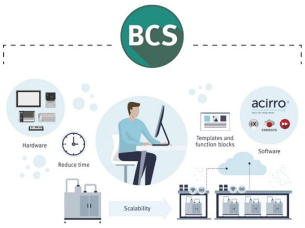
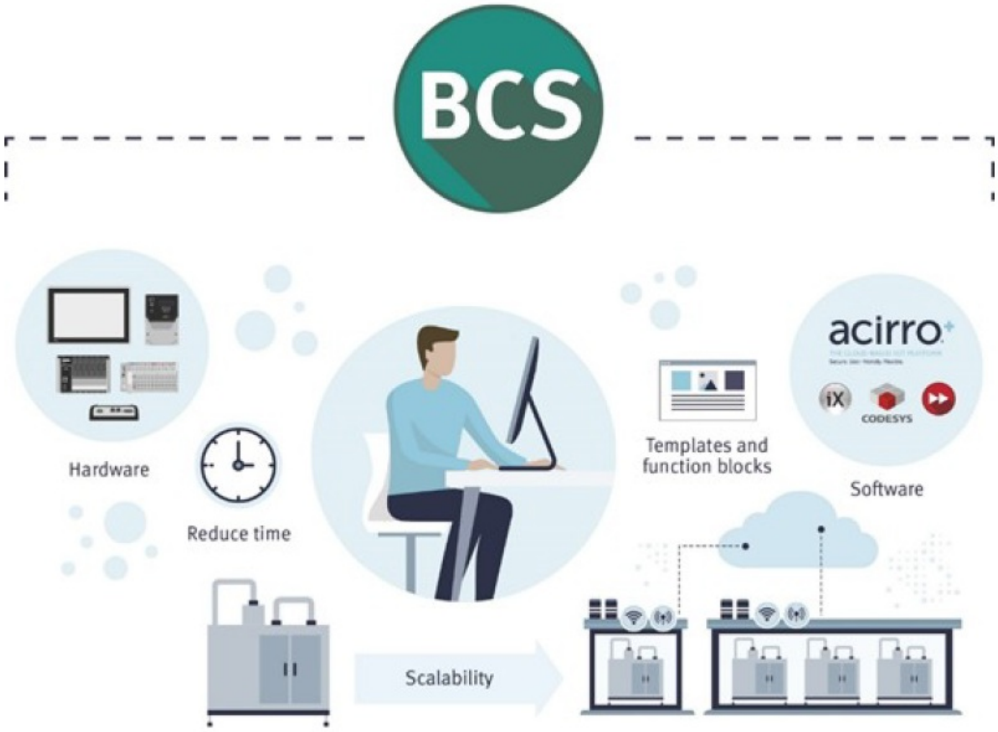


# Beijer ELECTRONICS SER0015 X2 Control Communication User Guide

Contents

# Beijer ELECTRONICS SER0015 X2 Control Communication



## Function and area of use

This document provides guidelines when working with X2/BoX2 control as Modbus RTU master in CODESYS or BCS Tools, communicating with Beijer Frequency Inverter, BFI-E3, BFI-P2 or BFI-H3.

- Note, it’s not possible to run CANopen or any other bus at the same time as Modbus RTU.
- Note, this project contains four BFI of any type.

## About this document

This quick start document should not be considered as a complete manual. It is an aid to be able to start up a normal application quickly and easily.

Use the following hardware, software, drivers and utilities in order to obtain a stable application:

**In this document, we have used the following software and hardware**

**Software:**

- BCS Tools 3.33 / CODESYS V3.5 SP13 Patch3
- iX Developer 2.40 SP5 / SP6
- BFI-Tools 2.8.0.0

**Hardware:**

- X2 control (marine/extreme) or BoX2 pro SC/motion
- BFI-P2 and BFI-H3 with at least firmware 2.40
- BFI-E3 with at least firmware 3.00

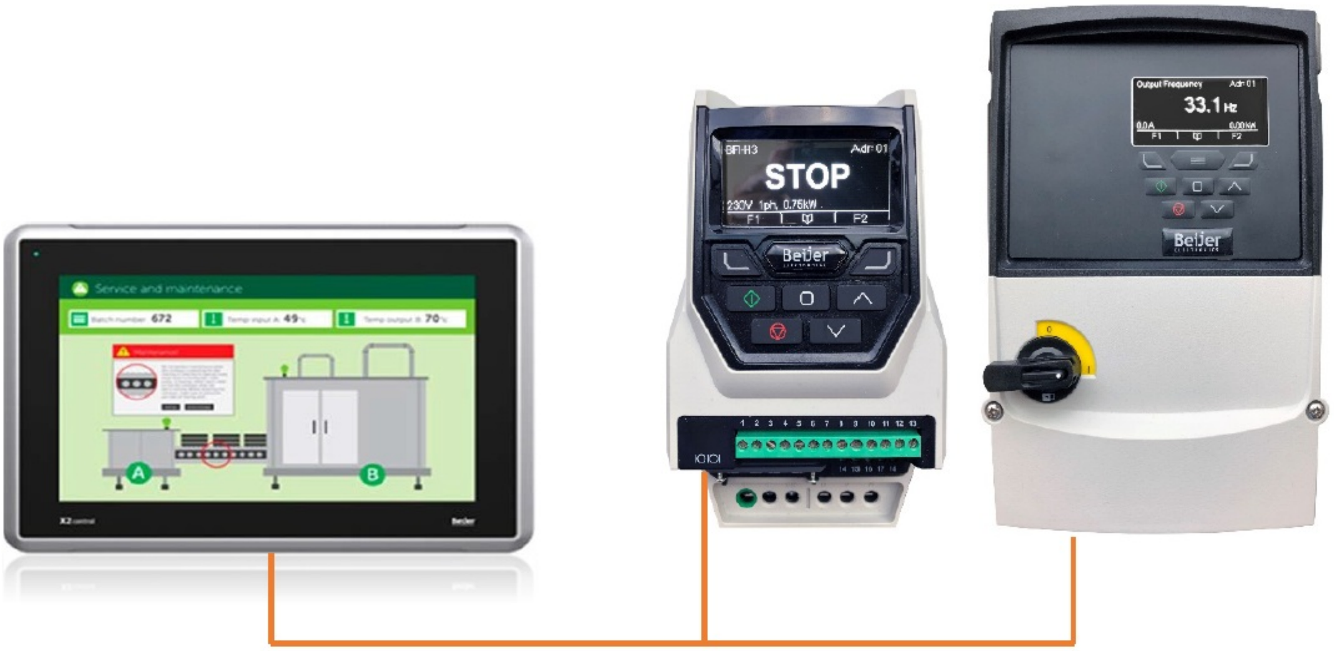
**For further information refer to**

- BFI-E3 / BFI-P2 / BFI-H3 User Guides
- BFI-P2 Start-Up Manual KI00306C
- BFI-H3 Start-Up Manual KI00363D
- BFI-E3 Start-Up Manual KI00369B
- Installation Manual X2 pro, X2 marine, X2 control
- Beijer Electronics knowledge database, HelpOnline

This document and other quick start documents can be obtained from our homepage. Please use the address support.europe@beijerelectronics.com for feedback.

## System set up

Here comes a short description of what needs to be done and tools to be used to make X2 control work together with one or several BFI-P2/E3/H3 in a Modbus RTU network:



1. Set up of Station number address and communication speed in each BFI
2. Either set them up on the display of BFI, PC software BFI-Tools or a device with BFI-Tools Mobile. See chapter 7.7
3. Set up other parameters in BFI-E3/P2/H3

Either set them up on the display of BFI-E3, PC software BFI-Tools or a device with BFI-Tools Mobile. See chapter 7.8

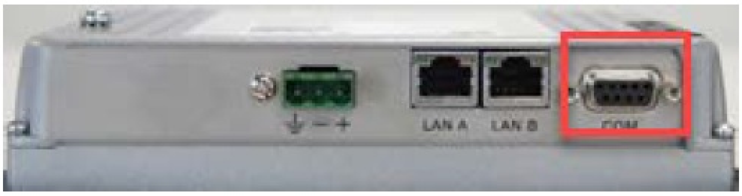
4. Software enable needs to be activated by a jumper between terminal 1 and 2 of BFI
5. If BFI-P2 or BFI-H3: STO inputs need to be properly connected. Either to a Safety system or jumpers between terminal 1 to 12 and terminal 9 to 13 of BFI- P2 and BFI-H3.
6. Set up the IP address of X2 control. See chapter 5.2
7. Connect the serial cable from X2 control, CAB113, to serial port of BFI
8. Termination of network
   CAB113 has built-in 120-ohm termination resistor. Make sure that the slave in the other end of network has another 120-ohm resistance. For BFI, see chapter 5.3.
9. Download the PLC project to X2 control. Adjust the device "X2 control" to fit the used hardware. Size and Powerfail or cyclic must be correct. Default is X2 control 4 Power Fail.
   Use BCS Tools or CODESYS for the PLC project.
10. Download the iX-project to X2 control. Set up type of HMI in project/settings. Default is X2 control 4. Use iX Developer.

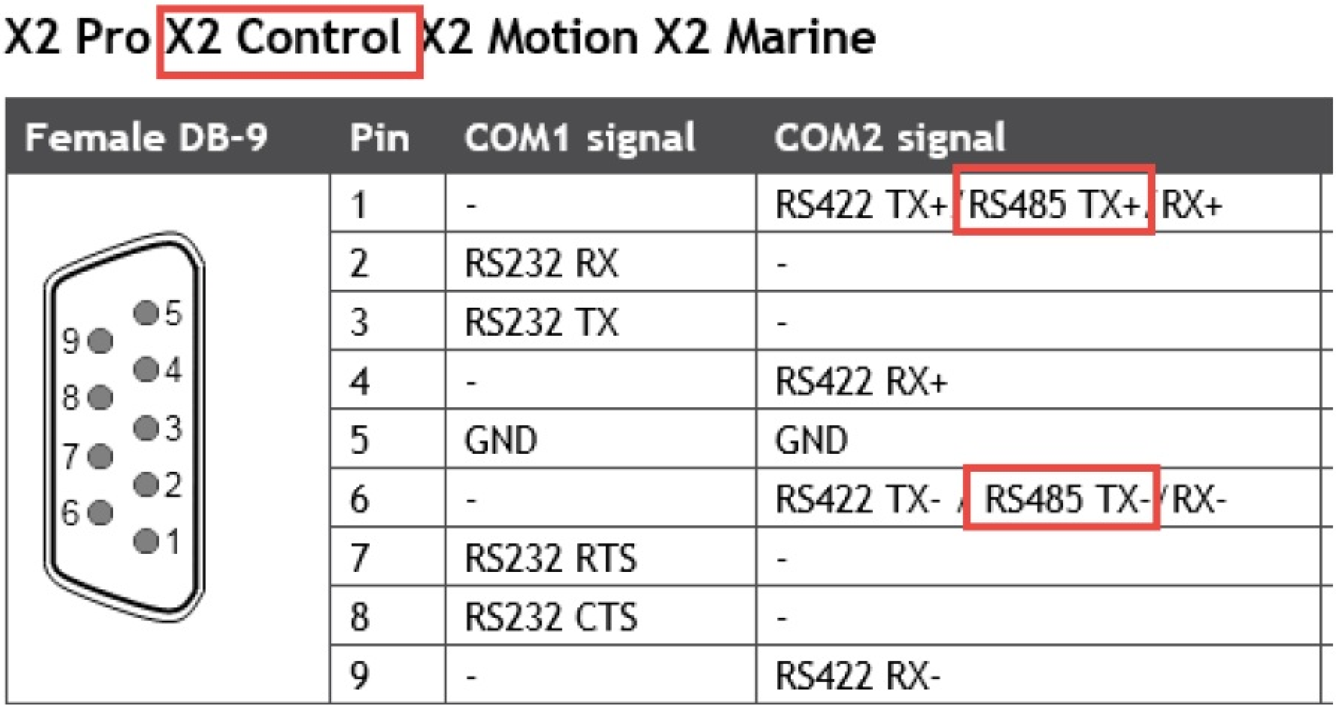
## Hardware

The following chapter describes the Modbus RTU connections of X2 control and BFI-P2/E3/H3.

### Modbus RTU of X2 control

The program example uses the RS485 port called COM2 on pin1 and pin6.



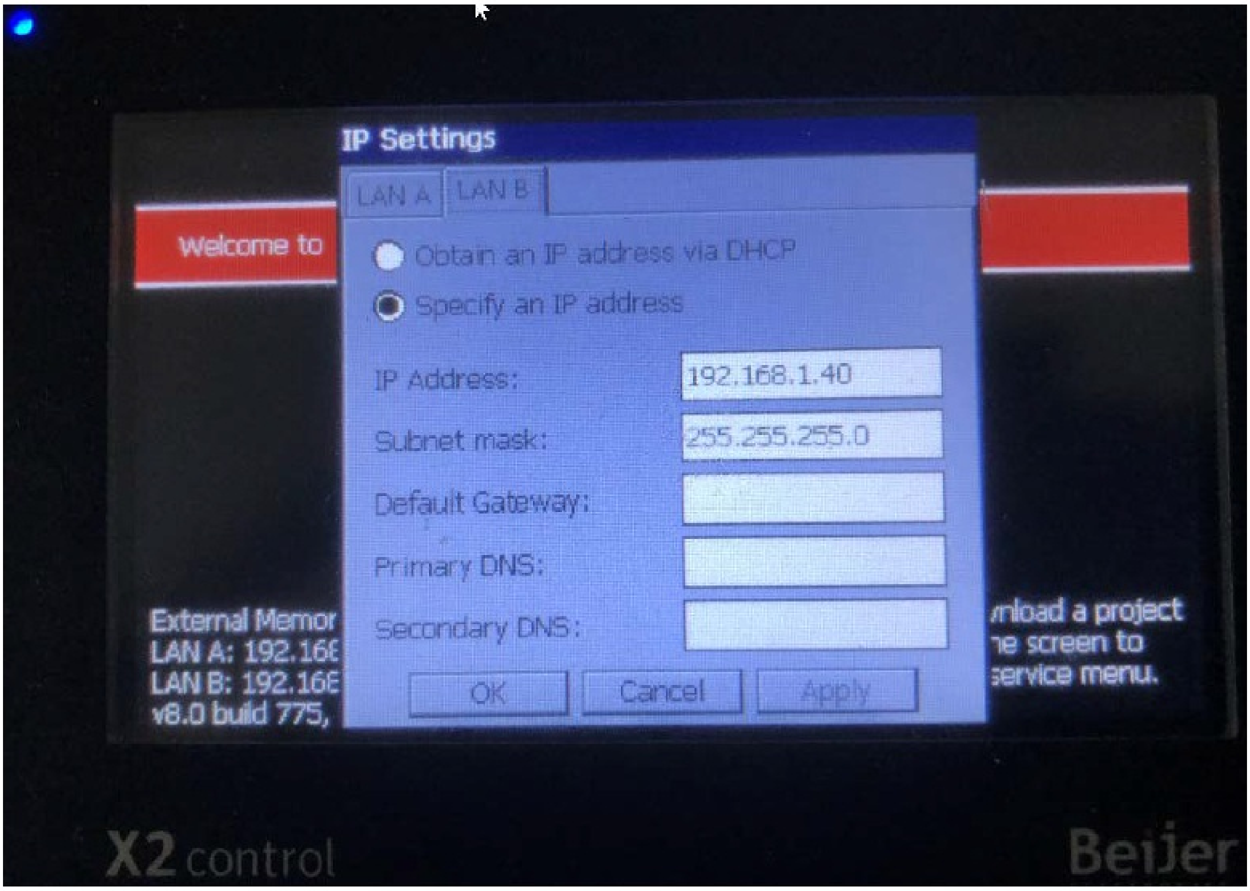


X2 Pro X2 Control X2 Motion X2 Marine

| Female DB-9 | Pin | COM1 signal | COM2 signal |
|---|---|---|---|
| | 1 | - | RS422 TX+/RS485 TX+/RX+ |
| | 2 | RS232 RX | - |
| | 3 | RS232 TX | - |
| | 4 | - | RS422 RX+ |
| | 5 | GND | GND |
| | 6 | - | RS422 TX- / RS485 TX-/RX- |
| | 7 | RS232 RTS | - |
| | 8 | RS232 CTS | - |
| | 9 | - | RS422 RX- |

**IP address setting of X2 control**

The setting of the Ethernet ports in X2 control is being done on the X2 control itself.
Turn Power on and keep your finger attached to the screen. After a while follow the instructions as in the picture below.



IP Settings

LAN A | LAN B

○ Obtain an IP address via DHCP

● Specify an IP address

IP Address: 192.168.1.40

Subnet mask: 255.255.255.0

Default Gateway:

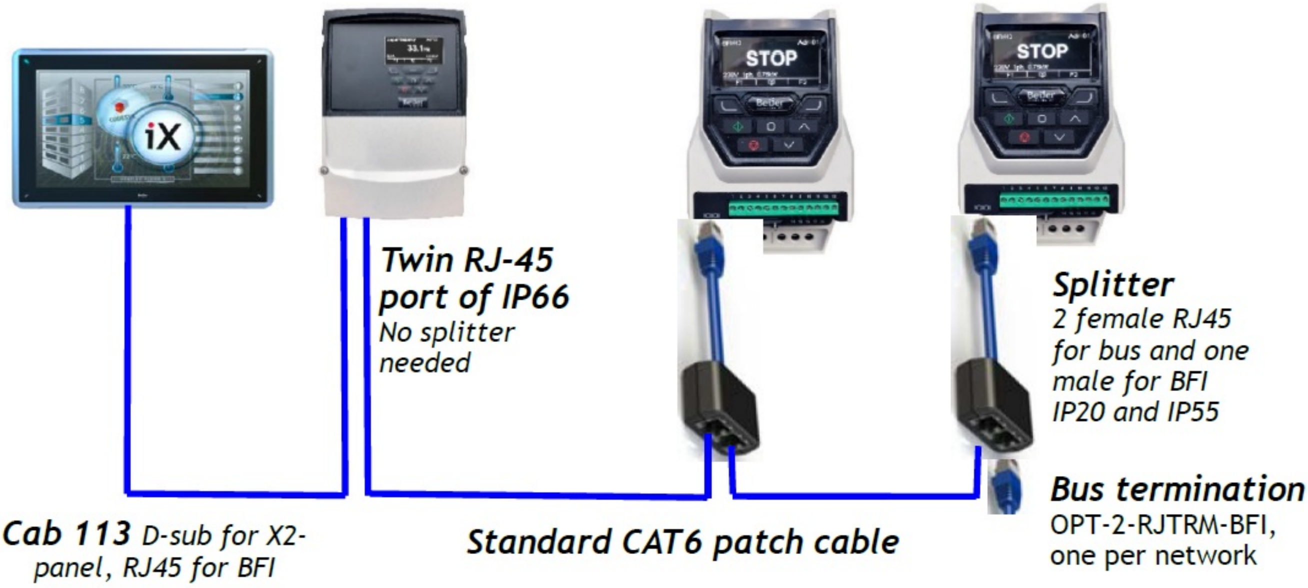
Primary DNS:

Secondary DNS:

OK | Cancel | Apply

**Modbus RTU port of BFI**

Modbus RTU port in BFI depends on model. It has either one or two RJ45 connector with pin configuration as in figure below. All BFI IP66 has two RJ45 connectors for incoming and outgoing network cabling. All BFI IP20 and IP55 has one RJ45 for Modbus RTU connection.

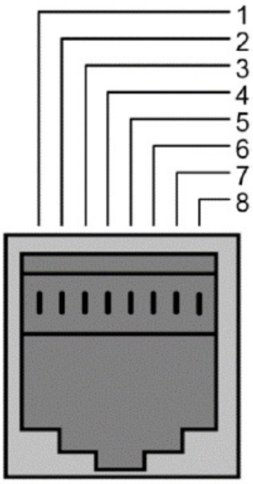


iX
STOP
STOP
Twin RJ-45 port of IP66
No splitter needed
Splitter
2 female RJ45 for bus and one male for BFI IP20 and IP55
Bus termination
OPT-2-RJTRM-BFI, one per network
Cab 113 D-sub for X2-panel, RJ45 for BFI
Standard CAT6 patch cable

| Name | Item number | Explanation |
|---|---|---|
| **CAB113** | 660000290 | 3m cable with 9-pole D-sub and RJ-45 between X2 control, BFI- P2/H3 /E3 for Modbus RTU communication including 120-ohm termination resistor. |
| **Splitter:**<br>OPT-2-J45SP-BFI | 63148 | RS485 Serial communication Data Cable 2-port Splitter for BFI-P2, BFI -H3, BFI-E3 for Modbus RTU and CANopen. |
| **Bus termination:**<br>OPT-2-RJTRM-BFI | 63202 | RJ-45 with End termination, 120 ohms, RJ45 plug for CANopen and Modbus RTU communication to BFI. |

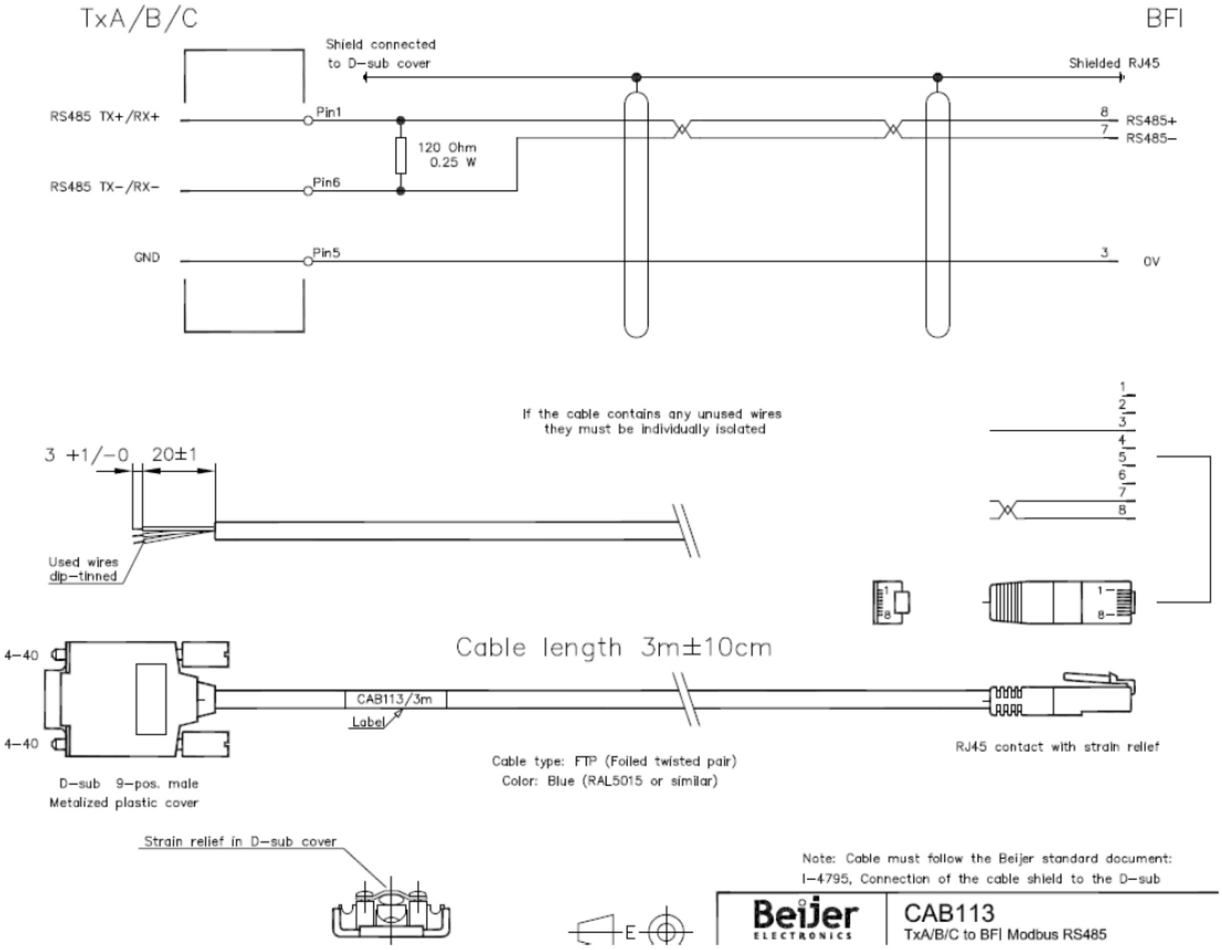
All BFI IP66 up to 22 kW has 2 serial ports in parallel which means that the splitter is not necessary.

**The pin configuration of the RJ-45 in BFI**

| | |
|---|---|
|  | 1: See manual for each BFI |
| | 2: See manual for each BFI |
| | 3: 0 Volt |
| | 4: Optibus / Remote keypad / PC Connection – |
| | 5: Optibus / Remote keypad / PC Connection + |
| | 6: +24 Volt Remote keypad Power Supply |
| | 7: RS 485- Modbus RTU |
| | 8 : RS 485+ Modbus RTU |
| | **Warning:** This is not an Ethernet connection. |

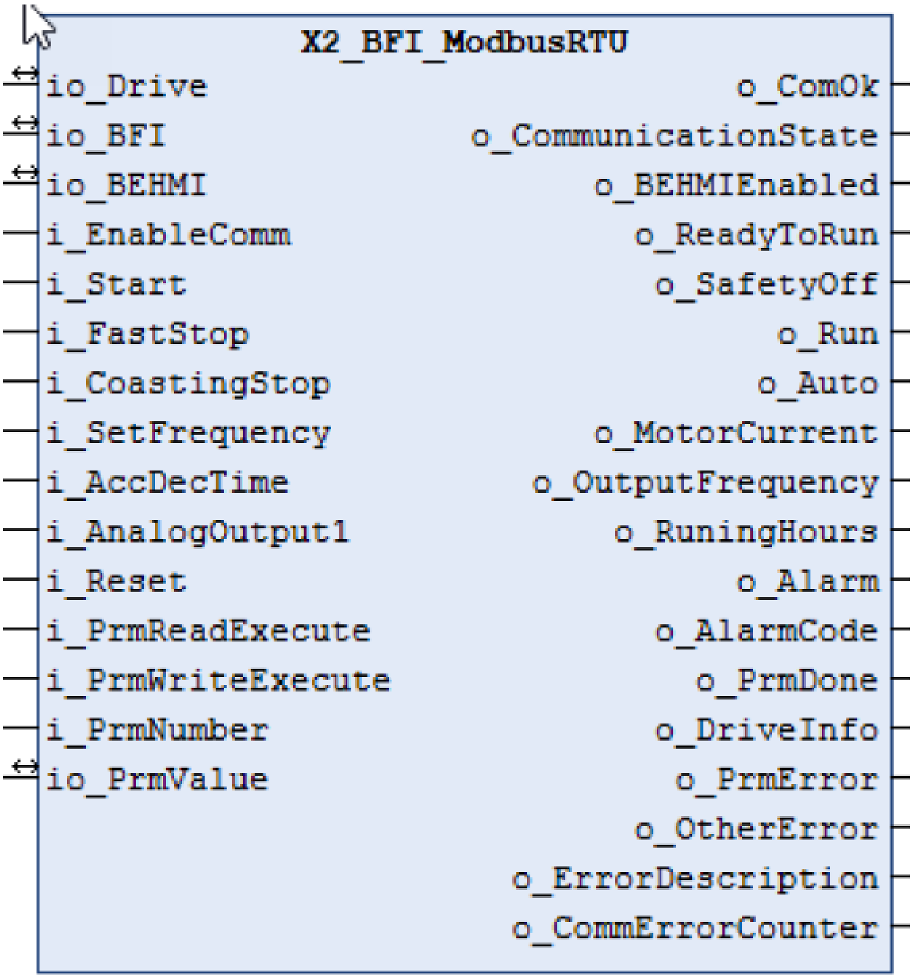
**Drawing CAB113**

3m cable with 9-pole D-sub and RJ-45 between X2 control, BFI-P2/H3/E3 for Modbus RTU communication including 120-ohm termination resistor.


TxA/B/C
BFI
Shield connected to D-sub cover
Shielded RJ45
RS485 TX+/RX+
Pin1
120 Ohm 0.25 W
RS485 TX-/RX-
Pin6
GND
Pin5
8
7
RS485+
RS485-
3
0V
If the cable contains any unused wires they must be individually isolated
3 +1/-0
20±1
Used wires dip-tinned
1
2
3
4
5
6
7
8
4-40
4-40
Cable length 3m±10cm
CAB113/3m
Label
D-sub 9-pos. male
Metalized plastic cover
Cable type: FTP (Foiled twisted pair)
Color: Blue (RAL5015 or similar)
RJ45 contact with strain relief
Strain relief in D-sub cover
Note: Cable must follow the Beijer standard document: I-4795, Connection of the cable shield to the D-sub
E
Beijer ELECTRONICS
CAB113
TxA/B/C to BFI Modbus RS485

## Function Block X2_BFI_ModbusRTU

| Information | Description |
|---|---|
| Library | X2 BFI Modbus RTU |
| Versions of CODESYS and BCS Tools | CODESYS runtime 3.5.13.30.7<br>Compiler: 3.5.13.30<br>Device (Beijer X2 control x -xxxxx) 3.5.13.30 iX image 775 or later<br>Modbus_Com: 3.5.11.20<br>Modbus Master: 3.5.13.10<br>Modbus_RTU_Slave: 3.5.10.30 Baud rate: 115,2 kbits/sec |
| Function Block | X2_BFI_ModbusRTU |
| Functionality | This function block is used to control and Read/Write Data<br>with a BFI-E3/P2/H3 by Modbus RTU. |
| PLC Types | X2 control (marine / extreme) or BoX2 pro (SC / motion) |
| Version | 1.0.2 |



| Input_Output | Type | Function |
|---|---|---|
| io_Drive | strDRIVE_REF_BFI | Reference to variables connected to BFI by Modbus RTU. |
| io_BFI | IoDrvModbus.ModbusSla veComPort_Diag | Connection to Modbus slave (Name of the Modbus Slave) |
| io_BEHMI | strBFIDrvBEHMI | Reference to variables from/to HMI |

| Input | Type | Function |
|---|---|---|
| i_EnableComm | BOOL | Activate the communication between BFI and CODESYS channels. |
| i_Start | BOOL | Start of Drive. 0=>Stop 1=>Running.<br>Direction decided by sign of Set frequency. |

| | | |
|---|---|---|
| i_FastStop | BOOL | 1 => Fast stop of motor with ramp in P2-25 for BFI-P2/H3 and P-24 for BFI-E3. |
| i_CoastingStop | BOOL | 1=> Coasting stop of motor. No output frequency. |
| i_SetFrequency | REAL | Set frequency for motor, 50.0=50.0 Hz. Positive value => Rotation clock wise. Negative value=> Rotation counter clockwise. |
| i_AccDecTime | REAL | Used if P5-07=Disabled in BFI-P2/H3 or P-12=4 in BFI-E3. If so, Acceleration and Deceleration time [0.1 sec]. |
| i_AnalogOutput1 | INT | Value 0-1000 sets Analog output to 0-10 VDC or 4-20 mA. |
| i_Reset | BOOL | Reset an alarm or error message from BFI. |
| i_PrmReadExecute | BOOL | Read a value from a Parameter in BFI. |
| i_PrmWriteExecute | BOOL | Write a value to a Parameter in BFI. |
| i_PrmNumber | INT | Specifies parameter number to be read or written. |
| **Input_Output** | **Type** | **Function** |
| io_PrmValue | INT | Value to be written to a parameter or read value from a parameter. |
| **Output** | **Type** | **Function** |
| o_ComOk | BOOL | True=Communication up running without any errors. |
| o_CommunicationState | UINT | 0=Communication succeeded, 1-8=Command Error, 9-254=Some kind of Communication error, 255=Noting sent on bus |

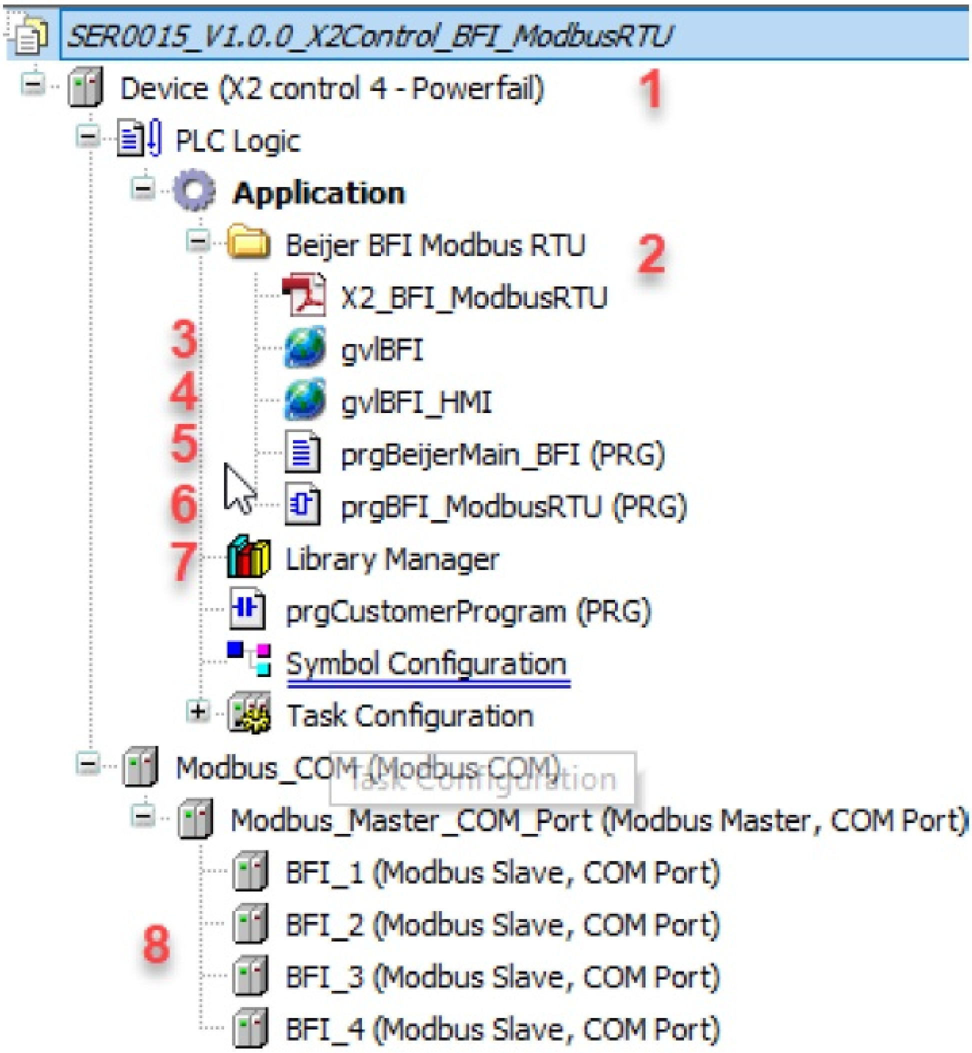
| | | |
|---|---|---|
| o_BEHMIEnabled | BOOL | BeHMI override enabled. |
| o_ReadyToRun | BOOL | BFI Ready to run. Power ON, Enabled, Safety ok. |
| o_SafetyOff | BOOL | Safe Torque off, False if STO wires is not connected. Not BFI-E3. |
| o_Run | BOOL | Start signal received by BFI. |
| o_Auto | BOOL | Only for BFI-H3. 1=Auto, 0= Hand. |
| o_MotorCurrent | REAL | Motor current / A. |
| o_OutputFrequency | REAL | Output frequency / Hz. |
| o_RunningHours | INT | Total Running Hours / Hours. |
| o_Alarm | BOOL | Alarm in inverter. |
| o_AlarmCode | STRING | Error code of the alarm, read automatically. |
| o_PrmDone | BOOL | Read or Write of parameter has been done. |
| o_DriveInfo | STRING | Drive ID, Type, Voltage, Firmware version, serial number. |
| o_PrmError | BOOL | Parameter operation error or not allowed message was sent to BFI. |
| o_OtherError | BOOL | Summary error bit. |
| o_ErrorDescription | STRING | Error and status description. |

| | | |
|---|---|---|
| o_CommErrorCounter | UDINT | Total number of errors or interruptions of communication messages on t he bus since PLC was put into Run. Can’t be reseted by user. |

## Program example CODESYS, BCS Tools and iX Developer

BFI and X2 control Start Up project is a very powerful tool providing a project starting point way beyond what would otherwise be possible.
The project contains one HMI-screen to handle the BFI, start/stop, read/write parameters, read type of BFI, handling alarms. The purpose of the screen is to keep them for future use by technicians and/or maintenance personnel. In addition, the project contains a CODESYS/BCS Tools PLC-program correctly configured for use with BFI with a function block comprising all standard drive functionality.

### Explanation of the CODESYS/BCS Tools project



SER0015_V1.0.0_X2Control_BFI_ModbusRTU
- Device (X2 control 4 - Powerfail)
  - PLC Logic
    - Application
      - Beijer BFI Modbus RTU
        - X2_BFI_ModbusRTU
        - gvlBFI
        - gvlBFI_HMI
        - prgBeijerMain_BFI (PRG)
        - prgBFI_ModbusRTU (PRG)
      - Library Manager
      - prgCustomerProgram (PRG)
      - Symbol Configuration
      - Task Configuration
  - Modbus_COM (Modbus COM)
    - Modbus_Master_COM_Port (Modbus Master, COM Port)
      - BFI_1 (Modbus Slave, COM Port)
      - BFI_2 (Modbus Slave, COM Port)
      - BFI_3 (Modbus Slave, COM Port)
      - BFI_4 (Modbus Slave, COM Port)

1. **Device (X2 control 4 – Powerfail)**

   The X2 control hardware must be defined in detail due to allocation of memory. In this case a 4 inch. How to handle Retain data at Power interruption must also be chosen. Either cyclic storage or when power interruption occurs. With limited Retain data Power fail is a sufficient choice. Should be at least version 3.5.13.30. This or later versions are to be downloaded on www.beijerelectronics.com.
2. **Folder Beijer BFI Modbus RTU.**

   Contain everything necessary for Modbus RTU control of BFI.

3. **gvlBFI**

   Contain Structs for Variables connected on pins to/from FB, Variables to/from iX used inside FB and Variables connecting each BFI with corresponding input/output data (ModbusGenericSerialSlave I/O Mapping).

4. **gvlBFI_HMI**

   Contain variables to be used together with HMI. Important are the constant describing number of BFI in network and Structure containing variables copied to/from HMI.

5. **prgBeijerMain_BFI**

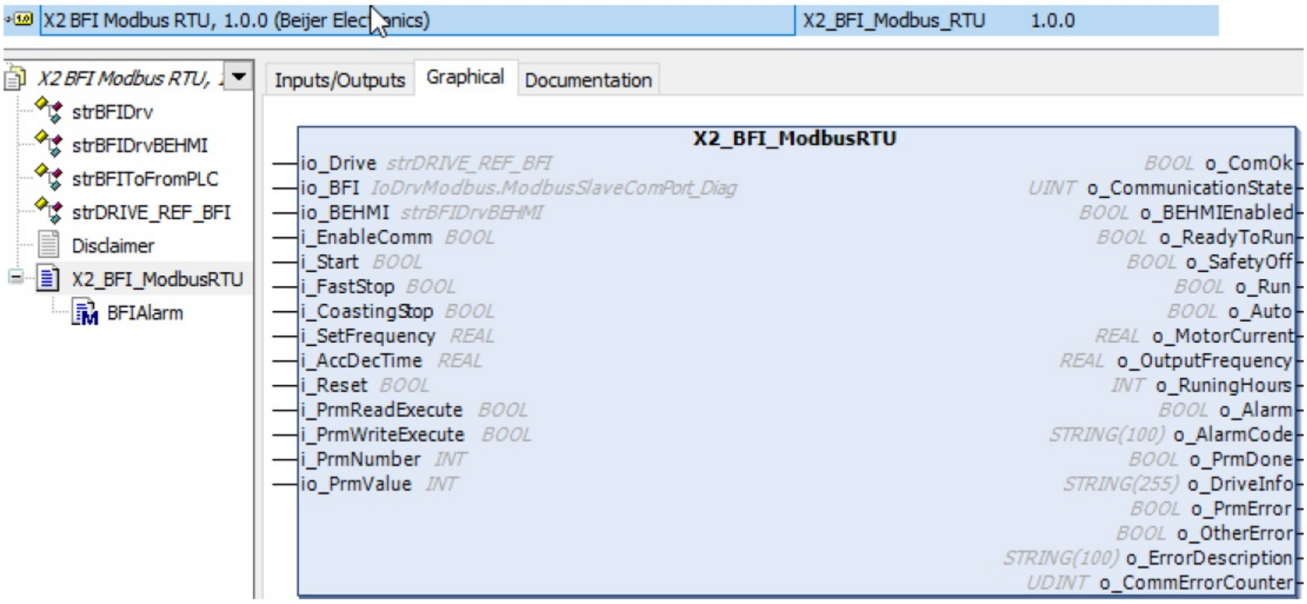
   Contain code for exchanging data between HMI and CODESYS/BCS Tools.

6. **prgBFI_ ModbusRTU**

   Containing Function blocks, one FB for each BFI.

7. **Library Manager**

   One of the libraries is X2_BFI_Modbus RTU. Contains the FB to control BFI by Modbus RTU. The FB is compiled and by that not possible to open up. Requires one FB per BFI.



X2 BFI Modbus RTU, 1.0.0 (Beijer Electronics) X2_BFI_Modbus_RTU 1.0.0

```
                                   X2_BFI_ModbusRTU
io_Drive strDRIVE_REF_BFI                                        BOOL o_ComOk
io_BFI IoDrvModbus.ModbusSlaveComPort_Diag        UINT o_CommunicationState
io_BEHMI strBFIDrvBEHMI                                   BOOL o_BEHMIEnabled
i_EnableComm BOOL                                          BOOL o_ReadyToRun
i_Start BOOL                                                 BOOL o_SafetyOff
i_FastStop BOOL                                                   BOOL o_Run
i_CoastingStop BOOL                                              BOOL o_Auto
i_SetFrequency REAL                                      REAL o_MotorCurrent
i_AccDecTime REAL                                     REAL o_OutputFrequency
i_Reset BOOL                                             INT o_RuningHours
i_PrmReadExecute BOOL                                           BOOL o_Alarm
i_PrmWriteExecute BOOL                                STRING(100) o_AlarmCode
i_PrmNumber INT                                               BOOL o_PrmDone
io_PrmValue INT                                       STRING(255) o_DriveInfo
                                                              BOOL o_PrmError
                                                            BOOL o_OtherError
                                                STRING(100) o_ErrorDescription
                                                   UDINT o_CommErrorCounter
```

8. **BFI Slaves**

   BFI Slaves in the network. Important that each physical BFI has an individual IP address in CODESYS/BCS Tools corresponds with the setting in each BFI.

### How to add an additional BFI in the network

- Enter GVL called gvlBFI_HMI. Adjust the global constant gvlBFI_HMI.gc_iNrOfDrive to number of BFI. If 5 BFI constant should be 5.
- Enter GVL called gvlBFI. Adjust the global array variable gvlBFI.g_aDrive[]. Set the array to [1..Number of BFI in the network]. If 5 BFI constant should be 5.
- Enter GVL called gvlBFI . Adjust the global array variable gvlBFI.g_aDriveBEHMI[]. Set the array to [1..Number of BFI in the network]. If 5 BFI constant should be 5.
- Enter GVL called gvlBFI. If more than 8 BFI. Add variables for BFI9, Drive9 of type strDRIVE_REF_BFI. And for BFI10…
- Add BFI into Modbus RTU network
  - Chose Modbus_Master_COM_port (Modbus Master, COM PORT)
  - Right click and ADD Device
  - Chose Modbus RTU Slave

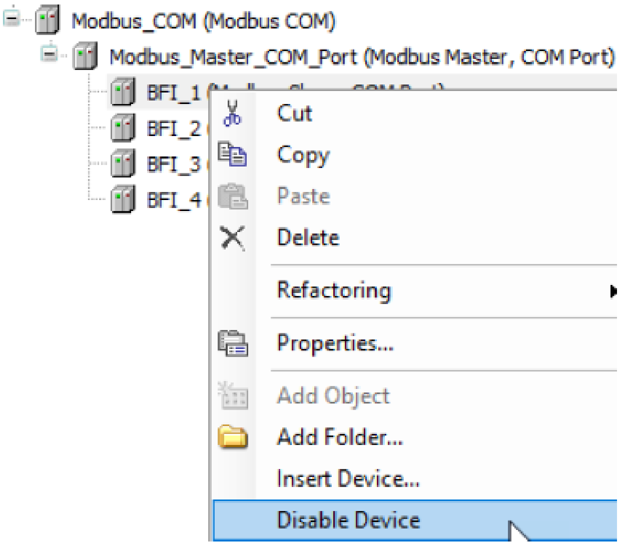
    - Rename it. If it is BFI number 5, refactoring it to BFI5.
- Choose the new BFI
    - In Folder General, set Slave address of the BFI. Must correspond with setting in BFI.
    - In Folder General, keep Response Timeout(ms): 50
    - In Folder Modbus Slave Channel, Copy all data from an existing BFI to new BFI.
    - In Folder MobusGenericSerialSlave I/OMapping, Copy all variable names from BFI_1, to the new BFI. If the new BFI will be BFI5 then: Rename
    - **Application.**Drive1.PlcToFromDrv.iRunningHours, to
    - **Application.**Drive5.PlcToFromDrv.iRunningHours,
- Add FB for a new BFI in prgBFI_ModbusRTU. Copy network 1 and paste. If new BFI is to be number 5 then:
    - Change name fbControlBFI1 to fbControlBFI5
    - Change name Drive1 to Drive5
    - Change name BFI_1 to BFI_5
    - Change name for all [1] to [5] for all other variables in the network

### How to remove BFI in the network

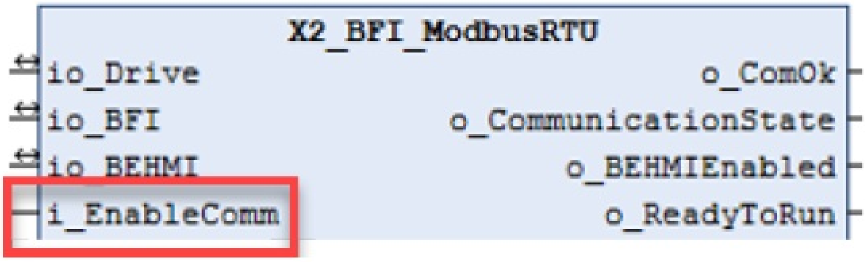
- Enter GVL called gvlBFI_HMI. Adjust the global constant gvlBFI_HMI.gc_iNrOfDrive to number of BFI.
- Enter GVL called gvlBFI. Adjust the global array variable gvlBFI.g_aDrive[]. Set the array to [1..Number of BFI in the network].
- Enter GVL called gvlBFI . Adjust the global array variable gvlBFI.g_aDriveBEHMI[]. Set the array to [1..Number of BFI in the network].
- Remove not used BFI in Modbus_RTU_Master

### Enable/Disable Devices

If a BFI is not to be used in the network during commissioning or other reasons it can be excluded, and communication shut off. This can either be done in configuration or by a tag in the PLC-program. Go Offline, chose the Slave and then Disable Device. When to enable a device chose Enable Device instead.

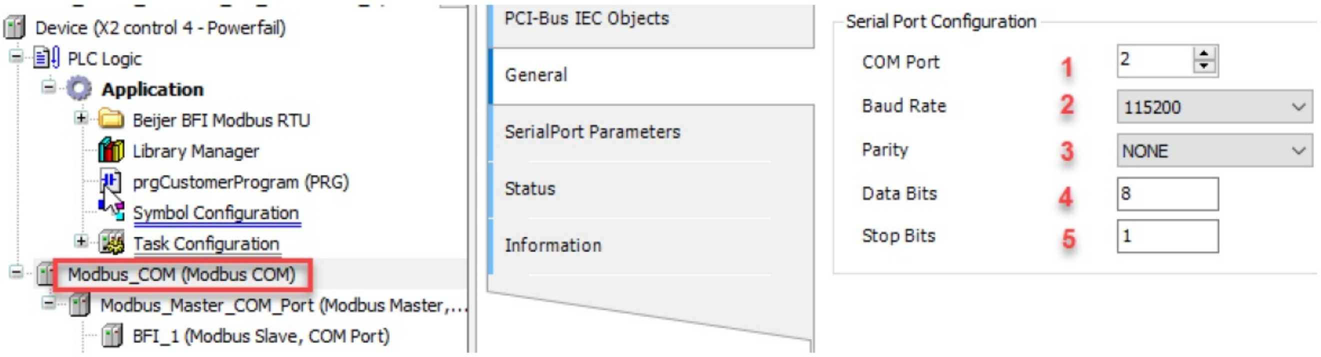


It is also possible to control it in the plc-program by the red marked variable



## Modbus RTU setting in CODESYS/BCS Tools

The setting of Modbus RTU ports must also be done in CODESYS/BCS Tools, both for the Master and for all Slaves.

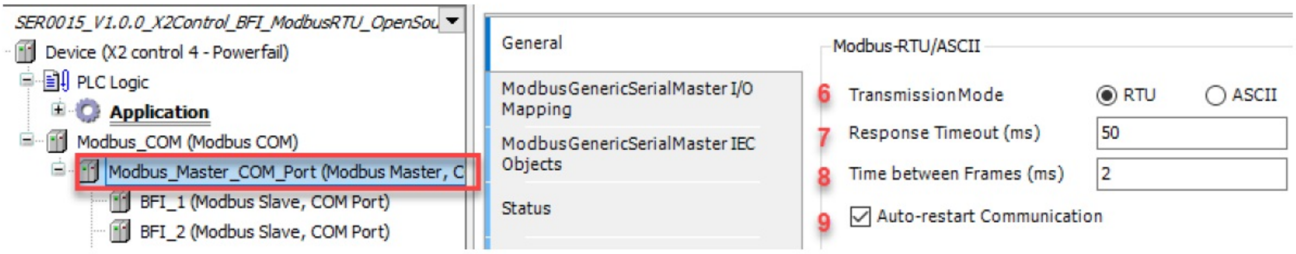


Serial Port Configuration

| | | |
|---|---|---|
| COM Port | 1 | 2 |
| Baud Rate | 2 | 115200 |
| Parity | 3 | NONE |
| Data Bits | 4 | 8 |
| Stop Bits | 5 | 1 |

1. 2 is the name of the Modbus RTU port on Pin 1 and 6 of the Com port.
2. Baud rate [bits per second] for transmitting data on the bus.

   Maximum working baud rate is decided by limitations of Modbus RTU Slaves, required bandwidth, network cable length and cross-area. Try to keep baud rate as low as possible to avoid communication interruption due to noise. 115200 is default in BFI.
3. Parity “NONE” is default in BFI.
4. Data Bits 8 is default in BFI.
5. Stop Bits 1 is default in BFI.


SER0015_V1.0.0_X2Control_BFI_ModbusRTU_OpenSou
Device (X2 control 4 - Powerfail)
PLC Logic
Application
Modbus_COM (Modbus COM)
Modbus_Master_COM_Port (Modbus Master, C
BFI_1 (Modbus Slave, COM Port)
BFI_2 (Modbus Slave, COM Port)
General
ModbusGenericSerialMaster I/O Mapping
ModbusGenericSerialMaster IEC Objects
Status

Modbus-RTU/ASCII

6 Transmission Mode ● RTU ○ ASCII

7 Response Timeout (ms) 50

8 Time between Frames (ms) 2

9 ☑ Auto-restart Communication

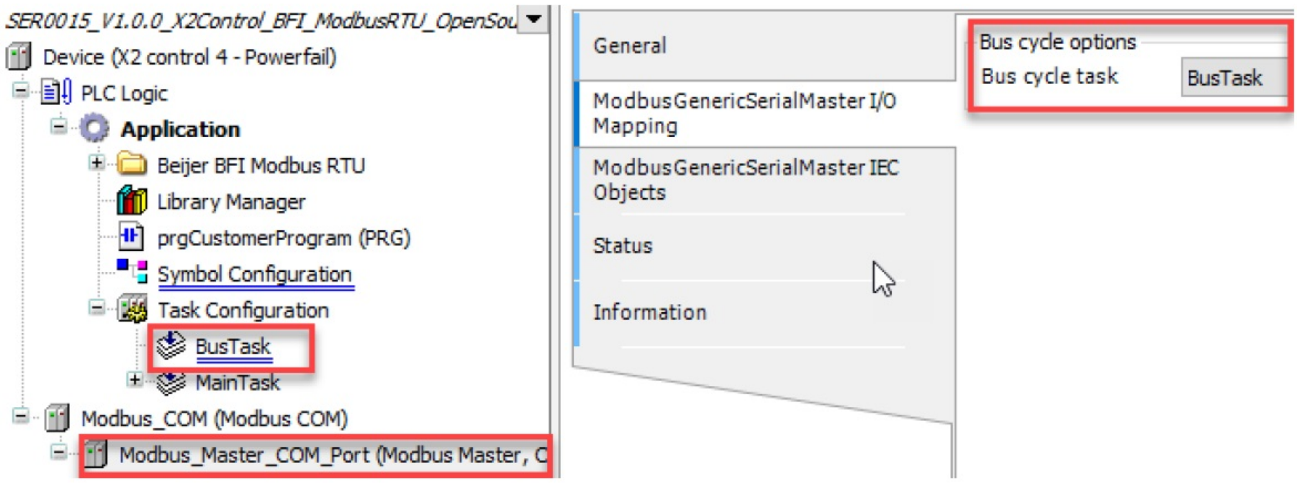
6. **Transmission Mode**

   Must be set to RTU.
7. **Response Timeout**

   The time from a message sent until a reply must have been received. 50 msec is a large value.
8. **Time between Frames**

   Time that master is quiet, time between one message arriving until next message being sent.
9. **Auto-restart Communication**

   The master is automatically trying to establish communication with a slave. If not used this must be done by plc-code.

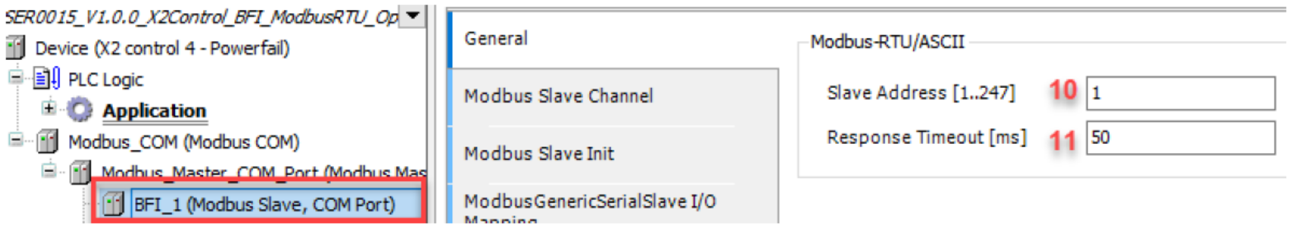


SER0015_V1.0.0_X2Control_BFI_ModbusRTU_OpenSou
Device (X2 control 4 - Powerfail)
PLC Logic
Application
Beijer BFI Modbus RTU
Library Manager
prgCustomerProgram (PRG)
Symbol Configuration
Task Configuration
BusTask
MainTask
Modbus_COM (Modbus COM)
Modbus_Master_COM_Port (Modbus Master, C

General
ModbusGenericSerialMaster I/O Mapping
ModbusGenericSerialMaster IEC Objects
Status
Information

Bus cycle options
Bus cycle task: BusTask

The time how often the Modbus master is to access the slaves are controlled by setting of a Task in Task Configuration. In this case a task called “BusTask” has been created and control how often the Modbus master access the network.



SER0015_V1.0.0_X2Control_BFI_ModbusRTU_Op
Device (X2 control 4 - Powerfail)
PLC Logic
Application
Modbus_COM (Modbus COM)
Modbus_Master_COM_Port (Modbus Mas
BFI_1 (Modbus Slave, COM Port)

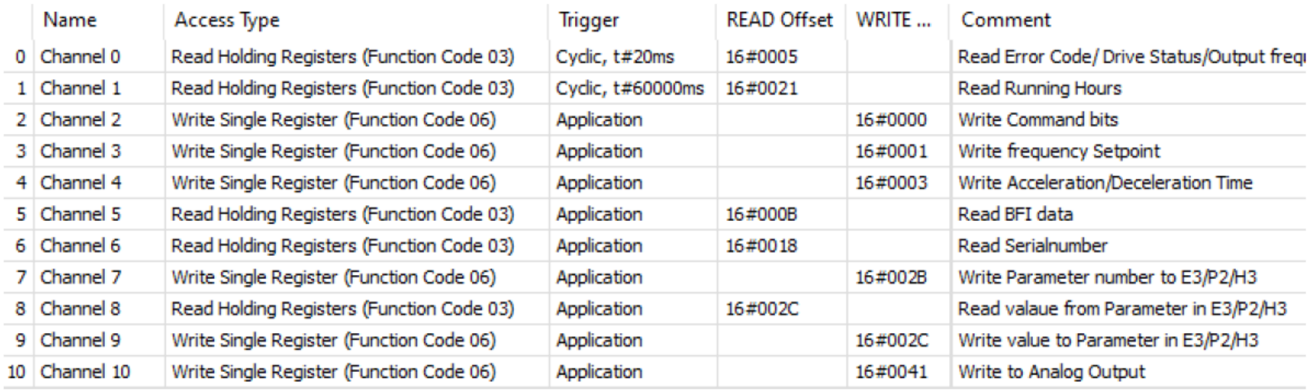
Modbus-RTU/ASCII
Slave Address [1..247] 10: 1
Response Timeout [ms] 11: 50

10. **Slave Addresses**

    Slave Address of the BFI. Must correspond to the settings in BFI.

11. **Response Timeout**

    Response Timeout[ms] Time interval for the master to wait for the response from the slave. This is especially configured for this slave node and overwrites the general response timeout setting of the respective master.

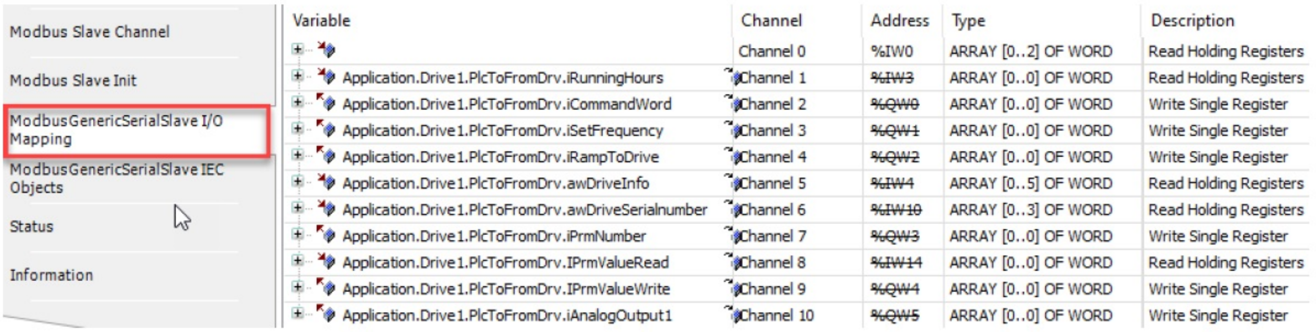


| | Name | Access Type | Trigger | READ Offset | WRITE ... | Comment |
|---|---|---|---|---|---|---|
| 0 | Channel 0 | Read Holding Registers (Function Code 03) | Cyclic, t#20ms | 16#0005 | | Read Error Code/ Drive Status/Output freq |
| 1 | Channel 1 | Read Holding Registers (Function Code 03) | Cyclic, t#60000ms | 16#0021 | | Read Running Hours |
| 2 | Channel 2 | Write Single Register (Function Code 06) | Application | | 16#0000 | Write Command bits |
| 3 | Channel 3 | Write Single Register (Function Code 06) | Application | | 16#0001 | Write frequency Setpoint |
| 4 | Channel 4 | Write Single Register (Function Code 06) | Application | | 16#0003 | Write Acceleration/Deceleration Time |
| 5 | Channel 5 | Read Holding Registers (Function Code 03) | Application | 16#000B | | Read BFI data |
| 6 | Channel 6 | Read Holding Registers (Function Code 03) | Application | 16#0018 | | Read Serialnumber |
| 7 | Channel 7 | Write Single Register (Function Code 06) | Application | | 16#002B | Write Parameter number to E3/P2/H3 |
| 8 | Channel 8 | Read Holding Registers (Function Code 03) | Application | 16#002C | | Read valaue from Parameter in E3/P2/H3 |
| 9 | Channel 9 | Write Single Register (Function Code 06) | Application | | 16#002C | Write value to Parameter in E3/P2/H3 |
| 10 | Channel 10 | Write Single Register (Function Code 06) | Application | | 16#0041 | Write to Analog Output |

Communication with a slave is defined as channels. Each channel is defined as Read or Write, how to read or write, time interval, Modbus address and amount of data. Following data are used in this example:

**Channel 0:** Cyclic Read of Digital status, Error codes, Output frequency and Motor current.
**Channel 1:** Cyclic Read of Total Running Hours once every minute.
**Channel 2:** Write of Command bits
**Channel 3:** Write of Frquency setpoint
**Channel 4:** Write of Acceleration and Deceleration time
**Channel 5:** Read of BFI data.
**Channel 6:** Read of serial number every time communication is established.

**Channel 7:** Write parameter number to access when required.
**Channel 8:** Read Parameter value when required.
**Channel 9:** Write parameter value when required.
**Channel 10:** Write to Analog Output.


Modbus Slave Channel
Modbus Slave Init
ModbusGenericSerialSlave I/O Mapping
ModbusGenericSerialSlave IEC Objects
Status
Information

| Variable | Channel | Address | Type | Description |
|---|---|---|---|---|
| | Channel 0 | %IW0 | ARRAY [0..2] OF WORD | Read Holding Registers |
| Application.Drive1.PlcToFromDrv.iRunningHours | Channel 1 | ~~%IW3~~ | ARRAY [0..0] OF WORD | Read Holding Registers |
| Application.Drive1.PlcToFromDrv.iCommandWord | Channel 2 | ~~%QW0~~ | ARRAY [0..0] OF WORD | Write Single Register |
| Application.Drive1.PlcToFromDrv.iSetFrequency | Channel 3 | ~~%QW1~~ | ARRAY [0..0] OF WORD | Write Single Register |
| Application.Drive1.PlcToFromDrv.iRampToDrive | Channel 4 | ~~%QW2~~ | ARRAY [0..0] OF WORD | Write Single Register |
| Application.Drive1.PlcToFromDrv.awDriveInfo | Channel 5 | ~~%IW4~~ | ARRAY [0..5] OF WORD | Read Holding Registers |
| Application.Drive1.PlcToFromDrv.awDriveSerialnumber | Channel 6 | ~~%IW10~~ | ARRAY [0..3] OF WORD | Read Holding Registers |
| Application.Drive1.PlcToFromDrv.iPrmNumber | Channel 7 | ~~%QW3~~ | ARRAY [0..0] OF WORD | Write Single Register |
| Application.Drive1.PlcToFromDrv.IPrmValueRead | Channel 8 | ~~%IW14~~ | ARRAY [0..0] OF WORD | Read Holding Registers |
| Application.Drive1.PlcToFromDrv.IPrmValueWrite | Channel 9 | ~~%QW4~~ | ARRAY [0..0] OF WORD | Write Single Register |
| Application.Drive1.PlcToFromDrv.iAnalogOutput1 | Channel 10 | ~~%QW5~~ | ARRAY [0..0] OF WORD | Write Single Register |

Data read from or written to a channel is copied to plc-variables in ModbusGenericSerialSlaveI/OMapping. Below is variables for BFI number1:

**Digital control bits from BFI =>** Application.Drive1.PlcToFromDrv.iStatusWord.
**Output frequency from BFI =>** Application.Drive1.PlcToFromDrv.iOutputFrequency
**Motor current from BFI =>** Application.Drive1.PlcToFromDrv.iMotorCurrent.
**Running Hours from BFI =>** Application.Drive1.PlcToFromDrv.iRunningHours
**Digital control bits to BFI =>** Application.Drive1.PlcToFromDrv.iCommandWord
**Set frequency to BFI=>** Application.Drive1.PlcToFromDrv.iSetFrequency
**Acceleration and Dec.time to BFI =>**Application.Drive1.PlcToFromDrv.iSetFrequency
**Detailed Drive info from BFI =>** Application.Drive1.PlcToFromDrv.awDriveInfo
**Serial number from BFI =>** Application.Drive1.PlcToFromDrv.awDriveSerialnumber
**Parameter number to/from BFI =>** Application.Drive1.PlcToFromDrv.iPrmNumber
**Read parameter value from BFI =>** Application.Drive1.PlcToFromDrv.iPrmValueRead
**Parameter value to be written =>** Application.Drive1.PlcToFromDrv.iPrmValueWrite
**Control of Analog Output =>** Application.Drive1.PlcToFromDrv.iAnalogOutput1

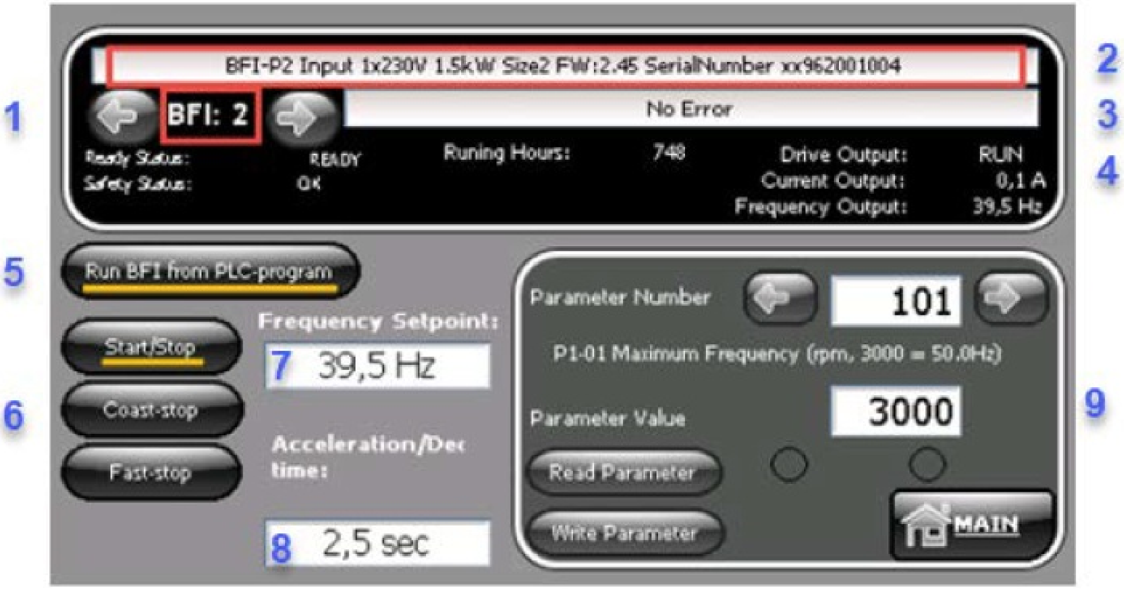
**iX-project**

CODESYS/BCS Tools project, X2Control_BFI_Modbus RTU, has a corresponding iX project. The project contains one Start_Screen and one BFI_Control Screen. Explanation of project:

1. Choose which BFI to be monitored. Either tap in the figure or use Increase/Decrease buttons
2. BFI data
    - Type of BFI
    - Input voltage and number of phases
    - Power
    - Physical size
    - Firmware version
    - Serial number
3. Description of Communication error or User error.
4. Actual status of BFI
    - Ready to Run or not
    - Safety Status (STO) in BFI-H3 and BFI-P2
    - Total Running Hours
    - Motor in Run or Stop
    - Motor current

- Output frequency
- If Alarm in BFI it is shown
- Explanation of what the alarm code means

5. Take control of the BFI from this menu. PLC-program is neglected.
6. Start/stop- Coast- and Fast stop buttons.
7. Frequency Setpoint.
8. Acceleration/Deceleration Time to be used if P-12=4 in BFI-P2/E3.
9. Parameter handling
   - Parameter number to be either tapped in or Increase/Decrease buttons
   - Name/Explanation of parameter is shown
   - Read Parameter value or to be written
   - Button Read Parameter
   - Button Write Parameter
   - Status whether Read/Write of parameter Succeeded or not



BFI-P2 Input 1x230V 1.5kW Size2 FW:2.45 SerialNumber xx962001004

BFI: 2

No Error

Ready Status: READY

Safety Status: OK

Runing Hours: 748

Drive Output: RUN

Current Output: 0,1 A

Frequency Output: 39,5 Hz

Frequency Setpoint: 39,5 Hz

Acceleration/Dec time: 2,5 sec

Parameter Number 101

P1-01 Maximum Frequency (rpm, 3000 = 50,0Hz)

Parameter Value 3000

It is possible to have a sheet containing Data from several BFI. But this needs to be configured in Symbol Manager of CODESYS/BCS Tools and then imported to iX Developer.

**Parameter settings for Modbus RTU BFI-P2/E3/H3**

**Set following parameters in BFI-E3**

- **P12=3:** Modbus RTU control with Acceleration in P-03 and Deceleration time in P-04.
- **P12=4:** Modbus RTU control with Acceleration and Deceleration time sent by Modbus RTU.
- **P14=201** to make all parameters available.
- **P25=13** Analog Output of BFI-E3 controlled by bus.
- **P36 =** Node ID number. Default=1.
- **P36 =** 115,2 Kbits/sec. Setting depending of network size, network length and how slaves are to be connected/disconnected.

**Set following parameters in BFI-P2 and BFI-H3**

- **P1-12=4:** Modbus RTU control.
- **P1-14=201** to make all parameters available.
- **P5-01=**Node ID number. Default=1.
- **P5-02=**115,2 Kbits/sec. Setting depending of network size, network length and how slaves are to be connected/disconnected.
- **P5-07:** Disable=Acceleration and Deceleration time set in parameter P1-03 and P1-04
- **P5-07:** Enable= Acceleration and Deceleration time ramp sent by Modbus RTU.
- **P9-33=**2 Analog Output1 on terminal 8 of BFI-P2/H3 will be controllable from PLC.

**Communication timeout**

- The reaction when communication interruption between Modbus RTU master and BFI is set in parameters of BFI.

**Set following parameters in BFI-E3**
Time until a reaction and how to react are to be set in P-36

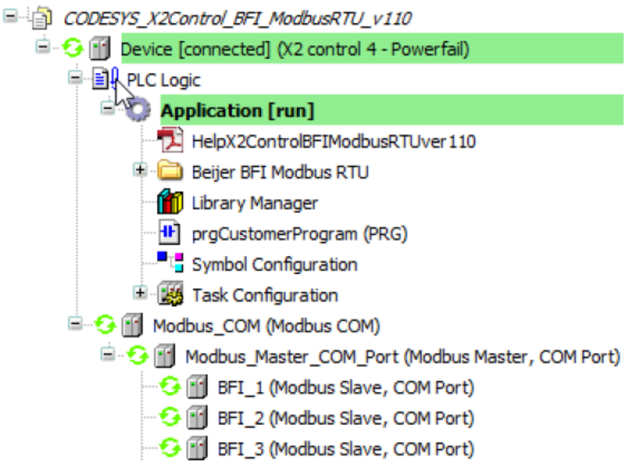
- 0: No reaction and drive continue with latest speed
- t 30: Drive trips within 30 msec, t 100: Drive trips within 100 msec
- t 1000: Drive trips within 1 sec t 3000: Drive trips within 1 sec
- r 30: Coast to stop without trip within 30 msec, r 100: Coast to stop without trip within 100 msec
- r 1000: Coast to stop without trip within 1 sec, r 3000: Coast to stop without trip within 3 sec.

**Set following parameters in BFI-P2 and BFI-H3**

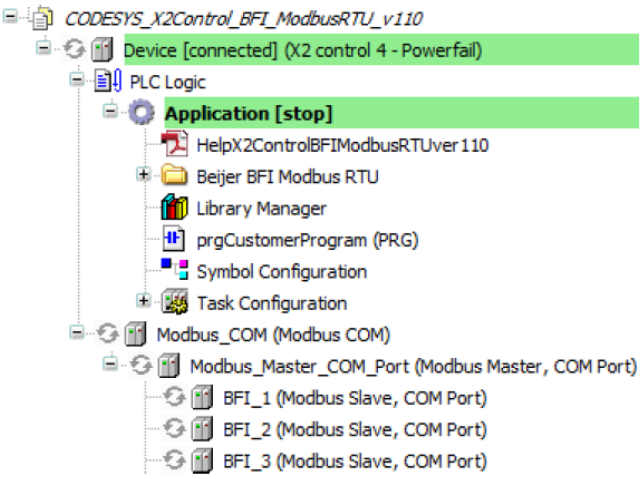
- P5-05=Timeout until BFI trip due to lost communication. If 0 then no trip.
- P5-06=Drive behaviour if communication is lost.
  0: Alarm and trip
  1: Ramp to stop and trip
  2: Ramp to stop and no trip
  3: Preset speed 8 (P2-08)

## Troubleshooting Communication problems

CODESYS/BCS Tools is the tool to check status of Modbus RTU communication.

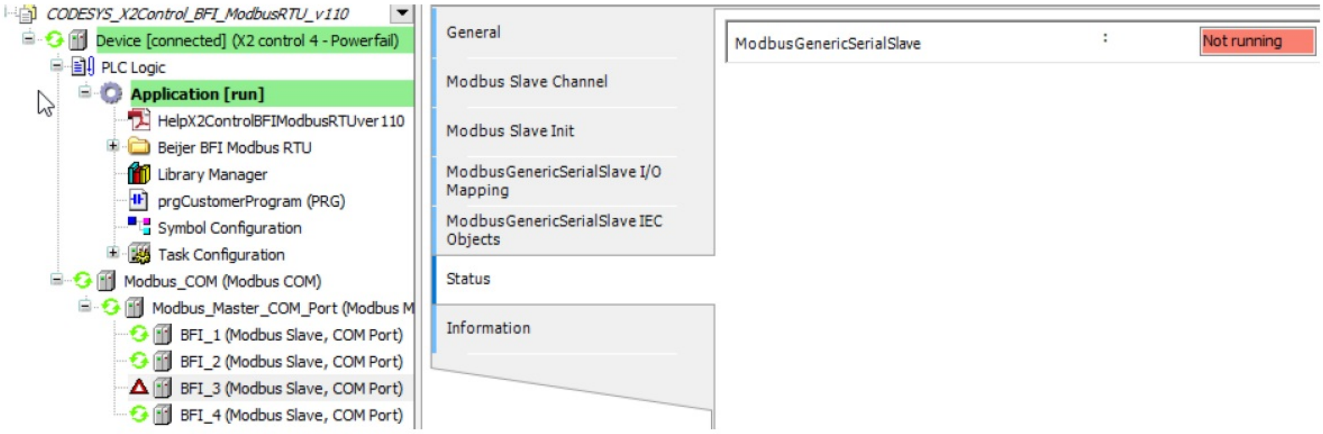

CODESYS_X2Control_BFI_ModbusRTU_v110
Device [connected] (X2 control 4 - Powerfail)
PLC Logic
Application [run]
HelpX2ControlBFIModbusRTUver110
Beijer BFI Modbus RTU
Library Manager
prgCustomerProgram (PRG)
Symbol Configuration
Task Configuration
Modbus_COM (Modbus COM)
Modbus_Master_COM_Port (Modbus Master, COM Port)
BFI_1 (Modbus Slave, COM Port)
BFI_2 (Modbus Slave, COM Port)
BFI_3 (Modbus Slave, COM Port)

CODESYS in RUN with communication with Slaves


CODESYS_X2Control_BFI_ModbusRTU_v110
Device [connected] (X2 control 4 - Powerfail)
PLC Logic
Application [stop]
HelpX2ControlBFIModbusRTUver110
Beijer BFI Modbus RTU
Library Manager
prgCustomerProgram (PRG)
Symbol Configuration
Task Configuration
Modbus_COM (Modbus COM)
Modbus_Master_COM_Port (Modbus Master, COM Port)
BFI_1 (Modbus Slave, COM Port)
BFI_2 (Modbus Slave, COM Port)
BFI_3 (Modbus Slave, COM Port)

CODESYS is in STOP

Example of communication when no Slave is replying could look like:


CODESYS_X2Control_BFI_ModbusRTU_v110
Device [connected] (X2 control 4 - Powerfail)
PLC Logic
Application [run]
HelpX2ControlBFIModbusRTUver110
Beijer BFI Modbus RTU
Library Manager
prgCustomerProgram (PRG)
Symbol Configuration
Task Configuration
Modbus_COM (Modbus COM)
Modbus_Master_COM_Port (Modbus M
BFI_1 (Modbus Slave, COM Port)
BFI_2 (Modbus Slave, COM Port)
BFI_3 (Modbus Slave, COM Port)
BFI_4 (Modbus Slave, COM Port)
General
Modbus Slave Channel
Modbus Slave Init
ModbusGenericSerialSlave I/O Mapping
ModbusGenericSerialSlave IEC Objects
Status
Information

ModbusGenericSerialSlave : Not running

Check following items if one or several slaves are not answering properly:

**Hardware**

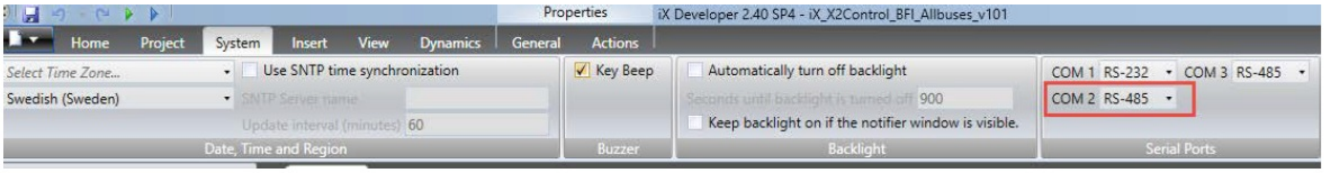
- If cable CAB113 is used, the setting in Modbus_COM(Modbus COM)/General COM Port must be set to 2.
- Check cabling of the network and splitters.
- COM Port 2 of X2 control means RS-485 TX+/RX+ on Pin1 and RS-485 TX-/RX- – is Pin6.
- Termination resistor applied correctly in both end of network? CAB 113 has termination built in. And then another 120-ohm resistor needs to be applied. When all system is powered off the resistance between RS-485 TX+/RX+ on Pin1 and RS-485 TX-/RX- on Pin6 should be 60 ohms. If not check termination.
- Power on to X2 Control?
- Power turned on to BFI?
- If slaves are disconnected/reconnected randomly there might be an EMC-problem with noise.
  Try to see if problems occur when a specific BFI is started or any other pattern.

**Software or settings**

- Control of BFI by Modbus is set by P1-12=4 in BFI-P2/H3 and P-12=3 or 4 in BFI-E3
- Communication speed setting in each slave corresponding to Master setting in CODESYS, P-36 in BFI-E3 or

P5-03 in BFI-P2/H3?

- No overlapping Station numbers, P-36 in BFI-E3 or P5-01 in BFI-P2/H3?
- Station numbers of slaves corresponding with settings in CODESYS?
- If using iX, make sure Com 2 setting is RS-485.



Select Time Zone...

Swedish (Sweden)

□ Use SNTP time synchronization

SNTP Server name

Update interval (minutes) 60

☑ Key Beep

□ Automatically turn off backlight

Seconds until backlight is turned off 900

□ Keep backlight on if the notifier window is visible.

COM 1 RS-232

COM 3 RS-485

COM 2 RS-485

### Power failure in BFI-P2 or BFI-E3

### Power 1*230 or 3*400 VAC Turned off to BFI-P2 =>

- Communication is still working with BFI-P2 due to 24 VDC supply from network cables and the other BFI in network.
- Diagnostic Message in CODESYS “EMCY Code 3207, Register 05
- BFI is tripped showing Undervoltage alarm
- Alarm can’t be reseted, disappears when power Turned on again.

### Power 1*230 or 3*400 VAC Turned off to BFI-E3 =>

- BFI-E3 turns black.
- Module not found in CODESYS “Diag String “Polling for Object 0x1000

### Communication is working but BFI-P2/E3 doesn’t run the motor correct:

- Inhibit in display? Safe Torque Off signals are missing to BFI-P2/H3, terminal 12 and 13. Terminal 12 must be connected to a 24 VDC and terminal 13 to 0 VDC.
- BFI-H3, is the Drive in Auto Mode? If not push Auto on keypad.
- Software enable on terminal 2 in place? Must be connected to 24 VDC.
- BFI-P2/H3/E3 is powered up?
- No alarm in BFI-P2/E3/H3?
- Parameter P1-12 = 4 in BFI-P2/H3 and P-12=3 or 4 in BFI-E3?
- Variable Ready to Run must be 1 to run the BFI-P2/E3/H3. Check
  - if not, Software Enable signal on terminal 2=24VDC?
  - if not, Safe Torque Off terminals are properly connected on BFI-P2/H3?
  - if not, 3×400 or 1×230 VAC connected to BFI?
  - BFI-H3, Drive in “Hand mode”
- Stop in display?
  - Start signal by bus is missing. As soon as BFI receives a start signal a value between 0 and Maximum speed is displayed instead of Stop.
- H 0.0 in display?

  Start signal arrived but no speed setpoint
  - Speed sent to BFI-P2/E3/H3 must be a value between parameter value P1-01 and P1-02.

- Speed into Function block is a REAL. Value 50.0 Hz means value 500 being sent to BFI-P2/E3
- Check that sent speed by bus corresponds with value in P0-07 in BFI-P2/H3 and P0-03 in BFI-E3.
- Check parameter P0-52:2 (word2 of the 4 cyclic words) in BFI-P2 and BFI-H3. This is the speed value send in cyclic data word2. The value is Hexadecimal. Mean if 50.0 Hz is sent(Value 500) P0-52:2 should be 01F4 in Hexadecimal. Corresponding register in BFI-E3 is P0-22:2.

## BFI-P2/H3/E3 Modbus Register Mapping

### MODBUS RTU Protocol Specification

The following table highlights the specification for the Modbus RTU implementation in BFI-P2/E3/H3.

| Protocol | Modbus RTU | Setting |
|---|---|---|
| **Error check** | CRC | |
| **Baud rate** | 9600bps, 19200bps, 38400bps, 57600bps,<br>115200bps (default) | Adjustable in Parameter P5-03 in BFI-<br>P2/H3 and P-36 in BFI-E3 |
| **Data format** | 1 start bit, 8 data bits, 1 stop bits, no parity. 1 start bit, 8 data bits, 2 stop bits, no parity. 1 start bit, 8 data bits, 1 stop bits, even parity.<br>1 start bit, 8 data bits, 1 stop bits, odd parity. | Adjustable in Parameter P5-04 in BFI- P2/ H3 and but not in BFI-E3 |
| **Physical sign al** | RS 485 (3-wire) | |

### Modbus Telegram Structure:

The following Modbus Commands are supported

- 03 Read Holding Registers
- 06 Write Single Holding Register
- 16 Write Multiple Holding Register (only register 1-4) The telegram structure is as follow

| Command 03 – Read Holding Registers | | | | | | |
|---|---|---|---|---|---|---|
| Master Telegram | Length | | | Slave Response | Length | |
| Slave Address | 1 | Byte | | Slave Address | 1 | Byte |
| Function Code (03) | 1 | Byte | | Function Code (03) | 1 | Byte |
| 1st Register Address | 2 | Bytes | | Byte Count | 1 | Byte |
| No. Of Registers | 2 | Bytes | | 1st Register Value | 2 | Bytes |
| CRC Checksum | 2 | Bytes | | 2nd Register Value | 2 | Bytes |
| | | | | Etc… | | |
| | | | | CRC Checksum | 2 | Bytes |

| Command 06 – Write Single Holding Register | | | | | | |
|---|---|---|---|---|---|---|
| Master Telegram | Length | | | Slave Response | Length | |
| Slave Address | 1 | Byte | | Slave Address | 1 | Byte |
| Function Code (06) | 1 | Byte | | Function Code 06 | 1 | Byte |
| Register Address | 2 | Bytes | | Register Address | 2 | Bytes |
| Value | 2 | Bytes | | Register Value | 2 | Bytes |
| CRC Checksum | 2 | Bytes | | CRC Checksum | 2 | Bytes |
| | | | | | | |

**Parameter Access Overview BFI-P2, BFI-H3 and BFI-E3**

Modbus addresses for control, parameters and Actual status is to be found in Application note or Manuals for BFI-H3, BFI-E3 and BFI-P2.

## About Beijer Electronics

Beijer Electronics is a multinational, cross-industry innovator that connects people and technologies to optimize processes for business-critical applications. Our offer includes operator communication, automation solutions, digitalization, display solutions and support. As experts in user-friendly software, hardware and services for the Industrial Internet of Things, we empower you to meet your challenges through leading-edge solutions. Beijer Electronics is a Beijer Group company.
Since its start-up in 1981, BEIJER GROUP has evolved into a multinational group with sales of 1.4 billion SEK 2020. BEIJER GROUP is listed on the NASDAQ Stockholm Main Market under the ticker BELE.
[www.beijergroup.com](http://www.beijergroup.com)

China

Denmark

France

Germany

India

Norway

South Korea

Sweden HQ

Taiwan

Turkey

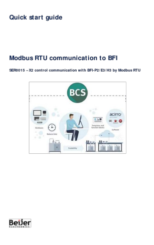
United Kingdom

USA

BeNeLux

**Contact us**
Global offices and distributors

## Documents / Resources

| | |
|---|---|
| <br>Quick start guide<br>Modbus RTU communication to BFI<br>SER0015 – X2 control communication with BFI-P2/E3/H3 by Modbus RTU<br>Beijer | [Beijer ELECTRONICS SER0015 X2 Control Communication](#) [pdf] User Guide<br>SER0015 X2 Control Communication, X2 Control Communication, SER0015 Communication, Communication, X2 Control, SER0015 X2 |

## References

- Beijer Electronics

Manuals+,