

TOPPERS 活用アイデア・アプリケーション開発 コンテスト

部門 : アプリケーション開発部門

作品のタイトル : TOPPERS ASP を使用した Azure IoT アプリ

作成者 : 手塚湧太郎

共同作業者 : 本田晋也, 長島宏明

対象者 : TOPPERS OS を用いて Azure IoT アプリ開発をお考えの方

使用する開発成果物 : TOPPERS ASP カーネル

目的・狙い

組込みシステムにおいて、IoT 技術が注目されている。IoT を活用するプラットフォームの 1 つとして「Azure IoT」が注目されており、Azure IoT 活用したアプリケーション開発を通し、TOPPERS OS を用いた IoT アプリケーション開発のさらなる可能性を示すことを狙いとしている。

アイデア/アプリケーションの概要

本アプリケーションは、Wio Terminal を用いて、ZUMO による走行制御を行うとともに、その走行データの一部を Azure IoT 上に蓄積するアプリである。走行モードやデータ送信モードの管理を TOPPERS ASP によって実行している。TOPPERS OS による Azure IoT アプリケーション開発手法について示すことができた。

アプリケーションの概要

- 方眼上のコースをコマンドに従い走行し、データをAzureに送信するロボット
 - リアルタイムOS
 - TOPPERS ASP3
 - メインタスク
 - 各setup
 - 走行コマンドの受け取り
 - 走行モードと送信モードを管理
 - 機能1(走行モード)
 - Zumoの走行を管理
 - コマンドを受け取り, ライン上を走行する
 - 機能2(送信モード)
 - Azure IoT Hubを利用しZumoセンサーデータをクラウドへ送信
 - ジャイロセンサーのデータを送信する

使用環境

- 以下のサイトを参考にAzure環境を用意した
 - <https://matsujirushi.hatenablog.jp/entry/2020/08/10/222132>
- Azureサブスクリプション
- Azure Portalから以下を作成
 - Azure IoT
 - Azure IoT Hub
- Azure IoT Explorer
 - メッセージ確認用に使用

動作環境

- Arduino • Arduino IDE
 - 組み込みシステムのプログラミング環境とボードコンピュータ
 - C++に近い言語でプログラミング
- Wio terminal
 - Arduino互換のマイクロコントローラ
 - 5方向スイッチ
 - WiFi接続
- ZUMO
 - 各種センサーを搭載した移動ロボット
 - LED
 - 加速度センサ
 - ジャイロセンサ
 - 電子コンパス
 - Arduinoにより制御
 - プログラムにより
 - モータの制御
 - センサの読み込み

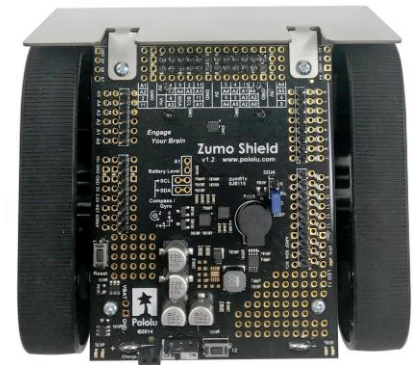
Wio terminal



Arduino



ZUMO

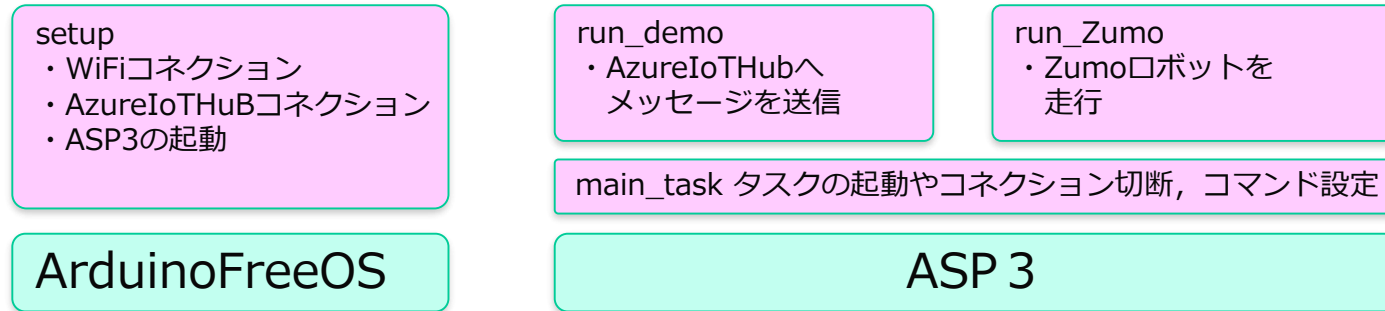


アプリケーション構築

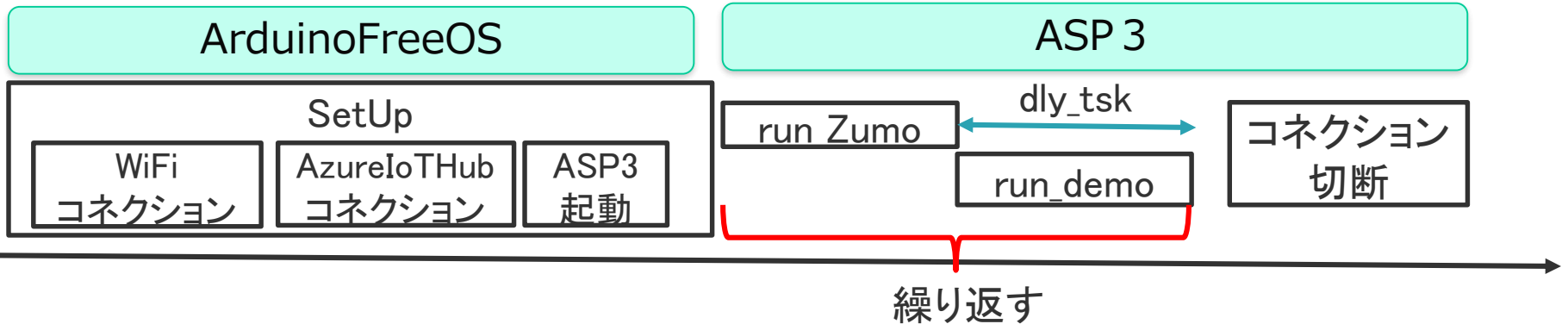
- 構築のためダウンロードしたもの
 - Wio TerminalのAzure IoT 接続サンプル
 - <https://github.com/Azure/azure-iot-arduino>
 - Wio TerminalのTOPPERS/ASP3サンプル
 - https://github.com/exshonda/Arduino_TOPPERS_ASP
 - Wio TerminalのZUMO管理ライブラリ
- アプリケーション開発
 - TOPPER ASP3サンプルに作成したrun_zumo関数とAzureサンプルのrun_demo関数をタスクとして登録
 - run_zumo : ZUMOによる走行を実行する関数
 - run_demo : Azure IoTにメッセージを送信する関数
 - メモリ不足やタスクIDの割り当て方, タスクルーチン開始のタイミングなどに注意した

アプリケーション構成

- アプリケーション構成



- アプリケーション実行の流れ



- 今回使用したサンプルではWiFiコネクションの確立はArduinoFreeOSを使用する必要があるため、確立してからASP3によるタスクルーチンを開始するようにした

動作手順

- Arduino IDEにライブラリを追加
 - ライブラリの管理で、下記を検索してインストール
 - Seeed_Arduino_rpcWiFi(Ver.1.0.4)
 - Seeed_Arduino_rpcUnified(Ver.2.1.3)
 - Seeed_Arduino_FS(Ver.2.0.3)
 - Seeed_Arduino_SFUD(Ver.2.0.1)
 - Seeed_Arduino_mbedtls(Ver.3.0.1)
 - AzureIoTHub (Ver.1.6.0)
 - AzureIoTUtility(Ver.1.6.1)
 - AzureIoTProtocol_MQTT(Ver.1.6.0)
 - AzureIoTProtocol_HTTP (Ver.1.6.0)
 - zip形式のライブラリをインクルードからインストール
 - TOPPERS_ASP
 - TOPPERS_ASP_FreeRTOS_API
 - ZumoShieldN
 - library.zipに添付

動作手順

- ビルドオプションの変更
 - platform.txtを書き換える
 - 従来のアプリの場合は以下にある
 - C:¥Users¥<yourusername>¥AppData¥Local¥Arduino15¥packages
 - compiler.cpp.extra_flags=に以下のオプション指定
 - -DONT_USE_UPLOADTOBLOB

動作手順

- WiFiの設定
 - ファイルを開くからTOPPERS_ASP_and_Azure.inoを開く
 - タブをiot_config.hに切り換え
 - 下記の部分を使用する環境に合わせて, 変更
 - DEVICE_CONNECTION_STRINGをAzure IoTから取得した接続文字列に変更

```
10 #define IOT_CONFIG_WIFI_SSID          "yourSSID"
11 #define IOT_CONFIG_WIFI_PASSWORD     "yourPASS"
12
13 /**
14  * IoT Hub Device Connection String setup
15  * Find your Device Connection String by going to your Azure portal, creatin
16  * navigating to IoT Devices tab on the left, and creating (or selecting an
17  * Then click on the named Device ID, and you will have able to copy the Pri
18  */
19 #define DEVICE_CONNECTION_STRING     "your-iothub-DEVICE-connection-string"
```

- マイコンボードに書き込みで実行開始

デモ動画

Azure IoT Explorerでのメッセージ

Tue Jun 22 2021 18:36:37 GMT+0900 (日本標準時):

```
{  
  "body": -93,  
  "enqueuedTime": "Tue Jun 22 2021 18:36:37 GMT+0900 (日本標準時)"  
}
```

Tue Jun 22 2021 18:36:27 GMT+0900 (日本標準時):

```
{  
  "body": -87,  
  "enqueuedTime": "Tue Jun 22 2021 18:36:27 GMT+0900 (日本標準時)"  
}
```

Tue Jun 22 2021 18:36:14 GMT+0900 (日本標準時):

```
{  
  "body": 0,  
  "enqueuedTime": "Tue Jun 22 2021 18:36:14 GMT+0900 (日本標準時)"  
}
```

Tue Jun 22 2021 18:36:02 GMT+0900 (日本標準時):

```
{  
  "body": 88,  
  "enqueuedTime": "Tue Jun 22 2021 18:36:02 GMT+0900 (日本標準時)"  
}
```

Tue Jun 22 2021 18:35:52 GMT+0900 (日本標準時):

```
{  
  "body": 92,  
  "enqueuedTime": "Tue Jun 22 2021 18:35:52 GMT+0900 (日本標準時)"  
}
```

Tue Jun 22 2021 18:35:40 GMT+0900 (日本標準時):

```
{  
  "body": -3,  
  "enqueuedTime": "Tue Jun 22 2021 18:35:40 GMT+0900 (日本標準時)"  
}
```

Tue Jun 22 2021 18:35:28 GMT+0900 (日本標準時):

```
{  
  "body": -86,  
  "enqueuedTime": "Tue Jun 22 2021 18:35:28 GMT+0900 (日本標準時)"  
}
```

Tue Jun 22 2021 18:35:16 GMT+0900 (日本標準時):

```
{  
  "body": 0,  
  "enqueuedTime": "Tue Jun 22 2021 18:35:16 GMT+0900 (日本標準時)"  
}
```

走行の様子

- <https://www.dropbox.com/s/fdya3kb4g2ioota/iOS%E3%81%AE%E7%94%BB%E5%83%8F.MOV?dl=0>

センサー値	入力コマンド
0	F
-86	R
-3	L
92	L
88	F
0	R
-87	R
-93	F