

RX65N Group

Sample Code for OTA Update of a Secondary Device by Amazon Web Services with the Use of FreeRTOS

Introduction

This application note is for a system in which an RX65N microcontroller is used as a primary MCU that communicates with Amazon Web Services[™] (hereafter, referred to as "AWS") and an RX microcontroller is used as a secondary MCU that receives data measured by sensors. This application note describes a demonstration where AWS services are used to perform an over-the-air (OTA) update of the secondary MCU (hereafter, referred to as "secondary OTA update").



Devices Used in Confirming Operation

- Primary MCU: RX65N
- Secondary MCU: RX140
- Temperature and humidity sensor: HS3001 high-performance relative humidity and temperature sensor

Boards Used in Confirming Operation

- Primary MCU: CK-RX65N v2 (RTK5CK65N0S04000BE)
- Secondary MCU: FPB-RX140 v1 (RTK5FP1400S00001BE)
- Temperature and humidity sensor: Relative Humidity Sensor Pmod[™] Board (US082-HS3001EVZ)



Related Documents

This application note refers to and further explains information in the following documents. Updating of a document may lead to changes to the structure of chapters and other items. Take care on this point when referring to the following documents.

RX Family Firmware Update Module Using Firmware Integration Technology (R01AN6850)

<u>RX Family Firmware Updating Communications Module Using Firmware Integration Technology</u> (R01AN7757)

RX Family Firmware Update Software Development Guide using AWS/Azure QE for OTA (R20AN0712)

RX Family How to implement FreeRTOS OTA using Amazon Web Services in RX65N (for v202210.01-LTSrx-1.1.0 or later) (R01AN7037)

RX65N Group CK-RX65N v2 User's Manual (R20UT5100)

FPB-RX140 v1 - User's Manual (R20UT5376)

(Download the latest versions from the Renesas Electronics Corp. website.)

Technical Updates and Technical News

(Download the latest versions from the Renesas Electronics Corp. website.)

Amazon Web Services, the "Powered by AWS" logo, and any other AWS trademarks used in such materials are trademarks of Amazon.com, Inc. or its affiliates in the United States and/or other countries.

FreeRTOS[™] and FreeRTOS.org[™] are trademarks of Amazon Web Services, Inc.

Pmod is a trademark of Digilent Inc.

All trademarks and registered trademarks are the property of their respective owners.



RX65N Group Sample Code for OTA Update of a Secondary Device by Amazon Web Services with the Use of FreeRTOS

Contents

1.	Overview	5
2.	Conditions for Confirming Operation	5
3.	Description of Hardware	6
4.	Description of Software	7
4.1	Control methods for cloud connection and OTA	
4.2	Firmware Update Methods	7
4.3	UART Communications between the Microcontrollers	9
4.3.1	5	
4.4	Folder/File Structure	
4.5	Code Size	11
5.	Operations of the Demonstration	12
6.	Setting up the Demonstration	12
6.1	Setting up the Hardware	
6.1.1	1 Setting up the CK-RX65N	
6.1.2	2 Setting up the FPB-RX140	
6.2	Setting up the Software	
6.2.1	1 Advance Preparation	
6.2.2	2 Logging in to AWS with QE for OTA	21
6.2.3	3 Creating and Running the Initial Firmware for the CK-RX65N	22
6.2.4	4 Creating and Running the Initial Firmware for the FPB-RX140	
6.3	Preparations for Displaying the Sensor Data Using the AWS Cloud	
7.	Procedure for Running the Demonstration	41
7.1	Checking the Initial State of Operation	41
7.2	Executing the OTA Update of the FPB-RX140	
7.2.1		
7.2.2	5	
7.2.3		
7.3	Checking Operation after the OTA Update	
8.	How to do the demo without using the HS3001 sensor	49
8.1	Changes to the demo procedure	
8.2	How to check demo operation when not using sensors	49
9.	Precautions	50
9.1	License Information on the Open-Source Software in Use	50
9.2	Region and User Privileges of AWS for the Demonstration	
9.3	Fees for Using AWS	



RX65N Group	Sample Code for OTA Update of a Secondary Device by Amazon Web Services with the Use of FreeRTOS
Revision Histor	y51



1. Overview

This demonstration involves using a secondary OTA update to add a working sensor and confirming addition of the sensor data to be acquired by the display of sensor data in an AWS display in your browser.

IoT devices require the appropriate fixing of security vulnerabilities and updating of functions in response to customer requests. Implementing the secondary OTA update to supplement OTA updating of the primary MCU that has been provided in the past enables product development that supports measures against vulnerabilities in the secondary MCU and the updating of flexible services.

2. Conditions for Confirming Operation

The sample demonstration programs for this application note have been confirmed to operate correctly under the following conditions.

Item	Description		
MCU	RX65N (R5F565NEHDFB)		
Board	CK-RX65N v2 (RTK5CK65N0S04000BE)		
Operating voltage	3.3 V		
RTOS	FreeRTOS v202210.01-LTS-1.3.1		
Integrated development	<u>e² studio 2025-04</u>		
environment (IDE)	<u>QE for OTA v2.2.0</u>		
C compiler	C/C++ Compiler Package for RX Family [CC-RX] v3.07.00		
	GCC for Renesas RX 8.3.0.202411		
Flash memory programming tool	Renesas Flash Programmer V3.19.00		

Table 2-1 Conditions for Confirming Demo Operation (RX65N)

Table 2-2 Conditions for Confirming Demo Operation (RX140)

Item	Description			
MCU	RX140 (R5F51406BGFN)			
Board	FPB-RX140 v1 (RTK5FP1400S00001BE)			
Operating voltage	3.3 V			
Integrated development	<u>e² studio 2025-04</u>			
environment (IDE)	QE for OTA v2.2.0			
C compiler	C/C++ Compiler Package for RX Family [CC-RX] v3.07.00			
	GCC for Renesas RX 8.3.0.202411			
Flash memory programming tool	Renesas Flash Programmer V3.19.00			
USB-UART converter	Pmod USBUART™			

Table 2-3 Condition for Confirming Demo Operation (Sensor)

Item	Description
Temperature and humidity sensor board	US082-HS3001EVZ board
Conversion board	US082-INTERPEVZ

Note : The HS3001 sensor will reach end-of-life (EOL) on September 30, 2025. For replacement products and other details, please refer to the EOL notification document below. The demonstration in this application note can be performed without the sensor. The method for performing the demonstration without the sensor is described in "8 How to do the demo without using the HS3001 sensor." https://www.renesas.com/document/eln/plc-250010-end-life-eol-process-select-part-numbers



Table 2-4 Condition for Confirming Demo Operation (Others)

Item	Version
Python	3.12.6

QE for OTA is available at <u>https://www.renesas.com/qe-ota/</u>.

Python is available at <u>https://www.python.org/</u>.

3. Description of Hardware

The system consists of an RX65N microcontroller (primary MCU) that provides functionality for controlling communications with AWS and an RX140 microcontroller (secondary MCU) connected to the HS3001 sensor. The two microcontrollers communicate with each other via UARTs.

The system configuration is shown in Figure 3-1.

The CK-RX65N v2 (hereafter referred to as "CK-RX65N") equipped with an RX65N microcontroller is used as the primary MCU.

The FPB-RX140 v1 (hereafter referred to as "FPB-RX140") equipped with an RX140 microcontroller is used as the secondary MCU.



Figure 3-1 System Configuration of This Demo



4. Description of Software

4.1 Control methods for cloud connection and OTA

"FreeRTOS[™] with IoT Library" is implemented in the RX65N firmware, which utilizes AWS-certified programs. This allows the use of AWS IoT Core and AWS IoT Device Management, which are managed services provided by AWS, to perform OTA firmware updating and data uploading to the cloud via MQTT communications.

The RX65N microcontroller on the primary MCU side uses the AWS IoT Over-the-air Update Library to control OTA updating of the secondary MCU. The update firmware for the secondary MCU, which is received from AWS, is transferred to the secondary MCU, where the firmware update is applied.

To communicate data between the primary MCU and secondary MCU, use the "<u>RX Family Firmware</u> <u>Updating Communications Module Using Firmware Integration Technology</u>"

The RX microcontroller on the secondary MCU side uses "<u>RX Family Firmware Update Module Using</u> <u>Firmware Integration Technology</u>" to control firmware updating of the secondary MCU.

4.2 Firmware Update Methods

The mechanism for firmware update in the secondary MCU of this sample program uses "linear mode partial update method" among the methods provided by the firmware update module. For details on this method, refer to "linear mode partial update method" in "1.3 Firmware Update Operation" in "<u>RX Family Firmware</u> <u>Update Module Using Firmware Integration Technology</u>".



Operations for the secondary OTA update are summarized in Figure 4-1. The states of the ROM in each phase during updating are shown in Figure 4-2. Note that the red frames in Figure 4-2 indicate the programs under execution at the given times.



Figure 4-1 Overview of Operations for the Secondary OTA Update







4.3 UART Communications between the Microcontrollers

To communicate data between the primary MCU and secondary MCU, use the "<u>RX Family Firmware</u> <u>Updating Communications Module Using Firmware Integration Technology</u>".

Communications related to secondary OTA update (communications between CK-RX65N and FPB-RX140 in the sequence diagram in Figure 4-1) use the commands from the FWUP command class.

In addition, the DATA_RECV command from the Common command class is used to receive sensor data from the secondary MCU, and when the command argument is 1, it is used as a sensor data transmission request.

For details on each command, refer to "1.6 Specifications of Commands" in "<u>RX Family Firmware Updating</u> <u>Communications Module Using Firmware Integration Technology</u>".

4.3.1 Settings of UART Communications

The UART communications settings are shown in Table 4-1.

ltem	Setting
Data length	8 bits
Parity bit	None
Stop bit	1 bit
Flow control	None
Bitrate	1 Mbps

Table 4-1 Settings of UART Communications between the Microcontrollers



4.4 Folder/File Structure

Figure 4-3 shows the folder/file structure.

r01an6220xx0300-rx-2nd-ota-apl ⊣Demo
afr-v202210.01-LTS-rx-1.3.1
-ccrx
demo_rx65n_ck_primary
│ │ │ └─demo_bl_rx65n_ck_primary
│
└─bootloader_rx140_fpb_w_buffer
└─gcc
│
demo_bl_rx65n_ck_primary
demo_app_rx140_fpb_w_buffer
└─bootloader rx140 fpb w buffer
r01an6220ej0300-rx-2nd-ota-apl.pdf
r01an6220jj0300-rx-2nd-ota-apl.pdf

Figure 4-3 Folder/File Structure

The demo_rx65n_ck_primary folder and demo_bl_rx65n_ck_primary folder contain project files for the CK-RX65N.

The demo_app_rx140_fpb_w_buffer folder and bootloader_rx140_fpb_w_buffer folder contain project files for the FPB-RX140.

The projects for the CK-RX65N and the FPB-RX140 support the CC-RX and GCC compilers.



4.5 Code Size

The ROM and RAM sizes for projects included in the sample code of this application note are listed in the tables below. The values in the tables were confirmed under the following conditions.

- CC-RX
 - Compiler

Optimization level (-optimize): Level 2: Performs whole module optimization Optimization type (-speed/-size): Optimizes with emphasis on code size

— Linker

Optimization type (-nooptimize/-optimize): All

— Library Generator

Optimization level (-optimize): Level 2: Performs whole module optimization Optimization type (-speed/-size): Optimizes with emphasis on code size

Table 4-2 Code Size (CC-RX)

Project	ROM	RAM	
demo_bl_rx65n_ck_primary	33 Kbytes	8 Kbytes	
demo_rx65n_ck_primary	601 Kbytes	382 Kbytes	
bootloader_rx140_fpb_w_buffer	29 Kbytes	6 Kbytes	
demo_app_rx140_fpb_w_buffer	50 Kbytes	13 Kbytes	

• GCC

Optimization level: Optimize for debug (-Og)

Table 4-3 Code Size (GCC)

Project	ROM	RAM	
demo_bl_rx65n_ck_primary	51 Kbytes	10 Kbytes	
demo_rx65n_ck_primary	654 Kbytes	383 Kbytes	
bootloader_rx140_fpb_w_buffer	24 Kbytes	13 Kbytes	
demo_app_rx140_fpb_w_buffer	41 Kbytes	12 Kbytes	



5. Operations of the Demonstration

- (1) In the initial state of the demonstration, the FPB-RX140 only acquires humidity data by using the HS3001 sensor.
- (2) The secondary OTA update mechanism is used to download the update firmware for the FPB-RX140 from AWS via the CK-RX65N and then update the firmware.
- (3) After the firmware updating is complete, the FPB-RX140 acquires temperature data in addition to humidity data from the HS3001 sensor.

In this sequence, the type of sensor data from which data are being acquired and the values can be checked from the log output from both microcontrollers to your PC and from the dashboard on AWS.

6. Setting up the Demonstration

This section describes the setting up required to run the demonstration covered by this application note.

The necessary steps are setting up the hardware, such as the wiring of the CK-RX65N and FPB-RX140 and connection of the HS3001 sensor, setting up the software, such as creating and writing the initial firmware for each microcontroller board, and the preparation on the AWS cloud side for execution of the OTA update and display of the sensor data from AWS.

6.1 Setting up the Hardware

Firstly, the overall hardware structure for this demonstration is shown below. For an actual image after setup is complete, see Figure 6-2. The methods for setting up each of the boards are described in detail in the subsequent subsections.



Figure 6-1 Overall Hardware Structure for This Demo



6.1.1 Setting up the CK-RX65N

The procedure for setting up the CK-RX65N is described in the following passages.

(1) Connecting the cable for UART communications with the FPB-RX140

TXD, RXD, and GND for UART communications with the FPB-RX140 are allocated to the following pins on the J23 and J24 connectors of the CK-RX65N. Connect the pins on the FPB-RX140 side as described in 6.1.2(2), with the corresponding UART signals listed in the table below.

Table 6-1 UART Connection Method between the Microcontrollers (CK-RX65N ↔ FPB-RX140)

CK-RX65N		FPB-RX140
J23 Pin 1: D0/RX	¢	J12 Pin 2: D1/TX
J23 Pin 2: D1/TX	¢	J12 Pin 1: D0/RX
J24 Pin 7: GND	¢	J10 Pin 7: GND





(2) Connecting the cable for log output to the PC

Connect the PC to the USB serial connector (USB Type-C) on the CK-RX65N with a USB cable.



(3) Connecting the power supply and debugger

Connect the PC to the E2OB Debugger connector (micro USB Type-B) on the CK-RX65N with a USB cable.



(4) Connecting the DA16600 Wi-Fi module

Connect the DA16600 Pmod module to the Pmod1 connector on the CK-RX65N.





(5) Closing jumper block J16 on the DEBUG side

To set the CK-RX65N to debug mode, close jumper block J16 on the DEBUG side (pins 1-2).



6.1.2 Setting up the FPB-RX140

The procedure for setting up the FPB-RX140 is described in the following passages.

(1) Connecting the HS3001 board

Connect the HS3001 board and the conversion board to the Pmod2 connector on the FPB-RX140.





(2) Connecting the cable for UART communications with the CK-RX65N

TXD, RXD, and GND for UART communications with the CK-RX65N are allocated to the following pins on the J12 and J10 connectors of the FPB-RX140. Connect the pins on the CK-RX65N as described in 6.1.1(1), with the corresponding UART signals listed in Table 6-1.



(3) Connecting the cable for serial communications to be used in log output to the PC

Connect the Pmod USBUART conversion board to the pins 1-6 of the Pmod1 connector on the FPB-RX140. Also, connect the PC and the micro USB Type-B connector on the Pmod USB-to-UART converter with a USB cable.





(4) Connecting the cable for power supply

Connect the PC and the micro USB Type-B connector on the FPB-RX0 with a USB cable.



(5) Opening the emulator reset header (J4)

Open the emulator reset header (J4) on the FPB-RX140.



The hardware setup for the demonstration is now completed.

Figure 6-2 is an image of the overall configuration for the demonstration.





Figure 6-2 Image of the Overall Configuration for the Demo



6.2 Setting up the Software

6.2.1 Advance Preparation

For each of the software versions used in confirming operation, see Table 2-1, Table 2-2, Table 2-3 and Table 2-4.

(1) Installing QE for OTA

From the e^2 studio menu bar, select [Renesas Views] \rightarrow [Renesas QE] to check whether QE for OTA is installed. If [OTA Main (QE)] and [OTA Manage IoT Device (QE)] are displayed, installation is completed.

oject Rer	esas Views Run Window	Hel	р	
5 84	C/C++	>	1 🔊	A + 🗾 🛯 🗊 👖 😓 + 🖓 + 🏷*
A Mai	Code Generator	>	uh X	
	Debug	>		
	Partner OS	>		1.1.1.1.1.
	Renesas QE	>	۸× I	Measuring Current Consumption (QE)
	Smart Configurator	>	4 (DTA Main (QE)
	Solution Toolkit	>	4 (DTA Manage IoT Device (QE)
2	Renesas Software Installer			n nuklis hau
_	* PERFERICOULOU COU	ae .	signe	r public key.

Regarding the version, from the e^2 studio bar, select [Help] \rightarrow [About e2 studio] \rightarrow [Installation Details] to confirm that the version of "Renesas QE for OTA" is "2.2.0.~" or higher.

	s Al Window		P Welcome	
8 🗖 🗖 esas GDB H;	476	?	Help Contents Search Show Context Help	
	+1.2	ö "	Show Active Keybindings Cheat Sheets	Ctrl+Shift+L
	480 481 482 483	89	Renesas Help Toolchain Help Add Renesas Toolchains	> >
	484 485 486	•	Eclipse User Storage Perform Setup Tasks	>
	487 488 489 490 491	0 ³ 😓 🕎 🍓	Check for Updates Install New Software Eclipse Marketplace IAR Embedded Workbench plugin manager	
	492 493		About e ² studio	



RX65N Group Sample Code for OTA Update of a Secondary Device by Amazon Web Services with the Use of FreeRTOS

Renesas e ² studio	^	Installed Software Installation History Features Plug-ins Configura	tion Renesas Device	Support Support Folders
Version: 2025-04 (25 Build Id: R20250411			non neneau benee	
$\Delta 4$	d	e Name	Version	ld ^
	2010-2025 Renesas Electronics Corp.	> 🚯 Renesas CMake Build Support Files	10.0.0.v20240909-0	com.renesas.ide.supportfiles
All rights reserved.	n		10.2.0.v20250217-1	com.renesas.e2studio.comm
e ² studio IDE is an e	extension of software developed for	O > Renesas Debug Views	10.2.0.v20250206-0	com.renesas.e2studio.debug
eclipse.org.	A A	c > 🚯 Renesas Debug Views (Lite)	10.2.0.v20250206-0	com.renesas.e2studio.debug
	A		25.4.0.v20241213-1	com.renesas.e2studio.tools.c
	ed on Eclipse Platform 4.34 (2024-12)	> Renesas e2 studio Code Generator for RX	25.4.0.v20241213-1	com.renesas.e2studio.tools.c
and CDT version 11.6			10.2.0.v20250325-1	com.renesas.e2studio.comm
China and Annua	Falless Foundation also include		10.2.0.v20250326-1	com.renesas.e2studio.comm
	Eclipse Foundation plug-ins is available wrg, under the Eclipse Public License "EPL",	O >	10.2.0.v20250326-1	com.renesas.e2studio.comm
	ipse.org/org/documents/epl-2.0/EPL-2.0.html	> 🚯 Renesas e2 studio Common Core	10.2.0.v20250305-1	com.renesas.e2studio.comm
	lines are loss (desuments (en], v10 html	> Renesas e2 studio Conflicts Checker	10.1.0.v20241014-1	com.renesas.e2studio.tools.c
A	ipse.org/org/documents/epi-vito.ntmi	> 🚯 Renesas e2 studio FIT	10.2.0.v20250226-0	com.renesas.e2studio.tools.fi
		P Renesas e2 studio Integration Service	10.2.0.v20250320-1	com.renesas.e2studio.tools.ir
🕒 🗁 🚍 🗠 e² 🕩	i 🏧 🚯 🥩 🤄 🔮 🤹 🖞	> 🚯 Renesas e2 studio Memory Usage View	10.2.0.v20250321-0	com.renesas.e2studio.comm
	s s s s s s s s s s s s s s s s s s s	R Renesas e2 studio Optimization Assistant	10.2.0.v20250321-0	com.renesas.e2studio.tools.c
		> Renesas e2 studio Reality Al Core	25.4.0.v20250313-1	com.renesas.e2studio.tools.re
		> Renesas e2 studio Reality AI for Renesas RL78	25.4.0.v20250313-1	com.renesas.e2studio.tools.re
Installation Details	Close	> 🚯 Renesas e2 studio Reality AI for Renesas RX	25.4.0.v20250313-1	com.renesas.e2studio.tools.re
		> Renesas e2 studio Stack Analysis View	10.1.0.v20241202-0	com.renesas.e2studio.comm
tode	Problems Console Pro	> 🚯 Renesas e ² studio	25.4.0.R20250411-0	com.renesas.ide.e2studio.prc
	Problems Console Pro	P Renesas QE common	2.4.0.v20250130-044	com.renesas.apltool.commo
itcredential_keys.h	All IoT devices	Renesas QE for OTA	2.2.0.202506141554	com.renesas.qe.cloud.feature
ntcredential.h	4 - X -	> Kenesas RL78 Debug Support	25.4.0.v20250124-0	com.renesas.e2studio.device
ner_public_key.h	T	Renesas RL78 Debug Support Files	25.4.0.v20250331-1	com.renesas.ide.supportfiles
ult_root_certificates.h	iot_device_ck-rx65n	Renesas RL78 Family Support (requiring .Net Framework)	25.4.0.R20250411-0	com.renesas.e2studio.rl78.dc
rap_code		Renesas RTOS Debug Views	10.2.0.v20250320-1	com.renesas.e2studio.debug
10p_0000		> Renesas RX Debug Support	25.4.0.v20241213-1	com.renesas.e2studio.device ¥
r_mcu_ota_demo		Renesas QE for OTA		^
				~

If they are not displayed, refer to "2.1 Install QE for OTA" in "<u>RX Family Firmware Update Software</u> <u>Development Guide using AWS/Azure QE for OTA (R20AN0712)</u>" and install QE for OTA.

(2) Installing the Python execution environment

Python can be downloaded from https://www.python.org/.

The pycryptodome library of Python is also to be used. After installing Python, execute the follow pip command to install it.

> pip install pycryptodome

(3) Installing the Renesas Flash Programmer

The Renesas Flash Programmer can be downloaded from <u>Renesas Flash Programmer (Programming GUI)</u> <u>Renesas</u>.



6.2.2 Logging in to AWS with QE for OTA

(1) Opening the QE for OTA window

From the e^2 studio menu bar, select [Renesas Views] \rightarrow [Renesas QE] \rightarrow [OTA Main (QE)].

oject	Renesas Views Run Window	Help	
5 %	C/C++	>	😕 🛷 🕶 🗾 🖼 🗉 👖 🕌 🕶 🏷 🕫
A Mai	Code Generator Debug	> >	.h ×
	Partner OS	>	
	Renesas QE	>	Measuring Current Consumption (QE)
	Smart Configurator	>	OTA Main (QE)
	Solution Toolkit	>	OTA Manage IoT Device (QE)
	Renesas Software Installer		igner public key.
	*		signer public key.

(2) <QE for OTA> [1. Cloud Settings] \rightarrow [Sign-in to Cloud]

From here, follow the steps displayed in the GUI window of QE for OTA.

Start by selecting "AWS" for [Cloud] and sign in. An AWS resource is generated in the region selected at the time of login.

	Cloud	Prepare	$\overline{\mathbf{O}}$	IoT	\rangle	OTA		
1.Cloud Se	ttings 🗸 🗸	<u>^</u>		Sign-in to Clou	ud			
🥊 Sign-in	to Cloud	Setup to sign-in to Cloud						
2.Prepare j	projects 👻	orup to sign-in to croad	_	Cloud		AWS	~	
Select j	projects					Ano		
Select J	provisioning					s	sign-in Settings	
3.Manage 1	loT device 👻							
🕢 Manage	e IoT device	Sign-in to AWS						
 Create initial firmware 		Ŭ						
🕢 Write p	rogram to IoT devices	 Create new IAM user account Sign-in to AWS 						
4.0TA	-	Create new IAM user account						
Create	update firmware							
 Execute 	e OTA and check status	 To use QE for OTA, you need an AWS refer to the following URL for new ac 					iew one. Please	



6.2.3 Creating and Running the Initial Firmware for the CK-RX65N

Create initial firmware for the CK-RX65N by using QE for OTA, then writing and running it. The procedure is described below.

(1) Importing projects

Import the "demo_bl_rx65n_ck_primary" project, a bootloader for the CK-RX65N, and the "demo_rx65n_ck_primary" project, a user program, into e² studio. There are CC-RX and GCC versions of the CK-RX65N project, but here we will use the CC-RX version.

When importing projects, uncheck [Copy projects into workspace] in the [Options] field.

	New	Alt+Shift+N >		🗐 Import — 🗆 🗙
	Open File		□ \$ 7	Select
۵,	Open Projects from File System			Create new projects from an archive file or directory.
	Recent Files	>		Create new projects from an archive the or directory.
	Close Editor	Ctrl+W		
	Close All Editors	Ctrl+Shift+W	ng existing code	Select an import wizard:
	Save	Ctrl+S		type filter text
	Save As			✓ 🕞 General
	Save All	Ctrl+Shift+S		Archive File
	Revert			Strate Projects into Workspace
	Move			File System
2	Rename	F2		Preferences
8	Refresh	F5		Projects from Folder or Archive
	Convert Line Delimiters To	>		Rename & Import Existing C/C++ Project into Workspace
2	Print	Ctrl+P		Renesas CC-RX project conversion to Renesas GCC RX
<u>होल</u>	Import			Renesas CS+ Project for CA78K0R/CA78K0
r_s	Export		•	Benesas CS+ Project for CC-RX, CC-RL and CC-RH
	Properties	Alt+Enter		🚘 Renesas GitHub FreeRTOS (with IoT libraries) Project
	Switch Workspace			Cample Projects on Renesas Website
	Restart	<i>(</i>		> 🧀 C/C++
	Exit			> 🗁 Code Generator 🗸 🗸
-			1	
				? < Back Next > Finish Cancel
				Cancel

🕲 Import				×
Import Projects				
Select a directory to sear	ch for existing Eclipse projects.			-
Select root directory:	C:\ws\fwup_comm\aws_ota\r01an6220xx0300-rx-2n	nc v	Browse.	
O Select archive file:		\sim	Browse.	
Projects:				
bootloader_rx140_	fpb_w_buffer (C:\ws\fwup_comm\aws_ota\r01an622	20xx0	Select Al	L
	fpb_w_buffer (C:\ws\fwup_comm\aws_ota\r01an622 pb_w_buffer (C:\ws\fwup_comm\aws_ota\r01an6220		Deselect /	AII
	pb_w_buffer (C:\ws\fwub_comm\aws_ota\t01an6220 pb_w_buffer (C:\ws\fwub_comm\aws_ota\t01an6220		Refresh	
	primary (C:\ws\fwup_comm\aws_ota\r01an6220xx0 primary (C:\ws\fwup_comm\aws_ota\r01an6220xx0			
	primary (C:\ws\twup_comm\aws_ota\r01an6220xx0 mary (C:\ws\fwup_comm\aws_ota\r01an6220xx0300			
demo_rx65n_ck_pri	mary (C:\ws\fwup_comm\aws_ota\r01an6220xx0300	0-rx-2		
<		>		
Options				
Search for nested pro				
Copy projects into wo	projects upon completion			
	eady exist in the workspace			
Working sets				
Add project to worki	na sets		New	
Working sets:			Select	
Working sets.		×	Select	
?	< Back Next > Finish		Cancel	
	A DUCK /VEAL > TIMBI		cancer	



(2) Entering Wi-Fi connection information

Enter the SSID of the Wi-Fi access point you want to use in clientcredentialWIFI_SSID and the password in clientcredentialWIFI_PASSWORD, as defined in "src/application_code/include/aws_clientcredential.h" in the demo_rx65n_ck_primary project.



(3) Selecting projects

Select the ck_rx65n_demo_bootloader project and the ck_rx65n_2ndota_demo project that were imported into e² studio earlier. Also, select "Primary" for Target Device.

Cloud Cloud	$\rangle \odot$	Prepare	\otimes	ΙοΤ	\rangle \otimes	ΟΤΑ
1.Cloud Settings	 ▲ ▲		s	elect projects		
⊘ Sign-in to Cloud						
2.Prepare projects	-				OTA projec	create New Import
Select projects						
Select provisioning					2nd OTA project	
3.Manage IoT device	•				Sample for 2nd C	TA project Output sample
✓ Manage IoT device						
✓ Create initial firmware	Select created projects					
✓ Write program to IoT devices			Boot Loa	ader	demo_bl_rx65n_ck_prima	ry ~
OTA	•		Firmw	vare	demo_rx65n_ck_primary	~ ~
✓ Create update firmware						
Execute OTA and check status	Information of firmwa	re project				
					D	evice: R5F565NEHxFB_DUAL Evaluation Board: CK-RX65N
	C lost to losis for C					
	Select the device for fit	rmware update				

(4) Selecting the provisioning method

Select "Source code includes credentials (asymmetric keys)" as the provisioning method.

Cloud	⊘ Prepare	⊙ IoT	\rangle $_{\odot}$	OTA	
1.Cloud Settings 🗸		Select provisioni	ng		
Sign-in to Cloud	Select a provisioning.				
2.Prepare projects Select projects 		Provisioning	Source onde includes o	redentials (asymmetric keys)	~
Select provisioning			Source code includes o	edendals (asymmetric keys)	
3.Manage IoT device 🗸	1				
⊘ Manage IoT device	Select provisioning(AWS)				
 Create initial firmware 	QE for V1.1.0 supports only 1 proviso	oning.			
⊘ Write program to IoT devices	QE Series How to install/uninstall	How to update FAQ License			
4.0TA -					
⊘ Create update firmware					
 Execute OTA and check status 					



(5) Creating an IoT device

Click "Open view" in Manage IoT device to open the "OTA Manage IoT Device (QE)" screen.

Click the "+" button under "All IoT devices" and select "Create IoT device."

Create IoT device Create IoT group	All IoT devices	IoT Device Initial Firr	nware Update Fir	mware OTA	Firmware L	og
	Create IoT device	IoT Device Name	Policy	Private Key	Public Key	Device

Enter an appropriate name in the "IoT device name" field of the dialog box, and then click "Create."

📴 Create loT dev	vice	×
IoT device name	iot_device_ck-rx65n	
Group name		~
Create number	1	•
Policy	qe_iot_policy	~
	Create	Cancel

(6) Creating the initial firmware

Click "Open view" in Create initial firmware.

\odot	Cloud	\odot	Prepare	$\supset \bigcirc$	IoT	$\overline{\bigcirc}$	OTA
1.Cloud Settings	-	$\hat{}$		Create init	tial firmware		
🧭 Sign-in to Cloud							
2.Prepare projects	•	Build Boot	loader and Firmware to create pr	ogram for IoT devices.			
Select projects						Create initial finitiw	Open view
Select provisionin	g						
3.Manage IoT devic	e 🗸						
Manage IoT dev		L 1	Create initial firmware : asymn	netric keys(AWS)			
🔗 Create initial fir			n				
			Install Python				
Write program t	o lo l'devices		Install OpenSSL				
4.OTA	-		Build initial firmware				



	IoT Device Initial	Firmware Update Firmware	e OTA Fir	rmware Log		
🕂 👻 👻 🔶 Sync Cloud	(1) Select IoT dev	ices for which want to creat	e initial firmw	are.		
iot_device_ck-rx65n	😥 Add all	IoT Device Name	Private Key	Public Key	Device Certificate	
	D Add	iot_device_ck-rx65n	0	0	0	
	Delete					
	👯 Delete all					
	🎋 Debug					

In the "Initial Firmware" tab, select the IoT device you created earlier and click "►Add".

Enter "1.0.0" for the version and click "Create initial firmware."

 (2) Specify version of firmware. (3) Create initial firmware. (4) Write initial firmware to IoT device and run it. Run the initial firmware, IoT device will start to conner
Write and Run 💿 Resume

The demo_bl_rx65n_ck_primary project and demo_rx65n_ck_primary project builds are executed, and if "Create firmware is complete." is displayed in the "Initial Firmware" tab, the process is successful.

	mware to IoT device and run it. firmware, IoT device will start to connect to the cloud. Connection is successful, it will wait
Write and Run	O Resume
now creating ini	
	ource and build bootloader project.
1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.	bl_rx65n_ck_primary/QE-OTA/bootloader/demo_bl_rx65n_ck_primary.mot
	bl_rx65n_ck_primary/QE-OTA/bootloader/demo_bl_rx65n_ck_primary.x
- 2/2 : Rewrite se	ource and build firmware project. Create initial firmware.
Output : /demo	rx65n_ck_primary/QE-OTA/apl/v1.0.0/iot_device_ck-rx65n/demo_rx65n_ck_primary.mot
Output : /demo	rx65n_ck_primary/QE-OTA/apl/v1.0.0/iot_device_ck-rx65n/demo_rx65n_ck_primary.x
Output : /demo	rx65n_ck_primary/QE-OTA/apl/v1.0.0/iot_device_ck-rx65n/merged.mot
Create firmware	is complete

(7) Writing and running the initial firmware to the CK-RX65N

Click "Write and Run" in the "Initial Firmware" tab.

	mware to IoT device and run it. firmware, IoT device will start to connect to the cloud. Connection is successful, it will wait
Write and Run	🔘 Resume
Output : /demo Output : /demo - 2/2 : Rewrite se Output : /demo Output : /demo	purce and build bootloader project. bl_rx65n_ck_primary/QE-OTA/bootloader/demo_bl_rx65n_ck_primary.mot bl_rx65n_ck_primary/QE-OTA/bootloader/demo_bl_rx65n_ck_primary.x purce and build firmware project. Create initial firmware. rx65n_ck_primary/QE-OTA/apl/v1.0.0/iot_device_ck-rx65n/demo_rx65n_ck_primary.mot rx65n_ck_primary/QE-OTA/apl/v1.0.0/iot_device_ck-rx65n/demo_rx65n_ck_primary.x rx65n_ck_primary/QE-OTA/apl/v1.0.0/iot_device_ck-rx65n/demo_rx65n_ck_primary.x



Select the COM port number for the CK-RX65N and click "Write." Each parameter for the Renesas Flash Programmer will be automatically entered if the CK-RX65N onboard emulator is connected. If any fields are blank, check that the CK-RX65N is connected to the PC and that jumper J16 is connected to the DEBUG side.



If "Rewrite provisioning information in data flash is complete." is displayed, the operation was successful. Click "Run program."

Elash Programming	×
COM POL: COM47 V CONNECTED to COM47.	
MIICODCCAd8CFHLgF+xxxxhaYXEH67DBgYb50UCIMAGCCCqSSM49BAMCMIGeMScw AQVIKoZilwcNAQ4BFhb5b3yUWFi72HiC3NAZ5hhbX8255jb20xCzABgNVBAYT AVTTMRMeCgY02QD2DAp2539ifYhc7XIRIINieA7D2VQDH02/E35ylEpdHicFTAT BgNVBAAMDFlvXIQ29tcGfterEYMBMGA1UECwwMWV91elBT2WN0aW9JMQ8WQWDY VQQDDAZJWYtha2whithNNIJMWbJMDQ0NAXWWKMVMHXMwIjE4MDQ0NAXWWC0Bin MCUCC5q50b3D02LARYVeW91c1h2C6Ry2ZNxQCV4W1wbGUV29tMQxxCQVDVQQG EWV/UZETMBE6A1UCCAWKW91c16R072TSMBAGA1UEBwWWW91eBDAX5MMUw EwYDVQQUDAz2b3yIENvbSbbhx6rTATBgNVBAMDFhvXlgUZyJdGivbjEPMA0G AUEAwwCbmFYMpHFirekYH1K2JCGAVVIX2UZGJCDA2CD364DWX2AVF5/ 1u1GXgwUWJXRV90zpfmg+j8VypwQINQE1+/ZThrtzRi2AkVeS0+1DXkzqdCUda ByXE8od92AkBggpihQPA2DAg1HADEEABAPIR-WSKINj77m399u3ag6diGYN 1B0/9TFDQQ8+qB3Nk97TCgwHabSiNH0/99Ve2aWTek5xolf2M0u80= END CERTIFICATE	
ОК.	-
>conf commit	
0 1662 [CLI] Destroyed Certificate.	
1 1671 [CLI] Write certificate	
2 1728 [CLI] Destroyed Private key.	
3 1792 [CLI] Write Private key	
Configuration save 2947 bytes to Data Flash. Total used size is 2947 bytes .	
>	
======================================	
Connecting the tool (E2 emulator Lite) Tool: E2 emulator Lite (OBE110024) Interface: FINE Tool Firmware Version: V3.05.01.000	
Disconnecting the tool	
No operation	
Rewrite provisioning information in data flash is complete.	
v	· ·
Weite Date source Day and the	
Write Data rewrite Run program Close	

Open the "Firmware Log" tab. Here you can see the logs output from the CK-RX65N.

Connected to COM, firmware log is dis	played.	
COM Port : COM47 V Disco	nnect Connected to COM47.	1 4 5 🔳 💥 🗁 🖬
✓ AWS	1 1671 [CLJ] Write certificate	1
✓ CLI - Data (5)	2 1728 [CLI] Destroyed Private key.	
- Complete (1)	3 1792 [CLI] Write Private key	
 Connection Complete (0) 	Configuration save 2947 bytes to Data Flash. Total used size is 2947 bytes .	
- Error (0)	>verify install area main [hash-sha256]OK	
- Thing Name (0) - Complete (0)	execute image FreeRTOS command server. Type Help to view a list of registered commands.	
 ✓ MCU -> AWS - Error (0) 	Standard procedure: 1. Set value for endpoint/thingname/certificate/key/codesigncert.	
 AWS -> MCU Error (0) 	Write the key value to Internal Data Flash Memory with 'commit' command. Reset the program to start the demo.	
✓ OTA - Download (0)	>Press CLI and enter to switch to CLI mode or wait 10secs to run demol	
- Swap Bank (0) - Version (0)	>0 10760 [MAIN_TASK] [INFO] R_WIFL_DA16XXX_Open: Test with baud rate 115200! 1 10760 [MAIN_TASK] [DEBUG] at_exec. ATZ	
- Complete (0)	2 10922 [MAIN_TASK] [DEBUG] 🔶	
	3 10924 [MAIN_TASK] [DEBUG] +INIT:DONE,0	
	4 11924 [MAIN_TASK] [WARN] at_recv: response timeout!	
	<	>



(8) Closing jumper block J16 on the RUN side

Close jumper block J16 on the CK-RX65N on the RUN side (pins 2-3).



6.2.4 Creating and Running the Initial Firmware for the FPB-RX140

Create initial firmware for the FPB-RX140 by using QE for OTA, then writing and running it. The procedure is described below.

(1) Importing projects

Similarly to the procedure for importing projects for the CK-RX65N described in the previous subsection, import the "bootloader_rx140_fpb_w_buffer" project, a bootloader for the FPB-RX140, and the "demo_app_rx140_fpb_w_buffer" project, a user program, into e² studio.

(2) Selecting projects

Select the bootloader_rx140_fpb_w_buffer project and the demo_app_rx140_fpb_w_buffer project that were imported into e² studio earlier. Also, select "Secondary" for Target Device.





(3) Creating the initial firmware

Click "Open view" in Create initial firmware to open the "OTA Manage IoT Device (QE)" screen.

Cloud	⊘ Prepare	\rangle \odot	IoT	$\supset \odot$	ΟΤΑ
1.Cloud Settings	▼	Create ini	itial firmware		
Sign-in to Cloud					
2.Prepare projects	Build Boot loader and Firmware to create pr	ogram for IoT devices.			
Select projects				Create Initial Initia	Open view
Select provisioning					
3.Manage IoT device	-				
Manage IoT device	Create initial firmware : asymm	etric keys(AWS)			
Create initial firmware	Despense consting initial formure				
(>) Write program to 101 devices	Install Python				
	Install OpenSSL				
4.OTA	Build initial firmware				

In the "Initial Firmware" tab, select the IoT device created 6.2.3(5) and click "►Add".

All IoT devices	IoT Device	Initial Firmware	Update Firmware	OTA F	irmware Log		
🕂 👻 🗶 🖛 Sync Cloud	(1) Select I	oT devices for wh	ich want to create	e initial firm	ware.		
iot_device_ck-rx65n	🚺 Add a			Private Key	Public Key	Device Certificate	
	DAd	iot_de	vice_ck-rx65n	0	0	 Image: A start of the start of	
	Delete	e					
	🔇 Delete	e all					
	🏇 Debu	a					

Enter "1.0.0" for the version and click "Create initial firmware."

 (2) Specify version of firmware. V 1 2 0 2 0 2 (3) Create initial firmware. (4) Write initial firmware to IoT device and run it. Run the initial firmware, IoT device will start to connect Write and Run Run Rum Run

The bootloader_rx140_fpb_w_buffer project and demo_app_rx140_fpb_w_buffer project builds are executed, and if "Create firmware is complete." is displayed in the "Initial Firmware" tab, the process is successful.

	rmware to IoT device and run it. I firmware, IoT device will start to connect to the cloud. Connection is successful, it will wait f
Write and Run	D Resume
Output : /demo, Output : /demo, - 2/2 : Rewrite s Output : /demo, Output : /demo,	ource and build bootloader project. bl_rx65n_ck_primary/QE-OTA/bootloader/demo_bl_rx65n_ck_primary.mot bl_rx65n_ck_primary/QE-OTA/bootloader/demo_bl_rx65n_ck_primary.x ource and build firmware project. Create initial firmware. rx65n_ck_primary/QE-OTA/apl/v1.0.0/iot_device_ck-rx65n/demo_rx65n_ck_primary.mot rx65n_ck_primary/QE-OTA/apl/v1.0.0/iot_device_ck-rx65n/demo_rx65n_ck_primary.x rx65n_ck_primary/QE-OTA/apl/v1.0.0/iot_device_ck-rx65n/demo_rx65n_ck_primary.x



(4) Writing and running the initial firmware to the FPB-RX140

Click "Write and Run" in the "Initial Firmware" tab.

	mware to IoT device and run it. firmware, IoT device will start to connect to the cloud. Connection is successful, it will wait f
Write and Run	O Resume
Output : /demo_ Output : /demo_ - 2/2 : Rewrite so Output : /demo_ Output : /demo_	burce and build bootloader project. bl_rx65n_ck_primary/QE-OTA/bootloader/demo_bl_rx65n_ck_primary.mot bl_rx65n_ck_primary/QE-OTA/bootloader/demo_bl_rx65n_ck_primary.x burce and build firmware project. Create initial firmware. rx65n_ck_primary/QE-OTA/apl/v1.0.0/iot_device_ck-rx65n/demo_rx65n_ck_primary.mot rx65n_ck_primary/QE-OTA/apl/v1.0.0/iot_device_ck-rx65n/demo_rx65n_ck_primary.x rx65n_ck_primary/QE-OTA/apl/v1.0.0/iot_device_ck-rx65n/demo_rx65n_ck_primary.x

Select the COM port number for the Pmod USBUART conversion board connected to the FPB-RX140 and click "Write." Each parameter for the Renesas Flash Programmer will be automatically entered if the FPB-RX140 onboard emulator is connected. If any fields are blank, check that the FPB-RX140 is connected to the PC and that jumper J4 is open.



If "Write initial firmware is complete." is displayed, the operation was successful. Click "Run program."

Flash Programming	×
Erasing the target device [Data Flash 1] 00100000 - 00101FFF [Code Flash 1] FFFF0000 - FFFFFFF	^
Disconnecting the tool	
Operation successful	
======= AUTO_NOERASE ======= Renesss Flash Programmer CLI V1.12 Module Version V3.19.00.000 Load: "C\ws\fwup_comm\aws_ota\ccrx\demo_app_rx140_fpb_w_buffer\QE-OTA\apI\v1.0.0\iot_device_ck-rx65n_0xAC	
Connecting the tool (E2 emulator Lite) Tool: E2 emulator Lite (OBE110029) Interface: FINE Tool Firmware Version: V3.05.01.000	
Emulator power supply: OFF Connecting the target device Main Clock 32 MHz Speed: 1,500,000 bps [Warning] ID Authentication is disabled Connected to RX100 Series	
Writing data to the target device [Code Flash 1] FFFC0000 - FFFCEFF [Code Flash 1] FFFD9000 - FFFFDFFF [Code Flash 1] FFFFF000 - FFFFFFFF Verifying data on the target device [Code Flash 1] FFFFF00 - FFFCEFFF [Code Flash 1] FFFD900 - FFFCEFFF [Code Flash 1] FFFFF00 - FFFFFFFF	
Disconnecting the tool	
Operation successful	
Write initial firmware is complete.	· ·
Write Run program Close	

Open the "Firmware Log" tab and click "Secondary."

COM Port: Auto Connect Coll AWS CUI - Data (0) - Complete (0) - Complete (0) - Error (0) V Fleet - Thing Name (0) - Complete (0) - Error (0) V AWS -> MCU - Error (0) - Error (0) - Error (0) - Error (0) - Error (0) - Complete (0) - Error (0) - Ero	Connected to COM, firmware lo	g is displayed.	
 ✓ CU Data (0) Complete (0) ✓ Connection - Complete (0) • Error (0) ✓ Fleet - Thing Name (0) - Complete (0) ✓ MCU -> AWS • Error (0) ✓ AWS -> MCU - Error (0) ✓ OTA - Download (0) - Swap Bank (0) - Version (0) 	COM Port : Auto ~	Connect	
	<pre>> CLI - Data (0) - Complete (0) - Complete (0) - Error (0) + Fireet - Thing Name (0) - Complete (0) - Complete (0) - MCU -> AWS - Error (0) - AWS -> MCU - Error (0) - ODA - Download (0) - Swap Bank (0) - Version (0)</pre>		



Here you can see the logs output from the FPB-RX140. If the humidity data measured by the HS3001 sensor is displayed after "ver 1.0.0" is displayed, the operation is successful.

COM Port : COM38 V Disconnect Connected to COM38.	Solution (1998)
	^
==== RX140 : BootLoader [with buffer] ====	
verify install area main [sig-sha256-ecdsa]OK	
execute image	
==== RX140 : SECONDARY OTA DEMO [with buffer] ver 1.0.0 ====	
[HS3001]HUMI:73.86[RH]	
[HS3001]HUMI:73.80[RH]	
	~



6.3 Preparations for Displaying the Sensor Data Using the AWS Cloud

To display the received sensor data in a graphical format, set up Amazon CloudWatch and AWS IoT Core through the following steps.

Note: If you do not need to display the data in a graphical format, but only need to confirm in your browser that the data have been received by AWS, you can omit the entire procedure in 6.3. In this case, as shown in Figure 6-3, you can subscribe to "iotdemo/topic/sensor" in [MQTT test client] of AWS IoT to confirm in a text format that the sensor data are being received as expected.

AWS IoT <	MQTT test client Info
Monitor	You can use the MQTT test client to monitor the MQTT messages being passed in your AWS a inform devices and apps of changes and events. You can subscribe to MQTT message topics a
Connect Connect one device Connect many devices Domain configurations	Connection details ② Connected To disconnect from the MQTT test client, choose Disconnect. To re-establish a connec Subscribe to a topic Publish to a topic
Test Device Advisor	Topic filter Info The topic filter describes the topic(s) to which you want to subscribe. The topic filter can include MQTT v iotdemo/topic/sensor

Figure 6-3 Confirming Data Reception by the MQTT Test Client

(1) Logging in to the AWS Management Console

Start by logging in to the AWS Management Console.

Manage AWS Resources - AWS Management Console - AWS (amazon.com)

Confirm the region displayed in the upper-right corner of the management console screen and select the same region as that set at the time of logging in to QE for OTA.





(2) Creating rules in AWS IoT

Click on [AWS IoT] \rightarrow [Rules] \rightarrow [Create rule].



(3) Specifying the rule properties

Enter a rule name in [Rule name] and click on [Next].

AWS IOT > Message routing > Ru	rs / Greate rule	
Step 1	Encrify rule properties	
Specify rule properties	Specify rule properties Info	
Step 2	A rule resource contains a list of actions based on the MQTT topic stream.	
O Configure SQL statement		
Step 3	Rule properties	
 Attach rule actions 	Rule name	
Step 4	cloudwatch_visualize_rule	
Review and create	Citer an approxime tract can also contain underscore [] characters, but no spaces.	
0	Cities an apprairiance summy that can app contain universive () characters, but no spaces.	
	Rule description - optional	
	Enter a description to provide additional details about the side to others.	
	A description of your new rule	
	▼ Tags - optional	
	No tags associated with the resource.	
	Add new tag	
	You can add up to 50 tags.	
		Cancel
dShell Feedback	@ 2025 Amazon Web	Services, Inc. or its affiliates. Privacy Terms Cookie p



(4) Setting the SQL statement

Enter the SQL statement by entering code like the following in the text editor field for [SQL statement]. Be sure to add a new line character at the end.

SELECT *, timestamp() as timestamp FROM 'iotdemo/topic/sensor'
(A new line has to be entered at the end of the line above.)

Step 1 Specify rule properties	Configure SQL statement Info		
Step 2	Add a simplified SQL syntax to filter messages received on an MQTT topic and push the data elsewhere.		
Configure SQL statement			
Step 3	SQL statement Info		
Attach rule actions	SQL version		
Step 4	The version of the SQL rules engine to use when evaluating the rule.		
 Review and create 	2016-03-23	•	
	SQL statement		
		emperature FROM 'iot/topic' WHERE temperature	50. To learn more, see <u>AWS IoT SQL Reference</u> .
	1 SELECT *, timestamp() as timestamp FROM 'iotdemo/topic/sensor' 2		
		_	
	Children Common American		
	SQL Ln 2, Col 1 ③ Errors: 0 🖄 Warnings: 0	۲	
	SQL Ln 2, Col 1 ③ Errore: 0 △ Warninge: 0	e e	
	SQL Ln 2, Col 1 ③ Errors: 0 🛆 Warnings: 0	ø	Carcel Previous Next
	SQL Ln 2, Col 1 ③ Errors: 0 🛆 Warnings: 0	ø	Cancel Previous Next

(5) Selecting rule actions in the [Attach rule actions] step

Select "CloudWatch logs" for [Action 1] and click on [Create CloudWatch Log group].

	Create rule	9
Step 1 Specify rule properties	Attach rule actions into	
Step 2	An action routes data to a specific AWS service.	
Configure SQL statement	SQL statement Back	
Step 3 Attach rule actions	SELECT *, timestamp() as timestamp FROM 'iotdemo/topic/sensor'	
ep 4		
Review and create	Rule actions into	
	Select one or more actions to happen when the above rule is matched by an inbound message. Actions define additional activities that occur when messages arrive, like storing them in a database,	
	invoking cloud functions, or sending notifications. You can add up to 10 actions. Action 1	
	CloudWatch logs	
	Send message data to CloudWatch logs	
	Log group name Info Choose a CloudWatch Log group (C) (C) (C) (C) (C) (C) (C) (C	
	Batch mode The payload that contains a JSON array of records will be sent to Cloud watch via a batch call.	
	Use batch mode	
	IAM role Choose a role to grant AWS IoT access to your endpoint.	
	Choose an IAM role View 🖄 Create new role	
	AWS IoT will automatically create a policy with a prefix of "aws-iot-rule" under your IAM role selected.	
	Add rule action	I
	Error action - optional	
	To a control - optional year and that will be executed when something goes wrong with processing your rule. If two rule actions in the same rule fail, the error action receives one message that contains	



(6) Creating a log group

Enter a log group name and click on [Create].

workes and recents workes and recents bashboards New Narms © O © O ogs oroups og Anomalies wer all ogs Insights ontributor insights wetris Application Signals Neve expire Vetwork Monitoring nsights ettings ettings ettings ettings ettings ettings ettings ettings ettings Started New Create log groups Couldwatch Logs offers two log classes: Standard and Infrequent Access. Learn more about the features offers two log classes: Standard and Infrequent Access. Learn more about the features offers two log class. If the features offers two log class. If two log class interport of two log class. If two log class inter	CloudWatch > Log groups	> Create log group	0 0
Automation Name A constraint of the set of t	CloudWatch <	Create log group	
etings elemetry config <u>New</u> that's new Tags A tag is a label that you assign to an Amazon Web Services resource. Each on consists of a key and an optional value. You can use tags to search and filter your resources or track your Amazon Web Services costs. No tags are associated with this log group. Add new tag You can add up to 50 more tag(s).	Dashboards <u>New</u> Alarms △ o ⊘ o ⊖ o Logs Log groups Log Anomalies Live Tail Logs insights Contributor Insights Metrics	Log group details Info CloudWatch Logs offers two log classes: Standard and Infrequent Access. Learn more about the features offered by each log class. [2] Log group name sensor-visualize Retention letting Newer expire Log class Info Standard	
	Insights Settings Telemetry config <u>New</u> Getting Started <u>New</u> What's new	A tag is a label that you assign to an Amazon Web Services resource. Each to consists of a key and an optional value. You can use tags to search and filter your resources or track your Amazon Web Vervices costs. No tags are associated with this log group. Add new tag You can add up to 50 more tag(t).	

(7) Creating a new role

Г

Select the created log group in [Log group name] and click on [Create new role].

Step 1	
Specify rule properties	Attach rule actions info
Step 2	An action routes data to a specific AWS service.
Configure SQL statement	
Step 3	SQL statement Back
Attach rule actions	SELECT *, timestamp() as timestamp FROM 'iotdemo/topic/sensor'
Step 4	
Review and create	
	Rule actions Info
	Select one or more actions to happen when the above rule is matched by an inbound message. Actions define additional activities that occur when messages arrive, like storing them in a database,
	invoking cloud functions, or sending notifications. You can add up to 10 actions.
	Action 1
	CloudWatch logs Remove
	senomessage out a to cooperation of the second se
	Log group name info
	sensor-visualize 🔻 🔿 (C) (Z) Create CloudWatch Log group (Z)
	Batch mode The payload that contains a JSON array of resolvengil be sent to Cloud watch via a batch call.
	Use batch mode
	IAM role
	IAM TORE Choose a role to grant AWS IoT access to your endpoint.
	Choose an IAM role 🔹 🐨 🕜 View 🖸 Create new role
	AWS IoT will automatically create a policy with a prefix of "aws-iot-rule" under your IAM role selected.
	Add rule action


Enter the roll name and click [Create].

Step 1 Specify rule properties Step 2	Attach rule actions Info An action routes data to a specific AWS service.
Configure SQL statement Step 3 Attach rule actions Step 4 Review and create	SQL statement Back SELECT *, timestamp() as timestamp FROM 'iotdemo/topic/sensor'
	Rule actions info Stext one or more actions to him Stext one or more actions to him CoudWatch log: Bendimizing data to Coudifier CoudWatch log: Bendimizing data to Coudifier Sensor-visualize In the polyadal that contains a 3DAL CoudWatch row CoudWatch row Best on contains a 3DAL The polyadal that contains a 3DAL Conce on IAM role Conce on IAM role Conce on IAM role Attribute log: Conce on IAM role Conce on IAM role
	Add rule action

(8) Selecting the created IAM role

Select the created role in [IAM role].

aws I III Q Search	[Alt+5] D D O O O O Inited States (N, California) • nakaki @ 8886-9508-8161 •
AWS IoT > Message routing > Rule: Step 4	SELECT *, timestamp() as timestamp FROM 'iotdemo/topic/sensor'
Review and create	Head column in the source water
	Error action - optional You can optionally set an action that will be executed when something goes wrong with processing your rule. If two rule actions in the same rule fail, the error action receives one message that contains both errors. Add error action
	Cancel Previous Next
CloudShell Feedback	© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookle preference



(9) Confirming successful creation of the rule

Click on [Next] and then click on [Create] on the subsequent page. Finally, confirm that the created rule is displayed in the list of rules.

aws I iii Q Search	[Alt+S]	Σ 🗘 ⑦ 🔞 United States (Ν. California) ▼ nakaki@ 8886-9508-8161 ▼
■ AWS IOT > Message routing > Rules	> Create rule	0
Step 4 Review and create	<pre>SELECT *, timestamp() as timestamp FROM 'iotdemo/topic/sensor'</pre>	
 Neview and Ueace 	Rule actions: Info Select one or more actions to happen when the above rule is matched by an inbound message. Actions define invoking cloud functions, or sending notifications. You can add up to 10 actions. Action 1 CouldWatch logs Send message data to CloudWatch logs Log group name Info sensor-visualize The payload that contains a JSON array of records will be sent to Cloud watch via a batch call. Use batch mode Hay Del Choose a role to grant AWS foT access to your endpoint.	• Remove
	sensor-visualize View C Create new role AWS IoT will automatically create a policy with a prefix of "aws-iot-rule" under your IAM role selected. Add rule action	
	Error action - optional You can optionally set an action that will be executed when something goes wrong with processing your rule. both errors.	If two rule actions in the same rule fail, the error action receives one message that contains
		Cancel Previous Next
CloudShell Feedback		\oplus 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookle preferences ,

AWS IOT > Message routing > Rules	> Create rule			0
Step 3 Attach rule actions Step 4 Review and create	Rule properties Name cloudwatch_visualize_rule Description -			
	Step 2: SQL statement			Edit
	SQL statement SQL version 2016-03-23 SQL query SELECT *, timestamp() as timestamp FROM 'iotdemo/topic/s Step 3: Rule actions Actions Log group name sensor-visualize [3]	IAM role amrawsiam:888695088161role/service-role/sensor- visualize [2]	Batch mode False	Edit
	Error action No error action		Cancel	Previous Create
CloudShell Feedback		© 202	25, Amazon Web Services, Inc. or its affiliates.	Privacy Terms Cookie preferences



AWS IoT <				
AWS IoT <	Successfully created rule cloudwatch_visualize	e_rule.		(View rule
Ionitor	Rules (1) Info		C Activate Deactivate Edit De	elete Create rule
	Rules allow your things to interact with other ser	vices. Rules are analyzed and perform specific actions ba	ased on messages published by your devices.	
onnect Connect one device	Q. Find rules			< 1 > 食
Connect many devices	□ Name	▲ Status	▼ Rule topic ▼ Created date	
Domain		'		
configurations Updated	cloudwatch_visualize_rule	Active	iotdemo/topic/s June 27, 2025, 12:37:10 (UTC+09:00)	
 All devices Software packages Remote actions 				
All devices Software packages Remote actions Message routing Rules				
All devices Software packages Remote actions Message routing Rules Destinations				
Remote actions Message routing Rules Destinations Retained messages				
All devices Software packages Remote actions Message routing Rules Destinations Retained messages				
All devices Software packages Remote actions Message routing Rules Destinations Retained messages Security				
All devices Software packages Remote actions Message routing Rules Destinations Retained messages Security evice software				
All devices Software packages Remote actions Message routing Rules Destinations				

(10) Checking the graphical display in Amazon CloudWatch

Display the screen of Amazon CloudWatch and click on [Logs Insights] on the menu at left.

Select the group that was created in 6.3.2.1(5) as the log group.

	(
Favorites and recents	Logs Insights Info Select log groups, and then run a que		Start tailing 5m 30n	n 1h 3h 12h Custom 🔢	Compare (Off) UTC times	zone 🔻
Dashboards	Select log groups by	Selection criteria			Browse log groups	Q
Alarms 🛆 o ⊘ o 💬 o	Log group name	Select up to 50 log groups			Browse log groups	Discover
Logs	1 fields@timestamp.g 2 sort@timestamp.g				Г	netus
.og groups	3 11mit 10000	Standard				D
.og Anomalies .ive Tail						Queries
Logs Insights	Run query Cancel	Save History				
Contributor Insights	Logs Insights QL query can run for m	naximum of 60 minutes.				
	Logs (-) Patterns (-)	Visualization				
Metrics	Logs (-)			Export result	s 🔻 🛛 🔍 Add to dashboard	8
Application Signals New (APM)						
Network Monitoring			No results Run a query to see related events			
Insights						
Settings						
Telemetry config New						
-						
elemetry config New						
elemetry config New						
elemetry config New						
elemetry config New						



Enter the following query and click on [Run query].

stats avg(hs300x_humidity), avg(hs300x_temperature) by bin(1m)

A graph is displayed on the [Visualization] tabbed page.

CloudWatch	<		
Favorites and recents	•	Logs Insights info Select log groups, and then run a query or choose a sample query.	mezone 🔻
ashboards		Select log groups by Selection criteria	
larms 🛆 o ⊘ o 💬 o		Log group name V Select up to 50 log groups V Browse log groups	
ogs		sensor-visualize X (Clear all	Discovered
g groups		<pre>i stats avg(hs300x_humidity), avg(hs300x_temperature) by bin(1m)</pre>	
g Anomalies			
re Tail			Queries
gs Insights		Run query Cancel Save History	
ntributor Insights		Logs Insights QL traver can run for maximum of 60 minutes.	
etrics		O Completed. Query execute for 1 log group (
pplication Signals APM)	New	Logs (5) Patterns (-) Visualization	
letwork Monitoring		Visualization Add to	fashboard
nsights		Graph type: Line 🔻	
ettings			
lemetry config New			00x_humidity) 00x_temperature)
tting Started		60	
nat's new		50	
		40	
		30	
		20	



7. Procedure for Running the Demonstration

The procedure for running the demonstration is described below.

7.1 Checking the Initial State of Operation

With the setup for the demonstration described in section 6 completed, check the logs from each microcontroller in the "Firmware Log" tab of QE for OTA.

Confirm that humidity data from the HS3001 sensor is being output in the CK-RX65N log. Below that, you can also see the log of sensor data being sent to AWS via MQTT communications.

Next, confirm that only the humidity data from the HS3001 sensor is being output in the FPB-RX140 log. Also, In the initial state, LED1 of the FPB-RX140 is blinking.

If they are not displayed correctly, press the reset switch (S2 RESET) on the FPB-RX140 to apply a hardware reset. Similarly, press the reset switch (S1 RESET) on the CK-RX65N to apply a hardware reset.

OM Port : COM47 V	Disconnect Connected to COM47.	🔶 🐣 🚺 💥 🗁 [
 AWS CLI Data (0) Complete (0) Connection Complete (0) Error (0) Fleet 	 4/5/5 6164/5/ [MQI I] [WAKN] K_WIH_DA16XXX_ReceiveSocket: timeout! 4/575 6164757 [MQTT] [INFO] R_WIFI_DA16XXX_ReceiveSocket: socket 0 recv_cnt=0 (450). 47577 6165257 [MQTT] [WARN] R_WIFI_DA16XXX_ReceiveSocket: timeout! 47578 6165257 [MQTT] [INFO] R_WIFI_DA16XXX_ReceiveSocket: socket 0 recv_cnt=0 (450). 47579 6165757 [MQTT] [WARN] R_WIFI_DA16XXX_ReceiveSocket: timeout! 47580 6165757 [MQTT] [WARN] R_WIFI_DA16XXX_ReceiveSocket: socket 0 recv_cnt=0 (450). 47581 6166097 [Sensor Task] [INFO] [HS300X SENSOR] HUMIDITY: 7647 [%RH] 47582 6166257 [MQTT] [WARN] R_WIFI_DA16XXX_ReceiveSocket: timeout! 47583 6166257 [MQTT] [INFO] R_WIFI_DA16XXX_ReceiveSocket: socket 0 recv_cnt=0 (450). 47584 6166257 [MQTT] [INFO] Publishing message to iotdemo/topic/sensor. 	
- Thing Name (0) - Complete (0) ✓ MCU -> AWS - Error (0) ✓ AWS -> MCU - Error (0)	47585 6166257 [MQTT] [DEBUG] SendSocket: 47588 6166282 [MQTT] [INFO] R_WIFI_DA16XXX_SendSocket: socket 0 ret=33 (25).	>
 OTA Download (0) Swap Bank (0) Version (0) Complete (0) 	COM Port : COM38 Disconnect Connected to COM38. [HS3001]HUMI:76.36[RH] [HS3001]HUMI:76.41[RH] [HS3001]HUMI:76.43[RH] [HS3001]HUMI:76.46[RH] [HS3001]HUMI:76.46[RH] [HS3001]HUMI:76.46[RH] [HS3001]HUMI:76.46[RH]	A A A A



Finally, Figure 7-1 shows the display for Amazon CloudWatch. Click on [Run query] and confirm that the humidity data acquired from the HS3001 sensor are displayed as a graph.

loudWatch <	Sensor-visualize X Clear all	Discovered fields
avorites and recents	<pre>stats avg(hs300x_humidity), avg(hs300x_temperature) by bin(im)</pre>	D
ashboards		Queries
larms ≙ ₀ ⊘ ₀ ⊕ ₀	Run query Cancel Save History	
ogs	Logs Insights QL query can run for maximum of 60 minutes.	
og groups	© Completed. Query executed for 1 log group. ()	
og Anomalies	Logs (5) Patterns (-) Visualization	
ve Tail		
ogs Insights	Visualization	Add to dashboard
ontributor Insights	Graph type: Line 🔻	
letrics		
pplication Signals <u>New</u> APM)	65	 1. avg(hs300x_humidity) 2. avg(hs300x_temperature)
letwork Monitoring	60	
-		
nsights	55	
-	55	
ettings	50	
ettings elemetry config <u>New</u>	50	
ettings elemetry config <u>New</u> etting Started		
ettings elemetry config <u>New</u> etting Started	50	
ettings elemetry config <u>New</u> etting Started	50 45 40	
ettings elemetry config <u>New</u> etting Started	50 45	
stights ettings elemetry config <u>New</u> etting Started hat's new	50 45 40 35	854

Figure 7-1 Graphical Display of Amazon CloudWatch before the Secondary OTA Update

This is the initial state before the secondary OTA update is run.



7.2 Executing the OTA Update of the FPB-RX140

7.2.1 Creating the Update Firmware

(1) Changing the source code of the demo_app_rx140_fpb_w_buffer project

Set the MEASURE_TEMPERATURE macro of demo_app_rx140_fpb_w_buffer/src/fwupcomm_demo_main.h to (1).

32	#define MEASURE_HUMIDITY #define MEASURE_TEMPERATURE	(1) (1)
34	#define DEMO_VER_MAJOR	(1)
36	#define DEMO_VER_MINOR #define DEMO_VER_BUILD	(0) (0)

(2) Creating the update firmware

Click "Open view" in Create initial firmware.

👚 🤇 🙅 QE for OTA						
Cloud	$\rangle \odot$	Prepare	\rangle \odot	IoT	$\diamond \oslash$	OTA
1.Cloud Settings	-		Create	update firmware		
Sign-in to Cloud						
2.Prepare projects	 Creat 	e update firmware for update via C	loud			
Select projects					Create update firmwar	e Open view
Select provisioning						
3.Manage IoT device	-					
✓ Manage IoT device		Create update firmware : As	symmetric keys(AWS)			
⊘ Create initial firmware		Build update firmware				
✓ Write program to IoT devices		Push [Open View] button.				
4.OTA	-	Build update firmware				
🔗 Create update firmware		1 Register IoT devices to crea				
Execute OTA and check status	-	Push [Add] button to create		of [all IoT devices].		
		To register all IoT devices,				
		2 Specify version of the unda	te firmware.			

In the "Update Firmware" tab, select the IoT device created 6.2.3(5) and click "►Add".

All IoT devices		IoT Device Initial	Firmware Update Firmware	e OTA Fi	rmware Log	
🕂 👻 🗶 🖛 Sync Cl	oud	(1) Select IoT dev	vices for which want to upda	te the firmwa	re.	
iot_device_ck-rx65n		🗱 Add all	IoT Device Name	Private Key	Public Key	Device Certificate
		D Add	iot_device_ck-rx65n	0	0	0
		Delete				
		🗱 Delete all				
		🏇 Debug				
		W. Debug				



Enter "1.0.0" for the version and click "Create update firmware."

(2) Specify version of firmware.
V 2 🗘 0 🗘 0 🗘
(3) Create update firmware using latest source.
Create update firmware

The demo_app_rx140_fpb_w_buffer project build is executed, and if "Create firmware is complete." is displayed in the "Update Firmware" tab, the process is successful.



7.2.2 Creating an OTA Job (1) Creating an OTA job

In the "OTA" tab, select the IoT device created 6.2.3(5) and click "►Add".

All IoT devices	IoT Device Initial Firmware Update Firmware OTA Firmware Log
🖶 🖛 🗙 🕶	nc Cloud (1) Select IoT devices for execution OTA.
iot_device_ck-rx65n	Add all IoT Device Name Policy OTA Job Status Add iot_device_ck-rx65n Image: Character of the status Delete Image: Character of the status

Select "v2.0.0" as the version for update firmware, and confirm that the role, upload to, and code signing profile have been specified.

(2) Select version for up	odate firmware.	
(3) Select role, upload t	to, code signing profile for execution OTA	. Not crea
Role :	qe_ota_role	~
Upload to	qe-bucket	~
Code signing profile :	qe_code_signing	~
(4) Execute OTA.		
Execute OTA File ty	ype: 1	

Enter "1" for the file type and click "Execute OTA."

(3) Select role, upload to, code signing profile for execution OTA. Not created and the signing profile is the signing profile is the signing of the signing profile is the signing of the	Role : qe_ota_role ~ Upload to : ge-bucket ~	v2.0.0 ~		
Upload to : ge-bucket	Upload to : qe-bucket v	(3) Select role, upload t	o, code signing profile for exec	ution OTA. Not crea
	Code signing profile : qe_code_signing ~	Role :	qe_ota_role	~
Code signing profile : qe_code_signing ~		Upload to :	qe-bucket-	~
	(4) Execute OTA.	Code signing profile :	qe_code_signing	~

If "Create the OTA execution job is completed." is displayed, the operation was successful. An OTA job for the secondary OTA update is created by following the above steps, and the OTA job is delivered to the specified IoT device.

(4) Execute OTA.
Execute OTA File type : 1 🖨
- 3/3 : Create OTA execution job Complete Create the OTA execution job is completed.
When IoT device is connected to the cloud, firmware update will start. You can confirm the execution status on "Firmware Log" tab.



7.2.3 Checking Operation during Execution of the Secondary OTA Update

The OTA update starts within a few seconds after creation of the job. Both the CK-RX65N and FPB-RX140 will output logs of the progress of the secondary OTA update.

COM Port: COM47 ~	Disconnect Connected to COM47.	😏 🕂 🔳 🍇 😕 [
 AWS CLI Data (0) Complete (0) Connection Complete (0) Error (0) Fleet Thing Name (0) Complete (0) WCU -> AWS Error (1) AWS -> MCU Error (0) OTA Download (0) Swap Bank (0) 	 51718 6689787 [MQTT] [INFO] R_WIFLDA16XXX_ReceiveSocket: so 51719 6689797 [MQTT] [INFO] R_WIFLDA16XXX_ReceiveSocket: so 51720 6689797 [MQTT] [INFO] R_WIFLDA16XXX_ReceiveSocket: so 51721 6689797 [MQTT] [INFO] De-serialized incoming PUBLISH pa 51722 6689797 [MQTT] [INFO] State record updated. New state=M 51723 6690249 [MQTT] [WARN] WARN: Received an unsolicited pi 51724 6690249 [MQTT] [WARN] R_WIFLDA16XXX_ReceiveSocket: so 51725 6690249 [MQTT] [INFO] R_WIFLDA16XXX_ReceiveSocket: so 51726 6690249 [MQTT] [INFO] Publishing message to \$aws/things/ 51726 6690249 [MQTT] [INFO] SendSocket: 51731 6690274 [MQTT] [INFO] R_WIFLDA16XXX_SendSocket: socket 51735 6690315 [MQTT] [INFO] R_WIFLDA16XXX_SendSocket: socket 51736 6690315 [MQTT] [INFO] R_WIFLDA16XXX_SendSocket: socket 51736 6690315 [MQTT] [INFO] R_WIFLDA16XXX_SendSocket: socket 51739 6690342 [MQTT] [INFO] R_WIFLDA16XXX_SendSocket: socket 51739 6690343 [OTA Agent T] [INFO] Sent PUBLISH packet to broket 	ocket 0 recv_cnt=157 (0). Incket: DeserializerResult=MQTTSuccess. IQTTPublishDone. Ublish from topic \$aws/things/iot_device_ck-rx65n/jobe timeout! Ocket 0 recv_cnt=0 (450). (iot_device_ck-rx65n/streams/ab84caba-69b8-43da-ab et 0 ret=33 (25). et 0 ret=114 (41). et 0 ret=55 (27).
- Version (0)	<	>
	COM Port : COM38 V Disconnect Connected to COM	M38.



7.3 Checking Operation after the OTA Update

Figure 7-2 shows the log screen of the CK-RX65N and FPB-RX140 after the update.

You can see that the temperature data are newly displayed in addition to the humidity data acquired from the HS3001 sensor.



Figure 7-2 Log Screen of the CK-RX65N and FPB-RX140 after the Firmware Update

Furthermore, LED2 will now also be blinking in addition to LED1 that was blinking in the initial state, in the FPB-RX140.



Finally, Figure 7-3 shows the display for Amazon CloudWatch. Confirm that the measured temperature and humidity data acquired from the HS3001 sensor are displayed as a graph.

CloudWatch > Logs Insights		O Discovered
CloudWatch <	sensor-visualize X	fields
Favorites and recents	<pre>1 stats avg(hs300x_humidity), avg(hs300x_temperature) by bin(1m)</pre>	D
Dashboards		Queries
▶ Alarms A 0 ⊘ 0 ⊡ 0	Run query Cancel Save History	
▼ Logs	Logs Insights QL query can run for maximum of 60 minutes.	
Log groups	O Completed. Query executed for 1 log group. ()	
Log Anomalies	Logs (6) Patterns (-) Visualization	
Live Tail	Visualization	Add to dashboard
Contributor Insights	Graph type: Line 🔻	
Metrics		1. avg(hs300x_humidity)
Application Signals <u>New</u> (APM)	60	 1. avg(hs300x_numidity) 2. avg(hs300x_temperature)
Network Monitoring	50	
Insights	<i></i>	
Settings	40	
Telemetry config New	30	
Getting Started	30	
What's new	20	
	10	
	0	
	03:54 03:54 03:54 03:55 03:55 03:55 03:55 03:56 03:56 03:56 03:56 03:57 03:57 03:57 03:57 03:58 03:58 03:58 03:58	9
CloudShell Feedback	© 2025, Amazon Web Services, Inc. or its affiliates. Pi	ivacy Terms Cookie preferences

Figure 7-3 Graphical Display of Amazon CloudWatch after the Secondary OTA Update

Operations for the demonstration are completed at this point.

Note : In 7.2.2(1), if the update method for the project subject to secondary OTA updates is "partial update method," set the File type value to "1." If it is "full update method," set the File type value to "2."



8. How to do the demo without using the HS3001 sensor

This section describes how to perform a secondary OTA update demo without connecting the HS3001 sensor to the FPB-RX140. In this case, the following sensor-related functions in the demo won't work.

- Acquire temperature and humidity data from the HS3001 sensor connected to the FPB-RX140 and output it to a log.
- Send temperature and humidity data from CK-RX65N to AWS and display it on AWS.

8.1 Changes to the demo procedure

Please change the following steps in the demo procedure for sections 6 and 7. The other steps are the same as when using the sensor.

- (1) Skip the step "6.1.2(1) Connecting the HS3001 board."
- (2) After completing "6.2.4(2) Selecting projects" change the MEASURE_HUMIDITY macro in demo_app_rx140_fpb_w_buffer/src/fwupcomm_demo_main.h to (0).
- (3) Skip the step "6.3 Preparations for Displaying the Sensor Data Using the AWS Cloud."
- (4) Skip the step "7.2.1(1) Changing the source code of the demo_app_rx140_fpb_w_buffer project."

8.2 How to check demo operation when not using sensors

If the sensor is not used, it is not possible to confirm whether the firmware update has been performed by the type of sensor data acquired. Please check the version information displayed in the log when the FPB-RX140 starts up.

<Before updating>

Secondary	*
COM Port : COM38 V Disconnect Connected to COM38.	
	^
==== RX140 : BootLoader [with buffer] ==== verify install area main [sig-sha256-ecdsa]OK execute image	
==== RX140 : SECONDARY OTA DEMO [with buffer] ver 1.0.0 ====	
<	, ×

<After updating>





9. Precautions

9.1 License Information on the Open-Source Software in Use

The following open-source software is used.

- TinyCrypt Cryptographic Library
 - URL: <u>https://github.com/intel/tinycrypt</u>
 - --- License: <u>https://github.com/intel/tinycrypt/blob/master/LICENSE</u>
- FreeRTOS
 - URL: <u>https://www.freertos.org/</u>
 - License: <u>FreeRTOS open source licensing</u>, <u>FreeRTOS license description</u>, <u>FreeRTOS license terms</u> and <u>OpenRTOS commercial licensing options</u>.

9.2 Region and User Privileges of AWS for the Demonstration

Regarding the setup of AWS for running the demonstration, notes on the region of use and user privileges are given below.

<Region of use>

This demonstration is provided in the ap-northeast-1 (Asia Pacific (Tokyo)) region of AWS.

If you want to run this demonstration in another region, confirm that the services used in the demonstration are available in that region beforehand.

<User privileges>

This demonstration is to be run by a user with Administrator Access permission in the AWS Identity and Access Management (IAM) system. Therefore, there is no particular description regarding the granting of necessary permissions in IAM when using various services.

9.3 Fees for Using AWS

A charge may apply to the cloud resources created and used in the demonstration depending on how AWS is used. To avoid inadvertently incurring charges, deleting the resources created in the cloud after running the demonstration is recommended.



Revision History

Description		n	
Rev.	Date	Page	Summary
1.01	Jan. 24, 2022		First edition issued.
1.10	Mar. 31, 2022 — Supported AWS IoT Over-		Supported AWS IoT Over-the-air Update Library v3.0.0.
		5 - 8	Added .settings folder to the folder structure of each project.
		5 - 8	Revised package and folder structure for RX65N project.
		9	Updated IDE environment to e2studio 2022-01, Toolchain to
			CC-RX V3.04.00 and FreeRTOS for RX65N project to Version 2021.07.
		9	Updated code size.
		19 - 20	Updated initial firmware creation method due to RX65N project changes.
		47	Updated screenshot of output log due to RX65N project changes.
		51 - 55	Changed the method of executing an update.
		56	Updated screenshot of output log due to RX65N project
			changes.
2.00	Mar. 31, 2024	_	The used boards were changed to the CK-RX65N and TB-RX660.
		—	The FreeRTOS package for the projects for the RX65N was updated.
			The version of the firmware update module (FWUP) using Firmware Integration Technology (FIT) for the projects for the RX660 was updated.
			The entire application note was revised due to changes in the used boards and projects.
3.00	2025/06/30		The used boards were changed to the CK-RX65N v2 and FPB-RX140 v1.
			Changed communication control between microcontrollers to Firmware Updating Communications Module.
		—	Changed the demo procedure to use the secondary OTA function of QE for OTA V2.2.0.



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 reseting 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 systemevaluation test for the given product.

Notice

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

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

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

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