

演習 2: Amazon FreeRTOS および AWS Greengrass の接続の確立

この演習では、MicroZed 上で動作する Amazon FreeRTOS および Ultra96 上で動作する AWS Greengrass の AWS クラウドへの基本的な接続を確立します。そのためには、AWS IoT コアへの認証に使用する一意のデバイス ID を確立する必要があります。これには、クライアント証明書と秘密キーを使用します。AWS アカウントでこれらの認証情報を作成し、各デバイスに対してこれらを設定する必要があります。

AWS IoT 認証情報を設定し、MicroZed の microSD カードへデプロイする

このセクションでは、AWS IoT コアの認証情報を設定およびデプロイし、MicroZed の microSD カードへコピーします。各デバイスの物理的な認証情報ファイル、秘密キー、および証明書は `$HOME/aws-cloud-and-xilinx-workshop/edge/auth-PREFIX-node-zynq7k` および `$HOME/aws-cloud-and-xilinx-workshop/edge/auth-PREFIX-gateway-ultra96` に格納します。PREFIX は、Greengrass グループと S3 バケットの接頭辞です。

1. MicroZed ボードの電源がオフであること (J2 と J7 のいずれにも USB ケーブルで電源が供給されていないこと)を確認します。
2. MicroZed に microSD カードが挿入されている場合はカードを取り出し、USB microSD カードリーダーに装着してから Ultra96 に挿入します。
 - i. MicroZed から microSD カードを取り出します。microSD カードをスロットに向かって押し込み、「カチッ」と音がしたら手前に戻してカードを引き出します。
 - ii. microSD カードを USB-SD カード アダプターに装着します。
 - iii. このアダプターを Ultra96 ボードの USB ポートに挿入します。5 ~ 10 秒ほど待ちます。
3. Ultra96 のデバッグ インターフェイスで、クラウド オブジェクトをデプロイするスクリプトがあるディレクトリに移動します。

```
cd $HOME/aws-cloud-and-xilinx-workshop/cloud/script
```

4. AWS IoT を利用して AWS アカウントに接続できるようにデバイスの認証情報を設定するスクリプトを実行します。このワークショップで使用するエッジ ハードウェアは、AWS アカウント内でグループ接頭辞によって一意に識別されます。これにより、複数のテーブルで 1 つのコーポレート AWS アカウントを共有する場合でも、それぞれが独自のハードウェアを使用して作業できます。接頭辞は、前の演習で使用したのと同じものを使用します。

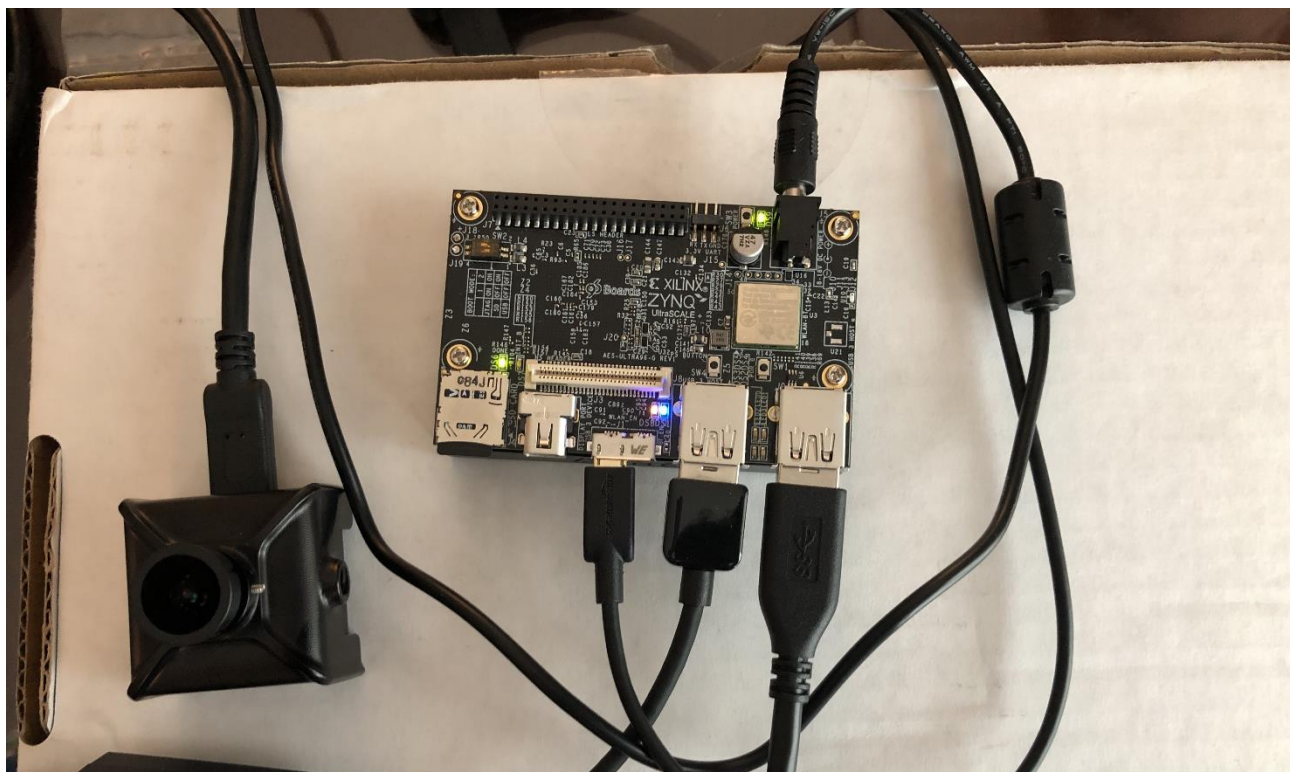
```
./deploy-awsiot-objects.sh <prefix>
```

スクリプトの実行が完了すると、キーと証明書が上記のディレクトリに保存されます。また、必要なファイルもすべてこのスクリプトによって MicroZed の microSD カードへ直接コピーされます。このスクリプトによって microSD カードに書き込まれる MicroZed プラットフォーム用ファイルは、次のとおりです。

- BOOT.bin
- PREFIX-node-zynq7k.crt.der
- PREFIX-node-zynq7k.key.prv.der
- ggconfig.txt

BOOT.bin には、MicroZed 上で実行されるアプリケーション、および関連するハードウェア デザインが含まれます。認証情報ファイルは、プラットフォームから事前定義済みの MQTT メッセージをサブスクライブできるように、デバイスとアカウントをリンクします。ファイル `ggconfig.txt` には、ブローカーエンドポイント情報、およびアプリケーションで使用する PREFIX が記載されています。

5. Ultra96 の USB ポートから USB アダプターを取りはずします。
6. USB-SD アダプターから microSD カードを取りはずします。
7. microSD カードを端子面を上にして MicroZed ボードのスロットに挿入します。「カチッ」と音がするまでカードをゆっくりとスロットに押し込んでから、指を離します。
8. Greengrass グループをデプロイする前に、演習 4 および 5 で使用する eCon USB カメラを Ultra96 ボードの J8 に接続しておきます。Ultra96 にカメラを接続すると、次のようになります。



ザイリンクス Ultra96 上で AWS Greengrass を設定、デプロイする

Ultra96 は、Linux と AWS Greengrass を実行します。Ultra96 を Greengrass コアとして使用できるように、次の手順を実行して Ultra96 ボードの認証情報を準備します。

1. 秘密キーと証明書を AWS Greengrass にコピーします。

```
sudo cp $HOME/aws-cloud-and-xilinx-workshop/edge/auth-*gateway-ultra96/*pem  
/greengrass/certs/
```

2. AWS Greengrass の設定ファイル `config.json` を AWS Greengrass のインストール ディレクトリにコピーします。

```
sudo cp $HOME/aws-cloud-and-xilinx-workshop/edge/auth-*gateway-
ultra96/config.json /greengrass/config/
```

3. 証明書と設定ファイルのコピーが完了したら、AWS Greengrass コア サービスを開始できます。USB カメラが正しく接続されていることをもう一度確認してから、Ultra96 のターミナル ウィンドウで次のコマンドを実行します。

```
sudo /greengrass/ggc/core/greengrassd start
```

Greengrass デーモンが起動すると、CLI に「Greengrass successfully started with PID: XXXX」という応答が表示されます。

4. 次に、このワークショップで使用する AWS Greengrass エッジ Lambda 関数をいくつかビルドします。まず、`xilinx-hello-world` という名前の AWS Lambda 関数をビルドしてアップロードします。

```
cd $HOME/aws-cloud-and-xilinx-workshop/cloud/script
./make-and-deploy-lambda.sh xilinx-hello-world
```

出力を見るとわかるように、このスクリプトは次の処理を実行しています。

- ロールが存在しない場合、この関数に対するロールを作成する。
- `$HOME/aws-cloud-and-xilinx-workshop/cloud/xilinx-hello-world` にあるコードを zip ファイルに圧縮する。
- コードを AWS Lambda サービスにアップロードする。
- 関数にバージョン番号を付与する。
- 関数のエイリアスを作成する。

5. この演習では AWS Lambda 関数を 1 つしか使用しませんが、この後の演習で使用する AWS Lambda 関数も、次のコマンドを実行してビルドしておきます。

```
./make-and-deploy-lambda.sh xilinx-bitstream-deploy-handler
./make-and-deploy-lambda.sh xilinx-image-upload-handler
./make-and-deploy-lambda.sh xilinx-video-inference-handler
./make-and-deploy-lambda.sh aws_xilinx_workshop_core_shadow_proxy_handler
./make-and-deploy-lambda.sh aws_xilinx_workshop_intelligent_io_error_handler
./make-and-deploy-lambda.sh aws_xilinx_workshop_telemetry_enrichment_handler
./make-and-deploy-lambda.sh aws_xilinx_workshop_aws_connectivity_handler
```

6. AWS Greengrass グループの初期設定を作成します。この手順を短時間で完了できるように、グループの作成は自動化されています。

ここで使用する接頭辞は、演習 1 で使用した Amazon S3 バケットの接頭辞と同じです。

```
cd $HOME/aws-cloud-and-xilinx-workshop/edge/script
./greengrass-core-init.sh <prefix>
```

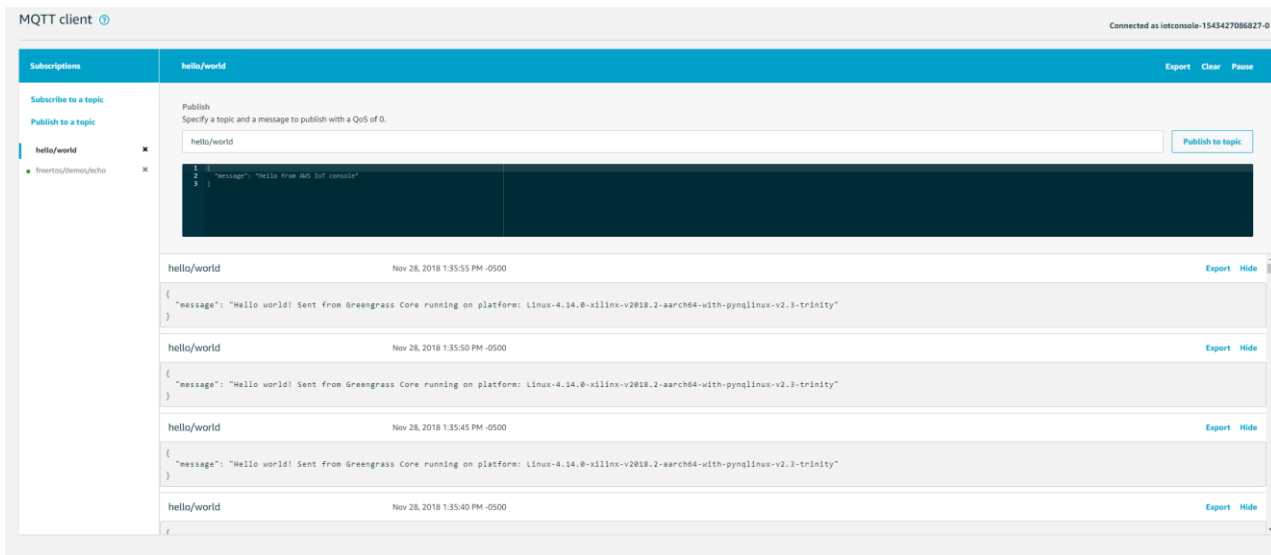
7. AWS Greengrass の最初のデプロイを実行します。

Ultra96 のターミナル ウィンドウで次のコマンドを実行します。

```
cd $HOME/aws-cloud-and-xilinx-workshop/cloud/script
./deploy-greengrass-group.sh <prefix>
```

数秒後に、グループが正しくデプロイされたことを確認できます。

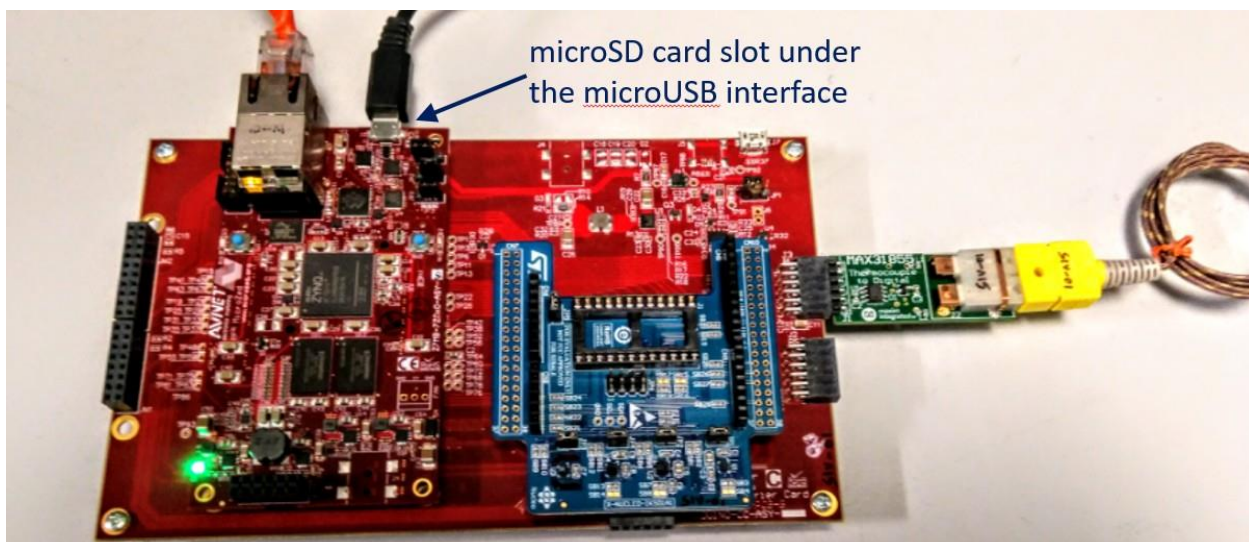
8. AWS IoT コンソール ページにアクセスし、左側メニューの **[Test]** をクリックします。
9. **[Subscriptions]** ヘッダーの下にある **[Subscribe to a topic]** をクリックします。
10. **[Subscription topic]** に「hello/world」と入力します。
11. **[Subscribe to topic]** をクリックします。テスト ウィンドウに、Ultra96 プラットフォームからの MQTT 応答が表示されます。次に、この応答を示します。



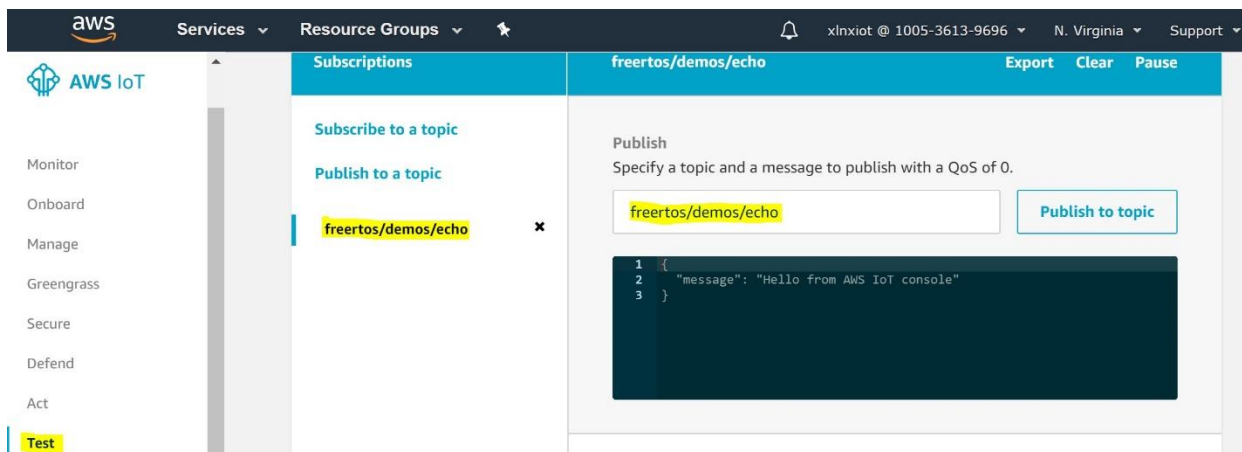
ザイリンクス Zynq-7010 上で Amazon FreeRTOS を設定、デプロイする

MicroZed デバイスは、先に準備した microSD カードから Amazon FreeRTOS をブートして、AWS クラウドに接続します。

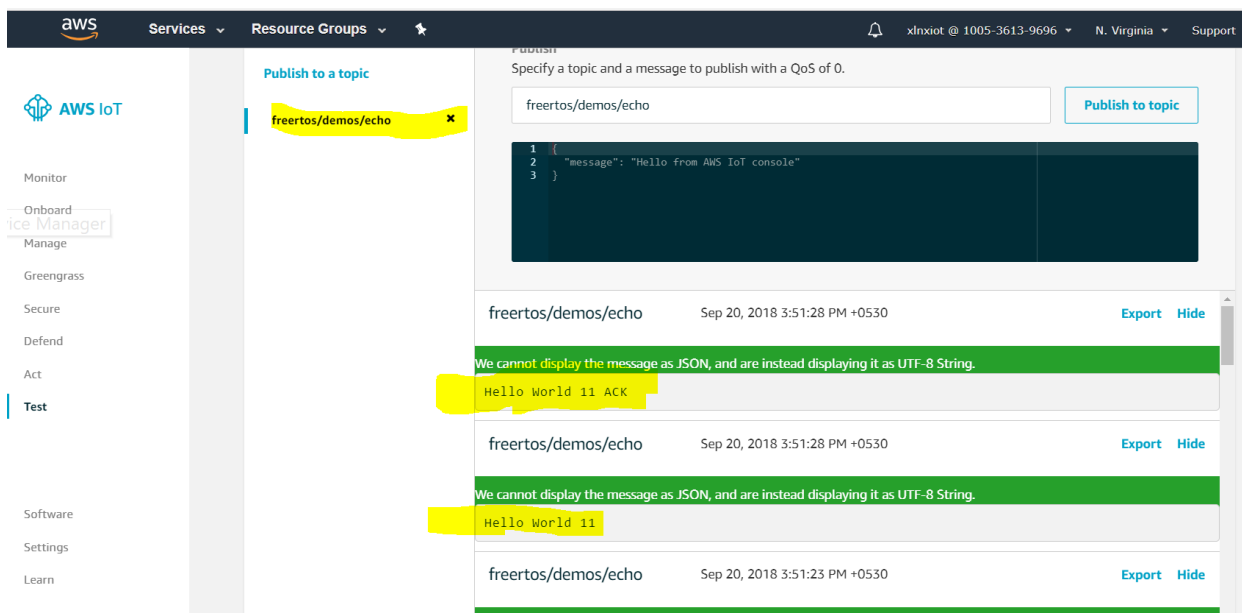
1. USB アダプターから microSD カードを取りはずし、MicroZed ボードに挿入します。microSD カードスロットの位置は、次の図を参照してください。



- J2 と J7 にそれぞれ USB-microUSB ケーブルを接続し、MicroZed に電源を供給します。
- これら 2 本の電源ケーブルを接続したら、SoM の RST/SW2 ボタンを押して MicroZed をリセットします。
- AWS IoT コンソール ページにアクセスし、左側メニューの **[Test]** をクリックします。
- [Subscriptions]** ヘッダーの下にある **[Subscribe to a topic]** をクリックします。
- [Subscription topic]** に「freertos/demos/echo」と入力します。
- [Subscribe to topic]** をクリックします。



- テスト ウィンドウに、MicroZed 上で動作する Amazon FreeRTOS からの MQTT 応答が表示されます。次に、この応答を示します。



まとめ

この演習では、MicroZed プラットフォーム上の Amazon FreeRTOS IoT ノード、および Ultra96 プラットフォームの組み込み Linux 上で動作する AWS Greengrass から、基本的な「hello world」接続を確立しました。

[次の演習](#)

[トップ ページ](#)

この資料は英語版を翻訳したもので、内容に相違が生じる場合には原文を優先します。資料によっては英語版の更新に対応していないものがあります。日本語版は参考用としてご使用の上、最新情報につきましては、必ず最新英語版をご参照ください。