Getting started with the X-CUBE-NFC7 dynamic NFC/RFID tag IC software expansion for STM32Cube

Introduction

The X-CUBE-NFC7 software expansion for STM32Cube provides a complete middleware for STM32 to build applications using the ST25DV64KC dynamic NFC/RFID tag IC.

The software is based on STM32Cube technology and expands STM32Cube-based packages. It is built on top of STM32Cube software technology to ease portability across different STM32 microcontrollers.

The software comes with sample implementations of the drivers running on the X-NUCLEO-NFC07A1 expansion board plugged on top of a NUCLEO-F401RE, NUCLEO-L053R8, or NUCLEO-L476RG development board.

RELATED LINKS

Visit the STM32Cube ecosystem web page on www.st.com for further information

1 Acronyms and abbreviations

Table 1. List of acronyms

Acronym	Description
NFC	Near field communication
RFAL	RF abstract layer
P2P	Peer-to-peer
MCU	Microcontroller unit
BSP	Board support package
HAL	Hardware abstraction layer
LED	Light emitting diode
SPI	Serial peripheral interface
CMSIS	Arm® Cortex® microcontroller software interface standard

2 X-CUBE-NFC7 software expansion for STM32Cube

2.1 Overview

The X-CUBE-NFC7 software package expands the STM32Cube functionality.

The package key features are:

- Complete middleware to build applications using the ST25DV64KC dynamic NFC/RFID tag IC
- Easy portability across different MCU families, thanks to STM32Cube
- Sample applications to:
  - Enable energy harvesting
  - Activate GPO interrupt
  - Activate low power down
  - Set I²C protection
  - Use ST25DV64KC mailbox
  - Write URI NDEF
- Free, user-friendly license terms
- Sample implementation available on the X-NUCLEO-NFC07A1 expansion board, plugged into a NUCLEO-F401RE, NUCLEO-L053R8, or NUCLEO-L476RG development board
- Package compatible with STM32CubeMX: it can be downloaded from and installed directly into STM32CubeMX

The package extends STM32Cube by providing a board support package (BSP) for the X-NUCLEO-NFC07A1 expansion board and NDEF application library.

The drivers abstract hardware low-level details and allow the component and applications to access NDEF data in a hardware-independent fashion.

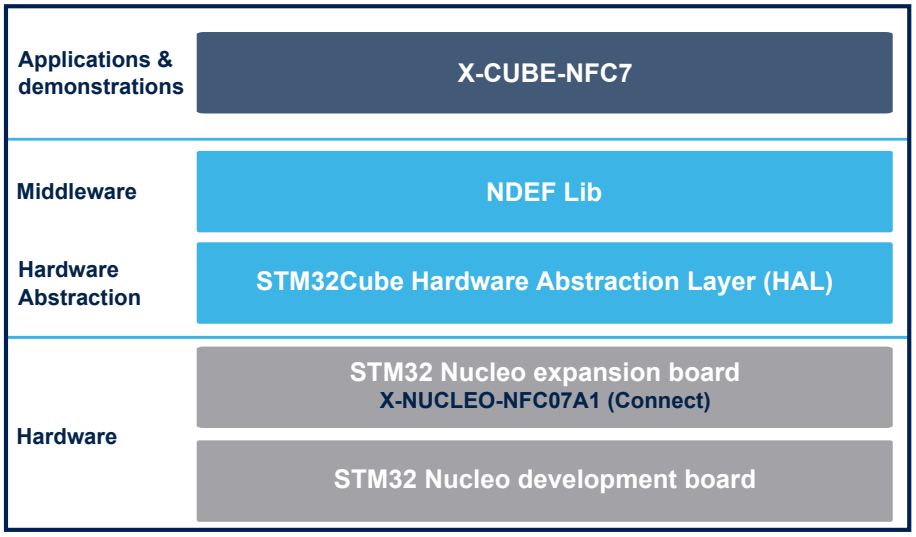
2.2 Architecture

This fully compliant software expansion for STM32Cube lets you develop applications using the ST25DV64KC dynamic NFC/RFID tag IC. It is based on the STM32CubeHAL hardware abstraction layer for the STM32 microcontroller and extends STM32Cube with a board support package (BSP) for the X-NUCLEO-NFC07A1 expansion board.

Application software can access and use the X-NUCLEO-NFC07A1 expansion board through the following layers:

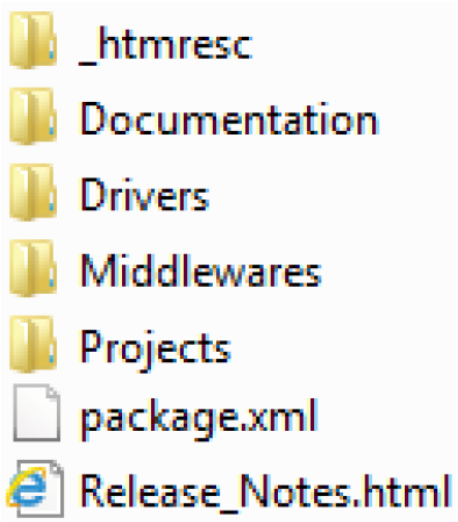
- **STM32Cube HAL layer**: the HAL driver layer provides a simple set of generic, multi-instance APIs (application programming interfaces) to interact with the upper layers (application, libraries, and stacks). These generic and extension APIs are directly built on a common architecture and allow overlying layers like middleware to implement their functions without depending on specific microcontroller unit (MCU) hardware information. This structure improves the library code reusability and guarantees easy portability across other devices.
- **Board support package (BSP) layer**: provides support for the peripherals on the STM32 Nucleo board (apart from the MCU). This set of APIs provides a programming interface for certain board-specific peripherals like the LED, the user button etc. This interface also helps you identify the specific board version.

Figure 1. **X-CUBE-NFC7 software architecture**



2.3 Folder structure

Figure 2. **X-CUBE-NFC7 package folders structure**



The X-CUBE-NFC7 software package includes the following folders:

- **Documentation**, which contains a compiled HTML file generated from the source code that details the software components and APIs.
- **Drivers**, which contains the HAL drivers, the board-specific drivers for each supported board or hardware platform, including the on-board components, and the CMSIS vendor-independent hardware abstraction layer for the Arm Cortex-M processor series.

- **Middlewares**, which contains NDEF application drivers and protocols related to NFC data communication:
  - NDEF AAR to add Android application record (AAR) to the tag
  - NDEF Email to manage the NDEF file that represents e-mails
  - NDEF Geo to manage the NDEF file that represents geo-location
  - NDEF MyApp to manage the NDEF file of a private application
  - NDEF SMS to manage the NDEF file that represents SMS
  - NDEF Text to manage the text NDEF file
  - NDEF URI to manage the URI NDEF file
  - NDEF Vcard to manage the Vcard NDEF file
- **Projects**, which contains the sample applications described in Section 2.1 for the NUCLEO-F401RE, NUCLEO-L053R8, or NUCLEO-L476RG platforms with three development environments (IAR Embedded Workbench for ARM, RealView MDK-ARM Microcontroller Development Kit, STM32CubeIDE).

2.4 APIs

Detailed technical information fully describing the APIs available to the user can be found in a compiled HTML file located inside the software package Documentation folder.

2.5 Sample applications

2.5.1 Energy harvesting

This sample shows how to enable the energy harvesting. You can either enable the EH dynamic (step 1) or static (step 2) register.

Step 1. Press the user button to enable the EH dynamic register.
This register allows energy harvesting until you reset the chip.
The blue LED is switched on when the dynamic register is enabled.

Step 2. Press the user button for more than 2 seconds to enable the EH static register.
This register allows energy harvesting by default each time the chip is powered.
The green LED is switched on when the static register is enabled.

2.5.2 Activate GPIO

This sample shows how to enable and use the GPO.

After initialization, an interrupt detects field changes in proximity of the ST25DV64KC. The green LED is switched on when the field is detected and switched off when the field disappears.

2.5.3 Set I²C protection

This sample shows how to create areas in the ST25DV64KC and how to protect them.

A UART console (via ST-LINK) displays the text when you connect a PC.

2.5.4 Use ST25DV64KC mailbox

This sample shows how to write a message into the mailbox and how to read the mailbox status register.

A UART console (via ST-LINK) displays the text when you connect a PC.

2.5.5 Write URI NDEF

This sample shows how to write an NDEF message to the ST25DV64KC EEPROM by using the NDEF lib middleware. The yellow LED switches on when the message has been successfully written.

You can read the tag URI message through an Android smartphone and the ST25 NFC app, available on Google Play.

3 System setup guide

3.1 Hardware description

3.1.1 STM32 Nucleo

STM32 Nucleo development boards provide an affordable and flexible way for users to test solutions and build prototypes with any STM32 microcontroller line.

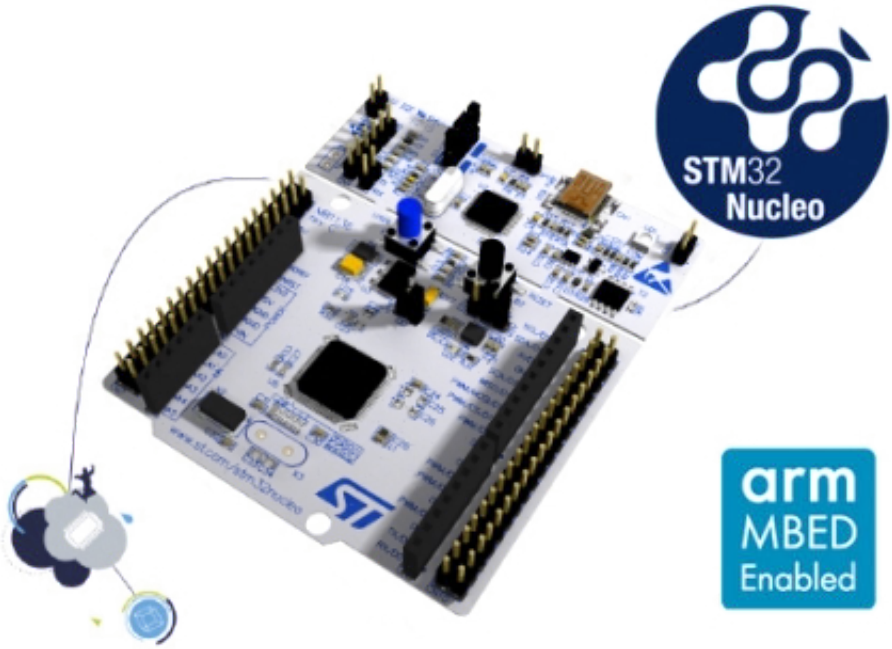
The Arduino connectivity support and ST morpho connectors make it easy to expand the functionality of the STM32 Nucleo open development platform with a wide range of specialized expansion boards to choose from.

The STM32 Nucleo board does not require separate probes as it integrates the ST-LINK/V2-1 debugger/programmer.

The STM32 Nucleo board comes with the comprehensive STM32 software HAL library together with various packaged software examples for different IDEs (IAR EWARM, Keil MDK-ARM, STM32CubeIDE, mbed and GCC/LLVM).

All STM32 Nucleo users have free access to the mbed online resources (compiler, C/C++ SDK and developer community) at www.mbed.org to easily build complete applications.

Figure 3. STM32 Nucleo board



3.1.2 X-NUCLEO-NFC07A1 expansion board

The X-NUCLEO-NFC07A1 dynamic NFC/RFID tag IC expansion board is based on the ST25DV64KC dynamic NFC/RFID tag IC with a 64-Kbit dual interface EEPROM and fast transfer mode feature. It can be powered through the STM32 Nucleo development board or directly through the received carrier electromagnetic field.

The X-NUCLEO-NFC07A1 expansion board is compatible with the Arduino UNO R3 connector pin assignment and can easily be plugged onto any STM32 Nucleo development board. You can stack other expansion boards to evaluate different devices that work together with the dynamic NFC tag.

The board also features a single-layer antenna with a diameter of 54 mm and copper etched on the PCB.

Figure 4. **X-NUCLEO-NFC07A1 expansion board**

3.2 Software description

You need the following software components to set up the suitable development environment for creating applications for the STM32 Nucleo equipped with the NFC expansion board:

- X-CUBE-NFC7 software expansion for STM32Cube dedicated to NFC applications development.
- Development tool-chain and Compiler. The STM32Cube expansion software supports the three following environments:
  - IAR Embedded Workbench for ARM® (EWARM) toolchain + ST-LINK
  - Keil Microcontroller Development Kit (MDK-ARM) toolchain + ST-LINK
  - STM32CubeIDE + ST-LINK

3.3 Hardware setup

You need the following hardware components:

- One STM32 Nucleo development platform (suggested order code: NUCLEO-F401RE, NUCLEO-L053R8, or NUCLEO-L476RG)
- One NFC/RFID tag IC expansion board (order code: X-NUCLEO-NFC07A1)
- One USB type A to mini-B USB cable to connect the STM32 Nucleo development board to the PC

3.4 Software setup

3.4.1 Development tool-chains and compilers

Select one of the integrated development environments (IDE) supported by the STM32Cube expansion software and read the system requirements and setup information provided by the IDE provider.

3.5 System setup

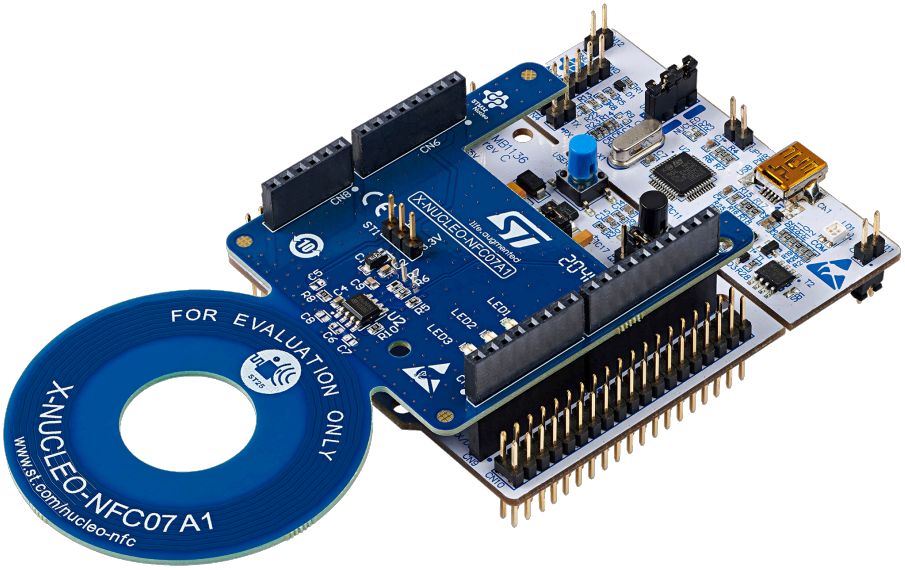
3.5.1 STM32 Nucleo and X-NUCLEO-NFC07A1 expansion board setup

The STM32 Nucleo board integrates the ST-LINK/V2-1 debugger/programmer. You can download the ST-LINK/V2-1 USB driver at STSW-LINK009.

You can easily plug the X-NUCLEO-NFC07A1 expansion board onto the STM32 Nucleo development board through the Arduino UNO R3 extension connector.

The expansion board can interface with the external STM32 microcontroller on STM32 Nucleo development board through the I²C transport layer.

Figure 5. X-NUCLEO-NFC07A1 expansion board plus STM32 Nucleo development board



Revision history

Table 2. **Document revision history**

Date	Revision	Changes
19-Jan-2022	1	Initial release.

Contents

List of tables

List of figures