AN11269

Software Design Guide for POS Development Kit OM5597/RD2663

Rev. 1.0 — 5 August 2014
242510

Application note
COMPANY PUBLIC

Document information

Info	Content
Keywords	RC663, TDA8026, LPC1768, Point of Sale Development Kit design, POS, guide, firmware documentation, software documentation, POS firmware architecture, Demo application architecture overview
Abstract	This document provides a comprehensive architectural overview of the Point of Sales Development Kit (POS DK).

Revision history

Rev	Date	Description
1.0	20140805	First release

Contact information

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

For sales office addresses, please send an email to: salesaddresses@nxp.com

1. Introduction

1.1 Scope

This document describes the firmware architecture of the POS Development Kit (POS DK) and gives guideline where to start for developing your own firmware.

For detailed information of how to set up a tool chain to build and apply the firmware to the POS DK please refer to [3].

2. Functional Overview

2.1 Block Diagram

The firmware is running on an LPC1768 Cortex M3 microcontroller.
As seen in 3.2.3 Fig 1 it is connected to

- a TDA8026 contact smart card reader IC over I2C [7]
- a CLRC663 contactless reader IC through the SPI interface [8]
- an LCD display through the SPI interface
- a PIN Pad over GPIOs

For debugging and firmware download the POS DK can be connected to a PC via the serial interface.

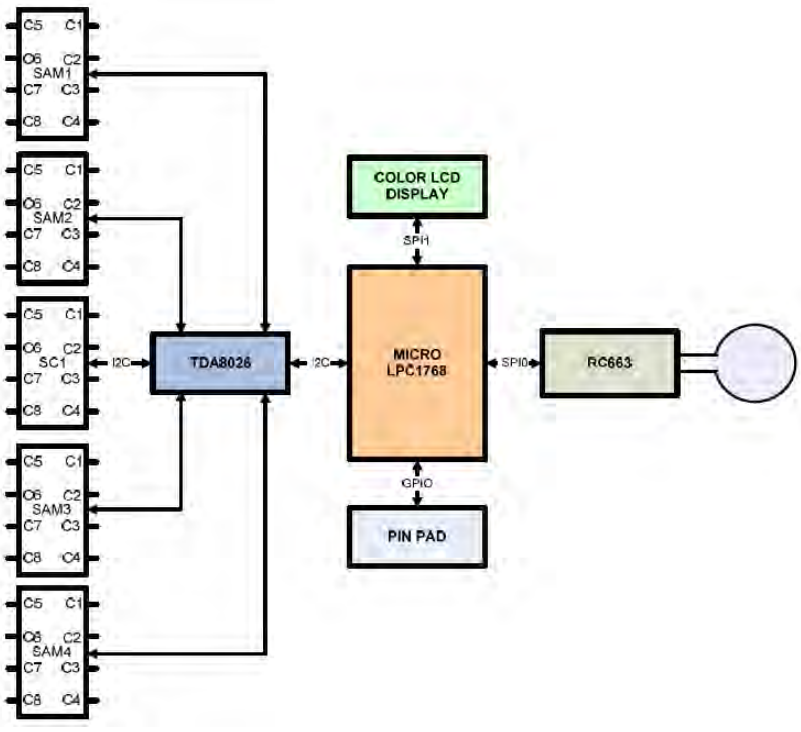

Fig 1. POS DK RD2663 Block diagram

2.2 Firmware Stack Overview

The following figure shows the different components of the POS DK firmware:

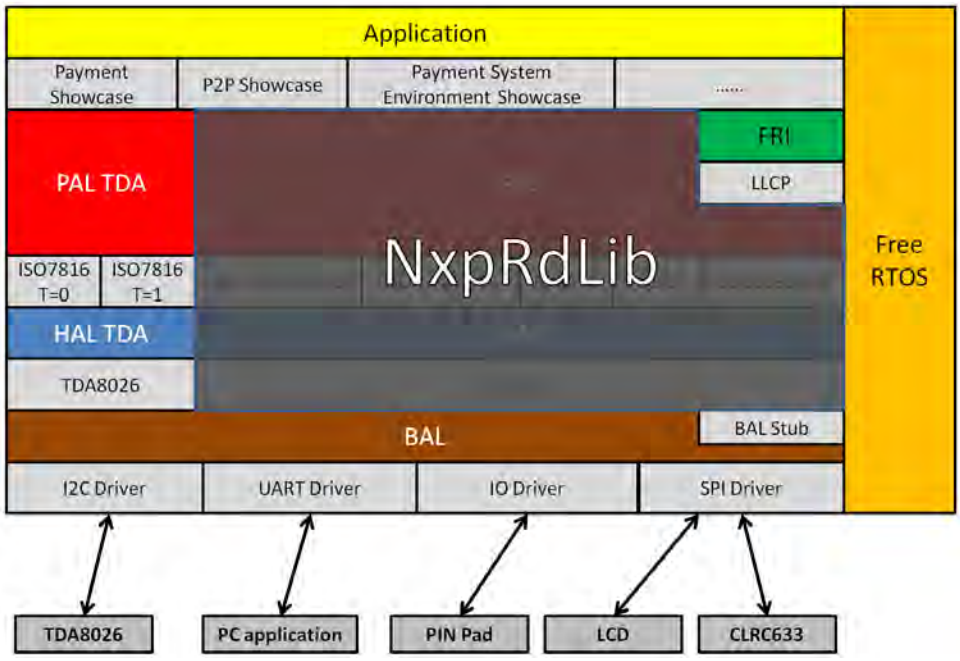

Fig 2. Firmware Stack Overview

The structure of the POS DK project and modules in this project can be seen in the picture below.

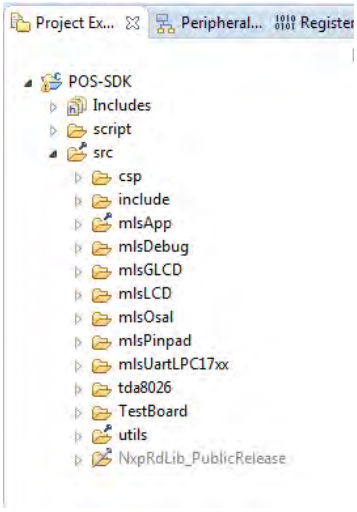

Fig 3. Project Structure

3. Contactless Stack

The contactless functionality of the POS DK is mainly based on the NXP Reader Library [11] (NxpRdLib) including some minor modifications for EMVCo compliance. These

modifications will be described in more detail in section 3.3. For enabling the a peer to peer (P2P) communication with an NFC mobile phone the Forum Reference Implementation (FRI) stack of the Android Operation system has been taken. This will be explained in more detail in section 3.4.

3.1 NXP Reader Library Software Architecture

The software for the contactless functionality of NXP NFC readers is based on the NXP Reader Library. It intends to be simple, modular, easily readable and quickly portable by all the customers.

The NXP Reader Library is a modular software library written in ANSI C language, which provides an API that enables customers to create their own software stack and applications for the NXP contactless reader ICs. This API facilitates the most common operations required in contactless applications such as reading or writing data into contactless cards or tags or exchanging data with other NFC-enabled devices.

The NXP Reader Library is designed as a versatile and multi-layered architecture. The layered structure of the library can be seen on the picture below.

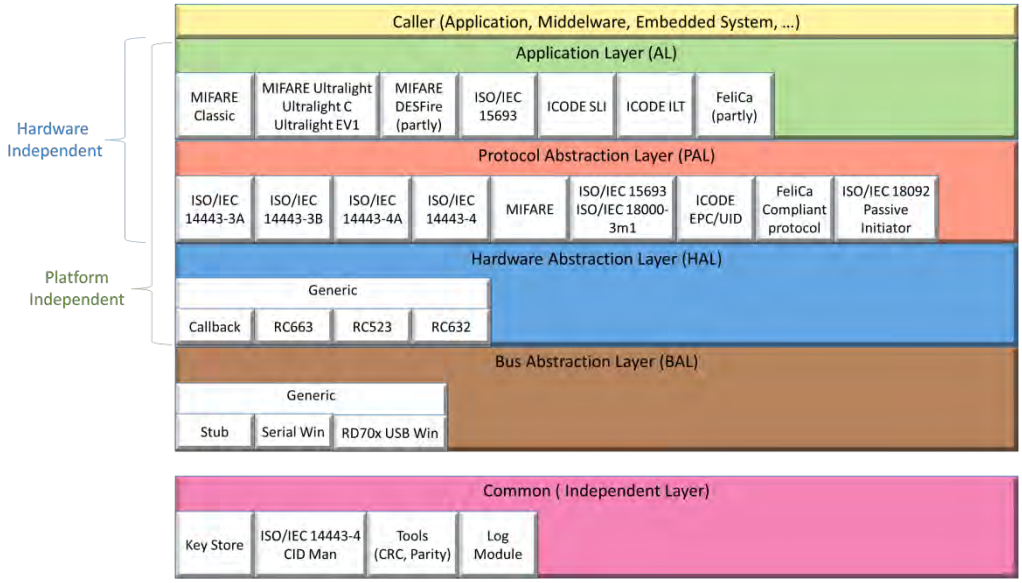

Fig 4. Layered Structure of the NXP Reader Library

3.2 NXP Reader Library Software Stack

The main advantage provided by this modular and multi-layered approach is flexibility. The Application Layer (AL) and the Protocol Abstraction Layer (PAL) are hardware-independent. This means that their functionality is not bound to or dependent on any specific hardware. Therefore, the developers can use them seamlessly on top of any of the supported contactless reader ICs implemented on the Hardware Abstraction Layer (HAL).

Similarly, the Application Layer (AL), the Protocol Abstraction Layer (PAL) and the Hardware Abstraction Layer (HAL) are also platform-independent. This means that their functionality is not dependent to any specific underlying communication interface with the

host. Therefore, the developers can use them seamlessly with any communication interface supported in the Bus Abstraction Layer (BAL).

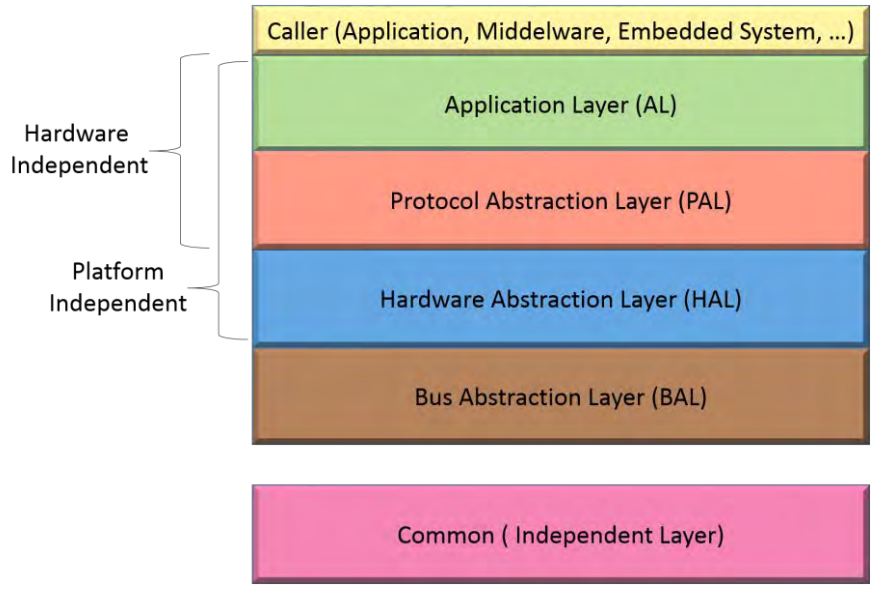

Fig 5. Hardware and Platform Independent Layers of the NXP Reader Library

3.2.1 Bus Abstraction Layer

The Bus Abstraction Layer implements the communication interface between the host device and the contactless reader IC. The master device can be a PC with Windows or Linux platform installed or MCU and it sends to the reader IC specific commands and generic commands containing addresses and data bytes. The reader IC responds to the host with data received from contactless cards or related information in requested registers.

The generic NXP Reader Library supports following communication interfaces:

- **SerialWin:** serial connection for Windows platform
- **Rd70x USB Win:** drivers for Windows platform to enable connection to Pegoda reader
- **PcscWin:** driver for PC/SC interface running on Windows platform
- **Stub:** Originally it was intended like component without functionality to ease implementation of additional busses. Currently it supports SPI, I2C and RS232 interfaces enabling connection to the Blueboard or Xpresso board.

For the POS DK the SPI interface is used due to EMVCo timing considerations especially EMD error handling.

3.2.2 Hardware Abstraction Layer

The Hardware Abstraction Layer (HAL) is responsible for the configuration and the execution of native commands of a particular contactless reader IC. These functions are mainly:

- Reading and writing from and into the reader's registers.
- RF field management, receiver and transmitter configuration.
- Timers configuration.
- Resolving interrupt sources from the reader chip.
- FIFO management.

The generic NXP Reader Library currently supports the following contactless readers:

- **PN512, MFRC523:** Highly integrated reader ICs supporting ISO/IEC 14443 Type A, ISO/IEC 14443 Type B, FeliCa and ISO/IEC 18092.
- **CLRC663**: Highly integrated reader IC with the highest RF output power fronted supporting ISO/IEC 14443 Type A and Type B, FeliCa and Passive Initiator mode according to ISO/IEC 18092; and its derivatives (MFRC631, MFRC630, SLRC610).

The NXP Reader Library is built in a way where upper layers are hardware independent. However, the developer must take into account the NFC capabilities of the selected NFC reader IC. For instance, the CLRC663 reader IC only supports passive communication mode whereas PN512 reader IC supports both active and passive communication modes.

For the POS DK the CLRC663 is used and the according to that, implementation of the NXP Reader Library is tailored to support this reader IC only.

3.2.3 Protocol Abstraction Layer

The protocol abstraction layer inherits hardware-independent implementation of the contactless protocol to be used for the communication.The generic NXP Reader Library supports the following ISO/IEC contactless standards protocols:

- **ISO14443-3A**: Contactless Proximity card air interface communication at 13.56MHz for the Type A and Jewel contactless cards.
- **ISO14443-3B:** Contactless Proximity card air interface communication at 13.56MHz for the Type B contactless cards.
- **ISO14443-4:** Specifies a half-duplex block transmission protocol featuring the special needs of a contactless environment and defines the activation and deactivation sequence of the protocol.
- **ISO14443-4A**: Transmission protocol for Type A contactless cards.
- **MIFARE (R):** Contains support for MIFARE authentication and data exchange.
- **FeliCa** (JIS: X6319): Contactless RFID smart card system from Sony.
- **ISO/IEC 18092 Initiator**: NFC Interface and Protocol standard that enables NFC Data Exchange protocol. Component for devices acting as communication initiators, which implies RF field generation and transmission of communication establishment request. Only passive mode is supported.
- **ISO/IEC 15693:** Contactless Vicinity card air interface communication at the 13.56 MHz frequency, with a maximum operating distance of 1 to 1.5 meters.

According to the Demo Application for the POS DK the ISO14443-3A, ISO14443-3B, ISO14443-4, ISO14443-4A, ISO18092 and FeliCa modules are used.

For the commercial solution any module from the protocol abstraction layer can be implemented. For more in-depth information on this aspect of implementation, please refer to the documents UM10721 [12] and UM10802 [13].

3.2.4 Application layer

The application layer implements the commands of contactless smart cards.The Application Layer enables the developer to access a particular card API by using its command set (e.g. reading, writing, modifying a sector etc.). The generic contactless card APIs provided is the following:

- **MIFARE Classic**: the leading industry standard for contactless and dual interface smart card schemes, with an immense worldwide installed base. The platform offers a full range of compatible contactless smartcard and reader ICs, as well as dual-interfaces ICs. The MIFARE Classic family covers contactless smart cards used in applications like public transport, access management, loyalty cards and many more. MIFARE Classic is fully compliant with ISO/IEC 14443 Type A up to layer 3 and available with 1k and 4k memory and 7 Byte as well as 4 Byte UIDs.
- **MIFARE Ultralight EV1**:It is intended for use with single trip tickets in public transportation networks, loyalty cards or day passes for events as a replacement for conventional ticketing solutions such as paper tickets, magnetic stripe tickets or coins. The mechanical and electrical specifications of MIFARE Ultralight are tailored to meet the requirements of paper ticket manufacturers. It can be easily integrated into existing contactless system without need for serious changes of the system. MIFARE Ultralight is fully compliant with ISO/IEC 14443 Type A up to layer 3
- **MIFARE Plus**: Migrate classic contactless smart card systems to the next security level. After the security upgrade, MIFARE Plus uses AES-128 (Advanced Encryption Standard) for authentication, data integrity and encryption. MIFARE Plus is based on open global standards for both air interface and cryptographic methods at the highest security level.
- **MIFARE DESFire**: Fully compliant with ISO/IEC14443A (part 1 - 4) and uses optional ISO/IEC7816-4 commands. The selectable cryptographic methods include 2KTDES, 3KTDES and AES128. The highly secure microcontroller based IC is certified with Common Criteria EAL4+. MIFARE DESFire is multi-application smart card used in public transport schemes, access management or closed-loop e-payment applications. It fulfils the requirements for fast and highly secure data transmission, flexible memory organization and interoperability with existing infrastructure.
- **ISO/IEC15693**: Contactless vicinity card defined by ISO/IEC Standard.
- **ICODE SLI**: The first member of a product family of smart label ICs based on ISO/IEC15693. This IC is dedicated for intelligent label applications like supply chain management as well as baggage and parcel identification in airline business and mail service.
- **Felica**: Contactless smart card developed by the Sony Company with usage spread in Japan.
- **ICODE ILT**: Dedicated chip for passive, intelligent tags and labels supporting the ISO18000-3 mode 3 RFID standard. It is especially suited for applications where reliable identification and high anti-collision rates are required. The ICODE ILT supports ISO/IEC18000-3mode3 RFID standard.

According to the Demo Application for the POS DK no module is used.
For the commercial solution any module from the application abstraction layer can be implemented. For more in-depth information on this aspect of implementation, please refer to the documents UM10721 [12] and UM10802 [13].

3.2.5 Common Layer

The NXP Reader Library includes a set of utilities which are grouped and encapsulated together in an independent layer called Common Layer. These utilities are not bound to any specific card or hardware, and as such they are functional regardless of the reader IC used. The modules implemented in the Common Layer are the following:

- **Tools**: This module provides 5, 8, 16 and 32 bit length CRC software calculation in addition to the parity encoding and decoding.
- **Key Store**: Key handling software module for storing cryptographic keys used in the authentication and encryption operations. Only the NFC Reader Library Export Controlled version supports high secure key storage capabilities.
- **ISO14443-4 CID Manager**: This module is used when a CID needs to be assigned to an ISO/IEC 14443-4 PICC or a CID is released by the PICC.
- **Log**: Useful module during debugging phase which enables a software tracing mechanism that records information about components during project execution in order to show them on the screen or store them to a file.
- **OSAL utils**: This module provides an API for timer and memory management related applications in a software and hardware independent way for an easier and quicker development.

According to the Demo Application for the POS DK no module is used.
For the commercial solution any module from the common abstraction layer can be implemented. For more in-depth information on this aspect of implementation, please refer to the documents UM10721 [12] and UM10802 [13].

3.3 Differences to the Library used in the POS DK Firmware

EMVCo specifications have certain requirements, which differ from ISO/IEC 14443.

Some of these requirements are:

- Only one PICC is allowed to be in the operating field (hence no anticollision required)
- Different Error codes and handling compared to ISO/IEC 14443
- Different FDT values compared to ISO/IEC 14443
- EMD error handling according EMVCo specification

Due to those requirements (EMVCo 2.3.1 L1 digital [10]), certain parts of the original NXP Reader Library had to be modified.

Please refer to [4] for more information how to configure the NXP Reader Library.

3.4 NFC Functionality

NF functionality implements the NFC Forum standardized protocol stack for Peer to Peer communication with a NFC device. The NFC P2P package functionalities include the correct management of the logical link between peers – according to LLCP protocol.

Logical Link Control Protocol (LLCP): LLCP is a link protocol layer that specifies the procedural means for transferring of upper layer information units between two NFC devices. It defines the logical link management and the synchronous exchange of data between peers in a connection-oriented or connectionless manner.

Please note: The implication of this specific software part is for demonstration purposes only and shall not be seen as a solution for a commercial design. For more in-depth information on this aspect of implementation, please refer to the documents UM10721 [12] and UM10802 [13].

4. Contact Stack

For communication with the contact smart card the POS DK firmware uses the ISO/IEC 7816 Protocol. This is implemented in the PAL TDA located at /tda8026/Prot7816.

This folder also contains the HAL TDA for controlling the TD8026 mainly in the modules

/tda8026/Prot7816/Config.c

/tda8026/Prot7816/TD8026.c

/tda8026/Prot7816/TD8026.h

/tda8026/Prot7816/tda_i2c_control.c

The communication between TDA and LPC controller is established via software UART interface (emulated using LPC1768 GPIO pins and timer) implemented in /tda8026/SW7816UART.

5. Bus Abstraction Layer and Microcontroller

The POS DK uses an LPC1768 microcontroller. Therefore this layer abstracts the hardware of the microcontroller and is located at /cps.

Based on this layer, the communication routines for the interface I2C, SPI and GPIOs can be managed.

Parts of the UART interface abstraction for is also located at /mlsUartLPC17xx

The POS DK implementation only uses SPI interface for communication to the CLRC663. For communication to the TDA the UART interface is used.

6. Operating System

The POS DK is based on the FreeRTOS™ operation system V6.0.0 which is configured to run on the LPC1768 microcontroller. The corresponding files are located at mlsOsal/FreeRTOS/.

For more information on FreeRTOS™ please refer to [9].

7. Application Layer

7.1 Demo Application

On top of the contact and contactless stack the demo application is implemented.

It is based on a state machine design and is located at /mlsApp/.
The state machine calls the functions of /utils/src/NFC.c which provides the high level control of the RC663 and connects to the NXP Reader Library.

7.1.1 Tasks Description

The demo application s made on several tasks running in parallel in the microcontroller:

- The Main task is only dedicated to the boot execution and the creation of the Foreground task
- The Foreground task (which create other tasks) is responsible for managing the LCD and handles the application state machine
- The Background task is responsible for executing card communication related actions (contact or contactless)
- The Card Detection task is responsible for contact and/or contactless card discovery as well as initializing the P2P communication over LLCP
- The Pinpad task is responsible for detecting when a key is pressed

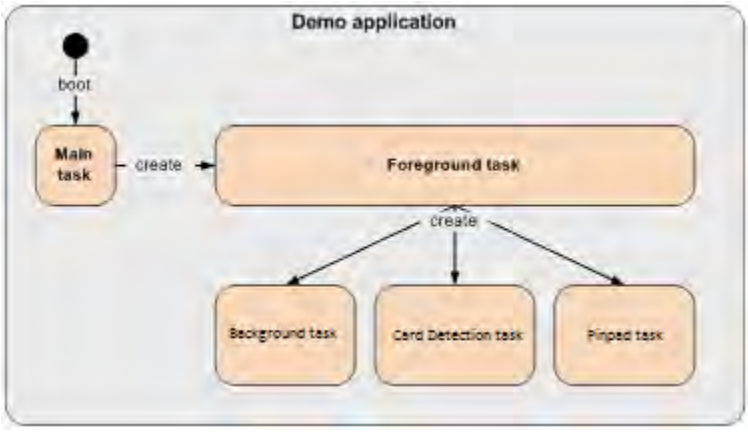

Fig 6. Demo Application Tasks

Tasks communicate together using a message mechanism based on messaging queues:

- The Foreground task indicates to the Background task which action needs to be performed according to the current application state.

- The Background task indicates to the Foreground task when the action completes.
- The Card Detection task indicates to the Foreground task when a card has been discovered and P2P data to be displayed at the LCD.
- The Pinpad task forwards to the Foreground task information entered by the user on the pin pad

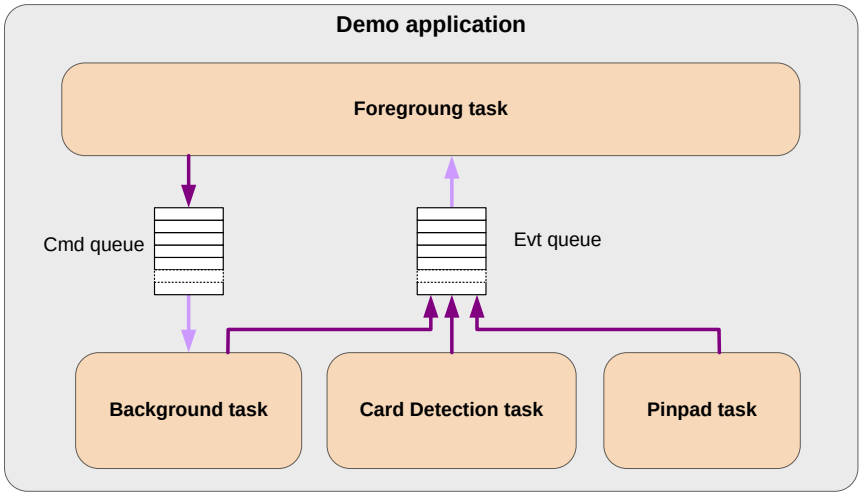

Fig 7. Tasks Communication Scheme

7.1.2 Application State Machine

The state machine is handled by *mlsTUIForgroundProcess* function (*mlsTUIMan.c* file).

Each state is linked to:

- A “Draw” function (*mlsTUI**State**_Draw)* which updates the LCD screen, according the current state, and triggers the execution of the related action.
- An “Execute” function (*mlsTUI**State**_Execute*) which sets the next state according to the current event received.

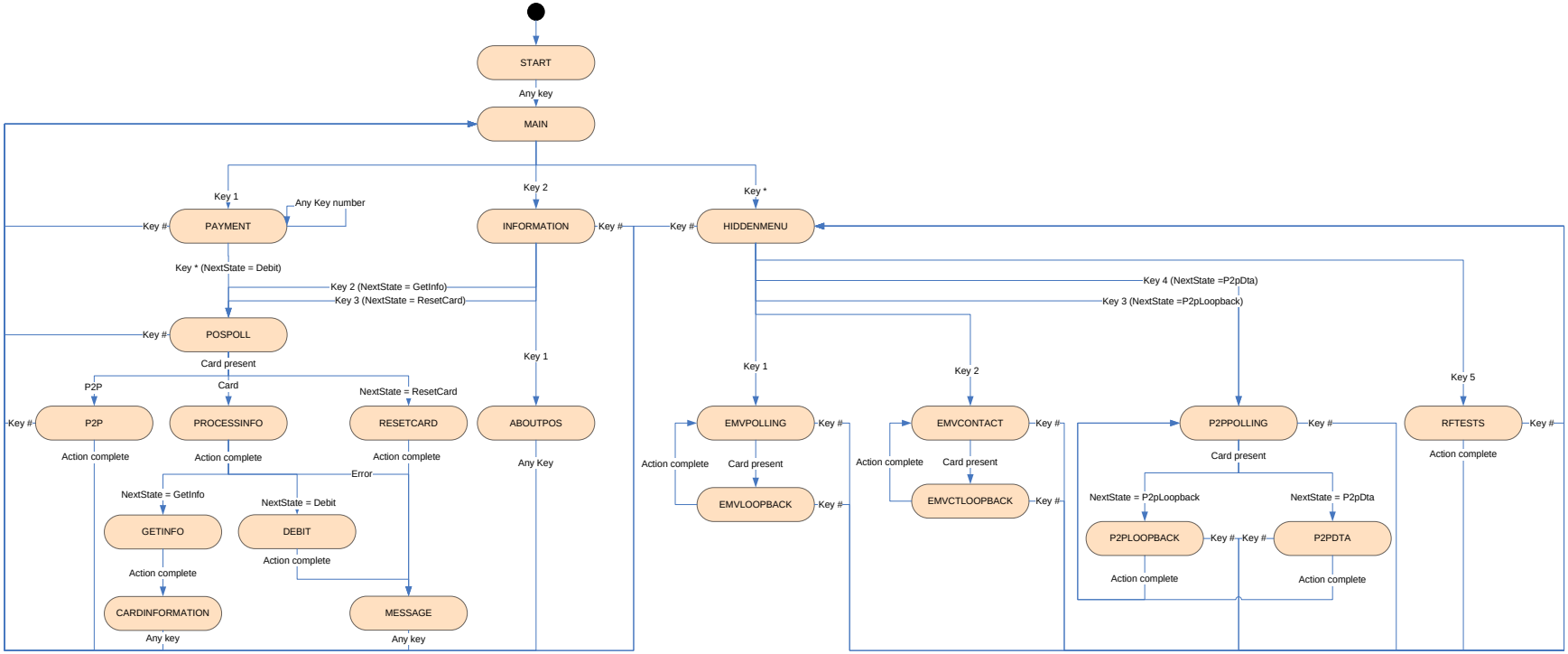

Fig 8. Application State Machine

8. Source Code Overview

8.1 Component List

The figure below shows the directory structure of the firmware components located in the src subfolder.

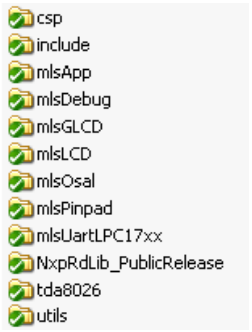

Fig 9. Files Directory Structure

8.1.1 csp

This component provides low level control of the LPC1768 Core peripherals (timer, I2C, SPI…).

8.1.2 mlsApp

This component is the demonstration application as described in section 7.1. It contains the following files

mlsMain.c

This component provides the function **`mlsMainSystemInit()`** which is called at system starup.

mlsCardUtils.c

Provides the functionality triggered by the tasks and inputs including the handling for the opportunities provided by the RfSettingsMenu.

mlsTUIMan.c

This is the main control module for the demonstration application state machine. Here the tasks of the application are generated and handled. For more information on the tasks please refer to section 7.1.

mlsTUI……c

The other modules starting with mlsTUI… control the displayed screens of the individual menus and Items, the possible interactions and state machine transitions.

8.1.3 mlsDebug

This component provides debug functionality to the demonstration application. This contains functions to send the debug info over UART or control the GPIOs.

8.1.4 mlsGLCD

This component provides low level control of the graphical color LCD display.

Two different LCD types can be used with the POS DK. They differ in the way to access them.

The file GLCD.c provides the two drivers, with a compilation option:

```
#define LCD_RM68050   (0)    // For LCD PCB before 2011 05 23
#define LCD_HX8347D   (1)    // For LCD PCB after 2011 05 23
#define LCD_CONTROL   LCD_HX8347D
```

Fig 10. LCD Type – Compiler Option

The OM5597/RD2663 comes with the LCD_HX8347D display which is selected as default LCD. The LCD_RM68050 is not used any more.

8.1.5 mlsLCD

Not used.

8.1.6 mlsOsal

This component is made of an OS (FreeRTOS) for the LPC1768, an OS abstraction layer and a memory management module.

8.1.7 mlsPinpad

This component provides low level control of the pinpad.

8.1.8 NxpRdLib_PublicRelease

This type of the NXP Reader Library is the public release and it provides low level control as well as contactless (ISO14443, ISO18092, MIFARE…) protocols support of all NXP reader ICs (RC663, RC570, etc...). To reduce the memory footprint, for this project, only the functionalities of the RC663 are enabled, the others disabled. And as described in Section 3.3 adaptations to achieve EMV L1 digital compliancy are included. For more information on the NxpRdLib please refer to [5] and [6].

8.1.9 mlsUartLPC17xx

This component provides low level control of the LPC1768 UART.

8.1.10 tda8026

This component provides low level control of ISO7816 UART (emulated using LPC1768 GPIO pins and timer) as well as ISO7816 protocol support.

8.1.11 Utils

This is a directory which contains the following components. These modules provide the lower functionality of the application layer to control the POS DK.

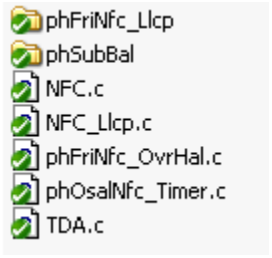

Fig 11. Contents of the directory utils

phFriNfc_Llcp

As described in section 3.4 this component is the logical link control protocol (LLCP) coming from the Android open source. It proposes functions allowing the peer-to-peer communications with a mobile phone.

phSubBal

Considered as sub BAL (Bus Abstraction Layer) layer, it provides the low level functions for communication between the micro LPC1768 the RC663 IC. Only SPI functions are proposed.

NFC.c

This component provides the high level control of the RC663, contactless (ISO14443, ISO18092, MIFARE…) protocols, EMVCo contactless loopback.

The most important functions in here are:

- **`void NFC_Initialize(void)`**

  Initializes and configures the reader IC.

- **`UInt8 NFC_Detect(UInt8 TypeMask)`**

  Performs the actual polling for contactless cards or NFC targets.

phFriNfc_OvrHal.c

This module provides an over Hal layer functions needed for peer-to-peer communications over llcp.

phOsalNfc_Timer.c

This module provides timer functionalities needed for peer-to-peer communications over llcp ptotocol.

TDA.c

This module provides the configuration and initialization functionality for the TDA with the following functions

- **`void TDA_Initialize(void)`**
  Starts initialization of the TDA

- **`UInt8 TDA_Detect(void)`**
  Checks if TDA if able to select and pre-power up a card in slot1

- **`static void TDADriverInit(void)`**
  Initializes the TDA driver

- **`static void SW7816UartInit(void)'`**
  Initializes the SW7816Uart

- **`static void GPIOInit(void)`**
  Enables the GPIO ports to act as UART interface to the TDA

- **`static void SW7816UartIrqHandler (void)`**
  Initializes the SW7816UART IRQ Handler

- **`void EINT2_IRQHandler(void)`**
  Initializes the EINT2 IRQ Handler

9. References

[1] **Application Note** AN11268: Quick Startup Guide for POS DK OM5597/RD2663, http://www.nxp.com/documents/application_note/AN11268.pdf

[2] **Application Note** AN11270: Hardware Design Guide for POS DK OM5597/RD2663, http://www.nxp.com/documents/application_note/AN11270.pdf

[3] **Application Note** AN11271: Toolchain Information for POS DK OM5597/RD2663, http://www.nxp.com/documents/application_note/AN11271.pdf

[4] **Application Note** AN11021: CLRC663, MFRC631, MFRC630, SLRC610 Software Design Guide for NXPRdLib, http://www.nxp.com/documents/application_note/AN11021.pdf

[5] **Manual** NXP Reader Library (in .chm format)

[6] **Application Note** AN111021: CLRC663, MFRC631, MFRC630, SLRC610 Software Design Guide for NXPRdLib http://www.nxp.com/documents/application_note/AN11021.pdf

[7] **Datasheet** TDA8026 Multiple smart card slot interface IC http://www.nxp.com/documents/data_sheet/TDA8026.pdf

[8] **Datasheet** CLRC663 Contactless reader IC http://www.nxp.com/documents/data_sheet/CLRC663.pdf

[9] **Webpage** FreeRTOS www.nxp.com/redirect/freertos.org/

[10] **Specification** EMV® Contactless Specifications for Payment Systems, Book D: EMV Contactless Communication Protocol Specification www.nxp.com/redirect/emvco.com/specifications

[11] **Software** NXP Reader Library http://www.nxp.com/documents/software/200312.zip

[12] **User Manual** UM10721 – NXP NFC Reader Library User Manual http://www.nxp.com/documents/software/SW297831.zip

[13] **User Manual** UM10802 – NXP NFC Reader Library API http://www.nxp.com/documents/software/SW297831.zip

10. Legal information

10.1 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.

10.2 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.

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 accepts 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.

10.3 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.

10.4 Trademarks

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

11. List of figures

12. Contents

For more information, visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 5 August 2014
242510
Document identifier: AN11269