kinematics: Rotary axis 1: B-axis, rotary axis 2: C axis
- Swivel data set 2 for the "Swivel tool - align tool" when turning
Swivel head: Rotary axis 1: B-axis, rotary axis 2: Spindle SP

When milling, the SP spindle is the master spindle. Rotary axis C is declared as the master spindle in rotational operation, e.g. SETMS(2).

When commissioning the "B-axis kinematics turning technology" in parameter \$TC_CARR37[n], the TEN THOUSANDS position must be set to 2 or 3 in swivel data set 2. This is the reason why in the program editor "Swivel tool → align turning or milling tool" is listed for swivel data set 2.

See also Section "Commissioning swivel data (kinematics chain) (Page 502)"

References

Description of Functions, Basic Functions; Tool Offset (W1), Parameter CUTMOD:

- The CUTMOD NC function is activated when data set 2 is called in the NC program (CYCLE800 align turning tool). This means that the cutting edge position or the tool reference point is updated corresponding to the tool orientation:
- When calling a swivel data set for rotational operation in the NC program (CYCLE800 align tool), the CUTMOD NC function is activated. The cutting edge position, tool angle, cut direction and tool reference point are thus calculated in accordance with the current tool orientation following the "Align tool" function.

21.7.5 Manufacturer cycle CUST_800.SPF

Function

During swiveling, all axis positions are approached using the CUST_800.SPF cycle. The call is exclusively made from the swivel cycle CYCLE800 or from the cycles E_TCARR (ShopMill) or F_TCARR (ShopTurn).

In cycle CUST_800.SPF, the function markers (_M2: to _M59) are prepared and documented. Also refer to the following "Structogram CYCLE800".

There are extensions in the CUST_800.SPF cycle for the "Turning on milling machine" function, see Chapter "Adaptations for CUST_800 (Page 539)".

Note

If you modify the CUST_800, ensure that the G commands of the G groups used are restored at the end of the CUST_800 (versions S_G_1 to S_G_5!).

Parameter

CUST_800 (INT _MODE, INT _TC1, REAL _A1, REAL _A2, INT _TC2, REAL _T_POS)

_MODE	A jump is made to markers _M2 to _M59
_TC1	Number of the swivel head/table
_A1	Angle of rotary axis 1
_A2	Angle of rotary axis 2
_TC2	Feed evaluation in percent (%) for swiveling in JOG mode and number of the new swivel head/table when replaced in ShopMill
_T_POS	Incremental position during retraction in the incremental tool direction (see marker _M44, _M45)

Retract prior to swiveling

If the CUST_800.SPF cycle is not modified, the Z axis (marker _M41) or the Z axis followed by the X, Y axes (marker _M42) are first traversed in the Machine to the positions when retracting prior to swiveling. The freely available position values are specified in the system variables \$TC_CARR38[n] to \$TC_CARR40[n]. When retracting, the active tool cutting edge is deselected (D0) and is reselected after retraction.

If retraction in the tool direction had been declared, the tool axis is retracted to the software end position (maximum in tool direction) or by an incremental distance away from the tool in the tool direction. The tool lengths are taken into account accordingly.

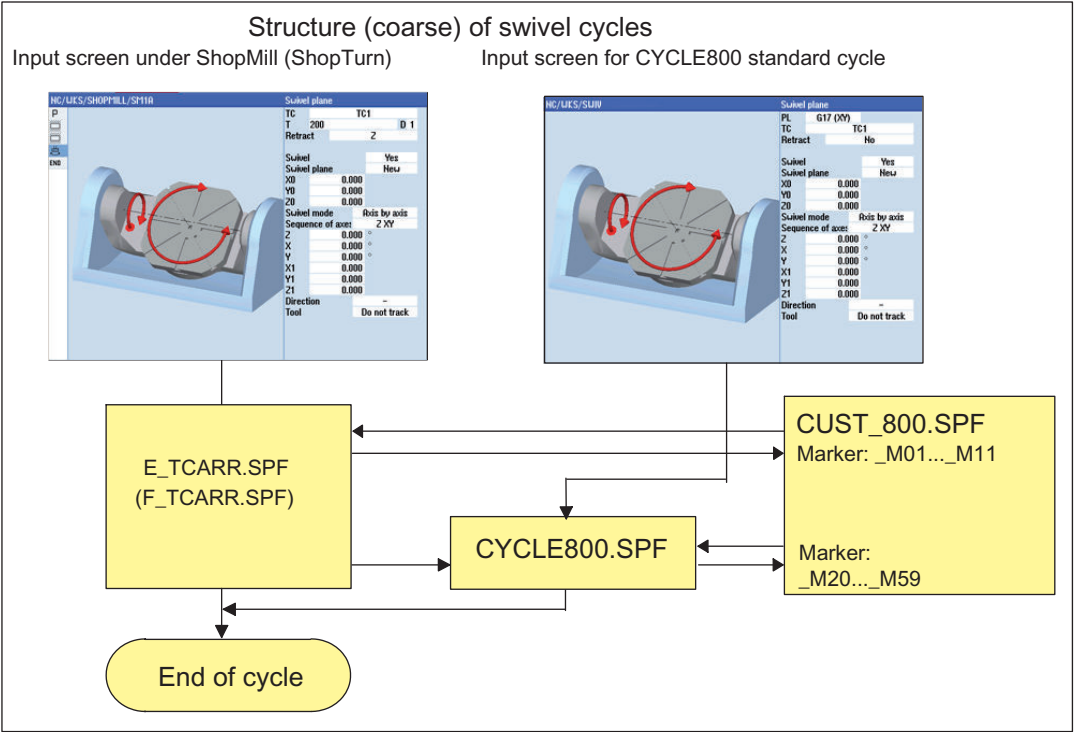


Figure 21-7 Structure (coarse) of swivel cycles

CYCLE800 structured chart

Sequence in AUTOMATIC mode

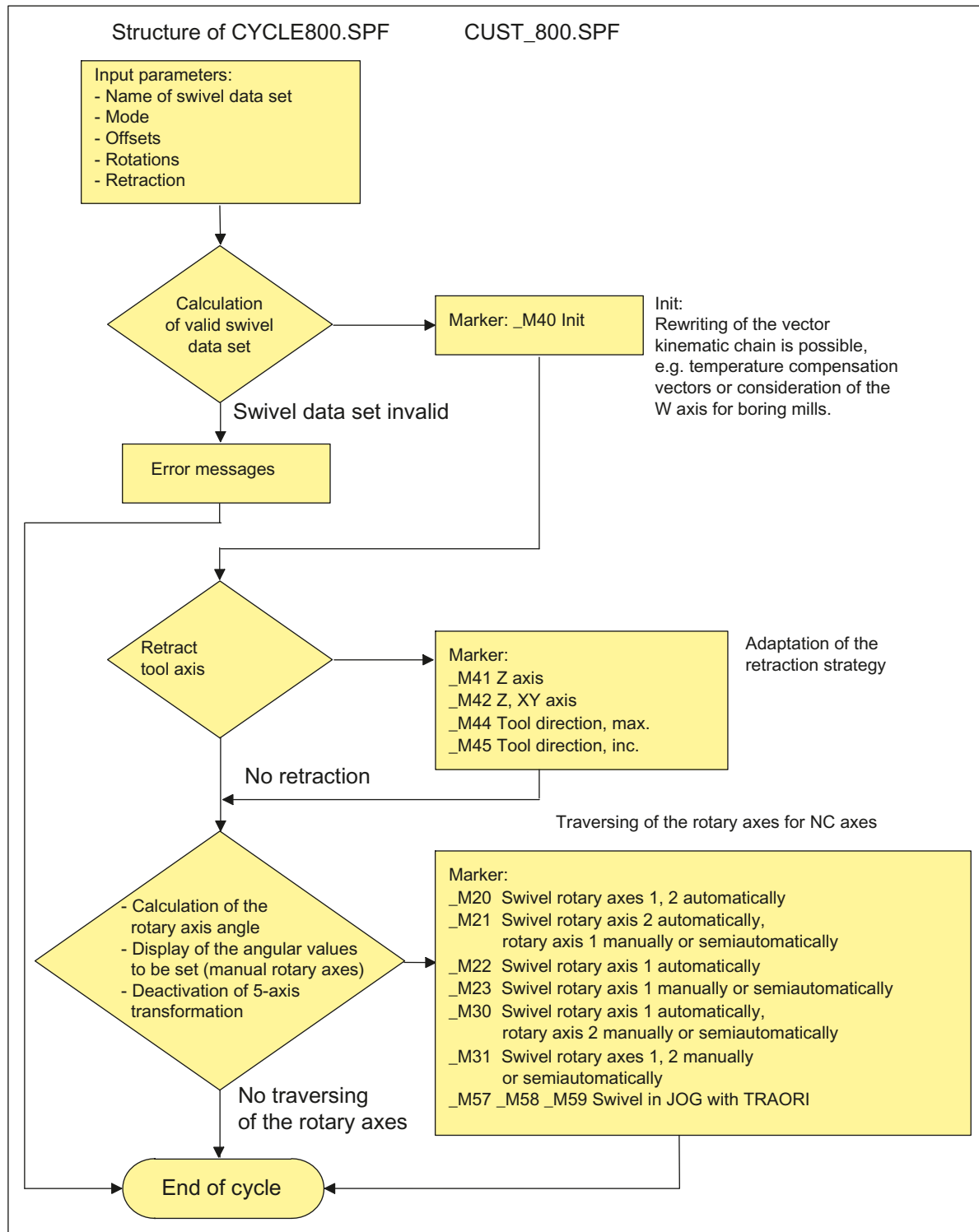


Figure 21-8 Structure: CYCLE800.SPF / CUST_800.SPF

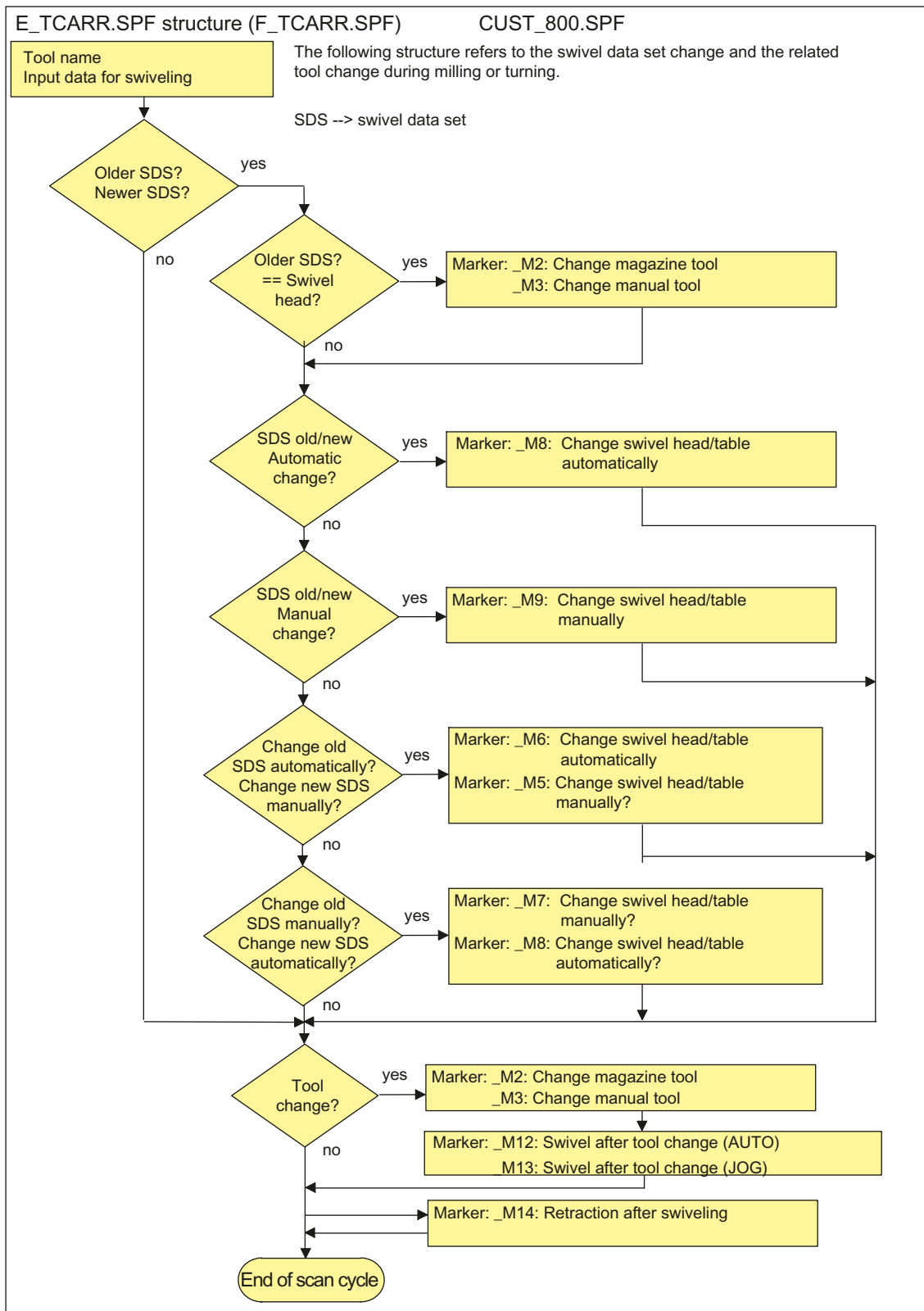


Figure 21-9 Structure: E_TCARR.SPF (F_TCARR.SPF) / CUST_800.SPF

Note on markers _M2 to _M13

When the swivel data set or tool is changed, the linear axes are recovered using the last retraction mode (modal).

If this behavior is not desired in milling/turning, the corresponding calls must be commented out with a semicolon (;). The E_SWIV_H or F_SWIV_H cycle is called in milling/turning (see markers _M2 to _M9) in the CUST_800.SPF manufacturer cycle.

Parameter E_SWIV_H (Par1, Par2, Par3)

- Par1: Number of swivel data set (_TC1)
- Par2: Angle of 1st rotary axis
- Par3: Angle of 2nd rotary axis

Modification examples

If the rotary axes (swivel head/table) are not to be positioned during swivel data change / tool change, the call of the E_SWIV_H cycle can be commented out at the relevant markers. If the rotary axes are to move to a certain position, an angle value can be transferred to parameters Par 2, Par 3.

Note on markers _M14, _M15

Depending on the values of the retraction plane and the programmed swivel plane, it is possible that the linear axes now also travel the swiveled retraction plane while running up from the current position to the software limit switches after a block search. To avoid this problem, marker _M14 in the CUST_800.SPF is called after swiveling. The E_SP_RP(30) cycle preset there runs up to the milling retraction plane, whereby travel may be along the software limit switches. An appropriate retraction after block search can be set at marker _M15.

Note on the marker _M16

If for a swivel data set change or when de-selecting the swivel data set, you wish to traverse the rotary axes to zero, the syntax of the _M16 marker can be modified accordingly, e.g. activate block N8006 E_SWIV_H(_TC1,0,0).

Note on markers _M20 to _M31

Markers _M20 to _M31 are distinguished by machine kinematics with two rotary axes or one rotary axis. A distinction is also made between automatic rotary axes (known to the NCU) and manual (semi-automatic) rotary axes. There is only ever one valid marker for swiveling with the active swivel data set.

Note on marker _M35

Run through _M35 for block search and a swivel data set with manual rotary axes.

Note on marker **_M40**

This marker is called before the calculation and activation of the Tool carrier function as user initialization. This makes it possible for special kinematics, to calculate or rewrite the current vectors of the active swivel data set.

Boring mill application example: For parallel axes, the vector in the Z direction can be updated with the actual value of the W axis.

A STOPRE is integrated for this marker.

Manufacturers can test as to whether this STOPRE for marker M40 in CUST_800.spf complies with their functional requirements. In this case, one must weigh up between velocity (without STOPRE) and functionality for interrupt conditions (RESET) for a swivel plane. This also depends on the type of machine kinematics.

Note on markers **_M41 to _M45**

The retraction strategy prior to swiveling the rotary axes can be modified at the markers _M41 to _M45.

The retraction to the MCS positions Z or Z, XY is made considering the tool length components without tool cutter deselection.

Note on marker **_M46**

Retraction before swiveling after a block search can be set at marker _M46. Variable _E_VER is 1 if it is a milling technology program.

Note on markers **_M57 to _M59**

Markers _M57 to _M59 are used for swiveling in JOG mode and active 5-axis transformation (TRAORI).

Note on "Correct tool"

"Correct tool" requires that a 5-axis transformation is set up which is equivalent to the corresponding swivel data set. The programming section for "Correct tool" is integrated in the markers _M20, _M21, _M22 and _M30. The first 5-axis transformation is called with TRAORI(1).

Note on tool change + swivel

In general, the swivel (CYCLE800) and tool change functions for a machine are independent of each other. Thus, the swiveled work plane can be retained in a technological sequence with multiple tools, e.g. centering, drilling, tapping.

If the rotary axes of the active swivel data set are involved in the mechanical sequence of the tool change or have to be retracted, this must be taken into account in the tool change program. After the tool change, the rotary axis positions are approached as prior to the tool change. If linear axes (geometry axes) are also involved in the tool change, the rotations in the NC (swivel frame) must not be deleted. Rather, the linear axes can be positioned as machine axes using the G153 or SUPA commands.

Note on swiveling without active tool compensation

If swiveling the rotary axes without active tool cutting edge (D0) is not possible, then you can adapt this in cycle CUST_800.SPF:

```
_M40 :  
IF ((NOT $P_TOOL) AND _TC1)  
  LOOP  
  MSG ("no tool cutting edge active")  
  M0  
  STOPRE  
  ENDLOOP  
ENDIF  
GOTOF _MEND
```

21.7.6 CYCLE996 measure workpiece kinematics

Function

With the Measure kinematics function, it is possible to calculate the geometric vectors used to define the 5-axis transformation (TRAORI and TCARR) by measuring the position of the ball in space.

The measurement is essentially carried out by means of workpiece probes, which scan three positions of a measuring ball on each rotary axis. The ball positions can be defined in accordance with user specifications so that they correspond to the geometric ratios on the machine. The only way of setting the ball positions is to reposition the rotary axis that is to be measured in each case.

Aside from the basic mechanics of the machine, no specific knowledge is required to use CYCLE996. No dimension drawings or machine location diagrams are necessary to carry out measuring.

Application range

The Measure kinematics function (CYCLE996) can be used to determine the data that is relevant to transformations in the case of kinematic transformations that involve rotary axes (TRAORI, TCARR).

Options:

- Redetermination of swivel data sets
 - Commissioning the machine
 - Use of swivel-mounted workholders as TCARR
- Checking swivel data sets
 - Service following collisions
 - Checking the kinematics during the machining process

Kinematics with manual axes (manually adjustable rotary tables, swivel-mounted workholders) can be measured in the same way as kinematics with NC-controlled rotary axes.

When CYCLE996 is started, a swivel data set with basic data (for kinematics type) must be parameterized. The measurement itself must be carried out without an active kinematic transformation.

If, in the setting data SD55740 \$MNS_MEA_FUNCTION_MASK bit 7 is set, then depending on the setting, the kinematics can be measured based on the calculated or on the rotary axis vectors saved in the swivel data set.

Requirements

The following requirements must be met in order to use CYCLE996 (Measure kinematics):

- SIEMENS measuring cycles package is installed
- Workpiece probe is calibrated
- Calibration ball is mounted
- Oriented tool carrier is set up (MD18088 MM_NUM_TOOL_CARRIER > 0)
- The basic geometry of the machine (X, Y, Z) is rectangular and referenced
The right angle refers to the workpiece spindle and should be preferably checked using a test mandrel.
- Defined position of the rotary axes involved in the transformation
- Defined traverse directions in compliance with the standard of all axes involved in the transformation according to ISO 841-2001 and/or DIN 66217 (righthand rule)
- Defined size of the protocol file (MD11420 \$MN_LEN_PROTOCOL_FILE ≥ 20)

The precise procedure when measuring and programming, including examples, is described in the following manual:

References

Programming Manual, Measuring Cycles: CYCLE996

21.8 Turning on milling machines

You can set up the turning technology as an extended technology on milling machines. The operating software has all the setting options of a milling machine and a turning machine. For example, the rotary table can be set at a high speed for the turning of the workpiece. The turning tool is then clamped in the milling spindle and can be used in various positions.



Machine manufacturer

Please refer to the machine manufacturer's specifications.

21.8.1 Setting up machine data and setting data

Setting up parameters

Machine data

Set the following machine data to set up the technology:

MD10602 \$MN_FRAME_GEOAX_CHANGE_MODE	Frames when switching over geometry axes
= 1	The current complete frame is calculated again when the geometry axes are switched over, and the translations, scalings and mirrorings of the new geometry axes are effective. The rotations of the old geometry axes are retained.

MD18114 \$MN_MM_ENABLE_TOOL_ORIENT	Assign orientation to cutting edges
= 0	The tool orientation function is not active.
	- OR -
= 3	Each tool cutting edge D = m of tool T = n is assigned to the following system parameters: \$TC_DPV[n, m] and \$TC_DPV3 - \$TC_DPV5 \$TC_DPVN3[n, m], \$TC_DPVN4[n, m] and \$TC_DPVN5[n, m]

Note

If you use angular tools (type 130) with variable tool orientation, you require the setting MD18114 = 3.

MD51226 \$MNS_FUNCTION_MASK_SIM	Simulation function screen
Bit 5 = 1	Turning tool in tool spindle rotatable (for simulation)

MD51781 \$MNS_MEA_T_PROBE_THICKNESS	Tool probe thickness for the automatic approach of turning tools to the probe on milling/turning machines
= 0	Default value

MD20100 \$MC_DIAMETER_AX_DEF	Geometry axis with transverse axis function
=X	The machine data defines a geometry axis as transverse axis.

MD20110 \$MC_RESET_MODE_MASK	Definition of control initial setting after reset / end of part program
Bit 0 = 1	Default value for system

MD20112 \$MC_START_MODE_MASK	Definition of the basic control setting after part program start
Bit 7 = 0	The current setting for the active transformation is retained.

MD20124 \$MC_TOOL_MANAGEMENT_TOOL-HOLDER	Toolholder number
= 1	Variable

MD20125 \$MC_CUTMOD_ERR	Alarm display for CUTMOD errors
= CFFFH	Display alarms for CUTMOD errors

MD20150 \$MC_GCODE_RESET_VALUES[21]	Initial setting of the G groups
= 2	Initial setting of G group 22 on CUT2DF

MD20150 \$MC_GCODE_RESET_VALUES[55]	Initial setting of the G groups
= 3	Initial setting of G group 56 TOWWCS

MD20152 \$MC_GCODE_RESET_MODE[21]	Reset behavior of the G groups
= 0	Reset behavior of G group 22

MD20152 \$MC_GCODE_RESET_MODE[55]	Reset behavior of the G groups
= 3	Reset behavior of G group 56 TOWWCS

MD20360 \$MC_TOOL_PARAMETER_DEF_MASK	Definition of the tool parameters
Bit 17 = 0	With CUTMOD for turning and grinding tools, the cutting edge for the calculation of the compensation values is not configured in the machining plane.
Bit 18 = 1	With CUTMOD for turning and grinding tools, always use the active plane for turning and grinding tools.

MD24030 \$MC_FRAME_ACS_SET	SZS coordinate system setting
= 1	SZS results from the WCS transformed with the \$P_CYCFRAME

MD35040 \$MA_SPIND_ACTIVE_AFTER_RESET[]		Reset behavior of the spindle
[Tool spindle] = 2	Do not stop tool spindle during jog → spindle is listed in the T,S,M screen	
[Turning spindle] = 2	Do not stop turning spindle during jog → spindle is listed in the T,S,M screen	

MD52001 \$MCS_DISP_COORDINATE_SYSTEM_2		Definition of the displayed directions in the user interface for turning operations (variable)
= 0		

MD52200 \$MCS_TECHNOLOGY		Technology
= 2	Milling	

MD52201 \$MCS_TECHNOLOGY_EXTENSION		Extended technology
= 1	Turning	

MD52206 \$MCS_AXIS_USAGE[]		Meaning of the axes in the channel
[Tool spindle] = 1	Tool spindle	
[Turning spindle] = 3	Turning spindle	

MD52207 \$MCS_AXIS_USAGE_ATTRIB[turning spindle]		Attributes of the axes
Bit 8 = 1	Offer rotary axis for blank clamping	

MD52212 \$MCS_FUNCTION_MASK_TECH		Cross-technology function screen
Bit 10 = 1	Align the turning tool via kinematics transformation	

MD52214 \$MCS_FUNCTION_MASK_MILL		Milling function screen
Bit 1 = 1	Offer fixed table for blank clamping	

Note

Only set bit 1 in MD52214 when a fixed table is used in addition to a rotary table.

MD52780 \$MCS_J_MEA_T_PROBE_APPR_MODE		Approach mode for measuring a tool in JOG
= 0	Technology-dependent approaching of the tool probe (default value) Turning technology: Manual approach Milling technology: Automatic approach	

= 1	Technology-independent approaching of the tool probe Manual approaching of the tool probe
= 2	Technology-independent approaching of the tool probe Automatic approaching of the tool probe

MD52242 \$MCS_TURN_TOOL_FIXING	Tool adapter for turning tools
= 0	Turning tools in tool spindle
= 1	Turning tools fixed on the tool headstock
= 2	Turning tools in the tool spindle or fixed on the tool headstock

Note

Depending on the set mounting alignment in MD52242, different parameters are available in the "Align Turning Tool" dialog.

Setting data

The following setting data is recommended: Set the setting data to set up the technology:

NOTICE	
Risk of collision	
Value changes of SD42940, SD42942 and SD42950 during the program execution can cause collisions. Only set the setting data during the commissioning.	

SD42935 \$SC_WEAR_TRANSFORM	Transformations for tool components
= 1	Orientation for milling tools: Standard G group 56: Observe or set TOWWCS

SD42940 \$SC_TOOL_LENGTH_CONST	Change of tool length components at change of plane
= 17	L1 = Z, L2 = Y, L3 = X

SD42942 \$SC_TOOL_LENGTH_CONST_T	Change of tool length components for turning tools at change of plane
= 19	L1 = Z, L2 = Y, L3 = X

Note

SD42942 is only evaluated after setting SD42950.

SD42950 \$SC_TOOL_LENGTH_TYPE		Assignment of the tool length offset independent of tool type
= 3	The assignment of the tool length components is performed separately for the turning and grinding tools (tool types 400 to 599) on the one hand and all other tools (milling tools) on the other hand.	
SD42954 \$SC_TOOL_ORI_CONST_M		Change of tool orientation components for milling tools at change of plane
= 0	Orientation for milling tools: Standard	
SD42956 \$SC_TOOL_ORI_CONST_T		Change of tool orientation components for turning tools at change of plane
= 0	Orientation for turning tools: Standard	
SD42998 \$SC_CUTMOD_PLANE_TOL		Difference between tool tip plane and machining plane
= 5	Maximum permissible difference for gamma of 0° to 180°	
SD42999 \$SC_ORISOLH_INCLINE_TOL		Difference between tool tip plane and machining plane for ORISOLH
= 1	Maximum permissible inclination of the tool tip compared to the G18 plane for "Align tool directly"	

Additional settings

- Set the current zero point in the center point of the turning tool, e.g. via a settable zero offset.
- Change the tool before the "Align turning tool" cycle call.
- Program the tool in the part program for the simulation.

Note

No tool is active after the start of the simulation.

- After the "Align turning tool" cycle call, traverse the tool to Y = 0.
- Limit the maximum speed of the tool spindle, e.g. with the tool change cycle, in order to avoid too high a speed of the turning tool.
- The turning tools in the tool management are automatically set up by the system. You can make some adaptations via the sltmmillturnlist-config.xml file, e.g. for the OEM tool list.

21.8.2 Setting up a kinematics transformation

In order to be able to use the turning on milling machines technology, at least one kinematics transformation must be set up for turning.

Requirement

A kinematic chain is already available for the machine.

For further information on kinematic chaining, see Chapter "Kinematic elements (Page 247)".

Setting up parameters

Set the following machine data to set up the kinematics transformation technology:

MD18866 \$MN_MM_NUM_KIN_TRAFOS	Maximum number of NC transformation objects
≥ 1	This machine data specifies the maximum number of NC transformations that can be defined with kinematic chains (variable).

Setting up system variables

Use the following system variables to set up the kinematics transformation:

System variable	Meaning
\$NT_NAME[n]	Name of the transformation objects that appear on the user interface
\$NT_TRAFO_TYPE[n]	TRAORI_STAT (fixed type for turning on milling machines)
\$NT_T_CHAIN_LAST_ELEM[n]	Name of the chain element that describes the end of the chain in the direction of the tool
\$NT_P_CHAIN_LAST_ELEM[n]	Name of the chain element that describes the end of the chain in the direction of the workpiece
\$NT_GEO_AX_NAME[n,0]	Name of the chain element that describes geometry axis X
\$NT_GEO_AX_NAME[n,1]	Name of the chain element that describes geometry axis Y
\$NT_GEO_AX_NAME[n,2]	Name of the chain element that describes geometry axis Z
\$NT_ROT_AX_NAME[n,0]	Name of the chain element that describes the rotary table
\$NT_ROT_AX_NAME[n,1]	Name of the chain element that describes the swivel axis
\$NT_ROT_AX_NAME[n,2]	Name of the chain element that describes the tool spindle
\$NT_IDENT[n,0]	Settings to adapt the screen form
\$NT_AUX_POS[n,0]	Value for the retraction in the direction of machine axis X ¹⁾
\$NT_AUX_POS[n,1]	Value for the retraction in the direction of machine axis Y ¹⁾
\$NT_AUX_POS[n,2]	Value for the retraction in the direction of machine axis Z ¹⁾
\$NT_CNTRL[n]	Settings to adapt to the machine

n = number of the transformation data set

¹⁾ See manufacturer cycle CUST_800.SPF markers _M47 and _M48

Meaning of the decimal places of \$NT_IDENT[n,0]

You can set the following display variants of the input screens for turning on milling machines:

Decimal place	Meaning
ONES	Selection of the alignment
	0 = Beta/gamma
	1 = Beta/gamma + direct alignment (B/SP)
THOUSANDS	Selection box: +/- direction reference of the swivel axis
	0 = No selection on the user interface, always minus
	4 = +/- selection via user interface, default: Minus
	5 = No selection on the user interface, always plus
	9 = +/- selection via user interface, default: Plus
TEN THOUSANDS	Selection box: Correct tool tip when swiveling
	2 = Do not list toggle
	3 = List toggle
ONE MILLION	Selection box: Retraction mode 0X and
TEN MILLION	Selection box: Retraction mode X0 (the permissible retraction modes result from the sum of the individual values: 0 ... 15)
	0 = No retraction
	1 = Retraction Z ²⁾
	2 = Retraction Z, XY ²⁾
	4 = Maximum retraction in tool direction
	8 = Incremental retraction in tool direction
HUNDRED MILLION	Enable data set
	4 = Kinematics active (fixed value)

²⁾ See system variables \$NT_AUX_POS[n,0] to \$NT_AUX_POS[n,2]

Meaning of the bits of \$NT_CNTRL[n]

You can make the following settings for the machine adaptation:

Bit 0 = 1	Fixed sequence of axes For easier handling of the data, the sequence of the axes is fixed: 1. Rotary table 2. Swivel axis 3. Tool spindle
Bit 1 = 1	Rotary table is spindle
Bit 2 = 0	Swivel axis is not a spindle
Bit 3 = 0	Tool spindle is not a spindle (during turning)
Bit 4 = 0	1st rotary axis has no Hirth joint (rotary table) ³⁾
Bit 5 = 0	2nd rotary axis has no Hirth joint (swivel axis) ³⁾
Bit 6 = 0	3rd rotary axis has no Hirth joint (tool spindle) ³⁾

³⁾ It is not permissible that either the 2nd or the 3rd rotary axis has a Hirth joint.

21.8.3 Behavior at reset and power on

Behavior at reset

Make the following settings so that the kinematics transformation is retained after reset and the turning tool can be used in JOG mode:

MD20110 \$MC_RESET_MODE_MASK		Definition of the basic control setting after reset / part program end
Bit 7 = 1	The current setting for active transformation is retained after reset / part program end.	
Bit 16 = 1	The current setting of the master spindle is retained.	

MD20112 \$MC_START_MODE_MASK		Definition of the basic control setting after part program start
Bit 7 = 0	Start behavior, "active kinematics transformation"	

MD20152 \$MC_GCODE_RESET_MODE[]		Reset behavior of G groups
[5] = 1	The current setting of the 6th G group is also retained after reset / part program end (G17, ...).	
[14] = 1	The current setting of the 15th G group is also retained after reset / part program end (G94, ...).	
[28] = 1	The current setting of the 29th G group is also retained after reset / part program end (DIAMxxx).	

Behavior at power on

In order to retain the kinematics transformation after a power on, you must also make the following additional settings:

MD20144 \$MC_TRAFO_MODE_MASK		Selection of the kinematics transformation function
Bit 1 = 1	Reselection of the transformation after power-up	

MD24008 \$MC_CHSFRAME_POWERON_MASK		Reset system frames after power on
Bit 4 = 0	Do not delete workpiece reference points after power on	

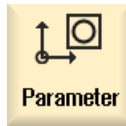
Note

The CUTMODK function cannot be retained after power on.

21.8.4 Setting the tool clamping angle

If the clamped turning tool is not in the G18 plane at the spindle zero position, the clamping angle must be set.

Procedure



1. Select the "Parameter" operating area.



2. Press the "Tool list" softkey.
The "Tool List" window opens.



3. Select a suitable turning tool in the list.
4. Press the "Additional data" softkey.
The "Additional Data - ..." window opens.
The "Additional data" softkey is only active if a turning tool for which the "Additional Data" window is configured is selected.
5. Enter the required angle in the "Angle" input field.

Note

The clamping angle is stored in the \$TC_DPROT cutting edge parameter.

21.8.5 Setting up the Hirth joint

If the tool spindle or the swivel axis has a Hirth joint with, for example, incrementing of 2.5° (360°/2.5° = 144 division), you can set up the following:

Settings

MD30505 \$MA_HIRTH_IS_ACTIVE	Axis is indexing axis with Hirth joint
= 0	Tool spindle / swivel axis is not a real Hirth axis

MD30330 \$MA_MODULO_RANGE	Size of the modulo range
= 360°	Effective for modulo axis

MD30340 \$MA_MODULO_RANGE_START	Start position of the modulo range.
= 0°	

MD30501 \$MA_INDEX_AX_NUMERATOR	Indexing axis, numerator for equidistant positions
= 360°	Effective for non-modulo axis

MD30502 \$MA_INDEX_AX_DENOMINATOR	Indexing axis, denominator for equidistant positions
= 144	Defines the value of the denominator for calculation of the distances between two indexing positions for equidistant positions. For modulo axes, this is therefore the number of indexing positions.

Swivel axis has Hirth joint

\$NT_CNTRL[n]	
Bit 5 = 1	Swivel axis has a Hirth joint [n] transformation name

Tool spindle has Hirth joint

\$NT_CNTRL[n]	
Bit 6 = 1	Tool spindle has a Hirth joint [n] transformation name

21.8.6 Different coordinate systems for milling and turning

In boring mills, the rotary table usually rotates around the Y axis (B axis). During turning, the workpiece must rotate around the Z axis (C axis) in the G18 plane. Therefore, different coordinate systems must be used on this machine for milling and turning.

Settings

Set the appropriate coordinate system via the following channel-specific machine data (values, for example, for boring mills):

MD52000 \$MCS_DISP_COORDINATE_SYSTEM	Position of the coordinate system (variable)
= 16	Y up, Z to the right

MD52001 DISP_COORDINATE_SYSTEM_2	Coordinate system for turning on milling machines (variable)
= 0	Z up, X to the right

Describe the rotation of the coordinate system as follows at the _M70 marker in the CUST_800.SPF cycle:

```
_M70:           ; Rotation of the coordinate system for turning
N800703 $P_WPFRAME=$P_WPFRAME:CROT(Y,-90):CROT(X,-90)
```

21.8.7 Adaptations for CUST_800

Adapt these markers in the CUST_800.SPF cycle for turning on milling machines:

Function markers

```

_M40:                                ; User Init
N800400 SETMS(1)                     ; Sets tool spindle as main spindle
N800401 DIAMOF                       ; No transverse axis defined as diameter
                                      programming for milling
GOTOF _MEND

_M47:                                ; Retract axis Z to MCS fixed position
                                      before tool alignment
GOTOF _MEND

_M48:                                ; Retract axis Z and then the XY axes to
                                      MCS fixed position before tool alignment
GOTOF _MEND

_M70:                                ; Alignment tool: Prepare
IF($P_SMODE[1]==1)                  ; Spindle mode of tool spindle
N800700 SPOS[1]=0                    ; Tool spindle as rotary axis C
ENDIF
N800701 SETMS(2)                     ; Sets table as main spindle
N800702 DIAM90                      ; Transverse axis defined as diameter
                                      programming for turning
GOTO _MEND

_M71:                                ; Alignment tool: Swivel rotary axis
_AX1=$PC_TRAFO_ROT_CHAN_AX_IN[1]    ; Rotary axis B
_AX2=$PC_TRAFO_ROT_CHAN_AX_IN[2]    ; Tool spindle
IF(_TC_N_WZ==0)                     ; No tooltip tracking
N800710 TRAFOOF
ENDIF
SBLON
N800711 G0 G40 G60 G90 AX[_AX1]=_A1 AX[_AX2]=_A2
SBLOF
GOTOF _MEND

_M72:                                ; Alignment tool: End
GOTOF _MEND

_M75:                                ; Alignment tool: Deselection
N800750 G17                          ; Standard plane for milling

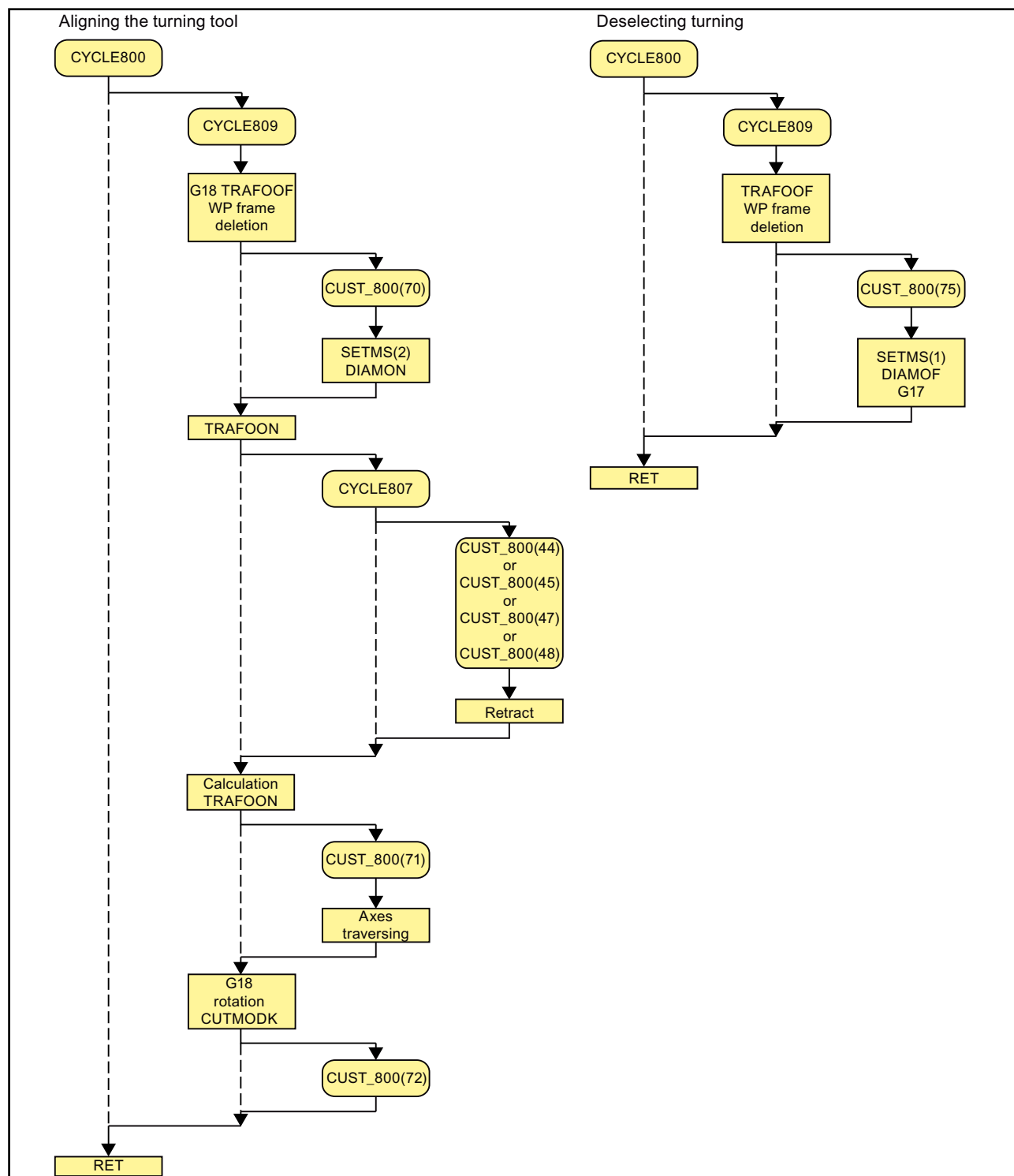
```

21.8 Turning on milling machines

N800751 SETMS (1)	; Sets tool spindle as main spindle
N800752 DIAMOF	; No transverse axis defined as diameter programming for milling
GOTOF _MEND	

Simplified flow diagrams

During tool retraction, the markers _M44 to _M48 are run through before the rotary axis positioning in accordance with the retraction variant.



21.8.8 Rotary axis positioning during block search

After a block search, you must position the rotary axis as follows (refer to swivel plane). Generally, there are two rotary axes (swivel axis and tool spindle):

Procedure

Read out the position value of the rotary axis via the following system variable:

System variable	Meaning
\$AC_RETPOINT[axis]	Repositioning point in the ASUB

You position the rotary axis in the CYCPE_MA.SPF user cycle.

Note

So that the old rotary axis position is not swiveled in, start the block search in the G code program after the block for the turning tool alignment.

For further information on CYCPE_MA.SPF, see Chapter "Extending the PROG_EVENT standard cycle (Page 452)".

If you have switched off the kinematics transformation, e.g. for a tool change, you can switch the transformation on again with the following program extension:

```
DEF STRING[32] U_TRAFO_NAME
U_TRAFO_NAME=$AC_TRAFO_NAME
TRAFOOF
...
TRAFOON(U_TRAFO_NAME)
```

21.9 High-speed machining

21.9.1 High speed settings: Configuring CYCLE832

Function

The High Speed Settings function (CYCLE832) is used to preset data for the machining of sculptured surfaces so that optimum machining is possible. The call of CYCLE832 contains three parameters:

- Tolerance
- Machining type (technology)
- Input of the orientation tolerance
The 3rd parameter is significant only for machines with multi-axis orientation transformation (TRAORI). By default, it is assigned the value 1.



Software option

You require the software option in order to use this function:
"Advanced Surface"

Tolerance / orientation tolerance

The tolerance value for the linear axes (geometry axes) is transferred to the CNC with the NC command CTOL. $CTOL = \text{root}(3) * \text{tolerance value}$.

Tolerance value = 0: Tolerance value (CTOL) is loaded from the following SDs:

SD55446 \$SCS_MILL_TOL_VALUE_ROUGH	Tolerance value for roughing CYCLE832
SD55447 \$SCS_MILL_TOL_VALUE_SEMIFIN	Tolerance value for semi-finishing CYCLE832
SD55448 \$SCS_MILL_TOL_VALUE_FINISH	Tolerance value for finishing CYCLE832

Example: CYCLE832(0,_ROUGH,1) → tolerance value from SD55446 is used.

If rotary axes are involved in the machining (multi-axis transformation), the tolerance value is transferred to the CNC with the NC command OTOL.

There are 2 options available for this:

1. Direct entry of the orientation tolerance OTOL
Machining types: _ORI_ROUGH, _ORI_SEMIFIN, _ORI_FINISH
To enter the orientation tolerance in the input screen for CYCLE832, bit 1 =1 must be set in the setting data SD55220 \$SCS_FUNCTION_MASK_MILL_TOL_SET
2. Tolerance is transferred with a factor on the NC command OTOL
Machining types: _ROUGH, _SEMIFIN, _FINISH
This factor can be set in the following channel-specific setting data for each machining type:

SD55441 \$SCS_MILL_TOL_FACTOR_ROUGH	Factor, tolerance of the rotary axes for roughing of the G group 59.
SD55442 \$SCS_MILL_TOL_FACTOR_SEMIFIN	Factor, tolerance of the rotary axes for roughing-finishing of the G group 59.
SD55443 \$SCS_MILL_TOL_FACTOR_FINISH	Factor, tolerance of the rotary axes for finishing of the G group 59.
= 10	

For machines for which the orientation tolerance is not significant, the cycle setting data SD55441 to 55443 can be set to a value of 0. No orientation tolerance is then calculated in CYCLE832 (OTOL= -1).

Machining type and technology G group 59

The machining types of the technology G group 59 are permanently assigned in CYCLE832 or in CUST_832.SPF:

Machining type	Plain text	Technology G group 59	Field index
Deselection	_OFF	DYNNORM	0
Roughing	_ROUGH	DYNROUGH	2
Rough-finishing	_SEMIFIN	DYNSEMIFIN	3
Finishing	_FINISH	DYNFINISH	4
Roughing ^{*)}	_ORI_ROUGH	DYNROUGH	2
Semi-finishing ^{*)}	_ORI_SEMIFIN	DYNSEMIFIN	3
Finishing ^{*)}	_ORI_FINISH	DYNFINISH	4

^{*)} With entry of an orientation tolerance

Dynamic parameters can be adapted to the respective machining operation with the technology G groups. Using the commands of technology G group 59, the value of the following channel and axis-specific machine data is activated using the corresponding **array index**:

MD20600 \$MC_MAX_PATH_JERK[0..4]	Path dependent maximum jerk.
MD20602 \$MC_CURV_EFFECT_ON_PATH_ACCEL[0..4]	Influence of path curvature on path dynamic response.
MD20603 \$MC_CURV_EFFECT_ON_PATH_JERK[0..4]	Influence of path curvature on path jerk.
MD32300 \$MA_MAX_AX_ACCEL[0..4]	Maximum axis acceleration.
MD32310 \$MA_MAX_ACCEL_OVL_FACTOR[0..4]	Overload factor for axial velocity jumps.
MD32431 \$MA_MAX_AX_JERK[0..4]	Maximum axial jerk for path motion.
MD32432 \$MA_PATH_TRANS_JERK_LIM[0..4]	Maximum axial jerk at the block transition in continuous-path mode.

Note

During the optimization of the machine axes, the values (note array index) of the specified machine data must be set correctly.

Machining type, deselection

When CYCLE832 is deselected, the G groups are programmed for the settings during the program run time; these settings are declared in machine data MD20150 \$MC_GCODE_RESET_VALUES[]. This involves the G groups 1, 4, 10, 12, 21, 24, 30, 45, 59, 61.

The contour and orientation tolerance is deactivated (CTOL=-1, OTOL=-1).

21.9.2 Manufacturer cycle CUST_832.SPF

Function

In contrast to the settings (G functions) by CYCLE832.SPF, these settings can be modified in the manufacturer cycle CUST_832.SPF.

The following markers are prepared in CUST_832.SPF:

<code>_M_NORM:</code>	For deselection <code>_OFF CYCLE832</code>
<code>_M_FINISH:</code>	For finishing <code>_FINISH</code> and <code>_ORI_FINISH</code>
<code>_M_SEMIFINISH:</code>	For semi-finishing <code>_SEMIFIN</code> and <code>_ORI_SEMIFIN</code>
<code>_M_ROUGH:</code>	For roughing <code>_ROUGH</code> and <code>_ORI_ROUGH</code>

The following G functions are programmed in CUST_832.SPF:

```
SOFT
COMPCAD
G645
FIFOCTRL
UPATH
;FFWON
```

Dependent on the machining type:

```
DYNNORM
DYNFINISH
DYNSEMIFIN
DYNROUGH
```

The programming of FGREF () is useful when machining with active multi-axis transformation (e.g. TRAORI). In this case, in CUST_832.SPF, variable `_FGREF` is pre-assigned a value of 10 mm. This value can also be modified. In CYCLE832.SPF, the value of variable `_FGREF` is written to the rotary axes involved in the machining, which are declared as orientation axis of a 5-axis transformation, using the FGREF(rotary axis) command. When G70/G700 is active, the value from `_FGREF` is converted into inches before writing to the command FGREF.

Take into account more than three rotary axes (orientation axes) in CYCLE832

In CYCLE832, a maximum of three rotary axes of the orientation transformation (TRAORI) are taken into account for FGREF. If more than three rotary axes are declared per channel for the orientation transformation, you can write the value to FGREF using the following syntax in the CUST_832:

```
FGREF[AA]=$AA_FGREF[C]
```

C = rotary axis 1 (axis is taken into account by CYCLE832)

AA = rotary axis 4

In the manufacturer cycle CUST_832.SPF, parameter `_MOTOL` is available for the evaluation of the "Multi-axis programming yes/no".

<code>_MOTOL=0</code>	No multi-axis programming
<code>_MOTOL=1</code>	Multi-axis programming where the orientation tolerance is entered

With the following setting data, additional G functions can be programmed in CUST_832.SPF for using multi-axis programming:

SD55220 \$SCS_FUNCTION_MASK_MILL_TOL_SET	Function screen form, high speed settings CY-CLE832
Bit 1	Orientation tolerance (OTOL) as a factor of the tolerance (CTOL) or directly entered

Example

Using _MOTOL in CUST_832.SPF for the marker for finishing:

```
_M_FINISH:
IF _MOTOL==1           ; if multi-axis programming
    ORISON
ELSE                   ; no multi-axis programming
    ORISOF
ENDIF
...
```

21.10 Measuring cycles and functions

21.10.1 Measuring cycles and measurement functions, general

Requirement

Two types of electronic probes are used for measuring:

- Probe to measure the workpiece
- Probe to measure the tool

The electronic probe must be connected to the NCU 7x0.

The electronic probe is called probe in the following document

References

NCU, Digital Inputs/Outputs Device Manual

Setting the switching behavior

You set the electrical polarity of the connected probe using the following general machine data:

MD13200 \$MN_MEAS_PROBE_LOW_ACTIVE		Polarity change of the probe
= 0	Probe in the non-deflected state, 0 V default value Probe in the deflected state, 24 V	
= 1	Probe in the non-deflected state, 24 V Probe in the deflected state, 0 V	

MD13210 \$MN_MEAS_TYPE		Measuring type for distributed drives
= 0	Default value	

Testing the probe function

You can test the switching function of the probe by manually deflecting it and checking the following PLC interface signals:

DB 10								
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB107							Probe actuated	
							Probe 2	Probe 1

To test the switching behavior and the measured value transfer, use an NC test program with, for example, the following NC commands:

MEAS	Measurement with deletion of distance-to-go
\$AC_MEA[n]	Check the switching operation, n = measurement input number
\$AA_MW[axis name]	Measured value of the axes in workpiece coordinates
\$AA_MM[axis name]	Measured value of the axes in machine coordinates

Example of a test program

Program code	Comment
%_N_PRUEF_MESSTASTER_MPF	
\$PATH=/_N_MPF_DIR	;Testing program probe connection
N00 DEF INT MTSIGNAL	;Bit memory to check the switching state
N05 G17 G54 T="3D_Taster" D1	;Select tool geometry for probe
N10 M06	;Activate tool
N15 G0 G90 X0 F150	;Starting position and meas. velocity
N20 MEAS=1 G1 X100	;Measurement at measuring input 1 in the X axis
N30 MTSIGNAL=\$AC_MEA[1]	;Switching operation at the 1st measuring input completed, ;YES / NO
N35 IF MTSIGNAL == 0 GOTOF _FEHL1	;Evaluation of the signal
N40 R1=\$AA_MM[X]	;Save measured value in machine coordinates at R1

Program code	Comment
N45 R2=\$AA_MW[X]	;Save measured value in workpiece coordinates at R2
N50 M0	;Check measured value in R1/R2
N55 M30	
N60 _FEHL1: MSG ("Probe does not switch!")	
N65 M0	
N70 M30	

General function settings for measuring cycles

MD51740 \$MNS_MEA_FUNCTION_MASK		Measuring cycles function screen
Bit 0	Activate calibration status monitoring – workpiece measurement for measuring in automatic	
Bit 1	Length reference for workpiece probe, milling technology	
= 0	Tool length L1, referred to the center of the probe sphere	
= 1	Tool length L1, referred to the sphere circumference of the probe sphere	
Bit 2	Support of tool carriers that can be orientated for workpiece measurement with correction in a tool	
Bit 4	Workpiece probe is permanently attached to the machine and cannot be positioned	
Bit 5	Correction in coarse ZO if no fine offset has been set up	
Bit 6	Measuring blocks (MEAS) are traversed in the measuring cycles with 100% feedrate override if the feedrate override is set > 0	
Bit 15	Workpiece measurement, traverse measuring block with path behavior G60 with exact stop at the end of the block	
Bit 16	Tool, turning technology, consideration of toolholders	

Note

After changing this machine data, the probe must be re-calibrated.

MD52740 \$MCS_MEA_FUNCTION_MASK		Measuring cycles function screen
Bit 0	Measuring input, workpiece probe	
= 0	Workpiece probe at the CNC measuring input 1, active - default	
= 1	Workpiece probe at the CNC measuring input 2, active	
Bit 1	Workpiece measurement, functional behavior of a third geometry axis (Y axis) when turning, based on the machining plane G18!	
Bit 2	Calculation of the correction angle with/without active orientation transformation	
Bit 16	Measuring input, tool probe	
= 0	Tool probe at the CNC measuring input 1, active	
= 1	Workpiece probe at the CNC measuring input 2, active - default	

MD54740 \$SNS_MEA_FUNCTION_MASK		Measuring cycles function screen
Bit 0	Repetitive measurement: workpiece measurement after exceeding a dimensional difference (parameter _TDIF) and or confidence range (parameter _TSA)	

Bit 1	Repetitive measurements: workpiece measurement with alarm and cycle stop at M0
Bit 2	Workpiece measurement: M0 for tolerance alarms 62304 oversize, 62305 undersize, 62306 permissible dimensional difference exceeded
Bit 3	Accept calibrated workpiece probe radius in the tool data.
Bit 4	For turning-milling technology, the calibration/measuring plane is different.
Bit 16	Repetitive measurement: tool measurement after a dimensional difference has been fallen below (parameter _TDIF) and or confidence range (parameter _TSA)
Bit 17	Repetitive measurement: tool measurement with alarm and cycle stop at M0
Bit 18	Workpiece measurement: M0, after the permissible dimensional difference has been exceeded
Bit 19	Tool measurement, milling tool: Spindle speed reduction at the last probing

MD55740 \$SCS_MEA_FUNCTION_MASK	Measuring cycles function screen
Bit 0	Collision monitoring using workpiece probe for intermediate positioning
Bit 1	Coupling the spindle alignment with coordinate rotation in the active plane, for workpiece measurement with multiprobe in the automatic mode
Bit 2	Direction of rotation of the spindle positioning for workpiece measurement regarding active coupling of the spindle alignment with coordinate rotation in the active plane
Bit 3	For workpiece measurement, number of repeat measurements if the probe does not switch
Bit 4	Measuring feedrate for workpiece measurement
Bit 5	Retraction velocity from the measuring point for workpiece measurement
Bit 6	Workpiece measurement: probe activation/deactivation when positioning the spindle
Bit 7	Measure kinematics, scaling basis for the orientation vectors
Bit 8	Measure kinematics, measurement with active TRAORI or TCARR (CYCLE 800)
Bit 9	User-specific scaling of the rotary axis vectors V1xyz and V2xyz
Bit 15	Workpiece measurement: calibration in the bore with known or unknown center point, measuring in JOG
Bit 16	Collision monitoring for tool measurement
Bit 17	Tool measurement: number of repetitive measurements if the probe does not switch
Bit 19	Retraction velocity from the measuring point for tool measurement
Bit 28	Continuous measuring standard log in JOG (attach)
Bit 29	Table format for measuring standard log in JOG

21.10.2 Manufacturer and user cycle CUST_MEACYC.SPF

Function

The CUST_MEACYC.SPF is part of the measuring cycle functionality. It is called in every measuring cycle before and after executing the measurement task. The CUST_MEACYC.SPF acts in the same way when measuring in the JOG mode and measuring in the AUTOMATIC mode.

You can use the CUST_MEACYC.SPF to program and execute sequences that are necessary before and/or after a measurement (e.g. activating/deactivating a probe).

Note

Copy the CUST_MEACYC cycle from the NC data/Cycles/Standard cycles directory to the NC data/Cycles/User cycles directory before making any changes. Changes for CUST_MEACYC only take effect in this directory.

CUST_MEACYC structogram

Sequence in the AUTOMATIC mode, using measure bore as example, CYCLE977 with probe type 712.

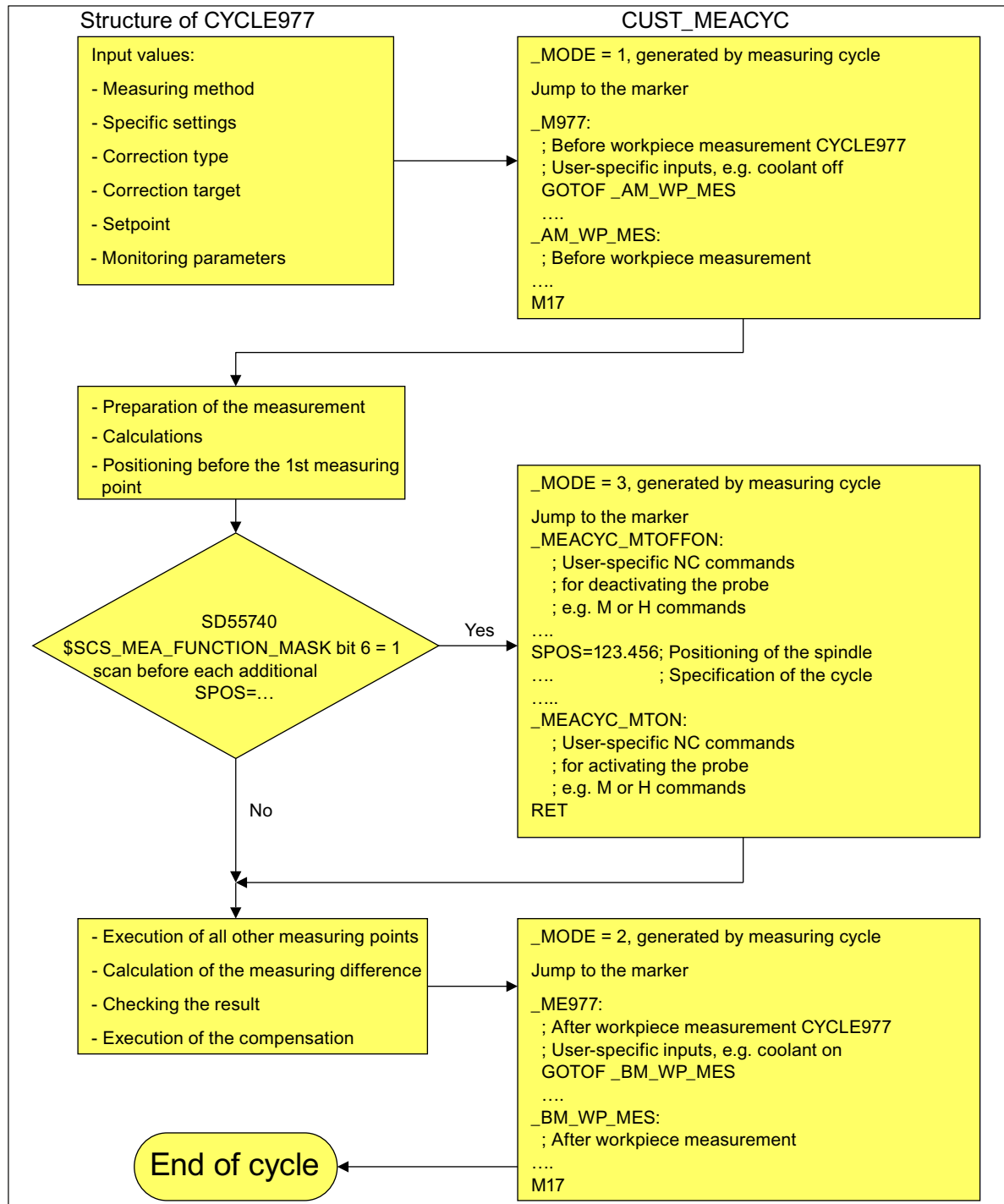


Figure 21-10 Structure: CYCLE977 / CUST_MEACYC

21.10.3 Measuring in JOG mode

Requirements

You have already made the settings from the previous Chapter "Measuring cycles and measurement functions, general (Page 546)".

Workpiece measurement

For milling:

- The probe has been inserted in the tool spindle.
- The probe has been selected in the tool list as type 7xx (probe).
- The probe is activated as tool in the current NC channel.

Tool measurement

To measure tools, an appropriate probe must be located in the machine space so that this can be reliably and safely reached with a tool in the spindle.

The following tool types are supported with measure tool:

- Milling technology: Tool types 1xx and 2xx
- Turning technology: Tool types 5xx, 1xx and 2xx

For the specified tool types, the tool lengths and the tool radii can be measured.

Setting general data

SD54782 \$SNS_J_MEA_FUNCTION_MASK_TOOL		Settings for the input screen, measuring in JOG, tool measurement.
Bit 2 = 1	Activates the "Enable automatic tool measuring" function.	
Bit 3 = 1	Selects the tool probe calibration data field, enable.	
Bit 10 = 1	Selects the individual cutting edge measurement	
Bit 11 = 1	Selects the spindle reversal when calibrating in the plane	

MD51784 \$MNS_J_MEA_T_PROBE_APPR_AX_DIR [i]	
Measuring in JOG: Approach direction in the plane to the tool probe	
-2	Negative direction, 2nd measuring axis
-1	Negative direction, 1st measuring axis (default setting)
0	Positive direction, 1st measuring axis
1	
2	Positive direction, 2nd measuring axis

MD52750 \$MCS_J_MEA_FIXPOINT	Z value for measurement with respect to a fixed point
MD52751 \$MCS_J_MEA_MAGN_GLAS_POS	Zoom-in position for tool measurement
[0]	Position in the 1st axis
[1]	Position in the 2nd axis

Checking and setting the general machine data

MD11450 \$MN_SEARCH_RUN_MODE	"Search run" programming.
Bit 1 = 1	Automatic ASUB start after output of the action blocks (see also MD11620 \$MN_PROG_EVENT_NAME). Alarm 10208 is not output until the ASUB is completed.

MD11604 \$MN_ASUP_START_PRIO_LEVEL	Priorities from which ASUP_START_MASK is effective.
= 1 - 64	Priorities for ASUP_START_MASK.

Setting channel-specific machine data

MD20050 \$MC_AXCONF_GEOAX_ASSIGN_TAB[]	Assignment, geometry axis to channel axis.
[0]	For measuring in JOG mode, all of the geometry axes must be available. Preferably XYZ.
[1]	
[2]	

MD20110 \$MC_RESET_MODE_MASK	Defines the basic control settings after reset / TP end,
= 4045H	Minimum value
Bit 0 = 1	Basic control setting after power on and reset.
Bit 2 = 1	
Bit 6 = 1	
Bit 14 = 1	

MD20112 \$MC_START_MODE_MASK	Definition of the basic control setting after part program start.
= 400H	Minimum value
Bit 6 = 0	Definition of the basic control setting after NC start.

MD20310 \$MC_TOOL_MANAGEMENT_MASK	Activates tool manager functions
= 4002H	Minimum value
Bit 1 = 1	Tool manager and monitoring functions active
Bit 14 = 1	Automatic tool change for RESET and Start.

MD28082 \$MC_SYSTEM_FRAME_MASK	System frames (SRAM).
Bit 0 = 0	Do not setup system frames for actual value setting and scratching.
Bit 0 = 1	Setup system frames for actual value setting and scratching.

The following channel-specific machine data is only effective if the system frame actual value setting and scratching is set up using the above mentioned machine data

MD28082 \$MC_SYSTEM_FRAME_MASK, bit 0 = 0.

MD24006 \$MC_CHSFRAME_RESET_MASK	Active system frames after reset.
Bit 0 = 1	Reset response of the system frames or actual value setting.

MD24007 \$MC_CHSFRAME_RESET_CLEAR_MASK	Clear system frames at reset.
Bit 0 = 0	Clear response of the system frames on reset.

Note

If you create the conditions described in this section and you have set and checked the machine setting data, then in JOG mode, you can measure a workpiece using a workpiece probe at a milling machine!

In JOG mode, you can measure a tool using a tool probe at a milling machine or lathe!

A description is provided in the following chapters as to which settings you can make in order to adapt measuring to the specific requirements of your particular machine.

21.10.3.1 Measuring workpieces at the milling machines

Measuring in the "Machine" operating area can be appropriately adapted to the specific requirements using the following channel-specific general machine data and channel-specific cycle setting data.

General configuration machine data

SD55630 \$SCS_MEA_FEED_MEASURE	Measuring feedrate for workpiece measurement in JOG and AUTOMATIC
300	Default value

Note**Measuring feedrate for workpiece measuring**

All measuring cycles use the value saved in SD54611 as the measuring feedrate value following calibration of the workpiece probe. A different measuring feedrate can be assigned for each calibration field [n].

When calibrating the probe, either the measuring feedrate from SD55630 \$SCS_MEA_FEED_MEASURE is used, or the measuring feedrate can be overwritten in the input screen when calibrating. To do this, SD54760 \$SNS_MEA_FUNCTION_MASK_PIECE bit 4 must be set to 1.

MD52207 \$MCS_AXIS_USAGE_ATTRIB[n]		Attributes of the axes [n] channel axis number
Bit 6	Request rotary axis as correction target for measuring.	
= 0	Default value	
= 1	Rotary axis is displayed as correction target in the "Measuring in JOG" window.	

MD51751 \$MNS_J_MEA_M_DIST_MANUELL		Measuring path in mm, before and after the measuring point.
= 10	Default value	

MD51757 \$MNS_J_MEA_COLL_MONIT_FEED		Position feedrate in mm/min, in the working plane for active collision monitoring.
= 1000	Default value	

MD51758 \$MNS_J_MEA_COLL_MONIT_POS_FEED		Position feedrate in mm/min, in the infeed axis for active collision monitoring.
= 1000	Default value	

MD51770 \$MNS_J_MEA_CAL_RING_DIAM[n]		Pre-assignment of the calibration diameter in mm specifically for the calibration data sets.
= -1	Default value	

MD51772 \$MNS_J_MEA_CAL_HEIGHT_FEEDAX[n]		Pre-assignment of the calibration height in mm, in the infeed axis, specific for the calibration data sets.
= -99999	Default value	

General cycle setting data

SD54780 \$SNS_J_MEA_FUNCTION_MASK_PIECE		Configuration of the input screens for measuring in the "Machine" operating area in the JOG operating mode.
= 512	Default value	
Bit 2 = 1	Activates measurements with an electronic probe.	
Bit 3 = 1	Selects the probe calibration data, enable.	
Bit 6 = 1	Selects WO offset in the basis reference (SETFRAME), enable.	
Bit 7 = 1	Selects WO offset in the channel-specific basic frame, enable.	
Bit 8 = 1	Selects WO offset in the global basic frame, enable.	
Bit 9 = 1	Selects WO offset in adjustable frames, enable.	
Bit 16 = 1	Selects workpiece measurement with spindle reversal.	
Bit 17 = 1	Selects align workpiece probe in the switching direction.	

21.10.3.2 Measuring tools at the milling machines**Measuring feedrate for tool measurement in JOG and AUTOMATIC**

SD55628 \$SCS_MEA_TP_FEED_MEASURE		Calibrate measuring feedrate for tool probe and measure tool with stationary spindle.
= 300	Default value	

Note**Measuring feedrate for tool measuring**

All measuring cycles use the value saved in SD54636 or SD54651 as the measuring feedrate after the tool probe has been calibrated. A different measuring feedrate can be assigned for each calibration field [n].

When calibrating the probe, either the measuring feedrate from SD55628 SCS_MEA_TP_FEED_MEASURE is used, or the measuring feedrate can be overwritten in the input screen when calibrating. To do this, SD54762 \$SNS_MEA_FUNCTION_MASK_TOOL bit 4 must be set to 1.

In the following setting data, index [k] stands for the number of the current data field (probe number -1) of the probe.

SD54633 \$SNS_MEA_TP_TYPE[k]		Probe type, cube/disk
= 0	Cube (default value)	
= 101	Disk in XY, working plane G17	
= 201	Disk in ZX, working plane G18	
= 301	Disk in YZ, working plane G19	

The following setting data is used to define in which axes and directions it is possible to calibrate a tool probe.

SD54632 \$SNS_MEA_T_PROBE_ALLOW_AX_DIR[k]	Axes and directions for "calibration"
= 133	Default value

Decimal place		
ONES		
1. Axis		
	= 0	axis not possible
	= 1	only minus direction
	= 2	only plus direction
	= 3	both directions
TENS		
2. Axis		
	= 0	axis not possible
	= 1	only minus direction
	= 2	only plus direction
	= 3	both directions
HUNDREDS		
3. Axis		
	= 0	axis not possible
	= 1	only minus direction
	= 2	only plus direction
	= 3	both directions

Example

If the general setting data SD54632 \$SNS_MEA_T_PROBE_ALLOW_AX_DIR[k] has the value 123, the tool probe is calibrated as follows in the G17 plane:

- X in both directions
- Y only in plus direction
- Z only in minus direction

SD54631 \$MNS_MEA_T_PROBE_DIAM_LENGTH[k]	Effective diameter of the tool probe for length measurement.
= 0	Default value

General cycle machine/setting data for tool probe

MD51780 \$MNS_J_MEA_T_PROBE_DIAM_RAD[k]	Effective diameter of the tool probe for radius measurement.
= 0	Default value

SD54634 \$SNS_MEA_TP_CAL_MEASURE_DEPTH[n]	Distance between the tool probe upper edge and the milling tool lower edge.
= 2	Default value

MD51752 \$MNS_J_MEA_M_DIST_TOOL_LENGTH	Measuring path to measure the tool length.
= 2	Default value

MD51753 \$MNS_J_MEA_M_DIST_TOOL_RADIUS	Measuring path to measure the tool radius.
= 1	Default value

MD51786 \$MNS_J_MEA_T_PROBE_MEASURE_DIST	Measuring path to calibrate the probe or for measurements with stationary spindle.
= 10	Default value

General cycle setting data when measuring with rotating spindle

SD54670 \$SNS_MEA_CM_MAX_PERI_SPEED[0]	Maximum permissible peripheral speed of the tool to be measured.
= 100	Default value

SD54671 \$SNS_MEA_CM_MAX_REVOLUTIONS[0]	Maximum permissible tool speed of the tool to be measured. The speed is automatically reduced when exceeded.
= 1000	Default value

SD54672 \$SNS_MEA_CM_MAX_FEEDRATE[0]	Maximum permissible feedrate to probe the tool to be measured at the probe.
= 20	Default value

SD54673 SNS_MEA_CM_MIN_FEEDRATE[0]	Minimum feedrate for the first probing of the tool to be measured at the probe. This avoids excessively small feedrates for large tool radii.
= 1	Default value

SD54674 \$SNS_MEA_CM_SPIND_ROT_DIR[0]	Spindle direction of rotation to measure tools.
= 4	Spindle rotation the same as M4 (default value)

Note

If the spindle is already rotating when the measuring cycle is called, this direction of rotation remains independent of the setting of this data.

SD54675 \$SNS_MEA_CM_FEEDFACTOR_1[0]	Feedrate factor 1
= 10	Default value
= 0	Only single probing with the feedrate calculated by the cycle. However, as a minimum, the value from SD54673[0] \$SNS_MEA_CM_MIN_FEEDRATE.
>= 1	First probing with feedrate. However, as a minimum with the value from SD54673[0] \$SNS_MEA_CM_MIN_FEEDRATE) SD54675[0] \$SNS_MEA_CM_FEEDFACTOR_1

SD54676 \$SNS_MEA_CM_FEEDFACTOR_2[0]	Feedrate factor 2
= 0	Second probing with the feedrate calculated by the measuring cycle. This is only effective for SD54673[0] \$SNS_MEA_CM_FEEDFACTOR_1 > 0, default value
>= 1	Second probing with the feedrate from SD54673[0] \$SNS_MEA_CM_MIN_FEEDRATE feedrate factor 2. Third probing with the calculated feedrate.

Note

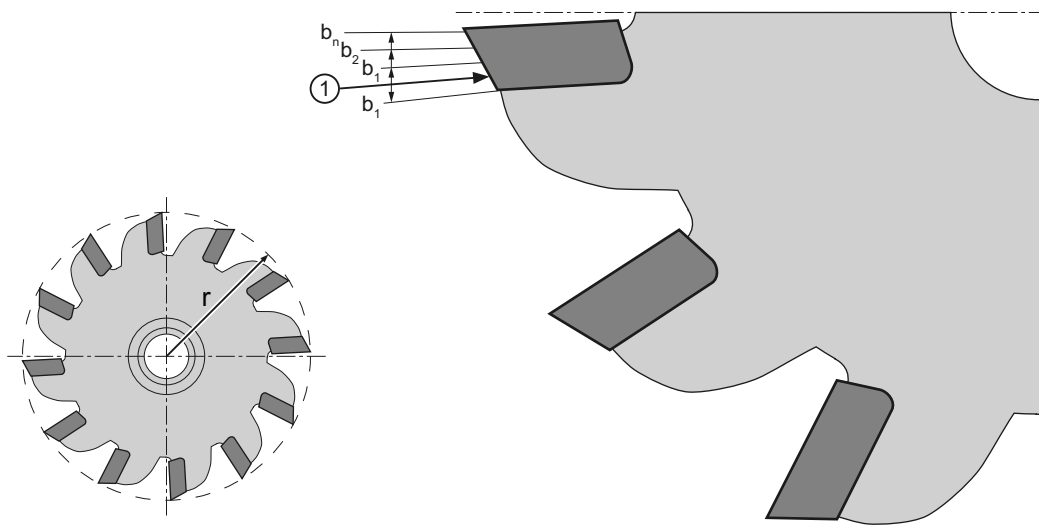
Feedrate factor 2 should be less than feedrate factor 1.

SD54677 \$SNS_MEA_CM_MEASURING_ACCURACY[0]	Specified measuring accuracy. The value of this parameter always refers to the last probing of the tool at the probe!
= 0.005	Default value

SD54692 \$SNS_MEA_T_CIRCULAR_ARC_DIST	Arc distance for the single cutting edge measurement of the radius.
= 0.25 °)	Default value

SD54693 \$SNS_MEA_T_MAX_STEPS		Maximum number of probe contacts for single cutting edge measurement of the radius.
= 10 ^{*)}	Default value	

- *) If the value = 0, then the "highest point on the longest cutting edge" is not sought, but the measured value of the contact with the rotating spindle taken.



- r Effective radius of the tool
b Arc distance = SD54692
n Maximum number of probe contacts = SD54693
① 1. Measuring point

Figure 21-11 Single cutting edge measurement

General cycle setting data for correction using the correction tables when forming measurements with rotating spindle.

SD54691 \$SNS_MEA_T_PROBE_OFFSET		Activate the measuring result compensation
= 0	No data, default value	
= 1	Correction in the cycle. This is only effective if SD54689 \$SNS_MEA_T_PROBE_MANUFACTURER>0.	
= 2	Correction using user-defined correction table	

SD54689 \$SNS_MEA_T_PROBE_MANUFACTURER		Activates pre-configured compensation tables for several tool probe models (customer-specific)
= 0	No data, default value	
= 1	TT130 (Heidenhain)	
= 2	TS27R (Renishaw)	

21.10.3.3 Measuring tools at the turning machines

Measuring feedrate for tool measurement in turning operation

SD55628 \$SCS_MEA_TP_FEED_MEASURE	Calibrate measuring feedrate for tool probe and measure tool with stationary spindle.
= 300	Default value

Note

Measuring feedrate for tool measuring

All measuring cycles use the value saved in SD54636 or SD54651 as the measuring feedrate after the tool probe has been calibrated. A different measuring feedrate can be assigned for each calibration field [n].

When calibrating the probe, either the measuring feedrate from SD55628 is used, or the measuring feedrate can be overwritten in the input screen when calibrating. To do this, SD54762 \$SNS_MEA_FUNCTION_MASK_TOOL bit 4 must be set to 1.

General cycle machine/setting data for measuring path/feedrate

SD42950 \$SC_TOOL_LENGTH_TYP	Assignment of the tool length offset independent of tool type.
= 0	Measuring turning tools, type 5xx (default value)
= 2	Measuring turning tools, type 5xx, drilling and milling tools, type 1xx, 2xx

MD51786 \$MNS_J_MEA_T_PROBE_MEASURE_DIST	Measuring path to calibrate the probe or for measurements with stationary spindle.
= 10	Default value

SD55628 \$SCS_MEA_TP_FEED_MEASURE	Feedrate for the calibration of a tool probe with stationary spindle.
= 300	Default value

Calibrating tool probes

Use the following general cycle machine data to set the access level as of which the "Calibrate probe" softkey is displayed. This requires that the automatic tool measurement is enabled using the setting data SD54782 (bit 2 = 1).

MD51070 \$MNS_ACCESS_CAL_TOOL_PROBE	Calibrate tool probe access level.
= 7	Access rights: Access level 7 (keyswitch 0)

= 4	Read access rights: Access level 4 (keyswitch 3), default value
= 3	Write access: Access level 3 (user)

21.10.3.4 Measuring without electronic probe in JOG

Measure workpiece

For manual measuring functions in the JOG operating mode, for which a rotating spindle is required, the following applies:

- The channel state must be reset at the instant in time that the measured value is transferred.
- The spindle can be moved or positioned either using the T,S,M menu or using the bits in DB3x.DBB30.
- The following preconditions apply when using the DB3x.DBB30 technology functions with fixed speed:

Setting the machine data			
MD10709 \$MN_PROG_SD_POWERON_INIT_TAB [0] = 0			Setting data to be initialized
MD35035 \$MA_SPIND_FUNCTION_MASK	Bit 4 = 0	Spindle functions	
	Bit 5 = 1		
SD43200 \$SA_SPIND_S			Speed setting value

Note

The entry in SD43200 \$SA_SPIND_S is kept at power on.

With these settings, the spindle velocity for manual operation, which is set in SD41200 \$SN_JOG_SPIND_SET_VELO, is no longer effective.

21.10.4 Measuring in AUTOMATIC mode

Requirement



Software option

In order to use the "Measuring in AUTOMATIC" function, you require the software option: "Measuring cycles"

You have already made the settings from Chapter "Measuring cycles and measurement functions, general (Page 546)".

Workpiece measurement

Sequence in milling technology:

1. The probe has been selected in the tool list as type 7xx (probe).
2. The probe has been inserted in the tool spindle.
3. The probe is activated in the actual NC channel.

Sequence in turning technology:

1. Tool type 580 (3D probe, turning) is selected.
2. Tool is activated in the actual NC channel.

Tool measurement

To measure tools, an appropriate probe must be located in the machine space so that this can be reliably and safely reached with a tool in the spindle.

The following tool types are supported with tool measurement:

- Milling technology: Tool types 1xx and 2xx
- Turning technology: Tool types 5xx, 1xx and 2xx

For the specified tool types, the tool lengths and the tool radii can be measured.

Channel-specific cycle setting data

Using the following channel-specific cycle setting data, you can adapt the **workpiece and tool measuring** in the "Program" operating area to specific requirements. As a general rule, no changes are required.

SD55613 \$SCS_MEA_RESULT_DISPLAY		Selects the screen display of the measuring result
= 0	No screen display of the measuring result (default value).	
= 1	Screen display of the measuring result is displayed for 8 seconds.	
= 3	The measuring cycle stops an internal machine data, the measuring result is statically displayed on the screen! Continue with NC start, the measuring result screen is deselected.	
= 4	The measuring result is only displayed on the screen for cycle alarms 61303, 61304, 61305, 61306. Continue with NC start, the measuring result display on the screen is deselected.	

SD55614 \$SCS_MEA_RESULT_MRD		Setting of the program control of the measurement result display MRD
= 0	Switch off the measurement result display	
= 1	Switch on the measurement result display	

SD55623 \$SCS_MEA_EMPIRIC_VALUE[n]	Empirical values
= 0	Default value

SD55618 \$SCS_MEA_SIM_ENABLE	Measuring cycle simulation
= 0	When calling measuring cycles in conjunction with SINUMERIK Operate, ShopMill or Shop-Turn simulation, the measuring cycles are skipped, default value.
= 1	Default setting: When calling measuring cycles in conjunction with SINUMERIK Operate, ShopMill or Shop-Turn simulation, the measuring cycles are run through. However, no corrections are performed and there is no logging. There is no screen display of the measuring result.

For further information on SD55618, please refer to:

Programming Manual, Measuring Cycles; Chapter "Behavior on block search, dry run, program testing, simulation"

Configuring input screens for measuring cycles in the program editor

Using the following setting data, you configure the display screen and the correction options when making measurements in the AUTOMATIC mode:

- SD54760 \$SNS_MEA_FUNCTION_MASK_PIECE measure workpiece: Measuring cycles in the milling area, see Chapter "Measuring workpieces, general (Page 564)"
- SD54762 \$SNS_MEA_FUNCTION_MASK_TOOL measure tool: Measuring cycles in the milling area, see Chapter "Measuring tools at the milling machines (Page 568)"

Note

If you have created the prerequisites described in this chapter and you have set and checked the machine/setting data, you can perform measurements at the machine with a workpiece or tool probe in the AUTOMATIC mode!

If you wish to set-up workpiece measurement on a lathe, then additional settings are required. Refer to Chapter "Measuring workpieces at the turning machines (Page 574)" for the settings.

If you wish to set-up tool measurement using a tool probe, then you must also make additional settings. Please refer to Chapters "Measuring tools at the turning machines (Page 575)" or "Measuring tools at the milling machines (Page 568)" for the settings.

A description is provided in the following chapters as to which settings you can make in order to adapt measuring to the specific requirements of your particular machine.

21.10.4.1 Measuring workpieces, general

Using the following general and channel-specific cycle setting data, you can appropriately adapt workpiece measurement to the specific requirements in the "Program" operating area.

Requirement

You have already made the settings from the following section:

See also: Measuring cycles and measurement functions, general (Page 546)

Settings

General cycle setting data

SD54760 \$SNS_MEA_FUNCTION_MASK_PIECE		Settings for the input screen, measuring cycles in AUTOMATIC, workpiece measurement.
Bit 1	Display select softkey 3D measurement	
Bit 3	Enable probe calibration field to be selected	
Bit 4	Select calibration input measuring feed ¹⁾	
Bit 6	Enable ZO to be selected in the basis reference (SETFRAME)	
Bit 7	Enable ZO to be selected in the channel-specific basis frame	
Bit 8	Enable ZO to be selected in the basis frame	
Bit 9	Enable ZO to be selected in the adjustable frame	
Bit 10	Enable ZO, coarse and fine, to be selected	
Bit 11	Select tool offset, geometry and wear	
Bit 12	Select tool offset, not inverted and inverted	
Bit 13	Select tool offset L1, R or L1, L2, L3 R	
Bit 14	Select tool offset, zero offset (_TZL)	
Bit 15	Select tool offset, dimensional difference monitoring (_TDIF)	
Bit 16	Select workpiece measurement with spindle reversal	
Bit 17	Selects align workpiece probe in the switching direction	
Bit 18	Select number of measurements (_NMSP)	
Bit 19	Select offset with mean value generation (_TMV) ¹⁾	
Bit 20	Select experience values (_EVNUM)	
Bit 21	Select total setting-up offset	
Bit 22	Select calibration to unknown or to known center point	
Bit 24	Select calibration with/without position deviation	
Bit 25	Select zero offset when measuring the angulation of the spindle	

Channel-specific cycle setting data

SD55630 \$SCS_MEA_FEED_MEASURE		Measuring feedrate [mm/rev]
= 300	Measuring feedrate when calibrating the workpiece probe	

Note**Measuring feedrate for workpiece measuring**

All measuring cycles use the value saved in SD54611 \$SNS_MEA_WP_FEED[n] as measuring feedrate after the tool probe has been calibrated.

A different measuring feedrate can be assigned for each calibration field [n].

When calibrating the probe, either the measuring feedrate from SD55630 \$SCS_MEA_FEED_MEASURE is used, or the measuring feedrate can be overwritten in the input screen when calibrating. To do this, SD54760 MEA_FUNCTION_MASK_PIECE bit 4 must be set to 1.

SD55632 \$SCS_MEA_FEED_RAPID_IN_PERCENT	Percentage reduction
= 50	Default value of the percentage reduction of the rapid traverse velocity for internal cycle intermediate positioning without collision monitoring. The setting of the channel-specific cycle setting data SD55600 SCS_MEA_COLLISION_MONITORING must be = 0.

SD55634 \$SCS_MEA_FEED_PLANE_VALUE	Feedrate of the intermediate positioning in the working plane with active collision monitoring.
= 1000	Default value

SD55636 \$SCS_MEA_FEED_FEEDAX_VALUE	Feedrate of the intermediate positioning in the infeed axis with active collision monitoring.
= 1000	Default value

SD55638 \$SCS_MEA_FEED_FAST_MEASURE	Fast measuring feedrate
= 900	Default value

SD55640 \$SCS_MEA_FEED_CIRCLE	Feedrate with circle programming
= 1000	Default value

SD55642 \$SCS_MEA_EDGE_SAVE_ANG	Safety angle when measuring a corner (CYCLE961)
= 0	Degrees

This angle is added to the angle that the user entered between the workpiece reference edge and the 1st axis of the active coordinate system in a positive direction, and refers to the corner measurement in the automatic mode (CYCLE961).

The angle that the user enters has the designation "α0" in the parameterizing screen form, corner measurement. For compatibility programs, the designation is _STA1.

Notice:

For compatibility programs, \$SCS_MEA_EDGE_SAVE_ANG should be set = 10.

21.10.4.2 Measuring workpieces at the milling machines

Measuring in the "Program" operating area can be adapted corresponding to the specific requirements using the channel-specific cycle setting data.

Settings

SD55625 \$SCS_MEA_AVERAGE_VALUE[n]	Number of mean values.
= 0	Default value

The following cycle setting data are written to by the measuring cycles with the "Calibrate probe" measuring function. User parameterization is not necessary here. However, after the probe has been calibrated, you can check these values and if required, evaluate the probe quality, e.g. for position deviations, no values > 0.1 mm should be reached. Otherwise, the probe must be mechanically readjusted.

The measuring feedrate at the time of calibration is used for all subsequent applications of the measuring cycles.

Note

Observe the manufacturer's instructions for the probe.

SD54600 \$SNS_MEA_WP_BALL_DIAM[n]	Effective diameter of the probe ball of the workpiece probe.
SD54601 \$SNS_MEA_WP_TRIG_MINUS_DIR_AX1[n]	Trigger point, minus direction, 1st measuring axis in the plane.
SD54602 \$SNS_MEA_WP_TRIG_PLUS_DIR_AX1[n]	Trigger point, plus direction, 1st measuring axis in the plane.
SD54603 \$SNS_MEA_WP_TRIG_MINUS_DIR_AX2[n]	Trigger point, minus direction, 2nd measuring axis in the plane.
SD54604 \$SNS_MEA_WP_TRIG_PLUS_DIR_AX2[n]	Trigger point, plus direction, 2nd measuring axis in the plane.
SD54605 \$SNS_MEA_WP_TRIG_MINUS_DIR_AX3[n]	Trigger point, minus direction, 3rd measuring axis in the tool direction.

SD54606 \$SNS_MEA_WP_TRIG_PLUS_DIR_AX3[n]	Trigger point, plus direction, 3rd measuring axis opposite to the tool direction. In the default case = 0.
SD54607 \$SNS_MEA_WP_POS_DEV_AX1[n]	Position deviation, 1st measuring axis in the plane.
SD54608 \$SNS_MEA_WP_POS_DEV_AX2[n]	Position deviation, 2nd measuring axis in the plane.
SD54609 \$SNS_MEA_WP_STATUS_RT	Calibration status of the axis positions
SD54610 \$SNS_MEA_WP_STATUS_GEN[n]	Calibration status
SD54611 \$SNS_MEA_WP_FEED[n]	Measuring feedrate when calibrating

21.10.4.3 Measuring tools at the milling machines

Settings

SD54762 \$SNS_MEA_FUNCTION_MASK_TOOL	Setting for the input screen, measuring cycles in the automatic mode, tool measurement
Bit 3	Setting for the input screen, measuring cycles in the automatic mode, tool measurement, enable tool probe calibration data field to be selected
Bit 4	Select calibration input measuring feed (VMS) ¹⁾
Bit 5	Select input feedrate and spindle speeds when probing
Bit 7	Select measurement in MCS and WCS
Bit 8	Select measurement, absolute and incremental
Bit 9	Select tool offset, geometry and wear
Bit 10	Select individual cutting edge measurement
Bit 11	Selects spindle reversal when calibrating in the plane
Bit 12	Select number of measurements (_NMSP)
Bit 13	Select experience values (_EVNUM)

¹⁾ Input, measurement feedrate is valid for AUTOMATIC and JOG

Calibration data of the tool probe, referred to the machine coordinate system

Before calibration is started, the position of the tool probe in the machine coordinate system (MCS) must be entered into the following general cycle setting data. In this case, the reference point is the outer diameter or the tool length of the active tool in the spindle. If there is no tool in the spindle, the reference points are the spindle center point and the tool reference point at the spindle.

Note**Calibrate probe**

If you have calibrated the tool probe in JOG mode, then the calibration data has already been correctly entered in: SD54632 \$SNS_MEA_TP_AX_DIR_AUTO_CAL[k]

You do not need to recalibrate the tool probe in the AUTOMATIC mode.

Index [k] stands for the number of the actual data field (probe number -1).

SD54625 \$SNS_MEA_TP_TRIG_MINUS_DIR_AX1[k]	Trigger point of the 1st measuring axis in the negative direction.
SD54626 \$SNS_MEA_TP_TRIG_PLUS_DIR_AX1[k]	Trigger point of the 1st measuring axis in the positive direction.
SD54627 \$SNS_MEA_TP_TRIG_MINUS_DIR_AX2[k]	Trigger point of the 2nd measuring axis in the negative direction.
SD54628 \$SNS_MEA_TP_TRIG_PLUS_DIR_AX2[k]	Trigger point of the 2nd measuring axis in the positive direction.
SD54629 \$SNS_MEA_TP_TRIG_MINUS_DIR_AX3[k]	Trigger point of the 3rd measuring axis in the negative direction.
SD54630 \$SNS_MEA_TP_TRIG_PLUS_DIR_AX3[k]	Trigger point of the 3rd measuring axis in the positive direction.
SD54631 \$SNS_MEA_TP_EDGE_DISK_SIZE[k]	Tool probe, edge length / disk diameter.
SD54632 \$SNS_MEA_TP_AX_DIR_AUTO_CAL[k]	Axes and directions for calibrating in AUTOMATIC mode.
SD54634 \$SNS_MEA_TP_CAL_MEASURE_DEPTH[k]	Distance between the upper edge of the tool probe and lower edge of the tool (calibration depth, measuring depth for milling radius).
SD54635 \$SNS_MEA_TPW_STATUS_GEN[k]	Calibration status
SD54636 \$SNS_MEA_TPW_FEED[k]	Measuring feedrate when calibrating

The general cycle setting data SD54632 \$SNS_MEA_TP_AX_DIR_AUTO_CAL, is used to define in which axes and directions it is possible to calibrate the tool probe.

Decimal place		
ONES		1st axis
	= 0	axis not possible
	= 1	only minus direction
	= 2	only plus direction
	= 3	both directions
TENS		2nd axis
	= 0	axis not possible
	= 1	only minus direction
	= 2	only plus direction
	= 3	both directions

Decimal place		
HUNDREDS	3rd axis	
	= 0	axis not possible
	= 1	only minus direction
	= 2	only plus direction
	= 3	both directions

Example

If the general cycle machine data SD54632 \$SNS_MEA_TP_AX_DIR_AUTO_CAL has the value 123, the tool probe is calibrated as follows in the G17 plane:

- X in both directions
- Y only in plus direction
- Z only in minus direction

SD54633 \$SNS_MEA_TP_TYPE[k]	Probe version
= 0	Cube, default value.
= 101	Disk in XY, working plane G17.
= 201	Disk in ZX, working plane G18.
= 301	Disk in YZ, working plane G19.

Calibration data of the tool probe referred to the workpiece coordinate system

Before calibration is started, the position of the tool probe in the workpiece coordinate system (WCS) must be roughly entered into the following general cycle setting data. In this case, the reference point is the outer diameter or the tool length of the active tool in the spindle. If there is no tool in the spindle, the reference points are the spindle center point and the tool reference point at the spindle.

Note

When measuring tools, ensure that the data of the adjustable zero offset or the basic reference always correspond to the data when calibrating (measuring in WCS!).

Always make measurements and calibrate with the same adjustable zero offset.

SD54640 \$SNS_MEA_TPW_TRIG_MINUS_DIR_AX1[k]	Trigger point of the 1st measuring axis in the negative direction.
SD54641 \$SNS_MEA_TPW_TRIG_PLUS_DIR_AX1[k]	Trigger point of the 1st measuring axis in the positive direction.
SD54642 \$SNS_MEA_TPW_TRIG_MINUS_DIR_AX2[k]	Trigger point of the 2nd measuring axis in the negative direction.
SD54643 \$SNS_MEA_TPW_TRIG_PLUS_DIR_AX2[k]	Trigger point of the 2nd measuring axis in the positive direction.

SD54644 \$SNS_MEA_TPW_TRIG_MINUS_DIR_AX3[k]	Trigger point of the 3rd measuring axis in the negative direction.
SD54645 \$SNS_MEA_TPW_TRIG_PLUS_DIR_AX3[k]	Trigger point of the 3rd measuring axis in the positive direction.
SD54646 \$SNS_MEA_TPW_EDGE_DISK_SIZE[k]	Tool probe, edge length / disk diameter.
SD54647 \$SNS_MEA_TPW_AX_DIR_AUTO_CAL[k]	Automatic calibration of tool probe, enable axes/directions
SD54648 \$SNS_MEA_TPW_TYPE[k]	Probe version
= 0	Cube (default value)
= 101	Disk in XY, working plane G17.
= 201	Disk in ZX, working plane G18.
= 301	Disk in YZ, working plane G19.
SD54649 \$SNS_MEA_TPW_CAL_MEASURE_DEPTH[k]	Distance between the upper edge of the tool probe and lower edge of the tool (calibration depth, measuring depth for milling radius).
SD54650 \$SNS_MEA_TPW_STATUS_GEN[k]	Calibration status
SD54651 \$SNS_MEA_TPW_FEED[k]	Measuring feedrate when calibrating

The following general cycle setting data SD54647 \$SNS_MEA_TPW_AX_DIR_AUTO_CAL is used to define in which axes and directions it is possible to calibrate a tool probe.

Decimal place		
ONES	1. Axis	
	= 0	axis not possible
	= 1	only minus direction
	= 2	only plus direction
	= 3	both directions
TENS	2. Axis	
	= 0	axis not possible
	= 1	only minus direction
	= 2	only plus direction
	= 3	both directions
HUNDREDS	3. Axis	
	= 0	axis not possible
	= 1	only minus direction
	= 2	only plus direction
	= 3	both directions

Example

If the general cycle machine data SD54647 \$SNS_MEA_TPW_AX_DIR_AUTO_CAL has the value 123, the tool probe is calibrated as follows in the G17 plane:

- X in both directions
- Y only in plus direction
- Z only in minus direction

Monitoring when measuring with a rotating spindle

SD54670 \$SNS_MEA_CM_MAX_PERI_SPEED[0]	Maximum permissible peripheral speed of the tool to be measured.
= 100	Default value

SD54671 \$SNS_MEA_CM_MAX_REVOLUTIONS[0]	Maximum permissible tool speed of the tool to be measured. The speed is automatically reduced when exceeded.
= 1000	Default value

SD54672 \$SNS_MEA_CM_MAX_FEEDRATE[0]	Maximum permissible feedrate to probe the tool to be measured at the probe.
= 20	Default value

SD54673 \$SNS_MEA_CM_MIN_FEEDRATE[0]	Minimum feedrate for the first probing of the tool to be measured at the probe. This avoids excessively small feedrates for large tool radii.
= 1	Default value

SD54674 \$SNS_MEA_CM_SPIND_ROT_DIR[0]	Spindle direction of rotation to measure tools.
4 = M4	Default value

Note

If the spindle is already rotating when the measuring cycle is called, this direction of rotation remains independent of the setting of this data.

SD54675 \$SNS_MEA_CM_FEEDFACTOR_1[0]	Feedrate factor 1
= 10	Default value

= 0	Only single probing with the feedrate calculated by the cycle. However, as a minimum, the value from SD54673[0] \$SNS_MEA_CM_MIN_FEEDRATE.
= ≥ 1	First probing with feedrate. However, as a minimum with the value from SD54673[0] \$SNS_MEA_CM_MIN_FEEDRATE) SD54675[0] \$SNS_MEA_CM_FEEDFACTOR_1

SD54676 \$SNS_MEA_CM_FEEDFACTOR_2[0]	Feedrate factor 2
= 0	Second probing with the feedrate calculated by the cycle. This is only effective for SD54673 \$SNS_MEA_CM_FEEDFACTOR_1[0] > 0, default value.
= ≥ 1	Second probing with the calculated feedrate from SD54673 \$SNS_MEA_CM_MIN_FEE- DRATE[0] feedrate factor 2. Third probing with the calculated feedrate.

Note

Feedrate factor 2 should be less than feedrate factor 1.

SD54677 \$SNS_MEA_CM_MEASURING_ACCURACY[0]	Specified measuring accuracy. The value of this parameter always refers to the last probing of the tool at the probe.
= 0.005	Default value

Measurement with rotating spindle: Measured value correction using correction tables

SD54691 \$SNS_MEA_T_PROBE_OFFSET	Activates the measuring result correction
= 0	No data, default value
= 1	Correction in the cycle. This is only effective if SD54690 \$SNS_MEA_T_PROBE_MANUFACTURER>0.
= 2	Correction using user-defined correction table

SD54689 \$SNS_MEA_T_PROBE_MANUFACTURER	Activate pre-configured compensation tables for several tool probe models (customer-specific).
= 0	No data, default value
= 1	TT130 (Heidenhain)
= 2	TS27R (Renishaw)

Correction values for users

If the general cycle setting data SD54691 \$SNS_MEA_T_PROBE_OFFSET= 2, the following settings apply:

SD54695 to SD54700	Correction values for radius measurement.	See the subsequent general cycle setting data.
SD54705 to SD54710	Correction values for length measurement.	See the subsequent general cycle setting data.

SD54695 \$SNS_MEA_RESULT_OFFSET_TAB_RAD1[n]	Radius measurement
SD54705 \$SNS_MEA_RESULT_OFFSET_TAB_LEN1[n]	Length measurement
= 0	0
= 1	1. Radius
= 2	2. Radius
= 3	3. Radius
= 4	4. Radius

21.10.4.4 Measuring workpieces at the turning machines**General settings for turning**

MD52740 \$MNS_MEA_FUNCTION_MASK	Measuring cycles function screen
Bit 1	Workpiece measurement, measuring in the third geometry axis (Y, when turning
= 0	The measuring cycles do not support a third geometry axis (Y axis)!
=1	Setpoint input and parameterization (SETVAL, _TUL, _TLL, SZO) refer to the third geometry axis (Y axis). The correction of the tool length or zero offset is however realized in the second geometry axis (X axis, ordinate) active components (i.e. measure in Y and correct in X). The correction target can be influenced using parameter _KNUM!
Bit 16	Measuring input, tool probe

Calibration data of the tool probe, referred to the machine coordinate system

Before calibration is started, the position of the tool probe in the machine coordinate system (MCS) must be entered into the following general cycle setting data.

SD54615 \$SNS_MEA_CAL_EDGE_BASE_AX1[n]	Calibration slot base referred to the 1st measuring axis.
SD54617 \$SNS_MEA_CAL_EDGE_PLUS_DIR_AX1[n]	Calibration slot edge in the positive direction of the 1st measuring axis.
SD54618 \$SNS_MEA_CAL_EDGE_MINUS_DIR_AX1[n]	Calibration slot edge in the negative direction of the 1st measuring axis.
SD54619 \$SNS_MEA_CAL_EDGE_BASE_AX2[n]	Calibration slot base referred to the 2nd measuring axis.
SD54620 \$SNS_MEA_CAL_EDGE_UPPERE_AX2[n]	Upper calibration slot edge referred to the 2nd measuring axis.

SD54621 \$SNS_MEA_CAL_EDGE_PLUS_DIR_AX2[n]	Calibration slot edge in the positive direction of the 2nd measuring axis.
SD54622 \$SNS_MEA_CAL_EDGE_MINUS_DIR_AX2[n]	Calibration slot edge in the negative direction of the 2nd measuring axis.

Note

For a standard lathe with axes X and Z (G18), axis Z is the 1st measuring axis and axis X is the 2nd measuring axis.

21.10.4.5 Measuring tools at the turning machines

Calibration data of the tool probe referred to the machine coordinate system

If you wish to calibrate the tool probe in the machine coordinate system, then the position of the tool probe in the machine coordinate system must be entered into the following general cycle setting data.

SD54625 \$SNS_MEA_TP_TRIG_MINUS_DIR_AX1[k]	Trigger point in minus direction of the 1st measuring axis (for G18 Z)
SD54626 \$SNS_MEA_TP_TRIG_PLUS_DIR_AX1[k]	Trigger point in plus direction of the 1st measuring axis (for G18 Z)
SD54627 \$SNS_MEA_TP_TRIG_MINUS_DIR_AX2[k]	Trigger point in minus direction of the 2nd measuring axis (for G18 X).
SD54628 \$SNS_MEA_TP_TRIG_PLUS_DIR_AX2[k]	Trigger point in plus direction of the 2nd measuring axis (for G18 X).

Calibration data of the tool probe referred to the workpiece coordinate system

If you wish to calibrate the tool probe in the workpiece coordinate system, then the position of the tool probe in the workpiece coordinate system must be entered into the following general cycle setting data. In this case, the reference point is the outer diameter or the tool length of the active tool in the spindle.

Index [k] stands for the number of the actual data field (probe number -1).

SD54640 \$SNS_MEA_TPW_TRIG_MINUS_DIR_AX1[k]	Trigger point minus direction of the 1st measuring axis (for G18 Z).
SD54641 \$SNS_MEA_TPW_TRIG_PLUS_DIR_AX1[k]	Trigger point plus direction of the 1st measuring axis (for G18 Z).
SD54642 \$SNS_MEA_TPW_TRIG_MINUS_DIR_AX2[k]	Trigger point minus direction of the 2nd measuring axis (for G18 X).
SD54643 \$SNS_MEA_TPW_TRIG_PLUS_DIR_AX2[k]	Trigger point plus direction of the 2nd measuring axis (for G18 X).

Tool measurement using the "Orientable toolholder" or "Swivel tool" function

If the general cycle machine data MD52740 \$MNS_MEA_FUNCTION_MASK, bit 16 = 1, then the following setting applies:

MD51618 \$MNS_MEA_CM_ROT_AX_POS_TOL	Tolerance parameter for rotary axis settings
= 0,5	Default value

The real angular position of rotary axes can deviate from that programmed (exact stop fine window) This deviation depends on the position control properties of the axis. The maximum deviation that can be expected at a specific axis should be entered into the parameter. When the tolerance is exceeded, Alarm 61442 is output - "Toolholder not parallel to the geometry axes".

21.10.5 Logging

21.10.5.1 Logging the measuring, general

General logging can be performed without special settings:

- In the passive file system of the NC
- On the local drive of the CF card

If you want to use a logical drive for the logging, you require the Execution from External Storage (EES) function, see Chapter "Setting the EES (Page 49)".

General machine data

Logging can be controlled via the following machine data settings:

MD11422 \$MN_PROTOCOL_FILE_MODE	Setting the behavior of the WRITE command
Bit 0	Persistence behavior
= 0	time-delayed WRITE command (immediately persistent)
= 1	= fast WRITE command (time-delayed persistent)

Note

Set MD11422 bit 0 = 1 so that the program runtimes are not lengthened significantly.

MD11420 \$MN_LEN_PROTOCOL_FILE	File size for log files (KB)
100	Default value

Increase the value of the machine data if larger log files are to be stored in the file system of the NC.

21.10.5.2 Logging while measuring in the JOG mode

You can set the following setting data in the "Settings for Measurement Log" window for the JOG mode.

References

Operating Manual, Turning/Milling/Grinding; Chapter "Settings for the measurement result log"

Cycle channel setting data

SD55774 \$SCS_J_MEA_PROTOCOL_FILE	Name and path for the log file for measuring in JOG
//NC:/WKS.DIR/TEMP.WPD/J_MEAPROT.TXT	Default setting

SD55740 \$SCS_MEA_FUNCTION_MASK	Measuring cycles function screen, workpiece/tool measurement
Bit 28	Select new log or continuous log for measuring standard log in JOG
= 0	New log (default value)
= 1	Continuous, i.e. always attach to the existing log
Bit 29	Select file format for measuring standard log in JOG
= 0	Text format (default value)
= 1	Tabular format

21.10.5.3 Logging while measuring in the AUTOMATIC mode

Cycle channel setting data

SD55730 \$SCS_MEA_PROTOCOL_USER_EXT	File extension for user log
TXT	Default setting

SD55614 \$SCS_MEA_RESULT_MRD	Setting of the program control of the measurement result display MRD
= 0	Switch off the measurement result display
= 1	Switch on the measurement result display (default value)

Definition file: User GUD block

Data for user logs are generally logged from the S_PROTTXT[10] GUD array (PGUD).

If the data array is not sufficient, you can create a second array with the predefined name S_USERTXT[n] in a separate GUD block (e.g. MGUD or UGUD):

```
DEF NCK STRING[200] S_USERTXT[n] ; n = number of field elements
```

Note

The logging operation checks whether the S_USERTXT array is available. If S_USERTXT is available, the content of the array is logged. If S_USERTXT is not available, the content of the S_PROTTXT array is logged.

21.10.5.4 Manufacturer and user cycle CUST_MEAPROT.SPF**Function**

You can make individual adaptations to the log form and content in the CUST_MEAPROT cycle.

Procedure

1. Open the CUST_MEAPROT.SPF cycle file.
2. Select the program lines in the _MARK1 block.
3. Copy the contents from the _MARK1 block to the _MARK2 block.
4. Adapt the parameters individually in the _MARK2 block.
5. Save the changed cycle file.
The original default values are retained and can be activated/deactivated at any time (setting of comment characters).

Parameter

In the CUST_MEAPROT cycle, all the possible parameter changes are described in detail in the _MARK2_1 block.

You can adapt the following parameters:

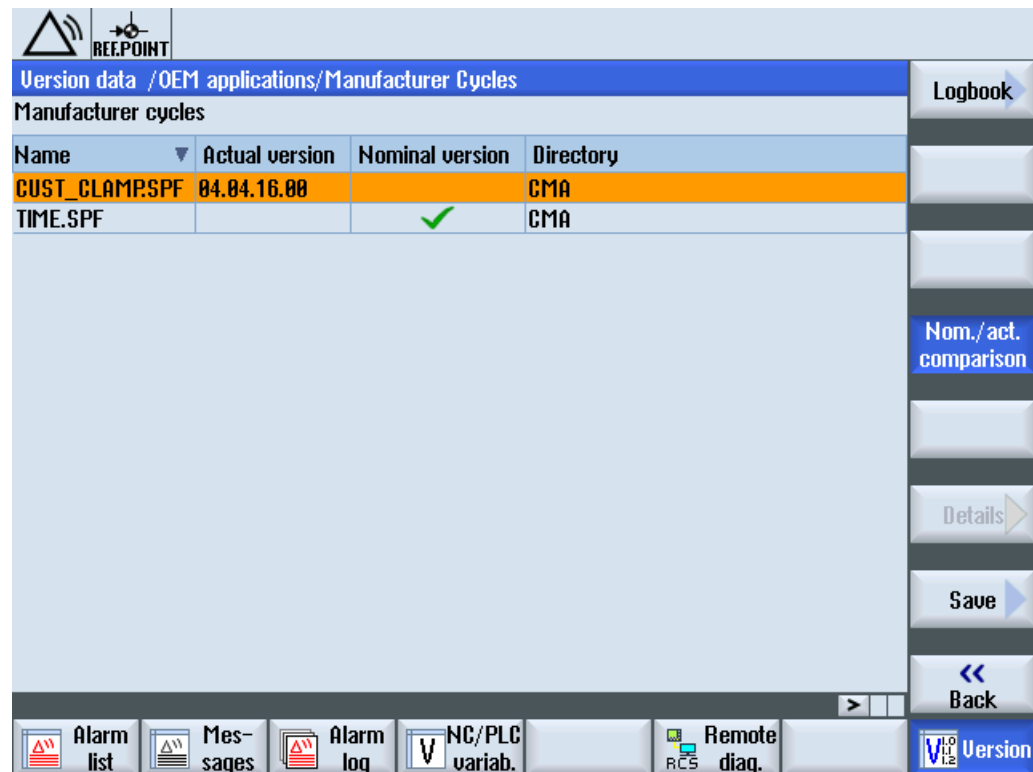
Parameter	Default value	Input area
Characters per line	66	> 66 possible ⁽¹⁾
Characters per column 2 to 4	14	Min. 12, max. 14
Workpiece counter	""	Identifier
Column space	" "	Blank string (≥ 2 * blanks)
Identifier for tolerance specification	"#"	1 character
Identifier for tolerance violation	"!"	1 character
Log header lines 5 to 7	""	Any text

- ¹⁾ 66 characters per line correspond to the screen width in the default setting. A log line is therefore displayed completely on the screen. If logging is mainly performed in the file system of the NC, this setting should be retained because of the better readability. If you perform logging mainly in external files, you can increase the line width accordingly. In this case, the width of column 1 is increased automatically (with the same width for columns 2, 3 and 4 and the same column space).

21.11 Compare cycles version

21.11.1 Display cycles version

If cycles have version details, you can display them in the version screen.



Precondition

The version details are included in the cycles files in the following form:

```
;VERSION: <Version> ;DATE: <YYYY-MM-DD>
```

Example:

```
;VERSION: 05.05.05.00 ;DATE: 2012-11-30
```

Procedure



1. Select the "Diagnostics" operating area.



2. Press the "Version" softkey.

It takes some time to call the version display. While the version data is being determined a progress message box and the appropriate text are displayed in the dialog line.



3. Select the "OEM applications /manufacturer cycles" areas and press the "Details" softkey.

The "Manufacturer Cycles" window opens.

The "Planned-actual comparison" softkey is selected by default.

See also

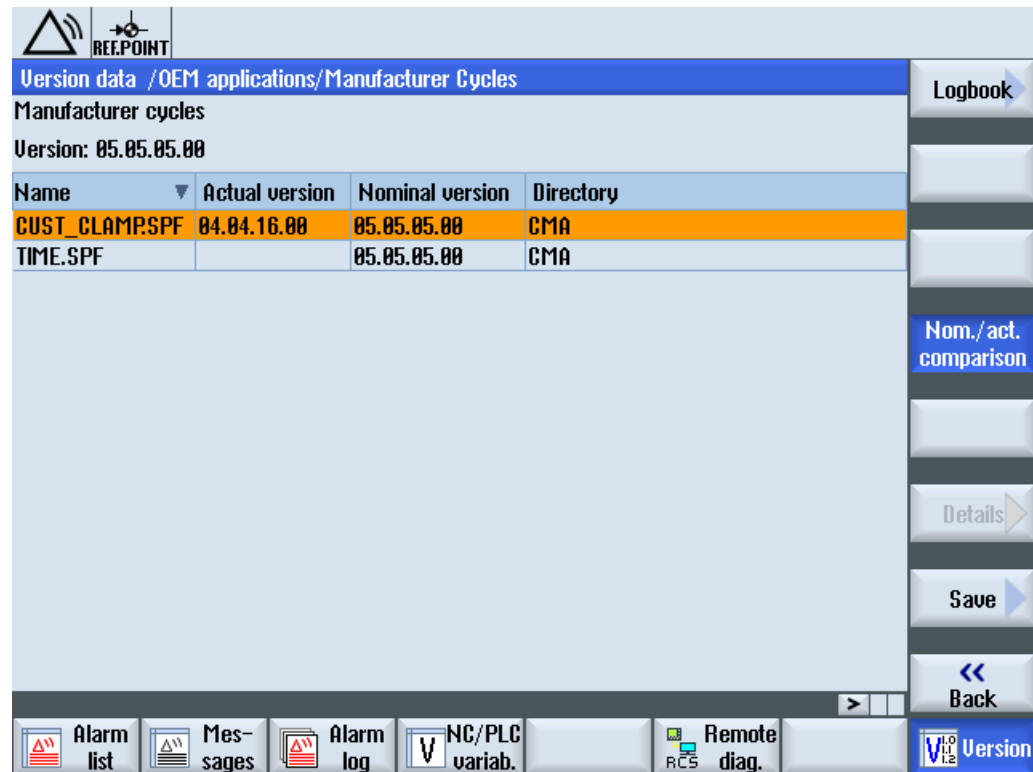
Save information (Page 354)

Logbook (Page 357)

21.11.2 Specify cycles version

General

You have the possibility to compare a required cycles version with the existing cycles version in the version view.



File overview

The following files are required:

Name	Path	Meaning
versions.xml	/siemens/sinumerik/hmi/data/version/oem/cma	Version file
versions.xml	/oem/	Version path file

Adapting the version file

The cycles version specification requires a manual adaptation of the version file.

1. Copy the "versions.xml" sample file from the following directory:
/siemens/sinumerik/hmi/data/version/oem/cma
2. Save the sample file in its own directory under /oem.
The standard directory for this purpose is /oem/sinumerik/hmi/data/version/oem/cma.
3. Open the file.

4. Enter the planned version for the cycles files in the <info> tag with "defaultFileVersion" and the details for the complete version in the <Version> tag.
5. Close the file to save the changes.

Example:

```
<info defaultFileType=" *.spf *.cpf" defaultFileVersion="05.05.00.00"
linkname="CMA" linkpath="//NC/CMA.DIR" EffDirOrder="CUS CMA">
<Name>Manufacturer cycles</Name>
<Version>5.5.0.0</Version>
<Link>
<Name>CUS</Name>
<Path>//NC/CUS.DIR</Path>
</Link>
</info>
```

Adapting the version path file

If the version file is saved in the standard directory, and a current oem\versions.xml is available, no changes are required. In all other cases, the path data can be manually changed:

1. Open the versions.xml file under /oem
2. Replace the standard path in the <Path> tag for the "Manufacturers Cycles" with your own path for the version file.
3. Close the file to save the changes.

Standard:

```
<Component>
<Name>Manufacturer Cycles</Name>
<Path>/siemens/sinumerik/hmi/data/version/oem/cma</Path>
</Component>
```

Example:

```
<Component>
<Name>Manufacturer Cycles</Name>
<Path>/oem/sinumerik/hmi/data/version/oem/cma</Path>
</Component>
```

Additional language

22.1 Installing additional languages

Additional languages

In addition to the already existing standard languages, additional user interface languages are available on a separate DVD.

The integrated online help is either displayed in the relevant language of the country or in English.

Refer to the following chapter for the languages that are available for the user interface:
Supported languages (Page 584)

Precondition



Software option

You require the following software option in order to use this function:

"Additional languages"

- After the installation, approx. 256 MB memory must be available on the CompactFlash Card.
- Language extension software "hmi_sl_language_xxx_02.06.00.00y.arc".
xxx = language code, y = internal version ID
Is supplied on the language extension DVD.

Installing a language

Install the language the same as for a commissioning archive (ARC). For the procedure, please refer to the following Chapter: Reading-in a start-up archive (Page 280)

Note

Additional software language of the previous version is compatible to the actual operating software.

22.2 Uninstalling languages

If you no longer require a language interface, then you can uninstall this.

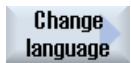
Note

Standard languages and the currently active language for the operating software cannot be uninstalled.

Procedure



1. Select the "Startup" operating area.



2. Press the "Change language" softkey.
The "Language selection" window opens. The language most recently set is highlighted.



3. Switch the cursor to the required language.

4. Press the "Delete language" softkey.
The "Delete language" window opens.



5. Press the "OK" softkey.
The operating software uninstalls the selected language.

22.3 Supported languages

Table 22-1 Supported languages

Language	Language code	Standard languages
Chinese (simplified)	chs	X
Chinese (traditional)	cht	
Danish	dan	
German	deu	X
English	eng	X
Finnish	fin	
French	fra	X
Indonesian	ind	
Italian	ita	X
Japanese	jpn	
Korean	kor	
Malaysian	msl	
Dutch	nld	
Polish	plk	

Language	Language code	Standard languages
Portuguese	ptb	
Romanian	rom	
Russian	rus	
Swedish	sve	
Slovenian	slv	
Slovakian	sky	
Spanish	esp	X
Thai	tha	
Czech	csy	
Turkish	trk	
Hungarian	hun	
Vietnamese	vit	

SINUMERIK Operate on PC/PCU

23.1 Link OEMFrame application

Integrating an OEM application

The following files should be adapted and created in order to integrate an OEMFrame application into the operating software:

- Configuration file `systemconfiguration.ini`
The operating software is started and controlled from the system manager; the system manager also controls the OEMFrame applications. The system manager is configured using the "systemconfiguration.ini" configuration file.
- Configuring the start softkey
In order to start an OEMFrame application from the operating software, configure a softkey on the expansion bar of the operating area.
- Configuration file "slamconfig.ini"
In order to configure the softkey position with text and/or symbol for the OEMFrame application, generate the "slamconfig.ini" file.
- Save the language-dependent text for a softkey in the file `mytext_<lng>.ts`.

"systemconfiguration.ini" file

In order to integrate an OEMFrame application into the operating software, copy the "systemconfiguration.ini" configuration file and place it in one of the two directories:

<Installation path>/**user**/sinumerik/hmi/cfg

<Installation path>/**oem**/sinumerik/hmi/cfg

All of the processes to be managed by the system manager, as well as the applications that are to be integrated as OEMFrame applications are in the **[processes]** section.

Value	Meaning
process	Symbolic name of the OEMFrame application. This is required to configure the operating areas.
cmdline	Command line which is transferred to the "oemframe.exe" process when starting.
oemframe	For OEMFrame applications, always set to "true".
windowname	Window name of the OEMFrame application - should be determined with "find-window.exe" or "spy++.exe".
classname	Class name of the OEMFrame application - should be determined with "findwindow.exe" or "spy++.exe".
deferred	true: OEMFrame application is not started when SINUMERIK Operate powers up, but only when selected for the first time.

Example

In the following example, the two Windows applications "notepad.exe" and "calc.exe" are configured as OEMFrame applications.

```
[processes]

PROC600= process:=notepadOEM,cmdline:="notepad.exe mytext1.txt",
oemframe:=true, deferred:=true, windowname:="mytext1.txt - Notepad",
classname:="Notepad"

PROC601= process:=calcOEM, cmdline:="calc.exe" oemframe:=true,
windowname:="calculator"
```

Section [areas]

The SINUMERIK Operate operating areas are configured in this section.

Value	Meaning
name	Symbolic name for the operating area.
process	Name of the OEMFrame application according to section [processes].

Note

Permissible number range

In the "processes" and "areas" sections, the number range 500-999 is reserved for OEM customers. If you use numbers less than 500, then it is possible that Siemens basis components will be overwritten!

Example

```
[areas]

AREA600= name:=AreaOEM, process:=notepadOEM
AREA601= name:=AreaCalc, process:=calcOEM
```

Note

Only OEMFrame applications are supported that do not use any of the programming interfaces of SINUMERIK Operate.

Section [miscellaneous]

You can make various settings in this section. Generally, only the start operating area is changed.

Key	Value
startuparea	Name of the start operating area

Example

```
[miscellaneous]
startuparea = AreaOEM
```

Configuring the operating area menu

The operating area menu is intended for switching over the operating areas configured in the "systemconfiguration.ini" configuration file. A softkey for selecting the appropriate operating area is provided on the horizontal softkey bar for each operating area configured.

The operating area displays the names of the operating areas from the "systemconfiguration.ini" configuration file as the text on the operating area softkeys. The system automatically searches for a free softkey on the horizontal softkey bar for each operating area.

Configuring additional settings

In order to configure the following settings you require the "slamconfig.ini" configuration file:

- Assigning a softkey position to a specific operating area.
- Creating a language-dependent text for the softkey.
- Displaying a symbol for the operating area on the softkey.

Creating the "slamconfig.ini" configuration file

Copy the "slamconfig.ini" and place the file in the same directory in which the "systemconfiguration.ini" file is located:

<Installation path>/user/sinumerik/hmi/cfg

<Installation path>/oem/sinumerik/hmi/cfg

"slamconfig.ini" file

In the "slamconfig.ini" configuration file, for every operating area, you can create a section that was configured in the "systemconfiguration.ini" file. The section must bear the configured name of the appropriate operating area, e.g. AreaOEM.

Value	Meaning
TextId	Text ID for a foreign-language text which will be displayed as the softkey label.
TextContext	Context of the foreign-language text.
TextFile	Name of the text file which includes the context and the foreign-language text.
Graphic	Name of an image file which will be used as an icon for the softkey.
SoftkeyPosition	Fixed softkey position of the area softkey. In this case, softkey positions 1 to 8 are located on the 1st horizontal bar and softkey positions 9 to 16 on the 2nd horizontal bar, etc.
AccessLevel	Access level from which the softkey will be displayed. If this value is not specified, the access level 7 (keyswitch position 0) is set.

Example

The softkey for the "AreaOEM" operating area with the following properties is configured in the following example:

- The softkey displays the text which has been stored in the "mytext_<lng>.ts" text file under the context "mycontext" with the "MY_AREA" TextID.
- The "mypicture.png" icon is displayed on the softkey.
- The softkey is located at position 7 in the operating area menu.
- The softkey with access level 5 (keyswitch position 2) is displayed.

```
[AreaOEM]
; Text-ID of a language dependent text
TextId = MY_AREA
; File name of the text file which contains the Text-ID
TextFile = mytext
; Context in the text file to which the Text-ID is assigned to
TextContext = mycontext
; File name of an icon shown on the area softkey
Picture = mypicture.png
; Position of the area softkey on area menu,
; If no position is specified, an empty position is searched
SoftkeyPosition = 7
; Access level of the area softkey
AccessLevel = 5
```

Note

Operating area position 7 is reserved for OEM customers.

Labeling text for the softkey

Storage path:

<Installation path>/**user**/sinumerik/hmi/lng

<Installation path>/**oem**/sinumerik/hmi/lng

The XML identifiers have the following meanings:

Attribute	Description
context	Context within the text file. Each file must have at least one context.
name	Name of the context.

Attribute	Description
message	Text translation. There must be at least one message per context.
source	Text identifier.
translation	Translated text.
remark	Text comment (optional).
chars	Maximum possible length of the text in characters. If nothing is specified, the text can have any length (optional).
lines	Maximum number of lines available for display. If nothing is specified, the number of lines is unlimited (optional).

Structure of the language-dependent ts file that contains the labeling text for the softkey:

```

mytext_<lng>.ts
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<!DOCTYPE TS>
<TS>
  <context>
    <name>mycontext</name>
    <message>
      <source>MY_AREA</source>
      <translation>Text, which is displayed on the softkey</translation>
      <remark>Remark (optional)</remark>
      <chars>20</chars>
      <lines>2</lines>
    </message>
  </context>
</TS>

```

<lng> stands for the language code.

23.2 Using interactive or silent installation versions

General Information

If you install SINUMERIK Operate using PC/PCU, then the execution is realized using an MSI setup. The following installation versions are available, which you can use in the interactive or silent mode using the appropriate Windows command line:

- SINUMERIK Operate complete installation
- SINUMERIK Operate base installation with auxiliary files
- SINUMERIK Operate base installation without auxiliary files

Execution

You control setup execution using command line arguments. The interactive mode `/v` allows you to transfer parameter strings to setup.exe. If you wish to run setup without user interaction, then you can extend the parameter string with the silent mode `/qn`.

Place the parameter string in inverted commas. If you use several parameters in the string, separate these using spaces.

If you do not specify any arguments when calling setup.exe, then the complete installation is run in the interactive mode.

Note

Do not put any spaces between `/v` and the parameter string.

Note

If you want a progress display in the silent mode, then instead of specifying `/qn` you can also specify `/qbn`.

Interactive installation versions

<code>setup.exe</code>	Complete installation in the interactive mode
<code>setup.exe /v"BASEONLY=1"</code>	Base installation in the interactive mode with help files
<code>setup.exe /v"BASEONLY=1 HELPFILES=0"</code>	Base installation in the interactive mode without help files

Silent installation versions

<code>setup.exe /v"/qn</code> <code>setup.exe /s /v"/qn</code>	Complete installation in the silent mode (<code>/s</code> suppresses all of the setup messages)
<code>setup.exe /v"/qn BASEONLY=1</code> <code>setup.exe /s /v"/qn BASEONLY=1</code>	Base installation in the silent mode with help files (<code>/s</code> suppresses all of the setup messages)
<code>setup.exe /v"/qn BASEONLY=1 HELPFILES=0</code> <code>setup.exe /s /v"/qn BASEONLY=1 HELPFILES=0</code>	Base installation in the silent mode without help files (<code>/s</code> suppresses all of the setup messages)

Additional parameter assignments

Installation path

If you install the PC version of SINUMERIK Operate, then you can additionally specify the installation path:

```
setup.exe /s /v"/qn INSTALLDIR=C:\Programme\testdir"
```

For spaces in the path:

```
setup.exe /s /v"/qn INSTALLDIR="C:\Program Files\testdir\""
```

Exit code

You can identify a successful installation or errors based on the exit code of the setup.exe call:

Exit code == 0	No error
Exit code <> 0	Error

Log file

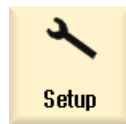
During the installation you can generate a log file with /L:

setup.exe /s /v"/qb! /L*vx log.txt"

23.3 Setting the IP address of the NCU

From the user interface of the SINUMERIK Operate, you can set the IP address of the NCU with which it should connect.

Procedure



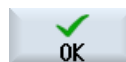
1. Select the "Start-up" operating area.



2. Press the "HMI" softkey and the ">>" softkey.



Press the vertical softkey "NCU connection".
The "NCU Connection" window opens.
Enter the IP address of the NCU into the input field.



3. Press the "OK" softkey to confirm the data.

23.4 SINUMERIK Operate exiting

Procedure



1. Press the <MENU SELECT> key on the operator panel.

OR:

Activate the <M> symbol on the touch panel.



External keyboard: Press the <F10> key.



2. Press the menu forward key.

External keyboard: Press the <Shift> + <F9> keys.



3. Press the "EXIT" softkey.

External keyboard: Press <F8>

- on the PCU to terminate SINUMERIK Operate and the Windows system.
- If you started SINUMERIK Operate operating software on the PCU in the service mode, then only the operating software is exited.
- On the PC itself, only SINUMERIK Operate is exited.

24.1 Configuring the traversing keys

The traversing keys of the HT 8 are not labeled as their type of action is not predefined. The labeling of the traversing keys should adapt dynamically to the type of action of the keys. To recognize the type of action of the traversing keys, they are shown within the display in the vertical softkey area.

The menu of the traversing keys consists of two vertical rows each with eight keys so that up to 16 texts can be configured. The top and bottom traversing key row remain empty and can be assigned other functions.

The following data can be displayed:

- Machine axis name
- Alias name for machine axis
- Any language-dependent text
- Symbol

The following user-specific files are required. You can use the sample files as template:

File	Meaning
"sljkconfig.ini" configuration file	File in which the traversing keys are configured.
Text file "oem_sljk_XXX.ts"	File for labeling traversing keys in a foreign language, XXX = language code

"sljkconfig.ini" configuration file

Entries	Meaning
[State_1]	Labeling type - changed via the PLC.
ParamText_x_y	Text of the traversing key that is made up of parameters. Two-line labeling is possible, whereby max. five characters are possible per line. x: Specifies the position of the key within the row (2 to 7). y: Specifies the key row (1 or 2).
	%m1 The machine axis name of the first axis is referenced and displayed as text. The current active axis number is read out from data block DB10. The name from the machine data is determined via this index.
	%n Defines the position in the text for the line break.
	%a1 The alias axis name of the first axis is referenced and displayed as text. The current active axis number is read out from data block DB10. The name from the "oem_sljk_eng.ts" text file is determined via this index.

Entries	Meaning
TextId_x_y	Text of the traversing key that is read from the text file (target language text).
Picture_x_y	File name of the icon to be displayed. The files with the icons must be located in the following directories: /oem/sinumerik/hmi/ico and depending on the resolution of the subdirectories: /ico640 /ico800 /ico1024 /ico1280

Procedure

1. Copy the "sljkconfig.ini" configuration file from the folder **/siemens/sinumerik/hmi/template/cfg**
2. Place the copy into the folder **/oem/sinumerik/hmi/cfg** or **/user/sinumerik/hmi/cfg**
3. Open the copy in the Editor and define the traversing key labeling.

Example of a "sljkconfig.ini" configuration file

12 axes are defined in the example, with text for softkey 1 and softkey 2, as well as an image on softkey 3 and softkey 4.

```
[Settings]
```

```
FileType = INI
```

```
; A few examples for further configuration options
```

```
; Alias names of machine-axes (%a1, %a2, etc.), from oem_sljk_deu.ts  
(example)
```

```
[State_1]
```

```
TextId_1_1 = OEM_JK_TEXT_1
```

```
TextId_1_2 = OEM_JK_TEXT_2
```

```
ParamText_2_1 = %a1%n-
```

```
ParamText_2_2 = %a1%n+
```

```
ParamText_3_1 = %a2%n-
```

```
ParamText_3_2 = %a2%n+
```

```
ParamText_4_1 = %a3%n-
```

```
ParamText_4_2 = %a3%n+
```

```
ParamText_5_1 = %a4%n-
```

```
ParamText_5_2 = %a4%n+
```



```

ParamText_6_1 = %a5%n-
ParamText_6_2 = %a5%n+
ParamText_7_1 = %a6%n-
ParamText_7_2 = %a6%n+
Picture_8_1 = AlarmCancel.png
Picture_8_2 = AlarmNCReset.png

```

"oem_sljk_eng.ts" text file

Entries	Meaning
name	Freely selectable name of the text context. In the text file template, the name of the text context is "SIJkLabels" and stands for the traversing key labeling (solution line jog key labels). This identifier is already stored in the configuration file.
source	Traversing key identifier of the respective axis. This text ID is referenced in the "sljkconfig.ini" configuration file with the "TextId_2_1". The text IDs for the alias names (JK_AXISNAME_2 to JK_AXISNAME_7) must not be changed.
translation	Input of the foreign language text for the axis specified in <source>.

Language code in file names "xxx"

Language	Language code
German	deu
English	eng
French	fra
Spanish	esp
Italian	ita
Chinese	chs

Procedure

1. You can copy the sample file "oem_sljk_deu.ts" from the following directory: **/siemens/sinumerik/hmi/template/Ing**.
2. Store or create the file in the **/oem/sinumerik/hmi/Ing** or **/user/sinumerik/hmi/Ing** directory.
3. Give the file a name, e.g. for German texts: "sljk_deu.ts".
If you wish to set-up the labeling for additional languages, a separate file must be created for each language. Save the file with the appropriate language code in the file name. To do this, use the language codes specified above.

4. Open the file and in the <message> and </message> area, define the labeling.
5. Restart the HMI.
In order that the labeling is displayed during the program runtime, the file must be converted into a binary format. This conversion is only executed when the HMI powers up.

Example of a text file "sljk_deu.ts"

12 axes are defined in the example, with text for softkey 1 (SF1) and softkey 2 (SF2):

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE TS>
<TS>
  <context>
    <name>SlJkLabels</name>
    <!-- Alias names for machine axis (%a1, %a2, etc.) -->
    <!-- Do not change the text ID (JK_AXISNAME_1, etc.) -->
    <message>
      <source>JK_AXISNAME_1</source>
      <translation>X</translation>
    </message>
    <message>
      <source>JK_AXISNAME_2</source>
      <translation>Y</translation>
    </message>
    <message>
      <source>JK_AXISNAME_3</source>
      <translation>Z</translation>
    </message>
    <message>
      <source>JK_AXISNAME_4</source>
      <translation>A</translation>
    </message>
    <message>
      <source>JK_AXISNAME_5</source>
      <translation>B</translation>
    </message>
    <message>
      <source>JK_AXISNAME_6</source>
      <translation>C</translation>
    </message>
    <message>
      <source>JK_AXISNAME_7</source>
      <translation>U</translation>
    </message>
```

```

<message>
  <source>JK_AXISNAME_8</source>
  <translation>V</translation>
</message>
<message>
  <source>JK_AXISNAME_9</source>
  <translation>W</translation>
</message>
<message>
  <source>JK_AXISNAME_10</source>
  <translation>UV1</translation>
</message>
<message>
  <source>JK_AXISNAME_11</source>
  <translation>UV2</translation>
</message>
<message>
  <source>JK_AXISNAME_12</source>
  <translation>UV3</translation>
</message>
<!-- User defined language dependent text (example) -->
<message>
  <source>OEM_JK_TEXT_1</source>
  <translation>SF1</translation>
</message>
<message>
  <source>OEM_JK_TEXT_2</source>
  <translation>SF2</translation>
</message>
</TS>

```

See also

Supported languages (Page 584)

24.2 Configuring user-specific key labeling

Labeling the keys for the HT 8

The texts of the CPF menu (CPF: Control Panel Function) for the HT 8 can be labeled with your own texts in the particular language of the country.

The texts are created in the "slck_XXX.ts" file.

"XXX" = language code for the corresponding language-specific labeling.

You can create and edit this file using the HMI or also externally on a PC.

Note

If you wish to create or edit the file on a PC, use an editor that supports UTF-8 coding.

Language code in file names "XXX"

Language	Language code
German	deu
English	eng
French	fra
Spanish	esp
Italian	ita
Chinese	chs

Creating key labeling

Tag	Meaning
source	Designation for the user softkey. "SK_USERKEY1" to "SK_USERKEY16" are possible, whereby the names may not be changed.
comment	User-specific description of the key assignment.
translation	Text that should be shown on the key. - A maximum of 10 characters are possible per line. 2-line labeling is possible, whereby the line break is set using "%n".
remark	Remark for key assignment.
chars	Number of characters. A maximum of 10 characters per line are possible.
lines	Number of lines. 2 lines are possible.

Procedure

1. You can copy the sample file "oem_slck_deu.ts" from the following directory: **/siemens/sinumerik/hmi/template/lng**.
2. Store or create the file in the **/oem/sinumerik/hmi/lng** or **/user/sinumerik/hmi/lng** directory.
3. Give the file a name, e.g. for German texts: "slck_deu.ts".
If you wish to create the key labeling for additional languages, then a separate file must be created for each language. Save the file with the appropriate language code in the file name. To do this, use the language codes specified above.

4. Open the file and in the <message> and </message> area, define the key labeling.
5. Restart the HMI.
In order that the key labeling is displayed during the program runtime, the file must be converted into a binary format. This conversion is only executed when the HMI powers up.

Example of a key label

```
<!DOCTYPE TS><TS>
<context>
  <name>SlCkDialog</name>
  <message>
    <source>SK_USERKEY1</source>
    <comment></comment>
    <translation>U1</translation>
    <remark>User key 1</remark>
    <chars>10</chars>
    <lines>2</lines>
    <languageIndependent>true</languageIndependent>
  </message>
  <message>
    <source>SK_USERKEY2</source>
    <comment></comment>
    <translation>U2</translation>
    <remark>User key 2</remark>
    <chars>10</chars>
    <lines>2</lines>
    <languageIndependent>true</languageIndependent>
  </message>
  <message>
    .....
  </message>
</context>
</TS>
```

See also

Range of alarms (Page 222)

24.3 Configuring the function display at user-specific keys (U keys)

Function

Active functions can be displayed at the configurable user keys via the PLC. For instance, small LEDs can be emulated on the softkeys.

You configure the function in the "slckcpf.ini" file.

Interface signals

The PLC bits are in the output image of the PLC-HT 8 interface and are analog to those in the input image.

Signals to the MCP1 (or MCP2) Interface PLC → HT 8								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AB n + 1		U4	U3		U2	U1		
AB n + 4	U9	U10	U11	U12	U13	U14	U15	U16
AB n + 5		U8	U7	U6	U5			

Adapt the display in the "slckcpf.ini" file

Section	Description
UserKeyLEDIcon	Name of the icon file. Standard entry: led_green.png
	OFF Deactivates the icon display.
	PRESSED Displays the active functions by pressing down the softkey.
UserKeyLEDIconAlignment	Specifies the position of the icon. Standard position: AlignLeft AlignTop Horizontal and vertical alignments can be combined. The two names are separated by the " " character. The following alignments are possible:
	AlignLeft Left
	AlignRight Right
	AlignHCenter Horizontal, center
	AlignTop Top
	AlignBottom Bottom
	AlignVCenter Vertical, center
UserKeyLEDMap	Specifies the start address of the output image. Entries can be made in the following form: "DBx.DBBy", "ABx", "MBx". Default setting: The start address is determined using DB7 MCP1Out (or MCP2Out).

24.3 Configuring the function display at user-specific keys (U keys)

Section	Description	
U1LED ... U16LED	The status bit address can be defined - different from the output image - using these entries. Note: The offsets from the HT 8 output image, for one or several softkeys, are not taken into consideration.	
VarIncLED		
SBLLED		
WCSLED		
showVarIncLED	true	When the increment mode is active, the icon is also displayed on the "[VAR]" softkey.
	false	The icon is not displayed.
showSBLLED	true	When the SingleBlock mode is active, then the icon is also displayed on the "Single Block" softkey.
	false	The icon is not displayed.
showWCSLED	true	When the SingleBlock mode is active, then the icon is also displayed on the "Single Block" softkey.
	false	The icon is not displayed.

1. You can copy the sample file "slckcpf.ini" from the following directory: **/siemens/sinumerik/hmi/template/cfg**
2. Store the file in the **/oem/sinumerik/hmi/cfg** or **/user/sinumerik/hmi/cfg** directory.
3. If you use your own icon, then locate it together with the corresponding resolution for HT 8 directory: **/oem/sinumerik/hmi/ico/ico640** or **/user/sinumerik/hmi/ico/ico640**.
4. Open the file and make the appropriate settings.

Sample file "slckcpf.ini"

```

Template for the configuration of the
; HT8 control panel function menu
;
; To activate the settings remove the
; commentary ';' at the beginning of the line
;; Display settings of the user key softkey leds
[UserKeyLED]
; Filename of the LED icon
;UserKeyLEDIcon = led_green.png
; Alignment of the LED icon
;UserKeyLEDIconAlignment= AlignLeft | AlignTop

; Use following led map start address instead of calculating
DB7.MCP1Out
;UserKeyLEDMap = AB0

```

24.3 Configuring the function display at user-specific keys (U keys)

```
; Use the following settings to use this status bits instead of the
led map for a specific sk
;U1LED=/channel/parameter/R[U1,1]
;....
;U16LED=/channel/parameter/R[U1,16]
;VarIncLED = DB11.DBX8.5
;SBLLED = DB21.DBX0.4
;WCSLED = DB19.DBX0.7

; Show a LED for the var inc sk
;showVarIncLED = true
; Show a LED for the var single block sk
;showSBLLED = true
; Show a LED for the var wcs/mcs sk
;showWCSLED = true
```


List of abbreviations/acronyms

Abbreviation	Meaning
ASCII	American Standard Code for Information Interchange: American coding standard for the exchange of information
BAG	Mode group
OPI	Operator Panel Interface
CNC	Computerized Numerical Control: Computerized numerical control
CEC	Cross Error Compensation: Sag compensation
CP	Communication Processor
DCP	Discovery and basic Configuration Protocol: ProfiNet protocol
DHCP	Dynamic Host Configuration Protocol
DIN	Deutsche Industrie Norm
DNS	Domain Name System
DIR	Directory: Directory
DRAM	Dynamic Random Access Memory
EES	Execution from External Storage
ERTEC	Enhanced Real Time Ethernet Controller
ESR	Extended Stop and Retract
ETC	ETC key ">"; Softkey bar extension in the same menu
FRAME	Data block (FRAME)
FIFO	First In - First Out: Method of storing/retrieving data in/from a memory.
GDIR	Global Directory: Global part program memory
GUD	Global User Data: Global user data
MSD	Main Spindle Drive
HSC	High Speed Cutting: High-speed machining
INC	Increment: Incremental dimension
INI	Initializing Data: Initializing data
IP	Internet Protocol
IPO	Interpolator
ISO	International Standard Organization
JOG	Jogging: Setup mode
LED	Light Emitting Diode: Light emitting diode
LLDP	Link Layer Discovery Protocol
LUD	Local User Data: Local user data
MAC	Media Access Control
MB	Megabyte
MD	Machine data
MDA	Manual Data Automatic: Manual input
MCS	Machine Coordinate System
MLFB	Maschinenlesbare Fabrikatbezeichnung

Abbreviation	Meaning
MPF	Main Program File: NC part program (main program)
MCP	Machine Control Panel: Machine control panel
NC	Numerical Control: Numerical control
NCK	Numerical Control Kernel: NC kernel with block preparation, traversing range, etc.
NCU	Numerical Control Unit: NCK hardware unit
NFS	Network File System
NTP	Network Time Protocol
ZO	Zero offset
OEM	Original Equipment Manufacturer
OID	Object Identifier: Information identifier
OP	Operator Panel
PCU	Programmable Control Unit
PI service	Program Invocation Services
PG	Programming device
PLC	Programmable Logic Control
RFC	Request for Comments
REF	REFerence point approach function
REPOS	REPOSition function
ROV	Rapid Override: Input correction
RPA	R-Parameter Active: Memory area in NCK for R- NCK for R parameter numbers
SBL	Single Block: Single block
SD	Setting Data
SDB	System Data Block
SEA	Setting Data Active: Identifier (file type) for setting data
SK	Softkey
SMB	Server Message Block
SNMP	Simple Network Management Protocol
SPF	Sub Program File: Subroutines
SRAM	Static Random Access Memory: Static RAM (non-volatile)
SSH	Secure Shell: Network protocol
SW	Software
SYF	System Files: System files
TCU	Thin Client Unit
TEA	Testing Data Active: Identifier for machine data
TMA	Tool Magazine Active: Magazine data
TO	Tool Offset: Tool offset
TOA	Tool Offset Active: Identifier (file type) for tool offsets
UDP	User Datagram Protocol: Network protocol
UFR	User frame
VNC	Virtual Network Computing
FD	Feed Drive
WCS	Workpiece Coordinate System
TM	Tool Management

Index

3

- 3D file
 - Protection area elements, 262
- 3D simulation, 101, 105

A

- Access levels, 30
- Access right
 - Effect, 33
 - Global, 33
 - individual, 32
 - Remote access, 33
- Action log
 - actlog.ini, 366
 - Log file, 364
 - Open log file, 362
 - Saving log file, 364
 - Search for events, 363
 - Setting the log size, 366
 - Settings, 360
- action.com, 364
- action.log, 364
- Activate keyboard, 28
- actlog.ini, 366
- Actual value window
 - Setting font size, 83
- actual_actionlog.com, 360
- actual_crashlog.com, 360
- Adapt tool measurement
 - Examples when milling, 570
 - for turning, 575
 - when milling, 568
- Adapt workpiece measurement, 564
- Additional language
 - installing, 583
 - Uninstalling, 584
- Alarm log
 - Filter properties, 227
 - Settings, 225
- Alarm number ranges, 222
- Alarm texts
 - Changing colors, 214
 - Create file, 206
 - Creating, 203, 206
 - Creating foreign-language texts, 205

- Deactivating a warning, 235
- Registering the file, 207
- Replacing, 218
- Replacing standard alarm texts, 218
- Select font color, 203
- Standard parameter specification, 223
- Supported languages, 584
- alarmtext_conversion.log, 207, 210, 213, 221, 224
- Analog spindle, 112
- Archive
 - Creating, 280
 - Reading in, 281
- AS-i
 - Components, 400
 - Configure diagnostics, 402
 - Diagnostics, 400
- Axis diagnostics, 335
- Axis-specific machine data, 61
- Axis-specific setting data, 64

B

- Backing up setup data, 289
- Box
 - Protection area element, 257

C

- Channel menu
 - Configuring, 51
 - Configuring operator stations, 55
 - Configuring the total structure, 53
 - Defining channel groups, 53
- Channel operational message, 85
 - Standard configuration, 86
- Channel-specific machine data, 61
- Channel-specific setting data, 64
- Checksum, 435
- Circular position pattern, 458
- Circumferential groove, 458
- Clamped state, 117
- Clamping
 - Setting up function, 107
- Collision avoidance, 237
 - Changing the section, 243
 - Creating a machine model, 240
 - Creating a tool protection area, 274
 - Creating collision pairs, 275

- Creating kinematic elements, 269
- Creating machine protection areas, 271
- Enable function, 239
- Fundamentals example, 266
- Parameter assignment, 239
- Collision element
 - Creating, 251
- Collision pair
 - Creating, 264
- Comment input
 - ini file, 23
 - XML file, 22
- Commissioning archive
 - Creating, 289
- Commissioning the drive system, 119
- config.ini, 55
- Configuring OEM icons, 99
- Configuring OEM texts, 98
- Configuring operator stations, 55
- Configuring the tool change reason, 170
- Contour grooving, 477
- Contour milling, 457
- Contour turning, 477
- Coolant
 - Assigning, 167
 - Identifier, 179
- Coordinate system
 - MD52000, 444
- crash.com, 364
- crash.log, 364
- Creating
 - Collision pair, 264
- Creating a machine model
 - Collision avoidance, 240
- Creating online help
 - Creating a directory, 416
 - HTML_Creating file, 413
 - Replacing the standard online help, 415
 - XML_Creating file, 415
- CSV file, 356
- CUST_800.SPF, 450
- CUST_832.SPF, 450, 545
- CUST_CLAMP, 450
- CUST_M6, 450, 452
- CUST_MEACYC
 - Structogram, 551
- CUST_MEACYC.SPF, 450
- CUST_MEAPROT.SPF, 450, 578
- CUST_MULTICHAN, 450, 456
- CUST_T, 450, 452
- CUST_TEHCYC.SPF, 450
 - Aligning, 454

- Cutting edge parameters
 - Identifiers, 140
 - OEM identifier, 178
- Cycle alarms, 501
- CYCLE495, 493
- CYCLE63, 457
- CYCLE79, 458
- CYCLE800
 - Structogram, 523
- CYCLE832, 542
- CYCLE84, 449
- CYCLE840, 449
- CYCLE930, 477
- CYCLE950, 477
- CYCLE951, 477
- CYCLE952, 477
- CYCLE99, 477
- CYCLE996, 527
- Cycles version
 - Define, 581
 - Displaying, 580
- Cylinder
 - Protection area elements, 260
- Cylinder surface transformation
 - Milling, examples, 462
 - under milling, 461
 - under ShopTurn, 484
 - under turning, 482
 - with groove side offset, 484
 - without groove side offset, 483
- Cylindrical grinding, 492
 - Displaying softkeys, 444

D

- Data backup, 277
- Date
 - Selecting the format, 26
 - Setting, 27
- Deactivate keyboard, 28
- dealer.xml, 351
- Defining channel groups, 53
- DHCP server detection, 318
- Diagnostics
 - AS-i, 402
 - Ethernet/PROFINET network, 318
 - Fail-safe modules, 437
 - HMI trace, 367
- Direction of rotation
 - Counterspindle, 474
 - Main spindle, 468
- Directory structure, 17

- Display filter
 - Machine data, 68
 - Display groups, 69
 - Display machine data, 64
 - Search, 72
 - Displaying AS-i, 402
 - Drilling, 448
 - Drive
 - Execution from External Storage, 49
 - Global part program memory, 50
 - Logical drive, 47
 - Set EXTCALL, 48
 - Setting up, 47
 - Drive parameters
 - Display/change, 66
 - for machine data, 61
 - Drive system
 - Diagnostics, 371
 - Diagnostics - details, 372
 - Drive-specific machine data, 61
- E**
- Editing an xml file, 22
 - Editing enum values, 71
 - Editing the tool and magazine management, 192
 - Entering BICO values, 71
 - Error analysis, 306
 - Error file of the alarms, 224
 - Ethernet/PROFINET
 - Diagnostics, 318
 - Displaying messages, 319
 - Network trace, 320
 - Trace parameters, 321
 - EUNA, 348
 - EXIT, 594
 - Displaying the softkey, 29
 - Expert parameters, 69
 - EXTCALL
 - switch off and on again, 48
 - Extended reset, 67
- F**
- Face machining
 - under ShopTurn, 488
 - under turning, 485
 - Fail-safe I/O modules, 437
 - fast block search, 92
 - File
 - Copy, 20
 - Copying with WinSCP, 23
 - Cut, 21
 - Delete, 21
 - Displaying in preview window, 22
 - External editing, 22
 - insert, 20
 - Open, 20
 - Protection area elements, 262
 - Renaming, 21
 - Form-truing, 493
 - Frame
 - Protection area element, 255
 - Full temperature protection, 116
- G**
- G code groups
 - Configuring the display, 84
 - G code program, 191
 - G group 59, 544
 - General machine data, 61
 - General setting data, 64
 - Generating a crashlog file, 361
 - Generating an archive for the Hotline, 285
 - Grinding parameters
 - Identifiers, 143
- H**
- Hardware components
 - Adding, 356
 - Entering, 355
 - HEAD_1, 512
 - HEAD_2, 513
 - High-Speed Settings, 542
 - Hirth gearing, 497
 - Hirth joint, 537
 - HMI PRO RT, 398
 - HMI restart
 - Displaying the softkey, 29
 - HMI trace, 367
 - HOLES2, 458
 - HT 8
 - Activating, deactivating the virtual keyboard, 28
 - Configuring the traversing keys, 595
 - Key labeling, 599
 - LED display on user keys, 602
 - oem_sljk_deu.ts, 597
 - slguiconfig.ini, 28
 - sljkconfig.ini, 595

I

Identifier

- Coolant, 179
- Cutting edge parameters, 140
- Grinding parameters, 143
- Magazine location adapter parameters, 145
- Magazine location parameters, 144
- Monitoring parameters, 142
- Multitool location parameter, 146
- Multitool parameters, 145
- OEM cutting edge parameter, 178
- OEM magazine location parameters, 179
- OEM monitoring parameters, 178
- OEM multitool parameters, 178
- OEM tool parameters, 178
- Tool parameters, 138
- Tool-specific functions, 179

identSNAPSHOT, 349

Inclined Y axis

- under ShopTurn, 489
- under turning, 488

Index text file for alarms

- Create file, 208
- Creating texts, 208
- Registering, 209

Inserting a run up screen, 30

Installation

- Interactive, 592
- silent, 592

K

Keyboard

- Setting, 28

Kinematic chain element

- Creating, 247

Kinematic elements

- Kinematic rotation, 248
- Linear axis, 249
- Offset, 250
- Parallel kinematic rotation, 248
- Parallel linear axis, 249
- Parallel offset, 250
- Parallel rotary axis, 249
- Rotary axis, 249

Kinematic rotation

- Kinematic element, 248

Kinematics

- Check list, 501

Examples, commissioning, 512

measured, 527

with Hirth gearing, 497

L

Language

- Defining the selection mode, 25

Language code, 584, 585

License key

- Entering, 45
- Reading in, 45

License requirement

- Exporting, 45

Licenses, 41

Linear axis

- Kinematic elements, 249

List parameters

- Changing, 147
- Creating new, 149

Log file

- Action log, 364

Logbook, 357

- Documenting startups, 358
- Entry search, 360
- Importing selection files, 353
- Machine identity, 349
- Making an entry, 359
- Output, 354

Logging the measurements

- Settings for AUTOMATIC, 577

Logging the measuring

- Settings for JOG, 577

M

ma_types.xml, 352

Machine data

- Axis selection, 63
- Configuring plain texts, 80
- Display filter, 68
- Displaying, 61
- Editing enum values, 71
- Editing hexadecimal values, 70
- Entering BICO values, 71
- Information, 62
- Overview, 61
- Search, 72

Machine function, defining texts, 168

Machine identity, 349

Machine model, 237

Machine operating area, 83
Machine protection area, 253
Machine-specific information
 Creating a selection file, 350
 Entering, 349
 Save, 354
Magazine location adapter parameters
 Identifiers, 145
Magazine location parameters
 Identifiers, 144
 OEM identifier, 179
Managing data, 74
Manual machine
 Activating the function, 94
 Parameter assignment, 94
Message texts
 Creating, 211
 creating for several channels, 212
 Part program, 210
Milling
 Cylinder surface transformation, 461
 Displaying softkeys, 443
 Examples, setting-up a milling machine, 462
 Settings under ShopMill, 459
 Tool measurement, 556
 Turning on milling machines, 529
 Workpiece measurement, 554, 567
MIXED_45, 516
Monitoring parameters
 Identifiers, 142
 OEM identifier, 178
Multiple clamping of different workpieces (option), 465
Multiple edge, 458
Multitool location parameter
 Identifiers, 146
Multitool parameters
 Identifiers, 145
 OEM identifier, 178

N

NC/PLC variables
 Changing, 329
 Displaying, 327
NCU connection
 setting - under PCU, 593
 setting - under Windows, 593
netnames.ini, 54
Network
 Configuring, 293

Network adapter, 304
 Configuring, 314
Network diagnostics, 303
Number ranges of alarms, 222

O

oem.xml, 352
oem_ae_database_conversion.log, 217
oem_alarms_eng.ts, 206
oem_indextxts_eng.ts, 208
oem_slaedatabase.xml, 214
oem_slaesvcadapconf.xml, 207
oem_slck_deu.ts, 600
oem_sljk_deu.ts, 597
oem_text_conversion.log, 207, 224
oemsubs.xml, 352
Offset
 Kinematic elements, 250
Operating state, 331
Operator panels
 Configuring, 300
Orientable toolholder, 576

P

Parallel box
 Protection area element, 257
Parallel cylinder
 Protection area elements, 260
Parallel frame
 Protection area element, 255
Parallel linear axis
 Kinematic elements, 249
Parallel offset
 Kinematic elements, 250
Parallel sphere
 Protection area elements, 259
Password
 Changing, 31
 Delete, 31
 Set, 31
Performing a start-up
 Reading in original status, 284
PLC hardware upgrade archive
 Creating, 281
Probe
 Example of a test program, 547
 Testing the function, 547
 to measure the tool, 546
 to measure the workpiece, 546

PROFIBUS

- Connections, 368
- DP slaves, 370

PROFIBUS/PROFINET diagnostics, 398**PROG_EVENT.SPF, 368, 450**

- Extend, 452

Program list

- Configuring a softkey, 37
- Creating, 38

Program test

- activate, 89
- deactivate, 89

Programming G code programs, 94**Protection area elements**

- 3D file, 262
- Box, 257
- Creating, 254
- Cylinder, 260
- File, 262
- Frame, 255
- Parallel box, 257
- Parallel cylinder, 260
- Parallel frame, 255
- Parallel sphere, 259
- Sphere, 259

R**RCS Commander, 16****Remote diagnostics**

- Adapting, 374

Replacing standard alarm texts

- Create file, 218
- Creating texts, 218
- Registering the alarm text file, 219

Replacing the logo, 83**Reset**

- Extended, 67

Rotary axis

- Kinematic elements, 249

Rotary axis vectors

- V1, 503
- V2, 503

S**Safety Integrated**

- Cam signals, 438
- Checksums, 435
- Communication data, 439
- Receive connections, 441

Safe programmable logic, 434**Safety-related inputs and outputs, 433****Send connections, 439****Status display, 432****SBC Safe Brake Control, 436****Screen**

- darken screen, 29
- saver, 29
- Time until the screen is darkened, 29

Screenshots

- Copy, 348
- Creating, 348
- Open, 348

Search

- Display machine data, 72
- Machine data, 72
- Setting data, 72

Search mode

- Configuring, 91

Searching

- Logbook entry, 360

Service drive

- Display, 339
- Parameter description, 340

Service overview, 331

- Axes and drives, 333

Setting data

- Axis selection, 66
- Channel selection, 66
- Displaying, 64
- Overview, 61
- Search, 72

Setting up a counterspindle

- ShopTurn, 473

Setting-up a lathe with milling tools, 481**Setting-up measuring in JOG, 552****SGE/SGA signals, 433****ShopMill**

- Settings for milling, 459
- Tool-specific functions, 167

ShopTurn

- Cylinder surface transformation, 484
- Face machining, 488
- Inclined Y axis, 489
- Setting up a counterspindle, 473
- Settings for turning, 468

Simulation

- Cycle application, 451
- Deactivate, 104
- General conditions, 101
- Grinding technology, 103

- Milling technology, 103
- Turning technology, 103
- Simultaneous recording, 105
 - Activating the machining time, 106
 - Option, 105
- slaesvcadapconf.xml, 207, 209, 211, 219, 235
- slaesvcconf.xml, 212, 221
- slckcpf.ini, 603
- slhlp.xml, 411
- slmagcodeconfig.ini, 84
- SLOT2, 458
- sltlprodiaghmi.ini, 400, 401
- Softkey
 - Access level, 36
 - Original setting, 36
- Source ID, 222, 223
- Source URL, 222, 223
- Specify the IP address of the NCU, 593
- Sphere
 - Protection area elements, 259
- Spindle analog, 112
- Spindle control, 111
- Spindle diagnostics, 114
- Spindle load, 116
- SS1 Safe Stop 1, 436
- Standard parameter specifications of alarm texts, 223
- Starting SINUMERIK Operate
 - NCU, 15
 - PCU, 15
 - under Windows, 15
- Start-up archive
 - Creating, 278
 - Reading in, 281
- Station-related network diagnostics, 303
- Status display
 - Inserting a logo, 83
- Status display with icons, 96
- STO Safe Torque Off, 436
- Stock removal, corner, 477
- Surface grinding, 492
 - Displaying softkeys, 444
- Swiveling
 - activate, 494
 - Configuring input screens, 494
 - Creating angled tools, 496
 - Declare data sets, 496
 - Offset vectors, 504
 - Retraction, 511
 - Rotary axes, 508
 - Set up kinematic chain, 502
- System Network Center, 55

System utilization, 347

T

- TABLE_45, 515
- TABLE_5, 518
- Tailstock
 - Option, 105
- Tapping, 449
- TEACH IN, 90
- Technology
 - Cylindrical grinding, 492
 - Drilling, 448
 - Milling, 457
 - Surface grinding, 492
 - Swiveling, 493
 - Turning, 477
- Template files, 20
- Test license, 44
- TextPad, 16
- Thread-cutting, 477
- Time
 - Selecting the format, 27
 - Setting, 27
- Tool clamping angle, 537
- Tool list
 - List views, 134
- Tool management
 - Enable functions, 124
 - Graphical tool/magazine display, 125
 - Grinding technology, 128
 - Manual tool, 126
 - MD settings, 121
 - MD settings for diameter / cutting edge radius, 124
 - Milling technology, 128
 - PLC, TRANSLINE 2000, 128
 - Tool wear, 126
 - Turning technology, 128
 - With magazine management, 122
 - Without magazine management, 121
- Tool Management
 - Assignments, 200
 - Buffer locations, 197
 - Editing the management, 194
 - Loading locations, 198
 - Magazine configuration, 199
 - Opening a function, 193
 - Tool unit, 196
- Tool measurement
 - under milling, 556
 - under turning, 561

- Tool parameters
 - Identifiers, 138
 - OEM identifier, 178
- Tool types
 - Drill, 151
 - Grinding tools, 151
 - Milling tool, 151
 - Special tools, 152
 - Turning tools, 152
- Toolholder, 190
- Tool-specific functions
 - Identifier, 179
- Topologies, 323
- TRAANG, 490
 - under ShopTurn, 489
 - under turning, 488
- Trace
 - Changing the scale, 394
 - Creating a session, 381
 - Cursor A/B positioning, 396
 - Data acquisition, 389
 - Data storage on CU, 390
 - Define zoom area, 396
 - Display peak-to-peak value, 397
 - Display peak-to-valley value, 397
 - Displaying variable details, 388
 - Editing a session, 385
 - Editing graphic display, 392
 - Filter/search, 383
 - Insert variable, 384
 - Load session, 383
 - Overview, 379
 - Replace variable, 384
 - Save options, 391
 - Save session, 381
 - Selecting a variable, 384, 393
 - Sequence, 380
 - Session, 384
 - Setting grid lines, 391
 - Settings (drive), 389
 - Settings (PLC, NC, servo), 389
 - Start recording, 391
 - Starting a session, 383
 - Stop recording, 391
 - Variable attributes, 385
 - Zooming characteristic graphs, 395
- TRACON, 490
- TRACYL, 491
 - under ShopTurn, 484
 - under turning, 482
- Transformation
 - Settings for milling, 463

- TRANSMIT
 - under ShopTurn, 488
 - under turning, 485
 - with Y axis, 486
- TRANSMIT command, 191
- Turning
 - Cylinder surface transformation, 482
 - Displaying softkeys, 443
 - Examples, setting-up a lathe, 481
 - Face machining, 485
 - Inclined Y axis, 488
 - Settings under ShopTurn, 468
 - Technology cycles, 477
 - Tool measurement, 561
 - Workpiece measurement, 574
- Turning on milling machines
 - Behavior at power on, 536
 - Behavior at reset, 536
 - Block search, 542
 - CUST_800, 539
 - Hirth joint, 537
 - Kinematics transformation, 534
 - Setting the coordinate system, 538
 - Setting up, 529
 - Tool clamping angle, 537

U

- USB FlashDrive, 16
- USB keyboard, 16
- User view
 - Creating, 77
 - Delete, 80
 - Editing, 78
 - Importing, 76
- user.xml, 352
- Using angular tools, 496

V

- Variable screen forms, 330

W

- WinSCP, 16, 23
- Workpiece measurement
 - under milling, 554, 567
 - under turning, 574

SINUMERIK

SINUMERIK 840D sl/828D Easy XML

Programming Manual

Valid for:

CNC Software Version 4.7 SP1

01/2015

6FC5397-3DP40-5BA2

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

DANGER

indicates that death or severe personal injury will result if proper precautions are not taken.

WARNING

indicates that death or severe personal injury may result if proper precautions are not taken.

CAUTION

indicates that minor personal injury can result if proper precautions are not taken.

NOTICE

indicates that property damage can result if proper precautions are not taken.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

WARNING

Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Table of contents

1	Generating user dialogs.....	5
1.1	Scope of functions.....	5
1.2	Fundamentals of Configuration.....	7
1.3	Configuration files.....	9
1.4	Structure of configuration file.....	13
1.5	Language dependency.....	15
1.6	XML identifier.....	15
1.6.1	General structure.....	15
1.6.2	Instruction/identifier descriptions.....	16
1.6.3	Color coding.....	40
1.6.4	Special XML syntax.....	40
1.6.5	Operators.....	41
1.6.6	Generating softkey menus and dialog forms.....	42
1.7	Generating user menus.....	68
1.7.1	Creating processing cycle forms.....	68
1.7.2	Substitution characters.....	71
1.8	Addressing components.....	71
1.8.1	PLC addressing.....	71
1.8.2	Addressing NC variables.....	73
1.8.3	Generating NC/PLC addresses during the runtime.....	73
1.8.4	Addressing drive components.....	73
1.8.5	Addressing machine and setting data.....	76
1.8.6	Addressing user data.....	77
1.9	Predefined functions.....	78
2	Generating commissioning dialogs.....	111
2.1	Overview of functions.....	111
2.2	Configuration in the PLC user program.....	114
2.3	Display on the user interface.....	116
2.4	Creating language-dependent texts.....	117
2.5	User example for a power unit.....	118
2.6	Script language.....	119
2.6.1	CONTROL_RESET.....	122
2.6.2	FILE.....	122
2.6.3	OPTION_MD.....	123
2.6.4	PLC_INTERFACE.....	124
2.6.5	POWER_OFF.....	125
2.6.6	WAITING.....	125
2.6.7	XML identifiers for the dialog.....	125

2.6.8 SOFTKEY_OK, SOFTKEY_CANCEL.....126

Index.....129

Generating user dialogs

1.1 Scope of functions

Overview

The "Generate user dialogs" function offers an open structure and enables the user to develop customer-specific and application-specific HMI interfaces.

The control system offers an XML-based script language for generating user dialogs.

This script language makes it possible to display machine-specific menus and dialog forms in the <CUSTOM> operating area on the HMI.

All dialog forms can be designed on a language-neutral basis. In such cases, the system reads out the texts to be displayed from the accompanying language database.

Use

The defined XML instructions offer the following properties:

1. Display dialogs containing the following elements:
 - Softkeys
 - Variables
 - Texts and Help texts
 - Graphics and Help displays
2. Call dialogs by:
 - Pressing the (start) softkeys
3. Restructure dialogs dynamically:
 - Edit and delete softkeys
 - Define and design variable fields
 - Insert, exchange, and delete display texts (language-dependent or language-neutral)
 - Insert, exchange, and delete graphics
4. Initiate operations in response to the following actions:
 - Displaying dialogs
 - Inputting values (variables)
 - Selecting a softkey
 - Exiting dialogs
5. Data exchange between dialogs

1.1 Scope of functions

6. Variables

- Read (NC, PLC and user variables)
- Write (NC, PLC and user variables)
- Combine with mathematical, comparison or logic operators

7. Execute functions:

- Subprograms
- File functions
- PI services

8. Apply protection levels according to user classes

The valid elements (tags) for the script language are described in the "XML tags" (Page 15) section.

Note

The following section is not intended as a comprehensive description of XML (Extensible Markup Language). Please refer to the relevant specialist literature for additional information.

1.2 Fundamentals of Configuration

Configuration files

The defining data for new user interfaces are stored in configuration files. These files are automatically interpreted and the result displayed on the screen. Configuration files are not stored in the software supplied and must first be set up and loaded by the user.

An XML editor or another form of text editor can be used to generate the configuration files.

Note

Important:

All file names must be written in lowercase.

Menu tree principle

Several interlinked dialogs create a menu tree. A link exists if you can switch from one dialog to another. You can use the newly defined horizontal/vertical softkeys in this dialog to call the preceding or any other dialog.

Configured start softkeys can be used to create a further menu tree behind the start menu:

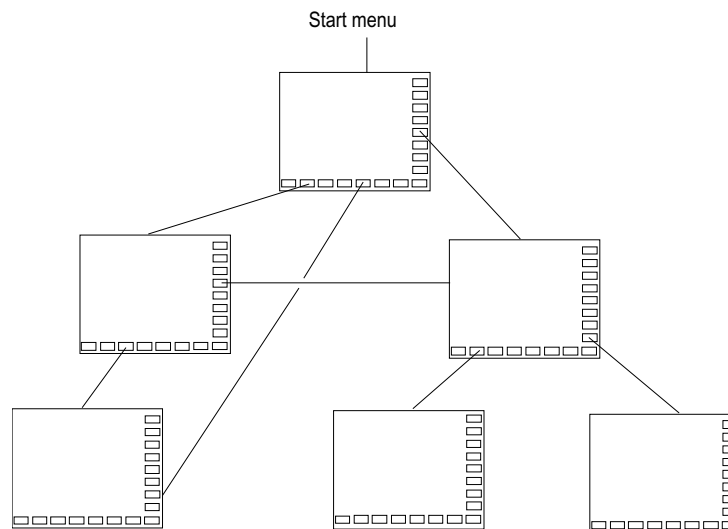


Figure 1-1 Menu tree for user dialogs

Start menu

The start menu is defined by the name "main" in the "xmldial.xml" file. The start menu is used to initiate your own operating sequences.

You can use the "main" menu to link your own dialogs or additional softkey bars so that they can be loaded and used for executing additional actions.

The figure below shows the manufacturer's drive on the control system.

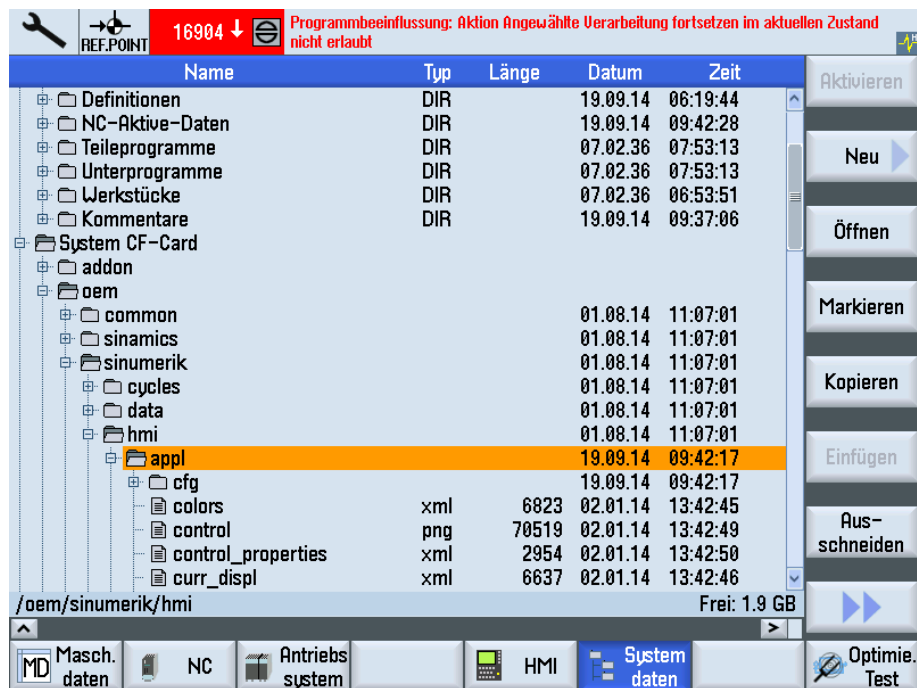


Figure 1-2 Manufacturer's drive

The following files in the control system's "Manufacturer's drive" directory are needed to configure the user dialogs:

Table 1-1 Files for configuration

File type	Name of the file	Description	Storage location in the HMI operating area Commissioning
Script file	"xmldial.xml"	This script file uses XML tags to control the process image of the configured softkey menus and dialog forms in the <CUSTOMER> operating area on the HMI.	"Systemcard" > "oem/sinumerik/hmi" > in the "appl" subdirectory for the applications
Text file	"oem_xml_screens_xx.ts"	This text file contains the texts for the menus and dialog forms for the individual languages.	"Startup files" > "Manufacturer's drive" > in the "lng" subdirectory for the desired languages
Bitmaps		The control system supports BMP and PNG formats.	"Startup files" > "Manufacturer's drive" > in the "ico" subdirectory
XML files inserted in the "xmldial.xml" control file with the "INCLUDE" XML tag.	E.g. "machine_settings.xml"	These files also contain programmed instructions for displaying the dialog forms and parameters on the HMI.	"Startup files" > "Manufacturer's drive" > in the "appl" subdirectory for the applications

1.3 Configuration files

Introduction

The screen below shows the manufacturer's drive on the control system.

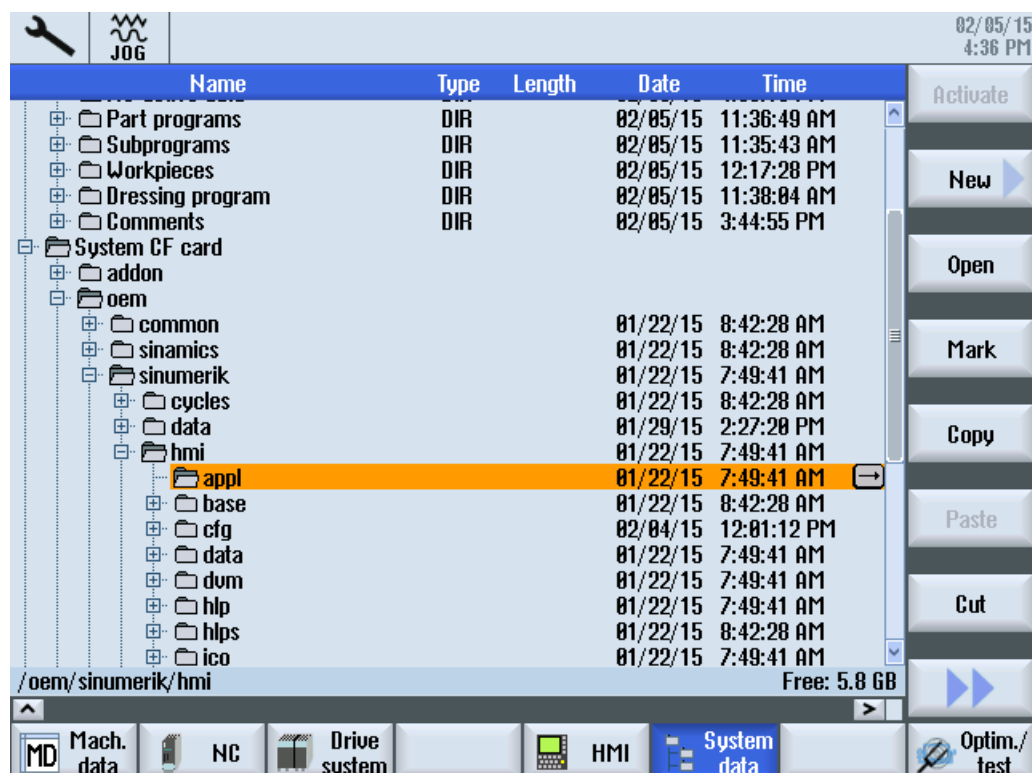


Figure 1-3 Manufacturer's drive

The following files in the control system's "Manufacturer's drive" directory are needed to configure the user dialogs:

Table 1-2 Files for configuration

File type	Name of the file	Meaning	Storage location in the HMI's <SYSTEM> or <PROGRAM MANAGER> operating area
Script file	"xmldial.xml"	This script file uses XML tags to control the process image of the configured softkey menus and dialog forms in the <COSTUMER> operating area on the HMI.	"Startup files" > "Manufacturer's drive" > in the "appl" subdirectory for the applications
Text file	"oem_xml_screens_xxx.ts"	This text file contains the texts for the menus and dialog forms for the individual languages.	"Startup files" > "Manufacturer's drive" > in the "lng" subdirectory for the languages
Bitmaps		The control system supports BMP and PNG formats.	"Manufacturer's drive" > in the "ico" subdirectory The bitmaps are saved in the subdirectory for the screen resolution belonging to the control. Note: If a path to the bitmap file is specified, the files can be stored in this directory directly.
XML files inserted in the "xmldial.xml" control file with the "INCLUDE" XML tag.	E.g. "machine_settings.xml"	These files also contain programmed instructions for displaying the dialog forms and parameters on the HMI.	"Manufacturer's drive" > in the "appl" subdirectory for the applications

Dependencies of files for configuring user dialogs

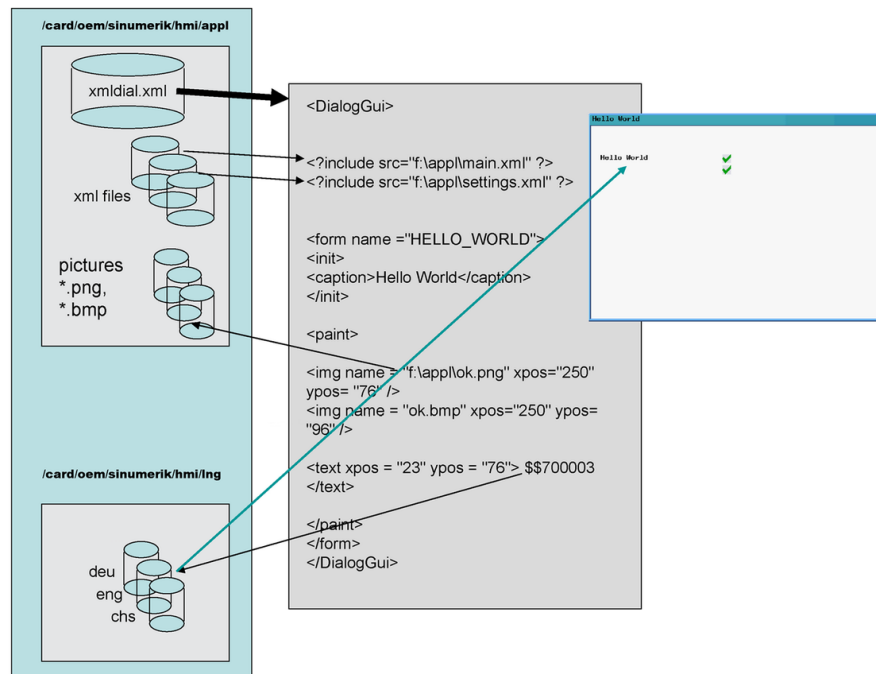


Figure 1-4 Dependencies

Load configuration

As described in the "Storage location in the HMI" column in the previous "Files for configuration" table, the generated files must be copied to the relevant subdirectories in the "Manufacturer's drive" menu.

Note

As soon as there is an "xmldial.xml" script file in the subdirectory for applications on the manufacturer's drive, the user can start this user dialog in the <CUSTOM> operating area.

After the initial copying process, the control system needs to be reset via "Normal power-up".

Example of a user dialog on the HMI

The configured softkey menus are displayed when the <CUSTOM> operating area is called. This enables the user to operate the dialog forms which have been configured.

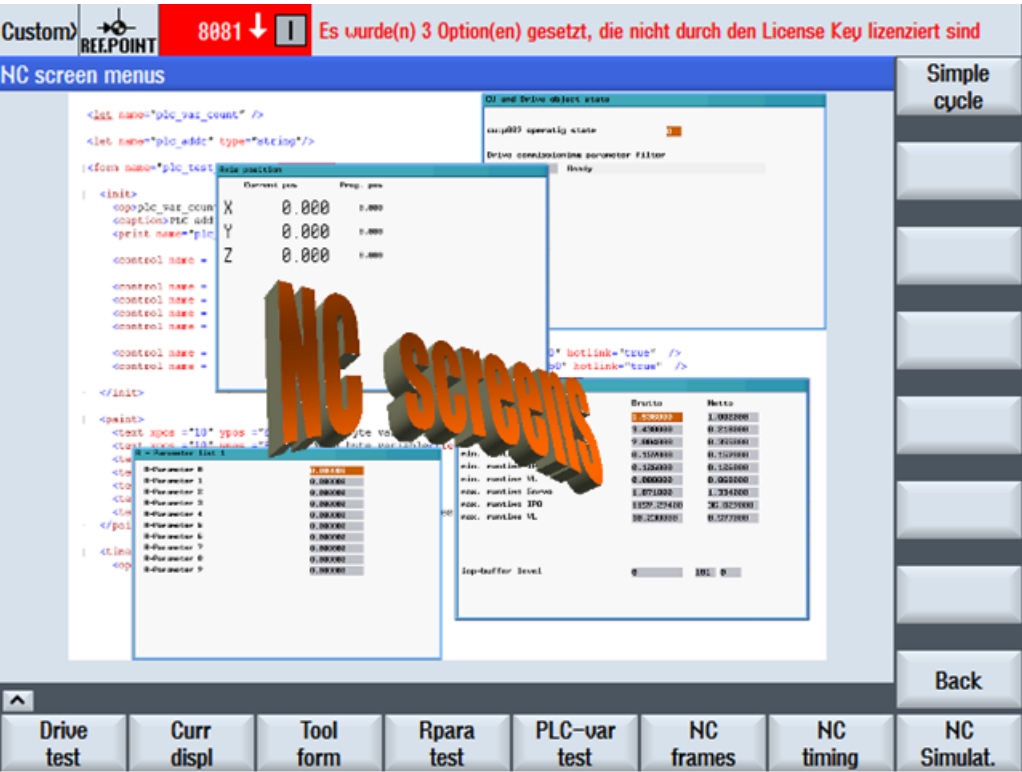


Figure 1-5 Example of a user dialog in the <CUSTOM> operating area

See also

Predefined functions (Page 78)

1.4 Structure of configuration file

Overview

A configuration file consists of the following elements:

- Description of the "main" start menu with start softkeys
- Definition of dialogs
- Definition of variables
- Description of the blocks
- Definition of softkey bars

The following screens show an XML script for the "xmldial.xml" file and the corresponding screenshots.

The script contains the dialogs for displaying actual values and residual distances, as well as an R parameter list.

Generating user dialogs

1.4 Structure of configuration file

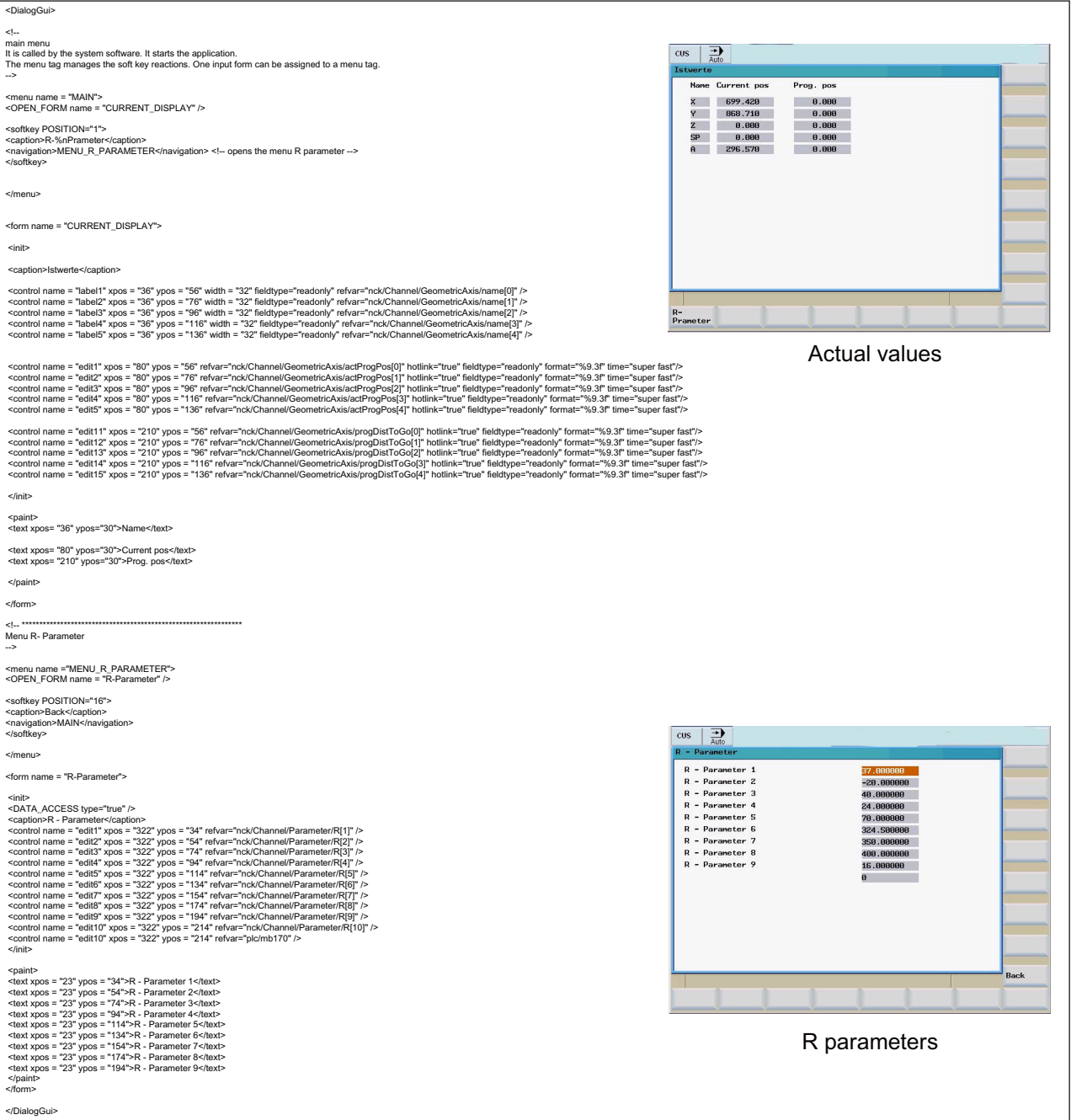


Figure 1-6 XML script and dialog forms with parameters

1.5 Language dependency

Language-dependent texts are used for:

- Softkey labels
- Headers
- Help texts
- Any other texts

The language-dependent texts are stored in text files.

Note

You will need to perform the following steps when using these text files:

- Make them available in the required languages.
 - Transfer them into the relevant language directories of the control system.
-

1.6 XML identifier

1.6.1 General structure

Structure and instructions of the script file for dialog configuration

All dialog configurations should be stored in the **DialogGui** tag.

```
<DialogGui>
...
</DialogGui>
```

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<DialogGui>
...
<FORM name ="Hello_World">
<INIT>
<CAPTION>Hello World</CAPTION>
</INIT>
...
</FORM>

</DialogGui>
```

Instructions

The language offers the following instructions for executing conditional instructions and loop controls:

- For loop
- While loop
- Do with loop
- Conditional processing
- Switch and case instructions
- Operator controls in a dialog form
- Softkey descriptions
- Define variables

For a detailed description of instructions, see Instruction/identifier descriptions (Page 16).

1.6.2 Instruction/identifier descriptions

The following **XML tags** are defined for generating dialogs and menus, and for executing program sequences:

Note

Attribute values that are in quotation marks "<...>" should be replaced by the currently used expressions.

Example:

<DATA_LIST action="read/write/append" id="<list name>">

is programmed as follows:

<DATA_LIST action="read/write/append" id="my datalist">

Tag identifier	Meaning
BREAK	Conditional cancellation of a loop.
CONTROL_RESET	The tag enables one or more control components to be restarted. Syntax: <CONTROL_RESET resetnc="TRUE" /> Attributes: • RESETNC = "TRUE" The NC component is restarted. • RESETDRIVE = "TRUE" The drive components are restarted.

Tag identifier	Meaning
CREATE_CYCLE_EVENT	If the parser starts to process the tag CREATE_CYCLE, initially, the message <CREATE_CYCLE_EVENT sent> is sent to the active form. This message can be used for preparing the cycle parameters, before the parser generates the NC operation from the parameter list and the generation rule. Syntax: <pre> <CREATE_CYCLE_EVENT> </CREATE_CYCLE_EVENT> </pre> Example: <pre> <SOFTKEY_OK> ... <CREATE_CYCLE /> <SOFTKEY_OK> <FORM> <NC_INSTRUCTION>MY_CYCLE(\$P1, \$P2)</ NC_INSTRUCTION > <CREATE_CYCLE_EVENT> <type_cast name="P1" type="int"/> <OP> P1 = P1 * 150 </OP> </CREATE_CYCLE_EVENT> </FORM> </pre>

Tag identifier	Meaning
DATA	The tag enables the NC, PLC, GUD and drive data to be directly written to. The "Component addressing" (Page 71) section contains details on address formation. Attribute: name Variable address Tag value: All alphanumeric terms are approved as tag values. If a value is to be written from a local variable directly, the \$ replacement operator preceding the name of the local variable should be used. Syntax: <pre><DATA name="<variable name>"> value </DATA></pre> Example: <pre><DATA name = "plc/mb170"> 1 </DATA> ... <LET name = "tempVar"> 7 </LET> <!-- the contents of the local variables "tempVar" are written to bit memory byte 170 -> <DATA name = "plc/mb170">\$tempVar</DATA></pre>
DATA_LIST	The tag enables the listed drive and machine data to be saved or restored. Addresses are listed in lines. The "Component addressing" (Page 71) section contains details on address formation. Up to 20 temporary data lists can be created. Attributes: action <i>read</i> – the values of the listed variables are stored in a temporary memory <i>append</i> – the values of the listed variables are added to an existing list <i>write</i> – the backed up values are copied to the relevant machine data id The identifier identifies the temporary memory Syntax: <pre><DATA_LIST action="<read/write/append>" id="<list name>"> NC/PLC address compilation </DATA_LIST></pre> Example: <pre><DATA_LIST action = "read" id="<name>"> nck/channel/parameter/r[2] nck/channel/parameter/r[3] nck/channel/parameter/r[4] \$MN_USER_DATA_INT[0] ... </ DATA_LIST> <DATA_LIST action = "write" id="<name>" /></pre>

Tag identifier	Meaning
DYNAMIC_INCLUDE	The tag includes an XML script file. Contrary to the INCLUDE tag, read-in is first only realized when executing the corresponding operation. For large projects, the use of the tag reduces the load time of the customer area and/or the cycle support. Further, the average level of resources is reduced, as not all of the dialogs are always called during a session. Syntax: <pre><DYNAMIC_INCLUDE src="path name"/></pre> Example: <pre><SOFTKEY POSITION="3"> <CAPTION>MY_MENU</CAPTION> <DYNAMIC_INCLUDE src/> <NAVIGATION>MY_MENU</NAVIGATION> </SOFTKEY></pre>
ELSE	Instruction for situations where the condition has not been met (IF, THEN, ELSE)
FORM	The tag contains the description of a user dialog. The relevant tags are described in the section on generating menus and dialog forms. Syntax: <pre><FORM name="<dialog name>" color="#ff0000"></pre> Attributes: • color Background color of the dialog form (color coding, see Chapter Color coding (Page 40)) – Default white • name Identifier of the form • type Permissible value is <i>cycle</i> , which identifies a user cycle screen form • xpos X-position of the top left corner of the dialog box (optional) • ypos Y position of the top left corner (optional) • width Extension in the X direction (in pixels) (optional) • height Extension in the Y direction (in pixels) (optional)
HMI_RESET	The tag initiates an HMI restart. The interpretation is cancelled after this operation.

Tag identifier	Meaning
IF	Conditional instruction (IF, THEN, ELSE) The THEN and ELSE tags are enclosed in the IF tag. The condition that is executed in the CONDITION tag follows the IF tag. The further processing of the instructions depends upon the result of the operation. If the function result is true, then the THEN branch is executed and the ELSE branch is skipped. If the result of the function is false, the parser executes the ELSE branch. Syntax: <pre><IF> <CONDITION> Condition != 7 </CONDITION> <THEN> Instruction for the case: Condition fulfilled </THEN> <ELSE> Instruction for the case: Condition not fulfilled </ELSE> </IF></pre> Example: <pre><IF> <CONDITION> "plc/mb170" != 7 </CONDITION> <THEN> <OP> "plc/mb170" = 7 </OP> ... </THEN> <ELSE> <OP> "plc/mb170" = 2 </OP> ... </ELSE> </IF></pre>
INCLUDE	The instruction includes an XML description. (see also DYNAMIC_INCLUDE in this table) Attribute: src Contains the path name. Syntax: <pre><?INCLUDE src="<Path name>" ?></pre>

Tag identifier	Meaning																													
LET	The instruction creates a local variable under the specified name. Fields: Using the attribute dim (dimension) single or two-dimensional fields can be created. The field index addresses the individual field elements. For a two-dimensional field, initially the line index is specified and then the column index. - Single-dimensional field: Indices 0 to 4 <table><tr><td>0</td></tr><tr><td>1</td></tr><tr><td>2</td></tr><tr><td>3</td></tr><tr><td>4</td></tr></table> - Two-dimensional field: Index line 0 to 3 and index column 0 to 5 <table><tr><td>0,0</td><td>0,1</td><td>0,2</td><td>0,3</td><td>0,4</td><td>0,5</td></tr><tr><td>1,0</td><td>1,1</td><td>1,2</td><td>1,3</td><td>1,4</td><td>1,5</td></tr><tr><td>2,0</td><td>2,1</td><td>2,2</td><td>2,3</td><td>2,4</td><td>2,5</td></tr><tr><td>3,0</td><td>3,1</td><td>3,2</td><td>3,3</td><td>3,4</td><td>3,5</td></tr></table> Attributes: name Variable name type The variable type can be an integer (INT), double (DOUBLE), float (FLOAT) or string (STRING). If there is no type instruction specified, the system creates an integer variable. <pre><LET name = "VAR1" type = "INT" /></pre> permanent If the attribute is set to true, then the variable value is saved permanently. This attribute is only effective for a global variable. dim The following number of field elements must be specified. For a two-dimensional field, the second dimension is specified after the first dimension separated by a comma. A field element is accessed via the field index, which is specified in square brackets after the variable name. name[index] or name[row,column] - Single-dimensional field: dim="<Number of elements>" - Two-dimensional field: dim="<Number of lines>,<number of columns>" Non-initialized field elements are pre-assigned with "0".	0	1	2	3	4	0,0	0,1	0,2	0,3	0,4	0,5	1,0	1,1	1,2	1,3	1,4	1,5	2,0	2,1	2,2	2,3	2,4	2,5	3,0	3,1	3,2	3,3	3,4	3,5
0																														
1																														
2																														
3																														
4																														
0,0	0,1	0,2	0,3	0,4	0,5																									
1,0	1,1	1,2	1,3	1,4	1,5																									
2,0	2,1	2,2	2,3	2,4	2,5																									
3,0	3,1	3,2	3,3	3,4	3,5																									

Tag identifier	Meaning
LET Continued	Example: Single-dimensional field: <let name="array" dim="10"></let> Two-dimensional field: <let name="list_string" dim="10,3" type="string"></let> Pre-assignment: A variable can be initialized with a value. <LET name = "VAR1" type = "INT"> 10 </LET> If values comprising NC or PLC variables are saved in a local variable, the assignment operation automatically adapts the format to that of the variables which have been loaded. • Pre-assignment for a string variable: Texts containing more than one line can be assigned to a string variable if the formatted text is transferred as a value. If a line is to end with a line feed <LF> (line feed) , the characters "\n" should be added at the end of the line.

Tag identifier	Meaning
	<pre> <LET name = "text" type = "string"> F4000 G94\\n G1 X20\\n Z50\\n M2\\n </LET>> </pre> Fields (Arrays): <pre> <let name="list" dim="10,3"> {1,2,3}, {1,20} </let> </pre> <pre> <let name="list_string" dim="10,3" type="string"> {"text 10","text 11"}, {"text 20","text 21"} </let> </pre> Assignment: Values made up of the machine data or subroutines can be assigned to a variable using the assignment operation "=". A variable remains valid until the end of the higher-level XML block. Variables which are to be available globally should be created directly after the DialogGUI tag. The following must be observed for a dialog box: • The message processing opens the corresponding tag. • The tag is closed after the message has been executed. • All variables within the tag are deleted when closing. Variable type struct This variable type contains a composition of variables that can be addressed using the structure name. A structure can contain all variable types and structures. Within the structure, a variable is declared with the "element" tag. The attributes of the tags and the initialization correspond to the attributes and initialization of the let instruction. <pre> <let name="name" type="struct"> <element name="<Name>" type="<Variable type>" /> </let> </pre> Access to a variable of the structure is via the structure name and variable name. Both names are separated by a point operator.

Tag identifier	Meaning
	<pre><op> Structure_name.variable_name = value; </op></pre> Example: <pre><let name="info" type="struct"> <element name="id" type="int" /> <element name="name" type="string" /> <element name="phone" type="string" /> </let></pre> <pre><op> info.id = 1; info.name = _T"my name"; info.phone = _T"0034 45634"; </op></pre>
MSG	The operator component shows the message which is indicated in the tag. If an alarm number is used, the dialog box displays the text which is saved for the number. Example: <pre><MSG text ="my message" /></pre>

Tag identifier	Meaning
MSGBOX	The instruction opens a message box whose return value can be used for branching. Syntax: <MSGBOX text="<Message>" caption="<caption>" retvalue="<variable name>" type="<button type>" /> Attributes: • text Text • caption Header • retvalue Name of the variables to which the return value is copied: 1 – OK 0 – CANCEL • type Acknowledgment options: "BTN_OK" "BTN_CANCEL" "BTN_OKCANCEL" If an alarm number is used for the "text" or "caption" attribute, the message box displays the text which is saved for the number. Example: <MSGBOX text="Test message" caption="Information" retvalue="result" type="BTN_OK" />

Tag identifier	Meaning
OP	The tag executes the specified operations. The operations listed in Chapter "Operators (Page 41) can be executed. For the purpose of accessing the NC, PLC, and drive data, the complete variable name should be placed in quotation marks. The "Component addressing" (Page 71) section contains details on address formation. PLC: "PLC/MB170" NC: "NC/Channel/..." Example: <pre><LET name = "tmpVar" type="INT"> </LET> <OP> tmpVar = "plc/mb170" </OP> <OP> tmpVar = tmpVar *2 </OP> <OP> "plc/mb170" = tmpVar </OP></pre> More than one equation can be used within an operation tag. A semicolon marks the end of the instruction. Example: <pre><op> x= x+1; y = y+1; </op></pre> Character string processing: The operation instruction is able to process character strings and assign the results to the string variable specified in the equation. The identifier <code>_T</code> should be placed at the start as a means of identifying text terms. Formatting of variable values is also possible. The identifier <code>_F</code> should be placed at the start of the formatting regulation, followed by the format instruction. The address is then specified for the variable. Example: <pre><LET name="buffer" type="string"></LET> <OP> buffer = _T"unformatted value R0= " + "nck/Channel/Parameter/ R[0]" + _T" and " + _T"\$85051" + _T" formatted value R1 " + _F %9.3f"nck/Channel/Parameter/R[1]" </OP></pre>

Tag identifier	Meaning
XML_PARSER	The "XML_PARSER" tag parses XML files. The parser interprets an XML file and calls defined call-back functions. Each call-back function belongs to a predefined event. The programmer can process the XML data within this function. Predefined events: • start document The parser opens the document and starts with the parsing. • end document The parser closes the document. • start element The parser has found an element and creates a list with all attributes and attribute values. These lists are forwarded to the call-back function. • end element The end of the element was found. • characters The parser forwards all characters of an element. • error The parser has detected a syntax error. When an event occurs, the parser calls the call-back function and checks the function return value. If the function returns "true", the parser continues the process. Interfaces The value of the name attribute contains the path to the XML file. To assign events to the call-back functions, the following properties must be specified: Default - startElementHandler - endElementHandler - charactersHandler Optional - errorHandler - documentHandler The value of an attribute defines the name of the call-back function. Example: <pre><XML_PARSER name="f:\appl\xml_test.xml"> <!-- standard handler --> <property startElementHandler="startElementHandler" /> <property endElementHandler="endElementHandler" /> <property charactersHandler="charactersHandler" /> <!-- optional handler --> <property errorHandler="errorHandler" /> <property documentHandler="documentHandler" /> </XML_PARSER></pre>

Tag identifier	Meaning
XML_PARSERContinued	The parser also supplies variables so that the call-back functions can access the event data. startElementHandler: Function parameters tag_name - tag name num - number of found attributes System variables \$xmlAttribute String array that contains the 0-num attribute name range. \$xmlValue String array that contains the 0-num attribute value range. Example: <pre><function_body name="startElementHandler" return="true" parameter="tag_name, num"> <print text="attribute name %s"> \$xmlAttribute[0]</print> <print text="attribute value %s"> \$xmlValue [0]</print> </function_body></pre> endElementHandler: Function parameters tag_name - tag name Example: <pre><function_body name="endElementHandler" parameter="tag_name"> <print text="name %s"> tag_name </print> </function_body></pre>

Tag identifier	Meaning												
XML_PARSER Continued	charactersHandler: System variables <table> <tr> <td>\$xmlCharacters</td><td>String with data</td></tr> <tr> <td>\$xmlCharactersStart</td><td>Always 0</td></tr> <tr> <td>\$xmlCharactersLength</td><td>Number of bytes</td></tr> </table> Example: <pre><function_body name="charactersHandler" return="true" > <print text="chars %s"> \$xmlCharacters </print> </function_body></pre> documentHandler: Function parameters <table> <tr> <td>state</td><td>1 start document, 2 end document</td></tr> </table> errorHandler: System variables <table> <tr> <td>\$xmlErrorString</td><td>Contains the invalid line (system variable)</td></tr> </table> Example: <pre><function_body name="errorHandler" > <print text="error %s">\$xmlErrorString</print> </function_body></pre> Call-back result: <table> <tr> <td>\$return</td><td>If 1 (true), the parser continues parsing the file</td></tr> </table>	\$xmlCharacters	String with data	\$xmlCharactersStart	Always 0	\$xmlCharactersLength	Number of bytes	state	1 start document, 2 end document	\$xmlErrorString	Contains the invalid line (system variable)	\$return	If 1 (true), the parser continues parsing the file
\$xmlCharacters	String with data												
\$xmlCharactersStart	Always 0												
\$xmlCharactersLength	Number of bytes												
state	1 start document, 2 end document												
\$xmlErrorString	Contains the invalid line (system variable)												
\$return	If 1 (true), the parser continues parsing the file												

Tag identifier	Meaning
CONTROL	Change control after the creation A control tag changes the control properties of an existing control after the creation. The tag must be specified with the name of the control to be changed and the new properties. It can be executed only within a form tag. The following properties can be changed: • xpos • ypos • width • height • color_bk • color_fg • access level • fieldtype • itemdata • min. • max • default The reference variable cannot be modified. If a property should be changed by triggering caused by a softkey event, the send message tag must transfer this request to the form context. The tag message acquires the message. Example <pre> <menu name = "main"> <open_form name = "attrib_form" /> <softkey POSITION="3"> <caption>Set%nro</caption> <send_message>1, 0</send_message> </softkey> <softkey POSITION="4"> <caption>Set%nrw</caption> <send_message>2, 0</send_message> </softkey> </menu> <form name="attrib_form"> <init> <control name="c_p0" xpos="60" ypos="70" width="272" /> </init> <message> <switch> <condition>\$message_par1</condition> <case value="1"> <control name="c_p0" fieldtype="readonly" /> </case> <case value="2"> <control name="c_p0" fieldtype="edit" /> </case> </switch> </message> </form> </pre>

Tag identifier	Meaning
OPERATION	Operation Instructions can be moved within an equation. Move left "<<" operator The << function moves bits to the left. You can specify the value and the number of move increments directly or with a variable. If the limit of the data format is reached, the bits will be moved beyond the limit without an error message being issued. Execution rule Execution from left to right; '+' and '-' have priority over '<<' and '>>' Example: <pre><op> idx = idx << 2 </op> <op> idx = 3 + idx << 2 </op></pre> Move right ">>" operator The >> function moves bits to the right. You can specify the value and the number of move increments directly or with a variable. If the limit of the data format is reached, the bits will be moved beyond the limit without an error message being issued. Execution rule Execution from left to right; '+' and '-' have priority over '<<' and '>>' Example: <pre><op> idx = idx >> 2 </op> <op> idx = 3+idx >> 2</op></pre>
PASSWORD	The tag opens a dialog for entering the password. Once the entry has been confirmed, the character string is available in the specified reference variable. Syntax: <pre><PASSWORD refVar ="<variable name>" /></pre> Attribute: • refVar Name of the reference variable Example: <pre><PASSWORD refvar="plc/mw107" /></pre>
POWER_OFF	A message prompts the operator to switch the machine off. The message text is permanently saved in the system.

Tag identifier	Meaning
PRINT	The tag outputs a text in the dialog line or copies the text to the variable specified. If the text contains formatting identifiers, the variable values are inserted at the appropriate places. Syntax: <code><PRINT name="Variable name " text="text %Formatting "> Variable, ...</code> <code></PRINT></code> <code><PRINT text="text %Formatting"> Variable, ... </PRINT></code> Attributes: • name Name of the variable where the text is to be stored (optional) • text Text Formatting: The character "%" causes the variable specified as the value to be formatted. %[Flags] [Width] [.decimal places] type • Flags: Optional character for defining output formatting: – Right-justified or left-justified ("-") for left-justified – Add leading zeros ("0") – Fill with blanks • Width: The argument defines the minimum output width for a non-negative number. If the value to be output has fewer places than the argument defined, the missing spaces are filled with blanks. • Decimal places: With floating-point numbers, the optional parameter defines the number of decimal places. • Type: The type character defines which data formats are transferred for the print instruction. These characters need to be specified. – d: Integer value – f: Floating-point number – s: String

Tag identifier	Meaning
PRINT Continued	Values: Number of variables whose values are to be inserted into the text. The variable types must match the corresponding type identifier for the formatting instruction and must be separated from one another with a comma. Example: Output of a text in the information line <code><PRINT text="Infotext" /></code> Output of a text with variable formatting <code><LET name="trun_dir"></LET></code> <code><PRINT text="M%d">trun_dir</PRINT></code> Output of a text in a string variable with variable formatting <code><LET name="trun_dir"></LET></code> <code><LET name="str" type="string" ></LET></code> <code><print name="str" text="M%d ">trun_dir</print></code>
PROGRESS_BAR	The tag opens or closes a progress bar. The bar is displayed below the application window. Syntax: <code><PROGRESS_BAR type="<true/false>"> value </ PROGRESS_BAR></code> Attributes: • type = "TRUE" - opens the progress bar • type = "FALSE" - closes the progress bar • min (optional) – minimum value • max (optional) – maximum value Value: • Value Percentage position of the bar Example: <code><PROGRESS_BAR type="true" min="0" max="101" >20< /</code> <code>PROGRESS_BAR>.....<PROGRESS_BAR >50< /</code> <code>PROGRESS_BAR>.....<PROGRESS_BAR type="false" >100< /PROGRESS_BAR></code>

Tag identifier	Meaning
SEND_MESSAGE	The tag sends a message with two parameters to the active form, which is processed in the tag message. Syntax: <code><SEND_MESSAGE>p1, p2</SEND_MESSAGE></code> Example: <code><SOFTKEY POSITION="3"> <caption>Set%nParameter</caption> <send_message>1, 0</send_message> </SOFTKEY></code> <code><FORM> <MESSAGE> <SWITCH> <CONDITION>\$message_par1</CONDITION> <CASE value="1"> </CASE> </SWITCH> </MESSAGE> </FORM></code>
SLEEP	The tag interrupts script execution for the specified period. The interruption time is obtained from the transferred value multiplied by the time base of 50 ms. Syntax: <code><SLEEP value="Interruption time" /></code> Example: Wait time, 1.5 sec. <code><SLEEP value="30" /></code>
STOP	Interpretation is cancelled at this point.

Tag identifier	Meaning
SWITCH	The SWITCH instruction describes a multiple choice. A term is evaluated once and compared with a number of constants. If the expression matches the constants, the instructions are executed within the CASE instruction. The DEFAULT instruction is executed when none of the constants match the expression. Syntax: <pre><SWITCH> <CONDITION> Value </CONDITION> <CASE value="Constant 1"> Instructions ... </CASE> <CASE value="Constant 2"> Instructions ... </CASE> <DEFAULT> Instructions ... </DEFAULT> </SWITCH></pre>
SHOW_CONTROL	The visibility of a control can be controlled using the tag. Syntax: <pre><SHOW_CONTROL name="<name>" type="<type>" /></pre> Attributes: • name Name of the control • type = "TRUE" - control becomes visible • type = "FALSE" - control becomes invisible (hidden) Example: <pre><SHOW_CONTROL name="myEditfield" type="false" /> ... <SHOW_CONTROL name="myEditfield" type="true" /></pre>
TYPE_CAST	The tag converts the data type of a local variable. Syntax: <pre><type_cast name="variable name" type=" new type" /></pre> Attributes: • name Variable name • type New data type
THEN	Operation, if the condition has been fulfilled (IF, THEN, ELSE)

Tag identifier	Meaning
FOR	For loop for (initialization; test; continuation) instruction(s) Syntax: <code><FOR></code> <code><INIT>...</INIT></code> <code><CONDITION>...</CONDITION></code> <code><INCREMENT>...</INCREMENT></code> Instructions ... <code></FOR></code> The For loop is executed as follows: 1. Evaluation of the term initialization (INIT). 2. Evaluation of the term test (CONDITION) as a Boolean term. If the value is false, the For loop is ended. 3. Execution of the following instructions. 4. Evaluation of the term continuation (INCREMENT). 5. Continue with 2. All the variables within the INIT, CONDITION, and INCREMENT branches should be created outside the FOR loop. Example: <code><LET name = "count">0</LET></code> <code><FOR></code> <code> <INIT></code> <code> <OP> count = 0</OP></code> <code> </INIT></code> <code> <CONDITION> count <= 7 </CONDITION></code> <code> <INCREMENT></code> <code> <OP> count = count + 1 </OP></code> <code> </INCREMENT></code> <code> <OP> "plc/qb10" = 1+ count </OP></code> <code></FOR></code>
WAITING	The tag waits for the component to undergo a hot restart after an NC or drive reset. Attributes: • WAITINGFORNC = "TRUE" - the system waits for the NC to restart • WAITINGFORDRIVE = "TRUE" - the system waits for the drives to restart Syntax: <code><WAITING WAITINGFORNC = "TRUE"/></code> Example: ... <code><CONTROL_RESET resetnc = "true" resetdrive = "true"/></code> <code><WAITING waitingfornc = "true" waitingfordrive = "true" /></code> ...

Tag identifier	Meaning
WHILE	WHILE loop <pre>WHILE (Test) Instruction</pre> Syntax: <pre><WHILE> <CONDITION>...</CONDITION> Instructions ... </WHILE></pre> The While loop executes a sequence of instructions repeatedly while a condition is met. This condition is tested before the sequence of instructions is executed. Example: <pre><WHILE> <CONDITION> "plc/ib9" == 0 </CONDITION> <DATA name = "PLC/qb11"> 15 </DATA> </WHILE></pre>

Tag identifier	Meaning
DO_WHILE	Do while loop <pre>DO Instructions WHILE (Test)</pre> Syntax: <pre><DO_WHILE> Instructions ... <CONDITION>...</CONDITION> </DO_WHILE></pre> The Do while loop comprises an instruction block and a condition. The code within the instruction block is executed first, then the condition is analyzed. If the condition is true, the function executes the code section again. This is continuously repeated until the condition is false. Example: <pre><DO_WHILE> <DATA name = "PLC/qb11"> 15 </DATA> <CONDITIION> "plc/ib9" == 0 </CONDITION> </DO_WHILE></pre>

Tag identifier	Meaning
TYPEDEF	A new identifier for a data type can be specified with the "typedef" tag. This has the benefit for the structure definitions that the data type can be defined once and then used as a data type in a LET instruction. The identifier and type are expected as attributes. The parser supports only the specification of structure definitions. Within the type definition, a variable is declared with the "element" tag. The attributes of the tag correspond to the attributes of the let instruction. <pre><typedef name="<identifier>" type="struct"> <element name="<Name>" type="<variable type>" /> </typedef></pre> After defining, the identifier can be used as data type for the LET instruction. <pre><let name="<Variable name>" type="<Identifier>"></let></pre> Example: <pre><typedef name="my_struct" type="struct"> <element name="id" type="int" /> <element name="name" type="string" /> <element name="phone" type="string" /> </typedef></pre> <pre><let name="info" type="my_struct"></let> <op> info.id = 1; info.name = _T"my name"; info.phone= _T"0034 45634"; </op></pre>

1.6.3 Color coding

The color attribute uses the color coding scheme for the HTML language.

In terms of syntax, color specifications consist of the "#" (hash) character and six digits from the hexadecimal system, with each color represented by two digits.

R – Red

G – Green

B – Blue

#RRGGBB

Example:

color = "#ff0011"

1.6.4 Special XML syntax

Characters with special meanings in XML syntax have to be rewritten if they are to be displayed correctly by a general XML editor.

The following characters are affected:

Character	Notation in XML
<	<
>	>
&	&
"	"
'	'

1.6.5 Operators

The operation instruction processes the following operators:

Operator	Meaning
=	Assignment
==	Equal to
<, <	Less than
>, >	Greater than
<=, <=	Less than or equal to
>=, >=	Greater than or equal to
	OR operation in bits
	Logic OR operation
&, &	AND operation in bits
&&, &&	Logic AND operation
+	Addition
-	Subtraction
*	Multiplication
/	Division
!	Not
!=	Not equal to

Operation instructions are processed from left to right. It may make sense to place terms in parentheses under certain circumstances in order to define the priority for executing subterms.

1.6.6 Generating softkey menus and dialog forms

User menus can only be inserted if there is a main-menu tag with the name "main" in the XML description. This tag is called by the system after the <CUSTOM> operating area has been activated. Further menu branches and dialog-box activation can be defined within the tag.

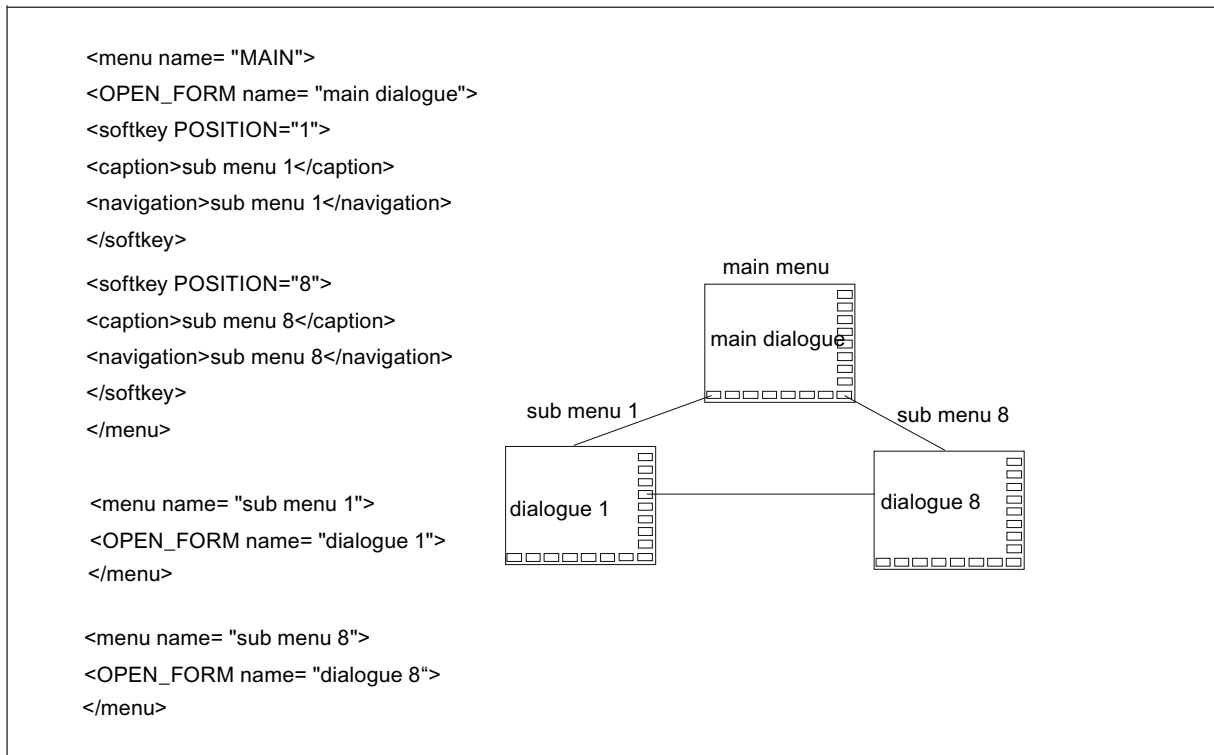
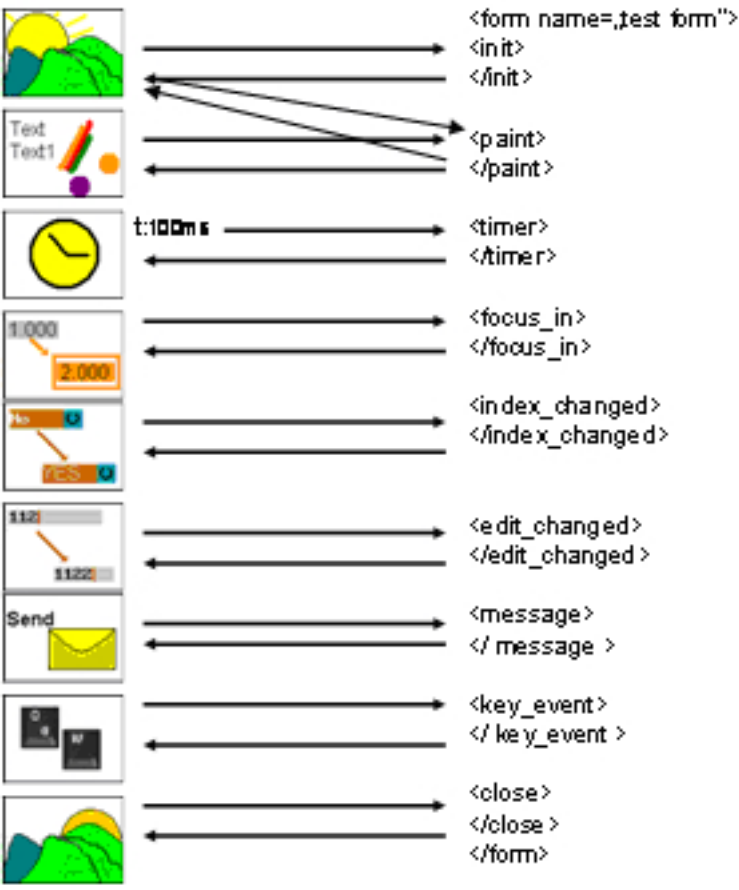


Figure 1-7 Menu structure

Tag identifier	Meaning
FORM	This tag contains the description of a user dialog. Attributes: • color Background color of the dialog box (color coding, see Chapter, color coding) • name Identifier of the form • type <i>cycle</i> -attribute specifies a cycle form • xpos X-position of the top left corner of the dialog box (optional) • ypos Y position of the top left corner (optional) • width Extension in the X direction (in pixels) (optional) • height Extension in the Y direction (in pixels) (optional)
FORM continued	Dialog messages: • INIT • PAINT • TIMER • CLOSE • FOCUS_IN • INDEX_CHANGED • EDIT_CHANGED • KEY_EVENT • MESSAGE

Tag identifier	Meaning
FORM continued	

Tag identifier	Meaning
FORM continued	Syntax: <code><FORM name = "<dialog name>" color = "#ff0000"></code> Example: <code><FORM name = "R-Parameter"></code> <code><INIT></code> <code><DATA_ACCESS type = "true" /></code> <code><CAPTION>R - Parameter</CAPTION></code> <code><CONTROL name = "edit1" xpos = "322" ypos = "34" refvar = "nck/Channel/Parameter/R[1]" /></code> <code><CONTROL name = "edit2" xpos = "322" ypos = "54" refvar = "nck/Channel/Parameter/R[2]" /></code> <code><CONTROL name = "edit3" xpos = "322" ypos = "74"</code> <code></INIT></code> <code><PAINT></code> <code><TEXT xpos = "23" ypos = "34">R - Parameter 1</TEXT></code> <code><TEXT xpos = "23" ypos = "54">R - Parameter 2</TEXT></code> <code><TEXT xpos = "23" ypos = "74">R - Parameter 3</TEXT></code> <code></PAINT></code> <code></FORM></code>
INIT	Dialog box message The tag is executed immediately after the dialog box is generated. All the input elements and hotlinks for the dialog form should be created here.

Tag identifier	Meaning
KEY_EVENT	Dialog message The tag KEY_EVENT can be integrated in the form to evaluate keyboard events. The system sends the MF2 keyboard code to the active form if the tag is available in a form. If the variable \$actionresult is not set to zero, the system then subsequently processes the keyboard event. The keyboard code is provided in the variable \$keycode as an integer value. Example: The character entered into the variable exclude_key should be filtered-out of the input stream. <pre> <LET name="stream" type="string"/> <LET name="exclude_key" type="string"/> <FORM name = "keytest_form"> <INIT> <CONTROL name = "p1" xpos = "120" ypos = "84" width ="200" refvar="stream" hotlink="true" /> <CONTROL name = "p2" xpos = "160" ypos = "104" width ="8" refvar="exclude_key" hotlink="true" /> </INIT> <PAINT> <text xpos = "8" ypos = "84">data stream</text> <text xpos = "8" ypos = "104">exclude key</text> </PAINT> <KEY_EVENT> <LET name="excl_keycode" type="string"/> <OP>excl_keycode = exclude_key</OP> <type_cast name="excl_keycode" type="int" /> <PRINT text="%d %d">\$keycode, excl_keycode</PRINT> <IF> <CONDITION>\$keycode == excl_keycode</CONDITION> <THEN> <op> \$actionresult = 0</op> </THEN> </IF> </KEY_IVENT> </FORM> </pre>

Tag identifier	Meaning
MESSAGE	Dialog message If the Send_message operation is executed in the script, then the parser processes the tag message. Values P1 and P2 are provided in the variables \$message_par1 and \$message_par2 (see the "SEND_MESSAGE" tag). Syntax: <code><MESSAGE></code> <code></MESSAGE></code> Example: <code><LET name="user_selection" /></code> <code><SOFTKEY POSITION="3"></code> <code> <CAPTION>Set%Parameter</CAPTION></code> <code> <SEND_MESSAGE>1, 10</SEND_MESSAGE></code> <code></SOFTKEY></code> <code>...</code> <code>...</code> <code><FORM></code> <code>...</code> <code>...</code> <code><MESSAGE></code> <code><SWITCH></code> <code><CONDITION>\$message_par1</CONDITION></code> <code><CASE value="1"></code> <code><OP> user_selection = \$message_par2 </OP></code> <code>...</code> <code>...</code> <code></CASE></code> <code></SWITCH></code> <code></MESSAGE></code> <code>...</code> <code>...</code> <code></FORM></code>

Tag identifier	Meaning
SEND_MESSAGE	The tag sends a message with two parameters to the active form, which is processed in the tag message (see also MESSAGE). Syntax: <code><SEND_MESSAGE>p1, p2</SEND_MESSAGE></code> Example: <code><LET name="user_selection" /></code> <pre> <SOFTKEY POSITION="3"> <CAPTION>Set%Parameter</CAPTION> <SEND_MESSAGE>1, 10</SEND_MESSAGE> </SOFTKEY> <FORM> <MESSAGE> <SWITCH> <CONDITION>\$message_par1</CONDITION> <CASE value="1"> <OP> user_selection = \$message_par2 </OP> ... </CASE> </SWITCH> </MESSAGE> </FORM> </pre>
FOCUS_IN	Dialog box message The tag is called if the system places the focus on a control. In order to identify the control, the system copies the name of the control to variable \$focus_name and the value of the attribute item_data to variable \$focus_item_data. The system creates the variables automatically. This message can be used, for example, to output images depending on the focus position. Example: <code><focus_in></code> <code><PRINT text="focus on filed:%s, %d">\$focus_name,</code> <code>\$focus_item_data </PRINT></code> <code></focus_in></code>
PAINT	Dialog box message The tag is executed when the dialog box is displayed. All the texts and images which are to be displayed in the dialog box should be specified here. Further, the tag is executed if the system identifies that parts of the dialog box are to be redisplayed. For example, this can be initiated by closing high-level windows.
TIMER	Dialog box message The tag is executed cyclically. Each form is assigned a timer that initiates that the timer - tag is executed approx. every 100 ms.

Tag identifier	Meaning
CAPTION	The tag contains the title of the dialog box. This tag should be used within the INIT tag. The title bar can be divided into multiple columns. To divide the title bar, the "caption" tag must be programmed with the "define_section" attribute before the text is output. The "define_section" attribute specifies the number of columns The "property" attribute is used to define the length, starting position, and text alignment for each column. The position and length specification is a percentage value referring to the width of the form. The value of the "value" attribute indicates the column number (beginning with zero) <pre> <caption define_sections="3"> <property value="0" length="20" position="2" alignment="left" /> <property value="1" length="20" position="79" alignment="right" /> </caption> </pre> The text can then be assigned to the column by additionally specifying the "index" attribute with the column index. <pre> <caption index="0">column1</caption> <caption index="1">column2</caption> </pre> Syntax: <CAPTION>Titel</CAPTION> Example: <CAPTION>my first dialogue</CAPTION>
CLOSE	Dialog box message This tag is executed before the dialog box is closed.
CLOSE_FORM	The tag closes the active dialog. This instruction is only necessary if the dialog is opened by the MMC command and the user is offered a softkey function to close the dialog. Generally, dialogs are automatically managed and do not have to be explicitly closed. Syntax: <CLOSE_FORM/> Example: <softkey_ok> <caption>OK</caption> <CLOSE_FORM /> <navigation>main_menu</navigation> </softkey_ok>

Tag identifier	Meaning
CONTROL	The tag is used to generate control elements. Syntax: <code><CONTROL name = "<control name>" xpos = "<X position>" ypos = "<Y position>" refvar = "<NC variable>" hotlink = "true" format = "<format>" /></code> Attributes: • name Identifier of the field. The identifier simultaneously represents a local variable, and must not be used a multiple number of times in the form. • xpos X position of the top left corner • ypos Y position of the top left corner • fieldtype Field type If no type is specified, the field is set as an edit field. – edit Data can be changed – readonly Data cannot be changed • combobox The field displays the corresponding identifiers instead of numerical values. If the field type "combobox" is selected, then the expressions to be displayed must also be assigned to the field. The <ITEM> TAG should be used for this purpose. The combo box saves the index of the currently selected text in the variable belonging to the control (see the attribute refvar). – progressbar A progress bar with a value range of 0 to 100 appears. The valley value and peak value properties can be used to adapt the value range to the data to be displayed.

Tag identifier	Meaning
CONTROL continued	• fieldtype – listbox The field type generates an empty list box control. Using the tag <ITEM> a list box element can be inserted in the list box. The ITEM attribute value allows this element to be assigned a unique value. For example, this can be used to identify the element. Parameters width and height specify the width and height of the list box. After the control has been created, additional list box elements can be inserted using the functions AddItem, InsertItem or LoadItem. After the control has been created, additional list box elements can be inserted using the function AddItem or InsertItem. The parameter itemdata is not evaluated for this control. – itemlist The field type generates a static control, which displays the corresponding identifier instead of numerical values. The <ITEM> tag can be used to assign an identifier to the field. • item_data A user-specific integer value can be assigned to the attribute. This value is given as part of the FOCUS_IN message for identifying the focus field. • refvar Identifier of the reference variable that can be linked to the field (optional). • hotlink = "TRUE" " If the value of the reference variable changes, then the field is automatically updated (optional). • format The attribute defines the display format of the specified variable. Formatting data, see print-Tag (optional).

Tag identifier	Meaning
CONTROL continued	Attributes: • time specifies the data refresh rate (optional). The default value is 200 ms. The following specifications are possible: – super fast Refresh time < 100 ms – fast Refresh time approx. 100 ms – normal Refresh time approx. 200 ms – slow Refresh time approx. 500 ms • color_bk The attribute sets the background color of the control. • color_fg The attribute sets the foreground color of the control. ("color coding" see Chapter "Color coding (Page 40)") • display_format The attribute defines the processing format of the specified variable. This attribute must be used when accessing a PLC float variable, as the access is realized by reading a double word. The following data formats are permitted: – FLOAT – INT – DOUBLE – STRING Assigning expressions (e.g. text or graphic element to be displayed) to a list box, graphics box or combo box: Syntax: <pre><ITEM>Expression</ITEM> <ITEM value ="<Value>">Expression</ITEM></pre>

Tag identifier	Meaning
CONTROL continued	Example: <pre><CONTROL name = "button1" xpos = "10" ypos = "10" fieldtype = " combobox "> <ITEM>text1</ITEM> <ITEM>text2</ITEM> <ITEM>text3</ITEM> <ITEM>text4</ITEM> </CONTROL></pre> If any integer value is to be assigned to an expression, the attribute value = "value" should be added to the tag. Rather than consecutive numbers, the control variable now contains the item's assigned value. Example: <pre><CONTROL name = "button1" xpos = "10" ypos = "10" fieldtype = " combobox "> <ITEM value = "10">text1</ITEM> <ITEM value = "20">text2</ITEM> <ITEM value = "12">text3</ITEM> <ITEM value = "1">text4</ITEM> </CONTROL></pre> Example of a progress bar: <pre><CONTROL name = "progress1" xpos = "10" ypos = "10" width = "100" fieldtype = "progressbar" hotlink = "true" refvar = "nck/Channel/ GeometricAxis/actProgPos[1]"> <PROPERTY min = "0" /> <PROPERTY max = "1000" /> </CONTROL></pre> Example, list box: <pre><let name="item_string" type="string"></let> <let name="item_data" ></let></pre> <pre><CONTROL name="listbox1" xpos = "360" ypos="150" width="200" height="200" fieldtype="listbox" /></pre> • Adding elements: Elements are added using the function additem or loaditem. • Deleting the content: The content is deleted using the function empty. <pre><op> item_string = _T"text1\\n" </op> <function name="control.additem">_T"listbox1", item_string, item_data </function> <op> item_string = _T"text2\\n" </op> <function name="control.additem">_T"listbox1", item_string, item_data </function></pre>
CONTROL continued	Example itemlist: <pre><CONTROL name = "itemlist1" xpos = "10" ypos = "10" fieldtype = " itemlist"> <ITEM value = "10">text1</ITEM> <ITEM value = "20">text2</ITEM> <ITEM value = "12">text3</ITEM> <ITEM value = "1">text4</ITEM> </CONTROL></pre>

Tag identifier	Meaning
HELP_CONTEXT	This tag defines the help topic to be called. It should be programmed in the INIT block. The name specified in the attribute is supplemented by the prefix XmlUserDlg_ and is transferred to the help system. The associated structure of the help file should be taken from the topic - generating an online help. Sequence when activating the help system: 1. Press the "Info" key. 2. The dialog supplies the expression "my_dlg_help". 3. Parser converts the expression into "XmlUserDlg_my_dlg_help" . 4. Activating the help system. 5. Submitting the search term "XmlUserDlg_my_dlg_help". Syntax: <code><HELP_CONTEXT name="<context name>" /></code> Example: <pre> ... <INIT> ... <CAPTION>my dialogue</CAPTION> <HELP_CONTEXT name="my_dlg_help" /> ... </INIT> </pre>
DATA_ACCESS	The tag controls the behavior of the dialog forms when user inputs are being saved. The behavior should be defined within the INIT tag. If the tag is not used, inputs are buffered in each case. Exception: Controls for which the hotlink attribute is set to true are always written to and read directly. Attribute:: • type = "TRUE" – the input values are not buffered. The dialog form copies the input values to the reference variables directly. • type = "FALSE" – the values are only copied to the reference variable with the UPDATA_DATA type = "FALSE" tag. Example: <code><DATA_ACCESS type = "true" /></code>

Tag identifier	Meaning
EDIT_CHANGED	Dialog box message This tag is called if the contents of an edit control have changed. To identify the control, the system copies the name of the control into variable \$focus_name and the value of the attribute item_data into variable \$focus_item_data. The system creates the variables automatically. Example: <pre><EDIT_CHANGED> <print text="index changed filed:%s, %d"> \$focus_name, \$focus_item_data </print> </EDIT_CHANGED></pre>
INDEX_CHANGED	Dialog box message The tag is called, if the operator changes the selection of a combo box. To identify the control, the system copies the name of the control into the variable \$focus_name and the value of the attribute item_data into the variable \$focus_item_data. The system creates the variables automatically. Note: A reference variable assigned to the control, has not been aligned to the control variable at this point in time and contains the index of the previous selection of the combo box. Example: <pre><INDEX_CHANGED> <print text="index changed filed:%s, %d"> \$focus_name, \$focus_item_data </print> </INDEX_CHANGED></pre>
MENU	The tag defines a menu containing the softkey description and the dialog to be opened. Attribute:: • name Menu name Syntax: <pre><MENU name = "<menu name>"> ... <open_form ...> ... <SOFTKEY ...> </SOFTKEY> </MENU></pre>

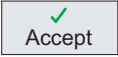
Tag identifier	Meaning
NAVIGATION	This tag defines the menu to be called. This tag can only be set within a softkey block. Syntax: <code><NAVIGATION>menu name</NAVIGATION></code> Example: <pre> <menu name = "main"> <softkey POSITION="1"> <caption>sec. form</caption> <navigation>sec_menu</navigation> </softkey> </menu> <menu name = "sec_menu"> <open_form name = "sec_form" /> <softkey_back> <navigation>main</navigation> </softkey_back> </menu> </pre>
OPEN_FORM	The tag opens the dialog form given under the name. Attribute:: name Name of the dialog form Syntax: <code><OPEN_FORM name = "<form name>" /></code> Example: <pre> <menu name = "main"> <open_form name = "main_form" /> <softkey POSITION="1"> <caption>main form</caption> <navigation>main</navigation> </softkey> </menu> <form name="main_form"> <init> </init> <paint> </paint> </form> </pre>

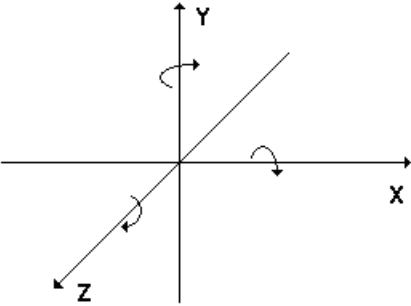
Tag identifier	Meaning
PROPERTY	This tag can be used to define additional properties for an operator control. Attributes: • max = "<maximum value>" • min = "<minimum value>" • default = "<pre-assignment>" • factor = "conversion factor" • color_bk = "<background color coding>" • color_fg = "" • password = "<true>" - entered character is displayed with "*" • multiline = "<true>" - permits multi-line inputs in an edit control • disable = "<true/false>" - locks/permits the input in an edit control • transparent= "Transparent color of a bitmap" Color coding (for details on color coding, see chapter, Color coding) • tooltip = information text is displayed if the cursor is set to the control. The syntax is: <property tooltip="information text" /> Example: <pre> <CONTROL name = "progress1" xpos = "10" ypos = "10" width = "100" fieldtype = "progressbar" hotlink = "true" refvar = "nck/Channel/ GeometricAxis/actProgPos[1]"> <PROPERTY min = "0" /> <PROPERTY max = "1000" /> </CONTROL> <CONTROL name = "edit1" xpos = "10" ypos = "10"> <PROPERTY min = "20" /> <PROPERTY max = "40" /> <PROPERTY default = "25" /> </CONTROL> </pre>

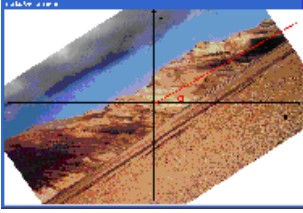
Tag identifier	Meaning
SOFTKEY	The tag defines the properties and responses of a softkey. Attributes: • position Number of the softkey. 1-8 horizontal softkeys, 9-16 vertical softkeys The following attributes become effective from: • type Defines the property of the softkey. user_controlled - The script defines how the softkey is displayed toggle_softkey - The softkey is displayed alternating between pressed and not pressed • refvar Should only be used in conjunction with toggle_softkey . Reference variable, into which the actual softkey property is copied. A variable, type "String" should be specified, which includes the properties pressed, not pressed or locked (see tag state). • picture Using the attribute, a bitmap can be output left justified on the softkey. The complete path name should be specified. The number of text characters that can be displayed is reduced to the width of the bitmap. The following additional actions can be defined within the softkey block: • picturealignment The image orientation is specified by this attribute. The image is aligned with the left side of the softkey by default. The following values can be specified for alignment: – top Top edge – bottom Bottom edge – left Left edge – right Right edge – center Centered • caption Softkey text • state Should only be used in conjunction with user_controlled . The tag assigns the required softkey display to the system. Syntax: <state type="<state>" /> The following strings can be specified: – notpressed The softkey is displayed as being not pressed. – pressed The softkey is displayed as being pressed.

Tag identifier	Meaning
	- disabled The softkey is locked and is displayed in gray. • navigation • update_controls • function
SOFTKEY continued	Syntax: Standard softkey: <code><state type="<softkey state>" /></code> <code><softkey position = "<1>"></code> <code></softkey></code> or Script-controlled softkey: <code><softkey position = "<1>" type="<user_defined>" ></code> <code><state type="<softkey state>" /></code> <code></softkey></code> or Toggle softkey: <code><softkey position = "<1>" type="<toggle_softkey>" refvar="<variable name>" ></code> <code></softkey></code> Example: <pre> <let name="define_sk_type" type="string">PRESSED</let> <let name="sk_type">1</let> <softkey POSITION="1" type="user_controlled" > <caption>Toggle%NSK</caption> <if> <condition>sk_type == 0 </condition> <then> <op> sk_type = 1 </op> <op> define_sk_type = _T"PRESSED" </op> </then> <else> <op> define_sk_type = _T"NOTPRESSED" </op> <op> sk_type = 0 </op> </else> </if> <state type="\$\$\$define_sk_type" /> </softkey> </pre>

Tag identifier	Meaning
SOFTKEY continued	Example: or <pre><let name="curr_softkey_state" type="string">PRESSED</let> </softkey> <softkey POSITION="3" type="toggle_softkey" refvar="curr_softkey_state"> <caption>Toggle%nSK</caption> ... </softkey></pre>  
SOFTKEY_OK	The tag defines the response of the softkey "OK".  The following additional actions can be defined within the softkey block: • navigation • update_controls • function Syntax: <pre><SOFTKEY_OK> </SOFTKEY_OK></pre>
SOFTKEY_CANCEL	The tag defines the response of the softkey "Cancel".  The following additional actions can be defined within the softkey block: • navigation • update_controls • function Syntax: <pre><SOFTKEY_CANCEL> </SOFTKEY_CANCEL></pre>

Tag identifier	Meaning
SOFTKEY_BACK	The tag defines the response of the softkey "Back".  The following additional actions can be defined within the softkey block: • navigation • update_controls • function Syntax: <SOFTKEY_BACK> </SOFTKEY_BACK>
SOFTKEY_ACCEPT	The tag defines the response of the softkey "Accept".  The following additional actions can be defined within the softkey block: • navigation • update_controls • function Syntax: <SOFTKEY_ACCEPT> </SOFTKEY_ACCEPT>
TEXT	The tag is used to display a text in the specified position. If an alarm number is used, the dialog box displays the text which is saved for the number. Syntax: <TEXT xpos = "<X position>" ypos = "<Y position>"> Text </TEXT> Attributes: • xpos X position of the top left corner • ypos Y position of the top left corner • color Text color (color coding, see Chapter Color coding (Page 40)) Value: Text to be displayed

Tag identifier	Meaning
IMG	The tag is used to display an image in the specified position. The BMP and PNG image formats are supported. Syntax: <pre><IMG xpos = "<X position>" ypos = "<Y position>" name = "<name>" /> </pre> Attributes: • xpos X position of the top left corner • ypos Y position of the top left corner • name complete path name • transparent Transparent color of the bitmap (see Chapter "Color coding"); • xrot, yrot, zrot (only for 802Dsl/808D) In the axes X, Y and Z, the image rotates around a point of rotation. • xref, yref, zref (only for 802Dsl / 808D) Define point of rotation If a point of rotation is not specified, then the image rotates around the image center point. In this case, the Z plane is 0. 

Tag identifier	Meaning
IMG Continued	Example: The image is rotated through 34 degrees around the Z axis: <pre> </pre>  Optional: If the image display is to differ from the original size, the dimensions can be defined using the attributes width and height. • width Width in pixels • height Height in pixels Examples: <pre> </pre>
BOX	The tag draws a rectangle at the specified position, colored as indicated. Syntax: <pre><BOX xpos = "<X position>" ypos = "<Y position>" width = "<X extension>" height = "<Y extension>" color = "<Color code>" /></pre> Attributes: • xpos X position of the top left corner • ypos Y position of the top left corner • width Extension in X direction (in pixels) • height Extension in Y direction (in pixels) • color Color coding (for details on color coding, see chapter, Color coding)

Tag identifier	Meaning
FUNCTION	Function call The tag executes the function body, which is specified under the attribute "name". Attributes: • name = "Name of the function body" • return = "Variable name for saving the result of the function" Values: List of variables to be transferred to the function body. The variables must be separated by a comma. A maximum of 10 parameters can be transferred. It is also possible to specify constants or text expressions as call parameters. The identifier _T should be placed at the start as a means of identifying text terms. Syntax: <pre><FUNCTION name = "<function name>" /></pre> Calling function expects a return value <pre><FUNCTION name = "<function name>" return = "<Variablenname>" /></pre> Parameter transfer <pre><FUNCTION name = "<function name>"> var1, var2, var3 </FUNCTION> <FUNCTION name = "<function name>"> _T"Text", 1.0, 1 </FUNCTION></pre> Examples: See "FUNCTION_BODY".

Tag identifier	Meaning
FUNCTION_BODY	Function body The tag contains the function body of a subfunction. The function body needs to be programmed within the DialogGui tag. Attributes: • name = "Name of the function body" • parameter = "Parameter list" (optional) The attribute lists the transfer parameters that are required. The parameters must be separated by a comma. When the function body is called, the values of the parameters specified in the function call are copied to the transfer parameters listed. • return = "true" (optional) If the attribute is set to true then the local variable \$return is created. The function's return value which is forwarded to the calling function on quitting the function should be copied to this variable. Syntax: Function body without parameter <pre><FUNCTION_BODY name = "<function name>"> </ FUNCTION_BODY></pre> Function body with parameter <pre><FUNCTION_BODY name = "<function_name>" parameter = "<p1, p2, p3>"> ... <LET name = "tmp"></LET> <OP> tmp = p1 </OP> ... </FUNCTION_BODY></pre> Function body with return value <pre><FUNCTION_BODY name = "<function_name>" parameter = "<p1, p2, p3>" return = "true"> ... <LET name = "tmp"></LET> <OP> tmp = p1 </OP> ... <OP> \$return = tmp </OP> </FUNCTION_BODY></pre>

Tag identifier	Meaning
FUNCTION_BODY continued	Example: <pre><function_body name = "test" parameter = "c1,c2,c3" return = "true"> <LET name = "tmp">0</LET> <OP> tmp = c1+c2+c3 </OP> <OP> \$return = tmp </OP> </function_body> <LET name = "my_var"> 4 </LET> <function name = "test" return = " my_var "> 2, 3,4</function> <print text = "result = %d"> my_var </print></pre>

Tag identifier	Meaning
REQUEST	The tag is used to add a variable to the cyclic reading service (hotlink). As a consequence, the access time to variables, which are not linked to the control, is reduced. If a function is to be called automatically when a value changes, then the name of the function should be specified as an additional attribute. This tag is only processed within the INIT operation. Attributes: • name Address identifier • function Function name Syntax: <pre><REQUEST name = "<NC-Variable>" /> or <REQUEST name = "<NC-Variable>" function="<function name>"/></pre> Example: <pre><request name ="plc/mb10" /> or <function_body name="my_function" > <print text="value changed" /> </function_body> <request name ="plc/mb10" function="my_function"/></pre>
UPDATE_CONTROLS	The tag runs a comparison between the operator controls and the reference variables. Attribute:: • type The attribute defines the direction of the data comparison. = TRUE – data is read from the reference variables and copied to the operator controls. = FALSE – Data is copied from the operator controls to the reference variables. Syntax: <pre><UPDATE_CONTROLS type = "<Direction>"/></pre> Example: <pre><SOFTKEY_OK> < UPDATE_CONTROLS type="false"/> </SOFTKEY_OK></pre>

1.7 Generating user menus

1.7.1 Creating processing cycle forms

The cycle support function allows the automatic creation and decompilation of a cycle call through the form dialog.

To manage this functionality, the following tags are available:

- **NC_INSTRUCTION**
- **CREATE_CYCLE**

To mark a cycle form, in the **FORM** tag, the attribute **type** should be specified with the value **cycle**. This marking allows the **NC_INSTRUCTION** to be processed.

Example

```
<FORM name = "cycle100_form" type= "CYCLE">
...
...
</FORM>
```

The **NC_INSTRUCTION** tag contains the cycle call to be generated. All cycle parameters should be reserved using space retainers.

Example

```
<FORM name = "cycle100_form" type= "CYCLE">
<NC_INSTRUCTION refvar= "cyc_string" >Cycle100 ($p1, $p2, $p3)</
NC_INSTRUCTION>
...
...
...
</FORM>
```

The **CREATE_CYCLE** tag prepares the values saved in the space retainer variables and generates the NC instruction.

This is then copied to the specified variable.

Tag identifier	Description
NC_INSTRUCTION	This tag is used to define the NC instruction to be generated. All listed cycle parameters are automatically created as string variables of the FORM and are available to the FORM. Precondition: The FORM attribute type is set to the value CYCLE. Attribute: • refvar If the tag is assigned a reference variable, all parameters are pre-assigned with the values from the NC block saved in the reference variables. Syntax: <pre><NC_INSTRUCTION> NC instruction with space retainers </NC_INSTRUCTION></pre> Example: <pre><let name="cyc_string" type="string"> Cycle100(0, 1000, 5)</let> <FORM name = "cycle100_form" type= "CYCLE"> <NC_INSTRUCTION refvar= "cyc_string" >Cycle100(\$p1, \$p2, \$p3)</ NC_INSTRUCTION> </FORM></pre>

Tag identifier	Description
CREATE_CYCLE	The tag generates an NC block, whose syntax is defined by the value of the NC_INSTRUCTION tag. Before generating the NC instruction, the parser calls the CYCLE_CREATE_EVENT tag of the FORM. This tag can be used to calculate the cycle parameters. Syntax: <CREATE_CYCLE/> Attribute: • refvar If the tag is assigned a reference variable, the NC instruction is copied to this variable. Example: <LET name="cyc_string" type="string"> Cycle100(0, 1000, 5)</LET> <pre><SOFTKEY_OK> <caption>OK</caption> <CREATE_CYCLE /> <close_form /> <navigation>main_menu</navigation> </SOFTKEY_OK></pre> or <pre><SOFTKEY_OK> <caption>OK</caption> <CREATE_CYCLE refvar= "cyc_string" /> <close_form /> <navigation>main_menu</navigation> </SOFTKEY_OK></pre>

1.7.2 Substitution characters

The system offers the option of defining control properties (attribute values) for the runtime. In order to use this function, the desired property must be set in a local variable and the variable name must be transferred to the tag as an attribute value preceded by the **character \$**.

If the tag expects a string as attribute value or value, the \$\$\$ characters must be placed in front of the variable name.

Example:

```
<let name="my_ypos">100</let>
<let name="field_name" type="string"></let>

<control name = "edit1" xpos = "322" ypos = "$my_ypos" refvar="nck/
Channel/Parameter/R[1]" />

<op>my_ypos = my_ypos +20 </op>

<control name = "edit2" xpos = "322" ypos = "$my_ypos" refvar="nck/
Channel/Parameter/R[2]" />

<print name = " field_name" text="edit%d">3</print>
<op>my_ypos = my_ypos +20 </op>

<control name = "$field_name" xpos = "322" ypos = "$my_ypos"
refvar="nck/Channel/Parameter/R3]" />

<caption>$$$field_name</caption>
```

1.8 Addressing components

Address identifiers for the desired data must be created to address NC variables, PLC blocks or drive data. An address consists of the subpaths **component name** and **variable address**. A slash should be used as a separating character.

1.8.1 PLC addressing

Addressing the PLC starts with the path section **plc**.

Table 1-3 The following addresses are permissible:

DBx.DB(f)	Data block
I(f)x	Input
Q(f)x	Output
M(f)x	Bit memory
V(f)x	Variable

DBx.DBXx.b	Data block
Ix.b	Input
Qx.b	Output
Mx.b	Bit memory
Vx.b	Variable

Table 1-4 Data format f:

B	Byte
W	Word
D	Double word

Data format identification is not applicable to bit addressing.

Address **x**:

Valid S7-200 address identifier

Bit addressing:

b – Bit number

Examples:

```
<data name = "plc/mb170">1</data>
```

```
<data name = "i0.1"> 1 </data>
```

```
<op> "m19.2" = 1 </op>
```

1.8.2 Addressing NC variables

Addressing the NC variables starts with the path section **nck**.

This section is followed by the data address; its structure should be taken from the SINUMERIK 802D sl Parameter Manual.

Example:

```
<LET name = "tempStatus"></LET>
<OP> tempStatus = "nck/channel/state/chanstatus" </OP>
```

1.8.3 Generating NC/PLC addresses during the runtime

It is possible to generate an address identifier during runtime.

In this case, the content of a string variable is used as address in an operation statement as well as in the nc.cap.read and nc.cap.write functions.

Observe the following for this type of addressing mode:

- Write the variable names in quotation marks.
- Use three '\$' characters as prefix for variable names.

Syntax:

```
"$$$variable name"
```

Example:

```
<LET name="var_adr" type="string"></LET>
<PRINT name="var_adr" text="VB9000%d"> 2000</PRINT>

or
<PRINT name="var_adr" text="DB9000.DBW%d"> 2000</PRINT>
<OP> "$$$var_adr" = 1 </OP>
```

1.8.4 Addressing drive components

Addressing the drive components starts with the path section **drive**.

Then the drive device is specified:

CU

DC

The parameter to be set is added to this section.

Example:

```
<LET name="r0002_content"></LET>
<LET name="p107_content"></LET>

<!-- Reading of value r0002 on the CU -->
```

1.8 Addressing components

```
<OP> r0002_content = "drive/cu/r0002" </OP>
<OP> r0002_content = "drive/cu/r0002[CU1]" </OP>

<!-- Reading of value r0002 on NX1 ->
<OP> r0002_content = "drive/cu/r0002[CU2]" </OP>

<!-- Reading of value p107[0] on the CU ->
<OP> p107_content = "drive/cu/p107[0]" </OP>
<PRINT text="%d"> p107_content </PRINT>

<!-- Reading of value p107[0] on the CU ->

<OP> p107_content = "drive/cu/p107[0, CU1]" </OP>
<PRINT text="%d"> p107_content </PRINT>

<!-- Reading of value p107[0] on the NX1 ->

<OP> p107_content = "drive/cu/p107[0, CU2]" </OP>
<PRINT text="%d"> p107_content </PRINT>
```

Addressing the drive objects:

To address individual objects, the desired object should be entered in square brackets after the parameter.

Note

The drive object number differs from the numbering used in the drive dialog, since the CU components, ALM, and all connected hubs are integrated with the continuous numbering.

The DO number can be established as follows:

- All connected drive objects are listed in field p978 of the relevant CU corresponding to their slot number.
 - It is a question of establishing the field index for the desired slot and adding one to this number. This value is the DO index needed for addressing purposes.
-

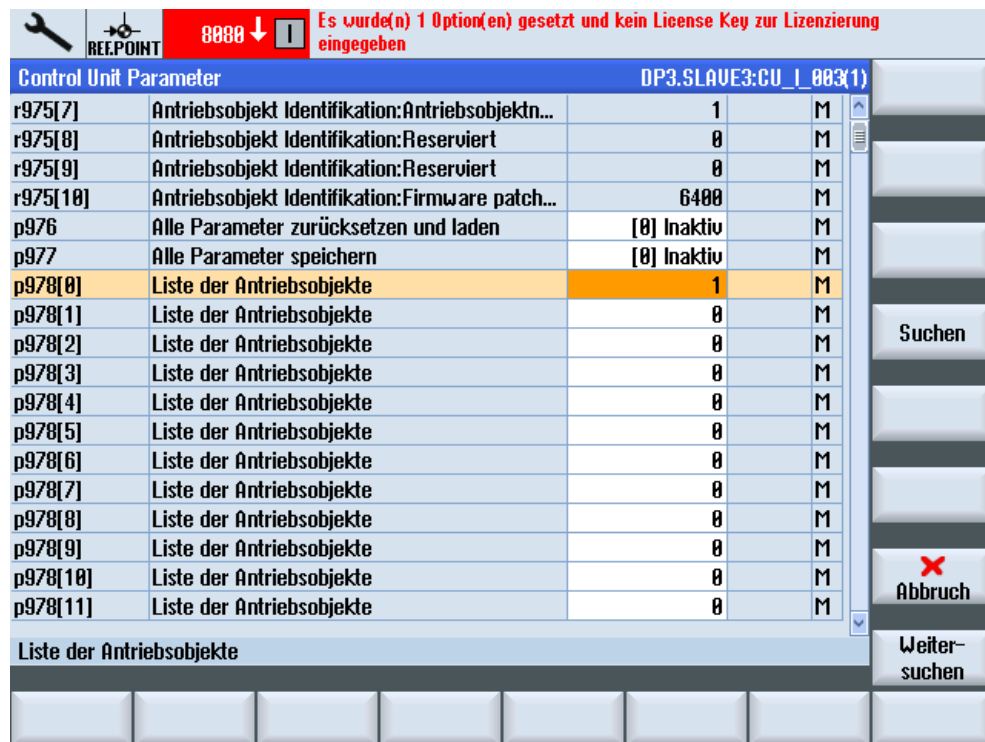


Figure 1-8 Drive parameter p0978 [...] at the control

Parameter number[do<DO-index>]

Example:

p0092[do1]

Alternatively, the drive index can be read from a local variable using \$<variable name> "substitution characters".

z.B. DO\$local variable

Example:

```
<DATA name ="drive/cu/p0092">1</DATA>
```

```
<DATA name ="drive/dc/p0092[do1] ">1</DATA>
```

Indirect addressing:

```
<LET name = "driveIndex" 0 </LET>
```

```
<OP> driveIndex = $ctrlout_module_nr[0, AX1] </OP>
```

```
<DATA name ="drive/dc[do$driveIndex]/p0092">1</DATA>
```

1.8.5 Addressing machine and setting data

Drive and setting data is identified by the character \$ followed by the name of the data.

Machine data:

\$Mx_<name[index, AX<axis_number>]>

Setting data:

\$Sx_<name[index, AX<axis_number>]>

x:

N – General machine or setting data

C – Channel-specific machine or setting data

A – Axis-specific machine or setting data

Index:

For a field, the parameter indicates the index of the data.

AX<axis_number>:

The required axis (<axis_number>) has to be specified for axis-specific data.

Alternatively, the axis index can be read from a local variable using \$<variable name> "substitution characters".

e.g. AX\$localvariable

Example:

```
<DATA name = "$MN_AXCONF_MACHAX_NAME_TAB[0] ">X1</DATA>
```

Direct addressing of the axis:

```
<DATA name = "$MA_CTRLOUT_MODULE_NR[0, AX1] ">1</DATA>
```

...

...

Indirect addressing of the axis:

```
<LET name = "axisIndex"> 1 </LET>
```

```
<DATA name = "$MA_CTRLOUT_MODULE_NR[0, AX$axisIndex] ">1</DATA>
```

1.8.6 Addressing user data

Addressing user data starts with the path section **gud**, followed by the GUD name.

For a field, after the name, the required field index should be specified in square brackets.

Example:

```
<DATA name ="gud/syg_rm[0]"  
<OP>"gud/syg_rm[0]" 0 10 </op>
```

Addressing the global user data

Addressing starts with the path section **gud**, followed by the specification of the area **CHANNEL**. This address section is followed by the specification of the GUD areas:

GUD areas	Assignment
sgud	Siemens GUD
mgud	Machine manufacturer GUD
ugud	User GUD

Then enter the GUD name. If an array is to be addressed, the name is followed by the array subscript in square brackets.

Example:

```
<data name ="gud/channel/mgud/syg_rm[0]">1</data>  
<op>"gud/channel/mgud/syg_rm[0]" = 5*2 </op>
```

1.9 Predefined functions

The script language offers various string processing and standard mathematical functions.

The function names listed below are reserved and cannot be overloaded.

Function name	Description
Ncfunc cap read	The function copies a value from the specified address into a local variable. If the read operation was error-free, then the return variable contains the value zero. Contrary to the operation instruction, in the event of a fault, this function does not interrupt the processing of the script operations. Syntax: <pre><function name="ncfunc.cap.read" return="error"> lokale variable, "address"</function></pre> Example: <pre><let name="error"></let> <function name="ncfunc.cap.read" return="error"> 3, "drive/cu/p0009"</function> <if> <condition>error != 0</condition> <then> <break /> </then> </if></pre>
Ncfunc cap write	The function writes a value into the specified variable. If the write operation was error-free, then the return variable contains the value zero. Contrary to the operation instruction, in the event of a fault, this function does not interrupt the processing of the script operations. Syntax: <pre><function name="ncfunc.cap.write" return="error"> local variable or constant, "address"</function></pre> Example: <pre><let name="error"></let> <function name="ncfunc.cap.write" return="error"> 0, "drive/cu/p0009"</function> <if> <condition>error != 0</condition> <then> <break /> </then> </if></pre>

Function name	Description
Ncfunc PI-Service	Jobs can be transferred to the NCK using the program invocation (PI) service. If the service has been executed error-free, the function returns the value 1 in the return variable. Manipulation of the tool list _N_CREATO - Create tool _N_DELETEO - Delete tool _N_CREACE - Create tool cutting edge _N_DELECE - Delete tool cutting edge Activation of work offsets _N_SETUFR - Activates the actual user frame _N_SETUDT - Activates the actual user data Block search _N_FINDBL - Activate block search _N_FINDAB - Cancel block search Syntax: <pre><function name="ncfunc.pi_service" return="return var"> pi name, var1, var2, var3, var4, var5 </ function></pre> Attributes: • name - function name • return- Name of the variable in which the execution result is saved – Value == 1 – job executed successfully – Value == 0 – faulty job Tag values: • pi name - Name of the PI service (string) • var1 to var5 - PI specific arguments

Function name	Description
Ncfunc PI-Service Continued	Arguments: • _N_CREATO var1 - Tool number • _N_DELETEO var1 - Tool number • _N_CREATEC var1 - Tool number var2 - Cutting edge number • _N_DELETEC var1 - Tool number var2 - Cutting edge number • _N_SETUFR No arguments • _N_SETUDT var1- User data area to be activated – 1 - Tool offset data – 2 - Active basic frame – 3 - Active adjustable frame • _N_FINDBL var1 - Search mode – 2 - Search with contour calculation – 4 - Search for the block end point – 1 - Block search without calculation. • _N_FINDAB No arguments Example: • Creating a tool – tool number 3 <pre><function name="ncfunc.pi_service">_T"_N_CREATO", 3</function></pre> • Delete cutting edge 1 of tool 5 <pre><function name="ncfunc.pi_service">_T"_N_DELETEC", 5, 1</function></pre>

Function name	Description
Ncfunc display resolution	This function supplies the conversion rule for floating point numbers defined in the control. A string variable must be provided as variable. See also display machine data MD203 DISPLAY_RESOLUTION and MD204 DISPLAY_RESOLUTION_INCH Syntax: <pre><function name="ncfunc.displayresolution" return="dislay_res" /></pre> Example: <pre><let name="dislay_res" type="string"></let> ... <function name="ncfunc.displayresolution" return="dislay_res" /> <control name = "cdistToGo" xpos = "210" ypos = "156" refvar="nck/Channel/GeometricAxis/ progDistToGo[2]" hotlink="true" height="34" fieldtype="readonly" format="\$\$\$dislay_res" time="superfast" color_bk="#ffffff"/></pre>
Ncfunc bico to int	The function converts a string specified in the BICO format into an integer value. (see SINAMICS). Syntax: <pre><function name="ncfunc.bicotoint" return="integer variable">bico-string</function></pre> Example: <pre><let name="s_np0480_0" type="string"></let> <let name="i_p0480_0">0</let> <function name="ncfunc.bicotoint" return="i_p0480_0">s_np0480_0</function></pre>
Ncfunc int to bico	The function converts an integer value into a BICO format string. (see SINAMICS). Syntax: <pre><function name="ncfunc.inttobico" return="string variable">integer variable</function></pre> Example: <pre><function name="ncfunc.inttobico" return="s_p0480_0">"drive/dc/p0480[0, D02]"</ function></pre>

Function name	Description
Ncfunc is bico str valid	This function returns the value zero if it involves a string specified in the BICO format. (see SINAMICS) Syntax: <pre><function name="ncfunc.isbicostrvalid" return="integer variable">string variable</function></pre> Example: <pre>... <let name="s_np0480_0" type="string"></let> <control name = "cp0480_0" xpos = "402" ypos = "76" hotlink="true" refvar="s_np0480_0" > <property item_data="4001" /> </control> <function name="ncfunc.isbicostrvalid" return="valid">cp0480_0</function></pre>
Ncfunc password	This function sets or deletes a password level. • Set password: The password should be specified for the required password level as parameter. • Delete password: A blank string deletes the password level. Syntax: <pre><function name="ncfunc.password">password </function></pre> Example: <pre><let name="password" type="string"></let> <function name="ncfunc.password" > password </function> <function name="ncfunc.password" > _T"CUSTOMER" </function></pre> Delete password: <pre><function name="ncfunc.password" > _T"" </function></pre>

Function name	Description
Control form color	This function provides the text or background color of the dialog box. Range: - BACKGROUND – request color value of the background - TEXT – request color value of the text (foreground) Syntax: <pre><function name="control.formcolor" return="variable">_T"range"</function></pre> Example: <pre><let name="bk_color"></let></pre> <pre><function name="control.formcolor" return="bk_color">_T"BACKGROUND"</function></pre>
Control local time	The function copies the local time in a field with 7 array elements. The name of the variable is expected as call parameter. The following is stored in an array element: - Index 0 - year - Index 1 - month - Index 2 - weekday - Index 3 - day - Index 4 - hour - Index 5 - minute - Index 6 - second Syntax: <pre><function name ="control.localtime">_T"time_array"</function></pre> Example: <pre><!-- index 0 = Year 1 = Month 2 = Day of week 3 = Day 4 = Hour 5 = Minute 6 = Second --> <let name="time_array" dim="7" /></pre> <pre><function name ="control.localtime">_T"time_array"</function></pre>

Function name	Description
String to compare	Two strings are compared with one another from a lexicographical perspective. The function gives a return value of zero if the strings are the same, a value less than zero if the first string is smaller than the second string or a value greater than zero if the second string is smaller than the first string. Parameter: str1 - string str2 - comparison string Syntax: <pre><function name="string.cmp" return ="<int var>" > str1, str2 </function></pre> Example: <pre><let name="rval">0</let> <let name="str1" type="string">A brown bear hunts a brown dog.</let> <let name="str2" type="string">A brown bear hunts a brown dog.</let></pre> <pre><function name="string.cmp" return="rval"> str1, str2 </function></pre> Result: rval= 0

Function name	Description
String to compare without making a distinction between uppercase/lowercase	Two strings are compared from a lexicographical perspective (the comparison is not case-sensitive). The function gives a return value of zero if the strings are the same, a value less than zero if the first string is smaller than the second string or a value greater than zero if the second string is smaller than the first string. Parameter: str1 - string str2 - comparison string Syntax: <code><function name="string.icmp" return ="<int var>" > str1, str2 </function></code> Example: <code><let name="rval">0</let></code> <pre> <let name="str1" type="string">A brown bear hunts a brown dog.</let> <let name="str2" type="string">A brown Bear hunts a brown Dog.</let> <function name="string.icmp" return="rval"> str1, str2 </function> </pre> Result: rval= 0
String left	The function extracts the first nCount character from string 1 and copies this to the return variable. Parameter: str1 - string nCount - number of characters Syntax: <code><function name="string.left" return="<result string>"> str1, nCount </function></code> Example: <code><let name="str1" type="string">A brown bear hunts a brown dog.</let> <let name="str2" type="string"></let></code> <pre> <function name="string.left" return="str2"> str1, 12 </function> </pre> Result: str2="A brown bear"

Function name	Description
String right	The function extracts the last nCount character from string 1 and copies this to the return variable. Parameter: str1 - string nCount - number of characters Syntax: <code><function name="string.right" return="<result string>"> str1, nCount </function></code> Example: <code><let name="str1" type="string">A brown bear hunts a brown dog.</let> <let name="str2" type="string"></let></code> <code><function name="string. right " return="str2"> str1, 10 </function></code> Result: str2="brown dog."
String middle	The function extracts the specified number of characters from string 1, starting from the iFirst index, and copies these to the return variable. Parameter: str1 - string iFirst - start index nCount - number of characters Syntax: <code><function name="string.middle" return="<result string>"> str1, iFirst, nCount </function></code> Example: <code><let name="str1" type="string">A brown bear hunts a brown dog.</let> <let name="str2" type="string"></let></code> <code><function name="string. middle " return="str2"> str1, 2, 5 </function></code> Result: str2="brown"

Function name	Description
String length	The function gives the number of characters in a string. Parameter: str1 - string Syntax: <code><function name="string.length" return="<int var>"> str1 </function></code> Example: <code><let name="length">0</let></code> <code><let name="str1" type="string">A brown bear hunts a brown dog.</let></code> <code><function name="string.length" return="length"> str1 </function></code> Result: length = 31
Strings to replace	The function replaces all the substrings found with the new string. Parameter: string - string variable find string - string to be replaced new string - new string Syntax: <code><function name="<string.replace>"> string, find string, new string </function></code> Example: <code><let name="str1" type="string">A brown bear hunts a brown dog. </let></code> <code><function name="string.replace" > str1, _T"a brown dog" , _T"a big salmon"</function></code> Result: str1 = "A brown bear hunts a big salmon!"

Function name	Description
Strings to remove	The function removes all the substrings found. Parameter: string - string variable remove string - substring to be deleted Syntax: <function name="string.remove"> string, remove string </function> Example: <let name="index">0</let> <let name="str1" type="string">A brown bear hunts a brown dog. </let> <function name="string.remove" > str1, _T"a brown dog" </function> Result: str1 = "A brown bear hunts"
Strings to insert	The function inserts a string at the index specified. Parameter: string - string variable index - index (zero based) insert string - string to be inserted Syntax: <function name="string.insert"> string, index, insert string </function> Example: <let name="str1" type="string">A brown bear hunts. </let> <let name="str2" type="string">a brown dog</let> <function name="string.insert" > str1, 19, str2</function> Result: str1 = "A brown bear hunts a brown dog"

Function name	Description
String delete	The function deletes the defined number of characters starting from the start position specified. Parameter: string - string variable start index - start index (zero based) nCount - number of characters to be deleted Syntax: <code><function name="string.delete"> string, start index , nCount </function></code> Example: <code><let name="str1" type="string">A brown bear hunts. </let></code> <code><function name="string.delete" > str1, 2, 5</function></code> Result: str1 = "A bear hunts"
String find	The function searches the transferred string for the first match with the substring. If the substring is found, the function provides the index to the first character (starting with zero) or, failing this, -1. Parameter: string - string variable find string - string to be found startindex – start index (optional) Syntax: <code><function name="string.find" return="<int val>"> str1, find string </function></code> Example: <code><let name="index">0</let></code> <code><let name="str1" type="string">A brown bear hunts a brown dog. </let></code> <code><function name="string.find" return="index"> str1, _T"brown" </function></code> Result: Index = 2 or <code><function name="string.find" return="index"> str1, _T"brown", 1 </function></code>

Function name	Description
String reverse find	The function searches the transferred string for the last match with the substring. If the substring is found, the function provides the index to the first character (starting with zero) or, failing this, -1. Parameter: string - string variable find string - string to be found startindex – start index (optional) Syntax: <code><function name="string.reversefind" return="<intval>"> str1, find string </function></code> Example: <code><let name="index">0</let></code> <code><let name="str1" type="string">A brown bear hunts a brown dog. </let></code> <code><function name="string.reversefind" return="index"> str1, _T"brown" </function></code> Result: Index = 21 or <code><function name="string.reversefind" return="index"> str1, _T"brown", 10 </function></code> Result: Index = 2
String trim left	The function trims the starting characters from a string. Parameter: str1 - string variable Syntax: <code><function name="string.trimleft" > str1 </function></code> Example: <code><let name="str1" type="string"> test trim left</let></code> <code><function name="string.trimleft" > str1 </function></code> Result: str1 = "test trim left"

Function name	Description
String trim right	The function trims the closing characters from a string. Parameter: str1 - string variable Syntax: <code><function name="string.trimright" > str1 </function></code> Example: <code><let name="str1" type="string"> test trim right</let> <function name="string.trimright" > str1 </function></code> Result: str1 = "test trim right"
Sine	The function calculates the sine of the value transferred in degrees. Parameter: double - angle Syntax: <code><function name="sin" return="<double val>"> double</function></code> Example: <code><let name= "sin_val" type="double"></let> <function name="sin" return="sin_val"> 20.0 </function></code>
Cosine	The function calculates the cosine of the value transferred in degrees. Parameter: double - angle Syntax: <code><function name="cos" return="<double val>"> double</function></code> Example: <code><let name= "cos_val" type="double"></let> <function name="cos" return="cos_val"> 20.0 </function></code>

Function name	Description
Tangent	The function calculates the tangent of the value transferred in degrees. Parameter: double - angle Syntax: <function name="tan" return="<double val>"> double </function> Example: <let name= "tan_val" type="double"></let> <function name="tan" return="tan_val"> 20.0 </function>
ARCSIN	The function calculates the arcsine of the value transferred in degrees. Parameter: double - x in the range from -PI/2 to +PI/2 Syntax: <function name="arcsin" return="<double val>"> double </function> Example: <let name= "arcsin_val" type="double"></let> <function name="arcsin" return=" arcsin_val"> 20.0 </function>
ARCOS	The function calculates the arccosine of the value transferred in degrees. Parameter: double - x in the range from -PI/2 to +PI/2 Syntax: <function name="arcos" return="<double val>"> double </function> Example: <let name= "arccos_val" type="double"></let> <function name="arccos" return=" arccos_val"> 20.0 </function>

Function name	Description
ARCTAN	The function calculates the arctan of the value transferred in degrees. Parameter: double - arctan of y/x Syntax: <code><function name="arctan" return="<double val>"> double </function></code> Example: <code><let name= "arctan_val" type="double"></let> <function name="arctan" return="arctan_val"> 20.0 </function></code>
DLL load	The function loads an additional user DLL to the memory. Parameter: dll_name - DLL name class_name - name of the function class Syntax: <code><function name="dll.load"> dll_name, class_name </function></code> Example: <code><function name="dll.load"> _T"customer.dll", _T"customer" </function></code>
DLL function	The function calls a function from a user DLL. All parameters listed after the parameter ID are transferred to the function called. Parameter: class_name - name of the function class id - of the function parameter - maximum seven function parameters (string variables) Syntax: <code><function name="dll.function"> class_name, id, parameter1, parameter2</function></code> Example <code><function name="dll.function"> _T"customer", 290, _T"par1", _T"par2"</function></code>
File processing	

Function name	Description
Loading a file	The function loads the contents of the file specified to a string variable. Attribute: Return - name of the local variable Parameter: Programe - file name Syntax: <code><function name="doc.readfromfile" return="<string var>"> programe </function></code> Example: <code><let name = "my_var" type="string" ></let></code> <code><function name=" doc.readfromfile " return="my_var"> _T"\spf\test.mpf" </function></code>
Writing to a file	The function writes the contents of a string variable to the file specified. Parameter: programe - file name str1 - string Syntax: <code><function name="doc.writetofile" > programe, str1 </function></code> Example: <code><let name = "my_var" type="string" > file content </let></code> <code><function name="doc.writetofile">_T"\spf \test.mpf", my_var </function></code>
Deleting a file	The function removes the file specified from the directory. Parameter: programe - file name Syntax: <code><function name="doc.remove" > programe </function></code> Example: <code><function name="doc.remove">_T"\mpf\test.mpf" </function></code>

Function name	Description
Exist	If the file exists, the function returns the value 1. Parameter: programe - file name Syntax: <function name="doc.exist" return="<int_var>" > programe </function> Example: <let name ="exist">0</let> <function name="doc.exist" return="exist">_T"\mpf \test.mpf" </function>
NC program selection	The function selects the program specified for execution. The program must be stored in the NC file system. Parameter: programe - file name Syntax: <function name="ncfunc.select"> programe </ function> Example: <function name="ncfunc.select"> _T"\mpf\test.mpf" </function>

Function name	Description
Setting an individual bit	The function is used to manipulate individual bits of the specified variables. The bits can either be set or reset. Syntax: <code><function name="ncfunc.bitset" refvar="address" value="set/reset" > bit0, bit1, ... bit9 </function></code> Attributes: refvar - specifies the name of the variable, in which the bit combination should be written value – bit value, value range 0 and 1 Values: The bit numbers starting with zero should be transferred as function values. A maximum of 10 bits per call can be modified. Example: <code><function name="ncfunc.bitset" refvar="nck/Channel/Parameter/R[1]" value="1" > 0, 2, 3, 7 </function></code> <code><function name="ncfunc.bitset" refvar="nck/Channel/Parameter/R[1]" value="0" > 1, 4 </function></code>
Delete control	The function deletes the specified picture control. Syntax: <code><function name="control.delete"> control name </function></code> or <code><function name="ncfunc.bitset" refvar="\$MN_USER_DATA_HEX" value="1" > 0, 2, 3, 7 </function></code> Attribute: name – function name Value: control name – name of the control Example: <code><function name="<control.delete>"> _T"my_editfield" </function></code>

Function name	Description
Add Item	The function inserts a new element at the end of the list. Note: The function is only available for the control types "listbox" and "graphicbox". Syntax: <pre><function name="control.additem"> control name, item </function></pre> Attribute: name – function name Values: control name – control name item - expression to be inserted itemdata - integer value; defined by the user Example: <pre><let name ="itemdata">1</let> <op> item_string = _T"text1" </op> <function name="control.additem">_T"listbox1", item_string, itemdata </function></pre>

Function name	Description
Insert Item	The function inserts a new element at the specified position. Note: The function is only available for the control types "listbox" and "graphicbox". Syntax: <pre><function name="control.insertitem"> control name, index, item, itemdata </function></pre> Attribute: name – function name Values: control name – control name index – position starting with zero item - expression to be inserted itemdata - integer value; defined by the user Example: <pre><let name ="itemdata">1</let> <op> item_string = _T"text2" </op> <function name="control.insertitem">_T"listbox1", 1, item_string, itemdata </function></pre>
Delete item	The function deletes an element at the specified position. Note: The function is only available for the control types "listbox" and "graphicbox". Syntax: <pre><function name="control.deleteitem"> control name, index </function></pre> Attribute: name – function name Values: control name – control name index– index starting at 0 Example: <pre><function name="control.deleteitem">_T"listbox1", 1</function></pre>

Function name	Description
Load Item	The function inserts a list of expressions into the control. The function is only available for the control types "listbox" and "graphicbox". Syntax: <code><function name="control.loaditem"> control name, list </function></code> Attribute: name – function name Values: control name – control name list- string variable Structure of the list: The list contains a number of expressions, which must be separated from one another using a \n. Example: <pre> <let name="item_string" type="string"></let> <let name="plotlist" type="string"></let> <print name ="item_string" text="p; %f; %f; %f; %f \n">s_z, s_x</print> <op>plotlist = plotlist + item_string</op> <print name ="item_string" text="l; %f; %f; %f; %f\n">s_z, s_x, e_z, e_x </print> <op>plotlist = plotlist + item_string</op> <op> s_x = e_x </op> <op> s_z = e_z</op> <op> e_x = s_x + 10 </op> <op> e_z = s_z - 100 </op> <print name ="item_string" text="l; %f; %f; %f; %f\n">s_z, s_x, e_z, e_x </print> <op>plotlist = plotlist + item_string</op> <function name="control.loaditem">_T"gbox", plotlist</function> </pre>

Function name	Description
Empty	The function deletes the contents of the specified list box or graphic box controls. Syntax: <function name="control.empty"> control name, </function> Attribute: name – function name Values: control name – Control name Example: <function name=" control.empty">_T"listbox1"</function>
Get focus	The function supplies the name of the control, which has the input focus. Syntax: <function name="control.getfocus" return="focus_name" /> Attributes: name – function name return – a string variable should be specified, into which the control name is copied. Example: <let name>="focus_field" type="string"></let> <function name="control.getfocus" return="focus_field"/>

Function name	Description
Set focus	The function sets the input focus to the specified control. The Controlname should be transferred as text expression of the function. Syntax: <code><function name="control.setfocus"> control name </function></code> Attribute: name – function name Value: control name – name of the control Example: <code><function name="control.setfocus" ">_T"listbox1"</function></code>
Get cursor selection	For a list box, the function supplies the cursor index. The Controlname should be transferred as text expression of the function. Syntax: <code><function name="control.getcurssel" retvar="var"> control name </function></code> Example: <code><let name="index"></let> <function name="control.getcurssel" ">_T"listbox1"</function></code>
Set cursor selection	For a list box, the function sets the cursor to the appropriate line. The Controlname should be transferred as text expression of the function. Syntax: <code><function name="control.setcurssel" > control name, index</function></code> Example: <code><let name="index">2</let> <function name="control.setcurssel" ">_T"listbox1",index</function></code>

Function name	Description
Get Item	For a list box, the function copies the contents of the selected line to the specified variable. A string variable should be specified as reference variable. The Controlname should be transferred as text expression of the function. Syntax: <pre><function name="control.getitem" return="var"> control name, index </function></pre> Example: <pre><let name>="index">2</let> <let name>="item" type="string"></let></pre> <pre><function name="control.getitem" return="item" ">_T"listbox1",index</function></pre>
Get Item Data	For a list box, the function copies the user-specific allocated value of an element to the specified variable. For an edit control, the function copies the user-specific allocated value (item_data) to the specified variable. An integer variable should be specified as reference variable. The Controlname should be transferred as text expression of the function. Syntax: <pre><function name="control.getitemdata" return="var"> control name, index </function></pre> Example: <pre><let name>="index">2</let> <let name>="itemdata"></let></pre> <pre><function name="control.getitemdata" return="itemdata" ">_T"listbox1",index</function></pre>
Abs	This function returns the absolute value of the specified number. Syntax: <pre><function name="abs" return="var"> value </ function></pre>
SDEG	The function converts the specified value into degrees. Syntax: <pre><function name="sdeg" return="var"> value </ function></pre>
SRAD	The function converts the specified value into RADian. Syntax: <pre><function name="srad" return="var"> value </ function></pre>

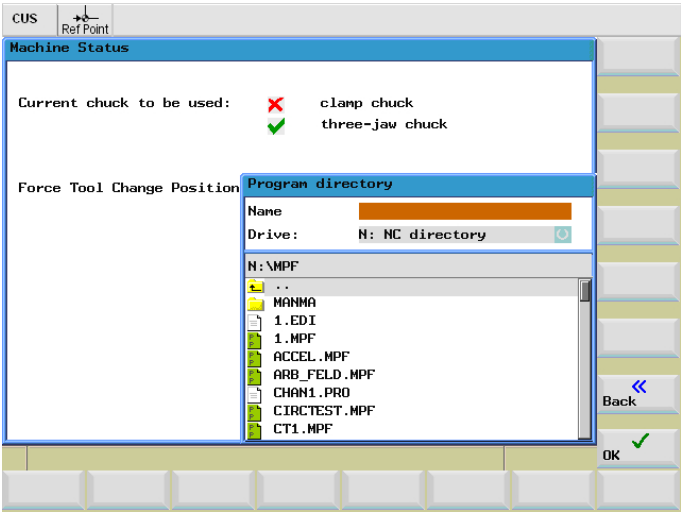
Function name	Description
SQRT	The function calculates the square root of the specified value. Syntax: <code><function name="sqrt" return="var"> value </function></code>
ROUND	The function rounds of the transferred number to the specified number of decimal places. If the number of decimal places is not specified, then the function rounds off the number, taking into account the first decimal place. Syntax: <code><function name="round" return="var"> value, nDecimalPlaces </function></code>
FLOOR	The function supplies the largest possible integer value, which is less than or equal to the transferred value. Syntax: <code><function name="floor" return="var"> value </function></code>
CEIL	The function supplies the smallest possible integer value, which is greater than or equal to the transferred value. Syntax: <code><function name="ceil" return="var"> value </function></code>
LOG	The function calculates the logarithm of the specified value. Syntax: <code><function name="log" return="var"> value </function></code>
LOG10	The function calculates the common (decadic) logarithm of the specified value. Syntax: <code><function name="log10" return="var"> value </function></code>
POW	The function calculates the value "a^b". Syntax: <code><function name="pow" return="var"> a, b </function></code>
MIN	The function compares the transferred value and returns the lower of the values. Syntax: <code><function name="min" return="var"> value1, value2 </function></code>

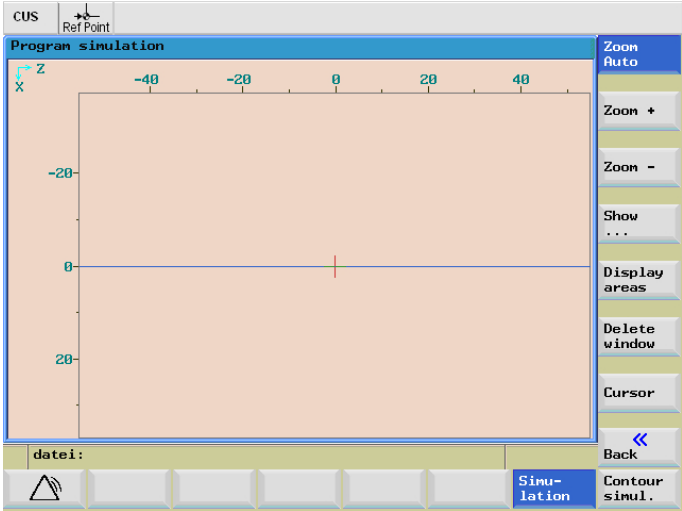
Function name	Description
MAX	The function compares the transferred value and returns the higher of the values. Syntax: <code><function name="max" return="var"> value1, value2 </function></code>
RANDOM	The function returns a pseudo random number. Syntax: <code><function name="random" return="var" </function></code>
MMC	Function User-defined dialog windows (dialog screens) from the part program can be displayed with the MMC command. The appearance of the dialog window is specified through purely textual configuration (XML file in the F:\appl directory (manufacturer drive)); here, the HMI software remains unchanged. Syntax MMC ("operating area, command, XML script file, menu name, reserved, reserved, display time or acknowledgment variable, reserved", "acknowledgment mode") Description • MMC Calling the dialog window interactively from the part program on the HMI. • EasyXml Identification for user dialogs that are to be started from a part program. • XML_ON or XML_OFF Command: Display selection or display deselection • XML file Name of the XML script file • Menu Name of the menu tag that manages the dialog to be displayed • reserved reserved for HMI Operate • reserved reserved for HMI Operate • Time Display time of the dialog for acknowledgments mode "N" • reserved reserved for HMI Operate • Acknowledgement mode "S" synchronous, acknowledgement via the "OK" softkey "N" asynchronous, dialog closes after the agreed time

Function name	Description
MMC continued	Example of synchronous call: NC instruction MMC("EASYXML oder CYCLES", XML_ON, mmc_cmd.xml, cmd1,,,,,"S") Datei: mmc_cmd.xml <pre> <menu name = "cmd1"> <open_form name = "cmd1_form" /> <softkey_ok> <close_form /> </softkey_ok> </menu> <form name = "cmd1_form" xpos ="12" ypos="100" width="500" height="240"> <init> </init> <paint> </paint> </form> </pre> Example of asynchronous call (acknowledgment not expected): NC instruction MMC("EASYXML oder CYCLES", PICTURE_ON, mmc_cmd.xml, cmd1,,,10,,,"N") Datei: mmc_cmd.xml <menu name = "cmd1"> <pre> <open_form name = "cmd1_form" /> <softkey_ok> <close_form /> </softkey_ok> </menu> <form name = "cmd1_form" xpos ="12" ypos="100" width="500" height="240"> <init> </init> <paint> </paint> </form> </pre>

Function name	Description
String find	The function searches the transferred string for the first match with the substring. If the substring is found, the function provides the index to the first character (starting with zero) or, failing this, -1. Parameter: string - string variable findstring - string to be found start index – start index (optional) Syntax: <pre><function name="string.find" return="<int val>"> str1, find string </function></pre> Example: <pre><let name="index">0</let> <let name="str1" type="string">A brown bear hunts a brown dog. </let> <function name="string.find" return="index"> str1, _T"brown" </function></pre> Result: Index = 2 or <pre><function name="string.find" return="index"> str1, _T"brown", 1 </function></pre>

Function name	Description
String reverse find	The function searches the transferred string for the last match with the substring. If the substring is found, the function provides the index to the first character (starting with zero) or, failing this, -1. Parameter: string - string variable find string - string to be found start index – start index (optional) Syntax: <code><function name="string.reversefind" return="<intval>"> str1, find string </function></code> Example: <code><let name="index">0</let> <let name="str1" type="string">A brown bear hunts a brown dog. </let> <function name="string.reversefind" return="index"> str1, _T"brown" </function></code> Result: Index = 21 or <code><function name="string.reversefind" return="index"> str1, _T"brown" 10 </function></code> Result: Index = 2
Dialogs	

Function name	Description
Program selection dialog	The function opens a dialog for selecting a part program. The selected program name is copied to the reference variable specified. The dialog can only be called from within a softkey tag.  Syntax: <pre><function name= "doc.fileselect" return="string var" /></pre> Example: <pre><let name = "program_content" type="string" > </let> <let name = "curr_program" type="string" > </let> <function name= "doc.fileselect" return="curr_program" /> <function name="doc.readfromfile " return=" program_content" > curr_program </function></pre>

Function name	Description
Program simulation dialog	The function opens program simulation in the form of a pop-up dialog. If a program name is specified, then the program is automatically selected. Otherwise, the simulation represents the traversing motion of the currently selected program. The dialog can only be called from within a softkey tag. The "Back" softkey function returns the user to the previous dialog.  Syntax: <pre><function name= "ncfunc.progsimulation"> prog name </function></pre> Example: <pre><let name = "curr_program" type="string" > </let> <function name= "doc.fileselect" return="curr_program" /> <function name= "ncfunc.progsimulation"> curr_program </function></pre>

Generating commissioning dialogs

2.1 Overview of functions

Purpose

The "Startup Wizard" allows additional devices to be simply commissioned, activated, deactivated and tested. The available equipment and device states are displayed in a list by the control system. The system can manage a maximum of 64 devices.

Softkeys are used to activate or deactivate a device.

The "Startup Wizard" function is available in the <SYSTEM> operating area.

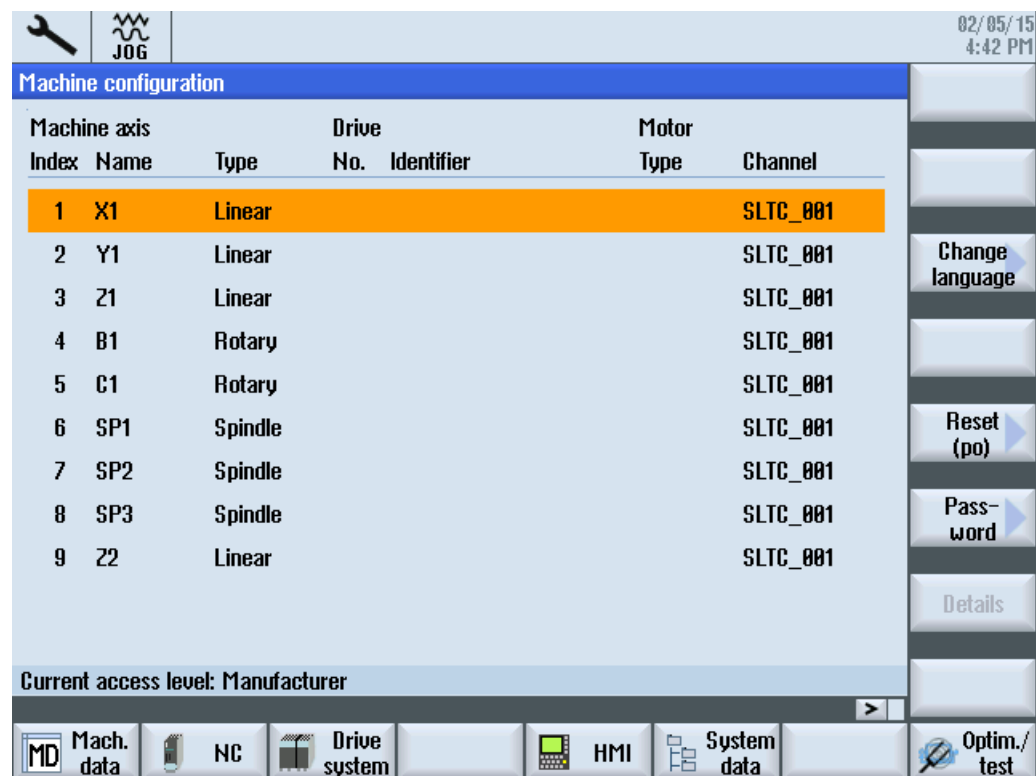


Figure 2-1 The SYSTEM main screen with active "Startup Wizard" softkey

Configuration

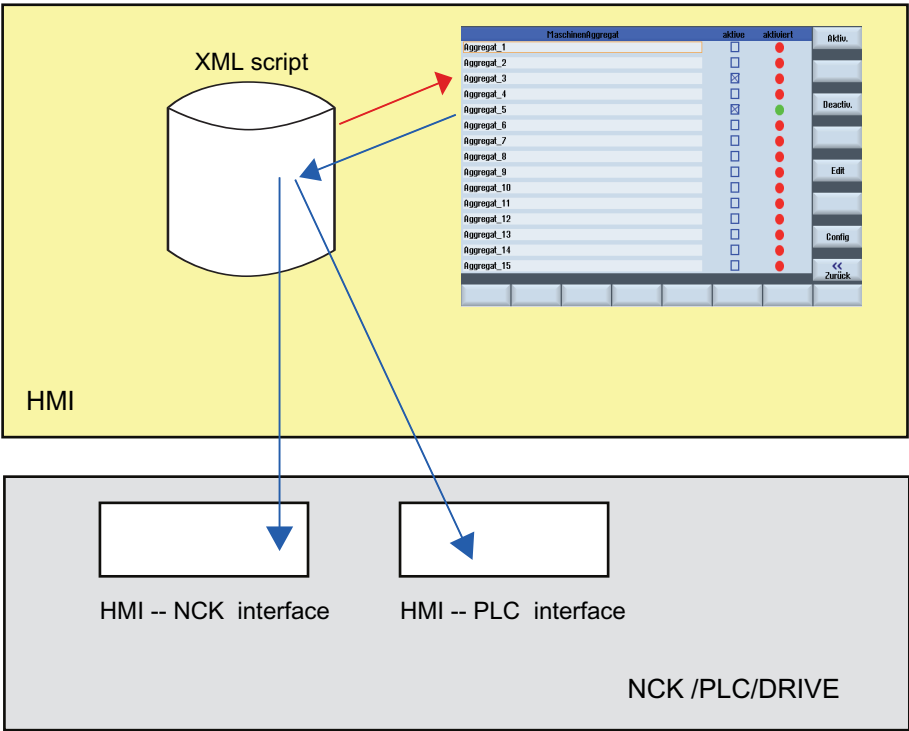


Figure 2-2 Mode of operation of the "Startup Wizard"

To use the "Startup Wizard", the following functions should be configured by the machine manufacturer:

- **PLC ↔ HMI interface**
The optional devices are managed via the interface between the user interface and the PLC.
- **Script processing**
The machine manufacturer saves the sequences to be executed for commissioning, activating, deactivating or testing a device, in a statement script.
- **Parameter dialog (optional)**
The parameter dialog shows device information that is saved in the script file.

Storage of the files

The "Startup Wizard" files are stored on the system CompactFlash card in the "dvm" (machine builder) directory.

File	Name	Target directory
Text file	oem_aggregate_xxx.ts	card\oem\sinumerik\hmi\lmg
Script file	agm.xml	card\oem\sinumerik\hmi\dvm

File	Name	Target directory
Archive file	Any	card\oem\sinumerik\hmi\dvm \archives
PLC user program	Any	PLC

2.2 Configuration in the PLC user program

Loading configurations

The configurations created are transferred to the manufacturer directory of the control, with the script and text file. Additionally, the corresponding PLC user program should be loaded.

Programming the equipment

Communication between the operator component and the PLC takes place in the PLC user program via data block VB99050000, in which 128 words are reserved for the device management.

PLC words are assigned beginning with Device 1:

Data block	Device designation
DB9905.DBB0	Device 1
DB9905.DBB4	Device 2
...	...
DB9905.DBB192	Device 49
DB9905.DBB196	Device 50

Four bytes with the following meanings are used for each device:

Byte	Bit	Description
0	0	== 1 Device has been started up (HMI acknowledgment)
	1	== 1 Device is to be activated (HMI request)
	2	== 1 Device is to be deactivated (HMI request)
	3-7	Reserved
1	0-7	Reserved
2	0	== 1 Device is active (PLC acknowledgment)
	1	== 1 Device has an error
	2-7	Reserved
3	0-7	Unique identifier for the device

Storage of the files

The Easy Extend files are stored on the system CompactFlash card in the "oem" (MANUFACTURER) and "oem_i" (INDIVIDUAL) directory.

File	Name	Target directory
Text file	oem_aggregate_xxx.ts	/oem/sinumerik/hmi/lng/ /oem_i/sinumerik/hmi/lng/
Script file	agm.xml	/oem/sinumerik/hmi/dvm /oem_i/sinumerik/hmi/dvm
Archive file	Any	/oem/sinumerik/hmi/dvm/archives /oem_i/sinumerik/hmi/dvm/archives
PLC user program	Any	PLC

General sequence

The machine manufacturer must execute the following steps to make the required data available:

1. Creating a PLC user program which activates the device during activation on the PLC.
2. Commissioning of the "standard machine" followed by backup of the data in a series startup archive.
3. Installation of the devices, commissioning, followed by read-out of the data as a differential series startup archive.

Note

Changing the machine configuration

Should there be any need to edit the drive machine data, this should be adapted in the control first. This procedure should be repeated for all devices and constellations.

Adding axes

If the machine is extended with machine axes, it is important to install the drive objects (DO) in a fixed sequence because the series startup archive contains the constellation of the machine manufacturer's reference machine and cannot be applied if the sequence is changed.

It is recommended that the following settings be selected for the "control components":

- NC data
- PLC data
- Drive data
 - ACX format (binary)

2.3 Display on the user interface

Dialogs on the user interface

The following dialogs are available for "Startup Wizard":

- The control offers a **configurable dialog**, in which the available devices are shown.
- If first commissioning has not taken place yet, the control opens the **commissioning dialog**.

If a commissioning procedure (XML instruction: "START_UP") has been programmed, and the device has still not been commissioned, then the control starts the commissioning procedure.

This involves a complete data backup before the series startup archives saved in the script file are read in.

In the event of an error, the commissioning engineer can decide whether to roll back the commissioning procedure or to rectify possible errors in machine configurations manually.

- Commissioning can be aborted early with the "Cancel" function. The control then copies the previously saved commissioning files back.

If the machine has to be switched off after successful completion of the commissioning, the XML statement "POWER_OFF" can be used to program that a corresponding message is output on the control.

2.4 Creating language-dependent texts

Structure of text file

The XML files with the language-dependent texts must be created in UTF8 format:

Example oem_aggregate_eng.ts

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE TS>
<TS>
  <context>
    <name>EASY_EXTEND</name>
    <message>
      <source>DEVICE_ONE</source>
      <translation>Device one</translation>
    </message>
    <message>
      <source>DEVICE_TWO</source>
      <translation>Device two</translation>
    </message>
  </context>
</TS>
```

Example oem_aggregate_deu.ts

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE TS>
<TS>
  <context>
    <name>EASY_EXTEND</name>
    <message>
      <source>DEVICE_ONE</source>
      <translation>Device one</translation>
    </message>
    <message>
      <source>DEVICE_TWO</source>
      <translation>Device two</translation>
    </message>
  </context>
</TS>
```

2.5 User example for a power unit

Activating the drive object

The drive object to be activated has already been commissioned and deactivated again by the machine manufacturer, to market the axis (axes) as an option.

To activate the axis carry out the following steps:

- Activate the drive object via p0105.
- 2. Enable the axis in the channel machine data.
- Back up the drive machine data via p0971.
- Wait until the data has been written.
- Restart the NCK and the drives.

Programming:

```

<DEVICE>
  <list_id>1</list_id>
  <name> "Activate the drive" </name>

  <SET_ACTIVE>
    <data name = "drive/dc/p105[D05]">1</data>
    <data name = "$MC_AXCONF_MACHAX_USED[4]">5</data>
    <data name = "drive/dc/p971[D05]">1</data>
    <while>
      <condition> "drive/dc/p971[D05]" !=0 </condition>
    </while>
    <control_reset resetnc ="true" resetdrive = "true"/>
  </SET_ACTIVE>

  <SET_INACTIVE>
    <data name = "drive/dc/p105[D05]">0</data>
    <data name = "$MC_AXCONF_MACHAX_USED[4]">0</data>
    <data name = "drive/dc/p971[D05]">1</data>
    <while>
      <condition> "drive/dc/p971[D05]" !=0 </condition>
    </while>
    <control_reset resetnc ="true" resetdrive = "true"/>
  </SET_INACTIVE>
</DEVICE>

```


Activating the PLC-controlled device

The device is addressed via output byte 10 and signals data set ready to the PLC via input byte 9.

The output byte is set to the specified coding for activation. The WHILE loop then waits for the data set ready of the device.

Programming:

```
<SET_ACTIVE>
  <DATA name = "plc/qb10"> 8 </DATA>
  <while>
    <condition> "plc/ib9" !=1 </condition>
  </while>
</SET_ACTIVE>
```

2.6 Script language

Note

All of the script elements described in the Generating user dialogs (Page 5) function form the basis for the "Startup Wizard" function. Additional script elements are defined to manage additional devices.

Program parts of the script

The script is divided into the following areas:

- "Startup Wizard" - frame
- Frame to define the actions that can be executed for a device
- Identifier for the device
- Identifier for commissioning the device
- Identifier for activating the device
- Identifier for deactivating the device
- Identifier for testing the device
- Identifier for the parameter dialog

The individual tags are described in the following chapters.

Description

Identifier <tag>	Meaning
AGM	Identifier for the "Startup Wizard"
DEVICE	Identifier for the description of the device. Attributes: option_bit The device is assigned a fixed bit number for the option management.
NAME	The identifier specifies the name of the device to be displayed in the dialog. If a text reference is used, the dialog displays the text which is saved for the identifier.
START_UP	The identifier contains a description of the sequences required for commissioning the device.
SET_ACTIVE	The identifier contains a description of the sequences required to activate the device. Attributes: timeout The attribute permits a timeout to be specified in seconds. The system interrupts processing if the script has still not been completed after this time.
SET_INACTIVE	The identifier contains a description of the sequences required to shut down the device. Attributes: timeout The attribute permits a timeout to be specified in seconds. The system interrupts processing if the script has still not been completed after this time.
TEST	The identifier contains the statements for testing the operating capability of a device. Attributes: timeout The attribute permits a timeout to be specified in seconds. The system interrupts processing if the script has still not been completed after this time.
UID	Unique numerical identifier to identify the device in the PLC ↔ HMI interface.
VERSION	Identifier for a version

Negative acknowledgment of the function execution

With the automatically provided variable "\$actionresult", the system can inform the XML parser of a negative execution result. If the value is set to zero, the parser aborts the function processing.

Example

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE AGM>
<AGM>                                Identifier for the "Startup Wizard"
<DEVICE>
  <NAME> Device 1 </NAME>             Identifier for the device
  <START_UP>                          Identifier for commissioning the device
  ...
</START_UP>
  <SET_ACTIVE>                        Identifier for activating the device
  ...
</SET_ACTIVE>
  <SET_INACTIVE>                      Identifier for deactivating the device
  ...
</SET_INACTIVE>
  <TEST>                              Identifier for testing the device
  ...
</TEST>
</DEVICE>
...
</AGM>
```

2.6.1 CONTROL_RESET

Description

This identifier allows one or more control components to be restarted. Execution of the script is only continued when the control has resumed cyclic operation.

Programming

Identifier:	CONTROL_RESET	
Syntax:	<CONTROL_RESET resetnc="TRUE" />	
Attributes:	resetnc="true"	The NC component is restarted.
	resetdrive="true"	The drive components are restarted.

2.6.2 FILE

Description

The identifier enables standard archives to be read in or created.

- Reading in an archive:
The file name of the archive must be specified for reading in an archive.
- Creating an archive:
If the attribute create="true" is specified, the function creates a standard archive (*.arc) under the specified name and stores the file in the .../dvm/archives directory.

Programming

Identifier:	FILE	
Syntax:	<file name ="<archive name>" /> <file name ="<archive name>" create="true" class="<data classes>" group="<area>" />	
Attributes:	name	Identifier for the file name

create	A commissioning archive is created under the specified name in the .../dvm/archives/ directory.
group	Specifies the data groups that are to be contained in the archive. If several data groups are to be saved, the groups should be separated by a blank. The following data groups can be contained in the archive: • NC • PLC • HMI • DRIVES

Example

```
<!-- Create data class archive -->
<file name="user.arc" create="true"
group="nc plc hmi" />

<!--Read archive into the control-->
<file name="user.arc" />
```

2.6.3 OPTION_MD

Description

The identifier allows option machine data to be redefined. As delivered, the system uses MD14510 \$MN_USER_DATA_INT[0] to \$MN_USER_DATA_INT[3].

If the PLC user program manages the options, the appropriate data words must be provided in a data block or GUD.

The data is structured in bits. Starting with bit 0, there is a fixed assignment of the bits to the listed devices, i.e. bit 0 is assigned to device 1, bit 1 to device 2, etc. If more than 16 devices are managed, the address identifiers of the device groups 1-3 are assigned via the area index.

Note

Converting the value range

The value range of MD14510 \$MN_USER_DATA_INT[i] is from -32768 to +32767. To activate the devices bit-by-bit via the machine data dialog, the bit combination must be converted to decimal representation.

Programming

Identifier:	OPTION_MD	
Syntax:	Area 0: <option_md name = "Address identifier of the data" /> OR: <option_md name = "Address identifier of the data" index= "0"/> Area 1 to 3: <option_md name = "Address identifier of the data" index= "Area index"/>	
Attributes:	name	Identifier for the address, e.g. \$MN_USER_DATA_INT[0]
	index	Identifier for the area index:
		0 (default setting): Device 1 to 16
		1: Device 17 to 32 2: Device 33 to 48 3: Device 49 to 64

2.6.4 PLC_INTERFACE

Description

This identifier permits the PLC ↔ HMI interface to be redefined. The system expects 128 addressable words.

Default: DB9905 (VB99050000)

Programming

Identifier:	PLC_INTERFACE	
Syntax:	<plc_interface name = "Address identifier of the data" />	
Attributes:	name	Identifier for the address, e.g. "plc/mb170"

Example: plc/mb170

2.6.5 POWER_OFF

Description

Identifier for a message prompting the operator to switch the machine off. The message text is permanently saved in the system.

Programming

Identifier: **POWER_OFF**
Syntax: `<power_off />`
Attributes: --

2.6.6 WAITING

Description

After a reset of the NC or the drive, there is a wait for the restart of the respective component.

Programming

Identifier: **WAITING**
Syntax: `<WAITING WAITINGFORNC ="TRUE" />`
Attributes: `waitingfornc="true"` There is a wait for the restart of the NC.
`waitingfordrive="true"` There is a wait for the restart of the drive.

2.6.7 XML identifiers for the dialog

Dialog for the parameterization

A dialog can be configured for each device so that additional parameters can be set or output during runtime. This is displayed by pressing the "Additional parameters" softkey.

All of the script elements described in Chapter Generating user dialogs (Page 5) can be used to generate the dialog.

Example

```

<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE AGM>
<AGM>
<DEVICE>
  <NAME> Device 1 </NAME>
  <START_UP>
  ...
</START_UP>
  <SET_ACTIVE>
  ...
</SET_ACTIVE>
  ...
  <FORM>                                Identifier for a user dialog
    <INIT>
      <CONTROL name = "edit1" .../>      Identifier for an input field
    </INIT>
    <PAINT>                              Identifier for text or image display
      <TEXT>hello world !</TEXT>
    </PAINT>
  </FORM>
</DEVICE>
  ...
</AGM>

```

2.6.8 SOFTKEY_OK, SOFTKEY_CANCEL**Description**

The identifier SOFTKEY_OK overwrites the standard behavior when closing a dialog by means of the "OK" softkey. The identifier SOFTKEY_CANCEL overwrites the standard behavior when closing a dialog by means of the "CANCEL" softkey.

The following functions can be performed within this identifier:

- Data manipulation
- Conditional processing
- Loop processing

Programming

Identifier:	SOFTKEY_OK
Syntax:	<SOFTKEY_OK> ... </SOFTKEY_OK>
Identifier:	SOFTKEY_CANCEL
Syntax:	<SOFTKEY_CANCEL> ... </SOFTKEY_CANCEL>

Index

C

Configuration file, 7

G

Generating commissioning dialogs, 111

M

Menu tree, 7

S

Start softkey, 7

Startup Wizard, 111

X

XML

- Operators, 41

- Syntax, 40

XML function No.

- Control form color, 83

XML functions

- Abs, 102

- AddItem, 97

- ARCOS, 92

- ARCSIN, 92

- ARCTAN, 93

- CEIL, 103

- Control local time, 83

- Copy value and read, 78

- Copying and writing a value, 78

- Cosine, 91

- Delete control, 96

- Delete number of characters of a string, 89

- DeleteItem, 98

- Deleting a file, 94

- display resolution, 81

- Empty, 100

- Execute user function, 93

- Exist, 95

- Find substring, 89

- Find substring (reverse), 90

- FLOOR, 103

- Get cursor selection, 101

- getfocus, 100

- GetItem, 102

- GetItemData, 102

- InsertItem, 98

- Load user DLL, 93

- Loading a file, 94

- LoadItem, 99

- LOG, 103

- LOG10, 103

- MAX, 104

- MIN, 103

- MMC, 104

- NC program selection, 95

- Ncfunc bico to int, 81

- Ncfunc int to bico, 81

- Ncfunc is bico str valid, 82

- Ncfunc password, 82

- PI service, 79, 80

- POW, 103

- Program selection dialog, 108

- Program simulation dialog, 109

- RANDOM, 104

- ROUND, 103

- SDEG, 102

- Set cursor selection, 101

- setfocus, 101

- Setting an individual bit, 96

- Sine, 91

- SQRT, 103

- SRAD, 102

- String comparison, 84, 85

- String find, 106

- String reverse find, 107

- string.icmp, 85

- string.insert, 88

- string.left, 85

- string.length, 87

- string.middle, 86

- string.remove, 88

- string.replace, 87

- string.right, 86

- Tangent, 92

- Writing to a file, 94

XML identifier

- AGM, 120

- BOX, 63

- BREAK, 16

- CAPTION, 49

CLOSE, 49
CLOSE_FORM, 49
CONTROL, 50
CONTROL_RESET, 16, 122
CREATE_CYCLE, 70
CREATE_CYCLE_EVENT, 17
DATA, 18
DATA_ACCESS, 54
DATA_LIST, 18
DEVICE, 120
DO_WHILE, 38
DYNAMIC_INCLUDE, 19
EDIT_CHANGED, 55
ELSE, 19
FILE, 122
FOCUS_IN, 48
FOR, 36
FORM, 19, 43
FUNCTION, 64
FUNCTION_BODY, 65
HELP_CONTEXT, 54
HMI_RESET, 19
IF, 20
IMG, 62
INCLUDE, 20
INDEX_CHANGED, 55
INIT, 45
KEY_EVENT, 46
LET, 21
MENU, 55
MESSAGE, 47
MSG, 24
MSGBOX, 25
NAME, 120
NAVIGATION, 56
NC_INSTRUCTION, 69
OP, 26
OPEN_FORM, 56
OPTION_MD, 124
PAINT, 48
PASSWORD, 31
PLC_INTERFACE, 124
POWER_OFF, 31, 125
PRINT, 32
PROGRESS_BAR, 33
PROPERTY, 57
REQUEST, 67
SEND_MESSAGE, 34, 48
SET_ACTIVE, 120
SET_INACTIVE, 120
SHOW_CONTROL, 35
SLEEP, 34
SOFTKEY, 58
SOFTKEY_ACCEPT, 61
SOFTKEY_BACK, 61
SOFTKEY_CANCEL, 60, 127
SOFTKEY_OK, 60, 127
START_UP, 120
STOP, 34
SWITCH, 35
TEST, 120
TEXT, 61
THEN, 35
TIMER, 48
TYPE_CAST, 35
UID, 120
UPDATE_CONTROLS, 67
VERSION, 120
WAITING, 36, 125
WHILE, 37
XML_PARSER, 27

SIEMENS

SINUMERIK

**SINUMERIK 840D sI/828D
SINUMERIK Integrate for
Engineering Access MyMachine /
OPC UA
Commissioning Manual**

Introduction

1

Setting up OPC UA
components

2

Testing the connection

3

OPC UA data access

4

Valid for:

CNC Software

Version 4.7

06/2014

6FC5397-1DP40-5BA1

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

 DANGER
indicates that death or severe personal injury will result if proper precautions are not taken.

 WARNING
indicates that death or severe personal injury may result if proper precautions are not taken.

 CAUTION
indicates that minor personal injury can result if proper precautions are not taken.

NOTICE
indicates that property damage can result if proper precautions are not taken.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

 WARNING
Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Table of contents

1	Introduction.....	5
2	Setting up OPC UA components.....	7
2.1	Requirement.....	7
2.2	Setup.....	8
3	Testing the connection.....	13
4	OPC UA data access.....	19
4.1	Functional scope.....	20
4.1.1	Browsing.....	20
4.1.2	Variable paths.....	22
4.1.3	Supported services.....	26
4.1.4	Special features with regard to the OPC UA specification.....	27
	Index.....	29

Introduction

Overview

OPC Unified Architecture (OPC UA) is a standard communication protocol for the industrial environment.



Software option

You require the following software option in order to use this function: SINUMERIK Integrate for Engineering "Access MyMachine / OPC UA".

Important features include:

- Independent of any particular platform
- Security
- Service call timeouts
- Heartbeat
- Buffering

Today, two protocols are available for transport via the network:

1. Binary protocol
URL: `opc.tcp://ServerTCP port 4840`
Optimized for
 - Low overhead
 - Low usage of resources
 - Interoperability
2. Web service (SOAP)
URL: `http://Server`
Ports 80 (http) and 443 (https)
Optimized for
 - Tool support (used from "Java" or under ".NET")
 - Firewall compatibility

Information is provided in what is known as the "address space". This is comprised of "nodes" and "references". Detailed information on this is provided on the OPC Foundation (<https://opcfoundation.org/>) page.

The binary protocol is supported in the SINUMERIK environment. Currently, OPC UA Data Access is offered as service.

Setting up OPC UA components

2.1 Requirement

The OPC UA components can be installed on PCU, NCU and PPU target systems.

The following steps are necessary to do this:

1. Extending the system configuration
2. Creating the OPC UA configuration file
3. Adapting the OPC UA configuration file
4. Setting the license
5. Enabling the communication port
6. Checking the HMI time
7. Performing a restart

2.2 Setup

Extending the system configuration

In order that the OPC UA components start when SINUMERIK Operate runs up, the system configuration must be appropriately extended.

Copy the template file

".../siemens/sinumerik/hmi/template/cfg/MiniWeb_<Target system>_systemconfiguration.ini" to
".../oem/sinumerik/hmi/cfg/systemconfiguration.ini".

Note

- The file name changes to "systemconfiguration.ini". The prefix "MiniWeb_<Target system>_" must be removed after copying.
 - If the ".../oem/sinumerik/hmi/cfg/systemconfiguration.ini" file already exists, then you only have to copy the content of the template file.
 - The template file is a link. Make sure that you copy the file and not only the link.
-

Creating the OPC UA configuration file

Copy the template file

".../siemens/sinumerik/hmi/template/cfg/OPC_UAApplication.xml" to
".../oem/sinumerik/hmi/miniweb/WebCfg/OPC_UAApplication.xml".

Note

The template file is a link. Make sure that you copy the file and not only the link.

Adapting the OPC UA configuration file

Open the copied file

".../oem/sinumerik/hmi/miniweb/WebCfg/OPC_UAApplication.xml" with a text editor.

Replace all the "localhost" entries in the file with the IPv4 address of the target system.

- For the NCU and PPU: -X130
- For the PCU 50: Local Area Connection 2

Example

```
<?xml version="1.0" standalone="yes"?>
<OPCUAAPPLICATION>
  <CERTDATA                                KeyLength="1024"
                                           Hash="SHA1"
                                           txtC="DE"
                                           txtP="Bavaria"
```

```

txtL="Erlangen"
txtO="Siemens AG"
txtOU="Industry"
txtCN=""
txtE="opcua@siemens.com"
txtCA="selfsigned" />
<KEYS
  PublicKey="MiniWeb OPCUA_certificate.crt"
  PrivateKey="keys/MiniWeb OPCUA_key.crt"
  CAKey="" />
<SESSION
  MaxSessionCount="100" />
<BROWSE
  MaxNodesPerBrowse="50" />
<SUBSCRIPTION
  MinSamplingRate="100"
  MaxKeepAlive="20"
  MinPublishRate="1000" />
<BUILDINFO
  ProductName="Sinumerik OPC UA"
  ProductUri="http://automation.siemens.com"
  ManufacturerName="Siemens AG I DT MC"
  SoftwareVersion="4.4.0"
  BuildNumber="2" />
<APPLICATIONDESCRIPTION
  ApplicationUri="urn:192.168.10.222:miniweb"
  ApplicationNameLocale="en_us"
  ApplicationNameText="Sinumerik OPC UA OEM"
  DNSNAME="192.168.10.222" />
<ENDPOINTDESCRIPTION
  URL="opc.tcp://192.168.10.222:4840" />
<NODEMANAGEMENT
  TargetProviderName="NodeManagementProvider" />
</OPCUAAPPLICATION>

```

Setting the license

To access data with OPC UA clients, set the "Access MyMachine / OPC UA" license via the "Startup > Licenses" operating area.

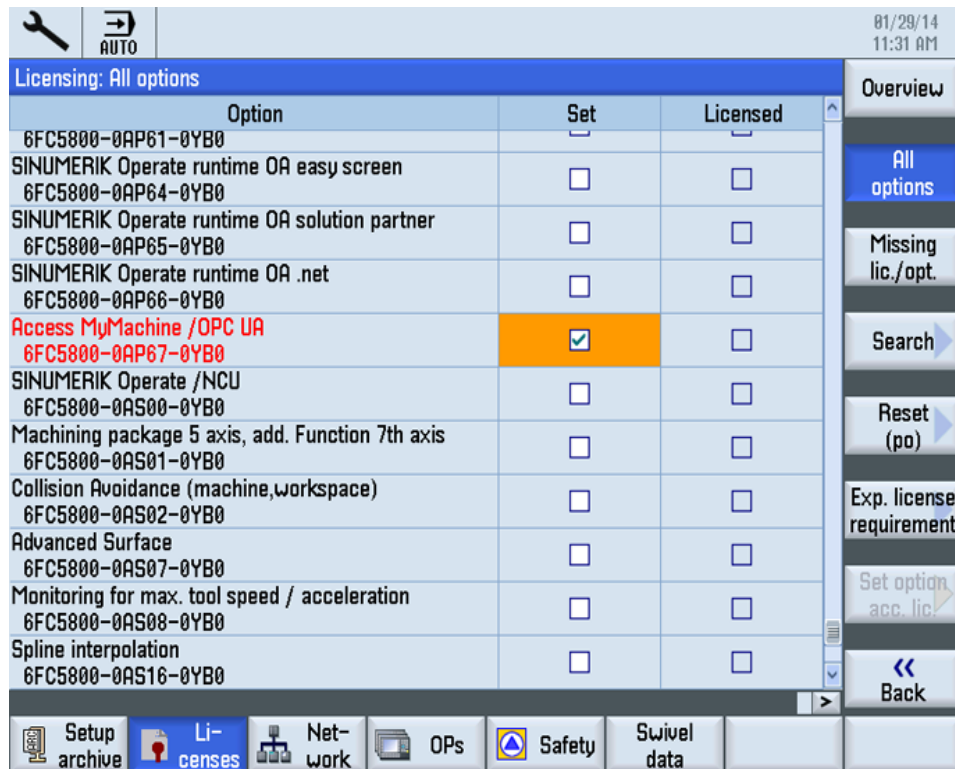


Figure 2-1 Setting the license

Enabling the communication port

Note

Only perform this step on the NCU and PPU target systems. Go directly to the next step on the PCU 50.

Enable the OPC UA communication port (TCP port 4840) via the "Startup > Network > Company network" operating area.

Press the "Change" button.

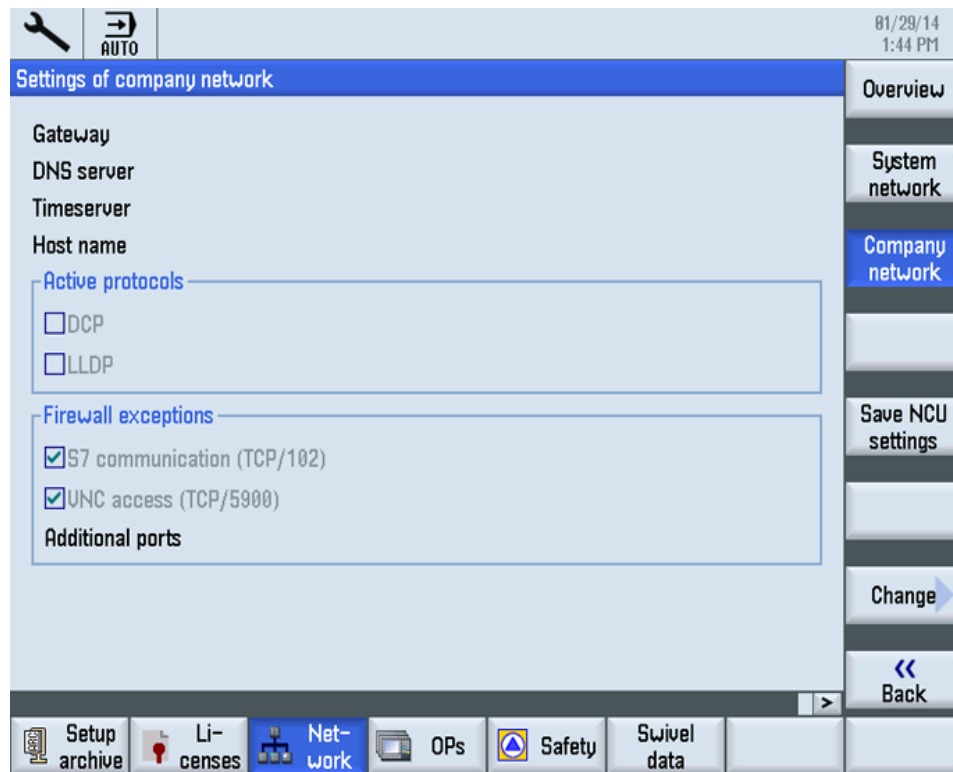


Figure 2-2 Enabling the communication port

Add the TCP port 4840 and confirm the changes with "OK".

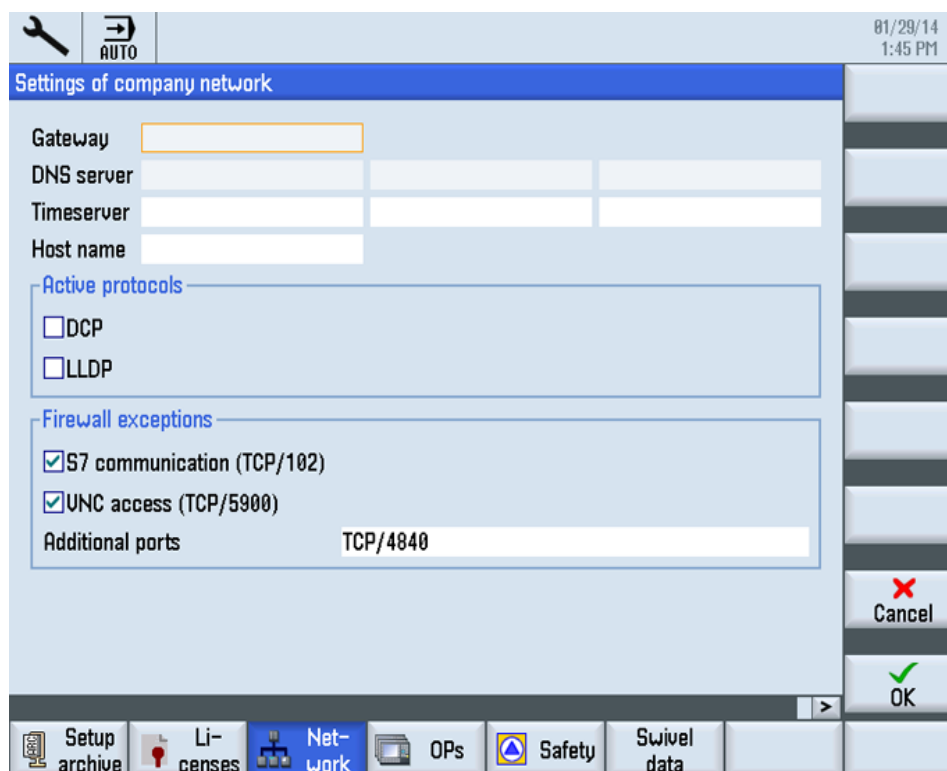


Figure 2-3 Adding the TCP port

Checking the HMI time

You must correctly set the HMI time so that the secure OPC UA communication functions.

Note

The certificate required for the secure OPC UA communication (../siemens/sinumerik/hmi/miniweb/System/SSL/MiniWeb OPCUA_certificate.crt) is generated automatically at the first run-up. The start of validity is set to the current time. The validity period is 30 years.

If the time is subsequently changed so that it lies outside the validity period, the secure OPC UA communication does not function.

Performing a restart

Perform a restart of the SINUMERIK Operate and a reset (po) to activate the changes and the license.

Testing the connection

Requirement

You can use the "Sample client" of the OPC Foundation (<http://www.opcf.org/>) at "Downloads / Sample Code and Apps / OPC UA 1.02 Sample Applications" to test the connection.

Note

There are two ways to establish the connection:

- Connection without security
- Connection with the security policy "Basic128Rsa15" and the security mode "SignAndEncrypt".

Procedure

1. Start the OPC UA "Sample client".

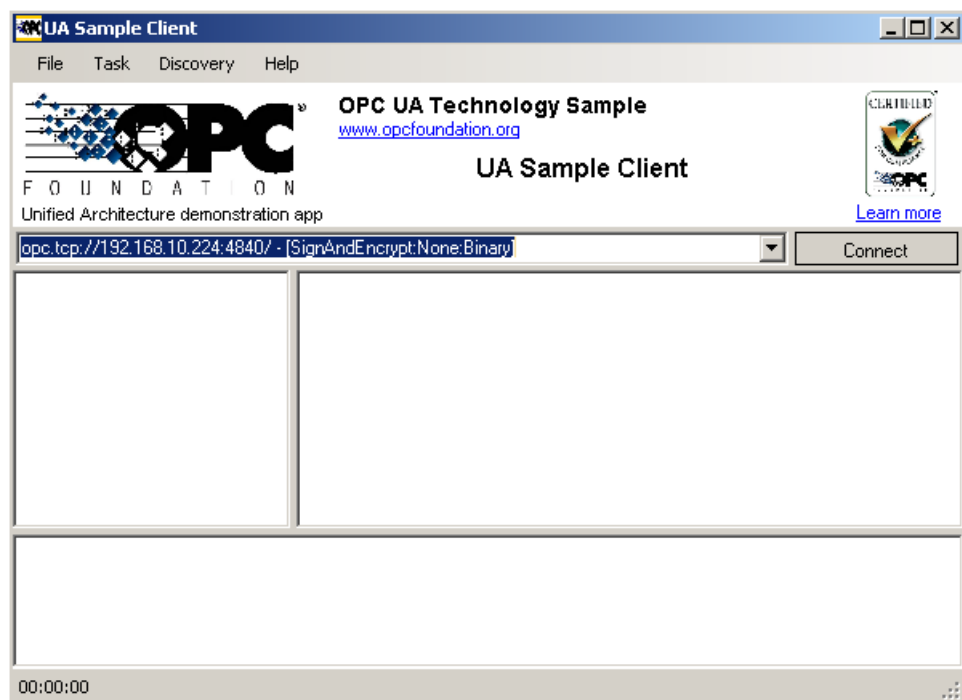


Figure 3-1 Sample Client main window

2. Select the "New" entry from the drop-down list.
The "Discover Servers" window opens.

- Now enter the IPv4 address of the target system and click the "Discover" button.

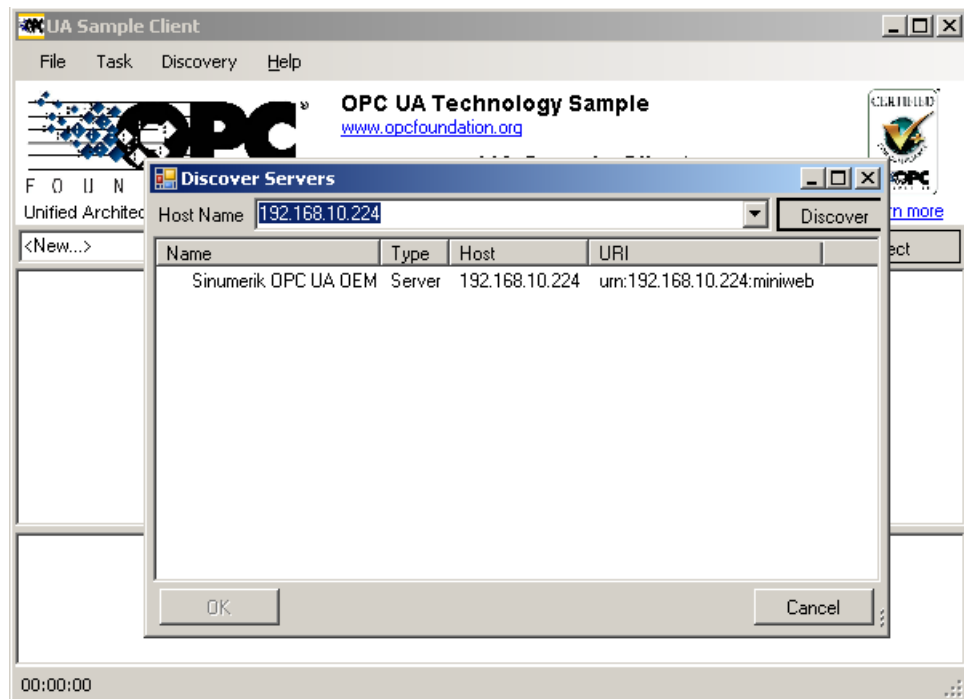


Figure 3-2 Discover servers

- The SINUMERIK OPC UA server appears in the list. Select the server and confirm with "OK".
- Return to the main window and click the "Connect" button.

6. To establish a simple connection without security, make the following settings and confirm the settings with "OK".

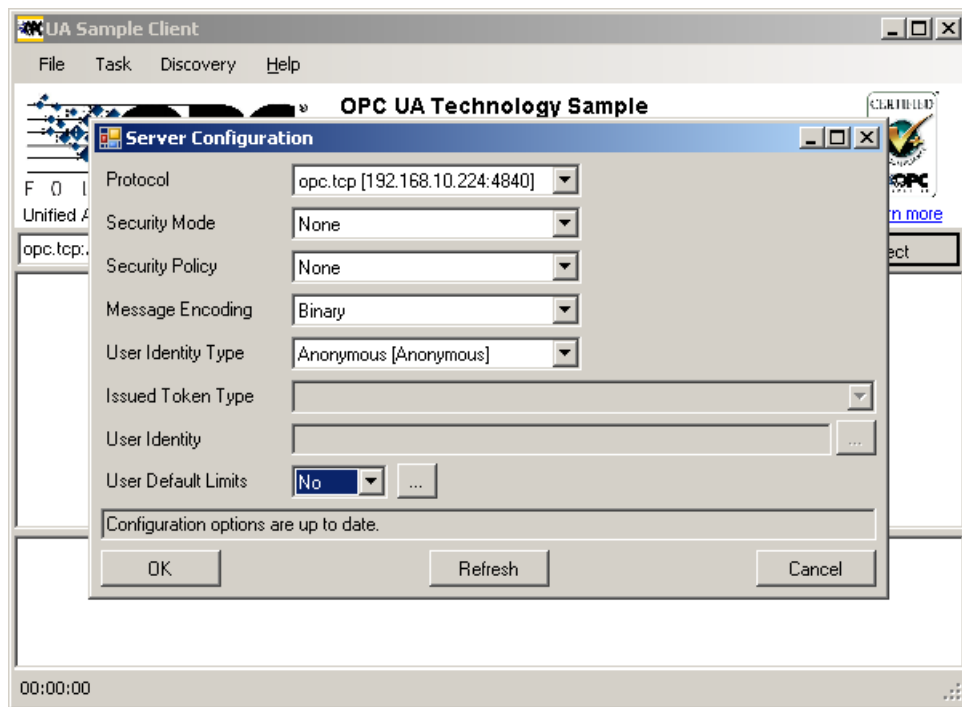


Figure 3-3 Server configuration

7. Confirm the following "Open Session" dialog with "OK".

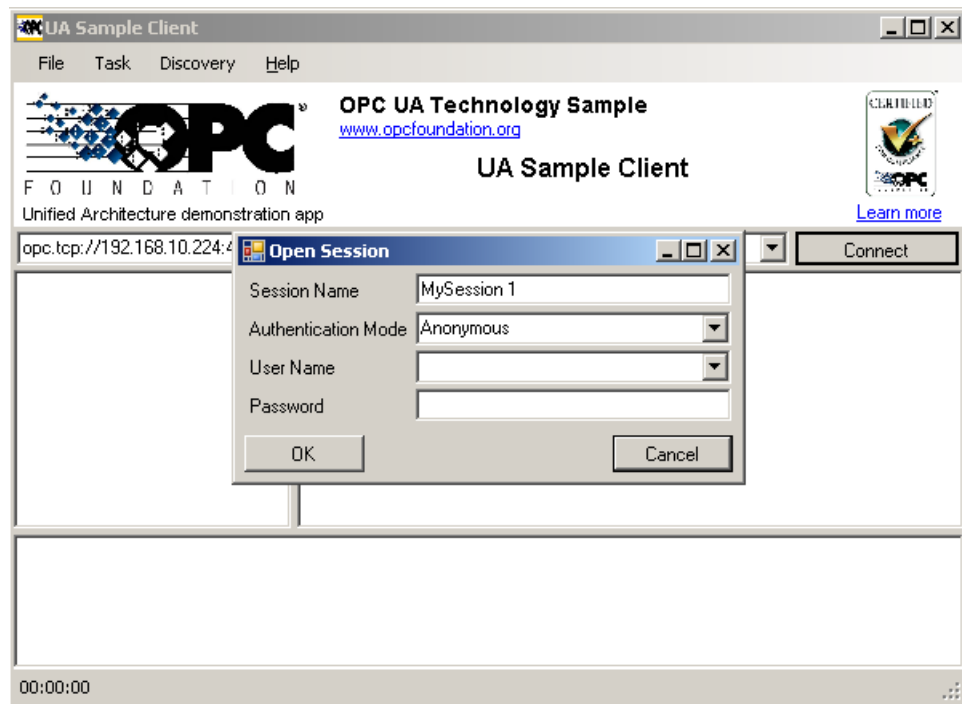


Figure 3-4 Open session

The connection to the SINUMERIK OPC UA server has now been established and the available address space is displayed.

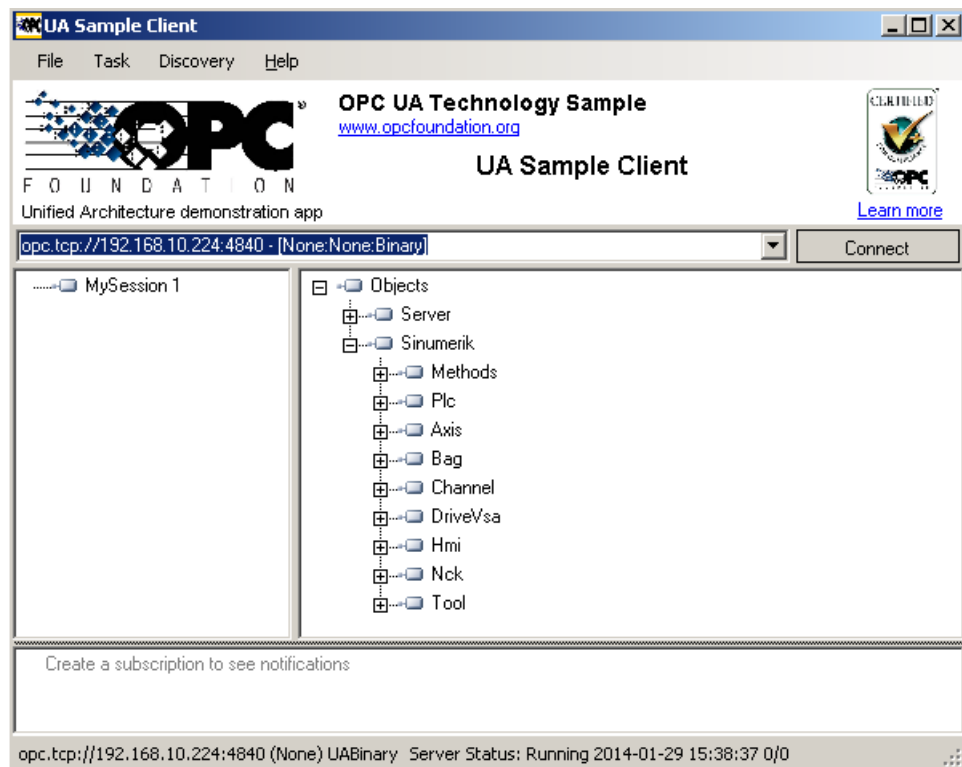


Figure 3-5 Address space of the SINUMERIK OPC UA server

8. Now navigate to a nodeID (e.g. R-parameter at Sinumerik > Channel > Parameter > R) and right-click the corresponding entry. You can now test various functions:
 - E.g. read, write, setup monitoring

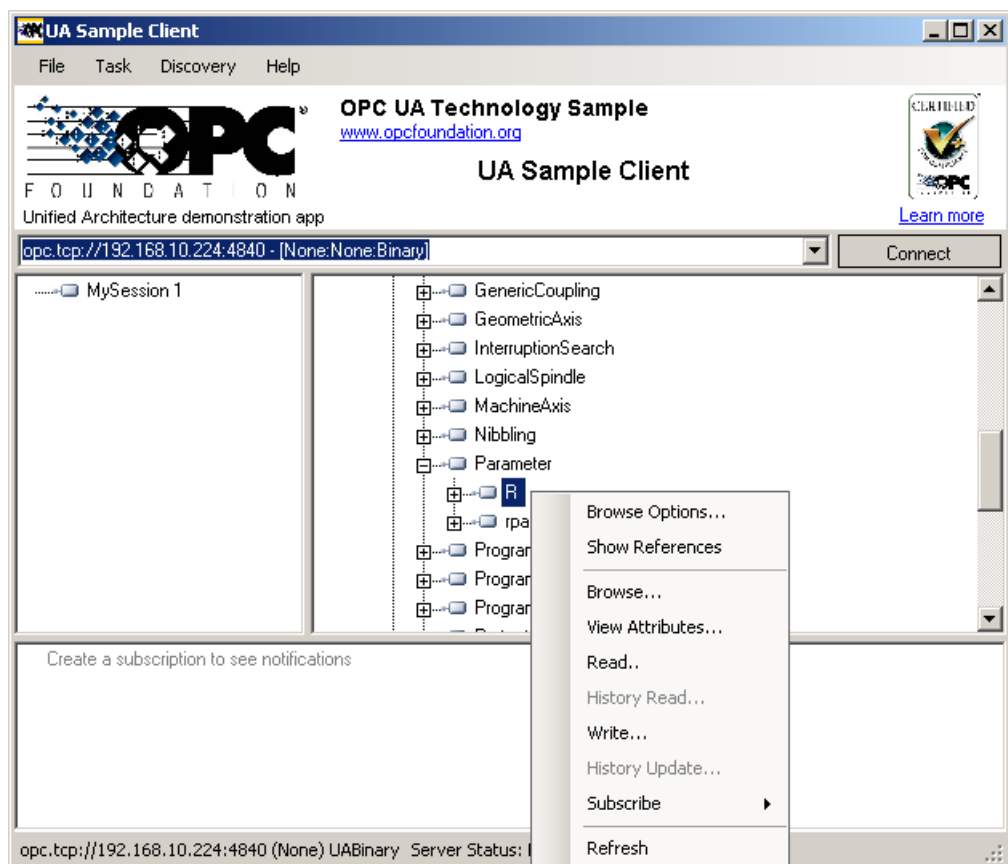


Figure 3-6 NodeID "Sinumerik > Channel > Parameter > R"

- The attributes of a nodeID can be viewed via the "View Attributes" entry. One of these attributes is "Value", which supplies the appropriate value of R1.

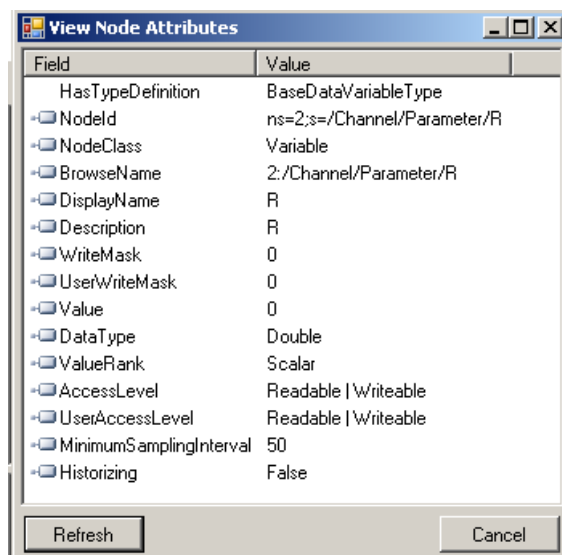


Figure 3-7 Viewing node attributes

OPC UA data access

Objective of the section

This section describes the functional scope and the SINUMERIK-specific special features of the OPC UA data access interface.

Note

The communication time is not guaranteed. As a consequence, realtime tasks cannot be implemented using the OPC UA data access interface.

Detailed information on the OPC UA specification is provided on the OPC Foundation (<https://opcfoundation.org/>) page.

4.1 Functional scope

Using the OPC UA data access interface, data can be accessed on the NC and PLC. In particular, variables can be read and written, and you can be notified when a value changes.

You can access individual variables (e.g. /Channel/Parameter/R[u1,1]) and array variables (e.g. /Channel/Parameter/R[u1,1,10]).

In addition, the following two methods are available for reading and writing:

- Sinumerik/Methods/ReadVar
- Sinumerik/Methods/WriteVar

Note

Limited number of simultaneously monitored variables

The number of variables from the "Sinumerik" object that can be monitored simultaneously is limited:

- SINUMERIK 828D, maximum 20 variables
- SINUMERIK 840D sl, maximum 200 variables

If you attempt to monitor more variables, the "BadWaitingForInitialData" error is returned.

4.1.1 Browsing

If the OPC UA data access interface is browsed, then the available address space is mapped under "Sinumerik".

Global User Data (GUD) can be found under the "/Sinumerik/GUD" node.

The PLC blocks (inputs, outputs, bit memory, data blocks) can be found under the "/Sinumerik/Plc" node.

Observe the following during browsing:

- In the address space of the NC, the displayed variables always represent only the first parameter of the corresponding first OPI unit.
Example: The R parameters are found at "Sinumerik > Channel > Parameter > R". The corresponding identifier is called: "/Channel/Parameter/R", which is eventually mapped to "/Channel/Parameter/R[u1, 1]". If you want to access other parameters, then you must correspondingly extend the identifier; you cannot directly accept the identifiers obtained when browsing, e.g. "/Channel/Parameter/R[u2, 56]".
- In the address space of the NC, the displayed variables represent the access format that has to be extended accordingly.
Example: The variable "/Plc/MB" is in the address space. This variable must be extended by the appropriate byte number, e.g. "/Plc/MB6". Only then is a value supplied.

- The address space of the NC also contains variables that are not available in a corresponding machine configuration. These variables return "BadAttributeIdInvalid" as value.
- The address space of the machine data and setting data cannot be accessed via browsing. How access is possible and how the identifiers are composed, is explained in Section Variable paths (Page 22).

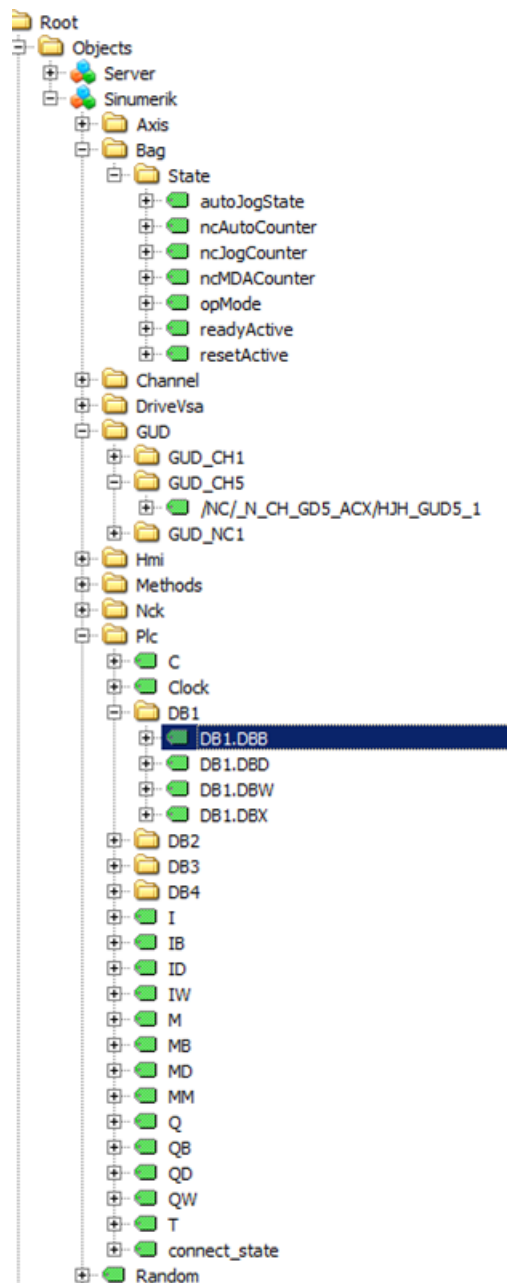


Figure 4-1 Excerpt from browsing the OPC UA data access interface

4.1.2 Variable paths

Variable paths for NC access operations

The variable paths for NC access are stored in the address space of the SINUMERIK Operate OPC UA server. Pay attention to the correct upper-case and lower-case of the "nodeID". The respective identifier of the "nodeID" provides information on the correct notation.

You can obtain additional information on this from the List Manuals for 840D sl and 828D sl "NC variable and interface signals".

Attribute	Value
[-] NodeId	NodeId
NamespaceIndex	2
IdentifierType	String
Identifier	/Channel/Parameter/R
NodeClass	Variable
BrowseName	2, "/Channel/Parameter/R"
DisplayName	"en_us", "R"
Description	"en_us", "R"
WriteMask	0
UserWriteMask	0
[-] Value	
SourceTimestamp	1/30/2014 3:17:25.822 PM
ServerTimestamp	1/30/2014 3:17:25.822 PM
SourcePicoSeconds	0
ServerPicoSeconds	0
Value	66
[-] DataType	Double
NamespaceIndex	0
IdentifierType	Numeric
Identifier	11
ValueRank	-1
ArrayDimensions	BadAttributeIdInvalid
AccessLevel	Readable, Writeable
UserAccessLevel	Readable, Writeable
MinimumSamplingInterval	50
Historizing	false

Figure 4-2 Identifier for R parameter

The displayed NC variables always represent only the first parameter of the corresponding NC data area (channel, TO area, mode group).

Example

The R parameters are found under the identifier "/Channel/Parameter/R", which is eventually mapped to "/Channel/Parameter/R[u1, 1]". If you want to access other parameters, then you must correspondingly extend the identifier, e.g. "/Channel/Parameter/R[u2, 56]".

Table 4-1 Examples of variable paths (NC accesses)

Variable path	Description
/Channel/Parameter/R[u1,10]	R parameter 10 in channel 1
/Channel/Parameter/R[u1,1,5]	R parameter array
/Channel/Parameter/R[u1,1,#5]	R parameters 1 to 5 in channel 1
/Channel/GeometricAxis/name[u2,3]	Name of the 3rd axis in channel 2
/Channel/GeometricAxis/actToolBasePos[u1,3]	Position of the 3rd axis in channel 1

Variable paths for GUD accesses

The variable paths for NC access are stored in the address space of the SINUMERIK Operate OPC UA server under the "/Sinumerik/GUD" node. Pay attention to the correct upper-case and lower-case of the "nodeID". The respective identifier of the "nodeID" provides information on the correct notation.

The displayed GUD variables always represent only the first parameter (for GUD arrays) of the first NC channel (for channel-dependent GUD variables). If you want to access a different parameter of a GUD array or a different channel, you must extend the identifier accordingly for the NC access.

GUD arrays are 1-indexed for access, and access is always one-dimensional. This means, the index must be calculated for multi-dimensional arrays.

Example 1: One-dimensional array, NC-global GUD array

"UGUD.DEF" file

```
DEF NCK INT ARRAY[2]
M17
```

Access is performed as follows:

```
ARRAY[0] → /NC/_N_NC_GD3_ACX/ARRAY[1]
ARRAY[1] → /NC/_N_NC_GD3_ACX/ARRAY[2]
```

Example 2: Two-dimensional array, channel-dependent GUD array

"UGUD.DEF" file

```
DEF CHAN INT ABC[3,3]
M17
```

Access is performed as follows:

```
ABC[0,0] → /NC/_N_CH_GD3_ACX/ABC[u1, 1]
ABC[0.1] → /NC/_N_CH_GD3_ACX/ABC[u1, 2]
ABC[0.2] → /NC/_N_CH_GD3_ACX/ABC[u1, 3]
ABC[1.0] → /NC/_N_CH_GD3_ACX/ABC[u1, 4]
ABC[1.1] → /NC/_N_CH_GD3_ACX/ABC[u1, 5]
ABC[1.2] → /NC/_N_CH_GD3_ACX/ABC[u1, 6]
ABC[2.0] → /NC/_N_CH_GD3_ACX/ABC[u1, 7]
ABC[2.1] → /NC/_N_CH_GD3_ACX/ABC[u1, 8]
ABC[2.2] → /NC/_N_CH_GD3_ACX/ABC[u1, 9]
```

Variable paths for PLC access operations

The variable paths for PLC access are stored in the address space of the SINUMERIK Operate OPC UA server under the "/Sinumerik/PLC" node. Pay attention to the correct upper-case and lower-case of the "nodeID". The respective identifier of the "nodeID" provides information on the correct notation.

In the address space of the NC, the displayed variables represent the access format that has to be extended accordingly.

Example

The variable "/Plc/MB" is in the address space. This variable must be extended by the appropriate byte number, e.g. to "/Plc/MB6". Only then is a value supplied.

The various access formats are shown in the following table. These access formats must be extended by the prefix "/Plc".

Note

The data type is converted during access with the OPC UA data access interface. Refer to the following table for the data type conversions.

Table 4-2 PLC syntax

Area	Address (IEC)	Permissible data types	OPC UA data type
Output image	Qx.y	BOOL	Boolean
Output image	QBx	BYTE , CHAR, STRING	UInt32 String
Output image	QWx	WORD , CHAR, INT,	UInt32 Int32
Output image	QDx	DWORD , DINT, REAL	UInt32 Int32 Double
Data block	DBz.DBXx.y	BOOL	Boolean
Data block	DBz.DBBx	BYTE , CHAR, STRING	UInt32 String
Data block	DBz.DBWx	WORD , CHAR, INT	UInt32 Int32
Data block	DBz.DBDx	DWORD , DINT, REAL	UInt32 Int32 Double
Input image	Ix.y	BOOL	Boolean
Input image	IBx	BYTE , CHAR, STRING	UInt32 String
Input image	IWx	WORD , CHAR, INT	UInt32 Int32
Input image	IDx	DWORD , DINT, REAL	UInt32 Int32 Double
Bit memory	Mx.y	BOOL	Boolean
Bit memory	MBx	BYTE , CHAR, STRING	UInt32 String
Bit memory	MWx	WORD , CHAR, INT	UInt32 Int32

Area	Address (IEC)	Permissible data types	OPC UA data type
Bit memory	MDx	DWORD , DINT, REAL	UInt32 Int32 Double
Counters	Cx	WORD	UInt32
timers	Tx	-	Int32

Notes regarding the table:

- In the table, "x" stands for the byte offset in the area; "y" for the bit number in the byte; and "z" for the data block.
- The **bold data type** is the default data type in each case and does not have to be specified when addressing. In addition, the specifications DB2.DBB5.BYTE and DB2.DBB5 are equivalent, for example.
- Square brackets are used to access arrays, e.g. "/Plc/DB5.DBW2:[10]" (word array of length 10).

Table 4-3 Examples of variable paths (PLC accesses)

Variable path	Description
/Plc/M5.0	Memory bit 0 at byte offset 5
/Plc/DB5.DBW2	Word (16-bit) at byte offset 2 in data block 5
/Plc/DB8.DBB2:STRING	UTF8 string beginning at byte offset 2 in data block 8
/Plc/DB8.DBW2:[10]	Array of 10 words beginning at byte offset 2 in data block 8
/Plc/DB100.DBB1	Byte at byte offset 1 in data block 100
/Plc/DB2.DBD0:REAL[10]	Array of 10 double words (32-bit) beginning at byte offset 0 in data block 2, which are formatted as a floating-point number

Note the following:

Note

- Timers can only be read. A timer is active if it contains a value other than 0.
- If the data type CHAR or STRING is used in conjunction with a byte access, UTF8 characters are read, but if either data type is used in conjunction with a word access, UTF16 characters are read.
- Variables of the STRING type contain the maximum length in the first byte and the actual length in the second byte. When strings are written, the actual length is adapted accordingly. The maximum length is not changed.
- For the STRING data type in conjunction with a byte access (e.g. "/Plc/DB99.DBB0:STRING"), the maximum string length is 255 characters. As a result of the UTF8 formatting, for some characters (e.g. for the "µ"), two bytes are required, so that the maximum string length is correspondingly reduced.
- Only one-dimensional arrays are supported. With STRING arrays, all elements have the same maximum length.

Variable paths for machine and setting data

The variable paths of the machine data and setting data cannot be accessed via browsing.

4.1 Functional scope

The variable path for reading, writing and monitoring machine data and setting data comprises a defined prefix and the machine data or setting data name.

The following prefixes are possible:

Table 4-4 Prefixes for machine and setting data

Area	Prefix
General machine data	/Random/NC/_N_NC_TEA_ACX
Channel-specific machine data	/Random/NC/_N_CH_TEA_ACX
Axis-specific machine data	/Random/NC/_N_AX_TEA_ACX
General setting data	/Random/NC/_N_NC_SEA_ACX
Channel-specific setting data	/Random/NC/_N_CH_SEA_ACX
Axis-specific setting data	/Random/NC/_N_AX_SEA_ACX

Machine data arrays are 1-indexed for access.

Table 4-5 Examples of variable paths (machine and setting data access operations)

Variable path	Description
/Random/NC/_N_NC_TEA_ACX/\$MN_IPO_CYCLE_TIME	IPO cycle clock (general MD)
/Random/NC/_N_CH_TEA_ACX/\$MC_DISPLAY_AXIS[u2, 1]	Display axis on HMI (channel-specific MD)
/Random/NC/_N_AX_SEA_ACX/\$SA_LEAD_TYPE[u2]	Type of master value (axis-specific SD)

4.1.3 Supported services

The following services of the OPC UA data access specification are supported:

- ActivateSessionService
- AddNodesService
- AddReferencesService
- BrowseNextService
- BrowseService
- CallService
- CloseSessionService
- CreateMonitoredItemsService
- CreateSessionService
- CreateSubscriptionService
- DeleteMonitoredItemsService
- DeleteNodesService

- DeleteReferencesService
- DeleteSubscriptionService
- FindServersService
- GetEndpointsService
- ModifyMonitoredItemsService
- PublishService
- ReadService
- RegisterNodesService
- RepublishService
- SetMonitoringModeService
- SetPublishingModeService
- TransferSubscriptionsService
- TranslateBrowsePathsToNodeIdsService
- UnregisterNodesService
- WriteService

4.1.4 Special features with regard to the OPC UA specification

TranslateBrowsePathsToNodeIdsService

If a non-existing nodeId is specified in TranslateBrowsePathsToNodeIds, then "BadNoMatch" is returned instead of BadNodeIdUnknown".

Index

A

Architecture, 5

B

BadWaitingForInitialData, 20
Binary protocol, 5

C

Communication port, 10

D

Data types, 24

L

License, 10

M

Method
 ReadVar, 20
 WriteVar, 20

O

OPC_UAApplication.xml, 8

R

ReadVar, 20

W

WriteVar, 20

SINUMERIK

SINUMERIK 840D sl/828D SINUMERIK Integrate Run MyScreens (BE2)

Programming Manual

<u>Introduction</u>	1
<u>Getting Started</u>	2
<u>Fundamentals</u>	3
<u>Dialogs</u>	4
<u>Variables</u>	5
<u>Programming commands</u>	6
<u>Graphic and logic elements</u>	7
<u>"Custom" operating area</u>	8
<u>Dialog selection</u>	9
<u>Reference lists</u>	A
<u>Tips and tricks</u>	B
<u>Animated elements</u>	C

Valid for:

CNC software

V4.7 SP1

01/2015

6FC5397-1DP40-5BA2

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

DANGER

indicates that death or severe personal injury will result if proper precautions are not taken.

WARNING

indicates that death or severe personal injury may result if proper precautions are not taken.

CAUTION

indicates that minor personal injury can result if proper precautions are not taken.

NOTICE

indicates that property damage can result if proper precautions are not taken.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

WARNING

Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Table of contents

1	Introduction.....	9
2	Getting Started.....	13
2.1	Introduction.....	13
2.2	Example.....	13
2.2.1	Task description.....	13
2.2.2	Creating the configuration file.....	17
2.2.3	Saving the configuration file in the OEM directory.....	20
2.2.4	Creating the online help.....	21
2.2.5	Saving the online help in the OEM directory.....	23
2.2.6	Copying easyscreen.ini into the OEM directory.....	23
2.2.7	Registering the COM file in easyscreen.ini.....	24
2.2.8	Testing the project.....	24
3	Fundamentals.....	25
3.1	Structure of configuration file.....	25
3.2	Structure of the menu tree.....	27
3.3	Definition and functions for start softkeys.....	28
3.3.1	Defining the start softkey.....	28
3.3.2	Functions for start softkeys.....	29
3.4	Troubleshooting (log book).....	31
3.5	Notes on the easyscreen.ini.....	32
3.6	Notes for personnel changing over to "Run MyScreens".....	34
3.7	Extended configuration syntax.....	36
4	Dialogs.....	39
4.1	Structure and elements of a dialog.....	39
4.1.1	Defining a dialog.....	39
4.1.2	Defining dialog properties.....	41
4.1.3	Defining dialog elements.....	45
4.1.4	Defining dialogs with multiple columns.....	47
4.1.5	Password dialogs.....	48
4.1.6	Using display images/graphics.....	50
4.2	Defining softkey menus.....	50
4.2.1	Changing softkey properties during runtime.....	53
4.2.2	Language-dependent text.....	56
4.3	Configuring the online help.....	58
5	Variables.....	61
5.1	Defining variables.....	61
5.2	Application examples.....	62

5.3	Example 1: Assigning the variable type, texts, help display, colors, tooltips.....	63
5.4	Example 2: Assigning the Variable Type, Limits, Attributes, Short Text Position properties.....	64
5.5	Example 3: Assigning the Variable Type, Default, System or User Variable, Input/Output Field Position properties.....	65
5.6	Example 4: Toggle field and list field.....	65
5.7	Example 5: Image display.....	66
5.8	Example 6: Progress bar.....	66
5.9	Example 7: Password input mode (asterisk).....	69
5.10	Variable parameters.....	69
5.11	Details on the variable type.....	75
5.12	Details on the toggle field.....	78
5.13	Details on the default setting.....	79
5.14	Details on the position of the short text, position of the input/output field.....	80
5.15	Use of strings.....	80
5.16	CURPOS variable.....	82
5.17	CURVER variable.....	82
5.18	ENTRY variable.....	83
5.19	ERR variable.....	83
5.20	FILE_ERR variable.....	84
5.21	FOC variable.....	85
5.22	Variable S_ALEVEL.....	86
5.23	S_CHAN variable.....	87
5.24	Variable S_CONTROL.....	87
5.25	Variable S_LANG.....	88
5.26	Variable S_NCCODEREADONLY.....	89
5.27	Variables S_RESX and S_RESY.....	89
6	Programming commands.....	91
6.1	Operators.....	91
6.1.1	Mathematical operators.....	91
6.1.2	Bit operators.....	94
6.2	Methods.....	95
6.2.1	ACCESSLEVEL.....	96
6.2.2	CHANGE.....	96
6.2.3	CHANNEL.....	98
6.2.4	CONTROL.....	98
6.2.5	FOCUS.....	99
6.2.6	LANGUAGE.....	100

6.2.7	LOAD.....	100
6.2.8	UNLOAD.....	101
6.2.9	OUTPUT.....	102
6.2.10	PRESS.....	103
6.2.11	PRESS(ENTER).....	104
6.2.12	PRESS(TOGGLE).....	104
6.2.13	RESOLUTION.....	105
6.2.14	RESUME.....	105
6.2.15	SUSPEND.....	106
6.2.16	Example: Version management with OUTPUT methods.....	106
6.3	Functions.....	108
6.3.1	Reading and writing drive parameters: RDOP, WDOP, MRDOP.....	108
6.3.2	Subprogram call (CALL).....	110
6.3.3	Define block (/B).....	111
6.3.4	Check Variable (CVAR).....	112
6.3.5	CLEAR_BACKGROUND.....	113
6.3.6	Copy Program file function (CP).....	114
6.3.7	Delete Program file function (DP).....	114
6.3.8	Exist Program file function (EP).....	115
6.3.9	Move Program file function (MP).....	116
6.3.10	Select Program file function (SP).....	117
6.3.11	File accesses: RDFILE, WRFILE, RDLINEFILE, WRLINEFILE.....	118
6.3.12	Dialog line (DLGL).....	120
6.3.13	DEBUG.....	121
6.3.14	Exit dialog (EXIT).....	122
6.3.15	Dynamic manipulation of the lists of toggle fields or list box fields.....	123
6.3.16	Evaluate (EVAL).....	126
6.3.17	Exit Loading Softkey (EXITLS).....	127
6.3.18	Function (FCT).....	127
6.3.19	Generate code (GC).....	129
6.3.20	Password functions.....	132
6.3.21	Load Array (LA).....	133
6.3.22	Load Block (LB).....	134
6.3.23	Load Mask (LM).....	135
6.3.24	Load Softkey (LS).....	136
6.3.25	Load Grid (LG).....	137
6.3.26	Multiple selection SWITCH.....	138
6.3.27	Multiple Read NC PLC (MRNP).....	139
6.3.28	PI services.....	141
6.3.29	Reading (RNP) and writing (WNP) system or user variables.....	142
6.3.30	RESIZE_VAR_IO and RESIZE_VAR_TXT.....	143
6.3.31	Register (REG).....	144
6.3.32	RETURN.....	145
6.3.33	Recompile.....	146
6.3.34	Recompile without user comment.....	147
6.3.35	Search Forward, Search Backward (SF, SB).....	150
6.3.36	STRING functions.....	151
6.3.37	WHILE/UNTIL loops.....	157
6.3.38	Cyclic execution of scripts: START_TIMER, STOP_TIMER.....	160
7	Graphic and logic elements.....	163
7.1	Line, dividing line, rectangle, circle and ellipse.....	163

7.2	Defining an array.....	165
7.2.1	Accessing the value of an array element.....	166
7.2.2	Example Access to an array element.....	168
7.2.3	Scanning the status of an array element.....	170
7.3	Table description (grid).....	170
7.3.1	Defining a table (grid).....	172
7.3.2	Defining columns.....	172
7.3.3	Focus control in the table (cell).....	174
7.4	Custom widgets.....	175
7.4.1	Defining custom widgets.....	175
7.4.2	Structure of the custom widget library.....	175
7.4.3	Structure of the custom widget interface.....	176
7.4.4	Interaction between custom widget and dialog box - Automatic data exchange.....	178
7.4.5	Interaction between custom widget and dialog box - Manual data exchange.....	179
7.4.5.1	Reading and writing properties.....	179
7.4.5.2	Executing a method of the custom widget.....	180
7.4.5.3	Response to a custom widget signal.....	182
8	"Custom" operating area.....	185
8.1	How to activate the "Custom" operating area	185
8.2	How to configure the "Custom" softkey.....	185
8.3	How to configure the "Custom" operating area.....	186
8.4	Programming example for the "Custom" area.....	188
9	Dialog selection.....	193
9.1	Dialog selection using PLC softkeys.....	193
9.2	Dialog selection using PLC hard keys.....	194
9.3	Dialog selection via NC.....	197
A	Reference lists.....	199
A.1	Lists of start softkeys.....	199
A.1.1	List of start softkeys for turning.....	199
A.1.2	List of start softkeys for milling.....	200
A.2	List of access levels.....	202
A.3	List of colors.....	202
A.4	List of language codes used in file names.....	203
A.5	List of accessible system variables.....	204
A.6	Behavior when opening the dialog (attribute CB).....	204
B	Tips and tricks.....	207
B.1	General tips.....	207
B.2	Tips for debugging.....	208
B.3	Tips for the CHANGE method.....	208
B.4	Tips for DO LOOP loops.....	209

C	Animated elements.....	211
C.1	Introduction.....	211
C.2	Modeling.....	212
C.2.1	Requirements.....	212
C.2.2	Rules for modeling.....	212
C.2.3	Importing graphics (models).....	215
C.2.4	Modeling templates.....	217
C.3	XML commands.....	218
C.3.1	Overview.....	218
C.3.2	Structure of the scene description file.....	219
C.3.3	Mirroring and rotations.....	222
C.3.4	View type.....	222
C.4	Conversion to hmi file.....	223
C.5	Display in Create MyHMI /3GL.....	223
C.5.1	X3D Viewer.....	223
C.5.2	Class SIX3dViewerWidget.....	223
C.5.3	Public methods.....	224
C.5.4	Public slots.....	224
C.5.5	Libraries.....	225
C.5.6	Implementation example.....	225
C.5.7	Machine data.....	225
C.5.8	Notes about use.....	226
C.6	Display in Run MyScreens.....	226
	Glossary.....	229
	Index.....	233

Introduction

Overview

"Run MyScreens" can be used to create user interfaces that display functional expansions designed by the machine manufacturer or user, or to implement your own layout.

Cycle calls can also be generated with user interfaces that you have created. Dialogs can be created directly on the control system.

"Run MyScreens" is implemented with an Interpreter and configuration files that describe the user interfaces.

"Run MyScreens" is configured using ASCII files: These configuration files contain the description of the user interface. The syntax that must be applied in creating these files is described in the following chapters.

Basic configuration

The "Run MyScreens" function enables machine manufacturers to configure their own dialogs. Even with the basic configuration, it is possible to configure five dialogs in the operator menu tree or for customer-specific cycle dialogs.



Software option

To expand the number of dialogs, you require one of the following software options:

- SINUMERIK 828D/840D sl, SINUMERIK Integrate Run MyScreens (6FC5800-0AP64-0YB0)
- SINUMERIK 840D sl, SINUMERIK Integrate Run MyScreens + Run MyHMI (6FC5800-0AP65-0YB0)
- SINUMERIK 840D sl, SINUMERIK Integrate Run MyHMI / 3GL (6FC5800-0AP60-0YB0)
- SINUMERIK 840D sl, SINUMERIK Integrate Run MyHMI / WinCC (6FC5800-0AP61-0YB0)

Supplementary conditions

The following conditions must be met:

- It is only possible to switch between dialogs within a single operating area.
- User variables may not have the same names as system or PLC variables (also see List Manual System Variables /PGAs/).
- The dialogs activated by the PLC form a separate operating area (similar to measuring cycle screens).

Tools

- UTF8-capable editor (e.g. the integrated editor of the user interface or Notepad)
- A graphics program is needed to create graphics/display images.

File names and coding

Note

When using HMI Operate in the NCU, note that all file names are written in lower case letters (com, png, txt) on the CF card. This is required because of Linux.

On the PCU, you can write file names either in upper case or lower case letters. However, it is recommended that you also use lower case letters here, in case of a possible subsequent transfer to Linux.

Note

Make sure that the coding is set to UTF-8 in the editor that you are using when saving the configuration and language files.

Storage paths

Note the following convention when storing the configuration files, language files, etc.

<i>[System Siemens directory]</i>		
	Linux:	/card/siemens/sinumerik/hmi
	Windows 7:	C:\ProgramData\Siemens\MotionControl\siemens
<i>[System oem directory]</i>		
	Linux:	/card/oem/sinumerik/hmi
	Windows 7:	C:\ProgramData\Siemens\MotionControl\oem
<i>[System user directory]</i>		
	Linux:	/card/user/sinumerik/hmi
	Windows 7:	C:\ProgramData\Siemens\MotionControl\user
<i>[System add_on directory]</i>		
	Linux:	/card/addon/sinumerik/hmi
	Windows 7:	C:\ProgramData\Siemens\MotionControl\addon
Storage locations		
	"Run MyScreens" configuration:	<i>[System oem directory]</i> \proj
	Configuration file:	<i>[System oem directory]</i> \cfg
	Language files:	<i>[System oem directory]</i> \lng

	Image files:	<i>[System oem directory]\ico\ico[Resolution]</i> Example: <i>[System oem directory]\ico\ico640</i>
	Online Help:	<i>[System oem directory]\hlp/[Language]</i> Example: <i>[System oem directory]\hlp/eng</i>

Application

You can implement the following functions:

Display dialogs containing the following elements:	• Softkeys • Variables • Texts and Help texts • Graphics and Help displays
Open dialogs by:	• Pressing the (start) softkeys • Selection from the PLC/NC
Restructure dialogs dynamically:	• Edit and delete softkeys • Define and design variable fields • Insert, exchange and delete display texts (language-dependent or independent) • Insert, exchange and delete graphics
Initiate operations in response to the following actions:	• Displaying dialogs • Input values (variables) • Select a softkey • Exiting dialogs • Value change of a system or user variable
Data exchange between dialogs:	
Variables:	• Read (NC, PLC and user variables) • Write (NC, PLC and user variables) • Combine with mathematical, comparison or logic operators
Execute functions:	• Subprograms • File functions • PI services
Apply protection levels according to user classes	

Getting Started

2.1 Introduction

Using the following example, you get to know the steps necessary to insert your own dialogs into the SINUMERIK Operate user interface using Run MyScreens. You also learn how to create your own dialogs, insert context-sensitive help screens and help calls, define softkeys and how you can navigate between the dialogs.

2.2 Example

2.2.1 Task description

You will create the following dialogs in the example.

Dialog 1

R parameters that can be written to (0 and 1) and geometry axis names (input fields) are displayed in the first dialog. The corresponding help screens are linked in for the two R parameters. A context-sensitive help is linked in for the geometry axes. The dialog also contains examples of dividing lines (horizontal and vertical), toggle fields, I/O fields with integrated unit selection and for progress bars (with and without color change).

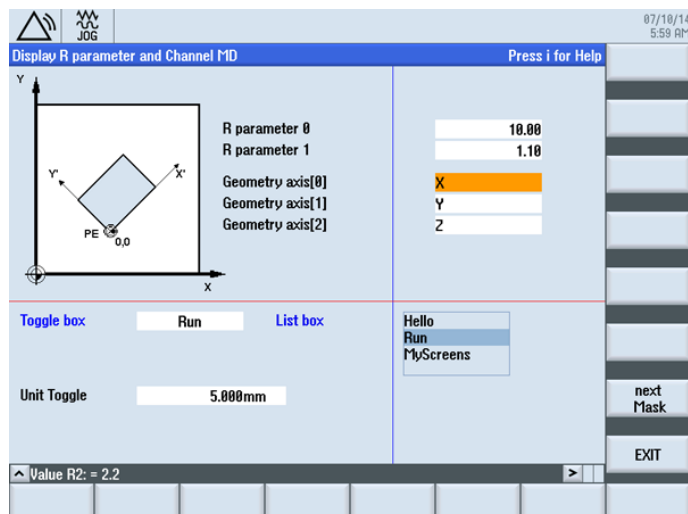


Figure 2-1 R parameters with help displays

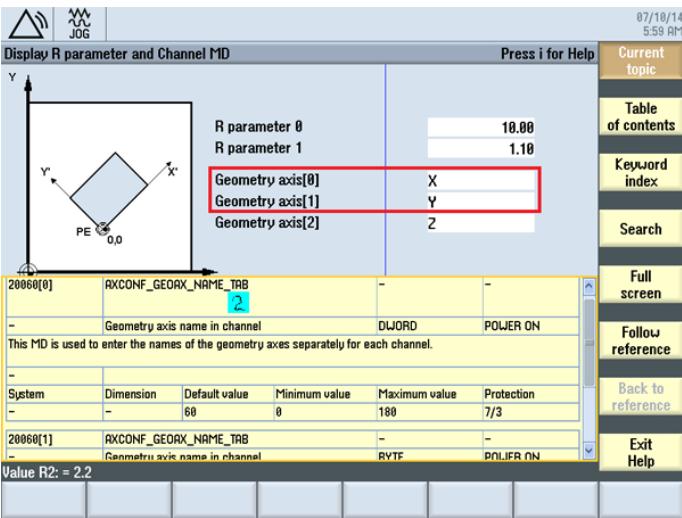


Figure 2-2 Geometry axes with context sensitive help

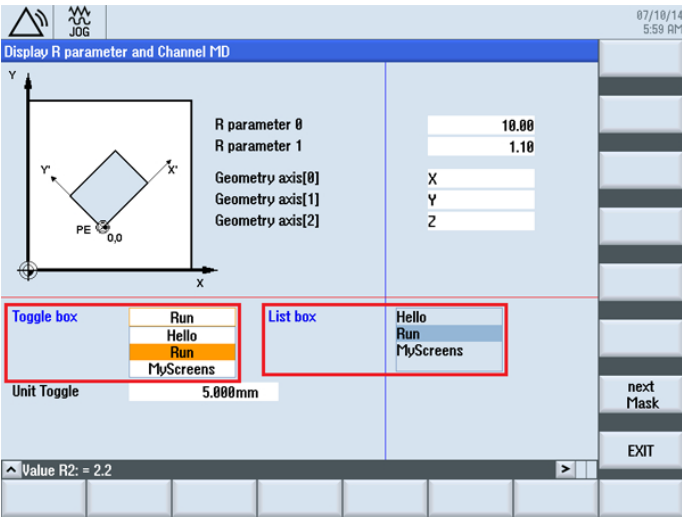


Figure 2-3 Toggle fields: List and box

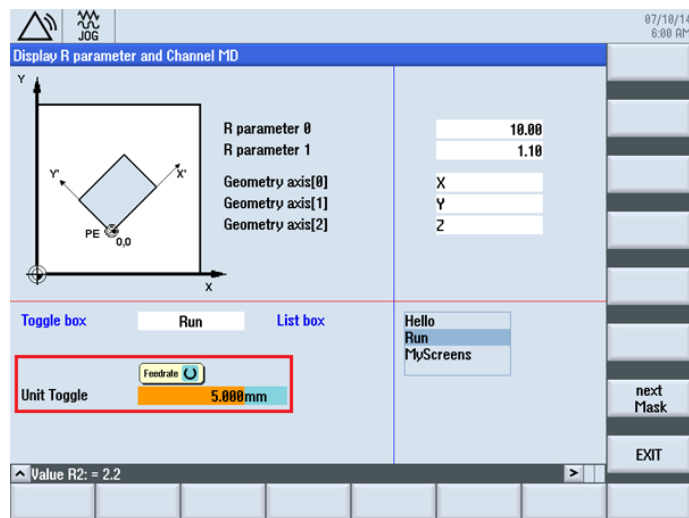


Figure 2-4 I/O field with integrated unit selection

Dialog 2

MCS and WCS values are displayed in the second dialog. The dialog also contains examples of progress bars with and without color change.

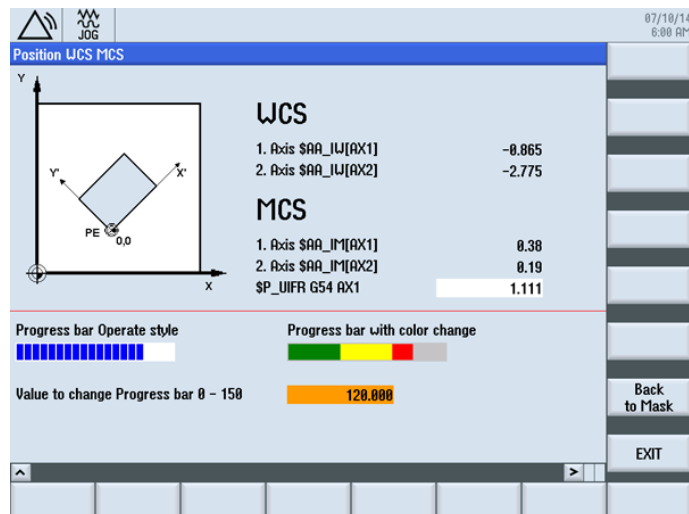


Figure 2-5 Progress bars with (right) and without (left) color change

Navigation

The first dialog is called using the "START" softkey in the diagnostics operating area. The horizontal SK7 softkey is used.

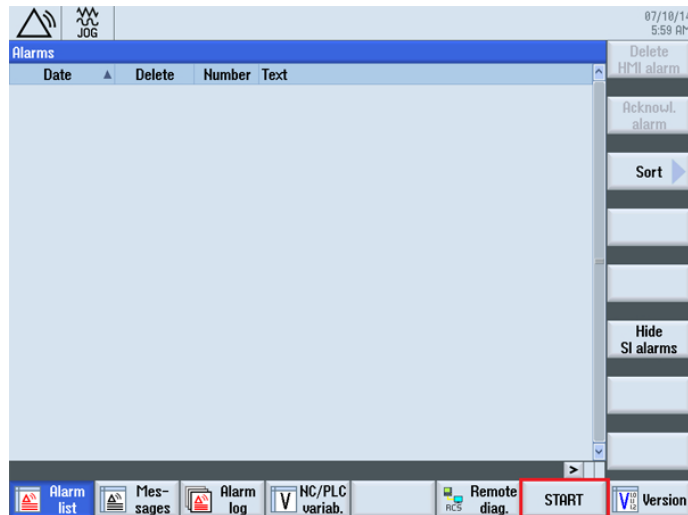


Figure 2-6 Start softkey „START“ in the machine operating area, AUTO mode

You can call the second dialog from the first dialog with the "Next Mask" softkey. With the "EXIT" softkey, you can return to the root screen of the operating area (see diagram above). Using the "EXIT" softkey, you can also return to the root screen of the operating area from the second dialog (see the diagram above). You can return to the first dialog via the "Back to Mask" softkey.

Procedure

The necessary steps are described in the following chapters:

1. Creating the configuration file (COM file)
2. Saving the configuration file in the OEM directory
3. Creating the online help
4. Saving the online help in the OEM directory
5. Copying easyscreen.ini into the OEM directory
6. Registering the COM file in easyscreen.ini
7. Testing the project

2.2.2 Creating the configuration file

Content of the configuration file

Using a UTF8-capable editor, create the configuration file `diag.com` for the two dialogs.

```
; Start identifier, start softkey
//S(START)
; Start softkey only text
HS7=("START")
; Start softkey with language-dependent text and png
;HS7=([$80792,"\\sk_ok.png"])

; Press method
PRESS(HS7)
; LM or LS function
LM("MASK1")
; LM, specifying a com file
LM("MASK1","TEST.COM")
END_PRESS
; End identifier of start softkey
//END

; Definition of dialog 1 with header and screen
//M(MASK1/"Display R parameter and Channel MD"/"mz961_01.png")
; Definition of the variables
DEF VAR1 = (R2///,"R parameter 0"///"$R[0]"/200,50,150/400,50,100,)
; With help screen
DEF VAR2 = (R2///,"R parameter 1"///"mz961_02.png"/"$R[1]"/200,70,150/400,70,100)
; With online help
DEF ACHS_NAM1 = (S///"Press i for Help","Geometry axis[0]"/200,100,150/400,100,100/"sinumer-
ik_md_1.html","20060[0]")
; With online help
DEF ACHS_NAM2 = (S///"Press i for Help","Geometry axis[1]"/200,120,150/400,120,100/"sinumer-
ik_md_1.html","20060[1]")
DEF ACHS_NAM3 = (S///,"Geometry axis[2]"/200,140,150/400,140,100)
; Definition of toggle fields and unit toggle
DEF VAR_TGL = (S/* "Hello", "Run", "MyScreens"/"Run"/,"Toggle box"/10,230,100/120,230,100/, ,6)

DEF VAR_TGB = (S/* "Hello", "Run", "MyScreens"/"Run"/,"List box"/
dt4///250,230,100/370,230,100,60/, ,"#0602ee")
; BC, FC, BC_ST, FC_ST, BC_GT, FC_GT, BC_UT, FC_UT, SC1, SC2
DEF VarEdit = (R///,"Unit Toggle",,,"Feedrate"///"$R[11]"/5,300,100/120,300,100/"VarTgl"),
VarTgl = (S/*0="mm",1="inch"/0//WR2///220,300,40)
```

```
; Softkey definition in the dialog
HS1=("")
HS2=("")
HS3=("")
HS4=("")
HS5=("")
HS6=("")
HS7=("")
HS8=("")
VS1=("")
VS2=("")
VS3=("")
VS4=("")
VS5=("")
VS6=("")
VS7="Next Mask")
VS8="EXIT")

; Definition LOAD block
LOAD
    ; Read value with RNP
    ACHS_NAM1 = RNP("$MC_AXCONF_GEOAX_NAME_TAB[0]")
    ACHS_NAM2 = RNP("$MC_AXCONF_GEOAX_NAME_TAB[1]")
    ACHS_NAM3 = RNP("$MC_AXCONF_GEOAX_NAME_TAB[2]")
    ; Output of a dialog line
    DLGL("Value R2: = " << RNP("$R[2]"))
    ; Write WNP to a value
    WNP("$R[3]",VAR0)

    ; Vertical dividing line
    V_SEPARATOR(360,1,6,1)
    ; Horizontal dividing line
    H_SEPARATOR(220,1,7,1)
END_LOAD

; Press method
PRESS(VS7)
    ; Load an additional dialog
    LM("MASK2")
; End identifier press method
END_PRESS

; Press method
PRESS(VS8)
```

```

; Exit the dialog
EXIT
; End identifier press method
END_PRESS
; End identifier dialog 1
//END

; Definition of dialog 2 with header and screen
//M(MASK2/"Position WCS MCS"/"mz961_01.png")

; Definition of the variables
DEF TEXT1 = (I///,"WCS"/WR0,fs2///230,30,120/, ,1)
DEF VAR1 = (R3///,"1. Axis $AA_IW[AX1]"/WR1/"$AA_IW[AX1]"/230,70,150/400,70,100)
DEF VAR2 = (R3///,"2. Axis $AA_IW[AX2]"/WR1/"$AA_IW[AX2]"/230,90,150/400,90,100)

DEF TEXT2 = (I///,"MCS"/WR0,fs2///230,120,120/, ,1)
DEF VAR3 = (R2///,"1. Axis $AA_IM[AX1]"/WR1/"$AA_IM[AX1]"/230,160,150/400,160,100)
DEF VAR4 = (R2///,"2. Axis $AA_IM[AX2]"/WR1/"$AA_IM[AX2]"/230,180,150/400,180,100)
DEF VAR5 = (R3///,"$P_UIFR G54 AX1"///"$P_UIFR[1,AX1,TR]"/230,200,150/400,200,100)

; Definition of the progress bar
DEF PROGGY0 = (R/0,150//,"Progress bar Operate style"/DT2,DO0/"$R[10]"/5,240,190/5,260,150/6,10)
DEF PROGGY4 = (R/0,150,50,100/120//,"Progress bar with color change"/DT1,DO0/"$R[10]"/
260,240,190/260,260,150/3,4,,,9,7)

; Definition of the variables
DEF PROGVAL = (R/0,150//,"Value to change Progress bar 0 - 150"///"$R[10]"/5,300,230/260,300,100)

; Softkey definition in the dialog
HS1=("")
HS2=("")
HS3=("")
HS4=("")
HS5=("")
HS6=("")
HS7=("")
HS8=("")
VS1=("")
VS2=("")
VS3=("")
VS4=("")
VS5=("")
VS6=("")
VS7=("Back to Mask")
VS8=("EXIT")

```

```
; Load method
LOAD
    H_SEPARATOR(230,1,7,1)
END_LOAD

; Change method
CHANGE(VAR5)
    ; Function PI_START
    PI_START("/NC,201,_N_SETUFR")
; End identifier change method
END_CHANGE

; Press method
PRESS(RECALL)
    ; Return to the calling mask
    LM("MASK1")
; End identifier press method
END_PRESS

; Press method
PRESS(VS7)
    ; Return to the calling mask
    LM("MASK1")
; End identifier press method
END_PRESS

; Press method
PRESS(VS8)
    ; Exit the dialog to the standard application
    EXIT
; End identifier press method
END_PRESS
; Dialog end identifier
//END
```

2.2.3 Saving the configuration file in the OEM directory

Storage path

Save the configuration file diag.com under the following path:

[System oem directory]\proj

2.2.4 Creating the online help

Content of the online help

Create the HTML file `sinumerik_md_1.html`. The section below shows the context-sensitive help call via **`name="20060[0]"`** and **`name="20060[1]"`**. The help file is not commented, as this manual does not discuss creating online help. Notes regarding the integration of online help are provided in Section Configuring the online help (Page 58).

```

<html><head><meta http-equiv="Content-Type" content="text/html; charset="UTF-8"/><title></
title></head>
<body>
  <table border="1" cellspacing="0" cellpadding="2" width="100%">
    <tr>
      <td><a name="20060[0]"><b>20060[0]</b></a></td>
      <td colspan="3">AXCONF_GEOAX_NAME_TAB</td>
      <center>
        
      </center>
      <td>--</td>
      <td>--</td>
    </tr>
    <tr>
      <td>--</td>
      <td colspan="3">Geometry axis name in channel</td>
      <td>DWORD</td>
      <td>POWER ON</td>
    </tr>
    <tr>
      <td colspan="6">This MD is used to enter the names of the geometry axes
separately for each channel.<br />
    <tr>
      <td>--</td>
      <td colspan="5">&nbsp;</td>
    </tr>
    <tr>
      <td width="16*">System</td><td width="16*">Dimension</td><td width="16*">Default
value</td>
      <td width="16*">Minimum value</td><td width="16*">Maximum value</td>
      <td width="16*">Protection</td>
    </tr>
    <tr>
      <td>--</td>
      <td>--</td>
      <td>60</td>
      <td>0</td>
      <td>180</td>
      <td>7/3</td>
    </tr>
  </table>
<p></p>
  <table border="1" cellspacing="0" cellpadding="2" width="100%">
    <tr>
      <td><a name="20060[1]"><b>20060[1]</b></a></td>
      <td colspan="3">AXCONF_GEOAX_NAME_TAB</td>
      <td>--</td>
      <td>--</td>
    </tr>
    <tr>
      <td>--</td>
      <td colspan="3">Geometry axis name in channel</td><td>BYTE</td>
      <td>POWER ON</td>
    </tr>
    <tr>

```


[illegible]

2.2.5 Saving the online help in the OEM directory

Storage path

Save the HTML file `sinumerik_md_1.html` for the English help under the following path:

[System oem directory]\hlp\eng

You must create a folder for additional languages (e.g. chs, deu, esp, fra, ita ...).

A list of language codes is provided in the Appendix.

2.2.6 Copying easyscreen.ini into the OEM directory

Storage path

Copy the file `easyscreen.ini` from the directory

```
[System SIEMENS directory]cfg
```

to the directory

```
[System oem directory]cfg
```

2.2.7 Registering the COM file in easyscreen.ini

Adaptation in easyscreen.ini

Make the following change in the easyscreen.ini in the OEM directory. You have thus registered the diag.com configuration file.

```
;#####  
;# AREA Diagnosis      #  
;#####  
;<=====>  
;< OEM Softkey on first horizontal Main Menu      >  
;< SOFTKEY position="7"                          >  
;<=====>  
StartFile28 = area := AreaDiagnosis, dialog:= SlDgDialog, menu := DgGlobalHu, startfile := diag.com
```

2.2.8 Testing the project

Testing the dialog call

Switch to the diagnostics operating area. Click the "START" horizontal softkey.

If "Run MyScreens" detects errors when interpreting the configuration files, these errors are stored in the easyscreen_log.txt text file (see Chapter Troubleshooting (log book) (Page 31)).

Testing context sensitive help screens and online help

Test the context sensitive help screens and the online help. If errors occur, then check the paths where files have been saved.

Fundamentals

3.1 Structure of configuration file

Introduction

The description of new user interfaces is stored in configuration files. These files are automatically interpreted and the result displayed on the screen. Configuration files are not stored in the software supplied and must be set up by the user.

A UTF-8-capable editor (e.g. the integrated editor of the user interface or Notepad) is used to create the configuration files and language files. The description can also be explained using comments. A ";" is inserted as comment character before every explanation.

Note

Make sure that the coding is set to UTF-8 in the editor that you are using when saving the configuration and language files.

However, keywords - also in an UTF-8-coded COM file - may only comprise characters that are contained in the ASCII character set. Otherwise the interpretation and therefore the display of the screen forms/dialogs cannot be guaranteed.

Configuration

Each HMI operating area has permanent start softkeys which can be used to access newly generated dialogs.

In the event of a "Load a screen form" (LM) or a "Load softkey menu" (LS) call in a configuration file, a new file name containing the object called can be specified. This makes it possible to structure the configuration, e.g. all functions in one operating level in a separate configuration file.

Structure of the configuration file

A configuration file consists of the following elements:

1. Description of the start softkeys
2. Definition of the dialogs
3. Definition of the variables
4. Description of the methods
5. Definition of the softkey menus

Note**Sequence**

The specified sequence in the configuration file must be maintained.

Example:

```
//S (START)                ; Definition of the start softkey (optional)
....
//END
//M (.....)              ; Definition of the dialog
DEF .....                 ; Definition of the variables
LOAD                      ; Description of the blocks
...
END_LOAD
UNLOAD
...
END_UNLOAD
...
//END
//S (...)                  ; Definition of a softkey menu
//END
```

Storage location for configuration files

The configuration files are located in the *[System user directory]/proj* directory and also correspondingly stored in the *[System add_on directory]* and *[System oem directory]* directories.

Converting texts from other HMI applications

Procedure to convert a text file with code page coding to text-coding UTF-8:

1. Open the text file on a PG/PC in a text editor.
2. When saving, set the UTF-8 coding.

The read-in mechanism via code page code is still supported.

3.2 Structure of the menu tree

Menu tree principle

Several interlinked dialogs create a menu tree. A link exists if you can switch from one dialog to another. You can use the newly defined horizontal/vertical softkeys in this dialog to call the preceding or any other dialog.

A menu tree can be created behind each start softkey:

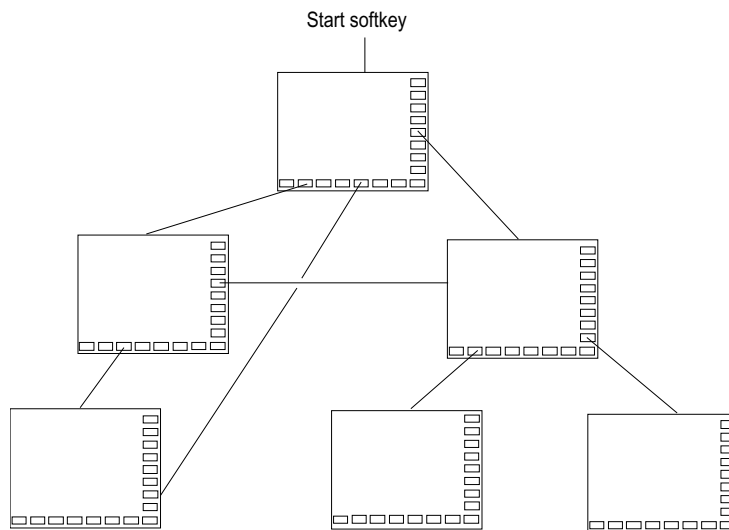


Figure 3-1 Menu tree

Start softkeys

One or more softkeys (start softkeys), which are used to initiate your own operating sequences, are defined in the configuration files specified in easyscreen.ini.

The loading of a dedicated dialog is associated with a softkey definition or another softkey menu. These are then used to perform the subsequent actions.

Pressing the start softkey loads the assigned dialog. This will also activate the softkeys associated with the dialog. Variables will be output to the standard positions unless specific positions have been configured.

Reverting to the standard application

You can exit the newly created dialog and return to the standard operating area.

You can use the <RECALL> key to close new user interfaces if you have not configured this key for any other task.

Note
Calling dialogs in the PLC user program

Dialogs can be selected from the PLC as well as via softkeys: An interface signal is available in DB19.DBB10 for signal exchange between the PLC → HMI.

3.3 Definition and functions for start softkeys

3.3.1 Defining the start softkey

Dialog-independent softkey

Start softkeys are dialog-independent softkeys which are not called from a dialog, but which have been configured **before** the first new dialog. In order to access the start dialog or a start softkey menu, the start softkey must be defined.

Programming

The definition block for a start softkey is structured as follows:

```
//S(Start)           ; Start identifier of start softkey
HS1=(...)            ; Define the start softkey: Horizontal SK 1
PRESS(HS1)           ; Method
    LM...             ; LM or LS function
END_PRESS             ; End of method
//END                ; End identifier of start softkey
```

Permissible positions for start softkeys

The following positions for "Run MyScreens" start softkeys are permissible in the operating areas:

Operating area	Position
Machine	HSK6
Parameter	HSK7
Program	HSK6 and HSK15 Measuring cycles: HSK13 and HSK14
Program manager	HSK2-8 and HSK12-16, if not assigned to drives.

Operating area	Position
Diagnostics	HSK7
Startup	HSK7

Start softkeys are configured in special files. The names of these files are stated in the easyscreen.ini file. They usually have a name which is specific to an operating area (e.g. startup.com for the startup area). The machine operating area represents an exception. There are several mode-specific files in the machine operating area (ma_jog.com, ma_auto.com).

The softkey menu with the start softkeys is called "Start". Existing configurations for start softkeys can still be used. The function whereby start softkeys are merged with the softkeys for the respective HMI application (operating area) in the start softkey menu is not supported.

This means that until the first dialog call is made - in other words, the time at which full functionality becomes available (e.g. execution of PRESS blocks) - menus or softkey menus can only be replaced by others in their entirety.

Template for configuration of the start softkey

A detailed description of all permissible positions for start softkeys and their configuration is located in the easyscreen.ini file in the following directory:

[System SIEMENS directory]\cfg

This file is used as a template for your own configuration of the start softkey.

See also

Lists of start softkeys (Page 199)

3.3.2 Functions for start softkeys

Functions for dialog-independent softkeys

Only certain functions can be initiated with start softkeys.

The following functions are permitted:

- The **LM function** can be used to load another dialog: **LM**("Identifier"[, "File"])
- The **LS function** can be used to display another softkey menu: **LS**("Identifier"[, "File"][, Merge])
- You can use the **"EXIT" function** to exit newly configured user interfaces and return to the standard application.
- You can use the **"EXITLS" function** to exit the current user interface and load a defined softkey menu.

PRESS method

The softkey is defined within the definition block and the "LM" or "LS" function is assigned in the PRESS method.

If the start softkey definition is designated as a comment (semicolon (;) at beginning of line) or the configuration file removed, the start softkey will not function.

```
//S(Start)                ; Start identifier
HS6=("1st screen form")    ; Horizontal SK 6 labeled "1st screen form"
PRESS(HS6)                ; PRESS method for horizontal SK 6
    LM("Screen form1")    ; Load screen form1 function, where screen form 1
                        ; must be defined within the same file.
END_PRESS                ; End of PRESS method
HS7=("2nd screen form")    ; Horizontal SK 7 labeled "2nd screen form"
PRESS(HS7)                ; PRESS method for horizontal SK 7
    LM("Screen form2")    ; Load screen form2 function, where screen form2
                        ; must be defined within the same file.
END_PRESS                ; End of PRESS method
//END                    ; End identifier of entry block
```

Example

```
HS1 = ("new softkey menu")
HS2=("no function")
PRESS(HS1)
    LS("Menu1")           ;Load new softkey bar
END_PRESS
PRESS (HS2)               ; Empty PRESS method
END_PRESS
```

Various configurations of the start softkeys

Various configurations of the start softkeys are merged. In this case, initially the name of the file to be interpreted is read-out of easyscreen.ini. A search is made for files with the .com extension in the following directories:

- *[System user directory]/proj*
- *[System oem directory]/proj*
- *[System add_on directory]/proj*
- *[System Siemens directory]/proj*

The configurations included for the start softkeys are now merged to form a configuration, i.e. the individual softkeys are compared. If there are two or more configurations for a softkey, the higher order configuration is always transferred into the merge version.

Softkey menus or dialogs that are possibly included are ignored. If a softkey has a command without file information, e.g. LM ("test"), as the required softkey menu or dialog is contained in the same file, then the corresponding file name is supplemented in the internal merge version so that in this case, no changes are required. The merge configuration contained is then subsequently displayed.

3.4 Troubleshooting (log book)

Overview

If "Run MyScreens" detects errors when interpreting the configuration files, these errors will be written to the easyscreen_log.txt text file. Only errors of the currently selected dialog are entered. Error entries from previously selected dialogs are deleted.

The file indicates:

- The action during which an error occurred
- The line and column number of the first faulty character
- The entire faulty line of the configuration file
- The entries made by the DEBUG function

Note

Each entry has a prefix with the current time stamp in square brackets. This can be helpful, for example, when analyzing time-critical configurations.

Saving easyscreen-log.txt

The file easyscreen_log.txt is saved in the following directory:

[System user directory]/log

Syntax

The system does not start to interpret the syntax until the start softkey has been defined and a dialog with start and end identifiers as well as a definition line has been configured.

```
//S(Start)
HS6= ("1st screen")
PRESS(HS6)
    LM("Maske1")
END_PRESS
//END

//M(Maske1)
```

```

DEF Var1=(R)
DEF VAR2 = (R)
LOAD
VAR1 = VAR2 + 1           ; Error message in log book, as VAR2 has no value
...
//END

; The right entry would be,
for example:
//M(Maske1)
DEF Var1=(R)
DEF VAR2 = (R)

LOAD
VAR2 = 7
VAR1 = VAR2 + 1 ;
...

```

3.5 Notes on the easyscreen.ini

As of SINUMERIK Operate V4.7, the easyscreen.ini has been extended by the entries described in this section. The easyscreen.ini can be found in the *[System Siemens directory]* cfg directory.

Help display start position

Entry in easyscreen.ini:

```

[GENERAL]
HlpPicFixPos=true

```

Note:

The start position of help displays is positioned at the configured pixel position independent of the resolution (default=true).

Elongation behavior, default line height and line spacing

Entry in easyscreen.ini:

```

[GENERAL]
SymmetricalAspectRatio=false
DefaultLineHeight=18

```

```
DefaultLineSpacing=3
```

Notes:

- The "SymmetricalAspectRatio" entry determines whether the same aspect ratio is used in the X and Y direction when adapting a configuration to a specific screen resolution.
 - "false" (default): Fields and graphics are shortened in widescreen resolution in the Y direction (asymmetrical elongation in relation to 640x480). For example, a square configured in 640x480 is shortened vertically for a widescreen panel and becomes a rectangle.
 - "true": The same aspect ratio is used in the X and Y direction and fields and graphics retain their originally configured proportions in relation to 640x480. For example, a square configured in 640x480 remains a square in a widescreen panel.
- The entries "DefaultLineHeight" and "DefaultLineSpacing" can be used to specify the default line height (default: 18 pixels) and the default line spacing (default: 3 pixels) in relation to 640x480. They only take effect when no Y position or height is specified in the configuration for the position of the short text or the input/output field.

Resolution-dependent screen position: Form panels**Entry in easyscreen.ini:**

```
[640x480]
MyPanel = x:=0, y:=220, width:=340, height:=174
[800x480]
MyPanel = x:=0, y:=220, width:=420, height:=174
...
```

Notes:

3.6 Notes for personnel changing over to "Run MyScreens"

The screen position can be defined as follows:

- The screen position can be specified in "Pixels in relation to 640x480".

Example:

```
//M(MyMask/"MyCaption"/"\myhelp.png"/0,219,335,174)
```

- The screen position can be coupled to the resolution-dependent position of a form panel from an Operate standard screen layout, e.g. "Form panel4" ("auxiliary functions") from the screen layout "slmastandardscreenlayout.SIMaStandardScreenLayout" of the "Machine" area

```
//M(MyMask/"MyCaption"/"\myhelp.png"/"slmastandardscreenlayout.SIMaStandardScreenLayout.FormPanel4")
```

It is therefore easy to superimpose standard forms from Operate or position Easyscreen screens precisely at their position.

Example:

```
//M(Mask/"Mask"/"slstandardscreenlayout.SIStandardScreenLayout.LowerForm")
```

Any other screen layout can also be used.

- The screen position can be coupled to a self-defined, resolution-dependent definition of form panels in the easyscreen.ini file.

Example:

```
//M(MyMask/"MyCaption"/"\myhelp.png"/"MyPanel")
```

3.6 Notes for personnel changing over to "Run MyScreens"

Note

When using HMI Operate in the NCU, note that all file names are saved in lower case letters on the CF card (com, png, txt).

Note

Make sure that the coding is set to UTF-8 in the editor that you are using when saving the configuration and language files.

Image files

Image files must be saved in the PNG format (xxx.png).

The data must be saved, e.g. for OEM modifications, under

[System oem directory]/ico/[Resolution]

For further information, see also Chapter Using display images/graphics (Page 50).

Adapting the configuration file

Check your configuration files regarding the following points:

- Compare the start softkeys with the currently permissible softkeys (see the list of start softkeys in the Appendix) and if required, adapt them.
- Rename the integrated image files corresponding to the "Image files" point above.

The data is stored, e.g. for OEM modifications, in the following directory:

[System oem directory]proj

[System user directory]proj

[System add_on directory]proj

Adapting the help files

All help files must be saved in UTF-8. Check your existing files, and resave them accordingly with a suitable editor.

The HTML files are saved in the following directory, e.g. for English:

[System oem directory]hlp/eng

[System user directory]hlp/eng

[System add_on directory]hlp/eng

You must create directories for additional languages corresponding to the language identifiers (see the Appendix).

Checking the "Run MyScreens" license

Check whether the number of dialogs that you have entered exceeds the basic scope of a maximum of five dialogs.

To expand the number of dialogs, you require one of the following software options:

- SINUMERIK 828D/840D sl, SINUMERIK Integrate Run MyScreens (6FC5800-0AP64-0YB0)
- SINUMERIK 840D sl, SINUMERIK Integrate Run MyScreens + Run MyHMI (6FC5800-0AP65-0YB0)
- SINUMERIK 840D sl, SINUMERIK Integrate Run MyHMI / 3GL (6FC5800-0AP60-0YB0)
- SINUMERIK 840D sl, SINUMERIK Integrate Run MyHMI / WinCC (6FC5800-0AP61-0YB0)

3.7 Extended configuration syntax

As of SINUMERIK Operate V4.7, a simplified syntax is available for the definition of screens, variables, softkeys and for table columns. Reading and servicing are improved with this alternative syntax. The properties and attributes can be specified in any order, empty entries are omitted. Instead of with round brackets "(" and ")" as in the previous syntax, the listing of the properties and attributes is now enclosed in curly brackets "{" and "}".

The properties and attributes are specified as follows:

```
{<Name> = <Value>, <Name> = <Value>, ...}
```

The previous syntax remains compatible.

Extended syntax for the definition of screens

```
//M {<Screen name> [,HD=<Header>] [,HLP=<Graphic>] [,X=<X position>] [,Y=<Y position>]
[,W=<Width>] [,H=<Height>] [,VAR=<>System or user variable] [,HLP_X=<X position help
display>] [,HLP_Y=<Y position help display>] [,CM=<Column alignment>] [,CB=Behavior
when opening the dialog] [,XG=<Interpret help display as X3d graphic>] [,PANEL=<Name of
the linked form panel>][,MC=<Screen background color>][,AL=<Screen header alignment>]}
```

Example:

```
//M{VariantTest, HD="My Mask"}
```

Extended syntax for the definition of variables

```
DEF <Variable name> = {[TYP=<Type>] [,MIN=<Minimum value>] [,MAX=<Maximum value>]
[,TGL=<Toggle values>] [,VAL=<Default>] [,LT=<Long text>] [,ST=<Short text>]
[,GT=<Graphic text>] [,UT=<Unit text>] [,TT=<Tooltip text>] [,TG=<Toggle option>]
[,WR=<Input mode>] [,AC=<Access level>] [,AL=<Text alignment>] [,FS=<Font size>]
[,LI=<Limit value processing>] [,UR=<Update rate>] [,CB=<Behavior when opening the
dialog>] [,HLP=<Help display>] [,VAR=<System or user variable>] >] [,TXT_X=<X position
short text>] [,TXT_Y=<Y position short text>] [,TXT_W=<Width short text>] [,TXT_H=<Height
short text>] [,X=<X position input/output field>] [,Y=<Y position input/output field>] [,W=<Width
input/output field>] [,H=<Height input/output field>] [,BC=<Background color input/output
field>] [,FC=<Foreground color input/output field>] [,BC_ST=<Background color short text>]
[,FC_ST=<Foreground color short text>] [,BC_GT=<Background color graphic text>]
[,FC_GT=<Foreground color graphic text>] [,BC_UT=<Background color unit text>]
[,FC_UT=<Foreground color unit text>] [,SC1=<Signal color 1 for progress bar>]
[,SC2=<Signal color 2 for progress bar>] [,SVAL1=<Threshold 1 for progress bar>]
[,SVAL2=<Threshold 2 for progress bar>] [,DT=<Display type>] [,DO=<Display alignment>]
[,OHLP=<Online help>][,LINK_TGL=<Name of the linked toggle variable>]}
```

Examples:

```
DEF MyVar5={TYP="R2", ST="MyVar5", VAL=123.4567, OHLP="myhelp.html", MIN=100.1,
MAX=200.9}
DEF MyVar2={TYP="I", TGL="*1,2,3", VAL=1}
DEF MyVar3={TYP="R2", TGL="*0=""Off"", 1=$80000", VAL=1}
DEF MyVar4={TYP="R2", TGL="*MyArray", VAL=1}
DEF MyVar1={TYP="R2", TGL="%grid99", X = 0, W=300, H=200}
```

```
DEF MyVar6={TYP="R2", TGL="+$80000", VAR="$R[10]", ST="Text offset"}
```

Extended syntax for the definition of softkeys

SK = {[ST=<Label>] [,AC=<Access level>] [,SE=<Status>]}

Examples:

```
HS1={ST=""MySk"", AC=6, SE=1}
HS3={ST="SOFTKEY_CANCEL"}
HS5={ST="[$81251, ""\sk_ok.png""]"}
HS8={ST="["Test"", ""\sk_ok.png""]"}
```

Extended syntax for the definition of table columns

{[TYP=<Type>] [,MIN=<Minimum value>] [,MAX=<Maximum value>] [,LT=<Long text>]
 [,ST=<Short text>] [,WR=<Input mode>] [,AC=<Access level>] [,AL=<Text alignment>]
 [,FS=] [,LI=<Limit value processing>] [,UR=<Update rate>] [,HLP=<Help display>]
 [,VAR=<System or user variable>] >] [,W=<Column width>] [,OF1=<Offset1>]
 [,OF2=<Offset2>] [,OF3=<Offset3>]}

Example:

```
DEF MyGridVar={TYP="R", TGL="%MyGrid1", X=10, W=550, H=100}

//G(MyGrid1/0/5)

  {TYP="I", ST="Index", WR=1, VAR="1", W=80, OF1=1}

  {TYP="S", LT="LongText2", ST="Text", WR=1, VAR="$80000", AL=2, W=330, OF1=1}

  {TYP="R3", LT="LongText1", ST="R9,R11,R13,R15", WR=2, VAR="$R[1]", W=110, OF1=2}

//END
```

Dialogs

4.1 Structure and elements of a dialog

4.1.1 Defining a dialog

Definition

A dialog is part of a user interface consisting of a display line, dialog elements and/or graphics, an output line for messages and 8 horizontal and 8 vertical softkeys.

Dialog elements are:

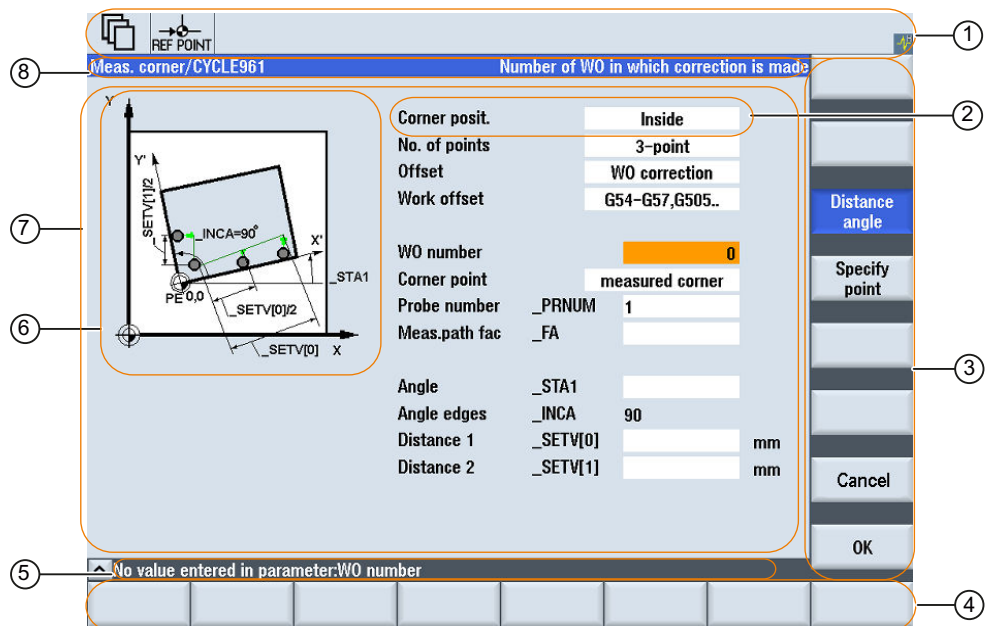
- Variables
 - Limits/toggle field
 - Default setting of variables
- Help display
- Texts
- Attributes
- System or user variable
- Position of short text
- Position of input/output field
- Colors

Dialog properties:

- Header
- Graphic
- Dimension
- System or user variable

4.1 Structure and elements of a dialog

- Graphic position
- Attributes



- ① Machine status display ("header")
- ② Dialog element
- ③ 8 vertical softkeys
- ④ 8 horizontal softkeys
- ⑤ Output of diagnostic messages
- ⑥ Graphic
- ⑦ Dialog
- ⑧ Header line of the dialog with header and long text

Figure 4-1 Structure of the dialog

Overview

The definition of a dialog (definition block) is basically structured as follows:

Definition block	Comment	Chapter reference
//M...	;Dialog start identifier	
DEF Var1=... ...	;Variables	See chapter "Variables"
HS1=(...) ...	;Softkeys	See chapter "Softkey menus"
PRESS (HS1) LM... END_PRESS	;Method start identifier ;Actions ;Method end identifier	See chapter "Methods"
//END	;Dialog end identifier	

Within the dialog definition block, various variables that appear as dialog elements in the dialog, as well as horizontal and vertical softkeys, are defined first. Different types of actions are then configured in methods.

4.1.2 Defining dialog properties

Description

The properties of the dialog are defined in the start identifier line of the dialog.

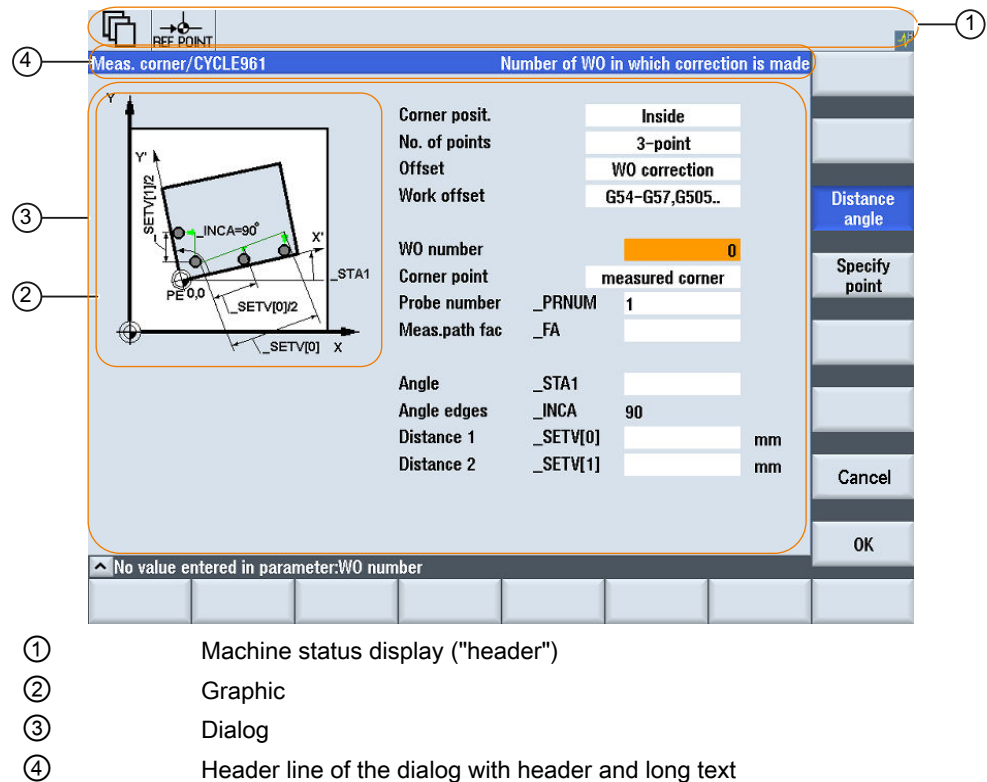


Figure 4-2 Dialog properties

Programming

Syntax:	//M(Identifier/[Header]/[Graphic]/[Dimension]/[System or user variable]/[Graphic position]/[Attributes]) See also Section Extended configuration syntax (Page 36).
Description:	Defines a dialog

4.1 Structure and elements of a dialog

Parameter:	Identifier		Name of the dialog.
	Header		Dialog header as text or call for text (e.g. \$85011) from a language-specific text file.
	Graphic		Graphics file with path in double quotation marks.
	Dimension		Position and size of the dialog in pixels (distance from left-hand side, distance from right-hand side, width, height), in relation to the upper left-hand corner of the screen. The entries are separated by a comma.
	System or user variable		System or user variable to which the current cursor position is assigned. The NC or PLC can be provided with the cursor position via the system or user variable. The first variable has the index 1. The order corresponds to the configuration order of the variables.
	Graphic position		Position of the graphic in pixels (distance from left-hand side, distance from right-hand side), in relation to the upper left-hand corner of the dialog. The entries are separated by a comma.
	Attributes		The specifications of the attributes are separated by a comma. Possible attributes are:
	CM	CM	Column mode: Column alignment
		CM0	Default setting: The column distribution is carried out separately for each line.
		CM1	The column distribution of the line with the most columns applies to all lines.
	CB		CHANGE block: Response when dialog is opened: cb attributes specified for a variable in a variables definition take priority over the default setting in the dialog definition. See also Behavior when opening the dialog (attribute CB) (Page 204).
		CB0	Default setting: All CHANGE blocks associated with the dialog are processed when it is opened.
		CB1	CHANGE blocks are then only processed if the relevant value changes.
	XG		Integration of an X3D animation as help display (only in the machining step programming). Example: <code>//M(Meas/ \$85605/"myx3dhelpfile.hmi",,Z_Animation,,G17"///30,10/ XG1)</code>
		XG0	Default setting = 0
		XG1	The screen attribute XG must already be set to 1 in the screen definition, a change during runtime is not possible. Note: The specification in the screen property HLP must also be set accordingly.
	AL		The alignment of the screen header can be influenced with the screen attribute AL. Example: Central setting of the "Set password" window header: <code>//M(MY_PWD_SET/"Set password"//"EasyPwdModalLayout"// AL2)</code>
		AL0	Left-justified, default
		AL1	Right-justified
		AL2	Centered
	MC		Screen background color (screen color)

			Example: Set screen background color blue (= 6) <pre>//M(MyMask/"MyMask"/////6)</pre> or <pre>PRESS(HS1) MC=6 END_PRESS</pre>

Accessing the dialog properties

Read and write access to the following dialog properties is permitted within methods (e.g. PRESS block).

- HD = Header
- HLP = Help display
- VAR = System or user variable
- MC = Screen background color
- CM = Column alignment (read-only)
- CB = Behavior when opening (read-only)
- XG = Integration of X3d (read-only)
- AL = Screen header alignment (read-only)

Example

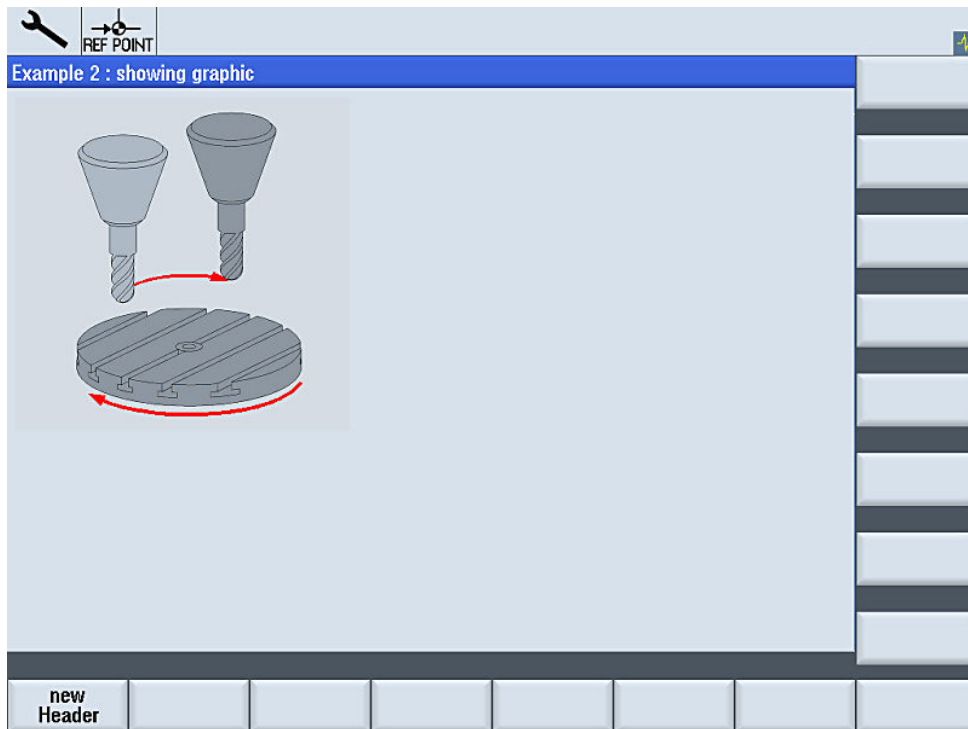


Figure 4-3 "Example 2: showing graphic"

```
//S(Start)
HS7=("Example", sel, ac7)

PRESS (HS7)
  LM("Mask2")
END_PRESS

//END
//M(Mask2/"Example 2 : showing graphic"/"example.png")
HS1("new\nHeader")
HS2("")
HS3("")
HS4("")
HS5("")
HS6("")
HS7("")
HS8("")
VS1("")
VS2("")
VS3("")
```

```
VS4= ("")  
VS5= ("")  
VS6= ("")  
VS7= ("")  
VS8= ("")  
  
PRESS (HS1)  
    Hd= "new Header"  
END_PRESS  
...  
//END
```

See also

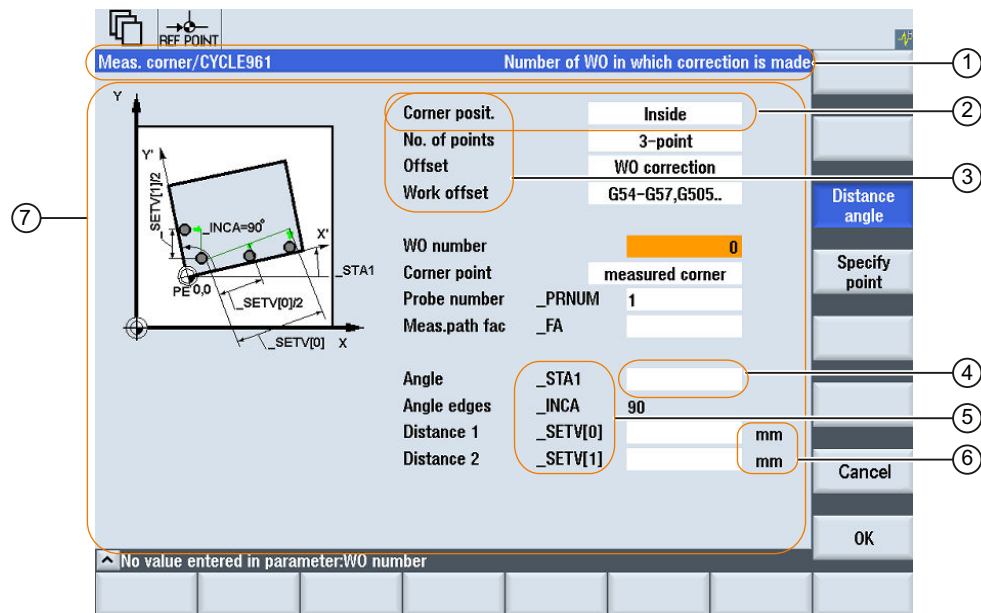
Programming example for the "Custom" area (Page 188)

4.1.3 Defining dialog elements**Dialog element**

The term "dialog element" refers to the visible part of a variable, i.e. short text, graphics text, input/output field, unit text and tooltip. Dialog elements fill lines in the main body of the dialog. One or more dialog elements can be defined for each line.

Variable properties

All variables are valid in the active dialog only. Properties are assigned to a variable when it is defined. The values of dialog properties can be accessed within methods (e.g. a PRESS method).



- ① Header line of the dialog with header and long text
- ② Dialog element
- ③ Short text
- ④ Input/output field
- ⑤ Graphic text
- ⑥ Text for units
- ⑦ Main body of the dialog

Figure 4-4 Elements of a dialog

Programming - Overview

The single parameters to be separated by commas are enclosed in round brackets:

DEF Identifier =	Identifier = Name of variable
	Variable type
	/[Limit values or toggle field]
	/[Default]
	/[Texts (Long text, Short text Image, Graphic text, Units text)]
	/[Attributes]
	/[Help display]
	/[System or user variable]
	/[Position of short text]
	/[Position of I/O field(Left, Top, Width, Height)]

	/[Colors]
	/[online help] (Page 58)

See also

Variable parameters (Page 69)

4.1.4 Defining dialogs with multiple columns**Overview**

Multiple variables can also be represented in a dialog on one line. In this case, the variables are all defined in the configuration file on a single definition line.

```
DEF VAR11 = (S///"Var11"), VAR12 = (I///"Var12")
```

To make individual variables in the configuration file more legible, the definition lines can be wrapped after every variables definition and following comma.

The key word "DEF" always indicates the beginning of a new line:

```
DEF Tnr1=(I//1/"", "T ", ""/wr1///, 10/20,, 50),
TOP1=(I///, "Type=" /WR2//"$TC_DP1[1,1]"/80,, 30/120,, 50),
TOP2=(R3///, "L1=" /WR2//"$TC_DP3[1,1]"/170,, 30/210,, 70),
TOP3=(R3///, "L2=" /WR2//"$TC_DP4[1,1]"/280,, 30/320,, 70),
TOP4=(R3///, "L3=" /WR2//"$TC_DP5[1,1]"/390,, 30/420,, 70)
DEF Tnr2=(I//2/"", "T ", ""/wr1///, 10/20,, 50),
TOP21=(I///, "Typ=" /WR2//"$TC_DP1[2,1]"/80,, 30/120,, 50),
TOP22=(R3///, "L1=" /WR2//"$TC_DP3[2,1]"/170,, 30/210,, 70),
TOP23=(R3///, "L2=" /WR2//"$TC_DP4[2,1]"/280,, 30/320,, 70),
TOP24=(R3///, "L3=" /WR2//"$TC_DP5[2,1]"/390,, 30/420,, 70)
```

Note

When configuring multi-column dialogs, please observe that a large number of columns might slow down the system!

4.1.5 Password dialogs

Using the standard password dialogs

You can integrate the following predefined password dialogs in the screen configuration:

- Set password

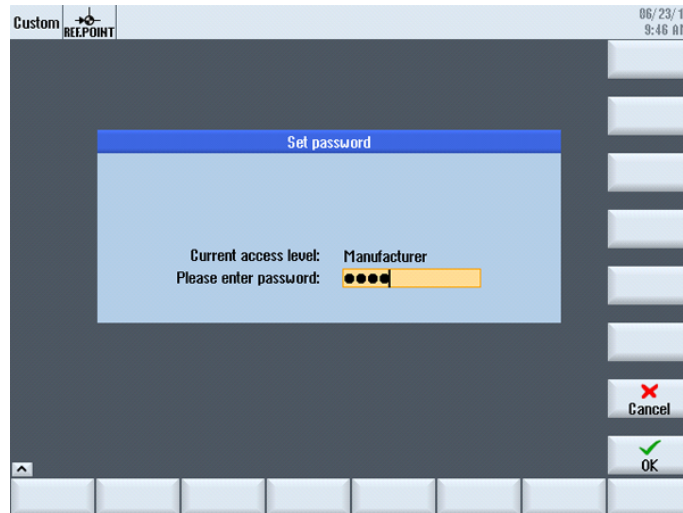


Figure 4-5 Set password dialog

- Change password

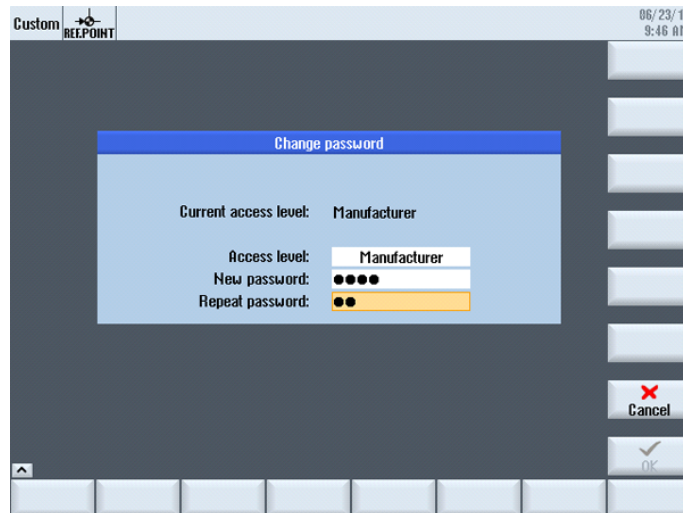


Figure 4-6 Change password dialog

- Delete password

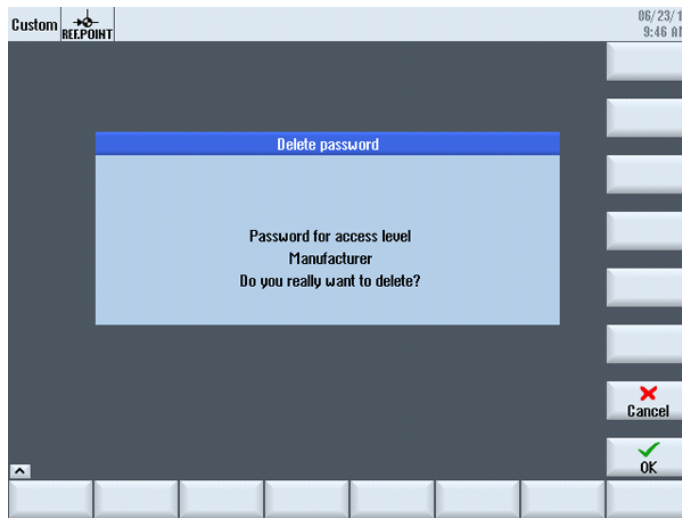


Figure 4-7 Delete password dialog

Note

The password functionality is already set with these dialogs. The dialogs do not correspond to the dialogs of SINUMERIK Operate.

The dialogs can be called either via the Load softkey (LS) function or the Load mask (LM) function:

1. Call via the Load softkey (LS) function:

```
LS ("Passwd", "slespasswd.com", 1)
```

Extension of the vertical softkeys:

- Softkey 1: Set password
- Softkey 2: Change password
- Softkey 3: Delete password

2. Alternatively, the three screens can be opened directly by calling the Load mask (LM) function:

- Set password: `LM ("PWD_SET", "slespasswd.com", 1)`
- Change password: `LM ("PWD_CHG", "slespasswd.com", 1)`
- Delete password: `LM ("PWD_CLEAR", "slespasswd.com", 1)`

4.1.6 Using display images/graphics

Use of graphics

A distinction is made between:

- Display images / graphics in the graphic area.
- Help displays illustrating, for example, individual variables, which are superimposed in the graphic area.
- More Help displays can be configured instead of short text or an input/output field, which you position where you like.

Storage locations

The image matching the resolution of the connected monitor is searched for in the associated resolution directories in the following sequence:

```
[System user directory]/ico/ico<Resolution>  
[System oem directory]/ico/ico<Resolution>  
[System add_on directory]/ico/ico<Resolution>
```

If the image is not displayed or not found, copy it into one of the following directories for a resolution of 640 x 480 pixels:

```
[System user directory]/ico/ico640  
[System oem directory]/ico/ico640  
[System add_on directory]/ico/ico640
```

Note

With the different panel resolutions, the images are positioned proportionally.

4.2 Defining softkey menus

Definition

The term softkey menu is used to refer to all the horizontal and vertical softkeys displayed on a screen form. In addition to the existing softkey menus, it is possible to define other menus, which partially or completely overwrite the existing menus.

The names of the softkeys are predefined. Not all softkeys need to be assigned.

HSx x 1 - 8, Horizontal softkeys 1 to 8

VSy y 1 - 8, Vertical softkeys 1 to 8

The definition of a softkey menu (softkey menu definition block) is basically structured as follows:

Definition block	Comment	Section reference
//S...	;Start identifier of softkey menu	
HSx=...	;Define softkeys	
PRESS (HSx) LM... END_PRESS	;Method start identifier ;Actions ;Method end identifier	See Section "Methods"
//END	;End identifier of softkey menu	

Description

Properties are assigned to softkeys during definition of the softkey menu.

Programming

Syntax:	<i>//S(Identifier)</i>	;Start identifier of softkey menu
	...	
	//END	;End identifier of softkey menu
	See also Section Extended configuration syntax (Page 36).	
Description:	Defines the softkey menu	
Parameter:	Identifier	Name of the softkey menu
	Text or image file name	
Syntax:	SK = (<i>Text</i> [, <i>Access level</i>][, <i>Status</i>][, <i>Alignment of the softkey image</i>][, <i>Text alignment in relation to the softkey image</i>])	
Description:	Define softkey	
Parameter:	SK	Softkey, e.g. HS1 to HS8, VS1 to VS8

	Text	Enter text
	Display file name	"\\my_pic.png" or via separate text file \$85199, e.g. with the following text in the (language-specific) text file: 85100 0 0 "\\my_pic.png". The image size which can be displayed on a softkey depends on the OP used:
		OP 08: 640 x 480 mm → 25 x 25 pixels OP 010: 640 x 480 mm → 25 x 25 pixels OP 012: 800 X 600 mm → 30 x 30 pixels OP 015: 1024 X 768 mm → 40 x 40 pixels OP 019: 1280 x 1024 mm → 72 x 72 pixels
	Access level	ac0 to ac7 (ac7: default)
	Status	se1: visible (default) se2: disabled (gray text) se3: displayed (last softkey used)
	Alignment of the softkey image	PA (PictureAlignment) Valid values: 0: Left 1: Right 2: Centered 3: Top (default) 4: Bottom
	Text alignment in relation to the softkey image	TP (TextAlignedToPicture) Valid values: 0: Text is not aligned to the image 1: Text is aligned to the image (default)

Note

Enter %n in the softkey text to create a line break.

A maximum of 2 lines with 9 characters each are available.

Assigning access level

Operators can only access information on this access level and lower access levels (see also List of access levels (Page 202)).

Example

```
//S(Menu1)                                ; Start identifier of softkey menu
HS1=("NEW", ac6, se2)                       ; Define softkey HS1, assign the label "NEW", protection level 6, and the status "disabled"
```

```

HS2=("\\image1.png")                ; Assign a softkey to the graphic
HS3=("Exit")
HS4=(["Confirm","\\sk_ok.png"],PA0,TP1) ; Softkey with text and graphic, Text="Confirm", Im-
                                         age="sk_ok.png", Alignment of the softkey image: Left,
                                         text is aligned to the image

VS1=("sub screen form")
VS2=($85011, ac7, se2)              ; Define softkey VS2, assign the text from the lan-
                                         guage file, protection level 1, and the status "disa-
                                         bled"
VS3=("Cancel", ac1, se3)            ; Define softkey VS3, assign the label "Cancel", pro-
                                         tection level 1 and the status "highlighted"
VS4=("OK", ac6, se1)               ; Define softkey VS4, assign the label "OK", protec-
                                         tion level 6 and the status "visible"
VS5=(SOFTKEY_CANCEL,,se1)          ; Define cancel standard softkey VS5 and assign the
                                         status "visible"
VS6=(SOFTKEY_OK,,se1)             ; Define OK standard softkey VS6 and assign the status
                                         "visible"
VS7=(["\\image1.png","OEM text"],,se1) ; Define softkey VS7, assign an image, assign the la-
                                         bel "OEM Text" and the status "visible"
VS8=(["\\image1.png",$83533],,se1) ; Define softkey VS8, assign an image, assign text
                                         from language file and the status "visible"

PRESS(HS1)                        ; Method start identifier
    HS1.st="Calculate"            ; Assign a label text to the softkey
...
END_PRESS                        ; Method end identifier

PRESS(RECALL)                    ; Method start identifier
    LM("Screen form21")          ; Load dialog
END_PRESS                        ; Method end identifier
//END                            ; End identifier of softkey menu

```

4.2.1 Changing softkey properties during runtime

Description

The softkey properties Text, Access Level and Status can be changed in the methods during runtime.

Programming

Syntax:	SK.st = "Text" SK.ac = Access level SK.se = Status SK.pa = "Alignment of the softkey image" SK.tp = "Text alignment in relation to the softkey image"		;Softkey with label ;Softkey with protection level ; Softkey with status ;Softkey with image ;Softkey with image and label
Description:	Assign properties		
Parameter:	Text		Label text in inverted commas
	Access level		Range of values: 0 ... 7
	Status	1: Visible and operator-controllable 2: Disabled (gray text) 3: Highlighted (last softkey used)	
	Alignment of the softkey image	0: Left 1: Right 2: Centered 3: Top (default) 4: Bottom	
	Text alignment in relation to the softkey image	0: Text is not aligned to the image 1: Text is aligned to the image (default)	

Example

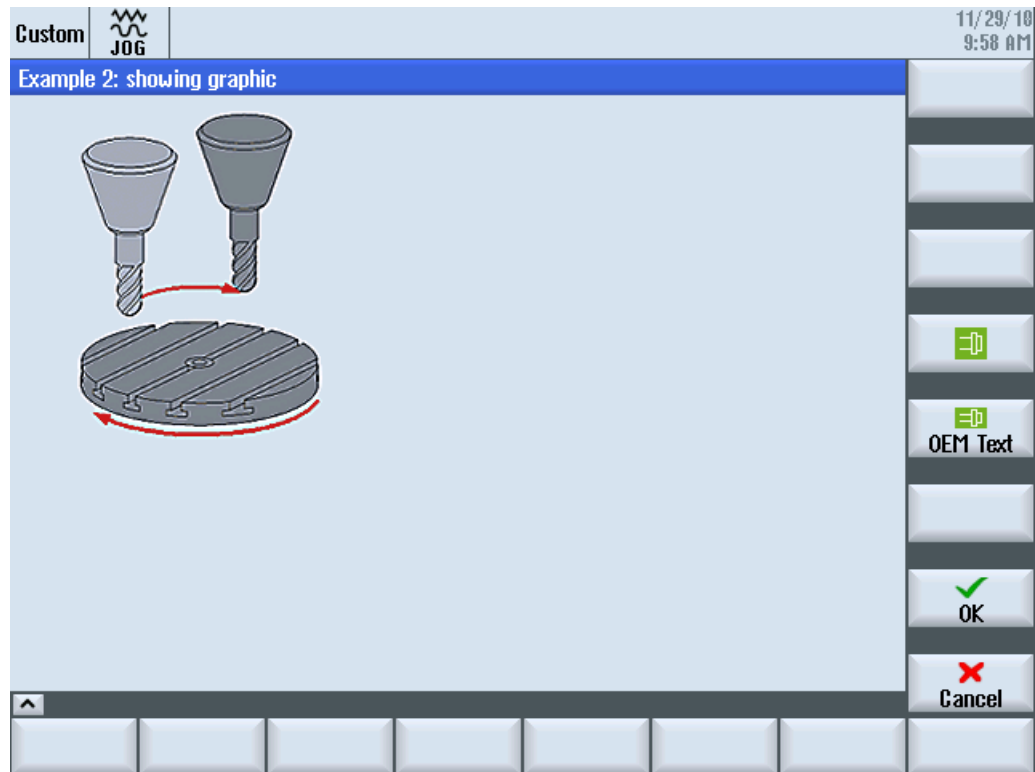


Figure 4-8 Example 3: Graphics and softkeys

```
//S(Start)
HS7=("Example", ac7, sel)

PRESS(HS7)
  LM("Maske3")
END_PRESS

//END

//M(Maske3/"Example 2: showing graphic"/"example.png")
HS1=""
HS2=""
HS3=""
HS4=""
HS5=""
HS6=""
HS7=""
HS8=""
```

```
VS1= ("")
VS2= ("")
VS3= ("")
VS4= ("\\sp_ok.png", , SE1)
VS5= (["\\sp_ok_small.png", "OEM Text"], , SE1)
VS6= ("")
VS7= (SOFTKEY_OK, , SE1)
VS8= (SOFTKEY_CANCEL, , SE1)
PRESS (VS4)
    EXIT
END_PRESS
PRESS (VS5)
    EXIT
END_PRESS
PRESS (VS7)
    EXIT
END_PRESS
PRESS (VS8)
    EXIT
END_PRESS
//END
```

4.2.2 Language-dependent text

Overview

Language-dependent texts are used for:

- Softkey labels
- Headings
- Help texts
- Any other texts

The language-dependent texts for dialogs are stored in text files.

The text files are stored in the following directories:

- *[System user directory]*\\lng
- *[System oem directory]*\\lng
- *[System add_on directory]*\\lng

Note

The text files must be saved in the same way as the project files.

For example:

[System user directory]\lng\[Text file]

[System user directory]\proj\[Configuration file]

alsc.txt Language-dependent texts for the Siemens standard cycles

almc.txt Language-dependent texts for the Siemens measuring cycles

The text files used during program runtime are specified in the easyscreen.ini file:

```
[LANGUAGEFILES]
LngFile03 = user.txt            ;->user<_xxx>.txt (e.g. user_eng.txt)
```

In this instance, the user.txt file has been chosen as an example of a text file. The name can always be freely selected. Depending on the language of the texts within the file, the relevant language code must be added using the following syntax. An underscore followed by the relevant language identifier is added after the name e.g. user_eng.txt.

Note

You must not use LngFile01 and LngFile02 for user texts because they are already assigned in the standard easyscreen.ini.

See also

List of language codes used in file names (Page 203)

Format of text files

The text files must be saved in UTF-8 format.

Note

Make sure that the coding is set to UTF-8 in the editor that you are using when saving the configuration and language files.

Format of a text entry

Syntax:	8xxxx 0 0 "Text"
Description:	Assignment between text number and text in the file

4.3 Configuring the online help

Parameter:	xxxx	85,000 to 89,999	Text identification number range reserved for users. You must assign unique numbers.
	"Text"		Text that appears in the dialog
	%n		Control characters in the text for creating a line break

Parameters 2 and 3 are separated by blanks and act as control characters for alarm text output. To ensure that the text format is identical to that of the alarm texts, you must set zero.

Examples of language-dependent texts:

85000 0 0 "Retraction plane"

85001 0 0 "Drilling depth"

85002 0 0 "Pitch"

85003 0 0 "Pocket radius"

4.3 Configuring the online help

Online help

An online help for the configured dialogs and elements can be created in the HTML format.

The syntax and procedure for the online help is essentially the same as for SINUMERIK Operate.

If an online help is configured for an input field, then the standard screens of the online help are used to display the user-specific online help:

```
DEF RFP=(R//1/, "RFP", "RFP"////////"sinumerik_md_1.html", "9006")
```

Note

As a result of LINUX, the HTML file must be written in lower case letters!

The HTML file is saved in the following directory, e.g. for English.

[System user directory]/hlp/eng

[System oem directory]/hlp/eng

[System add_on directory]/hlp/eng

You must create directories for additional languages corresponding to the language identifiers (see the Appendix).

References

Commissioning Manual "Base software and operating software" (IM9), Chapter "OEM-specific online help"

Variables

5.1 Defining variables

Variable value

The major property of a variable is its value.

The value of variables can be assigned by means of:

- Default settings when defining variables
- Assignment to a system or user variable
- A method

Programming

Syntax:	Identifier. val = Variable value Identifier = Variable value	
Description:	Variable value val (value)	
Parameter:	Identifier:	Name of the variable
	Variable value:	Value of the variable
Example:	<pre>VAR3 = VAR4 + SIN (VAR5) VAR3.VAL = VAR4 + SIN (VAR5)</pre>	

Variable status

The "Variable status" property scans a variable for valid content during runtime. This property can be read and written with the value FALSE = 0.

Programming

Syntax:	Identifier. vld	
Description:	Variable status vld (validation)	
Parameter:	Identifier:	Name of the variable
	FALSE = invalid value TRUE = valid value	The result of the scan can be:
Example:	<pre>IF VAR1.VLD == FALSE VAR1 = 84 ENDIF</pre>	

5.2 Application examples

Help variables

Help variables are internal arithmetic variables. Arithmetic variables are defined like other variables, but have no other properties apart from variable value and status, i.e. Help variables are not visible in the dialog. Help variables are of the VARIANT type.

Programming

Syntax:	DEF <i>Identifier</i>	
Description:	Internal arithmetic variables of the VARIANT type	
Parameter:	Identifier:	Name of the Help variable
Example:	DEF OTTO ;Definition of a Help variable	

Syntax:	Identifier.val = <i>Help variable value</i>	
	Identifier = <i>Help variable value</i>	
Description:	A value is assigned to a Help variable in a method.	
Parameter:	Identifier:	Name of the Help variable
	Help variable value:	Content of the Help variables

Example:

```

LOAD
    OTTO = "Test"                ; The value "Test" is assigned to help variable Otto.
END_LOAD
LOAD
    OTTO = REG[9].VAL           ; The value of the register is assigned to help variable
                                Otto
END_LOAD

```

Calculation with variables

Variables are calculated every time you exit an IO field (by pressing the ENTER or TOGGLE key). The calculation is configured in a CHANGE method that is processed every time the value changes.

You can scan the variable status to ascertain the validity of the value of the variable, e.g.

```

IF VAR1.VLD == FALSE
    VAR1 = 84
ENDIF

```


5.3 Example 1: Assigning the variable type, texts, help display, colors, tooltips

Addressing system variables indirectly

A system variable can also be addressed indirectly, i.e. as a function of another variable:

```
PRESS (HS1)
  ACHSE=ACHSE+1
  WEG.VAR="$AA_DTBW["<<ACHSE<<"]"      ;Address axis address via variable
END_PRESS
```

Changing softkey labels

Example:

```
HS3.st = "New text"      ;Change softkey label
```

5.3 Example 1: Assigning the variable type, texts, help display, colors, tooltips

Example 1a

The following example defines a variable for which the properties variable type, texts, help display and colors are set.

DEF Var1 = (R///,"Actual value",,"mm"//"/"Var1.png"////8,2)			
	Variable type:		REAL
	Texts:		
		Short text:	Actual value
		Unit text:	mm
	Help screen:		Var1.png
	Colors:		
		Foreground color:	8 (brown)
		Background color:	2 (orange)

Example 1b

The following example defines a variable for which the properties variable type, default setting, texts, tooltip, input mode and position of short text are set.

DEF Var2 = (l/5/"","Value","", "", " Tooltip text"/wr2///20,250,50)		
	Variable type:	INTEGER
	Default setting:	5
	Texts:	
	Short text:	Value (possible language text ID)
	Tooltip:	ToolTipText
	Attributes:	
	Input mode:	Reading and writing
	Position of short text:	
	Distance from left:	20
	Distance from top:	250
	Width:	50

See also

Variable parameters (Page 69)

5.4 Example 2: Assigning the Variable Type, Limits, Attributes, Short Text Position properties

Example 2

The following example defines a variable for which the properties variable type, limits, input mode, alignment and position are set.

DEF Var2 = (l/0,10///wr1,al1///,,300)		
	Variable type:	INTEGER
	Limits or toggle field entries:	MIN: 0 MAX: 10
	Attributes:	
	Input mode:	Read-only
	Alignment of short text:	Right-justified
	Position of short text:	
	Width:	300

See also

Variable parameters (Page 69)

5.5 Example 3: Assigning the Variable Type, Default, System or User Variable, Input/Output Field Position properties

Example 3

The following example defines a variable for which the properties variable type, default setting, system or user variable and position are set.

DEF Var3 = (R//10///"\$R[1]"//300,10,200//)		
	Variable type:	REAL
	Default setting:	10
	System or user variable:	\$R[1] (R-Parameter 1)
	Position of short text:	Default position in relation to input/output field
	Position of input/output field:	
	Distance from left:	300
	Distance from top:	10
	Width:	200

See also

Variable parameters (Page 69)

5.6 Example 4: Toggle field and list field

Example 4a

Various entries in the toggle field:

```

DEF Var1 = (I/* 0,1,2,3)           ;Simple toggle field to toggle numeric values
DEF Var2 = (S/* "On", "Off")       ;Simple toggle field to toggle strings
DEF Var3 = (B/* 1="On", 0="Off")    ;Extended toggle field to toggle numeric values whereby a
                                   ;display text is assigned to each numeric value
DEF Var4 = (R/* ARR1)              ;Toggle field which obtains its values to be toggled from an
                                   ;array

```

Example 4b

The list field corresponds to the configuration of a toggle field, but the display type for the list field (variable attribute DT = 4) must also be set.

DEF VAR_LISTBOX_Text = (S/*\$80000,\$80001,\$80002,\$80003,\$80004/\$80001//DT4////200,,340,60)		
	Variable type:	STRING
	Limits / toggle field:	*\$80000,\$80001,\$80002,\$80003,\$80004 (List of the language-dependent texts to be displayed)
	Default setting:	\$80001
	Attributes:	
	Display type:	4 (list field)
	Position of input/output field:	
	Distance from left:	200
	Width:	340
	Height:	60
	Colors:	
	Foreground color:	6 (blue)
	Background color:	10 (white)

5.7 Example 5: Image display**Example 5**

Displaying an image instead of a short text: The size and position of the image is defined under "Position of input/output field (left, top, width, height)".

DEF VAR6= (V///,"\\image1.png" ////160,40,50,50)		
	Variable type:	VARIANT
	Texts:	
	Short text:	image1.png
	Position of input/output field:	
	Distance from left:	160
	Distance from the top:	40
	Width:	50
	Height:	50

5.8 Example 6: Progress bar

The progress bar is a special display type of the input/output field and is designed for display without input.

Basically there are two progress bar types:

1. Progress bars with up to two color changes, e.g. for temperature or utilization display (see example 6a)
2. Progress bars to display the progress (without color change) in the Operate style (see example 6b)

Example 6a

Progress bar with two color changes:

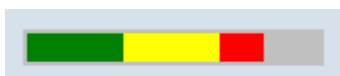


Figure 5-1 Progress bar with two color changes

DEF PROGGY0 = (R/0,150,50,100///DT1,DO0//"\$R[10]"//,,150/3,4,,,,,9,7)		
	Variable type:	REAL
	Limits / toggle field:	
	MIN:	0
	MAX:	150
	Signal value SVAL1:	50
	Signal value SVAL2:	100
	Attributes:	
	Display mode DT:	1 (progress bar)
	Display option DO:	0 (from left to right (default))
	System or user variable:	\$R[10]
	Position of input/output field:	
	Width:	150
	Colors:	
	Foreground color:	3 (dark green)
	Background color:	4 (light gray)
	Signal color SC1:	9 (yellow)
	Signal color SC2:	7 (red)

To use a progress bar with color change, the display mode DT (DisplayType) must be set to 1.

The orientation of the progress bar is determined via the attribute display option DO (DisplayOption):

- 0: From left to right (default)
- 1: From right to left
- 2: From bottom to top
- 3: From top to bottom

A MIN and a MAX value must be specified for the display of the progress bar (in the example MIN: 0, MAX: 150).

5.8 Example 6: Progress bar

Depending on the current value of the variable PROGGY0, this setting already displays a progress bar with the foreground color 3 (= dark green) and the background color 4 (= light gray).

Optionally, one or two signal values SVAL1 and SVAL2 (limit parameters) can be defined (in the example SVAL1: 50 and SVAL2: 100). With these signal values, the foreground color of the progress bar changes. The appropriate signal colors are specified via parameters SC1 and SC2 (in the above example SC1: 9 (= yellow) and SC2: 7 (=red)).

The following applies for the specification of the limits for the progress bar:

MIN < SVAL1 < SVAL2 < MAX.

Example 6b

Progress bar without color change:



Figure 5-2 Progress bar

DEF PROGGY0 = (R/0,150//DT2,DO0//"\$R[10]"//,,150/6,10)		
Variable type:		REAL
Limits / toggle field:		
	MIN:	0
	MAX:	150
Attributes:		
	Display type DT:	2 (progress bar)
	Display option DO:	0 (from left to right (default))
System or user variable:		\$R[10]
Position of input/output field:		
	Width:	150
Colors:		
	Foreground color:	6 (blue)
	Background color:	10 (white)

To use a progress bar without color change, the display mode DT (DisplayType) must be set to 2.

The orientation of the progress bar is determined via the attribute display option DO (DisplayOption) (see description for example 6a).

A MIN and a MAX value must be specified for the display of the progress bar (in the example MIN: 0, MAX: 150).

Depending on the current value of the variable PROGGY0, this setting displays a progress bar with the foreground color 6 (= blue) and the background color 10 (= white).

5.9 Example 7: Password input mode (asterisk)

Example 7

To implement a field with hidden input, e.g. for the input of a password, the display mode DT must be set to 5. Asterisks are then displayed instead of the entered characters.



Figure 5-3 Password input mode (asterisk)

DEF VAR_SET_PWD=(S//"/DT5)		
	Variable type:	STRING
	Default setting:	Empty string
	Attributes:	
	Display mode DT:	5 (password input mode)

5.10 Variable parameters

Parameter overview

The following overview provides a brief explanation of the variable parameters. Subsequent chapters contain a more detailed description.

Parameter	Description
Variable type (Page 75)	The variable type must be specified.
	R[x]: REAL (+ digit for the decimal place) I: INTEGER S[x]: STRING (+ digit for string length) C: CHARACTER (individual character) B: BOOL V: VARIANT

5.10 Variable parameters

Parameter	Description	
Limits (Page 64)	Limit value MIN, limit value MAX Default setting: Empty The limit values are separated by a comma. Limits can be specified for types I, C and R in decimal formats or as characters in the form "A", "F". Optionally, two signal colors SC1 and SC2 can be configured for the display of a progress bar with color change (variable attribute DT = 1) which are displayed as the respective foreground color of the bar when the signal value SVAL1 or SVAL2 is exceeded. The signal values are specified as integer values. See application example for Progress bar (Page 66).	
Default setting (Page 79)	If no default setting has been configured and no system or user variable has been assigned to the variable, the first element of the toggle field is assigned. If no toggle field has been defined, there is no default setting, which means the status of the variable is "not calculated". Default setting: No default	
Toggle field (Page 78)	List with predetermined entries in the IO field: The list is initiated by a *; the entries are separated by a comma. The entries can be assigned a value. For the toggle field, the entry for the limit is interpreted as a list. If only one * is entered, a variable toggle field is created. Default setting: None	
Texts (Page 63)	The sequence is specified. Instead of a short text, an image can also be displayed. Default setting: Empty	
	Long text: Short text: Graphic text: Unit text: Tooltip	Text in the display line Name of the dialog element Text refers to the terms in the graphics Unit of the dialog element Serves as brief information in a screen form configuration for the display and toggle fields. The information is configured via plain text and language text ID.

Parameter	Description	
Attributes (Page 64)	The attributes influence the following properties: • Display mode • Display option • Update rate • Toggle symbol • Tooltip • Input mode • Access level • Alignment of short text • Font size • Limits The attributes are separated by commas and appear in any order. A definition can be made for each component.	
	Display mode	dt0: Standard (input/output field or toggle field) (default) dt1: Progress bar with color change dt2: Progress bar without color change in the Operate style dt4: List field dt5: Password input mode (asterisk) dt6: Input/output field with multi-line display
	Display option	Particularly, e.g. for the display modes dt1 and dt2 (progress bars), a display option may also have to be configured in combination. do0: From left to right (default) do1: From right to left do2: From bottom to top do3: From top to bottom
	Update rate	The attribute UR (update rate) controls the update of the display and therefore the processing of the associated, configured CHANGE block of variables or grid columns. Depending on the configuration, the CPU load can be drastically reduced and therefore shorter response times achieved on the user interface. ur0: SICap::standardUpdateRate() (currently = 200 ms, default value) ur1: 50 ms ur2: 100 ms ur3: 200 ms ur4: 500 ms ur5: 1000 ms ur6: 2000 ms ur7: 5000 ms ur8: 10000 ms
	Toggle symbol	tg0: Toggle symbol off (default) tg1: Toggle symbol on If the attribute TG is set to 1, the toggle symbol also appears in the tooltip of the input field. Example:

Parameter	Description
	DEF OFFS = (R//123.456/,,,,"My ToolTip"/TG1)
Tooltip	The tooltip text can also be changed during runtime via the variable property "TT". Example: <pre>PRESS (VS1) MyVar.TT = "My new ToolTip" END_PRESS</pre> or <pre>DEF MyVar=(R///,,,,"My ToolTip-text")</pre>
Input mode	wr0: IO field invisible, short text visible wr1: Read (no focus possible for input) wr2: Read and write (line appears in white) wr3: wr1 with focus wr4: All variable elements invisible, no focus possible wr5: The value entered is saved immediately on every keystroke (in contrast to wr2, where it is only saved when the field is exited or RETURN is pressed). Default setting: wr2
Access level (Page 202)	Empty: Can always be written ac0...ac7: Protection levels If the access level is not adequate, then the first line is displayed in gray, default setting: ac7
Alignment of short text	al0: Left-justified al1: Right-justified al2: Centered Default setting: al0
Font size	fs1: Default font size (8 pt.) fs2: Double font size Default setting: fs1 The clearances between the lines is defined. With the default font size, 16 lines will fit into the dialog. Graphics and unit text can only be configured in the default font size.
Limits	Consequently, it is possible to check whether the values of the variable are within the MIN and MAX limits specified. Default setting: Determined by specified limits li0: No check li1: Check with respect to min. li2: Check with respect to max. li3: Check with respect to min. and max.
Behavior when opening	cb attributes specified for a variable in a variables definition take priority over the cb default setting in the dialog definition. Multiple attributes are separated by commas (see also Behavior when opening the dialog (attribute CB) (Page 204)).
CB0:	The CHANGE method is triggered when the screen is displayed if the variable has a valid value at this time (e.g. through default setting or NC/PLC variable).
CB1:	

Parameter	Description	
		The CHANGE method is not explicitly triggered when the screen is displayed. If the variable has a configured NC/PLC variable, then the CHANGE method is of course still called.
Help display (Page 63)	Help display file:	Name of the png file Default setting: Empty
	The name of the Help display file appears in double quotation marks. The display appears automatically (instead of the previous graphic) if the cursor is positioned on this variable.	
System or user variable (Page 65)	System or user data from the NC/PLC can be assigned to the variable. The system or user variable appears in double quotation marks. Reference: List Manual System Variables, /PGAsI/	
Position of short text (Page 80)	Position of short text (distance from left, distance from top, width) The positions are entered in pixels and relate to the upper left-hand corner of the main body of the dialog. The entries are separated by commas.	
Position of input/output field (Page 80)	Position of input/output field (distance from left, distance from top, width, height) The positions are entered in pixels and relate to the upper left-hand corner of the main body of the dialog. The entries are separated by commas. If this position changes, the positions of the short text, graphic text and unit text also change.	
Colors (Page 63)	Foreground and background colors for input/output fields, short texts, graphic texts, unit texts and signal colors for progress bars: The colors are separated by a comma. For specification of the color, see Chapter List of colors (Page 202). Default setting for input/output field: Foreground color: Black, background color: White The default colors of the input/output field depend on the "wr" write mode: Default setting for short text, graphic text, unit text: Foreground color: Black, background color: Transparent.	
	Optionally, two signal colors SC1 and SC2 can be configured for the display of a progress bar with color change (variable attribute DT = 1) which are displayed as the respective foreground color of the bar when the signal value SVAL1 or SVAL2 is exceeded. See application example for Progress bar (Page 66).	
	The colors are expected in the following order in the variables definition: 1. Foreground color of the input/output field: FC 2. Background color of the input/output field: BC 3. Foreground color of the short text: FC_ST 4. Background color of the short text: BC_ST 5. Foreground color of the graphic text: FC_GT 6. Background color of the graphic text: BC_GT 7. Foreground color of the unit text: FC_UT 8. Background color of the unit text: BC_UT 9. Signal color 1 10. Signal color 2	

5.10 Variable parameters

Parameter	Description
Online help file (Page 58)	The name of the online help file appears in double quotation marks.
I/O field with integrated unit selection	You can use the input/output fields with integrated unit selection to toggle between different units. If the cursor is moved to the input field, the integrated unit selection is highlighted (focus does not have to be explicitly placed on it). A tooltip with a toggle symbol is also displayed with the appropriate reference to this functionality. Examples: <pre>DEF VarEdit=(R/////////200,,100// "VarTgl") VarTgl=(S/*0="mm",1="inch"/0//WR2////302,,40)</pre> or <pre>DEF VarEdit_2={TYP="R", VAL=1.234, X=200, W=100, LINK_TGL="vartgl_2"}, VARTgl_2={TYP="S", TGL="* 0="mm", 1="inch", WR=2, X=302, W=40}</pre>

Variable: Changing properties

The variables are assigned a new value in the notation *Identifier.Property = Value* when a change is made. The expression to the right of the equality sign is evaluated and assigned to the variable or variable property.

Identifier. ac = Access level	(ac: access level)
Identifier. al = Text alignment	(al: alignment)
Identifier. bc = Background color of the input/output field	(bc: back color)
Identifier. bc_gt = Background color of the graphic text	(bc: back color) (gt: graphic text)
Identifier. bc_st = Background color of the short text	(bc: back color) (st: short text)
Identifier. bc_ut = Background color of the unit text	(bc: back color) (ut: unit text)
Identifier. do = Display option	(do: display option)
Identifier. dt = Display mode	(dt: display type)
Identifier. fc = Foreground color of the input/output field	(fc: front color)
Identifier. fc_gt = Foreground color of the graphic text	(fc: front color) (gt: graphic text)
Identifier. fc_st = Foreground color of the short text	(fc: front color) (st: short text)
Identifier. fc_ut = Foreground color of the unit text	(fc: front color) (ut: unit text)
Identifier. fs = Font size	(fs: font size)
Identifier. gt = Graphic text	(gt: graphic text)
Identifier. hlp = Help display	(hlp: help)
Identifier. li = Limit	(li: limit)
Identifier. lt = Long text	(lt: long text)
Identifier. max = MAX limits	(max: maximum)
Identifier. min = MIN limits	(min: minimum)

Identifier. sc = Signal color	(sc: signal color)
Identifier. st = Short text	(st: short text)
Identifier. tg = Toggle symbol	(tg: toggle)
Identifier. tt = Tooltip	(tt: tooltip)
Identifier. typ = Variable type	(typ: type)
Identifier. ur = Update rate	(ur: update rate)
Identifier. ut = Unit text	(ut: unit text)
Identifier. val = Variable value	(val: value)
Identifier. var = System or user variable	(var: variable)
Identifier. vld = Variable status	(vld: validation)
Identifier. wr = Input mode	(wr: write)

See also

Extended configuration syntax (Page 36)

5.11 Details on the variable type

Variable type INTEGER

The following extensions for determining the display in the input/output field and the memory utilization are possible for the "INTEGER" type:

2nd character in the extension data type

Display format	
B	Binary
D	Decimal signed
H	Hexadecimal
No data	Decimal signed

3rd and/or 4th character in the extension data type

Memory utilization	
B	Byte
W	Word
D	Double word
BU	Byte, Unsigned
WU	Word, Unsigned
DU	Double word, Unsigned

Sequence of characters in the INTEGER data type

1. "I" Basic INTEGER designation
2. Display format
3. Memory utilization
4. "U" Unsigned

Valid INTEGER type specifications:	
IB	Integer variable 32 bits in binary notation
IBD	Integer variable 32 bits in binary notation
IBW	Integer variable 16 bits in binary notation
IBB	Integer variable 8 bits in binary notation
I	Integer variable 32 bits in decimal notation signed
IDD	Integer variable 32 bits in decimal notation signed
IDW	Integer variable 16 bits in decimal notation signed
IDB	Integer variable 8 bits in decimal notation signed
IDDU	Integer variable 32 bits in decimal notation unsigned
IDWU	Integer variable 16 bits in decimal notation unsigned
IDBU	Integer variable 8 bits in decimal notation unsigned
IH	Integer variable 32 bits in hexadecimal notation
IHDU	Integer variable 32 bits in hexadecimal notation
IHWU	Integer variable 16 bits in hexadecimal notation
IHBU	Integer variable 8 bits in hexadecimal notation

VARIANT variable type

The VARIANT variable type is determined by the data type of the last value assignment. If the assigned or entered value starts with '-', '+', '.' or a number ('0'-'9'), then the value is interpreted as numeric. In all other cases as a string.

It can be scanned using the ISNUM or ISSTR functions. The VARIANT type is mainly suited to the purpose of writing either variable names or numerical values to the NC code.

Programming

The data type of variables can be checked:

Syntax:	ISNUM (VAR)	
Parameter:	VAR	Name of the variable whose data type is to be checked.
	FALSE = TRUE =	The result of the scan can be: not a numerical variable (data type = STRING) numerical variable (data type = REAL)

Syntax:	ISSTR (VAR)	
Parameter:	VAR	Name of the variable whose data type is to be checked.
	FALSE = TRUE =	The result of the scan can be: numerical variable (data type = REAL) not a numerical variable (data type = STRING)
Example:	<pre>IF ISNUM(VAR1) == TRUE IF ISSTR(REG[4]+2) == TRUE</pre>	

The display mode of variables can be changed:

- For INTEGER, the display type can be changed.

B	Binary
D	Decimal signed
H	Hexadecimal
unsigned	
With the addition of U for Unsigned	

- For REAL data types, only the number of places after the decimal point can be changed. Changing the type is illegal and generates an error message in the easyscreen_log.txt. file.
Example:

```
Var1.typ = "IBW"
Var2.typ = "R3"
```

Numerical formats

Numbers can be represented in either binary, decimal, hexadecimal or exponential notation:

Binary	B01110110
Decimal	123.45
Hexadecimal	HF1A9
exponential	-1.23EX-3
Examples:	
	VAR1 = HF1A9
	REG[0]= B01110110
	DEF VAR7 = (R/-1.23EX-3)

Note

When codes are generated with the "GC" function, only numerical values in decimal or exponential notation are evaluated, but **not** those in binary or hexadecimal notation.

See also

Variable parameters (Page 69)

5.12 Details on the toggle field

Description

The toggle field extension function displays texts (entries in toggle field) as a function of NC/PLC variables. A variable, which makes use of a toggle field extension, is read-only. The list of the toggle field is opened by pressing the ENTER key.

Programming

Syntax:	DEF VAR1=(IB/+ \$85000/15////"DB90.DBB5") or DEF VAR_TGL = (S/* "Hello", "Run", "MyScreens"/"Run")	
Description:	When the dialog is opened, the content of text number \$85015 is displayed in the IO field. Default value 15 is entered in system variable DB90.DBB5. If the value saved in system variable DB90.DBB5 changes, the displayed text number \$(85000 + <DB90.DBB5>) is recalculated in response to every change.	
Parameter:	Variable type	Type of variables specified in the system or user variable.
	Text number	Number (basis) of the language-specific text valid as the basis number.
	System or user variable	System or user variable (offset) via which the final text number (basis + offset) is displayed.

Toggle-field-dependent displays

The toggle field is overlaid with graphics, which change depending on the value of the memory byte. If the value of the memory byte is 1, "image1.png" will appear. If it is 2, "image2.png" will appear.

```
DEF VAR1=(IDB/*1="\image1.png", 2="\image2.png"//,$85000/wr1//"MB[130]"//160,40,50,50)
```

The size and position of the image is defined under "Position of IO field (left, top, width, height)".

Virtual toggle key

There is no list for toggle fields without configured list, e.g. `DEF NoTglList=(R/*)`. The next element is not generated until the toggle key is actuated or the associated `CHANGE()` method of the corresponding variable run through.

In this case, a small virtual keyboard (comprising only a toggle key) is displayed on the right next to the variable toggle field for operation with a touch panel.

The display of the virtual toggle key can also be forced via the following entry in the configuration file `slguiconfig.ini` irrespective of a touch panel.

```
[VirtualKeyboard]
```



```
; Forces easyscreen virtual toggle keyboard function
ForceEasyscreenVirtualToggleKey = true
```

See also

Variable parameters (Page 69)

5.13 Details on the default setting

Overview

A variable can assume various states depending on whether a default value, or a system or user variable, or both, is assigned to the variable field (input/output field or toggle field), (not calculated: toggling is only possible if a valid value has been assigned to the variable).

Scope of the default settings

Condition			Reaction of field type
Field type	Default setting	System or user variable	
I/O field	Yes	Yes	Write default value to system or user variable
	No	Yes	Use system or user variable as default value
	Fault	Yes	Not calculated, system or user variable is not written/used.
	Yes	No	Default setting
	No	No	Not calculated
	Fault	No	Not calculated
	Yes	Fault	Not calculated
	No	Fault	Not calculated
	Fault	Fault	Not calculated
Toggle	Yes	Yes	Write default value to system or user variable
	No	Yes	Use system or user variable as default value
	Fault	Yes	Not calculated, system or user variable not written/used
	Yes	No	Default setting
	No	No	Default = first toggle field element
	Fault	No	Not calculated
	Yes	Fault	Not calculated
	No	Fault	Not calculated
	Fault	Fault	Not calculated

See also

Variable parameters (Page 69)

5.14 Details on the position of the short text, position of the input/output field

Overview

The short text and graphic text, as well as the input/output field and unit text, are each treated like a unit, i.e., position settings for short text apply to the graphic text and settings for the input/output field and to unit text.

Programming

The configured position entry overwrites the default value, i.e., only one value can be changed. If no position settings have been configured for subsequent screen form elements, then the position settings for the preceding screen form element are applied.

If no positions have been specified for any dialog elements, the default setting is applied. By default, the column width for the short text and input/output field is calculated for each line based on the number of columns and maximum line width, i.e.,
 $\text{column width} = \text{maximum line width} / \text{number of columns}$.

The width of the graphics and unit text is predefined and optimized to suit the requirements of programming support. If graphics or unit text has been configured, the width of the short text or I/O field is reduced accordingly.

The order of short text and I/O field can be reversed by position settings.

See also

Variable parameters (Page 69)

5.15 Use of strings

Strings

Strings can be used as part of the configuration. These allow text to be displayed dynamically or different texts to be chained for the purpose of code generation.

Rules

The following rules must be observed with regard to string variables:

- Logic operations are processed from left to right.
- Nested expressions are solved from the inside outwards.

- No distinction is made between uppercase and lowercase type.
- String variables are generally displayed left justified.

Strings can be deleted simply by assigning a blank string.

Strings can be appended after the equality sign using the operator "<<". Quotation marks (") in the string are represented by two successive quotation mark symbols. Strings can be checked for equality in IF instructions.

Example

Default settings for the following examples:

```
VAR1.VAL = "This is an"
```

```
VAR8.VAL = 4
```

```
VAR14.VAL = 15
```

```
VAR2.VAL = "Error"
```

```
$85001 = "This is an"
```

```
$85002 = "Alarm text"
```

Editing strings:

- Chaining of strings:

```
VAR12.VAL = VAR1 << " Error." ;Result: "This is an error"
```

- Deleting a variable:

```
VAR10.VAL = "" ;Result: Blank string
```

- Setting a variable with a text variable:

```
VAR11.VAL = VAR1.VAL ;Result: "This is an"
```

- Data type matching:

```
VAR13.VAL = "This is the " << (VAR14 - VAR8) << ". error"
;Result: "This is the 11th error"
```

- Treatment of numerical values:

```
VAR13.VAL = "Error" << VAR14.VAL << ": " << $85001 << $85002
;Result: "Error 15: "This is an alarm text"
IF VAR15 == "Error" ;Strings in IF statement
VAR16 = 18.1234
;Result: VAR16 equals 18.1234,
;if VAR15 equals "Error".
ENDIF
```

- Quotation marks within a string:

```
VAR2="Hello, this is a " Test""
;Result: Hello, this is a " Test"
```

- System or user-variable strings dependent on variable content:

```
VAR2.Var = "$R[" << VAR8 << "]" ;Result: $R[4]
```

See also

STRING functions (Page 151)

5.16 CURPOS variable

Description

The CURPOS variable calls or manipulates the position of the cursor in the active input field of the current dialog. The variable shows how many characters are in front of the cursor. If the cursor is located at the start of the input field, then CURPOS assumes the value of 0. If the value of CURPOS is changed, then the cursor is positioned at the appropriate location in the input field.

In order to be able to respond to changes in the variable value, it is possible to monitor for changes using a CHANGE method. If the value of CURPOS changes, then a jump is made to the CHANGE method and the instructions contained there are executed.

5.17 CURVER variable

Description

The CURVER (CURrent VERsion) property allows the programming to be adapted in order to handle different versions. The CURVER variable is read-only.

Note

Even if previously recompiled with an older version, the code is automatically generated with the most recent version. The "GC" command always generates the most recent version. An additional identifier indicating the generated version is inserted in the user comment of the generated code in versions > 0.

Rules

The most recent dialog with all its variables is always displayed.

- Variables used previously may not be changed.
- New variables are inserted in the existing (cycle) programming in arbitrary order.
- It is not permissible to delete variables from a dialog from one version to the next.
- The dialog must contain all variables of all versions.

Example

```
(IF CURVER==1 ...) ; When the code is recompiled, CURVER is automati-  
                    cally assigned the version of the recompiled  
                    code.
```

5.18 ENTRY variable

Description

The ENTRY variable can be used to check by what method a dialog has been called.

Programming

Syntax:	ENTRY	
Description:	The ENTRY variable is a read only variable.	
Return Value:		The result of the scan can be:
	0 =	No programming support
	1 =	Start of a screen via softkey; no code has been generated yet (default as configured)
	2 =	Start of a screen via softkey; code has been generated (default from the code generated last by this screen)
	3 =	Recompilation with user comment (# lines)
	4 =	Code_type = 0: NC code with user comment (# lines)
	5 =	Code_type = 1: NC code without user comment (# lines)

Example

```

IF ENTRY == 0
    DLGL("The dialog was not called during programming")
ELSE
    DLGL("The dialog was called during programming")
ENDIF

```

5.19 ERR variable

Description

Variable ERR checks whether the preceding line has been executed correctly.

Programming

Syntax:	ERR	
Description:	The ERR variable is read-only.	
Return value:		The result of the scan can be:
	FALSE =	previous line was executed error-free
	TRUE =	previous line was not executed error-free

Example

<pre> VAR4 = Thread[VAR1,"CDM",3] IF ERR == TRUE VAR5 = "Error accessing array" ELSE VAR5 = "All OK" ENDIF </pre>	<pre> ; Output value from array ; Query whether the value has been found in the array ; If the value has not been found in the ar- ray, the value "Error accessing array" is as- signed to the variables. ; If the value has been found in the array, the value "All OK" is assigned to the varia- bles. </pre>

5.20 FILE_ERR variable**Description**

Variable FILE_ERR can be used to check whether the preceding GC or CP command has been executed correctly.

Programming

Syntax:	FILE_ERR	
Description:	The FILE_ERR variable is read-only.	
Return value:		Possible results are:
	0 =	Operation okay
	1 =	Drive/path not available
	2 =	Path/file access error
	3 =	Drive not ready
	4 =	Incorrect file name
	5 =	File is already open
	6 =	Access denied
	7 =	Target path not available or not permitted
	8 =	Copy source same as target
	10 =	Internal error: FILE_ERR = 10 means that the error cannot be classified in the other categories.

Example

```

CP("D:\source.mpf","E:\target.mpf")
; Copy from source.mpf to E:\tar-
; get.mpf

IF FILE_ERR > 0
; Query whether error has occurred
  IF FILE_ERR == 1
; Query specific error numbers and
; output associated error text

    VAR5 = "Drive/path not available"
  ELSE
    IF FILE_ERR == 2
      VAR5 = "Path/file access error"
    ELSE
      IF FILE_ERR == 3
        VAR5 = "incorrect file name"
      ENDIF
    ENDIF
  ENDIF
ELSE
  VAR5 = "All OK"
; If no errors have occurred in CP
; (or GC), "All OK" is output
ENDIF

```

5.21 FOC variable

Description

With the variable FOC, the input focus (current active input/output field) is controlled in a dialog. The reaction of the cursor left, right, up, down as well as PGUP, PGDN are predefined.

Note

The FOC function must not be initiated as a result of a navigation event. The cursor position may only be changed in softkey PRESS methods, CHANGE methods, etc.

The FOC function cannot be applied to variables with input mode wr = 0 and wr = 4 or to Help variables.

Programming

Syntax:	FOC
Description:	The variable can be read and written.

Return value:	Read	The result is the name of the variable to which the FOC function has been applied.
	Write	It is possible to assign either a string or a numerical value. A string is interpreted as a variable name and a numerical value as a variable index.

Example

```

IF FOC == "Var1"                ; Read focus
    REG[1] = Var1
ELSE
    REG[1] = Var2
ENDIF

FOC = "Var1"                    ; The input focus is assigned to variable 1.
FOC = 3                        ; The input focus is assigned to the 3rd dialog element with WR ≥ 2.

```

See also

FOCUS (Page 99)

5.22 Variable S_ALEVEL

Description

The current access level can be queried in the configuration with the screen property S_ALEVEL.

Programming

Syntax: **S_ALEVEL**

Description: Query of the current access level

Return value: 0: System
 1: Manufacturer
 2: Service
 3: User
 4: Key switch 3
 5: Key switch 2
 6: Key switch 1
 7: Key switch 0

Example

```
REG[0] = S_ALEVEL
```

See also

ACCESSLEVEL (Page 96)

5.23 S_CHAN variable

Description

The S_CHAN variable determines the number of the current channel for display or evaluation purposes.

Programming

Syntax:	S_CHAN
Description:	Query of the current channel number
Return value:	Channel number

Example

```
REG[0] = S_CHAN
```

See also

CHANNEL (Page 98)

5.24 Variable S_CONTROL

Description

The current control name can be queried in the configuration with the screen property S_CONTROL.

Programming

Syntax: **S_CONTROL**
Description: Query of the current control name
Return value: The control name is the section name in the mmc.ini

Example

```
REG[0] = S_CONTROL
```

See also

CONTROL (Page 98)

5.25 Variable S_LANG

Description

The current language can be queried in the configuration with the screen property S_LANG.

Programming

Syntax: **S_LANG**
Description: Query of the current language
Return value: Language code from resources.xml, e.g. "deu", "eng", etc.

Example

```
REG[0] = S_LANG
```

See also

LANGUAGE (Page 100)

5.26 Variable *S_NCCODEREADONLY*

Description

The screen property *S_NCCODEREADONLY* is only relevant when a cycle is recompiled in the editor with a "Run MyScreens" dialog. *S_NCCODEREADONLY* determines whether a recompiled NC code from the editor can be changed or not.

The following applies for the return value:

- TRUE: The recompiled NC code from the editor can be changed
- FALSE: The recompiled NC code from the editor cannot be changed (read-only), because, for example, it is already in the preprocessing (= TRUE).

5.27 Variables *S_RESX* and *S_RESY*

Description

The current resolution or its X and Y component can be queried in the configuration with the screen properties *S_RESX* and *S_RESY*.

Example

```
REG[0] = S_RESY
```

See also

RESOLUTION (Page 105)

Programming commands

6.1 Operators

Overview

The following operators can be used when programming:

- Mathematical operators
- Relational operators
- Logic (Boolean) operators
- Bit operators
- Trigonometric functions

6.1.1 Mathematical operators

Overview

Mathematical operators	Identifier
+	Addition
-	Subtraction
*	Multiplication
/	Division
MOD	Modulo operation
()	Parentheses
AND	AND operator
OR	OR operator
NOT	NOT operator
ROUND	Round off numbers with decimal places

Example: `VAR1.VAL = 45 * (4 + 3)`

ROUND

The ROUND operator is used to round off numbers with up to 12 decimal places during execution of a dialog configuration. The variable fields cannot accept the decimal places in the display.

6.1 Operators

Use

ROUND is controlled by the user with two parameters:

```
VAR1 = 5.2328543
```

```
VAR2 = ROUND ( VAR1, 4 )
```

Result: VAR2 = 5.2339

VAR1 contains the number to be rounded. The parameter “4” indicates the number of decimal places in the result, which is placed in VAR2.

Trigonometric functions

Trigonometric functions	Identifier
SIN(x)	Sine of x
COS(x)	Cosine of x
TAN(x)	Tangent of x
ATAN(x, y)	Arc tangent of x/y
SQRT(x)	Square root of x
ABS(x)	Absolute value of x
SDEG(x)	Conversion to degrees
SRAD(x)	Conversion to radian
CALC_ASIN(x)	Arc sine of x
CALC_ACOS(x)	Arc cosine of x

Note

The functions operate with radian measure. The functions SDEG() and SRAD() can be used for conversion.

Example: VAR1.VAL = SQRT(2)

Mathematical functions

Mathematical functions	Identifier
CALC_CEIL(x)	Determines the next highest integer of x (round up)
CALC_FLOOR(x)	Determines the next lowest integer of x (round down)
CALC_LOG(x)	Determines the (natural) logarithm for the base e of x
CALC_LOG10(x)	Determines the logarithm for the base 10 of x
CALC_POW(x, y)	Determines x to the power of y
CALC_MIN(x, y)	Determines the lower number of x and y
CALC_MAX(x, y)	Determines the higher number of x and y

Random function: Random number

Syntax:	RANDOM (<i>lower limit value, upper limit value</i>)	
Description:	The RANDOM function returns a pseudo random number in a specified range.	
Parameters:	Lower limit value	Lower limit value >= -32767, Lower limit value < upper limit value
	Upper limit value	Upper limit value <= 32767

Example

REG[0] = RANDOM(-10,10)		; Possible result = -3

Constants

Constants	
<i>PI</i>	<i>3.14159265358979323846</i>
<i>FALSE</i>	<i>0</i>
<i>TRUE</i>	<i>1</i>

Example: VAR1.VAL = PI

Comparison operators

Comparison operators	
==	<i>Equal to</i>
<>	<i>Not equal to</i>
>	<i>Greater than</i>
<	<i>Less than</i>
>=	<i>Greater than or equal to</i>
<=	<i>Less than or equal to</i>

Example:

```
IF VAR1.VAL == 1
    VAR2.VAL = TRUE
ENDIF
```

6.1 Operators

Conditions

The nesting depth is unlimited.

Condition with one command:

```
IF
...
ENDIF
```

Condition with two commands:

```
IF
...
ELSE
...
ENDIF
```

6.1.2 Bit operators

Overview

Bit operators	Identifier
BOR	Bit-serial OR
BXOR	Bit-serial XOR
BAND	Bit-serial AND
BNOT	Bit-serial NOT
SHL	Shift bits to left
SHR	Shift bits to right

SHL operator

Bits are shifted to the left using the SHL (SHIFT LEFT) operator. You can specify both the value to be shifted and the number of shift increments directly or via a variable. If the limit of the data format is reached, the bits are shifted beyond the limit without displaying an error message.

Use

Syntax:	variable = value SHL increment	
Description:	Shift Left	
Parameters:	value	value to be shifted
	increment	number of shift increments

Example

```
PRESS (VS1)
```



```

VAR01 = 16 SHL 2          ; Result = 64
VAR02 = VAR02 SHL VAR04   ; Convert content of VAR02 to 32-bit unsigned
                           , and shift content to left by number of bits specified
                           in VAR04. Then convert 32-bit value back to format of
                           variable VAR02.
END_PRESS

```

SHR operator

Bits are shifted to the RIGHT using the SHR (SHIFT RIGHT) function. You can specify both the value to be shifted and the number of shift increments directly or via a variable. If the limit of the data format is reached, the bits are shifted beyond the limit without displaying an error message.

Use

Syntax:	variable = value SHR increment	
Description:	Shift Right	
Parameters:	value	value to be shifted
	increment	number of shift increments

Example

```

PRESS (VS1)
  VAR01 = 16 SHR 2          ; Result = 4
  VAR02 = VAR02 SHR VAR04   ; Convert content of VAR02 to 32-bit unsigned
                           , and shift content to right by number of bits
                           specified in VAR04. Then convert 32-bit value back
                           to format of variable VAR02.
END_PRESS

```

6.2 Methods

Overview

Various types of event (exit input field, actuate softkey) can initiate specific actions in dialogs and dialog-dependent softkey menus (softkey menus that are called from a newly configured dialog). These actions are configured in methods.

The following table shows the basic principle used to program a method:

Definition block	Comment	Chapter reference
PRESS (HS1)	;Method start identifier	
LM... LS...	;Functions	See chapter "Functions"
Var1.st = ...	;Changing properties	see chapter "Softkey menu" and chapter "Dialog elements"
Var2 = Var3 + Var4 ... EXIT	;Calculation with variables	See chapter "Defining variables"
END_PRESS	;Method end identifier	

6.2.1 ACCESSLEVEL

Description

The ACCESSLEVEL method is run through when the current access level has changed for an opened screen.

Programming

Syntax:	ACCESSLEVEL <instructions> END_ACCESSLEVEL
Description:	Access level
Parameters:	- None -

See also

Variable S_ALEVEL (Page 86)

6.2.2 CHANGE

Description

CHANGE methods are executed if a variable value changes, i.e., variable calculations that are performed as soon as a variable value changes are configured within a CHANGE method.

There are two types of CHANGE method, i.e., element-specific and global:

- The **element-specific CHANGE method** is executed if the value of a specified variable changes. If a system or user variable is assigned to a variable, cyclic updating of the variable value can be configured in a CHANGE method.
- The **global CHANGE method** is executed if the value of any variable changes and no element-specific CHANGE method has been configured.

"Element-specific" programming

Syntax:	CHANGE(<i>Identifier</i>) ... END_CHANGE	
Description:	Changes the value of a specific variable	
Parameters:	Identifier	Name of the variable

Example

```

DEF VAR1=(I////////"DB20.DBB1")          ; A system variable is assigned to Var1
CHANGE (VAR1)
  IF VAR1.Val <> 1
    VAR1.st="Tool OK!"                    ; If the value of the system variable ≠ 1, the
                                          ; short text of the variable states: Tool OK!

    otto=1
  ELSE
    VAR1.st="Attention: Error!"            ; If the value of the system variable = 1, the
                                          ; short text of the variable states: Attention:
                                          ; Error!

    otto=2
  ENDIF
  VAR2.Var=2
END_CHANGE

```

"Global" programming

Syntax:	CHANGE() ... END_CHANGE
Description:	Changes any variable value
Parameters:	- None -

Example

```
CHANGE()
    EXIT                                ; If any of the variable values change, the di-
                                      ;alog will be terminated.
END_CHANGE
```

See also

Behavior when opening the dialog (attribute CB) (Page 204)

6.2.3 CHANNEL

Description

The CHANNEL method is run through when the current channel has changed for an opened screen, i.e. a channel switchover has been performed.

Programming

Syntax:	CHANNEL <instructions> END_CHANNEL
Description:	Channel switchover
Parameters:	- None -

See also

S_CHAN variable (Page 87)

6.2.4 CONTROL

Description

The CONTROL method is run through when the current control has changed for an opened screen, i.e. typically at a 1:n switchover.

Programming

Syntax:	CONTROL <instructions> END_CONTROL
Description:	Control switchover
Parameters:	- None -

See also

Variable S_CONTROL (Page 87)

6.2.5 FOCUS

Description

The FOCUS method is executed if the focus (cursor) is positioned on another field in the dialog.

The FOCUS method must not be initiated as a result of a navigation event. The cursor position may only be changed in softkey PRESS methods, CHANGE methods, etc. The response of cursor movements is predefined.

Note

Within the FOCUS method, it is not possible to select a different variable, nor can a new dialog be loaded.

Programming

Syntax:	FOCUS ... END_FOCUS
Description:	Positions the cursor
Parameters:	- None -

Example

```
FOCUS
  DLGL("The focus has been placed on variable << FOC << ".)
END_FOCUS
```

See also

FOC variable (Page 85)

6.2.6 LANGUAGE

Description

The LANGUAGE method is run through when the current language has changed for an opened screen.

Programming

Syntax:	LANGUAGE <instructions> END_LANGUAGE
Description:	Language
Parameters:	- None -

See also

Variable S_LANG (Page 88)

6.2.7 LOAD

Description

The LOAD method is executed after the variable and softkey definitions (DEF Var1= ..., HS1= ...) have been interpreted. At this time, the dialog is not yet displayed.

Programming

Syntax:	LOAD ... END_LOAD
Description:	Download
Parameters:	- None -

Example

LOAD	; Start identifier

```

Screen form1.Hd = $85111 ; Assign text for dialog header from language file
VAR1.Min = 0             ; Assign MIN variable limit
VAR1.Max = 1000          ; Assign MAX variable limit
END_LOAD                 ; End code

```

See also

Line, dividing line, rectangle, circle and ellipse (Page 163)

6.2.8 UNLOAD**Description**

The UNLOAD method is executed before a dialog is unloaded.

Programming

Syntax:	UNLOAD ... END_UNLOAD
Description:	Unload
Parameters:	- None -

Example

```

UNLOAD
    REG[1] = VAR1          ; Save variable in register
END_UNLOAD

```

6.2.9 OUTPUT

Description

The OUTPUT method is executed if the "GC" function is called. Variables and Help variables are configured as an NC code in an OUTPUT method. The individual elements in a code line are linked by means of blanks.

Note

The NC code can be generated in an extra file by means of file functions and transferred to the NC.

Programming

Syntax:	OUTPUT (<i>Identifier</i>) ... END_OUTPUT	
Description:	Outputs variables in the NC program.	
Parameters:	Identifier	Name of OUTPUT method

Block numbers and skip identifiers

The OUTPUT method must not contain line numbers or skip identifiers if you wish to keep the line numbers and skip identifiers directly set with active program support in the parts program in case of recompilations.

Editor changes in the parts program produce the following response:

Condition	Response
Number of blocks remains unchanged.	Block numbers are retained.
Number of blocks is reduced.	The highest block numbers are canceled.
Number of blocks is increased.	New blocks are not numbered.

Example

```
OUTPUT(CODE1)
  "CYCLE82(" Var1.val "," Var2.val "," Var3.val ","Var4.val "," Var5.val ","
Var6.val ") "
END_OUTPUT
```


6.2.10 PRESS

Description

The PRESS method is executed when the corresponding softkey is pressed.

Programming

Syntax:	PRESS(<i>softkey</i>) ... END_PRESS		
Designation:	Pressing a softkey		
Parameter:	Softkey	Name of softkey: HS1 - HS8 and VS1 - VS8	
	RECALL	<RECALL> key	
	ENTER	For the <ENTER> key, see PRESS(ENTER) (Page 104)	
	TOGGLE	For the <TOGGLE> key, see PRESS(TOGGLE) (Page 104)	
	PU	Page Up	Screen up
	PD	Page Down	Screen down
	SL	Scroll left	Cursor left
	SR	Scroll right	Cursor right
	SU	Scroll up	Cursor up
	SD	Scroll down	Cursor down

Example

```

HS1 = ("another softkey menu")
HS2= ("no function")
PRESS (HS1)
    LS ("Menu1")                ; Load another softkey menu
    Var2 = Var3 + Var1
END_PRESS
PRESS (HS2)
END_PRESS
PRESS (PU)
    INDEX = INDEX -7
    CALL ("UP1")
END_PRESS

```

6.2.11 PRESS(ENTER)

Description

The PRESS(ENTER) method is always called when the Enter key is pressed for a variable with input/output field with input mode WR3 or WR5:

- WR3: Navigation to field and pressing of the Enter key
- WR5: In the input mode, transfer of the value with the Enter key

Programming

Syntax:	PRESS(ENTER) <instructions> END_PRESS
Description:	Enter key pressed
Parameters:	- None -

6.2.12 PRESS(TOGGLE)

Description

The PRESS(TOGGLE) method is always called when the toggle key is pressed irrespective of the currently focused variable.

When required, the FOC screen property can be used to determine which variable is currently in focus.

Programming

Syntax:	PRESS(TOGGLE) <instructions> END_PRESS
Description:	Toggle key pressed
Parameter:	- None -

Example

```
PRESS(TOGGLE)
  DLGL("Toggle key pressed at variable " << FOC)      ; The FOC screen property determines which
                                                         variable is currently in focus
END_PRESS
```

6.2.13 RESOLUTION

Description

The RESOLUTION method is run through when the current resolution has changed for an opened screen, i.e. typically at a TCU switchover.

Programming

Syntax:	RESOLUTION <instructions> END_RESOLUTION
Description:	Resolution
Parameters:	- None -

See also

Variables S_RESX and S_RESY (Page 89)

6.2.14 RESUME

Description

The RESUME method is called when the screen was interrupted, e.g. for an area change, and is now shown again. The value changes are processed again, if required, the timers started and the RESUME block executed.

Programming

Syntax:	RESUME <instruction> END_RESUME
Description:	Screen active again
Parameters:	- None -

6.2.15 SUSPEND

Description

The SUSPEND method is called when the screen is interrupted and not unloaded. This is the case, for example, when the screen is exited by a simple area change, but not explicitly unloaded. The screen is retained in the background, but no value change or timers are processed. The screen is paused.

Programming

Syntax:	SUSPEND <instruction> END_SUSPEND
Description:	Interruption of the screen
Parameters:	- None -

Example

```
SUSPEND
    MyVar1 = MyVar1 + 1
END_SUSPEND
```

6.2.16 Example: Version management with OUTPUT methods

Overview

Additional variables can be added to existing dialogs when expanding the user interface. A version identifier in parentheses is appended to the additional variables in the definition following the variable name: (0 = Original, is not written), 1 = Version 1, 2 = Version 2, etc.

Example:

```
DEF var100=(R//1)           ; Original, corresponds to Version 0
DEF var101(1)=(S// "Hello") ; Extension as of Version 1
```

When writing the OUTPUT method, you can specify which variables are written, with reference to a particular version identifier.

Example:

```
OUTPUT(NC1)                ; Only the variables of the original are offered in
                           the OUTPUT method.
```

```

OUTPUT (NC1,1)                                ; The variables of the original and the extensions
                                              with version identifier 1 are offered in the OUTPUT
                                              method.

```

The OUTPUT method for the original does not need a version identifier, however you can specify it with 0. OUTPUT(NC1) is equivalent to OUTPUT(NC1,0). Version identifier n in the OUTPUT method includes all variables of the originals 0, 1, 2, ... up to and including n.

Programming with version identifier

```

//M(XXX)                                     Version 0 (default)
DEF var100=(R//1)
DEF var101=(S// "Hello")
DEF TMP
VS8= ("GC")
PRESS (VS8)
    GC ("NC1")
END_PRESS

OUTPUT (NC1)
var100", "var101
END_OUTPUT

; ***** Version 1, extended definition *****
//M(XXX)
DEF var100=(R//1)
DEF var101=(S// "Hello")
DEF var102(1)=(V// "HUGO")
DEF TMP
VS8= ("GC")
PRESS (VS8)
    GC ("NC1")
END_PRESS
...

OUTPUT (NC1)                                Original and the new version in addition
var100", "var101
END_OUTPUT
...

OUTPUT (NC1,1)                               Version 1
var100", "var101", " var102
END_OUTPUT

```

6.3 Functions

Overview

A variety of functions are available in dialogs and dialog-dependent softkey menus. These can be activated by specific events (exit input field, actuate softkey) and configured in methods.

Subprograms

Repeatedly used configuring instructions or others, which define the process for a particular operation can be configured in subprograms. Subprograms can be loaded into the main program or other subprograms at any time and executed as often as necessary, i.e. the instructions they contain do not need to be configured repeatedly. The definition blocks of the dialogs/softkey menu constitute a main program.

External functions

Additional, user-specific functions can be integrated by means of external functions. The external functions are stored in a DLL file and identified by an entry in the definition lines of the configuration file.

PI services

The PI_START function starts PI Services (Program Invocation Services) from the PLC in the NC area.

See also

Function (FCT) (Page 127)

PI services (Page 141)

6.3.1 Reading and writing drive parameters: RDOP, WDOP, MRDOP

Description

You can read and write drive parameters with the aid of RDOP, WDOP and MRDOP functions.

Note

Do not read drive parameters faster than in a 1 second cycle, slower is better.

Reason: The communication with the drives can otherwise be extremely disturbed or even cause failures.

Note

If errors occur when reading and writing drive parameters, the screen property ERR is set appropriately.

Programming

Syntax:	RDOP ("Identifier of the drive object", "Parameter number")	
Description:	Reading a drive parameter (Drive Object Parameter)	
Parameter:	Identifier of the drive object	The identifier of the drive object can be found in the "DO name" column in "Drive system diagnostics" in the diagnostics operating area (> ETC > HSK 8: Drive system) (see figure below).
	Parameter number	

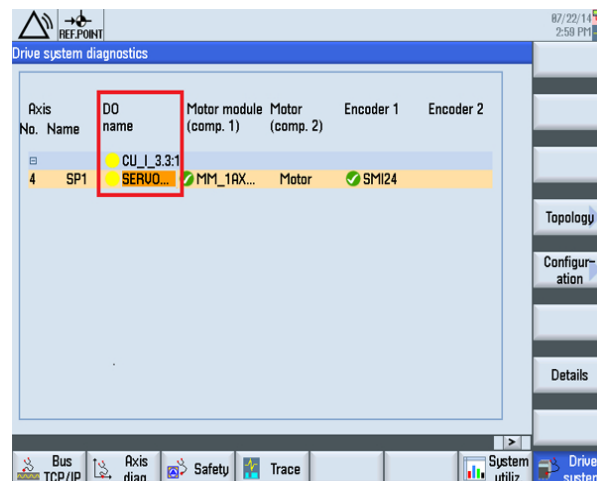


Figure 6-1 Identifier of the drive object

Syntax:	MRDOP ("Identifier of the drive object", "Parameter number1"*"Parameter number"[*...], Register index)
Description:	Multi-read of drive parameters. The MRDOP command transfers several drive parameters or drive objects in a single register access. This access method is significantly faster than reading via individual access attempts.

Parameter:	Identifier of the drive object	The identifier of the drive object can be taken, for example, from the root screen of the "Setup" operating area.
	Parameter number1 ... n	In the variable names, "*" is the separator. The values are transferred to register REG[Register index] in the order that the variable names appear in the command.
	Register index	The value of the first variable is located in REG[Register index]. The value of the second variable is located in REG[Register index + 1]

Syntax:	WDOP ("Identifier of the drive object", "Parameter number", Value)	
Description:	Writing a drive parameter (Drive Object Parameter)	
Parameter:	Identifier of the drive object	The identifier of the drive object can be taken, for example, from the root screen of the "Setup" operating area.
	Parameter number	Parameter number
	Value	Value to be written

Examples

Read motor temperature r0035 of drive object "SERVO_3.3:2":

```
MyVar=RDOP ("SERVO_3.3:2", "35") ;
```

Read motor temperature r0035 and actual torque value r0080 of drive object "SERVO_3.3:2" and store the respective results as of register index 10:

```
MRDOP ("SERVO_3.3:2", "35*80", 10)
```

6.3.2 Subprogram call (CALL)

Description

The CALL function calls a loaded subprogram from any point in a method. Subprogram nesting is supported, i.e. you can call a subprogram from another subprogram.

Programming

Syntax:	CALL ("Identifier")	
Description:	Subprogram call	
Parameter:	Identifier	Name of subprogram

Example

```
//M(SCREEN FORM1)
DEF VAR1 = ...
DEF VAR2 = ...
CHANGE (VAR1)
    ...
    CALL("MY_UP1")           ; Call subprogram and execute
    ...
END_CHANGE

CHANGE (VAR2)
    ...
    CALL("MY_UP1")           ; Call subprogram and execute
    ...
END_CHANGE

SUB (MY_UP1)
                                ;Do something

END_SUB
//END
```

6.3.3 Define block (//B)

Description

In the program file, subprograms are identified by the block identifier //B and terminated with //END. Several subprograms can be defined under each block identifier.

Note

The variables used in the subprogram must be defined in the dialog in which the subprogram is called.

Programming

A block is structured in the following way:

Syntax:	<pre>//B(Block name) SUB(Identifier) END_SUB [SUB(Identifier) ... END_SUB] ... //END</pre>	
Description:	Defines a subprogram	
Parameters:	Block name	Name of block identifier
	Identifier	Name of subprogram

Example

```
//B (PROG1)                ; Block start
SUB (UP1)                   ; Start of subprogram
...
  REG[0] = 5                ; Assign value 5 to register 0
...
END_SUB                     ; End of subprogram
SUB (UP2)                   ; Start of subprogram
  IF VAR1.val=="Otto"
    VAR1.val="Hans"
    RETURN
  ENDIF
  VAR1.val="Otto"
END_SUB                     ; End of subprogram
//END                       ; Block end
```

6.3.4 Check Variable (CVAR)

Description

You can use the CVAR (CheckVariable) function to run a scan to ascertain whether all or only certain variables or Help variables in a screen form are error-free.

It may be useful to check if variables contain a valid value before an NC code with the GC function.

A variable is error-free if the state of the variable Identifier.vld = 1.

Programming

Syntax:	CVAR(VarN)	
Description:	Checks variables for valid content	
Parameters:	VarN	List of variables to be checked. Up to 29 variables, each separated by a comma, can be checked. A character length of 500 must not be exceeded. The result of the scan can be:
	1 =	TRUE (all variables have valid content)
	0 =	FALSE (at least one variable has invalid content)

Example

```

IF CVAR == TRUE                                ; Check all variables
    VS8.SE = 1                                ; If all variables are error-free, soft-
                                           ; key VS8 is visible
ELSE
    VS8.SE = 2                                ; If a variable has an invalid value,
                                           ; softkey VS8 is disabled
ENDIF

IF CVAR("VAR1", "VAR2") == TRUE                ; Check variables VAR1 and VAR2
    DLGL ("VAR1 and VAR2 are OK")              ; If the values of VAR1 and VAR2 are er-
                                           ; ror-free, "VAR1 and VAR2 are OK" appears
                                           ; in the dialog line
ELSE
    DLGL ("VAR1 and VAR2 are not OK")          ; If the values of VAR1 and VAR2 are in-
                                           ; valid, "VAR1 and VAR2 are not OK" appears
                                           ; in the dialog line
ENDIF

```

6.3.5 CLEAR_BACKGROUND

Description

The CLEAR_BACKGROUND function deletes the LINE, RECT, ELLIPSE, V_SEPARATOR and H_SEPARATOR graphic elements.

See also

Line, dividing line, rectangle, circle and ellipse (Page 163)

6.3.6 Copy Program file function (CP)

Description

The CP (Copy Program) function copies files within the HMI file system or within the NC file system.

Programming

Syntax:	CP ("Source file", "Target file")	
Description:	Copies a file	
Parameter:	Source file	Complete path to the source file
	Target file	Complete path data of the target file

The return value (VAR1 defined as help variable) queries whether the function was successful:

```
CP ("//NC/MPF.DIR/HOHO.MPF", "//NC/MPF.DIR/ASLAN.MPF", VAR1)
```

Example

Application with return value:

```
CP ("//NC/MPF.DIR/HOHO.MPF", "//NC/MPF.DIR/ASLAN.MPF", VAR1)
CP ("CF_CARD:/wks.dir/MESS_BILD.WPD/MESS_BILD.MPF", "//NC/WKS.DIR/AAA.WPD/HOHO2.MPF", VAR1)
CP ("//NC/MPF.DIR/HOHO.MPF", "//NC/MPF.DIR/ASLAN.MPF") ; WPD must exist
```

Application without return value:

```
CP ("//NC/MPF.DIR/HOHO.MPF", "//NC/MPF.DIR/ASLAN.MPF")
CP ("CF_CARD:/mpf.dir/MYPROG.MPF", "//NC/MPF.DIR/HOHO.MPF")
CP ("//NC/MPF.DIR/HOHO.MPF", "CF_CARD:/XYZ/MYPROG.MPF") ; XYZ must exist
```

See also

Support of FILE_ERR: FILE_ERR variable (Page 84)

6.3.7 Delete Program file function (DP)

Description

The DP (Delete Program) function deletes a file from the passive HMI or active NC file system.

Programming

Syntax:	DP ("File")	
Description:	Delete file	
Parameter:	File	Complete path name of file to be deleted

Example

The following data management syntax is used for this function:

- With return value

```
DP (" //NC/MPF.DIR/MYPROG.MPF", VAR1)
DP (" //NC/WKS.DIR/TEST.WPD/MYPROG.MPF", VAR1)
DP (" //NC/CMA.DIR/MYPROG.SPF", VAR1)
VAR1 = 0      File was deleted.
VAR1 = 1      File was not deleted.
```

- Without return value:

```
DP (" //NC/MPF.DIR/MYPROG.MPF")
DP (" //NC/WKS.DIR/TEST.WPD/MYPROG.MPF")
DP (" //NC/CMA.DIR/MYPROG.SPF")
```

6.3.8 Exist Program file function (EP)

Description

The EP (Exist Program) function checks whether a particular NC program is stored on the specified path in the NC or HMI file system.

Programming

Syntax:	EP ("File")	
Description:	Checks the existence of the NC program	
Parameter:	File	Complete path to the file in the NC or HMI file system
Return value:	Name of a variable to which the result of the scan should be assigned	

The EP function can handle the new syntax and the old logic (with adapted syntax).

The file is directly addressed using a qualifying name:

```
//NC/MPF.DIR/XYZ.MPF
```

or

```
CF_CARD: /MPF.DIR/XYZ.MPF (points to /user/sinumerik/data/prog)
```

or

```
LOC: (corresponds to CF_CARD)
```

Examples of new syntax:

```
EP("//NC/WKS.DIR/TEST.WPD/XYZ.MPF",VAR1)
EP("CF_CARD:/mpf.dir/XYZ.MPF",VAR1)
EP("LOC:/mpf.dir/XYZ.MPF",VAR1)
; With return value:
; VAR1 = 0 File exists.
; VAR1 = 1 File does not exist.
```

Example of old syntax:

```
EP("/MPF.DIR/CFI.MPF", VAR1)
; With return value:
; VAR1 = M File is located in the HMI file system.
; VAR1 = N File is located in the NC file system.
; VAR1 = B File is located in the HMI and the NC file system.
```

Example

```
EP("/MPF.DIR/GROB.MPF",VAR1) ; Old - path now with / instead of \

IF VAR1 == "M"
    DLGL("File is located in the HMI file system")
ELSE
    IF VAR1 == "N"
        DLGL("File is located in the NC file directory")
    ELSE
        DLGL("File is located neither in the HMI nor in the NC
file system")
    ENDIF
ENDIF
```

6.3.9 Move Program file function (MP)

Description

The MP (Move Program) function copies files within the HMI file system or within the NC file system.

Programming

Syntax:	MP ("source", "target")	
	MP ("CF_CARD:/MPF.DIR/MYPROG.MPF", "//NC/MPF.DIR")	
Description:	Move file	
Parameter:	Source file	Complete path data
	Target file	Complete path data
Return value	Result of the query	

Examples

With return value:

```

MP("//NC/MPF.DIR/123.MPF", "//NC/MPF.DIR/ASLAN.MPF", VAR1)      ;Within NC
MP("//NC/MPF.DIR/123.MPF", VAR0, VAR1)                          ;Target via variable
MP(VAR4, VAR0, VAR1)                                             ;Source and target via variable
MP("CF_CARD:/mpf.dir/myprog.mpf", "//NC/MPF.DIR/123.MPF", VAR1) ;From CF card to NC
MP("//NC/MPF.DIR/HOHO.MPF", "CF_CARD:/XYZ/123.mpf", VAR1)      ;From NC to CF card

; With return value:
; VAR1 = 0 Executed
; VAR1 = 1 Not executed

```

6.3.10 Select Program file function (SP)

Description

The SP (Select Program) function selects a file in the active NC file system for execution, i.e. the file must be loaded into the NC beforehand.

Programming

Syntax:	SP ("File")	
Designation:	Select program	
Parameter:	"File"	Complete path name of NC file

Example

The following data management syntax is used for this function:

- With return value


```

SP("//NC/MPF.DIR/MYPROG.MPF", VAR1)
VAR1 = 0    File was loaded.

```

VAR1 = 1 File was not loaded without return value.

- Without return value:

```
SP (" //NC/MPF.DIR/MYPROG.MPF")
```

6.3.11 File accesses: RDFILE, WRFILE, RDLINEFILE, WRLINEFILE

Description

The RDFILE and WRFILE functions are available for read and write access to files with INI syntax.

The RDLINEFILE and WRLINEFILE functions are available for read and write access to individual lines of a file.

Programming

Syntax:	RDFILE ("File name + path", "Section", "Key")	
Description:	Reading from a file	
Parameter:	File name + path	Path and file name
	Section	Section in the INI file
	Key	Key in the INI file

Syntax:	WRFILE (Value, "File name + path", "Section", "Key")	
Description:	Writing to a file	
Parameter:	Value	Value to be written
	File name + path	Path and file name
	Section	Section in the INI file
	Key	Key in the INI file

Syntax:	RDLINEFILE ("File name + path", Line number)	
Description:	Reading a line from a file	
Parameter:	File name + path	Path and file name
	Line number	Line number Numbering begins at 0 for the first line.

Syntax:	WRLINEFILE (Value, "File name + path")	
Description:	Writing of a line at the end of a file	

Parameter:	Value	Value to be written
	File name + path	Path and file name

Note

- The files must not be in the file system of the NC (data management).
- If the file does not exist, or the end of the file is reached or other errors occur, the variables FILE_ERR and ERR are set accordingly. You can check whether a file exists first with the Exist Program (EP) file function.
- The processed file is generated in "UTF-8" coding (without BOM (Byte Order Mask)). When reading a file, it is expected in "UTF-8" coding.
- You can delete the file explicitly when required with the Delete Program (DP) file function.

Examples**Reading from an INI file:**

Requirement/assumption:

Content of file C:/tmp/myfile.ini:

```
<...>
[MyData]
MyName=Daniel
<...>
```

```
MyVar = RDFILE("C:/tmp/myfile.ini", "MyData", "MyName")
```

Result:

MyVar now contains the value "Daniel".

Writing to an INI file:

Requirement/assumption:

VARs=12

```
WRFILE(VARS, "C:/tmp/myfile.ini", "MySession", "NrOfSessions")
```

Result:

Content of file C:/tmp/myfile.ini:

```
<...>
[MySession]
NrOfSessions=12
<...>
```

Reading the 4th line of a file:

Requirement/assumption:

Content of c:/tmp/myfile.mpf:

```
R[0]=0
F3500 G1
MYLOOPIDX1: X0 y-50 Z0
    X150 Y50 Z10
    R[0]=R[0]+1
GOTOB MYLOOPIDX1
M30

MyVar = RDLINEFILE("C:/tmp/myfile.mpf", 4)
```

Result:

MyVar now contains the value " R[0]=R[0]+1 "

Writing at the end of a file:

Requirement/assumption:

```
VARX=123
VARY=456

WRLINEFILE("F100 X" << VARX << " Y" << VARY, "C:/tmp/mypp.mpf")
```

Result:

Content of c:/tmp/mypp.mpf:

```
<...>
F100 X123 Y456
```

6.3.12 Dialog line (DLGL)

Description

It is possible to configure short texts (messages or input tips) for output in the dialog line of the dialog in response to certain situations.

Possible number of characters in the default font size: approx. 50

Programming

Syntax:	DLGL ("String")	
Description:	Outputs text in the dialog line	
Parameter:	String	Text, which is displayed in the dialog line

Example

```

IF Var1 > Var2
                                ; The text "Value too large!" appears in the dialog
                                line if variable1 > variable2.
    DLGL("Value too large!")
ENDIF

```

6.3.13 DEBUG

Description

The DEBUG function provides an analysis help during the draft phase of Run MyScreen user screens. An expression in brackets transferred with the DEBUG function is evaluated during runtime. The result is attached to the "easyscreen_log.txt" logbook as a separate entry.

Each entry has the current time stamp in square brackets as prefix (see example below).

It is recommended that the DEBUG outputs are removed in time-critical parts when possible. Particularly after completing the draft phase, it is recommended that you comment out the DEBUG outputs. The write access to the constantly expanding "easyscreen_log.txt" logbook may slow down the processing.

Programming

Syntax:	DEBUG (<i>expression</i>)	
Description:	Makes an entry in the "easyscreen_log.txt" logbook	
Parameter:	Expression to be evaluated from which an entry is generated in the logbook	

Example

```

IF Var1 > Var2
    DEBUG("Value of ""Var1"": " << Var1)
                                ; Entry in the "easyscreen_log_txt:
                                [10:22:40.445] DEBUG: Value of "Var1":
                                123.456
ENDIF

```

6.3.14 Exit dialog (EXIT)

Description

The EXIT function is used to exit a dialog and return to the master dialog. If no master dialog is found, you will exit the newly configured user interfaces and return to the standard application.

Programming (without parameters)

Syntax:	EXIT
Description:	Exits a dialog
Parameters:	- None -

Example

```
PRESS (HS1)
  EXIT
END_PRESS
```

Description

If the current dialog has been called with a transfer variable, the value of the variables can be changed and transferred to the output dialog.

The variable values are each assigned to the variables transferred from the output dialog to the subsequent dialog using the "LM" function. Up to 20 variable values, each separated by a comma, can be transferred.

Note

The sequence of variables or variable values must be the same as the sequence of transfer values programmed for the LM function to preclude assignment errors. Any unspecified variable values will not be changed when the transfer is made. The modified transfer variables are immediately valid in the output dialog on execution of the LM function.

Programming with a transfer variable

Syntax:	EXIT [(VARx)]	
Description:	Exits the dialog and transfers one or more variables	
Parameters:	VARx	Label variables

Example

```
//M(Screen form1)
...
PRESS(HS1)
    LM("SCREEN FORM2","CFI.COM",1, POSX,
    POSY, DIAMETER)

                                ; Interrupt screen form1 and open screen
                                ; form2. In doing this, transfer variables
                                ; POSX, POSY and DIAMETER.

    DLGL("Screen form2 ended")    ; After returning from screen form2, the
                                ; following text appears in the dialog line
                                ; of screen form1: Screen form2 ended.

END_PRESS
...
//END

//M(Screen form2)
...
PRESS(HS1)
EXIT(5, , CALCULATED_DIAMETER)

                                ; Exit screen form2 and return to screen
                                ; form1 in the line after LM. In doing
                                ; this, assign the value 5 to the variable
                                ; POSX and the value of the CALCULATED_DI-
                                ; AMETER variable to the DIAMETER variable.
                                ; The variable POSY retains its current val-
                                ; ue.

END_PRESS
...
//END
```

6.3.15 Dynamic manipulation of the lists of toggle fields or list box fields**Description**

The LISTADDITEM, LISTINSERTITEM, LISTDELETEITEM and LISTCLEAR functions are used for the dynamic manipulation of the lists of toggle fields or list box fields.

These functions only have an effect on variables that have their own list, such as:

- "Simple" list

```
DEF VAR_AC1 = (I/* 0,1,2,3,4,5,6,7,8) or
```

- "Extended" list

```
DEF VAR_AC2 = (I/* 0="AC0", 1="AC1", 2="AC2", 3="AC3", 4="AC4",
5="AC5", 6="AC6", 7="AC7", 8="AC8").
```

If the variable points to an array, e.g. `DEF VAR_AC3 = (I/* MYARRAY)`, these functions are not available as otherwise the global array would be changed.

At least one value must be defined for the variable in the DEF line. This specifies the "Simple" or "Extended" list type.

However, the list can then be completely deleted and, if required, completely rebuilt. The "Simple" or "Extended" type must be retained however, it cannot be changed dynamically.

Programming

Syntax:	LISTINSERTITEM (<i>Variable name, Position, ItemValue[, ItemDispValue]</i>)	
Description:	Inserts an element at a specific position; reading from a file	
Parameter:	Variable name	
	Position	Position at which an element is to be inserted in the list
	ItemValue	Value of the list entry
	ItemDispValue	Value as it is to be displayed in the list

Syntax:	LISTADDITEM (<i>Variable name, ItemValue[, ItemDispValue]</i>)	
Description:	Adds an element to the end of the list	
Parameter:	Variable name	
	ItemValue	Value of the list entry
	ItemDispValue	Value as it is to be displayed in the list

Syntax:	LISTDELETEITEM (<i>Variable name, Position</i>)	
Description:	Deletes a specific element	
Parameter:	Variable name	
	Position	Position of the element to be deleted in the list

Syntax:	LISTCOUNT (<i>Variable name</i>)	
Description:	Returns the current number of elements in the list	
Parameter:	Variable name	

Syntax:	LISTCLEAR (<i>Variable name</i>)	
Description:	Deletes the complete list	
Parameter:	Variable name	

Examples

Note

The following examples build upon each other. The sequence of the examples is relevant in order to understand the respective results.

Requirement/assumption:

```
DEF VAR_AC = (I/* 0="Off",1="On"/1/,"Switch"/WR2)
```

Adding an element -1="Undefined" to the end of the list:

```
LISTADDITEM("VAR_AC", -1, ""Undefined"")
```

Result: 0="Off", 1="On", -1="Undefined"

Inserting an element 99="Maybe" at position 2:

```
LISTINSERTITEM("VAR_AC", 2, 99, ""Maybe"")
```

Result: 0="Off", 1="On", 99="Maybe", -1="Undefined"

Determining the current number of list elements:

```
REG[10]=LISTCOUNT("VAR_AC")
```

Result: REG[10] = 4

Deleting the element at position 1:

```
LISTDELETEITEM("VAR_AC", 1)
```

Result: 0="Off", 99="Maybe", -1="Undefined"

Deletion of the complete list:

```
LISTCLEAR("VAR_AC")
```

Result: List is empty

6.3.16 Evaluate (EVAL)

Description

The EVAL function evaluates a transferred expression and then executes it. With this function, expressions can be programmed during runtime. This can be useful, for example, for indexed access operations to variables.

Programming

Syntax:	EVAL (<i>exp</i>)	
Description:	Evaluates an expression	
Parameters:	exp	Logic expression

Example

```
VAR1= (S)
VAR2= (S)
VAR3= (S)
VAR4= (S)
CHANGE ()
    REG[7] = EVAL("VAR"<<REG[5]) ; The expression in parentheses produces VAR3 if the
                                value of REG[5] is equal to 3. The value of VAR3 is,
                                therefore, assigned to REG[7].

    IF REG[5] == 1
        REG[7] = VAR1
    ELSE
        IF REG[5] == 2
            REG[7] = VAR2
        ELSE
            IF REG[5] == 3
                REG[7] = VAR3
            ELSE
                IF REG[5] == 4
                    REG[7] = VAR4
                ENDIF
            ENDIF
        ENDIF
    ENDIF
END_CHANGE
```


6.3.17 Exit Loading Softkey (EXITLS)

Description

You can use the EXITLS function to exit the current user interface and load a defined softkey menu.

Programming

Syntax:	EXITLS ("Softkey menu"[, "Path"])	
Description:	Exits dialog and loads a softkey menu	
Parameters:	Softkey menu	Name of the softkey menu to be loaded
	Path name	Directory path of the softkey menu to be loaded

Example

```
PRESS (HS1)
    EXITLS( "Menu1", "AEDITOR.COM" )
END_PRESS
```

6.3.18 Function (FCT)

Description

The external functions are stored in a DLL file and identified by an entry in the definition lines of the configuration file.

Note

The external function must have at least one return parameter.

Programming

Syntax:	FCT <i>Function name = ("File"/Type of return/Types of permanent parameters/Types of variable parameters)</i>
	FCT InitConnection = ("c:\tmp\xyz.dll"/I/R,I,S/I,S)
Description:	An external function can, for example, be called from the LOAD method or executed in the PRESS method.

6.3 Functions

Parameters:	Function name	Name of external function
	File	Complete path to DLL file
	Type of return	Data type of the return value
	Type of fixed parameter	Value parameter
	Type of variable parameter	Reference parameter
	The data types are separated by commas.	

The external function can, for example, be called from the LOAD method or executed in the PRESS method.

Example:

```
press(vs4)
RET = InitConnection(VAR1,13,"Servus",VAR2,VAR17)
end_press
```

Structure of the external function

The external function must take into account a certain, specific signature:

Syntax:	extern "C" dllexport void InitConnection (ExtFctStructPtr FctRet, ExtFctStructPtr FctPar, char cNrFctPar)	
Description:	DLL export, only when implemented in Windows Specifiers and transfer parameters are strictly defined. The actual call parameters are transferred using the transferred structures.	
Parameters:	cNrFctPar	Number of call parameters = number of structure elements in FctPar
	FctPar	Pointer to a field of structure elements, which contain the particular call parameter with data type.
	FctRet	Pointer to a structure for the function value return with data type.

Definition of the transfer structure

```
union CFI_VARIANT
(
    char                b;
    short int           i;
    double              r;
    char*               s;
)
typedef struct ExtFctStructTag
(
```

```

char                                cTyp;
union CFI_VARIANT                    value;
)ExtFctStruct;
typedef struct ExtFct* ExtFctStructPtr;

```

If the external function is to be developed independently of the platform (Windows, Linux), then it is not permissible to use the keyword `__declspec(dllexport)`. This keyword is only required under Windows. For instance, the following macro can be used under Qt:

```

#ifdef Q_WS_WIN
    #define MY_EXPORT __declspec(dllexport)
#else
    #define MY_EXPORT
#endif

```

The function is declared as follows:

```

extern "C" MY_EXPORT void InitConnection
(ExtFctStructPtr FctRet, ExtFctStructPtr FctPar, char cNrFctPar)

```

If the screens, configured with "Run MyScreens", are used on the NCU and PCU/PC, then the extension of the binary file must be omitted:

```

FCT InitConnection = ("xyz"/I/R,I,S/I,S)

```

When the absolute path information is omitted, "Run MyScreens" first searches for the binary file in the configured directory.

6.3.19 Generate code (GC)

Description

The GC (Generate Code) function generates NC code from the OUTPUT method.

Programming

Syntax:	GC ("Identifier"[,"Target file"[,"Opt"],["Append"])
Description:	Generates NC code

Parameter:	Identifier	Name of the OUTPUT method from which code is generated
	Target file	Path name of target file for HMI or NC file system If the target file is not specified (only possible within Programming Support), the code will be written to the location of the cursor within the file that is currently open.
	Opt	Option for generating comments.
	0:	(Default setting) Generate code with comment for the purpose of recompilability (see also Recompile (Page 146)).
	1:	Do not create comments in the generated code. Note: This code cannot be recompiled (see also Auto-Hot-spot).
	Append	This parameter is only relevant if a target file is specified.
	0:	(Default setting) If the file already exists, the old content is deleted.
	1:	If the file already exists, the new code is written at the start of the file.
	2:	If the file already exists, the new code is written at the end of the file.

Example

```
//M(TestGC/"Code generation:")
DEF VAR1 = (R//1)
DEF VAR2 = (R//2)
DEF D_NAME
LOAD
    VAR1 = 123
    VAR2 = -6
END_LOAD
OUTPUT(CODE1)
    "Cycle123(" VAR1 "," VAR2 ")"
    "M30"
END_OUTPUT

PRESS(VS1)
    D_NAME = "\MPF.DIR\MESSEN.MPF"
    GC("CODE1",D_NAME)                ; Write NC code from the OUTPUT method to file
                                        \MPF.DIR\MESSEN.MPF:
                                        Cycle123(123, -6)
                                        M30
END_PRESS
```

Recompile

- **No entry for target file:**
The GC function can only be used in the Programming Support and writes the NC code to the file currently open in the editor. Recompilation of the NC code is possible. If the GC function is configured without a target file being specified under "Run MyScreens", an error message is output when it is executed.
- **Entry for target file:**
The code generated from the OUTPUT method is entered in the target file. If the target file does not already exist, it is set up in the NC file system. If the target file is stored in the HMI file system, it is stored on the hard disk. User comment lines (information required to recompile code) are not set up, i.e. the code cannot be recompiled.

Special considerations for target file specification

In principle, there are two different ways of specifying a target file:

- **NC notation:** `/_N_MPF_DIR/_N_MY_FILE_MPF`
The file is created in the MPF directory on the NC.
- **DOS notation:** `d:\abc\my_file.txt` or `\\RemoteRechner\files\my_file.txt`
The file is written to the specified directory on the hard disk or on the specified PC, provided that the directory is available on the hard disk or on a remote PC.

Note

Invalid variables generate an empty string in generated NC code and an error message in the log book when they are read.

Special features of recompilation

The GC function cannot be called in sub-dialogs because variables originating from master dialogs can be used in sub-dialogs. These variables would not, however, be available in response to a direct call.

When generated code is processed manually with the editor, the number of characters for values created by the code generation program must not be changed. Changing these values would make it impossible to recompile the code.

Remedy:

1. Recompile
2. Make change using the configured dialog (e.g. 99 → 101)
3. GC

6.3.20 Password functions

Overview

The following password functions are available:

- Set password
- Delete password
- Change password

HMI_LOGIN function: Set new password

Syntax:	HMI_LOGIN(<i>Passwd</i>)	
Description:	A password which sets the current access level is sent to the NCK with the HMI_LOGIN() function.	
Parameters:	Passwd	Password

Example

```
REG[0] = HMI_LOGIN("CUSTOMER")           ; Result = TRUE if the password has been set
                                           successfully, otherwise FALSE
```

HMI_LOGOFF function: Delete password

Syntax:	HMI_LOGOFF	
Description:	The current access level can be reset with the HMI_LOGOFF function.	
Parameters:	- None -	

Example

```
REG[0] = HMI_LOGOFF                       ; Result = TRUE if the password has been de-
                                           leted successfully, otherwise FALSE
```

HMI_SETPASSWD function: Change password

Syntax:	HMI_SETPASSWD(<i>AC</i>, <i>Passwd</i>)	
Description:	The password of the current level or a lower level can be changed with the HMI_SETPASSWD function.	
Parameters:	AC	Access level
	Passwd	Password

Example

```
REG[0] = HMI_SETPASSWD(4,"MYPWD")      ; Result = TRUE if the password has been
                                         changed successfully, otherwise FALSE
```

6.3.21 Load Array (LA)**Description**

The LA (Load Array) function can be used to load an array from another file.

Programming

Syntax:	LA (<i>Identifier</i> [, <i>File</i>])	
Description:	Loads an array from a file	
Parameters:	Identifier	Name of array to be loaded
	File	File in which the array is defined

Note

If an array in the current configuration file must be replaced by an array from another configuration file, then both arrays must have the same name.

Example

```

                                         ; Extract from file maske.com
DEF VAR2 = (S/*ARR5/"Out"/,"Toggle field")
PRESS(HS5)
    LA("ARR5","arrayext.com")           ; Load array ARR5 from file arrayext.com
    VAR2 = ARR5[0]                       ; "Above"/"Below"/"Right"/"Left" appears
                                         in the VAR2 toggle field
                                         instead of "Out/In"

END_PRESS
//A(ARR5)
("Out"/"In")
//END

                                         ; Extract from file arrayext.com
//A(ARR5)
("Above"/"Below"/"Right"/"Left")
//END
```

Note

Please note that a valid value must be assigned to a variable after the LA function has been used to assign another array to the toggle field of the variable.

6.3.22 Load Block (LB)

Description

The LB (Load Block) function can be used to load blocks containing subprograms during runtime. LB should be configured in a LOAD method so that the loaded subprograms can be called at any time.

Note

Subprograms can also be defined directly in a dialog so that they do not have to be loaded.

Programming

Syntax:	LB ("Block name"[,"File"])	
Description:	Loads subprogram during runtime	
Parameters:	Block name	Name of block identifier
	File	Path name of configuration file Default setting = Current configuration file

Example

```

LOAD
  LB("PROG1")           ; Block "PROG1" is searched for in the current configura-
                        ; tion file and then loaded.
  LB("PROG2","XY.COM")  ; Block "PROG2" is searched for in the configuration file
                        ; XY.COM and then loaded.
END_LOAD

```


6.3.23 Load Mask (LM)

Description

The LM function loads a new dialog. Depending on the mode of the dialog change (see table below), a message dialog can also be implemented with this function.

Master dialog / sub-dialog

A dialog, which calls another dialog, but is not ended itself, is referred to as a master dialog. A dialog that is called by a master dialog is referred to as a sub-dialog.

Programming

Syntax:	LM ("Identifier"[,"File"][,MSx[,VARx]])	
Description:	Loads a dialog	
Parameter:	Identifier	Name of the dialog to be loaded
	File	Path name (HMI file system or NC file system) of the configuration file; default setting: Current configuration file
	MSx	Mode of dialog change
	0:	(Default setting) The current dialog is rejected, the new dialog is loaded and displayed. EXIT will send you back to the standard application. You can use the MSx parameter to determine whether or not the current dialog should be terminated when changing dialogs. If the current dialog is retained, variables can be transferred to the new dialog. The advantage of the MSx parameter is that the dialogs do not always need to be reinitialized when they are changed. Instead, the data and layout of the current dialog are retained and data transfer is made easier.
	1:	The current master dialog is interrupted when the LM function is initiated. The new sub-dialog is loaded and displayed (e.g. for the implementation of a message dialog). EXIT will end the sub-dialog and return to the point at which the master dialog was interrupted. In the master dialog, the UNLOAD block is not processed during the interruption.
	VARx	Precondition: MS1 List of variables which can be transferred from the master dialog to the sub-dialog. Up to 20 variables, each separated by a comma, can be transferred.

Note

Parameter VARx transfers only the value of the variable in each case, i.e. variables can be read and written in the sub-dialog, but are not visible in it. Variables can be returned from the sub-dialog to the master dialog by means of the EXIT function.

Example

```
SCREEN FORM1 (master dialog): Jump to the sub-dialog SCREEN FORM2 with variable
transfer
PRESS (HS1)

LM("SCREEN FORM2","CFI.COM",1, POSX, POSY, DIAMETER)
                                ; Interrupt screen form1 and open screen form2:
                                In doing this, variables POSX, POSY and DIAME-
                                TER are transferred.

DLGL("Screen form2 ended")      ; After returning from screen form2, the fol-
                                lowing text appears in the dialog line of
                                screen form1: Screen form2 ended.

END_PRESS

SCREEN FORM2 (sub-dialog): Return to master dialog SCREEN FORM1 with transfer of the
variable contents
PRESS (VS8)

EXIT(POSX, POSY, DIAMETER)
                                ; Hide screen form2 and continue screen form1:
                                In doing this, the changed variables POSX, POSY
                                and DIAMETER are transferred.

END_PRESS
```

6.3.24 Load Softkey (LS)

Description

The LS function can be used to display another softkey menu.

Programming

Syntax:	LS("Identifier"[, "File"][, Merge])
Description:	Displays the softkey menu

Parameters:	Identifier	Name of the softkey menu
	File	Path (HMI file system or NC file system) to the configuration file Default setting: Current configuration file
	Merge	
	0:	All existing softkeys are deleted; the newly configured softkeys are entered.
	1:	Default setting Only the newly configured softkeys overwrite the available softkeys. The other softkeys (= softkeys of the HMI application) are kept with their functionality and text.

Example

```

PRESS (HS4)
  LS ("Menu2",,0)      ; Menu2 overwrites the existing softkey menu, the softkeys
                        ; that are displayed are deleted.
END_PRESS

```

Note

As long as the Interpreter has not displayed a dialog, i.e. no LM function has yet been processed, only one LS or one LM command, but no other action, can be configured in the PRESS method of the definition block for the start softkey and the softkey menu.

The LS and LM functions may only be called within a softkey PRESS method and will not react if navigation keys are pressed (PU, PD, SL, SR, SU, SD).

6.3.25 Load Grid (LG)

Description

The table description (grid) can be provided dynamically within the LOAD method using the LG method.

In order to assign a table using the LG method, the variable must have already been defined as a grid variable and cross-referenced to an existing, valid table.

Programming

Syntax:	LG (<i>Grid name, Variable name [,File name]</i>)
Description:	Loads a table

Parameters:	Grid name	Name of the table (grid) in inverted commas
	Variable name	Name of the variable to which the table is to be assigned, in inverted commas
	File name	Name of the file in which the table (grid) is defined, in inverted commas. Only needs to be specified if the table is not defined within the file that also contains the definition of the variable.

Example

```
//M(MyGridSample/"My Grid Sample")
DEF MyGridVar=(R/% MyGrid1//WR2///100,,351,100)
LOAD
    LG("MyGrid1","MyGridVar","mygrids.com")
END_LOAD
```

Content of mygrids.com:

```
//G(MyGrid1/0/5)
(I///,"MyGrid1"/wr1/"1"/80/1)
(R3///"LongText1","R1-R4"/wr2/"$R[1]"/80/1)
(IBM///"LongText2","M2.2-M2.5"/wr2/"M2.2"/80/1)
(R3///"LongText3","R9,R11,R13,R15"/wr2/"$R[9]"/110/2)
//END
```

6.3.26 Multiple selection SWITCH

Description

You can check a variable for different values with a SWITCH command.

The expression configured in the SWITCH statement is compared in succession to all the values configured in the following CASE statements.

If not the same, a comparison is made with the following CASE statement. If a DEFAULT statement follows and up to now no CASE statement was the same as the expression configured in the SWITCH statement, the statements following the DEFAULT statement are executed.

If the same, the following statements are executed until a CASE, DEFAULT or END_SWITCH statement occurs.

Example

```
FOCUS
  SWITCH (FOC)
    CASE "VarF"
      DLGL("Variable ""VarF"" has the input focus.")
    CASE "VarZ"
      DLGL("Variable ""VarZ"" has the input focus.")
    DEFAULT
      DLGL("Any other variable has the input focus.")
  END_SWITCH
END_FOCUS
```

6.3.27 Multiple Read NC PLC (MRNP)

Description

This MRNP command transfers several NC/PLC variables in a single register access. This access method is significantly faster than reading via individual access attempts. The NC/PLC variables must be included within an MRNP command of the same area.

The areas of the NC/PLC variables are organized as follows:

- General NC data (\$MN..., \$SN..., /nck/...)
- Channel-specific NC data (\$MC..., \$SC..., /channel/...)
- PLC data (DB..., MB..., /plc/...)
- Axis-specific NC data on the same axis (\$MA..., \$SA..)

Programming

Syntax:	MRNP (<i>Variable name 1*Variable name 2[* ...], Register index</i>)
Description:	Reads several variables
Parameter:	In the variable names, ""*"" is the separator. The values are transferred to register REG[Register index] and those following in the order that the variable names appear in the command. The following therefore applies: The value of the first variable is located in REG[Register index]. The value of the second variable is located in REG[Register index + 1], etc.

Note

It should be noted that the number of registers is restricted and the list of variables cannot exceed 500 characters.

Example

```
MRNP("$R[0]*$R[1]*$R[2]*$R[3]",1) ;The values of variables $R[0] to $R[3] are
                                   written to REG[1] to REG[4].
```

NC variable

All machine data, setting data and R parameters are available, but only certain system variables (see also: List of accessible system variables (Page 204)).

All global and channel-specific user variables (GUDs) can be accessed. However, local and program-global user variables cannot be processed.

Machine data	
Global machine data	\$MN_...
Axis-specific machine data	\$MA_...
Channel-specific machine data	\$MC_...

Setting data	
Global setting data	\$SN_...
Axis-specific setting data	\$SA_...
Channel-specific setting data	\$SC_...

System variables	
R parameter 1	\$R[1]

PLC variable

All PLC data are available.

PLC data	
Byte y bit z of data block x	DBx.DBXy.z
Byte y of data block x	DBx.DBBy
Word y of data block x	DBx.DBWy
Double word y v. of data block x	DBx.DB Dy
Real y of data block x	DBx.DBRy
Flag byte x bit y	Mx.y

PLC data	
Flag byte x	MBx
Flag word x	MWx
Flag double word x	MDx
Input byte x bit y	Ix.y or Ex.y
Input byte x	IBx or EBx
Input word x	IWx or EWx
Input double word x	IDx or EDx
Output byte x bit y	Qx.y or Ax.y
Output byte x	QBx or ABx
Output word x	QWx or AWx
Output double word x	QDx or ADx
String y with length z from data block x	DBx.DBSy.z

6.3.28 PI services

Description

The PI_START function starts PI Services (Program Invocation Services) from the PLC in the NC area.

Note

A list of the available PI services can be found in the Basic Functions Function Manual.

Programming

Syntax:	PI_START ("Transfer string")	
Description:	Executes the PI service	
Parameter:	"Transfer string"	Unlike the OEM documentation, the transfer string should be entered in inverted commas.

Example

```
PI_START("/NC,001,_N_LOGOUT")
```

Note

Channel-dependent PI Services always refer to the current channel.

PI services of the tool functions (TO area) always refer to the TO area that is assigned to the current channel.

6.3.29 Reading (RNP) and writing (WNP) system or user variables

Description

The RNP (Read NC PLC) command can be used to read NC or PLC variables or machine data.

Programming

Syntax:	RNP ("System or user variable", value)	
Description:	Reads NC or PLC variable or machine data	
Parameters:	System or user variable	Name of NC or PLC variable
	Value	Value that is to be written to the system or user variable. If the value is a String type, it must be written in double quotation marks.

Example

```
VAR2=RNP("$AA_IN[2]") ; Read NC variable
```

Description

The WNP (Write NC PLC) command can be used to write NC or PLC variables or machine data.

NC/PLC variables are accessed anew every time the WNP function is executed, i.e., NC/PLC access is always executed in a CHANGE method. It is advisable to use this option in cases where a system or user variable changes value frequently. If an NC/PLC variable is to be accessed only once, then it must be configured in a LOAD or UNLOAD method.

Programming

Syntax:	WNP ("System or user variable", value)	
Description:	Writes NC or PLC variable or machine data	
Parameters:	System or user variable	Name of NC or PLC variable
	Value	Value that is to be written to the system or user variable. If the value is a String type, it must be written in double quotation marks.

Example

```
WNP("DB20.DBB1",1) ; Write PLC variable
```


6.3.30 RESIZE_VAR_IO and RESIZE_VAR_TXT

Description

The `RESIZE_VAR_IO()` and `RESIZE_VAR_TXT()` commands changes the geometry of the input/output component or text component of a variable.

After setting the new geometry, all fields of the screen are positioned as if the screen had been configured with these positions from the start. In this way, all fields are correctly aligned, depending on the configuration.

Programming

Syntax:	RESIZE_VAR_IO (<i>Variable name</i> , [<i>X</i>], [<i>Y</i>], [<i>Width</i>], [<i>Height</i>]) RESIZE_VAR_TXT (<i>Variable name</i> , [<i>X</i>], [<i>Y</i>], [<i>Width</i>], [<i>Height</i>])	
Description:	Changes the geometry of the input/output component or text component of a variable.	
Parameter:	Variable name	Name of the variable whose geometry of the input/output component or text component of a variable is to be changed.
	X	X-coordinate, top left
	Y	Y-coordinate, top left
	Width	Width
	Height	Height

Note

The previous value is retained for each value of X, Y, Width or Height that is not specified. If -1 is specified for one of these values, "Run MyScreens" sets the specified component of the default position.

Example

```
RESIZE_VAR_IO("MyVar1", 200, , 100) ; The I/O component of the "MyVar1" variable
                                     is moved to the X position 200 pixels, the
                                     width is set to 100 pixels. The Y position and
                                     the height of the previous value are retained.
```

6.3.31 Register (REG)

Register description

Registers are needed in order to exchange data between different dialogs. Registers are assigned to each dialog. These are created when the first dialog is loaded and assigned the value 0 or an empty string.

Note

Registers must not be used directly in an OUTPUT method for generating NC code.

Programming

Syntax:	REG [<i>x</i>]	
Description:	Defines a register	
Parameters:	x	Register index with x = 0...19; Type: REAL or STRING = VARIANT Registers with x ≥ 20 have already been assigned by Siemens.

Description of register value

The assignment of values to registers is configured in a method.

Note

If a new dialog is generated from an existing dialog by means of the LM function, register content is automatically transferred to the new dialog at the same time and is available for further calculations in the second dialog.

Programming

Syntax:	<i>Identifier.val = Register value</i> or <i>Identifier = Register value</i>	
Description:		
Parameters:	Identifier	Name of the register
	Register value	Value of register

Example

```
UNLOAD
  REG[0] = VAR1          ; Assign value of variable 1 to register 0
```

```

END_UNLOAD

UNLOAD
    REG[9].VAL = 84          ; Assign value 84 to register 9
END_UNLOAD

                                ; These registers can then be assigned to local variables
                                ; again in a method in the next dialog.

LOAD
    VAR2 = REG[0]
END_LOAD

```

Description of register status

The Status property can be used to scan a register for valid content.

One possible use for the register scan function is to ensure that a value is written to a register only if the relevant dialog is a "master dialog".

Programming

Syntax:	<i>Identifier.vld</i>	
Description:	The property is read-only property.	
Parameters:	Identifier	Name of the register
Return value:		The result of the scan can be:
	FALSE =	invalid value
	TRUE =	valid value

Example

```

IF REG[15].VLD == FALSE      ; Scan validity of register value
    REG[15] = 84
ENDIF

VAR1 = REG[9].VLD            ; Assign the value of the REG[9] status request to Var1.

```

6.3.32 RETURN

Description

The RETURN function can be used to prematurely terminate execution of the current subprogram and to return to the branch point of the last CALL command.

If no RETURN command is configured in the subprogram, the subprogram will run to the end before returning to the branch point.

Programming

Syntax:	RETURN	
Description:	Returns to the branch point	
Parameters:	- None -	

Example

```
//B (PROG1)          ; Block start
SUB (UP2)            ; Start of subprogram
  IF VAR1.val=="Otto"
    VAR1.val="Hans"
    RETURN           ; If the variable value = Otto, the value "Hans" is as-
                    ; signed to the variable, and the subprogram ends at this
                    ; point.
  ENDIF
  VAR1.val="Otto"     ; If the variable value ≠ Otto, the value "Otto" is as-
                    ; signed to the variable.
END_SUB              ; End of subprogram
//END                ; Block end
```

6.3.33 Recompile

Description

In the programming support system, it is possible to **recompile** NC code that has been generated with the GC function and to display the variable values in the input/output field of the associated entry dialog again.

Programming

Variables from the NC code are transferred to the dialog. At the same time, the variable values from the NC code are compared with the calculated variable values from the configuration file. If the values do not coincide, an error message is written to the log book because values have been changed during NC code generation.

If the NC code contains the same variable several times, it is evaluated at the point where it last occurs during recompilation. A warning is also written to the log book.

Variables not utilized in NC code during code generation are stored as user comment. The term "user comment" refers to all information required to recompile codes. User comment must not be altered.

Note

The block consisting of NC code and user comment can be recompiled only if it starts at the beginning of a line.

Examples:

The programm contains the following NC code:

```
DEF VAR1=(I//101)
OUTPUT (CODE1)
  "X" VAR1 " Y200"
  "X" VAR1 " Y0"
END_OUTPUT
```

The following code is then stored in the parts program:

```
;NCG#TestGC#\cus.dir\aeditor.com#CODE1#1#3#
X101 Y200
X101 Y0
;#END#
```

The Editor reads the following during recompilation:

```
X101 Y200
X222 Y0           ; The value for X has been changed in the part program (X101 → X222)
```

The following value is displayed for VAR1 in the input dialog: VAR1 = 222

See also

Generate code (GC) (Page 129)

6.3.34 Recompile without user comment

Description

In the Programming Support, it is possible to **recompile without user comments** the NC code that has been generated with the GC function and to display the variable values in the input/output field of the associated entry dialog again.

Programming

The GC command can be executed in the following way in order to suppress comment lines that are generated for standard code generation:

```
GC ("CODE1", D_NAME, 1)
```

Normally, the resulting code cannot be recompiled. The following steps are required in order to be able to recompile the cycle calls generated in this way:

- **Expanding the easyscreen.ini**

Section [RECOMPILE_INFO_FILES] will be introduced into the easyscreen.ini file. In this section, all ini files are listed that contain descriptions for cycles recompiled without user comment:

```
[RECOMPILE_INFO_FILES]
IniFile01 = cycles1.ini
IniFile02 = cycles2.ini
```

Several ini files can be specified, whose names can be freely selected.

- **Creating an ini file for a cycle description**

Save the ini file with the cycle descriptions under the following path:

[System user directory]/cfg

[System oem directory]/cfg

[System add_on directory]/cfg

A separate section is required for each cycle. The section name corresponds to the name of the cycle:

```
[Cycle123]
Mname = TestGC
Dname = testgc.com
OUTPUT = Code1
Anzp = 3
Version = 0
Code_type = 1
Icon = cycle123.png
Desc_Text = This is describing text
```

Mname	Screen form name
Dname	Name of the file in which the screen is defined
OUTPUT	Name of the relevant OUTPUT method
Anzp	Number of parameters of the screen to be recompiled (all with DEF-created variables, also help variables)
Version	(optional) version specification for cycle

Icon	(optional) icon for display in the machining step program, format *.png Screen size for corresponding resolution: 640 X 480 mm → 16 x 16 pixels 800 X 600 mm → 20 x 20 pixels 1024 X 768 mm → 26 x 26 pixels 1280 X 1024 mm → 26 x 26 pixels 1280 X 768 mm → 26 x 26 pixels File loc.: <i>[System user directory]\ico\ico<Resolution></i> Note: For resolutions of 1280, the folder for 1024 x 768 mm is used (only suitable for machining step programs).
Desc_Text	(optional) Explanation text for display in the machining step program, max. length of string is 17 characters (only suitable for machining step programs).

Example

```
//M(TestGC/"Code generation:")
DEF VAR1 = (R//1)
DEF VAR2 = (R//2)
DEF D_NAME
LOAD
    VAR1 = 123
    VAR2 = -6
END_LOAD
OUTPUT(CODE1)
    "Cycle123(" VAR1 "," VAR2 ")"
    "M30"
END_OUTPUT

PRESS(VS1)
    D_NAME = "\MPF.DIR\MESSEN.MPF"
    GC("CODE1",D_NAME)                                ;Write NC code from the OUTPUT method to
                                                         file \MPF.DIR\MESSEN.MPF:
                                                         Cycle123(123, -6)
                                                         M30
END_PRESS
```

Supplementary conditions

- The "Recompile without user comment" functionality does not have the full range of functions as "Recompile with user comment".
Typical cycle calls, such as MYCYCLE(PAR1, PAR2, PAR3, ...), are supported during "Recompile without user comment". However, there must not be a user comment in the line of the function call. Optional parameters that are not transferred during the function and are of the string S type, must however be at least specified with empty quotation marks, e.g. "". Otherwise, "Run MyScreens" tries to fill in these parameters with commas and then recompiles the "filled-in cycle call".
- Parameters of the string type must not have any commas or semicolons in the string to be transferred.
- During "Recompile without user comment", all variables contained in the OUTPUT method must always be within the brackets so that the "Fill in missing cycle parameters with commas" functionality can take effect.

Example:

Permitted:

```
OUTPUT
    "MYCYCLE (" MYPAR1 " , " MYPAR2 " , " MYPAR3 " ) "
END_OUTPUT
```

Not permitted (the MYCOMMENT variable is positioned after the closing bracket):

```
OUTPUT
    "MYCYCLE (" MYPAR1 " , " MYPAR2 " , " MYPAR3 " ) " MYCOMMENT
END_OUTPUT
```

6.3.35 Search Forward, Search Backward (SF, SB)

Description

The **SF, SB (Search Forward, Search Backward)** function is used to search for a string from the current cursor position in the NC program currently selected in the Editor and to output its value.

Programming

Syntax:	SF ("String")
Identifiers:	Search Forward : Search forward from the current cursor position
Syntax:	SB ("String")

Identifiers:	Search Backward: Search backward from the current cursor position	
Parameters:	String	Text to be found

Rules governing text search

- A blank must be inserted before and after the search concept unit, consisting of search string and its value, in the currently selected NC program.
- The system does not search for concepts within comment text or other strings.
- The value to be output must be a numerical expression. Expressions in the form of "X1=4+5" are not recognized.
- The system recognizes hexadecimal constants in the form of X1='HFFFF', binary constants in the form of X1='B10010' and exponential components in the form of X1='-.5EX-4'.
- The value of a string can be output if it contains the following between string and value:
 - Nothing
 - Blanks
 - Equality sign

Example

The following notations are possible:

X100 Y200	;	The variable Abc is assigned the value 200
Abc = SB("Y")		
X100 Y 200	;	The variable Abc is assigned the value 200
Abc = SB("Y")		
X100 Y=200	;	The variable Abc is assigned the value 200
Abc = SB("Y")		

6.3.36 STRING functions

Overview

The following functions enable strings to be processed:

- Determine length of string
- Find a character in a string
- Extract substring from left
- Extract substring from right
- Extract substring from mid-string
- Replace substring

6.3 Functions

- Compare strings
- Insert a string in another string
- Remove a string from a string
- Remove blanks (from the left or from the right)
- Insert values or strings with formatting identifiers

LEN function: Length of a string

Syntax:	LEN (<i>string / varname</i>)	
Description:	Determines the number of characters in a string	
Parameters:	string	Every valid string expression. NULL is output if string is empty.
	varname	Any valid declared variable name
	Only one of the two parameters is allowed.	

Example

```

DEF VAR01
DEF VAR02

LOAD
  VAR01="HELLO"
  VAR02=LEN (VAR01)           ; Result = 5
END_LOAD

```

INSTR function: Search for character in string

Syntax:	INSTR (<i>Start, String1, String2 [,Direction]</i>)	
Description:	Searches for characters	
Parameters:	Start	Starting position for searching for string1 in string2. Enter 0 to start searching at the beginning of string2.
	String1	Character that is being searched for.
	String2	Character string in which the search is being made.
	Direction (optional)	Direction in which the search is being made. 0: From left to right (default setting) 1: From right to left
	0 is returned if string1 does not occur in string2.	

Example

```

DEF VAR01
DEF VAR02

LOAD
  VAR01="HELLO/WORLD"
  VAR02=INST(1,"/",VAR01)          ; Result = 6
END_LOAD

```

LEFT Function: String from left

Syntax:	LEFT (<i>string, length</i>)	
Description:	LEFT returns a string containing the specified number of characters starting from the left-hand side of a string.	
Parameters:	string	Character string or variable with the character string to be processed
	length	Number of characters that are to be read out

Example

```

DEF VAR01
DEF VAR02

LOAD
  VAR01="HELLO/WORLD"
  VAR02=LEFT(VAR01,5)              ; Result = "HELLO"
END_LOAD

```

RIGHT function: String from right

Syntax:	RIGHT (<i>string, length</i>)	
Description:	RIGHT returns a string containing the specified number of characters starting from the right-hand side of a string.	
Parameters:	string	Character string or variable with the character string to be processed
	length	Number of characters that are to be read out

Example

```

DEF VAR01
DEF VAR02

```

```

LOAD
  VAR01="HELLO/WORLD"
  VAR02=LEFT (VAR01,4)           ; Result = "WORLD"
END_LOAD

```

MIDS function: String from mid-string

Syntax:	MIDS (<i>string, start [, length]</i>)	
Description:	MIDS returns a string containing the specified number of characters starting at the specified position in the string.	
Parameters:	string	Character string or variable with the character string to be processed
	start	Start from where characters are to be read in the string
	length	Number of characters that are to be read out

Example

```

DEF VAR01
DEF VAR02
LOAD
  VAR01="HELLO/WORLD"
  VAR02=LEFT (VAR01,4,4)           ; Result = "LO/W"
END_LOAD

```

REPLACE Function: Replacing characters

Syntax:	REPLACE (<i>string, FindString, ReplaceString [, start [, count]]</i>)	
Description:	REPLACE replaces a character/string in a string with another character/string.	
Parameters:	string	String in which FindString is to be replaced with ReplaceString.
	FindString	String to be replaced
	ReplaceString	Replacement string (is used instead of the FindString)
	start	Starting position for search and replace operations
	count	Number of characters that are to be searched from the starting position after FindString
Return value:		
	string = Empty string	Copy of string
	FindString = Empty string	Copy of string
	ReplaceString = Empty string	Copy of string, in which all occurrences of FindString are deleted
	start > Len(String)	Empty string
	count = 0	Copy of string

STRCMP function: Compare strings

Syntax:	STRCMP (<i>string1</i> , <i>string2</i> [, <i>CasInsensitive</i>])	
Description:	STRCMP compares a character string with another character string.	
Parameters:	string1	Character string that is to be compared with a second character string
	string2	Character string that is to be compared with the first character string
	CasInsensitive	Comparison with/without consideration of upper/lower case: Not specified or FALSE = upper/lower case is considered TRUE = upper/lower case is not considered

Example

```

REG[0]=STRCMP("Hugo", "HUGO")           ; Result = 32
                                         ; (<> 0, strings are not identical)

REG[0]=STRCMP("Hugo", "HUGO", TRUE)      ; Result = 0
                                         ; (== 0, strings are identical)

```

STRINSERT function: Insert string

Syntax:	STRINSERT (<i>string1</i> , <i>string2</i> , <i>insert position</i>)	
Description:	STRINSERT inserts a character string at a specific position in a character string.	
Parameters:	string1	Character string in which a character string is to be inserted
	string2	Character string that is to be inserted
	insert position	Position at which the character string is to be inserted

Example

```

REG[0]=STRINSERT("Hello!", " world", 5) ; Result = "Hello world!"

```

STRREMOVE function: Remove string

Syntax:	STRREMOVE (<i>string1</i> , <i>remove position</i> , <i>count</i>)	
Description:	STRREMOVE removes a specific number of characters at a specific position within a character string.	

Parameters:	string1	Character string from which characters are to be removed
	remove position	Position at which characters are to be removed from the character string
	count	Number of characters to be removed

Example

```
REG[0]=STRREMOVE("Hello world!", 5, 6) ; Result = "Hello!"
```

TRIMLEFT function: Remove blanks from the left of the string

Syntax:	TRIMLEFT (<i>string1</i>)	
Description:	TRIMLEFT removes blanks on the left from a character string.	
Parameters:	string1	Character string from which blanks are to be removed from the left of the character string

Example

```
REG[0]=TRIMLEFT(" Hello!") ; Result = "Hello!"
```

TRIMRIGHT function: Remove blanks from the right of the string

Syntax:	TRIMRIGHT (<i>string1</i>)	
Description:	TRIMRIGHT removes blanks on the right from a character string.	
Parameters:	string1	Character string from which blanks are to be removed from the right of the character string

Example

```
REG[0]=TRIMRIGHT("Hello! ") ; Result = "Hello!"
```

FORMAT function: Insert values or string with formatting identifier

Syntax:	FORMAT (<i>Text with formatting identifier</i> [, <i>Value1</i>] ... [, <i>Value28</i>])	
Description:	Through the use of formatting identifiers, the FORMAT function enables up to 28 values or strings to be inserted at specified positions in a specific text.	

Formatting identifiers:	Syntax	%[Flags] [Width] [.decimal places] type
	Flags	Optional character for defining output formatting: • Right-justified or left-justified (" ") for left-justified • Add leading zeros ("0") • default: Fill in with blanks if the value to be output has fewer positions than specified with [Width].
	Width	The argument defines the minimum output width for a non-negative number. If the value has fewer positions than specified by the argument, the missing spaces are filled with blanks.
	Decimal places	With floating-point numbers, the optional parameter defines the number of decimal places.
	Type	The type character defines which data formats are transferred to the Print statement. These characters need to be specified. • d: Integer value • f: Floating-point number • s: String • o: Octal • x: Hexadecimal • b: Binary

Example

```

DEF VAR1
DEF VAR2
LOAD
VAR1 = 123
VAR2 = FORMAT("Hello %08b %.2f %s!", VAR1 + 1, 987.654321, "world")
; Result = "Hello 01111100 987.65 world!"
END_LOAD

```

See also

Use of strings (Page 80)

6.3.37 WHILE/UNTIL loops

Description

A loop can be implemented with the DO LOOP commands. Depending on the configuration, it is run through as long as a condition is satisfied (WHILE) or until a condition occurs (UNTIL).

6.3 Functions

As loops can impair the system performance, depending on the configuration, they should be used carefully and without time-intensive actions in the loops.

It is recommended that a register (REG[]) be used as run variable, because normal display variables (particularly those with system or user variables connection) also impair the system performance due to the frequent updates or write operations.

The runtime of "Run MyScreens" methods can be determined with the DEBUG function (see Section DEBUG (Page 121)). Problems caused by loops (high CPU load, reduced response capability) may be identified in this way.

Note

As each FOR loop can be replaced by a WHILE loop, the syntax to formulate a FOR loop is not supported in EasyScreen.

Programming

```
DO
    <instructions>
LOOP_WHILE <Condition to continue the loop>

DO
    <instructions>
LOOP_UNTIL <Condition to terminate the loop>

DO_WHILE <Condition to continue the loop>
    <instructions>
LOOP

DO_UNTIL <Condition to terminate the loop>
    <instructions>
LOOP
```

Example

```
REG[0] = 5
DO
    DEBUG("OUTER: " << REG[0])
    REG[0] = REG[0] + 1
    REG[1] = -5

    DO
        DEBUG("INNER: " << REG[1])
```



```
        REG[1] = REG[1] + 1
    LOOP_WHILE REG[1] < 0

LOOP_WHILE REG[0] < 10
```

```
    REG[0] = 5
DO
    DEBUG("OUTER: " << REG[0])
    REG[0] = REG[0] + 1
    REG[1] = -5

    DO
        DEBUG("INNER: " << REG[1])
        REG[1] = REG[1] + 1
        LOOP_UNTIL 0 <= REG[1]

LOOP_WHILE 10 > REG[0]
```

```
    REG[0] = 5
DO_WHILE 10 > REG[0]
    DEBUG("OUTER: " << REG[0])
    REG[0] = REG[0] + 1
    REG[1] = -5

    DO_UNTIL 0 <= REG[1]
        DEBUG("INNER: " << REG[1])
        REG[1] = REG[1] + 1
    LOOP

LOOP
```

```
    REG[0] = 5
DO_WHILE 10 > REG[0]
    DEBUG("OUTER: " << REG[0])
    REG[0] = REG[0] + 1
    REG[1] = -5

    DO
        DEBUG("INNER: " << REG[1])
```

6.3 Functions

```

    REG[1] = REG[1] + 1
    LOOP_UNTIL 0 <= REG[1]

LOOP

```

6.3.38 Cyclic execution of scripts: START_TIMER, STOP_TIMER

Description

SUB methods can be called cyclically with the aid of timers. The START_TIMER() and STOP_TIMER() functions are available for this purpose.

Note

Only one timer can be configured for each SUB method.

Programming

Syntax:	START_TIMER ("SUB name", Interval)	
Description:	Starts cyclic processing of a SUB method	
Parameters:	SUB name:	Name of the SUB method to be called cyclically
	Interval:	Interval in milliseconds

Syntax:	STOP_TIMER ("SUB name", Interval)	
Description:	Stops cyclic processing of a SUB method	
Parameters:	SUB name:	Name of the SUB method whose timer is to be stopped.

Example

```

//M(TimerSample/"My timer")
DEF MyVariable=(I//0/,"Number of cyclic calls:"/WR1)
VS1=("Start%ntimer")
VS2=("Stop%ntimer")

SUB(MyTimerSub)
    MyVariable = MyVariable + 1
END_SUB

```

```
//M(TimerSample/"My timer")  
PRESS(VS1)  
    ;Calls SUB "MyTimerSub" every 1000 milliseconds  
    START_TIMER("MyTimerSub", 1000)  
END_PRESS  
  
PRESS(VS2)  
    STOP_TIMER("MyTimerSub")  
END_PRESS
```

If START_TIMER is called again for a timer already assigned to a SUB method, then the new interval is taken over if it is different. Otherwise, the second call is ignored.

The smallest interval for the system is 100 milliseconds.

If STOP_TIMER is called for a SUB method for which no timer is currently running, the call is ignored.

Graphic and logic elements

7.1 Line, dividing line, rectangle, circle and ellipse

Description

Lines, dividing lines, rectangles and ellipses/circles are configured in the LOAD method:

- Transparent rectangles are created by setting the fill color to the system background color.
- With the ELLIPSE element, a circle is created by setting the height and the width to the same value.
- Horizontal and vertical dividing lines always have the exact window width or window height. No scrollbars are produced by dividing lines.

If you previously used the RECT element to display dividing lines, e.g.

RECT(305,0,1,370,0,0,1), this is detected by the Programming Support and automatically converted to V_SEPARATOR(305,1,"#87a5cd", 1). This applies to both vertical and horizontal dividing lines.

LINE element

Programming:

Syntax:	LINE (x1,y1,x2,y2,f,s)	
Description:	Defines a line	
Parameter:	x1	Start point x-coordinate
	y1	Start point y-coordinate
	x2	End point x-coordinate
	y2	End point y-coordinate
	f	Color of the line
	s	Line style: 1 = solid 2 = dashed 3 = dotted 4 = dashed and dotted

RECT element

Programming:

Syntax:	RECT (x,y,w,h,f1,f2,s)
Description:	Defines a rectangle

Parameter:	x	x-coordinate, top left
	y	y-coordinate, top left
	w	Width
	h	Height
	f1	Color of the border
	f2	Fill color
	s	Border style: 1 = solid 2 = dashed 3 = dotted 4 = dashed and dotted

ELLIPSE element

Programming:

Syntax:	ELLIPSE(x,y,w,h,f1,f2,s)	
Description:	Defines an ellipse or circle	
Parameter:	x	x-coordinate, top left
	y	y-coordinate, top left
	w	Width
	h	Height
	f1	Color of the border
	f2	Fill color
	s	Border style: 1 = solid 2 = dashed 3 = dotted 4 = dashed and dotted

A circle is created when the values for height and width are the same.

V_SEPARATOR element

Programming:

Syntax:	V_SEPARATOR (x,w,color,pen)	
Description:	Defines a vertical dividing line	
Parameter:	x	x position
	w	Line weight
	color	Color
	pen	Line style: 1 = solid 2 = dashed 3 = dotted 4 = dashed and dotted

H_SEPARATOR element

Programming:

Syntax:	H_SEPARATOR (y,h,color,pen)	
Description:	Defines a horizontal dividing line	
Parameter:	y	y position
	h	Line weight
	color	Color
	pen	Line style: 1 = solid 2 = dashed 3 = dotted 4 = dashed and dotted

See also

LOAD (Page 100)

See also

CLEAR_BACKGROUND (Page 113)

7.2 Defining an array**Definition**

An array can be used to organize data of the same data type stored in the memory in such a way that it is possible to access the data via an index.

Description

Arrays can be one- or two-dimensional. A one-dimensional array is treated like a two-dimensional array with just one line or column.

Arrays have start identifier //A and end identifier //END. The number of lines and columns is optional. An array is structured in the following way:

Programming

Syntax:	//A(Identifier) (a/b...) (c/d...) ... //END	
Description:	Defines array	
Parameters:	Identifier	Name of array
	a, b, c, d	Values of array Values of the STRING type must be enclosed in double quotation marks.

Example

```
//A(Thread)                                ; Size/lead/core diameter
(0.3 / 0.075 / 0.202)
(0.4 / 0.1   / 0.270)
(0.5 / 0.125 / 0.338)
(0.6 / 0.15  / 0.406)
(0.8 / 0.2   / 0.540)
(1.0 / 0.25  / 0.676)
(1.2 / 0.25  / 0.676)
(1.4 / 0.3   / 1.010)
(1.7 / 0.35  / 1.246)
//END
```

7.2.1 Accessing the value of an array element

Description

The value of an array access operation can be transferred with property Value (identifier.val).

The line index (line number of the array) and the column index (column number of the array) each begin at 0. If a line index or column index is outside the array, the value 0 or a blank string is output and the ERR variable is set to TRUE. Variable ERR is also set to TRUE if a search concept cannot be found.

Programming

Syntax:	Identifier [Z,[M,C]].val or Identifier [Z,[M,C]]
Description:	Access to one-dimensional array with only one column

Syntax:	Identifier [S,[M[,C]]].val or Identifier [S,[M[,C]]] or	
Description:	Access to one-dimensional array with only one line	
Syntax:	Identifier [Z,S,[M[,C]]].val or Identifier [Z,S,[M[,C]]]	
Description:	Access to two-dimensional array	
Parameters:	Identifier:	Name of array
	Z:	Line value (line index or search concept)
	S:	Column value (column index or search concept)
	M:	Access mode
		0 Direct
		1 Searches the line, column directly
		2 Searches the column, line directly
		3 Searches
		4 Searches line index
		5 Searches column index
	C:	Compare mode
		0 Search concept must be located in the range of values of the line or column.
		1 Search concept must be located exactly.
Example:	<pre>VAR1 = MET_G[REG[3],1,0].VAL ;Assign Var1 a value from array MET_G</pre>	

Access mode

- **"Direct" access mode**
With "Direct" access mode (M = 0), the array is accessed with the line index in Z and the column index in S. Compare mode C is not evaluated.
- **"Search" access mode**
In the case of access mode M = 1, 2 or 3, the search always commences in line 0 or column 0.

Mode M	Line value Z	Column value S	Output value
0	Line index	Column index	Value from line Z and column S
1	Search concept: Search in column 0	Column index of column from which value is read	Value from line found and column S
2	Line index of line from which return value is read	Search concept: Search in line 0	Value from line Z and column found
3	Search concept: Search in column 0	Search concept: Search in line 0	Value from line and column found

Mode M	Line value Z	Column value S	Output value
4	Search concept: Search in column S	Column index of search column	Line index
5	Line index of search line.	Search concept: Search in line Z	Column index

Compare mode

When compare mode C = 0 is used, the content of the search line or search column must be sorted in ascending order. If the search concept is smaller than the first element or greater than the last, the value 0 or a blank string is output and the error variable ERR is set to TRUE.

When compare mode C = 1 is used, the search concept must be present in the search line or search column. If the search concept cannot be found, the value 0 or a blank string is output and the error variable ERR is set to TRUE.

7.2.2 Example Access to an array element

Prerequisite

Two arrays are defined below. These are the basis for the following examples:

```
//A(Thread)
      (0.3 / 0.075 / 0.202)
      (0.4 / 0.1   / 0.270)
      (0.5 / 0.125 / 0.338)
      (0.6 / 0.15  / 0.406)
      (0.8 / 0.2   / 0.540)
      (1.0 / 0.25  / 0.676)
      (1.2 / 0.25  / 0.676)
      (1.4 / 0.3   / 1.010)
      (1.7 / 0.35  / 1.246)

//END

//A(Array2)
      ("DES" /      "PTCH" /      "CDM" )
      (0.3 /      0.075 /      0.202 )
      (0.4 /      0.1 /      0.270 )
      (0.5 /      0.125 /      0.338 )
      (0.6 /      0.15 /      0.406 )
      (0.8 /      0.2 /      0.540 )
      (1.0 /      0.25 /      0.676 )
      (1.2 /      0.25 /      0.676 )
```

```

(1.4 /      0.3 /      1.010 )
(1.7 /      0.35 /     1.246 )

//END

```

Examples

- Access mode example 1:**
 The search concept is in Z. This key is always sought in column 0. The value from column S is output with the line index of the concept found.
`VAR1 = Thread[0.5,1,1] ;VAR1 has the value 0.125`
 Explanation:
 Search for value 0.5 in column 0 of "Thread" array and output the value found in column 1 of the same line.
- Access mode example 2:**
 The search concept is in S. This concept is always searched for in line 0. The value from line Z is output with the column index of the concept found:
`VAR1 = ARRAY2[3,"PTCH",2] ;VAR1 has the value 0.125`
 Explanation:
 Search for column containing "PTCH" in line 0 of array "Array2". Output the value from the column found and the line with index 3.
- Access mode example 3:**
 A search concept is in each of Z and S. The line index is searched for in column 0 with the concept in Z and the column index in line 0 with the concept in S. The value from the array is output with the line index and column index found:
`VAR1 = ARRAY2[0.6,"PTCH",3] ;VAR1 has the value 0.15`
 Explanation:
 Search for the line with the content 0.6 in column 0 of array "Array2", search for the column with the content "STG" in line 0 of Array2. Transfer the value from the line and column found to VAR1.
- Access mode example 4:**
 The search concept is in Z. S contains the column index of the column in which concept is being searched for. The line index of the concept found is output:
`VAR1 = Thread[0.125,1,4] ;VAR1 has the value 2`
 Explanation:
 Search for value 0.125 in column 1 of array "Thread" and transfer the line index of the value found to VAR1.
- Access mode example 5:**
 Z contains the line index of line in which concept is being searched for. The search concept is in S. The column index of the concept found is output:
`VAR1 = Thread[4,0.2,5,1] ;VAR1 has the value 1`
 Explanation:
 Search in line 4 of the "Thread" array for the value 0.2 and transfer the column index of the value found to VAR1. Comparison mode 1 was selected because the values of line 4 are not sorted in ascending order.

7.2.3 Scanning the status of an array element

Description

The Status property can be used to run a scan to find out whether an array access operation is supplying a valid value.

Programming

Syntax:	<i>Identifier</i> [<i>Z</i> , <i>S</i> , [<i>M</i> , <i>C</i>]]]. vld
Description:	Status is a read-only property.
Parameters:	Identifier Name of array
Return Value:	FALSE =invalid value TRUE = valid value

Example

```
DEF MPIT = (R///"MPIT",, "MPIT", ""/wr3)
DEF PIT  = (R///"PIT",, "PIT", ""/wr3)
PRESS (VS1)
  MPIT = 0.6
  IF MET_G[MPIT,0,4,1].VLD == TRUE
    PIT  = MET_G[MPIT,1,0].VAL
    REG[4] = PIT
    REG[1] = "OK"
  ELSE
    REG[1] = "ERROR"
  ENDIF
END_PRESS
```

7.3 Table description (grid)

Definition

In contrast to the array, the values of a table (grid) are continually updated. It is a tabular display of values that can come from the NC or PLC. The table is organized in columns and always has variables of the same type in one column.

Assignment

The reference to a table description is performed in the variables definition:

- The variables definition determines the values to be displayed, and the definition of table elements determines the appearance and arrangement on the screen window. The properties of the input/output fields from the definition line of the variables are taken over in the table.
- The visible area of the table is determined by the width and height of the input/output field. Any lines or columns that cannot be seen can be displayed by scrolling horizontally and vertically.

Table identifiers

Identifiers of a table containing NCK or PLC values of the same type, which can be addressed via a channel block. The table identifier is differentiated from limits or toggle fields by the addition of a % sign in front of it. The file containing the table description can be specified by adding a comma after the identifier and then inserting the name of the file.

Identifier of a table for values of the same type, e.g. from the NCK or PLC. The table identifier is specified at the position at which the limit values or the toggle field are configured. The table identifier starts with a leading % character. This can be followed by a file name with a comma as separator. It specifies the file in which the table description is defined. Per default, the table is sought in the configuration file in which the screen is configured.

A table definition can also be loaded dynamically, e.g. in the LOAD method via the Load Grid (LG) function.

System or user variable

This parameter remains empty for tables because the column definition lines contain detailed information about the variables to be displayed. The table description can be provided dynamically.

Example

Definition of a MyGridVar variable which displays a "MyGrid1" grid in its input/output field (distance from left: 100, distance from top: Standard, width: 350, height: 100).

```
DEF MyGridVar=(V/% MyGrid1////////100,,350,100)
```

See also

Parameters of variables (Page 69)

Load Grid (LG) (Page 137)

7.3.1 Defining a table (grid)

Description

The table block comprises:

- Header
- 1 to n column descriptions

Programming

Syntax:	<i>//G(Table identifier/Table type/Number of lines/ [Fixed line attribute],[Fixed column attribute])</i>		
Description:	Defines a table (grid)		
Parameters:	Table identifier	The table identifier is used without a leading % sign. It can only be used once in a dialog.	
	Table type	0 (default)	Table for PLC or user data (NCK- and channel-specific data)
		1	and others, reserved
	Number of lines	Number of lines including header	
		The fixed line or fixed column is not scrolled. The number of columns is the number of columns configured.	
	Display of column header line	-1:	The column header line is not displayed
		0:	The column header line is displayed (default)
	Number of fixed columns	0:	No fixed column
		1 ... n:	Number of columns that are to be permanently visible, i.e. not scrolled horizontally

Examples

```
//G(grid1/0/5/-1,2)
```

Column header line is hidden and two fixed columns

```
//G(grid1/0/5/,1)
```

Column header line is displayed and one fixed column

```
//G(grid1/0/5)
```

Column header line is displayed and no fixed columns

7.3.2 Defining columns

Description

For tables (grids), it is advisable to use variables with an index. For PLC or NC variables, the index number with one or more indices is of significance.

The values displayed in a table can be modified directly within the restrictions of the rights granted by the attributes and within any defined limits.

Programming

Syntax:	(Type/Limits/Empty/Long text,column header/Attributes/Help display/System or user variable/Column width/Offset1, Offset2, Offset3) See also Extended configuration syntax (Page 36).	
Description:	Defines columns	
Parameters:	Similar to variables	
	Type	Data type
	Limits	Limit value MIN, limit value MAX
	Long text, column header	
	Attributes	
	Help display	
	System or user variable	As variable, PLC or NC variables should be entered in double quotation marks.
	Column width	Entry in pixels.
	Offset	The increment sizes to increment each index in order to fill the column are specified in the assigned offset parameter: • Offset1: Step width for the 1st index • Offset2: Step width for the 2nd index • Offset3: Step width for the 3rd index

Column header from text file

The column header can be entered as text or text number (\$8xxxx) and is not scrolled.

Modifying column properties

The column properties which can be modified dynamically (written) are:

- Limits (min,max),
- Column header (st),
- Attributes (wr, ac and li),
- Help display (hlp) and
- OPI-Variable (var).

Column properties are modified via the variable identifier in the definition line and the column index (starting at 1).

Example: `VAR1[1].st="Column 1"`

Column properties cannot be read in the LOAD method.

The wr, ac and li attributes can be specified for column definitions.

See also

Load Grid (LG) (Page 137)

7.3.3 Focus control in the table (cell)**Description**

The Row and Col properties can be used to set and calculate the focus within a table:

- Identifier.**Row**
- Identifier.**Col**

Programming

Each cell in a table has the Val and Vld properties.

In order to read and write cell properties, a line and column index must be specified in addition to the variable identifiers from the definition list.

Syntax:	Identifier[Line index, column index].val or Identifier[Line index, column index]
Description:	Val properties
Syntax:	Identifier[Line index, column index].vld
Description:	Vld properties

Example

```
Var1[2,3].val=1.203
```

If the line and column indices are not specified, the indices of the focused cell apply. This corresponds to:

```
Var1.Row =2
```

```
Var1.Col=3
```

```
Var1.val=1.203
```


7.4 Custom widgets

7.4.1 Defining custom widgets

Description

User-specific display elements are configured in the dialog using a custom widget.



Software option

In order to use custom widgets in dialog boxes, you require the following additional software option:

"SINUMERIK Integrate Run MyHMI /3GL" (6FC5800-0AP60-0YB0)

Programming

Definition:	DEF (<i>name</i>)	
Syntax:	(W//" , (<i>library name</i>).(<i>class name</i>) "/"a,b,c,d);	
Description:	W	Defining custom widgets
Parameters:	Name	Custom widget name, freely selectable
	Library name	Can be freely selected, name of the dll (Windows) or (Linux) library file
	Class name	Freely selectable, name of the class function from the previously named library
	a, b, c, d	Position and size of the configuration

Example

A custom widget is defined in the dialog configuration in the following way:

```
DEF Cus = (W//"","slestestcustomwidget.SlEsTestCustomWidget"////
20,20,250,100);
```

7.4.2 Structure of the custom widget library

Description

Essentially, the custom widget library contains a defined class. The name of this class must be specified in the dialog configuration in addition to the library names. Starting from library names, "Run MyScreens" accesses a dll file with the same name, e.g.:

```
slestestcustomwidget.dll
```

Programming

The class definition of the dll file should look like this:

```

#define SLESTESTCUSTOMWIDGET_EXPORT Q_DECL_EXPORT

class SLESTESTCUSTOMWIDGET_EXPORT SLEsTestCustomWidget : public QWidget
{
    Q_OBJECT
    ....
    ....
}

```

7.4.3 Structure of the custom widget interface

Description

The library is supplemented by an interface in order to display the custom widget in the dialog. This contains macro definitions with which "Run MyScreens" initiates the custom widget. The interface is available in the form of a cpp file. The file name can be freely selected, e.g.: sleswidgetfactory.cpp

Programming

The interface is defined as follows:

```

#include "slestestcustomwidget.h"      ; The header file for the relevant custom widg-
                                     ets is inserted at the beginning of the file
....
//Makros                             ; Macro definitions are not changed
....
WIDGET_CLASS_EXPORT(SLEsTestCustom-  ; The relevant custom widget is declared at
Widget)                             the end of the file

```

Example

Content of the file sleswidgetfactory.cpp for a custom widget with the class name SLEsTestCustomWidget":

```

#include <Qt/qglobal.h>
#include "slestestcustomwidget.h"

////////////////////////////////////

```

```

// MAKROS FOR PLUGIN DLL-EXPORT - DO NOT CHANGE
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

#ifndef Q_EXTERN_C
#ifdef __cplusplus
#define Q_EXTERN_C      extern "C"
#else
#define Q_EXTERN_C      extern
#endif
#endif

#define SL_ES_FCT_NAME(PLUGIN) sl_es_create_ ##PLUGIN
#define SL_ES_CUSTOM_WIDGET_PLUGIN_INSTANTIATE( IMPLEMENTATION , PARAM) \
    { \
        IMPLEMENTATION *i = new PARAM; \
        return i; \
    }

#ifdef Q_WS_WIN
#   ifdef Q_CC_BOR
#       define EXPORT_SL_ES_CUSTOM_WIDGET_PLUGIN(PLUGIN,PARAM) \
Q_EXTERN_C __declspec(dllexport) void* \
        __stdcall SL_ES_FCT_NAME(PLUGIN) (QWidget* pParent) \
        SL_ES_CUSTOM_WIDGET_PLUGIN_INSTANTIATE( PLUGIN,PARAM )
#   else
#       define EXPORT_SL_ES_CUSTOM_WIDGET_PLUGIN(PLUGIN,PARAM) \
Q_EXTERN_C __declspec(dllexport) void* SL_ES_FCT_NAME(PLUGIN) \
        (QWidget* pParent) \
        SL_ES_CUSTOM_WIDGET_PLUGIN_INSTANTIATE( PLUGIN,PARAM )
#   endif
#else
#   define EXPORT_SL_ES_CUSTOM_WIDGET_PLUGIN(PLUGIN,PARAM) \
Q_EXTERN_C void* SL_ES_FCT_NAME(PLUGIN) (QWidget* pParent) \
        SL_ES_CUSTOM_WIDGET_PLUGIN_INSTANTIATE( PLUGIN,PARAM )
#endif

#define WIDGET_CLASS_EXPORT(CLASSNAME) \
    EXPORT_SL_ES_CUSTOM_WIDGET_PLUGIN(CLASSNAME,CLASSNAME(pParent))

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
// FOR OEM USER - please declare here your widget classes for export
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

WIDGET_CLASS_EXPORT(SlEsTestCustomWidget)

```

7.4.4 Interaction between custom widget and dialog box - Automatic data exchange

Custom widgets interact with dialog boxes and can display values or manipulate them.

Conditions

Automatic data exchange takes place under the following conditions:

Condition	Direction
When starting or recompiling a dialog	Dialog → custom widget
When executing the GC command for generating cycle calls	Custom widget → Dialog

Programming

The following definitions are necessary for the interaction:

Expansion of the dialog configuration

Definition:	DEF (<i>variable</i>)	
Syntax:	((<i>type</i>)/ <i>l5</i> /"", "(<i>variable</i> ", ""/wr2/)	
Variable type:	Type	Standard input field (no grid or toggle) with any data type (no W)
Parameters:	Variable	Any designation of a variable for data exchange
Input mode:	wr2	Reading and writing

Example

```
DEF CUSVAR1 = (R//5/"", "CUSVAR1", ""/wr2/)
```

Expansion of the class definition

In the class definition of the custom widgets, a QProperty must be created whose name is identical to the selected variable of the dialog configuration, e.g.:

```
Q_PROPERTY(double CUSVAR1 READ cusVar1 WRITE setCusVar1);
```

Example

The class definition of the dll file should look like this:

```
#define SLESTESTCUSTOMWIDGET_EXPORT Q_DECL_EXPORT

class SLESTESTCUSTOMWIDGET_EXPORT SLEstTestCustomWidget : public QWidget
{
    Q_OBJECT
    Q_PROPERTY(double CUSVAR1 READ cusVar1 WRITE setCusVar1);
    ....
    ....
}
```

7.4.5 Interaction between custom widget and dialog box - Manual data exchange

Besides automatic data exchange, manual data exchange is also possible. Data exchange is dynamic, i.e. takes place while the dialog box is running. The following actions are possible:

- Properties of the custom widget can be read and written.
- Methods of the custom widget can be called from the Run MyScreens configuration.
- A response to a particular signal of the custom widget can be implemented in order to call subprograms (SUB) in the Run MyScreens configuration.

7.4.5.1 Reading and writing properties

Description

The functions ReadCWProperties and WriteCWProperties are provided in the Run MyScreens configuration for reading and writing properties of the custom widgets.

Programming

Syntax:	ReadCWProperty ("Variablename", "Propertyname")	
Description:	Reading a property of a custom widget	
Parameters:	Variablename	Name of a dialog box variable to which a custom widget is assigned
	Propertyname	Name of the custom widget property to be read
Return value:	Current value of the custom widget property	

Syntax:	WriteCWProperty ("Variablename", "Propertyname", "Value")	
Description:	Writing the property of a custom widget	
Parameters:	Variable name	Name of a dialog box variable to which a custom widget is assigned
	Propertyname	Name of the custom widget property to be written
	Value	Value to be written to the property of the custom widget

Examples

Example 1:

Read property "MyStringVar" of the custom widget that is linked to dialog box variable "MyCWVar1" and assign the value to register 7.

Custom widget class declaration:

```
class SLESTESTCUSTOMWIDGET_EXPORT SLEstTestCustomWidget : public QWidget
{
    Q_OBJECT
    Q_PROPERTY(QString MyStringVar
                READ myStringVar
                WRITE setMyStringVar);
    ...
}
```

Dialog box configuration:

```
DEF MyCWVar1 = (W///,"sleptestcustomwidget.SLEstTestCustomWidget")
PRESS (VS1)
    REG[7]=ReadCWProperty("MyCWVar1", "MyStringVar")
END_PRESS
```

Example 2:

Write the result of the calculation "3 + sin(123.456)" into property "MyRealVar" of the custom widget that is linked to dialog box variable "MyCWVar1."

Custom widget class declaration:

```
class SLESTESTCUSTOMWIDGET_EXPORT SLEstTestCustomWidget : public QWidget
{
    Q_OBJECT
    Q_PROPERTY(double MyRealVar
                READ myRealVar
                WRITE setMyRealVar);
    ...
}
```

Dialog box configuration:

```
DEF MyCWVar1 = (W///,"sleptestcustomwidget.SLEstTestCustomWidget")
PRESS (VS1)
    WriteCWProperty("MyCWVar1", "MyRealVar", 3 + sin(123.456))
END_PRESS
```

7.4.5.2 Executing a method of the custom widget**Description**

The function CallCWMethod is available in the Run MyScreens configuration for executing methods of the custom widget.

The custom widget method to be called must have no more than 10 transfer parameters.

The following transfer parameter data formats are supported:

- bool
- uint
- int
- double
- QString
- QByteArray

Programming

Syntax:	CallCWMMethod ("Variablename", "Methodname[, Argument 0][, Argument 1 ...[,Argument 9]")	
Description:	Calling a custom widget method	
Parameters:	Variable name	Name of a dialog box variable to which a custom widget is assigned
	Method name	Name of the custom widget name to be called
	Argument 0 - 9	Transfer parameter for the custom widget method Supported data formats: see above Note: The transfer parameters are always passed "By-Val," i.e. only the value and not, for example, the reference is passed to a variable.
Return value:	Return value of the custom widget method The following transfer parameter data formats are supported: • void • bool • uint • int • double • QString • QByteArray Notice: Even if the data format of the return value of the custom widget method is "void," it must be formally assigned to a variable, for example.	

Example

Custom widget class declaration:

```
class SLESTESTCUSTOMWIDGET_EXPORT SLEstTestCustomWidget : public QWidget
{
    Q_OBJECT
    public slots:
        void myFunc1(int nValue, const QString& szString, double dValue);
    ...
}
```

Dialog box configuration:

```

DEF MyCWVar1 = (W///,"slesteTestCustomWidget.SlEsTestCustomWidget")

DEF MyStringVar1 = (S)
DEF MyRealVar = (R)

PRESS (VS3)
    REG[9] = CallCWMethod("MyCWVar1", "myFunc1", 1+7, MyStringVar1, sin(MyRealVar) -
8)
END_PRESS

```

7.4.5.3 Response to a custom widget signal**Description**

In Run MyScreens it is possible to respond to a particular signal (invokeSub()) of the custom widget and call up a subprogram (SUB).

10 global variables, the so-called SIGARG, are available for passing values (custom widget signal -> SUB), which are comparable to registers (REG) in configuration. This is where the values transferred with the custom widget signal are stored.

The following transfer parameter data formats are supported:

- bool
- uint
- int
- double
- QString
- QByteArray

Programming**Calling the subroutine:**

Syntax:	void invokeSub(const QString& rszSignalName, const QVariantList& rvntList);	
Description:	Custom widget signal with which a Run MyScreens subprogram is called.	
Parameters:	rszSignalName	Name of the Run MyScreens subprogram to be called
	rvntList	QVariantList array for transferring parameters that are stored in the global parameter SIGARG and which are available in the configuration. Maximum size: 10 elements Data formats supported: see above Note: The transfer parameters are always passed "By-Val," i.e. only the value and not, for example, the reference to a variable is passed.

Subroutine to be called:

Syntax:	SUB (on_<Variablename>_<Signalname>) ... END_SUB	
Description:	Response to a custom widget signal	
Parameters:	Variable name	Name of a dialog box variable to which a custom widget is assigned.
	Signal name	Name of the custom widget signal
	SIGARG 0 - 9	Transfer parameters for the custom widget method. Supported data formats: See above Notice: The transfer parameters are always passed "By-Val," i.e. only the value and not, for example, the reference to a variable is passed.

Example**Customer widget class declaration:**

```
class SLESTESTCUSTOMWIDGET_EXPORT SLEsTestCustomWidget : public QWidget
{
    Q_OBJECT signals: void invokeSub(const QString& szSubName, const QVariantList&
vntList);
    ...
}
```

Custom widget class:

```
QVariantList vntList;
vntList << 123.456;
emit invokeSub("MySub", vntList);
```

Dialog box configuration:

```
DEF MyCWVar1 = (W///,"slesteStcustomwidget.SLEsTestCustomWidget")
SUB (on_MyCWVar1_MySub)
    DEBUG("SUB(on_MyCWVar1_MySub) was called with parameter: "" << SIGARG[0] <<
""")
END_SUB
```

Result easyscreen_log.txt:

```
[10:22:40.445] DEBUG: SUB(on_MyCWVar1_MySub) was called with parameter: "123.456"
```


"Custom" operating area

8.1 How to activate the "Custom" operating area

Activating the "Custom" operating area

The "Custom" operating area is not activated on delivery.

1. First copy the slamconfig.ini. file from the *[System Siemens directory]*cfg directory to the *[System oem directory]*cfg directory or correspondingly to the *[System add_on directory]*cfg directory or *[System user directory]*cfg directory.
2. To activate the "Custom" operating area, the following must be entered:

```
[Custom]  
Visible=True
```

Result

After activation is complete, the softkey for the "Custom" operating area can be found in the main menu (F10) on the menu continuation bar on the HSK4 (= default).

The "Custom" operating area displays an empty window covering the entire operating area, with a configurable header. All horizontal and vertical softkeys can be configured.

8.2 How to configure the "Custom" softkey

Configuring the softkey for the "Custom" operating area

The labeling and position of the softkey for the "Custom" operating area are configured in the slamconfig.ini file.

The following options are available for configuring the start softkey:

1. To replace a softkey label with a **language-dependent text**, the following must be entered in the [Custom] section:
`TextId=MY_TEXT_ID`
`TextFile=mytextfile`
`TextContext=mycontext`
In this example, the softkey shows the language-dependent text which was saved with the text ID "MY_TEXT_ID" in text file mytextfile_XXX.qm under "MyContext" (XXX stands for language code).
2. To replace a softkey label with a **language-neutral text**, the following must be entered in the [Custom] section:
`TextId=HELLO`
`TextFile=<empty>`
`TextContext=<empty>`
In this example, the softkey for the "Custom" operating area displays the text "HELLO" for every language.
3. **An icon** can also be displayed on the softkey in addition to the text.
To do this, the following must be entered in the [Custom] section:
`Picture=mypicture.png`
The softkey then displays the icon from the file mypicture.png. Graphics and bitmaps are stored at the following path: *[System oem directory]/ico/ico<Resolution>*. The directory that corresponds to the display resolution must be used.
4. **The position** of the softkey can also be set. The following entry in the [Custom] section makes this setting:
`SoftkeyPosition=12`
The default is position 12. This corresponds to the HSK4 on the menu continuation bar of the operating area's menu. Positions 1 - 8 correspond to HSK1 to HSK8 on the menu bar, positions 9 - 16 to HSK1 to HSK8 on the menu continuation bar.

8.3 How to configure the "Custom" operating area

Configuring the softkey for the "Custom" operating area

You need the easyscreen.ini and custom.ini files to configure the operating area. Templates for both these files are available in the *[System Siemens directory]/templates/cfg* directory.

1. First copy the files to the *[System oem directory]/cfg* directory and make your changes from there.
2. File easyscreen.ini already contains a definition line for the "Custom" operating area:
`;StartFile02 = area := Custom, dialog := SlEsCustomDialog,`
`startfile := custom.com`
The ";" at the start of the line represents the comment character. This means the line is commented out and, as such, not active. To change this, the ";" must be deleted.
The "startfile" attribute in this line defines that the entry will refer to the custom.com project file when the "Custom" operating area is selected.

3. You create the **custom.com project file** in the *[System oem directory]/proj* directory. This contains the relevant configuration which is created in the same way as the *aeditor.com* file of the "Program" operating area. The configured start softkeys are then displayed in the "Custom" operating area.
4. You configure the **language-neutral text** for the title bar of the dialog in the *custom.ini* file. The following entry is available in the template for this purpose:

```
[Header]  
Text=Custom
```

You can replace this text with a customized one.
5. The template contains the following entry for configuring a **start screen** for the "Custom" operating area:

```
[Picture]  
Picture=logo.png
```

Logo.png is the name of the start screen which appears on the "Custom" operating area's start dialog. Here you can display a company logo, for example, or another image. The file should be saved in the directory for the corresponding resolution under: *[System oem directory]/icol/ ...*
6. In order to display a specific "Run MyScreens" screen when the "Custom" operating area is shown the first time, the following entry is available in the template:

```
; Mask shown with startup of area "custom"  
[Startup]  
;Startup = Mask:=MyCustomStartupMask, File:=mycustommasks.com
```

Typically, SINUMERIK Operate is started directly with a specific "Run MyScreens" screen. The "startuparea" key must be set as follows in the "[miscellaneous]" section of the *systemconfiguration.ini* file:

```
[miscellaneous]  
;name of the area to be shown at system startup  
startuparea = Custom
```

8.4 Programming example for the "Custom" area

Example

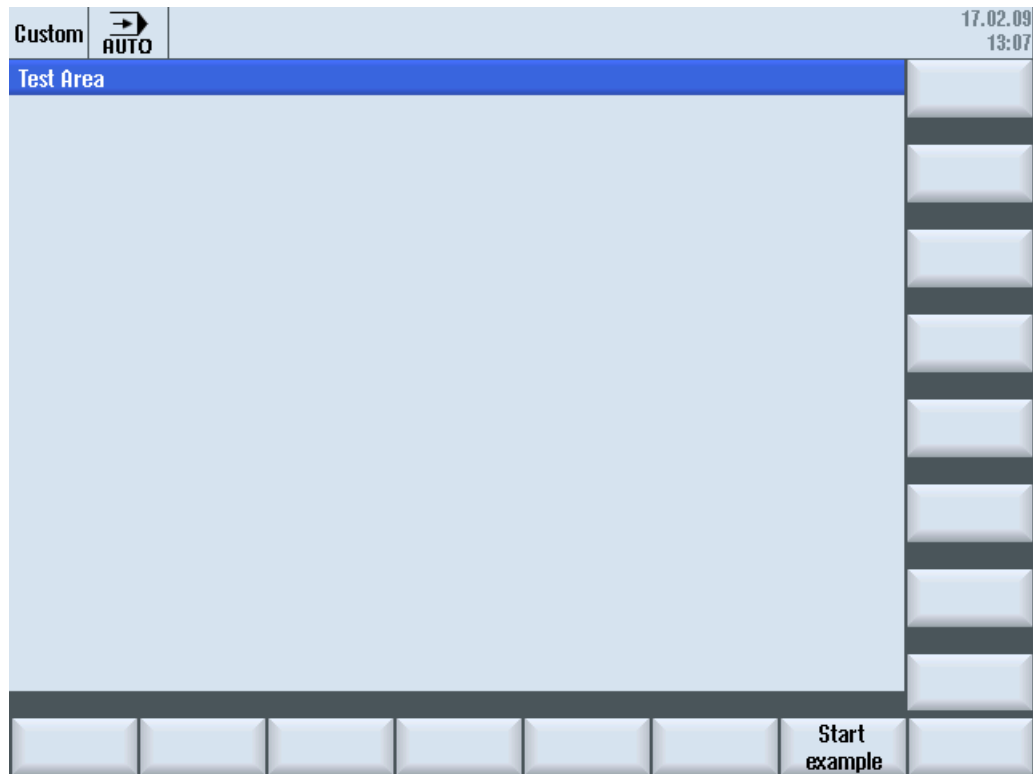


Figure 8-1 Example with "Start example" softkey

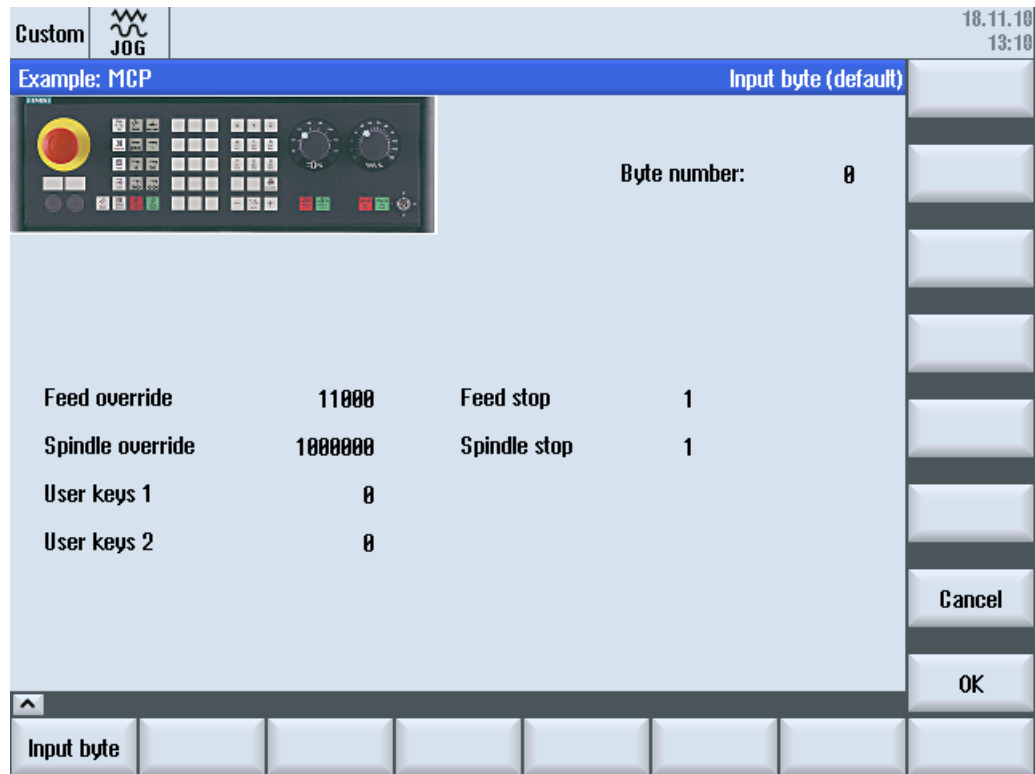


Figure 8-2 Example with bitmap and text fields

File overview

The following files are required:

- custom.com
- easyscreen.ini

Programming

Content of the custom.com file:

Note

The graphic file mcp.png integrated in the example is also only a sample file. If you wish to use this programming example in your application, you must replace the graphic by one of your own graphics.

```
//S(Start)
HS7("Start example", sel, ac7)
PRESS(HS7)
LM("Maske4")
```

8.4 Programming example for the "Custom" area

```
END_PRESS
//END
//M(Maske4/"Example: MCP"/"mcp.png")
DEF byte=(I/O/0/"Input byte=0 (default)", "Byte number:", ""/wr1,li1//380,40,100/480,40,50)
DEF Feed=(IBB//0/""/"Feed override", ""/wr1//EB3"/20,180,100/130,180,100), Axistop=(B//0/""/"Feed
stop", ""/wr1//E2.2"/280,180,100/380,180,50/100)
DEF Spin=(IBB//0/""/"Spindle override", ""/wr1//EB0"/20,210,100/130,210,100), spinstop=(B//
0/""/"Spindle stop", ""/wr1//E2.4"/280,210,100/380,210,50/100)
DEF custom1=(IBB//0/""/" User keys 1", ""/wr1//EB7.7"/20,240,100/130,240,100)
DEF custom2=(IBB//0/""/"User keys 2", ""/wr1//EB7.5"/20,270,100/130,270,100)
DEF By1
DEF By2
DEF By3
DEF By6
DEF By7

HS1=("Input byte", SE1, AC4)
HS2=("")
HS3=("")
HS4=("")
HS5=("")
HS6=("")
HS7=("")
HS8=("")
VS1=("")
VS2=("")
VS3=("")
VS4=("")
VS5=("")
VS6=("")
VS7=("Cancel", SE1, AC7)
VS8=("OK", SE1, AC7)

PRESS (VS7)
EXIT
END_PRESS

PRESS (VS8)
EXIT
END_PRESS

LOAD
By1=1
By2=2
By3=3
```



```
By6=6
By7=7
END_LOAD

PRESS (HS1)
  Byte.wr=2
END_PRESS

CHANGE (Byte)
  By1=byte+1
  By2=byte+2
  By3=byte+3
  By6=byte+6
  By7=byte+7
  Feed.VAR="EB"<<By3
  Spin.VAR="EB"<<Byte
  Custom1.VAR="EB"<<By6
  Custom2.VAR="EB"<<By7
  Axisstop.VAR="E"<<By2<<".2"
  Spinstop.VAR="E"<<By2<<".4"
  Byte.wr=1
END_CHANGE

CHANGE (Axis stop)
  IF Axistop==0
    Axistop.BC=9
  ELSE
    Axistop.BC=11
  ENDIF
END_CHANGE

CHANGE (Spin stop)
  IF Spinstop==0
    Spinstop.BC=9
  ELSE
    Spinstop.BC=11
  ENDIF
END_CHANGE
//END
```


Dialog selection

9.1 Dialog selection using PLC softkeys

Configuration

Description of the procedure:

- The systemconfiguration.ini contains a section [keyconfiguration]. The entry specifies an action for a special PLC softkey.
- A number is given as an action. A "Run MyScreens" call is involved if the number is greater than or equal to 100.
- A section for defining the action to be performed must be created in the easyscreen.ini file. The name of the section is based on the name of the operating area and the dialog name (see entry under [keyconfiguration] → Area:=..., Dialog:=...) → [<Area>_<Dialog>] → e.g. [AreaParameter_SIPaDialog].
- The action numbers (which were given in the systemconfiguration.ini → see Action:=...) are defined in this section. There are two commands involved:
 1. LS("Softkey menu1","param.com") ... Loading a softkey menu
 2. LM("Screen form1","param.com") ... Loading a screen form

Selecting softkey menus via PLC softkeys

"Run MyScreens" makes it possible to select "Run MyScreens" softkey menus and "Run MyScreens" dialogs via PLC softkeys. This can only be done if the "action" attribute to be specified when configuring the relevant PLC softkeys has a value greater than or equal to 100.

PLC softkeys are configured in the file `systemconfiguration.ini` in the section [keyconfiguration]:

```
[keyconfiguration]
KEY75.1 = Area:=area, Dialog:=dialog, Screen:=screen, Action:= 100,
```

9.2 Dialog selection using PLC hard keys

The LM and LS commands to be executed upon activation of the relevant PLC softkeys are configured in the easyscreen.ini file. The names of the sections that are used for the purpose of configuration are structured as follows:

[areaname_dialogname]	The first part of the name "areaname" refers to the operating area and the second part "dialogname" designates the dialog to which the commands configured in this section apply.
<pre>[AreaParameter_SlPaDialog] 100.screen1 = LS("Softkey1", "param.com") 101.screen3 = LM("Screen form1", "param.com")</pre>	The names given in the systemconfiguration.ini file for the operating area and dialog should be used. The dialog does not have to be specified. This is particularly true for operating areas which are only implemented by means of a single dialog. Please refer to the example on the left. If "screen1" is displayed in the AreaParameter operating area implemented by the SlPaDialog dialog, the "LS("Softkey1", "param.com")" command will be executed when the "action" with the value 100 occurs.
action.screen=Command	Both the "action" and "screen" attributes clearly indicate when the specified command will be executed. The "screen" information is optional. The following commands are permissible: LM (LoadMask) LS (LoadSoftkeys)

9.2 Dialog selection using PLC hard keys

Field of application

The following functions can be initiated in the operating software by the PLC:

- Select an operating area
- Select certain scenarios within operating areas
- Execute functions configured at softkeys

Hard keys

All keys - also the PLC keys - are subsequently referred to as hard keys. A maximum of 254 hard keys can be defined. The following allocation applies:

Key number	Application
Key 1 – key 9	Keys on the operator panel front
Key 10 – key 49	Reserved
Key 50 – key 254 Key 50 – key 81	PLC keys: reserved for OEMs
Key 255	Pre-assigned by control information.

Hard keys 1 - 9 are pre-assigned as follows:

Key designation		Action / effect
HK1	Position	Selects "Machine" operating area, last dialog
HK2	Program	Selects "Program" operating area, last dialog or last program
HK3	Offset	Selects "Parameter" operating area, last dialog
HK4	Program manager	No function
HK5	Alarm	Selects diagnostics operating area, "Alarm list" dialog

Configuration

The configuring is realized in the systemconfiguration.ini configuration file in the section [keyconfiguration]. Each line defines what is known as a hard key event. A hard key event is the n-th actuation of a specific hard key. For example, the second and third actuation of a specific hard key can result in different responses.

The entries in the systemconfiguration.ini configuration file can be overwritten with user-specific settings. The directories *[System user directory]\cfg* and *[System oem directory]\cfg* are available for this purpose.

The lines for configuring the hard key events have the following structure:

```
KEYx.n = Area:=area, Dialog:=dialog, Screen:=screen, Forms:=form,
Menus:=menu,
Action:=menu.action, Cmdline:=cmdline
KEYx.n = Area:=area, Dialog:=dialog, Cmdline:=cmdline, Action:=
action
```

x: Number of the hard key, range of values: 1 – 254

n: Event number – corresponds to the nth actuation of the hard key, value range: 0 – 9

Requirement

The PLC user program must fulfill the following requirement:

Only one hard key is processed. As a consequence, a new request can only be set if the operating software has acknowledged the previous request. If the PLC user program derives the hard key from an MCP key, it must provide sufficient buffer storage of the key(s) to ensure that no fast keystrokes are lost.

PLC interface

An area to select a hard key is provided in the PLC interface. The area is in DB19.DBB10. Here, the PLC can directly specify a key value of between 50 and 254.

Acknowledgment by the operating software takes place in two steps. This procedure is necessary so that the operating software can correctly identify two separate events if the same key code is entered twice consecutively. In the first step, control information 255 is written to byte DB19.DBB10. This defined virtual key activation enables the HMI to identify every PLC key sequence uniquely. The control information is of no significance to the PLC user program and must not be changed. In the second step, the actual acknowledgment takes place with

respect to the PLC by clearing DB19.DBB10. From this point in time, the PLC user program can specify a new hard key. In parallel, the actual hard key request is processed in the operating software.

Example

Configuration file:

```
; PLC hard keys (KEY50-KEY254)
[keyconfiguration]
KEY50.0 = name := AreaMachine, dialog := SlMachine
KEY51.0 = name := AreaParameter, dialog := SlParameter
KEY52.0 = name := AreaProgramEdit, dialog := SlProgramEdit
KEY53.0 = name := AreaProgramManager, dialog := SlPmDialog
KEY54.0 = name := AreaDiagnosis, dialog := SlDgDialog
KEY55.0 = name := AreaStartup, dialog := SlSuDialog
KEY56.0 = name := Custom, dialog := SlEsCustomDialog
```

The area and dialog identifiers can be found in the systemconfiguration.ini from *[System Siemens directory]\cfg*.

9.3 Dialog selection via NC

MMC command in HMI Operate

You can use MMC commands as described in the following:

1. Definition of MMC commands

The following is available in the standard `systemconfiguration.ini` file.

Combinations:

address:=MCYCLES --> command:=LM

address:=CYCLES --> command:=PICTURE_ON

This distinction is necessary in order to differentiate between measuring cycles and other cycles. This means that:

- LM is always applicable for measuring cycles, PICTURE_ON, always for non-measuring cycles
- It is not permissible that newly defined MMC commands are designated PICTURE_ON and LM

2. "Run MyScreens" license

All the dialogs opened by "Run MyScreen", with the exception of measuring cycles, fall under the "Run MyScreens" license, which means that without license, only a limited number of dialogs can be used.

Example call with test.com (Run MyScreens):

```
g0 f50
```

```
MMC ("CYCLES, PICTURE_ON, test.com, screen form1", "A")
```

```
m0
```

```
MMC ("CYCLES, PICTURE_OFF", "N")
```

```
M30
```


Reference lists

A.1 Lists of start softkeys

A.1.1 List of start softkeys for turning

Program operating area for turning

HSK1	HSK2	HSK3	HSK4	HSK5	HSK6	HSK7	HSK8
Edit	Drilling	Turning	Contour turning	Milling	Miscellaneous	Simulation	NC select
HSK9	HSK10	HSK11	HSK12	HSK13	HSK14	HSK15	HSK16
--	Straight-line circle (only for ShopTurn)	--	--	Measure workpiece	Measure tool	OEM	--

Turning

The following tables list the possible start softkeys for turning technology. Assignments of individual start softkeys can differ depending on the particular system. The specified OEM softkeys are permitted for "Run MyScreens".

programGUIDE (G-Code) start softkeys:

	Drilling	Turning	Contour turning		Milling		Miscellaneous	
	HSK2	HSK3	HSK4		HSK5		HSK6	
VSK1	Centering	Stock removal	Contour	--	Face milling	Contour	Settings	High speed settings
VSK2	Drilling reaming	Groove	Stock removal	--	Pocket	Path	Swivel plane	Parallel axes
VSK3	Deep-hole drilling	Undercut	Stock removal residual material	--	Multi-edge spigot	Predrilling	Swivel tool	--
VSK4	Boring	Thread	Grooving	--	Groove	Pocket	--	--
VSK5	Thread	Parting	Grooving residual material	--	Thread milling	Pocket res. mat.	--	--
VSK6	OEM	--	Plunge-turning	--	Engraving	Spigot	Subprogram	--

A.1 Lists of start softkeys

VSK7	Positions	OEM	Plunge turning residual material	OEM	OEM	Spigot res. mat.	--	OEM
VSK8	Repeat position.	--	>>	<<	Contour milling	<<	>>	<<

ShopTurn start softkeys:

	Drilling	Turning	Contour turning		Milling		Miscellaneous		
	HSK2	HSK3	HSK4		HSK5		HSK6		HSK10
VSK1	Drilling centered	Stock removal	New contour	--	Face milling	New contour	Settings	High speed settings	Tool
VSK2	Centering	Groove	Stock removal	--	Pocket	Path	Swivel plane	Parallel axes	Straight line
VSK3	Drilling reaming	Undercut	Stock removal residual material	--	Multi-edge spigot	Predrilling	Swivel tool	Repeat progr.	Circle center point
VSK4	Deep-hole drilling	Thread	Grooving	--	Groove	Pocket	Counter-spindle	--	Circle radius
VSK5	Thread	Parting	Grooving residual material	--	Thread milling	Pocket res. mat.	Transformations	--	Polar
VSK6	OEM	--	Plunge-turning	--	Engraving	Spigot	Subprogram	--	Approach/retract
VSK7	Positions	OEM	Plunge turning residual material	OEM	OEM	Spigot res. mat.	--	OEM	--
VSK8	Repeat position.	--	>>	<<	Contour milling	<<	>>	<<	--

See also

Defining the start softkey (Page 28)

A.1.2 List of start softkeys for milling

Program operating area when milling

HSK1	HSK2	HSK3	HSK4	HSK5	HSK6	HSK7	HSK8
Edit	Drilling	Milling	Contour milling	Turning (only for G code)	Miscellaneous	Simulation	NC select

HSK9	HSK10	HSK11	HSK12	HSK13	HSK14	HSK15	HSK16
--	Straight-line circle (only for ShopMill)	--	--	Measure workpiece	Measure tool	OEM	--

Milling

The following tables list the possible start softkeys for milling technology. Assignments of individual start softkeys can differ depending on the particular system. The specified OEM softkeys are permitted for "Run MyScreens".

programGUIDE (G-Code) start softkeys:

	Drilling	Milling	Contour milling		Turning		Miscellaneous	
	HSK2	HSK3	HSK4		HSK5		HSK6	
VSK1	Centering	Face milling	Contour	--	Stock re- moval	Contour	Settings	--
VSK2	Drilling ream- ing	Pocket	Path	--	Groove	Stock re- moval	Swivel plane	Parallel axes
VSK3	Deep-hole drill- ing	Multi-edge spigot	Predrilling	--	Undercut	Stock re- moval re- sidual ma- terial	Swivel tool	--
VSK4	Boring	Groove	Pocket	--	Thread	Grooving	High speed set- tings	--
VSK5	Thread	Thread mill- ing	Pocket res. mat.	--	Parting	Grooving residual material	--	--
VSK6	OEM	Engraving	Spigot	--	--	Plunge- turning	Subprogram	--
VSK7	Positions	OEM	Spigot res. mat.	OEM	OEM	Plunge turning re- sidual ma- terial	--	OEM
VSK8	Repeat posi- tion.	--	>>	<<	Contour turning	<<	>>	<<

ShopMill start softkeys:

	Drilling	Milling	Contour milling		Miscellaneous		Straight line circle
	HSK2	HSK3	HSK4		HSK6		HSK10
VSK1	Centering	Face milling	New contour	--	Settings	--	Tool
VSK2	Drilling ream- ing	Pocket	Path	--	Swivel plane	Parallel axes	Straight line
VSK3	Deep-hole drilling	Multi-edge spigot	Predrilling	--	Swivel tool	Repeat progr.	Circle center point
VSK4	Boring	Groove	Pocket	--	High speed set- tings	--	Circle radius
VSK5	Thread	Thread mill- ing	Pocket res. mat.	--	Transforma- tions	--	Helix

VSK6	OEM	Engraving	Spigot	--	Subprogram	--	Polar
VSK7	Positions	OEM	Spigot res. mat.	OEM	--	OEM	--
VSK8	Repeat position.	--	>>	<<	>>	<<	--

A.2 List of access levels

The meanings of the different access levels are as follows:

Protection level	Locked by	Area
ac0	Reserved for Siemens	
ac1	Password	Machine manufacturer
ac2	Password	Service
ac3	Password	User
ac4	Keyswitch position 3	Programmer, machine setter
ac5	Keyswitch position 2	Qualified operator
ac6	Keyswitch position 1	Trained operator
ac7	Keyswitch position 0	Semi-skilled operator

A.3 List of colors

System colors

A uniform color table is available for configuring dialogs (subset of the respective standard colors). The color of an element (text, input field, background, etc.) can be selected from the following options (between 0 and 133).

As an alternative to the predefined colors, you can also specify colors as RGB values ("#RRGGBB").

Index	Pictogram	Color	Color description
1		Black	
2		Orange	
3		Dark green	
4		Light gray	
5		Dark gray	
6		Blue	
7		Red	

A.4 List of language codes used in file names

Index	Pictogram	Color	Color description
8		Brown	
9		Yellow	
10		White	
126		Black	Font color of an input/output field that is currently in focus
127		Light orange	Background color of an input/output field that is currently in focus
128		Orange	System color focus
129		Light gray	Background color
130		Blue	Header color (active)
131		Black	Header font color (active)
132		Turquoise	Background color of a toggle field
133		Light blue	Background color of a list box

A.4 List of language codes used in file names

Supported languages

Standard languages:

Language	Abbreviation in file name
Chinese simplified	chs
German	deu
English	eng
French	fra
Italian	ita
Spanish	esp

Other languages:

Language	Abbreviation in file name
Chinese traditional	cht
Danish	dan
Finnish	fin

A.6 Behavior when opening the dialog (attribute CB)

Language	Abbreviation in file name
Indonesian	ind
Japanese	jpn
Korean	kor
Malay	msl
Dutch	nld
Polish	plk
Portuguese (Brazil)	ptb
Romanian	rom
Russian	rus
Swedish	sve
Slovakian	sky
Slovenian	slv
Thai	tha
Czech	csy
Turkish	trk
Hungarian	hun
Vietnamese	vit

A.5 List of accessible system variables

References

List Manual System Variables/PGAsI/

A.6 Behavior when opening the dialog (attribute CB)

The following table provides an overview of the conditions when the CHANGE method is called.

The following applies to attribute CB:

CB0

The CHANGE method is triggered when the screen is displayed if the variable has a valid value at this time (e.g. through default setting or NC/PLC variable).

CB1 (default)

The CHANGE method is not explicitly triggered when the screen is displayed. If the variable has a configured NC/PLC variable, then the CHANGE method is of course still called.

Condition				Response
Type	System or user variable	Default setting	Execute the CHANGE method	
I/O field	Yes	Yes	Yes	Due to the configured NC/PLC variable, there is always at least one automatic call of the CHANGE method with the current value of the NC/PLC variable. The pre-assignment of CB does not have an effect.
			No	
		No	Yes	
			No	
	No	Yes	Yes	The CHANGE method is called with the pre-assigned value.
			No	The CHANGE method is not called.
		No	Yes	No call because there is no valid value present to call the CHANGE method.
			No	
Toggle	Yes	Yes	Yes	Due to the configured NC/PLC variable, there is always at least one automatic call of the CHANGE method with the current value of the NC/PLC variable. The pre-assignment of CB does not have an effect.
			No	
		No	Yes	
			No	
	No	Yes	Yes	The CHANGE method is called with the pre-assigned value.
			No	The CHANGE method is not called.
		No (per default the first value from the toggle is assigned)	Yes	The CHANGE method is called with the first pre-assigned value of the toggle list.
			No	The CHANGE method is not called.

See also

CHANGE (Page 96)

A.6 Behavior when opening the dialog (attribute CB)

Tips and tricks

B.1 General tips

- If possible, use the OPI version for system variables (\$ variables).
Reason:
Avoids complex name resolution.
Example:
Instead of "\$R[10]", it is better to use "/Channel/Parameter/R[u1,10]".
- Avoids cyclic (similar) setting, e.g. of the screen and variable HLP property. Compare the value to be set with the current value first, and only set it when it is different.
Reason:
Avoid a time-consuming search of the resolution-dependent help screens.
Example:

```
DEF MyVar=(R3///,"X1",,"mm"/WR1//"$AA_IM[0]")
CHANGE(MyVar)
  IF MyVar.VAL < 100
    HLP="mypic1.png"
  ELSE
    HLP="mypic2.png"
  ENDIF
END_CHANGE
```

Recommendation:

```
CHANGE(MyVar)
  IF MyVar.VAL < 100
    IF HLP <> "mypic1.png"
      HLP="mypic1.png"
    ENDIF
  ELSE
    IF HLP <> "mypic2.png"
      HLP="mypic2.png"
    ENDIF
  ENDIF
END_CHANGE
```

- Replace several RNP() functions in succession with one MRNP() function.
Replace several RDOP() functions in succession with one MRDOP() function.
Reason:
Reduces the communication load and increases the performance.
- Do not read drive parameters faster than in a 1 second cycle, slower is better.
Reason: The communication with the drives can otherwise be extremely disturbed or even cause failures.
- Avoid successive arithmetic operations with system or user variable-connected dialog variables. Use, for example, register (REG[x]) or (invisible) help variables for this.
Reason: Each assignment of a value results in a write operation in the connected system or user variable.
- Similar code used in different blocks should be combined in a SUB block for reasons of transparency, serviceability and performance (when displaying the screen). It can then be called at the appropriate positions with the CALL() function.
- By monitoring the CPU load in the dialog line (setting in slguiconfig.xml), it is possible to investigate the effects of changes in the screen on the performance.

B.2 Tips for debugging

- Use the DEBUG() function for diagnostic purposes. When the DEBUG() function is called, the transferred string is written to the easyscreen_log.txt file. The output of information in the dialog line by the DLGL() function can also be helpful.
However, after completion of the screen development, this function should be removed again for performance reasons or commented out.
- The easyscreen_log.txt file should have no entries after the development of a screen is completed.

B.3 Tips for the CHANGE method

- Always keep CHANGE methods very short and small, particularly for those whose variables are connected to a system or user variable and change very frequently.
Reason:
Increases the screen performance.
- If possible, do not configure any RNP() functions in CHANGE methods. Instead, it is better to configure an invisible variable parallel to the system or user variable to be read and then use this.
Reason:
An RNP() function would be issued with each call. Otherwise, the current value which is already available would be simply accessed.

Example:

A name resolution is performed via RNP() for each change of the axis motion in order to read channel-specific machine data:

```
DEF AXIS_POSITION_X = (R///,""///"$AA_IM[X] ")

CHANGE (AXIS_POSITION_X)
    DLGL("Axis "" << RNP("$MC_AXCONF_GEOAX_NAME_TAB[0] ") << ""
has moved: "
<< AXIS_POSITION_X)
END_CHANGE
```

Keep the channel-specific machine data up-to-date with the aid of an invisible variable, copy each value change to a temporary variable, e.g. register.

This temporary variable can then be used in the CHANGE method of the value change for the axis position without making a name resolution of the machine data each time and the subsequent read access:

```
DEF AXIS_POSITION_X = (R///,""///"$AA_IM[X] ")
DEF AXIS_NAME_X = (S///,""/WR0///"$MC_AXCONF_GEOAX_NAME_TAB[0] ")

CHANGE (AXIS_NAME_X)
    REG[0] = AXIS_NAME_X
END_CHANGE

CHANGE (AXIS_POSITION_X1)
    DLGL("Axis "" << REG[0] << "" has moved " << AXIS_POSITION_X)
END_CHANGE
```

- The update rate and therefore the execution of the associated CHANGE method of variables that are connected to system or user variables with very frequent value changes, can be reduced by using the UR variable property, e.g. variable that is coupled to the actual axis values.
Reason:
In this way, the associated CHANGE method is executed in a fixed specified grid for a value change.

B.4 Tips for DO LOOP loops

As loops can impair the performance of SINUMERIK Operate, depending on the configuration, they should be used carefully and if possible without time-intensive actions.

A register (REG[]) should also be used as run variable, because normal display variables (particularly those with OPI connection) also impair the system performance due to the frequent updates or write operations.

B.4 Tips for DO LOOP loops

Note that the number of script lines executed "in one operation", is currently limited to 10,000. When this number is reached, the script execution is aborted with an appropriate entry in the easyscreen_log.txt file.

Reason:

- Prevention of endless loops
- "Run MyScreens" must remain operable

Animated elements

C.1 Introduction

Introduction

The manual describes working with the X3D Viewer with which you can integrate animated and non-animated graphical scenes (animated elements) – called help screens in the following – into the graphical user interface of SINUMERIK Operate as of Version V4.7 SP1 .

You can show motion sequences and parameters in context-sensitive help screens. This helps you to improve the operation of applications and make the user interface more appealing.

Implementation of your initial idea to the final help screen is described in the steps below:

- Creation of graphical elements and 3D models for the help screens that will eventually appear in the X3D Viewer (see Chapter Modeling (Page 212)).
- Creation of the scene description file in which the model data of the graphics file will be assigned to the scenes and animations to be displayed at particular points (see Chapter Structure of the scene description file (Page 219)).
- Conversion of the X3D files and XML files into HMI files (see Chapter Conversion to hmi file (Page 223)).
- C++ integration of the X3D Viewer into your own application (see Chapter Implementation example (Page 225)).
- Application in Run MyScreens (see Chapter Display in Run MyScreens (Page 226)).

Overview of the file formats

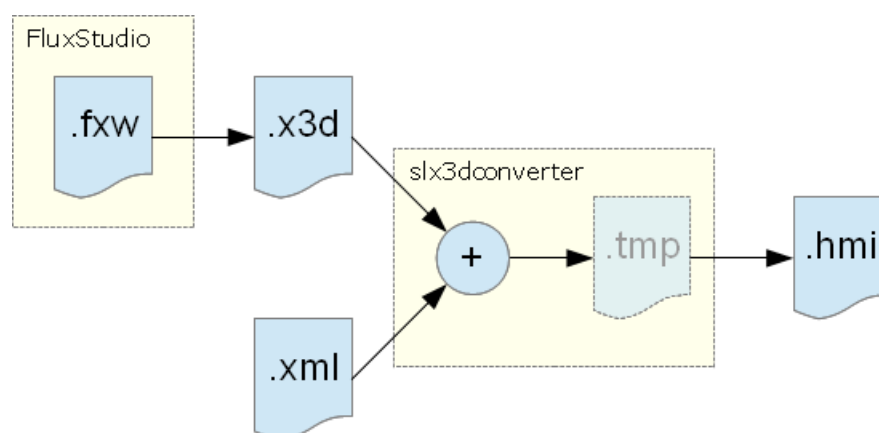


Figure C-1 File formats

File format	Description
.fxw	Flux Studio (3D) graphics file
.x3d	Model file in exchange format
.xml	Definition file for animations and scenes
.hmi	Output file for the X3D Viewer

C.2 Modeling

C.2.1 Requirements

The aim of the modeling is to create a file with the generated 3D models in .x3d file format, an official standard for 3D contents.

Modeling is performed in the **Flux Studio Web3D authoring tool**. The Flux Studio Web3D authoring tool is a freely available 3D modeling tool with versatile export and import options.

Requirements

- You have installed the Flux Studio Web3D authoring tool, Version 2.1.
- You are familiar with modeling and use of Flux Studio.
- File format .x3d is not described in any detail in this document, adequate knowledge of the subject is assumed.

Note

You can download the Flux Studio Web3D authoring tool from the Internet via the following link:

<http://mediamachines.wordpress.com/flux-player-and-flux-studio> (<http://mediamachines.wordpress.com/flux-player-and-flux-studio>)

C.2.2 Rules for modeling

Observe the following rules when modeling. You will only be able to process your help screen file if you have followed these rules correctly.

Rules for modeling

1. The time sensor must have the name "Animation".
2. All related elements must be assigned to a group.

3. All groups must be set to the following position:
x = 0, y = 0, z = 0
4. To make the groups invisible, the translation values are set as follows:
x = 1000000, y = 1000000 and z = 1000000
5. The following elements must have the same name both in the Flux Studio graphic models and in the XML scene definitions (cf. Chapter Overview (Page 218)):
 - Tool
 - Cameras
6. Before you can create a .x3d file you must make the following settings in dialog box "Export Options":

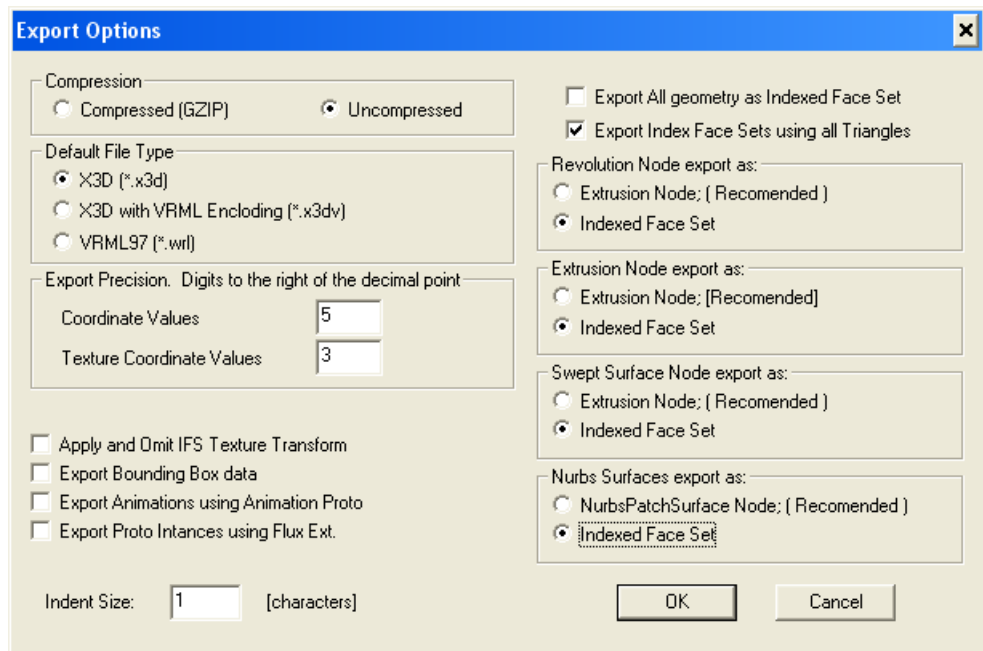


Figure C-2 Settings in the Export Options dialog box

7. We recommend the following settings for texts (see also the following dialog box):
 - Texts must always be centered.
 - The size of the text should be 0.2. This size allows you to position the text easily. The text output by the X3D Viewer is in the font size of the user interface, irrespective of this value.

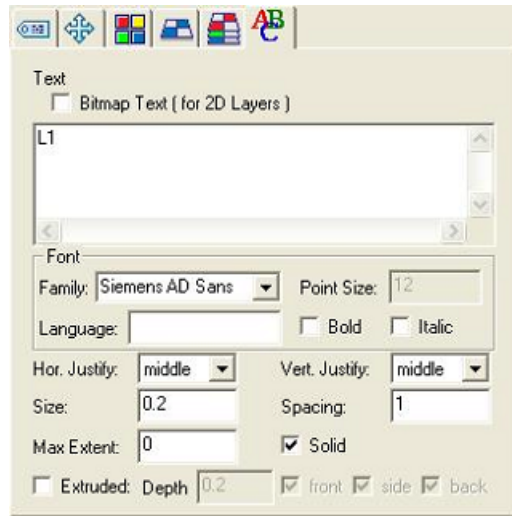


Figure C-3 Settings for text

8. To create hatching, use file `schnitt.png` as the texture. The lines always go from the bottom left to the top right at an angle of 45°. You should set the scaling to 3 in both dimensions. The rotation depends on the construction type and must be adapted manually.

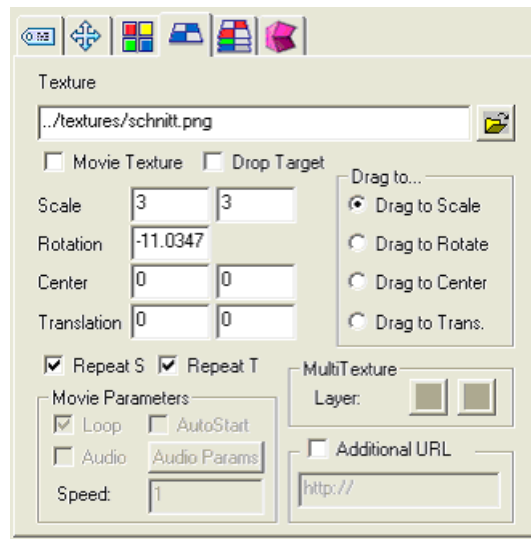


Figure C-4 Settings for hatching

9. We recommend the following sizes/dimensions for the blanks:
- Cylinder for turning operation:
Length = 5.5 ; Radius = 1.4 (cf. template *turning_blank.fxw*)
 - Cube for milling operation
x = 4.75 ; y = 3 ; z = 3 (cf. template *milling_blank.fxw*)

See also

XML commands (Page 218)

C.2.3 Importing graphics (models)

In Flux Studio, you can import external graphics (models). You can store frequently used models, such as tools, as .x3d or .hmi files centrally for further use as finished elements in other help screens. You will find a list of supplied modeling templates in Chapter Modeling templates (Page 217).

Complex objects can also be created with other modeling tools, such as Cinema4D, and then imported.

A description of how you can reuse these models is given below.

Import as inline

Flux Studio uses the object "Inline" for importing .x3d files. It allows you to link in external elements without multiplying the 3D data of these models.

Insert the object with menu item **Create -> Create Inline**. Then enter a file name in the object properties.

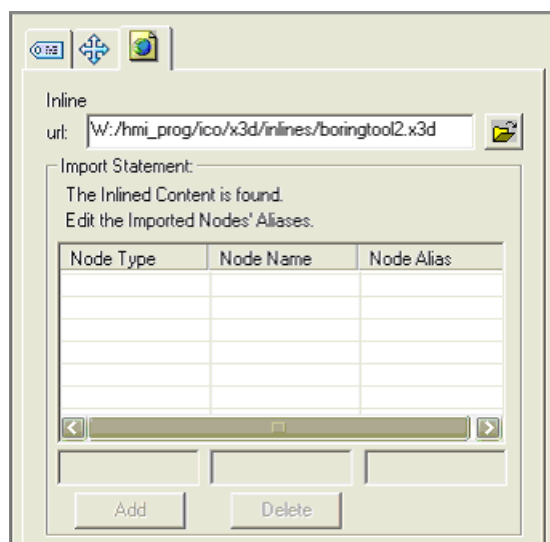


Figure C-5 Importing with object "Inline"

Importing 3D model data

Proceed as follows to import model data from a file.

1. Open dialog box "Flux Studio Accutrans3D Translation Utility" via menu item **File -> Import Other Format**.
2. Select a format from the "File Types" list box. For example, to import a Cinema4D file, select file type "3D Studio."
3. Then select a file with **Select Files** and start import.

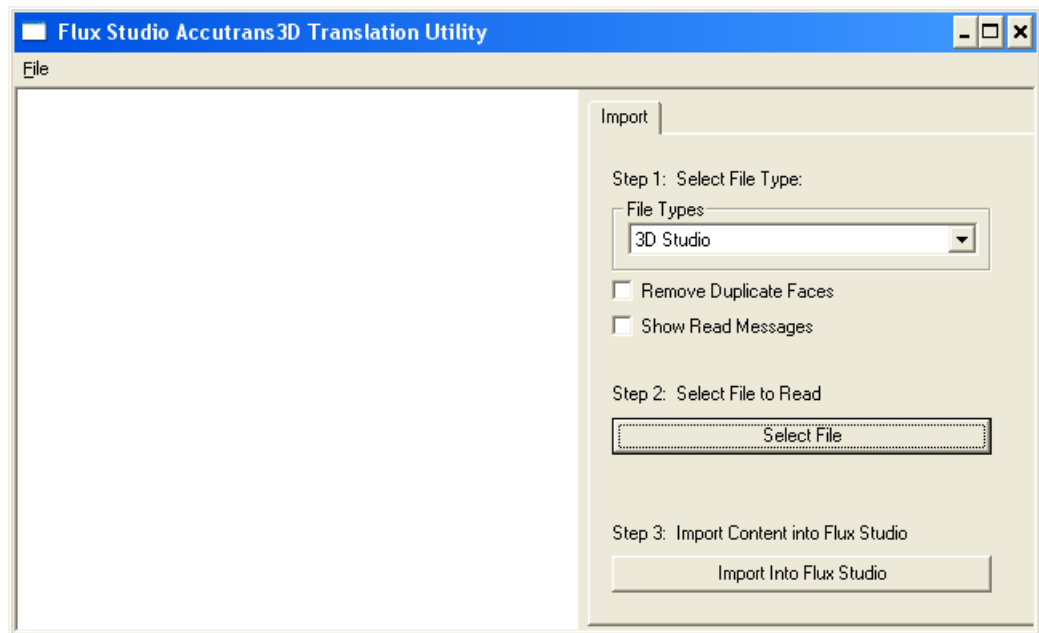


Figure C-6 Importing 3D model data

The model data from the Cinema4D file is inserted as a complete group. All cameras and any other information that is additionally imported should be deleted. Only the graphics elements of the Cinema4D file are required.

C.2.4 Modeling templates

In this chapter, you will find a list of graphics elements on which you can base your design, colors, and sizes. To maintain a uniform "look and feel" in the user interface, we recommend using the style of the templates or the templates themselves for creating your own graphics.

General

Template	Description
intersection_texture.png	Texture for an sectioned area
rapid_traverse_line_hori.fwx	Horizontal line to illustrate the tool traversing path during rapid traverse
rapid_traverse_line_vert.fwx	Vertical line to illustrate the tool traversing path during rapid traverse

Template	Description
feed_traverse_line_hori.fwx	Horizontal line to illustrate the tool path during feed
feed_traverse_line_vert.fwx	Vertical line to illustrate the tool path during feed
dimensioning_lines_hori.fwx	Horizontal projection lines
dimensioning_lines_vert.fwx	Vertical projection lines
z1_inc.fwx	Horizontal dimension line with the identifier Z1
x1_inc.fwx	Vertical dimension line with the identifier X1
3d_coordinate_origin.fwx	3D coordinate origin
3d_zero_point.fwx	3D zero point

Turning

Template	Description
turning_blank.fwx	The blank for turning technology: A simple cylinder
turning_centerline.fwx	Center line
turning_centerpoint.fwx	Coordinate origin to display the zero point in the WCS
turning_refpoint.fwx	Reference point, for example, for a machining operation
turning_machining_area.fwx	Boundary lines for the machining area

Milling

Template	Description
milling_blank.fwx	The blank for milling technology: A simple cube
milling_centerline.fwx	Center line
milling_refpoint.fwx	Reference point, for example, for a machining operation

C.3 XML commands

C.3.1 Overview

The scene description file (extension .xml) is required so that the X3D Viewer can access the graphics. In the scene description file, you assign the scene names that are called from the configuration to the times in the .x3d file (time sensor) at which the graphics are positioned.

C.3.2 Structure of the scene description file

The structure of the scene description file is shown below together with an explanation of the relevant XML commands:

Design and description

<!-- Treatment of the plane-dependent texts for turning -->

```
<TextPlane plane="G18_ZXY" />
```

Description

```
<TextPlane plane="G18_ZXY" />
```

Treatment of the plane-dependent texts (axis names) specifically for turning. If, for example, a text with the identifier "Z" is used (for G18 first geometry axis), the corresponding geometry axis identifier of the control will be shown in the help screen (for example, "Z").

<!--Text alignment -->

```
<TextPosition center='true' />
```

Description

```
<TextPosition center='true' />
```

Marking that the texts are centered. This is relevant to the display of the texts in rotated screens. We recommend using this setting.

<!--Adapting the tool speed -->

```
<ToolSpeedFactors planeSpeedFactor="0.4" rapidSpeedFactor="0.7"
reducedSpeedFactor="1.0"/>
```

Description

```
<ToolSpeedFactors
```

This entry can be added if the tool speeds are to be modified.

```
planeSpeedFactor="0.4"
```

Factor for the feedrate (0.4 = 40%)

```
rapidSpeedFactor="0.7"
```

Factor for rapid traverse (0.7 = 70%)

```
reducedSpeedFactor="0.1" />
```

Factor for a third speed (0.1 = 10%)

<!--Definition of an animation-->

```
<SceneKey name='BoringAnimation' masterRotationSpeed="-64.0"
maxRotAngle="45.0" begin-Time='0.110 endTime='0.118 view='camiso'
speedMaster='boringtool' Type="VIEW_3D_TURN_CYL">
```

Description

<SceneKey name='BoringAnimation'
masterRotationSpeed="-64.0"

Name of the animation

Direction of rotation and rotation speed of the tool.

The setting is optional.

maxRotAngle="45.0"

To avoid aliasing effects (tool seems to be turning in the wrong direction). The value is usually a fourth of the symmetry of the tool.

The setting is optional.

beginTime='0.110'

Start time of the animation on the time sensor

endTime='0.118'

End time of the animation on the time sensor

view='camiso'

Camera to be used for the animation

speedMaster='boringtool'

Name of the tool for the animation

type="VIEW_3D_TURN_CYL">

View type defines the view (see Section View type (Page 222))

<!-- Elements -->

```
<Element name='1' time='0.110' feed='rapid' dwell='2' />
<Element name='2' time='0.112' feed='rapid' />
<Element name='3' time='0.114' feed='rapid' />
<Element name='4' time='0.116' feed='plane' />
<Element name='5' time='0.118' feed='plane' />
```

Description

<Element name='1'

Specifies that it is the first element of the animation.

time='0.110'

Time of the first element on the time sensor

feed='rapid'

Traversing speed of the element

plane = machining feedrate

rapid = rapid traverse

reduced = reduced speed, slower than "plane"

dwell='2' />

Pause in the traversing motion of the animation. A rotating tool continues to rotate.

<!-- End of animation definition -->

`</SceneKey>`

Description

`</SceneKey>`

End of the animation definition

<!-- Existing animation as source for a new animation -->

`<SceneKey source='BoringAnimation' name='BoringAnimationRear' mirror='MirrorScreenX' />`

Description

`<SceneKey source="BoringAnimation"`

Name of an animation that is to use another animation as its source.

`name="BoringAnimationRear"`

Name of the new animation based on the source

`mirror='MirrorScreenX' />`

Mirroring in the direction of the X axis, output is in the new animation.

<!-- Definition of a static scene (screen), (four examples) -->

```
<SceneKey name='Default' time='0.026' view='camiso'
type="VIEW_3D_DRILL_CUT"/>
<SceneKey name='Cut' time='0.046' view='camside' type="VIEW_SIDE"/>
<SceneKey name='Zlink' time='0.056' view='camside' type="VIEW_SIDE"
highLightedGroup='dad_Zlink'/>
<SceneKey name='Zlabs' time='0.066' view='camside' type="VIEW_SIDE"
highLightedGroup='dad_Zlabs'/>
```

Description

`<SceneKeyname='Z1abs'`

Screen name

`time='0.066'`

Time of the screen on the time sensor

`view='camside'`

Camera to be used for the screen

`type="VIEW_SIDE"`

View type defines the view (see Section View type (Page 222))

`highLightedGroup='dad_Z1abs'/>`

Name of the group to be highlighted. The group has been called "Z1abs" in Flux Studio. The prefix "dad_" is automatically added by Flux.

<!-- End of xml file -->

C.3.3 Mirroring and rotations

Mirroring or rotation of the image or model is defined with the **mirror** command.

Description

<code>mirror="RotateScreenX"</code>	The screen is rotated around the X axis
<code>mirror="RotateScreenY"</code>	The screen is rotated around the Y axis
<code>mirror="RotateScreenZ"</code>	The screen is rotated around the Z axis
<code>mirror="MirrorScreenX"</code>	The screen is mirrored along the X axis
<code>mirror="MirrorScreenY"</code>	The screen is mirrored along the Y axis
<code>mirror="MirrorScreenZ"</code>	The screen is mirrored along the Z axis
<code>mirror="RotatePieceX"</code>	The model is rotated around the X axis
<code>mirror="RotatePieceY"</code>	The model is rotated around the Y axis
<code>mirror="RotatePieceZ"</code>	The model is rotated around the Z axis
<code>mirror="MirrorPieceX"</code>	The model is mirrored along the X axis
<code>mirror="MirrorPieceY"</code>	The model is mirrored along the Y axis
<code>mirror="MirrorPieceZ"</code>	The model is mirrored along the Z axis

All rotate and mirror options can be combined. A rotation angle must be defined for rotation.

Examples

```
mirror="RotatePieceZ=180"
mirror="RotatePieceZ=-90 MirrorScreenX"
mirror="RotateScreenY=90"
mirror="MirrorPieceX MirrorPieceY"
mirror="MirrorPieceZ RotatePieceX=-90"
```

C.3.4 View type

You define the views with the view type. The views are based on empirical values and rules that produce a sensible representation of the objects.

This ensures that the representation of the objects matches the coordinate system settings made in the user interface (MD 52000 \$MCS_DISP_COORDINATE_SYSTEM or MD 52001 \$MCS_DISP_COORDINATE_SYSTEM_2).

Description

<code>"VIEW_STATIC"</code>	Direct view without conversion
<code>"VIEW_3D_TURN_CYL"</code>	3D view for turning operations (cylinder)
<code>"VIEW_3D_MILL_CUBE"</code>	3D view for milling operations (cube)
<code>"VIEW_3D_DRILL_CUT"</code>	3D view for drilling operations (sectional view)

"VIEW_SIDE"	2D view for drilling operations (sectional view)
"VIEW_TOP_GEO_AX_1"	2D view from the direction of the 1st geometry axis
"VIEW_TOP_GEO_AX_2"	2D view from the direction of the 2nd geometry axis
"VIEW_TOP_GEO_AX_3"	2D view from the direction of the 3rd geometry axis

C.4 Conversion to hmi file

Note

The X3D files and their associated XML files are converted to HMI files when the HMI is powered up.

For each X3D file, a corresponding XML file of the same name must be created. To do this, you store the X3D files and XML files in directory `search path of HMI\ico\x3d\turning or milling`. You proceed in the same way with the .ts files.

C.5 Display in Create MyHMI /3GL

C.5.1 X3D Viewer

If you want to display particular help screens in a separate OA application you must integrate the X3D Viewer widget into its own OA application.

The X3D Viewer widget provides interfaces, which enable X3D contents to be presented in the HMI.

Class `SIX3dViewerWidget` is provided for displaying graphical scenes. The definition of the class can be found in the corresponding `slx3dviewerwidget.h` header file in the global GUI include directory `\hmi_prog\gui\include`.

C.5.2 Class `SIX3dViewerWidget`

The class provides a flexible widget that autonomously displays the contents of a model file specified during runtime and, if required, runs the animation.

The interface of the class comprises the Constructor, Destructor, and two methods for controlling the graphical output.

As a direct derivative of the Qt class `QWidget`, a much more expansive interface is available, which will not be described in any further detail here, (for example, `show()`, `hide()` and `resize(...)`). For more information, please refer to the Qt documentation).

C.5.3 Public methods

SIX3dViewerWidget (QWidget* pParent = 0)

Constructor of the X3D Viewer widget.

Parameters	Meaning
pParent	The parameter is passed on to the constructor of the QWidget.

~SIX3dViewerWidget ()

Destructor of the X3D Viewer widget.

Parameters	Meaning
-	-

C.5.4 Public slots

void viewSceneSlot (const QString& rsFileName, const QString& rsScene, const QString& rsAnimationScene, int nChannel, int nPlane, SStepTechnology nTechnology)

With the viewSceneSlot method, the X3D Viewer is instructed to load the rsScene static scene and the rsAnimationScene animated scene from the rsFileName file and display them alternately.

The static scene and then the animated scene are repeatedly alternately displayed for a fixed time.

If no static scene is specified, the animation will be shown immediately; an animated scene may also not have been specified.

Channel number, plane and technology are used to rotate the scenes to the correct position (depending on the set machine coordinate system).

Parameters	Meaning
rsFileName	Name of the file that contains the scenes to be displayed
rsScene	Name of the static scene.
rsAnimationScene	Name of the animated scene.
nChannel	Channel number

Parameters	Meaning
nPlane	Plane
nTechnology	Technology The following constants are defined for the SStep-Technology enumeration type: • SL_STEP_NO_TECHNOLOGY • SL_STEP_MILLING • SL_STEP_TURNING • SL_STEP_SURFACE_GRINDING • SL_STEP_CIRCULAR_GRINDING

void viewSceneSlot (const QString& rsFileName, const QString& rsScene, const QString& rsAnimationScene)

A simplified form of the viewSceneSlot method with which the X3D Viewer can be instructed to load and display the rsScene static scene and/or the rsAnimationScene animated scene from the rsFileName file.

Parameters	Meaning
rsFileName	Name of the file that contains the scenes to be displayed.
rsScene	Name of the static scene.
rsAnimationScene	Name of the animated scene.

C.5.5 Libraries

In order to be able to apply the X3D Viewer in your own projects, the list of library dependencies must be extended by the 'slx3dviewer.lib' entry.

C.5.6 Implementation example

You will find an implementation example in the package Create MyHMI/3GL under **\examples\GUIFrameWork\SIExGuiX3D**.

C.5.7 Machine data

Display MD 9104 \$MM_ANIMATION_TIME_DELAY

Time delay in seconds until the animation is started for help screens. The setting does not apply to help screens that are only animated.

The setting applies globally to all animations of SINUMERIK Operate.

C.5.8 Notes about use

- The animation is interrupted if the X3D Viewer widget is hidden. If this happens, there is no need to choose a scene selection without animation.
- You should avoid frequent or multiple instantiation of the X3D Viewer widget as this reduces performance and memory capacity. The use (implementation) of an X3D Viewer Singleton is recommended in such application scenarios.
- The X3D Viewer can also be used in signal slot concepts.
- If an error occurs (for example, a file not found or unknown scene name), the X3D Viewer widget opens a message area. This area is automatically hidden again when another help screen is activated.

C.6 Display in Run MyScreens

This section is intended for experienced "Run MyScreens" developers. Necessary background knowledge is provided in the associated documentation.

In addition to using screens in .bmp or .png format, animated help screens can also be displayed with the X3D Viewer.

To integrate help screens, use the Run MyScreens interface in the usual way.

To output images in .bmp or .png format, XG0 is entered in the definition of a dialog box in the Attributes section or the parameter is left empty. To integrate X3D help screens in a dialog, you must enter **XG1** in the attributes.

```
//M(MASK_F_DR_O1_MAIN/$85407////52,80/XG1)
```

The following parameters are required to activate the individual help screens; their meaning is described in Section 5.3 Public slots:

1. File name
2. StaticScene (optional)
3. AnimationScene (optional)
4. Technology (optional)
5. Plane (optional)

The parameters are combined into a character string in this order, separated by a comma.

```
Hlp = "File name,StaticScene,AnimationScene,Technology,Plane"
```

Examples

- The default help screen can be set with the specified variable Hlp.
In the following example, the "MyAnimation" animation is output from the "MyDlgHelp.hmi" file. No StaticScene is specified, i.e. no static scene is output. No specifications are made for the Technology and Plane, i.e. default values are used.

```
Hlp = "MyDlgHelp.hmi,,MyAnimation"
```

- The hlp property can be set on a specific help screen for the individual parameters of the input screen.
In the following example, the static "MyParam" scene and the "MyAnimation" animation are output from the "MyDlgHelp.hmi" file in the G17 plane. No specification is made for the Technology, i.e. a default value is used.

```
VarMyParam.hlp = "MyDlgHelp.hmi,MyParam,MyAnimation,,G17"
```


Glossary

Access level

Graduated system of authorization, which makes the accessibility and utilization of functions on the operator interface dependent on the authorization rights of the user.

Array

An array can be used to organize data of a standard data type stored in the memory in such a way that it is possible to access the data via an index.

Attribute

Characteristic that assigns specific → Properties to an object (→ Dialog or → Variable).

Column index

Column number of an array

Configuration file

File, which contains definitions and instructions that determine the appearance of → Dialogs and their → Functions.

Definition lines

Program section in which → Variables and softkeys are defined

Dialog

Display of the → User interface

- **Dialog-dependent softkey menu**
Softkey menu, which is called from a newly configured dialog.
- **Dialog-independent softkeys**
Softkeys, which are not called from a dialog, i.e., start softkey and softkey menus, which the user configures before the first, new dialog.

Editor

ASCII Editor with which characters can be entered in a file and edited.

Event

Any action, which initiates execution of a → Method: Input of characters, actuation of softkeys, etc.

Group

Reload unit for → Configuration file

Help variable

Internal arithmetic variable to which no → Properties can be assigned and is not, therefore, visible in the → Dialog.

Hotkeys

6 keys on OP 010, OP 010C and SINUMERIK keyboards with hotkey blocks. Pressing the keys selects an operating area directly. As an option, 2 additional keys can be configured as hotkeys.

Input/output field

Also I/O field: for inputting or outputting variable values.

Interpreter

The interpreter automatically converts the defined code from the → Configuration file into a → Dialog and controls its use.

Line index

Row number of an array

Menu tree

A group of interlinked → Dialogs

Method

Programmed sequence of operations executed when a corresponding → Event occurs.

Parameter

Parameters are variable elements of the programming syntax and are replaced by other words/symbols in the → Configuration file.

PI service

Function which, on an NC, executes a clearly defined operation. PI services can be called from the PLC and the HMI system.

PLC hard key

PLC hard keys are provided via the PLC interface of the HMI software, just like hotkeys. The functions, which can be triggered via the user interface, can be configured.

They take the form of MCP keys or evaluations of PLC signal logic operations in the PLC user program. For this reason, they are referred to as "virtual keys".

Programming support

Provision of → Dialogs to assist programmers in writing → Parts programs with "higher-level" components

Properties

Characteristics of an object (e.g. of a → Variable)

Recompile

NC code sections can be generated in a → Part program from input fields in → Dialogs in the → Programming support system. Recompilation is the reverse operation. The input fields used to generate a selected section of NC code are retrieved from the NC code and displayed in the original dialog.

Selecting

A program formulated in the NC language, which specifies motion sequences for axes and various special actions.

Simulation

Simulation of a → Parts program run without movement of the actual machine axes.

Softkey labels

Text/image on the screen, which is assigned to a softkey.

Softkey menu

All horizontal or all vertical softkeys

Start softkey

Softkey with which the first newly created → Dialog is started.

Toggle field

A list of values in the → Input/output field; check with toggle field: The value input in a field must be the same as one of the listed values.

User variable

Variables defined by the user in the → Parts program or data block.

Variable

Designation of a memory location, which can be displayed in a → Dialog by assigning
→ Properties and in which input data and the results of arithmetic operations can be entered.

Index

A

- Access level, 52
- Alarms
 - Language code, 203
- Array
 - Access mode, 167
 - Column index, 167
 - Compare mode, 167
 - Definition, 165
 - Element, 166
 - Line index, 167
 - Status, 170
- Attributes, 71

B

- Background color, 73

C

- Colors, 73
- Comparison operators, 93
- Conditions, 94
- Configuration file, 25
- Configuration syntax"; "Extended syntax, 36
- Configuration syntax"; "Screens, 36
- Configuration syntax"; "Softkey, 37
- Configuration syntax"; "Table columns, 37
- Configuration syntax"; "Variables, 36
- Configuring PLC softkeys, 193
- Constants, 93
- Custom widget
 - Automatic data exchange, 178
 - Definition, 175
 - Executing a method, 180
 - Interface, 176
 - Library, 175
 - Manual data exchange, 179
 - Reading and writing properties, 179
 - Responding to a custom widget signal, 182

D

- Default setting, 70
- Defines the softkey menu, 51

Dialog

- Definition, 39
- Definition block, 40
- Multiple columns, 47
- Properties, 41
- Dialog change mode, 135
- Dialog element, 45
- Display mode, 71
- Display option, 71
- DLL file, 127

E

- ELLIPSE (define circle), 164
- ELLIPSE (define ellipse), 164

F

- File
 - Copy, 114
 - Delete, 114
 - Moving, 116
- File formats
 - fxw, 211
 - hmi, 211
 - x3d, 211
 - xml, 211
- Flux Studio Web3D authoring tool, 212
- Focus control, 174
- Foreground color, 73
- Function
 - CALL (Subprogram call), 110
 - CLEAR_BACKGROUND), 113
 - CP (Copy Program), 114
 - CVAR (Check Variable), 112
 - Cyclic execution of scripts, 160
 - DEBUG, 121
 - DLGL (Dialog line), 120
 - DO LOOP), 158
 - DP (Delete Program), 114
 - EP (Exist Program), 115
 - EVAL (Evaluate), 126
 - EXIT, 122
 - EXITLS (EXIT Loading Softkey), 127
 - FCT, 127
 - File access, 118
 - FORMAT (string), 156
 - GC (Generate Code), 129

HMI_LOGIN, 132
HMI_LOGOFF, 132
HMI_SETPASSWD, 132
INSTR (String), 152
LA (Load Array), 133
LB (Load Block), 134
LEFT (string), 153
LEN (string), 152
LISTADDITEM, 124
LISTCLEAR, 124
LISTCOUNT, 124
LISTDELETEITEM, 124
LISTINSERTITEM, 124
LM (Load Mask), 135
LS (Load Softkey), 136
MIDS (string), 154
MP (Move Program), 116
MRDOP, 109
MRNP (Multiple Read NC PLC), 139
Overview, 108
PI_START, 141
RDFILE, 118
RDLINEFILE, 118
RDOP, 109
Reading and writing drive parameters, 109
Recompile NC code, 146
Recompile without comment, 148
REPLACE (string), 154
RESIZE_VAR_IO, 143
RESIZE_VAR_TXT, 143
RETURN (Back), 145
RIGHT (string), 153
RNP (Read NC PLC Variable), 142
SB (Search Backward), 150
SF (Search Forward), 150
SP (Select Program), 117
START_TIMER, 160
STOP_TIMER, 160
STRCMP (string), 155
STRINSERT (string), 155
STRREMOVE (string), 155
SWITCH, 139
TRIMLEFT (string), 156
TRIMRIGHT (string), 156
UNTIL, 158
WDOP, 109
WHILE, 158
WNP (Write NC PLC Variable), 142
WRFILE, 118
WRLINEFILE, 118

G

Generates NC code, 129
Graphic text, 70
Grid → Table, 170

H

H_SEPARATOR (define horizontal dividing line), 165
Help display, 73
Help file, 74
Help variable, 62

I

I/O field with integrated unit selection, 74
Image as short text, 66
Import
 Graphics (models), 215
Input mode, 72

L

Language code, 203
Limits, 70
LINE (define line), 163
List field, 66
Long text, 70

M

Machine data, 225
Machining step support, 148
Master dialog, 135
Mathematical functions, 92
Menu tree, 27
Method
 ACCESSLEVEL, 96
 CHANGE, 96
 CHANNEL, 98
 CONTROL, 98
 LANGUAGE, 100
 LOAD, 100
 LOAD GRID, 137
 OUTPUT, 102
 Overview, 96
 PRESS, 103
 PRESS(ENTER), 104

PRESS(TOGGLE), 104
 RESOLUTION, 105
 RESUME, 105
 SUSPEND, 106
 UNLOAD, 101

N

NC variable
 Read, 142
 Write, 142
 Numerical format, 77

O

Online help, 58
 Operator
 Bit, 94
 Mathematical, 91

P

Password dialogs, 48
 Password input mode (asterisk), 69
 PI services, 108
 PLC variable
 Read, 142
 Write, 142
 Position
 Input/output field, 73, 80
 Short text, 73, 80
 Progress bar with two color changes, 67
 Progress bar without color change, 68
 Protection levels, 202
 Public slots, 224

R

RECT (defining a rectangle), 163
 Registers
 Exchanging data, 144
 Status, 145
 Value, 144

S

Short text, 70
 Signal color"; "progress bar, 73
 Signal values"; "progress bar, 70
 SIX3dViewerWidget, 223

Softkey
 Assign properties, 51
 Properties, 53
 Start softkey, 27, 28
 Strings, 80
 Sub-dialog, 135
 Subprogram, 108
 Block identifier, 111
 Call, 110
 cancel, 145
 Variable, 111
 System colors, 202
 System variable, 63, 73

T

Table
 Defining columns, 173
 Definition, 170
 Programming, 172
 Text, 70
 Toggle field, 65, 70, 78
 Toggle symbol, 71
 Tooltip, 70, 72
 Trigonometric functions, 92

U

Unit text, 70
 Update rate, 71
 User variable, 73

V

V_SEPARATOR (define vertical dividing line), 164
 Variable
 calculating, 62
 Change property, 74
 Check, 112
 CURPOS, 82
 CURVER, 82
 End, 122
 ENTRY, 83
 ERR, 83
 FILE_ERR, 84
 FOC, 85
 Parameter, 69
 S_ALEVEL, 86
 S_CHAN, 87
 S_CONTROL, 87
 S_LANG, 88

- S_NCCODEREADONLY, 89
- S_RESX, 89
- S_RESY, 89
- Variable status, 61
- Variable type, 69
 - INTEGER, 75
 - VARIANT, 76
- Variable value, 61

W

- Write mode, 73

SIEMENS

SINUMERIK

SINUMERIK 840D sl NCU operating system (IM7)

Commissioning Manual

System settings

1

Configuring the system

2

Service and diagnostics

3

Appendix

A

Valid for:

CNC software for 840D sl/ 840DE sl Version 4.5 SP2

03/2013

6FC5397-1DP40-3BA1

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

 DANGER
indicates that death or severe personal injury will result if proper precautions are not taken.
 WARNING
indicates that death or severe personal injury may result if proper precautions are not taken.
 CAUTION
indicates that minor personal injury can result if proper precautions are not taken.
NOTICE
indicates that property damage can result if proper precautions are not taken.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

 WARNING
Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Table of contents

1	System settings	5
1.1	Delivery condition of the system	5
1.2	Meaning of switch positions	7
1.3	System booting	8
1.4	Displays during system booting	9
2	Configuring the system	13
2.1	Ethernet interfaces of the NCU	13
2.2	How to determine the IP address of the NCU	14
2.3	Configuration file "basesys.ini"	15
2.3.1	Section [ExternalInterface]	16
2.3.2	Section [InternalInterface]	18
2.3.3	Section [IBNInterface]	22
2.3.4	Section [SNMP]	23
2.3.5	Section [DCP]	24
2.3.6	Section [LLDP]	25
2.3.7	Section [LinuxBase]	26
2.3.8	Example: Configuration file 'basesys.ini'	29
2.4	Service Commands	35
2.4.1	Using service commands	35
2.4.2	Syntax for the Actions	36
2.4.3	Generic elements	37
2.5	Description of the actions	39
2.5.1	Help	39
2.5.2	Check-cf	40
2.5.3	Clear	40
2.5.4	Closeport	41
2.5.5	Connect	41
2.5.6	Disable	43
2.5.7	Disconnect	44
2.5.8	Distribute	44
2.5.9	Enable	45
2.5.10	Openport	47
2.5.11	Port status	48
2.5.12	Restart	48
2.5.13	Restore	49
2.5.14	Save	50
2.5.15	Show	52
2.5.16	Start, Stop	56

3	Service and diagnostics	59
3.1	Creating a service system.....	59
3.1.1	How do you create a service system for the NCU?	59
3.1.2	This is how you operate the service system	61
3.1.3	Diagnostics functions	62
3.1.4	This is how you connect to a network drive	64
3.2	Backup license	65
3.2.1	This is how you backup the license key.....	66
3.3	Backup data via the service menu	68
3.3.1	This is how you backup user data.....	68
3.3.2	This is how you restore user data	70
3.3.3	This is how you create a complete backup of the CompactFlash Card.....	71
3.3.4	This is how you install a complete system backup	73
3.4	Backing up data via the service shell.....	75
3.4.1	This is how you save data on a service system.....	75
3.4.2	This is how you restore data from the service system.....	76
3.4.3	This is how you save data on a network drive	76
3.4.4	This is how you restore data from the network drive	78
3.5	Updating the software	80
3.5.1	This is how you update the firmware	80
3.5.2	This is how you update the software.....	82
3.6	System diagnostics	84
3.6.1	Performing system diagnostics	84
3.6.2	This is how you change the system settings.....	86
3.6.3	This is how you check the CompactFlash Card.....	87
3.6.4	Calling the VNC Viewer.....	89
3.6.5	WinSCP and PuTTY	91
A	Appendix.....	93
A.1	Abbreviations	93
	Glossary	95
	Index.....	97

System settings

1.1 Delivery condition of the system

Scope of validity

This document is valid for all systems that are supplied with Linux as operating system on the CompactFlash Card, such as all NCUs of the type NCU 7x0.

Directory on the CompactFlash card

A Linux partition as well as a FAT partition are available on the CompactFlash Card. The memory space of the CompactFlash Card is occupied mainly by the Linux-EXT3 partition containing the system software and the user data. A 2MB large FAT partition still exists but only for internal administration purposes.

When the system is in operation, the following directories (this is a selection and not a complete list) are available in the EXT3 partition on the CompactFlash Card:

Directory	Use
/siemens	Reserved for Siemens system software
/addon	Reserved for Siemens add-on software
/oem	Additional software and configurations of the machine manufacturer
/user	- Storage of user data - Configuring the HMI application - Data that are created when commissioning the system using the HMI application.
/system	Linux operating system
/user/system/etc	File basesys.ini (modifications possible)
/user/common/tcu	TCU configuration files
/var/log/messages	System log file (same as event.log under Windows)

Note

Files in the directory under /user always have priority over files with the same name in the directory /oem → /addon → /siemens.

Partitioning of the memory

The memory of the CompactFlash Card is purely logically partitioned, i.e. the user can in principle assign the memory up to the end of the physical limit. An alarm under LINUX is triggered if less than 4 MB free memory is available.

Directory	1 GB card	8 GB card
/oem	100 MB	1 GB
/addon		
/user	256 MB	2 GB

Preset users

The following users have already been set up:

Users	Access level	Target group
Operator	Key switch position 0	Operators, users
Operator1	Key switch position 1	Operators, users
Operator2	Key switch position 2	Operators, users
Operator3	Key switch position 3	Operators, users
user	Password: CUSTOMER	Operators, users
service	Password: EVENING	Service personnel
manufact	Password: SUNRISE	Machine manufacturer

1.2 Meaning of switch positions

Overview

The NCU has two twist buttons in the lower section of the front panel.

NCK commissioning switch



SVC/
NCK

PLC mode switch



PLC

Rotary switch SVC/NCK

The switch positions have the following significance:

Switch position	Operating mode of the NCK
0	Normal boot-up NCK
1	NCK boot-up with default values (= memory reset)
2	The NCK (and PLC) starts up with the data that was saved at the last shutdown.
7	Debug mode (NCK is not started.)
8	IP address of the NCU is displayed on the seven-segment display.
All others	Not relevant

PLC twist button

The switch positions have the same significance as for a SIMATIC S7-CPU:

Switch position	Operating mode of the PLC
0	RUN
1	RUN (protected mode)
2	STOP
3	Memory reset (MRES)
All others	Not relevant

1.3 System booting

Sequence

To ensure unproblematic booting of the NCU, the CompactFlash Card must be inserted.

When the NCU is booting up, visual information on the current operating system is provided using the following displays:

- The RDY-LED flashes slowly orange when the CF card is accessed.
- During booting, the 7-segment display outputs different codes that indicate, for example, when the BIOS is started, when the CompactFlash Card is accessed, etc.

When the booting has been completed successfully, the following is displayed:

- The PLC LED lights up green.
- The 7-segment display shows "6." with a flashing dot.
- The RDY-LED lights up green. All other LEDs are not illuminated.

Performing a reset operation

The reset button is located behind the blanking plate of the NCU.

A reset operation resets the entire system and requires a system restart. This is comparable to a "Power On reset" except that the 24 V power supply does not have to be switched off.

Bootting for servicing purposes

For service or diagnosis purposes, the NCU can be booted from a service system, the Emergency Boot System.

See also

How do you create a service system for the NCU? (Page 59)

1.4 Displays during system booting

Behavior of the LEDs when booting

Of the LEDs on the front panel of the NCU, only the RDY LED and its status is important when booting up the NCU.

LED: RDY	LED designations: RUN STOP SU/PF SF DP1 DP2 OPT	State/phase	7-segment display
Orange	Orange	BIOS boot	Post codes, see table: Load BIOS
Slowly flashing orange (0.5 Hz)	Set by PLC/option module	Booting up the kernel Phase 2: Initialization of the driver	1
Slowly flashing orange/green (0.5 Hz)	Set by PLC/option module	Booting the basic system Phase 3: Initialization of the basic system	See table: Load operating system
OFF	Set by PLC/option module	NRK/NCK outputs	See table: NRK/NCK outputs
green/red flashing (0.5 Hz)	Set by PLC/option module	Basic system shutdown	OFF
red	Set by PLC/option module	Kernel shutdown	OFF
red (5 Hz)	Set by PLC/option module	SINAMICS fault	---

SINAMICS faults and alarms, also see: SINAMICS S120/S150 List Manual (LH1)

Load BIOS

7-segment display	Meaning
00	Before loading the Master Boot Record (MBR) from the CompactFlash Card
01	Before starting the Master Boot Record (MBR) codes
02	Master Boot Record (MBR) code started
03	Second stage of the boot loader started, before reading the configuration file
04	Before loading the kernel
05	Before starting the kernel

Booting the basic system

7-segment display	Meaning
1 (begin)	Basis initializations
2 (prepf)	Cleaning up and preparing the file system
3 (hostname, loopback)	The name is set and the loopback interface configured.
4 (syslog)	Starting the system logging Starting the CFS class 1, provide the network interfaces.
5 (network)	Initializing the network interfaces and the firewall
6 (NCU:rtai, TCU:tcuconfig)	Starting the time server (ntpd)
7 (lsh)	Starting the SSH server (if required, generate a host key if still not available).
8 (NCU:ftpd, TCU:mtouch)	Starting the FTP server
9 (NCU:tcuservices, TCU:usbexport)	Starting the TCU services (TFTP, VNC)

NRK/NCK outputs

7-segment display	Meaning
1	Not applicable.
2	Not applicable.
3	Debugger was initialized.
4	NRK operating system was successfully activated.
5	NRK operating system has booted, init task is being processed (→ the application boots).
6	Initialization was successfully performed; the control is in cyclic operation.
6.	with flashing point: The control is in cyclic operation and the cycles are active.
F	Internal error: Can only be read via additional diagnostics.
1 or 2	CompactFlash Card and SRAM data do not match. To resolve this problem, a general reset is required by booting with switch position 1 or 2.
Lxx=yyy	Internal error when starting NCK, "xx" is the line number in the mcsystem.ini, "yyy" is a unique error code for the code position.

System error

LED: RDY	LED designations: RUN STOP SU/PF SF DP1 DP2 OPT	Meaning	7-segment display
Rapidly flashing red (2Hz)	Set by PLC/option module	Permanent error: The system is stopped.	see the following table
Rapidly flashing red/orange (2 Hz)	Set by PLC/option module	Temporary error: The system may continue to run with restricted functions.	

7-segment display	Meaning
Permanent error:	
C	"crash": The operating system crashes (this approximately corresponds to a Blue Screen for Windows); details can be found in the system log file.
P	"partition": Error when repartitioning
Temporary error:	
E	"error": A read or write error has occurred on the CompactFlash Card (an additional point is displayed for a write error). It is possible that the CompactFlash Card is defective and should be replaced.
F	"full": The CompactFlash Card is too full, so it is probable that not all of the services will be able to be successfully started. When booting, this error is displayed for 1 minute; the system then continues to boot, however, it can be assumed that problems will occur.

Configuring the system

2.1 Ethernet interfaces of the NCU

Supplementary conditions

For the operation of an NCU:

- No more than one NCU may be operated as a DHCP server on the system network.
- An external mouse is recommended when using the System Network Center.

Configuration of the interfaces

The following connections can be established via the Ethernet interfaces:

Interface	Labeling	Internal name	Terminal settings
Ethernet IE1/OP	X120	(Eth 2)	Connection to the system network with preset IP address 192.168.214.1 with subnet screen form 255.255.255.0 and active DHCP server for SINUMERIK
Ethernet IE2/NET	X130	(Eth 1)	Connection to company network as standard DHCP client
Ethernet	X127	(lbn 0)	Service terminal with fixed IP address 192.168.215.1 and fixed subnet screen form 255.255.255.224 with active DHCP server

Network interface

The network interface is an interface that enables network communication. These are the Ethernet interfaces on the NCU.

VNC (Virtual Network Computing)

Virtual Network Computing is a software that displays the screen contents of a remote computer, with a running VNC server, on a local computer, with a running VNC viewer, and in return sends keyboard and mouse movements of the local computer to the remote computer.

2.2 How to determine the IP address of the NCU

NCU in the system network (X120)

On X120, the NCU is pre-selected as required: No settings are necessary.

NCU in the company network (X130)

On X130, the NCU is set to the address reference via DHCP. If the company network has a DHCP server no further settings are required.

Example: Determining the IP address

There are several possibilities for determining the IP address that the NCU obtained on the company network, and these possibilities are listed below:

- If the NCU boots in switch position 8, it shows the IP address on X130 on the 7-segment display.
- Once the NCU has booted successfully, open a service shell on the TCU and execute the following command to obtain the desired information:

```
SC SHOW IP
```

If the company network has no available DHCP server or if this cannot be used, you are given the option to set a fixed address for the NCU on X130 (in accordance with the addresses already used on the company network).

The following command displays the IP address at X130:

```
SC SHOW IP -X130
```

Note

In order to change network settings, for example the IP address at X130 and X120, SINUMERIK Operate dialogs are available in the diagnostics operating area.

See also

Generic elements (Page 37)

2.3 Configuration file "basesys.ini"

Storage path

For a Linux basic system the original file can be found on the CompactFlash Card under the path /system/usr/etc. OEM versions are stored under /oem/system/etc, as well as user versions under /user/system/etc.

The following additional options are available to make settings in the basesys.ini file:

- In SINUMERIK Operate, the values of the individual variables can be viewed and changed in the "Commissioning" operating area.
- The "System Network Center" program, is called with the "sc start snc" command. To operate it, you will need a mouse.

Use

Settings are made in the basesys.ini file that specify the behavior of the system in the network.

To edit the basesys.ini file:

- Comments are introduced by ';' or '#' at the beginning of the line, and span the width of the line. Empty lines are also handled as comments.
- The sections that start with a "[NAME]" line are ignored by the basic system itself, but are sometimes used by the HMI.
- Variable definitions are written in the form "NAME=VALUE". Blanks are permitted around the '=' character. The value may also be enclosed in double quotation marks, but this is optional.

Note

Suitable editors for Linux

In most Linux system files, lines may only be ended with LF, and not with CRLF as in Windows. Please take note of this when selecting an editor. The HMI application editor under "Commissioning" is suitable.

In the Linux operating system, the UNIX editor vi is available.

Please take note that the Linux operating system is case-sensitive.

Reference

Commissioning Manual "SINUMERIK Operate" (IM9)

"Operator Components and Networking" Manual

2.3.1 Section [ExternalInterface]

Description

In this section, the parameters and settings for the external Ethernet interface are defined. For an NCU7x0, this is X130 (Eth1). If the external interface is in DHCP mode, i.e. the value of ExternalIP is empty or not defined, then all parameters listed here, with the exception of "DHCPClientID", are accepted by the DHCP server as long as the server provides a value for the parameter.

DisableCompanyNet

If this variable is set to 1, the X130 interface to the company network is not initialized and is thus not available.

Value: Empty, 0 or 1
Default: Empty

ExternalIP

If ExternalIP is not empty, the fixed IP address specified there is used on the company network interface. The ExternalNetMask then also needs to be set, as well as the gateway, name servers, host name, and domain if required.

If ExternalIP is not set or is empty, a DHCP client is started on the interface.

Value: IP address
Default: Empty

ExternalNetMask

ExternalNetMask must be set together with ExternalIP, in order to define the size of the network.

Value: Subnet mask
Default: Empty

Gateway

If the value of gateway is not empty, the host specified there is used as the default gateway, which means that all IP packages that cannot be directly assigned are sent here for routing.

If no gateway is specified, only the networks that are directly connected are accessible.

Value: IP address
Default: Empty

Nameservers

If DNS name servers are specified here, then these must be used to resolve symbolic host names, i.e. at most of the points where an IP address is expected, a computer name can also be used instead.

The name server setting is also sent to your DHCP client (TCU, PG) via the DHCP server of the NCU, so that this can also work with symbolic names.

Value: List of IP addresses (separated by a space)
Default: Empty

Timeservers

Here you can specify a list of NTP servers (Network Time Protocol) that can be used by the NTPD on the NCU for time synchronization. The status of NTPD is output in the system logfile /var/log/messages.

Value: List of IP addresses (separated by a space)
Default: Empty

Hostname

Here you can define a name for the local host. This manually assigned name takes priority over all other names.

The host name is determined in the following order:

- Host name from the basesys.ini file, if set
- A name received from the DHCP server (if DHCP client mode and a name have been supplied), the result of a reverse DNS lookup, i.e. which name belongs to the IP received (if nameservers are defined)
- A default name ("NONAME_...")

Because the host name is also used as DNS name it must satisfy the corresponding requirements of the RFC:

- ASCII letters (a-z, A-Z), numbers (0-9) and '-'
- max. 63 characters

Value: Name
Default: Empty

Domain

These variables can be used to configure the DNS domain. The effect is that names in this domain can be resolved without qualification (e.g. if the domain is "test.local", you can also specify the name "computer1.test.local" as "computer1").

Value: Name
Default: Empty

DHCPClientID

With these variables, you can influence which ClientID the DHCP client presents to its server. This ID can be used by the server to assign certain parameters to the client, for example a static IP address.

Usually, the MAC address of the Ethernet interface is used for this purpose. This is also the default setting. Alternatively, you can also use the host name ("@NAME"), which must then be defined in the basesys.ini, in order to be recognized before the DHCP request. It is also possible to use an arbitrary string for ClientID.

Value: @MAC, @CFID, @NAME or any string
Default: @MAC

2.3.2 Section [InternalInterface]

Description

In the section [InternalInterface] the data is specified for synchronization of the DHCP server in the system network.

DisableSysNet

If this variable is set to 1, the X120 interface to the plant network is not initialized and is thus not available.

Value: Empty, 0 or 1
Default: Empty

InternalIP

Use this variable to set the fixed IP address in the system network. InternalIP should always be used together with InternalNetMask.

Value: IP address
Default: 192.168.214.1

InternalNetMask

Specifies the subnet mask for InternalIP and should always be used together with it.

Value: Subnet mask
Default: 255.255.255.0

InternalIP_Alias

Use this variable to define an additional alias IP address for the interface of the system network (X120); this is practical in certain application cases.

Value: IP address
Default: Empty

InternalIP_Alias should always be used together with InternalNetMask_Alias.

InternalNetMask_Alias

This variable specifies the subnet mask for InternalIP_Alias and should always be used together with it.

Value: Subnet mask
Default: Empty

EnableDHCPD_SysNet

If this variable is set to zero, then the start of the DHCP server on the system network X120 is prevented.

Value: Empty, 0 or 1
Default: Empty

SyncModeDHCPD_SysNet

If this variable is not set on "OFF" then the DHCP servers synchronize themselves in the system network (X120) so that only one of them actively assigns addresses. This makes it possible to operate multiple NCUs or PCUs concurrently without having to adjust the network settings.

Value: ON_MASTER, ON_HIGH (= ON), ON_LOW, or OFF

Default: ON_HIGH

The DHCP servers that are not active go into "standby" mode; in this mode they regularly get the current address data and TCU data from the active server so that if the active server fails a standby server can take over the active role without data loss.

- With the **ON_MASTER** setting instead of **ON_HIGH** you can influence the synchronization in such a manner that the server with the "Master" setting is always the active server (for this it must be active on the network). This means that in the normal situation deterministically the same controller is always the DHCP server and you can find the actual address data (/var/etc/udhcp-eth0.leases) and the TCU data (/user/common/tcu/etc.) there. ON_MASTER should only be set for a single DHCP server in the system network; there should not be multiple masters.
- The **ON_LOW** setting assigns the DHCP server a lower priority than normal. It then becomes the active server if no server with ON_HIGH or ON_MASTER is found in the network.

Note

Compatibility with earlier versions

The priority level "ON_LOW" is used automatically if the DHCP server is switched off. Although such an NCU does have a (standby) DHCP server and keeps a backup of the lease data, normally, however, it does not become an active server.

InternalDynRangeStart

With the two variables InternalDynRangeStart and InternalDynRangeEnd, you can explicitly specify the range of IP address assigned by the DHCP server. The number band of the presetting should normally suffice.

Value: IP address

Default: First address in the system network + 10 or +2 (if ≤ 16 addresses)

InternalDynRangeEnd

Value: IP address

Default: Last address in the system network - 15 or -1 (if ≤ 16 addresses)

DHCPDNoMasterWait

If a synchronizing DHCP server has once seen a master server in the past, then it waits for synchronization until the time specified here elapses before it becomes the active server.

This additional pause makes it possible for the intended DHCP master to become the active DHCP server without displacement even if it is shortly switched on thereafter, or takes longer to boot than other controllers.

Value: Time in seconds
Default: 120

InternalDNSDomain

This is used to specify the Top Level Domain (TLD) name that is used in the system network. The DNS server of the NCU assigns names to the devices in the system network in this zone. Name requests for all other zones are forwarded to an external name server in the company network.

As presetting the "local" recommended by the RFC1035 is used for local networks, to avoid conflicts with globally defined domain names.

Recommendation: The preset should be maintained.

Value: Domain name (letters, numbers, '-', '_', max. 63 characters)
Default: local

DisableNATRouting

The default setting permits that a machine can access a company network from the system via the control using the NAT routing (= "Network Address Translation"). In this case, the NCU converts the source address of the packages to its own address, and handles the response packages in the inverse way.

If this variable is set to zero, the NAT routing of the NCU from the internal interfaces eth0 =X120 and ibn0 =X127 to the external network on eth1 =X130 is prevented. In order that the NAT routing is not simultaneously shut down for both internal networks, using the values "X120" or "X127", the system network or the commissioning network is selectively shut down.

Value: empty, 0, 1, X120 or X127
Default: Empty

2.3.3 Section [IBNInterface]

Description

The settings in this section affect interface X127.

EnableDHCP_IBNNet

If this variable is set to 1 then this prevents the start of the DHCP server at the commissioning interface X127.

Value: Empty, 0 or 1
Default: Empty

DisableIBNNet

If this variable is set to 0, then the commissioning interface X127 is not initialized and is thus not available.

Value: Empty, 0 or 1
Default: Empty

EnableSysNetToIBNForwarding

If this variable is set to 1, it is possible to transmit packages from the system network (X120) to X127. A firewall usually prevents this.

It is important to remember that there is no NAT involved, and the sender is responsible for ensuring (with a suitable routing entry, for example) that packages sent to the PG or service PC on X127 with the address 192.168.215.x only reach the NCU to which the device is connected.

Value: Empty, 0 or 1
Default: Empty

DisableIBNForwarding

If this variable is set to 1, the NAT transfer of packages from the service connection X127 to the system network (X120) is deactivated.

Value: Empty, 0 or 1
Default: Empty

2.3.4 **Section [SNMP]**

Description

Character strings delivered via SNMP (Simple Network Management Protocol) are entered here. This is a way of giving out information.

SNMPLocation

This string is used for the standard OID SNMPv2-MIB::sysLocation. It is possible to specify a location here, which can be subsequently called with an SNMP client.

Value: Any character string
Default: Empty

SNMPContact

This string is used for the standard OID SNMPv2-MIB::sysContact. It is possible to specify a contact address here, which can be subsequently called with an SNMP client.

Value: Any character string
Default: Empty

SNMPAutLocation

This string is used for the Siemens-specific OID automationSystemIdent.automationLocationTag. It is possible to specify an additional location here, which can be subsequently called with an SNMP client. This is similar to SNMPLocation in terms of the MIB description, although it is intended for entries relating to automation.

Value: Any character string
Default: Empty

SNMPFunction

This string is used for the Siemens-specific OID automationSystemIdent.automationFunctionTag. It is possible to specify a function designation here, which can be subsequently called with an SNMP client.

Value: Any character string
Default: Empty

SNMPStation

This string is used for the SINUMERIK -specific OID mcSinumerikMIB.mcSinumerikMiscStation. The value is the name of a station to which the device belongs. Devices with the same station name are therefore identified as being contiguous.

Value: Any character string
Default: Empty

See also

Example: Configuration file 'basesys.ini' (Page 29)

2.3.5 Section [DCP]

Description

The parameters in this section define the properties of the DCP protocol (Discovery and Basic Configuration Protocol) e.g. that are used in the menu item "Accessible participants" of STEP 7 is used.

InternalDcpEnabled

This can be used to switch-on and switch-off the DCP in the system network (X120); for the NCU, the value is also used for X127.

Value: 0 or 1
Default: 1

ExternalDcpEnabled

This can be used to switch-on and switch-off the DCP in the company network (X130).

Value: 0 or 1
Default: 0

2.3.6 **Section [LLDP]**

Description

The parameters in this section define the properties of the LLDP protocol (Link Layer Discovery Protocol), which is used by several applications for network diagnostics.

InternalLldpEnabled

This can be used to switch-on and switch-off LLDP in the system network (X120); for the NCU, the value is also used for X127.

Value: 0 or 1
Default: 1

ExternalLldpEnabled

This can be used to switch-on and switch-off LLDP in the company network (X130).

Value: 0 or 1
Default: 0

InternalLldpTLVsTxEnabled

Using this parameter, additional information in the LLDP packages are enabled at X120/X127, which are normally not included.

Value: Numerical value from 0 ... 15
Default: 0

The value is a bit field, i.e. the total value is obtained as a sum (total) of the specified numbers, if the corresponding information is to be sent:

- 1: port description
- 2: system name
- 4: system description
- 8: capabilities

ExternalLldpTLVsTxEnabled

Using this parameter, additional information in the LLDP packages is enabled at X130, which are normally not included.

Value: Numerical value from 0 ... 15

Default: 0

The value corresponds to parameter InternalLldpTLVsTxEnabled.

2.3.7 Section [LinuxBase]

Description

Additional setting possibilities of the Linux system are combined in this section.

Synchronizing the time

There are two different cases:

- If there is an external NTP server to act as a time server (entered in the basesys.ini file or via DHCP), then the PLC clock is synchronized via the Linux clock.
- If there is no external time server, the PLC clock will be the master for the Linux clock:

Note

With "memory reset" the time of the PLC clock remains intact and is not reset.

If the PLC is switched on in switch position 3 "MRES" then the time will be reset. In this case the valid time is taken over from the Linux system and does not need to be reset.

Timezone

The time zone set here is used by the system to convert UTC time (Universal Time Coordinated) into local time. The time zone is sent to all applications via the environment variable "TZ", and is taken into account by the libc function localtime().

The time zone influences all time entries in the basic system, particularly for the command "date", for "ls -l", and in the system log file /var/log/messages.

Value: Time zone description (see syntax and examples)

Default: UTC

Examples:

If the control is to synchronize with one of the listed time zones, the examples in the table can simply be copied.

Zones with daylight saving time/standard time:

Europe:	WET0WEST,M3.5.0,M10.5.0
	CET-1CEST,M3.5.0,M10.5.0
	EET-2EEST,M3.5.0,M10.5.0
USA:	EST5EDT,M4.1.0,M10.5.0
	CST6CDT,M4.1.0,M10.5.0
	MST7MDT,M4.1.0,M10.5.0
	PST8PDT,M4.1.0,M10.5.0

Zones without daylight saving time/standard time:

China:	CST-8
Japan:	JST-9

Syntax for the time zone description

In order to synchronize with a different time zone, the following syntax applies:

tz offset dst,Mmwd,Mmwd

Syntax (obligatory):	
tz offset	Time zone and difference from UTC: West of the zero meridian, the difference must be added; east of the zero meridian, the difference should be subtracted.
Syntax (optional, only if there is daylight saving time):	
dst,Mmwd,Mmwd	Daylight saving time zone with beginning and end
m ($1 \leq m \leq 12$)	m: Month
w ($1 \leq w \leq 5$)	w: Week of the month, i.e. w = 5 is the last week of the month
d ($0 \leq d \leq 6$)	d: Day of the week (from 0 = Sunday to 6 = Saturday)

LogFileLimit

The size of the system logfile `/var/log/messages` can be restricted with the help of this variable. By entering "Zero", the logfile has no size limitations.

The specified limit is no hard limit for reasons of efficiency but rather a sort of limit value. If the logfile reaches a limit value times factor 1.5, then it is reduced to around 75% of the limit value. A check is carried out every two minutes maximum to see if the file has become too large.

Value:	Size in KB
Default:	100 KB

FirewallOpenPorts

Here you can specify a list of the ports which is to be enabled in the firewall of the external network interface. For example TCP/102 (S7 communication) is always enabled. Logfile is "TCP" or "UDP", the port is the (numerical) port number of the service to be enabled. Several entries are separated by a space.

Value: List of logfile/port pairs

Default: Empty

NOTICE

Security risk

Note that opening the ports in the firewall can represent a security risk. Only enable the ports that you actually require.

Note that every accessible service can have security gaps.

DisableSubsystems

Using this variable, you can suppress mounting and starting of certain subsystems (CFS systems). Several subsystem names, too, can be specified - separated by blanks.

Value: List of CFS names (separated by a space)

Default: Empty

The names can either be simple CFS basic names (e.g. "nck") - ignoring all CFS systems with this name - or you specify an absolute path (e.g. /siemens/sinumerik/nck) referring specifically to this CFS system.

It is also possible to set "DisableSubsystems=all" whereby not a single CFS system can be mounted or started any more.

DisableNTPTimeSync

If this variable is set to 1, then the NTPD server (Network Time Protocol Daemon) for the time synchronization is not started and the time of day is not synchronized.

Value: 0 or 1

Default: 0

DisablePLCTimeSync

If this variable is set to 1, synchronization of the system time with the PLC time (in both directions) will not be executed.

Value: 0 or 1
Default: 0

EnableCoreDumps

If this variable is set as equal to 1, a logfile for processes is created and stored under `/var/tmp`. The protocol file contains a memory print of the process that has crashed or that has terminated.

Value: Empty, 0 or 1
Default: 0

CheckTCUforUpdatePackage

If this variable is set to 1, when booting, the USB interface at the front of the TCU is checked to identify any storage medium. If, when checking the update packages, type *.usz files created with Create MyConfig are found, then the installation of these update packages is started.

With the default setting zero, this check is skipped in order to save time.

Value: 0 or 1
Default: 0

2.3.8 Example: Configuration file 'basesys.ini'

Configuration file basesys.ini

The following defaults are applied to the template_basesys.ini file which is supplied in the directory `/siemens/system/etc.`:

```
; -----  
; Default Linux basesystem configuration  
; -----  
  
;  
; section ID is for Windows compatibility and is ignored ;)  
[ExternalInterface]
```

2.3 Configuration file "basesys.ini"

```
; If DisableCompanyNet is set to 1, the whole interface will
; be disabled.
;DisableCompanyNet=1

; If ExternalIP is set, you can force the external Ethernet
; interface to use a fixed IP address etc. instead of using DHCP
; if a Hostname is set, it even overrides one received by DHCP
;ExternalIP=210.210.210.210;ExternalNetMask=255.255.255.0;Gateway=210.210.210.1;Names
servers=210.210.210.1
210.210.210.2;Timeservers=210.210.210.3;Hostname=somename;Domain=example.com

; if ExternalIP is empty (default), DHCP is used with the following
; ClientID the default is "@MAC" to use the MAC address,
; alternatives are "@NAME" to use the hostname (Hostname above)
; "@CFID" to use the ID of the CF card if one is available
; (like on X120), or any other arbitrary string
;DHCPClientID=@MAC

[InternalInterface]

; If DisableSysNet is set to 1, the whole interface will
; be disabled
;DisableSysNet=1

; With InternalIP and InternalNetMask (both must be set together), you
; can change the address on the internal/TCU/automation/system net.
;InternalIP=192.168.214.1;InternalNetMask=255.255.255.0
; These two can define an alias IP for X120.
;InternalIP_Alias=192.168.216.1;InternalNetMask_Alias=255.255.255.0
; This defines a "name of station" for ProfiNet (ERTEC interface)
; for use with DCP, default is the normal hostname.
; Please be careful if a PN-Name can also be set by other means,
; for example by NCU-Link. Usually it is not needed to define
; a name here.
;PN_StationName="PN_IO"

; Setting EnableDHCPD_SysNet to 0 suppresses that a DHCP server is started on
; that interface
```

```
;EnabledDHCPD_SysNet=0

; Set synchronization of all DHCP servers in the
; system/TCU network (X120). Possible values are:
; OFF,
; or ON_LOW (low priority),
; or ON or ON_HIGH (normal priority),
; or ON_MASTER (highest priority),
; or ON_CLIENT_SYNC (starts DHCP client only, synchronization active),
; or ON_CLIENT_NO_SYNC (starts DHCP client only, synchronization
; inactive),
; ON_MASTER is used to make this machine deterministically the
; DHCP master server, but should be used only for one server
; in the network.
;SyncModeDHCPD_SysNet=ON

; This is the range of dynamic IPs given out by the DHCP server
; Defaults should be sensible
InternalDynRangeStart=192.168.214.10InternalDynRangeEnd=192.168.214.239
; If once a DHCP master was seen, the DHCP server waits this many seconds
; for the master to show up before trying to become the active server
;DHCPDNoMasterWait=120

; DNS domain name used for names in system/TCU network
; (default is "local" to conform with RFC)
InternalDNSDomain=local

; FixedDomain and FixedNameservers can be used to define a DNS domain and
; nameservers, if the DHCP/DNS server on system network is disabled
;FixedDomain=local
;FixedNameservers=192.168.214.1

; If DisableNATRouting is set to 1, the NCU won't forward
; from TCU or IBN net
; (X120 and X127, resp.) to external net (X130).
```

2.3 Configuration file "basesys.ini"

```
; If set to "X120" or "X127", forwarding will only be
; disabled from that interface.
;DisableNATRouting=1

[IBNInterface]

; If DisableIBNNet is set to 1, the whole interface will be disabled
;DisableIBNNet=1

; Setting EnableDHCPD_IBNNet to 0 suppresses that a DHCP server is started on
; that interface
;EnableDHCPD_IBNNet=0

; If DisableIBNForwarding is set to 1, the NCU won't do NAT routing
; from IBN network (X127) to system network (X120).
;DisableIBNForwarding=1

; If EnableSysNetToIBNForwarding is set to 1, the NCU forwards packets from
; X120 to X127. (Please note that the sending host must be able to route
; the packets to the NCU by its own means, the NCU just accepts and forwards
; packets to 192.168.215.x with this setting.)
;EnableSysNetToIBNForwarding=1

[SNMP]

; The following variables define strings which are delivered on
; certain SNMP requests.
; SNMPLocation and SNMPContact are reported for the standard OIDs
; SNMPv2-MIB::sysLocation and SNMPv2-MIB::sysContact, resp.
; SNMPAutLocation and SNMPFunction are used in the Siemens A&D specific
; AUTOMATION-SYSTEM-MIB as automationSystemIdent.automationLocationTag
; and automationSystemIdent.automationFunctionTag, resp.
SNMPLocation="not specified"
SNMPContact="not specified"
SNMPAutLocation="not specified"
SNMPLocation="not specified"
```

```
SNMPStation=""

[DCP]

; These variables en/disable the DCP protocol on X120/X127
; and X130, resp.
;InternalDcpEnabled=0
;ExternalDcpEnabled=1

[LLDP]

; These variables en/disable the LLDP protocol on X120/X127
; and X130, resp.
;InternalLldpEnabled=0
;ExternalLldpEnabled=1

[LinuxBase]

; Keep size of /var/log/messages around this value
; (not followed exactly for performance reasons)
LogfileLimit=102400

; Protocol/Port pairs to open in the firewall
; (e.g., TCP/5900, UDP/514, ...)
;FirewallOpenPorts="TCP/5900 TCP/102"

; If there is only one Ethernet interface, it's used by
; default as an external (company) network. Alternatively,
; usage as automation net (TCU boot support etc.) is possible
; by setting NetworkModel to "automation". (Only NCU, Service
; and PCU20A variants!)
;NetworkModel=automation

; DisableSubsystems can be used to skip certain CFSes (= subsystems)
; a list of multiple names (separated by spaces) is possible
; a simple name means all CFSes with this name, an absolute path
; (e.g., /siemens/sinumerik/nck) exactly this one CFS with
; the special value "ALL", all subsystems can be disabled
;DisableSubsystems=nck
```

2.3 Configuration file "basesys.ini"

```
; Directory for OEM netboot files (served by tftp to diskless clients)
NetbootDirectory=/oem/common/netboot

; Properties of local time zone: names, offset, start and end day
; Some examples:
;   Europe: WETOWEST,M3.5.0,M10.5.0; CET-1CEST,M3.5.0,M10.5.0; EET-
2EEST,M3.5.0,M10.5.0; USA: EST5EDT,M4.1.0,M10.5.0; CST6CDT,M4.1.0,M10.5.0;
MST7MDT,M4.1.0,M10.5.0; PST8PDT,M4.1.0,M10.5.0; China: CST-8; Japan: JST-
9Timezone=UTC

; If DisablePLCTimeSync is set, no time synchronization with
; PLC will happen
;DisablePLCTimeSync=1

; Set to 1 to enable coredumps in /var/tmp
EnableCoreDumps=0
```


2.4 Service Commands

2.4.1 Using service commands

Overview

The service command 'sc' is a tool used for performing a range of service tasks on a SINUMERIK NCU. The required action is written in the command line after the 'sc', e.g.: `sc help`

This action produces a list of all actions with a short description. After the action, further parameters or options can follow.

'sc' is available in both the NCU basic system and the service system (mostly in the form of a USB memory.) However, some actions are only useful in one of the two systems (NCU/Service). This is noted for each action.

See also

The most important terms and abbreviations are explained in the glossary.

Privilege levels

The 'sc' command can execute its actions using more privileges than are normally granted to the calling user. For example, starting or stopping subsystems requires root privileges, but 'sc' allows every user who belongs to the 'service' group to do this.

Each action of 'sc' is assigned a "privilege level". This is a user group to which the user must belong in order to execute the action. As the groups are hierarchically nested, members of "higher" groups can also use the action in question. For example, the group 'manufact' is above 'service', which means that members of the group 'manufact' can call all actions that require the privilege level 'service'.

The privilege level that the caller requires is noted for each individual action. The possible levels are (in ascending order):

- none
- operator
- user
- service
- manufact

Example:

An action with the privilege level 'user' can also always be executed by members of the groups 'service' and 'manufact'. Actions with privilege level 'none' can be called by all users.

If a user does not have the required privileges, the following error message is output:

```
Action 'ACTION' needs at least GROUP privilege level.
```

2.4.2 Syntax for the Actions

Description

'sc' in the command line is essentially not case-sensitive.

The following entries are therefore interpreted as identical:

sc help show

SC help SHOW

sc HeLp sHoW

In some cases, however, upper/lower case can make a difference, for example in file or user names. This is avoided wherever possible.

The conventions used are as follows:

- Names completely in upper case represent objects to be used depending on the situation.

Example: `sc help ACTION`

In this case, ACTION is to be replaced by the action for which you want a description. If written in lower case, however, the entry should be entered as specified.

- Square brackets indicate optional entries.

Example: `sc help [ACTION]`

In this case the specification of an action is optional, which means that you can enter an action, but do not have to. Square brackets may also be nested:

... [USERNAME[/DOMAIN]] ...

In this case, USERNAME and DOMAIN are both optional, but you can only enter a DOMAIN if you have also entered the USERNAME.

- Alternatives are separated with '|'.

Example: `sc start all|system|SUBSYSTEM`

This means that any of the following commands applies:

`sc start all`

`sc start system`

`sc start SUBSYSTEM`

In the latter case, SUBSYSTEM in upper case can be replaced by a concrete subsystem name.

- As a shortened form, alternatives can also be written in square brackets:

```
sc save [-full|-user] ...
```

Here you can use the option "-full" or "-user", or none at all.

- Options that begin with '-' can always be entered in any order.

For example, the notation could be interpreted to mean that "-force" must come after "-full" or "-user", but this is not necessarily the case:

```
sc save [-full|-user] [-force] FILENAME
```

2.4.3 Generic elements

Overview

This sections describes the syntax elements that are used by several actions.

Permitted interface designations

The names of network interfaces are used by "sc show ip" and "sc enable DHCPSEvr", for example.

As the input of an interface is optional, it is always introduced by a '-' character. The '-' is followed by the actual name. In most cases, multiple names are possible for the same interface.

Accepted names are:

System network port: "X120", "eth2", "tcu", "internal"
Company network port: "X130", "eth1", "factory", "extern"
IBN port: "X127", "ibn0", "pg"

Permitted subsystem designations

Subsystem names are entered when using "sc enable" and "sc start", for example. In most cases, the name of the subsystem is simply the name of the corresponding CFS, without the path and without the extension ".cfs". For example, for the CFS /siemens/sinumerik/nck.cfs, the relevant subsystem name is simply "nck".

It is also possible to use absolute paths (beginning with /) in a subsystem name. In the example above, you could also use "/siemens/sinumerik/nck" as a subsystem name. The difference between names with and without a specified path is that without the path, all CFS with this name are included, but if the path is specified, only this exact CFS is meant.

Subsystem

A subsystem is a CFS that not only contains a collection of files, but also executes a program, for example, at runtime. To do this, the CFS contains a script that is used to control the starting and stopping of this program.

For this reason, only administrators are permitted to set up NFS file systems, and NFS is usually only implemented in uniformly administrated environments. Exported file systems on the server are addressed directly on the server via their path.

CFS (Compressed File System)

A CFS (file extension ".cfs") is a compressed file system, similar to a zip file. It contains files and subdirectories that look like normal files on the controller at runtime. Files and directories contained in a CFS cannot be changed. They are decompressed at runtime as required.

NFS (Network File System)

NFS is the most common protocol for remote file systems in the world of Unix, and is also available for Windows. NFS is closely based on the Unix privilege model – each time a file is accessed, a UID and GID are supplied which the server then uses to decide whether the operation is permitted. The server relies on the client to provide the correct IDs.

Remote File System

A file system that is contacted over the network. The files are physically located on another computer in the network (the "server"), but appear locally the same as all other files. Operations performed on these files are sent via the network to the server, instead of being executed directly on a local storage medium (such as a hard drive or CompactFlash Card).

As a server usually exports more than one file system, a name for the required file system must also be entered in addition to the name of the server.

SMB (Server Message Block)

SMB is the underlying protocol of MS Windows file systems (also known as drives, releases, shares, etc.). SMB connections are always active in the context of a specific user, who must be known to the server. Exported file systems have a name (release name), by which they can be addressed. The client does not need to know the concrete path on the server.

2.5 Description of the actions

2.5.1 Help

Description

Syntax: `sc help [Action]`
Alternative names: `-h, --help`
Privilege level: `none`

The call of "sc help" without any additional action outputs a list of possible actions with a short description. If you enter an additional action, you receive a more detailed description for this action.

Examples:

```
sc help
```

All actions:

```
help [ACTION]
```

Print help about a specific or list all actions

```
restart
```

Reboot the machine

```
enable hmi|nck|SUBSYSTEM...
```

```
enable DHCPSPvr -INTERFACE
```

Enable HMI, NCK, or any other subsystem

```
[...]
```

```
sc help enable
```

```
enable hmi|nck|SUBSYSTEM...
```

```
DHCPSPvr -INTERFACE
```

Enable subsystem(s), like 'hmi', 'nck', and so on. A subsystem name is the name of the CFS containing it, without the '.cfs' extension.

This enables all CFSs with that name, but you can also use a full path (e.g. /siemens/sinumerik/nck) to enable just a specific CFS.

Another form is to enable the DHCP server on a network interface, for example 'enable DHCPSPvr -X120'.

2.5.2 Check-cf

Description

Syntax: `sc check-df`
Alternative names: `checkcf`
Privilege level: `user`

With this action, the CF card is checked reading to see whether it contains defective sectors. If errors occur, this is noted in the file `/var/log/messages`.

2.5.3 Clear

Description

Various actions can be performed with the "clear" command; only the "clear dhcp" command is relevant here.

clear dhcp

Syntax: `sc clear dhcp [-INTERFACE]`
Alternative names: `---`
Privilege level: `service`

This command clears any state of the DHCP server at the specified interface (default setting is the system network) and resets it to its initial state. This means that the Lease data is deleted: all IP addresses are re-assigned in the network and the server forgets having ever seen a master server.

This action is only appropriate on a machine on which the active DHCP server runs.

Note

The clearing of the Lease data does not delete the entire file, but only the data contained therein. This action also increments the version number, so that available standby DHCP servers can also perform the deletion.

2.5.4 Closeport

Description

Syntax:	sc closeport ID
Alternative names:	---
Privilege level:	service

Mode of operation

This command closes a port in the firewall opened previously with the "sc openport" command. This is only necessary when the port is to be closed manually before the time set with "sc openport" expires. Otherwise the port is closed automatically when the port service life expires.

ON parameter is the ID number of the firewall rule output by "sc openport".

2.5.5 Connect

Description

Syntax:	sc connect [-ro] SERVER:/PATH [MOUNTPOINT] sc connect [-ro] [-public] //[USERNAME[/DOMAIN] [%PASSWORD]@]SERVER/SHARE [MOUNTPOINT]
Alternative names:	mount
Authorization level:	none

This action makes a remote file system on a server available on the controller. This is enabled by linking the remote file system to a local directory, known as the "MOUNTPOINT". The files offered by the server are then visible under this directory.

NOTICE

Password protection

Note that when entering this command, the password appears on the screen in plain text according to the specified syntax.

For path names, use the slash "/" and not the backslash "\".

Supported file systems

Two types of remote file system are supported: Windows SMB and Unix NFS. These two systems have completely different characteristics, particularly in terms of user administration:

- In Windows SMB, you connect to the server as a particular user that the server must recognize. Via this connection, you then access the files as this user, independently of which local user triggers the action.

This feature means that in SMB systems, you already have to enter a user name, if necessary its domain, and a password at the time that the connection is established.

- In NFS servers, the connection itself does not require a particular user to be entered. Instead, for each file operation, the user who wants to carry out the operation must log on to the server. The server then decides whether or not to permit this. Users are entered using a user ID and group ID, not with names. The server must therefore recognize the corresponding IDs (or permit access for all users.)

Another type of remote file system that is supported are USB memory devices exported from TCUs (USB Flash Drive). Since these are integrated using NFS, the entries for server and path are similar to those for NFS. However, the TCU names are administrated differently, and the USB memory devices have specific paths that do not physically exist.

Notation of the remote file system

For SMB and NFS/TCU, the file system is entered using different notations:

SMB: `//[USERNAME[/DOMAIN][%PASSWORD]@]SERVER/SHARE`

The fixed share is: `//SERVER/SHARE`

The server name can of course also be a numeric IP address. SHARE is the name of the release on the server. Note that the character '\$', which often occurs in this type of name, must be preceded by a backslash ('\') in the command line. Otherwise, the system tries to expand a variable.

A user name can also be entered in front of the server name, separated with a '@' character. If necessary, the user name can also be extended by adding '/' followed by the Windows domain to which it belongs. The password belonging to this user is normally queried interactively, so that it is not visible on the screen.

For some applications, however, it may be necessary to write the password on the command line. (For example, programs started from WinSCP cannot read from the keyboard.) In this case, you can append an additional '%' character, followed by the password. If this contains any special characters that are interpreted by the shell (<, >, &, ;, ", ', \$, (,), |), you should shield these by preceding them with a backslash. Commas in SMB passwords cannot be interpreted.

NFS: `SERVER:/PATH`

In NFS, entering the user is not required, which just leaves the two components SERVER and PATH. As before, SERVER is the name of the file server. In contrast to SMB, PATH is a path name that exists on the server, and not an arbitrarily assigned name.

The command "sc show drives SERVER" shows the file systems offered by a particular server in the correct notation. For SMB, if required, the user name etc. still need to be added.

After the entry of the remote file system, you can also specify the required directory ('MOUNTPOINT') in which the remote files will become visible locally. This should be an empty directory to which you have write access. If MOUNTPOINT is omitted, 'sc' itself generates a suitable directory. The name is /tmp/drvNN (numbered consecutively), and is displayed when the connection is successful.

The option "-ro" connects the remote file system in read-only mode, which means that you can only read and do not have write access. This can be useful if you want to exclude changes that originate from the controller, or if the server only allows read-only connections.

As explained above, an SMB file system is connected as the specified user. To prevent a third user from executing file operations on the controller in the name of the connected user, the local directory can normally only be accessed by the latter (and all members of the same group).

However, if you do want to make the remote directory available to further local users, you can use the option "-public" to make the MOUNTPOINT belong to the "operator" group, and this access is therefore guaranteed.

2.5.6 Disable

Description

Syntax:	<code>sc disable hmi nck SUBSYSTEM ...</code> <code>sc disable DHCPSPvr -INTERFACE</code> <code>sc disable DHCPSTync [-X120]</code> <code>sc disable IPALias [-X120]</code> <code>sc disable usb [-MINUTES] all HOSTS</code>
Alternative names:	---
Authorization level:	service

The "sc disable" command switches the specified subsystems off. These subsystems are then not loaded again at the next restart. You can also switch several subsystems off with a single "disable" command. "hmi", "nck" or any other subsystem names can be used for the subsystems.

Note

The "sc disable" command acts on the parameters in the file basesys.ini, which is in the following directory: card/user/system/etc/basesys.ini

"sc disable IPALias" deactivates the second alias IP address for the X120 network interface.

See also

Parameters "InternalIP_Alias" and "InternalNetMask_Alias" in the "basesys.ini" file.

Specific subsystems

The name "DHCP Svr" is a special case. This does not switch a normal subsystem on or off, but specifies whether or not a DHCP server should be started on a network interface. The interface is entered in the standard form.

The synchronization of the DHCP server is switched on and off with "DHCP Sync". The synchronization is only possible in the system network.

The commands "sc enable" and "sc disable" function by changing certain variables in the file /user/system/etc/basesys.ini. In subsystems, this variable is "DisableSubsystems", in DHCP Svr "DisableDHCPD<INTERFACE>".

USB memory devices are inhibited with "sc disable USB": Specifically, the devices of USB class "storage" are involved, i.e. input devices can still be operated at the USB interface.

- If a time is not specified [-MINUTES], then a permanent inhibit is set. When a time is specified, the inhibit is specified in minutes.
- The inhibit can refer to all network participants of the plant network or a list with host names or IP addresses can be specified, for which the inhibit is valid.

2.5.7 Disconnect

Description

Syntax:	sc disconnect MOUNTPOINT sc disconnect all
Alternative names:	umount, unmount
Privilege level:	user

The action "disconnect" terminates the connection to a remote file system previously connected using "connect". If the MOUNTPOINT has been automatically created by 'sc', the corresponding directory is also deleted again.

The variant "sc disconnect -all" can be used to remove all existing SMB and NFS file systems.

2.5.8 Distribute

Description

Syntax:	sc distribute [parameter]
Alternative names:	dist
Privilege level:	service

This command distributes TCU data to other machines in the system network. Only the parameter "tcudata" is relevant here.

distribute tcudata

Syntax: `sc distribute tcudata`
Alternative names: ---
Privilege level: `service`

This command informs the active DHCP server that TCU data in /user/common/tcu has been manually changed. The server can then distribute this data to the standby servers.

It is recommended that the version file in /user/common/tcu be changed manually as this is only read in once by the DHCP server at the start. An incrementing of the version would then only take effect at a restart.

Note

This command can be performed on any machine in the system network, not only on the active server. A message is always sent to the active server.

2.5.9 Enable

Description

Syntax: `sc enable hmi|nck|SUBSYSTEM ...`
`sc enable DHCPSPvr -INTERFACE`
`sc enable DHCPSPync [-X120] [-PRIORITY]`
`sc enable IPALias [-X120] IPADDR[/NETMASK]`
`sc enable usb [-MINUTES] all | HOSTS`
Alternative names: ---
Authorization level: `service`

The "sc enable" command switches on the specified subsystems such that these are also activated when a subsystem is started. You can also switch several subsystems on with a single "enable" command. "hmi", "nck" or any other subsystem names can be used for the subsystems.

Note

The "sc enable" command acts on the parameters in the file basesys.ini, which is located in the following directory: card/user/system/etc/basesys.ini

"sc enable IPAlias" defines a second alias IP address for a network interface. This is only supported for X120. The IP address itself has the usual notation with four numbers separated by decimal points. If not specified, the subnet screen form can be determined from the class (A/B/C) of the IP or determined explicitly. The CIDR notation (number of bits of the network section) is also possible.

See also

Parameters "InternalIP_Alias" and "InternalNetMask_Alias" in the "basesys.ini" file.

Specific subsystems

The name "DHCPsVr" is a special case. This does not switch a normal subsystem on or off, but specifies whether or not a DHCP server should be started on a network interface. The interface is entered in the standard form.

The commands "sc enable" and "sc disable" function by changing certain variables in the file /user/system/etc/basesys.ini. In subsystems, this variable is "DisableSubsystems", in DHCPsVr "DisableDHCPD<INTERFACE>".

With "sc enable usb" USB storage devices at permanently inhibited USB interfaces or at specific host names or IP addresses will be released again.

DHCP synchronization

The synchronization of the DHCP server is switched on and off with "DHCPsSync". The synchronization is only possible in the system network (X120). A priority can also be specified. -LOW, -HIGH or -MASTER.

The priorities have the following effect for the synchronization of the DHCP server:

- **MASTER:** The computer node will become an active DHCP server. If several servers have been configured in the system network, this computer has the highest priority.
- **HIGH:** The computer node belongs to the server candidates with high priority, i.e. if no server with "MASTER" priority becomes active, then a computer with "HIGH" priority can be the active server.
- **LOW:** The computer node belongs to the server candidates with low priority, i.e. if no server with "MASTER" priority or "HIGH" priority becomes active, then a computer with "LOW" priority can be the active server.

Note

Recommended settings are:

- DHCP operation and DHCP synchronization are switched on in the system network.
 - **Exactly one** NCU is set as DHCP server.
 - **Maximum of two** computers are candidates with "HIGH" priority.
 - All other components are set as DHCP clients or candidates with "LOW" priority.
-

See also

Parameter "SyncModeDHCPD_SysNet" in the "basesys.ini" file.

2.5.10 Openport

Description

Syntax:	sc openport [-MINUTES] PROTO/PORT SOURCE ...
Alternative names:	---
Authorization level:	service

Mode of operation

This command opens a port in the firewall to the company network (X130) for a certain time. The default time is 15 minutes, but this can be changed with the -MINUTES option. The maximum possible time is 60 minutes.

The port to be opened is specified in the form "PROTO/PORTNR". The protocol can be either "tcp" or "udp". This is followed by the specification of the hosts from which the port should be accessible. Several forms are possible here:

- A single IP address: "128.128.12.12"
- A host name (if this can be resolved via DNS), e.g. "server"
- An IP area with network screen form specification, e.g. "128.128.12.0/255.255.255.0"
- An IP area with specification of the valid bits, e.g. "128.128.12.0/24"

If the specification of the source host is completely missing, this is normally acknowledged with an error message. However, an exception is when the "sc openport" command is entered in a shell opened via SSH. The permitted host is taken from the environment variable \$SSH_CLIENT that contains the IP of the SSH client.

If the opening was successful, an ID number is output in a status message:

```
sc: Port tcp/25 is open, rule ID is 6620
```

This ID can be used with "sc closeport" to manually close the port earlier.

NOTICE

Security risk

Note that opening the ports in the firewall can represent a security risk. Only enable the ports that you actually require.

Note that every accessible service can have security gaps.

See also

Parameter "FirewallOpenPorts" in the "basesys.ini" file.

2.5.11 Port status

Description

Syntax:	sc portstatus [-x127] PORTSPEC SOURCE
Alternative names:	---
Authorization level:	service

Mode of operation

The command "sc portstatus" can be used to query whether a certain port in the firewall has been opened. Option "-x127" must be specified if X127 is intended as the firewall instead of X130. The port specification looks exactly the same as it does for "sc openport", e.g. "tcp/102". However, at this point, no names are permitted and the specified source-IP must be unique.

The following status is output:

- "Port not open" with exit status 1.
- "Port is open (rule ID is <N>)." and exit status 0.

The ID is also supplied which you must specify for "sc closeport".

2.5.12 Restart

Description

Syntax:	sc restart all system network [!]SUBSYSTEM
Alternative names:	reboot
Authorization level:	service

With the action "sc restart" the specified subsystems are first stopped and then restarted with the following secondary condition: Dependent subsystems are also restarted with a restart.

Syntax

'hmi', 'nck' or any other subsystem names can be used for the subsystems. Several subsystems can also be specified one after the other, which are then started or stopped – including all of the required dependencies. It is also possible to start and stop all subsystems with the exception of those specified. To do this you write an exclamation mark "!" in front of the list.

"all", "system" and "network" are special values for the subsystem:

- "all" stops and starts all available subsystems.
- On the other hand, "sc restart system" initiates a system restart (reboot)
- "network" can only be specified in conjunction with "restart". "sc restart network" reinitializes the network, e.g. after settings have been changed in basesys.ini.

Examples

```
# stop all subsystems:
$ sc stop all
# only stop HMI:
$ sc stop hmi
# start HMI and NCK:
$ sc start hmi nck
# start all subsystems except NCK
$ sc start ! nck
```

2.5.13 Restore

Description

Syntax:	sc restore [-full -addon -addon+ -oem -oem+ -user] [-force] [-nodelete] [-update] [-restart] FILENAME
Alternative names:	---
Authorization level:	user

A backup created using "sc save" can be restored to the controller using "sc restore".

Mode of operation

As for "sc save", a restore is only possible from the service system or if subsystems are stopped. If "sc restore" reports an error, for example that the restore will cause running subsystems to crash, you can use the option "-force" to force the operation – just as for "sc save".

Normally, "sc restore" deletes the whole target area before the backup is restored (complete CompactFlash card for full backup, /user in user data backups). This means that no files not included in the backup are subsequently available.

Archive and backup files larger than 4GB

If a backup is to be restored with "sc restore" - consisting of several archive files - only the first file of the backup must be specified in the FILENAME parameter. Once the end of the file has been reached, other parts will be searched automatically.

Decisive when determining the continuation files is the format of the file name which was transferred during calling, i.e. when calling "sc restore /tmp/backup.01of02.tgz", and after reaching the end of "/tmp/backup.01of02.tgz" the search is on for the next file "/tmp/backup.02of02.tgz". When calling "sc restore /tmp/backup.1of2.tgz", file "/tmp/backup.2of2.tgz" is expected correspondingly as a second part of the backup.

Options

If no further options are specified, the archive should represent a full backup and this should be restored in full. The status of all files is therefore the same after the "restore" as it was at the time of the backup.

- The -full option additionally forces the partition and the file system to be recreated on the CF card. This is, however, only possible from a service system. -full is necessary if the partition table and/or file system is missing or damaged.
- However, if you want to restore the backup files without losing any files that have been created in the meantime, you can use the option "-nodelete" to prevent these from being deleted. "-nodelete" is not executed together with "-full", as when the file system is recreated, all files are essentially deleted.
- The options -addon, -oem, -user, -addon+ and -oem+ also allow you to unpack only parts of an archive.
- The -update option is used for loading software updates supplied by Siemens. The -restart option can also be useful with -update. If user-defined system files are changed when loading an archive, a restart or reinitialization is necessary. In such cases, "sc restore" outputs a message to that effect at the end of the operation. The -restart option causes the required action to be performed automatically.

2.5.14 Save

Description

Syntax:	sc save [-full -addon -addon+ -oem -oem+ -user] [-force] [-update] FILENAME
Alternative names:	backup
Authorization level:	user

The call of "sc help" without any additional action outputs a list of possible actions with a brief description. If you enter an additional action, you receive a more detailed description for this action.

The action "Save" or "Backup" creates a backup of the CF card in FILENAME. If the action is used from a service system, the backup receives the files of the underlying controller, not of the service system itself.

Archive and backup files larger than 4GB

If the Backup is written on a FAT32 target file system, no file size of more than 4GB is permitted, then the Backup is divided into several files, which are correspondingly smaller than this threshold value.

In the target file names, which are indicated during the call of "sc save", an indication is given in this case which part of a number of backups it is and of how many files it consists: The call of "sc save /tmp/backup.tgz" thus generates, for example, the files "/tmp/backup.01of02.tgz" and "/tmp/backup.02of02.tgz".

Options

The following options are used for selecting which directories of the file system are to be written to the archive:

-full:	complete backup (default): all files of the controller, incl. Boot Loader
-addon:	only /addon directory
-oem:	only /oem directory
-user:	only /user directory (user data)
-addon+:	/addon, /oem and /user
-oem+:	/oem and /user

If "sc save" is used directly on the controller and the subsystems are running, this may lead to inconsistencies between saved files, because they can still change during the backup process. Therefore, in this case "sc save" normally ends in an error message to inform you that subsystems are still running.

- If you want to create a backup anyway, you can use the -force option. "sc save" still issues a warning, but the process continues. If "sc save" is started from the service system, no subsystems of the controller are running and "-force" is not necessary.
- The -update option writes an Img-D control file to the archive. This makes it possible to use the archive at a later date via "sc restore -update".

Example:

```
sc save -user /tmp/drv01/backup.tgz
```

2.5.15 Show

Description

Syntax: `sc show ip [-INTERFACE]`
 Alternative names: `----`
 Authorization level: `none`

In the "sc show" command, various displays are grouped together showing the status of the system.

show ip

Syntax: `sc show ip [-INTERFACE]`
 Authorization level: `none`

This command displays the IP address data of network interfaces. You have the option to enter a specific interface. If no particular interface is specified, the data for all existing interfaces is displayed, as well as the Default Gateway.

Example:

```
sc show ip
X120 (system network, eth0):
  configured: (default)
  current : IP=192.168.214.1 Netmask=255.255.255.0
MAC=08:00:06:73:55:fd
  DNS Name : ncu1.local
  Nameserver: 127.0.0.1
  DNS suffix: local
  DHCP : synced server, prio=high, active
  Statistics: RX=0.0MB (0.00% errors), TX=0.2MB (0.00% errors)
X130 (company network, eth1):
  configured: DHCP
  current : IP=111.222.333.64 Netmask=255.255.248.0
MAC=08:00:06:73:55:fe
  DNS Name : name.test.siemens.de
  Nameserver: 111.222.333.12 111.222.333.13 111.222.333.14
  DNS suffix: test.siemens.de
  DHCP : client (server: 111.222.333.221)
  Statistics: RX=1.2MB (0.00% errors), TX=0.0MB (0.00% errors)
X127 (engineering network, ibn0):
  current : IP=192.168.215.1 Netmask=255.255.255.224
MAC=08:00:06:73:55:ff
  DNS Name : ncu-ibn
  DHCP : server
  Statistics: RX=0.0MB (0.00% errors), TX=0.0MB (0.00% errors)
Default gateway: 111.222.333.1 (via eth1)
Used nameserver: 127.0.0.1
Used DNS suffix: test.siemens.de local
```

The following data is displayed for each interface:

- **Name:** the name of the connection socket (X1_ _), together with the name used by the operating system in brackets (ethN or ibnN).
- **"configured":** the IP address configured in the basesys.ini (variables ExternalIP/ExternalNetMask for X130, InternalIP/Internal-NetMask for X120), or "(default)" if nothing is configured in the basesys.ini, or "DHCP" if the address was sourced via DHCP.
- **"current":** the currently set IP address together with the network screen form and MAC address of the interface.
- **"DNS Name":** Result of a DNS reverse lookup on the current IP address.
- **"Nameserver":** Here the list of DNS servers is output which are related to this interface (receive e.g. DHCP via this interface).
- **"DNS Suffix":** DNS search suffix related to this interface.
- **"DHCP":** Indication is given here if a DHCP client or server is running for this interface. In the case of a client, the server where the IP address comes from is displayed, too. A DHCP server on X120 can additionally be synchronized on the system network. Then the information is displayed if this is the active server or if it is in standby mode.
- **"Statistics":** Total amount of data received or sent via this interface, and the percentage of faulty packages.

When all interfaces are displayed, the default gateway is also output, i.e. the address of a router to which all packages are sent that cannot reach their destination directly via a local interface.

The default gateway is therefore a piece of data that applies to all interfaces and of which there is only one instance. However, there is one interface via which it must be possible to address the default gateway. This is displayed in brackets after the address.

show drives

Syntax:	sc show drives SERVER
Authorization level:	none

"sc show drives" shows the available remote file systems for a particular server. The server name SERVER can be an NFS server or a "TCU", which represents any USB media that are connected to a TCU. For details of possible server names and entering a user name (often required for SMB), see the description of the "sc connect" command.

Examples

Notation:

```
sc show drives someuser/somedomain@somepc # Windows-Server
Password: *****
//somepc/C$
//somepc/D$
//somepc/images

sc show drives someserver # NFS-Server
someserver:/export/home1
someserver:/export/home2

sc show drives TCU # TCU USB-Medien
TCU1:/dev0-0
TCU2:/dev0-0
```

All the available remote file systems of the relevant server are listed in the notation that is expected by "sc connect".

- In NFS file systems, the server name is always first, followed by a colon and then the export path.
- A USB memory on a TCU is a special form of NFS, and therefore also has the same notation as an NFS file system. The specified path exists - but not physically - on the TCU, but is converted there to USB by the NFS server.

show net

Syntax: `sc show net [-xml] [-hw|-tco|-sw|-swfull|-loc|-panel|-dhcp|-switch|-all] [HOSTS...]`

Authorization level: none

The "sc show net" command displays the devices in the system network and gives additional information on these devices.

As both these functions rely on SNMP (Simple Network Management Protocol), only those devices able to use SNMP will be found. SNMP as from version 2.6 is available for Linux-based devices, for all other devices (e.g. MCP) this depends on the respective software version. If a simple call is made, without any additional options, a list of the devices found will appear showing the relevant IP address, DNS name (if known), and a short description (module name).

Options

The various switches, which can also be combined, allow additional information to be output on the devices found:

- Unless additional options are involved, the output format for "sc show net" is designed to be displayed on the screen. The alternative XML format is preferable, however, if further machine-based processing is required. This can be selected via the -xml option (this option must be the first one!).
- If the command line contains nothing after the switches, "sc show net" will search the entire system network by broadcast for devices able to use SNMP. This search takes time, however, and in the case of larger systems, the volume of information output may be considerable. In view of this, a list of IP addresses or (DNS) names can be specified to ensure a better overview. This will limit the search process and only the named devices will be included.

-hw	Information on hardware, such as the MLFB, serial number, hardware ID, hardware version, SRM version, manufacturer, and the serial number of the CF card. Some devices have two of these hardware blocks (the second one being intended for an integrated device). The PLC in the NCU and the TACO in the OP on an TCU are two such examples.
-tco	Operating state data: switching counter, operating hours counter, quantity of data written to the CF card, number of times maximum temperature exceeded
-loc	Location data: location, function, and contact details for the device (if configured there)
-sw	Displays brief details of the software installed, just the version of the overall status (or of the basic system, if this is all that is installed)
-swfull	Displays all the installed software components. These are represented hierarchically in the form of a tree. The following is provided for each component: information on the version, internal version (if different), target version (if different), version details, and the installation path.

-panel:	Displays panel-specific data: size and color depth of the display, MCP/TCU/DCK/EKS index, and the VNC server currently represented
-dhcp	Data on a synchronizing DHCP server: operating state (off/standby/active), priority, versions of the .leases file and TCU data, and the range for dynamically assigned IP addresses
-switch	Displays the settings of rotary switches on the device, e.g. the NC and PLC switches of an NCU or the DIPFIX of an HT 8
-all	Equivalent to all the switches above

Note

The data and information read out here are automatically written to the EUNA file (End User Notification Administration), and saved on the respective device.

2.5.16 Start, Stop

Description

Syntax:	sc start all system SUBSYSTEM sc start all system SUBSYSTEM
Alternative names:	---
Privilege level:	service

These two actions start or stop individual or all subsystems. Since these actions have a direct influence on the runtime behavior, they are only available directly on the PLC, but not in the service system (with the exception of "sc stop system"). The subsystems of the underlying PLC do not run here, and can therefore not be controlled.

Note

Only "all" and "system" function, but not arbitrary subsystems.

Both of these actions have only been implemented to a restricted extent.

Subsystem names

"hmi", "nck", or any other names, can be used for the subsystems. Multiple subsystems can also be specified in a row, which are then started and stopped in this order.

"all" and "system" are special values for the subsystem:

- "all" starts or stops all available subsystems.
- "sc stop system" similarly stops all subsystems, but also the basic system with the PLC.
- "start system" is identical to "start all" and exists mainly for reasons of symmetry.

start snc

Syntax:	sc start snc
Alternative names:	---
Privilege level:	service

This command starts the "System Network Center". To operate it, you will need a mouse.

The "System Network Center" you can end with "Exit". If you do not wish to end the "System Network Center", then switch to the HMI with the following key combination:

<Recall> + <MENU SELECT> (Area switchover key) or

<F9> + <F10> on an external keyboard

Reference

A description of the settings in the "System Network Center" is provided in the Operating Components and Networking Manual.

Service and diagnostics

3.1 Creating a service system

Overview

To **backup** user data or the complete CompactFlash Card, proceed as follows:

- Creating a service system.
- Backing up the license
- Backup: Data backup on USB storage medium or on a network drive.
- System diagnostics
 - Output network settings
 - Changing the network settings
 - Starting the VNC Viewer

To **restore** the user data or the complete CompactFlash Card, proceed as follows:

- Restore/Recover: Restore the data from USB storage medium or from network drive.
- Carry out software update.
- Carry out firmware update.

3.1.1 How do you create a service system for the NCU?

Purpose

In case service is needed, create a portable "Emergency Boot System" (EBS) on a USB memory. Thus you can start the booting of the NCU from the service system in order to carry out various service tasks, such as data backup or updates, in a service shell.

Two partitions are created on the service system:

- A Linux partition that is not displayed under Windows.
- An FAT partition, to backup files or save software updates.

The FAT partition can be read and written to under Linux as well as from a Windows system. The FAT partition can be addressed under the path/data in a command shell under Linux.

Scope of delivery

To create a service system on a USB storage medium, the following files are included on CD:

- an executable file installdisk.exe
- an image file for USB-FlashDrive
- a file with the latest information siemensd.txt / siemensse.txt

Recommendation:

It is better to use SIMATIC PC USB-FlashDrive.

Note

To create the service system, you need administrator rights.

All data already on the USB storage medium will be deleted.

The transfer is optimized for USB 2.0; therefore, the transmission to the USB storage medium takes longer when using USB 1.1 than USB 2.0

Proceed as follows

To create a service system on a USB storage medium:

1. Copy the service system onto a local hard disk of your PG/PC.
2. Connect the USB storage medium to the USB interface of the PG/PC.
3. Determine in Windows Explorer which drive letter the USB storage medium was assigned, e.g. H:
4. Open a DOS shell and change to the directory in which the files for the service system are stored.
5. In the DOS shell, enter the following command:

```
installdisk --verbose --blocksize 1m linuxbase.img h:
```

Result: The image is transferred to the USB storage medium.

3.1.2 This is how you operate the service system

Connecting-up the service system

Procedure:

1. Connect the USB storage medium to the USB interface X125 or X135 of the NCU.
2. Switch-off the system and then switch on again.

OR

3. Press the "Reset" button.

The service system is unzipped from the USB storage medium when the NCU boots for the first time: Only then can the USB storage medium be used as service system and the entire storage capacity is available.

Operating the service system

Keys and softkeys to navigate in the service system:

Softkey	Key on OP	External keyboard	Description
	HSK1	<F1>	Moves the cursor down a row
	HSK2	<F2>	Moves the cursor up a row
Page 	HSK3	<F3>	Moves the cursor down a page
Page 	HSK4	<F4>	Moves the cursor up a page
Char 	HSK5	<F5>	Inserts text or digits
Char 	HSK6	<F6>	Inserts text or digits
Cancel	VSK7	<Shift> + <F7>	Cancel / Return
Ok	VSK8	<Shift> + <F8>	OK / Confirm
---		Pos1	Moves the cursor to the top row
---		End	Moves the cursor to the bottom row

Note

Operation on touch panels

No touch operation is supported by the service system

- SINUMERIK OP019: An external keyboard is required for text input (connected via USB).
 - SIMATIC Thin Client Here, you can activate the integrated keyboard.
-

3.1.3 Diagnostics functions

Functions of the service menu

The following functions for service and diagnostics are available on the service system:

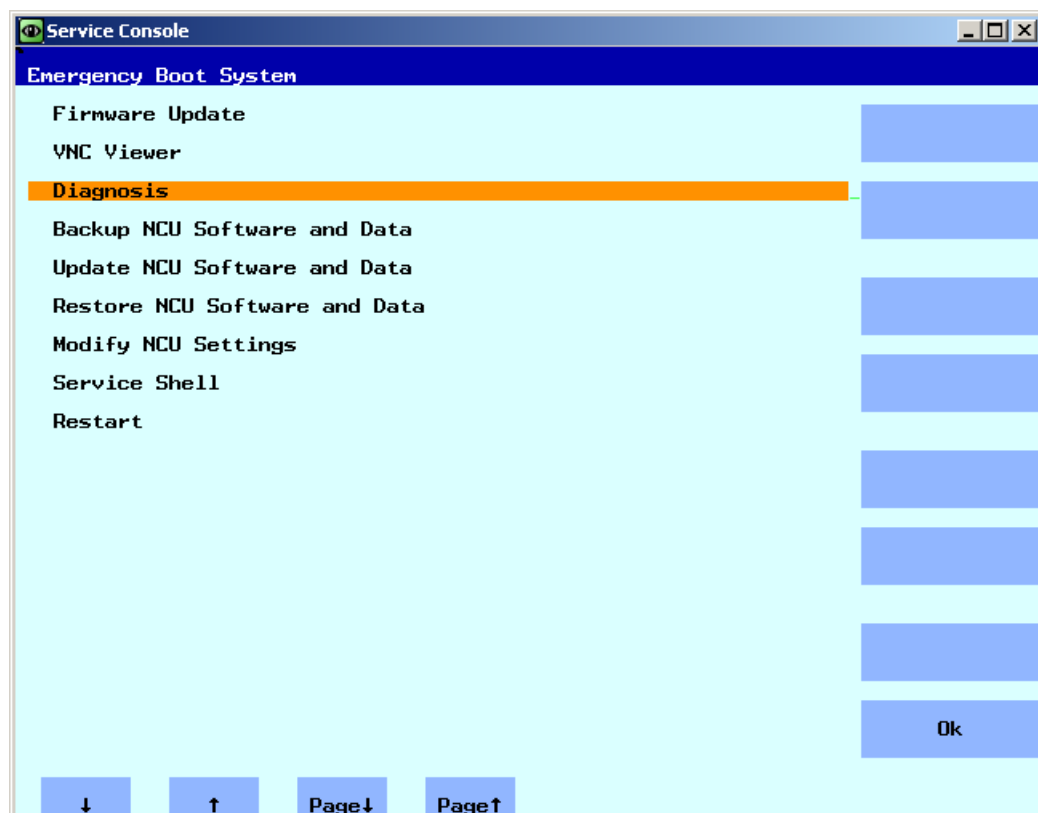


Figure 3-1 Service menu

Overview of the main menu:

Main menu	Description	Reference to chapter/manual
Firmware Update	Update BIOS: • PLC BIOS • FPGA • CBE30 Bootloader	This is how you update the firmware (Page 80)
VNC Viewer	Start a VNC Viewer	Calling the VNC Viewer (Page 89)
Diagnosis	Starting a diagnostic function to display network-connection data and data on the system CompactFlash Card.	Performing system diagnostics (Page 84)
Backup NCU Software and Data	Backup system and user data on the USB FlashDrive or network drive.	Backup data via the service menu (Page 68)
Update NCU Software and Data	Upgrade the system from the USB FlashDrive or network drive.	This is how you update the software (Page 82)
Restore NCU Software and Data	Restore system and user data from the USB storage medium or network drive.	Backup data via the service menu (Page 68)
Modify NCU Settings	• Change network settings • Activate/deactivate HMI • Restore delivery condition of the following components: – HMI – NCK – PLC – Drive	This is how you change the system settings (Page 86)
Service Shell	A Service Shell is opened in order to input service commands.	Backing up data via the service shell (Page 75) Service Commands (Page 35)

3.1.4 This is how you connect to a network drive

Connecting network drives

1. Select "Connect to network drive".
2. Enter the complete path name of the network drive.
3. Register yourself with a user name with access authorization to this drive.

Note the syntax of the examples specified:

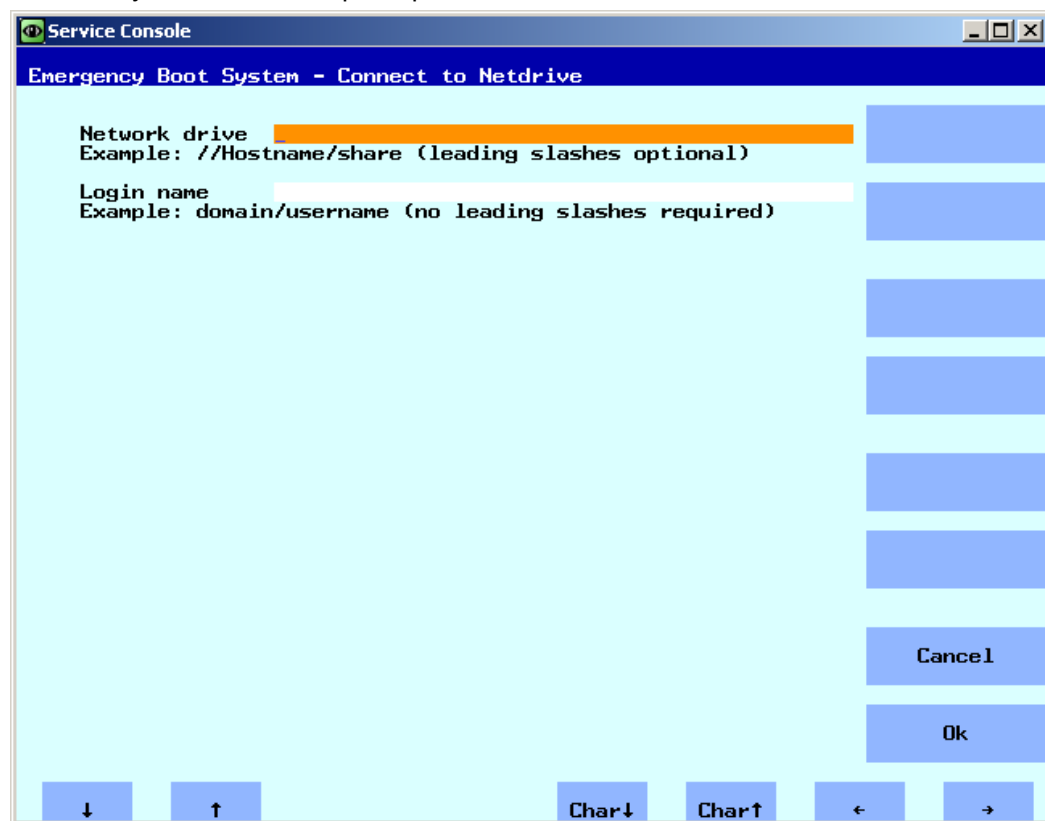


Figure 3-2 Connecting network drives

4. Confirm with "Ok". You are then prompted to enter a password.

3.2 Backup license

Validity of the license key

The license key corresponds to the serial number of the CompactFlash Card. If the CompactFlash Card is replaced for a SINUMERIK 840D sl, the license key loses its validity and the system is no longer ready for operation.

This situation can occur for a defective hardware of the system CompactFlash Card.

Application

For the following cases, backup the license key of the CompactFlash Card on a USB storage medium or on a network drive:

- Software update without license key
- System software error without defective hardware

Note

Replacement CompactFlash Card

The following data are required in order to obtain a valid license key after replacing the system CompactFlash Card: **The serial number of the defective and the new CompactFlash Card.**

3.2.1 This is how you backup the license key

Backing up the license key on a USB storage medium

Procedure:

1. In the main menu, select the menu item "Backup NCU Software and Data".

The following menu is displayed:

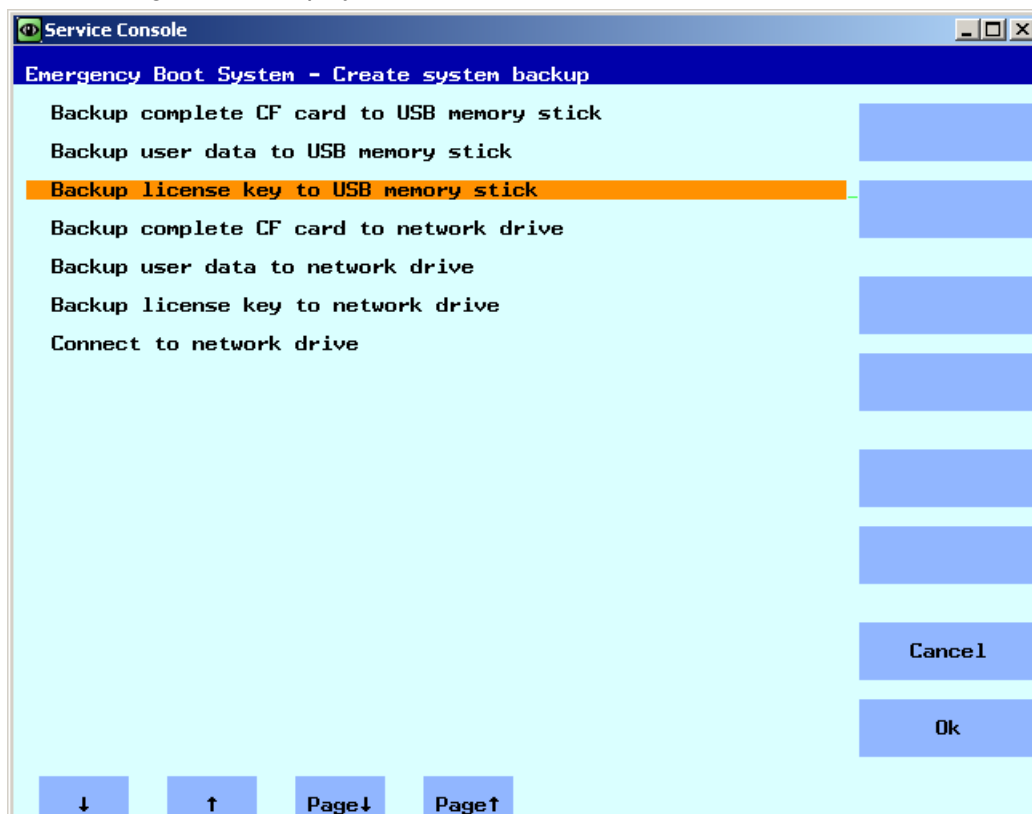


Figure 3-3 Backing up licenses

2. Select the menu item "Backup license key to USB memory stick".

The following menu is displayed:

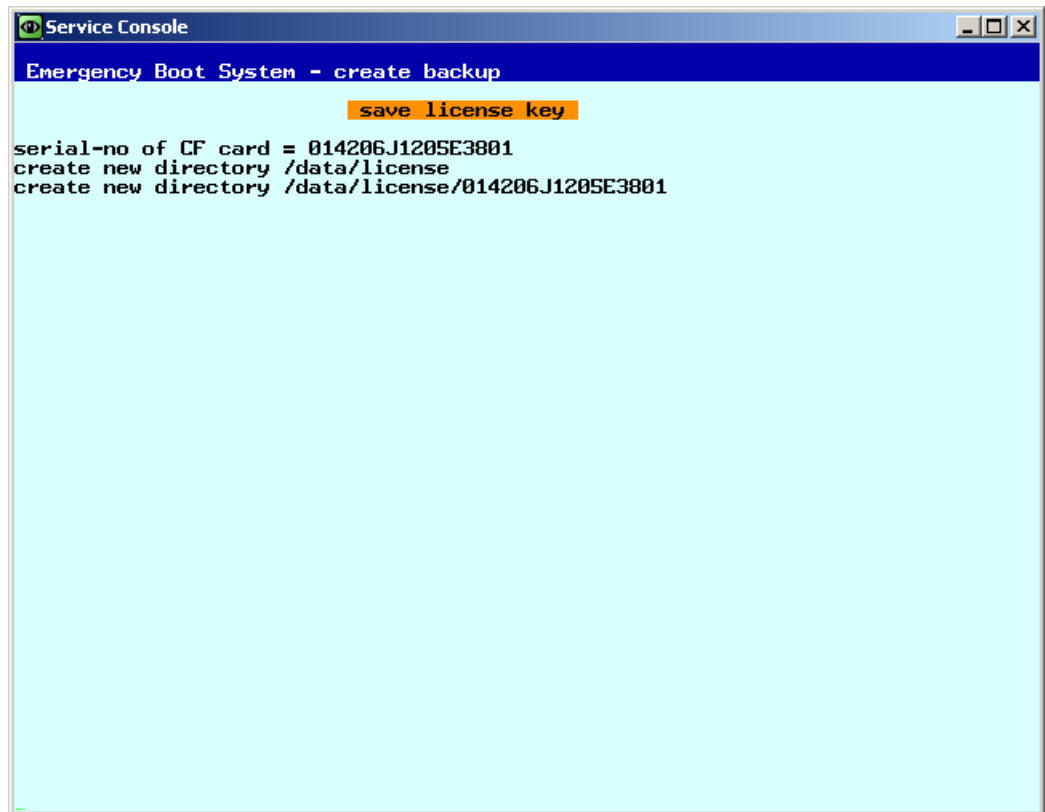


Figure 3-4 Save license key

The license key is backed up in the specified directory together with the serial number of the CompactFlash Card.

See also

The procedure is essentially the same as for "Backup license key to network drive". In addition, you connect to a network drive:

This is how you connect to a network drive (Page 64)

3.3 Backup data via the service menu

3.3.1 This is how you backup user data

Backing up and restoring user data

Procedure:

1. In the main menu, select the menu item "Backup NCU Software and Data".
2. Select "Backup user data to USB memory stick" and confirm with "Ok".
3. Enter a complete file name - as shown in the example:

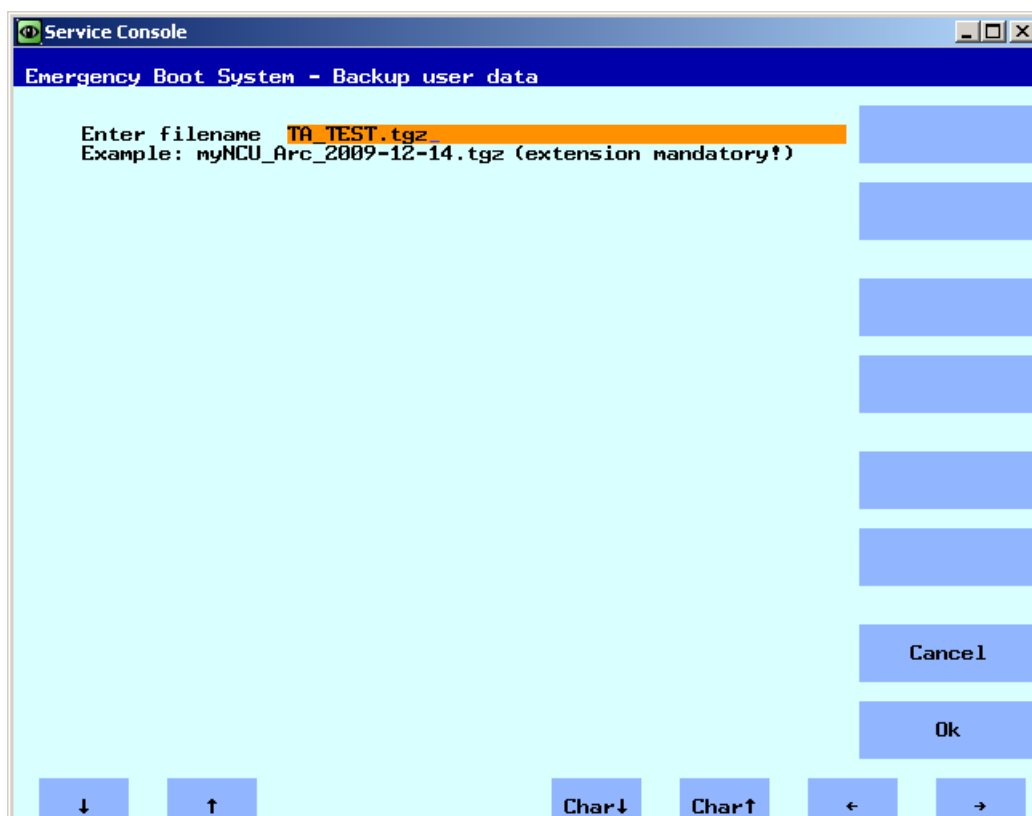


Figure 3-5 Entering a file name

4. Confirm with "Ok". The following message is output after the operation has been successfully completed:

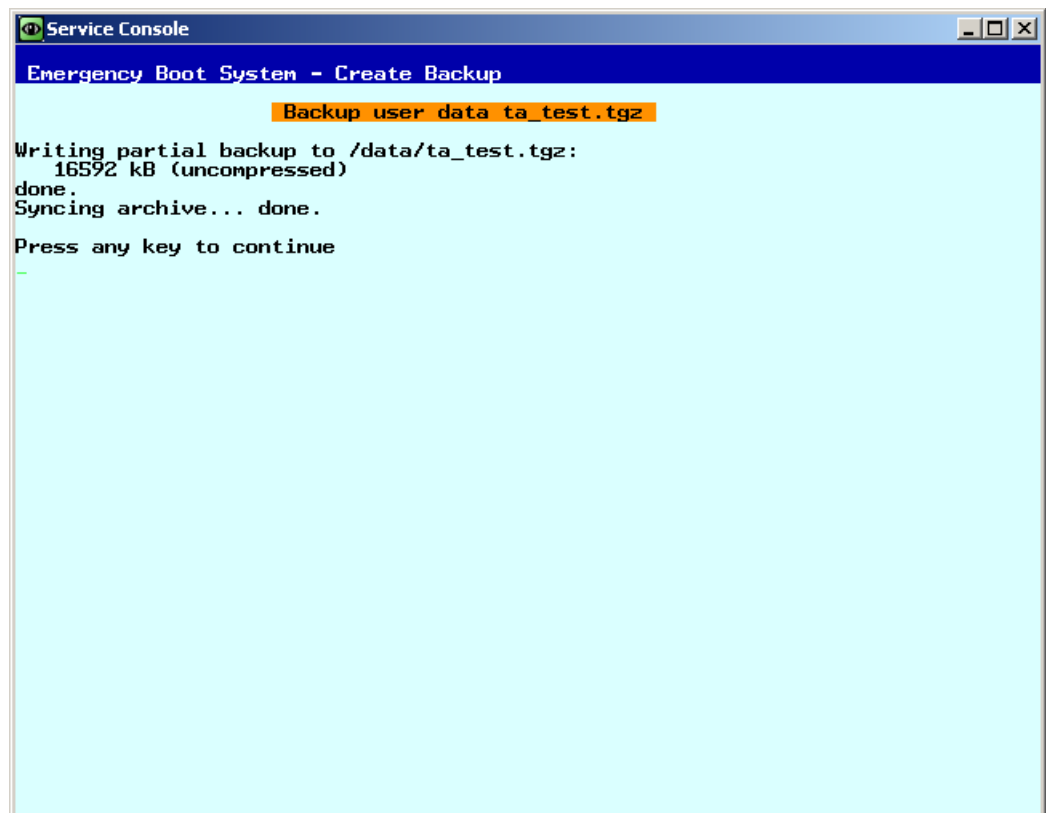


Figure 3-6 Operation completed

Note

Backup file $\geq$ 4GB

For a backup file $\geq$ 4GB, the space on the FAT partition of the service system (with 8GB) is no longer sufficient. For this reason, backup files of this size are partitioned (see also: sc save/sc restore) or they can be saved on a network drive.

See also

The procedure for "Backup user data to network drive" is analog. In addition, you connect to a network drive:

This is how you connect to a network drive (Page 64)

3.3.2 This is how you restore user data

Restoring user data

Procedure:

1. In the main menu, select the menu item "Restore NCU Software and Data".

The following menu is displayed:

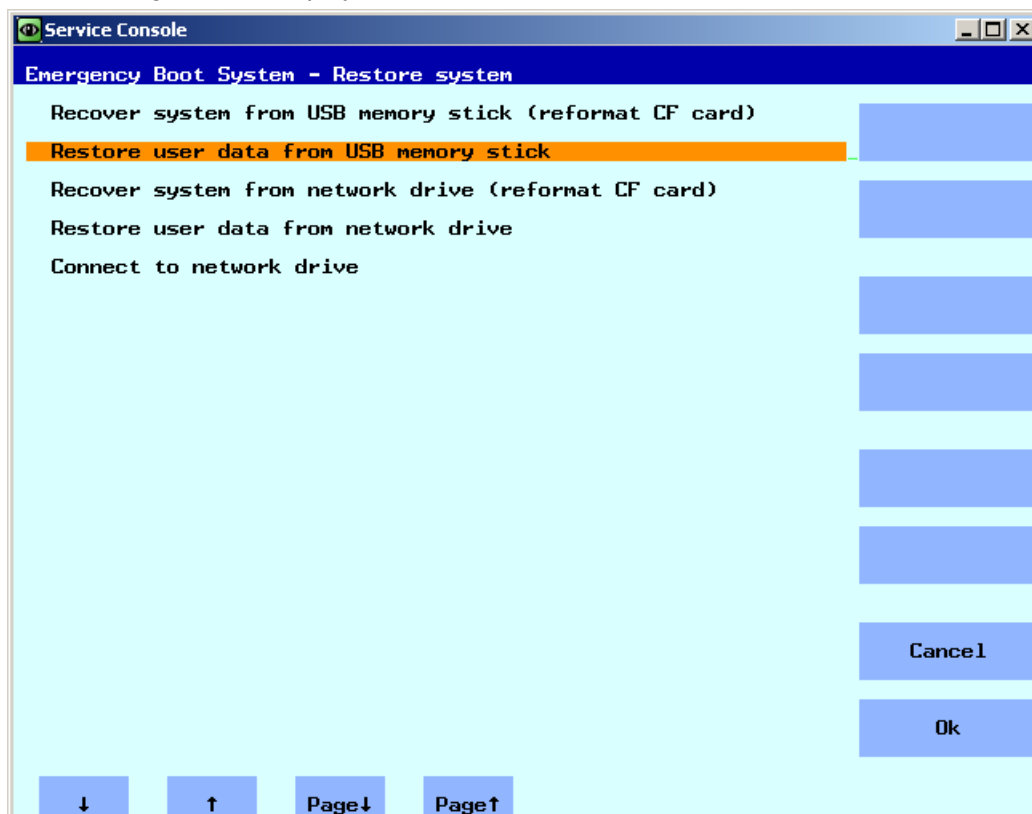


Figure 3-7 Restoring user data

2. Select "Restore user data from USB memory stick" and confirm with "Ok".
The list of the tgz files available on the USB storage medium are displayed.
3. Select the appropriate tgz file and confirm with "Ok".
A message is output after the operation has been successfully completed.

See also

The procedure for "Restore user data from network drive" is essentially the same. In addition, you connect to a network drive:

This is how you connect to a network drive (Page 64)

3.3.3 This is how you create a complete backup of the CompactFlash Card

Creating a complete backup

Procedure:

1. In the main menu, select the menu item "Backup NCU Software and Data".

The following menu is displayed:

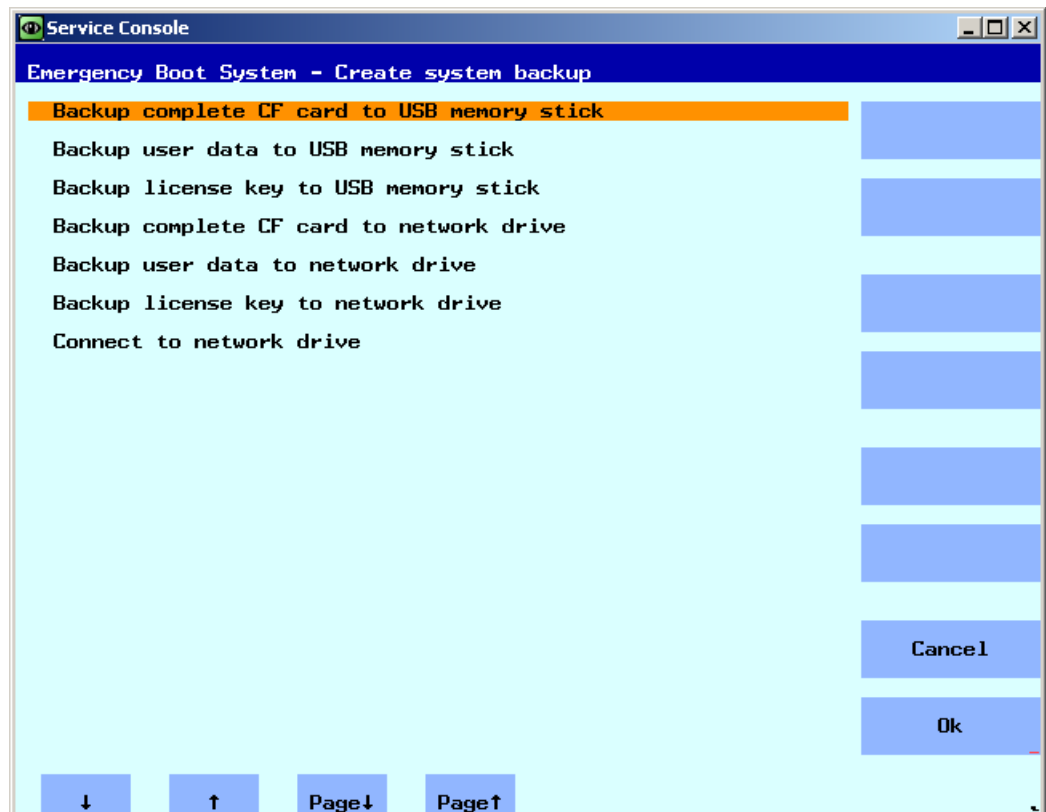


Figure 3-8 Backup up

2. Select "Backup complete CF card to USB memory stick" and confirm with "OK".

3. Enter a complete file name - as specified in the example.
4. Confirm with "OK". This can take several minutes.

The operation is displayed:

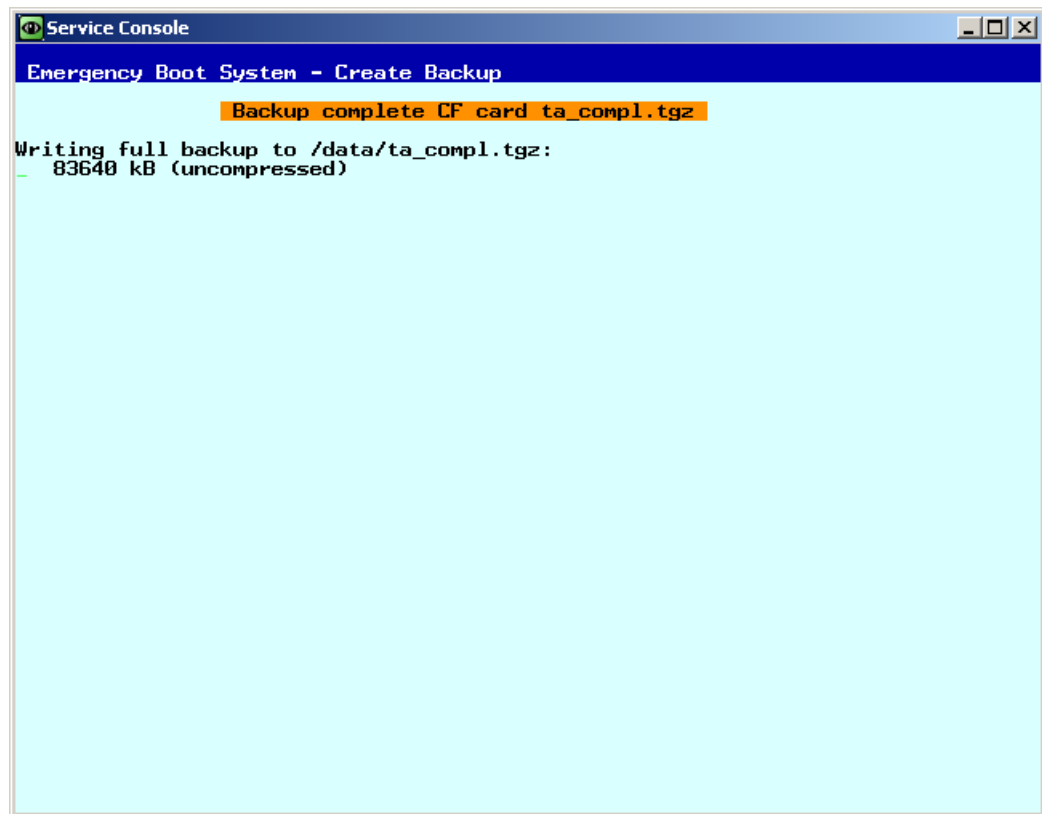


Figure 3-9 Backing up ...

A message is output after the operation has been successfully completed.

See also

The procedure for "Backup complete CF card to network drive" is essentially the same. In addition, you connect to a network drive:

This is how you connect to a network drive (Page 64)

3.3.4 This is how you install a complete system backup

Installing the complete backup

With this menu item, a complete backup of the data from the service system or from a network drive is installed on the CompactFlash Card, i.e. this backup includes both the system software as well as also the user data.

NOTICE

Formatting

This procedure corresponds to formatting the CompactFlash Card and all of the existing data are overwritten.

Installing the complete backup

Procedure:

1. In the main menu, select the menu item "Restore NCU Software and Data".

The following menu is displayed:

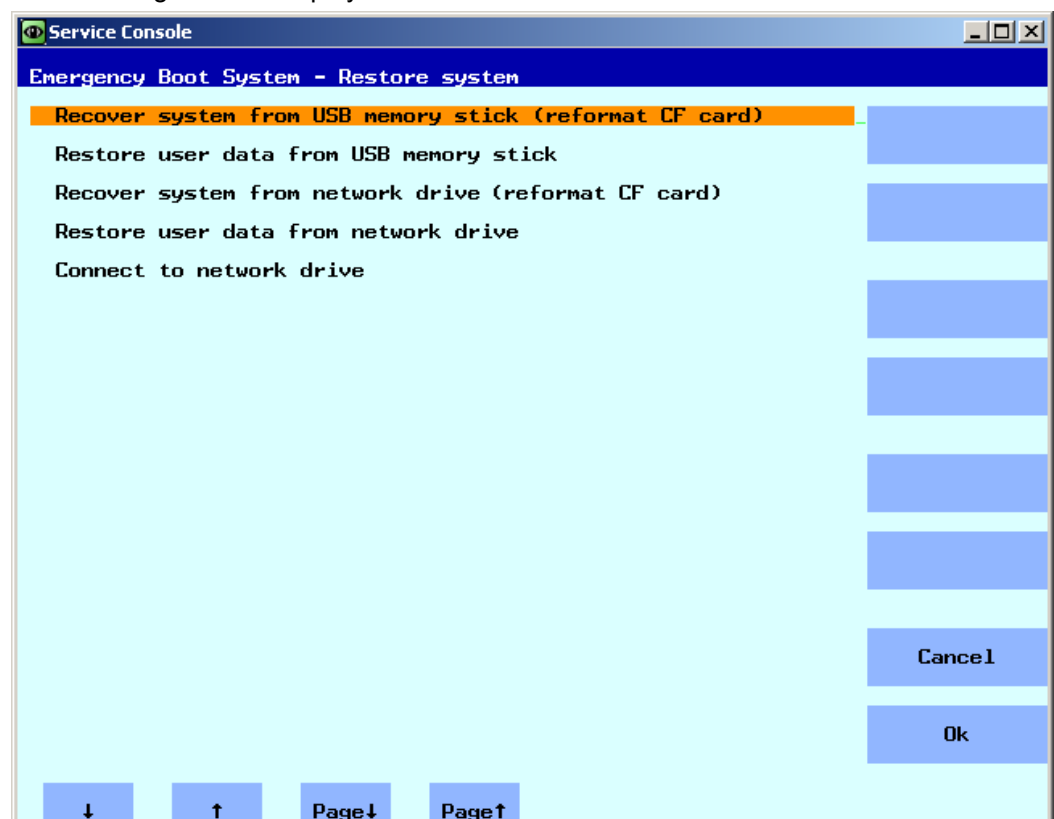


Figure 3-10 Restore system

2. Select the menu item "Recover system from USB memory stick (reformat CF card)" and confirm with "Ok".

The list of the tgz files available on the USB storage medium are displayed.

3. Select the appropriate tgz file and confirm with "Ok".

A message is output after the operation has been successfully completed.

Completion

The system must be restarted in order that the new data become effective.

See also

The procedure for "Recover system from network drive (reformat CF card)" is essentially the same. In addition, you connect to a network drive:

This is how you connect to a network drive (Page 64)

3.4 Backing up data via the service shell

Backing up and restoring using a service shell

To call a service shell, depending on the configuration, the following possibilities are available:

- (I) Configuration of NCU with TCU: the service shell is called under Linux.
- (II) Configuring the NCU with PCU 50 or programming device (PG).
A service shell can be called under:
 - on the NCU under Linux
 - on the PCU/PG using the VNC Viewer (system network or company network)
 - on the PCU/PG using WinSCP (system network or company network)

3.4.1 This is how you save data on a service system

Proceed as follows

To back up the complete system:

1. Connect the service system to a USB interface (X125 or X135) of the NCU and press the reset button.

Alternatively, you can switch the NCU off, connect the service system, and switch the NCU on again.

Result: The NCU boots from the service system.

2. In the main menu, select the menu item "Service Shell".
3. Log on as a service technician using the user name "manufact" and password "SUNRISE".
4. Using the command "sc backup" you can create the backup file "backup01.tgz".

The directory /data on the service system is provided for backup files. The complete path should be specified.

5. Choose –full, to backup all data on the CompactFlash Card, or –user, if you only want to backup user data.

Example: `sc backup -full /data/backup01.tgz`

Result:

A backup file of the complete CompactFlash Card is created under /data on the service system on the USB storage medium.

3.4.2 This is how you restore data from the service system

Proceed as follows

To restore the complete system:

1. Connect the service system to a USB interface (X125 or X135) of the NCU and press the reset button.

Alternatively, you can switch the NCU off, connect the service system, and switch the NCU on again.

Result: The NCU boots from the service system and the main menu is displayed.

2. In the main menu, select the menu item "Service Shell".
3. Log on as a service technician using the user name "manufact" and password "SUNRISE".
4. With the command "sc restore" you can write the backup file "backup01" from the service system back to the CompactFlash Card in the NCU. The complete path should be specified.

Example: `sc restore /data/backup01.tgz`

Result:

The system state stored in the file "backup01" is restored on the NCU.

Note

If access to the system data on the CompactFlash Card is not possible because the CompactFlash Card is defective or empty, you can only log in as user "admin" with the password "SUNRISE" and no longer as the user "manufact".

3.4.3 This is how you save data on a network drive

Sequence

Proceed as follows:

- Connect the service system.
- "Service Shell" open.
- Establish a connection to a network drive.
- Create a backup file.

Scenario I: Start Command Shell under Linux

Procedure:

1. Connect the service system to the USB interface X125 or X135 of the NCU.
2. In the main menu, select the menu item "Service Shell".
3. Log on as a service technician using the user name "manufact" and password "SUNRISE".
4. Use the command "sc connect" to connect the network drive:

```
sc connect //username%password@server/share /tmp/backup
```

Please enter here the user name and the password for the network drive that is to be connected to.

5. Using the command "sc save", create the backup file "backup01".

Choose -full, to backup all data on the CompactFlash Card, or -user, if you only want to backup user data in the directory /user.

Example: `sc save -full /tmp/backup/backup01`

Result:

A backup file of the complete CompactFlash Card is created under the specified path on the network drive.

Scenario II: Start the command shell with WinSCP on the programming device:

Procedure:

1. Start WinSCP and enter the following data in the log-on window:
 - IP address of the NCU (or, if required, host name)
 - User name "manufact" with password "SUNRISE".
2. Select in the menu "Commands" → "Open Terminal".
3. Execute the data backup using the commands described in steps 4 to 6 from scenario I.

Scenario III: Start the command shell using the VNC Viewer on the programming device

Procedure:

1. Start the VNC Viewer and connect to the NCU using the IP address (or where relevant, using the host name)
2. Log on under the "manufact" user name with the password "SUNRISE".
3. To restore the data, enter the corresponding commands as described under steps 3 to 5 in scenario I.

3.4.4 This is how you restore data from the network drive

Sequence

Proceed as follows:

- Connect the service system.
- "Service Shell" open.
- Establish a connection to a network drive.
- Restore the data.

Scenario I: Start command shell under Linux

1. Connect the service system to the USB interface X125 or X135 of the NCU.
2. In the main menu, select the menu item "Service Shell".
3. Log on as a service technician using the user name "manufact" and password "SUNRISE".
4. Use the command "sc connect" to connect the network drive:

```
sc connect //username%password@server/share /tmp/backup
```

5. Please enter here the user name and the password for the network drive that is to be connected to.

To completely restore the system (system data and user data), enter the following: `sc restore -full backup01`

Result: The whole system is overwritten by the backup data.

6. To restore user data only, enter the following command:

```
sc restore -user backup01
```

Result: The user data are written back again.

All subsystems are then restarted: `sc start all`

Result:

The system state stored in the file "backup01.tgz" is restored on the NCU.

Scenario II: Start the command shell with WinSCP on the programming device:

Procedure:

1. Start WinSCP and enter the following data in the log-on window:
 - IP address of the NCU (or, if required, host name)
 - User name "manufact" with password "SUNRISE".
2. Select in the menu "Commands" → "Open Terminal".
3. To restore the data, enter the corresponding commands as described under steps 3 to 5 in scenario I.

Scenario II: Start the command shell using the VNC Viewer on the programming device

Procedure:

1. Start the VNC Viewer and connect to the NCU using the IP address (or where relevant, using the host name)
2. Log on under the "manufact" user name with the password "SUNRISE".
3. To restore the data, enter the corresponding commands as described under steps 3 to 5 in scenario I.

3.5 Updating the software

3.5.1 This is how you update the firmware

Firmware Update

When service is required, it may be necessary for a service technician to install a BIOS update. Siemens provides the corresponding update file. Possible formats include: *.img or *.rom or *.bin. Or, the update file is already supplied on the service system, then step 1 is eliminated.

Procedure:

1. Copy the update file to the FAT partition of the service system.
2. Connect the service system to interface X125 or X135 of the NCU.
3. Switch on the system.
4. In the main menu, select the menu item "Firmware Update".

The following menu is displayed:

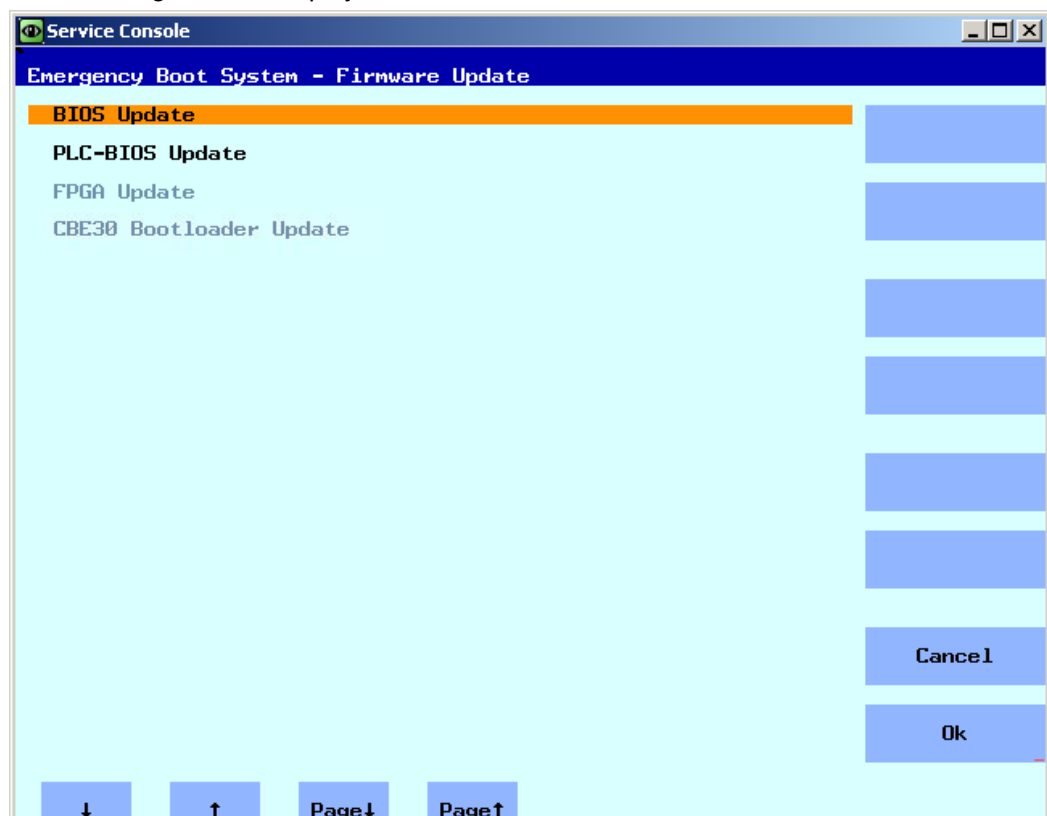


Figure 3-11 Firmware update

5. Select "BIOS Update" and follow the next instructions.

The BIOS version installed on the NCU and the update files available in the service system are output:

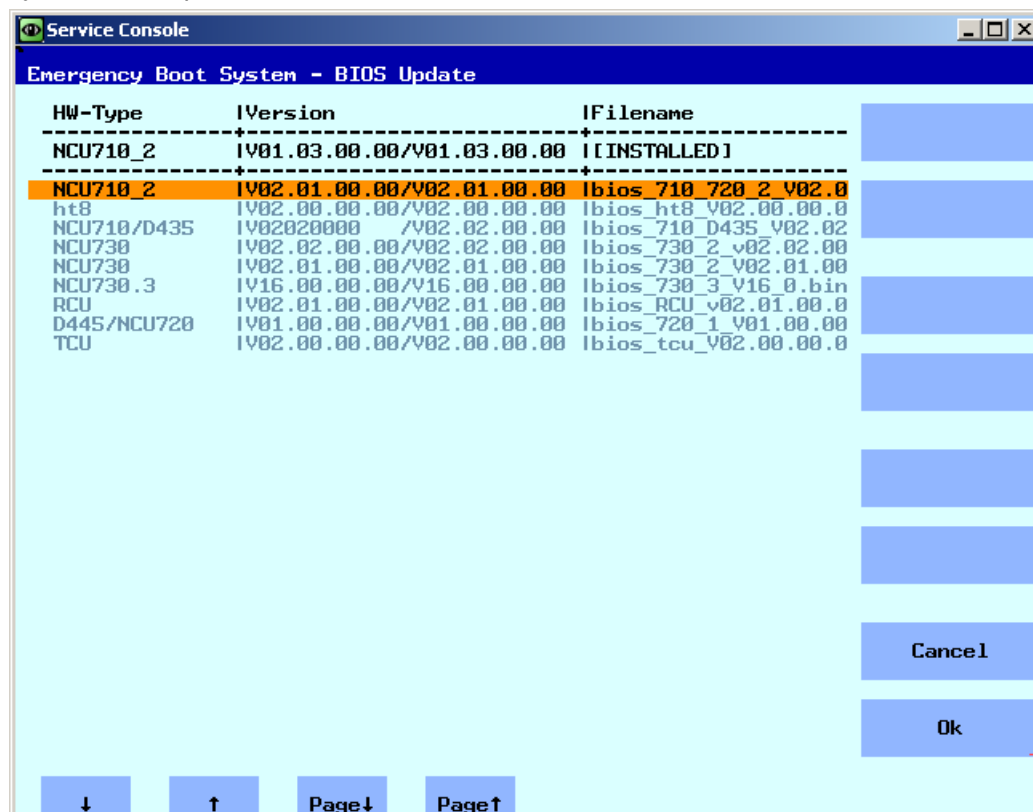


Figure 3-12 Select the updates

6. Select the update file and confirm with "OK".

A "Restart" is then required.

Note

Fall back strategy

Before the BIOS update is installed, a backup of the installed BIOS version is saved on the service system.

For safety reasons, only files can be selected that are suitable and are appropriate for this NCU.

Update other components

The following rules must be observed for the additional updates:

- The procedure to update the PLC-BIOS Update is essentially the same as the procedure for BIOS Update.
- FPGA Update: for NCU7x0.3
- CBE30 Bootloader Update: for CBE30 and CBE30-2 (option module)

3.5.2 This is how you update the software

Update / Recover

The following options are available:

- Update the software from the service system or from the network drive.
- Completely restore the CompactFlash Card with the system software from the service system or from the network drive (Recover system ...).

Updating the software

Procedure:

1. In the main menu, select the menu item "Update NCU Software and Data".

The following menu is displayed:

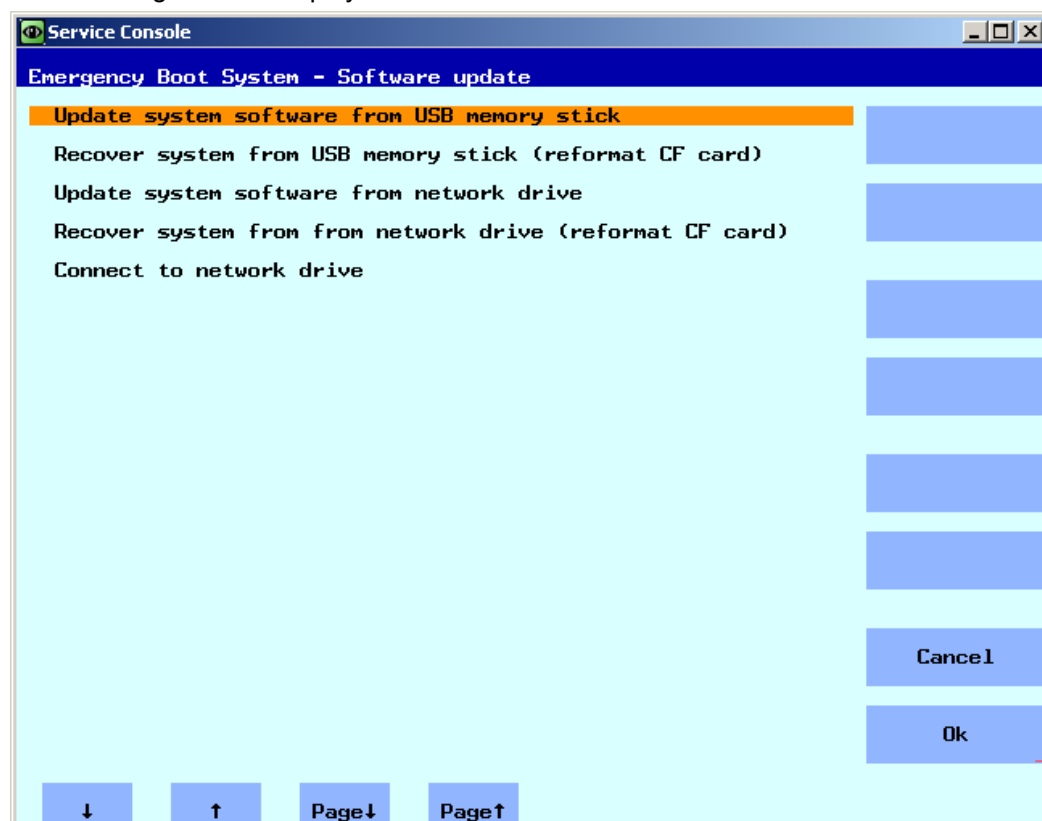


Figure 3-13 Software update

2. Select the menu item "Update system software from USB memory stick".
The list of the tgz files available on the USB storage medium are displayed.

3. In order to check which software version contains the tgz file, first select the tgz file, then press "Show version".

The following information is output:

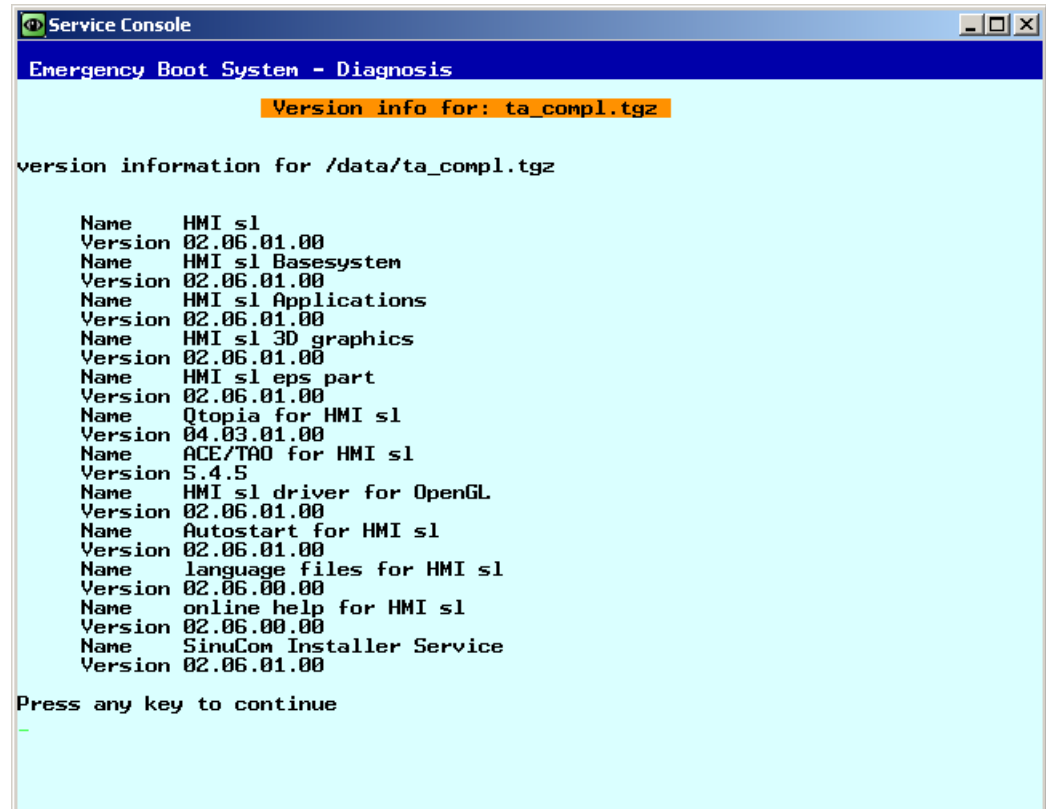


Figure 3-14 Version information

4. Select the appropriate tgz file and confirm with "Ok".

A message is output after the update has been successfully completed.

Completion

The system must be restarted in order that the new data become effective.

See also

The procedure for "Update system software from network drive" is essentially the same. In addition, you connect to a network drive: This is how you connect to a network drive (Page 64)

For a complete backup of the system: This is how you install a complete system backup (Page 73)

3.6 System diagnostics

3.6.1 Performing system diagnostics

Overview of diagnostic functions

The following diagnostic functions to display network settings are included in this menu:

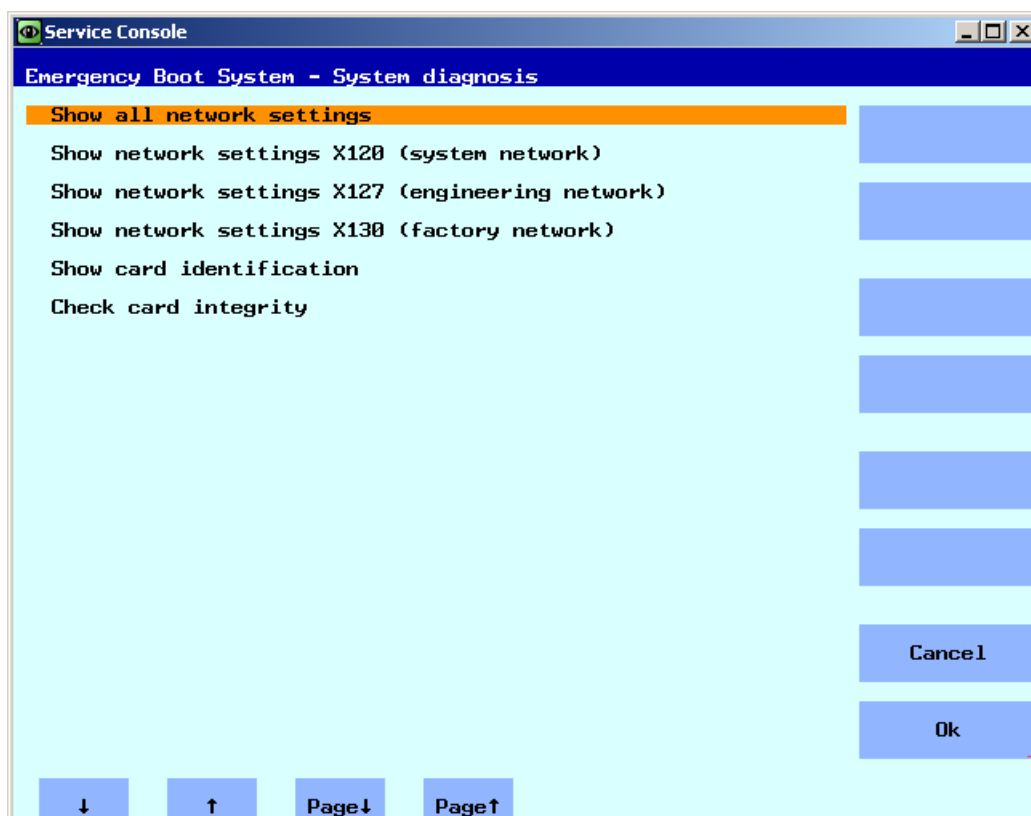
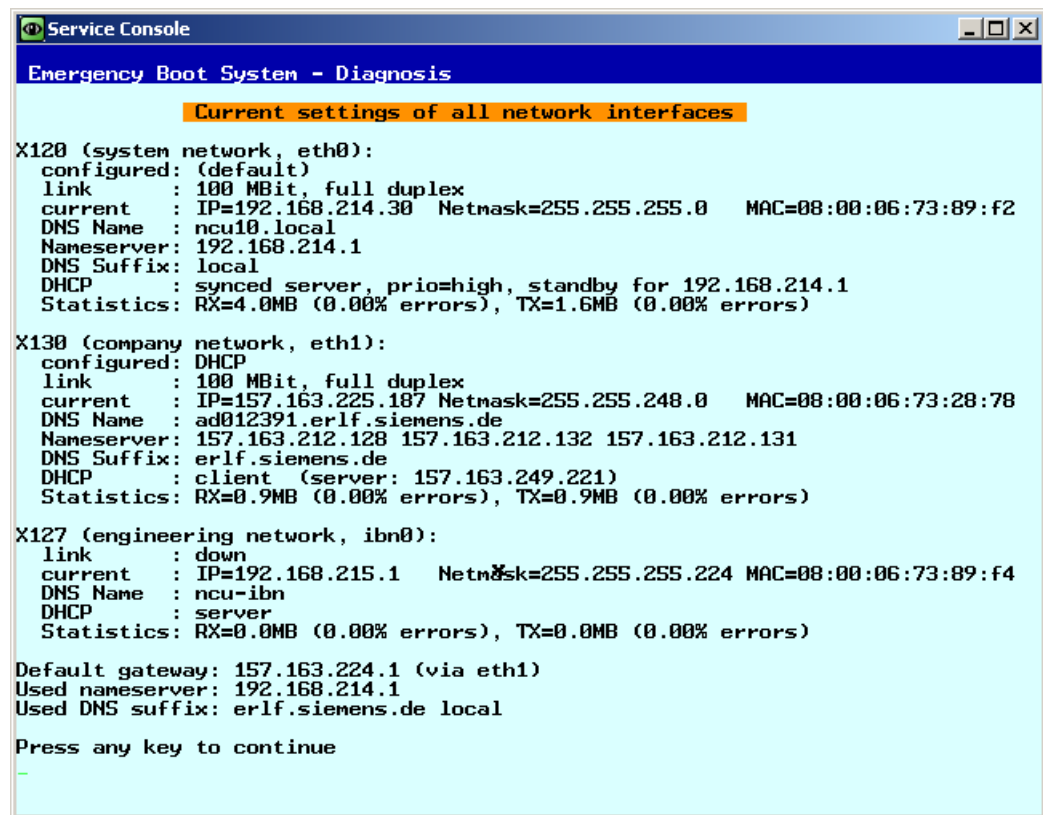


Figure 3-15 Diagnostics functions

System diagnosis	Description
Show all network settings	Connection data of all Ethernet interfaces is displayed.
Show network settings X120 (system network)	The connection data of interface X120 is displayed.
Show network settings X127 (engineering network)	The connection data of interface X127 is displayed.
Show network settings X130 (factory network)	The connection data of interface X130 is displayed.

System diagnosis	Description
Show card identification	The following data of the system CompactFlash Card is output: • Serial number • System software version • Hardware version (NCU)
Check card integrity	Checking the CompactFlash Card system

Example: Connection data of all Ethernet interfaces of the NCU:



```

Service Console
Emergency Boot System - Diagnosis

Current settings of all network interfaces

X120 (system network, eth0):
  configured: (default)
  link       : 100 MBit, full duplex
  current    : IP=192.168.214.30 Netmask=255.255.255.0   MAC=08:00:06:73:89:f2
  DNS Name   : ncu10.local
  Nameserver: 192.168.214.1
  DNS Suffix: local
  DHCP       : synced server, prio=high, standby for 192.168.214.1
  Statistics: RX=4.0MB (0.00% errors), TX=1.6MB (0.00% errors)

X130 (company network, eth1):
  configured: DHCP
  link       : 100 MBit, full duplex
  current    : IP=157.163.225.187 Netmask=255.255.248.0   MAC=08:00:06:73:28:78
  DNS Name   : ad012391.erlf.siemens.de
  Nameserver: 157.163.212.128 157.163.212.132 157.163.212.131
  DNS Suffix: erlf.siemens.de
  DHCP       : client (server: 157.163.249.221)
  Statistics: RX=0.9MB (0.00% errors), TX=0.9MB (0.00% errors)

X127 (engineering network, ibn0):
  link       : down
  current    : IP=192.168.215.1 Netmask=255.255.255.224 MAC=08:00:06:73:89:f4
  DNS Name   : ncu-ibn
  DHCP       : server
  Statistics: RX=0.0MB (0.00% errors), TX=0.0MB (0.00% errors)

Default gateway: 157.163.224.1 (via eth1)
Used nameserver: 192.168.214.1
Used DNS suffix: erlf.siemens.de local

Press any key to continue
-

```

Figure 3-16 NCU: Interfaces

See also

This is how you check the CompactFlash Card (Page 87)

3.6.2 This is how you change the system settings

Modify NCU Settings

You can change the following system settings in this menu:

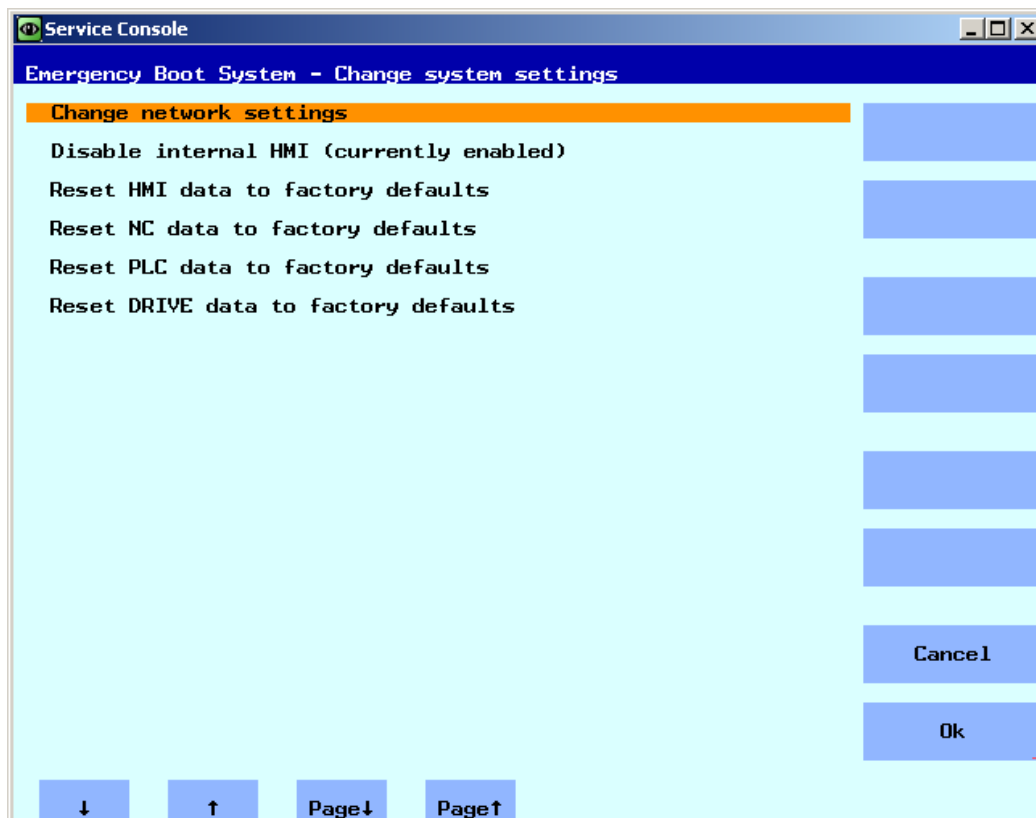


Figure 3-17 Changing system settings

Menu item	Description
Change network settings	Changing network settings of the NCU.
Disable internal HMI (currently enabled)	Activating/deactivating an HMI task
Reset HMI data to factory defaults	Restoring the delivery condition of HMI data.
Reset NC data to factory defaults	Restoring the delivery condition of NC data.
Reset PLC data to factory defaults	Restoring the delivery condition of PLC data.
Reset DRIVE data to factory defaults	Restoring the delivery condition of drive data.

NOTICE

Reset ... data to factory defaults

The selected data are reset to the state when delivered, therefore overwriting all of the previous specific commissioning settings that have been made.

3.6.3 This is how you check the CompactFlash Card

Displaying identification data

Procedure:

1. In the main menu, select the menu item "Diagnosis".

The following menu is displayed:

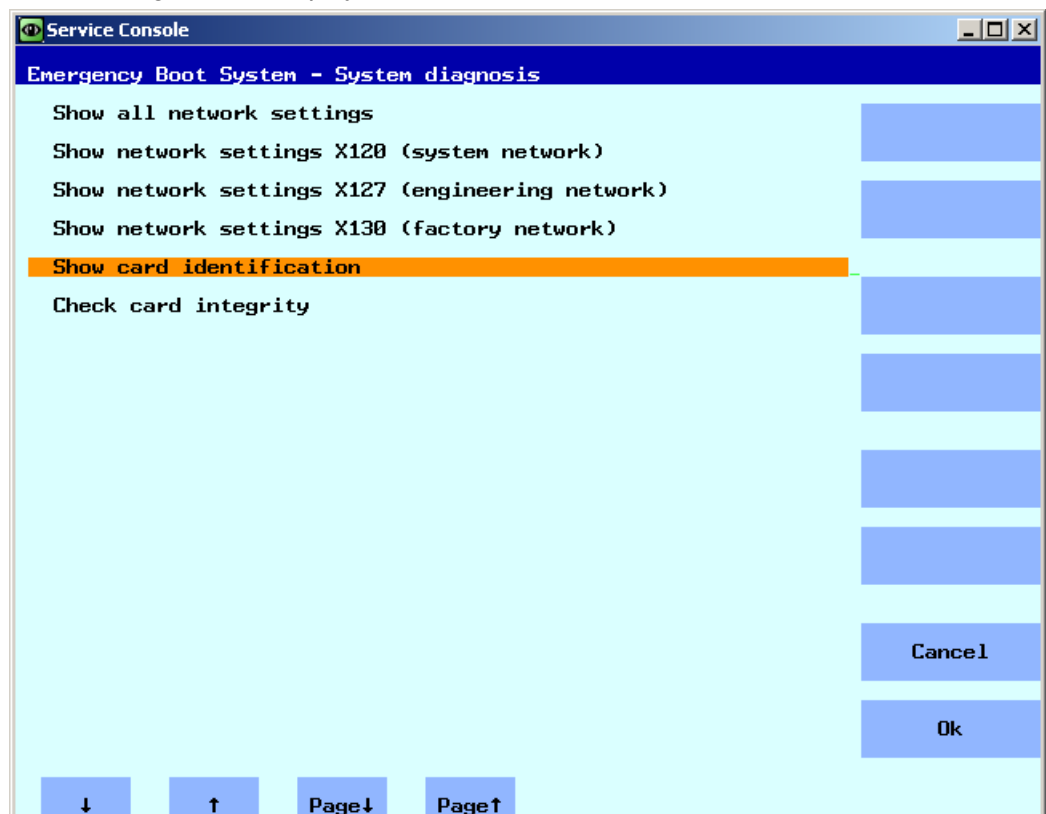


Figure 3-18 System diagnostics

2. Select the menu item "Show card identification".

The following data for identification are output:

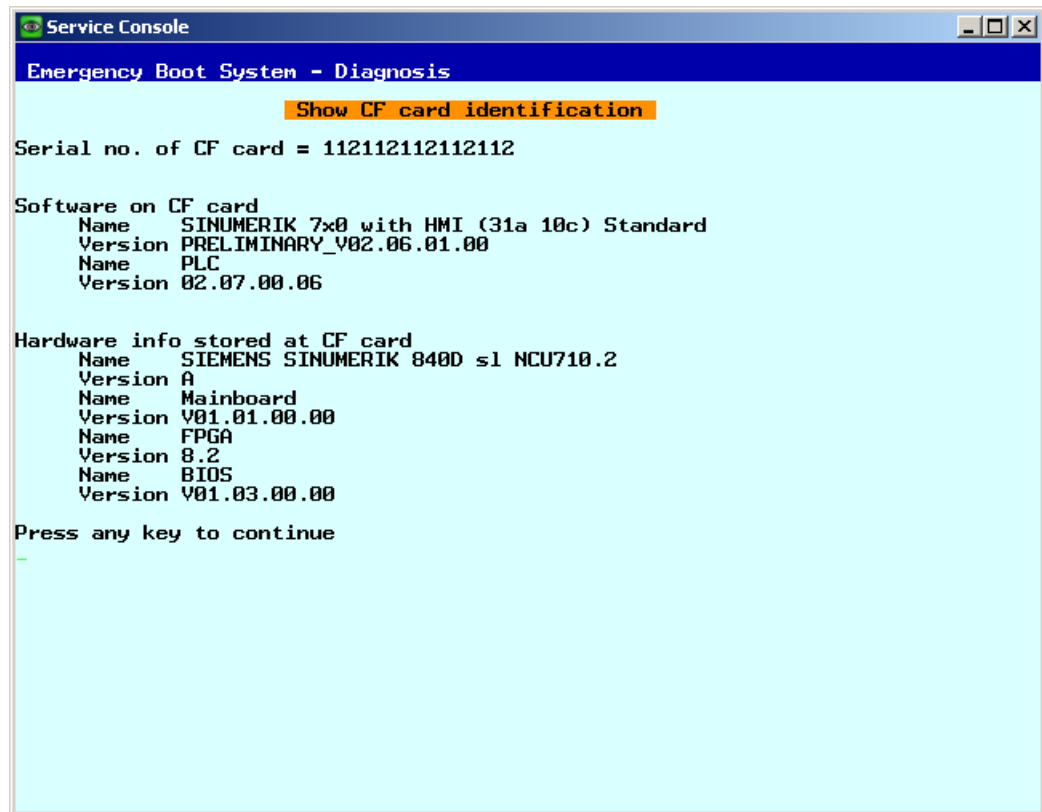


Figure 3-19 Identification

3. Select the menu item "Check card integrity".

The CompactFlash Card is checked for errors:

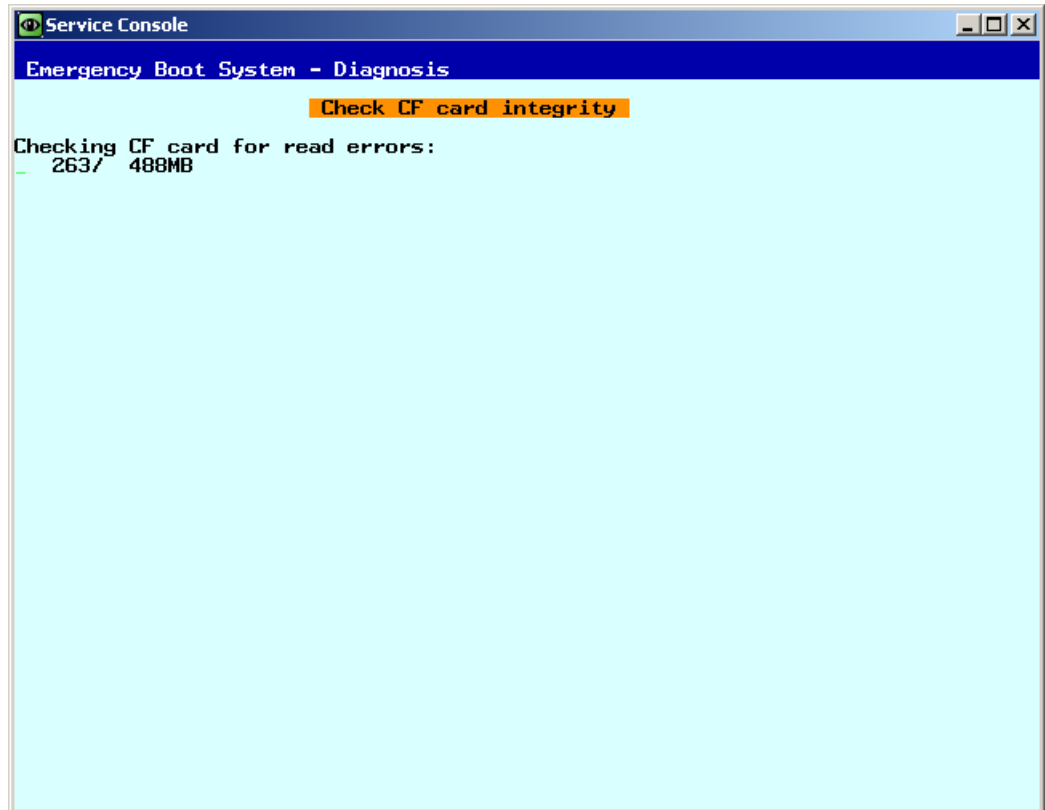


Figure 3-20 Check

Note

This procedure can take several minutes.

3.6.4 Calling the VNC Viewer

Purpose

The VNC Viewer is used to connect directly with a VNC Server and, for example, to operate an HMI application.

Furthermore, it enables you to call up a list of all the network devices, thereby obtaining an overview of the system network.

Starting the VNC Viewer

Procedure:

1. In the main menu, select the menu item "VNC Viewer".

The following menu is displayed:

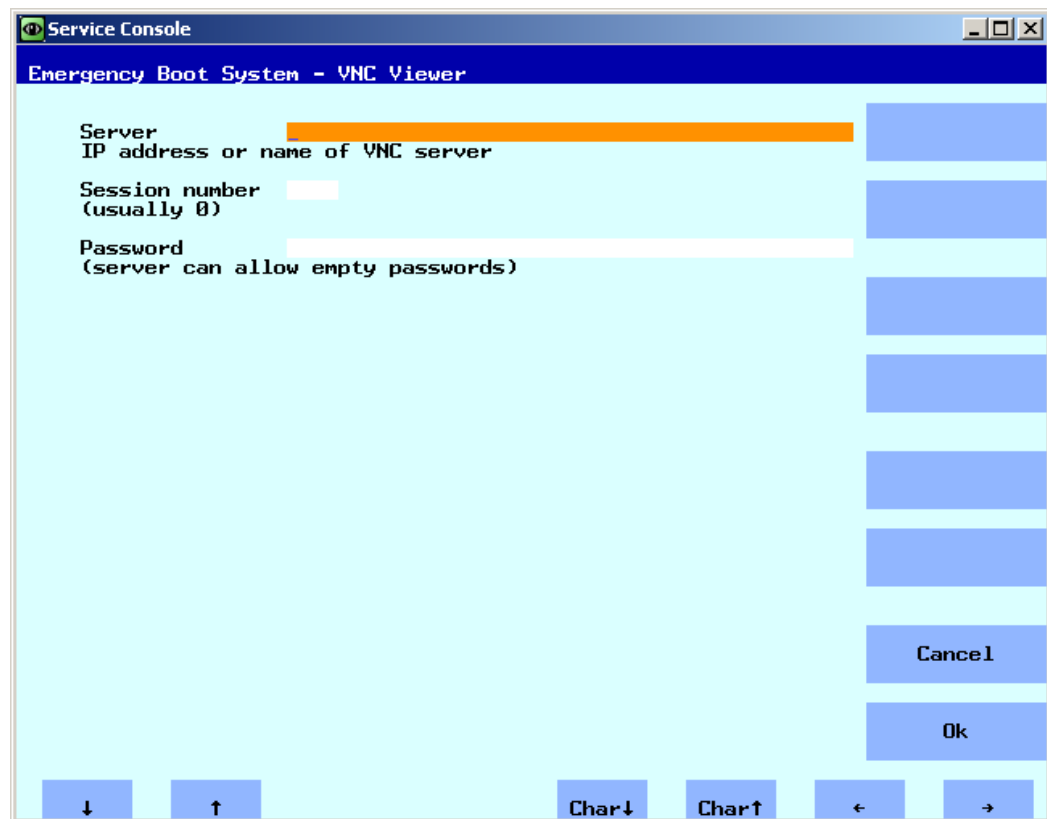


Figure 3-21 VNC Viewer

2. To start the VNC Viewer, enter the following connection data:
 - Server name, e.g. IP address or DNS.
 - The password is optional and can also be left empty.
 - The following assignment applies to the "Session number":

Number	Meaning
Session 0	HMI-Applikation
Session 4	Command Shell
Session 5	System logfile
Session 6	System Network Center (SNC)

3.6.5 WinSCP and PuTTY

Use

The programs WinSCP and PuTTY are freely-available open source programs for Windows. WinSCP is intended especially for transferring files from and to Linux systems, PuTTY for the remote operation of a command shell.

- WinSCP can be downloaded via the following link: <http://winscp.net/eng/download.php> (<http://winscp.net/eng/download.php>) (Installation Package).

WinSCP also offers a "command shell" that is limited so that commands can be issued, but no callbacks can be answered.

- PuTTY, by contrast, offers a complete command shell.

PuTTY web page: <http://www.chiark.greenend.org.uk/~sgtatham/putty>
(<http://www.chiark.greenend.org.uk/~sgtatham/putty>)

With both programs, a service technician can log onto the NCU and carry out service tasks. The username 'manufact' with the password 'SUNRISE' is available for the Siemens service technician.

Starting WinSCP

WinSCP is started from Windows after the service technician has logged onto the NCU with which he is connected using the corresponding authorization (e.g. as user "manufact" with password "SUNRISE").

From the "Commands" menu, select "Open terminal" to open a command shell. There you can carry out the service commands in the usual way.

Appendix

A.1 Abbreviations

CF	CompactFlash card: Memory card
CFS	Cluster File System
DCK	Direct Control Keys: Direct control keys
DCP	Discovery and Basic Configuration Protocol
DHCP	Dynamic Host Configuration Protocol: Dynamic assignment of an IP address and other configuration parameters on a computer in a network
DNS	Domain Name System: Conversion of domain names into IP addresses
EBS	Emergency Boot System
EKS	Electronic Key System: System to check the identity of a user (authentication system)
EUNA	End User Notification Administration
HMI	Human Machine Interface: Operator interface
IRT	Isochronous Realtime (Ethernet)
LLDP	Link Layer Discovery Protocol: multi-vendor Layer 2 Protocol defined in accordance with the IEEE-802.1AB standard, allows information to be exchanged between devices.
MAC	Media Access Control: The MAC address is a 48-bit Ethernet ID.
MCP	Machine Control Panel: Machine control panel
MPI	Multi-Point Interface: Multiple interface
MUI	Multilanguage User Interface
NAT	Network Address Translation
NCK	Numerical Control Kernel: NC kernel with block preparation, travel range, etc.
NCU	Numerical Control Unit: NCK hardware unit
NRT	Non-Realtime (Ethernet)
NTFS	New Technology File System
NTP	Network Time Protocol: Standard for synchronizing clocks in the entire network
NTPD	NTP Daemon: Utility that runs in the background and does not have to be started by the user.
PCU	PC Unit: Computer unit
PDEV	Physical device
PG	Programming device
PLC	Programmable Logic Control: Programmable logic controller
RAM	Random Access Memory: Program memory which can be read and written into
RDY	Ready: The system is ready to operate.
RFC	Remote Function Call
SNMP	Simple Network Management Protocol (network protocol for monitoring and controlling network elements such as routers, servers, switches, and printers from a central station).
SSD	Solid State Drive
SSH	Secure Shell: Protocol for an encrypted network connection with a remote device

Appendix

A.1 Abbreviations

TCU	Thin Client Unit
TFTP	Trivial File Transfer Protocol: Very simple data transmission protocol
UDP	User Datagram Protocol: NTP is mostly processed via UDP.
USB	Universal Serial Bus
UPS	Uninterruptible power supply
UTC	Universal Time, Coordinated: Coordinated Universal Time (previously: Greenwich Mean Time)
VNC	Virtual Network Computing

Glossary

CFS (Compressed File System)

A CFS (file extension ".cfs") is a compressed file system, similar to a zip file. It contains files and subdirectories that look like normal files on the controller at runtime. Files and directories contained in a CFS cannot be changed. They are decompressed at runtime as required.

Network interface

The network interface is an interface that enables network communication. These are the Ethernet interfaces on the NCU.

NFS (Network File System)

NFS is the most common protocol for remote file systems in the world of Unix, and is also available for Windows. NFS is closely based on the Unix privilege model – each time a file is accessed, a UID and GID are supplied which the server then uses to decide whether the operation is permitted. The server relies on the client to provide the correct IDs.

Remote File System

A file system that is contacted over the network. The files are physically located on another computer in the network (the "server"), but appear locally the same as all other files. Operations performed on these files are sent via the network to the server, instead of being executed directly on a local storage medium (such as a hard drive or CompactFlash Card).

As a server usually exports more than one file system, a name for the required file system must also be entered in addition to the name of the server.

SMB (Server Message Block)

SMB is the underlying protocol of MS Windows file systems (also known as drives, releases, shares, etc.). SMB connections are always active in the context of a specific user, who must be known to the server. Exported file systems have a name (release name), by which they can be addressed. The client does not need to know the concrete path on the server.

Subsystem

A subsystem is a CFS that not only contains a collection of files, but also executes a program, for example, at runtime. To do this, the CFS contains a script that is used to control the starting and stopping of this program.

For this reason, only administrators are permitted to set up NFS file systems, and NFS is usually only implemented in uniformly administrated environments. Exported file systems on the server are addressed directly on the server via their path.

VNC (Virtual Network Computing)

Virtual Network Computing is a software that displays the screen contents of a remote computer, with a running VNC server, on a local computer, with a running VNC viewer, and in return sends keyboard and mouse movements of the local computer to the remote computer.

Index

7

7-segment display, 9

A

Applications, 59

B

basesys.ini, 15
 CheckTCUforUpdatePackage, 29
 DHCPClientID, 18
 DHCPDNoMasterWait, 21
 DisableCompanyNet, 16
 DisableIBNForwarding, 22
 DisableIBNNet, 22
 DisableNATRouting, 21
 DisableNTPTimeSync, 28
 DisablePLCTimeSync, 29
 DisableSubsystems, 28
 DisableSysNet, 18
 Domain, 18
 EnableCoreDumps, 29
 EnableDHCP_IBNNet, 22
 EnableDHCPD_SysNet, 19
 EnableSysNetToIBNForwarding, 22
 ExternalDcpEnabled, 24
 ExternalIP, 16
 ExternalLldpEnabled, 25
 ExternalLldpTLVsTxEnabled, 26
 ExternalNetMask, 16
 FirewallOpenPort, 28
 Gateway, 16
 Host name, 17
 InternalDcpEnabled, 24
 InternalDNSDomain, 21
 InternalDynRangeEnd, 20
 InternalDynRangeStart, 20
 InternalIP, 19
 InternalIP_Alias, 19
 InternalLldpEnabled, 25
 InternalLldpTLVsTxEnabled, 25
 InternalNetMask, 19
 InternalNetMask_Alias, 19

LogFileLimit, 27
Nameservers, 17
SNMPAutLocation, 23
SNMPContact, 23
SNMPFunction, 23
SNMPLocation, 23
SNMPStation, 24
SyncModeDHCPD_SysNet, 20
Timeservers, 17
Timezone, 26

C

CFS (Compressed File System), 38
Coding switches, 7
Command
 Backup, 50
 Check-cf, 40
 Closeport, 41
 Connect, 41
 Disable, 43
 Disconnect, 44
 Enable, 45
 Help, 39
 Openport, 47
 Port status, 48
 Reboot, 48
 Restart, 48
 Restore, 49
 Save, 50
 Show, 52
 Start, 56
 Stop, 56
CompactFlash Card
 Memory partitioning, 6
 Partitions, 5

D

data
 Restoring, 78
DHCP synchronization, 46

E

Emergency Boot System, 59
Ethernet interfaces, 13

I

Interfaces

- Ethernet, 37
- NCU, 13

IP address

- Display, 14

L

- Load operating system, 9

N

- NCK commissioning switch, 7
- NCU booting, 8
- Network drive, 64
- Network interface, 13
- NFS (Network File System), 38

P

- PLC mode switch, 7
- Privilege, 35

R

- Remote File System, 38
- Replacement part, 65
- Reset
 - Button, 8
 - Execute, 8

S

Section

- [DCP], 24
- [ExternalInterface], 16
- [IBNInterface], 22
- [InternalInterface], 18
- [LinuxBase], 26
- [LLDP], 25
- [SNMP], 23
- Service command
 - Application, 35
 - Rights, 35
 - Syntax, 36
- Service menu, 62
- Service system
 - Backing up data, 75

- Creating for the NCU, 60

- Restoring data, 76

- Service Tool WinSCP, 91

- SMB (Server Message Block), 38

- Subnet mask, 14

- Subsystem, 38

- Designation, 37

- Synchronization

- DHCP, 46

- System

- Diagnostics, 9

- Log file, 5, 29

- System error, 11

T

- Time zone, 26

U

Update

- BIOS, 80

- Firmware, 80

- User group, 35

- Users (default setting), 6

V

- VNC (Virtual Network Computing), 13

W

- WinSCP, 91

SIEMENS

SINUMERIK

SINUMERIK 840D sl PCU-Basesoftware (IM8)

Commissioning Manual

Commissioning the system

1

Configuring the system

2

Install software and updates

3

Backing up and restoring
data

4

Service and diagnostics

5

List of Abbreviations

A

Valid for:

SINUMERIK PCU 50.5
with PCU base software V5XP1.4

SINUMERIK PCU 50.3
with PCU base software V8.6 SP3

02/2014

6FC5397-1DP40-4BA0

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

DANGER

indicates that death or severe personal injury will result if proper precautions are not taken.

WARNING

indicates that death or severe personal injury may result if proper precautions are not taken.

CAUTION

indicates that minor personal injury can result if proper precautions are not taken.

NOTICE

indicates that property damage can result if proper precautions are not taken.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

WARNING

Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Table of contents

1	Commissioning the system	5
1.1	Delivery condition of the system	5
1.1.1	Hard disk partitions	7
1.1.2	System features	9
1.2	User administration	11
1.2.1	Which users are set up?	11
1.2.2	User settings	13
1.2.3	This is how you create a new user.....	21
1.2.4	This is how you administer users.....	22
1.3	Boot up behavior of the PCU	23
1.3.1	Dialogs when booting.....	23
1.3.2	Boot up of the PCU: no HMI program installed.....	26
1.3.3	Boot up of the PCU: HMI program already installed.....	28
1.3.4	Setting the screen resolution	30
1.4	BIOS settings for PCU 50.3	33
1.5	BIOS settings for PCU 50.5	38
2	Configuring the system	43
2.1	System settings.....	43
2.1.1	Configuring the communication via MPI	43
2.1.2	How to change the name of the PCU	44
2.1.3	This is how you set the IP address of the PCU	45
2.1.4	How to add the PCU to a domain	47
2.1.5	How to connect an external monitor	48
2.1.6	This is how you switch off the USB interfaces	49
2.2	Configuring a customized operator interface	50
2.2.1	How to select the language for the Windows system	50
2.2.2	Displaying the boot screen.....	53
2.2.3	Changing the background of the service desktop.....	54
2.3	Customized settings during boot up.....	55
2.3.1	Configuring key filters for an HMI program	55
2.3.2	Saving service desktop settings.....	58
2.3.3	Starting programs during boot up	59
2.3.4	Starting OEM programs	60
2.3.5	Starting applications in service mode	61
2.4	Configuring the SITOP UPS module for use with the PCU Base Software.....	62
2.4.1	SITOP modules for the PCU 50.5.....	62
2.4.2	Configuring SITOP software for PCU Base Software.....	64
2.4.3	Parameterizing a delay time for quitting the HMI software	67
2.4.4	Hardware configuration of the SITOP UPS module.....	69
2.5	Configuration of the Service Center.....	73
2.5.1	Overview	73

2.5.2	Configuration of the network adapter	74
2.5.3	Configuration of the host	76
3	Install software and updates	77
3.1	Installing SINUMERIK products	77
3.2	Installation via service desktop	78
3.3	How to install and authorize SIMATIC STEP 7	82
3.4	How to install additional languages under Windows XP (DVD)	84
4	Backing up and restoring data	87
4.1	Overview	87
4.2	Starting the Service Center	87
4.3	Functions of the Service Center	88
4.4	Network settings in the Service Center	90
4.5	Create a disk image of the SSD	93
4.6	Restore a disk image of the SSD	95
4.7	Create a disk image of a partition	97
4.8	Restore a disk image of a partition	99
4.9	Network settings in the Service Center	102
4.10	Starting Symantec Ghost directly	104
4.11	Bootable USB flash drive	105
4.12	Operating a service PC/PC in the network	106
4.12.1	Overview	106
4.12.2	Connection options in the system network	107
4.12.3	Configuring routing in the network	109
4.12.4	Configure the network settings of the PG/PC	111
4.12.5	Release directory of the PC/PC in the network	114
4.12.6	Release drives C: and D: of the PCU 50.5 in the network for administrator accounts	118
4.12.7	Creating a shortcut to the network drive	119
4.13	Starting up the replacement SSD	120
5	Service and diagnostics	121
5.1	PCU Hardware Diagnostics	121
5.2	Evaluation of the 7-segment display	122
5.3	Setting of the service switch (PCU 50.5 only)	125
5.4	Enabling/disabling error log during boot up	126
5.5	How to search for stations within the system network	127
A	List of Abbreviations	135
A.1	Abbreviations	135
	Index	137

Commissioning the system

1.1 Delivery condition of the system

Overview

The PCU has onboard interfaces for communicating via Ethernet, MPI and PROFIBUS DP. The integrated free slots remain free for other tasks. The PCU is equipped with the Windows XP ProEmbSys operating system and for data backup tasks with the Symantec Ghost software.

Interfaces:

- Four USB ports (USB 2.0) offer points where a keyboard, mouse and other peripheral devices can be connected.
- For CF cards, there is a covered slot.
- Two internal PCI slots are available for specific expansions.

For commissioning:

- Two 7-segment displays and two LEDs are integrated for diagnostic purposes. They indicate the current operating status and display the BIOS error codes during startup.
- If the PCU is to be operated without an operator panel front, a monitor and an additional keyboard will also be required:
 - For diagnostics during startup of the PCU.
 - When installing a replacement hard disk
(alternatively, the hard disk can also be installed externally).

References: Operator Components and Networking Manual

Supplied software on the PCU

The software installed on delivery of the PCU includes the components below, among others:

MS Windows XP Professional SP3	
Internet Explorer	V 6.0
MPI driver (PCU 50.3 and PCU 50.5-C)	V 6.04.01.00
Symantec Ghost (default setting)	V 8.2 (incl. Ghost Explorer)
TCU Support (It is already installed and available on the hard disk under D:\Updates, if it must be re-installed.)	V 8.6

Documentation for all Ghost tools is supplied on the PCU's hard disk under E:\TOOLS

Note

For the system component versions contained in the PCU base software, see the C:\BaseVers.txt file.

Note

Acknowledgements

- This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)
 - This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (<http://www.openssl.org/>)
-

1.1.1 Hard disk partitions

Memory segmentation

The hard disk of the **PCU 50.3** has 40 GB of storage capacity and is segmented into a primary partition C and an expanded partition with the three logical drives D, E and F which operate with NTFS file access.

EMERGENCY (C:)

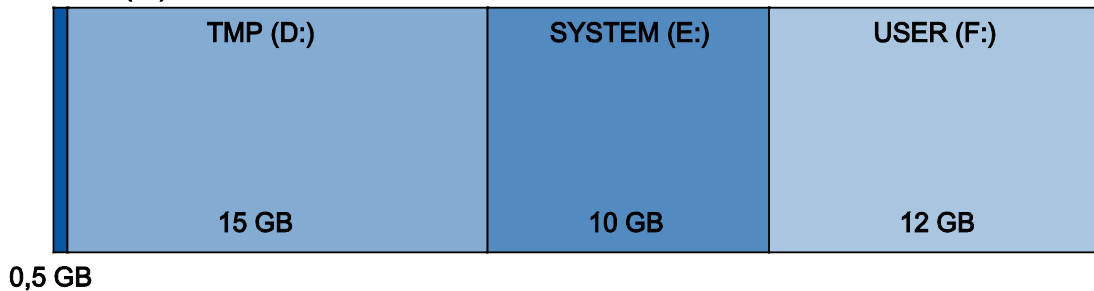


Figure 1-1 Division of the hard disk

The **PCU 50.5** is equipped with a solid state drive (SSD) with NTFS file access, which is also segmented to provide a primary partition C and an expanded partition with the three logical drives D, E and F. The SSD is available with 32 GB and 40 GB storage capacity.

EMERGENCY (C:)

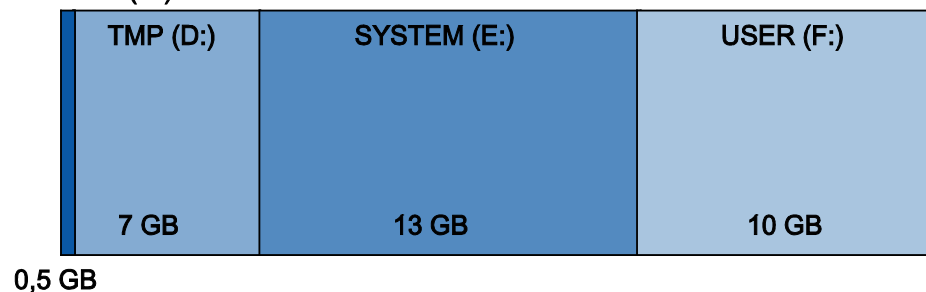


Figure 1-2 Segmentation of the SSD with 32 GB

Partitioning of the SSD with 40 GB is identical to that for partitioning of the hard disk.

Depending on the order, the operating software can already be installed when the unit is shipped. It is installed subsequently by the customer the first time the system starts up. For reasons of data security, the operating software and the Windows XP system software are distributed over the different hard disk partitions.

Content of the partitions

The individual partitions are intended for the following data or already contain this data:

EMERGENCY (C:)	Reserved for service tasks under WinPE 2005.
TMP (D:)	Used for storing Ghost images (e.g. of the delivery condition) and local backup images. Contains the installation directory where the software to be installed is first copied to from a remote PG/PC prior to the actual installation procedure.
SYSTEM (E:)	Reserved for the Windows XP software. The Windows XP software is available on the recovery media CD via network, for example to install drivers or updates as and when needed later on.
USER (F:)	For installing user programs. Applications such as HMI system software (incl. data storage and temporary data), STEP 7, OEM applications for HMI or customer-specific applications should only be installed here.

Note

Installing software

All of the applications must be exclusively installed on USER (F:) even if these applications have a different drive set as the default drive in their installation path.

The partition names EMERGENCY, TMP, SYSTEM, USER must not be changed; otherwise the "ServiceCenter" will no longer function.

See also

Installing SINUMERIK products (Page 77)

1.1.2 System features

Configuration of the operating system

For safety reasons, Windows XP has been preset as follows:

- The Autorun function is deactivated.
- Automatic Windows Update is deactivated.
- Monitoring and alerts for antivirus software and automatic update are deactivated.
- Links used to call up Internet Explorer from the service desktop and the start menu are removed.
- Remote Procedure Call (RPC) is possible for calls that are not connected.
- The firewall settings are activated on the network card Eth 1 and deactivated on Eth 2.

Changes to Windows services

Other default settings:

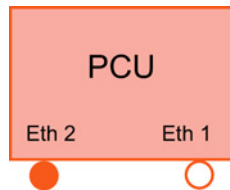
Windows services:	Start-up type:	
Computer Browser	Manual	(Not started)
Error Reporting Service	Disabled	
Portable Media Serial Number	Manual	(Not started)
SSDP Discovery Service	Disabled	
Universal Plug and Play Host	Disabled	
Web Client	Manual	(Not started)
Wireless Zero Configuration	Manual	(Not started)

Name of the PCU

Upon delivery of the system, a unique computer name is generated which can be read out under: "Start" → "Settings" → "Control Panel" → "System", "Computer Name" tab.

Pre-configuration of the PCU

The PCU has two Ethernet interfaces with default settings suitable to connect to SINUMERIK solution line:



Eth 1 is preset as a default DHCP client for connection to a company network.

Eth 2 is preset as a SINUMERIK DHCP server for connection to a system network. Eth 2 is preset to the fixed IP address 192.168.214.241.

See also

This is how you set the IP address of the PCU (Page 45)

References: Operator Components and Networking Manual

1.2 User administration

1.2.1 Which users are set up?

Preset users

Each user is from one user type and belongs to one user group. The user types are implemented under Windows in user groups with various user rights.

Upon delivery of the system, the following users are defined at the factory:

- **The "operator"**

In terms of type, the "operator" is classed as an HMI user and belongs to the operators' user group (operator.group). These operators have limited user rights under Windows.

- **The "user"**

In terms of type, the "user" is classed as an HMI user and belongs to the operators' user group (user.group). These operators have extended user rights under Windows.

- **The "auduser".**

The "auduser" is an HMI service user type of user and belongs to the system administrators' user group. The system administrators have the user rights of a local administrator under Windows.

User name	Password	User type	Windows user group	User Rights	User group
operator	operator	HMI (operator.group)	operator.group	Restricted	Operator
user	CUSTOMER	HMI (user.group)	user.group	"Power User"	Operator
auduser	SUNRISE	HMI+Service	Administrators	local administrator	System administrators
siemens	*****	---	---	---	System administrators

The individual user types differ in the following areas:

User type	Area of activity
HMI (operator.group)	• Boot up of the PCU • Operating the HMI program • Windows Desktop
HMI (user.group)	• Boot up of the PCU • Operating the HMI program • Windows desktop
HMI+Service	• Boot up of the PCU • Operating the HMI program • Service Desktop

Service Desktop

The service desktop provides the HMI+Service user with a Windows desktop, which is expanded to include tools and functions for service tasks, such as manage users, install software, save/restore data, check system integrity, etc.

Windows desktop

The HMI user is able to use a Windows desktop which can be expanded to suit his or her individual needs.

Starting ServiceCenter Users



Using this link on the service desktop, you can start ServiceCenter Users **as an HMI+Service user**. In ServiceCenter Users, the boot behavior of the PCU as well as settings for the HMI program and desktop are set globally for all users and individually for specific users.

The users are managed in the "ServiceCenter Users", so that the commissioner/service technician no longer has to make corresponding settings directly in the registry.

1.2.2 User settings

"Global Settings"

In the ServiceCenter Users, as **HMI service user**, you set parameters under "Global Settings" for the system behavior:

- When the PCU is booting up
- When the HMI program is starting
- With reference to the desktop

Note

The following diagrams show, as an example, an application with the SINUMERIK Operate software installed.

Using ServiceCenter Users

Buttons:

- Use "Exit" to quit ServiceCenter Users.
- If you press the "Reboot" button, the system immediately reboots (without prior prompting).

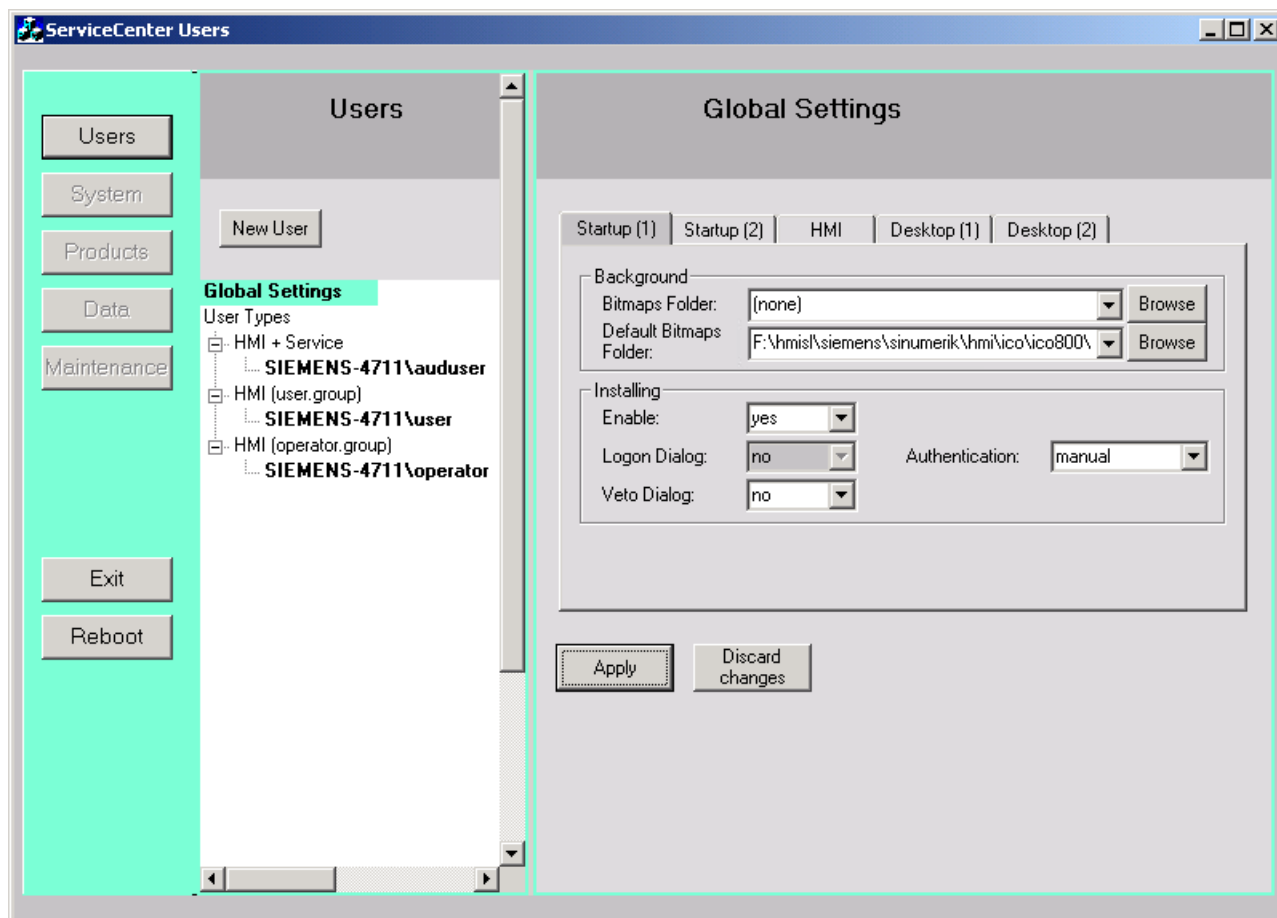
"Startup (1)" tab

Figure 1-3 Global Settings: Startup (1) tab

Table 1- 1 The default is marked in "**bold**".

Startup (1) tab	Option	Effect
Background		
Bitmaps folder:	F:\hmis\siemens\slnumerik\hmi\ico\ ...	Directory with boot screens
Default bitmap folder:	F:\hmis\siemens\slnumerik\hmi\ico\ico800	Directory containing default boot screens
Installing		
Enable:	"yes"	Authorization is in place to install user software during booting.
	"no"	No installation authorization
Logon dialog:	"yes"	Display "Installing Logon" dialog.
	"no"	Do not display "Installing Logon" dialog.
Veto Dialog:	"no"	For pending installation during the boot up: No prompting, the installation begins immediately.

Startup (1) tab	Option	Effect
	"yes"	For pending installation during the boot up: Prompt asking whether to install.
Authentication:	"manual"	Authentication of an HMI+Service user in the "Installing Logon" dialog is undertaken manually.
	"EKS"	Authentication of an HMI+Service user in the "Installing Logon" dialog involves EKS. The "Installing Logon" dialog appears if authentication by EKS has failed in the background.
	"manual+EKS"	Authentication of an HMI+Service user in the "Installing Logon" dialog either involves EKS or is performed manually.

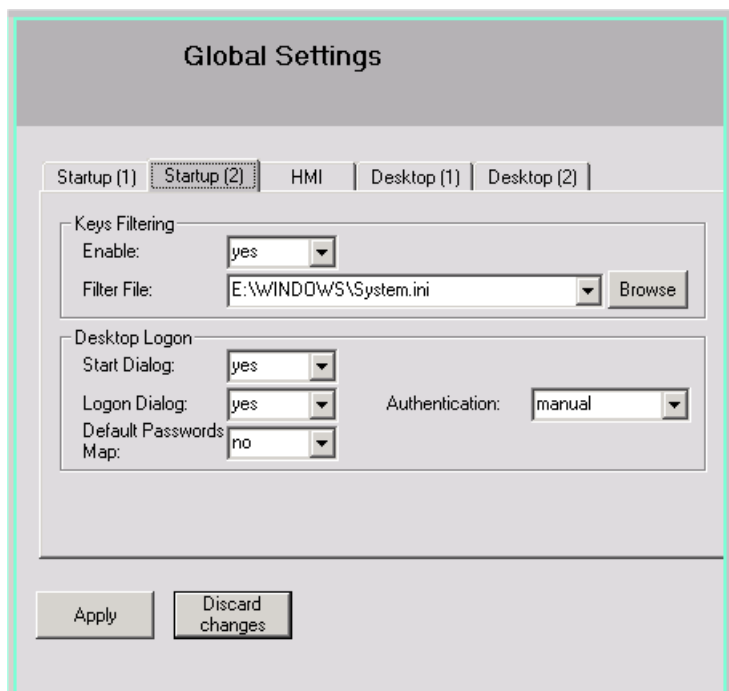
"Startup (2)" tab

Figure 1-4 "Startup (2)" tab

Table 1- 2 The default is marked in "**bold**".

Startup (2) tab	Option	Effect
Keys, filtering		
Enable	"yes"	Key filtering is activated.
	"no"	Key filtering not activated.
Filter file	E:\WINDOWS\System.ini	Select the file with filter settings
Desktop Logon		
Start dialog:	"yes"	Display "Desktop Access" dialog
	"no"	Do not display "Desktop Access" dialog
Logon Dialog:	"yes"	Display "Desktop Logon" dialog.
	"no"	Do not display "Desktop Logon" dialog.
Default password map:	"no"	The password must be entered exactly as specified.
	"yes"	The password is not case-sensitive.
Authentication:	"manual"	Authentication with user name and password (manual)
	"EKS"	Authentication with EKS
	"manual+EKS"	Either authentication with user name and password or with EKS

"HMI" tab

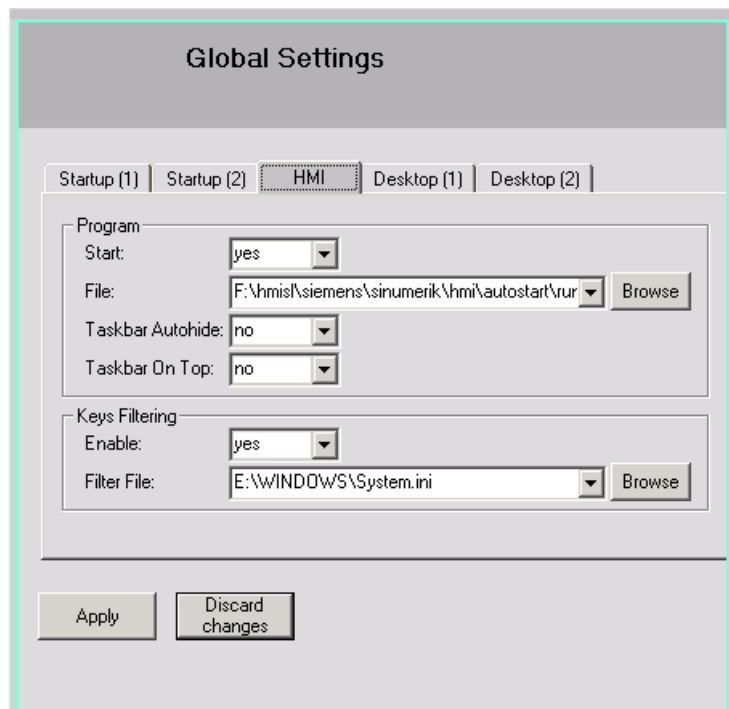


Figure 1-5 "HMI" tab

Table 1- 3 The default is marked in "**bold**".

"HMI" tab	Option	Effect
Keys Filtering		
Enable	"yes"	Key filtering is activated.
	"no"	Key filtering not activated.
Filter file	E:\WINDOWS\System.ini	Select the file with filter settings
Program		
Start:	"yes"	The HMI program is started.
	"no"	The HMI program is not started.
File:	F:\hmis\siemens\sinumerik\hmi\autostart\run_hmi.exe	Select the HMI program
Task bar autohide:	"no"	HMI program: Hide start task bar
	"yes"	HMI program: Show start task bar
Task bar on top:	"no"	HMI program: Start task bar in the background
	"yes"	HMI program: Start task bar always visible

"Desktop (1)" tab

Global Settings

Startup (1) | Startup (2) | HMI | **Desktop (1)** | Desktop (2)

Enable: **no**

Content

Icons Autoshow Tools: **yes** Startmenu Autoshow Tools: **yes**

Background

Bitmaps Folder: (none) Browse

Default Bitmaps Folder: (none) Browse

Taskbar

Autohide: **no** On Top: **no**

Apply Discard changes

Figure 1-6 "Desktop (1)" tab

Table 1- 4 The default is marked in "**bold**".

"Desktop (1)" tab	Option	Effect
Enable:	"no"	Desktop not displayed (Default: for HMI user)
	"yes"	Displaying the desktop (Default: only for HMI+Service user)
Content		
Icons Autoshow Tools:	"no"	Do not display links on the desktop.
	"yes"	Display links on the desktop.
Startmenu Autoshow Tools:	"no"	Do not display programs in the "Start menu"
	"yes"	Display programs in the "Start menu".
Background		
Bitmaps folder:	(none)	Directory with background screens for the desktop
Default bitmap folder:	(none)	Directory with default background screens for the desktop
Task bar		
Autohide:	"no"	Desktop: Hide start task bar
	"yes"	Desktop: Show start task bar
On Top:	"no"	Desktop: Start task bar in the background
	"yes"	Desktop: Start task bar always visible

"Desktop (2)" tab

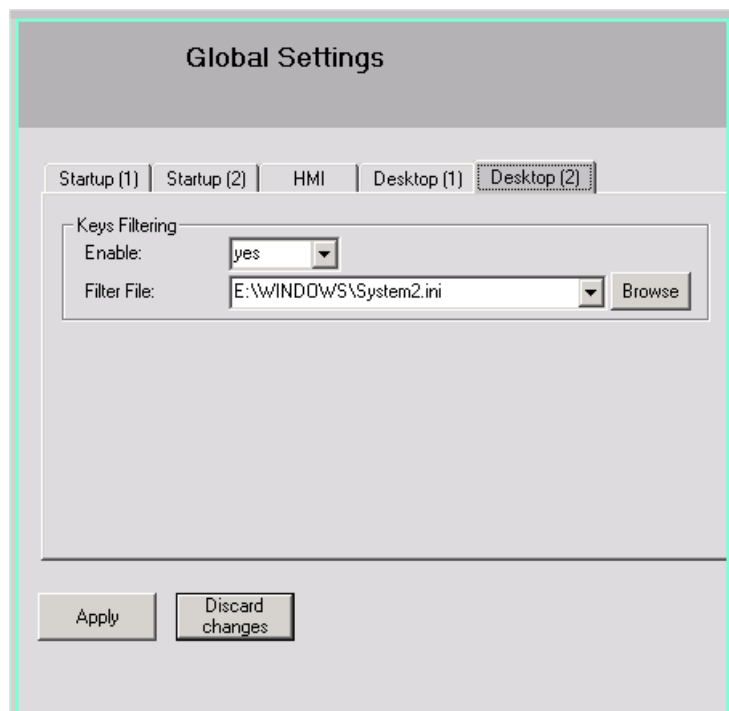


Figure 1-7 SCU "Desktop (2)" tab

Table 1- 5 The default is marked in "**bold**".

"Desktop (2)" tab	Option	Effect
Keys, filtering		
Enable	"yes"	Key filtering is activated.
	"no"	Key filtering not activated.
Filter file	E:\WINDOWS\System2.ini	Select the file with filter settings

Principle of passing on

The settings made under "Global Settings" are passed on to individual users. The settings which are passed on can still be adapted to suit each user.

Example: Specific user settings

The settings under "Global Settings" are inherited by all users:

1. Left-click the user in the "Users" list.
2. After entering the password, you can adapt the settings as required.

The settings that cannot be changed are displayed with a gray background.

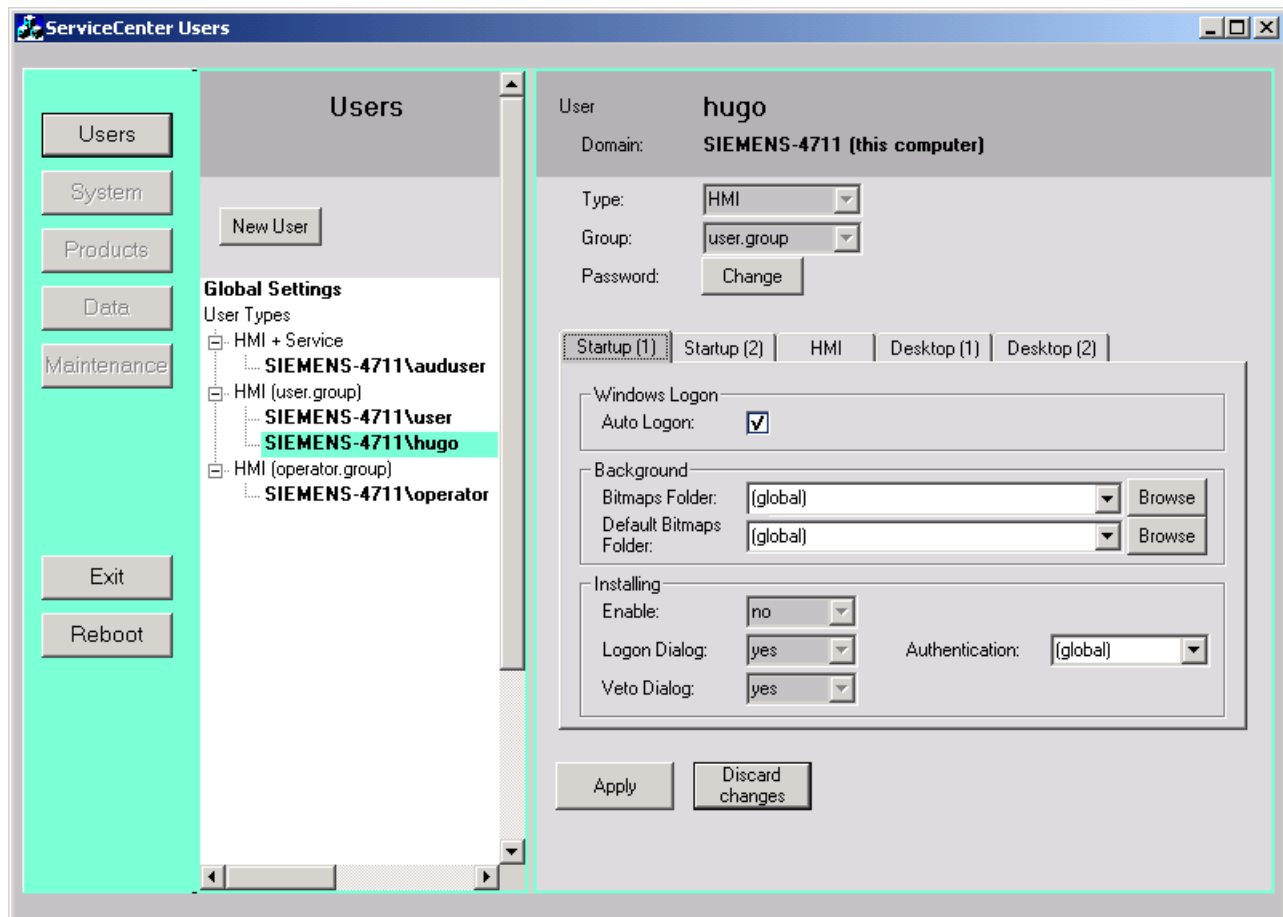


Figure 1-8 Specific settings for the "HMI (user.group)" user

1.2.3 This is how you create a new user

Creating new users

Procedure:

1. Click on "New User" button.
2. Enter a user name.
3. Assign a user type, e.g. "HMI", and a user group to the user.

The Windows user group is only active locally on the PCU.

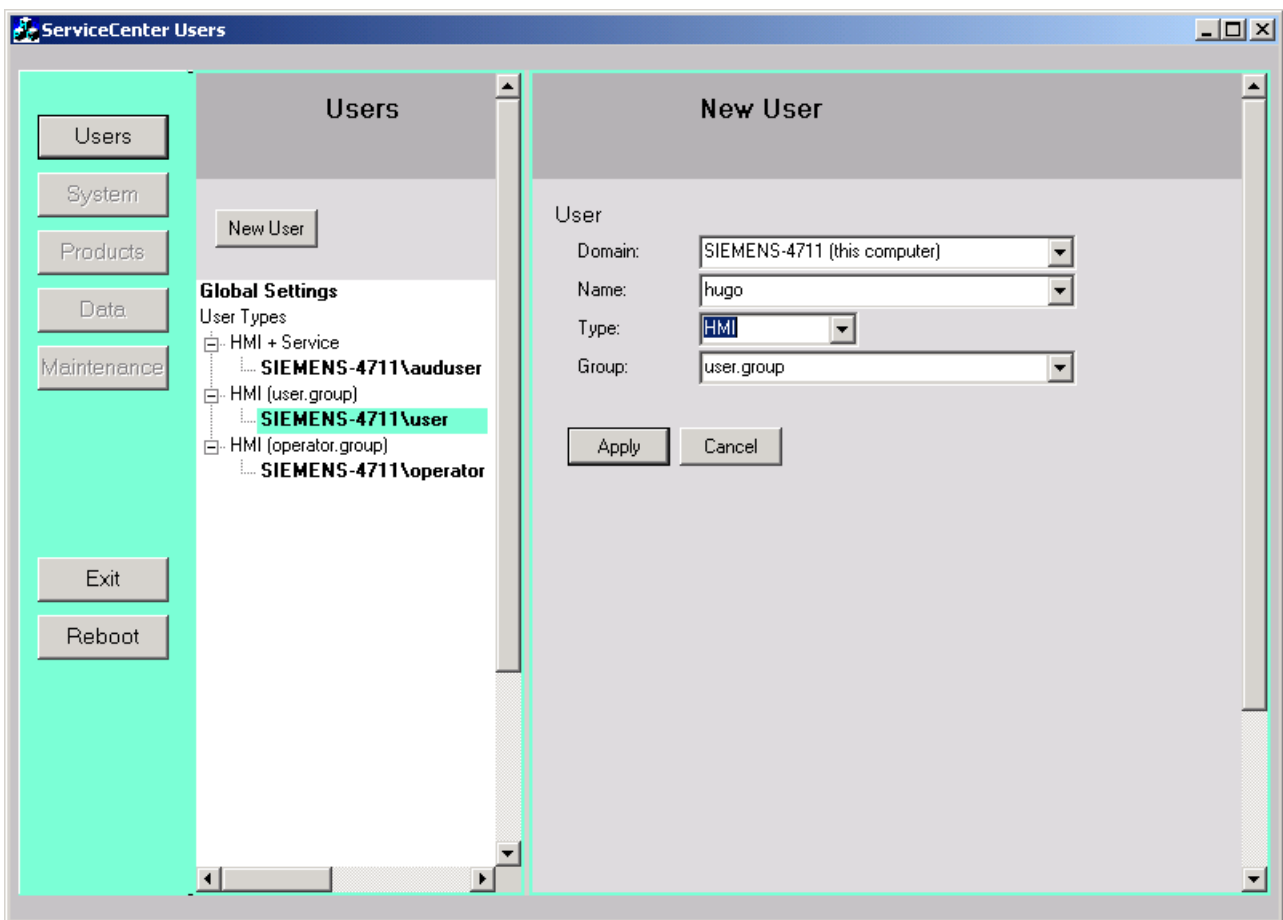


Figure 1-9 ServiceCenter Users: New user

4. Confirm with "Apply".
5. You will then be asked to specify a password.
6. After confirming with "OK", the new user is created and displayed in the list under "Users".

Adding new users from a domain

If the PCU is a member of a domain, users already existing in this domain can be added as HMI users or HMI+Service users:

1. Click on the "New User" button.
2. To do this, select the corresponding entry in the "Domain" list and a user from the "Name" list.
3. Assign a user type and a user group to the user.

The Windows user group is only active locally on the PCU.

See also

How to add the PCU to a domain (Page 47)

1.2.4 This is how you administer users

Overview

As an HMI+Service user, you can execute the following tasks in the Users ServiceCenter:

- Change user names.
- Delete users.

Change the user name

1. Left or right-click the user who you wish to rename in the list.
2. To change the user name, select "Rename" from the pop-up menu.
3. Enter a new name and confirm with OK.

Deleting a user

To delete a user again, follow these steps:

1. Left or right-click the user who you wish to delete in the list.
2. To delete the user, select "Delete" from the short-cut menu using righthand mouse key.

Result: The user is deleted after confirming the confirmation prompt.

1.3 Boot up behavior of the PCU

1.3.1 Dialogs when booting

Selection during startup

The following dialogs are shown during startup:

- **"Installing Logon" dialog**

This dialog is displayed if an "Install" directory is found on a bootable storage medium (e.g. USB-FlashDrive) during startup.

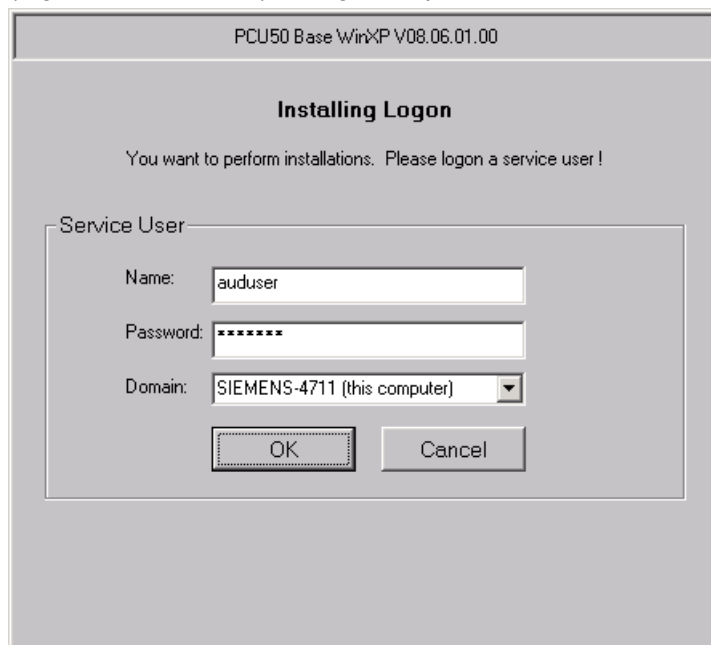


Figure 1-10 Installing Logon

- **"Desktop Access" dialog**

- This dialog is displayed if **either** no HMI program is installed, the installation has been skipped **or** the start of the HMI program has been deactivated during startup.

Displaying this dialog can be suppressed through configuration in the global or user settings. "Desktop" is then selected as the default.

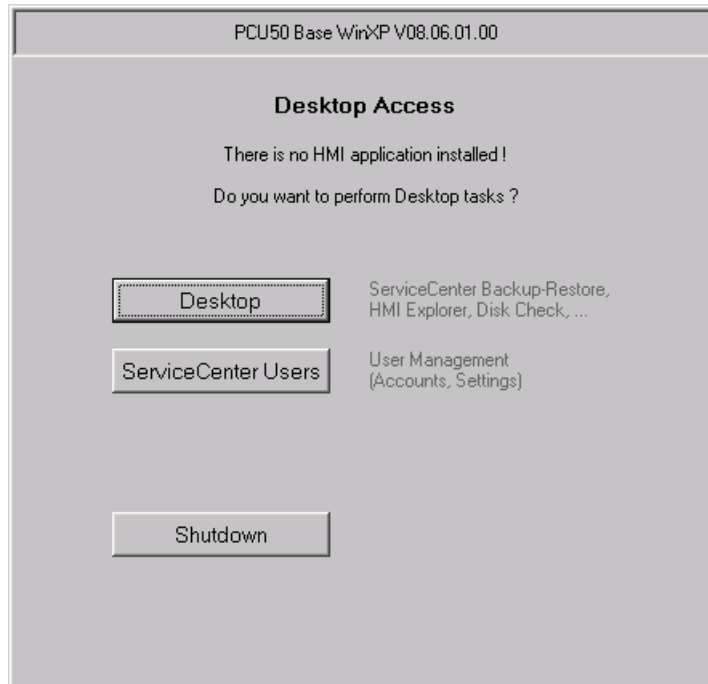


Figure 1-11 Desktop Access (example without HMI program)

Note

If an HMI program is installed, the "Start HMI" button is also available in the "Desktop Access" dialog.

- **"Desktop Logon" dialog**

The "Desktop Logon" dialog enables the user to log on to the Desktop. This dialog also appears

- If no valid user data has been provided by the authentication system (EKS).
- If authentication on the "Startup(1)" tab is set to "manual".

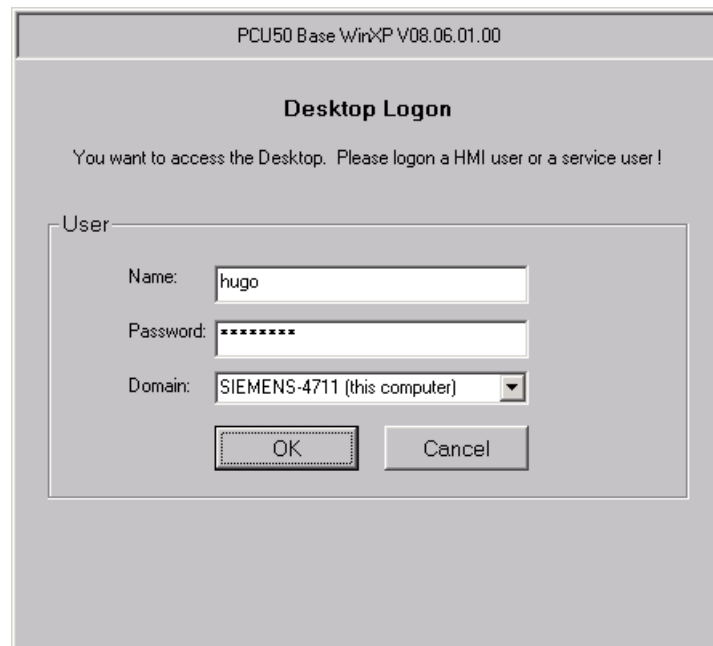


Figure 1-12 Desktop Logon (with authentication: "manual")

Note

Setting the access level via EKS

If an EKS unit is available, the following behavior applies:

- If an EKS unit is active, i.e. the key is inserted and can be evaluated, then the key information for the EKS unit alone determines the active access level. The access level in the HMI program cannot be changed when EKS is active.
 - If the key is withdrawn from the active EKS unit, i.e. the EKS unit becomes inactive, the system adopts the current access level as determined by the key switch.
 - If changing the operating right amongst the operating stations, the EKS unit of the new active operating station is authoritative. If no EKS unit is assigned there, the effect is the same as for an inactive EKS unit.
 - The key information is evaluated by the HMI. Once an access level has been determined from the key information, HMI adopts this access level and also sets it in the NCK: The last access level set always applies to the system.
-

1.3.2 Boot up of the PCU: no HMI program installed

Requirement

No HMI program is installed during the PCU boot up.

Ramp-up phase

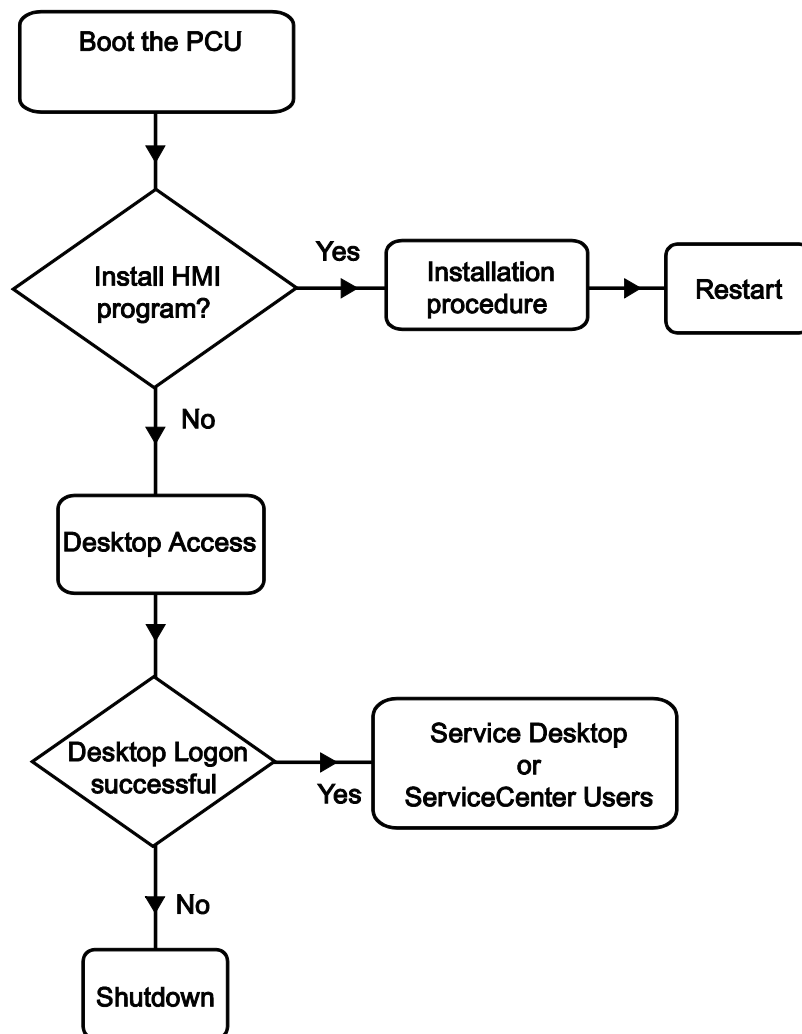


Figure 1-13 PCU bootup diagram (without HMI program)

Explanations:

- If installation programs are present in directory D:\Install, you will be prompted during the first boot up as to whether the installation procedure should be started. After installation is completed, you need to restart the system.

The installation can also be skipped and carried out later. To do this, use the HMI Explorer (Installation via service desktop (Page 78)).

- "Desktop", "ServiceCenter Users" or "Shutdown" can be selected in the dialog "Desktop Access".
 - When selecting "Desktop", the "Desktop Logon" dialog is opened.
 - When selecting "ServiceCenter Users", the "Installing Logon" dialog is opened.

Note

When booting the PCU for the **first** time, the user can only log on as 'auduser'.

1.3.3 Boot up of the PCU: HMI program already installed

Requirement

There is already an HMI program installed.

Ramp-up phase

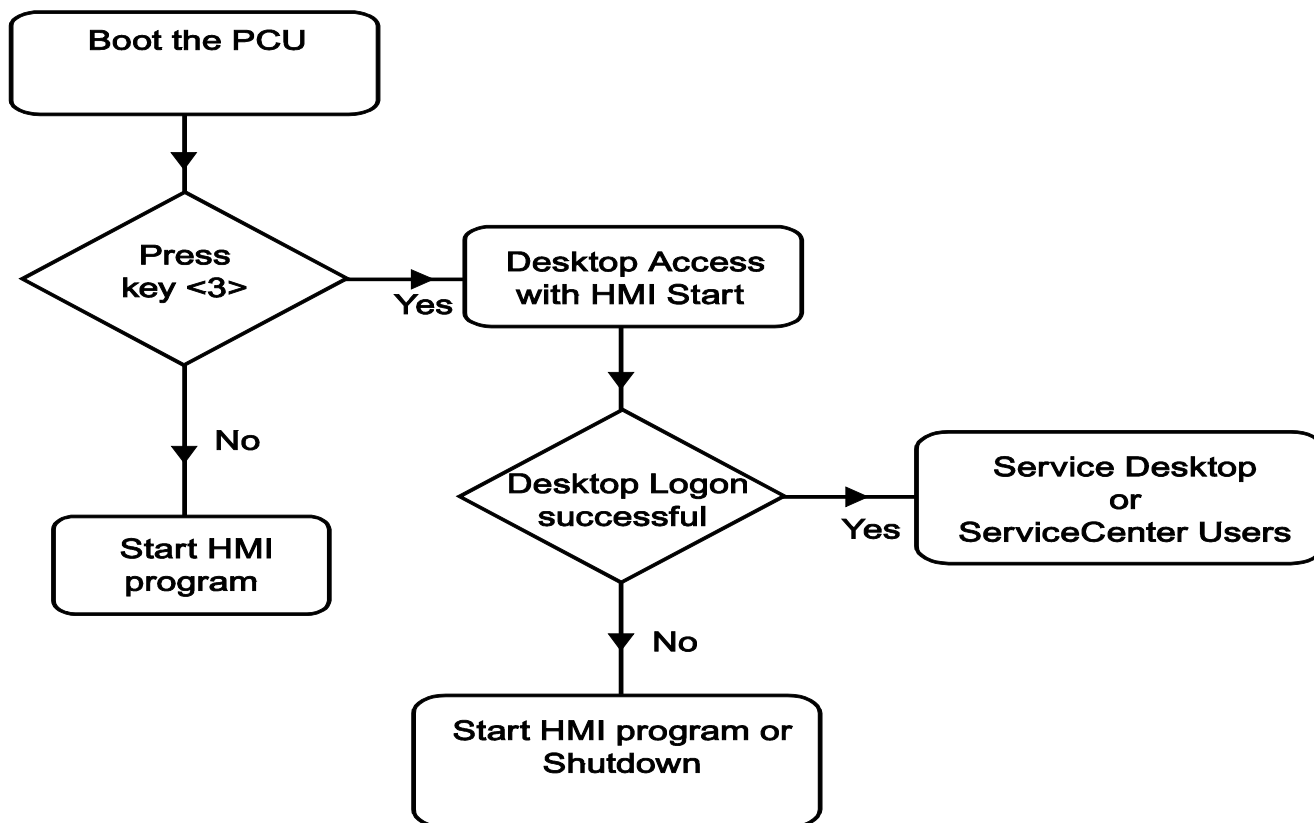


Figure 1-14 PCU bootup diagram (HMI program installed)

Explanations:

- If the HMI program is already installed, the PCU boots up and the HMI program is started (default).
- During the boot up, there is a time interval for pressing key <3> when the version information appears on the lower right of the background screen. Then the "Installing Logon" dialog opens.
- To carry out service tasks, you will have to log on as a service user.

The following input options are available:

- Manual logon to a domain using user name and password.
- Logon using EKS: A key and valid user data for authentication must be provided for this. If valid user data is not available via the EKS, the "Desktop Logon" dialog is displayed along with user name and password.
- Both options can be selected.

1.3.4 Setting the screen resolution

Overview

The system behavior during boot up for the screen resolution is set in the file tcu.ini. You will find the delivery status for tcu.ini in E:\siemens\system\etc.

Modified tcu.ini files are saved in F:\addon_base\..., F:\loem_base\..., F:\user_base\...

Reference: Operator Components Manual, "Networking" chapter

Set the resolution when booting up the PCU

The following options are available in the # RESOLUTION section in the tcu.ini:

0 = SYSTEM

1 = AUTO_OP_1 (default)

2 = AUTO_OP_2

3 = AUTO_MON_1

4 = AUTO_MON_2

5 = 640X480

6 = 800X600

7 = 1024X768

8 = 1280X1024

The meanings of the settings are as follows:

Settings	Meaning
SYSTEM	The resolution is not specially set; i.e., the resolution last used in the system is active, e.g., the resolution which had been set manually in the Control Panel.
AUTO_OP_1	Default: During boot up, the resolution is automatically set ("PCU panel" has priority) in accordance with the following scenarios:
Example 1:	There is a PCU panel (irrespective of whether there is a PCU monitor and TCU panels) [in active TCU mode: The following additional condition applies: PCU is activated (with dynamic resolution switching enabled)]: The resolution is set to the max. resolution of the PCU panel (max. 1280x1024).

Settings	Meaning
Example 2:	There is no PCU panel, however there is a PCU monitor (irrespective of whether there are any TCU panels): [in active TCU mode: The following additional condition applies: PCU is activated (with dynamic resolution switching enabled)]: The resolution is not specially set; i.e., the resolution last used in the system is active, e.g., the resolution which had been set manually in the Control Panel. (Different to AUTO_OP_2 !)
AUTO_OP_2	Like AUTO_OP_1, except: Example 2: There is no PCU panel, however there is a PCU monitor (irrespective of whether there are any TCU panels): [in active TCU mode: The following additional condition applies: PCU is activated (with dynamic resolution switching enabled)]: The resolution is set to the max. resolution of the PCU monitor, reduced to the next lowest SINUMERIK resolution. The SINUMERIK resolutions are 640x480, 800x600, 1024x768 and 1280x1024. Example: In the case of a PCU monitor with a max. resolution of 1440x900, the SINUMERIK resolution setting is 1280x1024.
AUTO_MON_1	During boot up, the resolution is automatically set ("PCU monitor" has priority) in accordance with the following scenarios:
Example 1:	There is a PCU monitor (irrespective of whether there is a PCU panel and TCU panels) [in active TCU mode: The following additional condition applies: PCU is activated (with dynamic resolution switching enabled)]: The resolution is set to the max. resolution of the PCU monitor, reduced to the next lowest SINUMERIK resolution. The SINUMERIK resolutions are 640x480, 800x600, 1024x768 and 1280x1024. Example: In the case of a PCU monitor with a max. resolution of 1440x900, the SINUMERIK resolution setting is 1280x1024. If there is a PCU panel, the display there is panned if the max. resolution of the PCU panel is lower than the max. resolution of the PCU monitor.
Example 2:	There is no PCU monitor, however there is a PCU panel (irrespective of whether there are any TCU panels): [in active TCU mode: The following additional condition applies: PCU is activated (with dynamic resolution switching enabled)]: The resolution is not specially set; i.e., the resolution last used in the system is active, e.g., the resolution which had been set manually in the Control Panel. (Different to AUTO_MON_2 !)

Settings	Meaning
Example 3:	There is no PCU monitor and no PCU panel (= headless operation): A) in active TCU mode: a) dynamic resolution switching is enabled (resolution adaptation entry in TCU.ini) and at least one TCU is already logged on: The resolution is set to the resolution of the TCU which is currently active. b) dynamic resolution switching is enabled (resolution adaptation entry in TCU.ini) and no TCU has logged on yet or dynamic resolution switching is disabled: The resolution is set to the max. resolution of the current PCU panel, i.e., which logged on during the previous session. Default: Default TCU resolution in accordance with the registry. Notice: The first TCU panel to logon (later) becomes activated. The focus handler then automatically sets the resolution to this TCU panel's resolution (in the case of dynamic resolution switching). A) in inactive TCU mode: The resolution is not specially set - i.e. the resolution used during the previous session in the system is active, e.g. the resolution set manually in Control Panel.
AUTO_MON_2	Like AUTO_MON_1, except: Example 2: There is no PCU monitor, however there is a PCU panel (irrespective of whether there are any TCU panels): [in active TCU mode: The following additional condition applies: PCU is activated (with dynamic resolution switching enabled)]: The resolution is set to the max. resolution of the PCU panel (max. 1280x1024).
640X480	During boot up, the SINUMERIK resolution is set to 640x480.
800X600	During boot up, the SINUMERIK resolution is set to 800x600.
1024X768	During boot up, the SINUMERIK resolution is set to 1024x768.
1280X1024	During boot up, the SINUMERIK resolution is set to 1280x1024.

1.4 BIOS settings for PCU 50.3

Overview

The BIOS of the PCU is preset in such a way that no changes are required. The date and time can be set under Windows or the user interface.

Note

Changes in the BIOS

Your device configuration is preset for working with the software supplied with the unit.

You should only change the preset values if you have modified your device in any way, or if a fault occurs when the unit is powered up.

Starting BIOS setup

1. Start the BIOS SETUP as follows:

Reset the device (warm or cold restart).

After the startup test, the following message appears:

Press < F2 > to enter SETUP or <ESC> to show boot menu

2. Press the F2 key as long as the BIOS prompt appears on the screen.

The BIOS main menu opens:

PhoenixBIOS Setup Utility					
Main	Advanced	Security	Boot	Version	Exit
System Time : [09:40:27] System Date : [08/15/2005]				Item Specific Help	
▶ IDE Channel 0 Master [None] ▶ IDE Channel 0 Slave [None] ▶ SATA Port 0 [40008MB] ▶ SATA Port 1 [None] ▶ SATA Port 2 [None] ▶ SATA Port 3 [None]				<Tab>, <Shift-Tab>, or <Enter> selects field. Example : Hour/Minute/Second Month/Day/Year	
Memory Cache : [Write Back]					
▶ Boot Options ▶ Keyboard Features ▶ Hardware Options					
System Memory: 640 KB Extended Memory: 514048 KB					
F1 Help ↑ ↓ Select Item + / - Change Values F9 Setup Defaults ESC Exit ← → Select Menu Enter Select Sub-Menu F10 Save and Exit					

Figure 1-15 BIOS Main Menu (Example)

BIOS setup: Default settings

The following system parameters are saved on delivery:

Menu: Main

System parameters	Default settings	Own entries
System Time	hh:mm:ss	
System Date	MM/DD/YYYY	
IDE Channel 0 Master	None	
IDE Channel 0 Slave	None	
SATA Port 0	40008 MB	
SATA Port 1	None	
SATA Port 2	None	
SATA Port 3	None	
Memory Cache	Write Back	

Boot options		
Quick boot mode	Enabled	
SETUP prompt	Enabled	
POST errors	All, but not keyboard	
Summary screen	Enabled	
Diagnostic screen	Enabled	
Post Code/Status	LPC Bus	

Keyboard features		
Numlock	ON	
Key click	Disabled	
Keyboard auto-repeat rate	30 / sec	
Keyboard auto-repeat delay	½ sec	

Hardware options		
PCI MPI/DP	Enabled	
Onboard Ethernet 1	Enabled	
On-board Ethernet 1 Address	08 00 06 90 xx xx	
On-board Ethernet 1 Remote Boot	Enabled	
Onboard Ethernet 2	Enabled	
On-board Ethernet 2 Address	08 00 06 90 xx xx	
On-board Ethernet 2 Remote Boot	Disabled	
SafeCard functions	Enabled	

Hardware options		
Fan control	Enabled	
CRT/LCD selection	Simultan. Auto	

Menu: Advanced

System parameters	Default settings	Own entries
Installed O/S	Other	
Reset configuration data	No	
Legacy USB support	Disabled	
USB controller restart	Enabled	
I/O Device Configuration		
Internal COM 1	Enabled	
Base I/O address	3F8	
Interrupt	IRQ 4	

PCI Configuration		
PCI device slot 1		
Option ROM scan	Enabled	
Enable master	Enabled	
Latency timer	Default	
PCI device slot 2		
Option ROM scan	Enabled	
Enable master	Enabled	
Latency timer	Default	

SATA/PATA Configuration		
PATA Controller:	Enabled	
SATA Controller mode	Enhanced	
AHCI Configuration	Disabled	
RAID support	Disabled	

Menu: Security

System parameters	Default settings	Own entries
Supervisor password is	Disabled	
User password is	Disabled	
Set user password	Enter	
Set supervisor password	Enter	

System parameters	Default settings	Own entries
Password on boot	Disabled	
Fixed disk boot sector	Standard	

Menu: Boot

System parameters	Default settings	Own entries
Boot priority order:		
1: SATA0:	Fujitsu MHT2040BHTBD	
2: PCI BEV:	VIA BootAgent	
3:		
4:		
5:		
6:		
7:		
8:		
Excluded from boot order:		

Menu: Version

System parameters	Default settings	
SIMATIC PC	SINUMERIK PCU 50.3	
BIOS version	V05.01.06	
BIOS number	A5E00370214-ES005	
MPI/DP firmware	V01	
CPU type	Celeron ® M processor 1.50 GHz	
CPU ID	06D8	
Code revision	0020	

Menu: Exit

Save Changes & Exit	All changes are saved; a system restart is carried out with the new parameters.

Changing BIOS settings

Once additional components have been installed or attached, it may be the case that the system has to be informed of this via the BIOS setup:

1. Start the device.
2. When the prompt to activate the BIOS setup appears, press the <F2> key (corresponds to horizontal softkey 2 on the OP).
3. The BIOS setup menu appears. In the menu, use the cursor keys to navigate to the desired selection box.
4. Change the setting using the <+> key (press <SHIFT> and <X> at the same time) or the ↔ key on the numeric keypad.
5. Using the left-right cursor keys, you can reach other setup menus.
6. Press <ESC> (<Alarm Cancel> key) to go to the "Exit" menu (or press the right cursor key again).
7. Press the <Enter> key to exit the setup menu.

Then the system starts.

Note

Changes to the BIOS settings, with the exception of the boot sequence, require an OEM contract to be concluded.

1.5 BIOS settings for PCU 50.5

Overview

The BIOS of the PCU is preset in such a way that no changes are required. The date and time can be set under Windows or the user interface.

Note

Changes in the BIOS

Your device configuration is preset for working with the software supplied with the unit.

You should only change the preset values if you have modified your device in any way, or if a fault occurs when the unit is powered up.

Starting BIOS setup

1. Start the BIOS SETUP as follows:
Reset the device (warm or cold restart).
After the startup test, the following message appears:
`PRESS < F2 > to enter SETUP or <ESC> to show Bootmenu`
2. Press the F2 key as long as the BIOS prompt appears on the screen.
The BIOS main menu opens:

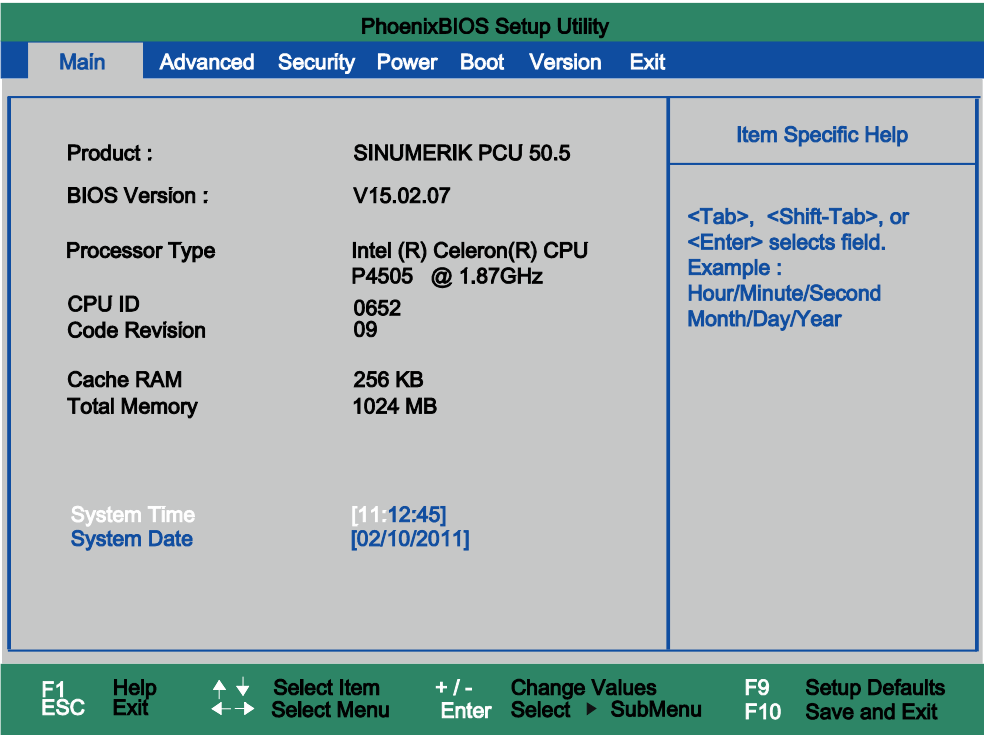


Figure 1-16 BIOS main menu PCU 50.5 (example)

BIOS setup: Default settings

The following system parameters are saved on delivery:

Menu: Main

System parameters	Default settings	Own entries
System Time	hh:mm:ss	
System Date	MM/TT/JJJJ	

Menu: Advanced

Peripheral Configuration	Default settings	Own entries
Internal COM 1	Enabled	
Onboard Ethernet 1	Enabled	
Onboard Ethernet 2	Enabled	
PCI - MPI / DP ¹⁾	Enabled	

¹⁾ only for CPB (-C)

SATA Configuration	Default settings	Own entries
SATA Controller	Enabled	
SATA Controller mode	AHCI	

Video Configuration	Default settings	Own entries
Primary video device	PEG	

USB Configuration	Default settings	Own entries
USB Port0	Enabled	
USB Port1	Enabled	
USB Port2	Enabled	
USB Port3	Enabled	
USB Port4	Enabled	
USB Port5	Enabled	
USB Port6	Enabled	
USB Port7	Enabled	
USB Port8	Enabled	
USB Port9	Enabled	
USB Port10	Enabled	

Chipset Configuration	Default settings	Own entries
Port 80h Cycles	LPC Bus	
VT-d	Enabled	
HPET	Enabled	

CPU Configuration	Default settings	Own entries
P-States (IST)	Enabled	
Turbo Mode	Enabled	
CMP Support	Auto	
HT Support	Auto	
Use XD Capability	Disabled	
VT Support	Enabled	
Turbo Mode	Disabled	
C-States	Enabled	

Active Management Technology Support	Default settings	Own entries
Intel AMT Support	Disabled	
Intel AMT Setup prompt	Disabled	
AMT CIRA Request Trig	Disabled	
AMT DIRA Timerout	30	
Un-Configure ME	Disabled	
USB Configure	Enabled	

Advanced	Default settings	Own entries
Fan control	Enabled	
Operating Mode Switch	Enabled	

Menu: Security

System parameters	Default settings	Own entries
Supervisor Password	Not installed	
User Password	Not installed	
Set Supervisor Password	Inactive (no password assigned)	
Set User Password	Inactive (no password assigned)	

Menu: Power

System parameters	Default settings	Own entries
Wake on LAN 1	Disabled	
Wake on PME/LAN2	Enabled	
Wake on Time	Disabled	
After Power Failure	Power On	

Menu: Boot

System parameters	Default settings	Own entries
Quick Boot	Disabled	
Quiet Boot	Disabled	
POST Errors	All without keyboard	
NumLock	on	
USB Boot	Enabled	
Bootmanager	Enabled	
PXE Boot to LAN 1	Enabled	
PXE Boot to LAN 2	Disabled	

Legacy	Default settings	Own entries
Normal Boot Menu	Advanced Placeholder	

Menu: Version

System parameters	Default settings
Product	SINUMERIK PCU 50.5
BIOS Version	V15.02.07
BIOS Number	A5E02619467-ES001
InsydeH20 Version	03.59.53.1053
Intel ME Version	6.0.3.1203
Video Option ROM	1976
RAID Option ROM	9.5.0.1037
PXE Option ROM	1.3.30
Mode switch 0	Normal

Menu: Exit

System parameters	Default settings	Own entries
Exit Saving Changes	All changes are saved; a system restart is carried out with the new parameters.	
Profile	Standard	

Changing BIOS settings

Once additional components have been installed or attached, it may be the case that the system has to be informed of this via the BIOS setup:

1. Start the device.
2. When the prompt to activate the BIOS setup appears, press the <F2> key (corresponds to horizontal softkey 2 on the OP).
3. The BIOS setup menu appears. In the menu, use the cursor keys to navigate to the desired selection box.
4. Change the setting using the <+> key (press <SHIFT> and <X> at the same time) or the ↔ key on the numeric keypad.
5. Using the left-right cursor keys, you can reach other setup menus.
6. Press <ESC> (<Alarm Cancel> key) to go to the "Exit" menu (or press the right cursor key again).
7. Press the <Enter> key to exit the setup menu.

Then the system starts.

Note

Changes to the BIOS settings, with the exception of the boot sequence, require an OEM contract to be concluded.

Boot sequence

BIOS version 05.01.12

In order to ensure correct functioning of the Software Ghost with Backup/Restore, please set in BIOS under "SATA/PATA Configuration" → "SATA Controller Mode" → "Compatible".

The CompactFlash Card is entered as SATA. The hard disk (or Solid State Drive) is DISK0, and the CompactFlash Card is DISK2.

Configuring the system

2.1 System settings

2.1.1 Configuring the communication via MPI

If you are using a PCU in conjunction with a SINUMERIK 840D *powerline* and require MPI communication, you must install the S7DOS 8.4 communication driver. In all other cases, you do not have to install this driver.

The setup for this communication driver can be found in the following directory:

- D:\updates\

2.1.2 How to change the name of the PCU

Default

The PCU is supplied with an automatically generated computer name.

Procedure

To change the name of the PCU:

1. Select "Start" → "Control Panel" → "System".
2. Select the "Computer Name" tab and click on "Change".

The following dialog opens:

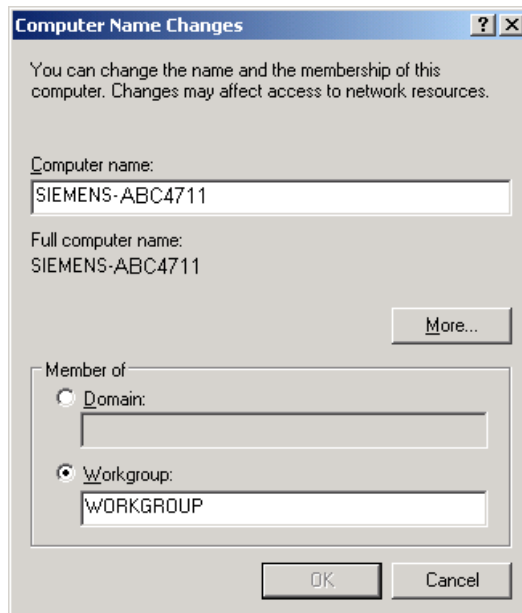


Figure 2-1 Changing the name of the PCU

2.1.3 This is how you set the IP address of the PCU

Default setting

Note

IP address in the system network

The IP address 192.168.214.241 is set as a factory default for every PCU on the system network.

You must only perform the steps described below if you wish to change this default.

Procedure

1. Select the following on the PCU on the Service Desktop: "Start" → "Settings" → "Network Connections".

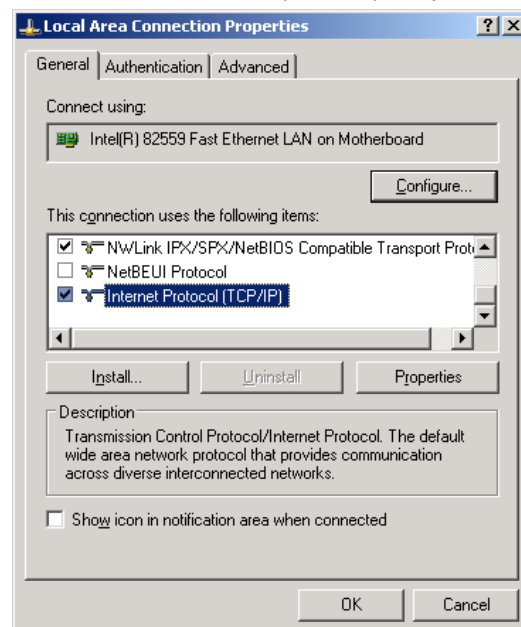
The "Network Connections" window opens.

2. Double-click the Ethernet 2 interface you want to parameterize which is to be used for connecting the TCU or system network.

The "Ethernet 2 (System Network) Properties" window opens.

3. Under the "General" tab, select "Internet Protocol (TCP/IP)" and click the "Properties" button.

The "Internet Protocol (TCP/IP) Properties" window opens:



4. Under the "General" tab, select the "Use the following IP address" option and enter the IP address and the subnet screen form.

Recommended setting for the first PCU:

The screenshot shows the 'Internet Protocol (TCP/IP) Properties' dialog box with the 'General' tab selected. The text inside reads: 'You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.' There are two radio button options: 'Obtain an IP address automatically' (unselected) and 'Use the following IP address:' (selected). Below the selected option are three input fields: 'IP address:' containing '192 . 168 . 214 . 241', 'Subnet mask:' containing '255 . 255 . 255 . 0', and 'Default gateway:' which is empty. Below these are another two radio button options: 'Obtain DNS server address automatically' (unselected) and 'Use the following DNS server addresses:' (selected). Below the selected option are two input fields: 'Preferred DNS server:' and 'Alternate DNS server:', both empty. At the bottom right of the dialog is an 'Advanced...' button. At the very bottom are 'OK' and 'Cancel' buttons.

5. Enter the required new IP address and confirm the settings with "OK".

2.1.4 How to add the PCU to a domain

Requirement

Only a user with the corresponding entitlement, e.g. a domain administrator, can add a PCU to an existing domain.

Domain Controller (DC)

A domain controller (DC) is a server for central authentication and authorization of computers and users in a network. In a network with a domain controller, several computers are combined to form one domain.

Proceed as follows

To add the PCU to a domain:

1. Select: "Start" → "Settings" → "Control Panel" → "System", "Computer Name" tab.

Default: The PCU belongs to a "WORKGROUP" and is not assigned to any domain.

2. Click "Change ...".

The following dialog opens:

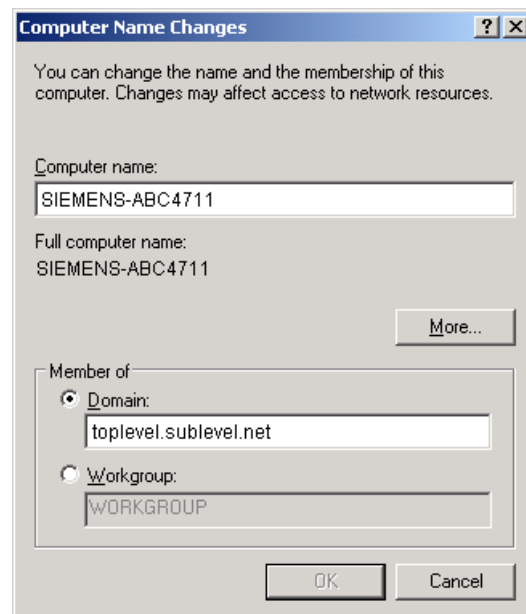


Figure 2-2 Add PCU to domain

3. Enter the name of the domain to which you want to add the PCU.
4. You will then be asked to log on as a user with the corresponding entitlement to conclude the process.

2.1.5 How to connect an external monitor

Preconditions

To connect an external monitor, the following preconditions apply:

- The external monitor is connected to the DVI interface on the PCU (using an adapter, if necessary).
- The monitor may not be connected while in use.

Procedure

To connect an external monitor:

1. Right-click the PCU's service desktop and select "Properties" from the pop-up menu.
2. Select the "Settings" tab and then click "Advanced".
3. Select the "Troubleshooting" tab and set the "Hardware acceleration" to a value other than zero. The recommended setting is "full".
4. Close the dialog and click OK to confirm all the dialogs.
5. Repeat steps 2 and 3. There is an additional tab for "Intel(R) ... Graphics Controller"
6. Click the "Graphics Properties" button.
 - In the case of a PCU 50.3: On the "Devices" tab the external monitor corresponds to the "Monitor" selection; the OP/TP on the PCU corresponds to the "Notebook" selection. Select a "Primary Device" and a "Secondary Device".
 - In the case of a PCU 50.5: On the "Display" tab, the OP/TP on the PCU corresponds to the selection "Primary Display"; the external monitor corresponds to the selection "Secondary Display". Click the button "Multiple Displays" to select the "Clone" or "Extended" mode.
7. Close the dialog and click OK to confirm all the dialogs: The external monitor is now ready.

2.1.6 This is how you switch off the USB interfaces.

Command `sc_usb disable`

To prevent harmful software entering via the USB ports to the control or into the system network, you can switch off the USB interfaces.

This command is called up in the DOS shell:

Default setting:

The USB interfaces are not locked.

Syntax:

`sc_usb disable [-minutes] [all] [HOSTS...]`

- Without a time indication [-minutes], a permanent disable is set. With time indication, the disable takes the time set in minutes.
- The disable may refer to all network nodes of the system network, or a list of host names or IP addresses may be specified to which the disable should apply.

Command `sc_usb enable`

With "sc_usb enable" USB storage units on permanently disabled USB interfaces or on certain host names or IP addresses are enabled again.

2.2 Configuring a customized operator interface

2.2.1 How to select the language for the Windows system

Default setting

In the delivery condition, the Windows XP operating system is only installed on the PCU in English and with a US keyboard layout.

Requirement

In order to be able to switch languages, the desired languages must be installed from the DVD of the "SINUMERIK Service Pack Recovery Media Win XP ProEmbSys SP2". With the "Multilingual User Interface" (MUI), you can switch to menus, dialog boxes and keyboard layouts for the Windows system in different languages:



WARNING

"Regional Options" tab

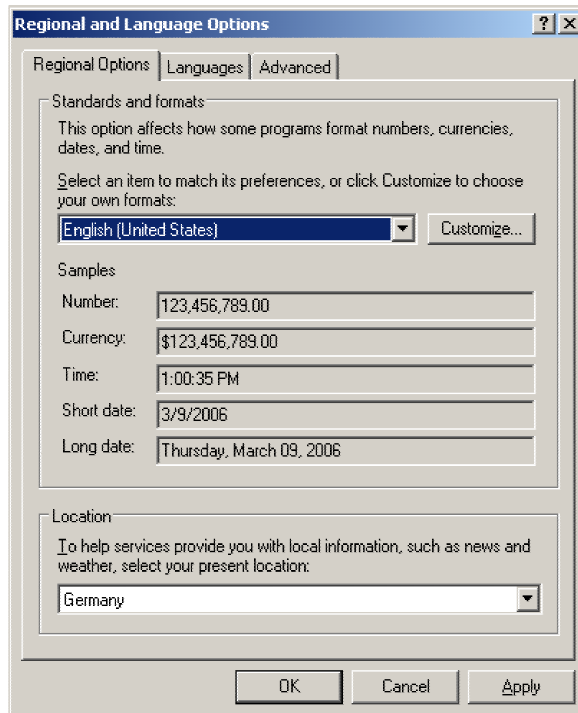
The settings for the keyboard assignment and the formats for date, time and number displays on the "Regional Options" tab **must not be changed under "Standards and formats"**.

These settings are automatically adapted depending on the language selected for the operating software.

Select a language

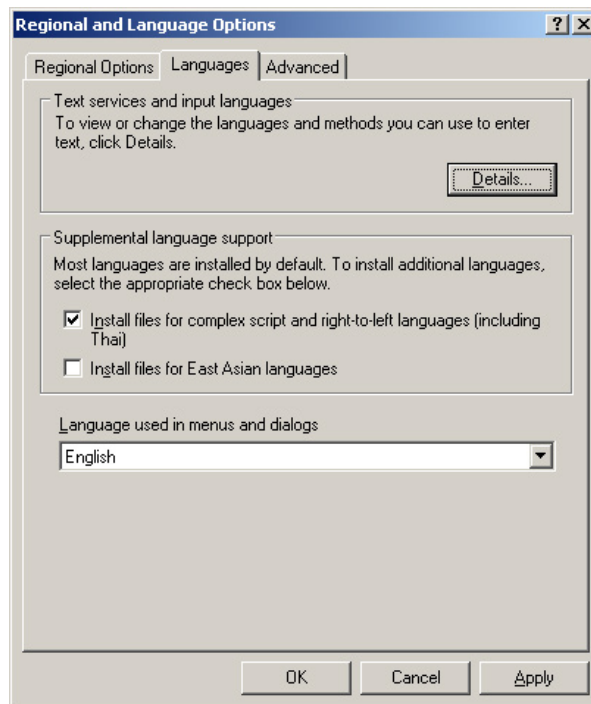
After installing a language from the corresponding CD, proceed as follows:

1. Select "Start" → "Control Panel" → "Language and Regional Options", to open the following dialog box:



This setting must not be changed!

2. Select the "Languages" tab, in order to switch the language for the Windows XP user interface. Under "Language used in menus and dialogs," select the new language and confirm with OK.



3. On the "Advanced" tab, select the language for programs that do not support Unicode.

Result

To make the language change effective, the PCU must be rebooted. The selectable languages are displayed using the font set of the respective language.

See also

How to install additional languages under Windows XP (DVD) (Page 84)

2.2.2 Displaying the boot screen

Default setting

The Siemens boot screen is archived in the PCU under the following path:

```
F:\hmis1\siemens\sinumerik\hmi\ico\ico640\splash.png
```

Display OEM-specific boot screen

You can choose any name.

The directory is set in ServiceCenter Users under:

- Startup: Bitmaps folder
- Startup: Default bitmap folder

2.2.3 Changing the background of the service desktop

You can configure user-specific background pictures for the Service Desktop in the ServiceCenter Users.

If you want to set a background pattern instead, make this setting in the registry.

Procedure

Background picture

1. Save the required pictures in the appropriate resolutions in the corresponding subfolders of `F:\Background picture`
2. Switch to the appropriate "User Types" in the ServiceCenter Users and specify the `F:\Background picture` folder at "Background" in the "Desktop" tab.

Note

An attempt to set the Service Desktop background (as in standard Windows) via the Control Panel does not affect the Service Desktop background display, but only the startup screen display.

Background pattern

A background pattern for the Service Desktop is not set via the "Control Panel" (system control), but in the registry:

Key:	HKLM\SOFTWARE\Siemens\SINUMERIK\Basesoftware\ <version>\HMIDesktop
Value:	Pattern (STRING)
Date:	<bit code>" (e.g. "0 80 114 32 0 5 39 2", see HKCU\Control Panel\Patterns) "(None)" (= NO background pattern)
InitDatum:	Background pattern previously set via the Control Panel
Default data:	"(None)" (if entry is not available/readable)

2.3 Customized settings during boot up

Overview

Regarding the settings and functionality, booting followed by the start of an HMI program starting differs from booting with subsequent start of the Service Desktop.

2.3.1 Configuring key filters for an HMI program

Filtering keys

During boot up of an HMI program, keystroke sequences and pressed function keys are simultaneously filtered. The keystroke sequences and functions that were pressed simultaneously and are to be filtered are configured in file E:\Windows\System.ini.

Filtering keystroke sequences:

Section:	MMC103Keyb
Key:	SeqAct
Value:	<bit mask> (= keystroke sequences to be filtered, specified in accordance with the comment in E:\Windows\System.ini)
Init value:	262143

Filtering function keys that were pressed simultaneously:

Section:	MMC103Keyb
Key:	ConcurrentKeyMask
Value:	<bit mask> (= function keys to be filtered, specified in accordance with the comment in E:\Windows\System.ini)
Init value:	255

In the ServiceCenter Users, the keys that are to be filtered can be configured for each specific user:

- while the system boots → "Startup (2)" tab
- while the HMI program runs: → "HMI" tab
- on the Service Desktop or Windows Desktop: → "Desktop (2)" tab

system.ini file

```
# -----  
# SIEMENS KEYBOARD-DRIVER SECTION (START)  
# -----  
# State of driver after startup  
# (0=NORMAL, 1=PERMANENT-SHIFT ACTIVE)  
keybStartState=0  
  
# Type of Keyboard to be used  
# (0=MMC103, 1=MFII)  
keybType=0  
  
# Filtering Key Sequences  
# =====  
# Activation of Key-Sequences to be ignored  
# SeqAct holds a BIT-Pattern for a max of 20 sequences  
# which could be ignored by the keyboard-driver  
# if bit-n is set in Seq-Act, the according sequence  
# will be ignored  
# The following 20 sequences are implemented  
# 0 CTRL-ALT-DEL  
# 1 ALT-F4  
# 2 ALT-TAB  
# 3 LEFTSHIFT-ALT-TAB  
# 4 RIGHTSHIFT-ALT-TAB  
# 5 CTRL-ESC  
# 6 ALT-ESC  
# 7 ALT-SPACE  
# 8 (SHIFT)-CTRL-F1  
# 9 (SHIFT)-CTRL-F2  
# 10 (SHIFT)-CTRL-F3  
# 11 (SHIFT)-CTRL-F4  
# 12 (SHIFT)-CTRL-F5  
# 13 (SHIFT)-CTRL-F6  
# 14 (SHIFT)-CTRL-F7  
# 15 (SHIFT)-CTRL-F8  
# 16 M$_1  
# 17 M$_2  
# 18 CAPSLOCK  
# 19 NUMLOCK  
# 20 (reserved)  
KeySequencesEnable =1
```



```
SeqAct=262143

# Filtering Concurrent Keys
# =====
# Definition and selection of function keys (buttons) which shall not be
# pushed in parallel or at the same time (function keys F1-F12 and
# additional hardkeys).
# If more of these buttons are pushed at the same time, the system removes the
# code of these buttons except the code of the button which was pushed first.
# This function of the keyboard-driver is activated by ConcurrentKeyEnable.
# ConcurrentKeyMask holds a BIT-Pattern for a max of 17 function keys, which
# must not be pushed simultaneously (concurrently). If bit-n is set in
# ConcurrentKeyMask, the corresponding button is supervised.
# The following 17 function keys are implemented:
# F1
# F2
# F3
# F4
# F5
# F6
# F7
# F8
# F9
# F10
# F11
# F12
# PAGE-UP (NUMLOCK) Alarm Hardkey on Operator Panel
# PAGE-DOWN (NUMLOCK) Tool Management Hardkey on Operator Panel
# HOME (NUMLOCK) Program Manager Hardkey on Operator Panel
# END (NUMLOCK) Program Hardkey on Operator Panel
# CURSOR-DOWN (NUMLOCK) Alarm Hardkey on Operator Panel
```

Note

Scope of the keys

Only keys, which have already been defined in the system.ini file, can be enabled.

No additional keys can be defined for filtering.

See also

ServiceCenter Users: User settings (Page 13)

2.3.2 Saving service desktop settings

Starting the service desktop

The Windows platform is freely accessible via the service desktop. When the service desktop is started, all of the programs that would automatically be started by Windows (standard version) during log-in are also started.

- **Executing an HMI program**

The HMI program can also be started from the service desktop.

- **Ending an HMI program**

When an HMI program started from the service desktop is exited, you are returned to the service desktop.

Saving the service desktop (default)

The settings on the service desktop (e.g. arrangement of the links on the service desktop) are not saved when you log off. A service technician should always find the same starting condition on the service desktop, not the settings from a previous session.

Saving the settings of the service Desktop

This behavior can be changed by making an entry in the registry. The following settings can be saved via this registry entry:

- Positions of open windows
- Size and position of the task bar
- Moving and deleting links

Key:	HKLM\SOFTWARE\Siemens\SINUMERIK\Basic software\ <version>\HMI Manager
Value:	SaveSINDesktopSettings (DWORD)
Date:	1 (the settings are saved) or 0 (the settings are NOT saved)
Init data:	- Value is not created by the basic software -
Default data:	0 (if entry is not available/readable)

The key is effective for all service users and other users.

Links on the service desktop are always saved, irrespective of the registry entry.

Note

Application windows that are still open before logging out, must be closed by the setting "Save settings" before exiting the service desktop. Otherwise, these application windows will briefly be displayed and then closed again during a restart immediately before the HMI program starts.

2.3.3 Starting programs during boot up

Starting additional programs

Programs can also be started at the same time as the HMI program and are started automatically by Windows when the service desktop is opened. This start in parallel to the HMI program can be configured.

If the programs to be started are located in the Windows directories of E:\Documents and Settings, the following registry entry must be set:

Key:	HKLM\SOFTWARE\Siemens\SINUMERIK\Basic software\ <version>\HMI Manager
Value:	StartSINHMIStartupDirsPrograms (DWORD)
Date:	1 (the programs are started) or 0 (the programs are NOT started)
Init data:	0
Default data:	0 (if entry is not available/readable)

If the programs to be started are set in the registry entries 'HKCU\Software\Microsoft\Windows\CurrentVersion\Run' and 'HKLM\Software\Microsoft\Windows\CurrentVersion\Run', the following registry entry must be set:

Key:	HKLM\SOFTWARE\Siemens\SINUMERIK\Basic software\ <version>\HMI Manager
Value:	StartSINHMIRunPrograms (DWORD)
Date:	1 (the programs are started) or 0 (the programs are NOT started)
Init data:	0
Default data:	0 (if entry is not available/readable)

Executing an HMI program

While the HMI program is being executed, Windows Explorer runs in the background and its settings (disabled browser functionality, taskbar, start menu, ...) prevent the Windows platform being accessed unintentionally.

The behavior of the taskbar can be user-specifically set in the ServiceCenter Users under "HMI program" using "Taskbar Autohide" and "Taskbar On Top".

Ending an HMI program

When the HMI program is ended, Windows XP shuts down completely.

2.3.4 Starting OEM programs

Overview

You can start OEM programs directly before starting the HMI system software. This requires these programs (or their links) to be stored in subdirectories of the directory C:\RunOEM.

Starting sequence

The subdirectories are executed in the order listed. The programs within a subdirectory are started in the chronological order in which they were placed in the subdirectory.

- Programs in the C:\RunOEM\SeqOnce subdirectory are started **once and sequentially**, i.e., a program is not started until the previously started program is completed.
- Programs in the C:\RunOEM\Seq subdirectory are started **sequentially whenever the system is ramped up**, i.e., a program is not started until the previously started program is completed.
- Programs in the C:\RunOEM\ParOnce subdirectory are started **once and simultaneously**. They run parallel with the HMI system software.
- Programs in the C:\RunOEM\Par subdirectory are started **simultaneously whenever the system is ramped up**. They run parallel with the HMI system software.

Not only program files, but also other types of file can be stored in the subdirectories, which are then opened in accordance with their file type.

For example, ".txt" files are opened using Notepad, ".htm" files are opened using Internet Explorer.

2.3.5 Starting applications in service mode

Starting other applications

If other applications are to be started in service mode, enter them with their complete path in the [OEMRun] section in the file WINBOM.INI:

Example: Starting the "Notepad" program

```
[OEMRunOnce]

"Start WinVnc", "x:\I386\system32\StartWinVnc.exe"

"Check Password", "x:\I386\system32\CheckPEPw.exe"

[OEMRun]

"Start Backup/Restore", "x:\I386\system32\GhostOrder.exe"

"notepad", "e:\windows\notepad.exe"
```

All other entries must not be changed.

2.4 Configuring the SITOP UPS module for use with the PCU Base Software

Overview

SITOP UPS modules can continue operation of the PCU 50.5 temporarily if a power failure occurs and/or shut down the system properly.

For example, a SITOP UPS module also protects against data loss when the PCU 50.5 hardware is shut down, because for technical reasons with Windows-based systems, data is still written to the SSD after shutdown.

- You will find general information about SITOP UPS modules used in combination with the PCU base software in Section SITOP modules for the PCU 50.5 (Page 62).
- If you want to use a SITOP UPS module in combination with the PCU base software, you must first adapt the settings of the SITOP software (Page 64).
- Optionally, you can adapt the delay time after which the HMI software will be forced to shut down (Page 67).
- You must configure the hardware (Page 69) of the SITOP UPS module before using it with the PCU base software.

2.4.1 SITOP modules for the PCU 50.5

Suitable SITOP UPS modules

SITOP UPS modules can continue operation of the PCU 50.5 temporarily if a power failure occurs and/or shut down the system properly.

In conjunction with the PCU base software, all specific applications are taken into account (e.g. HMI software) with the `Shutdown.bat` component.

Safe shutdown after a power failure of the delivery condition of the PCU 50.5 takes approximately 20 seconds. For a typical power consumption of approx. 60 W, a capacitor-buffered UPS with an energy storage device of 2.5 kW is recommended for a PCU 50.5. If longer buffer times are required, then several expansion modules can be connected in a cascade connection.

Example of a compatible SITOP UPS module:

SITOP 500S (15 A / 2.5 kW capacitor)

Article no.: 6EP1933-2EC41

Further information

Information about available SITOP UPS modules is given in the catalog or in the Internet:

- Catalog KT 10.1 Power supply SITOP
- SIEMENS Industry Mall > 24 V DC Uninterruptible Power Supplies
(<https://eb.automation.siemens.com/mall/en/US/Catalog/Products/7010117>)

Note

Information about the test environment for machine OEMs

The functionality has been tested in the standard configuration with the PCU base software.

When installing add-on or OEM software components, you must check the shutdown procedure of the entire system and adapt the factory setting (180 seconds).

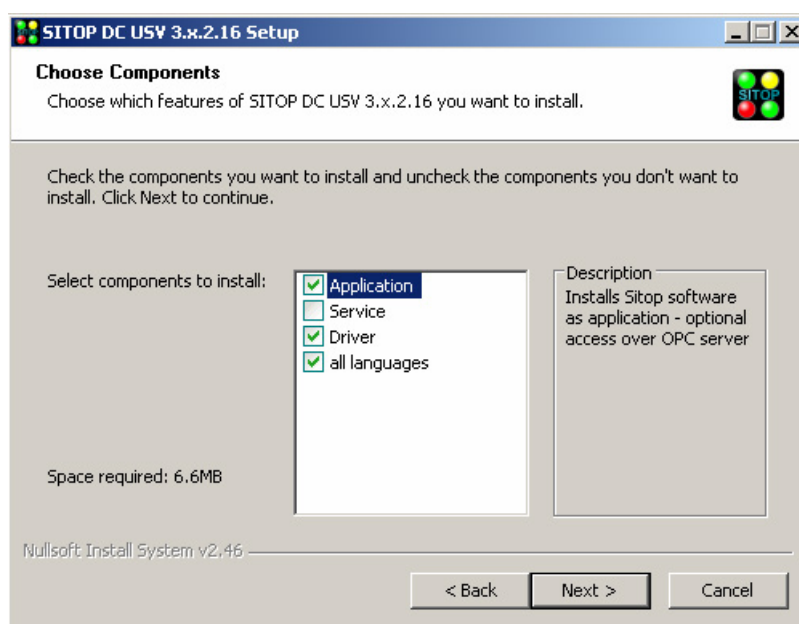
See: Parameterizing a delay time for quitting the HMI software (Page 67)

2.4.2 Configuring SITOP software for PCU Base Software

If you want to use a SITOP UPS module in combination with the PCU base software, you must first adapt the settings of the SITOP software.

Requirement

- The SITOP UPS hardware is connected via the USB interface.
- The SITOP software of version 3.x.2.16 or later is installed in directory `C:\Program Files (x86)\SITOP\`
- If you want to use SITOP software of version 3.2.1.16 or older, this must be installed as a normal application, not as a service.



Note

Windows service of the SITOP software version 3.2.1.16 or older is not compatible with the PCU base software

If you use SITOP software of version 3.2.1.16 or older as a Windows service with the PCU base software, the correct shutdown procedure cannot be performed!

If you have installed the SITOP software of version 3.2.1.16 as a Windows service, you must uninstall this and install it again as an application in order to be able to use it in conjunction with the PCU base software.

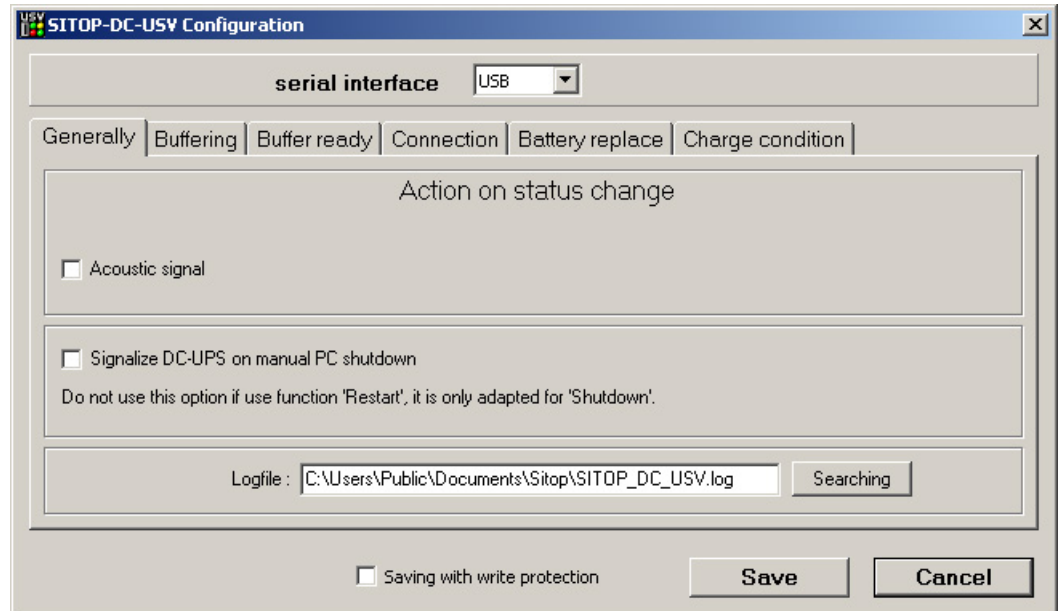
As of version 3.2.1.17, the SITOP software can be used as service with the PCU base software and is recommended.

- The SITOP software was configured during the installation so that it starts automatically when Windows is run-up (factory setting).
- The Service Desktop is active.

Procedure

Proceed as follows to configure the SITOP software for use with the PCU base software:

1. Call the settings of the SITOP software, for example, by right-clicking the SITOP UPS icon in the Windows information area and then selecting "Configuration". The "SITOP DC UPS Configuration" dialog box opens.



2. In the "General" tab, select "UPS" in the drop-down list at "Serial interface".

3. Switch to the "Floating operation" tab and make the following settings:

- Deactivate the "Display monitoring window after" checkbox

Note

Deactivating the monitoring window

Display of the monitoring window can cause malfunctioning of the HMI software.

- Click directly in the text field at "Start application after" and specify the path of the Shutdown.bat:

C:\Program Files

(x86)\Siemens\MotionControl\siemens\simumerik\hmi\base\Shutdown.bat

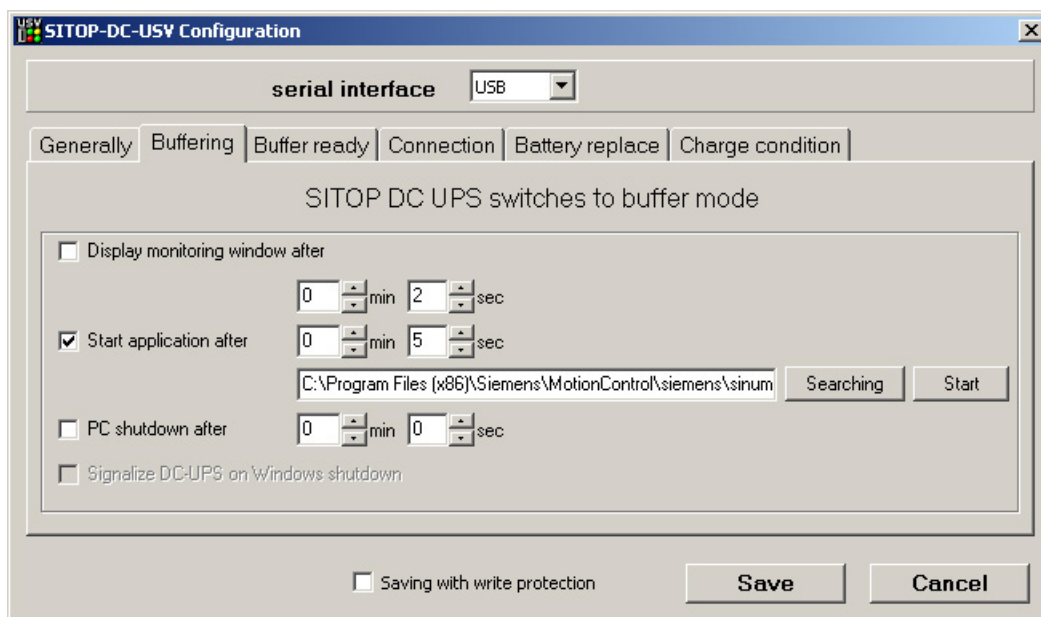
Note

Do not use the "Browse" button

The dialog can crash if you use the "Browse" button to specify the path of the Shutdown.bat.

Instead, enter the path directly in the text field.

- After you have specified the path of the Shutdown.bat, activate the "Start application after" checkbox. Optionally, you can specify the wait time after which the Shutdown.bat is to be executed (e.g. 5 seconds).



4. Confirm the settings with "Save."

2.4.3 Parameterizing a delay time for quitting the HMI software

Before the SITOP monitor closes Windows 7, `Shutdown.bat` shuts down operation of the HMI software.

If the HMI software cannot be shut down within 180 seconds (factory setting) due to an error, HMI software shutdown is forced and Windows 7 is shut down.

If the HMI software of your OEM installation is not shut down within 180 seconds, you can parameterize this wait time manually.

Note**Information about the test environment for machine OEMs**

The "SITOP UPS" function has been tested in the standard configuration with the PCU base software. When installing add-on or OEM software components, you must check the shutdown procedure of the entire system and adapt the factory setting.

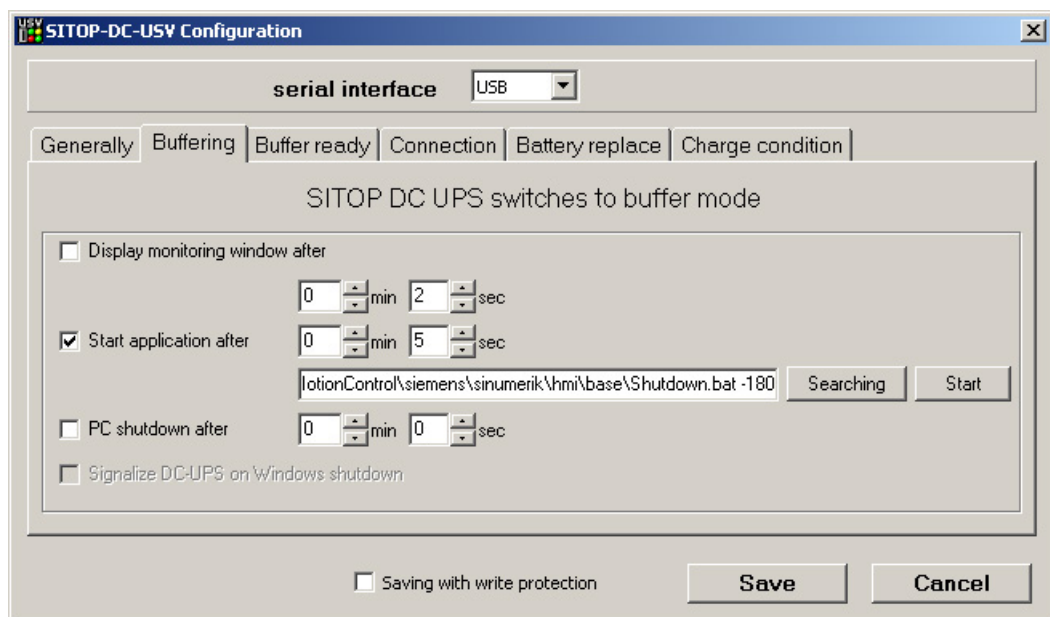
Requirement

- The SITOP UPS module is configured for use with the base software.
See Configuring SITOP software for PCU Base Software (Page 64).
- The Service Desktop is active.

Procedure

Proceed as follows to change the delay time:

1. Call the settings of the SITOP software, for example, by right-clicking the SITOP UPS icon in the information area and then selecting "Configuration".
2. In the "SITOP DC UPS Configuration" dialog box, enter the wait time in seconds as a command line parameter in the "Floating operation" tab:
 - **Syntax:** <Path>\Shutdown.bat -<Time in seconds>
 - **Example:** C:\Program Files
(x86)\Siemens\MotionControl\siemens\simumerik\hmi\base\Shutdown.bat -180



3. Confirm the setting with "Save".

2.4.4 Hardware configuration of the SITOP UPS module

Buffering parameterization

The UPS module can be used to select whether buffering should be completed after a predetermined period of time or not until the accumulator's lowest discharge threshold (= maximum buffer time) has been reached. Both buffering parameterizations result from this.

"Maximum buffer time" mode

This mode enables the system to be shut down in a time-optimized manner. The UPS module is synchronized with the shutdown of the operating system. Buffering is maintained until the operating system has been shut down. The operating system must shut down within a maximum of five minutes (including all applications). Otherwise, the UPS module buffers for the maximum buffer time (dependent on the accumulator state).

Required settings on the UPS module (USB interface)

	On - Off			
1			+2V	Cut-in threshold +22V fixed
2			+1V	
3			+0.5V	
4			+1V	End-of-charge voltage + 26.3V fixed
5			+1V	
6			+0.5V	
7			+0.2V	
8			+0.2V	
9			+0.1V	
10			0.35A / 0.7A	Charging current

	On - Off			
1		●		Set time/max. time
2		●	+320 s	Buffer time +5 s fixed
3		●	+160s	
4		●	+80s	
5		●	+40s	
6		●	+20s	
7		●	+10s	
8	●			Disconnection
9		●		Battery operating state on/off

Legend:



Delivery condition setting



Setting for operation on the PCU

“Fixed buffer time” mode

In this mode, the UPS module always buffers for the pre-selected, fixed period of time. It is not possible to synchronize the UPS module with the operating system shutdown.

Required settings on the UPS module

	On - Off			
1			+2V	Cut-in threshold +22V fixed
2			+1V	
3			+0.5V	
4			+1V	° ° End-of-charge voltage + 26.3V fixed
5			+1V	
6			+0.5V	
7			+0.2V	
8			+0.2V	
9			+0.1V	
10			0.35A / 0.7A	Charging current

	On - Off			
1	●			Set time/max. time
2		●	+320 s	° ° Buffer time +5 s fixed
3	●		+160s	
4		●	+80s	
5		●	+40s	
6		●	+20s	
7		●	+10s	
8	●			Disconnection
9		●		Battery operating state on/off

Legend:



Delivery condition setting



Setting for operation on the PCU

2.5 Configuration of the Service Center

2.5.1 Overview

You can permanently adapt the settings for the backup and restoration of disk images (Service Center (Page 87) component) in the "Service Center Backup/Restore" dialog box:

- Configuration of the network adapter (Page 74)
- Configuration of the host (Page 76)

Adaptations that you make here are stored in the `servicesystem.ini` configuration file.

You can also adapt the settings directly in the associated configuration files:

- `servicesystem.ini`
- Additional settings for the Service System on the USB flash drive: `ghost.ini`

You can also view the log files of the Service Center or restart the PCU 50.5 in the Service Center in the "Service Center Backup/Restore" dialog box.

To open the dialog box, click the ServiceCenter Backup-Restore icon on the Desktop.

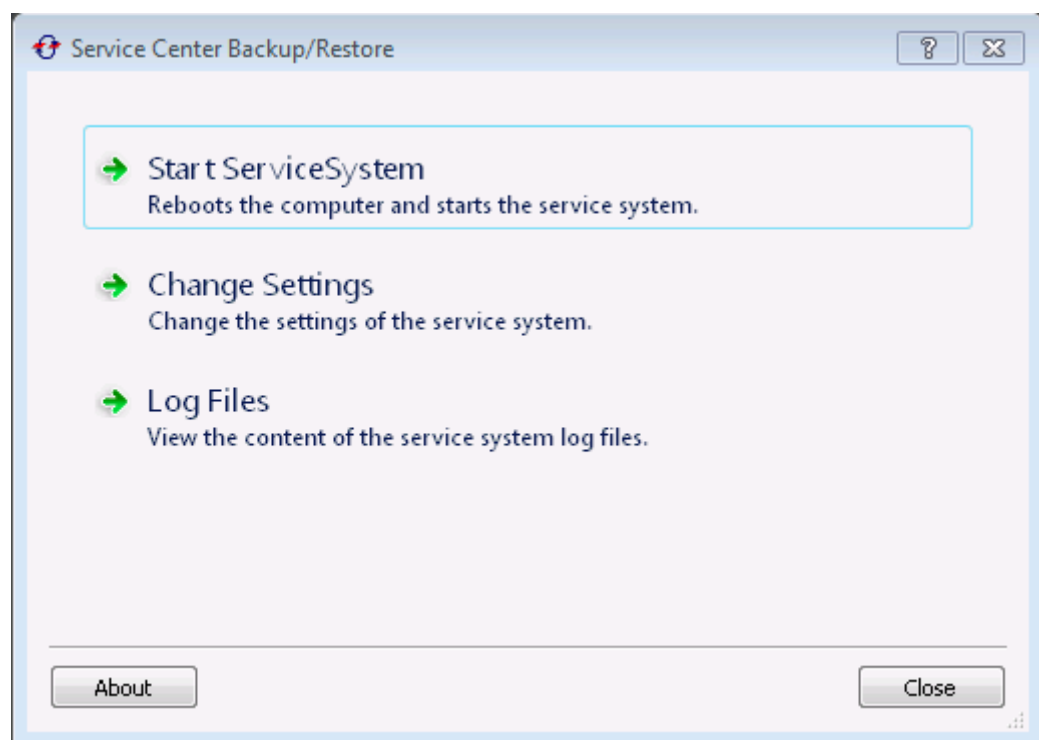


Figure 2-3 ServiceCenter Backup/Restore

2.5.2 Configuration of the network adapter

You can make the network settings in the "Adapter Settings" tab and permanently save this configuration, in contrast to the settings in the "Network Settings" dialog box of the Service Center (Page 90).

Settings that you make in this dialog box are stored in the `servicesystem.ini` configuration file.

Overview

You can make the following IP settings in the "Adapter Settings" dialog box:

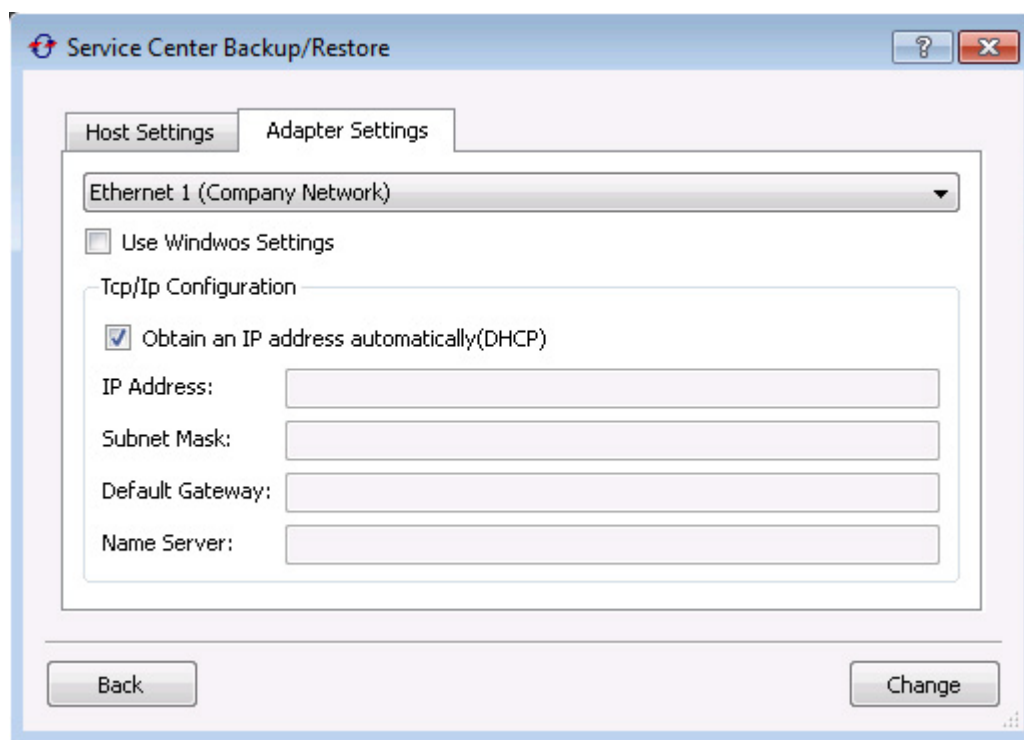


Figure 2-4 Adapter settings - Ethernet 1 (company network)

Table 2- 1 Settings in the "Adapter Settings" dialog box

Settings	Purpose
Adapter	Select the Ethernet interface that you want to configure. All settings in this dialog box refer to the selected Ethernet interface.
Use Windows Settings	Select whether the values set in the Control Panel are to be used.
Obtain an IP address automatically (DHCP)	Select whether the addresses are to be fetched dynamically from the DHCP server. If the checkbox is deactivated, you must make the settings manually.
IP Address	Set the IP address of the NCU. You can specify an IP address from the following range: 192.168.214.250 - 254
Subnet Mask	Specify a subnet mask, e.g. 255.255.255.0

Settings	Purpose
Default Gateway	Set the IP address of the standard gateway. A standard gateway creates a standard route in the IP routing table for all destinations that are not in the subnet.
Name Server	Set the IP address of the name server. A name server answers questions asked about a domain name zone using a DNS database.

2.5.3 Configuration of the host

You can make the network settings in the "Host Settings" tab and permanently save this configuration, in contrast to the settings in the "Network Settings" dialog box of the Service Center.

Settings that you make in this dialog box are stored in the `servicesystem.ini` configuration file.

Overview

You can make the following settings in the "Network Settings" dialog box:

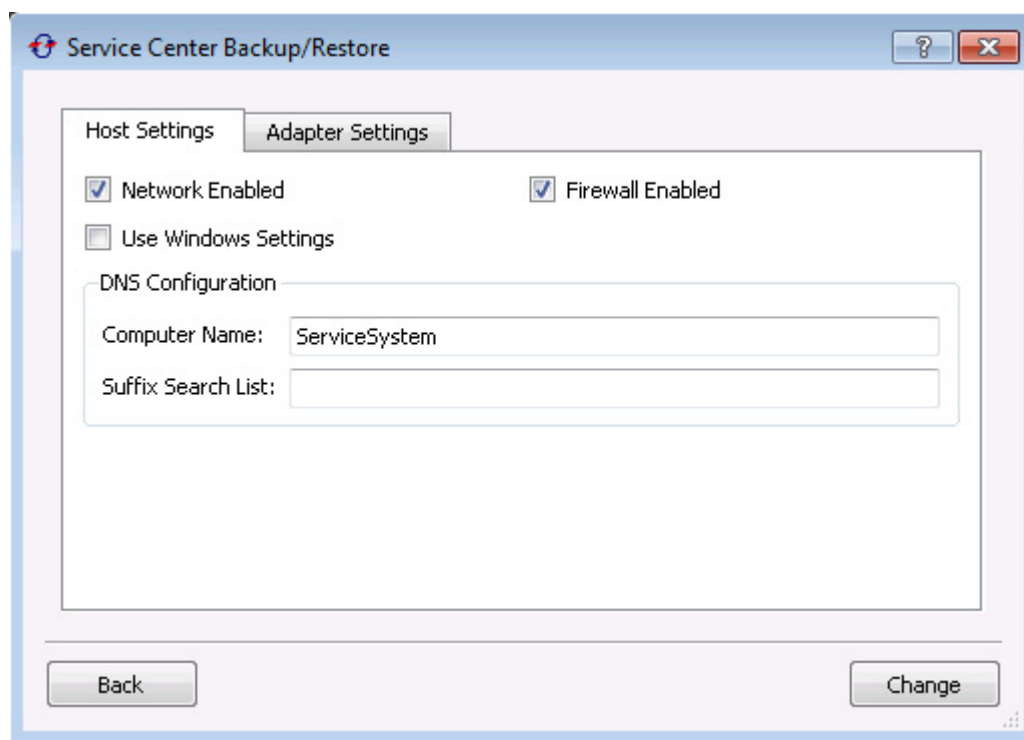


Figure 2-5 Host settings

Table 2- 2 IP settings in the "Host Settings" dialog box

Settings	Purpose
Network enabled	Activate or deactivate the network interfaces of the PCU 50.5.
Firewall enabled	Activate or deactivate the firewall of the PCU 50.5.
Use Windows Settings	Select whether the values set in the Control Panel are to be used.
DNS Configuration > Computer Name	Specify the DNS computer name.
DNS Configuration > Suffix Search List	Parameterize a search list for DNS suffix, e.g. "network.com". The factory setting of the Ethernet interface "Local Area Connection" is ".local".

Install software and updates

3.1 Installing SINUMERIK products

Overview

This chapter describes how to install additional software on the basis of the pre-installed PCE basic software or how to carry out an update.

The description below is based on the delivery condition of the hardware and software components.

The service desktop is, for example, used for the following tasks:

- Installing HMI system software
- Setting the running environment of the HMI system software
- Checking the hard disk or version
- Privilege for SIMATIC STEP 7

3.2 Installation via service desktop

Installing additional software

The service desktop makes it possible to install system software or a software update. This mostly affects installation/update packages that are to be installed via the Windows network.

Installation can be performed in two ways:

- The installation/update package is stored in directory D:\INSTALL. When booting the PCU the next time, the installation/update process is automatically started while booting. Only when the installation or update process is completed does normal boot manager continue and, if necessary, HMI software started.
- The installation/update process can be started from the service desktop directly by executing the installation/update package.

Using installation directories

Several installation directories can be set to enable an operator setup to be executed automatically. Installation directories include the subdirectory D:\Install and the directories listed in the [SetupDirs] section of the E:\Windows\System32\HMIServe.ini parameters file.

The "D:\Install" installation directory is preset there.

The key names contained within a section of the E:\Windows\System32\HMIServe.ini parameters file must be unique.

The installation directories are evaluated in the sequence described in the parameters file. If the parameters file is missing or it does not contain a [SetupDirs] section, the preset "D:\Install" installation directory is considered instead.

If the [SetupDirs] section contains installation directories, but not "D:\Install", the "D:\Install" directory is not considered within the context of the set installation directories.

When executing a setup using OpFile.txt, the issue of whether or not the setup requires a reboot once it has been completed is taken into account and displayed via a corresponding OpFile.txt entry. If a corresponding entry exists, a reboot is triggered. If there is a chain of setups to be executed one after the other, the reboot is performed once the final setup is complete.

Installing with HMI Explorer

The "HMI Explorer" Windows program is available on the service desktop. When this program is called up, detailed version information relating to the HMI system software applications installed and to Windows XP is displayed.

Applications can be individually started or uninstalled via the HMI Explorer. Select the menu "Install" to display and install the available software packages.

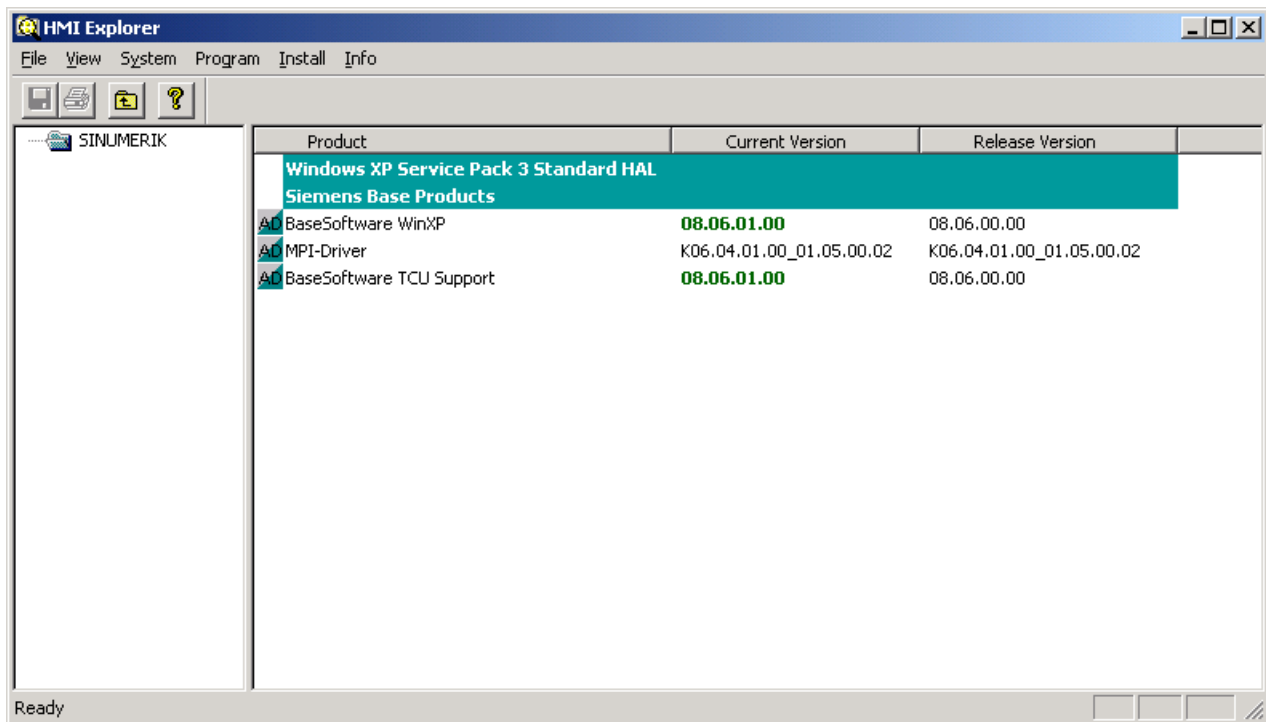


Figure 3-1 HMI Explorer (Example)

Description of HMI Explorer

The properties dialogue box gives detailed information on installing the software product:

- **Information on the SINUMERIK product:**

The "Info" dialogue box provides information on the selected SINUMERIK product:

Current version:	Specifies which version of the SINUMERIK product is currently installed. The version is shown in long form.
Internal version:	Shows the current internal version number of this product.
Installation Date/Time:	Shows the installation date and time of the current version.
Installation path:	Displays the path for the main directory of the SINUMERIK product.
Start application:	Gives information on the path to the *.exe file, which launches the SINUMERIK product.

- **Language of the product**

The "Language" dialogue box lists the installed languages for the respective SINUMERIK product and provides information on the name of the installed language. If the language is not known, an abbreviation of its name is displayed. Known languages of HMI Explorer are German, English, Spanish, French and Italian. The version of the installed language is also displayed. Information is also given regarding the installation time and date.

- **History of the product**

The "History" dialogue box shows information on the history of the SINUMERIK product. This dialogue box gives information on the release version, any service packs and hot fixes. The "release" entry is always available. The entries for "service pack" and "hot fix" only appear if they have been installed. Information on the "version", "internal version" and "installation date/time" is given for each entry on this list.

- **Component information**

The "Components" dialogue box shows the information on the components accompanying a product:

Component:	Component name
Version:	Internal version of the components
Path:	Path of the components
File:	*.exe file
Enable :	Shows whether or not the component is enabled
Description:	Description of components
Type:	Type of components

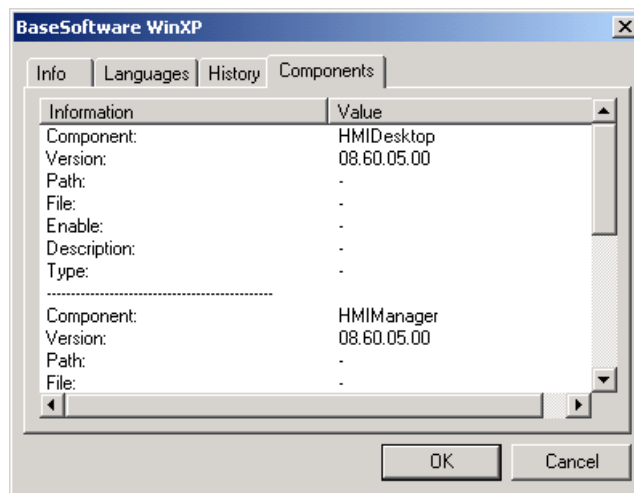


Figure 3-2 "Components" dialogue box (example)

3.3 How to install and authorize SIMATIC STEP 7

Overview

SIMATIC STEP 7 V5.5 SP1 can also be installed on the PCU.

Delivery item:	SIMATIC STEP 7 V5.5 SP1
Components:	SIMATIC STEP 7 V5.5 SP1 and AddOn for SINUMERIK 840D sl
Type of delivery:	1 installation DVD

Note

The "MPI driver" package available on the PCU is part of the HMI software and must not be uninstalled!

A network connection or a connection to a DVD drive is required.

Recommendation: Mouse port

Installation under Windows XP

You must follow the installation sequence described here!

1. Boot up the PCU in the service desktop.
2. Access DVD (via network or external DVD drive) and call up SETUP.EXE in the root directory there.
3. The installation proceeds with operator prompting. The installation directory for STEP 7 should be changed to F:\... where the directory in F: can be freely selected.
4. The prompt for "Transfer license keys" must be answered with "No, transfer the license keys later." The licensing is done after the installation of SINUMERIK add-on. Once installation is complete, the PC will need to be rebooted. During boot up, select the service desktop again.

5. Switch to the Sinumerik_Add_on directory on the DVD and call SETUP.EXE. The installation proceeds with operator prompting. Once installation is complete, the PC will need to be rebooted. During boot up, select the service desktop again.
6. Start the link "STEP7 authorizing" on the service desktop. This authorizes STEP 7 and it can be started from the operating software (STEP 7 appears as its own operating area on the expansion bar of the area menu, protected with access level 3).

The following entries are made automatically in F:\Add_on\oemframe.ini:

```
[s7tgtopx]

; with HMI Advanced: eliminate minimize/maximize buttons
; of the Step7 window

WindowState_Off=196608

; with HMI Advanced: switch to previous task when Step7 is terminated

nSwitchToTaskAfterTermination= -2
```

These entries may also need to be modified in OEM configurations.

3.4 How to install additional languages under Windows XP (DVD)

Use

Use the SINUMERIK service pack recovery media WIN XP ProEmbSys SP3 to:

- Subsequently install Windows components
- Re-establish the delivery condition of the PCU **without application software**
- To install other languages for Windows XP

Contents of the DVD

There are the following directories on the DVD:

Directory	Contents
1_WIN_Components	Windows XP ProEmbSys SP3 Windows XP ProEmbSys operating system, incl. SP3 for post-installation of software components that are no longer located on the PCU.
2_XP_Base	Symantec Ghost image for PCU 50.3 and EBOOT • Ghost image of the delivery condition of the PCU basic software Windows XP for PCU 50.3 without application software. • Ghost image for creating an "Emergency Boot System" (identical to the directory D:\EBOOT on the PCU)
3_MUI_1	Chinese (simplified) Traditional Chinese Japanese Korean Romanian Slovakian
4_MUI_2	Danish German French Dutch Italian Spanish Swedish

Directory	Contents
5_MUI_3	Brazilian Portuguese Finnish Polish Russian Czech Turkish Hungarian
EULA TERMS	Contained
Certificate of authenticity	Not contained

Installing languages

To install additional languages, proceed as follows:

1. If no DVD drive is directly connected via a USB port, the DVD can be accessed via a network to a released DVD disk drive via "Explorer" → "Tools" → "Map Network Drive". The letter G should be selected as the drive letter.
2. Select the directory with the appropriate language, the program "MUISETUP.EXE" starts. After accepting the licensing conditions, you can start the installation procedure with "Continue". A list of all of the pre-installed languages and the languages available on the DVD is displayed.
3. Now the desired languages can be installed/uninstalled by inserting or deleting a check mark in front of the language.
4. Other settings include:
 - Choice of language version for the standard user/new user
 - The language for programs without Unicode support must be set to "English (US)".
 - The font set must also be set to "English (US)".
5. After confirming with "OK," the installation begins. Many languages (e.g. Chinese) require system files that are also located on the DVD.

If the installation drive is other than the recommended "G:" the "Windows XP Professional Service Pack 2 CD" or the "Windows XP Professional CD" may be required. The path name must then be changed accordingly.

6. After successful installation, a reboot may be requested depending on the language.

Note

- The choice can only be made from among the languages that were previously installed on the PCU.
 - The new language of Windows XP only goes into effect if the user logs in again after the changeover or the PCU is turned off and then on again.
 - The language setting of the operating software is independent of this. It is set independently of this under "Start-up" → "HMI" → "Change Language".
-

Backing up and restoring data

4.1 Overview

You can perform installation and commissioning work on the PCU in the Service Center, which is based on Microsoft Windows PE.

When the Desktop is active, configuration of the Service Center is possible via the "ServiceCenter Backup-Restore" (Page 73) icon.

Most of the functions of the Service Center are available in the main menu (Page 88).

In particular, you can create or restore disk images in the Service Center. The following options are available:

- Create a disk image of the SSD (Page 93)
- Restore a disk image of the SSD (Page 95)
- Create a disk image of a partition (Page 97)
- Restore a disk image of a partition (Page 99)

The Service Center uses the Symantec Ghost (Page 104) software, which you can also call separately.

4.2 Starting the Service Center

To start the Service Center, proceed in one of the following ways:

- To call up the Service Center on the SSD while the PCU is starting, select the "Bootting Service System" during startup in the Windows boot menu.
- To call up the emergency boot system on a USB flash drive while the PCU is starting, set the service switch to E or press F12 during startup to show the boot selection.
- To switch to the Service Center while the Service Desktop (Windows 7) is active, click the "ServiceCenter Backup-Restore" icon on the Desktop and then click "Start ServiceSystem".

4.3 Functions of the Service Center

Overview

In the upper section of the Service Center main menu, you can select one of five tasks. Buttons for additional functions are available in the lower section.

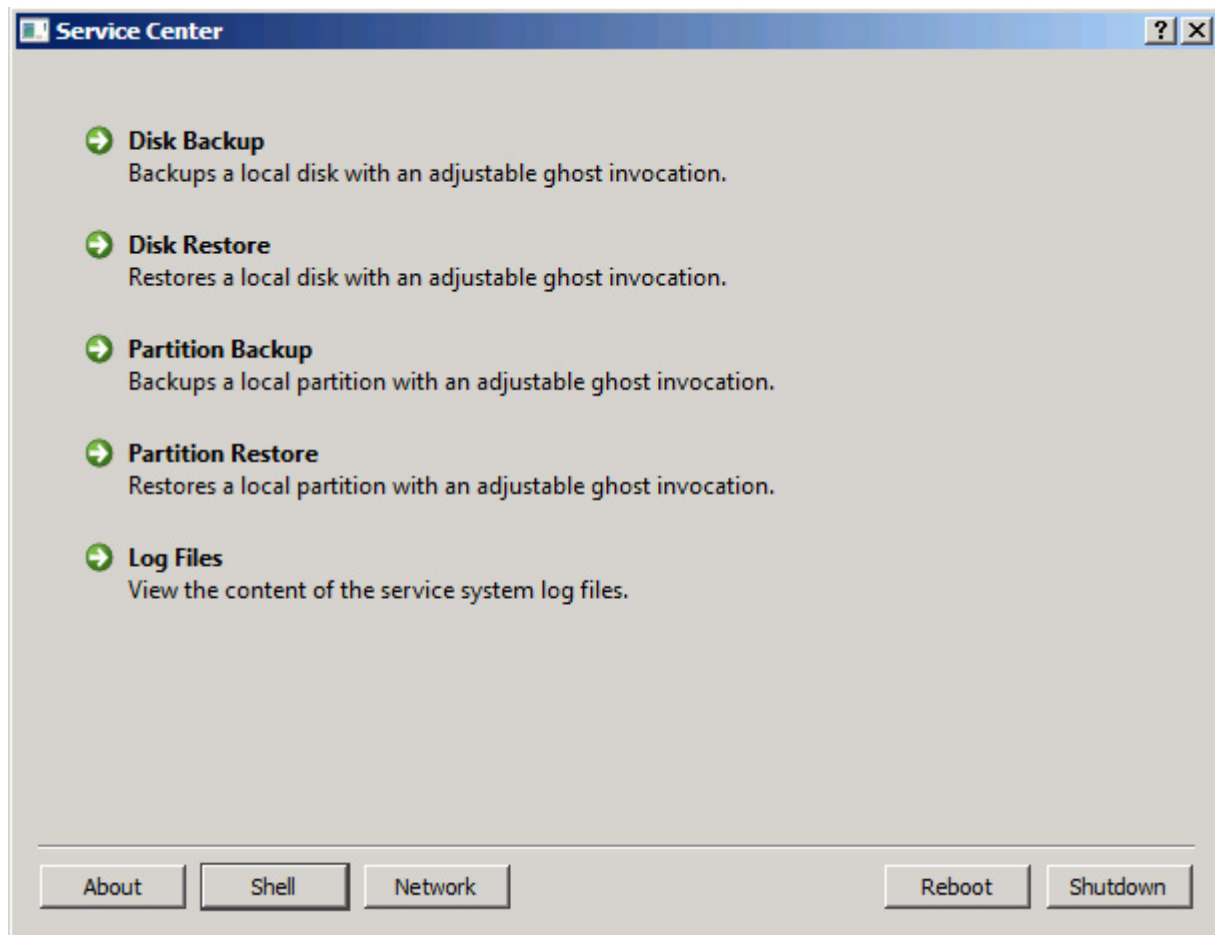


Figure 4-1 Service Center - Main menu

Table 4- 1 Functions in the main menu of the Service Center

Element of the window	Purpose
Disk Backup	Create a disk image of the local SSD. With this command, you can create a disk image of the entire SSD (with all partitions).
Disk Restore	Restore the local SSD with an existing disk image. With this command, you can restore the complete SSD (with all partitions) with a previously created disk image.
Partition Backup	Create a disk image of a partition on the local SSD. With this command, you can create a disk image of an individual partition of the SSD.

Element of the window	Purpose
Partition Restore	Restore a partition on the local SSD with a disk image. With this command, you can restore an individual partition of the SSD with a previously created disk image.
Log Files	View and save log files.
About	Viewing information about the Service Center.
Shell	Open the prompt. With this command, you can enter commands for executing tasks on the PCU without using a graphical user interface.
Network	Making network settings (Page 90). Any network settings you make in this dialog box are temporary. To change network settings permanently, switch to the Service Desktop.
Reboot	Restart the PCU.
Shutdown	Shut down the PCU.

Note

Data backup and restore with CompactFlash cards

If there is a CompactFlash card in the slot of the PCU, you will not be able to back up or restore data via the Service Center.

Instead, start the "ghost32.exe" program via the prompt of the Service Center.

4.4 Network settings in the Service Center

You can make network settings in the main menu of the Service Center by clicking "Network":

- IP settings at the enterprise network interface X1 (Local Area Connection 2)
- IP settings at the plant network interface X2 (Local Area Connection)
- Configuration of the host
- Configuration of IP routing

Note

Temporarily storing network settings in the Service Center

Any network settings you make in this dialog box are immediately effective, but temporary.

To change network settings permanently, switch to the Service Desktop.

For further information, see Section Configuration of the Service Center (Page 73).

Overview

IP settings

You can make the following IP settings in the "Network Settings" dialog box:

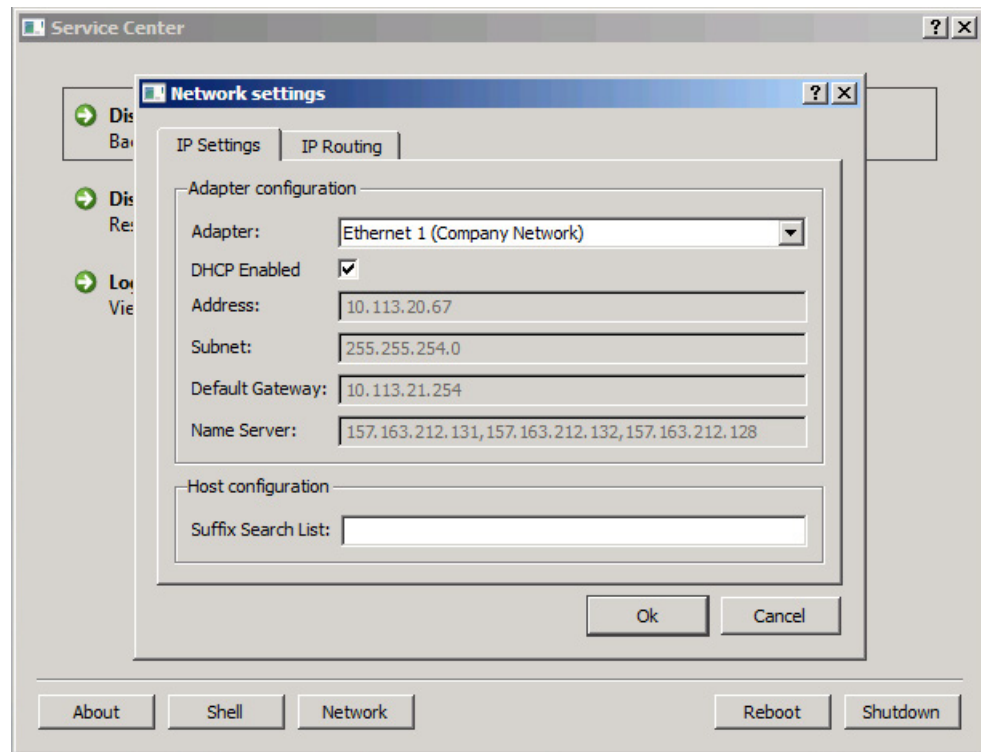


Figure 4-2 Service Center - IP settings

Table 4- 2 IP settings in the "Network Settings" dialog box

Section	Settings	Purpose
Adapter configuration	Adapter	Select the Ethernet interface that you want to configure. All settings in Section "Adapter configuration" refer to the selected Ethernet interface.
	DHCP Enabled	Select whether the addresses are to be fetched dynamically from the DHCP server. If the checkbox is deactivated, you must make the settings manually.
	Address	Set the IP address of the NCU. You can specify an IP address from the following range: 192.168.214.250 - 254
	Subnet	Specify a subnet mask, e.g. 255.255.255.0
	Default Gateway	Set the IP address of the standard gateway. A standard gateway creates a standard route in the IP routing table for all destinations that are not in the subnet.
	Name Server	Set the IP address of the name server. A name server answers questions asked about a domain name zone using a DNS database.
Host configuration	Suffix Search List	Parameterize a search list for DNS suffix, e.g. "network.com". The factory setting of the Ethernet interface "Local Area Connection" is ".local".

Routing settings

You can make the following routing settings in the "Network Settings" dialog box:

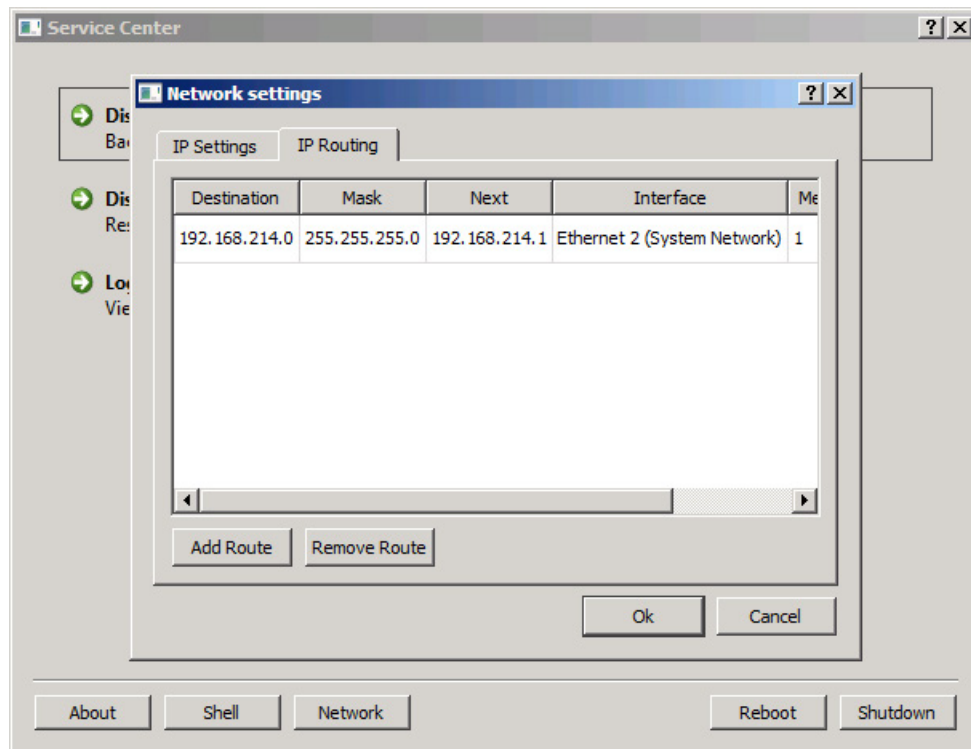


Figure 4-3 Service Center - IP routing

Table 4- 3 Routing settings in the "Network Settings" dialog box

Settings	Purpose
Add Route	Add a new route.
Remove Route	Delete a selected route.
Destination	Specify the network destination of the route as an IP address.
Mask	Subnet mask of the specified IP addresses.
Next	Next hop or forwarding IP address via which the network destination can be accessed.
Interface	Select an interface to which the configured IP routing applies: - Enterprise network interface X1 (Local Area Connection 2) - Plant network interface X2 (Local Area Connection)
Metric	Integer cost metric (1 to 9999). Required if several routes for sending a packet to a network destination are possible in the routing table.

See also

Release directory of the PC/PC in the network (Page 114)

Creating a shortcut to the network drive (Page 119)

Configuring routing in the network (Page 109)

4.5 Create a disk image of the SSD

You can create a disk image of the entire SSD in the Service Center with the "Disk Backup" function.

Procedure

To save the SSD as a disk image, proceed as follows:

1. Start the Service Center, e.g. via the Windows boot menu.
See also: Starting the Service Center (Page 87)
2. In the main menu of the Service Center, click "Disk Backup".

The "Service Center - Disk Backup" dialog box opens.

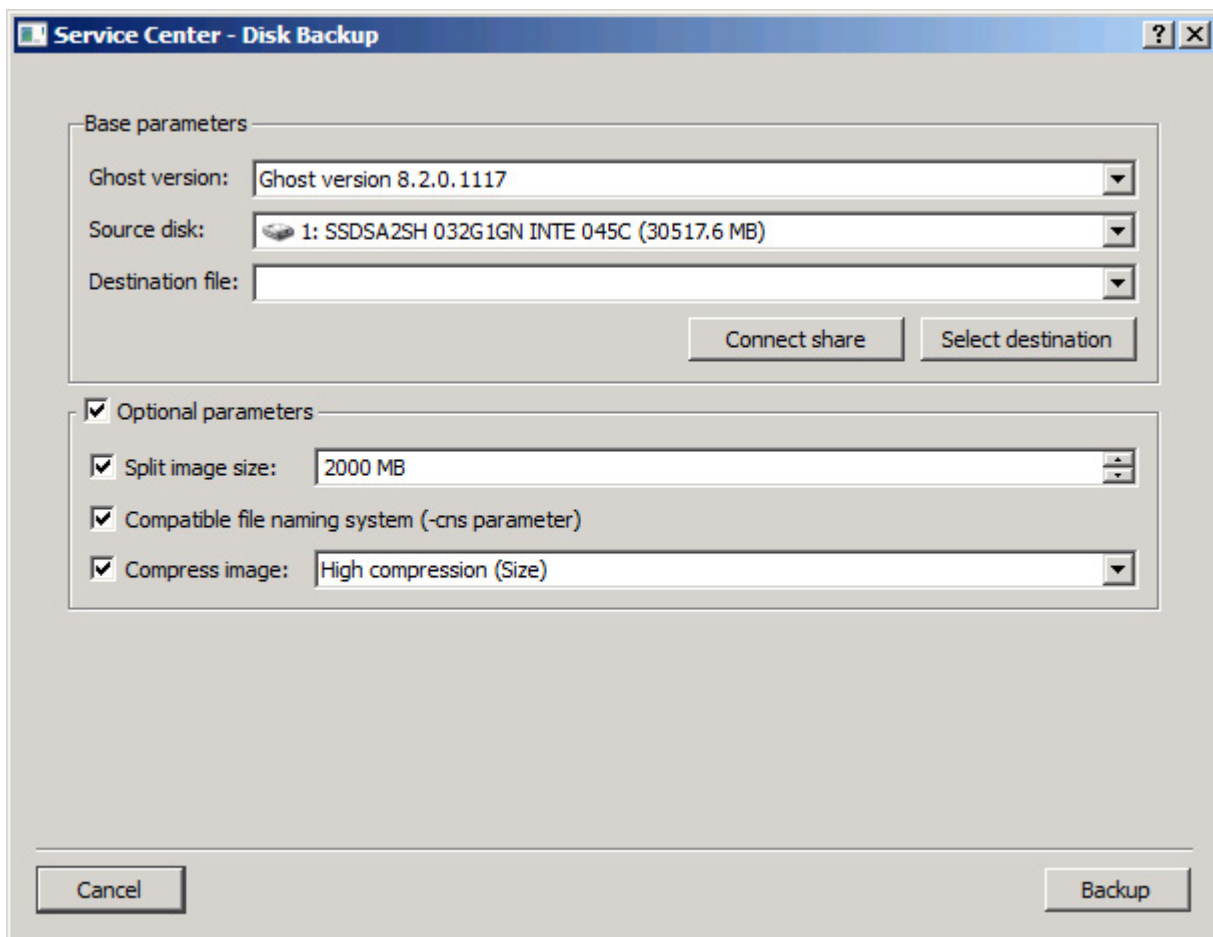


Figure 4-4 Service Center - Disk Backup

4.6 Restore a disk image of the SSD

3. Make the required settings here:

Section	Settings	Purpose
Base parameters	Ghost version	Select the Symantec Ghost version that is to be used when creating the disk image: - Version for Windows 7: "Ghost version 11.5.x" - Version for Windows XP: "Ghost version 8.2.x"
	Source disk	Select a data medium of which you want to create a data image.
	Destination file	Enter a destination directory and file name. Disk images can be stored either locally or in the network.
Base parameters > Destination file	Connect share	Open the "Connect a Share" dialog box (Page 102). You can set up a shared resource (e.g. a directory in the network), to which you can store the disk image.
	Select destination	Open the "Select Destination File" dialog box. You can navigate through a local directory on the PCU 50.5 and enter a file name.
Optional parameters	Split image size	Split the disk image and select a split image size. You can split the disk image into several files of a defined size. Split the disk image size in the following cases: - If the file system (e.g. FAT32) can only manage files of a certain size. - If the disk image is to be archived onto multiple data media, for example, when using CDs.
	Compatible file naming system	Select the compatibility of the name scheme. This option is required so that a disk image with older versions of the "PCU base software Windows XP" can be read in by Windows XP. It is not possible to use a different operating system on a PCU 50.x than the preinstalled system (e.g. Windows XP on a PCU 50.5 Windows 7).
	Compress image	Activate Compression or select a compression level. Higher compression results in a smaller file size, but increases the time needed to compress or decompress the data.

4. Confirm your settings with "Backup" to start generation of the disk image.

4.6 Restore a disk image of the SSD

If your system is no longer functioning stably, you can restore it with a disk image.

Note

Restoring replaces all current files

When you restore the PCU from a disk image, all programs, settings, and files are completely replaced by the disk image. You cannot restore individual files or exclude individual files from restoration.

Requirement

A disk image exists

See: Create a disk image of the SSD (Page 93)

Procedure

To restore a disk image, proceed as follows:

1. Start the Service Center, e.g. via the Windows boot menu.
See also: Starting the Service Center (Page 87)
2. In the main menu of the Service Center, click "Disk Restore".

The "Disk Restore" dialog box opens.

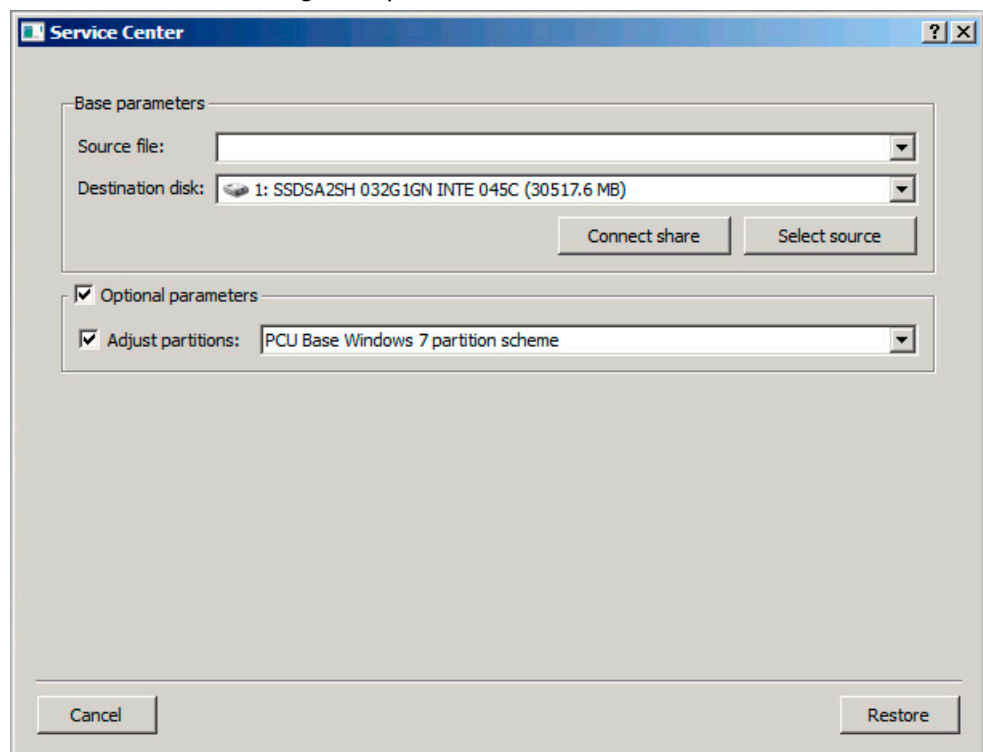


Figure 4-5 Service Center - Disk Restore

4.6 Restore a disk image of the SSD

3. Make the required settings here:

Section	Settings	Purpose
Base parameters	Source file	Select the disk image from which the SSD is to be restored. You can use local disk images or disk images stored in the network.
	Destination disk	Select the SSD to be restored from the disk image.
Base parameters > Destination disk	Connect share	Open the "Connect a Share" dialog box (Page 102). You can manage a shared resource (e.g. a directory in the network), to use a disk image stored on the network to restore the SSD.
	Select source	Open the "Select Source File" dialog box. You can navigate through a local directory on the PCU 50.5 and select a disk image for restoration.
Optional parameters	Adjust partitions	Select partition scheme to be used on restoration of the data medium image: • If the PCU base software Windows 7 is on the disk image, choose "PCU Base Windows 7 partition scheme". • If the PCU base software Windows XP is on the disk image, choose "PCU Base Windows XP partition scheme". It is not possible to use a different operating system on a PCU 50.x than the preinstalled system (e.g. Windows XP on a PCU 50.5 Windows 7).

4. To start the restore process and entirely replace the existing data on the SSD with the disk image, confirm the settings with "Restore".

Result

The restore process is started.

Note**Troubleshooting if reading in of the disk image fails**

If data transfer is interrupted during the restore process, the PCU will have no executable system.

In this case, use a bootable USB flash drive with Service Center to read the disk image in again.

4.7 Create a disk image of a partition

You can create a disk image of an individual partition of the SSD in the Service Center with the "Partition Backup" function.

Procedure

To save a partition of the SSD as a disk image, proceed as follows:

1. Start the Service Center, e.g. via the Windows boot menu.
See also: Starting the Service Center (Page 87)
2. In the main menu of the Service Center, click "Partition Backup".
The "Service Center - Partition Backup" dialog box opens.

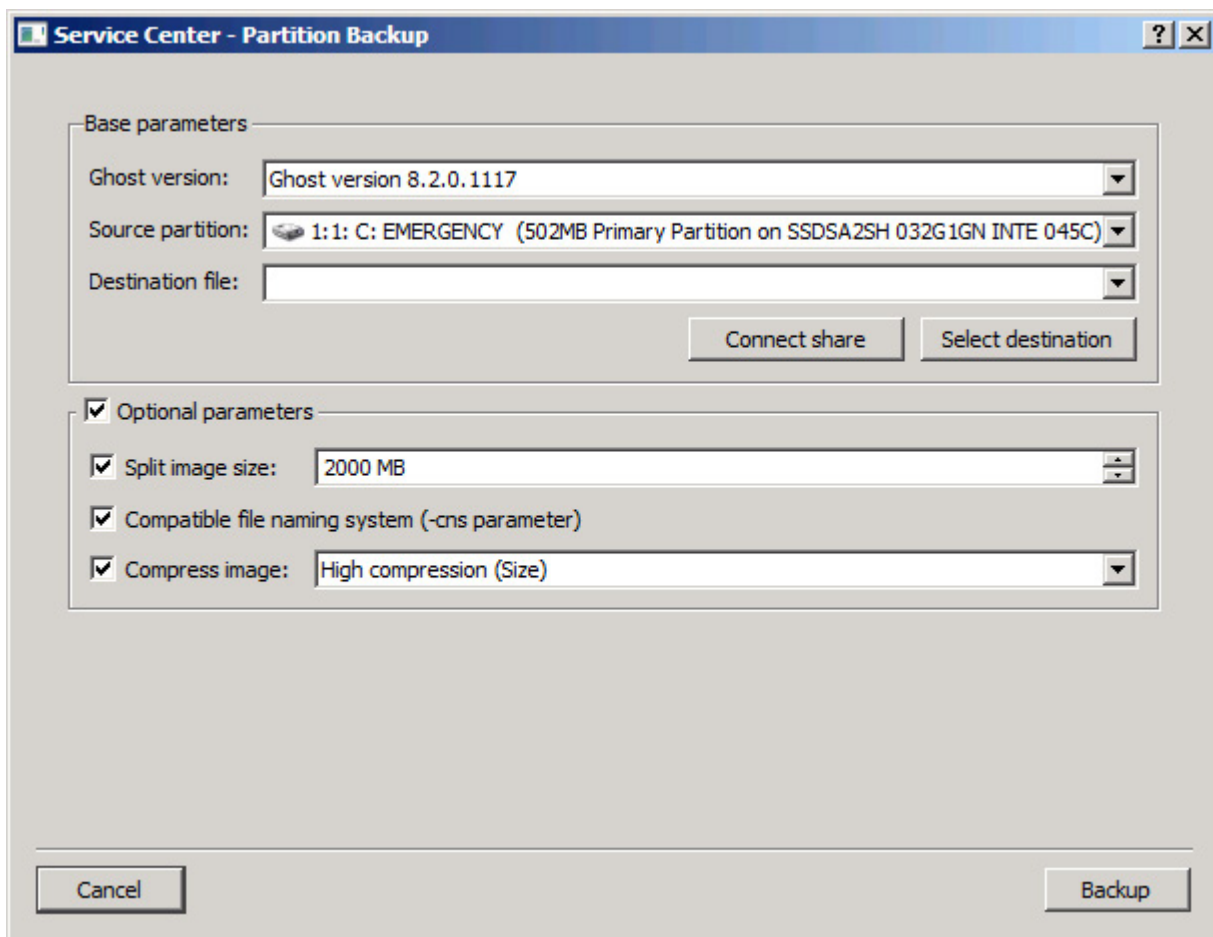


Figure 4-6 Service Center - Partition Backup

3. Make the required settings here:

Section	Settings	Purpose
Base parameters	Ghost version	Select the Symantec Ghost version that is to be used when creating the disk image: - Version for Windows 7: "Ghost version 11.5.x" - Version for Windows XP: "Ghost version 8.2.x"
	Source partition	Select a data medium of which you want to create a data image.
	Destination file	Enter a destination directory and file name. Disk images can be stored either locally or in the network.
Base parameters > Destination file	Connect share	Open the "Connect a Share" dialog box (Page 102). You can set up a shared resource (e.g. a directory in the network), to which you can store the disk image.
	Select destination	Call the "Select destination file" dialog. You can navigate through a local directory on the PCU 50.5 and enter a file name.
Optional parameters	Split image size	Split the disk image and select a split image size. You can split the disk image into several files of a defined size. Split the disk image size in the following cases: - If the file system (e.g. FAT32) can only manage files of a certain size. - If the disk image is to be archived onto multiple data media, for example, when using CDs.
	Compatible file naming system	Select the compatibility of the name scheme. This option is required so that a disk image with older versions of the "PCU base software Windows XP" can be read in by Windows XP. It is not possible to use a different operating system on a PCU 50.x than the preinstalled system (e.g. Windows XP on a PCU 50.5 Windows 7).
	Compress image	Activate Compression or select a compression level. Higher compression results in a smaller file size, but increases the time needed to compress or decompress the data.

4. Confirm your settings with "Backup" to start generation of the disk image.

4.8 Restore a disk image of a partition

If your system is no longer functioning stably, you can restore it with a disk image.

Note

Restoring replaces all current files

When you restore a partition from a disk image, all programs, settings, and files are completely replaced on this partition by the disk image. You cannot restore individual files or exclude individual files from restoration.

Requirement

A disk image of a partition exists

See: Starting the Service Center (Page 87)

Procedure

To restore a partition from a disk image, proceed as follows:

1. Start the Service Center, e.g. via the Windows boot menu.
See also: Create a disk image of a partition (Page 97)
2. In the main menu of the Service Center, click "Partition Restore".

The "Partition Restore" dialog box opens.

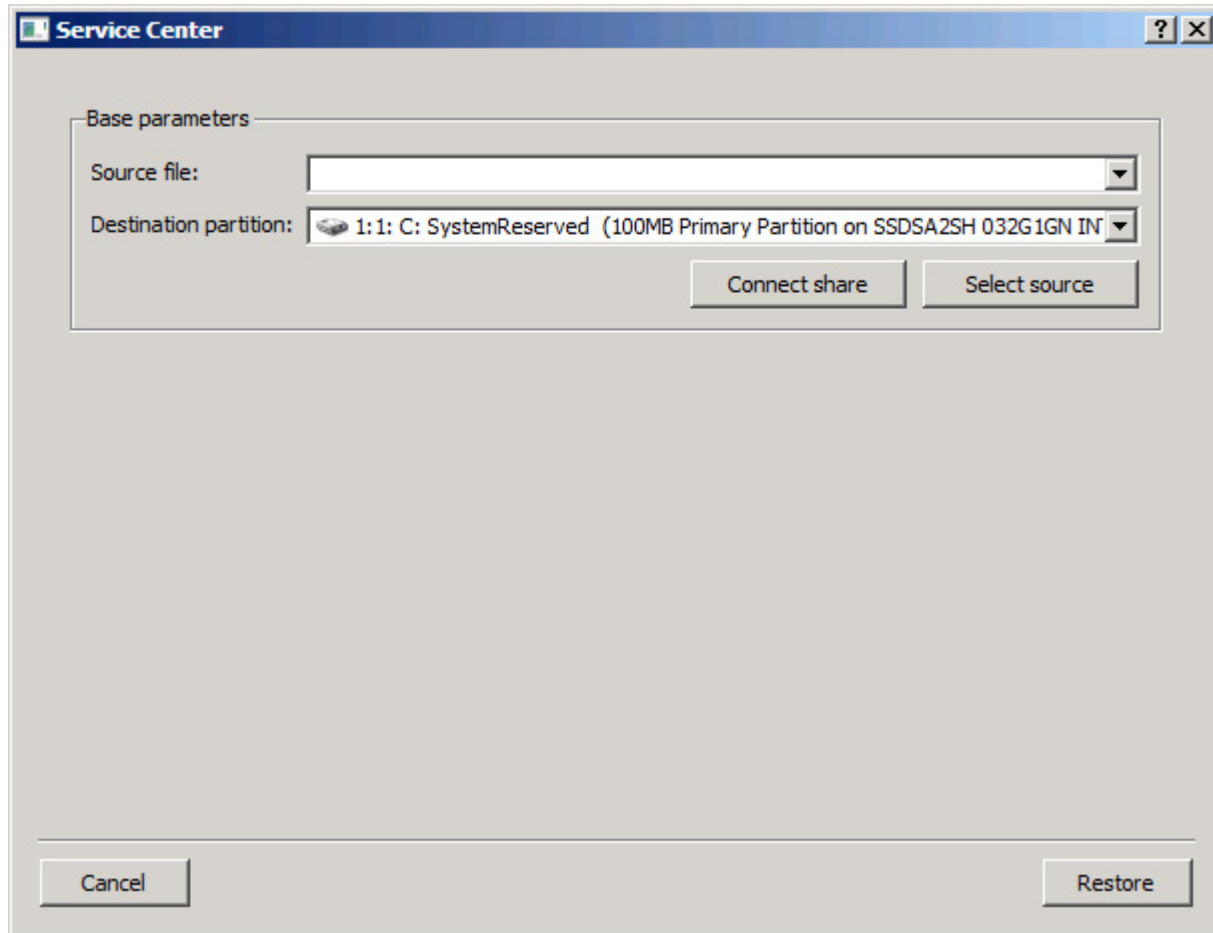


Figure 4-7 Service Center - Disk Restore

3. Make the required settings here:

Section	Settings	Purpose
Base parameters	Source file	Select the disk image from which the partition is to be restored. You can use local disk images or disk images stored in the network.
	Destination partition	Select the partition to be restored from the disk image.
Base parameters > Destination partition	Connect share	Open the "Connect a Share" dialog box (Page 102). You can manage a shared resource (e.g. a directory in the network), to use a disk image stored on the network to restore the SSD.
	Select source	Call the "Select source file" dialog. You can navigate through a local directory on the PCU 50.5 and select a disk image for restoration.

4. To start the restore process and entirely replace the existing data on the partition with the disk image, confirm the settings with "Restore".

Result

The restore process is started.

Note

Troubleshooting if reading in of the disk image fails

If data transfer is interrupted during the restore process, the PCU will have no executable system.

In this case, use a bootable USB flash drive with Service Center to read the disk image in again.

4.9 Network settings in the Service Center

To manage a disk image in the network, you can include a shared network directory on the PCU 50.5 as a drive.

In the Service Center, you configure network drives in the "Connect a share" dialog box. You can call this dialog box directly from the relevant dialog boxes to both create (Page 93) and restore (Page 95) disk images.

Requirement

The network directory is shared with the relevant user accounts in the domain and on the PCU.

See also: Release directory of the PC/PC in the network (Page 114)

Overview

The setting options in the "Connect a share" dialog box have the following meaning:

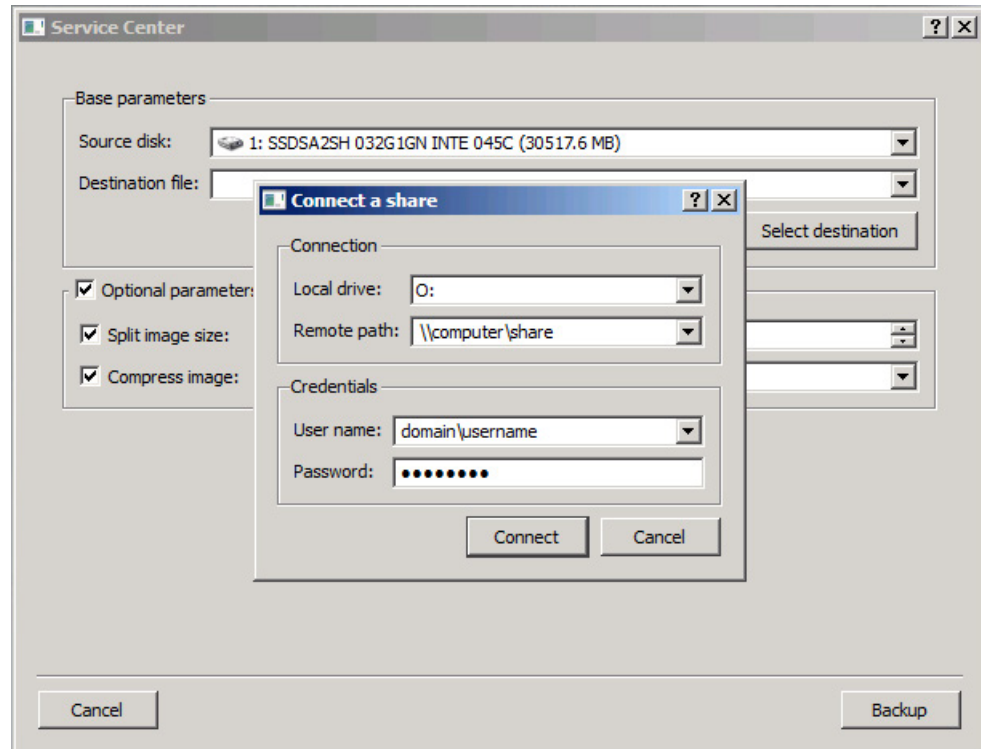


Figure 4-8 "Connect a Share" dialog box

Table 4- 4 Settings in the "Connect a Share" dialog box in the Service Center

Section	Settings	Purpose
Connection	Local drive	To select drive letters under which the network directories are to be accessible on the PCU 50.5.
	Remote path	Specify the device in the network and share name of the resource. Syntax: \\Name of the server\share name of the network directory Example: \\Backup_Server\PCU_Backup
Credentials	User name	Enter the user account under which the resource in the network can be accessed. If you have shared the resource on the server for certain users only, you must enter one of these user accounts here.
	Password	Enter the password of the user account.

Further information

You will find information about releasing directories in the network and network drives in the following sections:

- Release directory of the PC/PC in the network (Page 114)
- Creating a shortcut to the network drive (Page 119)

4.10 Starting Symantec Ghost directly

With the "Symantec Ghost" software you can store the entire contents of an SSD as a disk image. Disk images can be kept on various storage media for subsequent restoration of the data on the SSD.

Symantec Ghost is supplied from the factory on replacement SSDs for the PCU and on the SSD of the entire PCU. Symantec Ghost is used by the components of the PCU Base Software to back up or restore data.

You can also start Symantec Ghost directly from the prompt to execute the data backup tasks from the user interface of Ghost. This is necessary, for example, when using CompactFlash cards.

Procedure

To start Symantec Ghost directly, proceed as follows:

1. Call the prompt from the Service Center or the Service Desktop.
2. Enter "ghost.32.exe" and confirm the entry.

Further information

More information is available in the Internet on website Ghost.com (<http://www.ghost.com>).

4.11 Bootable USB flash drive

Use

You can generate a bootable Service Center (Emergency Boot System) based on Microsoft Windows PE on a USB flash drive as preventive measure against a defect

The Ghost image is located on the SSD under: `D:\Eboot\eboot.gho`

Using the same procedure, you can create a Service Center on a PG/PC. For this, Symantec Ghost must be installed on the PG/PC.

Precondition

- The Windows 7 Service Desktop is active
- You have a USB flash drive (e.g. SIMATIC PC USB flash drive)

Procedure

To create a bootable USD flash drive, proceed as follows:

1. Connect the USB flash drive to a USB interface of the PCU.
2. Start Symantec Ghost, for example, by entering `Ghost32.exe` in the search bar in the start menu.
3. In Symantec Ghost, select the "From Image" command from the "Local > Disk" menu.
4. Select `D:\Eboot\eboot.gho` as the source and the USB flash drive as the destination.

Result

The Emergency Boot System is installed on the USB flash drive and the data medium is renamed to "EBOOT." The service system on the bootable USB flash drive can now be used.

4.12 Operating a service PC/PC in the network

4.12.1 Overview

System network

In the following applications, you will need, for example, a connection between the PCU and a PG/PC in the system network:

- Save the disk image of the PCU-SSD to a PG/PC
- Restore the SSC of a PCU via the DVD drive of a PG/PC
- Start up the replacement SSD
- Install the software on the PCU via the DVD drive of a PG/PC

Company network

If you want to connect the service PG/PC via a company network (Eth 1), contact your network service center.

4.12.2 Connection options in the system network

Overview

The following figures show the typical connection options in the system network:

- PCU to "Eth 2" with service PG/PC, directly, using a crossover Ethernet cable
- PCU to "Eth 2" with service PG/PC, via a switch, using a 1:1 Ethernet cable

Table 4- 5 Configuration images: Meaning of the connections

Symbol	Meaning
	Eth 1 as a DHCP client
	Eth 2 with a fixed IP address
	1:1 Ethernet cable
	Crossover Ethernet cable

Configuration with PG/PC directly to PCU

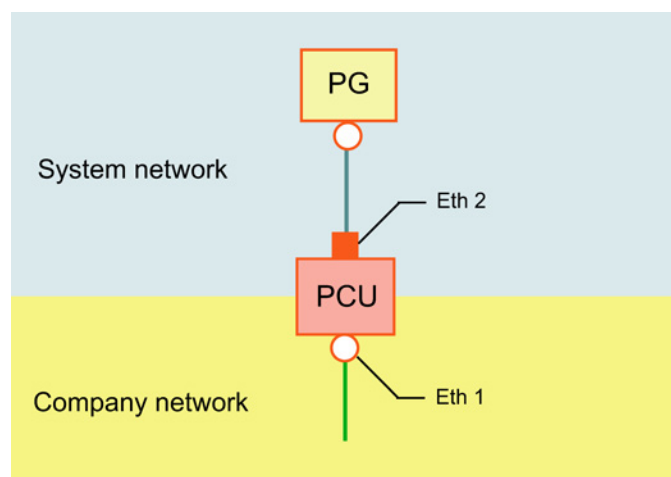


Figure 4-9 Connecting a PG directly to a PCU

Configuration with PG/PC and switch to PCU

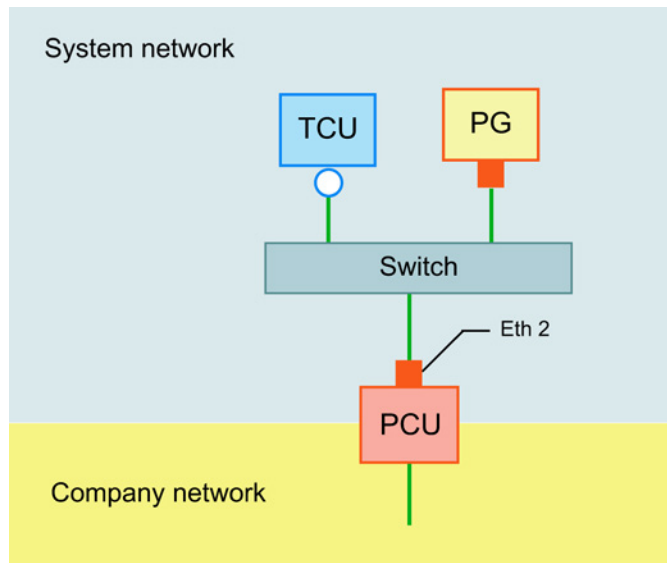


Figure 4-10 Connecting a PG via a switch to a PCU

Note

If a PCU is switched off and on again without its own OP/TP including TCU, and if the PCU boot-up is intended to be performed from the service system (EBS), an external VGA monitor and keyboard are needed to operate the PCU.

An external VGA monitor and keyboard are not needed if the TCUs are operating and the PCU is not switched off with the EBS during booting.

Basic procedure

On the PG/PC with Windows 7:

- Connecting a PG/PC to a PCU according to one of the configurations shown above.
- The network protocol used is: TCP/IP.
TCP/IP is already preconfigured on the PCU.
- Setting up IP addresses on the same subnetwork.
- Releasing a directory on the PG/PC for network access.

On the PCU in the Service Center (WinPE):

- On the PCU start the Service Center (WinPE).
- Establish a network connection to the shared directory of the PG/PC.
See: Functions of the Service Center (Page 88)

4.12.3 Configuring routing in the network

Overview

If you have connected a PG/PC at interface X127 of the NCU, configure the network routing to establish the connection between the PG/PC and the PCU.

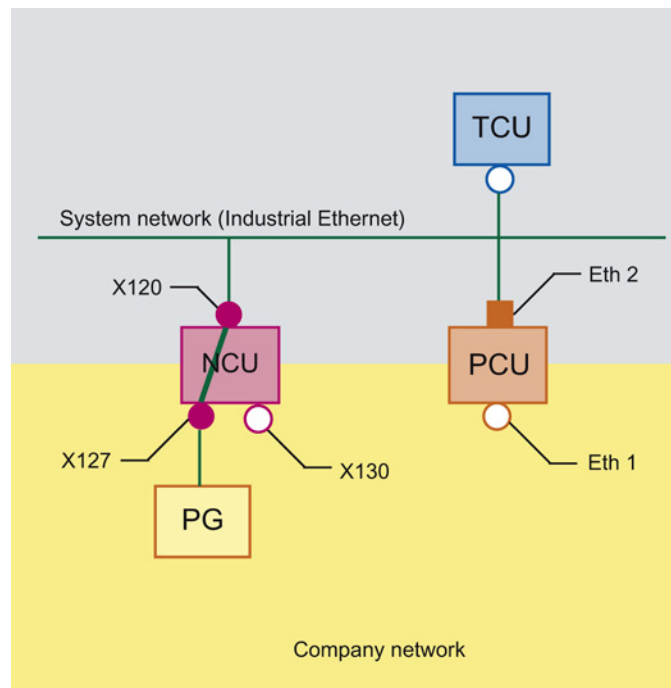


Figure 4-11 A networking example using IP routing

Conditions

The following conditions must be met in order to use this function:

- The PG must be connected to interface X127 of an NCU in the system network.
- The NCU and PCU must be connected via the system network.
- On the NCU, the routing is activated in one of the following ways:
 - In SINUMERIK Operate 4.5 SP2 or higher, the routing is configured in image "Settings of the system network".
 - In the configuration file basesys.ini, key EnableSysNetToIBNForwarding=1 is set.
- If you wish to access a network directory, it must be shared.
- The user account in question is known to the PG/PC.
- The Service Center is active.

Procedure

To configure the IP routing in the Service Center, proceed as follows:

1. In the main menu of the Service Center, click "Network".
2. In the "Network Settings" dialog box, click the "IP Routing" tab.

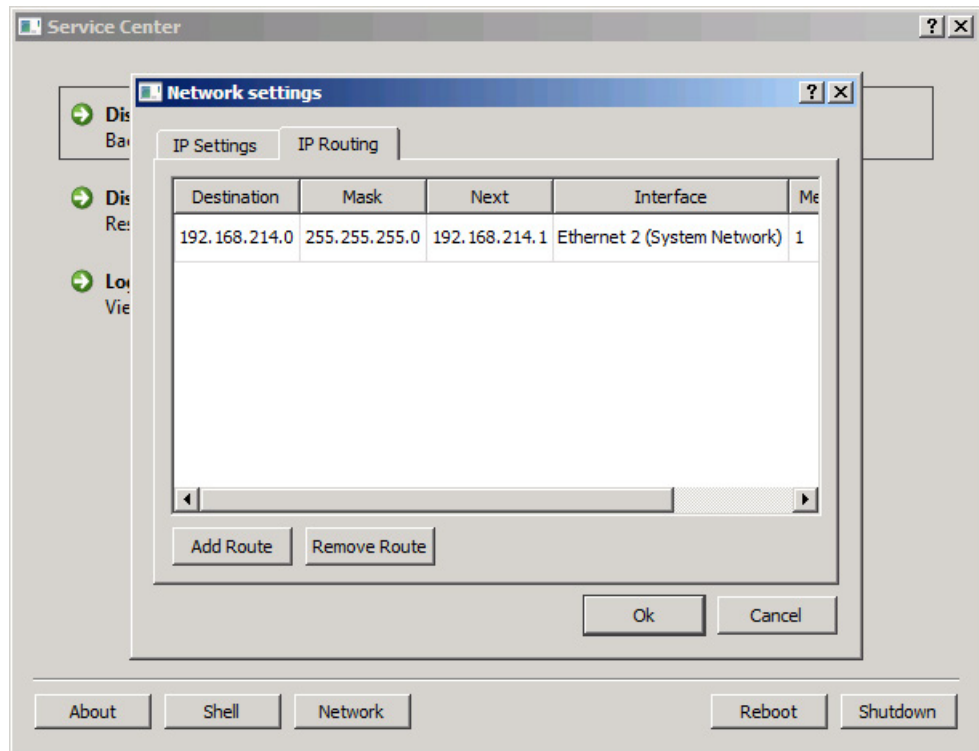


Figure 4-12 Service Center - IP routing

3. Configure the necessary routes. See also: Network settings in the Service Center (Page 90)

Result

The routing has been configured and the PG/PC and PCU are connected via the NCU.

Now, perform the desired steps, for example, a Restore a disk image of the SSD (Page 95).

4.12.4 Configure the network settings of the PG/PC

Once you have connected a PG/PC in the network, make the network settings on the PG/PC.

The basic procedure for setup under Windows 7 or Windows XP is described below. Detailed information from Microsoft is provided in Windows under "Help and Support" or in the Internet: Microsoft Windows support (<http://windows.microsoft.com/en-US/windows/windows-help#windows=windows-7>)

Precondition

- The PG/PC is networked with the PCU in one of the following ways:
 - The PG/PC is connected to the PCU with an Ethernet cable (either directly or via a switch)
See: Connection options in the system network (Page 107)
 - The PG/PC is networked with the PCU via an interface on the NCU and the routing is configured.
See: Configuring routing in the network (Page 109)
- A current Microsoft Windows operating system is installed on the PG/PC
- The PG/PC is switched on

Procedure

To configure the network settings of the PG/PC for connecting a PCU in the system network, proceed as follows:

1. Open the "network connections" window:
 - To do this under Windows XP: in the "Start > Control Panel > Network and Internet Connections" menu, choose "Network Connections."
 - To do this under Windows 7: in the Control Panel in category "Network and Internet > View network status and tasks," choose "Change adapter settings."
2. Right-click on the network adapter you want to configure, then choose "Properties" from the shortcut menu.
This opens the "Properties of <connection>" dialog box.

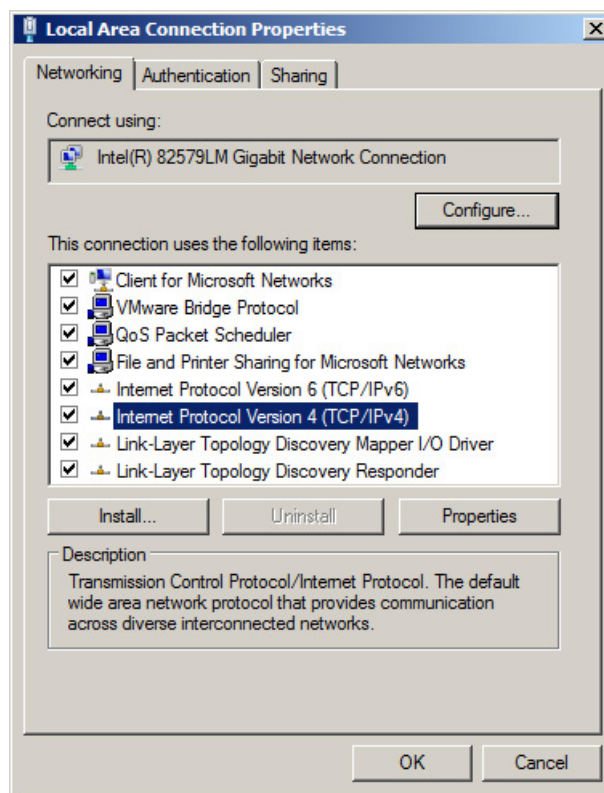


Figure 4-13 Properties of <connection>

3. Make the following settings:

- Click "File and Printer Sharing for Microsoft Networks."
- To change the IP address of the PG/PC, double-click "Internet Protocol Version 4 (TCP/IPv4)", then configure the settings as follows:

Section	Element	Settings
Use the following IP address	IP address	The following IP addresses in the system network can be used for the PG/PC: • 192.168.214.250 • 192.168.214.251 • 192.168.214.252 • 192.168.214.253 • 192.168.214.254
	Subnet mask	• 255.255.255.0

Result

The network settings of the PG/PC are now configured for use with the PCU 50.5.

4.12.5 Release directory of the PC/PC in the network

If you want to, for example, store disk images on a PG/PC that you have linked to the PCU via the network, set up a network drive.

As soon as it has been set up, a new drive will appear on the PCU, which is really a directory on the PG/PC.

The basic procedure for setup under Windows 7 and Windows XP is described below. Detailed information from Microsoft is provided in Windows under "Help and Support" or in the Internet: Microsoft Windows support (<http://windows.microsoft.com/en-US/windows/windows-help#windows=windows-7>)

Requirement

- The PG/PC is switched on.
- There is sufficient space on the PG/PC for saving, for example, disk images from the PCU.
- The user accounts to which you want to grant share rights are known in the domain or on the PCU.

Procedure

Proceed as follows to share a directory for use in the network:

1. On the PG/PC, in the Control Panel call up Computer Management.
2. Under "System > Shared Folders" select the file "Shares."
3. In the "Action" menu, click the "New File Share..." command.
The "Create A Shared Folder Wizard" opens.

4. Under step "Name, Description, and Settings," make a note of the Share path, which is a combination of the computer name and the share name. You will have to specify this name when you create a link to the network drive on the PCU.

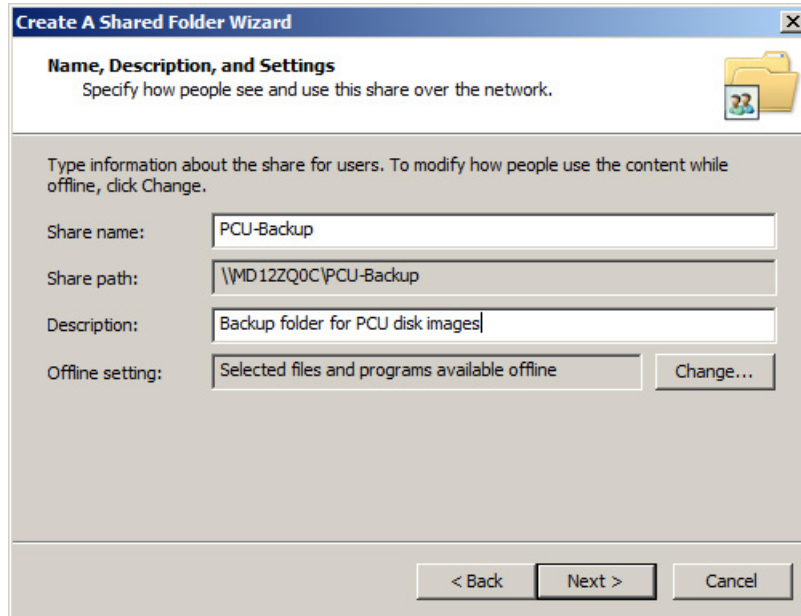


Figure 4-14 Create A Shared Folder Wizard - Name, Description, and Settings

5. Under step "Shared Folder Authorizations," select the "Customize authorizations" radio button, then click "Custom..."



Figure 4-15 Create a Shared Folder Wizard - Shared Folder Authorizations

The "Customize Authorizations" dialog box opens.

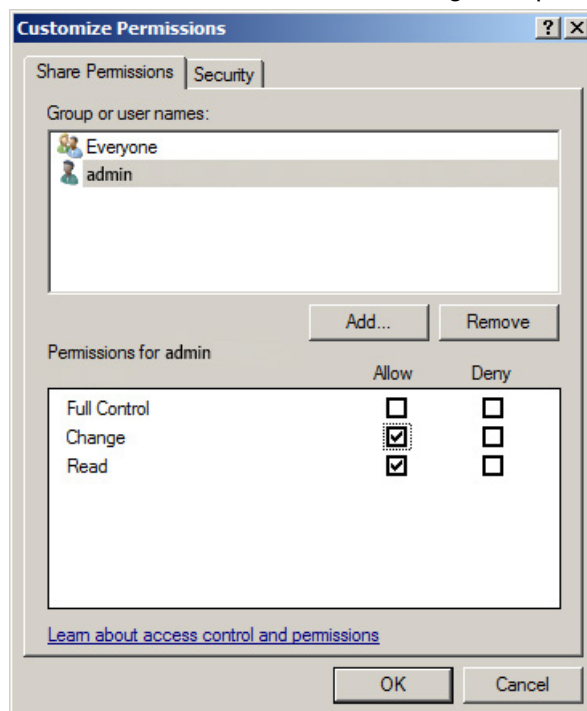


Figure 4-16 "Customize Authorizations" dialog box

6. In section "Group or user names," click "Add..."
The "Select Users, Computers, Service Accounts, or Groups" opens.

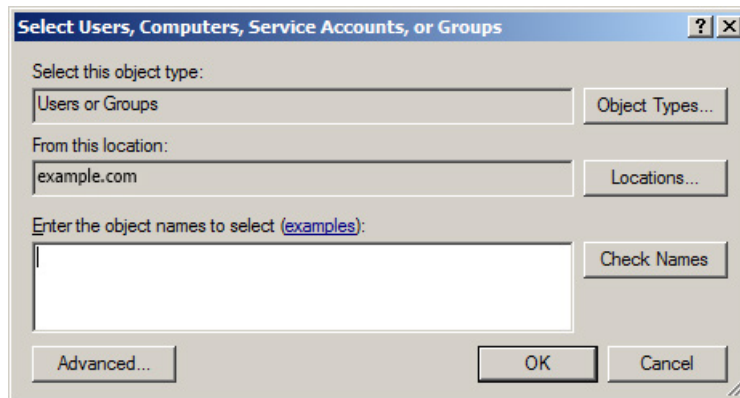


Figure 4-17 "Select Users, Computers, Service Accounts, or Groups" dialog box

7. In section "Authorizations for <user name>," under column "Allow," click the "Change" checkbox.
8. Confirm these settings in the "Customize PermiAuthorizations" dialog box with "OK."
9. In the next steps of the wizard, make all the settings you require and confirm the share with "Finish".

4.12.6 Release drives C: and D: of the PCU 50.5 in the network for administrator accounts

You can release local drives (C\$, D\$) of the PCU 50.5 in the network for administrator accounts

You can then access these PCU 50.5 drives, for example from a PG/PC in the network if you use an administrator user account that is available on the PCU 50.5.

Procedure

1. Enter "regedit" in the search field of the registry.
The registration editor opens.
2. Navigate to the following folder in the folder structure:
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System
3. Right-click in the working area and select the "New > DWORD (32-bit) Value" command in the shortcut menu.
4. In the "Name" field, enter the "LocalAccountTokenFilterPolicy" designation.
5. Double-click the key and enter "1" as "Value".
6. Confirm the input, close the registration editor and restart the PCU 50.5.

Result

The registry entry has been added and the drives are available in the network after the restart.

However, you cannot change the access rights, i.e. access is only possible via an administrator account that is known on the PCU 50.5.

To undo the release, delete the created key in the registration editor.

4.12.7 Creating a shortcut to the network drive

To be able to access the shared directory on the PG/PC from the PCU, after you have connected and configured the PG/PC in the network, you must set up the folder shared on the PG/PC on the PCU as a network drive.

As soon as it has been set up, a new drive will appear on the PCU, which is really a directory on the PG/PC.

The basic procedure for setup under Windows 7 and Windows XP is described below. Detailed information from Microsoft is provided in Windows under "Help and Support" or in the Internet: Creating a shortcut to a network drive (association)

(<http://windows.microsoft.com/en-us/windows7/create-a-shortcut-to-map-a-network-drive>)

Precondition

- The PG/PC is switched on.
- The PCU is switched on and the Service Desktop is active.
- The folder on the PG/PC is shared for the relevant user accounts on the domain and on the PCU.

Procedure

Proceed as follows to connect a shared directory as a network drive:

1. Start the Windows Explorer and click the "Map network drive" button.
2. Make the following settings in the "Map network drive" dialog box:

Settings	Purpose
Drive	To select drive letters under which the network directories are to be accessible on the PCU 50.5.
Folder	Specify the device in the network and share name of the resource. Syntax: \\Name of the server\share name of the network directory Example: \\Backup_Server\PCU_Backup
Reconnect at logon	Select this check box to automatically connect the network drive on booting.
Connect using different credentials	Select this check box if you have not shared the directory in the network with the user for whom you would like to set up the network drive. In the next dialog box "Enter Network Password," enter the log-on information of a user with whom the network directory is shared.

3. Confirm your settings with "Finish."

See also

Microsoft Windows support (<http://windows.microsoft.com/en-US/windows/windows-help#windows=windows-7>)

4.13 Starting up the replacement SSD

Information about replacing and installing an SSD on the PCU is provided in the Operator Components and Networking Manual.

Note

Troubleshooting if reading in of the disk image fails

If reading in of the data is interrupted, the PCU will have no executable system.

In this case, use a bootable USB flash drive with Service Center to read the disk image in again.

See also

Restore a disk image of the SSD (Page 95)

Service and diagnostics

5.1 PCU Hardware Diagnostics

Intended use

The PCU hardware supports the diagnostics of important system components via an integrated "safecard", which is designated as a Safecard-On-Motherboard (SOM).

These diagnostic functions are only evaluated by systems with HMI Advanced. The fault statuses of the hardware are reported in the form of alarms via the operator interface of HMI Advanced. This allows for visualization of the data in HMI Advanced and external evaluation.

Monitored Parameters

The following physical parameters of the PCU hardware are monitored:

- CPU temperature
- Housing temperature
- I/O chip temperature
- Speed of the two housing fans
- S.M.A.R.T - status of the hard disk

Logging faults

The PCU hardware monitor logs all hardware faults in the Windows event log so that the faults can be output even without installing an HMI application.

The alarms are output in the log under "Control Panel" → "Administrative Tools" → "Event Viewer".

See also

- Auto-Hotspot
- Operator Components and Networking Manual: Chapter, spare parts
- Description of the alarms: SINUMERIK 840D sl Diagnostics Manual

5.2 Evaluation of the 7-segment display

Intended use

The 7-segment display is intended for PCU diagnostics in operation without a local OP (known as "Headless mode"). If an error is detected during booting, a local display is needed for subsequent error analysis of the connection.

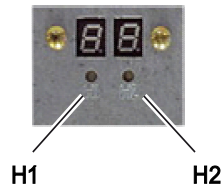


Figure 5-1 7-segment display with LEDs

The two displays have the following function:

- Segment display H1
Segment display H1 and the associated LED are assigned to the PCU basic software. Status codes are output during system booting, normal operations and shutdown. Compatibility in relation to the NCU modules is taken into account.
- Segment display H2
Segment display H2 and the associated LED are assigned to the application software.

Meaning of status codes

- During system power-up:

LED H1	LED H2	7-segment display		Status	
Orange	Orange	Output of BIOS post codes		After switching on the system	
Orange	Off			After cycling the BIOS	
Orange	Off			After starting Windows	
		1	0	Start Windows	Load device drivers needed for the Windows start
		2	0	PCU hardware service	The PCU hardware service has been started.
		5	0	Network	Wait for network interfaces to be ready.
		8	0	TCU support test step 1	Wait for FTP server to start
		9	0	TCU support test step 2	Wait for boot server to start for TCU network boot and for TCU hardware service to start
		A	0	TCU support test step 3	Wait for VNC server to start
		B	0	HMI manager	Wait for HMI manager to start
		E	0	PCU base software	General fault: see Windows "Event Viewer"

5.2 Evaluation of the 7-segment display

- During operation:

LED H1	LED H2	7-segment display		Status	
Flashing green	--	0	0	OK	
Flashing red	--			Error:	
		1	0	Temperature alarm	Housing or CPU temperature; temperature above limit value (SOM error)
		2	0	Fan alarm	Housing or CPU: Low fan speed or fan failure (SOM error)
		3	0	Hard disk alarm	S.M.A.R.T hard disk error
		6	0		Failure of VNC server or VNC server service stopped

- During shutdown

LED H1	LED H2	7-segment display		Status	
Flashing red/green	--	5	0	Shutdown active	

5.3 Setting of the service switch (PCU 50.5 only)

Intended use

The "PCU Hardware Service" component as part of the PCU Base software provides a function to read-out the switch positions.



Figure 5-2 Rotary coding switch (hexadecimal)

Meaning of switch positions

Only the settings of the service switch that are relevant for the commissioning engineer are listed:

Switch setting	Mode of operation	Function
0	---	Normal operation (default setting)
6	Users	Reserved for additional software
9	Service	Headless operation without OP/TP
A	Service	Firmware recovery
C	Service	Checking
D	Service	Diagnostics
E	Service	Booting from the emergency boot system

5.4 Enabling/disabling error log during boot up

Application

Each time the system is booted up, information is written to a block in the file D:\\$\$Base.log, which contains the date, time and nature of administrative interventions.

Information about the booting controlled by the HMI Manager, can be displayed on the screen and written to log file D:\\$\$Base.log.

Output in log file

The output is controlled by registry values:

- **Type of information**

The type of information to be output is set via registry value "HKLM\SOFTWARE\Siemens\SINUMERIK\Basic software\<version>\HMI Manager\InfoLevel":

InfoLevel (DWORD)

- = 1: Mandatory information is output (default setting)
- = 2: Mandatory and supplementary information is output
- = 3: Mandatory, supplementary and trace information is output
(InfoLevel <= 0 is treated as InfoLevel == 1;
InfoLevel <= 3 is treated as InfoLevel == 3)

- **Outputting information to the screen**

Whether the information is also to be displayed on the screen (as well as being output to the log file) is set via registry value 'HKLM\SOFTWARE\Siemens\SINUMERIK\Basic software\<version>\HMI Manager\ShowInfo':

ShowInfo (DWORD)

- = 0: Displays the mandatory information,
no display of the supplemental and trace information
- = 1: Mandatory, supplementary and trace information is displayed (default)

5.5 How to search for stations within the system network

Intended use

The VNC scanner is used within the system network. The scanner is primarily used to identify devices containing an active VNC server, e.g. NCU or PCU.

The VNC scanner offers the option of connecting directly to the VNC server and, for example, operating an HMI application. The integrated VNC viewer is used to do this.

Furthermore, it enables you to call up a list of all the network devices, thereby creating an overview of the system network.

Note

The VNC scanner is available on the Service Desktop of the PCU and on the PCU base software DVD for installation on a PG/PC.

You require a mouse to use this program.

Operating "VNC scanner"



1. You can launch the "VNC Scanner" program via this link on the Service Desktop of the PCU .

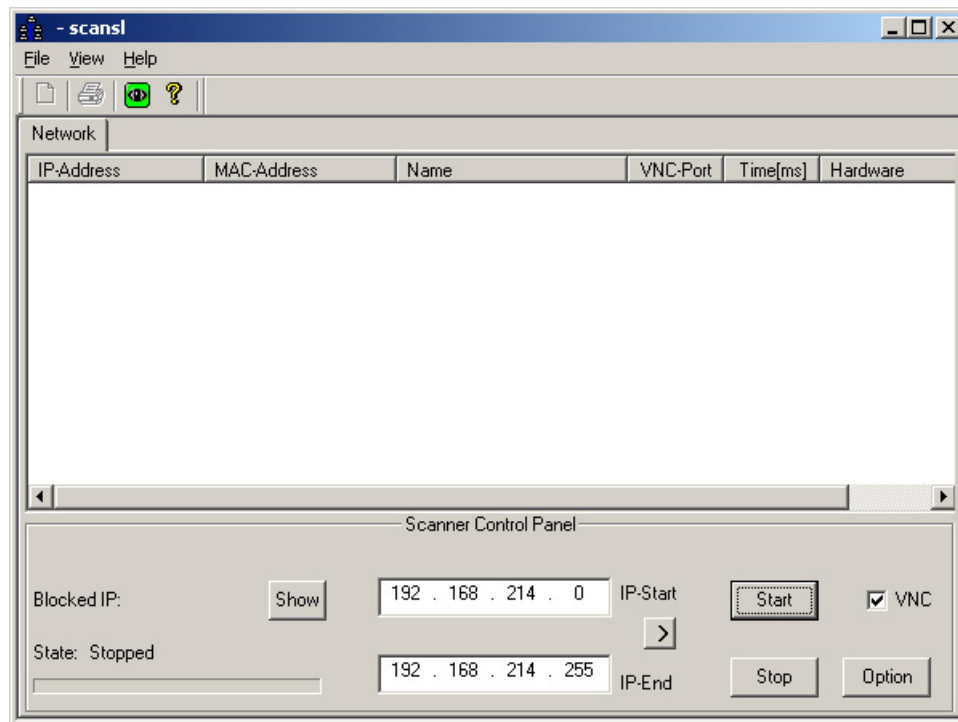


Figure 5-3 VNC scanner

2. "IP start / IP end": Set the IP-address browsing range.
Eight ranges can be saved. Click on button ">" to switch to the next range.
3. "VNC" option:
 - ON: only search for VNC servers.
 - OFF: search all the stations in the network.
1. Start the search with the "Start" button.
The following station parameters will be listed:
 - IP address
 - MAC address
 - Name in the network
 - VNC port
 - Response time of the station in milliseconds
 - Device type

Note**local device #**

When the devices are listed, an IP address will be identified with "#" to indicate that this is the local device on which the VNC scanner will be used.

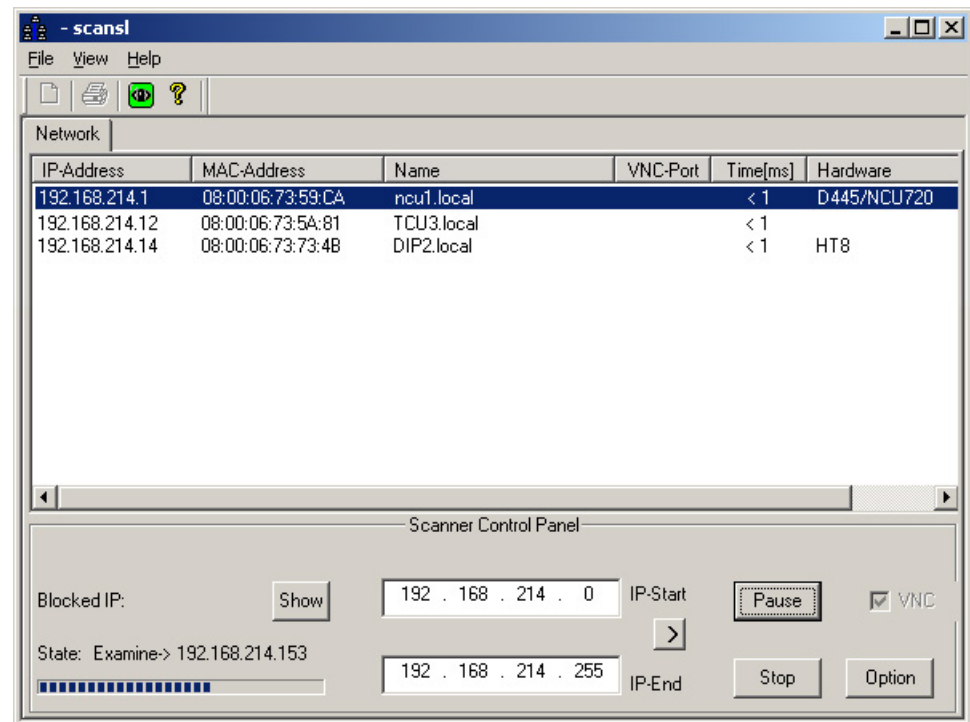


Figure 5-4 VNC scanner: List of stations

2. By right-clicking a network station from the list, you can execute the following actions via a pop-up menu:
 - "VNC": Connecting directly to the VNC server and, if applicable, operating the HMI.
 - "VNC → Auto reconnect": Corresponds to the option "Automatic reconnect if the server closes the connection" in order to restore the connection to the selected network station.
 - Ping: Opens a command shell and executes the "ping" command to the selected network station.

Note

Operation via a VNC viewer:

A system can only be monitored from another station via an external VNC viewer (default).

To view the system from another station, this function must be enabled by the system. You can find the settings for this in the "System Network Center" or directly in tcu.ini in the [VNCViewer] section.

Should you wish to access a PCU via the company network using the VNC scanner or VNC viewer, port 5900 must be added to the PCU under "Control Panel" → "Windows Firewall" → "Exceptions".

Reference: Operator Components and Networking Manual

Additional Options



1. Use this button to start the VNC viewer.
2. Use the "Option" button to set parameters for the search.

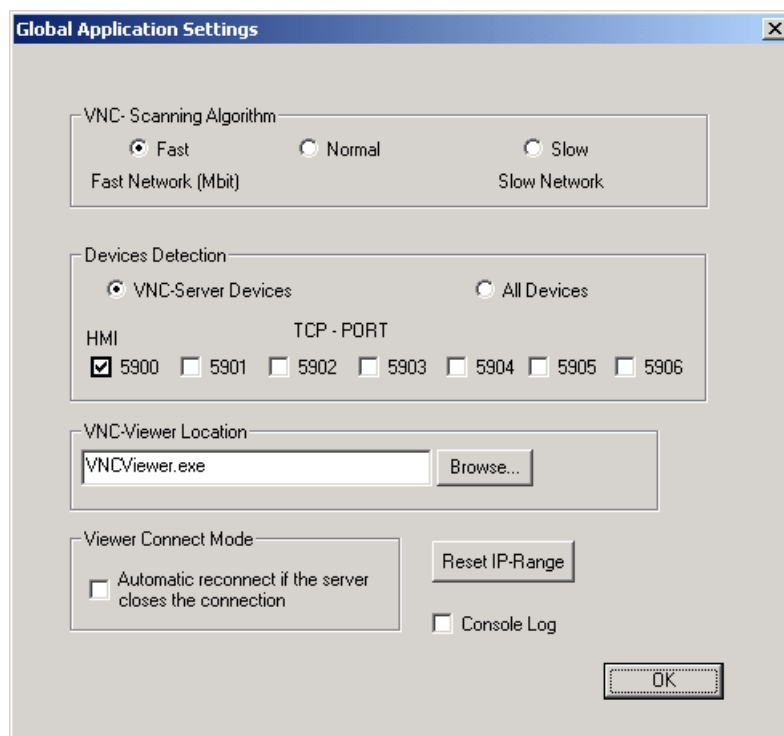


Figure 5-5 VNC scanner: Set search parameters

- VNC scanning algorithm:
Default is "Fast", i.e. device feedback within approximately 100 ms
- Devices detection: (corresponds to the "VNC" option, see point 3).
 - "VNC server devices": only search for VNC servers.
 - "All devices": search all the stations in the network.
- VNC viewer location:
Link to the VNC viewer (open source program: already pre-installed on the PCU; on a ServicePC this link must be specified).
- Viewer default connection mode:
"Automatic reconnect if the server closes the connection" (default: OFF)
 - OFF: One attempt is made to establish a connection.
 - ON (loop mode): For situations where access will be via Internet, for example for service purposes, then the system will wait longer and try again after several milliseconds to establish a connection to this station.
- "Reset IP range": All IP address ranges are reset.
- Console log: Opening a console to output a protocol, e.g. for error analysis (default: OFF).

Options for the VNC viewer

NOTICE

Do not change defaults!

To ensure that the VNC viewer works properly, the following options may not be changed.

After starting the VNC viewer, the following dialog opens:

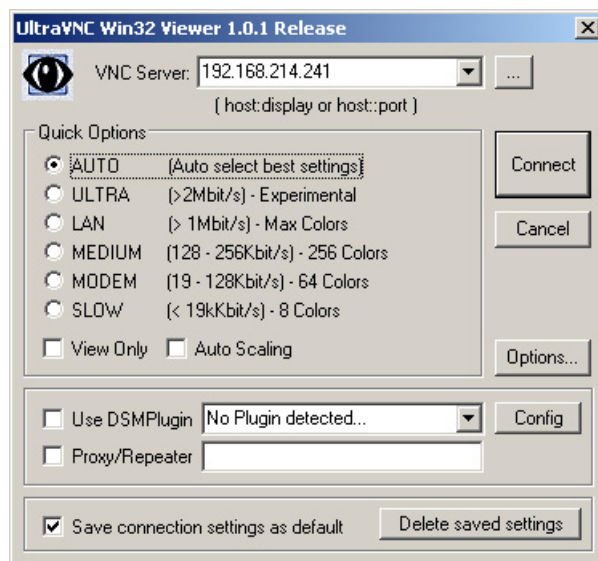


Figure 5-6 Default: UltraVNC

After clicking the "Options ..." button, the following dialog opens:

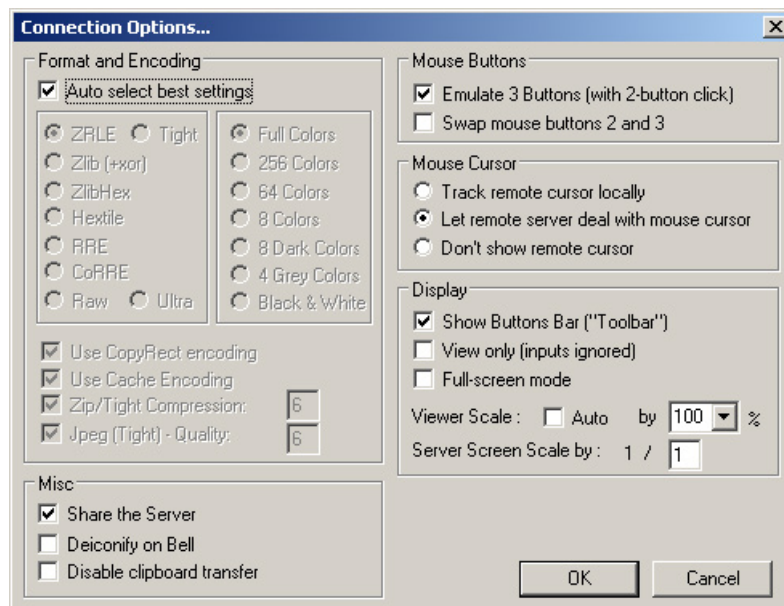


Figure 5-7 Default: Connection Options ...

List of Abbreviations

A.1 Abbreviations

CF	CompactFlash card: Memory card
CFS	Cluster File System
DCK	Direct Control Keys: Direct control keys
DCP	Discovery and Basic Configuration Protocol
DHCP	Dynamic Host Configuration Protocol: Dynamic assignment of an IP address and other configuration parameters on a computer in a network
DNS	Domain Name System: Conversion of domain names into IP addresses
EBS	Emergency Boot System
EKS	Electronic Key System: System to check the identity of a user (authentication system)
EUNA	End User Notification Administration
HMI	Human Machine Interface: Operator interface
IRT	Isochronous Realtime (Ethernet)
LLDP	Link Layer Discovery Protocol: multi-vendor Layer 2 Protocol defined in accordance with the IEEE-802.1AB standard, allows information to be exchanged between devices.
MAC	Media Access Control: The MAC address is a 48-bit Ethernet ID.
MCP	Machine Control Panel: Machine control panel
MPI	Multi-Point Interface: Multiple interface
MUI	Multilanguage User Interface
NCK	Numerical Control Kernel: NC kernel with block preparation, travel range, etc.
NCU	Numerical Control Unit: NCK hardware unit
NRT	Non-Realtime (Ethernet)
NTFS	New Technology File System
NTP	Network Time Protocol: Standard for synchronizing clocks in the entire network
NTPD	NTP Daemon: Utility that runs in the background and does not have to be started by the user.
PCU	PC Unit: Computer unit
PDEV	Physical device
PG	Programming device
PLC	Programmable Logic Control: Programmable logic controller
RAM	Random Access Memory: Program memory which can be read and written into
RDY	Ready: The system is ready to operate.
SNMP	Simple Network Management Protocol (network protocol for monitoring and controlling network elements such as routers, servers, switches, and printers from a central station).
SSD	Solid State Drive
TCU	Thin Client Unit
TFTP	Trivial File Transfer Protocol: Very simple data transmission protocol
UDP	User Datagram Protocol: NTP is mostly processed via UDP.

List of Abbreviations

A.1 Abbreviations

USB	Universal Serial Bus
UPS	Uninterruptible power supply
UTC	Universal Time, Coordinated: Coordinated universal time
VNC	Virtual Network Computing

Index

7

7-segment display, 122

A

Activating the firewall, 9

B

Backup time, 69

BIOS

Changing default setting, 37, 42

Start setup, 33, 38

System parameters, 34, 39

Boot screen

OEM-specific, 53

Storage location, 53

Boot stick, 105

C

Changing computer name, 44

Communication

MPI, 43

S7DOS, 43

Configuration file

basesys.ini, 109

ghost.ini, 73

servicesystem.ini, 73, 76

Configuring the user interface, 50

D

Dialog box

ServiceCenter Backup/Restore, 73, 76

Disk image

Redo, 95, 99

Troubleshooting, 96, 101

Domain, 47

Driver

S7DOS, 43

E

Emergency Boot System, 105

Enabling the log, 126

Error log, 126

G

Global Settings, 13

H

Hard disk

Replacing, 120

M

Mains-fed operation, 78

Multilingual User Interface (MUI), 84

N

Network

Adapting settings, 74, 76

Network release

Drive of the PCU 50.5, 118

Driven on PG/PC, 114

NTFS file system, 7

P

Partitioning the hard disk, 7

Password

For network drive, 102

PCU

Checking the hard disk, 121

Delivery condition, 5

Name (default setting), 9

Operating system, 9

PCU operating system, 9

Port 80 display, (See 7-segment display)

powerline, 43

R

Recovery Media, 84
RESOLUTION (tcu.ini), 30

S

Screen resolution, 30
Select language (MUI), 50
Service Desktop
 Background, 54
 install, 78
Service switch, 125
Service system for PCU, 105
Shutdown, 124
SIMATIC STEP 7
 Installing, 82
Software
 Components, 6
 install, 77
SSD
 Replacing, 120
Starting OEM programs, 60
System boot, 123

T

Troubleshooting
 Restore failure, 96, 101

U

USB FlashDrive, 105
User
 Change the name, 22
 Default, 11
 Delete, 22
 Inheritance, 19
User group, 11
User type, 11

V

Version software components, 6
VNC scanner, 128

W

Windows language (MUI), 50

SINUMERIK

SINUMERIK 840D sl PCU Base Software (IM10)

Commissioning Manual

<u>Delivery condition</u>	1
<u>First commissioning of the PCU 50.5</u>	2
<u>Configuration of the system</u>	3
<u>Installing and configuring updates and automation software</u>	4
<u>Backing up and restoring data</u>	5
<u>Service and diagnostics</u>	6

Valid for:

Software: SINUMERIK PCU base software V11

SINUMERIK PCU 50.5

01/2015

6FC5397-1DP40-5BA2

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

DANGER

indicates that death or severe personal injury will result if proper precautions are not taken.

WARNING

indicates that death or severe personal injury may result if proper precautions are not taken.

CAUTION

indicates that minor personal injury can result if proper precautions are not taken.

NOTICE

indicates that property damage can result if proper precautions are not taken.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

WARNING

Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Table of contents

1	Delivery condition.....	7
1.1	PCU Base Software Windows 7	7
1.1.1	Overview	7
1.1.2	Supplied software	7
1.2	Hardware configuration of PCU 50.5	8
1.2.1	Network settings	8
1.2.2	Partitioning of the SSD.....	9
1.3	Directory structure and file conventions.....	9
2	First commissioning of the PCU 50.5.....	11
2.1	Overview	11
2.2	Backing up and restoring the delivery condition	13
2.3	Install the PCU Base Software attended	14
2.4	Unattended installation of the PCU Base Software	15
2.4.1	Configuring an answer file	15
2.4.2	Installing the PCU Base Software attended	18
3	Configuration of the system.....	21
3.1	Overview	21
3.2	Managing user accounts.....	21
3.3	Changing the name of the PC system	22
3.4	Setting the IP address and domain.....	22
3.5	Configuring USB interfaces	23
3.6	Configuring the keyboard layout.....	24
3.7	Setting up an external screen	24
3.8	Setting the screen resolution	25
3.9	Set the resolution in the tcu.ini.....	25
3.10	Setting the color depth in the tcu.ini file.....	28
3.11	Configuring the SITOP UPS module for use with the PCU Base Software.....	29
3.11.1	Overview of SITOP UPS.....	29
3.11.2	SITOP modules for the PCU 50.5.....	30
3.11.3	Configuring SITOP software for PCU Base Software.....	31
3.11.4	Parameterizing a delay time for quitting the HMI software	35
3.11.5	Hardware configuration of the SITOP UPS module	36
3.12	Configuration of the Service Center.....	39
3.12.1	Overview	39
3.12.2	Configuration of the network adapter.....	40
3.12.3	Configuration of the host.....	42

4	Installing and configuring updates and automation software	43
4.1	Overview	43
4.2	Patch management and security updates	43
4.3	Setting up SINUMERIK Operate for autologon mode.....	44
4.4	Modifying the user account for autologon mode.....	45
4.5	Configuring a key filter for HMI software.....	47
4.6	Installing STEP 7.....	50
4.7	Linking the HMI software with STEP 7	51
4.8	PCU Installer.....	52
4.8.1	Overview	52
4.8.2	Activating the PCU Installer	53
4.8.2.1	Overview	53
4.8.2.2	Activating via a configuration template	53
4.8.3	Deactivating the PCU Installer	54
4.8.4	Copying and adapting the configuration file.....	55
4.8.5	Reference to the PCUInst.ini configuration file	56
4.8.6	Example configuration: Installation behavior as for PCU base software Windows XP	60
4.8.7	Installing software with the PCU Installer	60
4.8.8	Procedure example of an installation with the PCU Installer	61
4.8.9	Log files (.log).....	62
4.9	Migration	63
5	Backing up and restoring data.....	65
5.1	Overview	65
5.2	Starting the Service Center.....	65
5.3	Functions of the Service Center.....	66
5.4	Network settings in the Service Center.....	67
5.5	Create a disk image of the SSD.....	70
5.6	Restore a disk image of the SSD.....	72
5.7	Create a disk image of a partition	74
5.8	Restore a disk image of a partition	76
5.9	Network settings in the Service Center.....	78
5.10	Starting Symantec Ghost directly.....	80
5.11	Bootable USB flash drive	80
5.12	Operating a service PC/PC in the network	81
5.12.1	Overview	81
5.12.2	Connection options in the system network	81
5.12.3	Configuring routing in the network	83
5.12.4	Configure the network settings of the PG/PC	85
5.12.5	Release directory of the PC/PC in the network	88
5.12.6	Creating a shortcut to the network drive	91
5.13	Starting up the replacement SSD	93

6	Service and diagnostics	95
6.1	PCU Hardware Diagnostics	95
6.2	Evaluation of the 7-segment display.....	95
6.3	Setting of the operating mode during run-up	97
6.3.1	Service switch (PCU 50.5)	97
6.3.2	Possible operating modes	99
6.4	Switching to the Service Desktop during autostart / autologon operation	99
6.5	Remote access	101
6.5.1	Overview	101
6.5.2	Searching for devices in the system network	101
6.5.3	Display of accessible stations in SINUMERIK Operate.....	101
6.5.4	Remote access to the PCU for operation and maintenance	102
6.5.5	Setting up and using SSH.....	103
6.5.6	Encryption via SSH protocol	104
	Index.....	107

Delivery condition

1.1 PCU Base Software Windows 7

1.1.1 Overview

The **PCU base software Windows 7** provides the basis for operating SINUMERIK software (e.g. SINUMERIK Operate) on your SIEMENS PC system.

- **SINUMERIK PCU 50.5**

The PCU base software / 50.5W7 is included in the PCU 50.5 scope of delivery and preinstalled. However, for security reasons, the final installation and setup of user accounts must be performed during the first startup.

An example flowchart for initial startup is provided as an overview in Chapter Overview (Page 11). This overview assumes that the PCU 50.5 is already installed and networked.

Information about the hardware configuration, installation, and networking of the PCU 50.5 is provided in the *SINUMERIK 840D sl Operator Components and Networking Manual*, Chapter *Operator panels*.

Note

Do not switch off SIEMENS PC systems with Windows 7 on the hardware!

Loss of data may result on Windows-based systems if the system is not shut down properly before it is disconnected from the power source. For technical reasons, data is still being written to the SSD shortly after shutdown.

To avoid loss of data, use a SITOP UPS module.

See Configuring the SITOP UPS module for use with the PCU Base Software (Page 29).

1.1.2 Supplied software

Overview

The SINUMERIK PCU base software contains the following software and tools:

Software	Function
Microsoft Windows 7	Service Desktop
Microsoft WinPE	Service Center
Symantec Ghost	Creating and restoring disk images

Software	Function
SIEMENS PCU Installer	Installing software and updates
VNC Viewer	Remote access to other devices from the PCU
PuTTY	Remote access to other devices from the PCU
PuTTY Key Generator	Generating an SSH key pair
Various configuration templates and scripts in directory: C:\ProgramData\Siemens\MotionControl\SIEMENS	Templates for simplifying configuration

Note**Acknowledgements**

- This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)
- This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (<http://www.openssl.org/>)

1.2 Hardware configuration of PCU 50.5

1.2.1 Network settings

The PCU 50.5 has two Ethernet interfaces with default settings suitable to connect to SINUMERIK solution line:



Figure 1-1 Ethernet interfaces of the PCU

Hardware designations	Designation in PCU Base Software	Configuration
X1	Local Area Connection 2	Preset as a DHCP client for connection to a company network . The IP address is dynamically fetched from the DHCP server at X1.
X2	Local Area Connection	Preset as a SINUMERIK DHCP server for connection to a system network . A fixed IP address is preset at X2: If 192.168.214.241 is free in the system network, it is set. Otherwise you can view the allocated fixed IP address in the Control Panel.

Further information

- Setting the IP address and domain (Page 22)
- *SINUMERIK 840D sl Operator Components and Networking Manual*, Chapter *Operator panels*.

1.2.2 Partitioning of the SSD

The PCU 50.5 has a SSD with 80 GB storage capacity. The SSD is already partitioned and pre-assigned by the PCU base software as follows:

Name	System (C:)	Data (D:)
File system	NTFS	NTFS
Storage capacity	40 GB	Approx. 40 GB
Free capacity	20 GB	Approx. 40 GB
Already occupied by	• Microsoft Windows 7 • PCU Base Software incl. various applications and configuration templates (see Chapter Supplied software (Page 7))	-
Application	Installing user programs	• Customer-specific data • Backup images • Setup packages

1.3 Directory structure and file conventions**Templates for configuration files**

To assist you with parameterization of the system, templates for various configuration files are provided on the PCU 50.5. The templates are organized into various directories, which you will find below the following directory:

- `C:\ProgramData\Siemens\MotionControl\siemens`

These templates are already fully functional when supplied and will be used automatically if you do not create your own configuration files based on these templates.

Note

Do not modify files in the template directory `siemens`!

The `siemens` directory contains only files with the factory settings that can be copied and used as templates.

Never overwrite these files. Instead copy them into another directory (see Section *Your settings*).

Directory hidden in delivery condition

For safety reasons, the `C:\ProgramData` directory is hidden in the delivery condition of the PCU.

To view the directory, proceed in one of the following ways:

- To switch quickly to a hidden directory, type the address in the Windows Explorer.
- To show all hidden directories by default, change the setting in the Control Panel under "Appearance and Personalization > Folder Options > View > Advanced Settings".
- To show only one directory by default, first make the setting to show all hidden directories, then activate the attribute "Hidden" in the properties of the directory and finally again make the setting to not show all hidden directories.

Your settings

Several directories are available into which you can copy and adapt configuration files in order to make your own settings. In decreasing order of priority, these are:

- `C:\ProgramData\Siemens\MotionControl\user\`
- `C:\ProgramData\Siemens\MotionControl\oem\`
- `C:\ProgramData\Siemens\MotionControl\addon\`

The settings in a configuration file in a directory with higher priority always replace those from one of lower priority. Template directory `siemens` has the lowest priority.

To define your own settings, therefore, copy a setting from the relevant template file into a directory with higher priority, e.g. below directory `user`.

First commissioning of the PCU 50.5

2.1 Overview

Once you have installed and networked the PCU 50.5, you must set up a Windows administrator account before the PCU 50.5 can be accessed from a remote TCU. In headless mode, therefore make the configuration in a response file (unattend.xml) before switching on the PCU 50.5 for the first time.

The diagram shows a sample workflow for the first commissioning. Further information is provided in the accompanying legend tables.

Note

Do not switch off the PCU 50.5 Windows 7 hardware!

Loss of data may result on Windows-based systems if the system is not shut down properly before it is disconnected from the power source. For technical reasons, data is still being written to the SSD shortly after shutdown.

To avoid loss of data, use a SITOP UPS module.

See [Configuring the SITOP UPS module for use with the PCU Base Software \(Page 29\)](#).

Example workflow

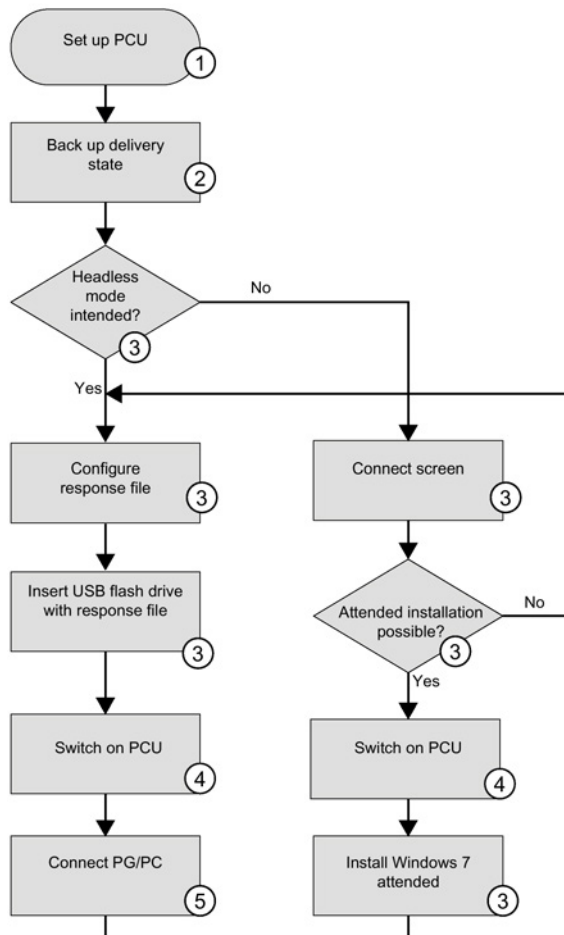


Figure 2-1 Example sequence of first commissioning

No.	Step	Description
1	Set up PCU	Information about installation and mounting of the PCU 50.5 is provided in the <i>SINUMERIK 840D sl Operator Components and Networking Manual</i> , Chapter <i>Operator panels</i> .
2	Back up delivery state	Before starting the actual first commissioning, you can back up the delivery condition of the Windows 7 PCU base software as disk image (Page 13).
3	- Headless mode intended? - Configure response file - Insert USB flash drive with response file - Attended installation possible? - Connect screen - Install Windows 7 attended	As soon as you start with the first commissioning of the PCU base software, you must decide on an installation type: - Configure a response file and complete installation and user settings unattended (Page 15). - Connect screen (Page 24) and complete installation attended (Page 14), at the same time set up users.
4	Switch on PCU	Do not switch on the PCU without the response file (Page 15) if you want the installation to be performed unattended.
5	Connect PG/PC	To be able to access a directory of a PG/PC in the network, for example, connect the PG/PC in the network (Page 81).

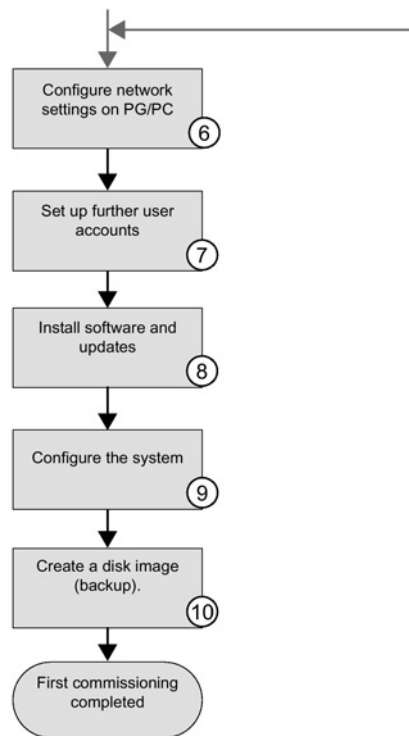


Figure 2-2 Example sequence of first commissioning (continued)

No.	Step	Description
6	Configure network settings on PG/PC	Once you have networked the PCU, a PG/PC and any other devices with one another, configure the network settings of your PG/PC (Page 81).
7	Set up further user accounts	Before you install software, at least one user account with limited rights must be set up (Page 21).
8	Install software and updates	Install necessary automation software and Windows updates (Page 43).
9	Configure the system	Configure the installed software (Page 43), Windows 7 and other system properties (Page 21) and, if necessary, the SITOP UPS module (Page 29).
10	Create a disk image (backup).	Create a disk image of the SSD as a backup (Page 65) to be able to return to this version in a maintenance situation.

2.2 Backing up and restoring the delivery condition

Before starting the actual first commissioning, you can backup the delivery condition of the PCU base software.

Requirement

The first commissioning of the PCU 50.5 (with "unattend.xml" response file or attended) has not been performed yet.

Procedure

To back up the delivery condition of the PCU base software, proceed as follows:

1. Start the Service Center directly after switching on the PCU 50.5 the first time, e.g. via the Windows boot menu.

Starting the Service Center (Page 65)

2. Back up the entire c: partition as disk image at another location, e.g. at
`D:\DeliveryState_PCU-Basesoftware\`

See also: Create a disk image of a partition (Page 74)

Result

The delivery condition is stored as disk image at the selected location. If required, you can move this disk image to another location later.

First create a bootable USB flash drive (Page 80) with which you can restore this disk image (Page 76).

After you have restored the delivery condition of a PCU 50.5 with a disk image, the first commissioning starts at the initial startup with response file or attended.

Further information

- Backing up and restoring data (Page 65)
- First commissioning of the PCU 50.5 (Page 11)

2.3 Install the PCU Base Software attended

If you have connected to the PCU 50.5 directly to the operator panel front or screen, you can complete installation of the PCU Base Software and Windows 7 attended. For this, user entries are required during installation in order to set up the Windows administrator account, for example.

Alternately, you can use unattended installation (Page 15) in which case you define the necessary user entries in an answer file (unattend.xml) before switching on the PCU 50.5.

Precondition

You can connect an operator panel front or screen and keyboard directly to the PCU 50.5.

Procedure

To perform attended installation of the PCU Base Software, proceed as follows:

1. Switch on the PCU 50.5.
The installation wizard guides you through the necessary steps.
2. During installation, create a Windows administrator account. You can create additional user accounts in the Control Panel later. See also: Managing user accounts (Page 21).
3. Complete installation.

Result

Installation of the PCU Base Software with Windows 7 is performed. After attended installation, you must activate the PCU Installer manually (Page 53).

2.4 Unattended installation of the PCU Base Software

You can also install the PCU Base Software unattended.

Use this procedure in the following cases:

- If you are operating a PCU as a headless system (without screen and input devices)
- If you want to start up several PCUs with identical settings

To start unattended installation (Page 18), during initial startup use a USB flash drive with answer file "unattend.xml" in its root directory. You preliminarily define the settings that you would normally have to make as user entries during installation in this answer file.

See also

Microsoft TechNet - Methods for performing Windows setup
(<http://technet.microsoft.com/en-us/library/dd744269.aspx>)

2.4.1 Configuring an answer file

Overview

The PCU base software, Windows 7, is pre-installed on your PCU 50.5, but for security reasons, final installation and the setting up of user accounts must be performed during initial startup.

Configuration using the unattend.xml response file

A response file (`unattend.xml`) allows Microsoft Windows to be installed unattended. In this file, you define the settings in advance, which you must normally enter as user data during installation. Before you switch on the PCU for the first time, copy your adapted `unattend.xml` into the root directory of a USB flash drive and connect this to the PCU 50.5.

Unattended installation is required in the following cases:

- If you are operating a PCU as a headless system (without screen and input devices)
- If you want to start up several PCUs with identical settings

Obligatory settings

You preliminarily define the settings that you would normally have to make as user entries during installation in `unattend.xml`. You must enter at least the following information:

- Name of the administrator account
- Password of the administrator account
- Password hint
- Confirmation of the EULA

Execute the command line commands unattended

The `unattend.xml` response file with the element `/FirstLogonCommands/SynchronousCommand` provides you with the option to automatically execute commands unattended when Windows first runs up. However, to do this, you must also activate the autologon mode with the administrator account.

You can define as many `FirstLogonCommands` as required. You define the sequence in which the commands are executed in the element `/FirstLogonCommands/SynchronousCommand/Order`

In the response file used as example, two different command line commands are executed with `FirstLogonCommands/SynchronousCommand`:

- Directory `D:\Install` released for the administrator account
- Activate the PCU Installer using the response file

Sample response file

Because settings are always user-specific, only a sample response file can be provided as an example:

- Sample answer file for PCU 50.5 Windows 7 in the Service&Support Portal (<http://support.automation.siemens.com/WW/view/en/69154062>)

Adapt this file to your requirements.

Table 2- 1 Mandatory and recommended settings in response file "unattend.xml"

Path in the XML data structure	Purpose	Example
unattend/ settings/ component/ AutoLogon	Modify the user account for the autologon mode (optional). If you want to automatically execute command line commands (<i>FirstLogonCommands</i>) after the unattended installation of Windows, activate the autologon mode in the <i>unattend.xml</i>. Enter the password and user name of an administrator account that you also want to create with this response file. <i>FirstLogonCommands</i> are only automatically executed when the administrator account is active.	<pre><AutoLogon> <Password> <Value>SUNRISE</Value> </Password> <Enabled>true</Enabled> <LogonCount>1</LogonCount> <Username>auduser</Username> </AutoLogon></pre>
unattend/ settings/ component/ OOBE/ HideEULAPage	Confirm EULA (obligatory) Set the value to <i>true</i> to automatically confirm EULA.	<pre><OOBE> <HideEULAPage>true</HideEULAPage> </OOBE></pre>
unattend/ settings/ component/ UserAccounts/ LocalAccounts/ LocalAccount	Create the administrator account (obligatory) Enter a user name in the <i>Name</i> element, and a secure password in the <i>Password/Value</i> element. Specify the administrator status in the <i>Group</i> element. The <i>DisplayName</i> and <i>Description</i> elements are optional. In order to create additional user accounts, insert the complete <i>LocalAccount</i> section a second time and there, change the values of <i>Name</i>, <i>Password/Value</i> and, if required, <i>Group</i>.	<pre><LocalAccount wcm:action="add"> <Name>auduser</Name> <DisplayName>auduser</DisplayName> <Description>First admin account</Description> <Group>Administrators</Group> <Password> <Value>SUNRISE</Value> <PlainText>true</PlainText> </Password> </LocalAccount></pre>

Path in the XML data structure	Purpose	Example
unattend/ settings/ component/ FirstLogonCommands/ SynchronousCommand	Activate PCU Installer (optional) In order that the PCU base software component PCU Installer searches for setup files directly after logging on, add this command. As result, a setting is made in the ini file of the PCU Installer. So that you can store setup files in the D:\Install default installation directory, you must release this for the administrator account in the unattend.xml (see above). For further information, see Chapter PCU Installer (Page 52).	<pre><SynchronousCommand wcm:action="add"> <CommandLine>"C:\Program Files (x86)\Siemens\MotionControl\Siemens\System\etc\w rite ini entry.exe" "C:\ProgramData\Siemens\MotionControl\user\Syste m\etc\PCUInst.ini" processing_loginphase Start- State activated ENH</CommandLine> <Description>Set PCU Installer to start at user login</Description> <Order>1</Order> </SynchronousCommand></pre>
unattend/ settings/ component/ FirstLogonCommands/ SynchronousCommand	Release the default installation directory (optional) Use this sample code to release directory D:\Install for the administrator code. The sample code sets the access rights for D:\Install to FULL (read and write access) for the "auduser" user account. Instead of "auduser", enter the name of your administrator account.	<pre><SynchronousCommand wcm:action="add"> <CommandLine>net share Install=D:\Install /GRANT:auduser,FULL /REMARK:Installshare</CommandLine> <Description>Share the local built-in installa- tion folder</Description> <Order>2</Order> </SynchronousCommand></pre>

Optionally, you can configure additional settings such as additional user accounts or network settings.

Further information

You will find information about all options for adapting unattend.xml on the Microsoft website.

- Microsoft TechNet - Methods for performing Windows setup (<http://technet.microsoft.com/en-us/library/dd744269.aspx>)
- Microsoft TechNet - Unattended Windows Setup Reference (<http://technet.microsoft.com/en-us/library/ff699026.aspx>)

2.4.2 Installing the PCU Base Software attended

If you want to operate a PCU as a headless system or start up several PCUs with identical settings, use unattended installation.

Requirement

- The PCU is in the delivery condition and switched off.
- In BIOS, the startup of the USB interfaces is deactivated.

In the PCU50.5, it is deactivated via a factory setting.

- A USB flash drive is available (e.g. SIMATIC PC USB-FlashDrive).
The USB flash drive must not use encryption, or it must be deactivated.
- The "unattend.xml" response file has been adjusted to your settings and at least one administrator account with password is defined.
See: Configuring an answer file (Page 15).

Procedure

Proceed as follows to start unattended installation:

1. Copy your adjusted "unattend.xml" response file directly into the root directory of the USB flash drive.
2. Connect the USB flash drive to the PCU.
3. Switch on the PCU.

Result

The first startup is performed as an unattended installation, using the settings that you specified in the response file.

Configuration of the system

3.1 Overview

Once you have completed the first commissioning, you can adapt the configuration of the system:

- For security reasons, set up user accounts (Page 21) with limited rights, e.g. for auto logon mode.
- You can view or change (Page 22) the name of the PC system.
- The network settings (Page 22) in the delivery condition have been adapted for use with SINUMERIK solution line and usually do not have to be changed.
- You can deactivate the USB interfaces (Page 23) to prevent malicious software from entering the system network via this route.
- You can set up an external screen (Page 24) or set the screen resolution (Page 25) and color depth (Page 28).
- If you use a SITOP UPS module for the uninterruptible power supply or plan to use one, configure it for use with the PCU base software (Page 29).

Other configuration options (e.g. Desktop background) are available directly from the Service Desktop and can be implemented by the known Windows 7 methods:

- To call up the Windows help, click "Help and Support" in the Start menu.
- You will find help for Windows 7 in the Internet on the Microsoft website: Microsoft Windows support
(<http://windows.microsoft.com/en-US/windows/windows-help#windows=windows-7>)

3.2 Managing user accounts

Overview

During initial commissioning, you must create a local Windows administrator account and define a password.

3.3 Changing the name of the PC system

You can set up and manage additional Windows user accounts in the Control Panel. To do that, log on with administrator rights.

Note

Administrator account cannot be recovered

If you have forgotten the password of the administrator account there is no way of recovering it!

However, after initial commissioning, you can create a file with which you can later assign a new password. To do that, in the Control Panel under user accounts use the "Create a password reset disc" command.

See also: Microsoft Windows support - creating a password reset disk

(<http://windows.microsoft.com/en-US/windows7/Create-a-password-reset-disk>)

Further information about user administration under Windows 7

You will find information about user administration under Windows 7 in the Windows help.

- To call up the Windows help, click "Help and Support" in the Start menu.
- You will find help for Windows 7 in the Internet on the Microsoft website: Microsoft Windows support
(<http://windows.microsoft.com/en-US/windows/windows-help#windows=windows-7>)

3.3 Changing the name of the PC system

A unique computer name must be set after installation of the PCU base software. You can view and change it in the Control Panel of the Service Desktop.

You will find the computer name in the Control Panel under Category "System and Security", Section "Computer name, domain, and workgroup settings",

3.4 Setting the IP address and domain

The network settings in the delivery condition have been adapted for use with SINUMERIK solution line and usually do not have to be changed.

See also: Network settings (Page 8)

If you want to distinguish your network configuration from the factory settings, you can adapt them in the Windows Control Panel.

You will find these settings in the following menu:

- "Start > Control Panel > Network and Sharing Center > View network connections"

Further information

You can adapt the network settings for backing up and restoring disk images in the configuration of the Service Center (Page 39).

You will find information about network settings under Windows 7 in the Windows help.

- To call up the Windows help, click "Help and Support" in the Start menu.
- You will find help for Windows 7 in the Internet on the Microsoft website: Microsoft Windows support
(<http://windows.microsoft.com/en-US/windows/windows-help#windows=windows-7>)

3.5 Configuring USB interfaces

The USB interfaces are activated in the as-delivered state of the PC system.

To prevent malicious software from entering the control or the system network via the USB interfaces, you can deactivate the USB interfaces.

You can configure the USB interfaces on the Service Desktop via a command in the "Run" dialog box or at the prompt.

Procedure

To activate or deactivate the USB interfaces, enter the relevant command on the Service Desktop in the "Run" dialog box or at the prompt:

Table 3- 1 Disabling USB interfaces with the "sc_usb disable" command

Factory setting:	The USB interfaces are enabled in the delivery condition.
Syntax:	sc_usb disable [-minutes] [all] [HOSTS...]
Description:	• Without a time indication [-minutes], a permanent disable is set. With time indication, the disable takes the time set in minutes. • The disable may refer to all network nodes of the system network, or a list of host names or IP addresses may be specified to which the disable should apply.

Table 3- 2 Activating USB interfaces with the "sc_usb enable" command

Factory setting:	The USB interfaces are enabled in the delivery condition.
Syntax:	sc_usb enable
Description:	With "sc_usb enable" USB storage units on permanently disabled USB interfaces or on certain host names or IP addresses are enabled again.

Further information

You can find general information about a prompt on the Microsoft website.

- Prompt: Frequently asked questions
(windows.microsoft.com/en-us/windows/command-prompt-faq)

3.6 Configuring the keyboard layout

It is assumed that the English keyboard layout is used in SINUMERIK Operate and for the SINUMERIK operator panel fronts.

If a different keyboard layout is set in Windows, this should therefore be changed before commissioning the SIEMENS PC system or before using SINUMERIK Operate.

The setting of the keyboard layout as well as the display of the active keyboard layout is shown in the Windows task bar.

Further information is available at: Microsoft Windows support: Language bar (overview) (<http://windows.microsoft.com/en-us/windows/language-bar-overview#1TC=windows-7>)

3.7 Setting up an external screen

You can connect and set up an external screen to the PCU. The procedure follows the usual procedure under Windows 7.

Requirements

- The PC system is switched off.
- The external screen is connected to the DVI interface of the PC system (with an adapter, if necessary).

Procedure

To use an external screen for the Service Desktop or service system, proceed as follows:

1. Let the PC system boot in the Service desktop.
2. Right-click on the Desktop, then select "Screen Resolution" from the shortcut menu.
3. In the "Screen Resolution" dialog box, select the screen and set the resolution:
 - If you are using an LCD or LED screen, you should ideally set the native resolution of your screen.
 - If you are using an older CRT screen, you can select any resolution.
4. Click "OK" to confirm the settings.

Result

The external screen has been connected and can be used.

3.8 Setting the screen resolution

You can set the screen resolution directly in Windows 7 or use the PCU base software specific file `tcu.ini`

The settings in `tcu.ini` overwrite the Windows system settings. Compared with Windows, they offer additional functions for switching over the resolution of the PC system depending on, for example, the screen connected during run-up.

- Information about the configuration file `tcu.ini` is provided in the following chapter (Page 25).
- Information for setting the screen resolution under Windows 7 is provided on the Microsoft website: Changing the screen resolution (<http://windows.microsoft.com/en-us/windows7/change-your-screen-resolution>)

3.9 Set the resolution in the `tcu.ini`

Overview

The system behavior during run-up for the screen resolution is set in the `tcu.ini` file in Section [VNCServer].

You will find a template of the `tcu.ini` under `C:\ProgramData\Siemens\Motion Control\siemens\System\etc\`

Do not overwrite this template, but rather create your own `tcu.ini` inside one of the user directories (see Chapter Directory structure and file conventions (Page 9)).

Further information

- Chapter Directory structure and file conventions (Page 9).
- Operator Components Manual, Chapter "Networking"

Set the resolution during run-up of the PCU

The following options are available for selection in the `tcu.ini` file in the [VNCServer] section (below the `# RESOLUTION` comment):

```
0 = SYSTEM
1 = AUTO_OP_1 (default setting)
2 = AUTO_OP_2
3 = AUTO_MON_1
4 = AUTO_MON_2
5 = 640x480
6 = 800x600
7 = 1024x768
8 = 1280x1024
```

3.9 Set the resolution in the *tcu.ini*

9 = 800x480
 10 = 1280x800
 11 = 1366x768
 12 = 1920x1080

The meaning of the settings are as follows:

Table 3- 3 Settings in the configuration file *tcu.ini*

Settings	Meaning
SYSTEM	The resolution is not specially set during run-up. i.e. the resolution last used in the system is active, e.g. the resolution which had been set manually in the Control Panel.
AUTO_OP_1	Default setting: During run-up, the resolution is automatically set ("PCU panel" has priority) in accordance with the following scenarios:
Example 1:	There is a PCU panel (irrespective of whether there is a PCU monitor and TCU panels): [In active TCU mode: The following additional condition applies: PCU is activated (with dynamic resolution switching enabled)]: The resolution is set to the max. resolution of the PCU panel (max. 1920x1080).
Example 2:	There is no PCU panel, however there is a PCU monitor (irrespective of whether there are any TCU panels): [In active TCU mode: The following additional condition applies: PCU is activated (with dynamic resolution switching enabled)]: The resolution is not specially set; i.e. the resolution last used in the system is active, e.g. the resolution which had been set manually in the Control Panel. (Different to AUTO_OP_2 !)
AUTO_OP_2	Like AUTO_OP_1, except: Example 2: There is no PCU panel, however there is a PCU monitor (irrespective of whether there are any TCU panels): [In active TCU mode: The following additional condition applies: PCU is activated (with dynamic resolution switching enabled)]: The resolution is set to the max. resolution of the PCU monitor, reduced to the next lowest SINUMERIK resolution. The SINUMERIK resolutions are 640x480, 800x600, 1024x768, 1280x1024, 800x480, 1280x800, 1366x768 and 1920x1080. Example: In the case of a PCU monitor with a max. resolution of 2048x1152, the SINUMERIK resolution setting is 1920x1080.
AUTO_MON_1	During run-up, the resolution is automatically set ("PCU monitor" has priority) in accordance with the following scenarios:

Settings	Meaning
Example 1:	There is a PCU monitor (irrespective of whether there is a PCU panel and TCU panels): [In active TCU mode: The following additional condition applies: PCU is activated (with dynamic resolution switching enabled)]: The resolution is set to the max. resolution of the PCU monitor, reduced to the next lowest SINUMERIK resolution. The SINUMERIK resolutions are 640x480, 800x600, 1024x768, 1280x1024, 800x480, 1280x800, 1366x768 and 1920x1080. Example: In the case of a PCU monitor with a max. resolution of 2048x1152, the SINUMERIK resolution setting is 1920x1080. If there is a PCU panel, the display there is panned if the max. resolution of the PCU panel is lower than the max. resolution of the PCU monitor.
Example 2:	There is no PCU monitor, however there is a PCU panel (irrespective of whether there are any TCU panels): [In active TCU mode: The following additional condition applies: PCU is activated (with dynamic resolution switching enabled)]: The resolution is not specially set; i.e. the resolution last used in the system is active, e.g. the resolution which had been set manually in the Control Panel. (Different to AUTO_MON_2 !)
Example 3:	There is no PCU monitor and no PCU panel (headless operation): A) In active TCU mode: a) Dynamic resolution switching is enabled (resolution adaptation entry in TCU.ini) and at least one TCU is already logged on: The resolution is set to the resolution of the TCU which is currently active. b) Dynamic resolution switching is enabled (resolution adaptation entry in tcu.ini) and no TCU has logged on yet or dynamic resolution switching is disabled: The resolution is set to the max. resolution of the current PCU panel, i.e. which logged on during the previous session. Default setting: Default TCU resolution in accordance with the registry. Notice: The first TCU panel to log on is activated. The focus handler then automatically sets the resolution to this TCU panel's resolution (in the case of dynamic resolution switching). B) In inactive TCU mode: The resolution is not specially set - i.e. the resolution used during the previous session in the system is active, e.g. the resolution set manually in Control Panel.
AUTO_MON_2	Like AUTO_MON_1, except: Example 2: There is no PCU monitor, however there is a PCU panel (irrespective of whether there are any TCU panels): [In active TCU mode: The following additional condition applies: PCU is activated (with dynamic resolution switching enabled)]: The resolution is set to the max. resolution of the PCU panel (max. 1920x1080).

Settings	Meaning
640x480	During run-up, the SINUMERIK resolution is set to 640x480.
800x600	During run-up, the SINUMERIK resolution is set to 800x600.
1024x768	During run-up, the SINUMERIK resolution is set to 1024x768.
1280x1024	During run-up, the SINUMERIK resolution is set to 1280x1024.
800x480	During run-up, the SINUMERIK resolution is set to 800x480.
1280x800	During run-up, the SINUMERIK resolution is set to 1280x800.
1366x768	During run-up, the SINUMERIK resolution is set to 1366x768.
1920x1080	During run-up, the SINUMERIK resolution is set to 1920x1080.

3.10 Setting the color depth in the `tcu.ini` file

Per default, the color depth of the SIEMENS PC system is set to 16-bit via the `tcu.ini` configuration file and is reset to this value at every run-up.

To use a different color depth, you must adapt the setting in the `tcu.ini` file in the [VNCServer] section.

Requirement

- Hidden directories are not shown
- A copy of the `tcu.ini` configuration file is stored in the `user` directory:
`C:\ProgramData\Siemens\MotionControl\user\System\etc\tcu.ini`
- The `tcu.ini` configuration file in the `user` directory is not write-protected

See: Directory structure and file conventions (Page 9)

Procedure

To adapt the color depth of the PC system, proceed as follows:

1. Open the `tcu.ini` configuration file in the
`C:\ProgramData\Siemens\MotionControl\user\System\etc\` directory.
2. Enter the required value in the [VNCServer] section (below the `#COLOR DEPTH` comment):
 - `ColorDepth=0`
 The color depth is not changed during run-up, but rather the color depth used last is retained. In this way, you can adapt the color depth in the Windows settings.
 - `ColorDepth=1` (default value)
 The color depth is set to 16-bit during run-up.
 - `ColorDepth=2`
 The color depth is set to 32-bit during run-up.
3. Save and close the file.

3.11 Configuring the SITOP UPS module for use with the PCU Base Software

3.11.1 Overview of SITOP UPS

SITOP UPS modules can continue operation of the SIEMENS PC system temporarily if a power failure occurs and/or shut down the system correctly.

For example, a SITOP UPS module also protects against data loss when the SIEMENS PC system hardware is shut down, because for technical reasons with Windows-based systems, data is still written to the SSD after shutdown.

- You will find general information about SITOP UPS modules used in combination with the PCU base software in Chapter SITOP modules for the PCU 50.5 (Page 30).
- If you want to use a SITOP UPS module in combination with the PCU base software, you must first adapt the settings of the SITOP software (Page 31).
- Optionally, you can adapt the delay time after which the HMI software will be forced to shut down (Page 35).
- You must configure the hardware (Page 36) of the SITOP UPS module before using it with the PCU base software.

NOTICE

Ensure the readiness for operation of the SITOP power supply

In order to prevent data loss, the SITOP power supply must be ready for operation. Note the following:

- The SITOP power supply may not be ready for operation immediately after being switched on (a capacitor-buffered UPS500 requires 1-2 minutes make time, for example).
- The operating system and the SITOP software must have run up so that the SITOP software can shut down the system correctly when a power failure occurs.
- The readiness for operation and the functional capability of the SITOP power supply must be ensured (e.g. battery of capacitor charged).
- Please also note all the information on the readiness for operation in the documentation of your SITOP device.

3.11 Configuring the SITOP UPS module for use with the PCU Base Software

Further information

Information on the use of SITOP UPS modules in combination with the PCU base software can also be found on the Internet:

- Service & Support Portal: SINUMERIK PCU 50.5: Use of an uninterruptible power supply (<http://support.automation.siemens.com/WW/view/en/90142681>)
- Service & Support Portal: Typical values for an operator station with PCU 50.5 Windows 7 (<http://support.automation.siemens.com/WW/view/en/76773241>)
- SITOP Selection Tool (<http://www.siemens.com/sitop-selection-tool>) (Tool for the selection of a SITOP UPS module)

3.11.2 SITOP modules for the PCU 50.5

Suitable SITOP UPS modules

SITOP UPS modules can continue operation of a PC system temporarily if a power failure occurs and/or shut down the system correctly.

In conjunction with the PCU base software, all specific applications are taken into account (e.g. HMI software) with the `USVShutdown.bat` component.

Safe shutdown after a power failure of the PCU 50.5 as delivered takes approximately 20 seconds. For a typical power consumption of approx. 60 W, a capacitor-buffered UPS with an energy storage device of 2.5 kW is recommended for a PCU 50.5. If longer buffer times are required, then several expansion modules can be connected in a cascade connection.

Example of a compatible SITOP UPS module:

SITOP 500S (15 A / 2.5 kW capacitor)

Article no.: 6EP1933-2EC41

Note

Information about the test environment for machine OEMs

The functionality has been tested in the standard configuration with the PCU base software.

When installing add-on or OEM software components, you must check the shutdown procedure of the entire system and adapt the factory setting (180 seconds).

See: Parameterizing a delay time for quitting the HMI software (Page 35)

Further information

Information about available SITOP UPS modules is given in the catalog or in the Internet:

- Service & Support Portal: SINUMERIK PCU 50.5: Use of an uninterruptible power supply (<http://support.automation.siemens.com/WW/view/en/90142681>)
- Service & Support Portal: Typical values for an operator station with PCU 50.5 Windows 7 (<http://support.automation.siemens.com/WW/view/en/76773241>)

- SITOP Selection Tool (<http://www.siemens.com/sitop-selection-tool>) (Tool for the selection of a SITOP UPS module)
- Catalog KT 10.1 Power supply SITOP
- SIEMENS Industry Mall > 24 V DC Uninterruptible Power Supplies (<https://eb.automation.siemens.com/mall/en/US/Catalog/Products/7010117>)

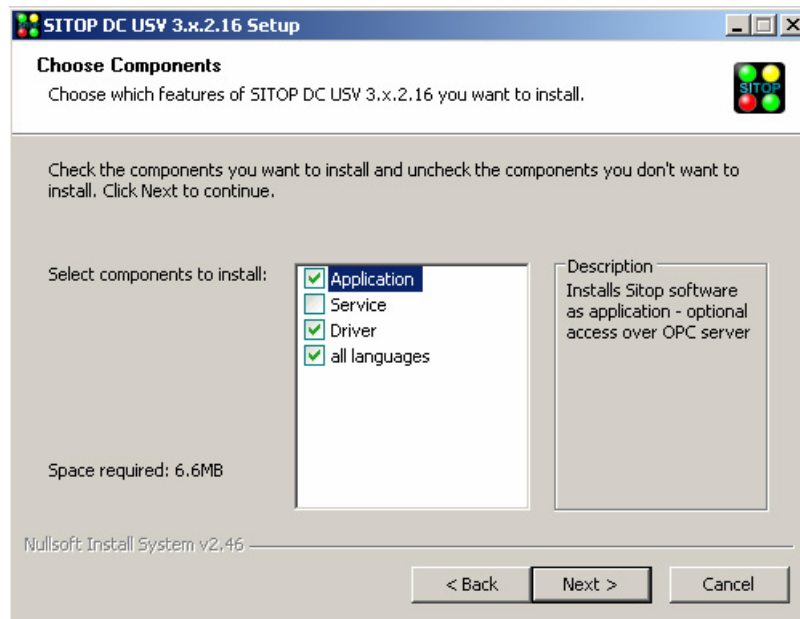
3.11.3 Configuring SITOP software for PCU Base Software

If you want to use a SITOP UPS module in combination with the PCU base software, you must first adapt the settings of the SITOP software.

Requirement

- The SITOP UPS hardware is connected via the USB interface.
- The SITOP software of version 3.x.2.16 or later is installed in directory `C:\Program Files (x86)\SITOP\`

- If you want to use SITOP software of version 3.2.1.16 or older, this must be installed as a normal application, not as a service.



Note

Windows service of the SITOP software version 3.2.1.16 or older is not compatible with the PCU base software

If you use SITOP software of version 3.2.1.16 or older as a Windows service with the PCU base software, the correct shutdown procedure cannot be performed!

If you have installed the SITOP software of version 3.2.1.16 as a Windows service, you must uninstall this and install it again as an application in order to be able to use it in conjunction with the PCU base software.

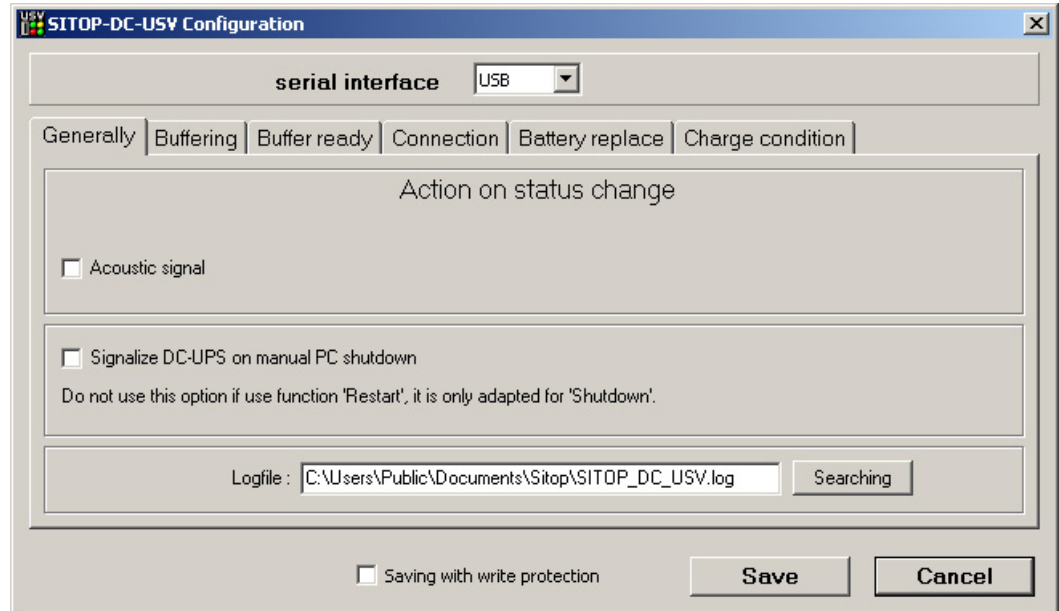
As of version 3.2.1.17, the SITOP software can be used as service with the PCU base software and is recommended.

- The SITOP software was configured during the installation so that it starts automatically when Windows is run-up (factory setting).
- The Service Desktop is active.

Procedure

Proceed as follows to configure the SITOP software for use with the PCU base software:

1. Call the settings of the SITOP software, for example, by right-clicking the SITOP UPS icon in the Windows information area and then selecting "Configuration". The "SITOP DC UPS Configuration" dialog box opens.



2. In the "General" tab, select "UPS" in the drop-down list at "Serial interface".

3. Switch to the "Floating operation" tab and make the following settings:

- Deactivate the "Display monitoring window after" checkbox

Note

Deactivating the monitoring window

Display of the monitoring window can cause malfunctioning of the HMI software.

- Click directly in the text field at "Start application after" and specify the path of the

USVShutdown.bat:

C:\Program Files

(x86)\Siemens\MotionControl\siemens\sinumerik\hmi\base\USVShutdown.bat

Note

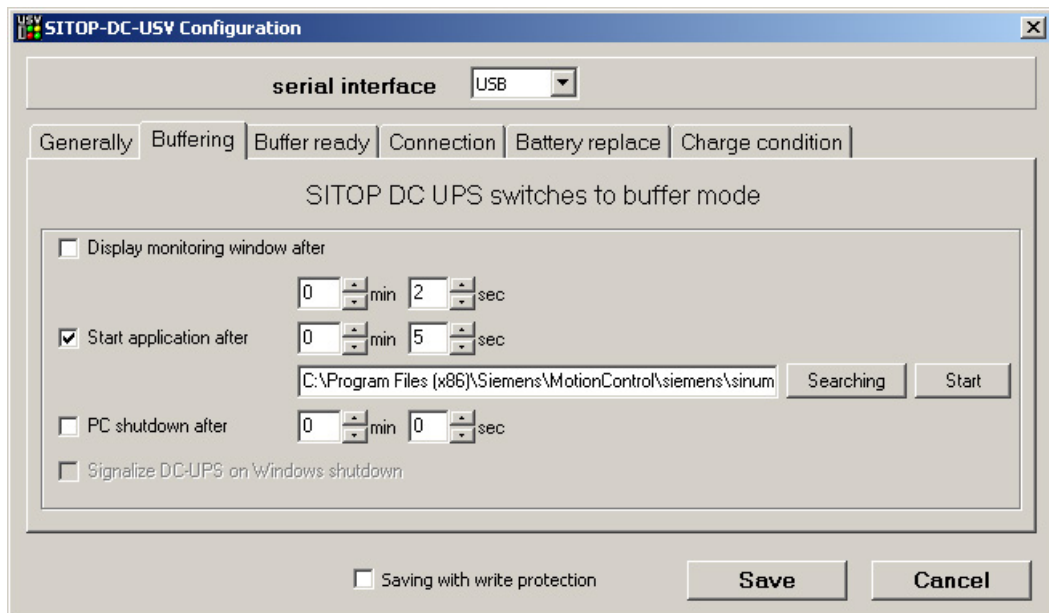
Do not use the "Browse" button

The dialog can crash if you use the "Browse" button to specify the path of the

USVShutdown.bat.

Instead, enter the path directly in the text field.

- After you have specified the path of the USVShutdown.bat, activate the "Start application after" checkbox. Optionally, you can specify the wait time after which the USVShutdown.bat is to be executed (e.g. 5 seconds).



4. Confirm the settings with "Save".

3.11.4 Parameterizing a delay time for quitting the HMI software

Before the SITOP monitor closes Windows 7, `USVShutdown.bat` shuts down operation of the HMI software.

If the HMI software cannot be shut down within 180 seconds (factory setting) due to an error, HMI software shutdown is forced and Windows 7 is shut down.

If the HMI software of your OEM installation is not shut down within 180 seconds, you can parameterize this wait time manually.

Note

Information about the test environment for machine OEMs

The "SITOP UPS" function has been tested in the standard configuration with the PCU base software. When installing add-on or OEM software components, you must check the shutdown procedure of the entire system and adapt the factory setting.

Requirement

- The SITOP UPS module is configured for use with the PCU base software. See Configuring SITOP software for PCU Base Software (Page 31).
- The Service Desktop is active.

Procedure

Proceed as follows to change the delay time:

1. Call the settings of the SITOP software, for example, by right-clicking the SITOP UPS icon in the information area and then selecting "Configuration".
2. In the "SITOP DC UPS Configuration" dialog box, enter the wait time in seconds as a command line parameter in the "Floating operation" tab:
 - **Syntax:** <Path>\USVShutdown.bat -<Time in seconds>
 - **Example:** C:\Program Files
(x86)\Siemens\MotionControl\siemens\simumerik\hmi\base\USVShutdown.bat -180

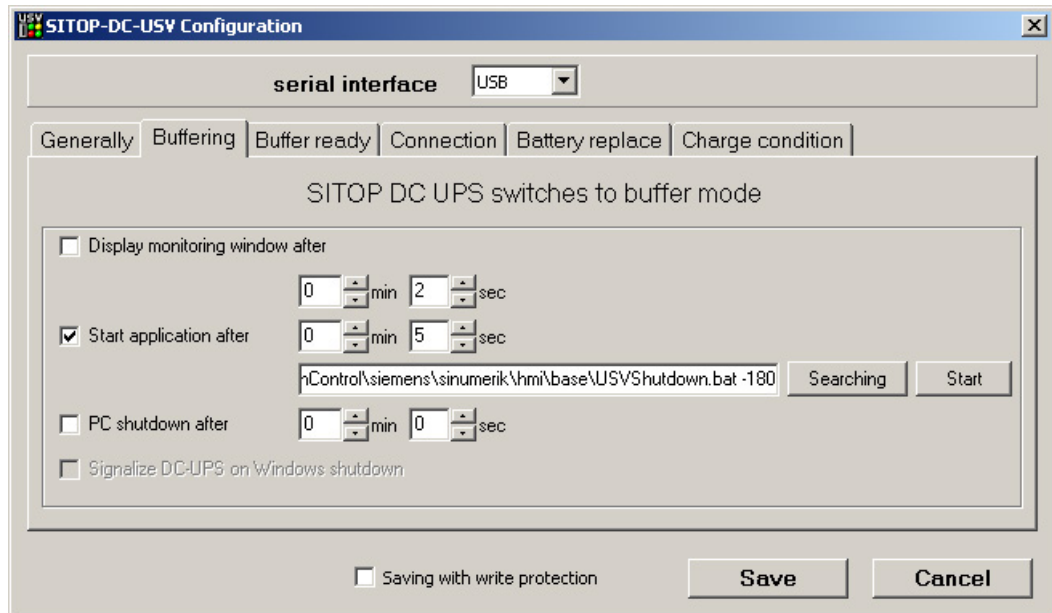


Figure 3-1 SITOP Monitor Floating operation

3. Confirm the setting with "Save".

3.11.5 Hardware configuration of the SITOP UPS module

Buffering parameterization

The UPS module can be used to select whether buffering should be completed after a predetermined period of time or not until the accumulator's lowest discharge threshold (= maximum buffer time) has been reached. Both buffering parameterizations result from this.

"Maximum buffer time" mode

This mode enables the system to be shut down in a time-optimized manner. The UPS module is synchronized with the shutdown of the operating system. Buffering is maintained until the operating system has been shut down. The operating system must shut down within a maximum of five minutes (including all applications). Otherwise, the UPS module buffers for the maximum buffer time (dependent on the accumulator state).

Required settings on the UPS module (USB interface)

	On - Off			
1			+2V	Cut-in threshold +22V fixed
2			+1V	
3			+0.5V	
4			+1V	End-of-charge voltage + 26.3V fixed
5			+1V	
6			+0.5V	
7			+0.2V	
8			+0.2V	
9			+0.1V	
10			0.35A / 0.7A	Charging current

	On - Off			
1				Set time/max. time
2			+320 s	Buffer time +5 s fixed
3			+160s	
4			+80s	
5			+40s	
6			+20s	
7			+10s	
8				Disconnection
9				Battery operating state on/off

Legend:



Delivery condition setting



Setting for operation on the PCU

“Fixed buffer time” mode

In this mode, the UPS module always buffers for the pre-selected, fixed period of time. It is not possible to synchronize the UPS module with the operating system shutdown.

Required settings on the UPS module

	On - Off			
1			+2V	Cut-in threshold +22V fixed
2			+1V	
3			+0.5V	
4			+1V	◦ ◦ End-of-charge voltage + 26.3V fixed
5			+1V	
6			+0.5V	
7			+0.2V	
8			+0.2V	
9			+0.1V	
10			0.35A / 0.7A	Charging current

	On - Off			
1				Set time/max. time
2			+320 s	° ° Buffer time +5 s fixed
3			+160s	
4			+80s	
5			+40s	
6			+20s	
7			+10s	
8				Disconnection
9				Battery operating state on/off

Legend:



Delivery condition setting



Setting for operation on the PCU

3.12 Configuration of the Service Center

3.12.1 Overview

You can permanently adapt the settings for the backup and restoration of disk images (Service Center (Page 65) component) in the "Service Center Backup/Restore" dialog box:

- Configuration of the network adapter (Page 40)
- Configuration of the host (Page 42)

Adaptations that you make here are stored in the `servicesystem.ini` configuration file.

You can also adapt the settings directly in the associated configuration files:

- `servicesystem.ini`
- Additional settings: `ghost.ini`

You can also view the log files of the Service Center or restart the PC system in the Service Center in the "Service Center Backup/Restore" dialog box.

To open the dialog box, click the ServiceCenter Backup-Restore icon on the Desktop.

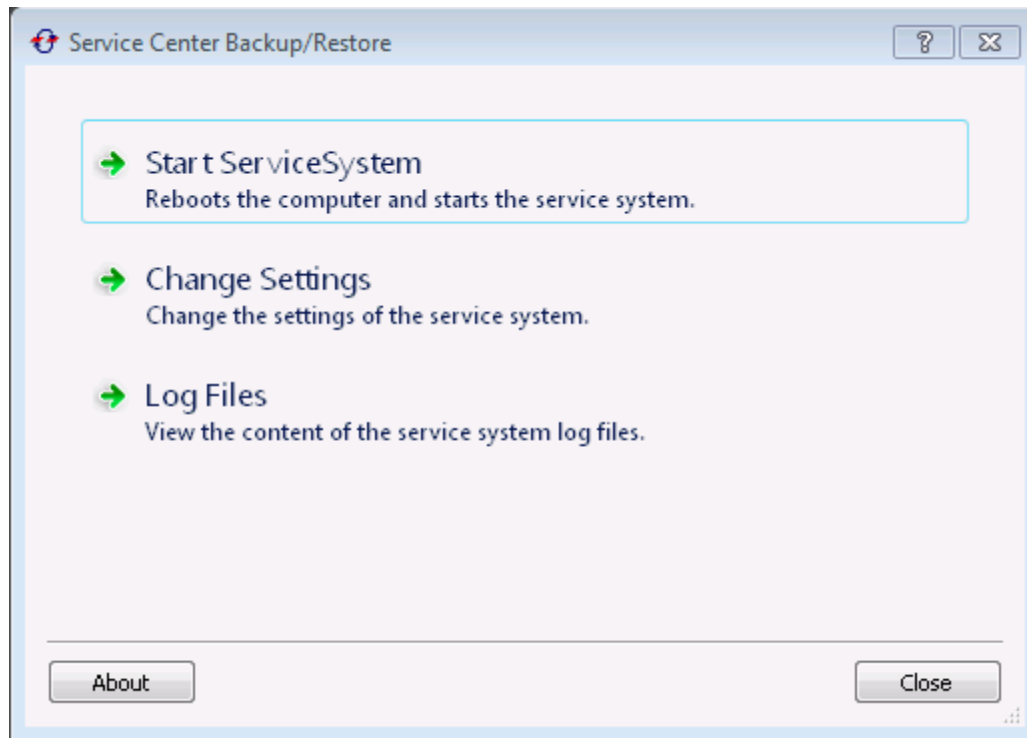


Figure 3-2 ServiceCenter Backup/Restore

3.12.2 Configuration of the network adapter

You can make the network settings in the "Adapter Settings" tab and permanently save this configuration, in contrast to the settings in the "Network Settings" dialog box of the Service Center (Page 67).

Settings that you make in this dialog box are stored in the `servicesystem.ini` configuration file.

Overview

You can make the following IP settings in the "Adapter Settings" dialog box:

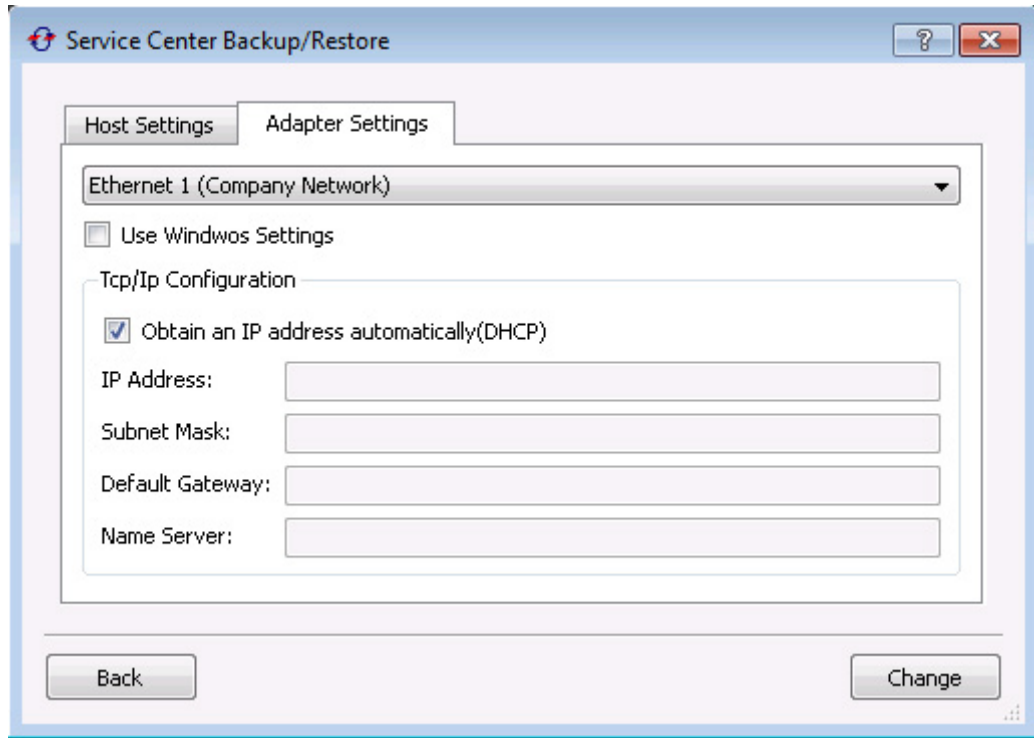


Figure 3-3 Adapter settings - Ethernet 1 (company network)

Table 3- 4 Settings in the "Adapter Settings" dialog box

Settings	Purpose
Adapter	Select the Ethernet interface that you want to configure. All settings in this dialog box refer to the selected Ethernet interface.
Use Windows Settings	Select whether the values set in the Control Panel are to be used.
Obtain an IP address automatically (DHCP)	Select whether the addresses are to be fetched dynamically from the DHCP server. If the checkbox is deactivated, you must make the settings manually.
IP Address	Set the IP address of the NCU. You can specify an IP address from the following range: 192.168.214.250 - 254
Subnet Mask	Specify a subnet mask, e.g. 255.255.255.0
Default Gateway	Set the IP address of the standard gateway. A standard gateway creates a standard route in the IP routing table for all destinations that are not in the subnet.
Name Server	Set the IP address of the name server. A name server answers questions asked about a domain name zone using a DNS database.

3.12.3 Configuration of the host

You can make the network settings in the "Host Settings" tab and permanently save this configuration, in contrast to the settings in the "Network Settings" dialog box of the Service Center.

Settings that you make in this dialog box are stored in the `servicesystem.ini` configuration file.

Overview

You can make the following settings in the "Network Settings" dialog box:

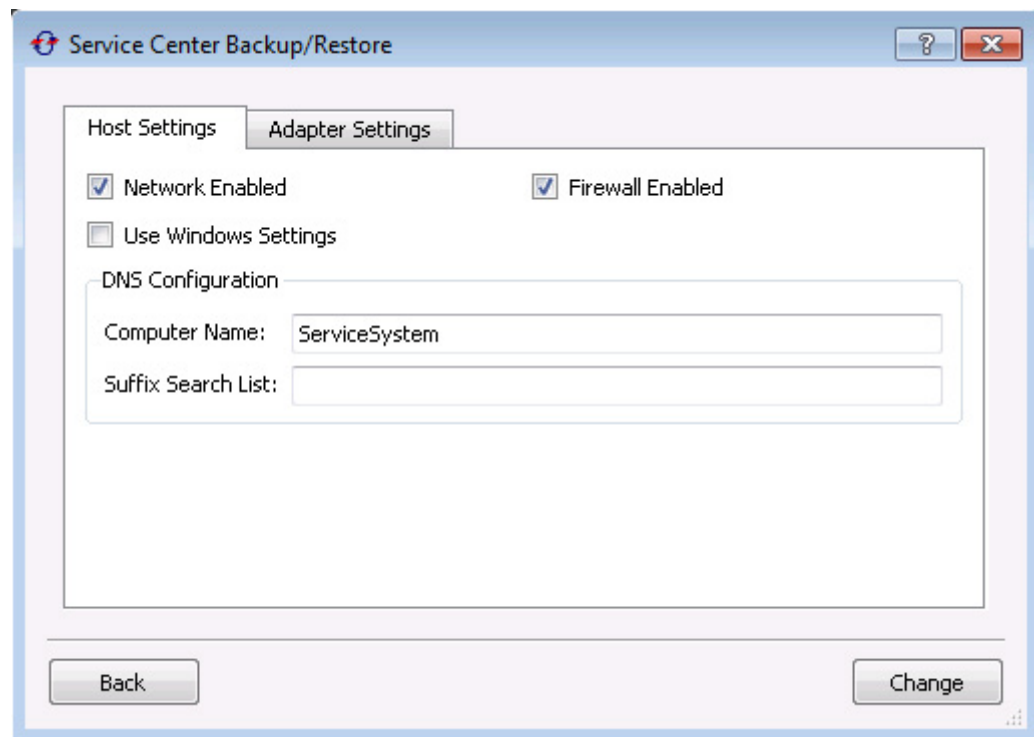


Figure 3-4 Host settings

Table 3- 5 IP settings in the "Host Settings" dialog box

Settings	Purpose
Network enabled	Activate or deactivate the network interfaces of the PC system.
Firewall enabled	Activate or deactivate the firewall of the PC system.
Use Windows Settings	Select whether the values set in the Control Panel are to be used.
DNS Configuration > Computer Name	Specify the DNS computer name.
DNS Configuration > Suffix Search List	Parameterize a search list for DNS suffix, e.g. "network.com". The factory setting of the Ethernet interface "Local Area Connection" is ".local".

Installing and configuring updates and automation software

4

4.1 Overview

Once you have completed installation of the PCU Base Software Windows 7 and possibly configured the network settings, you can set up handling of the security updates, patches, and software installations or install them directly.

- You will find information about Windows updates and the network security of an automation system in Chapter Patch management and security updates (Page 43).
- The PCU Installer (Page 52) will assist you with automatic installation of setup packages.
- You can install SINUMERIK Operate (Page 44) or other HMI software and at the same time (Page 44) or subsequently (Page 45) set it up for autologon mode.
- To prevent operation of the HMI software from being interrupted by key combinations such as CTRL+ALT+DEL, you can activate and configure the key filter (Page 47).
- You can install STEP 7 and the SINUMERIK add-on (Page 50) and add a softkey to the HMI software to start STEP 7 (Page 51).
- If you want to transfer existing data to the newly installed automation software, read Chapter Migration (Page 63).

4.2 Patch management and security updates

Overview

Microsoft regularly eliminates security holes in Windows and makes these corrections available on its "Windows Update" website.

As a solution for managing and making Windows updates available in a network, Microsoft offers the component Windows Server Update Services (WSUS).

Microsoft TechNet - Windows Server Update Services
(<http://technet.microsoft.com/en-us/wsus/default.aspx>)

Industrial security

You will find information about network security for an automation system and a comprehensive protection concept under Industrial Security on the SIEMENS website:

SIEMENS Industrial Security
(<http://www.industry.siemens.com/topics/global/en/industrial-security/>)

Compatibility with automation software

Information about compatibility of a Windows update with specific application software of SIEMENS may possibly be available on the product pages of the application software in question in the Service&Support portal:

SIEMENS Industry Service & Support Portal
(<http://support.automation.siemens.com/WW/view/en/4000024>)

Generally, it is recommended that you check the compatibility of updates in a dedicated project test environment before installing it in a productive environment.

4.3 Setting up SINUMERIK Operate for autologon mode

If you want the PC system to start automatically in SINUMERIK Operate after switching on without having to make any further entries, set up the Autologon mode.

During installation of SINUMERIK Operate, you can define an existing user account as an autologon account. The PC system then starts without the password for this user account being entered. SINUMERIK Operate is then started automatically because a link is created in the "Startup" Autostart folder.

For security reasons, you should use a special user account without administrator rights.

You can also configure the key filter and deactivate certain keyboard entries (Page 47), e.g. to prevent a switchover from SINUMERIK Operate to the Windows Desktop. (Desktop access is then only possible via the splash screen (Page 99).)

Requirement

- A Windows user account without administrator rights is set up.
- SINUMERIK Operate is not installed.
- You are logged on as an administrator or you know the password of the administrator account.
- The keyboard layout of the system is set to English.

See: Configuring the keyboard layout (Page 24)

Procedure

To set up autologon mode, proceed as follows:

1. Start installation of SINUMERIK Operate and, in the installation wizard, proceed to step "Autologon".

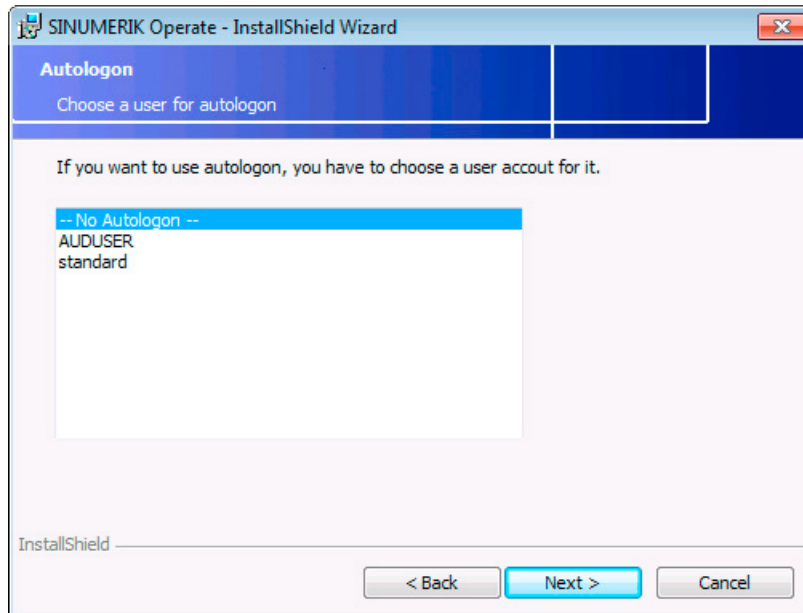


Figure 4-1 SINUMERIK Operate Installer - Autologon

2. In the list, select the user account to be logged on automatically.
3. Confirm with "Next", then in the next steps of the installation wizard make all the necessary settings and confirm the installation with "Finish".

Result

SINUMERIK Operate is installed and Autologon mode of the PC system is set up. During this process the password of the autologon user account is stored encrypted in Windows, and SINUMERIK Operate is linked with the autostart folder "Startup".

Autologon mode is active at the next startup. The autologon user account is automatically logged on and SINUMERIK Operate is started.

4.4 Modifying the user account for autologon mode.

Because of the way autologon works, only one user account can ever be set up for autologon mode.

If you set up autologon mode during installation of SINUMERIK Operate for one particular user account, but now want to use another user account for autologon mode, you can change this manually in the Service Desktop.

Requirement

SINUMERIK Operate is installed and autologon mode is set up.

The Service Desktop is active.

Procedure

To change the user account that is used for automatic startup in autologon mode, proceed as follows:

1. To open the "User Accounts" dialog box, proceed in one of the following ways:
 - On the Desktop, double-click the "Netplwiz" icon.
 - In the search field of the Start menu, enter `netplwiz` and confirm the entry.
 - At the prompt, enter `control userpasswords2`.

The "User Accounts" dialog box opens.

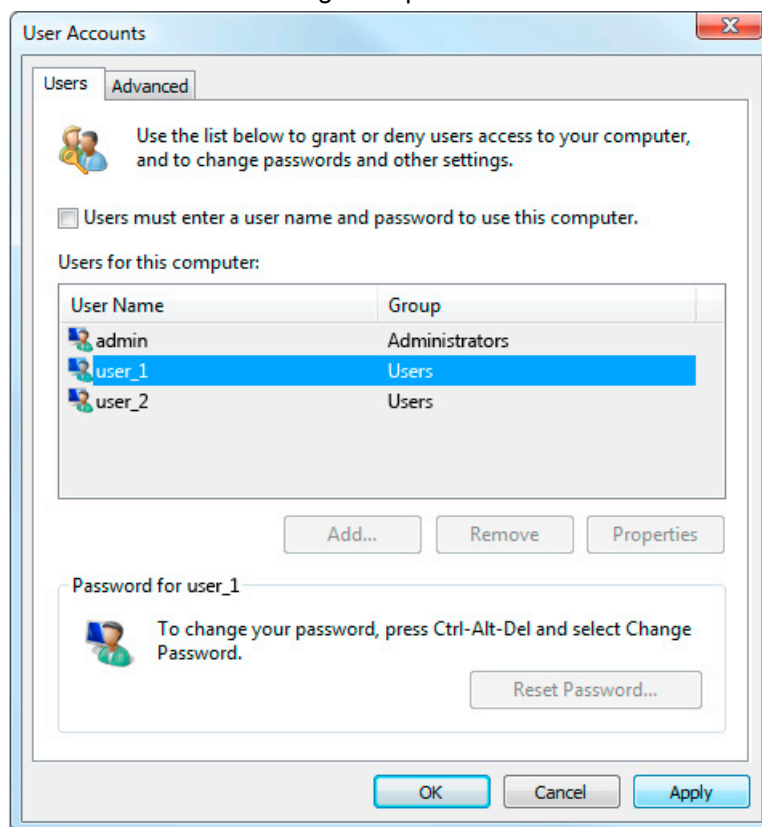


Figure 4-2 "User Accounts" dialog box in Windows 7: Please note that the deactivated checkbox refers to the selected user account.

2. Activate the "Users must enter a user name and password to use this computer" checkbox.
3. In the "Users for this computer" list, select the user account to be used for autologon mode.

4. Deactivate the "Users must enter a user name and password to use this computer" checkbox.
You may be prompted to enter a password.
5. Confirm these settings in the "User Accounts" dialog box with "OK".
6. In the Autostart folder of the autologon mode user account, create a link to SINUMERIK Operate.

4.5 Configuring a key filter for HMI software

To prevent unwanted interruption of the HMI software while it is running, you can specify that certain keyboard entries will be filtered out and thus ignored.

You can configure the key filter in the pcuhwsvc.ini configuration file:

- This key filter is not activated as delivered.
- You can activate the key filter and individually select which of the 20 filterable keyboard entries will be ignored. You cannot configure any additional keyboard entries to be filtered.
- You can also completely deactivate the repeat function of the F1-F12 keys and all hard keys with the key filter. When the repeat function is deactivated, a key held pressed is considered as pressed once and not as pressed repeatedly.

Per default, the repeat function is activated during Service or Desktop operation. The repeat function is always deactivated during operation of SINUMERIK Operate.

Requirement

- The Windows 7 Service Desktop is active
- Hidden directories are shown
See also: Directory structure and file conventions (Page 9).

Procedure

To configure the key filter, proceed as follows:

1. Switch to the template directory:
`C:\ProgramData\Siemens\MotionControl\siemens\System\etc\`
 The settings can be viewed in the template `pcuhwsvc.ini`
2. Save a copy of the template `pcuhwsvc.ini` to one of the user directories:
 - `C:\ProgramData\Siemens\MotionControl\user\System\etc\`
 - `C:\ProgramData\Siemens\MotionControl\oem\System\etc\`
 - `C:\ProgramData\Siemens\MotionControl\addon\System\etc\`
3. In the new file (e.g. `... \oem\System\etc\pcuhwsvc.ini`), delete all data except for the section name `[KEYB_FILTER]` and the keys that are to differ from the template.
All data and keys that do not occur in your file are automatically taken over from the

4.5 Configuring a key filter for HMI software

template in the `siemens` directory.

See also: Directory structure and file conventions (Page 9).

4. Save your file.

Example: Activating the key filter, but allowing use of the NUM-lock key.

This example activates the key filter (`KeySequencesEnable = 1`) and filters all possible keyboard entries except the NUM-lock key.

<code>KEYB_FILTER</code>	Section name (mandatory):
<code>KeySequencesEnable = 1</code>	Activate key filter
<code>SeqActive_l9= 0</code>	Do not filter NUM-lock key

In the adapted file in the user directory (e.g. `...\oem\System\etc\`) therefore only the section name `[KEYB_FILTER]` is mandatory as well as the settings that you want to adapt. All other settings and values are taken over from the template.

Template of the key filter in the delivery condition

The template is located in directory `C:\ProgramData\Siemens\MotionControl\siemens\System\etc\`

Do not overwrite this template, but save a copy of this template `pcuhwsvc.ini` in one of the user directories.

- `C:\ProgramData\Siemens\MotionControl\user\System\etc\`
- `C:\ProgramData\Siemens\MotionControl\oem\System\etc\`
- `C:\ProgramData\Siemens\MotionControl\addon\System\etc\`

See also: Directory structure and file conventions (Page 9).

How the configuration file works

You can configure the key filter in section `[KEYB_FILTER]`. The lines identified by a number sign (#) are comments as to which line further down deactivates a particular keyboard entry.

- To activate a particular keyboard entry, find the key in the comment and then set the value to 0.
- To allow all keyboard entries and thus deactivate the key filter (factory setting), set the value to 0 in the `KeySequencesEnable` key.
If the value 0 is set for this key, all other keys will be ignored and the key filter is deactivated.
- To completely deactivate the repeat function of the F1-F12 keys and all hard keys, set the value 1 at the `AutorepeatSuppression` key.

Table 4- 1 Overview: Key filter in the configuration file pcuhwsvc.ini

Section	[KEYB_FILTER]
Key	SeqActive_x
Value	0 = specific keyboard entry activated 1 = specific keyboard entry will be filtered and is deactivated
Factory setting	1

pcuhwsvc.ini (Vorlage im Verzeichnis ...\\siemens\\System\\etc)

```
[GLOBAL]
# -----
# SIEMENS GLOBAL SECTION
# -----
# enable logfile PCUHardwareService.log
EnableLogFile=0
# Emulating of "Mode-Switch" from Sinumerik PCU50 Box.
# "Mode-Switch" is a rotary switch, which is attached to the rear side of the PCU50 V5
Box.
# About this, different methods of booting the computer are controlled.
# 0 Normal mode
# 3 Desktop mode (welcome screen)
ModeSwitch=0
[SERVER]
FTP=hmisvr PCU betaftpd
DHCP=hmisvr PCU udhcpd
TFTP=hmisvr PCU netkit-tftpd
TCUHWS=hmisvr TCU hardware services
VNC=uvnc service

[APPLICATION]
# -----
# STARTUP APPLICATION DIAGNOSTIC SECTION
# -----

[KEYB_FILTER]
# -----
# SIEMENS KEYBOARD-FILTER DRIVER SECTION
# -----

# Filtering Key Sequences
# =====
# Activation of Key-Sequences to be ignored
# SeqAct x
# The following 20 sequences are implemented
# 0 CTRL-ALT-DEL
# 1 ALT-F4
# 2 ALT-TAB
# 3 LEFTSHIFT-ALT-TAB
# 4 RIGHTSHIFT-ALT-TAB
# 5 CTRL-ESC
# 6 ALT-ESC
# 7 ALT-SPACE
# 8 (SHIFT)-CTRL-F1
# 9 (SHIFT)-CTRL-F2
# 10 (SHIFT)-CTRL-F3
# 11 (SHIFT)-CTRL-F4
# 12 (SHIFT)-CTRL-F5
```

4.6 Installing STEP 7

pcuhwsvc.ini (Vorlage im Verzeichnis ...\\siemens\\System\\etc)

```
# 13 (SHIFT)-CTRL-F6
# 14 (SHIFT)-CTRL-F7
# 15 (SHIFT)-CTRL-F8
# 16 M$ 1
# 17 M$ 2
# 18 CAPSLOCK
# 19 NUMLOCK
# 20 (reserved)
```

```
KeySequencesEnable = 0
```

```
SeqActive 0= 1
SeqActive 1= 1
SeqActive 2= 1
SeqActive 3= 1
SeqActive 4= 1
SeqActive 5= 1
SeqActive 6= 1
SeqActive 7= 1
SeqActive 8= 1
SeqActive 9= 1
SeqActive 11= 1
SeqActive 12= 1
SeqActive 13= 1
SeqActive 14= 1
SeqActive 15= 1
SeqActive 16= 1
SeqActive 17= 1
SeqActive 18= 1
SeqActive 19= 1
SeqActive 20= 1
```

```
# Autorepeat-Suppression of Function Keys F1-F12 and Hardkeys
#=====
AutorepeatSuppression = 0
```

4.6 Installing STEP 7

Requirement

- The Service Desktop is active
- The STEP 7 product DVD is available on the PC system, e.g. via an external DVD drive or network.

Procedure

1. To start the installation assistant of STEP 7, on the product DVD in directory "CD_1", double-click `setup.exe`
2. Confirm the preselected directory under `C:\Program Files (x86)\Siemens\Step7\` as the installation directory.

3. In step "Transfer license keys", select "No, transfer license keys later".

Note

Licensing of STEP 7 with SINUMERIK Add-on

The licensing for STEP 7 is performed during installation of the SINUMERIK Add-on.

4. After you have installed STEP 7, restart the PCU and switch to the Service Desktop.
5. To start the installation wizard of the SINUMERIK Add-on, double-click `setup.exe` on the product DVD in directory "`Sinumerik_Add_On`".
The installation wizard guides you through the following steps.
6. After you have installed the SINUMERIK Add-on, restart the PC system.

Result

STEP 7 has been installed with SINUMERIK Add-on.

The software has been registered in the Control Panel under the following designations:

- STEP7 V5.5 SP3
- SINUMERIK Add-on for STEP7 V5.5

4.7 Linking the HMI software with STEP 7

If you want to make it possible to start STEP 7 directly from the user interface of the installed HMI software, add a softkey to it.

Requirement

- STEP 7 V5.5 SP3 or higher is installed
- The SINUMERIK Add-on for STEP 7 V5.5 or higher is installed
- An HMI software version is installed (SINUMERIK Operate or HMI Pro sl)
- The Service Desktop is active

Procedure

To add a softkey to the menu of the HMI software for starting STEP 7, proceed as follows:

1. Double-click the "STEP7-Authorizing" desktop icon.

Result

A softkey for starting STEP 7 has been added to the HMI software. This softkey requires access level 3.

In `C:\Program Files (x86)\Siemens\MotionControl\compat\add_on\oemframe.ini`, the following settings have been made for this purpose:

```
[s7tgttopx]
; with HMI-Advanced: eliminate minimize- / maximize-buttons of the STEP7-window
WindowState_Off=196608
; with HMI-Advanced: switch to previous task when STEP7 is terminated
nSwitchToTaskAfterTermination= -2
```

Note

Adjustments for OEM configurations

The entries in `oemframe.ini` can be modified for OEM configurations.

For more information about templates for parameterizing the system, see Section Directory structure and file conventions (Page 9).

4.8 PCU Installer

4.8.1 Overview

The PCU Installer assists you with the installation of setup packages on PC systems:

- The PCU Installer checks definable setup directories on the PC system, on removable disks, or in the network.
These setup directories must not be write-protected.
- If setup packages exist, they are automatically installed one after the other. The order in which these packages are installed can be parameterized (Page 56).

For further information on the activities of the PCU Installer, see Chapter Procedure example of an installation with the PCU Installer (Page 61).

Overview of the configuration and operation

- In the delivery condition of the PCU base software, the PCU Installer is deactivated. You can activate the PCU Installer during the unattended installation of the PCU base software via the response file (Page 15) or activate it manually after the attended installation (Page 53).
- You can make all settings in the PCU Installer in the associated configuration file (Page 56).
- Copy the setup packages to the defined directories to start the installation (Page 60).
- The PCU Installer consists of a client and a service, whereby setup packages can also be installed for a user account without administrator rights.

4.8.2 Activating the PCU Installer

4.8.2.1 Overview

In the delivery condition of the PCU base software, the PCU Installer is deactivated.

There are different ways of activating the PCU Installer:

- If you complete the installation of the PCU-Basesoftware unattended, you can have the PCU Installer activated via the `unattend.xml` response file. See Configuring an answer file (Page 15).
- If you complete the installation of the PCU base software with Windows 7 attended, or have not configured activation of the PCU Installer in the response file, you must activate the PCU Installer manually. To do this, use one of the following procedures:
 - Copy a preconfigured template of the PCUInst.ini to the "user" directory (Page 53)
 - Create your own version of the PCUInst.ini in one of the user directories and set the appropriate values
See: Reference to the PCUInst.ini configuration file (Page 56)

4.8.2.2 Activating via a configuration template

Requirement

- The PCU Installer is deactivated
- The Service Desktop is active
- The PCU Installer has not been configured yet and there is no adapted PCUInst.ini configuration file

Procedure

To activate the PCU Installer by copying a preconfigured template, proceed as follows:

1. Copy the following file to the clipboard: `D:\etc\Template_PCUInst.ini`
2. Paste the file into the following directory:
`C:\ProgramData\Siemens\MotionControl\user\system\etc`
3. Rename the copied file to `PCUInst.ini`

Result

The PCU Installer has been activated and checks the setup directories during startup of the PC system, but not during system operation.

The copied configuration file only differs from the default configuration file by the following value:

```
[processing_loginphase]
StartState=activated
```

4.8 PCU Installer

The settings of the copied configuration file are used because they are in a directory with higher priority than the default configuration file.

See: Directory structure and file conventions (Page 9)

To also use the PCU Installer during system operation, you must make an additional setting in the copied configuration file (Page 56):

```
[processing_systemphase]
StartState=activated
```

4.8.3 Deactivating the PCU Installer

In the delivery condition of the PCU base software, the PCU Installer is deactivated.

Note

Troubleshooting for incorrect installation routines

If completion of an installation requires a restart, but the installation cannot be completed after the restart, the Service Desktop is not started.

In this case, restart the PC system in protected mode to prevent the PCU Installer from starting. (To start in protected mode, press F8 repeatedly during startup until a selection menu opens.)

Then reconfigure the PCU Installer or remove the setup packages.

Requirement

- The Service Desktop is active
- Hidden directories are not shown (at least `C:\ProgramData\`)
See also: Directories hidden in delivery condition (Page 9).
- A copy of the `PCUInst.ini` configuration file is stored in the `user` directory:
`C:\ProgramData\Siemens\MotionControl\user\System\etc\PCUInst.ini`
- The `PCUInst.ini` configuration file in the `user` directory is not write-protected

Procedure

To partly or completely deactivate the installation of the setup files by means of the PCU Installer, proceed as follows:

1. Open the `PCUInst.ini` configuration file in the
`C:\ProgramData\Siemens\MotionControl\user\System\etc\PCUInst.ini` directory.
2. Deactivate the start of the PCU Installer in the desired phase:
 - To deactivate the PCU Installer during system operation, change the relevant key value in the `[processing_systemphase]` section: `StartState=stopped`.
 - To only deactivate the PCU Installer while a specific application is being executed (e.g. SINUMERIK Operate), enter the file name of the relevant application in the `[noactivation]` section, e.g. `APP001=run_hmi.exe`
(You can specify any number of applications, e.g.
`APP002=Siemens.Automation.Portals.exe`)
 - To deactivate the PCU Installer during run-up, change the relevant key value in the `[processing_loginphase]` section: `StartState=stopped`.
3. Save and close the file.

Further information

For information on all configuration options, see Reference to the `PCUInst.ini` configuration file (Page 56).

4.8.4 Copying and adapting the configuration file

You can define PCU Installer operation in the associated configuration file. To do this, you must first copy the configuration file to a user directory.

Alternatively, you can copy a template in which the PCU Installer has already been activated (Page 53).

For example, you can configure the following in the configuration file:

- Which directories have been checked how many times for new setup packages.
- That installations are not started automatically while certain software is active.

A list of all setting options can be found under Reference to the `PCUInst.ini` configuration file (Page 56).

Requirement

- The Service Desktop is active
- The PCU Installer has not been configured yet and there is no adapted `PCUInst.ini` configuration file

Procedure

To copy and adapt the configuration file of the PCU Installer, proceed as follows:

1. Switch to the template directory:

`C:\ProgramData\Siemens\MotionControl\siemens\System\etc\`

The settings can be viewed in the template `PCUInst.ini`

2. Save a copy of the template `PCUInst.ini` to one of the user directories:

– `C:\ProgramData\Siemens\MotionControl\user\System\etc\`

– `C:\ProgramData\Siemens\MotionControl\oem\System\etc\`

– `C:\ProgramData\Siemens\MotionControl\addon\System\etc\`

3. Adapt the values of the keys. You will find information about the syntax and setting options at Reference to the `PCUInst.ini` configuration file (Page 56) or in the comments of the configuration file.

Note

The factory setting is taken if key pairs are missing

All key pairs that are not available in your file are taken automatically from the template in the `siemens` directory.

Therefore, in the new file (e.g. `...\oem\System\etc\PCUInst.ini`), delete all the data except the section names and the keys that are to differ from the template.

See also: Directory structure and file conventions (Page 9), Reference to the `PCUInst.ini` configuration file (Page 56)

4. Save and close the file.

Result

The configuration file has been copied to a user directory and can be adapted.

You will find further information on the setting options in the following chapters.

4.8.5 Reference to the `PCUInst.ini` configuration file

You can define PCU Installer operation in the associated `PCUInst.ini` configuration file.

For example, you can configure the following:

- Which directories have been checked how many times for new setup packages.
- That installations are not started automatically while certain software is active.

The priority concept of this configuration file is the same as that of the other configuration files, see Directory structure and file conventions (Page 9).

You can also deactivate settings in `PCUInst.ini` by deleting the value of the key in the higher priority configuration file. For example, to deactivate `DIR002=C:\Setup` in a file of lower priority, enter `DIR002=` in the file with higher priority.

Note

Maximum length of the file path

The file path to a setup must not exceed 255 characters (including the file name).

Table 4- 2 Setting options in the configuration file of the PCU Installer

Section	Settings	Description
[local_setupdirs]	DIR001= DIR002= ...	Specifies the local setup directories on the PC system to be checked. The directories must not be write-protected. You can specify several directories, which are then checked in the specified order. You define the order with a number after "DIR". Subdirectories at the first level are also scanned.
[removable_setupdirs]	DIR001= DIR002= ...	Specifies the setup directories on removable disks to be checked. The directories must not be write-protected. You can specify several directories, which are then checked in the specified order. You define the order with a number after "DIR". Subdirectories at the first level are also scanned. If you do not know the drive letters to be checked, use the <code><REMOVABLE_MEDIA></code> drive variable, e.g.: <code>DIR001=<REMOVABLE_MEDIA>:\Install</code> The drive variable then represents all the removable disks that are known at the time in Windows. You can also specify fixed drive letters without the directories being checked twice. The drive variable is only valid in the [removable_setupdirs] section.
[net_setupdirs]	DIR001= DIR002= ...	Specifies the setup directories on network drives to be checked. The directories must not be write-protected. See also: Creating a shortcut to the network drive (Page 91) You can specify several directories, which are then checked in the specified order. You define the order with a number after "DIR". Subdirectories at the first level are also scanned.
[setupdirs_settings]	deleteTmpLocalSetupDirs=	Temporary directories are created during installation of a setup from a removable disk, network drive or the default setup directory <code>D:\Install</code> (See also: Procedure example of an installation with the PCU Installer (Page 61)) Here you can specify whether these temporary directories are to be deleted again after installation. Possible values: yes; no
	keepDInstallSubdirsWithPrefix=	You can specify that subdirectories of <code>D:\Install</code> with a particular prefix in the directory name are not to be deleted after setup is completed.

Section	Settings	Description
[setupnames]	NAME001= NAME002= ...	Specifies which files are setup packages and are to be included in the check of setup directories. You can specify entire file names with their file extensions or represent individual parts of them with wildcard character *.
[setupexecution]	checkSetupLogFile=	Specifies whether a check is to be performed as to whether the file has already been installed before the installation of a setup file. The check is performed based on the computer name and the setup name of the setup-specific log file. Possible values: yes; no
	copyToTmpLocalSetupDir=	You can specify whether setup files on removable disks are to be copied to a temporary, local directory before installation. Setup files on network drives are always copied before installation. Possible values: yes; no
[noactivation]	APP001= APP002=	Specifies which applications of the PCU Installer are inactive during operation. This setting is required to prevent operation of the HMI software or another application from being interrupted during execution of a setup.
[process- cess- ing_loginphase]	StartState=	Specifies whether the PCU Installer is active before the Desktop runs up. The setting <code>StartState=activated</code> is required, in particular, if an installation requires a cold start. Possible values: • <code>activated</code> PCU Installer starts setups • <code>stopped</code> PCU Installer is stopped completely
	continueSetups=	Some setups can only be completed after the system has been restarted. In this case, the PCU Installer waits until you restart the PC system. You can specify here whether such setups are to be continued after a restart. (See also: Note: Troubleshooting for incorrect installation routines (Page 60)) Possible values: yes; no
	scanSetupDirs=	Specifies whether setup directories are to be checked and, if required, setup files executed before the Desktop run-up. The following directories are checked if this setting has been activated: • Local setup directories from the <code>[local_setupdirs]</code> setting are always checked. • You can define separately whether additional directories on removable disks or in the network are to be checked. (See settings "scanRemovableDriveSetupDirs" and "scanNetworkDriveSetupDirs") If you deactivate this setting, neither local directories, nor directories on removable disks or in the network are checked, i.e. the setting <code>scanSetupDirs=no</code> overwrites, for example, <code>scanNetworkDriveSetupDirs=yes</code> . Possible values: yes; no
	scanRemovableDriveSetupDirs=	Specifies whether directories on removable disks are to be checked from the setting <code>[removable_setupdirs]</code> and, if required, setup files executed before the Desktop run-up. The setting "scanSetupDirs" must also be activated so that these directories are checked. Possible values: yes; no

Section	Settings	Description
	scanNetworkDriveSetupDirs=	Specifies whether network directories are to be checked from the setting [net_setupdirs] and, if required, setup files executed before the Desktop run-up. The setting "scanSetupDirs" must also be activated so that these directories are checked. Possible values: yes; no
[process- cess- ing_systemphase]	StartState=	Specifies whether the PCU Installer is active during operation of the Desktop. Possible values: - activated PCU Installer starts setups - stopped PCU Installer is stopped completely You can define exceptions in Section "[noactivation]".
	scanSetupDirs=	Specifies whether setup directories are to be checked and, if required, setup files executed during Desktop operation. The following directories are checked if this setting has been activated: - Local setup directories from the [local_setupdirs] setting are always checked. - You can define separately whether additional directories on removable disks or in the network are to be checked. (See settings "scanRemovableDriveSetupDirs" and "scanNetworkDriveSetupDirs") If you deactivate this setting, neither local directories, nor directories on removable disks or in the network are checked, i.e. the setting scanSetupDirs=no overwrites, for example, scanNetworkDriveSetupDirs=yes. Possible values: yes; no
	scanRemovableDriveSetupDirs=	Specifies whether directories on removable disks are to be checked from the setting [removable_setupdirs] and, if required, setup files executed during Desktop operation. The setting "scanSetupDirs" must also be activated so that these directories are checked. Possible values: yes; no
	scanNetworkDriveSetupDirs=	Specifies whether network directories are to be checked from the setting [net_setupdirs] and, if required, setup files executed during Desktop operation. The setting "scanSetupDirs" must also be activated so that these directories are checked. Possible values: yes; no
	numberOfScanCycles	Specifies whether and how often the setup directories check for new setup packages. Possible values: - infinite Infinity - Integer You can define the waiting time between checks in key waitTimeBetweenScanCycles
	waitTimeBetweenScanCycles	Specifies how long the PCU Installer waits after the setup directories have been checked before checking them again. Enter the waiting time in milliseconds. In key numberOfScanCycles, you can specify how often the setup directories will be checked.

Section	Settings	Description
report	InfoLevel	Specifies how detailed the information is to be which is stored in the log files. Possible values: • 1: Only basic information on the actions of the PCU Installer is logged • 2: All PCU Installer-specific information is logged • 3: All PCU Installer-specific information and some trace information is logged • 4: All PCU Installer-specific information and additional trace information is logged • 5: All PCU Installer-specific information and all trace information is logged

4.8.6 Example configuration: Installation behavior as for PCU base software Windows XP

If you want the automatic installation of setups to be configured in the same way as for the PCU base software Windows XP, you can use the following configuration example in your configuration file with higher priority.

```
[removable_setupdirs]
DIR001=E:\install

[processing_loginphase]
StartState=activated

scanRemovableDriveSetupDirs=yes
```

You can deactivate all other entries of a PCUInst.ini with lower priority by entering the appropriate key in your configuration file, but leaving the value empty. For example, to deactivate `DIR002=C:\Setup` in a configuration file of lower priority, enter `DIR002=` in your file.

You can make any number of adaptations in your file. See: Reference to the PCUInst.ini configuration file (Page 56)

4.8.7 Installing software with the PCU Installer

Requirement

- The PCU Installer is activated
See: Activating the PCU Installer (Page 53)
- If you want to install multiple setup packages one after the other, you must have defined multiple setup directories in the configuration file
- The setup directories to be used are empty and not write-protected

Procedure

To make the PCU Installer accessible to setup packages, proceed as follows:

1. Copy the setup packages to be installed into the setup directories. Copy each setup package into a separate directory.

Note

Maximum length of the file path

The file path to a setup must not exceed 255 characters (including the file name).

Result

The installation of the setup packages is started.

After a setup has been completed or cancelled, a specific log file is stored in the associated setup directory. The general log files of the PCU Installer are also supplemented in the log directory.

Further information

- Log files (.log) (Page 62)
- Procedure example of an installation with the PCU Installer (Page 61)

Note

Troubleshooting for incorrect installation routines

If completion of an installation requires a restart, but the installation cannot be completed after the restart, the Service Desktop is not started.

In this case, restart the PCU 50.5 in protected mode to prevent the PCU Installer from starting. (To start in protected mode, press F8 repeatedly during startup until a selection menu opens.)

Then reconfigure the PCU Installer or remove the setup packages.

4.8.8 Procedure example of an installation with the PCU Installer

The following example describes the activities of the PCU Installer if you have stored a setup file on a removable disk or network drive after the activation and configuration of the PCU Installer.

In this procedure example, it is assumed that you have already performed the following steps:

- You have activated the PCU Installer
- You have activated the checking of the network directories in the settings of the PCU Installer and defined an additional setup directory on a network drive.
In the example: F:\Setup with F: as network drive.

- You have stored a setup file in this additional setup directory (e.g. F:\Setup\Update.exe).
- The setup directory is not write-protected.

Example: Activities of the PCU Installer during installation of a network drive

- The defined setup directories are checked.
 - The Update.exe file is found in the F:\Setup directory.
As the Update.exe file has not been installed yet, there is no setup-specific log file in the directory.
 - The Update.exe file is copied to a local temporary directory, e.g. D:\Install39\Update.exe
 - The copied setup file is executed from the local temporary directory, e.g. D:\Install39\Update.exe
 - Log files are created after completion of the installation:
 - The general log files are extended
 - The setup-specific log file is created in the original setup directory F:\Setup and in the local temporary directory D:\Install39
- See: Log files (.log) (Page 62)

4.8.9 Log files (.log)

In order to be able to track which setup packages and updates have been installed at any time, the activities of the PCU Installer are recorded in various log files.

Setup-specific log files

A separate setup-specific log file is created for each setup package that is executed by the PCU Installer. The name of the setup and the name of the computer are entered in the file name of the log.

Archiving	The setup directory in which the associated setup file was originally located, and (if copied by the setup) the local temporary directory.
File name	The log file is designated according to the following scheme: <Name of the setup file>@<Computer name>.log, e.g. Update@MyPCU50.log

If you activate the checkSetupLogFile= setting in the "PCUInst.ini" configuration file (Page 56), then the setup-specific log file prevents the same setup being installed several times on the same computer. The check is performed based on the computer name and the setup name.

In the delivery condition of the base software, this setting is deactivated. The setting should remain deactivated for the series start-up of devices with the same name or for the repair of an installation.

General log files

The general PCU Installer log files record the operation and the communication of the PCU Installer client and PCU Installer service. There are therefore separate files for the client and service.

Archiving	C:\ProgramData\Siemens\MotionControl\User\System\Log\
File name	PCU Installer client: PCUInstaller_C.log
	PCU Installer service: PCUInstaller_S.log

4.9 Migration

Basic procedure

To transfer data from a PCU with Windows XP to a PC system with Windows 7, store the data on a USB flash drive, for example, and then copy it to the PC system:

- For SINUMERIK Operate, use the functions for generating and reading in commissioning archives.
- For STEP 7 V5.x, copy the generated STEP 7 project data onto the PC system.

Further information

- Information about how commissioning files work in **SINUMERIK Operate** can be found in the *SINUMERIK 840D sl SINUMERIK Operate Commissioning Manual (IM9)*, Chapter *Data backup*.
- Information on **STEP 7 V5.x** can be found in the associated online help.
- Information about other SINUMERIK applications is provided in the relevant documentation. You will find documentation on a large number of products in the SIEMENS Service&Support portal:
Service&Support portal > CNC automation system SINUMERIK
(<http://support.automation.siemens.com/WW/view/en/10805517/133300>)

Backing up and restoring data

5.1 Overview

You can perform installation and commissioning work on the PC system in the Service Center, which is based on Microsoft Windows PE.

When the Desktop is active, configuration of the Service Center is possible via the "ServiceCenter Backup-Restore" (Page 39) icon.

Most of the functions of the Service Center are available in the main menu (Page 66).

In particular, you can create or restore disk images in the Service Center. The following options are available:

- Create a disk image of the SSD (Page 70)
- Restore a disk image of the SSD (Page 72)
- Create a disk image of a partition (Page 74)
- Restore a disk image of a partition (Page 76)

The Service Center uses the Symantec Ghost (Page 80) software, which you can also call separately.

5.2 Starting the Service Center

To start the Service Center, proceed in one of the following ways:

- To call up the Service Center on the SSD while the PC system is starting, select the "Bootting Service System" during startup in the Windows boot menu.
- To call up the emergency boot system on a USB flash drive while the PC system is starting, set the service switch to E or press F12 during startup to show the boot selection.
- To switch to the Service Center while the Service Desktop (Windows 7) is active, click the "ServiceCenter Backup-Restore" icon on the Desktop and then click "Start ServiceSystem".

5.3 Functions of the Service Center

Overview

In the upper section of the Service Center main menu, you can select one of five tasks. Buttons for additional functions are available in the lower section.

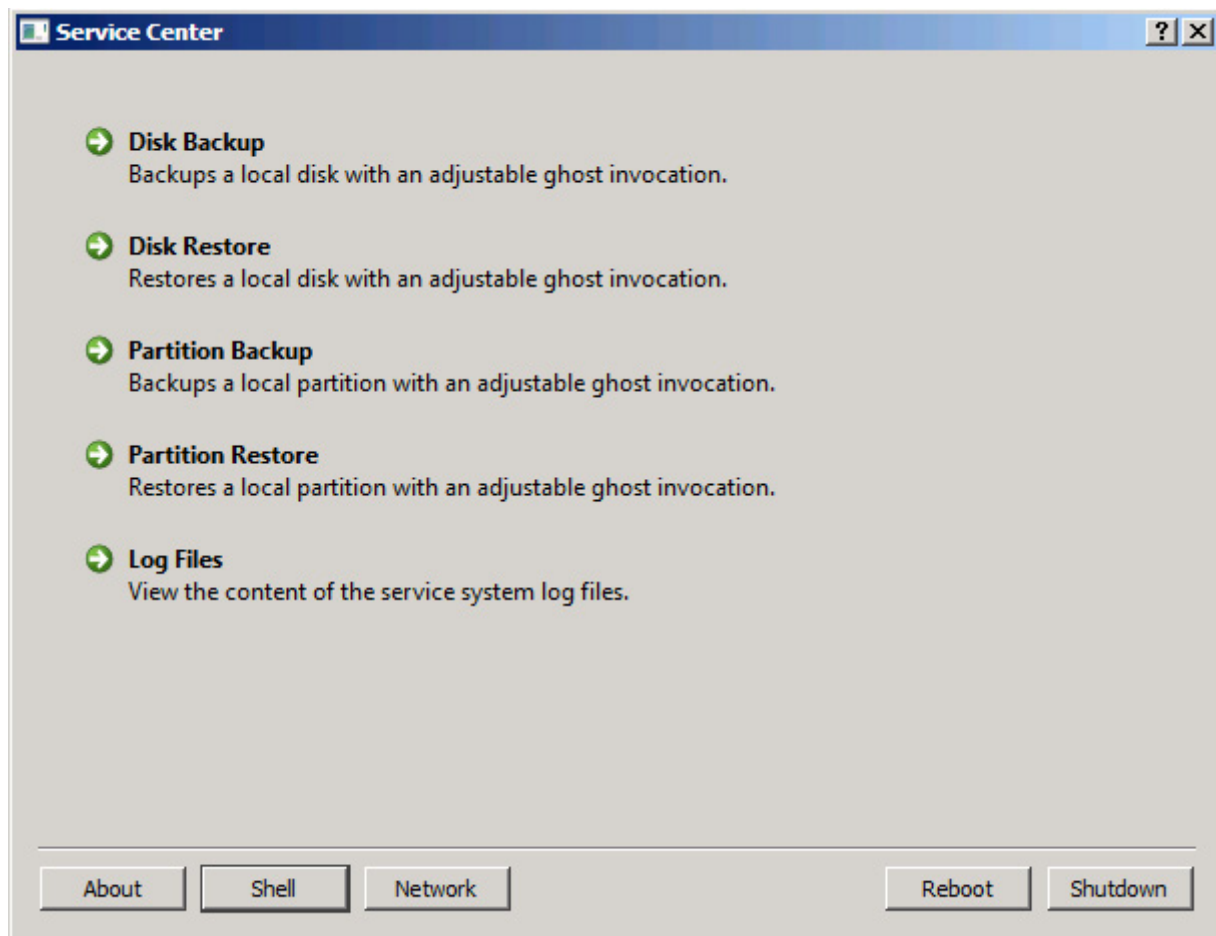


Figure 5-1 Service Center - Main menu

Table 5- 1 Functions in the main menu of the Service Center

Element of the window	Purpose
Disk Backup	Create a disk image of the local SSD. With this command, you can create a disk image of the entire SSD (with all partitions).
Disk Restore	Restore the local SSD with an existing disk image. With this command, you can restore the complete SSD (with all partitions) with a previously created disk image.
Partition Backup	Create a disk image of a partition on the local SSD. With this command, you can create a disk image of an individual partition of the SSD.

Element of the window	Purpose
Partition Restore	Restore a partition on the local SSD with a disk image. With this command, you can restore an individual partition of the SSD with a previously created disk image.
Log Files	View and save log files.
About	Viewing information about the Service Center.
Shell	Open the prompt. With this command, you can enter commands for executing tasks on the PC system without using a graphical user interface.
Network	Making network settings (Page 67). Any network settings you make in this dialog box are temporary. To change network settings permanently, switch to the Service Desktop.
Reboot	Restart the PC system.
Shutdown	Shut down the PC system.

Note**Data backup and restore with CompactFlash cards**

If there is a CompactFlash card in the slot of the PC system, you will not be able to back up or restore data via the Service Center.

Instead, start the "ghost32.exe" program via the prompt of the Service Center.

5.4 Network settings in the Service Center

You can make network settings in the main menu of the Service Center by clicking "Network":

- IP settings at the enterprise network interface X1 (Local Area Connection 2)
- IP settings at the plant network interface X2 (Local Area Connection)
- Configuration of the host
- Configuration of IP routing

Note**Temporarily storing network settings in the Service Center**

Any network settings you make in this dialog box are immediately effective, but temporary.

To change network settings permanently, switch to the Service Desktop.

For further information, see Section Configuration of the Service Center (Page 39).

Overview

IP settings

You can make the following IP settings in the "Network Settings" dialog box:

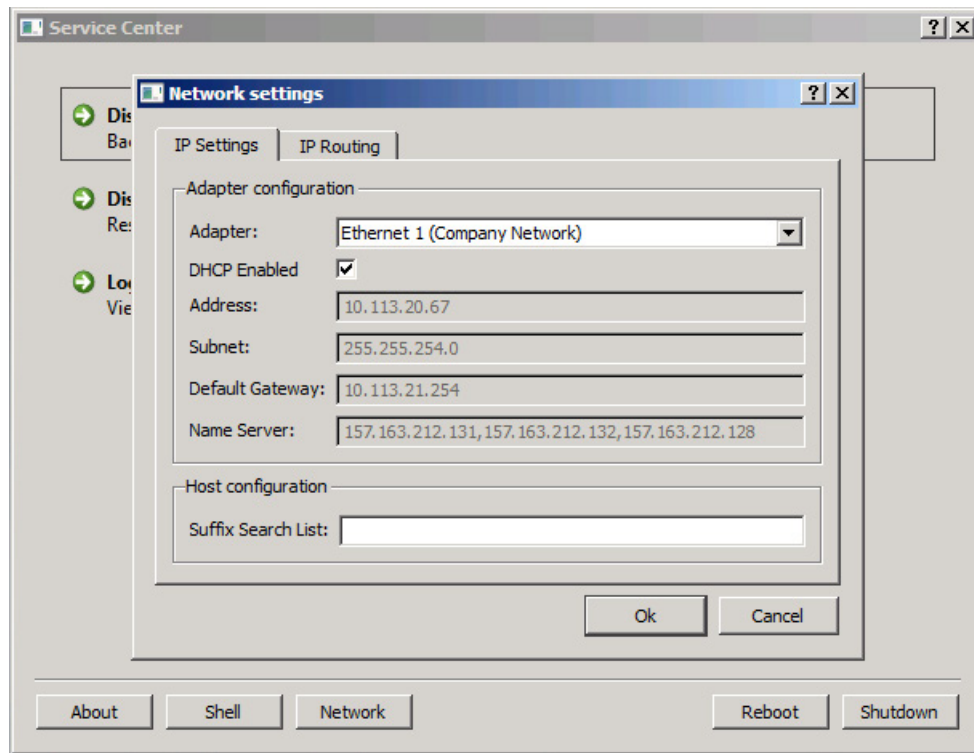


Figure 5-2 Service Center - IP settings

Table 5- 2 IP settings in the "Network Settings" dialog box

Section	Settings	Purpose
Adapter configuration	Adapter	Select the Ethernet interface that you want to configure. All settings in Section "Adapter configuration" refer to the selected Ethernet interface.
	DHCP Enabled	Select whether the addresses are to be fetched dynamically from the DHCP server. If the checkbox is deactivated, you must make the settings manually.
	Address	Set the IP address of the NCU. You can specify an IP address from the following range: 192.168.214.250 - 254
	Subnet	Specify a subnet mask, e.g. 255.255.255.0
	Default Gateway	Set the IP address of the standard gateway. A standard gateway creates a standard route in the IP routing table for all destinations that are not in the subnet.

Section	Settings	Purpose
	Name Server	Set the IP address of the name server. A name server answers questions asked about a domain name zone using a DNS database.
Host configuration	Suffix Search List	Parameterize a search list for DNS suffix, e.g. "network.com". The factory setting of the Ethernet interface "Local Area Connection" is ".local".

Routing settings

You can make the following routing settings in the "Network Settings" dialog box:

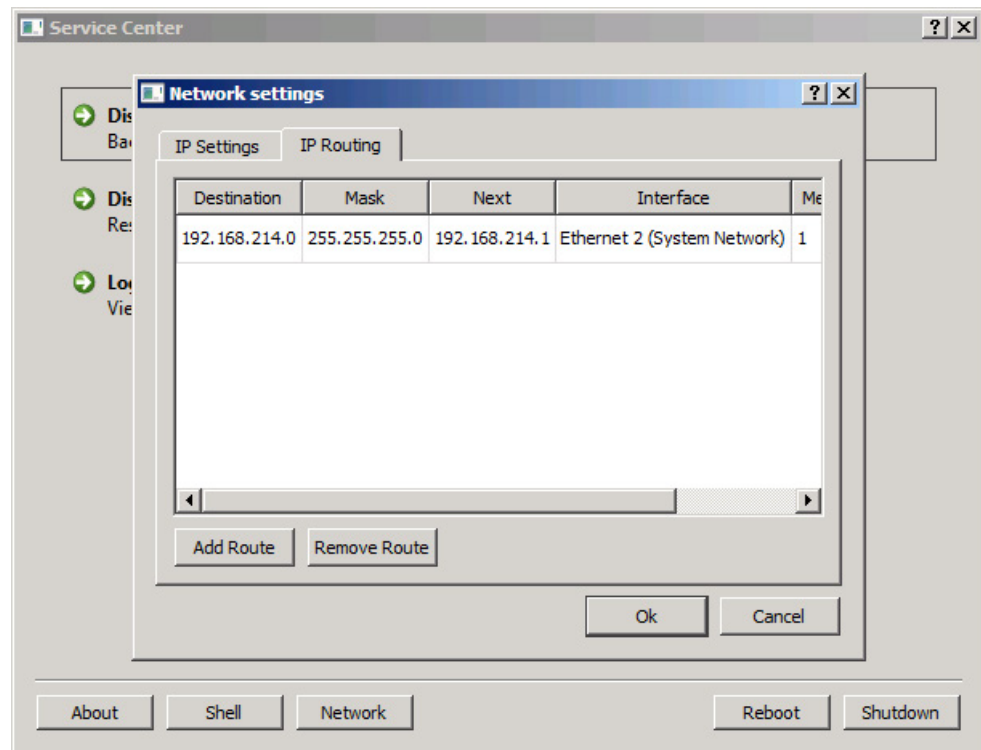


Figure 5-3 Service Center - IP routing

Table 5-3 Routing settings in the "Network Settings" dialog box

Settings	Purpose
Add Route	Add a new route.
Remove Route	Delete a selected route.
Destination	Specify the network destination of the route as an IP address.
Mask	Subnet mask of the specified IP addresses.
Next	Next hop or forwarding IP address via which the network destination can be accessed.

Settings	Purpose
Interface	Select an interface to which the configured IP routing applies: - Enterprise network interface X1 (Local Area Connection 2) - Plant network interface X2 (Local Area Connection)
Metric	Integer cost metric (1 to 9999). Required if several routes for sending a packet to a network destination are possible in the routing table.

See also

Network settings (Page 8)

Release directory of the PC/PC in the network (Page 88)

Creating a shortcut to the network drive (Page 91)

Configuring routing in the network (Page 83)

5.5 Create a disk image of the SSD

You can create a disk image of the entire SSD in the Service Center with the "Disk Backup" function.

Procedure

To save the SSD as a disk image, proceed as follows:

1. Start the Service Center, e.g. via the Windows boot menu.
See also: Starting the Service Center (Page 65)
2. In the main menu of the Service Center, click "Disk Backup".
The "Service Center - Disk Backup" dialog box opens.

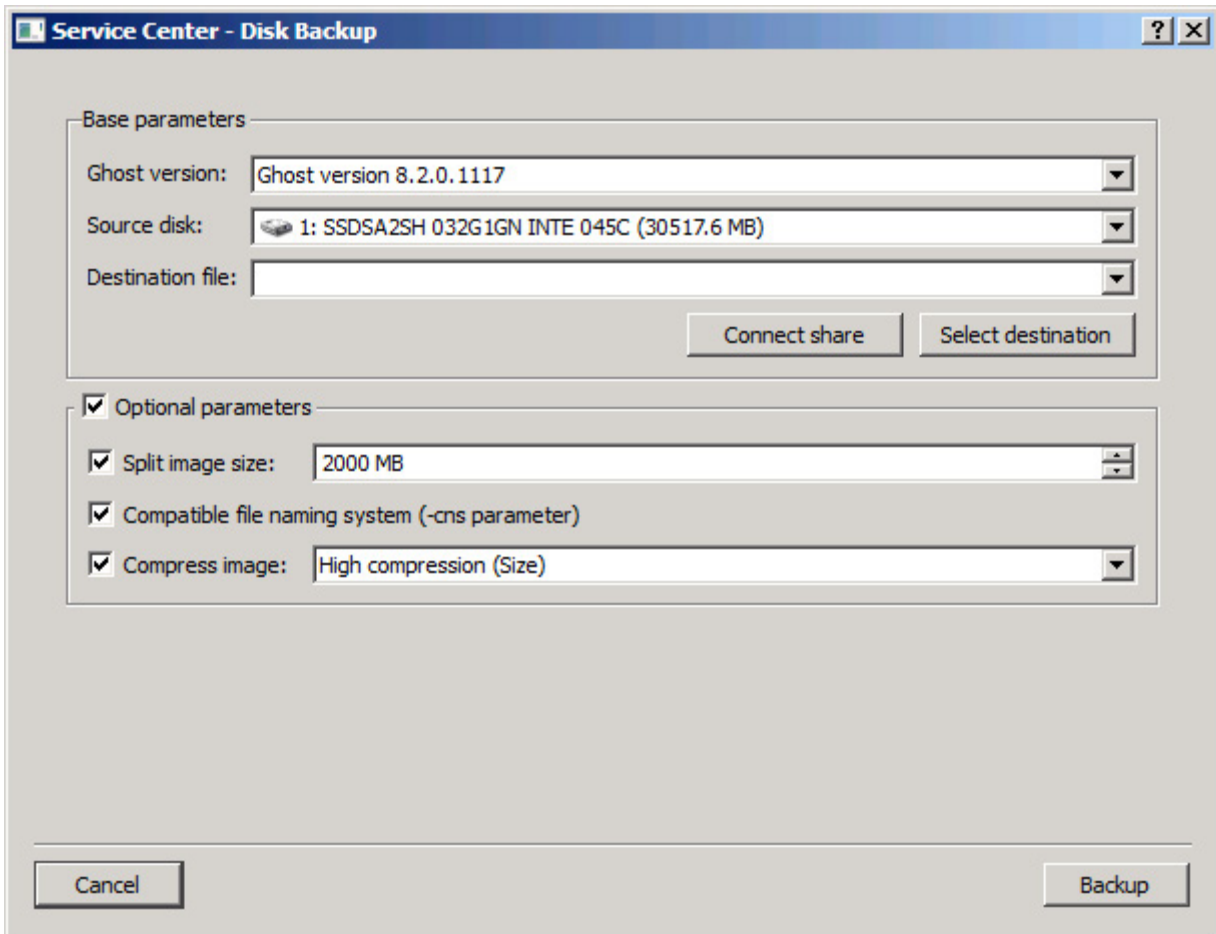


Figure 5-4 Service Center - Disk Backup

3. Make the required settings here:

Section	Settings	Purpose
Base parameters	Ghost version	Select the Symantec Ghost version that is to be used when creating the disk image: • Version for Windows 7: "Ghost version 11.5.x" • Version for Windows XP: "Ghost version 8.2.x"
	Source disk	Select a data medium of which you want to create a data image.

Section	Settings	Purpose
	Destination file	Enter a destination directory and file name. Disk images can be stored either locally or in the network.
Base parameters > Destination file	Connect share	Open the "Connect a Share" dialog box (Page 78). You can set up a shared resource (e.g. a directory in the network), to which you can store the disk image.
	Select destination	Open the "Select Destination File" dialog box. You can navigate through a local directory of the PC system and enter a file name.
Optional parameters	Split image size	Split the disk image and select a split image size. You can split the disk image into several files of a defined size. Split the disk image size in the following cases: • If the file system (e.g. FAT32) can only manage files of a certain size. • If the disk image is to be archived onto multiple data media, for example, when using CDs.
	Compatible file naming system	Select the compatibility of the name scheme. This option is required so that a disk image with older versions of the "PCU base software Windows XP" can be read in by Windows XP. It is not possible to use a different operating system on a PCU 50.x than the preinstalled system (e.g. Windows XP on a PCU 50.5 Windows 7).
	Compress image	Activate Compression or select a compression level. Higher compression results in a smaller file size, but increases the time needed to compress or decompress the data.

4. Confirm your settings with "Backup" to start generation of the disk image.

5.6 Restore a disk image of the SSD

If your system is no longer functioning stably, you can restore it with a disk image.

Note

Restoring replaces all current files

When you restore the PC system from a disk image, all programs, settings and files are completely replaced by the disk image. You cannot restore individual files or exclude individual files from restoration.

Requirement

A disk image exists

See: Create a disk image of the SSD (Page 70)

Procedure

To restore a disk image, proceed as follows:

1. Start the Service Center, e.g. via the Windows boot menu.
See also: Starting the Service Center (Page 65)
2. In the main menu of the Service Center, click "Disk Restore".

The "Disk Restore" dialog box opens.

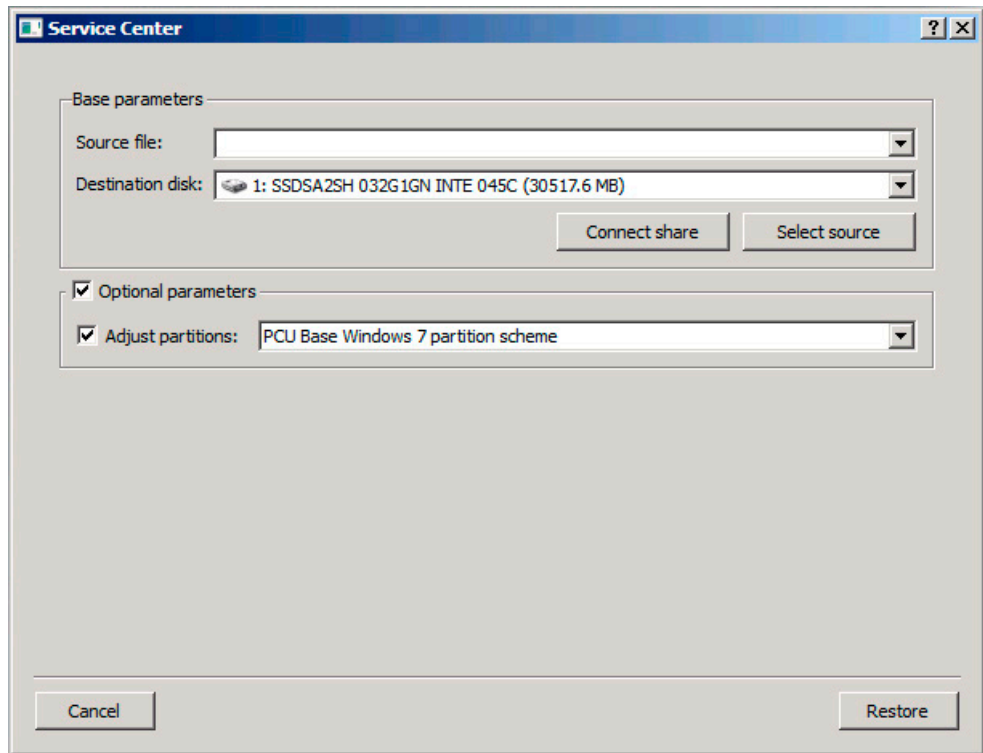


Figure 5-5 Service Center - Disk Restore

3. Make the required settings here:

Section	Settings	Purpose
Base parameters	Source file	Select the disk image from which the SSD is to be restored. You can use local disk images or disk images stored in the network.
	Destination disk	Select the SSD to be restored from the disk image.
Base parameters > Destination disk	Connect share	Open the "Connect a Share" dialog box (Page 78). You can manage a shared resource (e.g. a directory in the network), to use a disk image stored on the network to restore the SSD.
	Select source	Open the "Select Source File" dialog box. You can navigate through a local directory of the PC system and select a disk image for restoration.

Section	Settings	Purpose
Optional parameters	Adjust partitions	Select the partition scheme to be used when restoring the disk image on a PCU 50.x: • If the PCU base software Windows 7 is on the disk image, choose "PCU Base Windows 7 partition scheme". • If the PCU base software Windows XP is on the disk image, choose "PCU Base Windows XP partition scheme". It is not possible to use a different operating system on a PCU 50.x than the preinstalled system (e.g. Windows XP on a PCU 50.5 Windows 7). If you are using a different PC system than the PCU 50.5, you must deactivate the "Adjust partitions" setting.

4. To start the restore process and entirely replace the existing data on the SSD with the disk image, confirm the settings with "Restore".

Result

The restore process is started.

Note

Troubleshooting if reading in of the disk image fails

If data transfer is interrupted during the restore process, the PC system will have no executable system.

In this case, use a bootable USB flash drive with Service Center to read the disk image in again.

5.7 Create a disk image of a partition

You can create a disk image of an individual partition of the SSD in the Service Center with the "Partition Backup" function.

Procedure

To save a partition of the SSD as a disk image, proceed as follows:

1. Start the Service Center, e.g. via the Windows boot menu.
See also: Starting the Service Center (Page 65)
2. In the main menu of the Service Center, click "Partition Backup".
The "Service Center - Partition Backup" dialog box opens.

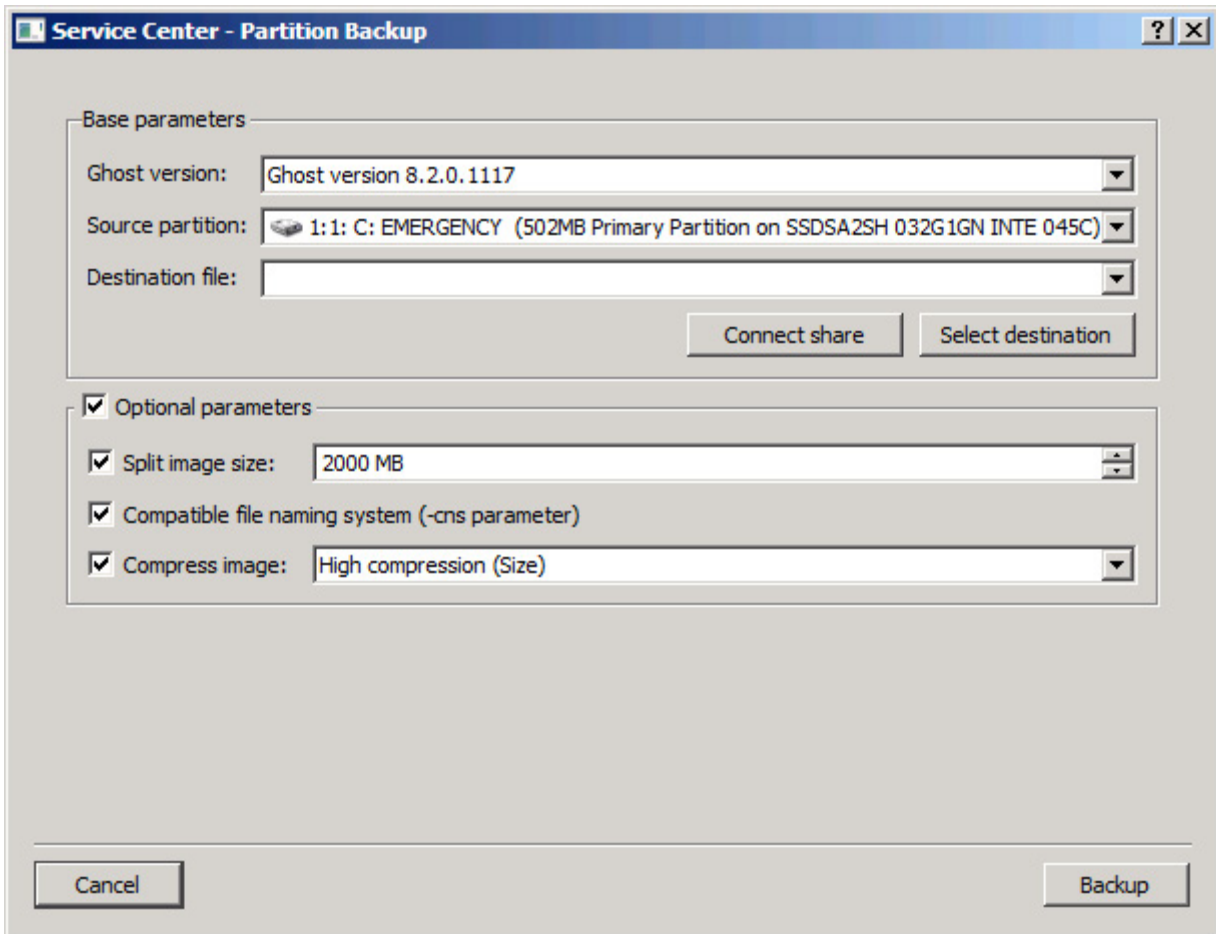


Figure 5-6 Service Center - Partition Backup

3. Make the required settings here:

Section	Settings	Purpose
Base parameters	Ghost version	Select the Symantec Ghost version that is to be used when creating the disk image: • Version for Windows 7: "Ghost version 11.5.x" • Version for Windows XP: "Ghost version 8.2.x"
	Source partition	Select a data medium of which you want to create a data image.

Section	Settings	Purpose
	Destination file	Enter a destination directory and file name. Disk images can be stored either locally or in the network.
Base parameters > Destination file	Connect share	Open the "Connect a Share" dialog box (Page 78). You can set up a shared resource (e.g. a directory in the network), to which you can store the disk image.
	Select destination	Open the "Select destination file" dialog box. You can navigate through a local directory of the PC system and enter a file name.
Optional parameters	Split image size	Split the disk image and select a split image size (PCU 50.5 only). If you are using a PCU 50.5, you can split the disk image into several files of a defined size. Split the disk image size in the following cases: • If the file system (e.g. FAT32) can only manage files of a certain size. • If the disk image is to be archived onto multiple data media, for example, when using CDs. If you are using a different PC system than the PCU 50.5, you must deactivate the "Split image size" setting.
	Compatible file naming system	Select the compatibility of the name scheme. This option is required so that a disk image with older versions of the "PCU base software Windows XP" can be read in by Windows XP. It is not possible to use a different operating system on a PCU 50.x than the preinstalled system (e.g. Windows XP on a PCU 50.5 Windows 7).
	Compress image	Activate Compression or select a compression level. Higher compression results in a smaller file size, but increases the time needed to compress or decompress the data.

4. Confirm your settings with "Backup" to start generation of the disk image.

5.8 Restore a disk image of a partition

If your system is no longer functioning stably, you can restore it with a disk image.

Note

Restoring replaces all current files

When you restore a partition from a disk image, all programs, settings, and files are completely replaced on this partition by the disk image. You cannot restore individual files or exclude individual files from restoration.

Requirement

A disk image of a partition exists
See: Starting the Service Center (Page 65)

Procedure

To restore a partition from a disk image, proceed as follows:

1. Start the Service Center, e.g. via the Windows boot menu.
See also: Create a disk image of a partition (Page 74)
2. In the main menu of the Service Center, click "Partition Restore".

The "Partition Restore" dialog box opens.

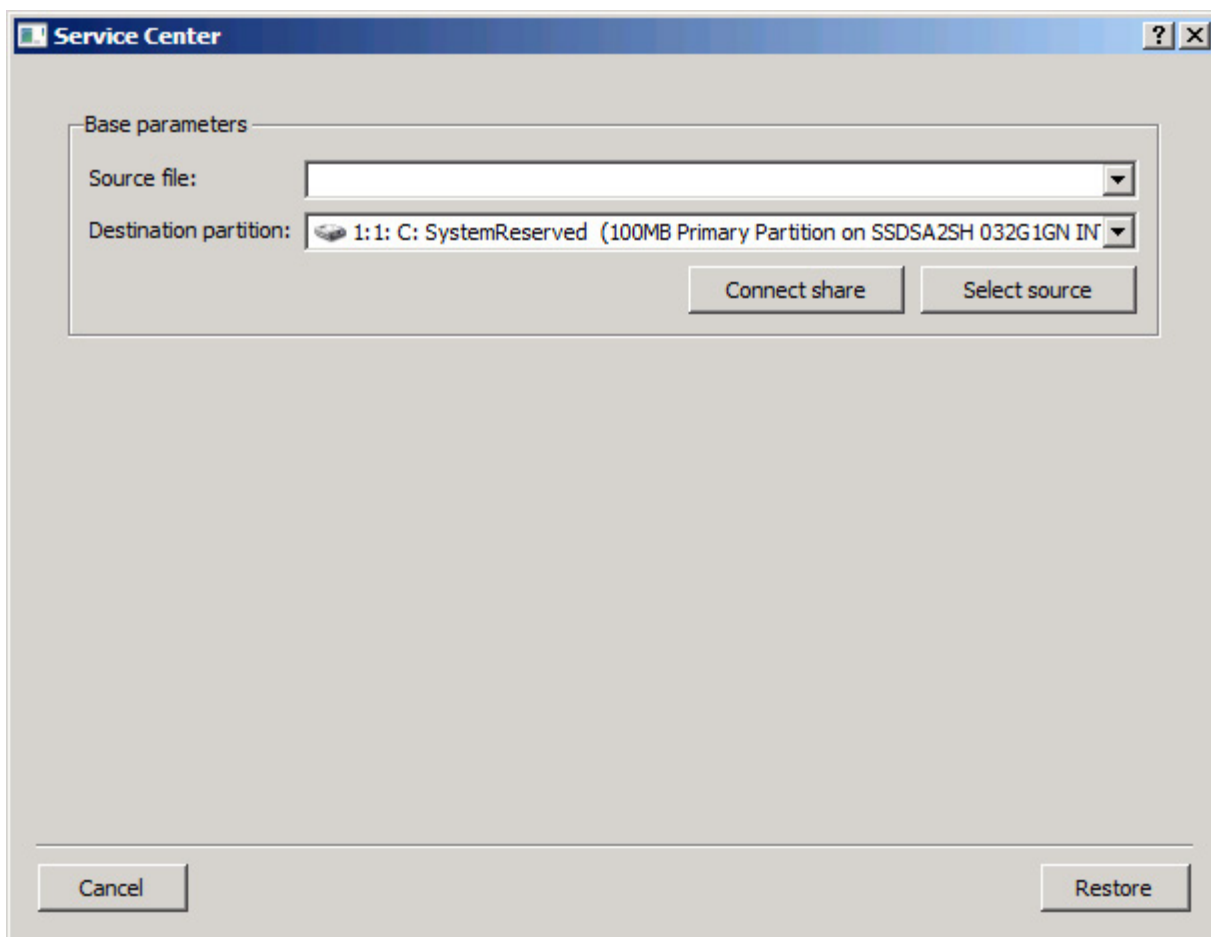


Figure 5-7 Service Center - Disk Restore

3. Make the required settings here:

Section	Settings	Purpose
Base parameters	Source file	Select the disk image from which the partition is to be restored. You can use local disk images or disk images stored in the network.
	Destination partition	Select the partition to be restored from the disk image.
Base parameters > Destination partition	Connect share	Open the "Connect a Share" dialog box (Page 78). You can manage a shared resource (e.g. a directory in the network), to use a disk image stored on the network to restore the SSD.
	Select source	Open the "Select source file" dialog box. You can navigate through a local directory on the PC system and select a disk image for restoration.

4. To start the restore process and entirely replace the existing data on the partition with the disk image, confirm the settings with "Restore".

Result

The restore process is started.

Note

Troubleshooting if reading in of the disk image fails

If data transfer is interrupted during the restore process, the PC system will have no executable system.

In this case, use a bootable USB flash drive with Service Center to read the disk image in again.

5.9 Network settings in the Service Center

To manage a disk image in the network, you can display a shared network directory on the PC system as a drive.

In the Service Center, you configure network drives in the "Connect a share" dialog box. You can call this dialog box directly from the relevant dialog boxes to both create (Page 70) and restore (Page 72) disk images.

Requirement

The network directory is shared with the relevant user accounts in the domain and on the PC system.

See also: Release directory of the PC/PC in the network (Page 88)

Overview

The setting options in the "Connect a share" dialog box have the following meaning:

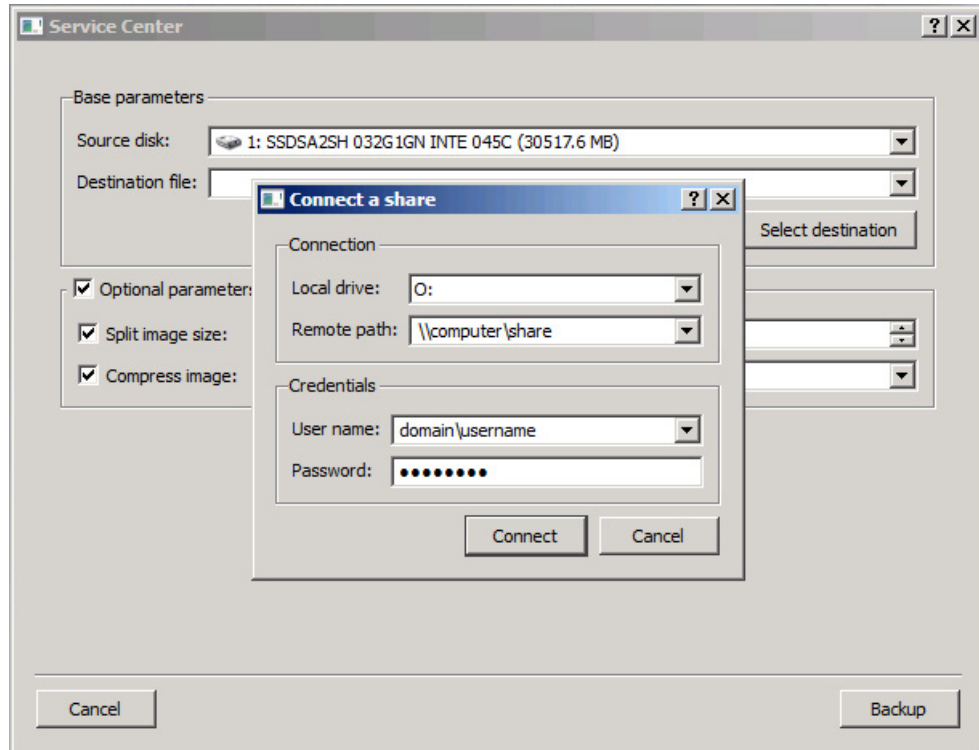


Figure 5-8 "Connect a Share" dialog box

Table 5- 4 Settings in the "Connect a Share" dialog box in the Service Center

Section	Settings	Purpose
Connection	Local drive	To select drive letters under which the network directories are to be accessible on the PC system.
	Remote path	Specify the device in the network and share name of the resource. Syntax: \\Name of the server\share name of the network directory Example: \\Backup_Server\PCU_Backup
Credentials	User name	Enter the user account under which the resource in the network can be accessed. If you have shared the resource on the server for certain users only, you must enter one of these user accounts here.
	Password	Enter the password of the user account.

Further information

You will find information about releasing directories in the network and network drives in the following chapters:

- Release directory of the PC/PC in the network (Page 88)
- Creating a shortcut to the network drive (Page 91)

5.10 Starting Symantec Ghost directly

With the "Symantec Ghost" software you can store the entire contents of an SSD as a disk image. Disk images can be kept on various storage media for subsequent restoration of the data on the SSD.

Symantec Ghost is included in the PCU base software scope of delivery and on the spare part SSD for PCU. Symantec Ghost is used by the components of the PCU base software to back up or restore data.

You can also start Symantec Ghost directly from the prompt to execute the data backup tasks from the user interface of Ghost. This is necessary, for example, when using CompactFlash cards.

Procedure

To start Symantec Ghost directly, proceed as follows:

1. Call the prompt from the Service Center or the Service Desktop.
2. Enter "ghost.32.exe" and confirm the entry.

Further information

More information is available in the Internet on website Ghost.com (<http://www.ghost.com>).

5.11 Bootable USB flash drive

Usage

You can generate a bootable Service Center (Emergency Boot System) based on Microsoft Windows PE on a USB flash drive as preventive measure against a defect.

The Ghost image for the creation of the bootable USB flash drive is on the installed PCU base software at: `D:\Eboot\eboot.gho`

Requirement

- The Windows 7 Service Desktop is active
- You have a USB flash drive (e.g. SIMATIC PC USB flash drive)

Procedure

To create a bootable USD flash drive, proceed as follows:

1. Connect the USB flash drive to a USB interface of the PC system.
2. Start Symantec Ghost, for example, by entering `Ghost32.exe` in the search bar in the start menu.

3. In Symantec Ghost, select the "From Image" command from the "Local > Disk" menu.
4. Select `D:\Eboot\ebboot.gho` as the source and the USB flash drive as the destination.

Result

The Emergency Boot System is installed on the USB flash drive and the data medium is renamed to "EBOOT". The Service Center on the bootable USB flash drive can now be used.

5.12 Operating a service PC/PC in the network

5.12.1 Overview

System network

In the following applications, you will need, for example, a connection between the PC system and a PG/PC in the system network:

- Save the disk image of the SSD to a PG/PC
- Restore the SSD of a PC system via the DVD drive of a PG/PC
- Start up the replacement SSD
- Install the software on the PC system via the DVD drive of a PG/PC

Company network

If you want to connect the service PG/PC via a company network (Eth 1), contact your network service center.

5.12.2 Connection options in the system network

Overview

The following figures show the typical connection options in the system network:

- PCU to "Eth 2" with service PG/PC, directly, using a crossover Ethernet cable
- PCU to "Eth 2" with service PG/PC, via a switch, using a 1:1 Ethernet cable

Table 5- 5 Configuration images: Meaning of the connections

Symbol	Meaning
	Eth 1 as a DHCP client
	Eth 2 with a fixed IP address
	1:1 Ethernet cable
	Crossover Ethernet cable

Configuration with PG/PC directly to PCU

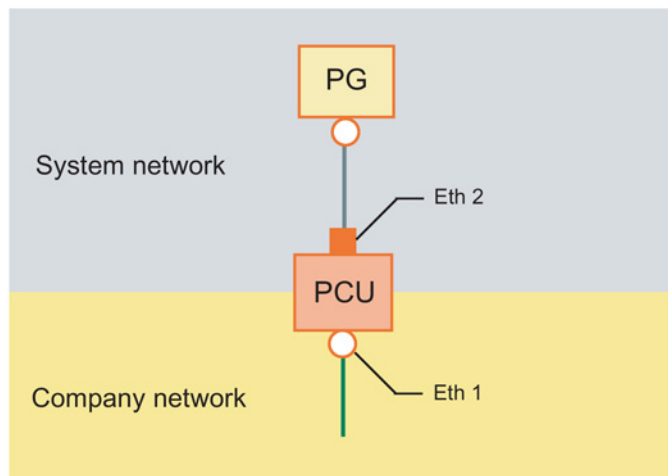


Figure 5-9 Connecting a PG directly to a PCU

Configuration with PG/PC and switch to PCU

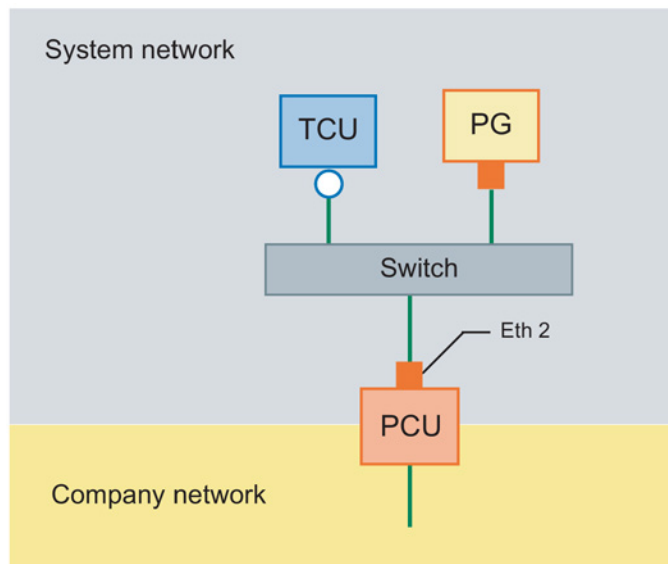


Figure 5-10 Connecting a PG via a switch to a PCU

Note

If a PCU is switched off and on again without its own OP/TP including TCU, and if the PCU boot-up is intended to be performed from the service system (EBS), an external VGA monitor and keyboard are needed to operate the PCU.

An external VGA monitor and keyboard are not needed if the TCUs are operating and the PCU is not switched off with the EBS during booting.

Basic procedure

On the PG/PC with Windows 7:

- Connecting a PG/PC to a PCU according to one of the configurations shown above.
- The network protocol used is: TCP/IP.
TCP/IP is already preconfigured on the PCU.
- Setting up IP addresses on the same subnetwork.
- Releasing a directory on the PG/PC for network access.

On the PCU in the Service Center (WinPE):

- On the PCU start the Service Center (WinPE).
- Establish a network connection to the shared directory of the PG/PC.
See: Functions of the Service Center (Page 66)

5.12.3 Configuring routing in the network

Overview

If you have connected a PG/PC at interface X127 of the NCU, configure the network routing to establish the connection between the PG/PC and the PCU.

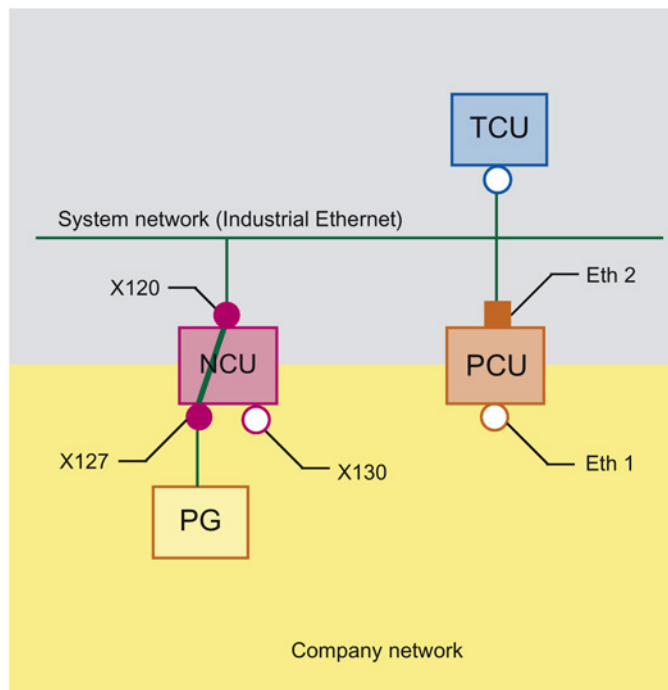


Figure 5-11 A networking example using IP routing

Conditions

The following conditions must be met in order to use this function:

- The PG must be connected to interface X127 of an NCU in the system network.
- The NCU and PCU must be connected via the system network.
- On the NCU, the routing is activated in one of the following ways:
 - In SINUMERIK Operate 4.5 SP2 or higher, the routing is configured in image "Settings of the system network".
 - In the configuration file basesys.ini, key EnableSysNetToIBNFForwarding=1 is set.
- If you wish to access a network directory, it must be shared.
- The user account in question is known to the PG/PC.
- The Service Center is active.

Procedure

To configure the IP routing in the Service Center, proceed as follows:

1. In the main menu of the Service Center, click "Network".
2. In the "Network Settings" dialog box, click the "IP Routing" tab.

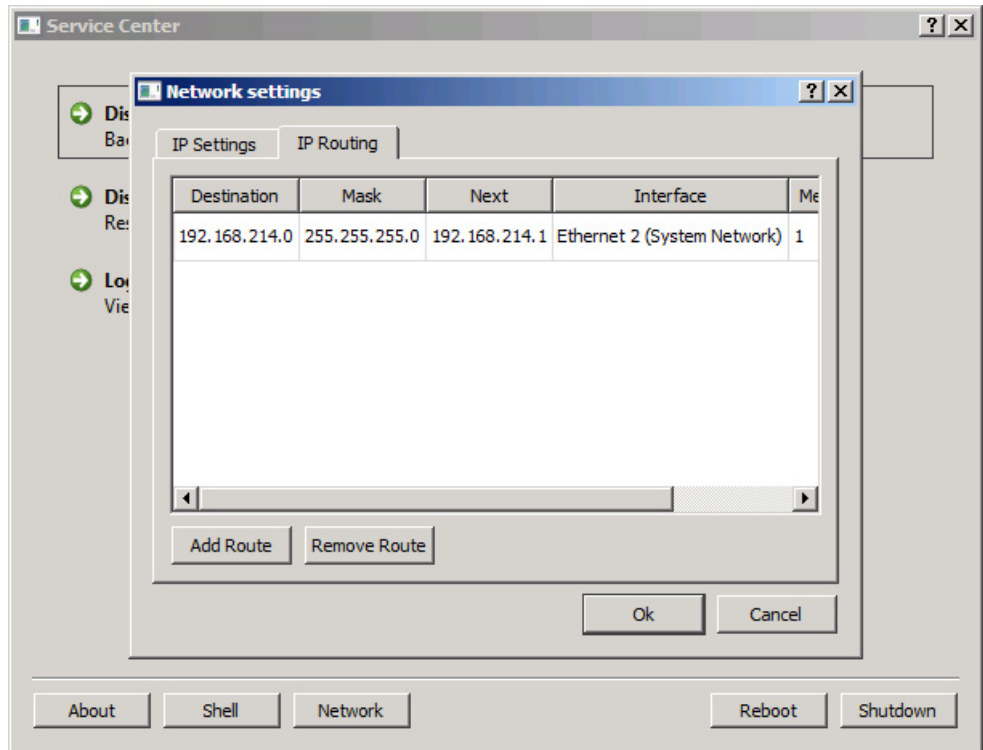


Figure 5-12 Service Center - IP routing

3. Configure the necessary routes. See also: Network settings in the Service Center (Page 67)

Result

The routing has been configured and the PG/PC and PCU are connected via the NCU.

Now, perform the desired steps, for example, a Restore a disk image of the SSD (Page 72).

5.12.4 Configure the network settings of the PG/PC

Once you have connected a PG/PC in the network, make the network settings on the PG/PC.

The basic procedure for setup under Windows 7 or Windows XP is described below. Detailed information from Microsoft is provided in Windows under "Help and Support" or in the Internet: Microsoft Windows support

(<http://windows.microsoft.com/en-US/windows/windows-help#windows=windows-7>)

Precondition

- The PG/PC is networked with the PCU in one of the following ways:
 - The PG/PC is connected to the PCU with an Ethernet cable (either directly or via a switch)
See: Connection options in the system network (Page 81)
 - The PG/PC is networked with the PCU via an interface on the NCU and the routing is configured.
See: Configuring routing in the network (Page 83)
- A current Microsoft Windows operating system is installed on the PG/PC
- The PG/PC is switched on

Procedure

To configure the network settings of the PG/PC for connecting a PCU in the system network, proceed as follows:

1. Open the "network connections" window:
 - To do this under Windows XP: in the "Start > Control Panel > Network and Internet Connections" menu, choose "Network Connections."
 - To do this under Windows 7: in the Control Panel in category "Network and Internet > View network status and tasks," choose "Change adapter settings."
2. Right-click on the network adapter you want to configure, then choose "Properties" from the shortcut menu.
This opens the "Properties of <connection>" dialog box.

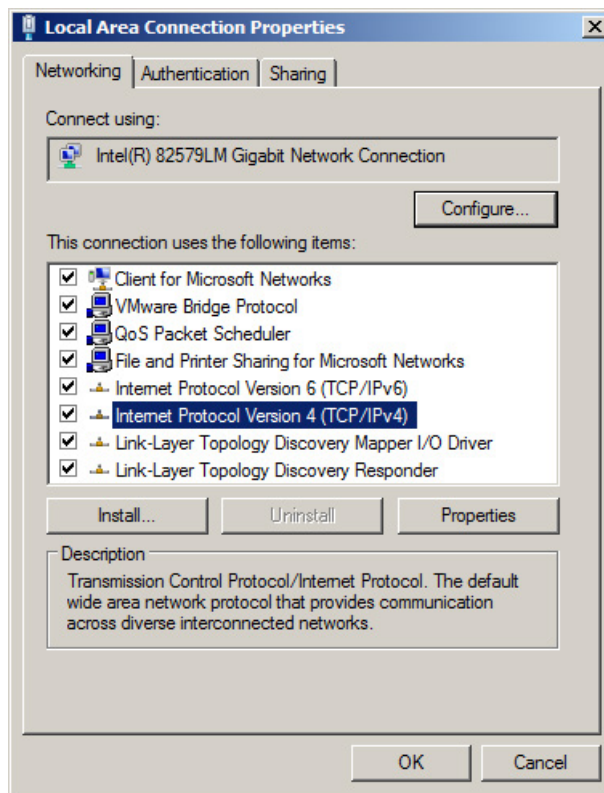


Figure 5-13 Properties of <connection>

3. Make the following settings:
 - Click "File and Printer Sharing for Microsoft Networks."
 - To change the IP address of the PG/PC, double-click "Internet Protocol Version 4 (TCP/IPv4)", then configure the settings as follows:

Section	Element	Settings
Use the following IP address	IP address	The following IP addresses in the system network can be used for the PG/PC: • 192.168.214.250 • 192.168.214.251 • 192.168.214.252 • 192.168.214.253 • 192.168.214.254
	Subnet mask	• 255.255.255.0

Result

The network settings of the PG/PC are now configured for use with the PCU 50.5.

5.12.5 Release directory of the PC/PC in the network

If you want to, for example, store disk images on a PG/PC that you have linked to the PCU via the network, set up a network drive.

As soon as it has been set up, a new drive will appear on the PCU, which is really a directory on the PG/PC.

The basic procedure for setup under Windows 7 and Windows XP is described below. Detailed information from Microsoft is provided in Windows under "Help and Support" or in the Internet: Microsoft Windows support

(<http://windows.microsoft.com/en-US/windows/windows-help#windows=windows-7>)

Requirement

- The PG/PC is switched on.
- There is sufficient space on the PG/PC for saving, for example, disk images from the PCU.
- The user accounts to which you want to grant share rights are known in the domain or on the PCU.

Procedure

Proceed as follows to share a directory for use in the network:

1. On the PG/PC, in the Control Panel call up Computer Management.
2. Under "System > Shared Folders" select the file "Shares."
3. In the "Action" menu, click the "New File Share..." command.
The "Create A Shared Folder Wizard" opens.

4. Under step "Name, Description, and Settings," make a note of the Share path, which is a combination of the computer name and the share name. You will have to specify this name when you create a link to the network drive on the PCU.

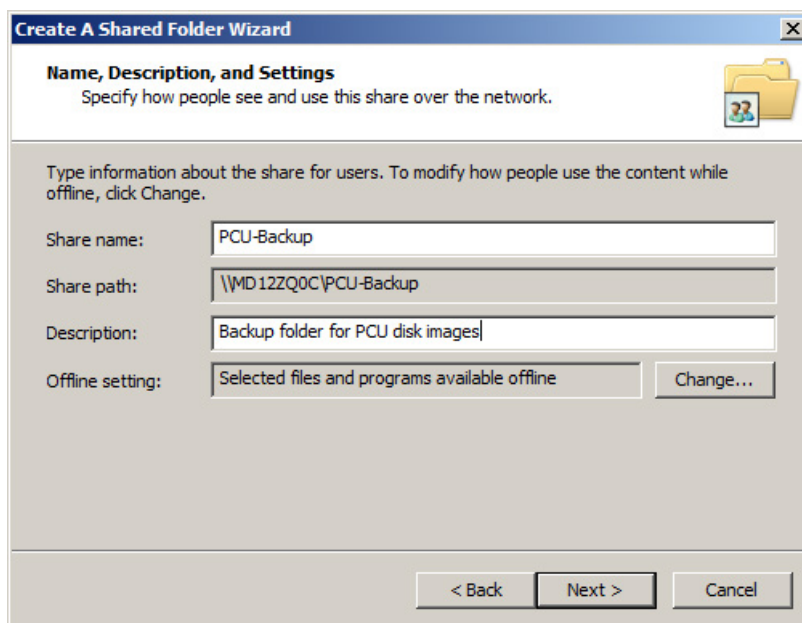


Figure 5-14 Create A Shared Folder Wizard - Name, Description, and Settings

5. Under step "Shared Folder Authorizations," select the "Customize authorizations" radio button, then click "Custom..."



Figure 5-15 Create a Shared Folder Wizard - Shared Folder Authorizations

The "Customize Authorizations" dialog box opens.

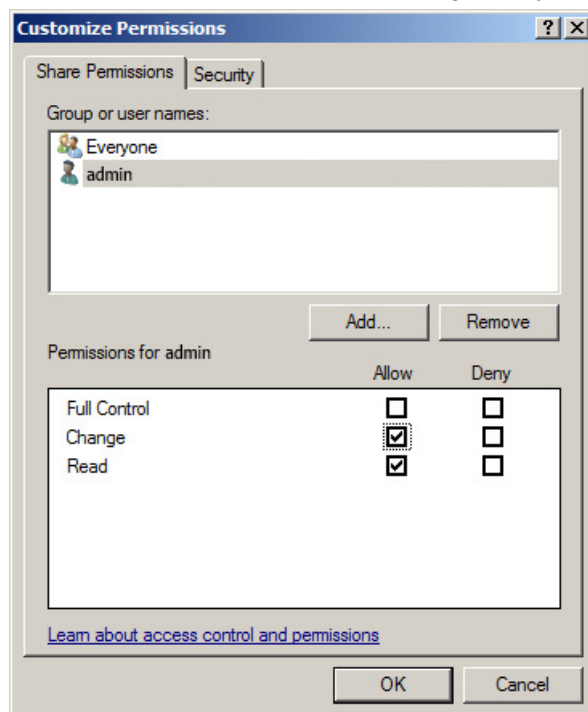


Figure 5-16 "Customize Authorizations" dialog box

6. In section "Group or user names," click "Add..."
The "Select Users, Computers, Service Accounts, or Groups" opens.

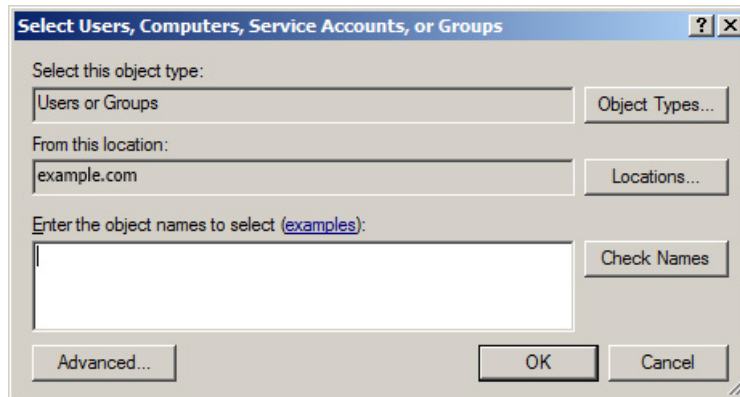


Figure 5-17 "Select Users, Computers, Service Accounts, or Groups" dialog box

7. In section "Authorizations for <user name>," under column "Allow," click the "Change" checkbox.
8. Confirm these settings in the "Customize PermiAuthorizations" dialog box with "OK."
9. In the next steps of the wizard, make all the settings you require and confirm the share with "Finish".

5.12.6 Creating a shortcut to the network drive

To be able to access the shared directory on the PG/PC from the PC system, after you have connected and configured the PG/PC in the network, you must set up the folder shared on the PG/PC on the PC system as a network drive.

As soon as it has been set up, a new drive will appear on the PC system, which is really a directory on the PG/PC.

The basic procedure for setup under Windows 7 and Windows XP is described below. Detailed information from Microsoft is provided in Windows under "Help and Support" or in the Internet: Creating a shortcut to a network drive (association)
(<http://windows.microsoft.com/en-us/windows7/create-a-shortcut-to-map-a-network-drive>)

Requirement

- The PG/PC is switched on.
- The Service Desktop is active.
- The folder on the PG/PC is shared for the relevant user accounts on the domain and on the PC system.

Procedure

Proceed as follows to connect a shared directory as a network drive:

1. Start the Windows Explorer and click the "Map network drive" button.
2. Make the following settings in the "Map network drive" dialog box:

Settings	Purpose
Drive	To select drive letters under which the network directories are to be accessible on the PC system.
Folder	Specify the device in the network and share name of the resource. Syntax: \\Name of the server\share name of the network directory Example: \\Backup_Server\PCU_Backup
Reconnect at logon	Select this checkbox to automatically connect the network drive on booting.
Connect using different credentials	Select this checkbox if you have not shared the directory in the network with the user for whom you would like to set up the network drive. In the next dialog box "Enter Network Password," enter the log-on information of a user with whom the network directory is shared.

3. Confirm your settings with "Finish."

See also

Microsoft Windows support

(<http://windows.microsoft.com/en-US/windows/windows-help#windows=windows-7>)

5.13 Starting up the replacement SSD

Information on the replacement or installation of an SSD can be found in the hardware documentation for your PC system:

- *Operator Components and Networking Manual*

Note

Troubleshooting if reading in of the disk image fails

If reading in of the data is interrupted, the PCU will have no executable system.

In this case, use a bootable USB flash drive with Service Center to read the disk image in again.

See also

Network settings (Page 8)

Restore a disk image of the SSD (Page 72)

Service and diagnostics

6.1 PCU Hardware Diagnostics

Intended use

The PCU hardware supports the diagnostics of important system components via an integrated "safecard", which is designated as a Safecard-On-Motherboard (SOM).

Monitored Parameters

The following physical parameters of the PCU hardware are monitored:

- CPU temperature
- Housing temperature
- I/O chip temperature
- Speed of the two housing fans
- S.M.A.R.T - status of the hard disk

Logging faults

The PCU hardware monitor logs all hardware faults in the Windows event log so that the faults can be output even without installing an HMI software.

The alarms are output in the log under "Control Panel" → "Administrative Tools" → "Event Viewer".

See also

- Starting up the replacement SSD (Page 93)
- Operator Components and Networking Manual: Chapter, spare parts
- Description of the alarms: SINUMERIK 840D sl Diagnostics Manual

6.2 Evaluation of the 7-segment display

Application

The 7-segment display is intended for PCU diagnostics in operation without a local OP (headless mode). If an error is detected during startup, a local display is needed for subsequent error analysis of the connection.

The two displays have the following function:

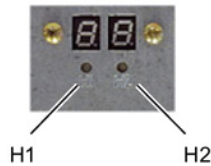


Figure 6-1 7-segment display on the PCU 50.5

- Segment display H1

Segment display H1 and the associated LED are assigned to the PCU base software. Status codes are output during system startup, normal operations and shutdown. Compatibility in relation to the NCU modules is taken into account.

- Segment display H2

Segment display H2 and the associated LED are assigned to the application software.

Meaning of status codes

- During system startup:

LED H1	LED H2	7-segment display		Status	
Orange	Orange	Output of BIOS post codes		After switching on the system	
Orange	Off			After cycling the BIOS	
Orange	Off			After starting Windows	
		1	0	Start Windows	Load device drivers needed for the Windows start
		2	0	PCU hardware service	The PCU hardware service has been started
		5	0	Network	Wait for network interfaces to be ready
		8	0	TCU support test step 1	Wait for FTP server to start
		9	0	TCU support test step 2	Wait for boot server to start for TCU network boot and for TCU hardware service to start
		A	0	TCU support test step 3	Wait for VNC server to start
		B	0	-	-
		E	0	PCU base software	General fault: see Windows "Event Viewer"

- During operation:

LED H1	LED H2	7-segment display		Status	
Flashing green	--	0	0	OK	
Flashing red	--			Error:	
		1	0	Temperature alarm	Housing or CPU temperature; temperature above limit value (SOM error)
		2	0	Fan alarm	Housing or CPU: Low fan speed or fan failure (SOM error)
		3	0	Hard disk alarm	S.M.A.R.T hard disk error
		6	0		Failure of VNC server or VNC server service stopped

- During shutdown

LED H1	LED H2	7-segment display		Status
Flashing red/green	--	5	0	Shutdown active

6.3 Setting of the operating mode during run-up

6.3.1 Service switch (PCU 50.5)

The service switch on the PCU 50.5 is a rotary switch with 16 possible positions.

The "PCU Hardware Service" component as part of the PCU base software provides a function to read-out the switch positions.

The switch position on the hardware can be used to specify the operating mode for the run-up of the PCU 50.5.

The meaning of the switch positions is described at Possible operating modes (Page 99).



Figure 6-2 Service switch

Further information

Details about the hardware of the PCU 50.5 are provided in the *SINUMERIK 840D sl Operator Components and Networking Manual*, Chapter *Operator panels*.

6.3.2 Possible operating modes

Only the settings of the service switch that are relevant for the commissioning engineer are listed:

Switch setting	Function
0	Normal mode (switch setting in delivery condition)
3	Desktop operation: Run-up in the "SINUMERIK" splash screen which enables the desktop selection. Relevant if HMI software is installed with autologon mode set. See also: Switching to the Service Desktop during autostart / autologon operation (Page 99).
9	Reset BIOS to factory settings
E	Boot device at the USB interface, e.g. USB flash drive with emergency boot system See also: Bootable USB flash drive (Page 80)

6.4 Switching to the Service Desktop during autostart / autologon operation

If you have set up the autologon operation and the autostart of SINUMERIK Operate with the activated keyboard filter, the PC system runs up with SINUMERIK Operate after being switched on.

In order to be able to switch to the Windows Desktop in this case, you must interrupt the run-up in SINUMERIK Operate with a keyboard entry (key 3) as described here.

Requirement

- SINUMERIK Operate is installed and autologon mode is set up.
See Chapter Setting up SINUMERIK Operate for autologon mode (Page 44).
- The PCU is switched off.

Procedure

In order to switch to the Desktop during activated autostart / autologon operation, proceed as follows:

1. Switch on the PCU and wait until the "SINUMERIK" welcome screen appears.
2. Press key 3 on the keyboard while the "SINUMERIK" welcome screen is being displayed. The counter for starting SINUMERIK Operate is suspended and you can select one of three options:

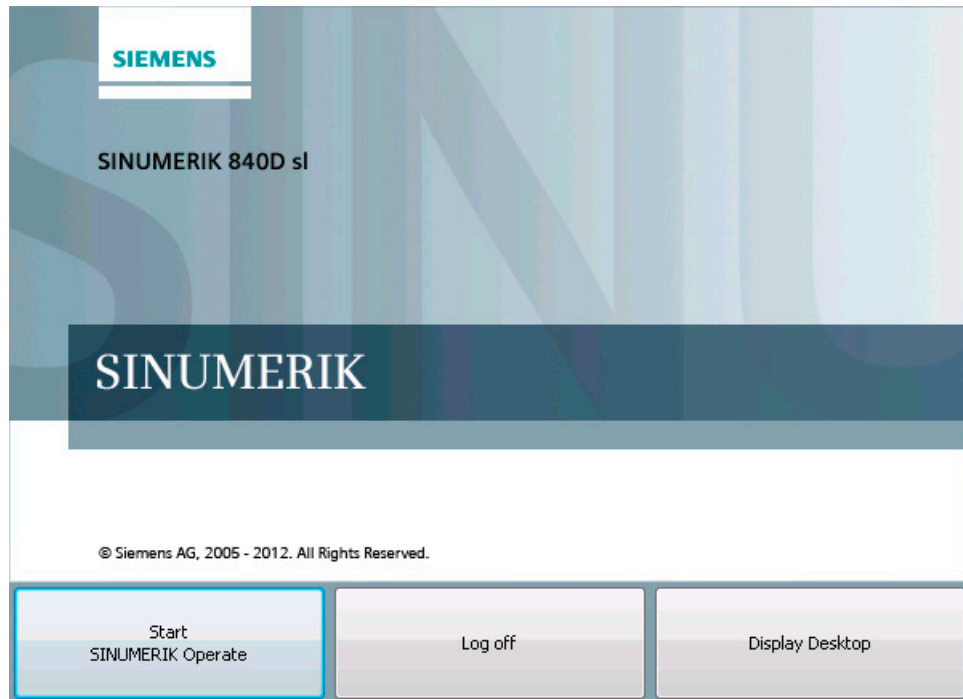


Figure 6-3 Welcome screen "SINUMERIK" after key input

3. Select the "Display Desktop" option.

Result

The Service Desktop opens.

6.5 Remote access

6.5.1 Overview

You can use a remote connection to access other devices via the network:

- Access to maintenance with prompt (SSH) via PuTTY.
- Access via VNC Viewer with all its operator and maintenance options, which are also available directly on the PCU.

See also

Setting up and using SSH (Page 103)

6.5.2 Searching for devices in the system network

The "sc_show_net" command shows devices available in the system network in the prompt and offers additional information about these devices.

As both these functions rely on SNMP (Simple Network Management Protocol), only SNMP-capable devices will be found. In the case of Windows-based devices, this depends on the software version. If a simple call is made, without any additional options, a list of the devices found will appear showing the relevant IP address, DNS name (if known), and a short description (module name).

sc_show_net command

This command is executed in the prompt:

Syntax: `sc_show_net [-xml] [-hw|-tco|-sw|-swfull|-loc|-panel] -dhcp|-switch|-all] [HOSTS...]`

Authorization level: None

6.5.3 Display of accessible stations in SINUMERIK Operate

You can use the network diagnostics function in SINUMERIK Operate to show accessible nodes in the system network.

Procedure

Proceed as follows to show accessible nodes:

1. Select "MENU SELECT" to open the menu.
2. In the horizontal softkey menu, choose "Diagnostics > Bus TCP/IP."
3. In the vertical softkey menu, choose "TCP/IP > Diagnostics Network > Accessible nodes." The "Accessible nodes" screen opens showing all nodes that use the SNMP or DCP protocol.

Further information

You will find Information about showing accessible nodes in SINUMERIK Operate in the *SINUMERIK 840D sl SINUMERIK Operate (IM9) Commissioning Manual, Section Service and diagnostics*.

6.5.4 Remote access to the PCU for operation and maintenance

Using the VNC Viewer, you can connect to devices by remote access that have an active VNC server, e.g. PCU or NCU.

The factory settings only enable you to monitor a system (e.g. PCU) via an external VNC Viewer.

To operate the system from another station, the system in question must grant permission. You make the necessary settings in the **tcu.ini** in section [VNCViewer].

Important factory settings

- To access the PCU by remote access, you require a password. The factory setting of the password in the VNC server on the PCU 50.5 is "password". Modify this password during commissioning in the settings of the VNC Server to avoid security risks.
- To be able to access a PCU via the enterprise network with the VNC Viewer, port 5900 must be enabled under "Control Panel > Windows Firewall > Exceptions." This port does not have to be enabled to access the PCU 50.5 via the plant network.

Further information

Operator Components and Networking Manual

Options for the VNC viewer

Note

Do not change the factory settings!

To ensure that the VNC viewer works properly, the following options must not be changed.

After starting the VNC viewer, the following dialog opens:

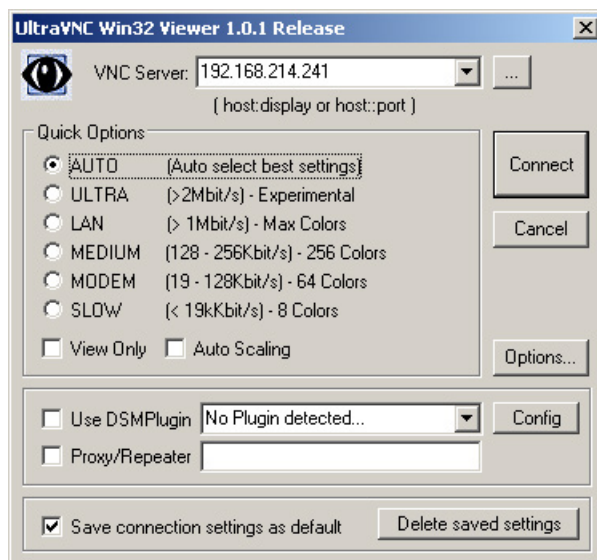


Figure 6-4 VNC Viewer - Factory Settings

After clicking the "Options ..." button, the following dialog opens:

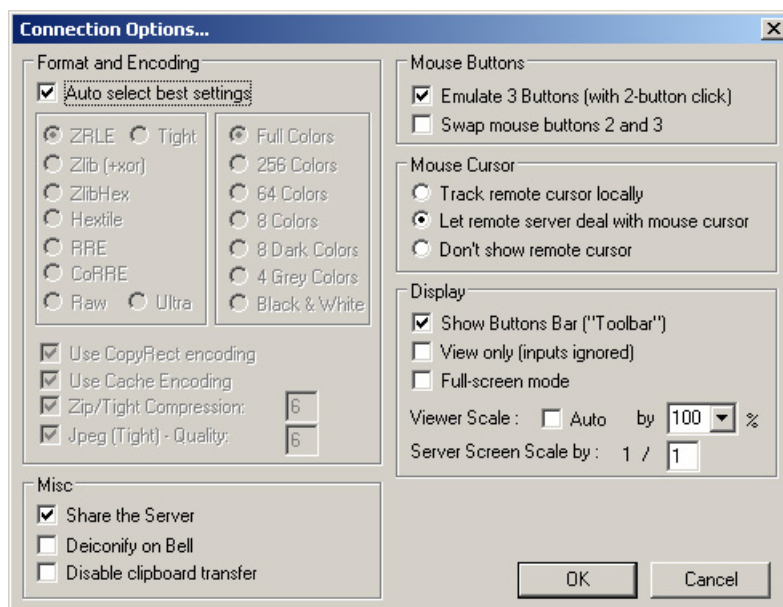


Figure 6-5 VNC Viewer > Options - Factory Settings

6.5.5 Setting up and using SSH

You can use remote access to access devices from the PCU.

An SSH environment is preinstalled when the PCU base software for Windows 7 is installed. You can log on with PuTTY.

You can use an SSH key to log on. A script for setting up the infrastructure on devices is provided. However, for security reasons no previously generated keys have been stored.

Setting up remote access with SSH and key pair

The procedure for setting up users and using remote access is the same as the usual procedure for remote access with SSH.

Table 6- 1 Support with setting up remote access

Task	Recommended application	Storage path	Help
Generate key pair	PuTTY Key Generator	C:\Program Files (x86)\PuTTY\puttygen.exe	Online help in PuTTY (Chapter <i>Public key for pasting into authorized_keys file</i>)
Set up infrastructure on devices	Supplied script ssh_key_login.bat	C:\ProgramData\Siemens\Motion Control\siemens\etc.\ssh_key_login.bat	Call up the script via the prompt. To do this, enter user name and file name of the public key as the parameters. Syntax: ssh_key_login.bat <user name> <public key> Example: ssh_key_login.bat username publicKey.pub
Access devices via remote connection	PuTTY	C:\Program Files (x86)\PuTTY\putty.exe	Online help in PuTTY IP address of the PCU (factory setting): • System network: 192.168.214.241 • Company network: IP address is taken from the DHCP server

See also

Remote access (Page 101)

Encryption via SSH protocol (Page 104)

6.5.6 Encryption via SSH protocol

Security

The security of SSH is ensured by a series of cryptographic algorithms for encryption and authentication.

Authentication

The server identifies itself to the client with an RSA, DSA or ECDSA certificate, which enables detection of manipulation in the network (no other server can identify itself as a known server).

The client can authenticate itself either by public key authentication with a private key whose public key is stored on the server, or with a normal password. While in the latter case, user interaction is always necessary (unless the password is stored unencrypted on the client computer), public key authentication allows client computers to log on to SSH servers without user interaction, without a password having to be stored in plain text on the client. However, for added security the private SSH keys can also be protected with a password.

Subsystems

In the case of Secure Subsystem Execution, subsystems that were defined in an SSH server installation can be executed remotely without knowing the precise path of the program executed on the server. SFTP is the most common subsystem.

Further information

- Chapter Setting up and using SSH (Page 103).
- "NCU Operating System" Commissioning Manual (IM7).
- Description of PuTTY.

Index

7

7-segment display, 95

A

Answer file, 15
Autologon, 15
 Switch to Service Desktop, 99
Autologon mode, 45
AutoRepeat, 47

B

Backup time, 36
Batch file
 USVShutdown.bat, 30, 31, 35
Batch processing file, (Batch file)
BIOS
 BIOS post code, 96
 Factory setting, 99
Boot stick, 80

C

Color depth, 28
Command, (Command)
 control userpasswords2, 46
 sc_show_net, 101
 sc_usb disable, 23
 sc_usb enable, 23
Command line, (Prompt)
Commissioning
 unattended, 15
 USB flash drive, 15
 with identical settings, 15
Configuration file
 Archive, 9
 basesys.ini, 84
 ghost.ini, 39
 oemframe.ini, 51
 pcuhwsvc.ini, 47
 PCUInst.ini, 53, 54, 55, 56
 servicesystem.ini, 39, 40, 42

tcu.ini, 25, 25, 25, 28, 102

Templates, 9

D

DHCP, 8
Dialog box
 netplwiz, 45
 ServiceCenter Backup/Restore, 39, 40, 42
Directory
 C:\ProgramData, 9
 D:\Install, 15
Disk image
 Restoring, 72, 76
 Troubleshooting, 74, 78
DOS shell, (Prompt)

E

Emergency Boot System, 80, 99

F

FirstLogonCommands, 15

H

Hard disk
 Replacing, 93
HMI software, 51

I

Initialization files, (Configuration file)
Input area scheme, (Keyboard layout), (Keyboard layout)
Installation, (See Commissioning)
Interface
 Ethernet, 8
 X1, 8
 X2, 8

K

- Key, (Repeat function)
 - Key filter, 47
- Key filter, 47
- Keyboard layout, 24, 44
- Kiosk mode, (Autologon mode; key filter)

L

- Language
 - Configuring the keyboard layout, (Keyboard layout), (Keyboard layout)
- Language bar, (Keyboard layout), (Keyboard layout)
- Log file
 - PCUHardwareservice.log, 49
 - PCUInstaller_C.log, 60, 61, 62, 63
 - PCUInstaller_S.log, 60, 61, 62, 63
 - Setup-specific, 62
- Log File, (Log file), (Log file)

M

- Mode of operation
 - Headless mode, 15

N

- Network
 - Adapting settings, 22, 40, 42
 - Company network, 8
 - Domain, 22
 - IP address, 22
 - System network, 8
- Network protocol
 - SNMP, (SNMP)
- Network release
 - Driven on PG/PC, 88

O

- OEM
 - Configuration, 51
- Operating mode, 99
 - Autologon mode, (Autologon mode)
 - Service switch, 97

P

- Password, (Password)
 - Administrator account, 15, 21
 - Authentication for SSH encryption, 105
 - Configure for VNC server, 102
 - For network drive, 78
 - For unattended installation, 15
 - Password reset diskette, 21
 - unattend.xml, 15
- PCU
 - Checking the hard disk, 95
- PCU Installer
 - Activating, 53, 54
 - Configuring, 55, 56
 - Log file, (Log file), (Log file)
 - PCUInst.ini, 56
 - Troubleshooting, 54, 61
 - Using, 60
- Port 80 display, (See 7-segment display)
- Power-up
 - Operating mode during run-up, 99
- Prompt
 - Open, 24

R

- Remote access, 102
 - Via VNC server, 102
 - With SSH encryption, 103
- Response file
 - unattend.xml, 19

S

- Screen resolution, 25, 25
- Service Desktop, 99
- Service switch, 97, 99
- Service system for PCU, 80
- Shutdown, 29, 97
- SINUMERIK Operate
 - Unattended installation, 15
- SITOP UPS, 29
- Softkey, 51
- SSD
 - Replacing, 93
- SSH encryption, 103, 105
- Startup
 - Initial startup, 15
- STEP 7, 51
- System startup, 96

T

- Troubleshooting
 - PCU Installer, 54, 61
 - Restore failure, 74, 78

U

- Uninterruptible power supply, 29
- USB FlashDrive, 80
- USB interface
 - Activating, 23
 - Disabling, 23
- User account, (Setting up for autologon mode)
 - Setting up, 21, 21
 - Setting up for autologon mode, 44
- User administration, 21

V

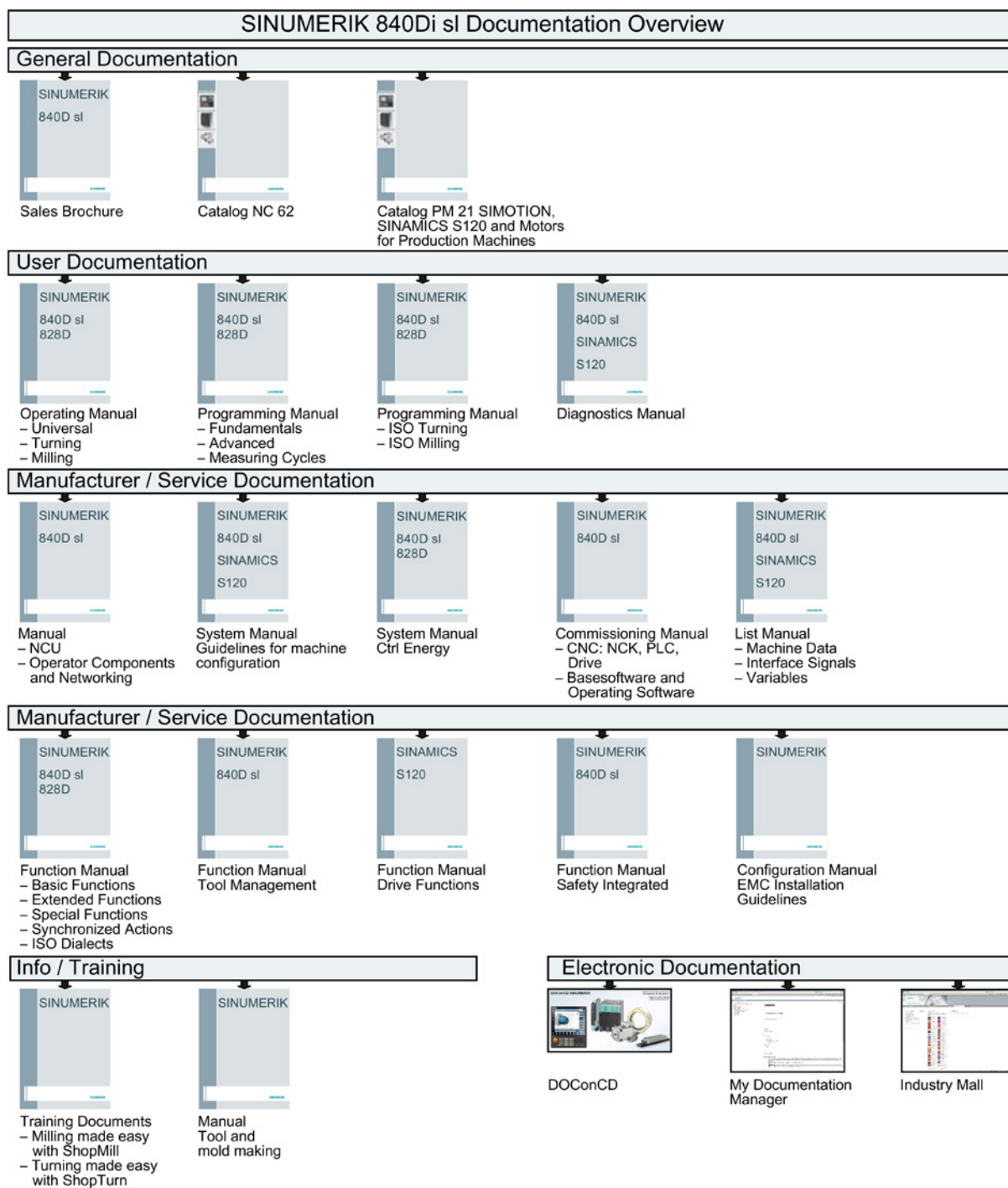
- VNC Viewer, 102

X

- XML
 - Response file, 15
 - unattend.xml, 15
- XML file
 - unattend.xml, 15, 19

Appendix

A.1 Overview



Basesoftware and operating software

Commissioning Manual, 01/2015, 6FC5397-1DP40-5BA2