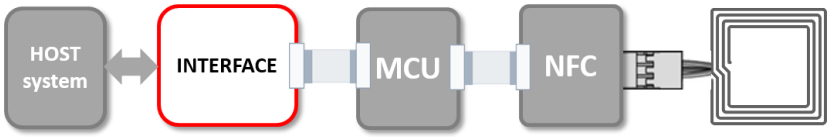
we deliver for your success



NFC NUTSHELL KIT

INTERFACE Modules

USER MANUAL

REVISION 1.3

Document information

Info	Content
Keywords	User Manual NFC Nutshell Kit Interface Modules
Abstract	This document describes how to use different host interface modules of GMMC´s NFC Nutshell KIT

GMMC GmbH
www.gmmc-biz.com

Contents

List of Figures

1. NFC Nutshell Interface Modules

1.1 General Topology

The Kit consists of different types of modules, which can be connected application specific. Each module type has a different color.

- Interface
- MCU
- NFC
- Antenna

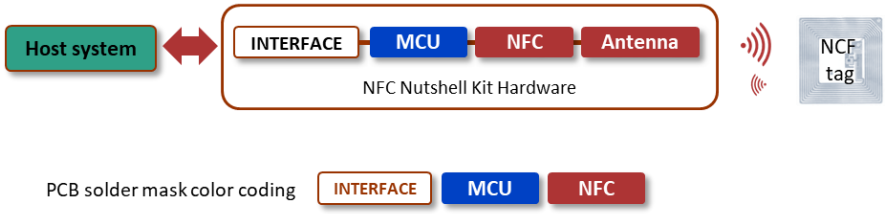


Fig 1. Kit modules and PCB solder mask color coding

1.2 Module interconnection concept

The Nutshell Kit can be operated in different configurations. The connection between the different modules is done via FFC cables and zero force FPC connector.

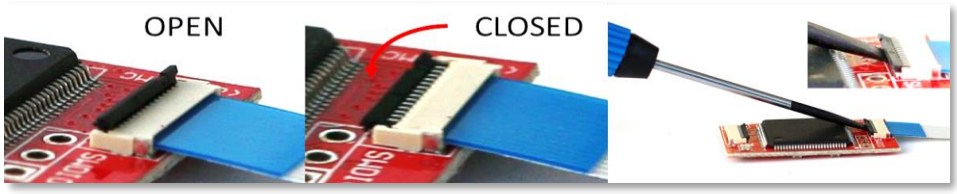


Fig 2 Kit modules interconnection via FFC cables

For ensuring a proper connection the FFC cable must be inserted straight at any time.

Fig 3 correct insertion of the FFC cable

FFC cable: 14 pins 0.5mm pitch, same side. Length 40/50mm.

The Kit modules can be connected to a host via different interface modules see details in the NFC Nutshell Kit interface module user manual.

Important Notes:

- *Open and close the connector **carefully** – do not apply force*
- *Always **insert straight***
- *Avoid cable/connector misalignment – potential short cut and damage*
- *If you observe communication issues over time, exchange the FFC cable with a spare one*
- *DO NOT disconnect, bend or move while modules are powered to avoid short circuits*

1.2.1 Module Connector Signal pin-out

The following figure describes the signal which can be analyzed using the Extender module.

Detailed information can be found in the user manual of the interface modules.

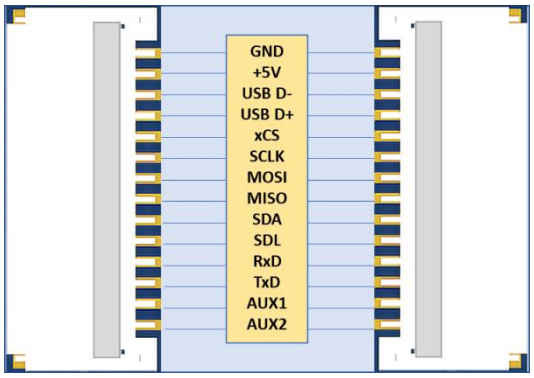


Fig 4. Module Zero Force FFC Connector Signal Pinout

1.3 Module power concept

The Kit requires a power supply of +5V DC. This can be applied using one of the following options:

- Using one of the interface module
  - USB supply voltage (4.75V to 5.25 V) is available on all connected modules using the USB plug.
- Using an external power supply and the Signal extender module
  - +5V is required

If other voltages than 5V are needed by the module, these are generated onboard via an LDO.

Important Note:

If using an external power supply, ensure the correct amplitude and polarity before power up of the Kit.

2. Interface Modules

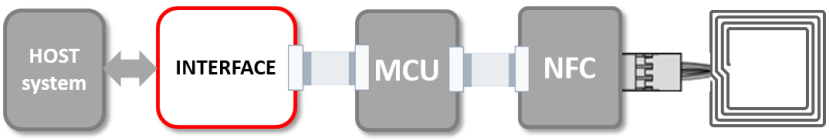


Fig 5 Interface Module

At the moment we offer three different types of Interface modules.

- **USB Protocol Converter features**
  - USB to I2C bridge
  - USB to SPI bridge
  - USB to USART bridge
  - USB Flash Magic In-Circuit Programming
  - USB NFC direct mode use the NFC frontend without MCU

- **USB Plug with Boot features**
  - USB connector for MCU-based USB applications
  - Boot Button for USB Mass Storage Device firmware upload (if supported by MCU)

- **Signal Interface and Debug extender**
  - Access to all connected Pins
  - Separate test pin for scope probe

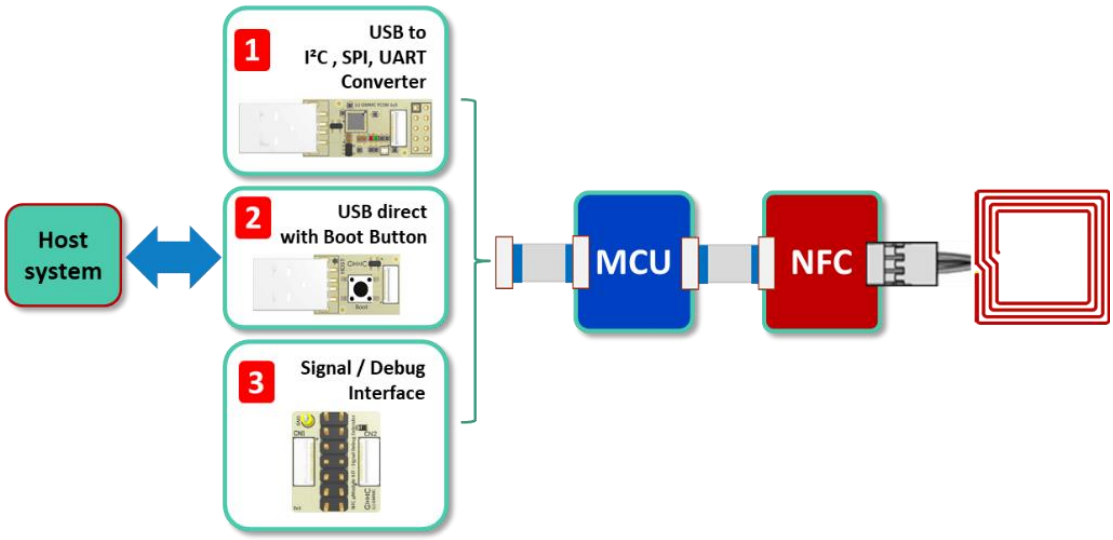


Fig 6. Host interface Module options

Depending in the used Interface module and selected interface, the communication with the Host system can be implemented in the firmware.

3. USB Protocol Converter Module

This modules allows to implement different interfaces on the MCU and communicate with the Host.

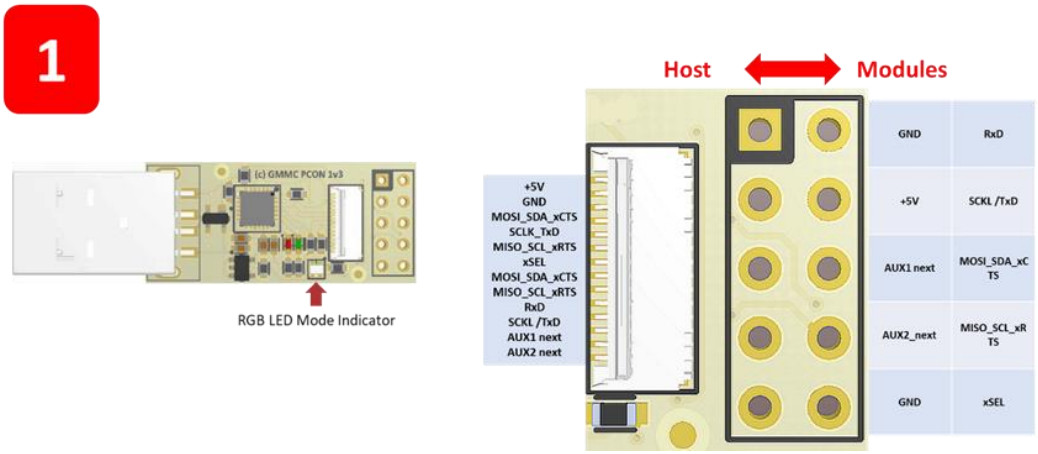


Fig 7. USB to I2C, SPI. UART Protocol converter module

This module features the CY7C65211A from Cypress. It can be used as an USB to I2C, SPI or UART Bridge.

The different operating modes need to be programmed into the Converter Module with GMMCs Protocol Converter Configuration Tool.

The selected (programmed) modes are indicated by the on board RGB LED on the converter Module.

Important Note:

For correct operation the Cypress USB driver needs to be installed. This driver can be found on the official cypress homepage or on the GMMC-homepage in the download section of the NFC Nutshell Kit.

3.1 GMMC's USB Protocol Converter Configuration Tool

This configuration tool allows to set the different output protocols like UART, SPI or I²C as well as the direct communication and the FW download mode via Flash Magic Tool.

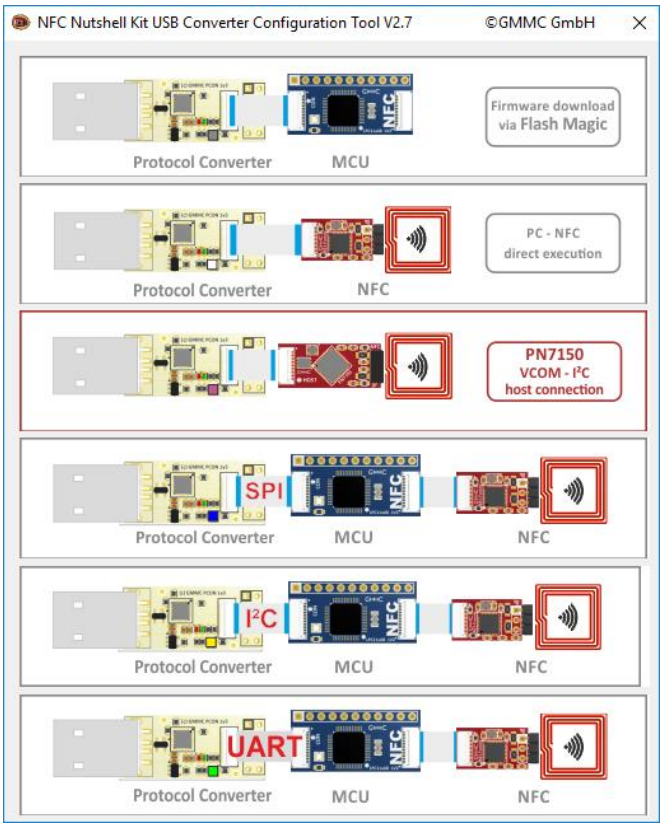


Fig 8. Protocol Converter Configuration Tool

The picture above shows the Protocol Converter Tool which can be used to change the mode of the Converter module. The tool can be downloaded from GMMC webpage here.

3.2 Hardware setup for Converter Mode programming

For changing the mode of the converter it has to be connected as shown next:

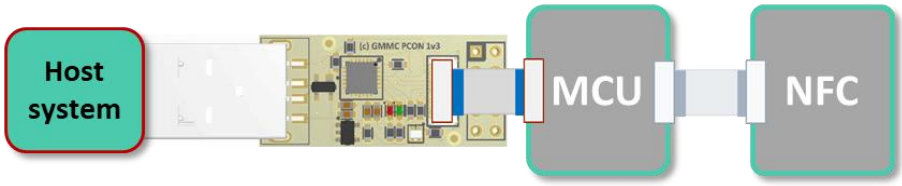


Fig 9. Protocol converter hardware configuration

The power is supplied via USB. Depending on the mode of the converter the RGB LED in the lower right corner is colored.

3.3 Step by step – Converter mode programming

1. Before using the CONVERTER module, the required Protocol Converter Configuration Tool and the driver installer needs to downloaded and installed on your system.
2. After the installation is completed, the CONVERTER module has to be configured in the required mode with GMMC´s software tool, which automatically detects the current configuration and indicates the mode with the respective color code.
3. Press the picture indicating the desired configuration, for programming the respective mode
4. After you have confirmed the change, the device gets reprogrammed and **you need to un-plug and re-plug the device**, so that the new settings become effective.
5. The RGB-LED of the Converter now shows the current configuration:

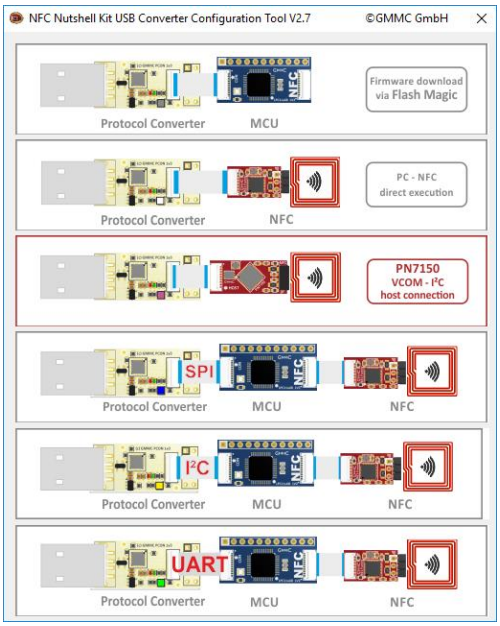


Fig 10. Protocol Converter Configuration Tool

Selected Mode	RGB LED color
UART	GREEN
I2C	YELLOW
SPI	BLUE
UART PC-NFC direct	WHITE
Flash Magic Tool for FW download	RED
PN7150 with enable	MAGENTA

Fig 11. Protocol Converter Configuration Tool

Note:

As some data lines are shared between the interfaces, only one HOST-interface can be used at the same time.

3.4 I2C Bridge Mode

In this configuration the Converter Module acts as I2C Master. The user hast to ensure, that the counterpart, the MCU module implements an I2C-Slave.

For the communication the GMMC´s own -Terminal program SangomaTerm can be used.

This host application polls the I^2C Bus for new data and prints it to the Terminal.

The following picture shows the hardware setup of this mode:

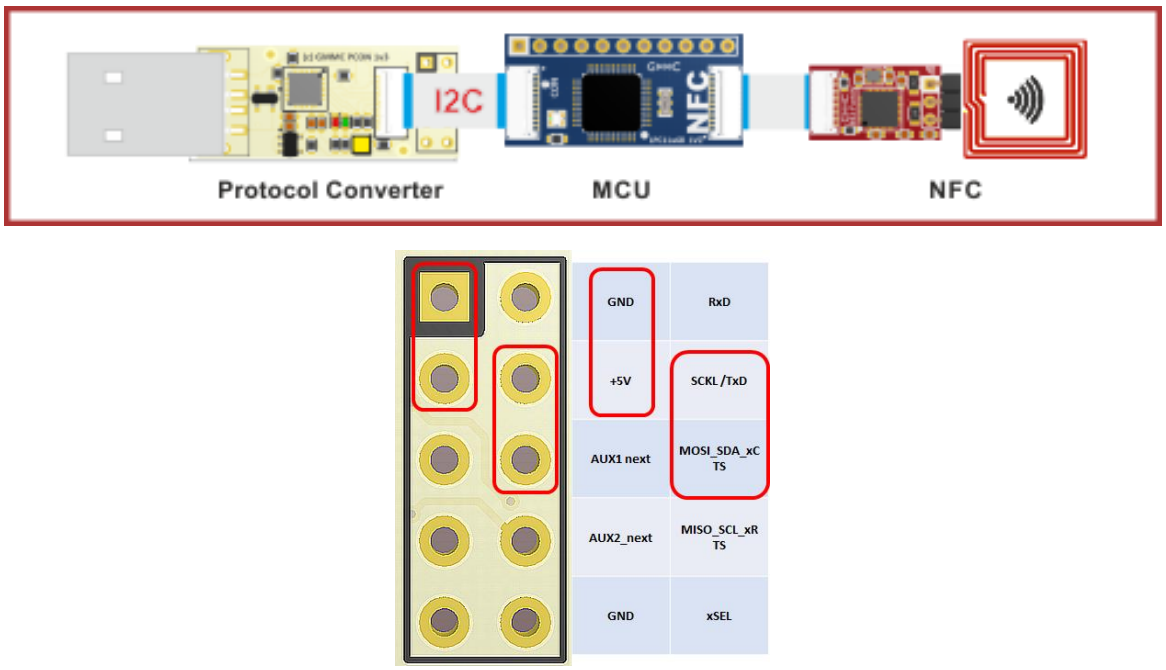


Fig 12. I2C Mode – Tool image selection and PCB signal pin out

Note:

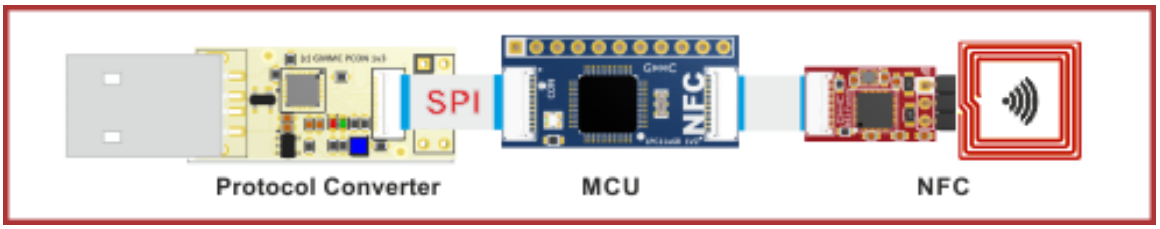
The use of the NFC Frontend is optional.

3.5 SPI Bridge Mode

This configuration implements a SPI Master in the converter module. It uses the following settings:

- Frequency 1MHz
- 8Bit Data
- SPI Mode: Master
- Protocol: Motorola
- CPHA mode: low
- CPOL mode: low
- SSN Toggle mode: Continuous
- Bit order MSB first

The topology of this mode is shown next:



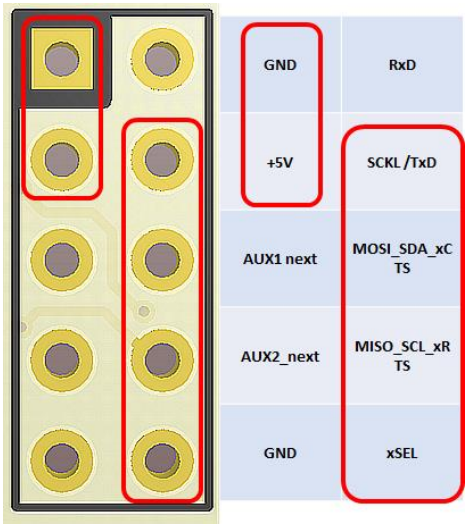


Fig 13.SPI Mode – Tool image selection and PCB signal pin out

The SangomaTerm -Terminal program can be used for communication with the MCU module, where an SPI slave has to be implemented. Also here the host application polls the device and prints new data.

3.6 UART Bridge Mode

This configuration implements an USB to UART Bridge. Typical terminal programs, or the GMMC-Term can be used for communication with the MCU module.

The topology of this mode is shown next:

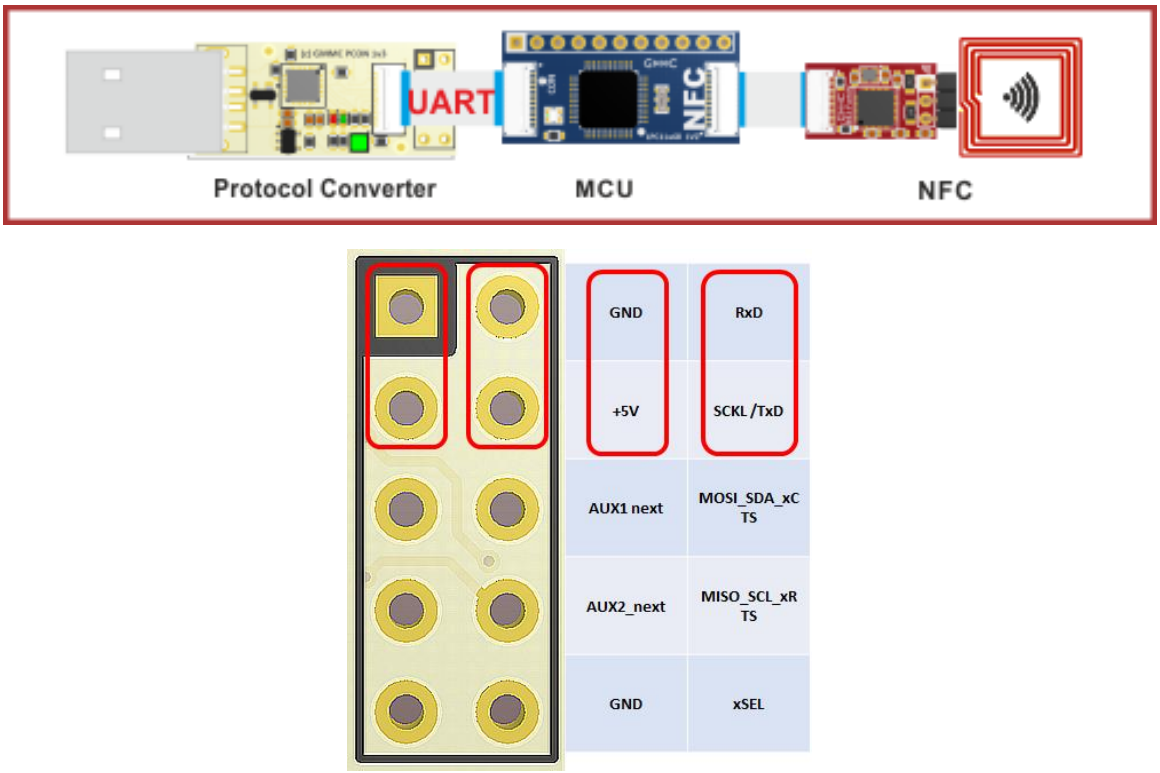


Fig 14.UART Mode – Tool image selection and PCB signal pin out

As the bridge implements a UART to CDC Bridge, the baud rate can be defined by the MCU and the application. The highest recommended baud rate is 1Mbit/s.

Note: *The baud rate is limited by the used cables, connecting the modules.*

3.7 UART Flash Magic FW Download Mode

This configuration allows to program the MCU using flash magic. This can be used for supported MCUs with the current version of the Flash Magic Tool.

In this special mode, the converter uses GPIOs to set the MCU in UART-ISP-Mode.

This means that as long as the converter is operating in this mode, the MCU executes no user code!

For downloading firmware into the MCU, the Flash Magic Tool is needed and can be downloaded here.

The topology of this mode is shown next:

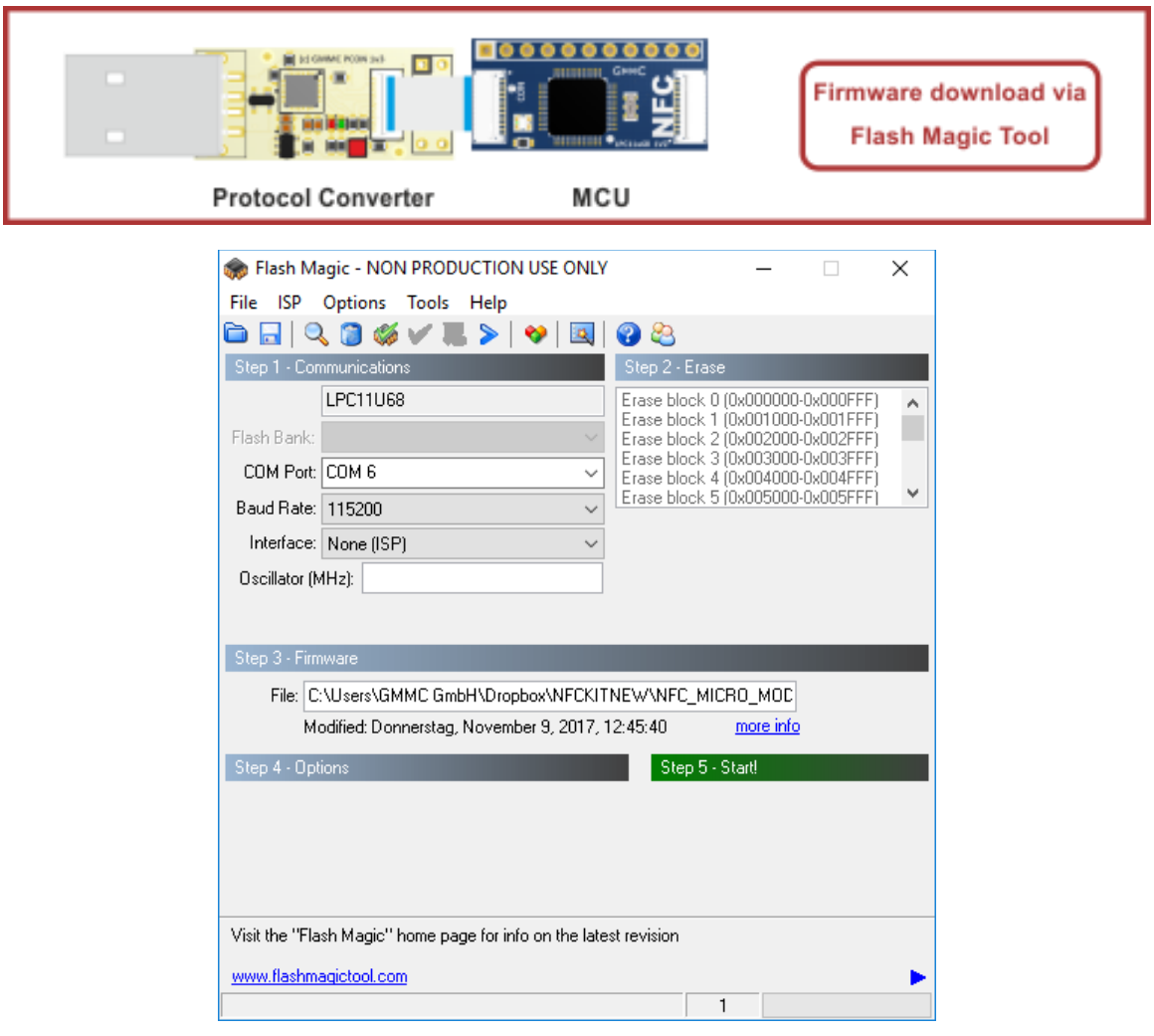


Fig 15. Flash-magic programming mode

The picture above shows the use of Flash Magic. The user has to modify the MCU as well as the used COM-Port.

3.7.1 How to download an application with Flash Magic

1. **Select correct CPU**
   (depends on the used module)
2. **Select correct COM-Port**
   (assigned COM-Port is different for each system and can be found in the OS device manager)
3. **Select *.hex-file**
   (only HEX-files are intendent to be flashed using this tool. Precompiled*.bin files are intendent to be used using the USB-Mass storage bootloader which is described in the USB-Plug-module section)
4. **Press Start**
   wait until download (and if selected the verification procedure) is completed
5. After the flashing and verifying procedure, the program is loaded into the microcontroller. To start the user application the converter has to be configured to another interface. This is needed because in the UART-Flash mode the module manipulates some data lines of the microcontroller, which keeps the MCU in flash mode.

Important Note: *If the MCU application is designed to communicate via USB directly, the converter module needs to be exchanged to the USB plug module with boot button to the below configuration:*

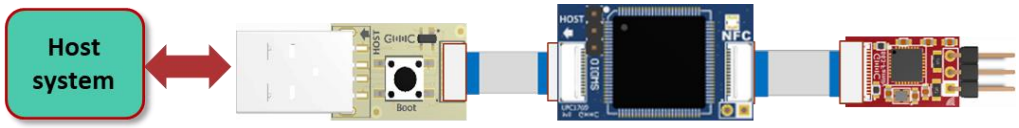


Fig 16. Topology for using the MCU based USB interface

3.8 PC – NFC Direct Communication Mode

This options allows to use the NFC Nutshell Kit in its minimum configuration, consisting of the converter, the analog frontend and the antenna only.

This Converter Module operating mode differs from the normal UART Bridge mode and the UART Flash mode, as some GPIOs are configured to set up the NFC frontend as a UART device.

The advantage of this configuration is, that the developer can create his own application in C++ and execute directly via the PC on the frontend using the converter, so no MCU is needed.

The topology of this mode is shown next:

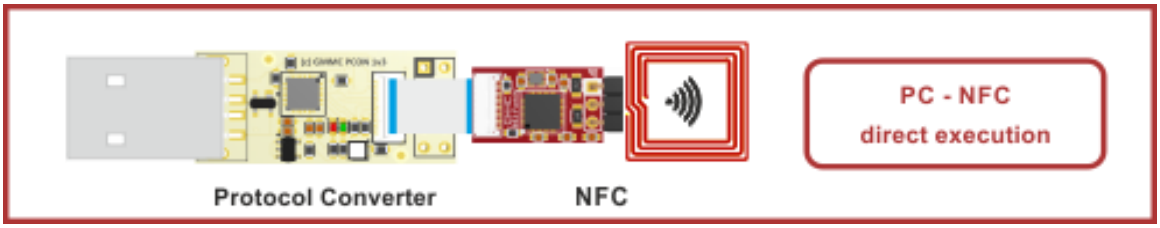


Fig 17. PC - NFC direct execution

***Note:** there might be timing constraints in high speed applications like NFC P2P.*

For this mode the C++ version of the NXP Reader library and Microsoft Visual studio are needed / have to be installed on the development system.

***Note:** it is a pre-requisite that the developer is familiar with Microsoft Visual studio. Access to the NXP Doc-Store is also mandatory for using this option.*

3.8.1 Using the NFC-Lib examples with Visual Studio and direct Interface Mode

Ensure that the converter module is configured in the right mode (NFC-Direct Mode). This can be verified by the color of the RGB-LED (white).

Navigate to the installation folder of the NXP-Reader library. And go to /ex/Rc663_Mful for example.

Open the *.vcxproj Visual Studio project. Change the COM-PORT to the one matching with your system (this can be found in the device manager) and start the program.

As mentioned above, the program is entirely running on the computer, there is no MCU in between.

Note:

Only the export-controlled version of the NXP reader library can be used for this purpose.

3.9 PN7150 – VCOM → I2C Communication Mode

This PN7150 dedicated configuration option is needed as the converter module sets the PN7150´s enable pin via the AUX2 line to ensure proper function.

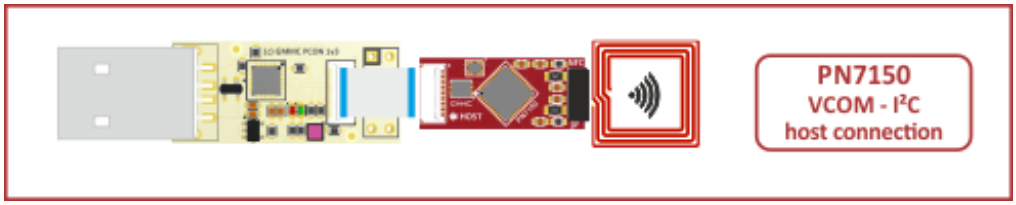


Fig 18. PN7150 - VCP I²C communication

3.10 Writing own applications

For implementing own PC applications, Cypress offers libraries with feature all needed functionality. This library can be downloaded from the cypress homepage.

The following settings are used for the different interfaces using the Protocol Converter Tool:

- UART
  - 115200 baud
  - 8n1
  - 2 wire (only RXD TXD)
- SPI
  - Master Mode
  - 1MHz
  - 8 bit Data
  - MSB first
  - CPHA = low
  - CPOL = low
  - Motorola format
  - SSN toggle continuous
- I2C
  - Master Mode
  - 100kHz

All other interface settings can be set using the Cypress USB Serial Configuration Utility, which can be found on the Cypress Semiconductors homepage.

Note:

All modes of the converter module are implemented using the CDC mode of the converter. This allows to work on all platforms, as USB CDC is a standard build-in device class.

4. USB Plug Module with Boot Button

This module allows the use of the MCU integrated USB functionality (supported MCUs only). It can also be used to trigger the USB Mass Storage bootloader (supported MCUs only), by using the Boot Button (pressed during power up).

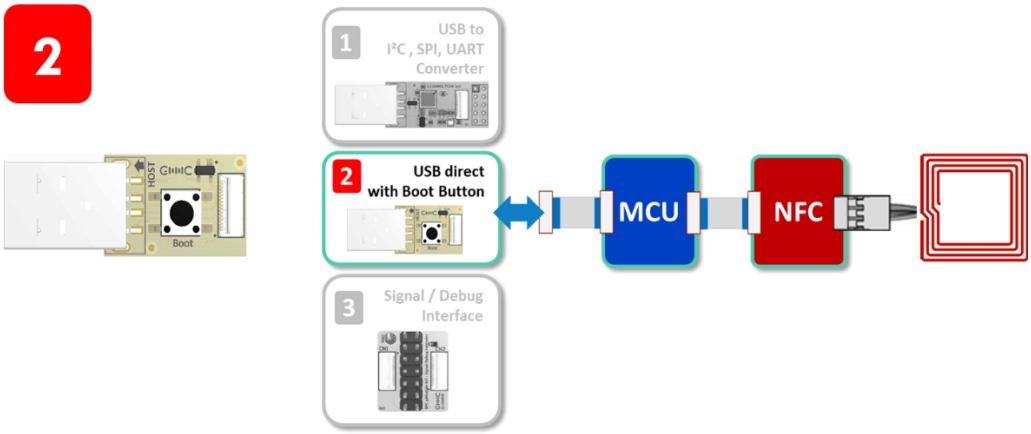


Fig 19.USB operation with MCU

Note:

depending on the used USB protocol, the cable length must be limited to ensure proper operation.

4.1 USB Mass Storage Device Application Flash

The following picture shows the procedure to enter USB Mass Storage Bootloader if supported by the selected MCU:

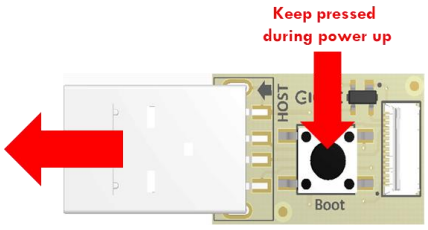


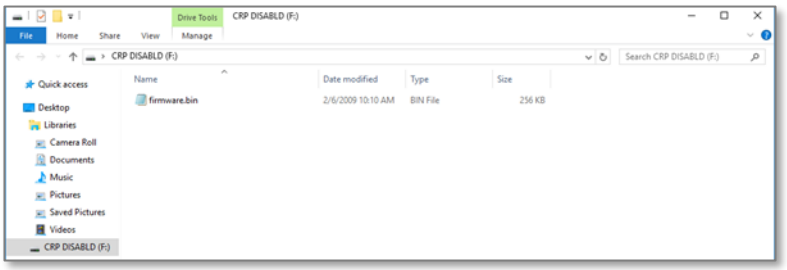
Fig 20. Enter MCU Bootloader mode when supported by the selected MCU

For downloading new firmware, the following supported MCUs offer a build in mass storage device bootloader.

- PN7462
- LPC11U68
- LPC1769 (See NXP AN10866 for more details)

4.1.1 Step by Step – enter mass storage device mode

1. Keep button pressed during plug-in (power up) of the module.
2. After several seconds a mass storage device will appear on the computer.
   - Beside the MCU name e.g. PN7462, the Code Read Protection (CRP) status is shown.



3. Delete the existing file called “firmware.bin” from the device (selecting the file and hit DEL button)
4. Paste the new firmware
5. Wait until the file has been entirely copied + another 2-3 seconds
6. To ensure, that the new firmware is written into the mass storage device, remove the mass storage by clicking on the USB-Symbol in the task bar and selecting “Eject LPC1XXX FLASH”.
7. To execute the new firmware, the user has to un-plug and re-plug the device to boot with its new firmware.

Detailed information about this CRP can be found in the User Manual of the respective MCU. In general, it is used to restrict the read or write capabilities for new firmware. Please be aware, that in CRP level 3, no further programming/firmware update are possible.

Note:

This mass storage device only supports the delete and paste commands.

4.1.2 Flashing Firmware into PN7462 family members

For using the PN7462 family, the flashing procedure looks different. The mass storage device contains four files instead of usually one file.

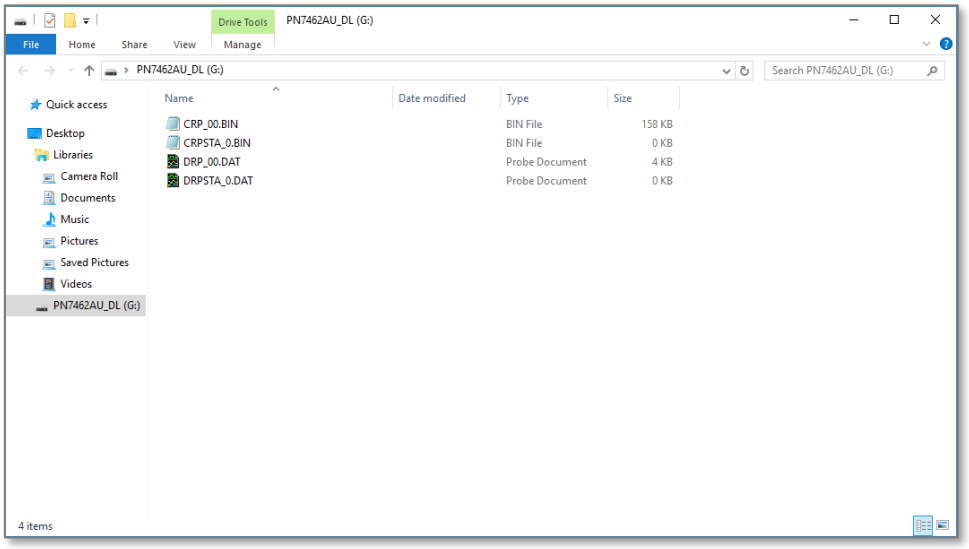


Fig 21. PN7462 family – firmware files

- CRP_XX.BIN is the current firmware
- CRPSTA_X.BIN indicates the status of the progress.

For the CRP_XX.BIN file XX indicated the different CRP levels, which can be found in the PN7462 User Manual. The CRPST_X.BIN file indicates if the firmware write was successful or not.

Like for the other mass storage devices this also only support the delete and paste commands. For flashing new firmware, the user must delete the CRP_XX.BIN file and paste the new firmware.

After that the device, automatically un- and re-plugs. After that the CRPSTA_C.BIN file shows the Status of the previous flashing procedure. If the file is called CRPSTA_0.BIN the new firmware was flashed successfully. Otherwise the user must delete the CRP_XX.BIN file and paste the new firmware again.

The mass storage bootloader of the PN7462 also supports to set the EEPROM of the MCU. This is what the DRP_XX.DAT and DRPSTA_X.DAT files are used.

The EEPROM of the PN7462 holds lots of configuration parameters for the RF frontend and interface. Therefore, it is needed to also program this for correct operation. The procedure is the same as for the firmware, but the name of the files differs.

After the CRP and DRP files are renewed, the module must be un- and re- plugged again (without holding the button), for executing the new user application.

5. Signal Interface and Debug Extender Module

This module features different debug and extension capabilities.

- Extend the NFC Kit with other components using the Pin header
- Enhanced signal debug possibilities
- Allows to connect to
  - external analyzer tools like an oscilloscope or logic analyzer
  - embedded host MCU

Note:

Do not connect, move or bend assembled setup while powered.
Don't forget to power the NFC Kit, if no other USB powered Interface module is connected.

The following figure shows the topology for using the Signal Interface module:

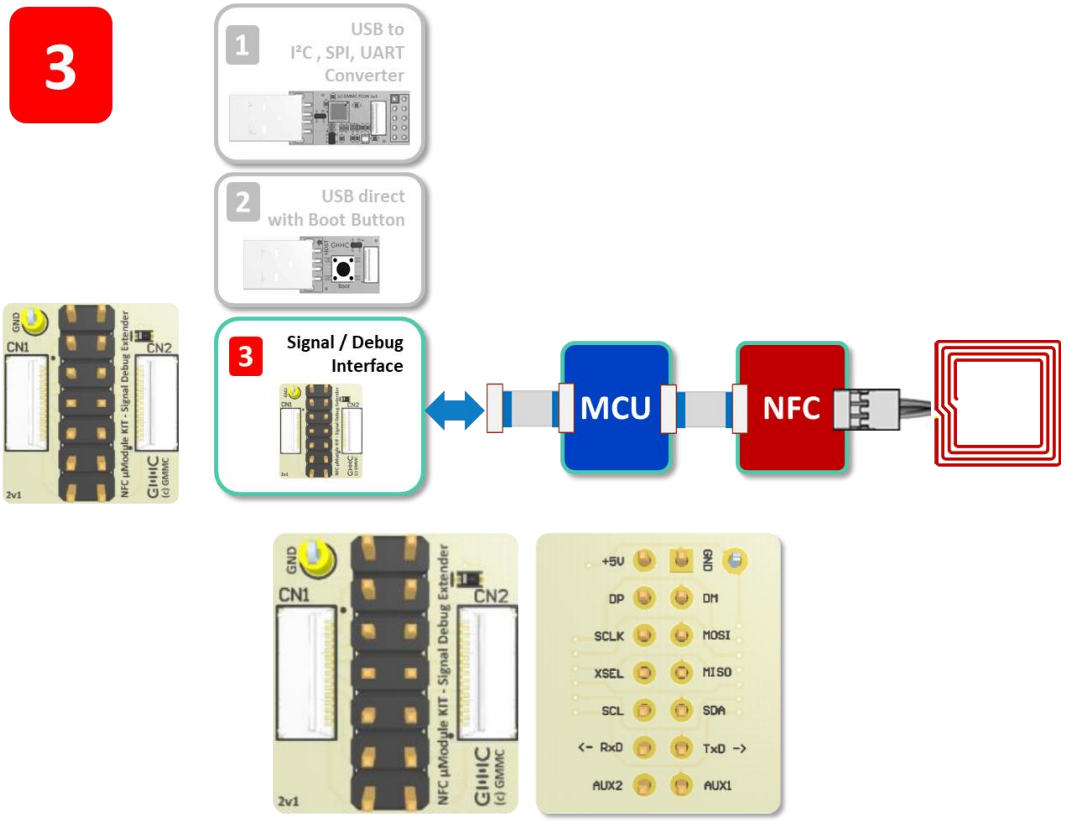


Fig 22. Signal Interface and debug setup top and bottom view

6. Software Tools

6.1 Required and supported software

In order to be able to benefit from all different available operating and connection modes the following software packages are needed.

- Cypress driver for Protocol converter (Windows, xxx) www.gmmc-biz.com
- GMMC´s Protocol Converter Configuration tool (Windows only)
- Flash Magic Tool (optional for MCU firmware download on Windows, Mac OS, Linux)
- MCUXpresso for PC based application development (Windows, Mac OS, Linux)
- Microsoft Visual Studio C++ for Windows based NFC application development
- NXP Reader Library for PC and NXP Reader Library User Manual
- NFC Cockpit configuration tool for NFC ICs

Note:

The software packages can be found for download via GMMC GmbH and NXP Semiconductors websites. Some versions of the tools require access to NXP´s Doc store. See here NXP´s video tutorial for user registration.

Revision history

Rev	Date	Description
1.0	2018 Feb. 4th	Initial version MGA
1.1	2018 Feb. 5th	Updated Template TPI
1.12	2018 Feb. 6th	Added features TPI
1.2	2018 Mar. 18th	CleanUp TPI

Note: *All product names, logos, and brands are property of their respective owners.*