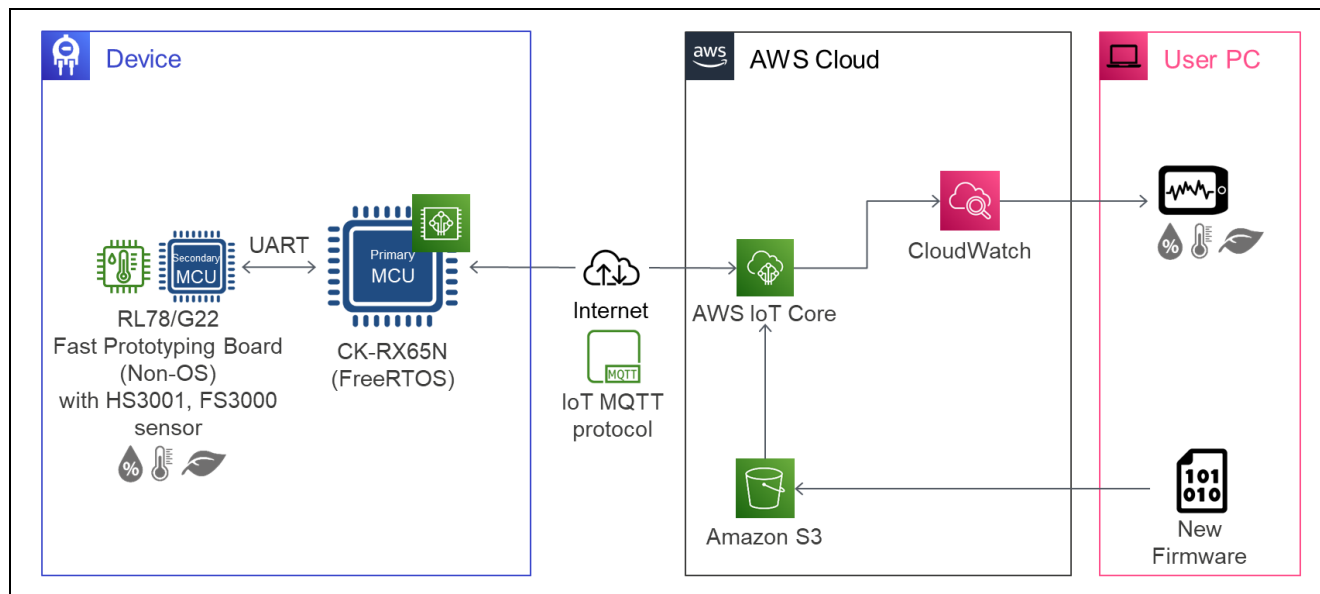


RL78/G22

セカンダリ MCU の OTA ファームウェアアップデート

要旨

本アプリケーションノートでは、Amazon Web Services™（以下、AWS と略します）との通信を行うプライマリ MCU の RX65N を用いて、センサデータの測定機能を持ったセカンダリ MCU の RL78 マイコンに対して、AWS のサービスを利用した OTA ファームウェアアップデート（以下、セカンダリ OTA アップデートと略します）を行うデモについて説明します。



動作確認デバイス

RX65N
 RL78/G22
 センサ
 HS3001 高性能 相対湿度・温度センサ
 FS3000 Air Velocity Sensor Module

動作確認ボード

プライマリ MCU: CK-RX65N (RTK5CK65N0S04000BE)
 セカンダリ MCU: RL78/G22 Fast Prototyping Board (RTK7RLG220C00000BJ)
 センサ
 温湿度センサ Pmod™ Board (US082-HS3001EVZ)
 空気速度センサ Pmod™ Board (US082-FS3000EVZ)

関連文書

[RL78/G22 ユーザーズマニュアル ハードウェア編 \(R01UH0978\)](#)

[RL78 ファミリ ユーザーズマニュアル ソフトウェア編 \(R01US0015\)](#)

[RL78/G22, RL78/G23, RL78/G24 ファームウェア アップデート モジュール \(R01AN6374\)](#)

[RL78/G22 Fast Prototyping Board ユーザーズマニュアル \(R20UT5121\)](#)

[RX65N Group CK-RX65N v1 User's Manual \(R20UT5100\)](#)

[RX65N グループ FreeRTOS を用いた Amazon Web Services によるセカンダリデバイスの OTA アップデートサンプルコード \(R01AN6220\)](#)

(最新版をルネサス エレクトロニクスホームページから入手してください。)

テクニカルアップデート／テクニカルニュース

(最新の情報をルネサス エレクトロニクスホームページから入手してください。)

目次

1. 概要	5
2. 動作確認条件	5
3. ハードウェア説明	6
3.1 システム構成	6
3.2 使用端子一覧	6
4. ソフトウェア説明	7
4.1 ファームウェアアップデート方式	7
4.1.1 半面更新方式	7
4.1.2 全面更新方式	9
4.2 オプション・バイトの設定一覧	11
4.3 フォルダ/ファイル構成	11
4.4 コードサイズ	11
5. デモの動作説明	12
6. デモのセットアップ	12
6.1 ハードウェアのセットアップ	12
6.1.1 全体の構成	12
6.1.2 CK-RX65N のセットアップ方法	13
6.1.3 RL78/G22 FPB のセットアップ方法	15
6.2 ソフトウェアのセットアップ	19
6.2.1 事前準備	19
6.2.2 ターミナルソフトの設定	19
6.2.3 CK-RX65N 用の初期ファームウェアの作成と実行	20
6.2.4 RL78/G22 FPB 用の初期ファームウェアの作成と実行	24
6.3 AWS クラウドの準備	30
6.3.1 OTA アップデートのための設定	30
6.3.2 センサデータ可視化のための設定	31
6.3.2.1 Amazon CloudWatch の設定	31
7. デモの実行手順	38
7.1 初期状態の動作確認	38
7.2 RL78/G22 FPB の OTA アップデートの実行	40
7.2.1 更新ファームウェアの作成	40
7.2.1.1 半面更新方式の場合	40
7.2.1.2 全面更新方式の場合	40
7.2.2 AWS で OTA ジョブの作成	41
7.2.3 セカンダリ OTA アップデート実行中の動作確認	46
7.3 OTA アップデート後の動作確認	47
8. 注意事項	49
8.1 使用するオープンソースソフトウェアのライセンス情報	49

8.2	デモを実施する AWS のリージョンとユーザ権限について	49
8.3	AWS の利用料金について	49
	改訂記録	50

Amazon Web Services、“Powered by AWS”ロゴ、およびかかる資料で使用されるその他の AWS 商標は、米国および/またはその他の国における Amazon.com, Inc. またはその関連会社の商標です。

FreeRTOS および FreeRTOS は Amazon Web Services, Inc. の商標です。

Pmod は Digilent Inc. の商標です。

すべての商標および登録商標は、それぞれの所有者に帰属します。

1. 概要

デモでは、セカンダリ OTA アップデートによって稼働センサを追加し、ブラウザ上の AWS 画面でセンサデータを可視化することで稼働センサの追加を確認できます。

IoT 機器にはセキュリティの脆弱性の適宜修正や、お客様要望に応じた機能のアップデートが求められます。従来、提供しているプライマリ MCU の OTA アップデートだけでなく、セカンダリ OTA アップデートを実現することで、セカンダリ MCU の脆弱性への対応やフレキシブルなサービスのアップデートが可能な製品開発を実現できます。

2. 動作確認条件

本アプリケーションノートのサンプルプログラムは以下に示す条件でデモの動作を確認しています。

表 2-1 デモ動作確認条件 (RL78/G22)

項目	内容
使用マイコン	RL78/G22 (R7F102GGE2DFB)
使用ボード	RL78/G22 Fast Prototyping Board (RTK7RLG220C00000BJ)
動作周波数	高速オンチップ・オシレータ・クロック: 32MHz
動作電圧	3.3V
統合開発環境	ルネサス エレクトロニクス製 e² studio V2024-01 (24.1.0)
C コンパイラ	ルネサス エレクトロニクス製 CC-RL V1.13.00
ファームウェア書き込みツール	Renesas Flash Programmer V3.14.00
スマート・コンフィグレータ(SC)	Renesas Smart Configurator for RL78 24.1.0.v20231218-0132
ボードサポートパッケージ(BSP)	v1.62 (r_bsp)

表 2-2 デモ動作確認条件 (センサ)

項目	内容
温湿度センサボード	US082-HS3001EVZ Board
空気速度センサボード	US082-FS3000EVZ Board

表 2-3 デモ動作確認条件 (RX65N)

項目	概要
使用マイコン	RX65N (R5F565NEHDFB)
使用ボード	CK-RX65N v1 (RTK5CK65N0S04000BE)
統合開発環境	e² studio V2024-01 (24.1.0)
C コンパイラ	CC-RX V3.06.00
RTOS	FreeRTOS v202210.01-LTS-1.1.3

表 2-4 デモ動作確認条件 (その他)

項目	内容
QE for OTA	V2.00
Python	3.10.4

QE for OTA は、<https://www.renesas.com/qe-ota/> から入手できます。

Python は、<https://www.python.org/> から入手できます。

3. ハードウェア説明

3.1 システム構成

本システムは、AWS との通信制御機能を持つ RX65N マイコン(プライマリ MCU)と、HS3001 センサと FS3000 センサが接続された RL78 マイコン(セカンダリ MCU)で構成されています。2つのマイコンは UART による相互通信を行います。

本デモでは、UART 通信を介して、センサが接続された RL78 マイコンのセカンダリ OTA アップデートやセンサボードで取得したセンサデータのクラウドへのアップロードとデータの可視化ができます。

図 3-1 にシステム構成を示します。

セカンダリ MCU の RL78 マイコンとして、RL78/G22 を搭載した RL78/G22 Fast Prototyping Board (以下、RL78/G22 FPB と略します)を使用します。

プライマリ MCU として、RX65N を搭載した CK-RX65N を使用します。

また、CK-RX65N と RL78/G22 FPB を UART 接続します。

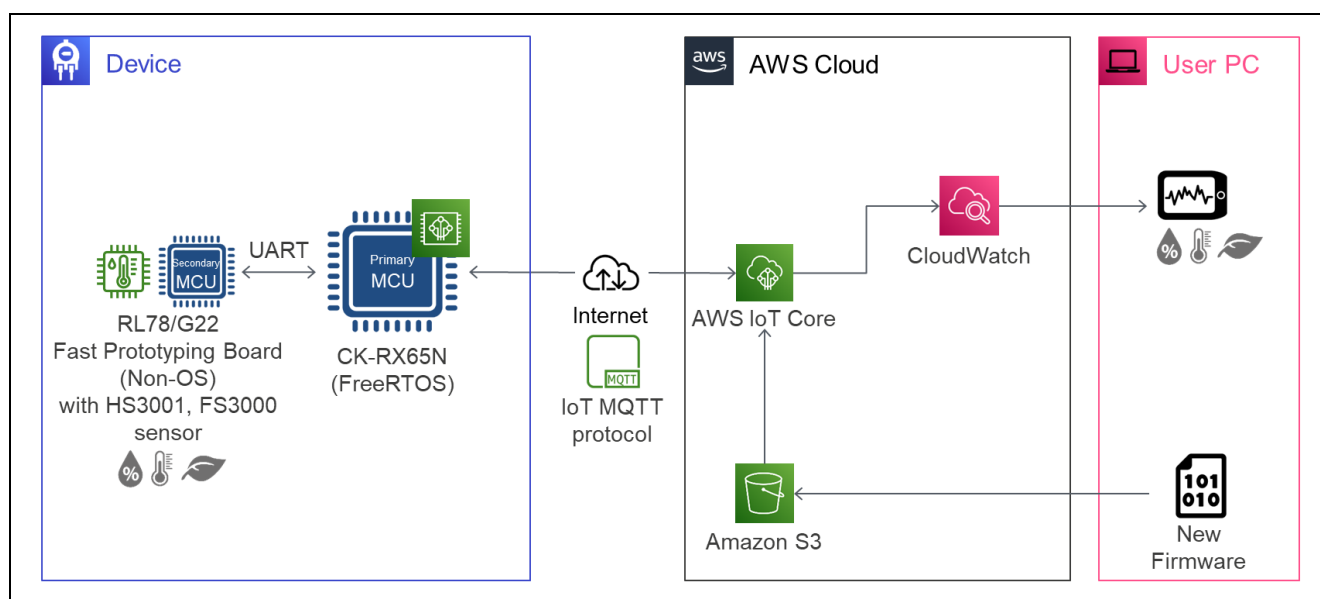


図 3-1 本デモのシステム構成

3.2 使用端子一覧

表 3-1 に RL78/G22 マイコンの使用端子と機能を示します。

表 3-1 使用端子と機能

端子名	入出力	内容
P12/TxD0	出力	PC へのログ出力
P00/TxD1	出力	RX65N との UART 通信(送信)
P01/RxD1	入力	RX65N との UART 通信(受信)
P70/SCL21	出力	センサとの I2C 通信(クロック)
P71/SDA21	入出力	センサとの I2C 通信(データ)

注意 本アプリケーションノートは、使用端子のみを端子処理しています。実際に回路を作成される場合は、端子処理などを適切に行い、電気的特性を満たすように設計してください。

4. ソフトウェア説明

AWS 認定済プログラムを使用した RX65N のファームウェアには、FreeRTOS™ with IoT Library が実装されています。そのため、AWS が提供しているマネージドサービスである AWS IoT Core および AWS IoT Device Management を利用して、OTA によるファームウェアアップデートや MQTT 通信によるクラウドへのデータのアップロードが実行可能です。

セカンダリ OTA アップデートの制御には、プライマリ MCU 側の RX65N マイコンでは AWS IoT Over-the-air Update Library を利用し、AWS から受信した RL78 用の更新ファームウェアをさらに RL78 マイコンに送信し、ファームウェア更新を実現しています。

セカンダリ MCU 側の RL78 マイコンのファームウェアアップデートの制御は「[RL78/G22,RL78/G23,RL78/G24 ファームウェア アップデート モジュール Rev.2.01](#)」を使用します。

4.1 ファームウェアアップデート方式

本アプリケーションノートでは、ファームウェアアップデートモジュールが提供している方式のうち、「半面更新方式（バッファ面は内蔵フラッシュ）」（以下、「半面更新方式」と略します）と「全面更新方式（バッファ無し）」（以下、「全面更新方式」と略します）の 2 種類の方式のサンプルプログラムを提供します。これらの方式の詳細は、「[RL78/G22,RL78/G23,RL78/G24 ファームウェア アップデート モジュール Rev.2.01](#)」の 1.3 章「各ファームウェアアップデート方式について」をご参照ください。

4.1.1 半面更新方式

以下に半面更新方式のサンプルプログラムのメモリマップを示します。

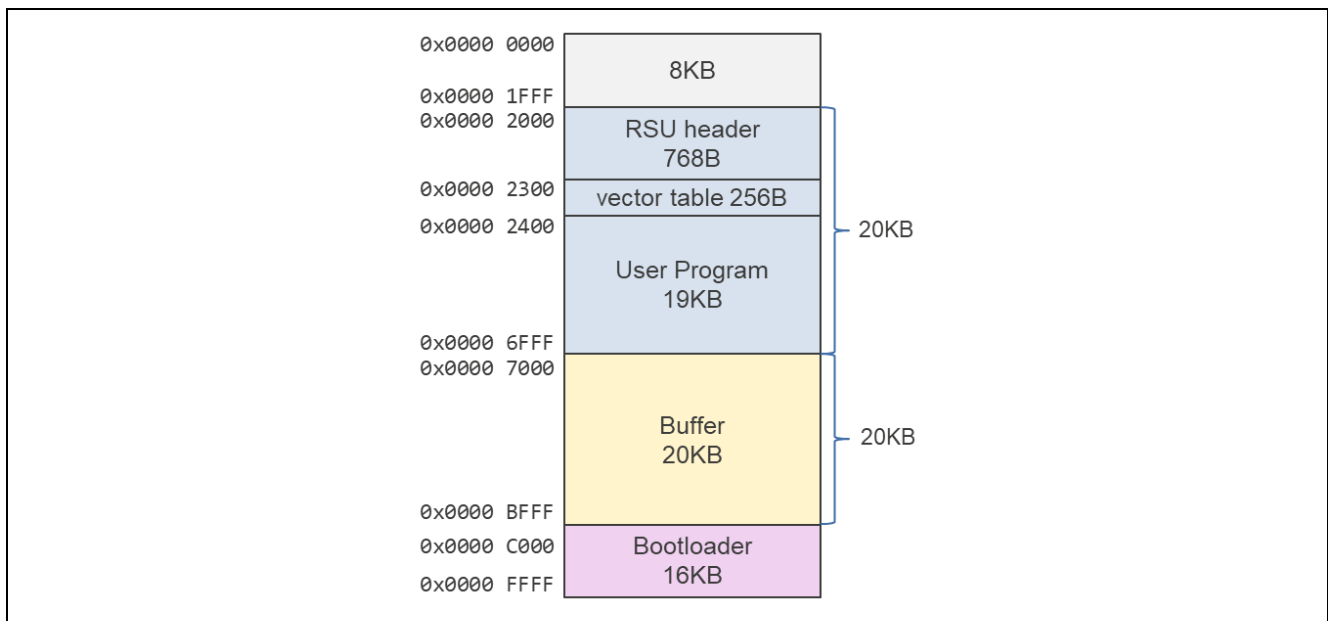


図 4-1 半面更新方式のサンプルプログラムのメモリマップ

以下の図 4-2 に半面更新方式の動作概要を示します。また、図 4-3 に OTA アップデートの各フェーズでの ROM の状態を示します。なお、図 4-3 中の赤枠はその時実行されているプログラムを表しています。

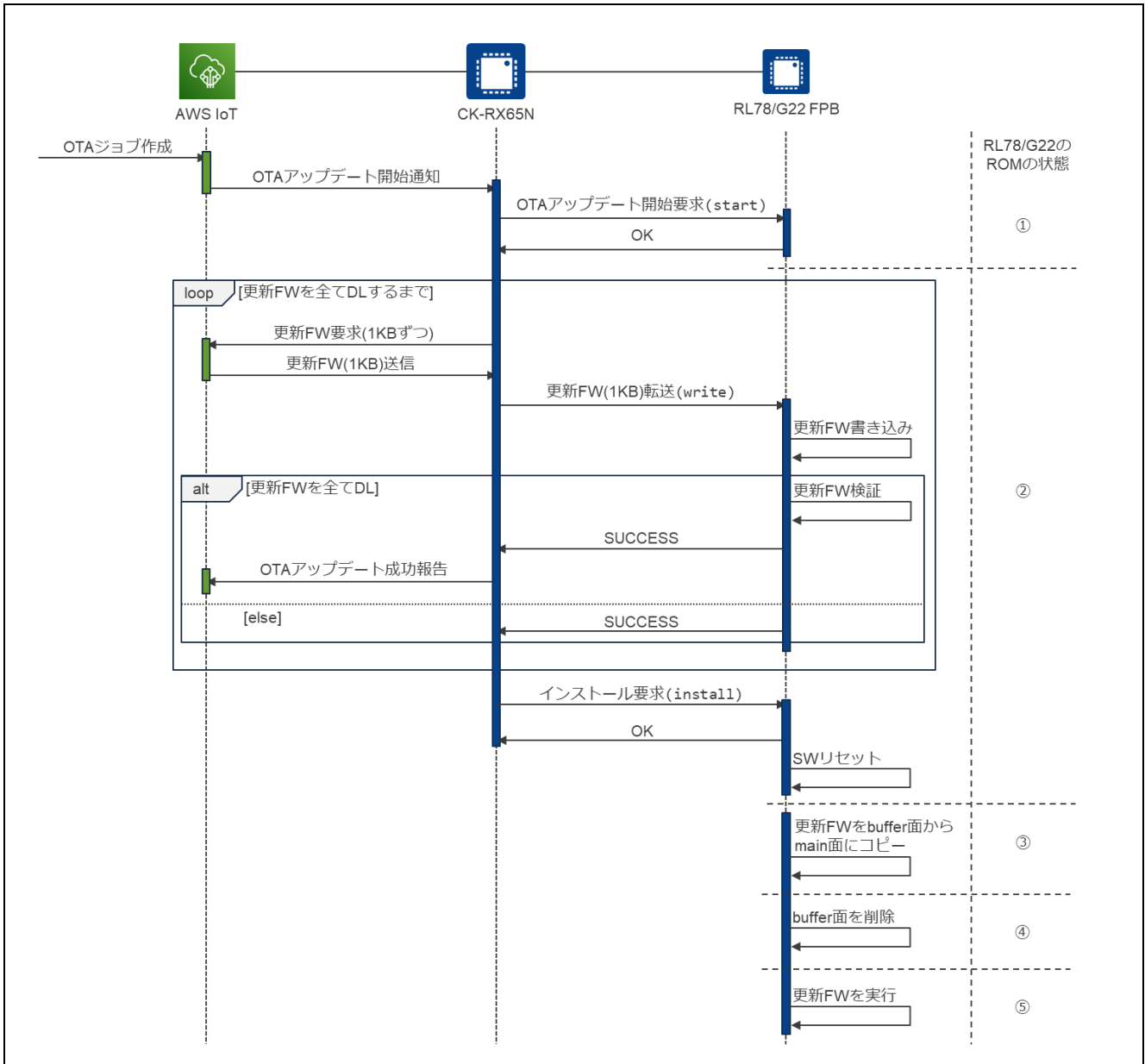


図 4-2 半面更新方式の動作概要

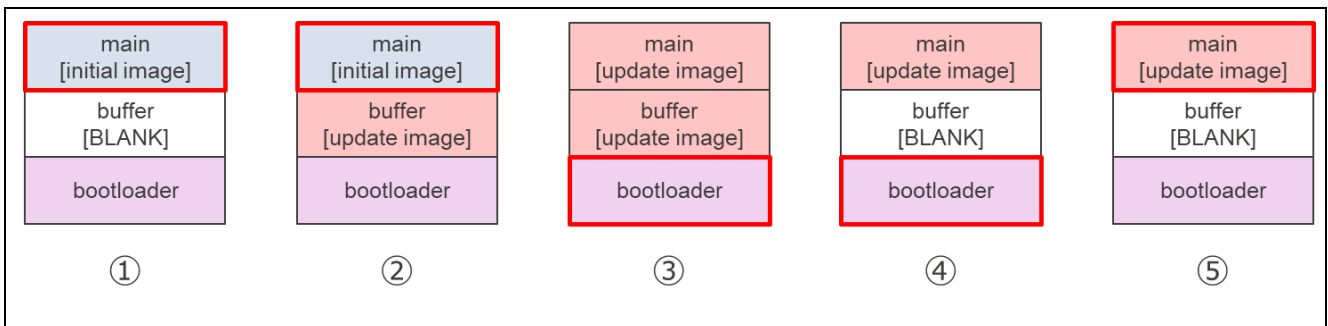


図 4-3 半面更新方式の OTA アップデートの各フェーズでの ROM の状態

4.1.2 全面更新方式

以下に全面更新方式のサンプルプログラムのメモリマップを示します。

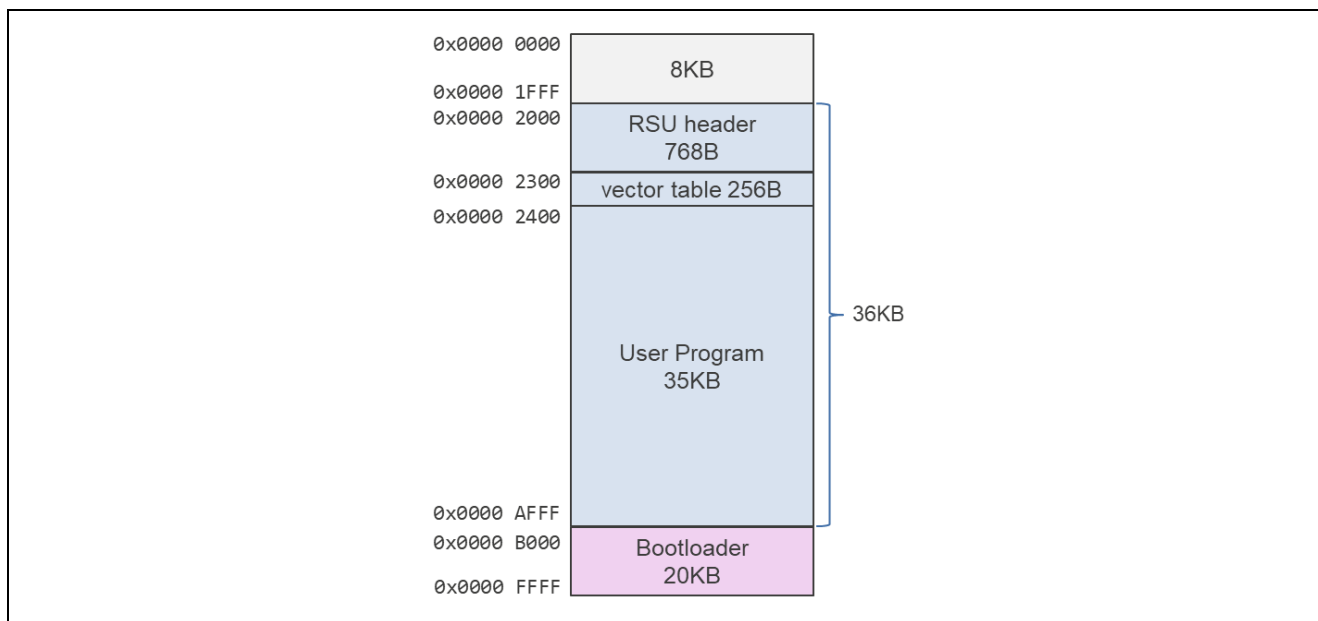


図 4-4 全面更新方式のサンプルプログラムのメモリマップ

以下の図 4-5 に全面更新方式の動作概要を示します。また、図 4-6 に OTA アップデートの各フェーズでの ROM の状態を示します。

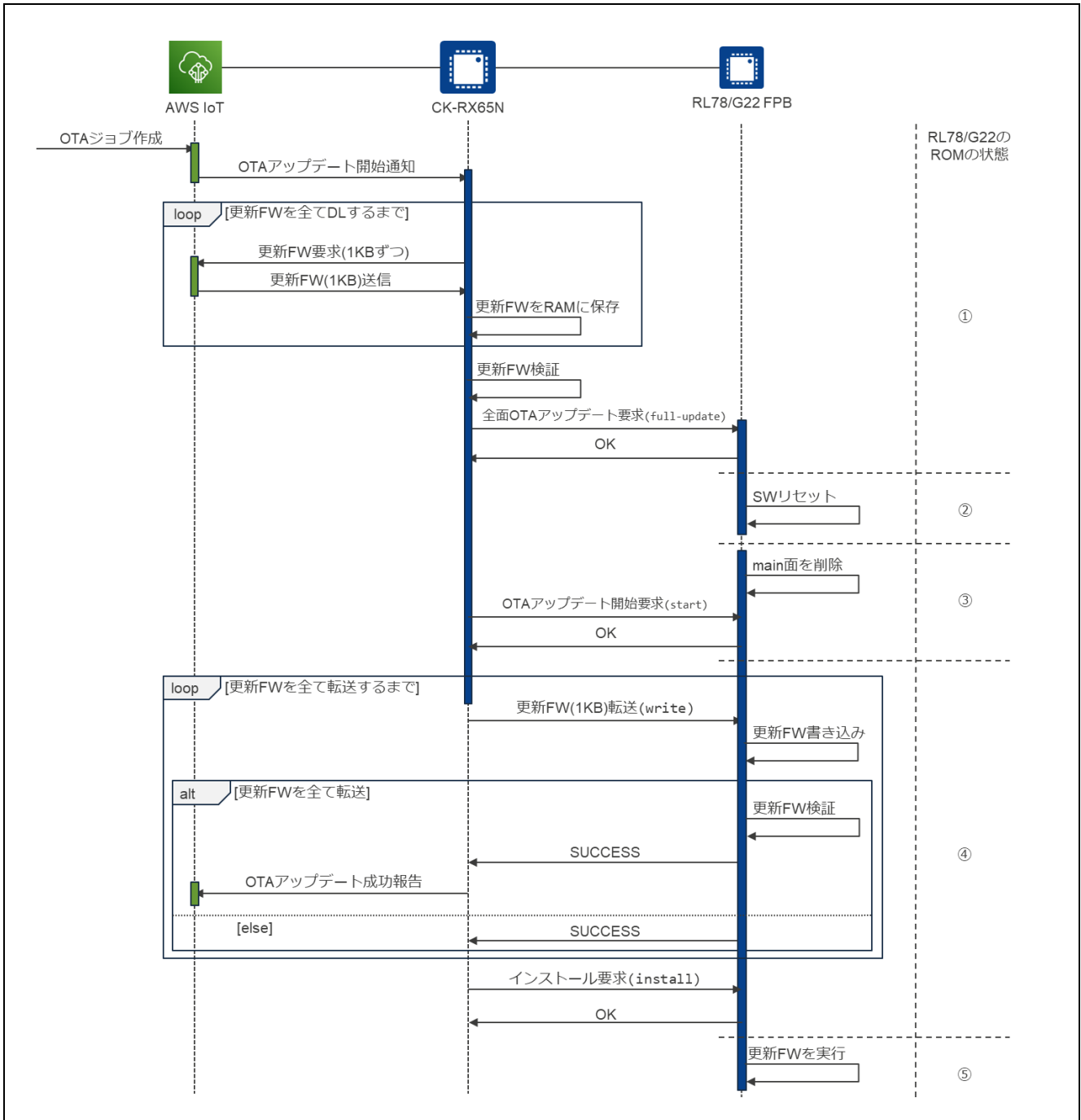


図 4-5 全面更新方式の動作概要

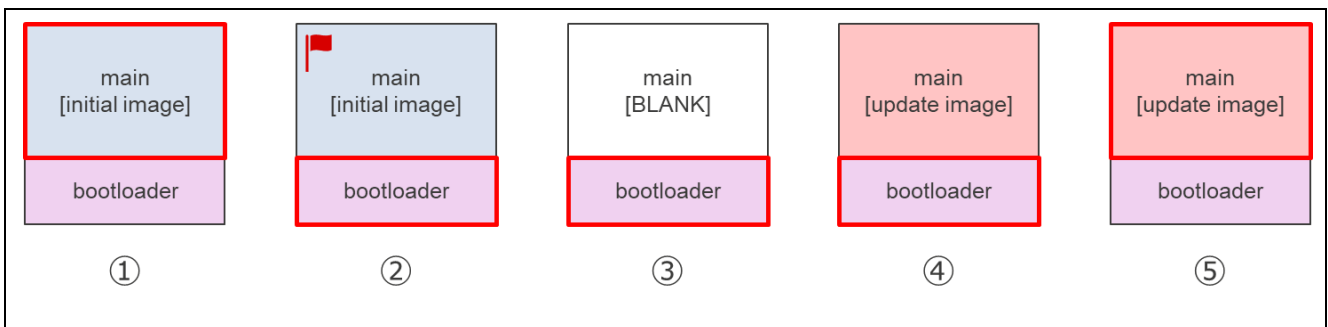


図 4-6 全面更新方式の OTA アップデートの各フェーズでの ROM の状態

4.2 オプション・バイトの設定一覧

以下にオプション・バイトの設定を示します。

表 4-1 オプション・バイト設定

アドレス	設定値	内容
000C0H / 020C0H	01101110B	ウォッチドッグ・タイマ 動作停止 (リセット解除後、カウント停止)
000C1H / 020C1H	11111110B	LVD 検出電圧: リセット・モード 立ち上がり時 TYP. 1.90 V (1.84 V ~ 1.95 V) 立ち下がり時 TYP. 1.86 V (1.80 V ~ 1.91 V)
000C2H / 020C2H	11101000B	HS (高速メイン) モード、 高速オンチップ・オシレータ・クロック : 32MHz
000C3H / 020C3H	00000100B	オンチップ・デバッグ動作禁止

4.3 フォルダ/ファイル構成

図 4-7 にサンプルプログラムのフォルダ構成を示します。

```

r01an6935xx0110-r178g22
├── Demo
│   ├── afr-v202210.01-LTS-rx-1.1.3
│   ├── r178g22_fpb_2ndota_demo
│   ├── r178g22_fpb_bootloader
│   ├── r178g22_fpb_2ndota_demo_full
│   ├── r178g22_fpb_bootloader_full
│   ├── ck_rx65n_2ndota_demo
│   ├── ck_rx65n_demo_bootloader
│   └── RenesasImageGenerator
├── r01an6935ej0110-r178g22.pdf
└── r01an6935jj0110-r178g22.pdf

```

図 4-7 サンプルプログラムのフォルダ構成

ck_rx65n_2ndota_demo フォルダと ck_rx65n_demo_bootloader フォルダには CK-RX65N 用のプロジェクトファイルが格納されています。

rl78g22_fpb_2ndota_demo フォルダと rl78g22_fpb_bootloader フォルダには、RL78/G22 FPB の半面更新方式用のプロジェクトファイルが格納されています。

rl78g22_fpb_2ndota_demo_full フォルダと rl78g22_fpb_bootloader_full フォルダには、RL78/G22 FPB の全面更新方式用のプロジェクトファイルが格納されています。

4.4 コードサイズ

表 4-2 にコードサイズを示します。

表 4-2 コードサイズ

更新方式	プロジェクト	ROM	RAM
半面更新方式	rl78g22_fpb_bootloader	11 KB	0.8 KB
	rl78g22_fpb_2ndota_demo	16 KB	2 KB
全面更新方式	rl78g22_fpb_bootloader_full	14 KB	2 KB
	rl78g22_fpb_2ndota_demo_full	18 KB	2 KB

5. デモの動作説明

デモの動作は半面更新方式/全面更新方式で共通です。

- (1) デモの初期状態では、RL78/G22 FPB は接続されている HS3001 センサを使って温湿度データのみ取得します。
- (2) セカンダリ OTA アップデートの仕組みを用いて、AWS から CK-RX65N 経由で RL78/G22 FPB の更新ファームウェアをダウンロードし、ファームウェアの更新を行います。
- (3) ファームウェア更新後は、RL78/G22 FPB は HS3001 センサに加えて FS3000 センサから流量データも取得します。

一連の流れで、取得しているセンサデータの種類とその値は、両マイコンからの PC へのログ出力と AWS 上のダッシュボードから確認できます。

6. デモのセットアップ

本アプリケーションノートのデモを実行するために必要なセットアップについて説明します。

CK-RX65N と RL78/G22 FPB の配線や HS3001・FS3000 センサの接続方法等のハードウェアのセットアップ、それぞれのマイコンボード用の初期ファームウェアの作成と書き込み等のソフトウェアのセットアップ、そして AWS 上でのセンサデータの可視化のための AWS クラウド側の準備が必要です。

6.1 ハードウェアのセットアップ

6.1.1 全体の構成

最初に、本デモを構成するハードウェア全体の構成を示します。以降でそれぞれのボードのセットアップ方法について詳しく説明します。

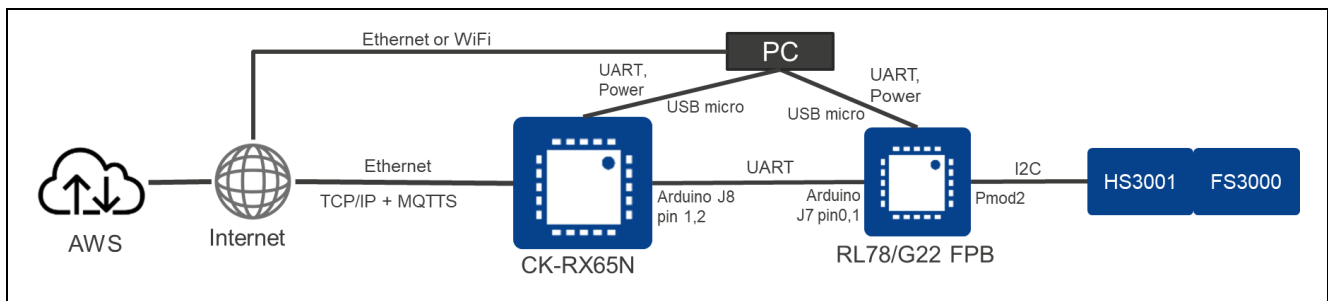


図 6-1 本デモのハードウェア全体構成

6.1.2 CK-RX65N のセットアップ方法

CK-RX65N のセットアップ方法を示します。

(1) RL78/G22 FPB との UART 通信用ケーブル接続

RL78/G22 FPB と UART 通信を行うための TXD, RXD, GND は CK-RX65N の J8, J3 コネクタの以下の端子に割り当てています。6.1.3(2)章に示す RL78/G22 FPB 側の端子と以下表のように UART 信号の対応を取って接続してください。

表 6-1 CK-RX65N と RL78/G22 FPB の UART 接続対応

CK-RX65N			RL78/G22 FPB	
UART 信号	コネクタ端子名		UART 信号	コネクタ端子名
RXD	J8-1 (RX◀0)	→	TXD	J7-2
TXD	J8-2 (TX▶1)	←	RXD	J7-1
GND	J3-7 (GND)	—	GND	J8-7

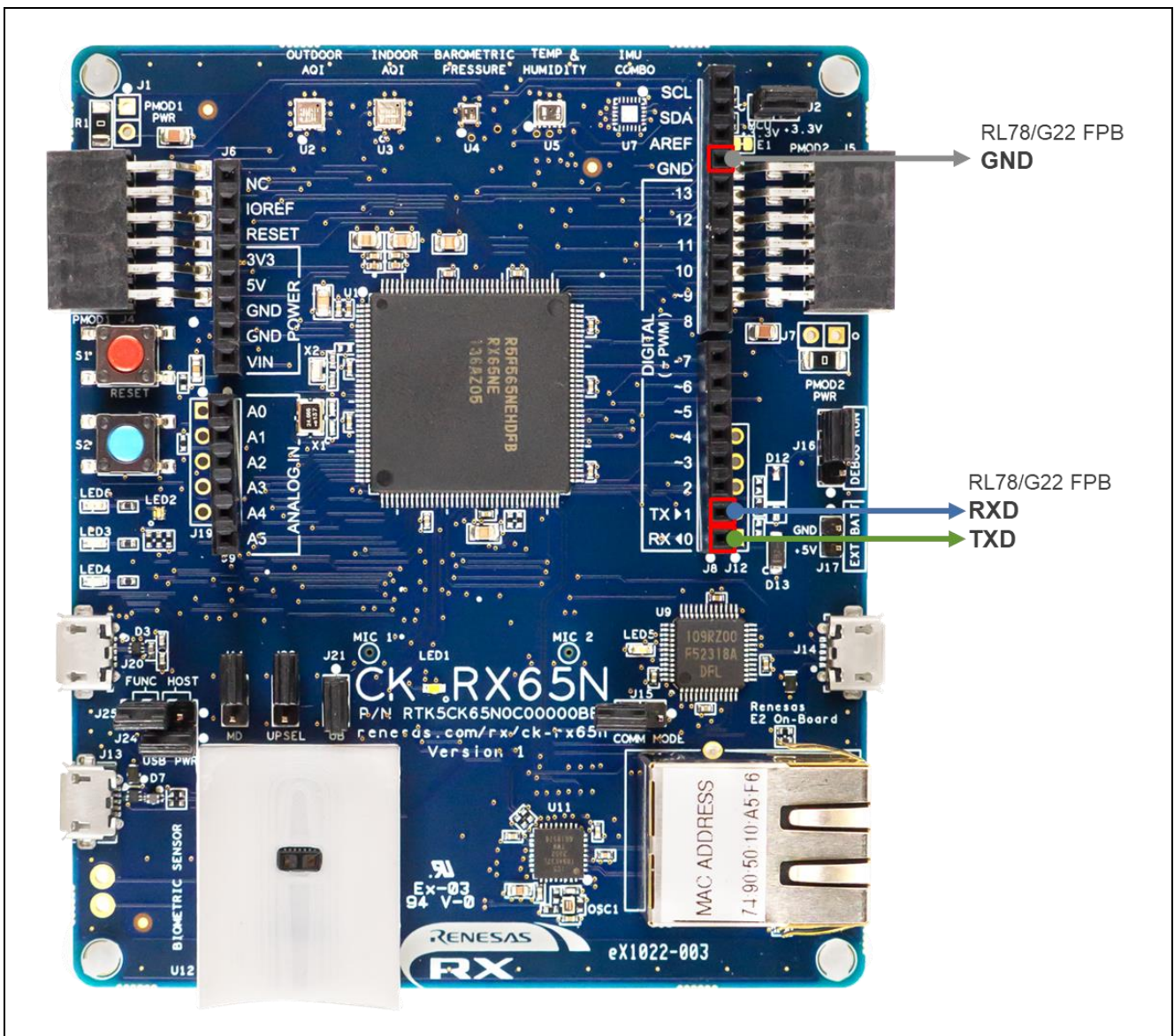


図 6-2 CK-RX65N のマイコン間 UART 通信に使用する端子位置

(2) PC へのログ出力用ケーブル接続

PC と CK-RX65N の USB シリアルコネクタ(micro USB Type-B)を USB ケーブルで接続してください。

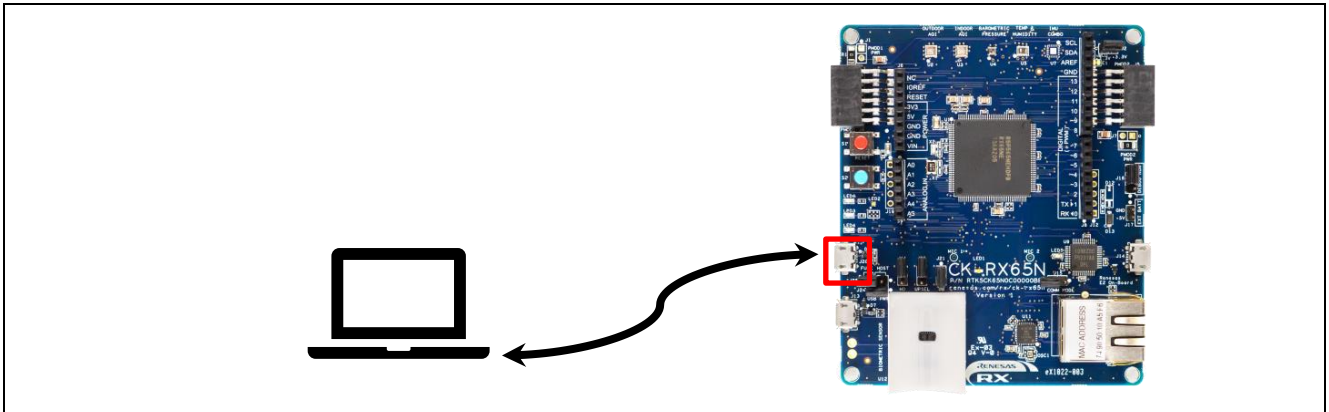


図 6-3 PC へのログ出力用の接続

(3) 電源供給・デバッガとの接続

PC と CK-RX65N の E2OB Debugger コネクタ (micro USB Type-B)を USB ケーブルで接続してください。

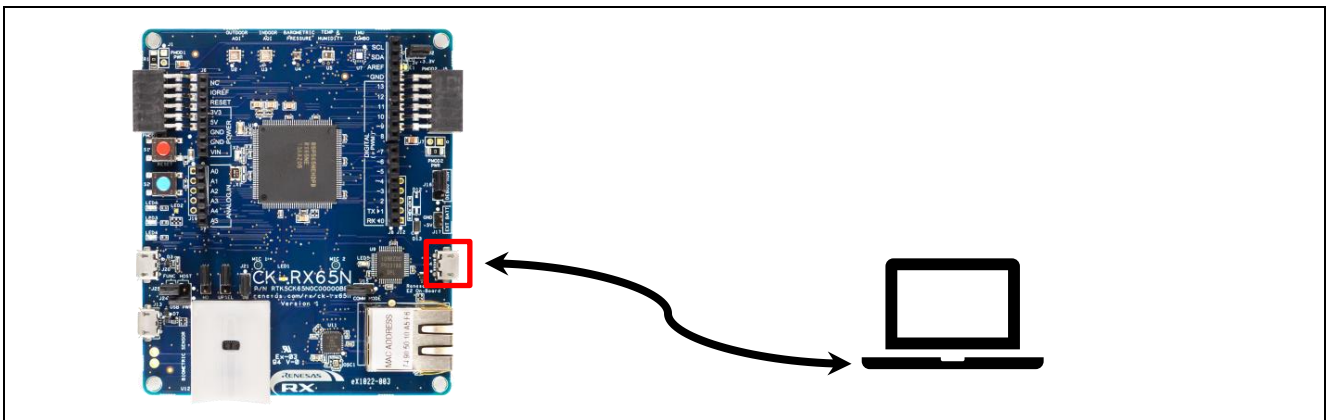


図 6-4 電源供給・デバッガ用の接続

(4) インターネット接続用の LAN ケーブル接続

CK-RX65N のイーサネットコネクタにインターネットに接続されている LAN ケーブルを接続します。

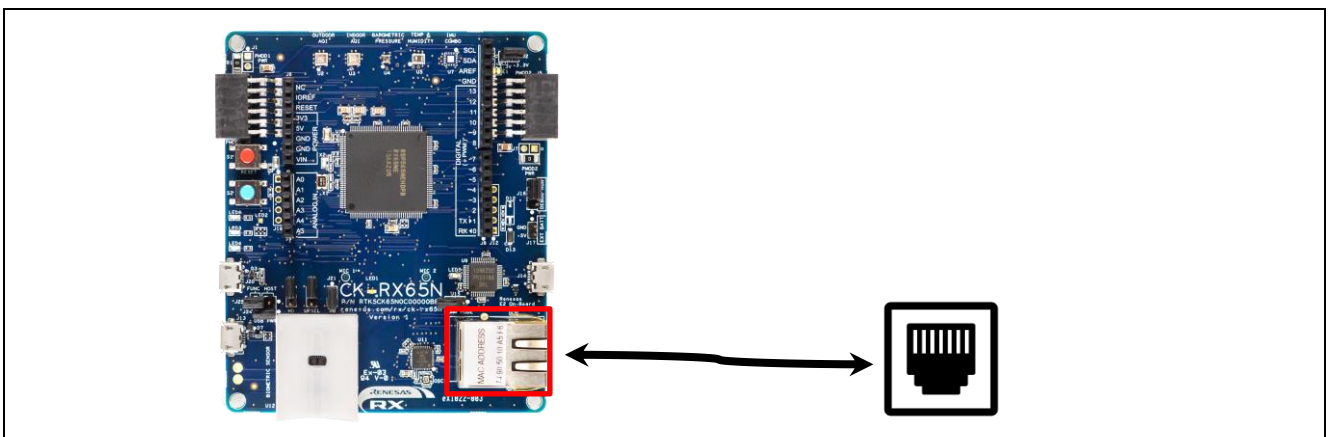


図 6-5 イーサネットによる有線インターネット接続

(5) ジャンパ J16 を DEBUG 側に短絡する

CK-RX65N をデバッグモードにするためにジャンパ J16 を DEBUG 側(pin 1-2)に短絡します。

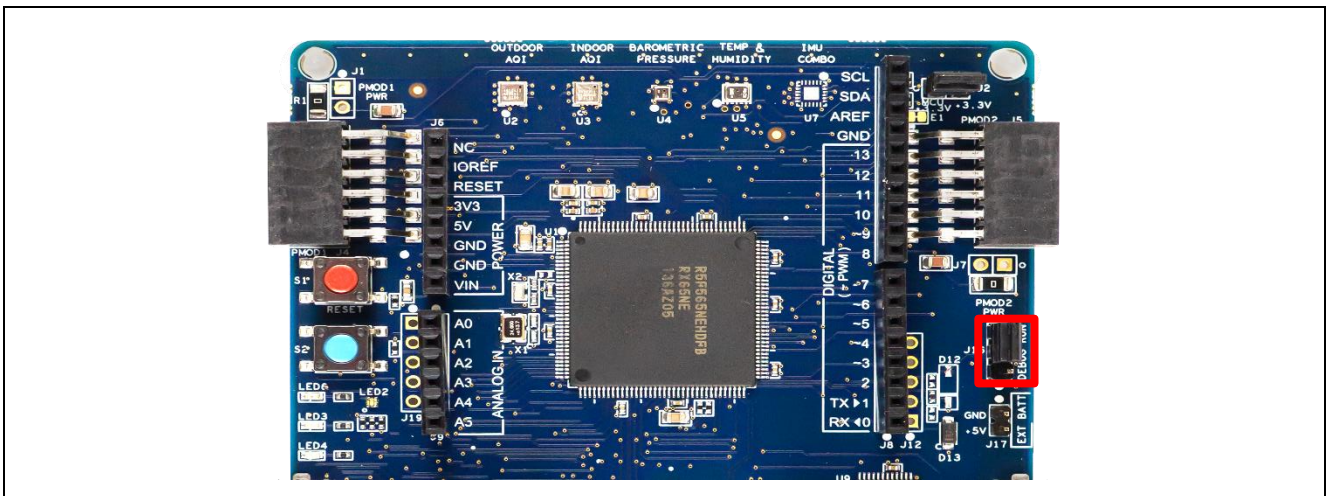


図 6-6 ジャンパ J16 の位置

6.1.3 RL78/G22 FPB のセットアップ方法

RL78/G22 FPB のセットアップ方法を示します。

(1) 動作電圧を 3.3V に設定

RL78/G22 を 3.3V 電源で動作させるために、電源選択ヘッダ(J17)の 2-3 側を短絡します。

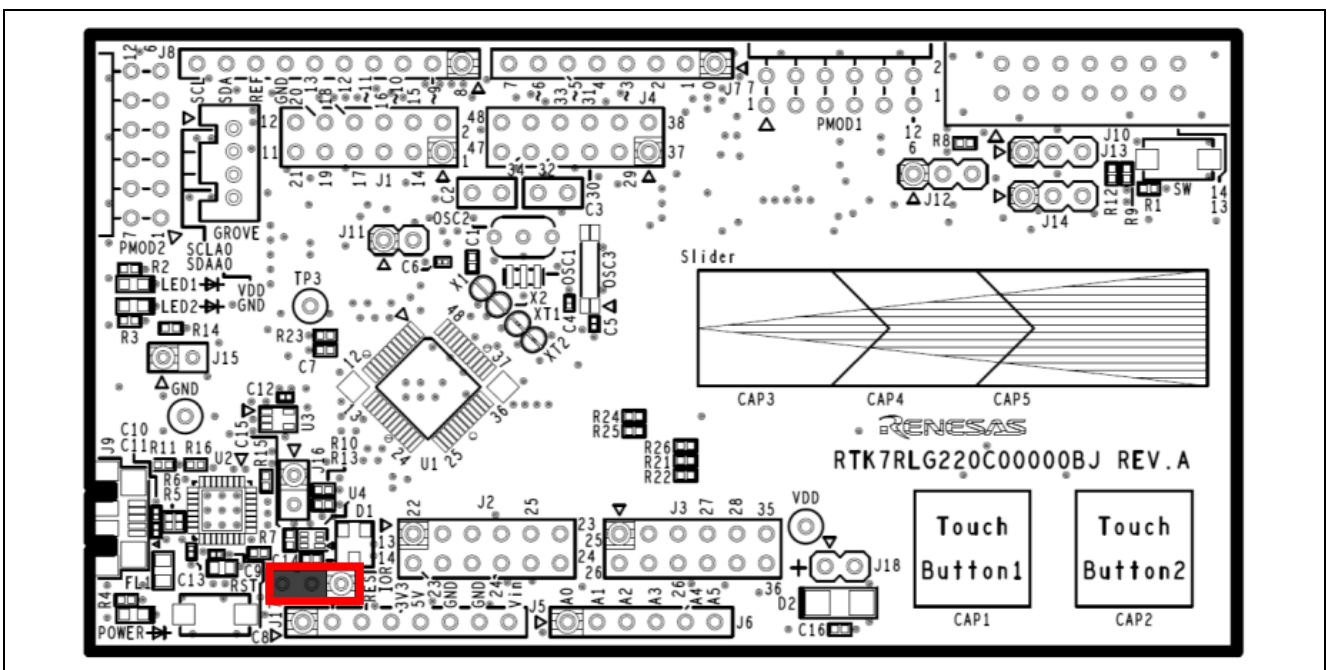


図 6-7 電源選択ヘッダ(J17)の位置

(2) CK-RX65N との UART 通信用ケーブル接続

CK-RX65N と UART 通信を行うための TXD, RXD, GND は RL78/G22 FPB の J7, J8 コネクタの以下の端子に割り当てています。6.1.2(1)章に示す CK-RX65N 側の端子と表 6-1 のように UART 信号の対応を取って接続してください。

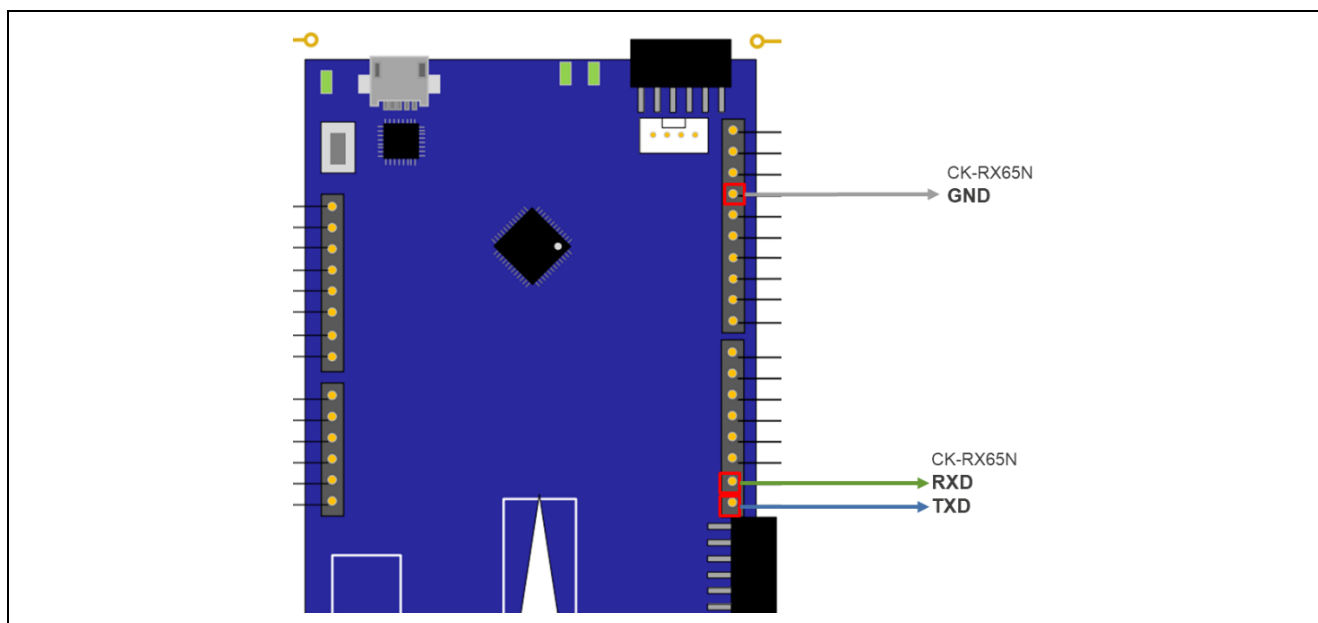


図 6-8 RL78/G22 FPB のマイコン間 UART 通信に使用する端子位置

(3) PC へのログ出力・電源供給用ケーブル接続

PC と RL78/G22 FPB の micro USB Type-B コネクタを USB ケーブルで接続します。

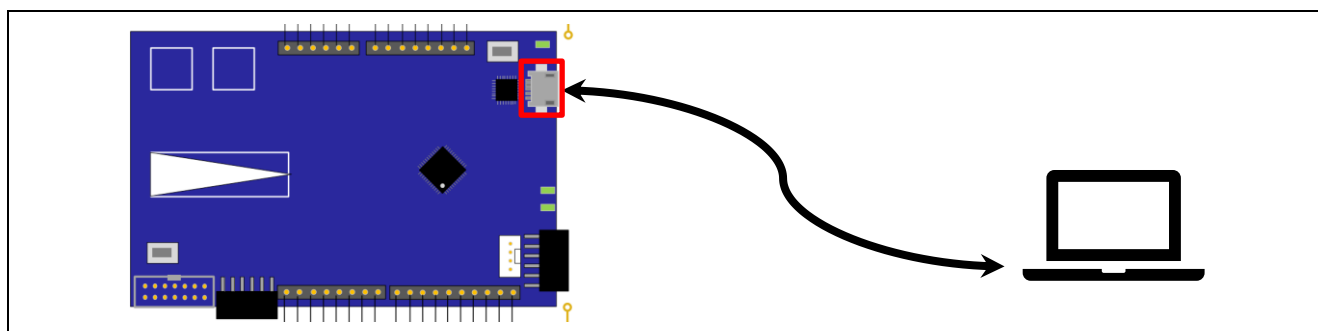


図 6-9 PC と RL78/G22 FPB の接続

(4) HS3001 ボード、FS3000 ボードの接続

RL78/G22 FPB の Pmod2 コネクタに HS3001 センサボード(以下、HS3001 ボードと略します)と FS3000 センサボード(以下、FS3000 ボードと略します)をデジチェーンで接続します。センサボードの重ね順番はどちらでも問題ありません。

I2C バス信号をプルアップ処理させるため、HS3001 ボード上のジャンパ端子(J4, J5)を 2 つともショートさせてください。FS3000 ボードについても同様に J4, J5 のジャンパ端子を 2 つともショートさせてください。この状態で正しく動作しない場合は、どちらかのボードのジャンパ端子をオープンにして抵抗を調整してください。

RL78/G22 FPB の Pmod2 コネクタは Pmod Interface Type 2A/3A であり、Type 6A モジュールである HS3001 ボードと FS3000 ボードは直接接続できないため、図 6-10 のように配線する必要があります。

もしくは、RL78/G22 FPB には Pmod2 コネクタを Type-6A として使用可能にするカットパターンが用意されているため、こちらの方法も利用可能です。詳細は [RL78/G22 FPB のユーザーズマニュアル](#) の「Pmod™コネクタ」の章をご参照ください。

注意 US082-INTERPEVZ 変換ボードは RL78/G22 FPB には使用できません。

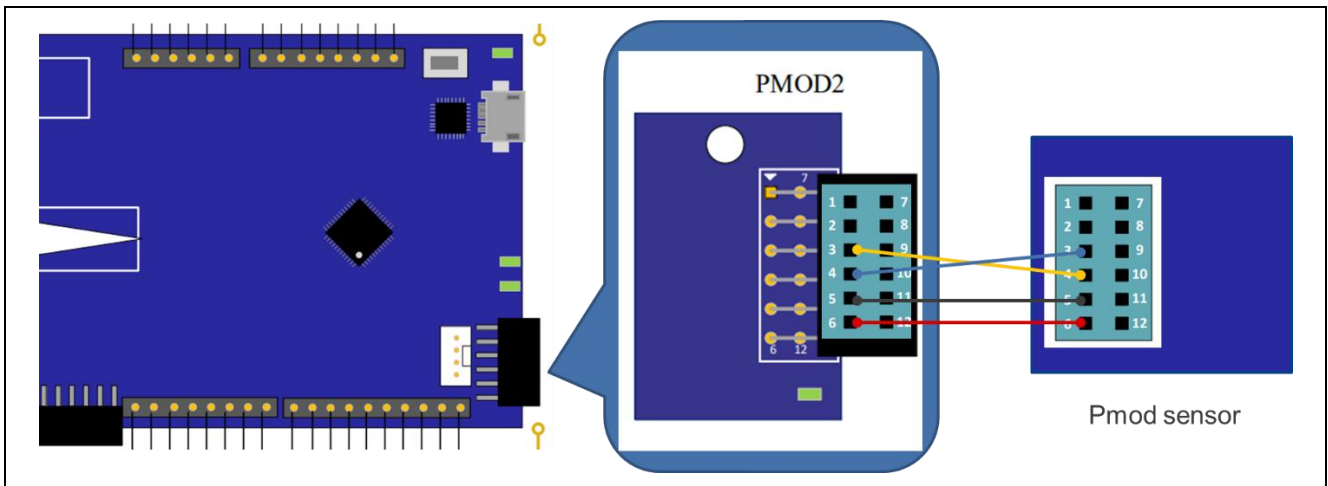


図 6-10 RL78/G22 FPB の Pmod2 コネクタとセンサボードの接続

(5) USB-シリアル変換器リセットヘッダ(J15)を開放する

Micro USB を COM ポートとして利用するために、USB-シリアル変換器リセットヘッダ(J15)を開放します。

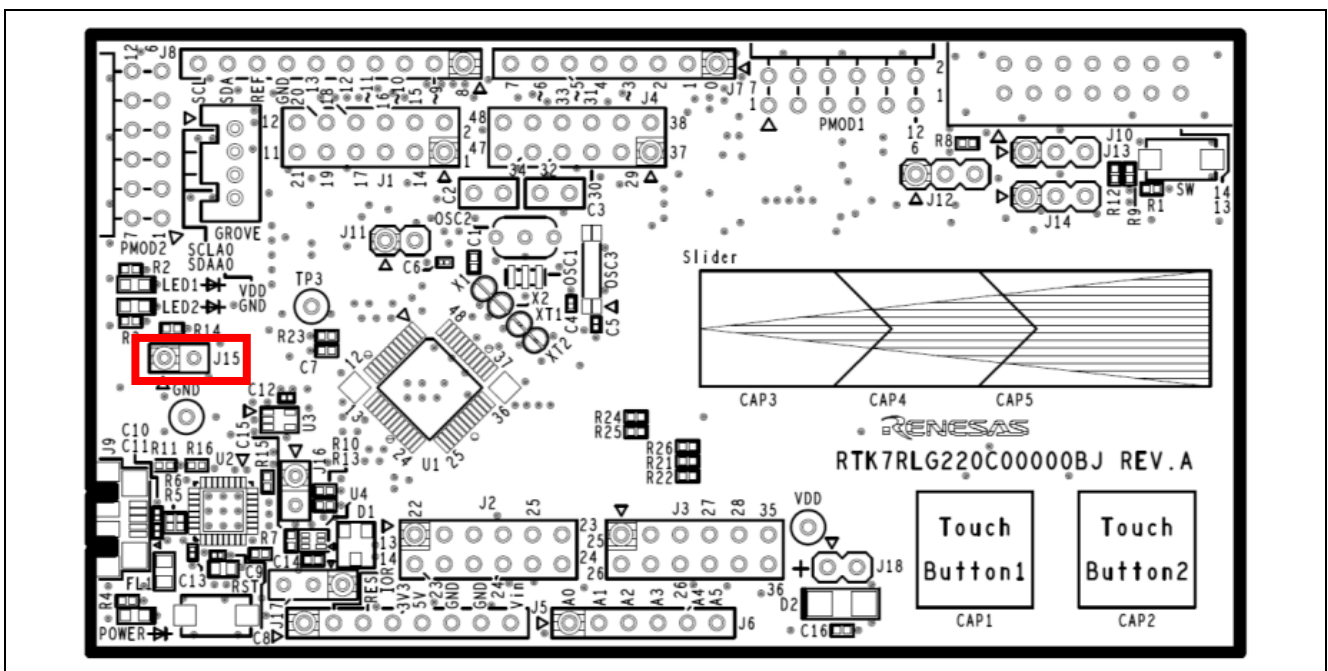


図 6-11 USB-シリアル変換器リセットヘッダ(J15)の位置

以上でデモを実施するためのハードウェアのセットアップは完了です。図 6-12 にデモ構成の全体画像を示します。

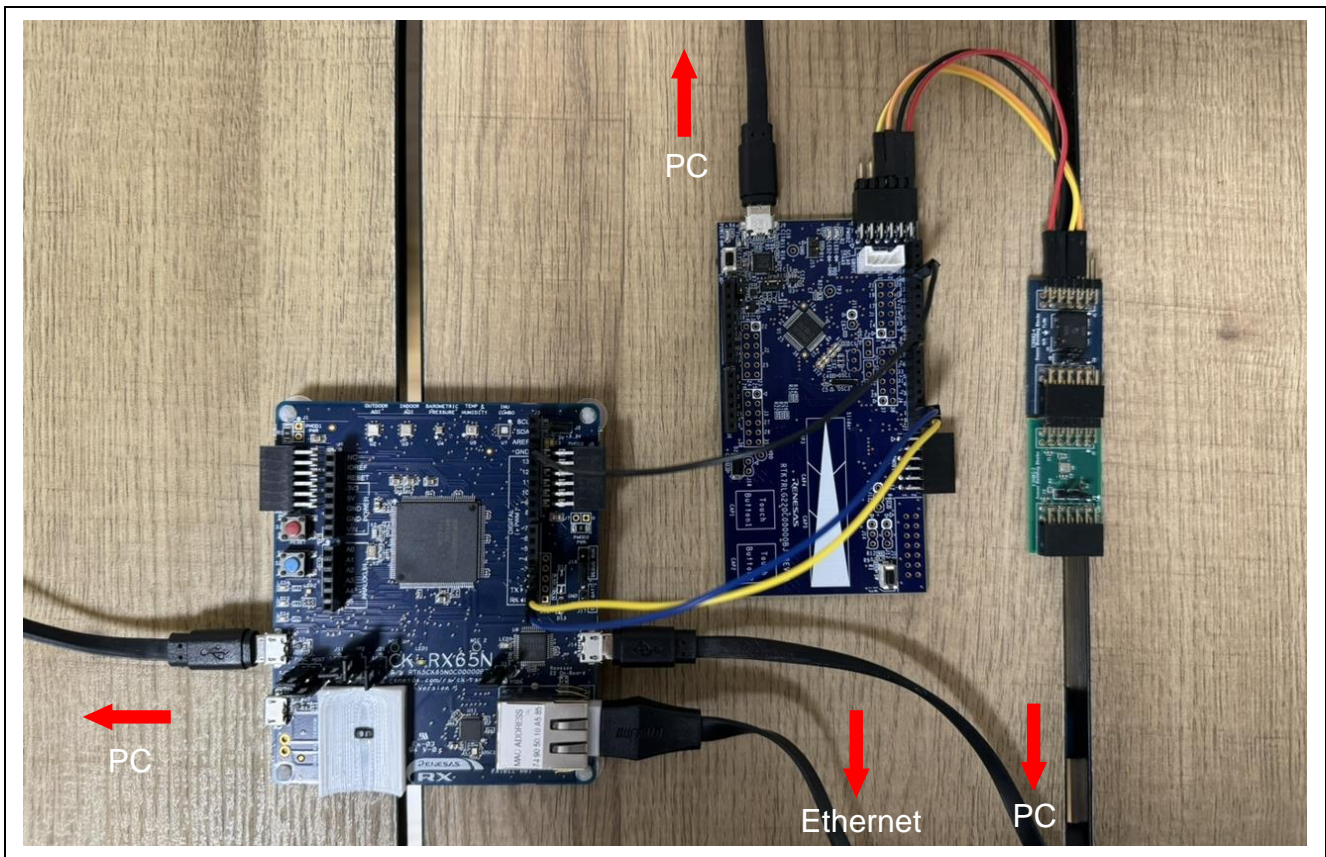


図 6-12 デモ構成の全体画像

6.2 ソフトウェアのセットアップ

6.2.1 事前準備

デモで使用するソフトウェアのインストールを行います。動作確認済みのそれぞれのソフトウェアのバージョンは「2. 動作確認条件」章をご参照ください。

(1) QE for OTA のインストール

e²studio のメニューバーから、[Renesas Views] → [Renesas QE]を開き、QE for OTA がインストールされているか確認します。OTA Main (QE), OTA Manage IoT Device (QE)があればインストール済みです。

無い場合は「[RX ファミリ AWS/Azure を利用したファームウェア更新ソフトの開発ガイド QE for OTA](#)」の「2.1 QE for OTA のインストール」章を参照し、QE for OTA をインストールしてください。

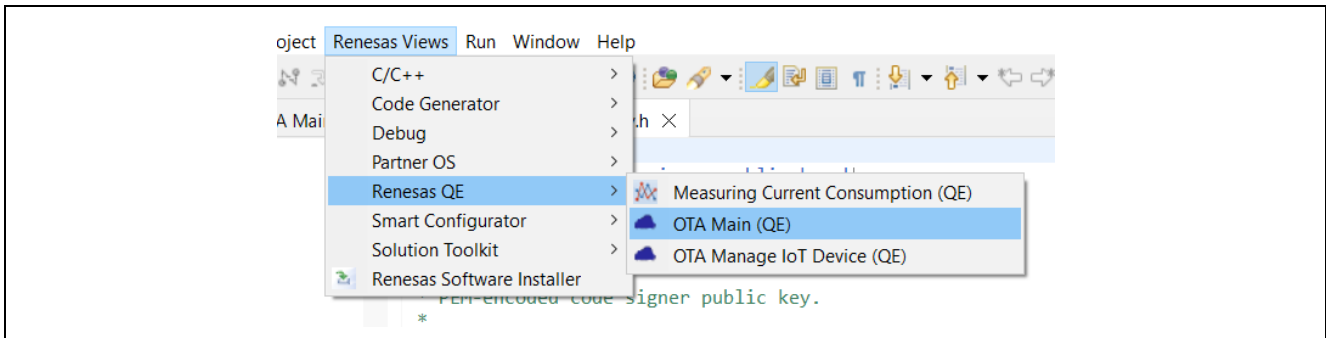


図 6-13 QE for OTA がインストールされているか確認

(2) Python 実行環境のインストール

Python は、<https://www.python.org/> から入手できます。

また、Python の pycryptodome ライブラリを使用します。Python をインストール後、以下の pip コマンドを実行し、インストールしてください。

```
> pip install pycryptodome
```

(3) Renesas Flash Programmer のインストール

Renesas Flash Programmer は、[Renesas Flash Programmer \(Programming GUI\) | Renesas](#) から入手できます。

6.2.2 ターミナルソフトの設定

シリアル通信を使ったログ出力を利用時にターミナルソフト（例：[Tera Term](#)等）が必要です。以下にシリアルポートの設定を示します。

表 6-2 シリアルポート設定

項目	設定
ボーレート	115,200 bps
データ	8-bit
パリティ	なし
ストップ	1-bit
フロー制御	なし

6.2.3 CK-RX65N 用の初期ファームウェアの作成と実行

QE for OTA を使って CK-RX65N 用の初期ファームウェアを作成し、e2 studio でデバッグ実行します。以下に手順を示します。

(1) プロジェクトのインポート

CK-RX65N 用のブートローダである **ck_rx65n_demo_bootloader** プロジェクトと、ユーザプログラムである **ck_rx65n_2ndota_demo** プロジェクトを e2 studio にインポートします。

インポート時は、オプションの「Copy projects into workspace」のチェックを外してください。

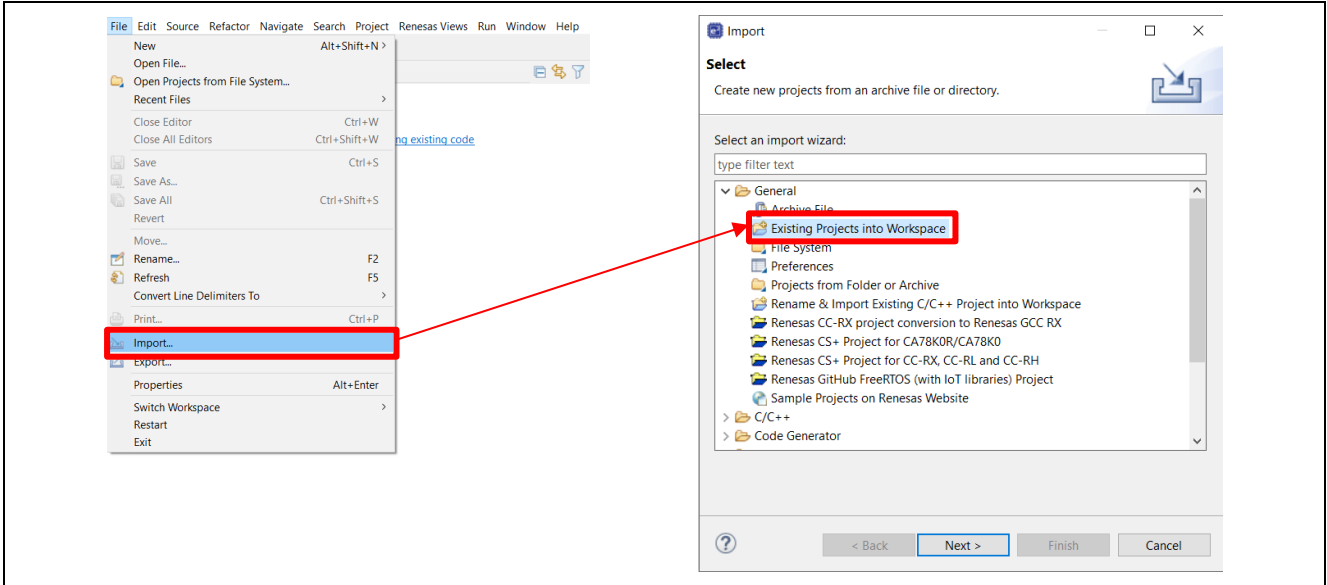


図 6-14 プロジェクトのインポート①

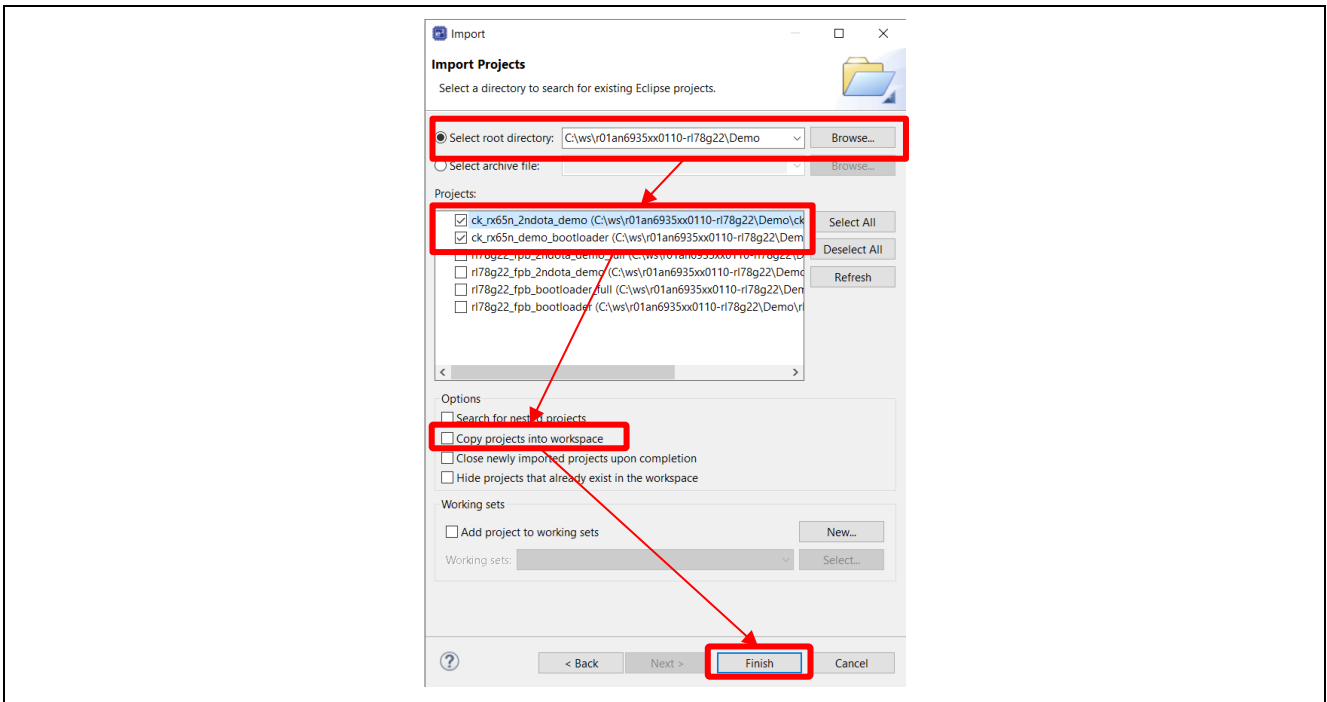


図 6-15 プロジェクトのインポート②

(2) QE for OTA 画面を開く

e2 studio のメニューバーから、[Renesas Views] → [Renesas QE] → [OTA Main (QE)]を選択します。

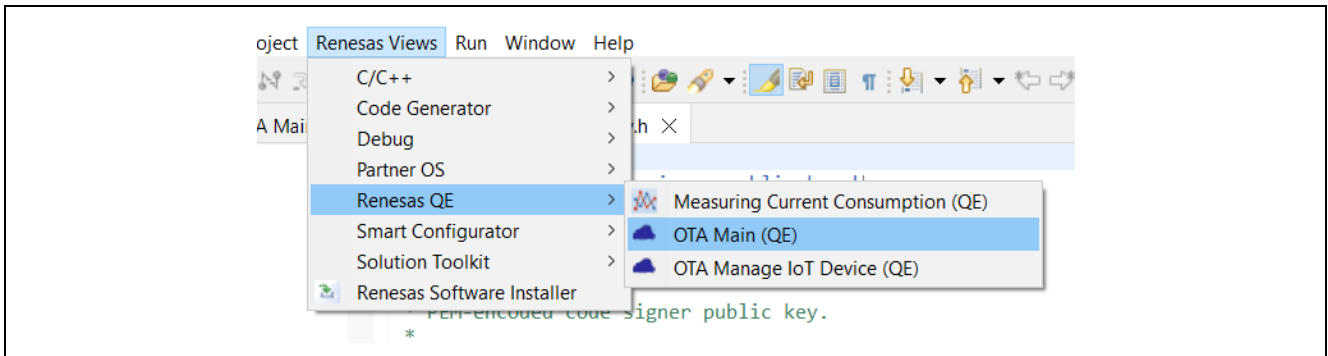


図 6-16 QE for OTA 画面を開く

(3) [QE for OTA] 1. Cloud Settings – Sign-in to Cloud

以降は QE for OTA の GUI 画面に表示された手順通りに進めます。

まず、Cloud に「AWS」を選択しサインインします。ログイン時に選択したリージョンに AWS のリソースが生成されます。

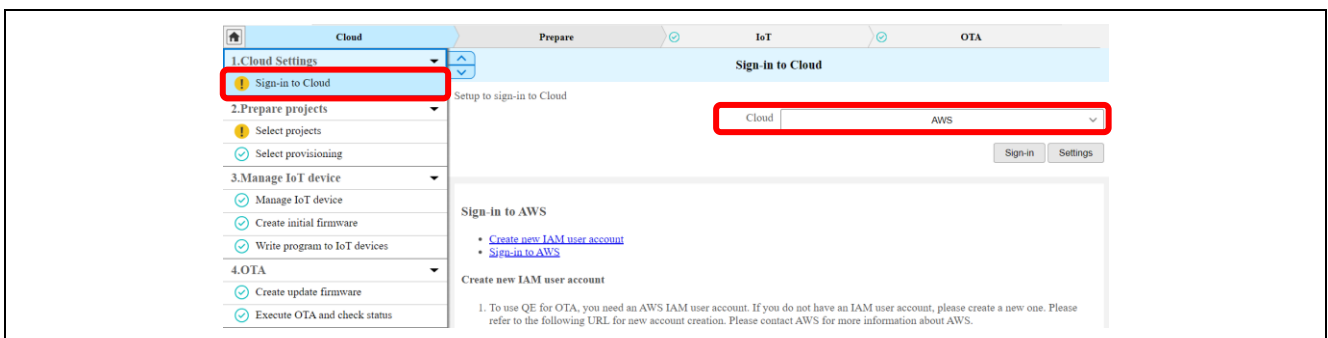


図 6-17 QE for OTA で AWS にサインイン

(4) [QE for OTA] 2. Prepare projects – Select projects

先ほど e2 studio にインポートした ck_rx65n_demo_bootloader プロジェクトと ck_rx65n_2ndota_demo プロジェクトを選択します。

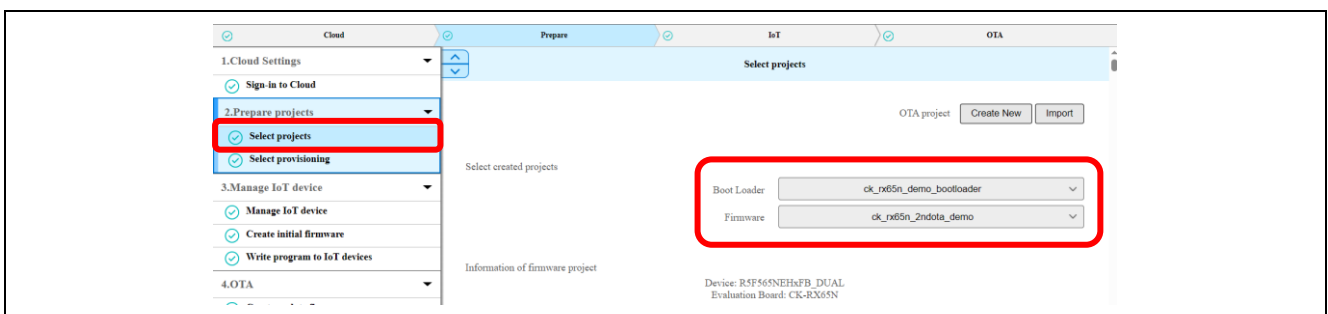


図 6-18 プロジェクトの選択

(5) [QE for OTA] 2. Prepare projects – Select provisioning

プロビジョニング方法として、「Source code includes credentials (asymmetric keys)」を選択します。

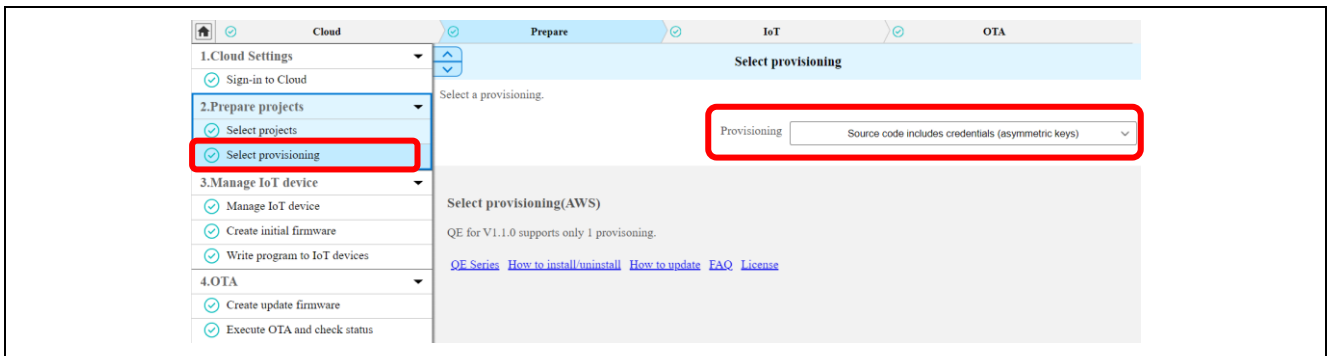


図 6-19 プロビジョニング方法の選択

(6) [QE for OTA] 3. Manage IoT device – Manage IoT device

QE for OTA の手順に従い IoT デバイスを作成します。

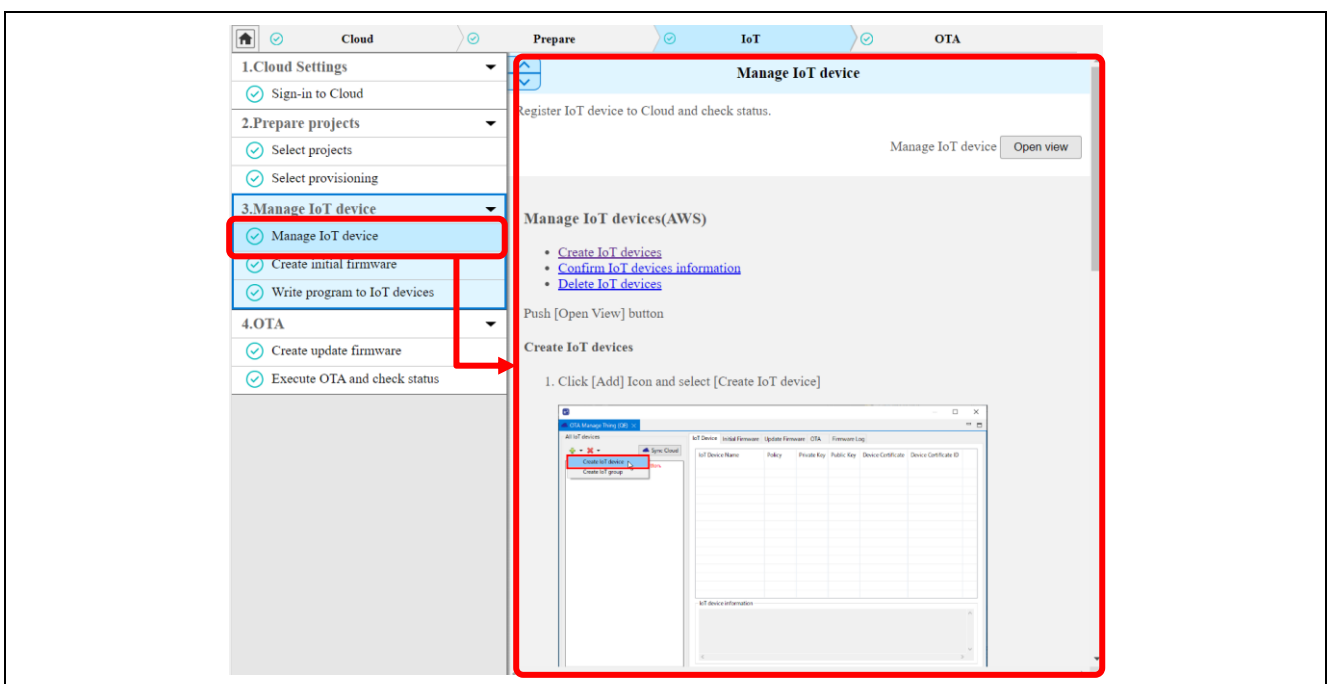


図 6-20 IoT デバイスの作成

(7) [QE for OTA] 3. Manage IoT device – Create initial firmware

QE for OTA の手順に従い初期ファームウェアを作成します。

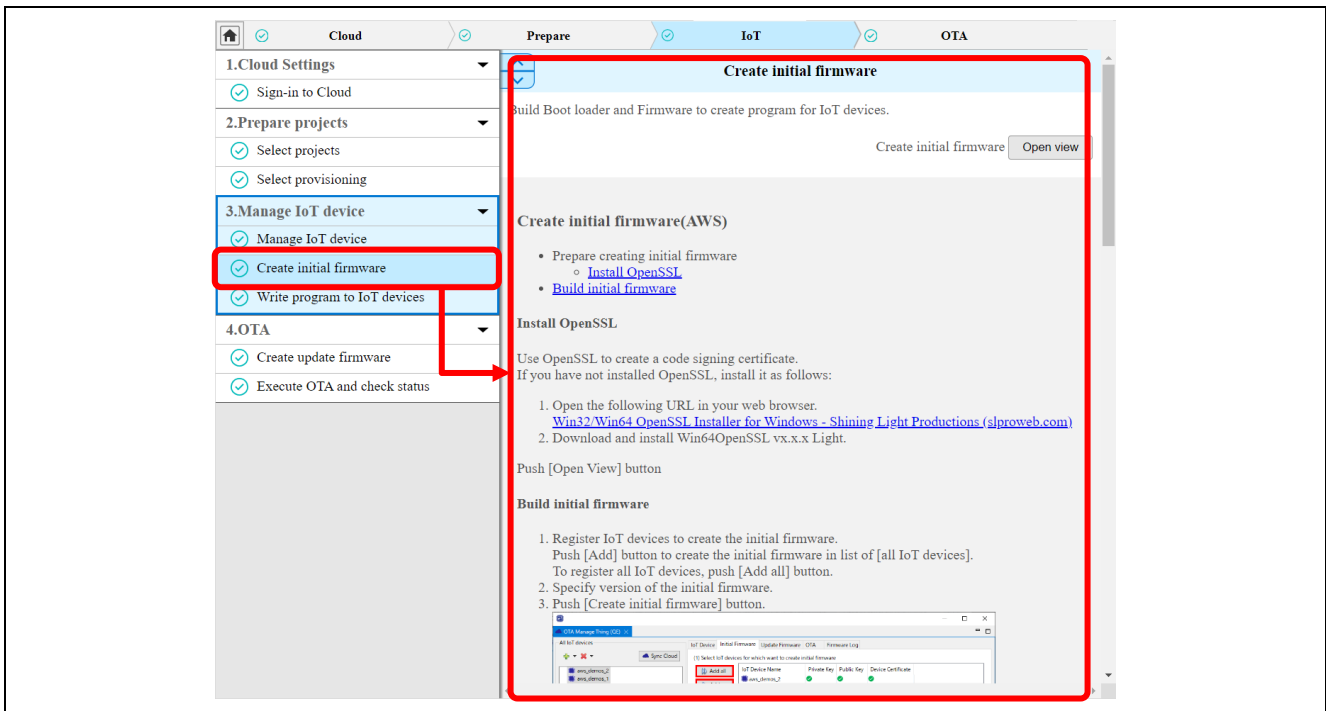


図 6-21 初期ファームウェアの作成

(8) [QE for OTA] 3. Manage IoT device – Write program to IoT devices

QE for OTA の手順に従い CK-RX65N に初期ファームウェアを書き込みます。

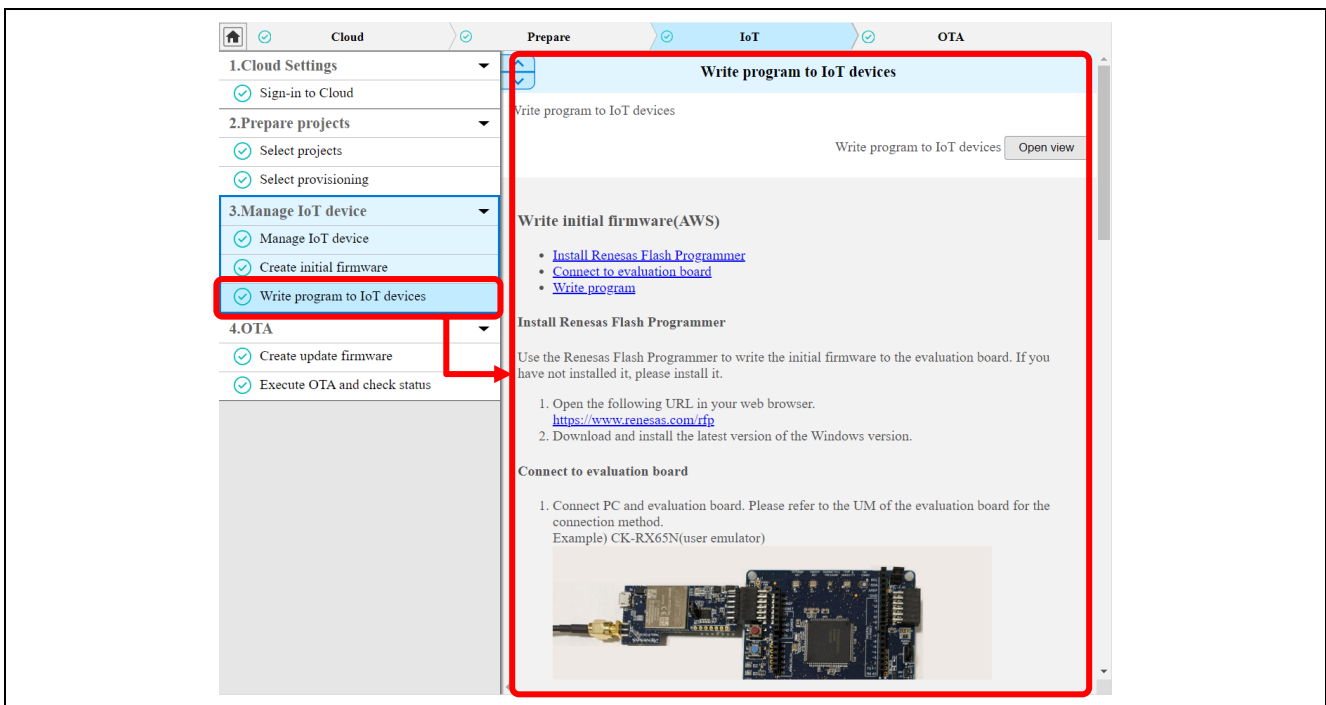


図 6-22 初期ファームウェアの書き込み

(9) 動作確認

図 6-6 のジャンパ J16 を RUN 側(pin 2-3)に短絡します。

ターミナルソフトを起動し、図 6-23 のようにログが出力されていれば CK-RX65N の準備は完了です。

ターミナルソフトで COM ポートの接続に失敗する場合は、QE for OTA の「Firmware Log」ビューを開き、QE for OTA が COM ポートに接続していないか確認してください。

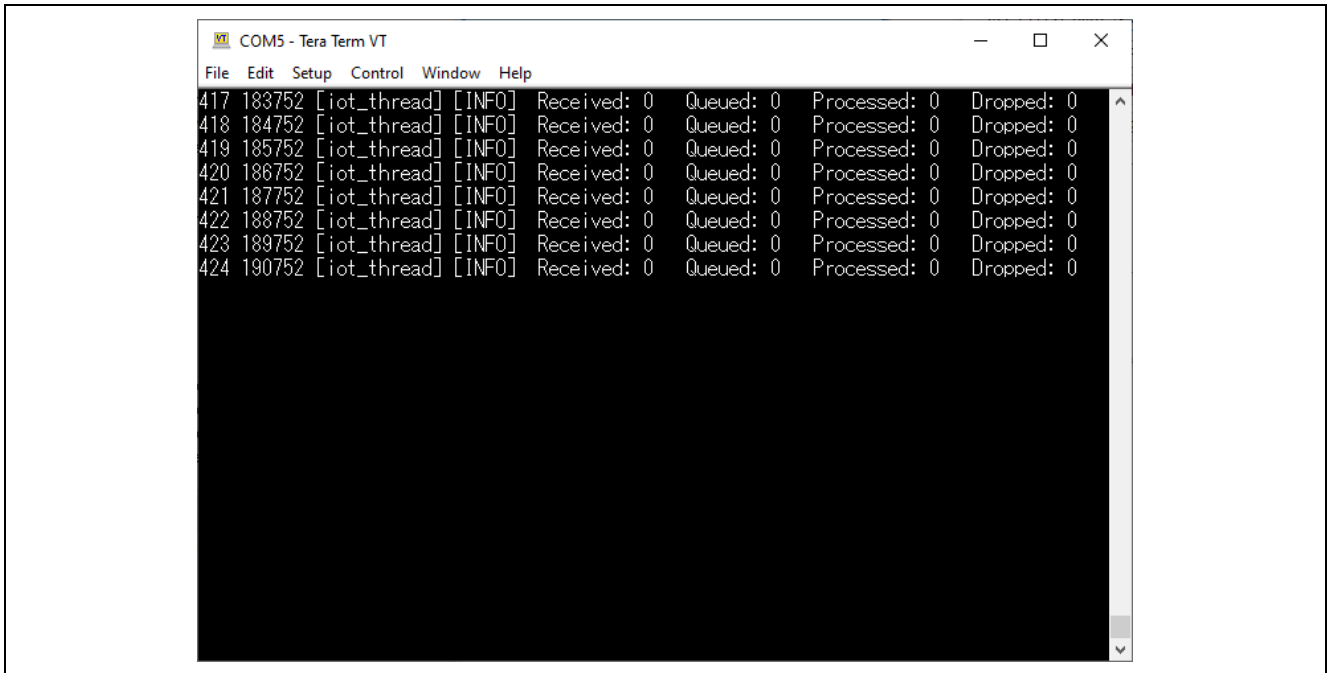


図 6-23 CK-RX65N のログ画面

6.2.4 RL78/G22 FPB 用の初期ファームウェアの作成と実行

RL78/G22 FPB 用の初期ファームウェアを作成し、Renesas Flash Programmer でマイコンに書き込みます。以下に半面更新方式の場合の手順を示します。全面更新方式の場合は、使用するプロジェクトを `rl78g22_fpb_2ndota_demo` プロジェクトから `rl78g22_fpb_2ndota_demo_full` プロジェクトに、`rl78g22_fpb_bootloader` プロジェクトから `rl78g22_fpb_bootloader_full` プロジェクトに置き換えてください。

(1) 事前準備

「[RL78/G22,RL78/G23,RL78/G24 ファームウェア アップデート モジュール Rev.2.01](#)」の 5.2 章「動作環境準備」の 5.2.2 項を実施し、Renesas Image Generator の実行環境構築を行います。

(2) プロジェクトのインポート

RL78/G22 FPB 用のブートローダである `rl78g22_fpb_bootloader` プロジェクトと、ユーザプログラムである `rl78g22_fpb_2ndota_demo` プロジェクトを e2 studio にインポートします。

e2 studio の [File] メニューから [Import] をクリックします。

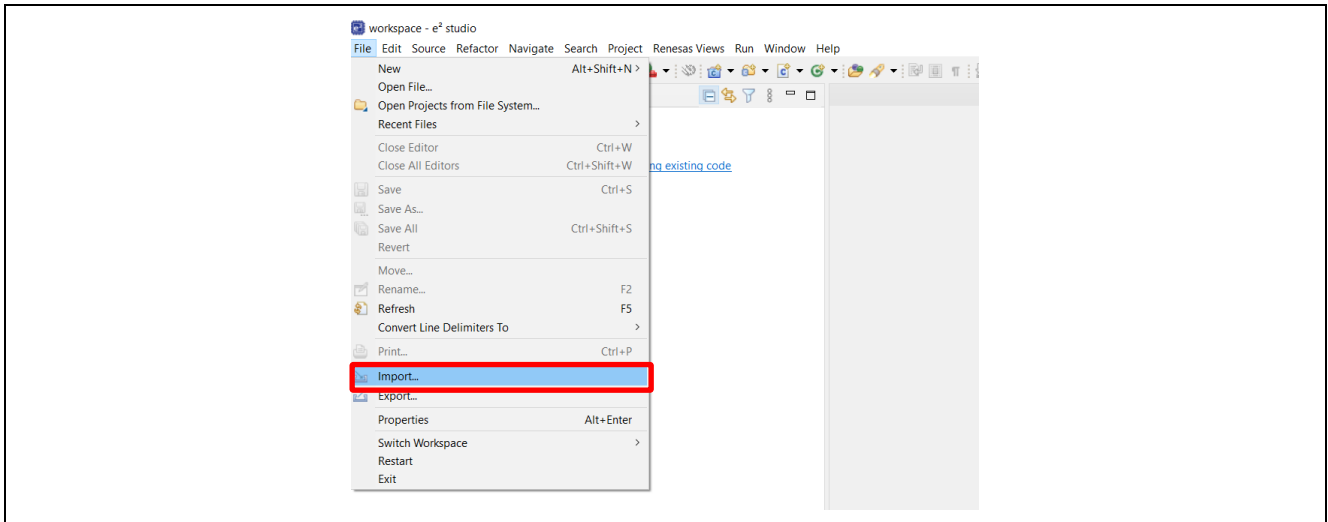


図 6-24 e2 studio プロジェクトのインポート手順①

[Existing Projects into Workspace]を選択し、[Next]をクリックします。

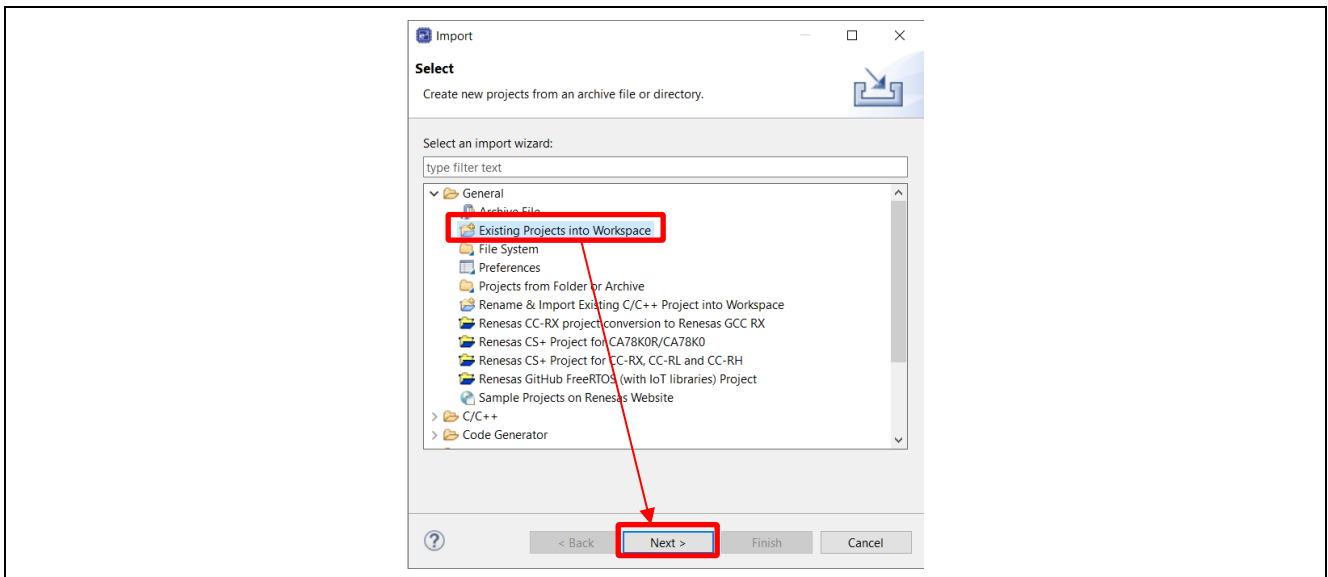


図 6-25 e2 studio プロジェクトのインポート手順②

[Select root directory]で、サンプルプログラムのフォルダを選択し、表示されたプロジェクトから、**rl78g22_fpb_2ndota_demo** と **rl78g22_fpb_bootloader** にチェックを入れ、Option 欄の[Copy projects into workspace]にチェックが入っていないことを確認し、[Finish]をクリックします。

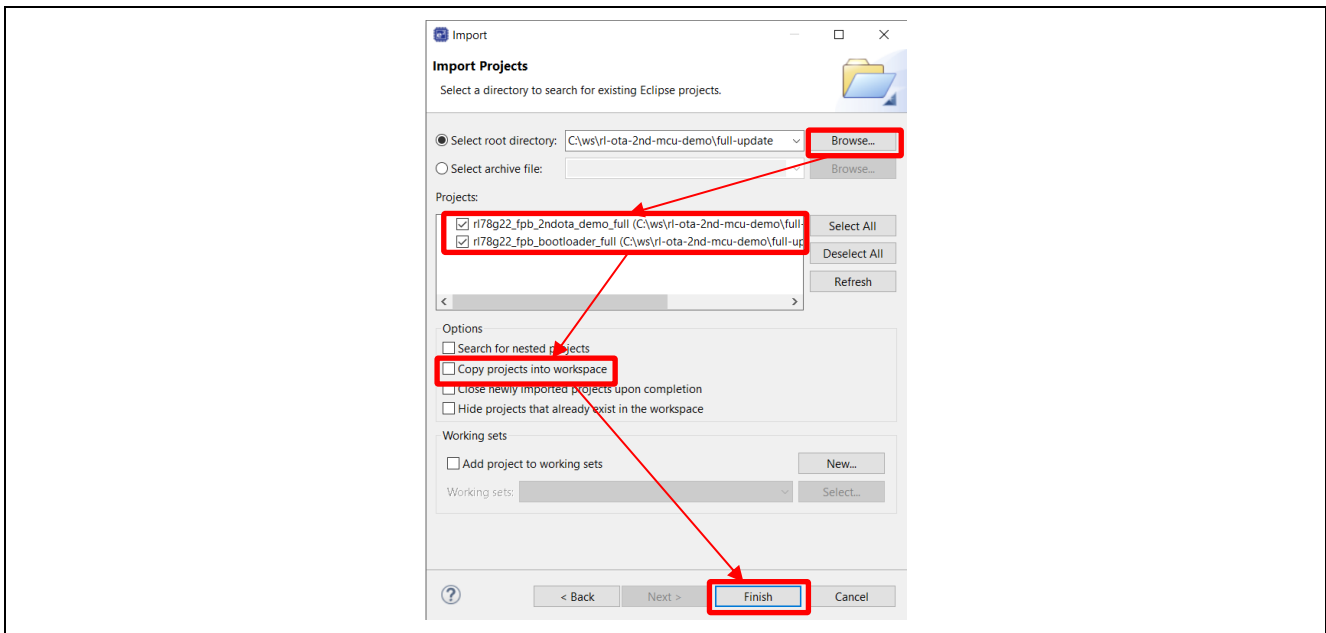


図 6-26 e2 studio プロジェクトのインポート手順③

(3) プロジェクトのビルド

rl78g22_fpb_bootloader プロジェクトと rl78g22_fpb_2ndota_demo プロジェクトをビルドし、MOT ファイルを作成します。MOT ファイルはプロジェクトフォルダ直下の HardwareDebug フォルダに作成されます。

(4) 初期ファームウェアの作成

作成された rl78g22_fpb_bootloader と rl78g22_fpb_2ndota_demo の MOT ファイルを結合して RL78/G22 FPB 用の初期ファームウェアを作成します。MOT ファイルの結合には Renesas Image Generator を使用します。Renesas Image Generator は「[RL78/G22,RL78/G23,RL78/G24 ファームウェア アップデート モジュール Rev.2.01](#)」に付属するツールです。詳細については、上記リンク先のアプリケーションノートの「Renesas Image Generator」章をご参照ください。

r01an6935jj0100-rl78g22/RenesasImageGenerator フォルダ内で以下のコマンドを実行し、初期ファームウェア initial_firm.mot を作成します。RenesasImageGenerator フォルダの howtouse_full/partial.txt ファイルにコマンド文があります。

```
> python image-gen.py -ip .\¥RL78_G22_ImageGenerator_PRM.csv -
ibp ..¥rl78g22_fpb_bootloader¥HardwareDebug¥rl78g22_fpb_bootloader.mot -
iup ..¥rl78g22_fpb_2ndota_demo¥HardwareDebug¥rl78g22_fpb_2ndota_demo.mot -o
initial_firm
```

全面更新方式の場合のコマンドは以下となります。

```
> python image-gen.py -ip .\¥RL78_G22_FullUpdate_ImageGenerator_PRM.csv -
ibp ..¥rl78g22_fpb_bootloader_full¥HardwareDebug¥rl78g22_fpb_bootloader_full.mot -
iup ..¥rl78g22_fpb_2ndota_demo_full¥HardwareDebug¥rl78g22_fpb_2ndota_demo_full.mot -o
initial_firm
```

(5) 初期ファームウェアの書き込み

Renesas Flash Programmer を用いて、上で作成した初期ファームウェア initial_firm.mot を RL78/G22 FPB に書き込みます。

Renesas Flash Programmer を起動し、[File]メニューから[New Project]をクリックします。

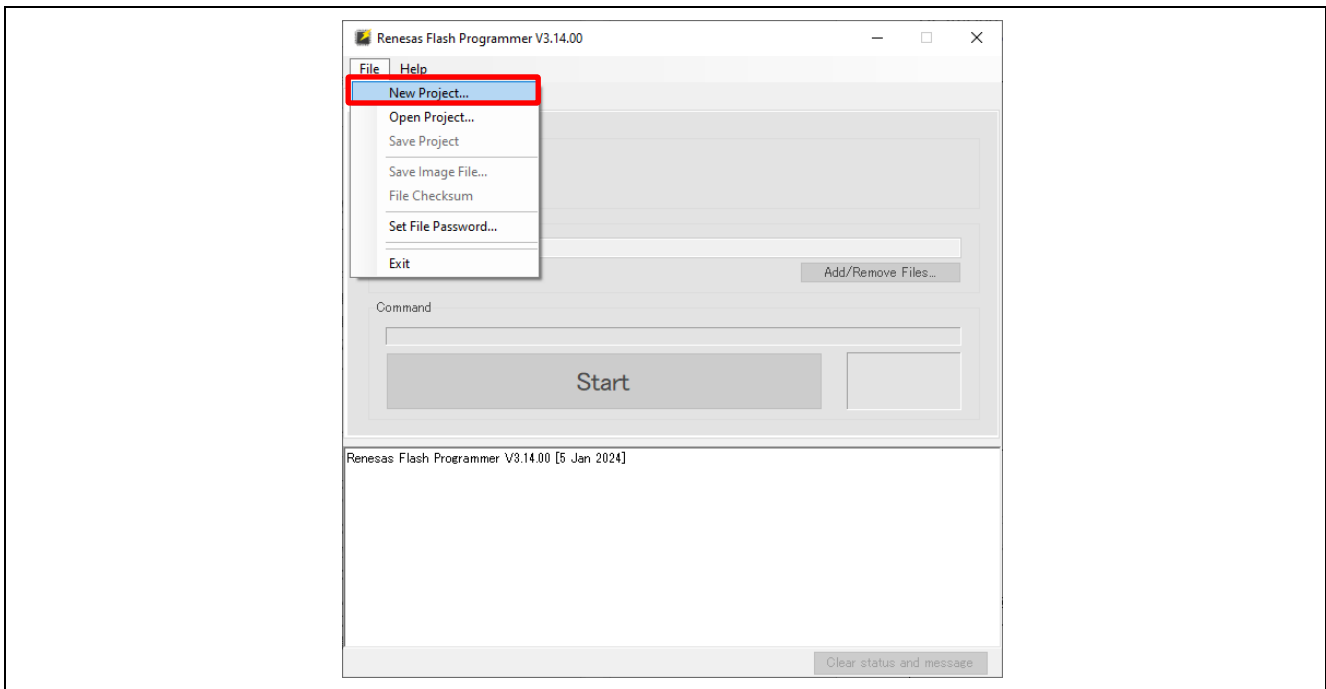


図 6-27 MOT ファイルの書き込み手順①

「Microcontroller」には「RL78/G2x」を選択し、「Project Name」を入力します。

Communication 欄の「Tool」は「COM port」、「Interface」は「2 wire UART」を選択します。

「Tool Details」をクリックし、RL78/G22 FPB の仮想 COM ポートを選択し[OK]をクリックします。

[Connect]をクリックし、「**Operation completed.**」と表示されることを確認します。

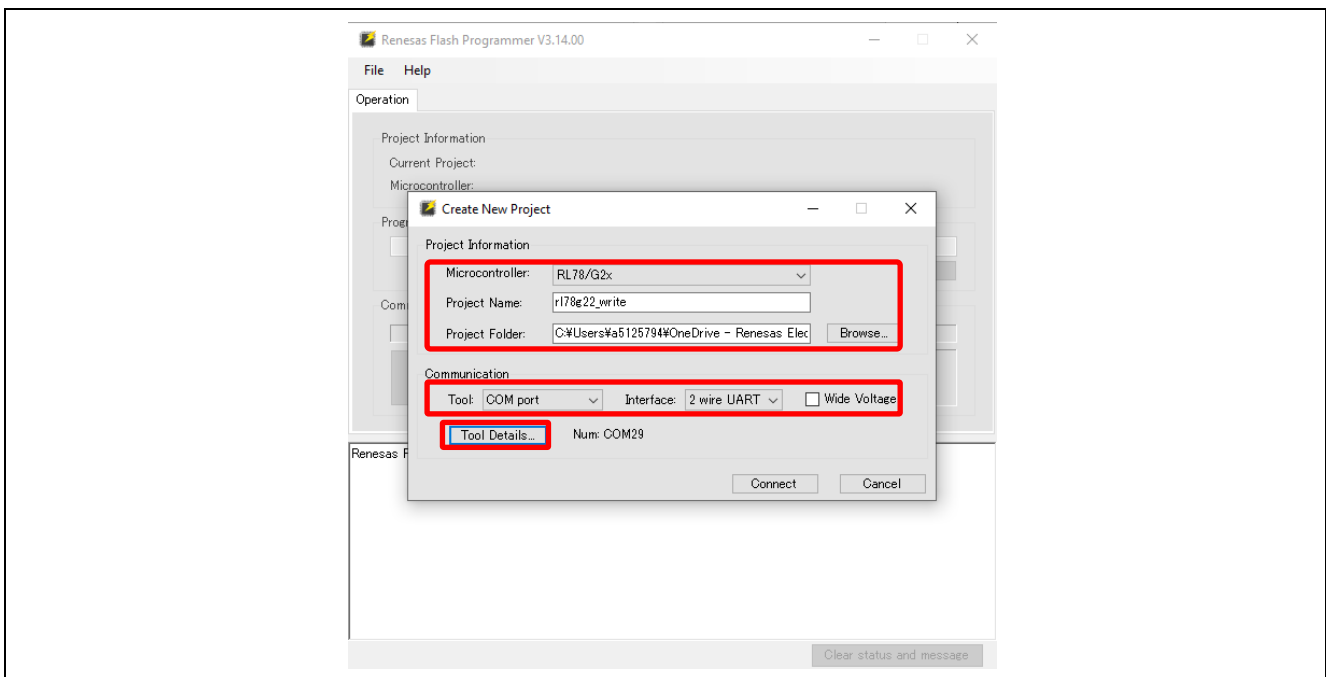


図 6-28 MOT ファイルの書き込み手順②

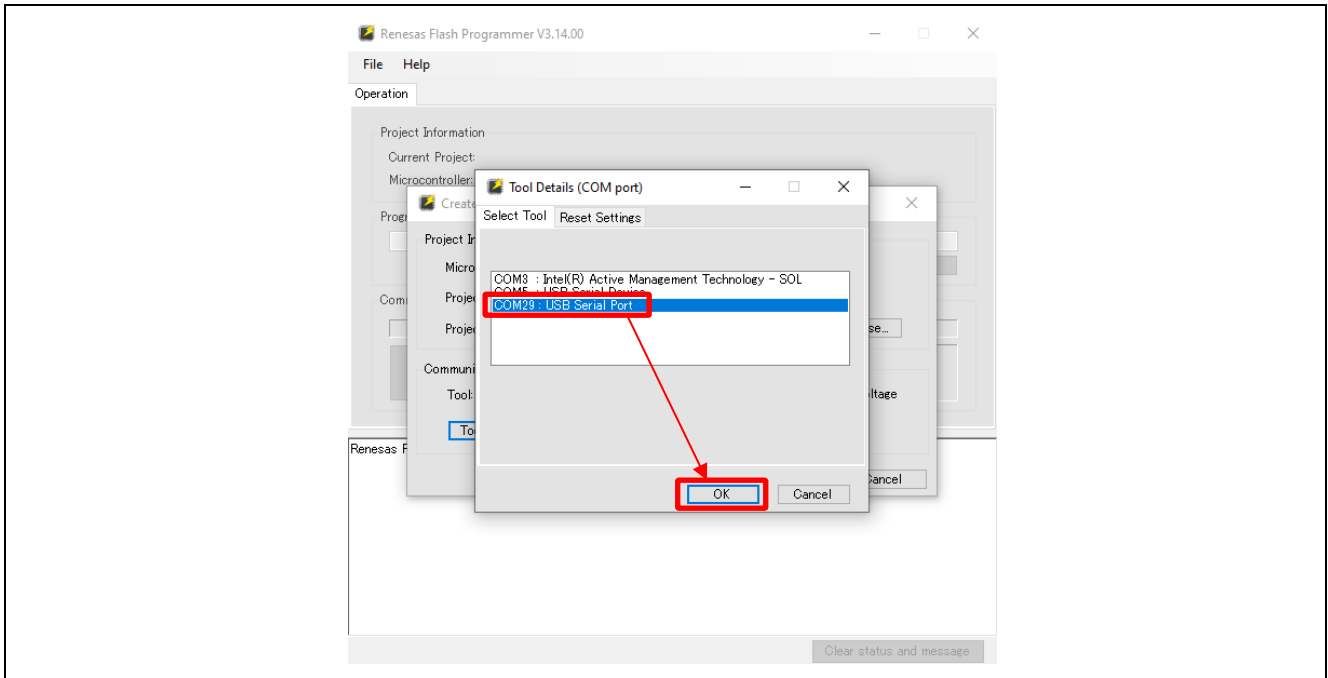


図 6-29 MOT ファイルの書き込み手順③

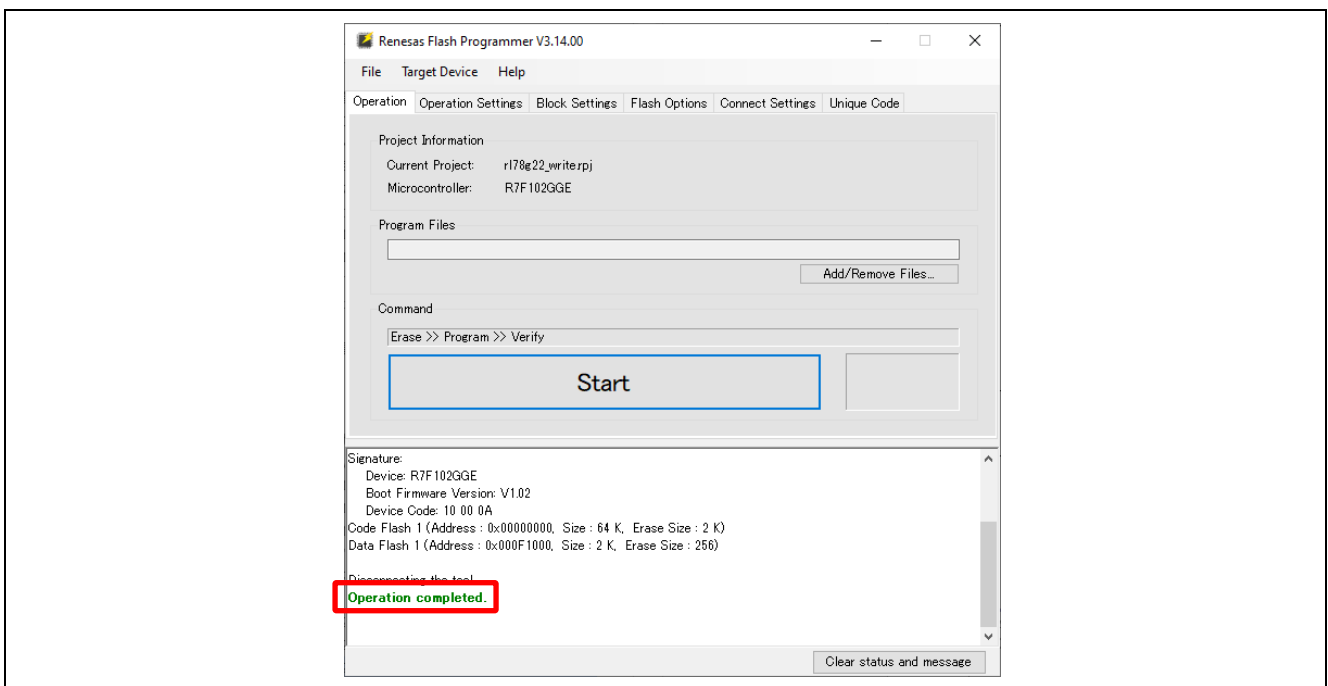


図 6-30 MOT ファイルの書き込み手順④

[Program File]欄の[Add/Remove Files...] ⇒ [Add File(s)...]をクリックし、Renesas Image Generator で作成した **initial_firm.mot** を選択し[OK]をクリックします。

[Start]をクリックすると書き込みが開始されます。

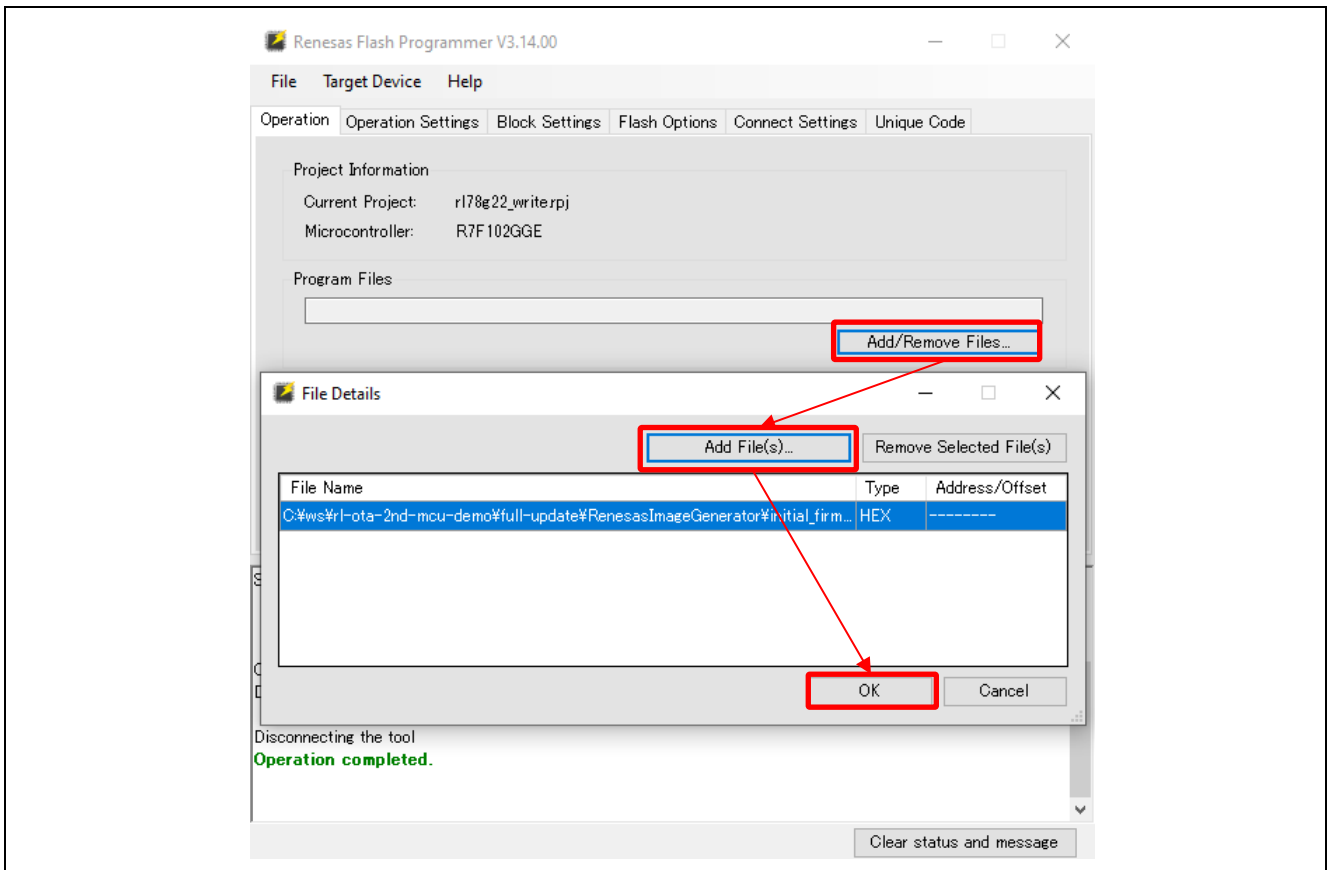


図 6-31 MOT ファイルの書き込み手順⑤

(6) 動作確認

ターミナルソフトを起動し、図 6-32 のように温度と湿度の値がログ出力されていれば RL78/G22 FPB の準備は完了です。

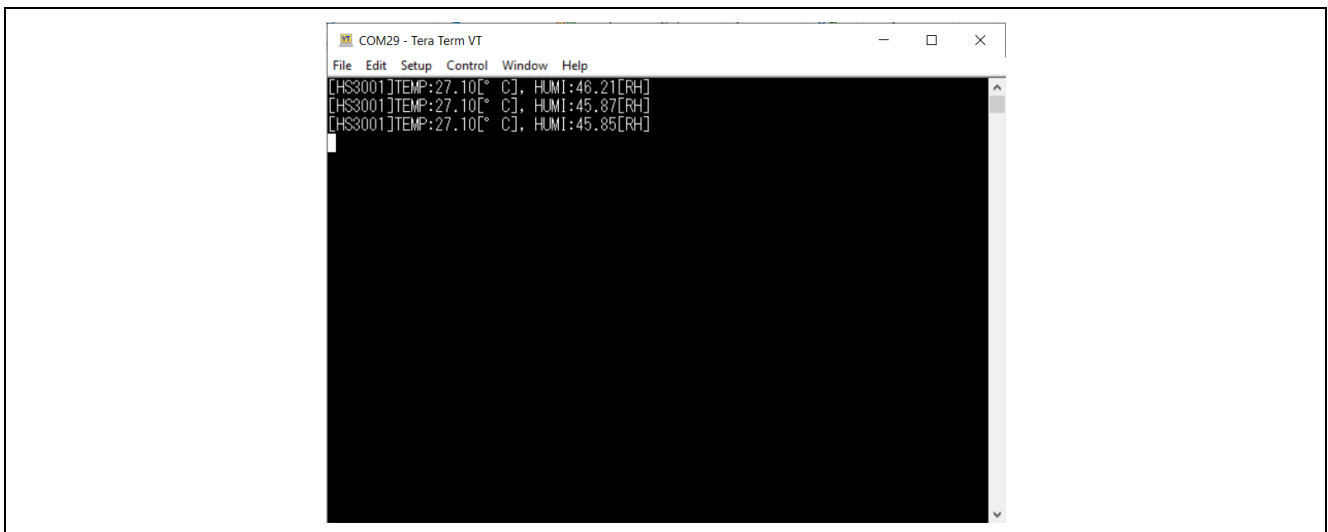


図 6-32 RL78/G22 FPB のログ画面

6.3 AWS クラウドの準備

AWS マネジメントコンソールにログインします。

[AWS マネジメントコンソール | AWS \(amazon.com\)](https://aws.amazon.com/)

マネジメントコンソール画面の右上に表示されているリージョンを確認し、QE for OTA ログイン時の設定と同じリージョンを選択します。

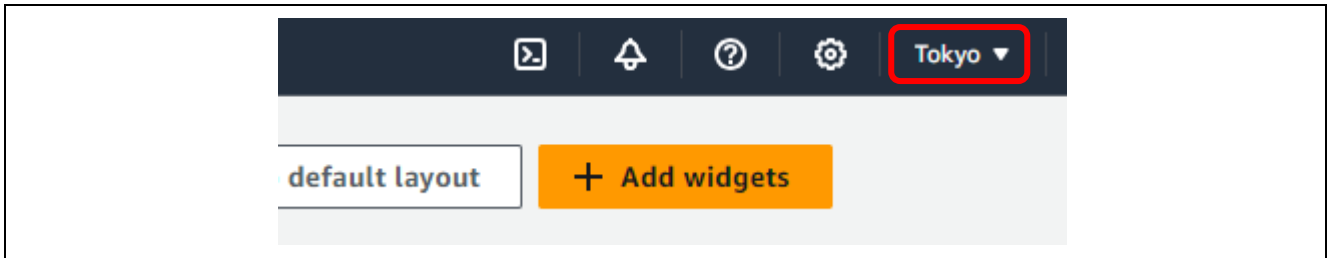


図 6-33 リージョンの確認

6.3.1 OTA アップデートのための設定

「[RX ファミリ RX65N における Amazon Web Services を利用した FreeRTOS OTA の実現方法 \(v202210.01-LTS-rx-1.1.0 以降対応版\)](#)」を参照し、設定を行います。

- (1) 上記アプリケーションノートの「3.4 Amazon S3 バケットの作成」の手順で Amazon S3 バケットを作成します。ここで設定したバケット名はデモ実行時に使用します。
- (2) 上記アプリケーションノートの「3.5 IAM ユーザーに OTA の実行権限を割り当てる」の手順でサービスロールを作成します。ここで設定したサービスロール名はデモ実行時に使用します。
- (3) 上記アプリケーションノートの「5.2 ファームウェアの更新 (5)~(9)」の手順でコード署名証明書を登録します。ここで、登録するコード署名証明書は 6.2.3(7)章で CK-RX65N 用の初期ファームウェアを QE for OTA で作成したときに作成した証明書を使用します。

証明書は ck_rx65n_demo_bootloader/QE-OTA/codesigning に作成されています。証明書は secp256r1.crt, 秘密鍵は secp256r1.privateKey, 証明書チェーンは ca.crt を指定します。

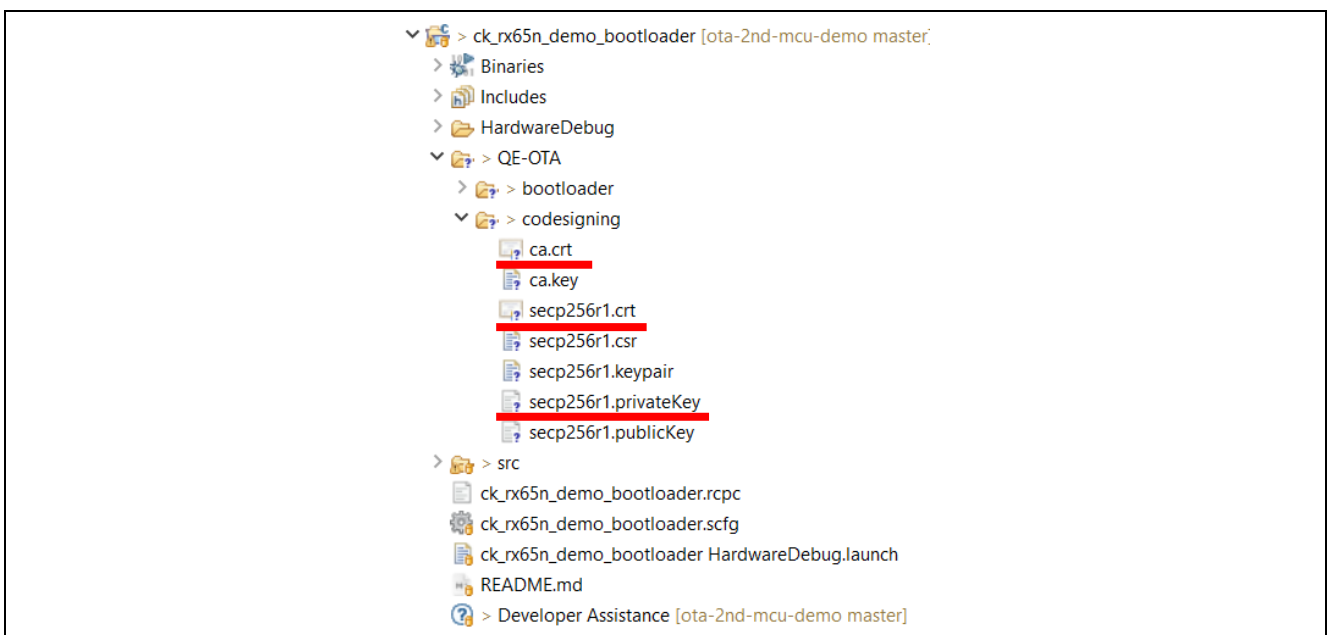


図 6-34 コード署名証明書の作成場所

6.3.2 センサデータ可視化のための設定

受信したセンサデータをグラフ形式で可視化するために、Amazon CloudWatch および AWS IoT Core で以下の手順で設定を行います。

Note グラフ形式での可視化は必要なく、データが AWS に届いていることをブラウザ上で確認できればいいという場合は 6.3.2 章の作業は丸ごと省略可能です。
この場合、図 6-35 のように AWS IoT の「MQTT テストクライアント」で「iotdemo/topic/sensor」をサブスクライブすることでセンサデータが期待通り受信できていることをテキスト形式で確認できます。

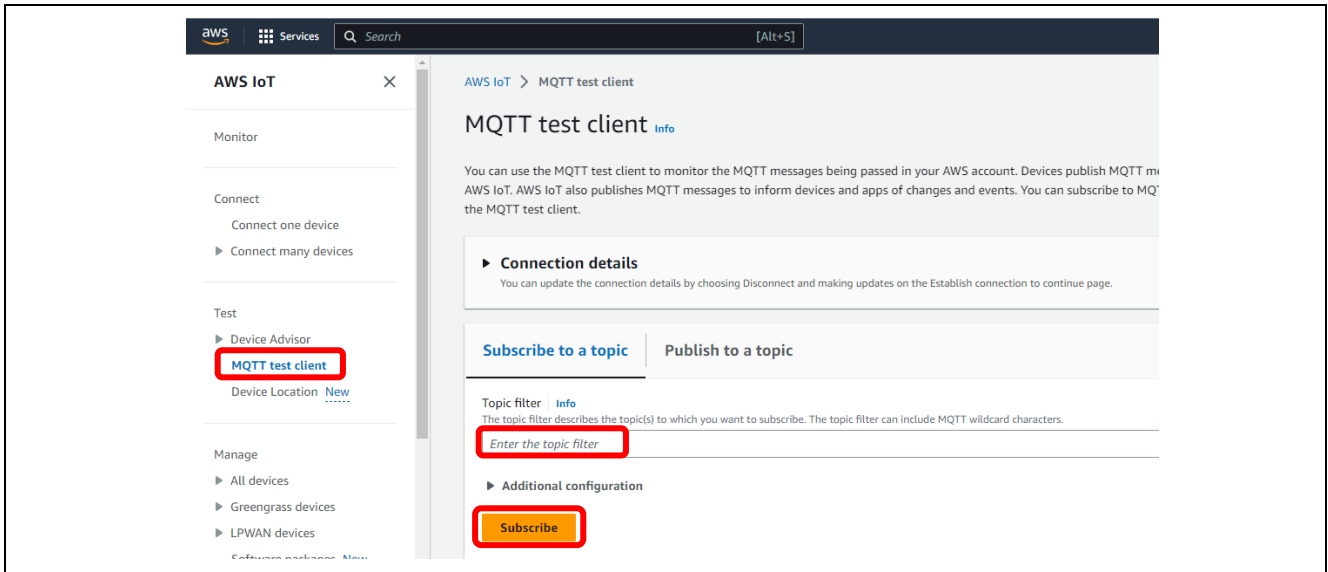
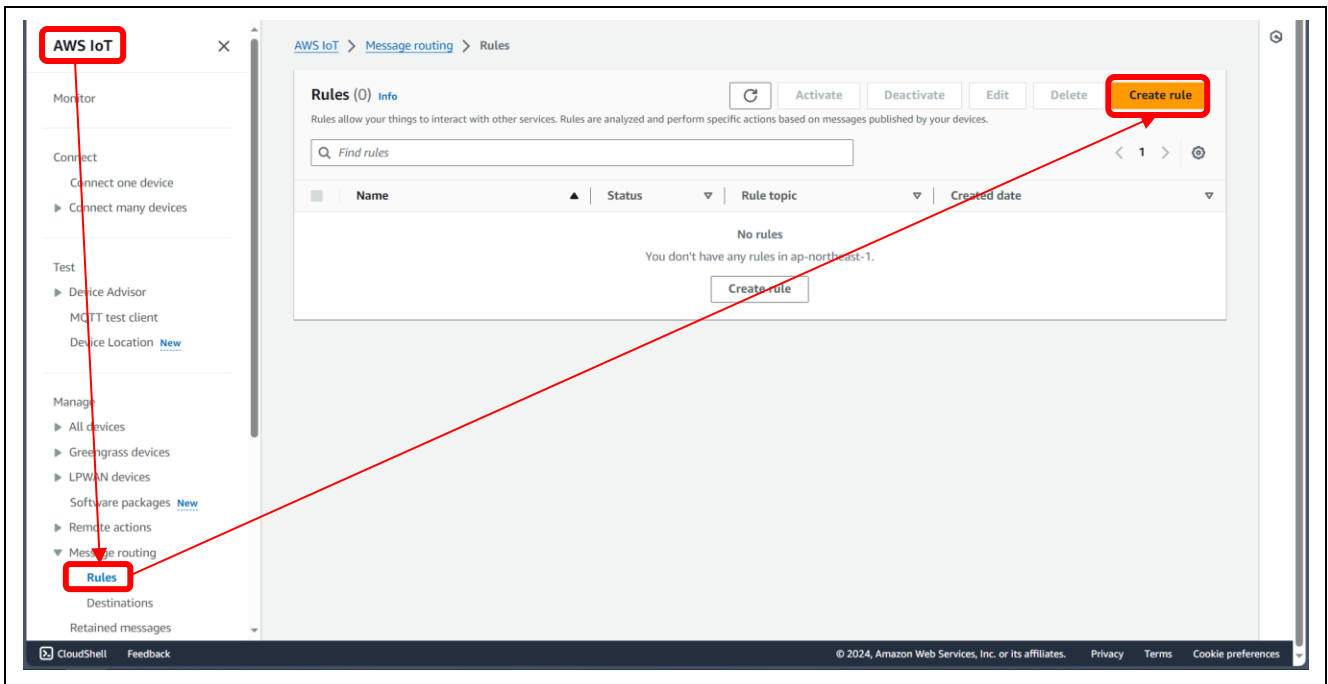


図 6-35 MQTT テストクライアントでのデータ受信確認

6.3.2.1 Amazon CloudWatch の設定

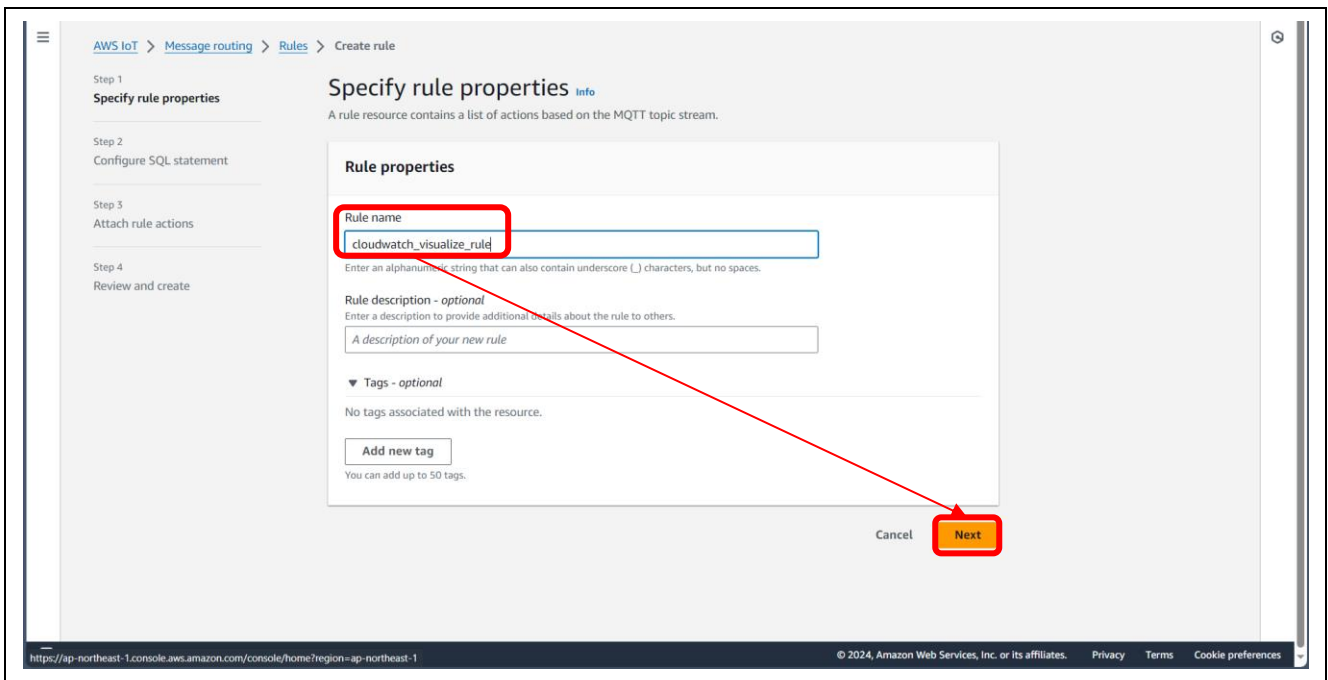
(1) AWS IoT でルールを作成

[AWS IoT] ⇒ [ルール] ⇒ [ルールを作成]をクリックします。



(2) ルールのプロパティを指定

ルール名を入力し、次へをクリックします。

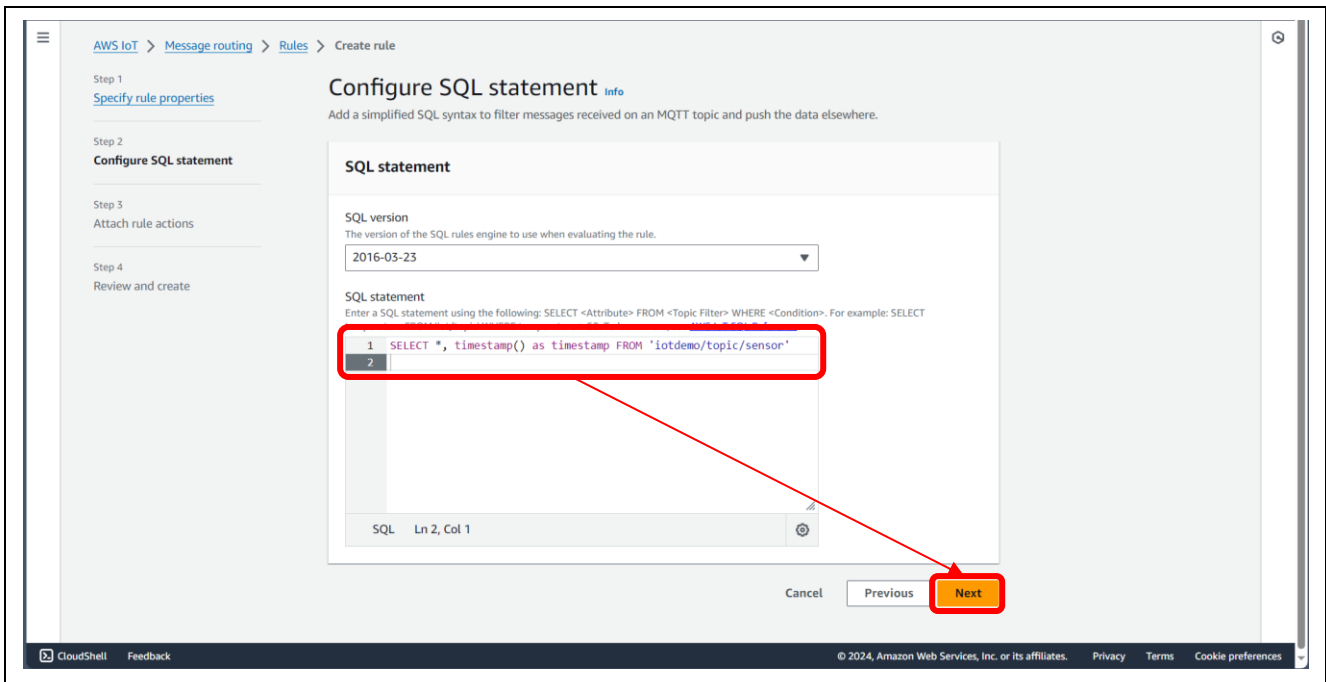


(3) SQL ステートメントを設定

SQL ステートメントを入力します。テキストエディタに以下のように入力します。最終行に改行が必要な点に注意してください。

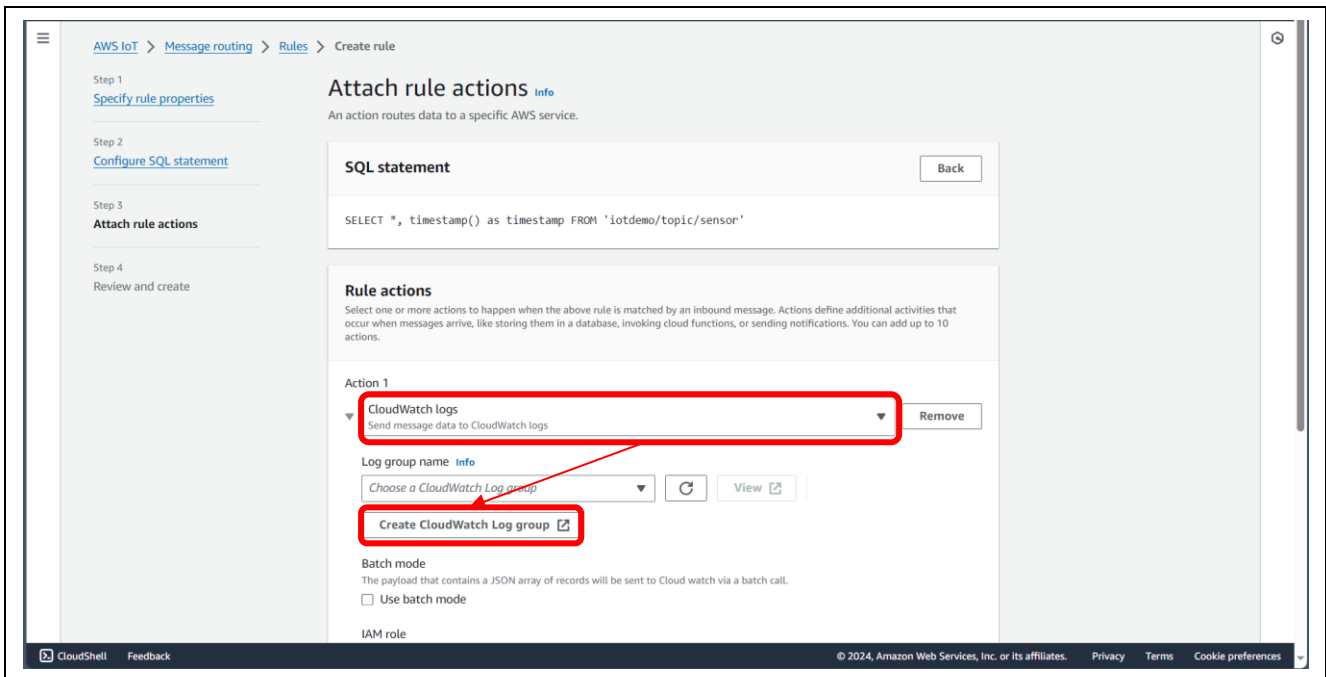
```
SELECT *, timestamp() as timestamp FROM 'iotdemo/topic/sensor'
```

(改行が必要)



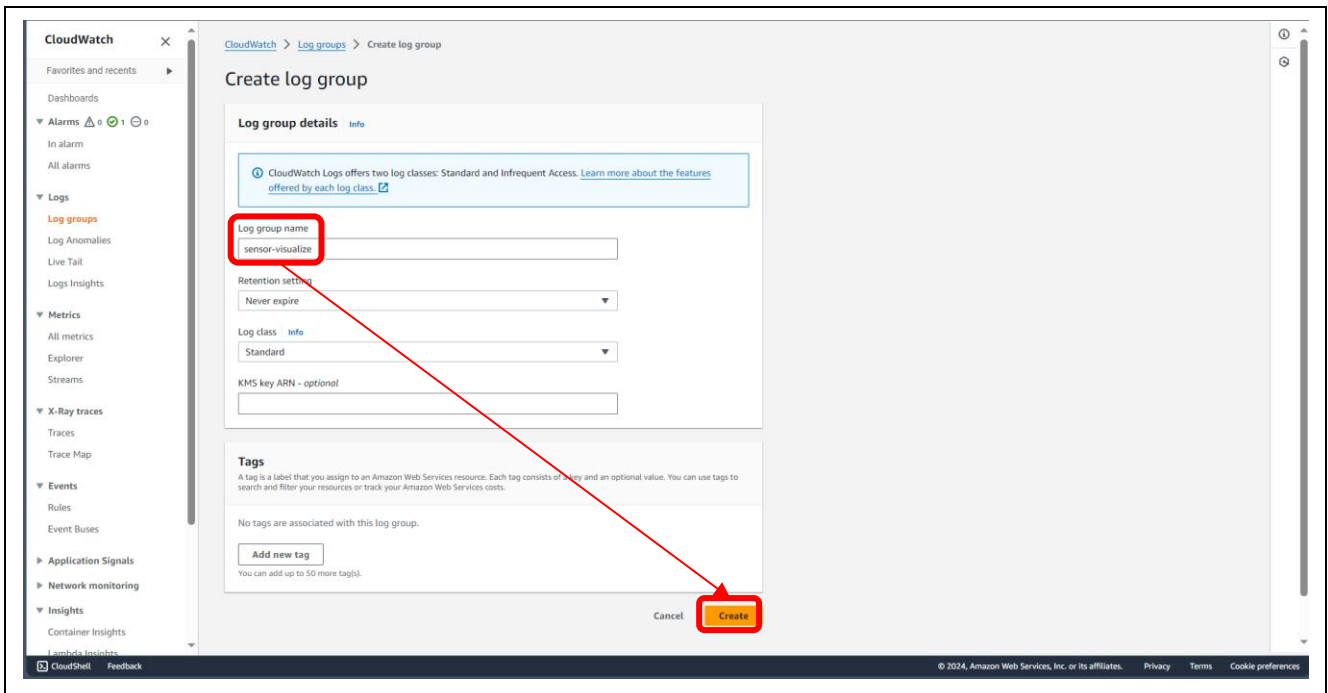
(4) 「ルールアクションをアタッチ」ステップでルールアクションを選択

アクション 1 に「CloudWatch logs」を選択し、[CloudWatch Log グループを作成]をクリックします。



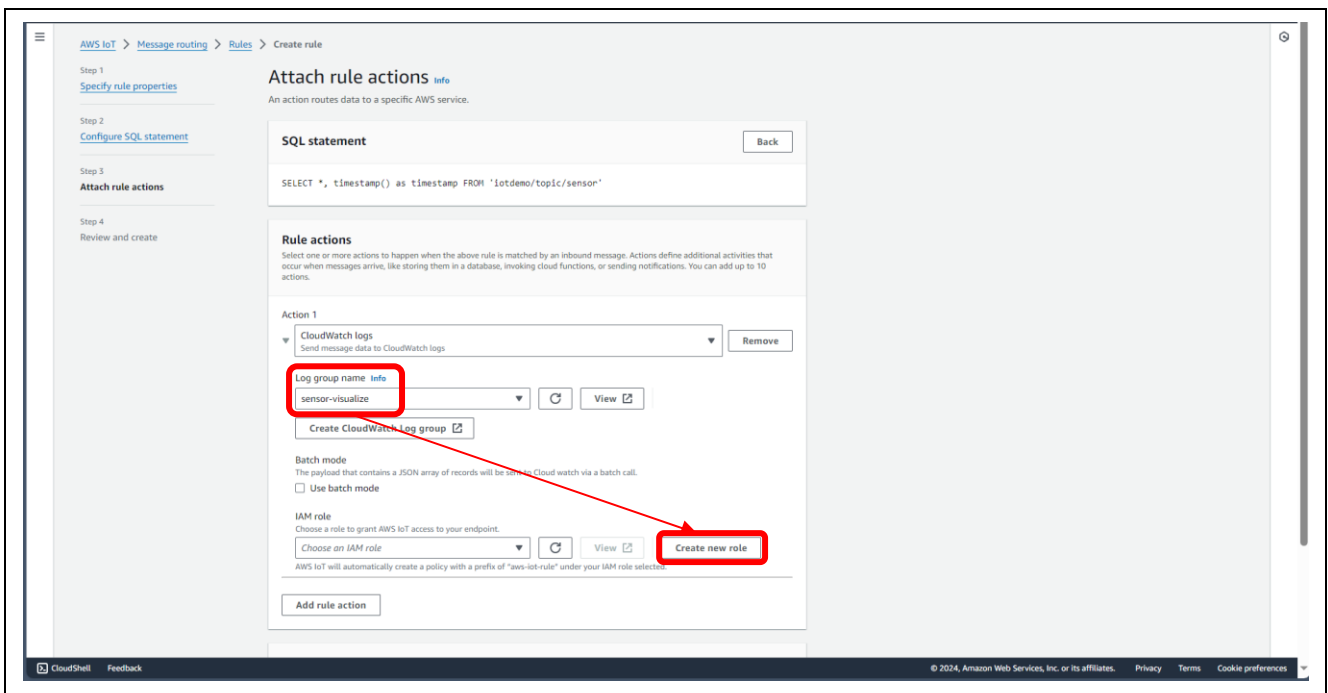
(5) ロググループを作成

ロググループ名を入力し、作成をクリックします。



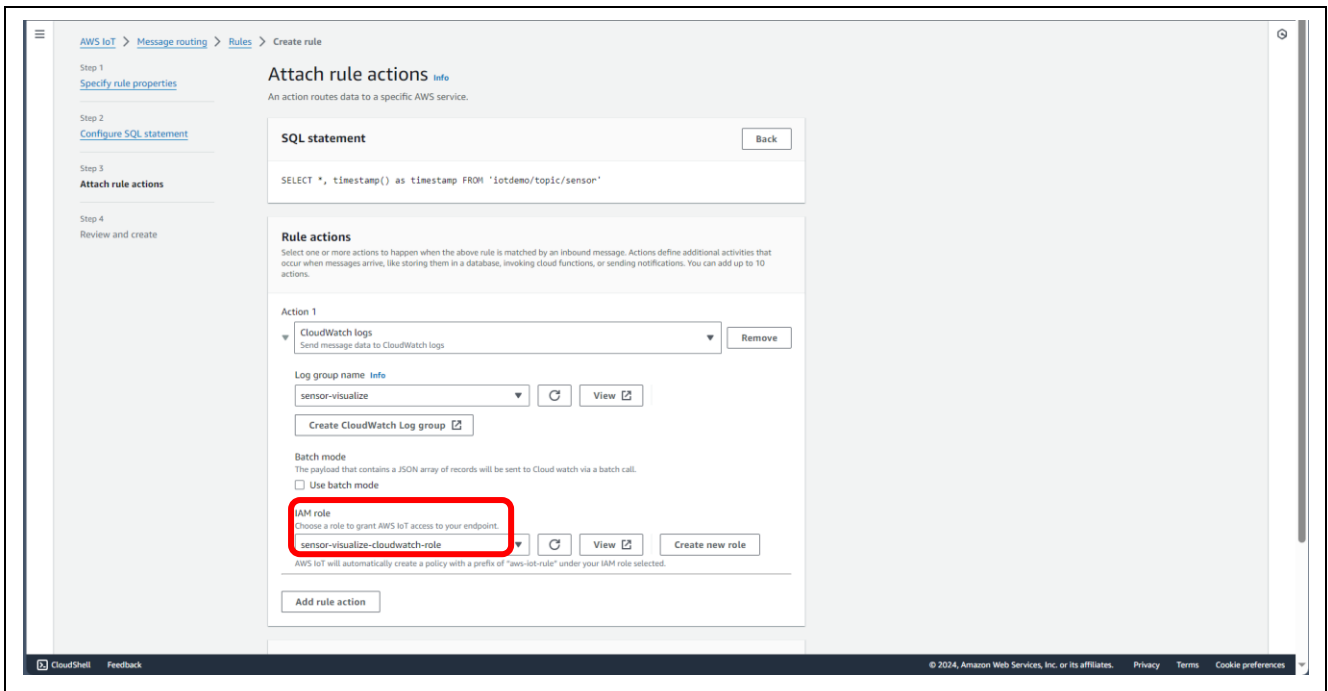
(6) 新しいロールを作成

ログのグループ名に、作成したロググループを選択し、[新しいロールを作成]をクリックします。



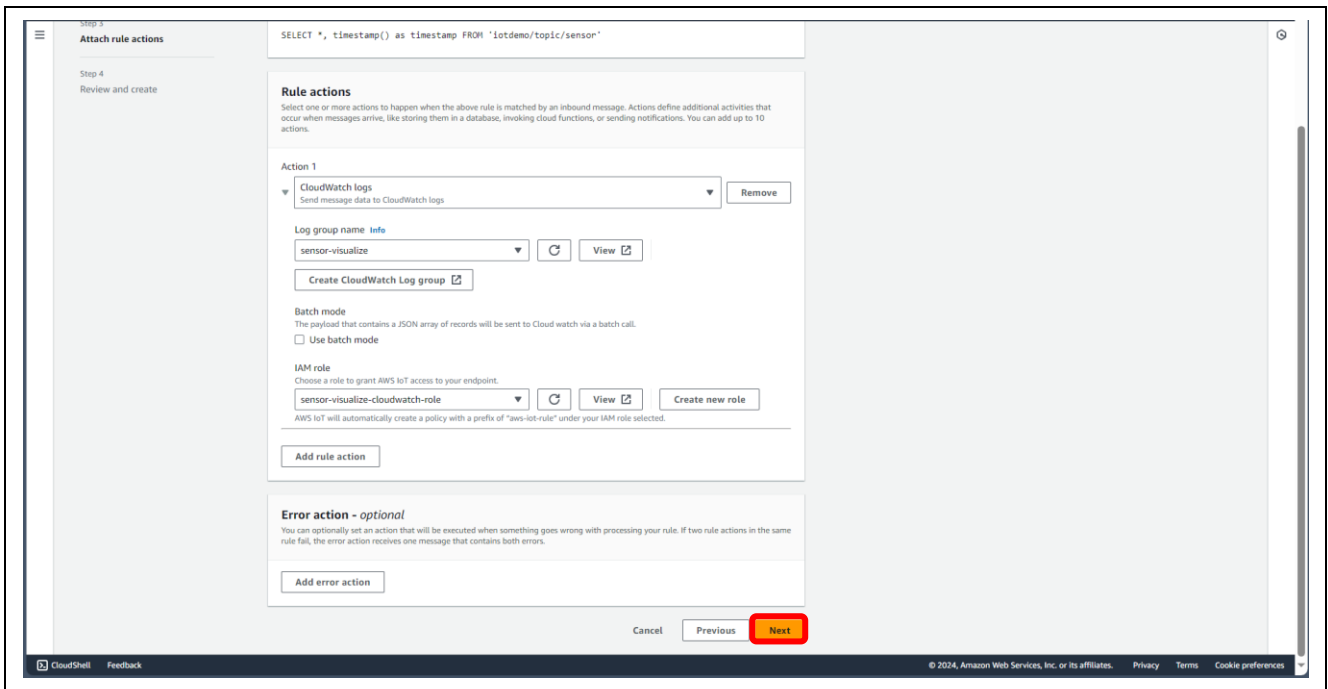
(7) 作成した IAM ロールを選択

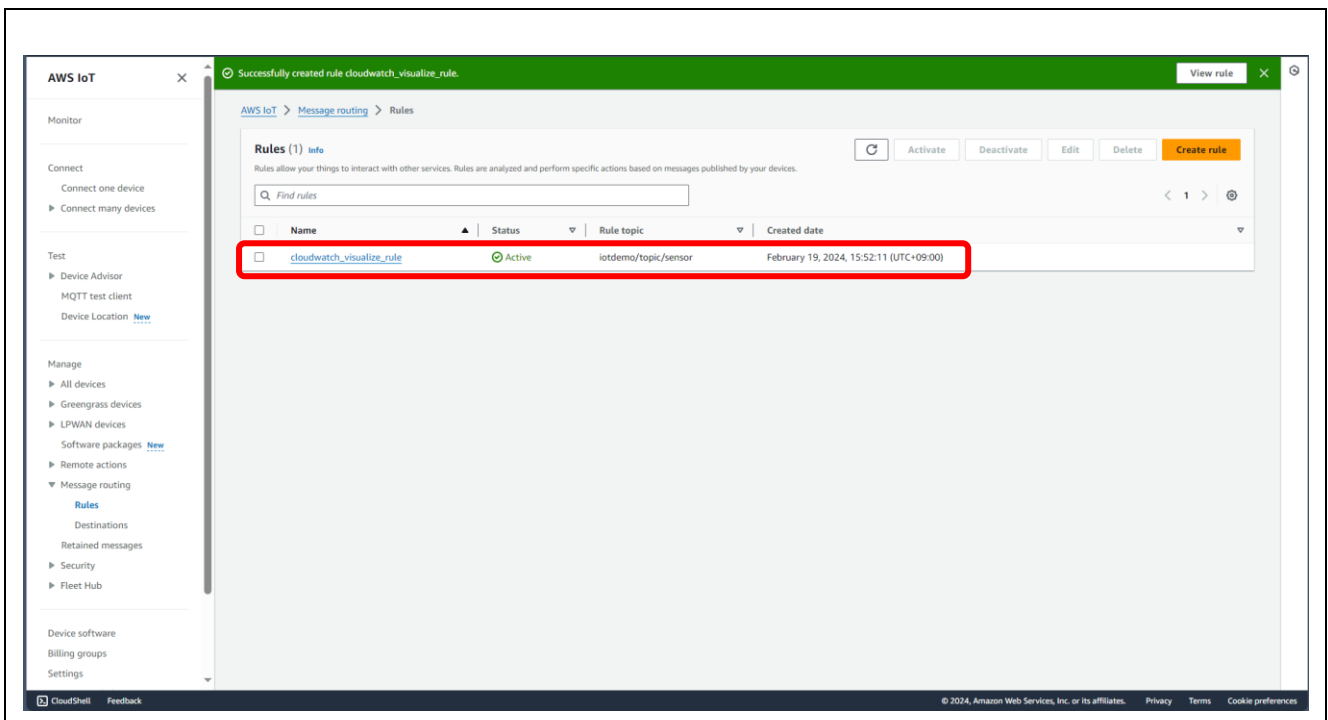
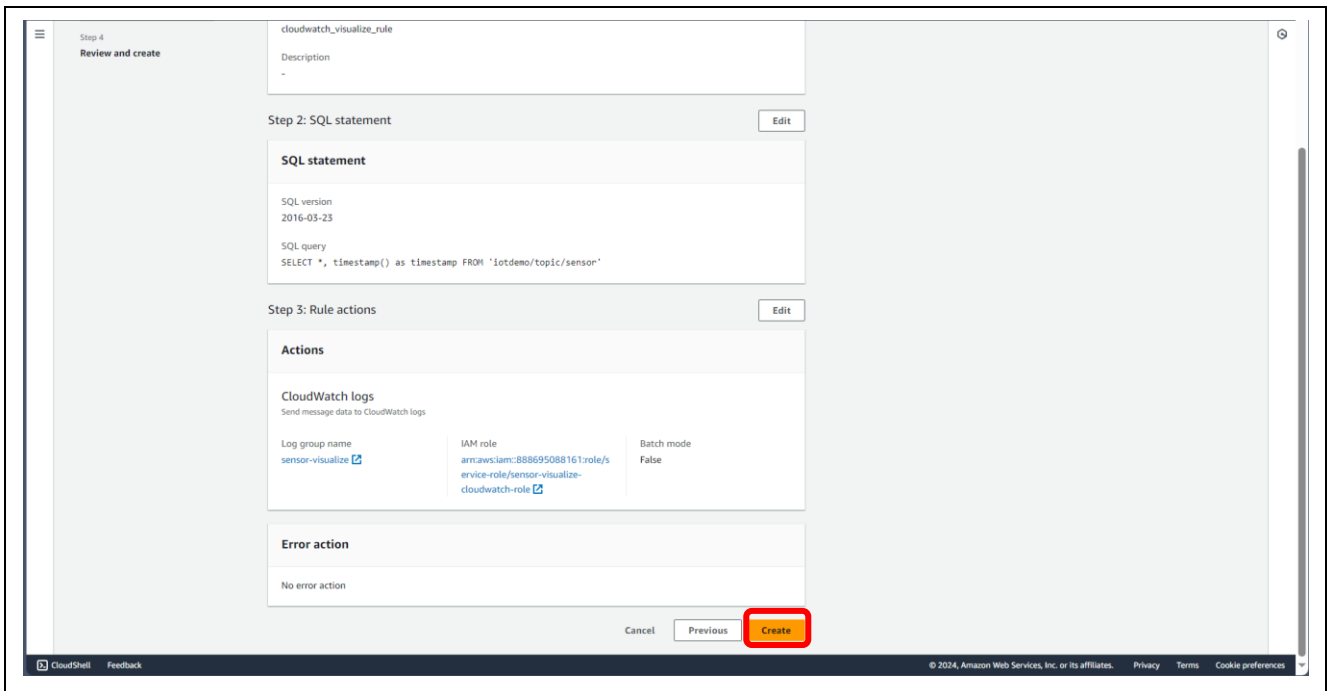
IAM ロールに、作成したロールを選択します。



(8) ルール作成に成功したことを確認

[次へ] ⇒ [作成]と進み、ルールの一覧に、作成したルールが表示されていることを確認します。





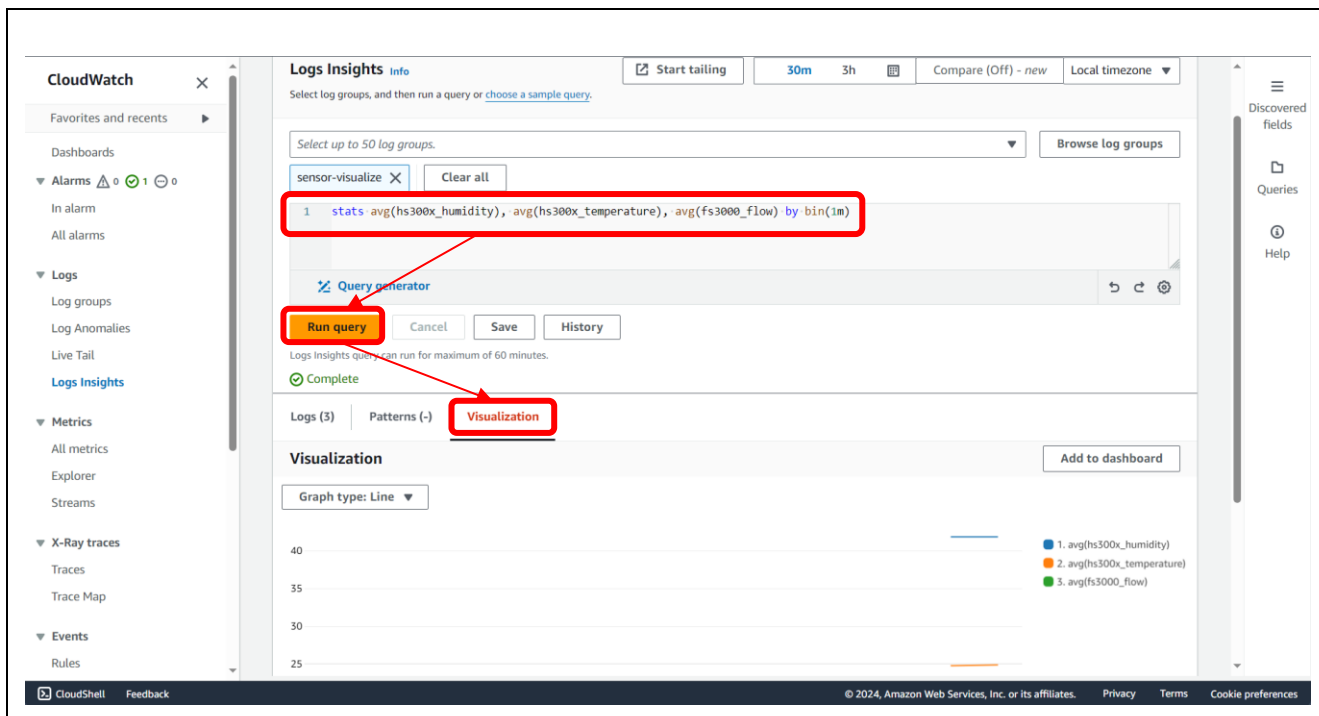
(9) CloudWatch でグラフ表示を確認

CloudWatch ページを表示し、左メニューの「ログのインサイト」をクリックします。

ロググループに、先ほど 6.3.2.1(5)で作成したグループを選択し、以下のクエリを入力し、[クエリの実行]をクリックします。

```
stats avg(hs300x_humidity), avg(hs300x_temperature), avg(fs3000_flow) by bin(1m)
```

可視化タブにグラフが表示されます。



7. デモの実行手順

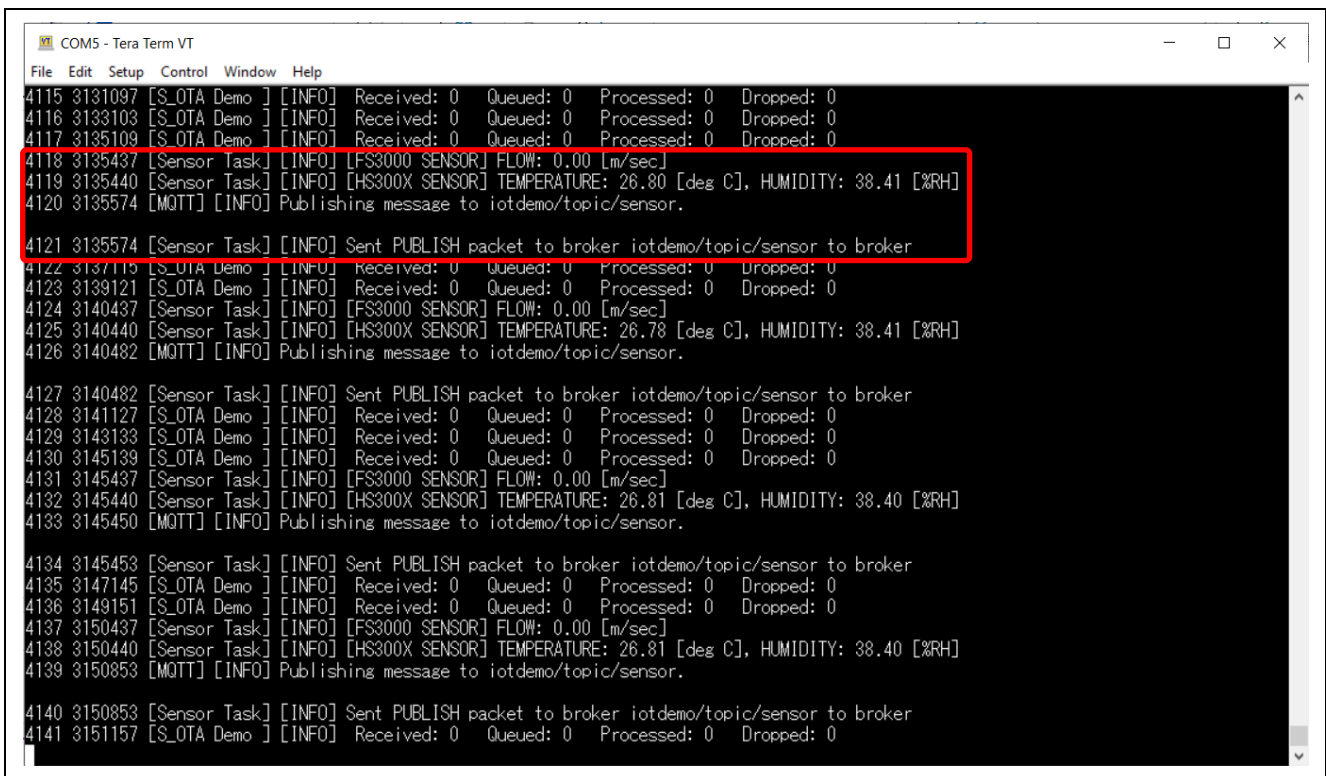
デモを実行する詳細な手順について説明します。

7.1 初期状態の動作確認

6章のデモのセットアップが完了している状態で、RL78/G22 FPB のリセットスイッチ(RST)を押下しハードウェアリセットを実行します。同様に CK-RX65N もリセットスイッチ(S1)を押下しハードウェアリセットを実行します。

ターミナルソフトで各マイコンからのログを確認します。

図 7-1 に CK-RX65N のログ画面を示します。HS3001 センサのデータが出力されていることを確認します。FS3000 センサはデータ取得していないため、値が 0.00 となっています。また、その下には MQTT 通信でセンサデータを AWS へ送信しているログが表示されます。



```
COM5 - Tera Term VT
File Edit Setup Control Window Help
4115 3131097 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
4116 3133103 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
4117 3135109 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
4118 3135437 [Sensor Task] [INFO] [FS3000 SENSOR] FLOW: 0.00 [m/sec]
4119 3135440 [Sensor Task] [INFO] [HS300X SENSOR] TEMPERATURE: 26.80 [deg C], HUMIDITY: 38.41 [%RH]
4120 3135574 [MQTT] [INFO] Publishing message to iotdemo/topic/sensor.
4121 3135574 [Sensor Task] [INFO] Sent PUBLISH packet to broker iotdemo/topic/sensor to broker
4122 3137115 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
4123 3139121 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
4124 3140437 [Sensor Task] [INFO] [FS3000 SENSOR] FLOW: 0.00 [m/sec]
4125 3140440 [Sensor Task] [INFO] [HS300X SENSOR] TEMPERATURE: 26.78 [deg C], HUMIDITY: 38.41 [%RH]
4126 3140482 [MQTT] [INFO] Publishing message to iotdemo/topic/sensor.
4127 3140482 [Sensor Task] [INFO] Sent PUBLISH packet to broker iotdemo/topic/sensor to broker
4128 3141127 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
4129 3143133 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
4130 3145139 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
4131 3145437 [Sensor Task] [INFO] [FS3000 SENSOR] FLOW: 0.00 [m/sec]
4132 3145440 [Sensor Task] [INFO] [HS300X SENSOR] TEMPERATURE: 26.81 [deg C], HUMIDITY: 38.40 [%RH]
4133 3145450 [MQTT] [INFO] Publishing message to iotdemo/topic/sensor.
4134 3145453 [Sensor Task] [INFO] Sent PUBLISH packet to broker iotdemo/topic/sensor to broker
4135 3147145 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
4136 3149151 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
4137 3150437 [Sensor Task] [INFO] [FS3000 SENSOR] FLOW: 0.00 [m/sec]
4138 3150440 [Sensor Task] [INFO] [HS300X SENSOR] TEMPERATURE: 26.81 [deg C], HUMIDITY: 38.40 [%RH]
4139 3150853 [MQTT] [INFO] Publishing message to iotdemo/topic/sensor.
4140 3150853 [Sensor Task] [INFO] Sent PUBLISH packet to broker iotdemo/topic/sensor to broker
4141 3151157 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
```

図 7-1 CK-RX65N のログ画面

次に、図 7-2 に RL78/G22 FPB のログ画面を示します。HS3001 センサのデータのみ表示されていることを確認します。

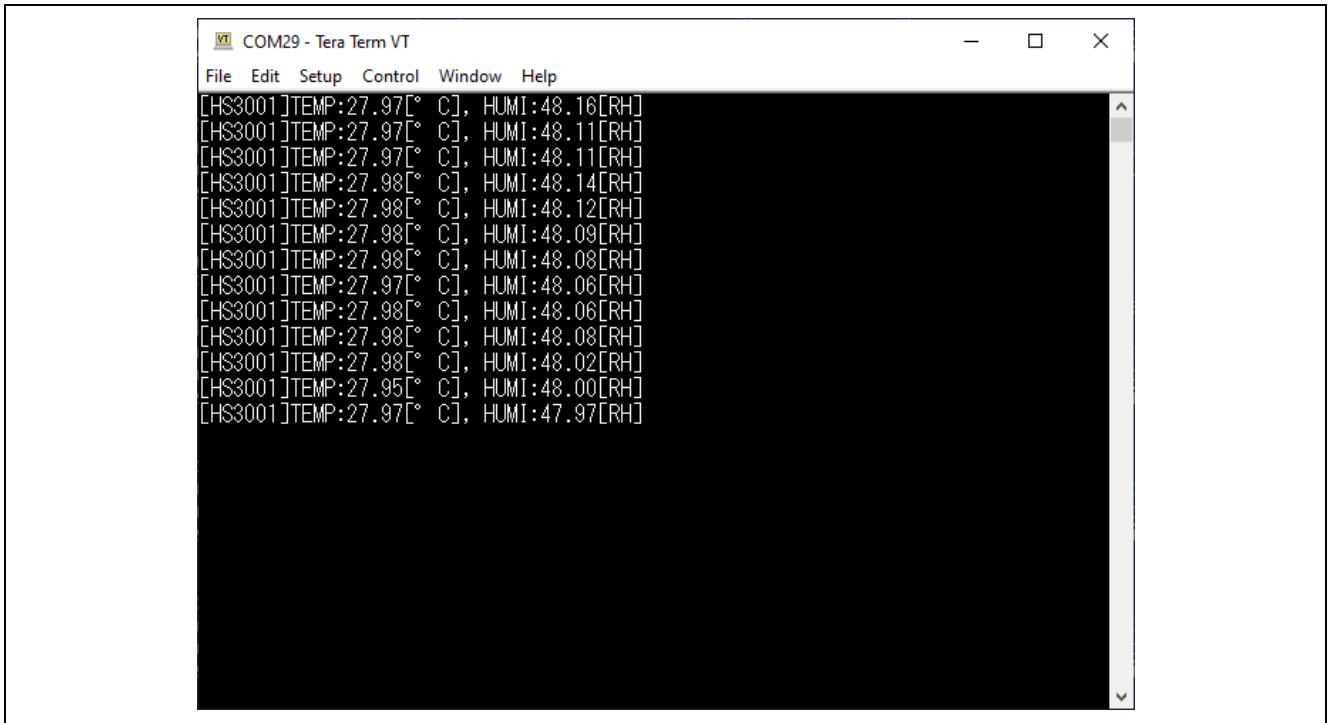


図 7-2 RL78/G22 FPB のログ画面

最後に、図 7-3 に Amazon CloudWatch の画面を示します。HS3001 センサから取得した温湿度データがグラフ化されていることを確認します。

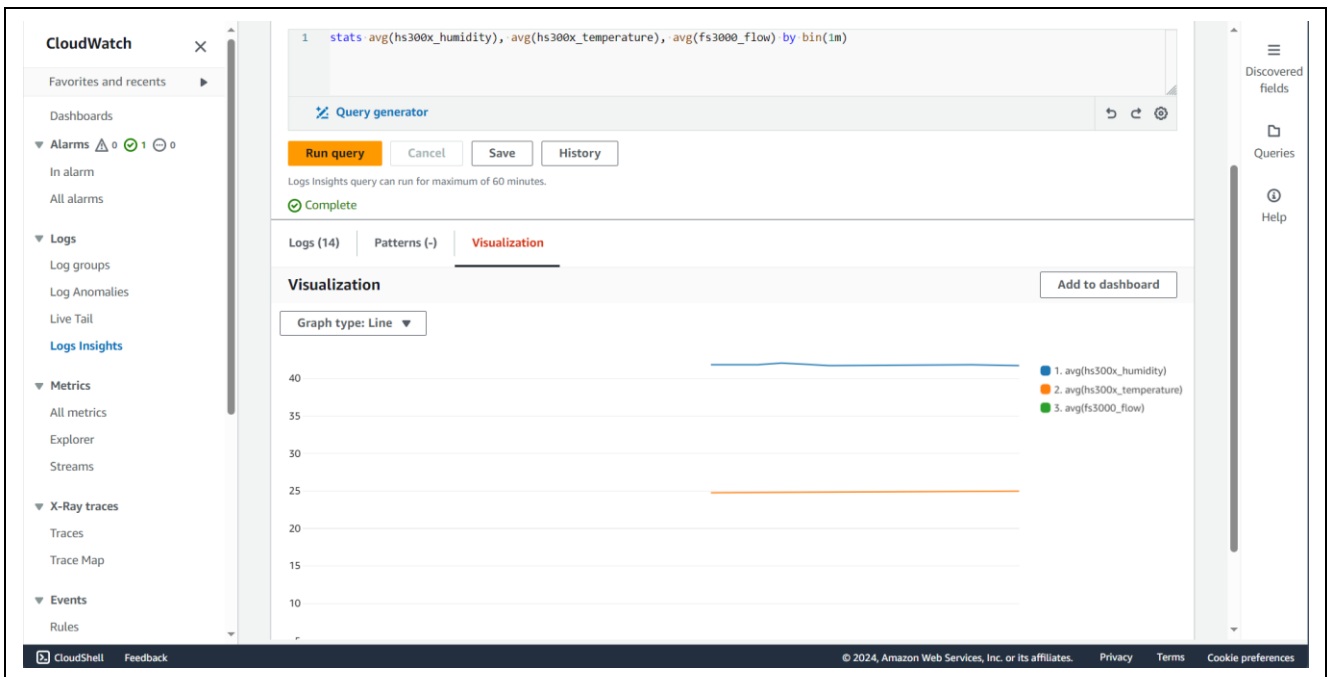


図 7-3 セカンダリ OTA アップデート前の Amazon CloudWatch のグラフ表示

この状態がセカンダリ OTA アップデート実行前の初期状態です。

7.2 RL78/G22 FPB の OTA アップデートの実行

7.2.1 更新ファームウェアの作成

7.2.1.1 半面更新方式の場合

(1) rl78g22_fpb_2ndota_demo プロジェクトのソースコードの変更

rl78g22_fpb_2ndota_demo/src/rl78g22_fpb_2ndota_demo.c の 53 行目の「USE_SENSOR_FS3000」マクロの定義を 0 から 1 に変更します。

また、FWUP_DEMO_VER_MAJOR, MINOR, BUILD を任意の値にすることで実行時に表示されるファームウェアバージョンの変更が可能です。

```

50      + Macro definitions
51
52      #define USE_SENSOR_HS3001          (1)
53      #define USE_SENSOR_FS3000        (0)
54
55      #define FWUP_DEMO_VER_MAJOR      (1)
56      #define FWUP_DEMO_VER_MINOR     (0)
57      #define FWUP_DEMO_VER_BUILD     (0)
58

```

(2) 更新ファームウェア(MOT ファイル形式)の作成

rl78g22_fpb_2ndota_demo プロジェクトをビルドし、MOT ファイルを作成します。

(3) 更新ファームウェア(RSU ファイル形式)の作成

作成された rl78g22_fpb_2ndota_demo の MOT ファイルを、Renesas Image Generator を使用して RSU 形式の更新ファームウェアに変換します。

r01an6935jj0100-rl78g22/RenesasImageGenerator フォルダ内で以下のコマンドを実行し、RSU 形式の更新ファームウェア update_firm.rsu を作成します。

```

> python .%image-gen.py -ip .%RL78_G22_ImageGenerator_PRM.csv -
iup ..%rl78g22_fpb_2ndota_demo%HardwareDebug%rl78g22_fpb_2ndota_demo.mot -o update_firm

```

7.2.1.2 全面更新方式の場合

「7.2.1.1 半面更新方式の場合」と同様の手順で更新ファームウェアを作成します。使用するプロジェクトを rl78g22_fpb_2ndota_demo プロジェクトから rl78g22_fpb_2ndota_demo_full プロジェクトに置き換えてください。

また、RSU ファイルを作成するコマンドは以下となります。

```

> python .%image-gen.py -ip .%RL78_G22_FullUpdate_ImageGenerator_PRM.csv -
iup ..%rl78g22_fpb_2ndota_demo_full%HardwareDebug%rl78g22_fpb_2ndota_demo_full.mot -o
update_firm

```


7.2.2 AWS で OTA ジョブの作成

(1) AWS マネジメントコンソールにサインインし、左上の「Service」→「IoT」→「IoT Core」を選択します。

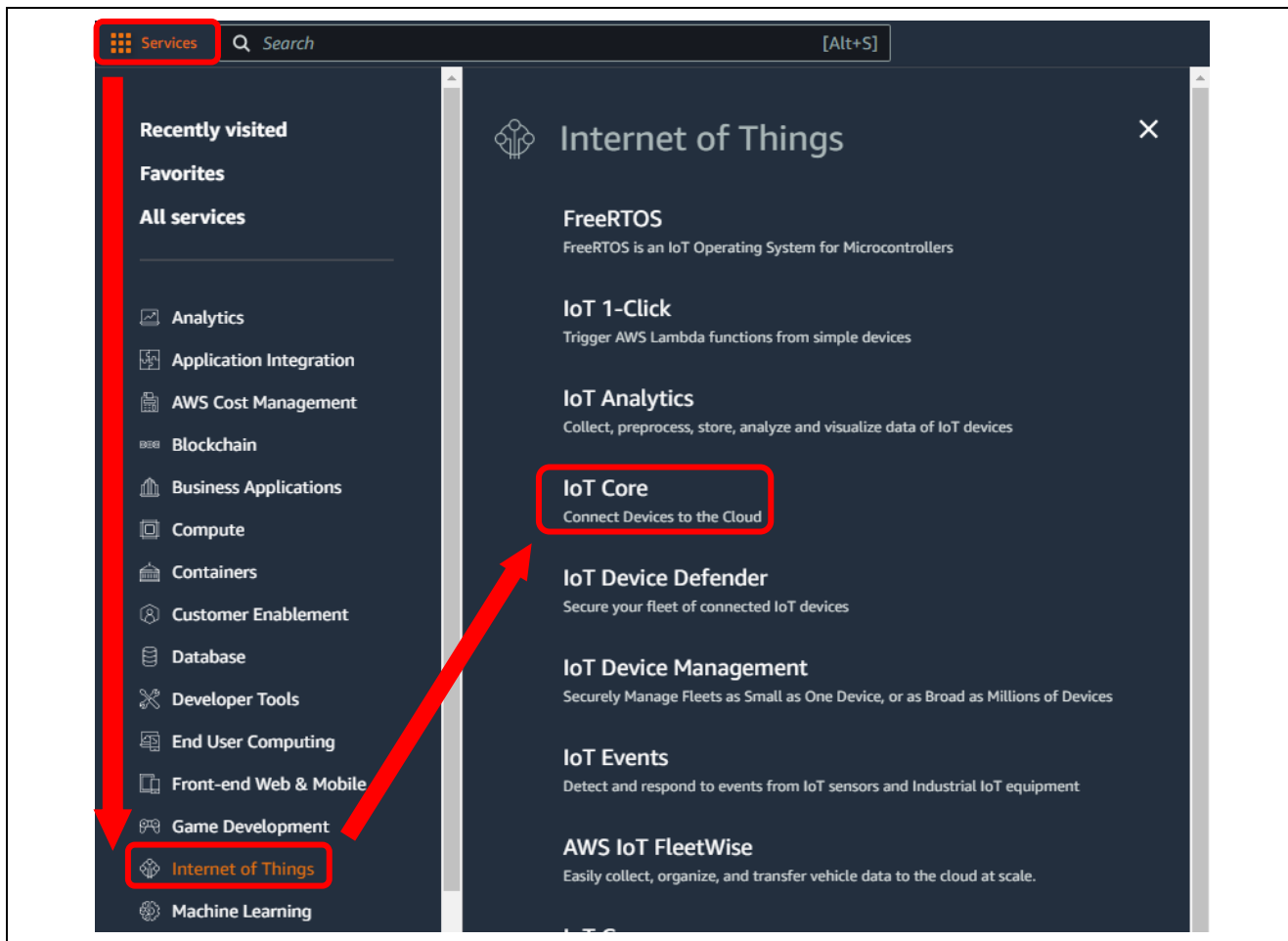


図 7-4 AWS のサービス画面

(2) AWS IoT Core の左のメニューから **Remote action** → **Jobs** を選択し、**[Create job]**をクリックします。

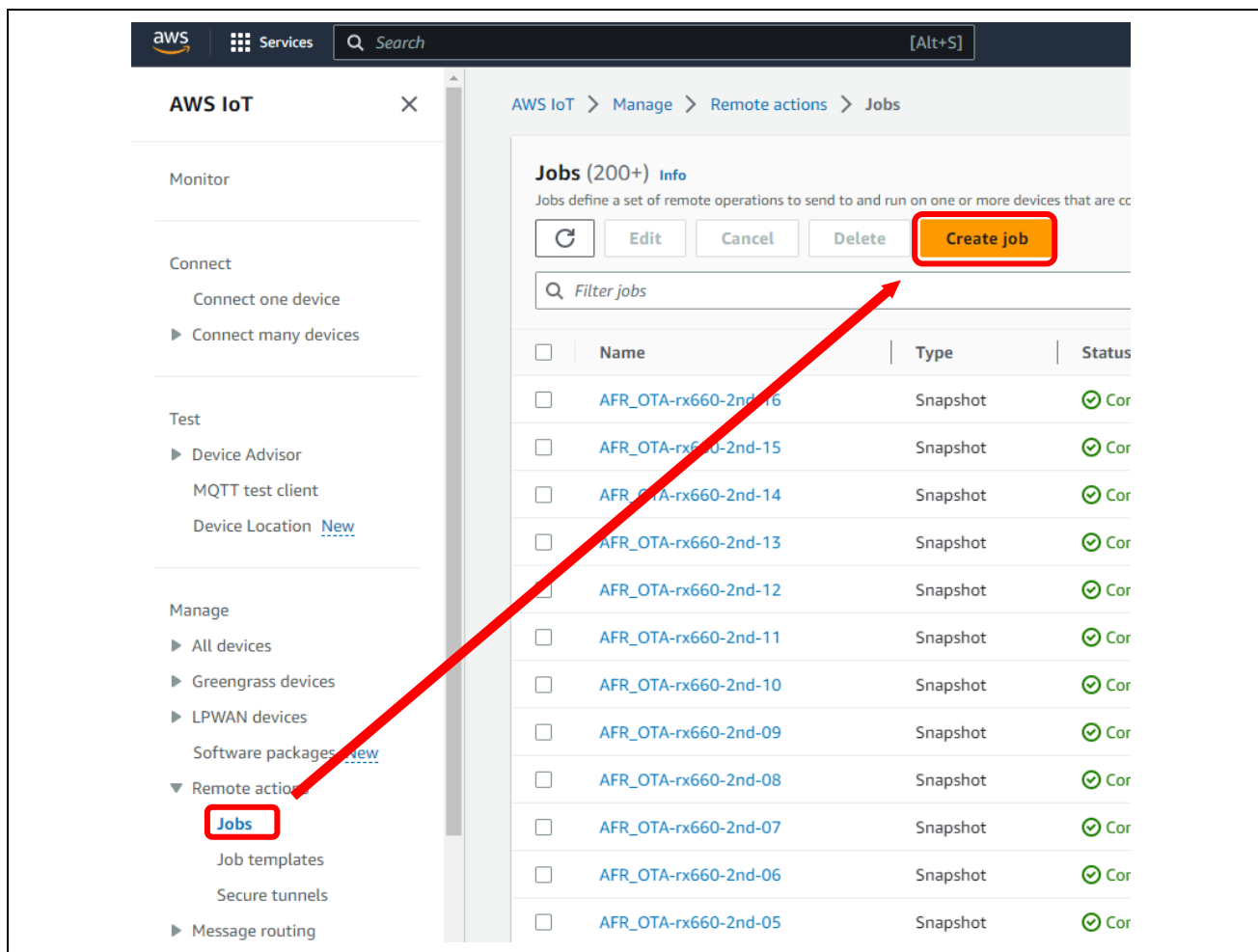


図 7-5 AWS IoT Core 画面

(3) 「Create job」画面で「Create FreeRTOS OTA update job」を選択して[Next]をクリックします。

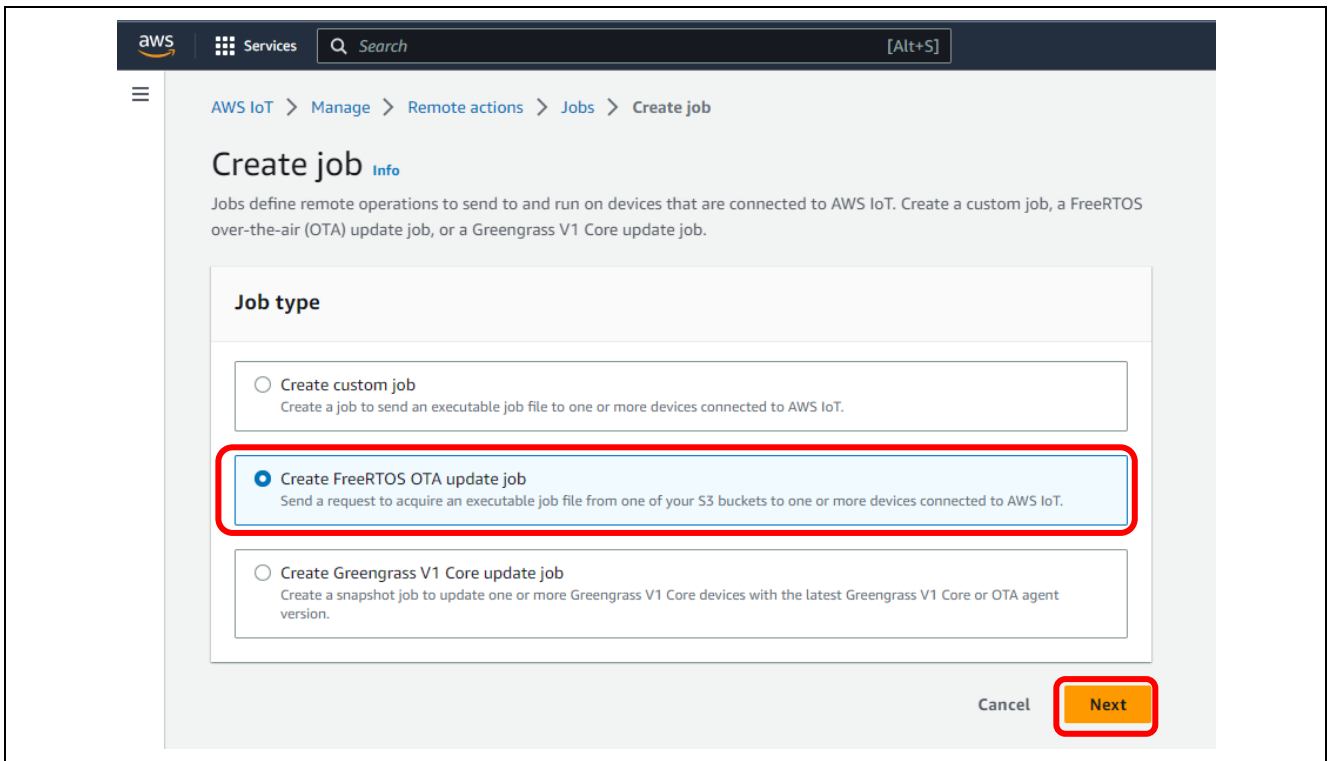


図 7-6 ジョブ作成画面

(4) 「OTA job properties」画面で「Job name」を入力し、[Next]をクリックします。

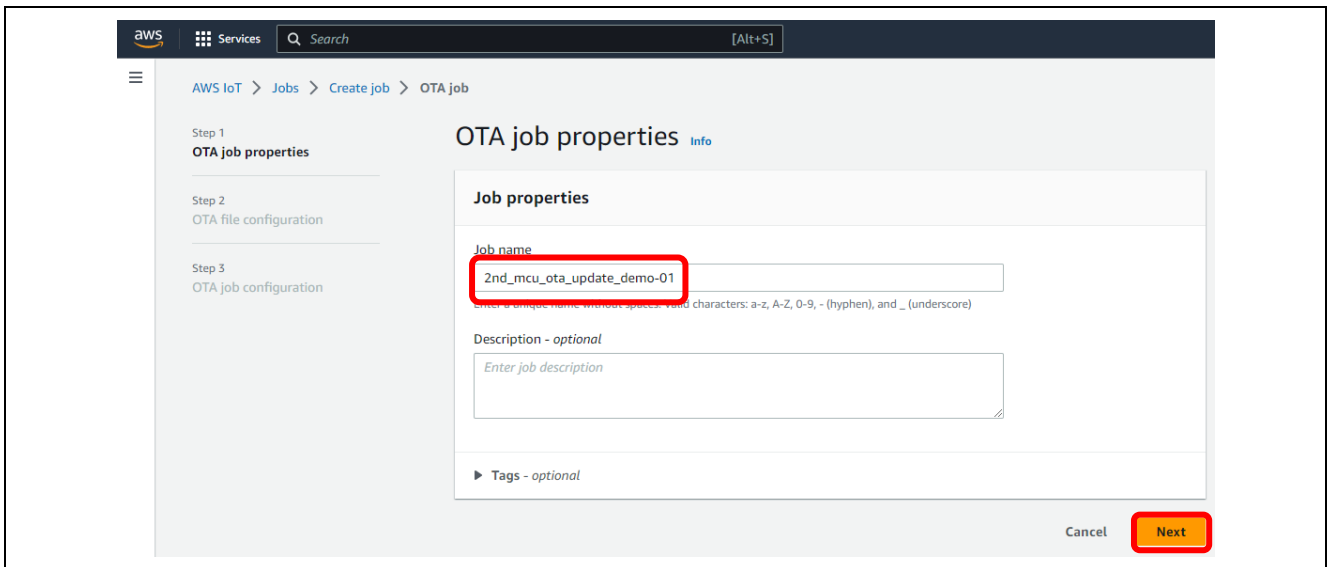


図 7-7 OTA ジョブのプロパティ入力画面

- (5) 「OTA ファイル設定」画面で各種項目を入力します。
- ① 「Device to update」では 6.2.3(6)章で QE for OTA で設定した IoT device name を選択します。
 - ② 「Select the protocol for file transfer」では MQTT を選択します。
 - ③ 「Sign and choose your file」では「Sign a new file for me.」を選択します。
 - ④ 「Code signing profile」では 6.3.1(3)章で作成した Code signing profile を選択します。
Note 半面更新方式の場合、ここで指定したコード署名証明書のプロファイルは更新ファームウェアのイメージ検証には使用されないため、任意のプロファイルを指定できます。
全面更新方式の場合、ここで指定したコード署名証明書のプロファイルはプライマリ MCU 側での更新ファームウェアのイメージ検証に使用されます。
 - ⑤ 「File」では「Upload a new file.」を選択します。
 - ⑥ 「File to upload」では「Choose file」をクリックし、7.2.1 章で作成した RL78/G22 FPB の更新用のファームウェア(RSU 形式)を選択します。
 - ⑦ 「S3 URL」では「Browse S3」をクリックし、6.3.1(1)章で設定した Amazon S3 バケットを選択します。
 - ⑧ 「Path name of file on device」には任意の文字列を入力します。
 - ⑨ 「File type」には、
 - 半面更新方式の場合は 1
 - 全面更新方式の場合は 2を入力します。
 - ⑩ 「Role」には 6.3.1(2)章で設定した OTA 更新用サービスロールを選択します。
- 以上を入力後、[Next]をクリックします。

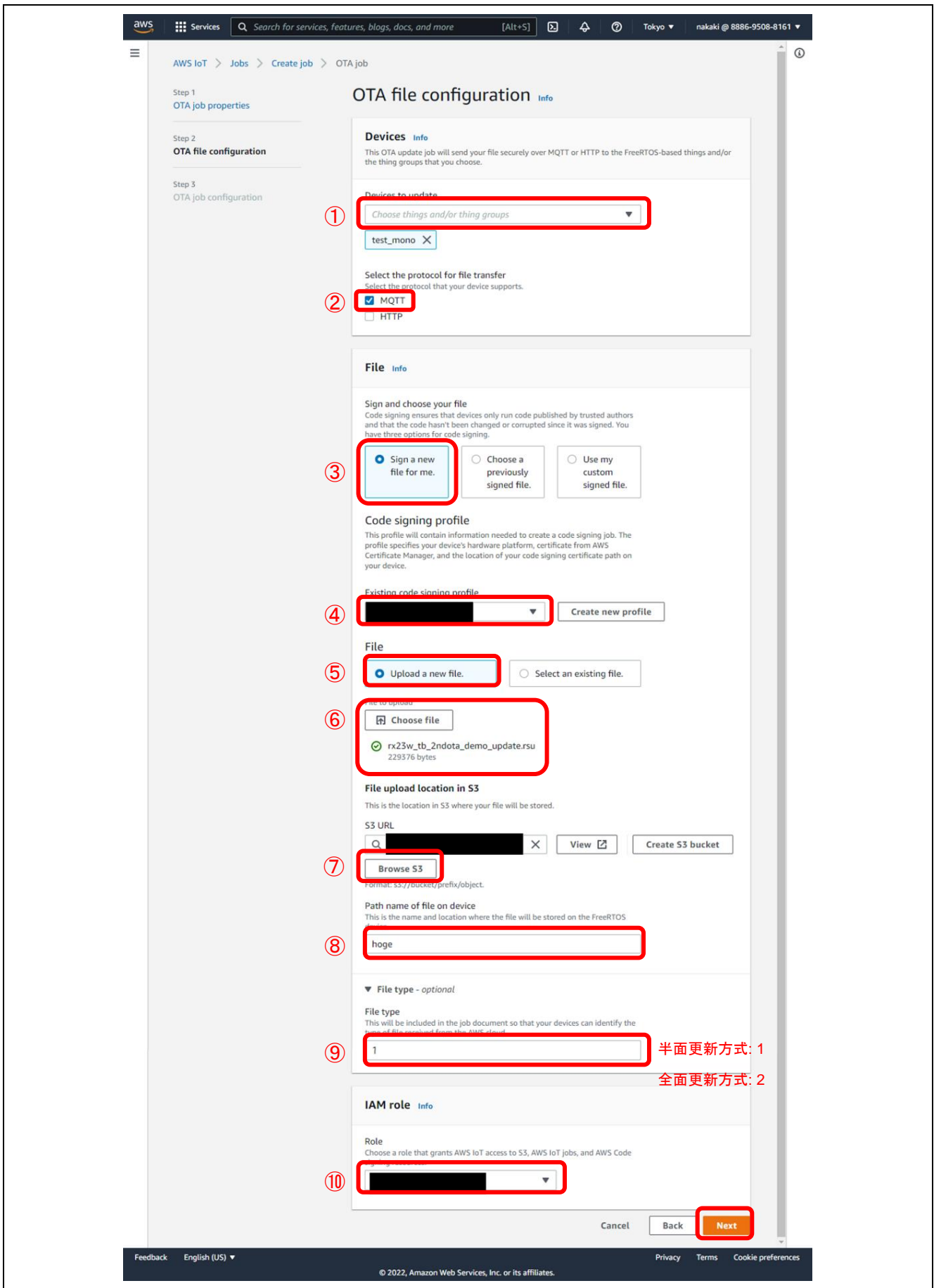


図 7-8 OTA ファイル設定画面

(6) 「OTA job configuration」画面では変更不要のため、そのまま[Create job]をクリックします。

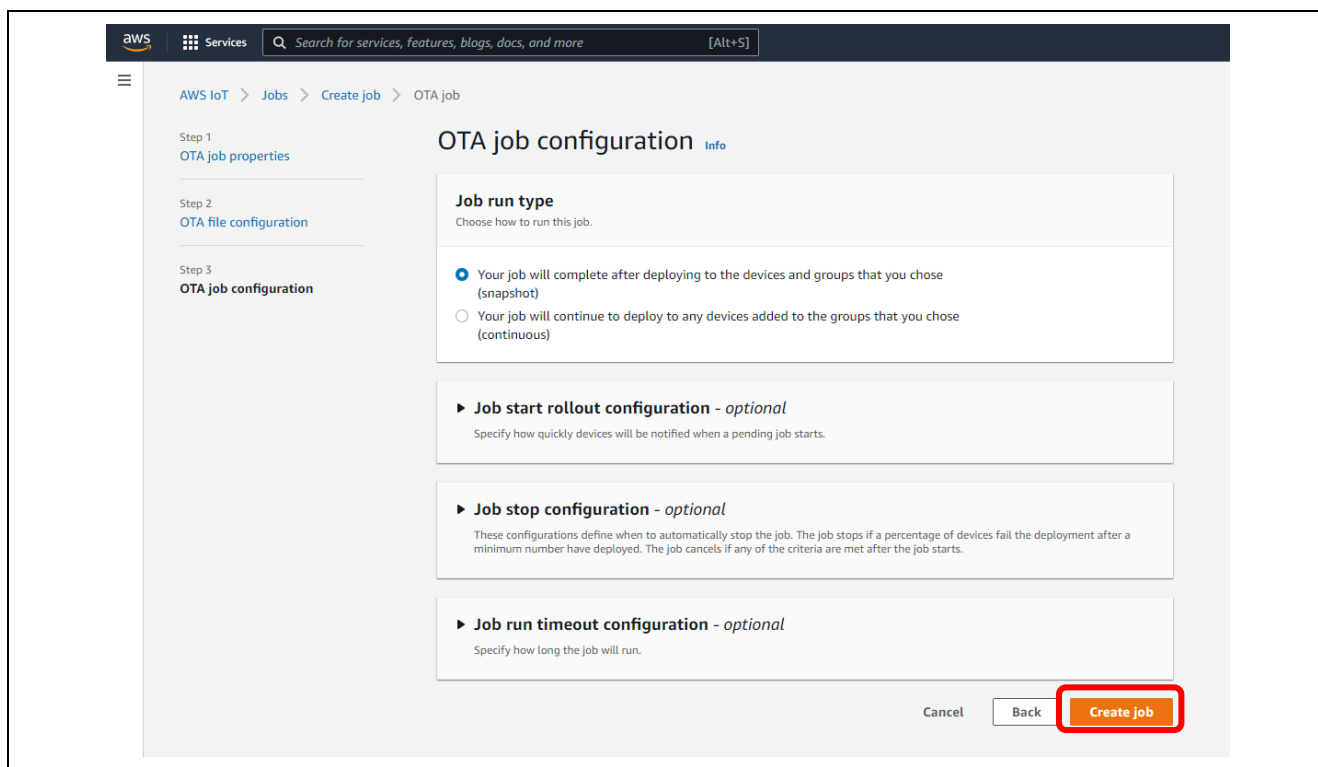


図 7-9 OTA ジョブ設定画面

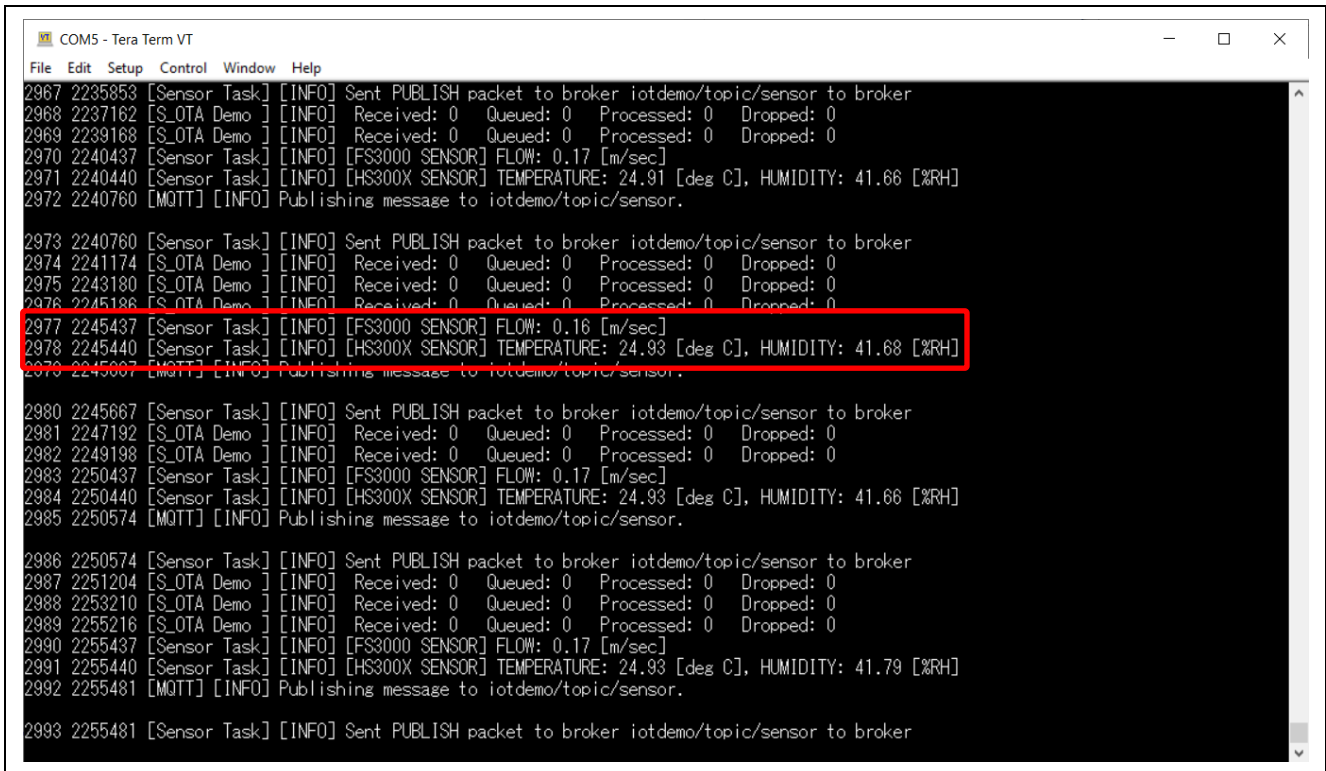
以上の手順でセカンダリ OTA アップデート用の OTA ジョブが作成され、指定したデバイスに向けて OTA ジョブが配信されます。

7.2.3 セカンダリ OTA アップデート実行中の動作確認

ジョブの作成から数秒で OTA アップデートが始まります。CK-RX65N と RL78/G22 FPB の両方から、セカンダリ OTA アップデートの進捗がログ出力されます。

7.3 OTA アップデート後の動作確認

図 7-10 にアップデート後の CK-RX65N のログ画面を示します。HS3001 センサに加えて FS3000 センサのデータが表示されていることを確認できます。



```
COM5 - Tera Term VT
File Edit Setup Control Window Help
2967 2235853 [Sensor Task] [INFO] Sent PUBLISH packet to broker iotdemo/topic/sensor to broker
2968 2237162 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
2969 2239168 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
2970 2240437 [Sensor Task] [INFO] [FS3000 SENSOR] FLOW: 0.17 [m/sec]
2971 2240440 [Sensor Task] [INFO] [HS300X SENSOR] TEMPERATURE: 24.91 [deg C], HUMIDITY: 41.66 [%RH]
2972 2240760 [MQTT] [INFO] Publishing message to iotdemo/topic/sensor.

2973 2240760 [Sensor Task] [INFO] Sent PUBLISH packet to broker iotdemo/topic/sensor to broker
2974 2241174 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
2975 2243180 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
2976 2245186 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
2977 2245437 [Sensor Task] [INFO] [FS3000 SENSOR] FLOW: 0.16 [m/sec]
2978 2245440 [Sensor Task] [INFO] [HS300X SENSOR] TEMPERATURE: 24.93 [deg C], HUMIDITY: 41.68 [%RH]
2979 2245807 [MQTT] [INFO] Publishing message to iotdemo/topic/sensor.

2980 2245867 [Sensor Task] [INFO] Sent PUBLISH packet to broker iotdemo/topic/sensor to broker
2981 2247192 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
2982 2249198 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
2983 2250437 [Sensor Task] [INFO] [FS3000 SENSOR] FLOW: 0.17 [m/sec]
2984 2250440 [Sensor Task] [INFO] [HS300X SENSOR] TEMPERATURE: 24.93 [deg C], HUMIDITY: 41.66 [%RH]
2985 2250574 [MQTT] [INFO] Publishing message to iotdemo/topic/sensor.

2986 2250574 [Sensor Task] [INFO] Sent PUBLISH packet to broker iotdemo/topic/sensor to broker
2987 2251204 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
2988 2253210 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
2989 2255216 [S_OTA Demo ] [INFO] Received: 0 Queued: 0 Processed: 0 Dropped: 0
2990 2255437 [Sensor Task] [INFO] [FS3000 SENSOR] FLOW: 0.17 [m/sec]
2991 2255440 [Sensor Task] [INFO] [HS300X SENSOR] TEMPERATURE: 24.93 [deg C], HUMIDITY: 41.79 [%RH]
2992 2255481 [MQTT] [INFO] Publishing message to iotdemo/topic/sensor.

2993 2255481 [Sensor Task] [INFO] Sent PUBLISH packet to broker iotdemo/topic/sensor to broker
```

図 7-10 ファームウェア更新後の CK-RX65N のログ画面

次に、図 7-11 に RL78/G22 FPB のログ画面を示します。

RL78/G22 FPB のファームウェア更新に成功すると、HS3001 センサと FS3000 センサの両センサから測定データを取得します。

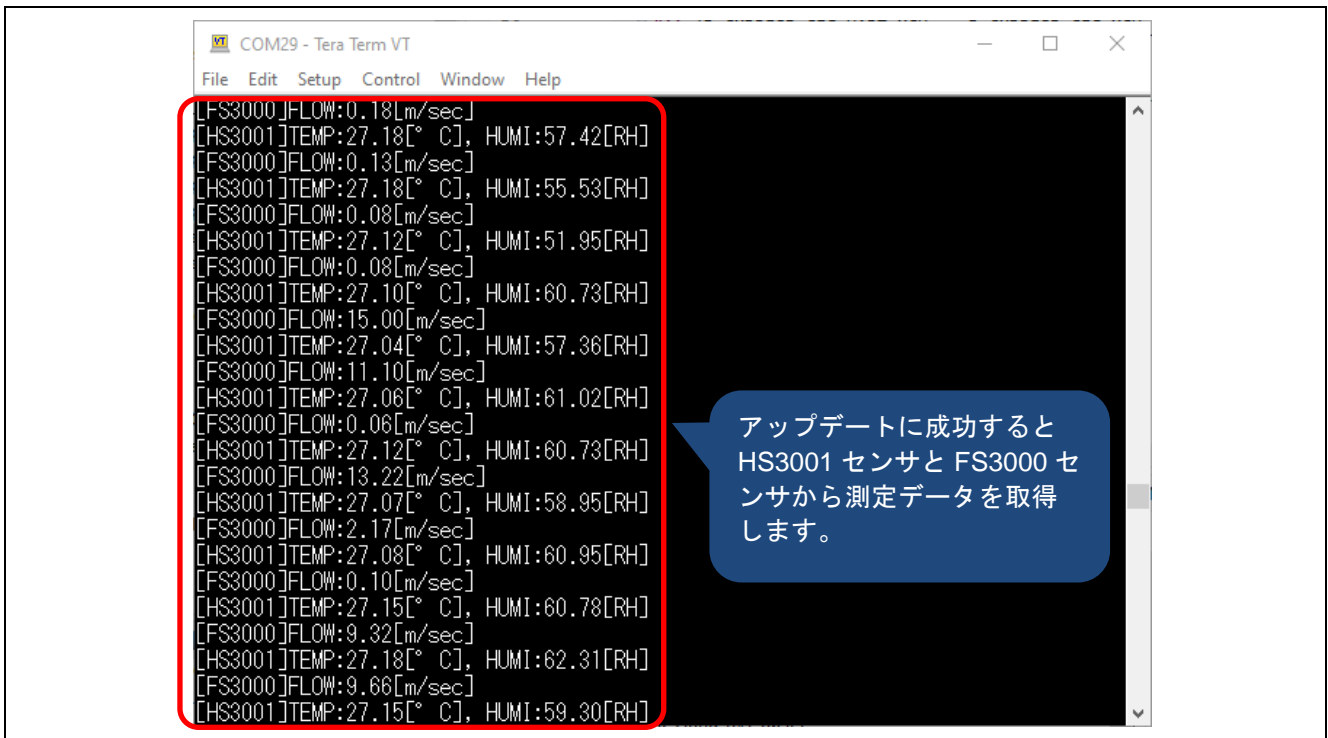


図 7-11 ファームウェア更新後の RL78/G22 FPB のログ画面

最後に、図 7-12 に Amazon CloudWatch の画面を示します。HS3001 センサから取得した温湿度データに加えて、FS3000 センサから取得した流量データがグラフ化されていることを確認します。

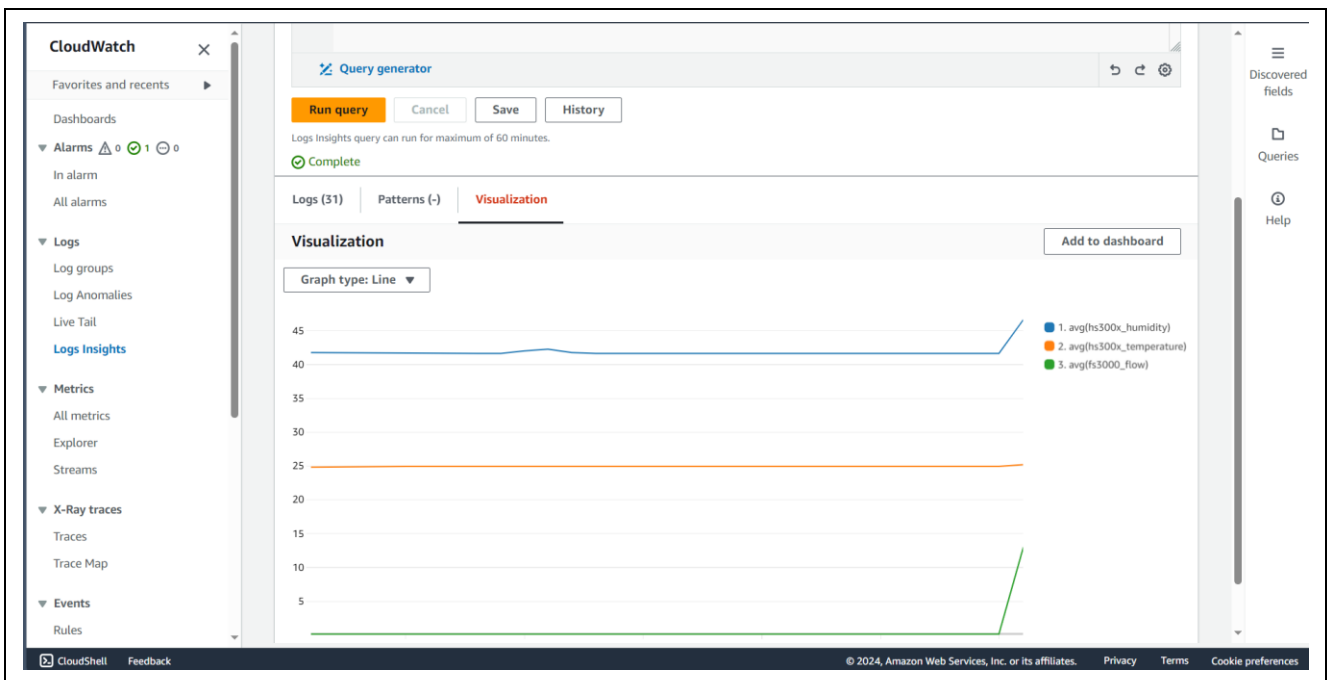


図 7-12 セカンダリ OTA アップデート後の Amazon CloudWatch のグラフ表示

デモ動作は以上となります。

8. 注意事項

8.1 使用するオープンソースソフトウェアのライセンス情報

以下のオープンソースソフトウェアを使用しています。

- TinyCrypt Cryptographic Library
 - URL <https://01.org/tinycrypt/>
 - ライセンス <https://github.com/intel/tinycrypt/blob/master/LICENSE>

- FreeRTOS
 - URL <https://www.freertos.org/>
 - ライセンス [FreeRTOS open source licensing, FreeRTOS license description, FreeRTOS license terms and OpenRTOS commercial licensing options.](#)

8.2 デモを実施する AWS のリージョンとユーザ権限について

デモ実施の際の AWS の設定について、使用リージョンとユーザ権限についての注意事項を以下に示します。

【使用リージョンについて】

このデモは AWS の ap-northeast-1（アジアパシフィック（東京））リージョンで実施しています。他のリージョンでこのデモを実施する場合は、デモで使用しているサービスがそのリージョンで提供されているか事前にご確認ください。

【ユーザ権限について】

このデモは AWS Identity and Access Management (IAM) で AdministratorAccess の権限が付与されたユーザで実行しています。そのため、各種サービスを使用する際の IAM での必要権限の付与に関して未記載です。

8.3 AWS の利用料金について

AWS の利用状況によっては、デモで作成・使用したクラウドリソースによって料金が発生する場合があります。意図しない課金を防ぐために、デモ実施後は作成したクラウド上のリソースを削除することをお勧めします。

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2023/6/30	—	初版
1.10	2024/3/29	—	ファームウェアアップデート方式に全面更新方式を追加
		—	呼称を 1 st MCU ⇒ プライマリ MCU、2 nd MCU ⇒ セカンダリ MCU に変更
		5	動作確認環境の各ソフトウェアのバージョンを更新
		16	動作電圧の説明を追加
		18	デモ構成の全体画像を追加
		23-24	プロジェクトのインポート方法の説明を追加
		25-27	RFP を用いた書き込み方法の説明を追加
		28	AWS クラウドの準備方法の参照先 APN を変更
		29	AWS のリージョンを確認する手順の説明を追加
		29-35	AWS 上でのセンサデータの可視化方法を OpenSearch から CloudWatch に変更
		47	「8. 注意事項」に AWS の利用料金に関する記述を追加

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本ドキュメントおよびテクニカルアップデートを参照してください。

1. 静電気対策

CMOS製品の取り扱いの際は静電気防止を心がけてください。CMOS製品は強い静電気によってゲート絶縁破壊を生じることがあります。運搬や保存の際には、当社が出荷梱包に使用している導電性のトレーやマガジンケース、導電性の緩衝材、金属ケースなどを利用し、組み立て工程にはアースを施してください。プラスチック板上に放置したり、端子を触ったりしないでください。また、CMOS製品を実装したボードについても同様の扱いをしてください。

2. 電源投入時の処置

電源投入時は、製品の状態は不定です。電源投入時には、LSIの内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. 電源オフ時における入力信号

当該製品の電源がオフ状態のときに、入力信号や入出力プルアップ電源を入れしないでください。入力信号や入出力プルアップ電源からの電流注入により、誤動作を引き起こしたり、異常電流が流れ内部素子を劣化させたりする場合があります。資料中に「電源オフ時における入力信号」についての記載のある製品は、その内容を守ってください。

4. 未使用端子の処理

未使用端子は、「未使用端子の処理」に従って処理してください。CMOS製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI周辺のノイズが印加され、LSI内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。

5. クロックについて

リセット時は、クロックが安定した後、リセットを解除してください。プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

6. 入力端子の印加波形

入力ノイズや反射波による波形歪みは誤動作の原因になりますので注意してください。CMOS製品の入力がノイズなどに起因して、 V_{IL} (Max.) から V_{IH} (Min.) までの領域にとどまるような場合は、誤動作を引き起こす恐れがあります。入力レベルが固定の場合はもちろん、 V_{IL} (Max.) から V_{IH} (Min.) までの領域を通過する遷移期間中にチャタリングノイズなどが入らないように使用してください。

7. リザーブアドレス（予約領域）のアクセス禁止

リザーブアドレス（予約領域）のアクセスを禁止します。アドレス領域には、将来の拡張機能用に割り付けられている リザーブアドレス（予約領域）があります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

8. 製品間の相違について

型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。同じグループのマイコンでも型名が違えば、フラッシュメモリ、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ輻射量などが異なる場合があります。型名が違う製品に変更する場合は、個々の製品ごとにシステム評価試験を実施してください。

ご注意書き

- 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。回路、ソフトウェアおよびこれらに関連する情報を使用する場合、お客様の責任において、お客様の機器・システムを設計ください。これらの使用に起因して生じた損害（お客様または第三者いずれに生じた損害も含まれます。以下同じです。）に関し、当社は、一切その責任を負いません。
 - 当社製品または本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
 - 当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
 - 当社製品を組み込んだ製品の輸出入、製造、販売、利用、配布その他の行為を行うにあたり、第三者保有の技術の利用に関するライセンスが必要となる場合、当該ライセンス取得の判断および取得はお客様の責任において行ってください。
 - 当社製品を、全部または一部を問わず、改造、変更、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、改造、複製、リバースエンジニアリング等により生じた損害に関し、当社は、一切その責任を負いません。
 - 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット等
高品質水準： 輸送機器（自動車、電車、船舶等）、交通管制（信号）、大規模通信機器、金融端末基幹システム、各種安全制御装置等
当社製品は、データシート等により高信頼性、Harsh environment 向け製品と定義しているものを除き、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じても、当社は一切その責任を負いません。
 - あらゆる半導体製品は、外部攻撃からの安全性を 100%保証されているわけではありません。当社ハードウェア/ソフトウェア製品にはセキュリティ対策が組み込まれているものもありますが、これによって、当社は、セキュリティ脆弱性または侵害（当社製品または当社製品が使用されているシステムに対する不正アクセス・不正使用を含みますが、これに限られません。）から生じる責任を負うものではありません。当社は、当社製品または当社製品が使用されたあらゆるシステムが、不正な改変、攻撃、ウイルス、干渉、ハッキング、データの破壊または窃盗その他の不正な侵入行為（「脆弱性問題」といいます。）によって影響を受けないことを保証しません。当社は、脆弱性問題に起因したまたはこれに関連して生じた損害について、一切責任を負いません。また、法令において認められる限りにおいて、本資料および当社ハードウェア/ソフトウェア製品について、商品性および特定目的との合致に関する保証ならびに第三者の権利を侵害しないことの保証を含め、明示または黙示のいかなる保証も行いません。
 - 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合および事故につきましては、当社は、一切その責任を負いません。
 - 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は、データシート等において高信頼性、Harsh environment 向け製品と定義しているものを除き、耐放射線設計を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
 - 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。かかる法令を遵守しないことにより生じた損害に関して、当社は、一切その責任を負いません。
 - 当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
 - お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものとなります。
 - 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
 - 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。
- 注 1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社が直接的、間接的に支配する会社をいいます。
- 注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

(Rev.5.0-1 2020.10)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレストシア）

www.renesas.com

お問合せ窓口

弊社の製品や技術、ドキュメントの最新情報、最寄の営業お問合せ窓口に関する情報などは、弊社ウェブサイトをご覧ください。

www.renesas.com/contact/

商標について

ルネサスおよびルネサスロゴはルネサス エレクトロニクス株式会社の商標です。すべての商標および登録商標は、それぞれの所有者に帰属します。