

RYZ012 and RA MCU

Firmware Update from BLE Radio OTA

Introduction

This document describes a sample application that updates the RYZ012 firmware via BLE Radio OTA.

The application example works in a configuration that uses the EK-RA4M2 board with RA4M2 as the host MCU and connects the PMOD[™] Expansion Board for RYZ012 Bluetooth LE Module to the PMOD connector. The steps in this document show the user how to load an RYZ012 firmware file on a Mobile Device running the OTA application and connect over BLE to program the firmware into the RYZ012.

Target Devices

- RA4M2
- RYZ012

Related Documents

- Renesas Flexible Software Package (FSP) User's Manual (R11UM0155)
- e² studio 2023-04 User's Manual: Quick Start Guide MCU RA Family (R20UT4989)
- EK-RA4M2 Quick Start Guide (R20QS0018)
- RYZ012A1 PMOD Expansion Board Quick Start Guide (R21QS0002)
- RYZ012 Datasheet (R12DS0002)
- RYZ012 Bluetooth LE Sample Application (R01AN6116EJ)
- QE for BLE [RA,RE,RX] Release Note (R20UT5145EJ)

Required Resources

To build and run the RYZ012 firmware update application example, the following resources are needed.

Development tools and software

- e² studio IDE v2023-04
- Flexible Software Package (FSP) v4.4.0
- QE for BLE Tool [RA, RE, RX] V1.6.0 for e² studio IDE
- SEGGER J-Link RTT Viewer V7.60f
- TelinkBleOTA app v2.0.5 for iOS (Apple App Store)

Hardware

- EK-RA4M2 kit (RTK7EKA4M2S00001BE)
- PMOD Expansion Board for RYZ012x1 (RTKYZ012A1B00000BE) Must be programmed with v5.4 Firmware or higher to support OTA firmware updates
- PC running Windows[®] 10
- 1 x Micro USB cable

PMOD[™] is registered to Digilent Inc.

Contents



1. Overview	4
1.1 Operating Environment	4
1.1.1 Hardware	4
1.1.2 Software	4
1.1.3 How to Assemble RYZ012 Module and EK-RA4M2	5
1.1.4 EK-RA4M2 USB Connection	6
1.1.5 Mobile Device OTA App Setup	6
1.2 RYZ012 Firmware Update Process	
2. Firmware Update Application	9
2.1 Importing the Application Project	9
2.1.1 Specify e ² studio Workspace	9
2.1.2 Import Project	9
2.1.3 Select Existing Project	
2.1.4 Select Project	10
2.2 Project Build and Download	11
2.3 Application Project Operation	11
2.3.1 Launch J-Link RTT Viewer	11
2.3.2 Start the Application Execution	12
2.3.3 RTT Viewer Logs	
2.4 RYZ012 Firmware Update Considerations	15
2.4.1 Firmware Update Issue #1	
2.4.2 Firmware Update Issue #2	16
2.4.3 Initialize BLE Driver Failed	17
3. Application Software Architecture	18
3.1 Microcontroller Peripheral Functions	19
3.2 FSP Modules	
4. Application Project Implementation	21
4.1 Entry Point	21
4.2 Main Loop	21
4.3 BLE Initialization Process	
4.4 Register BLE Callback Functions	23
4.5 Get Firmware Version Function	24
4.6 SPP BLE OTA Notify Events	
4.7 SPP BLE GAP Events	27
5. How to Make and Configure a New Project	27
5.1 Create a New Project	
5.2 BSP Heap and Stack Configuration	
5.3 Add the I/O Port Stack	
5.4 Add and Configure the SPP BLE Module	



RYZ012 and RA MCU

5.4.1	SPP BLE Abstraction Driver Configuration	29
5.4.2	Configure Peripherals for SPP BLE Abstraction Driver	30
5.5	Add General PWM Timers	34
5.6	Add RYZ012 BLE OTA Service Profile	35
5.7	Build and Modify Application Skeleton Code	38
6.	Next Steps	38
	Next Steps endix - Windows PC OTA App	



1. Overview

The RYZ012 BLE module is a highly integrated wireless communication module that provides a pre-certified solution for Bluetooth[®] 5.0 Low Energy (LE). The module is available in two configurations (RYZ012A1 and RYZ012B1) with or without a mounted antenna. Supported by the RA MCU family's Flexible Software Package (FSP) and the QE for BLE tool, customers can focus on application development without dealing with the details of Bluetooth LE.

Once an RYZ012 module is designed into an end customer application, designers will need to be able to update the firmware to adapt to changing conditions or requirements for the end systems. This update could be done via the host MCU or through a BLE radio OTA update. This application note covers an example of updating the RYZ012 module via BLE radio OTA including details of handling issues encountered when updating the BLE module.

The application project uses an EK-RA4M2 board connected with an RYZ012 module. Communication between the EK-RA4M2 and RYZ012 module is based on a command system called Serial Port Profile (here in after referred to as SPP).

Please check the SPP Bluetooth Low Energy Abstraction with RYZ012 (rm_ble_abs_spp) part of the RA Flexible Software Package User's Manual (R11UM0155) for more information on the APIs and callback event limitations.

1.1 Operating Environment

1.1.1 Hardware

The hardware requirements used in the sample application are shown in the following table.

Table 1. Hardware Requirements

Hardware	Description
EK-RA4M2	RTK7EKA4M2S00001BE
PMOD Expansion Board for RYZ012x1	RTKYZ012A1B00000BE
	Programmed with SPP SDK v5.4 Firmware to support MCU based firmware updates
Windows [®] 10 PC	
1 x Micro USB Cable	EK-RA4M2 USB Debug J10 connecter (micro-B)

1.1.2 Software

The software requirements used in the sample application are shown in the following table.

Table 2. Software Requirements

Software	Version
e ² studio IDE	2023-04
GCC Compiler	10.3.1
Renesas FSP	4.4.0
QE for BLE [RA,RE,RX]	V1.6.0 for e ² studio IDE
SEGGER J-Link RTT Viewer	V7.60f
TelinkBleOTA app	v2.0.5 for iOS



1.1.3 How to Assemble RYZ012 Module and EK-RA4M2

This section describes how to assemble RYZ012 PMOD module and EK-RA4M2. The RYZ012 PMOD module and EK-RA4M2 are connected by one of 2 x 6 PMOD connectors. In this application note, the PMOD connector must be mounted as:

RYZ012 PMOD : CN1 > EK-RA4M2 : J26 PMOD1 (SPI / UART)

Connect CN1 of RYZ012 PMOD and J26 PMOD 1 Connector of EK-RA4M2.

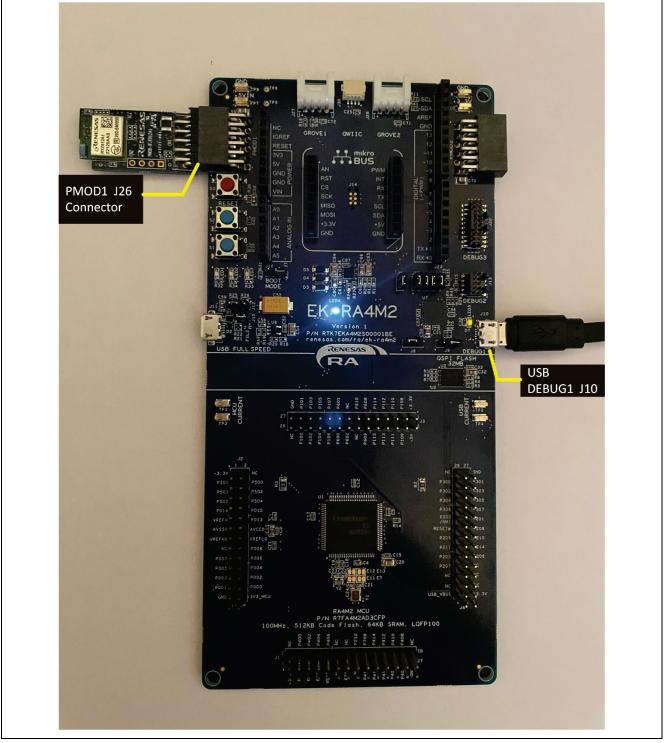


Figure 1. EK-RA4M2 and RYZ012 PMOD Assembly



1.1.4 EK-RA4M2 USB Connection

The USB Debug1 interface is used to communicate logging information with RTT Viewer.

Connect one micro-B USB cable to DEBUG1 (J10) with the other cable end connected to a PC port. See Figure 1.

1.1.5 Mobile Device OTA App Setup

In this app note, the firmware OTA Upgrade is demonstrated with an OTA App running on iOS devices. A basic overview of the mobile app install, and setup is provided here but is not supported by Renesas Support. Please contact Apple or the **TelinkBleOTA** app developer for support with mobile device or software installation issues. Make sure Bluetooth is enabled on the mobile device.

- Install the mobile app **TelinkBleOTA** from the iOS App Store. In the App Store, search for "Telink" to get a list of apps available and install the **TelinkBleOTA** app. See Figure 2.
- Locate the RYZ012 firmware bin files provided in the e² studio project workspace directory bleapp_fwupdate_ryz012_ra4m2_baremetal/8258_moduleV5_4.bin and
- •

- 8258_moduleV5_5.bin
- Transfer the RYZ012 firmware files by email or other method that can be accessed by the mobile device and store to app folder Locations->On My iPhone->TelinkBleOTA. The folder is created by the app during installation for newer versions (v16.3.1 or later) of iOS. If the folder is not created, then manually create the folder TelinkBleOTA on the mobile device. See Figure 3.

After the OTA App is installed on the iOS Device, skip to section 1.2 **RYZ012 Firmware Update Process** to learn about the OTA upgrade process and how to build and program the application project.

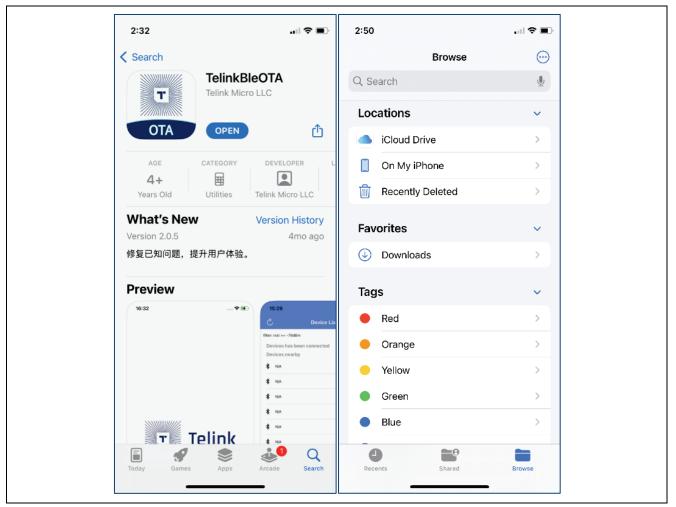


Figure 2. Install App and Load the RYZ012 Firmware Files



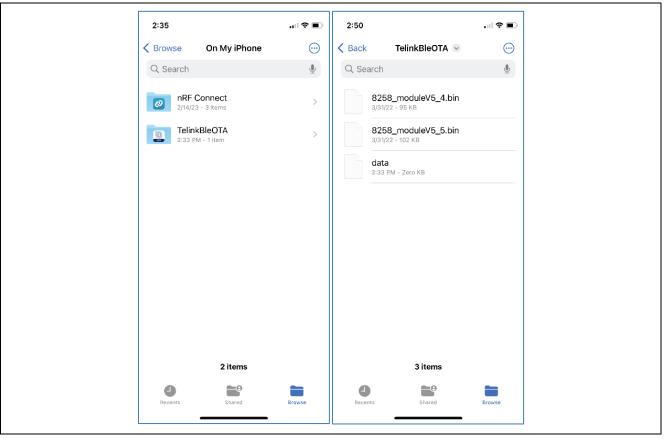


Figure 3. Store the RYZ012 Firmware Files

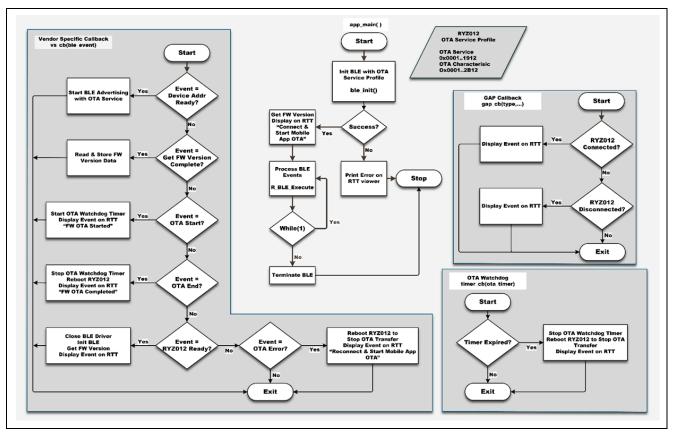


Figure 4. MCU User Application Flow Diagram



1.2 RYZ012 Firmware Update Process

In this application note, the RYZ012 module firmware is upgraded by BLE Radio OTA with a Mobile Device. The RYZ012 PMOD is connected to the EK-RA4M2 board. Only the RYZ012 firmware is upgraded and the host MCU is not upgraded.

The RYZ012 firmware files to use are supplied in the accompanying application project. See the root directory of the project that is imported into e² studio. See **Section 1.1.5 Mobile Device OTA App Setup** for more details. Instructions on how to import, build, and run the application project are provided in **Section 2**, **Firmware Update Application**.

The firmware image file that is used to upgrade the RYZ012 PMOD is transferred to the Mobile Device and opened by the OTA application. The OTA app scans for the custom OTA Service running on the RYZ012 and then connects to the RYZ012 module via BLE. The user selects the firmware file in the OTA app and then initiates the RYZ012 upgrade process. The upgrade process runs until the firmware file is fully transferred to the module via a BLE connection. The status of the upgrade process is monitored by the MCU and shown on the RTT Viewer.

After the RYZ012 is updated, the MCU restarts the RYZ012 which then runs the new image. Finally, the MCU reads the RYZ012 firmware version from the module and displays it in RTT Viewer.

The steps of the firmware upgrade process are shown in Figure 4 MCU User Application Flow Diagram.

Error! Reference source not found.4 shows the MCU user application (**project/qe_gen/ble/app_main.c**) flow diagram for the firmware update process. The FSP BLE SPP provided Vendor Specific Call back **vs_cb(ble_event)** simplifies the work required to perform the upgrade.

In this application note, the MCU is a monitor of the upgrade process and receives OTA notification events from the RYZ012 that are handled by the Vendor Specific Call back **vs_cb(ble_event)**.

Because the MCU is monitoring, it can intervene when it detects that an issue has occurred with the RYZ012 and Mobile App. We will cover the possible issues in section 2.4, RYZ012 Firmware Update Considerations.

- Note that the RYZ012 must receive the SPP_CMD_REBOOT_BLE command sent by R_BLE_VS_RestartModule(), else the firmware upgrade will not fully complete to run the new version firmware image. It will run the roll-back image, which is the version that ran prior to starting the upgrade.
- All firmware image frames must be sent by the Mobile Device OTA App for the transferred firmware image to pass the integrity check. If any of the data frames are missing or corrupted during the transfer process, the new image will fail integrity check and the RYZ012 will run the roll-back image instead after SPP_CMD_REBOOT_BLE is issued.
- If the RYZ012 is reset or loses BLE connection with the Mobile Device during the OTA process prior to completing all the update steps, then the RYZ012 will run the roll back image.
- If the RYZ012 loses power during the OTA process prior to completing all the update steps, then the RYZ012 will run the roll back image when power is restored.

To start using the application project immediately, see section 2, Firmware Update Application.

The design details of the application software architecture are covered in section 3, Application Software Architecture.

The implementation details of the application software are covered in section 4, Application Project Implementation.



2. Firmware Update Application

2.1 Importing the Application Project

The steps to import the application project into e² studio are shown in the following sections.

2.1.1 Specify e² studio Workspace

Launch e² studio, specify the workspace directory, and click the **Launch** button.

📴 e² studio Launcher	×
Select a directory as workspace e ² studio uses the workspace directory to store its preferences and development artifacts.	
Workspace: ⁰ C:¥MyWorkspace	✓ Browse
Use this as the default and do not ask again	
Recent Workspaces	
Launch	Cancel

Figure 5. e² studio Workspace

2.1.2 Import Project

Select **File > Import** from the menu bar.

6	MyWo	orkspace	e - e²	studio					
File	Edit	Navig	jate	Search	P	Project	Re	enesas View	s R
	New							Alt+Shift+I	<
	Open	File							
È,	Open	Project	ts fro	m File S	yste	em			
	Recen	nt Files							>
	Close	Editor						Ctrl+V	v
	Close	All Edit	ors				C	Ctrl+Shift+V	V
	Save							Ctrl+	s
	Save A	As							
Ð	Save /	All						Ctrl+Shift+	S
	Rever	t							
	Move								
	Renar	me						F	2
8	Refree	sh						F	5
	Conve	ert Line	Deli	miters To	D				>
Ð	Print.							Ctrl+	Р
\geq	Impor	rt		6					
4	Expor	rt		45					
	Prope	erties						Alt+Ente	er
	Switc	h Work	spac	e					>
	Resta								
	Exit								

Figure 6. Importing Project



2.1.3 Select Existing Project

Select Existing Projects into Workspace and click Next.

Import — Select Create new projects from an archive file or directory.		×
Select an import wizard: type filter text General CMSIS Pack CMSIS Pack CMSIS Projects into Workspace File System Preferences Projects from Folder or Archive Rename & Import Existing C/C++ Project into Workspace		~
(?) < <u>B</u> ack <u>N</u> ext > <u>F</u> inish	Cancel	

Figure 7. Selecting Existing Projects

2.1.4 Select Project

Choose **Select root directory**, click **Browse** and select the directory for the project to import. Check the box in **Projects**: window and click **Finish** to import the project. If importing from the zip file, then choose **Select archive file** instead and navigate to zip file to import.

Import – □ ×	
Import Projects	
Select a directory to search for existing Eclipse projects.	
Select root directory: bleapp_fwupdate_ryz012_ra4m2_baremetal Browse	
○ Select archive file: Browse	
Projects:	_
✓ bleapp_fwupdate_ryz012_ra4m2_baremetal (C:\Renesas\e2_studio\ Select All	
Deselect All	ī l
Refresh	
Options □ Search for nested projects ☑ Copy projects into workspace	
Close newly imported projects upon completion	
Hide projects that already exist in the workspace	
Working sets	
Add project to working sets Working sets: Verw	
? < Back Next > Finish Cancel	

Figure 8. Selecting Existing Projects



2.2 Project Build and Download

- 1. Select **Project > Build Project** from the menu bar or click the Build icon store to build the project.
- 2. Make sure that the hardware is connected according to section 1.1, Operating Environment. Click the debug icon 🔅 to launch the project. When the project starts, the application will be downloaded to the EK-RA4M2.

2.3 Application Project Operation

2.3.1 Launch J-Link RTT Viewer

Launch J-Link RTT Viewer, set as follows, and click the **OK** button.

- Connection to J-Link : USB
- Specify Target Device : R7FA4M2AD
- Target Interface & Speed : SWD, 4000 kHz
- RTT Control Block : Address, 0x2000050c

J-Link RTT Viewer V7.60f Configuration	×	
Connection to J-Link USB Serial No TCP/IP		
C Existing Session Specify Target Device R7FA4M2AD	~	
Script file (optional)		
	000 kHz 🔻	
○ Auto Detection ● Address ○ Sear Enter the address of the RTT Control block. Example: 0x20000000	rch Range	
0x2000050c	Cancel	

Figure 9. J-Link RTT Viewer Configuration

Note: The RTT Control Block Address can be found in the .bss._SEGGER_RTT section address of the map file generated in the Debug directory. If the application source code is modified with your custom changes, this address will change.

🎦 Project Explorer 🗙 📄 😫 🏹 🖇	🗖 🗖 📄 bleapp_fwup	date_ryz012_ra4m2_ba	remetal.map $ imes$	
v 😂 bleapp_fwupdate_ryz012_ra4m2_baremetal [Debug]	2000		0.000000000	ospi_device_i_non_recentive_si
> & Binaries	2634		0x10000000	ospi_device_1_region_max_size_
> 🔊 Includes	2635 2636		0x80030000 0x90030000	ospi_device_1_region_start_add
	2637		0x80030000	ospi_device_1_region_end_addre
> 🔑 qe_gen	2638		0X00020000	tz_OSPI_DEVICE_1_N =ospi_de
> 🔑 ra	2639	.noinit	0x200004d4	0x1c
> 😕 ra_gen	2639	.noinit	0x200004d4 0x200004d4	0x1c . = ALIGN (0x4)
> 🔑 src	2641		0x200004d4	
V 🔁 Debug	2641	*(.noinit*)	0X20000404	noinit_start = .
	2643	.noinit	0x200004d4	<pre>0x1c ./ra/fsp/src/bsp/mcu/all/bsp clocks</pre>
> 🗁 qe_gen	2644	.noinit	0x200004d4	. = ALIGN (0x8)
> 🗁 ra	2645	*(.heap.*)	0120000410	ALIGN (8X8)
> 🗁 ra_gen	2646	(.neap.)	0x200004f0	noinit end = .
> 🗁 src	2647		0720000410	IOIIIIC_end = .
> 🏂 bleapp_fwupdate_ryz012_ra4m2_baremetal.elf - [arm/le]	2648	.bss	0x200004f0	0x1444
bleapp_fwupdate_ryz012_ra4m2_baremetal_IssueRepro.zip	2649		0x200004f0	. = ALIGN (0x4)
	2650		0x200004f0	bss start = .
bleapp_fwupdate_ryz012_ra4m2_baremetal.elf.in	2651	*(.bss*)		
bleapp_fwupdate_ryz012_ra4m2_baremetal.hex	2652	.bss	0x200004f0	0x1c c:/users/a5134013/appdata/local/pro
bleapp_fwupdate_ryz012_ra4m2_baremetal.map	2653	.bss. SEGGER		
bleapp_fwupdate_ryz012_ra4m2_baremetal.rpd	2654		0x2000050c	0xa8 ./src/SEGGER RTT/SEGGER RTT.o
	0.000		0.0000050	

Figure 10. RTT Control Block Address



2.3.2 Start the Application Execution

Click the resume icon IP in the e² studio Debug Perspective to run the application.

RTT Viewer logging will show that the application is running, and 3 board LEDs (blue, green, red) will blink on and off to indicate that the program is running.

If the LEDs fail to blink and the white LED 4 is not lighted, then verify a USB cable is plugged into connector J10 (with the other end connected to PC). See section 1.1.4, EK-RA4M2 for more details.

In Figure 111, observe the instructions to start the firmware upgrade by connecting the Mobile Device OTA App to the RYZ012 and starting the Firmware OTA transfer.

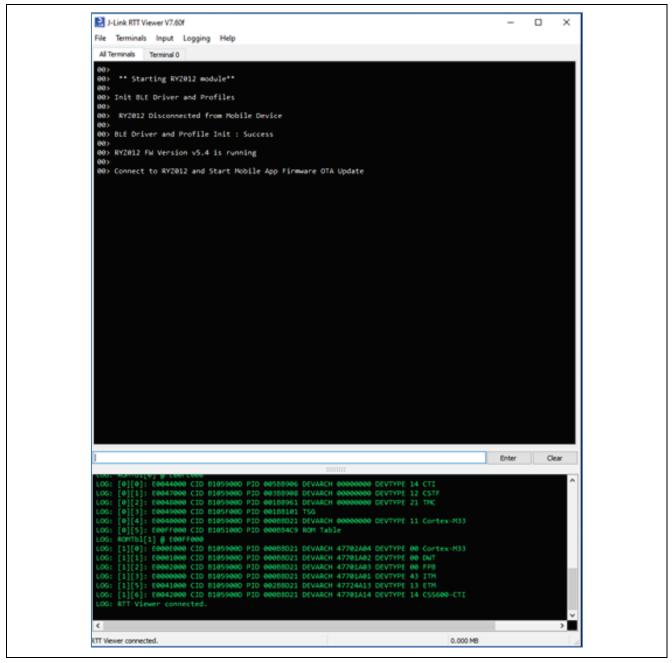


Figure 11. RTT Viewer- MCU App Start



1:23	l 🗢 🗩	1:14		al 🗢 🗈
C Device List	Filter	<	ΟΤΑ	
filter: rssi >= -100dBm		Name: RYZ012		
Apple TV	RSSI: -95 >	State: connected MTU: 244		DISCONNECT
Devices nearby	D99177 \	OTA Settings service: [use default(191	011	
RYZ012	RSSI: -58 >	characteristic: [use default(191 characteristic: [use defau read interval: 0 write interval: 0(ms)	2)] ult(2B12)]	5
≵ N/A	RSSI: -95 >	file path: 8258_moduleV Set Firmware Index: false protocol: Legacy	5_5.bin	
≵ N/A	RSSI: -94 >			
≵ N/A	RSSI: -82 >			
≵ N/A	RSSI: -94 >			
≵ N/A	RSSI: -97 >			
≵ N/A	RSSI: -97 >			
≵ N/A	RSSI: -93 >			
* rnet	RSSI: -96 >			
≵ N/A	RSSI: -84 >	Progress: 0%		
Q Search	Settings		START	
	-			-

Figure 12. Mobile App – Connect to RYZ012

Using the Mobile Device OTA App, scan for "RYZ012" module that is advertising and connect to the RYZ012. In the OTA App, enter the OTA Settings menu to select the RYZ012 firmware file that was previously loaded to the Mobile Device. See Figure 12.

See section 1.1.5, EK-RA4M2 Setup for more information on loading firmware files on the Mobile Device.

1:14			1:13		
<	OTA Settings	Reset	<	Choose Bin File	()
Service:	service uuid(Default: 1912)	~	Last Chose: 8	3258_moduleV5_4.bin	
Characteristi	c: characteristic uuid(Default: 2B12)	~	8267_module	_tOTA.bin (local), pid:0x0301 vid:0	×0000
	(0 as no-read): 0	•	8258_demo.b	in (local), pid:0x0000 vid:0x0000	
Write Interva	(0 as no-delay): 0(ms)		8267_module	tBase.bin (local), pid:0x0301 vid:	0x0000
Bin file path:			8258_module	V5_5.bin (iTunes), pid:0x0000 vid:	0x0000
8258_modu		>	8258_module	V5_4.bin (iTunes), pid:0x0000 vid:	0×0000
Firmware Ind	Firmware Index ex(HEX):				
00 Protocol:					
🔘 Leg	acy 🔿 Extend	I			
	SAVE				
	SAVE				

Figure 13. Mobile App – Choose OTA Settings to Select Firmware File

Once the firmware file is selected, press Save to store the OTA Settings. See Figure 13.

Press the Start button on the OTA App. The firmware file will start transferring to the RYZ012. See Figure 14.



1:14		al 🗢 💷	1:15		
<	ΟΤΑ		<	ΟΤΑ	
Name: RYZ012			Name: RYZ012		
State: connected		DISCONNECT	State: disconnected		CONNECT
MTU: 244			MTU: unknown		
OTA Settings service: [use defa characteristic: [us read interval: 0 write interval: 0 write interval: 0 set Firmwar protocol: Legacy	ult(1912)] e default(2B12)])	OTA Settings service: [use default(1: characteristic: [use de read interval: 0 write interval: 0(ms) file path: 8258_modult Set Firmware Index: fail protocol: Legacy		>
			start OTA OTA furand OTA success time: 31.355		
Progress: 0%			Progress: 100%		
	START			START	
		-	_		•

Figure 14. Mobile App – Start OTA Transfer & OTA Success

2.3.3 RTT Viewer Logs

At this point, move your attention to RTT Viewer to monitor the firmware upgrade status. You should observe logging similar to Figure 155.

RTT Viewer will show the results of the RYZ012 firmware OTA upgrade process.

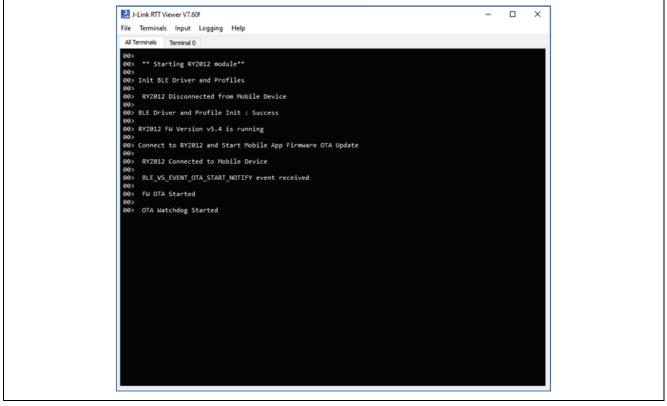


Figure 15. RTT Viewer – RYZ012 OTA Started



When the RYZ012 firmware upgrade completes with success, RTT Viewer will show logging as shown in Figure 166.

J-Link RTT Viewer V7.60f	-	×
File Terminals Input Logging Help		
All Terminals Terminal 0		
00> ** Starting RYZ012 module** 00>		
00> Init BLE Driver and Profiles 00>		
00> RYZ012 Disconnected from Mobile Device 00>		
00> BLE Driver and Profile Init : Success 00>		
00> RYZ012 FW Version v5.4 is running 00>		
00> Connect to RYZ012 and Start Mobile App Firmware OTA Update 00>		
00> RYZ012 Connected to Mobile Device 00>		
00> BLE_VS_EVENT_OTA_START_NOTIFY event received		
00> 00> FW OTA Started		
00> OTA Watchdog Started		
00> 00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received		
00> 00> BLE_VS_EVENT_OTA_END_NOTIFY event received		
00> FW OTA Completed		
00> 00> READY -> BLE_VS_EVENT_MODULE_READY_COMP event received		
00> 00> RYZ012 Disconnected from Mobile Device		
88>		
00> RYZ012 FW Version v5.5 is running		
		-

Figure 16. RTT Viewer – RYZ012 Firmware OTA Update Success

Notice that the RYZ012 firmware version was originally v5.4 before the OTA upgrade, and then after upgrade completes, is running at v5.5.

To verify that the RYZ012 is functioning use the OTA app on the mobile device and scan for BLE devices. The RYZ012 module will resume advertising again as "RYZ012" when the firmware upgrade has completed.

Currently, there is a known issue at 30 seconds after starting any OTA upgrade session where one **BLE_VS_EVENT_OTA_ERROR_NOTIFY** event is sent to the MCU by the RYZ012. This occurs even when the OTA is successful. If more than one **BLE_VS_EVENT_OTA_ERROR_NOTIFY** event is sent, then it is assumed that too many packet errors have occurred in the transfer between the Mobile Device OTA app and RYZ012. The transfer will be stopped by the MCU by restarting the RYZ012. The upgrade can be restarted by connecting the OTA app to the RYZ012 and starting the firmware upgrade.

In the event that the firmware upgrade fails with error messages shown in RTT Viewer, the next section 2.4, RYZ012 Firmware Update Considerations will help you to troubleshoot and resolve the issues.

2.4 RYZ012 Firmware Update Considerations

In the event that the upgrade process fails, see the following information to help troubleshoot your issue.

- First verify that the EK-RA4M2 and RYZ012 PMOD have been setup and connected correctly. Refer to section 1.1, Operating Environment.
- Next, verify the RYZ012 firmware image transferred is the one of two available images included in the app project.

The firmware images are located in the e² studio project workspace at the project directory

bleapp_fwupdate_ryz012_ra4m2_baremetal/8258_moduleV5_4.bin or

bleapp_fwupdate_ryz012_ra4m2_baremetal/8258_moduleV5_5.bin:

- The RYZ012 has built-in verification steps to recover from an incorrect or failing firmware update.
- The RYZ012 attempts to revert to the previous image if the update fails or halts.



- Once a firmware upgrade succeeds, the user must re-run the entire update process to downgrade to a previous firmware version.
- Review the troubleshooting section below for solutions to the most common issues. For all other issues that cannot be resolved in this section, please check the FAQs, search the Knowledge Base, or submit a support ticket at <u>Renesas Support</u>

2.4.1 Firmware Update Issue #1

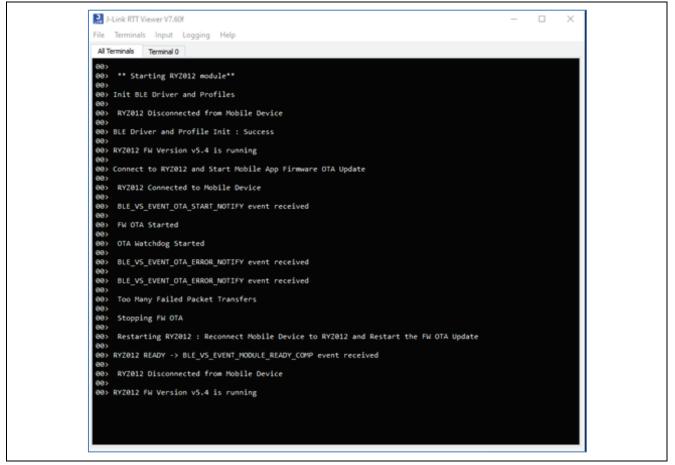


Figure 17. RTT Viewer – RYZ012 Firmware Packet Transfers Failed

In Figure 177, notice that the RYZ012 firmware version is v5.4 before the upgrade starts and is at v5.4 after the upgrade fails. This indicates that not all firmware packets were received during the transfer to the RYZ012. In this case, the integrity check failed on the RYZ012 before programming the memory in the RYZ012. Since the image failed integrity check, the old image v5.4 (roll back image) was retained and selected by the RYZ012 to execute.

Make sure that the EK-RA4M2 board remains powered during the file OTA transfer and write to RYZ012. If the EK-RA4M2 board is reset during the firmware OTA upgrade then the file integrity check will fail on the OTA transfer and the RYZ012 will run the last known good image version (roll back image).

2.4.2 Firmware Update Issue #2

For the case that the OTA App disconnects from the RYZ012 and does not complete sending the entire firmware file, the MCU based OTA Watchdog will expire and restart the RYZ012 module so that the Mobile Device can re-connect.

The OTA Watchdog expires 45 seconds after the start of the OTA transfer indicated by event BLE_VS_EVENT_OTA_START_NOTIFY. The OTA Watchdog time was chosen based on the OTA app fastest transfer rate and the current size of the firmware image. If the RYZ012 file size increases or a slower transfer rate is used then consider increasing the OTA Watchdog expiration time for you application. See **#define** BLE_APP_OTA_TRANSFER_FAST in /src/common_init.h

If the OTA App cannot re-connect to the RYZ012 after the OTA is started and disconnect occurs, wait for the OTA Watchdog to reboot and reset the RYZ012.



Figure 18 shows the issue where the Mobile Device disconnected after OTA start. See the two events BLE_VS_EVENT_OTA_ERROR_NOTIFY that were received by the MCU after the OTA Watchdog Started. Seconds later the OTA Timed-Out because the OTA Transfer did not complete which should have been indicated by event BLE_VS_EVENT_OTA_END_NOTIFY.

P.Link RTT Viewer V7.60f — … </th <th></th> <th></th> <th> </th>			
All Terminals Terminal 0 000 ** Starting RYZ012 module** 000 Init BLE Driver and Profiles 000 RYZ012 Disconnected from Mobile Device 000 RYZ012 Disconnected from Mobile Device 000 RYZ012 FW Version v5.4 is running 000 RYZ012 FW Version v5.4 is running 000 RYZ012 Connected to Mobile Device 000 RYZ012 Connected to Mobile Device 000 RYZ012 Connected to Mobile Device 000 RV2012 Connected to Mobile Device 000 RV2012 Connected to Mobile Device 000 RV2012 Connected to Mobile Device 001 RV2012 Connected to Mobile Device 002 RV2012 Connected to Mobile Device 003 RV2012 Connected to Mobile Pevice 004 FW 0TA Started 005 FW 0TA Started 006 BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 007 OTA Watchdog Started 008 BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 007 OTA Timed Out 008 Stopping FW OTA 009 Restarting RY2012 : Reconnect Mobile Device to RY2012 and Restart the FW OTA	🔜 J-Link RTT Viewer V7.60f	-	Х
<pre>%** Starting RYZ012 module** % % % % % % % % % % % % % % % % % %</pre>	File Terminals Input Logging Help		
<pre>60 ** Starting RYZ012 module** 60 60 Init BLE Driver and Profiles 60 60 BLE Driver and Profile Init : Success 60 60 BLE Driver and Profile Init : Success 60 60 Connect to RYZ012 and Start Mobile App Firmware OTA Update 60 60 Connect to RYZ012 and Start Mobile App Firmware OTA Update 60 60 BLE_VS_EVENT_OTA_START_NOTIFY event received 60 61 OTA Watchdog Started 62 63 BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 64 65 OTA Watchdog Started 66 OTA Timed Out 66 OTA Timed Out 67 Stopping FW OTA 66 Restarting RYZ012 : Reconnect Mobile Device to RYZ012 and Restart the FW OTA Update 66 RYZ012 READY -> BLE_VS_EVENT_MODULE_READY_COMP event received 67 RYZ012 Disconnected from Mobile Device 68 PR OTA Started 69 RYZ012 EXEMPT -> BLE_VS_EVENT_MODULE_READY_COMP event received 69 RYZ012 Disconnected from Mobile Device 69 RYZ012 Disconnected from Mobile Device 60 RYZ012 Disconnected from Mobile Device 61 PR PR</pre>	All Terminals Terminal 0		
<pre>000 000 First BLE Driver and Profiles 000 RYZ012 Disconnected from Mobile Device 000 000 BLE Driver and Profile Init : Success 000 000 RYZ012 FW Version v5.4 is running 000 Connect to RYZ012 and Start Mobile App Firmware OTA Update 000 000 RYZ012 Connected to Mobile Device 001 002 RYZ012 Connected to Mobile Device 002 003 BLE_VS_EVENT_OTA_START_NOTIFY event received 003 004 OTA Started 005 005 BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 005 005 OTA Timed Out 006 005 Stopping FW OTA 007 005 Restarting RYZ012 : Reconnect Mobile Device to RYZ012 and Restart the FW OTA Update 002 003 RYZ012 Disconnected from Mobile Device 003 004 RYZ012 Disconnected from Mobile Device</pre>	00>		
00> Init BLE Driver and Profiles 00> RYZ012 Disconnected from Mobile Device 00> BLE Driver and Profile Init : Success 00> RYZ012 FW Version v5.4 is running 00> Connect to RYZ012 and Start Mobile App Firmware OTA Update 00> RYZ012 Connected to Mobile Device 00> BLE_VS_EVENT_OTA_START_NOTIFY event received 00> FW OTA Started 00> OTA Watchdog Started 00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 00> OTA Timed Out 00> Stopping FW OTA 00> Restarting RYZ012 : Reconnect Mobile Device to RYZ012 and Restart the FW OTA Update 00> RYZ012 Disconnected from Mobile Device			
00> RYZ012 Disconnected from Mobile Device 00> BLE Driver and Profile Init : Success 00> RYZ012 FW Version v5.4 is running 00> Connect to RYZ012 and Start Mobile App Firmware OTA Update 00> RYZ012 Connected to Mobile Device 00> RYZ012 Connected to Mobile Device 00> BLE_VS_EVENT_OTA_START_NOTIFY event received 00> FW OTA Started 00> OTA Watchdog Started 00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 00> OTA Timed Out 00> Stopping FW OTA 00> RYZ012 READY -> BLE_VS_EVENT_MODULE_READY_COMP event received 00> RYZ012 Disconnected from Mobile Device	00> Init BLE Driver and Profiles		
00> BLE Driver and Profile Init : Success 00> RYZ012 FW Version v5.4 is running 00> RYZ012 and Start Mobile App Firmware OTA Update 00> RYZ012 Connected to Mobile Device 00> RYZ012 Connected to Mobile Device 00> BLE_VS_EVENT_OTA_START_NOTIFY event received 00> FW OTA Started 00> OTA Watchdog Started 00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 00> OTA Timed Out 00> Stopping FW OTA 00> Restarting RYZ012 : Reconnect Mobile Device to RYZ012 and Restart the FW OTA Update 00> RYZ012 READY -> BLE_VS_EVENT_MODULE_READY_COMP event received 00> RYZ012 Disconnected from Mobile Device			
<pre>000 000 RYZ012 FW Version v5.4 is running 000 000 Connect to RYZ012 and Start Mobile App Firmware OTA Update 000 000 RYZ012 Connected to Mobile Device 000 000 BLE_VS_EVENT_OTA_START_NOTIFY event received 000 000 FW OTA Started 000 000 OTA Watchdog Started 000 000 BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 000 000 BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 000 000 OTA Timed Out 000 000 Stopping FW OTA 000 000 Restarting RYZ012 : Reconnect Mobile Device to RYZ012 and Restart the FW OTA Update 000 000 RYZ012 READY -> BLE_VS_EVENT_MODULE_READY_COMP event received 000 000 RYZ012 Disconnected from Mobile Device 000 000 RYZ012 Disconnected from Mobile Device</pre>			
00> 0			
00> Connect to RYZ012 and Start Mobile App Firmware OTA Update 00> 00> 00> 01> 02> 03> 04> 05> 06> 07 08> 09> 09> 09> 09> 09> 09> 09> 09> 09> 09> 09> 09> 09> 09> 09> 09> 09> 09> 00> 00> 00> 01> 02> 03> 04> 05> 06> 07 08> 09> 09> 09> 09> 09> 09> 09> 09> 09> 09> 09>			
00> RYZ012 Connected to Mobile Device 00> BLE_VS_EVENT_OTA_START_NOTIFY event received 00> FW OTA Started 00> OTA Watchdog Started 00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 00> OTA Timed Out 00> Stopping FW OTA 00> Restarting RYZ012 : Reconnect Mobile Device to RYZ012 and Restart the FW OTA Update 00> RYZ012 READY -> BLE_VS_EVENT_MODULE_READY_COMP event received 00> RYZ012 Disconnected from Mobile Device	00> Connect to RYZ012 and Start Mobile App Firmware OTA Update		
00> BLE_VS_EVENT_OTA_START_NOTIFY event received 00> FW OTA Started 00> OTA Watchdog Started 00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 00> OTA Timed Out 00> Stopping FW OTA 00> Restarting RYZ012 : Reconnect Mobile Device to RYZ012 and Restart the FW OTA Update 00> RYZ012 READY -> BLE_VS_EVENT_MODULE_READY_COMP event received 00> RYZ012 Disconnected from Mobile Device			
00> FW OTA Started 00> OTA Watchdog Started 00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 00> OTA Timed Out 00> Stopping FW OTA 00> Restarting RYZ012 : Reconnect Mobile Device to RYZ012 and Restart the FW OTA Update 00> RYZ012 READY -> BLE_VS_EVENT_MODULE_READY_COMP event received 00> RYZ012 Disconnected from Mobile Device			
00> 00> 00> 00> 00> 00> 00> 00>	00>		
00> OTA Watchdog Started 00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 00> OTA Timed Out 00> Stopping FW OTA 00> Restarting RYZ012 : Reconnect Mobile Device to RYZ012 and Restart the FW OTA Update 00> RYZ012 READY -> BLE_VS_EVENT_MODULE_READY_COMP event received 00> RYZ012 Disconnected from Mobile Device			
00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 00> OTA Timed Out 00> Stopping FW OTA 00> Stopping FW OTA 00> Restarting RYZ012 : Reconnect Mobile Device to RYZ012 and Restart the FW OTA Update 00> RYZ012 READY -> BLE_VS_EVENT_MODULE_READY_COMP event received 00> RYZ012 Disconnected from Mobile Device			
00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received 00> 00> OTA Timed Out 00> 00> Stopping FW OTA 00> 00> Restarting RYZ012 : Reconnect Mobile Device to RYZ012 and Restart the FW OTA Update 00> 00> RYZ012 READY -> BLE_VS_EVENT_MODULE_READY_COMP event received 00> 00> RYZ012 Disconnected from Mobile Device 00>	<pre>00> BLE_VS_EVENT_OTA_ERROR_NOTIFY event received</pre>		
00> 00> OTA Timed Out 00> 00> Stopping FW OTA 00> 00> Restarting RYZ012 : Reconnect Mobile Device to RYZ012 and Restart the FW OTA Update 00> 00> RYZ012 READY -> BLE_VS_EVENT_MODULE_READY_COMP event received 00> 00> RYZ012 Disconnected from Mobile Device 00>			
00> 00> Stopping FW OTA 00> 00> Restarting RYZ012 : Reconnect Mobile Device to RYZ012 and Restart the FW OTA Update 00> 00> RYZ012 READY -> BLE_VS_EVENT_MODULE_READY_COMP event received 00> 00> RYZ012 Disconnected from Mobile Device 00>	00>		
00> 00> Restarting RYZ012 : Reconnect Mobile Device to RYZ012 and Restart the FW OTA Update 00> 00> RYZ012 READY -> BLE_VS_EVENT_MODULE_READY_COMP event received 00> 00> RYZ012 Disconnected from Mobile Device 00>	00>		
00> Restarting RYZ012 : Reconnect Mobile Device to RYZ012 and Restart the FW OTA Update 00> 00> RYZ012 READY -> BLE_VS_EVENT_MODULE_READY_COMP event received 00> RYZ012 Disconnected from Mobile Device 00>			
00> RYZ012 READY -> BLE_VS_EVENT_MODULE_READY_COMP event received 00> 00> RYZ012 Disconnected from Mobile Device 00>			
00> RYZ012 Disconnected from Mobile Device 00>	00> RYZ012 READY -> BLE_VS_EVENT_MODULE_READY_COMP event received		
90>			
00> KY2012 FW Version V5.4 is running	00>		
	00> RY2012 FW Version v5.4 is running		

Figure 18. RTT Viewer – Mobile Device Disconnected



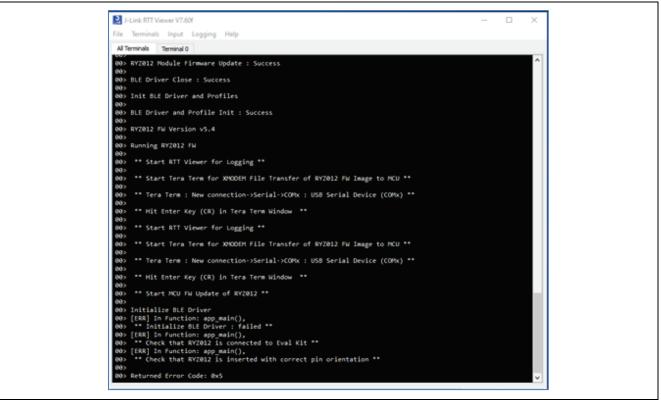


Figure 19. Init BLE Driver Failed



If you receive the **Initialize BLE Driver: failed** message then check that the RYZ012 PMOD is installed correctly in PMOD1 connector (J26) on the EK-RA4M2. Make sure the PMOD Mode Select GPIO pin is set for the correct initial logic state for the SPI or UART mode that you are using. See section 6, Next Steps, step 1 to review the SPP BLE Module and PMODx configuration settings. Make certain the RYZ012 PMOD has a factory firmware image programmed before attempting module upgrade.

3. Application Software Architecture

The application software architecture for the BLE Radio OTA firmware upgrade project is covered in this section.

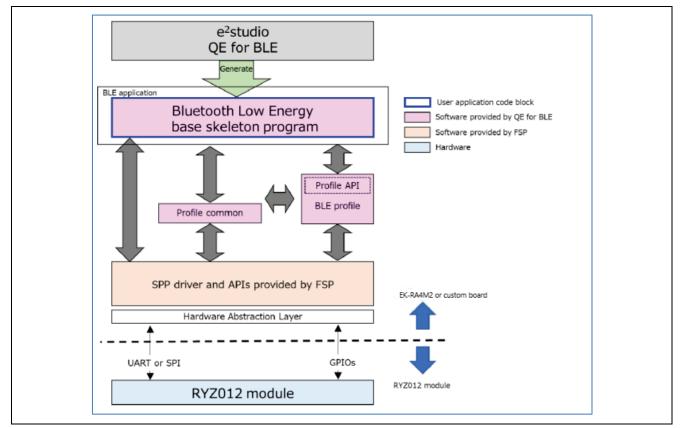


Figure 20. Application Software Architecture (Bare Metal)

Figure 20 shows the software architecture of a Bluetooth LE application in a BareMetal environment. The BLE Application performs initialization and BLE related processing. The QE for BLE tool generates C source code for the Bluetooth LE base skeleton program for the MCU application (located in project **ge_gen/ble/app_main.c**) and the BLE Profile. The base program is extended further for the specific functionality required in the user application. In this case, we are using the FSP BLE Abstraction APIs with SPP APIs to monitor the firmware update of the RYZ012 Module.

For more information on FSP APIs see **Renesas Flexible Software Package (FSP) User's Manual** (R11UM0155) and section on SPP Bluetooth Low Energy Abstraction with RYZ012 (rm_ble_abs_spp).

The FSP SPP APIs are used by the MCU to communicate with the RYZ012 module and monitor the Firmware OTA process with the Mobile Device. Figure 21 shows the software components of the application running on the RA4M2 MCU with a comms interface to the RYZ012. The user application and BLE Profile are customized to add features unique to your application.



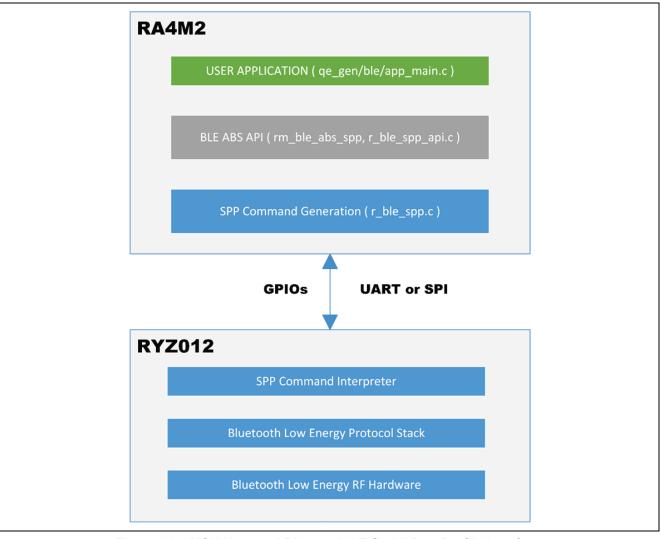


Figure 21. MCU Host and Bluetooth LE Serial Port Profile Interface

Figure 21 provides a more detailed look at the interface of the SPP and BLE APIs provided by FSP. The user application calls the BLE ABS APIs to interact with the RYZ012 BLE Module via the SPP Command Generation (SPP Driver) and SPP Command Interpreter residing on the RYZ012. The MCU communications interface to the RYZ012 PMOD can be UART or SPI. The accompanying software project uses the SPI interface.

The details of the update process are covered in section 1.2, RYZ012 Firmware Update Process.

The details of the application project software implementation are covered in section 4, Application Project Implementation.

3.1 Microcontroller Peripheral Functions

The microcontroller peripheral functions used in the application project are shown below.

Module	Pin	Description
Serial	SCI0(SPI0)	SPI communication with RYZ012 on PMOD1 (J26)
Communication		MOSI : P207
Interface		MISO : P206
		SCK : P400 SCLK
		SS : P401 Chip Select
		IRQ12 : P008 Interrupt Request
		RESET : P403
		MODE SELECT : P402 SPI Mode (High)

Table 3. Microcontroller peripheral functions



Module	Pin	Description
General-Purpose	GPT0	LED Blue PWM
Timer	GPT1	LED Red PWM
	GPT2	LED blink timer
	GPT3	LED Green PWM
I/O Port	P008	Interrupt Request RYZ012 PMOD1
	P403	Reset pin control of RYZ012 PMOD1
	P402	Mode Select (HIGH = SPI) of RYZ012 PMOD1
	P103	Blinker Timer PWM
	P415	User LED1 (Blue) PWM
	P404	User LED2 (Green) PWM
	P405	User LED3 (Red) PWM
External Interrupt Request	IRQ12	RYZ012 PMOD1 Interrupt Request

3.2 FSP Modules

The FSP modules used in the application project are shown below. See Figure 22, FSP Module Summary for a view of all the modules in the project.

Table 4. FSP Modules

Module Type	Module Name		Usage
System	I/O Port	r_ioport	GPIOs and LED indicators
BLE API	SPP BLE Abstraction	r_ble_abs_spp	RYZ012 BLE
Input	External IRQ	r_icu	RYZ012 PMOD Interrupt Req
Connectivity	SPI	r_sci_spi	RYZ012 PMOD comms
Timers	Timer	r_gpt	Blue, Red, Green Blink LED PWM

Note: This application program is a bare metal version.

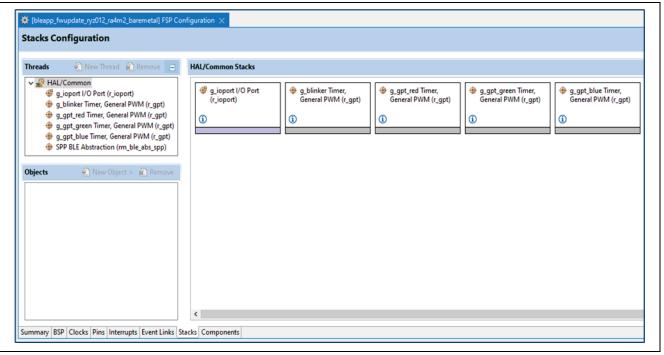


Figure 22. FSP Module Summary



acks Configuration				Generate Project Conter
kcks Configuration	HAL/Common Stacks (mer, VM (r,gpt)	General PWM (r_gpt)	(r_dtc) SCI4 T0	Generate Project Conte
	<			

4. Application Project Implementation

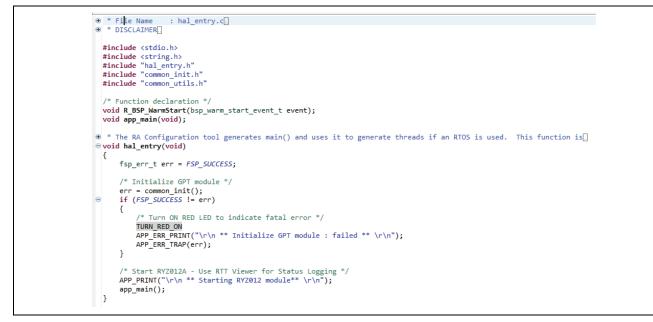
This section describes the application project implementation.

The firmware update is implemented in app_main.c. The *app_main()* includes BLE and system peripheral initialization and the implementation of the main loop.

When using QE for BLE tool, a minimal skeleton code of app_main.c is automatically generated which is customized with FSP APIs and helper functions to add the desired Bluetooth module update functionality.

4.1 Entry Point

In hal_entry.c the function hal_entry() initializes the GPT module for indicator LEDs and calls *app_main()* to perform the BLE module update as follows:



4.2 Main Loop

The *app_main()* includes the BLE and profile initialization, BLE module firmware update monitoring, and the main loop that processes the BLE events. See the source code below for *app_main()* with the implementation details. The steps of execution are:

1. Initialize the BLE driver and OTA Profile to start the communications with the RYZ012.

- 2. Display BLE driver and Profile initialization status on RTT Viewer
- 3. Get the current firmware version running on the RYZ012.
- 4. Display the RYZ012 firmware version on RTT Viewer.



- 5. Instruct user to connect BLE Device to RYZ012 and Start Mobile App OTA.
- 6. Start Watchdog Monitor and Log the status of the RYZ012 firmware update to RTT Viewer
- 7. Close the BLE driver and call ble_init() to re-start and initialize the BLE module and Profile, then run the new RYZ012 firmware.
- 8. Get the RYZ012 firmware version and display on RTT Viewer to confirm updated image is running.
- 9. Enter the main while loop to operate the RYZ012 and process BLE events.

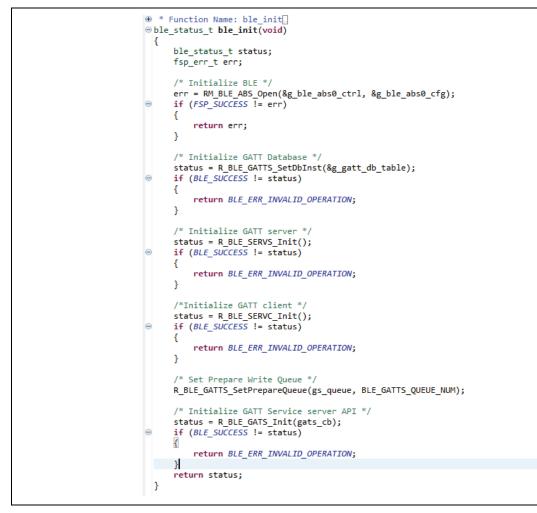
4.3 **BLE Initialization Process**

QE for BLE tool was used to create a basic BLE Profile with "RYZ012" set as the advertise name and supported OTA Service. From the tool, the source code of the *ble_init()* function is automatically generated and placed at the top of the app_main() functions. See below where the placement of **ble_init()** call must be relocated in the sequence after the firmware update process steps are completed.

```
⊖ void app_main(void)
 {
_{\odot}/* Hint: Input process that should be done before main loop such as calling initial function or variable definitions */
 /* Start user code for process before main loop. Do not edit comment generated here */
      ble_status_t status;
      APP_PRINT("\r\nInit BLE Driver and Profiles\r\n");
      /* Initialize BLE and profiles */
      status = ble_init();
Θ
      if (FSP_SUCCESS != status)
      {
          APP ERR PRINT("\r\n ** BLE Driver and Profiles Init : Failed ** \r\n");
          APP_ERR_TRAP(status);
      APP_PRINT("\r\nBLE Driver and Profile Init : Success\r\n");
      /* Get RYZ012 FW Version that is running */
      get_fw_version();
      APP_PRINT("\r\nConnect to RYZ012 and Start Mobile App Firmware OTA Update\r\n");
  /* End user code. Do not edit comment generated here *,
      /* main loop */
while (1)
      {
          /* Process BLE Event */
          R_BLE_Execute();
_{igodol} /* Hint: Input process that should be done during main loop such as calling processing functions */
 /* Start user code for process during main loop. Do not edit comment generated here *
          if (true == GPT_Is_Watchdog_Expired())
Θ
          {
               * monitor FW OTA Transfer watch dog
               * use RYZ012 SPP Vendor Specific Callback to handle OTA timeout
              vs_cb(BLE_VS_EVENT_OTA_ERROR_NOTIFY, BLE_ERR_RSP_TIMEOUT, NULL);
          3
  /* End user code. Do not edit comment generated here */
      }
_{igodol} /* Hint: Input process that should be done after main loop such as calling closing functions */
  /* Start user code for process after main loop. Do not edit comment generated here */
  /* End user code. Do not edit comment generated here */
      /* Terminate BLE */
      RM_BLE_ABS_Close(&g_ble_abs0_ctrl);
  }
```



See the details of the ble_init() function implementation below that initializes the BLE module, registers the callback functions for BLE, registers the GATT database, and any other additional services added with the QE for BLE tool during BLE Profile generation.



4.4 Register BLE Callback Functions

Registration of the callback functions are performed in **ble_init()** which are required to execute processing according to events from each layer such as GAP, GATT Server, GATT Client, and Profile Server API in the Bluetooth LE protocol stack. See **Bluetooth Low Energy Sample Application (R01AN6116EJ)** for more details of the callback registration process for RYZ012 and RA MCU and refer to **RA Flexible Software Package User's Manual (R11UM0155)** for more information on the type of callback events to handle.



4.5 Get Firmware Version Function

See the source code below for the details of utility function **get_fw_version(void)** which is used to read the firmware version that is currently running on the RYZ012 module. The function calls the FSP provided function **R_BLE_VS_GetFirmwareVersion()** and uses SPI to get the version response back from the module immediately parses the result for display on RTT Viewer.

```
/* Get the Firmware Version running on RYZ012 Module */

bool get_fw_version(void)

 {
       static r_ble_spp_payload_t payload_data;
       r_ble_spp_cmd_rsp_t ret_val = R_BLE_SPP_SUCCESS;
ble_status_t status;
       #define MIN_FW_VERSION_BYTES 3 /* Payload must have 3 or more bytes */
       status = R_BLE_VS_GetFirmwareVersion();
Θ
       if (BLE_SUCCESS != status)
       {
            APP_ERR_PRINT("\r\n ** Get RYZ012 FW Version : Failed ** \r\n");
            return false;
       }
       /* To get the FW version data immediately use R_BLE_SPP_SPI_Read(&payload_data)
        * otherwise, use Process BLE Event by calling R_BLE_Execute() and waiting
* for callback event BLE_VS_EVENT_GET_FW_VERSION_COMP to be received in vs_cb(...)
* vs_cb(uint16_t type, ble_status_t result, st_ble_vs_evt_data_t *p_data)
       R_BSP_SoftwareDelay(10, BSP_DELAY_UNITS_MILLISECONDS);
ret_val = R_BLE_SPP_SPI_Read(&payload_data);
if ( R_BLE_SPP_SUCCESS == ret_val )
            if ( ( BLE_VS_EVENT_GET_FW_VERSION_COMP == payload_data.event_id ) && ( MIN_FW_VERSION_BYTES <= payload_data.out_len ) )
            {
                 /* payload contains fw version data */
                 g_ble_version_data.major = payload_data.out_data[1];
g_ble_version_data.minor = payload_data.out_data[2];
                 APP_PRINT("\r\nRYZ012 FW Version v%d.%d \r\n", g_ble_version_data.major, g_ble_version_data.minor);
                 return true;
            }
            APP ERR PRINT("\r\n ** Get RYZ012 FW Version : Failed ** \r\n");
            return false:
       }
       APP PRINT("RYZ012 FW Version data read Error: %x\r\n", ret val);
       return false:
  }
```



4.6 SPP BLE OTA Notify Events

The FSP BLE SPP API provides a Vendor Specific Callback **vs_cb(type, result, ...)** that handles the OTA Notification Events received during the RYZ012 firmware upgrade process. The callback provides the ability for the MCU to monitor the RYZ012 firmware OTA process status. It is helpful for MCU intervention when issues arise during the Mobile Device firmware OTA transfer.

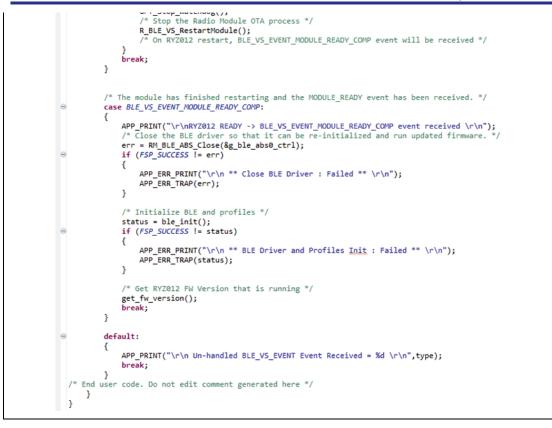
```
: st_ble_vs_evt_data_t *p_data -
Event parameters of Vendor Specific API.
    * Return Value : none

woid vs_cb(uint16_t type, ble_status_t result, st_ble_vs_evt_data_t *p_data)
   {
/* Hint: Input common process of callback function such as variable definitions */
   /* Start user code for vender specific callback function common process. Do not edit comment generated here */
         ble_status_t status;
   fsp_err_t err;
static uint8_t errors_ota;
/* End user code. Do not edit comment generated here */
         R_BLE_SERVS_VsCb(type, result, p_data);
         switch(type)
                case BLE VS EVENT GET ADDR COMP:
                       /* Start advertising when BD address is ready */
st_ble_vs_get_bd_addr_comp_evt_t * get_address = (st_ble_vs_get_bd_addr_comp_evt_t *)p_data->p_param;
memcpy(g_ble_advertising_parameter.own_bluetooth_address, get_address->addr.addr, BLE_BD_ADDR_LEN);
                       RM_BLE_ABS_StartLegacyAdvertising(&g_ble_abs0_ctrl, &g_ble_advertising_parameter);
                       break;
                3

@ /* Hint: Add cases of vender specific event macros defined as BLE_VS_XXX */
/* Start user code for vender specific callback function event process. Do not edit comment generated here */
@ case BLE_VS_EVENT_GET_FN_VERSION_COMP:

                       /* Received BLE Module FW Version info */
                      st_ble_vs_get_fw_version_comp_evt_t * get_version = (st_ble_vs_get_fw_version_comp_evt_t *)p_data->p_param;
g_ble_version_data.major = get_version->major;
g_ble_version_data.minor = get_version->minor;
break;
                3
               /* Handle BLE FW App OTA initiated FW update (Mobile app to RYZ012 Module) */
case BLE_VS_EVENT_OTA_START_NOTIFY :
                    /* Firmware update OTA process has started. */
APP_PRINT("\r\n BLE_VS_EVENT_OTA_START_NOTIFY event received \r\n");
APP_PRINT("\r\n FW OTA Started \r\n");
errors_ota = 0;
GPT_start_Watchdog();
APP_PRINT("\r\n OTA Watchdog Started \r\n");
herab:
                     break;
              }
               /* Handle BLE FW App OTA completed FW update (Mobile app to RYZ012 Module) */
case BLE_VS_EVENT_OTA_END_NOTIFY :
                    /* Firmware update OTA process has completed. */
APP_PRINT("\r\n BLE_VS_EVENT_OTA_END_NOTIFY event received \r\n");
APP_PRINT("\r\n FW OTA Completed \r\n");
errors_ota = 0;
GPT_Stop_Watchdog();
R_BLE_VS_RestartModule();
hereak:
                     break:
              }
              /* Handle BLE FW App OTA error during FW update (Mobile App to radio module) */ case BLE_VS_EVENT_OTA_ERROR_NOTIFY :
                                 reported during Firmware OTA process */
                     / croor reported auring tinmware OTA process */
APP_PRINT("\r\n BLE_VS_EVENT_OTA_ERROR_NOTIFY event received \r\n");
errors_Ota++;
                    errors_ota++;
/* Prevent In-operability of RYZ012 PMOD (Init BLE Driver and Profiles Init : Failed Returned Error Code 0x5)
which must be cleared by reprogramming FW with <u>Renesss</u> Module Programmer */
if ((errors_ota >= 2) || (true == GPT_Is_Watchdog_Expired()))
            /* Too many packet transfer errors reported by RYZ012 module
Trap the condition and disconnect the Mobile Device to stop FW OTA
else the RYZ012 will get "bricked" and will require to be programme
if (true == GPT_Is_Watchdog_Expired())
                                                                                                                              ammed with FW by <u>Renesas</u> Module Programmer */
                          {
                                APP PRINT("\r\n OTA Timed Out\r\n");
                          1
                           else
                                APP PRINT("\r\n Too Many Failed Packet Transfers\r\n");
                          / APP_PRINT("\r\n Stopping FW OTA \r\n");
APP_PRINT("\r\n Restarting RYZ012 : Reconnect Mobile Device to RYZ012 and Restart the FW OTA Update\r\n");
GPT_Stop_Watchdog();
/* Stop the Radio Module OTA process */
                           R BLE VS RestartModule():
```





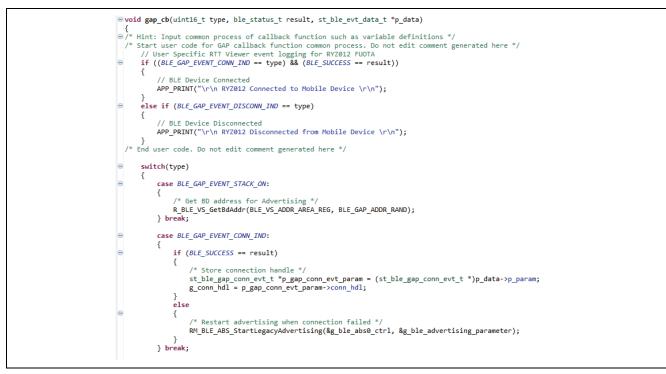
The steps of execution are:

- 1. Send command for RYZ012 to start Advertising "RYZ012" and indicate OTA Service is available.
- 2. The MCU will read the current running RYZ012 firmware version and display on RTT Viewer.
- 3. The RYZ012 waits for BLE Device connection and notifies the MCU when a device is connected.
- 4. When the BLE Device OTA app starts a firmware upgrade, the RYZ012 sends an BLE_VS_EVENT_OTA_START_NOTIFY event to the MCU which is displayed on RTT Viewer. The OTA Watchdog Timer is started to monitor the upgrade process.
- If errors are detected by the RYZ012 during the upgrade, <u>BLE_VS_EVENT_OTA_ERROR_NOTIFY</u> event is sent to MCU. This event can indicate a BLE Device disconnection, or failed firmware packet transfers. If more than two error events are detected, the OTA upgrade is halted by the MCU with an RYZ012 restart.
- 6. If the OTA Watchdog Timer expires, then the RYZ012 is restarted to stop the OTA upgrade and clear the error condition. Advertising is restarted on the RYZ012 after reboot is completed. The OTA app can now be re-connected to try the firmware upgrade again.
- 7. When the Device OTA app completes the firmware upgrade, the RYZ012 sends an <u>BLE_VS_EVENT_OTA_END_NOTIFY</u> event to the MCU which is displayed on RTT Viewer. The MCU restarts the RYZ012 module with R_BLE_VS_RestartModule() to run the new firmware.
- 8. After the RYZ012 reboot is completed, a device ready event *BLE_VS_EVENT_MODULE_READY_COMP* is sent by the RYZ012. The RYZ012 firmware version is then read by the MCU to validate the upgrade was successful.



4.7 SPP BLE GAP Events

The FSP BLE SPP API provides a GAP Callback **gap_cb(type, result, ...)** that handles the BLE GAP Events received during the RYZ012 firmware upgrade process. The callback provides the ability for the MCU to monitor the RYZ012 BLE connection status to the Mobile Device.



5. How to Make and Configure a New Project

This section describes the configuration to create an RA MCU based firmware OTA update application for the RYZ012. The application that is generated will include the system peripheral initialization, BLE Profile, and BLE API framework that will then need to be customized to include the RYZ012 module firmware update sequence. The update sequence is implemented by calling the FSP provided update and utility functions detailed in section 4, Application Project Implementation.

5.1 Create a New Project

Please see the **Renesas Flexible Software Package (FSP) User's Manual (R11UM0155EU)** for instructions on how to create a new project in **e**² **studio** for a BareMetal environment, choose **BareMetal – Minimal**. Follow the instructions to select the FSP Version, Board, Device, and Toolchains. Continue through all prompts until the project is created. Next the BSP Heap and Stack must be configured.

5.2 BSP Heap and Stack Configuration

Set heap and stack configuration as follows on the FSP configuration **BSP** tab. If the properties tab is not visible, choose **Window** > **Show View** > **Properties** on the e^2 studio menu bar.

- [RA Common] > [Main stack size (bytes)] : 0x4000
- [RA Common] > [Heap size (bytes)] : 0x1000

📳 Problems 📮 Console 🔲 Properties 🗙 ௸ Smart Browser 🐚 Stack Analysis 🤑 Smart Manual 🔗 Search 🕕 Memory 🖏 Progress	Summary BSP Clocks Pins Interrupts Event Links Stacks Components					
	🖹 Problems 📮 Console 🔲 Properties 🗙 鎟 Smart Browser 🐚 Stack Analysis 🤑 Smart Manual 🔗 Search 🕕 Memory 🖏 Progress					
EK-RA4M2						
Settings Property Value						
→ RA Common						
Main stack size (bytes) 0x4000	Main stack size (bytes) 0x4000					
Heap size (bytes) 0x1000	Heap size (bytes) 0x1000					
MCU Vcc (mV) 3300						

Figure 23. BSP Configuration



5.3 Add the I/O Port Stack

	Configuration			
Threads		🕘 New Thread 🔹 Remove 📄	g_ioport I/O Port (r_ioport) Stacl	G
	AL/Common g_ioport I/O Port (r_ioport) g_ghinker Timer, General PWM (r_gpt) g_gpt_red Timer, General PWM (r_gpt) g_gpt_blue Timer, General PWM (r_gpt) SPP BLE Abstraction (rm_ble_abs_spp)		 g_ioport I/O Port (r_ioport) i 	
Objects		🗿 New Object > 📓 Remove		
	BSP Clocks Pins Interrupts Event Link	s Stacks Components Smart Browser 📪 Smart Manual 🚺 Merr	nory 🔗 Search	
g_ioport	I/O Port (r_ioport)			
Settings API Info	Property ✓ Common			Value
API Into	Parameter Checking			Default (BSP)
	 Module g_ioport I/O Port (r_ioport) 			
	Name			g_ioport
	1st Port ELC Trigger Source			
				Disabled
	2nd Port ELC Trigger Source			Disabled
	3rd Port ELC Trigger Source			Disabled Disabled
	3rd Port ELC Trigger Source 4th Port ELC Trigger Source			Disabled Disabled Disabled
	3rd Port ELC Trigger Source 4th Port ELC Trigger Source Pin Configuration Name			Disabled Disabled
	3rd Port ELC Trigger Source 4th Port ELC Trigger Source Pin Configuration Name ✓ Pins			Disabled Disabled Disabled g_bsp_pin_cfg
	3rd Port ELC Trigger Source 4th Port ELC Trigger Source Pin Configuration Name V Pins TCK			Disabled Disabled Disabled g_bsp_pin_cfg P300
	3rd Port ELC Trigger Source 4th Port ELC Trigger Source Pin Configuration Name V Pins TCK TDI			Disabled Disabled Disabled g_bsp_pin_cfg P300 P110
	3rd Port ELC Trigger Source 4th Port ELC Trigger Source Pin Configuration Name Pins TCK TDI TDO			Disabled Disabled Disabled g_bsp_pin_cfg P300 P110 P109
	3rd Port ELC Trigger Source 4th Port ELC Trigger Source Pin Configuration Name V Pins TCK TDI TDO TMS			Disabled Disabled Disabled g_bsp_pin_cfg P300 P110 P109 P108
	3rd Port ELC Trigger Source 4th Port ELC Trigger Source Pin Configuration Name V Pins TCK TDI TDO TMS SWCLK			Disabled Disabled Disabled g_bsp_pin_cfg P300 P110 P109 P108 <unavailable></unavailable>
	3rd Port ELC Trigger Source 4th Port ELC Trigger Source Pin Configuration Name ✓ Pins TCK TDI TDO TMS SWCLK SWDIO			Disabled Disabled Disabled g_bsp_pin_cfg P300 P110 P109 P108 <unavailable></unavailable>
	3rd Port ELC Trigger Source 4th Port ELC Trigger Source Pin Configuration Name ✓ Pins TCK TDI TDO TMS SWCLK SWDIO TRACESWO			Disabled Disabled Disabled g_bsp_pin_cfg P300 P110 P109 P108 <unavailable> <unavailable></unavailable></unavailable>
	3rd Port ELC Trigger Source 4th Port ELC Trigger Source Pin Configuration Name V Pins TCK TDI TDO TMS SWCLK SWDIO TRACESWO TCLK			Disabled Disabled Disabled g_bsp_pin_cfg P300 P110 P109 P108 <unavailable> <unavailable> <unavailable></unavailable></unavailable></unavailable>
	3rd Port ELC Trigger Source 4th Port ELC Trigger Source Pin Configuration Name V Pins TCK TDI TDO TMS SWCLK SWDIO TRACESWO TCLK TDATA0			Disabled Disabled Disabled g_bsp_pin_cfg P300 P110 P109 P108 <unavailable> <unavailable> <unavailable> <unavailable></unavailable></unavailable></unavailable></unavailable>
	3rd Port ELC Trigger Source 4th Port ELC Trigger Source Pin Configuration Name ✓ Pins TCK TDI TDO TMS SWCLK SWDIO TRACESWO TCLK TDATA0 TDATA1			Disabled Disabled Disabled g_bsp_pin_cfg P300 P110 P109 P108 <unavailable> <unavailable> <unavailable> <unavailable> <unavailable></unavailable></unavailable></unavailable></unavailable></unavailable>
	3rd Port ELC Trigger Source 4th Port ELC Trigger Source Pin Configuration Name ✓ Pins TCK TDI TDO TMS SWCLK SWDIO TRACESWO TCLK TDATA0 TDATA1 TDATA2			Disabled Disabled Disabled g_bsp_pin_cfg P300 P110 P109 P108 <unavailable> <unavailable> <unavailable> <unavailable> <unavailable> <unavailable> <unavailable></unavailable></unavailable></unavailable></unavailable></unavailable></unavailable></unavailable>
	3rd Port ELC Trigger Source 4th Port ELC Trigger Source Pin Configuration Name ✓ Pins TCK TDI TDO TMS SWCLK SWDIO TRACESWO TCLK TDATA0 TDATA1 TDATA2 TDATA3			Disabled Disabled Disabled g_bsp_pin_cfg P300 P110 P109 P108 <unavailable> <unavailable> <unavailable> <unavailable> <unavailable> <unavailable> <unavailable> <unavailable></unavailable></unavailable></unavailable></unavailable></unavailable></unavailable></unavailable></unavailable>
	3rd Port ELC Trigger Source 4th Port ELC Trigger Source Pin Configuration Name ✓ Pins TCK TDI TDO TMS SWCLK SWDIO TRACESWO TCLK TDATA0 TDATA1 TDATA2			Disabled Disabled Disabled g_bsp_pin_cfg P300 P110 P109 P108 <unavailable> <unavailable> <unavailable> <unavailable> <unavailable> <unavailable> <unavailable></unavailable></unavailable></unavailable></unavailable></unavailable></unavailable></unavailable>

Figure 24. Adding I/O Stack

5.4 Add and Configure the SPP BLE Module

This section describes how to add and configure SPP BLE Abstraction Driver for communicating with the RYZ012 module in the application. Open **configuration.xml** in the project and add / configure **SPP BLE Abstraction Driver** on FSP configuration **Stacks** tab.

The procedure to add the SPP BLE Abstraction Driver is different for BareMetal and FreeRTOS environments. This document describes the procedure for BareMetal environment.

Add RYZ012 module in BareMetal environment

Click **New Stack** and add **Networking > SPP BLE Abstraction (rm_ble_abs_spp)** to **HAL/Common**. If the stack cannot be found use the **Search** box in e² studio since stack categories and organization may change in future FSP releases.



	fwupdate_ryz012_ra4m2_baremetal] FSP Conf				
STACKS CO	onfiguration				
Threads		🐑 New Thread 🔬 Remove 🕞	SPP BLE Abstraction (rm_ble	e_abs_spp) Stacks	
2 ⊕ 2 ⊕ 2 ⊕ 2 ⊕ 2 ⊕	L/Common g_ioport I/O Port (r_ioport) g_blinker Timer, General PWM (r_gpt) g_gpt_red Timer, General PWM (r_gpt) g_gpt_gene Timer, General PWM (r_gpt) g_gpt_blue Timer, General PWM (r_gpt) SPP BLE Abstraction (rm_ble_abs_spp)		 SPP BLE Abstraction (r RVZ012 SPP Driver Layer 3 	M_ble_abs_spp) BLE SPP Transport on SPI (rm_ble_ab	is_spp_transport)
				<pre> g_spi0 SPI (r_sci_spi) () </pre>	g_external irq0 External IRQ (r_icu)
Objects		🔊 New Object > 📓 Remove		(r_dtc) SCI4 TXI (r_dtc	I Transfer c) SCI4 RXI eived data full)
•	3SP Clocks Pins Interrupts Event Links Star s 📮 Console 🔲 Properties 🗙 🍑 Smar		iory 🛷 Search		
PP BLE A	Abstraction (rm_ble_abs_spp)				
Settings	Property Reset Pin UART/SPI Select Port (PB5) UART/SPI Select Pin (PB5) SPI Software SSL Port SPI Software SSL Pin Transmit Power Level (in dBm)			Value 03 04 02 04 01 4,57	
-	Module SPP BLE Abstraction (rm_ble_abs, General Name GAP callback Vendor specific callback	spp)		g_ble_abs0 gap_cb vs_cb	
-	GATT server callback parameter GATT server callback number GATT client callback number GATT client callback number			gs_abs_gatts_cb_param 2 gs_abs_gattc_cb_param 2	

Figure 25. SPP BLE ABS Module

5.4.1 SPP BLE Abstraction Driver Configuration

This section describes the SPP BLE Abstraction Driver configuration options and related modules. SPP BLE Abstraction Driver includes the following configuration for the **EK-RA4M2** and **SPI comms** interface to **PMOD1 with RYZ012**. Users can modify these configurations according to their own specific hardware requirements. See Figure 25, SPP BLE ABS Module for more details.

Configuration options	Comment
Reset port	Specify port number of Reset Pin
EK-RA4M2 case: 04	
Reset Pin	Specify pin number of Reset Pin
EK-RA4M2 case: 03	
UART / SPI Select Port (PB5)	Specify port number of PMOD1 Mode Select Pin.
EK-RA4M2 case: 04	
UART / SPI Select Pin (PB5)	Specify pin number of PMOD1 Mode Select Pin.
EK-RA4M2 case: 02	Configure mode as SPI
SPI Software SSL Port	Specify port number of SPI Slave Select (SS)
EK-RA4M2 case: 04	
SPI Software SSL Pin	Specify pin number of SPI Slave Select (SS)
EK-RA4M2 case: 01	
Transmit Power Level (in dBm)	Specify required transmit power level.
EK-RA4M2 case: 4.57dBm	



Gap callback	Do NOT change.
Default: gap_cb	
Vendor specific callback	Do NOT change.
Default: vs_cb	
GATT server callback parameter	Do NOT change.
Default: gs_abs_gatts_cb_param	
GATT server callback number	Do NOT change.
Default: 2	
GATT client callback parameter	Do NOT change.
Default: gs_abs_gattc_cb_param	
GATT client callback number	Do NOT change.
Default: 2	

5.4.2 Configure Peripherals for SPP BLE Abstraction Driver

The SPP BLE Abstraction Driver uses the SPI Driver on r_sci_spi to communicate with RYZ012 module attached to the RA4M2 with PMOD1 connector (J26). This section describes how to configure the SPI driver. See additional information in section 6, Next Steps for using the SPP BLE Driver with other communication interfaces such as UART.

1. Click g_spi0 SPI Driver on r_sci_spi.

Stacks	Configuration				O Generate Project Content
Threads	🐑 New Thread 🛛 🙀 Remove 📄	SPP BLE Abstraction (rm_ble	e_abs_spp) Stacks	🛃 New Stack >	🐣 Extend Stack > 🛛 👔 Remove
	AL/Common g_ioport I/O Port (r_ioport) g_blinker Timer, General PWM (r_gpt) g_gpt_cd Timer, General PWM (r_gpt) g_gpt_blue Timer, General PWM (r_gpt) g_gpt_blue Timer, General PWM (r_gpt) SPP BLE Abstraction (rm_ble_abs_spp) © New Object > Remove	 SPP BLE Abstraction (r RVZ012 SPP Driver Layer 3 	BLE SPP Transport on SPI (rm_ble) g_spi0 SPI (r_sci_spi) g_ g_spi0 SPI (r_sci_spi) g_rtansfer0 Transfer (r_dtc) SCI4 TXI (Transmit data empty) (R Transmit data empty)	abs_spp_transport)	↑ ¶ g.external.irq0 External IRQ (r_icu) ①
			<u>(</u>)		¥
noblen 🕄	BSP Clocks Pins Interrupts Event Links <u>Stacks</u> Co ns Console Properties X & Smart Browse PI (r_sci_spi)				× *
Problem	ns 📮 Console 🔲 Properties 🗙 🏟 Smart Browse	mponents		Value	``
Problem	ns Console Properties X Smart Browser	mponents			
Problem g_spi0 Si Settings	ns Console Properties X Smart Browsee PI (r_sci_spi) Property Common Parameter Checking	mponents		Default	(BSP)
Problem g_spi0 Si Settings	ns Console Properties X Smart Browsee PI (r_sci_spi) Property Common Parameter Checking DTC Support	mponents			(BSP)
Problem g_spi0 Si Settings	ns Console Properties X Smart Browsee PI (r_sci_spi) Property Common Parameter Checking DTC Support Module g_spi0 SPI (r_sci_spi)	mponents		Default Enabled	(BSP)
Problem g_spi0 Si Settings	ns Console Properties X Smart Browsee PI (r_sci_spi) Property Common Parameter Checking DTC Support Module g_spi0 SPI (r_sci_spi) Name	mponents		Default Enabled g_spi0	(BSP)
Problem g_spi0 Si Settings	ns Console Properties X Smart Browses PI (r_sci_spi) Property V Common Parameter Checking DTC Support V Module g_spi0 SPI (r_sci_spi) Name Channel	mponents		Default Enabled g_spi0 4	(BSP)
Problem g_spi0 Si Settings	ns Console Properties X Smart Browsee PI (r_sci_spi) Property Common Parameter Checking DTC Support Module g_spi0 SPI (r_sci_spi) Name Channel Operating Mode	mponents		Default Enabled g_spi0 4 a Mast	(BSP)
Problem g_spi0 Si Settings	ns Console Properties X Smart Browsee PI (r_sci_spi) Property Common Parameter Checking DTC Support Module g_spi0 SPI (r_sci_spi) Name Channel Operating Mode Clock Phase	mponents		Default Enabled g_spi0 4 (a) Mast Ga Data	(BSP) I ter sampling on odd edge, data variatior
Problem g_spi0 Si Settings	ns Console Properties X Smart Browsee PI (r_sci_spi) Property V Common Parameter Checking DTC Support V Module g_spi0 SPI (r_sci_spi) Name Channel Operating Mode Clock Phase Clock Planity	mponents		Default Enabled g_spi0 4 ŵ Mast ŵ Data ŵ Low	(BSP) I ter sampling on odd edge, data variatior when idle
Problem g_spi0 Si Settings	ns Console Properties X Smart Browsee PI (r_sci_spi) Property Common Parameter Checking DTC Support Module g_spi0 SPI (r_sci_spi) Name Channel Operating Mode Clock Phase Clock Polarity Mode Fault Error	mponents		Default Enabled g_spi0 4 â Mast â Data â Low Disable	(BSP) I ter sampling on odd edge, data variatior when idle
Problem g_spi0 Si Settings	ns Console Properties X Smart Browsee PI (r_sci_spi) Property Common Parameter Checking DTC Support Module g_spi0 SPI (r_sci_spi) Name Channel Operating Mode Clock Phase Clock Phase Clock Polarity Mode Fault Error Bit Order	mponents		Default Enabled g_spi0 4 3 Mast 3 Data 3 Data 3 Data 3 Data 3 MSB	(BSP) I ter sampling on odd edge, data variation when idle First
Problem g_spi0 Si Settings	ns Console Properties X Smart Browsee PI (r_sci_spi) Property Common Parameter Checking DTC Support V Module g_spi0 SPI (r_sci_spi) Name Channel Operating Mode Clock Polarity Mode Fault Error Bit Order Callback	mponents		Default Enabled g_spi0 4 @ Mast @ Data @ Low Disable @ MSB @ mp_t	(BSP) I ter sampling on odd edge, data variatior when idle First ple_spp_host_spi_callback
Problem g_spi0 Si Settings	ns Console Properties X Smart Browsee PI (r_sci_spi) Property Common Parameter Checking DTC Support Module g_spi0 SPI (r_sci_spi) Name Channel Operating Mode Clock Phase Clock Phase Clock Polarity Mode Fault Error Bit Order	mponents		Default Enabled g_spi0 4 3 Mast 3 Data 3 Data 3 Data 3 Data 3 MSB	(BSP) I ter sampling on odd edge, data variatior when idle First ple_spp_host_spi_callback
Problem g_spi0 Si Settings	ns Console Properties X Smart Browsee PI (r_sci_spi) Property Common Parameter Checking DTC Support Module g_spi0 SPI (r_sci_spi) Name Channel Operating Mode Clock Plase Clock Polarity Mode Fault Error Bit Order Callback Receive Interrupt Priority Transmit Interrupt Priority	mponents		Default Enabled g_spi0 4 3 Mast a Lata Data 3 Data 3 MsB 3 msB 3 msB 3 msB 4 mr. Priority Priority	(BSP) I sampling on odd edge, data variation when idle First Je_spp_host_spi_callback 12 12
Problem g_spi0 Si Settings	ns Console Properties X Smart Browsee PI (r_sci_spi) Property V Common Parameter Checking DTC Support V Module g_spi0 SPI (r_sci_spi) Name Channel Operating Mode Clock Phase Clock Phase Clock Plaste Clock Plaste	mponents		Default Enabled g_spi0 4 ŵ Mast ŵ Low Disable ŵ MSB ŵ m L Priority	(BSP) I sampling on odd edge, data variatior when idle First Je_spp_host_spi_callback 12 12

Figure 26. SPI Module Properties

See Figure 26, SPI Module Properties. For the SPI configuration, it is necessary to specify which SCI channel to use. Match the SCI channel used and configure it as the SPI interface for PMOD1 in the **Pins Tab** see Figure 28, Pin Configuration > SCI to configure SPI Mode and Pins. Here SCI Channel 4 is being used



on the RA4M2 in SPI Mode with the Pins assigned to PMOD1 (conn J26). See the Schematic for EK-RA4M2 for more details on the PMOD1 signals and pin assignments.

			🕀 g_spi0 SPI (r_sci_spi)	g_external_irq0 External IRQ (r_icu)
	New Object > R Remove New Object > R Remove SP Clocks Pins Interrupts Event Links Stacks C Gonsole Pins Interrupts & Pins Horows Gonsole Pins Properties X P Smart Brows			I Transfer (LR RVI d data full)
		er sy onarchionaar yy ocarch		
g_spiu SH	Pl (r_sci_spi)			
Settings	Property			Value
API Info	Clock Phase			🔒 Data sampling on odd edge, data va
Arring	Clock Polarity			🔒 Low when idle
	Mode Fault Error			Disable
	Bit Order			🔒 MSB First
	Callback			🔒 rm_ble_spp_host_spi_callback
	Receive Interrupt Priority			Priority 12
	Transmit Interrupt Priority			Priority 12
	Transmit End Interrupt Priority			Priority 12
	Error Interrupt Priority			Priority 12
	Bitrate			500000
	Bitrate Modulation			Disabled
	✓ Pins			
	TXD4			P207
	RXD4			P206
	SCK4			P400
	CTS4			None
	CTSRTS4			P401

Figure 27. SPI Module Properties – Pin Assignments

Change the following SPI Configuration Properties in the properties window as in Figure 26 and Figure 27.

Table 6. SPI configuration

Property	Changed Value	Default Value
Common→DTC Support	Enabled	Disabled
Module g_spi0 SPI→Channel	4 for EK-RA4M2 case	0
Pins→TXD4	P207	None
Pins→RXD4	P206	None
Pins→SCK4	P400	None
Pins→CTS4	None	None
Pins→CTSRTS4	P401	None

3. Add the DTC driver (r_dtc) to both transmit and receive. See Figure 26, SPI Module Properties, g_transfer0 SCI4 TXI and g_transfer1 SCI4 RXI for more details.

Select Pin Configuration		📑 Export to CS	🕅 file 🔚 Configure Pin D	river Warnings	
RA4M2 EK	✓ Manage configurations	🗹 Genera	ate data: g_bsp_pin_cfg		
Pin Selection	E ⊞ ⊟ ↓ªz	Pin Configuration			
Type filter text Analog:ADC Analog:ANLOG Analog:DAC Connectivity:CAN Connectivity:IIC Connectivity:SCI SCI0 SCI1 SCI2 SCI3 	^	Name Pin Group Selection Operation Mode V Input/Output TXD4 SCK4 CTS4 SDA4 SCL4 CTSRTS4	Value Mixed Simple SPI ✓ P207 ✓ P206 ✓ P400 None None None Y P401		Link
SCI3 SCI4 SCI9 Connectivity:SPI	•	Module name: SCI4 Usage: When using When switch	Simple I2C mode, ensure prind between I2C and other	ort pins output ty modes. first disal	pe is n-ch

Figure 28. Pin Configuration > SCI



For the TXD_MOSI and RXD_MISO pin assignments, it is necessary to specify which SCI channel to use on the **Pins** tab. For the EK-RA4M2 with RYZ012 PMOD connected to PMOD1 (conn J26), SPI is assigned to SCI Channel 4 (SCI4) and the **Operation Mode** set to **Simple SPI**. The Input/Output Pins are assigned as specified in Figure 28, Pin Configuration > SCI. If your application uses a different RA MCU or PMOD connector then refer to the RA MCU Schematic to determine the Pin assignments for the PMOD connector.

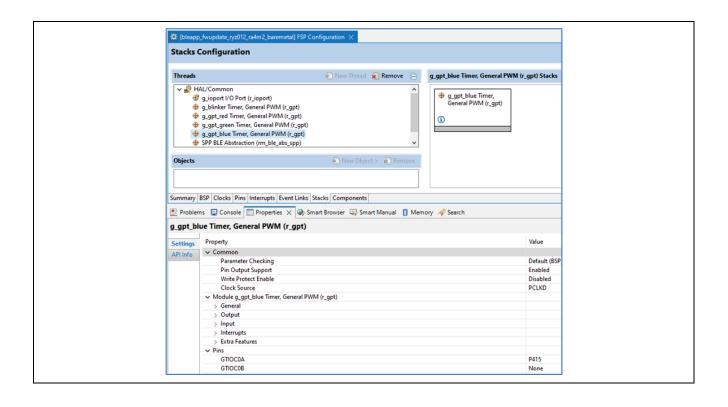
In the **Pin Configuration**>**Ports**, confirm that the **Ports** > **P4 Pin assignments** match the SPI Configuration in Table 5, SPP Module Configuration. These Pin assignments are for the EK-RA4M2 and RYZ012 connected to PMOD1 connector (J26). You will need to modify these assignments when using a different PMOD connector or evaluation kit / board type.

Pin Configuration				
Select Pin Configuration		Export to CSV file	Configure Pin Driver Warr	nings
RA4M2 EK	✓ Manage configurations	🗹 Generate da	ita: g_bsp_pin_cfg	
Pin Selection	⊨ ± ⊃ ↓ª₂	Pin Configuration		
Type filter text		Name	Value	Link
		Symbolic Name	PMOD1_SS	
✓ ✓ P4 ✓ P400	^	Comment		
✓ P400		Mode	Peripheral mode	
✓ P401		Pull up	None	
✓ P403		IRQ	None	
✓ P404		Drive Capacity	Low CMOS	
✓ P405		Output type V Input/Output	CMOS	
✓ P406 ✓ P407		P401	✓ SCI4_CTSRTS	4
P407 P408				
P409				
P410		Module name: P401		
P411		Port Capabilities: CAN0: CTX0		
D/12	•	GPT POEGO: G	ETRGA	
Pin Function Pin Number				
[bleapp_fwupdate_ryz012_ra4m2_	baremetal] FSP Configuration ×			
Pin Configuration				
Select Pin Configuration		Export to CSV fil	e 🖀 Configure Pin Driver Wa	rnings
[_		_
RA4M2 EK	✓ Manage configurations	🗹 Generate d	ata: g_bsp_pin_cfg	
Pin Selection	⊨ ∓ ⊨ ↓ªz	Pin Configuration		
		Name	Value	Link
Type filter text		Name Sumbalia Nama	Value PMOD1_RST	LINK
✓ ✓ P4	^	Symbolic Name Comment	PMOD1_KST	
✓ P400		Mode	Output mode (Initial Low)	
✓ P401 ✓ P402		Pull up	None	
✓ P402		IRQ	None	
✓ P404		Drive Capacity	Low	
✓ P405		Output type	CMOS	
✓ P406		✓ Input/Output P403	✓ GPIO	\Rightarrow
✓ P407		P405		
P408				
P409 P410		NA 11 0403		
P410 P411		Module name: P403 Port Capabilities: AGT0: AGTI00		
P/12	~	AGTI: AGTIO1		
Pin Function Pin Number				
Summary BSP Clocks Pins Interru	ots Event Links Stacks Components			
(bleapp_fwupdate_ryz012_ra4m2_b)	aremetal] FSP Configuration $ imes$			
Pin Configuration				
-		-		
		Export to CSV fr	e 🖀 Configure Pin Driver Wa	arnings
Select Pin Configuration		201		
Select Pin Configuration RA4M2 EK	<u>Manage configurations</u>		ata: g_bsp_pin_cfg	
_	 ✓ Manage configurations E ⊕ □ ↓^a_Z 		lata: g_bsp_pin_cfg	
RA4M2 EK Pin Selection		Pin Configuration		Link
RA4M2 EK Pin Selection Type filter text	⊨ + ⊢ Jªz	Generate of Pin Configuration Name	Value	Link
RA4M2 EK Pin Selection Type filter text		Pin Configuration		Link
RA4M2 EK Pin Selection Type filter text ✓ ✓ P4 ✓ P400	⊨ + ⊢ Jªz	Generate of Pin Configuration Name Symbolic Name	Value	Link
RA4M2 EK Pin Selection Type filter text ✓ ♥ P4 ✓ ₽ 400 ✓ ₱ 401	⊨ + ⊢ Jªz	✓ Generate of Pin Configuration Name Symbolic Name Comment Mode Pull up	Value PMOD1_IO1 Output mode (Initial Low) None	Link
RA4M2 EK Pin Selection Type filter text ✓ ✓ P4 ✓ P400 ✓ P401 ✓ P402	⊨ + ⊢ Jªz	Generate of Pin Configuration Name Symbolic Name Comment Mode Pull up IRQ	Value PMOD1_IO1 Output mode (Initial Low) None None	Link
RA4M2 EK Pin Selection Type filter text ✓ ♥ P4 ✓ ₽ 400 ✓ ₱ 401	⊨ + ⊢ Jªz	Generate of Pin Configuration Name Symbolic Name Comment Mode Pull up IRQ Drive Capacity	Value PMOD1_IO1 Output mode (Initial Low) None Low	Link
RA4M2 EK Pin Selection Type filter text ✓ ✓ P4 ✓ P400 ✓ P401 ✓ P402 ✓ P403	⊨ + ⊢ Jªz	✓ Generate of Pin Configuration Name Symbolic Name Comment Mode Pull up IRQ Drive Capacity Output type	Value PMOD1_IO1 Output mode (Initial Low) None None	Link
Pin Selection Type filter text ✓ ✓ P4 ✓ P400 ✓ P401 ✓ P402 ✓ P403 ✓ P404	⊨ + ⊢ Jªz	Generate of Pin Configuration Name Symbolic Name Comment Mode Pull up IRQ Drive Capacity Output type ∨ Input/Output	Value PMOD1_IO1 Output mode (Initial Low) None Low CMOS	
RA4M2 EK Pin Selection Type filter text ✓ ✓ P4 ✓ ✓ P401 ✓ P401 ✓ P403 ✓ P403 ✓ P404 ✓ P405 ✓ P406	⊨ + ⊢ Jªz	✓ Generate of Pin Configuration Name Symbolic Name Comment Mode Pull up IRQ Drive Capacity Output type	Value PMOD1_IO1 Output mode (Initial Low) None Low	
RA4M2 EK Pin Selection Type filter text ✓ P40 ✓ P401 ✓ P402 ✓ P403 ✓ P404 ✓ P405 ✓ P405 ✓ P406 ✓ P407 P408 P408	⊨ + ⊢ Jªz	Generate of Pin Configuration Name Symbolic Name Comment Mode Pull up IRQ Drive Capacity Output type ∨ Input/Output	Value PMOD1_IO1 Output mode (Initial Low) None Low CMOS	
RA4M2 EK Pin Selection Type filter text ✓ ✓ P4 ✓ P401 ✓ P403 ✓ P403 ✓ P404 ✓ P405 ✓ P406 ✓ P409	⊨ + ⊢ Jªz	✓ Generate of Pin Configuration Name Symbolic Name Comment Mode Pull up IRQ Drive Capacity Output type V Input/Output P402	Value PMOD1_IO1 Output mode (Initial Low) None Low CMOS	
RA4M2 EK Pin Selection ▼ype filter text ▼ ≠ P400 ▼ ≠ P401 ▼ ₽403 ▼ ₽404 ▼ ₽404 ▼ ₽405 ▼ ₽406 ♥ ₽407 ₽408 ₽409 ₽410	⊨ + ⊢ Jªz	✓ Generate of Pin Configuration Name Symbolic Name Comment Mode Pull up IRQ Drive Capacity Output type ✓ Input/Output P402 Module name: P402	Value PMOD1_IO1 Output mode (Initial Low) None Low CMOS ✓ GPIO	Link
RA4M2 EK Pin Selection Type filter text ✓ ✓ P4 ✓ P401 ✓ P403 ✓ P403 ✓ P404 ✓ P405 ✓ P406 ✓ P409	⊨ + ⊢ Jªz	✓ Generate of Pin Configuration Name Symbolic Name Comment Mode Pull up IRQ Drive Capacity Output type V Input/Output P402	Value PMOD1_IO1 Output mode (Initial Low) None Low CMOS ✓ GPIO	

Figure 29. Pin Configuration > Ports



Stacks Configuration	on			
Threads	🐑 New Thread	🔊 Remove 😑	g_gpt_red Timer, General PWM (r	_gpt) Stacks
 g_gpt_red Time g_gpt_green Time g_gpt_blue Time 	General PWM (r_gpt) ; General PWM (r_gpt) er, General PWM (r_gpt) er, General PWM (r_gpt) tion (rm_ble_abs_spp)	t> 🔊 Remove	g_gpt_red Timer, General PWM (r_gpt) (1)	
Problems 📮 Console	s Interrupts Event Links Stacks Components Properties × Smatter Stocks Stacks Stacks Components rail PWM (r_gpt)	rt Manual 🏮 Mem	ory 🔗 Search	
Settings Property				Value
API Info V Common				
Paramet	er Checking			Default (BSP)
	ut Support			Enabled
Write Pro	otect Enable			Disabled
Clock So	urce			PCLKD
✓ Module g_g	pt_red Timer, General PWM (r_gpt)			
> General				
> Output				
i land				
> Input	-			
> Input > Interrupt	3			
> Interrupt > Extra Fea				
> Interrupt	itures			P405





Stacks Configuration Inreads New Thread Remove g.gpt.green Timer, General PWM	
WHAL/Common Image: gipped trip Image:	
	/M (r_gpt) Stacks
Summary BSP Clocks Pins Interrupts Event Links Stacks Components Problems Console Properties X Smart Browser S Smart Manual Memory Search paget green Timer, General PWM (r_gpt) V Common Parameter Checking Pin Output Support Write Protect Enable Clock Source V Module g_gpt.green Timer, General PWM (r_gpt) General Output Input	
Problems Console Properties X Smart Browser S Smart Manual Memory Search <u>g_gpt_green Timer, General PWM (r_gpt)</u> Settings Property Common Parameter Checking Pin Output Support Write Protect Enable Clock Source Module g_gpt_green Timer, General PWM (r_gpt) Sentral Source Module g_not green Timer, General PWM (r_gpt) Sentral Source Module g_total provide the second seco	
Problems Console Properties X Smart Browser Smart Manual Memory Search g_gpt_green Timer, General PWM (r_gpt) Settings Property API Info Parameter Checking Pin Output Support Write Protect Enable Clock Source V Module g_gpt_green Timer, General PWM (r_gpt) > General > Output > Input > Input > Input > Interrupts > Extures V Pins	
Problems Console Properties X Smart Browser Smart Manual Memory Search g_gpt_green Timer, General PWM (r_gpt) Settings Property API Info Parameter Checking Pin Output Support Write Protect Enable Clock Source V Module g_gpt_green Timer, General PWM (r_gpt) > General > Output > Input > Input > Input > Interrupts > Extures V Pins	
Problems Console Properties X Smart Browser S Smart Manual Memory Search g.gpt_green Timer, General PWM (r_gpt) Settings Property Common Parameter Checking Pin Output Support Write Protect Enable Clock Source Module g.gpt_green Timer, General PWM (r_gpt) Sentral Source Sourc	
g_gpt_green Timer, General PWM (r_gpt) Settings Property API Info Parameter Checking Pin Output Support Write Protect Enable Clock Source Module g_gpt_green Timer, General PWM (r_gpt) General Output Input Input Interrupts Interrupts Interrupts Vina Features Vina Featu	
Settings Property API Info 	
API Info	
API Info	Value
Parameter Checking Pin Output Support Write Protect Enable Clock Source Module g_gpt_green Timer, General PWM (r_gpt) > General > Output > Input > Input > Extra Features Y Pins	
Pin Output Support Write Protect Enable Clock Source Module g_gpt_green Timer, General PWM (r_gpt) General Output Input Input Extra Features V Pins	Default (BSP)
Write Protect Enable Clock Source Module g_ppt_green Timer, General PWM (r_gpt) Seneral Output Interrupts Extra Features Vins	Enabled
Clock Source V Module g_gpt_green Timer, General PWM (r_gpt) General Output Input Input Extra Features V Pins	Disabled
Module g_gpt_green Timer, General PWM (r_gpt) Seneral Output Input Interrupts Extra Features Pins	PCLKD
Seneral Output Input Interrupts Extra Features Pins	1 CERE
Output Input Input Interrupts Extra Features Pins	
> Input > Interrupts > Extra Features V Pins	
> Interrupts > Extra Features	
> Extra Features V Pins	
✓ Pins	
	None
GTIOC3A GTIOC3B	P404

Figure 30. Stack Configuration Settings

Stacks	Configuration					
Threads		🐑 New Thread	d 🔊 Remove 😑	g_blinker Tim	er, General PWM	(r_gpt) Stacks
4	HAL/Common 9 g.joport I/O Port (r_ioport) 9 g.johrker Timer, General PWM (r_gpt) 9 g.gpt_rde Timer, General PWM (r_gpt) 9 g.gpt_green Timer, General PWM (r_gpt) 9 g.gpt_blue Timer, General PWM (r_gpt) 9 SPP BLE Abstraction (rm_ble_abs_spp)		^ •	 g_blink General i 	er Timer, PWM (r_gpt)	
Objects		🐔 New Obj	ject > 🔊 Remove			
Summary		- <u> </u>	-			
Proble	ms Console Properties X &			nory 🛷 Search		Value
Proble g_blinke Settings	Console Properties X Console Properties X Property			nory 🔗 Search		Value
Proble g_blinke Settings	Timer, General PWM (r_gpt) Property Common			nory 🛷 Search		
Proble g_blinke Settings	Console Properties X Properties X Property Common Parameter Checking			nory 🔗 Search		Default (BSP)
Proble g_blinke Settings	Console Properties X Properties X Property Common Parameter Checking Pin Output Support			nory 🔗 Search		Default (BSP) Enabled
Proble g_blinke Settings	Console Properties X Properties X Property Common Parameter Checking Pin Output Support Write Protect Enable			nory 🔗 Search		Default (BSP) Enabled Disabled
Proble g_blinke Settings	Console Properties X Property Common Parameter Checking Pin Output Support Write Protect Enable Clock Source	Smart Browser 🤑 Sr		nory 🔗 Search		Default (BSP) Enabled
Proble g_blinke Settings	Console Properties X Properties X Property Common Parameter Checking Pin Output Support Write Protect Enable	Smart Browser 🤑 Sr		nory 🔗 Search		Default (BSP) Enabled Disabled
Proble g_blinke Settings	Console Properties X Property Common Parameter Checking Pin Output Support Write Protect Enable Clock Source Module g_blinker Timer, General PI	Smart Browser 🤑 Sr		nory 🔗 Search		Default (BSP) Enabled Disabled
Proble g_blinke Settings	Console Properties X Properties X Property Common Parameter Checking Pin Output Support Write Protect Enable Clock Source Module g_blinker Timer, General Pi S General	Smart Browser 🤑 Sr		nory 🔗 Search		Default (BSP) Enabled Disabled
Proble	Console Properties X Properties X Property Common Parameter Checking Pin Output Support Write Protect Enable Clock Source Module g_blinker Timer, General S Output	Smart Browser 🤑 Sr		nory 🔗 Search		Default (BSP) Enabled Disabled
Proble g_blinke Settings	Console Properties X Property Common Property Common Property Common Parameter Checking Pin Output Support Write Protect Enable Clock Source Module g_blinker Timer, General PI S General Output S Input	Smart Browser 🤑 Sr		nory 🔗 Search		Default (BSP) Enabled Disabled
Proble	Console Properties X Properties X Property Common Parameter Checking Pin Output Support Write Protect Enable Clock Source Module g_blinker Timer, General Pi S General S Output Input Interrupts	Smart Browser 🤑 Sr		nory 🛷 Search		Default (BSP) Enabled Disabled
Proble g_blinke Settings	Console Properties X Properties X Property Common Parameter Checking Pin Output Support Write Protect Enable Clock Source Module g_blinker Timer, General PI S General S Output S Input Interrupts Extra Features	Smart Browser 🤑 Sr		nory 🔗 Search		Default (BSP) Enabled Disabled

5.5 Add General PWM Timers

This step is required for Firmware Update OTA Watchdog functionality. In addition, it provides an indication on the board LEDs that the application is running. Add the General PWM Timers (r_gpt) for the LED blinker timer, watchdog, and board LEDs. See Figure 30, Stack Configuration Settings



5.6 Add RYZ012 BLE OTA Service Profile

QE for BLE can generate a custom BLE profile and the Bluetooth LE application skeleton code. You must modify this source code according to your application BLE functional requirements. See Renesas QE for BLE Tool information <u>QE BLE Tool Dev Assistance Documents</u> about the usage of QE for BLE.

The BLE Profile created in this application project is as follows. From the e² studio Menu go to **Renesas Views > Renesas QE > R_BLE Custom Profile RA,RE,RX (QE).** Once the BLE Custom Profile opens, then select the **Project:** <YourProject> and **Module:** RYZ012 from the drop-down boxes.

Profile Peripheral Central						
P Profile Si [Server, Client] GAP Service	Customize	Server 🗹 Client				^
C Device Name	Name:	GAP Service				
C Peripheral Preferred Connection Parameters C Central Address Resolution	UUID:	0x1800			16 bits \sim	
C Resolvable Private Address Only	Abbreviation:	gap				
 ✓ (§) [Server] GATT Service ✓ (²) Service Changed 	Description:	The generic_access	service contains	generic information about the dev	ice.	
Client Characteristic Configuration	Aux Properties:	Authorization				
	Security Level:	C Level 2: Unauthe Level 3: Authent	nticated pairing v icated pairing wit		ryption	
	Callback:	Enable Write Cal				
	Included:	GATT Service				
		÷ X				
		Name	Code			
	Error Codes:					

Figure 31. Creating BLE Profile with QE for BLE

The BLE Profile is shown above in Figure 31 is required for the OTA Firmware Update of the RYZ012A. Select the **Peripheral** tab to change **Local Name** to use for BLE Advertising in the RYZ012. Set **Complete local name** to "**RYZ012**". See Figure 32 Setting Complete Local Name.

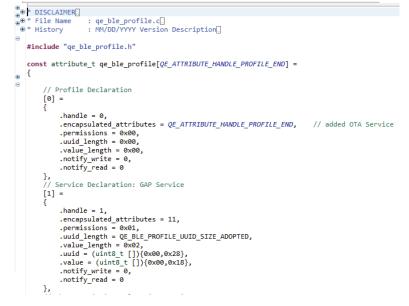


BLE Custom Profile	Module: RYZ012 v Project: bleapp_fwupdate_ryz012_ra4m2_baremetal v
Profile Peripheral Central	
 Advertising Data 11/31 Flags Service Class UUIDs Local Name Tx Power Level Slave Connection Interval Range Service Solicitation UUIDs Service Data Public Target Address Random Target Address Appearance Advertising Interval Manufacturer Specific Data Service Class UUIDs Service Class UUIDs Local Name Tx Power Level Slave Connection Interval Range Service Class UUIDs Local Name Tx Power Level Slave Connection Interval Range Service Solicitation UUIDs 	Local Name Short local name Complete local name RYZ012
 Service Data Public Target Address Random Target Address Appearance Advertising Interval Manufacturer Specific Data Advertising Parameter 	Sets the name of the device. Specify the local name so that it is the same as the device name for the characteristics of the GAP Service. The type of name can be selected as either of the following two. • Short local name: Shows an abbreviated name of the device. • Complete local name: Shows the complete name of the device.

Figure 32. Setting Complete Local Name

The OTA app or GATT Browser app on the mobile device can be used to verify that the RYZ012 is running (advertising) after a firmware update.

After the BLE Profile is created for your application, press **Generate Code** to create the BLE application skeleton code and BLE Profile. The tool generated source code will appear under /qe_gen/ble/app_main.c and qe_ble_profile.c. Edit qe_ble_profile.h and qe_ble_profile.c files with the following code additions to add the OTA Service to the Profile sent to the RYZ012 at BLE initialization.





.... <Lines of Code are Omitted Here>

```
// Descriptor: Client Characteristic Configuration
             [15] =
                           .handle = 15.
                          .encapsulated_attributes = 0,
.permissions = 0x03,
.uuid_length = QE_BLE_PROFILE_UUID_SIZE_ADOPTED,
                         .uula_length = 0x02,
.uula = (uint8 t []){0x00,0x20},
.uula = (uint8 t []){0x00,0x20},
.notify_write = 0,
.notify_read = 0
            },
// 16 : OTA Service and Characteristic
       //!< OTA Service : required to perform RYZ012 FW Update with TelinkOTA mobile app, choose this Service on mobile
[QE_ATTRIBUTE_HANDLE_SERVICE_DECL_OTA_SERVICE] =
                     .handle = QE_ATTRIBUTE_HANDLE_SERVICE_DECL_OTA_SERVICE,
.encapsulated_attributes = 4,
.permissions = 0x01,
.uuid_length = QE_BLE_PROFILE_UUID_SIZE_ADOPTED,
.value_length = QE_BLE_PROFILE_UUID_SIZE_CUSTOM,
.uuid = (uint8_t []){0x12,0x19,0x0d,0x0c,0x0b,0x0a,0x09,0x08,0x07,0x06,0x05,0x04,0x03,0x02,0x01,0x00},
.notify_write = 0,
.notify_read = 0
       },
//!< OTA Service : character value
//!< OTA Service : character.istl</pre>
        [QE_ATTRIBUTE_HANDLE_CHARACTERISTIC_DECL_OTA_CHARVAL] =
                     .handle = QE_ATTRIBUTE_HANDLE_CHARACTERISTIC_DECL_OTA_CHARVAL,
                     .handle = QE_ATTRIBUTE_HANDLE_CHARACTERISTIC_DECL_OTA_CHARVAL,
encapsulated_attributes = 0,
.permissions = 0x01,
.uuid_length = 0{EBLE_PROFILE_UUID_SIZE_ADOPTED,
.value_length = 0x00013,
.uuid = (uint8_t []){0x03,0x28},
.value = (uint8_t []){0x02}{0x04,0x48,0x00,0x12,0x28,0x0d,0x0c,0x0b,0x0a,0x09,0x08,0x07,0x06,0x05,0x04,0x03,0x02,0x01,0x00},
.notify_write = 0,
       },
          //!< OTA UUID : SPP data required to perform RYZ012 FW Update with TelinkOTA mobile app, choose this Characteristic on mobile
[QE_ATTRIBUTE_HANDLE_CHARACTERISTIC_VALUE_OTA_UUID] =</pre>
                           .handle = QE_ATTRIBUTE_HANDLE_CHARACTERISTIC_VALUE_OTA_UUID,
                         .handle = QE_AIINLBUIT_MANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHANDLE_CHA
                         .notify_write = 0x02,
.notify_read = 0x02
          },
//!< OTA Nam
            [QE_ATTRIBUTE_HANDLE_DESCRIPTOR_OTA_NAME] =
                           .handle = QE_ATTRIBUTE_HANDLE_DESCRIPTOR_OTA_NAME,
                        .handle = QE_ATTRIBUTE_HANDLE_DESCRIPTOR_OTA_NAM.
encapsulate_attributes = 0,
.permissions = 0x01,
.uuid_length = QE_BLE_PROFILE_UUID_SIZE_ADOPTED,
.value_length = 0x0003,
.uuid = (uint8_t []){0x01, 0x29},
.value = (uint8_t []){0x01, 0x29},
.notify_write = 0,
.notify_read = 0
             }
}:
```



5.7 Build and Modify Application Skeleton Code

- Select **Project > Build Project** from the menu bar or click the Build icon 10 to build the project.
- Click the debug icon 🖄 to launch the project. When the project starts, the sample application will be downloaded to EK-RA4M2.

6. Next Steps

Now that you have a basic understanding of the process to update the RYZ012 firmware with a BLE Device App you may want to customize the application. Please consider this information as you make those changes:

1. If you need to change the RYZ012 to a different PMOD connector, use a different RA kit, or use UART comms instead of SPI with the RYZ012, then:

To change RYZ012 Communications Interface Mode

 To change the RYZ012 comms interface to UART see section 5.4, Add and Configure the SPP BLE Module. When adding the SPP BLE Abstraction Stack, add the UART for "Transport Interface for communicating with the module".

Similar to configuring SPI, see section 5.4.2, Configure Peripherals for SPP BLE Abstraction Driver, to configure the UART Ports and Pins required for the PMOD connector on your board. In the step where Input/Output Pins are assigned as specified in Figure 28, Pin Configuration > SCI, set the Operation Mode to **Asynchronous UART**.

As a reference, see Figure 29, Pin Configuration > Ports, the PMOD Mode Select GPIO (UART or SPI mode is set by GPIO PMODx_IOx) must be configured to **UART** by setting the Pin Configuration > Mode to **Output mode (Initial High).**

Consult the schematic for your particular board to determine the correct Ports / Pins for the PMOD connector signals used in your application. Configure and confirm the **Peripherals > SCIx settings** and **Ports > Px > Pin Configuration** settings for all the PMOD signals.

If the PMOD signal pins do not appear as available in the Pin Configuration or Peripheral SCI channel, it may be required to disable unused SCI channels to gain access to the required PMOD Pins on your desired SCIx channel. Use **Peripherals > SCIx > Operation mode = Disabled**

To change RYZ012 PMOD Connector

To use a different PMOD connector other than PMOD1 (J26):

Consult the schematic for your particular board to determine the correct Ports / Pins for the PMOD connector signals used in your application.

Follow section 5.4.2, Configure Peripherals for SPP BLE Abstraction Driver to configure the Ports & Pins and comms mode (UART or SPI) required for the PMOD connector on your board.

Configure and confirm the **Peripherals** > **SCIx** settings and **Ports** > **Px** > **Pin Configuration** settings for all the PMOD signals.

If the PMOD signal pins do not appear as available in the Pin Configuration or Peripheral SCI channel, it may be required to disable unused SCI channels to gain access to the required PMOD Pins on your desired SCIx channel. Use **Peripherals > SCIx > Operation mode = Disabled**

To change the RA MCU kit or Custom Board

 To use a different RA MCU kit or custom board design it may be required to reassign the communications interface Port / Pins for the RYZ012 PMOD connector on your board. You will also be required to change the e² studio FSP Configuration BSP settings for the RA MCU kit or custom board Device used.

See To change RYZ012 PMOD Connector above.

It may be required to change the board LED port and pin assignments.

See section 0

Add General PWM Timer to change the LED pins.

Keep in mind that your RA MCU must have enough Code Flash to contain the entire RYZ012 firmware image. See Item 4 below for more details.



 To convert this e² studio project to use with IAR or Keil see the App Note for converting the project to IAR. Converting Applications from e² studio to IAR or Keil for RA – Application Note (R11AN0555EU0100)

Renesas FSP v3.8.0 and higher must be used to support the RYZ012 (FW SPP SDK v5.4 and higher) for BLE Radio based firmware update.

7. Appendix - Windows PC OTA App

An alternative to using a mobile device app is to use a PC with Web-based app <u>Telink OTA</u> (<u>pvvx.github.io</u>) to upgrade the firmware with a Windows 10 PC's built-in Bluetooth support. Note this Web-based app is shown for illustrative purposes only and is not supported by Renesas. Make sure Bluetooth is enabled on the PC.

- To use the OTA app, open with web-browser at Telink OTA (pvvx.github.io).
- Press "Connect" to search for "RYZ012" and select Pair. See Figure 33.

\equiv Telink OTA $_{D}$
Services UUID: 00010203-0405-0607-0809-0a0b0c0d1912 Characteristic: 00010203-0405-0607-0809-0a0b0c0d2b12
Connect Reconnect Hide_unknown Z BLE device name prefix filter(s) LYWSD03.ATC
Please select a .bin file you want to flash to a Telink BLE device. Select Firmware: Choose File No file chosen Status: waiting for you to connect a device
Start Flashing Clear Log

Figure 33. Windows PC Update – Press Connect and Search Devices

- Use "Choose File" to select Firmware BIN file from the e² studio project workspace in the project directory bleapp_fwupdate_ryz012_ra4m2_baremetal/8258_moduleV5_4.bin or 8258_moduleV5_5.bin. See Figure 34.
- Start the firmware update with "Start Flashing". See Figure 35.
- Monitor firmware update progress on RTT Viewer and OTA app. See Figure 11 and 36.

In the event that the firmware upgrade fails with error messages shown in RTT Viewer, the section 2.4, RYZ012 Firmware Update Considerations will help you to troubleshoot and resolve the issues.



Telink		9
Services UUID:		0
00010203-0405-0607-0809-0a0b0c0d1912		<u> </u>
Characteristic:		
00010203-0405-0607-0809-0a0b0c0d2b12		+
Connect Reconnect		
Hide unknown	C Open	
BLE device name prefix filter(s) LYWSD03,ATC	← → × ↑ 📙 « work > bleapp_fwup v i	ට 🔎 Search b
BLE device name prenx inter(s)	Organize New folder	
Please select a .bin file you want to flash to a Te Select Firmware: Choose File No file chosen Status: Start Flashing Clear Log 10:56:18 AM: No file selected 11:11:03 AM: Disconnected.	Cuick access Desktop Cuick access Debug Ge.gen ca.gen ca.gen cog.elipse.cc.# Azure iothubSetup IPM_Response Web Code Code Code Code Code Code Code Code	Date m(2/15/20, 2/10, 2
	File name: 8258_moduleV5_5.bin	
	File name: 8258_moduleV5_5.bin	→ BIN File (*.bin Open

Figure 34. Windows PC Update – Choose Firmware Bin File

	Telink OTA D
Services UUID):
00010203-0405	i-0607-0809-0a0b0c0d1912
Characteristic:	
00010203-0405	i-0607-0809-0a0b0c0d2b12
Connect	Reconnect
Hide unknown	
BLE device na	Ime prefix filter(s) LYWSD03,ATC
Please select a Select Firmwa	a .bin file you want to flash to a Telink BLE device. re: Choose File 8258_moduleV5_5.bin
Start Flashi Clear Log	Status: Connected.
11:18:58 AM 11:18:59 AM 11:20:24 AM	: Searching for devices : Connecting to: RYZ012 : Connected. : File was selected, size: 102436 bytes : Count: 6403

Figure 35. Windows PC Update – Start Firmware Update



Services UUID	
00010203-0405-0607-0809-0a0b0c0d1912	
Characteristic:	
00010203-0405-0607-0809-0a0b0c0d2b12 Connect Reconnect Hide unknown	
	_
BLE device name prefix filter(s) LYWSD03,ATC	
Please select a .bin file you want to flash to a Telink BL	E device.
Select Firmware: Choose File 8258_moduleV5_5.bin	
Status: Update done after 36.659 sec	conds
Start Flashing	
Clear Log	
11:17:42 AM: Searching for devices 11:18:58 AM: Connecting to: RYZ012 11:18:59 AM: Connected. 11:20:24 AM: File was selected, size: 102436 bytes 11:20:24 AM: Count: 6403 11:21:37 AM: Start DFU 11:22:14 AM: Update done after 36.659 seconds 11:22:22 AM: Disconnected.	

Figure 36. Windows PC Update – Firmware Upgrade Completed



Website and Support

Visit the following vanity URLs to learn about key elements of the RA family, download components and related documentation, and get support.

RA Product Information					
RA Product Support Forum					
RA Flexible Software Package					
Renesas RA™ EK-RA4M2 kit					
RYZ012x1 Bluetooth LE Module					
PMOD Expansion Board RYZ012x1					
Renesas Support					

renesas.com/ra renesas.com/ra/forum renesas.com/FSP renesas.com/ra/ek-ra4m2 renesas.com/ryz012x1-bluetooth-le-module renesas.com/pmod-expansion-board-ryz012x1 renesas.com/support



Revision History

		Description		
Rev.	Date	Page	Summary	
1.00	Jun.09.23	—	Initial release	
1.01	Sep.26.23	18	Revised section 2.4 RYZ012 Firmware Update Considerations	



General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)

A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.

2. Processing at power-on

The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power reaches the level at which resetting is specified.

3. Input of signal during power-off state

Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.

4. Handling of unused pins

Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.

5. Clock signals

After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.

6. Voltage application waveform at input pin

Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).

7. Prohibition of access to reserved addresses

Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.

8. Differences between products

Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Notice

- Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
- 2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
- 3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
- 4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
- 5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
- 6. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.

"Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.

"High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user's manual or other Renesas Electronics document.

- 7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION ("Vulnerability Issues"). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
- 8. When using Renesas Electronics products, refer to the latest product information (data sheets, user's manuals, application notes, "General Notes for Handling and Using Semiconductor Devices" in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
- 9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
- 10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
- 11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
- 12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
- This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
 Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.
- (Note1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.
- (Note2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu, Koto-ku, Tokyo 135-0061, Japan www.renesas.com

www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit: www.renesas.com/contact/.