

Getting started with the STMicroelectronics X-CUBE-NFC4 software package for STM32CubeMX

Introduction

This document provides the guidelines to configure and use the X-CUBE-NFC4 software package V1.3.0 for STM32CubeMX (minimum required version V5.1.0). The document contains a description of the provided sample applications, a description of the steps required to configure a generic project using the X-NUCLEO-NFC04A1 expansion board with a Nucleo board or ST25DV component with custom boards, as well as a description of the steps to configure and use the sample applications provided in the package. Information and documentation related to the NFC components, the X-NUCLEO-NFC04A1 expansion board and the ST expansion software for NFC tag are available on www.st.com.

Contents

Introduction	1
Contents.....	2
List of figures.....	3
1 Acronyms and abbreviations	4
2 What is STM32Cube?	5
3 License	5
4 Sample Applications and Examples Description	6
4.1 NFC04A1_NDEF_URI	6
4.2 NFC04A1_EnergyHarvesting	6
4.3 NFC04A1_GeneralPurposeOutput	6
4.4 NFC04A1_I2CPROTECTION	6
4.5 NFC04A1_LowPowerDown	6
4.6 NFC04A1_Mailbox	6
5 Installing the X-CUBE-NFC4 pack in STM32CubeMX.....	7
6 Starting a new project	8
7 STM32 Configuration Steps.....	10
7.1 Use of NFC component without NDEF Library for X-NUCLEO-NFC04A1	11
7.2 Use of NDEF Library with sample applications for X-NUCLEO-NFC04A1	13
7.3 Use of ST25DV component without sample applications for custom boards.....	19
8 Generated Folders Structure	24
9 Known Limitations and workarounds	25
9 References	26
10 Revision history	27

List of figures

Figure 1 Managing embedded software packs in STM32CubeMX	7
Figure 2 Installing the X-CUBE-NFC4 pack in STM32CubeMX	8
Figure 3 The X-CUBE-NFC4 pack in STM32CubeMX	8
Figure 4 STM32CubeMX main page	8
Figure 5 STM32CubeMX MCU/Board Selector windows	9
Figure 6 STM32CubeMX Pinout & Configuration window	9
Figure 7 STM32CubeMX Additional Software Components selection window	10
Figure 8 STM32 Nucleo 64 and Nucleo 144 pins with X-NUCLEO-NFC04A1	10
Figure 9 X-NUCLEO-NFC04A1 Pinout	11
Figure 10 STM32CubeMX Additional Software Components selection window	12
Figure 11 STM32CubeMX Pinout & Configuration tab and I2C settings	13
Figure 12 STM32CubeMX Additional Software Components selection window	14
Figure 13 STM32CubeMX Pinout & Configuration tab	15
Figure 14 STM32CubeMX Pinout & Configuration tab and Additional Software settings for NFC04A1_NDEF_URI applications	16
Figure 15 STM32CubeMX Pinout & Configuration tab and Additional Software settings for NFC04A1_I2CProtection example	16
Figure 16 STM32CubeMX Pinout & Configuration tab and Additional Software settings for NFC04A1_LowPowerDown example	17
Figure 17 STM32CubeMX Pinout & Configuration tab and Additional Software settings for NFC04A1_Mailbox example	17
Figure 18 STM32CubeMX NVIC Configuration	18
Figure 19 STM32CubeMX I2C Configuration	19
Figure 20 STM32CubeMX USART Configuration	19
Figure 21 STM32CubeMX Additional Software Components selection window	20
Figure 22 STM32CubeMX Pinout & Configuration tab	20
Figure 23 STM32CubeMX Parameter Settings for a custom board	21
Figure 24 STM32CubeMX NVIC Configuration	22
Figure 25 STM32CubeMX I2C Configuration	23
Figure 26 STM32CubeMX USART Configuration	24
Figure 27 STM32CubeMX Application Structure Configuration	25

1 Acronyms and abbreviations

Table 1: list of acronyms

Acronym	Description
NFC	Near Field Communication
NDEF	NFC data exchange format
RFID	Radio frequency identification
HAL	Hardware Abstraction Layer
I2C	Inter-Integrated Circuit
NVIC	Nested Vectored Interrupt Controller
RTC	Real Time Operating System
RTOS	Serial Peripheral Interface
U(S)ART	Universal (Synchronous) Asynchronous Receiver Transmitter
URI	Uniform resource identifier

2 What is STM32Cube?

STM32Cube™ represents an original initiative by STMicroelectronics to ease developers' life by reducing development effort, time and cost. STM32Cube covers the STM32 portfolio. Version 1.x of STM32Cube includes:

- STM32CubeMX, a graphical software configuration tool that allows the generation of C initialization code using graphical wizards.
- A comprehensive embedded software platform, delivered per series (such as the STM32CubeF4 for STM32F4 series).
 - STM32Cube HAL, an STM32 abstraction layer embedded software, ensuring maximized portability across the STM32 portfolio;
 - a consistent set of middleware components, such as RTOS, USB, TCP/IP, graphics;
 - all embedded software utilities, including a full set of examples.

3 License

The software provided in this package is licensed under [Software License Agreement SLA0077](#).

4 Sample Applications and Examples Description

In this section, a short overview of the sample applications and examples included in the X-CUBE-NFC4 pack is provided.

The sample applications/examples:

- are ready-to-use projects that can be generated through the STM32CubeMX for any Nucleo board and using the X-NUCLEO-NFC04A1 expansion board.
- show the users how to use the APIs to correctly initialize and use the dynamic NFC/RFID tag IC (ST25DV device).

4.1 NFC04A1_NDEF_URI

This application shows how to use the X-NUCLEO-NFC04A1 write an NDEF message to the ST25DV EEPROM using the NDEF lib middleware. The yellow LED is switched ON when the message has been successfully written.

4.2 NFC04A1_EnergyHarvesting

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

1. Press the user button to enable the EH dynamic register and allow energy harvesting until the chip is reset. The EH dynamic is enabled and allows energy harvesting until the chip is reset. The blue LED is switched ON when the dynamic register is enabled.
2. Press the user button for more than 2 seconds. The EH static register is enabled and allows energy harvesting by default each time the chip is powered. The green LED is switched ON when the static register is enabled.

4.3 NFC04A1_GeneralPurposeOutput

This example shows how to enable and use the GPO.

After initialization, an interrupt is programmed to detect field changes in proximity of the ST25DV. The green LED is switched ON when the field is detected and switched OFF when the field disappears.

4.4 NFC04A1_I2CPROTECTION

This example shows how to create areas in the ST25DV and how to protect them.

Text is displayed on a UART console (via ST-LINK) if a PC is connected. The example serial settings can be configured by user changing the settings of USART2 in the STM32CubeMX GUI.

4.5 NFC04A1_LowPowerDown

This example shows how to activate low power down (LPD) pin.

By pressing the user button, the LPD pin is activated (green LED is switched ON) or deactivated. When the LPD pin is activated, the ST25DV VCC is cut off, the power consumption is minimum and communication via I²C is not available.

4.6 NFC04A1_Mailbox

This example shows how to write a message into the mailbox and how to read mailbox status register of ST25DV device. Text is displayed on a UART console (via ST-LINK) if a PC is connected. The example serial settings can be configured by user changing the settings of USART2 in the STM32CubeMX GUI.

5 Installing the X-CUBE-NFC4 pack in STM32CubeMX

After downloading (from www.st.com), installing and launching the STM32CubeMX (V≥5.1.0), the X-CUBE-NFC4 pack can be installed in few steps.

1. From the menu, select Help > Manage embedded software packages

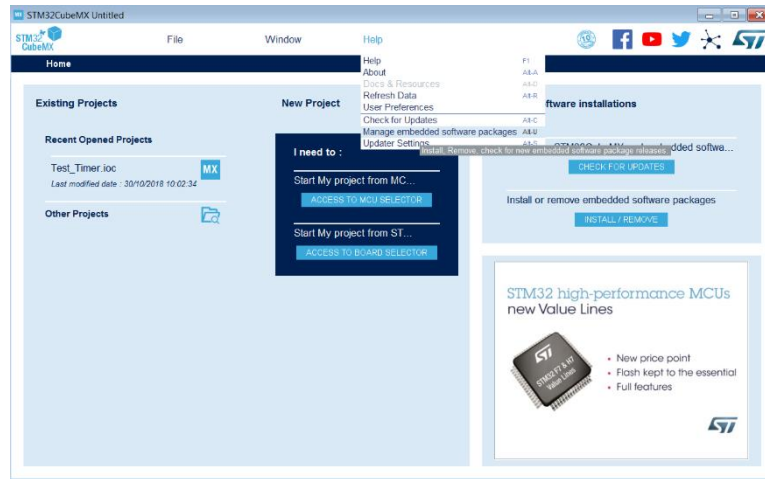


Figure 1 Managing embedded software packs in STM32CubeMX

2. From the Embedded Software Packages Manager window, press the 'Refresh' button to get an updated list of the add-on packs. Go to the 'STMicroelectronics' tab to find the X-CUBE-NFC4 pack.

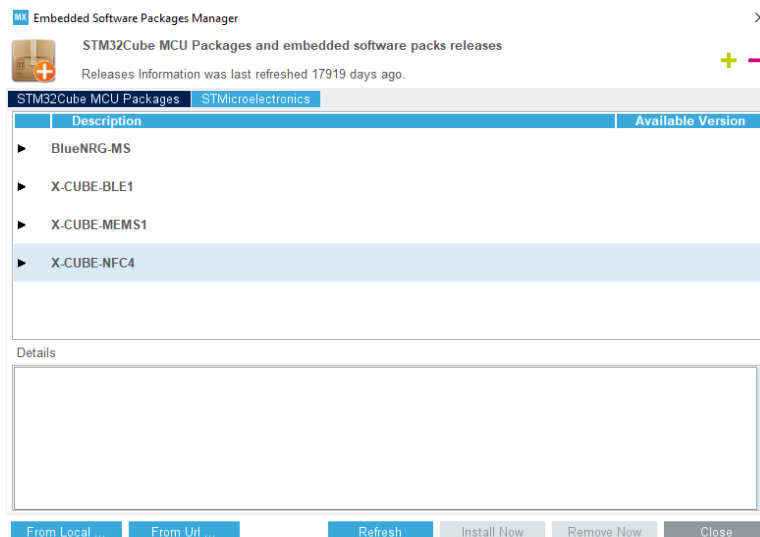


Figure 2 Installing the X-CUBE-NFC4 pack in STM32CubeMX

3. Select it checking the corresponding box and install it pressing the 'Install Now' button. Once the installation is completed, the corresponding box will become green, the 'Close' button can be pressed and the configuration of a new project can start.

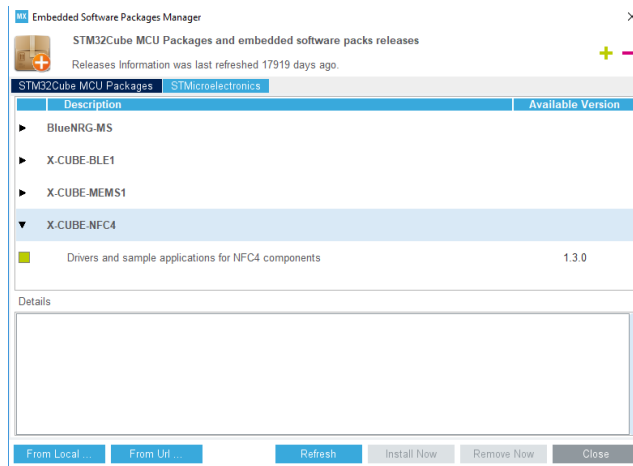


Figure 3 The X-CUBE-NFC4 pack in STM32CubeMX

6 Starting a new project

After launching the STM32CubeMX, you can choose if starting a [New Project](#) from the MCU Selector or from the Board Selector.

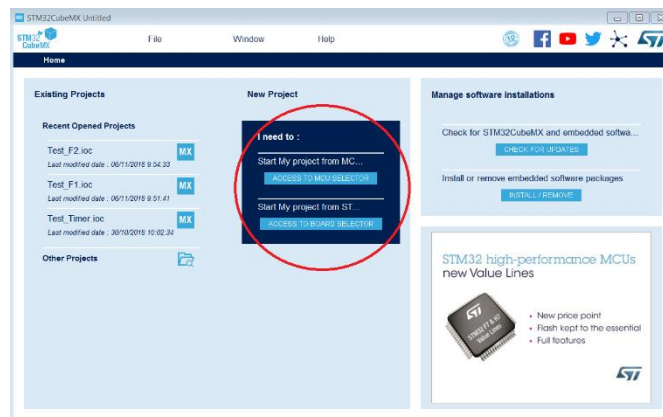


Figure 4 STM32CubeMX main page

The **MCU/Board selector** window will pop up. From this window, the STM32 MCU or platform can be selected.

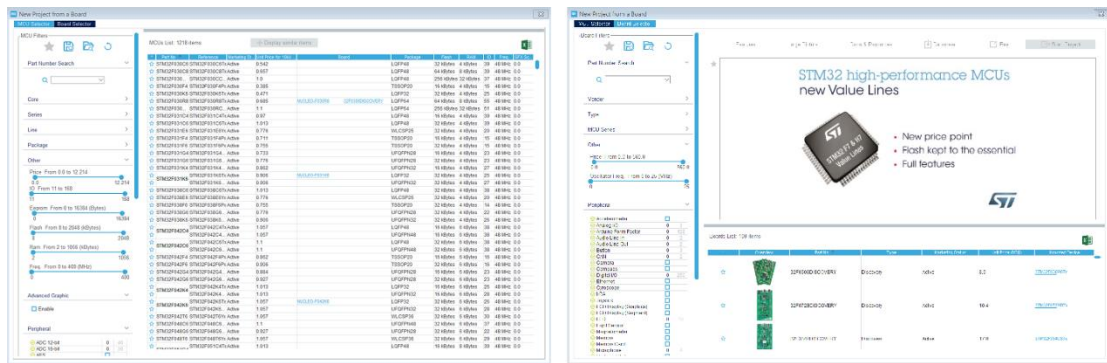


Figure 5 STM32CubeMX MCU/Board Selector windows

After selecting the MCU or the Board, the selected STM32 pinout will appear. From this window the user can set up the project, by adding one or more Additional Software and peripherals and configuring the clock.

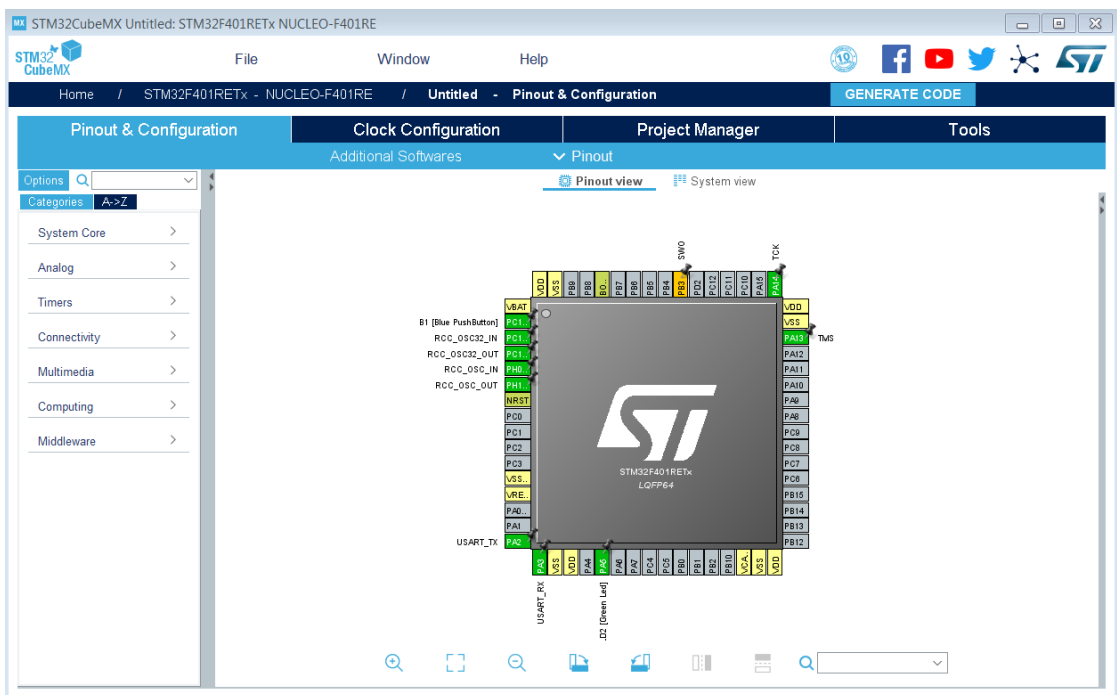


Figure 6 STM32CubeMX Pinout & Configuration window

To add the X-CUBE-NFC4 additional software to the project, the “Additional Softwares” button must be clicked.

From the Additional Software Component Selection window, the user can either choose to generate, for the selected MCU/Board, one of the enclosed sample applications or a new project. In this latter case, the user must just implement the main application logic without bothering with the pinout and peripherals configuration code that will be automatically generated by STM32CubeMX.

Additional Software Components selection									
Vendor	Pack/Bundle	Pack Version	Class	Pack Action	Group/Subgroup	Selection	Condition	Status	Description
> STMicroelectronics	X-CUBE-BLE1/Application	4.2.0	Wireless	Installed					BLE stack and sample applications for BlueNRG-MS module
> STMicroelectronics	X-CUBE-BLE1/Application	1.1.0	Wireless	Installed					BLE stack and sample applications for BlueNRG-MS module
> STMicroelectronics	X-CUBE-BLE1/BlueNRG-MS	1.1.0	Wireless	Installed					BLE stack and sample applications for BlueNRG-MS module
> STMicroelectronics	X-CUBE-BLE1/BlueNRG-MS	4.2.0	Wireless	Installed					BLE stack and sample applications for BlueNRG-MS module
> STMicroelectronics	X-CUBE-MEMS1/Application	5.2.1	Device	Installed					Drivers and sample applications for MEMS components
> STMicroelectronics	X-CUBE-MEMS1/MEMS	5.2.1	Board Extension	Installed					Drivers and sample applications for MEMS components
> STMicroelectronics	X-CUBE-MEMS1/MEMS	5.2.1	Board Component	Installed					Drivers and sample applications for MEMS components
> STMicroelectronics	X-CUBE-MEMS1/STM32Cube_Custom_BSP_Drivers	5.2.1	Board Support	Installed					Drivers and sample applications for MEMS components
> STMicroelectronics	X-CUBE-NFC4/Application	1.3.0	Device	Installed					Drivers and sample applications for NFC4 components
STMicroelectronics	X-CUBE-NFC4/Application	1.3.0	Device		Application	--Select--			
> STMicroelectronics	X-CUBE-NFC4/NFC	1.3.0	Board Component	Installed		--Select--			Drivers and sample applications for NFC4 components
> STMicroelectronics	X-CUBE-NFC4/NFC	1.3.0	Board Extension	Installed		NFC04A1_NDEF_URI			Drivers and sample applications for NFC4 components
> STMicroelectronics	X-CUBE-NFC4/NFC	1.3.0	Board Support	Installed		NFC04A1_EnergyHarvesting			Drivers and sample applications for NFC4 components
> STMicroelectronics	X-CUBE-NFC4/STM32Cube_Custom_BSP_Drivers	1.3.0	Board Support	Installed		NFC04A1_GeneralPurposeOut			Drivers and sample applications for NFC4 components
> STMicroelectronics	X-CUBE-NFC4/lib_nfc	1.3.0	Wireless	Installed		NFC04A1_I2CProtection			Drivers and sample applications for NFC4 components
						NFC04A1_LowPowerDown			Drivers and sample applications for NFC4 components
						NFC04A1_MAILBOX			
						NFC04A1_SERIALUSB			

Figure 7 STM32CubeMX Additional Software Components selection window

7 STM32 Configuration Steps

The X-NUCLEO-NFC04A1 interfaces with the STM32 microcontroller via the I2C bus. Hence, assuming a user wants to interface the ST X-NUCLEO-NFC04A1 expansion board with a STM32 Nucleo 64 pins board (e.g. a Nucleo-F401RETx) no particular hardware modification must be done.

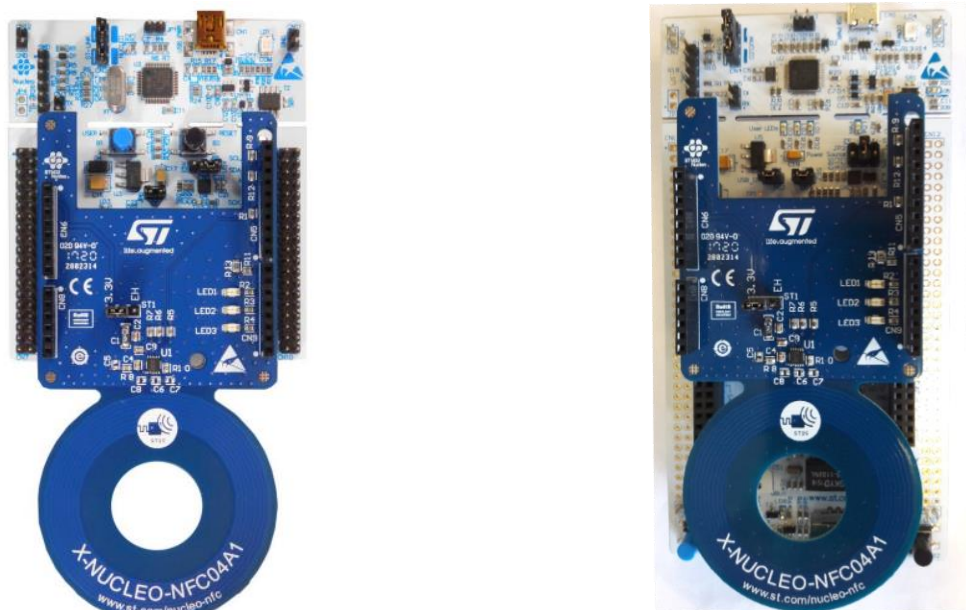


Figure 8 STM32 Nucleo 64 and Nucleo 144 pins with X-NUCLEO-NFC04A1

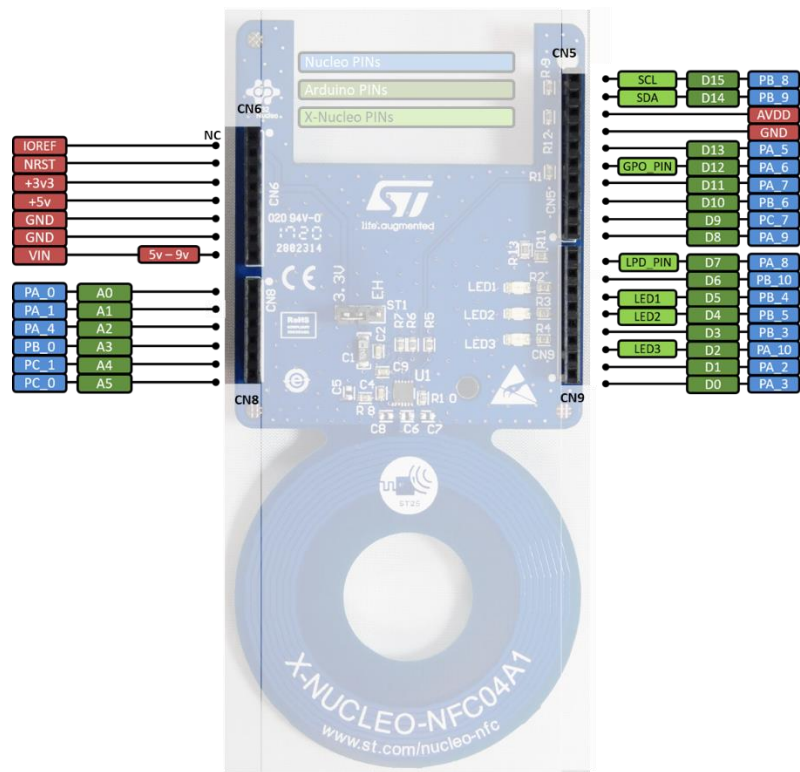


Figure 9 X-NUCLEO-NFC04A1 pinout

7.1 Use of NFC component without NDEF Library for X-NUCLEO-NFC04A1

This section outlines how to configure STM32CubeMX with X-NUCLEO-NFC04A1 when the use of the sample example is required without the dependencies over NDEF library. With such setup, only driver layers will be configured.

To add the X-CUBE-NFC4 additional software to the project, the “Additional Softwares” button must be clicked. From the “Additional Software Components selection” window, the user has to select the example from the “Device” class and “Board Extension” class as shown in the figure below.

Vendor	Pack/Bundle	Pack Version	Class	Pack Action	Group/Subgroup	Selection	Condition	Status	Description
STMicroelectronics	X-CUBE-BLE1/Application	1.1.0	Wireless	Installed					BLE stack and sample applications for BlueNRG-MS...
STMicroelectronics	X-CUBE-BLE1/Application	4.2.0	Wireless	Installed					BLE stack and sample applications for BlueNRG-MS...
STMicroelectronics	X-CUBE-BLE1/BlueNRG-MS	1.1.0	Wireless	Installed					BLE stack and sample applications for BlueNRG-MS...
STMicroelectronics	X-CUBE-BLE1/BlueNRG-MS	4.2.0	Wireless	Installed					BLE stack and sample applications for BlueNRG-MS...
STMicroelectronics	X-CUBE-MEMS1/Application	5.2.1	Device	Installed					Drivers and sample applications for MEMS compone...
STMicroelectronics	X-CUBE-MEMS1/MEMS	5.2.1	Board Extension	Installed					Drivers and sample applications for MEMS compone...
STMicroelectronics	X-CUBE-MEMS1/MEMS	5.2.1	Board Component	Installed					Drivers and sample applications for MEMS compone...
STMicroelectronics	X-CUBE-MEMS1/STM32Cube_Custom_BSP_Drivers	5.2.1	Board Support	Installed					Drivers and sample applications for MEMS compone...
STMicroelectronics	X-CUBE-NFC4/Application	1.3.0	Device	Installed					Drivers and sample applications for NFC4 components
STMicroelectronics	X-CUBE-NFC4/Application	1.3.0	Device	Application	NFC04A1_LowPow...	NFC04A...			Low Power Down feature of ST25DV
STMicroelectronics	X-CUBE-NFC4/NFC	1.3.0	Board Extension	Installed					Drivers and sample applications for NFC4 components
STMicroelectronics	X-CUBE-NFC4/NFC	1.3.0	Board Extension	NFC04A1/		NFC04A...			BSPTAG for X-NUCLEO-NFC04A1, ST25DV
STMicroelectronics	X-CUBE-NFC4/NFC	1.3.0	Board Component	Installed					Drivers and sample applications for NFC4 components
STMicroelectronics	X-CUBE-NFC4/STM32Cube_Custom_BSP_Drivers	1.3.0	Board Support	Installed					Drivers and sample applications for NFC4 components

Figure 10 STM32CubeMX Additional Software Components selection window

From the **Pinout & Configuration** tab:

- from the **Pinout** scheme, click on PB8 and set it as I2C1_SCL;
- from the **Pinout** scheme, click on PB9 and set it as I2C1_SDA;
- enable the I2C1 as I2C from the “Connectivity” category;
- Configure the I2C1 settings with I2C speed at 400KHz (Fast Mode) from the “Configuration” view;

From the **Pinout** scheme set:

Nucleo 64		Nucleo 144	
PB8	I2C1_SCL	PB8	I2C1_SCL
PB9	I2C1_SDA	PB9	I2C1_SDA

Home / STM32F401RETx - NUCLEO-F401RE / LPD.Ioc - Pinout & Configuration
GENERATE CODE

Pinout & Configuration
Clock Configuration
Project Manager
Tools

Additional Softwares
Pinout

Options
Categories
A-Z

Connectivity
I2C1
I2C2
I2C3
SDIO
SPI1
SPI2
SPI3
USART1
USART2
USART6
USB_OTG_FS
Multimedia
Computing
Middleware
Additional Software

Mode
Board Extension NFC
Device Application

Configuration
Reset Configuration
Parameter Settings
Platform Settings

Name	IPs or Components	Found Solutions
YELLOW LED	GPIO:Output	PA10
BLUE LED	GPIO:Output	PB5
GREEN LED	GPIO:Output	PB4

Name	IPs or Components	Found Solutions
NFC4 GPO PIN	GPIO:EXTI	PA6
NFC4 LPD PIN	GPIO:Output	PA8
BSP BUTTON	GPIO:EXTI	PC13-ANTI_TAMP [B1] E

Pinout view
System view

Figure 11 STM32CubeMX Pinout & Configuration tab and I2C settings

From the **Additional Software** category, press the 'Stmicroelectronics.X-CUBE-NFC4.1.3.0' item, enable the "Board Extension NFC" checkbox from the "Mode" view and set the following Platform Settings from the "Configuration" view (take into account that according the example chosen some settings can appear or not):

Name	BSP_Api	Supported IPs	Nucleo 64	Nucleo 144
NFC04A1 BUS IO driver	BSP_BUS_DRIVER	I2C:I2C	I2C1	I2C1
NFC4 GPO PIN	HAL_EXTI_DRIVER	GPIO:EXTI	PA6	PA6
NFC4 LPD PIN		GPIO:Output	PA8	PF13
YELLOW LED		GPIO:Output	PA10	PF15
BLUE LED		GPIO:Output	PB5	PF14
GREEN LED		GPIO:Output	PB4	PE11
BSP USART	BSP_COMMON_DRIVER	USART:Asynchronous	USART2	USART3
BSP BUTTON	BSP_COMMON_DRIVER	GPIO:EXTI	PC13	PC13

Once all the above described steps have been performed, the source code of the project using the **STMicroelectronics X-CUBE-NFC4** software can be generated clicking the "GENERATE CODE" button.

7.2 Use of NDEF Library with sample applications for X-NUCLEO-NFC04A1

This section outlines how to configure STM32CubeMX with X-NUCLEO-NFC04A1 when the use of the sample applications is required. With such setup, all the components of the expansion software package, including applications, will be properly configured.

To add the X-CUBE-NFC4 additional software to the project, the "Additional Softwares" button must be clicked. From the "Additional Software Components selection" window, the user has to select the "Board Extension" class, NFC04A1_NDEF_URI application from the "Device" class and select the "Wireless" class with "Basic" Interface as shown in the figure below.

Vendor	Pack/Bundle	Pack Version	Class	Pack Action	Group/Subgroup	Selection	Condition	Status	Description
> STMicrelectronics	X-CUBE-BLE1/BlueNRG-MS	4.2.0	Wireless	Installed					BLE stack and sample applications for BlueNRG-MS module
> STMicrelectronics	X-CUBE-MEMS1/Application	5.2.1	Device	Installed					Drivers and sample applications for MEMS components
> STMicrelectronics	X-CUBE-MEMS1/MEMS	5.2.1	Board Extension	Installed					Drivers and sample applications for MEMS components
> STMicrelectronics	X-CUBE-MEMS1/MEMS	5.2.1	Board Component	Installed					Drivers and sample applications for MEMS components
> STMicrelectronics	X-CUBE-MEMS1/STM32Cube_Custom_BSP_Drivers	5.2.1	Board Support	Installed					Drivers and sample applications for MEMS components
> STMicrelectronics	X-CUBE-NFC4/Application	1.3.0	Device	Installed					Drivers and sample applications for NFC4 components
■ STMicrelectronic	X-CUBE-NFC4/Application	1.3.0	Device		Application	NFC04A1_NDEF_URI	NFC04A...	⊙	URI NDEF sample application for X-NUCLEO-NFC04A1
> STMicrelectronics	X-CUBE-NFC4/NFC	1.3.0	Board Component	Installed					Drivers and sample applications for NFC4 components
> STMicrelectronics	X-CUBE-NFC4/NFC	1.3.0	Board Extension	Installed					Drivers and sample applications for NFC4 components
■ STMicrelectronic	X-CUBE-NFC4/NFC	1.3.0	Board Extension		NFC04A1/	<input checked="" type="checkbox"/>	NFC04A...	⊙	BSPTAG for X-NUCLEO-NFC04A1, ST25DV
> STMicrelectronics	X-CUBE-NFC4/STM32Cube_Custom_BSP_Drivers	1.3.0	Board Support	Installed					Drivers and sample applications for NFC4 components
> STMicrelectronics	X-CUBE-NFC4/lib_nfc	1.3.0	Wireless	Installed					Drivers and sample applications for NFC4 components
■ STMicrelectronic	X-CUBE-NFC4/lib_nfc	1.3.0	Wireless		Core/	<input checked="" type="checkbox"/>	libNDEF ...	⊙	Single component for libNDEF. Condition to any NFCTAG B...
■ STMicrelectronic	X-CUBE-NFC4/lib_nfc	1.3.0	Wireless		Interface	Basic	libNDEF ...	⊙	Single component for libNDEF. Condition to any NFCTAG B...

Figure 12 STM32CubeMX Additional Software Components selection window

From the **Pinout & Configuration** tab:

- from the **Pinout** scheme, click on PB8 and set it as I2C1_SCL;
- from the **Pinout** scheme, click on PB9 and set it as I2C1_SDA;
- enable the I2C1 as I2C from the “Connectivity” category;

From the **Pinout** scheme, if not already set, set:

Nucleo 64			Nucleo 144		
PIN	Mode	Label	PIN	Mode	Label
PB5	GPIO_Output		PF14	GPIO_Output	
PB4	GPIO_Output		PE11	GPIO_Output	
PA5	GPIO_Output	LD2 [Green Led]	PB7	GPIO_Output	LD2[Blue]
PC13	GPIO_EXTI13	B1 [Blue PushButton]	PC13	GPIO_EXTI13	USER_Btn[B1]

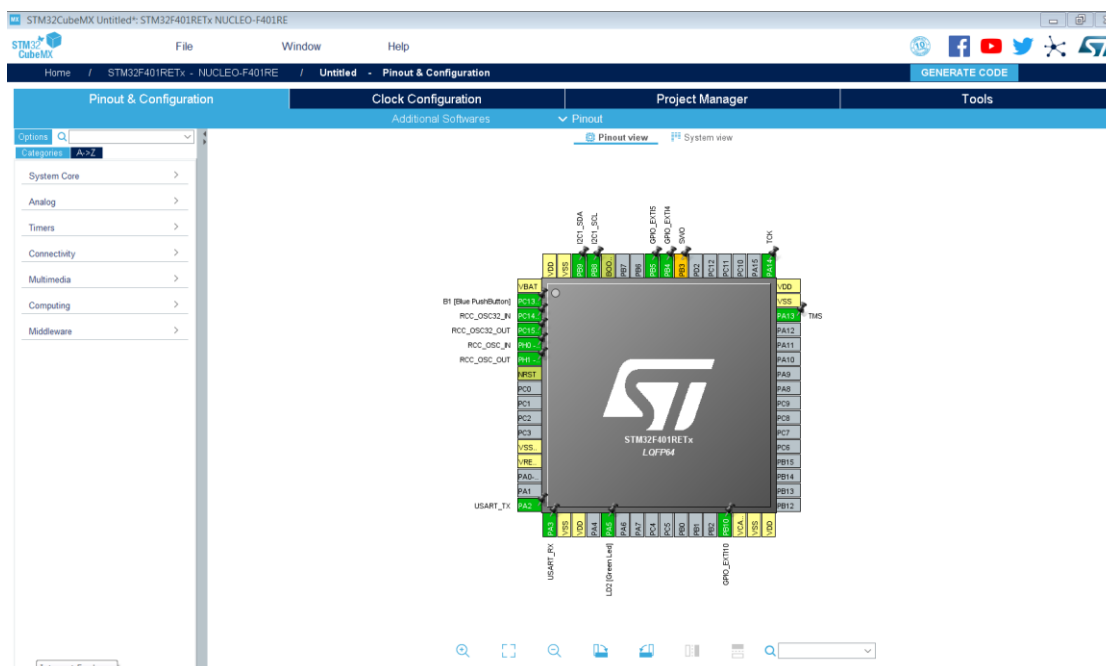


Figure 13 STM32CubeMX Pinout & Configuration tab

From the **Additional Software** category, press the 'Stmicroelectronics.X-CUBE-NFC4.1.3.0' item, enable the "Board Extension NFC" and the "Device Application" checkboxes from the "Mode" view and set the following Platform Settings from the "Configuration" view (take into account that according the example chosen some settings can appear or not):

Name	BSP_Api	Supported IPs	Nucleo 64	Nucleo 144
NFC04A1 BUS IO driver	BSP_BUS_DRIVER	I2C:I2C	I2C1	I2C1
NFC4 GPO PIN	HAL_EXTI_DRIVER	GPIO:EXTI	PA6	PA6
NFC4 LPD PIN		GPIO:Output`	PA8	PF13
YELLOW LED		GPIO:Output	PA10	PF15
BLUE LED		GPIO:Output	PB5	PF14
GREEN LED		GPIO:Output	PB4	PE11

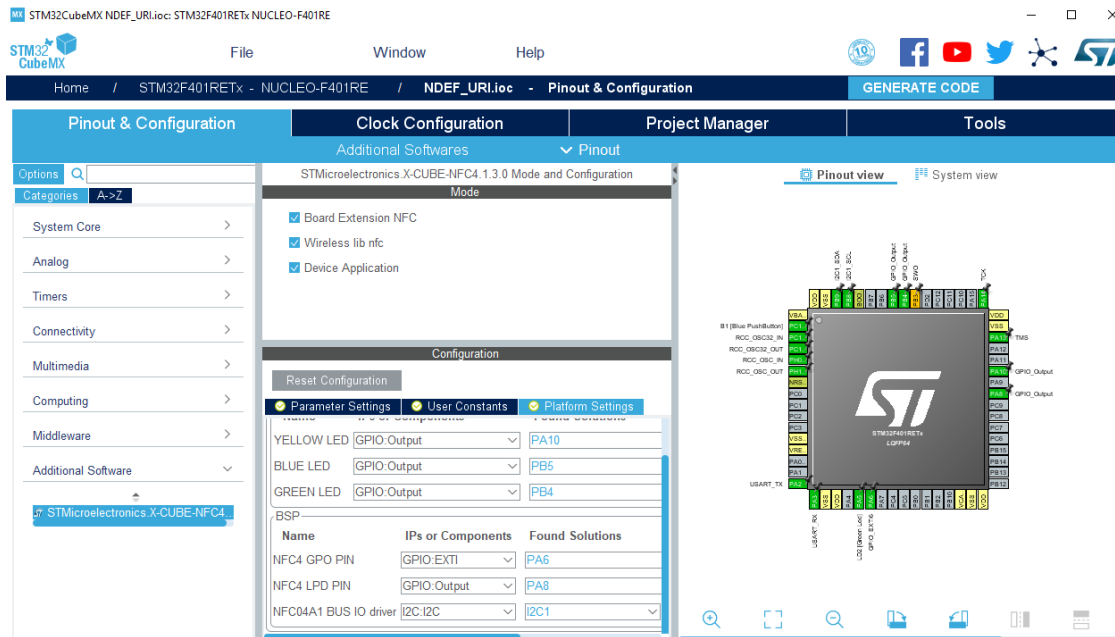


Figure 14 STM32CubeMX Pinout & Configuration tab and Additional Software settings for NFC04A1_NDEF_URI applications

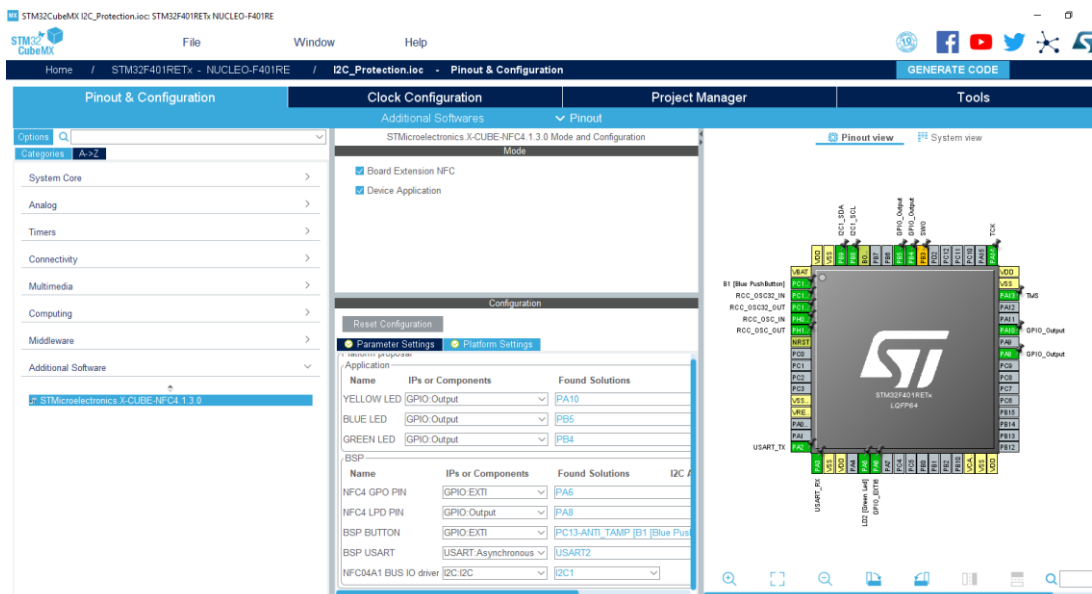


Figure 15 STM32CubeMX Pinout & Configuration tab and Additional Software settings for NFC04A1_I2CProtection example

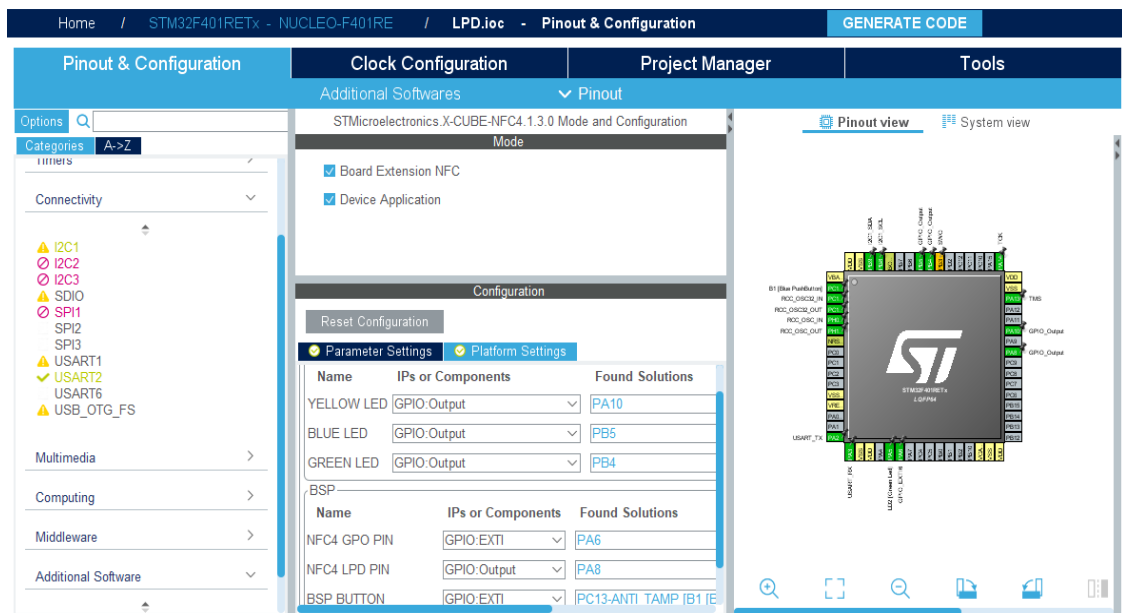


Figure 16 STM32CubeMX Pinout & Configuration tab and Additional Software settings for NFC04A1_LowPowerDown example

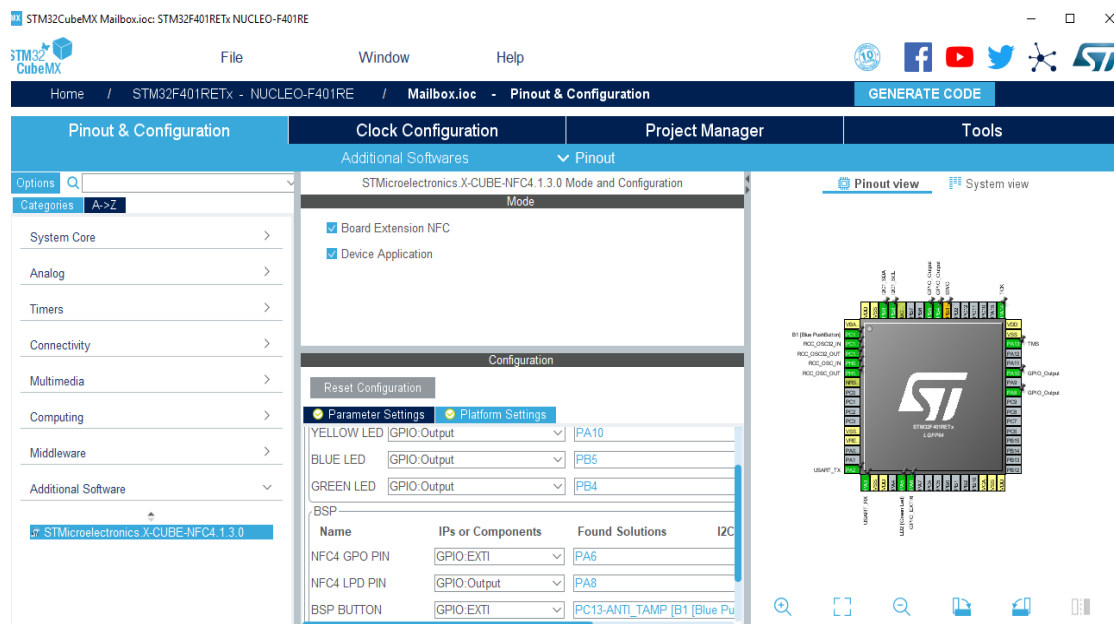


Figure 17 STM32CubeMX Pinout & Configuration tab and Additional Software settings for NFC04A1_Mailbox example

For all the sample applications, the default parameters can be used.

From the **Configuration & Pinout** tab, click on “System Core” category and then on NVIC item to enable the EXTI line interrupts:

Nucleo 64	Nucleo 144
EXTI line 6 interrupt	EXTI line 6 interrupt
EXTI line 13 interrupt	EXTI line 13 interrupt

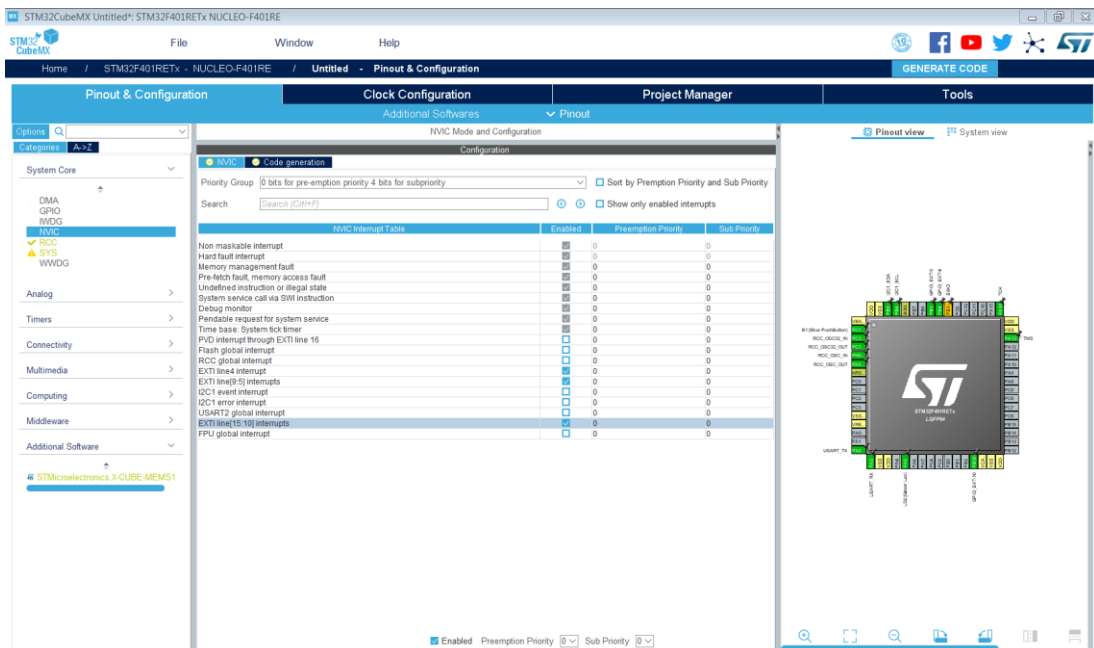


Figure 18 STM32CubeMX NVIC Configuration

From the **Configuration & Pinout** tab, click on “Connectivity” category and then on I2C1 item to set the I2C speed at 400KHz or 1MHz (for STM32L0/STM32L4 families). If STM32L0/STM32L4 MCU families are used, kindly set the Coefficient of Digital Filter to 2 in Parameter settings:

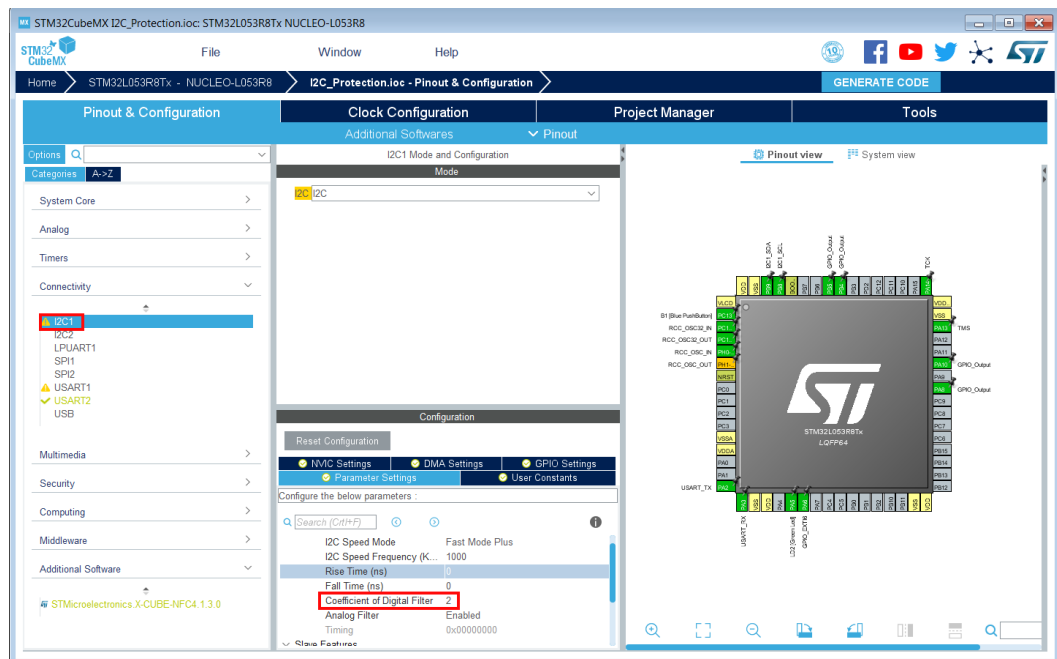


Figure 19 STM32CubeMX I2C Configuration

From the **Configuration & Pinout** tab, click on “Connectivity” category and then on USART2 item and check that the following configuration is set:

Baud Rate	115200 Bits/s
Word Length	8 Bits (including Parity)
Parity	None
Stop Bits	1

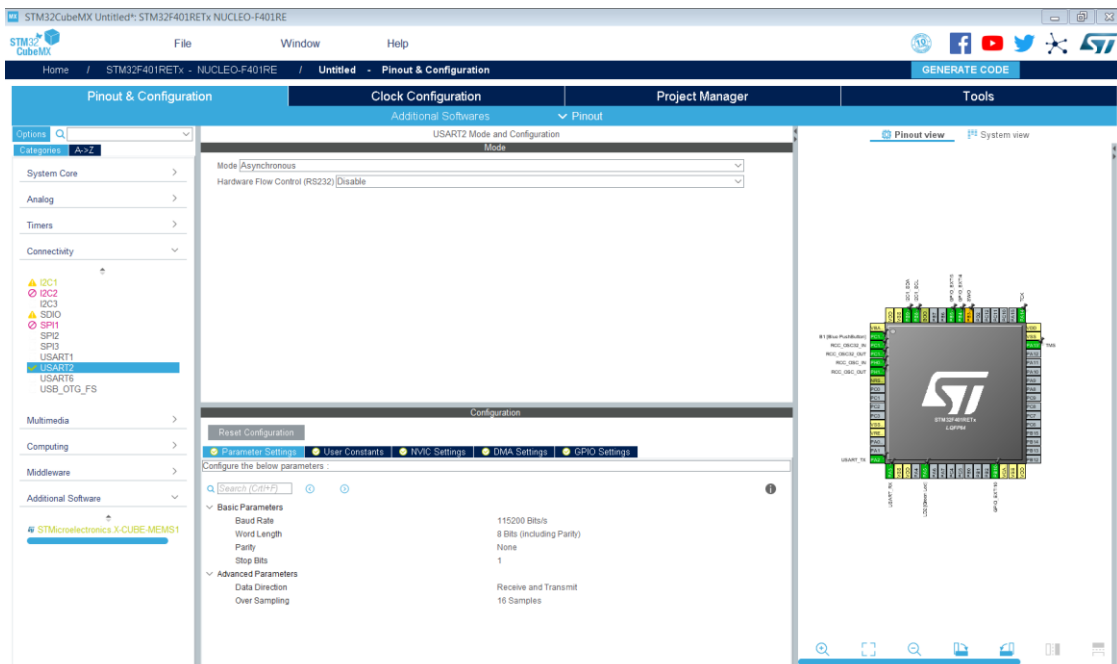


Figure 20 STM32CubeMX USART Configuration

Once all the above described steps have been performed, the sample applications for NFC04A1 using the **STMicroelectronics X-CUBE-NFC4** software can be generated clicking the “GENERATE CODE” button.

7.3 Use of ST25DV component without sample applications for custom boards

This section outlines how to configure STM32CubeMX with a custom board that mounts ST25DV device. With such setup, only drivers layer will be configured. This setup is useful when user does not intend to leverage the sample application provided in the package, to develop his own application code. In this case you can configure the ST25DV device in order to be used. To add the X-CUBE-NFC4 additional software to the project, the “Additional Softwares” button must be clicked. From the “Additional Software Components selection” window, the user has to select the “Board Components” class and the “Board Support” class in “STM32Cube_Custom_BSP_Drivers” bundle as shown in figure below.

Additional Software Components selection									
Vendor	Pack/Bundle	Pack Version	Class	Pack Action	Group/Subgroup	Selection	Condition	Status	
> STMicorelectronics	X-CUBE-BLE1/Application	4.2.0	Wireless	Installed					BLE stack
> STMicorelectronics	X-CUBE-BLE1/BlueNRG-MS	1.1.0	Wireless	Installed					BLE stack
> STMicorelectronics	X-CUBE-BLE1/BlueNRG-MS	4.2.0	Wireless	Installed					BLE stack
> STMicorelectronics	X-CUBE-MEMS1/Application	5.2.1	Device	Installed					Drivers and
> STMicorelectronics	X-CUBE-MEMS1/MEMS	5.2.1	Board Extension	Installed					Drivers and
> STMicorelectronics	X-CUBE-MEMS1/MEMS	5.2.1	Board Component	Installed					Drivers and
> STMicorelectronics	X-CUBE-MEMS1/STM32Cube_Custom_BSP_Drivers	5.2.1	Board Support	Installed					Drivers and
> STMicorelectronics	X-CUBE-NFC4/Application	1.3.0	Device	Installed					Drivers and
> STMicorelectronics	X-CUBE-NFC4/NFC	1.3.0	Board Extension	Installed					Drivers and
> STMicorelectronics	X-CUBE-NFC4/NFC	1.3.0	Board Component	Installed					Drivers and
> STMicorelectronics	X-CUBE-NFC4/NFC	1.3.0	Board Component	Installed	NFC4/ST25DV	<input checked="" type="checkbox"/>	ST25DV Condition		NFC4 librar
> STMicorelectronics	X-CUBE-NFC4/STM32Cube_Custom_BSP_Drivers	1.3.0	Board Support	Installed					Drivers and
> STMicorelectronics	X-CUBE-NFC4/STM32Cube_Custom_BSP_Drivers	1.3.0	Board Support	Installed	Custom/NFCTAG	<input checked="" type="checkbox"/>	NFCTAG Condition		BSP for NF
> STMicorelectronics	X-CUBE-NFC4/lib_nfc	1.3.0	Wireless	Installed					Drivers and

Figure 21 STM32CubeMX Additional Software Components selection window

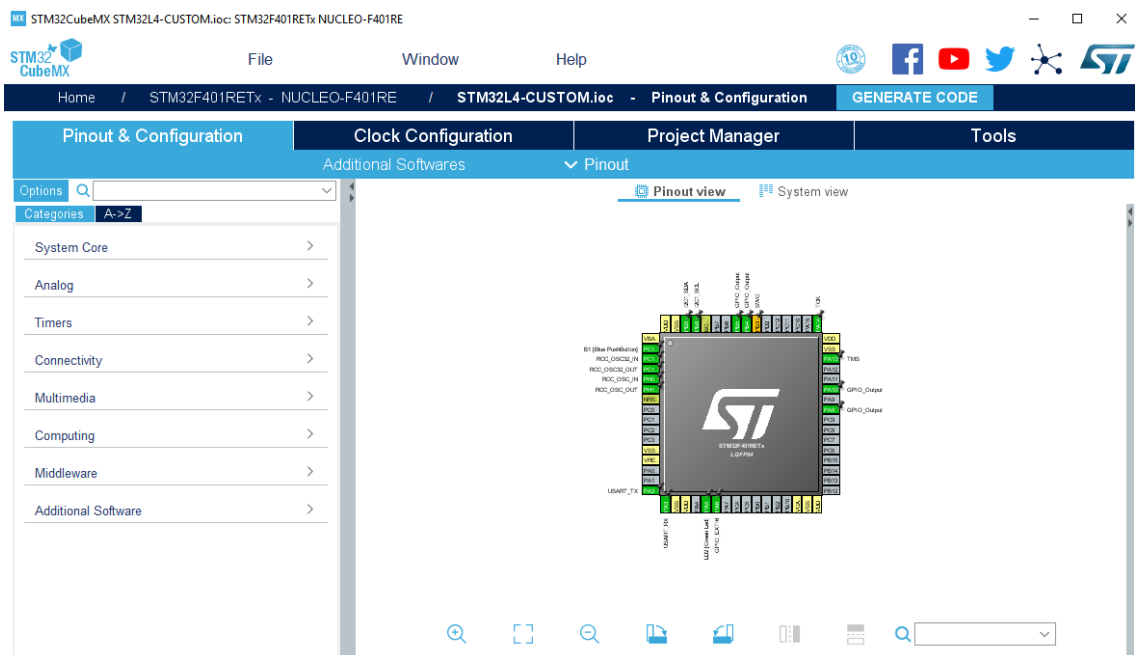


Figure 22 STM32CubeMX Pinout & Configuration tab

From the **Pinout & Configuration** tab:

- from the **Pinout** scheme, click on PB8 and set it as I2C1_SCL;
- from the **Pinout** scheme, click on PB9 and set it as I2C1_SDA;
- enable the I2C1 as I2C from the “Connectivity” category;

From the **Pinout** scheme, if not already set, set:

Nucleo 64			Nucleo 144		
PIN	Mode	Label	PIN	Mode	Label
PB5	GPIO_Output		PF14	GPIO_Output	
PB4	GPIO_Output		PE11	GPIO_Output	
PA5	GPIO_Output	LD2 [Green Led]	PB7	GPIO_Output	LD2[Blue]
PC13	GPIO_EXTI13	B1 [Blue PushButton]	PC13	GPIO_EXTI13	USER_Btn[B1]

. You can see an example of configuration below.

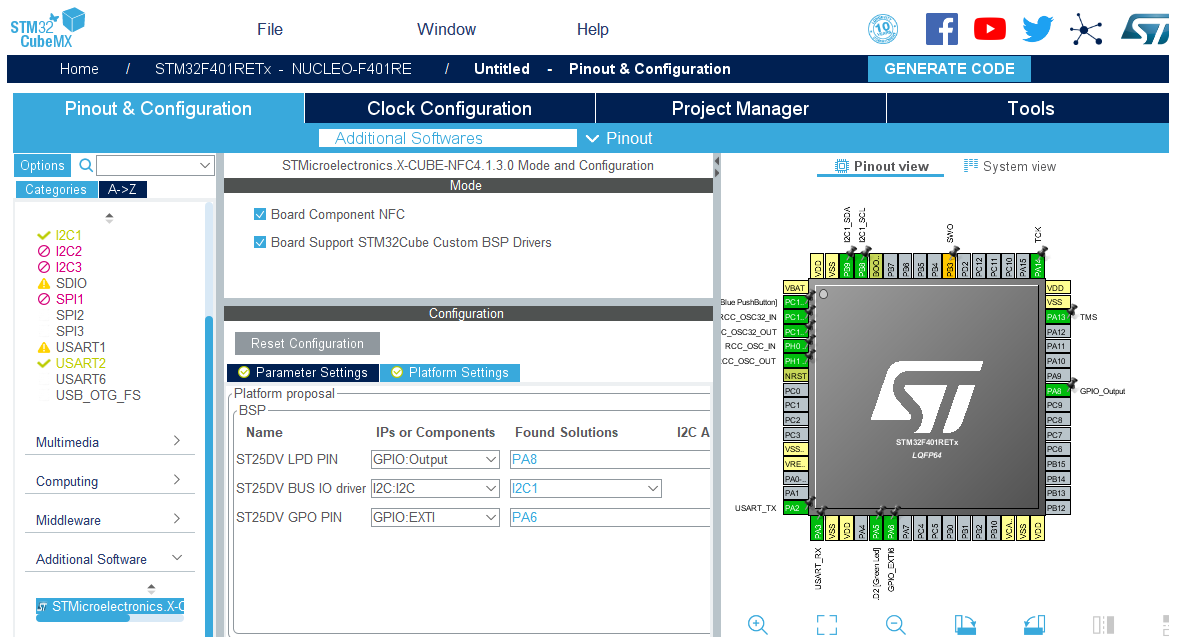


Figure 23 STM32CubeMX Parameter Settings for a custom board

Set the following Platform Settings from the “Configuration” view:

Name	BSP_Api	Supported IPs	Nucleo 64	Nucleo 144
ST25DV BUS IO driver	BSP_BUS_DRIVER	I2C:I2C	I2C1	I2C1
ST25DV GPO PIN	HAL_EXTI_DRIVER	GPIO:EXTI	PA6	PA6
ST25DV LPD PIN		GPIO:Output	PA8	PF13

From the **Configuration & Pinout** tab, click on “System Core” category and then on NVIC item to enable the EXTI line interrupts.

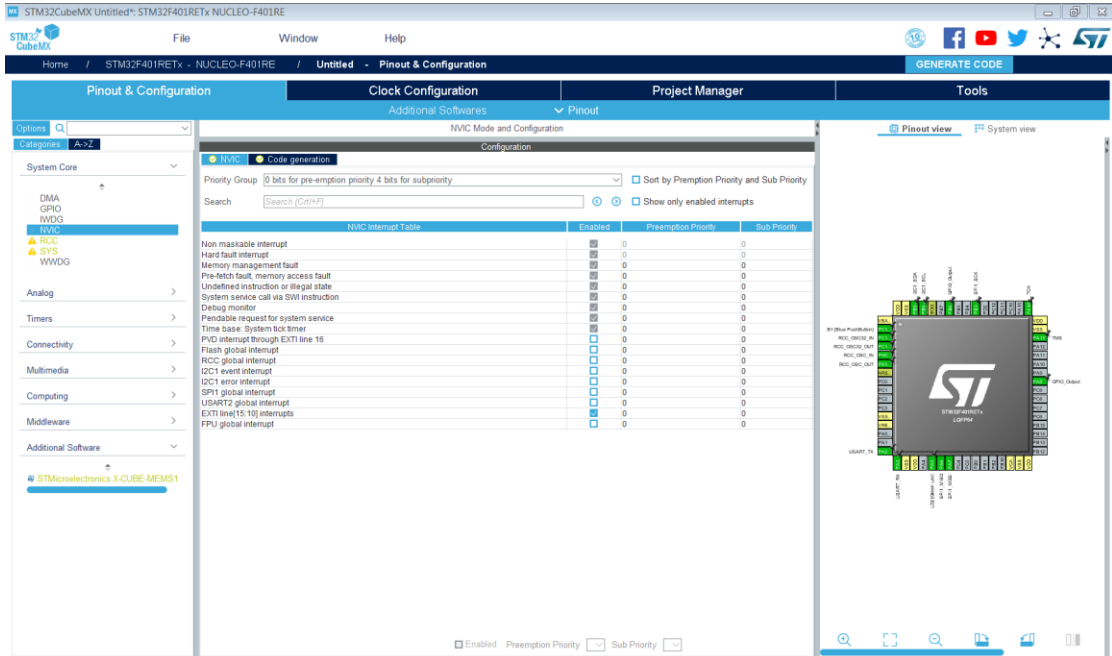


Figure 24 STM32CubeMX NVIC Configuration

From the **Configuration & Pinout** tab, click on “Connectivity” category and then on I2C1 item to set the I2C speed at 400KHz (Fast Mode):

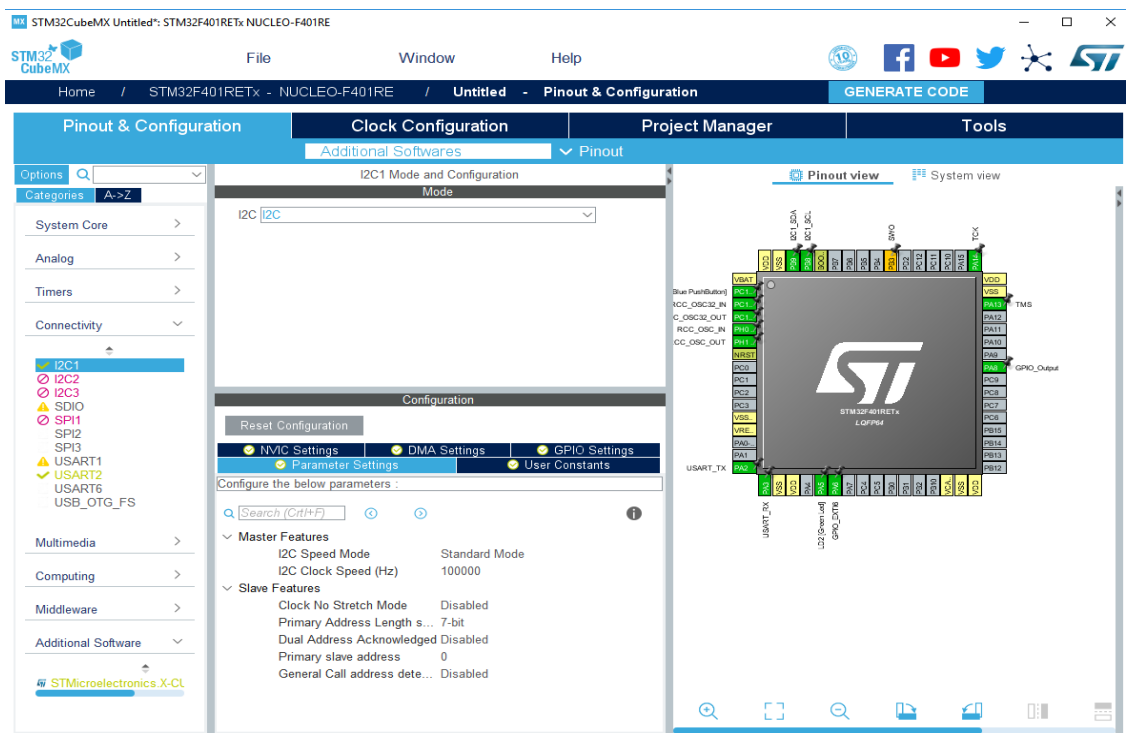


Figure 25 STM32CubeMX I2C Configuration

From the **Configuration & Pinout** tab, click on “Connectivity” category and then on USART2 button and check the following configuration is set:

Baud Rate	115200 Bits/s
Word Length	8 Bits (including Parity)
Parity	None
Stop Bits	1

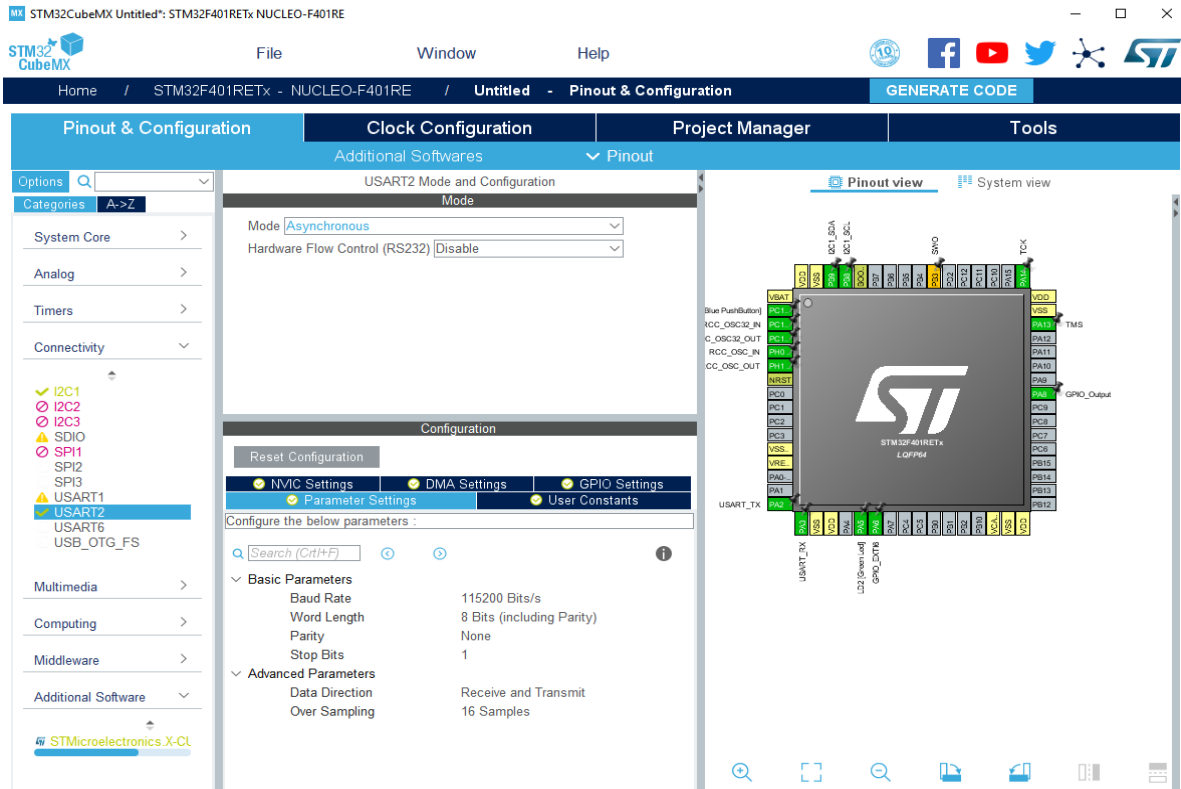


Figure 26 STM32CubeMX USART Configuration

Once all the above described steps have been performed, the sample applications for a custom board using the **STMicroelectronics X-CUBE-NFC4** software can be generated clicking the “GENERATE CODE” button.

8 Generated Folders Structure

When generating a project, two models of folders structure can be adopted when using a high level firmware component (i.e. a middleware in the STM32Cube MCU package):

- **Basic Structure:** the basic structure is often used with HAL examples and single package projects. This structure consists of having the IDE configuration folder in the same level as the sources (organized in *Inc* and *Src* subfolders).
- **Advanced Structure:** the advanced structure provides a more efficient and organized folders model that allows ease middleware applications integration when several packages are used.

In the Advanced mode *Src* and *Inc* are generated under folder *Core*.

For each package, the list of the generated files is under *<Package_Name>* (*X-CUBE-NFC4* for the X-CUBE-NFC4 pack), at the same level as *Core* and containing inside the *App* and the *Target* subfolder.

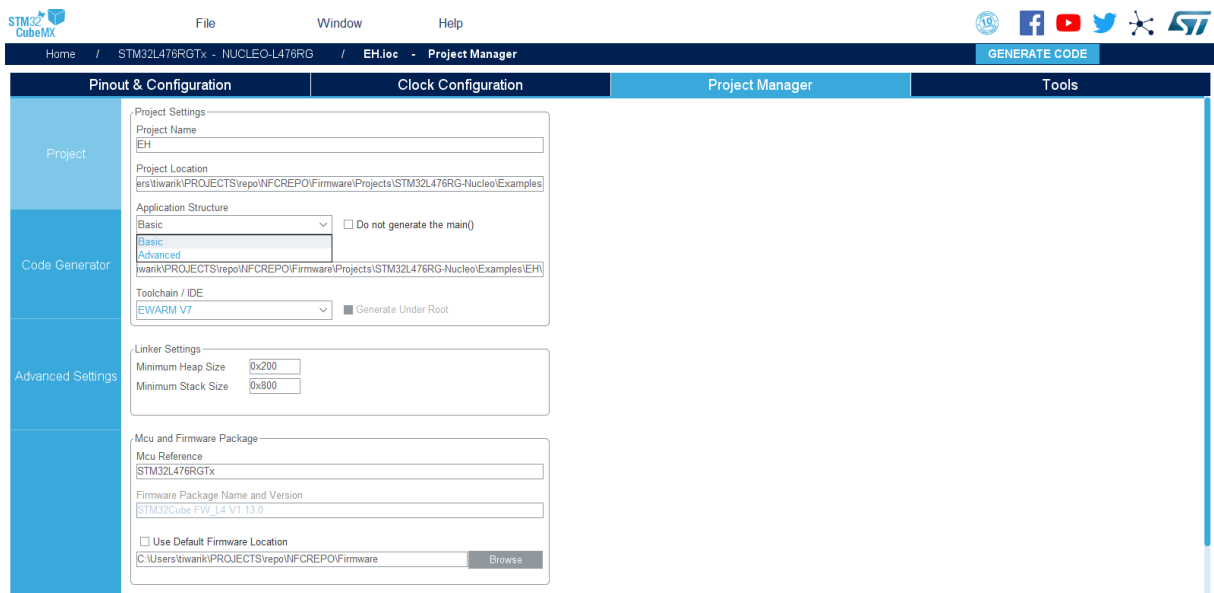


Figure 27 STM32CubeMX Application Structure Configuration

9 Known Limitations and workarounds

STM32CubeMX NFC4 pack v1.3.0 is fully compatible with STM32CubeMX v5.1.0 and upwards. It is not fully compatible with previous version of STM32CubeMX ($\leq v5.0.1$).

The example Serial-USB_PCSoftware is not currently integrated in STM32CubeMX.

Custom Board generation is currently not supported in this version of the NFC4 pack.

9 References

[1] [UM2239](#) – User Manual - *Getting started with the X-CUBE-NFC4 dynamic NFC/RFID tag IC software expansion for STM32Cube*

10 Revision history

Table 2: Document revision history

Date	Version	Changes
31-Jan-2019	1	Initial release

IMPORTANT NOTICE – PLEASE READ CAREFULLY

STMicroelectronics NV and its subsidiaries ("ST") reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice. Purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST's terms and conditions of sale in place at the time of order acknowledgement.

Purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of Purchasers' products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

ST and the ST logo are trademarks of ST. All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.

© 2018 STMicroelectronics – All rights reserved