

UM10948

EEPROM Management of PN746X and PN736X

Rev. 1.0 — 2 February 2017
353310

User manual
COMPANY PUBLIC

Document information

Info	Content
Keywords	PN7462, PN7362, PN7360; EEPROM
Abstract	This document describes how to manage EEPROM of PN7462 family.



Revision history

Rev	Date	Description
1.0	20170202	First revision

Contact information

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

1. Introduction

The EEPROM of the PN7462 family can be generated and loaded in several ways, through the USB Mass Storage mode, by NFC Cockpit tool or using In-Application Programming (IAP) feature of the PN7462 IC.

This document describes how generate and load EEPROM in the right way.

2. PN7462 User EEPROM

2.1 EEPROM memory mapping

The PN7462 embeds 4 kB of on-chip byte-erasable and byte-programmable EEPROM data memory. The user EEPROM memory is 3.5 K Bytes and is situated in the physical address range of 0x201200 to 0x201FFF.

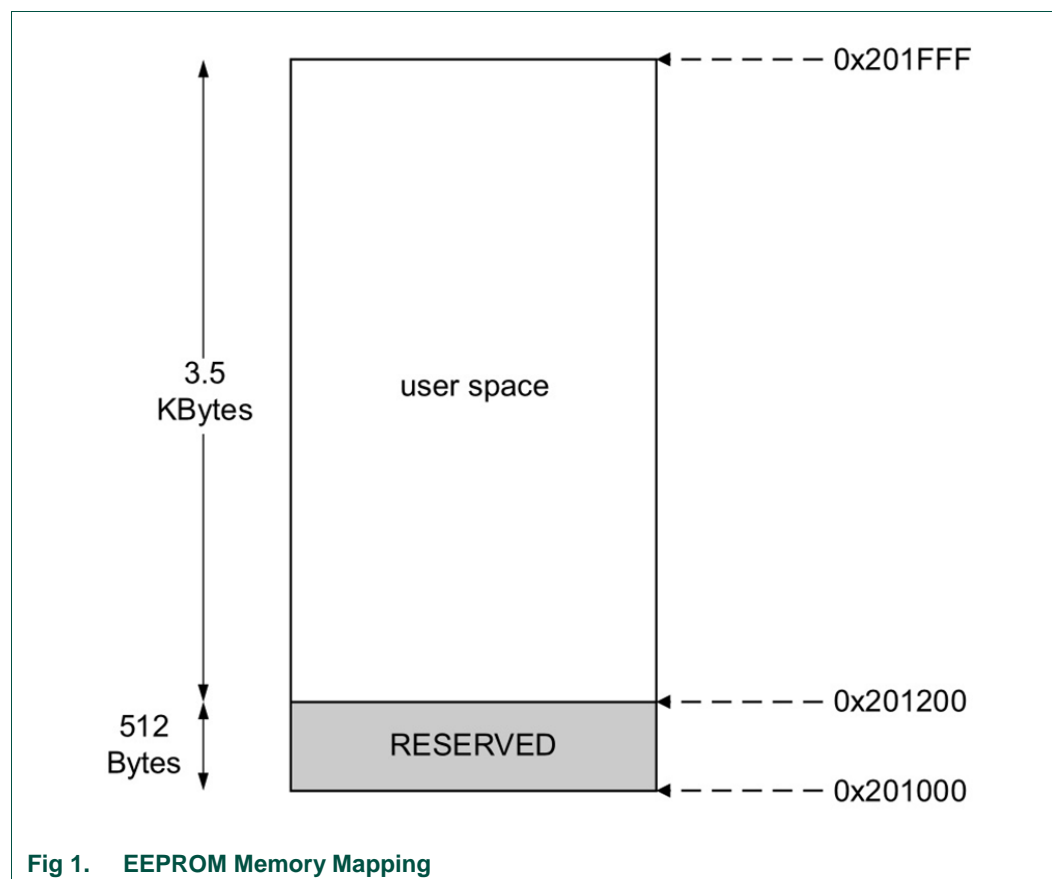
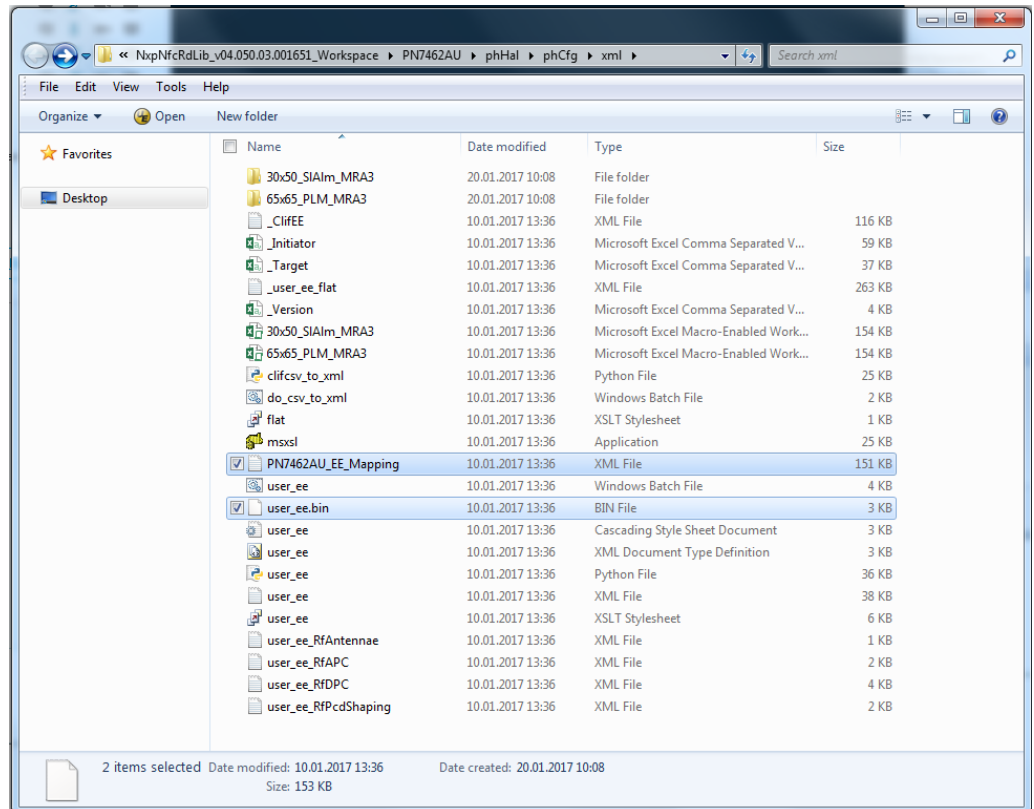


Fig 1. EEPROM Memory Mapping

2.2 PN7462 EEPROM Availability

PN7462 EEPROM binary and configuration tools are provided with the PN7462 FW release package [3].

Content and the path to the folder containing EEPROM.



- (1) Extract "PN7462AU-FW_v04.050.03.011702_Full.zip" file
- (2) Path to the EEPROM:" PN7462AU\PN7462AU\phHal\phCfg\xml"

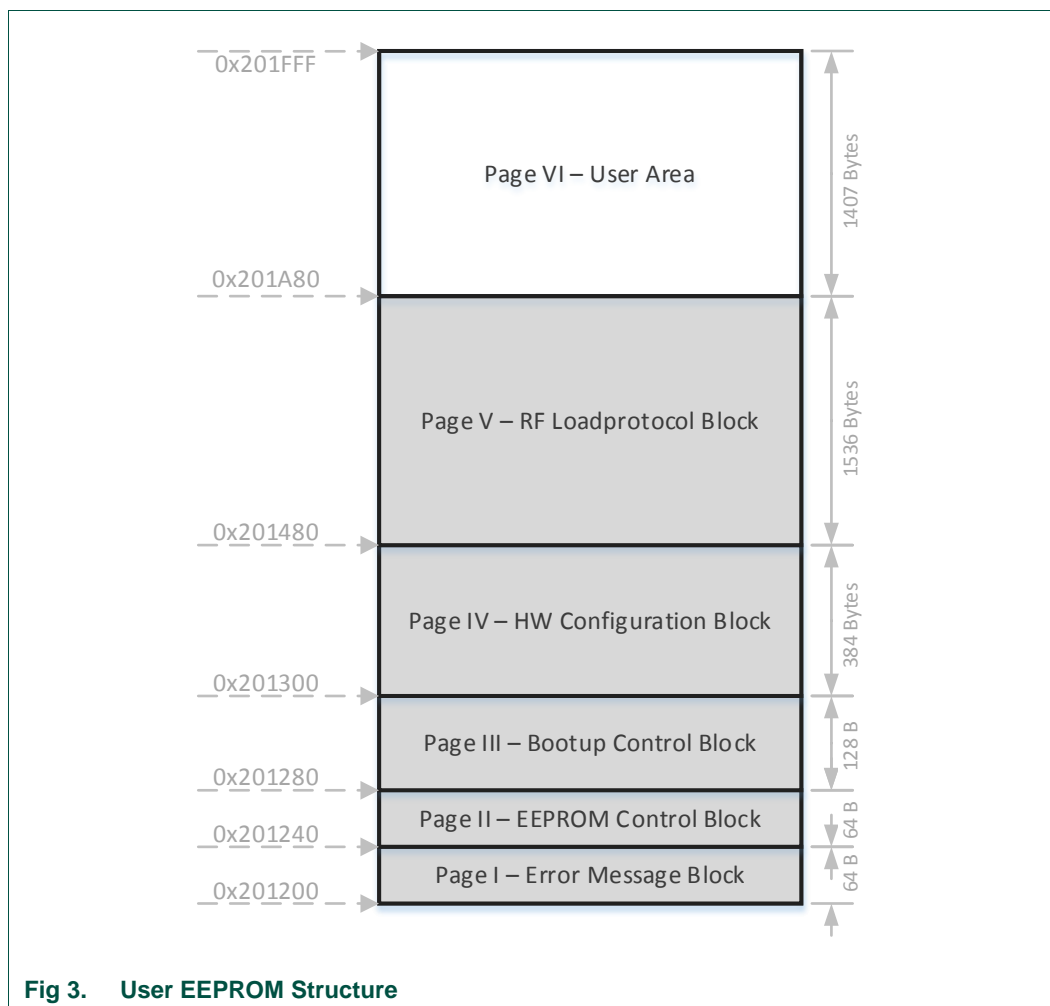
Fig 2. EEPROM Availability

The file containing user EEPROM binary is "user_ee.bin".

"PN7462AU_EE_Mapping.xml" file contains all information about the EEPROM content in the xml format.

2.3 Structure of the User EEPROM

User EEPROM is composed by 6 Pages. Pages 1 to 5 are reserved for PN7462 IC configuration and can be updated by user.



Page I – Error Message Block

- Contains error message in case of failure during ROM boot up.

Page II – EEPROM Control Block

- Contains EEPROM version information.

Page III – Boot up Control Block

- These values are used during the Boot up and applied to HW IPs.

Page IV – HW Configuration Block

- Contains default HW configuration

- These settings are not directly applied at boot up, but eventually during the course of usage of the IPs.

Page V – RF Load Protocol Block

- Values for RF Load Protocol based on the RF tuning.

Page VI

- This region of the EEPROM memory does not contain any data and may be freely used.
- This part of the memory is reserved for user data.

2.4 Building a new EEPROM binary

Provided PN7462 package contains tools to generate a new user EEPROM binary. To build a new binary it is required to install Python v2.7 on the user PC.

The new binary is created out of the xml configuration files:

- user_ee.xml
- user_ee_RfAntennae.xml
- user_ee_RfAPC.xml
- user_ee_RfDPC.xml
- user_ee_RfPcdShaping.xml

user_ee.xml

Main configuration file, defines structure and values of the EEPROM, start address and configuration Pages.

user_ee_RfAntennae.xml

Defines the type of the RF antenna (ALM/PLM).

user_ee_RfAPC.xml

Defines APC settings.

user_ee_RfDPC.xml

Defines DPC settings.

user_ee_RfPcdShaping.xml

Contains configuration of the RF PCD Shaping.

All xml configuration files can be updated manually.

To create a new EEPROM binary, “user_ee.bat” batch must be executed.

As a result, we get:

- “user_ee.bin”
- “phCfg_EE.c” and “phCfg_EE.h”
- “phCfg_PF.c”
- “PN7462AU_EE_Mapping.xml”
- “_output_xml_sizes.html”

user_ee.bin

Binary file with the user EEPROM.

phCfg_EE.c and phCfg_PF.c

Those two files are automatically generated and added to the project structure. All changes in the EEPROM are also reflected in those files.

PN7462AU_EE_Mapping.xml

This xml file contains the complete user EEPROM and It must be copied to NFC Cockpit structure in case of a new version of the EEPROM. Details are described in the chapter 4 of this document.

_output_xml_sizes.html

This html file contains EEPROM structure and parameter description with defined values.

Figure below show part of the “_output_xml_sizes.html” file describing RF DPC feature.

Address	Type	Field Name	Default Value	Description
[see phhalRf_DPCCConfig_t]				
0x201328	u16	wControlCycle	0x4E20 (hex)	Sets the value for the periodic regulation. Time base is 1/20Mhz. (Example: Value of 20000 is equal to 1ms)
0x20132a	u16	wAgcFastModeConfig	0x2540 (hex)	Controls AGC FastMode (StepSizeEnabled: 13 + StepSize: 12..11 + DurationEnabled: 10 + Duration: 9..0)
0x20132c	u16	wAgcTrshLow	0x144 (hex)	Low threshold for gearshift
0x20132e	u16	wGuardTimeFastMode	0x88B8 (hex)	Guard time after AGC fast mode has been triggered. This happens in the following scenarios: - End of Receive - End of Transmit - After a gear switch Time base is 1/20Mhz (Example: Value of 2000 is equal to 100us)
0x201330	u16	wGuardTimeSoFDetected	0x61A8 (hex)	Guard time after SoF or SC detection. This is to avoid any DPC regulation between SoF/SC and actual begin of reception. Time base is 1/20Mhz (Example: Value of 2000 is equal to 100us)
0x201332	u16	wGuardTimeFieldOn	0x0190 (hex)	Guard time after Gear Switch during FieldOn instruction. Time base is 1/20Mhz (Example: Value of 2000 is equal to 100us)
[Struct:4] RfDPC 0x201328	u16 [15]	wAgcTrshHigh	0x014A (hex)	High thresholds for each gear
			0x014B (hex)	
			0x014C (hex)	
			0x014D (hex)	
			0x014E (hex)	
			0x014F (hex)	
			0x0150 (hex)	
			0x0151 (hex)	
			0x0152 (hex)	
			0x0153 (hex)	
			0x0154 (hex)	
			0x0155 (hex)	
			0x0156 (hex)	
			0x0157 (hex)	
			0x0158 (hex)	
			0x0159 (hex)	
0x201352	u8	bOcProtControl	0x73 (hex)	Control byte (StartGear: 7..4 bits + GearStep: 3..1 bits + OcProtLoopEnabled: 0 bit)
0x201353	u8	bAgcXi	0x0 (hex)	Compensation value for the AGC
0x201354	u8	bDebug	0x0 (hex)	Enable/Disable debug signals
0x201355	u8	bAgcShiftValue	0x05 (hex)	Shift value for AGC dynamic low threshold adjustment
0x201356	u8	bSizeOfLUT	0x09 (hex)	Number of fields in the following configuration look up table
0x201357	u8 [15]	bConfigLUT	0xF9 0xF1	Look up table for configuration values
			0xF3 0xF5	
			0xF7 0xF0	
			0xF2 0xF4	
			0xF6 0x96	
			0x66 0x46	
			0x36 0x26	
			0x16 (hex)	

Fig 4. Snippet of the _output_xml:sizes.htm File

2.5 How to Add Custom Parameter

At the end of the EEPROM it is reserved space for the user parameters.

Figures below shows how to add and use custom parameters.

Follow guidelines below to add a new parameter to the user EEPROM:

Step 1:

Open “user_ee.xml” and add new parameters at the end of the file as shown in the figure below. It is mandatory to follow xml structure:

```
<page name="PageName">
  <description>PageName description</description>
  <structure name="StructureName">
    <description>Description of the structure</description>
    <field name="FieldName" type="fieldType" value="FieldValue" valuetype="ValueType">
      <description>Description of the field</description>
    </field>
  </structure>
</page>
```



```
<page name="UserBlock">
  <description>
    This part of the EEPROM memory is reserved for the user parameters.
  </description>
  <structure name="ReleaseMode">
    <description>
      Used to distinguish between development and production version.
    </description>
    <field name="bIsProductionVersion" type="u8" value="0x00" valuetype="hex" >
      <description>
        Defines if the version of the application is development or production.
      </description>
      <option value="0" description="Development Version" />
      <option value="1" description="Production Version" />
    </field>
    <field name="bPrimaryDownloadDisable" type="u8" value="0x00" valuetype="hex" >
      <description>Disable Primary Download feature</description>
      <option value="0" description="Enabled" />
      <option value="1" description="Disabled" />
    </field>
    <field name="bSWDAccessDisable" type="u8" value="0x00" valuetype="hex" >
      <description>Disable SWD access</description>
      <option value="0" description="Enabled" />
      <option value="1" description="Disabled" />
    </field>
  </structure>
</page>
```

Fig 5. Adding new Parameters to the “user_ee.xml”

Step 2:

Next step is to execute “user_ee.bat” batch file which results with new EEPROM binary, new html file and new project files ("phCfg_EE.c", "phCfg_EE.h" and "phCfg_PF.c").

Fig 6. Execute Batch File

Project files are automatically copied to the project structure, path “PN7462AU\phHal\phCfg\src”.

New parameters, with the description, are added to the “_output_xml_sizes.html” file.

Page VII [UserBlock]

This part of the EEPROM memory is reserved for the user parameters.

Starting Address : 0x201ac0

Address	Type	Field Name	Default Value	
Used to distinguish between development and production version.				
[Struct:1] ReleaseMode	0x201ac0	u8	bIsProductionVersion	0x00 (hex)
				Defines if the version of the application is development or production. • 0 : Development Version • 1 : Production Version
	0x201ac1	u8	bPrimaryDownloadDisable	0x00 (hex)
				Disable Primary Download feature • 0 : Enabled • 1 : Disabled
	0x201ac2	u8	bSWDAccessDisable	0x00 (hex)
				Disable SWD access • 0 : Enabled • 1 : Disabled

Fig 7. Snippet from the new “_output_xml_sizes.html” File

Step 3:

Flash the new EEPROM binary to the PN7462. This step is detailed described in chapter “Update EEPROM by USB Mass Storage Mode”.

The EEPROM is now updated and new parameters can be used in the application. Code snippet below shows how to use them. New variables were added to the “*phCfg_EE.h*” file and the values from the EEPROM are accessible through:

```
gpkphCfg_EE_UserBlock_ReleaseMode->bIsProductionVersion
gpkphCfg_EE_UserBlock_ReleaseMode->bPrimaryDownloadDisable
gpkphCfg_EE_UserBlock_ReleaseMode->bSWDAccessDisable
```

```
phStatus_t phExMain_DisableFlashMemoryAccess()
{
    phStatus_t wStatus = PH_ERR_READ_WRITE_ERROR;
    phhalSysSer_Secrow_Config_t sSecrowConfig;
    phhalSysSer_USB_Primary_Dnld_Config_t sPriDwnlCfg;

    // Check the application version (development/production)
    if(gpkphCfg_EE_UserBlock_ReleaseMode->bIsProductionVersion)
    {
        if(gpkphCfg_EE_UserBlock_ReleaseMode->bPrimaryDownloadDisable)
        {
            /* Disable primary download. */
            sPriDwnlCfg.Primary_Dnld_Disable[0] = 1;
            wStatus = phhalSysSer_USB_PrimaryDnldConfig(PH_SYSRV_SET_DATA, &sPriDwnlCfg);
        }
        if(gpkphCfg_EE_UserBlock_ReleaseMode->bSWDAccessDisable)
        {
            /* Block SWD. */
            sSecrowConfig.bBlockSWD[0] = 1;
            wStatus = phhalSysSer_OTP_SecrowConfig(PH_SYSRV_SET_DATA, &sSecrowConfig);
        }
    }

    return wStatus;
}
```

Fig 8. The use of new parameters

3. Update EEPROM by USB Mass Storage Mode

PN7462 supports EEPROM update by USB Mass Storage Mode. This chapter describes steps how to update EEPROM by USB Mass storage mode.

3.1 Mount PN7462 IC as a USB mass storage device

Mount PN7462 IC as USB Mass storage drive:

- Ensure that “HIF selection” is USB, see figure below
- Connect PC with PN7462 by USB
- Press “RST_N” switch, (Reset pin high)
- Press “DWL_REQ” switch, (Download pin high)
- Release “RST_N” and keep holding “DWL_REQ”. (Reset the board)
- Release “DWL_REQ” after some seconds.

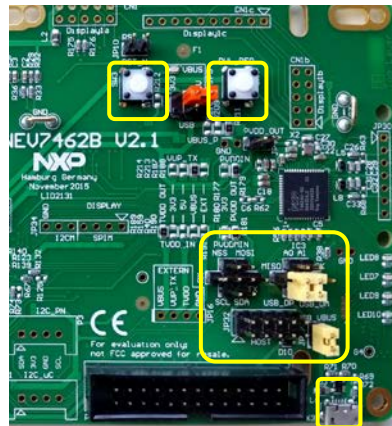


Fig 9. PN7462AU as USB mass storage device

Now PN7462 should be detected by PC as a new Mass Storage device.

Note:

PN7462 supports only USB Mass Storage mode on Windows operating system.

Figure below shows the list of files available by PN7462 as mass storage device.

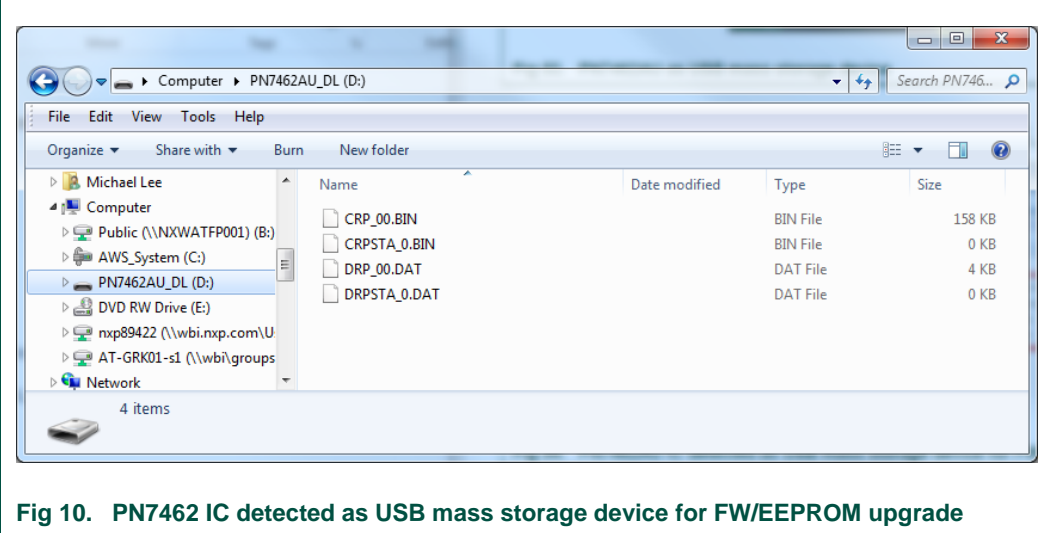


Fig 10. PN7462 IC detected as USB mass storage device for FW/EEPROM upgrade

When the PN7462AU is mounted as a USB mass storage device, files listed in the table below are available in the device root folder.

Table 1. Files available in USB mass storage

File	Description
CRP_<nn>.BIN	PN7462AU’s user flash code (see Table 4 for description of <nn>)
CRPSTA_<s>.BIN	Status of previous write operation to user flash (see Table 5 for description of <s>)
DRP_<nn>.DAT	PN7462AU’s user EEPROM date (see Table 4 for description of <nn>)
DRPSTA_<s>.DAT	Status of previous write operation to user EEPROM (see Table 5 for description of <s>)

Level of data or code protection is presented as a number extension <nn> of the binary file name. The protection level of data and code in the flash memory can be changed by the “System Services”, see SW UM [6], chapter 3.3 PN7462AU ROM services, for more details.

Table below lists all protection levels for Flash code and EEPROM data.

Table 2. Code and data protection level

<nn>	Description
00	Read and write allowed
01	Cannot read. Write allowed. Only applicable sectors erased before writing.
02	Cannot read. All sectors of the applicable memory are erased before writing.
03	Cannot read. Cannot write via USB mass storage.

Status of the last write operation is presented as an extension <s> of the file names “CRPSTA_<s>.bin” and “DRPSTA_<s>.DAT”.

Table 3. Status of read write operating code

<s>	Description
0	Last write operation was successful.
1	Memory region formatted.
2 – or anything else	Failed.
3	Fresh memory (FLASH/ EEPROM has never been downloaded via USB mass storage).

3.2 Update User EEPROM by USB Mass Storage Mode

To update the user EEPROM by mass storage mode, IC must to be in primary download mode. To enter to the primary download mode it is required to follow steps described in the previous chapter.

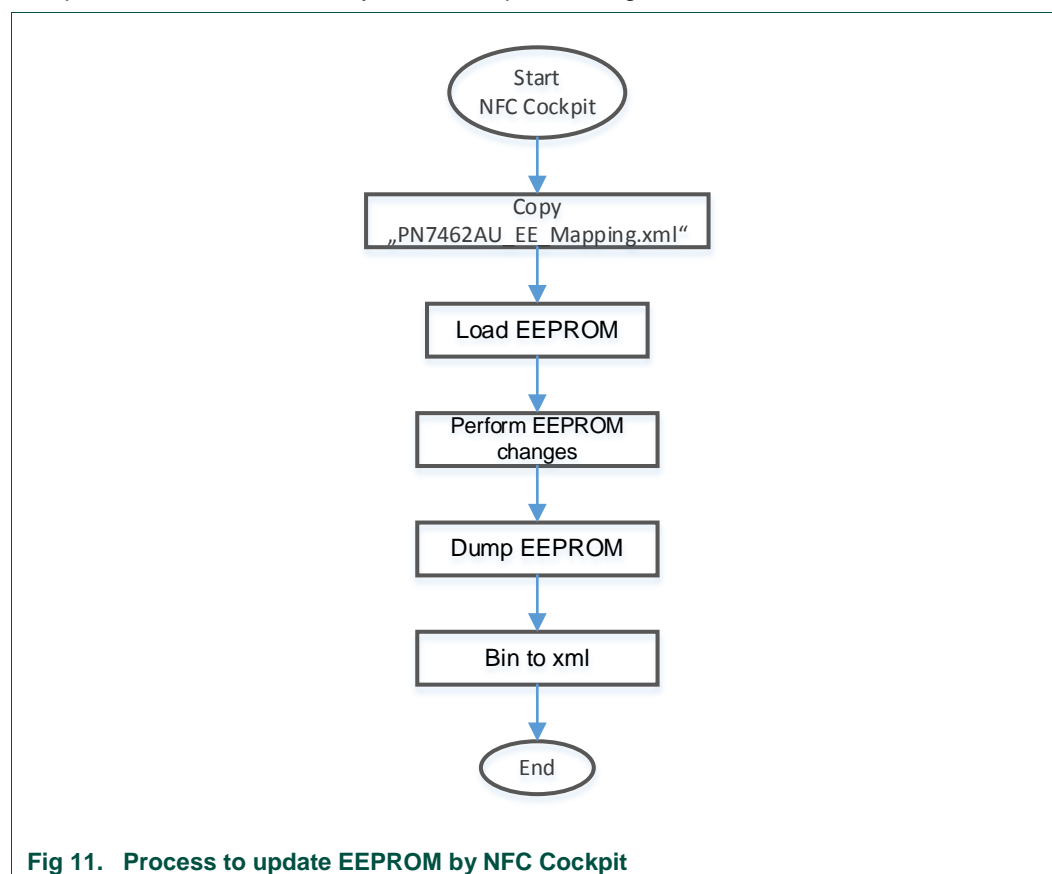
Next step is to exchange the EEPROM on the IC:

- Browse to the newly mounted Drive.
- Delete DRP_<nn>.bin file
- Copy the user EEPROM binary file (“user_ee.bin”) to the new drive.
- PN7462AU should automatically un-mount and mount again.
- If the status file DRPSTA_00.bin is present, the download was successful, otherwise check Table 3 about the error description.
- Reset the board. (Press the “RST_N”)
- PN7462 is rebooted with the new EEPROM.

4. Update EEPROM by NFC Cockpit Tool

The NFC Cockpit is an intuitive graphical user interface (GUI), which streamlines development by separating the tasks associated with hardware and software. It supports software-independent tuning of the antenna's register settings, so hardware designers can optimize antenna parameters while software designers work on other aspects of the system.

To update EEPROM values by NFC Cockpit, follow guidelines below:



Note:

the latest version of the "PN7462AU_EE_Mapping.xml" xml file and "user_ee.bin" are available in the PN7462 FW package and both files are also available in the NFC Cockpit configuration folder - "NxpNfcCockpit_v3.7.0.0\cfg\PN7462AU".

4.1 Select EEPROM Configuration (xml) Folder

PN7462 EEPROM is combined of RF and FW configurations, details are described in the chapter 2.3. NFC Cockpit can update only RF related configuration values, therefore it is mandatory to assign the right xml file with the NFC Cockpit tool.

To do that it is mandatory to copy “PN7462AU_EE_Mapping.xml” xml file to the NFC Cockpit structure. The latest version of the EEPROM mapping file can be found in the PN7462 FW package – “PN7462AU\phHal\phCfg\xml\PN7462AU_EE_Mapping.xml”.

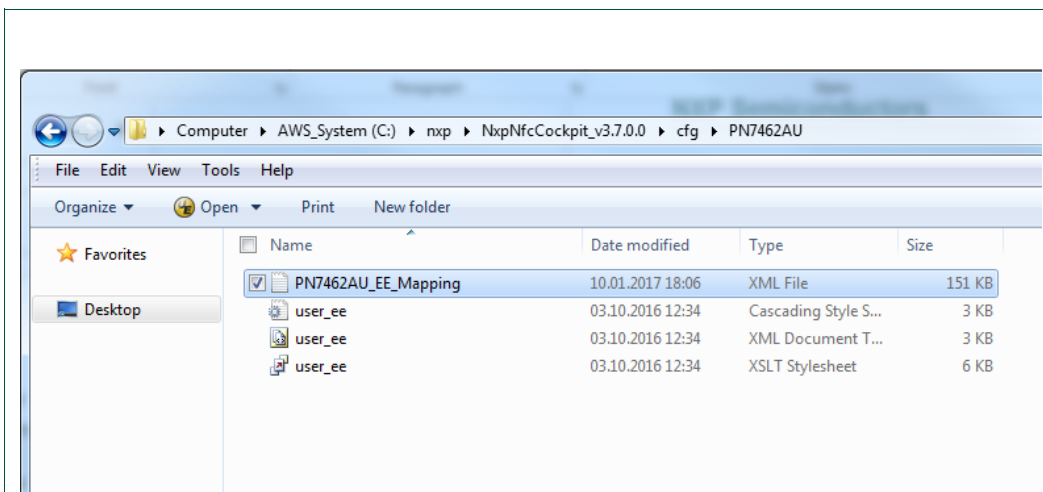


Fig 12. Path to the PN7462 EEPROM configuration folder

4.2 Flash EEPROM and Create EEPROM XML files

NFC Cockpit tool supports flashing EEPROM to the IC with the provided binary. The latest version of the EEPROM binary is available in the PN7462 FW: “PN7462AU\phHal\phCfg\xml\user_ee.bin”.

Follow steps below to flash PN7462 user EEPROM:

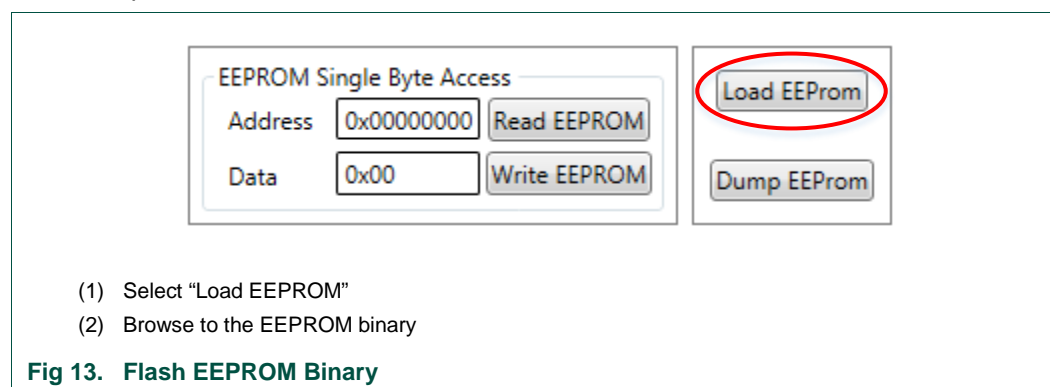


Fig 13. Flash EEPROM Binary

4.3 EEPROM management with NFC Cockpit Application

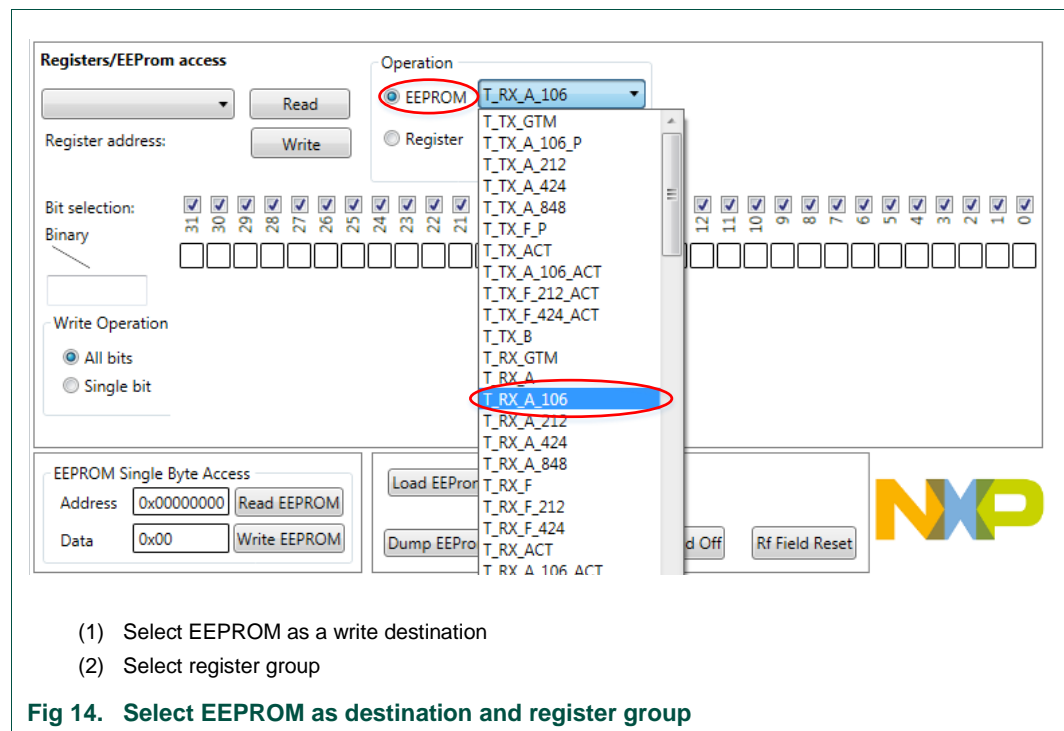
PN7462 EEPROM values can be changed by NFC Cockpit tool in two ways:

- Through the register selection
- Through the single byte access

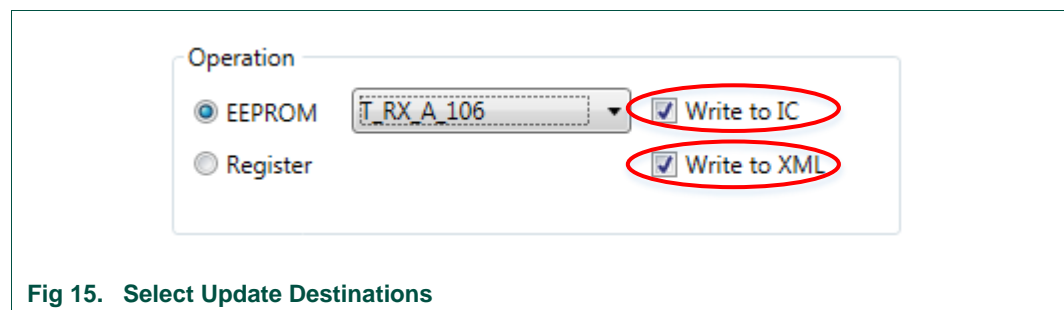
Update EEPROM by register selection

To change the EEPROM values through the registers, follow guidelines below:

- Choose the “EEPROM” option as a destination in the “Operation” panel
- Select appropriate register group from the list



Next step is to define where to write updates, IC EEPROM or xml files.



Select the register from the register list of the selected register group. Next step is update register value and save it to the EEPROM.

Registers/EEProm access

CLIF_ANA_RX_REG

Read

Write

Register address: 40004110

Operation

☒ EEPROM

☐ Register

T_RX_A_106

Bit selection:

Binary

000390A3

Write Operation

☒ All bits

☐ Single bit

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 1 0 0 0 0 1 0 1 0 0 0 0 1 1

INTERNAL_USE...

RX_HPCF

RX_GAIN

(1) Select the register

(2) Use the "Read" button to get the register value from the EEPROM

(3) Update the register value

(4) Use the "Write" button to save the register value to the EEPROM

Write Operation option allows us to update only one bit or the complete register.

Bit selection:

Binary

Hex. Value:

00004080 0h

Write Operation

☐ All bits

☒ Single bit

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

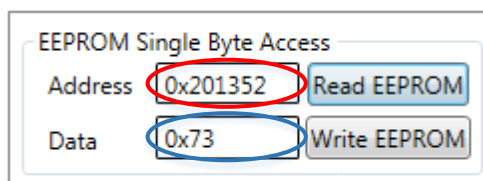
0 0

INTERNAL_USE...

Fig 17. Select Write Option

Update EEPROM by single byte access

NFC Cockpit allows single byte EEPROM access. One Byte long data can be read or written to the EEPROM.



EEPROM Single Byte Access

Address

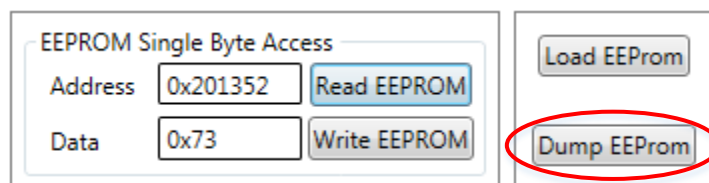
Data

- (1) Insert the address of the EEPROM value as hex data
- (2) Use "Read" or "Write" button to perform operation

Fig 18. Single Byte EEPROM Access

4.4 Dump EEPROM

PN7462 user EEPROM can be dumped as a binary file or as a xml file.



EEPROM Single Byte Access

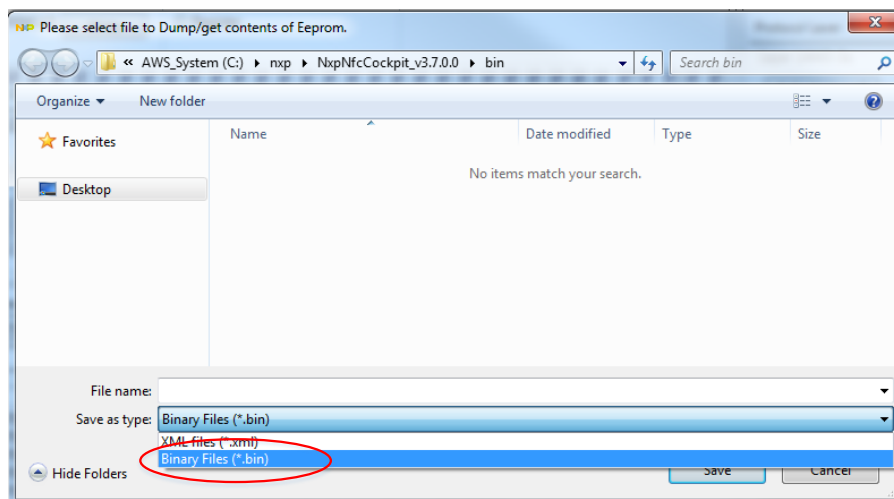
Address

Data

- (1) Select "Dump EEPROM"
- (2) Browse to the destination folder

Fig 19. Dump PN7462 EEPROM

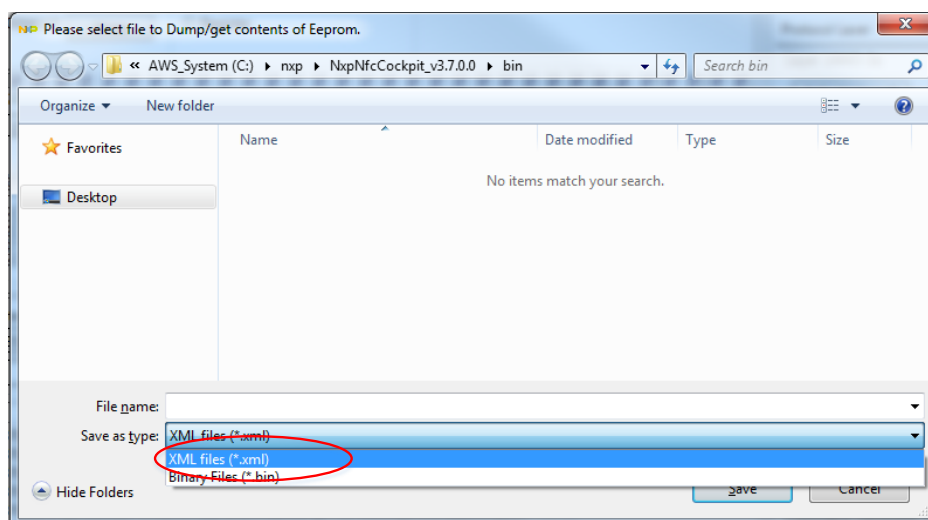
To dump the EEPROM binary select “*.bin” as a file type in the save window.



- (1) Select “Binary Files” as a file type
- (2) Add a file name and click on “Save” button

Fig 20. Select “*.bin” to save EEPROM binary

To dump the EEPROM as a xml file select “*.xml” as a file extension in the “Save As” window.



- (3) Select “XML Files” as a file type
- (4) Add a file name and click on “Save” button

Fig 21. Select “*.xml” to save EEPROM xml

5. Update EEPROM using In-Application Programming (IAP)

PN7462 EEPROM can be also updated directly from the PN7462 FW using In-Application Programming (IAP).

Note:

In case of using IAP to update PN7462 EEPROM, EEPROM xml files are not updated. This can cause problems in the case that we want to modify EEPROM later with NFC Cockpit or by creating a new EEPROM manually with EEPROM xml files.

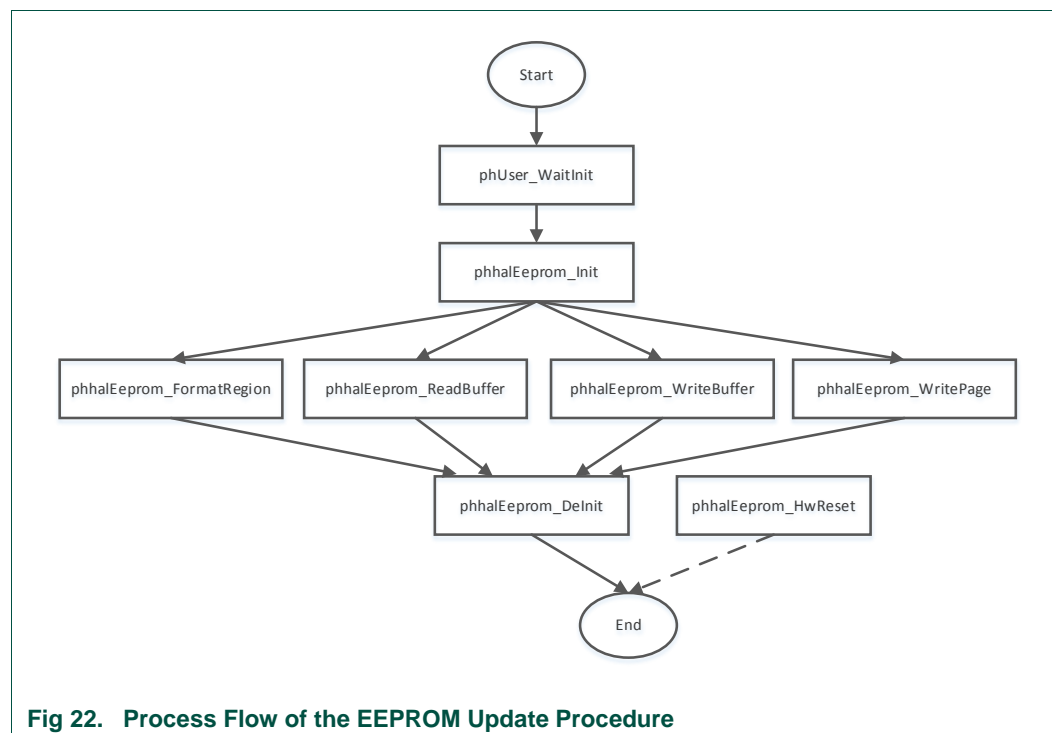
Table 4. IAP functions for EEPROM management

Function	Description
phhalEeprom_WriteBuffer()	This function writes data to EEPROM.
phhalEeprom_WritePage()	This function writes a single full 64byte EEPROM page, with data provided by the user.
phhalEeprom_FormatRegion()	This function fills a specified region in the EEPROM with a format pattern.
phhalEeprom_ReadBuffer()	This function reads EEPROM data.

Detailed description of all functions is provided in API Documentation:

Path: "PN7462AU\docs\14_user_doc".

Figure below presents process flow how to manage EEPROM by IAP.



6. Update EEPROM over SWD

PN7462 EEPROM can be programmed over the Serial Wire Debug Interface (SWD). Guidelines below describes how it can be done using LPCXpresso IDE tool.

Note:

Before the start it is necessary to install LPCXpresso IDE tool and connect debug probe (LPC-LINK2) as described in the “PN7462 Quick Start Guide” user manual [7].

Here are the steps to program EEPROM over SWD:

- 1) Ensure that the latest LPCXpresso plugin, for the PN7462, is installed. LPCXpresso plugin is part of the PN7462 FW installation package.
- 2) Open LPCXpresso IDE tool with any PN7462 project.
- 3) Open flash programming utility within the LPCXpresso IDE environment by clicking on the "Program Flash" icon on the toolbar at the top of the IDE window

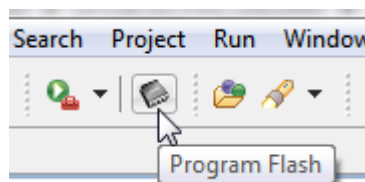


Fig 23. Open Program Flash utility

- 4) In the dialog box select correct flash driver - PN7xxxxx_EE_3_5k.cfx. it can be found in the LPCXpresso IDE installation folder - "C:\nxp\LPCXpresso_8.2.2_650\lpcxpresso\bin\Flash".
- 5) For the binary in the dialog box browse to the EEPROM binary: "PN7462AU\phHal\phCfg\xml\user_ee.bin".
- 6) As a "Base address" in the dialogue box use: 0x201200.
- 7) Click "OK" to start programming.

Figure below shows how to set right values in the "Program Flash" dialog box.

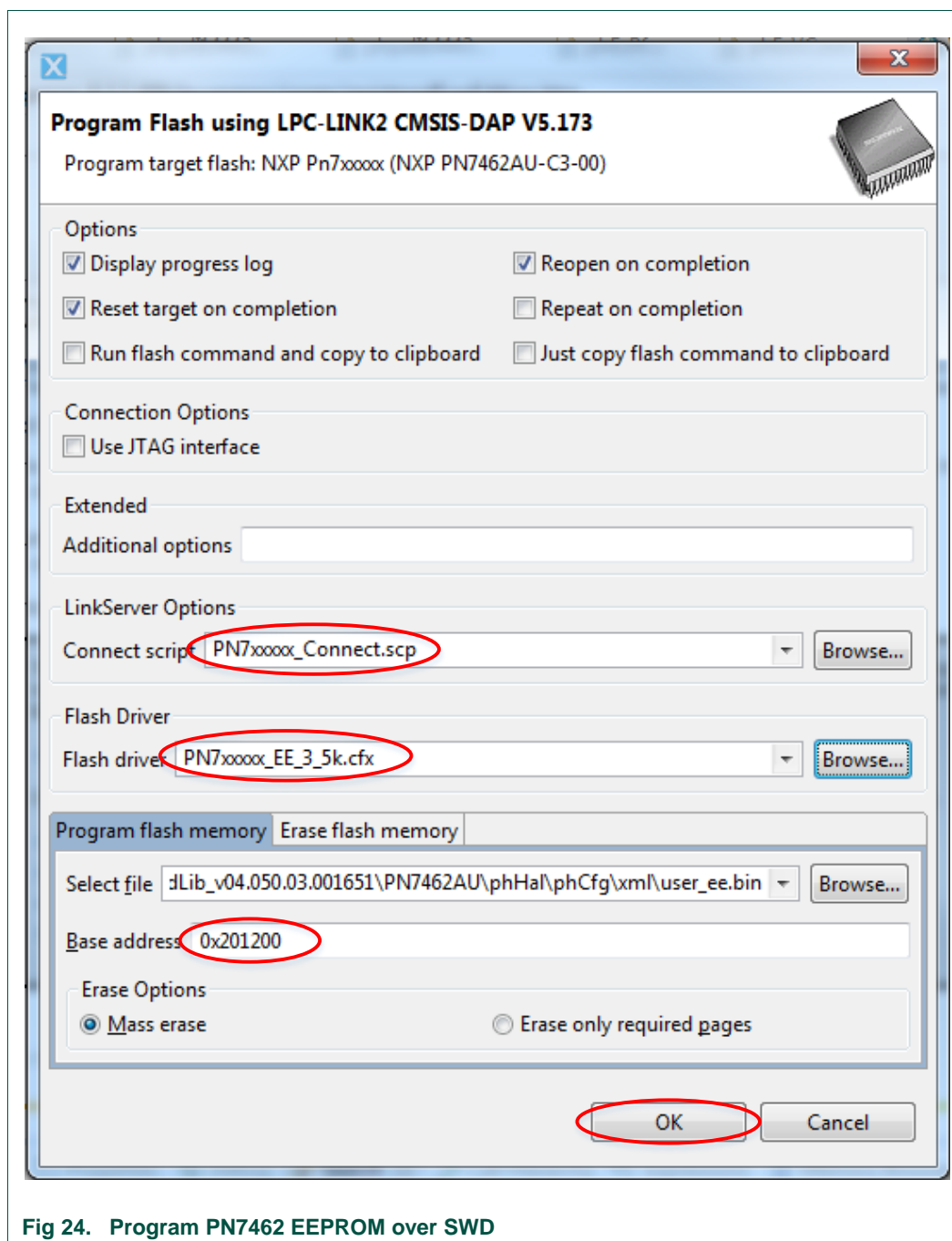


Fig 24. Program PN7462 EEPROM over SWD

7. References

- [1] PN736X Data sheet
http://www.nxp.com/documents/data_sheet/PN736X.pdf
- [2] PN7462 Data sheet
http://www.nxp.com/documents/data_sheet/PN7462.pdf
- [3] PN746X_736X SERIES Product page
http://www.nxp.com/pages/nfc-cortex-m0-microcontroller:PN746X_736X_SERIES
- [4] PN7462 Release Package, SW package on product page, BU S&C document number SW3683.
<http://www.nxp.com/products/identification-and-security/nfc-and-reader-ics/nfc-controller-solutions/nfc-cortex-m0-microcontroller-with-160-kb-flash-and-contact-interface:PN7462AUHN>
- [5] OM27462CDK: NFC Controller development kit
<http://www.nxp.com/products/identification-and-security/nfc-and-reader-ics/nfc-controller-solutions/nfc-controller-development-kit:OM27462CDK>
- [6] PN7462 NFC Cockpit Tool
https://www.nxp.com/webapp/Download?colCode=SW3707&Parent_nodeId=1459363882695729020240&Parent_pageType=product&Parent_nodeId=1459363882695729020240&Parent_pageType=product
- [7] UM10913 PN7462AU Software User Manual
http://www.nxp.com/documents/user_manual/UM10913.pdf
- [8] UM10883_PN7462AU Quick Start Guide - Customer Board
http://www.nxp.com/documents/user_manual/UM10883.pdf

8. Legal information

8.1 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.

8.2 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the *Terms and conditions of commercial sale* of NXP Semiconductors.

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

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

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

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

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

Evaluation products — This product is provided on an "as is" and "with all faults" basis for evaluation purposes only. NXP Semiconductors, its affiliates and their suppliers expressly disclaim all warranties, whether express, implied or statutory, including but not limited to the implied warranties of non-infringement, merchantability and fitness for a particular purpose. The entire risk as to the quality, or arising out of the use or performance, of this product remains with customer.

In no event shall NXP Semiconductors, its affiliates or their suppliers be liable to customer for any special, indirect, consequential, punitive or incidental damages (including without limitation damages for loss of business, business interruption, loss of use, loss of data or information, and the like) arising out of 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.

8.3 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

8.4 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.

9. Contents

1.	Introduction	3
2.	PN7462 User EEPROM.....	3
2.1	EEPROM memory mapping	3
2.2	PN7462 EEPROM Availability.....	4
2.3	Structure of the User EEPROM.....	5
2.4	Building a new EEPROM binary.....	6
2.5	How to Add Custom Parameter.....	8
3.	Update EEPROM by USB Mass Storage Mode	11
3.1	Mount PN7462 IC as a USB mass storage device.....	11
3.2	Update User EEPROM by USB Mass Storage Mode	13
4.	Update EEPROM by NFC Cockpit Tool.....	14
4.1	Select EEPROM Configuration (xml) Folder	15
4.2	Flash EEPROM and Create EEPROM XML files	15
4.3	EEPROM management with NFC Cockpit Application.....	16
4.4	Dump EEPROM.....	18
5.	Update EEPROM using In-Application Programming (IAP).....	20
6.	Update EEPROM over SWD.....	21
7.	References	23
8.	Legal information	24
8.1	Definitions	24
8.2	Disclaimers.....	24
8.3	Licenses	24
8.4	Trademarks	24
9.	Contents.....	25

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