

# $\mu$ IO-Stick

## About this document

### Scope and purpose

The  $\mu$ IO-Stick is an interface device for controlling Infineon boards/kits during run time through PC.



### Intended audience

This document is intended for anyone using a  $\mu$ IO-Stick.

---

**Important notice****Important notice**

**“Evaluation Boards and Reference Boards” shall mean products embedded on a printed circuit board (PCB) for demonstration and/or evaluation purposes, which include, without limitation, demonstration, reference and evaluation boards, kits and design (collectively referred to as “Reference Board”).**

**Environmental conditions have been considered in the design of the Evaluation Boards and Reference Boards provided by Infineon Technologies. The design of the Evaluation Boards and Reference Boards has been tested by Infineon Technologies only as described in this document. The design is not qualified in terms of safety requirements, manufacturing and operation over the entire operating temperature range or lifetime.**

**The Evaluation Boards and Reference Boards provided by Infineon Technologies are subject to functional testing only under typical load conditions. Evaluation Boards and Reference Boards are not subject to the same procedures as regular products regarding returned material analysis (RMA), process change notification (PCN) and product discontinuation (PD).**

**Evaluation Boards and Reference Boards are not commercialized products, and are solely intended for evaluation and testing purposes. In particular, they shall not be used for reliability testing or production. The Evaluation Boards and Reference Boards may therefore not comply with CE or similar standards (including but not limited to the EMC Directive 2004/EC/108 and the EMC Act) and may not fulfill other requirements of the country in which they are operated by the customer. The customer shall ensure that all Evaluation Boards and Reference Boards will be handled in a way which is compliant with the relevant requirements and standards of the country in which they are operated.**

**The Evaluation Boards and Reference Boards as well as the information provided in this document are addressed only to qualified and skilled technical staff, for laboratory usage, and shall be used and managed according to the terms and conditions set forth in this document and in other related documentation supplied with the respective Evaluation Board or Reference Board.**

**It is the responsibility of the customer’s technical departments to evaluate the suitability of the Evaluation Boards and Reference Boards for the intended application, and to evaluate the completeness and correctness of the information provided in this document with respect to such application.**

**The customer is obliged to ensure that the use of the Evaluation Boards and Reference Boards does not cause any harm to persons or third party property.**

**The Evaluation Boards and Reference Boards and any information in this document is provided "as is" and Infineon Technologies disclaims any warranties, express or implied, including but not limited to warranties of non-infringement of third party rights and implied warranties of fitness for any purpose, or for merchantability.**

**Infineon Technologies shall not be responsible for any damages resulting from the use of the Evaluation Boards and Reference Boards and/or from any information provided in this document. The customer is obliged to defend, indemnify and hold Infineon Technologies harmless from and against any claims or damages arising out of or resulting from any use thereof.**

**Infineon Technologies reserves the right to modify this document and/or any information provided herein at any time without further notice.**

**Table of contents**

**Table of contents**

	<b>Table of contents</b> .....	3
<b>1</b>	<b>Overview</b> .....	4
1.1	Description .....	4
1.2	Key features .....	4
1.3	Block diagram .....	5
<b>2</b>	<b>Hardware connection</b> .....	6
2.1	Pin assignment .....	6
2.2	Examples .....	8
<b>3</b>	<b>μIO-Stick firmware update</b> .....	11
<b>4</b>	<b>FastLIN BSL support</b> .....	12
4.1	Installation guideline .....	12
4.2	BSL tool .....	12
4.2.1	The configuration dialog .....	12
4.2.2	Establish a connection .....	14
4.2.3	Loading a HEX File .....	15
4.2.4	Basic Mode .....	16
4.2.5	Expert Mode .....	17
4.2.5.1	Get and set NAC and NAD values .....	18
4.2.5.2	Read and write 100TP pages .....	19
4.2.6	BSL actions .....	20
4.2.6.1	Auto Execute .....	20
4.2.6.2	Erase .....	21
4.2.6.3	Download .....	21
4.2.6.4	Verify .....	22
4.2.6.5	Protection enable/disable .....	22
4.2.6.6	Run .....	24
4.2.6.7	Read Out .....	24
4.3	Command line tool .....	25
4.4	FAQ .....	26
4.5	Updates and purchases .....	27
<b>5</b>	<b>Run-time control - programming and monitoring</b> .....	28
5.1	Installation guideline .....	28
5.2	Config Wizard .....	31
<b>6</b>	<b>Acronyms</b> .....	33
	<b>Revision history</b> .....	34
	<b>Disclaimer</b> .....	35

---

**1 Overview****1 Overview****1.1 Description**

The μIO-Stick is an interface device for controlling Infineon boards/kits during run time through PC. It enables the connection between the evaluation board and USB for SPI programming and monitoring. Besides, it plugs into the evaluation board via a standard 16-pin connector and allows easy interface to the microcontroller via USB for communication through SPI, LIN. It also implements GPIO ports, PWM generation up to 20 kHz and an analog measurement interface (ADC).

The μIO-Stick provides two exclusive functions:

**1. Infineon MOTIX™ MCU FastLIN BSL support**

The Infineon MOTIX™ MCU devices provide a built-in bootstrap loader (BSL) for programming the MOTIX™ MCU devices over the built-in LIN transceiver. The μIO-Stick acts as an interface between the MOTIX™ MCU device and a PC. It handles the BSL protocols as well as the physical layers. A PC application provides access to all BSL functions of the MOTIX™ MCU device.

Supported MOTIX™ MCU ICs:

- TLE983x family
- TLE984x family
- TLE985x family
- TLE986x family
- TLE987x family

**2. Run-time control - programming and monitoring**

The μIO-Stick connects the Evaluation Board with the computer. Via the ConfigWizard software – a powerful and intuitive graphical user interface (GUI) – the user gets a customized programming, monitoring and evaluation support for various Infineon automotive products.

Supported devices:

- [System Basis Chip \(SBC\)](#)
- [DC Motor System ICs](#)
- [High/low Side Switch \(Spider+, SPOC™\)](#)
- [Half and Full Bridge Driver](#)

**1.2 Key features**

The μIO-Stick provides the following features:

- Communication with the connected device via LIN, SPI
- Four GPIOs
- PWM signal generation up to 20 kHz
- Measurement of analog voltages up to 20 V
- Read and keep functionality to detect reset of the connected device
- USB HID device, no additional driver installation required
- μIO-Stick can power a target device
- Connected device can also be powered by external power supply for higher current
- 3 on-board status LEDs
- Additional BSL updater GUI for BSL communication with MOTIX™ MCU devices

1 Overview

1.3 Block diagram

The μIO-Stick consists of an XMC4200 micro-controller which provides the necessary hardware interface and handles the USB and BSL protocols.

LIN and RS232 are implemented as true physical layers (using LIN and MAX transceivers), while the digital lines (SPI, GPIOs) provide a 5V-TTL level.

In addition, a small switchable charge-pump is implemented to generate a +12 V/200 mA supply for the target device and the integrated LIN transceiver.

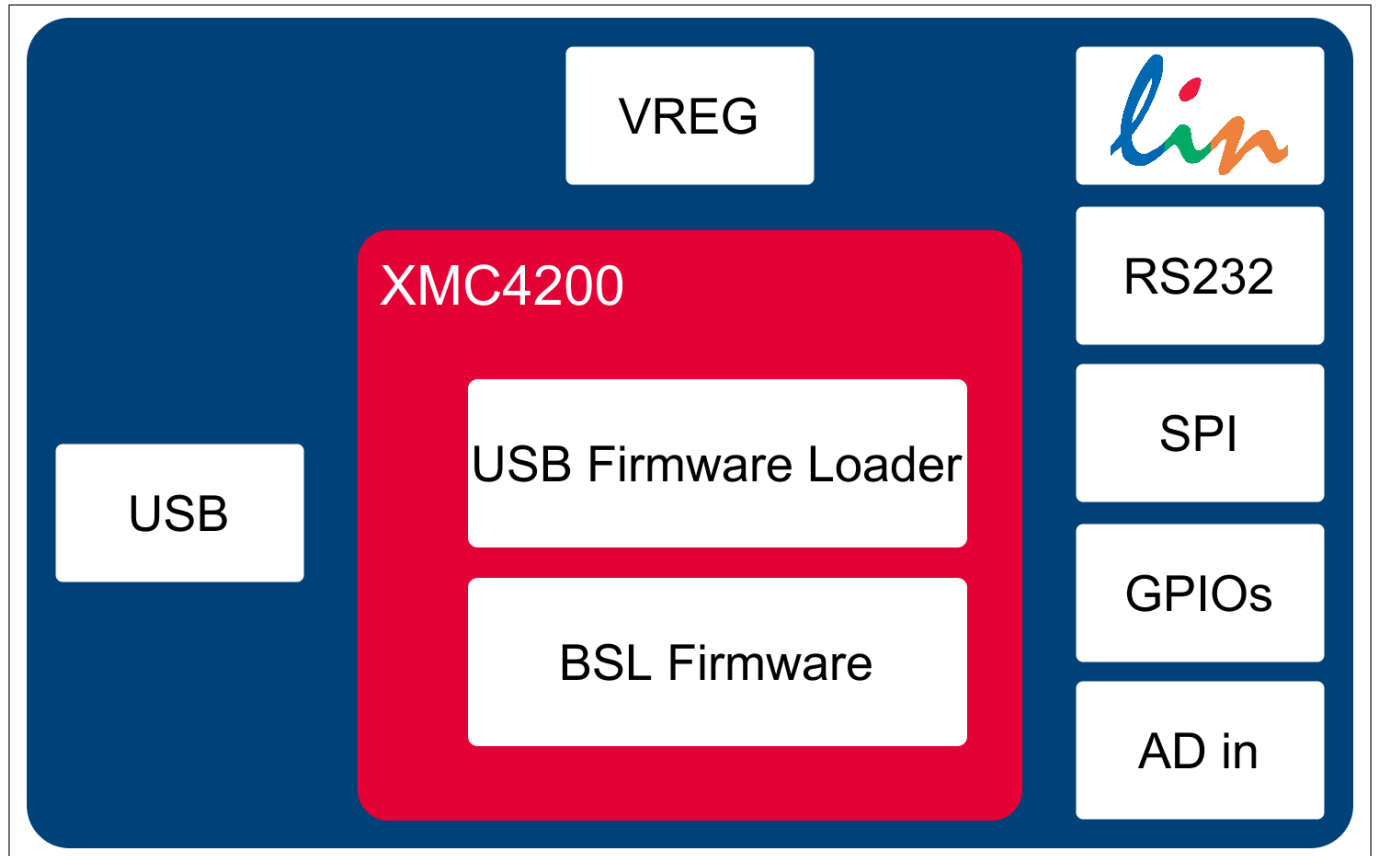


Figure 1 μIO-Stick block diagram

2 Hardware connection

2 Hardware connection

2.1 Pin assignment

The μIO-Stick provides two interfaces: The USB interface to be plugged into the PC and a 16-pin header to access the BSL interface. Figure 2 provides a view into the 16-pin header of the μIO-Stick, including the pin numbering.



Figure 2 μIO-Stick pin view

Table 1 lists the pin usage of the 16-pin header. The gray shaded pins are not used for BSL functions and therefore they are not considered further.

Table 1 μIO-Stick pin assignment

PIN	Description
1	RS232-TxD (out)
2	GND
3	RS232-RxD (in)
4	+5 V (USB, out)
5	LIN (physical, in/out)
6	VS (target supply, in/out)
7	Target Reset (5 V, out)
8	GPIO3 (5 V, in/out)

(table continues...)

**2 Hardware connection**

**Table 1** (continued) μIO-Stick pin assignment

<b>PIN</b>	<b>Description</b>
9	SPI/CS(5 V, out)
10	GPIO2 (5 V, in/out)
11	SPI CLK (5 V, out)
12	GPIO1 (5 V, in/out)
13	SPI MISO (5 V, in)
14	GPIO0 (5 V, in/out)
15	SPI MOSI (5 V, out)
16	analog in (max. 18 V)

*Note:* The RS232 pins are no TTL pins but real RS232 levels (MAX232 driven).

**Pin 5 - LIN** This is the LIN bus connection to the LIN transceiver implemented in the μIO-Stick. It is suited to drive a corresponding LIN input of the target system.

**Pin 7 - Target Reset** This pin is intended to drive the reset input of the target device in order to synchronize the device state with the attempts to establish a BSL connection. The connection of this pin to the reset of the target device is not necessarily required as synchronization can also be achieved by:

- Power cycling of the target device by pin 6 (VS)
- Keeping the target device in BSL mode by configuring an appropriate NAC value

**Pin 2 - GND** This is the common ground connection to the target system.

**Pin 6 - VS** This pin can either be driven from the implemented charge-pump of the μIO-Stick (+12 V/200 mA) or can be overridden by an external supply, for example the supply of the target device. This pin is also used to internally drive the LIN transceiver available on the μIO-Stick. If BSL communication over the LIN interface is intended, then the supply of pin 6 is mandatory.

## 2 Hardware connection

### 2.2 Examples

Figure 3 shows the minimum required connection from the μIO-Stick to the target device in order to establish a BSL communication over the LIN interface. This applies to both Normal-LIN and FastLIN protocols. To be able to establish a BSL connection, the target device either has to stay in BSL mode upon power-up (fresh device), or a power cycling (VS - pin 6) done by the μIO-Stick is required.

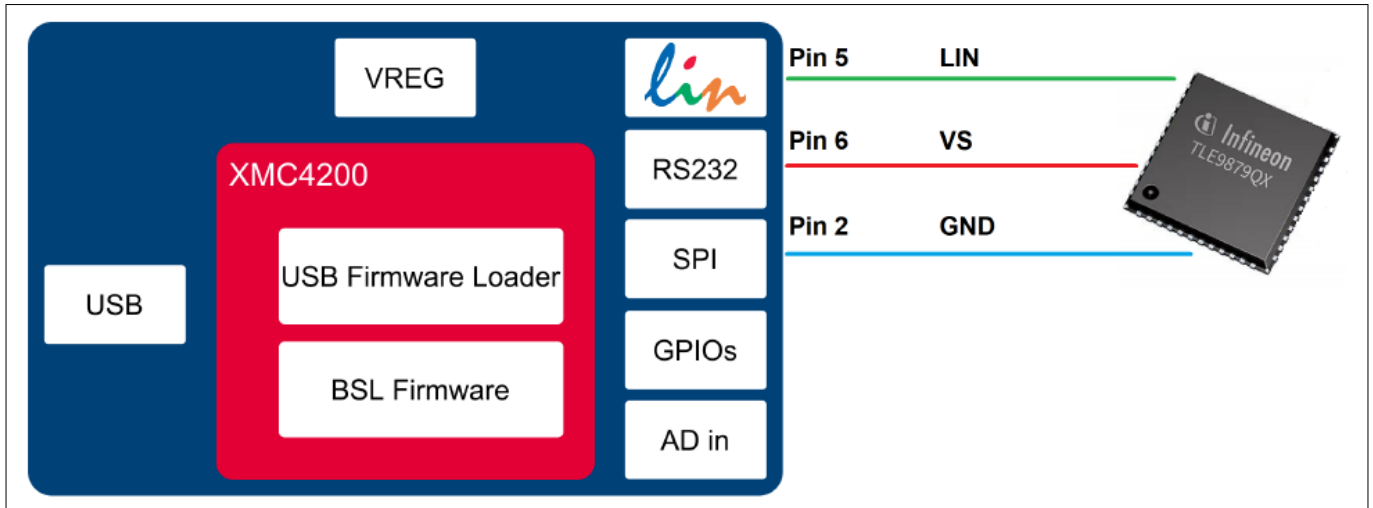


Figure 3 Normal LIN and FastLIN connection, without reset, target supplied by μIO-Stick

Figure 4 shows a connection scenario with reset. The μIO-Stick actively resets the target device in order to start BSL communication. In this case a power cycling of VS (pin 6) by the μIO-Stick is not required.

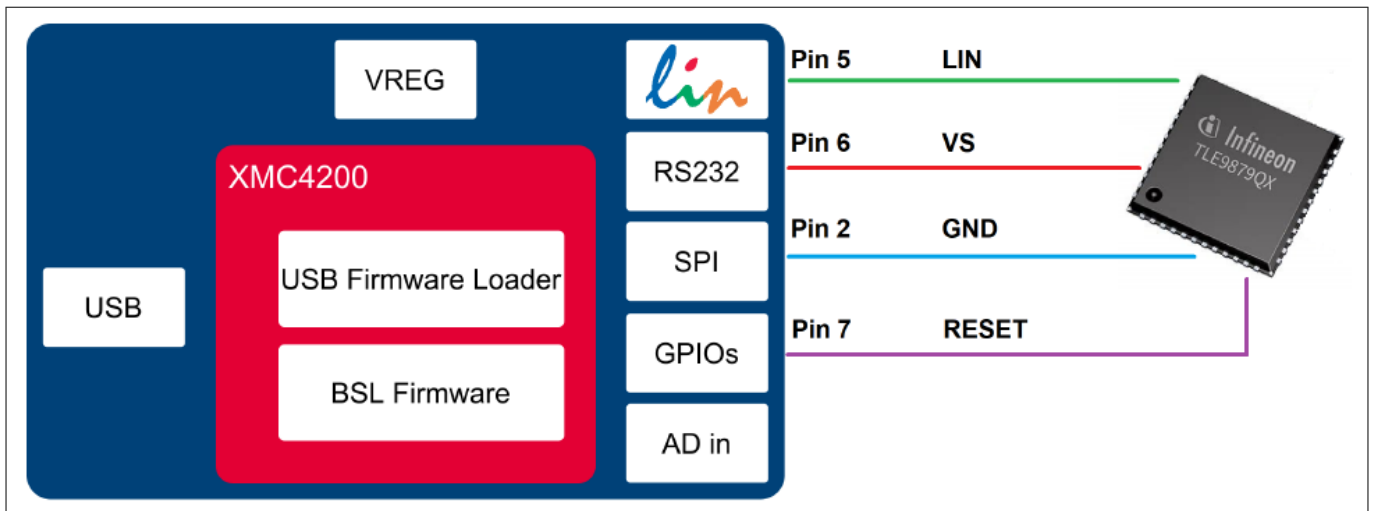
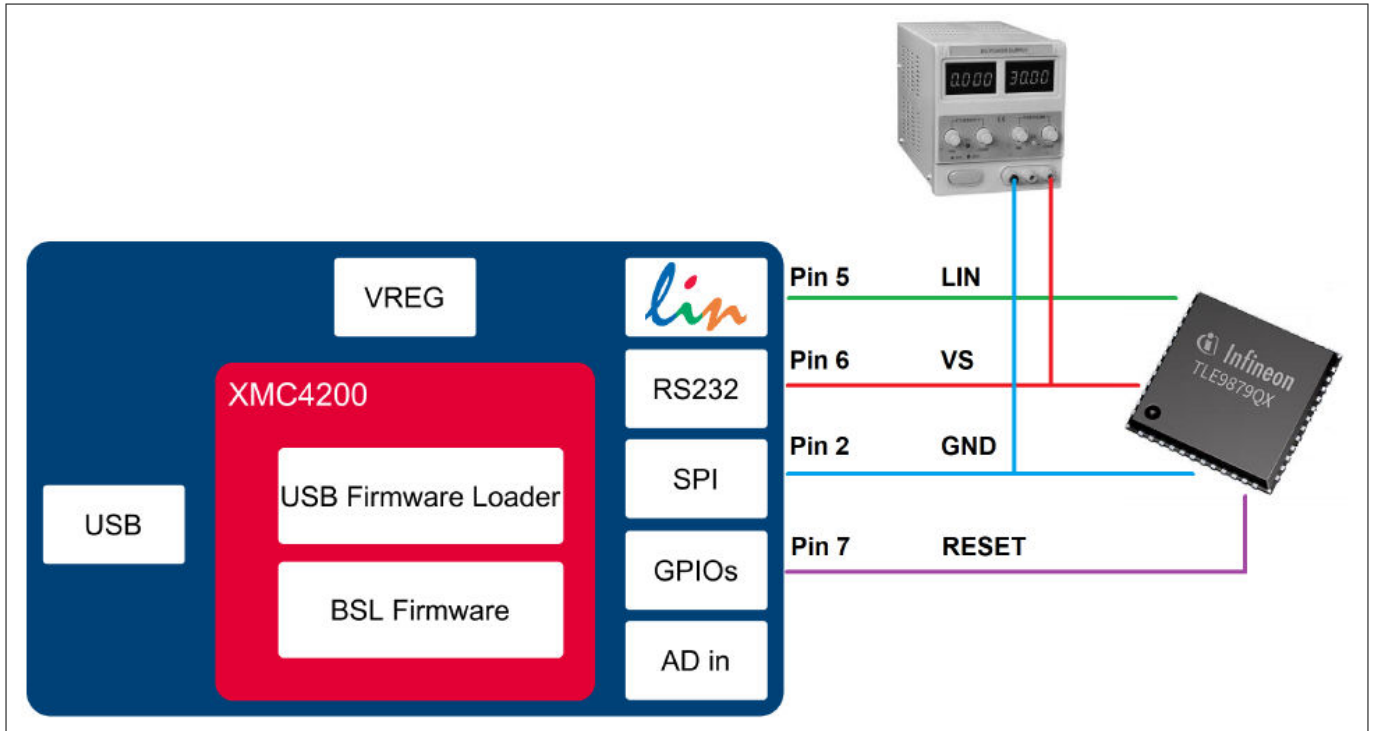


Figure 4 Normal LIN and FastLIN connection, with reset, target supplied by μIO-Stick



## 2 Hardware connection

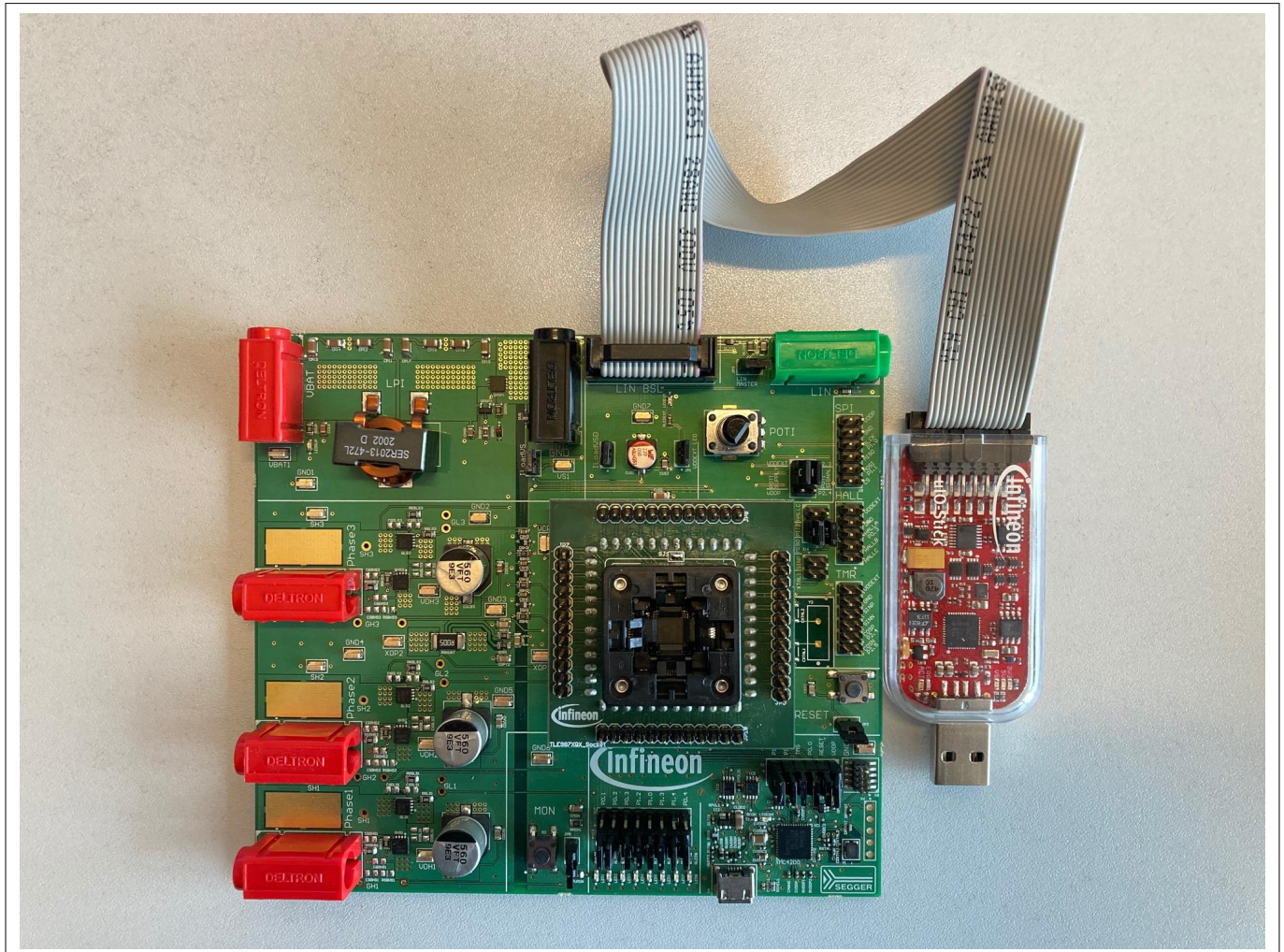
Figure 5 shows a connection example where the target device and also the μIO-Stick are supplied by a separate power supply. Since in this case the μIO-Stick would not be able to perform a VS power cycling to restart the device it is wise to connect the reset line as well. If the target device stays in BSL mode upon power-up, such as for a new, unprogrammed device, then the reset line is not required.



**Figure 5** Normal LIN and FastLIN connection, with reset, target supplied by separate power supply

## 2 Hardware connection

Figure 6 shows another example of how to connect the μIO-Stick to the target device, by simply plugging the ribbon cable to the BSL connector of the evaluation board. Hereby only the LIN interface is supported.



**Figure 6** Direct connection of μIO-Stick to evaluation board

---

## 3 μIO-Stick firmware update

### 3 μIO-Stick firmware update

The μIO Stick comes with the capability to update the firmware of the integrated XMC4200 micro-controller either to provide new features to the customer or to be able to correct issues with the firmware.

To update the firmware, the file **uIO-Updater.exe** and a HEX file are required. uIO-Updater.exe is a command line tool, where the hex file to be loaded has to be given as command line input. For example:

```
C:\>uIO-Updater.exe -FW="bsl_1.1.346.fw"
```

The example above flashes the file BSL\_V1.1.346.hex into the μIO-Stick. After the firmware update has finished, unplug the μIO-Stick from USB and reconnect it.

4 FastLIN BSL support

4 FastLIN BSL support

There are two software components required to use the BSL feature of the target device. Part 1 is the firmware of the XMC4200 microcontroller mounted on the μIO-Stick. This firmware is responsible to generate the BSL communication protocol. Part 2 is the BSL user software running on a Microsoft Windows PC. It provides the user interface and prepares the data packets to be sent to the μIO-Stick and further to the target device.

4.1 Installation guideline

1. Go to the section [Tools & Software](#) of the μIO-Stick
2. Download **uIO\_BSL\_Tool.zip** and extract all the files
3. Connect the μIO-Stick via USB to the PC and via the 16-pin connector to the target device
4. Update the μIO-Stick with the current BSL firmware version by executing the script **UpdateBSL**
5. Launch the application **BSL\_Tool.exe**

4.2 BSL tool

The BSL GUI tool is the user interface to control the BSL interactively. It provides two modes of operation, a Basic Mode and an Expert Mode.

4.2.1 The configuration dialog

Before a BSL connection is established, the target device and the BSL interface and protocol have to be selected. The **Configuration** block lists the currently selected target device, while the status bar at the bottom of the window displays the selected BSL protocol and baud rate. In order to change these settings, press the **Change** button inside the configuration block. Please also refer to [Figure 7](#).

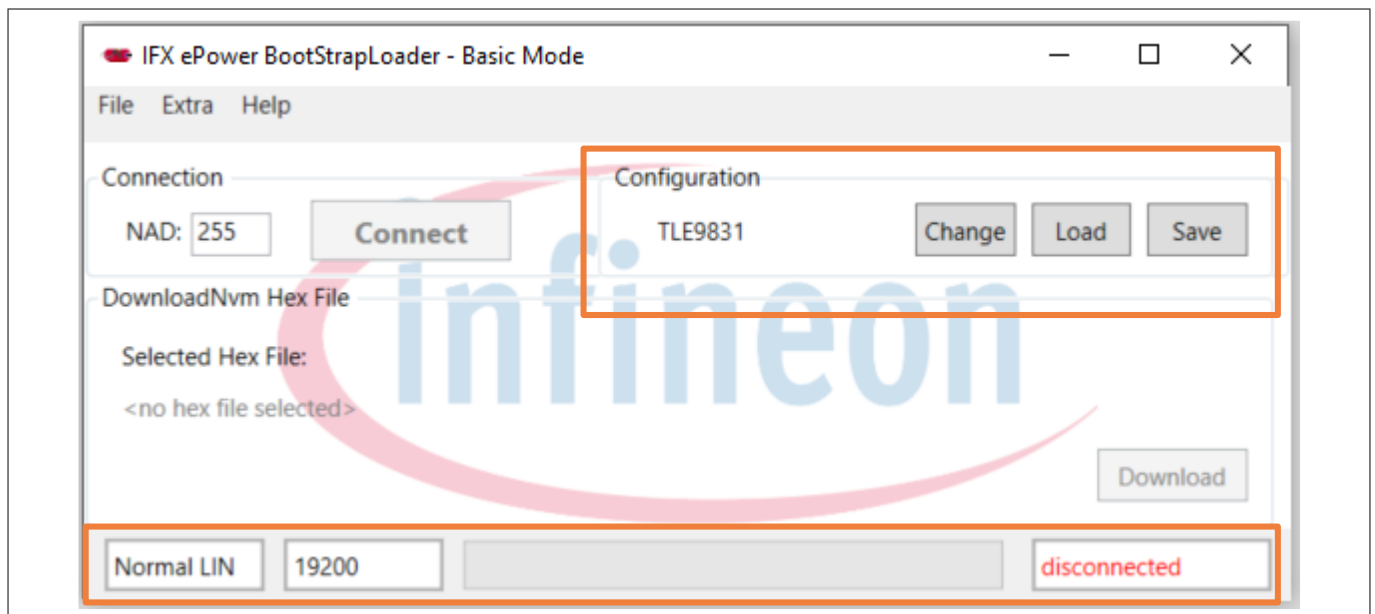


Figure 7 BSL tool - configuration

Figure 8 displays the **BSL Configuration** dialog. Here the user selects the desired target device, BSL protocol, baud rate, and the behavior of the integrated VS charge pump.

4 FastLIN BSL support

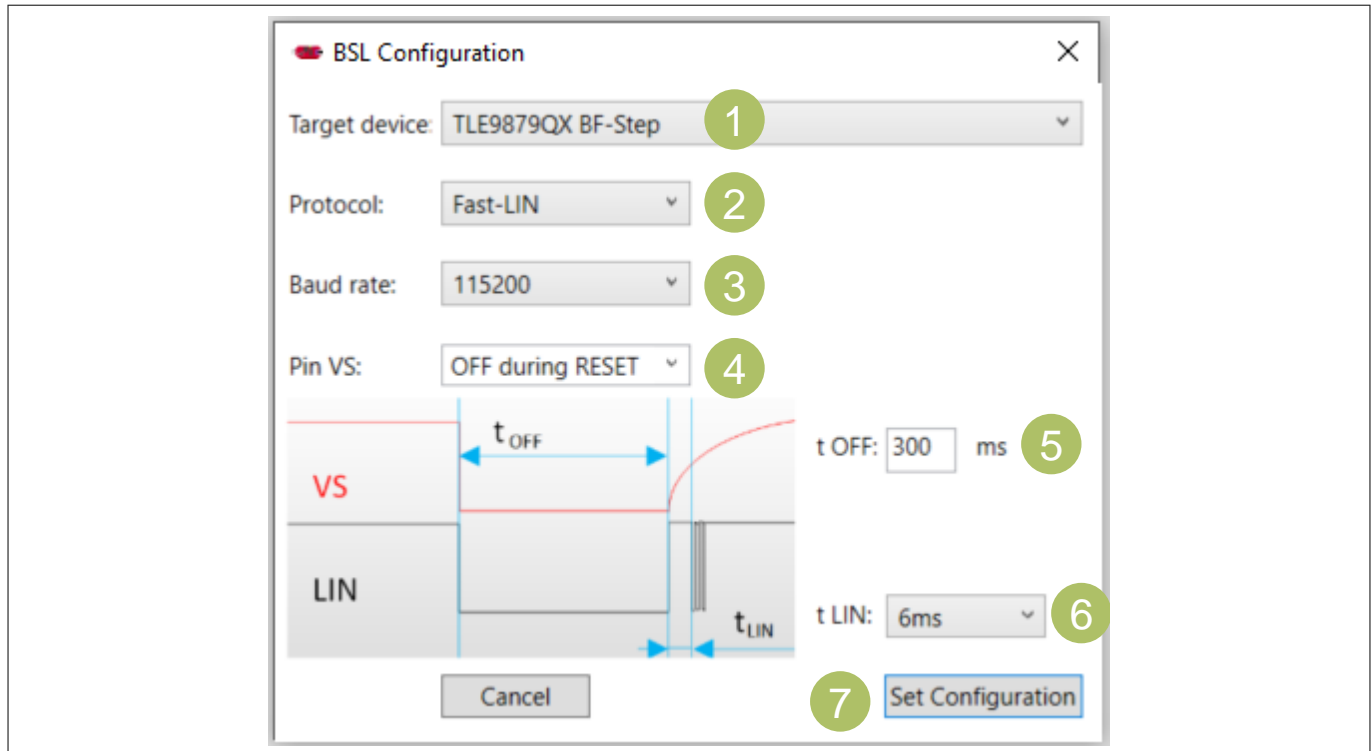


Figure 8 BSL tool - Configuration dialog

1. **Target Device** - select the target device from the pull-down list
2. **Protocol** - select the desired BSL protocol and BSL interface:
  - Normal LIN - BSL communication uses real LIN frames, Master-Request frames, and Slave-Response frames. In order to distinguish these BSL LIN frames from real LIN frames, the checksum is inverted. Communication is done via the LIN transceiver that is integrated in the target device.
  - Fast-LIN - UART like BSL communication using the device integrated LIN transceiver. This is the default protocol for new, unprogrammed devices
3. **Baud Rate** - select the communication baud rate
4. **Pin VS** - select the behavior of the VS pin:
  - OFF - the μIO-Stick integrated charge pump is switched off, VS must be supplied by the target system, see also [Figure 5](#) on page 9.
  - ON - the μIO-Stick integrated charge pump is switched on, and remains on. Please see also [Figure 4](#).
  - OFF during Reset - the μIO-Stick integrated charge pump is switched off during the reset phase before the connection attempt is started, see also [Figure 3](#). This setting is meant to provide a power-on reset and can be used in case the RESET pin of the target device is not accessible.
5. **tOFF** - defines the duration of the RESET pulse in ms. If **Pin VS** is set to OFF during RESET this setting is applied to the reset pulse of both the RESET pin and VS pin. If **Pin VS** is set to OFF or ON, this setting is only applied to the reset pulse of the RESET pin.
6. **tLIN** - defines the delay between the release of the reset pulse and the first connection sequence
7. **Set Configuration** - click to apply the selected setting and close the BSL Configuration dialog.

Once a configuration is selected, it can be saved as an XML file by clicking the **Save** button. A previously saved configuration XML file can be loaded by clicking the **Load** button. These functions can also be selected in the menu **File**, see [Figure 9](#).



#### 4 FastLIN BSL support

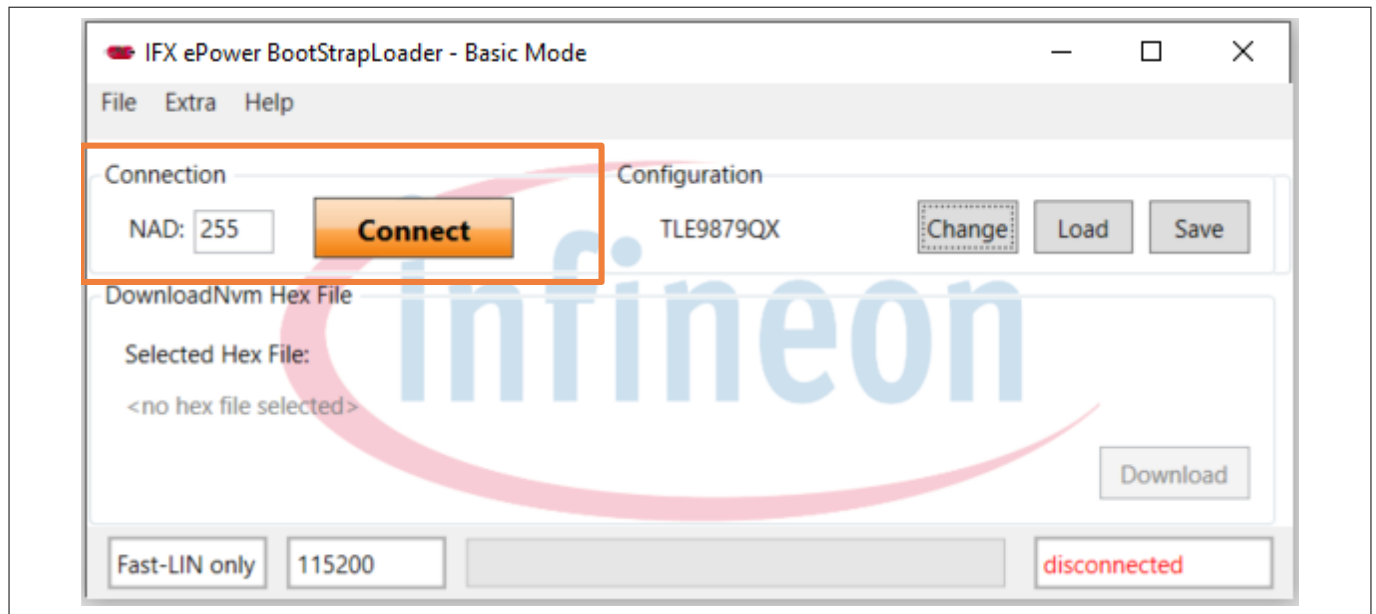


Figure 9 BSL tool - configuration menu

#### 4.2.2 Establish a connection

Once the configuration has been completed, the connection to the target device can be established. If the μIO-Stick is not connected to the PC system, the **Connect** button is shaded gray, as shown in Figure 10.

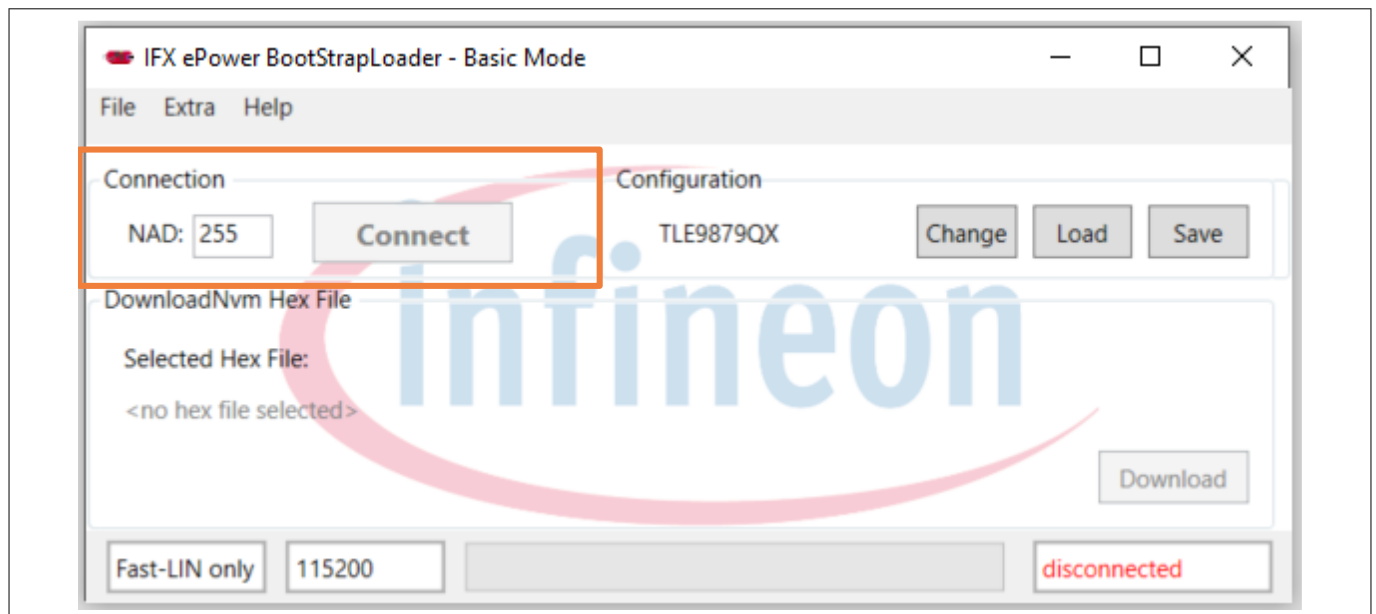


Figure 10 BSL tool - no μIO-Stick found

Once the μIO-Stick is recognized by the BSL tool the **Connect** button turns orange. By pressing the button, the BSL tool tries to establish a connection using the selected BSL interface and BSL protocol. The NAD value used for connection is set to 255 by default. If required, it can be changed.

#### 4 FastLIN BSL support

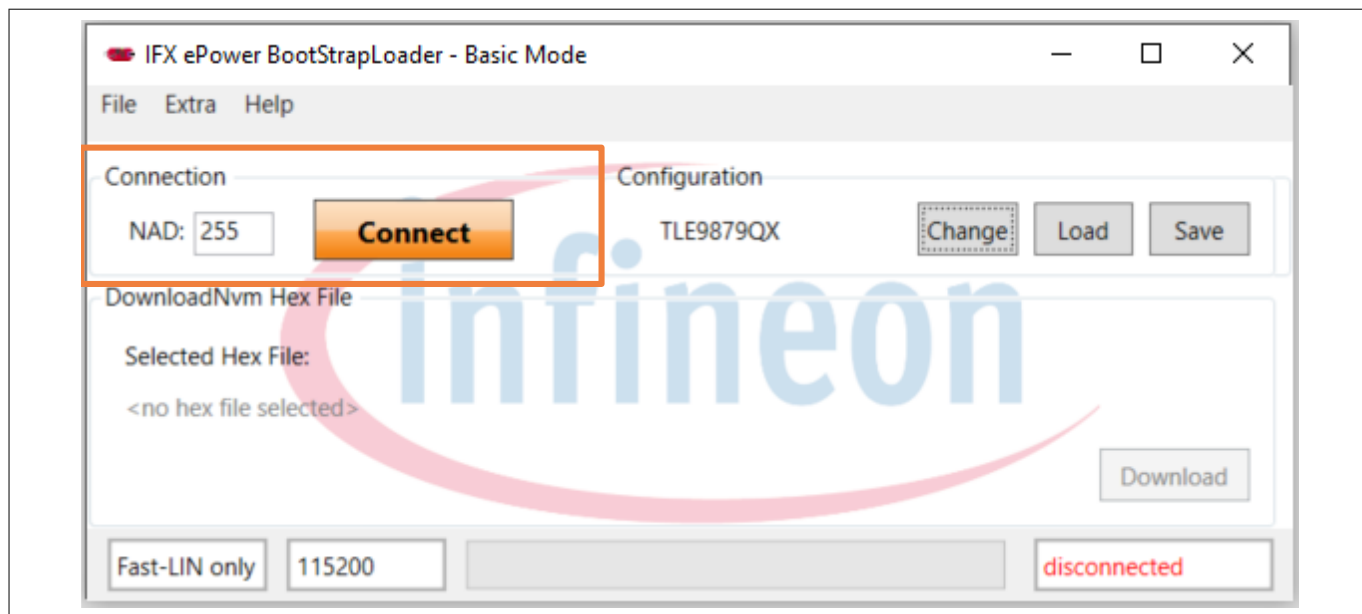


Figure 11 BSL tool - ready to connect to target system

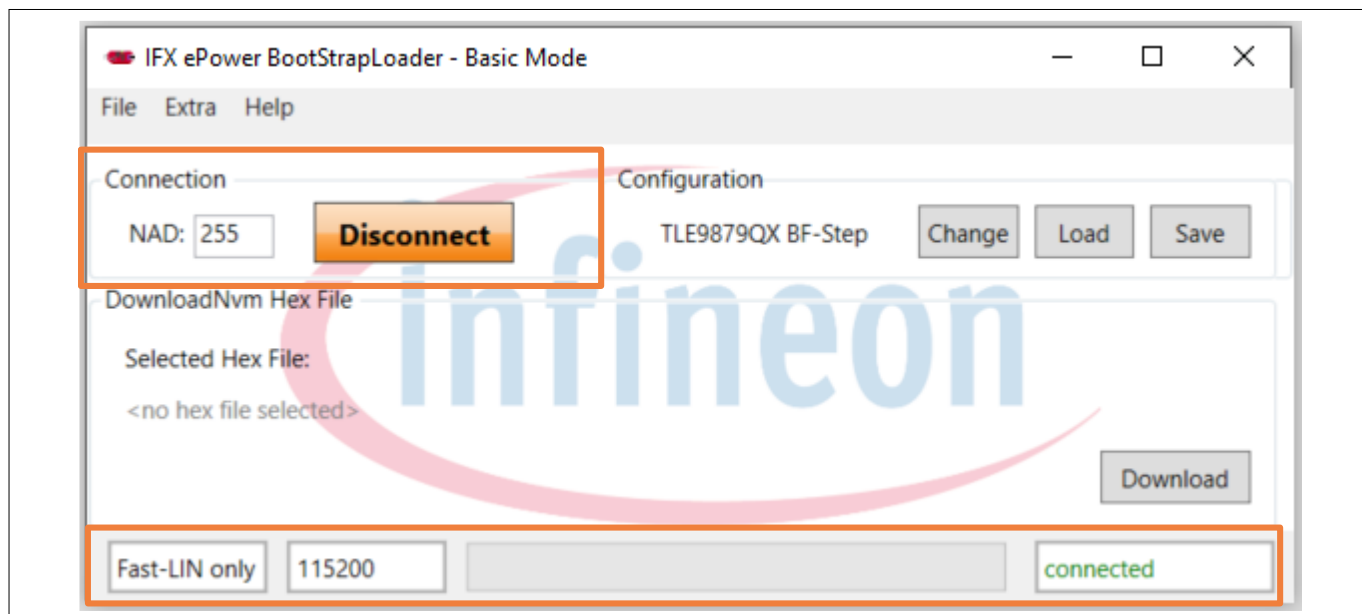


Figure 12 BSL tool - Connected

Figure 12 displays the connected state. The **Connect** button turns into a **Disconnect** button. The status bar at the bottom of the window shows connected.

#### 4.2.3 Loading a HEX File

The main use of the BSL tool is downloading a HEX file that contains the user application code. Via the menu **File** a HEX file can be chosen and loaded, see Figure 13. This step can be done before or after connecting to the target device because it does not program the HEX file to the target device yet.

4 FastLIN BSL support

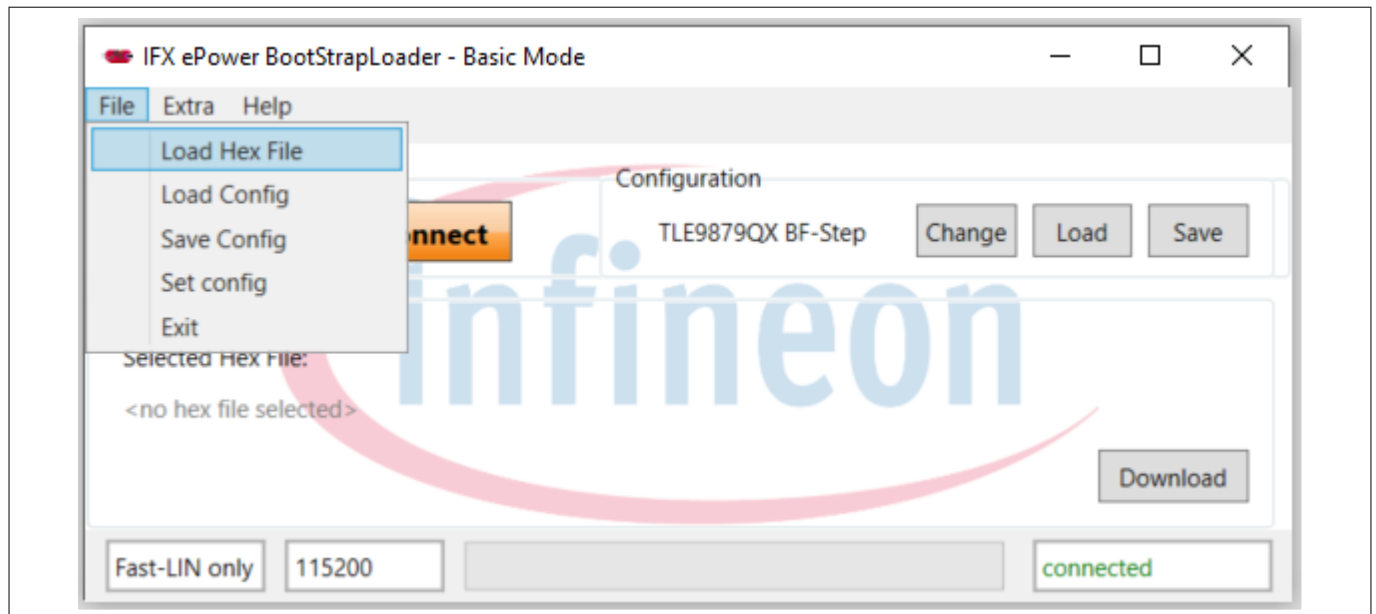


Figure 13 BSL tool - load a HEX file

4.2.4 Basic Mode

The Basic Mode provides a reduced user interface limited to the minimum required actions, which are **Erase**, **Download**, **Verify**, and **Run of code** on the target device. After a connection was established and a HEX file was loaded, all mentioned actions are executed consecutively by clicking the button **Download**. The status bar on the bottom shows the currently running action. After successful download and run, the BSL tool disconnects automatically.

Figure 14 shows the user interface of the Basic Mode during download.

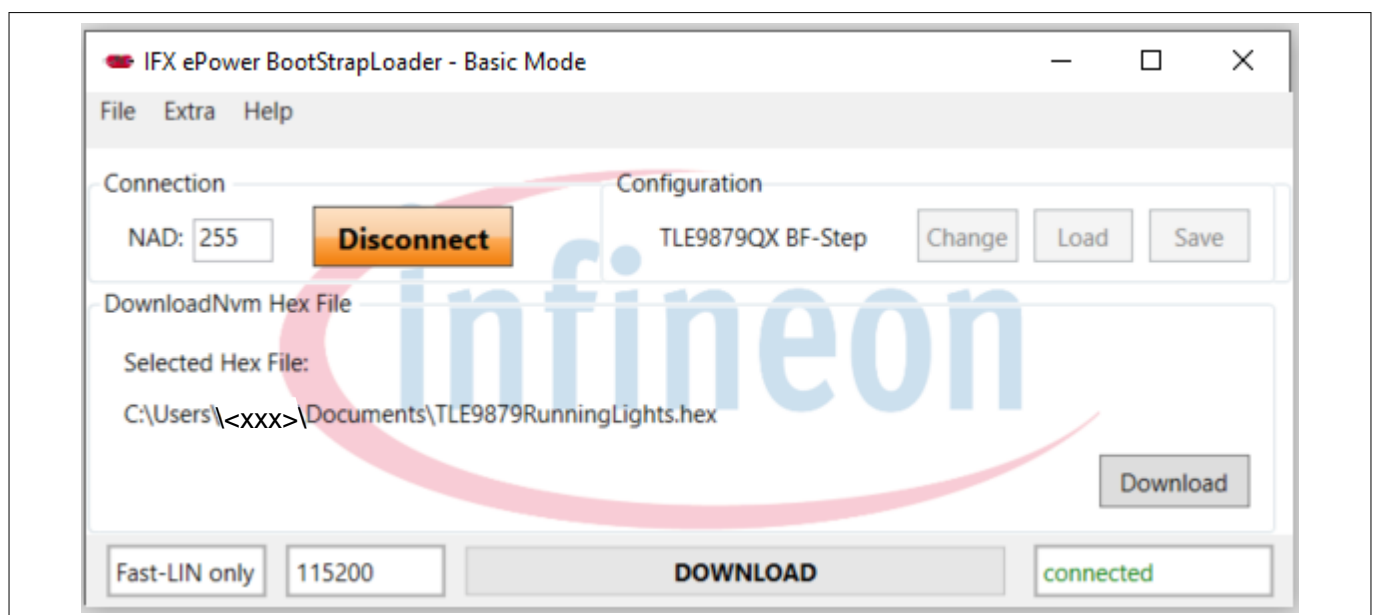


Figure 14 BSL tool - Basic Mode



4 FastLIN BSL support

4.2.5 Expert Mode

The Expert Mode provides more functionality, all functions of the BSL modes supported by the target device are available in this mode. In order to switch to Expert Mode from Basic Mode, select **Expert Mode** in menu **Extra**, as shown in [Figure 15](#).

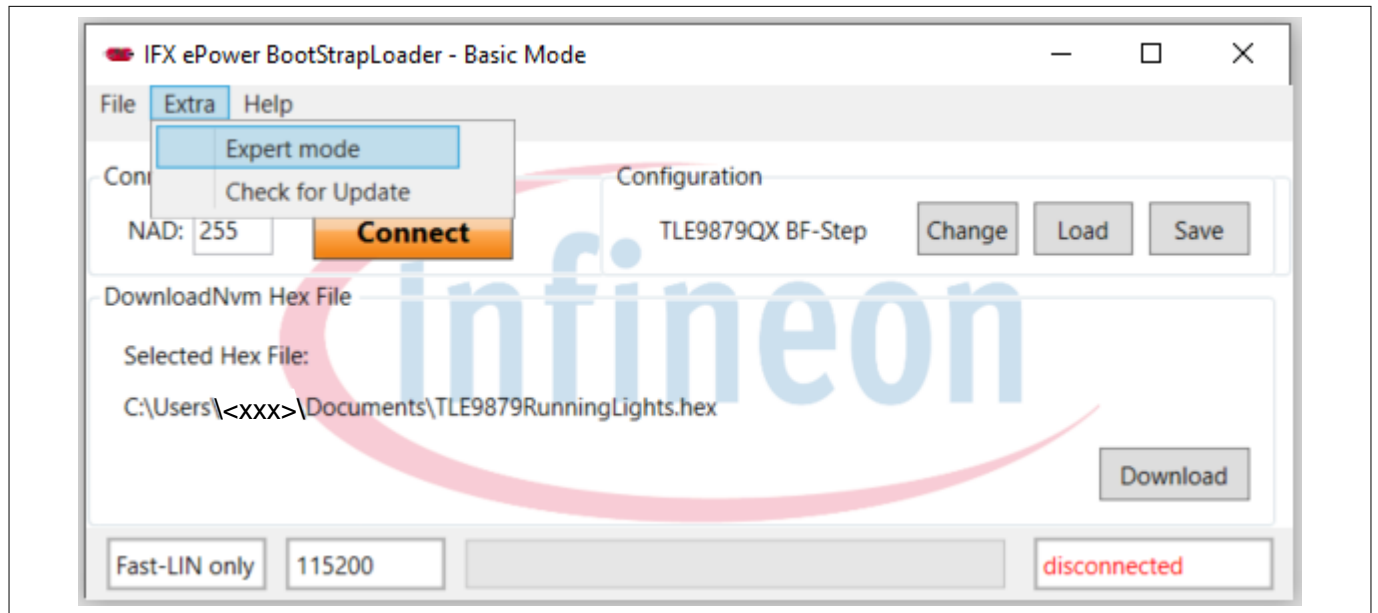


Figure 15 BSL tool - switch to Expert Mode

[Figure 16](#) displays the user interface of the Expert Mode. The Expert Mode enables all possible BSL actions, including target device erase, code download, device protection, code execution, and code upload. The arrangement of these functions in the GUI is kept in the natural order of execution of these actions, such as the erase function at the top, followed by download, verify, and so on.

## 4 FastLIN BSL support

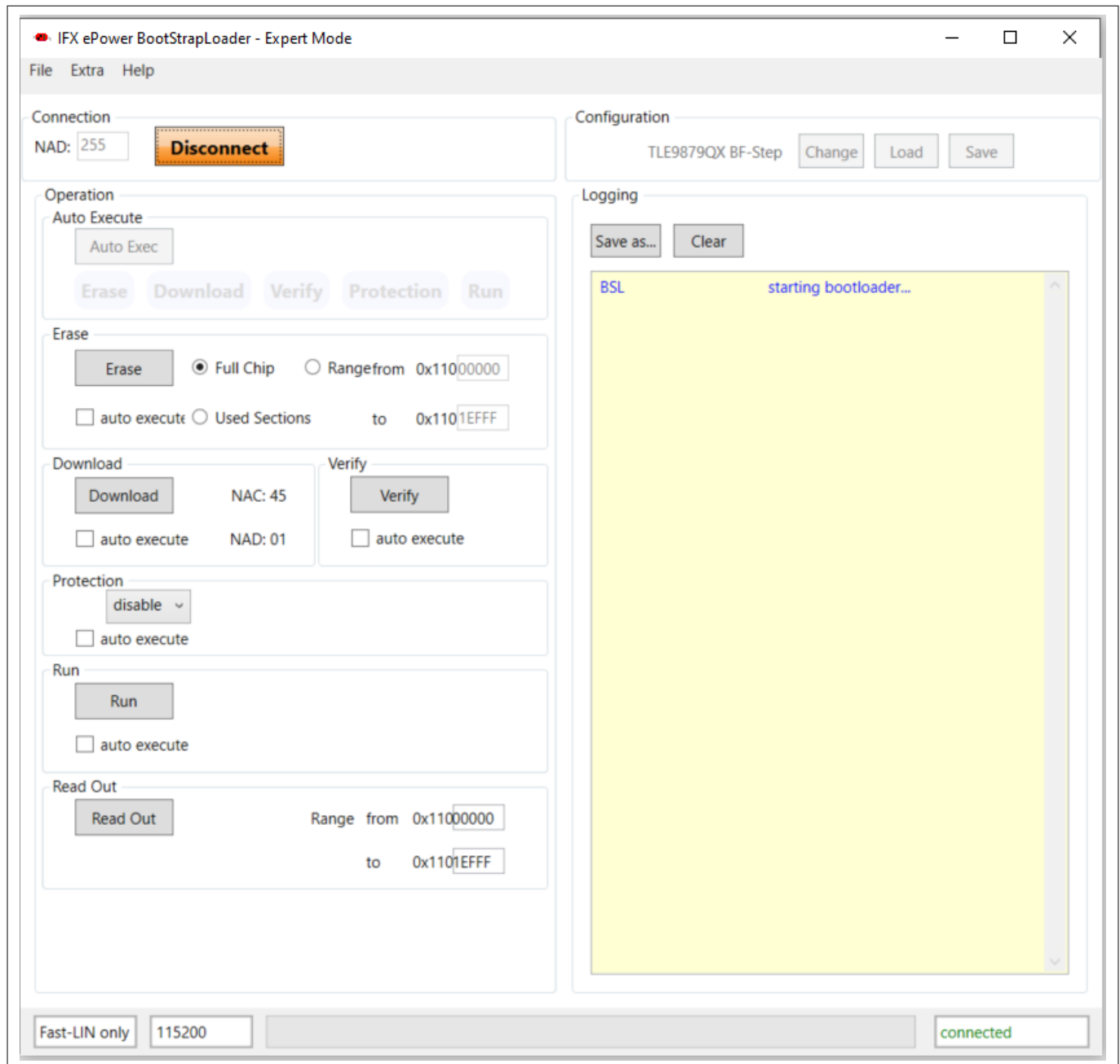
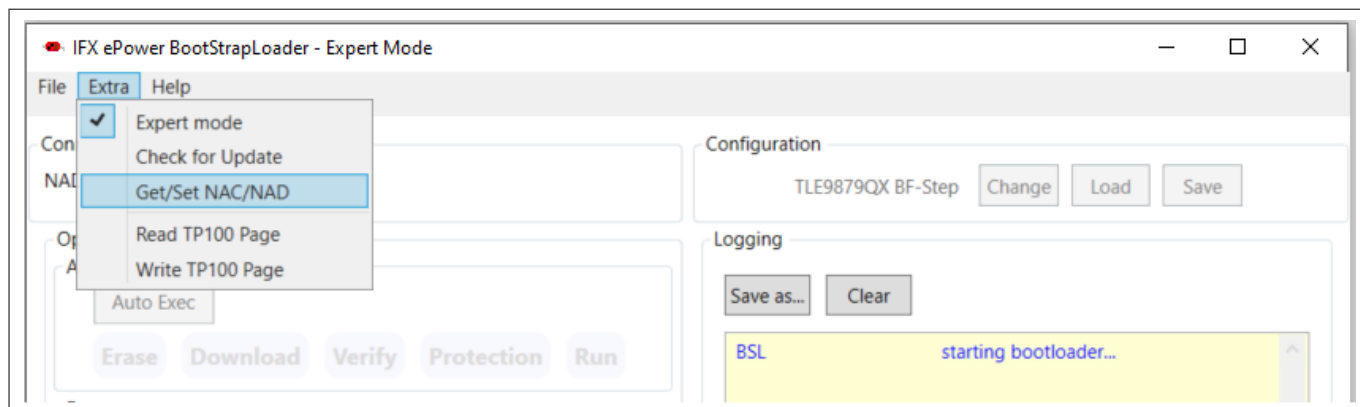


Figure 16 BSL tool - Expert Mode

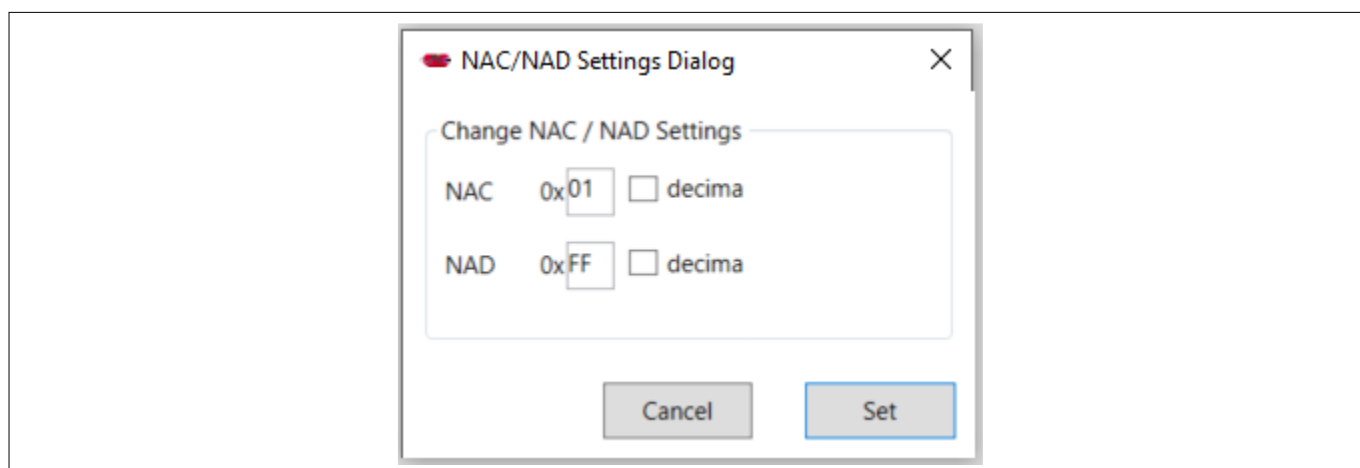
### 4.2.5.1 Get and set NAC and NAD values

In Expert Mode, the BSL tool offers the option to change the NAC and NAD values. In the menu **Extra, Get/Set NAC/NAD** can be selected as shown in the figure below. This opens the **Change NAC/NAD Settings** dialog which shows the NAC and NAD value of the currently connected device (see [Figure 18](#)). The values can be changed and set with the **Set** button.

## 4 FastLIN BSL support



**Figure 17** BSL tool - get and set NAC/NAD value



**Figure 18** BSL tool - change NAC/NAD Settings Dialog

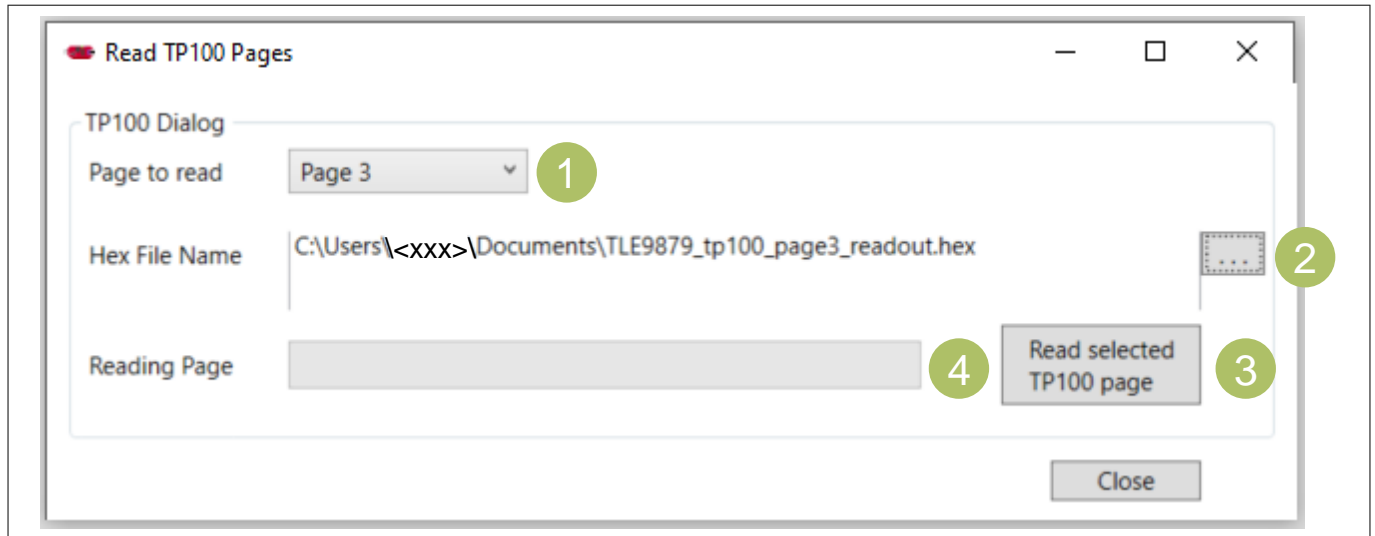
### 4.2.5.2 Read and write 100TP pages

In Expert Mode, the BSL tool enables the user to read or write 100 times programmable pages of the target device. Via the menu **Extra**, the read or write 100TP operation can be selected.

In the **Read TP100 Pages** dialog

1. Select the page to be read via the drop-down menu
2. Select a location and file name for the resulting HEX file via the ... button
3. Click **Read selected TP100 page**
4. The process bar shows the status of the reading page operation

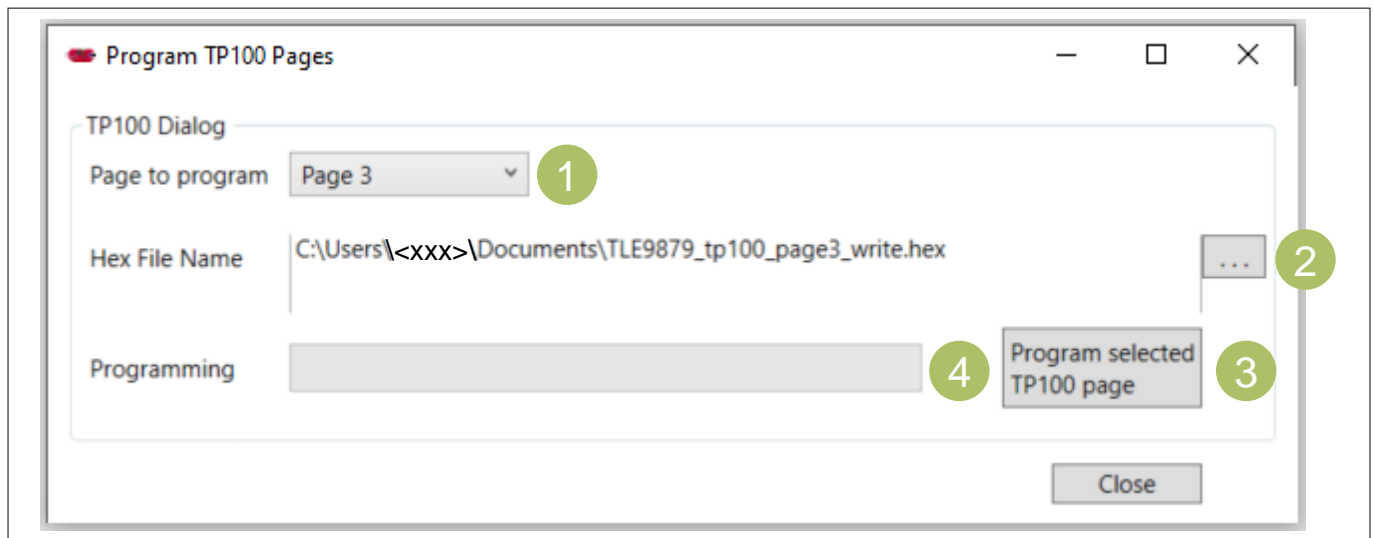
4 FastLIN BSL support



**Figure 19** BSL tool - Read TP100 Pages dialog

In the **Program TP100 Pages** dialog

1. Select the page to be programmed via the drop-down menu
2. Select the HEX file that is to be written to the 100TP page via the ... button
3. Click **Program selected TP100 page**
4. The process bar shows the status of the programming operation



**Figure 20** BSL tool - Program TP100 Pages dialog

**4.2.6 BSL actions**

Once the desired Hex file has been selected, the supported BSL actions can be executed. The order of the actions placed on the GUI follows a natural flow: Erase → Download → Verify → Protection → Run.

**4.2.6.1 Auto Execute**

Each of these action groups provides an **auto execute** check box. If it is checked, the corresponding action is added to the **Auto Execute** list. Figure 21 shows an example of the Auto Execute list, where everything except "Protection" is added to the list. By pressing the **Auto Exec** button, the actions added to the list are executed with a single click.

4 FastLIN BSL support

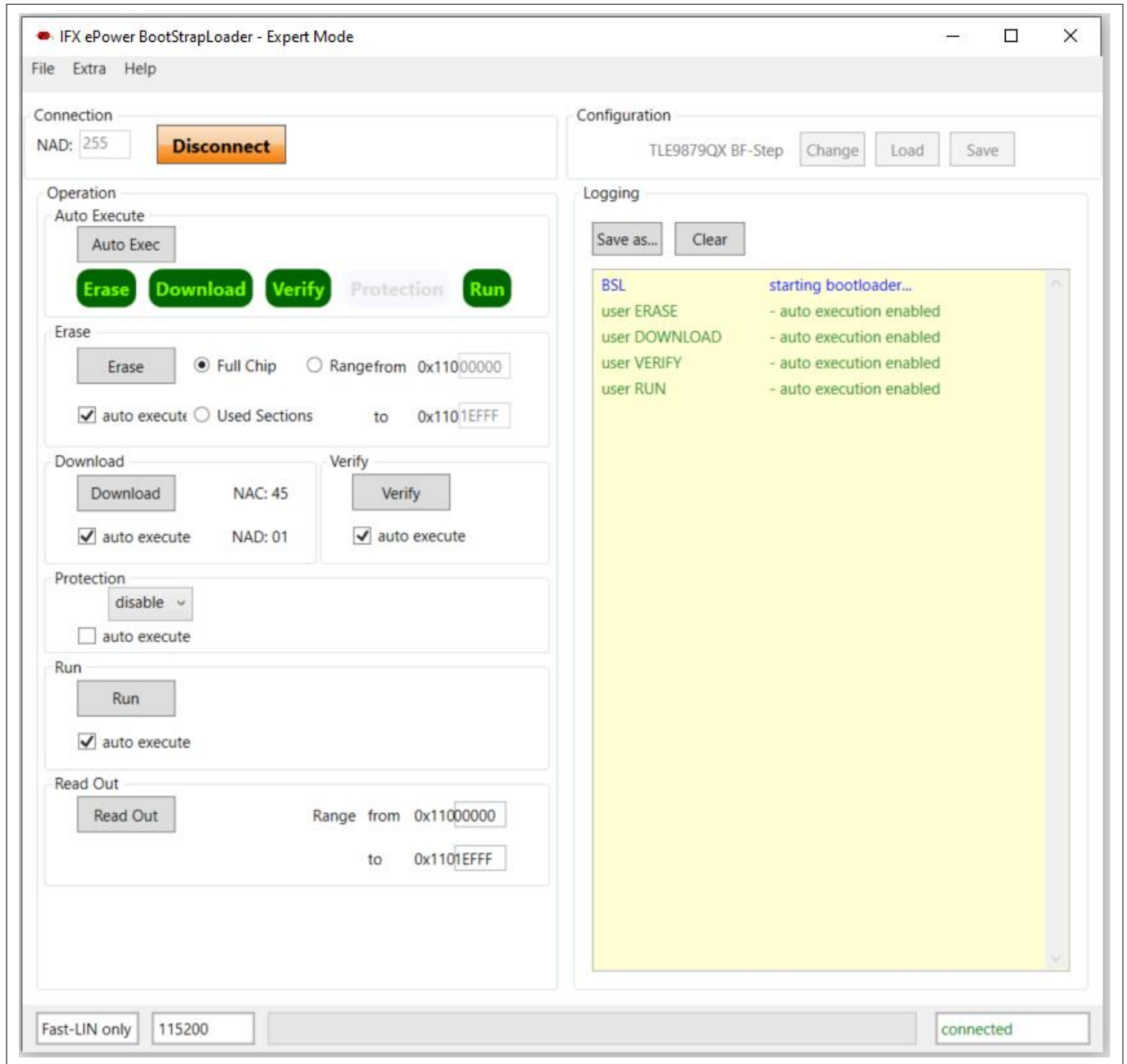


Figure 21 BSL tool - Auto Execute

4.2.6.2 Erase

The action **Erase** performs an erase of the target device built-in NVM module. The user can select between various sections of the NVM to be erased, which are:

- Full Chip - the entire user accessible NVM is erased
- Range from ... to ... - only the user defined range is erased
- Used Sections - only those NVM regions occupied by the selected HEX file are erased

4.2.6.3 Download

The action **Download** transfers a loaded HEX file into the target device.

4 FastLIN BSL support

4.2.6.4 Verify

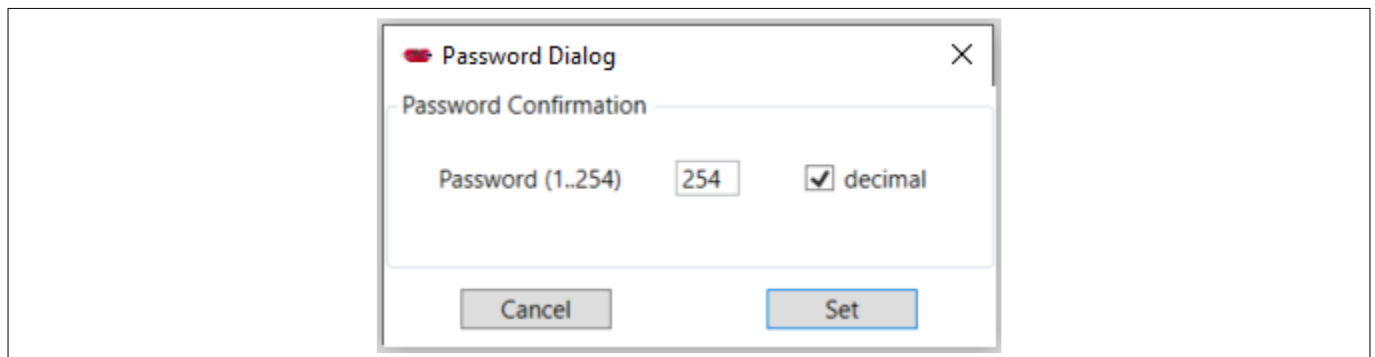
The action **Verify** is used to check the integrity of the code inside the target device NVM against the selected HEX file.

4.2.6.5 Protection enable/disable

The action **Protection** either enables or disables the protection of the target device. Depending on the selected target device, the GUI is adapted to the respective protection concept.

Protection of TLE986x and TLE987x devices

For enabling or disabling the protection, the protection action needs to be added to the Auto Execute operation. When the auto execute tick box is ticked, a password dialog opens up. The user has to enter an 8-bit password, as shown in [Figure 22](#)

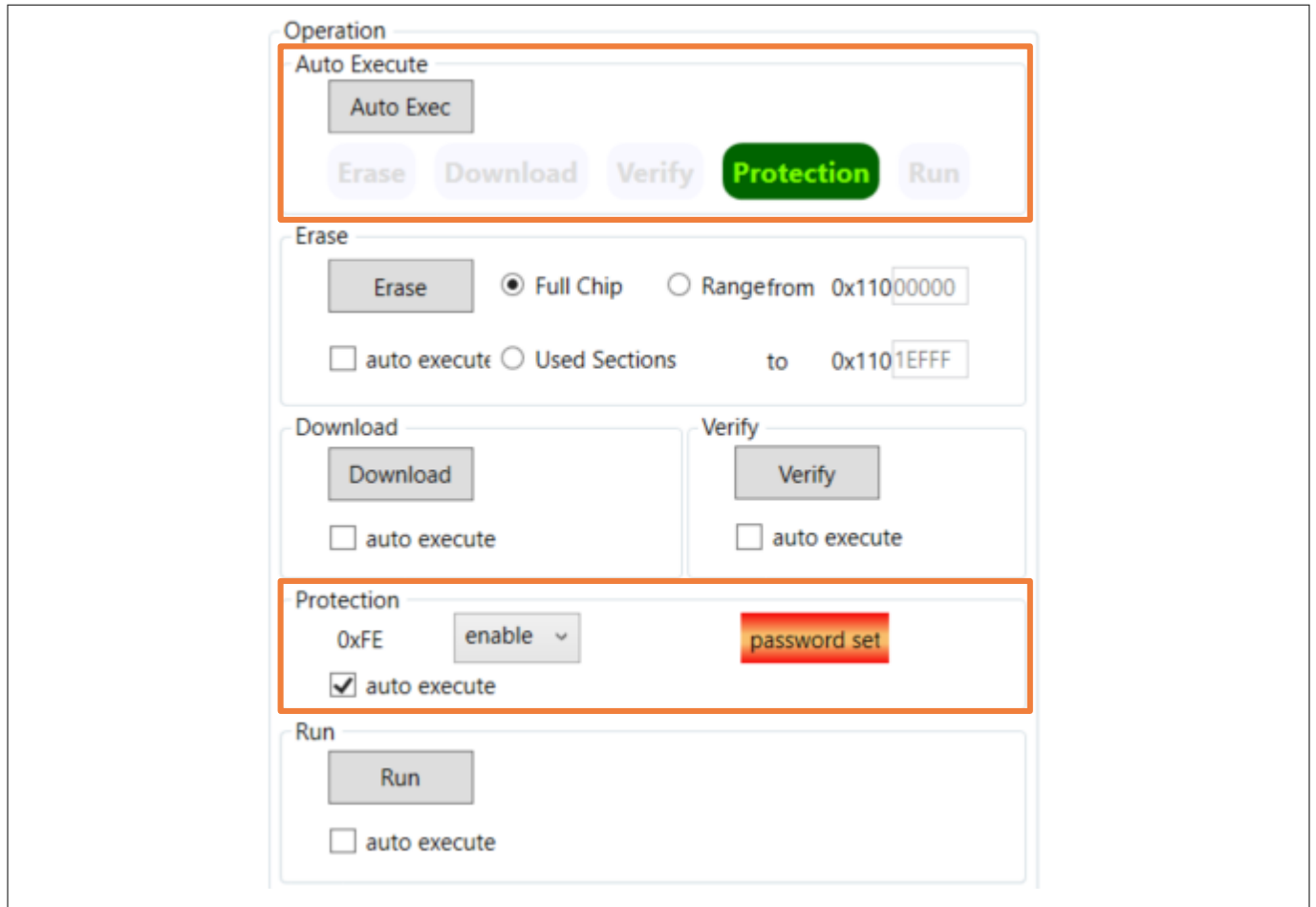


**Figure 22** BSL tool - Password Dialog for TLE986x and TLE987x devices

If a password is set, this is stated in red inside the **Protection** field, as shown in [Figure 23](#).

*Note: Before each enable or disable action, the auto execute tick box needs to be unticked and ticked again in order to open the Password Dialog and enter the password.*

4 FastLIN BSL support



**Figure 23 BSL tool - password set and ready to be sent for TLE986x and TLE987x devices**

If the target device is not protected, more actions can be added to the auto execute operation, so that enabling the protection is part of the download flow. If the target device is protected, the (disable) protection action can also be executed separately by clicking the **Auto Exec** button.

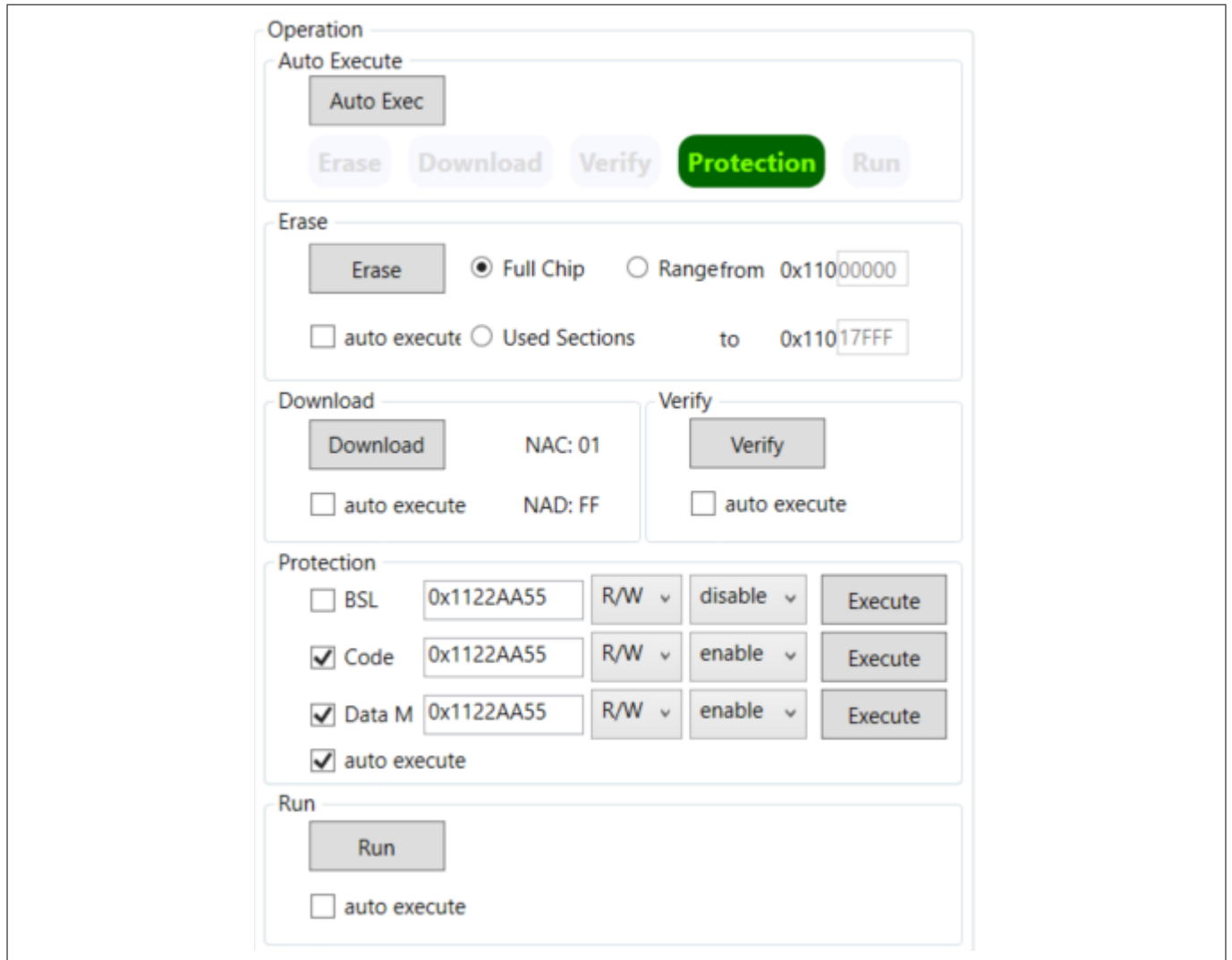
If the protection is set, the target device does no longer accept any NVM change operations, nor allows code readout. In addition, the SWD interface (ARM) or the DAP interface (8051) is disabled.

To reset the protection status, the same matching 8-bit password used to set the protection has to be provided in the password dialog. If the password matches, the entire NVM of the target device is erased. The target device is fully accessible again.

Protection of TLE984x and TLE985x devices

For TLE984x and TLE985x devices the protection can be set specifically for BSL, code and data. For each protection a separate password can be set in the corresponding field. To execute the protection, either click on each **Execute** button or add required protection actions to the auto execute operation by ticking the boxes for required protection in addition to the tick box for auto execute. [Figure 24](#) shows an example of how to add enabling read/write protection for code and data to Auto Execute.

4 FastLIN BSL support



**Figure 24** BSL tool - enable read/write protection for TLE984x and TLE985x devices

*Note:* A device reset is required for the protection to take effect. To reset the device, disconnect and (re-)connect the BSL tool.

**4.2.6.6 Run**

The action **Run** starts the execution of the code inside the target device. It terminates the BSL connection..

**4.2.6.7 Read Out**

The action **Read Out** is available for the protocol FastLIN. It is not available for Normal LIN. It reads and transfers the target device NVM content to the PC system. The user has to define the address range to be read.



## 4 FastLIN BSL support

### 4.3 Command line tool

In order to run the BSL actions none-interactively, a command line BSL tool is provided. It is named **BSL\_Cmd.exe**, and should be run from a terminal window. Executing BSL\_Cmd.exe without any parameters generates a list of the supported options, as shown below:

```

C:\>BSL_Cmd.exe
-----
Bootstrap Loader Command Line Tool v1.2 build 1818
-----
ERROR          : missing arguments
Valid commands are:

-Cmode=baud,NAD      connection mode and baud rate
-EF                  erase full chip
-ER=start-end        erase range
-D="hex_file"        download hex file (repeatable)
-VS=n                function of VS pin. 0: off, 1:on, 2:off during reset
-V=1                 verify after download
-PS=password         set protection
-PR=password         reset protection
-R100TP=page,"hex_file" read the selected 100TP page and save it
-RO="hex_file",start-end read out range into hex
-R=NVM               run in NVM
-R=RAM               run in RAM
-T=device            target device
-NACNAD=nac,nad,bslMode set NAC, NAD and BSL mode (F:fastLIN, L:LIN)
-W100TP=page,"hex_file" write hex file to selected 100TP page
-Q                  quite mode
-tOFF=n              defines the duration of the RESET pulse [m]
-tLIN=n              defines the delay between the release of the RESET pulse and the
                    1st LIN command. Rounded to next valid value (2,6,11,17,75,90 ms)

mode                : N/F/FO/U  N=Normal LIN, F=Fast-LIN, FO=Fast-LIN-only, U=UART
baud                 : baud rate 9600..115200
hex_file             : defines the hex file for the download
page                 : selects 100TP page (1..8)
start                : defines the start address for the operation
end                  : defines the end address for the operation
device               : defines the target device (s. TargetConfiguration.xml)
-----

```

The code above shows a call of BSL\_Cmd.exe with various options, which are:

- -CFO=115200,1 : Fast-LIN-only, 115200 baud, NAD = 1
- -T=TLE9879QX : target device TLE9879QX, it has to match the device select list in the configuration dialog
- -VS=1 : VS (Pin 6) on all the time
- -EF : Erase full chip
- -D="Blinky.hex" : downloads the Blinky.hex file to the target device
- -PS=0x55 : sets protection with the 8-bit password 0x55

---

**4 FastLIN BSL support****4.4 FAQ**

Why does the connection fail?

- Check if the NAD value is correct or change to NAD=255
- Check the NAC value. If the NAC value is very small, the BSL tool might not be able to establish a connection
- For better performance of the μIO-Stick hardware, use a powered USB hub to ensure stable voltage at VS
- If power-on reset with VS is used, large capacitors at the VS pin might prevent a proper reset. Adjust reset timings to ensure a proper reset and adjust the timing for the beginning of the connection sequence, so that the NAC window is not missed

Why is it not possible to read a 100TP page?

- Check whether the device is (read) protected. If so, disable the protection before reading a 100TP page

Why is it not possible to program a 100TP page?

- Check whether the device is protected. If so, disable the protection before reading or writing a 100TP page
- Check whether the page has already been written 100 times. In order to do so, read the page and check the write counter. Please refer to the firmware user manual for more details
- Check whether you are trying to write a section that is write protected. There are sections that are preloaded prior to shipment and cannot be modified by the user, for example to store a device ID. In case the user tries to write these values, an error is reported and the original content is preserved

Why is the protection not enabled (or not disabled) after setting (or clearing) it?

- A restart is necessary to activate enabling (or disabling) the protection. To do this, the device can be disconnected and connected again with the BSL tool

Why is it not possible to clear the BSL protection of a TLE984x device?

- For TLE984x devices, the protection of CBSL is permanent. If a wrong password is sent, the whole device is erased including the protection. For further details please refer to the TLE984x firmware user manual

Why is it not possible to erase or program a TLE984x device, when only read protection is enabled?

- For TLE984x devices, the read protection does not allow any write operations via LIN BSL or SWD, however, write operations from within the device are still possible, for example for updating the firmware. For further details please refer to the TLE984x firmware user manual

Why is it not possible to enable the CBSL protection?

- Check the size of the CBSL. If it is zero, so technically the CBSL does not exist, no protection can be set

## **4.5 Updates and purchases**

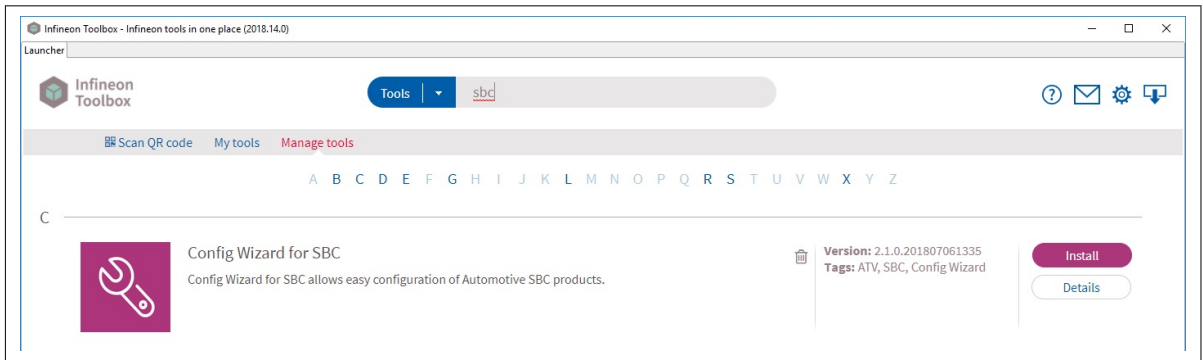
For updates of the BSL tool software or the μIO-Stick firmware as well as for purchasing the μIO-Stick, please visit [www.hitex.com/uio](http://www.hitex.com/uio).

5 Run-time control - programming and monitoring

5 Run-time control - programming and monitoring

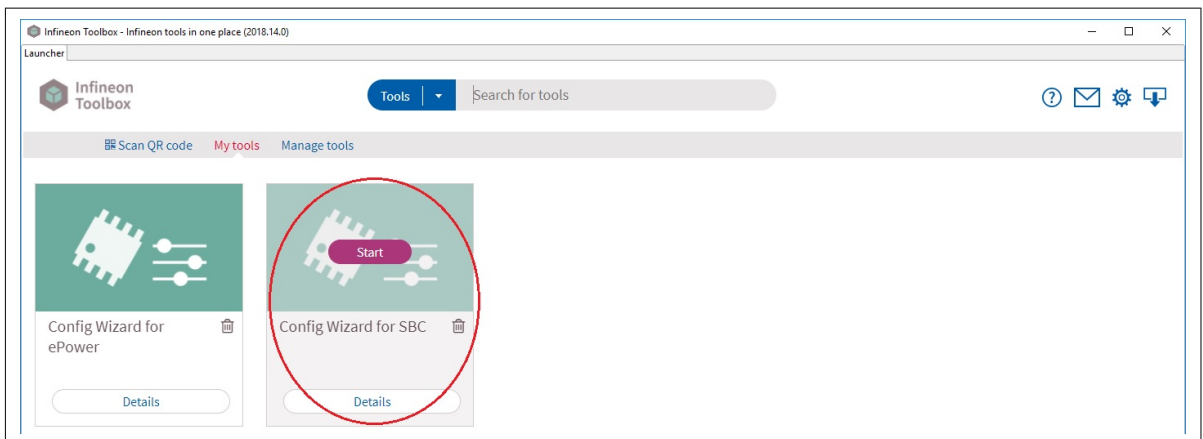
5.1 Installation guideline

1. Download and install the [Infineon Toolbox](#)
2. Select the tab "Manage tools", search for SBC, and click "Install"



**Figure 25 Installation of Config Wizard for SBC**

3. Select the tab "My tools". If the installation was successful, you should see the icon for "Config Wizard for SBC"



**Figure 26 Config Wizard for SBC installed successfully**

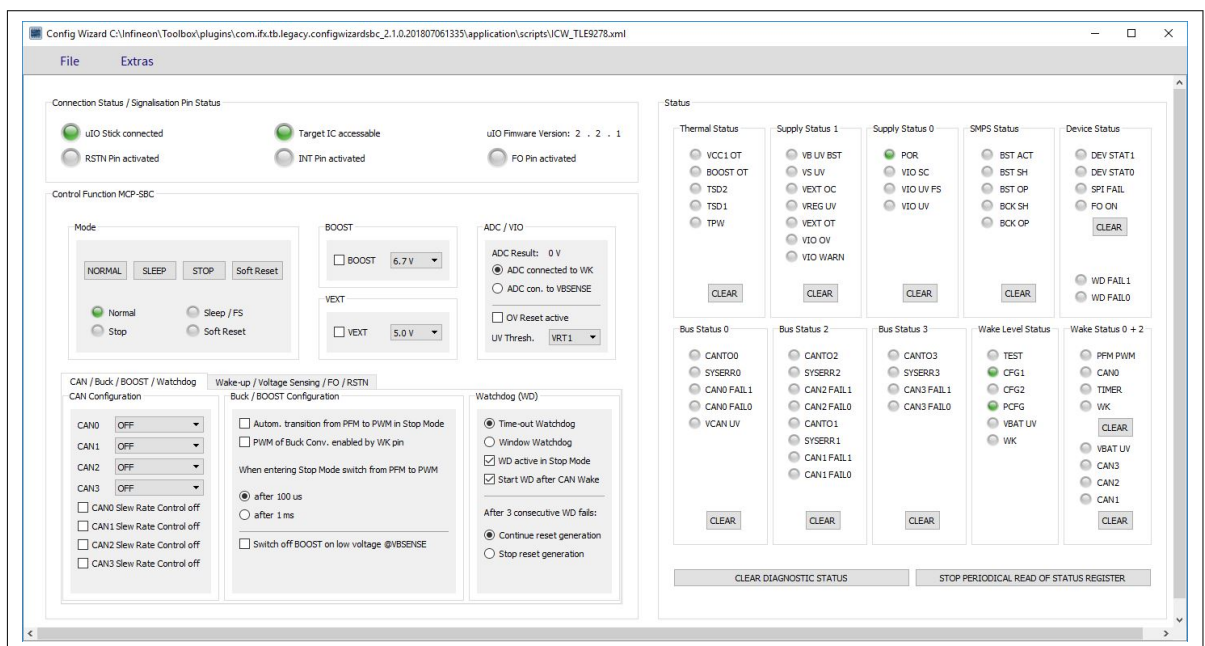
4. Click on "Start" and select the required chip.

## 5 Run-time control - programming and monitoring



**Figure 27 Chips for SBC in Config Wizard**

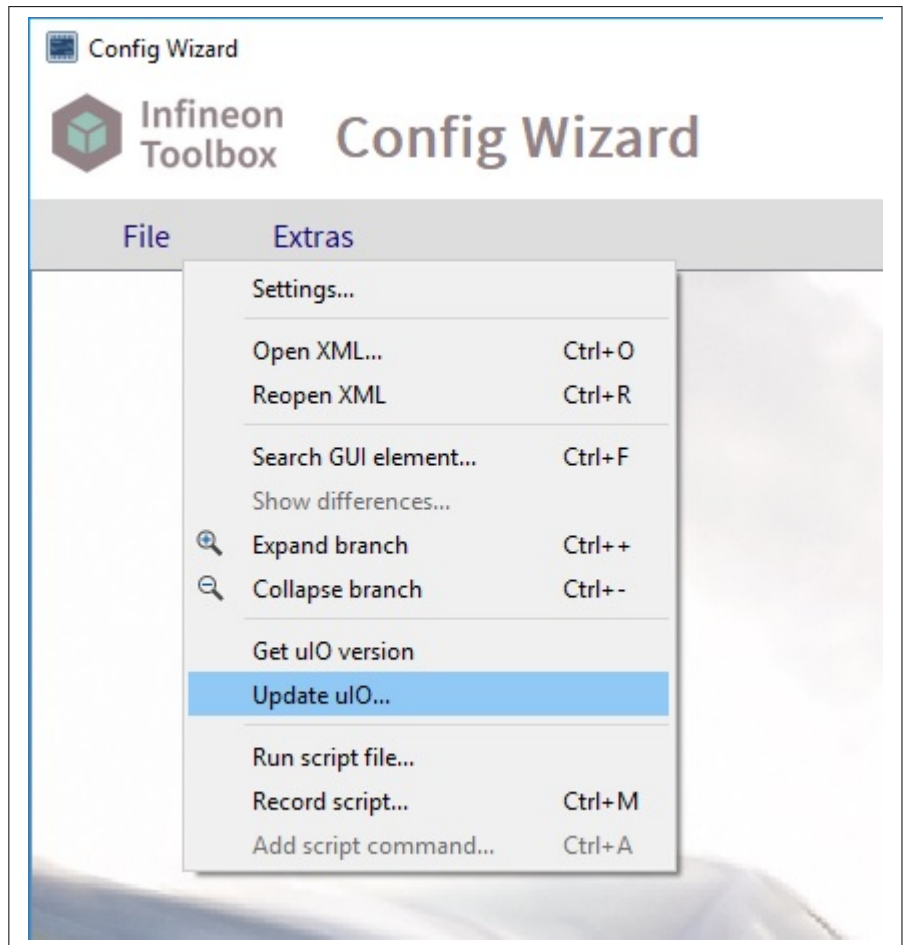
For example, here with the TLE9278-3QX Evalboard correctly supplied, the following screen should display:



**Figure 28 Config Wizard with SBC TLE9278**

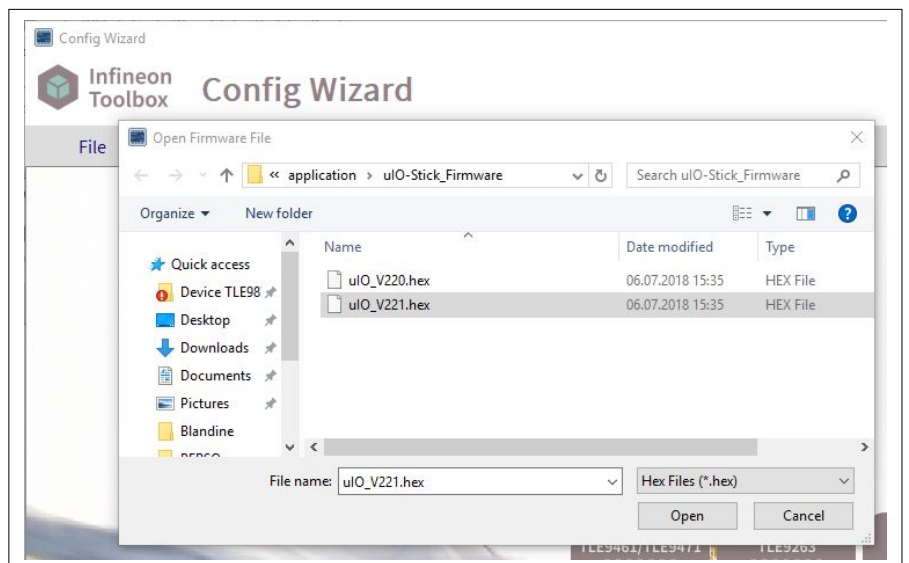
5 Run-time control - programming and monitoring

**Warning:** Please ensure that the latest firmware is flashed onto the μIO-Stick. Otherwise, select the tab "Extra", then click "Update uIO...!"



**Figure 29** Select Extra > Update uIO...

Select the file "uIO\_V221.hex" and click "OK".



**Figure 30** Select HEX file with the right version for μIO-Stick

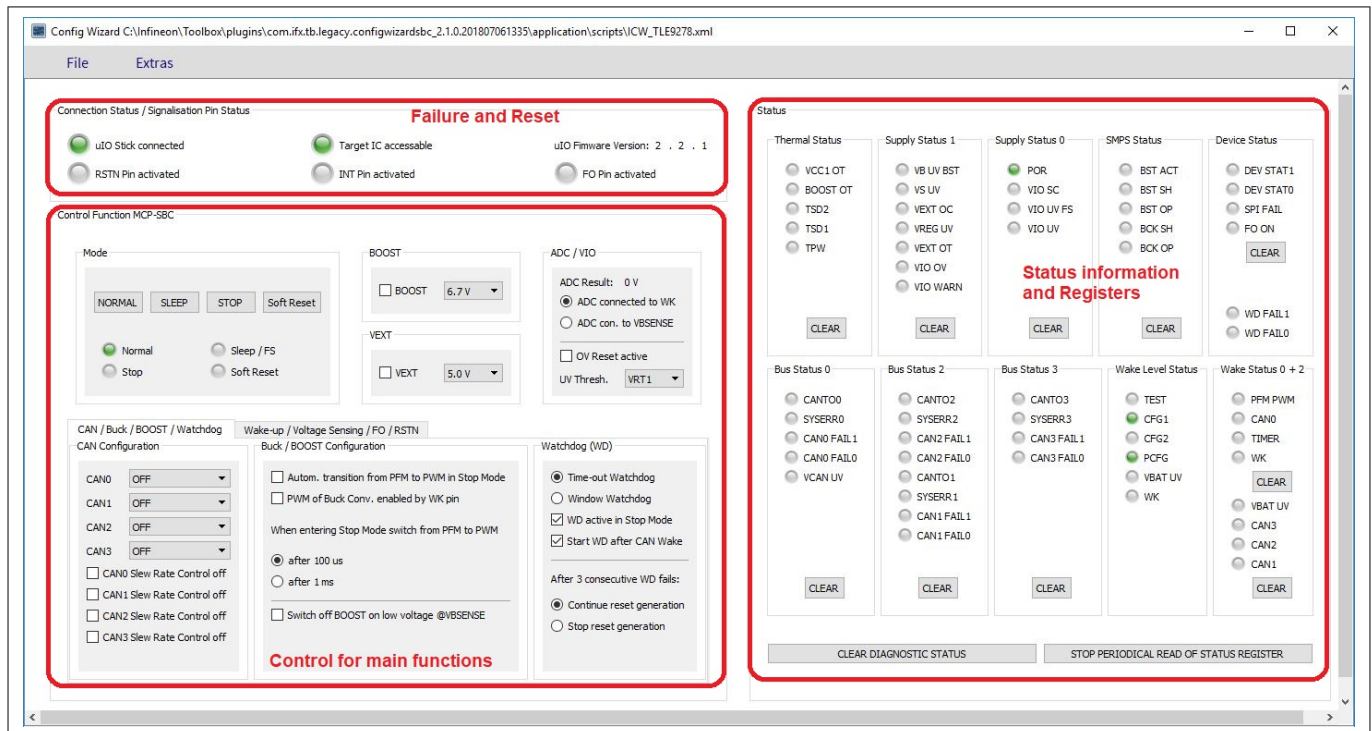
## 5 Run-time control - programming and monitoring

*When the update is completed, the µIO Firmware version should be 2.2.1.*

- Repeat step 1 to 4 for the ConfigWizard for Motor System IC, if required.

### 5.2 Config Wizard

The Config Wizard is a Graphical User Interface which provides an easy access to SBC features and the SPI registers.



**Figure 31 Config Wizard Graphical User Interface**



5 Run-time control - programming and monitoring

Control Function MCP-SBC

**Mode**

**SBC Mode Selection (State Machine)**

NORMAL SLEEP STOP Soft Reset

Normal       Sleep / FS

Stop             Soft Reset

**BOOST**

**Boost Control**

BOOST 6.7 V

**ADC / VIO**

ADC Result: 0 V

ADC connected to WK

ADC con. to VBSENSE

OV Reset active

UV Thresh. VRT1

**CAN / Buck / BOOST / Watchdog**

**CAN Configuration**

CAN0 OFF

CAN1 OFF

CAN2 OFF

CAN3 OFF

CAN0 Slew Rate Control off

CAN1 Slew Rate Control off

CAN2 Slew Rate Control off

CAN3 Slew Rate Control off

**4xCAN Channel Control**

**Wake-up / Voltage Sensing / FO / RSTN**

**Buck / BOOST Configuration**

Autom. transition from PFM to PWM in Stop Mode

PWM of Buck Conv. enabled by WK pin

When entering Stop Mode switch from PFM to PWM

after 100 us

after 1 ms

Switch off BOOST on low voltage @VBSENSE

**Buck Control in SBC Stop Mode**

**Watchdog (WD)**

Time-out Watchdog

Window Watchdog

WD active in Stop Mode

Start WD after CAN Wake

After 3 consecutive WD fails:

Continue reset generation

Stop reset generation

**Watchdog Configuration**

Figure 32 Config Wizard Control Functions



6 Acronyms

## 6 Acronyms

The following table summarizes the acronyms and their meanings throughout this document.

**Table 2 Acronyms**

Acronyms	Names
USB	universal serial bus
BSL	bootstrap loader
CAN	controller area network
GND	ground
GPIO	general purpose input output
GUI	graphical user interface
HID	human interface device
LIN	local interconnect network
MISO	master input slave output
MOSI	master output slave input
NAC	no activity counter
NVM	non volatile memory
PWM	pulse width modulation
SBC	system basis chip
SPI	serial peripheral interface
UART	universal asynchronous receiver transmitter
VS	voltage supply

---

**Revision history**

## Revision history

<b>Document version</b>	<b>Date of release</b>	<b>Description of changes</b>
1.00	2014-11-20	<ul style="list-style-type: none"><li>Initial version</li></ul>
1.10	2018-09-18	<ul style="list-style-type: none"><li>Updates, editorial changes</li></ul>
1.20	2021-12-10	<ul style="list-style-type: none"><li>Updates, editorial changes</li></ul>
1.30	2022-12-07	<ul style="list-style-type: none"><li>Chapter 4.3 Command line tool: update of codeblock</li></ul>

#### **Trademarks**

All referenced product or service names and trademarks are the property of their respective owners.

**Edition 2022-12-07**

**Published by**

**Infineon Technologies AG**

**81726 Munich, Germany**

**© 2022 Infineon Technologies AG**

**All Rights Reserved.**

**Do you have a question about any aspect of this document?**

**Email: [erratum@infineon.com](mailto:erratum@infineon.com)**

**Document reference**

**IFX-eyes1637238880783**

For further information on the product, technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies office ([www.infineon.com](http://www.infineon.com))

#### **Warnings**

Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.