

CK-RX65N

SIM activation, Creating the trial account and using Dashboard with RYZ014A or Ethernet Application for AWS - Getting Started Guide

Introduction

This document describes a system that uses the CK-RX65N V1 Cloud Kit board from Renesas. This system incorporates the CK-RX65N V1 running Amazon FreeRTOS and via Ethernet/Cellular connection. It visualizes HS3001, ZMOD4410, ZMOD4510, OB1203, ICP10101 and ICM20948 sensor information on Amazon Web Services (AWS) and controls LEDs on the board. In addition, this application note also describes several feature options for users when using CK-RX65N V1 Cloud Kit with AWS: OTA (Over-The-Air) feature (section 6) and Fleet Provisioning feature (section 7).

This document shows two methods of connectivity for CK-RX65N V1. The first one is the Ethernet, and the second is the Cellular CAT M1 using RYZ014A.

In addition, this document describes the following:

- How to activate the SIM card that is contained with the CK-RX65N V1.
- How to create the 10 USD free trial account for AWS.
- How to operate and install the information of certification for cloud.
- How to see and run the sensor data on the dashboard.
- How to use OTA feature to update firmware via Cloud.
- How to use Fleet Provisioning via Cloud.
- Note: Renesas announced the discontinuation of the Sequans-sourced LTE module, part number RYZ014A, and will no longer be shipping this product. If you have this in a current design or production, the Sequans part numbers, GM01Q is a pin and functionally compatible replacement for RYZ014A. Below is the cellular driver alternate product.

- RYZ014A Cellular control module: Sequans GM01Q is the compatible module. Regarding EOL notice of the RYZ014A, please see:

[The link] <u>https://www.renesas.com/document/eln/plc-240004-end-life-eol-process-select-part-numbers</u> [The product page] <u>https://www.renesas.com/products/wireless-connectivity/cellular-iot-</u> modules/ryz014a-lte-cat-m1-cellular-iot-module



Figure 1. CK-RX65N V1 (with RYZ014A Pmod)



Contents

1.	Terms	4
2.	Preparation	4
2.1	Hardware Configuration	4
2.2	Software Configuration	
2.3	Tera term Setting	5
0		-
3.	System Diagram	5
4.	Cloud Connectivity Application Example	6
4.1	Overview	6
4.2	MQTT/TLS Application Software Overview	8
5.	Connection to AWS	11
5.1	Hardware Preparation and Import the Project	11
5.1.1	Hardware Preparation	11
5.1.2	Connecting the Board to the Serial Port Console of the PC	11
5.1.3	Importing the Project	13
5.1.4	Running the Application Project	26
5.1.5	Activating SIM Card	28
5.2	For Users Using the Provided Dashboard and AWS Account of Kit	30
5.2.1	Getting the Board UUID Information	30
5.2.2	Getting the Account 10 USD for Trial of AWS	30
5.3	Software Preparation - Running Project from IDE	33
5.3.1	Storing the Device Certificate, Key, MQTT Broker Endpoint and IoT Thing Name	33
5.3.2	Starting the Application	37
5.4	For Users Using Their Own AWS Account	39
5.4.1	Get an AWS Account	39
5.4.2	Log in to the AWS Management Console	39
5.4.3	Move to IoT Core Control Panel	39
5.4.4	Create a Security Policy	40
5.4.5	Register your device (thing) with AWS IoT	42
5.4.6	Check AWS IoT Endpoint	45
5.4.7	Running Application	45
5.5	Verifying the Application Project using AWS Dashboard and Renesas Dashboard	46
5.5.1		
5.5.2	Publish a Topic Messages on the AWS Dashboard and Renesas Dashboard	47
6.	OTA over MQTT	51
6.1	Overview OTA	51
6.2	Prerequisites	51



CK-RX65N SIM activation, Creating the trial account and using Dashboard with RYZ014A or Ethernet Application for AWS - Getting Started Guide

6.2.1	Installing Python	51
6.2.2	Installing OpenSSL	53
6.2.3	Installing Renesas Image Generator	53
6.3	Setting up AWS for OTA	54
6.3.1	Register your Device in AWS	54
6.3.2	Creating an Amazon S3 bucket	54
6.3.3	Allocating OTA execution permission to IAM users	57
6.4	Setting up the Device	63
6.4.1	Generating Key Pairs and Certificates	63
6.4.2	Setting up the project	65
6.5	Updating the Firmware	79
6.5.1	Creating the updated firmware	79
6.5.2	Updating the firmware	80
7		
	Fleet Provisioning	
7.1	Overview Fleet Provisioning	
7.2	Setting up AWS for Fleet Provisioning	
7.2.1	Policy Settings	
7.2.2		
7.2.3	5 5 1	
7.3	Setting up the Project	
7.4	Running Fleet Provisioning	
8.	Note and Troubleshooting	
8.1	Sensor Stabilization Time	
8.2	Connection Issue When Using Ethernet (Wired cable)	
8.3	Current Supply Short Issue When Using RYZ014A	107
8.4	When Build Errors Occur	107
8.5	When Unable to Log in to the Dashboard (Grafana Account)	107
8.6	Notes on Performing Firmware Update Over-The-Air on AWS FreeRTOS	
8.7	When the Trial 10 USD is Used Up	107
8.8	How to Enable/Disable EC2 Instance	
8.9	How to check the total amount spent in AWS account	
8.10	An error occurs when connecting to AWS	
8.11	Command to create the initial firmware fails (OTA)	
8.12	Initial firmware cannot be written/ does not start. (OTA)	
8.13	Firmware does not start after starting the boot loader (OTA)	
8.14	Firmware does not start after an OTA update (OTA)	
9.	Website and Support	
Revie	sion History	11/
1.001		



1. Terms

Terms used in this document are explained below.

Table 1. Terms

Term	Meaning
AWS	AWS Amazon Web Service
Pmod	Peripheral Module
MQTT	Message Queuing Telemetry Transport
ΟΤΑ	Over-The-Air
TLS	Transport Layer Security
UUID	Unique ID for each kit

2. Preparation

2.1 Hardware Configuration

The hardware configuration of the demo project is listed in the table below. **Table 2. Hardware Configuration**

Item	Content	Description
CK-RX65N V1 Cloud Kit	Target board for CK-RX65N V1	Please see detail at: https://www.renesas.com/rx/ck-rx65n
RYZ014A Cellular Pmod module	SIM card	This Pmod is contained with CK- RX65N V1 kit with SIM card
PC	Windows [®] 10 Google Chrome / Microsoft Edge	Recommended OS. Web browser used.

2.2 Software Configuration

The software configuration of the demo project is listed in the table below.

Table 3. Software Configuration

Item	Content	Version
Integrated development environment	e2 studio (<u>e² studio Renesas</u>)	2024-01
Compiler	CC-RX (<u>CC-RX Compiler</u>)	V3.05
Communication Software	Tera term (<u>Tera Term -</u> <u>Download (softonic.com)</u>)	Version 4.99
Emulator	E2 emulator Lite (on-board)	-
RTOS	AWS FreeRTOS	V202210.01
Python	(Please see detail at: 6.2.1)	V3.11.0 or later
Keygen tool	Win64 OpenSSL (Please see detail at: 6.2.2)	V3.0.12
Flash programming tool	Renesas Flash Programmer (Renesas Flash Programmer (Programming GUI) Renesas)	V3.12.00
Renesas Image Generator	Supplied with Firmware Update module Rev.2.01 (Please see detail at: 6.2.3)	V3.02



2.3 Tera term Setting

Table 4. Tera term Setting

Item	Settings
Baud rate	115200
Data length	8
Parity	None
Stop bits	1
Flow Control	None

3. System Diagram



Figure 2. System Diagram

Note: The 9-AXIS MEMS Motion Tracking Sensor TDK ICM-20948 is not fitted on this CK-RX65N board due to shortages in component availability (For more information: <u>CK-RX65N v1 – Release Note</u> (renesas.com))



4. Cloud Connectivity Application Example

4.1 Overview

This application project demonstrates the use of Driver, Middleware and RTOS components, FIT configurator on Renesas RX65N MCU to establish AWS Cloud connectivity using Ethernet/Cellular. It illustrates how the cloud service provider is configured and operated.

This documentation illustrates Subscribe and Publish communications between MQTT Client and MQTT Broker, on-demand publication of sensor data, and asynchronous publication of a "sensor data" event from the MCU to the Cloud.



Figure 3. MQTT Publish/Subscribe to/from AWS IoT Core



Figure 4. Thread Diagram



Application also supports:

• OTA over MQTT feature for updating new firmware (please refer to the section 6. OTA over MQTT):



Figure 5. Thread Diagram when enabling OTA feature

Fleet Provisioning (please refer to the section 7. Fleet Provisioning):



Figure 6. Thread Diagram when enabling Fleet feature



4.2 MQTT/TLS Application Software Overview

The following files from these application projects serve as a reference as shown in Table 5.

Table 5. Application Project File

No.	Filename	Purpose
1.	<pre>src/application_code/main.c</pre>	Contains initialization code of the connection, provisioning cloud credentials used in Cloud Connectivity, main function of application.
2.	<pre>src/application_code/user_init.c</pre>	Contains initialization functions.
3.	<pre>src/application_code/CommandLine/ cli_thread_entry.c</pre>	Contains data structures and functions used in CLI thread.
4.	<pre>src/application_code/CommandLine/c ommon_init.h</pre>	Contains macros, data structures, and functions prototypes used to initialize common in the project.
5.	<pre>src/application_code/CommandLine/c onsole.c</pre>	Contains data structures and functions used to print data on console using UART
6.	<pre>src/application_code/CommandLine/c onsole.h</pre>	Contains the function prototypes used to print data on console using UART
7.	<pre>src/application_code/CommandLine/m enu_flash.c</pre>	Contains data structures and functions used to provide CLI flash memory related menu
8.	<pre>src/application_code/CommandLine/m enu_flash.h</pre>	Contains the function prototypes and macros used to provide CLI flash memory related menu
9.	<pre>src/application_code/CommandLine/m enu_kis.c</pre>	Contains functions to get the application's version, get UUID and help option for main menu on CLI
10.	<pre>src/application_code/CommandLine/m enu_kis.h</pre>	Contains the function prototypes and macros used to get application's version, get UUID and help option for main menu on CLI
11.	<pre>src/application_code/CommandLine/m enu_main.c</pre>	Contains data structures and functions used to provide CLI main menu options
12.	<pre>src/application_code/CommandLine/m enu_main.h</pre>	Contains the function prototypes and macros used to provide CLI main menu options
13.	<pre>src/application_code/CommandLine/c ommon_utils.h</pre>	Contains macros, data structures, and functions prototypes commonly used across the project.
14.	<pre>src/application_code/CommandLine/r _typedefs.h</pre>	Contains typedefs used in application
15.	<pre>src/application_code/sensor_thread _entry.c</pre>	Contains the code for sensor thread (HS3001, ICP10101 and ICM20948)
16.	<pre>src/application_code/ICM20948/icm2 0948.c</pre>	Contains the code for the 9-Axis MEMS Motion Tracking™ Sensor
17.	<pre>src/application_code/ICM20948/icm2 0948.h</pre>	Contains the Data structure function prototypes for the 9-Axis MEMS Motion Tracking™ Sensor
18.	<pre>src/application_code/ICM20948/icm_ i2c.c</pre>	Contains the I2C code to communicate with 9- Axis MEMS Motion Tracking [™] Sensor
19.	<pre>src/application_code/ICM20948/icm_ i2c.h</pre>	Contains the I2C function prototypes to communicate with 9-Axis MEMS Motion Tracking [™] Sensor
20.	<pre>src/application_code/ICP10101/ICP_ 20100.c</pre>	Contains the code for Barometric Pressure and Temperature Sensor



CK-RX65N	SIM activation, Creating the trial account and using Dashboard with RYZ014A or Ethernet
	Application for AWS - Getting Started Guide

No.	Filename	Purpose
21.	<pre>src/application_code/ICP10101/ICP_ 20100.h</pre>	Contains the data structure and function prototypes for Barometric Pressure and Temperature Sensor
22.	<pre>src/application_code/OB1203/RX_OB1 203.c</pre>	Contains data structures and functions used for the oximeter sensor
23.	<pre>src/application_code/OB1203/ob1203 _bio.c</pre>	Contains the Data structure for the oximeter sensor
24.	src/application_code/OB1203/ob1203 _bio_rx.c	Contains data structures and functions used for the oximeter sensor
25.	src/application_code/OB1203/ob1203	Contains the Data structure and function prototypes for the oximeter sensor
26.	src/application_code/OB1203/KALMAN /kalman.c	Contains algorithm for Heart Rate, Blood Oxygen Concentration, Pulse Oximetry,
27.	<pre>src/application_code/OB1203/KALMAN /kalman.h</pre>	Proximity, Light and Color Sensor sample calculations
28.	<pre>src/application_code/OB1203/SAVGOL /SAVGOL.c</pre>	
29.	<pre>src/application_code/OB1203/SAVGOL /SAVGOL.h</pre>	
30.	src/application_code/OB1203/SPO2/S PO2.c	
31.	src/application_code/OB1203/SPO2/S PO2.c	
32.	<pre>src/application_code/HS3001/RX_HS3 001.c</pre>	Contains the code and function for Renesas Relative Humidity and Temperature Sensor.
33.	<pre>src/application_code/HS3001/RX_HS3 001.h</pre>	Contains the common data structure's function prototypes for the Renesas Relative Humidity and Temperature sensors.
34.	src/application_code/ZMOD4x10/RX_Z MOD4XXX_Common.c	Contains the common code for the Renesas ZMOD sensors
35.	src/application_code/ZMOD4x10/RX_Z MOD4XXX_Common.h	Contains the common data structure's function prototypes for the Renesas ZMOD sensors
36.	src/application_code/ZMOD4x10/RX_Z MOD4XXX_IAQ1stGen.c	Contains the common code for the Renesas ZMOD Internal Air Quality sensors
37.	src/application_code/ZMOD4x10/RX_Z MOD4XXX_OAQ1stGen.c	Contains the common code for the Renesas ZMOD Outer Air Quality sensors
38.	<pre>src/application_code/frtos_skeleto n/obl203_thread.c</pre>	Contains the OB1203 sensor thread (for oximeter sensor)
39.	<pre>src/application_code/frtos_skeleto n/sensor_thread.c</pre>	Contains the sensor's thread (for Renesas Relative Humidity and Temperature Sensor, Barometric Pressure and Temperature Sensor and the 9-Axis MEMS Motion Tracking [™] Sensor)
40.	<pre>src/application_code/frtos_skeleto n/zmod_thread.c</pre>	Contains the ZMOD's thread (for Renesas ZMOD Internal Air Quality sensors)
41.	<pre>src/application_code/frtos_skeleto n/task_function.h</pre>	Contains the common data structure's function prototypes for thread
42.	<pre>src/application_code/frtos_startup /freertos_object_init.c</pre>	Contains the source code for FreeRTOS thread
43.	<pre>src/application_code/frtos_startup /freertos_start.c</pre>	Contains FreeRTOS user-defined functions



CK-RX65N SIM activation, Creating the trial account and using Dashboard with RYZ014A or Ethernet Application for AWS - Getting Started Guide

No.	Filename	Purpose
44.	<pre>src/application_code/frtos_startup /freertos_start.h</pre>	FreeRTOS's user-defined functions header file
45.	<pre>src/application_code/frtos_config/ *.h</pre>	Contains FreeRTOS configuration header file.
46.	<pre>src/application_code/sensorsData.h</pre>	Contains the common data structure's function prototypes for sensors
47.	Demos/SimplePubSub/simple_pub_sub_ task.c	Contains code and functions used in MQTT interface for Cloud Connectivity.
48.	Demos/mqtt_agent/mqtt_agent_task.c	Contains the code for running the MQTT task
49.	Demos/OtaOverMqtt/OtaOverMqttDemoE xample.c	Contains function for running OTA over MQTT
50.	Demos/Fleet_Provisioning_With_CSR_ Demo	Contains function for running Fleet Provisioning
51.	Demos/cli/serial.c	Contains function for serial communication.
52.	Demos/cli/serial.h	Contains the common data structure's function prototypes for serial.c
53.	Demos/include/*.h	Contains the common data structure's function prototypes for demo function.

Note: The above table only lists some important files in applications.



5. Connection to AWS

AWS account is necessary to connect CK-RX65N V1 Cloud Kit to AWS.

Note: Renesas provides 10 USD of AWS account credit to users who buy the CK-RX65N V1 and this 10 USD credit cannot be used for an existing account.

This document covers two ways of connecting to the AWS account:

- Case 1: For users who want to use trial AWS account with 10 USD credits and Renesas Dashboard, please refer to section 5.2 For Users Using the Provided Dashboard and AWS Account of Kit to get this AWS account.
- Case 2: For users who already have an AWS account and want to use it instead of trial account, please skip section 5.2 For Users Using the Provided Dashboard and AWS Account of Kit and refer to section 5.4 For Users Using Their Own AWS Account to use the account with the application.

5.1 Hardware Preparation and Import the Project

5.1.1 Hardware Preparation

- Connect micro-USB cables to debug port (J14 on the CK-RX65N V1 board)
- Connect micro-USB cables to serial port (J20 on the CK-RX65N V1 board)
- Connect Ethernet cable to the connector (J18) on the board.
- Set the Jumper of J16 "Debug"

Note: In case user runs the RYZ014A Cellular Application, please skip the Ethernet's setting and connect the RYZ014A Cellular Pmod module which is inserted SIM card to the Pmod 1.



Figure 7. Connecting the USB and RYZ014A Pmod

5.1.2 Connecting the Board to the Serial Port Console of the PC

- 1. On the host PC, open Windows Device Manager. Expand **Ports (COM & LPT)**, locate **USB Serial Device (COMxx)** and note down the COM port number for reference in the next step.
- **Note:** USB Serial Device drivers are required to communicate between the CK-RX65N V1 board and the PC.



🛃 Device Manager	_	×
File Action View Help		
✓ #		^
> 🐗 Audio inputs and outputs		
> 😼 Batteries		
> 📓 Biometric devices		
> 🚯 Bluetooth		
> 👰 Cameras		
> 💻 Computer		
> 🔜 Disk drives		
> 🥃 Display adapters		
> 📓 Firmware		
> 🖓 Human Interface Devices		
> 🚍 Jungo Connectivity		
> 🥅 Keyboards		
> III Mice and other pointing devices		
> 🛄 Monitors		
> 🚍 Network adapters		
> 🚺 Other devices		
🗸 🛱 Ports (COM & LPT)		
 Intel(R) Active Management Technology - SOL (COM3) USB Serial Device (COM4) 		
> 🚍 Print queues		
> Processors		
> 🖗 Renesas USB Development Tools		
> Security devices		
> 🔚 Sensors		
Software components		*

Figure 8. USB Serial Device in Windows Device Manager

2. Open Tera Term select **New connection** and select **Serial** and **COMxx: USB Serial Device (COMxx)** and click **OK**.

Tera Term: New connecti	ion		×
O TCP∕IIP	Host: myhost.exa ✓ History Service: ○ Telnet ◎ SSH ○ Other	TCP port#: 22 SSH version: SSH2 Protocol: UNSPE	~ ~ C ~
Serial	Port: COM4: USI OK Cancel	3 Serial Device (COM4) Help	~

Figure 9. Selecting the Serial Port on Tera Term



3. Using the Setup menu, select **Setup** > **Terminal...** and select "**AUTO**" as Receive, select "**CR**" as Transmit, as shown below.

Tera Term: Terminal setup	×
Terminal size 80 × 24 ☑ Term size = win size ☐ Auto window resize	New-line Receive: AUTO ~ Transmit: CR ~ Cancel
Terminal ID: VT100 ~ Answerback:	Help
Coding (receive) UTF-8 ~	Coding (transmit) UTF-8 ~
locale: american	CodePage: 65001

Figure 10. Select Receive: "Auto" and Transmit: "CR" on the Terminal Setting

4. Using the **Setup** menu pull-down, select **Serial port...** and ensure that the speed is set to 115200, as shown below.

Tera	Term: Serial port setup		×	
	Port: Speed:	COM4 ~ 115200 ~	ок	
	Data:	8 bit 🗸	Cancel	
	Parity:	none v		
	Stop bits:	1 bit v	Help	
	Flow control:	none v		
	Transmit delay 0 msec/char 0 msec/line			

Figure 11. Select 115200 on the Speed Pulldown

5.1.3 Importing the Project

Use the following steps to prepare the software for the demo program:

 Extract the project files from the archive and copy them to the C drive. Please unzip the project file to a short path of your PC. If the path is deep, a build error may occur due to the file path length issue.



2. Launch e² studio and specify a workspace directory and click Launch.

📴 e² studio Launcher	— 🗆 X
Select a directory as workspace	
e ² studio uses the workspace directory to store its preference artifacts.	es and development
C:\Users\a \CK-RX65Nv1	Srowse
Use this as the default and do not ask again	
Recent Workspaces	
	Launch Cancel

Figure 12. Launch e² studio

3. Select File > Import....

File Edit Source Refactor Navigate Search Project New Alt+Shift+N > Open File Alt+Shift+N > Open File > Open Projects from File System Recent Files > > > Close Editor Ctrl+W Close All Editors Ctrl+Shift+W
Open File Open Projects from File System Recent Files Close Editor Ctrl+W
Save Ctrl+S Save As
Save All Ctrl+Shift+S Revert
Move
Rename F2
Image: Refresh F5 Convert Line Delimiters To >
Print Ctrl+P
🔤 Import
🖆 Export
Properties Alt+Enter
Switch Workspace > Restart Exit

Figure 13. Select Import



4. Click General > Existing Projects into Workspace > Next.

Select an import wizard: type filter text CMSIS Park CMSIS Park CMSIS Park Existing Projects into Workspace HIE System Proferences Projects from Folder or Archive Rename & Import Existing C/C++ Project into Workspace Renesas CC-RX project conversion to Renesas GCC RX Renesas CS+ Project for CA78K0R/CA78K0 Renesas CS+ Project for CC-RX, CC-RL and CC-RH Renesas GitHub FreeRTOS (with IoT libraries) Project Sample Projects on Renesas Website C/C++	Import Select Create new projects from an archive file or directory.		×	
	type filter text	ice	~	

Figure 14. Select Existing Projects into Workspace

5. Click **Browse**..., then specify the root directory as described later in this section.

📴 Import		_	
Import Projects Select a directory to sear	rch for existing Eclipse projects.		
• Select root directory:		~	Browse
O Select archive file:		~	Browse
Projects:			

Figure 15. Find the Project

You can choose two types of connectivity when importing the project. Please go to "[Project Root folder]\Projects" folder.

Table 6. Details of Each Project

Project Name	Compiler	Connectivity
aws_ether_ck_rx65n		Ethernet
aws_ryz014a_ck_rx65n	CC-RX	Cellular
boot_loader_ck_rx65n (for OTA)		-



and other short (Fr
 aws_ether_ck_rx65n
 aws_ryz014a_ck_rx65n
boot_loader_ck_rx65n

Figure 16. Project Files

This example uses "**aws_ether_ck_rx65n**" as the ethernet project. Open the "[**Project Root folder**]**Projects\aws_ether_ck_rx65n\e2studio_ccrx**" folder.



Figure 17. Select the Project Folder

Finally, click Finish.

Note: Make sure that the Copy projects into workspace option is unchecked.

🙆 Import		– 🗆 X
Import Projects Select a directory to see	arch for existing Eclipse projects.	
	C\Users\	
 Select archive file: Projects: 	· · · · · · · · · · · · · · · · · · ·	Browse
✓ aws_ether_ck_rx6	Sn (Cl/Users)	Select All Deselect All
		Refresh
<	>	
Options		
	ed projects upon completion	
Working sets	Iready exist in the workspace	
Add project to wo Working sets:	rking sets	New Select
?	< Back Next > Finish	Cancel
\odot	COACK INEXT > THINST	Califer

Figure 18. Finish Importing the Project



6. Check and set the SIM card information. (In case of using cellular application) Double click "aws_ryz014a_ck_rx65n.scfg" to open the smart configurator.

 CK-RX65Nv1 - aws_ryz014a_ck_rx65n/aws_ryz014a_ck_rx65n.scfg - e² st File Edit Navigate Search Project Renesas Views Run Renesas Al W S < S < I <	
	API manual Application Notes Tool news Current Configuration Selected board/device: R5F565NEHxFB (ROM size: 2MB, RAM size: 640KB, Pin cc Generated location (PROJECT_LOC\): src\smc_gen Overview Board Clocks System Components Pins Interrupts

Figure 19. Open the Smart Configurator

If using Cellular application, please check following configurations for SIM card of cellular driver:

Table 7. Information for Setting SIM of Cellular Driver When Using Truphone SIM Card in the Kit

Items	Truphone
APN (Access point name)	iot.truphone.com
UserID (Access point login ID)	(Null)
Password (Access point password)	(Null)
SIM card PIN code	(Null)
Authentication protocol type	1 (PAP)

Note: Please use the configuration of the included SIM card.

Set the above configuration values when using the SIM contained in the kits.

If you are using a different SIM card, please set the value provided by the SIM provider.

Choose "Components > Middleware > Generic > r_cellular":



CK-RX65N SIM activation, Creating the trial account and using Dashboard with RYZ014A or Ethernet Application for AWS - Getting Started Guide

aws_ryz014a_ck_rx65n.scfg ×		Generate Code Generate Repo
omponents 🚵 🖄 🖓 🕒 🕀 🛱	▼ Configure	(
type filter text Communications r_riic_rx r_sc_rx Filters fromt_rx Middleware Fromt_rx Formunications r_comms_i2c_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx Filters fromt_rx	Property	Value iot.truphone.com iot.
 r_cellular r_fwup r_ob1203_rx RTOS Kernel FreeRTOS_Kernel RTOS Object FreeRTOS_Object RTOS Library FreeRTOS_LittleFS 	# Enable user-defined URC charget functions # User URC charget function name # Debug log output level. # Reset signal logic. Macro definition: CELLULAR_CFG_AP_NAME Enter the access point name. Example: globaldata.iot, plus.4g, etc	Disable my_sw_urc_charget_function 4 1 ~

Figure 20. Set the SIM Information when Using Contained SIM Card in the Kit

7. Execute code generation.

If you have changed the Smart Configurator settings, click **Generate Code**.

Note: If the user's environment does not have the FIT component's version that matches the application, please download it by choosing **aws_ether_ck_rx65n.scfg** > **Components** > **downloading it**

Components		🖮 🖆 🖹 🖯 🕀 🏶 🍷	Configure
		10 To	Component is missing. The issue may be resolved by downloading it or sy changing the version
type filter text	t		
🗸 🗁 Startup		^	
🗸 🗁 Gene			
🔮 r,	bsp		
✓ → Drivers			
🗸 🏱 Inter	rupt		
w r	jirq_nx		
aws_ether_ck_rx	65n.scfg ×		•
oftware con	nponent configuration	n	Generate Code Generate Repo
Components	≧⊿l₂⊡≣≱▼	Configure	
	10		
1			
Lu Carta			
✓ → Startup	^		

Figure 21. Generate Code



Table 8 shows the common configurations of each component in Projects.

Table 8. Components Configuration

No	Component	Configuration
1	Startup→Generic→r_bsp (v7.20)	User stack setting: 2 stacks
		User stack size:0x2000
		Interrupt stack size: 0x400
		Heap size: 0x1000
		Initializes C input and output library functions: Enable
		Enable user stdio charget function: Use BSP charget()
		function
		User stdio charget function name:
		my_sw_charget_function
		Enable user stdio charput function: Use BSP charput()
		function
		User stdio charput function name:
		my_sw_charput_function
		Software Interrupt Unit1 (SWINT1): Unused
		Software Interrupt Unit2 (SWINT2): Unused
		Serial terminals select: Enable
		Channel for serial terminal: Channel 5
		Bitrate for serial terminal: 115200
		Interrupt priority for serial terminal: Priority level 15
		(highest)
2	Drivers→Interrupt→r_irq_rx (v4.10)	Locking function for IRQ APIs: Enable
		Resources \rightarrow ICU:
		IRQ0 Pin:
		IRQ1 Pin: 🖌
		IRQ2 Pin: 🗸
		IRQ3 Pin:
		IRQ4 Pin:
		IRQ5 Pin:
		IRQ6 Pin:
		IRQ7 Pin: ✓
		IRQ8 Pin:
		IRQ9 Pin:
		IRQ10 Pin:
		IRQ11 Pin:
		IRQ12 Pin:
		IRQ13 Pin: ✓
		IRQ14 Pin: ✓
		IRQ15 Pin: 🗸
3	Drivers→A/D	Resources→S12AD→S12AD1: ✓
	Converter→r_s12ad_rx (v5.00)	→AN115 Pin: ✓
		→AN117 Pin: ✓
4	Drivers→Memory→r_flash_rx (v5.00)	Enable code flash programming: Includes code to program ROM area
	(Enable BGO/Non-blocking data flash operations: Enable BGO (background operations/interrupt) mode
		Enable BGO/Non-blocking code flash operations: Enable
		BGO (background operations/interrupt) mode
		Enable code flash self-programming: Programming code flash while executing from another segment in ROM



No	Component	Configuration
5	Drivers→Security→r_tsip_rx (v1.17.l)	-
6	Drivers→Communications→r_ether _rx (v1.23) (For Ethernet project only)	Ethernet interface: RMII (Reduced Media Independent Interface)
		PHY-LSI address setting for ETHER0: 5
		The register bus of PHY0 for ETHER0/: Use ETHER0
		The polarity of the link signal output by the PHY-LSI: Rise -> Fall
		Use ICS1894032 of the Renesas Electronics Corporation: Used
7	Drivers→Communications→r_riic_r	MCU supported channels for CH0: Supported
	x (v2.50)	CH0 RIIC bps(kbps): 400
		Resources→RIIC
		RIICO: 🗸
		SCL0 Pin: ✓ Used
0	Diana Orange in the second in	SDA0 Pin: ✓ Used
8	Drivers→Communications→r_sci_ii c_rx (v2.50)	MCU supported channels for CH0: Supported
9	Drivers→Communications→r_sci_r	Include software support for channel 5: Include
	x (v4.40)	Include software support for channel 6: Include
		(For RYZ014A project only)
		ASYNC mode TX queue buffer size for channel 5: 80
		ASYNC mode TX queue buffer size for channel 6: 2180
		(For RYZ014A project only)
		ASYNC mode RX queue buffer size for channel 5: 80
		ASYNC mode RX queue buffer size for channel 6: 8192
		(For RYZ014A project only)
		Transmit end interrupt: Enable (For RYZ014A project only) GROUPBL0 (ERI, TEI) interrupt priority: 3
		Resources→SCI
		→SCI5: ✓
		RXD5/SMISO5/SSCL5 Pin: ✓ Used
		TXD5/SMOSI5/SSDA5 Pin: ✓ Used
		→SCI6: ✓ (For RYZ014A project only)
		 RXD6/SMISO6/SSCL6 Pin: ✓ Used
		 TXD6/SMOSI6/SSDA6 Pin: ✓ Used
		 CTS6#/RTS6#/SS6# Pin: ✓ Used
10	Drivers→Timers→r_cmt_rx (v5.40)	CMT interrupts priority level: 4
11	Middleware→	Number of I2C Share Buses: 1
	Communications→r_comms_i2c_rx	Number of I2C Communication Devices: 7
	(v1.21)	I2C Driver Type for I2C Shared Bus0: RIIC
		Channel No. for I2C Shared Bus0: 0
		I2C Shared Bus No. for I2C Communication Device0: I2C Shared Bus0
		Slave address for I2C Communication Device0: 0x44
		Callback function for I2C Communication Device0:
		rm_hs300x_callback0



CK-RX65N

No	Component	Configuration
-		I2C Shared Bus No. for I2C Communication Device1: I2C
		Shared Bus0
		Slave address for I2C Communication Device1: 0x32
		Callback function for I2C Communication Device1:
		rm_zmod4xxx_callback0
		I2C Shared Bus No. for I2C Communication Device2: I2C
		Shared Bus0
		Slave address for I2C Communication Device2: 0x33
		Callback function for I2C Communication Device2:
		rm_zmod4xxx_callback1
		I2C Shared Bus No. for I2C Communication Device3: I2C
		Shared Bus0
		Slave address for I2C Communication Device3: 0x53
		Callback function for I2C Communication Device3:
		rm_ob1203_callback0
		I2C Shared Bus No. for I2C Communication Device4: I2C
		Shared Bus0
		Slave address for I2C Communication Device4: 0x53
		Callback function for I2C Communication Device4: rm_ob1203_callback1
		I2C Shared Bus No. for I2C Communication Device5: I2C
		Shared Bus0
		Slave address for I2C Communication Device5: 0x63
		Callback function for I2C Communication Device5:
		comms_i2c_callback_icp
		I2C Shared Bus No. for I2C Communication Device6: I2C
		Shared Bus0
		Slave address for I2C Communication Device6: 0x68
		Callback function for I2C Communication Device6:
		comms_i2c_callback_icm
12	Middleware→Sensors→r_hs300x_rx	Number of HS300x Sensors: 1
	(v1.22)	Data types from HS300x Sensor: Humidity and
		Temperature
		I2C Communication device No. for HS300x sensor device0:
		I2C Communication Device0
		Callback function for HS300x sensor device0:
		hs300x_callback
13	Middleware→Sensors→r_zmod4xxx	Number of ZMOD4xxx Sensors: 2
	_rx (v1.20)	Operation mode of ZMOD4XXX Sensor0: IAQ 1st Gen.
		(Continuous)
		I2C Communication device No. for ZMOD4XXX sensor
		devices: I2C Communication Device1
		I2C callback function for ZMOD4XXX sensor device0:
		zmod4xxx_user_i2c_callback0
		IRQ callback function for ZMOD4XXX sensor device0:
		zmod4xxx_user_irq_callback0
		Enable IRQ from ZMOD4XXX sensor device 0: Enabled
		IRQ number for ZMOD4XXX sensor device0: IRQ14
		IRQ interrupt priority for ZMOD4XXX sensor device0: 10
		Operation mode of ZMOD4XXX Sensor1: OAQ 1 st Gen.



No	Component	Configuration
		I2C Communication device No. for ZMOD4XXX sensor
		devices: I2C Communication Device2
		I2C callback function for ZMOD4XXX sensor device1:
		zmod4xxx_user_i2c_callback1
		Enable IRQ from ZMOD4XXX sensor device 1: Enabled
		IRQ callback function for ZMOD4XXX sensor device1:
		zmod4xxx_user_irq_callback1
		IRQ number for ZMOD4XXX sensor device1: IRQ13
		IRQ interrupt priority for ZMOD4XXX sensor device0: 5
14	Middleware→Generic→r_byteq (v2.00)	Memory allocation for queue control blocks: Static memory allocation
		Number of static queue control block: 32
15	Middleware→Generic→r_cellular	Access point name: iot.truphone.com
	(v1.11)	Note: Depending on SIM Card information
	(For RYZ014A project only)	Access point login ID: -
		Note: Depending on SIM Card information
		Access point password: -
		Note: Depending on SIM Card information
		SIM card PIN code: -
		Note: Depending on SIM Card information
		Authentication protocol type: 1
		Network status notification level: 2
		Reset signal logic: 1
		SCI Channel: 6
		UART hardware flow control: CTS(Hardware),
		RTS(Software)
		RTS port number: PORT0
		RTS pin number: BIT2
		Reset port number: PORT5
		Reset pin number: BIT5
		IRQ Number: 4
16	Middleware→Generic→r_fwup	Select the update mode: Dual bank
	(v2.01)	Select the function mode: user for User program
		Main area start address: 0xFFF00000
		Buffer area start address: 0xFFE00000
		Install area size: 0xF0000
17	MIddleware→Generic→r_ob1203_rx	Number of OB1203 Sensors: 2
	(v1.01)	Sensor mode of OB1203 Sensor device0: Proximity sensor
		mode
		I2C Communication device No. for OB1203 sensor device0: I2C Communication Device 3
		I2C callback function for OB1203 sensor device0:
		ob1203_comms_i2c_callback
		Enable IRQ from OB1203 sensor device0: Enabled
		IRQ callback function for OB1203 sensor device0:
		ob1203_irq_callback
		IRQ number for OB1203 sensor device0: IRQ15
		IRQ trigger for OB1203 sensor device0: Falling
		IRQ trigger for OB1203 sensor device0: Falling IRQ interrupt priority for OB1203 sensor device0: Priority 14
		IRQ trigger for OB1203 sensor device0: Falling



CK-RX65N

No	Component	Configur	ation					
	· ·		munication device	No. for OB12	03 sensor o	device1:		
		I2C Communication Device 4						
		I2C callback function for OB1203 sensor device1:						
		ob1203_	comms_i2c_callb	back				
			RQ from OB1203 s					
			ack function for O	B1203 sensor	device1:			
			irq_callback					
		IRQ number for OB1203 sensor device1: IRQ15 IRQ trigger for OB1203 sensor device1: Falling						
					-			
			rupt priority for OB		device1: Pr	iority 14		
18	RTOS→RTOS		heduler: Preempt			_		
	Kernel→FreeRTOS_Kernel (V202210.01)		n number of prioriti					
	(v202210.01)	The frequence 1000	ency of the RTOS	tick interrupt:	(TickTyp	e_t)		
		The size	of the stack used I	by the idle tas	k: 768			
		The confi	gTOTAL_HEAP_S	SIZE_N: 256				
		The maxi	mum permissible l	ength of name	e: 12			
		Idle shou	ld yield: 🗸 Enable	d				
		Mutex fur	Mutex functionality: 🗸 Enable					
		Counting	semaphore function	onality: 🗸 Ena	ble			
		Software timer functionality: ✓ Enable						
		Priority of the software timer task: 6						
		The length of the software timer command queue: 5						
		Kernel interrupt priority: 1						
		Maximum syscall interrupt priority: 4						
		Tick vector: _CMT0_CMI0						
			tack high address:					
		,	allocation: 🗸 Enal	ble				
		Static allo	ocation: 🗸 Enable	1	1	1		
19	RTOS→RTOS	Initial	Task Code	Task	Priority	Stack		
	Object→FreeRTOS_Object	ize		Handler		Size		
	(V202210.01)	kernel	zmod_thread	handle_z	2	1024		
		start		mod_thre ad				
		kernel	ob1203_thread	handle_ob	3	1024		
		start		1203_thre	5	1024		
				ad				
		kernel	sensor_thread	handle_se	3	2048		
		start		nsor_thre ad				
20	RTOS→RTOS	.block_cc	ount: 70		1	<u> </u>		
	Library→FreeRTOS_LittleFS (V202210.01)							



8. Data Publishing Interval Settings (Optional)

Data publish interval can be set by the user. The default publishing interval time is 2 seconds. "Demos/SimplePubSub/simple_pub_sub_task.c" file has the macro to change the publish time interval. #define mqttexampleDELAY_BETWEEN_PUBLISH_OPERATIONS_MS (2000U)



Figure 22. Data Publishing Interval Settings (Optional)

9. Select **Project** > **Build All** and confirm that 0 errors are reported.

Note: Check the compiler setting before building project:

For both projects, from the **Projects** menu, select **Properties**, expand the **C/C++ Build** menu, and click **Settings**. On the **Toolchain** tab, confirm that the toolchain is **Renesas CC-RX**

type filter text	Settings
 Resource Builders C/C++ Build Build Variables 	Configuration: HardwareDebug [Active]
Environment Logging	🛞 Tool Setting: Toolchain Device 🎤 Build Steps 🙅 Build Artifact 扇 Binary Par
Settings Stack Analysis Tool Chain Editor C/C++ General Project Natures Project References	Enable toolchain integration Current Toolchain Toolchain: Renesas CC-RX Version: v3.05.00
Refactoring History Renesas QE	Change Toolchain (click Apply before switching tabs) Toolchain: Renesas CC-RX
Run/Debug Settings Task Tags	Version: v3.05.00
> Validation	

Figure 23. Project toolchain

Note: Make sure to clean the project before building it for the first time. If a demo build error occurs after the initial build, clean the project again and then rebuild it.





Figure 24. Build the Project

10. Debug Configuration

<u>Eile Edit Source Refactor N</u> avigate		un Renesas Al	<u>W</u> indow <u>H</u> el	p	
📃 📎 🕶 🔦 🕶 🔯	🏘 🕶 💁 🕶 📴 👖				
🍐 Project Explorer 🛛 🕹	(no launch history)	8 - 0	🔅 aws_ethe	r_ck_rx65n.scfg ×	
✓	Debug As >	ertos iot-refe	Device s	election	
> 🖑 Binaries	Debug Configurations		Device 5	election	
> 🔊 Includes	Organize Favorites				
> 鐛 Common			Device se	lection	
🗸 😪 Demos					
> 📴 cli			Board:	CK-RX65N (V1.02)	×
> 🔐 common			Device:	R5F565NEHxFB	
> 💦 dev_mode_key_provisioni	ng			Download more boards	
> 房 Fleet_Provisioning_With_C	SR_Demo				
> 房 mqtt_agent					
> 💦 OtaOverMqtt			- Fastur	e Selection	
> 房 SimplePubSub				e Selection	
> 🔀 Middleware			To add a	component, make the selection from	the table below and
> 🛃 src			The confi	igurations for each added component	can be further con
> 🗁 HardwareDebug			Features	5	Compo
🏹 > aws_ether_ck_rx65n.rcpc			Appl	ication Header	í
🏟 aws_ether_ck_rx65n.scfg			Ether	rnet	Etherne
	a Dahua Jaunch		LEDs		Ports
🖄 aws_ether_ck_rx65n Hardwar	e Debug.iaunch				

Figure 25. Configuration Debugger (1/2)



Go to **aws_ether_ck_rx65n HardwareDebug > Debugger > Connection Settings** tab then configure for "Main Clock Source: **EXTAL**" and "Connection Type: **Fine**".

			15ml
Create, manage, and run configura	itions		Ú,
			
	Name: aws_ether_ck_rx65n Hardware Debug		
type filter text	📄 Main 🔻 Debugger 🕨 Startup 🦞 Source 🔲 Common		
C/C++ Application C/C++ Remote Application	Debug hardware: E2 Lite (RX) Target Device: RSF565NE	DUAL	
GDB Hardware Debugging	GDB Settings Connection Settings Debug Tool Settings		
C GDB Simulator Debugging (RH:	✓ Clock		^
Launch Group	Main Clock Source	EXTAL	v 1
✓ C [∗] Renesas GDB Hardware Debugr.	Extal Frequency[MHz]	24	
aws_ether_ck_rx65n Hardwar	Operating Frequency [MHz]	120.000	
C* Renesas Simulator Debugging (Permit Clock Source Change On Writing Internal Flash Memory	/ Yes	~
	✓ Connection with Target Board		
	Emulator	(Auto)	
	Connection Type	Fine	~
	JTag Clock Frequency[MHz]	6.00	\sim
	Fine Baud Rate[Mbps]	1.50	~
	Hot Plug	No	~
	✓ Power		
	Power Target From The Emulator (MAX 200mA)	No	~
	Supply Voltage (V)	3.3	~
	✓ CPU Operating Mode		
	Register Setting	Single Chip	~
	Mode pin	Single-chip mode	ž
	Change startup bank Startup bank	No Bank 0	~
	Communication Mode	Darik U	· ·
	Mode	Debug Mode	~
	Execute The User Program After Ending The Debugger	No	
	Execute the oser Program Arter Ending the Debugger	NO	• •
< >			
Filter matched 9 of 11 items		Revert	Apply
n			

Figure 26. Configuration Debugger (2/2)

5.1.4 Running the Application Project

To run the Application project, use the following instructions.

The serial port console (Tera Term) of the PC set up in section **5.1.2** will display as below.



Figure 27. Start the Application

For the first-time users running the application, or for users who want to change the configuration, please press any key to set the necessary credentials for the application.

Note: After 10s, the application will automatically run the "Run Sensor App with MQTT" option from the application's menu. User can modify this time value by changing the value of WAIT_USER_TIME macro in the file: Projects\[Project folder]\e2studio_ccrx\src\application_code\CommandLine\menu_main.h before building the project:



CK-RX65N SIM activation, Creating the trial account and using Dashboard with RYZ014A or Ethernet Application for AWS - Getting Started Guide



Figure 28. Modify macro to Wait for User Input

After pressing any key, the settings are as shown below.



Figure 29. Main Menu

Choose a number to select the commands. For example, when you press '1', the firmware version of the application will be displayed as shown below. To return to the main menu, press the "space bar" key.



Figure 30. Get Version Information



5.1.5 Activating SIM Card

This section explains how to activate the SIM card that is contained in CK-RX65N V1 for using cellular application. If you use ethernet application, you can skip this step.



Figure 31. Main menu of cellular application

From the main menu of cellular application shown in the above figure, press '4' to display CAT-M Information (IMEI & ICCID). This menu will communicate with the CAT-M module to obtain the IMEI and ICCID values needed for activating the SIM card. Upon success, the IMEI and ICCID values will be displayed on the terminal screen. The program will continue to attempt to communicate with the CAT-M module until it has successfully connected or timed out. The IMEI and ICCID values are used to activate the SIM card.



Figure 32. CAT-M Information

After completing the above steps, use the information to activate the SIM card.

Note: A Truphone or MicroAl SIM card is included in this kit.

Note: The MicroAI SIM card has been discontinued to support CK-RX65N. If a MicroAI SIM card was included in the kit, please contact <u>Renesas support</u> to request a replacement to a Truphone SIM card. A MicroAI SIM card can be identified by observing that the manufacturer's name is not printed on the card.



Figure 33. MicroAl SIM Card



Please activate the SIM card using the following steps:

To activate the included SIM card, please visit the Truphone SIM Activation platform at <u>truphone.com/connectit</u> and use the following steps:

1. On the Business page, click **Start activation** button under **IoT SIM Activation**.



Figure 34. Activating the SIM Card

- 2. Create a new Truphone Account by selecting **Sign up** (next to **Don't have an account yet?**) and fill-in your full name, Email, and a password. Then click **Sign up** to create a new account.
- 3. Select **Personal** as the account type and press **Get Started**.
- 4. Verify your email by entering the activation code sent to your email account. (Note: check your Junk folder if the email is not received to your Inbox).
- 5. Complete the **Profile information** form then select **Create account**.
- Select Activate SIMs to activate your individual SIM by ICCID and PUK found on the SIM Card packaging. Note: The ICCID value can also be obtained when running cellular project. See the ICCID value in Figure 32. CAT-M Information. Fill other fields as needed.
- You can open the page using the link <u>https://account.truphone.com/login</u> and select Home > SIM Cards > ICCID#xxx", and "Activate" the SIM from the status page as shown in Figure 35. Activating the SIM Card on Truphone, the status changes from Pre-Active to Active.

Service Status		
	Status	Available Actions
SIM Card	Pre-Active	Provision Activate Retire
SMS MO Service	Active	Suspend
SMS MT Service	Active	Suspend
Current Device	Unavailable	N/A
Last Location	Unavailable	N/A
Ongoing Data Session		
Status	Offline	
Start Date	N/A	

Figure 35. Activating the SIM Card on Truphone



 You will receive email confirmation when the SIM card activation is complete. The CK-RX65N V1 kit and SIM card should be activated and can be validated on the Tera Term terminal.

Note: The SIM card includes free credit for the first 90 days / 50 MB. After the free data charge is used up, communication charges will be incurred.

Disclaimer

The activation steps above are provided by the SIM provider, Truphone. They are the most current at the time of publishing this application note. If you need help activating your SIM card, contact Truphone support <u>iot.truphone.com</u> or <u>Contact Support | Truphone</u>.

If you have a SIM card from any other provider then contact the technical support for that provider.

For any other issue that cannot be resolved please contact Renesas Support at Technical Support.

Note: The SIM card provider for the Quick Start Guide example project is Truphone. If you use any other SIM card provider, you must change the Access Point Name required for the SIM card provider in your global region (Please refer to the **6 Check and set the SIM card information** to re-configure). Failure to do so could result in the RYZ014A not connecting to the cellular network.

5.2 For Users Using the Provided Dashboard and AWS Account of Kit

This section explains getting the board "UUID" information, account registration, and accessing the dashboard.

5.2.1 Getting the Board UUID Information

Press '**3**' from the **Main Menu** to display board UUID. This displays the board UUID information on the console as shown in the screenshot below. You will need this information to register on the Cloud Dashboard.

III COM4 - Tera Term VT	_	×
File Edit Setup Control Window Help		
3. GET UUID		^
RX MCU 128-bit Unique ID (hex) : 5		
> Press space bar to return to MENU S∎		

Figure 36. Getting Board UUID Information

5.2.2 Getting the Account 10 USD for Trial of AWS

- 1. Register/sign up at "<u>https://renesas.cloud-ra-rx.com/</u>" with an email account that was not used previously for signing up to an AWS account.
- Note: The provided free credit starts being used when users register their email and UUID on this system. Renesas recommends disabling the AWS EC2 service when users do not use this system. Please refer to section **8.8** How to Enable/Disable EC2 Instance.



CK-RX65N SIM activation, Creating the trial account and using Dashboard with RYZ014A or Ethernet Application for AWS - Getting Started Guide

RENES	AS		
	RA	RX	Don't have an account? Sign up
	Sign in to Dashboard	k	
	Email address		
	Password	<i>N</i>	
		Forgot password?	
	L	ogin	

Figure 37. Get the Account 10 USD for Trial of AWS (1/2)

RENESAS		
	(RA (RX)	Already have an account? Login
	flag up i	
	Sign up Enter your details below	
	/ na	
	ार्ग्स होते. Register	

Figure 38. Get the Account 10 USD for Trial of AWS (2/2)

2. Wait for AWS verification email (it may take up to 10 minutes). Then, enter the email and UUID to register the kit as shown in the following window. You can get the UUID from section **5.2.1 Getting the Board UUID Information**

Note: Only 1 device will be assigned to an account.

RENES	<u>AS</u>	
My Devices	(RA) (RX)	0
	Register Device	
	Submit	

Figure 39. Register Device



- 3. Verify the AWS account in your email that you registered.
- 4. Wait for the status change on the registration page or wait for provisioning to complete. Please refresh the page in case the "Registration in progress" screen still shows up.

м.		3
4	Device 1	~
My Devices	Device is being provisioned. May take up to 1 hour	Status: Under Provisioning

Figure 40. Dashboard Build In Progress

5. Once the account status shows up as active on the registration page, click on the device to see device UUID.

RENESA	S					
My Devices		RA	A CONT	RX		0
	Device 1				Status: Provisioned	~

Figure 41. Active Device

6. After finishing the progress, you can get the connection certificate using the "Download Certificate" button. This is used for installation on the application demo of kits that you got in the previous step.

RENES		
My Devices	RENESAS RA DOD RX	9
	Device 1	Status: Provisioned
	Account name:	Download Certificate
	Email: Franklich under State de 1977	Go To Dashboard →
	UUID: ಅವರ್ಷ-ಆರ್ಟ್ರೋಂಡ್ ಹಾಡ್ ಎಂದ್ರಿಯ	

Figure 42. Dashboard Build Complete



 Click "Go To Dashboard" to access the dashboard. First time users will access the dashboard with credentials "admin" for both username and password and will be directed to change the password. Once completed, users can access the dashboard.



Figure 43. Dashboard for this Application

5.3 Software Preparation - Running Project from IDE

5.3.1 Storing the Device Certificate, Key, MQTT Broker Endpoint and IoT Thing Name

Device Certificate, Device Private Key, MQTT Broker Endpoint and IOT Thing name need to be stored in the data flash for the application to work. These are obtained after registering to the Cloud Dashboard.

1. Press '2' on the **Main Menu** to display **Data Flash** related commands as shown in the following screenshot. This sub menu has commands to store, read, and validate the data.



Figure 44. Data Flash related Menu and Commands

- 2. Unzip the cert.zip from the dashboard.
- To store the Device Certificate, press the option 'b'. Click the File tab of the Tera Term, select the Send File option, and choose the downloaded Device certificate file from the dashboard "xxxxcertificate.pem.crt". The details for downloading the certificates are provided in the Dashboard document linked as part of this Application Note.



CK-RX65N SIM activation, Creating the trial account and using Dashboard with RYZ014A or Ethernet Application for AWS - Getting Started Guide

<u>File Edit Setup Contro</u>		<u>H</u> elp	
New connection	Alt+N		· · · · · · · · · · · · · · · · · · ·
Duplicate session	Alt+D	ta in data flash	
Cygwin connection	Alt+G		
<u>L</u> og			
Pause Logging			
Comment to Log			
<u>V</u> iew Log			
Sho <u>w</u> Log dialog			
Stop Logging (Q)			
<u>S</u> end file			
Transfer	>		
SSH SCP			
Change directory			
<u>R</u> eplay Log			
TT <u>Y</u> Record			
TTY Replay			
Print	Alt+P		
Disconnect	Alt+I		
Exit	Alt+Q		
Exit <u>A</u> ll			



file to write data in data flash I Tera Term: Send file	×
Look in: Certs	G 🗊 📂 🛄 -
Name	Date modified
AmazonRootCA1.pem	03-06-2022 08:42 PM
certificate.pem.crt	03-06-2022 08:42 PM
iot-data.json private.pem.key	03-06-2022 08:42 PM 03-06-2022 08:42 PM
<	
File name: certificate.pem.crt	Open
Files of type: All(*.*)	✓ Cancel

Figure 46. Downloading the Device Certificate into the Data Flash

Г



٦





4. To store the Device Private Key, press the option 'c', click the File tab in the Tera Term and select the Send File option. Choose the downloaded Device Private Key "xxxxxxprivate.pem.key" which is downloaded from the Dashboard download link.

5. Open the "iot-data.json" file.

This file has information about IoT things name and IoT endpoint.

iot-data.json private.pem.key	AmazonRootCA1.pem	
	0 iot-data.json	

Figure 48. Getting the IoT Things Name and IoT Endpoint Information

To store the MQTT Broker end point, copy the end point string between the quotes xxxxxxxx.iot.us-east-1.amazonaws.com from the downloaded certificate link, press the option 'd' and click the Edit tab in the Tera Term and "Paste<CR>" and verify and confirm the valid string and press OK.
 Note: Please copy the IOTEndpoint without "".

2	1	"IOTThingName":	*	23".
3		"IOTEndpoint":		Lamazonaws.com
4				A CONTRACTOR OF



DATA FLASH WRITE MQTT END POINT	^
Select the file to write data in data flash \Box	
Tera Term: Clipboard confirmation	×
amazona	οκ
	Cancel









ATA FLASH WRITE THING NAME	
Select the file to write data in data fla	5-3
Tera Term: Clipboard confirmation	×
3647384e-3	ОК

Figure 52. Storing the Thing Name into the Data Flash

- 8. Press option 'j' and 'k' to read and validate the stored information in the data flash.
- **Note:** Validation of the stored data is very limited and validates minimum set of data points. Users are required to input the valid data to the flash obtained from the Dashboard for the proper working of the application.
- **Note:** Option **'I)** Format Flash data' will erase all saved value in data flash. Please be careful when using this option in application.


5.3.2 Starting the Application

After activating the SIM card (if using Cellular project), registering to the Dashboard and configuring the required Cloud credentials via the CLI, the application is ready to run. Press option '**Run Sensor App with MQTT**' to start the application. The application prints a welcome screen along with the status of validating the Cloud credentials data present in the data flash as shown below.

☑ COM4 - Tera File Edit Setu	Term VT up Control Window Help
CHECK CREDE	NTIALS STORED IN DATA FLASH
Fleet is d	isabled, do not need Claim private key ID
Fleet is d	isabled, do not need Claim cert ID
Fleet is d	isabled, do not need template name
OTA is dis	abled, do not need code sign certificate
IOT thing	name saved in data flash is verified and successful
MQTT Endpo	int saved in data flash is verified and successful
Private Ke	y saved in data flash is verified and successful
Certificat	e saved in data flash is verified and successful
	tials in data flash is verified and successful] Write certificate
** Alternat 1 8979 [IP-]	e Key Provisioning successfully ** Task] prvIPTask started
2 8981 [ETH]	ER_RECEI] Deferred Interrupt Handler Task started
3 8981 [ETH]	ER_RECEI] Queue space: lowest 8
4 8981 [IP-]	[ask] InitializeNetwork returns OK
5 8982 [IP-]	[ask] xNetworkInterfaceInitialise returns Ø
6 9081 [ETH]	ER_RECEI] R_ETHER_Read_ZC2: rc = -5
7 9181 [ETH]	ER_RECEI] R_ETHER_Read_ZC2: rc = -5
8 9281 [ETH]	ER_RECEI] R_ETHER_Read_ZC2: rc = -5

Figure 53. Welcome Screen on the Console



When the connection is successful, the data is shown.

🗵 COM4-	Tera Term VT
File Edit	Setup Control Window Help
25 10981	[ETHER_RECE1] R_EIHER_Read_ZC2: $\mu c = -5$
26 10982	[ETHER_RECE1] prvLinkStatusChange< 1 >
27 10982	[ETHER_RECEI] prvEMACHandlerTask: PHY LS now 1
28 11982	[]P-Task] xNetworkInterfaceInitialise returns 1
29 11982	[1P-Task] prvInitialiseDHCP: start after 250 ticks
30 12232	(IP-Task) uDHCPProcess: discover
31 12234	(ETHER_RECEI) Network buffers: 3 lowest 2
32 17482	[IF-Task] vDHCPProcess: discover
33 17482	[IP-Task] vDWCPProcess: timeout 10000 ticks
34 27732	[IP-Task] vDHCPProcess: discover
35 27732	[IP-Task] vDHCPProcess: timeout 20000 ticks
36 28517	[IF-Task] vDHCPProcess: offer c0a80395ip
37 28517	[IF-Task] vDHCPProcess: reply c0a80395ip
38 28521	[IP-Task] vDHCPProcess: offer c0a90395ip
39 28521	[IP-Task] vDWCPProcess: acked c0a90395ip
40 28521	[IP-Task] IP Address: 192.168.3.149
41 28521	[IP-Task] Subnet Hask: 255.255.255.0
42 28522	[IP-Task] Gateway Address: 192.168.3.1
43 28522	[IF-Task] DNS Server Address: 8.8.8
44 28800	[CLI] Initialise the RTOS's TCP/IP stack
45 28800	[CLI]STARTING DEHO
46 28804	(HQTT) (INFO)Start HQTT Agent Task
47 28804 48 28905	[MQTT] [INFO] Creating a TLS connection to [ETHER_RECEI] Heap: current 199048 lowest 194816
49 33838	[HQIT] FreeRIOS_ProcessDNSCache: add: 'east-1.amazonaus.com' @ 36a288clip (TIL 60)
50 33838	(MQTT) DNS(0xAA23): The answer to '
51 33840	[MqTT] FreeRTOS_connect: 61969 to 36a288c1ip:8883
52 33840	[IP-Task] prvSocketSetHSS: 1400 bytes for 36a288c1ip:8883
53 33840	[IP=Task] prvVinScaleFactor: uxRxVinSize 2 MSS 1400 Factor 0
54 33840	[IP-Task] Connect[36a288c1ip:8893]: next timeout 1: 3000 ns
55 34074	[IP-Task] MSS change 1480 -> 1452
56 34074	[IP-Task] TCP: active 61069 => 36a288c1ip:8883 set ESTAB (scaling 1>
57 34075 58 34174	[MQTT] [INFO] Established TCP connection with aeast-1.anazonaws.com. [ETHER_RECE]] Heap: current 138816 lowest 138344
59 34895	(ETHER_RECEI) Heap: current 124720 lovest 122568
	[ETHER_RECE]] Network buffers: 1 lowest 1
61 36889 62 36891 63 36891 64 37171	[MQIT] [INFO] (Network connection 805238) ILS handshake successful. [MQIT] [INFO] (Network connection 805238) Connection to ([
	[MqII] [INFO] MqII connection established with the broker. [MqII] [INFO] Successfully connected to MqII broker. [sensor_thre] 12C bus 8 setup success
	[sensor_thre] H\$3001 open sensor instance successful: 0
69 37190	[sensor_thre] ICP18181 open sensor instance successful: 8
70 37190 0B1203 De 71 37200	[ob1203_thre] vice open success [sensor_thre] ICH20948 open sensor instance successful: 0

Figure 54. Application with MQTT

Note: Above is the log of the Ethernet application running.

- **Note:** Sensor's data will be able to read correctly after stabilization time. You can also check the sensor's operation by choosing the option "**Run Only Sensors App**".
- Note: With OB1203 sensor, besides the stabilization time, OB1203 sensor data which is sent to the MQTT (shown in the terminal) is affected by the "Data Publishing Interval Settings" (refer to **Data Publishing** Interval Settings (Optional) to set this value). So, please keep your finger on the sensor until the terminal displays the correct data. It can be slightly longer than the stabilization time.
- **Note:** For Ethernet applications, firewalls in the network may prevent connectivity to AWS IoT. Configure the network to allow access to the MQTT Port 8883.

About the detail of stabilization time, please see Table 12. Sensor Stabilization Time



5.4 For Users Using Their Own AWS Account

Note: Complete the steps up to "Check AWS IoT endpoints."

5.4.1 Get an AWS Account

Get an AWS account > Click the "Sign into the Console" button.

When considering using AWS, you can use the <u>AWS Free Tier</u>.

5.4.2 Log in to the AWS Management Console

Amazon Web Services > My Account > AWS Management Console



Figure 55. Login the AWS

5.4.3 Move to IoT Core Control Panel

AWS services > All services > IoT Core

aws	Services	Q Search		[Alt+S]
≡		Console Home > A	ll services	
		[All services	
			Services by category	
			Compute	ම් Manag
			FC2	

Figure 56. Search the IoT Core (1/2)



AWS HealthOmics Amazon Bedrock	Internet of Things
AWS HealthImaging	IoT 1-Click
Amazon Q	IoT Analytics
Anglation	IoT Device Defender
Analytics	IoT Device Management
Athena	IoT Greengrass
Amazon Redshift	IoT SiteWise
CloudSearch	IoT Core
Amazon OpenSearch Service	IoT Events
Kinesis	AWS IoT FleetWise
QuickSight	IoT RoboRunner
Data Pipeline	IoT TwinMaker

Figure 57. Search the IoT Core (2/2)

5.4.4 Create a Security Policy

Security > Policies > Create policy.





VS IOT > Security > Policies > Create policy	
reate policy Info	
VS IoT Core policies allow you to manage access to the AWS IoT Core data plane operations.	
Policy properties AWS IoT Core supports named policies so that many identities can reference the same policy document.	
Policy name [ck_nx65n_test]
A policy name is an alphanumeric string that can also contain period (.), comma (.), hyphen(-), underscore (_), plus sign (+), equal sign (+), and at sign (@) characters, but no spaces.]
▶ Tags - optional	
Policy statements Policy examples	
Policy document Info	Builder JSON
An AWS IoT policy contains one or more policy statements. Each policy statement contains actions, resources, and an effect that grants or denies the actions by the resources.	
Policy document	
1 1	
2 "Version": "2012-10-17", 3 • "Statement": [
4♥ { 5 "Effect": "Allow",	
6 "Action": "", 7 "Resource": ""	
8 }	
10 3	

Figure 59. Create the Policy (2/3)



Copy the following code:

```
"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": "*"
 }
]
```

Paste the copied code into the policy document > Create







5.4.5 Register your device (thing) with AWS IoT Manage > Things > Create things



Figure 61. Creating the Things (1/5)

Creating AWS IoT things > Create single thing

Create things Info	
· · ·	esentation of a physical device or logical entity in AWS IoT. Your device or entity needs a t WS IoT features such as Device Shadows, events, jobs, and device management features.
Number of things to c	reate
• Create single thing	
Create a thing resource to r IoT.	register a device. Provision the certificate and policy necessary to allow the device to connect to AWS
Create many things	nultiple thing resources to register devices and provision the resources those devices require to
connect to AWS IoT.	nutuple uning resources to register devices and provision the resources those devices require to

Figure 62. Creating the Things (2/5)



Add your device to the thing name > **Next**

Make a note of the name with a text editor (this will be used later)

Step 3 - optional Attach policies to certificate Thing properties info Configure device certificate Thing name Curve, 55n_test] Cuter a unspice name containing one: letters, numbers, hyphens, colons, or underscores. A thing name carlt contain any spaces. Additional configurations Vou can use these configurations to add detail that can help you to organize, manage, and search your things. • Thing groups - optional • Searchable thing attributes - optional • Billing group - optional Device Shadow info Device Shadow reference • Nature of the information of this thing's tabute information of this thing's tabuto using either HTPs or NQTT topics.
Vou can use these configurations to add detail that can help you to organize, manage, and search your things. Thing type - optional Searchable thing attributes - optional Thing groups - optional Billing group - optional Billing group - optional Device Shadow Info Device Shadow Info Device Shadow unfo No shadow Named shadow Named shadow with different names to manage access to properties, and logically group Named shadow Context the state information of the state information of the shadow with different names to manage access to properties, and logically group Named shadow Context the state information of the state information of the state information of the state information of the shadow informa
Device Shadows allow connected devices to sync states with AWS. You can also get, update, or delete the state information of this thing's shadow using either HTTP's or MQIT topics. No shadow Named shadow Create multiple shadows with different names to manage access to properties, and logically group

Figure 63. Creating the Things (3/5)

Auto-generate a new certificate.

Step 1 Specify thing properties	Configure device certificate - optional Info
Step 2 - optional Configure device certificate	A device requires a certificate to connect to AWS IoT. You can choose how you to register a certificate for your device now, or you can create and register a certificate for your device later. Your device won't be able to connect to AWS IoT until it has an active certificate with an appropriate policy.
Step 3 - optional Attach policies to certificate	Device certificate
	• Auto-generate a new certificate (recommended) Generate a certificate, public key, and private key using AWS IoT's certificate authority.
	Use my certificate Use a certificate signed by your own certificate authority.
	Upload CSR Register your CA and use your own certificates on one or many devices.
	 Skip creating a certificate at this time You can create a certificate for this thing and attach a policy to the certificate at a later time.

Figure 64. Creating the Things (4/5)



Add a policy for your thing.

Step 1 Specify thing properties	Attach policies to certificate – optional Info AWS IoT policies grant or deny access to AWS IoT resources. Attaching policies to the the device.	e device certificate applies this access to
Step 2 - optional Configure device certificate		
Step 3 - optional Attach policies to certificate	Policies (1/1) Select up to 10 policies to attach to this certificate.	C Create policy [2] < 1 > ⊚
	Name	
	✓ ck_rx65n_test	

Figure 65. Creating the Things (5/5)

Download certificate, public key and private key for this thing.

Download certificates and keys	×
Download certificate and key files to install on your device so that it can AWS.	connect to
Device certificate	
You can activate the certificate now, or later. The certificate must be active for a dev AWS IoT.	ice to connect to
Device certificate Deactivate certificate] Download
6f328a5a610te.pem.crt	Jounnaa
Key files The key files are unique to this certificate and can't be downloaded after you leave t Download them now and save them in a secure place.	his page.
A This is the only time you can download the key files for this certi	ficate.
Public key file] Download
6f328a5a6105a56cc38d92dba42118-public.pem.kev	ey downloaded
Private key file] Download
6f328a5a6105a56cc38d92da42118-private.pem.key	ey downloaded
Root CA certificates	
Download the root CA certificate file that corresponds to the type of data endpoint you're using. You can also download the root CA certificates later.	and cipher suite
Amazon trust services endpoint] Download
RSA 2048 bit key: Amazon Root CA 1	,
Amazon trust services endpoint] Download
ECC 256 bit key: Amazon Root CA 3	Johnoud
If you don't see the root CA certificate that you need here, AWS IoT supp	oorts additional
root CA certificates. These root CA certificates and others are available i guides. Learn more 🖸	n our developer
	Done

Figure 66. Download certificate, public key and private key



5.4.6 Check AWS IoT Endpoint

• Make a note of the Endpoint in a text editor and so forth (will be used later)



Figure 67. Check AWS IoT Endpoints

You can also access the tutorial to register a device to AWS IoT tutorial on GitHub at https://github.com/renesas/amazon-freertos/wiki/Register-device-to-AWS-IoT

5.4.7 Running Application

After collecting Cloud Credentials, it includes:

- Device Certificate, Key, IoT Thing name: after registering device (see section 5.4.5)
- MQTT Broker endpoint: (see section 5.4.6)

Please refer to the section **5.3 Software Preparation - Running Project from IDE** for storing Cloud credentials and running application.

Note: Instead of getting cloud credentials from cert.zip file, users collected them directly from AWS cloud.



5.5 Verifying the Application Project using AWS Dashboard and Renesas Dashboard

5.5.1 Subscribe to a Topic Messages on the AWS IoT

This section describes the steps on how to verify this application example's functions.

- **Note:** Wait for the board to get the IP address from the service provider upon successful cellular/ethernet initialization, and the board to resolve the DNS lookup for the endpoint. After the successful MQTT connection message on the Console, "*Successfully connected to MQTT broker*", the device is ready for Publishing and Subscribing of Messages.
- **Note:** This Application involves AWS MQTT IOT Core, user has an option to use the AWS IOT Dashboard for the Validation purpose, in addition to using the Renesas GUI based Dashboard for customized view of all the Sensor Data.

For verification purposes, the user can use the AWS IoT core Dashboard for configuring and controlling the subscription and publishing of the topics as described in the following sections.

On the AWS cloud Dashboard, go to IoT Core and select **Test**, then choose **MQTT test client**. Subscribe to a topic listed below one at a time. The sample snapshot for subscribing to the topics is shown below.

Note: The messages shown below are **case-sensitive**; users need to take care of this when entering the publish or subscribe messages.

Only enter one message at a time. Copy the message 'as-is' between the quotes and do not include any extra spaces.

```
"<u>aws</u>/topic/iaq_sensor_data"
"<u>aws</u>/topic/oaq_sensor_data"
"<u>aws</u>/topic/hs3001_sensor_data"
"<u>aws</u>/topic/icm_sensor_data"
"<u>aws</u>/topic/icp_sensor_data"
```

- "aws/topic/ob1203 sensor data"
- **Note:** After the subscription to the Topics, the Dashboard is ready to receive the messages being published from the device.

AWS IoT ×	Subscribe to a topic	Publish to a topic
Monitor	Topic filter Info	https://ou want to subscribe. The topic filter can include MQTT wildcard characters.
	aws/topic/ob1203_sensor_data	
Connect Connect one device Connect many devices	Additional configuration Subscribe	
Test	Subscriptions	aws/topic/ob1203_sensor_data
Device Advisor	Subscriptions	aws/topic/ob/205_sensor_data
MQTT test client Device Location New	🖼 aws/topic/iaq_sensor_data	X Message payload
		♥ ★ { "message": "Hello from AWS IoT console"
Manage	aws/topic/hs3001_sensor_data	⇒ x [}
 All devices Greengrass devices 	aws/topic/icm_sensor_data	♥ X ► Additional configuration
LPWAN devices	aws/topic/icp_sensor_data	♡ × Publish
Software packages New	aws/topic/ob1203_sensor_data	♥ × [
Remote actions		
Message routing		 aws/topic/ob1203_sensor_data
Retained messages		
Security		{
▶ Fleet Hub		"081203": { "spo2 ()": "092.000", "North Data ()": "095.000",
Device software		"Heart Rate ()": "066.000", "Breath rate ()": "066.000", "P2P ()": "000.094"

Figure 68. Subscribe to a Topic Messages on the AWS IoT Screen



Subscriptions		aws/topic/icm_sensor_data
aws/topic/oaq_sensor_data aws/topic/hs3001_sensor_data aws/topic/icm_sensor_data	⇒ × > ×	Message payload { "message": "Hello from AWS IoT console" } Additional configuration Publish aws/topic/icm_sensor_data { "ICM": { "acc": { "acc": { "acc": { "x ": "000.005", "y ": "-000.005", "z ": "000.000", "y ": "-000.000", "y ": "-000.000", "y ": "-000.005", "z ": "000.000", "y ": "-000.050", "y ": "-000.050", "y ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "y ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "y ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-000.051", "z ": "-0

Figure 69. Subscribed Messages on the AWS IoT Screen

5.5.2 Publish a Topic Messages on the AWS Dashboard and Renesas Dashboard 5.5.2.1 With AWS Dashboard

The board subscribed to the topic: aws/topic/<topicRx>

If we publish the below data from AWS console

HS3001 temperature alerts:

Based on temperature, dashboard will send the alert messages to CK-RX65N V1 kit via below topic

Topic: aws/topic/set_temperature_led_data

```
Message: {"Temperature_LED": "HOT"}Will turn on RED in Tri-Color LEDMessage: {"Temperature_LED": "WARM"}Will turn on GREEN in Tri-Color LEDMessage: {"Temperature_LED": "COLD"}Will turn on BLUE in Tri-Color LED
```

Example:

Click Test > MQTT test client



AWS IOT ×	AWS IoT > MQTT test client
Monitor	MQTT test client Info
Connect	You can use the MQTT test client to monitor the MQTT messages being passed in your AWS account. Devices publish MQTT messages that are identified by topics to communicate the to inform devices and apps of changes and events. You can subscribe to MQTT message topics and publish MQTT messages to topics by using the MQTT test client.
Connect one device Connect many devices	 Connection details You can update the connection details by choosing Disconnect and making updates on the Establish connection to continue page.
Test ▶ Device Advisor MQTT test client	Subscribe to a topic Publish to a topic
Device Location New	Topic name The topic name identifies the message. The message payload will be published to this topic with a Quality of Service (QoS) of 0.
Manage	Q aws/topic/set_temperature_led_data X
All devices	Message payload
Greengrass devices	("Temperature_LED": "HOT")
LPWAN devices	
Software packages New	
Remote actions	▶ Additional configuration
Message routing	
Retained messages	Publish
Security	

Figure 70. Publish the MQTT Message (1/2)

OB1203 SPO2 alerts:

Based on SPO2 value, dashboard will send the alert messages to CK-RX65N V1 kit via below topic

Topic: aws/topic/set_spo2_led_data

Message: {"Spo_LED": "ON"} Will turn on BLUE LED in CK-RX65N V1 Message: {"Spo_LED": "OFF"}Will turn off BLUE LED in CK-RX65N V1

AWS IoT ×	 Connection details You can update the connection details by choosing Disconnect and making updates on the Establish connection to continue particular the connection of the establish connection to continue particular the establish connection the establish connection the establish connection to continue particular the establish connection the es
Connect	Subscribe to a topic Publish to a topic
Connect one device Connect many devices	Topic name The topic name identifies the message. The message payload will be published to this topic with a Quality of Service (QoS) of 0.
	Q aws/topic/set_spo2_led_data
Test	Message payload
Device Advisor	{"Spo_LED": "ON"}
MQTT test client	
Device Location New	
	Additional configuration
Manage	
All devices	Publish
Greengrass devices	
LPWAN devices	Subscriptions Topic
Software packages New	
Remote actions	You have no topic subscriptions.

Figure 71. Publish the MQTT Message (2/2)



5.5.2.2 With Renesas Dashboard

(1) Dashboard Types

Depending on the sensors, you can choose one of the dashboard types: Renesas 9-Axis sensor or Renesas. Click on **Renesas** option.

Note: The 9-AXIS MEMS Motion Tracking Sensor TDK ICM-20948 is not fitted on this CK-RX65N board due to shortages in component availability (For more information: <u>CK-RX65N v1 – Release Note</u> (renesas.com))



Figure 72. Renesas AWS Cloud Dashboard Types

Choose Renesas 9-Axis sensor.

۵ ک	Search dashboards by name		×
Q	🗅 ≡ i≣ Sort (Default A-Z) ~	S Filter by tag	*
+	Ø Recent		*
88	D General Alerting		>
0	D General		÷
Ф ©	Home		
Ø	Renesas		
	Renesas 9-Axis sensor		

Figure 73. Choosing Renesas 9-Axis Sensor



CK-RX65N SIM activation, Creating the trial account and using Dashboard with RYZ014A or Ethernet Application for AWS - Getting Started Guide



Figure 74. Renesas Dashboard displays 9-Axis Sensor

(2) Sensor Data for Cloud Kits

The Grafana dashboard displays the following Data from sensors. **Table 9. Sensor Data from Grafana Dashboard**

Sensor	Data		
HS3001- Humidity and Temperature Sensor	Temperature, F		
	Humidity, %		
ZMOD4410- Indoor Air Quality Sensor	Etoh, ppm		
	ECO2- Estimated Carbon dioxide, ppm		
	TVOC - Total Organic Compounds, mg/m^3		
OB1203 - Heart Rate, Blood Oxygen Concentration,	SPO2, %		
Pulse Oximetry, Proximity, Light and Color Sensor	HR (Heart Rate), bpm (beats per minute)		
	RR (Respiration Rate), breaths per minute		
	P2P		
ICP-10101 - Barometric Pressure and Temperature	Temperature, F		
Sensor	Barometric Pressure, mbar		
ICM-20948 Motion Tracking Sensor	Acc values, unit: g		
	Gyro Data, unit: dps (degrees per sec)		
	Mag Data, unit: mT		
OAQ – Outdoor Air Quality	OAQ, ppm		



(3) Alerting and Anomaly Detection

Grafana's alerts are ways to send notifications when a metric crosses a threshold that has been configured. By default, the dashboard has thresholds for the following sensors:

- OB1203-SPO2: SPO2 above 90, SPO2 below 90
- HS3001 Temperature, F:
 - Temperature Cold: below 65
 - Temperature Warm: within range from 65 to 85
 - Temperature Hot: above 85



Figure 75. Sensor Status Feedback

Sensor status feedback is sent to the device which is indicated by the LEDs.

6. OTA over MQTT

6.1 Overview OTA

This section describes the steps on using OTA in this application.

OTA (Over-The-Air) updates are crucial in maintaining the functionality, security, and performance of IoT devices. This feature allows the user to efficiently deploy updates, patches, or new versions of software to connected IoT devices without requiring physical access to each device.

Please refer to the document about OTA: <u>OTA-using-Amazon-Web-Services-in-RX65N-FreeRTOS-for-v202210.01-LTS-rx-1.1.0</u>

Note: This OTA feature in the application is not available for users who use AWS account with free 10\$ credit and dashboard that are provided by Renesas, because of some limitations with this account's permission.

6.2 Prerequisites

6.2.1 Installing Python

1. Access the Python download website: <u>Python downloaded website</u> and download the Python 3.11.0 installer: Click the **Download** link for Python 3.11.0



	on number:			
Release version	Release date		Click for more	
Python 3.9.16	Dec. 6, 2022	🕹 Download	Release Notes	
Python 3.8.16	Dec. 6, 2022	🕹 Download	Release Notes	1
Python 3.7.16	Dec. 6, 2022	🕹 Download	Release Notes	
Python 3.11.0	Oct. 24, 2022	🕹 Download	Release Notes	
Python 3.9.15	Oct. 11, 2022	🕹 Download	Release Notes	
Python 3.8.15	Oct. 11, 2022	Second Download	Release Notes	

Figure 76. Python Download Website

2. Run the installer and follow the prompts to install Python

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

Python 3.11.0 (64-bit) Set	up	1		×
	Install Python 3.11.0 (64-bit) Select Install Now to install Python with default setting Customize to enable or disable features.	gs, or cl	hoose	
2	→ Install Now C:¥Users¥brich¥AppData¥Local¥Programs¥Python¥Python Includes IDLE, pip and documentation Creates shortcuts and file associations	on311		
- Arrando	→ Customize installation Choose location and features			_
python	Use admin privileges when installing py.exe			
windows	Add python.exe to PATH		Can	cel

Figure 77. Python 3.11.0 installer

3. Open a command prompt and confirm that Python 3.11.0 is installed.

Execute the following command and confirm that information appears: \$python -V



Figure 78. Checking version of Python



4. Install the Python encryption library (pycryptodome)

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

E Command Prompt	-		×
C:\Users\>pip install pycryptodome Requirement already satisfied: pycryptodome in c:\users\`\appdata\local\programs\python\python311 es (3.19.0)	\lib\s	ite-pac	^ kag
<pre>[notice] A new release of pip available: 22.3 -> 23.3.1 [notice] To update, run: python.exe -m pip installupgrade pip C:\Users\></pre>			

Figure 79. Installing Python encryption library

6.2.2 Installing OpenSSL

1. Access the Win32/Win64 download website for OpenSSL: <u>OpenSSL Download Website</u> and download the installer for the operating system you are using.

Win3 <u>EXE</u>	32 OpenSSL v3.0.12 <u>MSI</u>	120MB Installer

Figure 80. OpenSSL Download Website

- 2. Run the installer and follow the prompts to install OpenSSL.
- 3. Open the Win64 OpenSSL Command Prompt and confirm that OpenSSL is installed.

Execute the following command and confirm that information appears: \$openssl version

ன Win64 OpenSSL Command Prompt	_	×
C:¥>openssl version OpenSSL 3.1.2 1 Aug 2023 (Library: OpenSSL 3.1.2 1 Aug 2023)		^
C:¥>		
		~

Figure 81. Checking version of OpenSSL

6.2.3 Installing Renesas Image Generator

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

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

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

- **Note:** Version Rev.2.00 and later of the Firmware Update module only support firmware generation using Python scripts.
- 1. Access the link <u>RX Family Firmware Update module Using Firmware Integration Technology Application</u> <u>Notes Rev.2.01 - Sample Code | Renesas</u> and download the firmware update module.



CK-RX65N SIM activation, Creating the trial account and using Dashboard with RYZ014A or Ethernet Application for AWS - Getting Started Guide



2. Extract the downloaded firmware update module.

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

🗸 🔋 Download	🛃 image-gen.py	Python File
✓	RX24T_Linear_Full_ImageGenerator_PRM.csv	Microsoft Excel CSV ファイル
> 📜 FITDemos	RX24T_Linear_Half_ImageGenerator_PRM.csv	Microsoft Excel CSV ファイル
> 📜 FITModules	RX26T_DualBank_ImageGenerator_PRM.csv	Microsoft Excel CSV ファイル
✓	RX26T_Linear_Full_ImageGenerator_PRM.csv	Microsoft Excel CSV ファイル
RenesasImageGenerator.zip	RX26T_Linear_Half_ImageGenerator_PRM.csv	Microsoft Excel CSV ファイル
	RX65N_DualBank_ImageGenerator_PRM.csv	Microsoft Excel CSV ファイル
> 🔯 r01an6850xx0200-rx-fwupdate.zip	RX65N_Linear_Full_ImageGenerator_PRM.csv	Microsoft Excel CSV ファイル
> 📙 ota_sample	RX65N_Linear_Half_ImageGenerator_PRM.csv	Microsoft Excel CSV ファイル

Figure 83. Renesas Image Generator package

6.3 Setting up AWS for OTA

6.3.1 Register your Device in AWS

See chapter **5.4** for details on how to sign up for an AWS account.

6.3.2 Creating an Amazon S3 bucket

Amazon S3 is an online storage web service used to store the firmware with the device will be updated.



1. From the Services menu, select Storage and then choose S3.





2. On the **Buckets** page, click the **Create bucket** button.

aws III Services	Q Search	[Alt+S] > + Ø Ø Global •
Amazon S3	×	General purpose buckets Directory buckets
Access Grants New Access Points Object Lambda Access Multi-Region Access F		General purpose buckets (3) Info C Copy ARN Empty Delete Buckets are containers for data stored in S3. Learn more [2]
Batch Operations		Q Find buckets by name < 1

Figure 85. Create AWS S3 bucket (2/2)



3. Enter a bucket name (example: s3test-rx65n)

Amazon S3 > Buckets > Create bucket Create bucket Info Buckets are containers for data stored in S3. Learn more [2]	
General configuration	
AWS Region	
US East (N. Virginia) us-east-1	▼
Bucket type Info	
• General purpose Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.	Directory - New Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.
Bucket name Info	
s3test-rx65n	
Bucket name must be unique within the global namespace and follow	w the bucket naming rules. See rules for bucket naming 🔀
Copy settings from existing bucket - <i>optional</i> Only the bucket settings in the following configuration are copied.	
Choose bucket	

Figure 86. Naming for bucket

Note: The bucket name must be globally unique. And it follows the bucket naming rules.

4. Create the bucket.

Enter the settings as follows, and then click the Create bucket button.

Block Public Access setting for this bucket: **Block all public access.** Bucket Versioning: **Enable**

Block Public Access settings for this bucket Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order t ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more Block all public access	et your 1
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one and	cher.
Block public access to buckets and objects granted through new access control lists (ACLs) S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public acces ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resou using ACLs.	
 Block public access to buckets and objects granted through any access control lists (ACLs) S3 will ignore all ACLs that grant public access to buckets and objects. 	
 Block public access to buckets and objects granted through new public bucket or access point policies S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change a existing policies that allow public access to S3 resources. 	апу
Block public and cross-account access to buckets and objects through any public bucket or access point policies S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.	

Figure 87. Bucket setting (1/2)



Bucket Versioning O Disable		every version of ev	ioning cans of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore very object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions illures. Learn more 🔀
○ Disable		Bucket Versionir	ng
	• Enable	O Disable	

Figure 88. Bucket setting (2/2)

6.3.3 Allocating OTA execution permission to IAM users

Create a role with the appropriate access permissions to create OTA update jobs.

1. Enter "IAM" in the search box at the top of the screen and click IAM in the search results.

Services	QIAM	× D 4 0
	Services (11)	Search results for 'IAM' Try searching with longer queries for more relevant results
∷ Welco	Features (20) Resources New	Services
	Documentation (48,933) Knowledge Articles (552)	国 IAM ☆ Manage access to AWS resources
P	Marketplace (714) Blogs (1,719) Events (12)	A IAM Identity Center 会 Manage workforce user access to multiple AWS accounts and cloud
	Tutorials (2)	B Resource Access Manager ☆ Share AWS resources with other accounts or AWS Organizations

Figure 89. IAM search box

2. In the menu, click Roles and then click the Create role button.



Figure 90. Creating a role



3. Under Select trusted entity, enter the following settings, and then click Next:

Under Trusted entity type, select AWS service Under Use cases for other AWS services, select IoT Select the IoT option button

Step 1 Select trusted entity	Select trusted entity Info
Step 2	Trusted entity type
Add permissions 	• AWS service Allow AWS services like EC2, Lambda, or others to perform actions in this account. • AWS accounts Helonging to you or a 3rd party to perform actions in this account. • Web identity Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
	SAML 2.0 federation Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account. Custom trust policy Create a custom trust policy to enable others to perform actions in this account.
	Use case
	Io1 Io1 Io1
	IoT SiteWise IoT Things Graph IoT TwinMaker
	Allows IoT to call AWS services on your behalf IoT - Device Defender Audit Provides AWS IoT Device Defender read access to IoT and related resources.
	 IoT - Device Defender Mitigation Actions Provides AWS IoT Device Defender write access to IoT and related resources for execution of Mitigation Actions.
	Cancel

Figure 91. Selecting trusted entity

4. Click **Next** on the **Add permissions** page without making any changes.

Permissions policies (3) The type of role that you selected		ing policy.
Policy name 🖓 🔹 🔻	Туре 🔻	Attached entities
AWSIOTRuleActions	AWS m	1
🕀 道 AWSIoTLogging	AWS m	1
🕀 🏮 AWSIoTThingsRegi	AWS m	1
Set permissions boundar Set a permissions boundary to control management to others.		Info ions this role can have. This is not a common setting, but you can use it to delegate permission

Figure 92. Add Permissions for Role



5. Enter a role name (example: ota_role_rx65n), and then click the **Create role** button

Name, review,	and create
Role details	
Role name Enter a meaningful name to identit	fy this role.
ota_role_rx65n	
Maximum 64 characters. Use alph	nanumeric and '+=,.@' characters.
Description Add a short explanation for this rol	le.
Allows IoT to call AWS serve	ices on your behalf.
Maximum 1000 characters. Use a	lphanumeric and '+=,.@' characters.

Figure 93. Naming for Created Role

6. Click on the role you created.

Identity and Access × Management (IAM) Q Q Search IAM	IAM > Roles Roles (20) Info An IAM role is an identity you can creat short durations. Roles can be assumed	ite that has specific permissions with credentials that are va d by entities that you trust.
Dashboard • Access management	Q. Search Role name	▽ Trusted entities
User groups Users		AWS Service:
Roles		AWS Service:
Policies Identity providers		AWS Service:
Account settings		AWS Service: s
 Access reports Access analyzer 		AWS Service: .
Archive rules		AWS Service: 1
Analyzers		AWS Service: 1
Settings Credential report	cta_role_nx65n	AWS Service: .

Figure 94. Created Role

7. Select Attach policies.

Permissions	Trust relationships T	ags Access Ad	visor Revoke sessions		
	s policies (3) Info		2 Simulate	Remove	Add permissions
You can attach u	p to 10 managed policies.				Attach policies
Q Filter polici	es by property or policy name a	nd press enter.			Create inline policy
Policy	name 🖓 🛛 🗢	Type 🗢	Description		
- 🕂 🚺	AWSIoTRuleActions	AWS managed	Allows access to all AWS services suppo	rted in AWS IoT	Rule Actions
. 🕀 🏮	AWSIoTLogging	AWS managed	Allows creation of Amazon CloudWatch L	og groups and s	treaming logs to the groups
🗆 🕀 🚺	AWSIoTThingsRegistrati	AWS managed	This policy allows users to register things	at bulk using AV	VS IoT StartThingRegistrati

Figure 95. Add permission (1/3)



8. Enter AmazonFreeRTOSOTAUpdate in the **Permissions policies** search box, and then press the **Enter** key.

You ca	an attach up to 10 managed policies.				Simulate Remove	Add permissions 🔻
Q /	AmazonFreeRTOSOTAUpdate			×		< 1 > @
	Policy name 🗗	\bigtriangledown	Туре 🗢	Description		
	AWSIoTRuleActions		AWS managed	Allows access to all AWS service	s supported in AWS IoT Rule Actions	
	🕀 📫 AWSIoTLogging		AWS managed	Allows creation of Amazon Cloud	Watch Log groups and streaming logs to the gro	ups
_	AWSIoTThingsRegistration		AWS managed	This policy allows users to regist	er things at bulk using AWS IoT StartThingRegist	rationTask API

Figure 96. Add permission (2/3)

9. Select the check box beside the AmazonFreeRTOSOTAUpdate policy, and then click the **Add permissions** button.

Attach policy to ota_role_rx65n				
• Current permissions policies (3)				
Other permissions policies (Selec	ted 1/881)			Create policy 🖉
Q Filter policies by property or policy name	and press enter.		1 match	< 1 > 🥥
"AmazonFreeRTOSOTAUpdate" X	Clear filters			
Policy name 🖉	▽ Туре	▽ Description		
AmazonFreeRTOSOTAUpda	te AWS managed	Allows user to	access Amazon FreeRTOS OTA Update	
				$\overline{}$
				Cancel Add permissions

Figure 97. Add permission (3/3)

10. From the Add permissions drop-down list, select Create inline policy.

Permissio	ons Trust relationships Tags	Access Advis	or Revoke sessions
You can at	sions policies (4) Info ttach up to 10 managed policies. r policies by property or policy name and p	ress enter.	C Simulate Remove Add permissions ▲ Attach policies Create inline policy
- P	olicy name 🗗 🗢 🗸	Type 🗢	Description
	AWSIOTRuleActions	AWS managed	Allows access to all AWS services supported in AWS IoT Rule Actions
	AWSIoTLogging	AWS managed	Allows creation of Amazon CloudWatch Log groups and streaming logs to the gro
	AWSIoTThingsRegistration	AWS managed	This policy allows users to register things at bulk using AWS IoT StartThingRegist
	AmazonFreeRTOSOTAUpd	AWS managed	Allows user to access Amazon FreeRTOS OTA Update

Figure 98. Create inline policy (1/3)

11. Click JSON, paste the following code, and then click Next.

This code grants permission to pass the IAM role to AWS services.



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

Specify permissions Info Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor. Policy editor JSON Visual Actions 🔻 1 - { Edit statement "Version": "2012-10-17", 2 3 -"Statement": [4 -{ "Effect": "Allow", 5 6 -"Action": [7 "iam:GetRole", Select a statement "iam:PassRole" 8 9], Select an existing statement in the policy or add 10 "Resource": "*" a new statement. 11 } + Add new statement 12 1 13 14

Figure 99. Create inline policy (2/3)

12. Enter a policy name (example: rx65n_ota_demo_iam_policy), and then click the **Create policy** button.

Review and create Review the permissions, specify details, a	ind tone			
Policy details	no tags.			
Policy name Enter a meaningful name to identify this noticy				
rx65n_ota_demo_iam_policy				
Maximum 128 characters. Use alphanumeric and	d '+=,.@' characters.			
Permissions defined in this Permissions in the policy document sp		or denied.		Edit
Q Search				
Allow (1 of 384 services)			Show remaining 383 s	ervices
Service \bigtriangledown	Access level		Request condition	
IAM	Limited: Read, Write	All resources	None	
			Cancel Previous Create	policy

Figure 100. Create inline policy (3/3)



13. Again, from the Add permissions drop-down list, select Create inline policy.

				·		
You c	nissions policies (4) Info an attach up to 10 managed policies. Filter policies by property or policy nat	me and press enter.		2 Simulate	Remove	Add permissions Attach policies Create inline policy
	Policy name 🔄	▽ Туре		escription		
	AWSIoTRuleActions	AWS ma	naged Al	ows access to all AWS services su	pported in AWS I	oT Rule Actions
	🕀 🏮 AWSIoTLogging	AWS ma	naged Al	ows creation of Amazon CloudWa	ch Log groups ar	nd streaming logs to the gro
	AWSIoTThingsRegistratio	on AWS mai	naged Th	is policy allows users to register th	ings at bulk using	AWS IoT StartThingRegist

Figure 101. Create inline policy (1/3)

14. Click JSON, paste the following code, and then click Next.

This code allows access to Amazon S3 where the updated firmware is stored.

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

olicy editor	Visual JSON Actions
1 • { 2 "Version": "2012-10-17", 3 • "Statement": [Edit statement
<pre>4 - { 5</pre>	Select a statement Select an existing statement in the policy or add a new statement. Add new statement





15. Enter a policy name (example: rx65n_ota_demo_s3_policy), and then click the **Create policy** button.

Review and	create		
Review the permissions,	specify details, and tags.		
Policy details			
Policy name Enter a meaningful name to i	dentify this policy		
rx65n_ota_demo_s3_p			
Maximum 128 characters. Us	e alphanumeric and "+=,,@' characters.		
Permissions de	fined in this policy Info		Edit
Permissions in the po	licy document specify which actions are allowed o	or denied.	
Q Search			
Allow (1 of 384 se	rvices)		Show remaining 383 services
Service			Request condition
S3	Limited: Read, Write	All resources	None
			Cancel Previous Create policy

Figure 103. Create inline policy (3/3)

6.4 Setting up the Device

6.4.1 Generating Key Pairs and Certificates

1. Open the Win64 OpenSSL Command Prompt and create a CA private key using ECDSA.

Execute the following command: \$openssl ecparam -genkey -name secp256r1 -out ca.key

C:¥openss1>openss1	ecparam -genkey	-name secp256r1	-out ca.key	
using curve name p	rime256∨1 instea	d of secp256r1		

Figure 104. Creating CA private key

2. Create a CA certificate from the created CA private key.

Execute the following command: \$openssl req -x509 -sha256 -new -nodes -key ca.key -days 3650 -out ca.crt

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 défault value, If you enter '.', the field will be left blank.	
Country Name (2 letter code) [AU]: State or Province Name (full come) [Some-State]: Locality Name (eg. city) []	
Organization Name (eg, company) [Internet Widgits Pty Ltd] Organizational Unit Name (eg, section) []:	
Common Name (e.g. server FOON or YOUR name) [] Email Address []	

Figure 105. Creating CA certificate



3. Create an ECDSA key pair.

Execute the following command: \$openssl ecparam -genkey -name secp256r1 -out secp256r1.keypair

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

Figure 106. Creating ECDSA key pairs

4. Create a certificate signing request from the created ECDSA key pair.

Execute the following command: \$openssl req -new -sha256 -key secp256r1.keypair > secp256r1.csr



Figure 107. Creating certificate signing request

5. Create a certificate from the certificate signing request, CA certificate, and CA private key.

Execute the following command: \$openssl x509 -req -sha256 -days 3650 -in secp256r1.csr -CA ca.crt - Cakey ca.key -Cacreateserial -out secp256r1.crt



Figure 108. Creating code signing certificate

6. Extract the private key from the ECDSA key pair.

Execute the following command:

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



Figure 109. Create private key (secp256r1.privatekey)

7. Extract the public key from the ECDSA key pair.

Execute the following command:

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



CK-RX65N SIM activation, Creating the trial account and using Dashboard with RYZ014A or Ethernet Application for AWS - Getting Started Guide

C:¥openssl>openssl ec −in secp256r1.keypair -outform PEM -pubout -out secp256r1.publickey Tead EC key writing EC key

Figure 110. Create public key (secp256r1.publickey)

8. After creating, please check the result:

Name	Date modified	Туре	Size
📮 ca.crt	12/10/2023 10:06 AM	Security Certificate	1 KB
🗋 ca.key	12/10/2023 10:05 AM	KEY File	1 KB
🗋 ca.srl	12/10/2023 10:07 AM	SRL File	1 KB
属 secp256r1.crt	12/10/2023 10:07 AM	Security Certificate	1 KB
secp256r1.csr	12/10/2023 10:07 AM	CSR File	1 KB
secp256r1.keypair	12/10/2023 10:06 AM	KEYPAIR File	1 KB
secp256r1.privatekey	12/10/2023 10:18 AM	PRIVATEKEY File	1 KB
secp256r1.publickey	12/10/2023 10:14 AM	PUBLICKEY File	1 KB

Figure 111. Total created files

6.4.2 Setting up the project

6.4.2.1 Creating initial firmware

The following steps explain how to create the initial firmware that combines the boot loader (boot_loader_ck_rx65n) and the firmware (aws_ryz014a_ck_rx65n (when using Cellular project), or aws_ether_ck_rx65n (when using Ethernet project).

1. Additional import **boot_loader_ck_rx65n** project:

Make sure that the configuration in **boot_loader_ck_rx65n.scfg** file of boot_loader_ck_rx65n project is as below:

Table 10.	Components	Configuration	of boot loader project
-----------	------------	---------------	------------------------

No	Component	Configuration
1	Startup→Generic→r_bsp (v7.20)	Enable user stdio charput function: Use user charput() function
2	Drivers→Memory→r_flash_rx (v5.00)	Enable code flash programming: Includes code to program ROM area
		Enable code flash self-programming: Programming code flash while executing from another segment in ROM
3	Drivers→Communications→r_sci_rx (v4.40)	Include software support for channel 5: Include
4	Middleware→Generic→r_byteq (v2.00)	Memory allocation for queue control blocks: Static memory allocation
		Number of static queue control blocks: 32
5	Middleware→Generic→r_fwup	Select the function mode: user for Boot Loader
	(v2.01)	Main area start address: 0xFFF00000
		Buffer area start address: 0xFFE00000
		Install area size: 0xF0000



Check the boot loader device.

And also, in the **boot_loader_ck_rx65n.scfg** file, click the **Board** tab. Confirm that R5F565NEHxFB_DUAL appears in the **Board** field.

🔅 boot_lo	ader_ck_rx65n.	scfg $ imes$					
Device	selection						
Device s	election						
Board:	Custom Use	r Board			·		
Device	R5F565NEF	IxFB_DU	AL				
	Download n	<u>nore boa</u>	rds				
							_
Overview	Board Clocks	System	Components	Pins	Interrupts		

Figure 112. bootloader's device selection

Assign public key: Copy the contents of the secp256r1.publickey file you created in "Extract the public key from the ECDSA key pair.", and paste the contents into CODE_SIGNER_PUBLIC_KEY_PEM defined in the following files: boot_loader_ck_rx65n\src\key\code_signer_public_key.h



Figure 113. Add Code Signing Public Key to Boot Loader Project



2. Set ENABLE_OTA_UPDATE_DEMO to 1 (Enable) in [Project Loc]\e2studio ccrx\src\frtos config\demo config.h. (The default is 0)

h demo_config.h	
aemo_comg.n	
77	⊖ /* Please select a provisioning method
78	* (0) : Pre-provisioning
79	* (1) : Fleet provisioning
80	*/
81	#define ENABLE_FLEET_PROVISIONING_DEMO (0)
82	
83	⊖/* Please select whether to enable or disable the OTA demo
84	* (0) : OTA demo is disabled
85	* (1) : OTA over MQTT demo is enabled
86	*/
87	<pre>#define ENABLE_OTA_UPDATE_DEMO (1)</pre>
88	
89	#define democonfigROOT_CA_PEM tlsSTARFIELD_ROOT_CERTIFICATE_PEM
90	
92	● * @brief Path of the file containing the provisioning claim certificate. This
109	#define democonfigCLAIM CERT PEM "insert here"
110	
112	⊕ * @brief Path of the file containing the provisioning claim private key. This
122	#define democonfigCLAIM_PRIVATE_KEY_PEM "insert here"
122	Weithe demotor ingertal rate in the second s
	A Abrief An entire to disable Course News Todisation D
125	● * @brief An option to disable Server Name Indication.[]
131	#define democonfigDISABLE_SNI (pdFALSE)

Figure 114. Enable OTA Demo

3. Confirm the initial project version:

Confirm the version definitions in [Project_Loc]\src\frtos_config\demo_config.h:

- APP_VERSION_MAJOR
- APP_VERSION_MINOR
- APP_VERSION_BUILD

Example: The initial project version is 0.9.2

```
ⓑ demo_config.h ×
 330
                ⊖ /**
                  * @brief Major version of the firmware.
 331
 332
 333
                  * This is used in the OTA demo to set the appFirmwareVersion variable that is
                    declared in the ota_appversion32.h file in the OTA library.
 334
                   */
 335
                #ifndef APP_VERSION_MAJOR
 336
                      #define APP_VERSION_MAJOR
 337
                                                   0
 338
                  #endif
 339
                🕀 * @brief Minor version of the firmware...
 341
                #ifndef APP_VERSION_MINOR
 346
 347
                     #define APP_VERSION_MINOR
                                                   9
 348
                  #endif
 349
                * @brief Build version of the firmware...
 351
 356
                #ifndef APP_VERSION_BUILD
 357
                      #define APP_VERSION_BUILD
                                                   2
 358
                  #endif
```

Figure 115. The initial project version 0.9.2



On the **Tool Settings** tab, expand the **Converter** menu and select **Output**. Confirm that the **Motorola S** format file check box is selected.

type filter text	Settings $(\neg \bullet \bullet \bullet)$
 > Resource Builders > C/C++ Build Build Variables Environment Logging Settings Stack Analysis Tool Chain Editor > C/C++ General Git Project Natures Project References Renesas QE Run/Debug Settings Task Tags > Validation 	Configuration: HardwareDebug [Active] Manage Configurations Image: Tool Settings Tool Chain Device Image: Build Steps Build Artifact Image: Build Steps Imag

Figure 116. Project setting

4. Device selection setting

Open the file aws_ether_ck_rx65n.scfg and click the **Board** tab. Click the ellipsis (...) beside the **Board** field in the **Device selection** area.

🏠 Project Explorer 🗙 📄 🛱 🖓 🖇 🗖 🗖	aws_ether_ck_rx65n.scfg ×											
Sector State St	Device selection	Device selection										
> 🔊 Includes	1											
> 🚱 Common	Device selection											
> 🚑 Demos												
> 🚱 Middleware	Board: CK-RX65N (V1.02)											
> 🕞 > src	Device: R5F565NEHxFB											
> 🗁 HardwareDebug	Download more boards											
> aws_ether_ck_rx65n.rcpc		/										
aws_ether_ck_rx65n.scfg												
26 > aws_ether_ck_x65n Hardware Debug.launch	✓ Feature Selection											
(?) > Developer Assistance [iot-reference-rx iot-reference]												
> 😽 > aws_ryz014a_ck_rx65n (h_e2studio_ccrx) [iot-refere												
> 🔓 > boot_loader_ck_rx65n (in extudio_ccrx) [iot-refere	The configurations for each added component ca	n be further configured in the "Components" page.										
	Features	Components	Action	Link								
	Application Header	0	Add	4								
	Ethernet	Ethernet Driver. (r_ether_rx)	Apply	⇒								
	LEDs	Ports	Add	⇒								
	PMOD 1 (Cellular/UART/IIC/SPI)	 Cellular Module control functions for Renesas MCUs. (r_cellular) 	Add	⇒								
	PMOD 2 (Cellular/UART/IIC/SPI)	 Cellular Module control functions for Renesas MCUs. (r_cellular) 	Add	⇒								
	Universal Seria Bus	 USB Host Communication Device Class (r_usb_hcdc) 	Add	⇒								
	Overview Board Clocks System Components Pin	Interrupts										

Figure 117. Device selection (1/3)



Click the ellipsis (...) beside the **Target Device** field and select R5F565NEHxFB_DUAL. The value in the **Target Board** drop-down list changes to **Custom**.

Refactoring	- 🗆 X	0			_		×
Change Device		Device Selection					
Select the new device for aws_ether_ck_rx65n		You can filter devices by regular express	sion				
Current Device: R5F565NEHxFB		Search Device					
Current Board: CK-RX65N		Device	RAM	ROM	Pin		^
		✓ RX65N	10-101	nom			
Target Board: Custom	~	> RX65N - 100pin					
	Download additional boards	✓ RX65N - 144pin					_
Target Device: R5F51101AxLM	1 🛄	R5F565N4AxFB	256 KB	512 KB	144		
		R5F565N4BxFB	256 KB	512 KB	144		_
	Unlock Devices	R5F565N4ExFB	256 KB	512 KB	144		
		R5F565N4FxFB	256 KB	512 KB	144		
		R5F565N7AxFB	256 KB	768 KB	144		
		R5F565N7BxFB	256 KB	768 KB	144		
		R5F565N7ExFB	256 KB	768 KB	144		
		RSE565N7FxFB	256 KB	768 KB	144		
		R5F565N9AxFB	256 KB	1 MB	144		
		R5F565N9BxFB	256 KB	1 MB	144		
		R5F565N9ExFB	256 KB	1 MB	144		
		R5F565N9FxFB	256 KB	1 MB	144		
		R5F565NCDxFB	640 KB	1.5 MB	144		
		R5F565NCDxFB_DUAL	640 KB	1.5 MB	144		
		R5F565NCHxFB	640 KB	1.5 MB	144		
		R5F565NCHxFB_DUAL	640 KB	1.5 MB	144		
		R5F565NEDxFB	640 KB	2 MB	144		
		R5F565NEDxFB_DUAL	640 KB	2 MB	144		
		R5F565NEHxFB	640 KB 💊	2 MB	144		
		2 R5F565NEHxFB_DUAL	640 KB	2 MB	144		
		> RX65N - 145pin					
		> RX65N - 176pin					
		> RX65N - 177pin					
		> RX65W-A					
		> RX660					
		> RX66N					
		> RX66T		3			~
		DV671		- 3	1		
Rext >	Finish Cancel				ок	Cance	
I Vext >	Concer				UK	Cance	

Figure 118. Device selection (2/3)

Under Build Settings > HardwareDebug > Toolchain Settings, clear the ROM to RAM mapped section (-rom) and Sections (-start) check boxes and then click Finish.

Refactoring	— 🗆 X	
Change Device The following changes to 4 files are necessary to perform the refactoring.		
Changes to be performed	\$ } \	
 Build Settings Build Settings Build Settings Device Name Device Name Device Name Device Command Device Command ROM to RAM mapped section (-rom) Sections (-start) Project Files Smart Configurator 		
No preview available ? < Back	Cancel	

Figure 119. Device selection (3/3)



5. Change the firmware (aws_ether_ck_rx65n (or aws_ryz014a_ck_rx65n)) vector.

Open the aws_ether_ck_rx65n (or aws_ryz014a_ck_rx65n) project, select Project and then Properties.

Expand the C/C++ Build menu and click Settings. In the menu tree on the Tool Settings tab, expand the Linker menu and click Section, and open the Section Viewer. Allocate EXCEPTVECT to 0xFFFEFF80 and RESETVECT to 0xFFFEFFFC.



Figure 120. Vector setting

Build the project as described in Topic: **5.1.3 Importing the Project**

6. Generate the initial firmware.

Place the following files in the Renesas Image Generator folder:

The results of building the firmware: **aws_ether_ck_rx65n.mot (or aws_ryz014a_ck_rx65n.mot)** The results of building the boot loader: **boot_loader_ck_rx65n.mot** The private key created in "**Extract the private key from the ECDSA key pair.**": **secp256r1.privatekey**



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

\$ python image-gen.py -iup aws_ether_ck_rx65n.mot -ip RX65N_DualBank_ImageGenerator_PRM.csv -o userprog -ibp boot_loader_ck_rx65n.mot -key secp256r1.privatekey -vt ecdsa -ff RTOS

6.4.2.2 Running OTA project

1. Start Renesas Flash Programmer and create new project:

1	Renesas Flash Progr	rammer \	V3.12.00			_	×
File	Target Device	Help					
	New Project		ck Settings	Connect Settings	Unique Code		
	Open Project		on ooungo	Connect Cottings	chique coue		
	Save Project						

Figure 121. Create New Flash Project (1/4)

After that:

- Choose Microcontroller: RX65x
- Input project name.
- Browse "Project Folder"
- Communication: Tool: E2 emulator Lite
- Communication: Interface: **FINE**
- Choose "Connect"

Note: Jumper of J16 is "Debug" mode.

📓 Renesas Flash Programmer V3.12.00 — 🗉 🗙
File Target Device Help
Operation Operation Settings Block Settings Connect Settings Unique Code
Z Create New Project - X
Project Information
Microcontroller: RX65x 1 ~
Project Name: Rash_RX65N 2 Browse
Project Folder: C:\Users\5\Documents\Renesas Rash I] 3Browse
Communication Tool: E2 emulator Lite 4 Interface: FINE 5 Tool Details Num: Auto Select: Power: None
Connect 6 Cancel
Clear status and message

Figure 122. Create New Flash Project (2/4)



CK-RX65N SIM activation, Creating the trial account and using Dashboard with RYZ014A or Ethernet Application for AWS - Getting Started Guide

🕻 Create New Pro	ject — 🗆 🗙
Project Information	
Microcontroller: Progress Re F	BY65v V
F Running	
Comr	Authentication – 🗆 X
Tool Details	Authentication Code ID Code: FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
ж цемье	Auto Authentication 1 OK 2 Cancel

Figure 123. Create New Flash Project (3/4)

Renesas Flash Programmer V3.12.00	_		Х
File Target Device Help			
Operation Operation Settings Block Settings Connect Settings Unique Code			
Project Information Current Project: Flash_RX65N.rpj Microcontroller: RX Group Endian:	Little	~	•
Program File	В	rowse	
Rash Operation Erase >> Program >> Verify			
Start			
Connecting to the target device Setting the target device Setting the target device Query the device information. Signature: Device: RX Group Code Flash 1 (Address : 0xFFFF0000, Size : 64 K, Erase Size : 8 K) Code Flash 1 (Address : 0xFFE00000, Size : 1984 K, Erase Size : 32 K) Data Flash 1 (Address : 0xFFF0000, Size : 32 K, Erase Size : 64) Config Area (Address : 0xFF75D00, Size : 128, Erase Size : 0)			^
Disconnecting the tool Operation completed.			
			*

Figure 124. Create New Flash Project (4/4)


2. Choose code flash area in configuration:

Only tick option "Select" for "Code Flash 1".

Figure 125. Choose Code Flash Area

3. Erase "Code Flash" before loading new image:

Choose "**Operation Settings**" > "**Erase**".

Operation Operation Settings Block Settings Connect Settings Unique Code Command	Command Erase Program Erase Selected Blocks Verify Program & Verify Options Checksum Erase Before Program Verify by reading the device Verify by reading the device Fill with 0xFF 32bit additive method Checksum / User Boot Error Settings	Renesas Flash Programmer V3.12.00 File Target Device Help	– – X
Code Flash / User Boot Error Settings	Code Flash / User Boot Error Settings	Command Erase Program Verify	Erase Options Erase Selected Blocks Program & Verify Options Erase Before Program Verify by reading the device Checksum Type
		Code Flash / User Boot	

Figure 126. Erase Code Flash (1/2)



CK-RX65N SIM activation, Creating the trial account and using Dashboard with RYZ014A or Ethernet Application for AWS - Getting Started Guide

📕 Renesas Flash Programmer V3.12.00	- ×
File Target Device Help	
Operation Operation Settings Block Settings Connect Settings Unique Code	
Project Information	
Current Project: Flash_RX65N.rpj	
Microcontroller: RX Group	Endian: Little 🗸
Program File	
	Browse
Flash Operation	
Erase	
Start	Run
Start	Run
Start	Run
Emulator's supply power : OFF	Run
Emulator's supply power : OEF Emulator's firmware ver Connecting to the target	Run
Emulator's supply power : QEF Emulator's firmware ver Connecting to the targe Main Clock : 16.000000 Setting the target devic	Run
Emulator's supply power : QEF Emulator's firmware ver Connecting to the targe Main Clock : 16.000000 Setting the target devic Setting the target devic	Run
Emulator's supply power : OEF Emulator's firmware ver Connecting to the target Main Clock : 18,000000 Setting the target devic Setting the target devic Communication speed : Signature: Device: RX Group	Run
Emulator's supply power : QEF Emulator's firmware ver Connecting to the targe Main Clock : 16,000000 Setting the target devic Setting the target devic Communication speed : Signature: Device: RX Group	Run
Emulator's supply power : OEE Emulator's firmware ver Connecting to the targe Main Clock : 16.000000 Setting the target devic Setting the target devic Communication speed : Signature: Device: RX Group Erasing the selected blo	Run
Emulator's supply power : QEF Emulator's firmware ver Connecting to the targe Main Clock : 16,00000 Setting the target devic Setting the target devic Communication speed : Signature: Device: RX Group Erasing the selected blc [Code Flash 1] 0xFFE	Run

Figure 127. Erase Code Flash (2/2)

4. Write the initial firmware (userprog.mot)

This flash project will use commands: Erase, Program and Verify.

🌠 Renesas Flash Programmer V3.12.00	— □
File Target Device Help	
Operation Operation Settings Block Settings Conne	ct Settings Unique Code
Command Erase Program Verfy Checksum	Erase Options Erase Selected Blocks ✓ Program & Verify Options □ Erase Before Program ✓ Verify by reading the device ✓
	Checksum Type 32bit additive method ~
Fill with 0xFF Code Flash / User Boot Data Flash	Error Settings

Figure 128. Flash the Firmware (1/3)



🕻 Renesas Flash Programmer V3.12.00		_	
File Target Device Help			
Operation Operation Settings Block Settings Conr	ect Settings Unique Code		
Project Information			
Current Project: Flash_RX65N.rpj			
Microcontroller: RX Group		Endian: Little	\sim
Program File			
userprog.mot	1	Bro	owse
	CRC-3	32 : 7FB26A1A	
Flash Operation			
Erase >> Program >> Verify 2			
Start		Ru	n
Progress Report			
Main Clock : 16.000000M Setting the target device Bunning			
Communication speed : 1			
Signature: Device: RX Group			
Erasing the selected bloc [Code Flash 1] 0xFFE0	Abort		
Writing data to the target device [Code Flash 1] 0xFFEF0000 - 0xFFEF7BFF siz [Code Flash 1] 0xFFEFF80 - 0xFFF6887F siz	m : 21 K		
[Code Flash 1] 0xFFEFF80 - 0xFFF6887F siz	e: 418.3 K		
I			~
		Clear status and	

Figure 129. Flash the Firmware (2/3)

If it is successful, it will display "Operation completed":

Verifying data [Code Flash 1] 0xFFEF0000 - 0xFFEF7BFF size : 31 K [Code Flash 1] 0xFFEFF80 - 0xFFF6887F size : 418.3 K [Code Flash 1] 0xFFFEFF80 - 0xFFFF7BFF size : 31.1 K [Code Flash 1] 0xFFFFFF80 - 0xFFFFFFFF size : 128 Disconnecting the tool Operation completed.	
[Code Flash 1] 0xFFEF0000 - 0xFFEF7BFF size : 31 K [Code Flash 1] 0xFFEFFF80 - 0xFFF6887F size : 418.3 K [Code Flash 1] 0xFFFEFF80 - 0xFFFF7BFF size : 31.1 K [Code Flash 1] 0xFFFFFF80 - 0xFFFFFFFF size : 128 Disconnecting the tool	
[Code Flash 1] 0xFFEF0000 - 0xFFEF7BFF size : 31 K [Code Flash 1] 0xFFEFFF80 - 0xFFF6887F size : 418.3 K [Code Flash 1] 0xFFFEFF80 - 0xFFF7BFF size : 31.1 K	
[Code Flash 1] 0xFFEF0000 - 0xFFEF7BFF size : 31 K [Code Flash 1] 0xFFEFF80 - 0xFFF6887F size : 418.3 K	
/erifying data	
[Code Flash 1] 0xFFFEFF80 - 0xFFFF7BFF size : 31.1 K [Code Flash 1] 0xFFFFFF80 - 0xFFFFFFFF size : 128	
[Code Flash 1] 0xFFEF0000 - 0xFFEF7BFF size : 31 K [Code Flash 1] 0xFFEFFF80 - 0xFFF6887F size : 418.3 K	

Figure 130. Flash the Firmware (3/3)



5. Running the application:

Set jumper J16 to "Run" mode. The application will run as below:





Press any key to configure the application. For setting Cloud's credentials for IoT Device, please refer to the "**0**" section.

Besides that, for OTA, please store the code signing certificate (user created "**secp256r1.crt**" file at step: *Create a certificate from the certificate signing request, CA certificate, and CA private key*).

Press '2' on the Main Menu to display Data Flash.



Figure 132. Main Menu

Press 'f' for storing code signing certificate:

```
Minimizer COM4-Tera Term VT
File Edit Setup Control Window Help
> Select from the options in the menu below:
2. DATA FLASH
a> Info
b> Write Certificate
c> Write Private Key
d> Write MQIT Broker end point
e> Write IOT Thing name
f> Write code signing certificate (for OTA)
g> Write template name (for Fleet)
h> Write claim cert ID (for Fleet)
i> Write claim private key ID (for Fleet)
j> Read Flash
k> Check credentials stored in flash memory
I> Format Flash data
m> Help
```





File Edit Setup Control Window Help		
ATA FLASH WRITE CODE SIGNING CERT		
Select the file to write data in data flash		
🚾 Tera Term: Send file		×
Look in:	G 🌶 📂 🛄 -	
Name	Date modified	^
🔄 secp256r1.crt	12/10/2023 10:07 AM	
secp256r1.csr	12/10/2023 10:07 AM	
secp256r1.keypair	12/10/2023 10:06 AM	
secp256r1.privatekey	12/10/2023 10:18 AM	
secp256r1.publickey	12/10/2023 10:14 AM	~
<	>	
File name: secp256r1.crt	Open	
Files of type: All(*.*)	 ✓ Cancel 	
	Help	

Figure 134. Store code signing certificate into flash (1/2)

Note: please check the EOL of secp256r1.crt and convert it to LF before saving it into data flash.



Figure 135. Store code signing certificate into flash (2/2)



After saving all Cloud credentials, user starts the application by choosing option "**Run Sensor App with MQTT**" on Main Menu:

COM4 - Tera Term VT File Edit Setup Control Window Help
CHECK CREDENTIALS STORED IN DATA FLASH
Fleet is disabled, do not need Claim private key ID
Fleet is disabled, do not need Claim cert ID
Fleet is disabled, do not need template name
Code signing certificate saved in data flash is verified and successful
IOT thing name saved in data flash is verified and successful
MQTT Endpoint saved in data flash is verified and successful
Private Key saved in data flash is verified and successful
Certificate saved in data flash is verified and successful
All credentials in data flash is verified and successful 0 37498 [CLI] Write certificate
** Alternate Key Provisioning successfully ** 1 37696 [IP-Task] prvIPTask started
2 37698 [ETHER_RECEI] Deferred Interrupt Handler Task started
3 37698 [ETHER_RECEI] Queue space: lowest 8
4 37698 [IP-Task] InitializeNetwork returns OK
5 37698 [IP-Task] xNetworkInterfaceInitialise returns 0
6 37798 [ETHER_RECEI] R_ETHER_Read_ZC2: rc = -5
7 37898 [ETHER_RECEI] R_ETHER_Read_ZC2: rc = -5
8 37998 [ETHER_RECEI] R_ETHER_Read_ZC2: rc = -5
9 38098 [ETHER_RECEI] R_ETHER_Read_ZC2: rc = -5
10 38198 [ETHER_RECEI] R_ETHER_Read_ZC2: rc = -5
11 38298 [ETHER_RECEI] R_ETHER_Read_ZC2: rc = -5
12 38398 [ETHER_RECEI] R_ETHER_Read_ZC2: rc = -5
13 38498 [ETHER_RECEI] R_ETHER_Read_ZC2: rc = -5
14 38598 [ETHER_RECEI] R_ETHER_Read_ZC2: rc = -5
15 38698 [ETHER_RECEI] R_ETHER_Read_ZC2: rc = -5

Figure 136. Application with OTA (1/2)



While subscribing/publishing messages to MQTT (sensor's data, control LEDs and so forth), application will also check the event of OTA from Cloud.

COM4 - Tera Term VT
File Edit Setup Control Window Help
49 57222 [MQTT] [INFO]Start MQTT Agent Task
50 57222 [MQIT] [INFO] Creating a TLS connection tous-east-1.amazonaws.com:8883. 51 57253 [ETHER_RECEI] Heap: current 187784 lowest 183808
52 57253 [MQIT] FreeRIOS_ProcessDNSCache: add: 'us-east-1.amazonaws.com' @ 36a66e33ip (IIL 60)
53 57253 [MQIT] DNS[0x5487]: The answer to 'us-east-1.amazonaws.com' (54.166.110.51) will be stored
54 57255 [MQIT] FreeRIOS_connect: 26808 to 36a66e33ip:8883
55 57255 [IP-Task] prvSocketSetMSS: 1400 bytes for 36a66e33ip:8883
56 57255 [IP-Task] prvWinScaleFactor: uxRxWinSize 2 MSS 1400 Factor 0
57 57255 [IP-Task] Connect[36a66e33ip:8883]: next timeout 1: 3000 ms
58 57490 [IP-Task] MSS change 1400 -> 1452
59 57490 [IP-Task] TCP: active 26808 => 36a66e33ip:8883 set ESTAB (scaling 1)
60 57491 [MQIT] [INFO] Established TCP connection with,us-east-1.amazonaws.com. 61 57590 [ETHER_RECEI] Heap: current 127808 lowest 127336
62 58314 [ETHER_RECEI] Heap: current 113712 lowest 111560
63 60304 [MQIT] [INFO] (Network connection 805234) TLS handshake successful. 64 60304 [MQIT] [INFO] (Network connection 805234) Connection to 65 60304 [MQIT] [INFO] Creating an MQIT connection to the broker. 66 60581 [MQIT] [INFO] MQIT connection established with the broker. 67 60581 [MQIT] [INFO] Successfully connected to MQIT broker. 68 60587 [OIA Demo Ta] [INFO]Start OIA Task
70 60593 [sensor_thre] I2C bus 0 setup success
71 60593 [sensor_thre] HS3001 open sensor instance successful: 0
72 60593 [sensor_thre] ICP10101 open sensor instance successful: 0
73 60593 [ob1203_thre] OB1203 Device open success 74 60602 [sensor_thre] ICM20948 open sensor instance successful: 0
69 60592 [OTA Demo Ta] [INFO] OTA over MQTT demo, Application version 0.9.2 75 60746 [MainAWSDemo] [INFO]Start AWS - MQTT Demo Task
76 60757 [OTA Demo Ta] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0 77 60781 [ETHER_RECEI] Heap: current 83096 lowest 83096
78 60781 [OTA Agent T] [INFO] Current State=[RequestingJob], Event=[Start], New state=[RequestingJob] 79 62209 [MainAWSDemo] [INFO] Successfully subscribed to topic: aws/topic/set_temperature_led_data 80 62843 [zmod_thread] ZMOD4410 open sensor instance successful: 0
81 63049 [OTA Demo Ta] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0 82 63075 [zmod_thread] ZMOD4510 open sensor instance successful: 0
83 63076 [zmod_thread] Task zmod4410 measurement Success:0
84 63208 [OTA Agent T] [INFO] Subscribed to topic \$aws/things//jobs/notify-next.
85 63215 [OTA Agent T] [INFO] Subscribed to MQTT topic: \$aws/things//jobs/notify-next 86 64223 [MainAWSDemo] [INFO] Successfully subscribed to topic: aws/topic/set_spo2_led_data 87 64230 [MainAWSDemo] [Send Data] ZMOD4410-IAQ TUOC: 000.000 82 64230 [MainAWSDemo] [Send Data] ZMOD4410-IAQ TUOC: 000.000

Figure 137. Application with OTA (2/2)

6.5 Updating the Firmware

6.5.1 Creating the updated firmware

6.5.1.1 Changing the firmware version

Change the firmware version to a higher version. (Example: The previous version is 0.9.2, so the new version we can choose is 0.9.3)

Repeat the build process, this time with 3 specified for the APP_VERSION_BUILD definition in [Project_Loc]\e2studio_ccrx\src\frtos_config\demo_config.h.



CK-RX65N SIM activation, Creating the trial account and using Dashboard with RYZ014A or Ethernet Application for AWS - Getting Started Guide

🎦 Project Explorer 🗙 📄 🛱 🏹 🖇 🖵 🗖	h demo_config.h ×
Studio_ccrx) [HardwareD]	312 #endif
> 🐇 Binaries	313 315 ⊕ * @brief Flag to enable or disable provisioning mode for the demo.□
> 🔊 Includes	315 ⊕ * @brief Flag to enable or disable provisioning mode for the demo. 321 #define appmainPROVISIONING MODE (1)
> 🙀 Common	
	324 • * @brief Certificate used for validating code signing signatures in the OTA PAL.
> 🙀 Demos	326
> 🔀 Middleware	327 #define otapalconfigCODE_SIGNING_CERTIFICATE "Insert code signing certificate"
✓ 🔐 > src	328 #endif
> 🔒 application_code	329
✓ Rea > frtos_config	330 ⊖ /**
> h core_http_config.h	331 * @brief Major version of the firmware. 332 *
> Gre_mqtt_agent_config.h	332 * This is used in the OTA demo to set the appFirmwareVersion variable that is
> Is core_mqtt_config.h	334 * declared in the ota appresion 32.h file in the OTA library.
	335 */
> 🔓 core_pkcs11_config.h	336
> In core_sntp_config.h	337 #define APP_VERSION_MAJOR 0
b) defender_config.h	338 #endif
> 🕞 > demo_config.h	339
In fleet_provisioning_config.h	341 ⊕ * @brief Minor version of the firmware.
> 🕞 FreeRTOSConfig.h	346 ⊖#ifndef APP_VERSION_MINOR 347 #define APP VERSION MINOR 9
> FreeRTOSIPConfig.h	347 #define APP_VERSION_MINOR 9
> h ota_config.h	349
> 🕞 rm_littlefs_flash_config.h	356
> 🕞 shadow_config.h	357 #define APP_VERSION_BUILD 3
> 🕞 sigv4_config.h	358 #endif



6.5.1.2 Use Renesas Image Generator to Generate the Updated Firmware

Overwrite the file in the Renesas Image Generator folder with the firmware you rebuilt in **6.5.1.1** (aws_ether_ck_rx65n.mot), and then execute the following command at the command prompt:

\$ python image-gen.py -iup aws_ether_ck_rx65n.mot -ip RX65N_DualBank_ImageGenerator_PRM.csv -o user_093 -key secp256r1.privatekey -vt ecdsa -ff RTOS

This command generates a file named user_093.rsu.

6.5.2 Updating the firmware

In AWS, create an OTA update job that will update the firmware.

6.5.2.1 Creating New Job

In the IoT Core menu, select Manage > Remote actions > Jobs, and then click the Create job button.

Manage	AWS IOT > Manage > Remote actio	ns \ lobs	
 All devices 	And the state of the sector	113 / 2003	
Things Thing groups Thing types Fleet metrics Greengrass devices	operations that are frequently performed,	Such as rebooting or installing new application	are connected to AWS IoT. If you have remote ons, use job templates to create reusable jobs.
LPWAN devices	Q Filter jobs	All status values 🔻 Al	ll types 🔻 < 1 > 💿
Software packages New	Name ⊽ Type	▼ Status	▼ Created date
 Remote actions Jobs Job templates Secure tunnels 		No jobs You don't have any jobs in ap-north Create job	heast-1.
Message routing			

Figure 139. Create New Job for OTA (1/2)



6.5.2.2 Creating FreeRTOS OTA Job Update

Select Create FreeRTOS OTA update job and then click Next.



Figure 140. Create New Job for OTA (2/2)

6.5.2.3 Entering a Job Name

Enter a job name (example: rx65n_ota_demo_job) and then click Next.



Figure 141. Enter OTA Job Name



6.5.2.4 Updating Devices

Click the Devices to update drop-down list and select the device to update.

OTA file configuration Info
Devices Info This OTA update job will send your file securely over MQTT or HTTP to the FreeRTOS-based things and/or the thing groups that you choose.
Devices to update
Choose things and/or thing groups
Q
Thing groups
Things
✓ rx65n_ota_demo_thing

Figure 142. Choose Device to Update

6.5.2.5 Creating New Profile

Click Create new profile.

	devices only run code publ een changed or corrupted s		
 Sign a new file for me. 	 Choose a previously signed file. 	O Use my custom signed file.	
profile specifies your devi	formation needed to create ce's hardware platform, cert he location of your code sig	ificate from AWS	

Figure 143. Create New Code Signing Profile



You can skip steps **6.5.2.5** to **6.5.2.9** if you have already created a profile. Click **Choose existing code signing profile** and select the profile you created from the drop-down list.





6.5.2.6 Creating a Profile Naming

Create a profile (1): Profile name and device hardware platform:

Enter the profile name (example: rx65n_ota_demo_profile) Select **Windows Simulator** as the device hardware platform.

Profile name		
rx65n_ota_demo_profile		
nter a unique name without spa	ces. Valid characters: a-z, A-Z, 0-9, and _ (underscore)	
Device hardware platform		

Figure 145. Create New Code Signing (1/2)



6.5.2.7 Creating a Profile: Importing Certificate

Create a profile (2): Import a certificate.

- In the Code signing certificate area, click Import new code signing certificate.
- In Certificate body, select the file secp256r1.crt.
- In Certificate private key, select the file secp256r1.privatekey
- In Certificate chain, select the file ca.crt.
- Note: You have created above files in 6.4.
- > Click Import

 Import new code signing 	an ACM Certificate or import a third-party certificate that you cate to sign code.	
certificate		
Certificates		
Certificate body Choose file	secp256r1.crt 753 bytes O Uploaded	
X		
Certificate private key Choose file	secp256r1.privatekey 232 bytes O Uploaded	
X		
Certificate chain - optional	ca.crt 890 bytes O Uploaded	
4		
Import		

Figure 146. Create New Code Signing (2/2)



6.5.2.8 Creating a Profile: Entering Path

Create a profile (3): Enter the path of the code signing certificate of the device and then click **Create**.

You can enter any path. (Example: dummy)

Path name of code signing certificate on device		
This is the name and location of the certificate that your FreeRTOS device OTA image signature verification.	e firmware uses to perform	
dummy		
	Cancel	Create

Figure 147. Enter the Path of the Code Signing Certificate of the Device

6.5.2.9 Confirming Profile Name

Confirm that the name of the profile you created earlier is selected in the Existing code signing profile dropdown list.

hat the code hasn't been ch options for code signing.	evices only run code publish nanged or corrupted since it	ed by trusted authors and was signed. You have three	
 Sign a new file for me. 	Choose a previously signed file.	Use my custom signed file.	
pecifies your device's hardy	rmation needed to create a ware platform, certificate fro de signing certificate path or	code signing job. The profile m AWS Certificate Manager, I your device.	

Figure 148. Choose Existing Code Signing Profile



6.5.2.10 Updating the Firmware

Select Upload a new file.

In File to upload, select the file user_093.rsu you created in 6.5.1.2.

Click Browse S3 and select the S3 bucket you created in 6.3.

Enter a path name in **Path name of file on device** (You can enter any path name. Example:

/device/updates).

File	
• Upload a new file.	 Select an existing file.
File to upload Choose file user_093.rsu 427776 bytes	
File upload location in S3 This is the location in S3 where your file will be store S3 URL	ed.
Q s3://s3test-rx65n Format: s3://bucket/prefix/object.	X View 🖸 Browse S3 Create S3 bucket
Path name of file on device This is the name and location where the file will be s	stored on the FreeRTOS device.
/device/updates	
File type - optional	

Figure 149. Choose Firmware to Update

6.5.2.11 Choosing the Role for the Job

In the Role drop-down list, select the role you created in 6.3 and then click Next.

IAM role Info	
Role Choose a role that grants AWS IoT access to S3, AWS IoT jobs, and a resources.	AWS Code signing
ota_role_rx65n 🔹	
	Cancel Back Next

Figure 150. Choose Role for Creating Job



Click Create job.

OTA job configuration Info
Job run type Choose how to run this job.
 Your job will complete after deploying to the devices and groups that you chose (snapshot) Your job will continue to deploy to any devices added to the groups that you chose (continuous)
► Job start rollout configuration - optional Specify how quickly devices will be notified when a pending job starts.
Job stop configuration - optional These configurations define when to automatically stop the job. The job stops if a percentage of devices fail the deployment after a minimum number have deployed. The job cancels if any of the criteria are met after the job starts.
Job run timeout configuration - optional Specify how long the job will run.
Cancel Back Create job

Figure 151. Create Job

6.5.2.12 Waiting until Firmware Reception is Complete

Wait until firmware reception is complete.

When the job starts, the job receives and writes the firmware.

The Received counter is incremented when reception starts.

🔟 COM4 - Tera Term VT	
File Edit Setup Control Window Help	
1604 307599 [OTA Demo Ta] [INFO] Received: 66 Queued: 66 Processed: 66 Dropped: 0 1605 308169 [MQTT] [INFO] De-serialized incoming FUBLISH packet: DeserializerResult=HQTTSuccess. 1606 308169 [MQTT] [INFO] State record updated. New state=MQTTPublishDone. 1607 308169 [MQTT] [INFO] Received data message callback, size 4121.	

Figure 152. OTA Job is Processed

When the update process is complete, the device resets and the initial menu appears.

COM4 - Tera Term VT
File Edit Setup Control Window Help
2172 367335 [MQTI] [INFO] De-serialized incoming PUBLISH packet: DeserializerResult=MQTISuccess. 2173 367335 [MQTI] [INFO] State record updated. New state=MQTPublishDone. 2174 367335 [MQTI] (UARNI WARN: Received an unsolicited publish from topic \$aws/things/CanhTr_Update_OTA/jobs/AFR_OTA-rx65n_ota_demo_job_4/update/accepted 2175 367569 [MQTI] [INFO] Publishing message to aws/topic/iaq_sensor_data.
2176 367808 [MOTI] [INFO] Ack packet deserialized with result: MQITSuccess. 2177 367808 [MQII] [INFO] State record updated. New state=MQITPublishDone. 2178 367821 [OTA Deno Ia] [INFO] Received: 105 Queued: 105 Processed: 105 Dropped: 0 2179 369828 [OTA Deno Ia] [INFO] Received: 105 Queued: 105 Processed: 105 Dropped: 0 2180 369720 [MainMiSDeno] [Send Data] ZOD4510-004 : 600.000
2181 370624 [MQIT] [INFO] Publishing message to avs/topic/oaq_sensor_data. 2182 370907 [MUT] [INFO] Ack packet deserialized with result: MUTSInccess. 2183 370907 [MUT] [INFO] State prepord undated. New state=MUTFublishDone.
2184 371810 [OTA Agent T] [INFO] r_fwup_wrap_bank_swap: The startup bank = 0 2185 371834 [OTA Deno Ta] [INFO] Received: 185 Queued: 185 Processed: 105 Dropped: 0 verify install area main [sig-sha256-ecdsa]0K execute inage Press any key for going to application's setting area or after 10 second, application will run automatically
Overtime for setting, will run application now
Running sensor app with MQIT

Figure 153. Device Reset and Update New Firmware



6.5.2.13 Confirming that the Firmware Version is a New Version

Example: 0.9.3 (updated at 6.5.1.1)

COM4 - Tera Term VT
File Edit Setup Control Window Help
67 35264 [HQII] [INFO] MQII connection established with the broker. 68 35264 [HQII] [INFO] Successfully connected to HQII broker. 69 35270 [OIN Demo Tal [INFO]Start OIA Task
70 35275 [OIA Demo Ia] [INFO] OIA over MQII demo, Application version 0.9.3 71 35281 [sensor_thre] iac has a secup success
72 35281 [sensor_thre] HS3001 open sensor instance successful: 0
73 35281 [sensor_thre] ICP10101 open sensor instance successful: 0
74 35281 [ob1203_thre] OB1203 Device open success 75 35292 [sensor_thre] ICH20948 open sensor instance successful: 0
76 35440 [MainAWSDeno] [INFO]Start AVS - HQIT Demo Task
77 35463 LETHER_RECEIJ Heap: current 83216 lovest 83216
78 35465 [OTA Demo Ta] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0 29 36643 [OTA Agent T] [INFO] Current State=[RequestingJob]. Event=[Start]. New state=[RequestingJob] 80 36889 [MainAWSDeno] [INFO] Successfully subscribed to topic: aws/topic/set_temperature_led_data 81 37524 [zmod_thread] ZMOD4140 open sensor instance successful: 0
82 37756 [zmod_thread] ZHOD4510 open sensor instance successful: 0
83 37757 [zmod_thread] Task zmod4410 measurement Success:0
84 37902 [OIA Agent T] [INFO] Subscribed to topic \$aws/things/CA/jobs/notify-next.
85 37909 [OTA Agent T] [INFO] Subscribed to MQIT topic: \$aws/things/CA/jobs/notify-next 86 38730 [OTA Demo Ta] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0 87 38919 [MainAWSDeno] [INFO] SuccessFibed to topic: avs/topic/set_spo2_led_data 88 38926 [MainAWSDeno] [Send Data] ZMOD4410-IAQ TVOC: 000.000
89 38927 [MainAl/SDeno] [Send Data] ZMOD4410-JAQ ETOH: 000.000
90 38933 [MainAWSDeno] [Send Data] ZMOD4410-JAQ ECO2 : 000.000
91 39669 [MQIT] [INFO] Publishing nessage to \$aws/things/(
92 39739 [zmod_thread] ZHOD4410 in stabilization:196689
93 39921 [MQTT] [INFO] Ack packet deserialized with result: MQTTSuccess. 94 39921 [MQTT] [INFO] State record updated. New state=MQTTPublishDone. 95 39927 [OTA Agent T] [INFO] Sent PUELISH packet to broker \$aws/things/[A/jobs/\$next/yet to broker.
96 39941 [OTA Agent I] [UARM] OTA Iimer handle NULL for Timerid-Q, can't stop. 97 39951 [OTA Agent I] [UMCD] Current State=[UnitingForJob], Event=[RequestJobDocument], Nev state=[WaitingForJob] 94 40671 [MOTI Dur-serialized State=[UnitingForJob]] 99 40673 [MOTI] [UMFO] State Precord updated. New state=MOTIFublishDome. 100 40673 [MOTI] [UMFO] State Precord updated. New state=MOTIFublishDome.
101 40686 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[execution.jobId: AFR_[] 102 40699 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[execution.jobDcument.afr_pta.steanamae: AFR_OTA99-bc?6-8bf992f73d36] 103 40709 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[execution.jobDcument.afr_pta.steanamae: AFR_OTA99-bc?6-8bf992f73d36] 104 40717 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[execution.jobDcument.afr_pta.steanamae: AFR_OTA99-bc?6-8bf992f73d36] 104 40717 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[execution.jobDcument.afr_pta.steanamae: AFR_OTA99-bc?6-8bf992f73d36] 106 40731 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[filesteat]: dummy] 106 40748 [OTA Demo Ta] [INFO] Extracted parameter: [key: value]=[filesteat]: dummy] 106 40755 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[filesteat]: dummy] 108 40755 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[filesteat]: dummy] 1104 40778 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[certfile: dummy] 1104 40778 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[certfile: dummy] 1104 40778 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[certfile: dummy] 1104 40778 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[certfile: dummy] 1104 40778 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[certfile: dummy] 111 40778 [OTA Agent T] [INFO] Lextracted parameter [[sig=sha255-ccdsa: HEYC]QDQCek9mAE2Uj1Cy3hzxfdgh6V8] 112 40780 [OTA Agent T] [INFO] Image wersion number than the current image: New image version=0.9.3, Previous image version=0.9.2
113 40791 [OTA Agent T] [INFO] Image version is valid: Begin testing file: File JD=0 114 40894 [OTA Agent T] [Info] Fasing- 115 41423 [MgTT] [INFO] Publishing nessage to avs/topic/iag_sensor_data.

Figure 154. Firmware with New Version is Updated

When OTA updated firmware successfully, status will be "Succeeded" like below:

aws Services Q Search		[Alt+S]		D & Ø @
MQTT test client	Details Job execut	ions Job document Job targets	Tags	
Manage ▼ All devices	Execution overview When canceling and deleting	V execution refresh this page if the summary does not	match.	
Things	Succeeded	Failed	Canceled	Rejected
Thing groups	1	0	0	0
Thing types				
Fleet metrics	Queued	In progress	Removed	Timed out
Greengrass devices	0	0	0	0
LPWAN devices				
Software packages New				
▼ Remote actions	Job executions (1)	Info		C Cancel execution
Jobs	Devices currently processing	this job. If this job has a scheduled start time, executi	ions will queued at the start time.	
Job templates	Q. Find job executions		All job	executions (1) 🔹
Cosuro tuppols				

Figure 155. Job's Status in AWS Portal



7. Fleet Provisioning

7.1 Overview Fleet Provisioning

This section describes the steps on using Fleet Provisioning in these applications.

Fleet provisioning is a procedure in which provisioning takes place when each IoT device is started for the first time.

Generally speaking, it can be implemented in either of the following two ways.

- 1. Provisioning by claim (approach using provisioning claim certificates)
- 2. Provisioning by trusted user (mobile or web app user, etc.)

In addition, either of the following two procedures can be used to obtain the individual certificates and private keys used for fleet provisioning.

A) Having the AWS certification authority generate a new individual certificate and private key and send it to the device (CreateKeysAndCertificate).

B) Generating a key pair on the device internally and sending a certificate signature request (CSR) to AWS to have them generate only an individual certificate and send it to the device (CreateCertificateFromCsr).

This document describes the implementation of a fleet provisioning that combines 1. and B).

Advantages

- The device's private key never leaves the device.
- There is no need to establish a connection between the manufacturing plant and AWS IoT.
- There is no need to put in place a structure for issuing individual certificates or registering devices.

On the other hand, it also has the following disadvantages. It is necessary to be aware of both the advantages and the disadvantages when using this provisioning method.

Disadvantages

- It is necessary to take into account the possibility that the provisioning claim certificate could leak to an unauthorized party.
- It is necessary to implement functionality on the device to issue a provisioning request and receive a response.

For details about Fleet Provisioning, please refer to chapters 3 and 4 of the AN: <u>RX Family Provisioning</u> <u>Procedure for IoT Devices Rev.1.00 (renesas.com)</u> (Demo application for Cellular + Ethernet).

7.2 Setting up AWS for Fleet Provisioning

It is necessary to configure AWS settings in order to run the fleet provisioning demo.

- 1. Policy settings
- 2. Generating a claim certificate and claim key pair
- 3. Creating a fleet provisioning template



7.2.1 Policy Settings

Follow the steps below to create AWS IoT Core policies. The first policy you create will be used when fleet provisioning is run.





In the Policy name field, enter the policy name of your choice.

AWS IoT > Security > Policies > Create policy	
Create policy into AWS IoT Core policies allow you to manage access to the AWS IoT Core data plane operations.	
Policy properties AWS IoT Core supports named policies so that many identities can reference the same policy document.	
Policy name	
PolicyName	
A policy name is an alphanumeric string that can also contain period (), comma (), hyphen(-), underscore (), plus sign (+), equal sign (o), and at sign (@) characters, but no spaces.	
► Tags - optional	
Policy statements Policy examples	
Policy document info An AWS isoT policy contains one or more policy statements. Each policy statement contains actions, resources, and an effect that grants or denies the actions by the resources.	Builder JSON
Policy document	
<pre>1 v { "Version": "2012-10-17", "Statement": [4 {</pre>	
Copy and paste the text of the policy document here.	
Copy and paste the text of the policy document here.	
JSON Line 1, Column 1 ③ Errors: 0 🛆 Warnings: 0	
	Cancel

Figure 157. Creating an AWS IoT Policy (2/2)



Click the JSON button to display the policy document input field, then copy and paste the policy document shown in **Table 11**. **Policy Document** into the input field. When copying and pasting the policy **Policy Document**, make the following changes:

• Change "us-east-1" to match the region used.

• Change <account id> to your own account ID (account ID is the 12-digit number after @ that is displayed by clicking on the account name in the upper right corner, excluding the hyphen)

 Table 11.
 Policy Document

```
"Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
"Action": "iot:Connect",
      "Resource": "*"
    {
      "Effect": "Allow",
      "Action": [
        "iot:Publish",
        "iot:Receive",
       "iot:RetainPublish"
      1.
      "Resource": [
        "arn:aws:iot:us-east-1:<account id>:topic/$aws/certificates/create-from-csr/*",
        "arn:aws:iot:us-east-1:<account id>:*"
      ]
    },
    ł
      "Effect": "Allow",
      "Action": "iot:Subscribe",
      "Resource": [
        "arn:aws:iot:us-east-1:<account id>:topicfilter/$aws/certificates/create-from-csr/*",
        "arn:aws:iot:us-east-1:<account id>:*
      1
    }
 ]
}
```

7.2.2 Generating a Claim Certificate and Claim Key Pair

Generate a provisioning claim certificate and provisioning claim key pair for use in fleet provisioning.

Select Security \rightarrow Certificates and then click Add certificate \rightarrow Create certificate.



Figure 158. Create a Certificate



Certificates authenticate devices and clients so that they can IoT without authentication and an appropriate policy.	connect to AWS IoT. Your device won't be able to connect to AWS
Certificate	
 Auto-generate new certificate (recommended) Generate a new certificate, public key, and private key using AWS IoT's certificate authority and register it with AWS IoT. 	Create certificate with certificate signing request (CSR) Upload your own certificate signing request (CSR) file to create and register a certificate that's based on a private key you own.
Certificate status	
	e active before it can be used to connect to AWS IoT. You can change its
 Inactive A device won't be able to connect to AWS using this certificate it 	until it's activated.
O Active A device will be able to connect to AWS using this certificate im	mediately after you create it.
	Cancel Create

Figure 159. Creating a Certificate Automatically

Download the newly created certificate (1) and key pair (2)(3), then click the **Continue** button.

Download certificates and keys	×
Download certificates and keys	
Download and install the certificate and key files to your device so that it can connect so IoT. You can download the certificate now, or later, but the key files can only be download	
Device certificate	vnload te downloaded
Key files The key files are unique to this certificate and can't be downloaded after you leave this p Download them now and save them in a secure place.	lage.
A This is the only time you can download the key files for this certification	te.
	ownload
	ownload
Root CA certificates	
Download the root CA certificate file that corresponds to the type of data endpoint and you're using. You can also download the root CA certificates later.	cipher suite
Amazon trust services endpoint	ownload
RSA 2048 bit key: Amazon Root CA 1	
Amazon trust services endpoint	ownload
ECC 256 bit key: Amazon Root CA 3	
If you don't see the root CA certificate that you need here, AWS IoT support root CA certificates. These root CA certificates and others are available from developer guides.	
•	Continue

Figure 160. Downloading the Certificate and Key Pair





Figure 161. Certificate Settings

Click Actions \rightarrow Activate to activate the certificate. Also click the Attach policies button.

		Activate Dealtivate
Details		Revoke
iertificate ID	Status	Accept transfer
	⊖ Inactive	Reject transfer
ertificate ARN	Created	Start transfer
	January 08, 2024, 15:00:27 (UTC+07:00)	Attach policy
ubject	Valid	Attach to thing
N=AWS IOT Certificate	January 08, 2024, 14:58:27 (UTC+07:00)	Download
		Delete
isuer IU=Amazon Web Services O=Amazon.com Inc. L=Seattle ST=Washington C=US	Expires January 01, 2050, 06:59:59 (UTC+07:00)	
tolicies Things Noncompliance		
Policies (0) Info		C Detach policies Attach policies
WS IoT policies allow you to control access to the AWS IoT Core data plane operations.		
Name		

Figure 162. Certificate Settings: Attach Policies (1/2)

Clicking the Attach policies button opens the dialog box shown in Figure 163.

Select the policy to be used when fleet provisioning is run, created in **7.2.1**, Policy Settings, and then click the **Attach policies** button to attach it to the certificate. Example, the name of policy, which is created for Fleet demo, is "**Fleet_policy**".

This completes the settings related to generation of the claim certificate and claim key pair.



Attach policies to the certificate	X 457750 X 257 X X 37
Policies Choose policies to attach to this certificate. The certificate can h Choose AWS IoT policy	nave up to 10 policies attached to it.
C C C C C C C C C C C C C C C C C C C	ancel Attach policies

Figure 163. Certificate Settings: Attach Policies (2/2)

7.2.3 Creating a Fleet Provisioning Template

Select Connect many devices \rightarrow Connect many devices, then click the Create provisioning template button.

Monitor	★ How it works		
Connect Connect one device Connect many devices Bulk registration		Q Q Q Q Q Q Q Q Q Q Q Q Q Q Q	
Test > Device Advisor MQTT test client Device Location New	Step 1. Determine providenting searching Devices need a unique certificate to connect to AWS 10 Hoursen install this certificate during the devices annual/scure, when the device is provisions by a authenticated case, or by installing a claim certificate duris endanged for a unique device certificate the first time the device connects to AWS IoT. Learn more C	Step 2. Define device management structure Connected devices are represented in <i>NWS</i> is 10 thing resources, which help you organize, manage, and maincing word devices. Thing resources, thing groups, thing synchronized attributes, and billing props, size help you manage your devices and can also be created intervalue device is provisioned. Learn more 🕑	Step 3. Create a provisioning template Approxidening template is a JSDN document that describes the resources, policies and permissions to create for the device when it's provisioned. Learn more C
Manage > All devices > Greengrass devices	Connect many devices (1) tota To connect many devices, the provisioning kernalistic automates the provisioning required to connect	There divises.	Activate Deactivate Delete Create provisioning template

Figure 164. Creating a Provisioning Template (1/7)

Select Provisioning devices with claim certificates, then click the Next button.



Figure 165. Creating a Provisioning Template (2/7)



CK-RX65N SIM activation, Creating the trial account and using Dashboard with RYZ014A or Ethernet Application for AWS - Getting Started Guide

On the template creation screen, specify the provisioning template status, template name, and provisioning role. For **Provisioning template status** select **Active** and enter the name of the provisioning template. Then click the **Create new role** button and enter the role name.

P	Provisioning template properties Info
T	rovisioning template status he provisioning template status determines whether the template can be used to provision a new device. Only active templates can rovision devices.
C	Inactive Inactive templates can't provision any devices that are configured to use it. You can create an inactive template to prevent devices from being provisioned until you're ready.
C	Active An active template can provision the devices that are configured to use it.
Р	rovisioning template name
Γ	Enter_template_name
Т	he name can have up to 36 characters and must not costain spaces. Valid characters: A-Z, a-z, 0-9, and _ (underscore) and - (hypher
D	vescription - optional
	A description of the provisioning template you're creating.
5	00 character remaining
	rovisioning role he provisioning role uses an IAM role that authorizes AWS IoT to access resources on your behalf.
Г	Choose an IAM role 🔹 C View 🔀 Create new role

Figure 166. Creating a Provisioning Template (3/7)



For **Claim certificate policy**, select the policy to be used when fleet provisioning is run, created in **7.2.1**. For **Claim certificate**, select the certificate created in **7.2.3**, and click the **Next** button.

Claim certificate provisioning policy Choose the AWS IoT policy that authorizes	the claim certificate to connect and provision the IoT device. This policy is attached to the claim
certificates you choose in the next section.	
Fleet_policy	▲ C View C Create IoT policy C
Q	
Fleet_policy	~
in a limited number of lol dev	
	ces limits your exposure in case a claim certificate is compromised.
Claim certificates - optional Choose the claim certificates to attach the	
Claim certificates - optional Choose the claim certificates to attach the	(1/15) Info policy to, or attach the policy later by editing the provisioning template's provisioning initiator. he claim certificate provisioning policy attached.
Claim certificates - optional Choose the claim certificates to attach the Claim certificates must be active and have	(1/15) Info policy to, or attach the policy later by editing the provisioning template's provisioning initiator. he claim certificate provisioning policy attached.

Figure 167. Creating a Provisioning Template (4/7)

For **Pre-provisioning actions**, select **Don't use a pre-provisioning action**. Also, under **Automatic thing creation**, turn on **Automatically create a thing resource when provisioning a device**, and if necessary, enter a character string of your choice as the thing name prefix. The thing name registered with AWS will be generated from this character string and the serial number set by the program. After entering the prefix, click the **Next** button.

Note: The demo does not use pre-provisioning actions. Refer to the page linked to below for information on using pre-provisioning actions: Using pre-provisioning hooks with the AWS CLI





Figure 168. Creating a Provisioning Template (5/7)



For **Set device permissions**, check the box next to the policy attached to newly created things, which was created in **5.4.4**, then click the **Next** button.

Set device permissi	ONS Info		
AWS IoT policies authorize devices to a Shadows.	iccess AWS IoT resources such as o	ther thing resources, MQTT to	pics, and Device
Policies (1/5) Info Choose up to 10 policies to attach to this	. certificate.	C	Create policy 🖪
Q Find policies			< 1 > 💿
Policy name	▼ ARN		∇
	đ		·-····
	đ		·
	đ		
	đ		
✓ <u>ck_rx65n_test</u>	đ	1 1 0 47C01F 4F 7 40	<u> </u>
		Cancel Pr	evious Next

Figure 169. Creating a Provisioning Template (6/7)

Click the **Create template** button to complete the process of creating a fleet provisioning template.

Policies			
Policies			
Policy name	Policy action	Policy	effect
ck_rx65n_test 🔼	iot:Connect	Allow	
	iot:Publish	Allow	
	iot:Subscribe	Allow	
	iot:Receive	Allow	
Without a certificate p	er exists in this account provider, AWS IoT Core will sign all certif sign CSRs using a certificate provider, ch		



7.3 Setting up the Project

Change the value of **ENABLE_FLEET_PROVISIONING_DEMO** to **1** in **aws_ether_ck_rx65n/src/frtos_config/demo_config.h**:

h demo_config.h	×
	', statet sens companycaun to runn ;
76	
77	⊖ /* Please select a provisioning method
78	* (0) : Pre-provisioning
79	* (1) : Fleet provisioning
80	*/
81	#define ENABLE_FLEET_PROVISIONING_DEMO (1)
0.2	

Figure 171. Enable Fleet Provisioning macro

Build and Debug the project as described in Topic: 5.1.3 Importing the Project (number 9, 10)

7.4 Running Fleet Provisioning

- Note: User can run both Fleet Provisioning and OTA (**Section 6**) in this application. This section only describes Fleet Provisioning feature.
- 1. After loading debugging to board, press any key to configure the application. (in case users have not config Cloud's credential for Fleet Provisioning before, or users want to save another value for Cloud's credential)



Figure 172. Save credentials for Fleet Provisioning (1/8)



Figure 173. Save credentials for Fleet Provisioning (2/8)



CK-RX65N SIM activation, Creating the trial account and using Dashboard with RYZ014A or Ethernet Application for AWS - Getting Started Guide

Press '2' to choose "2. Data flash" > press 'l' to choose "I) Format Flash data" to erase all stored information in flash (if application was run previously) (optional)





Different from normal provisioning, Fleet Provisioning requires credentials: MQTT broker endpoint, fleet template name, claim cert ID, claim private key ID.

3. Get MQTT endpoint and save to flash like topic 5.4.6 and 5.3.1:



Figure 175. Save credentials for Fleet Provisioning (4/8)



 Next, storing fleet template name that user created at 7.2.3, copy this value, press the option 'g' and click the Edit tab of the Tera Term, choose "Paste<CR>", verify and confirm the valid string then press OK.



Figure 176. Save credentials for Fleet Provisioning (5/8)

Next, storing fleet claim cert ID, claim private key ID that users created at 0, users downloaded (Figure 160. Downloading the Certificate and Key Pair):



Figure 177. Save credentials for Fleet Provisioning (6/8)



CK-RX65N SIM activation, Creating the trial account and using Dashboard with RYZ014A or Ethernet Application for AWS - Getting Started Guide

From "2. DATA FLASH" menu, press 'h' to save Claim cert ID. Then click the File tab of the Tera Term and Send File option and choose the downloaded Device certificate file "xxxxxcertificate.pem.crt".

File Edit Setup Control Window Help	
DATA FLASH WRITE CLAIMCERT ID FLEET	
Select the file to write data in data flash \$[]	
🔟 Tera Term: Send file	×
Look in: 📙 fleet_cert 🗸 🎸 🎯 🏂 🔛	
^	
-certificate.	.pem.crt
-private.per	
-public.pen	n.key
<	>
File name:certificate.pem.crt (Open
Files of type: All(*.*) V	Cancel
la	Help

Figure 178. Save Credentials for Fleet Provisioning (7/8)



Figure 179. Save Credentials for Fleet Provisioning (8/8)

To store the **Claim private key ID**, press the option 'i' and click the **File** tab of the Tera Term and **Send File** option, choose "**xxxxxprivate.pem.key**".

6. Start application with Fleet Provisioning:

Back to the Application Menu and press 'Run Sensor App with MQTT' to run application:





Figure 180. Running Application with Fleet (1/8)

Application will check stored data flash. When it has enough necessary credentials, application will start. Otherwise, the application will send a notice to the user and the user has to save the lacking credentials for running application.

```
🔟 COM4 - Tera Term VT
File Edit Setup Control Window Help
CHECK CREDENTIALS STORED IN DATA FLASH
Claim private key ID saved in data flash is verified and successful
Claim cert ID saved in data flash is verified and successful
template name saved in data flash is verified and successful
OTA is disabled, do not need code sign certificate
Fleet is enabled, IOT thing name is not available, it will be updated by Fleet
MQTT Endpoint saved in data flash is verified and successful
Fleet is enable, do not need Private Key (2b, 2c section)
Fleet is enable, do not need Certificate (2b, 2c section)
All credentials in data flash is verified and successful
197727 [IP-Task] prvIPTask started
 197729 [ETHER_RECEI] Deferred Interrupt Handler Task started
 197729 [ETHER_RECEI] Queue space: lowest 8
3 197729 [IP-Task] InitializeNetwork returns OK
4 197729 [IP-Task] xNetworkInterfaceInitialise returns Ø
5 197829 [ETHER_RECEI] R_ETHER_Read_ZC2: rc = -5
6 197929 [ETHER_RECEI] R_ETHER_Read_ZC2: rc = -5
 198029 [ETHER_RECEI] R_ETHER_Read_ZC2: rc = -5
 198129 [ETHER_RECEI] R_ETHER_Read_ZC2: rc = -5
 198229 [ETHER_RECEI] R_ETHER_Read_ZC2: rc = -5
10 198329 [ETHER_RECEI] R_ETHER_Read_ZC2: rc = -5
```

Figure 181. Running Application with Fleet (2/8)



CK-RX65N SIM activation, Creating the trial account and using Dashboard with RYZ014A or Ethernet Application for AWS - Getting Started Guide

🔟 COM4 - Tera Term VT
File Edit Setup Control Window Help
41 216763 [IP-Task] Subnet Mask: 255.255.255.0
42 216763 [IP-Task] Gateway Address: 192.168.3.1
43 216763 [IP-Task] DNS Server Address: 8.8.8.8
44 216769 [ETHER_RECEI] Network buffers: 1 lowest 1
45 217005 [CLI] Initialise the RTOS's TCP/IP stack
46 217005 [CLI]STARTING DEMO
47 217009 [DemoTask] [INFO]Start Fleet Provisioning Task
48 217016 [ETHER_RECEI] Heap: current 186344 lowest 182896
49 217195 [DemoTask] [INFO] Establishing MQIT session with claim certificate 50 217202 [DemoTask] [INFO] Using default rootCA cert. 51 217202 [DemoTask] [INFO] Create a TCP connection to a transmission at straight at st
53 217251 [DemoTask] DNS[ØxDDDC]: The answer to 'ats.iot.us-east-1.amazonaws.com' <52.55.136.67> will be stored
54 217262 [DemoTask] FreeRIOS_connect: 54649 to 34378843ip:8883
55 217266 [IP-Task] prvSocketSetMSS: 1400 bytes for 34378843ip:8883
56 217267 [IP-Task] prvWinScaleFactor: uxRxWinSize 2 MSS 1400 Factor 0
57 217267 [IP-Task] Connect[34378843ip:8883]: next timeout 1: 3000 ms
58 217502 [IP-Task] MSS change 1400 -> 1452
59 217502 [IP-Task] TCP: active 54649 => 34378843ip:8883 set ESTAB (scaling 1)
60 217513 [DemoTask] [INFO] Established TCP connection with a standard ats.iot.us-east-1.amazonaws.com. 61 217558 [ETHER_RECEI] Heap: current 126208 lowest 126064
62 218120 [ETHER_RECEI] Heap: current 112216 lowest 110064
63 228068 [DemoTask] [INFO] (Network connection 804d14) ILS handshake successful. 64 228075 [DemoTask] [INFO] (Network connection 804d14) Connection to a stat, iot.us-east-1.amazonaws.com established. 65 228362 [DemoTask] [INFO] MQII connection established with the broker. 66 228362 [DemoTask] [INFO] MQII connection successfully established with broker.
67 228375 [DemoTask] [INFO] Established connection with claim credentials. 68 228376 [DemoTask] [INFO] SUBSCRIBE topic \$aws/certificates/create-from-csr/cbor/accepted to broker.
69 228653 [DemoTask] [INFO] MQTT_PACKET_TYPE_SUBACK.
70 225575 [DemoTask] [INFO] SUBSCRIBE topic \$aws/certificates/create-from-csr/cbor/rejected to broker.
71 225842 [DemoTask] [INFO] MQTT_PACKET_TYPE_SUBACK.
72 238761 [DemoTask] [INFO] the published payload:ixcertificateSigningRequesty]BEGIN CERTIFICATE REQUEST
HII

Figure 182. Running Application with Fleet (2/8)

If the text string "**Demo completed successfully.**" appears at the end of the log, the fleet provisioning demo completed successfully. Successful completion of the demo means that a new thing has been registered on AWS IoT Core and an individual device certificate assigned to it.

💆 СОМ4 -	Tera Term VT		
File Edit	Setup Control	Window H	leip
128 2745	83 [DemoTask	1 EINFOl 🤉	Sucessfully established connection with provisioned credentials. (Network connection 804d14) TLS close-notify sent. active socket on port 5706 (34378843ip:8883)
131 2748	25 [DemoTask	I [INFO]]	Demo iteration 1 is successful. Demo completed successfully.

Figure 183. Running Application with Fleet (3/8)



After running the fleet provisioning demo, users can use the individual device certificate and private key obtained from AWS to run the MQTT with sensors demo:

152 275479 [MOTT] [INFO]	Established TCP connection with a ats.iot.us-east-1.amazonaws.com.
153 276944 [MQTT] [INF0] 154 276944 [MQTT] [INF0]	(Network connection 895888) TLS handthake successful.
155 276944 [MQTT] [INFO] 156 277229 [MQTT] [INFO] 157 277229 [MQTT] [INFO]	Greating an MOII connection to the broker. MOII connection established with the broker. Successfully connected to MOII broker.
158 277235 [sensor_thre]	12C bas 6 setup success
159 277236 [sensor_thre]	H\$3001 open sensor instance successful: 0
160 277236 [sensor_thre]	ICP18181 open sensor instance successful: 0
161 277236 [ob1203_thre] OB1203 Device open succe 162 277245 [sensor_thre]	ss ICH20948 open sensor instance successful: 0
163 277288 [MainAWSDemo]	[INFO]Start AVS - MQIT Demo Task
164 278738 [MainAWSDemo] 165 279492 [zmod_thread]	[INFO] Successfully subscribed to topic: aws/topic/set_temperature_led_data ZMOD4418 open sensor instance successful: 0
166 279722 [zmod_thread]	ZMOD4510 open sensor instance successful: 0
167 279723 [zmod_thread]	Task zmod4410 measurement Success:0
168 279741 [MainAWSDemo] 169 279748 [MainAWSDemo]	[INFO] Successfully subscribed to topic: aws/topic/set_spo2_led_data [Send Data] ZMOD4418-IAQ TVOC: 000.000
170 279750 [MainAWSDemo]	[Send Data] ZM0D4410-16Q ETOH: 000.000
171 279756 [MainAWSDemo]	[Send Data] ZM0D4418-1A9 EC02 : 000,000

Figure 184. Running Application with Fleet (4/8)

Users can check on the thing registered by the fleet provisioning demo from AWS IoT console. When running fleet provisioning successfully, please check the log in Tera Term with line: "**Received AWS IoT Thing name: FleetFPDemoID_xxxxxxxxx**"



Figure 185. Running Application with Fleet (5/8)

Note: Prefix **Fleet** is the name that is set by user when creating Fleet template.

Under **All devices**, select **Things**. The registered thing is shown below (It will have a name that matches the "Received AWS IoT Thing name" shown in **Figure 185**):

AWS IoT	X AWS IOT > Manage > Things
Monitor	Things (7) Info An IoT thing is a representation and record of your physical device in the cloud. A physical device
Connect Connect one device	needs a thing record in order to work with AWS IoT. Q Filter things by: name, type, group, billing, or searchable attribute.
Connect many devices	Name FleetFPDemoID
Test	
Device Advisor	
MQTT test client	
Device Location New	
Manage	
 All devices 	
Things	

Figure 186. Running Application with Fleet (6/8)



By checking the registered things, user can confirm that the individual device certificate generated and assigned by fleet provisioning has been attached and activated:

AWS IoT	×	AWS IoT > Manage > Things > FleetFPDemoID_	-
Monitor		FleetFPDemoID	Info
Connect		Thing details	
Connect one device Connect many devices 		Name FleetFPDemoID_t	Type -
Test ▶ Device Advisor MQTT test client Device Location New		ARN D arn:aws:iot:us-east- 1: thing/FleetFPDemoID	Billing group -
Manage		Attributes Certificates Thing groups Device Shadows Activity	Packages and versions Job
 All devices Things Thing groups Thing types 	l	Certificates (1) Info The device certificates attached to this thing resource. Q. Find certificates	
Fleet metrics		Certificate ID	Status
 Greengrass devices LPWAN devices 		881	⊘ Active

Figure 187. Running Application with Fleet (7/8)

This Certificate ID will be shown in Tera Term with line "Received certificate with Id: xxxxxxxxxxxxxxxxxxx.".

79 231141 [DemoTask] [INFO] De-serialized incoming PUBLISH packet: DeserializerResult=MQTTSuccess. 80 231151 [DemoTask] [INFO] State record updated. New state=MQTTPubAckSend. 81 231153 [DemoTask] [INFO] Received acceted response from Fleet Provisioning CreateCertificateFromCsr API. 82 235853 [DemoTask] [INFO] Received certificate with Id: 8815842c293795519d84b2a2e649b 83 235863 [DemoTask] [INFO] Writing certificate into Iabel "Device Cert". 84 235908 [DemoTask] [INFO] UNSUBSCRIBE sent topic \$aws/certificates/oreate-from-csr/cbor/accepted to broker.

Figure 188. Running Application with Fleet (8/8)

8. Note and Troubleshooting

8.1 Sensor Stabilization Time

Each sensor has a stabilization time during which correct values cannot be read.

The following table shows the details of sensor stabilization times.

Table 12. Sensor Stabilization Time

Sensor Name	When power up first time	After soft or hard reset
ZMOD4410 IAQ	Up to 1 minutes	Up to 1 minute
ZMOD4510 OAQ	Up to 1.5 hours	Up to 1 hour
OB1203	Up to 20 minutes	Up to 20 seconds
	(After putting finger on sensor, it may take up to 60 seconds to sense data)	(After putting finger on sensor, it may take up to 60 seconds to sense data)
HS3001	Up to 30 seconds	Up to 10 seconds
ICP	Up to 30 seconds	Up to 10 seconds
ICM	Up to 30 seconds	Up to 10 seconds



8.2 Connection Issue When Using Ethernet (Wired cable)

The Ether PHY only supports full-duplex communication. If your router or Ethernet hub only supports half duplex, it cannot connect to the internet. Please use full duplex devices.

8.3 Current Supply Short Issue When Using RYZ014A

If the CK-RX65N V1 board is not powered through the Debug port (J14), the current available to the board may be limited to 100mA. When using the supplied RYZ014A Pmod module with other code (found here: <u>RYZ014A - LTE Cat-M1 Cellular IoT Module | Renesas</u>) be aware that this Pmod has a maximum operating current of **480 mA** dependent upon the LTE band, Tx/Rx settings, and network coverage. Please ensure that the host board can supply sufficient power or provide supplemental USB power via CN4 on the Pmod to avoid RF instability.

8.4 When Build Errors Occur

If a 'No such file or directory' error occurs, the project path may be too long. When the path is longer than 256 characters, e² studio outputs errors at build time.

When this error occurs, move the project to a shorter path location (such as, under C:\).

8.5 When Unable to Log in to the Dashboard (Grafana Account)

If you cannot log in to the Dashboard with the password you changed in step 6 of section **5.2.2 Getting the Account 10 USD for Trial of AWS**, try the following:

Set "admin" in the Email or username field and set the changed password in the password field.

When changing the password for the initial session, the username is not changed from admin. Therefore, admin must be entered in the username field. To enable users to log in with their own username and email address, please change the user information in the Server Admin menu after logging in.

8.6 Notes on Performing Firmware Update Over-The-Air on AWS FreeRTOS

The following symptoms occur when performing Firmware Update Over-The-Air (FOTA) on AWS FreeRTOS.

- 1. Device (thing) names and credential information are overwritten.
- 2. Processing of OTA fails after recovering from low power consumption mode.

For more information, refer to the Tool News at:

[Notes] RX Family Notes on Performing Firmware Update Over-The-Air on AWS FreeRTOS (renesas.com)

8.7 When the Trial 10 USD is Used Up

When the trial 10 USD is used up, status will change to "Quarantined". At this time, the user cannot download the certificate or go to Dashboard.



Figure 189. Account Used Up Trial 10 USD

Note: Please add payment options before the account is quarantined and enable the option on the dashboard. If the account is quarantined, please contact Renesas Customer Support.



To enable option on the dashboard, go to the user profile.

My Devices	FRA		
Device 1		Status: Provisioned	to a second
			Profile Support Page
			Reset Grateme's password

Figure 190. Choose Profile

Choose "Yes" for the Payment Preference Update Confirmation.

RENES	5	
My Devices	RA MAN RX	9
	Profile Page First name:	

Figure 191. Payment Confirmation

To add a credit card to continue using this account or using another AWS account, go to <u>https://cloud-ra-rx.awsapps.com/start#/</u>, sign up with AWS account.

	aws
[Sign in to cloud-ra-rx
	Username
	Next
	By continuing, you agree to the AWS Customer Agreement or other agreement for AWS services, and the Privacy Notice . This site uses essential cookies. See our Cookie Notice for more information.

Figure 192. Sign up to AWS Account



Go to Manage Console > Account > AWS Billing > Payment preferences > Add payment method.

aws	MFA devices Sign out
AWS Account (1)	Q Search
ra-rx-20032500-37505052-00000000000000000000000000000000	32-befa4e36-4e4b292d@dm.renesas.com Management console Command line or programmatic access



Resource Groups & Tag Editor Console Home Info			Account ID: 3005 September 20 C
Recently visited Info		1	Welcome to AWS
Billing	AWS Cost Explorer		Organization Getting started with A Service Quotas
loT Core			Learn the fundamentals a information to get the m Billing Dashboard
AWS Organizations	AWS Organizations		
Lambda		-6	Training and certificat Switch role Sign
IAM Identity Center Image: CC Image: S3			Learn from AWS experts skills and knowledge.
			What's new with AWS? Discover new AWS services, features, and Regions.

Figure 194. Add Credit Card for Trial Account (2/4)

Home	Payment preferences Info				
Billing	Default payment preferences info				
Bills					
Payments					
Credits	Default payment method does not exist. Add a default payment method and reload.				
Purchase orders					
Cost & usage reports					
Cost categories	Payment methods Payment profiles				
Cost allocation tags					
Free tier					
Billing Conductor 🖸	Payment methods (0) Iefo Set as default Edit Delete Add payment method				
Cost Management	Q. Find payment method				
Cost explorer					
Budgets	No payment methods				
Budgets reports	There are no payment methods on file. Add a new payment method.				
Savings Plans 🖸	inere are no payment ineurops on mir. And a new payment method,				
Preferences					
Billing preferences					
Payment preferences					
Consolidated billing 🖸					
Tax settings					

Figure 195. Add Credit Card for Trial Account (3/4)



aWS iii Services Q iii Resource Groups & Tag Edito	ж	[Alt+5]	
Billing	×	AWS Billing > Payment preferences > Add payment method	
Home		Add payment method Info	
▼ Billing Bills		Choose a payment method type Info	
Payments Credits Purchase orders Cost & usage reports Cost categories Cost allocation tags		with at least one fully payment currencies ca	his payment method. Accounts paid monthly invoice with USD an add bank account as a payment age your payment currency to USD
Free tier Billing Conductor 🖸		Card information Info	
▼ Cost Management Cost explorer 🗗 Budgets Budgets reports Savings Plans 🗗		Expiry date	
 Preferences Billing preferences Payment preferences 		MM/YY Use MM/YY format. Name on card	
Consolidated billing 🖸 Tax settings Permissions		Set as default payment method	
Affected policies 🗹		Billing address Info	Use existing address V

Figure 196. Add Credit Card for Trial Account (4/4)

8.8 How to Enable/Disable EC2 Instance

AWS trial accounts start billing immediately after device registration. The following steps will disable the EC2 instance, saving the user's credits.

Access AWS account from <u>https://cloud-ra-rx.awsapps.com/start#/</u> using the dashboard credentials. (Refer to **Figure 192** to log in).

To disable the EC2 instance, use the following steps:

1. From the **Services** menu, select **Compute** and then choose **EC2**.



Figure 197. EC2 AWS Service



2. Choose the instance, then change the Instance state to Stop instance

aws III Services	Q Search	[Alt+S]	🖸 😓 🦁 🕘 N. Virginia 🔹 🛛 Alvan 🗤 👘
EC2 Dashboard EC2 Global View Events Console-to-Code Proteov Instances Instance Types Launch Templates Spot Requests Savings Plans	×	Instances (1/1) infe Q. Find Instance by attribute or tog (cose-sensitive) 2 Name ∠ ▼ Instance ID Instance Live T Instance type 2 Genter Cost of the cost of t	C Connect Instance state A Actions V Stop instance
Reserved Instances Dedicated Hosts			

Figure 198. Disable Instance

Note: If you wish to use the dashboard again, please enable the EC2 instance before using it.

8.9 How to check the total amount spent in AWS account

Access AWS account from <u>https://cloud-ra-rx.awsapps.com/start#/</u> using the dashboard credentials. (Refer to the **Figure 192** to login).

Go to Account > Bills.

User can choose the "Billing period" to see the amount spent during that period:

aws Services Q Search	[A][1+S] D 🖉 Ø Global 🕶 AWSAdministratorAccess.	
Billing and Cost ×	Billing and Cost Management > Bills	6
Home Getting Started	Page refresh time: Thursday, May 9, 2024 at 3:56:47 PM GMT+7	
Billing and Payments Bills Payments	AWS bill summary info Total charges and payment information	
Credits Purchase Orders	Account ID Billing period Info 370 February 1 - February 28, 2023	
Cost Analysis Cost Explorer Cost Explorer Saved Reports	Service provider Total in USD Amazon Web Services, Inc. USD 8.88	
Cost Anomaly Detection Free Tier Data Exports	Grand total: USD 8.88 Payable by Account ID: 18	

Figure 199. Check the amount spent in AWS Account

8.10 An error occurs when connecting to AWS

The AWS IoT information is not set yet or is set incorrectly. Please check and set AWS IoT information again. (5.3.1)

8.11 Command to create the initial firmware fails (OTA)

The cause of this issue is the Python installation folder is not set correctly in the Path variable or the encryption library is not installed.

Users have to re-install Python. Also, make sure that the Add python.exe to PATH check box is selected when you perform the step in **6.2.1 Installing Python** and install the encryption library.

8.12 Initial firmware cannot be written/ does not start. (OTA)

Make sure that the jumper on J16 of CK-RX65N board is on pins 1-2 (debug mode) when writing initial firmware and on pins 2-3 (run mode) when starting the initial firmware.

8.13 Firmware does not start after starting the boot loader (OTA)

Please review the public key setting in the bootloader because it is not correctly set in the boot loader.



8.14 Firmware does not start after an OTA update (OTA)

Users can review the public key setting in the firmware because the public key is not set correctly in the firmware. If not, please review the device setting in the firmware and the boot loader.



9. Website and Support

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

CK-RX65N V1 Kit Information RX&RA Cloud Solutions RX Cloud solution web RX Product Information RX Product Support Forum RX Driver Package Renesas Support renesas.com/rx/ck-rx65n renesas.com/cloudsolutions renesas.com/rx-cloud renesas.com/rx renesas.com/rx/forum renesas.com/RDP renesas.com/support



CK-RX65N SIM activation, Creating the trial account and using Dashboard with RYZ014A or Ethernet Application for AWS - Getting Started Guide

Revision History

		Description		
Rev.	Date	Page	Summary	
1.00	Jun.14.22		First version	
1.10	Jun.29.22	15	Improved 5.1.4 Activating a SIM card on MicroAl Launchpad of activation way of SIM	
		49	Added the restrictions section	
		Program	Fixed the program about getting the issue of UUID lacking parts of the number.	
1.20	Jul.06.22	10-11	Added 8. Apply Patch files	
		50	Removed "SpO2 sensor of OB1203 (Cellular version of Project)" from the restriction section. Updated 5.1 Communication time (Cellular version of Project).	
		Program	Updated Cellular version of Project - Supported SpO2 sensor of OB1203. - Added patch file to project.	
1.30	Jun.02.23	11	Added the settings for Truphone SIM	
		26	Added 4.1.6 about the activation procedure for Truphone SIM	
		50	Added 6.4 about when build errors occur.	
			Added 6.5 about when unable to log in to the Dashboard (Grafana account)	
			Added 6.6 Notes on Performing Firmware Update Over-The- Air on AWS FreeRTOS	
1.31	Jan.19.24	—	Fix typo mistake	
		5	Updated Figure 2	
		5	Updated description for 4. Connection to AWS section	
		11, 19	Removed information of MicroAl Sim card, added note about SIM Card's information	
		—	Removed 4.1.5.2 Activate a SIM card on MicroAl Launchpad Section due to the MicroAl SIM card is discontinued to support CK-RX65N.	
		44	Correct description for "LED Alerts" Section	
		45	Added "Automatic alert from Renesas Dashboard feature" into Restrictions section	
		23-26, 33-40, 46-50	Updated information about latest Renesas Dashboard (add payment, enable/disable EC2 instance,), AWS	
1.40	Jun.24.24	49-106	Added OTA feature (section 6), Fleet Provisioning feature (section 7)	
		6-8	Add section "4. Cloud Connectivity Application Example": Add application overview.	
		9-48	Re-arrange section "5. Connection to AWS"	
		110	Add restrictions (8.9 \rightarrow 8.14) in "Note and Troubleshooting"	



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 is supplied until the 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

- 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.
- Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.

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

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

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

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

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu, Koto-ku, Tokyo 135-0061, Japan

www.renesas.com

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/.