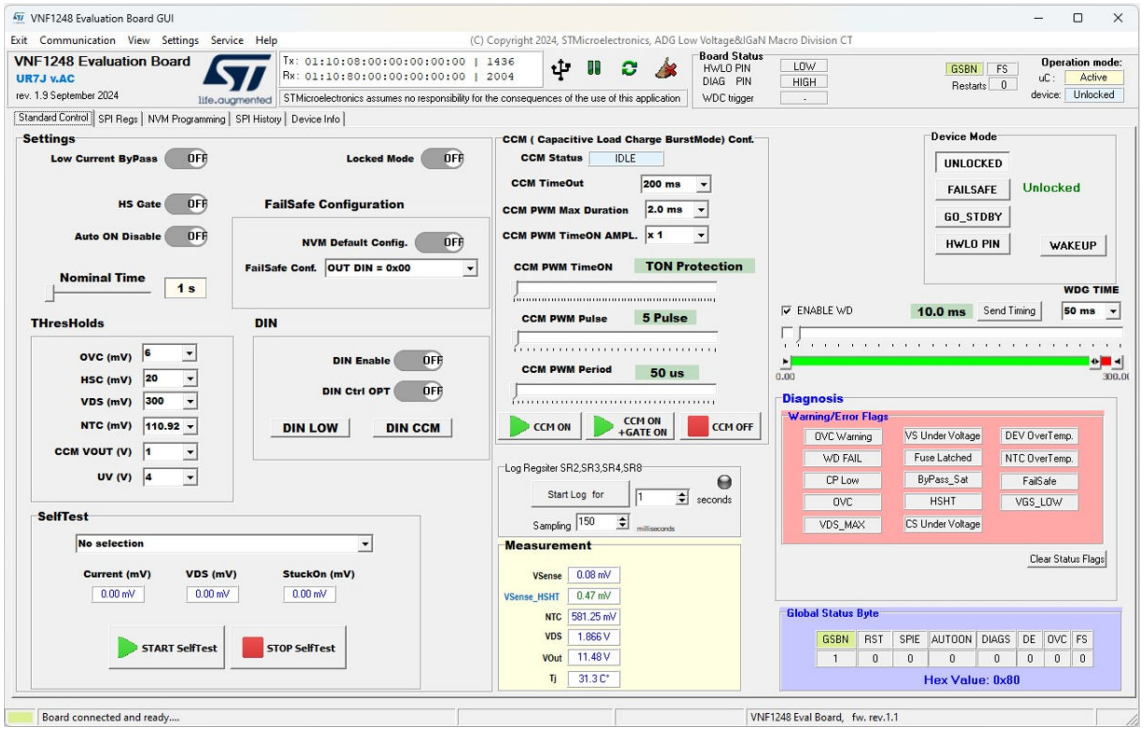
Getting started with STSW-EV-VNF1248F software package for EV-VNF1248F

Introduction

The STSW-EV-VNF1248F is the companion software package of the EV-VNF1248F. It contains the graphical user interface (GUI) and the firmware to be flashed onto the EV-SPC582B MCU board.

The combined system of STSW-EV-VNF1248F and EV-VNF1248F offers a simple evaluation system for the VNF1248F device, an advanced controller with eFuse algorithm driving high-side MOSFET in 12 V, 24 V and 48 V automotive applications.

Figure 1. STSW-EV-VNF1248F Graphical User Interface



1 Get software

To setup and control the EV-VNF1248F evaluation board through the dedicated Graphical User Interface, a software environment must be acquired, and specifically:

1. The STSW-EV-VNF1248F software package, which includes the GUI application and the related control firmware to flash onto the EV-SPC582B microcontroller board
2. The SPC5-UDESTK Starterkit software application to be used to flash the control firmware on the EV-SPC582B MCU board

1.1 Graphical User Interface and Control Firmware

Visit the dedicated page for STSW-EV-VNF1248F: from the first tab **Overview**, a button is available to get the latest release of the software package in the form of a zip file. It includes the firmware package to control the EV-VNF1248F evaluation board and a folder with the GUI application to be launched on a desktop PC to communicate with the VNF1248F device.

Select the tab **Documentation** to retrieve the available documentation about the software package, and the tab **Tools & Software** to find the reference to the EV-VNF1248F ST web page.

1.2 Software ecosystem

If you have a pre-flashed EV-SPC582B MCU board with the control firmware for EV-VNF1248F evaluation board, you can skip this section.

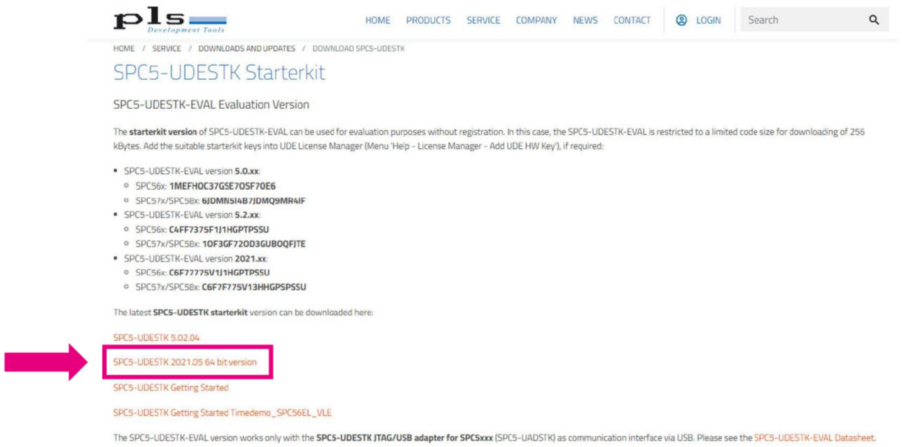
On the contrary, if you need to program the EV-SPC582B MCU board, a software package installed on your computer is required: SPC5-UDESTK Starterkit.

UDE Starterkit 2021

The UDE tool **SPC5-UDESTK 2021.05 64-bit** version (file: **udestk-2021-05-spc5-stk.exe**) is available at:

https://www.pls-mc.com/support/downloads/download-spc5-udestk/

Figure 2. SPC5-UDESTK 2021.05 64-bit



2 Software installation

The software ecosystem needed to setup a PC workstation to run the Graphical User Interface provided in STSW-EV-VNF1248F software package is now described.

2.1 Firmware programming tool installation

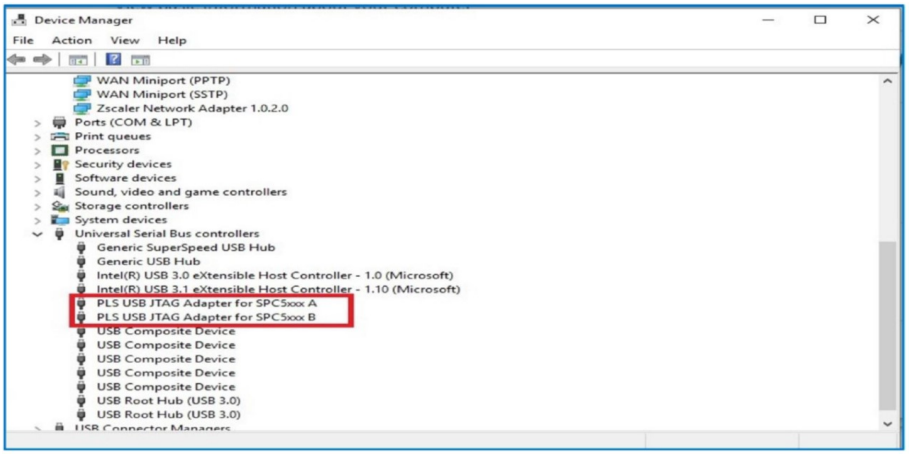
To program the control firmware in the EV-SPC582B MCU board, if needed, the **UDE Starterkit 2021** application must be installed on your PC workstation.

PLS UDE Starterkit 2021

Run the **udestk-2021-05-spc5-stk.exe** utility and follow the installation wizard. As a final step of the installation process, UDE STK driver is also installed.

After driver installation is finished, connect the USB cable to the EV-SPC582B and open the **Device Manager** in Windows OS: these two PLS controllers must have been recognized if everything went well.

Figure 3. Universal Serial Bus controllers

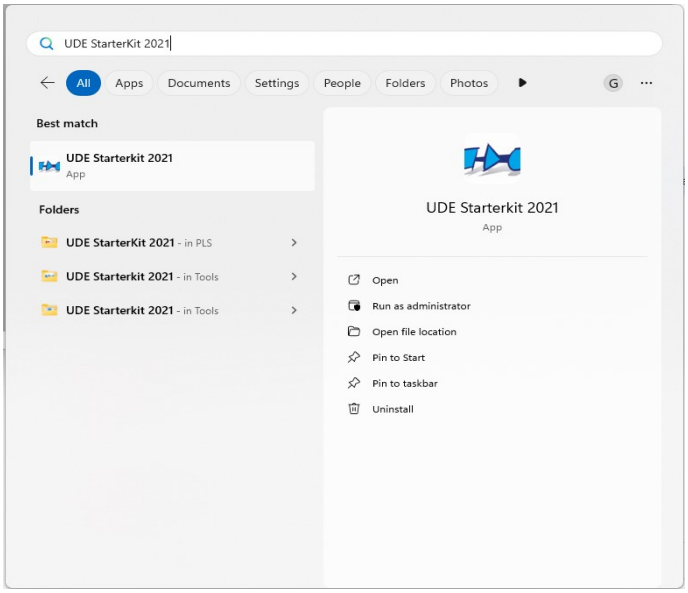


2.2 Firmware programming

The steps required to perform the firmware programming operation on the microcontroller board are detailed below.

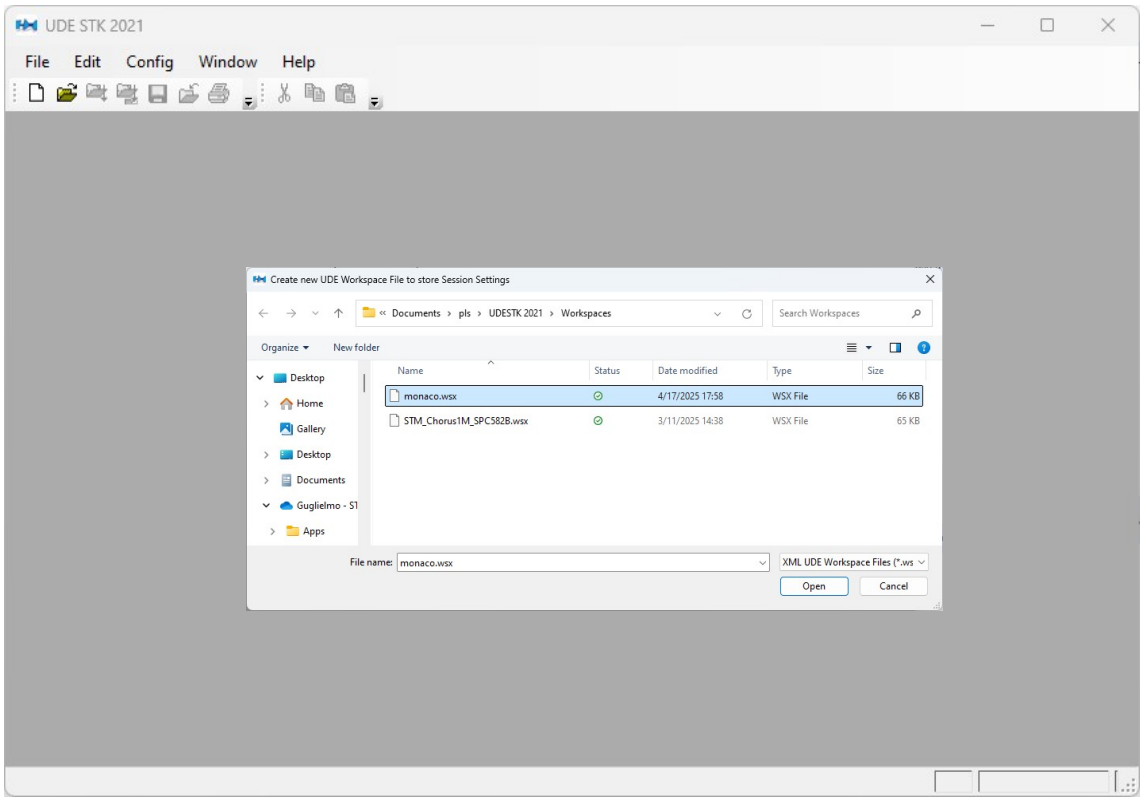
1. Setup the EV-SPC582B MCU board by putting jumpers on JP3 and JP4 on the right side (5 V) and connecting USB cable for board communication and programming.
2. Run **UDE Starterkit 2021** application on PC workstation

Figure 4. UDE Starterkit 2021



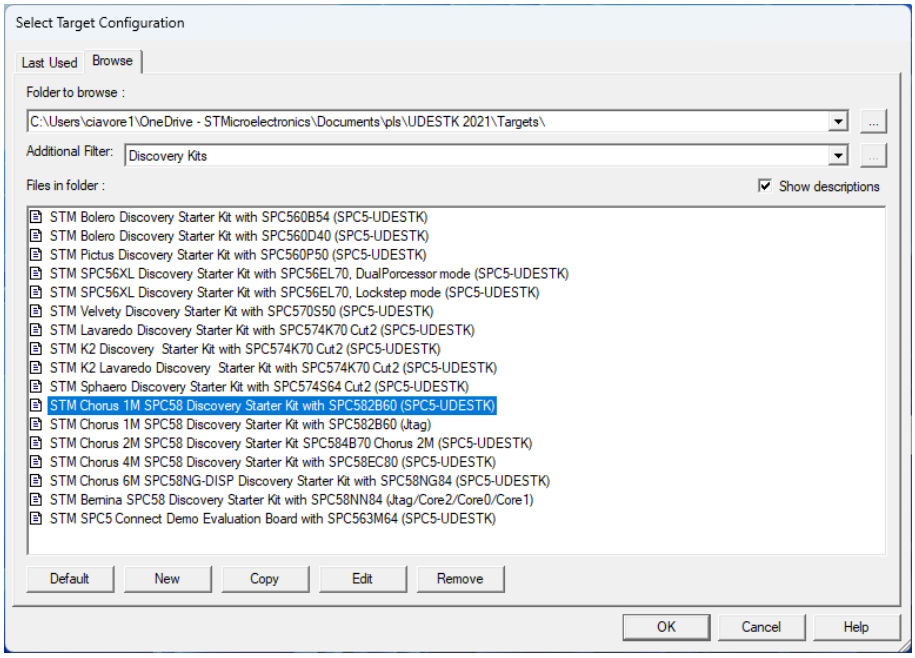
3. Select from **File** menu, the item **New Workspace**, and type a name for the workspace to be created, or overwritten if it already exists. In this example, the new workspace is named monaco.wsx

Figure 5. New workspace creation



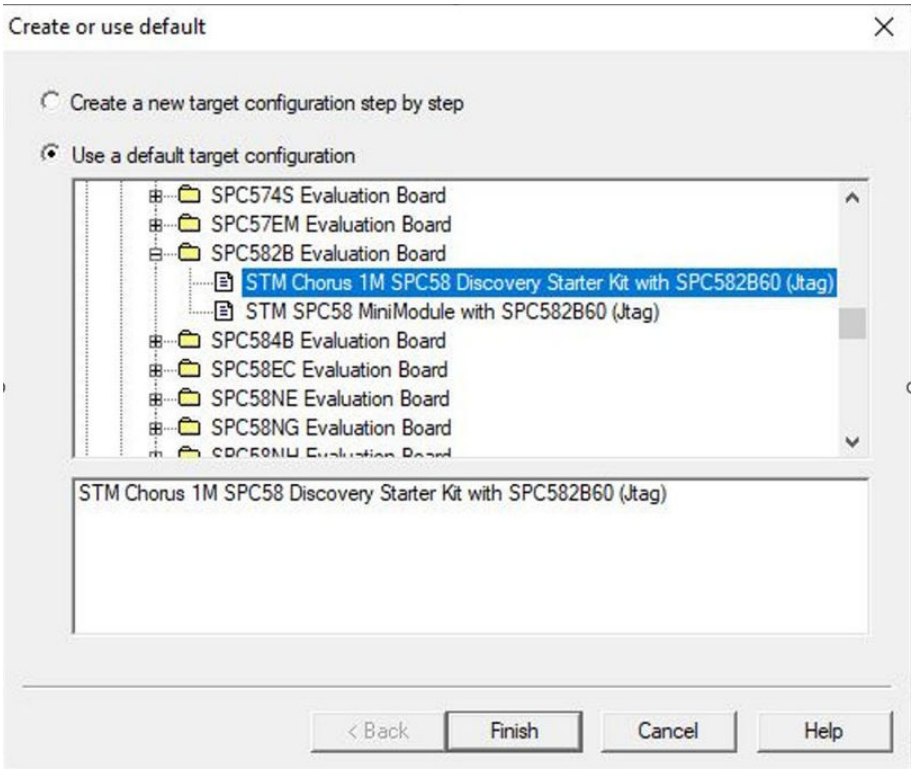
4. **Select Target Configuration** window appears: set **Additional Filter: Discovery Kits**, and then click on **Default**.

Figure 6. Select Target Configuration



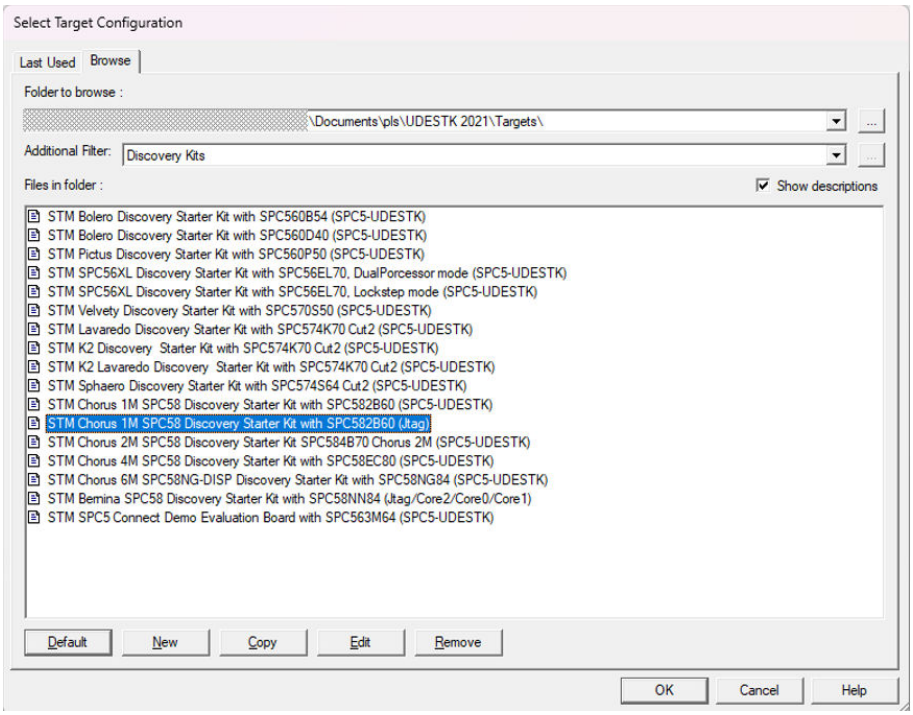
5. In the new dialog window, **Create or use default**, select **Use a default target configuration** and, in the Device tree dialog below, select in order:
   PowerPC
   ↳**STM**
   ↳**SPC582B Evaluation Board**
   ↳**STM Chorus 1M SPC58 Discovery Starter Kit with SPC582B60 (Jtag)**
   Press **Finish**: choose a name for this new configuration file

Figure 7. Create or use default



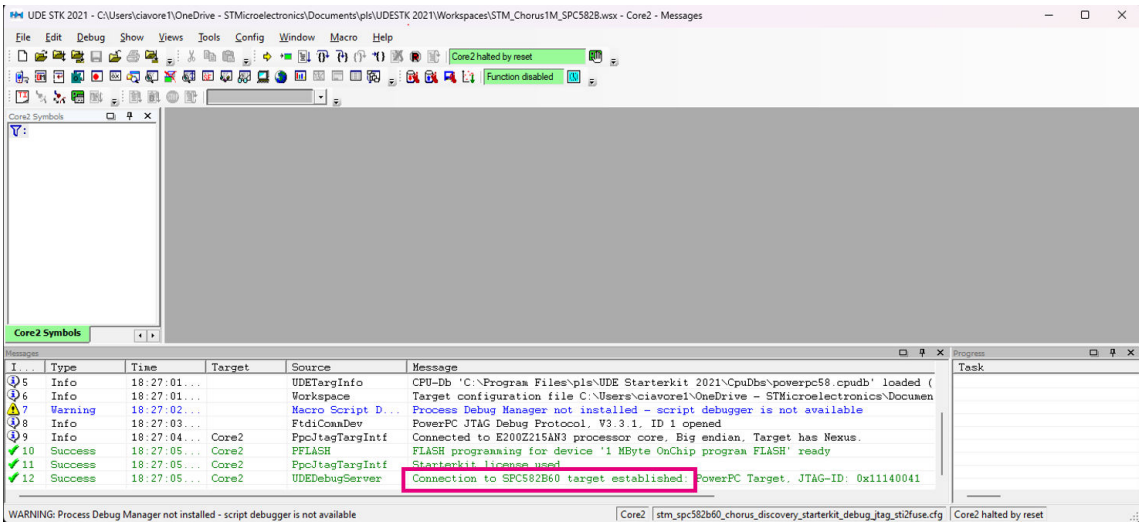
6. Back in the **Select Target Configuration**, now select:
STM Chorus 1M SPC58 Discovery Starter Kit with SPC582B60 (Jtag) and then press **OK**

Figure 8. Select Target Configuration dialog window



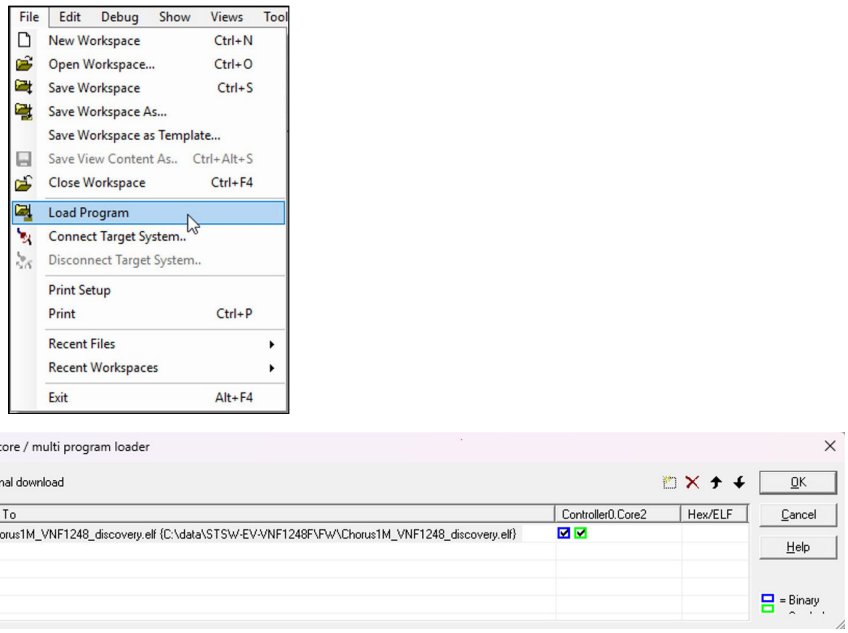
7. In the **UDE STK 2021** window, verify that in the **Messages** window contains a line indicating that a successful connection to the target has been established:

Figure 9. Connection established



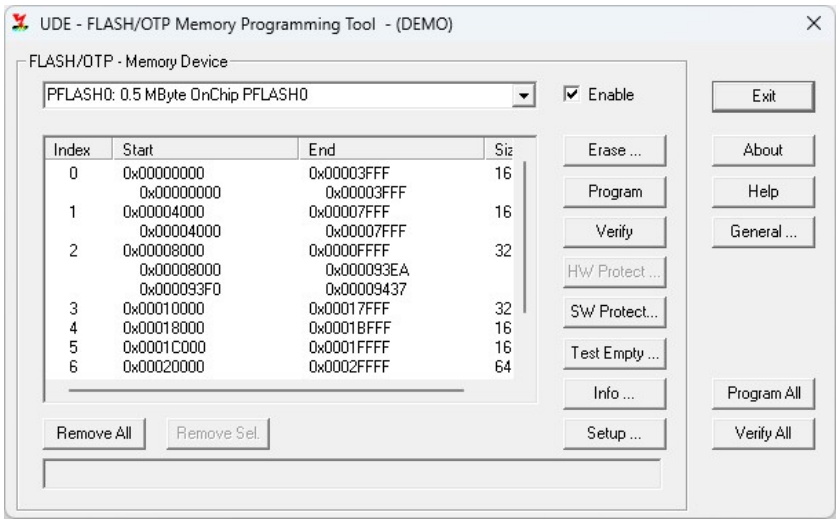
8. Select **Load Program** from **File** menu, browse your hard drive to select the provided control firmware (**.elf** file) and then press **OK** in the **Multicore / multi program loader** dialog window:

Figure 10. Load Program dialog windows



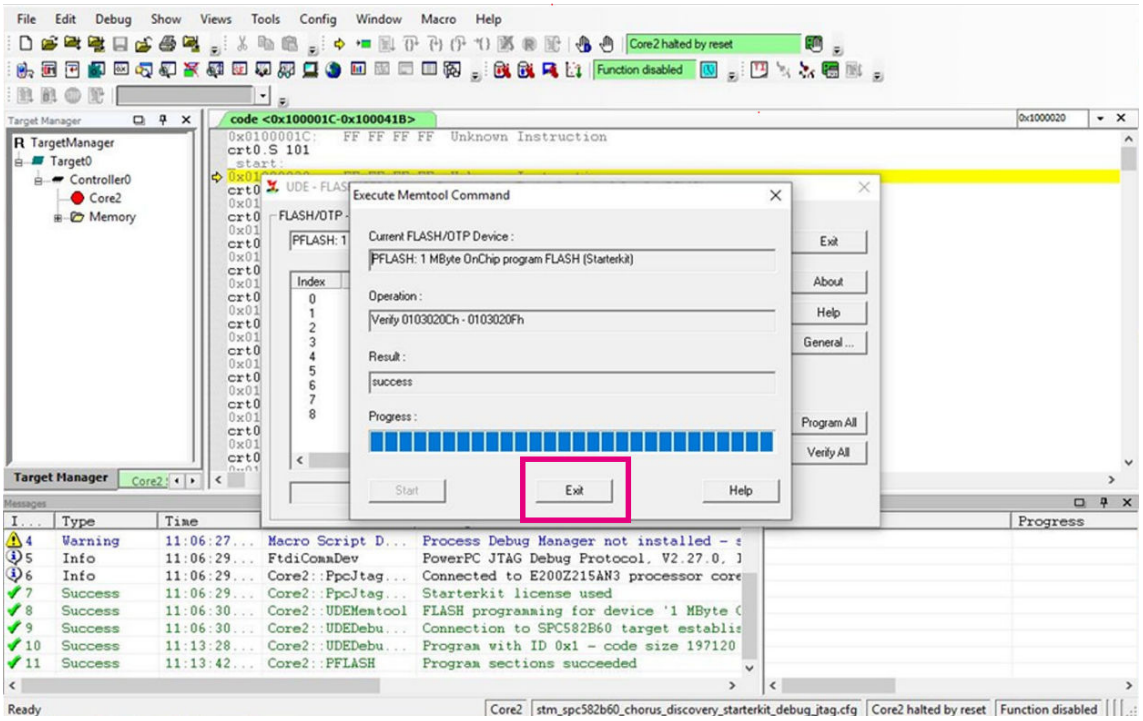
9. Press **Cancel** for any requests about Controller0.Core source files (**.c** or **.h**)
10. The **UDE - FLASH/OTP Memory Programming Tool - (DEMO)** window now appears. Press **Program All**:

Figure 11. FLASH/OTP Memory Programming Tool dialog window



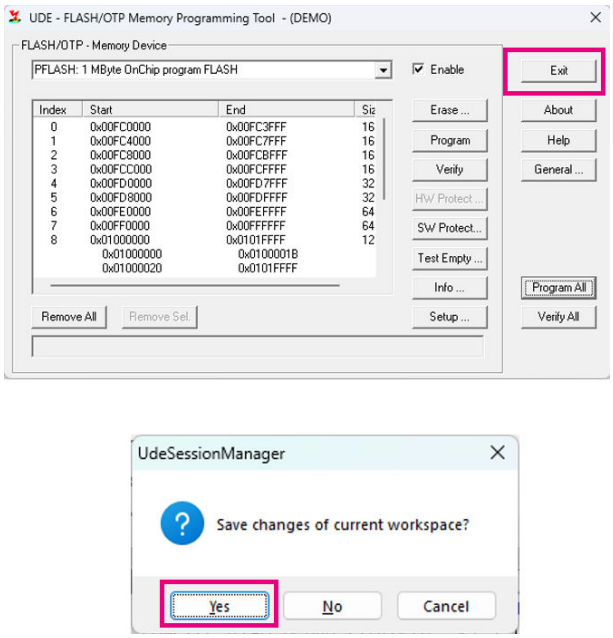
11. Follow the execution of programming operation through the **Execute Memtool Command** dialog window. At the end of the operation, check the **Result:** field reporting **success**, and then press **Exit**:

Figure 12. FLASH/OTP Memory Programming



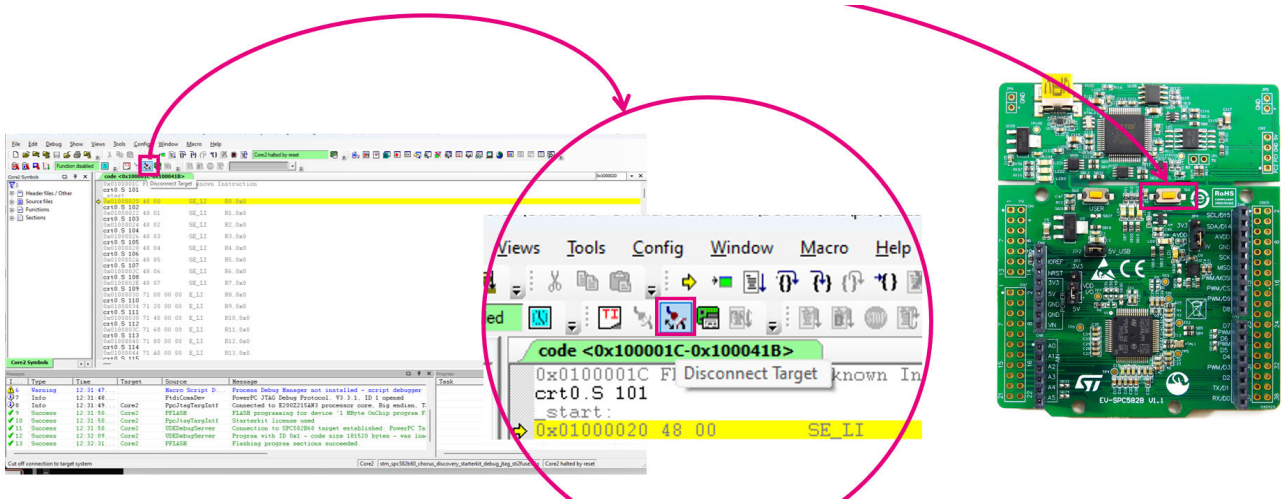
12. Press **Exit** also in the **UDE - FLASH/OTP Memory Programming Tool - (DEMO)** dialog window, and press **Yes**, if requested, to save workspace changes for future use:

Figure 13. FLASH/OTP Memory Programming in progress



13. To complete the firmware programming, simply disconnect the GUI from the target (see next figure) and reset it through the **RESET** push-button available on the EV-SPC582B microcontroller board:

Figure 14. Target disconnected



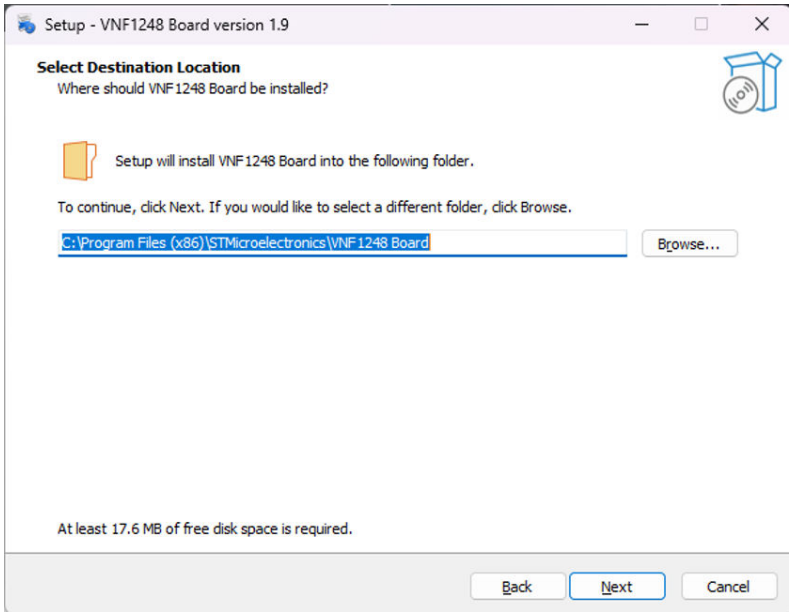
2.3 Graphical User Interface installation

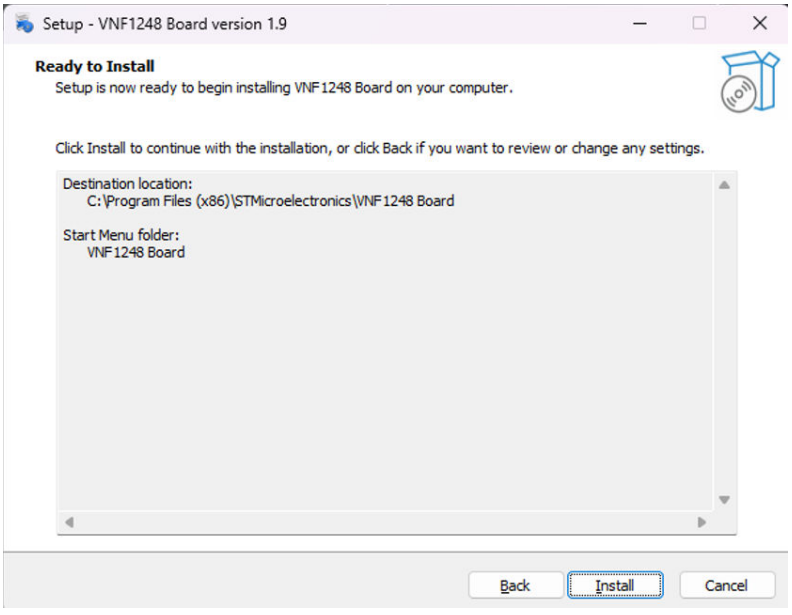
To install the Graphical User Interface, the following steps must be executed.

1. Run the application provided in STSW-EV-VNF1248F: **Setup_VNF1248_vx.y.exe**

Note: x.y is a generic version number, reported only as example

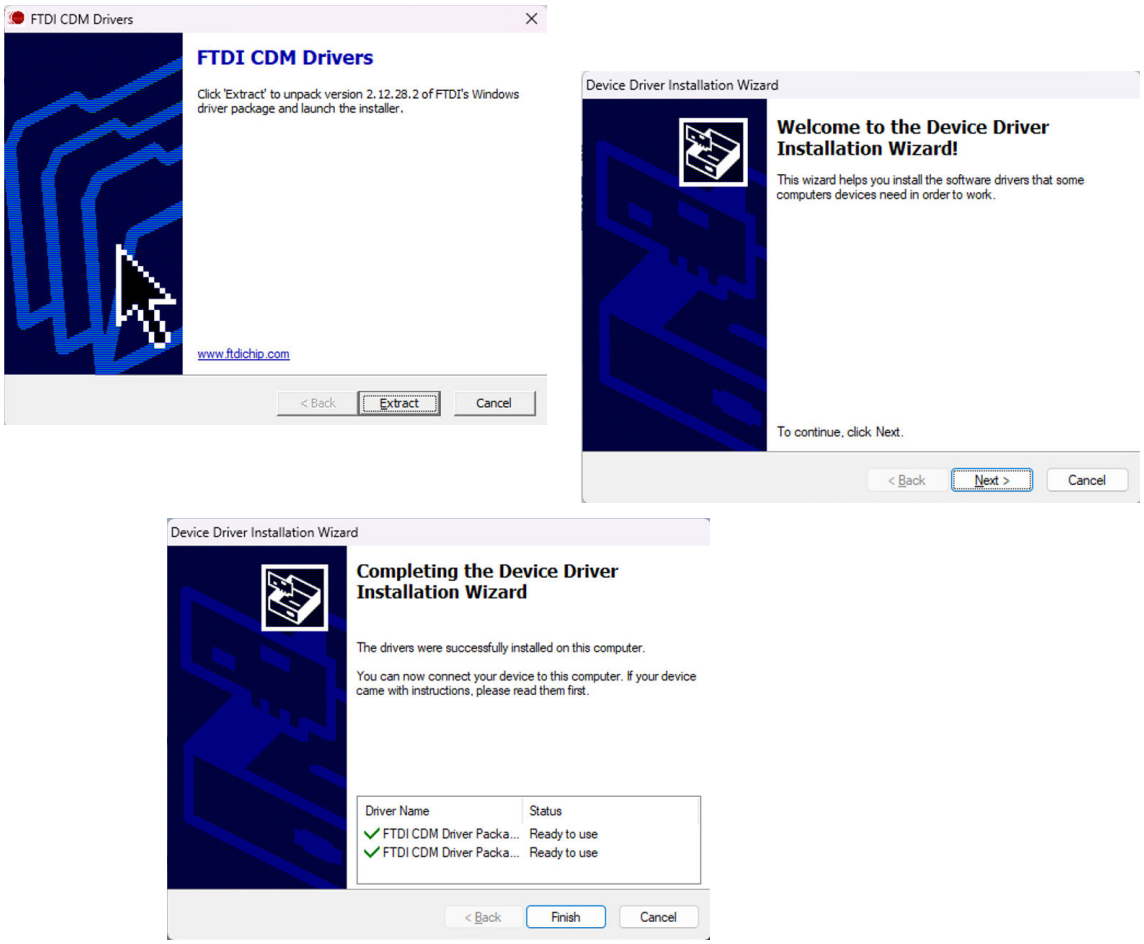
Figure 15. Setup - VNF1248 Board





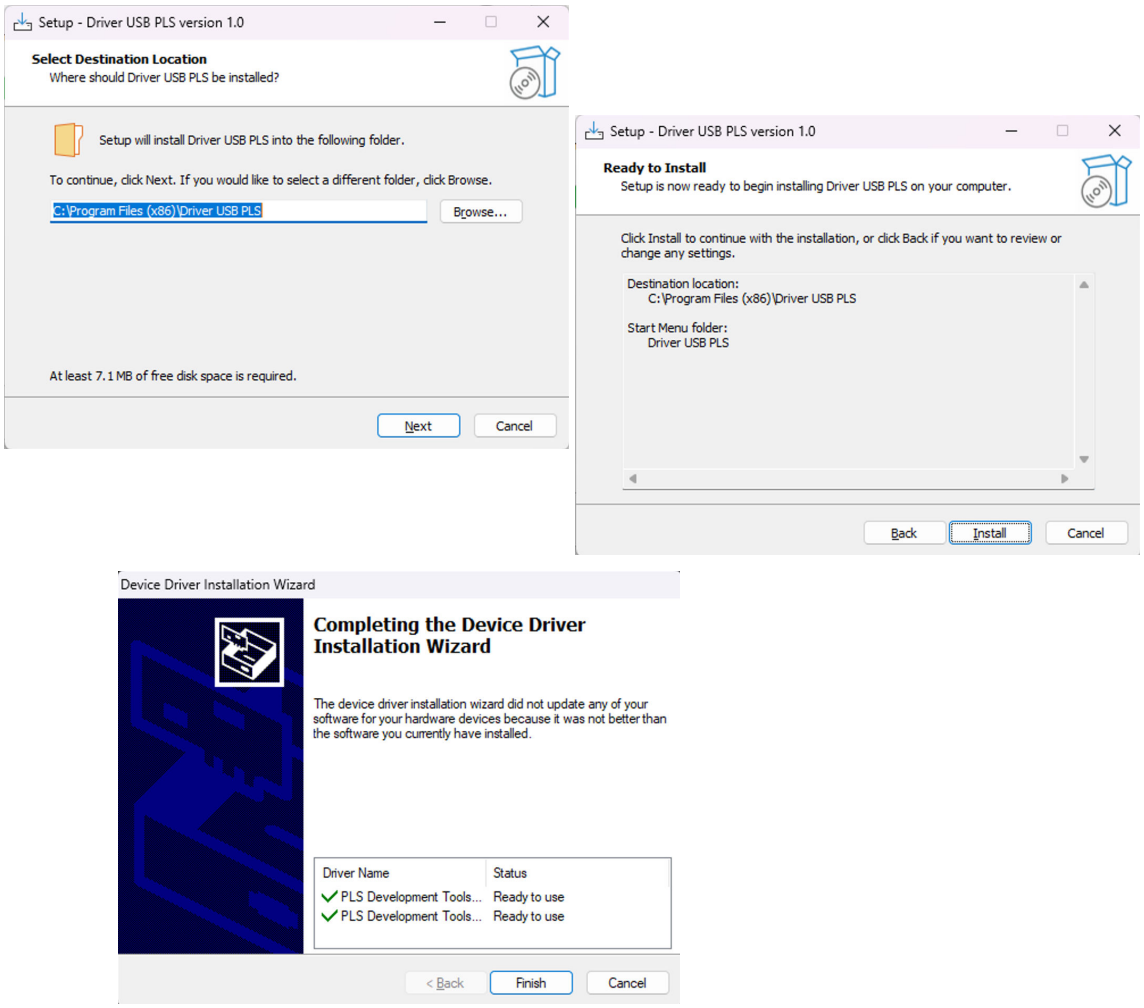
2. **FTDI COM** Drivers installation now starts:

Figure 16. FTDI COM Drivers



3. **USB PLS** Drivers will now be installed:

Figure 17. Setup - USB PLS Drivers



4. Press **Finish** to close the installation wizard, and launch **VNF1248 Evaluation board GUI** application.

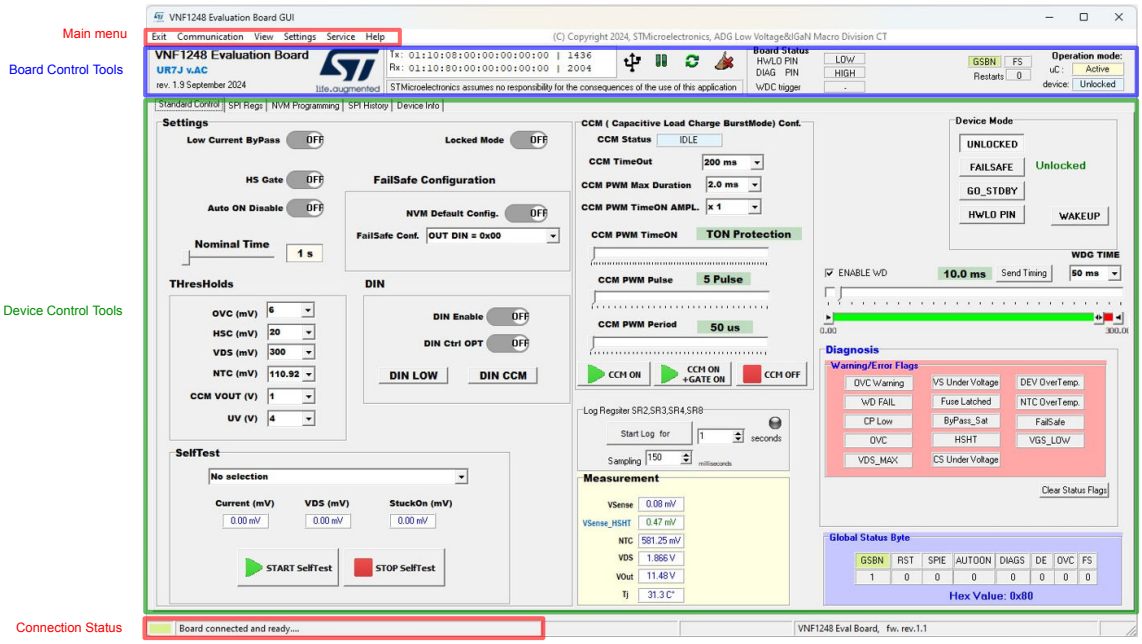
3 Graphical User Interface description

The Graphical User Interface included in the STSW-EV-VNF1248F software package is a powerful tool designed to setup and control the VNF1248F device embedded in the EV-VNF1248F evaluation board.

Launch the **VNF1248 Board** application, or run the **VNF1248_EvalBoard.exe** related executable, to start the GUI.

The Graphical User Interface is composed by some sections that will be detailed in the next paragraphs. Below is an overview of the GUI graphic.

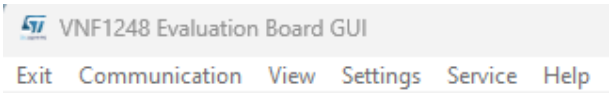
Figure 18. GUI sections



3.1 Main menu

The Graphical User Interface includes a command menu that allows you to perform some application settings, access support services, execute Windows OS operations, and visualize device data. Main menu items are described below in more details.

Figure 19. Main menu

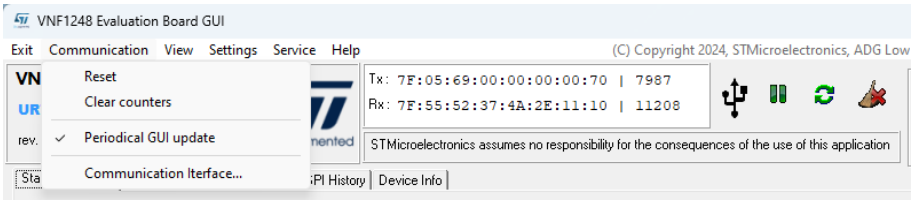


Exit

It simply closes the application

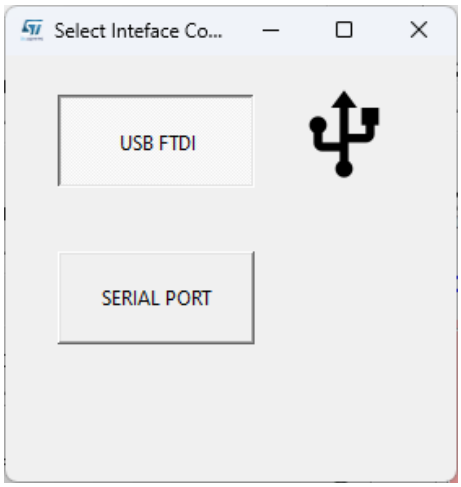
Communication

Figure 20. Communication



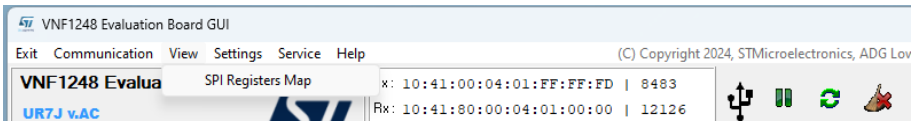
The proposed menu items allows you to reset communication traffic, clear all related counters, refresh the GUI periodically, and, most importantly, select the type of interface for the specific hardware connected. You can choose between a USB FTDI interface and a serial interface.

Figure 21. Communication Interface selection



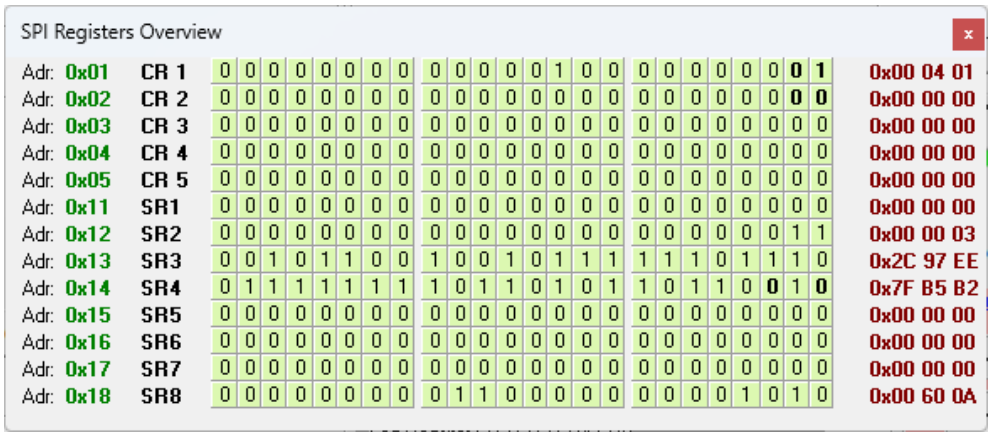
View

Figure 22. View



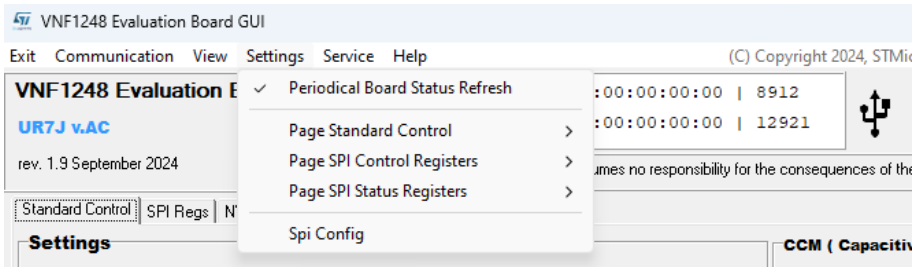
The SPI register map can be shown:

Figure 23. SPI Registers Overview



Settings

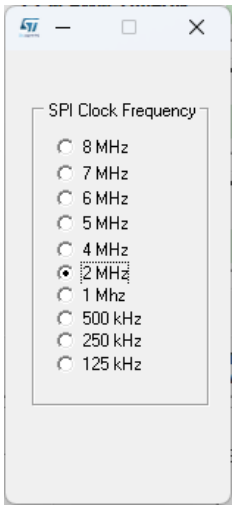
Figure 24. Settings



To configure periodical refresh of registers is allowed here.
Specific menu items are available to periodically:

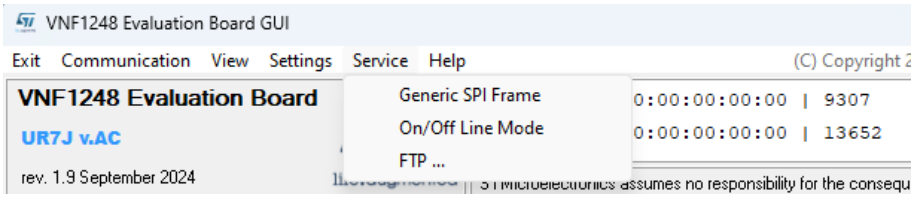
- refresh board status
- refresh and/or clear SPI Status Regs
- refresh SPI Control Regs
- config SPI Clock Frequency

Figure 25. SPI Clock Frequency



Service

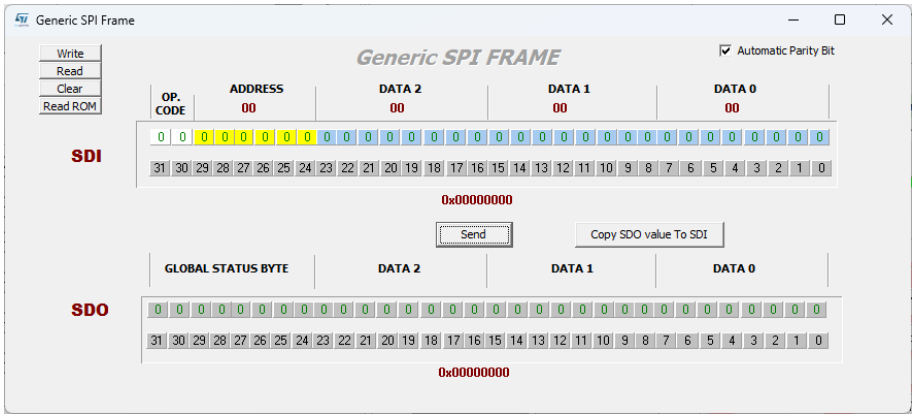
Figure 26. Service



Some services are available:

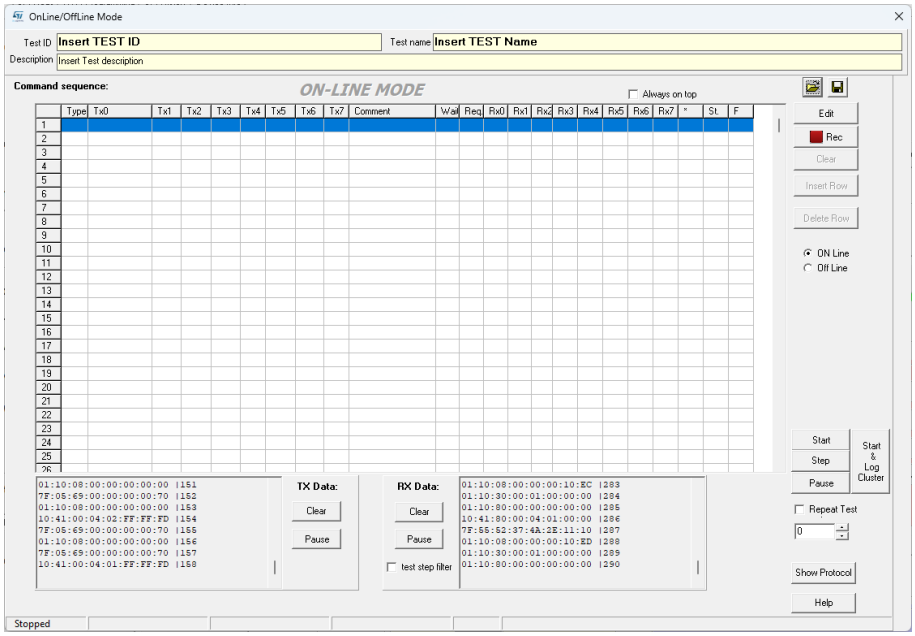
- **Generic SPI Frame** allows to send a customizable SPI frame to the VNF1248F device and to read its answer:

Figure 27. Generic SPI Frame



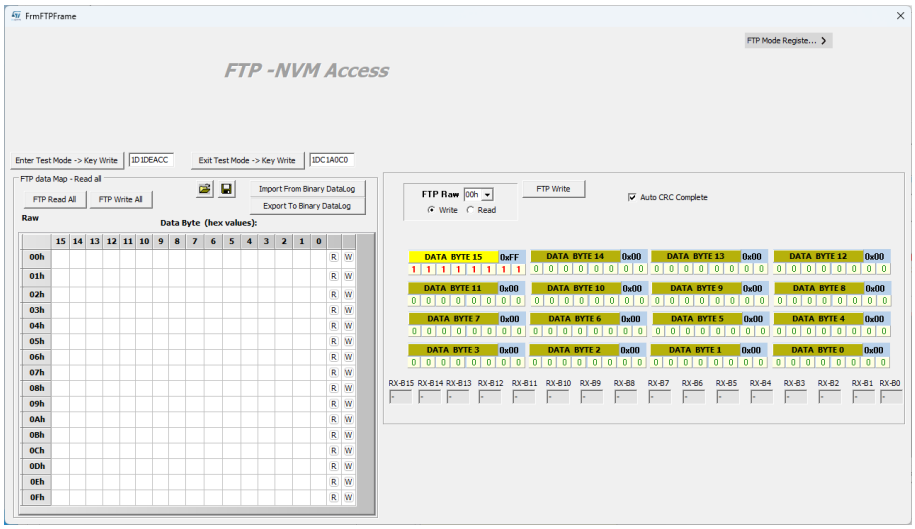
- **On/Off Line Mode** provides a powerful test and debug utility:

Figure 28. On/Off Line Mode



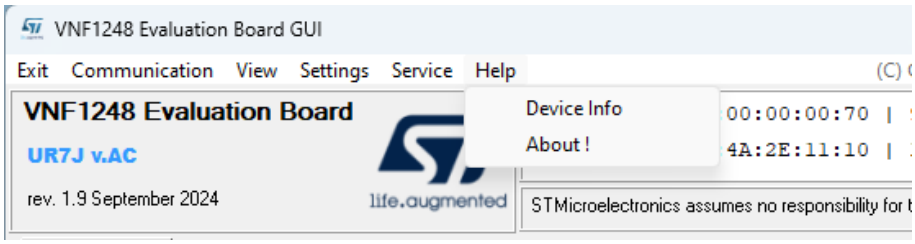
- **FTP** menu item is related to the Few Times Programmable feature of the embedded non-volatile memory, and specifically to customer sector program/erase/read operations:

Figure 29. FTP



Help

Figure 30. Help



3.2 Board control tools

The Graphical User Interface shows, under the main menu area, a graphical section that reports details about the GUI version, the VNF1248F device silicon cut, the communication traffic with the EV-VNF1248F evaluation board, some buttons with shortcuts to common actions, and the EV-VNF1248F board status.

Figure 31. Board Control Tools

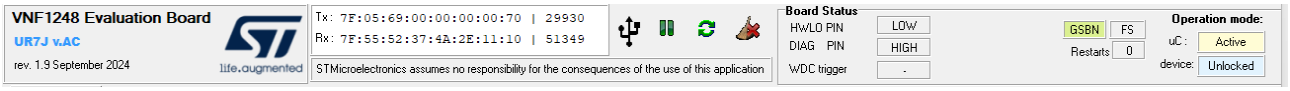


Figure 32. Common actions

Communication interface selection

Enable/Disable periodical reading of status registers and SPI traffic

Refresh all registers (control and status ones)

Clear all status registers

3.3 Connection status

At the bottom of the Graphical User Interface, a Connection Status strip is presented. The normal application operation (communication between MCU board and GUI correctly established) is reported below. In case of missing communication, a red flag is shown in place of the green one here reported, and an error message is displayed.

Figure 33. Connection status

3.4 Device control tools

The most part of the Graphical User Interface is used for the pages dedicated to setup and control the VNF1248F device and its various features accessible through the EV-VNF1248F evaluation board. The pages are accessible from the tabs available on top of the section.

3.4.1 Standard Control

The main tab in the Graphical User Interface gives access to the main VNF1248F device features:

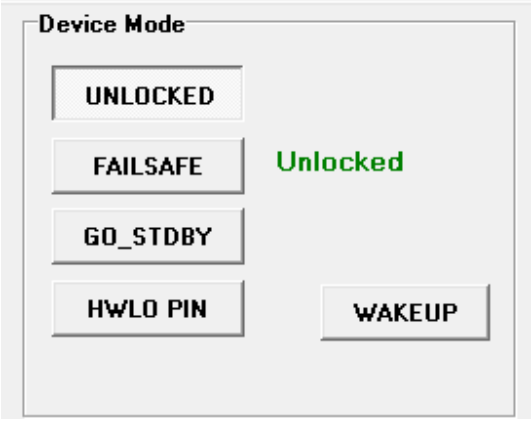
- Different device modes
- Device settings
- Different thresholds
- Perform self-test
- Log register data
- Perform measurements
- Configure CCM, DIN, Fail-safe features
- Retrieve diagnostic flags and Global Status data

Device Mode

All the device modes can be set by pressing the proper button.

HWLO PIN button enables/disables device HWLO pin.

Figure 34. Device Mode

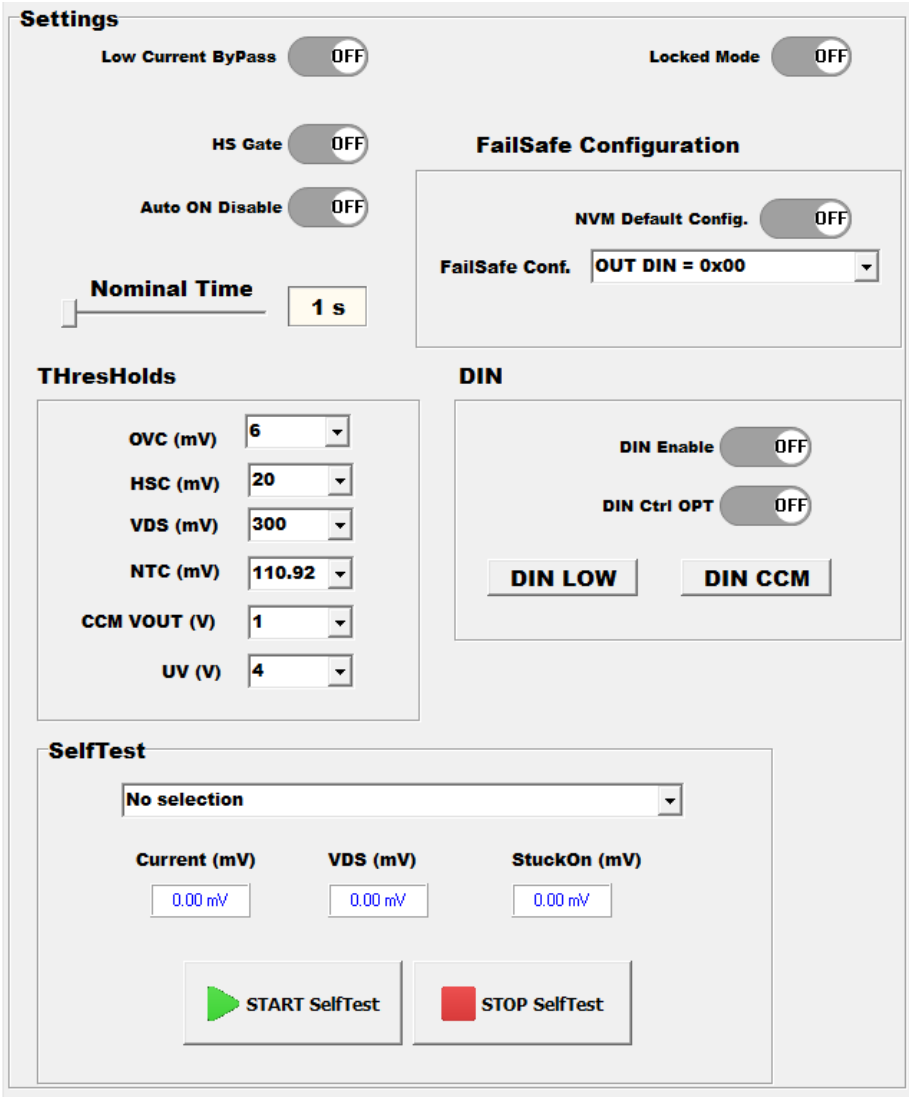


Settings

General settings for VNF1248F device are available:

- **Low Current ByPass**: enables/disables the embedded P-channel Bypass, setting/resetting bit 3 of the Control Registers 1
- **HS Gate**: enables/disables the external MOSFET (and consequently turns ON/OFF the OUT line), setting/resetting bit 4 of Control Register 1
- **Auto ON Disable**: disables the automatic ON state of the external MOSFET in case of Low current bypass switch saturation fault, forced through HS_GATE output, setting bit 23 of Control Register 1
- **Nominal Time**: Configures the fuse nominal time, setting bits from 23 to 16 of Control Register 2

Figure 35. **Settings**



THresHolds

Several thresholds can be configured:

- **OVC (mV)**: Configures the value of Nominal Overcurrent Threshold, setting bits from 15 to 11 of Control Register 2
- **HSC (mV)**: Configures the threshold for Hard Short Circuit Latch-off, setting bits from 10 to 7 of Control Register 2
- **VDS (mV)**: Configures the threshold for External MOSFET desaturation shut-down, setting bits from 6 to 2 of Control Register 2
- **NTC (mV)**: Configures the threshold for External MOSFET overtemperature shutdown via NTC, setting bits from 8 to 5 of Control Register 3
- **CCM VOUT (V)**: Configures capacitive charge mode VOUT threshold, setting bits from 16 to 14 of Control Register 3
- **UV (V)**: Configures VS supply undervoltage threshold, setting bits from 11 to 10 of Control Register 3

SelfTest

This section allows to select the Self-test to be executed (setting bits from 7 to 5 of Control Register 1) and to start and stop the test (setting bit 9 and bit 8 of Control Register 1).

A feedback on test result is also shown (value of Status register 5, 6, 7).

FailSafe Configuration

This section allows to configure the Fail-safe state behavior (setting bits from 15 to 14 of Control Register 1).

DIN

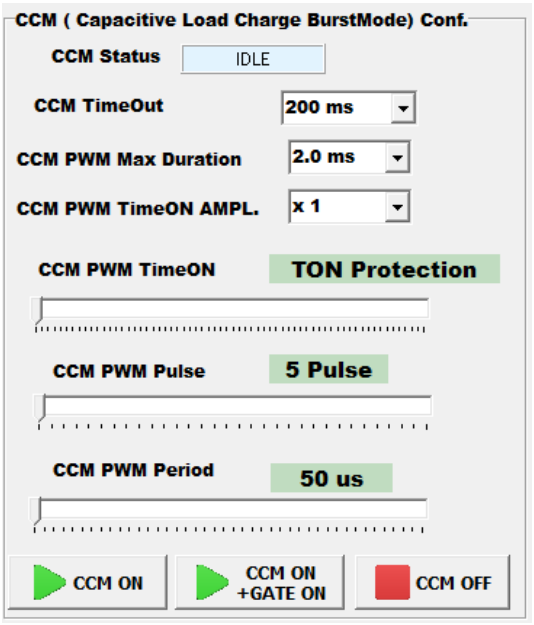
DIN can be used to enable capacitive charging mode functionality when device is in fail-safe state. DIN control enable can be accessed with setting bit 12 of Control Register 1 and its behavior configured setting bit 13 of Control Register 1.

Capacitive Charging Mode

Digital logic implements a PWM-based control of high-side driver to allow charging a capacitive load through the off-chip Power MOSFET driven by the device.

The CCM section of the GUI allows to fully configure (some features are NVM mapped, other selectable through Control Register 5 bits), start (bit 18 Control Register 1) and stop (bit 19 Control Register 1) this feature.

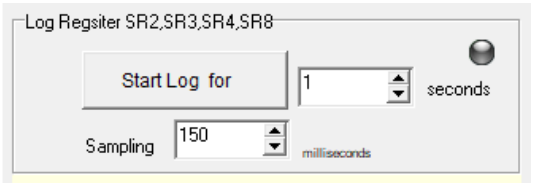
Figure 36. CCM



Log Register data

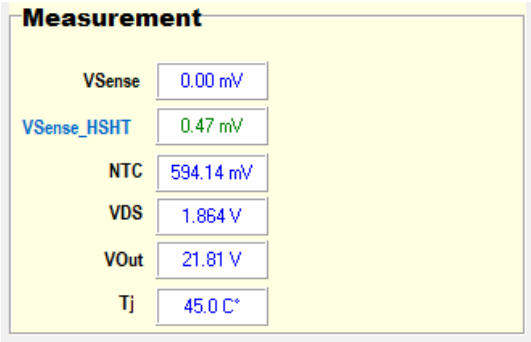
A set of Status Registers can be periodically read from device according to the duration and sampling rate selected.

Figure 37. Log Registers



Measurement

Figure 38. **Measurement**

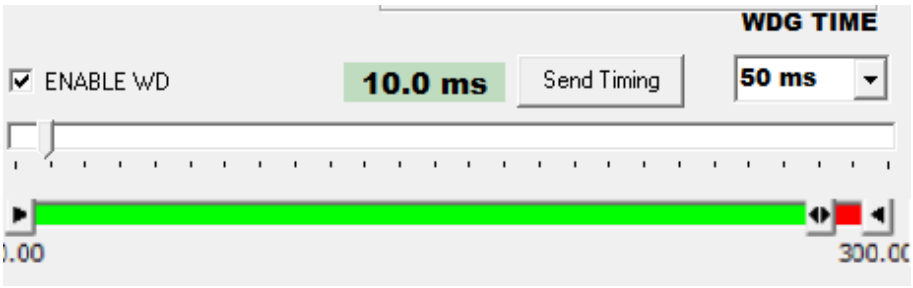


This section gives an overview about the following status registers:

- **VSense**: bits from 14 to 2 of Status Register 2
- **Vsense_HSHT**: bits from 11 to 2 of Status Register 8
- **NTC**: bits from 11 to 2 of Status Register 3
- **VDS**: bits from 22 to 13 of Status Register 4
- **VOut**: bits from 11 to 2 of Status Register 4
- **Tj**: bits from 22 to 13 of Status Register 3

Watchdog

Figure 39. **Watchdog**



WD serving is applied by refreshing the WD_TRIG bit in one of the control registers.

Enabled WD: enables/disables WD serving by refreshing the WD_TRIG bit

Period for Watchdog (WD) serving is adjustable by item **WDG TIME**.

There is also the possibility of setting the **WD refresh time** sent by MCU through a dedicated bar and button **Send Timing**. This allows to test the device WD timeout failure.

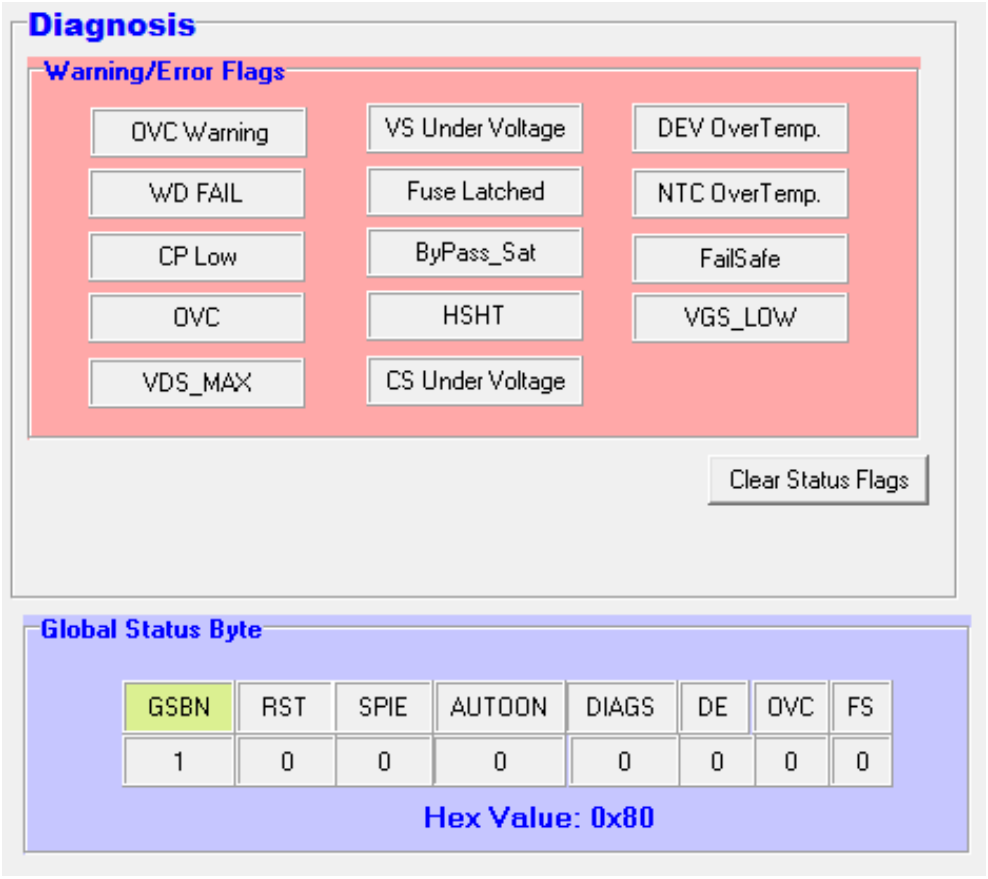
Diagnostics

Diagnostic data is refreshed according to the device mode selected.

Clear Status Flags button allows to clear the error bits in the related status register.

The **Global Status Byte** is also presented.

Figure 40. **Diagnostics**



3.4.2 SPI Regs

The Graphical User Interface in this form shows the content of all SPI Registers: Control Registers 1-5 and Status Registers 1-8.

Each register has its own set of applicable actions.

The possible actions for the RAM registers are the following:

- Button **R**: read register content from the device
- Button **W**: store current content of the selected register to the device
- Button **C**: read and clear action on selected register

Register bits that can be modified are available for editing.

The Graphical User Interface provides side-by-side access to all the SPI Control Registers and SPI Status Registers of VNF1248F device in this form.

Note: For Ctrl Reg4 both Function 1 and Function 2 register bits are shown. For more detailed description of all SPI registers refer to the device datasheet (DS14109)

Figure 41. **SPI Control Registers**

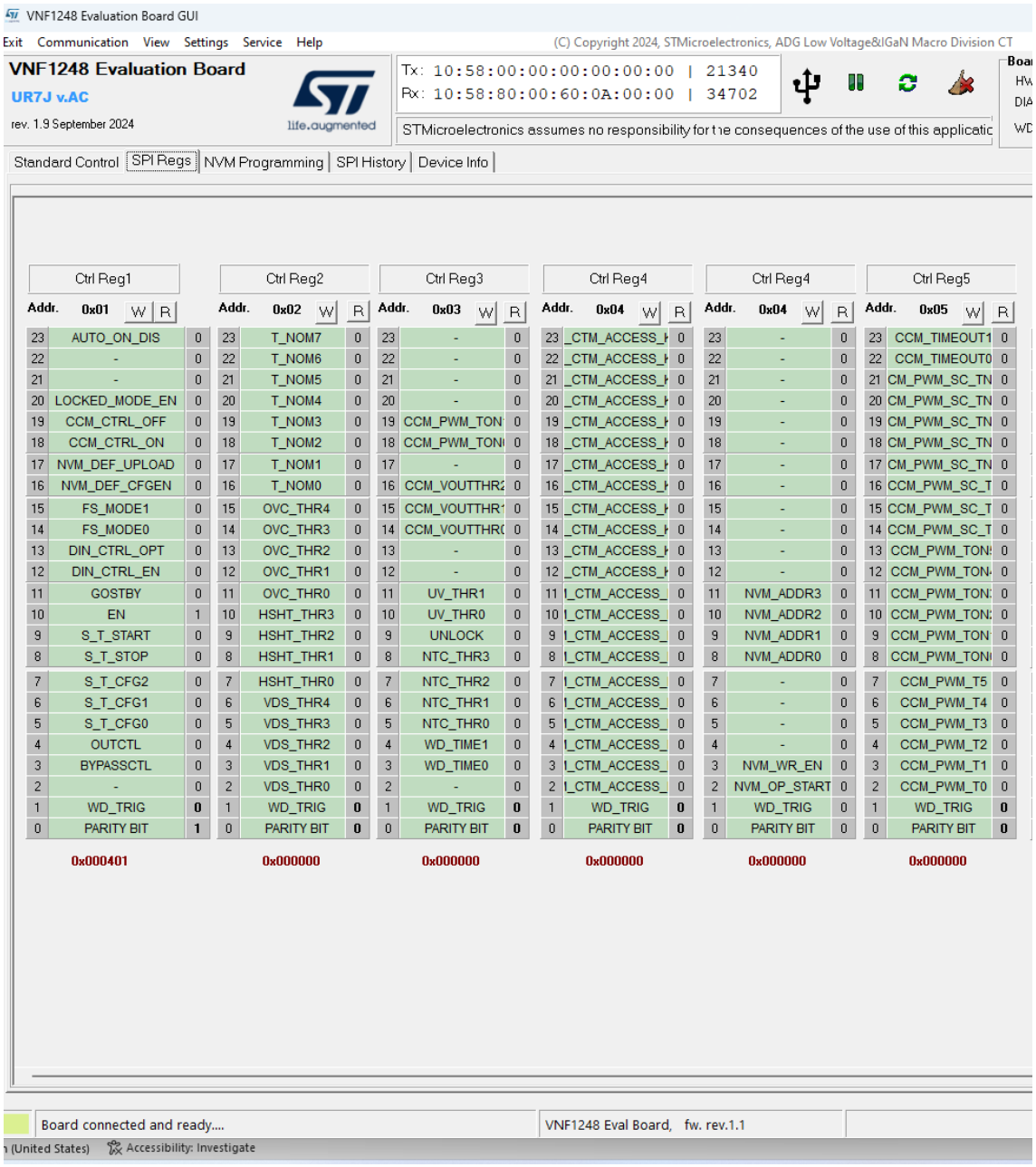
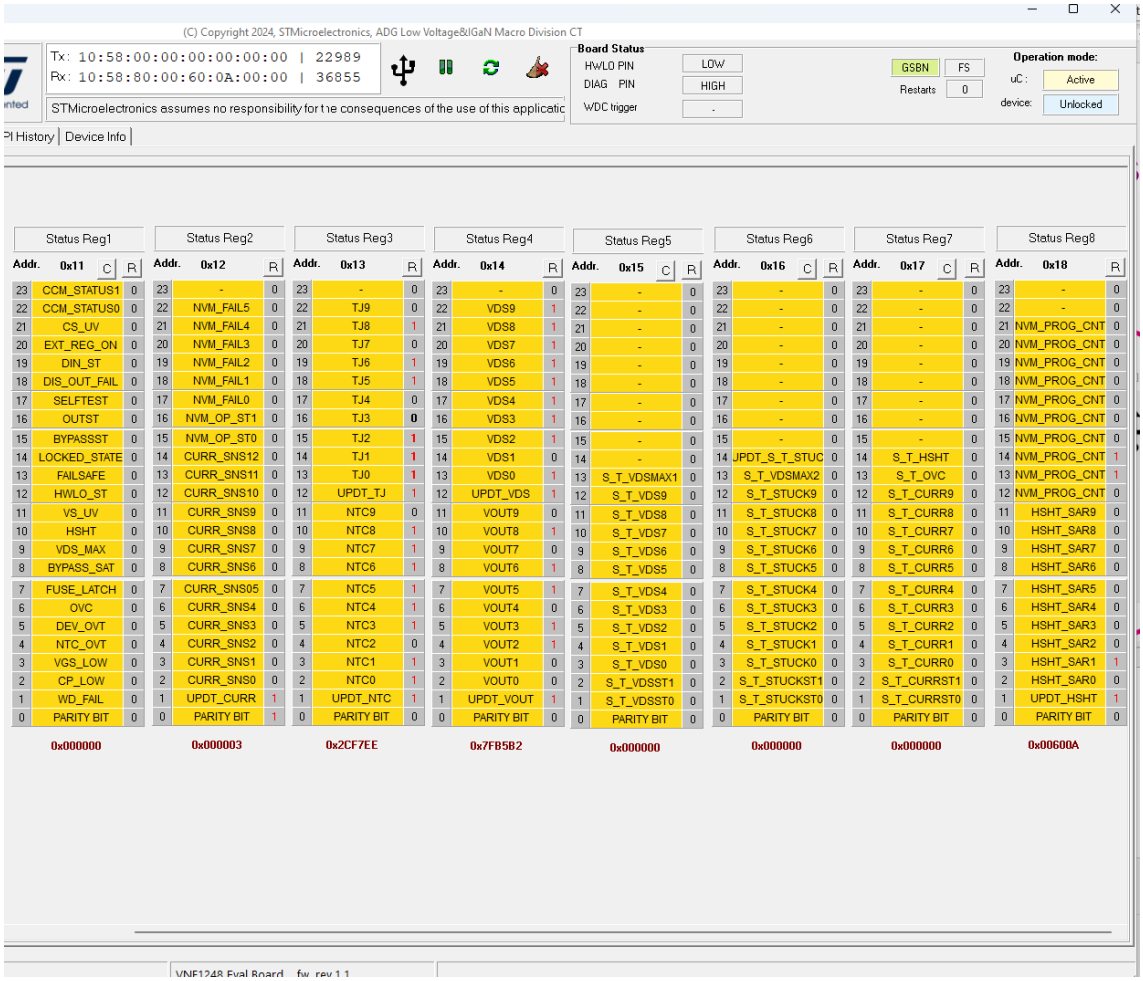


Figure 42. **SPI Status Registers**



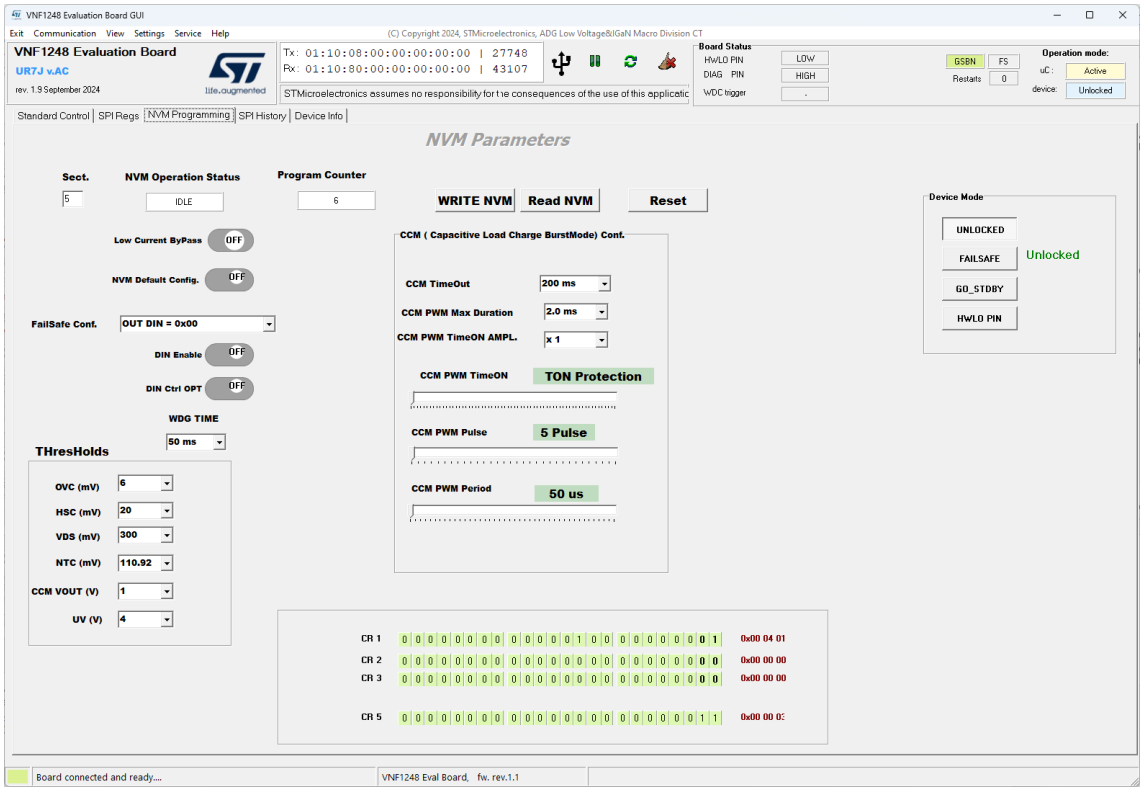
3.4.3 NVM Programming

The control logic unit available in VNF1248F enables the user to program a portion of the embedded NVM memory. This allows testing and storing specific settings for key parameters to be used during operation. Specifically, one of the six available sectors (sector 5) can be written to or read by the customer after entering the proper command sequence. This sequence ensures a secure and safe access to the functionality, avoiding unexpected write operations that could corrupt NVM content.

More details about the non-volatile memory programming feature can be found in the device datasheet (DS14109)

Below is the form provided in the Graphical User Interface to accomplish the NVM programming feature being reported.

Figure 43. NVM Programming



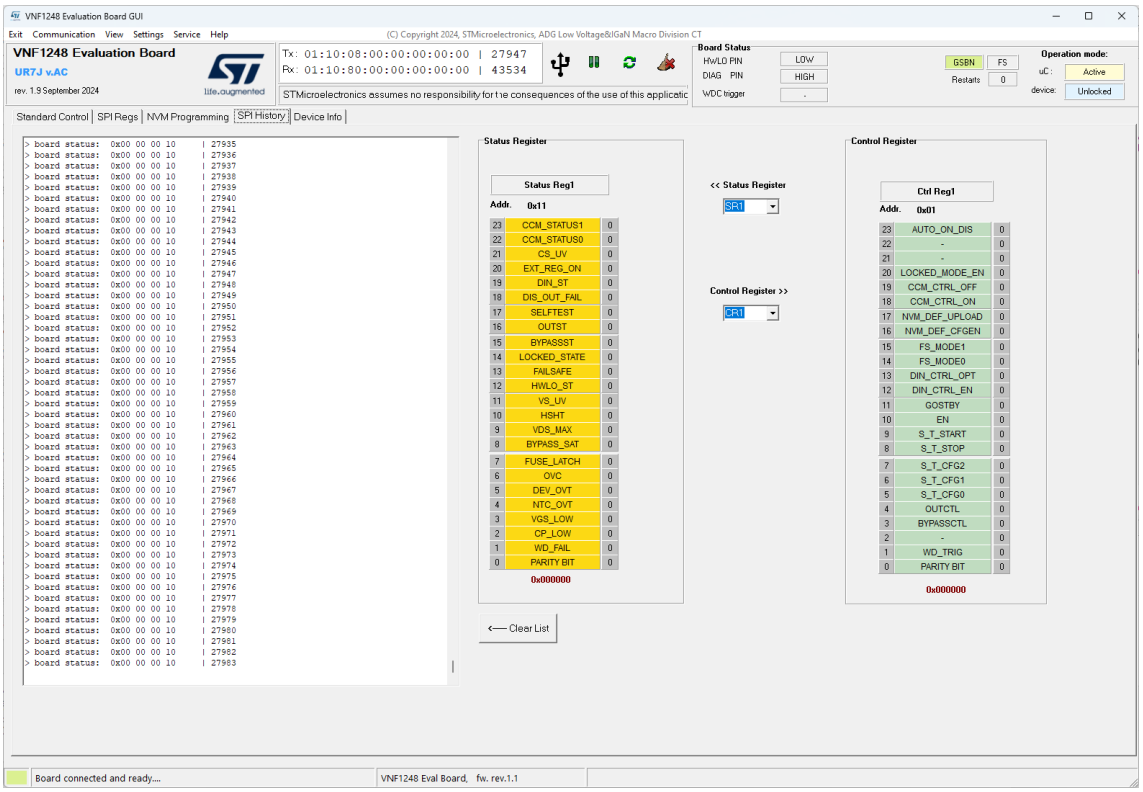
3.4.4 SPI History

This form shows the SPI frames of the communication traffic between the Graphical User Interface and the VNF1248F device.

The **Clear List** button allows to erase the whole communication history from the log window.

It's also possible to select a Control Register and a Status Register for an easy access.

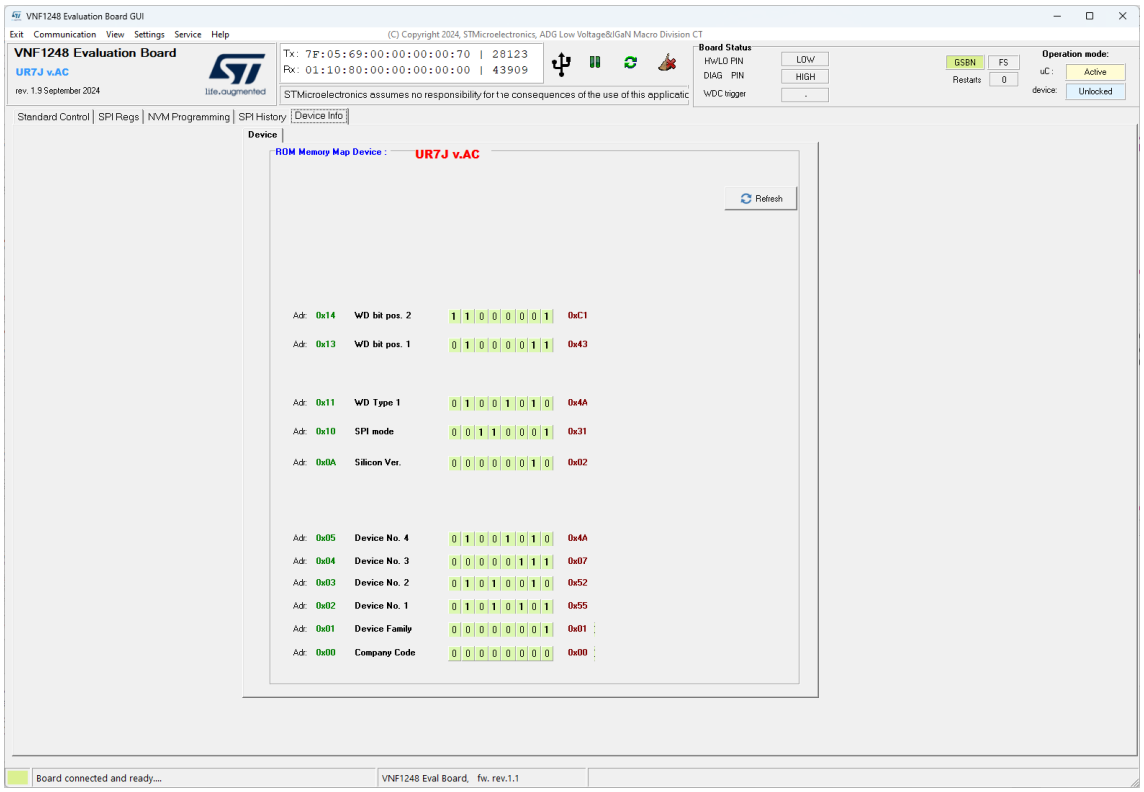
Figure 44. SPI History



3.4.5 Device Info

This form shows the ROM memory map of VNF1248F device. It can be refreshed manually with the dedicated button.

Figure 45. ROM Memory Map



Revision history

Table 1. **Document revision history**

Date	Revision	Changes
24-Jul-2025	1	Initial release.

Contents

List of figures

IMPORTANT NOTICE – READ CAREFULLY

STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice.

In the event of any conflict between the provisions of this document and the provisions of any contractual arrangement in force between the purchasers and ST, the provisions of such contractual arrangement shall prevail.

The purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST’s terms and conditions of sale in place at the time of order acknowledgment.

The purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of the purchasers’ products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

If the purchasers identify an ST product that meets their functional and performance requirements but that is not designated for the purchasers' market segment, the purchasers shall contact ST for more information.

ST and the ST logo are trademarks of ST. For additional information about ST trademarks, refer to www.st.com/trademarks. All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.