

IoTシステムのプロトタイピングに関する問題点の考察

徳田 瑛[†]
九州大学電気情報工学科[†]

林 健太 石田 繁巳 荒川 豊[‡]
九州大学システム情報科学研究院[‡]

1 はじめに

近年, IoT (Internet of Things) が広く世の中に普及しつつある. IoT は, 何らかの現象を測るセンシング部とそこから得られたデータを可視化, 分析するクラウドシステムという構成が一般的である. センシング部は, センサとそれを制御したり通信を行うマイコンボードから構成される. センサについては, 動きを計測する加速度センサーやジャイロセンサ, 環境を計測する温湿度センサや照度センサなどさまざまなものが簡単に入手できる. 複数のセンサを搭載した SenStick[1] や SensorTag のようなデバイスも登場している. 同様にマイコンボードも Arduino や Raspberry Pi など安価なものが普及している. また, これらをつなぐコネクタも Grove や Qwiic といった共通規格が広がり, 誰もが簡単にセンサ構築できるようになっている. そして, クラウド側も, Ambient や Azure IoT Central などさまざまな IoT 向けのサービスが広がっている.

しかしながら, 実際にプロトタイピングをしようとした場合, センサとマイコンボードを購入し, 無料のクラウドサービスのアカウントを作成するだけでは実現することはできず, ある程度, プログラミングの知識を必要とする. 本稿では, IoT システムのプロトタイピング手法について調査し, それらの特徴と課題についてまとめる.

2 プロトタイピングのプラットフォームの調査

ここでは我々が想定している IoT システムについて説明した後, そのシステムを実現する場合の現状と課題を説明する.

2.1 想定する IoT システム

図1に示したIoTシステムを想定する. センサが取得した値を, クラウド上へWi-Fiで送ることを想定する. 想定される経路は, Gatewayを介してクラウド上へ送る経路, もしくは直接基板からクラウド上へ送る経路である.

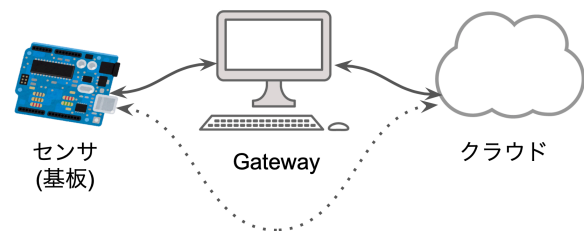


図1 想定する IoT システム

2.2 Arduino による開発

上記に書いたIoTシステムをArduinoで開発する場合は, C++でネットワークの設定やセンサの動作に関するプログラムを作成し, パソコンにシリアルケーブルで接続して, ボードに書き込むという作業が必要となる.

2.3 IoT Plug & Play

その問題に対して, MicrosoftはIoT Plug & Playというプラットフォームを提唱している. このプラットフォームでは, IoTセンサ部のマイコンにスマートフォンと接続に必要な機能が実装されており, 電源を入れるとスマートフォンからアクセス可能になる. そのため, デバイスの認定制度があり, 認定されたデバイスしか利用することはできない. しかしながら, 認定されたデバイスであれば, スマートフォンから簡単に設定を書き込むことができる. また, その設定は, データ送信のための通信設定も含め, クラウドサーバ上で作成することができ, すぐにクラウド上でデータを参照できる.

2.4 Javascript Robotics

もう1つ別のアプローチとして, 認定されたデバイスに限定することなく, プロトタイピングを簡単化する方法として, Javascript Robotics

Investigating a platform to prototype IoT systems

[†] Akira Tokuda, Kyushu University

[‡] Kenta Hayashi Shigemi Ishida Yutaka Arakawa, Graduate School and Faculty of Information Science and Electrical Engineering, Kyushu University

	基板	開発環境	初期設定	動作		
				動作	配置	拡張性
通常	Arduino	Arduino IDE	PC	固定	基板	高
IoT Plug & Play	認定 デバイス	Azure IoT Central	スマートフォン	固定	基板	低
Firmata + Johnny-Five	Firmata がインストールされた Arduino	node.js	PC	動的	クラウド (Johnny-Five)	中

表 1 手法の比較

と呼ばれるプラットフォームがある。近年、node.js など Javascript 言語が改めて注目されるようになっており、NodeBots^{*1}と呼ばれる、node.js でハードウェアを制御するイベントやコミュニティもできている。特に、C++ プログラミングを必要とする Arduino を Javascript で行えるようにしたもものとして Johnny-Five^{*2}がある。Johnny-Five では、クラウド側に Johnny-Five をインストールし、Arduino ボードに Firmata^{*3}と呼ばれるスケッチを書き込む。Firmata は、Arduino を Javascript から制御可能にするプロトコルであり、Arduino ボード上のセンサをクラウド側から Javascript の関数のように呼び出すことができるようになる。このプラットフォームのもう 1 つの特徴は、Arduino 側にはボードの動作についてプログラムを書かず、すべてクラウド側からボード上の機能呼び出して使う点である。そのため、取得したセンサ値に応じて、センシング頻度を変更するといったことが可能になる。しかしながら、Firmata がきちんと動作する Arduino ボードが少ないという問題がある。我々が調査した結果、Wi-Fi のチップが違う場合や拡張ボードがついている場合はほとんど動作しなかった。

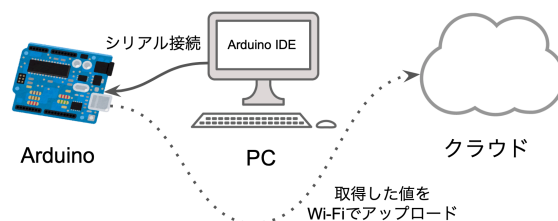
3 まとめと今後の課題

それぞれの手法の特徴についての比較を図 2 と表 1 に示す。これらの手法を比較した上で、それぞれの手法にはまだ改善の余地がある。改善点を踏まえた上で、うまくそれを補える方法や新たな手法を模索していくことが今後の課題と言える。

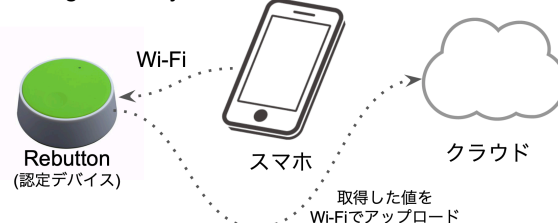
4 おわりに

Firmata と Johnny-Five を組み合わせた手法には、きちんと動作する基板が少ないことや、動作制御の度に通信が発生すること等の問題点もある。しかしながら、この手法は既存の手法

従来の方法



IoT Plug and Play



Firmata + Johnny-Five

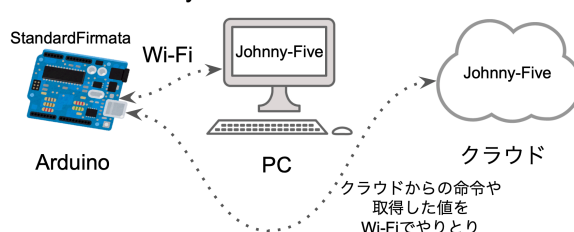


図 2 手法の比較

と比較して、柔軟性の観点で優れていると言える。一方、IoT Plug & Play は初期設定が非常に簡単である。今後は、両者の利点を活かした新たなプラットフォームを設計していく。

5 謝辞

本研究の一部は、内閣府が進める「戦略的イノベーション創造プログラム (SIP) 第 2 期 / フィジカル空間デジタルデータ処理基盤」(管理法人: NEDO) の助成で行われた。

参考文献

[1] Y. Nakamura, Y. Arakawa, M. Kanehira, T. and Fujiwara, and K. Yasumoto. Senstick: Comprehensive sensing platform with an ultra tiny all-in-one sensor board for iot research. *Journal of Sensors*, Vol. 2017, , 2017.

*1 NodeBots <https://nodebots.io>

*2 Johnny-Five <http://johnny-five.io>

*3 Firmata <http://firmata.org>