DESIGN AND IMPLEMENT NFC APPLICATIONS

SESSION 4: NFC READER LIBRARY
SW SUPPORT FOR NFC FRONTENDS AND NFC CONTROLLERS WITH CUSTOMIZED FIRMWARE

October 2016
Agenda

Design and implement NFC applications

Session I, 7th September
Product support package for NXP NFC readers
https://attendee.gotowebinar.com/rt/2329750067403618817

Session II, 28th September
Antenna design considerations for NXP NFC reader solutions
https://attendee.gotowebinar.com/rt/282682617345186049

Session III, 18th October
The NFC Cockpit - the complete design tool for engineers
https://attendee.gotowebinar.com/rt/4665515186055692345

Session IV, 31st October
NFC Reader Library - SW support for NFC frontend solutions
https://attendee.gotowebinar.com/rt/7151741873899128067
Agenda

Design and implement NFC applications

Session 4, 31st October
NFC Reader Library: SW support for NFC frontends and NFC controllers w. custom. FW

- Introduction to the NFC Reader Library
- NFC Reader Library architecture
- Using the NFC Reader Library
  - Building the SW stack
  - Discovery loop configuration
  - Simplified API
- Software examples
- Documentation
Our support for NFC

NFC IMPLEMENTATION PROCESS

Evaluate the functionality

Select IC

Evaluate Features

Prototyping

- Investigate which NFC functionality you need for your application (e.g., only read cards or write tags or exchange information with another device etc.)

- Nobody gives you more options to choose from. Start with the specs given in this brochure, then go online to get detailed parametric searches.

- Explore the possibilities with one of our development boards, then use that same board to start prototyping.

- PCB design & antenna design

Software

- CONNECTED TAGS
  - Write your code for your MCU using the available code example. Connect additional memory over I2C to your controller enabling e.g. firmware update.

- NFC FRONTENDS & CONTROLLER WITH CUSTOMIZABLE FW
  - Write your code for your MCU using the NFC Library incorporating support for all relevant cards and phones.

- NFC CONTROLLER
  - Write your code for your MCU or MPU using the available code examples calling the functions already embedded in the FW of the NFC controller

Test & Debug

- Standards-based design and support for the most popular development tools make it easy to fine-tune performance, catch errors, and fix bugs.

Get Certified

- Our NFC solutions are designed to help you meet CE, FCC, and other regional requirements, and make it easier to pass EMVCo certification.

NXP SUPPORT

- Product selection app Parametric search and product details on nxp.com
- Z-card with NFC Reader Portfolio
- NFC Everywhere brochure, pp. X-Y
- NFC use case and product webinars
- Software and development kits for every NFC challenge
- Design files of development kits online

Today's session

- Independent Design Houses certified by NXP/DHI partner

https://www.nxp.com/community/nfc
INTRODUCTION TO THE NFC READER LIBRARY
NFC Reader Library

Everything you need to create your own software stack and application for a contactless reader

- Modular
- Multi-layered
- ANSI-C language
- Portable to multiple design environments and platforms
- Free download

NFC Reader Library

Benefits

**Scalability:** Only required software components and protocol implementations need to be enabled, so the final application has a smaller footprint.

**Optimum performance:** Fine-tune your design with built-in MCU support, interrupt-based event handling, free RTOS support and compilers that produce highly compact and efficient code.

**Faster development:** Save time and effort by using the APIs and the rich set of sample applications for all the most common functions.

**Simpler certification:** Get ready for certification with test applications for EMVCO L1, NFC Forum and ISO/IEC10373-6 PICC/PCD
NFC Reader Library
The same software stack for PN512, CLRC663, PN5180 and PN7462 solutions

SW support for NFC frontend solutions

- PN512
- CLRC663
- PN5180
- PN7462
- OM25180FDK
- CLEV663B
- PNEV512B
NFC Reader Library

The same software stack for PN512, CLRC663, PN5180 and PN7462 solutions

SW support for NFC controllers with customizable firmware
NFC Reader Library
Support for multiple platforms

LPC MCUs
Kinetis MCUs
Linux OS
i.MX MCUs *

NFC Reader Library v4.040.03.001640 supports:
LPC1769/LPC11U68 connected to PNEV512B, CLEV663B and PNEV5180B boards
FRDM-K82 connected to PNEV512B, CLEV663B, PNEV5180B and PNEV7462B boards
Raspberry Pi connected to PNEV512B, CLEV663B, PNEV5180B and PNEV7462B boards

*Available soon
NFC Reader Library

Release contents

NFC Reader Library API

Software examples

Documentation
NFC Reader Library
Development environment

1. Choose an NFC product and pick its development board.
2. Download the corresponding NFC Reader Library package depending on your platform.
3. Install a development IDE.
4. Import software examples.
5. Evaluate functionality, re-use code and start developing.

Note: LPC-Link2 board may be required as a stand-alone debugger as well.
NFC Reader Library

Where to get it

Latest version and information available at:
http://www.nxp.com/pages/:NFC-READER-LIBRARY
NFC READER LIBRARY ARCHITECTURE
NFC Reader Library

Release contents

NFC Reader Library API

Software examples

Documentation

NXP NFC Reader Library API

NFC Reader Library

Release contents

Software examples

Documentation
NFC Reader Library

Architecture

- The NFC Reader Library is encapsulated into layers and components.
  - Each layer:
    - Consists of different components, having a generic interface and specific implementation
    - Is independent from the other layers
    - Is used as an entry point for the immediate upper layer
  - Provides a modular way of programming and setting up the reader interface.
NFC Reader Library

Common Layer

- Key handling component for storing crypto keys used in the authentication and encryption operations.
- Component for multiple card management according to ISO/IEC14443-4
- Utility component for CRC and parity calculations
- Component enabling extended logging
- Component which abstracts the OS simplifying development in multiple SW platforms
**NFC Reader Library**

**Common Layer**

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NFC Reader Library Common Layer</td>
<td>Component providing an API for easily porting the NFC Reader Library on a different host platform. Implementations included for LPC, Kinetis and Raspberry Pi</td>
</tr>
<tr>
<td>Component providing crypto random number generator implementations</td>
<td></td>
</tr>
<tr>
<td>Component providing symmetric crypto SW implementation</td>
<td></td>
</tr>
</tbody>
</table>
NFC Reader Library

Bus Abstraction Layer (BAL)

BAL layer implement the interface between host MCU and the NFC reader IC

BAL components offer functions to abstract the hardware parts of the host MCU

BAL abstract functions such as:

- `phbalReg_GetPortList()`
- `phbalReg_GetConfig()`
- `phbalReg_SetPort()`
- `phbalReg_SetConfig()`
- `phbalReg_OpenPort()`
- `phhalHw_GetConfig()`

...
HAL components abstract the functionality of the NFC reader IC to a generic interface.

HAL abstract functions such as:

- phhhalHw_FieldOn()
- phhhalHw_FieldOff()
- phhhalHw_Transmit()
- phhhalHw_SetConfig()
- phhhalHw_GetConfig()
NFC Reader Library
Protocol Abstraction Layer (PAL)

PAL components contain hardware-independent implementations of contactless protocols

PAL implements communication modes and data exchange regarding the contactless protocol.

PAL components support:
- MIFARE
- ISO/IEC14443-A
- ISO/IEC14443-B
- FeliCa

- ISO18092 initiator / target (P2P)
- ISOIEC15693
- ISO/IEC18000-3M3

PAL implements communication modes and data exchange regarding the contactless protocol.
NFC Reader Library

Application layer

AL components support:
- MIFARE Classic
- MIFARE Ultralight family
- FeliCa (partly)
- NFC Forum Tag operations (T1T, T2T, T3T, T4T and T5T)
- ICODE SLI
- ISO/IEC15693 commands
- ISO/IEC18000-3M3 commands
- Jewel /Topaz

AL contains application-specific implementations for various contactless cards (card command sets)

Additional MIFARE DESFire NDA patch for full feature set in NXP DocStore
NFC Reader Library

Activity layer

Discovery loop API is:

- `phacDiscLoop_GetConfig()`: Returns discovery loop configuration
- `phacDiscLoop_SetConfig()`: Configure the technologies to be detected
- `phacDiscLoop_Run()`: Starts the discovery loop
- `phacDiscLoop_ActivateCard()`: Activates a detected card

Discovery loop component implements a poll mode* and a listen mode** for contactless card and NFC device detection

* Based on NFC Forum Activity Specification v1.1 and EMVCo 2.5
** Based on NFC Forum Activity Specification v1.1
NFC Reader Library
HCE and P2P layers

HCE component implements the card emulation of NFC Forum Type 4A tag

NFC P2P package contains:
- The implementation of the LLCP protocol
- The implementation of the SNEP protocol
NFC Reader Library

Simplified API

High level abstraction of the NFC Reader Library.
- Focused on simplicity.
- Aimed to write NFC application with minimum code

Three profiles:
- NFC Forum channel: Transfer of NDEF messages
- EMVCo channel: Transfer of APDUs according to EMVCO L1 spec
- ISO channel: Raw data transfer
NFC Reader Library

Demo apps and reference applications

SW examples illustrating all operations and commands required in contactless applications such as:

- Reading / writing data to cards,
- data exchange with other NFC devices
- NFC Reader ICs emulating cards

SW examples illustrating the use of the Simplified API for:

- EMVCo channel profile
- ISO channel profile
USING THE NFC READER LIBRARY: BUILDING THE SW STACK
Using the NFC Reader Library
Building the SW stack

- To use the NFC Reader Library, a stack of components has to be initialized.

- Each layer generates a context data which is used as an entry point for the immediate upper layer.

- The component initialization must be built up by a *bottom to top* approach.
Using the NFC Reader Library
Building the SW stack (Example)

Which NFC modes do we need to support?
- The application requirements define which modules need to be enabled.
- NFC Read/Write, P2P and Card Emulation modes are supported by separate components.
- In the following slides, we will show an example with these parameters:
  - **MIFARE Classic** application
  - NFC Reader IC: **CLRC663**
  - Host controller: **LPC1769**
Using the NFC Reader Library
Building the SW stack (Example II)

1. Initialize the host controller interface (e.g. LPC1769)
2. Initialize the NFC reader IC (e.g. CLRC663)
3. Initialize the RF protocols (e.g. ISO14443-3 MIFARE)
4. Initialize discovery loop for card detection
5. Initialize the card command set (e.g. MIFARE Classic module)

Initialize the Key Store module (e.g. Crypto-1 keys)
Using the NFC Reader Library
Building the SW stack (Example III)

1. Host interface initialization generates a data context used to initialize and configure the upper layer (HAL)

   `phhalHw_Rc663_Init(…, void *pBalDataParams, …)`

2. NFC Reader IC initialization generates a data context used to initialize PAL and Discovery Loop components

   `phpalMifare_Sw_Init(…, void *pHalDataParams, …)`
   `phpanI14443p3a_Sw_Init(…, void *pHalDataParams, …)`
   `phacDiscLoop_Sw_Init(…, void *pHalDataParams, …)`

3. MIFARE PAL initialization generates a data context used to initialize MIFARE Classic component

   `phalMFc_Sw_Init(…, void *pPalMifareDataParams, …)`
Using the NFC Reader Library
Building the SW stack (Example IV) - Project build setup (ph_NxpBuild.h)

- The build setup and functionality is set in the file:
  ../intfs/ph_NxpBuild.h.

- This file defines the modules to be included into the build setup or to be excluded from the build setup.

- There is a specific macro defined for including / excluding each SW component.

- Components can be included / excluded depending on the application requirements or to optimize memory footprint.
USING THE NFC READER LIBRARY: DISCOVERY LOOP CONFIGURATION
Using the NFC Reader Library

Discovery loop

- It is a routine that sequentially sets the reader IC in different functional states and configurations so it can discover the presence of tags or NFC devices in the RF field.

- The discovery loop completes the following functionalities:
  - **Technology detection**: Detects whether there is another device or tag to communicate with and, if so, what technologies it supports.
  - **Collision resolution**: Detects the presence of multiple devices or tags and enumerates the different identifiers.
  - **Device activation**: Activates a particular device or tag to establish a communication.
Using the NFC Reader Library

Discovery loop – Configuration settings

phacDiscLoop_SetConfig(…, PHAC_DISCLOOP_CONFIG_PAS_POLL_TECH_CFG)

/** Configure technologies to be detected in passive poll mode */
#PHAC_DISCLOOP_POS_BIT_MASK_A -> Detect Type A.
#PHAC_DISCLOOP_POS_BIT_MASK_B -> Detect Type B.
#PHAC_DISCLOOP_POS_BIT_MASK_F212 -> Detect Type FELICA @ 212K
#PHAC_DISCLOOP_POS_BIT_MASK_F424 -> Detect Type FELICA @ 424K
#PHAC_DISCLOOP_POS_BIT_MASK_V -> Detect Type V / ISO 15693.
#PHAC_DISCLOOP_POS_BIT_MASK_18000P3M3 -> Detect ISO 18000p3m3

For scanning just Type A cards:
phacDiscLoop_SetConfig(…, PHAC_DISCLOOP_CONFIG_PAS_POLL_TECH_CFG, PHAC_DISCLOOP_POS_BIT_MASK_A)
Using the NFC Reader Library

Discovery loop – Configuration settings

```c
phacDiscLoop_SetConfig(...,
PHAC_DISCLOOP_CONFIG_ACT_POLL_TECH_CFG)
/* Configure technologies to be detected in passive poll mode */
#PHAC_DISCLOOP_ACT_POS_BIT_MASK_106 -> Detect active target @ 106K
#PHAC_DISCLOOP_ACT_POS_BIT_MASK_212 -> Detect active target @ 212K
#PHAC_DISCLOOP_ACT_POS_BIT_MASK_424 -> Detect active target @ 424K
```

Affected discovery modes for the macro
Using the NFC Reader Library
Discovery loop – Configuration settings

phacDiscLoop_SetConfig(..., PHAC_DISCLOOP_CONFIG_PAS_LIS_TECH_CFG)

/* Configure technologies to listen for in passive listen mode. */
#PHAC_DISCLOOP_POS_BIT_MASK_A -> Listen for Type A
#PHAC_DISCLOOP_POS_BIT_MASK_F212 -> Listen for Type FELICA @ 212K
#PHAC_DISCLOOP_POS_BIT_MASK_F424 -> Listen for Type FELICA @ 424K
#PHAC_DISCLOOP_POS_BIT_MASK_B -> [Not supported]
#PHAC_DISCLOOP_POS_BIT_MASK_V -> [Not supported]
#PHAC_DISCLOOP_POS_BIT_MASK_ISO18000_3M3 -> [Not supported]
Using the NFC Reader Library

Discovery loop - Start

```
phacDiscLoop_Run(pDataParams, PHAC_DISCLOOP_ENTRY_POINT_POLL);
```
Using the NFC Reader Library

Discovery loop – Find detected technologies

- Get information about the detected NFC tags or NFC devices.
  
  ```c
  phacDiscLoop_GetConfig(
  pDataParams,
  PHAC_DISCLOOP_CONFIG_TECH_DETECTED,
  &wTagsDetected);
  ```

- Macro to check that a Type A tag was detected
  
  ```c
  PHAC_DISCLOOP_CHECK_ANDMASK(
  wTagsDetected, PHAC_DISCLOOP_POS_BIT_MASK_A)
  ```

- Checking whether one particular NFC tag or NFC device has been detected can be done by using the defined bitmasks
  
  ```c
  PHAC_DISCLOOP_POS_BIT_MASK_A
  PHAC_DISCLOOP_POS_BIT_MASK_B
  PHAC_DISCLOOP_POS_BIT_MASK_F212
  PHAC_DISCLOOP_POS_BIT_MASK_F424
  PHAC_DISCLOOP_POS_BIT_MASK_V
  PHAC_DISCLOOP_POS_BIT_MASK_18000P3M3
  PHAC_DISCLOOP_ACT_POS_BIT_MASK_106
  PHAC_DISCLOOP_ACT_POS_BIT_MASK_212
  PHAC_DISCLOOP_ACT_POS_BIT_MASK_424
  ```
USING THE NFC READER LIBRARY: SIMPLIFIED API
Simplified API

Write NFC applications using as little code as possible

- Provides a high level abstraction access to the NFC Reader Library.
- It is focused on simplicity and aimed to write an NFC application with minimum code.
- It can be combined with the use of the “standard” NFC Reader Library stack API.

- Three different channels available:
  - **NFC Forum channel**: used to transfer NDEF messages.
  - **EMVCo channel**: used to transfer ISO 7816 APDU’s over the ISO 14443-4 protocol, according to EMVCo L1 spec.
  - **ISO channel**: general channel, used to transfer raw data transfer on block/frame level.
Simplified API
Write NFC applications using as less code as possible

1. Initialize the NFC Reader Library
   phNfcLib_Init();

2. Set the profile (ISO, EMVCo or NFC Forum)
   phNfcLib_SetConfig_Value(
   PH_NFCLIB_CONFIG_ACTIVATION_PROFILE,
   PH_NFCLIB_ACTIVATION_PROFILE_ISO);

3. Activate the card or peer
   phNfcLib_Activate(wTechnologyMask, &PeerInfo, NULL);

4. Send / Receive data
   phNfcLib_Transmit (...);
   phNfcLib_Receive (...);

5. Deactivate and release peer
   phNfcLib_Deactivate(PH_NFCLIB_DEACTIVATION_MODE_RELEASE,
   &PeerInfo);

6. De-initialize and reset internal structures
   phNfcLib_DeInit();

Profiles
- NFC Forum
- EMVCo
- ISO
SOFTWARE EXAMPLES
NFC Reader Library

Release contents

NFC Reader Library API

Software examples

Documentation
Example 1: BasicDiscoveryLoop

**Functionality**

- Scans for NFC tags and NFC devices and displays the detected device information (such as UID, SAK, Product Type for MIFARE cards)
- If multiple technologies are detected, the example will select the first detected technology to resolve.

**Benefits**

- Shows how to poll for different technologies (NFC tag, P2P, HCE), detect and activate them.
- Shows the discovery loop initialization and configuration
- Shows the use of both POLL and LISTEN discovery loop modes

![Diagram of discovery loop process]

- Initializes BAL&HAL components
  - `phPlatform_Init()`
- Initializes PAL/AL and other components
  - `phApp_RdLibInit()`
- Get poll config
  - `phacDiscLoop_GetConfig()`
- Configure poll state and LPCD (if active)
  - `phacDiscLoop_SetConfig()`
- Start Discovery loop
  - `phacDiscLoop_Run()`
- Multiple detection
  - yes
  - Solve collision
  - no
- Activate and display info
Example 2: AdvancedDiscoveryLoop

**Functionality**

- Scans for NFC tags and NFC devices and displays the detected device information (such as UID, SAK, Product Type for MIFARE cards)
- If multiple technologies are detected, the example will select the first detected technology to resolve.

**Benefits**

- Shows how to poll for different technologies (NFC tag, P2P, HCE), detect and activate them.
- Shows the different configuration options of the discovery loop.
- Shows the use of both POLL and LISTEN discovery loop modes
Example 3: NFCForum

Functionality

- Scans for NFC tags and NFC devices and displays the detected device information (such as UID, SAK, Product Type for MIFARE cards)
- It implements an SNEP Server and SNEP client for NDEF data exchange between peers.

Benefits

- Shows how to implement the P2P mode and reader function to detect Type A, Type B, Type F, Type V and ISO18000-3M3 tags.
- Shows how to exchange NDEF messages between two NFC-enabled devices.
- Shows active and passive communication modes.
Example 4: MIFARE Classic

Functionality

• Detects MIFARE Classic cards and displays information like UID, SAK and ATQA.
• Performs a sector authentication, block read and block write operations.

Benefits

• Shows the discovery loop configuration for MIFARE Classic detection.
• Shows the usage of standard MIFARE Classic commands (Authenticate, Read, Write).
• Shows KeyStore use.

Diagram:

1. Initializes BAL&HAL components
   - phPlatform_Init()
2. Initializes PAL/AL and other components
   - phApp_RdLibInit()
3. Initializes Discovery loop and keyStore
4. Configure Discovery loop
   - phacDiscLoop_SetConfig()
5. Configure Discovery loop for Poll Mode
   - phacDiscLoop_SetConfig()
6. Start Discovery loop
   - phacDiscLoop_Run()
7. Detect and activate card
8. Operates with activated card
9. Display info

Benefits:

• Detects MIFARE Classic cards and displays information like UID, SAK and ATQA.
• Performs a sector authentication, block read and block write operations.
Example 5: ISO/IEC15693

**Functionality**

- Detects an ISO15693-compliant card and displays tag type and tag UID information.
- Performs a block read and block write operations.

**Benefits**

- Shows the discovery loop configuration for ISO15693-compliant tag detection.
- Shows the usage of the most important commands for working with ISO15693-compliant tags.
Example 6: EMVCo Loopback

**Functionality**
- Implements a loopback application which is used to perform EMVCo 2.5 (L1) digital compliance validation.
- It can be tested with payment cards

**Benefits**
- It can be used to conduct EMVCo Level 1 PCD certification

Diagram:
- Initializes BAL&HAL components
  `phPlatform_Init()`
- Initializes PAL/AL and other components
  `phApp_RdLibInit()`
- Load EMVCo default settings
  `LoadEmvcoSettings()`
- Configure Discovery loop for Poll Mode
  `phacDiscLoop_SetConfig()`
- Start Discovery loop
  `phacDiscLoop_Run()`
- Detect card
- Get parameters of detected technology
  `phacDiscLoop_GetConfig()`
- EMVCo data exchange

- Load EMVCo default settings
- Configure Discovery loop for Poll Mode
- Start Discovery loop
- Detect card
- Get parameters of detected technology
- EMVCo data exchange
Example 7: EMVCo Polling

Functionality

- Detects an EMVCo payment cards.
- It displays the response of the detected EMVCo card after the select PPSE command.

Benefits

- Shows the discovery loop configuration for EMVCo payment card detection.
- Shows the exchange of APDU commands with EMVCo payment cards.
Example 8: HCE T4T

**Functionality**

- Implements a Type 4-A tag card emulation with a default NDEF message stored on its memory.
- It supports all specified commands such as Select, ReadBinary, UpdateBinary, etc.
- It allows both reading and writing and NDEF messages into the emulated card.

**Benefits**

- Shows how to emulate a Type 4 tag and to perform read and write operations.
- Shows how to set NDEF data into a Type 4 tag.
- It can be used to test against ISO10373-6 PICC test specification.

```plaintext
Initializes BAL&HAL components
phPlatform_Init()

Initializes PAL/AL and other components
phApp_RdLibInit()

Configure CE parameters and semaphores
phApp_Initialize()

Start Discovery loop in listen mode
phacDiscLoop_Run()

Detect card

Activate ISO 14443p4mC
phpalI14443p4mC_Activate()

Activate HCE, perform all functionalities
phceT4T_Activate()

Reset Reader Library layers
phpalI14443p4mC_ResetProtocol()
```
Example 9: NTAG I²C

**Functionality**
- Detects an NTAG I²C tag and displays information like UID, ATQA, SAK and version.
- Performs a SECTOR_SELECT, page READ and page WRITE operation.

**Benefits**
- Shows the discovery loop configuration for NTAG I²C tag detection.
- Shows the use of commands to communicate with NTAG I²C product.
Example 10: Simplified API ISO

**Functionality**


**Benefits**

- Shows how to use the Simplified API with an ISO profile.
Example 11: Simplified API EMVCo

**Functionality**
- Implements a loopback application using the Simplified API to perform EMVCo 2.5 (L1) digital compliance validation.

**Benefits**
- Shows how to use the Simplified API with an EMVCo profile.
- It can be used to conduct EMVCo Level 1 PCD certification
DOCUMENTATION
NFC Reader Library
Release contents

NFC Reader Library API

Software examples

Documentation
The NFC Reader Library reference API is also delivered in HTML format.

It is the most friendly way to go through the NFC Reader Library API.

It is generated from source code annotations.

It can be found in: "NxpRdLib\docs\14_user_doc"
NFC Reader Library reference API documentation

Browse the module API

Explore the API

Get detailed function description

Function Documentation

Generated by doxygen 1.8.11
TO SUM UP
NFC Reader Library

In a nutshell

- It is the SW support for NXP’s NFC frontends and NFC controllers with customizable firmware

- A modular, free, multi-layer software library written in ANSI-C that provides all the APIs needed to complete a design and prepare it for certification

- Easily portable to multiple design environments and platforms

- Complemented with software examples illustrating typical use cases.

Latest version and information available at:
http://www.nxp.com/pages:/NFC-READER-LIBRARY
Software development in Android and iOS
Embedded software for MCUs
JCOP, Java Card operating Systems
Hardware design and development
Digital, analog, sensor acquisition, power management
Wireless communications WiFi, ZigBee, Bluetooth, BLE
Contactless antenna RF design, evaluation and testing

MIFARE applications
End-to-end systems, readers and card-related designs
EMVco applications
Readers, cards, design for test compliancy (including PCI)
Secure Element management
GlobalPlatform compliant backend solutions
Secure services provisioning OTA, TSM services

We help companies leverage the mobile and contactless revolution
Design and implement NFC applications

Session 4: NFC Reader Library- SW support for NFC frontends and NFC controllers w. custom. FW solutions

Jordi Jofre (Speaker)
Angela Gemio (Host)

Thank you for your kind attention!

Please remember to fill out our evaluation survey (pop-up)

Check your email for material download and on-demand video addresses

Please check NXP and MobileKnowledge websites for upcoming webinars and training sessions

www.themobileknowledge.com/content/knowledge-catalog-0