

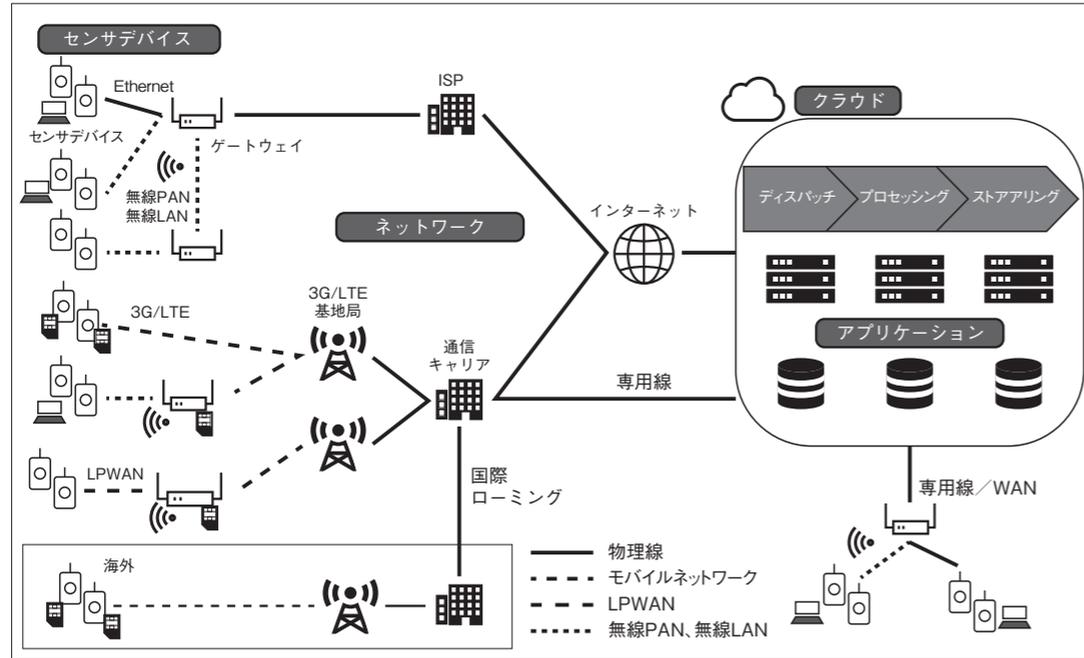
IoTシステムのオーバービュー

図2-1はIoTシステムの全体像で、さまざまな形でセンサやデバイスが接続され、データがクラウド上に集まり、そのデータに対して処理が行われることを示しています。

とを示しています。

IoTシステムでは、大きく分けて「センサデバイス」「ネットワーク」「クラウド」「アプリケーション」のセグメントがあり、またそれを包括する形で「セキュリティ」のセグメントがあります。これら各セグメントについて説明します。

■図2-1: IoTシステムの全体像



センサデバイス

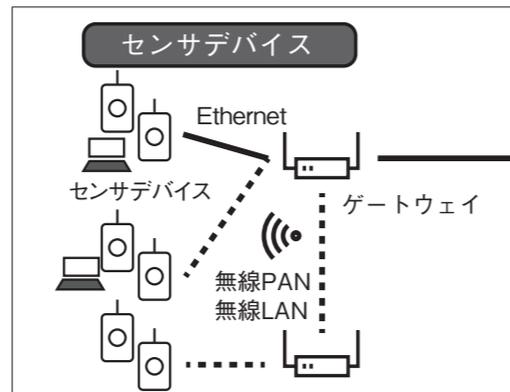
図2-1で、一番左側に位置するのが「センサデバイス」です(図2-2)。物理世界につながり、実際のデータを取得したり、クラウドからのフィードバックを受けてモーターやポンプ、LEDなど、物理世界で稼働するモノ(アクチュエータ)を動作させたりします。

センサで取得できる情報によって多種多様にあり、例えば次のような情報を取得できます。

- 生体情報
- 振動
- 音声/周波数
- ON/OFF 状態

- 位置情報 (GPS)
- 重量/体積/密度
- 圧力/衝撃力
- 加速度/角速度
- 方向/距離
- 画像/動画
- 光量/色
- 磁力
- 電流/電圧
- 温度/湿度
- 成分量
- 流量

■図2-2: センサデバイス



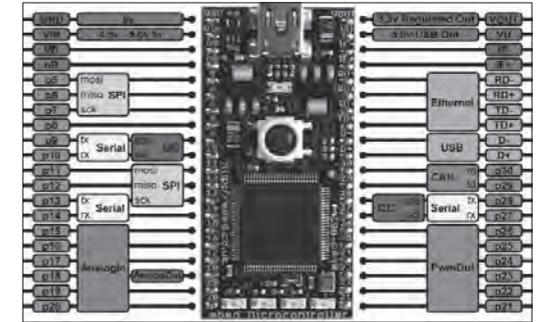
センサはI2C^{注1}やGPIO^{注2}といった物理インタフェースでデバイスに接続され、デバイス上のプログラムから読み取りが行われます(図2-3)。また同様に、物理インタフェース経由でアクチュエータを動作させます。

デバイスはセンサからデータを受け取り、ネットワークを経由してサーバと通信を行います。このため、デバイスには演算能力とネットワーク接続が必要となります。

演算用のプロセッサとして多く使われているのが、ARMアーキテクチャのプロセッサです。2016年にソフトバンクが買収したニュースで一躍有名となりましたが、このARM社が開発したCPUのアーキテクチャが、多くのIoTデバイスで使われています。

図2-4は、32bitのARMプロセッサの乗った製品です。CPUの処理性能としてはクロック周波数96MHz、32KB RAMと、処理能力は高くありません。mbed OSと呼ばれるOSが載っていますが、WindowsやLinuxのようなGUIやコマンドラインインタフェースはなく、C言語で書いた単一のプログラ

■図2-4: mbed LPC1768



ムを実行し続けるようなものになっています。図2-4からもわかるとおり、I2Cなどのデバイスとの入出力ポートや、ルーターなどの接続するためのEthernetポート、シリアル通信ポートなどがついているため、IoTデバイスとして使う場合は、センサの情報を取得し、データを加工してEthernetなどで指定のサーバに送るような動作をさせる形になります。

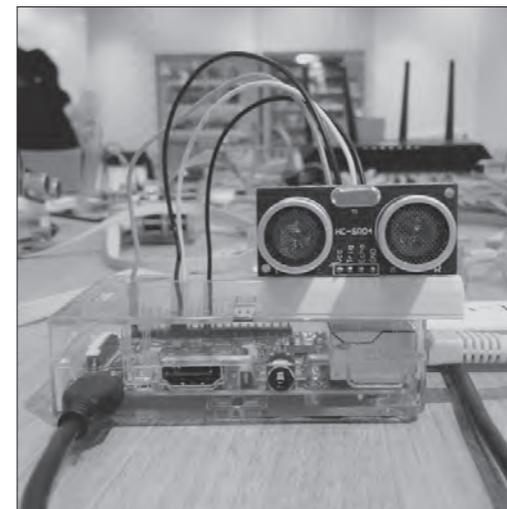
またARMプロセッサ以外では、PCなどでおなじみのインテルアーキテクチャのプロセッサが使われるデバイスもあります。

図2-5は、インテルのAtomプロセッサがCPUとして搭載されています。クロック周波数が500MHz、デュアルコア、メモリ1GB RAMと、一昔前のPC並の処理能力となっています。OSもLinuxが動作するため、CやPython、JavaScriptなどで作られた複数のプログラムを動作させることが可能です。ま

注1) Inter-Integrated Circuit. センサとデバイスを結ぶシリアル通信用のインタフェース。2本の物理線で接続する。

注2) General Purpose Input/Output. デジタル信号と電圧の入出力が行えるインタフェース。GPIOは、入力される電圧をソフトウェアで読み取ったり、指定したりできる。場合によっては0か1を読み取り、ソフトウェアからその値を取得することができる。またデジタル入出力(0 or 1での入出力)を行うことができるため、ソフトウェアと接続されるセンサやアクチュエータとの取り決めを行えば、自由に入出力が行える。

■図2-3: GPIOで接続された超音波センサ



■図2-5: OpenBlocks EX1



配置するのか)、そのうえでどのような通信プロトコルでデータをやりとりするのか、データはデバイス側でバッファリングする必要があるのか、それともクラウド側で行うのか、といった点です。IoTシステムにおける各技術要素との通信方法をあらかじめ整理しておくことで、各セグメントがそれぞれの役割や機能に集中できたり、機能を隣接する技術要素に移譲(オフロード)することで、開発コストを下げるということが可能になります。

注意すべきは他の技術要素に比べて、デバイスは改修に要する期間やコストが多くなる傾向にあるため、IoTシステムの検討段階や検証段階の早い時期に、デバイスに何が必要で何が不要でないかを確認しておく必要があります。

IoTエンジニアとして知っておきたいデバイスのあれこれ

「温度センサ」の怪

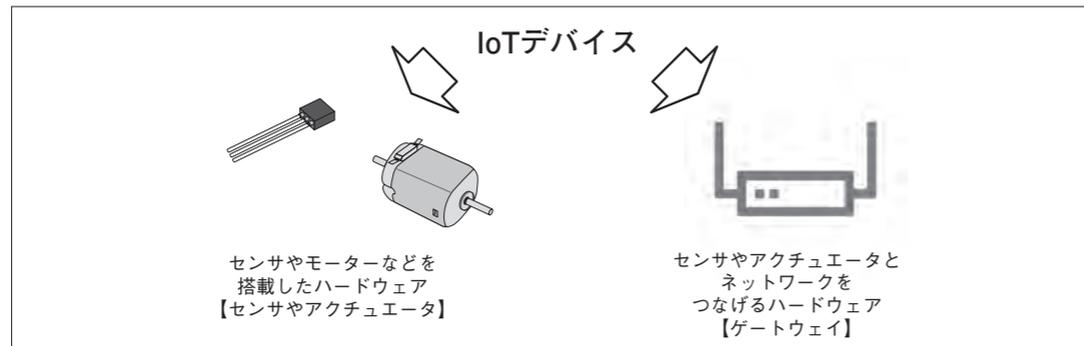
「温度センサ」を検索してみてください。ICチップや細長い棒のようなものが検索結果に出てくるばかりで、例えばBLE (Bluetooth Low Energy) 対応の温度センサを見つけることが難しいことに気がつきます。これはなぜなのでしょう? 実はセンサやデバイスといった単語の定義が、人によって違うことに起因します。

ここでは、IoTのプロジェクトを進めるにあたって、知っているようで知らなかった、デバイスについて解説します。

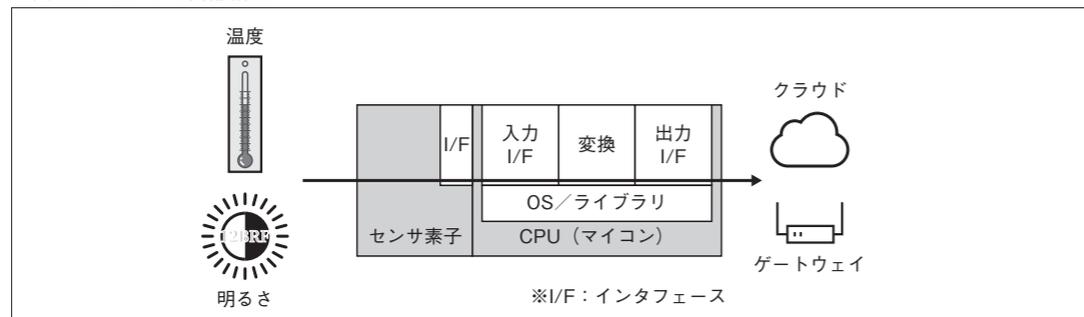
デバイスとは?

IoTにおけるデバイスとは、センサやアクチュエータとゲートウェイといったハードウェアの総称で、それぞれの定義は図3-4のとおりです。

■図3-4: IoTデバイスの大別



■図3-5: センサの内部構成



センサの構成

センサの基本的な内部構成を見てみましょう。コンポーネントとしてはセンサ素子とCPU (マイコン) がインタフェース (I/F) で接続されている構成になります。また、センサ素子からの信号をCPUで処理するために、I/F制御や信号変換を行うOSやライブラリが搭載されています(図3-5)。それぞれの各ブロックが通信し合い、処理を行うことでセンサからの信号をアプリケーションで利用できるようになります。

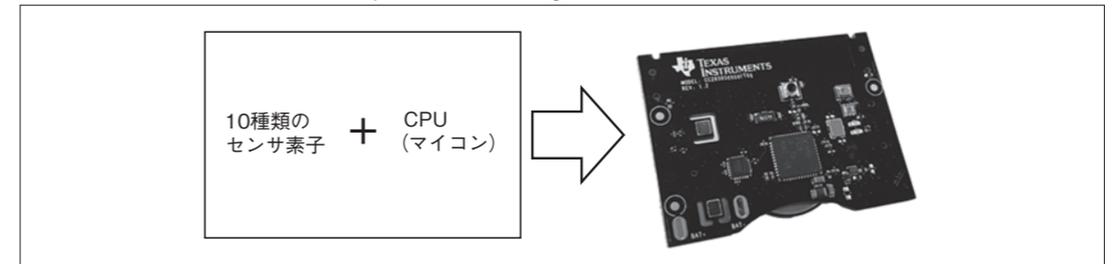
ここで紹介しているような、センサ素子とCPU (マイコン) が一体になっているタイプのセンサの代表

例はTexas Instruments社のSimpleLink Sensor Tag (CC2650STK) です(図3-6)。これは10種類のセンサ素子に加え、BLE (Bluetooth Low Energy) 通信をするためCPUが1つの製品に収められています。

アクチュエータの構成

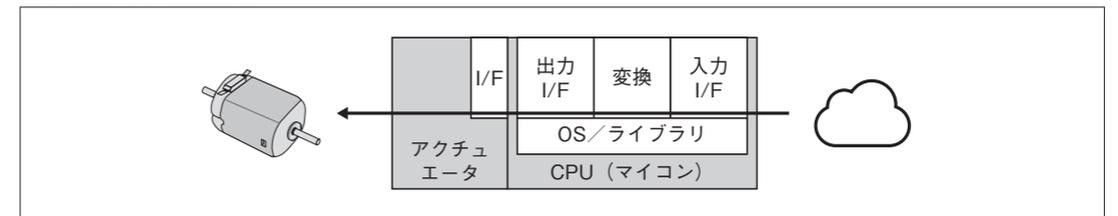
アクチュエータは、信号の入出力の方向が逆転し、センサ素子の部分がモーターなどの可動部品になるといった点以外は、基本的な構成要素はセンサと変わりません(図3-7)。

■図3-6: Texas Instruments社 SimpleLink SensorTag (CC2650STK)

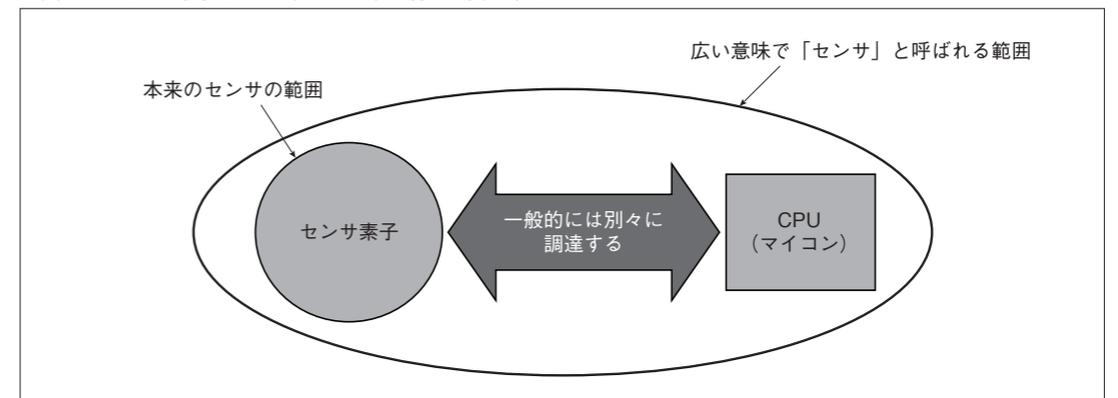


URL <http://www.tij.co.jp/tool/jp/cc2650stk>

■図3-7: アクチュエータの内部構成



■図3-8: センサ素子とCPU (マイコン) は別々に調達する



❖超音波センサ (HC-SR04) (図8-7)

超音波を飛ばして、戻ってくるまでの時間を計測することで、物体までの距離を測ることができるセンサです。

❖温度センサ (DS18B20+) (図8-8)

1-wire方式のデジタル温度センサです。A/D変換機能を持たないRaspberry Piで利用しやすいです。

❖抵抗 (図8-9)

電流が流れすぎないようにしたり、電圧を調整するために使います。本章では、LED用に330Ω、温度センサ用に4.7kΩの抵抗を使います。

ネットワーク

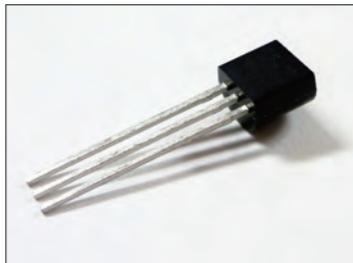
IoTシステムに不可欠なネットワーク接続に関しては、設置場所に関係なく通信が行える3G/LTEネットワークを利用でき、またIoTに便利な機能を有するIoTプラットフォーム「SORACOM」を利用します。

SORACOMを利用するには、次のものが必要になります。

■図8-7：超音波センサ (HC-SR04)



■図8-8：温度センサ



■図8-9：抵抗



❖SORACOMアカウント

「SORACOM オペレーター登録」([URL](https://console.soracom.io/#/signup) <https://console.soracom.io/#/signup>) からアカウントを作成できます(アカウントを作成するには、有効なメールアドレスが必要になります)。日本向けのSIMを利用するため、カバレッジタイプは「Japan」を選択しましょう。

❖SORACOM Air SIM

SORACOM Air SIMカードを入手する必要があります。SORACOM Air SIMカードは、SORACOMのユーザコンソールから直接発注することもできますし、Amazon.co.jpなどの通販サイトで1枚単位で購入することができます。もし1枚だけ購入するのであれば、Amazonで購入することをお勧めします。なお、本書では標準サイズのSIMを利用しますが、他の用途でも使いたい場合は、一番小さいナノサイズと、SIMアダプタを購入してもよいでしょう。

3G/LTE 接続モジュール

Raspberry PiをSORACOM Air SIMで接続するには、通信モジュールが必要になります。通信モジュールにはUSBモデムタイプと専用モジュールがあります。

USB モデムタイプ

動作確認の取れているものでは、図8-10から図8-12のようなものがあります。図8-12はRaspberry Pi側では特に設定が必要がないので、比較的簡単に接続ができます。なお本書の説明では、図8-11の「Abit AK-020」を使用します。

■図8-10：富士ソフト FS01BU (3G)



■図8-11：Abit AK-020 (3G)



■図8-12：PIX-MT100 (LTE)



専用モジュール

メカトラックス 3GPI (図8-13) は、Raspberry Pi用に専用設計された3G通信モジュール基板。電源を強化しているため、安定して稼働します。また、モバイルルータ (図8-14) も利用できます。できれば、クレードルを使うなどして有線接続ができるモデルがよいでしょう。なお、SORACOM Air SIMの使えるAndroidやiOS端末 (図8-15) を使って、テザリングすることも可能です。

■図8-13：メカトラックス3GPI



■図8-14：モバイルルータ



Raspberry Piをセットアップする (Raspbian OSの設定)

Raspberry Piが利用できるように設定していきます。Raspberry PiのOSは、イメージ形式で配布されており、SDカードに書き込むことで利用できます。

イメージファイルのダウンロード

IoT用途であればグラフィカルなデスクトップ環境は必要ありませんので、Liteイメージを利用することをお勧めします。本家サイト ([URL](https://www.raspberrypi.org/downloads/raspbian/) <https://www.raspberrypi.org/downloads/raspbian/>) からダウンロードできますが、国内のミラーサイト ([URL](http://ftp.jaist.ac.jp/pub/raspberrypi/raspbian_lite/images/) http://ftp.jaist.ac.jp/pub/raspberrypi/raspbian_lite/images/) も利用可能です。

本章では、執筆時点での最新バージョンである2019/04/08バージョン (2019-04-08-raspbian-stretch-lite.zip) を使用します。

使用するイメージ (ZIPファイル) をダウンロードできたら、解凍してイメージ (img) ファイルを取り出します。

■図8-15：スマホ (Android/iOS)

