

1. STM32WL 简介
2. STM32WL 硬件简介
3. STM32WL 软件简介
4. LoRa和LoRaWAN介绍
5. STM32WL LoRa 例程介绍
6. STM32WL 使用STM32 CubeMX 创建LoRa 节点应用
7. STM32WL LoRa RF 测试
8. STM32WL 安全特性介绍
9. STM32WL FUOTA 应用设计



life.augmented

STM32WL 使用STM32CubeMX 创建LoRa 节点应用

David Liu



- 1 STM32CubeMX介绍
- 2 使用CubeMX创建LoRaWAN节点
- 3 总结

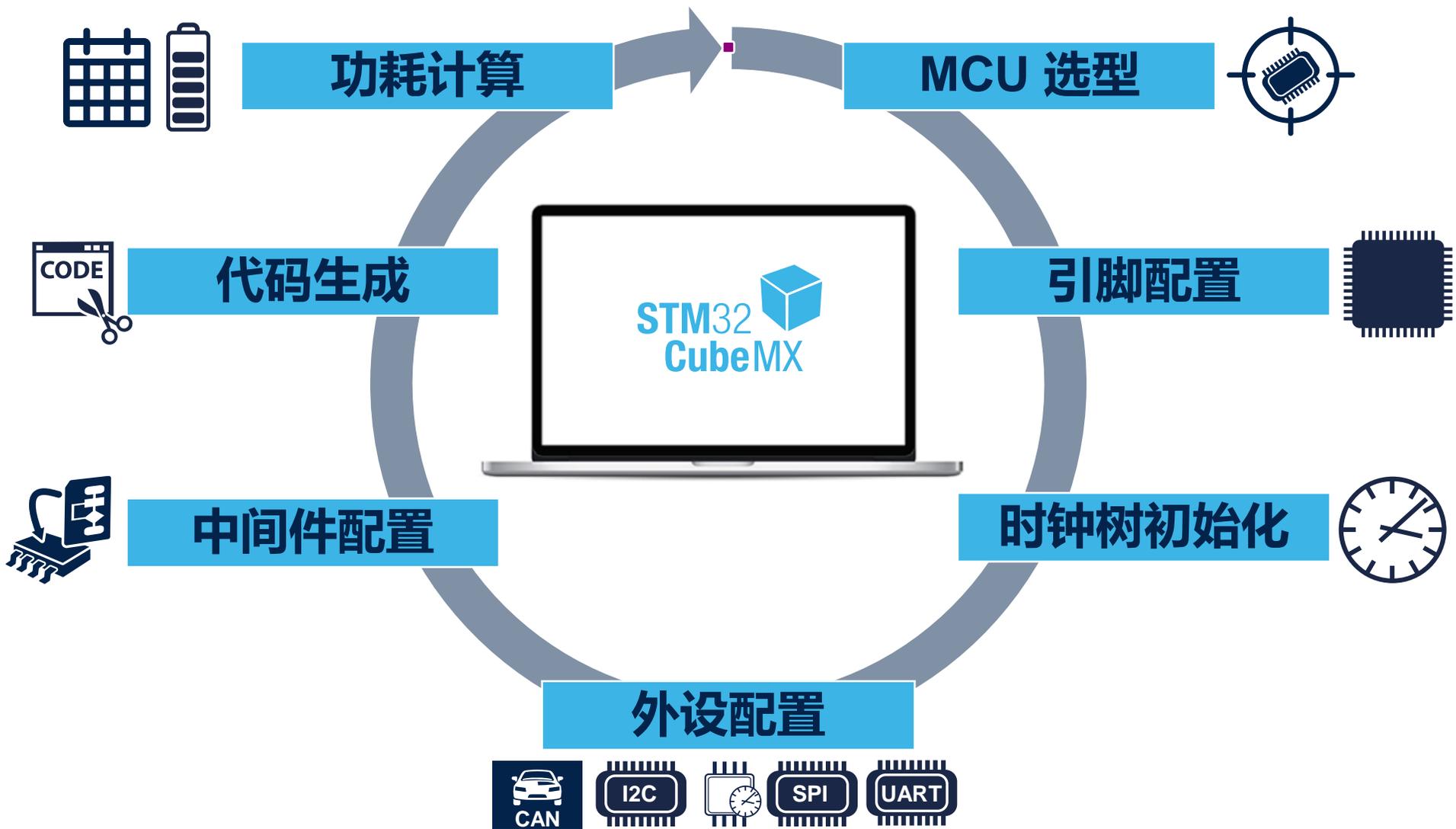


life.augmented

STM32CubeMX介绍



STM32CubeMX主要功能



安装设置

- ST官网下载最新版本
 - 如：STM32CubeMX 6.2.0, 下载
- 支持跨平台运行：Windows, Linux 和 macOS
- 安装之后，按Alt+S来配置更新器——不仅用于GUI，还用于Cube FW库
- 选择软件库存放位置。

Get Software					
Part Number	General Description	Software Version	Download	Previous versions	
+ STM32CubeMX-Lin	STM32Cube init code generator for Linux	6.2.0	Download	Select version ▾	
+ STM32CubeMX-Mac	STM32Cube init code generator for macOS	6.2.0	Download	Select version ▾	
+ STM32CubeMX-Win	STM32Cube init code generator for Windows	6.2.0	Download	Select version ▾	

MX Updater Settings

Updater Settings | Connection Parameters

Firmware Repository

Repository Folder

C:/Users/.../STM32Cube/Repository/ [Browse](#)

Check and Update Settings

Manual Check

Automatic Check Interval between two Checks (days)

Data Auto-Refresh

No Auto-Refresh at Application start

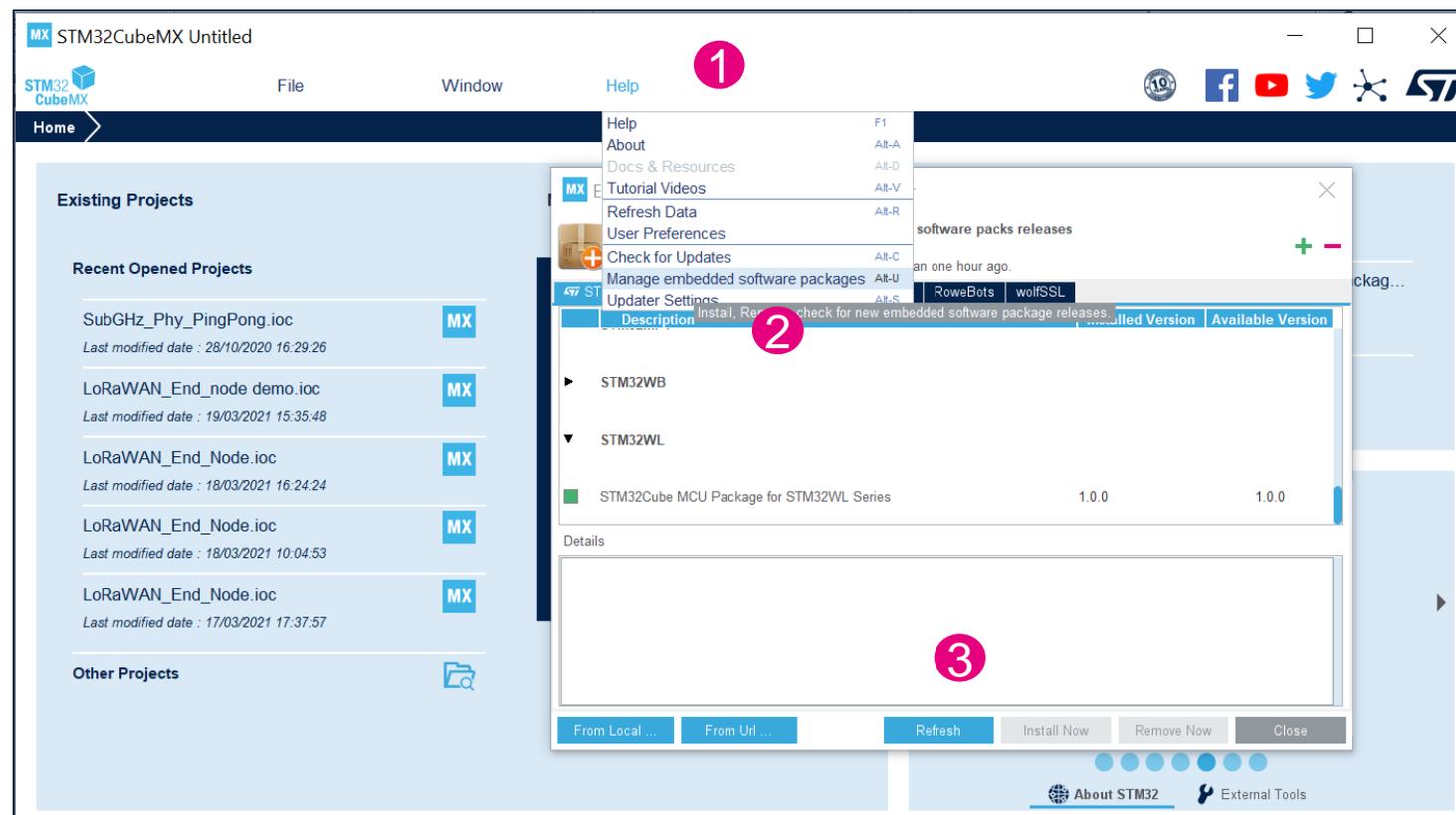
Auto-Refresh Data-only at Application start

Auto-Refresh Data and Docs at Application start

Interval between two data-refreshes (days)

[OK](#) [Cancel](#)

- 在Help下拉菜单选择Manage Embedded Software Packages 可以查看支持的开发包
- 点击Refresh可以检测有无最新的开发包，如果有可选择，然后点击Install Now



开始一个STM32CubeMX工程

The screenshot displays the STM32CubeMX software interface. The window title is "MX STM32CubeMX Untitled". The menu bar includes "File", "Window", and "Help". The interface is divided into several sections:

- Existing Projects:** Lists "Recent Opened Projects" and "Other Projects". Under "Recent Opened Projects", three items are visible: "SubGHz_Phy_PingPong.ioc" (last modified 19/03/2021 15:35:48), "LoRaWAN_End_Node.ioc" (last modified 17/03/2021 17:37:57), and another "LoRaWAN" project. Three yellow callout boxes are overlaid on these items, containing the text: "从选择MCU型号开始", "从选择开发板开始", and "从选择例程开始".
- New Project:** A dark blue panel titled "I need to :" with three options, each with a red-bordered button: "Start My project from MCU" (ACCESS TO MCU SELECTOR), "Start My project from ST Board" (ACCESS TO BOARD SELECTOR), and "Start My project from Example" (ACCESS TO EXAMPLE SELECTOR).
- Manage software installations:** A light blue panel with two sections: "Check for STM32CubeMX and embedded software package..." with a "CHECK FOR UPDATES" button, and "Install or remove embedded software packages" with an "INSTALL / REMOVE" button.
- Safety Certifications:** A blue banner at the bottom right featuring shields for "SIL Ready", "ASIL Ready", and "ClassB Ready", along with a "Partner Program" logo. Below the shields, it says "Build your certified safety system with STM32 and STM8".

At the bottom of the interface, there are links for "About STM32" and "External Tools".

从MCU开始一个STM32CubeMX工程

过滤项

- 选中合适的芯片，双击，或者点击右上角的Start Project.

芯片说明

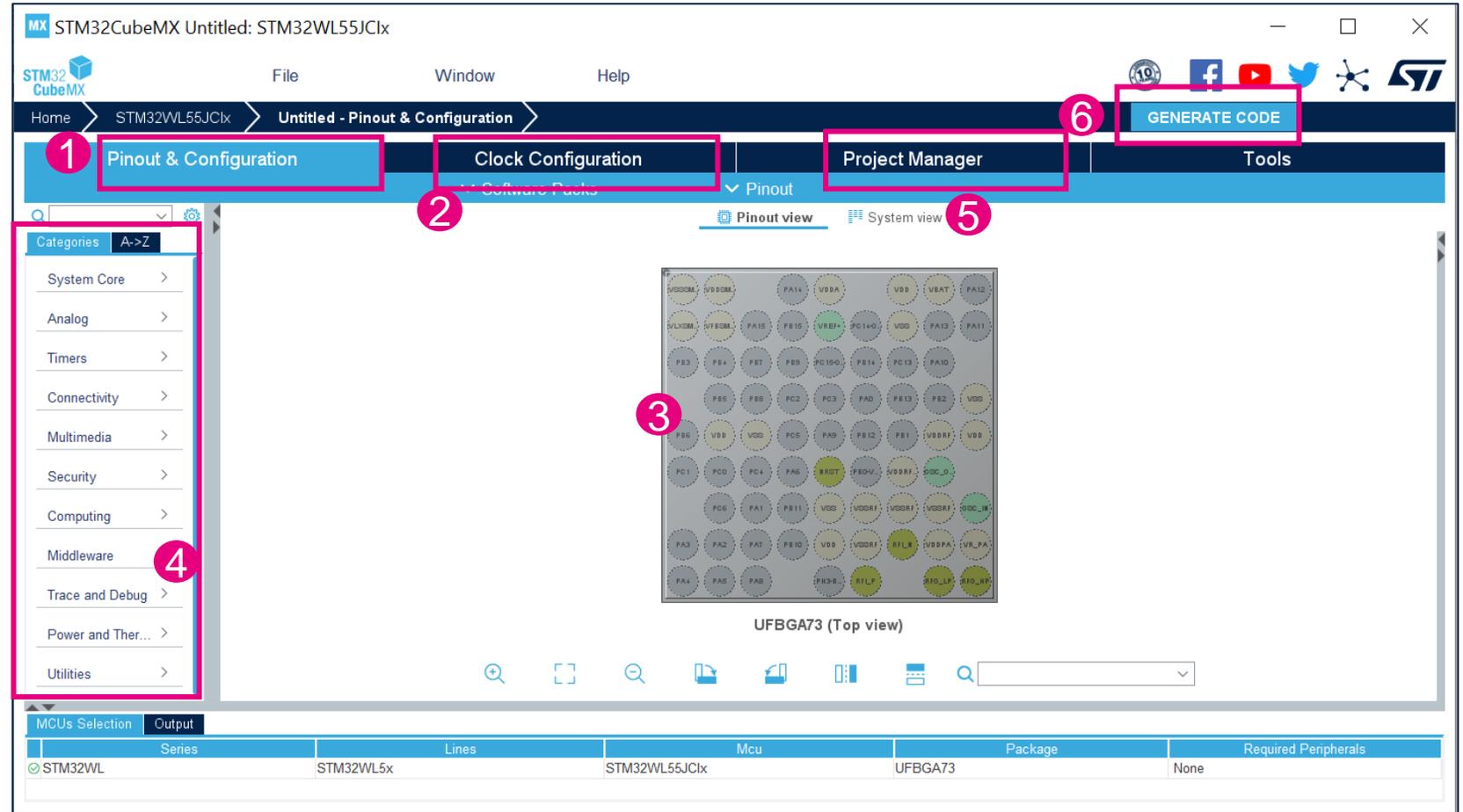
芯片列表

*	Part No	Reference	Marketing ...	Unit Price for ...	Board	Package	Flash	RAM	IO	Freq.
☆	STM32WL54...	STM32WL54C...	Active	3.286		UFQFPN48	256 kBytes	20 kBytes	29	48 MHz
☆	STM32WL54JC	STM32WL54J...	Active	3.668		UFPGA73	256 kBytes	64 kBytes	43	48 MHz
☆	STM32WL55...	STM32WL55C...	Active	3.564		UFQFPN48	256 kBytes	64 kBytes	29	48 MHz
☆	STM32WL55JC	STM32WL55J...	Active	3.945	NUCLEO...NUCLEO...NUCLEO...	UFPGA73	256 kBytes	64 kBytes	43	48 MHz
☆	STM32WL55...	STM32WL55U...	NA	NA		WLCSP59	256 kBytes	64 kBytes	22	48 MHz
☆	STM32WLE4...	STM32WLE4...	NA	NA		UFQFPN48	64 kBytes	64 kBytes	29	48 MHz
☆	STM32WLE4...	STM32WLE4...	NA	NA		UFQFPN48	128 kBytes	64 kBytes	29	48 MHz
☆	STM32WLE4...	STM32WLE4...	Active	2.777		UFQFPN48	256 kBytes	64 kBytes	29	48 MHz
☆	STM32WLE4J8	STM32WLE4J...	NA	NA		UFPGA73	64 kBytes	20 kBytes	43	48 MHz
☆	STM32WLE4...	STM32WLE4J...	NA	NA		UFPGA73	128 kBytes	48 kBytes	43	48 MHz
☆	STM32WLE4...	STM32WLE4J...	Active	3.159		UFPGA73	256 kBytes	64 kBytes	43	48 MHz
☆	STM32WLE5...	STM32WLE5...	Active	2.545		UFQFPN48	64 kBytes	20 kBytes	29	48 MHz
☆	STM32WLE5...	STM32WLE5...	Active	2.73		UFQFPN48	128 kBytes	48 kBytes	29	48 MHz
☆	STM32WLE5...	STM32WLE5...	Active	3.054		UFQFPN48	256 kBytes	64 kBytes	29	48 MHz
☆	STM32WI F5.I8	STM32WI F5.I	Active	2.927		UFRGA73	64 kBytes	20 kBytes	43	48 MHz

进入CubeMX界面

- 进入STM32CubeMX工程界面.

1. 进行引脚配置
2. 时钟树初始化
3. 外设配置
4. 中间件配置
5. 项目管理
6. 代码生成



代码生成

The screenshot displays the STM32CubeMX software interface. The title bar reads "MX STM32CubeMX Untitled: STM32WL55JClx". The menu bar includes "File", "Window", and "Help". The breadcrumb navigation shows "Home > STM32WL55JClx > Untitled - Project Manager". A "GENERATE CODE" button is visible in the top right. The main interface is divided into four tabs: "Pinout & Configuration", "Clock Configuration", "Project Manager" (highlighted with a red box), and "Tools".

The "Project Manager" tab contains the following settings:

- Project Settings:**
 - Project Name: [Empty text box]
 - Project Location: C:\Work\STM32WL\STM32WL Project\STM32WL CubeMx project\ [Browse button]
 - Application Structure: Advanced [Dropdown] Do not generate the main()
 - Toolchain Folder Location: C:\Work\STM32WL\STM32WL Project\STM32WL CubeMx project\
 - Toolchain / IDE: EWARM [Dropdown] Min Version: V8 [Dropdown] Generate Under Root
- Linker Settings:**
 - Minimum Heap Size: 0x200 [Text box]
 - Minimum Stack Size: 0x400 [Text box]
- Mcu and Firmware Package:**
 - Mcu Reference: STM32WL55JClx
 - Firmware Package Name and Version: [Empty text box]

On the left side, there are three blue panels with red borders: "Project", "Code Generator", and "Advanced Settings".

At the bottom, there is a table with two tabs: "MCUs Selection" and "Output". The "Output" tab is active, showing a table with the following data:

	Series	Lines	Mcu	Package	Required Peripherals
✔	STM32WL	STM32WL5x	STM32WL55JClx	UFBGA73	None

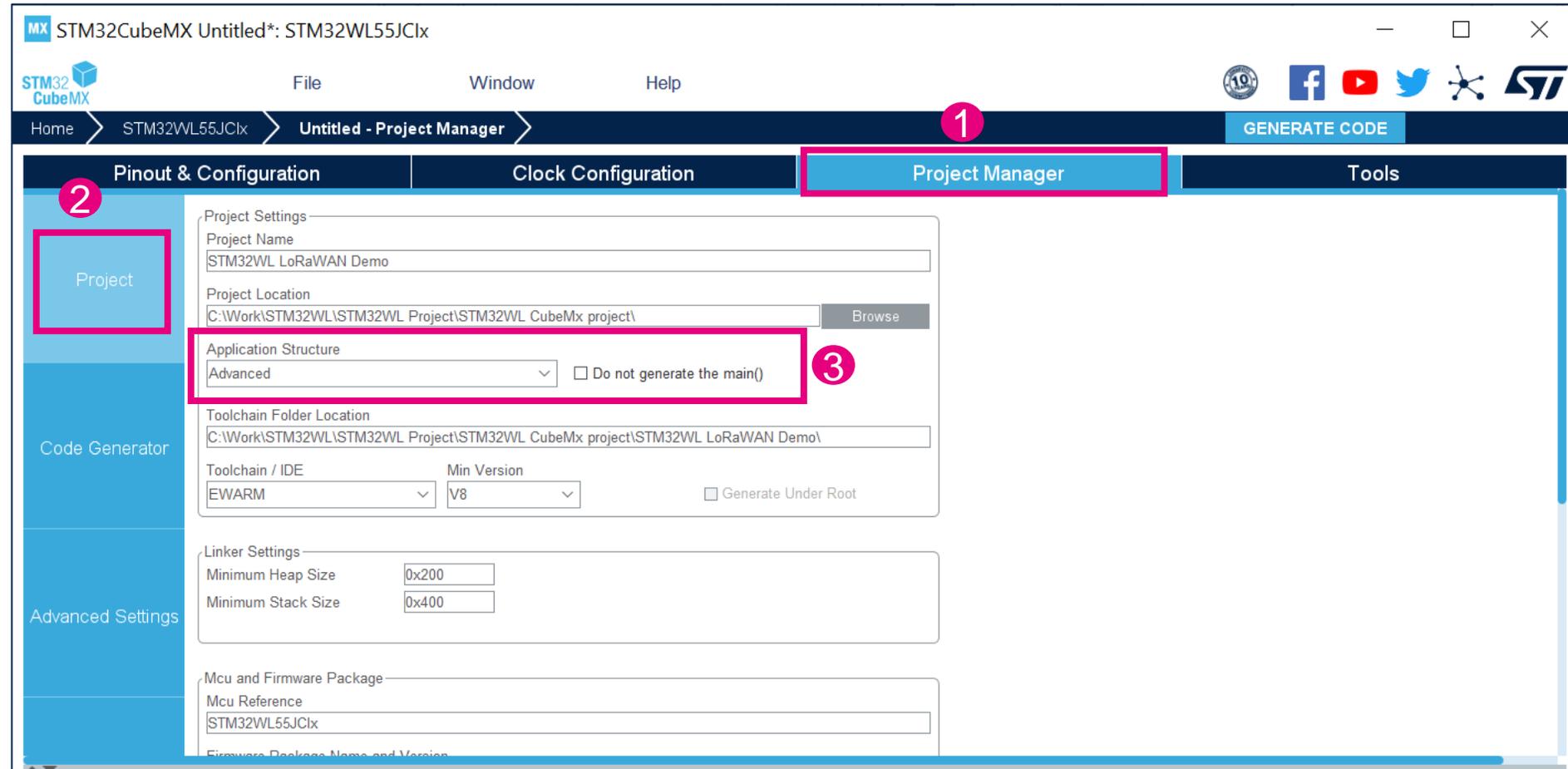
代码生成 ——选择不同的项目结构

Basic Structure:

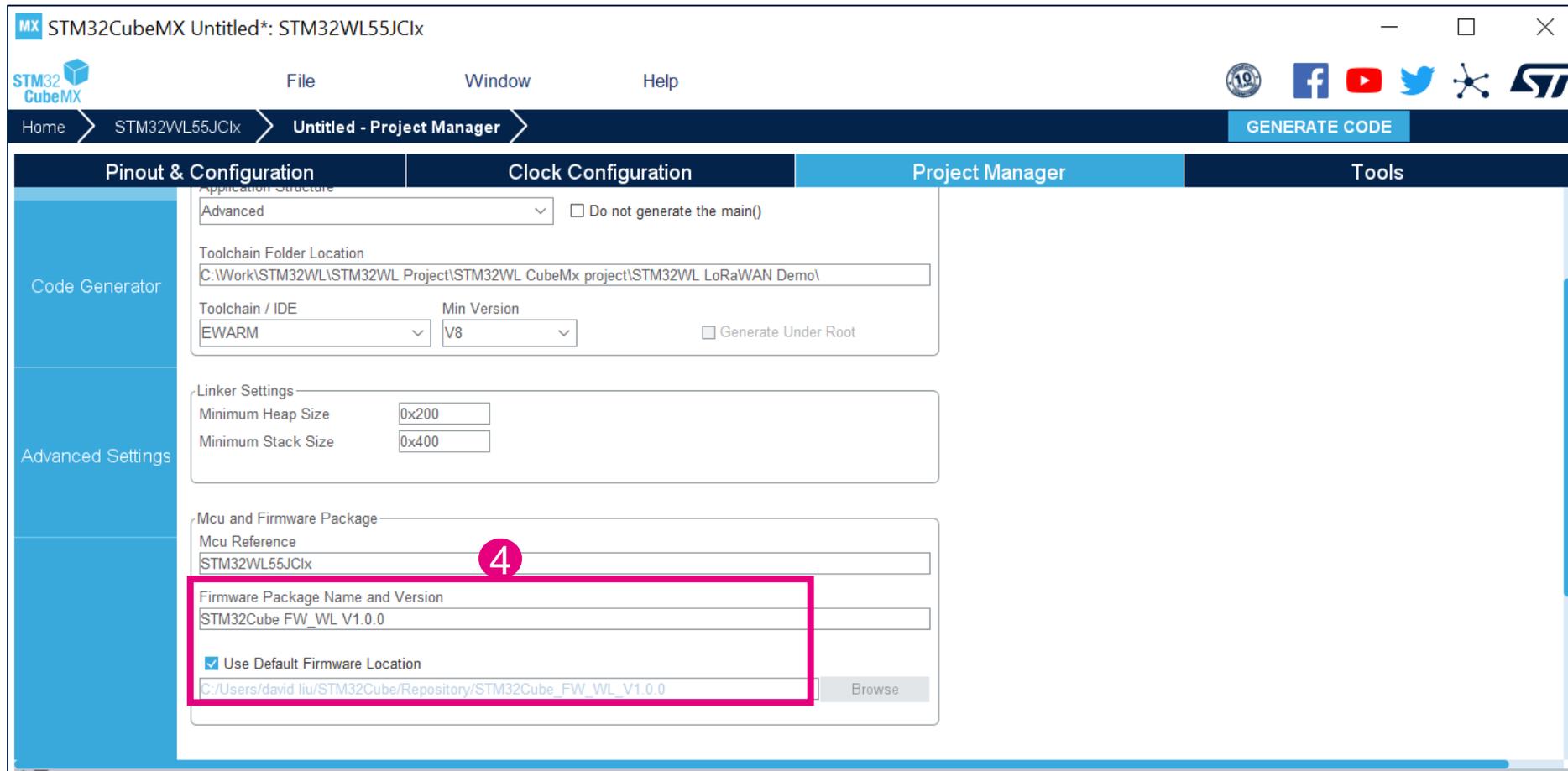
适合没有使用中间件或者只使用了一种中间件的项目。用户代码分为Src和Inc两个文件夹，和IDE文件夹位于同一层

Advanced Structure:

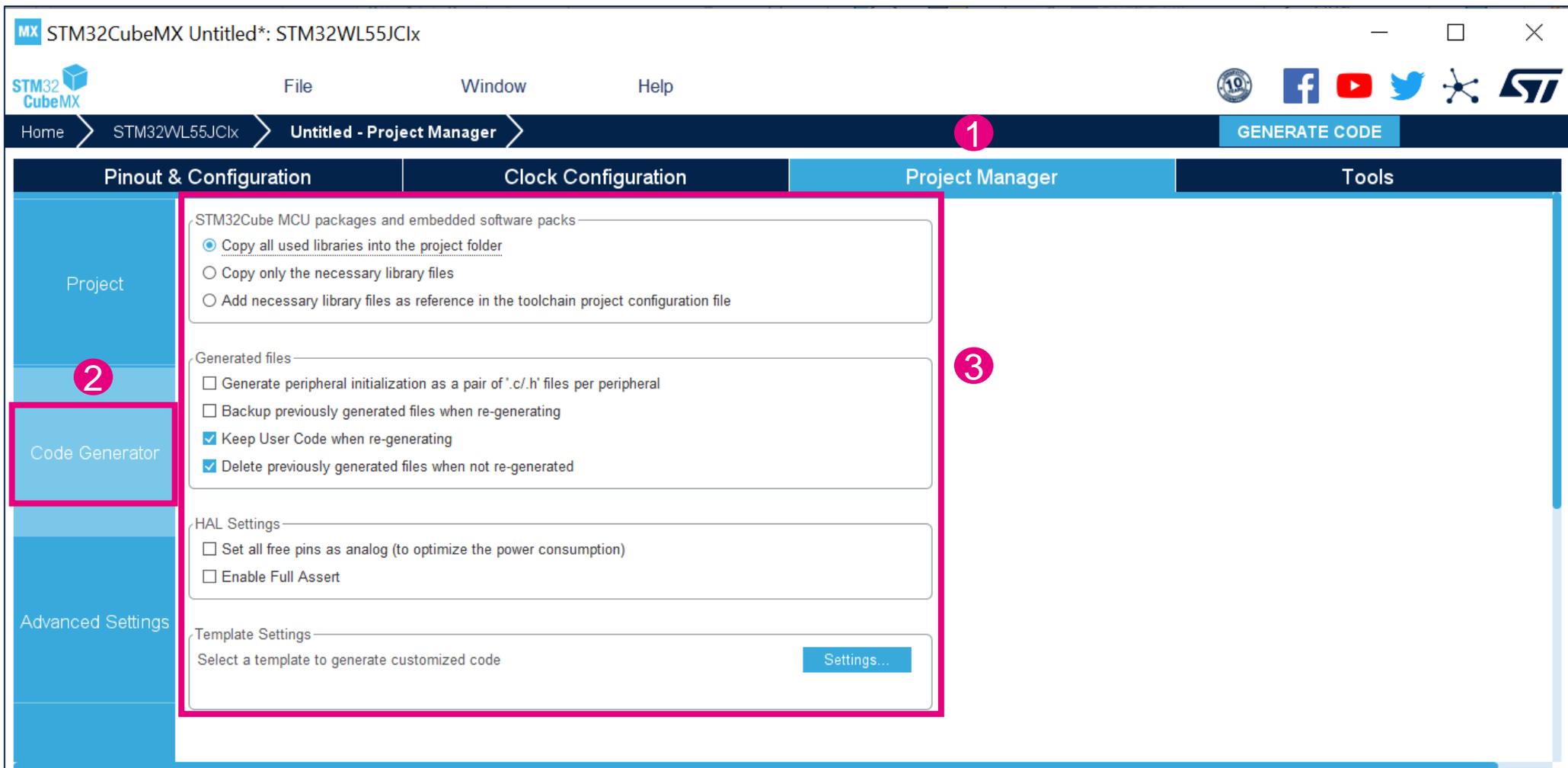
用户应用逻辑代码放在Core文件夹下，每个中间件的适配文件各自放在独立的文件夹内



代码生成 ——选择不同的Cube库版本



代码生成 ——设置代码相关选项



代码生成 高级设置

MX STM32CubeMX LoRaWAN_End_Node.ioc: STM32WL55JClx

1 Project Manager

选择HAL或LL接口

2 Advanced Settings

Generate Code	Rank	Function Name	Peripheral Instance Name	Do Not Generate Function Call	Visibility (Static)
<input type="checkbox"/>	1	MX_GPIO_Init	GPIO	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	2	MX_DMA_Init	DMA	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	3	SystemClock_Config	RCC	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	4	MX_ADC_Init	ADC	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	5	MX_RTC_Init	RTC	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	6	MX_SUBGHZ_Init	SUBGHZ	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	7	MX_LoRaWAN_Init	LORAWAN	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	8	MX_USART2_UART_Init	USART2	<input checked="" type="checkbox"/>	<input type="checkbox"/>

是否调用该函数

是否生成静态函数

是否注册对应的callback函数

代码生成

The screenshot shows the STM32CubeMX software interface. The title bar reads "MX STM32CubeMX Untitled: STM32WL55JClx". The menu bar includes "File", "Window", and "Help". The breadcrumb navigation shows "Home > STM32WL55JClx > Untitled - Project Manager". A pink box highlights the "GENERATE CODE" button in the top right corner. The main workspace is divided into four tabs: "Pinout & Configuration", "Clock Configuration", "Project Manager", and "Tools". The "Project Manager" tab is active, showing a form with the following sections:

- Project Settings:** Project Name (text input), Project Location (text input with "Browse" button, value: C:\Work\STM32WL\STM32WL Project\STM32WL CubeMx project\), Application Structure (dropdown: Advanced, checkbox: Do not generate the main()), Toolchain Folder Location (text input, value: C:\Work\STM32WL\STM32WL Project\STM32WL CubeMx project\).
- Code Generator:** Toolchain / IDE (dropdown: EWARM), Min Version (dropdown: V8), checkbox: Generate Under Root.
- Linker Settings:** Minimum Heap Size (text input: 0x200), Minimum Stack Size (text input: 0x400).
- Mcu and Firmware Package:** Mcu Reference (text input: STM32WL55JClx), Firmware Package Name and Version (text input).

At the bottom, the "MCUs Selection" tab is active, showing a table with the following data:

	Series	Lines	Mcu	Package	Required Peripherals
✓	STM32WL	STM32WL5x	STM32WL55JClx	UFPGA73	None

功耗计算

The screenshot displays the STM32CubeMX software interface for a project named "STM32WL55JClx". The "Tools" tab is highlighted with a red box. The "Power" configuration panel is active, showing a sequence table and summary statistics.

Default Sequence Table

Step	Mode	Vdd	Range/Scale	Memory	CPU/Bus Freq	Clock Config	Peripherals	Step Current	Duration
1	RUN	3.0	Range1-Medi...	SRAM1	48 MHz	MSI		5.65 mA	0.1 ms
2	STOP2	3.0	NoRange	NA	0 Hz	ALL_CLOCK...		885 nA	0.9 ms

Sequence Information Notes

- 1/ Manual change in Sequence Table will disable the **Auto Refresh** of the Sequence Generator
- 2/ **Default** sequence = RUN at max CPU frequency + STOP with the lowest consumption
- 3/ **Default** sequence is an example which does not match any pinout, configuration nor clock settings and can be directly edited for reuse or removed
- 4/ PCC sequence has **no impact** on code generation

Display Selection

Select your Preferred Display Plot: All Steps

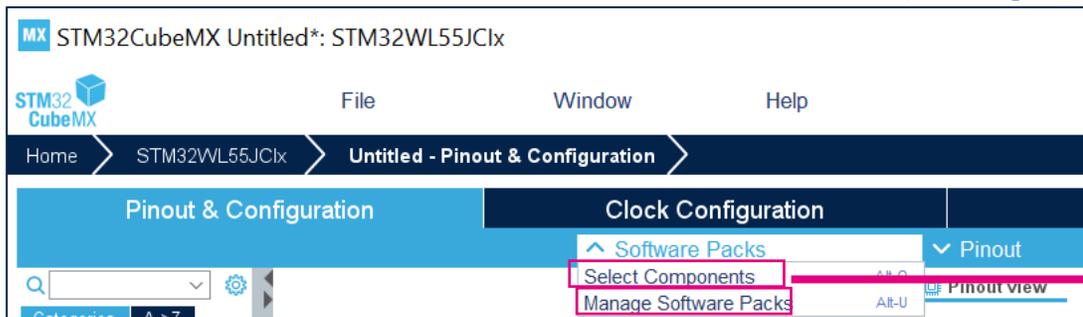
Sequence Time / Ta Max 1 ms / 124.47 °C

Battery Life Estimation 1 month, 21 days, 1 hour

Average Consumption 565.8 µA

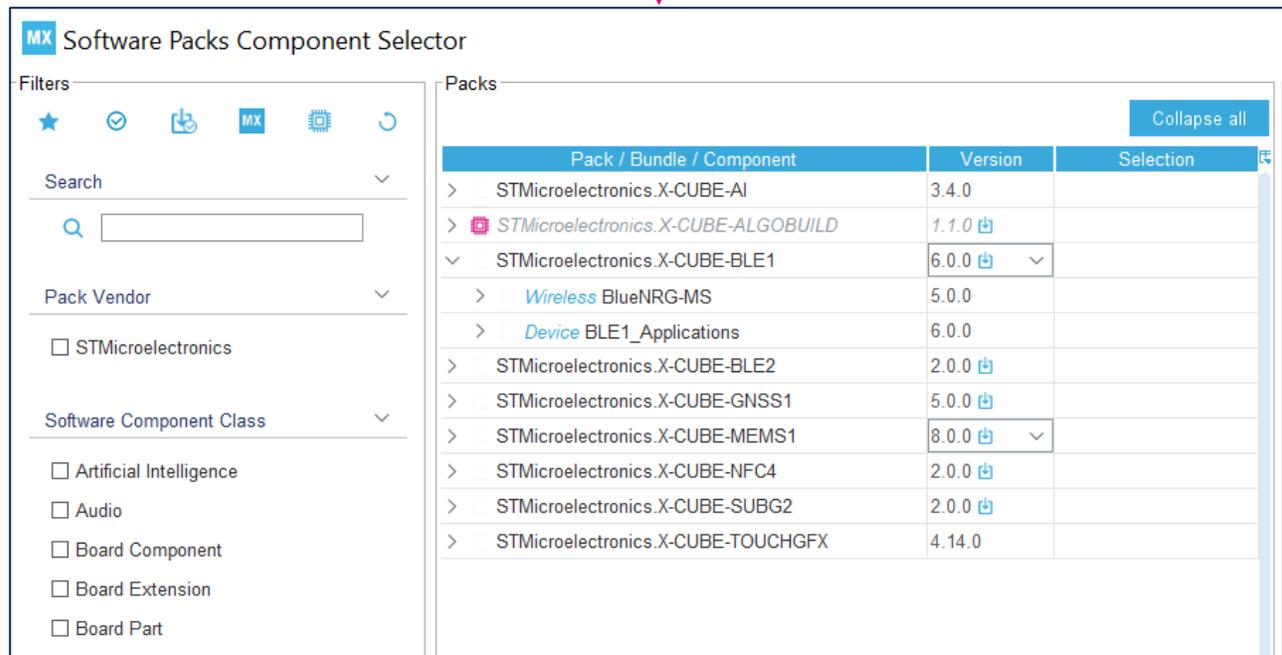
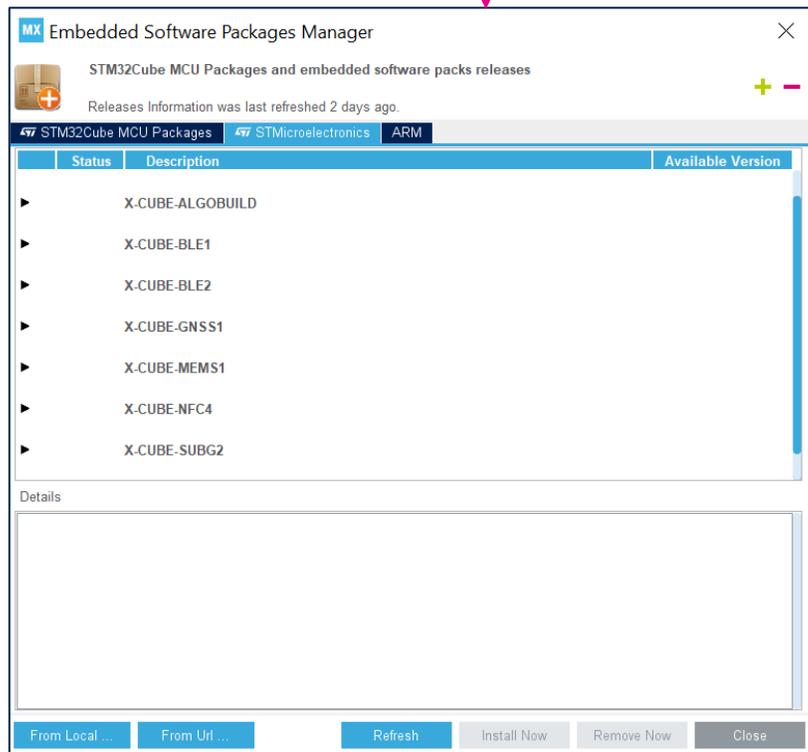
Average DMIPS 60 DMIPS

如何使用STM32Cube功能扩展插件



添加插件组件
并进行配置

安装插件





life.augmented

使用CubeMX创建LoRaWAN节点



开始一个LoRaWAN 节点工程

- 从MCU开始一个工程
- 从Example开始一个工程



life.augmented

从MCU开始一个工程



从MCU/开发板开始一个工程举例 (1/18)

The screenshot displays the STM32CubeMX software interface. The window title is "MX STM32CubeMX Untitled". The menu bar includes "File", "Window", and "Help". The main content area is divided into three sections:

- Existing Projects:** Contains a list of "Recent Opened Projects" and "Other Projects". The first project, "SubGHz_Phy_PingPong.ioc", is highlighted with a yellow box and a red circle containing the number "1". A yellow text box with the Chinese characters "从选择MCU型号开始" (Start from selecting MCU model) is overlaid on this project.
- New Project:** A dark blue panel with the heading "I need to :". It contains three options, each with a red-bordered button:
 - "Start My project from MCU" with the button "ACCESS TO MCU SELECTOR".
 - "Start My project from ST Board" with the button "ACCESS TO BOARD SELECTOR".
 - "Start My project from Example" with the button "ACCESS TO EXAMPLE SELECTOR".
- Manage software installations:** Contains a section for checking updates and installing/removing packages.

At the bottom of the interface, there are several logos and text: "SIL Ready", "ASIL Ready", "ClassB Ready", "Partner Program", "Build your certified safety system with STM32 and STM8", and the ST logo.

从MCU开始一个工程举例 (2/18)

过滤项

- 选中STM32WL55JC芯片，双击

MX New Project from a MCU/MPU

MCU/MPU Selector Board Selector Example Selector Cross Selector

芯片说明

STM32WL Series

STM32WL55JC

Sub-GHz Wireless Microcontrollers. Dual-core Arm Cortex-M4/M0+ @48 MHz with 256 Kbytes of Flash memory, 64 Kbytes of SRAM. LoRa, (G)FSK, (G)MSK, BPSK modulations. AES 256-bit. Multiprotocol System-on-Chip.

Unit Price for 10kU (US\$) : 3.945

Boards: NUCLEO-WL55JC - NUCLEO-WL55JC1 - NUCLEO-WL55JC2

UFPGA73

The STM32WL55/54xx long-range wireless and ultra-low-power devices embed a powerful and ultra-low-power LPWAN-compliant radio solution, enabling the following modulations: LoRa®, (G)FSK, (G)MSK, and BPSK. The LoRa® modulation is available in STM32WLx5xx only. These devices are designed to be extremely low-power and are based on the high-performance Arm® Cortex®-M4 32-bit RISC core operating at a frequency of up to 48 MHz. This core implements a full set of DSP instructions. It is complemented by an Arm® Cortex®-M0+ microcontroller. Both cores implement an independent memory protection unit (MPU) that enhances the application security.

MCUs/MPUs List: 19 items

	Part No	Reference	Marketing	Unit Price for	Board	Package	Flash	RAM	IO	Freq.
☆	STM32WL54...	STM32WL54C...	Active	3.286		UFQFPN48	256 kBytes	20 kBytes	29	48 MHz
☆	STM32WL54JC	STM32WL54J...	Active	3.668		UFPGA73	256 kBytes	64 kBytes	43	48 MHz
☆	STM32WL55...	STM32WL55C...	Active	3.564		UFQFPN48	256 kBytes	64 kBytes	29	48 MHz
3	STM32WL55JC	STM32WL55J...	Active	3.945	NUCLEO_WL55JC	UFPGA73	256 kBytes	64 kBytes	43	48 MHz
☆	STM32WL55...	STM32WL55U...	NA	NA		WLCSP59	256 kBytes	64 kBytes	22	48 MHz
☆	STM32WLE4...	STM32WLE4...	NA	NA		UFQFPN48	64 kBytes	64 kBytes	29	48 MHz
☆	STM32WLE4...	STM32WLE4...	NA	NA		UFQFPN48	128 kBytes	64 kBytes	29	48 MHz
☆	STM32WLE4...	STM32WLE4...	Active	2.777		UFQFPN48	256 kBytes	64 kBytes	29	48 MHz
☆	STM32WLE4J8	STM32WLE4J...	NA	NA		UFPGA73	64 kBytes	20 kBytes	43	48 MHz
☆	STM32WLE4...	STM32WLE4J...	NA	NA		UFPGA73	128 kBytes	48 kBytes	43	48 MHz
☆	STM32WLE4...	STM32WLE4J...	Active	3.159		UFPGA73	256 kBytes	64 kBytes	43	48 MHz
☆	STM32WLE5...	STM32WLE5...	Active	2.545		UFQFPN48	64 kBytes	20 kBytes	29	48 MHz
☆	STM32WLE5...	STM32WLE5...	Active	2.73		UFQFPN48	128 kBytes	48 kBytes	29	48 MHz
☆	STM32WLE5...	STM32WLE5...	Active	3.054		UFQFPN48	256 kBytes	64 kBytes	29	48 MHz
☆	STM32WI F5.I8	STM32WI F5.I...	Active	2.927		UFPGA73	64 kBytes	20 kBytes	43	48 MHz

芯片列表

从MCU开始一个工程举例 (3/18)

- 选择as Single core project, 点击OK

The screenshot shows the STM32CubeMX software interface. The 'New Project' dialog box is open, asking 'Do you want to create a new project :'. The 'as Single core project ?' option is selected and highlighted with a red box. A red circle with the number '4' is placed over the 'OK' button. The background shows the 'Existing Projects' list and the 'Manage software installations' section.

MCUs Selection	Output	Series	Lines	Mcu	Package	Required Peripherals
✓		STM32WL	STM32WL5x	STM32WL55JC1x	UFPGA73	None



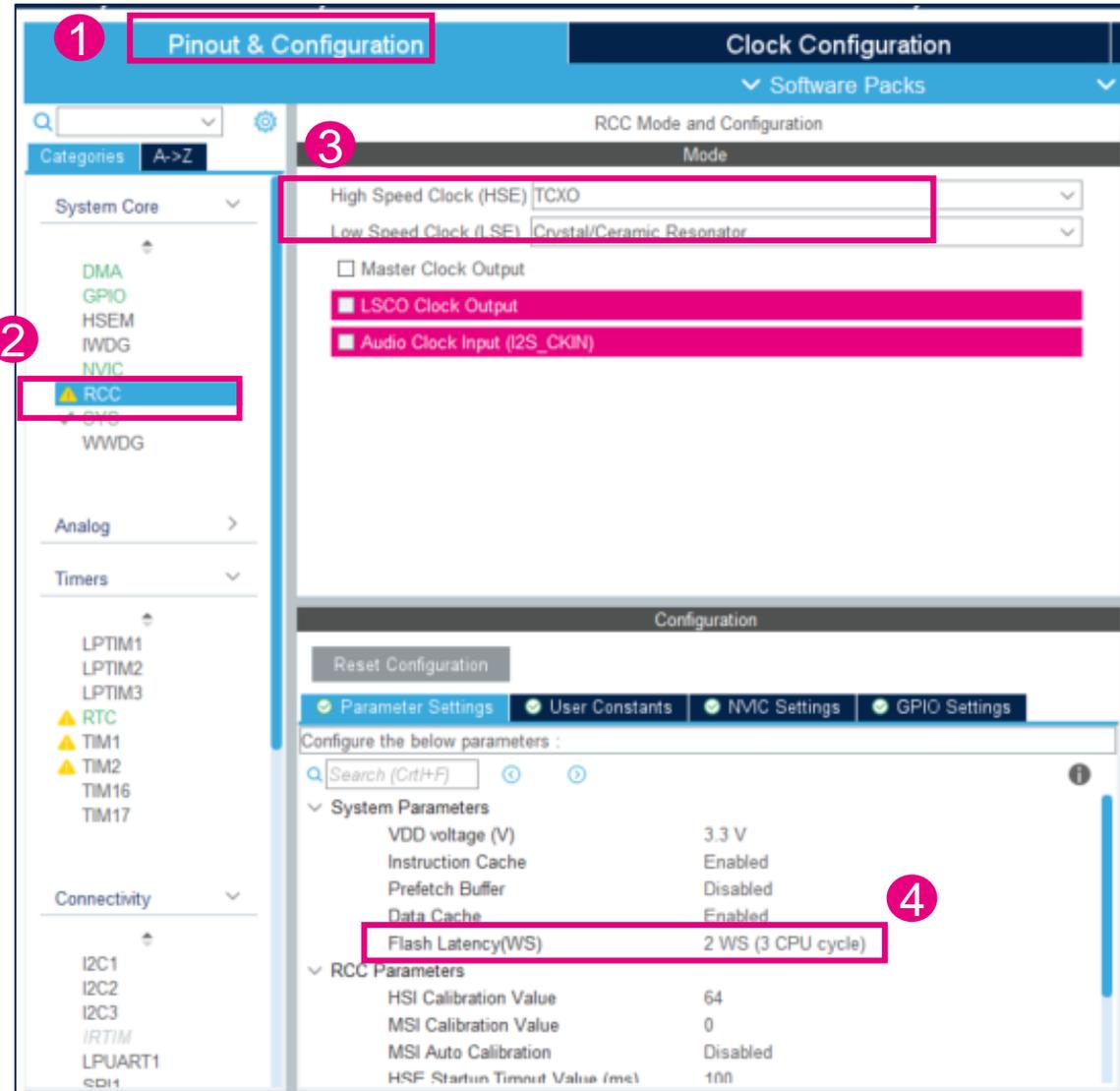
从MCU开始一个工程举例 (4/18)

- 开始外设配置:

- 配置RCC:

HSE:TCXO

LSE:Crystal/Ceramic Resonator

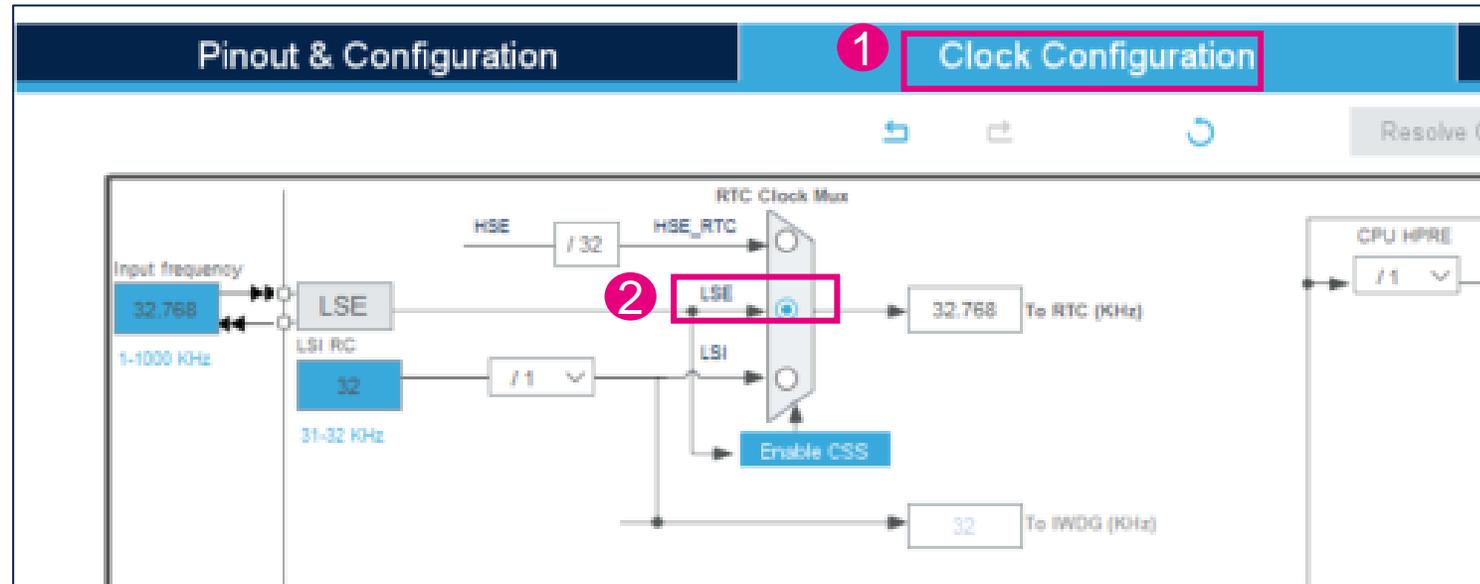


从MCU开始一个工程举例 (5/18)

- 开始外设配置

- 2.RTC配置

- RTC Clock Mux选择LSE



从MCU开始一个工程举例 (6/18)

开始外设配置

2.RTC配置

- RTC_N_PREDIV_S
- RTC_PREDIV_S
- RTC_PREDIV_A
- USART_BAUDRATE
- LPUART_BAUDRATE

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

Parameter Settings | User Constants | NVIC Settings

Search Constants

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
LPUART_BAUDRATE	9600

Parameter Settings | User Constants | NVIC Settings

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

Parameter Settings | User Constants | NVIC Settings

Configure the below parameters :

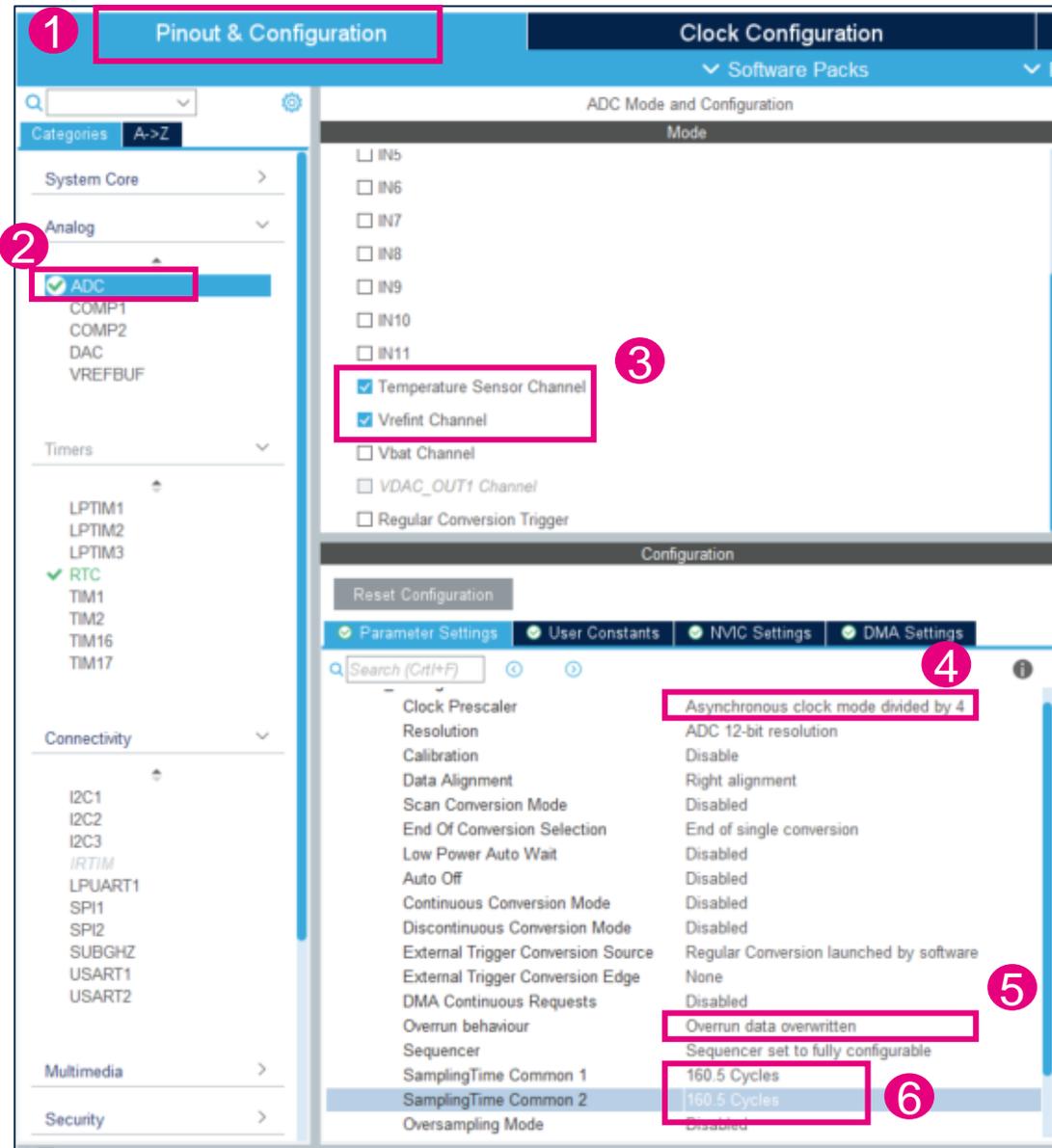
Search (Ctrl+F)

General

Asynchronous Predivider value	RTC_PREDIV_A
Bin Mode	Free running Binary mode
SSRU Underflow interrupt	Enabled

从MCU开始一个工程举例 (7/18)

- 开始外设配置
- 3.使能ADC



从MCU开始一个工程举例 (8/18)

- 开始外设配置
- 4.配置USART2

1 Pinout & Configuration

2 USART2

3 Mode Asynchronous

4 User Constants

5 USART_BAUDRATE 115200

Constant Name	Constant Value
RTC_N_PREDIV_S	10
RTC_PREDIV_S	$((1 \ll \text{RTC_N_PREDIV_S}) - 1)$
RTC_PREDIV_A	$((1 \ll (15 - \text{RTC_N_PREDIV_S})) - 1)$
USART_BAUDRATE	115200

从MCU开始一个工程举例 (9/18)

- 开始外设配置
- 4.配置USART2

The screenshot shows the 'Pinout & Configuration' window with the 'Clock Configuration' tab selected. The 'Software Packs' dropdown is set to 'USART2 Mode and Configuration'. The 'Mode' is set to 'Asynchronous'. The 'Hardware Flow Control (RS232)' and 'Hardware Flow Control (RS485)' are both set to 'Disable'. The 'Slave Select(NSS) Management' is set to 'Disable'. The 'Parameter Settings' tab is selected, showing the following configuration:

- USART BAUDRATE Bits/s: 115200 (highlighted with a red box and number 7)
- Word Length: 8 Bits (including Parity)
- Parity: None
- Stop Bits: 1
- Data Direction: Receive and Transmit
- Over Sampling: 16 Samples
- Single Sample: Disable
- Clock Prescaler: 1
- Fifo Mode: Enable (highlighted with a red box and number 8)
- Txfifo Threshold: 1 eighth full configuration
- Rxfifo Threshold: 1 eighth full configuration

The screenshot shows the 'NVIC Settings' window with the 'Parameter Settings' tab selected. The 'NVIC Interrupt Table' is shown with the following settings:

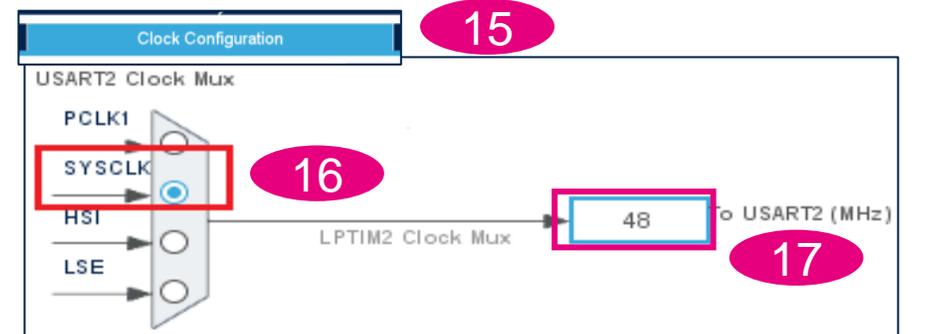
Interrupt	Enabled	Priority	Sub Priority
DMA1 Channel 5 Interrupt	<input checked="" type="checkbox"/>	0	0
USART2 Interrupt	<input checked="" type="checkbox"/>	0	0

The screenshot shows the 'DMA Settings' window with the 'Parameter Settings' tab selected. The 'DMA Request' table is shown with the following settings:

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

The screenshot shows the 'GPIO Settings' window with the 'Parameter Settings' tab selected. The 'Pin Name' table is shown with the following settings:

Pin Name	Signal on Pin	GPIO outp.	GPIO mode	GPIO Pull...	Maximum	Fast Mode	User Label	Modified
PA2	USART2_TX	n/a	Alternate F...	No pull-up ...	Very High	n/a	USART1x_TX	<input checked="" type="checkbox"/>
PA3	USART2_RX	n/a	Alternate F...	No pull-up ...	Very High	n/a	USARTx_RX	<input checked="" type="checkbox"/>



从MCU开始一个工程举例 (10/18)

- 开始外设配置

5.配置BUT,LED, DBG,
RF_CTRL GPIO

The screenshot shows the STM32CubeIDE Pinout & Configuration window for a UFBGA73 microcontroller. The interface is divided into several sections:

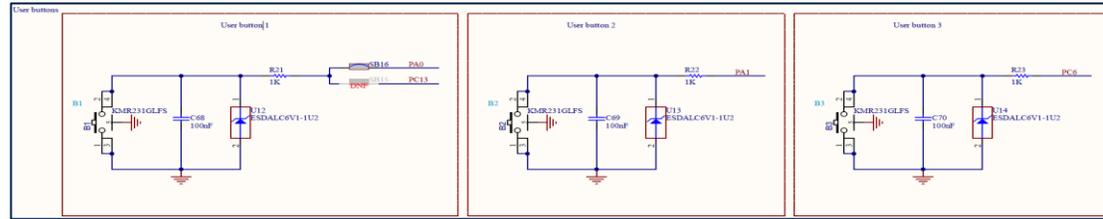
- 1. Tab Selection:** The 'Pinout & Configuration' tab is selected at the top.
- 2. Peripheral Selection:** 'GPIO' is selected in the 'System Core' tree on the left.
- 3. Pinout Diagram:** The 'Pinout view' shows the physical pinout of the UFBGA73 package. A red box highlights the configuration for a specific pin.
- 4. Peripheral Grouping:** The 'Configuration' section shows 'GPIO' selected under 'Group by Peripherals'.
- 5. Signal Selection:** A table lists available signals for each pin. A red box highlights the selection of signals like BUT1, LED1, etc.
- 6. User Label:** The 'User Label' column in the table is highlighted, showing labels like BUT1, LED1, etc.
- 7. NVIC Configuration:** The 'NVIC' peripheral is selected in the top bar.
- 8. Interrupt Configuration:** The 'NVIC Interrupt Table' shows that 'EXTI Line 0 Interrupt' and 'EXTI Line 1 Interrupt' are enabled.

Pin No.	Signal on Pin	GPIO outp.	GPIO mode	GPIO Pull-	Maximum	Fast Mode	User Label	Modified
PA0	n/a	n/a	External Int...	No pull-up ...	n/a	n/	BUT1	✓
PA1	n/a	n/a	External Int...	No pull-up ...	n/a	n/	BUT2	✓
PB9	n/a	Low	Output Pus...	No pull-up ...	Low	Disabl	LED2	✓
PB10	n/a	Low	Output Pus...	No pull-up ...	Low	n/	DBG4	✓
PB11	n/a	Low	Output Pus...	No pull-up ...	Low	n/	LED3	✓
PB12	n/a	Low	Output Pus...	No pull-up ...	Low	n/	DBG1	✓
PB13	n/a	Low	Output Pus...	No pull-up ...	Low	n/	DBG2	✓
PB14	n/a	Low	Output Pus...	No pull-up ...	Low	n/	DBG3	✓
PB15	n/a	Low	Output Pus...	No pull-up ...	Low	n/	LED1	✓
PC3	n/a	Low	Output Pus...	No pull-up ...	Low	n/	RF_CTRL3	✓
PC4	n/a	Low	Output Pus...	No pull-up ...	Low	n/	RF_CTRL1	✓
PC5	n/a	Low	Output Pus...	No pull-up ...	Low	n/	RF_CTRL2	✓
PC6	n/a	n/a	External Int...	No pull-up ...	n/a	n/	BUT3	✓

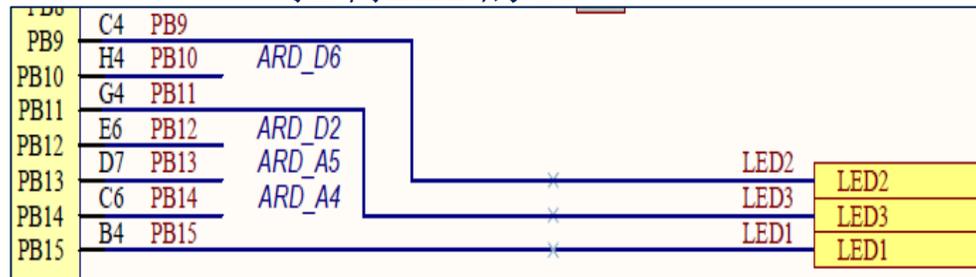
Interrupt	Enabled	Preemption Priority	Sub Priority
EXTI Line 0 Interrupt	✓	0	0
EXTI Line 1 Interrupt	✓	0	0
EXTI Lines [9:5] Interrupt	☐	0	0

从MCU开始一个工程举例 (11/18)

1 按键控制的GPIO



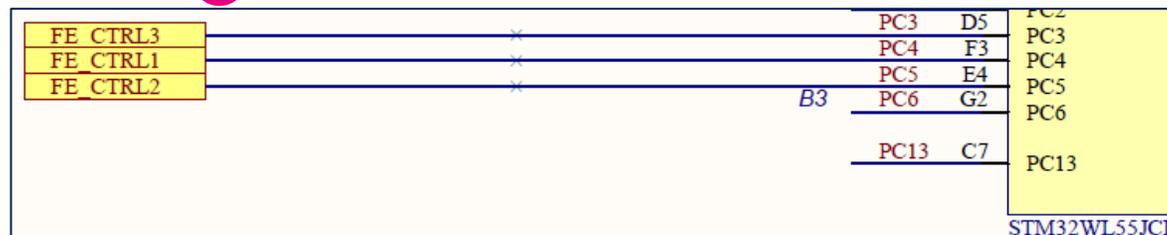
2 控制LED的GPIO



3 DBG 的GPIO



4 RF控制开关的GPIO



开始外设配置

5.配置BUT,LED, DBG, RF_CTRL GPIO.

STM32WL Nucleo 板上对应线路参考。

从MCU开始一个工程举例 (12/18)

- 开始外设配置
6.使能SUBGHZ

1 Pinout & Configuration

2 SUBGHZ

3 Activated

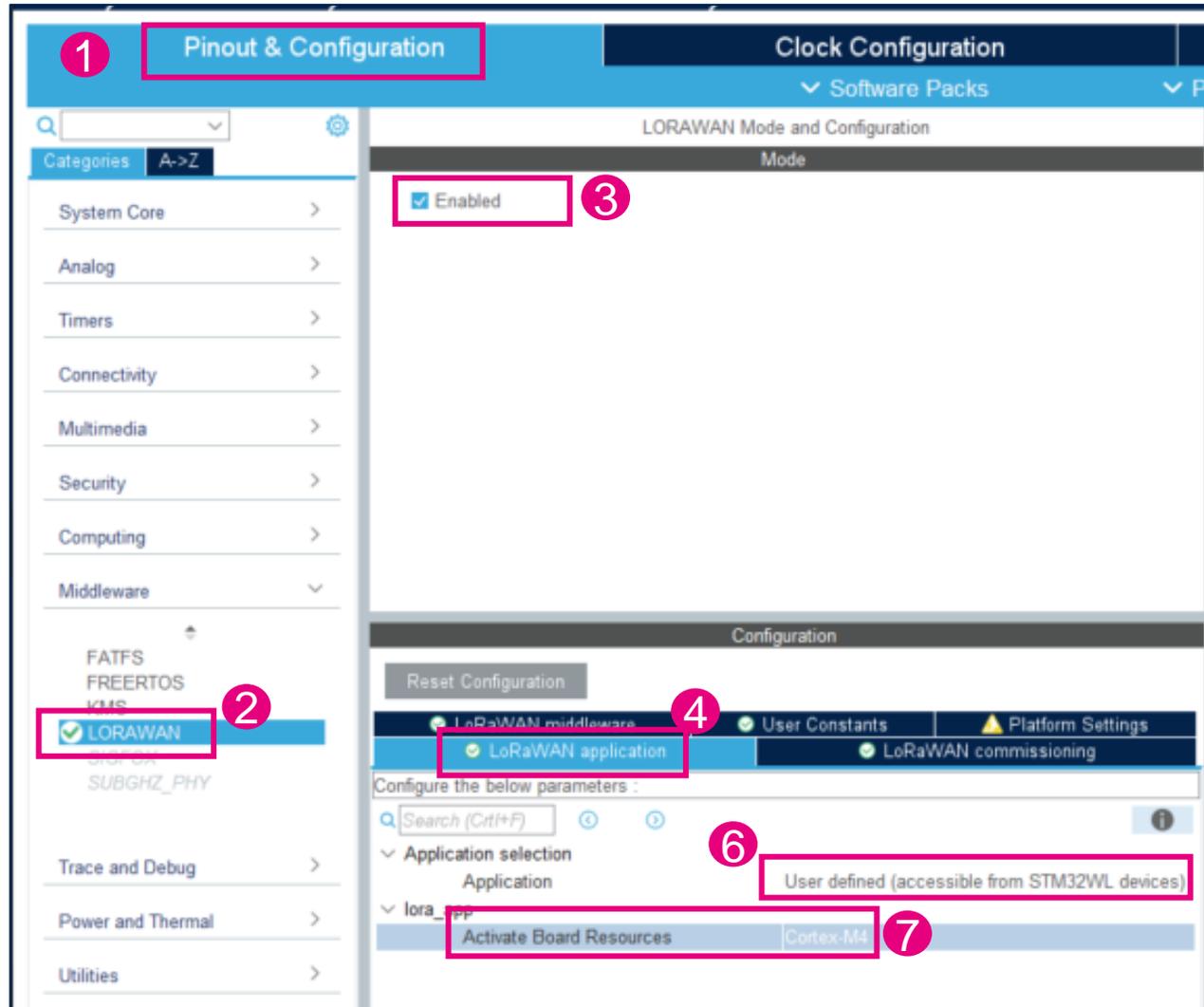
3 NVIC Settings

4 SUBGHZ Radio Interrupt

NVIC Interrupt Table	Enabled	Preemption Priority	Sub Priority
SUBGHZ Radio Interrupt	<input checked="" type="checkbox"/>	0	0

从MCU开始一个工程举例 (13/18)

- 开始外设配置
7.Middleware 选择LoRaWAN



从MCU开始一个工程举例 (14/18)

- 开始外设配置

- 7.Middleware 选择LoRaWAN

- 配置LoRaWAN参数



Configure the below parameters :

Search (Ctrl+F)

Commissioning

- Public network
- Current network ID 0

se-identity

- Static Device EUI
- App/Join EUI 01, 01, 01, 01, 01, 01, 01, 01
- Application key 2B,7E,15,16,28,AE,D2,A6,AB,F7,15,88,09,CF,4F,3C
- Network key 2B,7E,15,16,28,AE,D2,A6,AB,F7,15,88,09,CF,4F,3C
- Static Device Address
- Network session key 2B,7E,15,16,28,AE,D2,A6,AB,F7,15,88,09,CF,4F,3C
- Application session key 2B,7E,15,16,28,AE,D2,A6,AB,F7,15,88,09,CF,4F,3C

LoRaWAN commissioning LoRaWAN middleware User Constants Platform Settings

LoRaWAN application

Configure the below parameters :

Search (Ctrl+F)

Region(s) selection please select the desired region(s) in the list below

- Region Asia freq: 923
- Region Australia freq: 915
- Region China freq: 470
- Region China freq: 779
- Region Europe freq: 433
- Region Europe freq: 868
- Region Korea freq: 920
- Region India freq: 865
- Region USA freq: 915
- Region Russia freq: 864
- Enable Hybrid mode
- Enable LoRaMAC ClassB

radio_board_if

- Radio maximum wakeup time (in ms) 10
- TCXO support
- DCDC support
- Activate Radio Board Interface
- Activate Debug Line

mw_log_conf

- Enable Middleware log

LoRaWAN application LoRaWAN commissioning LoRaWAN middleware User Constants Platform Settings

Platform proposal

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

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	Undefined	Unknown
Debug Line 2	GPIO Output	Undefined	Unknown
Debug Line 3	GPIO Output	Undefined	Unknown
Debug Line 4	GPIO Output	Undefined	Unknown



从MCU开始一个工程举例 (15/18)

- 完成配置
8. 工程选项

The screenshot shows the STM32CubeMX Project Manager interface for a project named "LoRaWAN_End_node demo". The interface is divided into several sections, with numbered callouts (1-7) highlighting specific configuration options:

- 1**: Project Manager tab
- 2**: Project section
- 3**: Project Name field (LoRaWAN_End_node demo)
- 4**: Toolchain Folder Location field (C:\Work\STM32WL\STM32WL Project\STM32WL CubeMx project\LoRaWAN_End_node demo\)
- 5**: Toolchain / IDE (EWARM) and Min Version (V8) dropdowns
- 6**: Linker Settings (Minimum Heap Size: 0x200, Minimum Stack Size: 0x800)
- 7**: Firmware Package Name and Version field (STM32Cube FW_WL V1.0.0)

从MCU开始一个工程举例 (16/18)

- 完成配置
8. 工程选项

Home > STM32WL55JClx > LoRaWAN_End_node demo.ioc - Project Manager

Pinout & Configuration | Clock Configuration | Project Manager

Project

Code Generator

Advanced Settings

STM32Cube MCU packages and embedded software packs

- Copy all used libraries into the project folder
- Copy only the necessary library files
- Add necessary library files as reference in the toolchain project configuration file

Generated files

- Generate peripheral initialization as a pair of '.c/.h' files per peripheral
- Backup previously generated files when re-generating
- Keep User Code when re-generating
- Delete previously generated files when not re-generated

HAL Settings

- Set all free pins as analog (to optimize the power consumption)
- Enable Full Assert

Template Settings

Select a template to generate customized code

Settings...

从MCU开始一个工程举例 (17/18)

- 生成代码
9. 点击右上角
GENERATE CODE

The screenshot shows the STM32CubeMX Project Manager interface for a project named 'LoRaWAN_End_node demo.ioc'. The 'GENERATE CODE' button in the top right corner is highlighted with a red circle and the number 1. The interface is divided into several sections:

- Project:** Contains options for STM32Cube MCU packages and embedded software packs:
 - Copy all used libraries into the project folder
 - Copy only the necessary library files
 - Add necessary library files as reference in the toolchain project configuration file
- Code Generator:** Contains options for generated files:
 - Generate peripheral initialization as a pair of '.c/.h' files per peripheral
 - Backup previously generated files when re-generating
 - Keep User Code when re-generating
 - Delete previously generated files when not re-generating
- Advanced Settings:** Contains HAL and Template settings:
 - HAL Settings:**
 - Set all free pins as analog (to optimize the power consumption)
 - Enable Full Assert
 - Template Settings:**
 - Select a template to generate customized code
 - Settings...

A progress bar is visible in the Code Generator section, labeled 'Generating user source code...'.

从MCU开始一个工程举例 (18/18)

- 完成应用代码
 - MX_LoRaWAN_Init()
 - MX_LoRaWAN_Process()
- etc

Project - IAR Embedded Workbench IDE - Arm 8.50.9

File Edit View Project ST-Link Tools Window Help

Workspace

LoRaWAN_End_node demo

Files

- LoRaWAN_End_node demo - LoRa...
- Application
 - EWARM
 - startup_stm32wl55xx_cm4.s
 - User
- Drivers
 - CMSIS
 - STM32WLxx_HAL_Driver
- Middlewares
 - LoRaWAN
 - SubGHz_Phy
- Utilities
- Output

main.c x app_lorawan.c lora_app.c

```
main()
84  /* USER CODE END SysInit */
85
86  /* Initialize all configured peripherals */
87  MX_LoRaWAN_Init();
88  /* USER CODE BEGIN 2 */
89
90  /* USER CODE END 2 */
91
92  /* Infinite loop */
93  /* USER CODE BEGIN WHILE */
94  while (1)
95  {
96    /* USER CODE END WHILE */
97    MX_LoRaWAN_Process();
98
99    /* USER CODE BEGIN 3 */
100 }
101 /* USER CODE END 3 */
102 }
103
104 /**
```

Build

Messages

Converting

Total number of errors: 0

Total number of warnings: 1



life.augmented

从Example开始一个工程



从Example开始一个工程(1/8)

The screenshot displays the STM32CubeMX software interface. The window title is "MX STM32CubeMX Untitled". The menu bar includes "File", "Window", and "Help". The interface is divided into three main sections:

- Existing Projects:** Lists "Recent Opened Projects" and "Other Projects". Under "Recent Opened Projects", there are four entries, each with a file name, a last modified date, and an "MX" icon. A yellow box highlights the text "从选择例程开始" (Start from selecting an example) over the "LoRaWAN_End_Node.ioc" entry.
- New Project:** A dark blue panel titled "I need to :" contains three options, each with a blue button: "Start My project from MCU" (ACCESS TO MCU SELECTOR), "Start My project from ST Board" (ACCESS TO BOARD SELECTOR), and "Start My project from Example" (ACCESS TO EXAMPLE SELECTOR). The "ACCESS TO EXAMPLE SELECTOR" button is highlighted with a red border.
- Manage software installations:** Contains two sections: "Check for STM32CubeMX and embedded software package..." with a "CHECK FOR UPDATES" button, and "Install or remove embedded software packages" with an "INSTALL / REMOVE" button.

At the bottom right, there is a promotional banner for "Build your certified safety system with STM32 and STM8" featuring shields for "SIL Ready", "ASIL Ready", and "ClassB Ready", along with the "Partner Program" logo and the ST logo.

从Example开始一个工程(2/8)

MX New Project from Example

MCU/MPU Selector Board Selector Example Selector Cross Selector

Example Filters

Name

Keyword Aa [ab]

Vendor

Board **1**

Name

Type

Discovery Kit

Evaluation Board

Nucleo-144

Nucleo-32

Nucleo-64

MCU / MPU

Project

STM32CubeMX Compatible

Toolchain / IDE

Project Type

Based on Driver

Middleware

MCU/MPU Library

Features

Start Project

Projects/NUCLEO-WL55JC1/Applications/LoRaWAN/LoRaWAN_End_Node/

STM32WL **LoRaWAN_End_Node** V1.0.0

Required Software Package: STM32Cube_FW_WL_V1.0.0 (size: 152.0 MB) ✓

Vendor: STMicroelectronics

Board: NUCLEO-WL55JC1

Mounted device: STM32WL55JC1x UFPGA73

Supported Toolchain/IDE: EWARM, MDK-ARM, STM32CubeIDE 6.1.0 ✓

Keywords: Applications, End_Node, LoRaWAN, SingleCore, SubGHz, Phy

This directory contains a set of source files that implements a LoRa application device sending sensors data to LoRa Network server.

Examples List: 250 items

Name	Board	Board Type	Series	Project Type	Driver
KMS_Embedded_AES_Keys	NUCLEO-WL55JC1	Nucleo-64	STM32WL	Application	HAL
KMS_Embedded_RSA_Key	NUCLEO-WL55JC1	Nucleo-64	STM32WL	Application	HAL
LocalNetwork_Concentrator	NUCLEO-WL55JC1	Nucleo-64	STM32WL	Demonstration	HAL
LocalNetwork_Sensor	NUCLEO-WL55JC1	Nucleo-64	STM32WL	Demonstration	HAL
LoRaWAN_AT_Slave	NUCLEO-WL55JC1	Nucleo-64	STM32WL	Application	HAL
LoRaWAN_AT_Slave_DualCore	NUCLEO-WL55JC1	Nucleo-64	STM32WL	Application	HAL
LoRaWAN_End_Node	NUCLEO-WL55JC1	Nucleo-64	STM32WL	Application	HAL
LoRaWAN_End_Node_DualCore	NUCLEO-WL55JC1	Nucleo-64	STM32WL	Application	HAL
LoRaWAN_End_Node_DualCore	NUCLEO-WL55JC1	Nucleo-64	STM32WL	Application	HAL
LoRaWAN_End_Node_DualCoreFreeRTOS	NUCLEO-WL55JC1	Nucleo-64	STM32WL	Application	HAL
LPTIM_PulseCounter	NUCLEO-WL55JC1	Nucleo-64	STM32WL	Example	HAL
LPTIM_PulseCounter_Init	NUCLEO-WL55JC1	Nucleo-64	STM32WL	Example	LL

• 根据demo板选择合适的例程

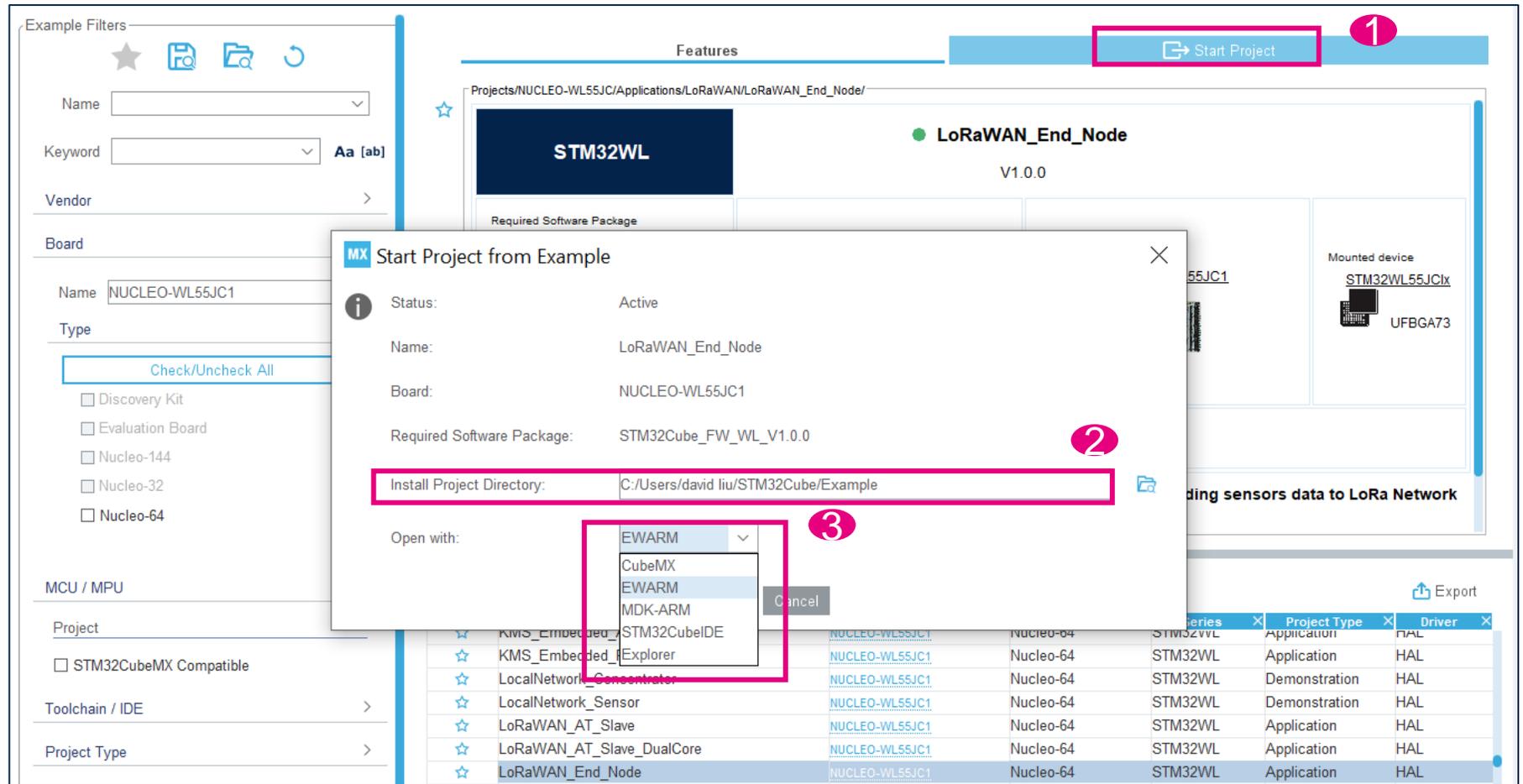
• 这里选择NUCLEOWL55JC1

LoRaWAN_End_Node



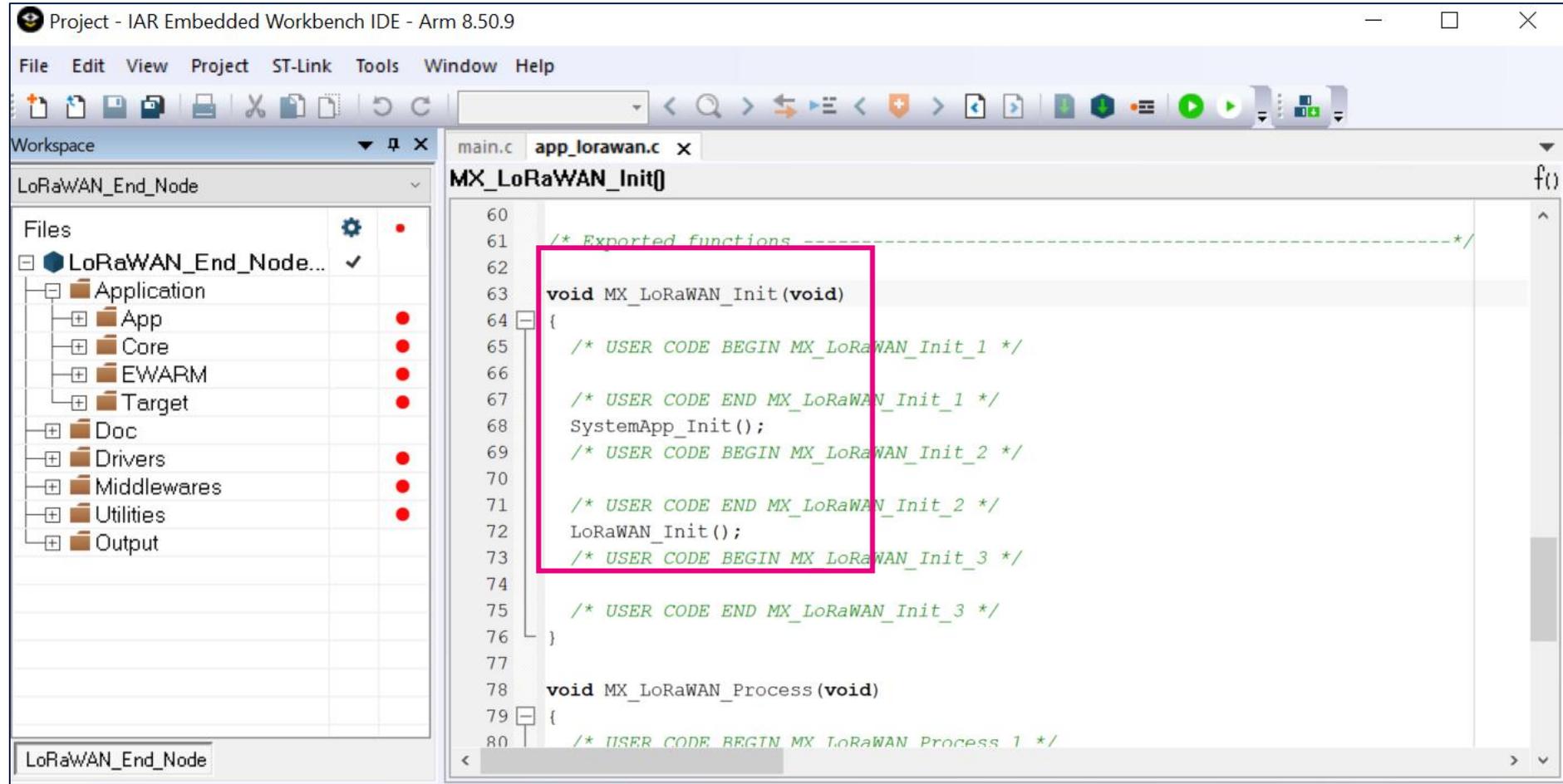
从Example开始一个工程(3/8)

- 点击右上角Start Project
- 选择路径以及合适的编译器, 比如选择IAR



从Example开始一个工程(4/8)

- 点击OK, 后会自动打开Example的IAR工程

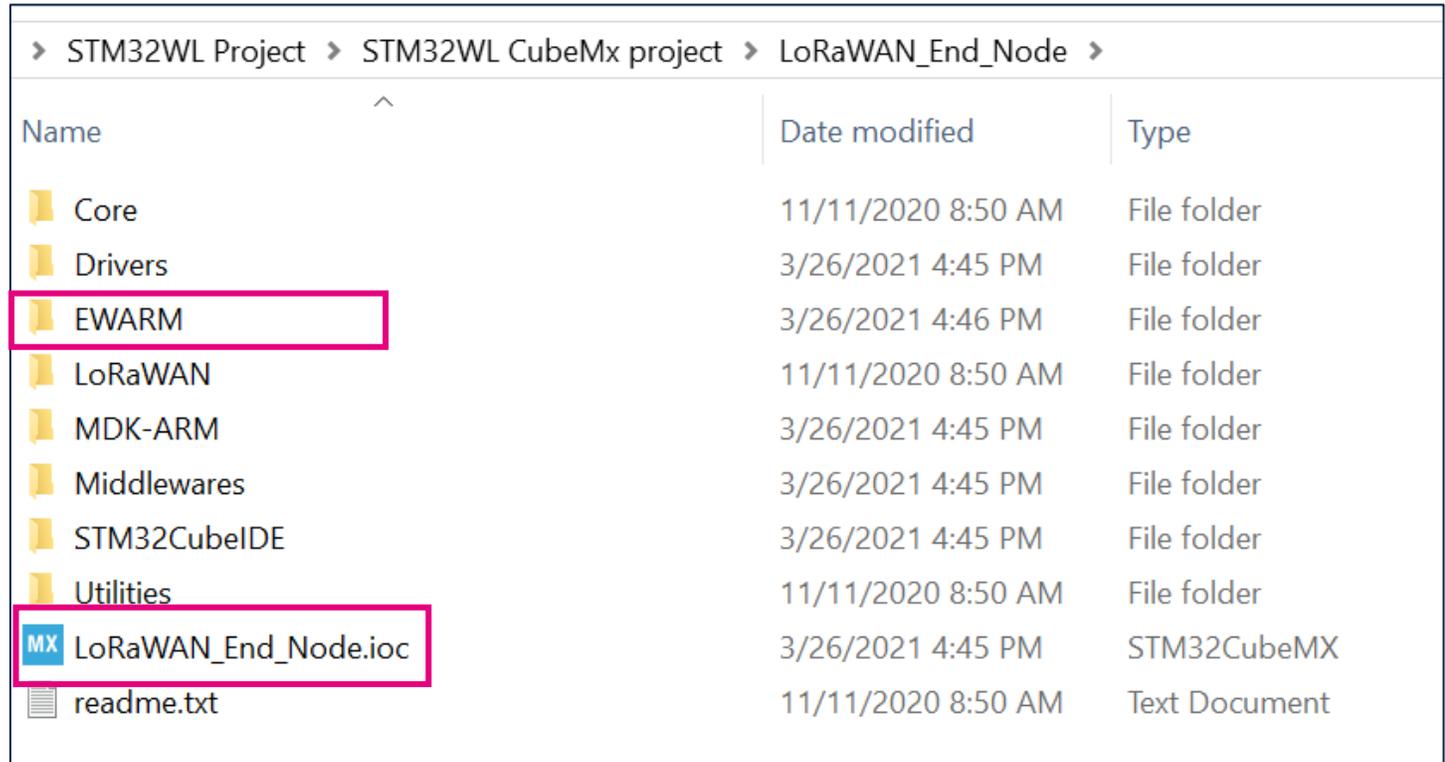


The screenshot shows the IAR Embedded Workbench IDE interface. The title bar reads "Project - IAR Embedded Workbench IDE - Arm 8.50.9". The menu bar includes "File", "Edit", "View", "Project", "ST-Link", "Tools", "Window", and "Help". The toolbar contains various icons for file operations and execution. The workspace is divided into two main panes. On the left, the "Files" pane shows a project tree for "LoRaWAN_End_Node" with folders like "Application", "App", "Core", "EWARM", "Target", "Doc", "Drivers", "Middlewares", "Utilities", and "Output". On the right, the "Code" pane displays the source code for "MX_LoRaWAN_Init()". The code is as follows:

```
60
61  /* Exported functions -----*/
62
63  void MX_LoRaWAN_Init(void)
64  {
65      /* USER CODE BEGIN MX_LoRaWAN_Init_1 */
66
67      /* USER CODE END MX_LoRaWAN_Init_1 */
68      SystemApp_Init();
69      /* USER CODE BEGIN MX_LoRaWAN_Init_2 */
70
71      /* USER CODE END MX_LoRaWAN_Init_2 */
72      LoRaWAN_Init();
73      /* USER CODE BEGIN MX_LoRaWAN_Init_3 */
74
75      /* USER CODE END MX_LoRaWAN_Init_3 */
76  }
77
78  void MX_LoRaWAN_Process(void)
79  {
80      /* USER CODE BEGIN MX_LoRaWAN_Process_1 */
```

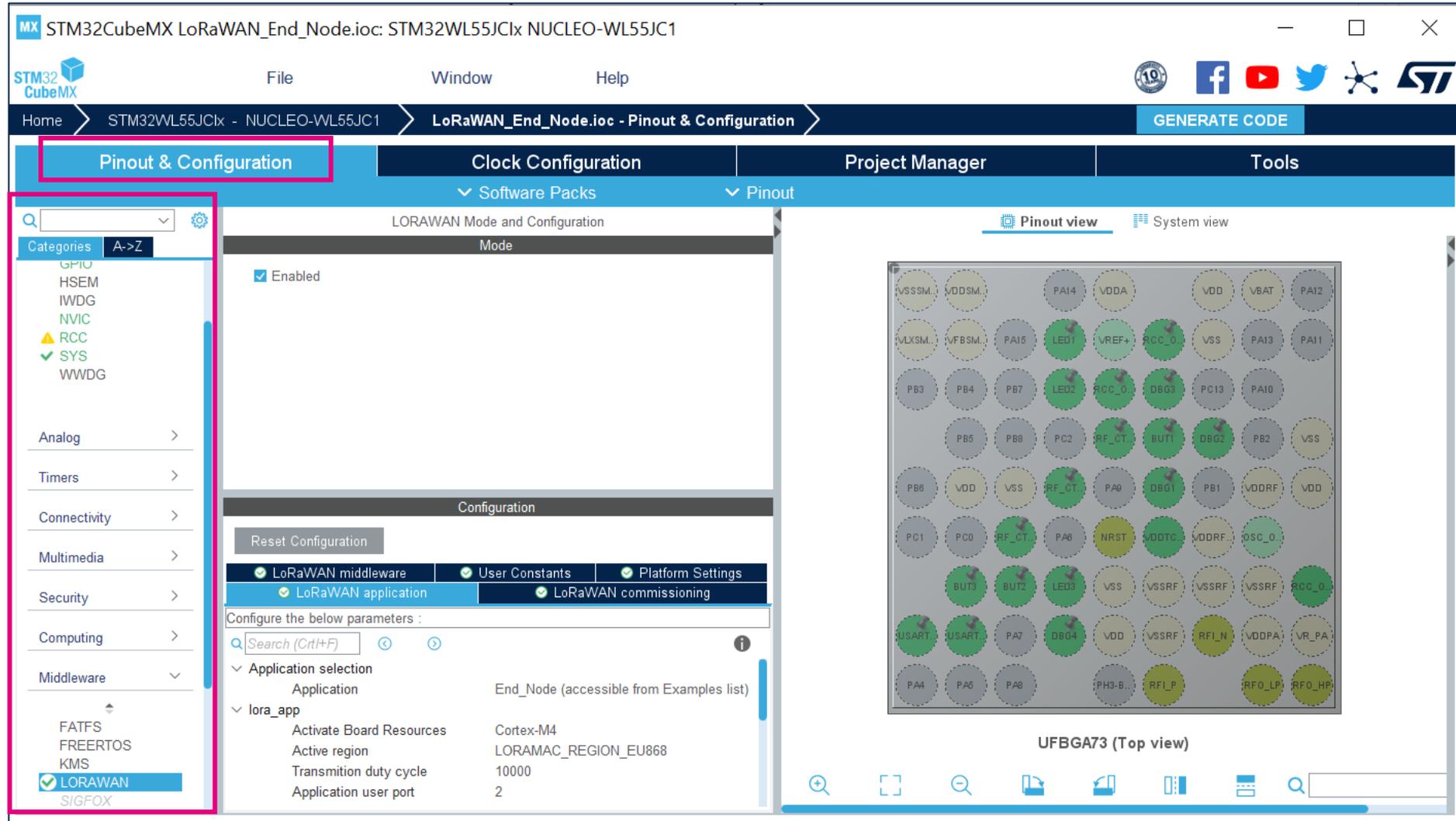
从Example开始一个工程(5/8)

- 在安装文件路径.ioc的STM32CubeMX应用文件



Name	Date modified	Type
Core	11/11/2020 8:50 AM	File folder
Drivers	3/26/2021 4:45 PM	File folder
EWARM	3/26/2021 4:46 PM	File folder
LoRaWAN	11/11/2020 8:50 AM	File folder
MDK-ARM	3/26/2021 4:45 PM	File folder
Middlewares	3/26/2021 4:45 PM	File folder
STM32CubeIDE	3/26/2021 4:45 PM	File folder
Utilities	11/11/2020 8:50 AM	File folder
MX LoRaWAN_End_Node.ioc	3/26/2021 4:45 PM	STM32CubeMX
readme.txt	11/11/2020 8:50 AM	Text Document

从Example开始一个工程(6/8)



- 点击打开.ioc文件
- 客户可以在example的基础上增删外设，或添加应用

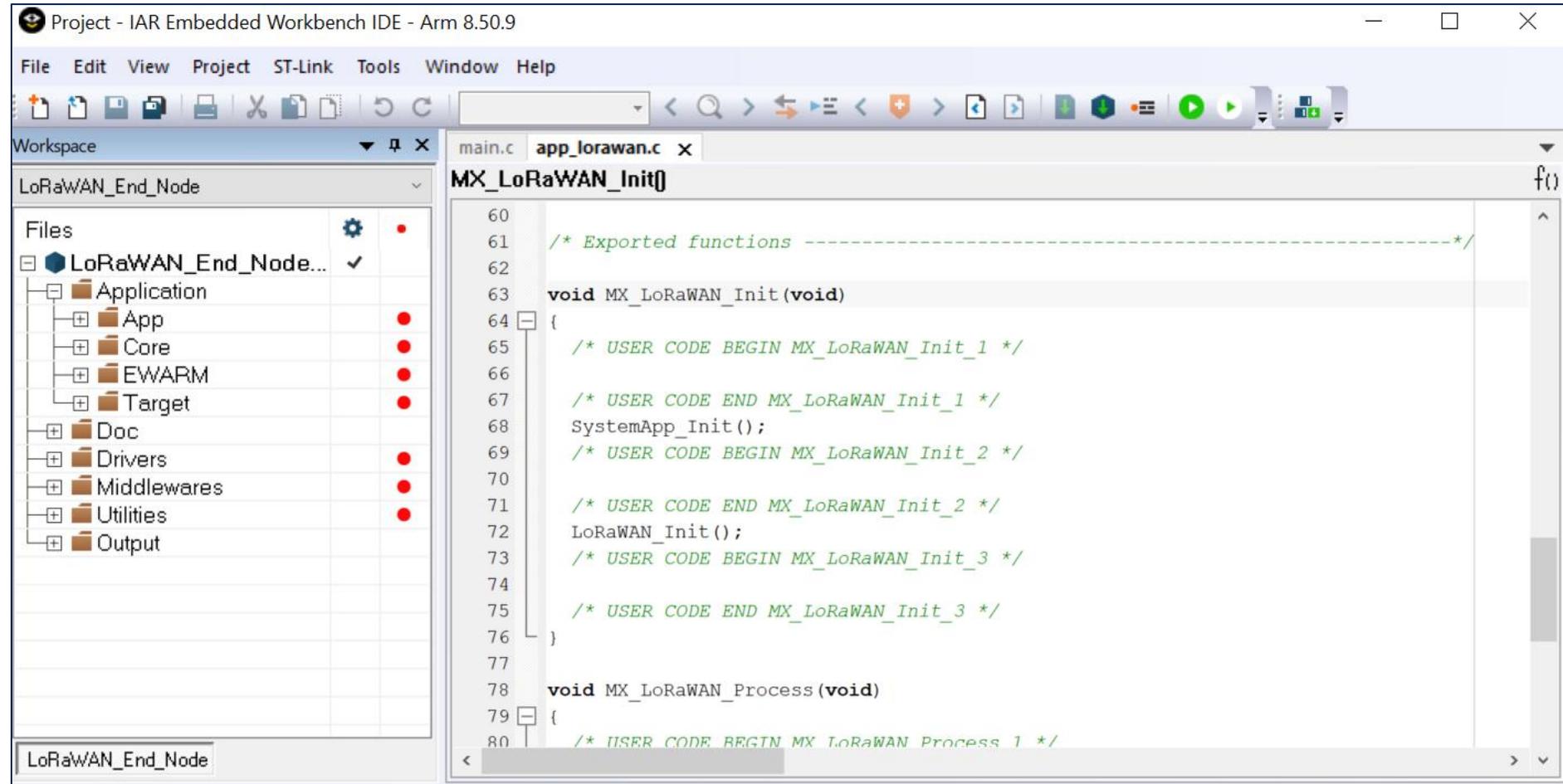
从Example开始一个工程(7/8)

- 生成代码
点击右上角
GENERATE CODE

The screenshot displays the STM32CubeMX Project Manager interface for a project named 'LoRaWAN_End_node demo.ioc'. The top navigation bar includes 'Home', 'STM32WL55JClx', and 'LoRaWAN_End_node demo.ioc - Project Manager'. A 'GENERATE CODE' button is highlighted in the top right corner. The main interface is divided into four tabs: 'Pinout & Configuration', 'Clock Configuration', 'Project Manager', and 'Tools'. The 'Project Manager' tab is active, showing settings for 'STM32Cube MCU packages and embedded software packs', 'Generated files', 'HAL Settings', and 'Template Settings'. A progress bar indicates 'Generating user source code...'. The 'Project' section has three radio button options: 'Copy all used libraries into the project folder', 'Copy only the necessary library files', and 'Add necessary library files as reference in the toolchain project configuration file' (selected). The 'Generated files' section has three checked checkboxes: 'Generate peripheral initialization as a pair of '.c/.h' files per peripheral', 'Keep User Code when re-generating', and 'Delete previously generated files when not re-generate'. The 'HAL Settings' section has two unchecked checkboxes: 'Set all free pins as analog (to optimize the power consumption)' and 'Enable Full Assert'. The 'Template Settings' section has a 'Settings...' button.

从Example开始一个工程(8/8)

- 完成应用代码



The screenshot displays the IAR Embedded Workbench IDE interface. The title bar reads "Project - IAR Embedded Workbench IDE - Arm 8.50.9". The menu bar includes "File", "Edit", "View", "Project", "ST-Link", "Tools", "Window", and "Help". The toolbar contains various icons for file operations and execution. The workspace is divided into two main panes. On the left, the "Files" pane shows a project tree for "LoRaWAN_End_Node" with subfolders: Application, App, Core, EWARM, Target, Doc, Drivers, Middlewares, Utilities, and Output. On the right, the "Code" pane shows the source code for "app_lorawan.c". The code defines two functions: "MX_LoRaWAN_Init()" and "MX_LoRaWAN_Process()". The "MX_LoRaWAN_Init()" function includes user code markers and calls "SystemApp_Init()", "LoRaWAN_Init()", and "MX_LoRaWAN_Process()". The "MX_LoRaWAN_Process()" function also includes a user code marker.

```
60
61  /* Exported functions -----*/
62
63 void MX_LoRaWAN_Init(void)
64 {
65     /* USER CODE BEGIN MX_LoRaWAN_Init_1 */
66
67     /* USER CODE END MX_LoRaWAN_Init_1 */
68     SystemApp_Init();
69     /* USER CODE BEGIN MX_LoRaWAN_Init_2 */
70
71     /* USER CODE END MX_LoRaWAN_Init_2 */
72     LoRaWAN_Init();
73     /* USER CODE BEGIN MX_LoRaWAN_Init_3 */
74
75     /* USER CODE END MX_LoRaWAN_Init_3 */
76 }
77
78 void MX_LoRaWAN_Process(void)
79 {
80     /* USER CODE BEGIN MX_LoRaWAN_Process_1 */
```



life.augmented

总结



1. STM32CubeMX介绍

2. 使用CubeMX创建LoRaWAN节点

- 从MCU开始一个工程
- 从Example开始一个工程

Help	
Help	F1
About	Alt-A
Docs & Resources	Alt-D
Tutorial Videos	Alt-V
Refresh Data	Alt-R
User Preferences	
Check for Updates	Alt-C
Manage embedded software packages	Alt-U
Updater Settings ...	Alt-S



Tutorial Videos | 选择www.stm32.com.cn

Video server: www.stm32.com.cn

Suggested videos

- Clock tree configuration overview**
The video explains how the user can adjust and configure the system and peripheral clocks. STM32CubeMX Clock Configuration window provides a schematic overview of the ...
Published on 31 Jul 2020 Duration: 0:04:20

Other videos

- Clock conflict resolution**
The video explains the clock conflict resolution mechanism
Published on 31 Jul 2020 Duration: 0:02:50
- STM32CubeMX command lines**
The video shows how to install/uninstall STM32CubeMX from command line and how to run it in interactive command-line mode.
Published on 31 Jul 2020 Duration: 0:05:06
- NVIC configuration and code generation**
NVIC configuration consists in selecting a priority group, enabling/disabling interrupts, and configuring interrupt priority levels. Use the "Code Generation" view ...
Published on 31 Jul 2020 Duration: 0:04:00
- Code generator view**

用户手册/应用笔记

用户手册/应用笔记

UM1718	STM32CubeMX for STM32 configuration and initialization C code generation 《使用STM32CubeMX对STM32进行配置以及生成初始化代码》
UM2739	How to create a software pack enhanced for STM32CubeMX using STM32 Pack Creator tool 《如何使用STM32Pack Creator工具生成STM32CubeMX的插件包》
AN5418	How to build a simple USB-PD sink application with STM32CubeMX 《使用STM32CubeMX构建一个简单的USB-PD接收器应用程序》
AN5426	Migrating graphics middleware projects from STM32CubeMX 5.4.0 to STM32CubeMX 5.5.0 《将图形中间组件从STM32CubeMX5.4.0移植到STM32CubeMX5.5.0》



Thank you

© STMicroelectronics - All rights reserved.

ST logo is a trademark or a registered trademark of STMicroelectronics International NV or its affiliates in the EU and/or other countries.

For additional information about ST trademarks, please refer to www.st.com/trademarks.

All other product or service names are the property of their respective owners.



life.augmented