

ESP32 を用いた教材開発

—IoT 開発教材作成についての検討—

電子情報技術科 末永 聖平

Teaching material development using ESP 32

— Consideration about preparation of IoT development teaching materials —

Shohei SUENAGA

概要 現在、様々な物が IoT (Internet of Thing) 化されている。これを実現するためには、ネットワーク、ハードウェア、ソフトウェアなどの総合的な技術力が必要となる。そこで、電子情報技術科の学生に対しどのような開発課題が良いかを検討する。開発検討した結果として、科の実技科目の中に取り入れ実施していく予定である。

1. はじめに

現在、中国職業能力開発大学校附属島根職業能力開発短期大学校（以下、島根校）の電子情報技術科では、ネットワーク、ハードウェア、ソフトウェアなどの科目が実施されている。しかし、それぞれの科目の内容があまり密接に結びついていない部分がある。今日では、これらが融合させたものモノの開発がなされ、様々な製品が登場している。今後もこの流れが変わることはないだろう。このような製品は IoT (Internet of Thing) と呼ばれ、今までインターネットと接続する必要がないと思われていた製品が次々と接続され始めている。例としては、外出時、自宅ドアが開けられると自動的にメール送信、通報などが行われたり、消耗品がなくなる前に自動的に注文をしたりなど、様々な活用がなされている。企業においても AI (人工知能) と並び主要技術とされている。

このような技術分野の人材が求められており、この要素における技術教育が必要となっている。

この IoT を実現するためには、ネットワーク、ハードウェア、ソフトウェアなどの複数の技術を組み合わせる必要がある。そこで、学生たちにこれらを組み合わせる技術を養うための教材開発について検討する。

2. 教材の選択

IoT 開発を行うにあたりどのような教材を用いて開発を行うかの検討に入った。学生全員が同一の教材を用いて開発を行うにあたり、安価であり、開発が比較的容易であるもの検討した。選択肢として、Raspberry Pi、Arduino と限定することとした。どちらも条件に一致し開発も行いやすい等の利点を有している。実際に開発を行い、比較検討した結果、Arduino を利用したものがよいという結論に達した。理由としては、Raspberry Pi では開発環境を構築し、開発開始までの前準備が多く、学習を予定している学生に Linux 操作なども含めて講義をする必要がある。これらの技術も学生たちが就職後に必要となる技術ではあるが、開発を開始するまでに時間がかかりすぎてしまうため、今回は採用を見送ることとした。

Arduino に関しては、学生にとっては授業において開発経験があり、開発を開始するまでの時間を短縮することができる。Arduino には無線 LAN などの IoT を行う上での必要となる無線機能が搭載されていないため、Espressif 社が販売している ESP-WROOM-32 を選択し、これを教材とすることとした。

この機器には、無線 LAN、Bluetooth、Bluetooth Low Energy (以下、BLE) といった、現在の IoT 機器で用いられている無線技術が事前に搭載済みである

からである。ESP-WROOM-32 は、単体で開発を行うには、配線も必要となるため、これを搭載した ESP32-DevKitC 基板を選択した（写真 1）。

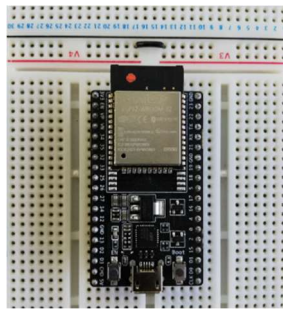


写真 1 ESP32-DevKitC

ESP32-DevKitC は、現在授業で利用している Arduino IDE 開発環境に追加インストールを行うだけで開発が開始でき、容易に IoT 開発を体験させることが可能である。

開発を行うマイコンボードが決定したので、制御対象として、LED、小型モータ、各種センサ、押しボタンスイッチ等を利用することとした。この制御対象についても学生は、授業において取り組んだことがあるからである。これらはすべて島根校内の既存機器等を活用する。

3. 環境構成

予定している実習は、週に 1 回 200 分、連続 9 週で行う。実習を受講する学生は、ネットワーク、ハードウェア、ソフトウェアに関する講義、実習は事前に受講済みであるため、この前提に合わせた教材開発を行う。9 回の講義内容を以下である。

1 週目	開発環境の構築、ハードウェアの準備
2 週目	ハードウェアの構成、接続
3 週目	ソフトウェア開発
4 週目 ～7 週目	ネットワークの構成、ソフトウェア開発
8 週目 ～9 週目	自由課題とし、今までの実習内容に基づき開発を行う

実習の目標としては、Arduino と LED、センサなどのハードウェアを正しく結線し、動作するプログラムが記述することができ、さらに、作成したものを

無線によって他の端末から制御することができることとする。

3.1. 開発環境の構築

ESP-WROOM-32 には、いくつかの開発方法が用意されているが、学生が使い慣れた Arduino IDE 開発環境を活用したものとした。インターネット接続環境を用意することができれば、一部の設定変更により、ESP-WROOM-32 の開発環境をダウンロードし、環境を整えることができる。

3.2. ハードウェアの構成

ハードウェア作成の流れは以下のである。

- ① LED 点滅制御
↓
- ② センサを利用した制御
↓
- ③ モータ制御
↓
- ④ ①～③の組み合わせ

授業が始まる前半は復習を兼ねて、ハードウェア、ソフトウェア開発の流れを確認していくところから実施する。

3.3. 無線ネットワークの構成

ESP32-DevKitC には、仕様上、無線機能として無線 LAN (IEEE802.11b/g/n)、Bluetooth (v4.2、BLE) が搭載されている。

学生は、IP アドレスを用いたネットワークについて理解していることが前提である。そこで、無線 LAN 環境を構築し、これを利用し ESP32-DevKitC を制御できるようにする。

ネットワーク構成において、本来であればインターネット接続環境を用意し、Amazon 社 AWS などのクラウド技術を取り入れた授業を実施することが理想的である。しかし、利用におけるライセンス、費用面、学内ネットワークの調整等の準備などが必要となる。これを実現することは、容易ではないため、概要等については説明を行うがインターネット、クラウド環境を利用しない範囲で構築することとした。

無線環境の構築としては、図 2 ようにする。ESP32-

DevKitC を無線アクセスポイント兼 Web サーバーとし、パソコンと接続し、パソコンからハードウェアを制御する。またその逆のハードウェアの状況をパソコンに送信する。

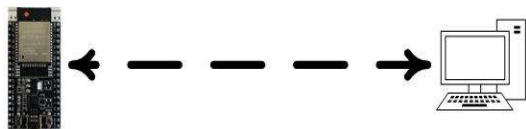


図2 ネットワーク環境1

併せて、図3のネットワーク構成も作成する。アクセスポイントを用意したうえで、ESP32-DevKitCにはWebサーバーの機能のみにした接続環境を構成する。これは、将来、アクセスポイントを経由しクラウド環境へ接続するために活用できるようにするためである。

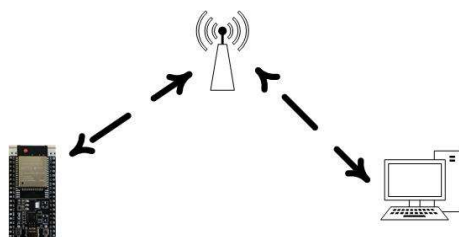


図3 ネットワーク環境2

Bluetooth に関する内容についても実施するとよいと思われたが、前提の授業ではBluetoothに関する授業は実施していない。そのため、内容が増えすぎると理解度にも影響が出ることも懸念される。今回の開発実習には触れる程度で抑えておくこととした。別の実習授業の際にBluetooth環境構築を試みることとする。

3.4. ソフトウェア開発の構成

ソフトウェア開発においては、基礎的な開発の方法について事前に学習していることを前提に進める。そこに、新たに無線IPネットワークを構成するために必要となるソフトウェア開発を行う。

4. ソフトウェア開発

基本となるLEDを点滅させる回路を作成する。そこにスイッチ、センサを利用し、センサでオン・オフを制御するものを作成する。

LEDに関して、たくさん使いたいが、配線が煩雑になるためフルカラーLEDを用意した。センサには、赤外線人感センサを用意した。

配線を写真2のように行うこととした。

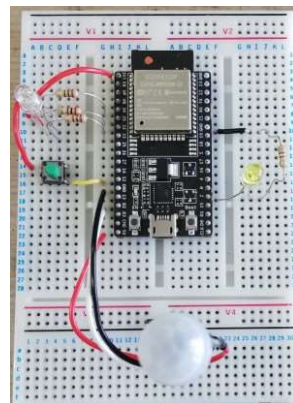


写真2 LED点灯、センサ回路

モータ制御には、小型DCモータ、サーボモータを採用することにした。モータに関しては、必要となる電流が大きくなることから別電源から電源供給を行うこととした。

基本的なプログラミングはすでに学習済みであるため、実習開始直後はおさらいを兼ねた実習とし、以下の流れに沿ってプログラミングを行う。

- ① LED点灯、点滅
- ↓
- ② スイッチを利用した点灯、点滅
- ↓
- ③ センサを利用した点灯、点滅
- ↓
- ④ モータ動作
- ↓
- ⑤ スイッチ、センサを利用した動作
- ↓
- ⑥ LED、モータを組み合わせた動作

配線と基本的なプログラムが終了するとESP32にネットワーク機能を追加する方法に関する内容となる。ネットワークに関してもある程度の知識を有していることが前提で、ネットワーク構築を指示することとし、以下の流れに沿ってプログラミングを行うこととした。

- ① ネットワーク環境構成についての確認
↓
- ② ESP32 に IP アドレスを割り振る
↓
- ③ ESP32 を Web サーバーにする
↓
- ④ クライアントがアクセスしてきた際の表示
↓
- ⑤ クライアントから操作により制御対象を動作

図 4 のようなクライアントから操作する画面を作成する。これには、簡単な HTML、CSS、JavaScript を記述して作成する。

<p>番号・名前</p> <p>IoT開発実習 操作画面</p> <hr/> <p>フルカラーLED制御</p> <p>消灯 <input type="button" value="赤色"/> <input type="button" value="緑色"/> <input type="button" value="青色"/></p> <hr/> <p>人感センサー状態</p> <p>反応なし</p> <hr/> <p>モーター制御</p> <p>停止 <input type="button" value="正転"/> <input type="button" value="逆転"/></p>

図 4 クライアントから Web 操作画面

ここまでのプログラミングを行うことができれば、学生自身で通信環境を用意し、対象となる制御対象を制御することができるはずである。

ここまでの内容について、資料をパワーポイントで作成し、配布、提示することとした。

5. 検証

当校 2 年生を対象に実施したが、すべての内容を実施することはできなかった。しかし、ネットワーク開発を行うプログラミングに関する実習を一通り行った。学生は、プログラムを記述し、目の前にあるハードウェアを制御するだけでなく、無線環境を利用して、遠隔操作が行えることに対して興味を持って取り組んでいた。今後は、さらに実習課題を増やし、学生がより興味を持って取り組めるようなものを検討する必要がある。

6. 問題点について

実際に実習を実施してみて、Web サーバー機能を持たせ、クライアントに表示を行うための HTML、CSS、JavaScript などについてまだ授業で実習を行っていなかったため表示部を作成する作業に、想定より多くの時間を費やすことになってしまった。併せて、この表示部を変更したいという要望に対して、個別に対応することになり、ハードウェアを制御するところに関心を持たせ切れていないような感じを受ける学生もいたように感じた。

7. まとめ

制御から通信までを行い、比較的興味や関心を持って実習を行ってくれたのではないかと考えられ、今後、開発が盛んになるであろう IoT に興味を持たせることができた。この実習を基にして、総合制作において新たな装置などを学生が検討し、開発することを期待したい。

今後の展開として、今回の実習内容としては採用しなかった Bluetooth を利用した制御方法を応用として取り入れ、また、学生のネットワークに関する知識・技術を向上させ、さらにインターネット接続環境を用意し、クラウド環境を用いた IoT 機器制御方法に関する実習、セキュリティに関する内容等を追加していきたいと考えている。

本実習を通して、今後必要となる技術が学生も理解できたはずである。これを今後のモノづくりの現場において生かされていくことを希望する。

参考文献

- 1) CQ 出版：2017.7月号 インターフェイス
- 2) CQ 出版：2017.11月号 トランジスタ技術
- 3) 日本経済新聞出版社：IoT 入門 三菱総合研究所

著者 E-mail Suenaga, Shohei[\$]jeed.or.jp