ANxxxxx

Android NFC Setup Guide for SN100x for SN100x

Rev. 1.00 — 21 February 20188 **Application note**

Document information

Info	Content
Keywords	NFC, Android

Revision history

Rev	Date	Description
1.0	2015-11-06	Initial version for NXP NFC Setup Guide

Contact information

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

1. Introduction

NXP's NCI based NFC controllers (SN100U/T) are designed to work with Android open source using NCI based stack for Android NFC

2. Scope

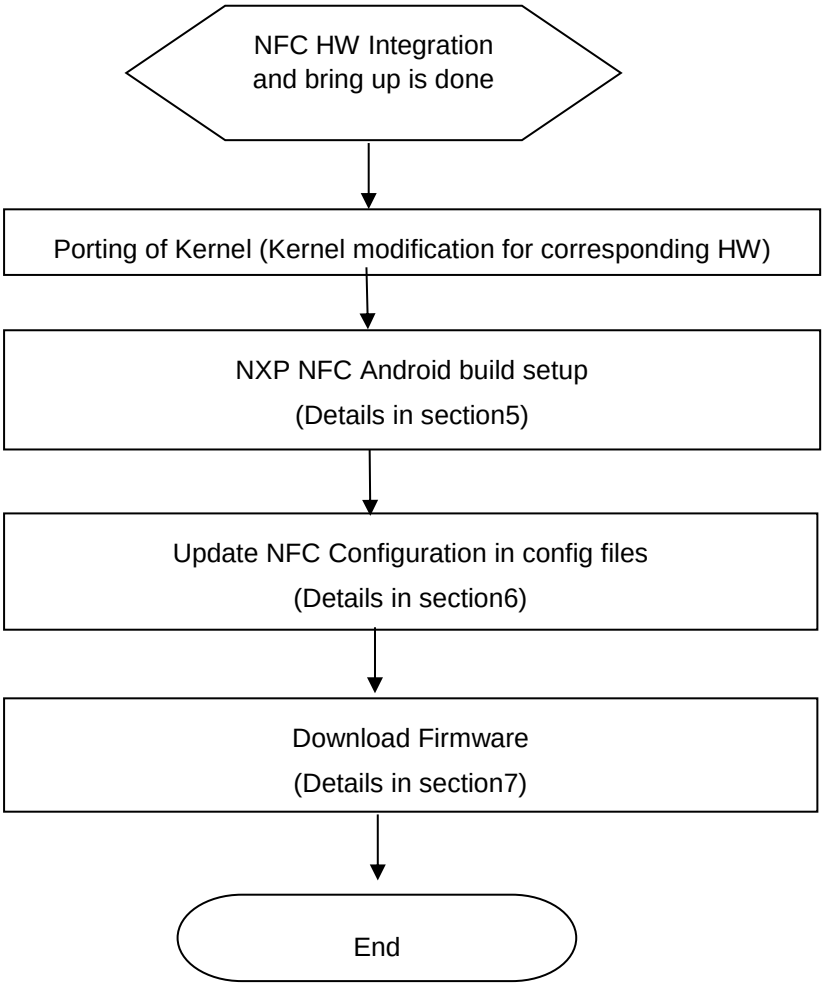
This setup guide provides guidelines for setting up NXP NFC in android build environment. It is an example guideline for basic system integration. OEM integration may have variations based on actual system and integration.

3. General steps for Android NFC integration

For the NFC software integration with Android, it is hereby assumed that NFC IC HW integration is done in a platform with following checks.

- Schematic reviewed with NXP
- HW IC interface like I2C/SPI, SWP (if used) working.
- Antenna designed and reviewed
- Antenna connection working
- GPIO connections checked

Picture below shows basic flow for Android NFC SW bring up. Following sections describe these steps in detail.



4. Architecture Overview

Figure 2 describes the architecture of Android NFC with NXP-NFCC. NCI HALx provides the Hardware abstraction for the NXP's NFCC. NXP additional features/enhancements are part of NXP Extensions provided on top of AOSP NFC Stack.

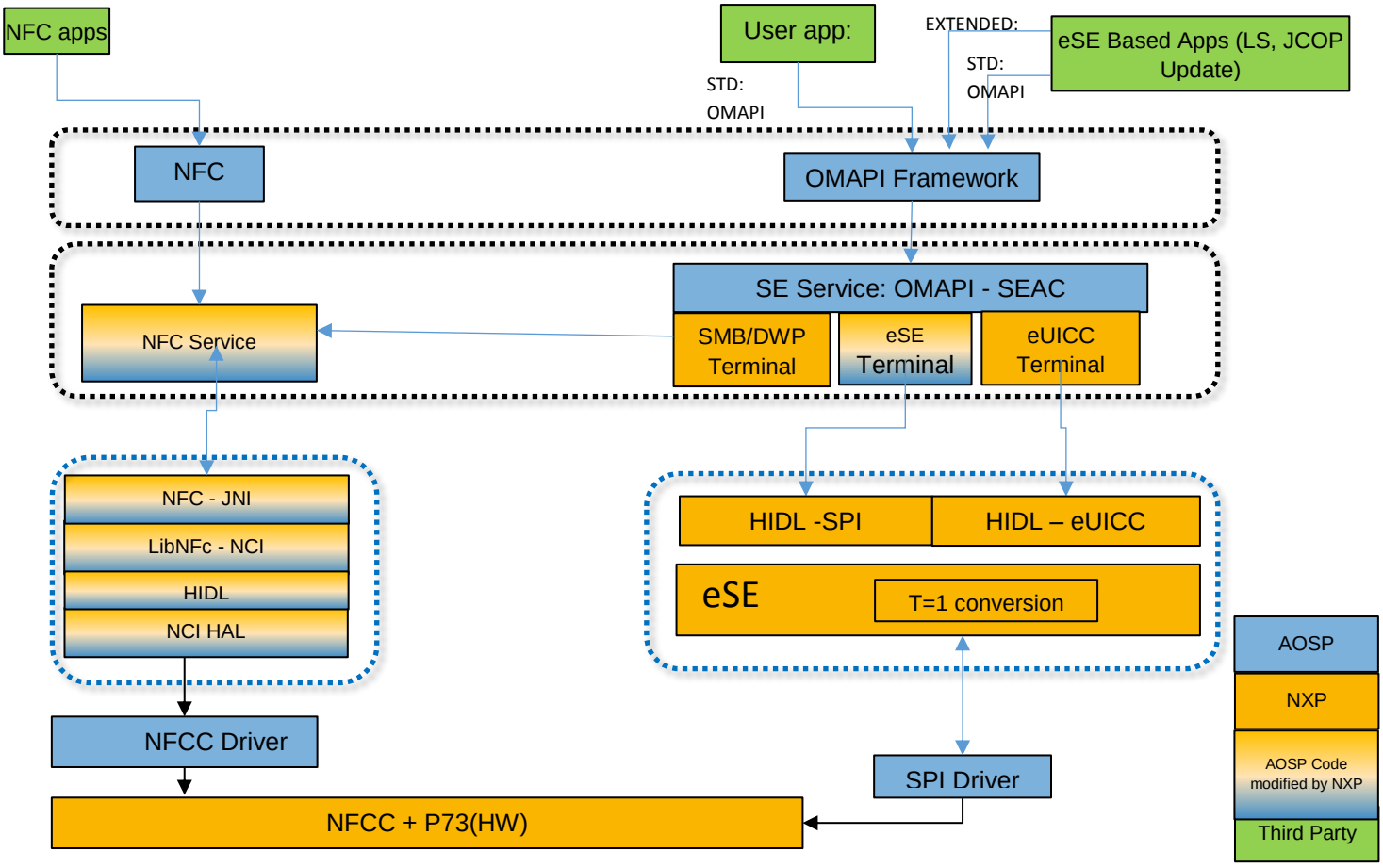


Figure 2: Android NFC with NXP-NFCC

5. Setup of Android NFC

5.1 Downloading Android Source Code

Following the instructions from Android website:

http://source.android.com/source/downloading.html

Use following command to get source code for respective branch Android-x.y:

```
repo init -u https://android.googlesource.com/platform/manifest -b android-x.y
repo sync -f
```

5.2 Building the Android Source Code

Use android build instructions from Android website for building android OS image:

http://source.android.com/source/building.html

Information about the public APIs supported by Android NFC are available on following links:

http://developer.android.com/reference/android/nfc/package-summary.html

http://developer.android.com/reference/android/nfc/tech/package-summary.html

5.3 Android NFC package description

Information of NXP's NFC Project repositories in the GitHub are as below:

Repository Name	Checkout Command
NFC_NCIHAL_Nfc	*git clone https://github.com/NXPNFCProject/NFC_NCIHAL_Nfc.git* The contents of this folder needs to be placed in packages/apps/Nfc directory in the android build.
NFC_NCIHAL_libnfc-nci	*git clone https://github.com/NXPNFCProject/NFC_NCIHAL_libnfc-nci.git* The contents of this folder needs to be placed in system/nfc directory in the android build.
NFCNCIHAL_base	*git clone https://github.com/NXPNFCProject/NFCNCIHAL_base.git* The contents of this folder needs to be merged in frameworks/base directory in the android build.
nfcandroid_hidlintf_nfc	*git clone https://github.com/NXPNFCProject/nfcandroid_hidlintf_nfc.git* The content of this folder needs to be merged in hardware/interfaces/nfc in the android build.
nfcandroid_hidlintf_se	git clone https://github.com/NXPNFCProject/nfcandroid_hidlintf_se.git The content of this folder needs to be merged in hardware/interfaces/secure_element in the android build.
nfcandroid_nfc_hidlimpl	*git clone https://github.com/NXPNFCProject/nfcandroid_nfc_hidlimpl.git* The content of this folder needs to be merged in hardware/nxp/nfc in the android build.
nfcandroid_se_hidlimpl	git clone https://github.com/NXPNFCProject/nfcandroid_se_hidlimpl.git The content of this folder needs to be merged in

	hardware/nxp/secure_element in the android build.
nfcandroid_secureelement	*git clone https://github.com/NXPNFCProject/nfcandroid_secureelement.git* The content of this folder needs to be merged in Packages/apps/SecureElement in the android build.
NXPNFC_Reference	The content of this folder needs to be merged in vendor/nxp in the android build.
NXPNFCC_FW	*git clone https://github.com/NXP/nfc-NXPNFCC_FW.git*

Description of the contents of the directories of NXP's NFC Project in the GitHub are as below:

Module Type	Path	Description
NFC Interfaces and public APIs	NFC_NCIHAL_base/core/java/android/nfc NFC_NCIHAL_base/core/java/android/se	Contains Android NFC Framework files.Contains OMAPI framework interfaces
	NXPNFC_Reference/vendor/nxp/opensource/frameworks	NXP's extensions to android NFC Framework implemented only for Android master(P)
NFC JNI and Java implementation of NCI stack	NFC_NCIHAL_Nfc/nci	Contains files for Nfc Nci stack.
	NFC_NCIHAL_Nfc/nci/jni/extns/pn54x	Contains implementation of extension features developed by NXP in JNI layer. E.g. Mifare classic support.
	NFC_NCIHAL_Nfc [Remaining parts]	Contains android NFC application source files.
NCI based NFC stack implementation	NFC_NCIHAL_libnfc-nci	Contains NCI based Native NFC stack
HAL implementation	nfcandroid_hidlintf_nfc	Contains hardware interfaces for NFCC
	nfcandroid_hidlintf_se	Contains hardware interface for eSE
	nfcandroid_nfc_hidlimpl	Contains hardware abstraction layer for NXP specific controllers
	nfcandroid_se_hidlimpl	Contains T=1 protocol stack and hardware abstraction layer for eSE
Firmware	NXPNFCC_FW	It is a directory, which includes the firmware file for NXP-NFCC

5.4 Android NFC Apps and Lib on Target

Projects	Compiled Files	Location in target device
NFCNCIHAL_base/core/java/android/nfc	Will be part of framework.jar	/system/framework
	NfcNci.apk	/system/app/NfcNci
NFC_NCIHAL_libnfc-nci	libnfc_nci.so	/system/lib64
nfcandroid_nfc_hidlimpl	nfc_nci_nxp.so	/system/vendor/lib64
nfcandroid_nfc_hidlimpl/extns	vendor.nxp.nxpnfc@1.0.so	/system/lib64
nfcandroid_se_hidlimpl	ese_spi_nxp.so	/system/vendor/lib64
nfcandroid_se_hidlimpl/extns	vendor.nxp.nxpese@1.0.so	/system/lib64
nfcandroid_hidlintf_nfc	android.hardware.nfc@1.0.so android.hardware.nfc@1.1.so	/system/lib64
nfcandroid_hidlintf_se	android.hardware.secure_element@1.0.so	/system/lib64

Following table lists the binaries used for Android NFC using NCI based stack.

5.5 Building the Kernel Source Code

Follow the following steps for building the kernel:

- Create nxp/pn8xT folder inside kernel/driver/
- Copy pn54x-i2c from NXPNFC_I2CDriver and keep inside as per chip type.
- Copy p73-spi from NXPNFC_SPIDriver put inside the nxp/pn8xT with respect to the chip type.
- Compile the kernel using corresponding cross compiler and copy the generated <platform>. dtb and Zimage file to the ANDROID_ROOT/device/vendor/platform-kernel.

Generate the corresponding boot and dt (device tree) images for the target platform

6. Compilation of NCI HALx and Extensions

6.1 Compilation Flags

Below are some of the compilation flags defined in Android.bp files.

Compilation Flags	Description
NFC_NXP_ESE	Set to false in Android make files as referred in the NFC_NXP_CHIP_TYPE compilation flag if Embedded secure element is not used.

6.2 Configuration files

There are configuration files used by Android NFC which is located in `system/etc` (`vendor/etc` for Android master) directory of target. These files are provided in GitHub project `nfcandroid_nfc_hidlimpl/halimpl/pn54x` as example files. This section describes the different flags in configuration files as examples. There are many additional flags in the configurations per NFC chip (please refer the config files part of GitHub releases)

6.2.1 Configurations in `libnfc-brcm.conf` files

Configurations	Descriptions
`HOST_LISTEN_TECH_MASK`	Forcing HOST to listen for a selected technology • 0x00: Disable Host Listen • 0x01: Enable Host Listen for Tech A • 0x02: Enable Host Listen for Tech B • 0x03: Enable Host Listen for Tech AB
`NXP_FWD_FUNCTIONALITY_ENABLE`	In case a communication is initiated in a RF technology (A or B) supported by host or eSE,but not supported by UICC, the forward function allows to forward the ISO/IEC 14443 level 4 commands to UICC. • To Disable: Set to 0x00 • To Enable: Set to 0x01 • Default: 0x01

6.2.2 Configurations in `libnfc-nxp.conf` file

Configurations	Descriptions
`DEFAULT_AID_ROUTE`	Configuration to set default route location for AID. This settings will be used when application does not set this parameter using the DefaultRouteSet() API defined in android framework. • Host: 0x00 • eSE(embedded Secure Element): 0x01 • UICC: 0x02 • Default: 0x00 However, if the NFCC routing table entries overflow with set default AID route, then the default routing location may be modified internally to accommodate all the AID's.
`DEFAULT_DESFIRE_ROUTE`	Configuration to set default route location for ISO-DEP Protocol. This settings will be used when application does not set this parameter using the MifareDesfireRouteSet() API defined in android framework.

	• Host: 0x00 • eSE: 0x01 • UICC: 0x02 • Default: 0x02
DEFAULT_MIFARE_CLT_ROUTE	Configuration to set default route location for A, B & F Technology. This settings will be used when application does not set this parameter using the MifareCLTRouteSet() API defined in android framework. • Host: 0x00 • eSE: 0x01 • UICC: 0x02 • Default: 0x02
NXP_FW_NAME	Name of the firmware file (ex: libsn100u_fw.so). This name shall be name of the file as in /system/vendor/firmware directory

Note1: There are example libnfc-nxp.conf files provided with release package. Please contact NXP support engineer to create the libnfc-nxp.conf for customer platform based on requirement and antenna design, which can be used for end product in production.

Note2: Optionally, NXP can provide the tool and training for creating libnfc-nxp.conf based on customer platform requirements. Please contact NXP support engineer for more details.

7. Firmware Download

NXP provides precompiled firmware for ARM platforms. NXP also can provide firmware as .c file and it can be compiled as .so file with the platform compiler. Firmware resides at location /system/vendor/firmware/ on the android target system. The firmware filename can be set in NXP_FW_NAME configuration in libnfc-nxp.conf file

Firmware can be updated when NXP releases a new version. Steps to update are as follows:

1. Compile the firmware to .so file using the file received in .C file format. If firmware is in .so format then this step can be skipped.
2. Set the FW name in libnfc-nxp.conf file in NXP_FW_NAME
3. Push the firmware file to /system/vendor/firmware directory on target.
4. Reboot the device or disable and enable NFC service. New firmware will be downloaded during the NFC service boot up

Note 1: Firmware download can take up to 20 seconds. Boot can take more time when FW is being downloaded.

Note 2: It is recommended not to modify the original firmware download logic of Android NFC.

Note 3: It is recommended that Firmware is always upgraded and not downgraded. If firmware version is required to be downgraded, then please consult NXP.

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 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 <xxx> components
<License statement text>

8.4 Patents

Notice is herewith given that the subject device uses one or more of the following patents and that each of these patents may have corresponding patents in other jurisdictions.

<Patent ID> — owned by <Company name>

8.5 Trademarks

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

<Name> — is a trademark of NXP Semiconductors N.V.

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

Date of release: 21 February 20187

Document identifier: AN11762-NXP NFC Setup Guide