AN11802

NFC Reader Library for Linux Installation Guidelines

Rev. 1.0 — 4 April 2016
367910

Application note
COMPANY PUBLIC

Document information

Info	Content
Keywords	NFC, Reader Library, Linux, PN512, PN5180, CLRC663, EXPLORE-NFC, Raspberry Pi
Abstract	This note describes how to install NXP’s NFC Reader Library on a GNU/Linux system

Revision history

Rev	Date	Description
1.0	20160404	First release

Contact information

For more information, please visit: http://www.nxp.com

1. Introduction

The NFC Reader Library is a feature complete software support library for NXP's NFC Frontend ICs. It is designed to give developers a faster and simpler way to deliver NFC-enabled products. This multi-layer library, written in C, makes it easy to create NFC based applications. See [1] for more details.

The purpose of the present document is to give instructions on how to install the NFC Reader Library on a generic GNU/Linux platform.

It takes as reference the support of EXPLORE-NFC (see [2]), including PN512 NXP's NFC Frontend (see [4]), on Raspberry Pi platform (refer to [5] for more details) running Raspbian Jessie distribution.

Fig 1. EXPLORE-NFC on Raspberry Pi

In below chapters, information highlighted thanks to surrounded borders relates to examples on this reference platform.

In case of use of this reference platform, it is advised to first follow the EXPLORE-NFC Quick start-up guide [3] before going further here.

Finally it gives details about modifications to be done to make the porting of the NFC Reader Library for Linux to other Linux platform than the one used as reference or for other NXP NFC Frontend ICs.

2. NFC Reader Library for Linux delivery

2.1 Step 1: installing NFC Reader Library for Linux delivery

The NFC Reader Library delivery consists of a zip file to be uncompressed on the Linux target:

```
$ unzip sw3693.zip
```

This creates the following directory structure in the current path:

```
├── addition
├── build
├── examples
│   └── ...
├── linux
│   └── ...
├── nxprdlib
│   └── ...
├── CMakeLists.txt
├── NXP_License.pdf
└── README.txt
```

- {addition} contains optional “bal” kernel module source (refer to 3.5)
- {build} is the folder later used for the binaries generation
- {examples} contains code examples of the NFC Reader Library use
- {linux} contains OS and platform related specific implementation
- {nxprdlib} is the NFC Reader Library source code
- {CMakeLists.txt} allows building makefiles structure
- {README.txt} gives information about the NFC Reader Library for Linux delivery

2.2 Step 2: generating makefiles structure

Generating the makefiles structure requires the following modules been installed on the target:

- *gcc*
- *make*
- *cmake* (version 2.8.11 minimum)

On reference platform (Raspberry Pi running Raspbian Jessie), *cmake* installation is done using command:

```
$ sudo apt-get install cmake
```

Makefiles generation is then done running *cmake* command from the build sub-directory:

```
$ cd build
$ cmake ..
```

2.3 Step 3: building the library

Building the example, including the NFC Reader Library source code) is then just done with the simple *make* command:

```
$ make
```

This generates the library binary **libNxpRdLibLinux**<*NfcFrontendReference*>**.a** under *build/linux* sub-folder, and **NfcrdlibEx**<*#_ExampleName*><*NfcFrontendReference*> examples binaries under *build/examples*.

2.4 Step 4: enabling the SPI physical interface

The physical link used to interface the NFC Frontend must be enabled on the platform: node */dev/spidev* must be present and accessible.

On the Raspberry Pi, enabling SPI is done through raspi-config tool:

```
$ sudo raspi-config
```

The option to activate SPI can be found in:
Advanced Options → SPI → <Yes>.
Then reboot is required to take into account the change:

```
$ sudo reboot
```

2.5 Step 5: running the examples

Running the examples is done in executing the related command:

```
$ ./NfcrdlibEx<#_ExampleName><NfcFrontendReference>
```

Below is the output obtained running *NfcrdlibEx1_BasicDiscoveryLoopPN512*, tapping an NFC tag:

```
$ ./NfcrdlibEx1_BasicDiscoveryLoopPN512
 BasicDiscoveryLoop Example:


 Card detected and activated successfully...
  Technology  : Type A
     Card: 1
```

```
UID : 04 60 32 6A 64 34 80

SAK : 0x00

Type: Type 2 Tag
```

3. Porting of the NFC Reader Library for Linux

The NFC Reader Library for Linux delivery is suitable to run the reference platform (Raspberry Pi together with PN512 EXPLORE-NXP board). In case of different setup, definition of the header file *phhwConfig.h*, located under *linux/shared* sub-folder of the delivery, must be adapted.

3.1 Support of different Raspberry Pi revision

By default, the implementation fits the EXPLORE-NFC board attached to Raspberry Pi revision 2 (models A/B, A+/B+ and 2B with P5 header). In case of use of an older revision of the Raspberry Pi board (model A/B with no P5 header), the following line in *phhwConfig.h* must be modified to define "`BOARD`" as "`RaspberryPi_Rev1_Pinout`" instead of "`RaspberryPi_Rev2_Pinout`":

```
…
#ifndef BOARD
#define BOARD RaspberryPi_Rev2_Pinout
#endif
…
```

3.2 Support of PN5180

The delivery already allows (through `NXPBUILD__PHHAL_HW_PN5180` compile flag) generating all binaries (reader library, examples …) required to interface to a PN5180 [6], instead of PN512, on Raspberry Pi.

The PNEV5180B board must be connected to the Raspberry Pi according to Table 1 and Fig 2.

The Power of PNEV5180 board still come from either the USB or the 5V Power connectors.

Table 1. PNEV5180 connection to Raspberry Pi

Wiring to be done to connect PNEV5180 demo board to Raspberry Pi

PNEV5180 pin	Raspberry Pi	
	Pin number	Pin function
MOSI	19	SPI-MOSI (GPIO 10)
MISO	21	SPI-MISO (GPIO 9)
SCK	23	SPI-CLK (GPIO 11)
NSS/SSEL	24	SPI-CE0 (GPIO 8)
nRESET/PN_RST	26	GPIO 7
BUSY	22	GPIO 25
IRQ	16	GPIO 23
GND	6, 9, 14, 20, 25, 30, 34 or 39	GND

In case a different wiring is used, this must be reflected in *linux/shared/phhwConfig.h* header file.

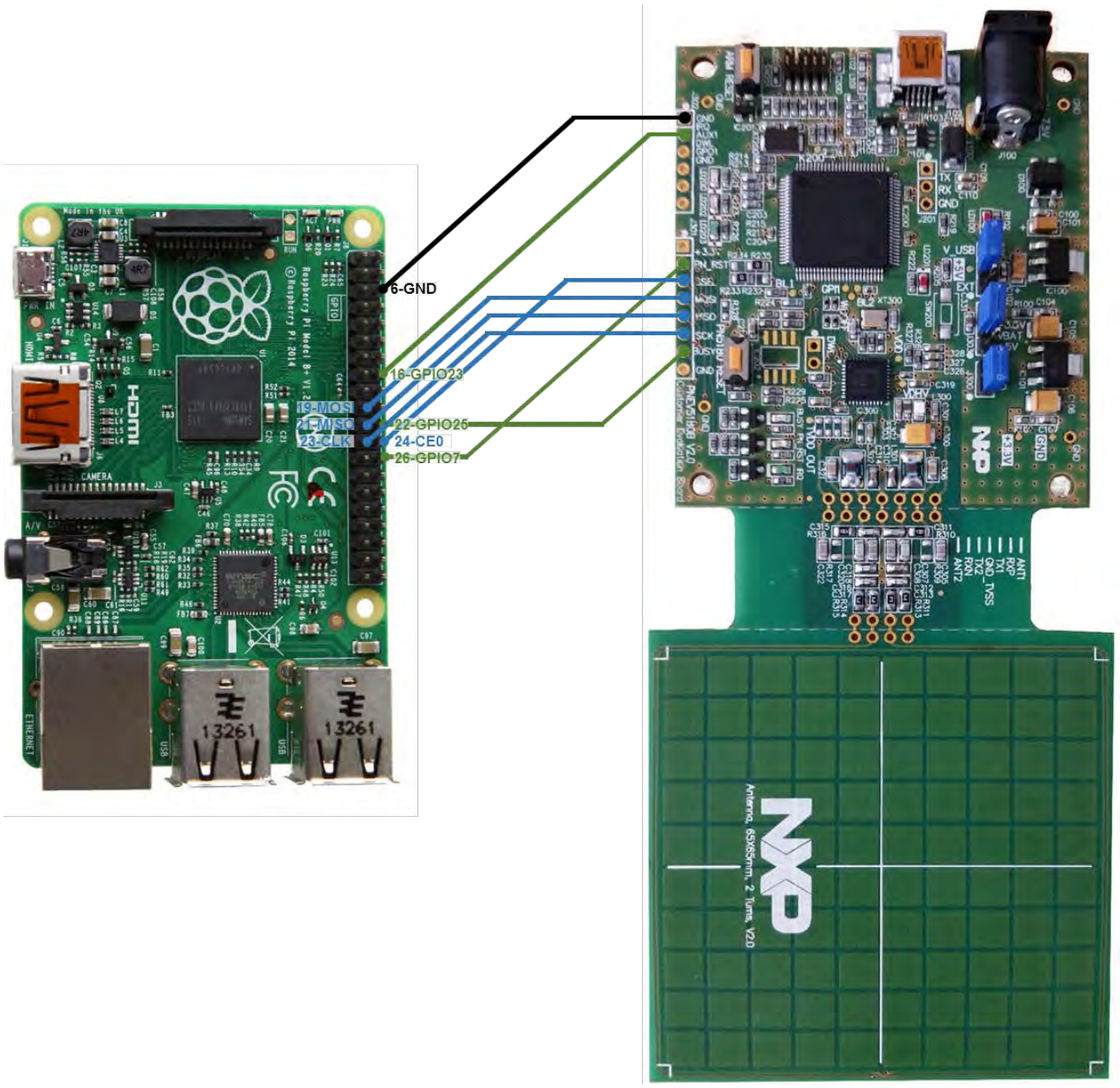

Fig 2. PNEV5180 connection to Raspberry Pi

3.3 Support of CLRC663

The delivery already allows (through `PHBAL_REG_HAL_HW_RC663` compile flag) generating all binaries (reader library, examples …) required to interface to a CLRC663 [7], instead of PN512, on Raspberry Pi.

The CLEV663B board must be connected to the Raspberry Pi according to Table 2 and Fig 3.

The CLEV6663B board must have been into SPI configuration previously as indicated in the related document [8].

Table 2. CLEV663B connection to Raspberry Pi

Wiring to be done to connect CLEV663B demo board to Raspberry Pi

CLEV663B		Raspberry Pi	
Pin number	**Pin function**	**Pin number**	**Pin function**
2	5V_VIN	2	+5V
5	MOSI	19	SPI-MOSI (GPIO 10)
6	MISO	21	SPI-MISO (GPIO 9)
7	SCK	23	SPI-CLK (GPIO 11)
8	SSEL	24	SPI-CE0 (GPIO 8)
28	3V3	1	+3V
39	PDOWN (P0.3)	18	GPIO 24
46	IFSEL0 (P2.4)	13	GPIO 27
47	IFSEL1 (P2.5)	15	GPIO 22
53	IRQ (P3.3)	16	GPIO 23
1 or 54	GND	6, 9, 14, 20, 25, 30, 34 or 39	GND

In case a different wiring is used, this must be reflected in *linux/shared/phhwConfig.h* header file.

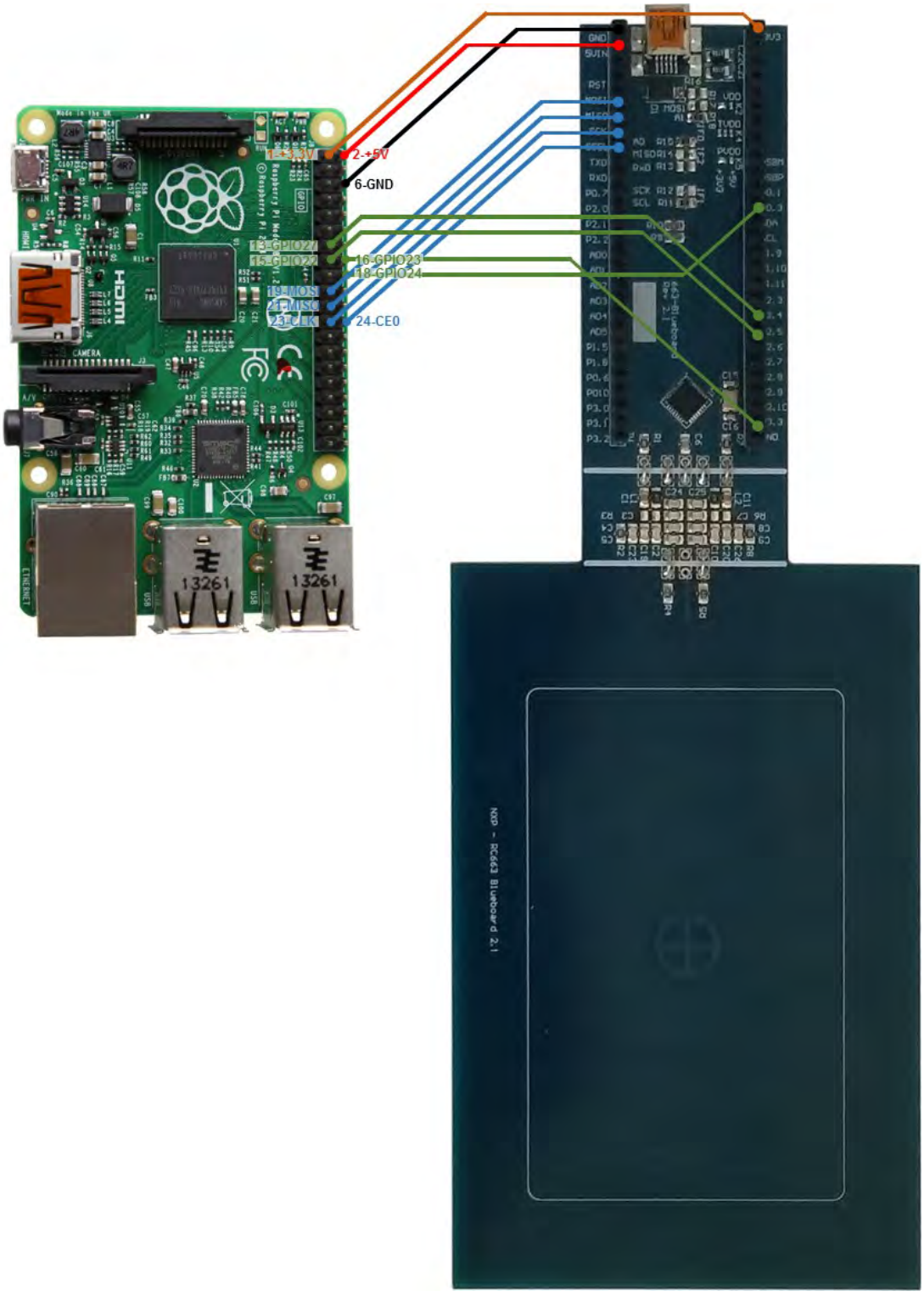

Fig 3. CLEV663B connection to Raspberry Pi

3.4 Support of other Linux platform

Porting of the current delivery to other Linux platform than the reference one (Raspberry Pi) is of course possible while there are providing access from Userspace to SPI and GPIO resources through */dev/spidev* and */sys/class/gpio* nodes.

The following definitions in *phHwConfig.h* header file must then be set accordingly.

```
…
#define SPI_CONFIG    "0:0:25" //SPI bus, CS pin, busy pin
#define PIN_IRQ       23
#define PIN_NRST      7
…
```

- {SPI_CONFIG} defines:
  - the SPI bus number (/dev/spidev node number)
  - the CS pin mapping (0 or 1)
  - the BUSY pin mapping (X = GPIO number, 0 if not used)
- {PIN_IRQ} defines the NFC IC's IRQ pin mapping
- {IN_NRST} defines the NFC IC's nRESET pin mapping

3.5 Support of "bal" kernel module

In order to insure critical timings (for instance for EMVco compliancy) on slow platform, a kernel module is offered. It abstracts SPI access which significantly speed-up the communication time between the Linux platform and the NFC Frontend IC.

Source code of this "bal" kernel module is provided within *addition* sub-folder of the NFC Reader Library for Linux release. All information to build and install this kernel module on the Linux platform is described in the related *README.md* file.

When properly loaded on the Linux platform, the module should expose */dev/bal* device node.

Then, to make use of this kernel module, library Makefiles must be adapted so the relevant BAL layer is later built and linked. This is done using directive ***USE_KERNEL_SPACE_BAL=1*** when generating the makefiles structure (refer to 2.2 Step 2: generating makefiles structure):

```
$ cmake -DUSE_KERNEL_SPACE_BAL=1 ..
```

!!! Only PN5180 is supported by the current "bal" kernel module !!!

Support for PN512 and CLRC663 will be added in further release.

3.6 Support for debug

The Library can be compiled with debug capabilities (debug symbols). This is done using directive ***CMAKE_BUILD_TYPE=Debug*** when generating the makefiles structure (refer to 2.2 Step 2: generating makefiles structure):

```
$ cmake -DCMAKE_BUILD_TYPE=Debug ..
```

4. References

[1] The NFC Reader Library gives Software support for NFC Frontend solutions.

For more information about it please visit http://www.nxp.com/products/identification-and-security/nfc-and-reader-ics/nfc-frontend-solutions/nfc-reader-library-software-support-for-nfc-frontend-solutions:NFC-READER-LIBRARY.

[2] EXPLORE-NFC is a high performance full NFC expansion board for the Raspberry Pi.

For more information about it please visit http://www.nxp.com/products/identification-and-security/nfc-and-reader-ics/nfc-frontend-solutions/explore-nfc-exclusive-from-element14:PNEV512R.

[3] AN11480 Quick Start-up Guide for EXPLORE-NFC working with Raspberry Pi.

[4] PN512 is a highly integrated NFC frontend for contactless communication at 13.56 MHz.

For more information about it please visit http://www.nxp.com/products/identification-and-security/nfc-and-reader-ics/nfc-frontend-solutions/full-nfc-forum-compliant-solution:PN512A0HN1.

[5] The Raspberry Pi is a credit card sized computer. To get started quickly, the Raspberry Pi Foundation provides several preconfigured Linux distributions.

For more information about it please visit www.nxp.com/redirect/raspberrypi.org/

[6] PN5180 is a highly integrated high performance full NFC Forum-compliant frontend IC for contactless communication at 13.56 MHz.

For more information about it please visit http://www.nxp.com/products/identification-and-security/nfc-and-reader-ics/nfc-frontend-solutions/high-power-nfc-frontend-solution:PN5180.

[7] CLRC663, the high performance multi-protocol NFC reader frontend.

For more information about it please visit http://www.nxp.com/products/identification-and-security/nfc-and-reader-ics/nfc-frontend-solutions/high-performance-nfc-reader-solution:CLRC66302HN.

[8] AN11211 Quick Start Up Guide RC663 Blueboard.

5. Legal information

5.1 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.

5.2 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.

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.

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.

5.3 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.

5.4 Trademarks

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

MIFARE — is a trademark of NXP Semiconductors N.V.

6. List of figures

7. Contents

For more information, visit: http://www.nxp.com

Date of release: 4 April 2016
367910
Document identifier: AN11802