

CubeMX 创建 WL LoRa EndNode 应用

关键字: STM32WL, LoRa

前言

本篇 LAT 介绍使用 CubeMX 创建基于采用 STM32WLE5JC/STM32WL55JC 的 STM32WL 板子的 LoRaWAN 应用。

其中, 参考使用了 STM32Cube_FW_WL_V1.0.0\Projects\NUCLEO-WL55JC\Applications\LoRaWAN\LoRaWAN_End_Node 创建一个 WLE5Jx_EndNode 的例子。

关于不同 MCU 间的移植可参考

UM1718 11.9 Switching to another MCU

UM1718 15 Tutorial 5: Exporting current project configuration to a compatible MCU

需要提前准备的环境:

STM32WL 官网		https://www.st.com/STM32WL https://www.stmcu.com.cn/STM32WL
硬件	HW	NUCLEO-WL55JC https://www.st.com/en/evaluation-tools/nucleo-wl55jc.html MB1389-WL55JC-highband-D04 Schematic MB1389-WL55JC-lowband-D04 Schematic
代码生成工具	CubeMX	STM32CubeMX v6.2.0 https://www.st.com/en/development-tools/stm32cubemx.html
软件源码库	CubeWL	STM32Cube_FW_WL_V1.0.0 https://www.st.com/en/embedded-software/stm32cubewl.html
集成开发环境	IAR	8.50.9 (以上版本不需要打补丁) https://netstorage.iar.com/SuppDB/Public/UPDINFO/015020/arm/doc/infocenter/readme.ENU.html

IAR v8.50.9 以下版本需要打补丁

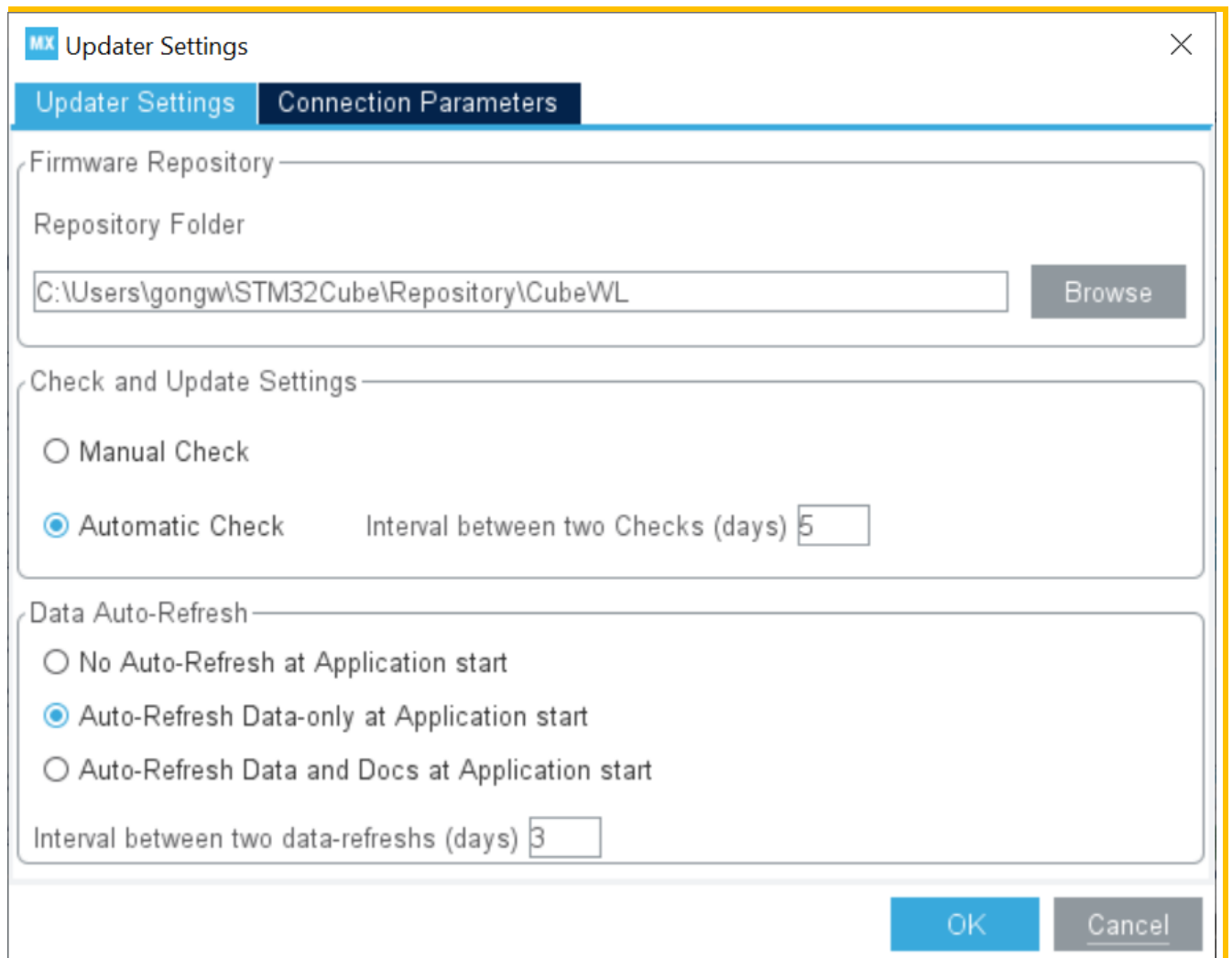
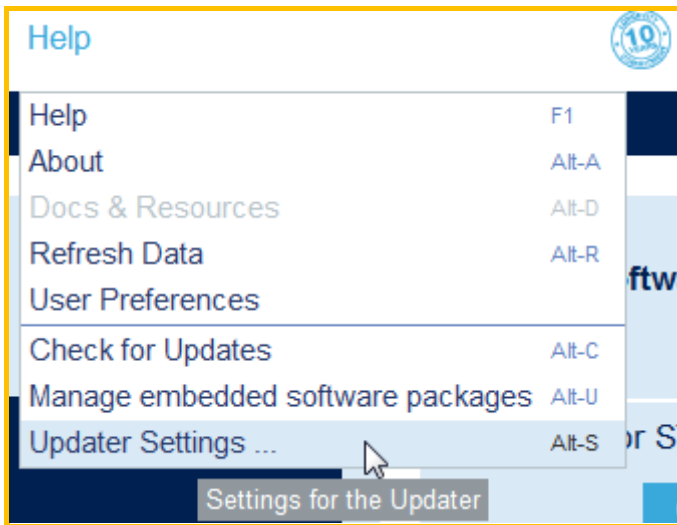
STM32Cube_FW_WL_V1.0.0\Utilities\PC_Softwar\EWARMv8_STM32WLxx_V4.6.zip

1 配置 CubeMX

1.1 配置 CubeMX 并下载 CubeWL

可以使用 CubeMX 默认安装的仓库目录

也可以自定义仓库目录，本例使用自定义的仓库目录 `C:\Users\gongw\STM32Cube\Repository\CubeWL`，如下
 Help=>Updater Settings=>Repository Folder=>`C:\Users\gongw\STM32Cube\Repository\CubeWL`=>OK



选择 Manage software installations 中的 Install or remove embedded software packages 下的
 “INSTALL / REMOVE”

Manage software installations

Check for STM32CubeMX and embedded software packages updates

CHECK FOR UPDATES

Install or remove embedded software packages

INSTALL / REMOVE

MX Embedded Software Packages Manager ✕

STM32Cube MCU Packages and embedded software packs releases + -

Releases Information was last refreshed less than one hour ago.

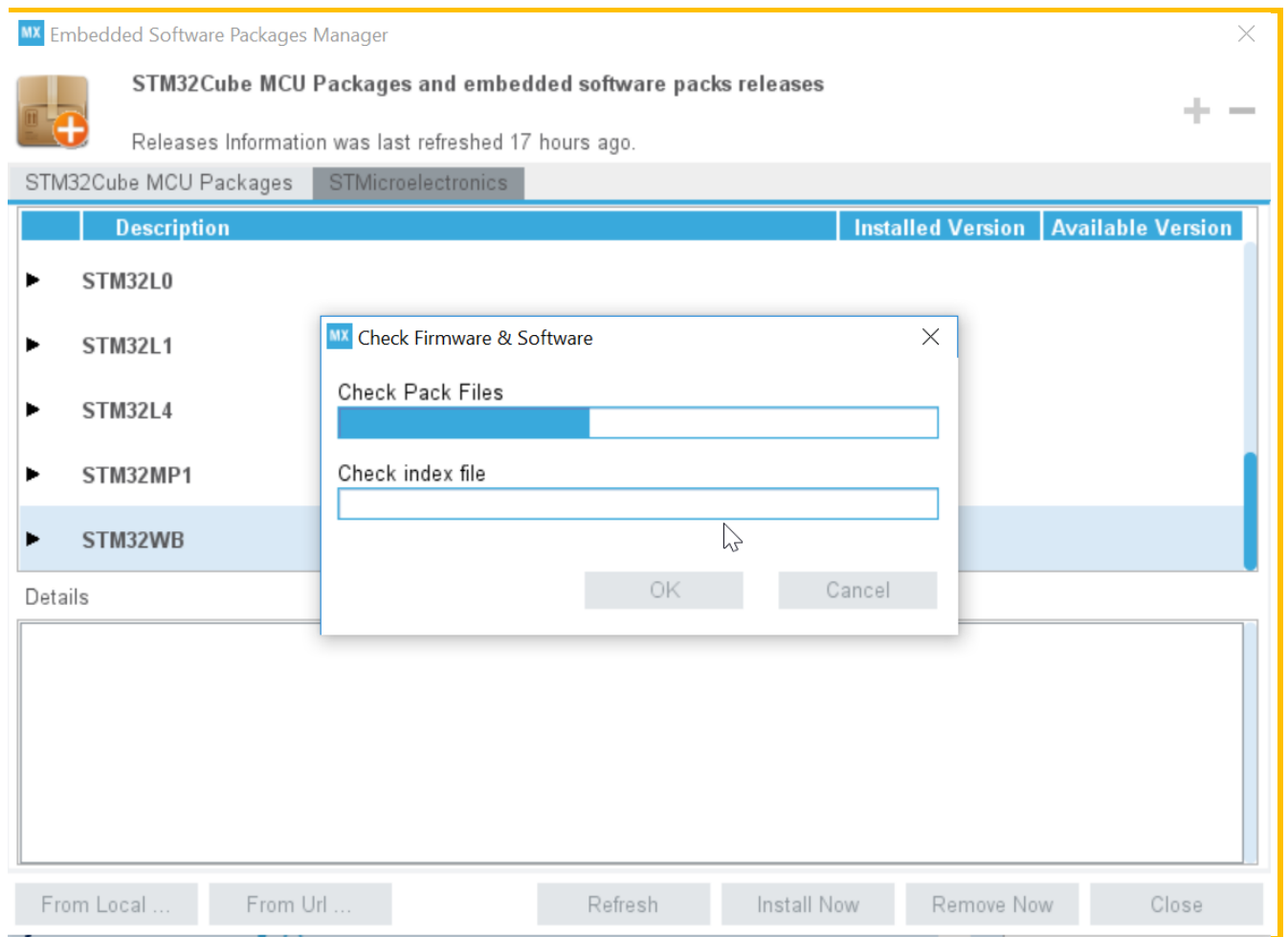
STM32Cube MCU Packages
STMicroelectronics
ARM

	Description	Installed Version	Available Version
▶	STM32L5		
▶	STM32MP1		
▶	STM32WB		
▶	STM32WL		

Details

From Local ...
From Url ...
Refresh
Install Now
Remove Now
Close

点 Refresh 更新数据库



Fresh 结束后，点开 STM32WL 左侧的黑色三角，选中 STM32WL 下版本 1.0.0 的 STM32Cube MCU Package for STM32WL Series，点 Install Now

MX Embedded Software Packages Manager

STM32Cube MCU Packages and embedded software packs releases

Releases Information was last refreshed less than one hour ago.

Description	Installed Version	Available Version
▶ STM32MP1		
▶ STM32WB		
▼ STM32WL		
STM32Cube MCU Package for STM32WL Series	1.0.0	1.0.0

Details

[STM32CubeWL Firmware Package V1.0.0 / 28-October-2020](#)

Main Changes

- CMSIS/LL/HAL Drivers with full Quality criteria (MISRA-C@ 2012 & CodeSonar)
- All Middlewares Legacy (FatFS, FreeRTOS)
- All Middlewares RF (LoRaWAN, Sigfox, SubGhz_Phy)
- All Middlewares Security (STM32_Secure_Engine, STM32_Key_Management_Services, mbed-crypto)
- All Utilities
- CubeFW contains 254 examples & applications with FW/ARM as basic techniques, ported to MDK/ARM & CubeIDE

From Local ... From Url ... Refresh Install Now Remove Now Close

点击 Close

1.2 CubeMX 新建项目

打开 STM32CubeMX，在 New Project 下有三中选择，**请根据需求选择 New Project 的方式！**

New Project

I need to :

- 1 Start My project from MCU
ACCESS TO MCU SELECTOR
- 2 Start My project from ST Board
ACCESS TO BOARD SELECTOR
- 3 Start My project from Example
ACCESS TO EXAMPLE SELECTOR

1.2.1 Start My project from MCU

第 1 种可适用于所有的 WL 产品，只能生成和配置的用户自定义(User Defined)的 LoRaWAN/SigFox/SUBGHZ 应用，不能生成 CubeWL 中 AT_Slave/EndNode/PingPong demo

1.2.2 Start My project from BOARD

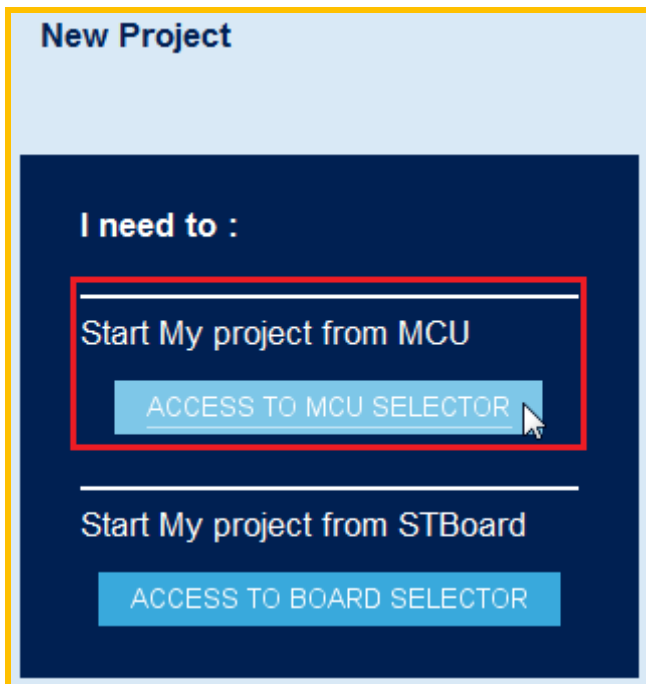
第 2 种适用于 NUCLEO-WL55JC1 和 NUCLEO-WL55JC1 两种 Demo 板，只能生成和配置的用户自定义(User Defined)的 LoRaWAN/SigFox/SUBGHZ 应用，不能生成 CubeWL 中 AT_Slave/EndNode/PingPong demo

1.2.3 Start My project from EXAMPLES

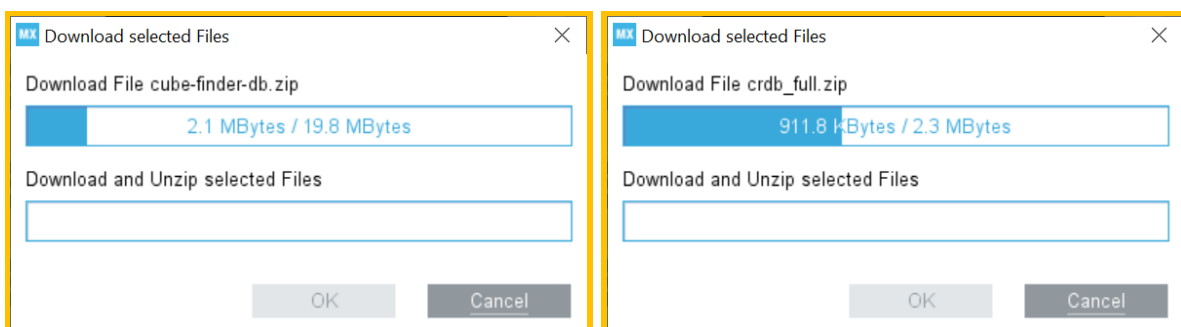
第 3 种适用于生成 CubeWL 中 NUCLEO-WL55JC1 和 NUCLEO-WL55JC1 的 Examples，可生成和配置 CubeWL 中 AT_Slave/EndNode/PingPong demo

1.2.4 Start My project from MCU STM32WLEx

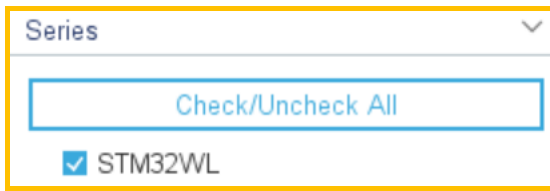
在 New Project 下选择，"Start My project from MCU "下的"ACCESS TO MCU SELECTOR"



此过程中可能会更新些数据库，如



选择 Series 下的 STM32WL



根据自己的需求选择 STM32WLE5Jx 中的任意一款

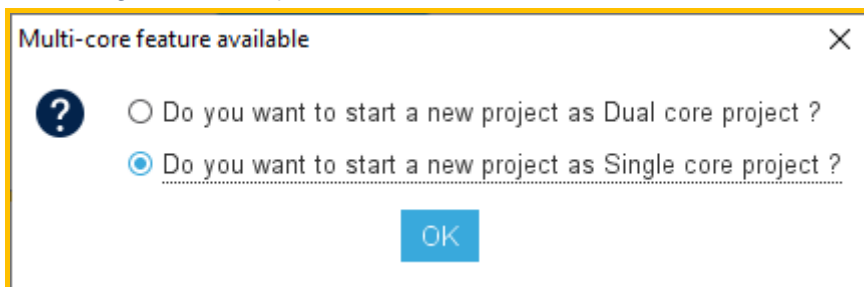
MCUs/MPUs List: 19 items + Display similar items Export

☆	Part No	Reference	Marketing St...	Board	Package	Flash	RAM	IO	Freq.
☆	STM32WL54CC	STM32...	Active		UFQFPN48	256 kBytes	20 kBytes	29	48 MHz
☆	STM32WL54JC	STM32...	Active		UFBGA73	256 kBytes	64 kBytes	43	48 MHz
☆	STM32WL55CC	STM32...	Active		UFQFPN48	256 kBytes	64 kBytes	29	48 MHz
☆	STM32WL55JC	STM32...	Active	NUCLEO-WL55JC1/NUCLEO-WL55JC2	UFBGA73	256 kBytes	64 kBytes	43	48 MHz
☆	STM32WL55UC	STM32...	NA		WLCSP59	256 kBytes	64 kBytes	22	48 MHz
☆	STM32WLE4C8	STM32...	NA		UFQFPN48	64 kBytes	64 kBytes	29	48 MHz
☆	STM32WLE4CB	STM32...	NA		UFQFPN48	128 kBytes	64 kBytes	29	48 MHz
☆	STM32WLE4CC	STM32...	Active		UFQFPN48	256 kBytes	64 kBytes	29	48 MHz
☆	STM32WLE4J8	STM32...	NA		UFBGA73	64 kBytes	20 kBytes	43	48 MHz
☆	STM32WLE4JB	STM32...	NA		UFBGA73	128 kBytes	48 kBytes	43	48 MHz
☆	STM32WLE4JC	STM32...	Active		UFBGA73	256 kBytes	64 kBytes	43	48 MHz
☆	STM32WLE5C8	STM32...	Active		UFQFPN48	64 kBytes	20 kBytes	29	48 MHz
☆	STM32WLE5CB	STM32...	Active		UFQFPN48	128 kBytes	48 kBytes	29	48 MHz
☆	STM32WLE5CC	STM32...	Active		UFQFPN48	256 kBytes	64 kBytes	29	48 MHz
☆	STM32WLE5J8	STM32...	Active		UFBGA73	64 kBytes	20 kBytes	43	48 MHz
☆	STM32WLE5JB	STM32...	Active		UFBGA73	128 kBytes	48 kBytes	43	48 MHz
☆	STM32WLE5JC	STM32...	Active		UFBGA73	256 kBytes	64 kBytes	43	48 MHz
☆	STM32WLE5U8	STM32...	NA		WLCSP59	64 kBytes	20 kBytes	22	48 MHz
☆	STM32WLE5UB	STM32...	NA		WLCSP59	128 kBytes	48 kBytes	22	48 MHz

点击“Start Project”开始创建项目

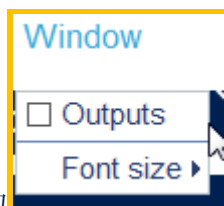
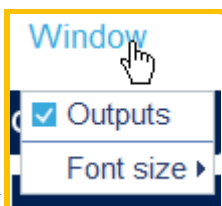


选择 Single Core Project, 点击 OK

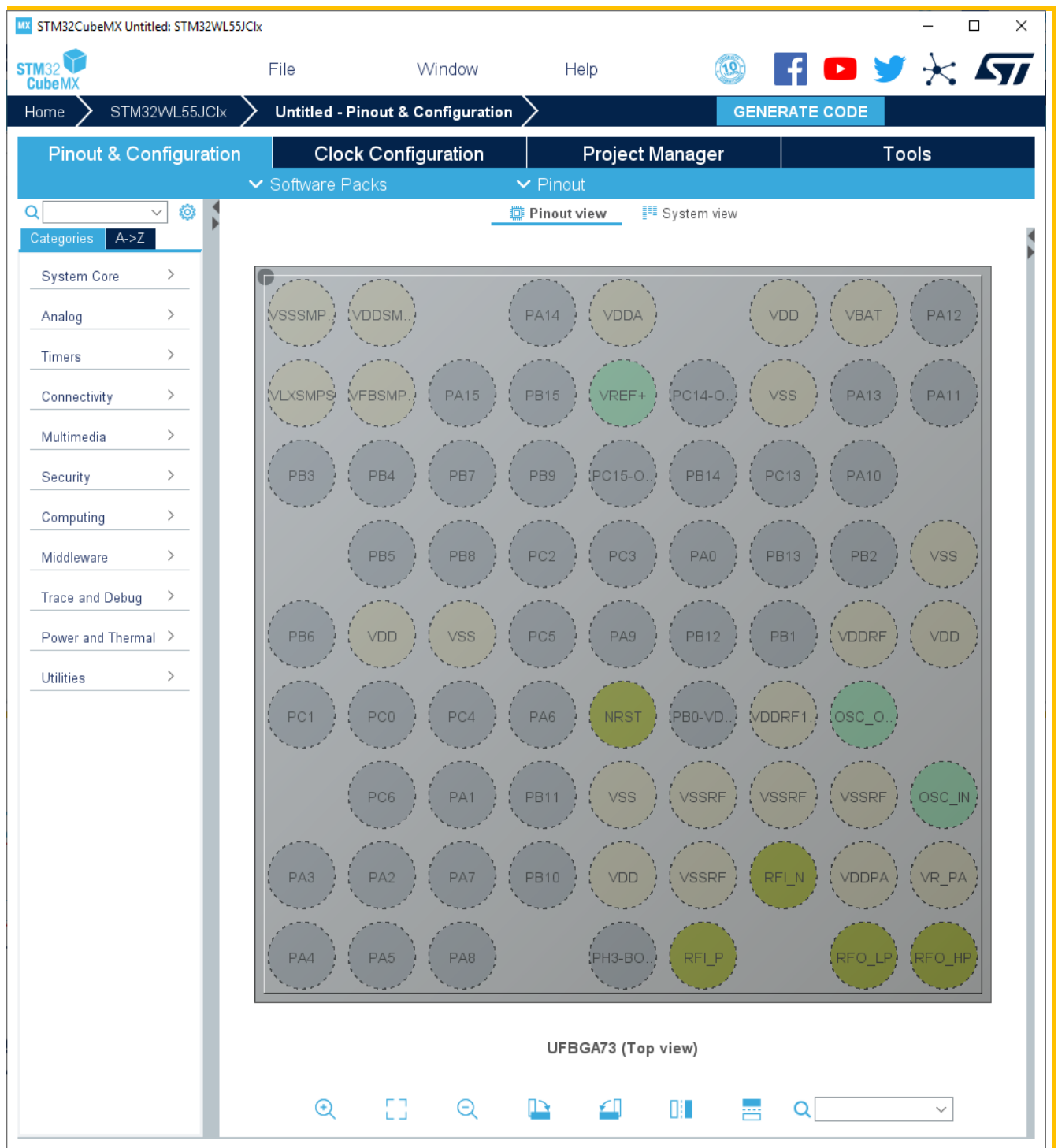


Series	Lines	Mcu	Package	Required Peripherals
STM32WL	STM32WL5x	STM32WL54CCUx	UFQFPN48	None
STM32WL	STM32WL5x	STM32WL54JClx	UFBGA73	None
STM32WL	STM32WL5x	STM32WL55CCUx	UFQFPN48	None
STM32WL	STM32WL5x	STM32WL55JClx	UFBGA73	None
STM32WL	STM32WL5x	STM32WL55UCYx	WLCSP59	None
STM32WL	STM32WLEx	STM32WLE4C8Ux	UFQFPN48	None
STM32WL	STM32WLEx	STM32WLE4CBUx	UFQFPN48	None
STM32WL	STM32WLEx	STM32WLE4CCUx	UFQFPN48	None
STM32WL	STM32WLEx	STM32WLE4J8Ix	UFBGA73	None
STM32WL	STM32WLEx	STM32WLE4JB1x	UFBGA73	None
STM32WL	STM32WLEx	STM32WLE4JClx	UFBGA73	None
STM32WL	STM32WLEx	STM32WLE5C8Ux	UFQFPN48	None
STM32WL	STM32WLEx	STM32WLE5CBUx	UFQFPN48	None
STM32WL	STM32WLEx	STM32WLE5CCUx	UFQFPN48	None
STM32WL	STM32WLEx	STM32WLE5J8Ix	UFBGA73	None
STM32WL	STM32WLEx	STM32WLE5JB1x	UFBGA73	None
STM32WL	STM32WLEx	STM32WLE5JClx	UFBGA73	None
STM32WL	STM32WLEx	STM32WLE5U8Yx	WLCSP59	None
STM32WL	STM32WLEx	STM32WLE5UBYx	WLCSP59	None

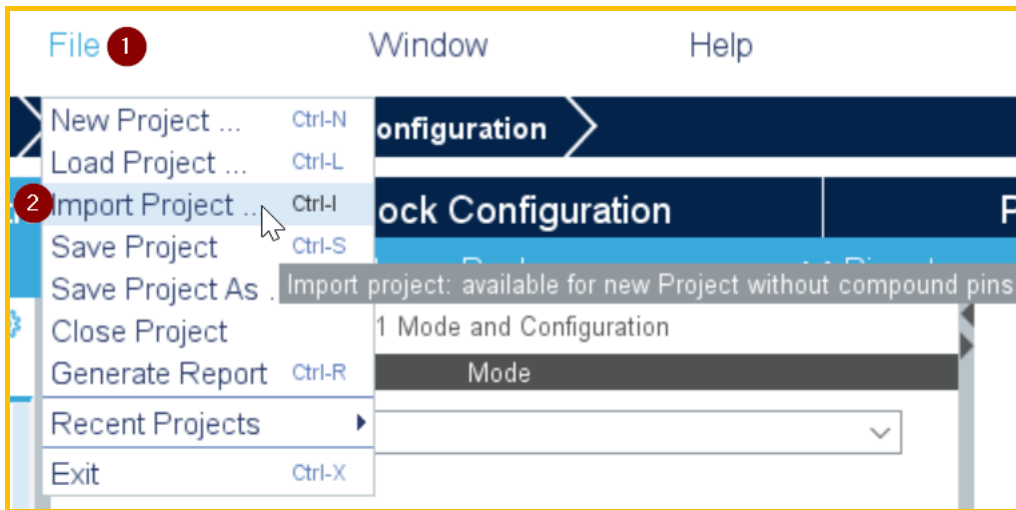
(可选) 为最大化窗口, 可在 Window 中取消 Output 窗口的选择



由 修改为 , 则



1.3 CubeMX 导入参考项目



Import Project

Imported Project
 \STM32Cube_FW_WL_V1.0.0\Projects\NUCLEO-WL55JC\Applications\LoRaWAN\LoRaWAN_End_Node\LoRaWAN_End_Node.ioc

Import MX Settings
 Import Power Consumption Calculator Settings
 Import Project Settings

Import Pinout/Clock Configuration/Configuration Settings
 Automatic Import
 Manual Import
 Import Pinning Status
 Import Peripherals Configuration

Peripheral List

From STM32WL55JCix	To STM32WL55JCix
ADC	<input checked="" type="checkbox"/> ADC
ADV_TRACE	<input checked="" type="checkbox"/> ADV_TRACE
DMA	<input checked="" type="checkbox"/> DMA
LORAWAN	<input checked="" type="checkbox"/> LORAWAN
MISC	<input checked="" type="checkbox"/> MISC
NVIC	<input checked="" type="checkbox"/> NVIC
RCC	<input checked="" type="checkbox"/> RCC
RTC	<input checked="" type="checkbox"/> RTC
SEQUENCER_M4	<input checked="" type="checkbox"/> SEQUENCER_M4
SUBGHZ	<input checked="" type="checkbox"/> SUBGHZ
SYS	<input checked="" type="checkbox"/> SYS
TIMER	<input checked="" type="checkbox"/> TIMER
TINY_LPM	<input checked="" type="checkbox"/> TINY_LPM
USART2	<input checked="" type="checkbox"/> import to [USART2]
GPIO	<input checked="" type="checkbox"/> GPIO

Try Import Show View Pinout

Import Status
 _WL_V1.0.0\Projects\NUCLEO-WL55JC\Applications\LoRaWAN\LoRaWAN_End_Node\LoRaWAN_End_Node.ioc project

OK Cancel

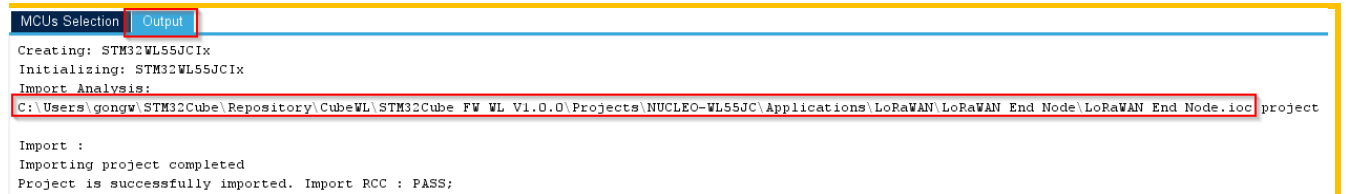
Import Project

i Project is successfully imported. Import RCC : PASS;

Close



可通过 Window=>Output 切换是否查看 import 的提示 log

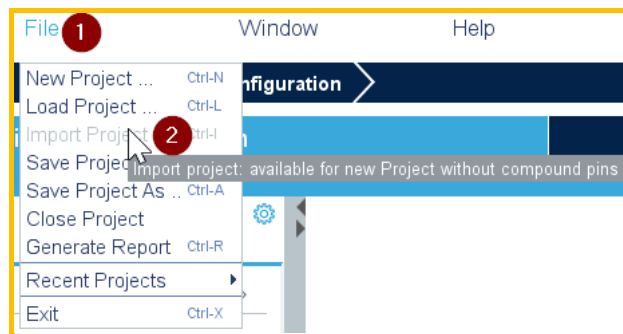


```

Creating: STM32WL55JC1x
Initializing: STM32WL55JC1x
Import Analysis:
C:\Users\gongw\STM32Cube\Repository\CubeWL\STM32Cube_FW_WL_V1.0.0\Projects\NUCLEO-WL55JC\Applications\LoRaWAN\LoRaWAN_End_Node\LoRaWAN_End_Node.ioc project

Import :
Importing project completed
Project is successfully imported. Import RCC : PASS;
    
```

此时再查看 File =>Import Project 选项，变为灰色，表示新项目已被配置，不能再 import 参考项目。



1.4 CubeMX 配置项目

1.4.1 CubeMX 配置 Project Manager

选择 Project Manager

1.4.1.1 配置 Project

选择右侧 Project 并配置如下：

```

Project Name: WLE5Jx_EndNode
Project Location :
C:\Users\gongw\STM32Cube\Repository\CubeWL\STM32Cube_FW_WL_V1.0.0\Projects\NUCLEO-WL55JC\Applications\LoRaWAN
Application Structure : Advanced
    
```

Toolchain / IDE EWARM V8
 Minimum Heap Size: 0x200
 Minimum Stack Size: 0x800
 CubeWL 版本: STM32Cube_FW_WL_V1.0.0

注意:

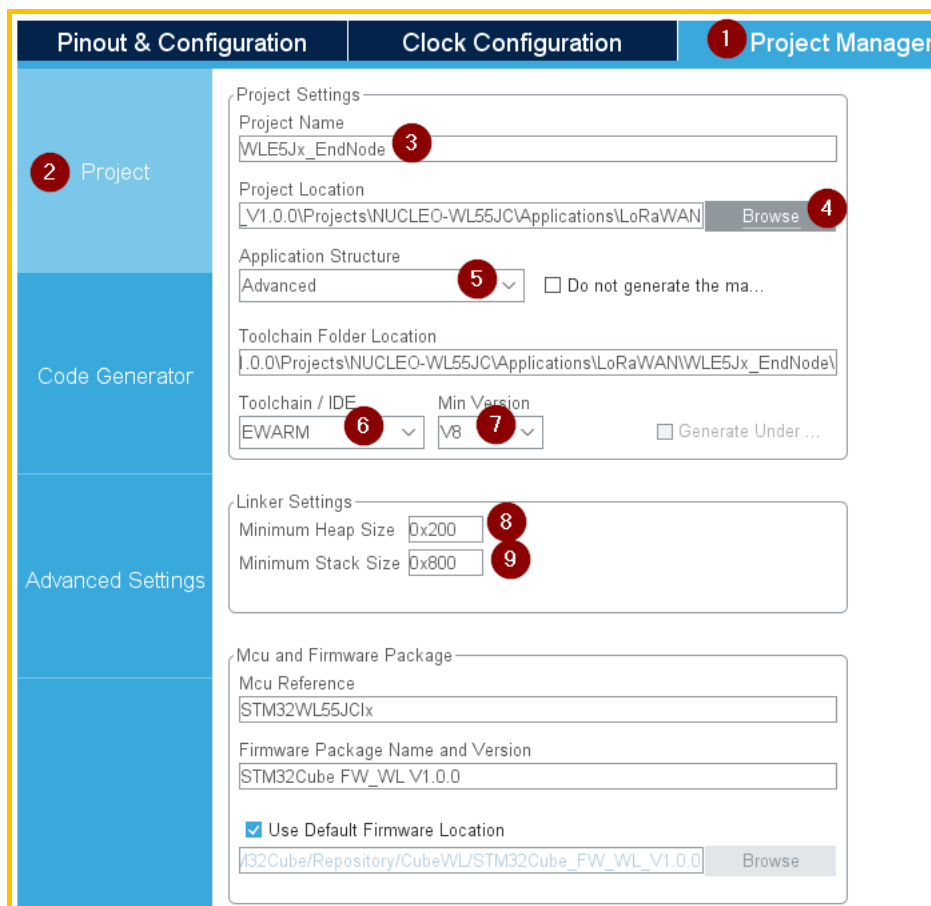
通过 **Browse** 选择 **Project** 保存的目录

- 项目名称 WLE5Jx_EndNode 创建在 STM32Cube_FW_WL_V1.0.0\Projects\NUCLEO-WL55JC\Applications\LoRaWAN 下, 和 LoRaWAN_End_Node 同级目录, 即 C:\Users\gongw\STM32Cube\Repository\CubeWL\STM32Cube_FW_WL_V1.0.0\Projects\NUCLEO-WL55JC\Applications\LoRaWAN\WLE5Jx_EndNode, 以便于与 LoRaWAN_End_Node 进行比较
- 配置 Application Structure 为 Advanced 与 basic 有什么区别?

Application Structure 为 **Advanced** 时目录结构分类分级目录更清晰。

Application Structure 为 **Basic** 时, 所有的*.h 都在 WLE5Jx_EndNode\Inc 目录下, 所有的*.c 在 WLE5Jx_EndNode\Src 目录下。

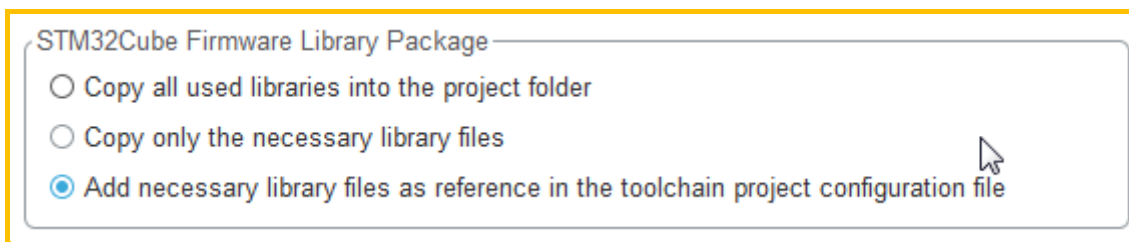
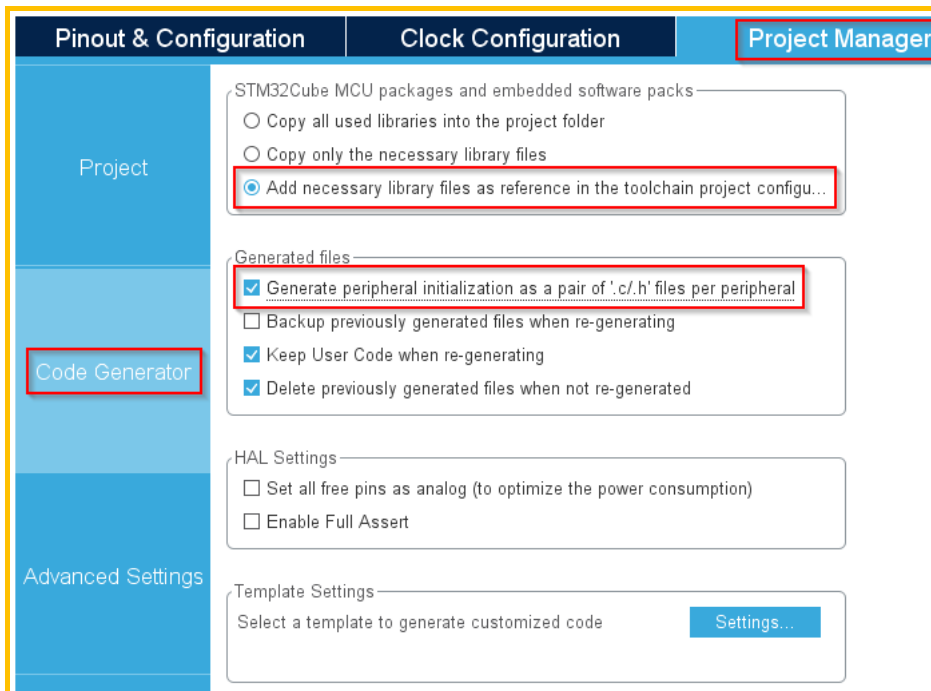
一旦生成过一次代码后目录结构就**无法更改**了!!!



1.4.1.2 配置 Code Generator

选择右侧 **Code Generator** 并配置如下, 选择“Add necessary library files as reference in the toolchain project configuration file”, 这样将不生成 Drivers 和 Middlewares 目录。

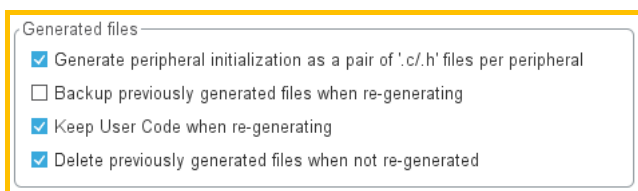
选择 **Generated files** 下的 “Generate peripheral initialization as a pair of ‘.c/.h’ per peripheral”



注意:

- Copy all used library Package (如复制 Drivers\STM32WLxx_HAL_Driver 下的所有驱动) WLE5Jx_EndNode 复制到其他目录下, 还能工作
- Copy only the necessary library files(如仅复制 Drivers\STM32WLxx_HAL_Driver 下使用到的驱动) WLE5Jx_EndNode 复制到其他目录下, 还能工作
- **Add necessary library files as reference in the toolchain project configuration file**(不生成 Drivers 和 Middlewares 目录, 仅参考 Drivers\STM32WLxx_HAL_Driver 下的所有驱动) WLE5Jx_EndNode 复制到其他目录下, 不能工作

参考 UM1718 4.9 Project Manager view

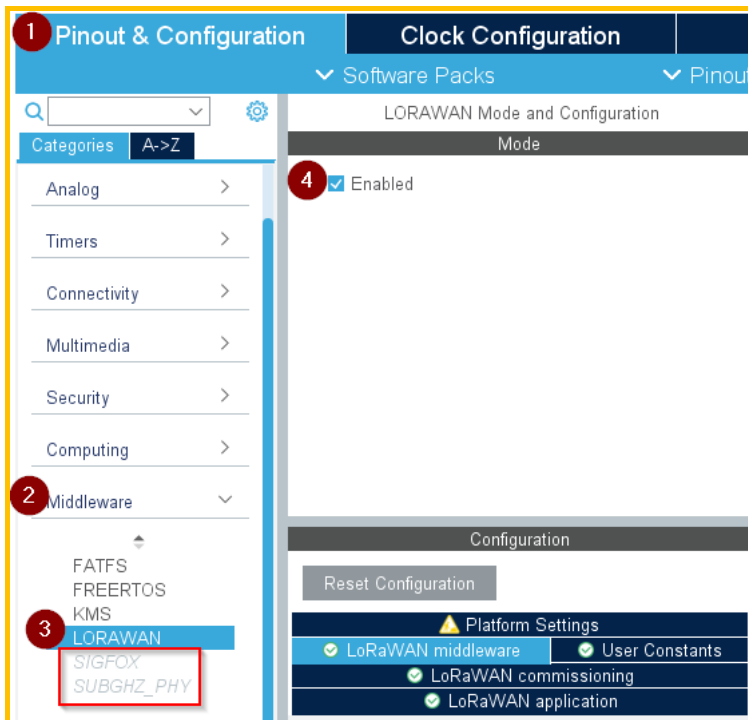


2 配置外设

2.1 使能外设

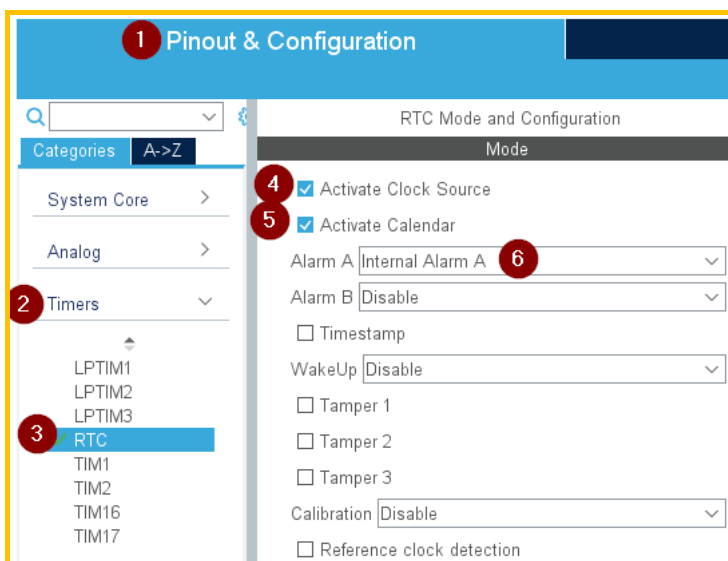
2.1.1 Middleware=>LORAWAN

Middleware=>LORAWAN=>Mode 中勾选 Enabled, LORAWAN 功能就使能了, 同时 SIGFOX 和 SUBGHZ_PHY 就失能了, 默认界面如下, 后续再继续配置 LORAWAN 应用。



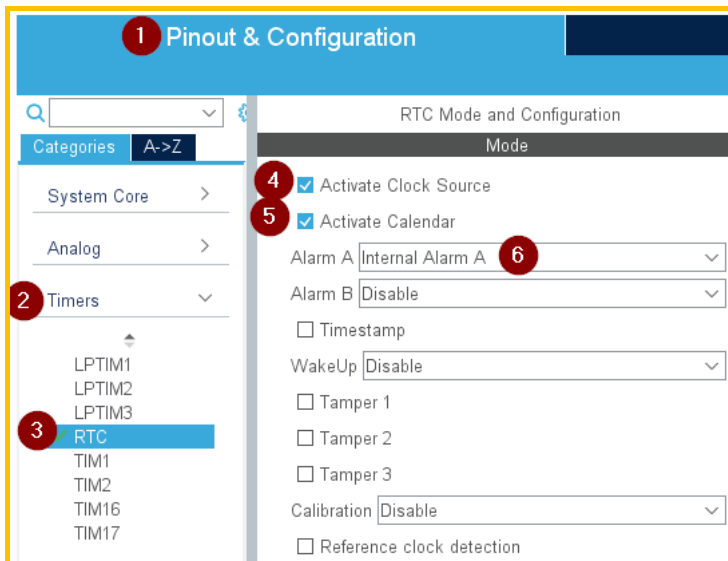
2.1.2 Timers=>RTC 使能

Timers=>RTC=>勾选 Activate Clock Source, 并配置 Alarm A 为 Internal Alarm A



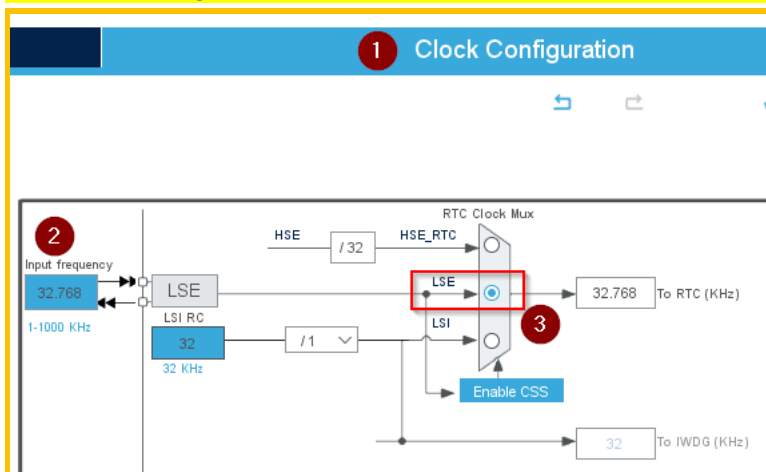
2.2 RTC

配置 Alarm A 为 Internal Alarm A



Clock Configuration => RTC Clock Mux=>LSE

在 Clock Configuration 中检查 RTC Clock Mux 是否配置为 LSE



Timer =>RTC=>Users Constants

1 Pinout & Configuration
⌵

Categories A->Z

System Core >

Analog >

2 Timers >

LPTIM1

LPTIM2

LPTIM3

3 RTC

TIM1

TIM2

TIM16

TIM17

Connectivity >

Multimedia >

Security >

Computing >

Middleware >

Trace and Debug >

Power and Thermal >

RTC Mode and Configuration

Mode

4 Activate Clock Source

5 Activate Calendar

6 Alarm A

Alarm B

Timestamp

WakeUp

Tamper 1

Tamper 2

Tamper 3

Calibration

Reference clock detection

Configuration

7

Parameter Settings

User Constants

NVIC Settings

Search Constants

8

Constant Name	Constant Value
RTC_N_PREDIV_S	10
RTC_PREDIV_S	((1<<RTC_N_PREDIV_S)-1)
RTC_PREDIV_A	((1<<(15-RTC_N_PREDIV_S))-1)
USART_BAUDRATE	115200

9

RTC_N_PREDIV_S	10
RTC_PREDIV_S	$((1 \ll \text{RTC_N_PREDIV_S}) - 1)$
RTC_PREDIV_A	$((1 \ll ((15 - \text{RTC_N_PREDIV_S})) - 1)) - 1)$
USART_BAUDRATE	115200
LPUART_BAUDRATE	9600

Timer =>RTC=>Parameter Settings

1 Pinout & Configuration

Categories: A->Z

- System Core >
- Analog >
- 2 Timers >**
 - LPTIM1
 - LPTIM2
 - LPTIM3
 - 3 RTC**
 - TIM1
 - TIM2
 - TIM16
 - TIM17
- Connectivity >
- Multimedia >
- Security >
- Computing >
- Middleware >
- Trace and Debug >
- Power and Thermal >
- Utilities >

RTC Mode and Configuration

Mode

- Activate Clock Source
- Activate Calendar
- Alarm A: Internal Alarm A
- Alarm B: Disable
- Timestamp
- WakeUp: Disable
- Tamper 1
- Tamper 2
- Tamper 3
- Calibration: Disable
- Reference clock detection

Configuration

Reset Configuration

4 Parameter Settings User Constants NVIC Settings

Configure the below parameters :

Search (Ctrl+F)

- General
 - Asynchronous Predivider value: RTC_PREDIV_A **6**
 - Bin Mode: Free running Binary mode **5**
 - SSRU Underflow Interrupt: Enabled
- Alarm A
 - Free running 32 bit value: 0
 - Binary AutoControl: RTC_ALARMSUBSECONDBIN_AUTOCLR_NO **7**
 - Free running 32 bit mask: SS[31:0] are compared and must match to activate alarm.

Timer =>RTC=>NVIC Settings

1 Pinout & Configuration

RTC Mode and Configuration

Mode

- Activate Clock Source
- Activate Calendar
- Alarm A: Internal Alarm A
- Alarm B: Disable
- Timestamp
- WakeUp: Disable
- Tamper 1
- Tamper 2
- Tamper 3
- Calibration: Disable
- Reference clock detection

Configuration

Reset Configuration **4**

Parameter Settings User Constants NVIC Settings

NVIC Interrupt Table		Enabled	Preemption Priority	Sub Priority
RTC Tamper, RTC TimeStamp, LSECSS and RTC SSRU Interru.	<input checked="" type="checkbox"/>	5	0	0
RTC Alarms (A and B) Interrupt	<input checked="" type="checkbox"/>	6	0	0

Middleware => LORAWAN => Platform Settings => Timer Server => RTC

1 Pinout & Configuration

Clock Configuration Project Manager

Software Packs Pinout

LORAWAN Mode and Configuration

Mode

4 Enabled

Configuration

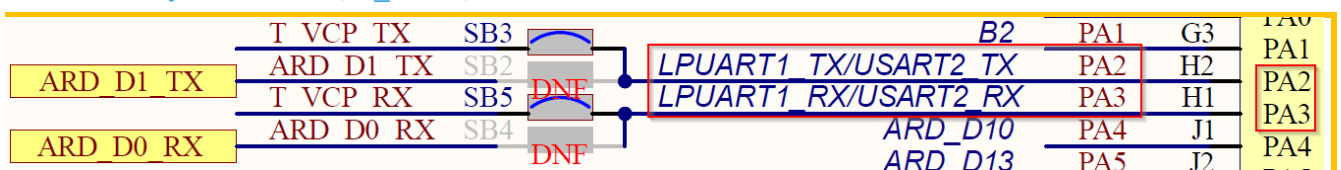
Reset Configuration

LoRaWAN application LoRaWAN commissioning LoRaWAN middleware User Constants Platform Settings **5**

Name	IPs or Components	Found Solutions	BSP API
RTC	RTC:RTC Enabled	RTC 6	Unknown

2.3 DebugLine

Connectivity=> USART2 (AT_Slave)



1 Pinout & Configuration

USART2 Mode and Configuration

Mode: Asynchronous **4**

Hardware Flow Control (RS232): Disable

Hardware Flow Control (RS485)

Slave Select(NSS) Management: Disable

2 Connectivity

- I2C1
- I2C2
- I2C3
- IRTIM
- LPUART1
- SPI1
- SPI2
- SUBGHZ
- USART1
- 3 USART2**

1 Pinout & Configuration | **Clock Configuration** | **Project Manager**

Software Packs | Pinout

USART2 Mode and Configuration

Mode: Asynchronous

Configuration

Reset Configuration **4**

Parameter Settings | **User Constants** | NVIC Settings | DMA Settings | GPIO Settings

Search Constants

Search (Ctrl+F)

Constant Name	Value
RTC_N_PREDIV_S	10
RTC_PREDIV_S	((1<<RTC_N_PREDIV_S)-1)
RTC_PREDIV_A	((1<<(15-RTC_N_PREDIV_S))-1)
USART_BAUDRATE 5	115200

2 Connectivity

- I2C1
- I2C2
- I2C3
- IRTIM
- LPUART1
- SPI1
- SPI2
- ✓ SUBGHZ
- USART1
- 3 USART2**

1 Pinout & Configuration **Clock Configuration** **Project**

▼ Software Packs ▼ Pinout

USART2 Mode and Configuration

Mode

Mode: Asynchronous **4**

Configuration

Reset Configuration

5 Parameter Settings User Constants NVIC Settings DMA Settings GPIO Settings

Configure the below parameters :

Search (Ctrl+F)

- Basic Parameters
 - Baud Rate **6** USART_BAUDRATE Bits/s
 - Word Length 8 Bits (including Parity)
 - Parity None
 - Stop Bits 1
- Advanced Parameters
 - Data Direction Receive and Transmit
 - Over Sampling 16 Samples
 - Single Sample Disable
 - ClockPrescaler 1
 - Fifo Mode **7** Enable
 - Txfifo Threshold 1 eighth full configuration
 - Rxfifo Threshold 1 eighth full configuration
- Advanced Features
 - Auto Baudrate Disable
 - TX Pin Active Level Inversion Disable
 - RX Pin Active Level Inversion Disable
 - Data Inversion Disable
 - TX and RX Pins Swapping Disable
 - Overrun Enable
 - DMA on RX Error Enable
 - MSB First Disable

2 Connectivity

- I2C1
- I2C2
- I2C3
- IRTIM
- LPUART1
- SPI1
- SPI2
- SUBGHZ
- USART1
- 3** USART2

Multimedia

Security

Computing

Middleware

Trace and Debug

Power and Thermal

Pinout & Configuration

USART2 Mode and Configuration

Mode

Mode: Asynchronous

Hardware Flow Control (RS232) Disable

Hardware Flow Control (RS485)

Slave Select(NSS) Management: Disable

Configuration

Reset Configuration

NVIC Settings DMA Settings GPIO Settings

Parameter Settings User Constants

DMA Request	Channel	Direction	Priority
USART2_TX	DMA1 Channel 5 2	Memory To Peripheral	Low

1

1 Pinout & Configuration Clock Configuration Project Manager Tools

Software Packs Pinout

USART2 Mode and Configuration

Mode: Asynchronous **4**

Configuration

Reset Configuration **5**

Parameter Settings User Constants NVIC Settings DMA Settings GPIO Settings

Search Signals
Search (Ctrl+F) Show only Modified Pins

Pin No...	Signal on Pin	GPIO output level	GPIO mode	GPIO Pull-up/Pull-down	Maximum output speed	Fast Mode	User Label	Modified
PA2	USART2_TX	n/a	Alternate Function Push Pull	No pull-up and no pull-down	Very High	n/a	USARTx_TX 7	<input checked="" type="checkbox"/>
PA3	USART2_RX	n/a	Alternate Function Push Pull	No pull-up and no pull-down	Very High	n/a	USARTx_RX 9	<input checked="" type="checkbox"/>

2 Connectivity

- I2C1
- I2C2
- I2C3
- IRTIM
- LPUART1
- SPI1
- SPI2
- ✓ SUBGHZ
- USART1
- 3** USART2

Pinout & Configuration

USART2 Mode and Configuration

Mode

Mode: Asynchronous

Hardware Flow Control (RS232): Disable

Hardware Flow Control (RS485)

Slave Select(NSS) Management: Disable

Configuration

Reset Configuration

1 NVIC Settings DMA Settings GPIO Settings

Parameter Settings User Constants

NVIC Interrupt Table	Enabled	Preemption Priority	Sub Priority
DMA1 Channel 5 Interrupt	<input checked="" type="checkbox"/>	0	0
USART2 Interrupt	<input checked="" type="checkbox"/> 2	0	0

1 Pinout & Configuration
Clock Configuration

Categories A->Z

2 System Core

- DMA
- GPIO
- HSEM
- IWDG
- 3 NVIC
- RCC
- SYS
- WWDG

Analog >

Timers >

Connectivity >

- I2C1
- I2C2
- I2C3
- IRTIM
- LPUART1
- SPI1
- SPI2
- SUBGHZ
- USART1
- USART2

NVIC Mode and Configuration

Configuration

Priority Group 4 bits for pre-emption priority 0 bits for subpriority Sort by Preemption Priority and Sub Priority

Search Show only enabled interrupts Force DMA channels Interrupts

NVIC Interrupt Table	Enabled	Preemption Priority	Sub Priority
Non maskable interrupt	☑	0	0
Hard fault interrupt	☑	0	0
Memory management fault	☑	0	0
Prefetch fault, memory access fault	☑	0	0
Undefined instruction or illegal state	☑	0	0
System service call via SWI instruction	☑	0	0
Debug monitor	☑	0	0
Pendable request for system service	☑	0	0
Time base: System tick timer	☑	0	0
PVD and PVM detector	☐	0	0
FLASH (CFI) global Interrupt	☐	0	0
RCC Interrupt	☐	0	0
DMA1 Channel 5 Interrupt	☑	2	0
ADC Interrupt	☐	0	0
USART2 Interrupt	☑	2	0
RTC Alarms (A and B) Interrupt	☐	0	0
SUBGHZ Radio Interrupt	☐	0	0

Clock Configuration => USART2 Clock Mux => SYSCLK

Middleware => LORAWAN => LoRaWAN middleware => radio_board_if => Activate Debug Line

LAT1054 - Rev 1.0

page 24/46

Platform Settings => VCOM => USART2

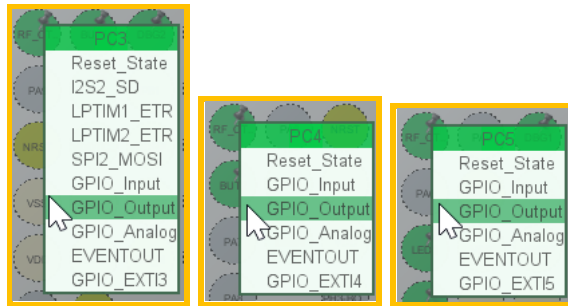
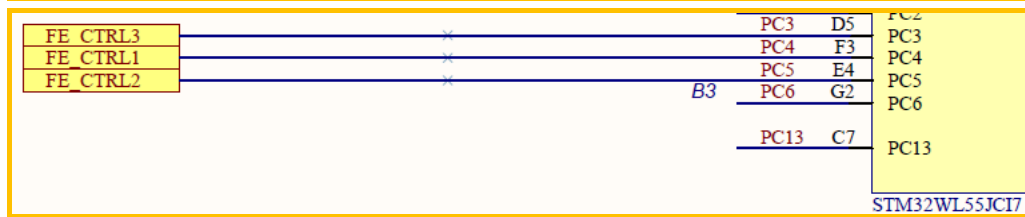
Name	IPs or Components	Found Solutions	BSP API
USART	USART:Asynchronous	USART2	Unknown

2.4 System Core => GPIO (RF SW CTRL / BUTTON / LED)

2.4.1 RF_CTRL

NUCLEO-WL55JC

RF front-end configuration	FE_CTRL1	FE_CTRL2	FE_CTRL3
Transmit high output power	Low	High	High
Transmit low output power	High	High	High
Receive	High	Low	High



1 Pinout & Configuration

2 System Core

3 DMA, GPIO, HSEM, IWDG, NVIC, RCC, SYS, WWDG

4 GPIO, RCC, USART

5 Pin Name

6 User Label

7 Signal on Pin

8 Modified

9 GPIO output level

10

Pin Name	Signal on Pin	GPIO output level	GPIO mode	GPIO Pull-up/Pull-down	Maximum output speed	Fast Mode	User Label	Modified
PC3	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	RF_CTRL3	<input checked="" type="checkbox"/>
PC4	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	RF_CTRL1	<input checked="" type="checkbox"/>
PC5	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	RF_CTRL2	<input checked="" type="checkbox"/>

1 Pinout & Configuration **Clock Configuration**

Software Packs Pinout

Search: LORAWAN Mode and Configuration

Mode: **4** Enabled

Configuration

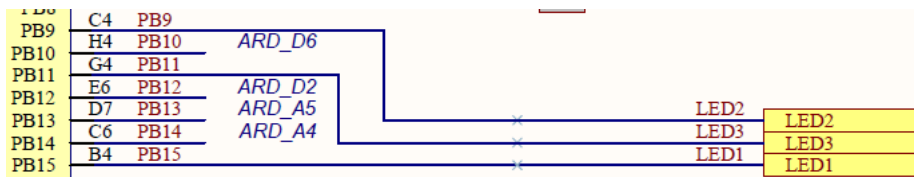
Reset Configuration **5**

LoRaWAN application LoRaWAN commissioning LoRaWAN middleware User Constants Platform Settings

Name	IPs or Components	Found Solutions	BSP API
RF SW CTRL 3	GPIO:Output 6	Undefined	Unknown
RF SW CTRL 1	GPIO:Output 6	PA4 [RF_CTRL1] 7	Unknown
RF SW CTRL 2	GPIO:Output 8	PA5 [RF_CTRL2] 9	Unknown

2.4.2 LED

配置控制 LED 的 GPIO



GPIO Configuration Menu 1:

- Reset_State
- I2C2_SCL
- I2S2_SD
- SPI2_MOSI
- TIM1_CH3N
- GPIO_Input
- GPIO_Output
- GPIO_Analog
- EVENTOUT
- GPIO_EXTI5

GPIO Configuration Menu 2:

- Reset_State
- DAC_EXTI9
- I2C1_SDA
- I2S2_WS
- IR_OUT
- SPI2_NSS
- TIM17_CH1
- TIM1_CH3N
- GPIO_Input
- GPIO_Output
- GPIO_Analog
- EVENTOUT
- GPIO_EXTI9

GPIO Configuration Menu 3:

- Reset_State
- ADC_EXTI11
- COMP2_OUT
- I2C3_SDA
- LPUART1_TX
- TIM2_CH4
- TIM1_CH3N
- GPIO_Input
- GPIO_Output
- GPIO_Analog
- EVENTOUT
- GPIO_EXTI11

1 Pinout & Configuration **Clock Configuration** **Project Manager**

Software Packs Pinout

Search: GPIO Mode and Configuration

Configuration

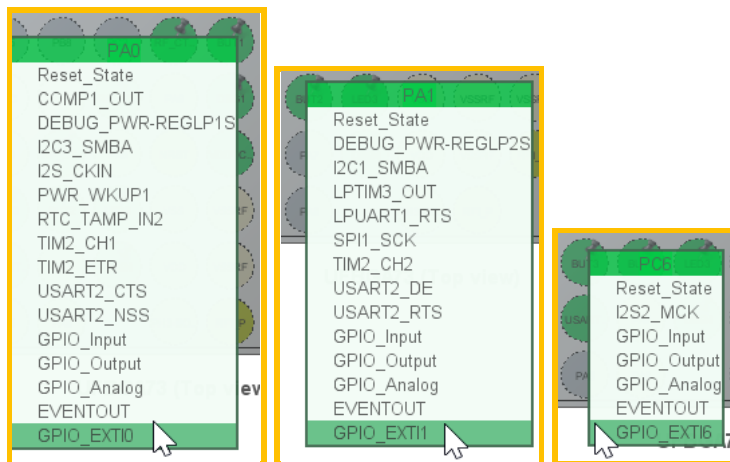
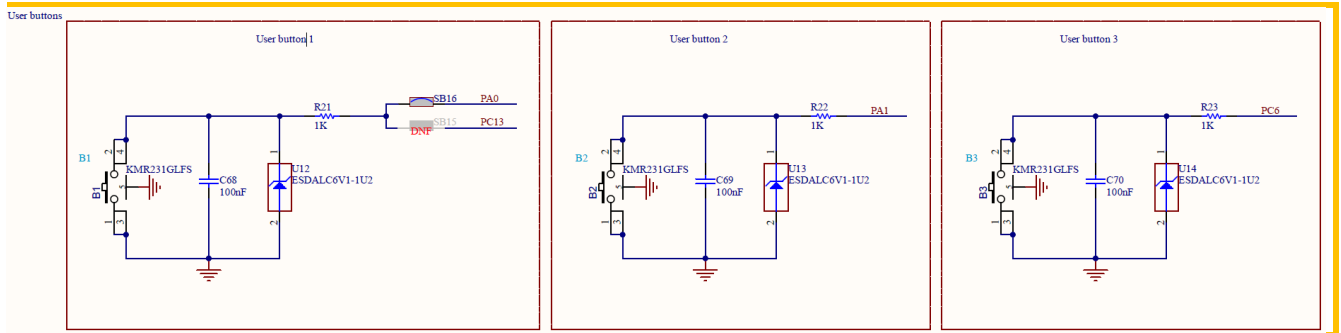
Group By Peripherals: **4** GPIO RCC USART

Search Signals: [Search (Ctrl+F)] Show only Modified Pins

Pin No	Signal on Pin	GPIO output level	GPIO mode	GPIO Pull-up/Pull-down	Maximum output speed	Fast Mode	User label	Modified
PB15 5	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	LED1 6	<input checked="" type="checkbox"/>
PB9 7	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	Disable	LED2 8	<input checked="" type="checkbox"/>
PB11 9	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	LED3 10	<input checked="" type="checkbox"/>
PC4	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	RF_CTRL1	<input checked="" type="checkbox"/>
PC5	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	RF_CTRL2	<input checked="" type="checkbox"/>
PC3	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	RF_CTRL3	<input checked="" type="checkbox"/>

2.4.3 BUT

配置控制 BUTTON 的 GPIO



1 Pinout & Configuration

Clock Configuration

Project Manager

Software Packs

Pinout

GPIO Mode and Configuration

Configuration

Group By Peripherals

4 GPIO

Search Signals

Search (Ctrl+F)

Show only Modified Pins

Pin No	Signal on Pin	GPIO output level	GPIO mode	GPIO Pull-up/Pull-down	Maximum output speed	Fast Mode	Usart	Modified
PA0	n/a	n/a	External Interrupt Mode with Rising edge trigger detection	no pull-up and no pull-down	n/a	n/a	BUT1	<input checked="" type="checkbox"/>
PA1	n/a	n/a	External Interrupt Mode with Rising edge trigger detection	no pull-up and no pull-down	n/a	n/a	BUT2	<input checked="" type="checkbox"/>
PC6	n/a	n/a	External Interrupt Mode with Rising edge trigger detection	no pull-up and no pull-down	n/a	n/a	BUT3	<input checked="" type="checkbox"/>
PB15	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	LED1	<input checked="" type="checkbox"/>
PB9	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	Disable	LED2	<input checked="" type="checkbox"/>
PB11	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	LED3	<input checked="" type="checkbox"/>
PC4	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	RF_CTRL1	<input checked="" type="checkbox"/>
PC5	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	RF_CTRL2	<input checked="" type="checkbox"/>
PC3	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	RF_CTRL3	<input checked="" type="checkbox"/>

1 Pinout & Configuration

Clock Configuration

GPIO Mode and Configuration

Configuration

Group By Peripherals

GPIO

NVIC

NVIC Interrupt Table	Enabled	Preemption Priority	Sub Priority
EXTI Line 0 Interrupt	<input checked="" type="checkbox"/>	0	0
EXTI Line 1 Interrupt	<input checked="" type="checkbox"/>	0	0
EXTI Lines [9:5] Interrupt	<input type="checkbox"/>	0	0

2. 4. 4 DBG

Pin No.	Signal on Pin	GPIO output level	GPIO mode	GPIO Pull-up/Pull-down	Maximum output s...	Fast Mode	User Label	Modified
PB10	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	DBG4	<input checked="" type="checkbox"/>
PB12	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	DBG1	<input checked="" type="checkbox"/>
PB13	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	DBG2	<input checked="" type="checkbox"/>
PB14	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	DBG3	<input checked="" type="checkbox"/>
PA0	n/a	n/a	External Interrupt Mode with Rising edge trigger detection	No pull-up and no pull-down	n/a	n/a	BUT1	<input checked="" type="checkbox"/>
PA1	n/a	n/a	External Interrupt Mode with Rising edge trigger detection	No pull-up and no pull-down	n/a	n/a	BUT2	<input checked="" type="checkbox"/>
PC6	n/a	n/a	External Interrupt Mode with Rising edge trigger detection	No pull-up and no pull-down	n/a	n/a	BUT3	<input checked="" type="checkbox"/>
PB15	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	LED1	<input checked="" type="checkbox"/>
PB9	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	Disable	LED2	<input checked="" type="checkbox"/>
PB11	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	LED3	<input checked="" type="checkbox"/>
PC4	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	RF_CTRL1	<input checked="" type="checkbox"/>
PC5	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	RF_CTRL2	<input checked="" type="checkbox"/>
PC3	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	n/a	RF_CTRL3	<input checked="" type="checkbox"/>

2. 4. 5 SWD (可选)

需使能 Trace and Debug => DEBUG => JTAG and Trace => Serial Wire

T SWDIO	*	T SWDIO	PA13	B8	PA12
T SWCLK	*	T SWCLK	PA14	A4	PA13
T JTDI	*	T JTDI	PA15	B3	PA14
					PA15

BGA73	QFN48
-------	-------



SWD : PA13-DEBUG_JTMS-SWDIO, DEBUG_PA14-JTCK-SWCLK,
 JTAG (4 pins) + : PA15-DEBUG_JTDI, PB3-DEBUG_JTDO-SWO
 JTAG (5 pins) + : PB4-DEBUG_NJTRST

2. 4. 6 Middleware=>LORAWAN=>Platform Settings

Board resources => LED & BUTTON

Radio => RF SW CTRL

Debug = > Debug Line

1 Pinout & Configuration Clock Configuration v Software

LORAWAN Mode and Configuration

Categories A->Z

System Core **4** Enabled

Analog

Timers

Connectivity

Multimedia

Security

Computing

2 Middleware

- FATFS
- FREERTOS
- KMS
- 3** LORAWAN
- SIGFOX
- SUBGHZ_PHY

Trace and Debug

Power and Thermal

Utilities

Mode

Configuration

Reset Configuration **5**

LoRaWAN application LoRaWAN commissioning LoRaWAN middleware User Constants Platform Settings

Board resources

Name	IPs or Components	Found Solutions	BSP API
LED 3	GPIO:Output	PB11 [LED3]	Unknown
LED 2	GPIO:Output	PB9 [LED2]	Unknown
LED 1	GPIO:Output	PB15 [LED1]	Unknown
BUTTON 1	GPIO:EXTI	PA0 [BUT1]	Unknown
BUTTON 3	GPIO:EXTI	PC6 [BUT3]	Unknown
BUTTON 2	GPIO:EXTI	PA1 [BUT2]	Unknown

TimeServer

Name	IPs or Components	Found Solutions	BSP API
RTC	RTC:RTC Enabled	RTC	Unknown

Radio

Name	IPs or Components	Found Solutions	BSP API
RF SW CTRL 3	GPIO:Output	PC3 [RF_CTRL3]	Unknown
RF SW CTRL 1	GPIO:Output	PC4 [RF_CTRL1]	Unknown
RF SW CTRL 2	GPIO:Output	PC5 [RF_CTRL2]	Unknown

Debug

Name	IPs or Components	Found Solutions	BSP API
Debug Line 1	GPIO:Output	PB12 [DBG1]	Unknown
Debug Line 2	GPIO:Output	PB13 [DBG2]	Unknown
Debug Line 3	GPIO:Output	PB14 [DBG3]	Unknown
Debug Line 4	GPIO:Output	PB10 [DBG4]	Unknown

6

7

8

9

2.5 ADC

Analog=>ADC

The screenshot displays the configuration interface for the ADC. The left sidebar shows a tree view with categories: System Core, Analog, ADC (selected), COMP1, COMP2, DAC, VREFBUF, Timers, Connectivity, Multimedia, Security, Computing, and Middleware. The main panel is titled 'ADC Mode and Configuration' and contains a list of modes with checkboxes. The 'Temperature Sensor Channel' and 'Vrefint Channel' are checked, while others are unchecked. Red circles with numbers 1 through 5 highlight specific elements: 1 points to the 'Pinout & Configuration' tab, 2 to the 'Analog' category, 3 to the 'ADC' sub-category, 4 to the checked 'Temperature Sensor Channel', and 5 to the checked 'Vrefint Channel'.

Mode	Selected
<input type="checkbox"/> IN0	
<input type="checkbox"/> IN1	
<input type="checkbox"/> IN2	
<input type="checkbox"/> IN3	
<input type="checkbox"/> IN4	
<input type="checkbox"/> IN5	
<input type="checkbox"/> IN6	
<input type="checkbox"/> IN7	
<input type="checkbox"/> IN8	
<input type="checkbox"/> IN9	
<input type="checkbox"/> IN10	
<input type="checkbox"/> IN11	
<input checked="" type="checkbox"/> Temperature Sensor Channel	Yes
<input checked="" type="checkbox"/> Vrefint Channel	Yes
<input type="checkbox"/> Vbat Channel	
<input type="checkbox"/> VDAC_OUT1 Channel	
<input type="checkbox"/> Regular Conversion Trigger	

Middleware=>LORAWAN=>Platform Settings => Vrefint T calibration => ADC (EndNode)

Name	IPs or Components	Found Solutions	BSP API
ADC	ADC:Vrefint Channel	ADC	Unknown

3 STM32WL Templates Patches for CubeMX

STM32WL Templates 在如下目录，如需要可自行定制自己的 Templates

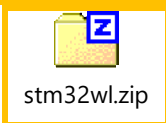
C:\Program Files\STMicroelectronics\STM32Cube\STM32CubeMX\db\templates\stm32wl

针对不同的客户定制化的板子，硬件外设配置不一致，客户可根据需要定制化自己的外设配置。从 NUCLEO-WL55JC(BGA73) 移植到 STM32WLExCxUx(QFN48)定制化的板子上，射频开关 (RF SW CTRL)，按键(BUTTON), LED 的配置会不一样，可能导致编译错误，如想生成的代码不会产生编译错误，需要编辑一下 STM32WL 的 template

3.1 CubeMX template patches for stm32wl

请参考

C:\Program Files\STMicroelectronics\STM32Cube\STM32CubeMX\db\templates\stm32wl



其中包含

radio_board_if_c.ftl for RF SW CTRL

radio_conf_h.ftl for DBG_GPIO_RADIO_TX

board_resources_c.ftl & board_resources_h.ftl for BUTTON

SubGHz_Phy_subghz_phy_app_c.ftl for LED

LoRaWAN_lora_app_c.ftl for LED

3.2 CubeMX Updates from v6.1.1 to v6.2.0

main.c

<pre>void SystemClock_Config(void) { RCC_OscInitTypeDef RCC_OscInitStruct = {0}; RCC_ClkInitTypeDef RCC_ClkInitStruct = {0}; /** Configure LSE Drive Capability */ __HAL_RCC_LSEDRIVE_CONFIG(RCC_LSEDRIVE_ LOW); /** Configure the main internal regulator output voltage */ </pre>	<pre>void SystemClock_Config(void) { RCC_OscInitTypeDef RCC_OscInitStruct = {0}; RCC_ClkInitTypeDef RCC_ClkInitStruct = {0}; /** Configure LSE Drive Capability */ HAL_PWR_EnableBkUpAccess(); __HAL_RCC_LSEDRIVE_CONFIG(RCC_LSEDRIVE_ LOW); </pre>
<pre>/** * @brief Macro to configure the External Low Speed oscillator (LSE) drive capability. * @note As the LSE is in the Backup domain and write access is denied to * this domain after reset, you have to enable write access using * HAL_PWR_EnableBkUpAccess() function before to configure the LSE * (to be done once after reset). * @param __LSEDRIVE__ specifies the new state of the LSE drive capability. * This parameter can be one of the following values: </pre>	

```

* @arg @ref RCC_LSEDRIVE_LOW LSE oscillator low drive capability.
* @arg @ref RCC_LSEDRIVE_MEDIUMLOW LSE oscillator medium low drive capability.
* @arg @ref RCC_LSEDRIVE_MEDIUMHIGH LSE oscillator medium high drive capability.
* @arg @ref RCC_LSEDRIVE_HIGH LSE oscillator high drive capability.
* @retval None
*/
#define __HAL_RCC_LSEDRIVE_CONFIG(__LSEDRIVE__)
LL_RCC_LSE_SetDriveCapability(__LSEDRIVE__)

```

rtc.c

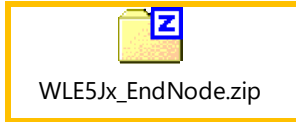
CubeMX v6.1.1	CubeMX v6.2.0
if (HAL_RTC_SetAlarm_IT(&hrtc, &sAlarm, RTC_FORMAT_BCD) != HAL_OK)	if (HAL_RTC_SetAlarm_IT(&hrtc, &sAlarm, 0) != HAL_OK)
<pre> #define RTC_FORMAT_BIN 0x00000000u /* This parameter will trigger a SW conversion to fit with the native BCD format of the HW Calendar. It should not be confused with the Binary mode @ref RTCEx_Binary_Mode. */ #define RTC_FORMAT_BCD 0x00000001u /* Native format of the HW Calendar. It should not be confused with the Binary mode @ref RTCEx_Binary_Mode. */ </pre>	
<p>Ticket 96721 - [Diff Examples] WL RTC: RTC_FORMAT_BCD removed from HAL_RTC_SetAlarm_IT call</p> <p>Ticket 100003 - [Diff Examples] WL RTC: RTC_FORMAT_BCD removed from HAL_RTC_SetAlarm_IT call</p> <p>Ticket 90404 - [HAL_RTC_SetAlarm & HAL_RTC_SetAlarm_IT] Unnecessary assertion on parameter when Alarm Masks defined</p>	

usart.c

CubeMX v6.1.1	CubeMX v6.2.0
<pre> if (HAL_DMA_ConfigChannelAttributes(&hdma_lpuart1_tx, DMA_CHANNEL_NPRIV) != HAL_OK) { Error_Handler(); } </pre>	删除了

4 验证

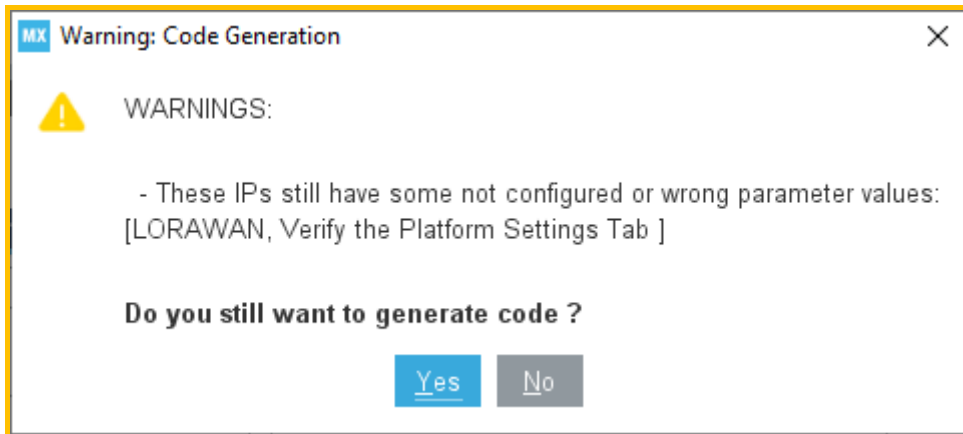
4.0 参考工程



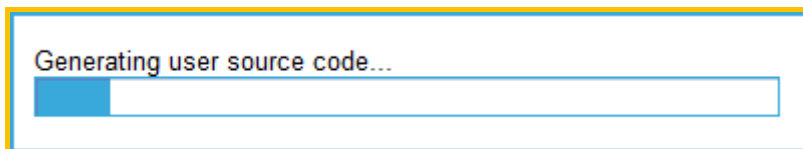
其中包含 WLE5Jx_EndNode.ioc

4.1 生成代码 GENERATE CODE

此时，可以先选择 GENERATE CODE 生成 Project, (如需，请自己选择项目保存的目录)



选择 Yes



可选择 Open Project 使用 IDE 直接打开 Project。



或选择 Open Folder 后，还需再选择 EWARM\Project.eww 打开 Project

STM32Cube_FW_WL_V1.0.0 > Projects > NUCLEO-WL55JC > Applications > LoRaWAN > WLE5Jx_EndNode >

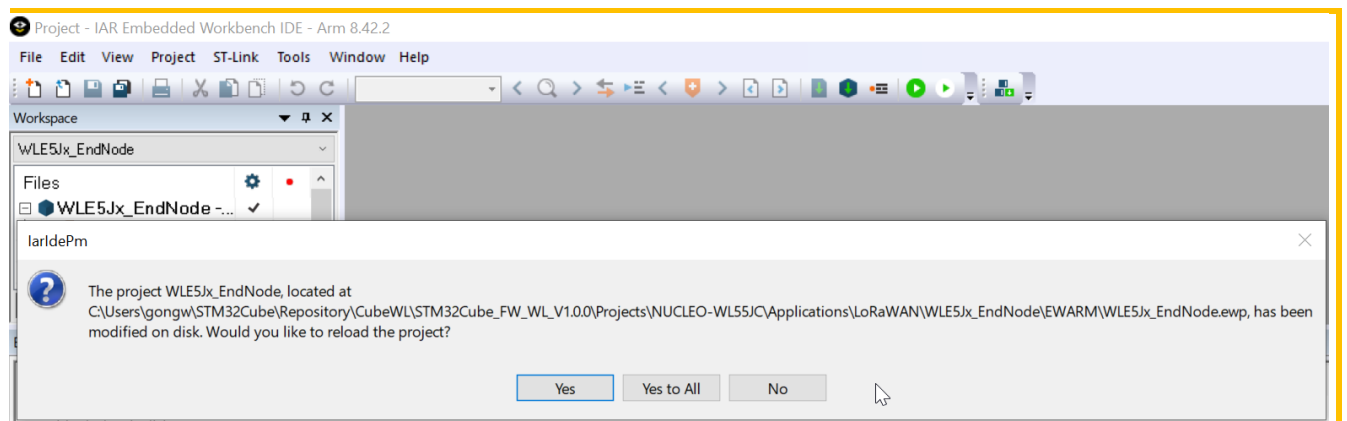
Name	Date modified	Type	Size
Core	2/10/2021 10:35	File folder	
EWARM	2/10/2021 10:35	File folder	
LoRaWAN	2/10/2021 10:35	File folder	
.mxproject	2/10/2021 10:35	MXPROJECT File	35 KB
WLE5Jx_EndNode.ioc	2/10/2021 10:33	STM32CubeMX	22 KB

4.2 编译并下载

用 IAR 打开 STM32Cube_FW_WL_V1.0.0\Projects\NUCLEO-

WL55JC\Applications\LoRaWAN\WLE5Jx_EndNode\EWARM\Project.eww

如果 IAR 之前已经打开了 WLE5Jx_EndNode，在每次重新生成代码时都如有如下提示，选择“**Yes to All**”



按 **F7** 或选择 **Project=>Make(F7)**来编译工程，如果编译无错误，选择

Project=>Download=>Download Active Application 下载 WLE5Jx_EndNode 应用

5 接入 cn1.LoRiot Network Server

如果不需要将 WL 的 Sensor 数据转发到 cn1.LoRiot Network Server 网络服务器上，可跳过此步骤。

如果要将 WL 的 Sensor 数据转发到 cn1.LoRiot Network Serve 网络服务器上，需用 CubeMX 重新配置下 LORAWAN 的一些参数。

5.1 WL Sensor End Device 信息更改

5.1.1 配置 REGION_EU868

```
Middleware=>LORAWAN=>LoRaWAN middleware=>lorawan_conf
Middleware=>LORAWAN=>LoRaWAN application=>lora_app=>Active
region=>LORAMAC_REGION_CN470
```

默认配置即可

5.1.1 配置 REGION_CN470

Middleware=>LORAWAN=>LoRaWAN middleware=>lorawan_conf

勾选 Region China freq :470 , 勾选 Enable Hybrid mode, 即在 LoRaWAN\Target\lorawan_conf.h 中使能 REGION_CN470 和 HYBRID_ENABLED

取消勾选 Region Europe freq :868 和 Region USA freq :915 , 失能 EU868 和 US915

The screenshot displays the 'LORAWAN Mode and Configuration' interface. The left sidebar shows the navigation tree with 'Middleware' expanded and 'LORAWAN' selected. The main panel shows the 'lorawan_conf' configuration section. The 'Mode' is set to 'Enabled'. The 'Configuration' section includes a 'Reset Configuration' button and tabs for 'LoRaWAN application', 'LoRaWAN commissioning', 'LoRaWAN middleware', 'User Constants', and 'Platform Settings'. The 'LoRaWAN middleware' tab is active, showing a list of regions with checkboxes. The 'Region China freq: 470' checkbox is checked, while 'Region Europe freq: 868' and 'Region USA freq: 915' are unchecked. The 'Enable Hybrid mode' checkbox is checked. Other parameters like 'radio_board_if' and 'mw_log_conf' are also visible.

Middleware=>LORAWAN=>LoRaWAN application=>lora_app=>Active region=>LORAMAC_REGION_CN470

在 LoRaWANApp\lora_app.h 中更改默认 ACTIVE_REGION 为 LORAMAC_REGION_CN470

1 Pinout & Configuration

2 Middleware

3 LORAWAN

4 Enabled

5 LoRaWAN application

6 LORAMAC_REGION_CN470

配置 CN470 的起止频率和调频频道的 MASK

Middlewares\Third_Party\LoRaWAN\Mac\region\RegionCN470.h	
<pre> /*! * Defines the first channel for RX window 1 for CN470 band */ </pre>	<pre> /*! * Defines the first channel for RX window 1 for CN470 band */ </pre>
<pre> #define CN470_FIRST_RX1_CHANNEL ((uint32_t) 500300000) </pre>	<pre> #define CN470_FIRST_RX1_CHANNEL ((uint32_t) 470300000) </pre>
<pre> /*! * Defines the last channel for RX window 1 for CN470 band */ </pre>	<pre> /*! * Defines the last channel for RX window 1 for CN470 band */ </pre>
<pre> #define CN470_LAST_RX1_CHANNEL ((uint32_t) 509700000) </pre>	<pre> #define CN470_LAST_RX1_CHANNEL ((uint32_t) 489300000) </pre>
<p>Middlewares\Third_Party\LoRaWAN\Mac\region\RegionCN470.c 更改 NvmCtx.ChannelsDefaultMask[0]以使 WL 使用更少的信道加入 LoRa 网络，这样可使 WL 更快的加入 LoRa 网络</p>	
<pre> void RegionCN470InitDefaults(InitDefaultsParams_t* params) case INIT_TYPE_DEFAULTS: { </pre>	<pre> void RegionCN470InitDefaults(InitDefaultsParams_t* params) case INIT_TYPE_DEFAULTS: { </pre>

<pre> // Default bands memcpy1((uint8_t*)NvmCtx.Bands, (uint8_t*)bands, sizeof(Band_t) * CN470_MAX_NB_BANDS); // Channels for(uint8_t i = 0; i < CN470_MAX_NB_CHANNELS; i++) { // 125 kHz channels NvmCtx.Channels[i].Frequency = 470300000 + i * 200000; NvmCtx.Channels[i].DrRange.Value = (DR_5 << 4) DR_0; NvmCtx.Channels[i].Band = 0; } // Initialize channels default mask /* ST_WORKAROUND_BEGIN: Hybrid mode */ #if (HYBRID_ENABLED == 1) NvmCtx.ChannelsDefaultMask[0] = 0x00FF; NvmCtx.ChannelsDefaultMask[1] = 0x0000; NvmCtx.ChannelsDefaultMask[2] = 0x0000; NvmCtx.ChannelsDefaultMask[3] = 0x0000; NvmCtx.ChannelsDefaultMask[4] = 0x0000; NvmCtx.ChannelsDefaultMask[5] = 0x0000; #else NvmCtx.ChannelsDefaultMask[0] = 0xFFFF; NvmCtx.ChannelsDefaultMask[1] = 0xFFFF; NvmCtx.ChannelsDefaultMask[2] = 0xFFFF; NvmCtx.ChannelsDefaultMask[3] = 0xFFFF; NvmCtx.ChannelsDefaultMask[4] = 0xFFFF; NvmCtx.ChannelsDefaultMask[5] = 0xFFFF; #endif /* HYBRID_ENABLED == 1 */ /* ST_WORKAROUND_END */ // Copy channels default mask RegionCommonChanMaskCopy(NvmCtx.ChannelsMa sk, NvmCtx.ChannelsDefaultMask, CHANNELS_MASK_SIZE); break; } </pre>	<pre> // Default bands memcpy1((uint8_t*)NvmCtx.Bands, (uint8_t*)bands, sizeof(Band_t) * CN470_MAX_NB_BANDS); // Channels for(uint8_t i = 0; i < CN470_MAX_NB_CHANNELS; i++) { // 125 kHz channels NvmCtx.Channels[i].Frequency = 470300000 + i * 200000; NvmCtx.Channels[i].DrRange.Value = (DR_5 << 4) DR_0; NvmCtx.Channels[i].Band = 0; } // Initialize channels default mask /* ST_WORKAROUND_BEGIN: Hybrid mode */ #if (HYBRID_ENABLED == 1) NvmCtx.ChannelsDefaultMask[0] = 0x03C0; NvmCtx.ChannelsDefaultMask[1] = 0x0000; NvmCtx.ChannelsDefaultMask[2] = 0x0000; NvmCtx.ChannelsDefaultMask[3] = 0x0000; NvmCtx.ChannelsDefaultMask[4] = 0x0000; NvmCtx.ChannelsDefaultMask[5] = 0x0000; #else NvmCtx.ChannelsDefaultMask[0] = 0xFFFF; NvmCtx.ChannelsDefaultMask[1] = 0xFFFF; NvmCtx.ChannelsDefaultMask[2] = 0xFFFF; NvmCtx.ChannelsDefaultMask[3] = 0xFFFF; NvmCtx.ChannelsDefaultMask[4] = 0xFFFF; NvmCtx.ChannelsDefaultMask[5] = 0xFFFF; #endif /* HYBRID_ENABLED == 1 */ /* ST_WORKAROUND_END */ // Copy channels default mask RegionCommonChanMaskCopy(NvmCtx.ChannelsMa sk, NvmCtx.ChannelsDefaultMask, CHANNELS_MASK_SIZE); break; } </pre>
--	--

5.2 配置 DevEUI, AppEUI 和 AppKey

在 LoRaWAN\App\se-identity.h 中更改设备信息 DevEUI, AppEUI 和 AppKey

DevAddr 不需要设置静态地址，可以配置的

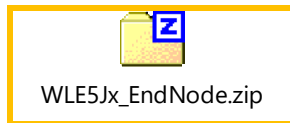
在 LoRaWAN\App\se-identity.h 中更改设备信息
<pre> /*! * When set to 1 DevEui is LORAWAN_DEVICE_EUI * When set to 0 DevEui is automatically set with a value provided by MCU platform */ #define STATIC_DEVICE_EUI 1 /*! * end-device IEEE EUI (big endian) */ // #define LORAWAN_DEVICE_EUI { 0x00, 0x80, 0xE1, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01 } #define LORAWAN_DEVICE_EUI { 0x01, 0x50, 0x36, 0x32, 0x73, 0x33, 0x45, 0x20 } /*! * App/Join server IEEE EUI (big endian) */ #define LORAWAN_JOIN_EUI { 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01 } </pre>

<pre> /*! * When set to 1 DevdAddr is LORAWAN_DEVICE_ADDRESS * When set to 0 DevdAddr is automatically set with a value provided by a pseudo * random generator seeded with a value provided by the MCU platform */ #define STATIC_DEVICE_ADDRESS 1 </pre>	
<pre> /*! * Device address on the network (big endian) * * \remark see STATIC_DEVICE_ADDRESS comments */ // #define LORAWAN_DEVICE_ADDRESS (uint32_t)0x0100000A #define LORAWAN_DEVICE_ADDRESS (uint32_t)0x303CC786 </pre>	
<pre> /*! * Application root key */ #define LORAWAN_APP_KEY 2B,7E,15,16,28,AE,D2,A6,AB,F7,15,88,09,CF,4F,3C /*! * Network root key */ #define LORAWAN_NWK_KEY 2B,7E,15,16,28,AE,D2,A6,AB,F7,15,88,09,CF,4F,3C /*! * Forwarding Network session key */ #define LORAWAN_NWK_S_KEY 2B,7E,15,16,28,AE,D2,A6,AB,F7,15,88,09,CF,4F,3C /*! * Application session key */ #define LORAWAN_APP_S_KEY 2B,7E,15,16,28,AE,D2,A6,AB,F7,15,88,09,CF,4F,3C </pre>	

5.3 转发到 myDeuces Cayenne 应用服务器(可选)

LoRaWAN\App\lora_app.h 中使能 CAYENNE_LPP	
<pre> /*! * CAYENNE_LPP is myDevices Application server. */ /*#define CAYENNE_LPP*/ </pre>	<pre> /*! * CAYENNE_LPP is myDevices Application server. */ #define CAYENNE_LPP </pre>

5.4 WL Sensor End Device 信息获取和确认



其中包含更新过的 WLE5Jx_EndNode.ioc

用 IAR 编译 Projects\NUCLEO-WL55JC\Applications\LoRaWAN\WLE5Jx_EndNode\EWARM\Project.eww，并下载到 WL 中，注意 STLINKv3 的设置。

连接 WL 串口到 PC，PC 端打开串口，配置如下，然后重启 WL，会看到 DEV EUI，APP EUI，APP KEY

Open a terminal emulation software such as Tera Term and configure it with the following settings:

- Port: (Virtual COM port number assigned to the board)
- Baud rate: 115200
- Data: 8 bit
- Parity: none
- Stop: 1 bit

```

APP_VERSION:    V1.0.0
MW_LORAWAN_VERSION: V2.2.1
MW_RADIO_VERSION: V0.6.1
##### OTAA #####
##### AppKey: 2B 7E 15 16 28 AE D2 A6 AB F7 15 88 09 CF 4F 3C
##### NwkKey: 2B 7E 15 16 28 AE D2 A6 AB F7 15 88 09 CF 4F 3C
##### ABP #####
##### AppSKey: 2B 7E 15 16 28 AE D2 A6 AB F7 15 88 09 CF 4F 3C
##### NwkSKey: 2B 7E 15 16 28 AE D2 A6 AB F7 15 88 09 CF 4F 3C
##### DevEui: 01-50-36-32-73-33-45-20
##### AppEui: 01-01-01-01-01-01-01-01
0s111:TX on freq 471700000 Hz at DR 0
1s614:MAC txDone
6s636:RX_1 on freq 471700000 Hz at DR 0
6s774:PRE OK
7s309:HDR OK
7s965:MAC rxDone

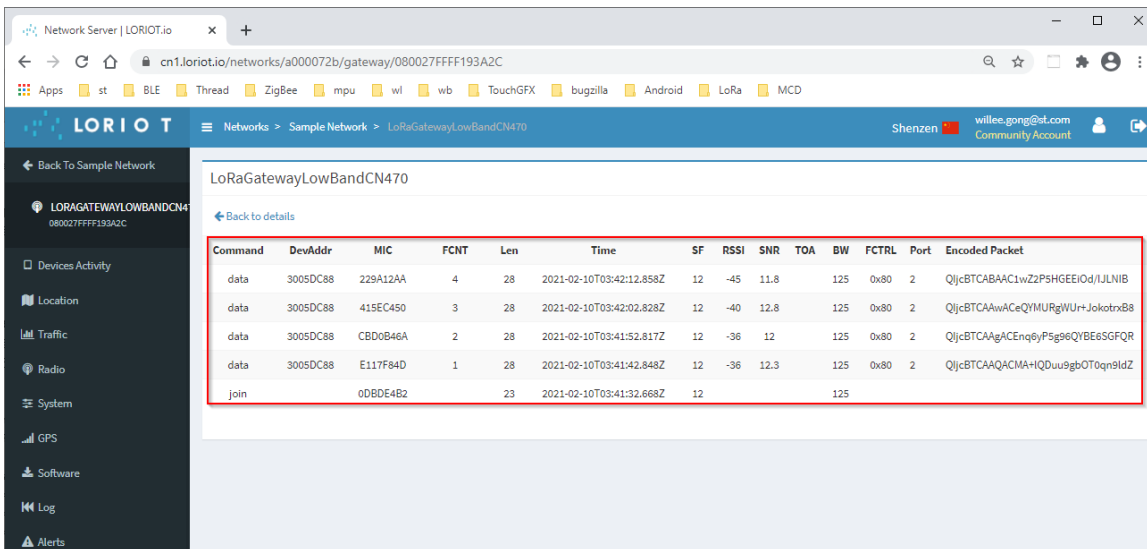
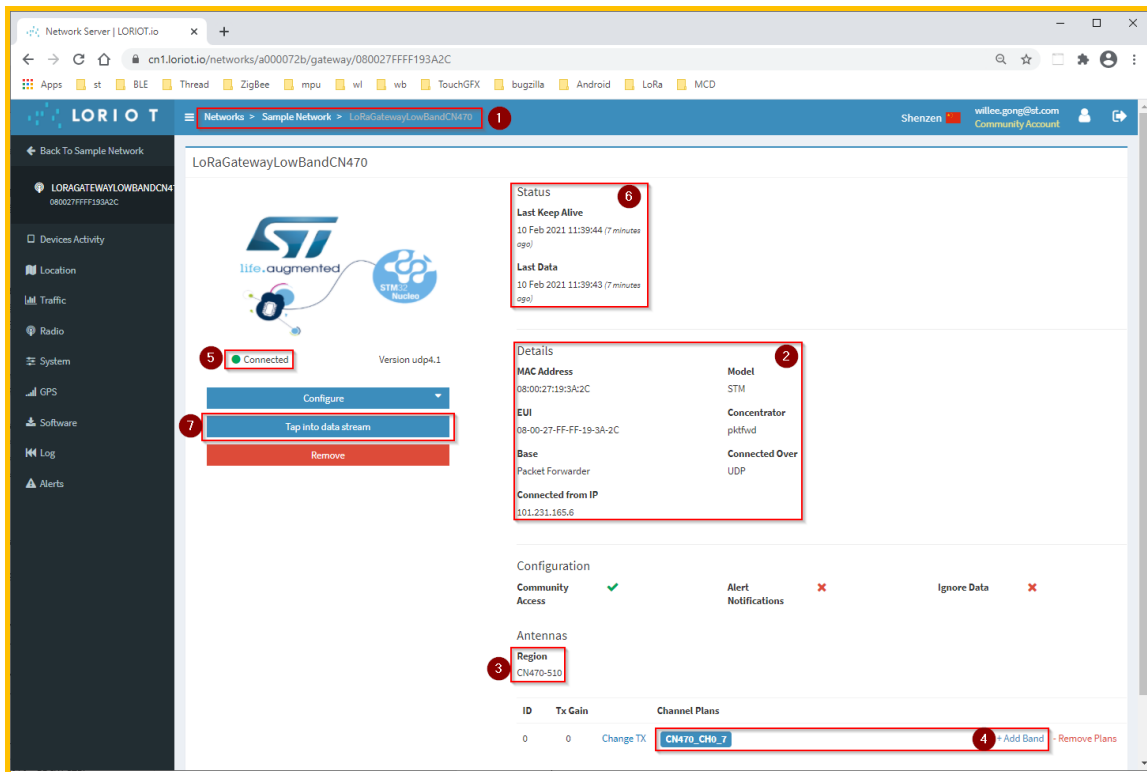
##### = JOINED = OTAA =====
10s132:temp= -274
10s132:VDDA= 0
10s133:TX on freq 472100000 Hz at DR 0
10s144:SEND REQUEST
11s791:MAC txDone
12s813:RX_1 on freq 472100000 Hz at DR 0
13s020:MAC rxTimeOut
13s813:RX_2 on freq 505300000 Hz at DR 0
    
```

```

14s020:MAC rxTimeOut

##### ===== MCPS-Confirm =====
20s132:temp= -274
20s132:VDDA= 0
    
```

Device Details / 0150363273334520	
Name	WNodeLbc80
Device Profile	No Device Profile Linked
Battery	100%
ADR	Enabled
Description	WNodeLbc80 sensor
EUI	0150363273334520 big endian (use by default) 2045337332365001 little endian (for LoRaWAN® non-compliant devices)
AppEUI	0101010101010101 big endian (use by default) 0101010101010101 little endian (for LoRaWAN® non-compliant devices)
DevAddr	31F7BACC big endian (use by default) CCBAF731 little endian (for LoRaWAN® non-compliant devices)



版本历史

日期	版本	变更
2022年04月11日	1.0	首版发布

重要通知 - 请仔细阅读

意法半导体公司及其子公司 (“ST”) 保留随时对 ST 产品和 / 或本文档进行变更的权利，恕不另行通知。买方在订货之前应获取关于 ST 产品的最新信息。ST 产品的销售依照订单确认时的相关 ST 销售条款。

买方自行负责对 ST 产品的选择和使用，ST 概不承担与应用协助或买方产品设计相关的任何责任。

ST 不对任何知识产权进行任何明示或默示的授权或许可。

转售的 ST 产品如有不同于此处提供的信息的规定，将导致 ST 针对该产品授予的任何保证失效。

ST 和 ST 徽标是 ST 的商标。若需 ST 商标的更多信息，请参考 www.st.com/trademarks。所有其他产品或服务名称均为其各自所有者的财产。

本文档是 ST 中国本地团队的技术性文章，旨在交流与分享，并期望借此给予客户产品应用上足够的帮助或提醒。若文中内容存有局限或与 ST 官网资料不一致，请以实际应用验证结果和 ST 官网最新发布的内容为准。您拥有完全自主权是否采纳本文档（包括代码，电路图等信息，我们也不承担因使用或采纳本文档内容而导致的任何风险。

本文档中的信息取代本文档所有早期版本中提供的信息。