# **UM10957**

PN7462AU Door Access User Manual

Rev. 1.1 — 13 September 2016 357911

User manual COMPANY PUBLIC

#### **Document information**

Info	Content
Keywords	Door Access, HSU, SAM, MIFARE DESFire EV1, Classic
Abstract	This document serves as a User Manual for Door Access Demo Application



**Revision history** 

Rev	Date	Description
1.1	20160913	Figures updated
1.0	20160308	First release

### **Contact information**

For more information, please visit: <a href="http://www.nxp.com">http://www.nxp.com</a>

UM10957

User manual COMPANY PUBLIC

### 1. Introduction

This document describes "Door Access application" and shows how PN7462AU Customer Demo board can be used in the door access management.

Two types of Door Access examples are available and explained here:

- Export controlled version,
- Non Export Controlled version

#### 1.1 Door Access Application – Non Export Controlled version

In this version of the application a MIFARE Classic card is used for the authentication and for data exchange the contactless interface is used.



The application is running a NFC polling loop and goes to standby mode after each polling loop in case no CL card is detected by the RF field of the reader. The polling loop is implemented for Type A, B, F, ISO15693 and ISO 1800-P3M3. The application prints out the detected type of the card and UID if available. In case the NFC device is detected, the application sends a NDEF message containing the NXP webpage address.

After each polling loop the application goes to standby mode and remains for 500 ms. A timer is used as a wakeup source from standby. This process continues until a card is detected by the RF field of the reader. If a card is detected, the card type with its UID is sent via HSU and will be printed on the PC console.

In case a MIFARE Classic card is detected, application tries to authenticate the card using the default MIFARE key. If the authentication is successful a block of data is read from the card. The type of the card, UID and data are sent via HSU and printed on the PC console.

Table 1

LEDs (Table 1) are used to indicate status or activity flow, like card detection, authentication pass/ fail, reading data pass/ fail. The application is using four LEDs on the demo board (green, yellow, red and blue).

Table 1. LEDs status specification		
	LED	
CL card detected	Yellow	
card authenticated	Yellow + green	+
Card authentication failed	Yellow + red	+
Transaction successful completed	Yellow + green blinking	+ blinking 3x
Transaction failed	Yellow + red blinking	+ blinking 3x
Polling	all 4 LEDs blink circular	

The reader can detect Type A/B, Felica, ISO15693 and ISO18000-3M3 cards.

P2P functionality is integrated in the Non Export controlled version. When a NFC enabled phone is detected from the RF field as an active or passive target the LLCP SNEP will be activated and NDEF message will be sent to the mobile device.

If LPCD (Low Power Card Detection) is enabled, the reader checks during the wakeup time the presence of a card and enters the card detected mode when a card is present & repeats the cycle. If no card is present it goes back to standby mode. LPCD is enabled by default.

#### 1.2 Door Access Application – Export Controlled version

This version of the application is using a MIFARE DESFire EV1 card for the authentication, data exchange is done over contactless interface and software key store or SAM (Secure Access Module) key store is used for storing the authentication key. By default the software key store is enabled. The user can use the SAM key store by enabling a Macro as explained in the section 3.5. SAM is a key storage element and it should be inserted in the CT main.



UM10957

Door Access UM

On power up, NFC Reader starts polling for (PICC) cards (Type A, B or F, ISO15693, ISO18000-3M3) and if no card is present, the reader goes to standby mode. Timer is used as a wakeup source from standby - timer periodically every 500 ms. This process continues until a card is detected by the RF field of the reader. If a card is detected, the card type with its UID is sent via HSU & this is printed on the PC console

If a MIFARE DESFire EV1 card is detected, the reader tries to select a pre written custom application on the card. It tries to authenticate the card using the key stored in SAM. If SAM is not present, then software keys can be used for authentication. If authentication is successful a block of data will be read from the card. Type of the vcard, UID along with the data read is sent via HSU & this is printed on the PC console.

LEDs (Table 2) are used to indicate status or activity flow, like card detection, authentication pass/ fail, reading data pass/ fail. Application is using four LEDs on the demo board (green, yellow, red and blue).

	LED	
CL card detected	Yellow	
card authenticated	Yellow + green	+
Card authentication failed	Yellow + red	+
card authenticated, using SAM	Yellow + blue + green	+ +
Card authentication failed, using SAM	Yellow + blue + red	+ +
Transaction successful completed	Yellow + green blinking	+ blinking 3x
Transaction failed	Yellow + red blinking	+ blinking 3x
Polling	all 4 LEDs blink circular	

#### Table 2. LEDs status specification

Reader is capable to detect Type A/B, Felica, ISO15693, and ISO18000-3M3 cards.

If LPCD (Low Power Card Detection) is enabled, the reader checks during the wakeup time the presence of a card and enters the card detected mode when a card is present & repeats the cycle. If no card is present it goes back to standby mode. LPCD is enabled by default.

#### 1.3 Development environment

To prepare the project and build the source code the components listed in the Table 3 are required.

Item	Version	Purpose
PN7462AU Customer Board	2.1	Engineering development board.
LPC-Link2	1.0	Stand-alone debug adapter
LPCXpresso IDE	8.0.0.526	Development IDE
LPCXpresso PN7462AU Plugin	com.nxp.pn7xxxx x.update-8.0.0- SNAPSHOT-150	Add PN7462AU reader to the LPCXpresso
LISPToSorial Convertor		

USB I oSerial Converto

JM1	0957

#### Note

Installation procedure of the *"LPCXpresso IDE" and "LPCXpresso PN7462AU Plugin"* is described in the *"UM10883 - PN7462AU Quick Start Guide - Customer Board"* document. Latest LPCXPresso plugin is required.

#### 1.4 Project download link

The source code of the Door Access application is part of a delivered "PN7462AU customer support package" and the project can be found in ".\PN7462AU Software" folder.

Note:

Export controlled version is available only by NXP Docstore. Docstore tool is online tool for sharing documents and data. Access to NXP Docstore is available only to customers with valid NDA.

#### 1.5 Driver Installation for USB to serial converter

Windows 7 driver is required for USB to serial converter and it is part of the delivered "PN7462AU customer support package" and the driver can be found in "Drivers" folder. <a href="https://www.nxp.com/external/prolific">www.nxp.com/external/prolific</a>

UM10957

### 2. Door Access Use Case – Non Export Controlled Version

This section describes in detail the setup procedure and execution environment required for Non-Export Controlled version of "Door Access Use Case application".

This version of the application is using MIFARE Classic card for the authentication and data exchange is done over contactless interface.

#### 2.1 Hardware Setup

For preparing the project and building the source code components listed in the Table 4 are required.

Fable 4.	<b>Development enviro</b>	nment	
ltem		Version	Purpose
PN7462AU	J Customer Board	2.1	Engineering development board.
LPC-Link2		1.0	Stand-alone debug adapter
USBToSer	ial converter		USB To Serial
Power ada	pter		Optional



#### **Connection Details:**

1) DC Power Supply of 7.5 V

Board is powered using the 7.5 V power adapter.

2) Power supply selection jumper

Jumper should be placed to connect JP41.1 -> JP41.2

This is to choose the external power supply to the board.

#### 3) Connect Rx of 'USB - Serial converter' to ATX\_B

Take a USB to Serial converter (USB to TTL to UART RS232 COM Cable Module Converter) as shown in the Fig 4.

One side of this cable has USB connector and other side has 4 wires.

- Vcc (Red wire)
- Tx (Green wire)
- Rx (White wire)
- Ground (Black wire)



Connect the RX line to the ATX\_B (jp32.3) pin of the Customer Board.

#### 4) Common Ground connection

Connect ground wire (Black wire) of the 'USB to Serial converter' to one of the common ground pins on the customer board as shown Fig 3.

#### 5) CTS to ground connection

Connect ATX\_D (JP32.7) pin of the Customer board to one of the common ground pins using a wire. This grounds the CTS line and this connection is necessary to print on the PC console using USB to serial converter.

#### 2.1.1 Hardware setup without any external supply

To eliminate the use of external power supply, we can power the board using Vcc line (+5 V) of USB to serial converter. For this, Vcc line (Red wire) of USB to serial converter should be connected to JP41.2 of the board. (Remember to remove any external power supply selection jumper already placed on JP41).



### 2.1.2 LPC-Link2 Debug Adapter Connection

To flash the binary or to debug the code LPC-Link2 debug adapter should be used. It can be connected to JTag connector (JP4) of the customer board as shown in the figure below.



#### 2.2 Importing project

After installation of the "PN7462AU FW & SW Examples Full Version" please follow steps described below.

The sequence of preparing the project is:

UM10957

**Door Access UM** 

- Open the LPCXpresso IDE and select an empty workspace
- Select the option "Import project(s)" in Quick Panel

<ul> <li>Import project(s)</li> <li>New project</li> <li>Build all projects []</li> <li>Build " []</li> <li>Clean " []</li> <li>Clean " []</li> <li>Clean " []</li> <li>Edit." project settings</li> <li>No consoles:</li> </ul>		Start here ×	<b>^</b>	
Image: Second	K	Import project(s)		
Build all projects []  Build " []  Clean " []  Debug " []  Edit." project settings  No consoles:  No consoles:		👸 New project	E	
Build " [] Clean " [] Clean " [] Clean " [] Chain " []		📆 Build all projects []		
Clean " []  Debug " []  Edit." project settings No consoles:		🔏 Build " []		
Debug "[]     Debug "[]     Debug "[]     Debug troject settings     No consoles:		🖌 Clean " []		📃 C 🔀 🚦
Edit " project settings <u>No consoles:</u>		🕸 Debug '' []		
XML Import project(s) from XML Descripti		🛞 Edit " project settings		No consoles
impore project(s) non xivie beschpt		Import project(s) from XML Descript	+	

The dialog for project import opens.

Г

Projects are contained within archives (xip) or are unpacked within a directory. Select your project archive or root directory and press <next>. On the next page, select those projects you wish to import, and press <finish>.         Project archives for LPCOpen and 'legacy' examples are provided.         Project archive (xip)         Archive C:\Users\nxp89422\NXP Semiconductors\PN7462AUPspPackageFull-v04_01_00.ft         Browse_n         Project directory (unpacked)         Root directory         Browse_n         LPCOpen         LPCOpen         LPCOpen is the recommended code base for Cortex-M based NXP LPC Microcontrollers.         LPCOpen         LPCOpen         LPCOpen is the recommended code base for Cortex-M based NXP LPC Microcontrollers.         LPCOpen is the recommended code base for Cortex-M based NXP LPC Microcontrollers.         LPCOpen resources on LPCWare.com website for latest resources.         Browse LPCOpen resources on LPCWare.com</finish></next>	Select the examples archive	e file to import.	
Project archives for LPCOpen and 'legacy' examples are provided.         Project archive (zip)         Archive       C\USers\nxp89422\NXP Semiconductors\PN7462AUPspPackageFull=v04_01_0010         Project directory (unpacked)         Root directory       Browse         LPCOpen       LPCOpen is the recommended code base for Cortex-M based NXP LPC Microcontrollers.         LPCOpen is the recommended code base for Cortex-M based NXP LPC Microcontrollers.         LPCXpresso includes the LPCOpen packages which can be imported directly by pressing the Browse button in the Project archive (zip) section, above, and navigating to the Examples/LPCOpen directory.         Alternatively, press the button below to Browse the LPCWare.com website for latest resources.         Browse LPCOpen resources on LPCWare.com         ?       < Back	Projects are contained within project archive or root direct wish to import, and press <	in archives (.zip) or are unpacked within a directory. Select tory and press <next>. On the next page, select those proj Finish&gt;.</next>	your ects you
Project archive (zip)         Archive       C\Users\nxp89422\NXP Semiconductors\PN7462AUPspPackageFull-v04_01_001         Project directory (unpacked)         Root directory       Browse         IPCOpen       IPCOpen is the recommended code base for Cortex-M based NXP LPC Microcontrollers.         LPCXpresso includes the LPCOpen packages which can be imported directly by pressing the Browse button in the Project archive (zip) section, above, and navigating to the Examples/LPCOpen directory.         Alternatively, press the button below to Browse the LPCWare.com website for latest resources.         Browse LPCOpen resources on LPCWare.com         ?       < Back	Project archives for LPCOpe	en and 'legacy' examples are provided.	
Archive       C:\Users\mp89422:\NXP Semiconductors\PN7462AUPspPackageFull-\04_01_001       Browse.         Project directory (unpacked)       Browse         Root directory       Browse         LPCOpen       Browse         LPCXpresso includes the LPCOpen packages which can be imported directly by pressing the Browse button in the Project archive (2p) section, above, and navigating to the Examples/LPCOpen directory.         Alternatively, press the button below to Browse the LPCWare.com website for latest resources.         Browse LPCOpen resources on LPCWare.com         ?         < Back	Project archive (zip)		
Project directory (unpacked)         Root directory         IPCOpen         IPCOpen is the recommended code base for Cortex-M based NXP LPC Microcontrollers.         LPCXpresso includes the LPCOpen packages which can be imported directly by pressing the Browse buttorn in the Project archive (2p) section, above, and navigating to the Examples/LPCOpen directory.         Alternatively, press the button below to Browse the LPCWare.com website for latest resources.         Browse LPCOpen resources on LPCWare.com.website for latest resources.         IPCOPEN         IPCOPEN         IPC	Archive C:\Users\nxp894	22\NXP Semiconductors\PN7462AUPspPackageFull-v04_0	1_00\F Browse
Root directory       Browse         LPCOpen       LPCOpen is the recommended code base for Cortex-M based NXP LPC Microcontrollers.         LPCXpresso includes the LPCOpen packages which can be imported directly by pressing the Browse button in the Project archive (2p) section, above, and navigating to the Examples/LPCOpen directory.         Alternatively, press the button below to Browse the LPCWare.com website for latest resources.         Browse LPCOpen resources on LPCWare.com         ?       < Back	Project directory (unpacks	ed)	$\smile$
IPCOpen         LPCOpen is the recommended code base for Cortex-M based NXP LPC Microcontrollers.         LPCXpresso includes the LPCOpen packages which can be imported directly by pressing the Browse button in the Project archive (ap) section, above, and navigating to the Examples/LPCOpen directory.         Alternatively, press the button below to Browse the LPCWare.com website for latest resources.         Browse LPCOpen resources on LPCWare.com         Image: Comparison of the state	Root directory		Browse
IPCOpen         IPCOpen is the recommended code base for Cortex-M based NXP LPC Microcontrollers.         IPCXpresso includes the LPCOpen packages which can be imported directly by pressing the Browse button in the Project archive (2p) section, above, and navigating to the Examples/LPCOpen directory.         Alternatively, press the button below to Browse the LPCWare.com website for latest resources.         Browse LPCOpen resources on LPCWare.com         ?       < Back			
PCCprease is the recommendee Code Date for Contexts and be imported directly by pressing the Browse button in the Project archive (2ip) section, above, and navigating to the Examples/JPCOpen directory. Alternatively, press the button below to Browse the LPCWare.com website for latest resources. Browse LPCOpen resources on LPCWare.com  C	LPCOpen		1
PCXpresso includes the LPCOpen packages which can be imported directly by pressing the Browse button in the Project archive (zip) section, above, and navigating to the Examples/LPCOpen directory. Alternatively, press the button below to Browse the LPCWare.com website for latest resources. Browse LPCOpen resources on LPCWare.com  C	LPCOpen is the recommen	nded code base for Contex-INI based INAP LPC Microcontrol	iers.
Alternatively, press the button below to Browse the LPCWare.com website for latest resources.         Browse LPCOpen resources on LPCWare.com	LPCXpresso includes the L button in the Project archi	.PCOpen packages which can be imported directly by press ive (zip) section, above, and navigating to the Examples/LP	COpen directory.
Browse LPCOpen resources on LPCWare.com)  (2) (3) (4) (4) (5) (5) (5) (5) (5) (5) (5) (5) (5) (5	Alternatively, press the bu	tton below to Browse the LPCWare.com website for latest r	esources.
(2) < Back Next> Einish Cancel	Browse LPCOpen resource	ces on LPCWare.com	
(			
(			
	?	< Back Next > Einish	Cancel

Browse to the project zip file ".\PN7462AU Software\PN7462AU-FW\_v04.01.00.zip" and click "Next".

import project(s) Select a directory to search for existing Eclipse projects.	
Projects:	
✓ FreeRTOS_Library (FreeRTOS_Library)         NfcrdlibEx1_BasicDiscoveryLoop (NfcrdlibEx1_BasicDiscoveryLoop)         NfcrdlibEx1_MIFAREDESFire (NfcrdlibEx1_MIFAREDESFire)         NfcrdlibEx2_AdvancedDiscoveryLoop (NfcrdlibEx2_AdvancedDiscoveryLoop)         NfcrdlibEx3_NFCForum (NfcrdlibEx3_NFCForum)         NfcrdlibEx4_MIFAREClassic (NfcrdlibEx4_MIFAREClassic)         NfcrdlibEx5_ISO15693 (NfcrdlibEx5_ISO15693)         NfcrdlibEx6_EMVCo_Loopback (NfcrdlibEx6_EMVCo_Loopback)         NfcrdlibEx8_MCC_TeNVCo_Polling (NfcrdlibEx8_MVCo_Polling)         NfcrdlibEx8_MCC_TeNVCo_Polling (NfcrdlibEx8_MVCo_Polling)         NfcrdlibEx8_NTagI2C (NfcrdlibEx8_MCE_T4T)         NfcrdlibEx8_NTagI2C (NfcrdlibEx9_NTagI2C)         ✓ NxpCtLib (NxpNfcRdLib)         ✓ PN7462AU (PN7462AU)         PN7462AU_ex_phExCctd (PN7462AU_ex_phExCctd)         PN7462AU_ex_phExCt1816 (PN7462AU_ex_phExCT7816)         PN7462AU_ex_phExCTEMVCo (PN7462AU_ex_phExCT816)         PN7462AU_ex_phExDoorAccess (PN7462AU_ex_phExDoorAccess)         PN7462AU_ex_phExDoorAccessEC (PN7462AU_ex_phExDoorAccessEC)	Select All
© Copy projects into workspace Working sets □ Add project to working sets Working sets: □ • • • • • • • • • • • • • • • • • • •	Sgleet Cancel

Fig 9. Select project

In the projects window all available projects in the package will be listed. To import only DoorAccess examples, it is mandatory to select next projects in the list:

- FreeRTOS\_Library
- NxpCtLib
- NxpNfcRdLib
- PN7462AU
- PN7462AU\_ex\_phExDoorAccess

All projects in the list can be also selected and imported to the workspace.

### 2.3 Building project

Building projects in a workspace is a simple case of using the "Quickstart Panel" - 'Build all projects'. Alternatively a single project can be selected in the "Project Explorer View" and built separately. Note that building a single project may also trigger a build of any associated library projects.

To build the project select appropriate project and press "Build" as shown in the figure below.

🔀 Start here 🕿 🚔
Import project(s)
📸 New project
Build all projects [Debug]
K Build 'PN7462AU_ex_phExDoorAccess
Clean PN740ZHU_EX_pnexDoorAccess
🗱 Debug 'PN7462AU_ex_phExDoorAccess
🀲 Terminate, Build and Debug 'PN7462AU_es_phExDoorAcce
8 Edit 'PN7462AU_ex_phExDoorAccess
🕲 Quick Settings 👻

As a part of the build output, the binary for Flash file is created. This binary file can be used to update PN7462AU Flash via USB mass storage interface or by using Flash tool or debug in LPCXpresso IDE.

Signaries Signa
BU Include:      Get NapOfclib      Get external      Get phCommon      Get phRtos      G
Get NapoYtLib     Get NapoYtLib     Get Restand     Get R
Image: Map Nation     Image: Map Nation       Image: State S
Image: Antibactor       Image
b 29 inc b 22 phCommon b 23 phRtos
e GegephCommon e GegephRtos
b Cableau
ice docs
ReadMe.txt
× m +

#### Note:

Due to legal issues, NXP cannot provide source code for the FreeRTOS and the

UM10957

**Door Access UM** 

FreeRTOS source code should be integrated by users. Detailed description on how to integrate source code of the FreeRTOS to the project it is described in the "AN11784 PN7462AU How to integrate RTOS" document.

After successful integration of the FreeRTOS source code, project build should be compiled without errors as shown on the picture below.

Connele 22 Vintions Diversion Content Content Content Content Content Content Content Content (Vintional) (Vintional) (Vintional) (Vintional)	*** (S) 3 (S) = (S) = (S) * (S
Finished building: D:/Projects/PH640/V-Gate/RC03.00.00.00/R03.00.00.00_INTERNAL/HupCtLib/phpalCt/src/phpalCt_TL.	c
Bullding target: MYA452AL ex_phExDoorAccess@M03.00.00.00_INTERNAL.oxf Invoking: M2U Linker ser-none-exbig:cc-noxidlib =1"D: Wrojects/W9440V-Gate/MC03.00.00.00[M03.00.00_DVTERNALIMY2462AU/ph+al/phhell Finished building target: MY2452AL ex_phstoporAccess@M03.00.00.00[INTERNAL.axf	Rfllib" -Xlinker -Map="NN74674U_ex_phExDoorAccess@R63.4
<pre>makemo-print-directory post-build Performing post-build steps an=none-eabl-inter "PMFAZUM_ex_philoDoorAccess@R03.00.00_DTERMALast"; arm-none-sabl-objcopy -v -0 binary "Pf text data bis dec hex filensme 10240 10 4376 12320 10ff0# PMFAZUM_ex_philoDoorAccess@R03.00.00_DTERMALast copy from "PMFAGAM_ex_philoDoorAccess@R03.00.00.00_DTERMALast" [eff32-littlearm] to "PMFAGAM_ex_philoDoorAccess copy from "PMFAGAM_ex_philoDoorAccess@R03.00.00.00_DTERMALast"</pre>	N746ZAU_ex_phixDoorAccess@003.00.00.00_INTERNAL.axf" "+ s@003.00.00.00_INTERNAL.bin' [Dimary]
14:17:20 Build Finished (took 275.977es)	
12. Successful build	

### 2.4 Activity flow

The activity flow of Non Exported version project is as shown below

User manual

**COMPANY PUBLIC** 

UM10957

**Door Access UM** 

### UM10957 Door Access UM



### 3. Door Access Use Case – Export Controlled Version

This section describes in detail the setup procedure and execution environment required for Export Controlled version of Door Access Use Case application.

This version of the application is using MIFARE DESFire EV1 card for the authentication, data exchange is done over contactless interface and software key store or SAM (Secure Access Module) key store is used for key storing.

#### 3.1 Hardware Setup

For preparing the project and building the source code components listed in the Table 5 are required.

Table 5. Development envi	ronment	
Item	Version	Purpose
PN7462AU Customer Board	2.1	Engineering development board.
LPC-Link2	1.0	Stand-alone debug adapter

UM10957

### UM10957 Door Access UM

Item	Version	Purpose
USBToSerial converter		USB To Serial
Power adapter		Optional
SAM card		MIFARE SAM AV2 in AV1 mode



### **Connection Details:**

1) DC Power Supply of 7.5 V

Board is powered using the 7.5 V power adapter.

2) Power supply selection jumper

Jumper should be placed to connect JP41.1 -> JP41.2

This is to choose the external power supply to the board.

3) SAM in main CT slot

MIFARE SAM AV2 in AV1 mode must be insert in CT main slot.

4) Connect Rx of 'USB - Serial converter' to ATX\_B

Take a USB to Serial converter (USB to TTL to UART RS232 COM Cable Module Converter) as shown in the figure.

One side of this cable has USB connector and other side has 4 wires.

- Vcc (Red wire)
- Tx (Green wire)
- Rx (White wire)
- Ground (Black wire)





Connect the RX line to the ATX\_B (jp32.3) pin of the Customer Board.

#### 5) Common Ground connection

Connect ground wire (Black wire) of the 'USB to Serial converter' to one of the common ground pins in the customer board as shown Fig 14.

#### 6) CTS to ground connection

Connect ATX\_D (JP32.7) pin of the Customer board to one of the common ground pins using a wire. This grounds the CTS line and this connection is necessary to print on the PC console using USB to serial converter.

#### 3.1.1 Hardware setup without external supply

To eliminate the use of external power supply, we can power the board using Vcc line (+5 V) of USB to serial converter. For this, Vcc line (Red wire) of USB to serial converter should be connected to JP41.2 of the board. (Remember to remove any external power supply selection jumper already placed on JP41).

UM10957



**UM10957** 



#### 3.1.2 LPC-Link2 Debug Adapter Connection

To flash the binary or to debug the code LPC-Link2 debug adapter should be used. It can be connected to JTag connector (JP4) of the customer board as shown in the figure below.



### 3.2 Importing project

After installation of the "PN7462AU FW & SW Examples Full Version" please follow steps described below.

The sequence of preparing the project is:

- Open the LPCXpresso IDE and select an empty workspace
- Select the option "Import project(s)" in Quick Panel

X Sta	rt here	* *		
9 Imp	ort project(s)			
C New	v project	-		
Di Buil	d all projects []	-		
🔏 Buili	d []			
🧹 Clea	in "]]		🗐 C 🔀 🛛	
🐝 Deb	ug '' []			
🛞 Edit	" project settings		No consoles	
Imp	ort project(s) from XM	L Descripti 👻		
			-	

The dialog for project import opens.

chive file to import.		T	
within archives (.zip) or are u directory and press <next>. ss <finish>.</finish></next>	inpacked within a directory. S On the next page, select thos	Select your se projects you	
Open and 'legacy' examples	s are provided.		
		6	
p89422\NXP Semiconductor	s\PN7462AUPspPackageFull	-v04_01_00\F Brow	wse
acked)		~	-
		Brow	NSe
nmended code base for Cort the LPCOpen packages whic	ex-M based NXP LPC Microc h can be imported directly b	controllers. y pressing the Brows	rse
archive (zip) section, above,	and navigating to the Examp	les/LPCOpen directo	tory.
e button below to Browse th ources on LPCWare.com	e LPCWare.com website for	latest resources.	
2 Park	Next > Emi	sh Cance	el
	chive file to import. within archives (.zip) or are to directory and press <next>. ess <finish>. COpen and 'legacy' example: p89422(NXP Semiconductor packed) mmended code base for Cont the LPCOpen packages while archive (zip) section, above, the button below to Browse the sources on LPCWare.com</finish></next>	chive file to import. within archives (zip) or are unpacked within a directory. directory and press <next>. On the next page, select tho: ess <finish>. COpen and 'legacy' examples are provided. p89422\NXP Semiconductors\PN7462AUPspPackageFull backed) mmended code base for Cortex-M based NXP LPC Microo the LPCOpen packages which can be imported directly b archive (zip) section, above, and navigating to the Examp e button below to Browse the LPCWare.com website for sources on LPCWare.com</finish></next>	chive file to import.

UM10957

Browse to the project zip file ".\PN7462AU Software\PN7462AU-FW\_v04.01.00-Full.zip" and click "Next".

Import project(s) Select a directory to search for existi	ing Eclipse projects.	
Projects:		
<ul> <li>FreeRTOS_Library (FreeRTOS_ NfcrdlibEx1_BasicDiscoveryL</li> <li>NfcrdlibEx1_BasicDiscoveryL</li> <li>NfcrdlibEx1_MIFAREDESFin</li> <li>NfcrdlibEx2_AdvancedDisco</li> <li>NfcrdlibEx3_NFCForum (Nfc</li> <li>NfcrdlibEx4_MIFAREClassic (</li> <li>NkpNfcRdLib (NxpNfcRdLib)</li> <li>NxpNfcRdLib (NxpNfcRdLib)</li> <li>PN7462AU_ex_phExcCid (PN74</li> <li>PN7462AU_ex_phExCTEMVC</li> <li>PN7462AU_ex_phExDoorAcce</li> <li>PN7462AU_ex_phExDoorAcce</li> <li>PN7462AU_ex_phExDoorAcce</li> </ul>	S_Library) .oop (NfcrdlibEx1_BasicDiscoveryLoop) e (NfcrdlibEx1_MIFAREDESFire) veryLoop (NfcrdlibEx2_AdvancedDiscoveryLoop) rrdlibEx3_NFCForum) (NfcrdlibEx4_MIFAREClassic) HibEx5_ISO15693) tock (NfcrdlibEx6_EMVCo_Loopback) (NfcrdlibEx7_EMVCo_Polling) HibEx8_HCE_T4T) HibEx8_HCE_T4T) HibEx9_NTagI2C) ) //462AU_ex_phExCcid) //62AU_ex_phExCT7816) to (PN7462AU_ex_phExCT7816) to (PN7462AU_ex_phExCT7816) tocs (PN7462AU_ex_phExDoorAccess) tess (PN7462AU_ex_phExDoorAccess)	Select All     Deselect All     Refresh
Copy projects into workspace     Working sets     Add project to working sets     Working sets.		Sgleet
0	< <u>B</u> ack <u>N</u> ext ≥ <u>Einish</u>	Cancel

In the projects window all available projects in the package will be listed. To import only Door Access Export Controlled example, it is mandatory to select next projects in the list:

- FreeRTOS\_Library
- NxpCtLib
- NxpNfcRdLib
- PN7462AU

**UM10957** 

**Door Access UM** 

- PN7462AU\_ex\_phExDoorAccessEC

All projects in the list can be also selected and imported to the workspace.

Select appropriate projects and click Finish. Selected applications are going to be imported to the workspace.

#### 3.3 Building project

Building projects in a workspace is a simple case of using the "Quickstart Panel" - 'Build all projects'. Alternatively a single project can be selected in the "Project Explorer View" and built separately. Note that building a single project may also trigger a build of any associated library projects.

To build the project select appropriate project and press "Build" as shown in the figure below.



As a part of the build output, the binary for Flash file is created. This binary file can be used to update PN7462AU Flash via USB mass storage interface or by using Flash tool or debug in LPCXpresso IDE.

Bullering Strategy Strate	PN7462AU_ex_phExDoorAccess@R03.00.00_INTERNAL	*
<ul> <li>▷ Includes</li> <li>▷ RepEdLib</li> <li>▷ Qe PN7462AU</li> <li>▷ Qe external</li> <li>○ Ze inc</li> <li>▷ Qe phRtos</li> <li>▷ Erc</li> <li>▷ Debug</li> <li>▷ Debug</li> <li>▷ docs</li> <li>○ ReadMe.txt</li> </ul>	▲ ※ Bioster ▶ 读 PN7462AU_ex_phExDoorAccess.axf - [arm/le]	-
General     G	b) Includes b) RxpCtLib	
General     G	▶ 🚑 NxpNfcLib	н
Common     Common	b      brain PN/462AU     b      CR     external	
e g pricommon 0	b 2 inc	
h ≧ src b ⊵ Debug i≥ dous ReadMe.bt =	phCommon	
Cebug     Cebug     Cecs     ReadMe.txt     T	þ 📴 src	
ReadMe.txt -	<ul> <li>b b Debug</li> <li>b docs</li> </ul>	
* H	ReadMe.txt	-
	×	F.

UM10957

#### Note:

Due to legal issues, NXP cannot provide source code for the FreeRTOS and the FreeRTOS source code should be integrated by users. Detailed description on how to integrate source code of the FreeRTOS to the project it is described in the "AN11784 PN7462AU How to integrate RTOS" document.

After successful integration of the FreeRTOS source code, project build should be compiled without errors as shown on the picture below.

CDT Build Console [PNJ462AU, ex_phExDoorAccess(\$P03.00.00.00_INTERNAL]	
Finished building: D:/Projects/PH640/V-Gate/RC03.00.00.00/R03.00.00_INTERNAL/HupCtLib/phpalCt/sr	c/phpalit_T1.c
Building target: PW7462AU_ex_phExDoorAccess@R03.00.00.00_INTERNAL.oxf	
Invoking: MCU Linker meroane-eabi-gcc -mostalib -L"D:\Projects\PM640\V-6ate\R083.00.00.00\R083.00.00.808.00_INTERNAL\PM7462A Finished building target: PM7462AaLex phtsDoorAccess@R08.00.00.00_INTERNAL.axf	W\piMal\phholRf\lib* 'Xlinker 'Map="PN74524U_ex_phExDcorAccess@R03.1
makemo-print-directory post-build	
Performing post-build steps arm-none-sabi-ize "PN7462AU_ex_phExDoorAccess@R03.00.00.00_INTERNAL.axf"; arm-none-sabi-objcopy -v text data bis dec hex filename	-U binary "PH746ZAU_ex_phixDoorAccess@R03.00.00.00_INTERNAL.axf" "
113240 16 4576 122832 Ldfd0 PM7462AU_ex_phtxDoorAccesa@R03.00.00_DMTERMAL.axf copy from `PM7462AU_ex_phtxDoorAccess@R03.00.00.00_IMTERMAL.axf' [elf32-littlearm] to `PM7462AU_ex_	phExDoorAccess@R03.00.00.00_INTERNAL.tEn' [binary]
18-17-20 Build Efwirked (Pank 17c 077er)	

#### 3.4 Activity flow

Activity flow of Export controlled version without SAM and with software keys is shown as below

User manual

**COMPANY PUBLIC** 

### UM10957 Door Access UM



### UM10957 Door Access UM



UM10957

### 3.5 Enabling the use of SAM in Export Controlled version

In export controlled version, by default use of SAM is disabled and software keys are enabled. To enable SAM support, a macro "PHFL\_ENABLE\_SAM\_KEY\_STORE" should be added in the projects settings as shown below.

type filter text	Settings		\$*9
<ul> <li>Builders</li> <li>C/C++ Build</li> <li>Build Variables</li> <li>Environment</li> <li>Logging</li> </ul>	Configuration Release (Active)	Build Artifisct 📾 Binary Parsers 🧿 Error Parsers	Manage Configurations.
MCU settings Settings Tool Chain Editor C/C++ General Project References Rurv/Debug Settings	B MCU C Compiler     Dialect     Beprocessor     Symbols     Sincludes     Optimization     BOcbugging     Warnings     Miscellaneous     Architecture     MCU Assembler     B General     Architecture & Headers     MCU Linker	Defined symbols (-D) CORE NO _REDIB_ NDEBUG=1 PHFL_APP_NXBUILD_CONFIG PHFL_ENABLE_SAM_KEY_STORE	· (1) (1) (1) (1) (1) (1) (1) (1) (1) (1)
0	General     General     Gibraries     Maccellaneous     Shared Library Settings     Architecture     Managed Unker Script     Multicore	Undefined symbols (-U)	€ € ≥ (r d) OK Cancel

UM10957

User manual

### 4. Application Messages displayed in the PC console

Door Access application can send application messages through the HSU interface to the PC. Please check chapter 2.1 to see how to connect.

Type of the Card, UID and Data read will be displayed on the PC Console. This can be viewed by opening the corresponding COM port using any Serial COM tools like SSCOM.

Below figure shows the display of the prints for different cards on SSCOM tool in the case of Door Access Export Control. Here the COM port for USB to Serial connector is COM6.

Type B Card - UID: - Type A Card - Type I ICode III:11! Type A Card - ISO14 ICode Sii!!! - UID: A- Felica Card - UID: 0	1443-4A - UID: 04247ACADF258 4FF90640 2 Tag 1443-4A - UID: 040D53A961288 4B3F604500304E0 127006054F2ACA4	80 Data: 0123456789ABCDEF	
OpenFile FileNm		SendFile SaveData HexDat	+ a
ComNum COM6 💌	CloseCom Help	WWW. MCU51.COM EXT	1
10000		Serial port debug tool: SSCOM ! Auth: NieXiaoMeng.	
BaudRa 9600 DataBi 8 StopBi 1 Verify None FlowCo None	Send evi     Hos     Hisriki       SendHEX     Sendblew       Data input:     SEND	Get new version: http://www.mcu51.com/download/sscom.ra Your suggestion	ır

### 5. Enabling standby mode

By default standby mode is not enabled. It can be enabled by defining a symbol (MACRO) in the project properties:

PHFL\_ENABLE\_STANDBY

Standby time is set to 500 ms and LPCD also enabled.

Below figure shows how to enable Standby mode.

#### Door Access UM

**UM10957** 



UM10957

User manual

**COMPANY PUBLIC** 

### 6. References

[1] LPCXpresso webpage <u>http://www.nxp.com/products/software-and-tools/software-development-</u> <u>tools/software-tools/lpc-microcontroller-utilities/lpcxpresso-ide-v8.2.2:LPCXPRESSO</u>

[2] Windows drivers for USBToSerial Convertor www.nxp.com/external/prolific

UM10957

User manual

**COMPANY PUBLIC** 

### 7. Legal information

#### 7.1 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.

### 7.2 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.

### 7.3 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.

#### Purchase of NXP ICs with ISO 14443 type B functionality



This NXP Semiconductors IC is ISO/IEC 14443 Type B software enabled and is licensed under Innovatron's Contactless Card patents license for ISO/IEC 14443 B.

The license includes the right to use the IC in systems and/or end-user equipment.

RATP/Innovatron Technology

### 7.4 Trademarks

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

MIFARE - is a trademark of NXP B.V.

I<sup>2</sup>C bus logo — is a trademark of NXP B.V.

ICODE and I-CODE - are trademarks of NXP B.V.

User manual COMPANY PUBLIC



## 8. List of figures

Fig 1.	Block Diagram – Non Export Controlled Versi	on 3
Fig 2.	Block Diagram - Export Controlled Version	4
Fig 3.	PN7462AU Customer Board – Hardware Setu (with power adapter)	ир 7
Fig 4.	USB to serial converter	8
Fig 5.	USB powered setup (setup without any extern power supply)	nal 9
Fig 6.	LPC Link2 connection for debugging and flashing	9
Fig 7.	Importing project to LPCXpresso IDE	.10
Fig 8.	Importing project to LPCXpresso IDE	.10
Fig 9.	Select project	.11
Fig 10.	Build project	.12
Fig 11.	Build output - flash binary file	.12
Fig 12.	Successful build	.13
Fig 13.	Activity flow of Non Exported version	.14
Fig 14.	Door Access Export controlled version hardwas setup (with power adapter)	are .15
Fig 15.	USB to serial converter	.16
Fig 16.	Hardware setup without external supply	.17
Fig 17.	LPC Link2 connection for debugging and flashing	.17
Fig 18.	Importing project to LPCXpresso IDE	.18
Fig 19.	Importing project to LPCXpresso IDE	.18
Fig 20.	Select project	.19
Fig 21.	Build project	.20
Fig 22.	Build output - flash binary file	.20
Fig 23.	Authentication with SW Keys	.22
Fig 24.	Authentication with SAM DES Keys	.23
Fig 25.	Enabling the use of SAM	.24
Fig 26.	Application printouts using SSCOM tool	.25
Fig 27.	Project Properties – Enable Standby Mode	.26

User manual

COMPANY PUBLIC

### 9. List of tables

Table 1.	LEDs status specification	4
Table 2.	LEDs status specification	5
Table 3.	Development environment	5
Table 4.	Development environment	7
Table 5.	Development environment	.14

User manual

COMPANY PUBLIC

### 10. Contents

1.	Introduction3
1.1	Door Access Application – Non Export Controlled
1.2	Door Access Application – Export Controlled
	version4
1.3	Development environment5
1.4	Project download link6
1.5	Driver Installation for USB to serial converter6
2.	Door Access Use Case – Non Export Controlled
	Version7
2.1	Hardware Setup7
2.1.1	Hardware setup without any external supply8
2.1.2	LPC-Link2 Debug Adapter Connection9
2.2	Importing project9
2.3	Building project
2.4	Activity flow13
3.	Door Access Use Case – Export Controlled
	Version14
3.1	Hardware Setup14
3.1.1	Hardware setup without external supply16
3.1.2	LPC-Link2 Debug Adapter Connection
3.2	Importing project
3.3	Activity flow
3.4 3.5	Activity now
5.5	version 24
4	Application Messages displayed in the PC
	console 25
5	Enabling standby mode 25
J. с	Deferences
o. _	References
7.	Legal information
7.1 7.0	Definitions
1.2	Discialmers
1.3 7 1	LICENSES
1.4	11autiliains20

Please be aware that important notices concerning this document and the product(s) described herein, have been included in the section 'Legal information'.

#### © NXP B.V. 2016.

#### All rights reserved.

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

> Date of release: 13 September 2016 357911 Document identifier: UM10957