AN11744

PN5180 Evaluation board quick start guide

Rev. 1.3 — 2 February 2018
371213

Application note
COMPANY PUBLIC

Document information

Information	Content
Keywords	PN5180, PNEV5180B, PN5180 evaluation board, PN5180 customer board, PN5180 GUI, GUI, PN5180 Support Tool, NFC Cockpit
Abstract	This document describes the PNEV5180B 2.0 (PN5180 evaluation board), and how to use it. It describes the NFC Cockpit (PN5180 GUI Version 4.7), which allows an easy basic access to the PN5180 registers and EEPROM in combination with basic reader functionality.

Revision history		
Rev	**Date**	**Description**
1.3	20180202	Update for NFC Cockpit 4.7 (VCOM version), RxMatrix test added, Scripting added
1.2	20160407	Update for NFC Cockpit 2.3, EMVCO App added
1.1	20151125	Section 8 References updated
1.0	20151124	First release

1 Introduction

This document describes the PNEV5180B 2.2 (PN5180 evaluation board), which provides an easy evaluation of the features and functions of the PN5180.

It provides the first steps to operate the board, using the NFC Cockpit (PN5180 GUI Version 4.7).

The default antenna is a 65mm x 65mm antenna with some metal layer inside the antenna area. This antenna is not an optimum antenna as such, but intends to demonstrate the performance and register settings of the PN5180 under typical design constraints like LCD or some metal (e.g. PCB) inside the antenna area.

1.1 PN5180 registers & EEPROM concept

The PN5180 uses internal registers to adapt and optimize the functionality and performance for each of the supported protocols and data rates dependent on the connected antenna, matching network and receiver path. It offers an EEPROM, which contains the default settings for all the supported protocols. These settings are loaded into the registers with the LOAD RF_CONFIG command for each supported protocol and data rate

The default EEPROM configuration settings are optimized for the 65mmx65mm antenna of the board PNEV5180B, and can be updated by the user in case a customized antenna and matching network is used. The command LOAD_RF_CONFIG allows to initialize multiple registers with an efficient single command, and allows to distinguish between transmit and receive configuration. Update of the registers relevant for a selected protocol is done by copying the content of EEPROM addresses to registers. Not all protocols require the initialization of all or the same registers, the command LOAD_RF_CONFIG considers this by initializing the registers relevant for the currently selected protocol only.

The EEPROM content can be updated using the command UPDATE_RF_CONFIG. The command does not require any physical EEPROM address, but works directly with the register address information and the protocol for which this data is intended to be used. This allows a convenient initialization of all relevant values for operation.

Some of these settings can or even **must** be adapted towards a new antenna design (e.g. the DPC calibration). All those settings should be stored in the PN5180 EEPROM to allow a proper functionality.

Some EEPROM configuration data is independent from the used protocols and defines e.g. the startup behavior of the PN5180 or the functionality of Low Power Card detection and requires attention as well for optimum performance of the chip.

1.2 PNEV5180B concept

The basic **concept of the PNEV5180B** is to enable the user to perform a quick evaluation of the PN5180, and also connect his own antenna to the PNEV5180 board. In addition, dedicated boards which allow to solder custom matching components are available. The NFC Cockpit can be used to optimize the PN5180 antenna tuning, to perform the DPC calibration and the related TX and Rx optimization without touching any source code.

All the relevant PN5180 registers can be modified and fine-tuned using the NFC Cockpit. After successful register optimization, the found settings can be stored in the PN5180 EEPROM.

The NFC Cockpit also allows a dump of the complete user EEPROM content into an XML file. This file then can be loaded again into the EEPROM. That allows to manage and exchange different user or antenna configurations. In addition, the register settings found to work well using the NFC Cockpit, can be used during user code development as well.

As soon as the register settings for the targeted protocols and data rates are defined, the NFC Reader Library including the HAL can be used to start the development of the user application. Examples illustrate the usage of the library for typical use cases.

The source code examples of the NFC Reader Library can be used to develop an own application directly on the LPC1769 (see [5]), or can serve as a starting point for porting the NFC Library to any other microcontroller platform.

2 Hardware

The PNEV5180B V2.2, as shown in Fig 1, provides a lot of test functions which might not be used for the typical hardware and software evaluation. It can be used as a simple standard reader without modification, it can be used to define and optimize the analog settings for any connected antenna or it can be used to develop and modify any RFID and NFC application based on the NFC Reader Library.

2.1 Hardware introduction

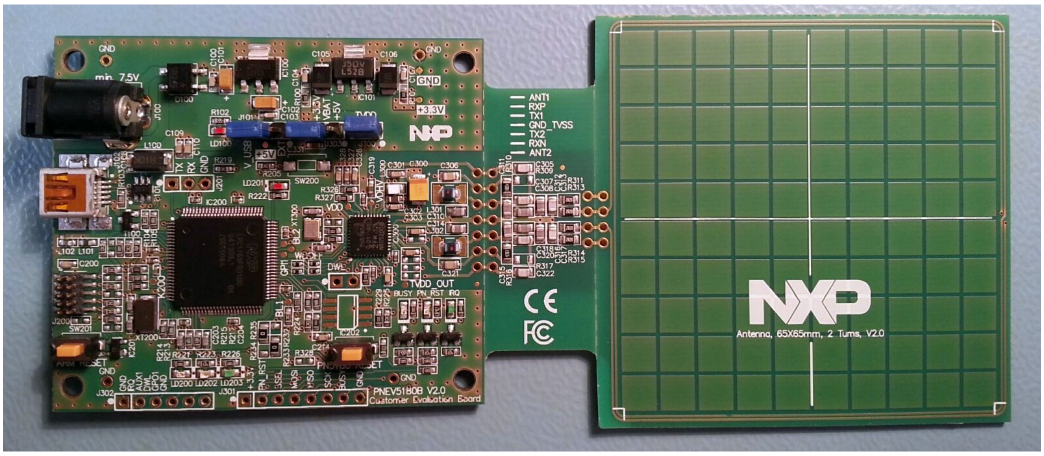
The PN5180 is supplied with a supply voltage, which can be chosen between: internal and external supply. For the internal supply either 5V or 3.3V can be used. The external power supply can be an AC or DC supply (polarity does not matter) with at least 7.5V, since the board provide a rectifier and LDO to supply the circuit with 5V and 3.3V.

The PN5180 is connected to an NXP LPC1769 µC via SPI. A specific firmware on the LPC1769 allows to use the PNEV5180B in together with the NFC Cockpit.

The connection to the PC is done via USB. Both USB Mini and USB Micro connectors are supported.

Another connection option allows to connect a LPC-LINK2 board the PN5180B by means of a debug cable. This allows the development of custom software or the execution of the NFC Reader Library code including samples.

In case a different host microcontroller shall be used, the SPI interface is available for connection to an external host (the on board LPC1769 is not used in this case).



1. Version 2.0

Figure 1. PNEV5180B Customer Evaluation Board

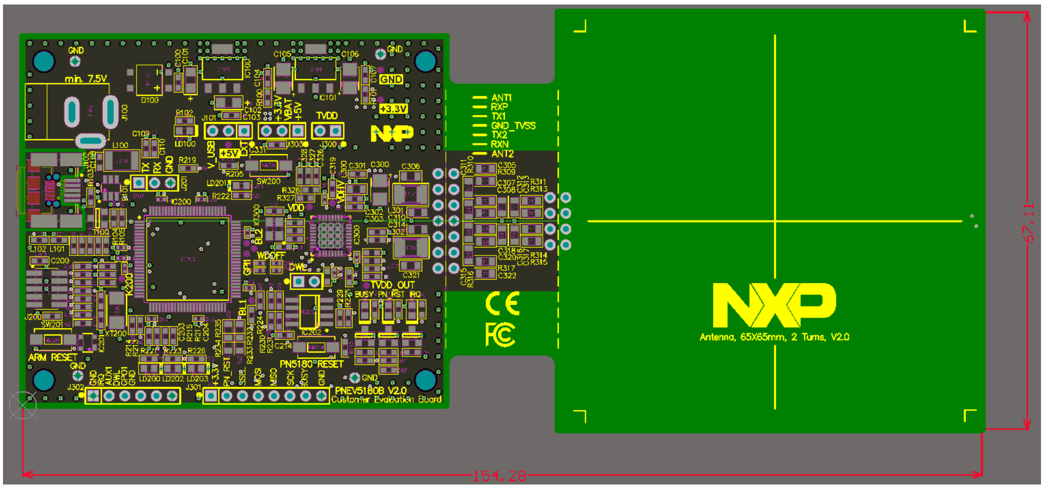


Figure 2. PNEV5180B 2.0 top view

2.2 Schematics

The complete schematics of the PN5180 evaluation board are shown in the Fig 3, Fig 4, Fig 5, Fig 6, and Fig 7.

2.2.1 LPC1769

The PNEV5180B contains an NXP LPC1769 (see Fig 3).

An LPC Linker can be connected to the LPC1769 via the JTAG interface (see Fig 4).

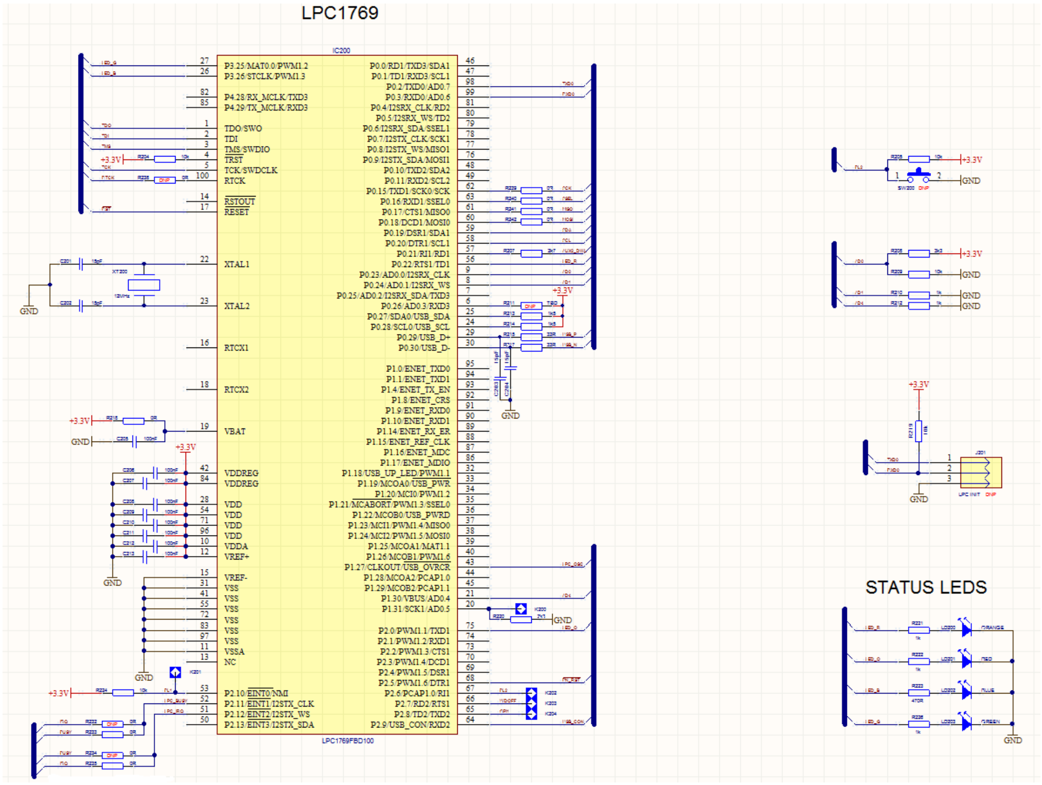
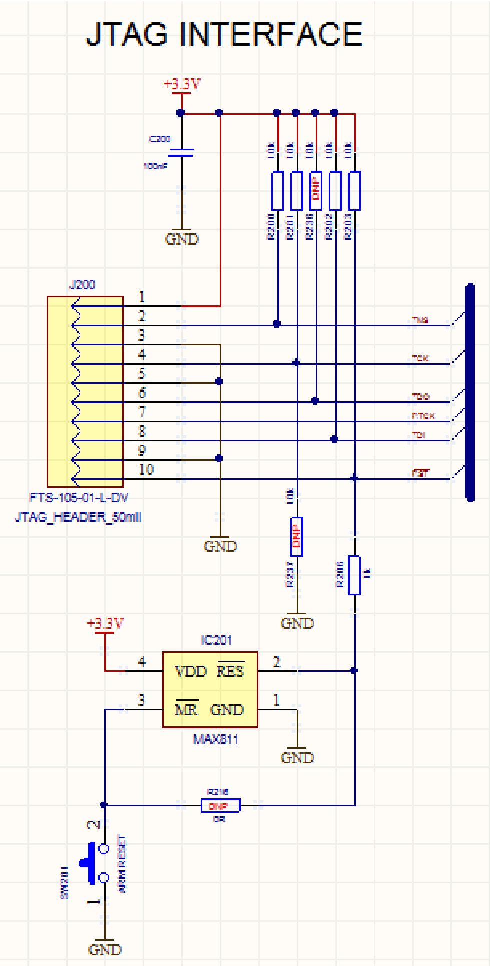
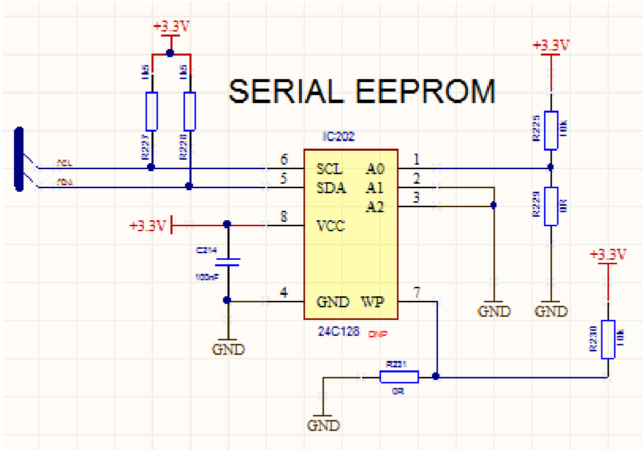


Figure 3. LPC1769



JTAG Interface

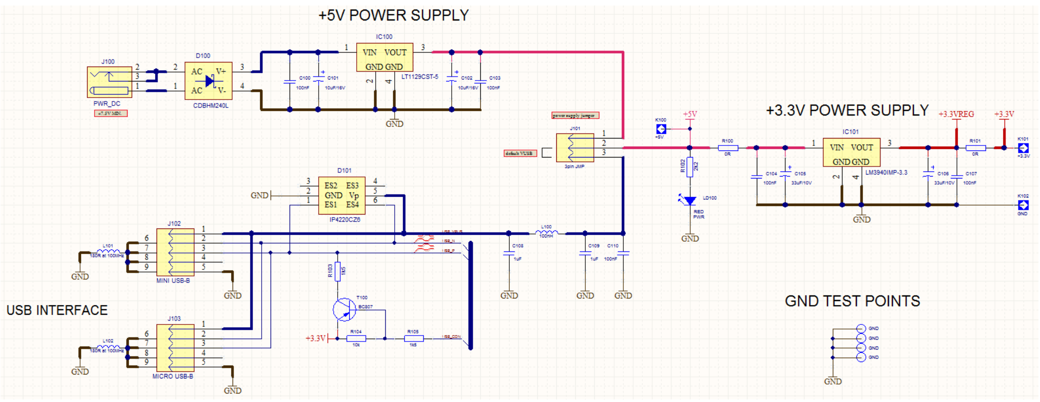


Serial EEPROM not assembled

Figure 4. JTAG interface and Serial EEPROM

2.2.2 Power supply

The default settings use the power supply from the USB connector. For the maximum performance and a better test capability the external power supply should be connected. The AC or DC power input can cover any power supply providing an AC or DC voltage between 7.5 and 12V.



1. USB supply, external supply & and LPC1769 supply

Figure 5. Power supply

As soon as the board is supplied with power, the red LED LD100 must be on.

The PN5180 evaluation board provides two LDOs, one for 5V and one for 3.3V. 5V LDO is only be used, if the external power supply is connected and used (J101 default).

Three jumpers can be used to evaluate the different power supply options:

J101: either external or USB power supply (default)

J303: either VBAT = 5V or 3.3V (default)

J300: closed (default) or to measure the ITVDD

2.2.3 PN5180

The PN5180 is shown in Fig 6.

The default clock is based on a 27.12 MHz crystal, but the board supports external clock input, if needed.

During the antenna tuning and overall hardware design typically the ITVDD must be checked. This can be done with the JP300 (“TVDD”), either using an external power supply or just using an ampere meter instead of the jumper.

The relevant test signals can be derived from the test pins at the bottom of the board.

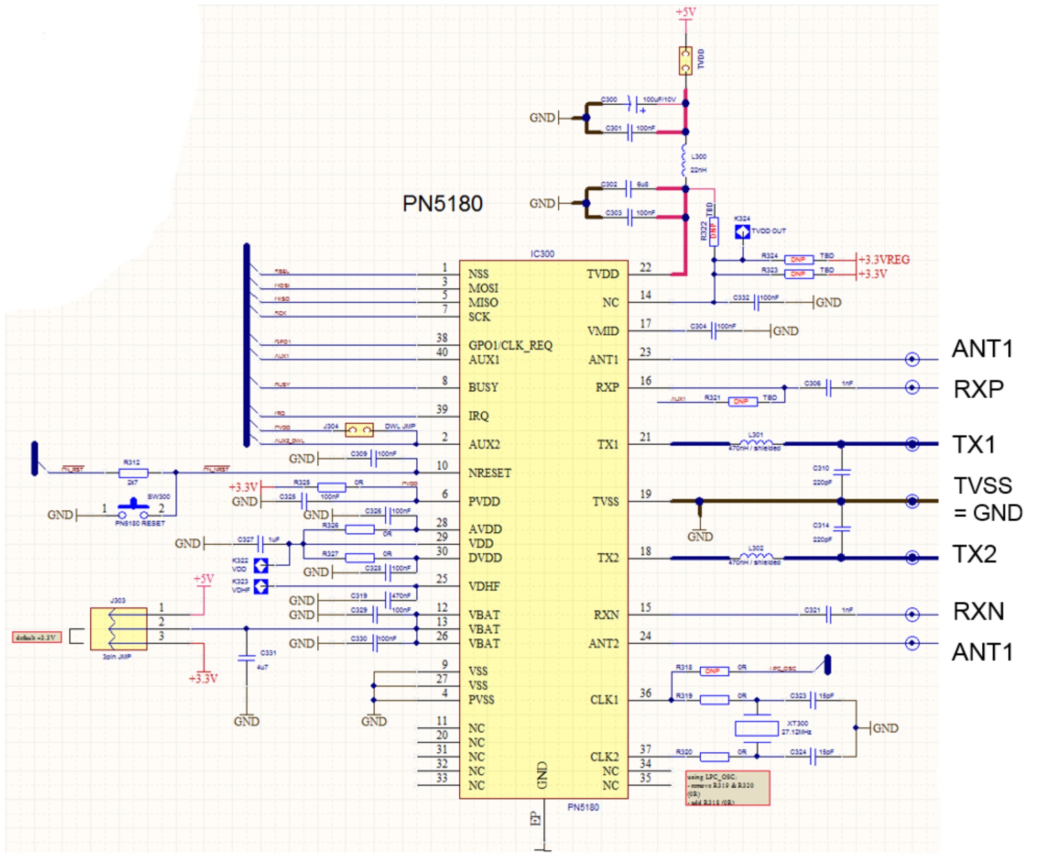
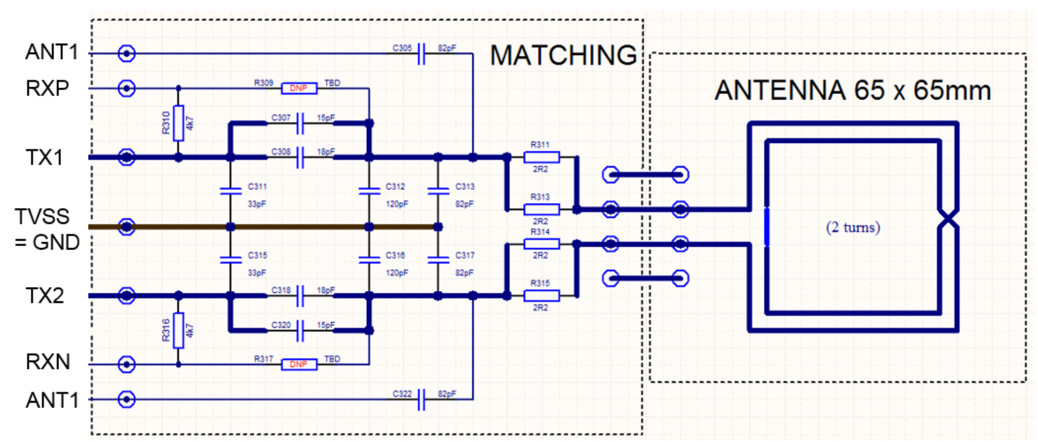


Figure 6. PNEV5180B 2.0 PN5180

The antenna connection uses the standard tuning circuit. The EMC filter is designed with a cut off frequency of f_{EMC} = 15 MHz, and the antenna impedance is tuned to Z ≈ 20Ω.

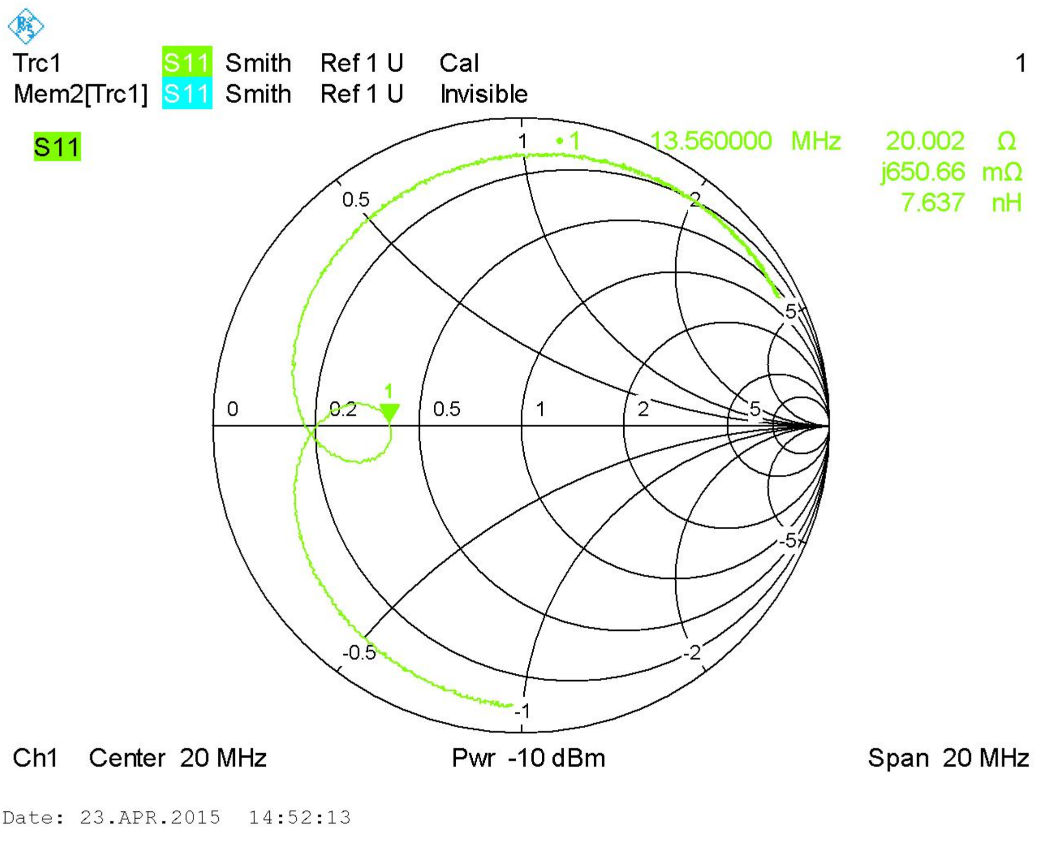


1. The matching components might be adapted due to antenna layout changes.

Figure 7. Antenna

Note: The antenna impedance tuning and measurement must be done with R = 10Ω between ANT1 and ANT2.

The symmetrical tuning (see Fig 8) improves the transfer function compared to the standard “asymmetrical” tuning and thus allows to use a higher system Q factor, which results in a higher field strength. The disadvantage of the loading effect, which causes an increased current ITVDD, is compensated with the PN5180 Dynamic Power Control (DPC, for details refer to [4]).



1. Including 10Ω between ANT1 and ANT2

Figure 8. PN5180 evaluation board antenna tuning

2.3 Jumper settings

Three jumpers can be used to evaluate the different power supply options:

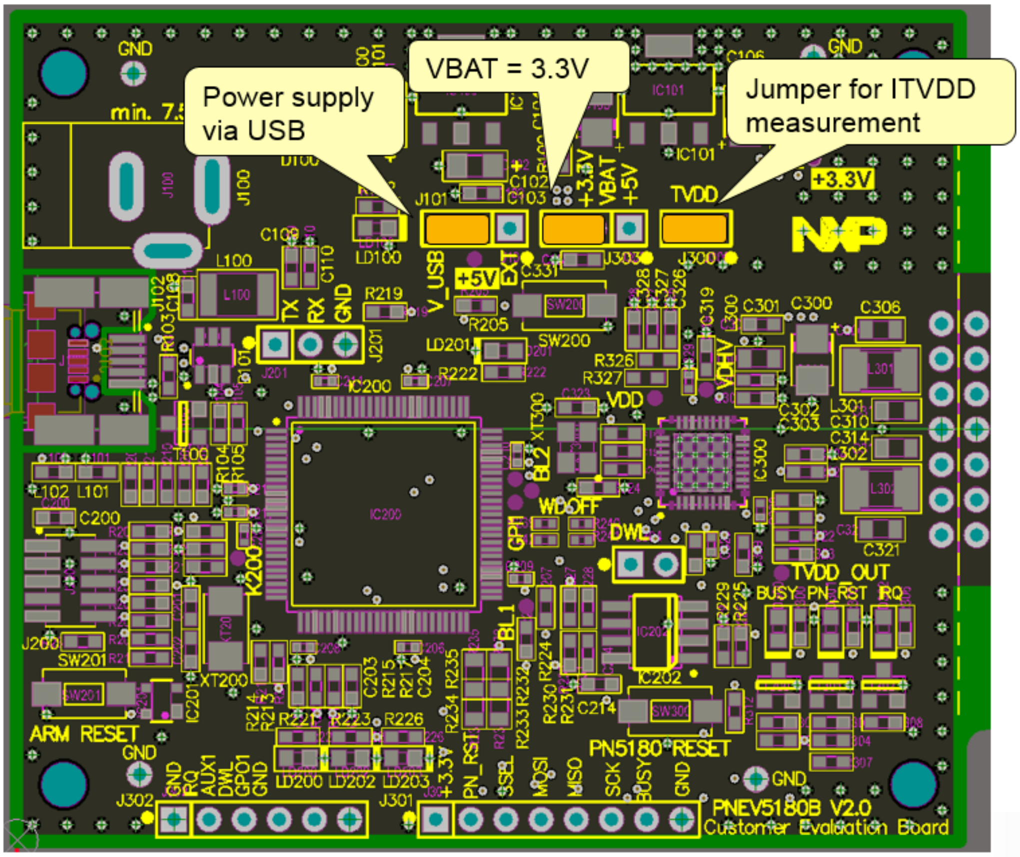
J101: either external or USB power supply (default)

J303: either VBAT = 5V or 3.3V (default)

J300: closed (default) or to measure the ITVDD

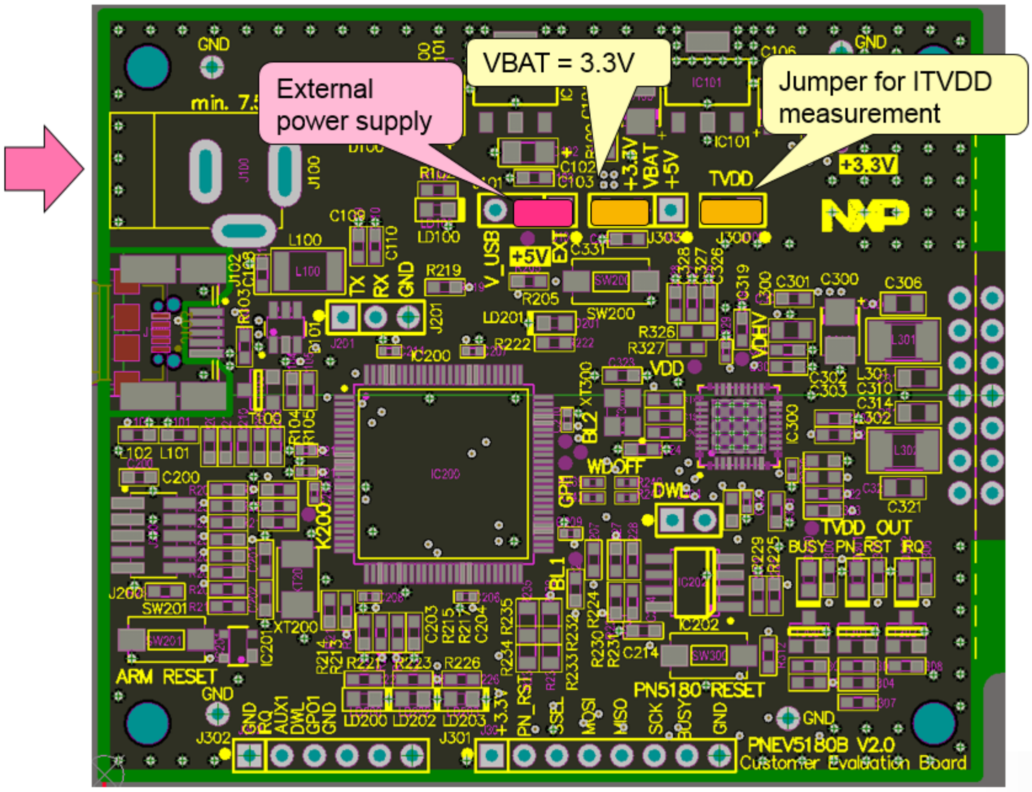
Fig 9 shows the default jumper settings for operation powered via USB.

Fig 10 shows the jumper setting for the operation externally powered.



1. Default settings, power supply via USB

Figure 9. PNEV5180B 2.0 Default jumper settings



1. External power supply: V ≥ 7.5V

Figure 10. PNEV5180B 2.0 jumper settings for external power supply

3 Driver installation

The PNEV5180B 2.0 evaluation board is delivered with a graphical user interface application (GUI), the NFC Cockpit (refer to [6]). The NFC Cockpit can be used to explore the functionality of the PN5180 and perform RF and antenna design related tests. It allows a direct register access as well as EEPROM read and write access, and it allows to test and calibrate the DPC. It offers the major relevant functions to support the antenna tuning in customer designs and optimize the PN5180 settings. The NFC Cockpit therefore can be used to fully configure & test the PN5180.

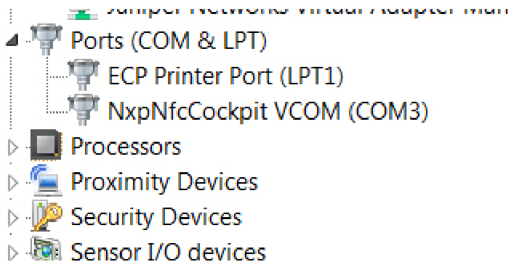
3.1 PNEV5180B PC driver

The PNEV5180B requires a PC driver to be installed. The NFC Cockpit package contains two different PC drivers: The older version of the NFC Cockpit (2.3 or lower) required an ABEND driver (using the ABEND interface). This interface has been replaced with a VCOM interface, so the ABEND driver is not required anymore.

The VCOM driver is automatically installed with the installation of the NFC Cockpit itself. However, it can also be installed by simply starting the c:\nxp \NxpNfcCockpit_v<VERSION>\VCOM\install_vcom.bat

After installation of the VCOM driver, the connected PNEV5180B must show up like shown in Fig 11, if the latest LPC FW is installed on the PNEV5180B.

Note: The (default) PNEV5180B still uses the ABEND interface – and therefore does not show up like shown in Fig 11 in the first-time use! First time users please continue with 3.2.3.



1. VCOM driver installed & PNEV5180B connected: here showing up as COM3

Figure 11. PNEV5180B with new LPC FW connected to the PC

3.2 PNEV5180B LPC Firmware

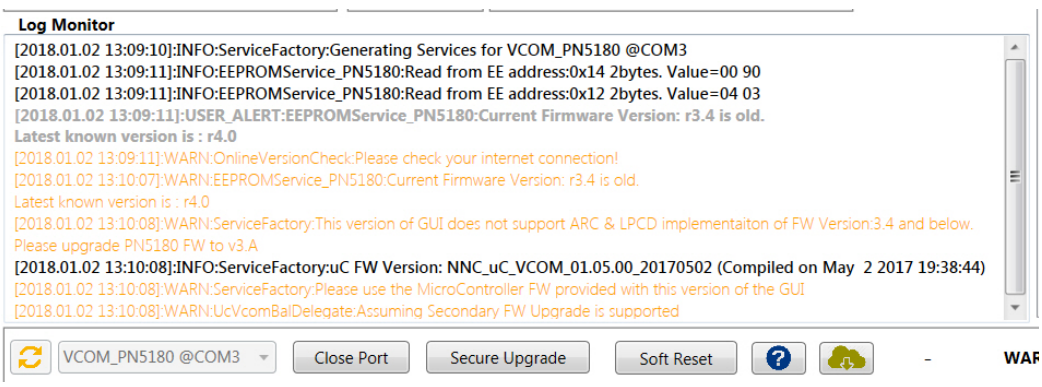
The LPC firmware as installed by default on the PNEV5180B 2.0 is outdated. So, an LPC firmware update might be required.

Note: Do not mix up the LPC 1769 firmware (LPC FW or "Secondary FW") with the PN5180 FW. The LPC FW is required to "connect" the PN5180 to the NFC cockpit, while the PN5180 FW is part of the PN5180 functionality and can only be updated via the secure firmware update procedure.

It is strongly recommended to use the latest NFC Cockpit version (check [6]) in combination with the latest LPC FW. The LPC FW is part of the NFC cockpit installation package.

3.2.1 Legacy 1: Update from previous version, VCOM already installed & used

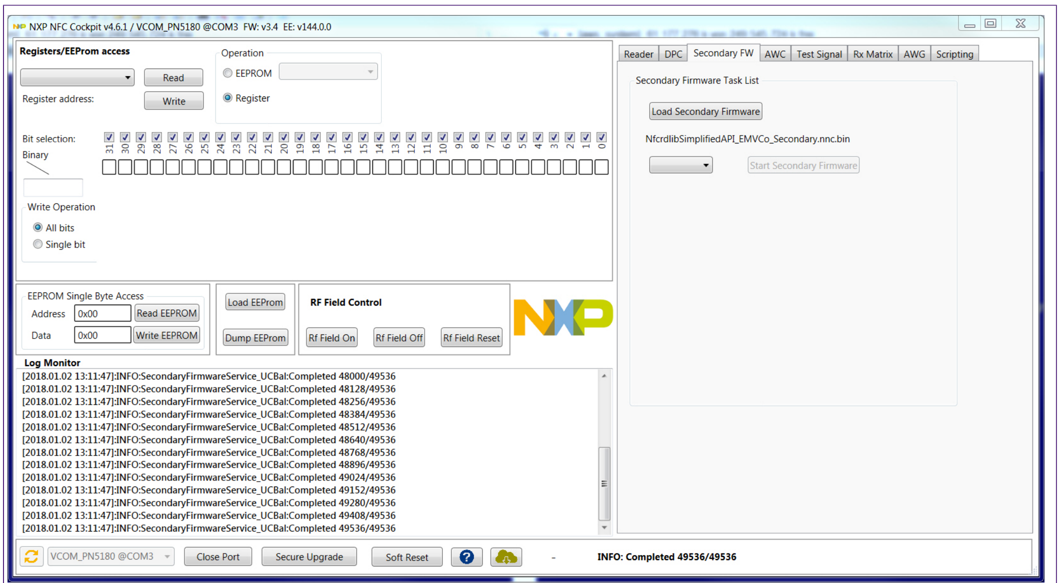
The PN5180B can be connected to the PC, and the NFC Cockpit can be started, if the VCOM drivers had been installed and used before. The NFC Cockpit shows a warning, if LPC FW or PN5180 FW or not up to date, as shown in Fig 12.



1. Example: LPC FW as well as PN5180 FW is not up to date

Figure 12. NFC Cockpit warnings in the Log Monitor

The LPC FW update can be easily done with the <Secondary FW> tab of the NFC cockpit, as shown in Fig 13. The recommended LPC FW is located in: C:\nxp\NxpNfcCockpit_v<VERSION>\firmware\Secondary_PN5180\



1. The secondary FW can be found in C:\nxp\NxpNfcCockpit_v<VERSION>\firmware\Secondary_PN5180\

Figure 13. NFC Cockpit Secondary FW (LPC FW) update

After the LPC FW has been flashed, the connection must be reconnected (Close Port -> Open Port): Follow the NFC Cockpit guidance.

3.2.2 Legacy 2: Update from previous version, ABEND driver installed

The LPC FW update is triggered automatically, when the NFC Cockpit is started the first time, if the correct (old) ABEND drivers are installed. In such a case, the user simply needs to accept the update and follow the NFC Cockpit guidance.

Note: These ABEND drivers are not used anymore for the operation of the NFC Cockpit, except to update the LPC FW to a VCOM version.

3.2.3 First time use: Update from ABEND to VCOM

The LPC firmware as installed by default on the PNEV5180B 2.0 uses the ABEND interface and is outdated. So, an LPC firmware update is required.

There are two options to update the LPC FW:

Option 1: Using the automatic update function of the NFC cockpit. This requires the ABEND drivers to be installed!

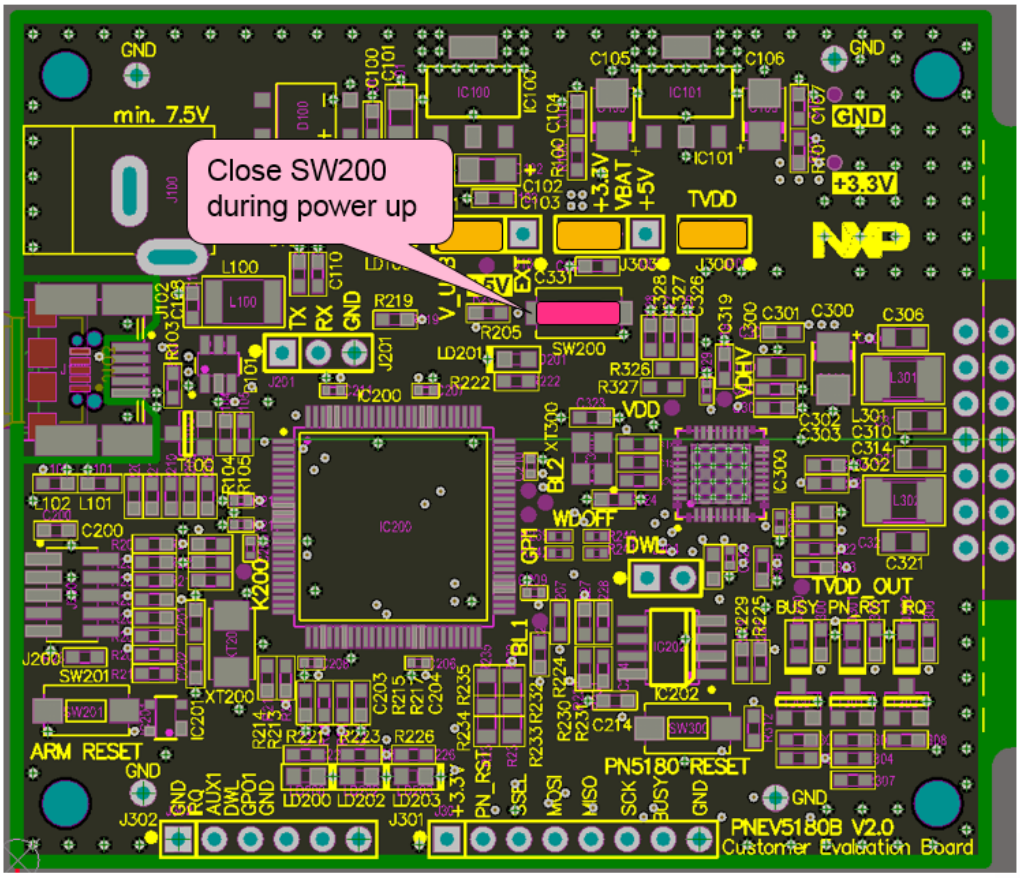
Step 1: Install the ABEND driver (see 7.1).

Step 2: Update the LPC FW with NFC Cockpit (see 3.2.2).

Option 2: Using the mass storage mode of the LPC FW. No ABEND driver needs to be installed. This is the recommended option.

Step 1:

The original (default) LPC firmware provides a mass storage mode, which can be used to install or update the firmware. To activate the boot loader, close the SW200 (button not assembled), while powering up the PNEV5180B, as shown in Fig 14.



1. SW200 to enter “USB mass storage” mode

Figure 14. LPC Bootlader mode

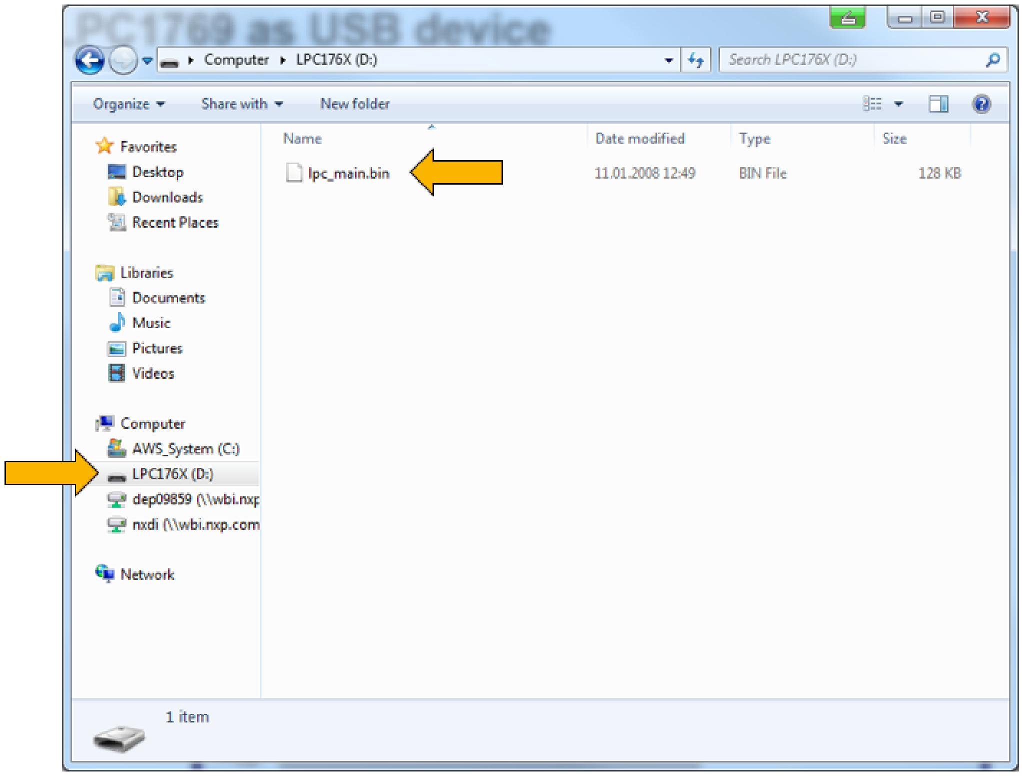
This registers the device as USB mass storage device on the PC as shown in Fig 15. This mass storage device contains one file: the “lpc_main.bin”.

This file needs to be removed (delete), and the new firmware binary file must be copied from

C:\nxp\NxpNfcCockpit_v<version>\ABENDToVCOM\fw\lpc_main.bin

into the storage device.

As soon as the upload is done, the folder closes and the USB storage device is automatically disconnected from the PC.



1. After reboot with SW200 shorted.

Figure 15. LPC 1769 as USB device (from default LPC FW)

The LPC on the PNEV5180 will automatically reboot and then the PNEV5180B is ready to be used with the NFC Cockpit.

3.2.4 Re-installing LPC FW after IDE use

The LPC1769 might be used for software development together with one of the samples (including the NFC Reader Library, see [7]). In such case the LPC FW must be re-installed afterwards, if the PNEV5180B is supposed to be used together with the NFC Cockpit again. Reason for this is that any software development using the LPCXpresso will erase the default firmware.

Therefore, the LPC FW needs to be re-flashed:

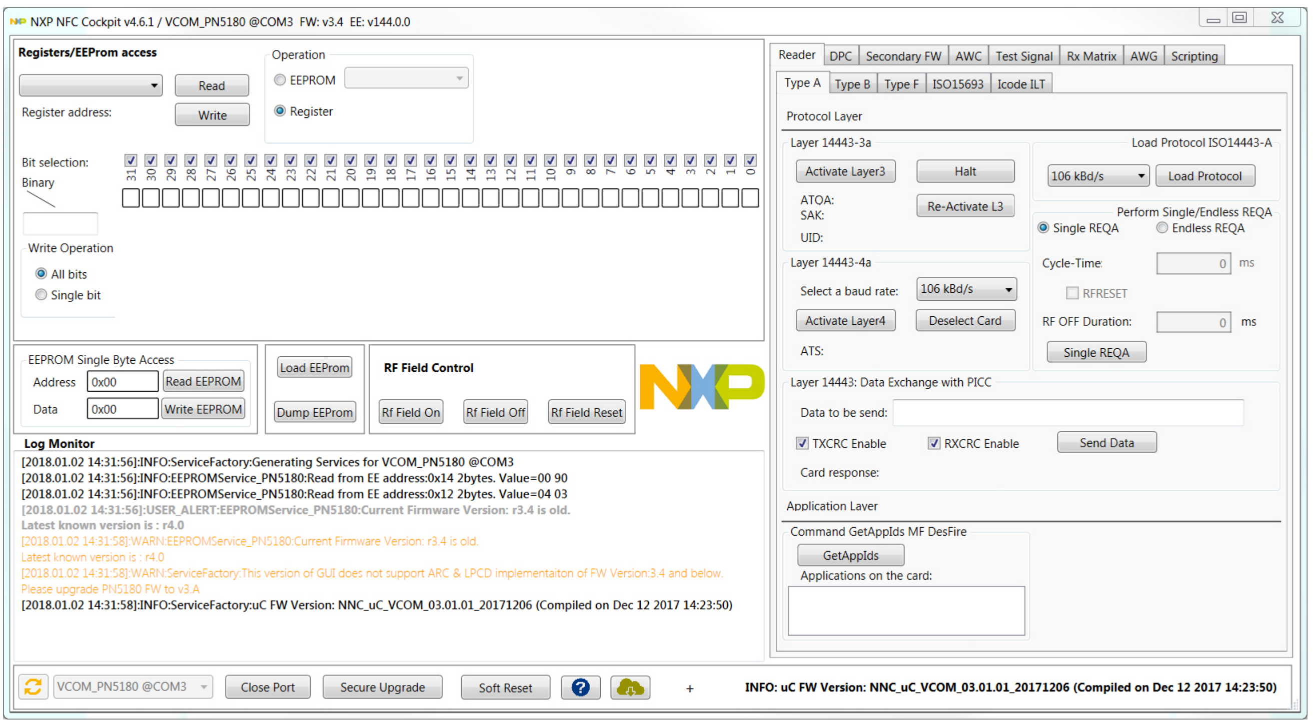
c:\nxp\NxpNfcCockpit_v<VERSION>\firmware
\Secondary_PN5180\BootLoader_And_Nfcrdlib_SimplifiedAPI_EMVCo_Secondary.bin

Binary file that has already stacked Boot loader from address 0x0000 onwards and Upper Secondary Application at address 0x4000. So, this file can be used to flash on LPC1769 as a single binary from address 0x0000 onwards using relevant Debuggers/ Flash Programmers.

In any case the correct PC driver must be installed, before the NFC Cockpit can be used with the PNEV5180B 2.0 evaluation board.

4 PN5180 NFC Cockpit

The PN5180 NFC Cockpit can be installed and started (see Fig 16).



1. Status when starting the NFC Cockpit with connected PNEV5180B board, but older PN5180 FW is used.

Figure 16. NFC Cockpit Initial view

After starting the NFC Cockpit, the communication link between the PC and the PNEV5180B (via the LPC interface) is enabled automatically.

Note: The PN5180 NFC Cockpit is a development tool, and therefore allows many different kind of operations, even “useless” ones at a first glance. The correct use of the NFC Cockpit is required to operate the PN5180 properly.

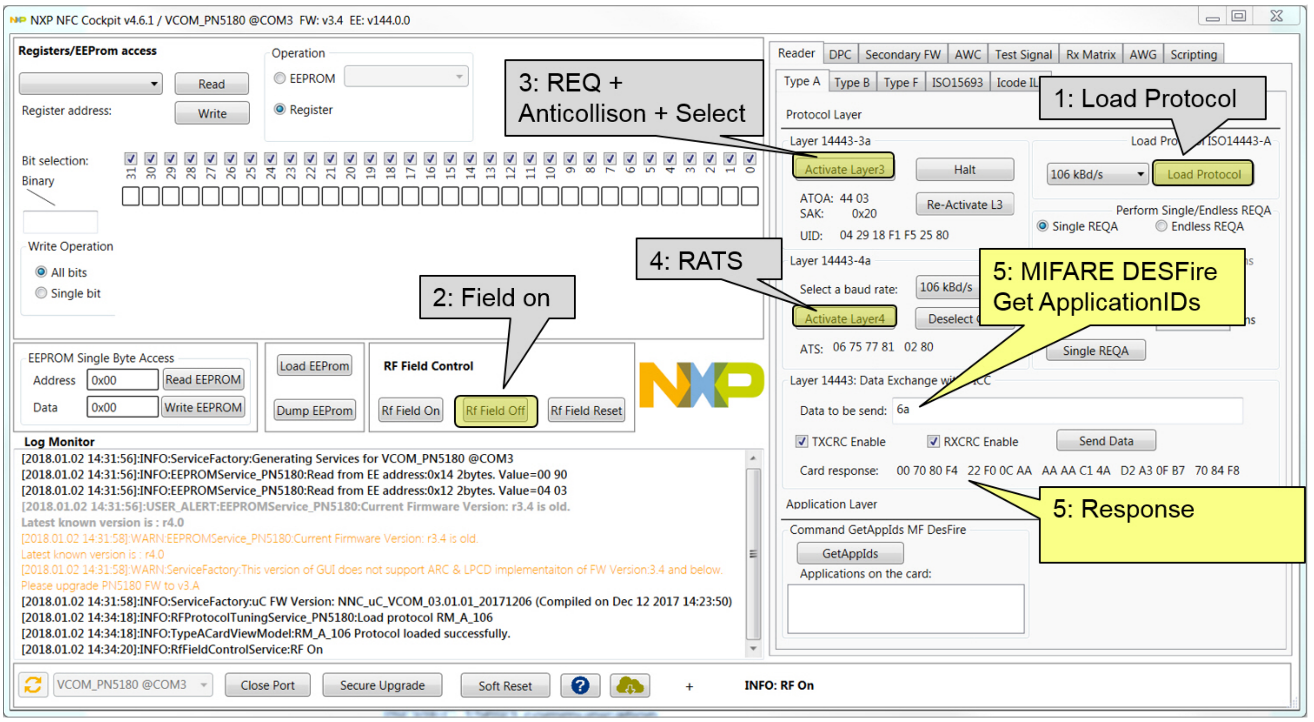
Example: without enabling the Field no card can be operated, even though the PN5180 can be operated.

4.1 NFC Cockpit Reader tab

The Fig 17 shows the activation of a MIFARE DESFire card, using the <Load Protocol> + <Field On> + <Activate Layer3>, followed by <Activate Layer4>. The PN5180 NFC Cockpit shows the card responses like ATQA, SAK, and ATS.

Afterwards the ISO/IEC 14443-4 protocol can be used to exchange data. The Fig 17 shows the MIFARE DESFire command “Get Application ID” (0x6A), which returns the AIDs.

Note: Make sure that either the CRC is enabled or added manually in the data field.



1. 0x6a = Get Application ID command of MIFARE DESFire EV1

Figure 17. PN5180 NFC Cockpit: Activation of a MIFARE DESFire EV1 card + Get Application ID

Similar functionality does exist for ISO/IEC 14443 A and B, for NFC type F, for ISO/IEC 15693 and I-Code ILT communication.

Be aware that a LOAD_RF_CONFIG command must be executed manually before the corresponding protocol settings are loaded from the EEPROM into the registers. This can be used to perform

1. <Load Protocol> (e.g. type A 106)
2. <Field On>
3. <Single REQA> (using the EEPROM settings)
4. Select a TX register, e.g. RF_CONTROL_TX, enable TX_SET_BYPASS_SC_SHAPING
5. Change some register bits, and write back into RAM
6. <Single REQA> shows the register changes (probing the field and checking the envelop)

This allows an easy and quick optimization of Tx and Rx parameters before changing the EERPOM.

1. <Load Protocol> (e.g. type A 106)
2. <Single REQA> (using again the EEPROM settings)

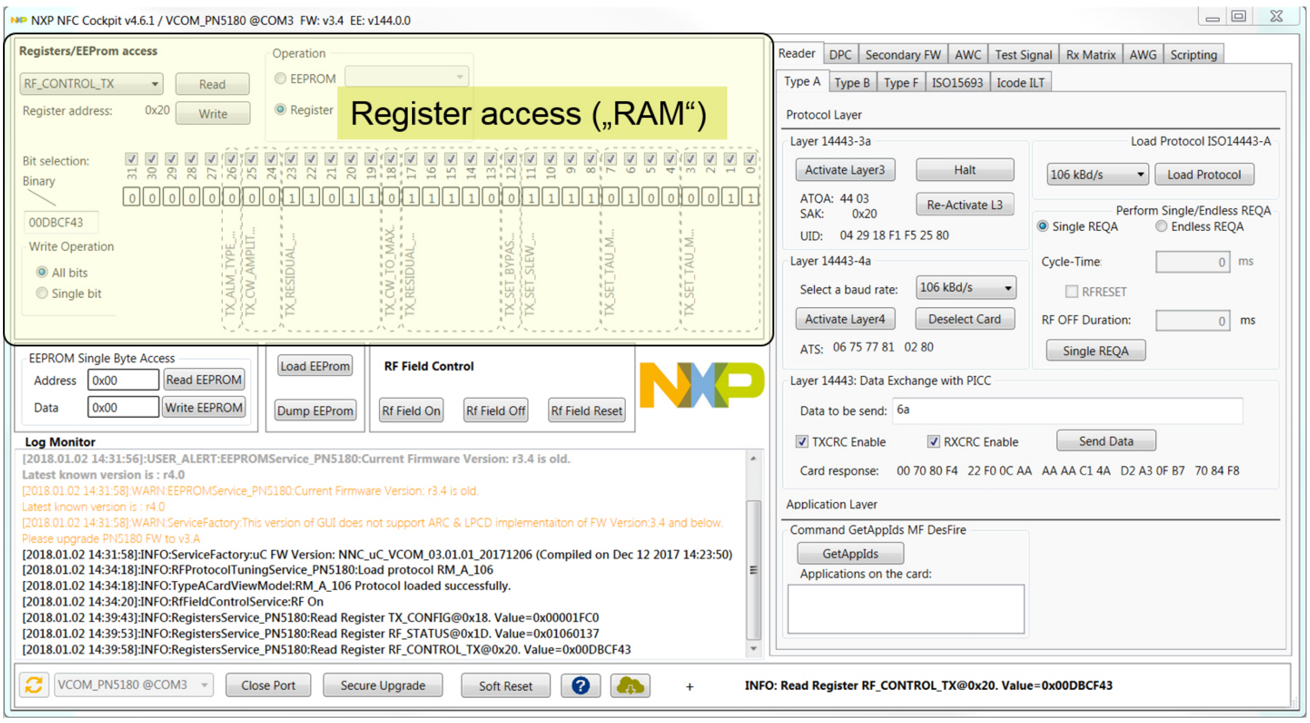
4.2 PN5180 Register access

The PN5180 NFC Cockpit allows the reading and writing of all the PN5180 registers (see Fig 18).

Selecting a register reads and shows the hexadecimal content as well as the corresponding bit values. The input allows to change each bit separately as well as writing hexadecimal values. Writing back the value changes the PN5180 register.

A help function automatically shows a brief description of the (part of the) registers itself, if the mouse is moved over the names.

Note: Some register content cannot be changed manually ("read only") and some content might be overwritten by the PN5180 firmware.



1. Register area is a RAM area, i.e. might be overwritten or changed automatically.

Figure 18. PN5180 register access

4.3 PN5180 EEPROM access

The NFC Cockpit allows different options of EEPROM access (see Fig 19):

EEPROM area for RF_CONFIG

All registers, which are used in the LOAD_RF_CONFIG command, can be read directly from the EEPROM: The user must select the "register" and the protocol.

All registers, which are used in the LOAD_RF_CONFIG command, can be directly written into the EEPROM. The user must select the "register" and the protocol.

This allows an easy EEPROM update of the relevant TX and Rx registers after optimization in RAM.

EEPROM area 0x00 … 0xFC

- Read EEPROM

Reads a single byte from EEPROM using byte address from 0x00 to 0xFC

- Write EEPROM

Writes a single byte into EEPROM using byte address from 0x00 to 0xFC

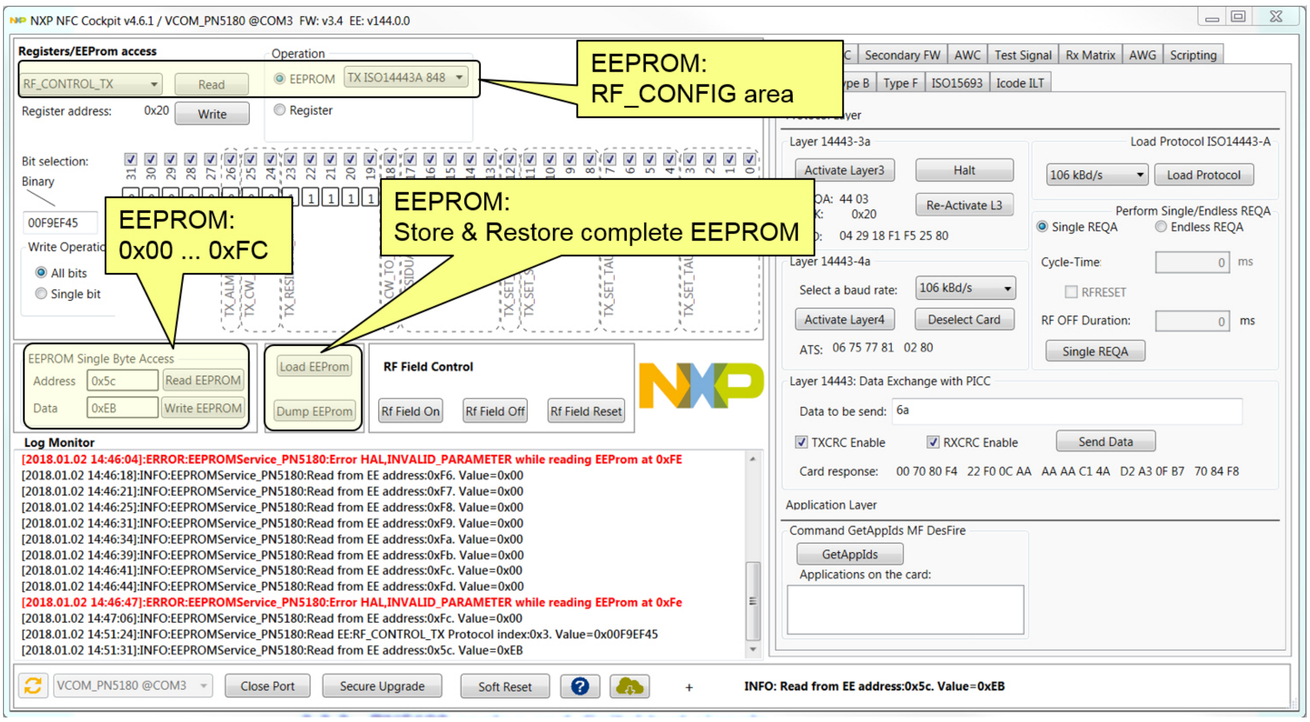
Store and restore the complete EEPROM

- Dump EEPROM

Stores the complete “user area” of the PN5180 EEPROM into a XML file. This can be used to generate a backup of all settings or to transfer optimized settings onto another board or into own software. This includes the RF_CONFIG area.

- Load EEPROM

Load a XML file and stores the content into the user area of the PN5180 EEPROM. The format is fixed and must fit.

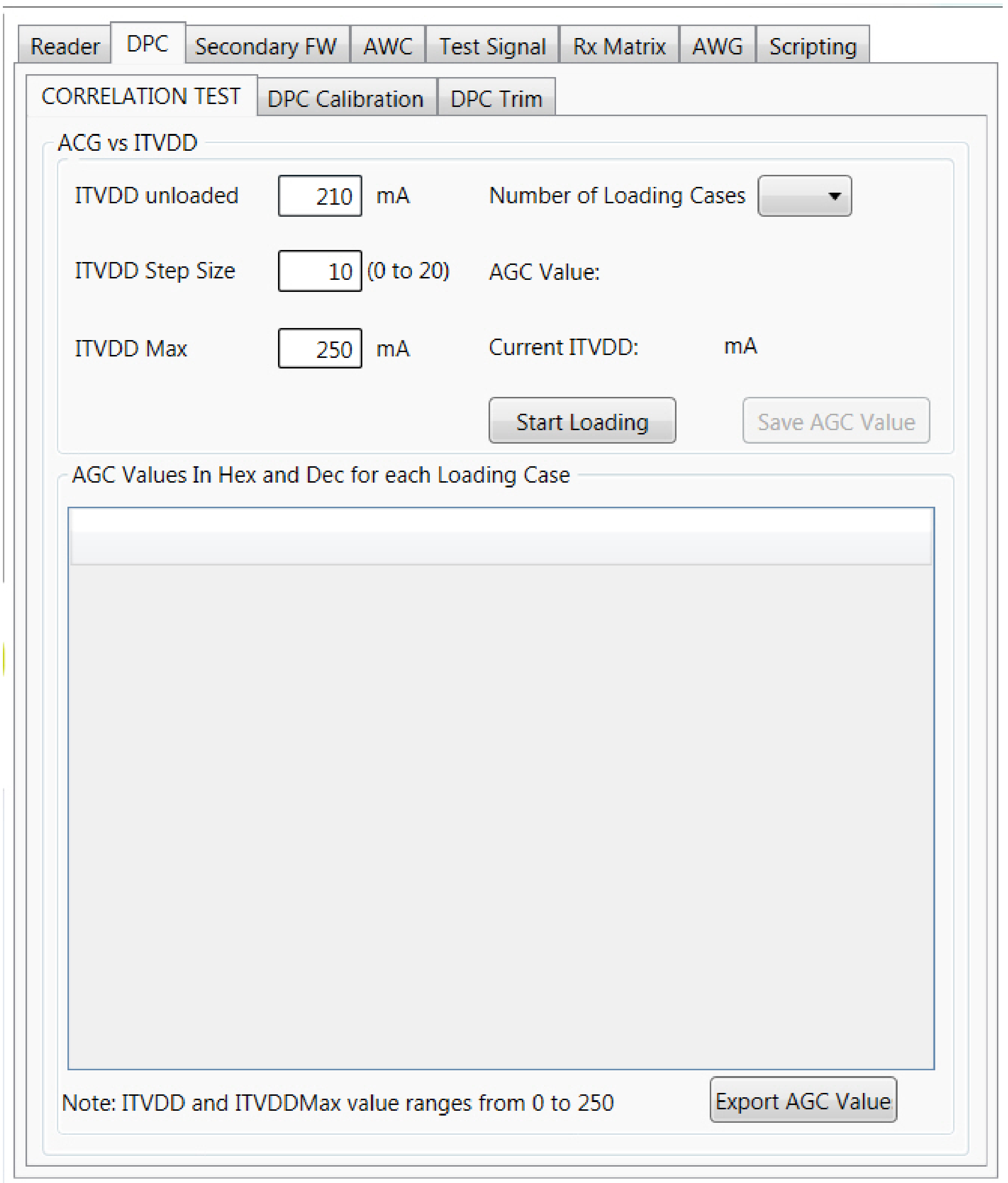


1. EEPROM means User area of the EEPROM, details refer to [1]

Figure 19. PN5180 direct EEPROM access

4.4 PN5180 Dynamic Power Control

The DPC tab provides the functionality to easily perform a correlation test, a DPC calibration and the DPC trimming (see Fig 20). The detailed functionality is described in [4], but also the video tutorials might be a good help (refer to [8]).



1. For more details refer to [4].

Figure 20. PN5180 DPC

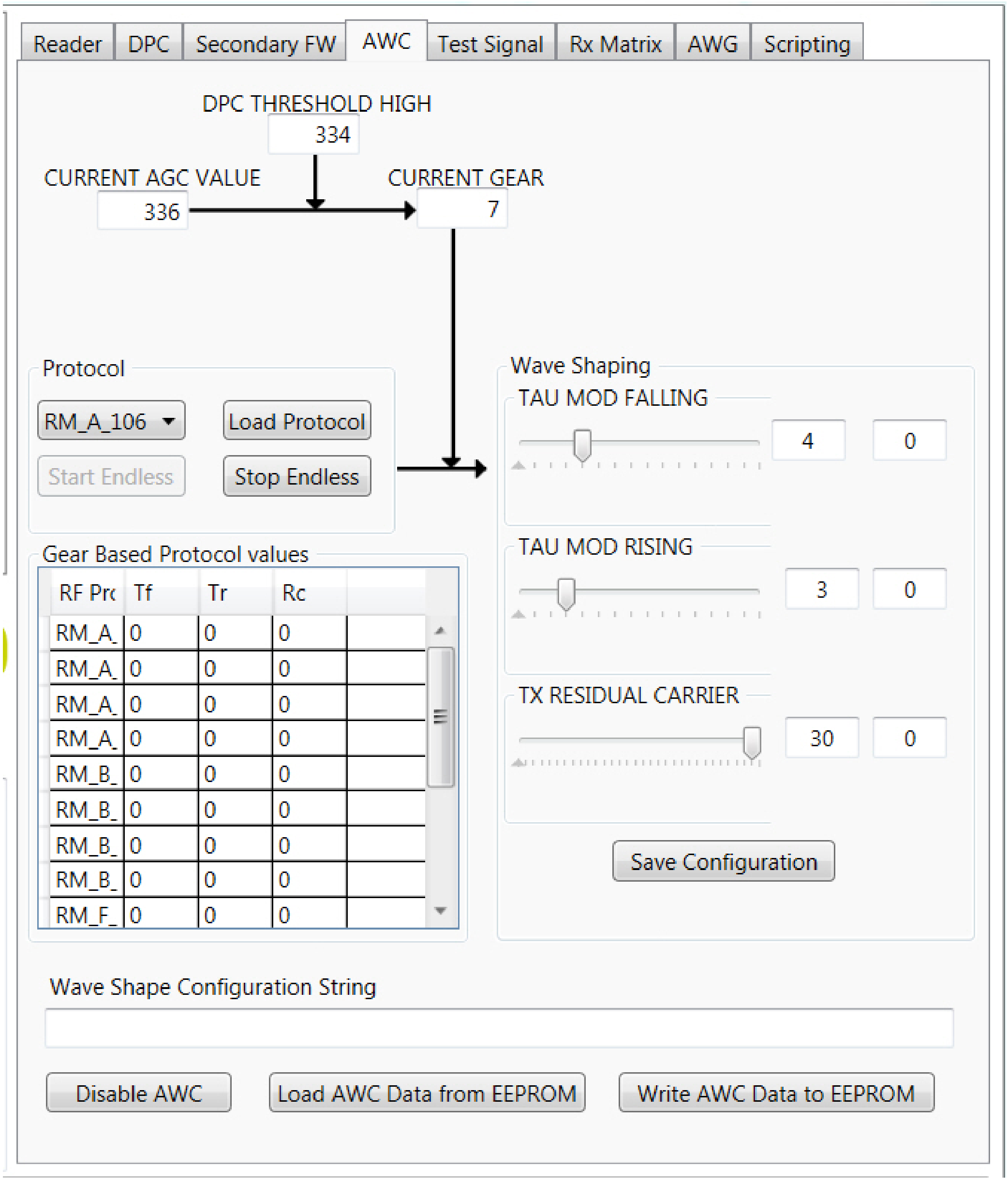
4.5 PN5180 Adaptive Wave Control

The PN5180 DPC functionality offers the option to use a lookup table to dynamically control the TX shaping. This feature is called Adaptive Wave Control (AWC). The NFC Cockpit provides a AWC functionality to allow an easy optimization of the shaping functions (see Fig 21).

Requirement: a properly tuned antenna is connected and the DPC is calibrated.

Note: It is recommended to disable the AWC, before starting the AWC function in the NFC Cockpit to avoid confusion: the AWC itself is done inside the PN5180 FW, and the NFC cockpit tries to overrule that in the AWC tab. Disabling the AWC changes the EEPROM.

Note: It is recommended to store the complete EEPROM content (backup!) using the <Dump EEPROM> before disabling the AWC. This allows an easy recovery at any time, even if the EEPROM is messed up.



1. Example: Running in an endless loop for Type A @ 106 settings

Figure 21. PN5180 AWC

4.5.1 Proposal for “static” Tx shaping adjustment

Step1: Save EEPROM for backup (<Dump EEPROM>), then disable the AWC (<Disable AWC>).

Step2: Operate the antenna in gear 0 (“unloaded”): <Load Protocol> (for e.g. Type A 106) and enable RF field (<RF Field On>).

Step 3: <Start Endless> and watch the current gear: must be 0!

Step 4: Check the pulse shape with a Reference PICC and an oscilloscope. Move the sliders <TAU MOD FALLING>, <TAU MOD RISING> and <TX RESIDUAL CARRIER> to optimize the shaping.

Step 5: Note down the optimum settings and save the corresponding register settings into the EEPROM (Read RF_CONTROL_TX register and write the value back into the required EEPROM (e.g. TX ISO14443A 106).

Step 6: <Stop Endless>

Step 7: <Load Protocol> (with the same protocol, e.g. Type A 106) and then send single or endless REQA. Check the wave shape in gear 0 position.

4.5.2 Proposal for “dynamic” Tx shaping adjustment

Requirement: “static” TX shaping adjustment is done properly.

Step1: Save EEPROM for backup (<Dump EEPROM>), then disable the AWC (<Disable AWC>), if not done before.

Step2: Start in gear 0 (“unloaded”): <Load Protocol> (for e.g. Type A 106) and enable RF field (<RF Field On>).

Step 3: <Start Endless> and watch the current gear: must be 0!

Step 4: Check the pulse shape with a Reference PICC and an oscilloscope: Must be ok.

Step 5: Load the antenna, until the gear changes to the next higher one. Move the sliders <TAU MOD FALLING>, <TAU MOD RISING> and <TX RESIDUAL CARRIER> to optimize the shaping.

Step 6: <Safe Configuration> -> This stores the AWC settings into the NFC Cockpit table. The PN5180 EEPROM is not changed at all.

Step 7: Continue with Step 5, until the last gear is reached.

Step 8: <Stop Endless>

Step 9: <Write AWC Data to EEPROM> -> This writes the new AWC data into the look up table in the PN5180 EEPROM.

Step 10: <Load Protocol> (with the same protocol, e.g. Type A 106) and then send single or endless REQA. Check the wave shape in all gear positions.

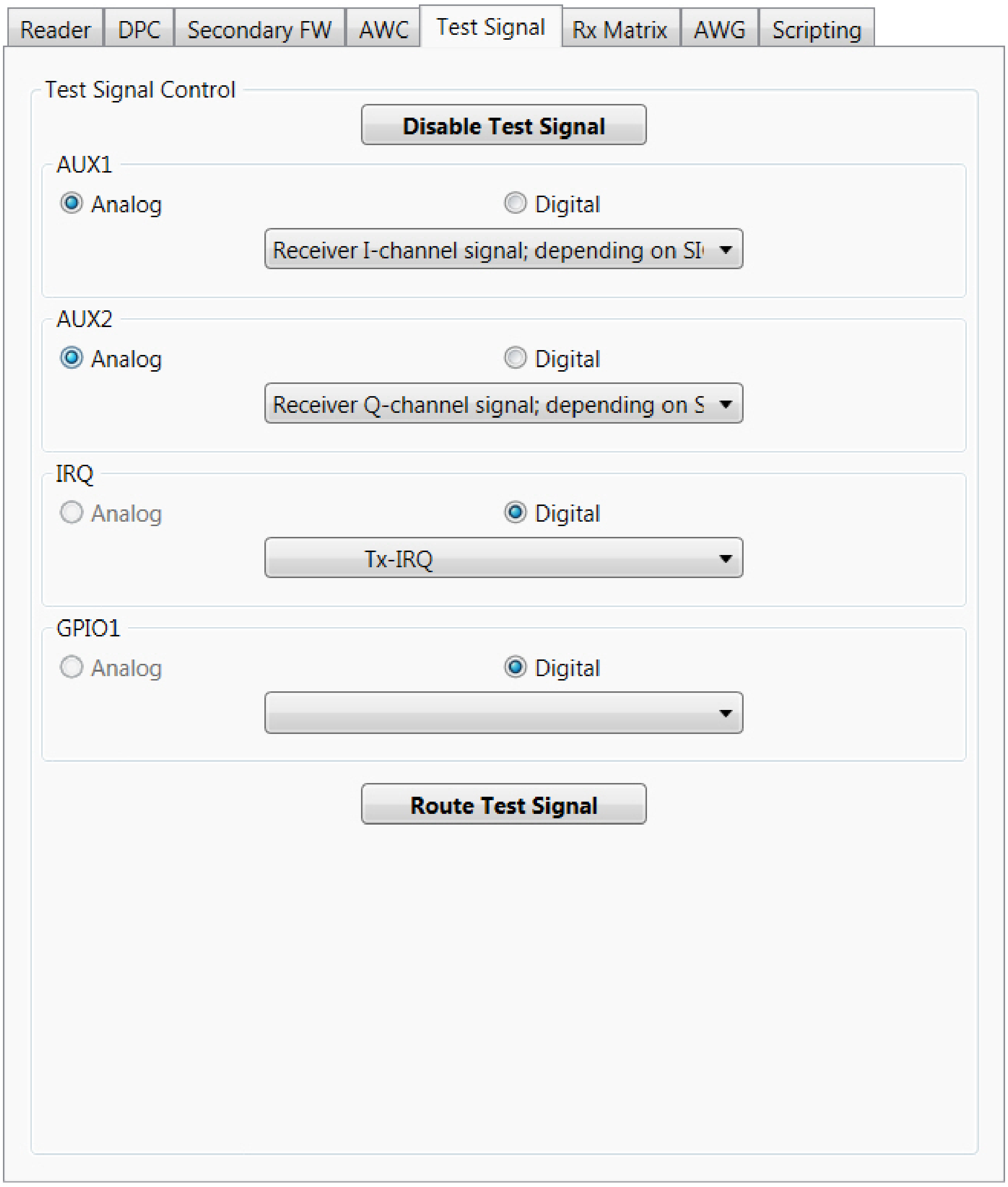
4.6 PN5180 analog and digital test signals

The NFC cockpit allows to route analog and digital test signals to test pins of the PN5180, if the PN5180 internal test bus is enabled (see Fig 22).

Enabling the test bus changes the EEPROM adr. 0x17 (refer to TESTBUS_ENABLE, see [1]).

Enabling the test bus changes the communication between PN5180 and LPC, since the IRQ pin is used for test signals, and therefore cannot be used as IRQ pin anymore. Therefore, the communication changes into a polling mode: This automatically disables the LPCD functionality.

The test pins can be found at J302 (pin row).



1. Enabling/disabling test bus is stored in the EEPROM

Figure 22. PN5180 analog and digital test signals

The analog test signals can directly be selected, for the digital signals a test bus group must be chosen first. Two digital test signals can only be selected from one test bus group at the same time.

4.7 PN5180 Rx Matrix test

The receiver settings of the PN5180 normally need to be optimized to achieve the best performance. This optimization can be done manually, using the test signals. However, this manual optimization can be cumbersome, since on one hand some of the register settings depend on each other, so it is almost impossible to derive a deterministic adjustment. On the other enabling the test bus slightly changes the Rx performance, so the behavior with enabled or disabled test but can be different.

Therefore, it is recommended to use a Matrix test, which simply tests all relevant combination of register settings. The result matrix shows easily the optimum settings. This Matrix test is provided in the NFC cockpit.

4.7.1 Rx parameters

Typically, 3 or 4 (or even more) receiver settings need to be optimized for each protocol and antenna design:

- RxGain: 0 ... 3
- HPCF: 0 ... 3
- MinLevel: 0 ... 15
- MinLevelP: 0 ... 15 (only BPSK)

Even though the default values (as delivered with the PNEV5180B) can be taken as reference and starting point, the optimum might be different from the default. Changing one parameter might require another parameter to change, too. At the end even several “optima” might occur, which show similar performance.

There might be external influence like noise (e.g. from an LCD or other electronic circuitry) resulting in a different optimum.

So, it is very difficult to define a clear and deterministic approach to optimize these Rx settings, especially without knowing the external influence. However, for a high-end reader design these settings play a significant role for a good performance.

An easy solution is the Rx Matrix Test. This tool simply tries each combination of settings, and reports the number of proper receptions.

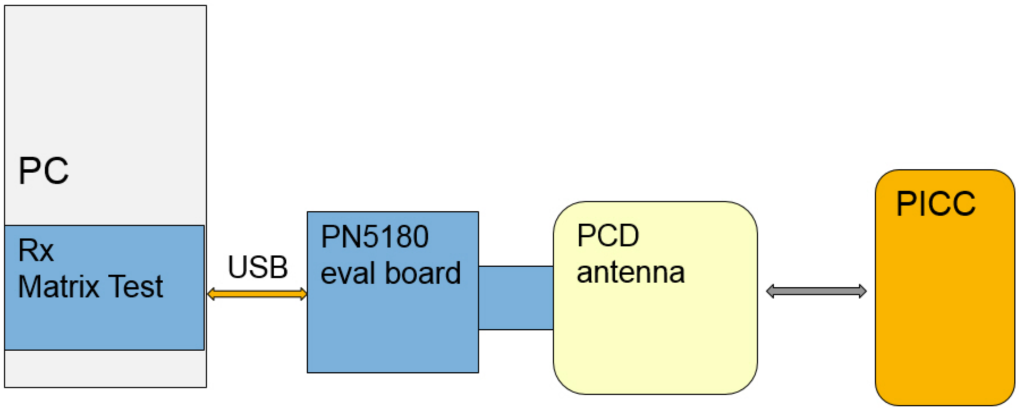
4.7.2 Rx Matrix test principle

The Rx matrix controls the PNEV5180B evaluation board and allows to configure:

- Free number of trials (per register combination)
- Free number and combination of register bits
- Free limit of minimum and maximum value
- Free choice of “protocol” (RF Configuration)
- Optional voltage level control of the LMA, using a Keysight AWG

The tool supports the digital and analog test signals, if needed. Especially the digital test signal might be helpful to trigger a LMA test setup.

The Fig 23 shows the basic test setup, using a reference card, placed on the PCD antenna in a certain distance. The reference card might a typical type A or B card (or any other card that is supported by the PN5180).

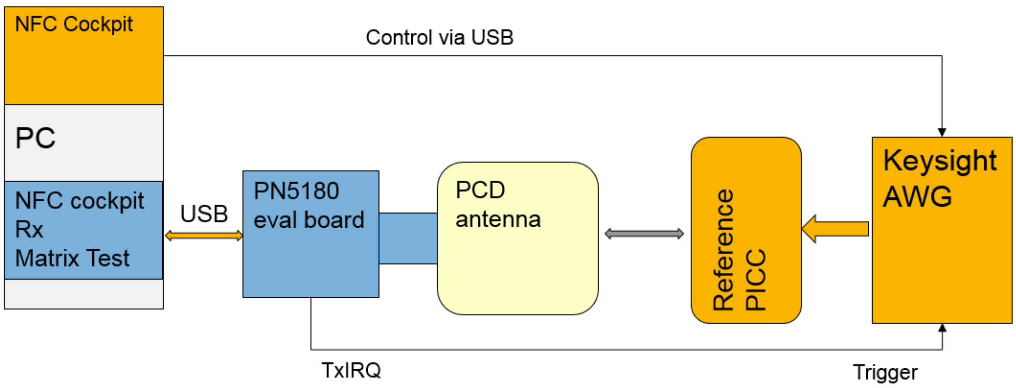


1. The PICC can be any type A or B card in principle, depending on the used protocol.

Figure 23. Basic Rx Matrix Test set up

Of course, the real smart card does not allow to vary the load modulation level, which helps to find the optimum (sensitivity). So, an extended test setup as shown in Fig 24 can be used to control the LMA voltage level. This setup contains

- Reference PICC (ISO, EMVCo or NFC)
- Keysight Arbitrary wave generator (AWG, see [9])
- NFC Cockpit with PNEV5180B



1. Keysight AWG refer to

Figure 24. Enhanced Rx Matrix Test setup with AWG

The input parameters for the test matrix run are defined in an XML file (see 4.7.3.1). The test can be started in the NFC Cockpit (<Start RxMatrix>). The test result is stored as table, when the test is finished. The table can be opened with e.g. Microsoft Excel for interpretation (<View Output>).

4.7.3 Rx Matrix XML input file

The Rx Matrix Test requires the input configuration in an XML file. A few example XML files (for type A, B, F, 15693 and I-Code ILT) are part of the NFC Cockpit package:

c:\nxp\NxpNfcCockpit_v<VERSION>\cfg\RxMatrix\RxMatrix_PN5180\

Refer also to 6 for one example for type A without AWG and one example for type B with AWG.

Such an XML file defines all test parameters. The user can create (copy & paste) own XML files, chose any from the existing XML files and then start the test.

4.7.3.1 Input parameters

Table 1. Test input

Parameter	Meaning	Example value
numberMaxOfPasses	How many trials per combination	10
skipAfterFailures	Continue with the next combination if more failures occur than defined	4
delayMS	Additional delay between trials, if needed	0
fieldReset	Enable RF Reset, if needed (e.g. for type A card)	YES
protocolType	Defines the used protocol and bit rate	RM_A_106

Table 2. Send Data input

Parameter	Meaning	Example value
shortFrame	Enables a short frame (e.g. for REQA)	YES
rxCRC	Enables the CRC check for the card response	NO
txCRC	Enables the CRC on the TX data	NO
timeOutInUs	Defines the time out of the test command in µs	1000
Byte(s) to be sent	These bytes are sent.	0x26

Table 3. Read Data input

Parameter	Meaning	Example value
invertedMaskBytes	Allows to mask certain bytes (for e.g. the PUPI) in the card response	0x00, 0x00
Bytes to be received	These bytes are checked as Rx data	0x44, 0x03

Optional, if AWG is connected:

<VoltageLevel minValueInmV="100" maxValueInmV="2000" stepSizeInmV="500"/>

Defines the LMA voltage level of the AWG from minimum to maximum value with given step size.

4.7.3.2 Permutation settings

For the permutation test the registers and fields must be defined. The RxMatrix test needs to know the register (by name or address), the field, which has to be permutated (by field name or bit position and length), and it has to know the range or values, which shall be used for the permutation.

For the register and field definition in the input file, two options are available:

a) Register by name

The registers can be defined by name and field. The names of all available registers and the related fields can be found in a YAML file provided with the installer:

C:\nxp\NxpNfcCockpit_v<VERSION>\cfg\PN5180\RegisterList_PN5180.yaml

The field name appears in the output file.

Example register by name:

```
<Parameter register="RF_CONTROL_RX" field="RX_GAIN" minValue="0x01"
maxValue="0x03" />
```

The test is done with the RX_GAIN from the RF_CONTROL_RX register. The result file will contain a column, named “RX_GAIN”.

b) Register by range

The registers can be defined by address, bit position, and bit length. The address defines the register address. The bit position defines the lowest bit of the field, and the bit length defines the available permutation range of the field. In addition a nickname has to be defined, which appears in the output file.

Example register by range:

```
<Parameter name="Rx Gain" minValue="0x01" maxValue="0x03"
registerAddress="0x22" bitPosition="0" bitLength="2" />
```

The test is done with the RX_GAIN from the RF_CONTROL_RX register. The result file will contain a column, named “Rx Gain”.

The permutation itself must be defined with either a minimum and a maximum value. In this case all values from minimum until maximum are tested.

Example with min and max value:

```
<Parameter name="Rx Gain" minValue="0x01" maxValue="0x03"
registerAddress="0x22" bitPosition="0" bitLength="2" />
```

This test is done, testing all RX_GAIN values from 1 up to 3 (but not 0).

Or the permutation is defined with specific values, which then are tested.

Example with specific values:

```
<Parameter name="Rx Gain" registerAddress="0x22" bitPosition="0" bitLength="2"
values="0x0, 0x03" />
```

This test is done, testing only the value 0 and 3 (but neither 1 nor 2).

Example scripts refer to 6 and find under

c:\nxp\NxpNfcCockpit_v<VERSION>\cfg\RxMatrix\RxMatrix_PN5180\

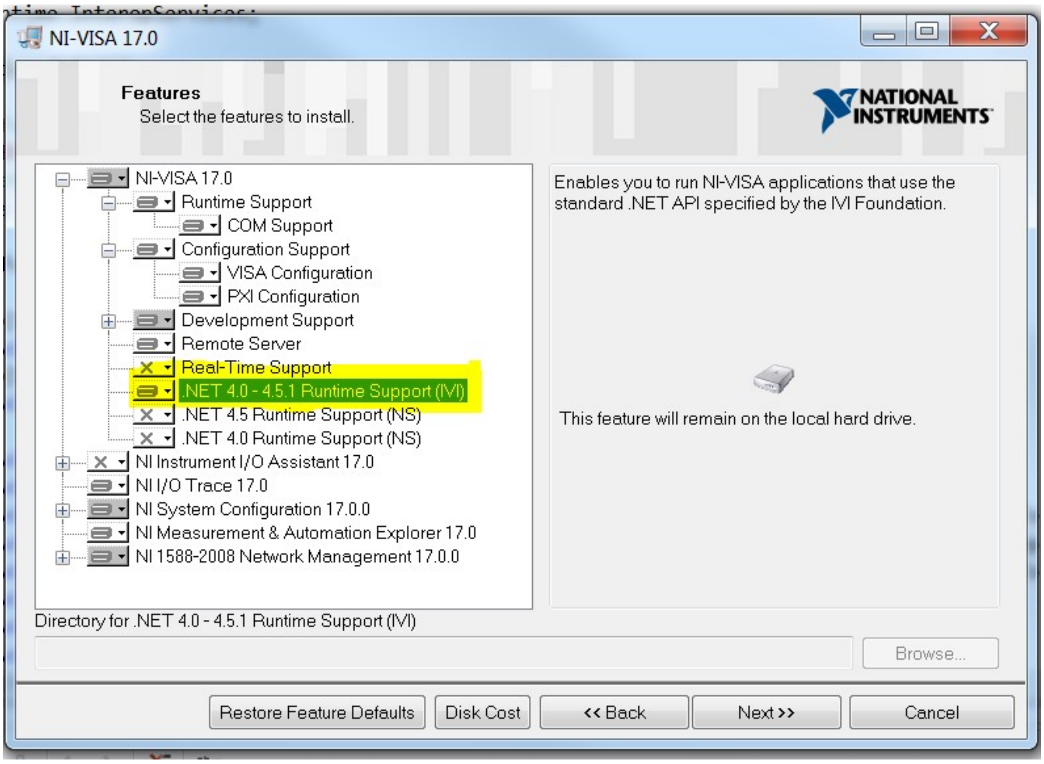
4.8 NFC Cockpit with AWG

4.8.1 NI VISA installation

The NFC Cockpit supports the control of a Keysight AWG (see [9]) via USB. As a prerequisite, the USB driver and National Instruments VISA driver package (refer to [10]) have to be installed.

Remark: This NI VISA version does not conflict with the CTC Advanced WavePlayer tool.

Make sure that the .NET development support is installed, as shown in Fig 25.

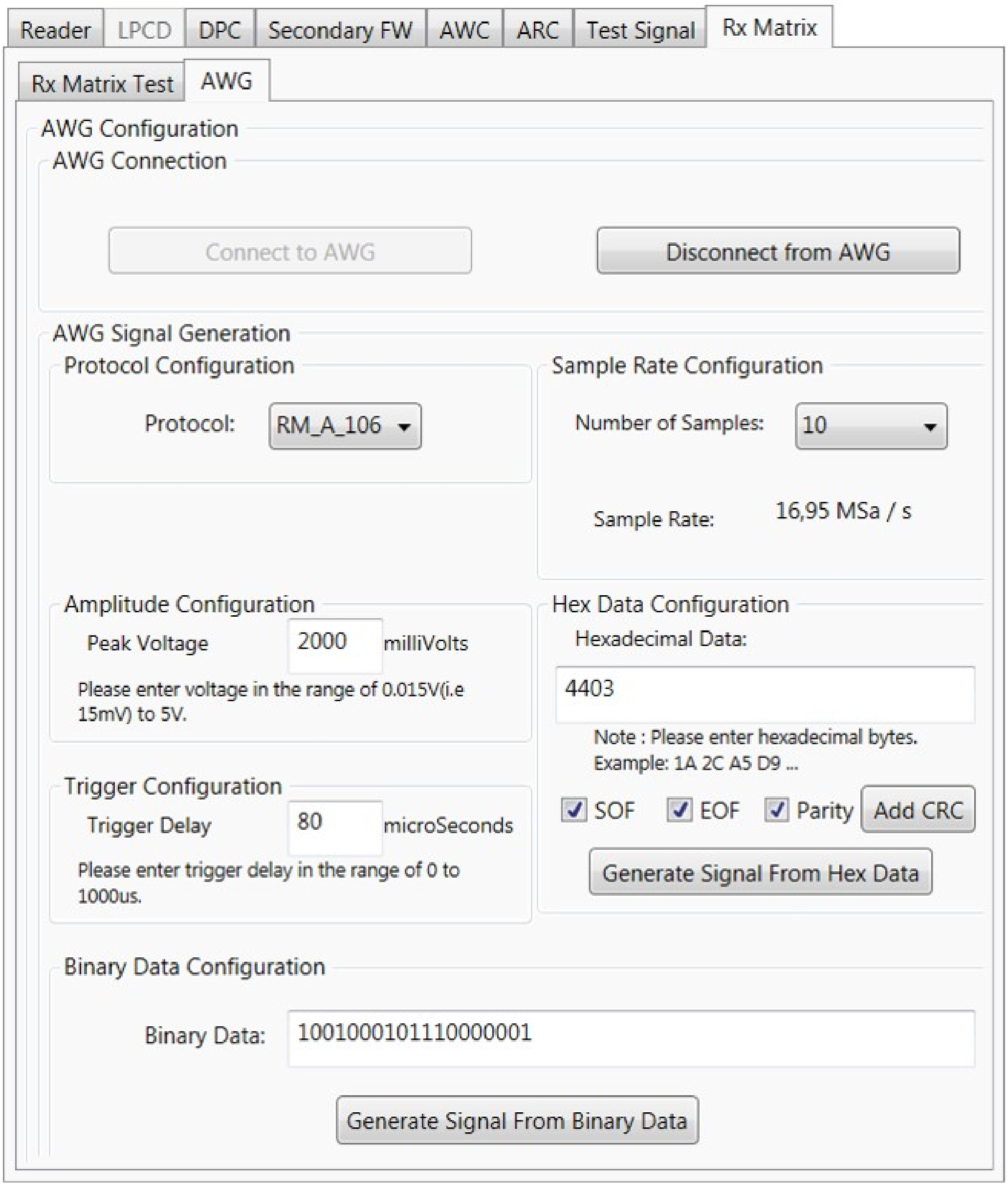


1. Install the .NET (IVI) development support!

Figure 25. NI VISA installation

4.8.2 AWG setup and test for type A @ 106

The Fig 26 shows a typical setup for a type A response. After connecting the AWG the type A protocol with 106 kbit/s can be chosen. The sample rate defines the number of samples per half of a subcarrier cycle.



1. ATQA response for TX_IRQ as trigger

Figure 26. AWG setup for type A @ 106

The amplitude defines the peak voltage level of the LMA: the LMA output toggles between 0V and the defined amplitude. The AWG output can directly drive a Reference PICC modulation input.

Note: EMVCo LMA levels normally are in the range of 700 … 800 mV for minimum LMA test in operating Volume 1. For compliance test it is recommended to use a calibrated reference tool like e.g. the CTC Advanced WavePlayer.

The PICC response itself can be defined as hexadecimal data. The Fig 26 shows the example of a MIFARE DESFire like ATQA, which does not contain a CRC, but SOF, EOF and parity.

The trigger delay defines the delay between AWG trigger input and the LMA sequence start. The given 80µs define a standard FDT for type A, if the TX_IRQ signal is taken as trigger signal.

<Generate Signal From Hex Data> generates the binary as well as the subcarrier sequence, and automatically loads this sequence and related settings into the AWG.

A simple test can be done:

Step 1: Setup the hardware as shown in Fig 24. Place the Reference PICC close to the PCD antenna.

Step 2: Setup the AWG in the NFC Cockpit as defined above.

Step 3: Enable the test bus and route TX_IRQ to a test pin (e.g. IRQ pin).

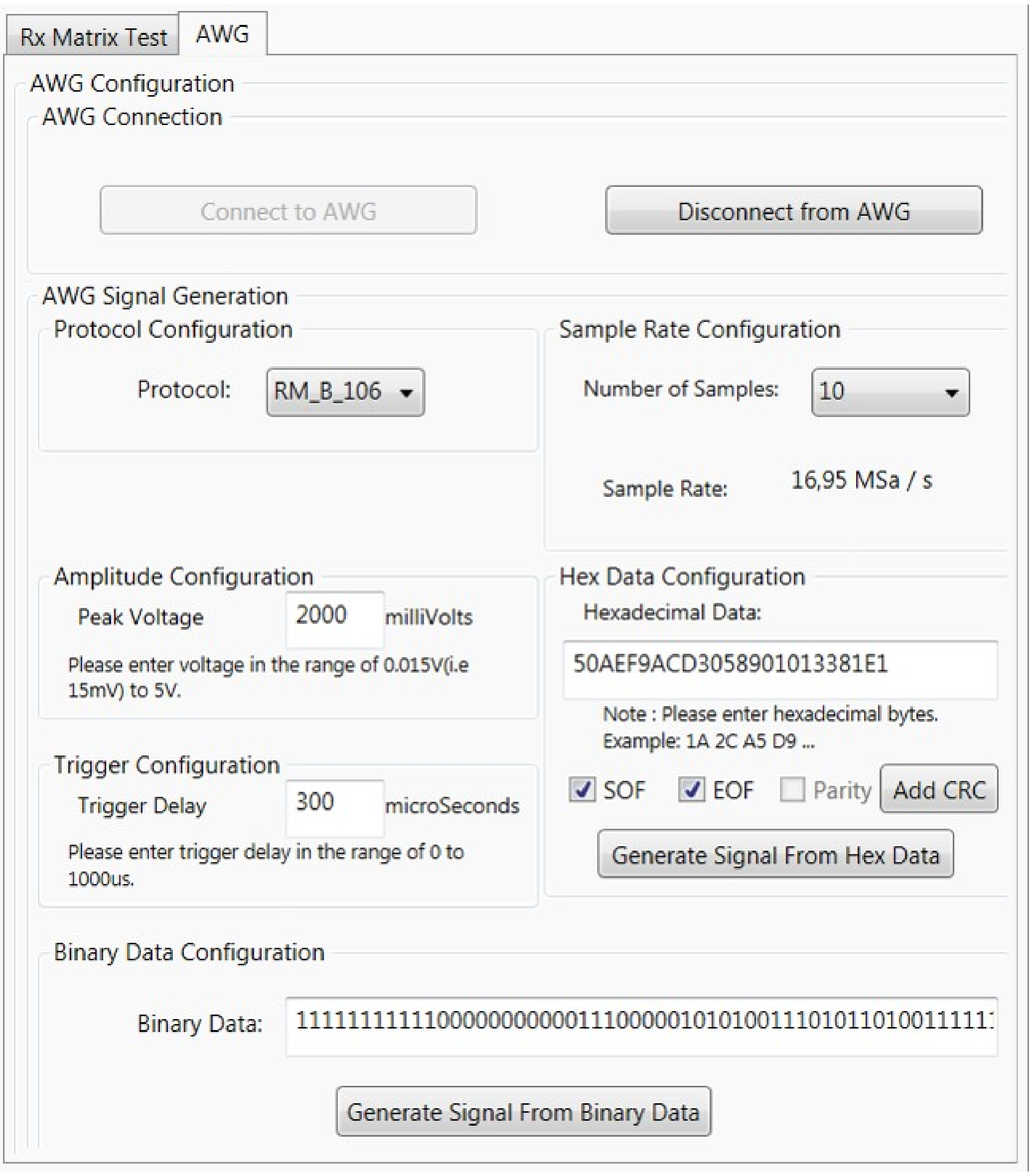
Step 4: Load Protocol with type A 106 and enable the RF Field.

Step 4: Send a single REQA. -> the ATQA should be received properly.

Note: After loading the settings and the sequence into the AWG, the AWG can be switched to “local control”. This allows a faster direct control for manual tests of e.g. the trigger delay or the LMA amplitude at the AWG without reloading all the settings again.

4.8.3 AWG setup and test for type B @ 106

The Fig 27 shows a typical setup for a type B response. After connecting the AWG the type B protocol with 106 kbit/s can be chosen. The sample rate defines the number of samples per half of a subcarrier cycle.



1. ATB response 0x50AEF9ACD3058901013381E1 with TX_IRQ as trigger.

Figure 27. AWG setup for type B @ 106

The amplitude defines the peak voltage level of the LMA: the LMA output toggles between 0V and the defined amplitude. The AWG output can directly drive a Reference PICC modulation input.

Note: EMVCo LMA levels normally are in the range of 700 … 800 mV for minimum LMA test in operating Volume 1. For compliance test it is recommended to use a calibrated reference tool like e.g. the CTC Advanced WavePlayer.

The PICC response itself can be defined as hexadecimal data. The Fig 27 shows the example of a ATQB (0x50AEF9ACD3058901013381E1), which does not yet contain a CRC, but SOF and EOF.

<Add CRC> adds the CRC to the given string (0xE012).

The trigger delay defines the delay between AWG trigger input and the LMA sequence start. The given 300µs define a standard TR0 for type B, if the TX_IRQ signal is taken as trigger signal.

<Generate Signal From Hex Data> generates the binary as well as the subcarrier sequence, and automatically loads this sequence and related settings into the AWG.

A simple test can be done:

Step 1: Setup the hardware as shown in Fig 24. Place the Reference PICC close to the PCD antenna.

Step 2: Setup the AWG in the NFC Cockpit as defined above.

Step 3: Enable the test bus and route TX_IRQ to a test pin (e.g. IRQ pin).

Step 4: Load Protocol with type B 106 and enable the RF Field.

Step 4: Send a single REQB. -> the ATQB should be received properly.

<u>Note:</u> After loading the settings and the sequence into the AWG, the AWG can be switched to “local control”. This allows a faster direct control for manual tests of e.g. the trigger delay or the LMA amplitude at the AWG without reloading all the settings again.

4.8.4 Rx Matrix test with AWG

The Rx Matrix test allows to control the LMA level of the PICC response, if the AWG is setup as described above.

The type B script file example as shown in 6.2 can be used to check the Rx performance in e.g. 2cm operating distance (see Fig 28).

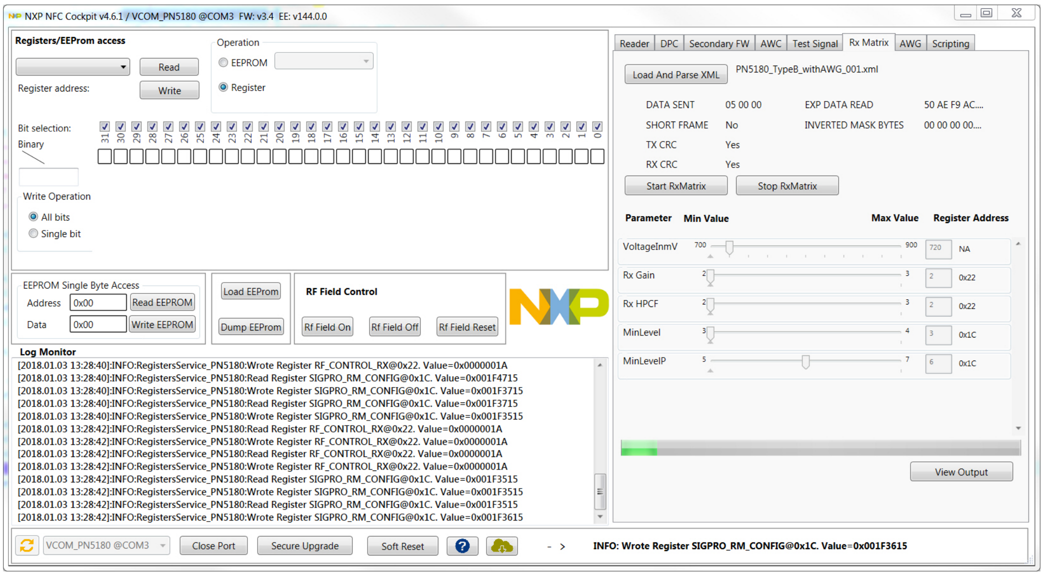
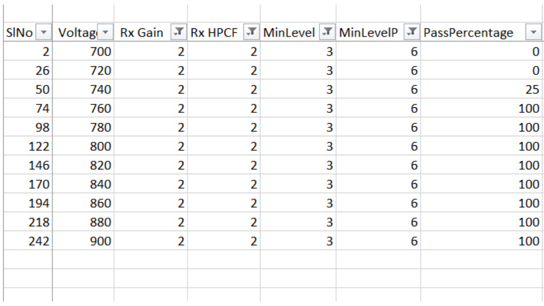
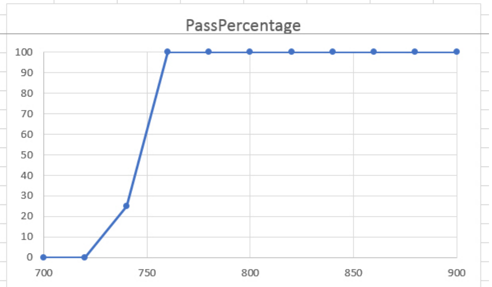


Figure 28. Rx Matrix test run example with AWG

The Fig 29 shows the result of such a test run to indicate the sensitivity limit with RxGain = 2, HPCF = 2, MinLevel = 3 and MinLevel = 6.



SlNo	Voltage	Rx Gain	Rx HPCF	MinLevel	MinLevelP	PassPercentage
2	700	2	2	3	6	0
26	720	2	2	3	6	0
50	740	2	2	3	6	25
74	760	2	2	3	6	100
98	780	2	2	3	6	100
122	800	2	2	3	6	100
146	820	2	2	3	6	100
170	840	2	2	3	6	100
194	860	2	2	3	6	100
218	880	2	2	3	6	100
242	900	2	2	3	6	100



1. The graph shows pass rate versus LMA input level @ Reference PICC
2. With RxGain = 2 the performance is less than optimum.

Figure 29. Rx Matrix Result example with AWG

4.9 PN5180 Low power card detection

The NFC Cockpit allows the configuration and test of the Low Power Card Detection (LPCD) of the PN5180 as shown in Fig 30.

The LPCD parameter, which are stored in the EEPROM (details refer to [1]), can be changed and the LPCD can be started.

Note: Disable the test bus, if you do not see the LPCD tab. Then restart the NFC Cockpit.

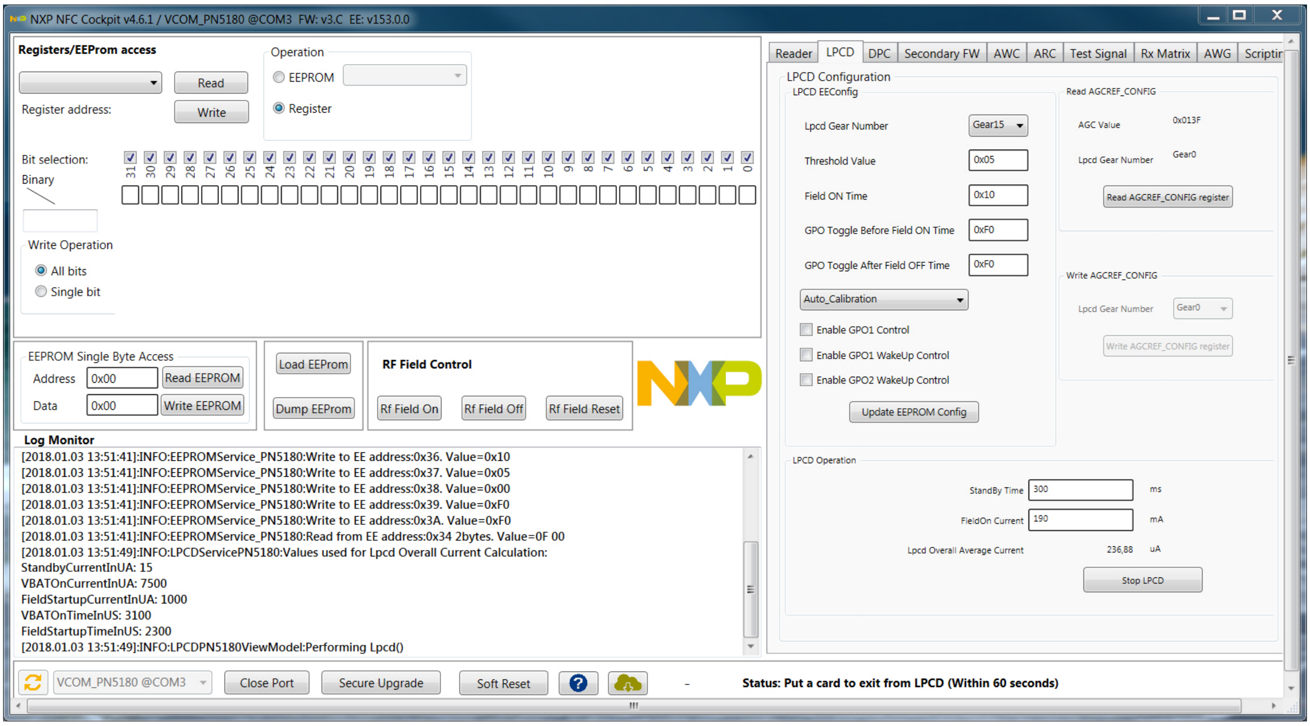
The LPCD tab allows to directly define and write the related EEPROM addresses:

LPCD Gear #: Defines the gear number, which is used for the LPCD in autocalibration mode, stored in adr. 0x34, bit 0..3

Threshold Value: Defines the threshold window. As soon as the AGC value during the LPCD ping exceeds the AGC reference value + threshold window, the IRQ will be raised and the PN5180 wakes up.

Field On Time: Defines the ping length in multiple of 8µs + a fix time of 62µs.

Example: 0x10 -> 16 x 8µs + 62 µs = 190µs



1. LPCD has not been started yet.

Figure 30. PN5180 LPCD

Auto_Calibration: the LPCD calibration is done automatically when the LPCD is started, using the gear and threshold as defined in the EEPROM. This mode always uses the same gear for the LPCD, and is the fastest and easiest way to start the LPCD. It is recommended to choose a gear, which always keeps the ITVDD and field strength limits, so normally e.g. the highest available gear number.

Self_Calibration: The LPCD calibration must be manually triggered, with reading or writing into the AGCREF_CONFIG register. The LPCD calibration can be done in two different options:

Option 1: Read AGC_REF_CONFIG register: This command executes a standard RF Field on. So, depending on the load condition the DPC adjusts the output power. The final gear is take as gear for the LPCD. This option guarantees that the maximum output power is taken for the LPCD.

Option 2: Write AGC_REF_CONFIG register: This command executes a LPCD calibration ping with the gear number, as defined in the AGC_REF_CONFIG, bit 10..13. This option allows a flexible use of any of the defined gears for the LPCD.

Enable GPO1 or GPO2 Control and Time: Refer to [1].

Standby time: This value defines the time between two pings in ms.

FieldOn Current: This value is the ITVDD under the loading condition, when RF field is on with the used gear. This value does not have any influence on the LPCD execution or performance as such, but simply is used to estimate the overall average current consumption. This current estimation is calculated, when the LPCD is started.

Note: The related timings and current consumptions during the ping, as needed for the estimation of the overall average current consumption, are defined in: c:\nxp \NxpNfcCockpit_v4.6.1.0\cfg\NxpNfcCockpit_Configuration.ini

Note: The LPCD requires a stable TVDD, otherwise the AGC value is not stable and the PN5180 wakes up easily. So, for standard USB only operation (no external power supply) it is recommended to use

a) a higher gear number (which reduces ITVDD and TVDD ripple)

b) a longer RFON time (which stabilizes the TVDD)

Note: The LPCD mode disconnects the PN5180 from the LPC (to save power), so the LPCD mode can only be left, if either a card loads the antenna and wakes up the PN5180, or the LPC resets the PN5180.

4.10 Secondary FW and EMVCo Loopback application

The Secondary Firmware tab allows to flash another secondary FW. The secondary FW of the LPC handles the communication between the NFC Cockpit and the PN5180. The secondary FW might contain additional applications, which can then be started via the Secondary Firmware tab.

The default applications are:

EMVCo Loopback: Test function for EMVCo test

Transaction Send A: Test function for EMVco test

Transaction Send B: Test function for EMVco test

The EMVco Loopback (or other application) can be started by pressing the <Start Secondary Firmware> button, and then is executed on the LPC. The function can be stopped by pressing the <Stop Secondary Firmware> button, but afterwards the PN5180 must be reset to continue with the standard NFC Cockpit functionality.

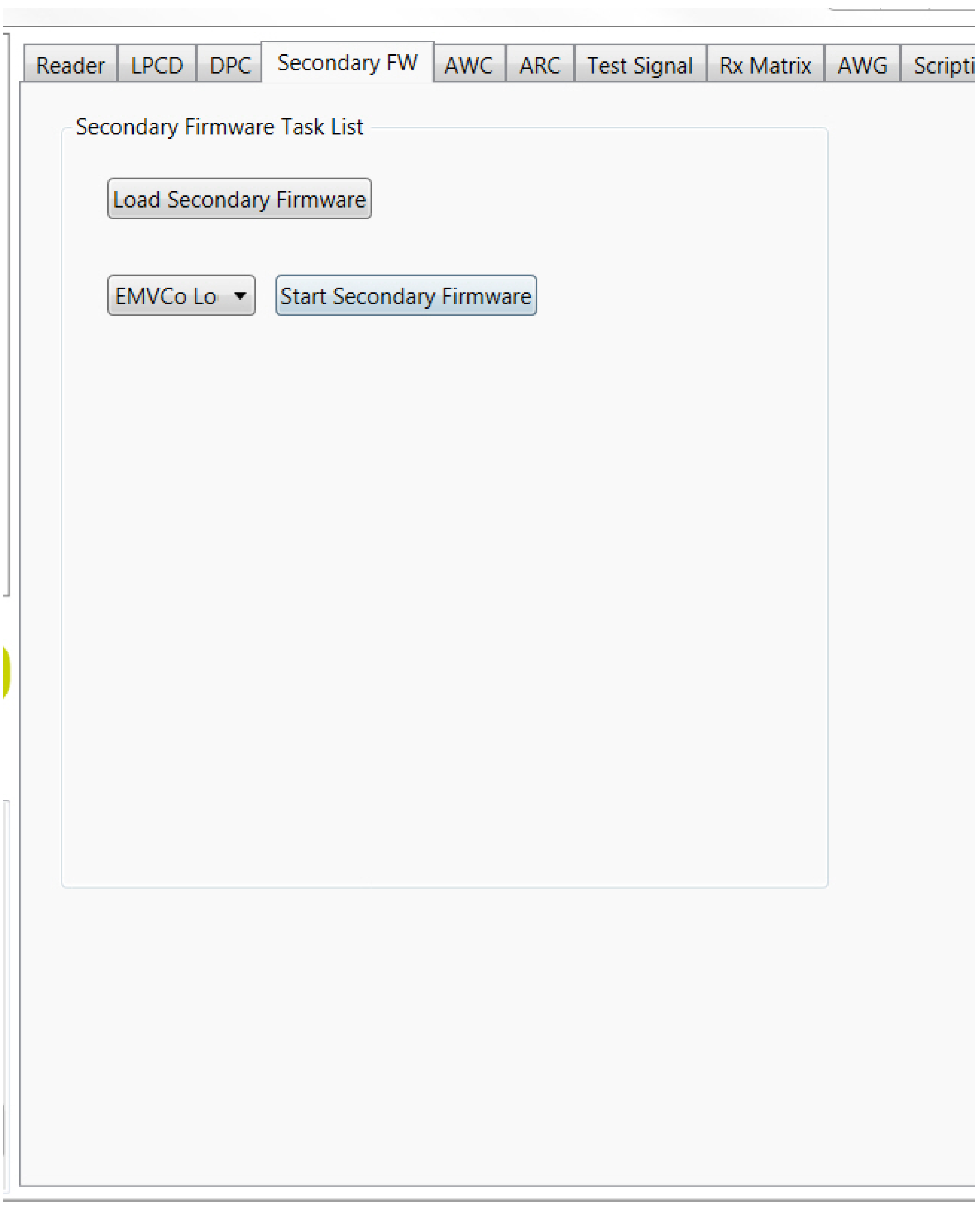


Figure 31. Secondary Firmware tab with EMVCo Loopback function

4.11 PN5180 Scripting

The NFC Cockpit allows to use a simple script language to program own scripts for test purpose. This feature is mainly developed for the CLRC663, and the number of implemented commands for the PN5180 is limited.

There is a sample script, which simply reads the FW version of the PN5180:

c:\nxp\NxpNfcCockpit_v<VERSION>\scripts\

5 First time use

Make sure the LPC1769 is flashed with the correct LPC FW (see 3.2).

Make sure that the VCOM driver is installed (see 3.1).

5.1 Jumper settings

The default jumper settings allow a direct use with the USB connector only. This might show limited performance due to a current limitation on the USB host. So, for real performance measurements the external power supply should be used.

5.1.1 USB only

The jumper settings as shown in Fig 9 provide the default settings, using only USB for power supply (no external supply required).

5.1.2 External power supply

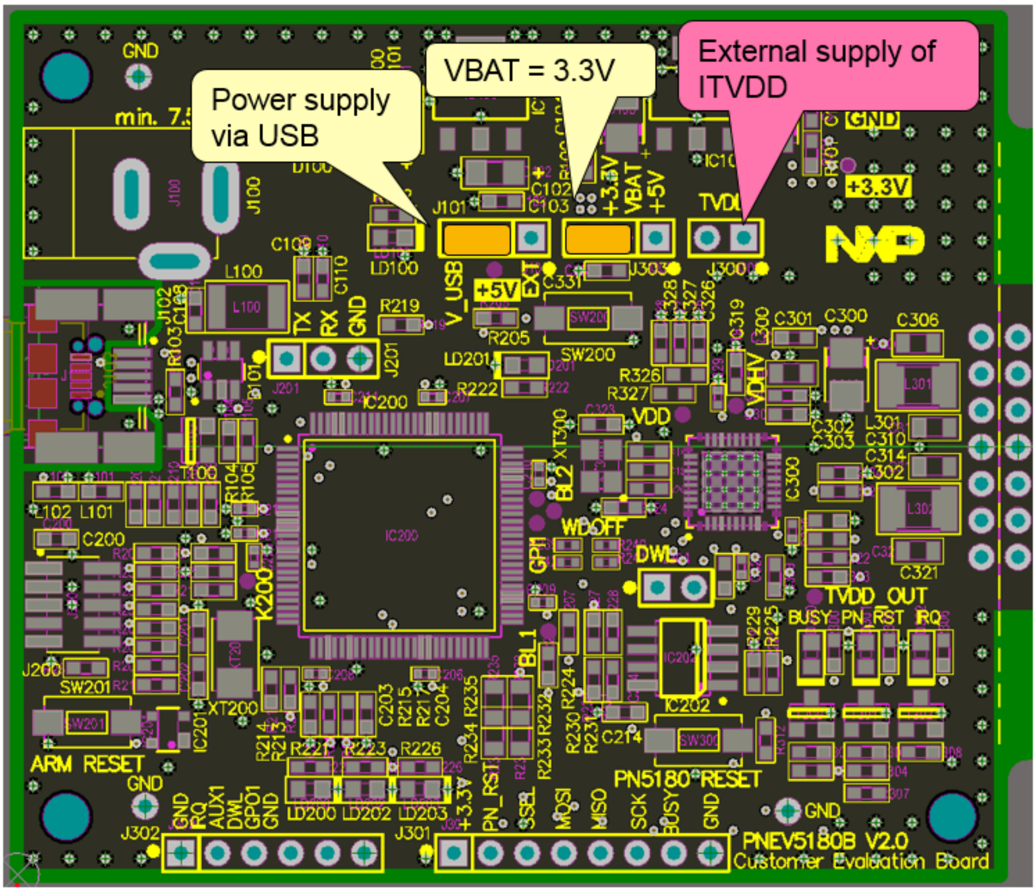
For the use of an external power supply the jumper J101 must be changed as shown in Fig 10.

The external power supply must provide a voltage level of V_{ext} = 7.5 … 12V with 500mA.

For some of the analog tests (i.e. measuring ITVDD) it might be useful to only power the TVDD supply externally. This can be done using the jumper JP300, as shown in Fig 32.

Either the jumper can be replaced with a DC ampere meter to measure the ITVDD, or an external 5Vdc power supply can be directly connected to the right pin of JP300.

Note: Several GND pins are provided on the board. They all are connected.



1. JP300 can be used to externally supply TVDD.

Figure 32. PN5180 jumper settings with external TVDD

6 Annex A Rx Matrix XML input file examples

6.1 Type A example without AWG control

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<!DOCTYPE Test SYSTEM "NNC_RxMatrix_Pn5180.dtd">
<!-- This is an example of a TypeA test script for PN5180 where
 we acess registers by names -->
<Test
numberMaxOfPasses="10"
skipAfterFailures="4"
delayMS="0"
fieldReset="YES"
protocolType="RM_A_106"
>
<SendData shortFrame="YES" rxCRC="NO" txCRC="NO"
 timeOutInUs="1000">
0x26
</SendData>
<ReadData invertedMaskBytes="0x00, 0x00">
```

```
0x44, 0x03
</ReadData>
<Parameter register="RF_CONTROL_RX" field="RX_GAIN"
 minValue="0x01" maxValue="0x03" />
<Parameter register="RF_CONTROL_RX" field="RX_HPCF"
 minValue="0x00" maxValue="0x03" />
</Test>
```

6.2 Type B example with AWG control

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<!DOCTYPE Test SYSTEM "NNC_RxMatrix_Pn5180.dtd">
<!-- This is an example of a TypeB test script for PN5180 with
 AWG control-->
<!--  numberMaxOfPasses: 10 trial per combination is tested
skipAfterFailures: continue with next combination, if too many
 (>4) failures
delayMS: enter delay in ms, if needed (default = 0)
fieldReset: enable RF-Reset, if needed (default = disabled)
protocolType: we use type B at 106 -->
<Test
numberMaxOfPasses="20"
skipAfterFailures="18"
delayMS="0"
fieldReset="NO"
protocolType="RM_B_106"
>
<SendData shortFrame="NO" rxCRC="YES" txCRC="YES"
 timeOutInUs="50000">
<!-- REQB: -->
0x05, 0x00, 0x00
</SendData>
<!-- As a reponse we expect ATQB = 12 bytes + CRC: -->
<ReadData invertedMaskBytes="0x00, 0x00, 0x00, 0x00, 0x00,
 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00">
<!-- no masking required, since we defined exactly these bytes
 in the AWG -->
0x50, 0xAE, 0xF9, 0xAC, 0xD3, 0x05, 0x89, 0x01, 0x01, 0x33,
 0x81, 0xE1
</ReadData>
<!-- AWG Voltage levels in mV -->
<VoltageLevel minValueInmV="700" maxValueInmV="900"
 stepSizeInmV="20"/>
<Parameter name="Rx Gain" minValue="0x02" maxValue="0x03"
 registerAddress="0x22" bitPosition="0" bitLength="2" />
<Parameter name="Rx HPCF" minValue="0x02" maxValue="0x03"
 registerAddress="0x22" bitPosition="2" bitLength="2" />
<Parameter name="MinLevel" minValue="0x03" maxValue="0x04"
 registerAddress="0x1c" bitPosition="12" bitLength="4" />
<Parameter name="MinLevelP" minValue="0x05" maxValue="0x07"
 registerAddress="0x1c" bitPosition="8" bitLength="4" />
</Test>
```

7 Annex B

7.1 LPC ABEND Driver installation (if needed)

Note: this section is not needed anymore, if the LPC FW update is done a described in 3.2.3.

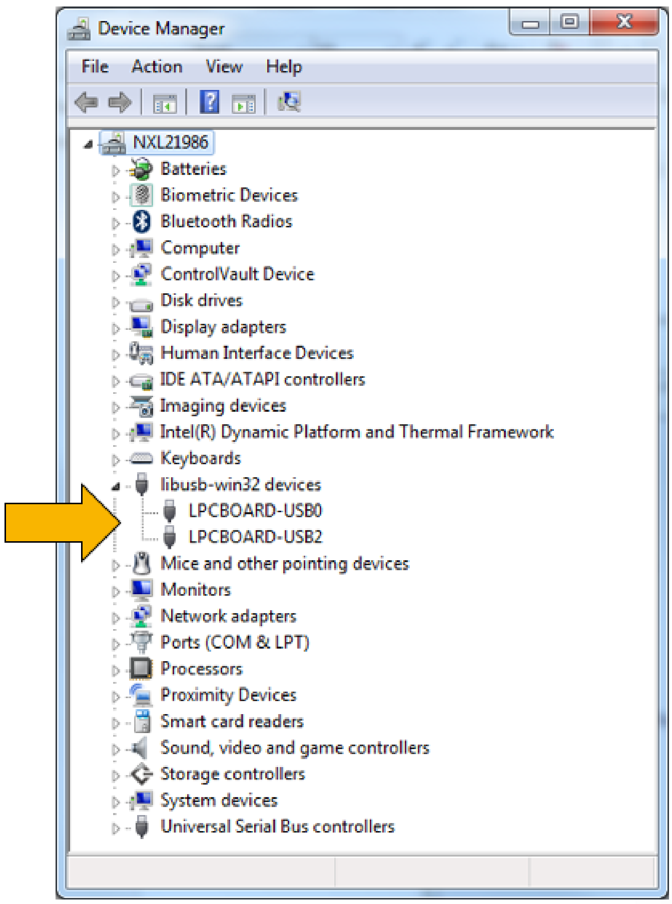
At first connection of the PNEV5180B (with default firmware) to the PC, the device asks for a driver. The driver must be chosen from

c:\nxp\NxpNfcCockpit_v<VERSION>\AbendPCDrivers\\PC_SW
\LPCBOARD_DRIVER_WIN

After successful installation of the driver, the device must be registered as two LIBUSB_WIN32 devices (one for each SPI line), as shown in Fig 33.

If only one device is registered, either old drivers have been chosen or an older version of lpc_main.bin is used.

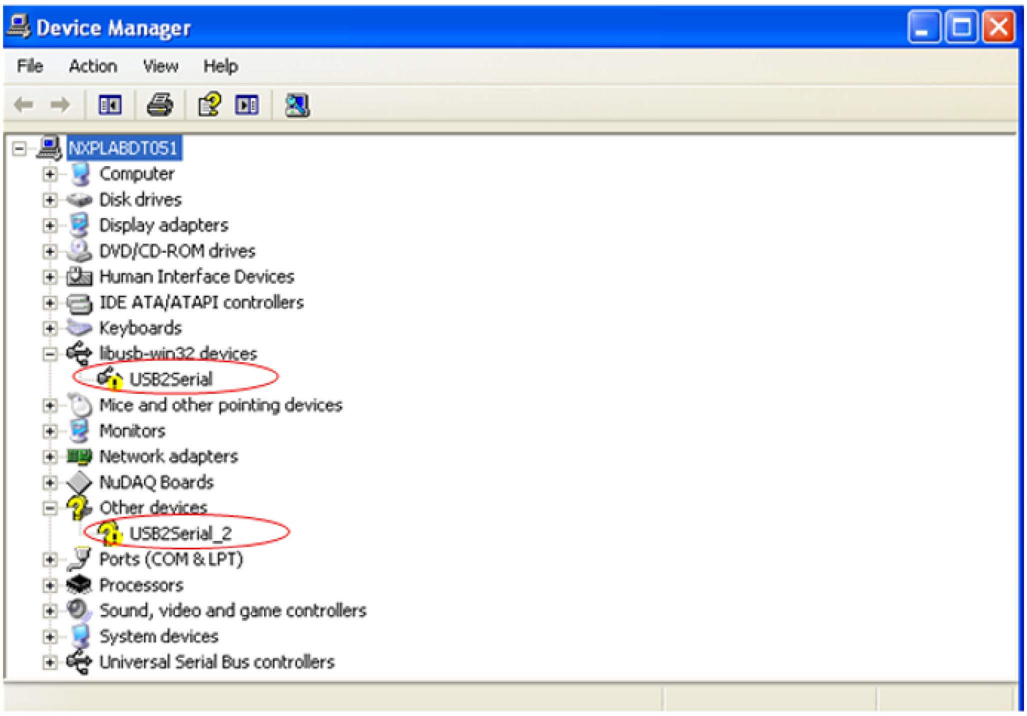
<u>Note for possible future NFC Cockpit updates:</u> Please make sure to use latest driver version, otherwise the application might not work correctly. In case of doubt re-install the driver of the corresponding NFC Cockpit package.



1. Two LPCBOARD-USB devices must show up.

Figure 33. LPC driver

Note: In some cases a wrong driver might be automatically installed without notice after connection of the PNEV5180B board (see Fig 34). In such case the driver needs to be manually updated.



1. This setup does not work properly.

Figure 34. Wrong driver installed

8 References

1. PN5180 datasheet, www.nxp.com
2. AN11740 PN5180 Antenna design guide
3. AN11741 PN5180 DPC Antenna design
4. AN11742 PN5180 Dynamic Power Control
5. UM10954 PN5180 SW Quick start guide
6. NFC Cockpit: https://www.nxp.com/products/identification-and-security/nfc/nfc-reader-ics/nfc-cockpit-configuration-tool-for-nfc-ics:NFC-COCKPIT
7. NFC Reader Library: https://www.nxp.com/products/identification-and-security/nfc/nfc-reader-ics/nfc-reader-library-software-support-for-nfc-frontend-solutions:NFC-READER-LIBRARY
8. Technical Video Tutorials: https://www.nxp.com/products/identification-and-security/nfc/nfc-reader-ics/high-performance-multi-protocol-full-nfc-forum-compliant-frontend-optimized-for-pos-terminals:PN5180
9. Keysight 33500B: https://www.keysight.com
10. NI VISA driver: http://www.ni.com/download/ni-visa-17.0/6646/en/

9 Legal information

9.1 Definitions

Draft — The document is a draft version only. The content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included herein and shall have no liability for the consequences of use of such information.

9.2 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors. In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory. Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification. Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products. NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Evaluation products — This product is provided on an "as is" and "with all faults" basis for evaluation purposes only. NXP Semiconductors, its affiliates and their suppliers expressly disclaim all warranties, whether express, implied or statutory, including but not limited to the implied warranties of non-infringement, merchantability and fitness for a particular purpose. The entire risk as to the quality, or arising out of the use or performance, of this product remains with customer. In no event shall NXP Semiconductors, its affiliates or their suppliers be liable to customer for any special, indirect, consequential, punitive or incidental damages (including without limitation damages for loss of business, business interruption, loss of use, loss of data or information, and the like) arising out the use of or inability to use the product, whether or not based on tort (including negligence), strict liability, breach of contract, breach of warranty or any other theory, even if advised of the possibility of such damages. Notwithstanding any damages that customer might incur for any reason whatsoever (including without limitation, all damages referenced above and all direct or general damages), the entire liability of NXP Semiconductors, its affiliates and their suppliers and customer's exclusive remedy for all of the foregoing shall be limited to actual damages incurred by customer based on reasonable reliance up to the greater of the amount actually paid by customer for the product or five dollars (US$5.00). The foregoing limitations, exclusions and disclaimers shall apply to the maximum extent permitted by applicable law, even if any remedy fails of its essential purpose.

Translations — A non-English (translated) version of a document is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

9.3 Licenses

Purchase of NXP ICs with NFC technology

Purchase of an NXP Semiconductors IC that complies with one of the Near Field Communication (NFC) standards ISO/IEC 18092 and ISO/IEC 21481 does not convey an implied license under any patent right infringed by implementation of any of those standards. Purchase of NXP Semiconductors IC does not include a license to any NXP patent (or other IP right) covering combinations of those products with other products, whether hardware or software.

9.4 Trademarks

Notice: All referenced brands, product names, service names and trademarks are the property of their respective owners.

MIFARE — is a trademark of NXP B.V.

DESFire — is a trademark of NXP B.V.

Tables

Figures

Contents

Date of release: 2 February 2018
Document identifier: AN11744
Document number: 371213