

ハードウェアを意識しない組み込み開発環境

— ハードウェアを API のように触れる組み込み開発環境 —

1. 背景

ハードウェア、ソフトウェアは別の技術であると認識され、その両方に深い知識があるエンジニアは少なかった。そのためエンジニアが課題の解決策を考えると、そのエンジニアがハードウェア、ソフトウェアどちらに詳しいかによって解決手法がハードウェアなのかソフトウェアなのかが大きく変わっていた。本来はその課題を解決するのにどちらが適切かを考えるべきである。

ハードウェア開発を伴わないソフトウェア開発は、あらかじめ決められたハードウェアで開発を行う。例えばスマートフォンアプリケーションを作成するにはスマートフォン(ハードウェア)の仕様は決まっているため、スマートフォンに気圧センサが内蔵されていないならば気圧センサを使用するアプリケーションは作成できない。

ソフトウェアエンジニアがハードウェアの知識を身につける、また、ハードウェアエンジニアがソフトウェアの知識を身につけるというのはどちらも現実的には難しい。そのため技術を API 化する必要がある。

2. 目的

本プロジェクトでは、ソフトウェアエンジニアがハードウェアを扱えるよう、ハードウェアを API 化するシステムの開発を行う。ソフトウェアエンジニアがハードウェアを手軽に扱えるようになれば、既存のソフトウェアアプリケーションにハードウェアが組み合わさった新たなアプリケーションが生まれてくる。

3. 開発の内容

本プロジェクトでは、従来は 1 つの基板上に複数のセンサを実装していたものを複数枚に分割し、それらにケースを付けたものを組み合わせることによってシステムを構築する。開発したプロダクトには「Base」と「Module」があり、センサ、アクチュエータなどを搭載するキューブ状のハードウェアを「Module」、スマートフォンと Module を接続するためのアダプターを「Base」と呼んでいる (Fig 1)。Module は Fig 2 のように Base に差し込んで使用する。

本プロダクトは Module と Base を組み合わせるという意味から、組み合わせを意味する「mCn」と命名した。



Fig 1 本プロダクトの Module と Base

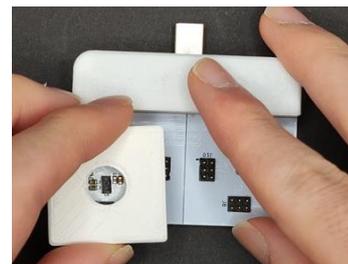


Fig 2 Module と Base の組み立て

本プロダクトを使用した開発の流れを示す。

STEP1

mCn のホームページ(Fig 3)を開き、プロジェクトに必要な Module を選定し、入手する。

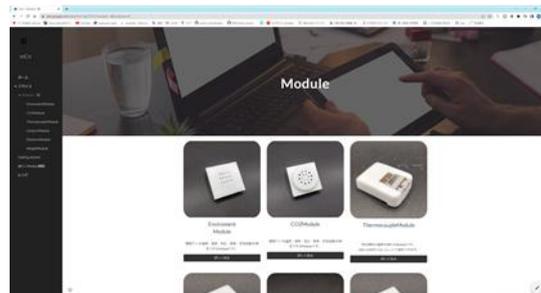


Fig 3 mCn ホームページ Module ラインアップ

STEP2

Base に Module を組み合わせスマートフォンに接続する(Fig 4)。

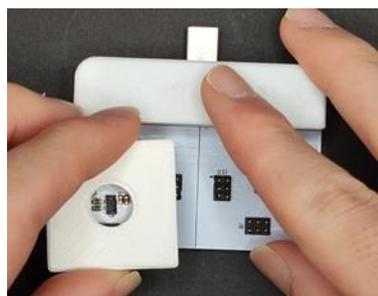


Fig 4 Base と Module の組み立て

STEP3

シリアルから入力される文字列を受け、アプリケーションに反映させる(Fig 5)。

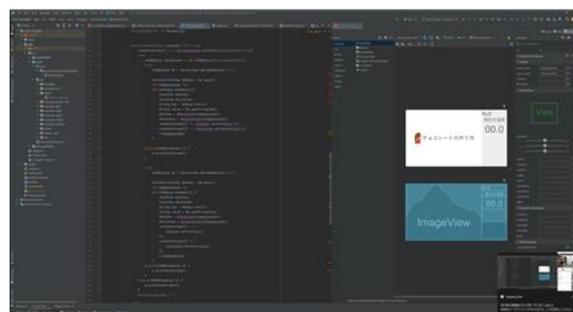


Fig 5 Android Studio を使用したアプリケーション開発

本プロジェクト期間では、Module を CO₂Module、ENVModule、熱電対 Module、ジェスチャー Module、距離 Module、重さ Module、ジョイスティック Module の計 7 種類 (Fig 6)、Base を BasicBase、CaseBase (Fig 7) の計 2 種類を作成した。



Fig 6 Module ラインナップ

BasicBase



typeC ポート

CaseBase



激しい動かす用途でも使用可能

Fig 7 Base ラインナップ

4. 従来の技術(または機能)との相違

本プロジェクトでは、ハードウェアを API のように扱えるようにし、スマートフォンに搭載されていないセンサを簡単に追加できるようにした。実際にユーザ体験では「API みたいでとても簡単だった」という意見をもらった。

簡単に、ハードウェアを用いるシステムを開発できるツールとして M5Stack が有名である。M5Stack との比較を Table 1 に示す。

Table 1 mCn と M5Stack の比較

	M5Stack	mCn
誰のためのものか	HW & SW エンジニア	SW エンジニア
必要な開発 ^(※1)	Unit、Base の組み合わせ調査、組み込みソフト、アプリケーション開発	アプリケーション開発
通信規格	UART, I2C, GPIO, SPI, I2S	UART, USB

※1：スマートフォンと組み合わせる開発の場合

M5Stack は様々な通信規格が提供されているため自由度は高いが、ある程度のハードウェアの知識を必要とする。一方、本プロジェクトではハードウェアの自由度は低いがソフトウェアエンジニアにフォーカスしハードウェアを API 化するための開発環境を構築した。

5. 期待される効果

ソフトウェアエンジニアがハードウェアを手軽に扱えるようになれば、既存のソフトウェアアプリケーションにハードウェアが組み合わさった新たなアプリケーションが生まれてくる。実際に、本システムで作成したアプリケーションを Fig 8、Fig 9、Fig 10 示す。



Fig 8 定量的な料理アプリケーション



Fig 9 重さ記録アプリケーション



Fig 10 薬管理アプリケーション

6. 普及(または活用)の見通し

本システムにより、ソフトウェアエンジニアがハードウェアを手軽に扱えるようになれば、既存のソフトウェアアプリケーションにハードウェアが組み合わさった新たなアプリケーションが生まれてくる。例えば既存の料理アプリケーションにセンサが接続された場合、今までユーザの感覚に頼っていた部分をセンサにより定量化することができれば、誰でもおいしい料理を作ることができる。

7. クリエータ名(所属)

饗庭 陽月(茨城工業高等専門学校)

(参考)関連 URL

ホームページ: <https://sites.google.com/view/mcn-hp/%E3%83%9B%E3%83%BC%E3%83%A0>