RL78/G23

AWS Cloud Connectivity for MCU Firmware Update Over-the-Air on RL78/G23-128p Fast Prototyping Board with Wi-Fi DA16600

Introduction

This document provides instructions for running the AWS Cloud Connectivity for MCU Firmware Update Over-the-Air on RL78/G23-128p Fast Prototyping Board with Wi-Fi DA16600, utilizing the MCU firmware update commands of Wi-Fi DA16600 module.

Target Device

RL78 Family
RL78/G2x Series
RL78/G23 Group

Related Documents

[1] RL78/G23 User's Manual: Hardware (R01UH0896)

[2] RL78/G22, RL78/G23, RL78/G24 Firmware Update Module (R01AN6374)

[3] RL78/G23-128p Fast Prototyping Board User's Manual (R20UT4870)

[4] US159-DA16600EVZ Evaluation Board Manual (R15UZ0006)

Contents

1. Overview

This demo project shows the integration of AWS Cloud Connectivity for MCU Firmware Update Over-the-Air on the RL78/G23-128p FPB development board, utilizing the MCU firmware update commands of Wi-Fi DA16600 module for seamless wireless communication.

Key Steps in the Project:

- **Prepare an AWS Account and S3 Bucket:** Set up an AWS account and create an S3 bucket to serve as the server for uploading firmware.
- **Generate Key Pairs and Certificates:** Generate a public key and a private key to create both the initial firmware and the firmware used for updates.
- **Generate and Upload the Firmware File:** Generate the new firmware to be flashed onto the board via OTA and upload it to the Amazon S3 bucket.
- **Generate the initial firmware:** Configure and build the initial firmware, then generate the firmware file using the designated tool.
- **Execute the Demonstration Project:** Execute the demo project to validate the OTA process.

The following section provides a network stack related structure for Firmware Update Over-the-Air (OTA) demonstration.

DA16600 Wi-Fi Module with OTA On-Chip Commands. The DA16600 module connects to the AWS server and downloads the firmware directly. The downloaded firmware is then transferred to the MCU host for deployment. In this demonstration, HTTP and firmware downloads are offloaded, the firmware file URL is securely retrieved and pre-signed using a web browser.

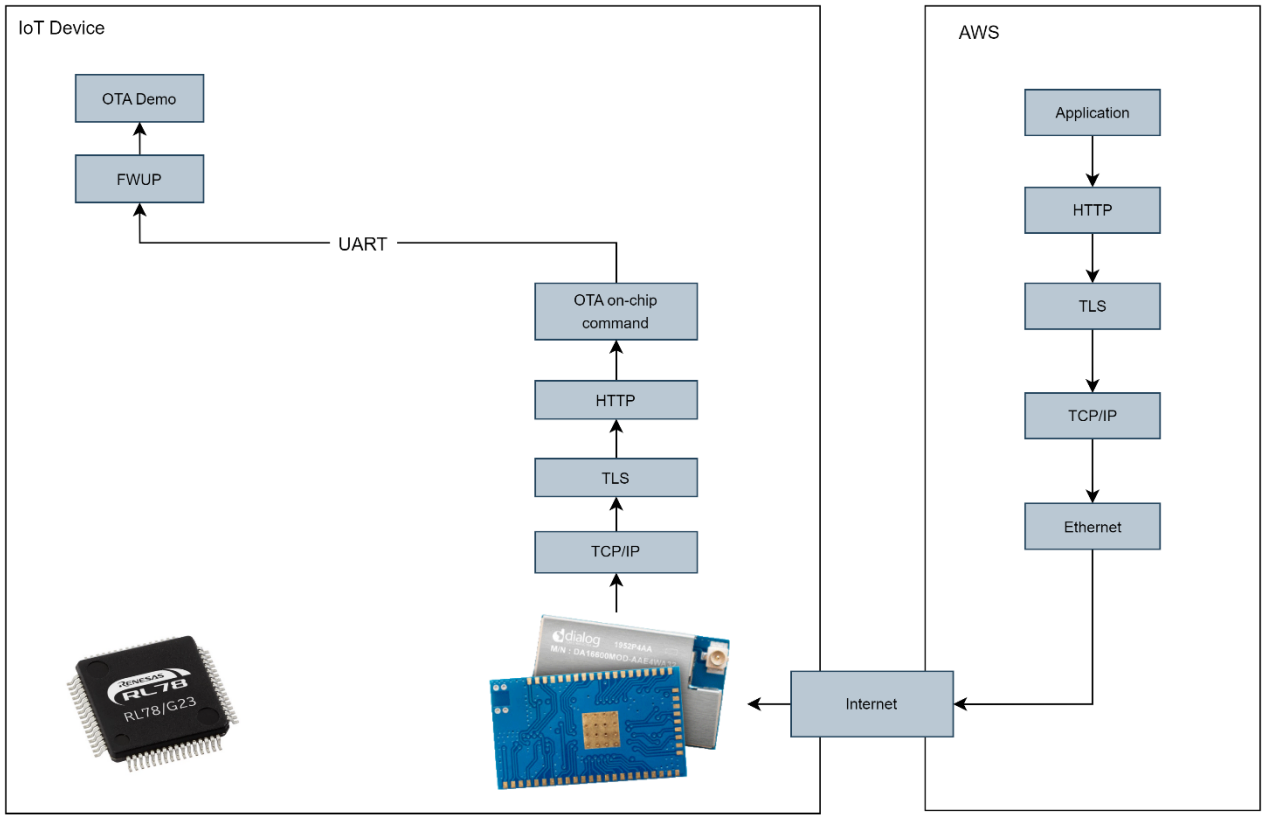

Figure 1.1 OTA Demonstrates with Wi-Fi DA16600 OTA On-Chip Commands

1.1 Workflow

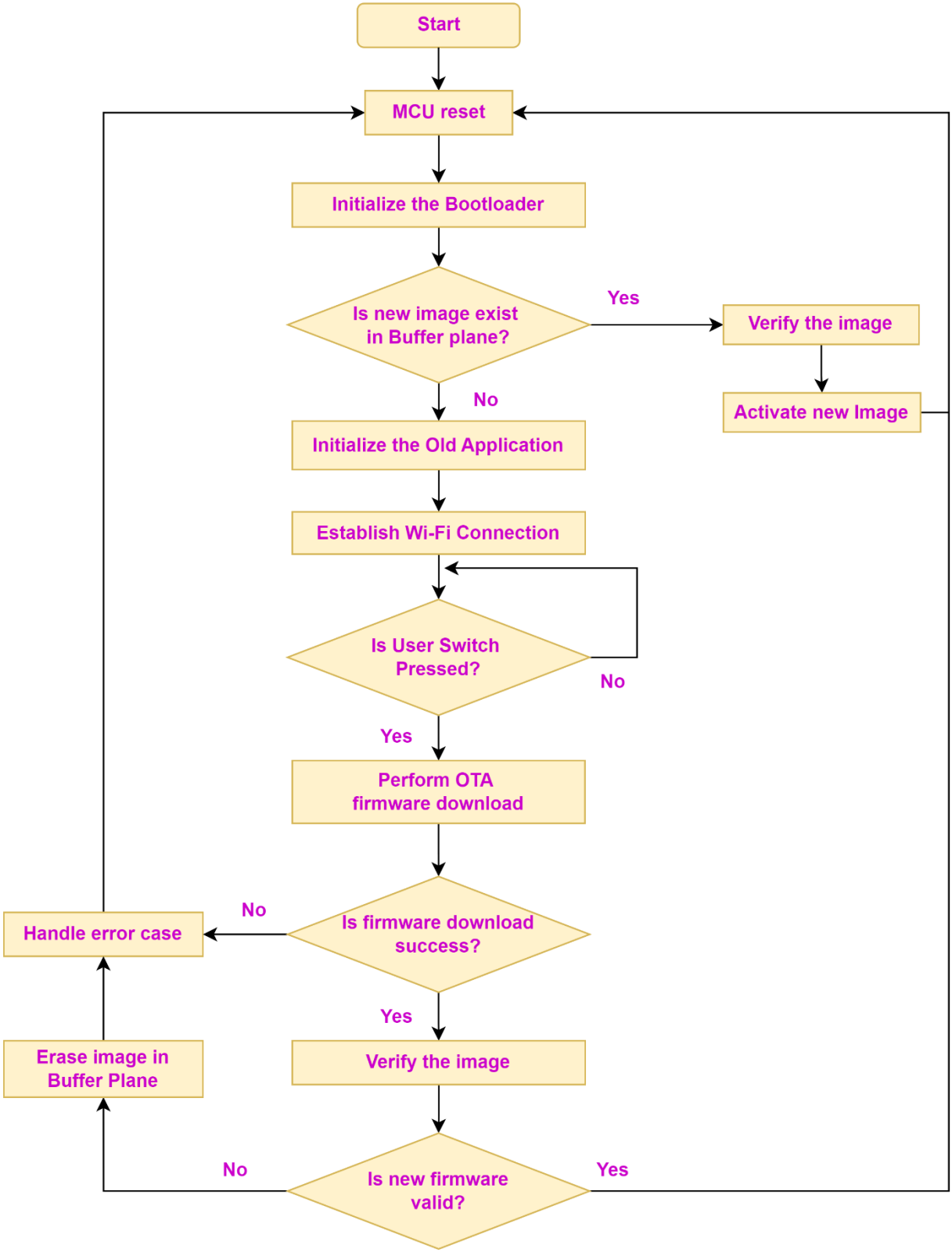

Figure 1.2 End-to-End OTA Workflow

1.2 Operation Confirmation Conditions

Demo project operations have been confirmed in the following conditions.

Table 1.1 Operation Confirmation Conditions

Item	Description
MCU	R7F100GSNxFB
Board	RL78/G23-128p Fast Prototyping Board (Product no.: (RTK7RLG230CSN000BJ)
IDE (Integrated Development Environment)	Renesas Electronics e2 studio 2025-01
C compiler	Renesas Electronics CC-RL V1.15.0
Firmware programming tool	Renesas Flash Programmer V3.18.00
Firmware update module (FWUP)	https://www.renesas.com/document/apn/rl78g22-rl78g23-rl78g24-firmware-update-module v2.01
Python	Python 3.12.7
Keygen tool	Win64 OpenSSL v3.0.12
SDK (Software Development Kit)	DA16200/DA16600 SDK V3.2.9.2

1.3 Equipment List

The following lists the equipment required for the demo project.

Table 1.2 Equipment List

Item	Description
Board	RL78/G23-128p Fast Prototyping Board https://www.renesas.com/rtk7rlg230csn000bj
Wi-Fi DA16600 module	PMOD Expansion Board for DA16600MOD US159-DA16600EVZ - Ultra-Low-Power Wi-Fi + Bluetooth Low Energy Combo Pmod Board
USB-UART conversion board	CP2101 USB TO TTL BOARD V4.2 CP2101 USB TO TTL BOARD V4.2 (agencyelectronics.com)
Micro USB Type-B cable	Connect another USB port on the base board to a PC for debugging purposes.
Jumper pin x 3	It is used to enable debugging mode.
Jumper wire x 3	Used to connect the USB-UART conversion board to the MCU board

1.4 Sample Project Code Sizes

The tables below show the ROM and RAM sizes for the sample projects included in the package associated with this application note. The values in the table below have been confirmed under the following conditions:

Compiler version: Renesas Electronics C/C++ Compiler for RL Family V1.15.00

CC-RL

- Optimization level: Size and execution speed (-Odefault)

Memory Usage			
Device	**Category**	**Memory Used (bytes)**	**Remarks**
RL78/G23	ROM	47055	ota_da16600_rl78g23_fpb
	RAM	5395	
	ROM	31313	boot_loader
	RAM	1655	

2. Demo Project Setup

2.1 Hardware Setup

First, the following shows the overall configuration of hardware that makes up the demo project.

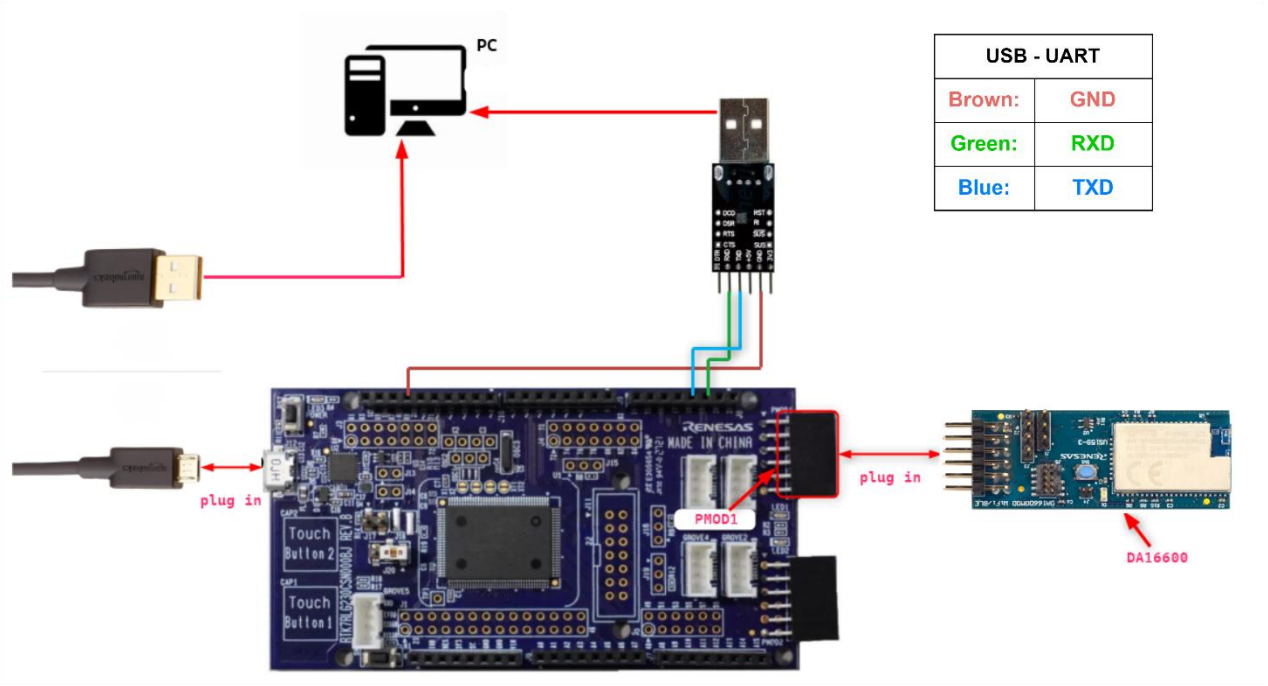

Figure 2.1 Hardware Connection

2.2 Software Setup

2.2.1 Installing Tool

2.2.1.1 Install Python

Python generates initialization firmware from bootloader and application projects, and application firmware from the new application project.

Follow the steps below to install Python:

(1) Access the Python download web site.

Download Python | Python.org

(2) Download the Python 3.12.7 installer.

Click the **Download** link for Python 3.12.7.

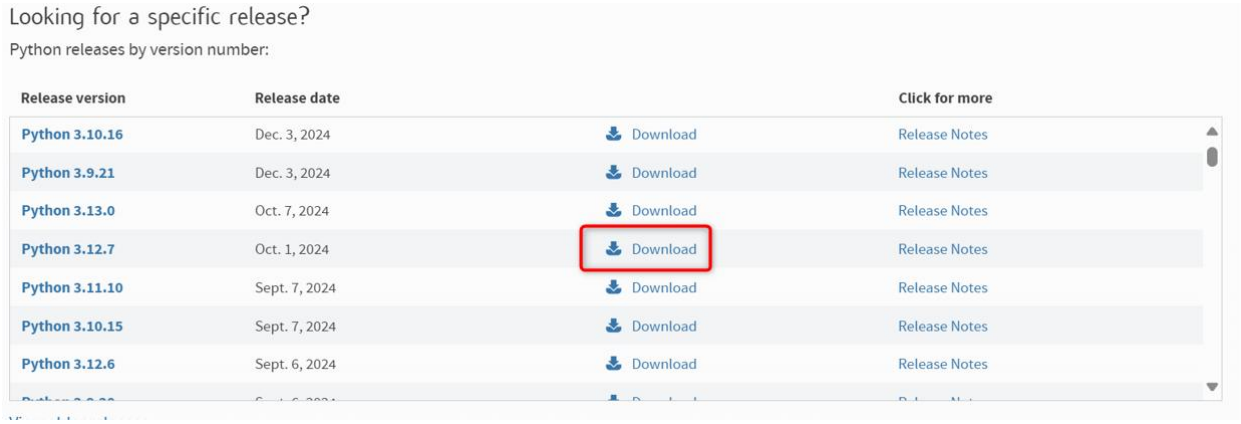

Figure 2.2 The Options for Installing the Release Version of Python

Download the installer for the operating system you are using.

Version	Operating System	Description	MD5 Sum	File Size	GPG	Sigstore	SBOM
Gzipped source tarball	Source release		5d0c0e4c6a022a87165a9addcd869109	25.8 MB	SIG	.sigstore	SPDX
XZ compressed source tarball	Source release		c6c933c1a0db52597cb45a7910490f93	19.5 MB	SIG	.sigstore	SPDX
macOS 64-bit universal2 installer	macOS	for macOS 10.13 and later	82711848a795f6d7b25e81844d5a9a3f	43.3 MB	SIG	.sigstore	
Windows installer (64-bit)	Windows	Recommended	b51e0889be50c55fbdd809f4ad587120	25.3 MB	SIG	.sigstore	SPDX
Windows installer (32-bit)	Windows		5d5452249401822cb3ad1bce7105d5fd	24.1 MB	SIG	.sigstore	SPDX
Windows installer (ARM64)	Windows	Experimental	19bdd2de8a7ccb6f1115f85bc54c1764	24.6 MB	SIG	.sigstore	SPDX
Windows embeddable package (64-bit)	Windows		4c0a5a44d4ca1d0bc76fe08ea8b76adc	10.6 MB	SIG	.sigstore	SPDX
Windows embeddable package (32-bit)	Windows		21a051ecac4a9a25fab169793ecb6e56	9.4 MB	SIG	.sigstore	SPDX
Windows embeddable package (ARM64)	Windows		6fc899d8dbd46dd2b585a038f7cf68a4	9.8 MB	SIG	.sigstore	SPDX

Figure 2.3 Python Windows Installer

(3) Run the installer and follow the prompts to install Python

On the installation screen, select the **Add python.exe to PATH** check box.

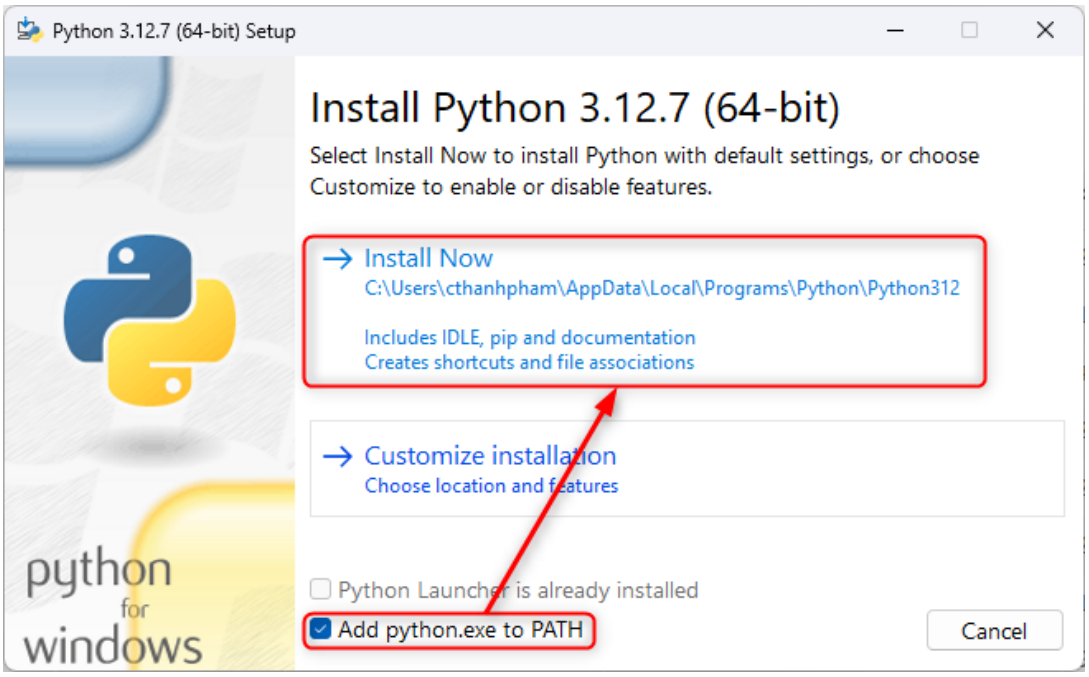

Figure 2.4 Python 3.12.7 installer

(4) Install the Python encryption library (pycryptodome)

Install the encryption library by executing the following command: *$ pip install pycryptodome*

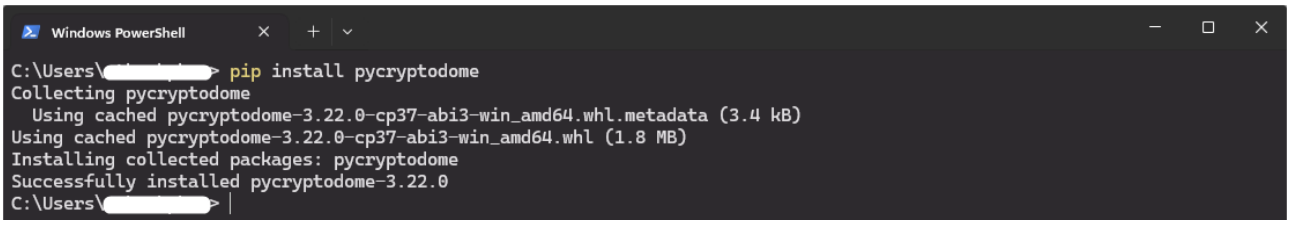

Figure 2.5 Installing Python Encryption Library

2.2.1.2 Installing OpenSSL

OpenSSL is a tool used to generate the cryptographic key pair required for firmware encryption and decryption during initialization and application firmware creation. OpenSSL can generate the following keys for use in the firmware update process:

- **Private key**: Used to encrypt the firmware and ensure its integrity.
- **Public key**: Used by the bootloader to decrypt and verify the firmware during update.

Follow the steps below to install and configure OpenSSL for this purpose.

(1) Access the Win32/Win64 Download Website for OpenSSL

Win32/Win64 OpenSSL Installer for Windows - Shining Light Productions

(2) Download the OpenSSL Installer

Download the installer for the operating system you are using.

Win64 OpenSSL v3.0.12 Light EXE | MSI	5MB Installer	Installs the most commonly used essentials of Win64 C by the creators of OpenSSL). Only installs on 64-bit ve chipsets. Note that this is a default build of OpenSSL a information can be found in the legal agreement of the
Win64 OpenSSL v3.0.12 EXE | MSI	140MB Installer	Installs Win64 OpenSSL v3.0.12 (Recommended for so OpenSSL). Only installs on 64-bit versions of Windows this is a default build of OpenSSL and is subject to loca found in the legal agreement of the installation.
Win32 OpenSSL v3.0.12 Light EXE | MSI	4MB Installer	Installs the most commonly used essentials of Win32 C 32-bit OpenSSL for Windows. Note that this is a defau and state laws. More information can be found in the le
Win32 OpenSSL v3.0.12	116MB Installer	Installs Win32 OpenSSL v3.0.12 (Only install this if you

Figure 2.6 The Options for Installing the Release Version of OpenSSL

(3) Run the Installer and Follow the Prompts to Install OpenSSL.

Select the option to copy the OpenSSL DLLs to the OpenSSL binaries directory.

(4) From the Start Menu, Open the Win64 OpenSSL Command Prompt.

Figure 2.7 OpenSSL Windows (64-bit)

(5) Confirm the OpenSSL Command from the Command Prompt.

Execute the following command and confirm that version information appears.

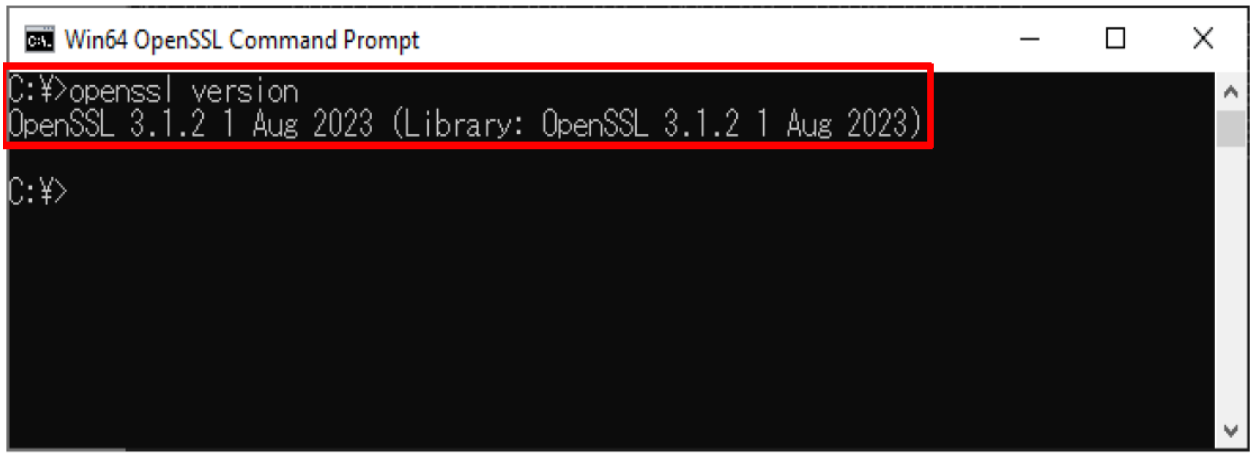

Figure 2.8 Checking OpenSSL Version

2.2.1.3 Installing Renesas Image Generator

Renesas Image Generator is a tool that generates the firmware images used by the firmware update module. Renesas Image Generator can generate the following images for use by the firmware update module:

- Initial image: An image file containing the bootloader and application program written by flash writer during initial system configuration (extension: mot).
- Update image: An image file containing the updated firmware (extension: rsu).

Renesas Image Generator is provided as part of the Firmware Update Module.

(1) Download the Firmware Update Module:

https://www.renesas.com/document/scd/rl78g22-rl78g23-rl78g24-firmware-update-module

(2) Extract the Downloaded Firmware Update Module

Extract the file RenesasImageGenerator.zip in the firmware update module.

The **RenesasImageGenerator** folder contains the Renesas Image Generator script file (image-gen.py) and the parameter files for various devices (*_ImageGenerator_PRM.csv).

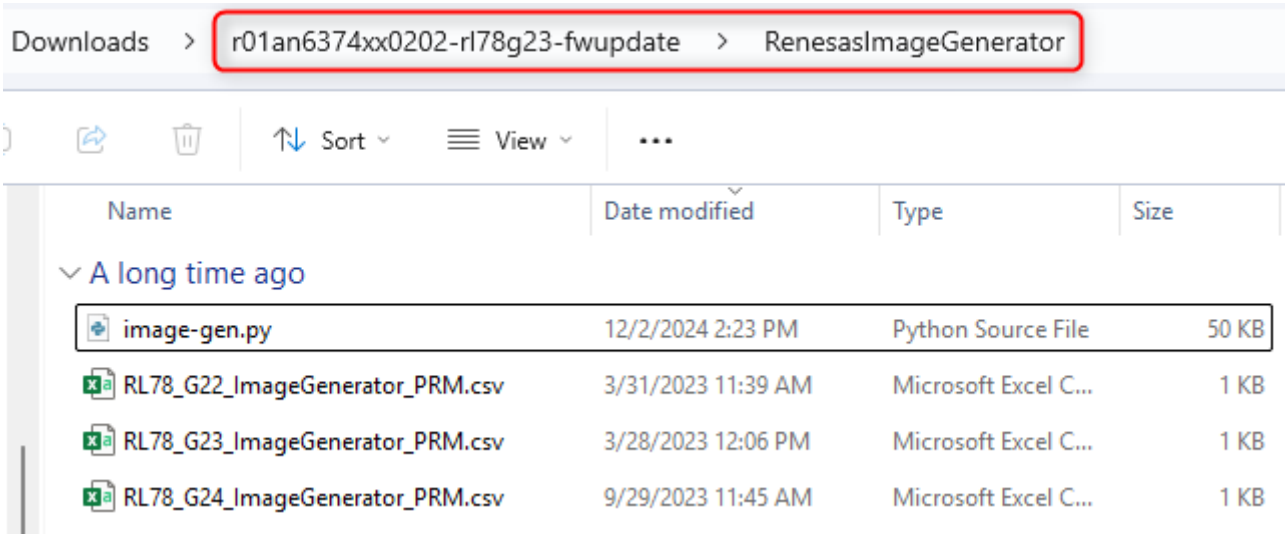

Figure 2.9 Renesas Image Generator Package

2.2.1.4 Installing Tera Term

Terminal software (example: Tera Term) is required to output demo project logs. The following show the serial port settings.

(1) Access the Tera Term Download Site.

Releases · TeraTermProject/osdn-download · GitHub

(2) Download the Tera Term Installer.

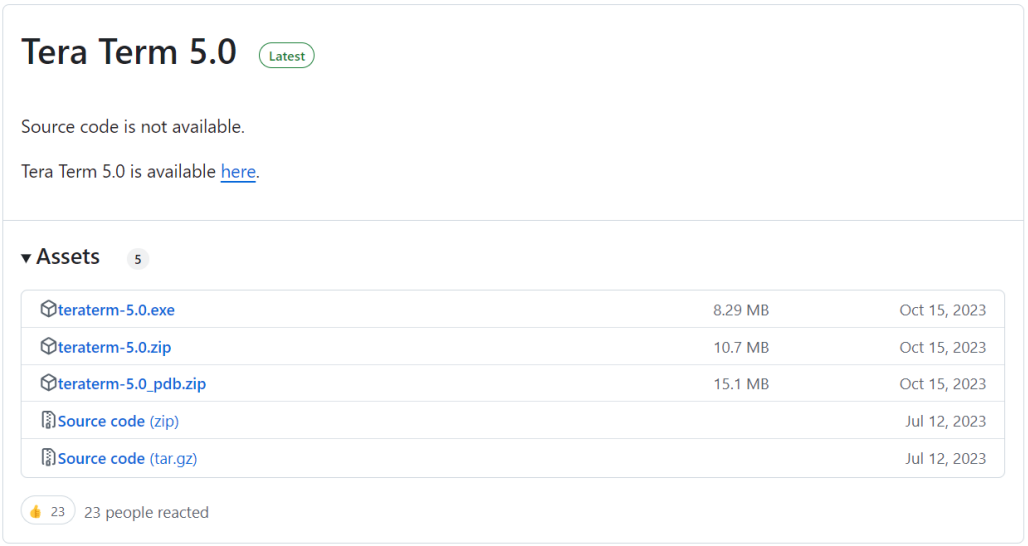

Figure 2.10 Tera Term Version 5.0

- Run the installer and follow the prompts to install Tera Term.
- Confirm that Tera Term starts when you click the Tera Term icon in the Start menu.

2.2.1.5 Installing Renesas Flash Programmer

Renesas Flash Programmer (RFP) is a utility provided by Renesas that allows users to write firmware to support Renesas MCUs via various interfaces such as USB, UART, or serial programming. It is an essential tool for flashing both the initial firmware and subsequent updates during development and production. Follow the steps below to install Renesas Flash Programmer on your computer.

(1) Access the Renesas download web site.

Renesas Flash Programmer (Programming GUI)

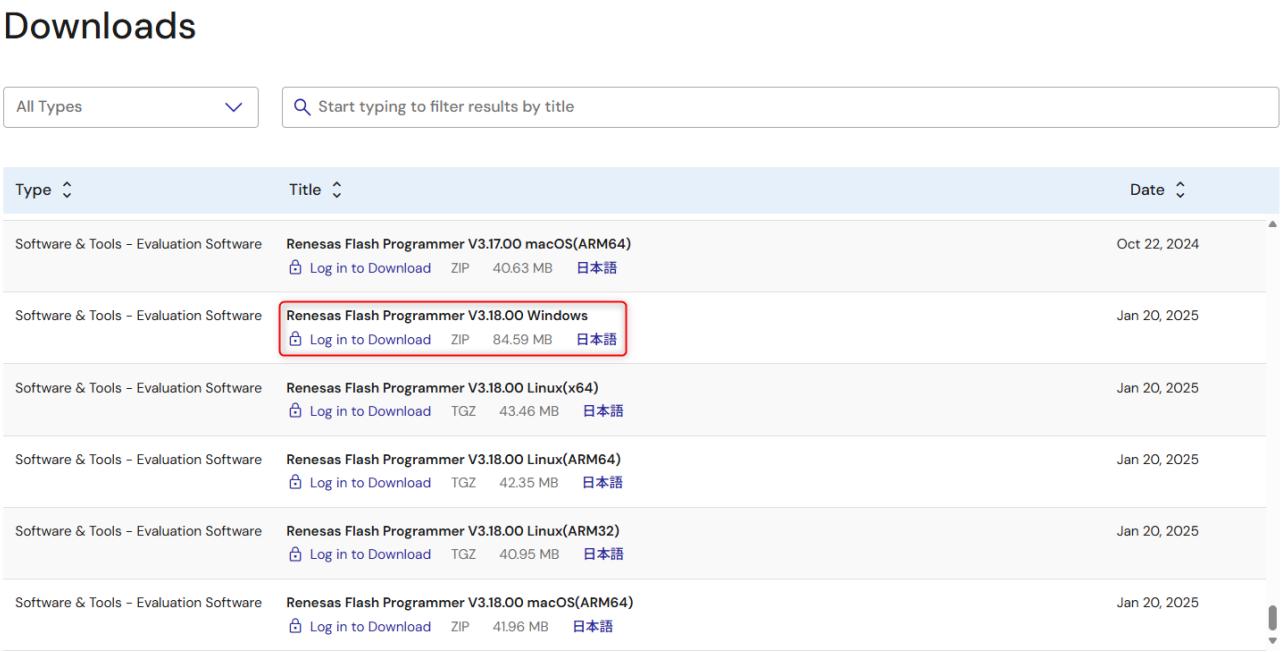

Figure 2.11 Renesas Flash Programmer

2.2.2 Terminal Software Setting

❖ With the UART TTL connection port of the RL78/G23-128p FPB:

(1) Open Tera Term select **New connection** and select Serial and the appropriate COM port for your **UART-to-USB** adapter, and **click OK**

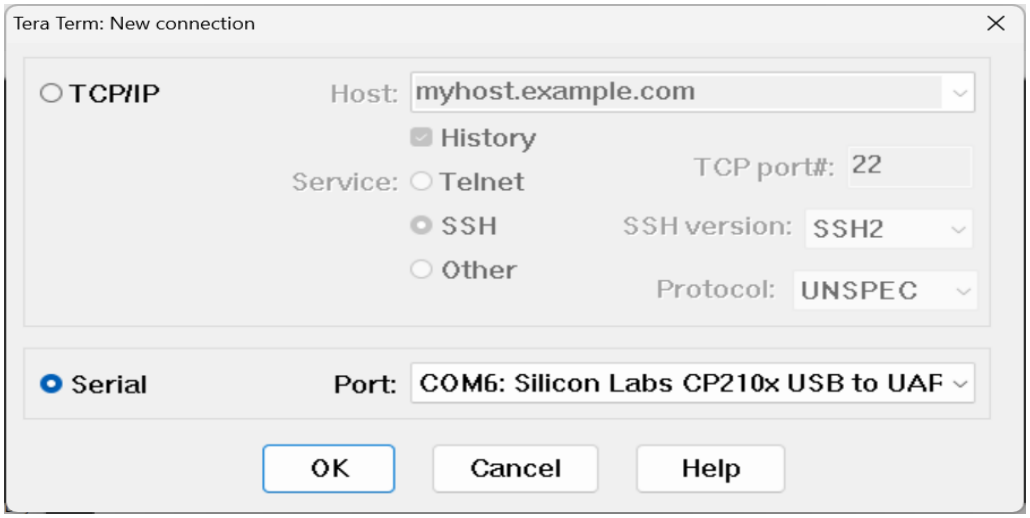

Figure 2.12 Tera Term Serial Connection

(2) Click **Setup > Terminal**…, in "**New-line"** section, set **"Receive**" as **AUTO.**

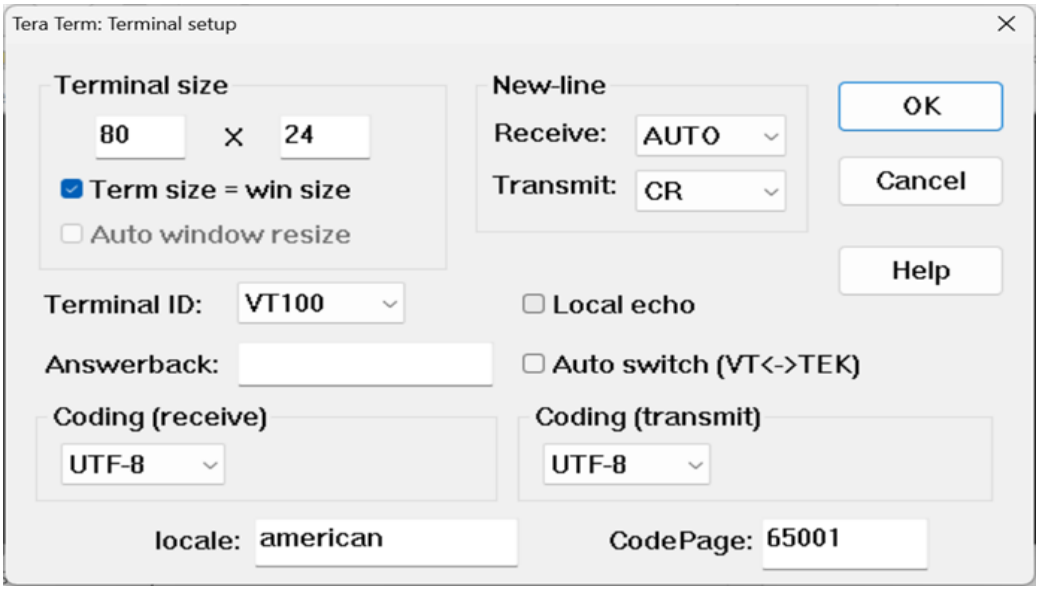

Figure 2.13 Terminal Setup for the UART TTL

(3) Click **Setup > Serial port…** and ensure that the speed is set to **115200**.

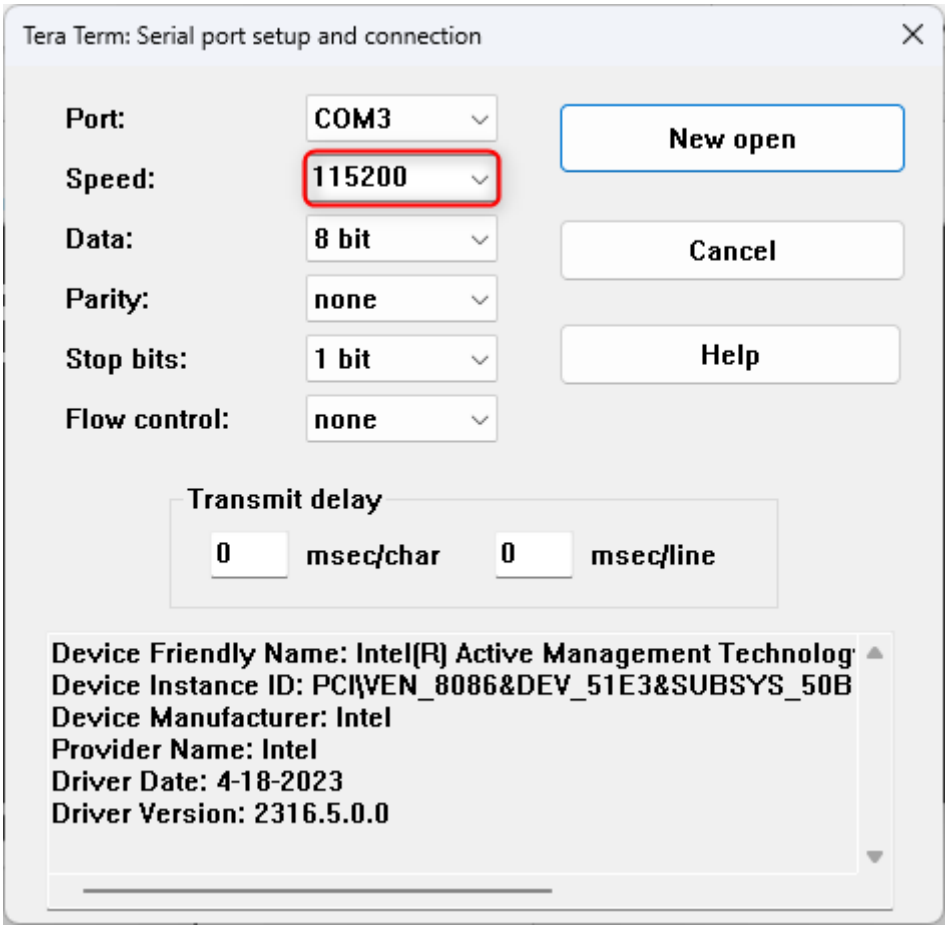

Figure 2.14 Serial Port Setup for UART TTL

2.2.3 Generate Key Pairs and Certificates

This section will generate a public key and a private key to create the initial firmware and the firmware used for updates.

To do this, open OpenSSL and enter the commands highlighted in yellow to generate the firmware verification keys.

```
openssl ecparam -genkey -name secp256r1 -out secp256r1.keypair

using curve name prime256v1 instead of secp256r1

openssl ec -in secp256r1.keypair -outform PEM -out secp256r1.privatekey

read EC key

writing EC key

openssl ec -in secp256r1.keypair -outform PEM -pubout -out secp256r1.publickey

read EC key

writing EC key
```

2.2.4 Project Description

The demo project is structured into multiple components to support the firmware update process. Each component plays a specific role in demonstrating the OTA update mechanism on the RL78/G23-128p FPB platform.

- **ota_da16600_rl78g23_fpb**: The existing application that the MCU runs **before** the firmware update process begins.
- **bootloader**: The bootloader is responsible for handling the firmware update process.

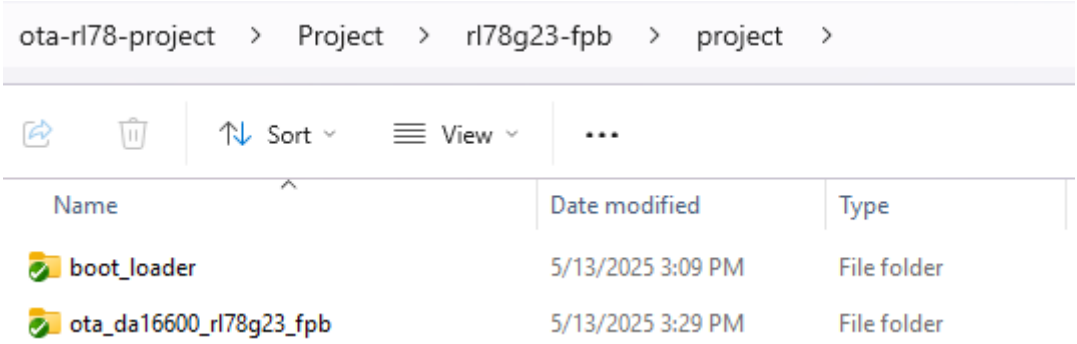

Figure 2.15 The Project Folder Structure Consists of Two Main Components

2.2.5 Importing The Demo Project

(1) Clone the demo project

(2) Extract the demo project

(3) Start e2 studio

(4) From the **File** menu, select **Import**

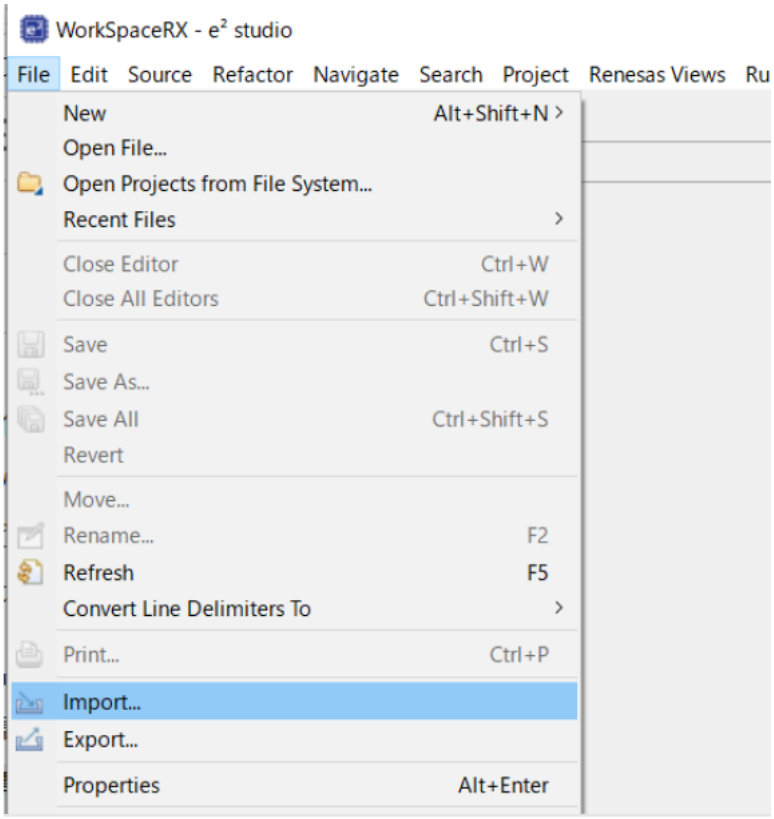

Figure 2.16 Importing the Project

(5) Select **Existing Projects into Workspace**

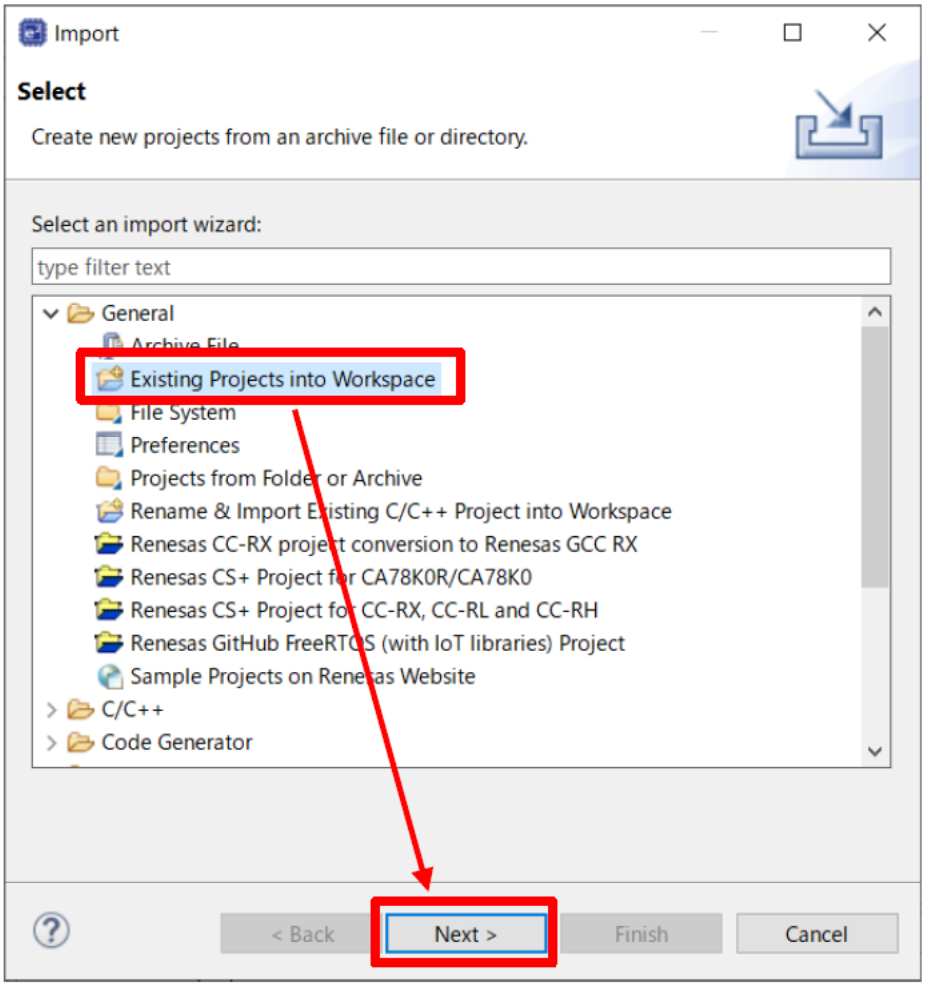

Figure 2.17 Select Existing Projects into Workspace

(6) In **Select root directory**, select the folder extracted, select the check boxes for the following projects, and then click **Finish**

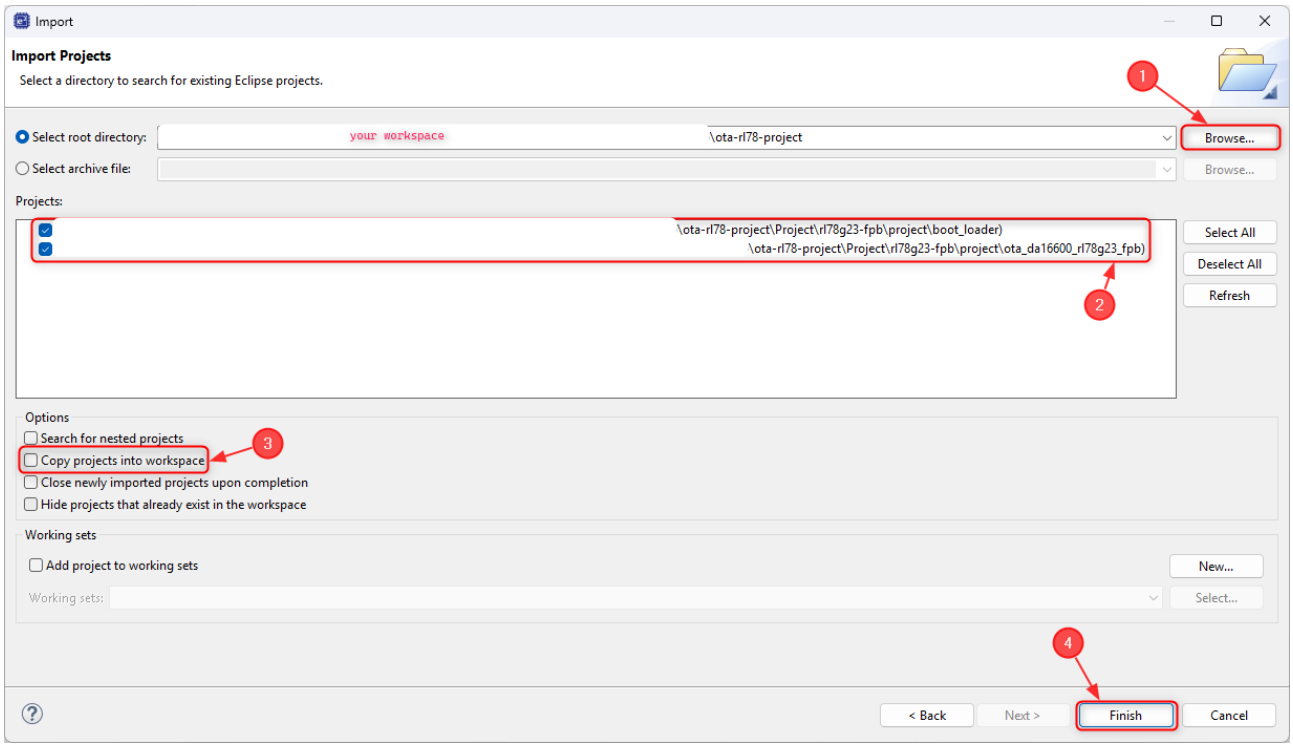

Figure 2.18 Complete Import Project

2.2.6 Create Firmware Initialization

2.2.6.1 Building Bootloader Project

(1) Update public key

Since the Renesas Image Generator is used to create the initial firmware, the public key must be pasted into the bootloader project to match the private key located in the Renesas Image Generator folder. This ensures that the initial firmware can be generated successfully. Follow the instructions below to complete this setup.

1. Copy the contents of the **secp256r1.publickey** file you created in **Section 2.2.3.**

2. Paste the public key into **CODE_SIGNENR_PUBLIC_KEY_PEM** in **\bootloader\src\key\code_signer_public_key.h.**

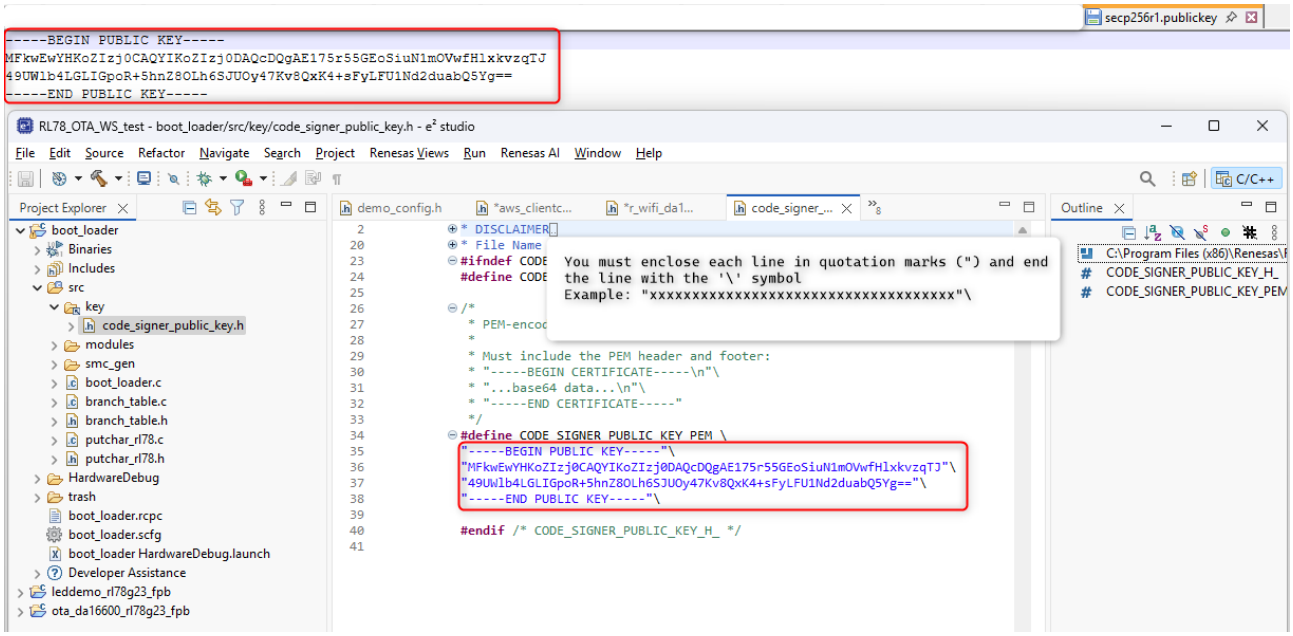

Figure 2.19 Assign a Public Key to Bootloader Project

(2) Building the project.

2.2.6.2 Building ota_da16600_rl78g23_fpb Project

(1) Configure firmware version, Wi-Fi, firmware type, and URL in demo_config.h at

\ota_da16600_rl78g23_fpb\src\helper\demo_config.h

```
/*
 * @brief Wi-Fi network to join.
 *
 * @todo If you are using Wi-Fi, set this to your network name.
 */
#define clientcredentialWIFI_SSID                    "SSID"

/*
 * @brief Password needed to join Wi-Fi network.
 * @todo If you are using WPA, set this to your network password.
 */
#define clientcredentialWIFI_PASSWORD                "PASSWORD"
```

SSID

PASSWORD

Figure 2.20 Configure Firmware Version, Wi-Fi Network, and Firmware URL

- Check the current firmware version.
- AP_WIFI_SSID: Set the access point name (SSID) of the local Wi-Fi network that the board will connect to.
- AP_WIFI_PASSWORD: Set the password for the local Wi-Fi network

(2) Building the project.

2.2.6.3 Creating the Initial Firmware

This section is used to create the initial firmware(**initial_firm.mot**), which is executed on the MCU before the Over-The-Air process begins. Follow the setup steps below to generate the initial firmware correctly:

(1) Place the following files in the Renesas Image Generator folder:

- The results of the building process in **Section 2.2.6.2**: **ota_da16600_rl78g23_fpb.mot**
- The results of building the bootloader in **Section 2.2.6.1**: **boot_loader.mot**
- The private key created in **Section 2.2.3: secp256r1.privatekey**

(2) Use Renesas Image Generator to generate the initial firmware

Open a command prompt, navigate to the Renesas Image Generator folder, and execute the following command to generate the file **initial_firm.mot**.

```
python image-gen.py -iup ota_da16600_rl78g23_fpb.mot -ip ^
RL78_G23_ImageGenerator_PRM.csv -o initial_firm -ibp boot_loader.mot -vt ecdsa
```

```
[redacted]\RenesasImageGenerator> python image-gen.py -iup ota_da16600_rl
78g23_fpb.mot -ip RL78_G23_ImageGenerator_PRM.csv -o initial_firm -ibp boot_loader.mot -vt ecdsa

Successfully generated the initial_firm.mot file.
```

Figure 2.21 Create the Initial Firmware

❖ **Parameter explanation:**

- **-iup**: Input user program (.mot file for application firmware)
- **-ip:** Input parameter file (.csv with image generation settings)
- **-o:** Output file prefix (e.g., userprog.mot)
- **-ibp:** Input bootloader program (.mot file)
- **-vt:** Verification type (e.g., ecdsa for digital signature)

2.2.7 Create Firmware File and Upload to the Amazon S3 Bucket

2.2.7.1 Create Firmware File

(1) Changing the firmware version

Change the firmware version to a higher version. (Example: Because previous versions is 1.0.0, so the new version we can choose 1.0.1)

Repeat the build process in **Section 2.2.6.2**, this time 3 specified for the DEMO_VERSION_BUILD definition in **\ota_da16600_rl78g23_fpb\src\helper\demo_config.h**

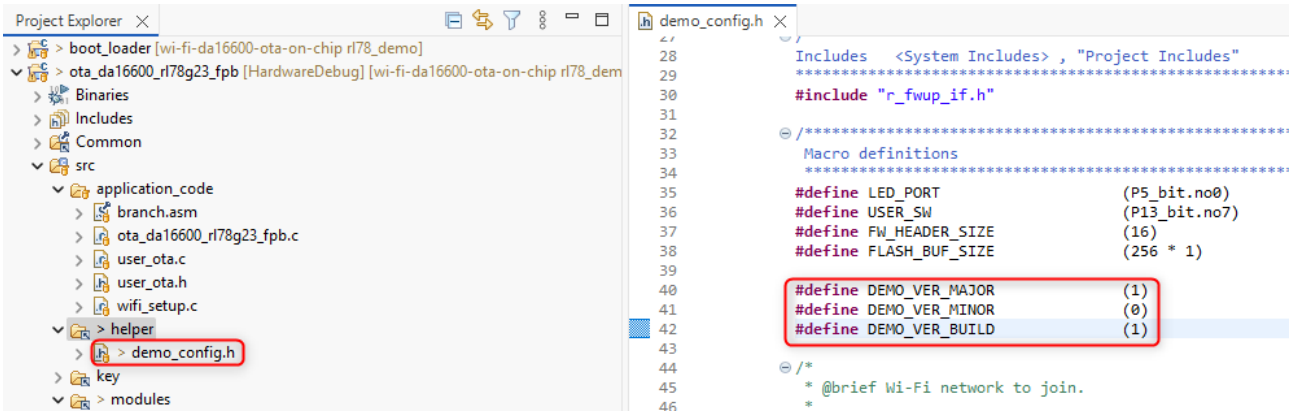

Figure 2.22 Setting New Version for Firmware

(2) Use Renesas Image Generator to Generate the Updated Firmware

Overwrite the file in the Renesas Image Generator folder with the firmware you rebuilt in 2.2.7.1(1) (ota_da16600_rl78g23_fpb.mot), and then execute the following command at the command prompt:

```
python image-gen.py -iup ota_da16600_rl78g23_fpb.mot -ip RL78_G23_ImageGenerator_PRM.csv -o ^
ota_da16600_rl78g23_fpb_v101 -vt ecdsa -key secp256r1.privatekey
```

This command generates a file named ota_da16600_rl78g23_fpb_v101.rsu.

2.2.7.2 Uploading Firmware to Amazon S3 Bucket

(1) Create Amazon S3 Bucket

(1) Access the AWS web site (Cloud Computing Services - Amazon Web Services (AWS)) and click **Sign In to the Console**.

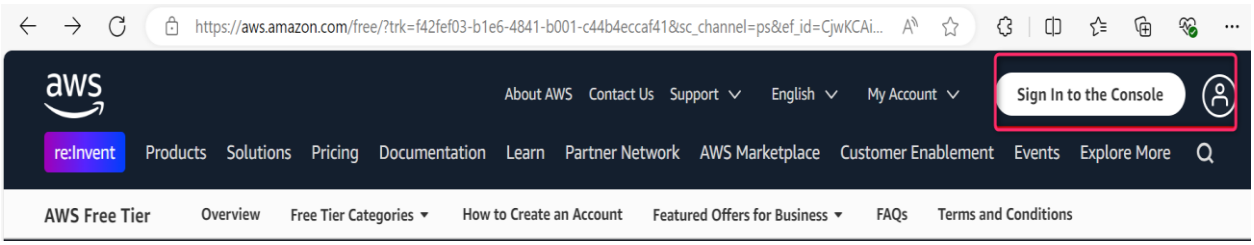

Figure 2.23 Sign-In to Console AWS

(2) Enter your email address or account ID, and then click **Next.**

If you are using the root account to sign in, select the "Sign in using root user email" option and enter the email address of the root account. If you are an IAM user, enter the **Account ID** (12-digit number or account alias), **IAM username**, and **Password** in the corresponding fields.

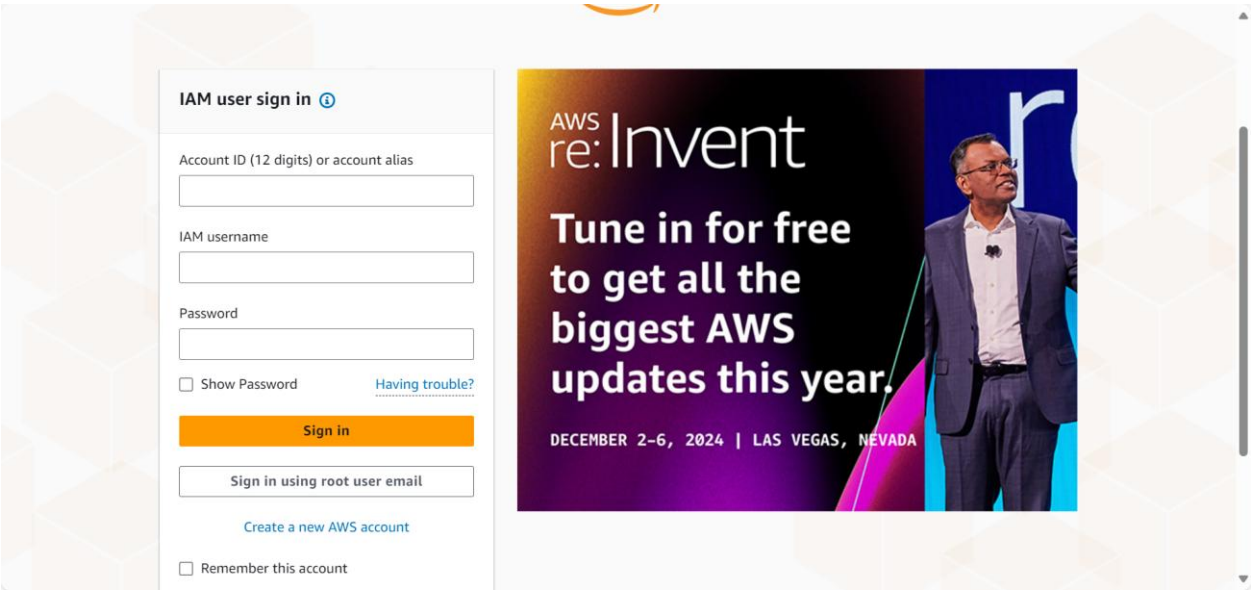

Figure 2.24 User Sign-In

(3) After logging in to AWS, select your region in the top right of the screen.

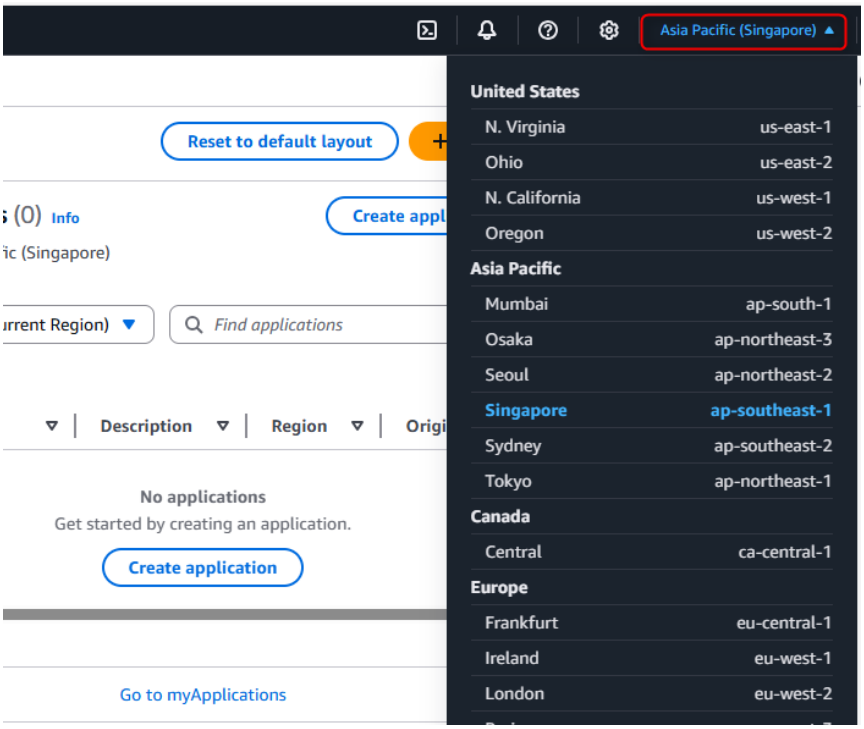

Figure 2.25 Setting Region in AWS

(4) From the **Services** menu, select **Storage** and then **S3.**

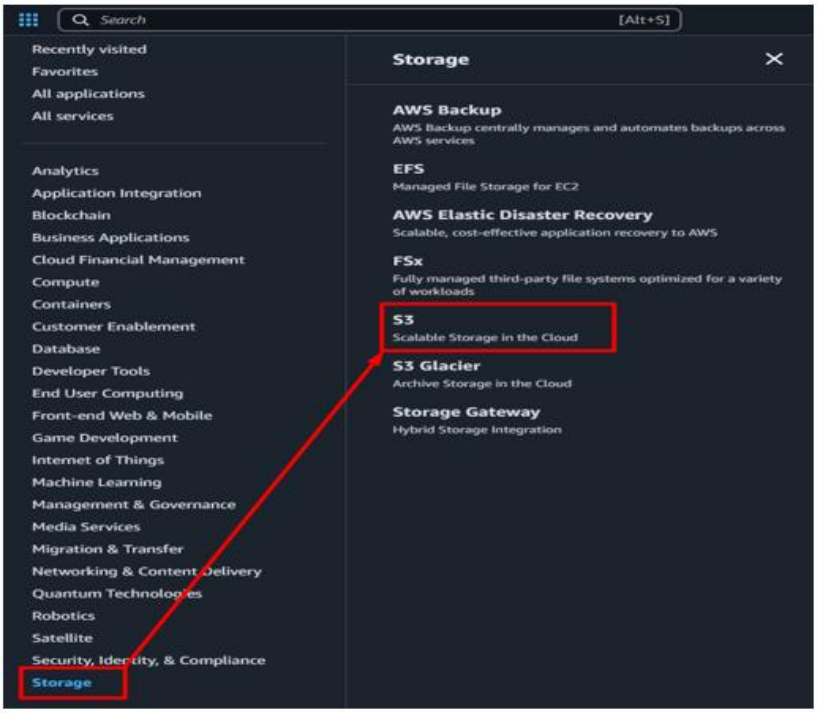

Figure 2.26 S3 AWS Bucket

(5) On the **Buckets** page, click the **Create bucket** button.

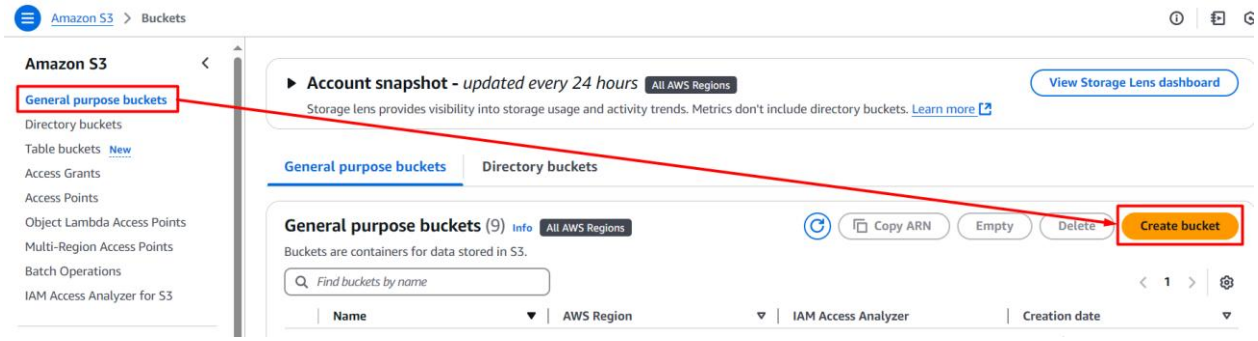

Figure 2.27 Create a Bucket

(6) Enter an S3 Bucket name.

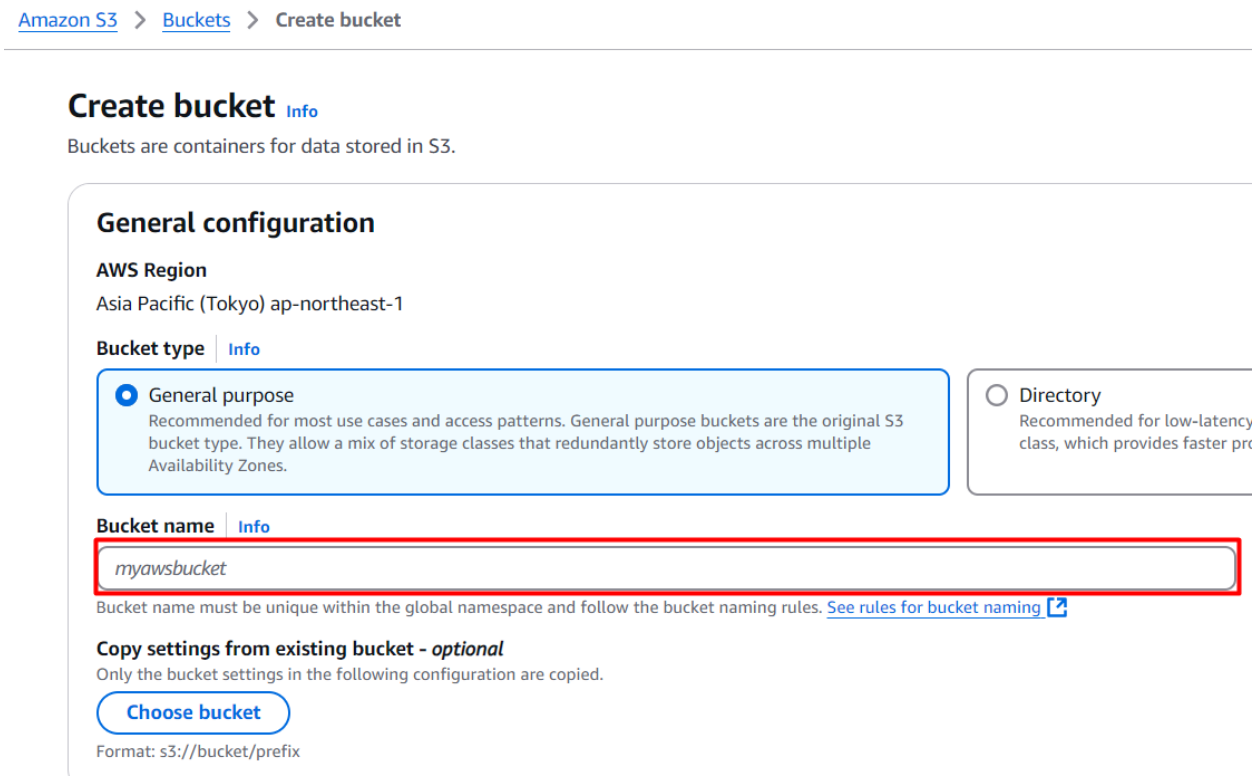

Figure 2.28 Create a Bucket Name

(7) Create Bucket.

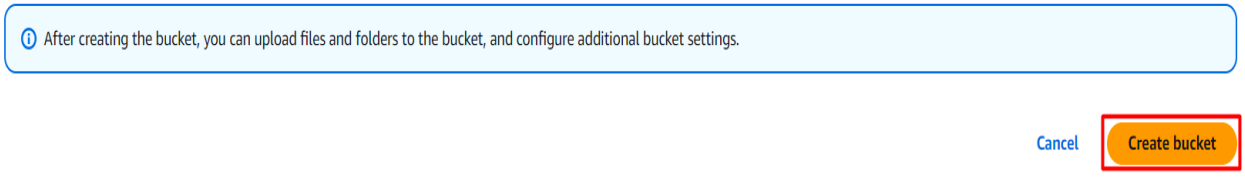

Figure 2.29 Create Bucket

(2) Upload the Firmware File to the Amazon S3 Bucket

(1) Choose your S3 bucket, and then click **Upload**.

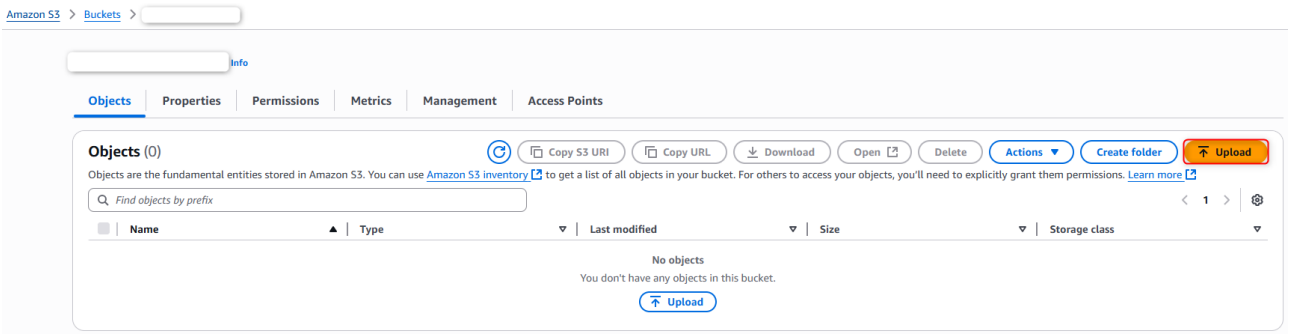

Figure 2.30 Uploading the Firmware File

(2) Click on **Add File**, then select the firmware you created in the previous section. Double-check to ensure it is the correct firmware before clicking **Upload**.

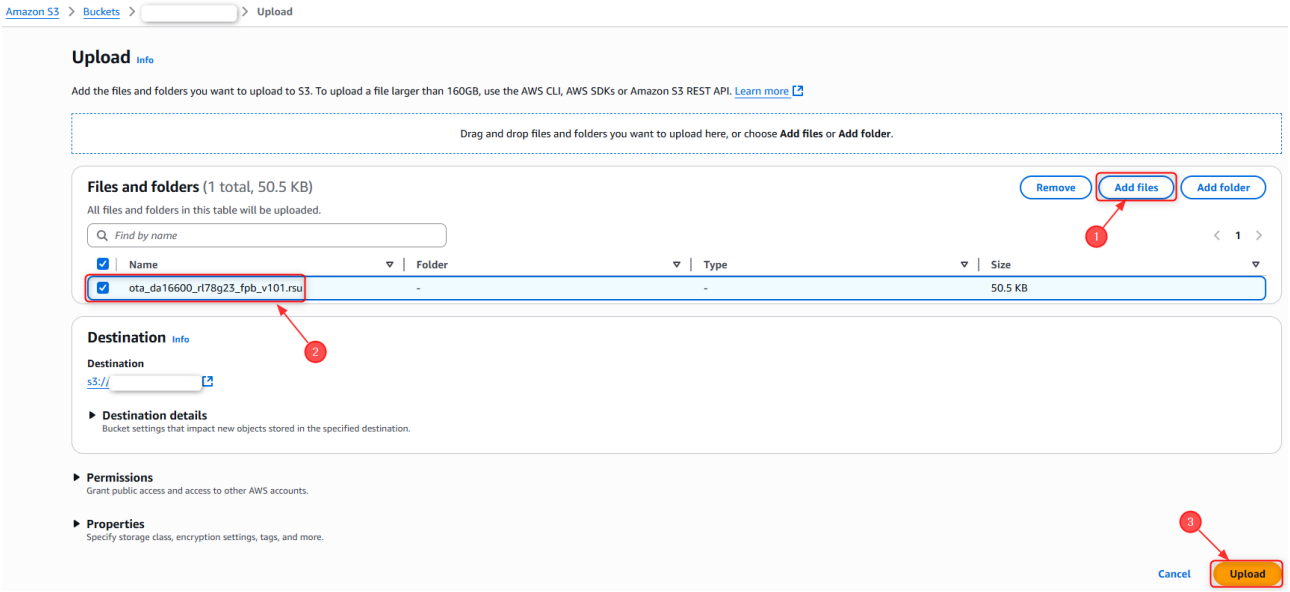

Figure 2.31 Add Firmware File

(3) Get the Object URL of the firmware

(1) On the **Objects** page of your S3 bucket, locate and select your firmware file (ota_da16600_rl78g23_fpb_v101.rsu).

Click **Copy S3 URL** to copy the file's download link.

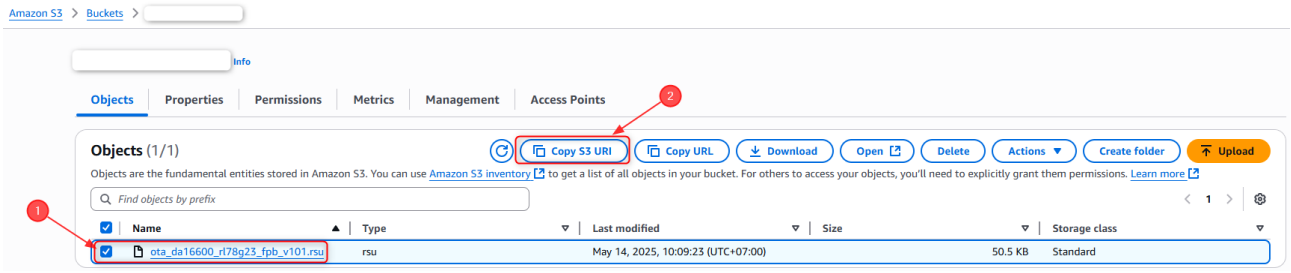

Figure 2.32 Get Firmware URL

(2) Open CloudShell and create presigned URL

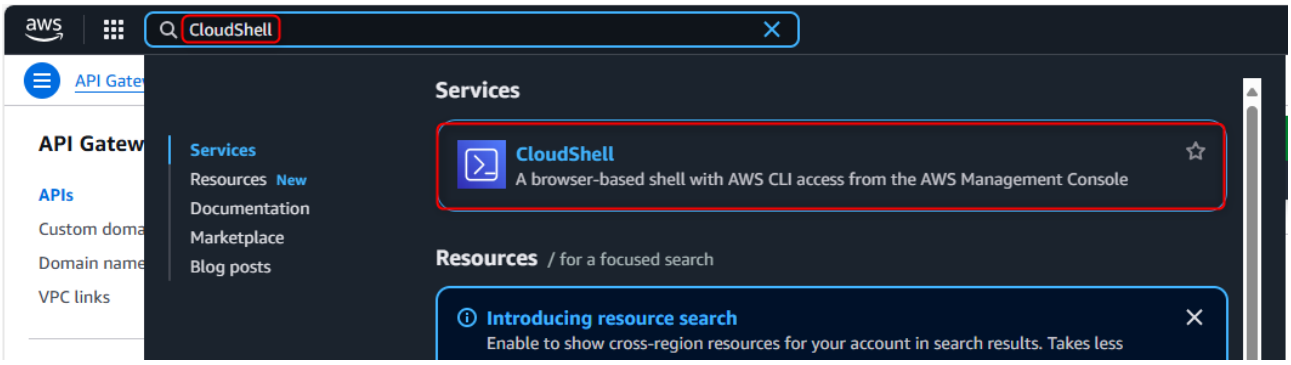

Figure 2.33 Open CloudShell

Execute the following command at the command prompt of CloudShell:

```
aws s3 presign <your-s3-url> --expires-in 600
```

Figure 2.34 Generate presigned URL

(3) Open API Gateway

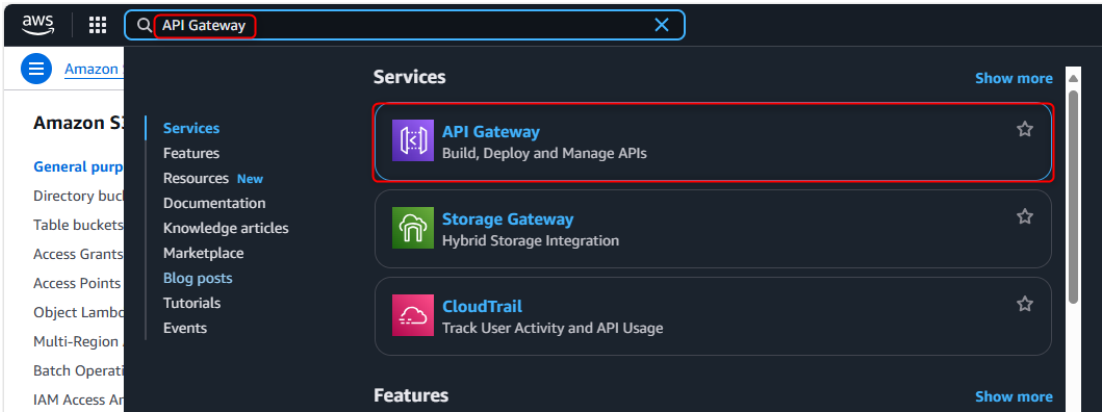

Figure 2.35 Open API Gateway

(4) Create short link for pre-signed URL

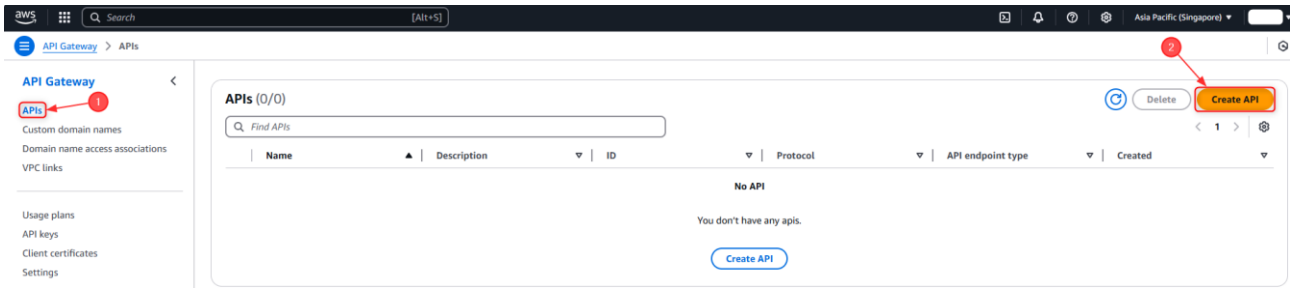

Figure 2.36 Create API

- Click on **Create API**

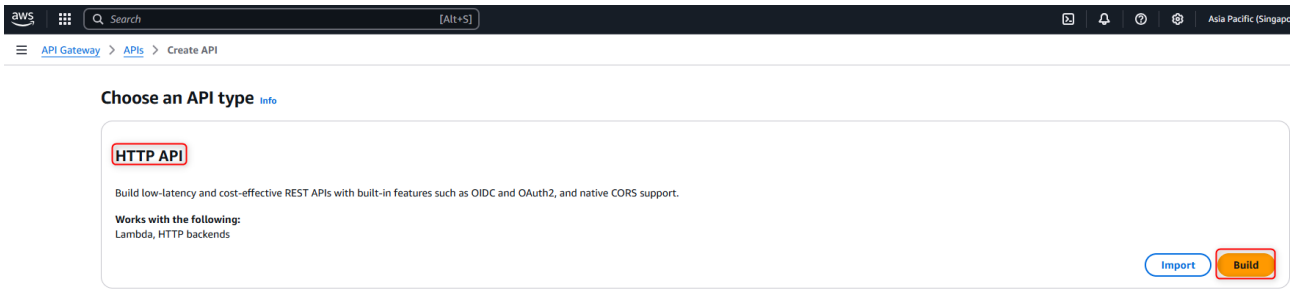

Figure 2.37 Create HTTP API

- Create **HTTP API**

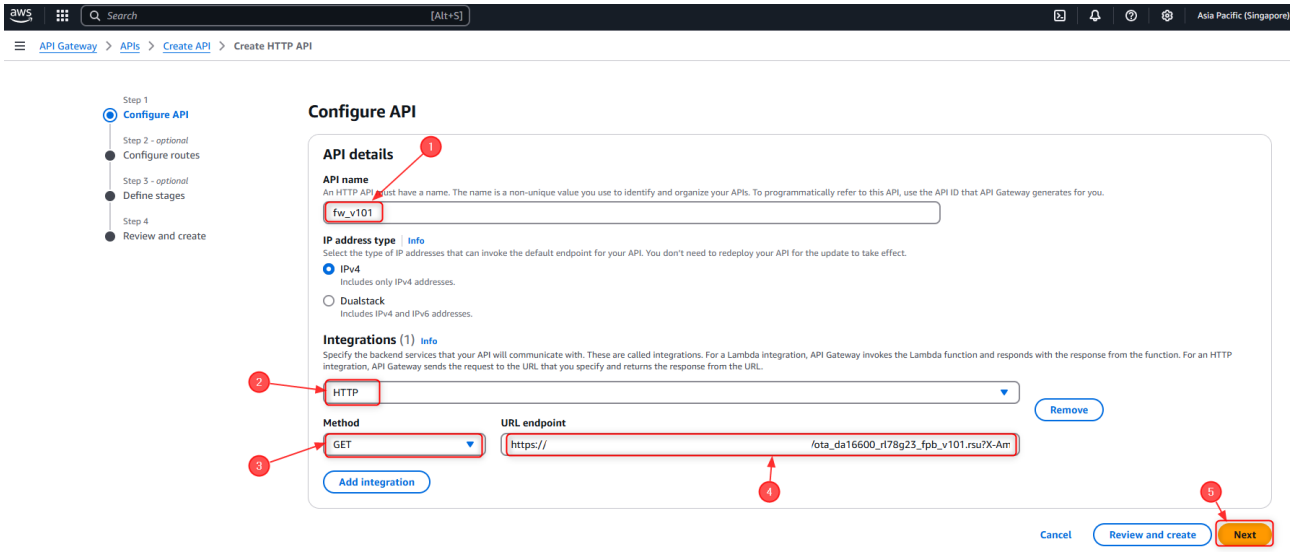

Figure 2.38 Configure API

- Step 1: Enter your API name.
- Step 2: Select HTTP.
- Step 3: Select the GET method.
- Step 4: Enter your generated pre-signed URL from **Section 2.2.7.2(3)(2)**.
- Step 5: Click Next to finish the API configuration.

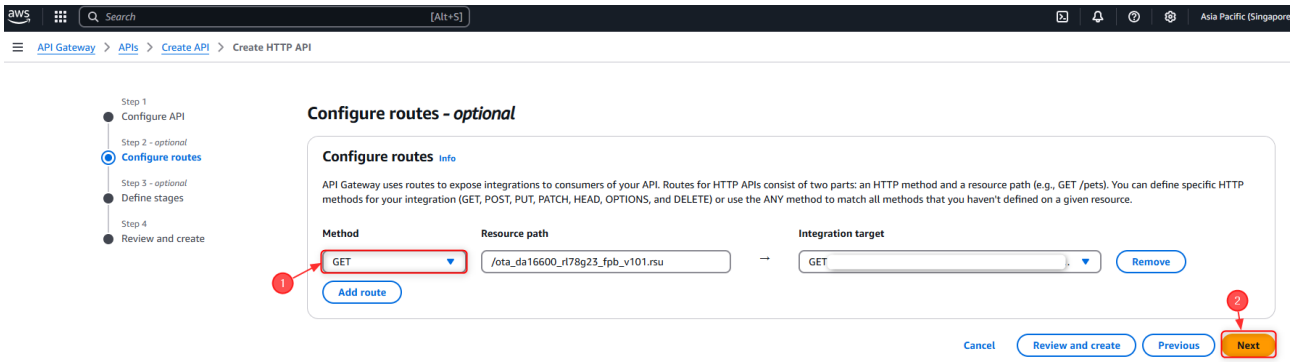

Figure 2.39 Configure routes

- Step 1: Select GET method.
- Step 2: Click Next to finish the routes configuration.

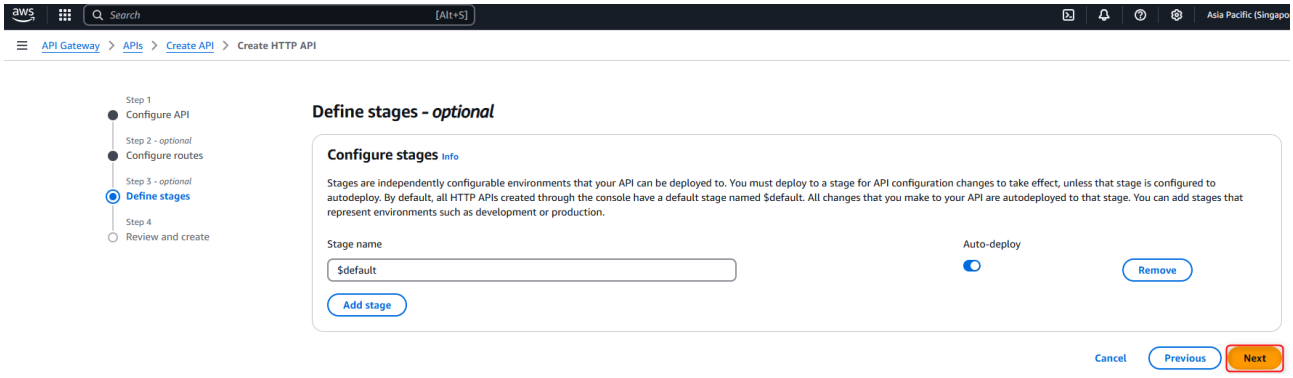

Figure 2.40 Define stages

- Let the “Define stages” by default, click Next to finish the define stages configuration.

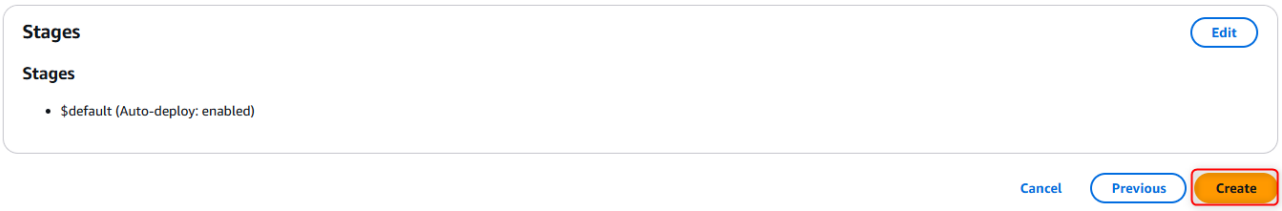

Figure 2.41 Review and create

- Click Create to finish creating the API.

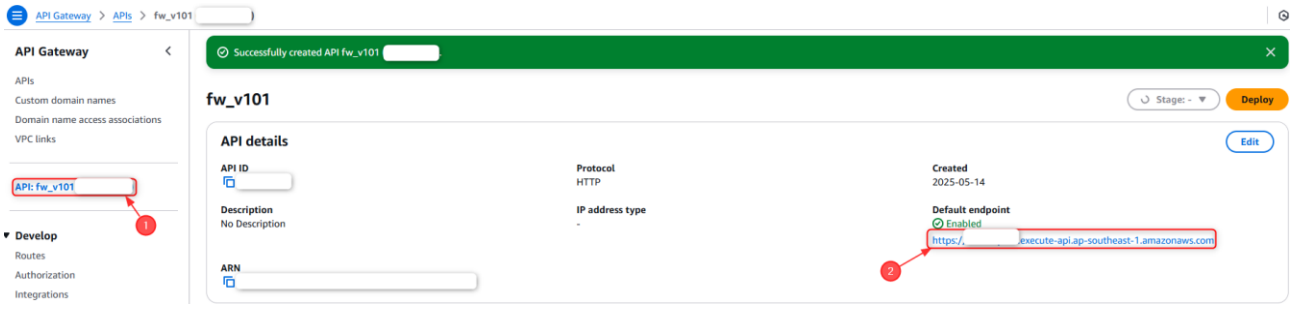

Figure 2.42 API details

- Copy default endpoint URL.

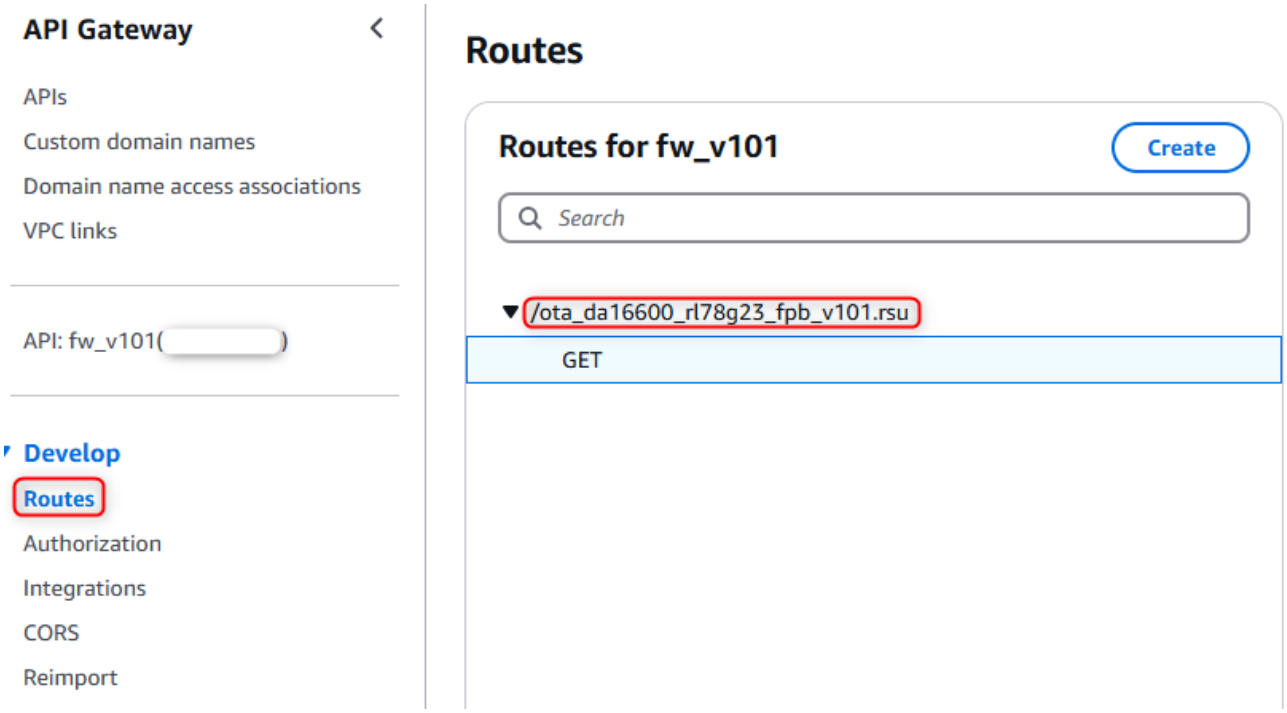

Figure 2.43 API routes

- Get API routes

Our final URL will have a form as below:

<your-default-endpoint>/<your-route> Example: https://xadwczzqe.execute-api.ap-southeast-1.amazonaws.com/ota_da16600_rl78g23_fpb_v101.rsu

Note: The obtained URL will be used to input from the terminal in the **Section 3.3(Figure 53)**, so make sure to save it**.**

3. Execute the Demonstration Project

3.1 Creating a New Project and Connecting to the MCU Board

(1) Start the Renesas Flash Programmer

- Microcontroller: RL78/G2x
- Project Name: Any (Example: rl78g23-fpb)
- Project Folder: Any
- Tool: COM port
- Interface: 2 wire UART
- Tool Details…: COM port number
- Click “Connect”

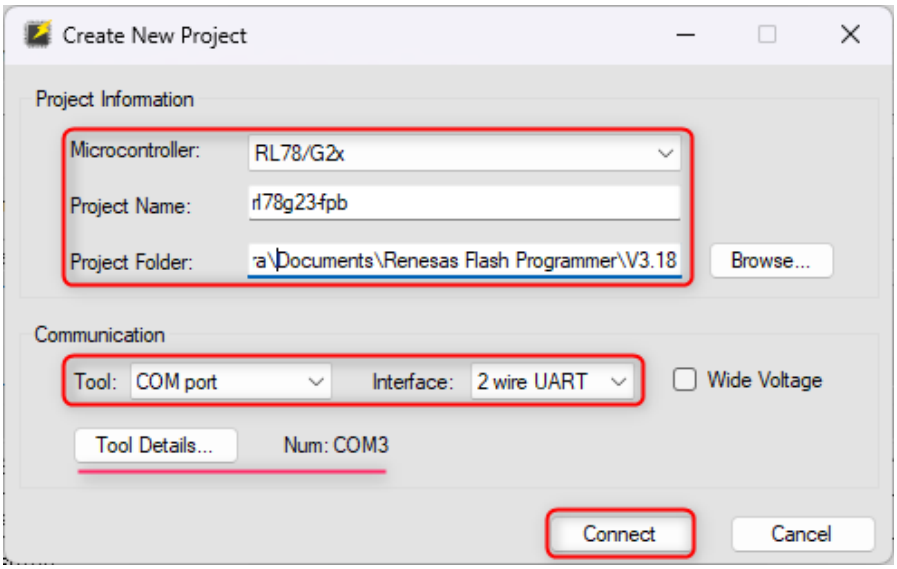

Figure 3.1 Creating a New Project and Connecting to the MCU Board

(2) The connection is successful if the following window appears.

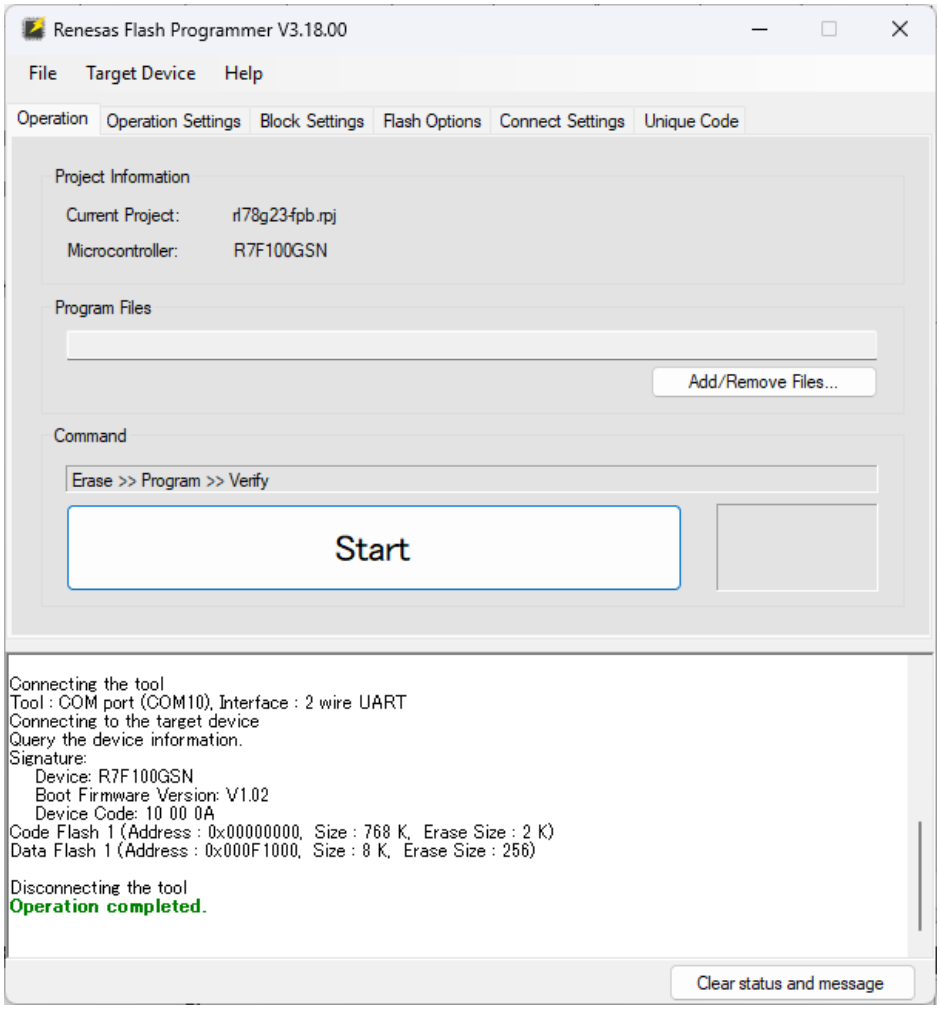

Figure 3.2 Operation completed (Connect)

3.2 Programming a MOT File to the MCU Board

(1) In the Program File field, enter the path to the MOT file to be programmed, and then click “Start”.

- Program File: MOT file to be programmed (Example: initial_image.mot, ota_da16600_rl78g23_fpb.mot)
- Click “Start”

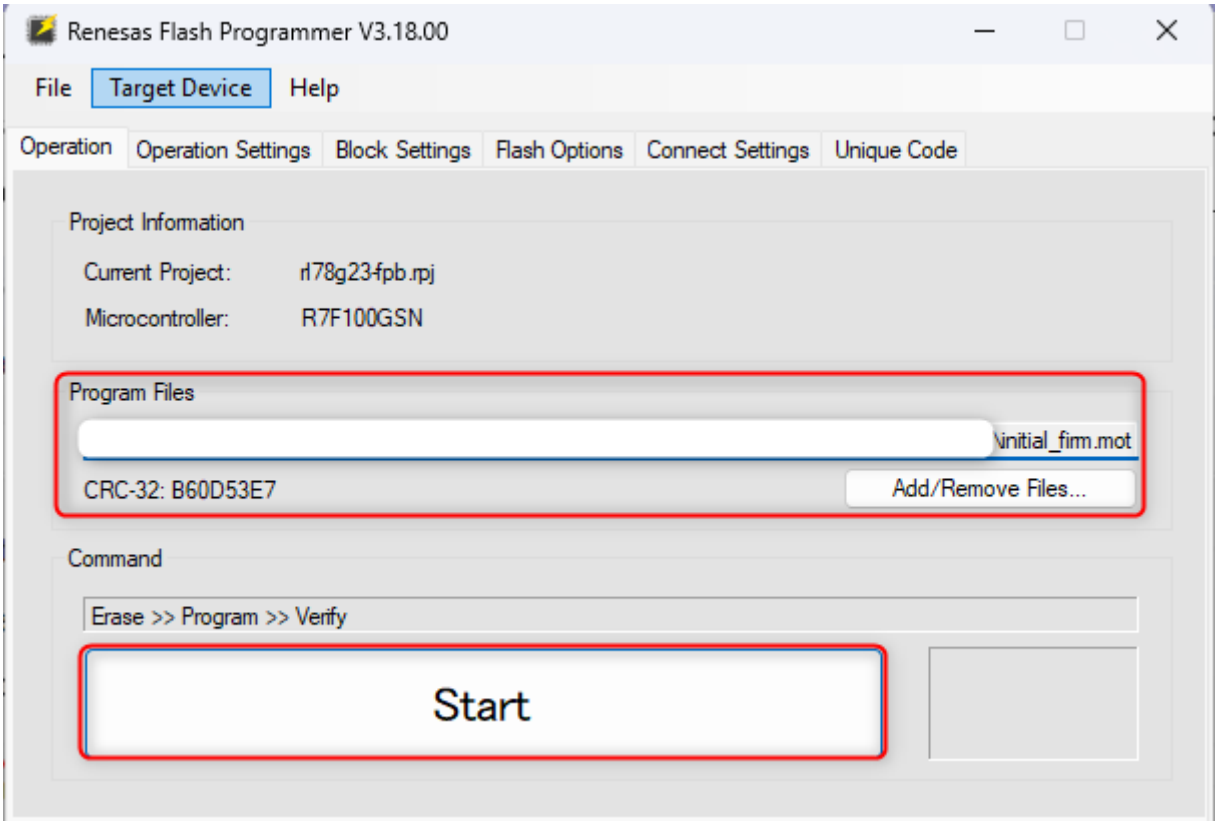

Figure 3.3 Programming a MOT File to the MCU Board

(2) Make sure that programming is successful.

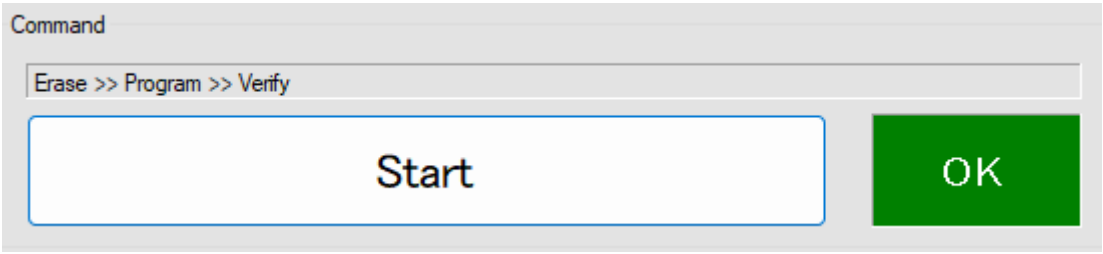

Figure 3.4 Successful programming

3.3 Request to update the firmware.

1. Waiting for the network connection to be successfully established (see **Figure 3.7**).

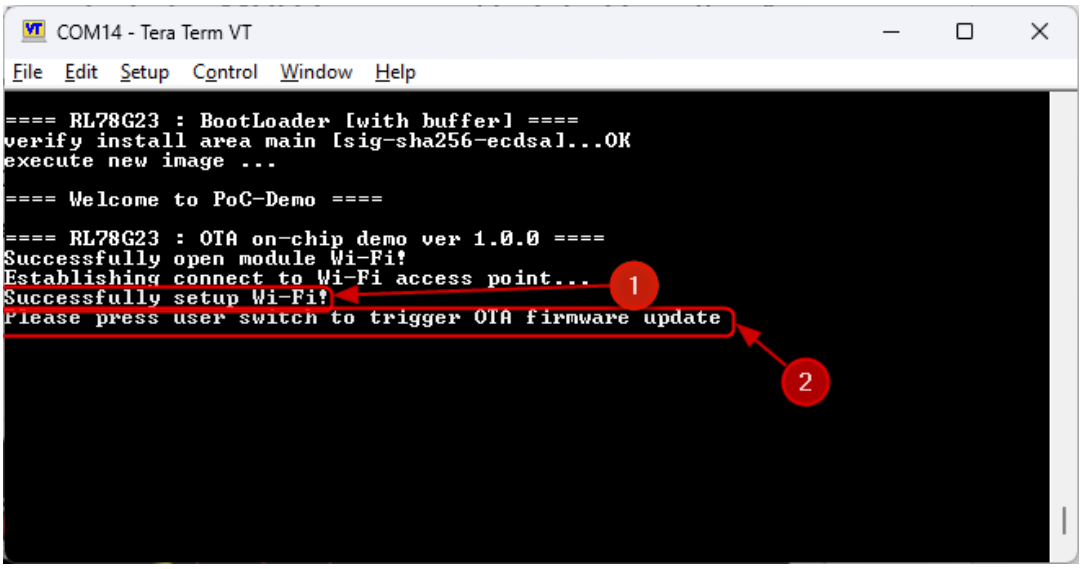

Figure 3.5 Successfully setup Wi-Fi

2. Press User Switch button to trigger OTA firmware update

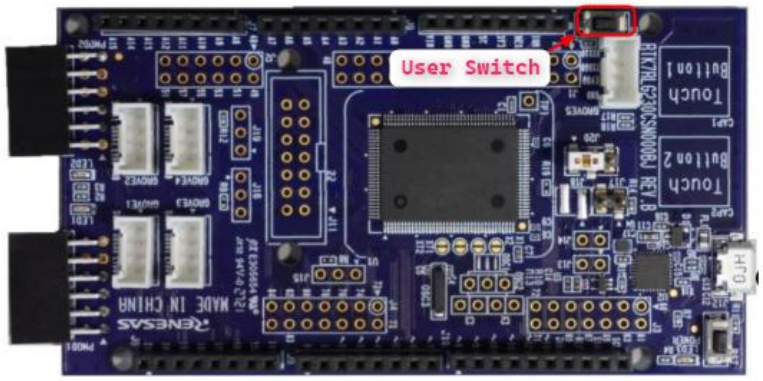

Figure 3.6 User Switch

3. After the user switch was pressed, input the firmware URL go get from **Section 2.2.7.2(3)**

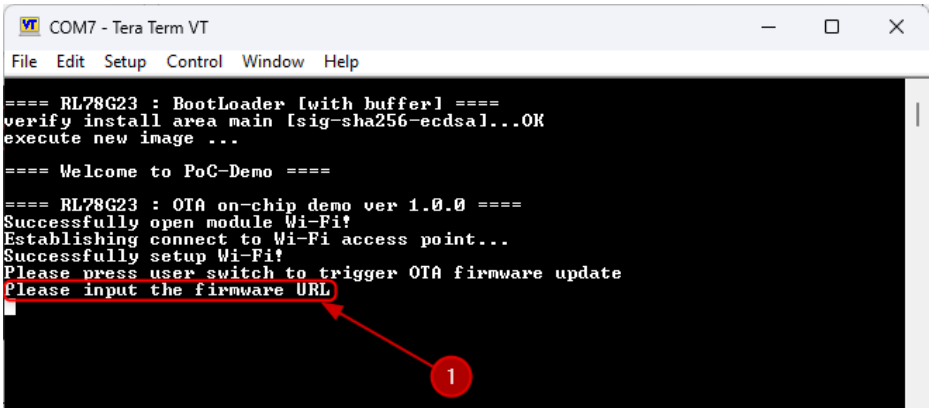

Figure 3.7 Wait for input firmware URL

Click the **Edit** tab of the Tera Term and "**Paste<CR>**" and verify and confirm the valid string and press OK.

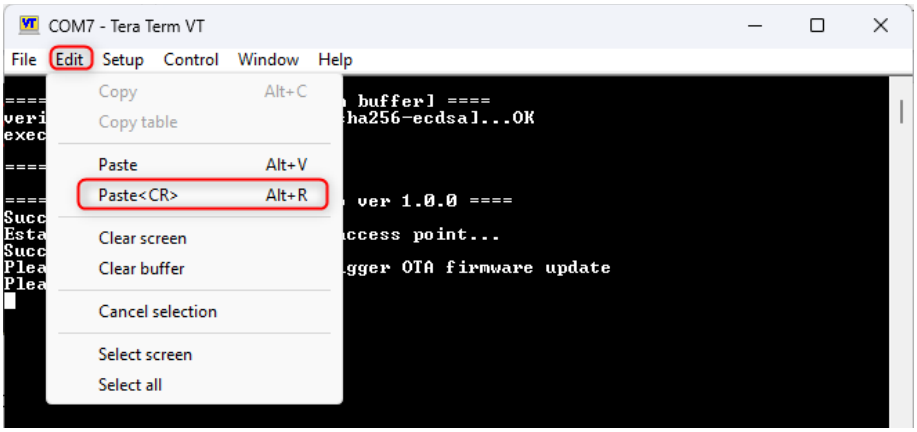

Figure 3.8 Input firmware URL (1)

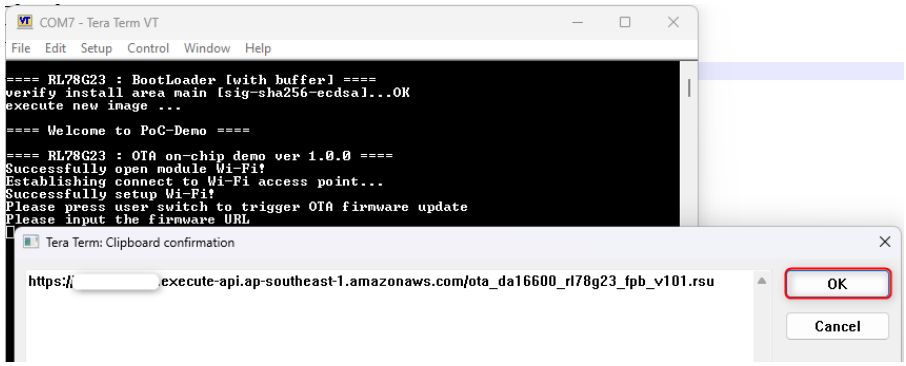

Figure 3.9 Input firmware URL (2)

3.4 Observe the Firmware Update Over-the-Air process

- The firmware update process starts with progress messages such as download progress, “W 0x59000, 128” (indicating that address of flash memory and block size that was written new firmware) displayed in the terminal. **Upon successful completion**, the firmware is updated from **1.0.0** to **1.0.1**. If an error occurs during the update, the MCU automatically resets, and the previous stable firmware version is retained.

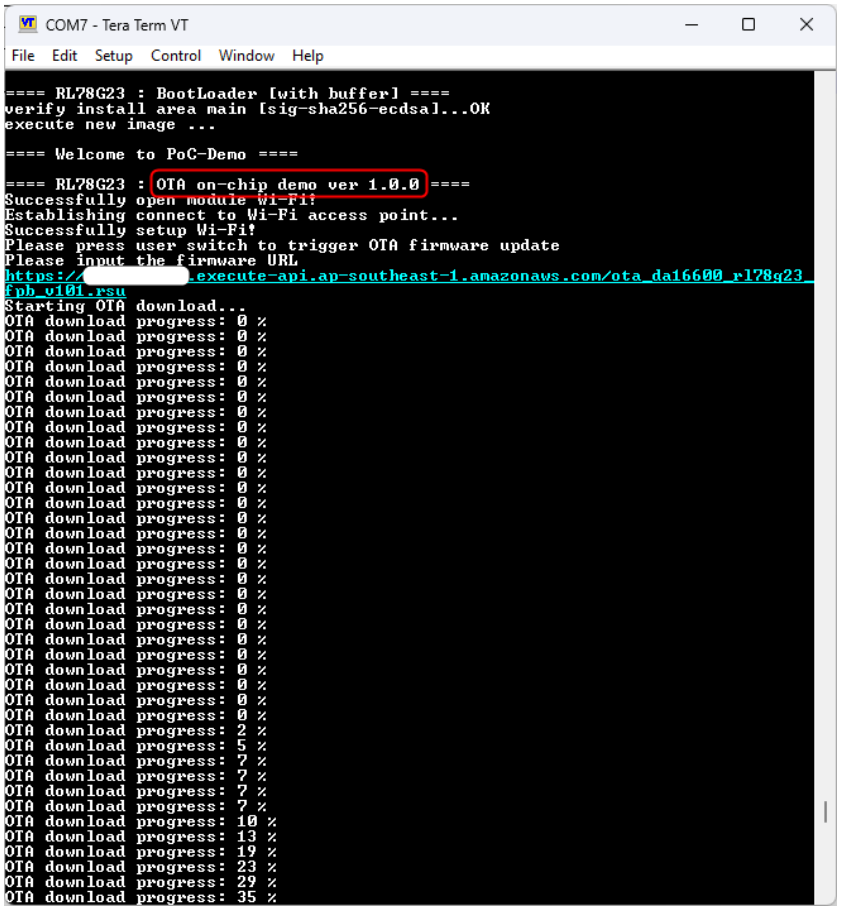

Figure 3.10 OTA Result (1)

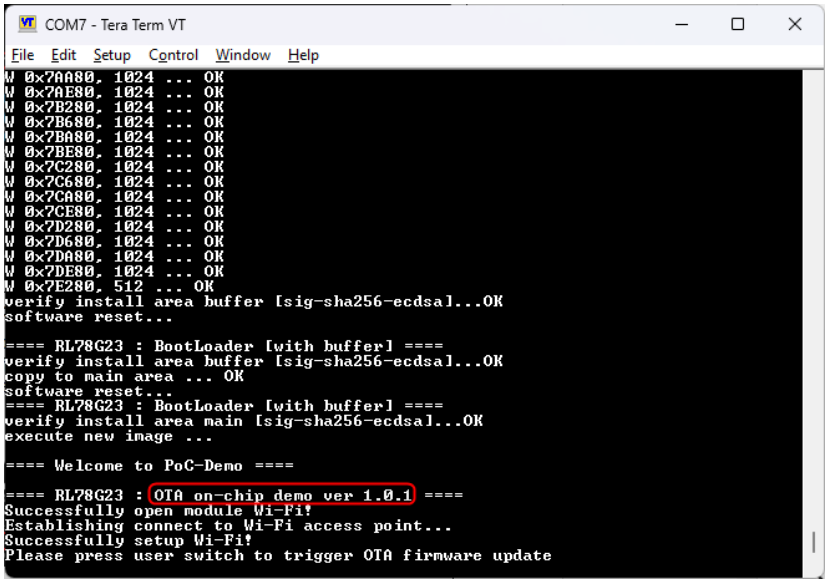

Figure 3.11 OTA Result (2)

Remark: Successfully updated the version from **1.0.0** to **1.0.1.**

4. Appendix

4.1 Known Issues for DA16600

4.1.1 Firmware Size Limit Causes Update Failure

- The DA16200/DA16600 SDK V3.2.9.2 restricts firmware downloads from the server to under 256 KB. Attempting to download firmware exceeding this limit results in an error, preventing successful completion. This affects users needing larger firmware updates.

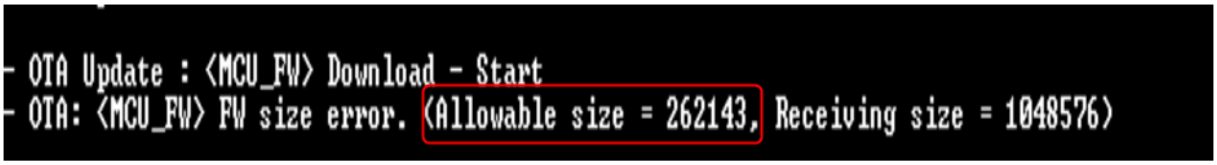

Figure 4.1 Error Log

4.1.2 Resolution

(1) Download **DA16200/DA16600 FreeRTOS SDK V3.2.9.2**

- Access the DA16XXX - Ultra-Low Power Wi-Fi SoC for Battery-Powered IoT Devices | Renesas to Download the **DA16200/DA16600 FreeRTOS SDK V3.2.9.2**

(2) Importing **DA16600 FreeRTOS SDK Project** into e2 studio.

- Importing the **DA16600 FreeRTOS SDK Project** into e2 studio is similar to importing **the DA16200 FreeRTOS SDK Project**. Therefore, please refer to UM-WI-056 DA16200 DA16600 FreeRTOS Getting Started Guide under **Section 5.4 Importing DA16200 FreeRTOS SDK Project** into e2 studio for detailed instructions on the process.

(3) Modify the project.

- In e2 studio Project Explorer, open the file **config_generic_sdk.h** in the folder **da16600\get_started\include\user_main** and modify the macro highlighted as below.

```
#if defined ( __SUPPORT_OTA__ )
#define __OTA_UPDATE_MCU_FW__
#endif //__SUPPORT_OTA__
```

- In e2 studio Project Explorer, open the file **da16200_map.h** in the folder **da16600\core\bsp\driver\include\DA16200** and modify the two macros highlighted as below.

```
/* DA14531 BLE Firmware Download start */

#define SFLASH_BLE_FW_BASE
(SFLASH_14531_BLE_AREA_START)



/* DA14531 BLE Security DB Area start */

#define SFLASH_USER_AREA_BLE_SECURITY_DB (SFLASH_BLE_FW_BASE +
__BLE_IMG_SIZE__)



/* SFLASH User Area */

#define SFLASH_USER_AREA_1_START        0x00600000

#define SFLASH_USER_AREA_1_END                  0x00800000
```

(4) Building project.

- Please refer to UM-WI-056 DA16200 DA16600 FreeRTOS Getting Started Guide under **Section 5.5 Building Projects** for detailed instructions on the process.

(5) Flash the new firmware.

- Flash the new firmware via Tera Term by following the instructions provided in the UM-WI-056 DA16200 DA16600 FreeRTOS Getting Started Guide under **Section 4.5.2 Using Macro Script of Tera Term**.

4.2 Debugging

printf() in the project is used to provide additional error information during the debugging process.

Revision History

Rev.	Date	Revision History	
		Page	Summary
1.00	May. 30, 2025	-	First edition issued

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 is supplied until the 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

1. 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 products, 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 Apromulgated 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.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. 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

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/.