

APPLICATION NOTE:

Using LittleFS MXIC Middleware on Renesas FSP

Rev. 2.2, December, 2023

Contents

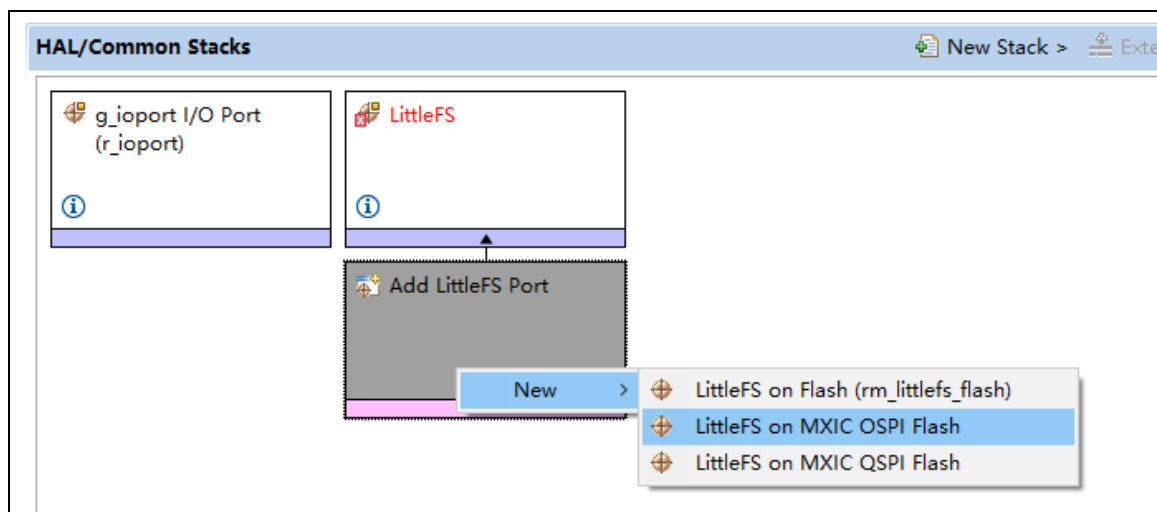
1	Introduction.....	- 3 -
2	Required Resources	- 4 -
3	Support MXIC LittleFS Middleware on FSP.....	- 5 -
3.1	Store the Pack File into the Specified Location	- 5 -
3.2	Create a New Renesas FSP Project	- 6 -
3.3	Add related Modules into the Project.....	- 7 -
	Revision History	- 13 -

1 Introduction

This application note describes how to use the MXIC LittleFS Middleware provided by Macronix Company to develop Macronix Flash to LittleFS on Renesas FSP.

In Renesas FSP, Users can add LittleFS to their project, and the filesystem memory is internal flash. The **LittleFS on Flash** provide a middleware to connect LittleFS and internal flash driver.

Here we provide two choice, the **LittleFS on MXIC QSPI Flash** is based on **QSPI** module and **LittleFS on MXIC OSPI Flash** is based on **OSPI** module. So user can use any MXIC Flash which has QSPI/OSPI interface in FSP. In addition, **LittleFS on MXIC QSPI Flash** can also support SPI_Nand Flash, SPI_Nand is Nand Flash while it can be accessed by SPI interface. Here we add SPI_Nand driver to **LittleFS on MXIC QSPI Flash** Middleware and user can select SPI_Nand as the LittleFS memory.



2 Required Resources

To perform the following steps in this application note, you will need:

■ **Module pack file:**

- ☐ MXIC.LittleFS_MXIC_QSPI_Flash.5.0.0.pack
- ☐ MXIC.LittleFS_MXIC_OSPI_Flash.5.0.0.pack

■ **Test bench:**

- ☐ hal_entry.c
- ☐ common_init.c
- ☐ common_init.h
- ☐ SEGGER_RTT folder

Note: You can get the above files from

“LittleFS_on_MXIC_QSPI_Flash_Sample_Project” and

“LittleFS_on_MXIC_OSPI_Flash_Sample_Project”.

■ **Flash device:**

- ☐ Macronix MX25L Series NOR Flash
- ☐ Macronix MX25LM Series NOR Flash
- ☐ Macronix MX31 Series SPI_Nand Flash

■ **Development platform:**

- ☐ Renesas RA Series (This document uses RA6M5 as an example)

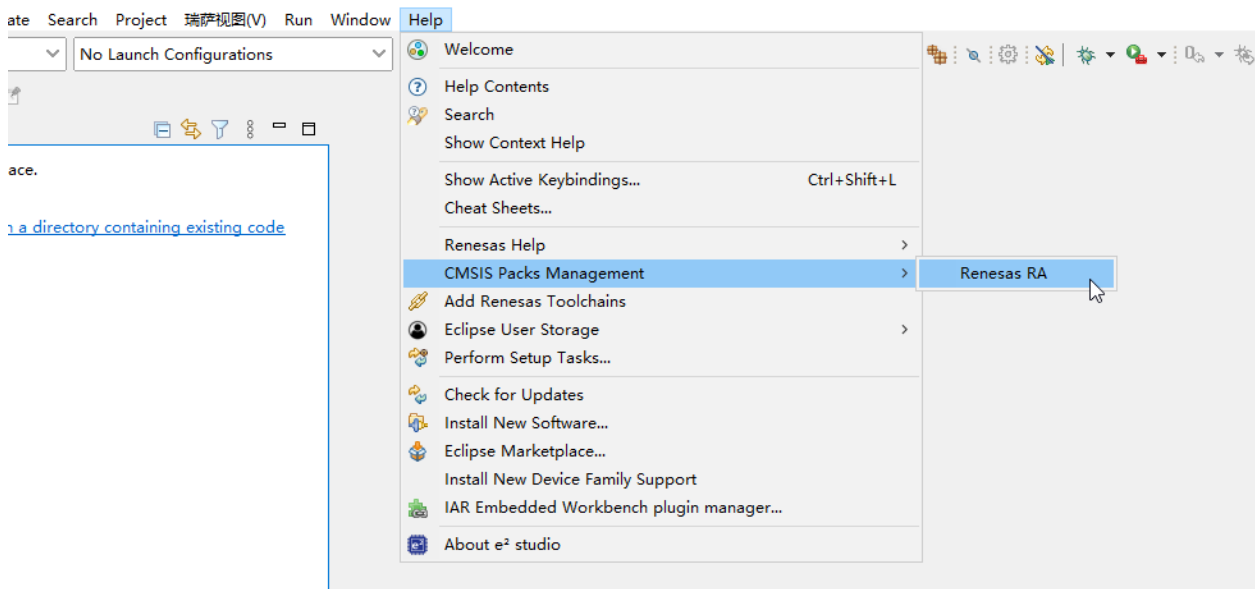
■ **FSP: FSP V5.0.0**

3 Support MXIC LittleFS Middleware on FSP

In this section, we'll describe how to use the **LittleFS on MXIC QSPI Flash** to enable Macronix NOR Flash and SPI_Nand Flash on Renesas RA6M5 platform. The usage of OSPI middleware is similar to that of QSPI. Here we choose to using FSP V5.0.0 to generate the project.

3.1 Store the Pack File into the Specified Location

We assume that FSP is installed successfully, so you should import the pack as follows.



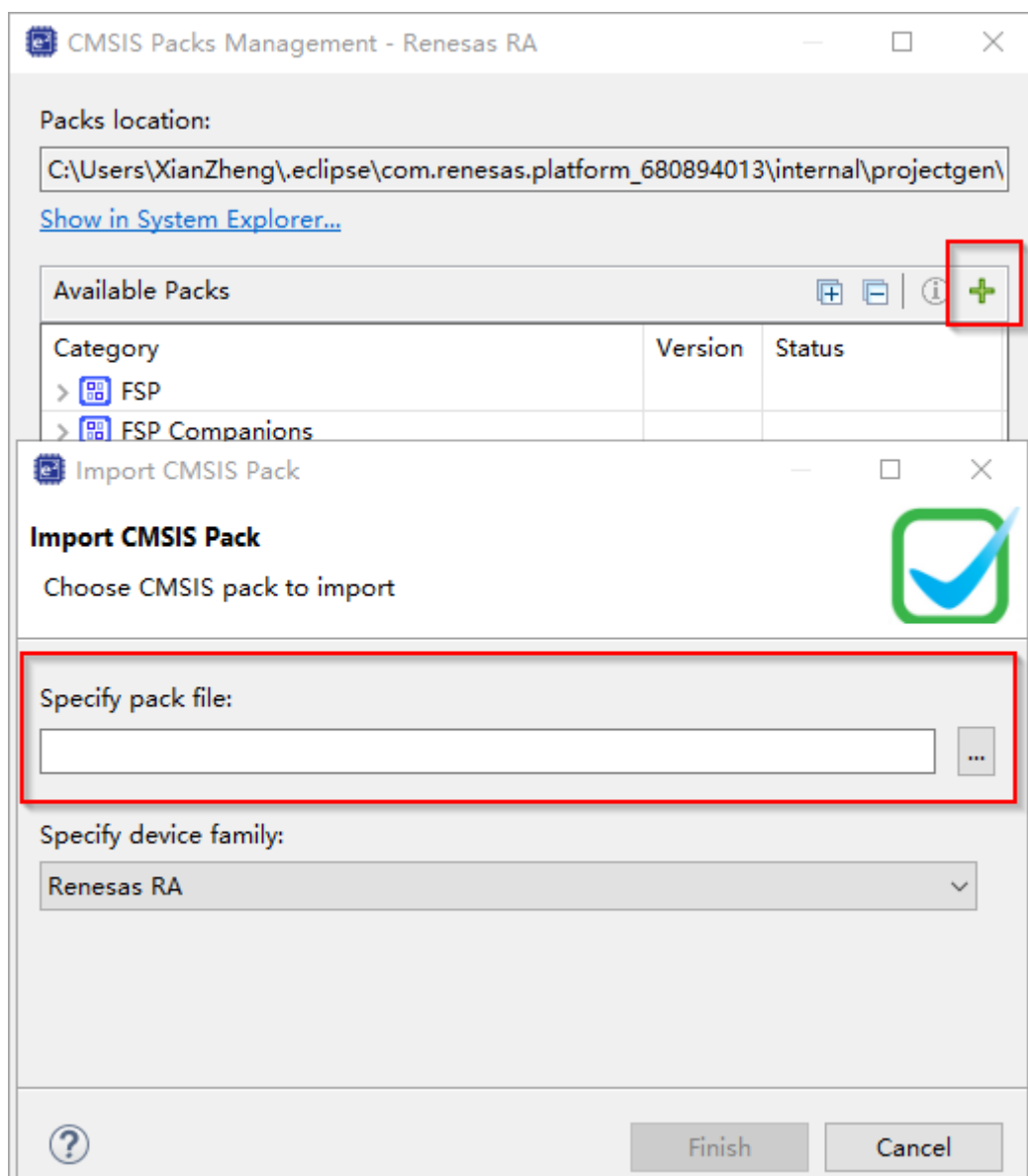


Figure 3.1 Import pack

3.2 Create a New Renesas FSP Project

In this case, we choose FSP V5.0.0 and Renesas RA6M5 to realize whole functions. Figure 3.2 shows the selection of project.

Note:

- For the Board, select EK-RA6M5.
- For the FSP version, select FSP v5.0.0
- For the device, keep the R7FA6M5BH3CFC.

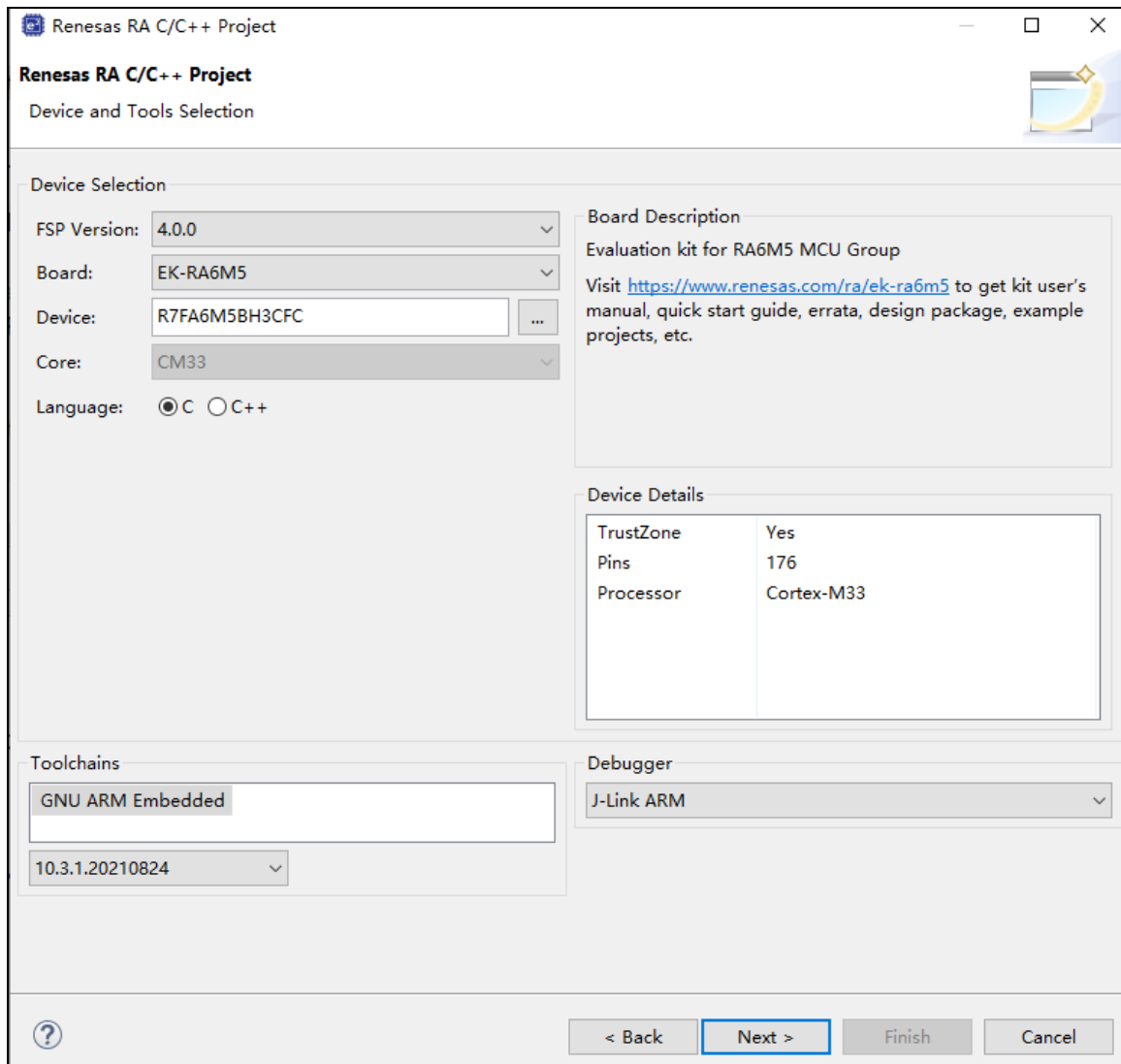


Figure 3.2 Creating a New Project

Upon clicking the 'Next' button, you should select **Executable** and **Bare Metal – Minimal**, then click 'Finish' button.

3.3 Add related Modules into the Project

1) Add LittleFS Module

Open the project created in section 3.2, in the **FSP Configuration** window, click the **Stacks** tab. Add the **LittleFS** into your existing project.

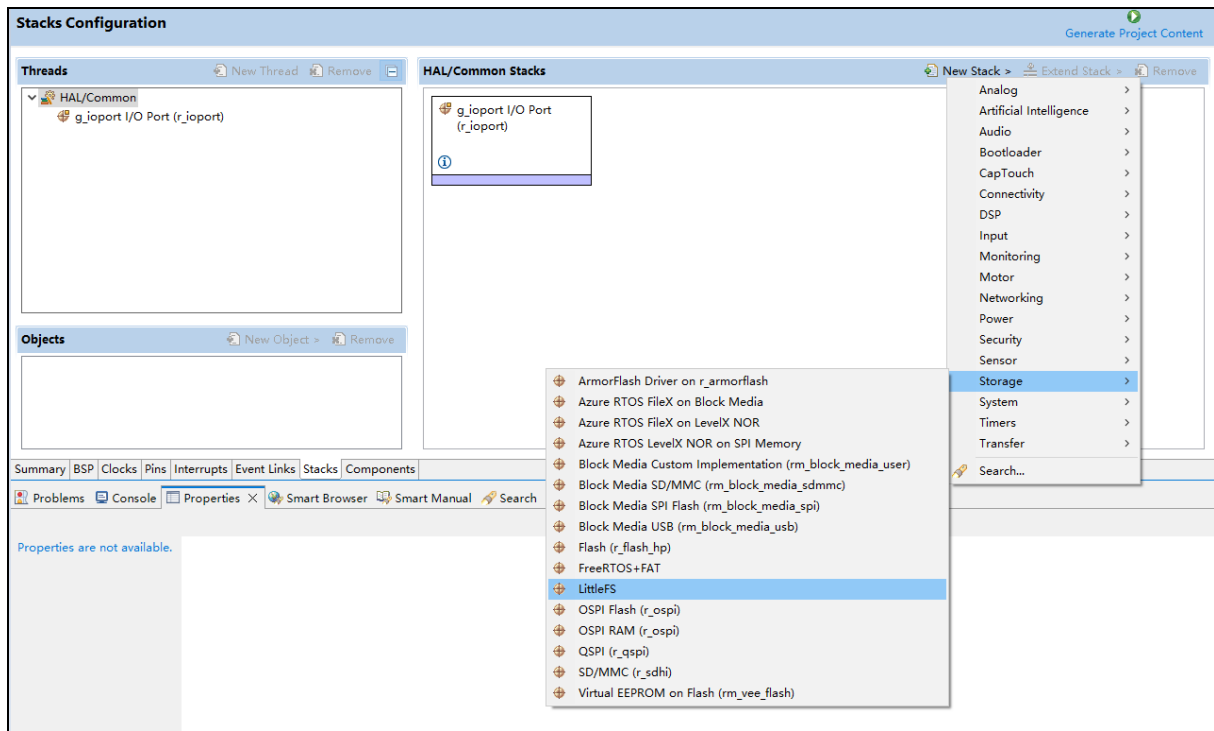


Figure 3.3 Add LittleFS Module to the Project

2) Add MXIC LittleFS Middleware Module

After add **LittleFS** Module, you can add LittleFS on MXIC Flash Module as Introduction shows.

You can select QSPI Middleware or OSPI Middleware according to the actual situation, and then make the correct configuration according to the FSP prompt(stack, heap, clock, etc.).

Note: Please take care of QSPI/OSPI controller's work mode and pins configuration.

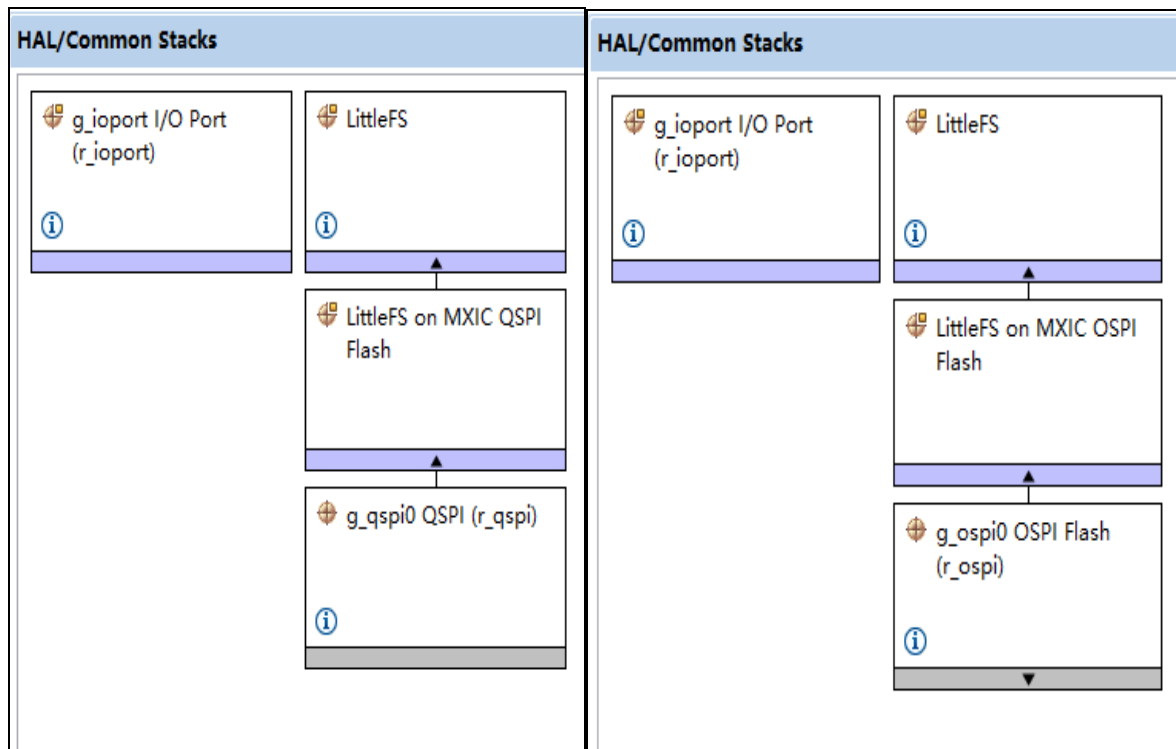


Figure 3.4 Add MXIC Middleware to the project

3) Parameter Setting

When using ‘**LittleFS on MXIC QSPI Flash**’ module, you should select Normal Nor Flash or SPI_Nand Flash, and then configure LittleFS settings.

- User can select Normal Nor Flash or SPI_Nand Flash, but you have to make sure the hardware is connected correctly.
- The other setting about LittleFS should refer to flash data sheet and actual requirement.

LittleFS on MXIC QSPI Flash		
Settings	属性	值
	▼ Common	
	Parameter Checking Enable	Default (BSP)
	Select Flash Type	Normal Nor Flash
	NFTL	Normal Nor Flash
	▼ Module LittleFS on MXIC QSPI Flash	SPI_Nand Flash
	Name	g_rm_littlefs0
	Read Size	256
	Program Size	256
	Block Size (bytes)	4096
	Block Count	1024
	Block Cycles	1024
	Cache Size	256
	Lookahead Size	16

Figure 3.5 Parameter Setting

The SPI_Nand parameter has a little different from normal nor flash, please refer SPI_Nand datasheet for more information.

LittleFS on MXIC QSPI Flash		
Settings	Property	Value
	▼ Common	
	Parameter Checking Enable	Default (BSP)
	Select Flash Type	SPI_Nand Flash
	NFTL	Disabled
	▼ Module LittleFS on MXIC QSPI Flash	
	Name	g_rm_littlefs0
	Read Size	2048
	Program Size	2048
	Block Size (bytes)	0x20000
	Block Count	1024
	Block Cycles	1024
	Cache Size	2048
	Lookahead Size	16

Figure 3.6 SPI_Nand Parameter Setting

When using ‘**LittleFS on MXIC OSPI Flash**’ module, you will not select Flash type and NFTL, but you should pay attention to whether the OSPI Mapping Address setting is correct. For the other configurations, refer to QSPI Normal Nor Flash’s settings.

4) Generate Source Code and Configuration File

After you have made all your customization on the Properties and Pins, click the **Generate Project Content** button. The LittleFS module source code, middleware code, and QSPI Driver files will be automatically generated in the project as shown in Figure 3.6.

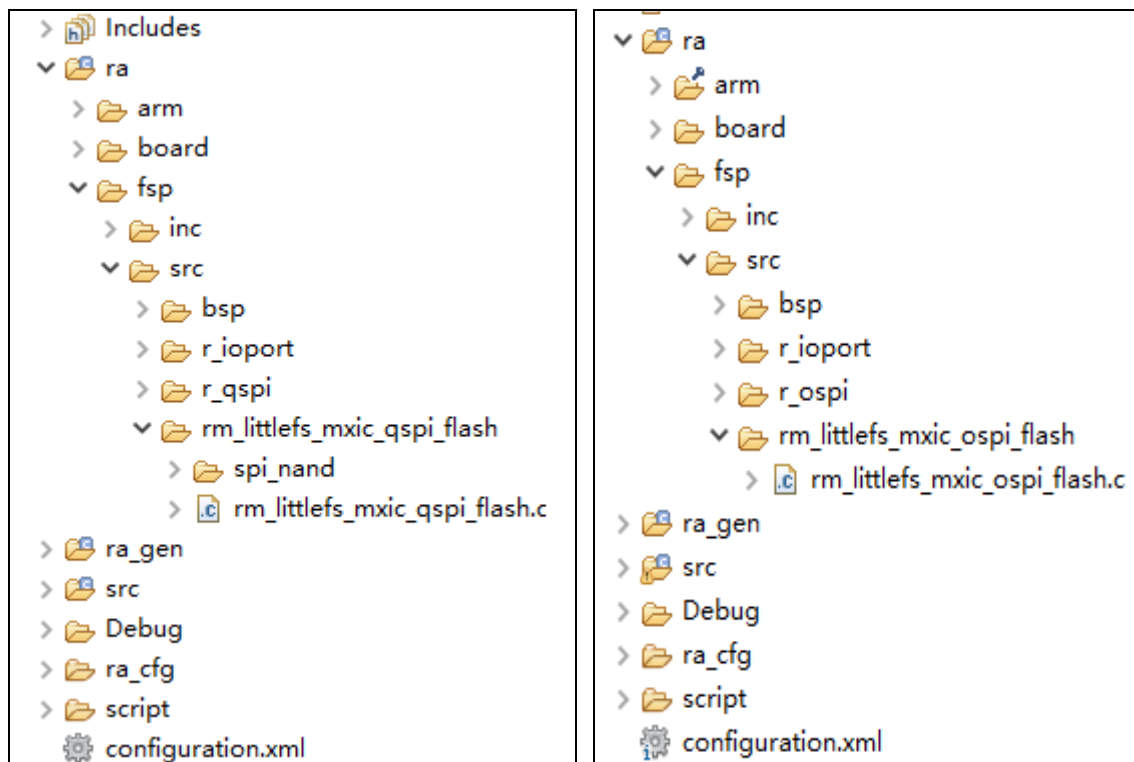


Figure 3.7 Generate Source Code

5) Enable NFTL

If you want to add NFTL in your project, you can enable NFTL option in the configuration.

LittleFS on MXIC QSPI Flash		
Settings	属性	值
	▼ Common	
	Parameter Checking Enable	Default (BSP)
	Select Flash Type	SPI_Nand Flash
	NFTL	Enabled
	▼ Module LittleFS on MXIC QSPI Flash	
	Name	g_rm_littlefs0
	Read Size	2048
	Program Size	2048
	Block Size (bytes)	2048
	Block Count	16000
	Block Cycles	1024
	Cache Size	2048
	Lookahead Size	16

Figure 3.8 Enable NFTL

Please note that NFTL can only use for SPI_Nand Flash, and you should set the parameter according to actual Flash.

6) Add RTT and Test Files to the Project

As section 2 introduced, we offer Sample project, and you should add test bench into the project as shown in Figure3.8.

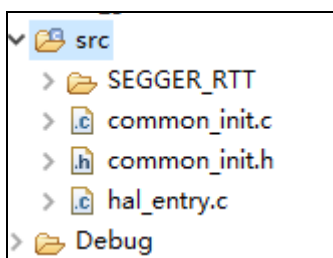


Figure 3.9 Adding RTT and Test File to the Project

7) Compile and Run the Demo

After all above has been done, you should pay attention to the project's stack and heap. You can set appropriate stack/heap value in normal flash, but for SPI Nand, especially in NFTL enabled, you should have a large value. In our sample project, we set stack:0x4000, heap:0x4000 with NFTL disabled and stack:0x8000, heap:0x25000 with NFTL enabled.

Please compile and run the demo, observe the result.

Revision History

Date	Revision No.	Description
May/27/2021	1.0	First Version.
Feb/15/2022	1.1	Add NFTL
Jan/11/2023	2.0	Update to FSP V4.0
April/24/2023	2.1	Update pack import
December/1/2023	2.2	Update to FSP V5.0

Except for customized products which have been expressly identified in the applicable agreement, Macronix's products are designed, developed, and/or manufactured for ordinary business, industrial, personal, and/or household applications only, and not for use in any applications which may, directly or indirectly, cause death, personal injury, or severe property damages. In the event Macronix products are used in contradicted to their target usage above, the buyer shall take any and all actions to ensure said Macronix's product qualified for its actual use in accordance with the applicable laws and regulations; and Macronix as well as its suppliers and/or distributors shall be released from any and all liability arisen therefrom.

Copyright© Macronix International Co., Ltd. 2023. All rights reserved, including the trademarks and tradename thereof, such as Macronix, MXIC, MXIC Logo, MX Logo, Integrated Solutions Provider, Nbit, Macronix NBit, HybridNVM, HybridFlash, HybridXFlash, XtraROM, KH Logo, BE-SONOS, KSMC, Kingtech, MXSMIO, Macronix vEE, Macronix MAP, RichBook, Rich TV, OctaRAM, OctaBus, OctaFlash, and FitCAM. The names and brands of third party referred thereto (if any) are for identification purposes only.

For the contact and order information, please visit Macronix's Web site at: <http://www.macronix.com>.