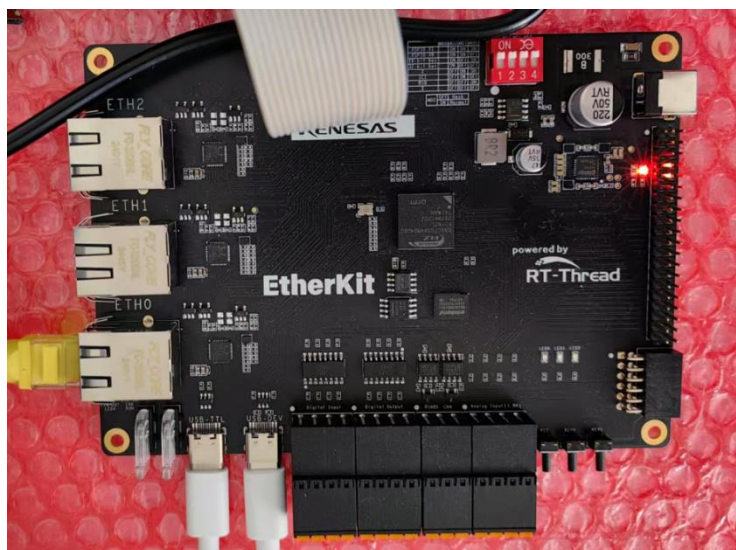


## RZN2L GPT 例程操作手册-----基于 Etherkit 开发板

### 简介

本应用笔记介绍了基于 RZ/N2 Etherkit 开发板的定时器 WDT 的操作。分别介绍 IDE IAR 和 E2studio 软件下的操作。



#### 开发工具

- IDE: IAR EW for Arm 9.50.2  
E2studio 2024-01.1
- FSP: RZ/N2 FSP V2.0
- 仿真器: Jlink V12

#### 实验材料

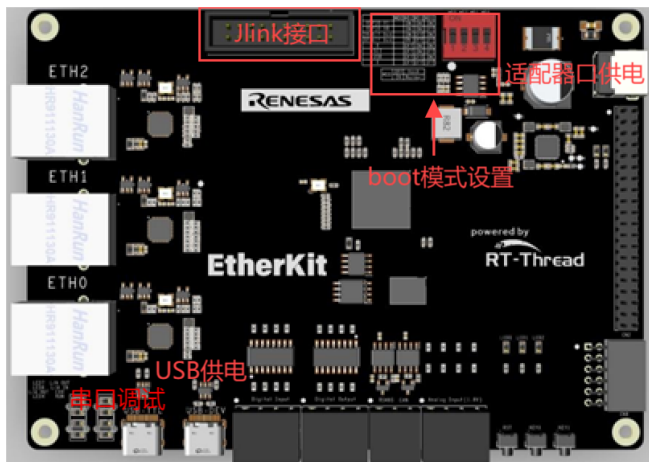
- Etherkit 开发板
- Jlink 仿真器, 需支持瑞萨 R52 内核

## 实验部分

1.硬件设置及软件安装 .....	2
2 .IAR 环境工程介绍.....	3
3 .E2studio 环境工程介绍 .....	8

## 1 .硬件设置及软件安装

本节 EtherKit 开发板硬件设置。

1.1	<div>开发板设置：</div> <div><div><div>● 供电：可选 USB 供电或适配器供电</div><div>● Boot 模式设置：推荐 xSPI0 x1 boot mode</div><div>● Jlink v12</div></div></div> <div></div>
1.2	<div>硬件原理图：</div> <div>本实验用到 LED1,LED2，对应 P140 P141 引脚。</div> <div><div><div>USER LED</div><div><div><div>[8] LED0</div><div>[8] LED1</div><div>[8] LED2</div></div><div><div>&gt;&gt;</div><div>&gt;&gt;</div><div>&gt;&gt;</div></div><div><div>R197_2k</div><div>R199_10k</div><div>R200_12k</div></div><div><div><div>D26</div><div>D28</div><div>D29</div></div><div><div>LED_RED</div><div>LED_BLUE</div><div>LED_GREEN</div></div></div><div><div>VCC_3V3</div></div></div></div></div>

本节完

## 2 .IAR 环境工程介绍

---

本节介绍 IAR 环境下 GPT 工程介绍。

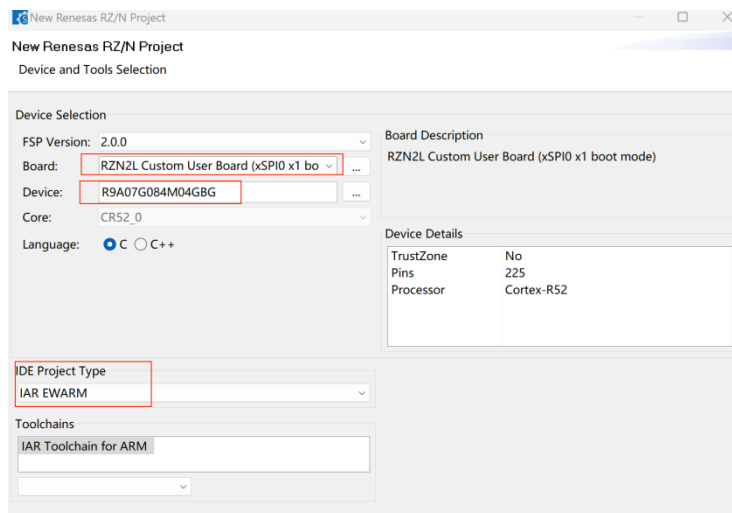
## 2.1

### ● 打开 FSP 新建工程：WDT

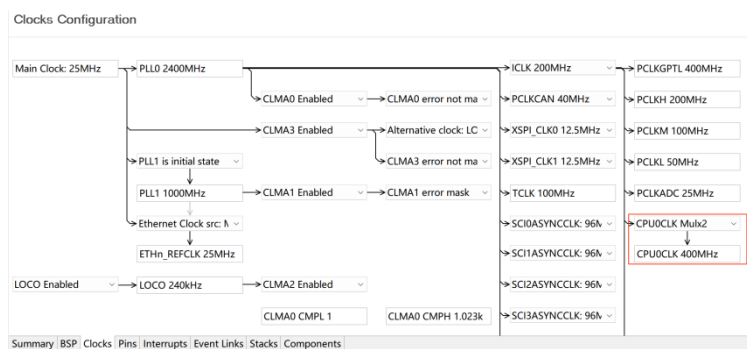
#### 1. Boot 模式选择 RZN2L Custom User Board ( XSPI0 x1 boot mode )

芯片型号：R9A07G084M04GBG

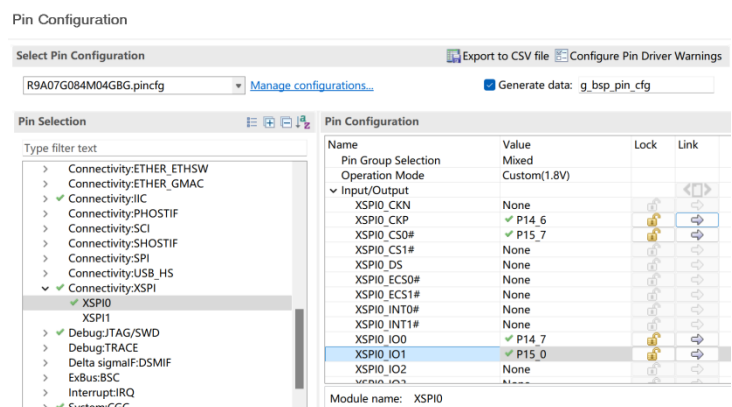
IDE Project Type 选择 IAR EWARM



#### 1. 主时钟设置 400MHZ



#### 2. 配置 XSPI0



#### 3. P140 P141 引脚配置

设置输出模式，初始电平 0

Select Pin Configuration Export to CSV file Configure Pin Driver Warnings

R9A07G084M04GBG.pincfg [Manage configurations...](#) ☒ Generate data: g\_bsp\_pin\_cfg

Pin Selection
Type filter text

- ✓ P14
  - ✓ P14\_0
  - ✓ P14\_1
  - P14\_2
  - ✓ P14\_3
  - P14\_4
  - P14\_5
  - ✓ P14\_6
  - ✓ P14\_7
- > P15
- > P16
- > P17
- > P18

Pin Configuration

Name	Value	Link
Comment		
Mode	Output mode (Low & Not Into Input)	
Pull up/down	None	
Output Type	CMOS	
Drive Capacity	Low	
Region	Safety	
Schmitt Trigger	None	
Slew Rate	Slow	
Input/Output		
P14_0	GPIO	

Module name: P14\_0  
Port Capabilities: ETHER\_ESC: ESC\_SYNC0  
ETHER\_ESC: ESC\_SYNC1

Pin Function Pin Number

Summary BSP Clocks Pins Interrupts Event Links Stacks Components

#### 4. 添加 new Stack: WDT0

- New Stack--Driver--Monitoring---Watchdog (r\_wdt)

Stacks Configuration

Threads
New Thread Remove

- HAL/Common
  - g\_ioport I/O Port (r\_ioport)
  - Memory config check
  - g\_wdt0 Watchdog (r\_wdt)

HAL/Common Stacks

- g\_ioport I/O Port (r\_ioport)
- Memory config check
- g\_wdt0 Watchdog (r\_wdt)

Objects New Object Remove

Summary BSP Clocks Pins Interrupts Event Links Stacks Components

Properties Problems

g\_wdt0 Watchdog (r\_wdt)

Property	Value
Common	
Parameter Checking	Default (BSP)
Multiplex Interrupt	Disabled
Module g_wdt0 Watchdog (r_wdt)	
Name	g_wdt0
Timeout	16,384 Cycles
Clock Division Ratio	PCLK/8192
Window Start Position	100 (Window Position Not Specified)
Window End Position	0 (Window Position Not Specified)
WDT Callback	NULL

#### 5. 添加 new stack : ERROR

- New Stack--System--ERROR(r\_icu\_error)

Stacks Configuration

Threads
New Thread Remove

- HAL/Common
  - g\_ioport I/O Port (r\_ioport)
  - Memory config check
  - g\_wdt0 Watchdog (r\_wdt)

HAL/Common Stacks
New Stack >

- g\_ioport I/O Port (r\_ioport)
- Memory config check
- g\_wdt0 Watchdog (r\_wdt)
- g\_error0 ERROR (r\_icu\_error)

Objects New Object Remove

Summary BSP Clocks Pins Interrupts Event Links Stacks Components

Properties Problems

g\_error0 ERROR (r\_icu\_error)

Property	Value
Common	
Parameter Checking	Default (BSP)
Multiplex Interrupt	Disabled
Module g_error0 ERROR (r_icu_error)	
General	
Interrupts	
Error Events	
Peripheral Error Handler	

- 看门狗复位使能:

## Stacks Configuration

Threads

- HAL/Common
  - g\_ioport I/O Port (r\_ioport)
  - Memory config check
  - g\_wdt0 Watchdog (r\_wdt)

Objects

HAL/Common Stacks

- g\_ioport I/O Port (r\_ioport)
- Memory config check
- g\_wdt0 Watchdog (r\_wdt)
- g\_error0 ERROR (r\_icu\_error)

Summary BSP Clocks Pins Interrupts Event Links **Stacks** Components

Properties Problems

g\_error0 ERROR (r\_icu\_error)

Settings	Property	Value
Module g_error0 ERROR (r_icu_error)		
General		
Interrupts		
Error Events		
CPU0		
Peripherals		
PERI_ERR0 interrupt enable		
PERI_ERR1 interrupt enable		
Reset enable		
Peripheral Error 0		
CLMA3_INT (ERREVENT0)		<input type="checkbox"/>
CLMA0_INT (ERREVENT1)		<input type="checkbox"/>
CLMA1_INT (ERREVENT2)		<input type="checkbox"/>
CLMA2_INT (ERREVENT3)		<input type="checkbox"/>
BSC_WTO (ERREVENT4)		<input type="checkbox"/>
DMAC0_ERR (ERREVENT5)		<input type="checkbox"/>
DMAC1_ERR (ERREVENT6)		<input type="checkbox"/>
WDT_NMIUNDF0 (ERREVENT7)		<input checked="" type="checkbox"/>
USB_FDMAERR (ERREVENT9)		<input type="checkbox"/>

6. 点击: Generate Project Content 生成代码

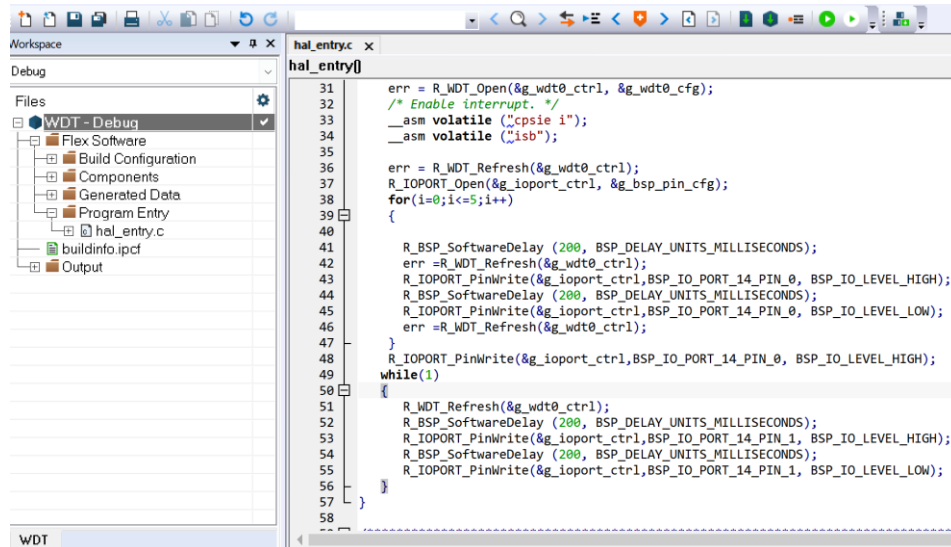
Generate Project Content

New Stack > Extend Stack > Remove

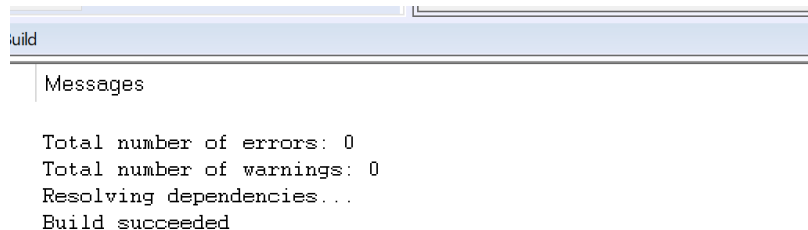
2.2

## 7. 打开生成的代码

- 仿真器由 Ijet 切换为 Jlink
- 编写用户代码：LED1 蓝灯 闪烁 5 次之后，LED2 绿灯闪烁。如果运行之后，蓝灯再次闪烁 5 次则可判定看门狗溢出引起复位。



- Rebuild All---编译工程 无报错



2.3

- Download ---下载程序，看门狗无法在仿真状态复位，下载后需复位开发板。
- 下载工程到开发板，复位开发板,观察 LED 灯变化。
- 如果屏蔽喂狗操作，重新下载代码到板子，复位板子，则能看到板子复位情况。

```
48 R_IOPORT_PinWrite(&g_ioport_ctrl,BSP_IO_PORT_14_PIN_0, BSP_IO_LEVEL_HIGH);
49 while(1)
50 {
51 //R_WDT_Refresh(&g_wdt0_ctrl);
52 R_BSP_SoftwareDelay (200, BSP_DELAY_UNITS_MILLISECONDS);
53 R_IOPORT_PinWrite(&g_ioport_ctrl,BSP_IO_PORT_14_PIN_1, BSP_IO_LEVEL_HIGH);
54 R_BSP_SoftwareDelay (200, BSP_DELAY_UNITS_MILLISECONDS);
55 R_IOPORT_PinWrite(&g_ioport_ctrl,BSP_IO_PORT_14_PIN_1, BSP_IO_LEVEL_LOW);
56 }
```

本节完

### 3 .E2studio 环境工程介绍

---

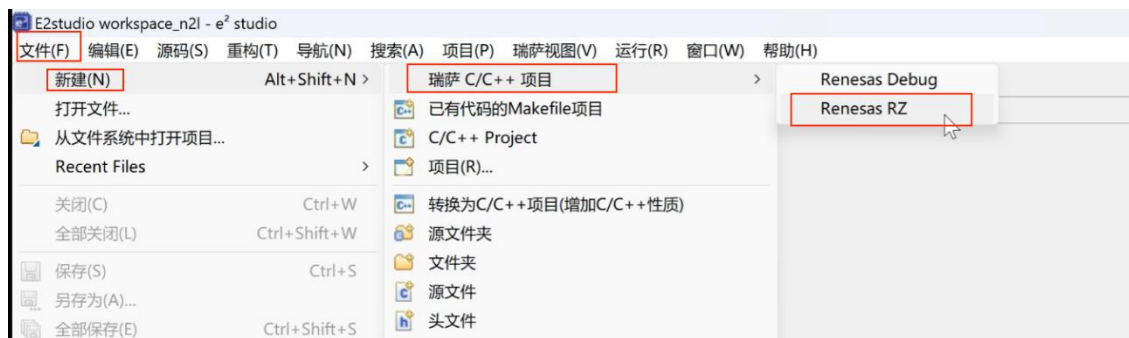
本节介绍使用 E2studio 环境创建 IIC 工程。



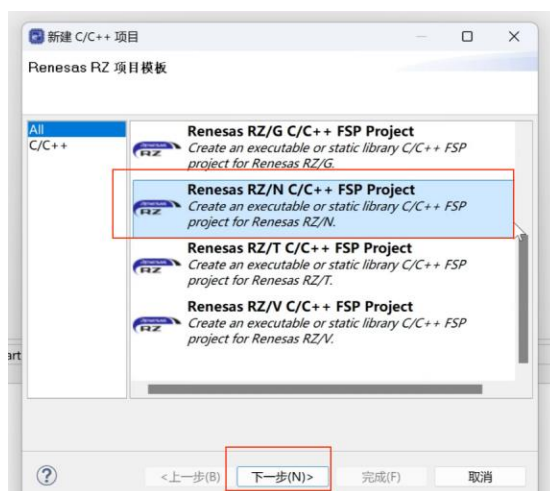
## 3.1

### ● 打开 E2studio，新建工程

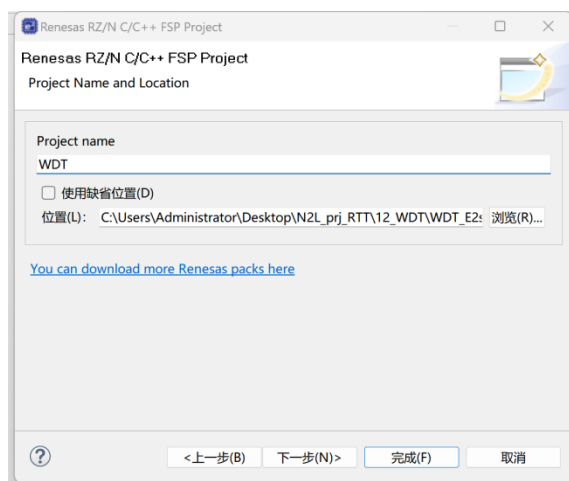
#### 1. 选择 文件--新建--瑞萨 C/C++项目--Renesas RZ:



#### 2. 选择 Renesas RZ/N C/C++ FSP Project

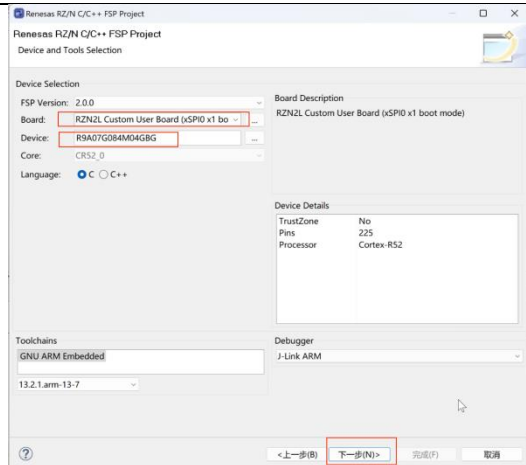


#### 3. 设置项目名称: WDT, 选择工程保存路径

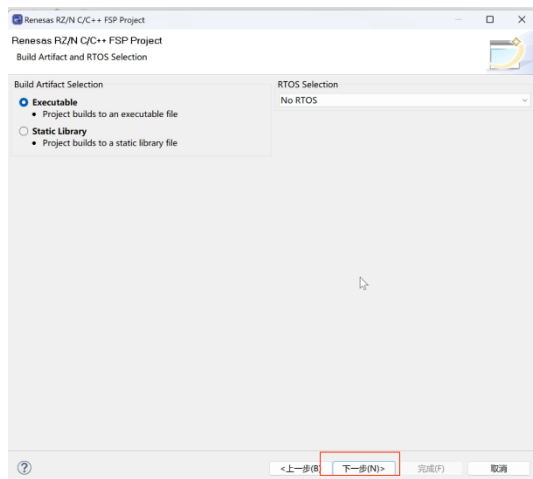


#### 4. Boot 模式选择 RZN2L Custom User Board ( XSPI0 x1 boot mode )

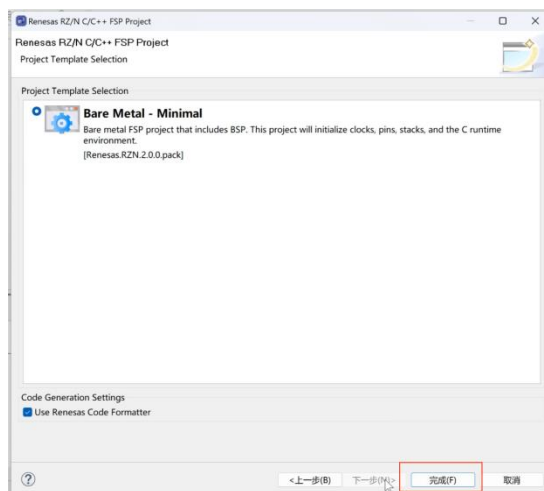
芯片型号: R9A07G084M04GBG



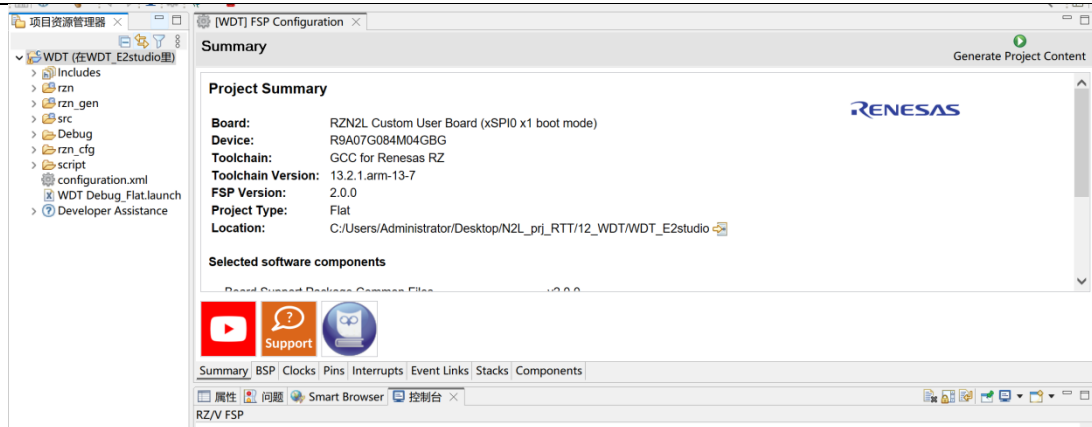
## 5. 直接下一步



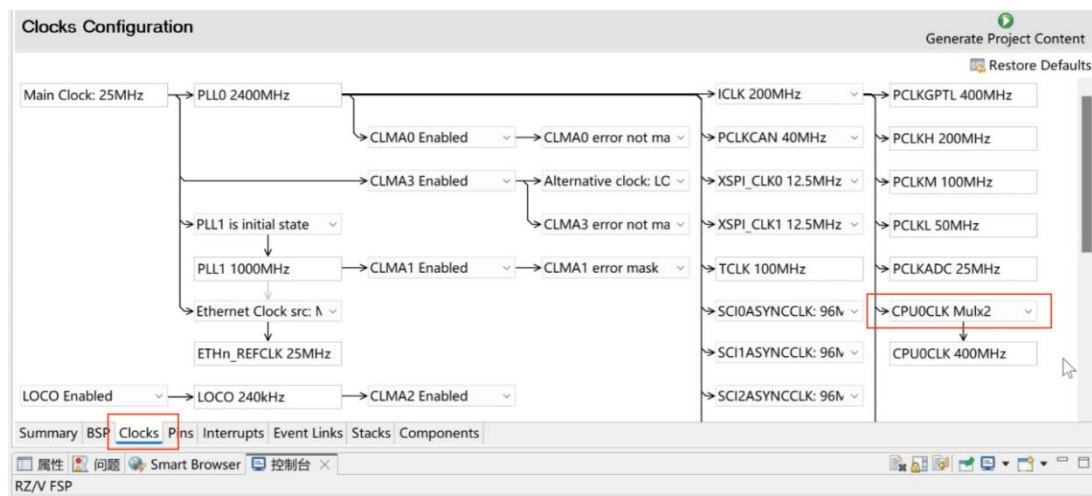
## 6. 点击 完成



## 7. 进入到工程界面



## 8. 设置时钟 CPU0CLK 400MHZ



## 3.2

### 9. 配置 XSPI0

The screenshot shows the 'Pin Configuration' window for the R9A07G084M04GBG.pincfg file. The 'Pin Selection' pane on the left shows 'Connectivity:XSPI' expanded, with 'XSPI0' selected. The 'Pin Configuration' table on the right shows the following settings:

Name	Value	Lock	Link
Pin Group Selection	Mixed		
Operation Mode	Custom(1.8V)		
Input/Output			
XSPI0_CKN	None		
XSPI0_CKP	✓ P14_6		
XSPI0_CS0#	✓ P15_7		
XSPI0_CS1#	None		
XSPI0_DS	None		
XSPI0_ECS0#	None		
XSPI0_ECS1#	None		
XSPI0_INT0#	None		
XSPI0_INT1#	None		
XSPI0_IO0	✓ P14_7		
XSPI0_IO1	✓ P15_0		

Module name: XSPI0

### 10. P140 P141 引脚配置

设置输出模式，初始电平 0

The screenshot shows the 'Pin Configuration' window for the R9A07G084M04GBG.pincfg file. The 'Pin Selection' pane on the left shows 'P14\_0' and 'P14\_1' selected. The 'Pin Configuration' table on the right shows the following settings:

Name	Value	Link
Symbolic Name		
Comment		
Mode	Output mode (Low & Not Into Input)	
Pull up/down	None	
Output Type	CMOS	
Drive Capacity	Low	
Region	Safety	
Schmitt Trigger	None	
Slew Rate	Slow	
Input/Output	✓ GPIO	

### 11. 添加 new Stack: WDT0

➤ New Stack--Driver--Monitoring---Watchdog (r\_wdt)

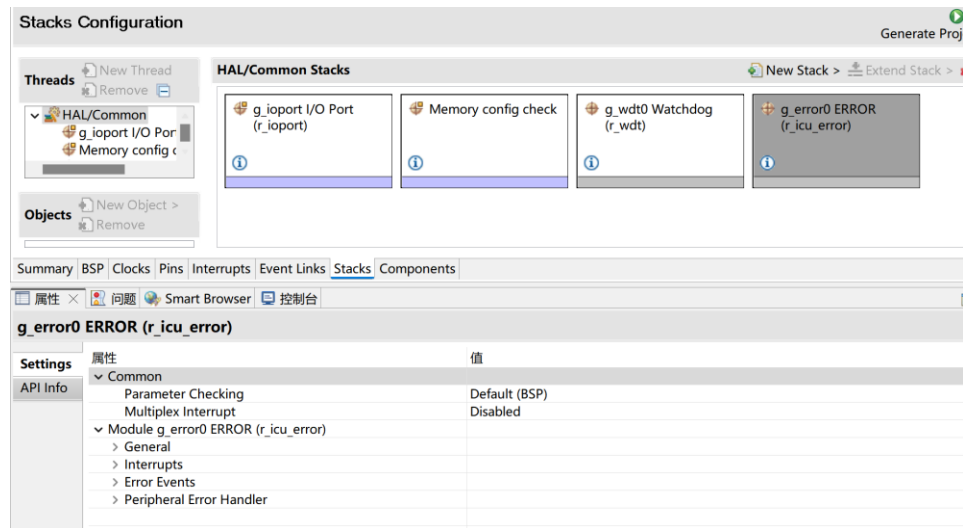
设置回调函数 wdt\_callback

The screenshot shows the 'Stacks Configuration' window. The 'HAL/Common Stacks' section shows the 'g\_wdt0 Watchdog (r\_wdt)' stack added. The 'Summary' tab at the bottom shows the configuration for 'g\_wdt0 Watchdog (r\_wdt)':

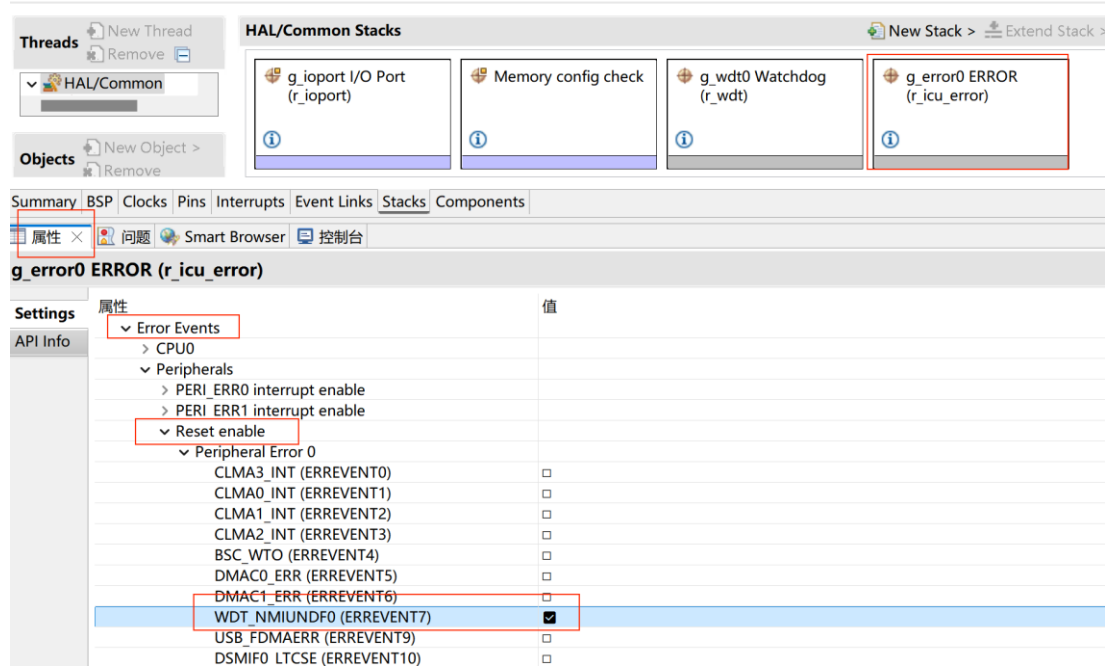
属性	值
Common	
Parameter Checking	Default (BSP)
Multiplex Interrupt	Disabled
Module g_wdt0 Watchdog (r_wdt)	
Name	g_wdt0
Timeout	16,384 Cycles
Clock Division Ratio	PCLK/8192
Window Start Position	100 (Window Position Not Specified)
Window End Position	0 (Window Position Not Specified)
WDT Callback	NULL

## 12. 添加 new stack : ERROR

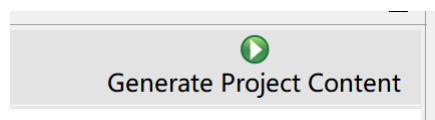
### ➤ New Stack--System--ERROR(r\_icu\_error)



### ➤ 看门狗复位使能:



## 13. 点击: Generate Project Content 生成代码



## 14. 用户代码编写

编写用户代码: LED1 蓝灯 闪烁 5 次之后, LED2 绿灯闪烁。如果运行之后, 蓝灯再次闪烁 5 次则可判定看门狗溢出引起复位。

	
<p>3.3</p>	<ul style="list-style-type: none"> <li>● 下载代码到开发板</li> </ul> <p>编译代码，无报错。</p> <ul style="list-style-type: none"> <li>➤ 下载工程到开发板，复位开发板,观察 LED 灯变化。</li> <li>➤ 如果屏蔽喂狗操作，重新下载代码到板子，复位板子，则能看到板子复位情况。</li> </ul> <pre> R_IOPORT_PinWrite(&amp;g_ioport_ctrl,BSP_IO_PORT_14_PIN_0, BSP_IO_LEVEL_HIGH); while(1) {     //R_WDT_Refresh(&amp;g_wdt0_ctrl);     R_BSP_SoftwareDelay(200, BSP_DELAY_UNITS_MILLISECONDS);     R_IOPORT_PinWrite(&amp;g_ioport_ctrl,BSP_IO_PORT_14_PIN_1, BSP_IO_LEVEL_HIGH);     R_BSP_SoftwareDelay(200, BSP_DELAY_UNITS_MILLISECONDS);     R_IOPORT_PinWrite(&amp;g_ioport_ctrl,BSP_IO_PORT_14_PIN_1, BSP_IO_LEVEL_LOW); }     </pre>

本节完