

ESP32でサッ! AWSとAmazon FreeRTOSのセキュア接続

古城 隆

ご購入はこちら

● トライすること

実際のIoTデバイスは、実世界のモノをリアルタイムに制御したりデータを取り込んだりします。その一方で、ネットワーク通信やセキュリティ処理、あるいは画像処理やAIといった時間のかかる処理も混在していて、これを並行して実行することが求められます。このようなときに、中心的な役割を果たすのがRTOSです。

RTOS上で動作するIoTデバイスの例として、周期的にデータを収集するシンプルな処理と、クラウド・サーバを安全に結ぶネットワーク・プロトコルの2つの処理をするサンプル・プログラムを作成しました。本章では、3つのサンプル・プログラムを使ってIoTデバイスのプログラミング手法について解説します。

サンプル・プログラムは、クラウド・サーバにAWS IoT、デバイスに写真1のESP32-WROOM-32 (Espressif Systems)を使います。Amazon FreeRTOSは、ESP32-WROOM-32上で動作させます。実行の状況は、コンソール出力と、Wiresharkのパケット・キャプチャで見ることができます。

● 本章で解説すること

▶ (1) データ取得と表示

最初のサンプル・プログラム (V1) は、10msごとの短い周期で簡単なデータを周期的に取得して、それをコンソールに表示します。

一見、簡単に見えますが、これを1つのプログラムで実現するのは難しいです。周期の短いデータ収集の処理は、比較的時間の掛かるコンソールへのプリントアウト処理に邪魔されます。このため、プリントアウト中は周期処理が進みません。

▶ (2) タスクに分けて処理する

2つ目のサンプル・プログラム (V2) では、データ収集処理とプリントアウト処理をAmazon FreeRTOS上の2つのタスクに分けて処理させてみます。

これにより、データ収集の周期性がきちんと一定に保たれつつ、リアルタイム性が実現することを確認します。

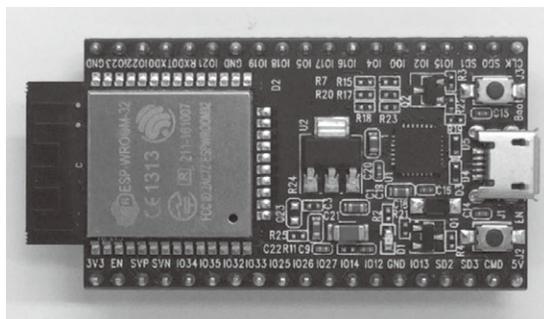


写真1 本章でやること…ESP32-DevKitCを使って Amazon FreeRTOSとAWS IoTをつないでみる

ESP32-WROOM-32 (Espressif Systems) を搭載している。開発環境の構築手順はAWSのウェブ・ページで公開されている

▶ (3) クラウドにデータを送信する

3つ目のサンプル・プログラム (V3) では、プリントアウト処理タスクをネットワーク送信処理に置き換え、収集したデータをクラウドのMQTTブローカに送信します。

ブローカに送信されたデータは、同じブローカにサブスクライブしているMQTTコンソールにデータが表示されます。これで2つ目のサンプル・プログラムと同じ処理をネットワークを介して実現したことになります。

▶ (4) セキュリティの仕組み

本稿の後半では、3つ目のサンプル・プログラムのネットワーク上の処理を少し詳しく見ながら、どのようにセキュリティを実現しているかを解説します。

ネットワークのセキュリティは、TLS (Transport Layer Security) と呼ばれるプロトコルで実現されています。

セキュリティと聞くと、他人に情報を盗まれないための暗号化 (秘匿性) をまず思い浮かべます。その他にも、データの改ざんを防ぐ一貫性のチェックや、通信の相手方のなりすましを防ぐ認証なども重要なセキュリティ機能です。本稿では、IoTデバイスで特に重要なデバイス認証を中心に見ていきます。