

RX Family

How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

Introduction

This application note explains the procedures for using an OTA demo application under FreeRTOS with IoT Libraries.

Note: The procedures described in this application note are compatible with v202406.01-LTS-rx-1.1.0 and later versions of *iot-reference-rx*, a FreeRTOS reference project for RX microcontrollers.

When using v202210.01 or an earlier version of FreeRTOS, refer to *How to implement FreeRTOS OTA using Amazon Web Services in RX65N (for v202210.01-LTS-rx-1.1.3 or later)* ([R01AN7037](#)).

Note: The examples of implementations this application note describes are based on the CK-RX65N v2 board with Ethernet communications as the operating environment, but the examples can also be used with combinations of other boards and communications control modules, such as the RYZ014A PMOD module and DA16600 module (Wi-Fi).

For the combinations of boards and communications control modules, refer to the page at the following link.

[GitHub] https://github.com/renesas/iot-reference-rx/blob/main/Getting_Started_Guide.md#getting-started-guide

Note: Renesas has announced discontinuation of production of the existing LTE module with the part number RYZ014A and has announced the cessation of shipment of the product.

Accompanying cessation of shipment of the RYZ014A, the CK-RX65N v1 board will also no longer be shipped.

If the RYZ014A is in use for a current design or item in production, replace it with the Sequans product GM01Q, which has pin and functional compatibility with the RYZ014A.

The EOL notice for the RYZ014A is given at the following link.

[Link] <https://www.renesas.com/document/elc/plc-240004-end-life-eol-process-select-part-numbers>

[Product page] <https://www.renesas.com/products/wireless-connectivity/cellular-iot-modules/ryz014a-lte-cat-m1-cellular-iot-module>

Target Devices

RX65N and RX651 groups

Hardware

CK-RX65N v2

Contents

1. Overview	4
1.1 Overview of the System	4
1.2 Hardware Environment for Confirming Operation	5
1.3 Software Environment for Confirming Operation	5
2. Preparations in Advance.....	6
2.1 Installing Tera Term.....	6
2.2 Installing Python	8
2.3 Installing OpenSSL.....	9
2.4 Installing the Renesas Image Generator.....	10
2.5 Installing the AWS Command Line Interface (CLI)	12
2.6 Connecting the CK-RX65N v2.....	14
3. Setting Up AWS.....	17
3.1 Setting Up an Environment for Signing In to AWS.....	17
3.1.1 Signing In to the AWS Management Console.....	17
3.1.2 Setting Your Region in AWS	18
3.1.3 Creating a Workforce User of the IAM Identity Center.....	19
3.1.4 Preparing for an AWS CLI Sign-In	20
3.1.5 Creating a Work Folder	24
3.2 SSO Login to AWS through the CLI.....	25
3.3 Registering a Device with AWS.....	25
3.3.1 Setting Policies.....	25
3.3.2 Creating an Amazon S3 Bucket.....	27
3.3.3 Allocating Permission for OTA Execution to an IAM User	28
3.3.4 Registering a Device as a Thing in AWS IoT	33
4. Setting Up a Device	36
4.1 Generating a Key Pair and Certificate.....	36
4.2 Creating the Initial Version of the Firmware	39
4.2.1 Creating the Sample Projects.....	39
4.2.2 Setting Up Projects.....	52
4.2.3 Settings for the Individual Connectivity Types	54
4.2.4 Creating the Initial Firmware	56
4.2.5 Registering AWS IoT Information.....	61
5. Updating the Firmware	68
5.1 Creating the Update Firmware	68
5.1.1 Changing the Firmware Version.....	68

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

5.2	Updating the Firmware	69
5.2.1	Uploading the Update Firmware to AWS	69
5.2.2	Creating a Code Signing Certificate and Profile.....	71
5.2.3	Executing an OTA Job	73
6.	Appendix	78
6.1	Points to Keep in Mind when Operating Multiple Devices within the Same LAN Environment	78
7.	Troubleshooting.....	79
	Revision History.....	80

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

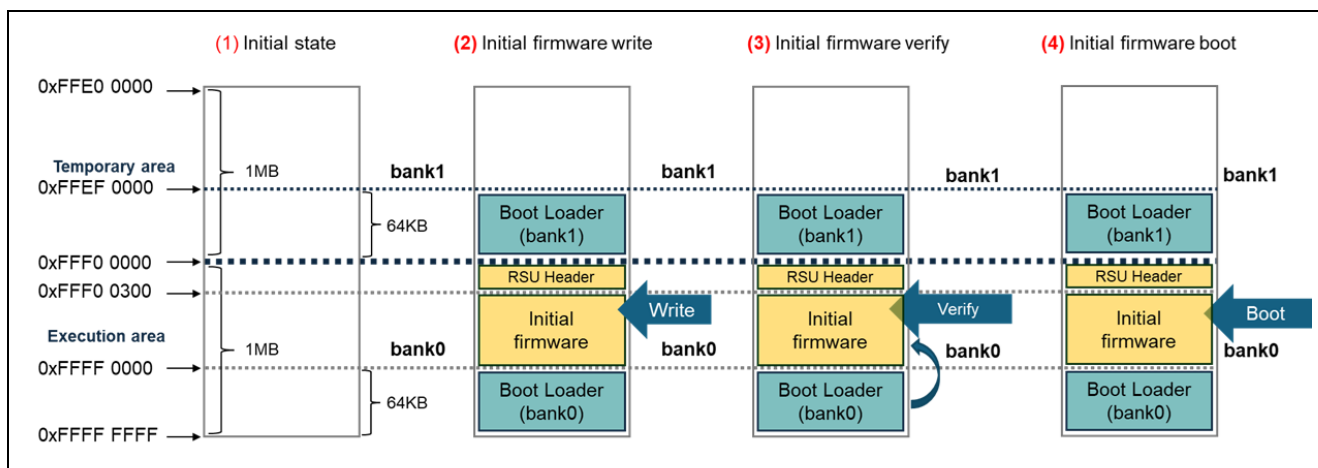
1. Overview

1.1 Overview of the System

This section outlines the operations for implementing OTA updates by using the RX65N microcontroller that is mounted on the CK-RX65N v2 board and supports the dual-bank function.

In a microcontroller with the dual-bank function, the ROM area can be divided into an execution area and a temporary area. By dynamically swapping the execution area and temporary area, the update firmware can be written to the temporary area while the existing version of firmware continues to run in the execution area.

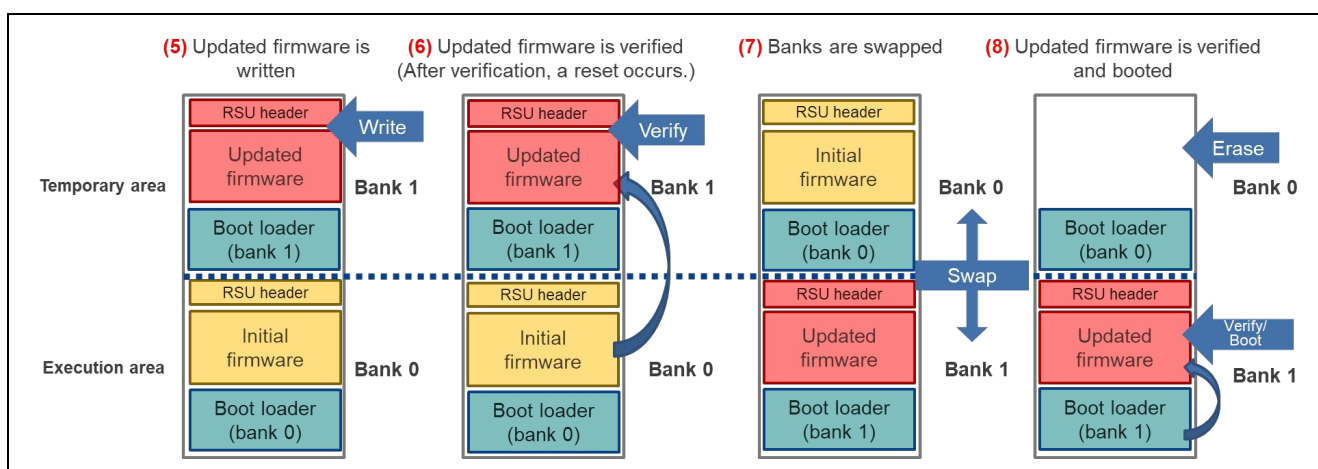
The following figures show memory allocation and how the memory areas are switched by bank swapping with the dual-bank function during an OTA update.



- (1) In this state, all data have been erased (blank state) by the Renesas Flash Programmer.
- (2) In this state, the data^{Note} obtained by combining the boot loader and initial firmware in the Renesas Flash Programmer are written.

Note: This refers to the data obtained by combining the boot loader (in bank 0), initial firmware, RSU header, and boot loader (in bank 1). For details on the RSU header, refer to “5.2 Image File” in *RX Family Firmware Update Module Using Firmware Integration Technology Application Note (R01AN6850)*.

- (3) After release from the reset state, the boot loader (in bank 0) verifies the initial firmware.
- (4) The initial firmware is booted up.



RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

- (5) Upon receiving the update firmware from Amazon Web Services (AWS), the update firmware is written to bank 1.

While this writing to bank 1 is in progress, operations by the initial firmware are executed through the BGO function.

- (6) The update firmware is verified by the initial firmware.
- (7) Bank 0 and bank 1 are swapped (bank swap), and bank 1 is designated as the execution area.
- (8) The update firmware is verified by the boot loader (in bank 1).

If verification fails, control is returned to the previous firmware and the previous firmware is booted up.

If verification succeeds, the update firmware that was written to bank 1 is booted up.

After the update firmware has been executed, a self-test proceeds. If the result of the self-test is a pass, the initial firmware in bank 0 is erased.

If the result of the self-test is not a pass, control is returned to the previous firmware and a reset is executed.

1.2 Hardware Environment for Confirming Operation

The hardware environment for confirming operation of the demo project is indicated in the table below.

Item	Product
Board	CK-RX65N v2 (Ethernet, Cellular, Wi-Fi)

1.3 Software Environment for Confirming Operation

The elements of the software environment for confirming operation of the demo project are listed in the table below.

Item	Product
Integrated Development Environment (IDE)	e ² studio 2025-12
Compiler	Renesas CC-RX v3.07.00 GCC for Renesas RX v14.2.0.202505
FreeRTOS	v202406.01-LTS-rx-1.1.1
Log monitoring tool	Tera Term v5.50
Python	Python 3.11.0
Keygen tool	Win64 OpenSSL v3.6.1
Flash memory programming tool	Renesas Flash Programmer V3.22.00
Renesas Image Generator	Version 3.03 (supplied with Firmware Update Module Rev.2.04)
AWS command line interface (CLI)	AWS Command Line Interface version 2

2. Preparations in Advance

2.1 Installing Tera Term

(1) Access the Tera Term download site.

Access the Tera Term download site at the following link.

[Tera Term download site \(GitHub\)](#)

Tera Term 5.5.0 Latest

Tera Term (Ver 5.5.0), TTSSH (Ver 3.5.0)

主な変更点

- x64, arm64 バイナリを作成するようにした。 (issue [#228](#))
- シリアルポートの送信待ち時間を設定するマクロコマンド `setserialdelaychar`, `setserialdelayline` を追加した。 (issue [#437](#))
- SSH2 の `curve25519-sha256`, [curve25519-sha256@libssh.org](#) 鍵交換方式に対応した。 (issue [#89](#))
- 既存の言語ファイルのファイル名を変更した。 (issue [#825](#))
 - 以前の言語ファイル名の指定は無効になります。
 - 個人の設定ファイルフォルダにある設定ファイルの内容はインストーラによって変更されません。 UIの設定から使いたい言語を選択して、設定を保存し直してください。

すべての変更については、変更履歴を確認してください。

https://teratermproject.github.io/manual/5/ja/about/history.html#teraterm_5.5.0

Major changes

- Now x64 and arm64 binaries are provided. (issue [#228](#))
- Added macro commands `setserialdelaychar` and `setserialdelayline` to set transmission delays for the serial port. (issue [#437](#))
- Added support for SSH2 key exchange methods: `curve25519-sha256`, [curve25519-sha256@libssh.org](#) (issue [#89](#))
- Changed filenames of the existing language files. (issue [#825](#))
 - Now the previous language filename is invalid.
 - The setup file in user's setup files folder is not modified by the installer. Please select the language you want to use in the UI settings, and save the setup file explicitly.

To check all changes, see the changelog.























https://teratermproject.github.io/manual/5/en/about/history.html#teraterm_5.5.0

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

- (2) Download the Tera Term installer.

Please select and download the latest version of the Tera Term installer that is compatible with your operating system.

▼ Assets 13

 teraterm-5.5.0-arm64.exe	sha256:b4e443c6f4936fa8a...		10.5 MB	2 weeks ago
 teraterm-5.5.0-arm64.zip	sha256:cb993ead5d50aa823...		14.3 MB	2 weeks ago
 teraterm-5.5.0-arm64_pdb.zip	sha256:2fba5c707b876bf8f...		29.7 MB	2 weeks ago
 teraterm-5.5.0-x64.exe	sha256:a34a110ab559df72d...		10.8 MB	2 weeks ago
 teraterm-5.5.0-x64.zip	sha256:ab4cfe73de8f9162c...		14.9 MB	2 weeks ago
 teraterm-5.5.0-x64_pdb.zip	sha256:4580bab7d9c7408ec...		30.5 MB	2 weeks ago
 teraterm-5.5.0-x86.exe	sha256:d722f67fa6f990e63...		10.5 MB	2 weeks ago
 teraterm-5.5.0-x86.zip	sha256:c354287e54e61c010...		13.9 MB	2 weeks ago
 teraterm-5.5.0-x86_pdb.zip	sha256:6456a99c937753ba1...		30.1 MB	2 weeks ago
 teraterm-5.5.0.sha256sum	sha256:758901c3322ed3bc4...		819 Bytes	2 weeks ago
 Source code (zip)				2 weeks ago
 Source code (tar.gz)				2 weeks ago
Show all 13 assets				

- (3) Run the installer.

Run the installer and follow the prompts to install Tera Term.
Execute the installer with administrative permissions.

- (4) Confirm that Tera Term starts up.

Click on the Tera Term icon in the Start menu and confirm that Tera Term starts.

2.2 Installing Python

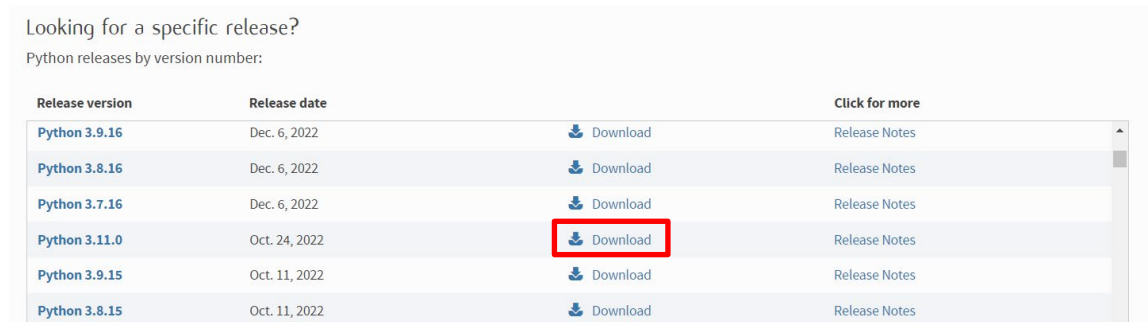
- (1) Access the Python download site.

Access the Python download site at the following link.

[Python download site](#)

- (2) Download the Python installer.

Click on the “Download” link for Python 3.11.0 in the version list.



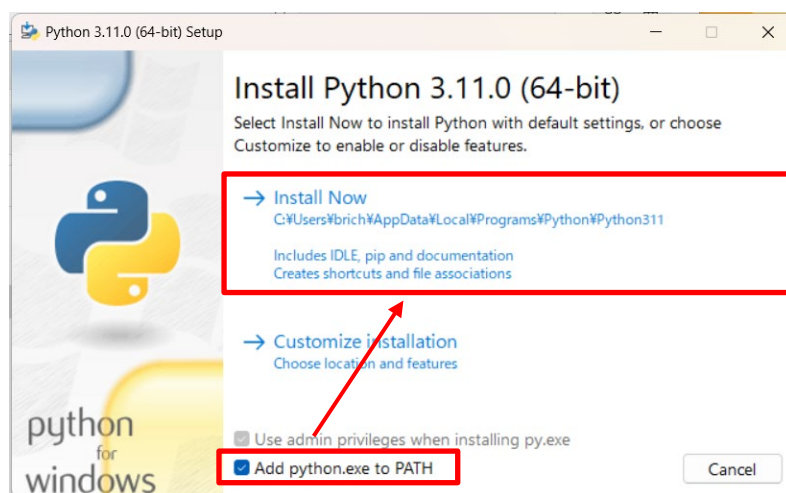
Download the installer for the OS you are using.

Version	Operating System	Description	MD5 Sum	File Size	GPG	Sigstore	
Gzipped source tarball	Source release		c5f77f1ea256dc5bdb0897eeb4d35bb0	26333656	SIG	CRT	SIG
XZ compressed source tarball	Source release		fe92acfa0db9b9f5044958edb451d463	19819768	SIG	CRT	SIG
macOS 64-bit universal2 installer	macOS	for macOS 10.9 and later	98fa94815780c9330fc2154559365834	42602603	SIG	CRT	SIG
Windows embeddable package (32-bit)	Windows		0888959642cc8af087d88da3866490a5	9560053	SIG	CRT	SIG
Windows embeddable package (64-bit)	Windows		7df0f4244e5a66760b7caaed58e86c93	10545380	SIG	CRT	SIG
Windows embeddable package (ARM64)	Windows		e3dbbd5d63c6cb203adc6c0c8ca5f5f7	9765886	SIG	CRT	SIG
Windows installer (32-bit)	Windows		e369a267acaad62487223bd835279bb9	23987136	SIG	CRT	SIG
Windows installer (64-bit)	Windows	Recommended	4fe11b2b0bb0c744cf74aff537f7cd7f	25157416	SIG	CRT	SIG
Windows installer (ARM64)	Windows	Experimental	18e5bd9a4854109adf3b77c7c9dc1ded	24289144	SIG	CRT	SIG

- (3) Install Python.

Run the installer and follow the prompts to install Python.

In the installation window, tick the [Add python.exe to PATH] checkbox.



RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

- (4) Confirm the installation of Python.

Open the command prompt window, and confirm that Python 3.11.0 is installed.
Execute the following command and confirm that the version information is displayed.

```
> python -V
```

Note: "V" must be entered as a capital letter.

After the command has been executed, the display in the window will be as shown below.

```
C:\Users>python -V  
Python 3.11.0
```

- (5) Install an encryption library for Python.

Install the encryption library "pycryptodome" for Python.
Execute the following command to install the encryption library.

```
> pip install pycryptodome
```

After the command has been executed, the display in the window will be as shown below.

```
C:\Users>pip install pycryptodome  
Requirement already satisfied: pycryptodome in c:\users\██████████\appdata\local\programs\python\python311\lib\site-packages (3.19.0)  
  
[notice] A new release of pip available: 22.3 -> 25.3  
[notice] To update, run: python.exe -m pip install --upgrade pip
```

2.3 Installing OpenSSL

- (1) Access the OpenSSL download site.

Access the Win32/Win64 OpenSSL download site at the following link.

<http://www.slproweb.com/win32-win64-openssl-installer-for-windows>

- (2) Download the OpenSSL installer.

Download the OpenSSL installer.
Download the installer for the OS you are using.

File	Type	Description
Win64 OpenSSL v3.6.1 Light EXE MSI	6MB Installer	Installs the most commonly used essentials of Win64 OpenSSL v3.6.1 (Recommended for users by the creators of OpenSSL). Only installs on 64-bit versions of Windows and targets Intel x64 chipsets. Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
Win64 OpenSSL v3.6.1 EXE MSI	248MB Installer	Installs Win64 OpenSSL v3.6.1 (Recommended for software developers by the creators of OpenSSL). Only installs on 64-bit versions of Windows and targets Intel x64 chipsets. Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
Win32 OpenSSL v3.6.1 Light EXE MSI	5MB Installer	Installs the most commonly used essentials of Win32 OpenSSL v3.6.1 (Only install this if you need 32-bit OpenSSL for Windows). Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.

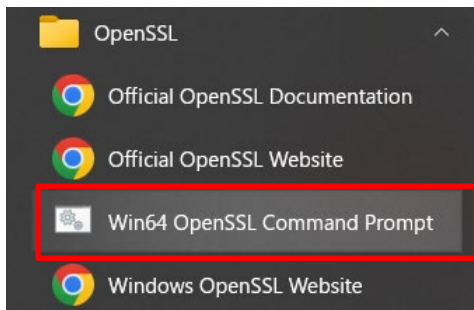
RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(3) Run the OpenSSL installer.

Run the installer and follow the prompts to install OpenSSL.
Select [The OpenSSL binaries directory] as the location to save the OpenSSL DLLs.

(4) Open the OpenSSL command prompt window.

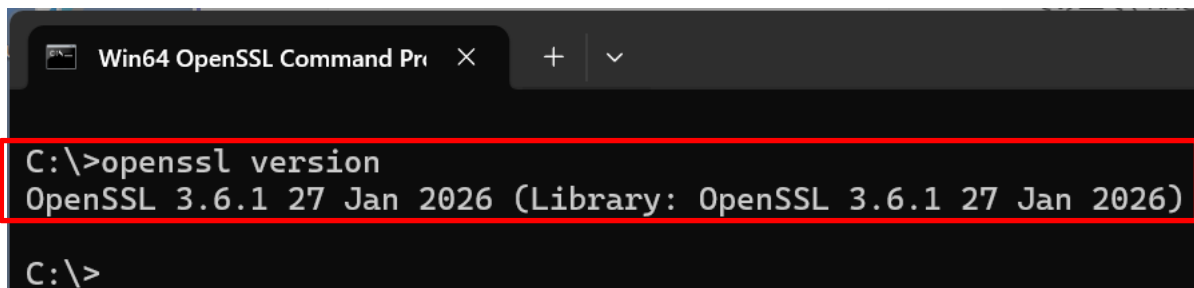
Select [Win64 OpenSSL Command Prompt] in the Start menu.



(5) Confirm the execution of OpenSSL.

Confirm that the openssl command can be executed at the command prompt.
Execute the following command and confirm that the version information is displayed.

```
> openssl version
```



2.4 Installing the Renesas Image Generator

The Renesas Image Generator is a utility tool for creating firmware images to be used by the firmware update module. The Renesas Image Generator can generate the following types of images for use by the module.

- Initial image: An image file (name extension: mot) containing the boot loader and application program. It is programmed with a flash programmer at the time of initially setting up the system.
- Update image: An image file (name extension: rsu) containing the update firmware.

Note: Firmware Update Module Rev.2.00 and later versions only support the generation of firmware images by using Python scripts.

The Renesas Image Generator is supplied as part of Firmware Update Module Rev.2.04. The Renesas Image Generator can be obtained through the following steps.

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

- (1) Open the Renesas Image Generator installation folder.

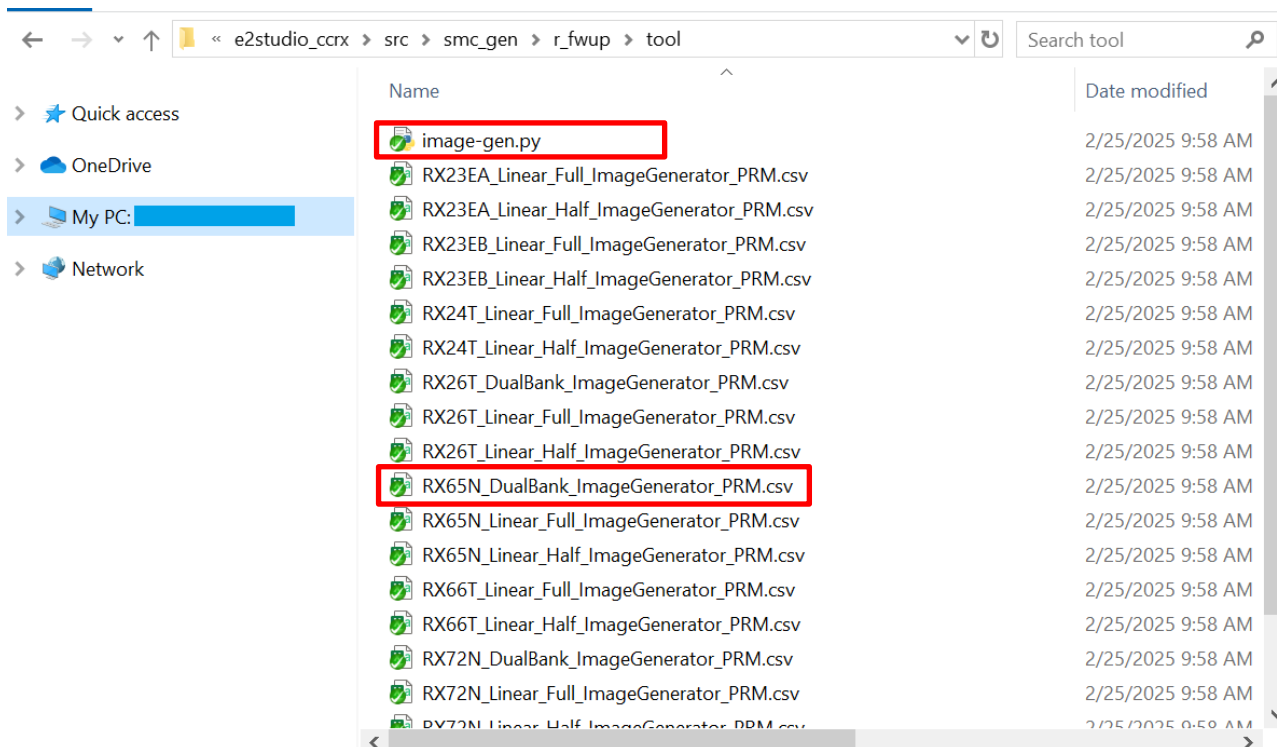
After importing the sample project, open the following folder.
For the procedure for importing the sample project, see 4.2.1.

```
\iot-reference-  
rx\Projects\aws_ether_ck_rx65n_v2\e2studio_ccrx\src\smc_gen\r_fwup\tool
```

The above pathname consists of the folder and project names for the example of a case of importing the project given in 4.2.1.

- (2) Confirm the Renesas Image Generator folder.

The “tool” folder contains the Renesas Image Generator script file (image-gen.py) and parameter files (*_ImageGenerator_PRM.csv) for the various device types, as shown below.



- (3) Confirm the files to be used.

Confirm the files to be used in the Renesas Image Generator.
The following files are used in this application note.

- image-gen.py: Renesas Image Generator script file
- RX65N_DualBank_ImageGenerator_PRM.csv: Parameter file for the RX65N dual-bank function

Create a folder with the name “Renesas Image Generator” in a desired folder, and copy the above files to that folder.

2.5 Installing the AWS Command Line Interface (CLI)

The AWS command line interface (CLI) is a tool for managing and operating the various Amazon Web Services with the use of commands from a command line shell.

This application note provides guidance on making AWS settings with operations through the AWS CLI.

Note: AWS CLI version 1 and AWS CLI version 2 are not compatible with each other. Be sure to install version 2.

If version 1 has already been installed, update it to version 2 with reference to the following Web page:

[Migration from AWS CLI version 1 to AWS CLI version 2](#)

(1) Execute the command line shell.

Execute the Windows PowerShell or Windows command prompt. The PowerShell is used for the display examples explained in this application note.

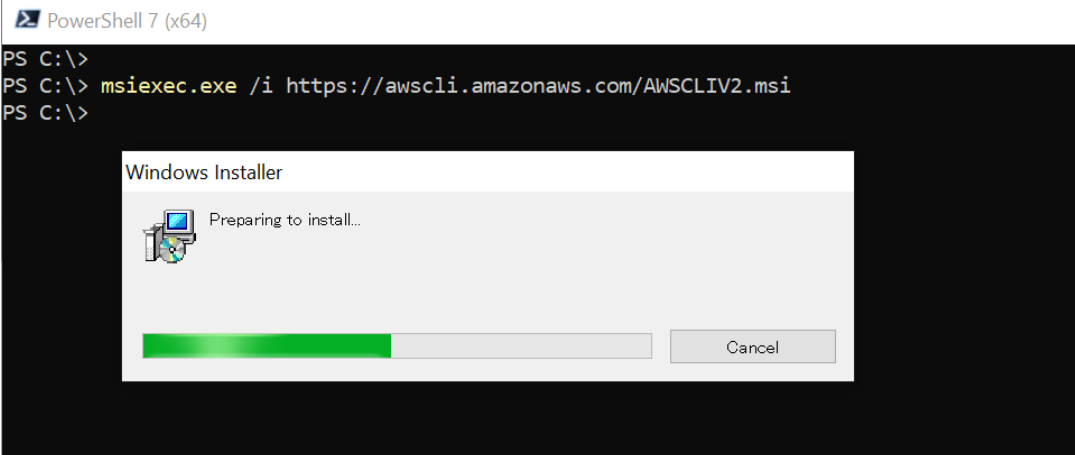
The command line shell is referred to as the command prompt in this application note.

Note: When using PowerShell, it is recommended to use Version 7 or later.

(2) Install the AWS CLI.

Enter the following command at the command prompt. This leads to the automatic downloading and execution of the AWS CLI v2 installer.

```
> msisexec.exe /i https://awscli.amazonaws.com/AWSCLIV2.msi
```



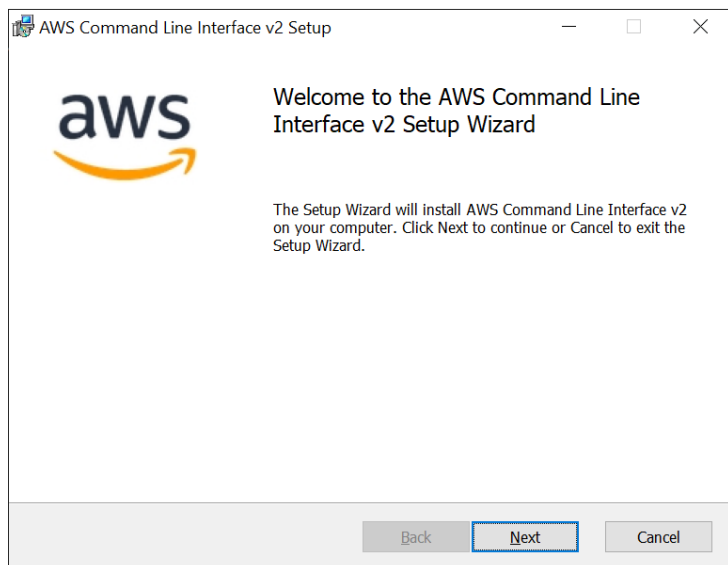
The steps for download and installation can be referred to on the following Web page.

[AWS CLI install and update instructions](#)

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(3) Run the installer.

Run the installer and wait for a while until the [Next] button becomes selectable. After clicking on the [Next] button, follow the prompts to install the AWS CLI. After you have finished the installation, restart the command prompt window.



RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(4) Confirm the installation.

After installation is completed, enter the following command at the command prompt. The version of the installed AWS CLI is displayed.

```
> aws --version
```

PowerShell 7 (x64)

```
PS C:\>
PS C:\> aws --version
aws-cli/2.24.9 Python/3.12.6 Windows/10 exe/AMD64
PS C:\> _
```

After you have confirmed that the installed version is 2.xx.x, the installation procedure is complete.

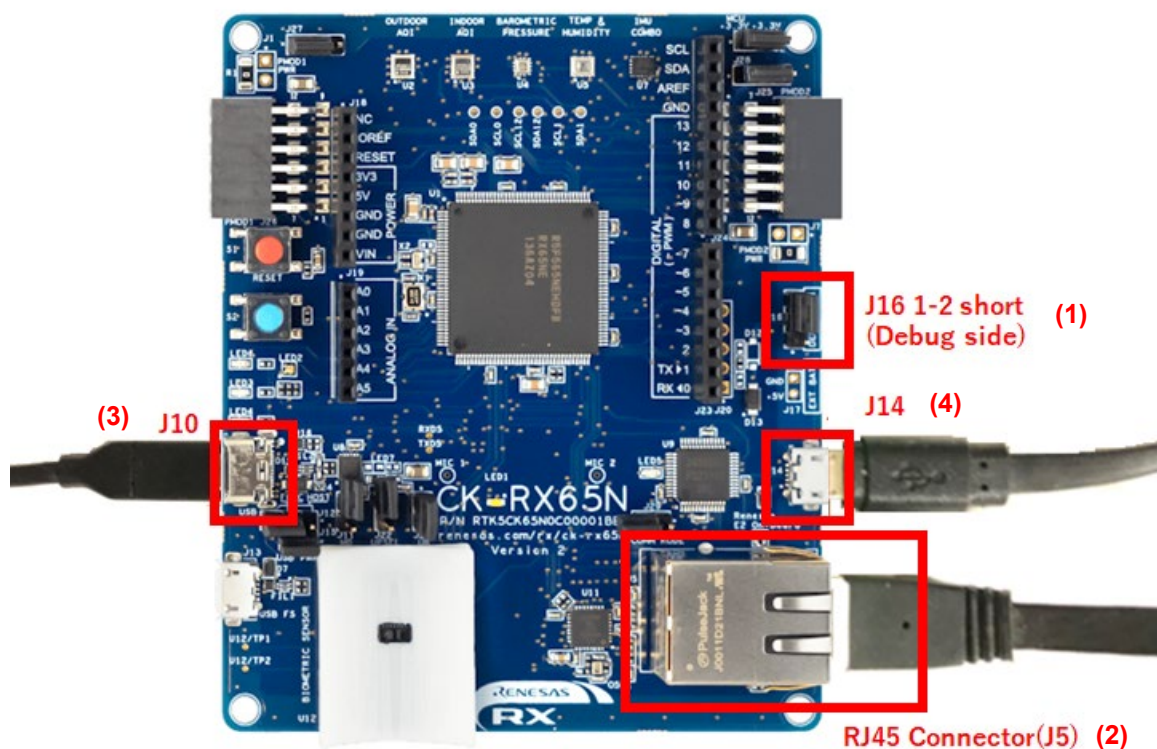
Note: The Python version displayed here indicates the version of the Python libraries embedded in the AWS CLI executable.

It may differ from the Python version installed on your system, but this does not cause any issues.

2.6 Connecting the CK-RX65N v2

(1) For Ethernet connection

Connect the base board and set the jumper blocks.



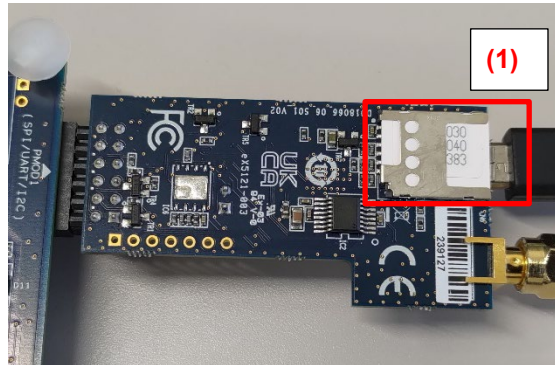
CK-RX65N v2 Base Board

- (1) Short pins 1-2 of jumper block J16 on the base board (enables debugging).
- (2) Connect a LAN cable to the RJ45 connector (J5) on the base board (Ethernet connection).
- (3) Connect J10 on the base board to a PC via a USB cable (USB serial connection).
- (4) Connect J14 on the base board to a PC via a USB cable (debugger connection).

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

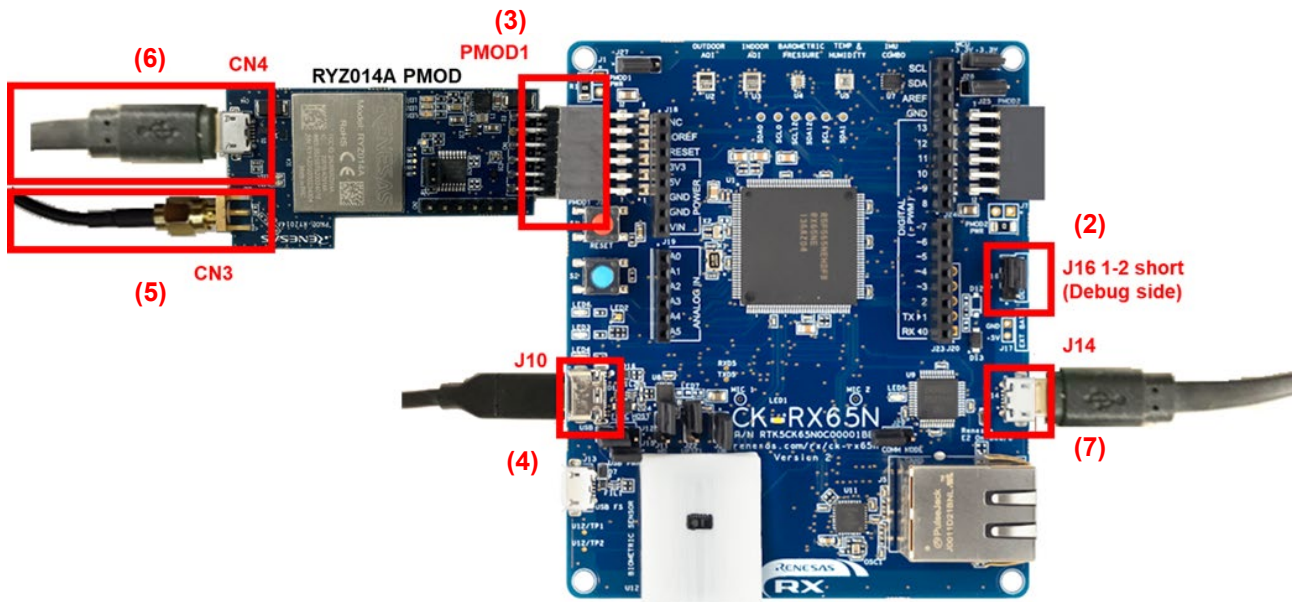
(2) For cellular connection (RYZ014A)

For cellular communications with the use of the RYZ014A, refer to the following connections.



Back of the RYZ014A PMOD Module

(1) Insert a SIM card into the CN6 slot on the RYZ014A PMOD module.



Top Surfaces of the Base Board and RYZ014A PMOD Module

- (2) Short pins 1-2 of jumper block J16 on the base board (enables debugging).
- (3) Connect the RYZ014A PMOD module to PMOD1 on the base board.
- (4) Connect J10 on the base board to a PC via a USB cable (USB serial connection).
- (5) Connect an antenna to CN3 of the RYZ014A PMOD module.
- (6) Supply power by connecting a USB cable to CN4 of the RYZ014A PMOD module.
- (7) Connect J14 on the base board to a PC via a USB cable (debugger connection).

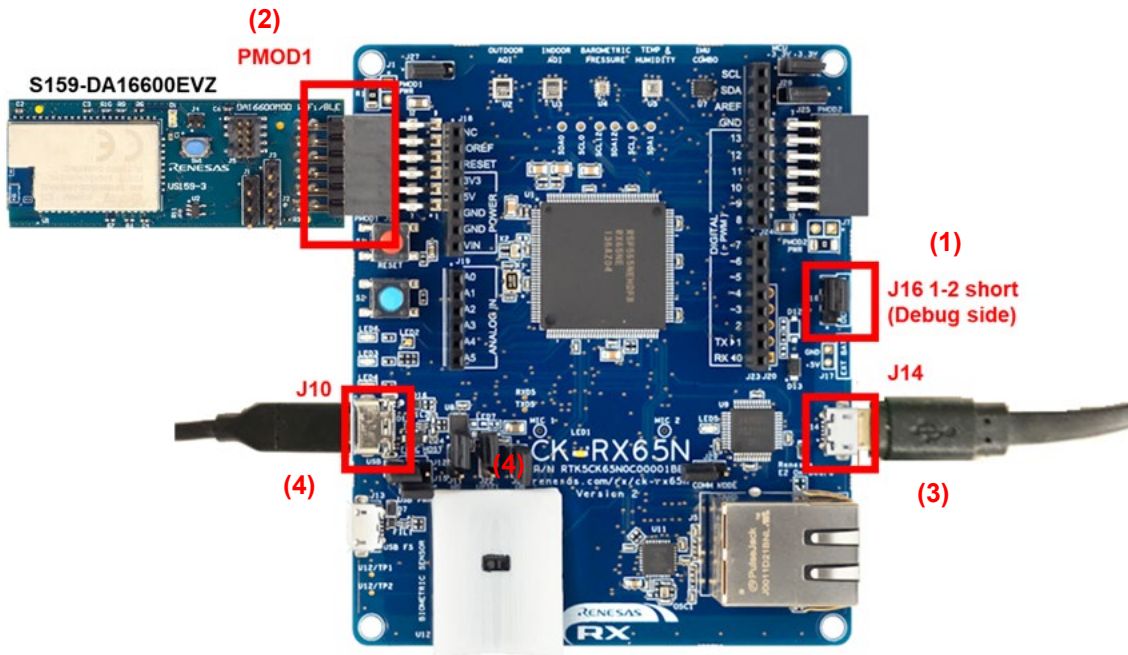
Note: Perform step (6) if you have a spare USB cable.

If power is not directly supplied to the RYZ014A PMOD module, communications might become unstable.

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(3) For Wi-Fi connection (DA16600)

For Wi-Fi communications with the use of the DA16600, refer to the following connections.



Top Surfaces of the Base Board and US159-DA16600EVZ Module

- (1) Short pins 1-2 of jumper block J16 on the base board (enables debugging).
- (2) Connect the US159-DA16600EVZ module to PMOD1 on the base board.
- (3) Connect J10 on the base board to a PC via a USB cable (USB serial connection).
- (4) Connect J14 on the base board to a PC via a USB cable (debugger connection).

3. Setting Up AWS

This chapter describes the procedure for setting up AWS to execute a FreeRTOS demo project.

A procedure for using the AWS command line interface (CLI) with the single sign-on (SSO) technique as an IAM Identity Center user (workforce user) is explained.

Note that accounts are managed differently for workforce users of the IAM Identity Center and for IAM users.

For details on the AWS CLI commands, refer to the following AWS document.

[AWS CLI command examples - AWS Command Line Interface](#)

3.1 Setting Up an Environment for Signing In to AWS

Follow the steps below to sign in to AWS, create a workforce user of the IAM Identity Center, and set up an environment for signing in to AWS.

3.1.1 Signing In to the AWS Management Console

Create a root user (management) account of AWS and sign in to the AWS management console.

- (1) Obtain a sign-in account.

An AWS account is required for signing in to AWS.

Create an AWS account with reference to the following document for AWS.

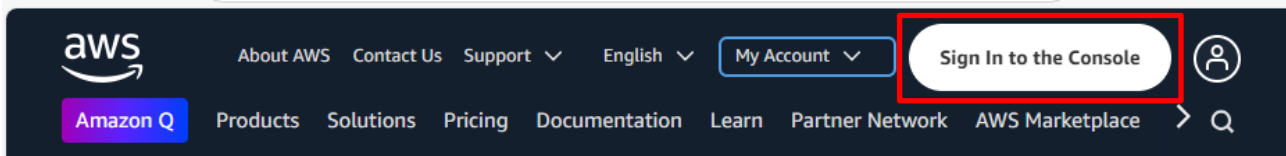
- [Set up AWS account - AWS IoT Core](#)

- (2) Sign in to the AWS console.

Sign in to the AWS management console with the AWS (management) account that you created.

- (a) Sign in to the AWS management console.

Access AWS (<https://aws.amazon.com/>) and click on [Sign In to the Console].

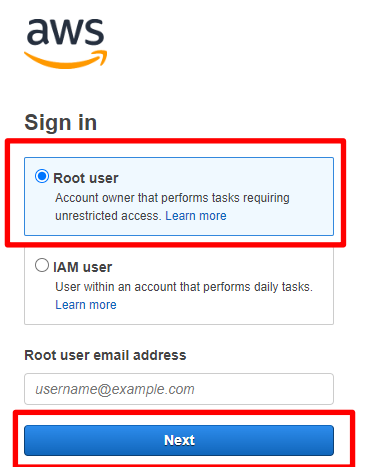


RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(b) Enter your email address.

Tick the [Root user] checkbox, enter the email address for the root user, and click on the [Next] button. You may be able to skip this step after having already signed in.

Sign in as the root user because IAM Identity Center services can only be created with a management account.



aws

Sign in

Root user
Account owner that performs tasks requiring unrestricted access. [Learn more](#)

IAM user
User within an account that performs daily tasks. [Learn more](#)

Root user email address

username@example.com

Next

(c) Enter your password.

Enter your password and then click on [Sign in].



aws

Root user sign in

Email: [Redacted]

Password [Forgot password?](#)

.....

Sign in

[Sign in to a different account](#)

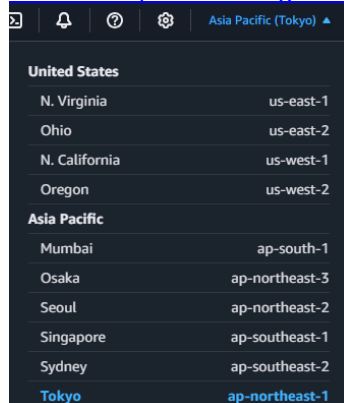
[Create a new AWS account](#)

3.1.2 Setting Your Region in AWS

After logging in to the AWS management console, select your region in the top right of the window.

Since the regions usable in the IAM Identity Center are pre-determined, select the appropriate region. For the usable regions, see the figure below.

[IAM Identity Center Region data storage and operations](#)



Region	Region Code
United States	
N. Virginia	us-east-1
Ohio	us-east-2
N. California	us-west-1
Oregon	us-west-2
Asia Pacific	
Mumbai	ap-south-1
Osaka	ap-northeast-3
Seoul	ap-northeast-2
Singapore	ap-southeast-1
Sydney	ap-southeast-2
Tokyo	ap-northeast-1

3.1.3 Creating a Workforce User of the IAM Identity Center

After obtaining an AWS account, sign in to the AWS management console with the management account. Then, create a workforce user with IAM Identity Center services and set up an environment for performing SSO to AWS.

By following the procedures below, start by creating instances of the IAM Identity Center, then create users, and finally assign the necessary permission sets to users or groups.

(1) Enable the IAM Identity Center.

Enable the IAM Identity Center with reference to the following Web page. After the IAM Identity Center is enabled, instances for use by the management account will be generated.

[Enable IAM Identity Center - AWS IAM Identity Center](#)

Management by the user is only possible through IAM Identity Center instances that were created with the management account.

(2) Create a workforce user.

Create a user of the IAM Identity Center with reference to the following Web page. The user created here will become the account of the workforce user.

[Add users to your Identity Center directory - AWS IAM Identity Center](#)

When a user has been created, AWS sends authentication mail to the mail address of the created user. Register a password and set multi-factor authentication (MFA), as directed by the contents of the mail.

(3) Create groups.

Create groups to which users belong as required with reference to the following Web page.

[Add groups to your Identity Center directory - AWS IAM Identity Center](#)

When a group has been created, users can be added to the created group.

(4) Create permission sets.

Create permission sets to be assigned to users or groups with reference to the following Web page.

[Create, manage, and delete permission sets - AWS IAM Identity Center](#)

Normally, the creation of AdministratorAccess would be satisfactory. However, when setting a permission set other than AdministratorAccess according to organization policies, the following permission sets need to be assigned to users for executing OTA updates.

- AWSIoTFullAccess
- AmazonFreeRTOSOTAUpdate
- AWSIoTDeviceTesterForFreeRTOSFullAccess

Note: The AdministratorAccess permission set includes the above permission sets.

(5) Assign the users and groups to access the AWS account.

Assign the created users and groups to the AWS management account and grant permission with reference to the following Web page.

[Assign user access to AWS accounts - AWS IAM Identity Center](#)

This will enable the workforce user to sign in to AWS.

Note: The AWS CLI can be used even in a sign-in by an IAM user with the use of the access key. In this case, the access key can be obtained by proceeding with reference to the following Web page.

[Manage access keys for IAM users - AWS Identity and Access Management](#)

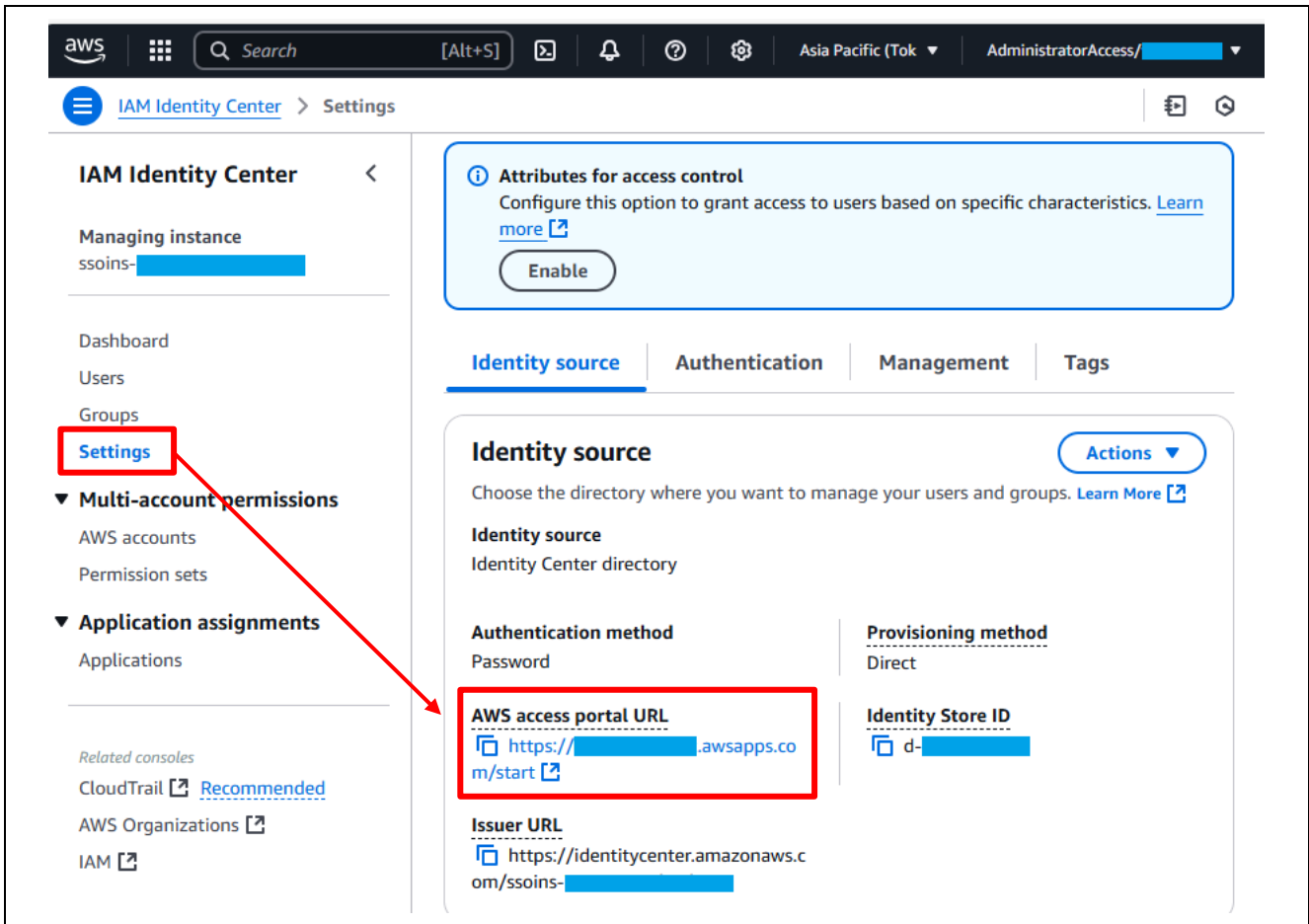
RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

3.1.4 Preparing for an AWS CLI Sign-In

Create a profile for signing in to the AWS CLI as a workforce user of the IAM Identity Center. Since a profile is reusable, it only has to be created once.

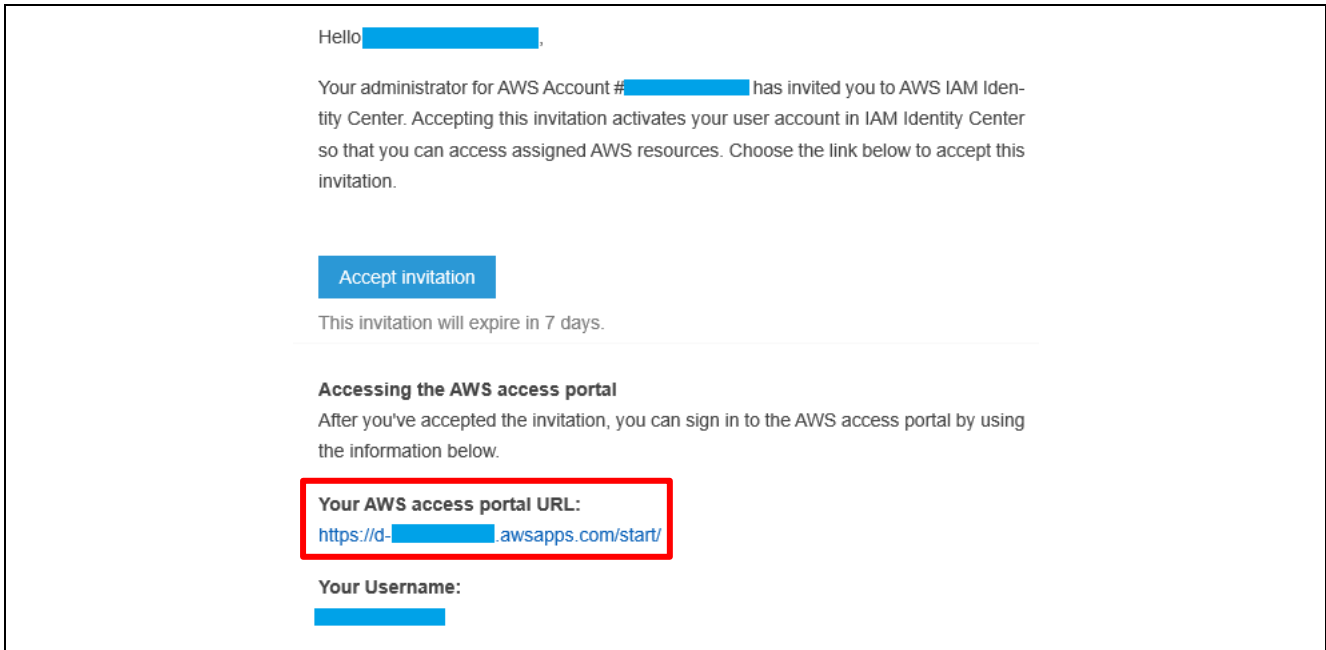
- (1) Obtain the URL of the AWS access portal.

Sign in to the AWS management portal as a root user, click on [Settings] in the IAM Identity Center menu, and make a note of [AWS access portal URL] in the [Identity source] field. When signing in as a root user is not possible, ask the account administrator about the AWS access portal URL.



RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

The AWS access portal URL is in the email you will have received to guide you in authentication as a workforce user for the first time.



(2) Create an SSO profile.

Open the command prompt window and execute the following command.

```
> aws configure sso
```

```
PowerShell 7 (x64)
```

```
PS C:\aws> aws configure sso
SSO session name (Recommended): Renesas
SSO start URL [None]: https://d-[redacted].awsapps.com/start
SSO region [None]: ap-northeast-1
SSO registration scopes [sso:account:access]:
```

Following execution of the command, you will be requested to enter the following four items. Enter the contents to be set for each item.

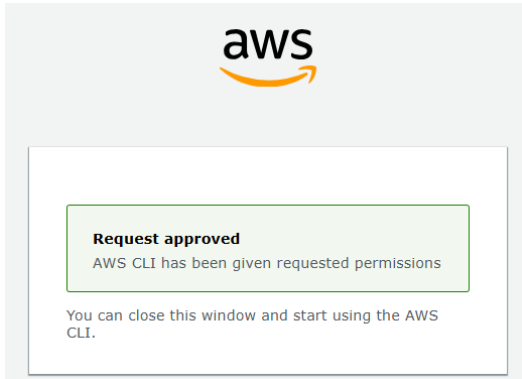
Item	Description	Contents to be Set
SSO session name	Session name	This is the session name for use in registration of the SSO profile. Enter a desired character string.
SSO start URL	URL for starting SSO	The URL for starting with an SSO is registered here. Enter the URL of the AWS access portal, which was obtained in step (1).
SSO region	Region name	The region in which the SSO is to be used is entered here. Enter the character string for the region in which the IAM Identity Center instance was created.
SSO registration scopes	Scopes of access permission	Simply press the Enter key without entering anything. This will set the initial value "sso:account:access".

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

After you have pressed the Enter key in response to the “SSO registration scopes” item, a Web browser will open and display the AWS sign-in window. Log in with the account for the workforce user.

If signing in is successful, the display in the window will be as shown below.

Note: If the information from having completed signed in is still in place, the window may display “Request approved” instead of making the transition to the sign-in window.



Once the account and permission sets have been authenticated, the following will be displayed in the command prompt window.

```
PowerShell 7 (x64)
Attempting to automatically open the SSO authorization page in your default browser.
If the browser does not open or you wish to use a different device to authorize this request, open the following URL:
https://oidc.ap-northeast-1.amazonaws.com/authorize?response_type=code&client_id=
redirect_uri=http
ge_method=S256&scopes=sso%3Aaccount%3Aaccess&code_challenge=
There are 2 AWS accounts available to you.
Using the account ID
The only role available to you is: AdministratorAccess
Using the role name "AdministratorAccess"
CLI default client Region [ap-northeast-1]:
CLI default output format [json]:
CLI profile name [AdministratorAccess-]: Renesas

To use this profile, specify the profile name using --profile, as shown:

aws s3 ls --profile Renesas
PS C:\aws>
```

You will be requested to enter the following three items. Enter the contents to be set for each item.

Item	Description	Contents to be Set
CLI default client Region	Region of the CLI	Enter the region where the CLI is to be used. Basically, do not change the region from the one that was set the first time.
CLI default output format	Format for output from the CLI	Enter "json".
CLI profile name	Profile name for the CLI	The character string of a desired profile name can be entered. This profile name is to be specified when entering CLI commands.

The creation of a profile is now complete.

When entering a CLI command, add “`--profile PROFILE_NAME`” to the end of the command to specify a profile.

The profile name set in this step should be specified as “PROFILE_NAME”.

The display examples used for explanations in this application note have the profile name setting “Renesas”.

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

- (3) Add a setting to the profile.

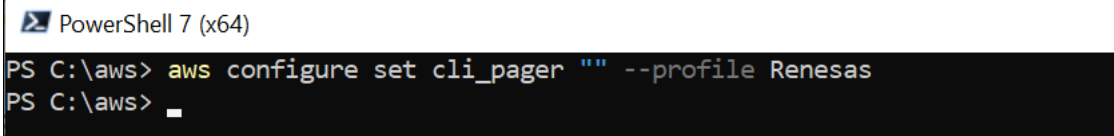
Make an additional setting to the profile for operating the CLI.

This setting will cancel the function of temporarily halting display when the result of a CLI command does not fit in one command prompt window.

Since the display of JSON data often involves showing a large amount of information, this setting was made in creating this application note.

Enter the following command.

```
> aws configure set cli_pager " " --profile <PROFILE_NAME>
```



```
PowerShell 7 (x64)
PS C:\aws> aws configure set cli_pager "" --profile Renesas
PS C:\aws> █
```

- (4) Confirm the login to AWS through the CLI.

Log in to AWS with the created profile. Enter the following command.

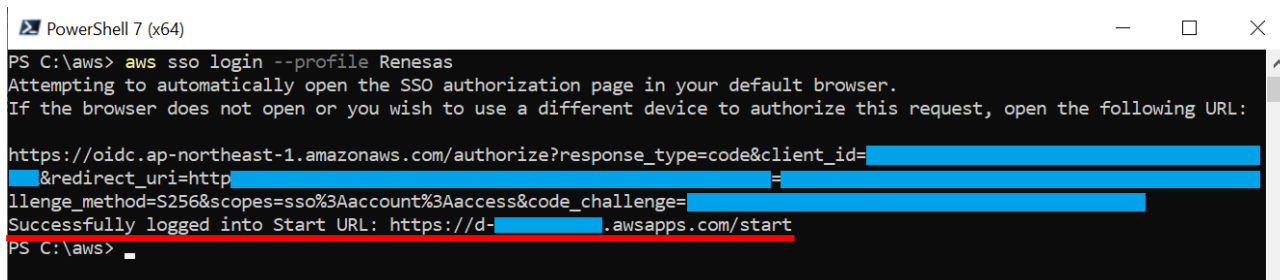
```
> aws sso login --profile <PROFILE_NAME>
```

After the command has been executed, a Web browser will open and display the AWS sign-in window. Sign in with the account for the workforce user.

Note: If the information from having completed signed in is still in place, the window may display “Request approved” instead of making the transition to the sign-in window.

After the login to AWS through the CLI has finished, the following will be displayed in the command prompt window.

The login is successful if “Successfully logged into Start URL: ~”, as in the underlined part in the figure below, is displayed.



```
PowerShell 7 (x64)
PS C:\aws> aws sso login --profile Renesas
Attempting to automatically open the SSO authorization page in your default browser.
If the browser does not open or you wish to use a different device to authorize this request, open the following URL:
https://oidc.ap-northeast-1.amazonaws.com/authorize?response_type=code&client_id=
&redirect_uri=http
llenge_method=S256&scopes=sso%3Aaccount%3Aaccess&code_challenge=
Successfully logged into Start URL: https://d-...awsapps.com/start
PS C:\aws> █
```

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(5) Check the operation of the CLI.

After an SSO login succeeds, enter the following command for obtaining a list of S3 buckets to confirm whether the CLI is operating normally.

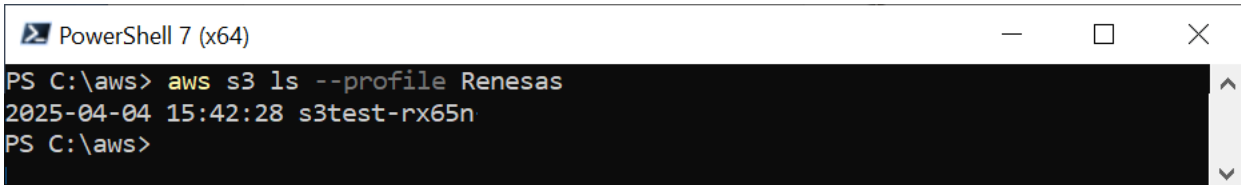
If no error occurs, the CLI command has been accepted normally.

If S3 buckets are available with your account, a list of buckets is displayed.

```
> aws s3 ls --profile <PROFILE NAME>
```

After the command has been executed, the display in the window will be as shown below.

The following window shows an example in which the existing bucket (s3test-rx65n) is available.

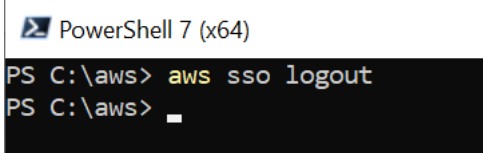


```
PowerShell 7 (x64)
PS C:\aws> aws s3 ls --profile Renesas
2025-04-04 15:42:28 s3test-rx65n
PS C:\aws>
```

(6) Log out from AWS.

Enter the following command at the command prompt.

```
> aws sso logout
```



```
PowerShell 7 (x64)
PS C:\aws> aws sso logout
PS C:\aws> █
```

3.1.5 Creating a Work Folder

Create a desired folder for performing steps with the AWS CLI. The configuration files and other files to be used in CLI commands are to be stored in this folder.

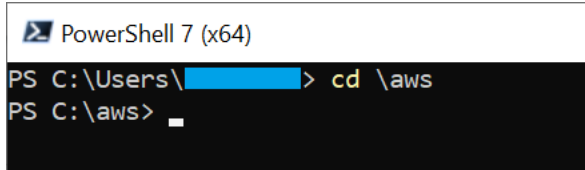
The work folder is created as "C:\aws" in the display examples used for explanations in this application note.

3.2 SSO Login to AWS through the CLI

Log in to AWS with the use of the AWS CLI.

- (1) Jump to the work folder.

Start the command prompt and change the current directory to the work folder that was created in 3.1.5.



```
PowerShell 7 (x64)
PS C:\Users\<user> > cd \aws
PS C:\aws>
```

- (2) Log in to AWS.

Enter the following command at the command prompt.

```
> aws sso login --profile <PROFILE_NAME>
```

Enter the profile name that was created in “3.1.4(2)” into `<PROFILE_NAME>`.

The specification of the profile name has to be entered at the end of all CLI commands explained from this point.

For details on the SSO login process, see 3.1.4(4).

3.3 Registering a Device with AWS

Create a device and assign the necessary permissions.

3.3.1 Setting Policies

Use AWS IoT Core services to set permissions (policies) for granting access to AWS resources in the device to be connected.

Set the following policies for the device to be connected according to this application note.

- `iot:Connect`: Connects to AWS IoT
- `iot:Publish`: Publishes (transmits) a topic
- `iot:Subscribe`: Subscribes to (receives) a topic
- `iot:Receive`: Receives messages from AWS IoT

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

- (1) Create a json file in which policies are set.

Create a configuration file of policies.

With a text editor, create a file with the name "policy.json" in the work folder that was created in 3.1.5 and enter the following code.

- policy.json

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iot:Connect",
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "iot:Publish",
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "iot:Subscribe",
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "iot:Receive",
      "Resource": "*"
    }
  ]
}
```

The text strings in blue are the names of access permissions to be assigned.

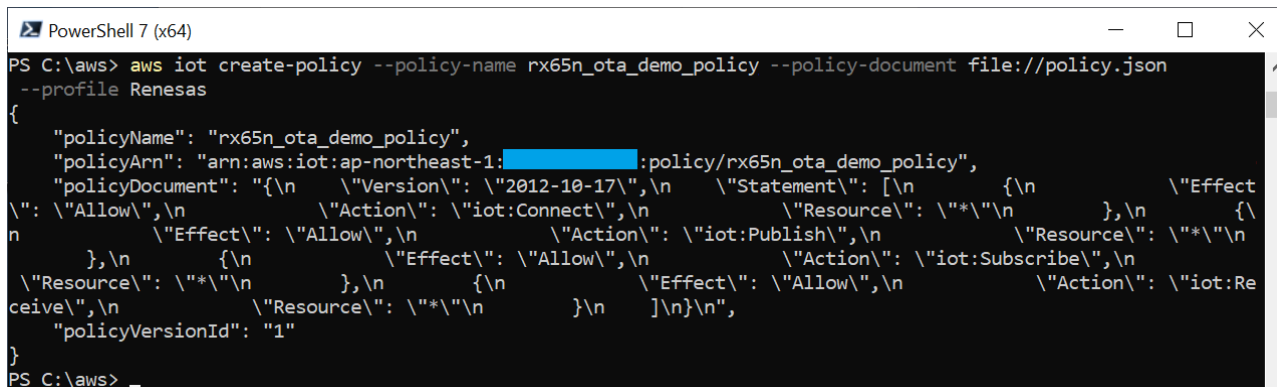
- (2) Create a policy.

Enter the following command and create a policy.

```
> aws iot create-policy --policy-name <POLICY_NAME> --policy-document
file://policy.json --profile <PROFILE NAME>
```

- A desired policy name can be entered into <POLICY_NAME> (example: rx65n_ota_demo_policy).

After the command has been executed, the display in the window will be as shown below.



```
PowerShell 7 (x64)
PS C:\aws> aws iot create-policy --policy-name rx65n_ota_demo_policy --policy-document file://policy.json
--profile Renesas
{
  "policyName": "rx65n_ota_demo_policy",
  "policyArn": "arn:aws:iot:ap-northeast-1:██████████:policy/rx65n_ota_demo_policy",
  "policyDocument": "{\n  \"Version\": \"2012-10-17\", \n  \"Statement\": [\n    {\n      \"Effect\": \"Allow\", \n      \"Action\": \"iot:Connect\", \n      \"Resource\": \"*\" \n    }, \n    {\n      \"Effect\": \"Allow\", \n      \"Action\": \"iot:Publish\", \n      \"Resource\": \"*\" \n    }, \n    {\n      \"Effect\": \"Allow\", \n      \"Action\": \"iot:Subscribe\", \n      \"Resource\": \"*\" \n    }, \n    {\n      \"Effect\": \"Allow\", \n      \"Action\": \"iot:Receive\", \n      \"Resource\": \"*\" \n    } \n  ] \n}",
  "policyVersionId": "1"
}
```

Make a note of the policy name that was created. You will need it in a later process.

3.3.2 Creating an Amazon S3 Bucket

Amazon S3 is an online storage Web service used to store the firmware for use in updating.

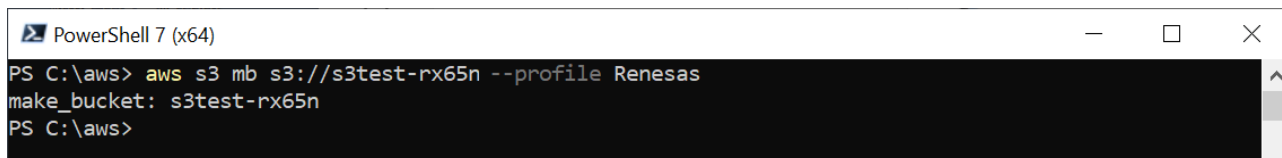
(1) Create an S3 bucket.

Enter the following command.

```
> aws s3 mb s3://<BUCKET_NAME> --profile <PROFILE_NAME>
```

- A desired bucket name can be entered into **<BUCKET_NAME>** (example: s3test-rx65n).

After the command has been executed, the display in the window will be as shown below.



```
PowerShell 7 (x64)
PS C:\aws> aws s3 mb s3://s3test-rx65n --profile Renesas
make_bucket: s3test-rx65n
PS C:\aws>
```

Make a note of the bucket name that was entered. You will need it in a later process.

Note 1. The bucket name must be globally unique. The following error message appears if the bucket name is already in use. In this case, use another name.

```
make_bucket failed: s3://s3test-rx65n An error occurred
(BucketAlreadyExists) when calling the CreateBucket operation: The
requested bucket name is not available. The bucket namespace is shared
by all users of the system. Please select a different name and try
again.
```

Note 2. Only lowercase letters, numerals, periods (.), and hyphens (-) can be used in a bucket name.

(2) Enable version control for the S3 bucket.

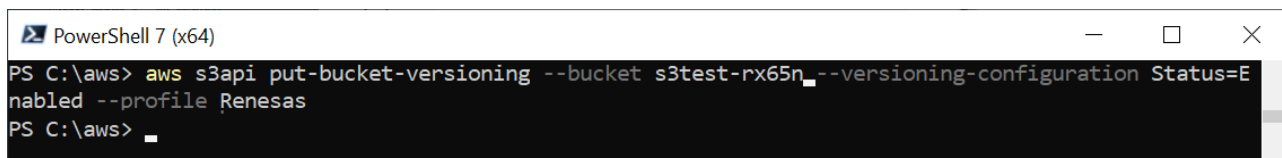
Enable this function to manage the versions of the files to be stored in the created bucket.

Enter the following command.

```
> aws s3api put-bucket-versioning --bucket <BUCKET_NAME> --versioning-
configuration Status=Enabled --profile <PROFILE_NAME>
```

- Enter the bucket name that was created in step (1) into **<BUCKET_NAME>**.

After the command has been executed, the display in the window will be as shown below (no part of the execution result is displayed).



```
PowerShell 7 (x64)
PS C:\aws> aws s3api put-bucket-versioning --bucket s3test-rx65n --versioning-configuration Status=E
nabled --profile Renesas
PS C:\aws>
```

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

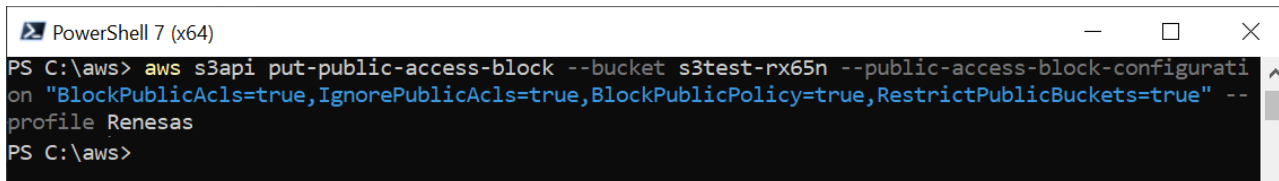
(3) Make a setting to block public access to the S3 bucket.

This setting is for restricting public access to the created bucket.
Enter the following command.

```
> aws s3api put-public-access-block --bucket <BUCKET_NAME> --public-access-block-configuration "BlockPublicAcls=true,IgnorePublicAcls=true,BlockPublicPolicy=true,RestrictPublicBuckets=true" --profile <PROFILE_NAME>
```

- Enter the bucket name that was created in step (1) into <BUCKET_NAME>.

After the command has been executed, the display in the window will be as shown below (no part of the execution result is displayed).



```
PowerShell 7 (x64)
PS C:\aws> aws s3api put-public-access-block --bucket s3test-rx65n --public-access-block-configuration "BlockPublicAcls=true,IgnorePublicAcls=true,BlockPublicPolicy=true,RestrictPublicBuckets=true" --profile Renesas
PS C:\aws>
```

3.3.3 Allocating Permission for OTA Execution to an IAM User

Use AWS IAM services to create a role and assign it the appropriate access permissions for creating OTA update jobs.

(1) Create a json file in which the permission for OTA execution is set.

Create a configuration file of the permission for OTA execution.
With a text editor, create a file with the name "Test-Role-Trust-Policy.json" in the work folder that was created in 3.1.5 and enter the following code.

- Test-Role-Trust-Policy.json

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "iot.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

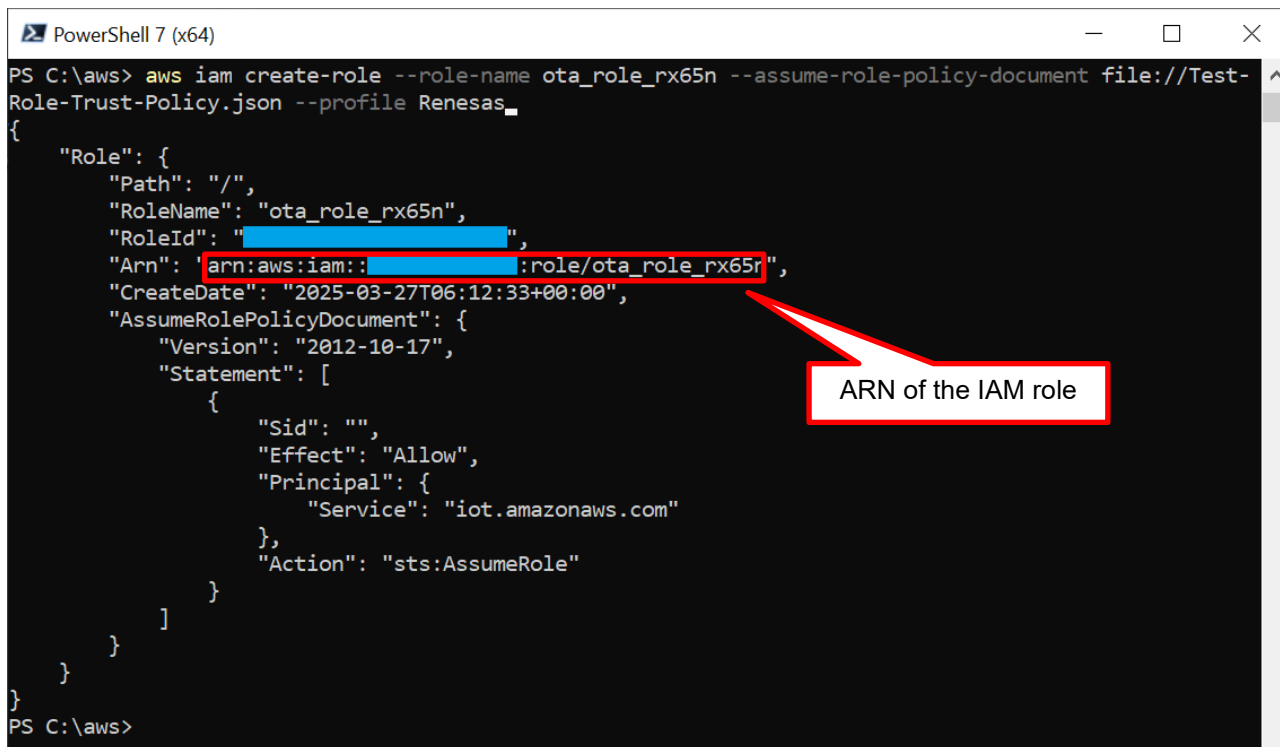
(2) Create an IAM role.

Create an IAM role for use in OTA updates.
Enter the following command.

```
> aws iam create-role --role-name <ROLE_NAME> --assume-role-policy-document  
file://Test-Role-Trust-Policy.json --profile <PROFILE_NAME>
```

- A desired role name can be entered into **<ROLE_NAME>** (example: ota_role_rx65n).

After the command has been executed, the display in the window will be as shown below.



```
PowerShell 7 (x64)
PS C:\aws> aws iam create-role --role-name ota_role_rx65n --assume-role-policy-document file://Test-Role-Trust-Policy.json --profile Renesas_
{
  "Role": {
    "Path": "/",
    "RoleName": "ota_role_rx65n",
    "RoleId": "A...",
    "Arn": "arn:aws:iam:::role/ota_role_rx65n",
    "CreateDate": "2025-03-27T06:12:33+00:00",
    "AssumeRolePolicyDocument": {
      "Version": "2012-10-17",
      "Statement": [
        {
          "Sid": "",
          "Effect": "Allow",
          "Principal": {
            "Service": "iot.amazonaws.com"
          },
          "Action": "sts:AssumeRole"
        }
      ]
    }
  }
}
```

Make a note of the role name entered here and the Amazon Resource Name (ARN) of the IAM role (Arn: text inside the red frame in the above figure) which is displayed. You will need them in a later process.

(3) Attach managed policies to the IAM role.

Attach the AWS IoT and FreeRTOS OTA managed policies to the IAM role.
Attach the following four managed policies to the IAM role.

- AWSIoTThingsRegistration
- AWSIoTRuleActions
- AWSIoTLogging
- AmazonFreeRTOSOTAUpdate

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

Enter the following four commands in order.

```
> aws iam attach-role-policy --role-name <ROLE_NAME> --policy-arn
arn:aws:iam::aws:policy/service-role/AWSIoTThingsRegistration --profile
<PROFILE_NAME>
```

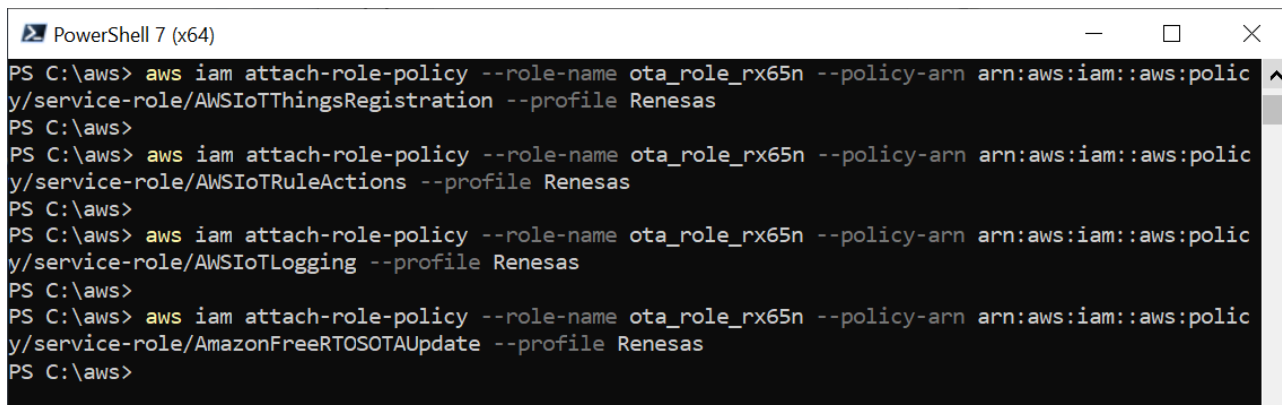
```
> aws iam attach-role-policy --role-name <ROLE_NAME> --policy-arn
arn:aws:iam::aws:policy/service-role/AWSIoTRuleActions --profile
<PROFILE_NAME>
```

```
> aws iam attach-role-policy --role-name <ROLE_NAME> --policy-arn
arn:aws:iam::aws:policy/service-role/AWSIoTLogging --profile <PROFILE_NAME>
```

```
> aws iam attach-role-policy --role-name <ROLE_NAME> --policy-arn
arn:aws:iam::aws:policy/service-role/AmazonFreeRTOSOTAUpdate --profile
<PROFILE_NAME>
```

- Enter the IAM role name that was created in step (2) into **<ROLE_NAME>**.

After the commands have been executed, the display in the window will be as shown below (no part of any execution result is displayed).



```
PowerShell 7 (x64)
PS C:\aws> aws iam attach-role-policy --role-name ota_role_rx65n --policy-arn arn:aws:iam::aws:policy/service-role/AWSIoTThingsRegistration --profile Renesas
PS C:\aws>
PS C:\aws> aws iam attach-role-policy --role-name ota_role_rx65n --policy-arn arn:aws:iam::aws:policy/service-role/AWSIoTRuleActions --profile Renesas
PS C:\aws>
PS C:\aws> aws iam attach-role-policy --role-name ota_role_rx65n --policy-arn arn:aws:iam::aws:policy/service-role/AWSIoTLogging --profile Renesas
PS C:\aws>
PS C:\aws> aws iam attach-role-policy --role-name ota_role_rx65n --policy-arn arn:aws:iam::aws:policy/service-role/AmazonFreeRTOSOTAUpdate --profile Renesas
PS C:\aws>
```

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(4) Make a permission setting for passing the IAM role to AWS services.

Use an inline policy to make a permission setting for passing the IAM role to AWS services.

(a) Create a json file in which the inline policy is set.

Create a file containing an inline policy to which an access permission is set.

With a text editor, create a file with the name "inline-policy1.json" in the work folder that was created in 3.1.5 and enter the following code.

- inline-policy1.json

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iam:GetRole",
        "iam:PassRole"
      ],
      "Resource": "*"
    }
  ]
}
```

(b) Attach the inline policy to the IAM role.

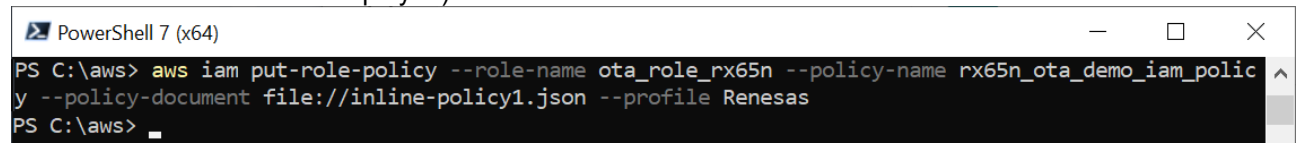
Attach the inline policy to the IAM role.

Enter the following command.

```
> aws iam put-role-policy --role-name <ROLE_NAME> --policy-name <POLICY_NAME>
--policy-document file://inline-policy1.json --profile <PROFILE_NAME>
```

- Enter the IAM role name that was created in step (2) into <ROLE_NAME>.
- A desired policy name can be entered into <POLICY_NAME> (example: rx65n_ota_demo_iam_policy).

After the command has been executed, the display in the window will be as shown below (no part of the execution result is displayed).



```
PowerShell 7 (x64)
PS C:\aws> aws iam put-role-policy --role-name ota_role_rx65n --policy-name rx65n_ota_demo_iam_polic
y --policy-document file://inline-policy1.json --profile Renesas
PS C:\aws> _
```

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(5) Assign permission for granting access to Amazon S3 to the IAM role.

Use an inline policy to add permission for access by the IAM role to Amazon S3, where the update firmware is to be stored.

(a) Create a json file in which an inline policy is set.

Create a file containing an inline policy to which an access permission is set.

With a text editor, create a file with the name "inline-policy2.json" in the work folder that was created in 3.1.5 and enter the following code.

- inline-policy2.json

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetObjectVersion",
        "s3:GetObject",
        "s3:PutObject"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```

(b) Attach the inline policy (Amazon S3) to the IAM role.

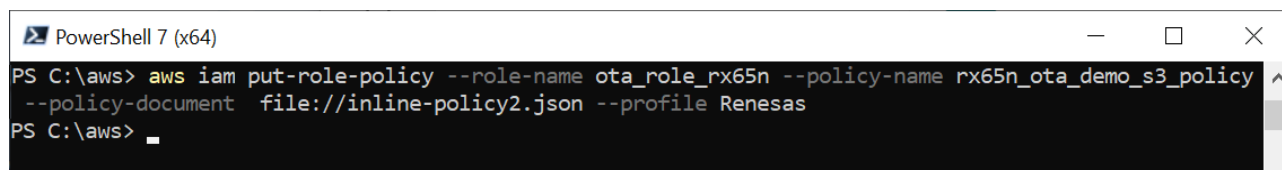
Attach the inline policy to the IAM role.

Enter the following command.

```
> aws iam put-role-policy --role-name <ROLE_NAME> --policy-name <POLICY_NAME>
--policy-document file://inline-policy2.json --profile <PROFILE_NAME>
```

- Enter the IAM role name that was created in step (2) into <ROLE_NAME>.
- A desired policy name can be entered into <POLICY_NAME> (example: rx65n_ota_demo_s3_policy).

After the command has been executed, the display in the window will be as shown below (no part of the execution result is displayed).



```
PowerShell 7 (x64)
PS C:\aws> aws iam put-role-policy --role-name ota_role_rx65n --policy-name rx65n_ota_demo_s3_policy
--policy-document file://inline-policy2.json --profile Renesas
PS C:\aws> █
```

3.3.4 Registering a Device as a Thing in AWS IoT

Use AWS IoT Core services to create a device (“thing”) and certificate for the AWS IoT Core to be connected.

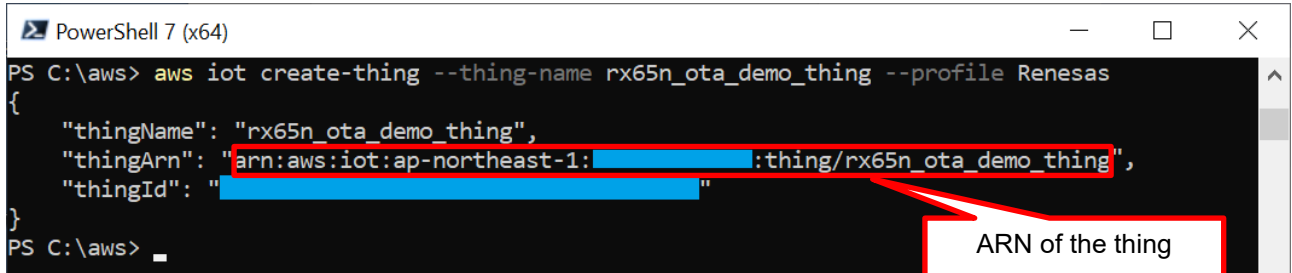
- (1) Create a device (thing).

Enter the following command and create a thing.

```
> aws iot create-thing --thing-name <THING_NAME> --profile <PROFILE_NAME>
```

- A desired thing name can be entered into <THING_NAME> (example: rx65n_ota_demo_thing).

After the command has been executed, the display in the window will be as shown below.



```
PowerShell 7 (x64)
PS C:\aws> aws iot create-thing --thing-name rx65n_ota_demo_thing --profile Renesas
{
  "thingName": "rx65n_ota_demo_thing",
  "thingArn": "arn:aws:iot:ap-northeast-1:xxxxxx:thing/rx65n_ota_demo_thing",
  "thingId": "xxxxxx"
}
PS C:\aws>
```

Make a note of the thing name entered here and the ARN of the thing (thingArn: text inside the red frame in the above figure) which is displayed. You will need them in a later process.

- (2) Create a device certificate and key.

Create a device certificate and make it active, then download the device certificate, public key, and private key files.

Enter the following command.

```
> aws iot create-keys-and-certificate --set-as-active --certificate-pem-
outfile "certificate.pem.crt" --public-key-outfile "public.pem.key" --private-
key-outfile "private.pem.key" --profile <PROFILE_NAME>
```


4. Setting Up a Device

Create a certificate for use in code signing verification of the firmware during the execution of an OTA update.

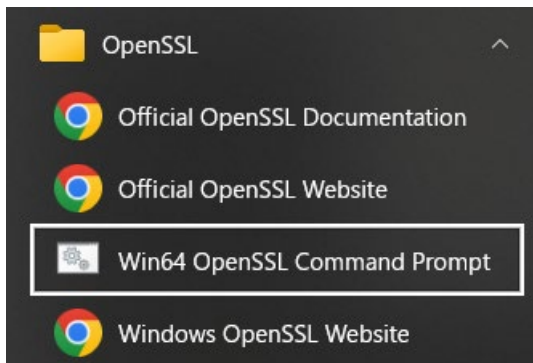
Create the certificate through the following steps.

Note: The certificate that is created here is different from that created in 3.3.4(2).

4.1 Generating a Key Pair and Certificate

(1) Start the Win64 OpenSSL Command Prompt.

From the Start menu, open the Win64 OpenSSL Command Prompt.



(2) Create an ECDSA CA private key.

Issue an OpenSSL command to create an elliptic curve digital signature algorithm (ECDSA) certificate authority (CA) private key. Execute the following command.

```
> openssl ecparam -genkey -name secp256r1 -out ca.key
```

After the command has been executed, the display in the window will be as shown below.

```
C:\openssl>openssl ecparam -genkey -name secp256r1 -out ca.key  
using curve name prime256v1 instead of secp256r1
```

(3) Create a CA certificate.

Create a CA certificate from the CA private key created in the previous step. Execute the following command. Desired character strings can be entered into the fields from "Country Name" onward.

```
> openssl req -x509 -sha256 -new -nodes -key ca.key -days 3650 -out ca.crt
```

After the command has been executed, the display in the window will be as shown below.

```
C:\openssl>openssl req -x509 -sha256 -new -nodes -key ca.key -days 3650 -out ca.crt  
You are about to be asked to enter information that will be incorporated  
into your certificate request.  
What you are about to enter is what is called a Distinguished Name or a DN.  
There are quite a few fields but you can leave some blank  
For some fields there will be a default value,  
If you enter '.', the field will be left blank.  
-----  
Country Name (2 letter code) [AU]:JP  
State or Province Name (full name) [Some-State]:Tokyo  
Locality Name (eg, city) []:Kodaira  
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Renesas Electronics  
Organizational Unit Name (eg, section) []:Software Development Division  
Common Name (e.g. server FQDN or YOUR name) []:Renesas Taro  
Email Address []:Taro.Renasas@sample.com
```

Enter any character string.

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

- (4) Create an ECDSA key pair.

Create an ECDSA key pair.
Execute the following command.

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

After the command has been executed, the display in the window will be as shown below.

```
C:\openssl>openssl ecparam -genkey -name secp256r1 -out secp256r1.keypair
using curve name prime256v1 instead of secp256r1
```

- (5) Create a certificate request from the ECDSA key pair.

Create a certificate request from the ECDSA key pair created in the previous step.
Execute the following command.

Desired character strings can be entered into the fields from “Country Name” onward. For the last two lines, press the Enter key with the fields left blank.

```
> openssl req -new -sha256 -key secp256r1.keypair > secp256r1.csr
```

After the command has been executed, the display in the window will be as shown below.

```
C:\openssl>openssl req -new -sha256 -key secp256r1.keypair > secp256r1.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:JP
State or Province Name (full name) [Some-State]:Tokyo
Locality Name (eg, city) []:Kodaira
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Renesas Electronics
Organizational Unit Name (eg, section) []:Software Development Division
Common Name (e.g. server FQDN or YOUR name) []:Renesas Taro
Email Address []:Taro.Renesas@sample.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

Enter any character string.

Press the Enter key with the fields left blank.

- (6) Create a certificate.

Create a certificate from the certificate request, CA certificate, and CA private key created so far.
Execute the following command.

```
> openssl x509 -req -sha256 -days 3650 -in secp256r1.csr -CA ca.crt -CAkey
ca.key -CAcreateserial -out secp256r1.crt
```

After the command has been executed, the display in the window will be as shown below.

```
C:\openssl>openssl x509 -req -sha256 -days 3650 -in secp256r1.csr -CA ca.crt -CAkey ca.key -CAcreateserial -out secp256r
1.crt
Certificate request self-signature ok
subject=C=JP, ST=Tokyo, L=Kodaira, O=Renesas Electronics, OU=Software Development Division, CN=Renesas Taro, emailAddres
s=Taro.Renesas@sample.com
```

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(7) Extract the private key from the ECDSA key pair.

Extract the private key from the ECDSA key pair.
Execute the following command.

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

After the command has been executed, the display in the window will be as shown below.

```
C:\openssl>openssl ec -in secp256r1.keypair -outform PEM -out secp256r1.privatekey
```

(8) Extract the public key from the ECDSA key pair.

Extract the public key from the ECDSA key pair.
Execute the following command.

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

After the command has been executed, the display in the window will be as shown below.

```
C:\openssl>openssl ec -in secp256r1.keypair -outform PEM -pubout -out secp256r1.publickey
```

(9) Confirm the generated files.

Following the procedure in this section will lead to creation of the four files listed below.

- ECDSA public key: secp256r1.publickey
- ECDSA private key: secp256r1.privatekey
- ECDSA certificate (key pair certificate): secp256r1.crt
- ECDSA certificate chain: ca.crt

The above files created by this point are used for making settings at the times of creating projects and creating OTA jobs in AWS.

4.2 Creating the Initial Version of the Firmware

This section describes how to create the initial version of the firmware.

4.2.1 Creating the Sample Projects

Follow the procedure below to create the OTA sample project and Secure Bootloader sample project needed to implement OTA using Amazon Web Services, as described in this section.

If you wish to run the demo project using an imported project instead, please refer to the instructions in [Getting Started Guide](#).

This application note explains the procedure to generate an Ethernet project for the CC-RX compiler.

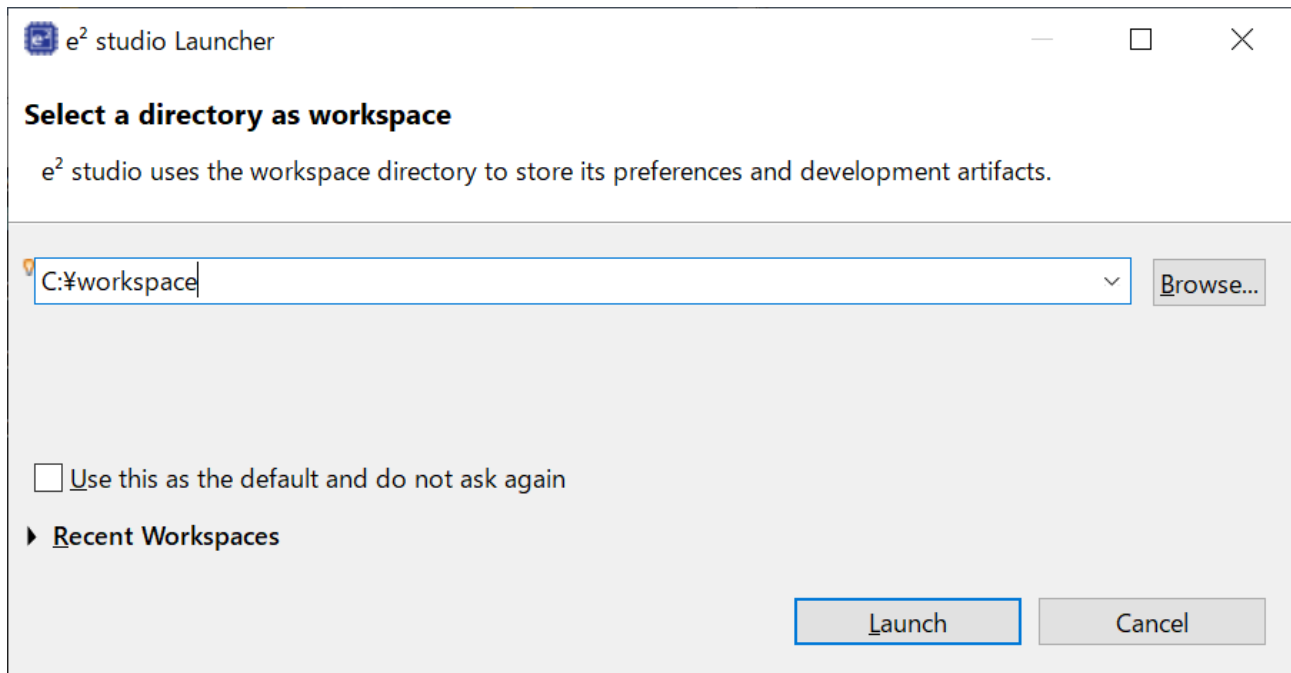
(1) Create a workspace in e² studio.

Launch e² studio and create a new workspace.

Keep the names of the workspace and the project files as short as possible. If the lowest-level directory pathname exceeds 256 bytes, an error will occur when you build the project.

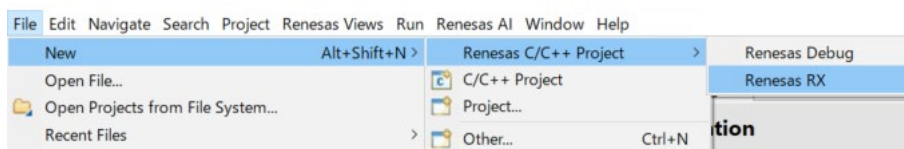
Errors may also occur if the file path contains Japanese characters, so make sure the name you enter contains only alphanumeric characters.

Example: Creating a workspace in location **C:\workspace**



After launching e² studio, from the **File** menu select **New > Renesas C/C++ Project > Renesas RX**.

This will open the **New C/C++ Project** dialog box.



RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

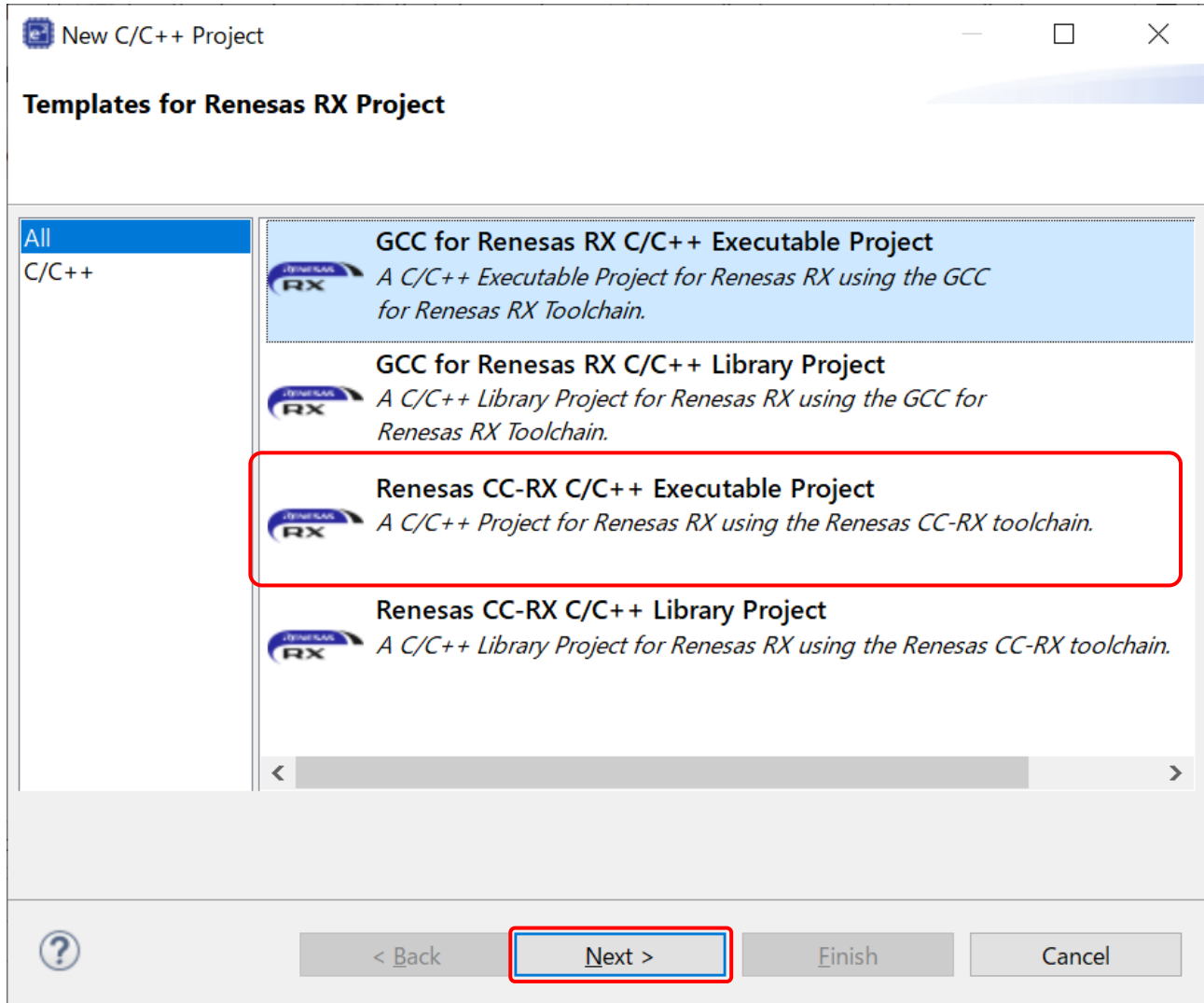
(2) Selecting the project to be generated

In the **New C/C++ Project** dialog box, select the type of project to be created.

Under the “**All**” category, select **Renesas CC-RX C/C++ Executable Project** and click the **Next** button.

To create a GCC project, select **GCC for Renesas RX C/C++ Executable Project** as the project type instead.

Selecting a project type opens the **New Renesas CC-RX Executable Project** (or **GCC for Renesas RX**) dialog box.



RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(3) Create the OTA sample project.

In the **New Renesas CC-RX Executable Project** dialog box, specify your project name.

Enter **aws_ether_ck_rx65n_v2** as the project name and click the **Next** button. This opens the **Select toolchain, device & debug settings** dialog box.

New Renesas CC-RX Executable Project

New Renesas CC-RX Executable Project

Project name:

Use default location

Location:

Create Directory for Project

Choose file system:

Working sets

Add project to working sets

Working sets:

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

In the **Select toolchain, device & debug settings** dialog box, configure the settings to use for the project.

For **Toolchain Settings**, the **Toolchain** is pre-selected based on the type of project. To change the toolchain version, select the version of your choice from the drop-down list next to **Toolchain Version**.

For **RTOS**, select **Free RTOS (with IoT libraries)**, and for **RTOS Version** select **202406.01-LTS-rx-1.1.1**.

For **Device Settings > Target Board**, select **CK-RX65N-V2**. The setting for **Target Device** will be selected automatically.

For **Bank Mode**, select **Dual Bank**. OTA update application requires the device code flash memory to be configured to dual bank mode.

After all the settings have been configured, click the **Next** button.

The screenshot shows the 'New Renesas CC-RX Executable Project' dialog box with the following settings:

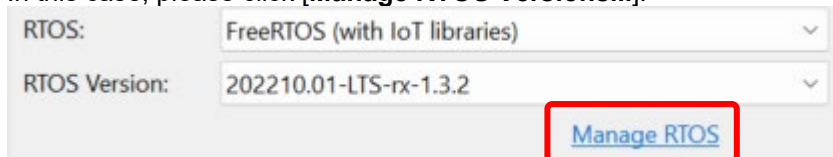
- Toolchain Settings:**
 - Language: C C++
 - Toolchain: Renesas CC-RX
 - Toolchain Version: v3.07.00
 - Manage Toolchains...
- RTOS:**
 - RTOS: FreeRTOS (with IoT libraries)
 - RTOS Version: 202406.01-LTS-rx-1.1.1
 - Manage RTOS
- Device Settings:**
 - Target Board: CK-RX65N-V2
 - Download additional
 - Target Device: R5F565NEHxFB
 - Unlock
 - Endian: Little
 - Bank Mode: Dual Bank
- Configurations:**
 - Create Hardware Debug Configuration
 - E2 Lite (RX)
 - Create Debug Configuration
 - RX Simulator
 - Create Release Configuration

At the bottom, the **Next >** button is highlighted with a red box.

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

If you are running the e2 studio for the first time or if the desired version does not appear in the [Version] list, you will have to download FreeRTOS from GitHub.

In this case, please click **[Manage RTOS Versions...]**.

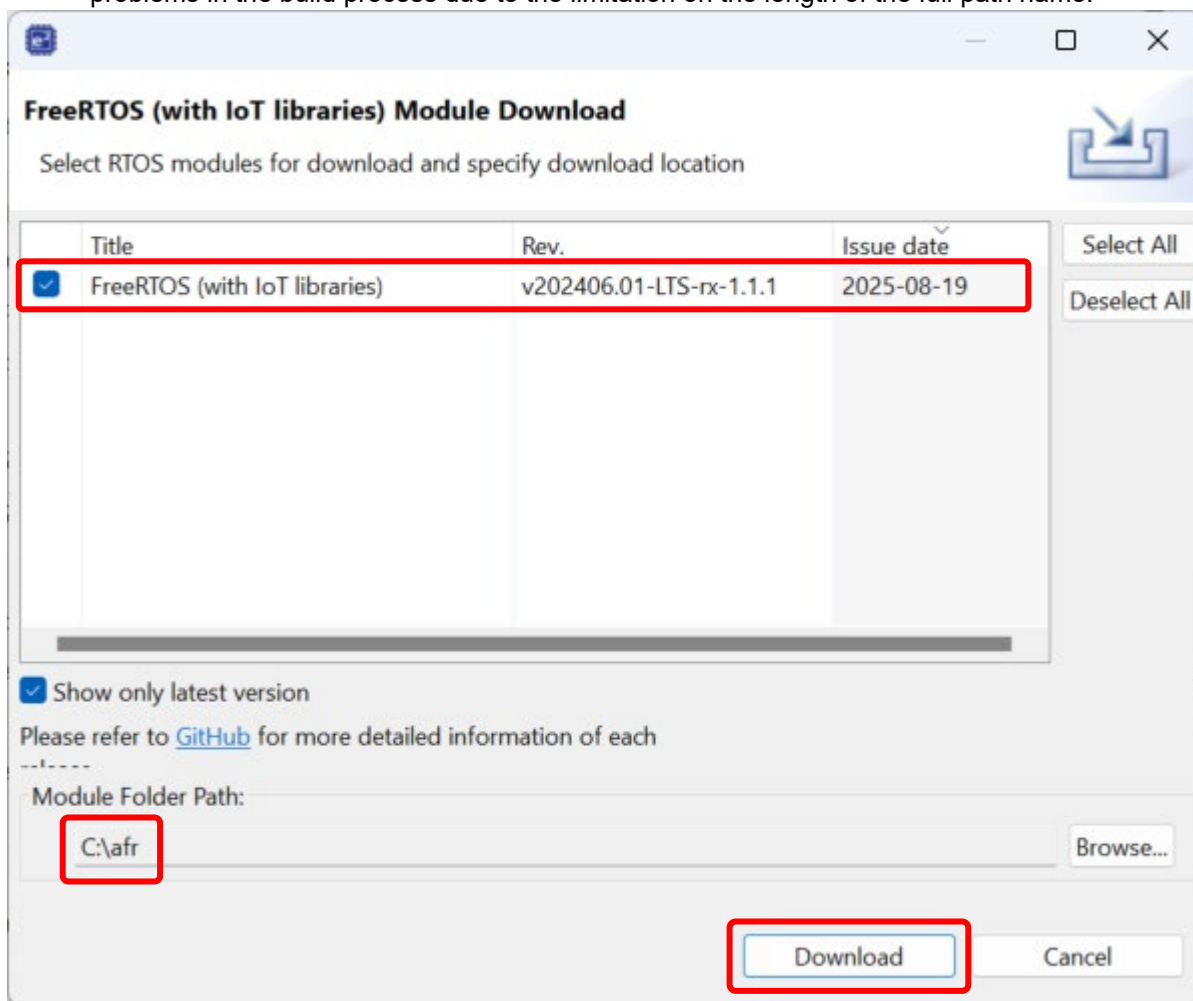


RTOS: FreeRTOS (with IoT libraries) ▾
RTOS Version: 202210.01-LTS-rx-1.3.2 ▾
Manage RTOS

The **[FreeRTOS (with IoT libraries) Module Download]** dialog box will appear.

Select the desired version and click on the [Download] button to download it.

Note: Specify a path name no longer than ten characters, such as "c:\afr", in [Module Folder Path] to avoid problems in the build process due to the limitation on the length of the full path name.



FreeRTOS (with IoT libraries) Module Download
Select RTOS modules for download and specify download location

Title	Rev.	Issue date
<input checked="" type="checkbox"/> FreeRTOS (with IoT libraries)	v202406.01-LTS-rx-1.1.1	2025-08-19

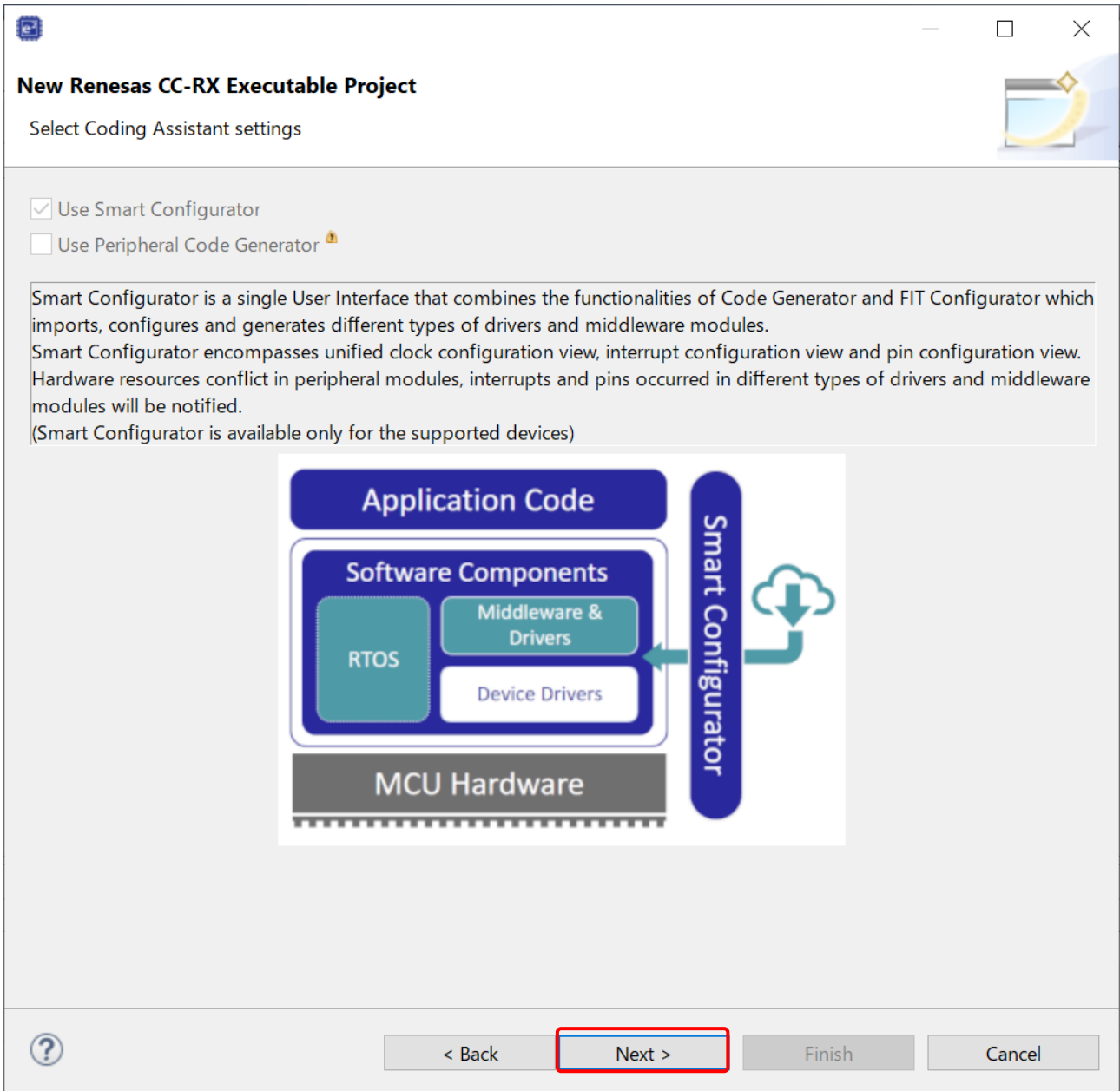
Show only latest version
Please refer to [GitHub](#) for more detailed information of each

Module Folder Path:

After the download is complete, the screen will return to the project selection view.

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

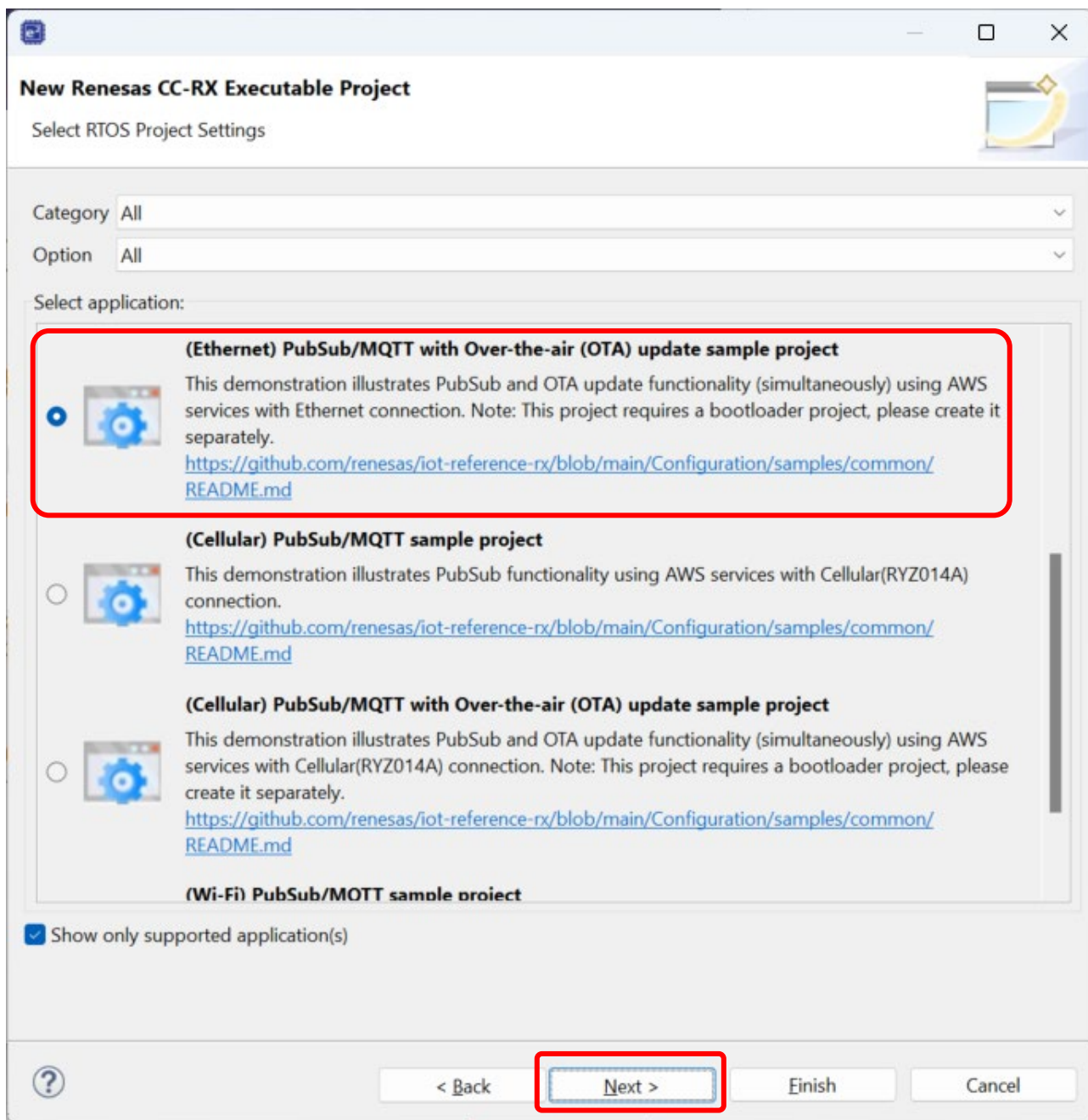
When the **Select Coding Assistant settings** dialog box appears, click the **Next** button without changing any settings.



RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

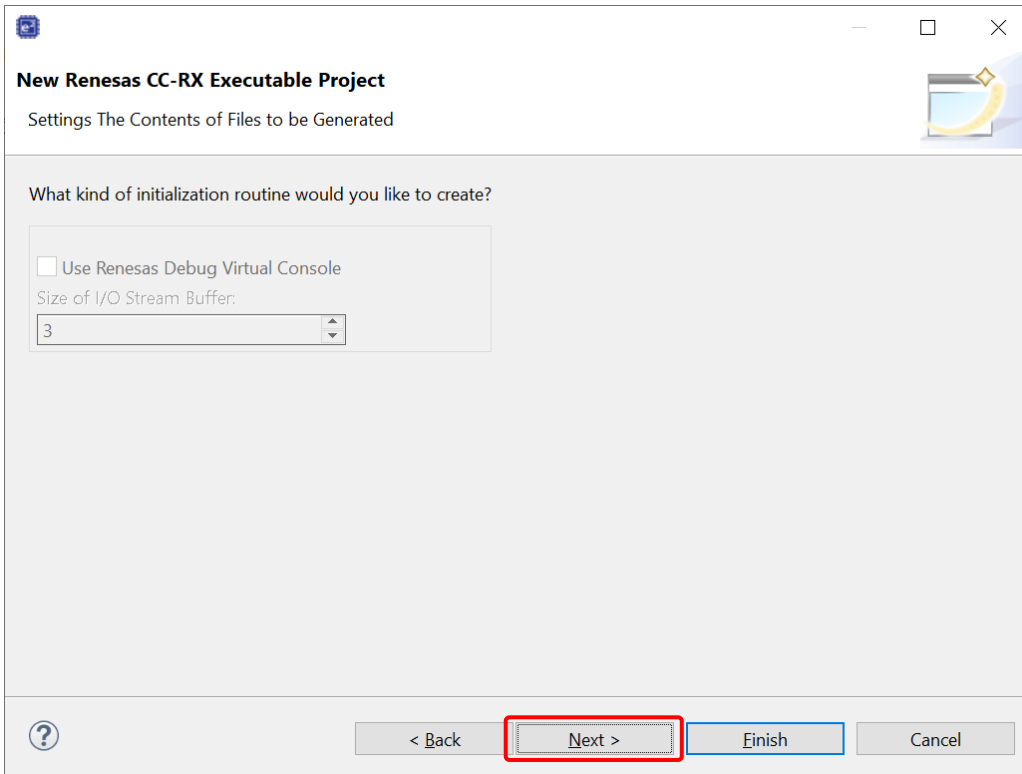
In the **Select RTOS Project Settings** dialog box, a list of sample projects is displayed.

Use the scroll bar to scroll down the list, select **(Ethernet) PubSub/MQTT with Over-the-air (OTA) update sample project**, and click the **Next** button.

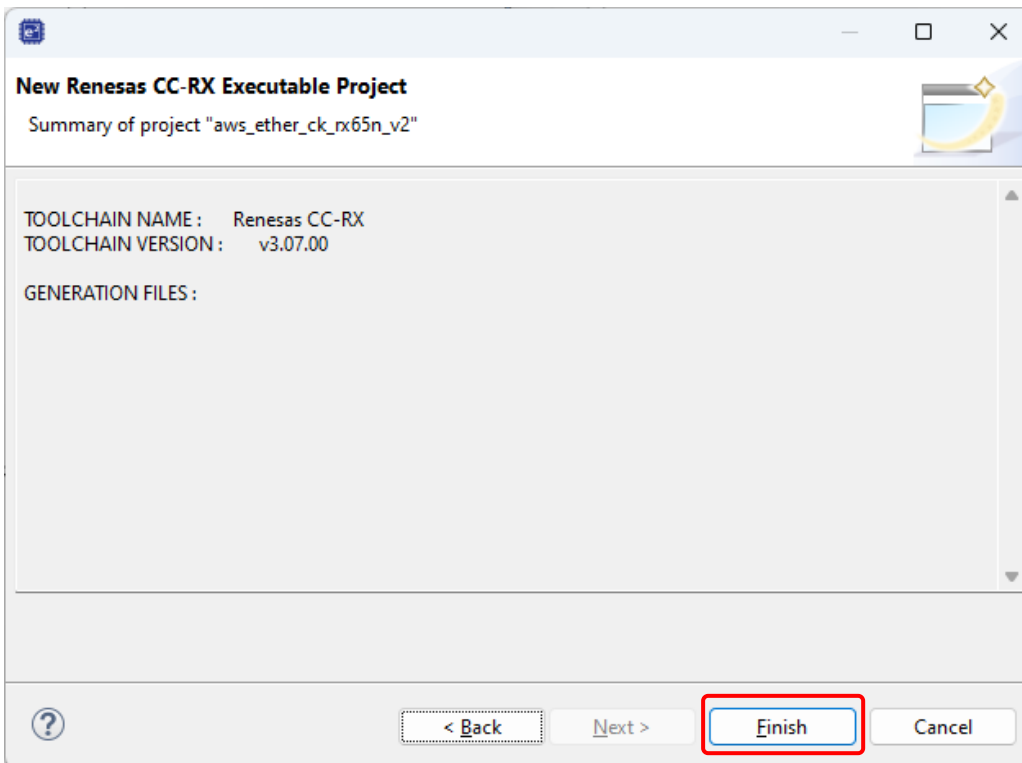


RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

When the **Settings The Contents of Files to be Generated** dialog box appears, click the **Next** button without changing any settings.

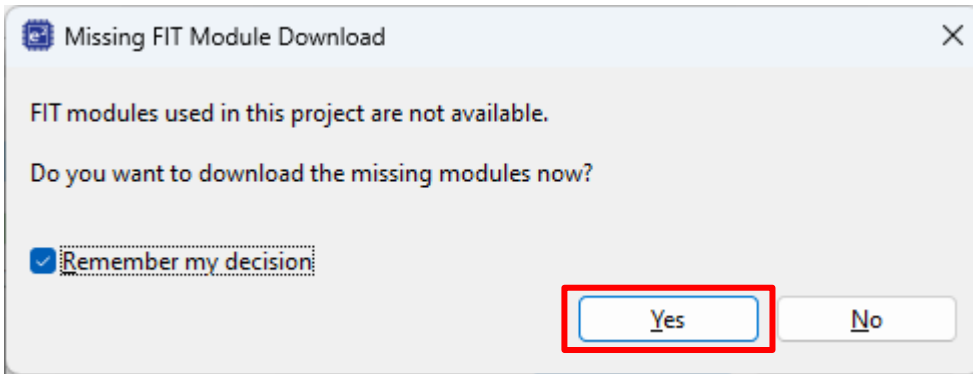


A dialog box appears indicating that the project has been created. If everything appears to be in order, click the **Finish** button.

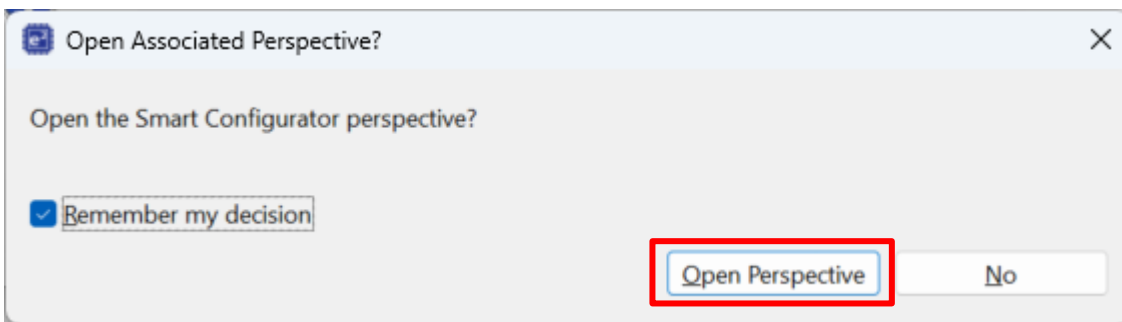


RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

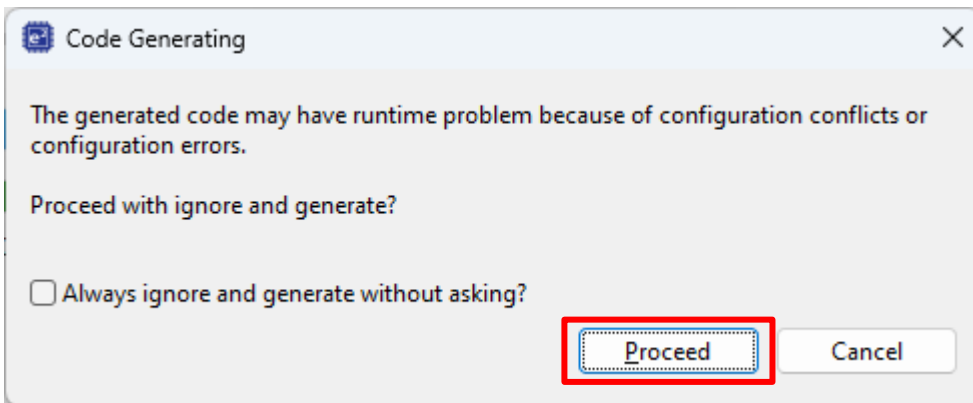
If the **Missing FIT Module download** dialog box appears on the screen, click **Yes** button.
If the screen for the **Download missing components** appears next, click **OK** button.



When the **Open Associated Perspective?** dialog box appears on the screen, click **Open Perspective** button.

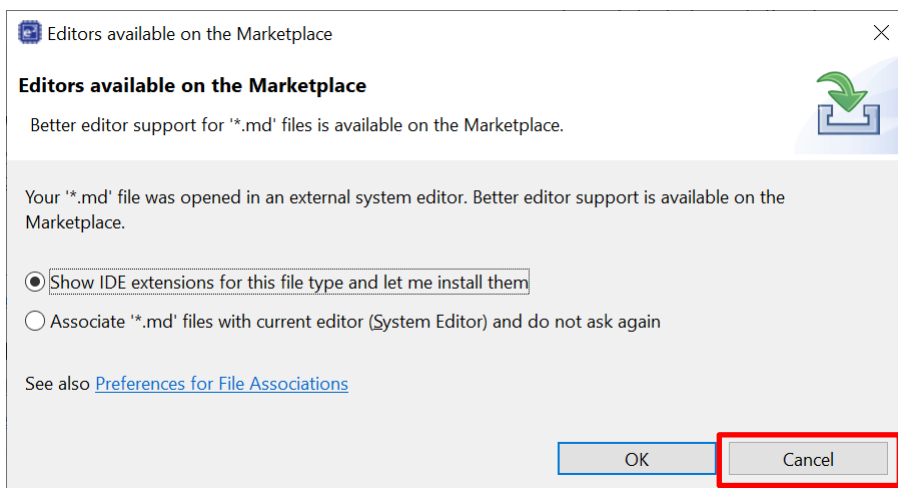


If the **Code Generating** dialog box appears on the screen, click **Proceed** button.



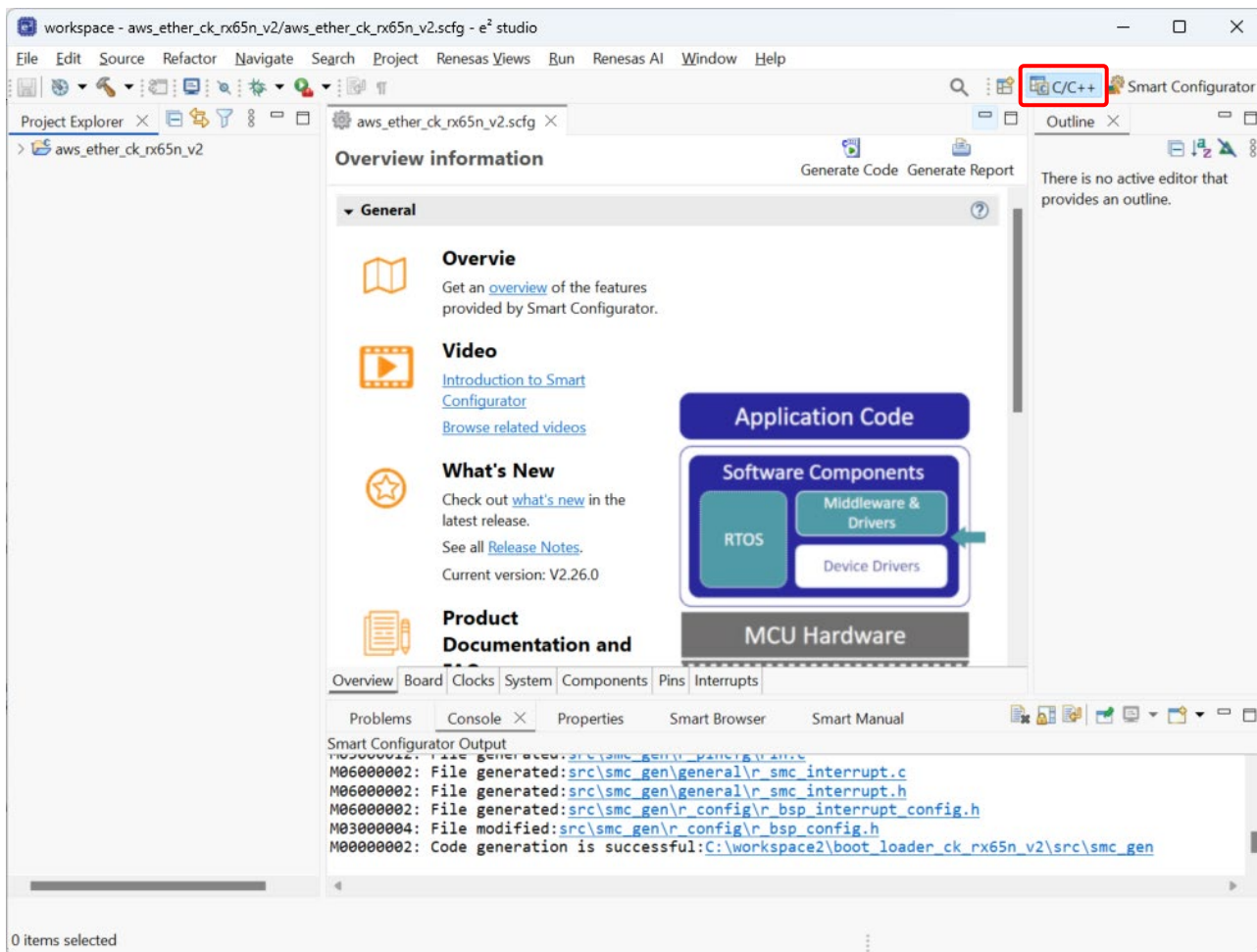
RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

When the **Editors available on the Marketplace** dialog box appear, click the **Cancel** button to close it.



This completes the procedure of creating the OTA update demo project in e² studio. If the **Welcome to e² studio** window appears, you may close it.

If **Project Explorer** is not visible, click the button for the **C/C++** perspective in the upper right corner of the window, then select **Window > Show View > Project Explorer**.

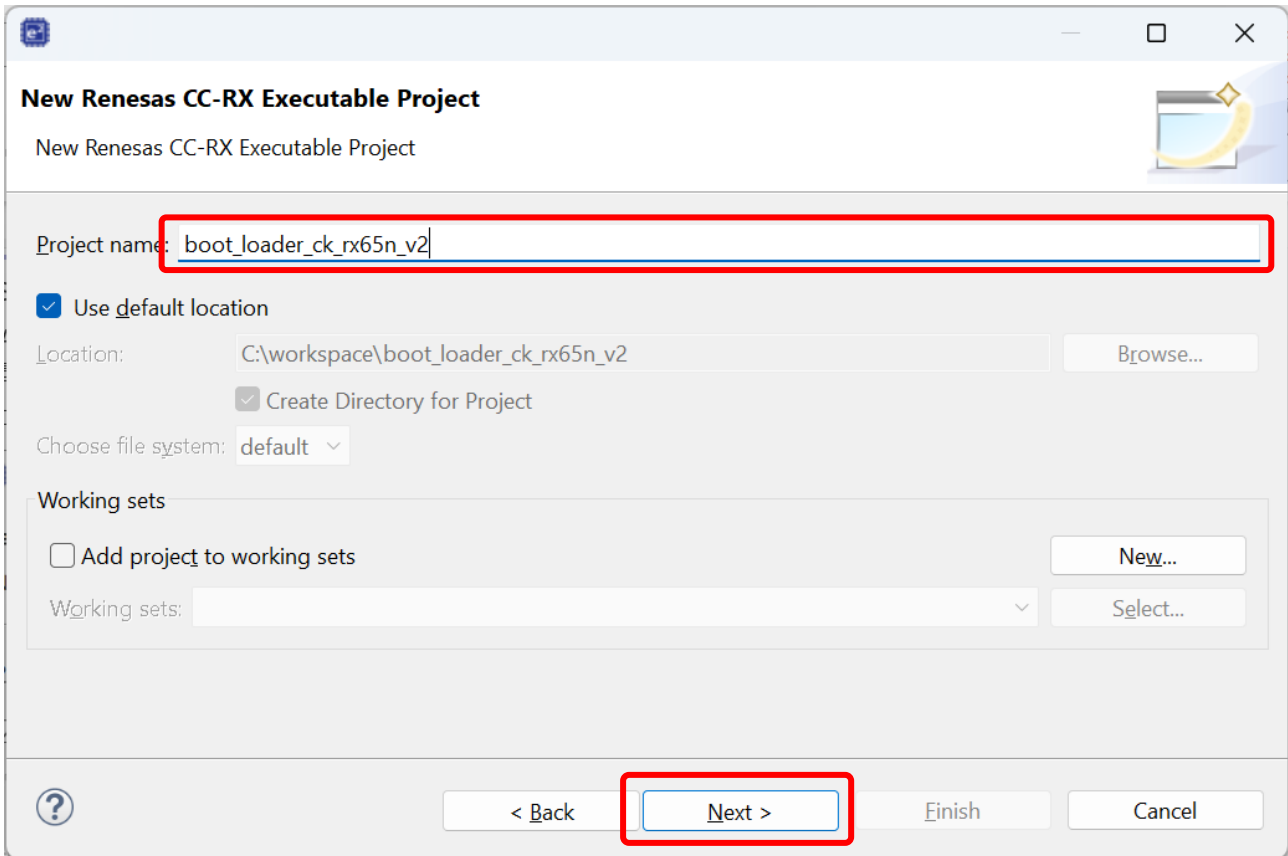


RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(4) Create the Secure Bootloader sample project.

Create the Secure Bootloader sample project in the same manner as the OTA sample project. The procedure involved is basically the same as the previous step.

Enter `boot_loader_ck_rx65n_v2` as the project name and click the **Next** button.

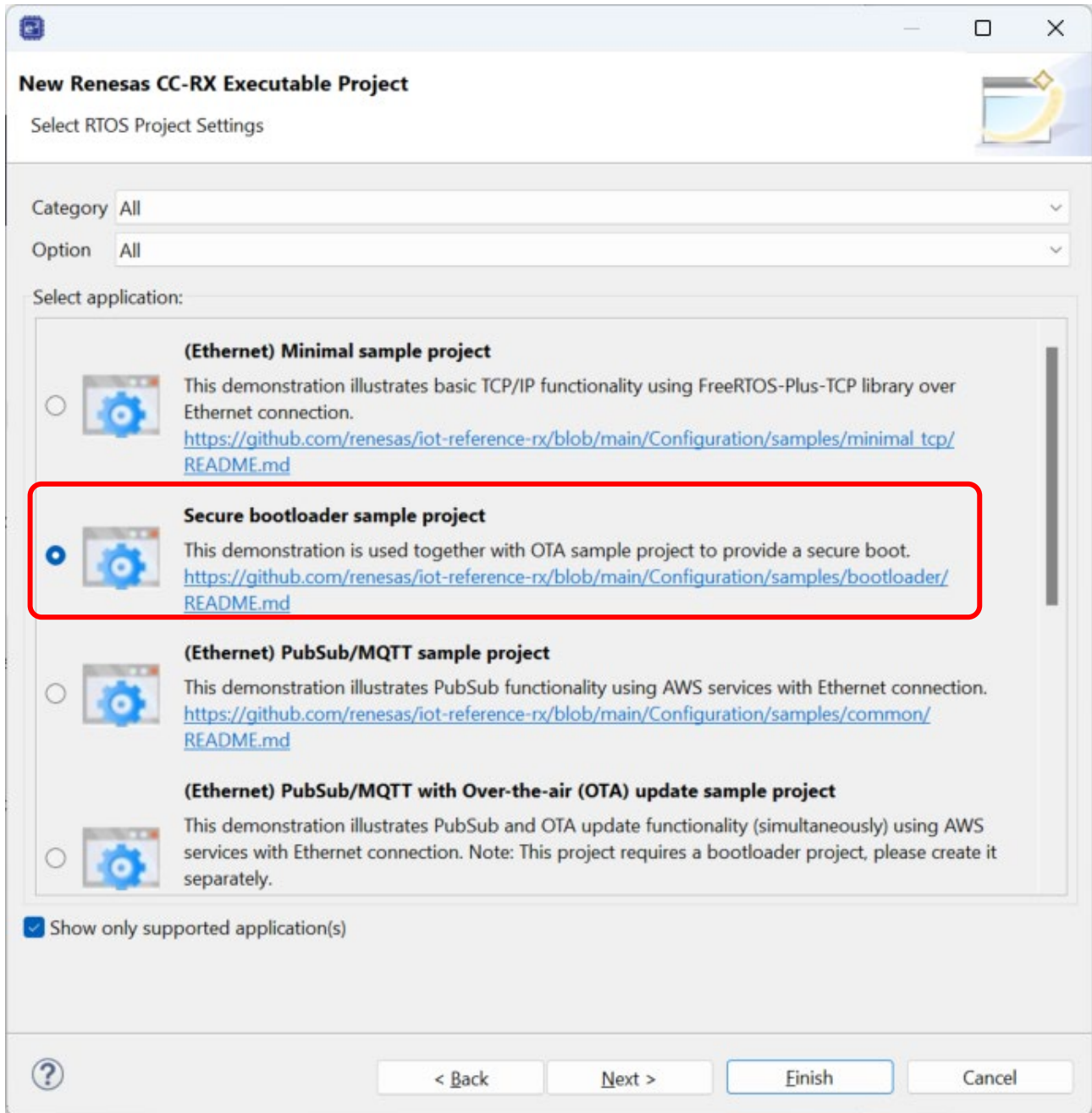


The screenshot shows the 'New Renesas CC-RX Executable Project' dialog box. The 'Project name' field is highlighted with a red box and contains the text 'boot_loader_ck_rx65n_v2'. The 'Next >' button at the bottom is also highlighted with a red box. Other fields include 'Location' set to 'C:\workspace\boot_loader_ck_rx65n_v2', 'Use default location' checked, 'Create Directory for Project' checked, and 'Choose file system' set to 'default'. The 'Working sets' section has 'Add project to working sets' unchecked and 'Working sets' empty. The bottom navigation buttons are '< Back', 'Next >', 'Finish', and 'Cancel'.

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

Next, select **Secure bootloader sample project** in the **Select RTOS Project Settings** dialog box.

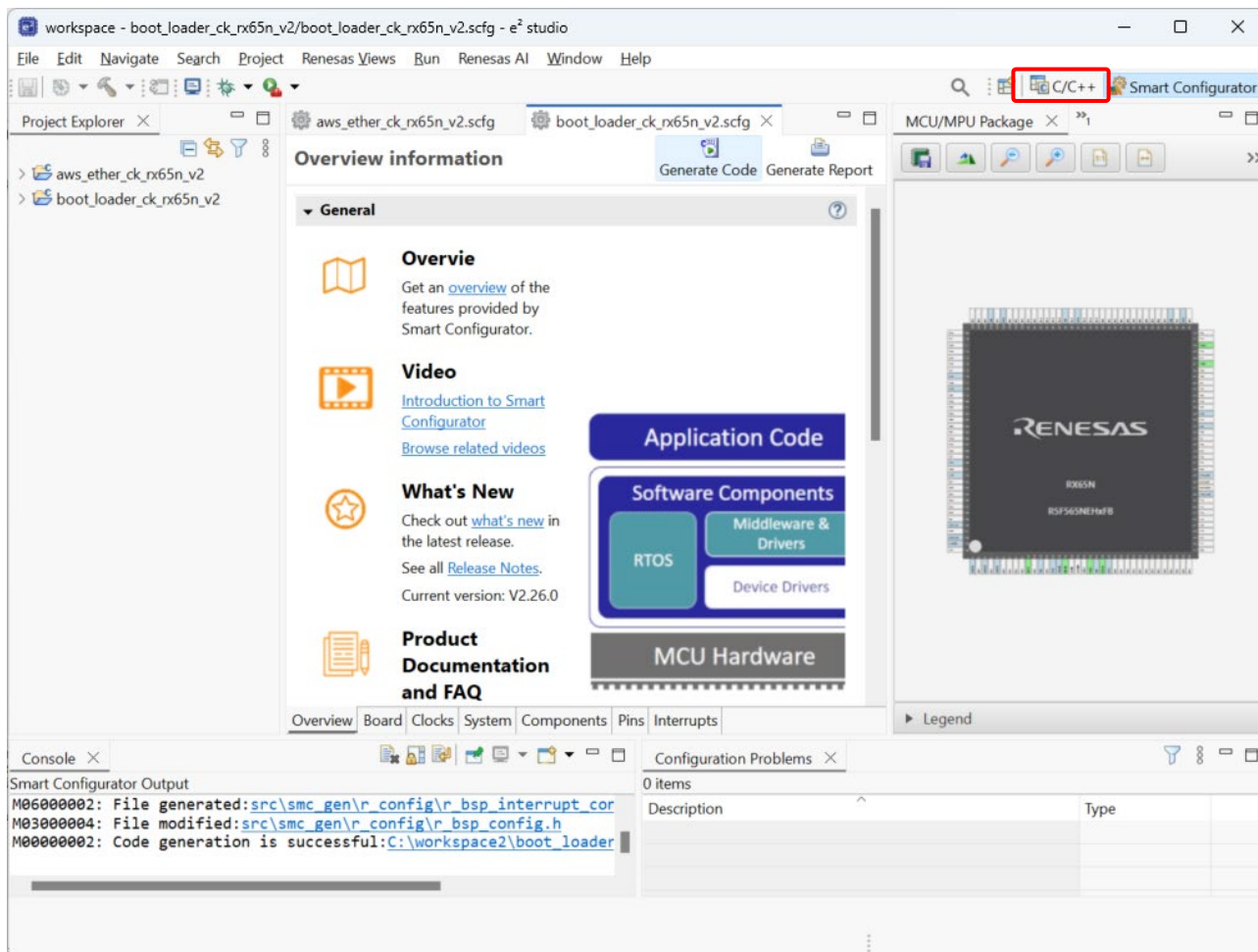
The other settings are the same as that of the OTA sample project.



RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

The projects appear in e² studio as shown below.

If Project Explorer is not visible, click the button for the **C/C++** perspective in the upper right corner of the window, then select **Window > Show View > Project Explorer**.



The procedure explained above uses an Ethernet project for the CC-RX compiler as an example. However, when selecting RTOS project settings, you can choose different combinations of connectivity and compiler options.

The following types of projects can be generated:

Demo type	Compiler
Demo project: Ethernet	CC-RX
	GCC
Demo project: Cellular (RYZ014A)	CC-RX
	GCC
Demo project: Wi-Fi (DA16600)	CC-RX
	GCC
Boot loader project	CC-RX
	GCC

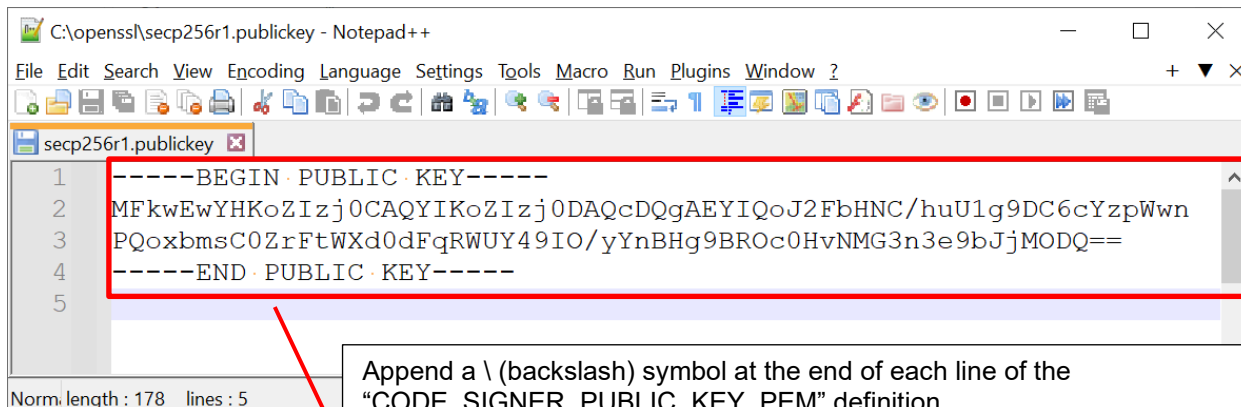
RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

4.2.2 Setting Up Projects

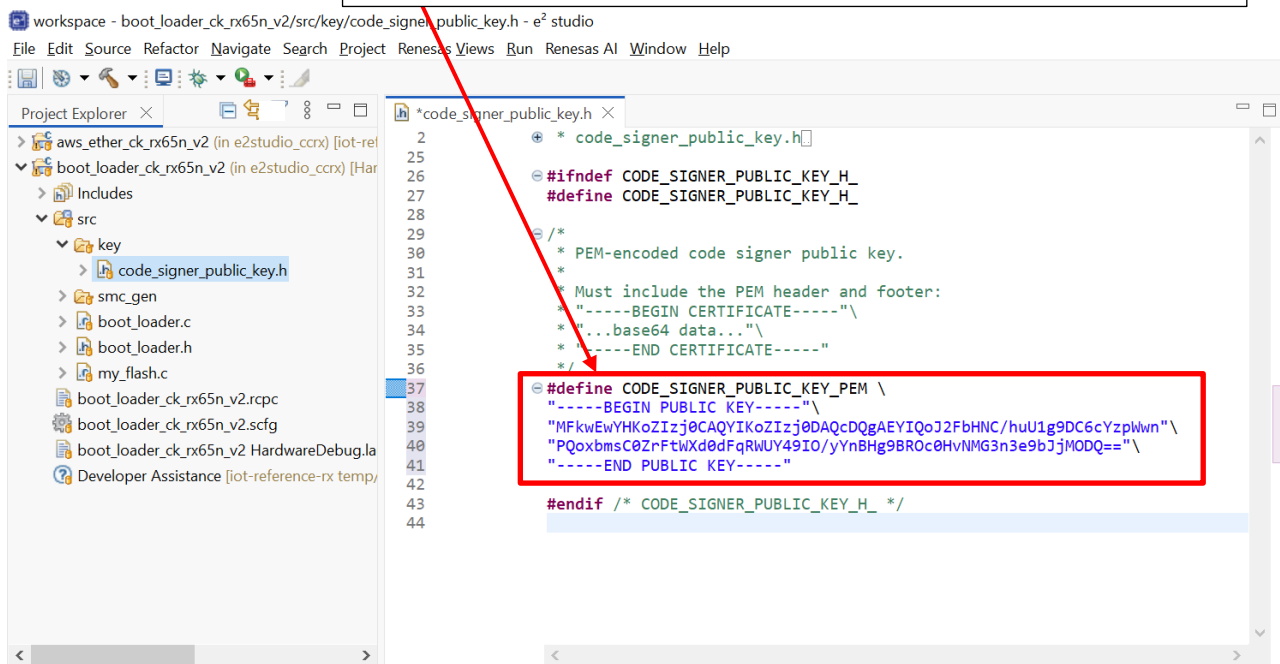
(1) Assign a public key to the boot_loader_ck_rx65n_v2 project.

To attach a public key to the Secure Bootloader project, copy the contents of the **secp256r1.publickey** file created in step 4.1(8), and paste the public key to the "CODE_SIGNER_PUBLIC_KEY_PEM" macro definition in the following file:

- CC-RX: \boot_loader_ck_rx65n_v2\studio_ccrx\src\key\code_signer_public_key.h
- GCC: \boot_loader_ck_rx65n_v2\studio_gc\src\key\code_signer_public_key.h



Append a \ (backslash) symbol at the end of each line of the "CODE_SIGNER_PUBLIC_KEY_PEM" definition. Each line must be enclosed in quotation marks ("), and do not forget to end the line with the \ symbol. (Example: "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx") However, do not enter the \ symbol at the end of the final line "-----END PUBLIC KEY-----".



RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(2) Set the definition for the OTA update demo to “Enable”.

Confirm that the project is configured to run the OTA update demo by setting the ENABLE_OTA_UPDATE_DEMO macro definition to 1 (Enable) in the following file:

- CC-RX: \aws_ether_ck_rx65n_v2\e2studio_ccrx\src\frtos_config\demo_config.h
- GCC: \aws_ether_ck_rx65n_v2\e2studio_gcc\src\frtos_config\demo_config.h

Note: The default value for the macro definition is 0

```

64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
o1
  
```

```

/* Select a combination of demos to run
 * These demo has not been evaluated outside of the combination of demos li
 *
 *     PUBSUB demo only :
 *     ENABLE_FLEET_PROVISIONING_DEMO (0) + ENABLE_OTA_UPDATE_DEMO
 *     PUBSUB demo with fleet provisioning :
 *     ENABLE_FLEET_PROVISIONING_DEMO (1) + ENABLE_OTA_UPDATE_DEMO
 *     PUBSUB and OTA over MQTT demo :
 *     ENABLE_FLEET_PROVISIONING_DEMO (0) + ENABLE_OTA_UPDATE_DEMO
 *     PUBSUB and OTA over MQTT demo with fleet provisioning :
 *     ENABLE_FLEET_PROVISIONING_DEMO (1) + ENABLE_OTA_UPDATE_DEMO
 */
/* demo is configured for PUBSUB */
/* Select demo combination to run. */

/* Please select a provisioning method
 * (0) : Pre-provisioning
 * (1) : Fleet provisioning
 */
#define ENABLE_FLEET_PROVISIONING_DEMO    (0)

/* Please select whether to enable or disable the OTA demo
 * (0) : OTA demo is disabled
 * (1) : OTA over MQTT demo is enabled
 */
#define ENABLE_OTA_UPDATE_DEMO            (1)
  
```

(3) Confirm the initial version of the project.

Confirm that the initial version of the project is 0.92 in the following file:

- CC-RX: \aws_ether_ck_rx65n_v2\e2studio_ccrx\src\frtos_config\demo_config.h
- GCC: \aws_ether_ck_rx65n_v2\e2studio_gcc\src\frtos_config\demo_config.h

Confirm that the version definitions are set as follows:

- APP_VERSION_MAJOR: 0
- APP_VERSION_MINOR: 9
- APP_VERSION_BUILD: 2

```

323
325
327
328
329
330
332
333
334
335
336
337
338
339
340
342
343
344
345
346
347
348
349
350
352
353
354
355
356
357
358
359
360
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
  
```

```

 * @brief Certificate used for validating code signing signatures in the OT
 * @brief Major version of the firmware.
 * @brief Minor version of the firmware.
 * @brief Build version of the firmware.
 * @brief Server's root CA certificate.
 * @brief The length of the queue used to hold commands for the agent.
  
```

```

#ifndef otapalconfigCODE_SIGNING_CERTIFICATE
#define otapalconfigCODE_SIGNING_CERTIFICATE    "Insert code signing ce
#endif

#define APP_VERSION_MAJOR    (0)
#define APP_VERSION_MINOR    (9)
#define APP_VERSION_BUILD    (2)

#define democonfigROOT_CA_PEM    (t1sSTARFIELD_ROOT_CERTIFIC
MQTT_AGENT_COMMAND_QUEUE_LENGTH    ( 25 )
  
```

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

4.2.3 Settings for the Individual Connectivity Types

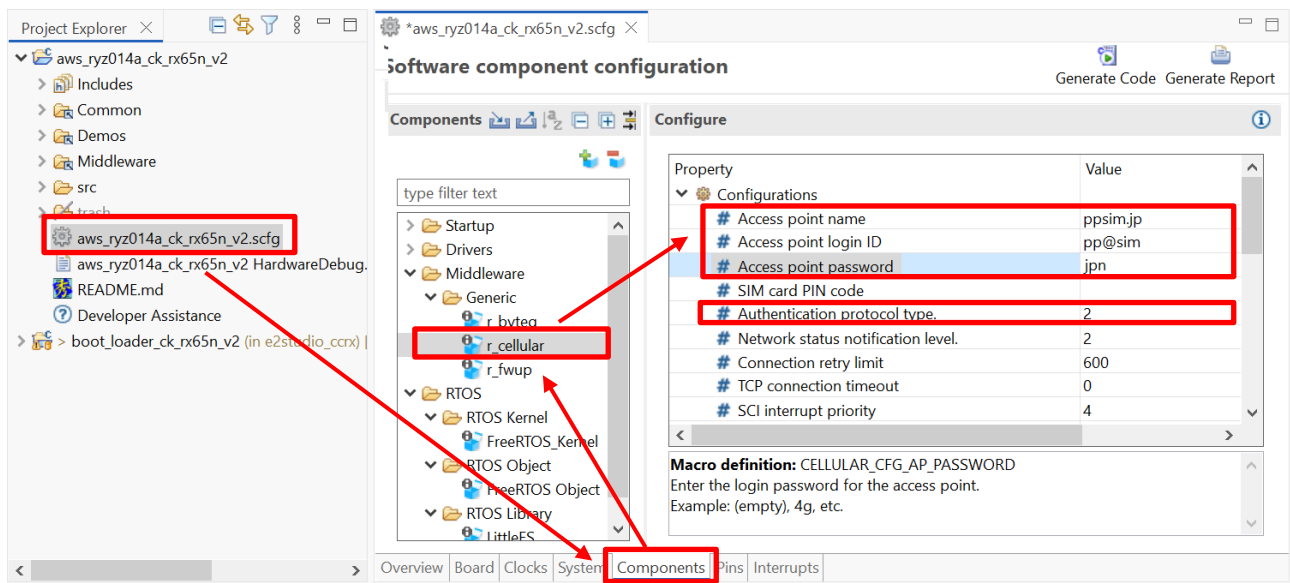
If you are using cellular or Wi-Fi connection, set up the corresponding connectivity module.

(1) Setting up the RYZ014A cellular module control module

When a cellular module is to be used for the AWS connection, set up the FIT module (`r_cellular`) for the RYZ014A cellular module control module.

Open "`aws_ryz014a_ck_rx65n_v2.scfg`" Smart Configurator file and click on the [Components] tab. Configure the following [`r_cellular`] settings according to the settings of your target SIM card:

- [Access point name]: Enter the access point name.
- [Access point login ID]: Enter the user ID for login.
- [Access point password]: Enter the password for login.
- [Authentication protocol type]: Enter "1" (PAP) or "2" (CHAP) as the type of authentication protocol.



RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

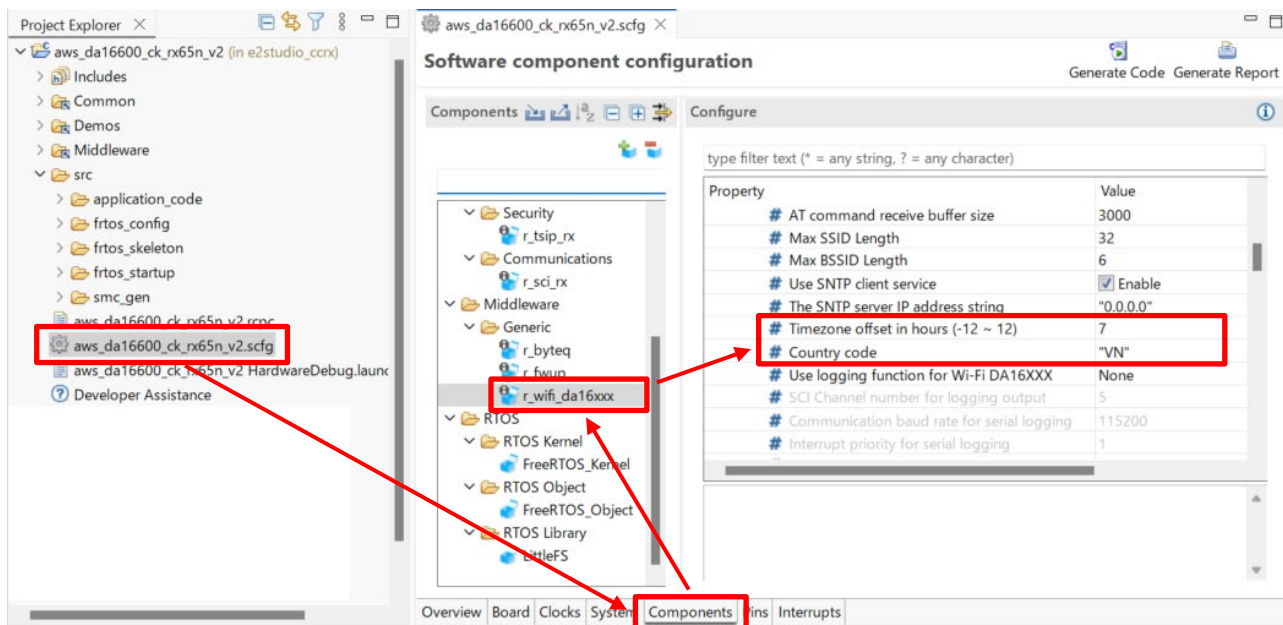
(2) Setting up the DA16600 Wi-Fi module

When a Wi-Fi module is to be used for the AWS connection, set up the "aws_clientcredential.h" configuration file and the FIT module (r_wifi_da16xxx) for the DA16xxx Wi-Fi module.

(a) Setting the country code and timezone

Open "aws_da16600_ck_rx65n_v2.scfg" Smart Configurator file and click on the [Components] tab. Configure the following [r_wifi_da16xxx] settings:

- [Timezone offset in hours]: Enter a value from -12 to 12 as the offset of the timezone from GMT.
- [Country code]: Enter the country code defined by the ISO 3166-1 alpha-2 standard such as US, JP, or CH.

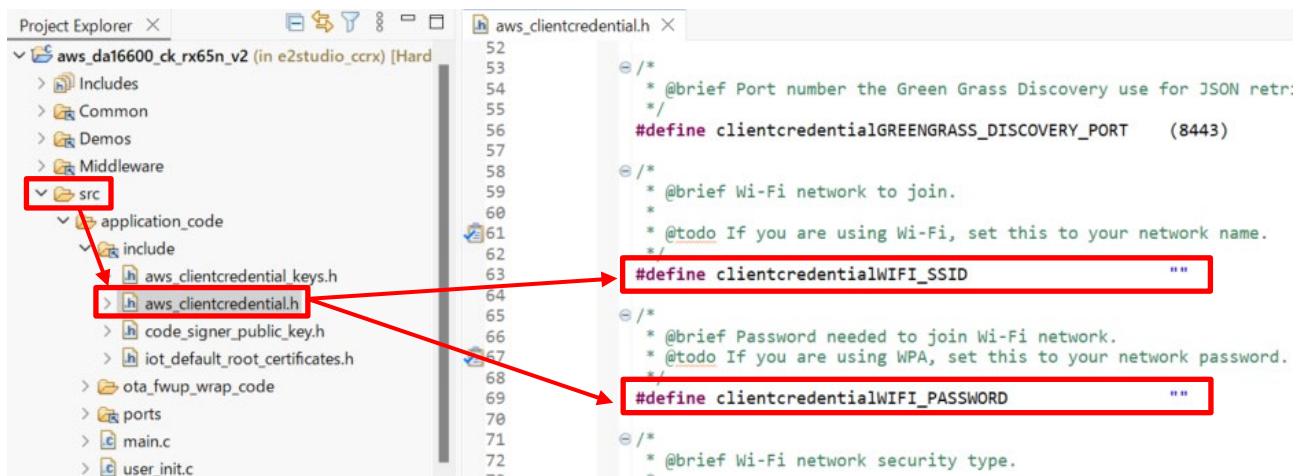


(b) Setting the Wi-Fi network

Set the Wi-Fi network for connection.

Configure the settings of the following macros in "src\application_code\include\aws_clientcredential.h".

- clientcredentialWIFI_SSID: Enter the access point name (SSID) for the Wi-Fi network.
- clientcredentialWIFI_PASSWORD: Enter the password for the Wi-Fi network.



Note: Enter the configuration values within the pair of double quotation ("") marks.

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

4.2.4 Creating the Initial Firmware

This section explains how to create the initial firmware by combining the boot loader (boot_loader_ck_rx65n_v2) and user application (aws_ether_ck_rx65n_v2).

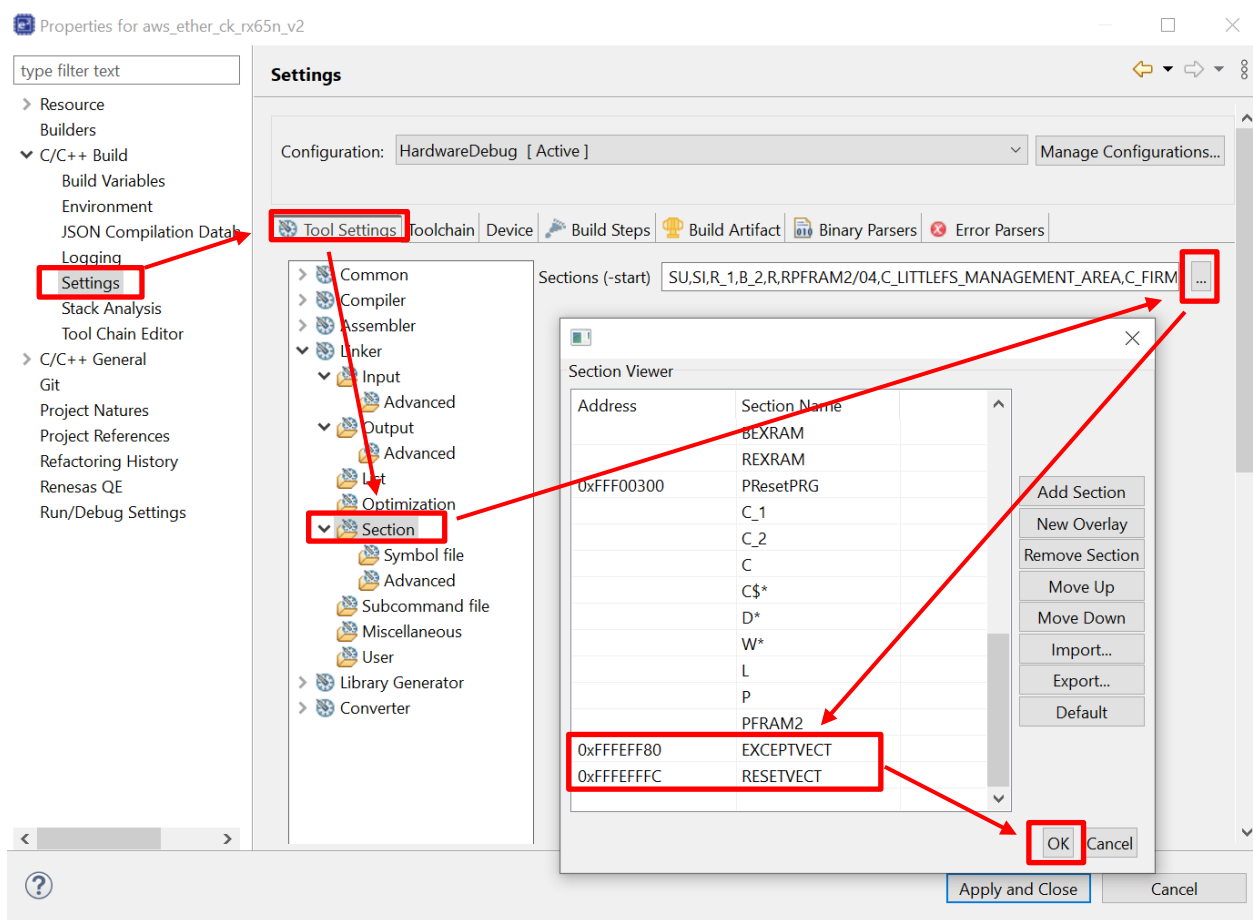
(1) Change the section setting of the user application (aws_ether_ck_rx65n_v2) and build the firmware.

Confirm the changes to the linker section setting in the user application and build the firmware. Right-click on the **aws_ether_ck_rx65n_v2** project, and [Properties].

Select [Settings] in the [C/C++ Build] menu.

Under [Tool Settings] tab, select [Linker] → [Section] → [...]. This opens the “Section Viewer” dialog box. Modify the address of respective sections as shown below.

- EXCEPTVECT: 0xFFFFE**F**F80
- RESETVECT: 0xFFFFE**F**FC

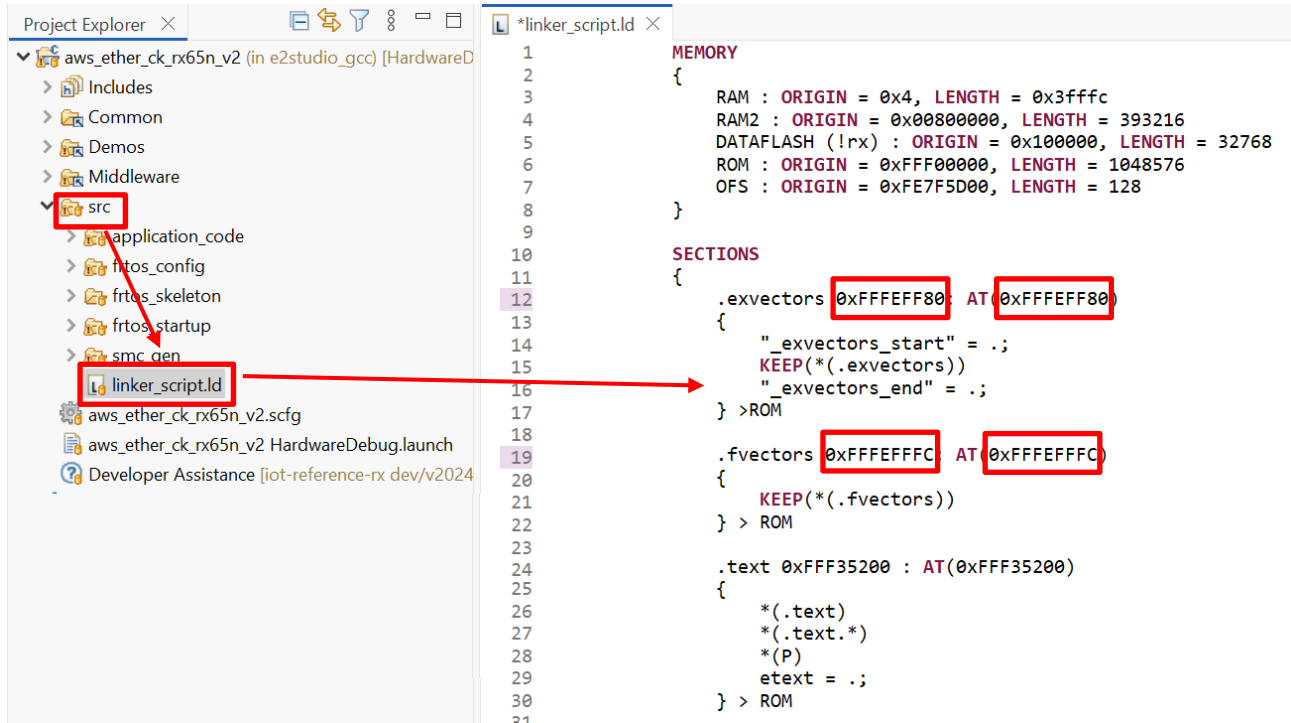


After the section setting is complete, build the **aws_ether_ck_rx65n_v2** project in the e² studio to create the firmware.

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

When creating a GCC project, the method for changing the linker section setting is different. In Project Explorer, open the "src\linker_script.ld" file and change the following addresses.

- .exvectors: 0xFFFFE80 (2 places)
- .fvectors: 0xFFFFE80 (2 places)



After the section setting is complete, build the **aws_ether_ck_rx65n_v2** project in the e² studio to create the firmware.

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

- (2) Create the boot loader.

Build a boot loader project to create the Secure Bootloader.

- (3) Use the Renesas Image Generator to generate the initial firmware.

Use the Renesas Image Generator to generate the initial firmware.

Start by placing the following files in the Renesas Image Generator folder.

- Output of the build process in 4.2.4(1): `aws_ether_ck_rx65n_v2.mot`
- Output of the build process in 4.2.4(2): `boot_loader_ck_rx65n_v2.mot`
- Private key created in 4.1(7): `secp256r1.privatekey`

Open the command prompt window, navigate to the Renesas Image Generator folder, and execute the following command to generate the initial firmware file (**userprog.mot**).

The execution of this command may take up to several minutes.

```
python image-gen.py -iup aws_ether_ck_rx65n_v2.mot -ip RX65N_DualBank_ImageGenerator_PRM.csv -o userprog -ibp boot_loader_ck_rx65n_v2.mot -key secp256r1.privatekey -vt ecdsa -ff RTOS
```

- (4) Install the flash programming tool (Renesas Flash Programmer).

Access the [download site](#) of the flash programming tool, then download and install “**Renesas Flash Programmer V3.22.00 Windows**”.

- (5) Connect the PC to the CK-RX65N v2 board.

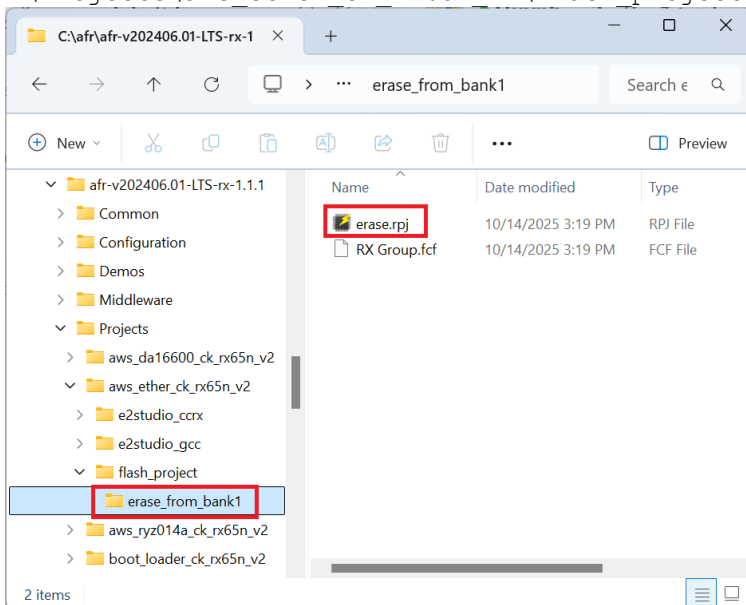
Follow the CK-RX65N v2 hardware setup instructions in “2.6 Connecting the CK-RX65N v2” to connect the PC to the CK-RX65N v2 board.

- (6) Open the erase project in the Renesas Flash Programmer.

In the Renesas Flash Programmer, select [Open Project] in the File menu, and open the “erase.rpj” project.

The erase project can be found in the following folder in your downloaded FreeRTOS package.

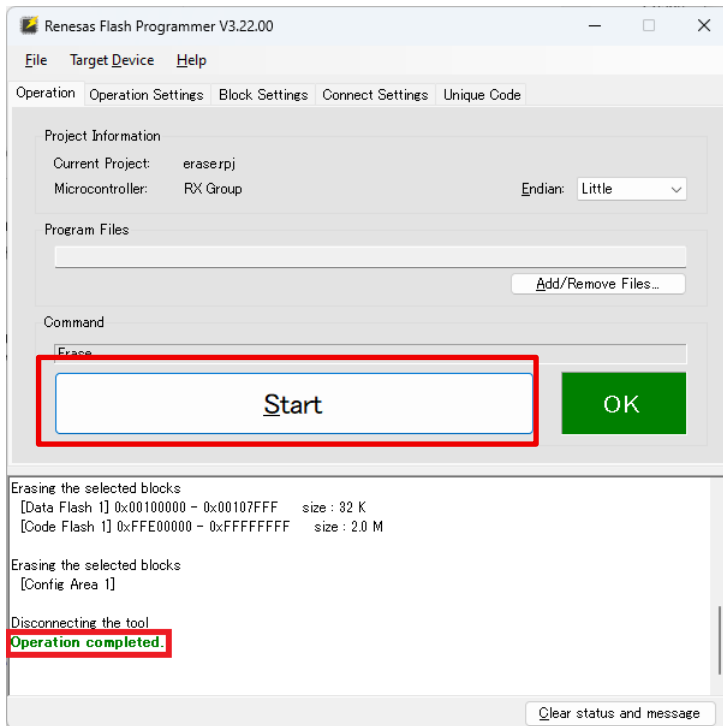
```
\Projects\aws_ether_ck_rx65n_v2\flash_project\erase_from_bank1
```



RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(7) Erase the device.

Click on the [Start] button to commence erasing the device.
Erasure is complete when the message “Operation completed” is displayed.

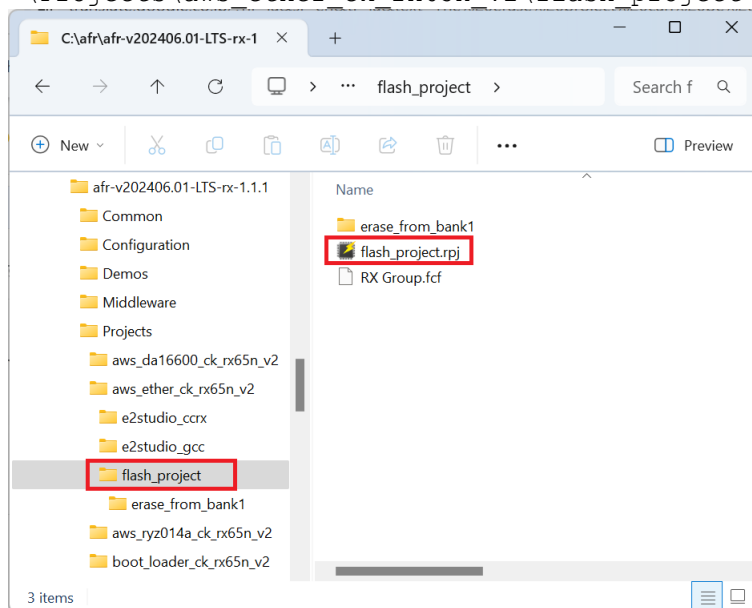


Note: If the message “Error (E3000107): This device does not match the connection parameters.” is output, proceed to step (8) because this indicates that the device information has already been initialized.

(8) Open the flash programming project in the Renesas Flash Programmer.

In the Renesas Flash Programmer, select [Open Project] in the File menu, and open the “flash_project.rpj” project.
The flash write project can be found in the following folder in your downloaded FreeRTOS package.

```
\Projects\aws_ether_ck_rx65n_v2\flash_project
```

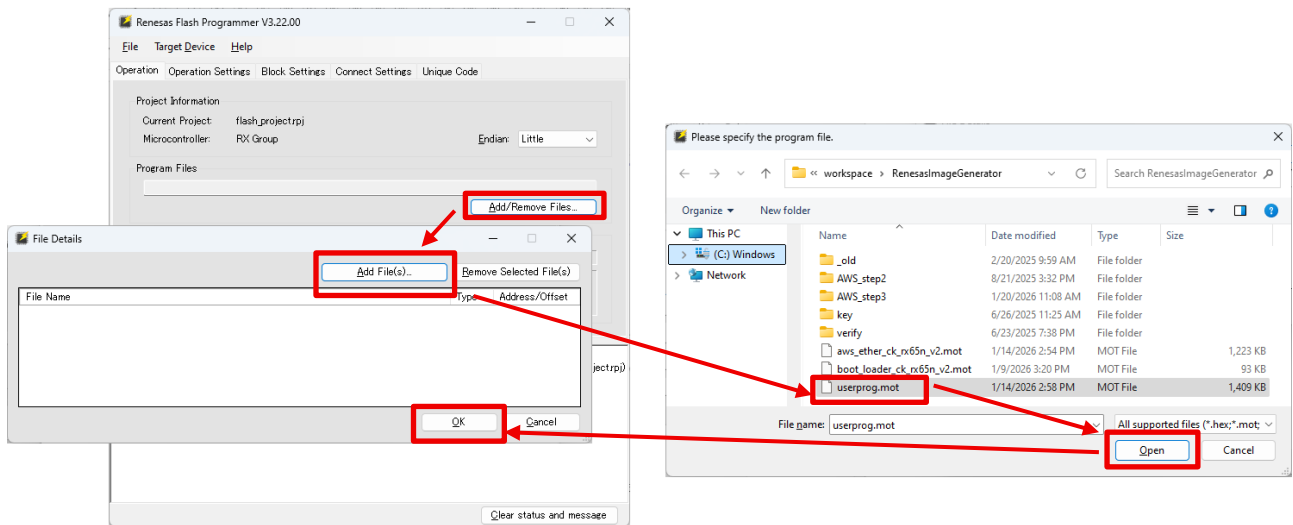


RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(9) Select the initial firmware.

Next, click on [Add/Remove Files], then click on the [Add File(s)] button.

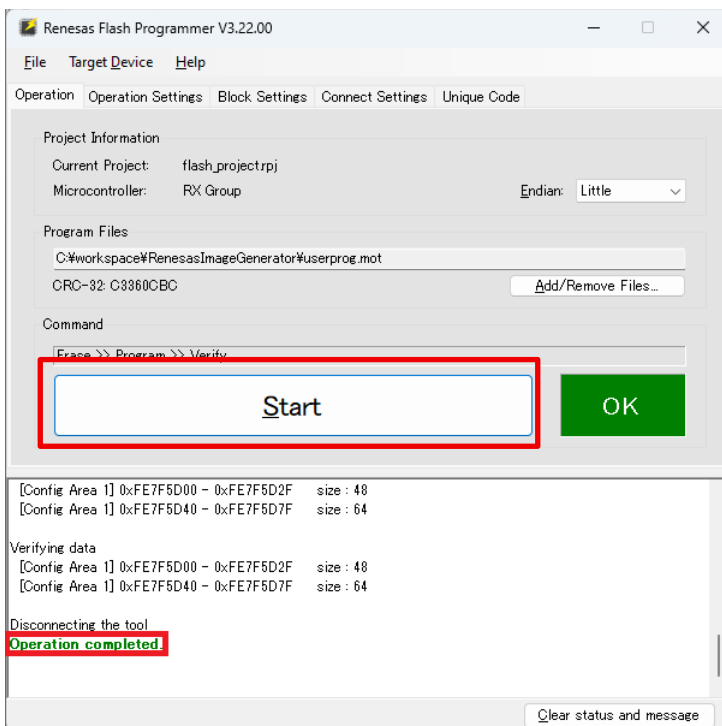
In the [File Details] dialog box, select the initial firmware (userprog.mot) that was created in 4.2.3(3).



(10) Write the initial firmware.

Click on the [Start] button to write the initial firmware.

Writing of the initial firmware is complete when the message “Operation completed” is displayed.



Note: If the message “Error (E3000107): This device does not match the connection parameters.” is output, perform the erasure processing in step (7) because this indicates that the device information has not been initialized.

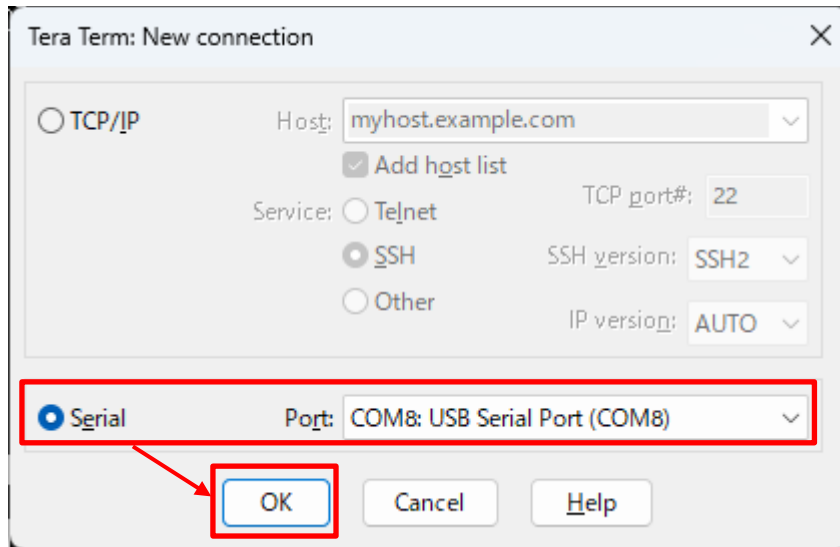
4.2.5 Registering AWS IoT Information

This section explains how to set AWS IoT information in Tera Term by running the aws_ether_ck_rx65n_v2 project. The information set through this process is written to the data flash memory.

- (1) Start Tera Term and set the serial port.

Start Tera Term, select [New Connection] in the File menu.

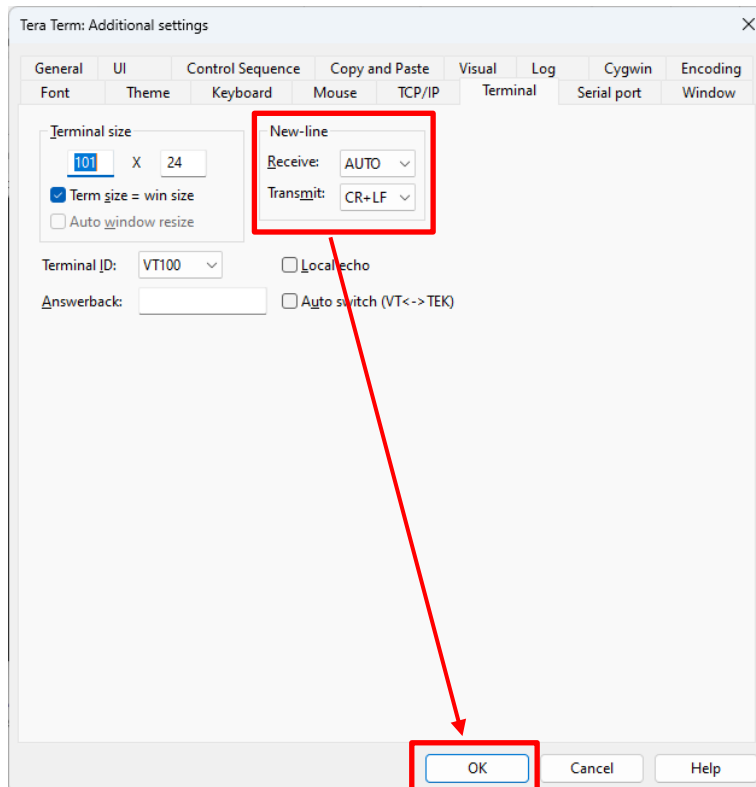
In the [New Connection] dialog box, select the [Serial] checkbox, then click on [OK].



- (2) Set the new-line character.

Set the new-line character for transmission and reception through the serial port.

Select [Terminal] in the Setup menu. In the [New-line] field of the dialog box that appears, select "Auto" for [Receive] and "CR+LF" for [Transmit], then click on [OK].

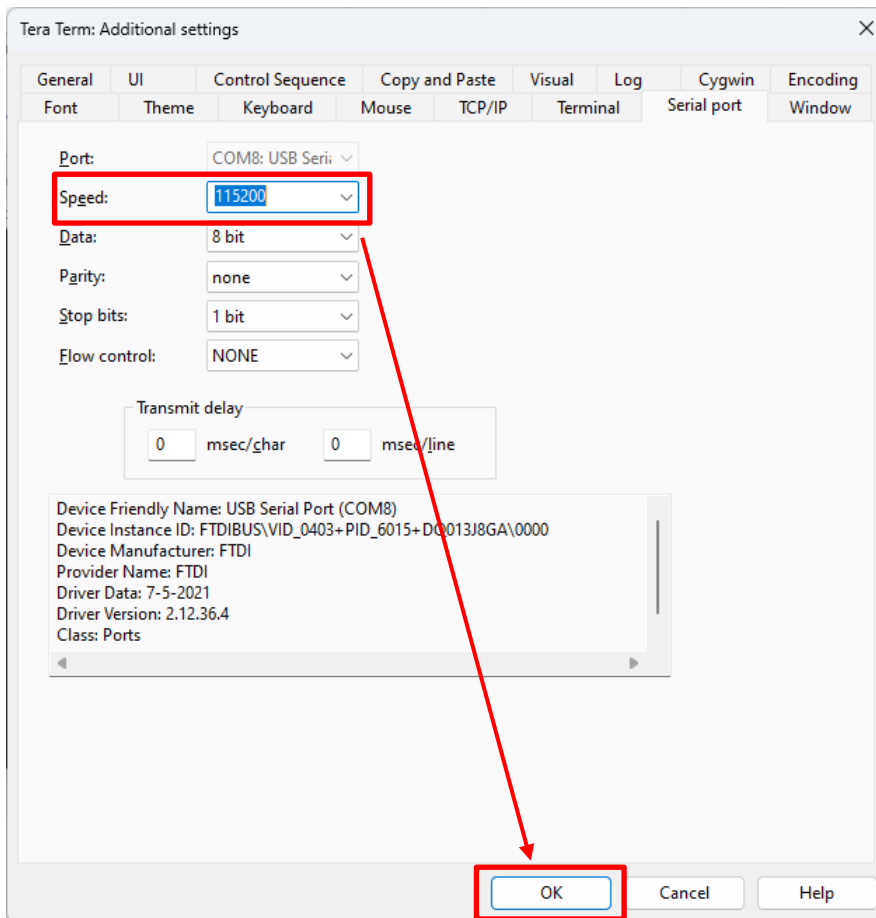


RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

- (3) Set the serial communications speed.

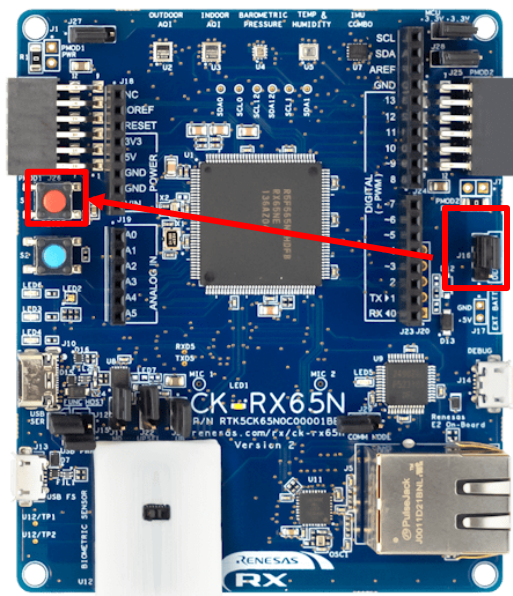
Select [Serial port...] in the Setup menu.

In the dialog box that appears, select "115200" for [Speed], then click on [New setting].



- (4) Reset the target board and execute the firmware.

Set jumper block J16 on the CK-RX65N v2 board to the RUN side (J16 2-3 short), and then press the RESET switch. The initial firmware will start running.

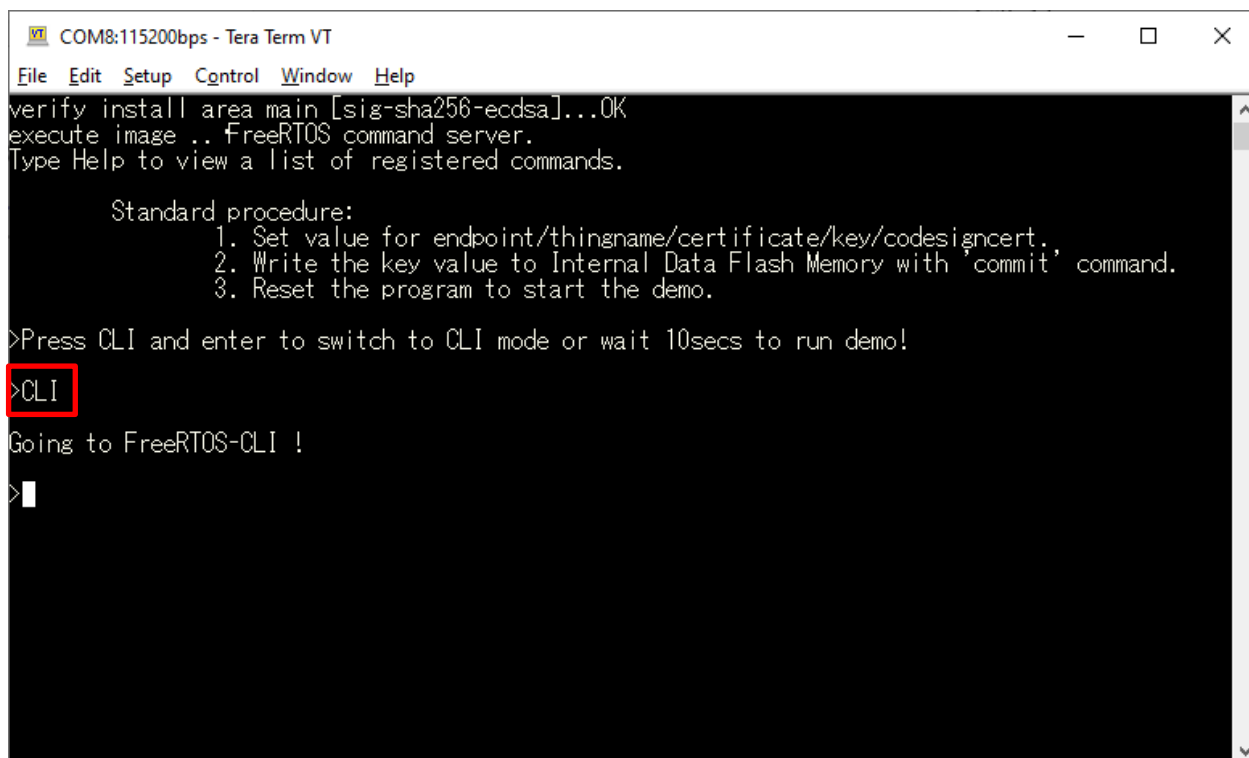


RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(5) Display the CLI menu.

When the initial firmware starts running, a menu will appear in the Tera Term window.

Within 10 seconds of this, enter "CLI" and press the Enter key.



```
COM8:115200bps - Tera Term VT
File Edit Setup Control Window Help
verify install area main [sig-sha256-ecdsa]...OK
execute image .. FreeRTOS command server.
Type Help to view a list of registered commands.

Standard procedure:
  1. Set value for endpoint/thingname/certificate/key/codesigncert.
  2. Write the key value to Internal Data Flash Memory with 'commit' command.
  3. Reset the program to start the demo.

>Press CLI and enter to switch to CLI mode or wait 10secs to run demo!
>CLI
Going to FreeRTOS-CLI !
>|
```

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(6) Register the device certificate.

Register the certificate that was downloaded in 3.3.4(2) to the target board.

Enter "conf set cert" in Tera Term, then drag and drop the certificate file (certificate.pem.crt) to the Tera Term window to send the file. Finally, press the Enter key in the Tera Term window.

Note: Change the End-Of-Line (EOL) code of the certificate file to LF with a text editor before dragging and dropping the file.

Enter "conf set cert" and a space, drag and drop the certificate file to the Tera Term window. Select "Binary", click on the [OK] button, and then press the Enter key.

```
>conf set cert ----BEGIN CERTIFICATE-----  
  
----END CERTIFICATE-----  
  
OK.  
>
```

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(7) Register the private key.

Register the private key that was downloaded in 3.3.4(2) to the target board.

Enter "conf set key " in Tera Term, then drag and drop the private key file (private.pem.crt) to the Tera Term window to send the file. Finally, press the Enter key in the Tera Term window.

Note: Change the End-Of-Line (EOL) code of the private key file to LF with a text editor before dragging and dropping the file.

The image shows a Windows File Explorer window at the path C:\aws\cert. It contains three files: certificate.pem.crt (Security Certificate, 2 KB), private.pem.key (highlighted with a red box), and public.pem.key. A Tera Term window titled "Tera Term: File Drag and Drop" is overlaid on the File Explorer. The dialog shows the source file path as C:\aws\private.pem.key and asks "Are you sure that you want to send the file content?". The "Send File (Paste content of file)" option is selected, and the "Binary" checkbox is checked. The "OK" button is also highlighted with a red box. A red arrow points from the "private.pem.key" file in the File Explorer to the "Send File" option in the dialog. Another red arrow points from the "OK" button in the dialog to the Tera Term window. The Tera Term window shows a command prompt with the command >conf set key -----BEGIN RSA PRIVATE KEY----- and a large blue area representing the file content. At the bottom of the Tera Term window, the command -----END RSA PRIVATE KEY----- is visible, followed by "OK.".

Enter "conf set key" and a space, drag and drop the private key file to the Tera Term window. Select "Binary", click on the [OK] button, and then press the Enter key.

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(8) Register the thing name and the endpoint.

Register the thing name that was set in 3.3.4(1) and the endpoint that you collected in 3.3.4(5) to the target board.

Execute the following commands in Tera Term.

- `conf set thingname <thing name>`
- `conf set endpoint <endpoint name>`

```
>conf set thingname rx65n_ota_demo_thing
OK.
>conf set endpoint [redacted]ot.ap-northeast-1.amazonaws.com
OK.
```

(9) Register the certificate for code signing verification.

Register the key pair certificate (secp256r1.crt) that was generated in 4.1(6) to the target board.

Enter “`conf set codesigncert`” in Tera Term, then drag and drop the key pair certificate file (secp256r1.crt) to the Tera Term window to send the file. Finally, press the Enter key in the Tera Term window.

Note: Change the End-Of-Line (EOL) code of the certificate file to LF with a text editor before dragging and dropping the file.

Enter “conf set codesigncert” and a space, drag and drop the certificate file to the Tera Term window. Select “Binary”, click on the [OK] button, and then press the Enter key.


```
>conf set codesigncert -----BEGIN CERTIFICATE-----
[Large Blue Area]
-----END CERTIFICATE-----
OK.
```

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(10) Write the settings to data flash memory.

Commit the AWS IoT settings (write the settings to data flash memory).
Execute the following command in Tera Term.

```
> conf commit
```



```
>conf commit
0 4472481 [CLI] Destroyed Certificate.
1 4472485 [CLI] Write certificate...
2 4472545 [CLI] Destroyed Private key.
3 4472685 [CLI] Write Private key...
Configuration saved to Data Flash and used 2879 bytes.
```

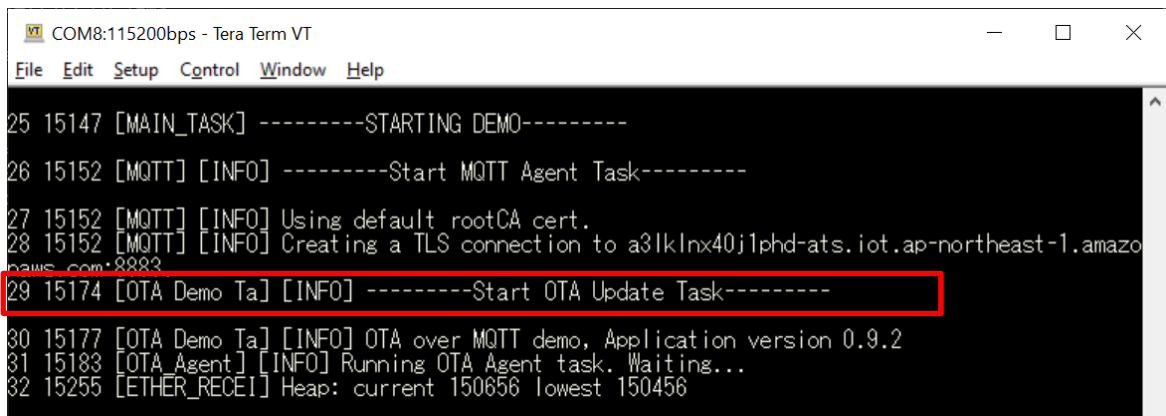
(11) Execute a reset.

Restart the firmware by executing a software reset by executing the following command in Tera Term.

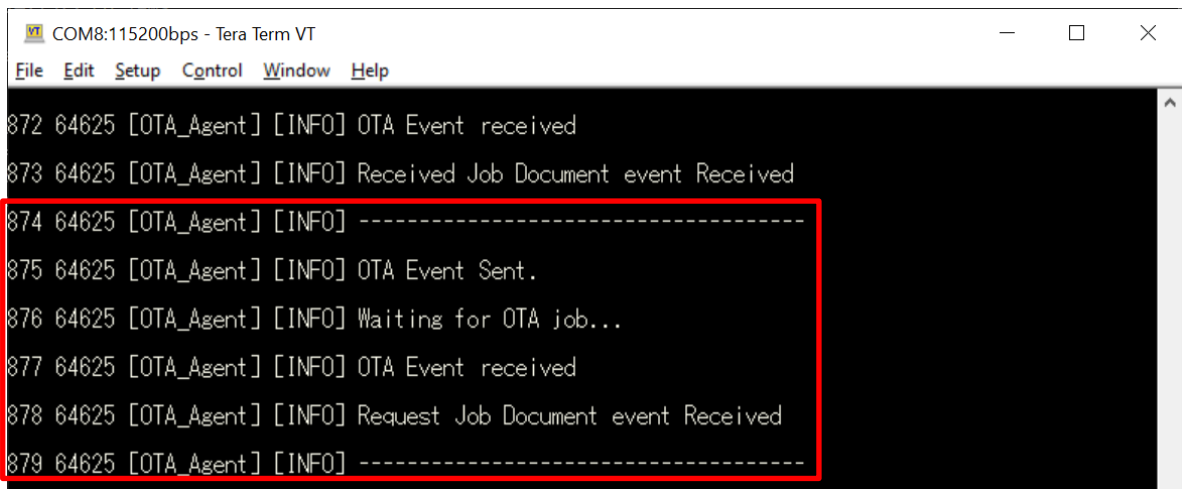
```
> reset
```

After the reset processing is complete, the demo will start after 10 seconds if nothing is entered in Tera Term.

Confirm that a communications log is displayed as shown below, OTA tasks are executed, and the application is waiting for OTA jobs.



```
COM8:115200bps - Tera Term VT
File Edit Setup Control Window Help
25 15147 [MAIN_TASK] -----STARTING DEMO-----
26 15152 [MQTT] [INFO] -----Start MQTT Agent Task-----
27 15152 [MQTT] [INFO] Using default rootCA cert.
28 15152 [MQTT] [INFO] Creating a TLS connection to a3lklnx40j1phd-ats.iot.ap-northeast-1.amazonaws.com:8883
29 15174 [OTA Demo Ta] [INFO] -----Start OTA Update Task-----
30 15177 [OTA Demo Ta] [INFO] OTA over MQTT demo, Application version 0.9.2
31 15183 [OTA_Agent] [INFO] Running OTA Agent task. Waiting...
32 15255 [ETHER_RECE1] Heap: current 150656 lowest 150456
```



```
COM8:115200bps - Tera Term VT
File Edit Setup Control Window Help
872 64625 [OTA_Agent] [INFO] OTA Event received
873 64625 [OTA_Agent] [INFO] Received Job Document event Received
874 64625 [OTA_Agent] [INFO] -----
875 64625 [OTA_Agent] [INFO] OTA Event Sent.
876 64625 [OTA_Agent] [INFO] Waiting for OTA job...
877 64625 [OTA_Agent] [INFO] OTA Event received
878 64625 [OTA_Agent] [INFO] Request Job Document event Received
879 64625 [OTA_Agent] [INFO] -----
```

5. Updating the Firmware

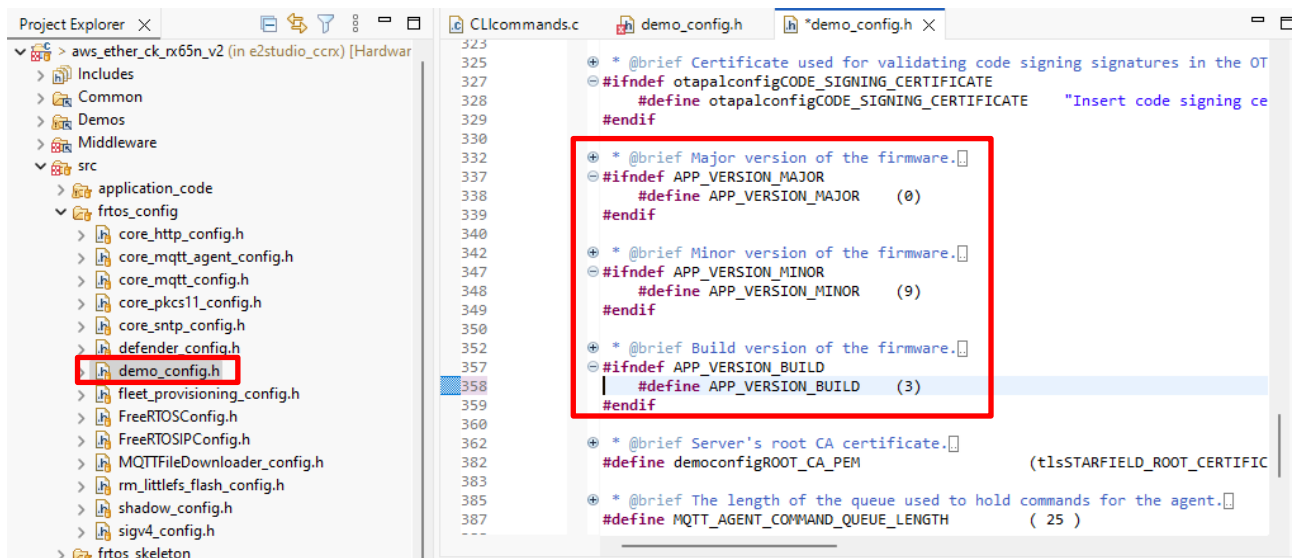
5.1 Creating the Update Firmware

5.1.1 Changing the Firmware Version

(1) Change the firmware version.

Change the version of the update firmware to “v0.9.3”.

Repeat the build process, this time with “3” specified for the APP_VERSION_BUILD definition in laws_ether_ck_rx65n_v2\src\frtos_config\demo_config.h.



The screenshot shows the Project Explorer on the left with the file tree expanded to 'demo_config.h'. The main editor window displays the content of 'demo_config.h' with line numbers 323 to 387. A red rectangular box highlights the following code block:

```
337 * @brief Major version of the firmware.[]
338 #ifndef APP_VERSION_MAJOR
339 #define APP_VERSION_MAJOR (0)
340 #endif
341 * @brief Minor version of the firmware.[]
342 #ifndef APP_VERSION_MINOR
343 #define APP_VERSION_MINOR (9)
344 #endif
345 * @brief Build version of the firmware.[]
346 #ifndef APP_VERSION_BUILD
347 | #define APP_VERSION_BUILD (3)
348 #endif
```

(2) Use the Renesas Image Generator to generate the update firmware.

Overwrite the file in the Renesas Image Generator folder with the firmware (aws_ether_ck_rx65n_v2.mot) that was rebuilt in 5.1.1(1) above, then execute the following command in the command prompt.

The execution of this command may take up to several minutes.

```
python image-gen.py -iup aws_ether_ck_rx65n_v2.mot -ip RX65N_DualBank_ImageGenerator_PRM.csv -o user_093 -key secp256r1.privatekey -vt ecdsa -ff RTOS
```

This command will generate the update firmware file (user_093.rsu).

5.2 Updating the Firmware

In AWS, upload the update firmware and create an OTA update job to update the firmware.

Use the AWS CLI to perform the following steps. As a pre-requisite (as explained in 3.2), using AWS CLI in the command prompt, log in to AWS and navigate to the work folder that was created in 3.1.5.

5.2.1 Uploading the Update Firmware to AWS

Upload the update firmware to the S3 bucket of AWS.

- (1) Copy the firmware to the work folder.

Copy the update firmware (user_093.rsu) that was created in 5.1.1(2) to the work folder that was created in 3.1.5.

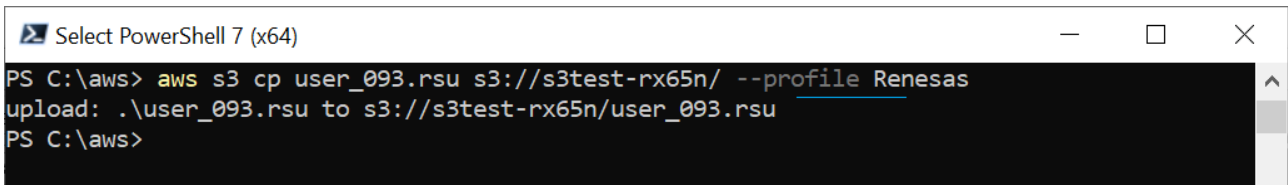
- (2) Upload the firmware to the S3 bucket.

Upload the created firmware to the S3 bucket.
Enter the following command.

```
> aws s3 cp <FIRMWARE_NAME> s3://<BUCKET_NAME>/ --profile <PROFILE_NAME>
```

- Enter the file name of the update firmware into <FIRMWARE_NAME>.
- Enter the bucket name that was created in 3.3.2(1) into <BUCKET_NAME>.

After the command has been executed, the display in the window will be as shown below.



```
Select PowerShell 7 (x64)
PS C:\aws> aws s3 cp user_093.rsu s3://s3test-rx65n/ --profile Renesas
upload: .\user_093.rsu to s3://s3test-rx65n/user_093.rsu
PS C:\aws>
```

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(3) Confirm the version ID.

Confirm the version ID of the update firmware that was uploaded.
Enter the following command.

```
> aws s3api list-object-versions --bucket <BUCKET_NAME> --prefix  
<FIRMWARE_NAME> --profile <PROFILE_NAME>
```

- Enter the S3 bucket name that was uploaded in step (2) into <BUCKET_NAME>.
- Enter the file name of the update firmware into <FIRMWARE_NAME>.

After the command has been executed, the display in the window will be as shown below.

```
PS C:\aws> aws s3api list-object-versions --bucket s3test-rx65n --prefix user_093.rsu --profile Renesas  
{  
  "Versions": [  
    {  
      "ETag": "\"[REDACTED]\"",  
      "ChecksumAlgorithm": [  
        "CRC64NVME"  
      ],  
      "ChecksumType": "FULL_OBJECT",  
      "Size": 364288,  
      "StorageClass": "STANDARD",  
      "Key": "user_093.rsu",  
      "VersionId": "[REDACTED]",  
      "IsLatest": true,  
      "LastModified": "2025-04-17T05:27:39+00:00",  
      "Owner": {  
        "DisplayName": "[REDACTED]",  
        "ID": "[REDACTED]"  
      }  
    },  
    {  
      "ETag": "\"[REDACTED]\"",  
      "ChecksumAlgorithm": [  
        "CRC64NVME"  
      ],  
      "ChecksumType": "FULL_OBJECT",  
      "Size": 364288,  
      "StorageClass": "STANDARD",  
      "Key": "user_093.rsu",  
      "VersionId": "[REDACTED]",  
      "IsLatest": false,  
    }  
  ]  
}
```

Make a note of the version ID of the update firmware file (VersionId: text inside the red frame in the above figure) which is displayed. You will need it in a later process.
Read the VersionId field of the latest version, i.e., that for which "true" is displayed in the "IsLatest" field (within the yellow frame) one line below. When a file with the same name has been uploaded multiple times, multiple entries will appear under the "Versions" field in the above figure.

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

5.2.2 Creating a Code Signing Certificate and Profile

Create a code signing certificate to ensure the reliability of the update firmware that was uploaded in an OTA update.

The procedure in this section can be skipped if a code signing certificate was uploaded and a profile was created in the past. Write the profile name created in this section to the json file when creating an OTA job.

(1) Prepare the certificate and key files.

Copy the following three files that were created in 4.1(9) to the work folder that was created in 3.1.5.

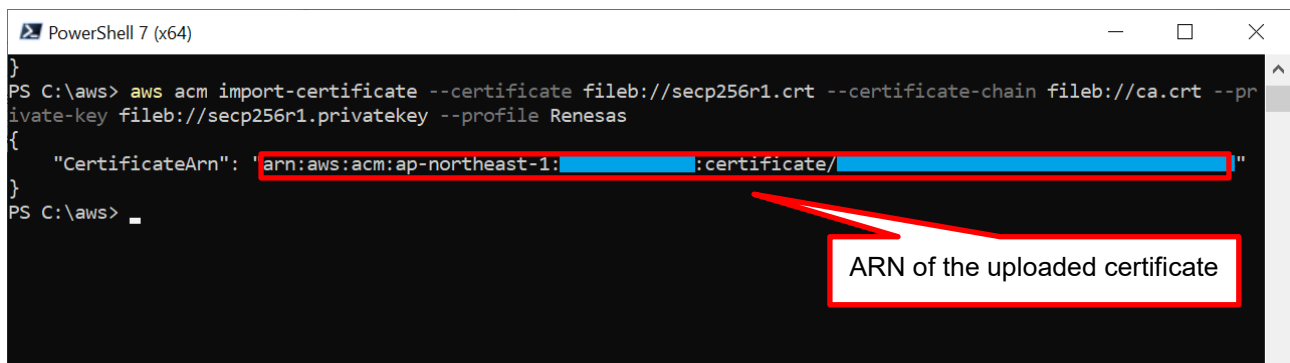
- ECDSA certificate: secp256r1.crt
- ECDSA private key: secp256r1.privatekey
- ECDSA certificate chain: ca.crt

(2) Import a new code signing certificate.

Import the files prepared in step (1) for the code signing certificate to AWS.
Enter the following command.

```
> aws acm import-certificate --certificate fileb://secp256r1.crt --  
certificate-chain fileb://ca.crt --private-key fileb://secp256r1.privatekey --  
profile <PROFILE_NAME>
```

After the command has been executed, the display in the window will be as shown below.



Make a note of the ARN of the code signing certificate (CertificateArn: text inside the red frame in the above figure) which is displayed. You will need it in a later process.

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(3) Create a profile for the code signing certificate.

(a) Create a json file containing a profile.

Create a configuration file of the code signing certificate profile.

With a text editor, create a file with the name "profile.json" in the work folder that was created in 3.1.5 and enter the following code.

- profile.json

```
{
  "profileName": "rx65n_ota_demo_profile",
  "signingMaterial": {
    "certificateArn": "arn:aws:acm:ap-northeast-1:xxxxxxx:certificate/yyyyyyy-yyy-yyyy-yyyy-yyyyyyyyyy"
  },
  "platformId": "AmazonFreeRTOS-Default",
  "signingParameters": {
    "certname": "dummy"
  }
}
```

Enter the following information into the fields underlined in red in the above json file.

Item	Description	Contents to be Set
profileName	Profile name	A desired profile name can be entered (example: rx65n_ota_demo_profile).
certificateArn	ARN of the certificate	Enter the ARN of the code signing certificate that was imported in step (2).

Make a note of the profile name (profileName) that was set here. You will need it in a later process.

(b) Create a code signing certificate profile.

Create a profile using the json file that was created in step (a).

Enter the following command.

```
> aws signer put-signing-profile --cli-input-json file://profile.json --profile <PROFILE_NAME>
```

After the command has been executed, the display in the window will be as shown below.

```
PowerShell 7 (x64)
PS C:\aws> aws signer put-signing-profile --cli-input-json file://profile.json --profile Renesas
{
  "arn": "arn:aws:signer:ap-northeast-1:xxxxxxx:/signing-profiles/rx65n_ota_demo_profile",
  "profileVersion": "1",
  "profileVersionArn": "arn:aws:signer:ap-northeast-1:xxxxxxx:/signing-profiles/rx65n_ota_demo_profile/1"
}
PS C:\aws>
```

5.2.3 Executing an OTA Job

Create an OTA job and execute an OTA update.

- (1) Create a json file in which details of the OTA job are set.

Create a configuration file of the OTA job.

With a text editor, create a file with the name "create-ota-update.json" in the work folder that was created in 3.1.5 and enter the following code.

- create-ota-update.json

```
{
  "otaUpdateId": "rx65n ota demo job",
  "targets": [
    "arn:aws:iot:ap-northeast-1:xxxxxxxxxxxx:thing/rx65n ota demo thing"
  ],
  "protocols": [
    "MQTT"
  ],
  "targetSelection": "SNAPSHOT",
  "awsJobExecutionsRolloutConfig": {
    "maximumPerMinute": 1000
  },
  "files": [
    {
      "fileName": "/device/updates",
      "fileLocation": {
        "s3Location": {
          "bucket": "s3test-rx65n",
          "key": "user_093.rsu",
          "version": "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
        }
      },
      "codeSigning": {
        "startSigningJobParameter": {
          "destination": {
            "s3Destination": {
              "bucket": "s3test-rx65n",
              "prefix": "SignedImages/"
            }
          },
          "signingProfileName": "rx65n ota demo profile"
        }
      }
    }
  ],
  "roleArn": "arn:aws:iam:xxxxxxxxxxxx:role/ota_role_rx65n"
}
```

Enter the following information into the fields underlined in red in the above json file.

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

Item	Description	Contents to be Set
otaUpdateId	OTA update ID	This is the name of the OTA job to be executed. Enter a desired job name (example: rx65n_ota_demo_job). Note: The name set must be different from any job names used in the past because the duplication of an OTA job name is not allowed.
targets	Target ARN	Enter the ARN of the thing (device) for which the OTA update is to be executed. Enter the ARN of the thing that was set in 3.3.4(1).
bucket (files→fileLocation→s3Location)	S3 bucket	Enter the bucket name to which the update firmware was uploaded. Enter the bucket name that was set in 3.3.2(1).
key (files→fileLocation→s3Location)	S3 key	Specify the file name of the update firmware. Enter the name of the file that was uploaded in 5.2.1(2).
version (files→fileLocation→s3Location)	Version of the S3 bucket	Enter the version ID of the update firmware. Enter the version ID of the uploaded file that was confirmed in 5.2.1(3).
bucket (files→codeSigning→startSigningJobParameter→destination→s3Destination)	Bucket used in code signing	Enter the S3 bucket name for which code signing is performed. Enter the bucket name that was set in 3.3.2(1).
signingProfileName (files→codeSigning→startSigningJobParameter)	Profile name used in code signing	Enter the profile name used in code signing. Enter the profile name that was set in 5.2.2(3)(a).
roleArn	Role ARN	Enter the ARN of the IAM role used in the OTA update. Enter the ARN of the IAM role that was created in 3.3.3(2).

Make a note of the OTA update ID (otaUpdateId) that was set here. You will need it in a later process.

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(2) Execute the OTA job.

Before running the job, execute the initial firmware in the CK-RX65N v2 board and confirm that it reaches the state of waiting for the execution of an OTA job (see 4.2.4(11)).

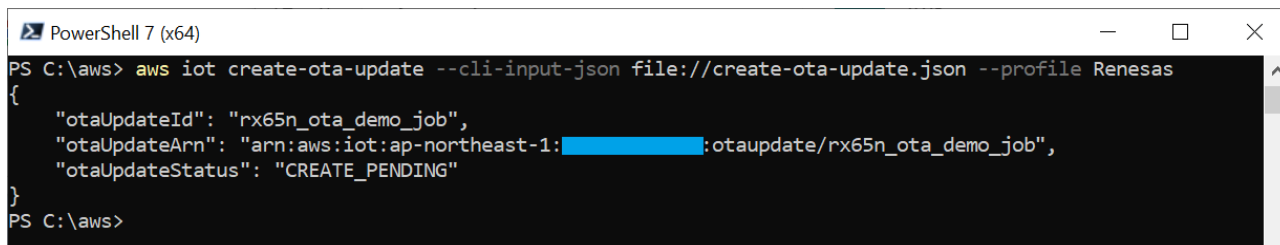
Execute the OTA job by using the json file that was created in step (1).

Enter the following command.

```
> aws iot create-ota-update --cli-input-json file://create-ota-update.json --profile <PROFILE_NAME>
```

After the command has been executed, the display in the window will be as shown below.

If execution of the job proceeded normally, downloading of the update firmware to the target board will start.



```
PS C:\aws> aws iot create-ota-update --cli-input-json file://create-ota-update.json --profile Renesas
{
  "otaUpdateId": "rx65n_ota_demo_job",
  "otaUpdateArn": "arn:aws:iot:ap-northeast-1:[redacted]:otaupdate/rx65n_ota_demo_job",
  "otaUpdateStatus": "CREATE_PENDING"
}
PS C:\aws>
```

(3) Confirm the execution status of the OTA job.

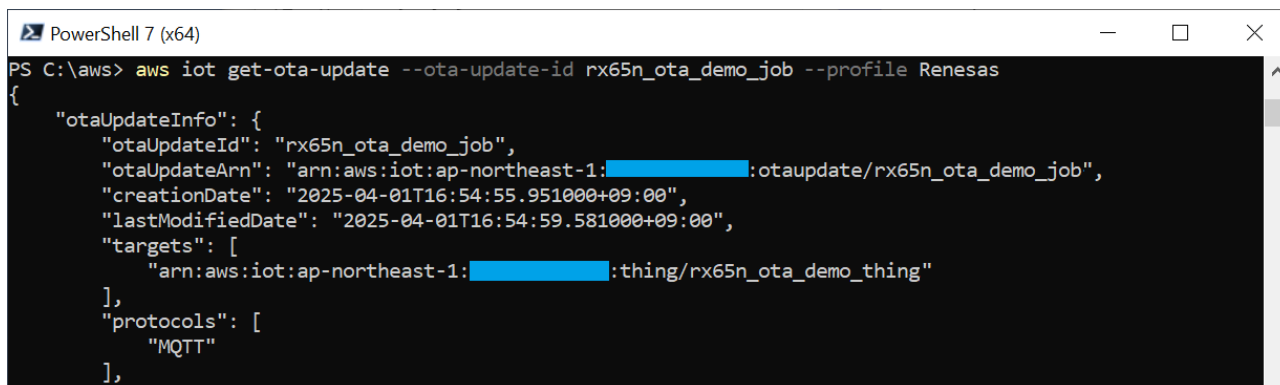
Confirm the status after the OTA job has been executed.

Enter the following command.

```
> aws iot get-ota-update --ota-update-id <JOB_NAME> --profile <PROFILE_NAME>
```

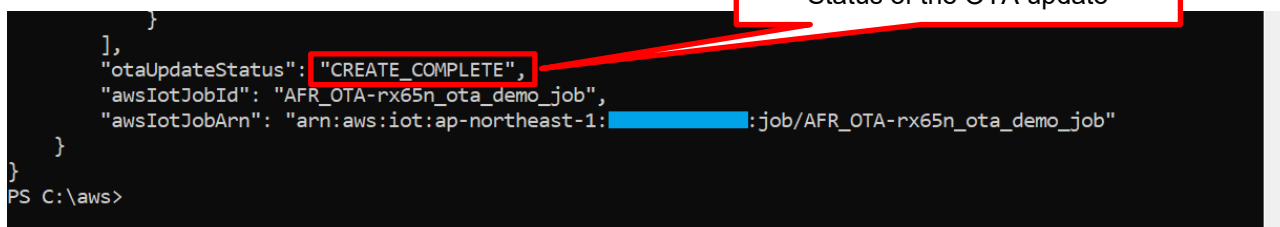
- Enter the OTA job name (OTA update ID) that was set in step (1) into <JOB_NAME>.

After the command has been executed, the display in the window will be as shown below.



```
PS C:\aws> aws iot get-ota-update --ota-update-id rx65n_ota_demo_job --profile Renesas
{
  "otaUpdateInfo": {
    "otaUpdateId": "rx65n_ota_demo_job",
    "otaUpdateArn": "arn:aws:iot:ap-northeast-1:[redacted]:otaupdate/rx65n_ota_demo_job",
    "creationDate": "2025-04-01T16:54:55.951000+09:00",
    "lastModifiedDate": "2025-04-01T16:54:59.581000+09:00",
    "targets": [
      "arn:aws:iot:ap-northeast-1:[redacted]:thing/rx65n_ota_demo_thing"
    ],
    "protocols": [
      "MQTT"
    ]
  },
  "otaUpdateStatus": "CREATE_PENDING",
  "awsIotJobId": "AFR_OTA-rx65n_ota_demo_job",
  "awsIotJobArn": "arn:aws:iot:ap-northeast-1:[redacted]:job/AFR_OTA-rx65n_ota_demo_job"
}
PS C:\aws>
```

[Partially omitted]



```
    ],
    "otaUpdateStatus": "CREATE_COMPLETE",
    "awsIotJobId": "AFR_OTA-rx65n_ota_demo_job",
    "awsIotJobArn": "arn:aws:iot:ap-northeast-1:[redacted]:job/AFR_OTA-rx65n_ota_demo_job"
  }
}
PS C:\aws>
```

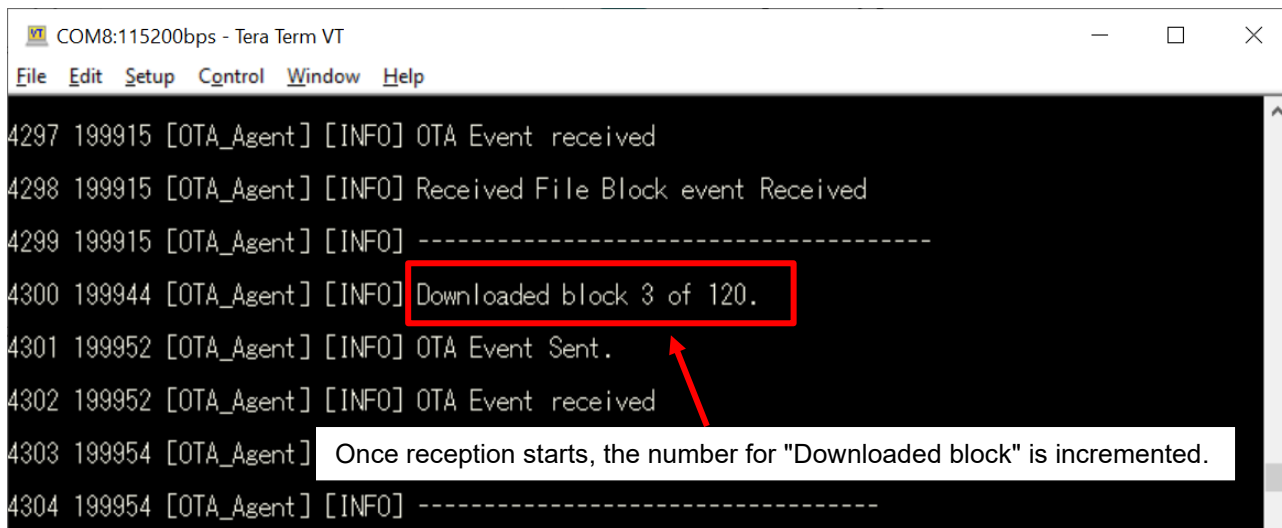
If "CREATE_COMPLETE" is displayed as the status of the OTA update (otaUpdateStatus: text inside the red frame in the above figure), the OTA job was executed successfully.

If the OTA job ends in an error due to an incorrect setting or some other factor, a message such as "CREATE_FAILED" will be displayed.

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

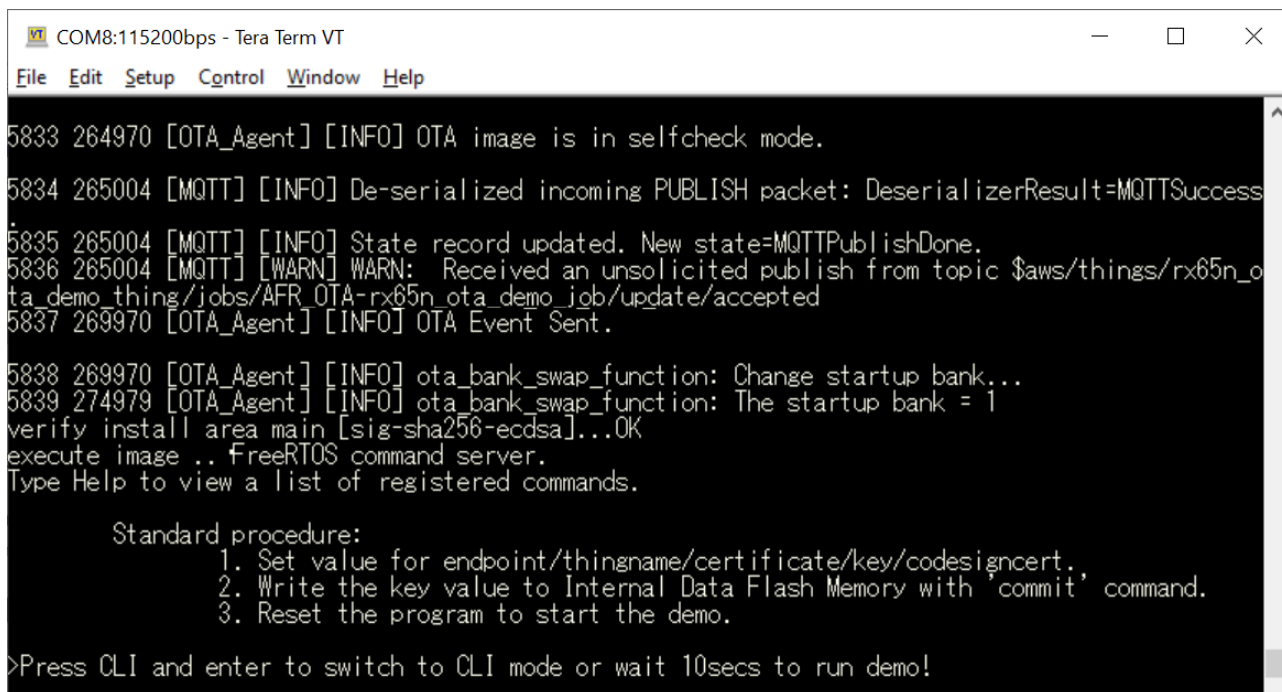
(4) Wait until the reception of the firmware is completed.

When the OTA job is started, the reception and writing of the firmware proceed.



```
COM8:115200bps - Tera Term VT
File Edit Setup Control Window Help
4297 199915 [OTA_Agent] [INFO] OTA Event received
4298 199915 [OTA_Agent] [INFO] Received File Block event Received
4299 199915 [OTA_Agent] [INFO] -----
4300 199944 [OTA_Agent] [INFO] Downloaded block 3 of 120.
4301 199952 [OTA_Agent] [INFO] OTA Event Sent.
4302 199952 [OTA_Agent] [INFO] OTA Event received
4303 199954 [OTA_Agent] [INFO] Once reception starts, the number for "Downloaded block" is incremented.
4304 199954 [OTA_Agent] [INFO] -----
```

When updating is completed and code signing verification has succeeded, the update firmware is executed following a CPU reset, and the first messages are displayed as shown below.



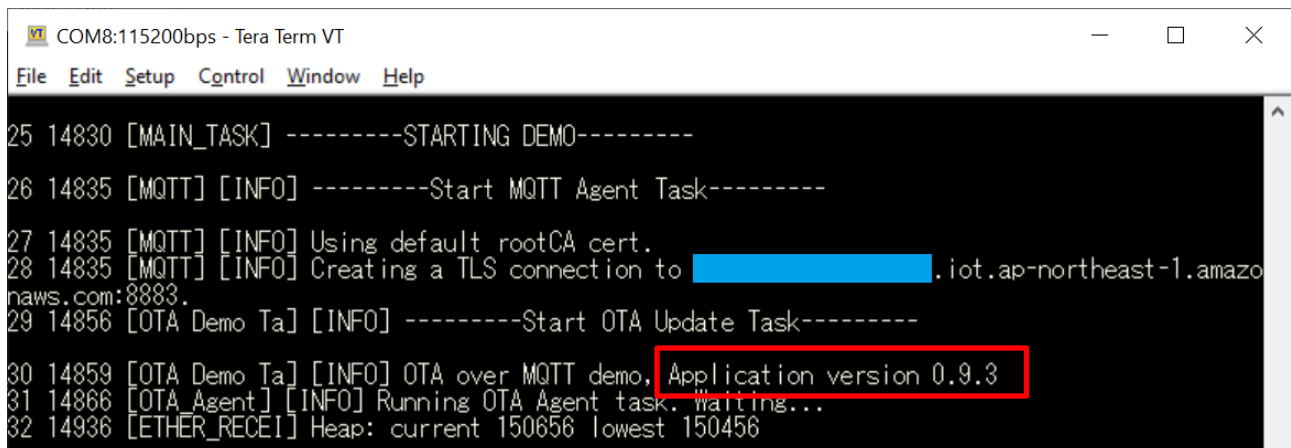
```
COM8:115200bps - Tera Term VT
File Edit Setup Control Window Help
5833 264970 [OTA_Agent] [INFO] OTA image is in selfcheck mode.
5834 265004 [MQTT] [INFO] De-serialized incoming PUBLISH packet: DeserializerResult=MQTTSuccess
5835 265004 [MQTT] [INFO] State record updated. New state=MQTTPublishDone.
5836 265004 [MQTT] [WARN] WARN: Received an unsolicited publish from topic $aws/things/rx65n_ota_demo_thing/jobs/AFR_OTA-rx65n_ota_demo_job/update/accepted
5837 269970 [OTA_Agent] [INFO] OTA Event Sent.
5838 269970 [OTA_Agent] [INFO] ota_bank_swap_function: Change startup bank...
5839 274979 [OTA_Agent] [INFO] ota_bank_swap_function: The startup bank = 1
verify install area main [sig-sha256-ecdsa]...OK
execute image .. FreeRTOS command server.
Type Help to view a list of registered commands.

Standard procedure:
1. Set value for endpoint/thingname/certificate/key/codesigncert.
2. Write the key value to Internal Data Flash Memory with 'commit' command.
3. Reset the program to start the demo.

>Press CLI and enter to switch to CLI mode or wait 10secs to run demo!
```

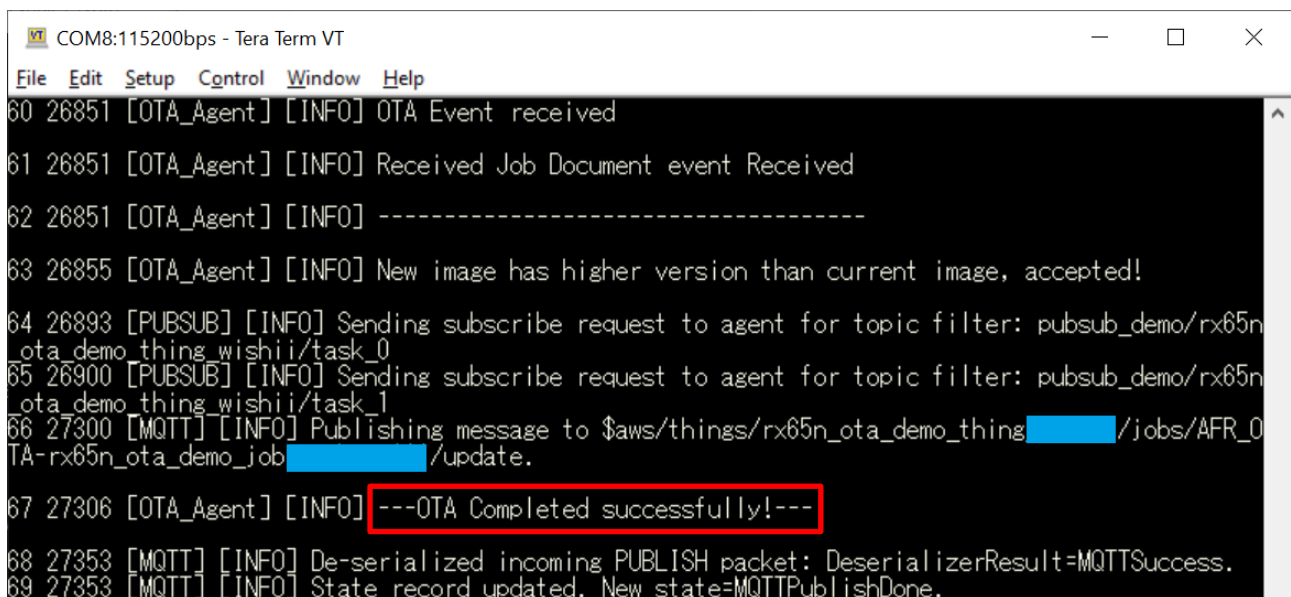
RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

(5) Confirm that the firmware version has become “version 0.9.3” after the reset.



```
COM8:115200bps - Tera Term VT
File Edit Setup Control Window Help
25 14830 [MAIN_TASK] -----STARTING DEMO-----
26 14835 [MQTT] [INFO] -----Start MQTT Agent Task-----
27 14835 [MQTT] [INFO] Using default rootCA cert.
28 14835 [MQTT] [INFO] Creating a TLS connection to [REDACTED].iot.ap-northeast-1.amazonaws.com:8883.
29 14856 [OTA Demo Ta] [INFO] -----Start OTA Update Task-----
30 14859 [OTA Demo Ta] [INFO] OTA over MQTT demo, Application version 0.9.3
31 14866 [OTA Agent] [INFO] Running OTA Agent task. Waiting...
32 14936 [ETHER_RECEI] Heap: current 150656 lowest 150456
```

When the OTA job is completed normally, “OTA Completed successfully!” is displayed as shown below.



```
COM8:115200bps - Tera Term VT
File Edit Setup Control Window Help
60 26851 [OTA_Agent] [INFO] OTA Event received
61 26851 [OTA_Agent] [INFO] Received Job Document event Received
62 26851 [OTA_Agent] [INFO] -----
63 26855 [OTA_Agent] [INFO] New image has higher version than current image, accepted!
64 26893 [PUBSUB] [INFO] Sending subscribe request to agent for topic filter: pubsub_demo/rx65n_ota_demo_thing_wishii/task_0
65 26900 [PUBSUB] [INFO] Sending subscribe request to agent for topic filter: pubsub_demo/rx65n_ota_demo_thing_wishii/task_1
66 27300 [MQTT] [INFO] Publishing message to $aws/things/rx65n_ota_demo_thing[REDACTED]/jobs/AFR_0TA-rx65n_ota_demo_job[REDACTED]/update.
67 27306 [OTA_Agent] [INFO] ---OTA Completed successfully!---
68 27353 [MQTT] [INFO] De-serialized incoming PUBLISH packet: DeserializerResult=MQTTSuccess.
69 27353 [MQTT] [INFO] State record updated. New state=MQTTPublishDone.
```

RX Family How to Implement FreeRTOS OTA Using Amazon Web Services (202406-LTS Version)

6. Appendix

6.1 Points to Keep in Mind when Operating Multiple Devices within the Same LAN Environment

Addresses assigned from the vendor ID of Renesas Electronics Corporation are used as the MAC addresses in the sample code.

When using the sample program to operate multiple devices at the same time within the same LAN environment, the MAC addresses must be changed to avoid duplication.

The sample program may not operate properly if the same MAC addresses are duplicated among multiple devices.

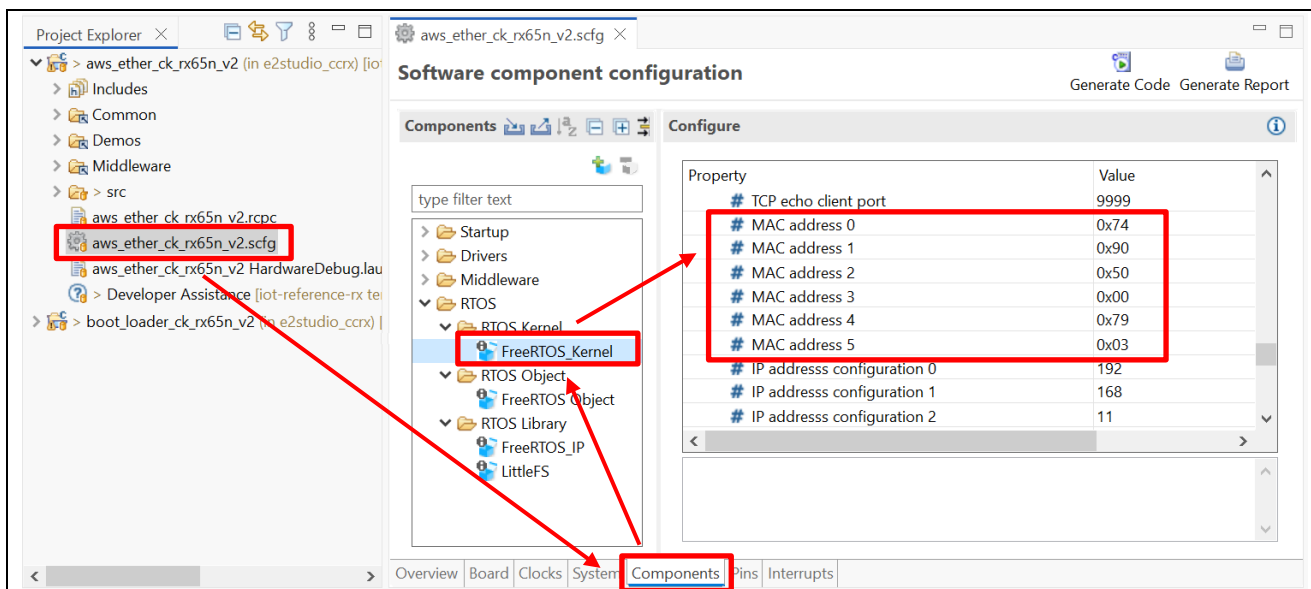
The procedure for changing the MAC addresses is described below.

Open “aws_ether_ck_rx65n_v2.scfg” in the Smart Configurator, and select the [Components] tab.

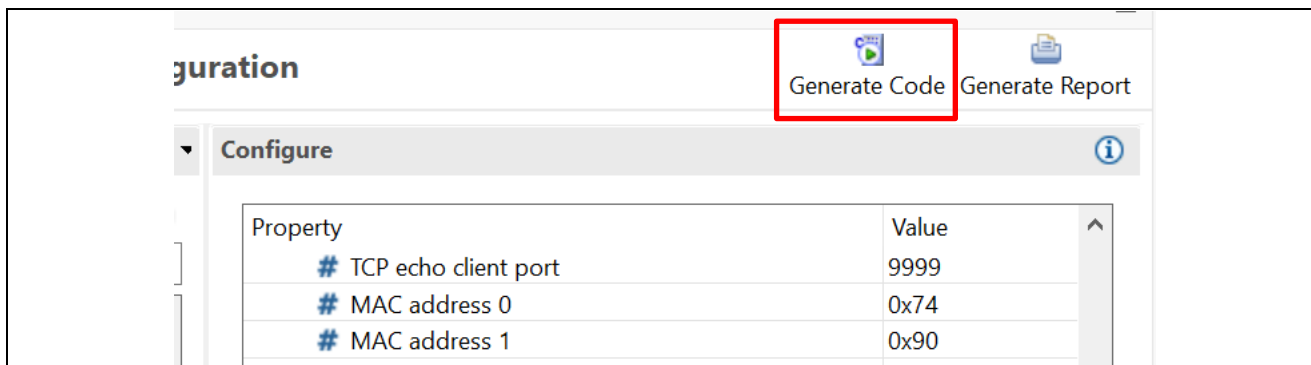
In the tree, select [RTOS] → [RTOS Kernel] → [FreeRTOS_Kernel], then under [Property] set hexadecimal values of your choice as “Value” for “MAC address 0” to “MAC address 5”.

Enter values in the format 0xXX (where XX represents a hexadecimal value of your choice).

When creating commercial products for sale, be sure to use MAC addresses that have been submitted to the IEEE.



After making the above changes, click the [Generate Code] button in the upper-right corner of the window to have the code reflect the changes made in the Smart Configurator.



7. Troubleshooting

The following table lists problems that might arise when executing the sample program, and how to resolve them.

No.	Issue	Reason	Solution	Refer to
1	The command to create the initial firmware fails.	The Python installation folder is not set correctly in the path of the environment variable.	Reinstall Python. Also, make sure that the [Add python.exe to PATH] checkbox is selected when performing step 2.2(3).	2.2
2		The encryption library is not installed.	Install the encryption library.	2.2(5)
3	The initial firmware cannot be written.	J16 on the CK-RX65N v2 board is not set to the DEBUG side.	Make sure that pins 1-2 of J16 on the CK-RX65N v2 board are closed (DEBUG).	2.6
4	The initial firmware does not start.	J16 on the CK-RX65N v2 board is not set to the RUN side.	Make sure that pins 2-3 of J16 on the CK-RX65N v2 board are closed (RUN).	4.2.4(4)
5	Cellular communication cannot start.	The RYZ014A PMOD module is not connected properly.	Check the connection of the RYZ014A PMOD module.	2.6
6		A SIM card is not inserted.	Insert a SIM card.	2.6
7		The SIM card is configured incorrectly.	Review the configuration settings for the r_cellular module.	4.2.2(4)
8		The SIM card supplied with the CK-RX65N v1 board is in use but is not activated.	Activate the SIM card.	4.2.2(4)
9	An error occurs during cellular communications.	The communications environment is poor.	Connect an antenna and power supply to the RYZ014A PMOD module. Also, place the antenna in a location with good reception, such as near a window.	2.6
10	An error occurs when connecting to AWS.	The AWS IoT information is not set or is set incorrectly.	Set the AWS IoT information again.	4.2.5
11	The firmware does not start after the boot loader is started.	The public key is not correctly set in the boot loader.	Review the public key setting in the boot loader.	4.2.2(1)
		The version ID of the firmware is registered incorrectly.	Confirm the version of the uploaded firmware.	5.2.1(3)
12	The firmware does not start after an OTA update.	The project is not set correctly.	Review the settings of the firmware and boot loader.	4.2.2 4.2.4(1)
13	An error occurs when an OTA job is re-executed.	The OTA job name is a duplicate.	OTA job names cannot be reused. Set a different name.	5.2.3(1)

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Jun.20.25	—	First edition issued.
1.10	Mar.10.26	p.1 p.16 p.55	Additional information regarding Wi-Fi functionality has been included.
		p.5	Update the environment to the latest version. (e2studio/GCC/FreeRTOS/Tera Term/OpenSSL)
		p.6-7	Revisions to the instructions due to the Tera Term version update.
		p.9-10	Revisions to the instructions due to the OpenSSL version update.
		p.12 p.14 p.62	Addition of a note.
		p.39-51	The procedure for creating a project has been revised to the procedure for generating a project.
		p.58-59	Revised the explanation regarding the placement of the flash project.
		p.60-61	Revised the instructions based on the Tera Term v5 interface.
		p.63-65	Added checkbox operation when dropping files into Tera Term.

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.
7. 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 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 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.
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/.