

# 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

### 要旨

本アプリケーションノートでは、クラウド向け開発支援ツール(QE for OTA)を用いて、クラウドを利用したOTA(Over the Air)によるファームウェア更新の導入を行う方法を説明します。

### 動作確認デバイス

CK-RX65N (Cloud Kit for RX65N Microcontroller Group) セルラー対応版

### 目次

1. 概要	3
1.1 システム概要	3
1.2 QE for OTA の構成	3
1.2.1 OTA メイン(QE) ビュー	3
1.2.2 OTA IoT デバイス管理(QE) ビュー	4
1.3 動作環境	5
1.4 参考資料	7
2. 事前準備	8
2.1 QE for OTA のインストール	9
2.1.1 Renesas Software Installer を使用する場合	9
2.1.2 Web から QE をダウンロードしてインストールする場合	10
2.1.3 インストール手順 (共通)	11
2.2 Cloud アカウントの作成	12
2.2.1 【AWS の場合】	12
2.2.2 【Azure の場合】	12
3. 開発手順	13
4. 動作確認例	14
4.1 準備	14
4.2 QE for OTA 起動	14
4.3 Cloud の設定	16
4.3.1 Cloud にサインイン	16
4.3.1.1 【AWS の場合】	16
4.3.1.2 【Azure の場合】	18
4.4 プロジェクトの準備	23
4.4.1 プロジェクトの選択	23
4.4.1.1 【AWS の場合】	23
4.4.1.2 【Azure の場合】	35
4.4.1.3 スマート・コンフィグレータのコンポーネントのダウンロード	43
4.4.2 プロビジョニングの選択	44
4.5 IoT デバイスの管理	45

# 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

---

4.5.1 IoT デバイスの管理 .....	45
4.5.2 初期ファームウェアの作成 .....	47
4.5.2.1 ツールチェーンのバージョンの確認 .....	47
4.5.2.2 初期ファームウェアの作成 .....	48
4.5.2.3 サンプルプロジェクトのファームウェア .....	51
4.5.2.4 コード署名証明書について .....	52
4.5.3 IoT デバイスに書き込み .....	54
4.6 OTA .....	57
4.6.1 更新ファームウェアの作成 .....	57
4.6.2 OTA の実行と状況確認 .....	59
5. ヘルプ機能 .....	63
改訂記録 .....	64

## 1. 概要

### 1.1 システム概要

QE for OTA は、統合開発環境 e<sup>2</sup> studio 上で動作するソリューション・ツールキットのひとつで、クラウドサービス AWS や Microsoft Azure を利用した OTA (Over the Air) を簡単に適用していただくための開発支援ツールです。ワークフローに従って操作することで、クラウド関連情報の入手から OTA を実施するに必要なクラウドシステムへの登録、MCU へのセキュリティ情報の組み込み、OTA の実行ができます。

### 1.2 QE for OTA の構成

QE for OTA は、クラウドを利用した OTA の導入のガイドを行う OTA メイン(QE) ビューと、ファームウェアの作成や OTA の設定、動作状態の表示を行う OTA IoT デバイス管理(QE) ビューで構成されています。

#### 1.2.1 OTA メイン(QE) ビュー

このビューは、QE for OTA のメインビューです。クラウドを利用した OTA の導入を行うまでのワークフローをガイドします。



図 1-1 OTA メイン(QE) ビュー

- Cloud の設定  
クラウドへのサインインの設定をします。  
Azure 使用時は OTA に使用するクラウドのリソースを設定します。
- プロジェクトの準備  
OTA プロジェクトを作成します。
- IoT デバイスの管理  
IoT デバイスの追加、削除、情報の閲覧ができます。また、初期ファームウェアを作成し、IoT デバイスに書き込みます。
- OTA  
更新用ファームウェアを作成し、OTA を実行します。

### 1.2.2 OTA IoT デバイス管理(QE) ビュー

このビューは IoT デバイス、初期ファームウェア、更新ファームウェア、OTA、ファームウェアログの 5 つの機能で構成されています。それぞれの機能はビュー右側のタブで表示の切り替えができます。

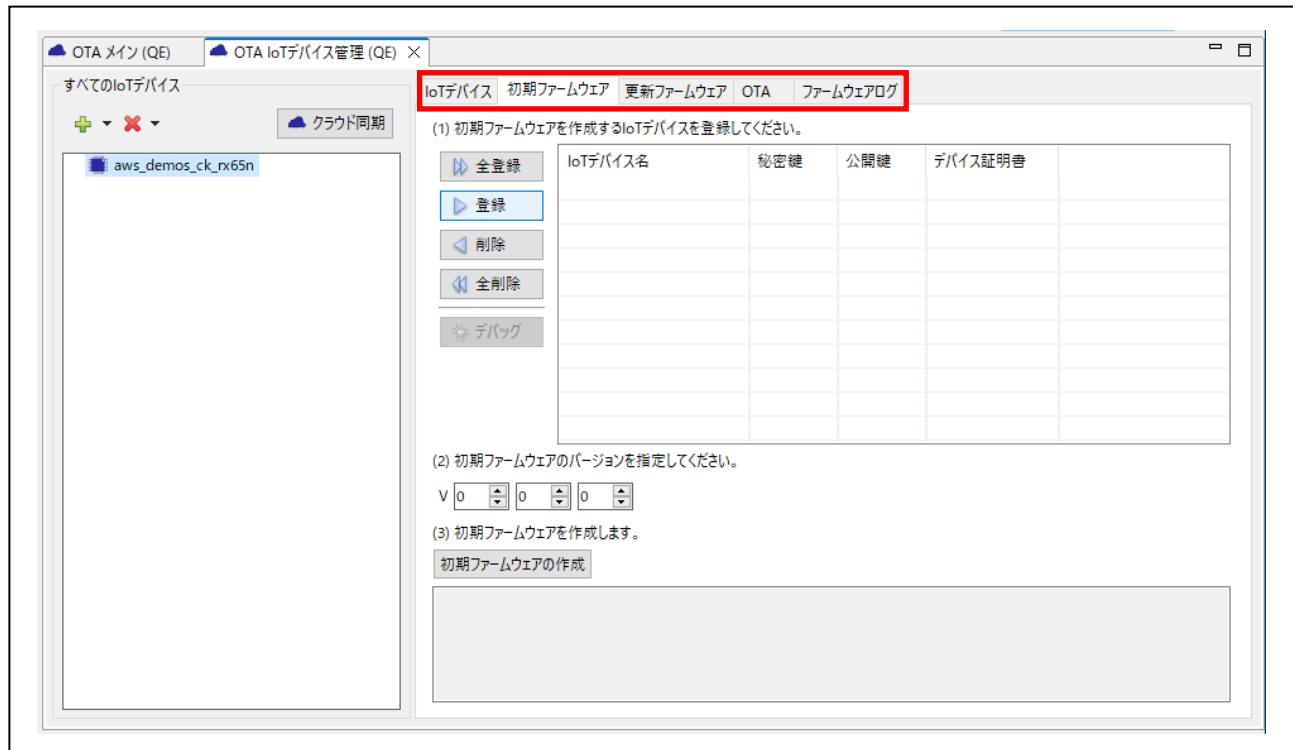


図 1-2 OTA IoT デバイス管理(QE) ビュー

- IoT デバイス  
IoT デバイス作成時に作成されたデバイス証明書、セキュリティキーなどの IoT デバイス情報が閲覧できます。
- 初期ファームウェア  
IoT デバイスごとの情報をソースコードに埋め込み、初期ファームウェアを作成できます。
- 更新ファームウェア  
IoT デバイスごとの情報をソースコードに埋め込み、更新ファームウェアを作成できます。
- OTA  
OTA を実行します。
- ファームウェアログ  
ターゲットボードの動作ログが閲覧できます。

### 1.3 動作環境

本アプリケーションノートの動作環境は以下の通りです。

表 1-1 動作環境

項目	内容
PC OS	Windows 10、Windows 11 (いずれも 64 ビット版)
統合開発環境(IDE)	Renesas e <sup>2</sup> studio 2023-04 以降
ツールチェーン	CC-RX V3.04 以降 / GCC for Renesas RX 8.3.0.202104 以降
QE	QE for OTA V1.1.0 以降
ターゲットデバイス	RX65N
評価ボード	CK-RX65N (キット同梱の RYZ014A PMOD を含む)
SIM カード	セルラーLTE 通信に使用できる micro SIM(CK-RX65N に同梱)
RX Driver Package (RDP)	AWS : V1.34 / Azure : V1.37 ※注 1
OpenSSL	V3.0.4 Light 以降 ※注 2
Renesas Flash Programmer	V3.09 以降 ※注 2
FreeRTOS	v202107.00-rx-1.0.1
Azure RTOS	6.2.1_rel-rx-1.0.1
Azure CLI	V2.49.0 以降 ※注 2

- 【注】 1. 本アプリケーションノートで使用する RTOS で使用しているバージョンです。使用する RTOS のバージョンに対応したものをインストールする必要があります。  
 2. e<sup>2</sup> studio や QE には付属しません。別途インストールが必要となります。

本アプリケーションノートでは、AWS を使用したサンプルプロジェクトの動作には、RYZ014A ボードを使用したセルラー通信を使用します。Azure を使用したサンプルプロジェクトの動作には、Ethernet を使用します。

セルラー通信を使用する場合の機器の接続、構成は以下の通りです。RYZ014A ボードの電源(下図中の "USB Cable - Power Supply")は、PC の USB ポートからの給電・コンセント接続の USB 電源どちらも使用可能です。

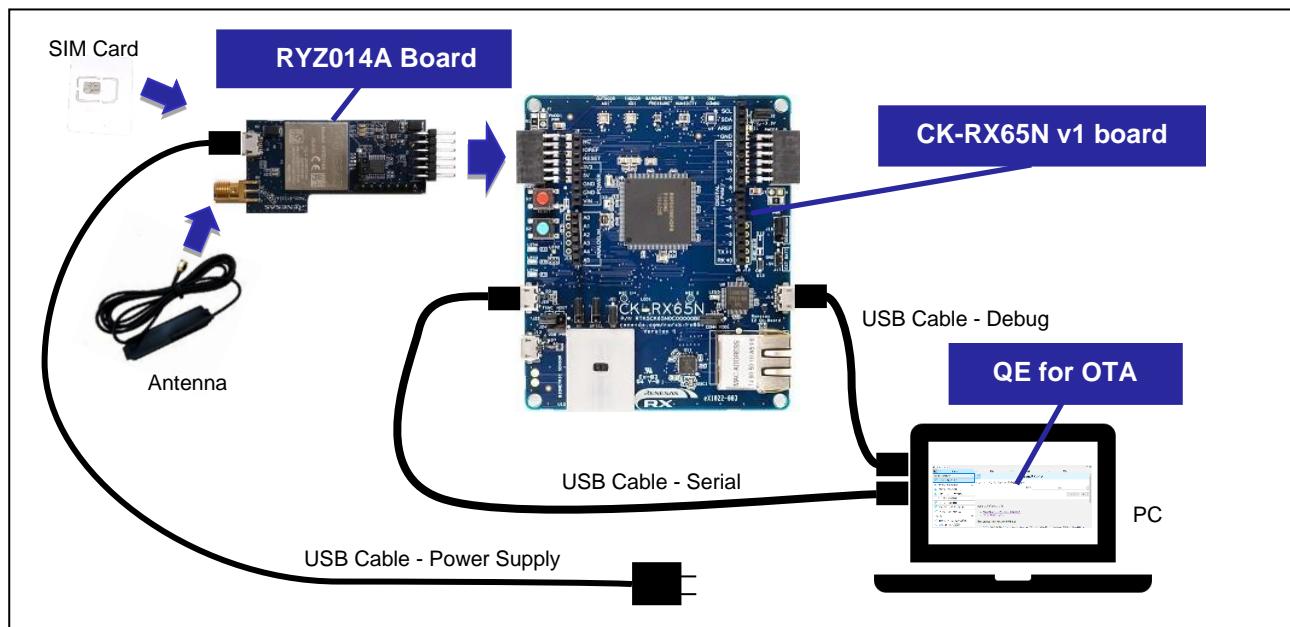


図 1-3 動作環境 接続図 セルラー通信時

Ethernet を使用する場合の機器の接続、構成は以下の通りです。CK-RX65N ボードの Ethernet ポートにインターネットに接続可能な LAN ケーブルを接続してください。

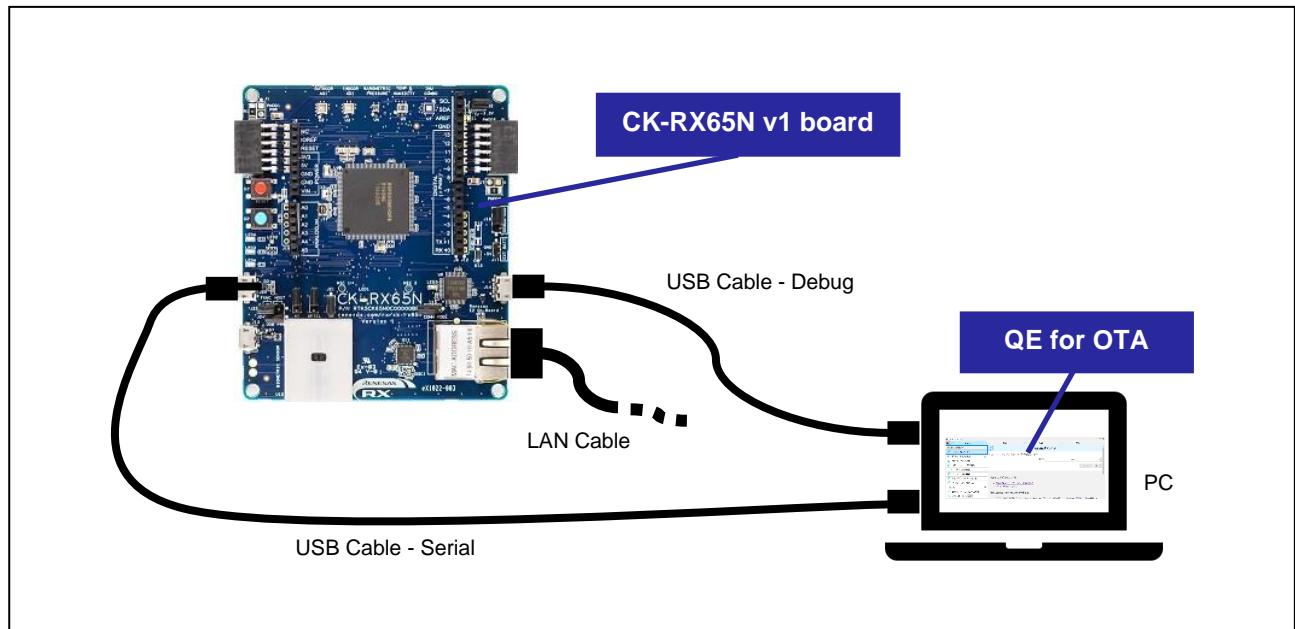


図 1-4 動作環境 接続図 Ethernet 通信時

### 1.4 参考資料

- R01UH0590  
RX65N グループ、RX651 グループ ユーザーズマニュアル ハードウェア編
- R20UT5100  
RX65N Group Cloud Kit for RX65N Microcontroller Group CK-RX65N v1 User's Manual
- R01AN5549  
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
- R01AN6357  
Microsoft Azure ADU 環境構築方法
- R01AN5548  
ルネサス MCU におけるファームウェアアップデートの設計方針
- R19DS0111  
RYZ014A LTE カテゴリ M1 モジュールデータシート

## 2. 事前準備

必要な準備は以下の通りです。

- e<sup>2</sup> studio のインストール
- e<sup>2</sup> studio 以下のツールを適用
  - ツールチェーン
  - QE for OTA
- Renesas Flash Programmer のインストール
- 評価ボードの接続
- AWS / Azure アカウントの作成
- OpenSSL のインストール
- Azure CLI のインストール (Azure 使用時)

本章では、QE for OTA のインストールと AWS / Azure のアカウントの作成について説明します。

OpenSSL のインストールについては OpenSSL の [Web サイト](#) を参照してください。

## 2.1 QE for OTA のインストール

### 2.1.1 Renesas Software Installer を使用する場合

e<sup>2</sup> studio の Renesas Software Installer から QE をインストールする方法を説明します。

e<sup>2</sup> studio のメニューから Renesas Software Installer を選択します。Renesas Software Installer のダイアログが表示されたら、Renesas QE を選択し [次へ(N)>] ボタンを押下します。

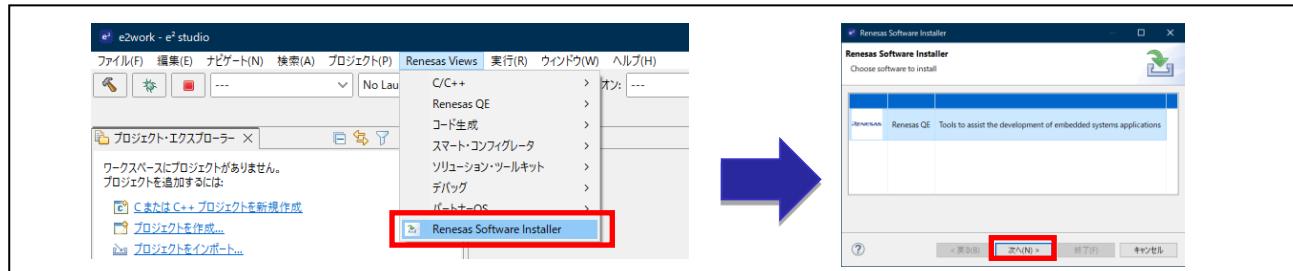


図 2-1 Renesas Software Installer

「拡張をインストール」が表示されたら、QE for OTA にチェックを入れ、[終了(F)]ボタンを押下します。

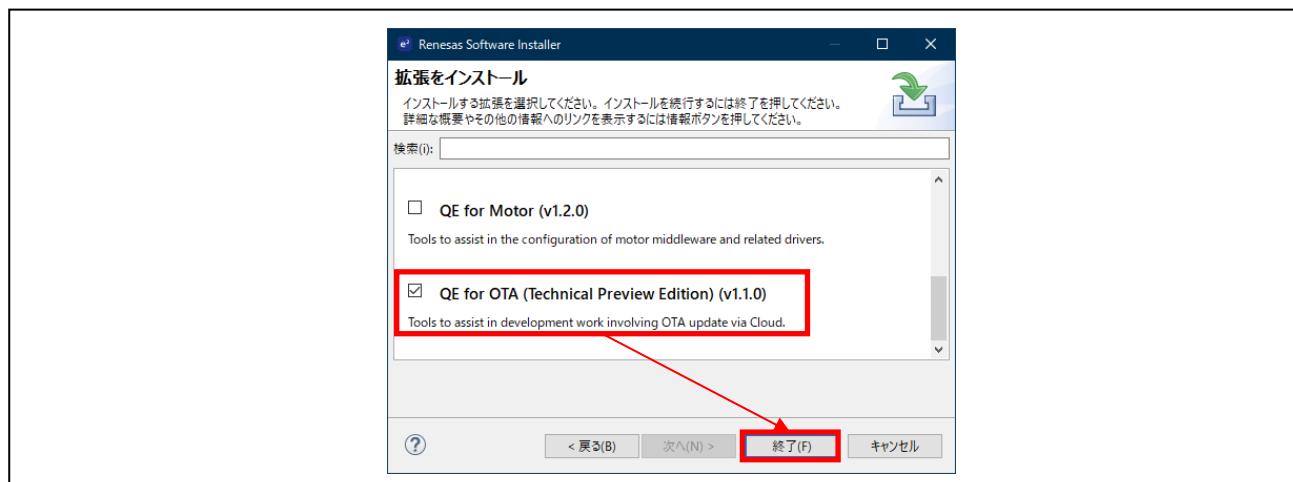


図 2-2 拡張をインストール

インストールしたい項目の [Renesas QE for OTA] にチェックを入れ、[次へ(N)>] ボタンを押下します。

インストール対象が [Renesas QE for OTA] となっていることを確認し、[終了(F)] ボタンを押下します。

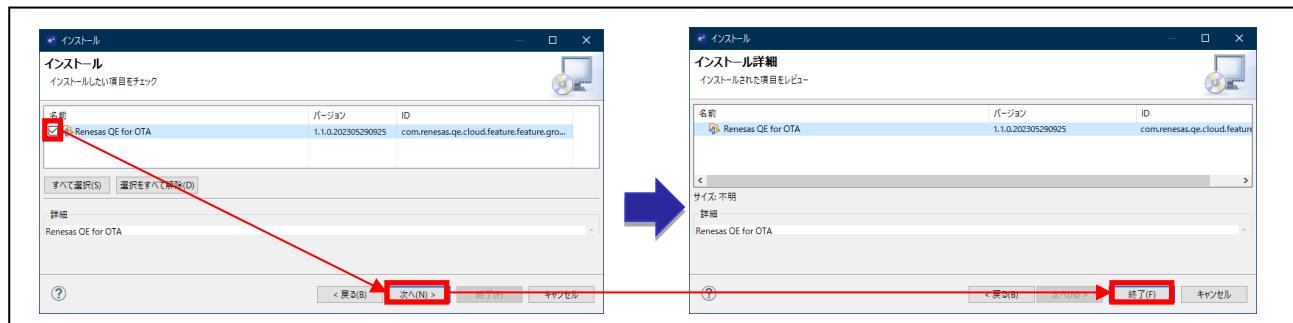


図 2-3 QE for OTA インストール

以降は「[2.1.3 インストール手順 \(共通\)](#)」を参照してください。

## 2.1.2 Web から QE をダウンロードしてインストールする場合

e<sup>2</sup> studio のメニューから [ヘルプ(H)]→[新規ソフトウェアのインストール...]メニューを選択し、[インストール]ダイアログを開きます。

その後、[追加(A)...]ボタンを押下し、[リポジトリを追加]ダイアログを開きます。

[アーカイブ(A)...]ボタンを押下し、開いたファイル選択ダイアログで、インストール用ファイル(zip ファイル)を選択し、[開く(O)]ボタンを押下し、[リポジトリを追加]ダイアログで、[追加(D)]ボタンを押下します。

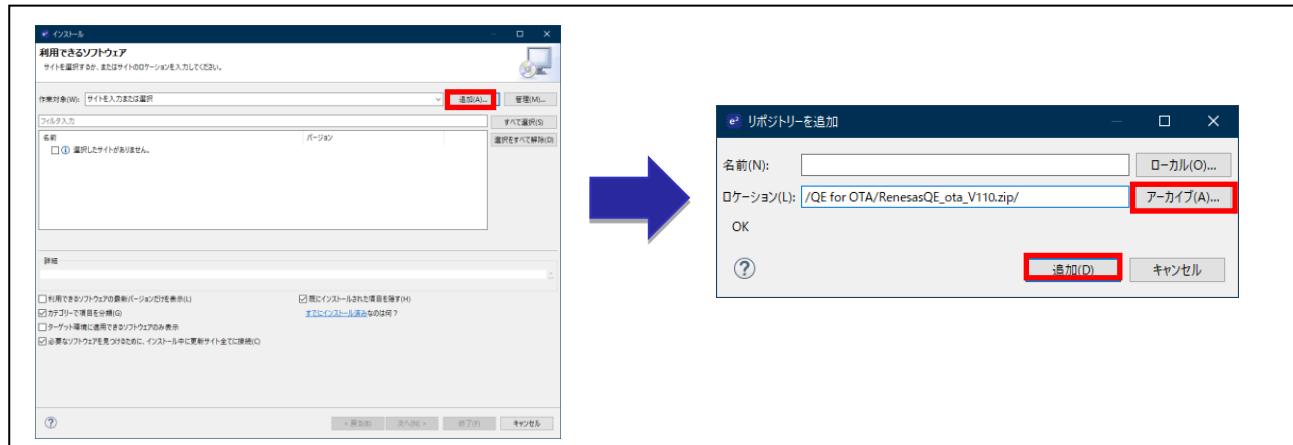


図 2-4 Web からダウンロードしてインストール 1

[インストール]ダイアログに表示された[Renesas QE]項目を展開し、表示された製品のチェックボックスをチェックし、[次へ(N)>]ボタンを押下します。この時、[必要なソフトウェアを見つけるために、インストール中に更新サイト全てに接続]チェックを外すことでインストール時間を短縮できます。

インストール対象を確認し、[終了(F)>]ボタンを押下します。

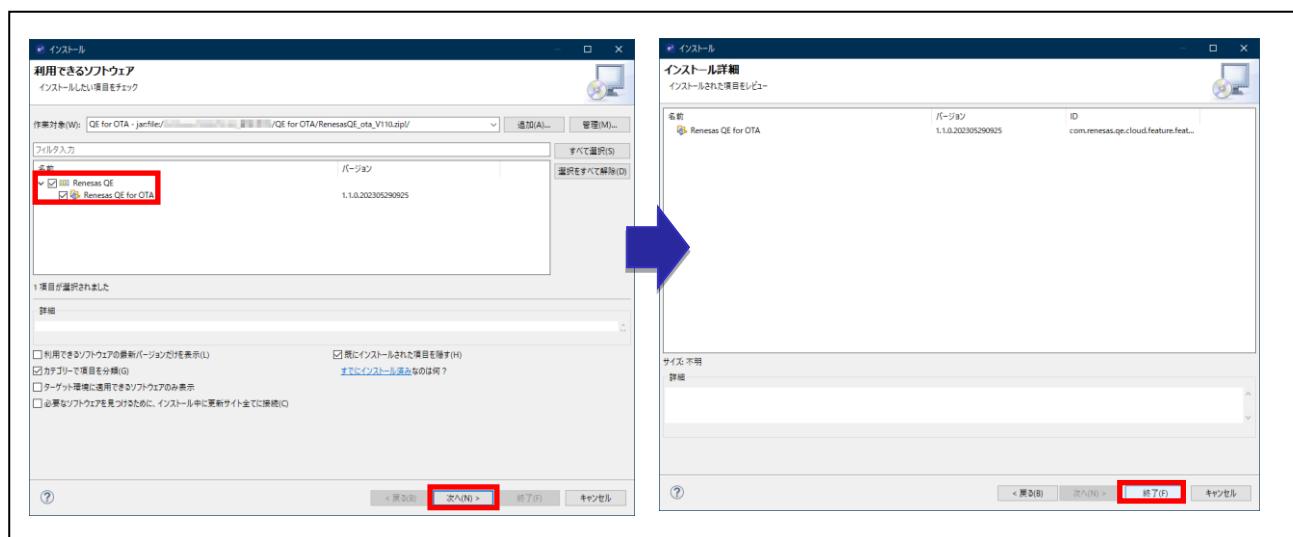


図 2-5 Web からダウンロードしてインストール 2

以降は「[2.1.3 インストール手順 \(共通\)](#)」を参照してください。

## 2.1.3 インストール手順 (共通)

以下のダイアログが表示された場合は、信頼するツールに設定すると、インストールが実行されます。

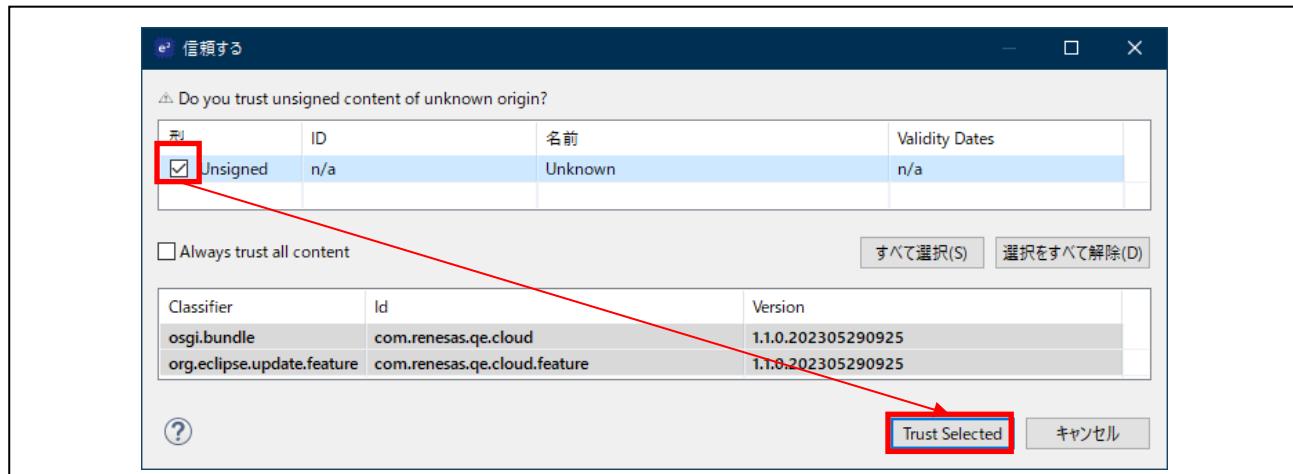


図 2-6 ツールの信頼性

QE for OTA のインストール状況は e<sup>2</sup> studio の下部に表示されます。e<sup>2</sup> studio の再起動の要求がありますので、再起動すればインストールは完了です。



図 2-7 インストール状況

## 2.2 Cloud アカウントの作成

AWS / Azure のアカウントの作成について説明します。詳細はそれぞれのガイドを参照してください。

### 2.2.1 【AWS の場合】

AWS を使用する場合の準備について説明します。

1. [AWS のウェブサイト](#)にアクセスして AWS のアカウントを作成します。
2. WEB ブラウザから AWS コンソールにサインインして、OTA を実行する IAM ユーザを作成します。作成する IAM ユーザには、IAM サービスのすべてにアクセスするための権限(IAMFullAccess)を付加してください。
3. AWS コンソールでセキュリティ認証情報の設定を開き、AWS IAM 認証情報の設定から、アクセスキーを作成してください。アクセスキーとシークレットアクセスキーが発行されます。

### 2.2.2 【Azure の場合】

Azure を使用する場合の準備について説明します。

1. [Azure のウェブサイト](#)にアクセスして Azure のアカウントを作成します。
2. WEB ブラウザから Azure アカウントにサインインして、サブスクリプション ID を確認します。
3. Microsoft の Web サイトから Azure CLI をダウンロードしてインストールします。  
Azure CLI のインストール後、QE for OTA の初回動作時にブラウザが立ち上がり、Azure へのサインインを求められます。アカウント情報を入力し、サインインしてください。

### 3. 開発手順

QE for OTA のワークフローに従って操作することで、初期ファームウェアの作成から OTA の実行までを行なうことができます。



図 3-1 OTA メイン(QE) ビュー

各項目・手順を下表に示します。

表 3-1 QE for OTA の項目と内容

項目	内容		章番号
Cloud の設定	Cloud にサインイン	サインインの設定とサインイン	<a href="#">4.3.1</a>
	Cloud のリソース設定	リソースの設定(Azure のみ)	<a href="#">4.3.1</a>
プロジェクトの準備	プロジェクトの選択	サンプルプロジェクトの作成やインポート	<a href="#">4.4.1</a>
	プロビジョニングの選択	プロビジョニングの確認	<a href="#">4.4.2</a>
IoT デバイスの管理	IoT デバイスの管理	IoT デバイスの登録や管理	<a href="#">4.5.1</a>
	初期ファームウェアの作成	初期ファームウェアの作成・ビルド	<a href="#">4.5.2</a>
	IoT デバイスに書き込み	初期ファームウェアの書き込み	<a href="#">4.5.3</a>
OTA	更新用ファームウェアの作成	OTA で使用するファームウェアの作成・ビルド	<a href="#">4.6.1</a>
	OTA の実行と状況確認	OTA の実行と状況の確認	<a href="#">4.6.2</a>

## 4. 動作確認例

本章では、QE for OTA を使用して OTA を行うまでの一連の流れを説明します。

AWS (FreeRTOS) と、Azure (Azure RTOS) で異なる部分がありますが、大きな差異がない場合は AWS の内容で記述しています。適宜読み替えてください。

以下の動作を行います。

- 初期ファームウェアを書き込み、デバッグでファームウェア V1.0.0 動作
- OTA 実行により更新ファームウェア書き込み後、バンクスワップによって V1.1.0 動作に切り替わる

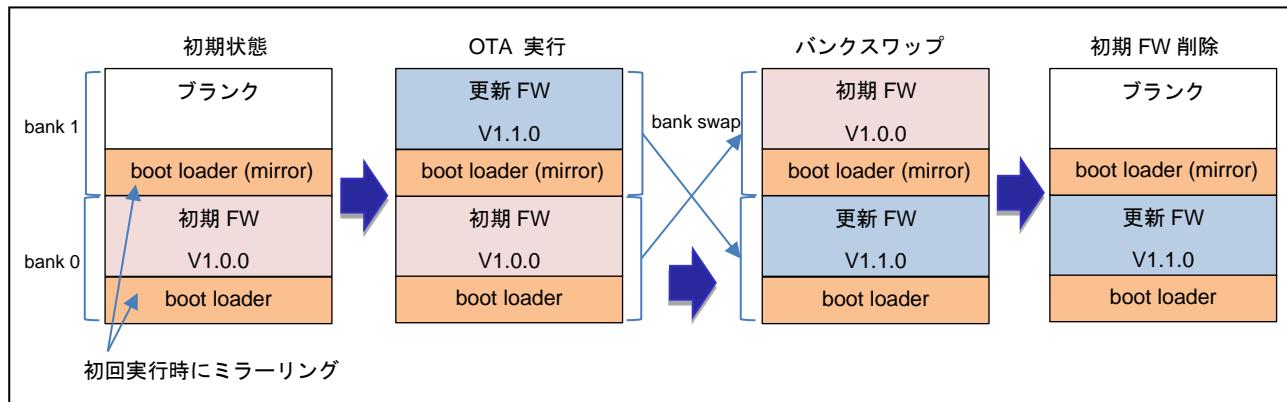


図 4-1 OTA イメージ

### 4.1 準備

e<sup>2</sup> studio を起動します。任意の場所に以下のフォルダを用意します。

- e<sup>2</sup> studio 用ワークスペースフォルダ
- RTOS モジュール・ダウンロードフォルダ
- プロジェクトフォルダ

本書では、ワークスペース及びプロジェクトフォルダを以下とします。以下に記載のないものは任意のフォルダとしています。

- C:\workspace (ワークスペースフォルダ)
- C:\workspace\freertos (AWS FreeRTOS 用)
- C:\workspace\azure\_demos (Azure RTOS ADU プロジェクト用)
- C:\workspace\azure\_boot\_loader (Azure RTOS ブートローダープロジェクト用)
- C:\afr (FreeRTOS モジュール・ダウンロードフォルダ)

### 4.2 QE for OTA 起動

下図のように、e<sup>2</sup> studio のメニューから[OTA メイン(QE)]を選択し、OTA メイン(QE)ビューを表示します。

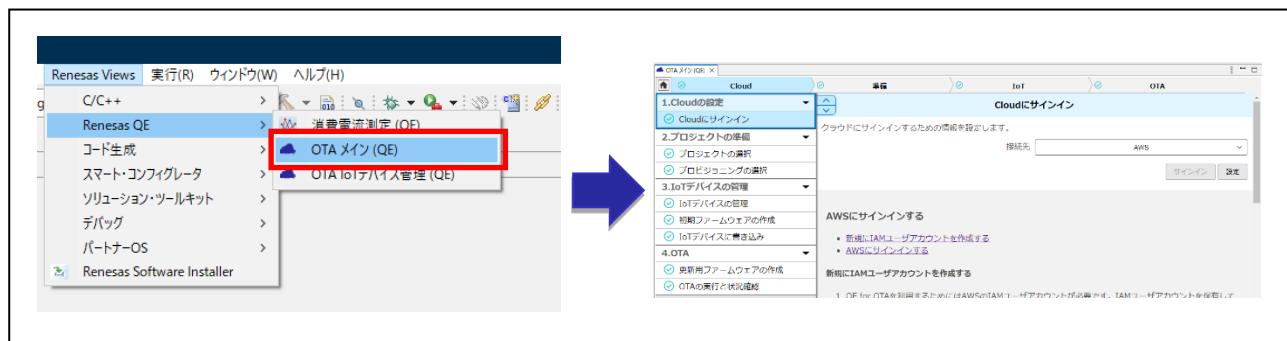


図 4-2 OTA メイン(QE)の選択

## 4.3 Cloud の設定

クラウドへのサインインを行います。Azure を使用する場合は、リソースの設定も合わせて行う必要があります。

### 4.3.1 Cloud にサインイン

OTA メイン(QE) ビューの左側に表示されているメニューから[Cloud にサインイン]を選択します。

接続先を選択し、[設定]を押下すると、サインインダイアログが開きます。



図 4-3 Cloud にサインイン

#### 4.3.1.1 【AWS の場合】

IAM ユーザの Access Key ID と Secret Access Key を入力して[Sign-in]を押下します。

[OTA に必要な権限を付加]ボタンの横に警告マークが表示される場合は、IAM ユーザに OTA を実行するのに必要なパーミッションが付加されていません。その場合は、[OTA に必要な権限を付加]ボタンを押下すると、自動的に必要な権限が付与されます。

Region は、AWS の各サービスで使用するリージョンを選択してください。アップロード先と現在のリージョンが異なる場合、OTA の実行に失敗します。同じリージョンを指定してください。

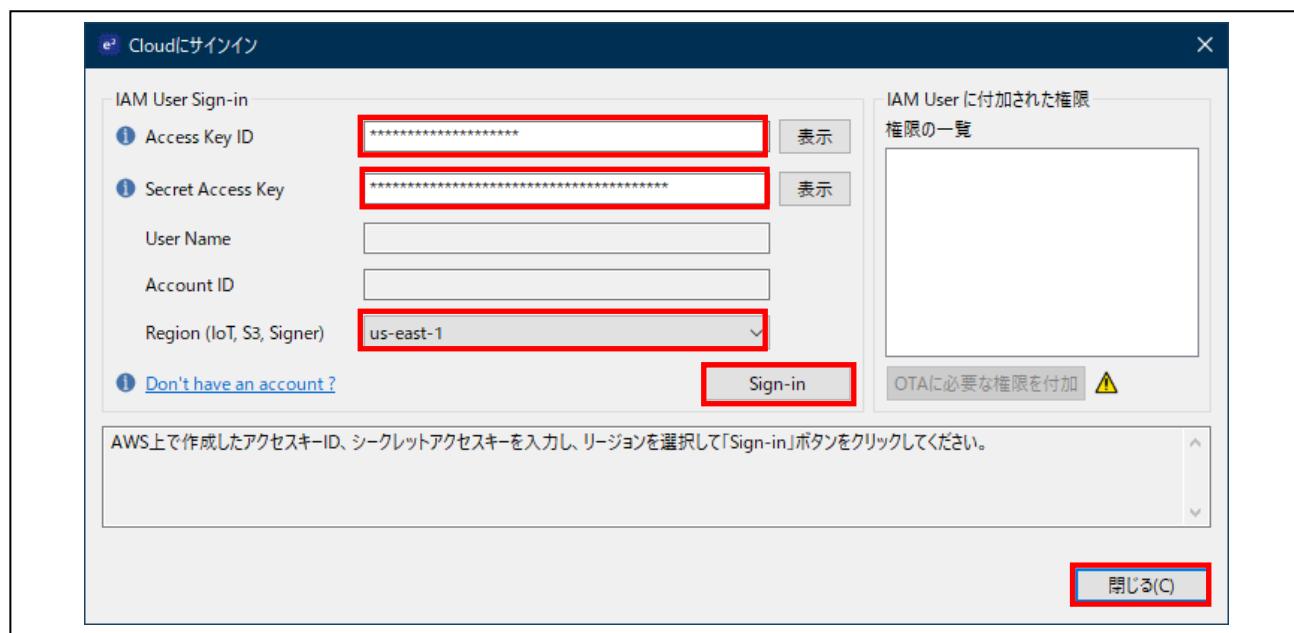


図 4-4 Cloud サインイン (AWS)

サインイン後、「クラウドにサインインしました。」のメッセージ表示を確認後、[閉じる]ボタンを押下してダイアログを閉じます。

## 4.3.1.2 【Azure の場合】

## (1) サインイン

## (a) アプリケーションを作成する場合

ApplicationName, Subscription ID を入力して、[Register Application]ボタンを押下します。以下の点に注意してください。

- Subscription の所有者権限が必要になります。
- 既に同じ Application Name で登録している場合は失敗します。
- Azure CLI がインストールされていない場合は失敗します。

Client ID, Client Secret, Tenant ID が設定されたら、[Sign-in]ボタンを押下します。

OTA を実行するのに必要な権限が付加されていない場合は、警告マークが表示されます。その場合は、[OTA に必要な権限を付加]ボタンを押下すると、自動的に必要な権限が付与されます。

サインイン後、「クラウドにサインインしました。」のメッセージ表示を確認後、[閉じる(C)]ボタンを押下し、ダイアログを閉じます。

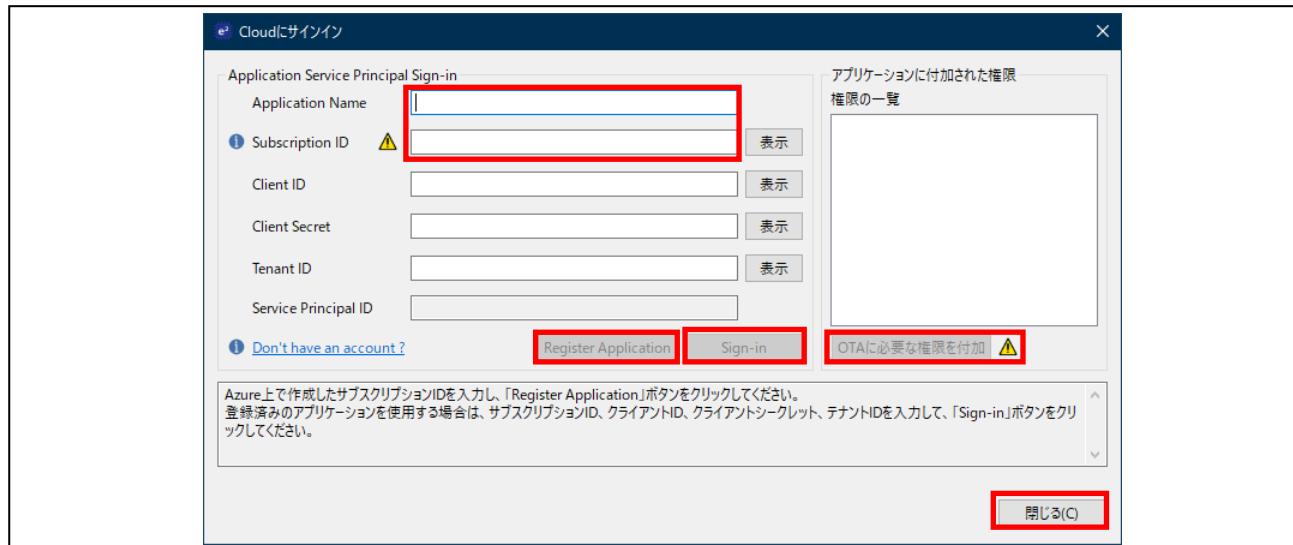


図 4-5 Cloud にサインイン (Azure)

## (b) 登録済みのアプリケーションを使用する場合

Subscription ID, Client ID, Client Secret, Tenant ID を入力して[Sign-in]ボタンを押下します。

サインイン後、[閉じる(C)]ボタンでダイアログを閉じます。

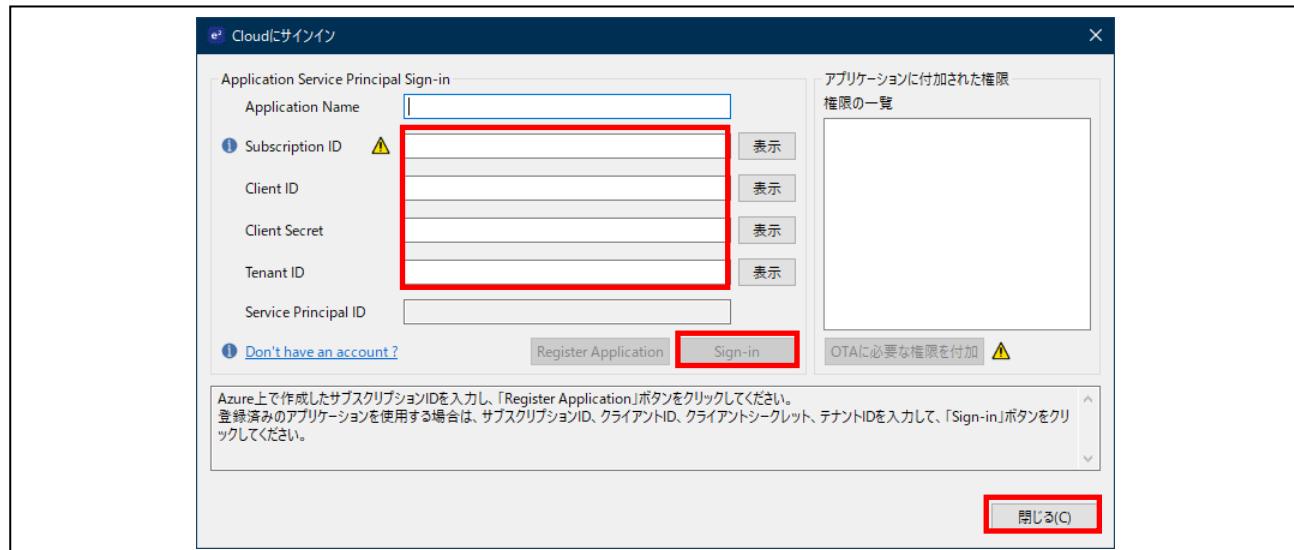


図 4-6 Cloud にサインイン 登録済みのアプリケーションを使用する場合 (Azure)

## (2) クラウドのリソース設定

Azure を使用する場合、クラウドのリソース設定を行う必要があります。

OTA メイン(QE) ビューの左側に表示されているメニューから[Cloud のリソース設定]を選択します。

このとき、Azure 側に QE が作成するリソースグループ "qe\_ota\_resource\_group" が存在していないければ、[設定]ボタンの横に[新規作成]ボタンが表示されますので、押下します。[新規作成]ボタンが無い場合は[設定]ボタンを押下します。

既に"qe\_ota\_resource\_group" が作成されている場合は、それを選択してください。



図 4-7 Cloud のリソース設定 (Azure)

## 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

リソースの設定ダイアログが開きます。リソースグループ "qe\_ota\_resource\_group" が無い場合、[すべて作成]ボタンを押下し、新規作成を行ってください。

個別に設定する場合は、各リソースの項目で設定してください。

現在設定されている情報は、ダイアログ中央の情報欄に表示されています。設定に不足がないことを確認してください。

また、各リソースに警告アイコンが表示されている場合、[OTA に必要な権限を付加]ボタンを押下して、権限を付加することができます。

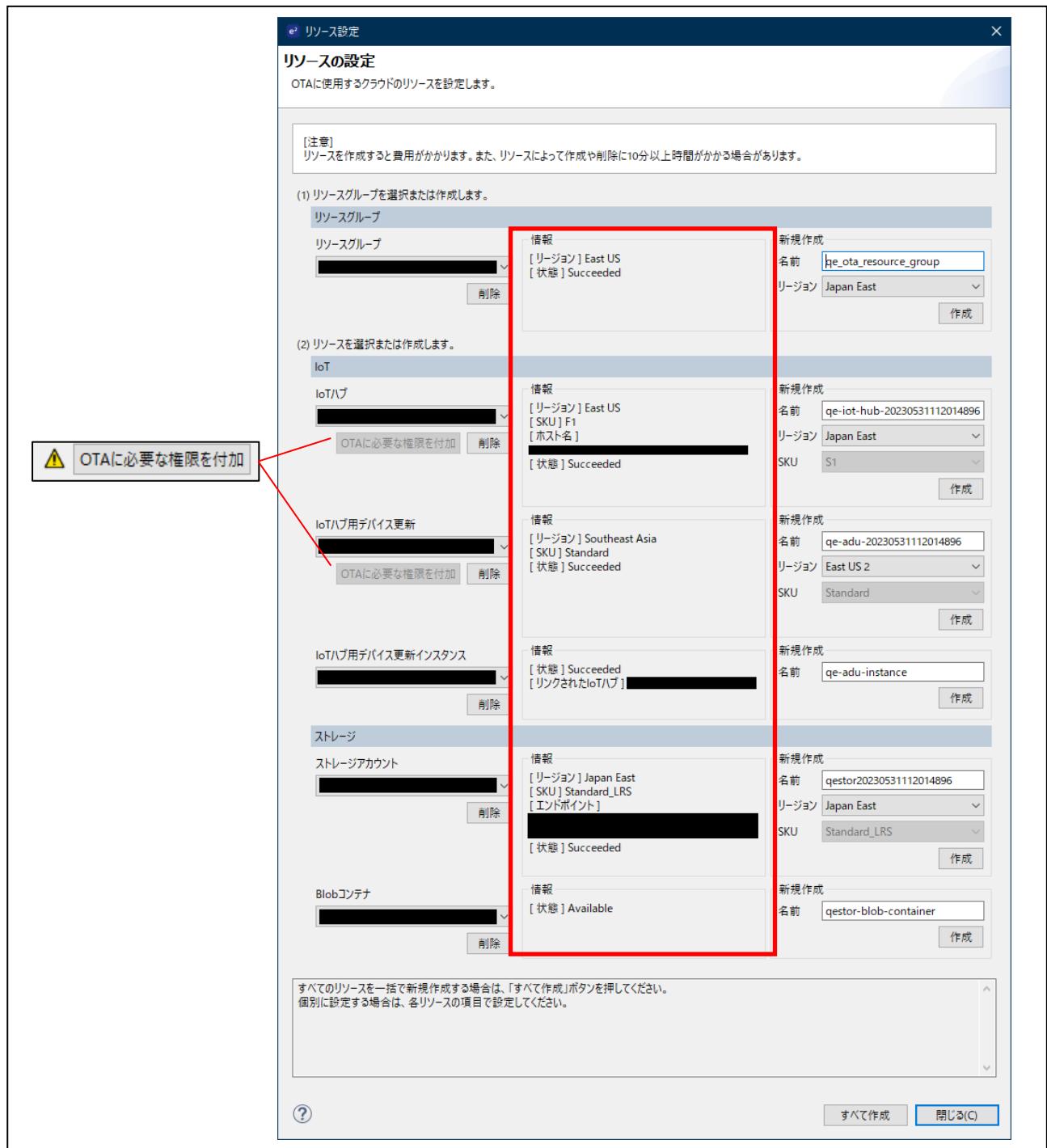


図 4-8 リソースの設定 (Azure)

## 4.4 プロジェクトの準備

### 4.4.1 プロジェクトの選択

OTA 用のサンプルプロジェクトの準備を行います。[AWS の場合](#) と [Azure の場合](#) で手順が異なります。

#### 4.4.1.1 【AWS の場合】

##### (1) プロジェクトのインポート

OTA メイン(QE) ビューの左側に表示されているメニューから[プロジェクトの選択]を選択し、[インポート]ボタンを押下します。既存のプロジェクトを対象とする場合は、インポート手順をスキップして、ドロップダウンからプロジェクトを選択してください。



図 4-9 プロジェクトの選択 (AWS)

[参照...]ボタンを押下し、RTOS をコピーするフォルダを指定します。(4.1 で用意したフォルダ)

次に、RTOS のバージョンを指定します。

このとき、バージョンがリストに表示される場合は、設定内容を確認し、[次へ(N)>]を押下します(図 4-12 FreeRTOS バージョン選択 次へ を参照)。バージョンがリストに表示されない場合は、[他のバージョンを確認...]ボタンを押下します。

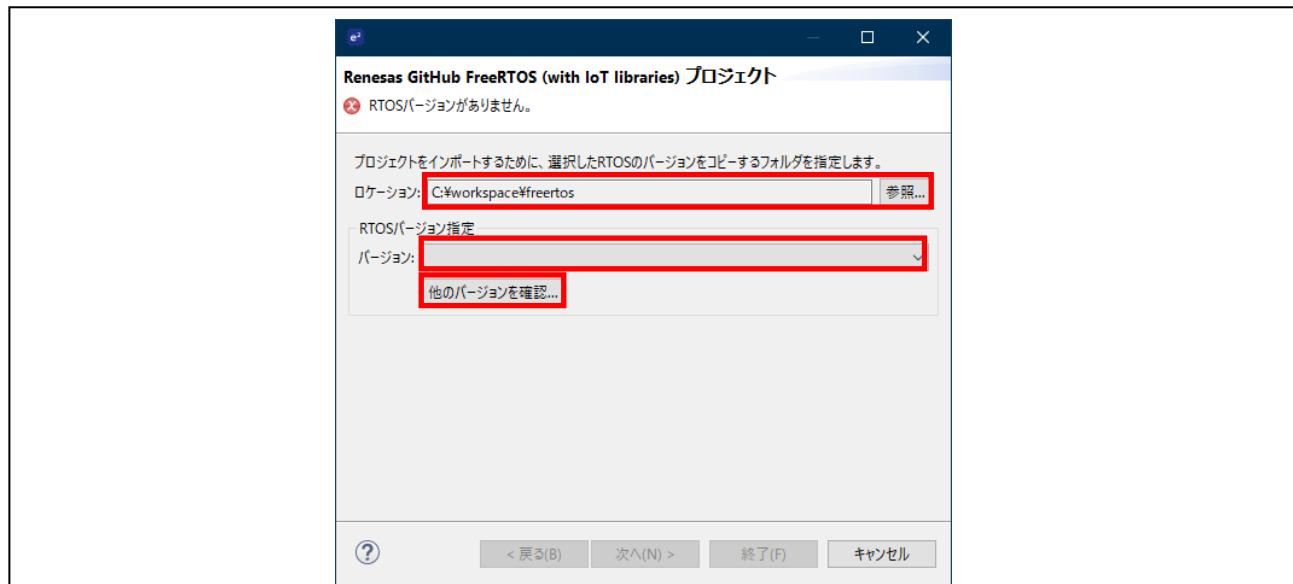


図 4-10 FreeRTOS バージョン選択

## 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

開いたダイアログで、ダウンロードするバージョンとダウンロード先となる[モジュール・フォルダー・パス] (4.1 で用意したフォルダ) を指定して[ダウンロード]ボタンを押下すると、免責事項ダイアログが表示されます。内容を確認し、[Accept]ボタンを押下します。

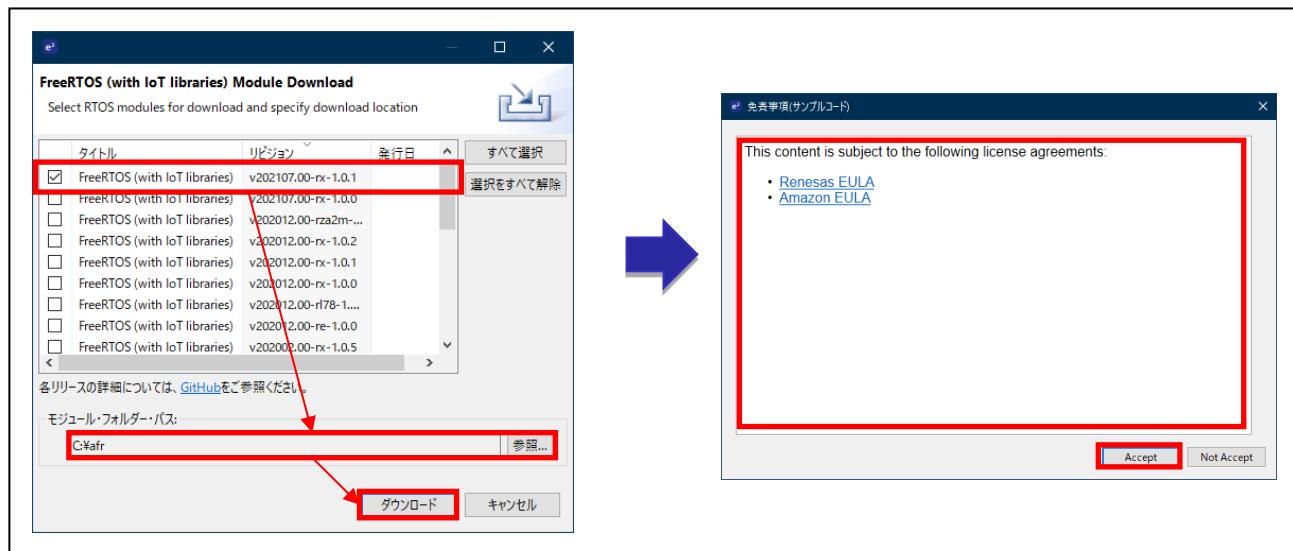


図 4-11 FreeRTOS モジュールダウンロード

使用するリビジョンの詳細や選定については GitHub([Releases · renesas/amazon-freertos \(github.com\)](https://github.com/renesas/amazon-freertos/releases))を参照してください。

設定内容を確認し、[次へ(N)>]を押下します。

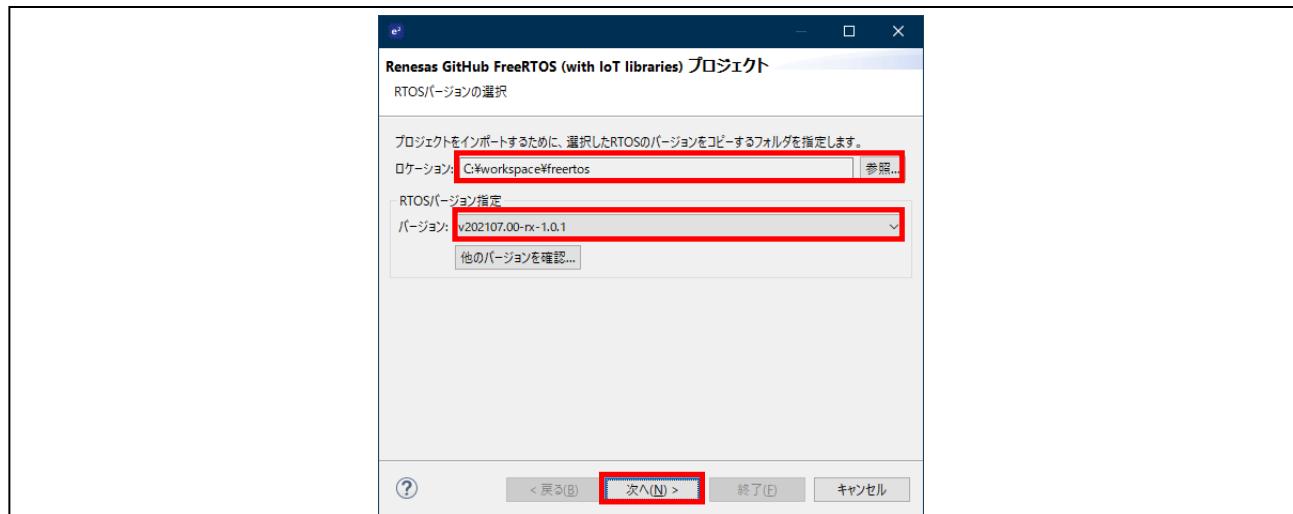


図 4-12 FreeRTOS バージョン選択 次へ

インポートするプロジェクトを一覧から選択します。環境により選択するプロジェクトが異なります。適切なプロジェクトを選択してインポートしてください。

下図は、CK-RX65N ボードで RYZ014A ボードによるセルラー通信を使用し、コンパイラに CC-RX を選択した例です。

aws\_demos と boot\_loader から、動作させる環境に合わせて 1 つずつ選択し、[終了(F)]を押下します。

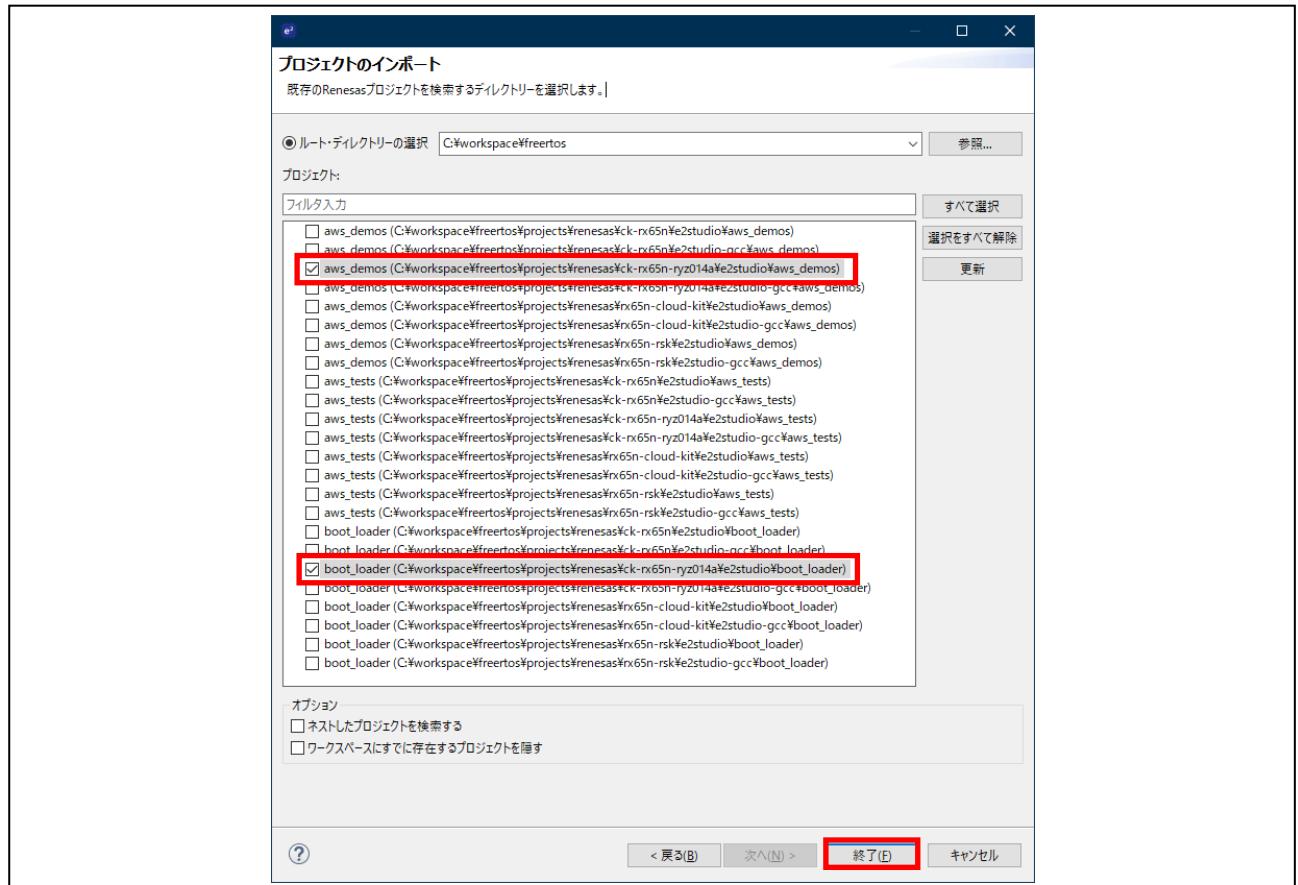


図 4-13 インポートするプロジェクトの選択

以下は、使用するプロジェクトの選択例です。

- RYZ014A を使用する場合(CC-RX コンパイラ):  
aws\_demos (<FreeRTOS コピー先>¥projects¥renesas¥ck-rx65n-ryz014a¥e2studio¥aws\_demos)  
boot\_loader (<FreeRTOS コピー先>¥projects¥renesas¥ck-rx65n-ryz014a¥e2studio¥boot\_loader)
- RYZ014A を使用する場合(GCC コンパイラ):  
aws\_demos (<FreeRTOS コピー先>¥projects¥renesas¥ck-rx65n-ryz014a¥e2studio-gcc¥aws\_demos)  
boot\_loader (<FreeRTOS コピー先>¥projects¥renesas¥ck-rx65n-ryz014a¥e2studio-gcc¥boot\_loader)
- Ethernet を使用する場合(CC-RX コンパイラ):  
aws\_demos (<FreeRTOS コピー先>¥projects¥renesas¥ck-rx65n¥e2studio¥aws\_demos)  
boot\_loader (<FreeRTOS コピー先>¥projects¥renesas¥ck-rx65n¥e2studio¥boot\_loader)
- Ethernet を使用する場合(GCC コンパイラ):  
aws\_demos (<FreeRTOS コピー先>¥projects¥renesas¥ck-rx65n¥e2studio-gcc¥aws\_demos)  
boot\_loader (<FreeRTOS コピー先>¥projects¥renesas¥ck-rx65n¥e2studio-gcc¥boot\_loader)

インポート後、下図のようにプロジェクト情報が自動的に選択されます。



図 4-14 AWS プロジェクトのインポート後

boot\_loader プロジェクト用の領域として 256KB を確保するため、aws\_demos プロジェクトはセクション設定の変更が必要です。

(a) セクションの設定 CC-RX の場合

プロジェクト・エクスプローラーから aws\_demos プロジェクトを選択し、右クリックからプロパティを開きます。[設定]を選択し、下図のボタンからセクション設定を開きます。

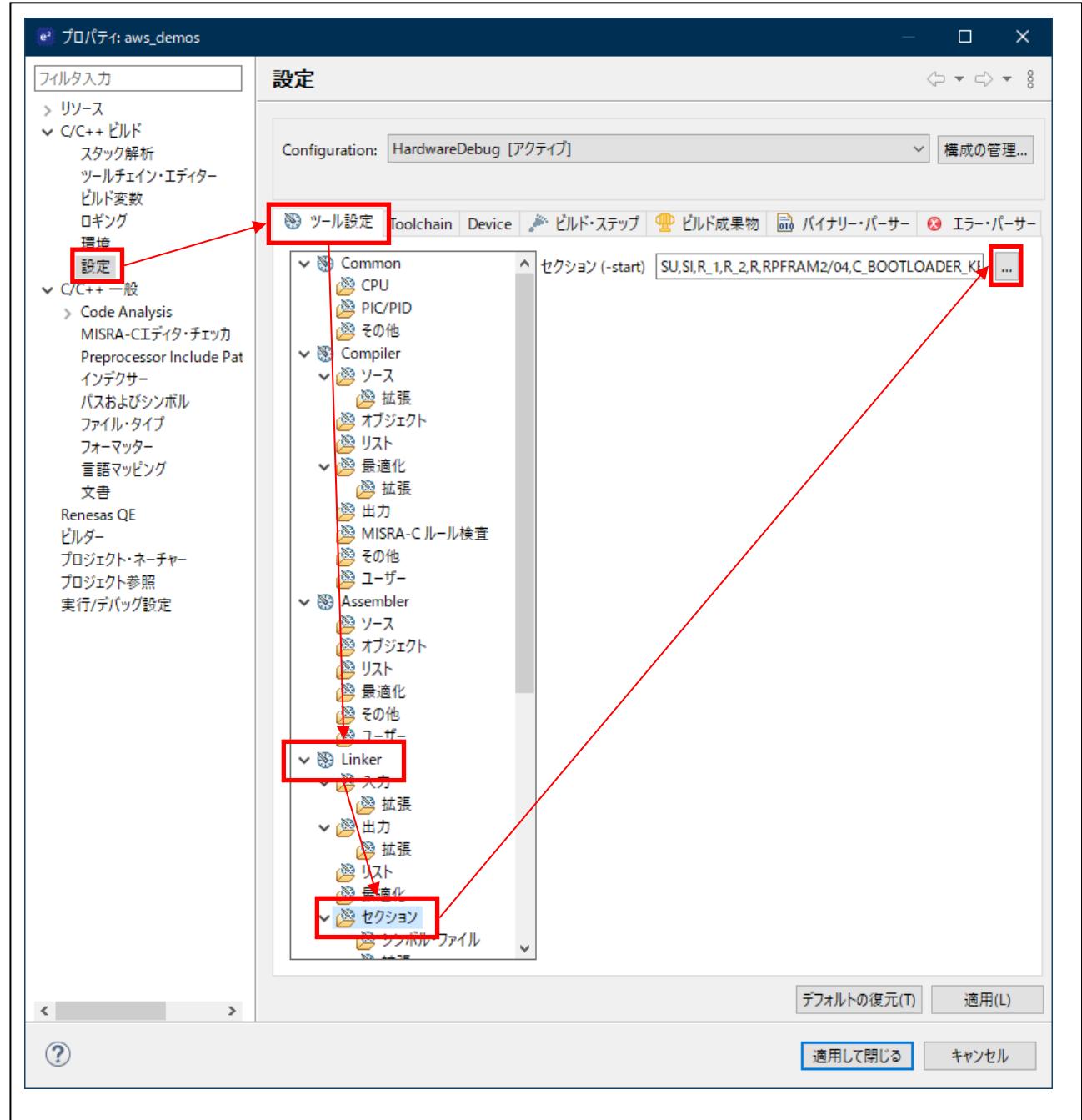


図 4-15 aws\_demos プロジェクト セクション設定 1 (CC-RX)

下図赤枠部の "EXCEPTVECT" および "RESETVECT" のアドレスを変更して[OK]ボタンを押下します。



図 4-16 aws\_demos プロジェクト セクション設定 2 (CC-RX)

[適用]ボタンもしくは[適用して閉じる]ボタンを押下し、変更を適用します。

## (b) セクションの設定 GCC の場合

GCC の場合は、プロジェクト・エクスプローラーから aws\_demos プロジェクトにある [linker\_script.ld] ファイルを編集します。

上記ファイルをダブルクリックしてリンクスクリプトエディタで開き、[linker\_script.ld] タブを選択します。その後、下図のように、".exvectors" および ".fvecsors" のアドレスを変更します。

```

1 *linker_script.ld X
2
3 MEMORY
4 {
5     RAM : ORIGIN = 0x4, LENGTH = 0x3ffffc
6     RAM2 : ORIGIN = 0x00800000, LENGTH = 393216
7     DATAFLASH (!rx) : ORIGIN = 0x100000, LENGTH = 32768
8     ROM : ORIGIN = 0xFFFF0000, LENGTH = 1048576
9     OFS : ORIGIN = 0xFE7F5D00, LENGTH = 128
10
11 SECTIONS
12 {
13     .exvectors 0xFFFFBFF80 : AT(0xFFFFBFF80)
14     {
15         "_exvectors_start" = .;
16         KEEP(*(.exvectors))
17         "_exvectors_end" = .;
18     } >ROM
19     .fvecsors 0xFFFFFBFFC : AT(0xFFFFFBFFC)
20     {
21         KEEP(*(.fvecsors))
22     } >ROM
23     .text 0xFFFF35000 : AT(0xFFFF35000)
24     {
25         ...
26     }
27 }

```

図 4-17 aws\_demos プロジェクト セクション設定 (GCC)

尚、セクション編集時、リンクスクリプトエディタでエラーが表示される箇所がありますが、無視してください。

## (2) デバイスの変更(FreeRTOS v202107.00-rx-1.0.x を使用する場合)

デバイスの変更を行います。この手順は aws\_demos プロジェクトと boot\_loader プロジェクトどちらにも必要です。

aws\_demos プロジェクトを使用して説明します。

プロジェクト・エクスプローラーで、[aws\_demos/aws\_demos.scfg] をダブルクリックしてスマート・コンフィグレータを開きます。

[ボード]タブを選択し、ボードの選択ダイアログを開きます。

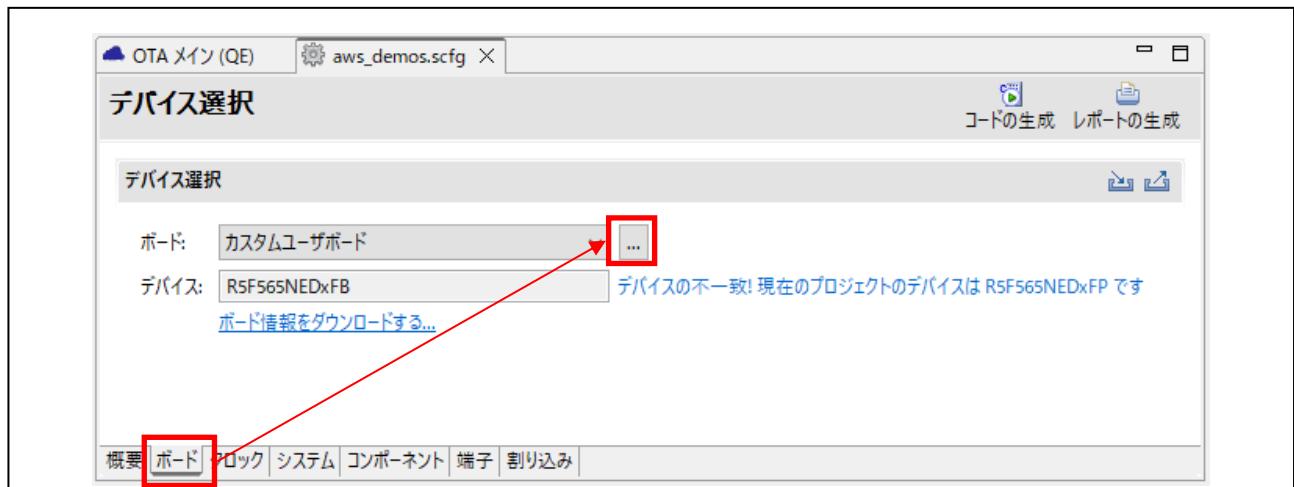


図 4-18 デバイスの変更 1

ターゲットデバイスの参照ボタンを押します。

本書の構成を使用する場合、"R5F565NEDxFB\_DUAL" を選択し、[OK]を押下します。

[次へ(N) >]ボタンを押下します。

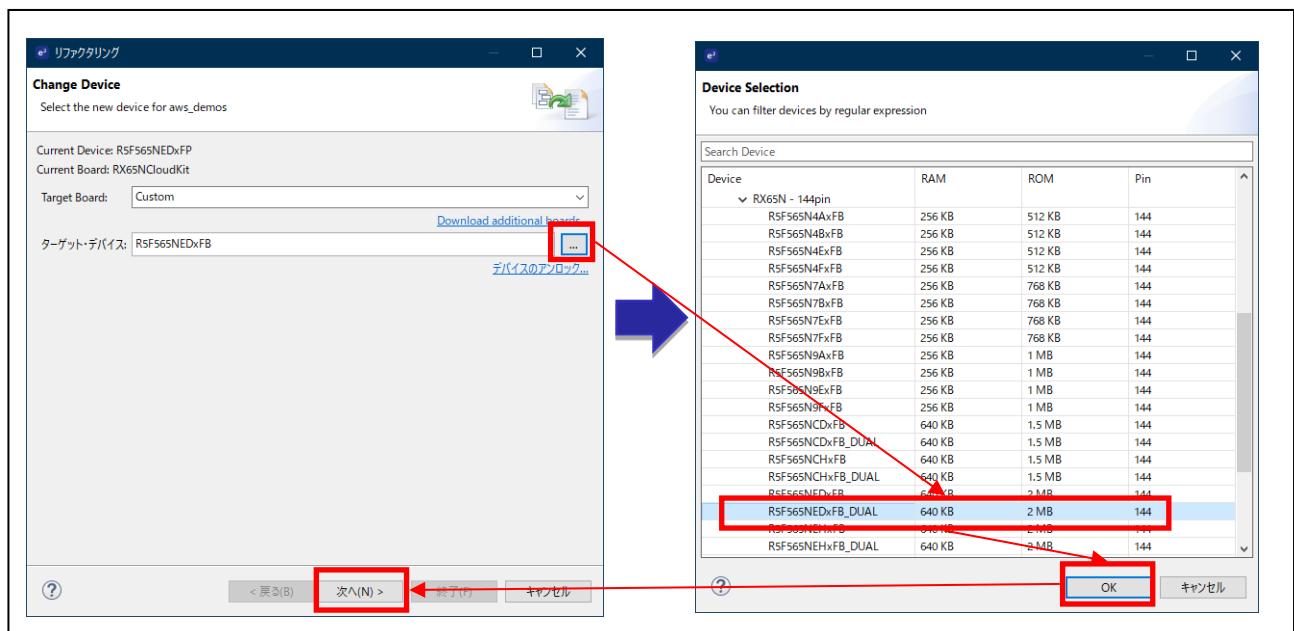


図 4-19 デバイスの変更 2

次のようなダイアログが表示されたら、[次へ(N)>]を押下します。

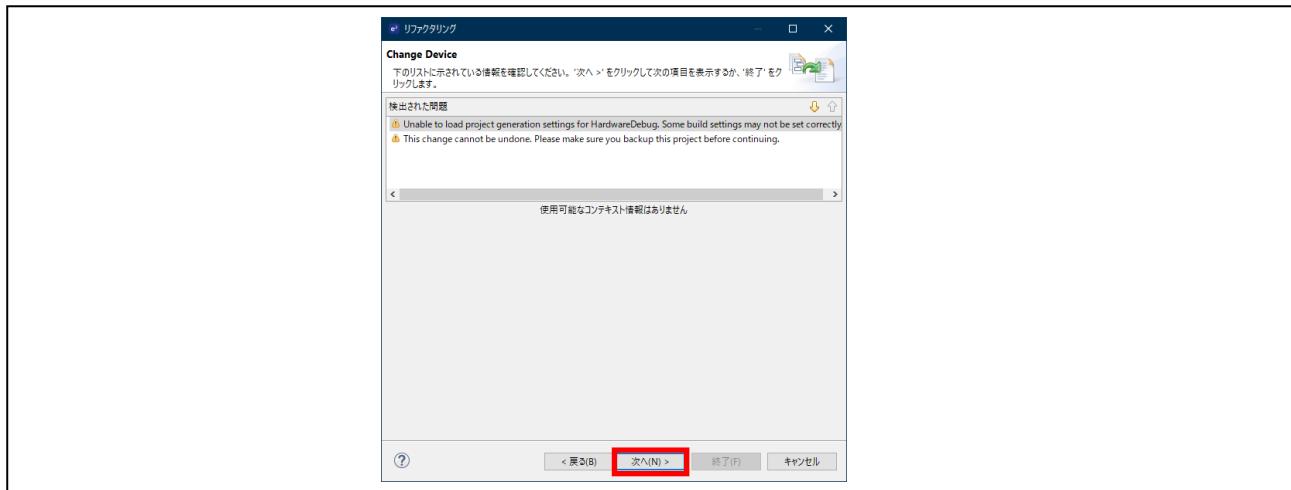


図 4-20 デバイスの変更 3

(a) デバイスの変更 CC-RX の場合

[Build Settings]>[HardwareDebug]>[Toolchain Settings]を選択します。

[ROM から RAM へマップするセクション(-rom)]と[セクション(-start)]のチェックを外します。

(RYZ014A セルラー用 boot\_loader プロジェクトの場合、[セクション(-start)]がありません。)

[終了(F)]を押下します。

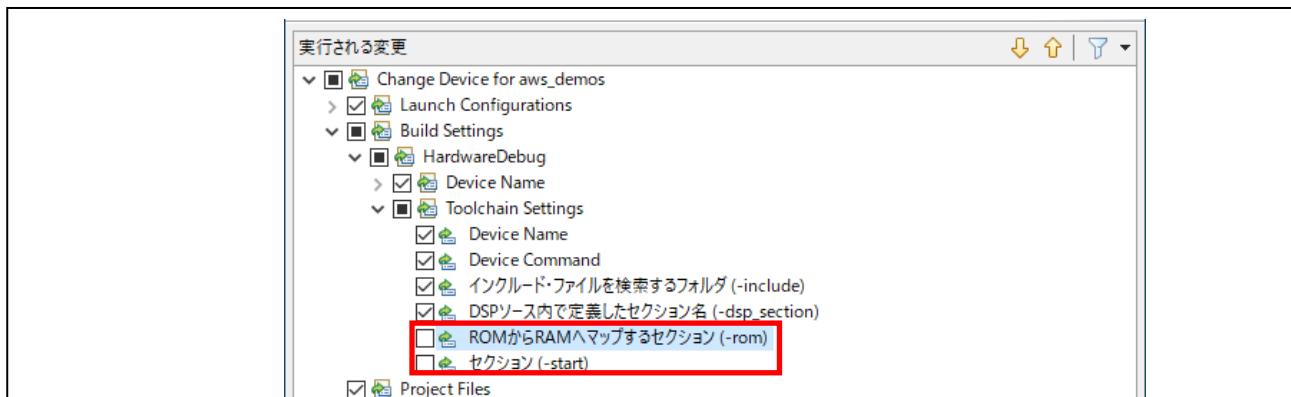


図 4-21 デバイスの変更 4 (CC-RX)

(b) デバイスの変更 GCC の場合

[Project Files]を選択します。

[src/linker\_script.ld]のチェックを外します。

[終了(F)]を押下します。



図 4-22 デバイスの変更 4 (GCC)

## (3) RYZ014A の設定(RYZ014A 使用時)

RYZ014A を使用する場合は、コンポーネントの設定が必要です。

プロジェクト・エクスプローラーで、[aws\_demos/aws\_demos.scfg] をダブルクリックしてスマート・コンフィグレータを開きます。

[コンポーネント]タブを開き、"Middleware/ジェネリック/r\_cellular" を選択します。

[設定]エリアで次の値を入力します。入力する値は SIM カードの説明を参照してください。

- Access point name
- Access point login ID
- Access point password
- SIM card PIN code

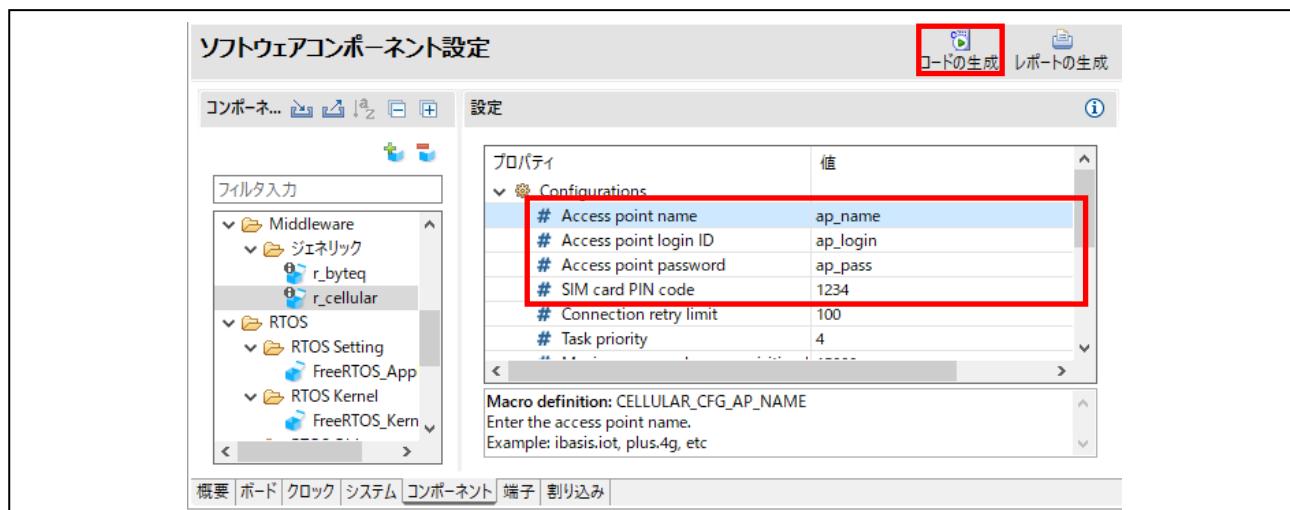


図 4-23 セルラーコンポーネント設定

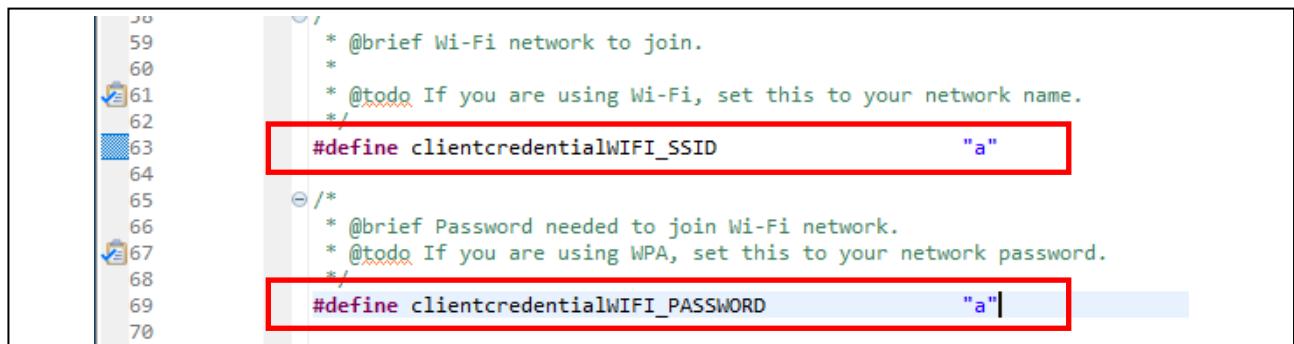
スマート・コンフィグレータから変更を加えた場合は、[コードの生成]ボタンを押下し、コード生成を行ってください。

## (4) Wi-Fi 設定(RYZ014A 使用時)

CK-RX65N へ RYZ014A を接続して使用する場合、Wi-Fi を使用しませんが、実行時エラー(ネットワーク設定に失敗)を回避するために Wi-Fi の SSID と PASSWORD にダミーの文字列を設定します。

[aws\_demos¥demos¥include¥aws\_clientcredential.h] を開き、次のマクロにダミーの文字列を設定します。(下図では例として “a” を設定しています。)

- clientcredentialWIFI\_SSID
- clientcredentialWIFI\_PASSWORD



```
58
59     * @brief Wi-Fi network to join.
60     *
61     * @todo If you are using Wi-Fi, set this to your network name.
62     */
63     #define clientcredentialWIFI_SSID           "a"
64
65     /*
66     * @brief Password needed to join Wi-Fi network.
67     * @todo If you are using WPA, set this to your network password.
68     */
69     #define clientcredentialWIFI_PASSWORD        "a"
```

図 4-24 Wi-Fi ネットワーク ダミー設定

## 4.4.1.2 【Azure の場合】

## (1) プロジェクトの作成

ADU サンプルプロジェクトとブートローダープロジェクトを作成します。まずはブートローダープロジェクトを作成しますが、ADU サンプルプロジェクトの作成方法も、一部を除き同じ手順です。本書では [4.1](#) に記載のプロジェクトフォルダにプロジェクトを作成します。既存プロジェクトを使用する場合はこの手順をスキップしてください。

e<sup>2</sup> studio のメニューから[ファイル][新規(N)][ C/C++ Project]を選択します。

新規プロジェクト作成ダイアログが開きますので、コンパイラに合わせて"Renesas CC-RX C/C++ Executable Project" か "GCC for Renesas RX C/C++ Executable Project" を選択し、[次へ(N)>]ボタンを押下します。任意のプロジェクト名を入力後、[次へ(N)>]ボタンを押下します。

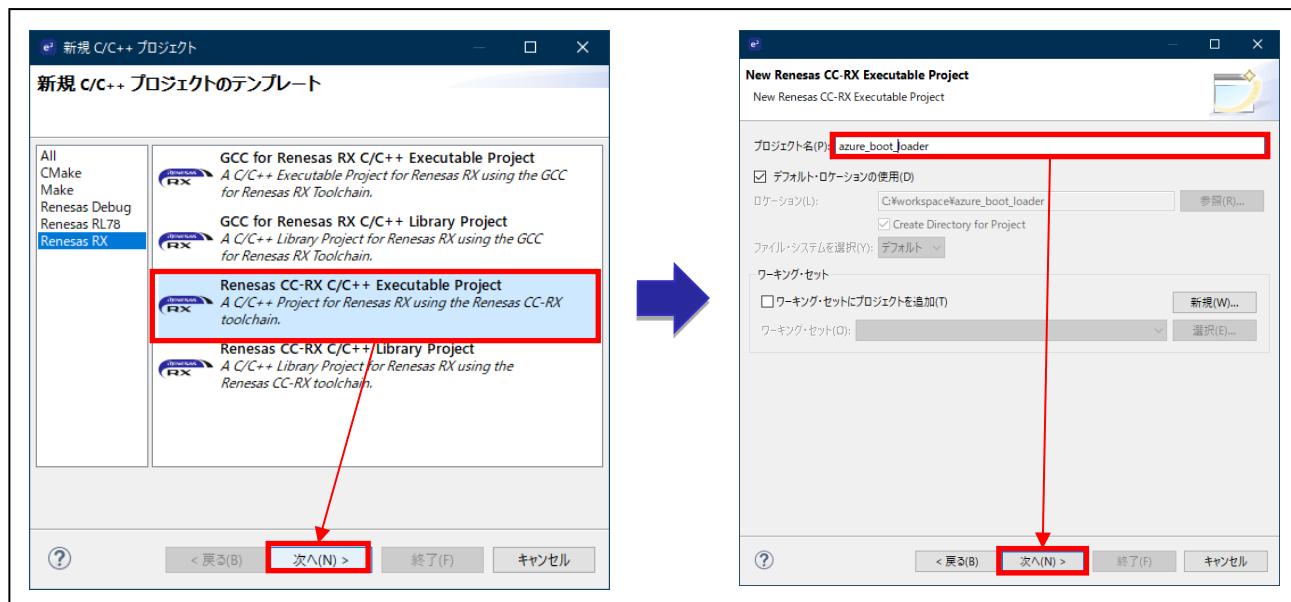


図 4-25 Azure プロジェクトの作成 1

[RTOS]コンボボックスで、"Azure RTOS"を選択します。

[RTOS Versions]コンボボックスで、"6.2.1\_rel-rx-1.0.1"を選択します。コンボボックスが空白の場合は、その下の[Manage RTOS Versions...]リンクをクリックすると、ダウンロードダイアログが開きますので、ダウンロード先となる[モジュール・フォルダー・パス]を指定し、ダウンロードします。

ダウンロード後、[RTOS Versions]コンボボックスでダウンロードしたバージョンが選択可能になります。

使用するリビジョンの詳細や選定については GitHub([Releases · renesas/azure-rtos \(github.com\)](https://github.com/renesas/azure-rtos))を参照してください。

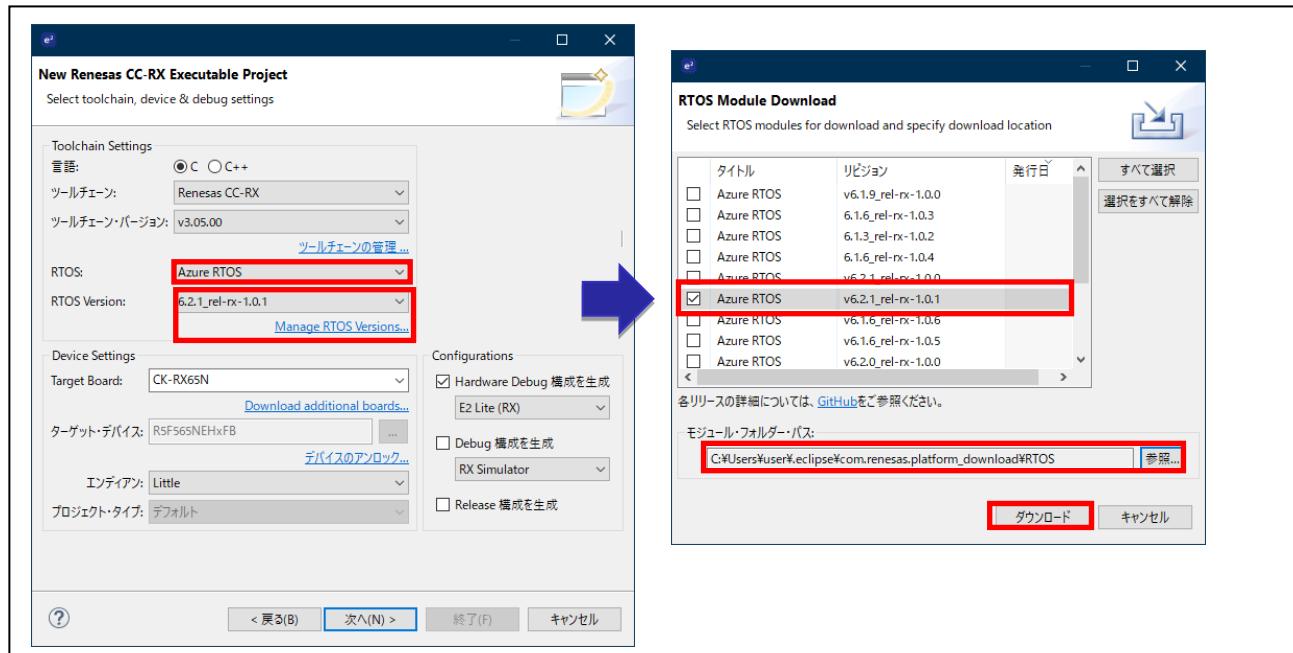


図 4-26 Azure プロジェクトの作成 2

[Target Board]で、"CK-RX65N"を選択し、[次へ(N)>]ボタンを押下します。

スマート・コンフィグレータの説明が表示されますので、[次へ(N)>]ボタンを押下して進みます。

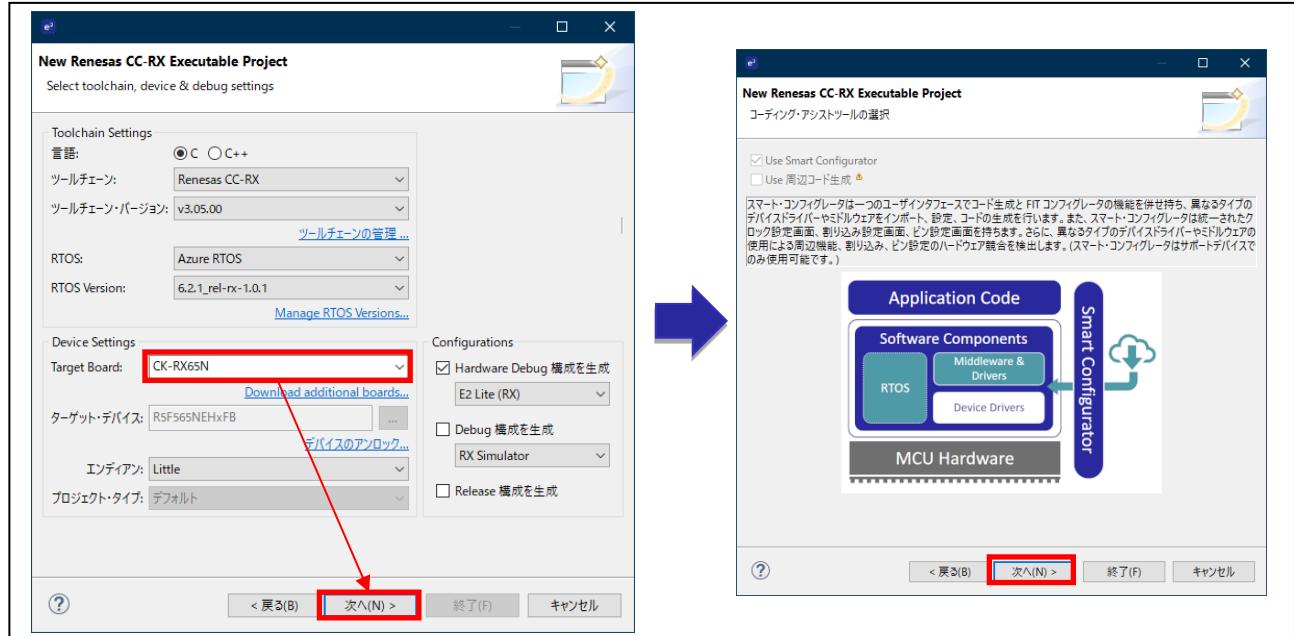


図 4-27 Azure プロジェクトの作成 3

[RTOS プロジェクト設定]の[アプリケーションを選択]エリアで、"Secure bootloader sample project"を選択し、[終了(F)]ボタンを押下します。

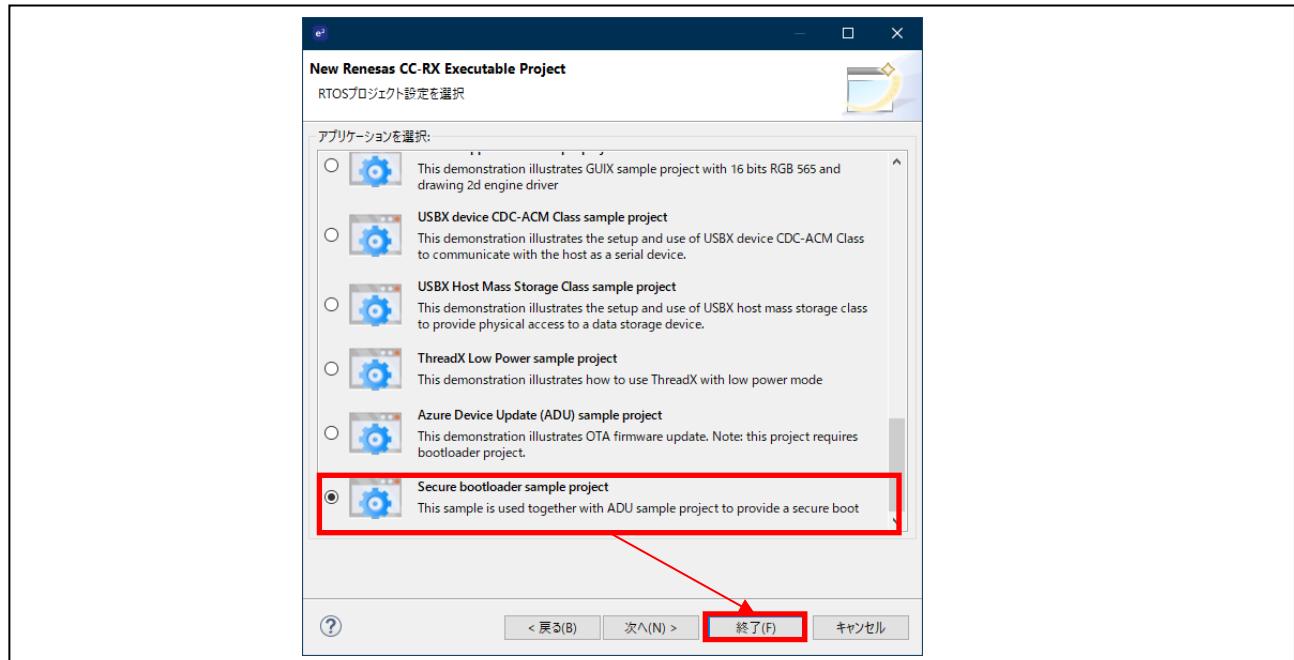


図 4-28 Azure RTOS プロジェクトの選択 ブートローダープロジェクト

プロジェクトが生成され、[README.md] ファイルが自動で開きます。その記述に従って、プロジェクトの設定を行います。

プロジェクト・エクスプローラーで、[<プロジェクト名>.scfg] ファイルをダブルクリックしてスマート・コンフィグレータを開きます。その後、[ボード]タブを選択し、ボードの選択ダイアログを開きます。

ターゲットデバイス名の後ろに”\_DUAL”を追加します。

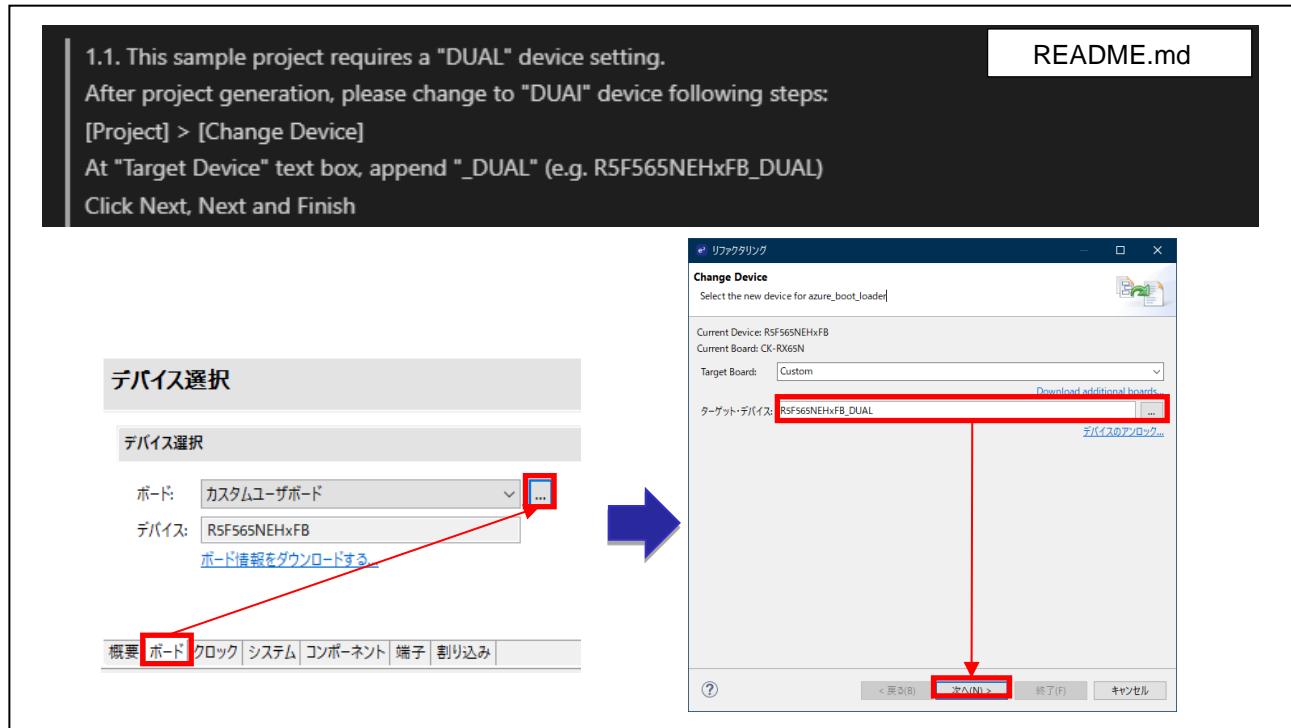


図 4-29 Azure プロジェクトのデバイス選択 1

その後、[次へ(N)>]、[終了(F)]と順に押下し、デバイスの変更を完了します。デバイスの変更を行うと、コンパイラの設定がリセットされますのでご注意ください。

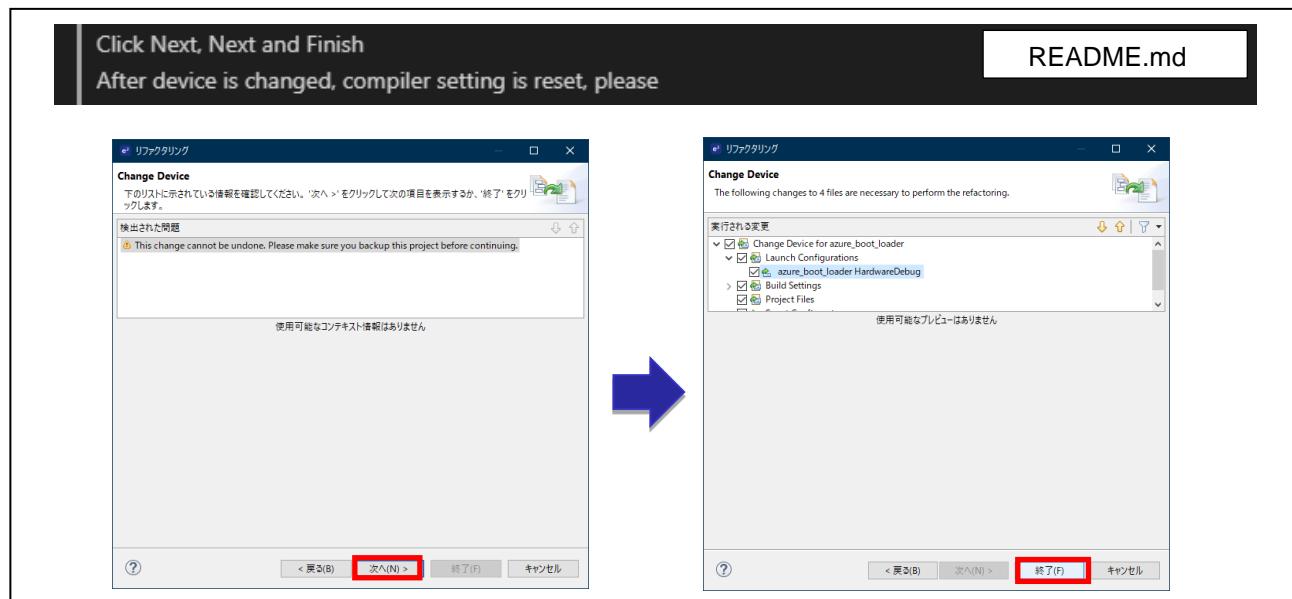


図 4-30 Azure プロジェクトのデバイス選択 2

上記の手順の途中、以下のようなダイアログが出た場合は、「はい」を押下してください。

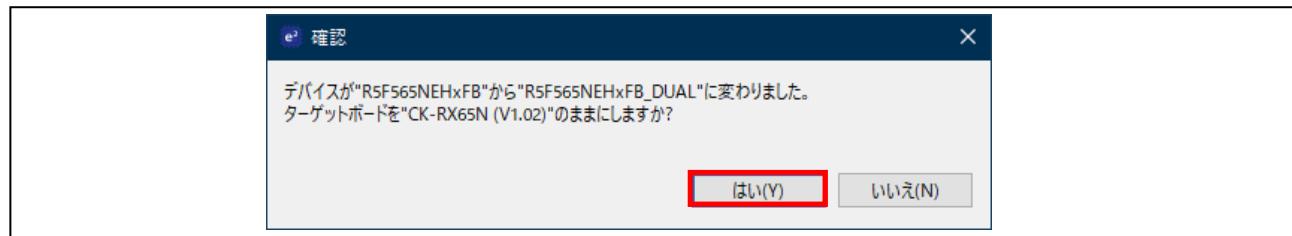


図 4-31 Azure プロジェクトのデバイス選択中のダイアログ

## (a) セクションの設定 CC-RX の場合

セクションの設定を行います。プロジェクトを選択し、右クリックからプロパティを開きます。「設定」を選択し、下図のボタンからセクション設定を開きます。開いたダイアログで[インポート...]ボタンを押下します。

ファイル選択ダイアログが開きますので、プロジェクト内 src フォルダにある[linker\_section\_sample.esi]ファイルを選択し、インポートします。※注 1

【注】 1. ターゲットデバイス名変更後、最初の初期ファームウェア作成時にセクション設定が変更されてしまい、初期ファームウェア作成に失敗することがあります。その場合は、再度セクションの設定を実施してください。

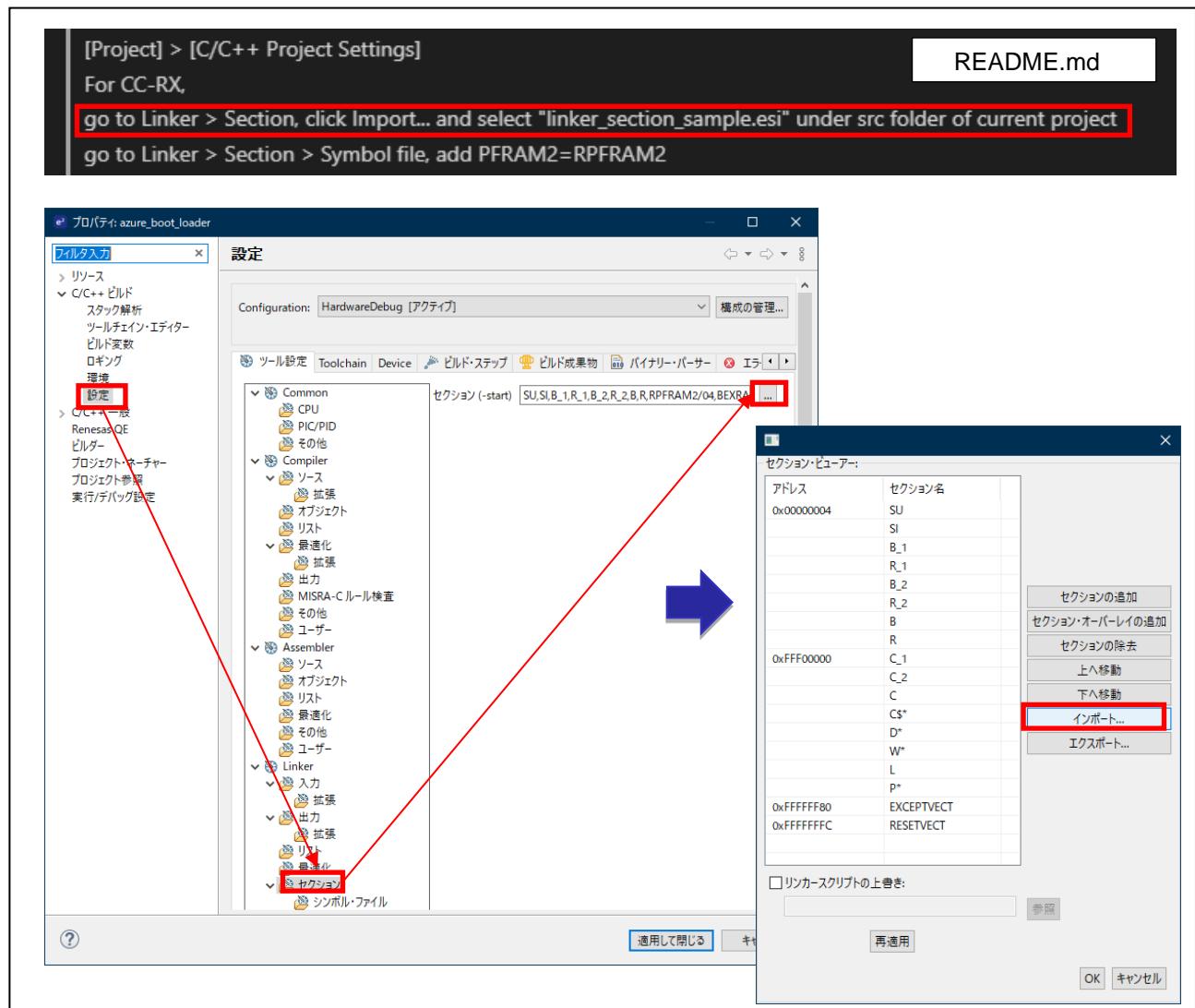


図 4-32 Azure プロジェクトのセクション設定 1

その後、[セクション]>[シンボル・ファイル]から、"PFRAM2=RPFRAM2" を追加します。

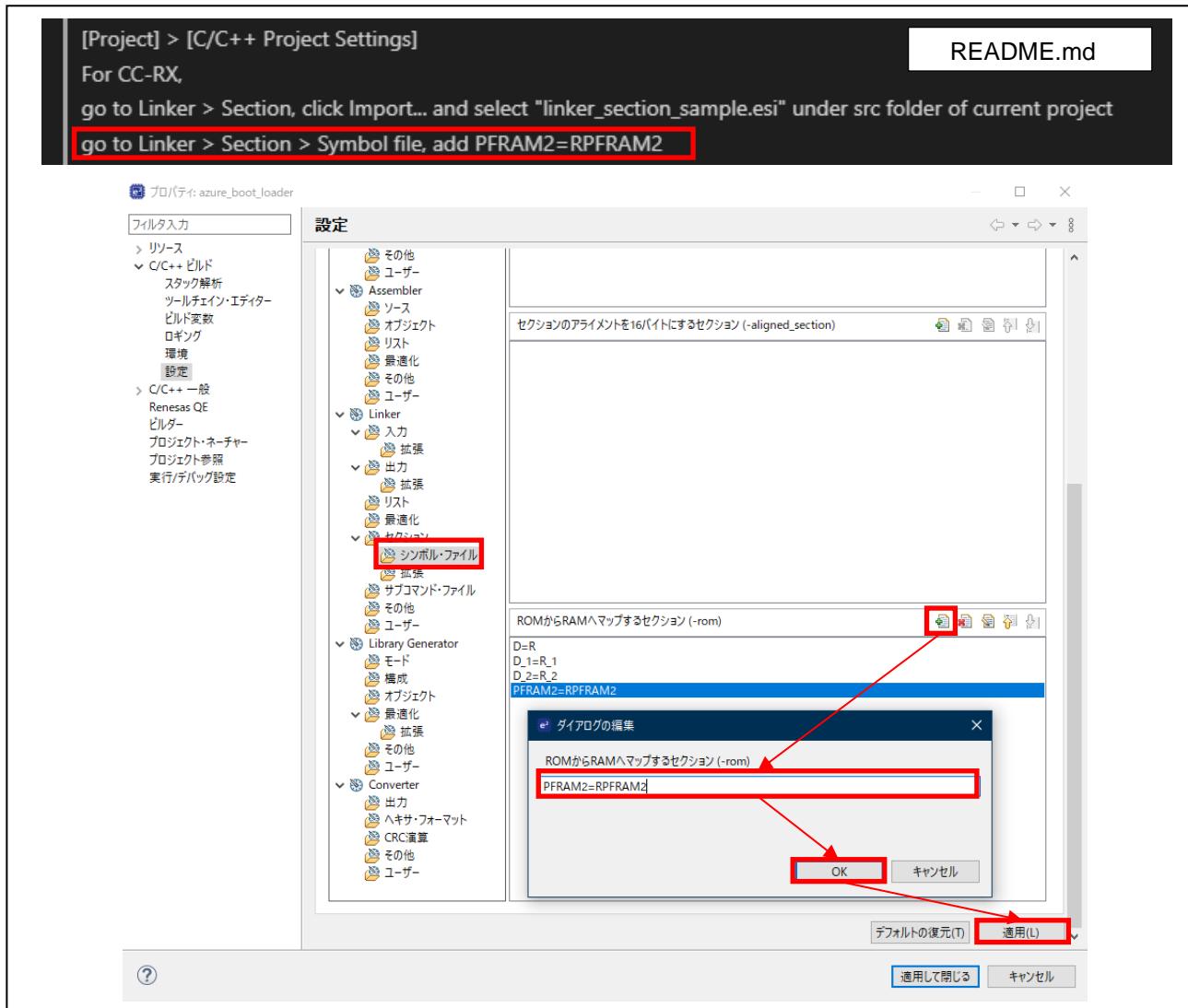


図 4-33 Azure プロジェクトのセクション設定 2

(b) セクションの設定 GCC の場合

GCC を使用する場合は、リンクの設定ファイルを置き換えてください。

置き換え元 :

<プロジェクトフォルダ>/src/linker\_script\_sample.ld

置き換え先 :

<プロジェクトフォルダ>/src/linker\_script.ld

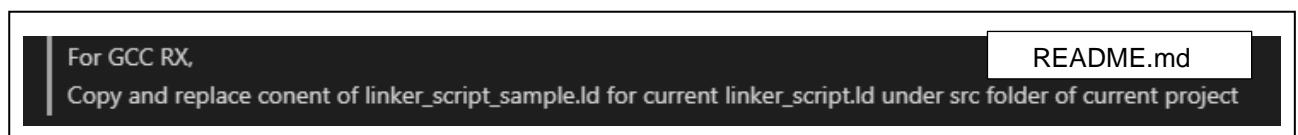


図 4-34 Azure プロジェクトのセクション設定 3

## (2) ADU サンプルプロジェクトの作成

次に ADU サンプルプロジェクトの作成を行います。手順はブートローダープロジェクトと同じですが、RTOS プロジェクト設定の選択時、"Azure Device Update (ADU) sample project"を選択します。

デバイスの変更やセクションの設定についても同様に必要です。

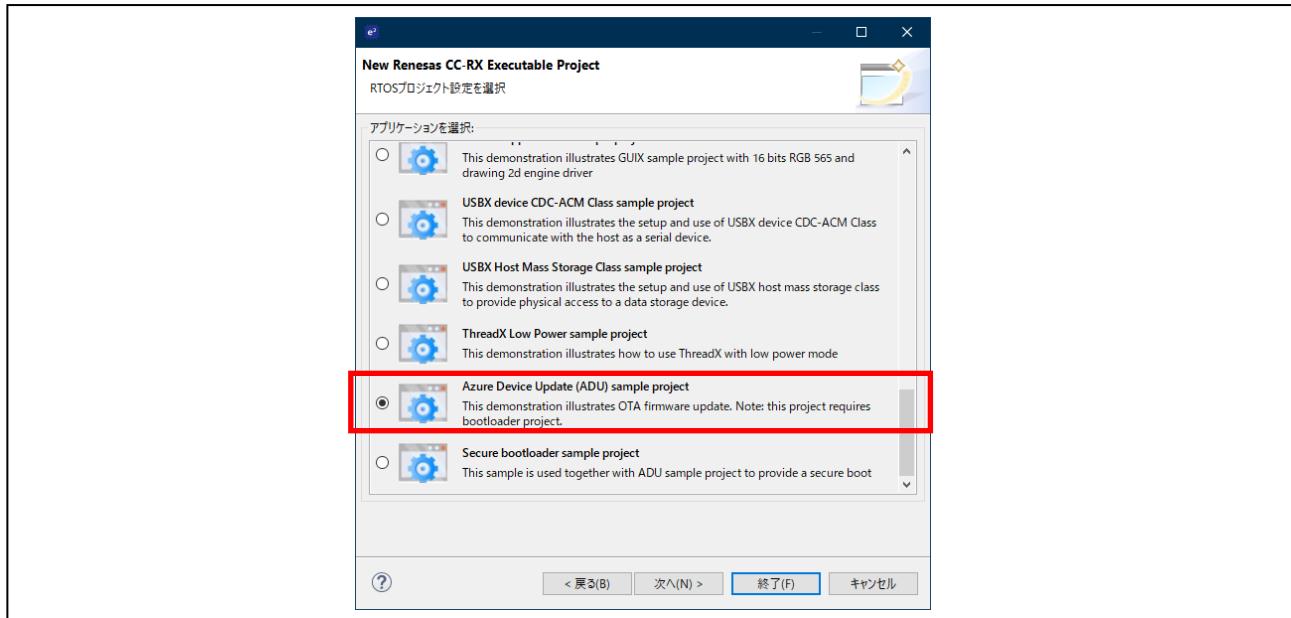


図 4-35 Azure RTOS プロジェクトの選択 ADU プロジェクト

## (3) プロジェクトの選択

作成したプロジェクトを選択します。



図 4-36 Azure 作成したプロジェクトの選択

## 4.4.1.3 スマート・コンフィグレータのコンポーネントのダウンロード

サンプルプロジェクトとブートローダープロジェクトの作成ができたら、プロジェクト・エクスプローラーで、それぞれのプロジェクト内にある[<プロジェクト名>.scfg]ファイルをダブルクリックし、スマート・コンフィグレータを表示します。

[コンポーネント]タブを開き、立方体のアイコンがグレーになっているノードがあれば[コンポーネントをダウンロード]リンクをクリックしてダウンロードを行ってください。

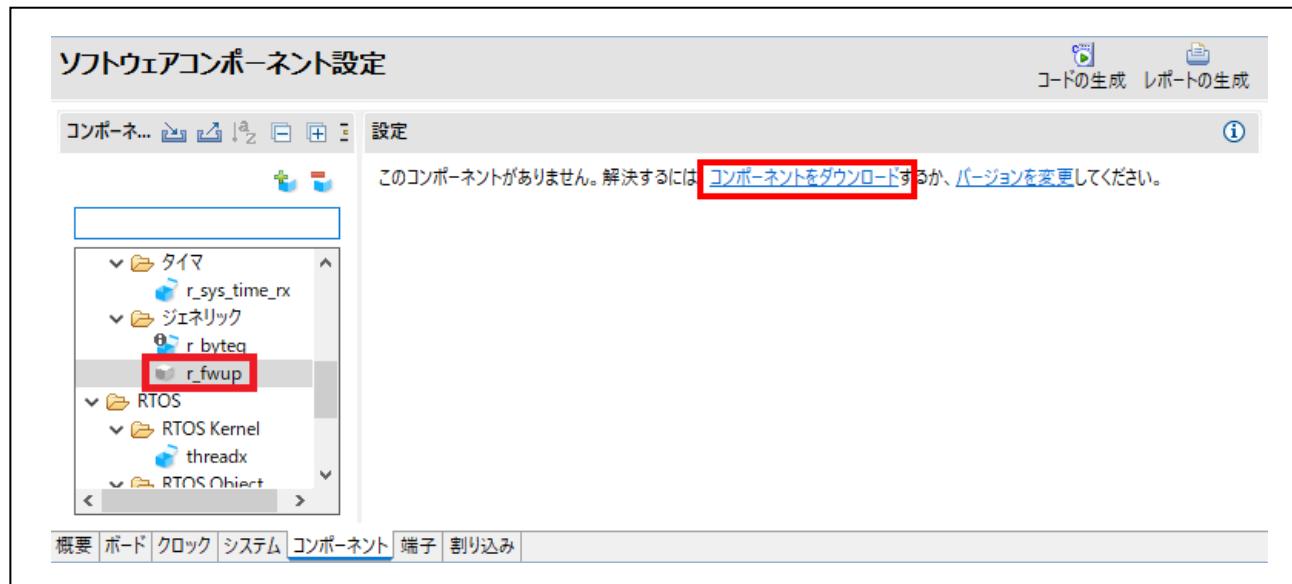


図 4-37 コンポーネントのダウンロード

## 4.4.2 プロビジョニングの選択

QE for OTA V1.1.0 では、1 つのプロビジョニングにのみ対応しています。

特に操作は必要ありません。



図 4-38 プロビジョニングの選択

## 4.5 IoT デバイスの管理

クラウド上に、IoT デバイスを新規に作成します。

### 4.5.1 IoT デバイスの管理

OTA メイン(QE) ビューの左側に表示されているメニューから[IoT デバイスの管理]を選択し、[ビューを開く]ボタンを押下します。

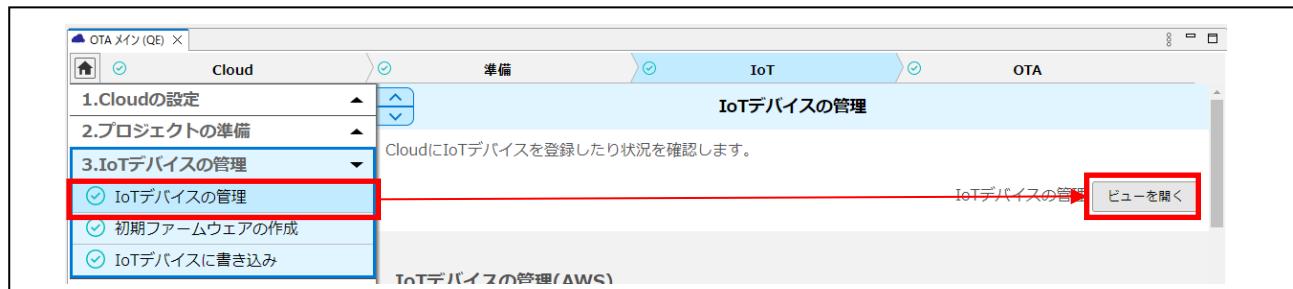


図 4-39 IoT デバイスの管理

OTA IoT デバイス管理 (QE) ビューが開きます。

[追加]アイコンをクリックして[IoT デバイス作成]を選択します。

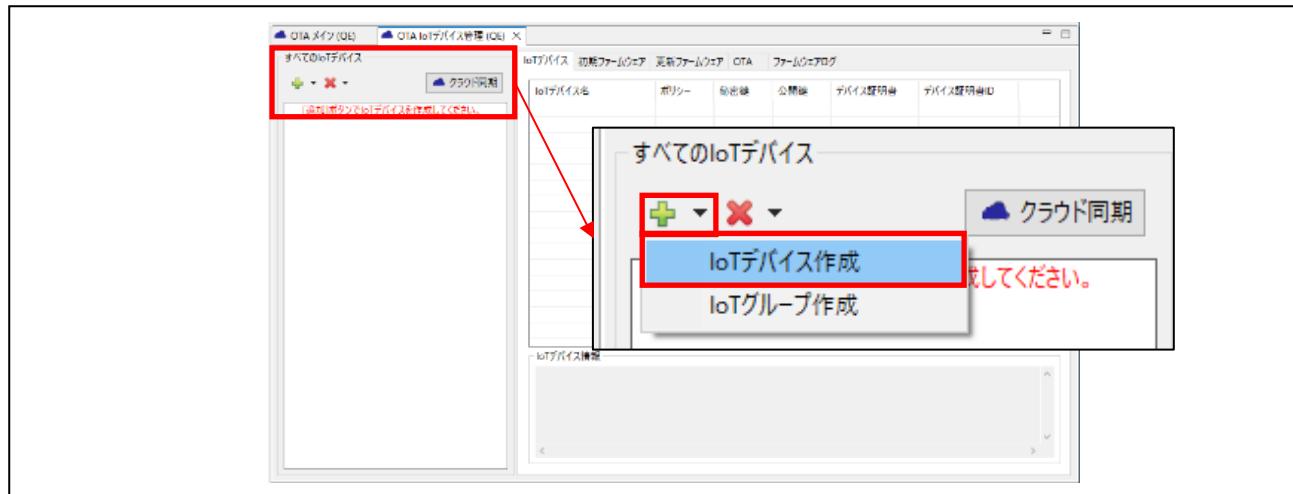


図 4-40 IoT デバイスの管理 IoT デバイス作成

作成する IoT デバイスの情報を入力して[作成]ボタンを押下します。



図 4-41 IoT デバイス作成ダイアログ

IoT デバイス情報の項目内容を以下に示します。

表 4-1 IoT デバイス作成パラメータ

項目	内容
IoT デバイス名	IoT デバイス名を指定します。作成数が 2 以上の場合、QE が自動的に IoT デバイス名に連番を付加します。
グループ名	グループを指定する必要がある場合は指定してください。(AWS のみ)
作成数	作成する数を指定してください。
ポリシー	ポリシーを選択します。新規作成が選択されている場合は、QE が自動的に作成します。(AWS のみ)

作成された IoT デバイス情報を確認します。

デバイス証明書及び鍵が自動的にダウンロードされ、表示されます。

AWS と Azure で表記が異なります。

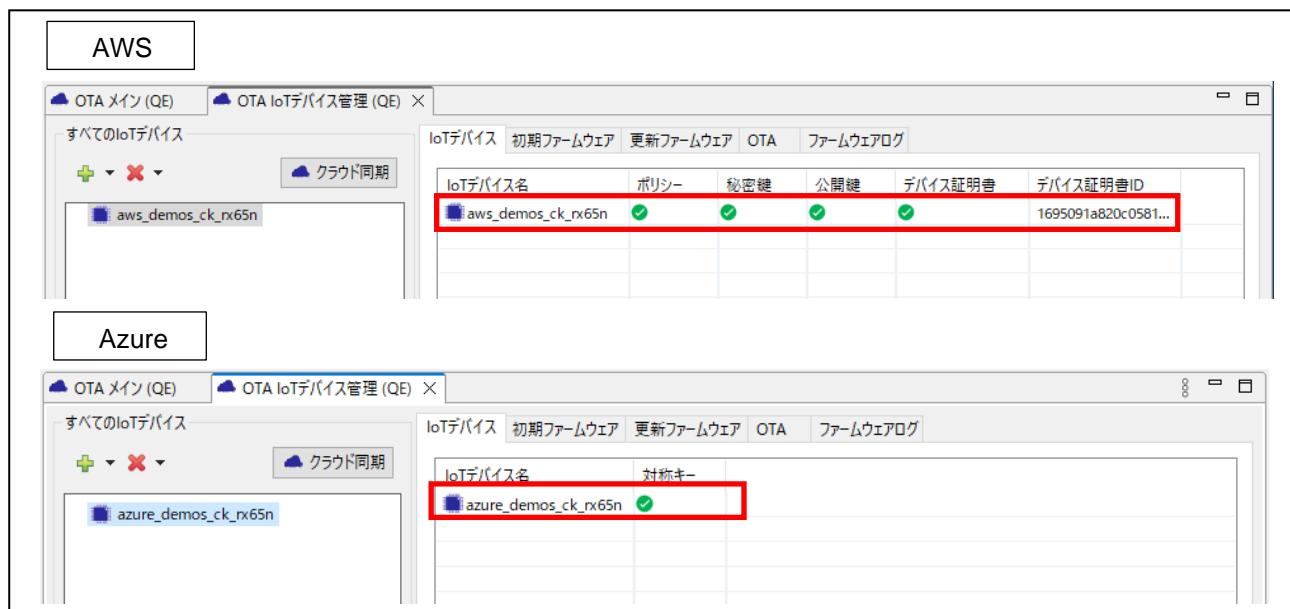


図 4-42 IoT デバイスの管理 IoT デバイス作成後の表示

#### 4.5.2 初期ファームウェアの作成

初期ファームウェアの作成手順について説明します。

##### 4.5.2.1 ツールチェーンのバージョンの確認

各プロジェクトで使用しているツールチェーンと同じバージョンのツールチェーンがインストールされていない場合、ビルド時にエラーとなります。

同じバージョンのツールチェーンをインストールするか、より新しいツールチェーンを使用するように変更する必要があります。

使用されているツールチェーンは、各プロジェクトのプロパティ内の設定から確認できます。

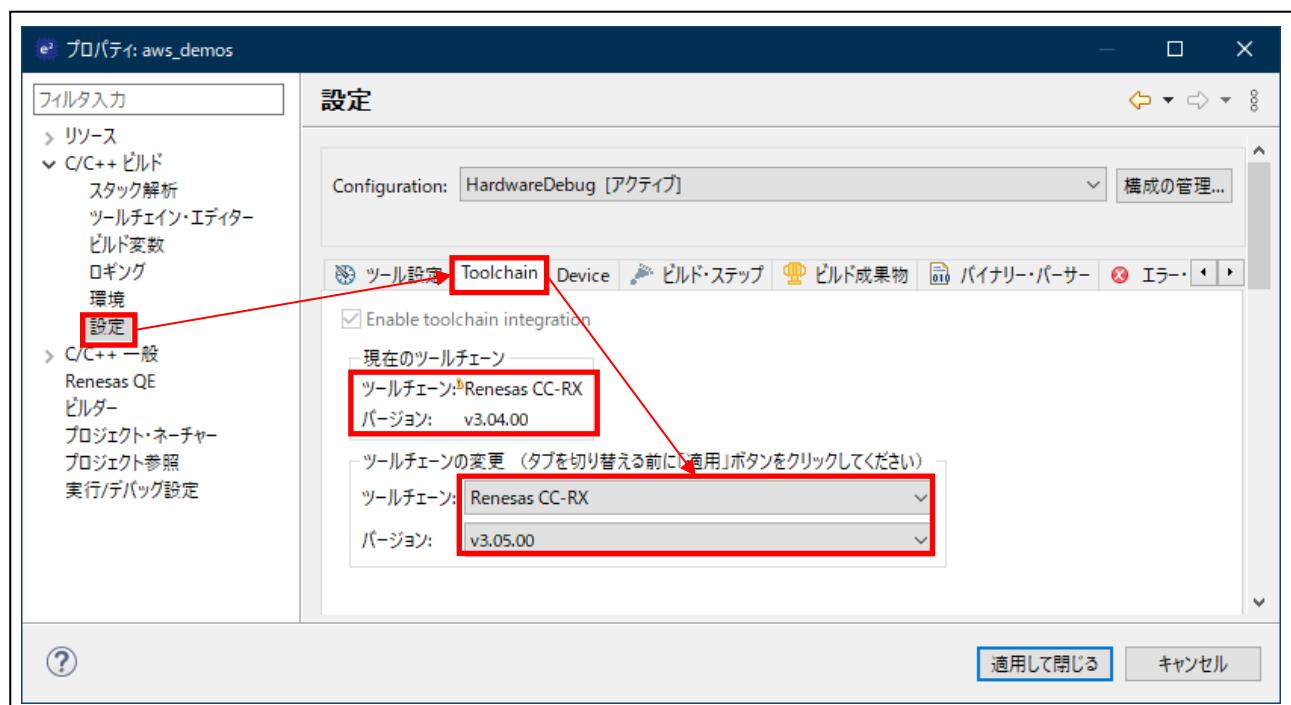


図 4-43 ツールチェーンのバージョン

#### 4.5.2.2 初期ファームウェアの作成

必要な設定を終えたら、初期ファームウェアを作成します。

OTA メイン(QE) ビューの左側に表示されているメニューから[初期ファームウェアの作成]を選択し、[ビューを開く]ボタンを押下します。OTA IoT デバイス管理(QE) ビューの[初期ファームウェア]タブが開きます。



図 4-44 初期ファームウェアの作成 ビューを開く

IoT デバイスを選択し、[登録]ボタンを押下すると、右側に状態が表示されます。表示されている全ての IoT デバイスを登録する場合は、[全登録]ボタンを押下します。

その後、登録した中から書き込む IoT デバイスを選択し、初期ファームウェアのバージョンを指定し、[初期ファームウェアの作成]ボタンを押下します。

AWS と Azure ではタブ右上の登録した IoT デバイスの表示が異なりますが、手順は同じです。

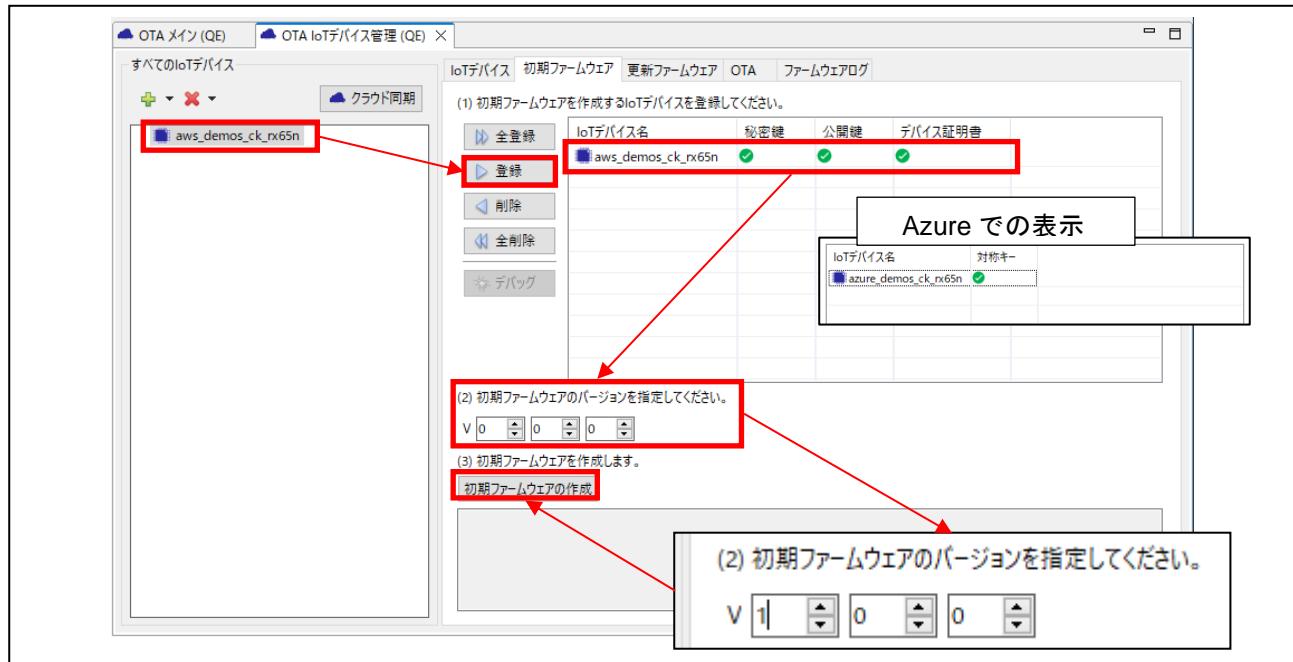


図 4-45 初期ファームウェアの作成

[初期ファームウェアの作成]ボタンを押下すると、コード署名ダイアログが開きます。既にコード署名証明書の作成が完了している場合は、証明書のkopiedialogが開きます。(図 4-51 コード署名証明書ファイルのkopiedialogを参照)

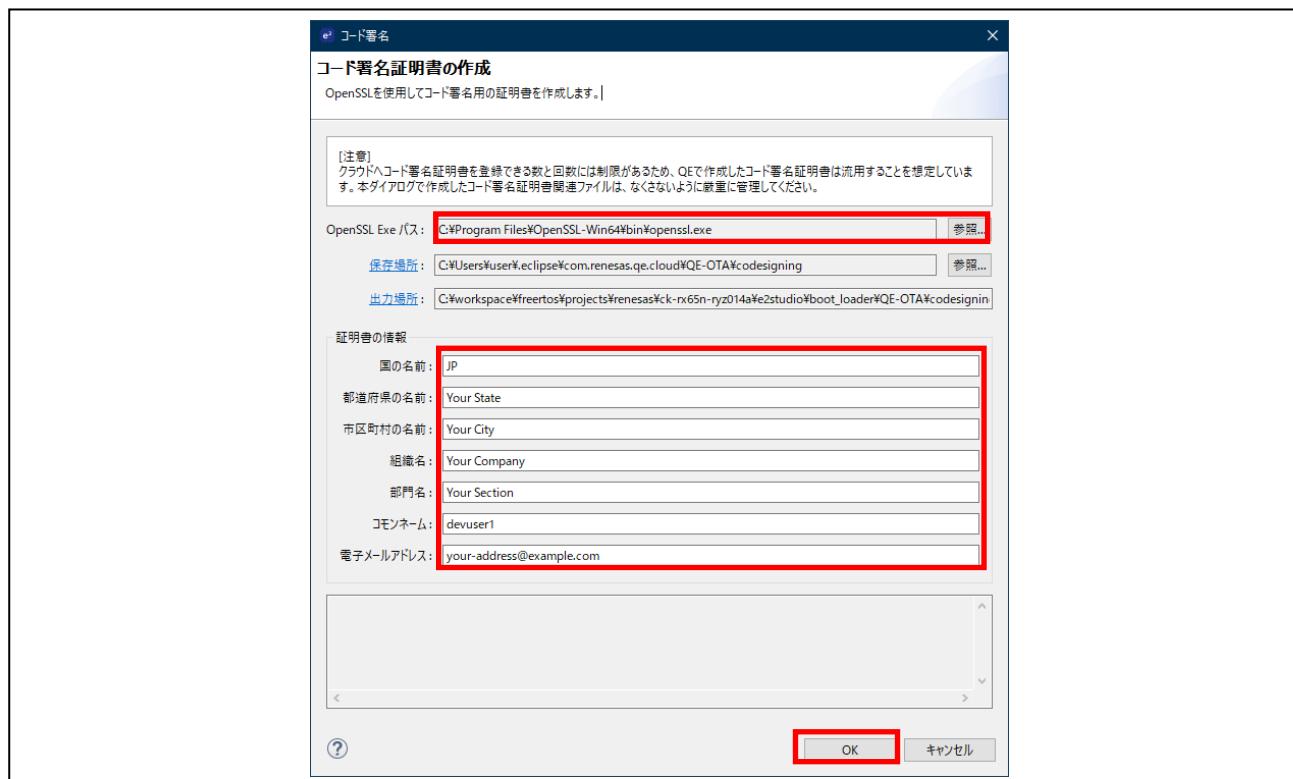


図 4-46 コード署名証明書の作成ダイアログ

OpenSSL のパスを設定し、証明書の内容を入力後、OK ボタンを押下すると、自動的に初期ファームウェアの作成が行われます。

尚、クラウドへコード署名証明書を登録できる数と回数には制限があるため、QE で作成したコード署名証明書は流用することを想定しています。流用する場合は、コード署名ダイアログは表示されません。流用方法については後述します。(4.5.2.4 コード署名証明書について)

正常に作成できた場合は、以下のよう表示になります。これで、初期ファームウェアの作成は完了です。

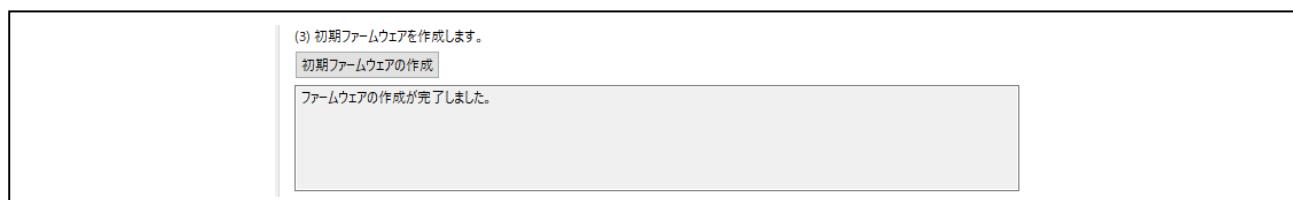


図 4-47 初期ファームウェアの作成 正常終了

これらの手順を行う際に、次のようなダイアログが表示される場合があります。使用しているボードを選択してください。

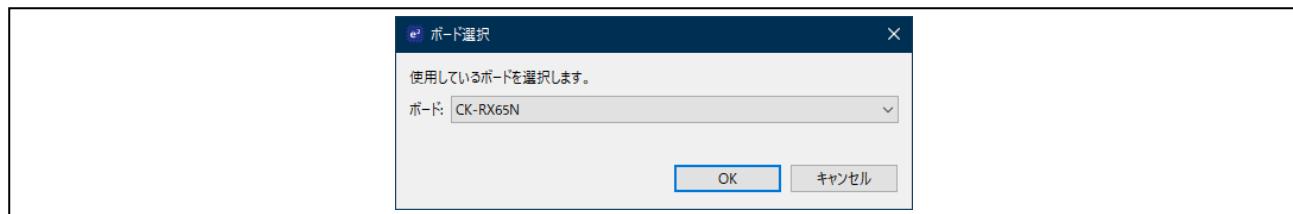


図 4-48 ボード選択ダイアログ

## 4.5.2.3 サンプルプロジェクトのファームウェア

ファームウェアは下図の位置に以下の内容で生成されます。

<プロジェクトフォルダ>/QE-OTA/apl/<バージョン>/<IoT デバイス名>

- <プロジェクト名>.mot  
サンプルプロジェクトのモトローラ形式の HEX ファイルです。
- <プロジェクト名>.x (CC-RX の場合), <プロジェクト名>.elf (GCC の場合)  
サンプルプロジェクトのロードモジュールファイルです。
- merged.mot  
サンプルプロジェクトとブートローダーをマージしたモトローラ形式の HEX ファイルです。このファイルを IoT デバイスに書き込みます。

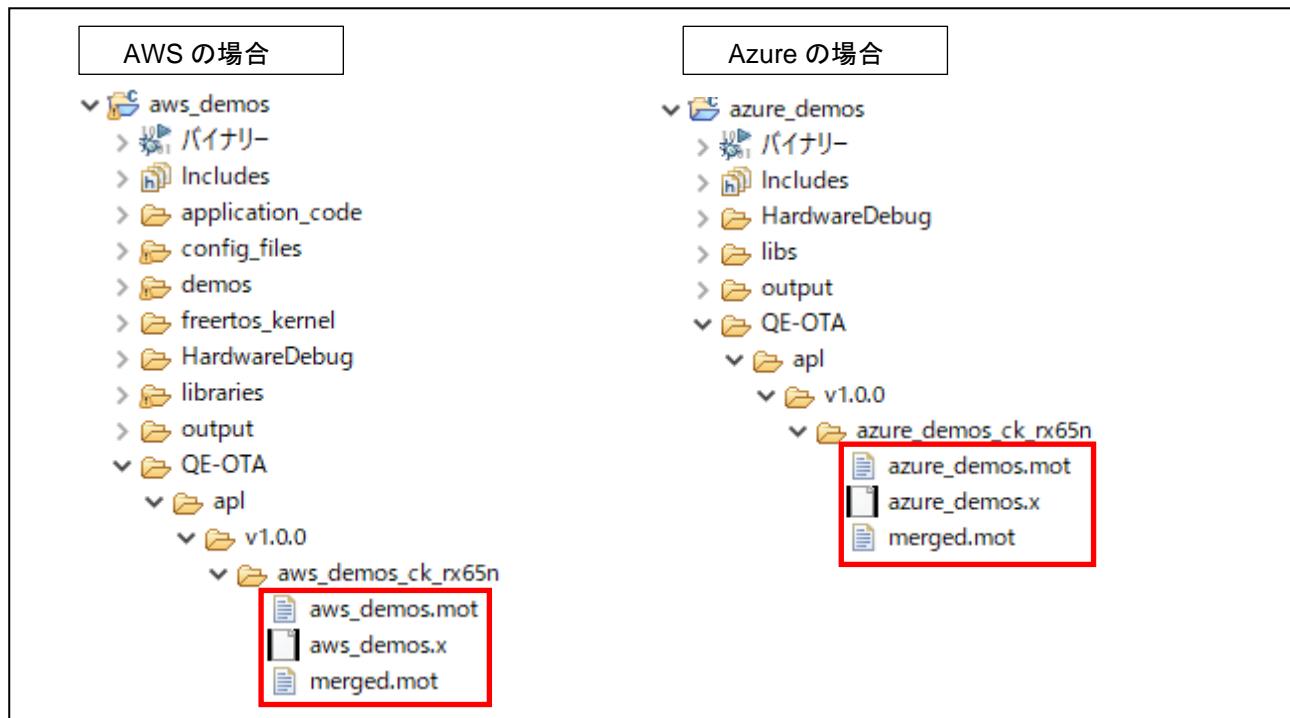


図 4-49 初期ファームウェアの作成位置 (CC-RX)

## 4.5.2.4 コード署名証明書について

作成されたコード署名証明書は、ブートローダープロジェクト内の以下の位置に格納されます。

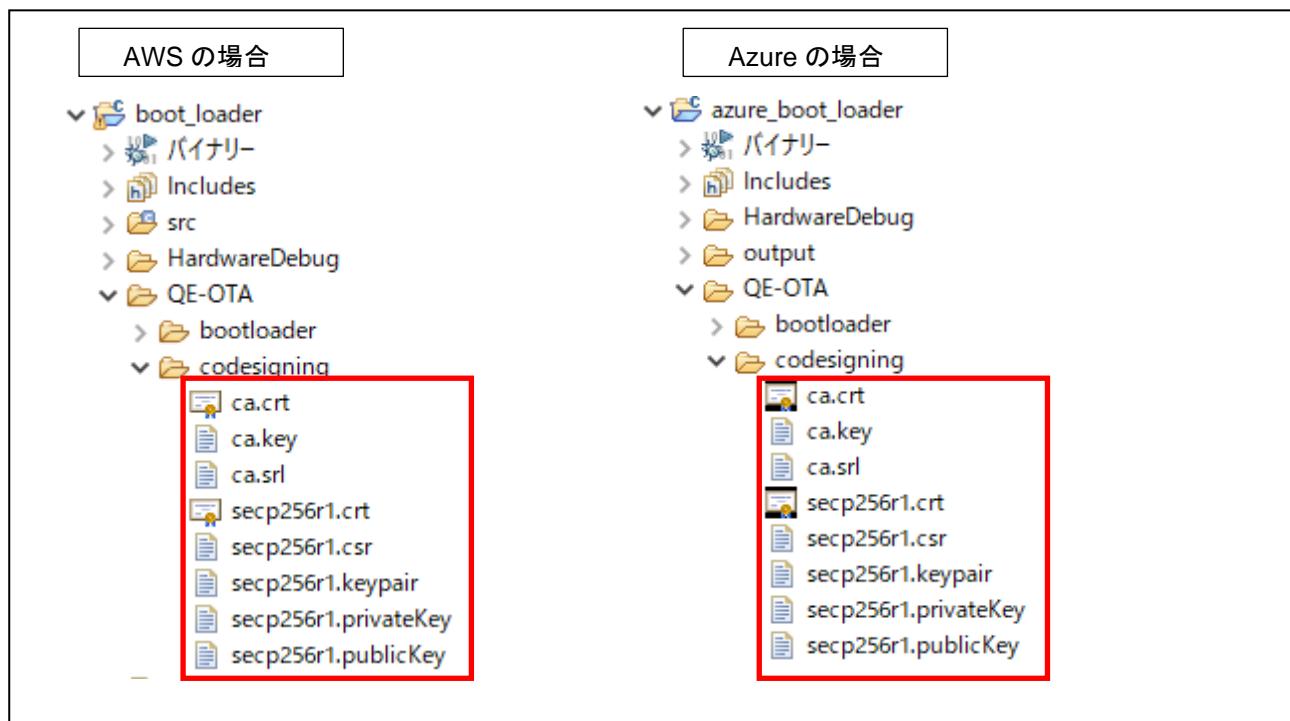


図 4-50 コード署名証明書の位置

この時作成したコード署名証明書と関連ファイルは、なくさないように厳重に管理してください。最初の OTA 実行時に、ここで生成されたコード署名証明書を使用してコード署名プロファイル(qe\_code\_signing)を作成しますが、一度コード署名プロファイルを作成すると、QE for OTA からは再作成ができません。別のコード署名を使用する場合は、AWS コンソールを使用してコード署名プロファイルを作成してください。

そのため、QE for OTA で作成した証明書は、流用することを想定しています。流用は、以下の手順で行います。

1. コード署名証明書を作成したプロジェクトの下記フォルダに移動して、codesigning フォルダをコピーします。  
<ブートローダープロジェクトフォルダ>/QE-OTA/codesigning
2. 新しいプロジェクトの同じ位置に codesigning フォルダをペーストします。  
<ブートローダープロジェクトフォルダ>/QE-OTA/codesigning

尚、コード署名証明書の作成ダイアログで指定する[保存場所]に証明書が保存されている場合は、自動での流用が可能です。

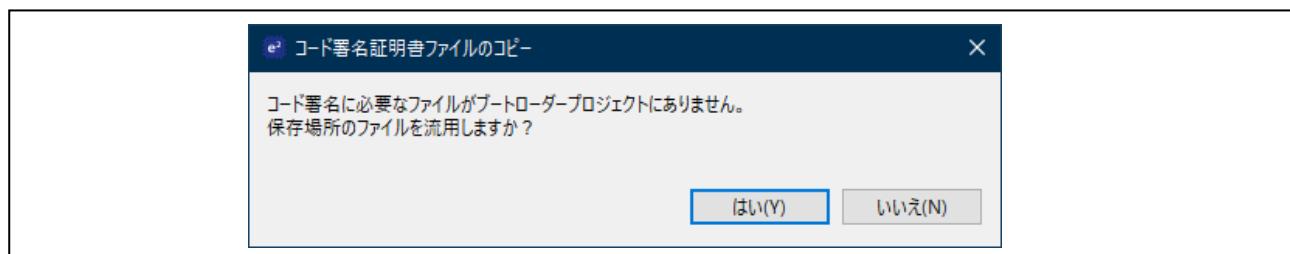


図 4-51 コード署名証明書ファイルのコピーダイアログ

## 4.5.3 IoT デバイスに書き込み

初期ファームウェアをボードに書き込みます。

OTA メイン(QE) ビューの左側に表示されているメニューから[IoT デバイスに書き込み]を選択し、[ビューを開く]ボタンを押下します。



図 4-52 IoT デバイスに書き込み

[初期ファームウェア]タブが開きます。登録した中から書き込む IoT デバイスを選択し、[デバッグ]ボタンを押下します。

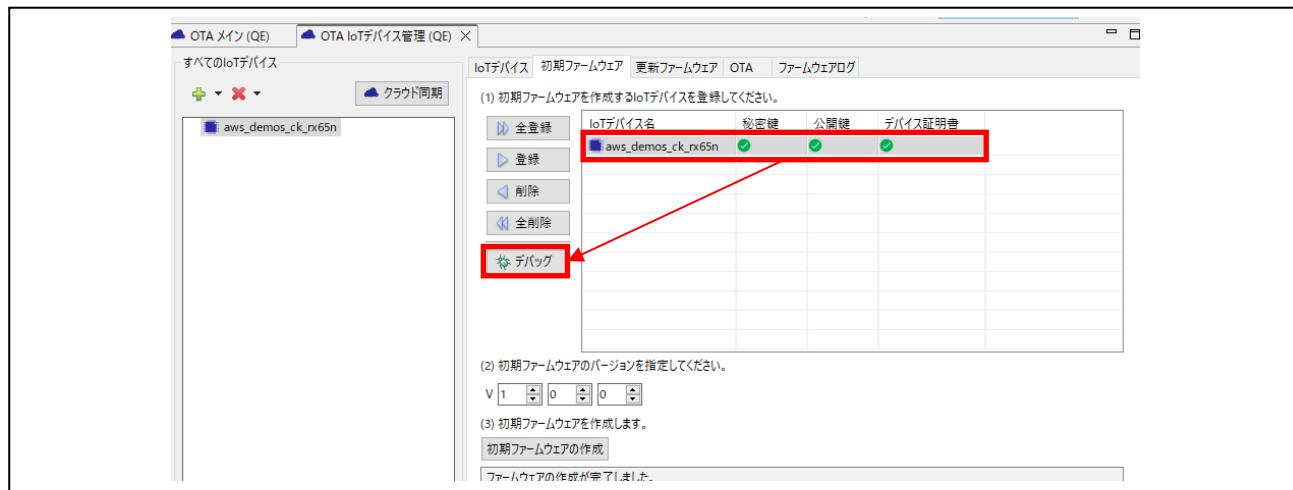


図 4-53 デバッグボタン

ファームウェアを書き込むための、Flash Programming ダイアログが開きます。

ボードを接続し、電源を入れます。

Renesas Flash Programmer (rfp-cli.exe) のパスや ID コード、使用するボードとエミュレータを指定し、[書き込み]ボタンを押下します。書き込みに成功すると、「Operation successful」と表示されます。

続けてデバッグ実行を行う場合は、[デバッグ]ボタンを押下します。

[デバッグ]ボタンを押下すると、初期ファームウェアを動作させることができます。

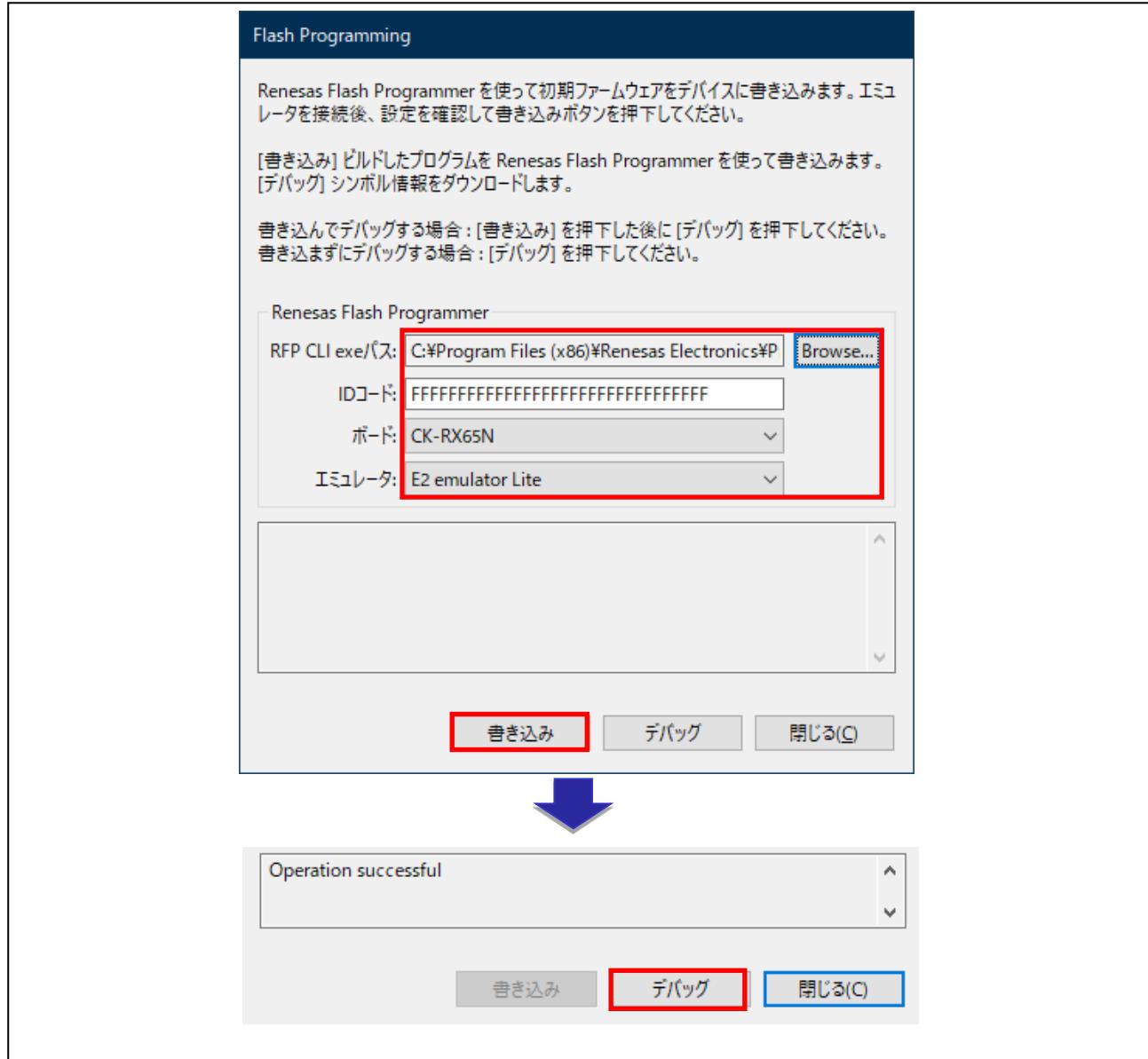


図 4-54 初期ファームウェアの書き込み

デバッグ動作を開始直後、デフォルトでは開始関数のブレークで(デフォルトでは 2 回)止まります。

OTA IoT デバイス管理(QE) ビューの、[ファームウェアログ]タブで、動作ログをモニタすることができます。COM ポートを指定し(もしくは「自動」に設定)、[接続]ボタンを押すと、動作ログが表示されます。

ログを確認する場合は、このブレーク中に COM 接続を行うと、ログが始めから途切れなく表示されます。

COM へ接続後、[再開(M)]ボタンを押下し、動作させてください。



図 4-55 ファームウェアログ表示

初期ファームウェアが正常に動作していれば、ファームウェアログに以下のような内容が表示されます。エラーを示すログがないことや、バージョン表示が 1.0.0 であることを確認してください。

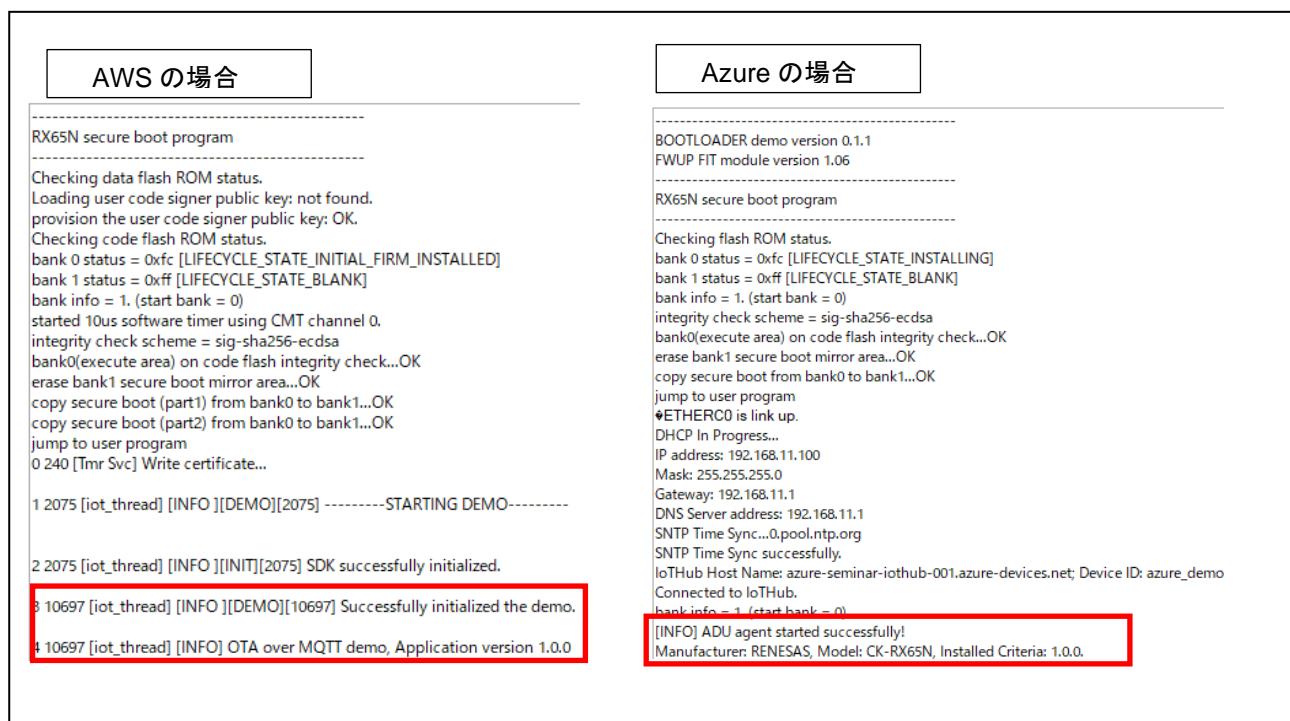


図 4-56 初期ファームウェアファームウェアログ

## 4.6 OTA

更新用のファームウェアの作成と、OTA の実行について説明します。

### 4.6.1 更新ファームウェアの作成

OTA メイン(QE) ビューの左側に表示されているメニューから[更新ファームウェアの作成]を選択し、[ビューを開く]ボタンを押下すると、OTA IoT デバイス管理 (QE) ビューの[更新ファームウェア]タブが開きます。



図 4-57 更新ファームウェアの作成

操作の流れは初期ファームウェア作成時と同様です。

まず、ソースファイルに機能追加や修正を適用した更新ファームウェア用に更新します。本アプリケーションノートでは、この更新は割愛していますが、必要な変更を実施してください。

IoT デバイスを登録・選択し、更新後のバージョンを指定し、[更新ファームウェアの作成]ボタンを押下します。正常に作成されれば「ファームウェアの作成が完了しました。」というメッセージが表示されます。

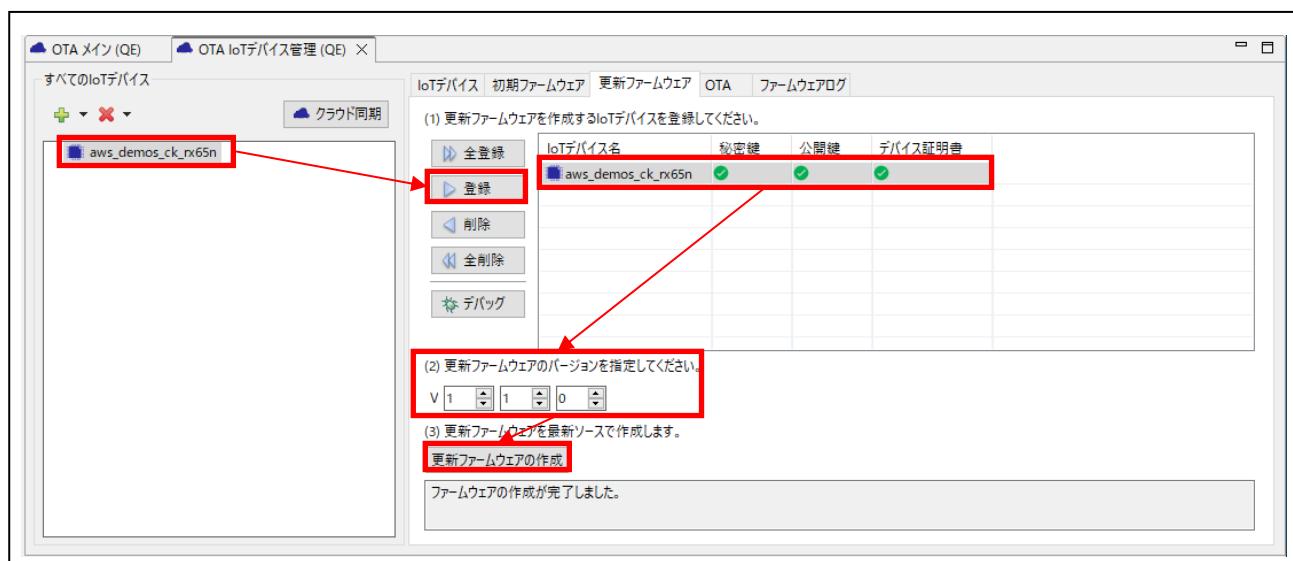


図 4-58 更新ファームウェア作成手順

更新ファームウェアは下図の位置に以下の内容で生成されます。

<プロジェクトフォルダ>/QE-OTA/apollo/<バージョン>/<IoT デバイス名>

- <プロジェクト名>.mot  
サンプルプロジェクトのモトローラ形式の HEX ファイルです。
- <プロジェクト名>.x (CC-RX の場合), <プロジェクト名>.elf (GCC の場合)  
サンプルプロジェクトのロードモジュールファイルです。
- <IoT デバイス名>.rsu  
Renesas Secure Flash Programmer で作成された更新ファームウェアです。このファイルをクラウドにアップロードします。
- <IoT デバイス名>.importmanifest.json (Azure の場合のみ)  
インポートマニフェストファイルです。このファイルをクラウドにアップロードします。

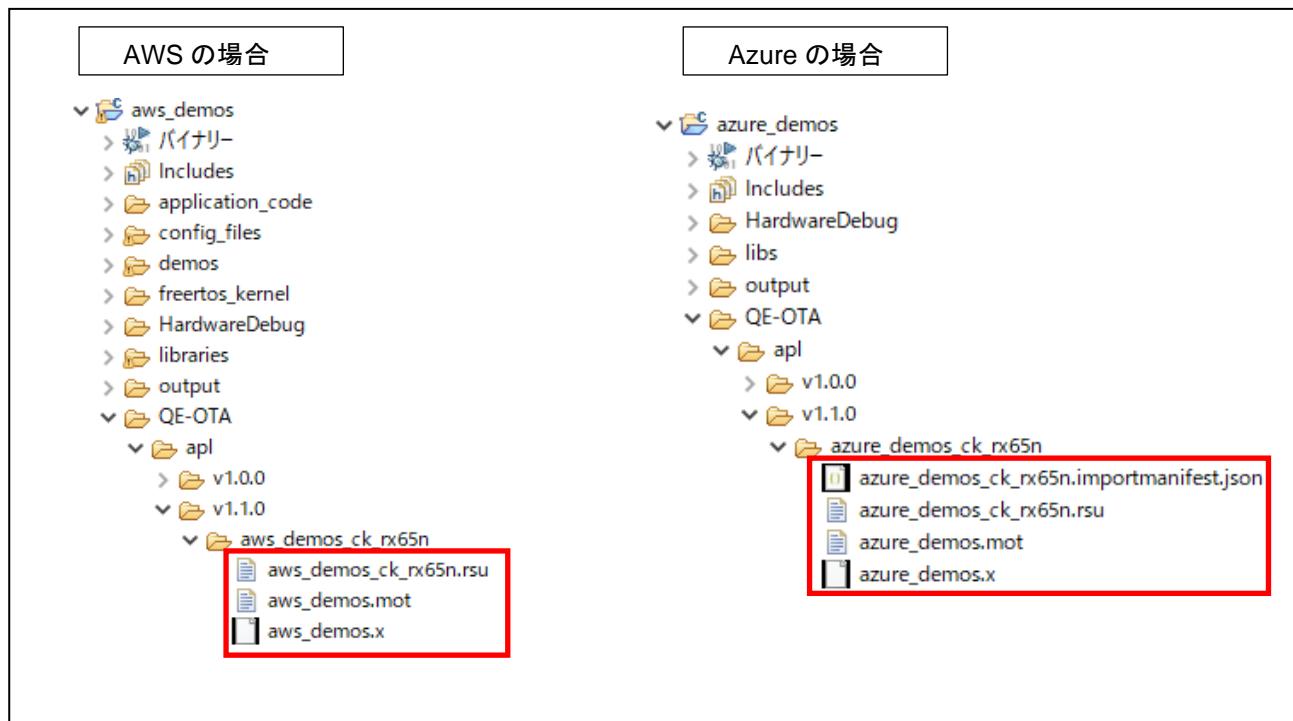


図 4-59 更新ファームウェアの作成位置 (CC-RX)

#### 4.6.2 OTA の実行と状況確認

OTA メイン(QE) ビューの左側に表示されているメニューから[OTA の実行と状況確認]を選択し、[OTA の実行]の横にある[ビューを開く]ボタンを押下すると、OTA IoT デバイス管理 (QE) ビューの、[OTA]タブが開きます。



図 4-60 OTA の実行と状況確認

他のタブ同様、OTA を実行する IoT デバイスを選択後、[登録]ボタンを押し登録します。

登録した IoT デバイスを選択し、作成した更新ファームウェアのバージョンを指定します。

AWS の場合、OTA を実行するロール、アップデート先、コード署名プロファイルを指定します。存在しない場合は “新規作成” を選択すると、OTA 実行時に QE for OTA が自動的に作成します。Azure の場合はこの部分の表示がなく、設定不要です。

[OTA 実行]ボタンを押すと、OTA を開始します。

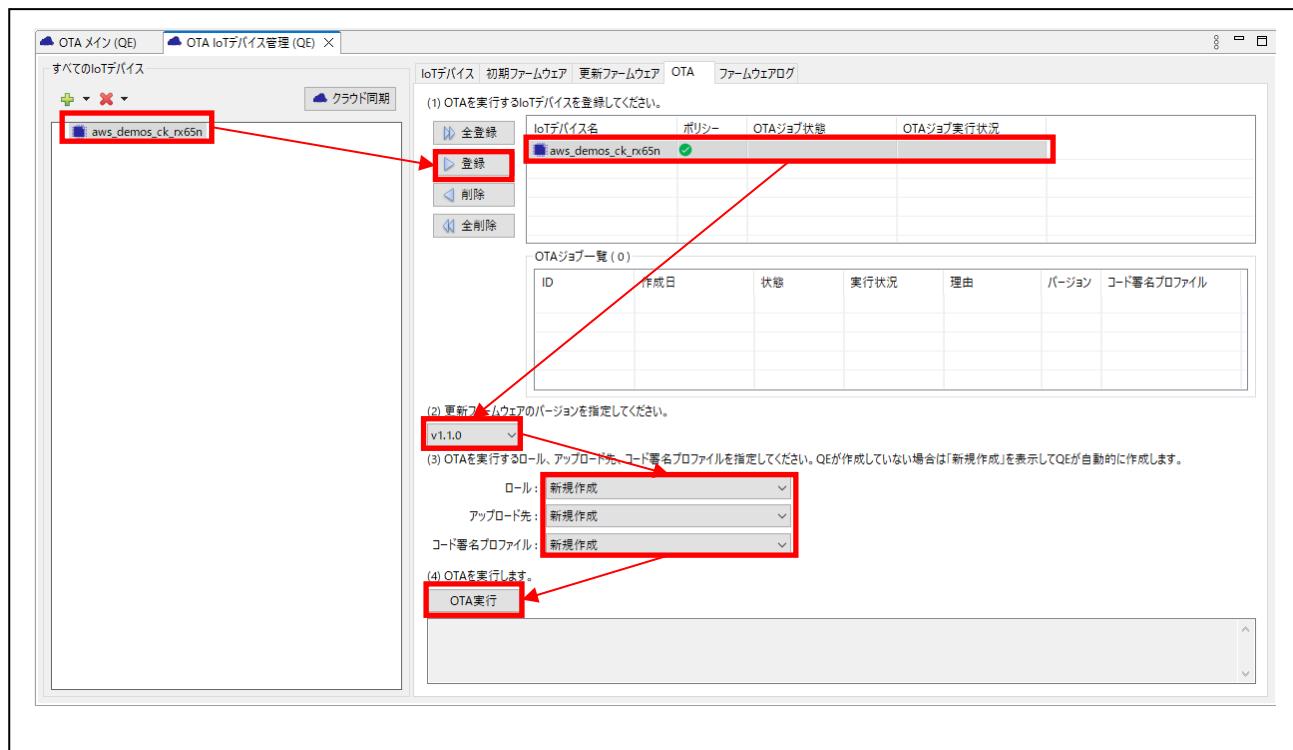


図 4-61 OTA 設定 (AWS)

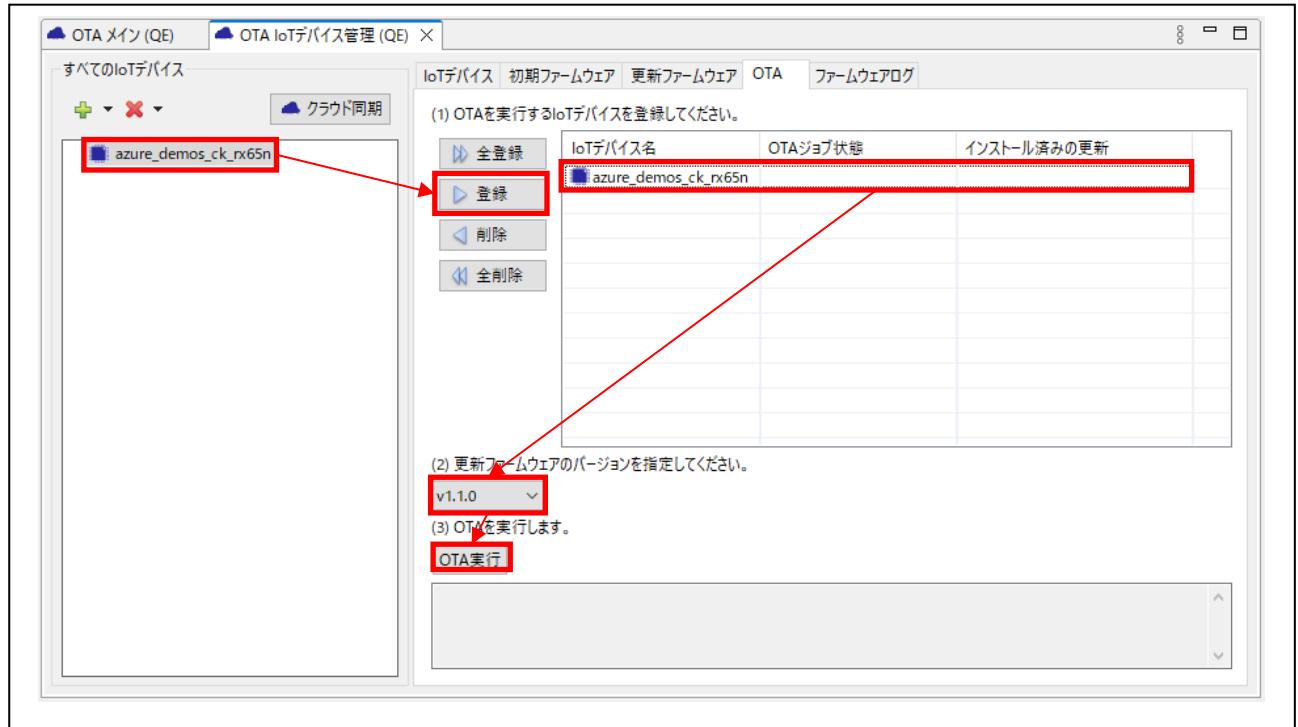


図 4-62 OTA 設定 (Azure)

## 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

自動的に OTA 実行ジョブが作成され、AWS の場合は OTA ジョブ状態が “IN\_PROGRESS” になります。Azure の場合は OTA ジョブ状態が “InProgress” になります。

この状態で IoT デバイス(ターゲットボード)が動作しており、ネットワークに接続されていると、更新を検知し、ファームウェアのアップデートが自動的に開始されます。IoT デバイスがネットワークに接続されていない場合は、ネットワーク接続後にファームウェアのアップデートが開始されます。

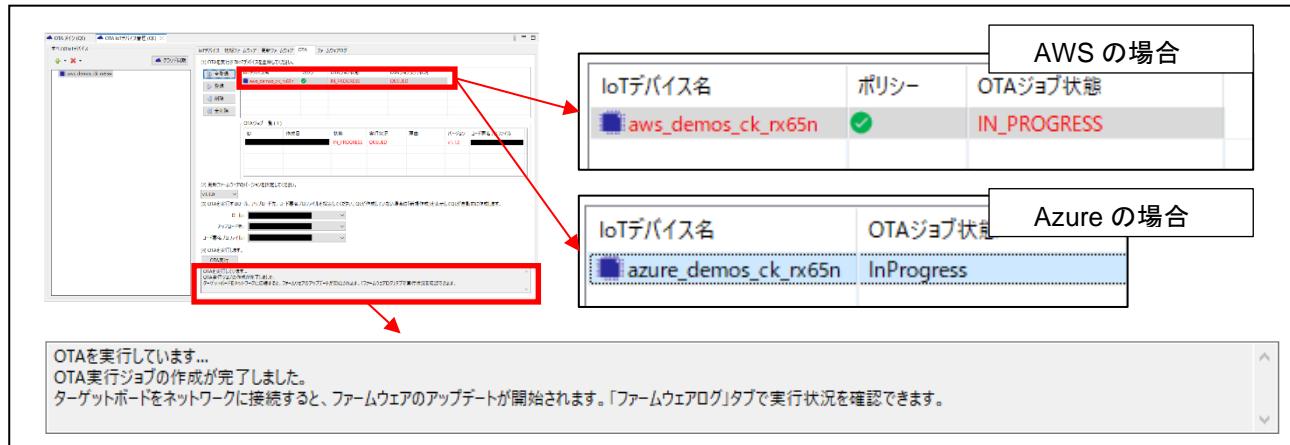


図 4-63 OTA 実行後

進行状況は、[ファームウェアログ]タブで確認できます。

OTA メイン(QE) ビューの左側に表示されているメニューから[OTA の実行と状況確認]を選択し、[ファームウェアログの確認]の横にある[ビューを開く]ボタンを押下します。

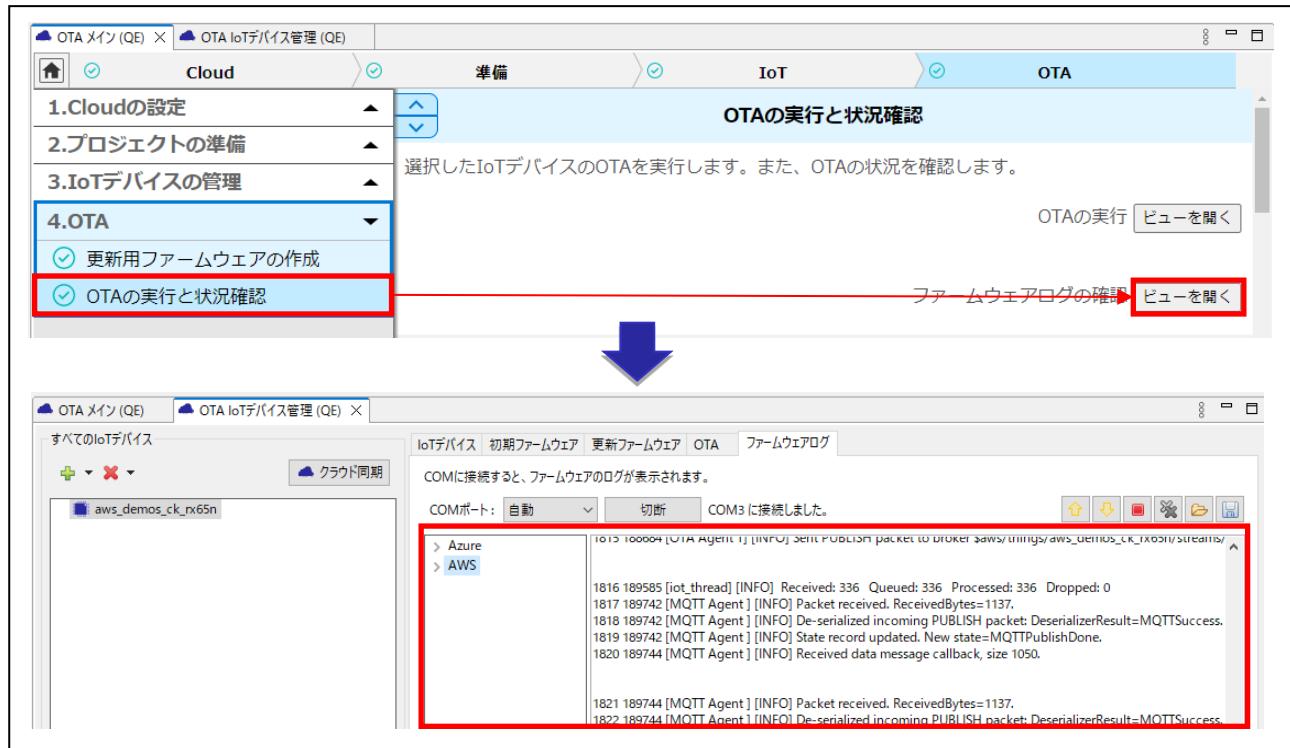


図 4-64 ファームウェアログの確認

ダウンロード完了後、ファームウェアのチェックが実施され、問題がなければ IoT デバイスが再起動して、更新後のファームウェアが動き出します。

ファームウェアログでバージョンの表記を確認することで、OTA が成功したかを確認することができます。ファームウェアログの左側のメニューで、内容を検索すると探しやすいです。

AWS の場合 : [AWS > OTA > Version (x)]

Azure の場合 : [Azure > Version (x)]

バージョン表記が OTA の前後で異なり、更新されていることを確認します。

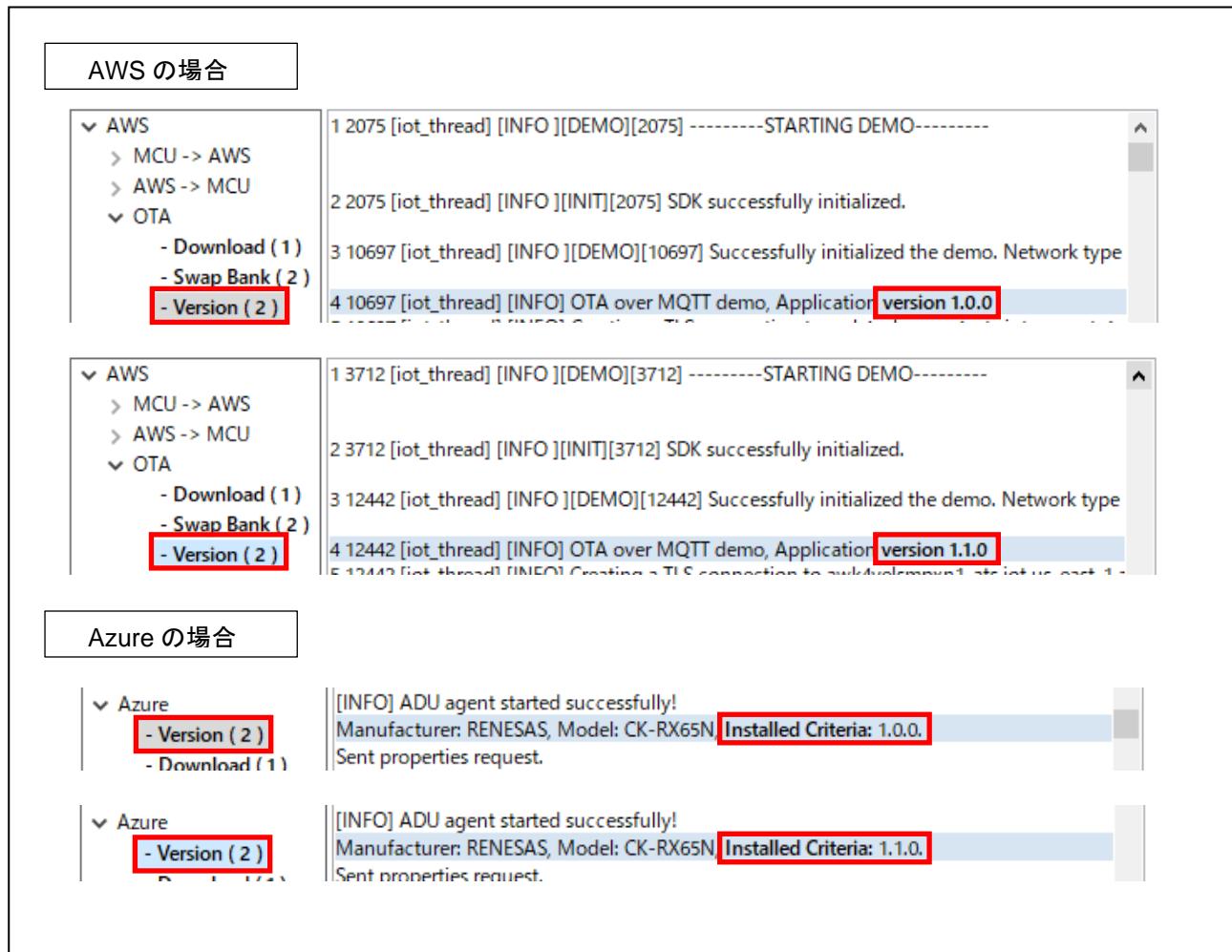


図 4-65 ファームウェアログ

更新完了後、[OTA]タブに戻ると、クラウド側の OTA ジョブが完了したことを確認できます。

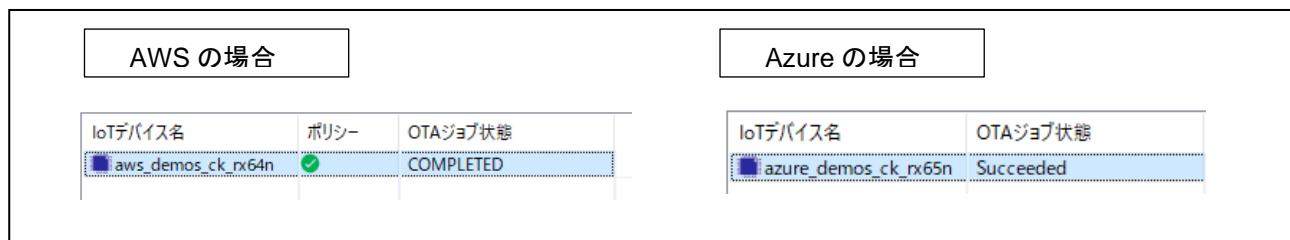


図 4-66 OTA 実行後のジョブ状態

## 5. ヘルプ機能

e<sup>2</sup> studio のヘルプから、QE for OTA の機能や使い方を確認することができます。

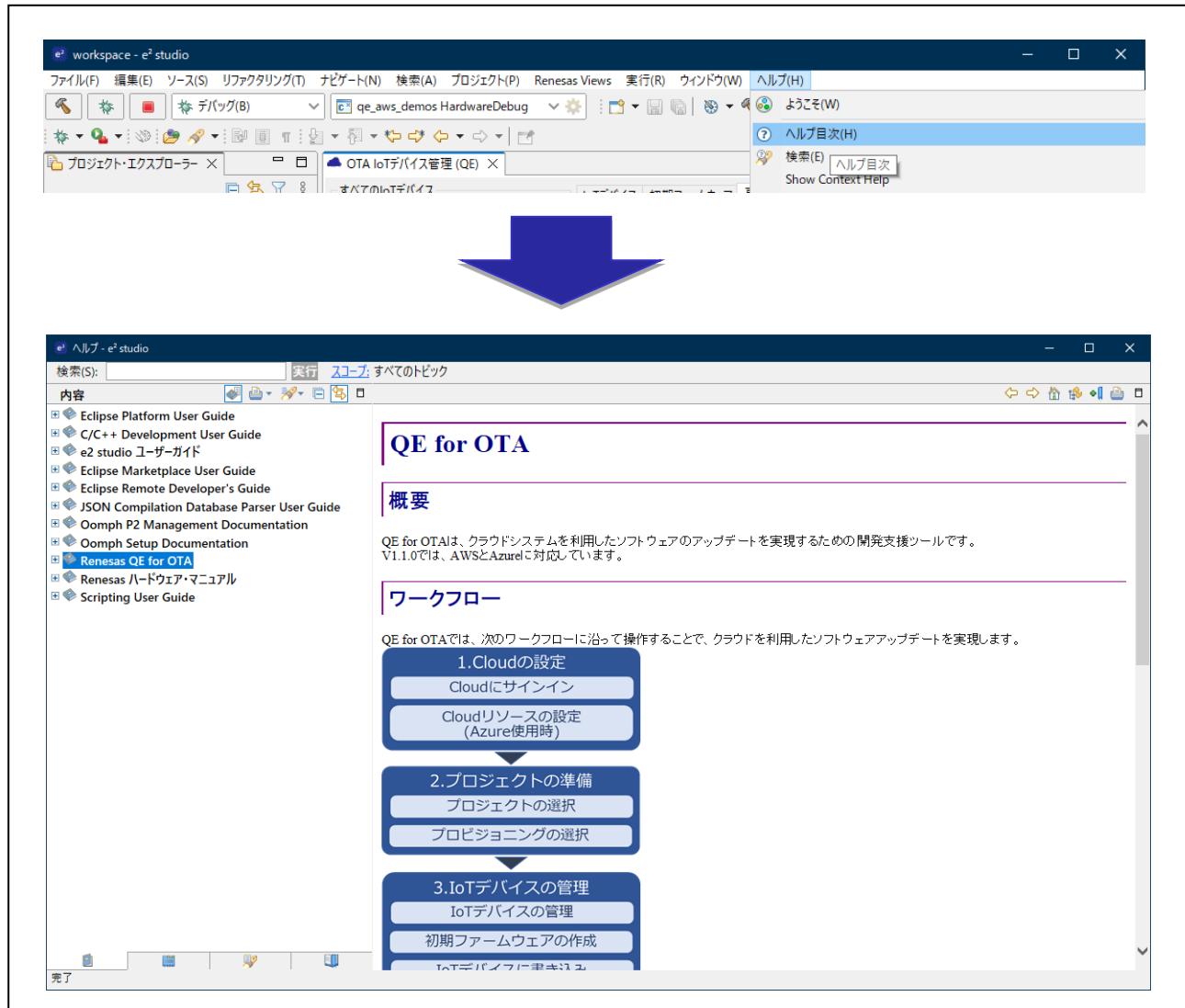


図 5-1 ヘルプ機能

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	Jun.30.23	-	初版

## 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、 $V_{IL}(\text{Max.})$ から $V_{IH}(\text{Min.})$ 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、 $V_{IL}(\text{Max.})$ から $V_{IH}(\text{Min.})$ 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うと、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、ノイズ幅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

1. 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みます。以下同じです。)に關し、当社は、一切その責任を負いません。
  2. 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、何らの保証を行ふものではなく、また責任を負うものではありません。
  3. 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。
  4. 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。
  5. 当社製品を、全部または一部を問わず、改造、改変、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、改変、複製、リバースエンジニアリング等により生じた損害に關し、当社は、一切その責任を負いません。
  6. 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)、交通制御(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。
  7. 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、これに限りません。)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。
  8. 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。
  9. 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。
  10. 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關して、当社は、一切その責任を負いません。
  11. 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。
  12. 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いたします。
  13. 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。
  14. 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(豊洲フォレシア)

[www.renesas.com](http://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/](http://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。