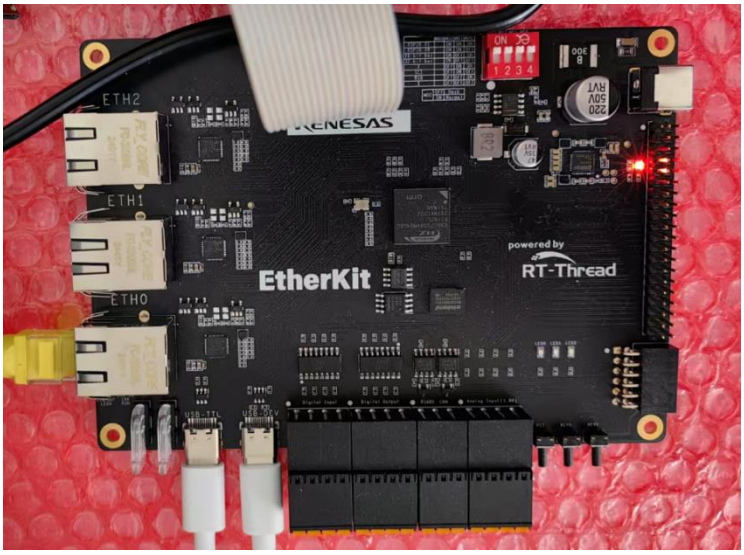
RZN2L SPI 操作手册-----基于 Etherkit 开发板

简介

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

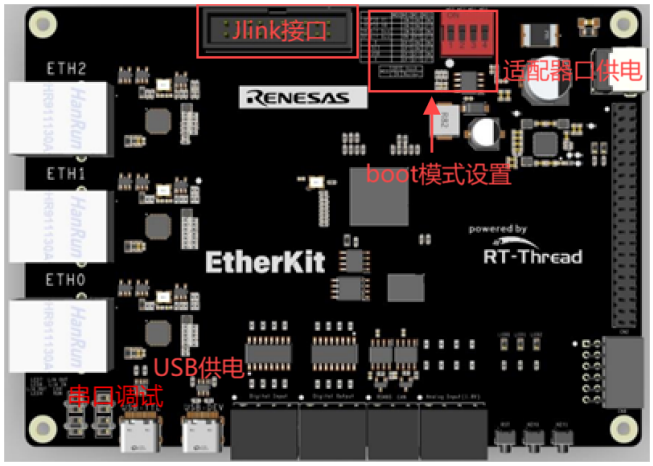
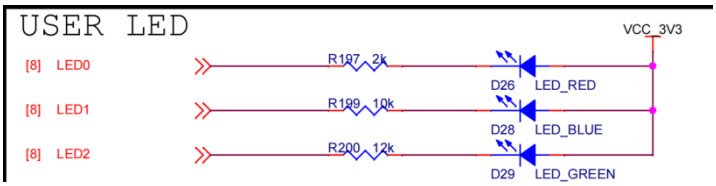
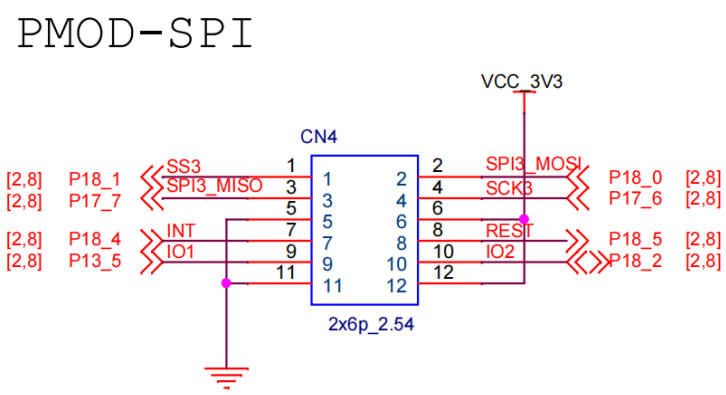


开发工具	实验材料
• IDE: IAR EW for Arm 9.50.2 E2studio 2024-01.1 • FSP：RZ/N2 FSP V2.0 • 仿真器： Jlink V12	• Etherkit 开发板 • Jlink 仿真器，需支持瑞萨 R52 内核

实验部分

1 .硬件设置及软件安装

本节 EtherKit 开发板硬件设置。

1.1	**开发板设置：** ● 供电：可选 USB 供电或适配器供电 ● Boot 模式设置：推荐 xSPI0 x1 boot mode ● Jlink v12 
1.2	**硬件原理图：** 本实验用到 LED0，对应 P143 引脚。 板载资源 PMOD 接口，连接到 N2L 芯片的 SCI _SPI3；  

本节完

2 .IAR 环境工程介绍

本节介绍 IAR 环境下 GPT 工程介绍。

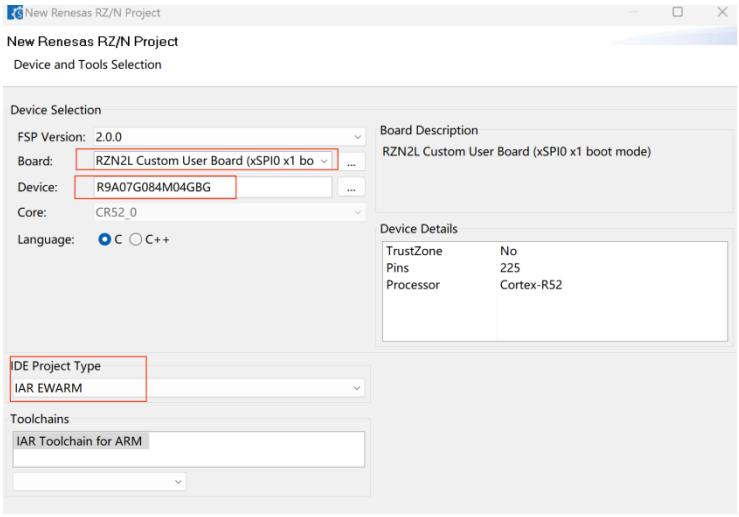
2.1

- 打开 FSP 新建工程：**SPI**

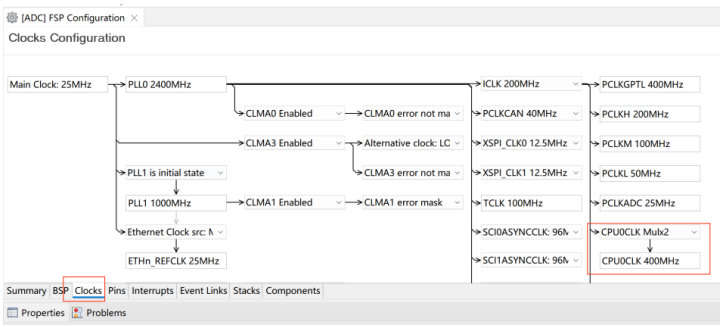
1. Boot 模式选择 RZN2L Custom User Board（ XSPI0 x1 boot mode）

芯片型号：R9A07G084M04GBG

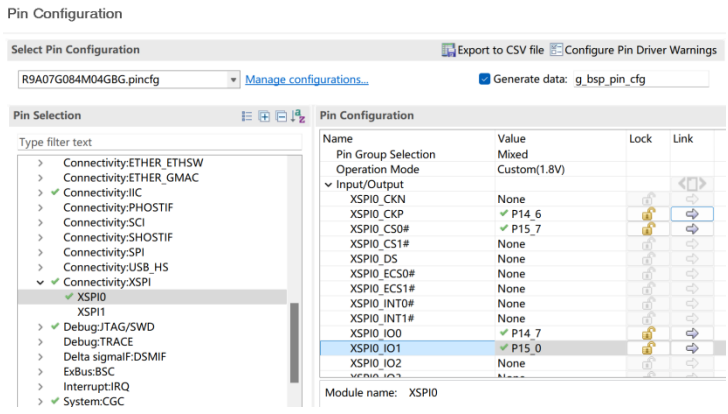
IDE Project Type 选择 IAR EWARM



1. 主时钟设置 400MHZ

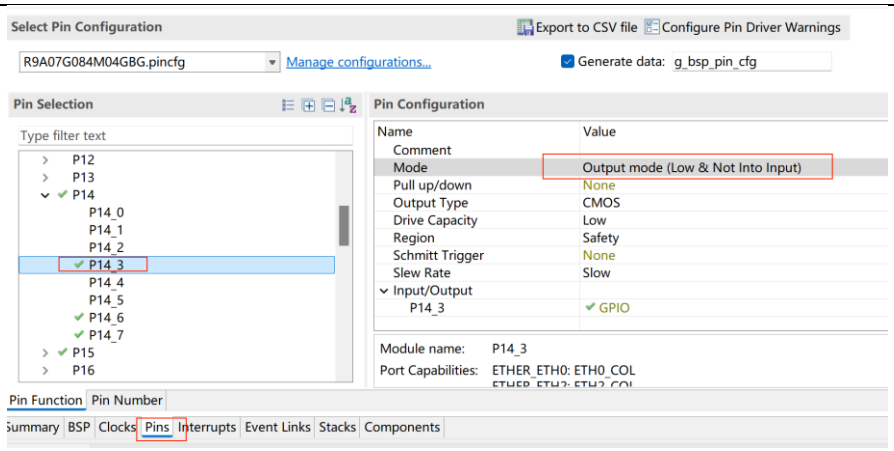


2. 配置 XSPI0

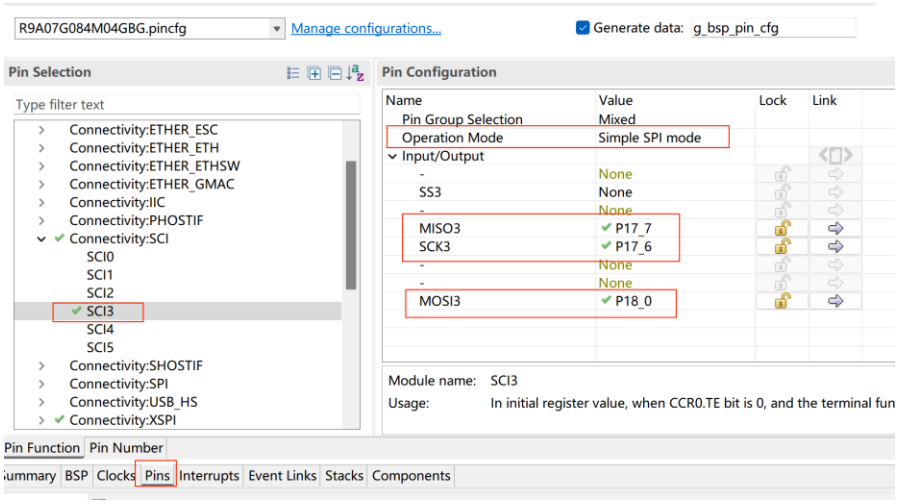


3. P143 引脚配置

P143 设置输出模式，初始电平 0

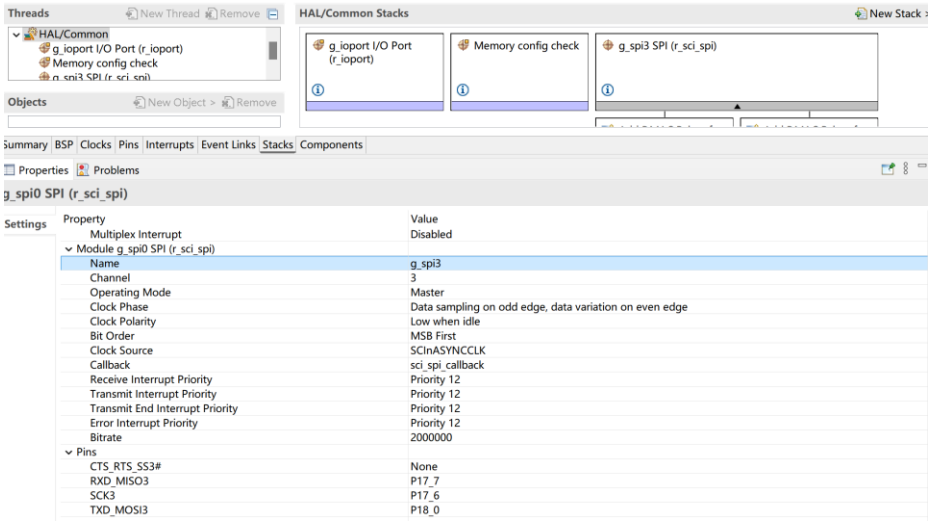


4. 配置 SCI3

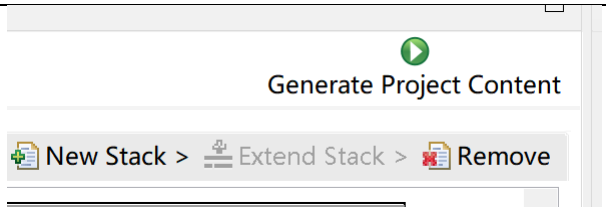


5. 添加 new Stack： SPI

New Stack---Connectivity--SPI(r_sci_spi)，删除 DMA



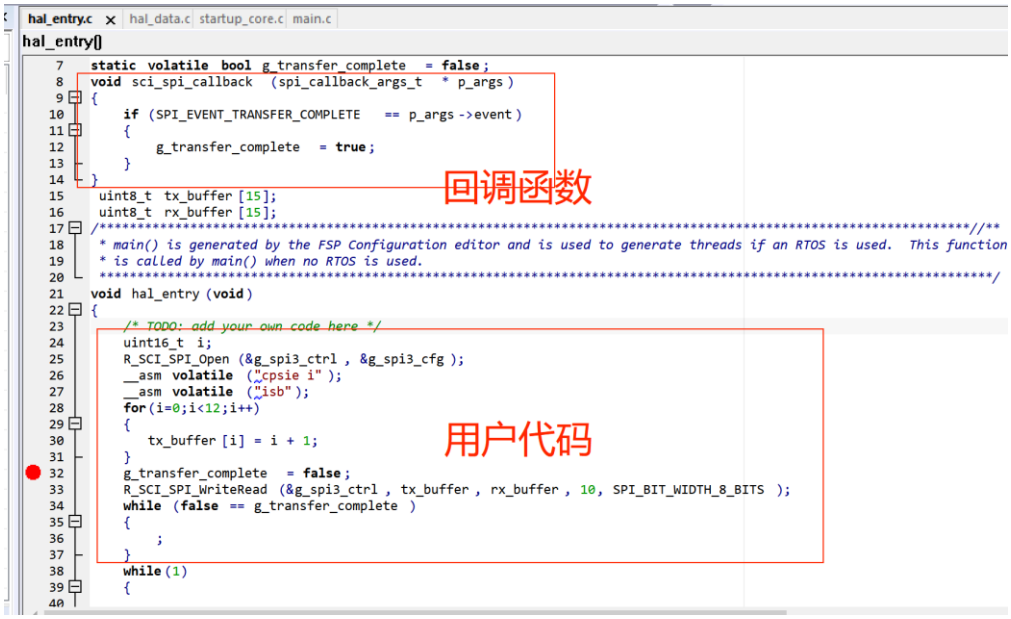
6. 点击： Generate Project Content 生成代码



2.2

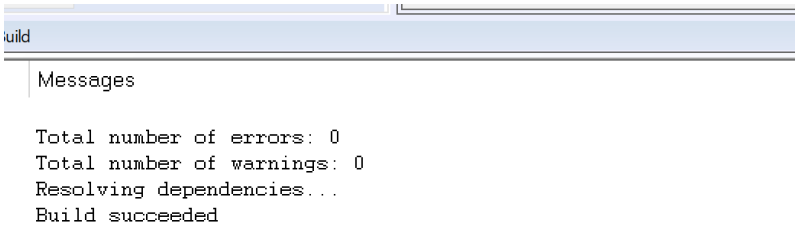
7. 打开生成的代码

- 仿真器由 Ijet 切换为 Jlink
- 编写用户代码：将 P17_7 和 P18_0 短接，进行 SPI 回环测试，自发自收。




```
static volatile bool g_transfer_complete  = false;
void sci_spi_callback (spi_callback_args_t * p_args)
{
    if (SPI_EVENT_TRANSFER_COMPLETE == p_args->event)
    {
        g_transfer_complete  = true;
    }
}
uint8_t tx_buffer[15];
uint8_t rx_buffer[15];
/**********************************************************************************************************************
 * main() is generated by the FSP Configuration editor and is used to generate threads if an RTOS is used.  This function
 * is called by main() when no RTOS is used.
 *********************************************************************************************************************/
void hal_entry (void)
{
    /* TODO: add your own code here */
    uint16_t i;
    R_SCI_SPI_Open (&g_spi3_ctrl , &g_spi3_cfg );
    __asm volatile ("cpsie i" );
    __asm volatile ("isb" );
    for(i=0;i<12;i++)
    {
        tx_buffer[i] = i + 1;
    }
    g_transfer_complete  = false;
    R_SCI_SPI_WriteRead (&g_spi3_ctrl , tx_buffer , rx_buffer , 10, SPI_BIT_WIDTH_8_BITS );
    while (false == g_transfer_complete )
    {
        ;
    }
    while(1)
    {
```

- Rebuild All---编译工程 无报错

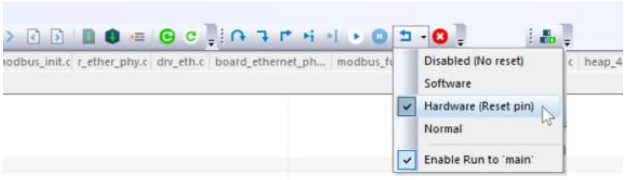


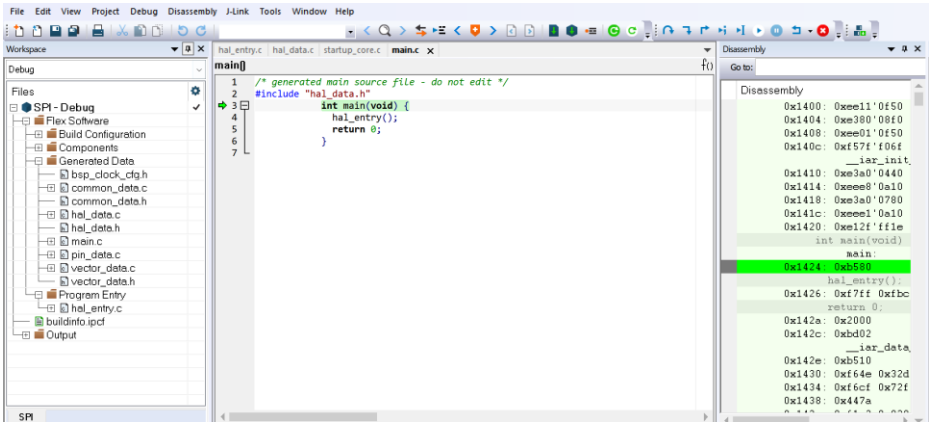
2.3

- Download and Debug ---下载程序

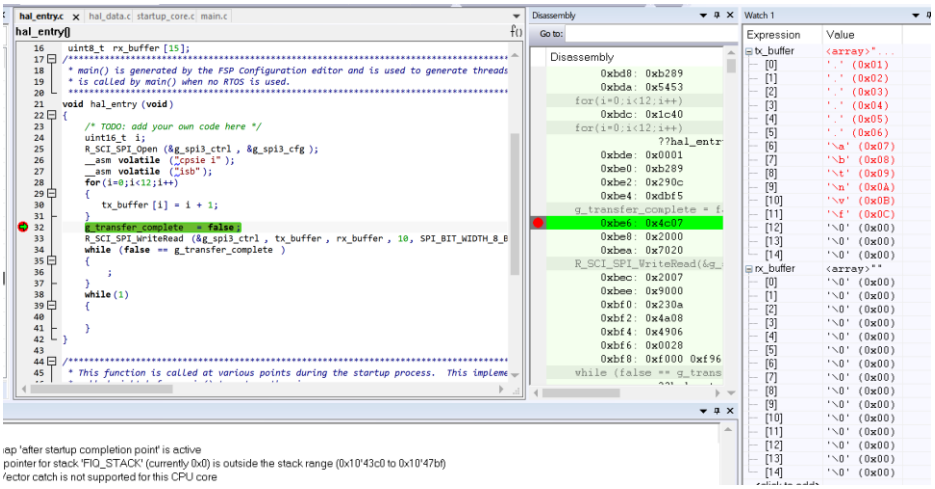
➢ 下载工程到开发板，进入仿真界面

➢ Debug 复位设置为 Hardware

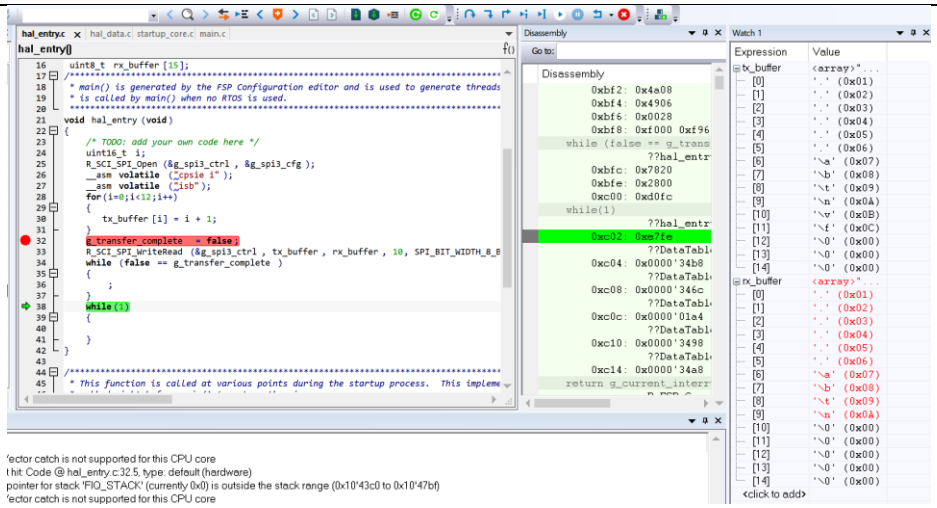




➢ 设置断点，程序运行之断点处，watch 查看 tx_buffer 和 rx_buffer 值。



➢ 全速运行代码，然后点击暂停：



可以看到 rx_buffer 中接收到 SPI 发送来的数据。

本节完

3 .E2studio 环境工程介绍

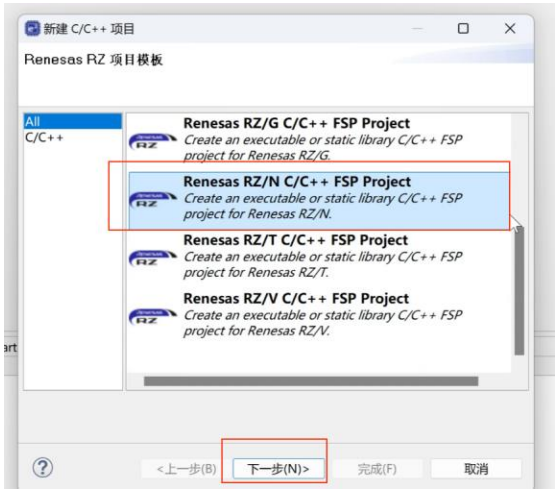
本节介绍使用 E2studio 环境创建 IIC 工程。

3.1

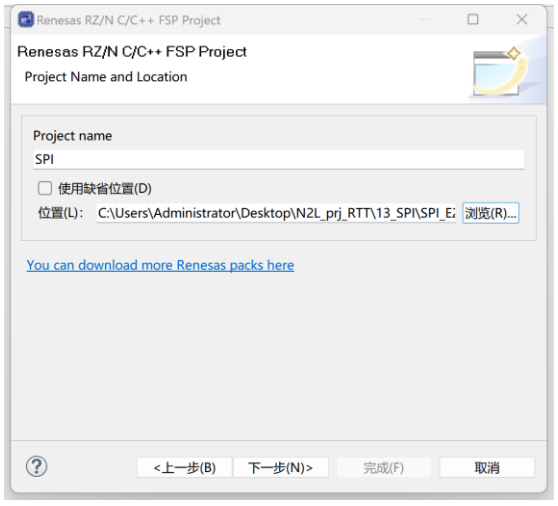
- 打开 E2studio，新建工程

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

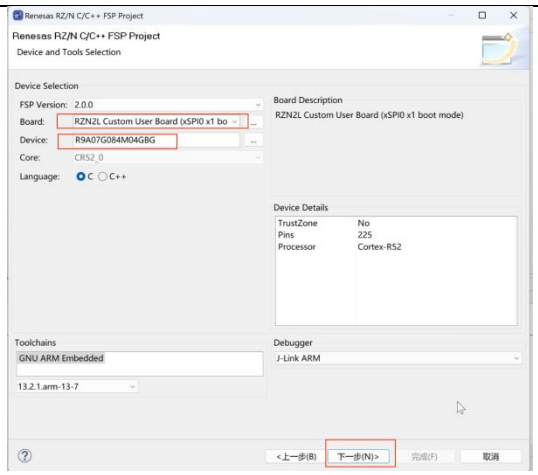


3. 设置项目名称：SPI，选择工程保存路径

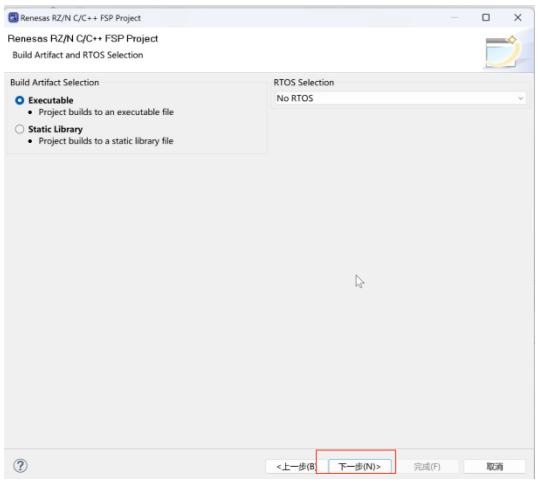


4. Boot 模式选择 RZN2L Custom User Board（ XSPI0 x1 boot mode）

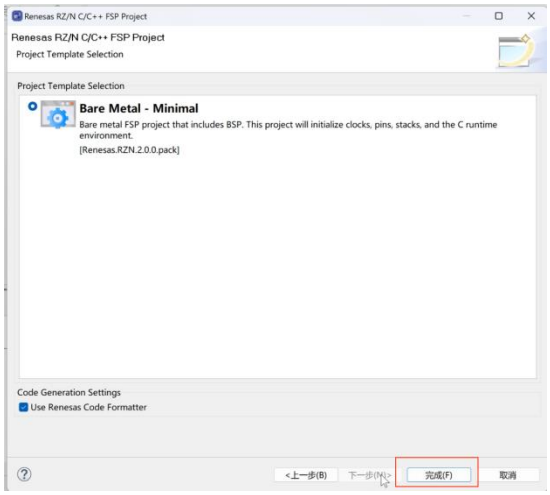
芯片型号：R9A07G084M04GBG



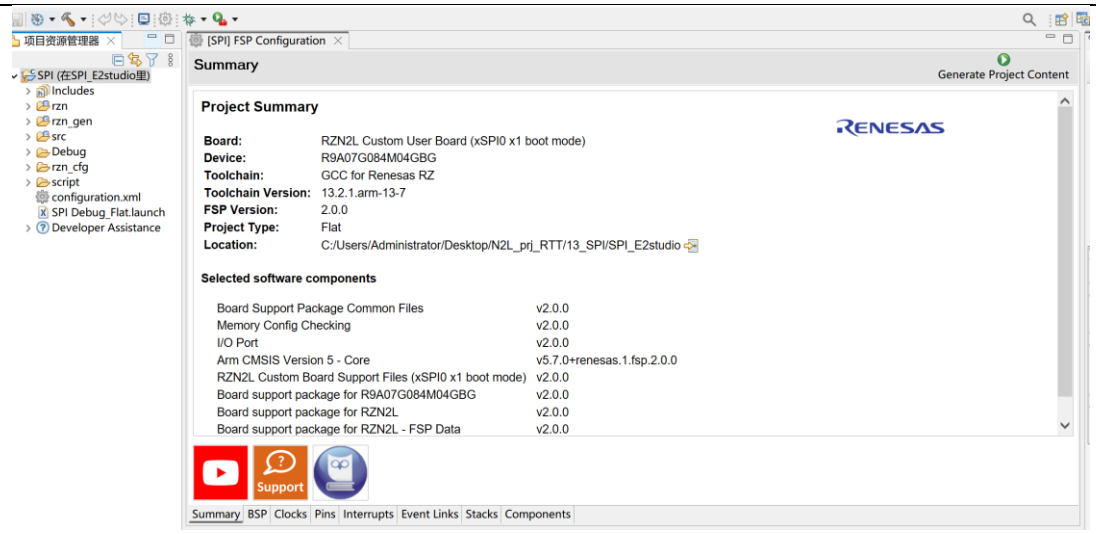
5. 直接下一步



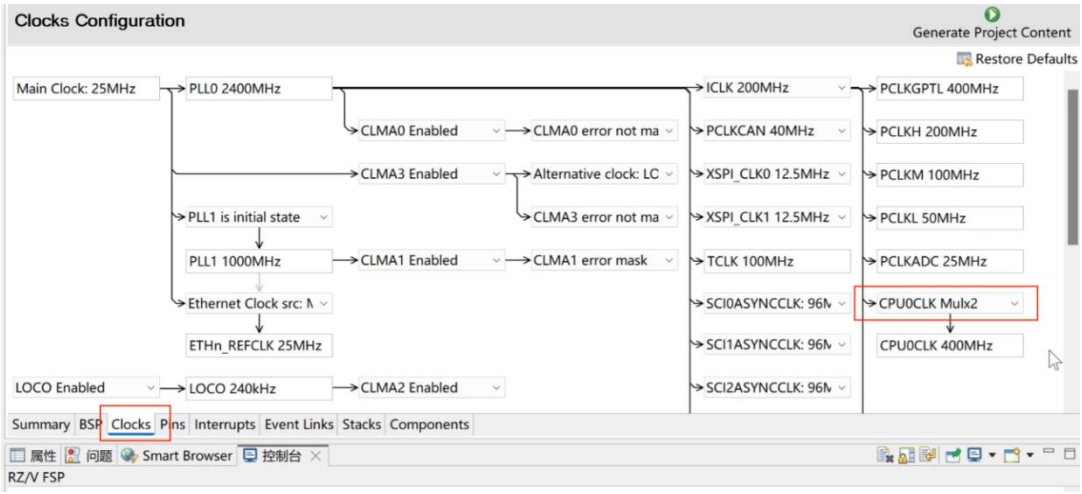
6. 点击 完成



7. 进入到工程界面

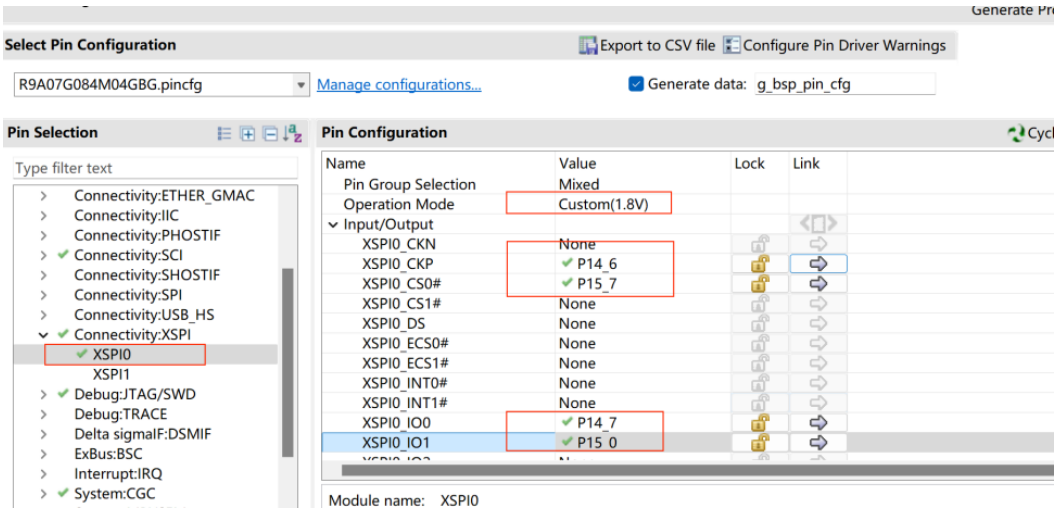


8. 设置时钟 CPU0CLK 400MHZ



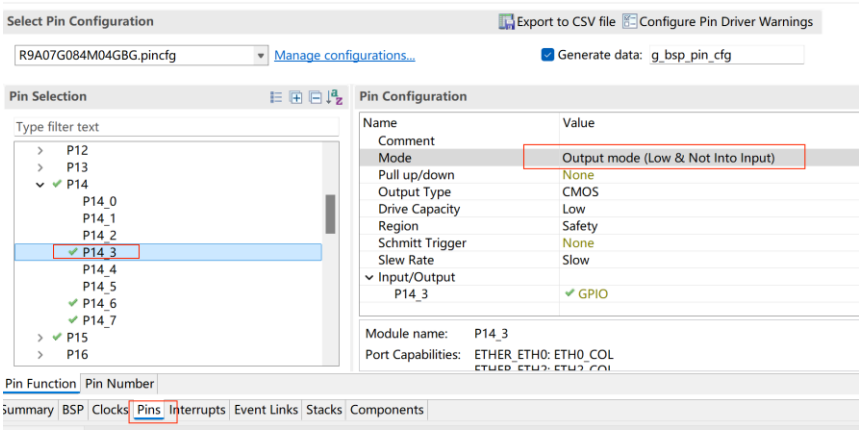
3.2

9. 配置 XSPI0

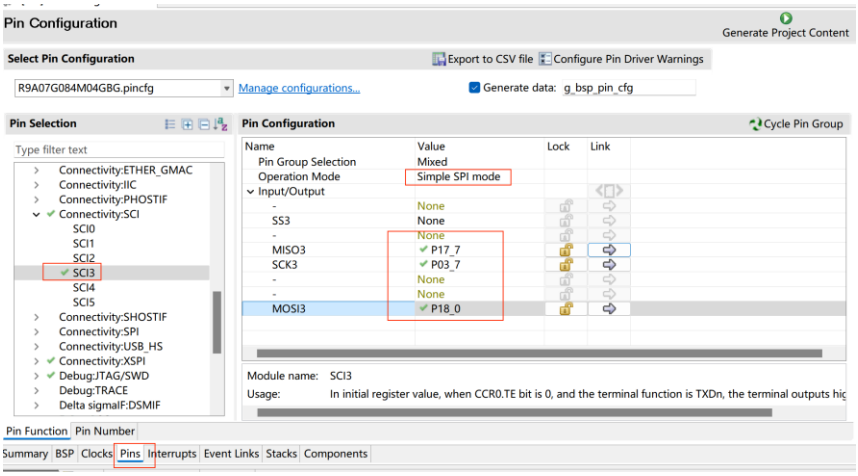


10. P143 引脚配置

P143 设置输出模式，初始电平 0

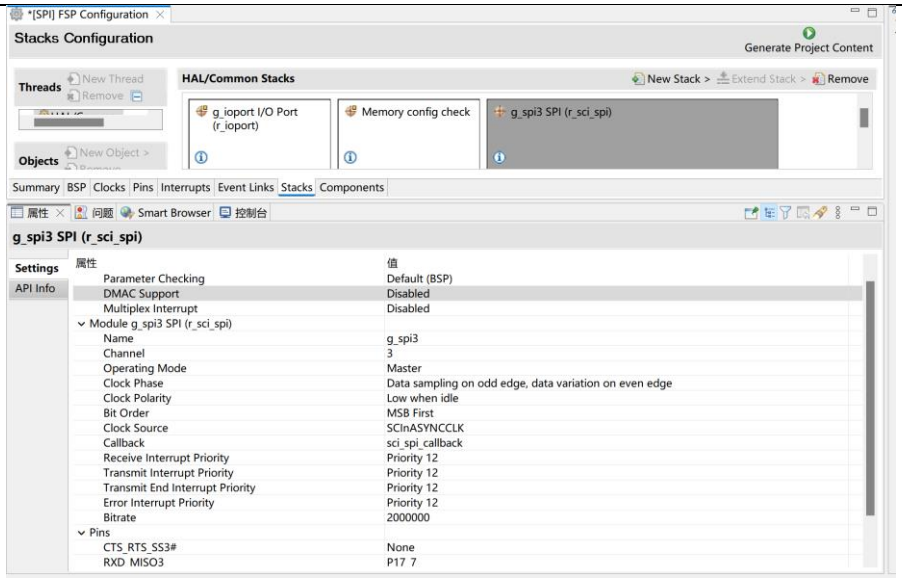


11. 配置 SCI3



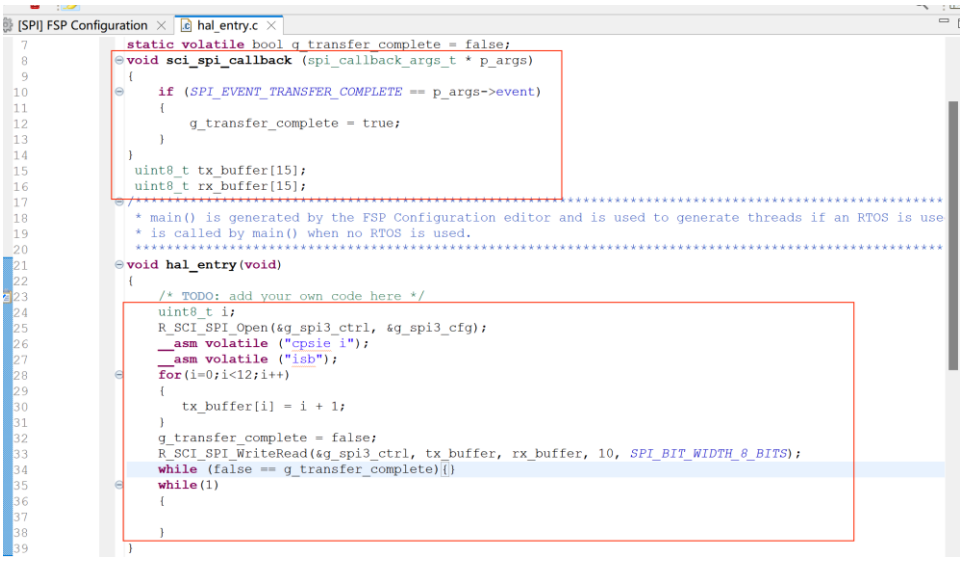
12. 添加 new Stack： SPI

New Stack---Connectivity--SPI(r_sci_spi) 删除默认的 DMA



13 . 点击：Generate Project Content 生成代码

编写用户代码

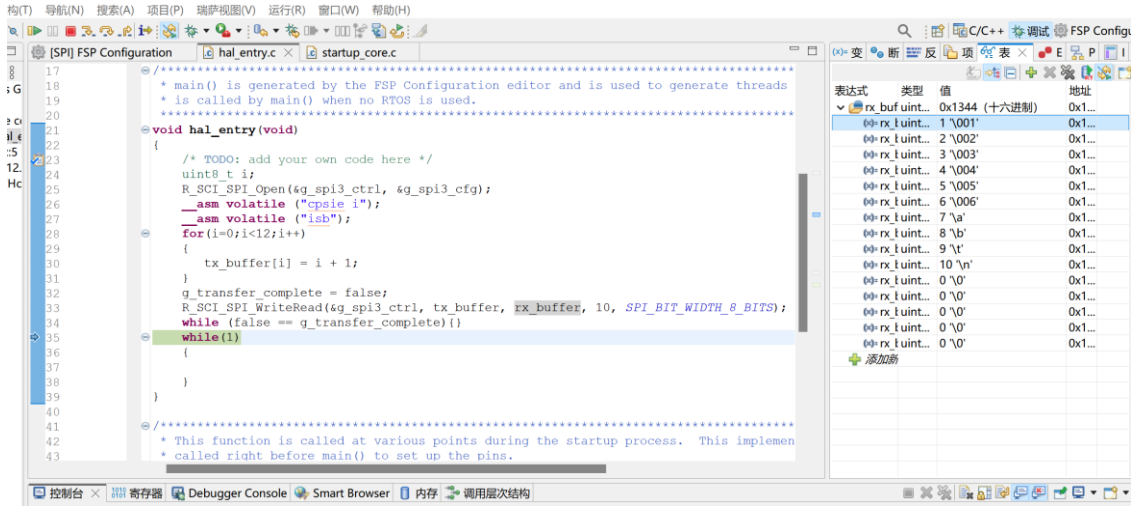


3.3	● 下载代码到开发板 编译代码，无报错。 E2studio 在线仿真下载，全速运行

3.4

➢ 全速运行代码。

暂停仿真，watch 窗口观察 rx_buffer 收到 SPI 发送的数据。



本节完