

平成30年度「専修学校による地域産業中核的人材養成事業」  
(Society5.0等対応カリキュラムの開発・実証)

# 富山県をモデルとした「モノづくり」現場に IoTを導入する中核的人材育成

## 〈製造IoT基礎概論 テキスト教材〉

学校法人 浦山学園 富山情報ビジネス専門学校

平成 30 年度「専修学校による地域産業中核的人材養成事業」  
(Society5.0 等対応カリキュラムの開発・実証)

富山県をモデルとした「モノづくり」現場に  
IoT を導入する中核的人材育成

## <製造 IoT 基礎概論 テキスト教材>

本紙は、文部科学省の生涯学習振興事業委託費による委託事業として  
学校法人浦山学園富山情報ビジネス専門学校が実施した平成 30 年度  
「専修学校による地域産業中核的人材養成事業」の成果物です。

学校法人浦山学園 富山情報ビジネス専門学校

# 目 次

## 製造 IoT 基礎概論 テキスト教材

Step1	IoT とは何か	3
Step2	「モノづくり」企業の部門間連携	16
Step3	部門内情報	31
Step4	情報整理	59
Step5	部門間ネットワークの情報	64
Step6	部門間 IoT 導入ポイント	67
Step7	情報通信の方法	71
Step8	目指す IoT 人材の立場	74
Step9	モノづくり現場のネットワーク	76
Step10	モノづくり現場の IoT 導入ポイント	79
Step11	IoT システム開発の方法	82
Step12	無線 IoT 入門	86
Step13	マイコン IoT 入門	90
Step14	PLC による制御と IoT	94
Step15	WEB 連携 IoT の開発	100

## Appendix 【基礎概論】

A	Excel VBA によるプログラミング	118
B	シリアル通信 (RS-232 c)	141
C	生産技術者マネジメントスキル認定試験サンプル	154
D	SW の情報を検出する 2 つの方法	160
E	TWE-Lite Data Sheet (抜粋)	162
F	ESP-8266 Data Sheet (抜粋)	163
G	Arduino IDE のインストール	165

**富山情報ビジネス専門学校**

**なんでもつながる！！**

**IoT基礎講座**

**【製造IoT基礎概論】**

## 【概論編】 目次

- Step1 IoTとは何か
- Step2 「モノづくり」企業の部門間連携
- Step3 部門内情報
- Step4 情報整理
- Step5 部門間ネットワークの情報
- Step6 部門間IoT導入ポイント
- Step7 情報通信の方法
- Step8 目指すIoT人材の立場
- Step9 モノづくり現場のネットワーク
- Step10 モノづくり現場のIoT導入ポイント
- Step11 IoTシステム開発の方法
- Step12 無線IoT入門
- Step13 マイコンIoT入門
- Step14 PLCによる制御とIoT
- Step15 WEB連携IoTの開発

## Step1 IoTとは何か

◇IoTとは、  
機器制御システム から 情報処理システム に  
またがる、横断的コンセプト  
(2016年：東京大学：坂村教授[TRON、ubiquitous])

機器制御 → センサから情報を取り込む  
情報処理 → 情報をDB化し統計・分析する  
★これらがインターネットにつながる

・インターネット  
米国防省の研究 (1970年代始まる)  
↓  
プロトコル (TCP/IP) (通信規約のこと)  
↓  
民間利用許可→オープン化  
↓  
利用の広がり = 普及 (現在に至る)

2

## Step1 IoTとは何か

【IoT】という言葉聞いたことがある人も多いでしょう。その言葉が何を意味するのかを考えます。

◇IoTとは、**機器制御システムから情報処理システムにまたがる、横断的コンセプト**としての位置づけもあります。情報をDB化して、統計・分析を行い、何らかの傾向・特性などを見つけて、制御の舵を切るものです。これらが一体となったモノが、インターネットに繋がる、そのことを指してIoTと言っている場合もあります。

### 1. インフラとプロセッサの性能向上

◇インターネットは、もともと米国防省の研究で開発されたプロトコル(TCP/IP)がオープン化されて民間利用が可能になったことで、PCの性能向上と共に利用が急速に広がったものです。世界中の人たちがこの恩恵を受けています。

◇インターネットはインフラであり、普及には30年以上もかかっています。これに対して、電子デバイスの性能向上と技術の進歩は、どんどん加速してきました。

◇IoTデバイスとして小さな空間を占めるモノが、コンピュータを内蔵し「インテリジェント化」されて、**オープンなネットワーク(=インターネット)**につながる社会が到来しています。そして、その「インテリジェント化」の要(かなめ)であるプロセッサ(マイクロコンピュータ=マイコン)の能力がどんどん向上しているので、末端に位置する「モノ」の内部で処理できる情報処理能力も向上して、IoTデバイス側で或いは、IoTデバイスを群

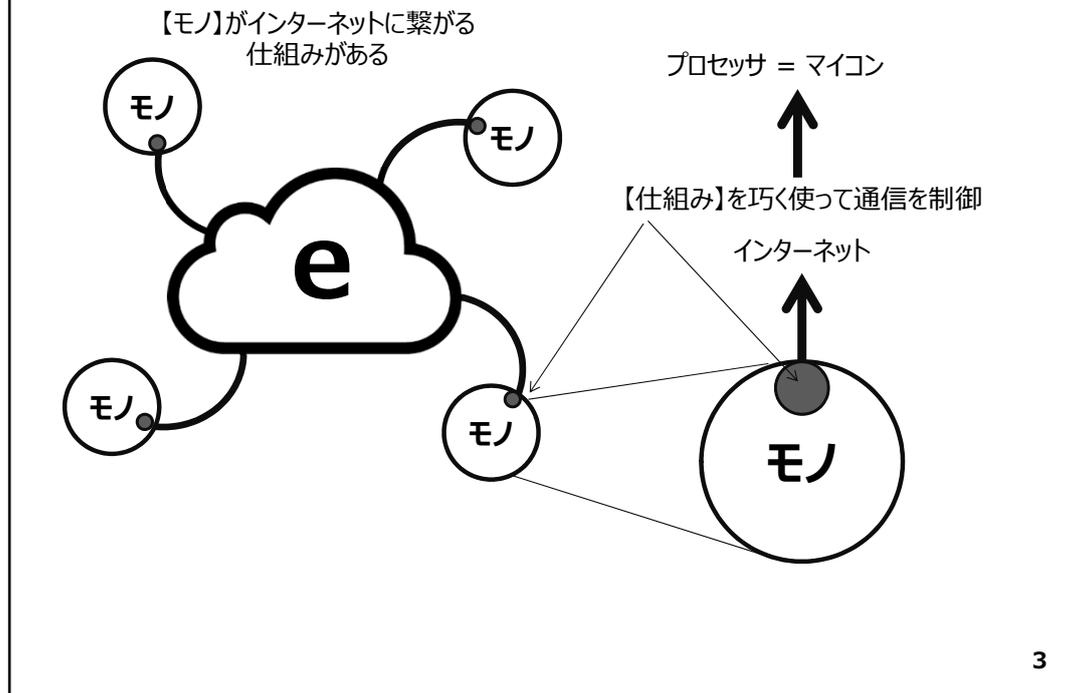
で管理する「ノード」で情報を分類・整理・吟味してインターネット上のホストコンピュータに送る「エッジコンピューティング(※1)」も利用しやすくなってきました。

(※1)エッジコンピューティング：エッジ処理とも呼ばれ、「端末の近くにサーバを分散配置する」ネットワーク技法のひとつを意味します。ユーザや端末の近くでデータを処理することで、上位システムへの負荷や通信遅延を解消します。製造現場では、工場内に多数設置されたセンサや測定器から得られる大容量のデータに対し、高速またはリアルタイムなアプリケーション処理(データの見える化)を可能とします。

一方、インターネットを介してユーザから離れた場所にあるコンピュータを利用する「クラウドコンピューティング」は、遠隔であることから「クラウド(cloud)」つまり「雲」に例えられています。産業における、制御の実行や大きなデータのやり取りには、通信やクラウドでの処理時間がかかるといった課題が生じます。

クラウドを利用する場合のエッジコンピューティングでは、インターネットに送り出す工場内の「ふち(エッジ)」で、データを最適化処理します。これによってインターネットには必要なデータだけを送信するため、通信時の諸問題を解決することができるのです。

## IoTデバイスモデル



## 2. IoTデバイスモデル

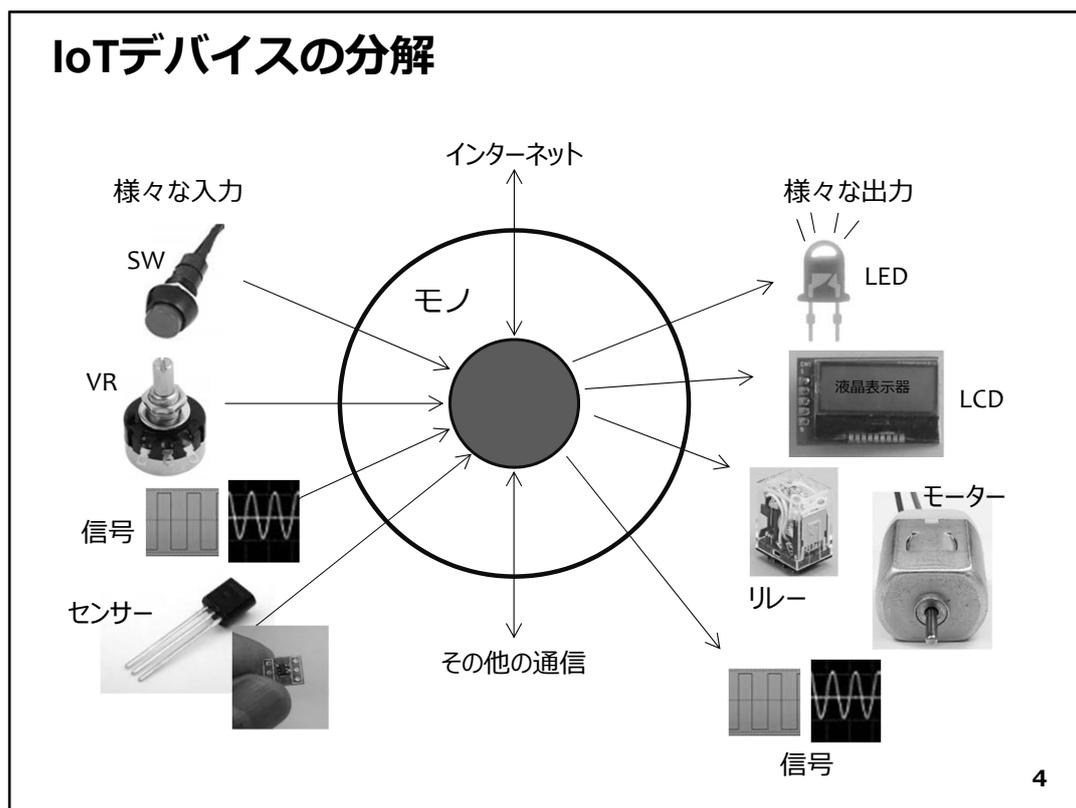
◇IoTデバイスに必要な機能を考えるために、IoTデバイスモデルを観察してみましょう。

図は、IoTデバイスがインターネットに繋がっている様子を示しています。モノがインターネットに繋がるには、その仕組みが必要です。図では青い丸印が該当する部分です。モノは、その【仕組み】を巧く使って通信機能を制御して、インターネットと情報の交換を行っています。

通信を制御するためには、プロセッサ(=マイクロコンピュータ:マイコン)が必要です。小規模でも高性能のマイコンを利用することで、高度な通信制御が可能になり、同時にモノの外部デバイスとして、様々な別のモノを接続することができるようになりました。つまり【モノ】 $\leftrightarrow$ 【モノ】 $\leftrightarrow$ 【インターネット】という構図です。

【モノ】には他の【モノ】も接続できるので、単純機能のモノを数多く組み合わせれば、複雑だったり、手間が掛かったりすることも実現可能になります。それが容易にインターネットにつながるのです。

## IoTデバイスの分解



### 3. IoTデバイスの分解

#### ◇IoTデバイスを分解して調べる

モノはインターネットに繋がっていますが、同時に他の通信を行い外部の設備・装置やPCなどと連携することもあります。中枢部のマイクロコンピュータには、入出力ポートがあり、入力・出力とも様々な外部デバイスが繋がります。

#### ◇入出力の代表的なものは、

入力:

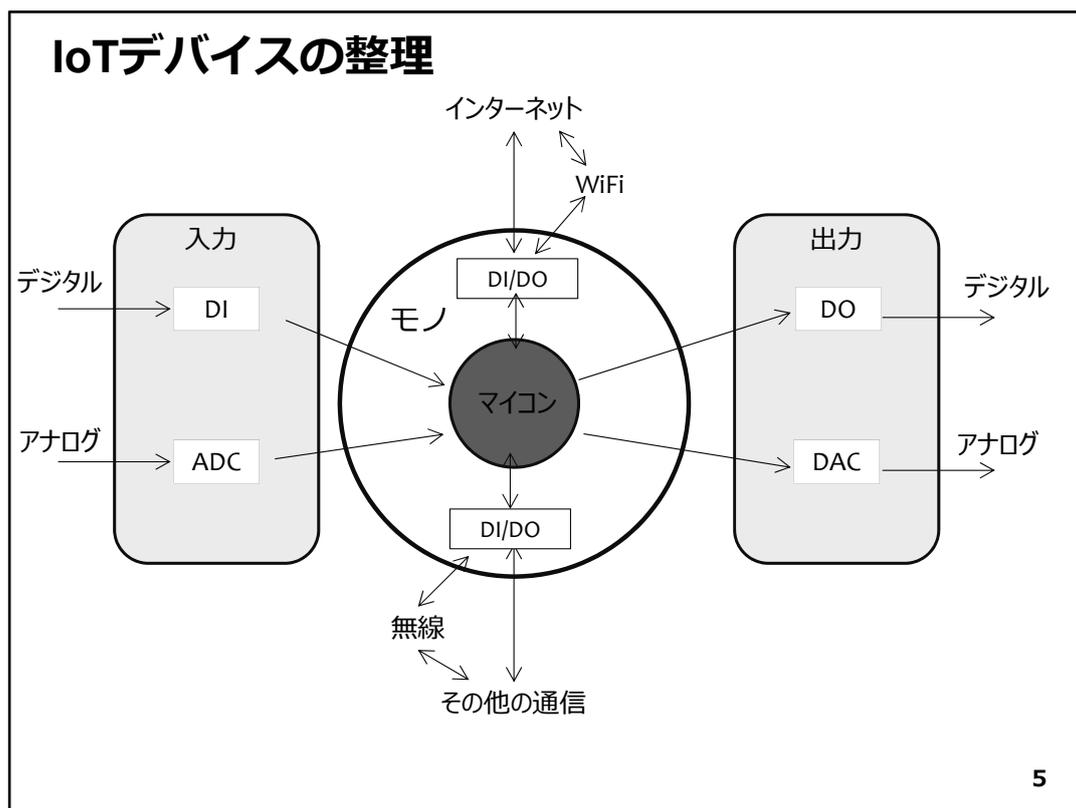
- ・SW(デジタル入力)
- ・VR(可変抵抗器:アナログ入力)
- ・信号(デジタル信号もあればアナログ信号もある)
- ・センサー(出力が電圧のものが多いが、電流出力のものもあり、対象のデータにするには、換算計算が必要)
- ・デジタルセンサー(対象データが直接読める)

出力:

- ・LED(デジタル出力)
- ・LCD(液晶表示器:消費電力が小さいので、IoTデバイスに好都合)
- ・リレー(トランジスタなどを介して接続し、大きな電力を制御したり、シーケンス制御などに利用する)
- ・モーター(リレーと同様、トランジスタなどを介して接続する。モータードライバICなどを利用すれば、回転方向の制御などが容易に行える)

- ・信号(デジタル信号は、周期の制御がソフトウェアでコントロールできる。  
電圧レベルの制御は、アナログ信号で行う)

入力・出力・その他の通信の先には、設備・装置・他のIoTデバイスが繋がります。上記の他にも、外部メモリやSDカードなど、マイコン内部には記憶しきれない容量のデータを、外部に保存し必要なときに読み出すデバイスもあります。そこに記憶できるデータは、デジタルデータなので、デジタル入出力の機能で使うことができます。



#### 4. IoTデバイスの整理

##### ◇分解したIoTデバイスの要素

ここまでの分析から分かる、IoTデバイスの構成要素を整理すると、図の様に

##### 1. 入力

- ①. DI(デジタル入力)
- ②. ADC(アナログ値の入力はデジタル値に変換してコンピュータで処理するため、アナログーデジタル変換器を介して入力する)

##### 2. 出力

- ①. DO(デジタル出力)
- ②. DAC(アナログ出力は、コンピュータが取り扱うデジタル値から直接出力できないので、デジタルーアナログ変換器を介して出力する)

##### 3. インターネット通信

- ①. ワイヤ(ケーブル)を使ってインターネット通信を行う
- ②. WiFi等の無線を使ってインターネット通信を行う

##### 4. その他の通信

- ①. ワイヤ(ケーブル)を使って通信を行う  
(今までのシリアル通信→例:Rs-232c)
- ②. 無線を使いワイヤレスで通信を行う

(2.4GHz帯を用いたもの、LoRa(※1)など)

(※1)LoRaとは、LPWA(Low Power, Wide Area)という、少ない消費電力で広いエリアをカバーする無線通信方式の一つで、IoT向けの通信ネットワークに用いられています。LPWAは、免許不要の周波数帯域を利用するものと、免許が必要な周波数帯域を利用するものに大別できます。LoRaは前者で、1GHzよりも低い「サブギガ帯」と呼ばれる周波数帯域を利用します。日本では、免許不要の特定小電力無線局が利用できる周波数帯(920～928MHz)がこれに相当します。最大伝送速度は250kbps程度で、伝送距離は最大10km程度。LoRaはオープンな通信規格であり、広く普及させるためにLoRa Allianceという非営利団体が組織されています。LoRaによるネットワークを「LoRaWAN」と呼びます。

## IoTデバイスの要素

前の図で分かるように、IoTデバイスは、

- ①. デジタル入出力
- ②. アナログ入出力

の使い方を理解すれば、概ね利用できる。

また、大量のデータを処理したい場合は、外部デバイスとしてメモリカードなどを利用するが、これもデジタル入出力の範疇である。

特殊な制御方法の一つとして、デジタル出力でアナログ出力を模擬する、PWM制御（パルス幅制御）という手法がある。

6

### 5. IoTデバイスの要素

◇大量のデータを利用する場合は、IoTデバイスではデータ蓄積だけを行って、処理はホストコンピュータに任せるのが良い方法です。ホストコンピュータは、情報をテキストファイル・表・DB化して、統計・分析を行います。

このときのホストコンピュータを置く位置によって、

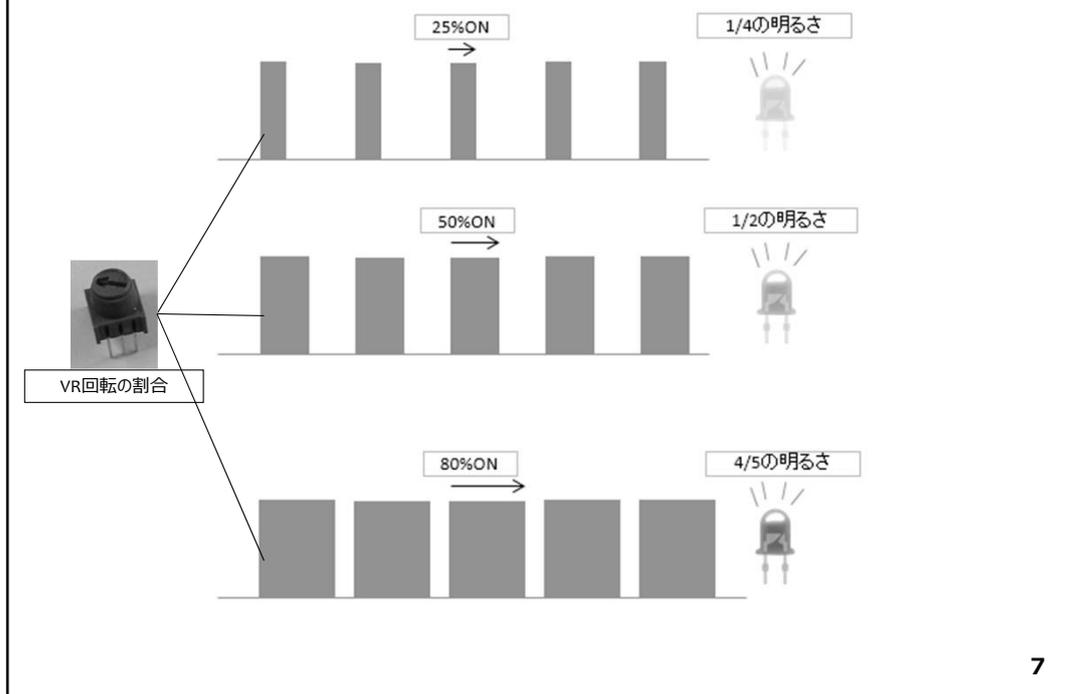
- ①. クラウドコンピューティング  
→ 全てのデータをクラウドに送り、クラウド上のサーバーで全てのデータを処理する。
- ②. エッジコンピューティング  
→ IoTデバイスに近い所にサーバーを置き、そこで、中間処理して必要なデータのみ、クラウドに送る。

に分かれます。

◇特殊な制御方法として、デジタル出力でアナログ出力を模擬する、**PWM制御**（パルス幅変調制御）という手法があります。既にLED電球に置き換わってしまいましたが、蛍光灯が高速で点滅を繰り返しているのに、人の目には連続点灯している様に見えるのもその例です。普通のLEDも点・滅の時間幅を変化させると明るさが連続的に変化してPWM制御を調光機能として使えます。

◇PWM制御についてもう少し詳しく説明します。

## PWMとは パルス幅制御



## 6. PWM制御

◇PWM制御は(Pulse Width Modulation)パルス幅変調制御と言われています。

LEDを点滅する信号があり、OFFではLEDの明るさは0%で、ONした時100%の明るさだとします。このON/OFF信号を一定周期で繰り返すパルスとして発生させます。ONの割合が例えば周期の25%の時はLEDが1/4の明るさで光ります。次にこのONの幅を50%に広げると、LEDは半分の明るさで光る・・・というように、ONの幅(パルス幅)をコントロールすることで、ある対象の動作を制御する仕組みがPWM制御(パルス幅変調制御)と言われるものです。

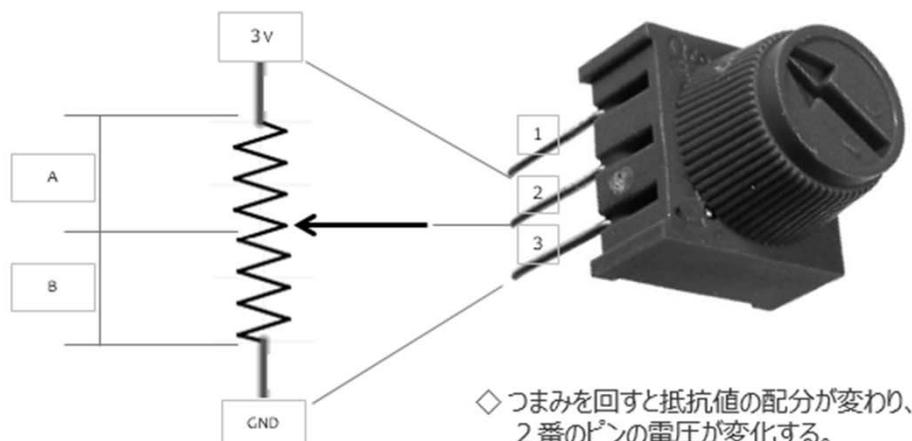
図で示すようにVR(Variable Resistor: 可変抵抗器)を使い、回転の割合に応じて、このパルス信号のHighとLowの幅の割合を変化させると、LEDの明るさが無段階に調整できます。LEDの代わりにモータ駆動回路を接続すると、無段階の回転数制御ができます。明るさ制御や、モータ回転数制御の他、同様にして速度制御や流量、ヒータ(加熱)の制御などに利用されています。

では、VRの回転の割合が、どのようにしてパルスの幅に変換できるのでしょうか。

次にその仕組みを説明します。

## VR（可変抵抗器）の役割

電圧を分ける → 分圧



8

### ◇VR(可変抵抗器)の役割

VRには3本のピンが出ています。このピンに図のように1～3の番号を付けます。1番ピンに3V、3番ピンにGND(0V)を接続すると、2番ピンからはVRの回転の割合に応じて内部の抵抗がAとBの部分に分割されて、それに対応する電圧が取り出せます。

VRのつまみを回すと、図の矢印の部分が上下に移動します。一番上に移動すると2番ピンは電圧が3Vになり、反対に一番下に移動すると2番ピンは0V(GND電圧)になります。このように内部の抵抗値の配分で両端にかかっている電圧を比例配分するので、このことを抵抗値による分圧(※1)と言います。

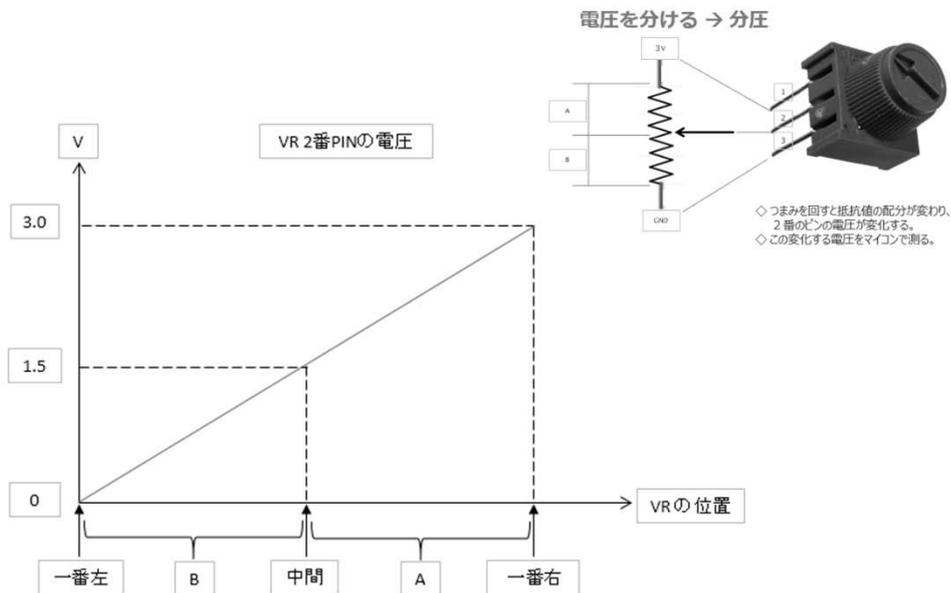
この2番ピンをマイコンのアナログ入力信号に接続して電圧を測り、その値をPWM制御の元になるパルス幅へと変換します。電圧に応じたデジタル値に対応する時間割合でパルスのON時間を長くするのです。(=ON時間が長くなるようなプログラムを組む。)そうすれば、意図したようにLEDの明るさなどを無段階で調整することができます。

マイコンの場合には、一般にADC(A/D変換器)を内蔵しているので、そこで電圧計測が行えます。内蔵していない場合は、外部にADCを取り付けて計測し、その出力をマイコンで読み取ります(=そのような回路を作り、プログラムを組む)。

読み取った電圧(測定電圧)からパルス幅への変換は、マイコンのプログラムで行います。

(※1)分圧:VR両端の電圧は、中央の端子で分けられたA,B部分の両端の電圧の合計に等しくなります。ここでは、VRに3Vの電圧を加えています。VRは、A、Bの部分に分割されていると考えます。A、Bそれぞれの抵抗にかかる電圧を $V_a, V_b$ とすると電圧の足し算ができて、 $3V = V_a + V_b$  になります。

## 分圧



9

### ◇分圧

前掲の図を右上に示します。左下の図は、VR2番ピンの電圧を示すグラフです。VR両端には3Vの電圧が掛かっていて、VRのつまみを回すとVR2番ピンの位置が変化します。一番左に回し切ると、VR2番ピンはGND(=0[V])の位置になるので、その時のVR2番ピンの電圧は0[V]になり、一番右に回し切ると、VR2番ピンは、電源(3[V])の位置になるので、その時のVR2番ピンの電圧は3[V]となります。

つまみの位置が中央であれば、A:B=1:1となって、VR2番ピンの電圧は、

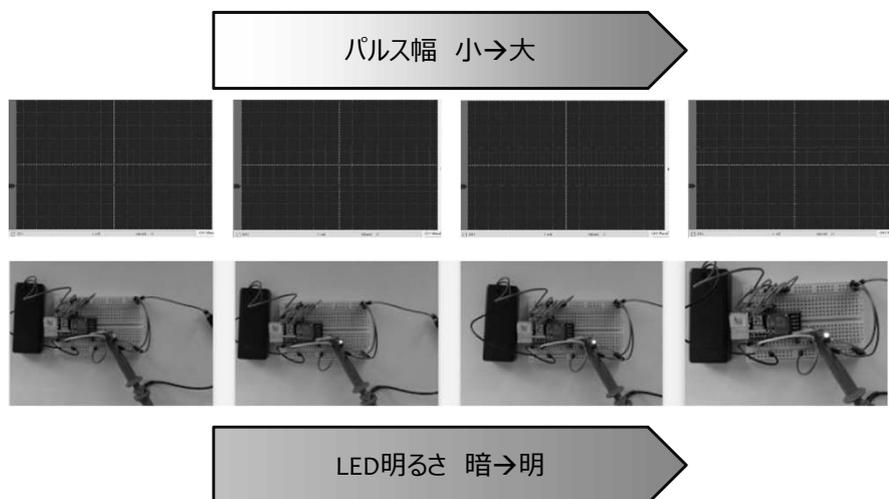
$$3[V] \times 1/2 = 1.5[V]$$

になります。

**問題:** VRのつまみの回転範囲が、0~180度だとする。この時、上の図の様にVRの両端には3[V]の電圧が掛かっている。VRのつまみを120度の位置まで回転すると、VR2番ピンの電圧は、いくらか。但し、0度がGNDの位置、180度が電源の位置とする。

## パルス幅とLEDの明るさ

◇繰り返し出力するパルスの幅に比例して、LEDの明るさが変化します。



10

### ◇PWM制御の例

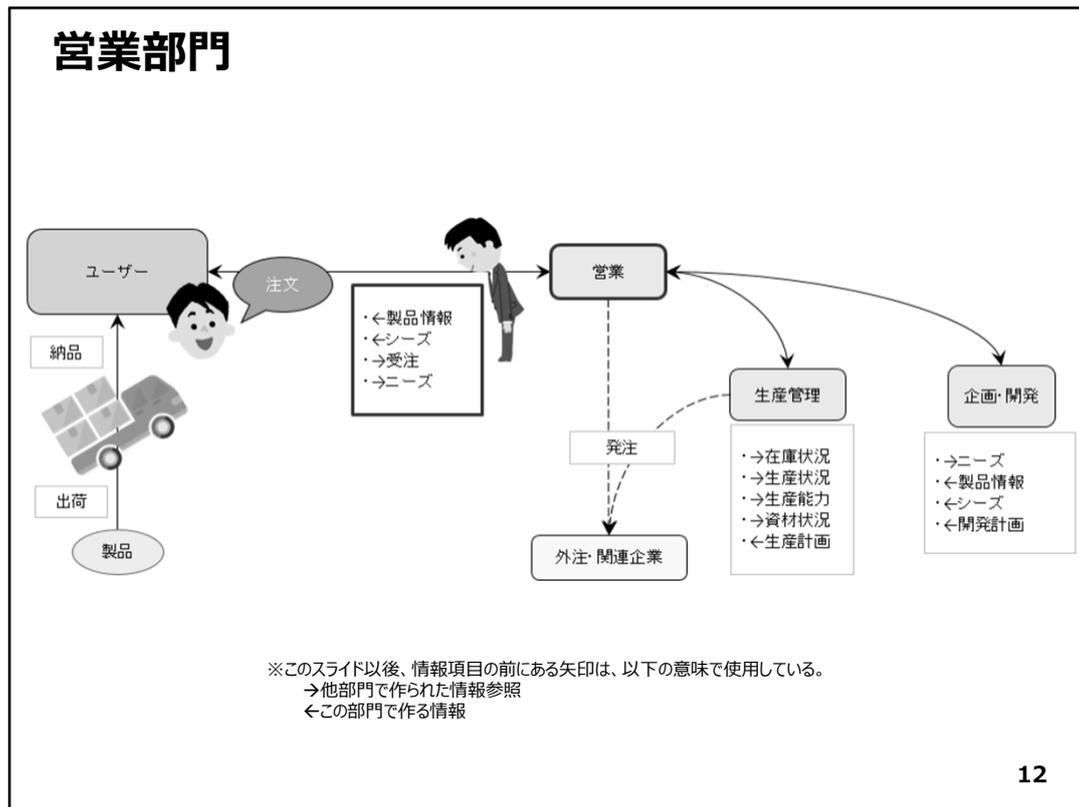
図は、LEDの明るさをPWM制御によって制御している例です。写真下には、実験回路が写っており、オシロスコープ(※1)で、LEDに与えている信号を観察しています。写真上は、その時の波形を示します。VRの回転に対応して、写真左から右に進むに従い、パルスの幅が広がり、対応する写真下のLEDが明るくなっている様子が見えています。

(※1)オシロスコープ: 時間と共に変化する電気信号を捉えて、表示する装置。  
信号振幅や周期などが分かる。通常横軸が時間軸、縦軸が電圧(図上の写真)となっている。



★将来は、他の部門への提案などが出来るようになることを目標としています。

◇これから、図に掲げたそれぞれの部門の仕事と、各部門で接する情報について分析します。



## 営業部門

◇営業部門は、モノづくり企業とお客様を結びつける大切な役割があります。当然ですが、企業の製品をお客様に買っていただき売り上げにつなげる、企業の重要な部門です。

◇お客様に採用していただく製品によって、お客様の企業活動も支えるという重要な使命があります。

◇営業部門はお客様と接して、伝えるべき情報として、

- ①. 製品情報(在庫状況や納期、どのような場面で使用できるモノか、どれ程の価格か、耐久性はどうか、大きさ・重さは・・・、他社との比較)
- ②. 新製品情報(新しい製品の企画・開発状況、市場への投入時期、現行製品との違い、・・・他社との比較)
- ③. まだ製品としての形はないが、自社の持つ技術やシーズ(※1)、それらを活用した将来展望

などが考えられます。

また、お客様から頂く情報として

- ①. 注文
- ②. 今後のニーズ
- ③. お客様の同業他社情報

#### ④. お客様の企業活動状況

などがあります。当然、企業秘密情報があるので、お客様企業との窓口となる営業部門は、企業情報管理を行う部門と協議して、公開してよい情報やお聞きしてよい情報を確認しながら、コンプライアンス(※2)を守った営業活動が求められます。

◇お客様から頂いた注文(受注)情報は、生産管理部門に伝えられ、生産計画が作られます。

◇ニーズ情報は、企画・開発部門に伝えられ、今後の製品開発に役立てられます。地方都市の名産品などを扱うアンテナショップというのがありますが、営業部門は企業のアンテナショップ担当ともいえます。

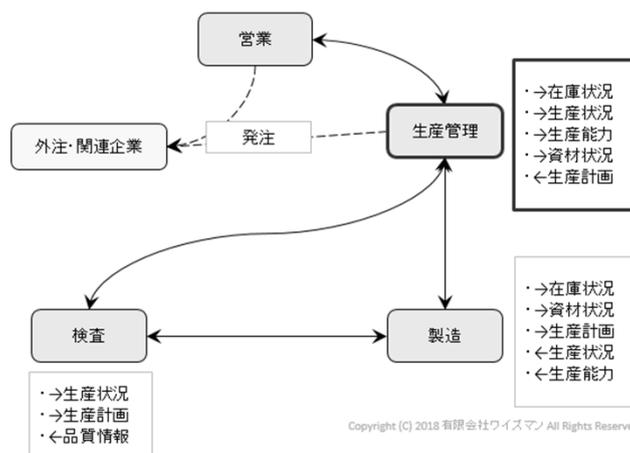
◇企業によっては、営業部門が社外生産の判断権限を持ち、在庫や生産状況などを考慮して、営業部門が外注に直接発注する場合があります。

◇一般的には、生産管理部門で社内生産か社外生産(外注)かを判断する場合があります。

(※1)シーズ:シーズとは、研究開発や新規事業創出を推進していく上で必要となる発明(技術)や能力、人材、設備などのことを言います。「ニーズ」という用語と対比で用いることもあり、その場合はニーズが顧客の要望によって必要とされるものである一方で、シーズはメーカー・企業などが必要に応じて提供する技術や商品などを意味します。

(※2)コンプライアンス:法律や社会的な通念を守ること。法令順守と訳されることが多い。1990年代後半から企業の法律違反に端を発する事件が相次いで発生したことから、企業はより厳密に法律を守るべきという社会的要請が強くなっています。

## 生産管理部門



13

◇生産管理部門は、次の情報を基に、製造部門の負荷が平均化するように生産計画を作成します。

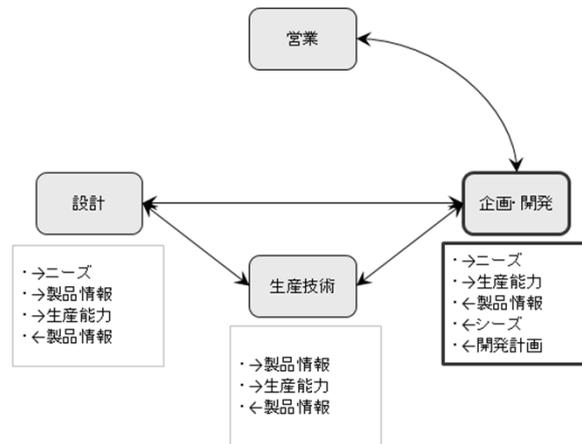
- ①. 在庫状況：(完成品としての在庫と組立・製造途中の中間在庫があり、製造工程数が多い製品は、中間在庫が多くなります。また、在庫が存在する場所が定まっていなくて、正確な把握が難しくなります。あるはずなのに見つからない！ということが起こります。)
- ②. 生産状況：生産実績とも言う。実際にどれくらいのペースで製品がどれだけ出来上がっているか、製品の種類やロットごとに管理されるべき情報です。
- ③. 生産能力：製造現場の能力を指す場合が多いが、企業としての出口の能力、つまり、検査にかかる時間なども含めて製品を外部に供給できる能力を指すこともあります。
- ④. 資材供給状況：モノづくりには、材料や加工する際に必要な設備の消耗品(油や工具、配線材料、ネジ・・・etc)の供給状況で不足が生じると、現場は余力があるのに、製造できない事も起こります。また、検査などで必要な資材にも配慮が必要です。

◇生産管理部門は、受注の情報から単に計画を作成しているだけでなく、リアルタイムにモノづくりの状況を把握する必要があり、実際の製品は作りませんが、モノづくり企業を外部から見たとき、安定して常に同じ品質の製品が一定のタイミングで供給できる能力を左右する重要な部門です。このことが、企業評価のポイントにもなります。

◇生産管理部門は、外注先も含めた在庫や生産能力、生産状況を管理して、社内製造か外注へ委託かの判断をする場合もあります。外注関連企業も含めた総合力で企

業評価が行われることもあるので、生産管理部門の気遣いは、外注・関連企業などの社外にも及ぶことがあります。

## 企画・開発部門



14

◇企画・開発部門は、営業部門から伝えられたお客様のニーズを満たす、新たな製品の仕様を決めて、実際にそれが製造できるのか、仕様を満たし利益につながるのかなどを検討します。

◇机上のアイデアから始まり、製造部門でその製品が作れるのか、必要な資材は調達できるのか、既存製品の部分流用などが可能かなど、検討事項は多岐にわたります。

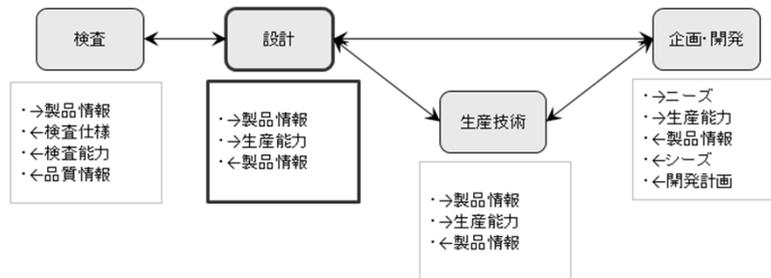
◇様々な検討のために試作品などを作り検討します。試作品といえども設計が必要なので、設計部門と協調して開発を進めます。企業によっては開発設計部門や開発試作部門などが独立している場合もあります。試作品作りは、実際に製造部門の設備・人材でモノが実際に作れるのかも検討の対象となり、製造コストも検討します。モノの作り方の専門家の生産技術部門とも密に連携して仕事を進めます。

- ①. ニーズ:お客様のニーズ。
- ②. 生産能力:生産管理部門で説明したものに加えて、新製品の製造に現行の設備・人材が対応できるかという意味の能力＝性能ともいえる。製品の加工精度なども含みます。
- ③. 新製品情報:新たな製品の仕様は社外秘情報を含むので、社内部門向けの新製品情報と、お客様に伝える新製品情報は別と考える方が良いでしょう。
- ④. シーズ:既に開発した製品の仕様、製造技術、設計技術、製品に使用する部品や部材、或いは、検査技術やモノの管理技術など、あらゆるノウハウがシーズと成り得ます。

⑤. 開発計画: 成長を目指し社内技術力を向上させ、製品開発の将来展望をお客様や社会に提供するために、開発のロードマップを作製しておくことは重要です。

◇図には示していませんが、製品検査で新しい製品を効率よく検査できるのかも検討課題なので、検査部門との協調も必要になります。

## 設計部門



15

◇設計部門は、製品を製造するのに必要な設計図と製造(または組立)仕様書を作ります。企業によっては、製造仕様書を生産技術部門が主導して作る場合があります。

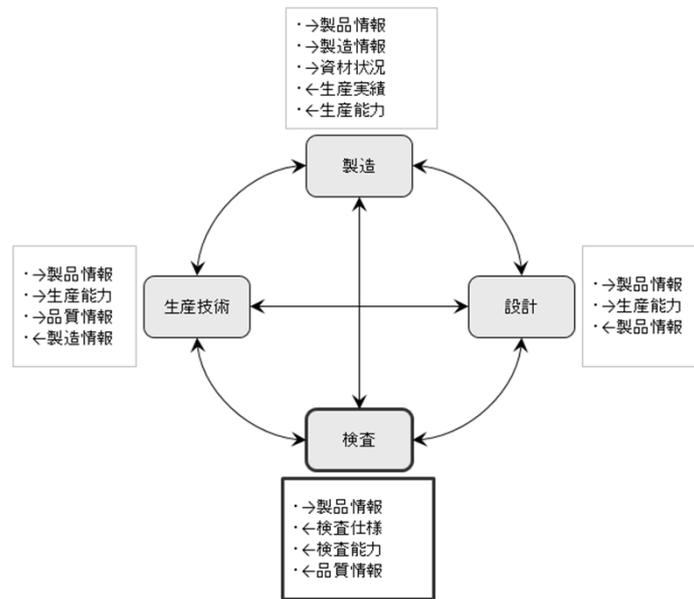
◇企画・開発部門での試作品の為の設計→試作設計 と 量産するための設計→量産設計 とに区別されている場合があります。

- ①. 新製品情報: 企画・開発部門で作られた新しい製品の仕様です。
- ②. 生産能力: 生産管理部門で説明したものに加えて、新製品の製造に現状の設備・人材が対応できるかという意味の能力＝性能とも言えます。  
★製品の加工精度なども含みます。
- ③. 製品情報: 設計図・製造仕様書など、製品を製造するために必要な情報。

◇設計された製品は、最終工程で製品検査を受けるので、検査部門と協調しながら検査仕様を作る場合があります。

◇社会・公共に広く利用されるようなモノ(道具・建築物・薬品・食品・機械・設備・・・)は、国や国際間で決められている安全規格などがあり、企画・開発段階から十分な検討が行われます。設計段階を経た製品は、完全に明確な詳細仕様が決まっていなければいけません。

## 検査部門



16

◇検査部門では、工程途中では、部品・部材など(外部から納品される場合もある)を製造工程中で検査して、仕様通りに製造されているかを確認します。最終的に、製品として組立・製造されたものを製品検査して、【お客様に出荷・納品できるもの】ということを確認すると同時に、品質管理データを蓄積します。

◇社外から調達する資材についての検査(仕入検査)を検査部門が行う企業もあるし、仕入・購買部門が行う企業もあります。

◇工程途中の検査で不合格品が出た場合は、製造部門・生産技術部門と協調してその原因を分析します。不合格の原因は、資材不良・作業ミス・設備不良・・・などいろいろ考えられますが、時には設計ミスということも有り得ます。

◇素早い検査と、適正な検査精度が求められます。

◇新たな製品が開発された場合には、検査手法などの検討も必要です。

◇出荷する全ての製品について、製品検査を行うべきですが、例えば小さなネジなどの大量生産品で全数検査が難しい場合は、一定割合の抜き取り検査を行うこともあります。

◇検査部門では、以下の情報についての配慮が必要です。

- ①. 新製品情報: 新しい製品の検査基準となる情報
- ②. 検査仕様: 検査指示書という場合もある。検査の項目と内容、検査方法。記録

としての検査か、判断材料としての検査かの区別や閾値など。

- ③. 検査能力: 検査部門の精度管理の為の指針となる尺度。例えば、あらかじめ検査結果が分かっているモノを検査対象に混在させて検査し、精度や検査の品質を評価する。(精度管理(※1))

(★食品・医薬品などの分析・検査では、GLP: Good Laboratory Practice という精度管理手法が用いられています。)

自部門の能力をあらかじめ知っておくことは、部門に関わらず重要な点であることは言うまでもないことです。

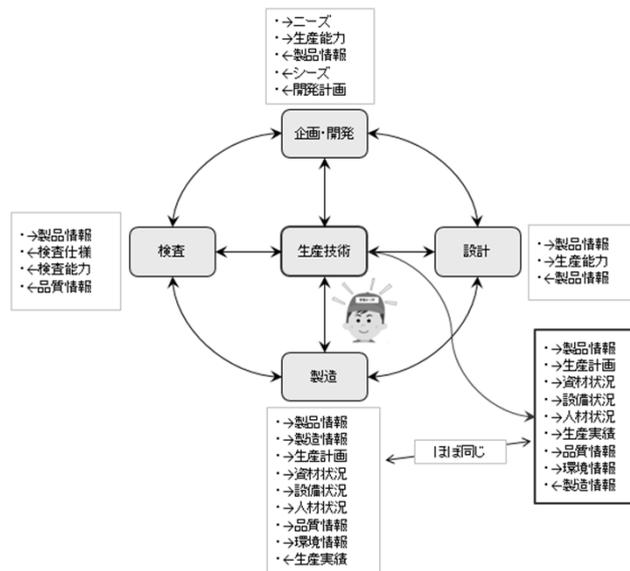
- ④. 品質情報: 検査結果の実績をまとめたDataが品質情報となり、適正に分析された結果は企業外にも広報され、評価されます。検査結果は原材料・資材・部品・製品一つ一つに独立した情報であり、蓄積されたものはBig Dataとなります。検査結果をいちいち人が記録するのではなく、検査を行えばその結果がDataとして自動的に記録・保存されて、後の分析に役立つようなシステムにもIoTが利用されています。統計・分析システムにシームレスにつながれば、検査結果＝製品品質の見える化システムとなります。

(※1)精度管理: 精度管理の目的の1つは、「安定して正確な測定結果を得る」ことにあります。「安定して」とは、「再現性が良い」、あるいは「バラツキが小さい」という言葉に置き換えることもできます。したがって、安定しているかどうかは、再現性やデータのバラツキ程度を評価するのと同じ方法で評価することができます。同一の検体を複数回測定したときに得られる測定値が、ほぼ同じであり安定した測定値が得られている状態を、精度管理では一般に「精密性」が高いと言います。数値としては、多くの場合SD(標準偏差)やCV(変動係数)といった統計量が用いられ、これらの数値が小さいほど精密性が高いといえることができます。

$$\text{測定値} = \text{真値} + \text{偶然誤差} + \text{系統誤差}$$

一方、正確性は実際の測定によって得られた測定値(観測値)が、どの程度「真値」に近接しているかを評価することになります。真値とは、検体中(検査対象)に含まれる測定対象物の真実の値を示す概念ですが、多くの場合真値を知ることは非常に困難です。そこで、多くの場合様々な条件(試薬構成や使用する測定機器、施設など)で測定した場合の測定値を真値に替わる値としています。外部精度管理プログラムであるコントロールサーベイで、全参加施設から報告された測定値の平均値を基準として正確性の評価が行われるのはこうした理由によります。測定値は真値に測定誤差を合わせたものと考えられることができますので、正確性は誤差の大きさを評価していると考えられることもできます。

## 生産技術部門



17

◇生産技術部門は、全ての部門と連携してモノが効率よく、安定して製造されるよう、モノづくりの情報(ここでは、製造情報と表現した。モノづくりのノウハウも含まれています。)を製造部門に提供します。(モノづくりのノウハウは、それ自体が、商品となる場合もあります。)

◇生産技術部門が利用し、出力する情報には、次のようなものが考えられます。

- ①. 製品情報: 新製品・既存製品の情報を得て、製造部門でどのように作り上げるのが適当かを検討し、効率よい製造手順(製造情報)を製造部門に提供します。既存製品であっても、製造工程(作り方)を見直すことで、大きく省力化・原価低減につながる事もあり得るので、既に製造されている製品にも気遣いが必要です。
- ②. 生産計画: 製造部門でのモノづくりが生産計画通りに進まない場合、原因を調べる必要があるので、生産計画を把握しておくべきです。
- ③. 資材状況: 製造に必要な資材を見直すことで作業工程が省けたり、必要な部材が省略できたりする。原価低減につながります。
- ④. 設備状況: 生産計画と生産実績の差が、設備に起因する場合もあるので、設備の使用状況や保守のタイミングや内容なども把握するのがベターです。
- ⑤. 人材状況: マンパワーが必要な場合、その時稼働できる人員の状況は、生産能力と等価です。熟練技能者や匠の技を持つ技能者への偏った負荷などにも注意が要ります。  
★製造部門全体を客観的に見なければいけません。
- ⑥. 生産実績: 製造部門で製品がどの様に(タイミング、数量、...)作られているかという情報。
- ⑦. 品質情報: 検査部門からもたらされる、検査実績に該当します。品質が変化して

いる場合、その原因を調べて対応をする必要があります。

- ⑧. 環境情報: 製造部門にある設備やロボット、現場に設置したセンサーなどから取得できる情報をすべて取り入れ、それらの内容から直接・間接的に製造現場の状況を把握しようとする取り組みに利用されます。自動的に常時取得される情報なので、Big Dataとなる。Dataが発生するのは製造部門ですが、生産技術部門でも共有して統計・分析します。このDataから現場の見える化を推す目的でIoTを取り入れることもあります。
- ⑨. 製造情報: 生産技術部門の製品ともいえる情報です。製造部門に与える、「モノづくりノウハウ」です。設備に設定する製品ごとのパラメータや自動機械やロボットのプログラムなども含まれます。

◇生産技術者としてできることは、沢山あります。

- ①. 企画・開発部門や、設計部門の上流にフィードバックする製品の設計変更要求。
  - ②. 検査部門に与える作り手側から見た検査指針(こうすれば、検査がやりやすい。高精度の検査ができるのではないか、といった情報。)
  - ③. 既に稼働している設備の保守やその計画。
  - ④. 設備性能向上のための改良提案。
  - ⑤. 新規設備の検討と導入提案。
  - ⑥. 製造工数(※1)低減のための現場レイアウトの検討や変更設計。
  - ⑦. 容易に可能な省力化(LCA(※2))の提案や実施。
- ・・・etc。

上げればきりがありません。それだけ充実した活動が求められる部門ということです。

◇生産技術部門は、上流部門(企画・開発、設計)と下流部門(製造、検査)との間にあり、モノづくりの中核を担うと言って良い部門です。

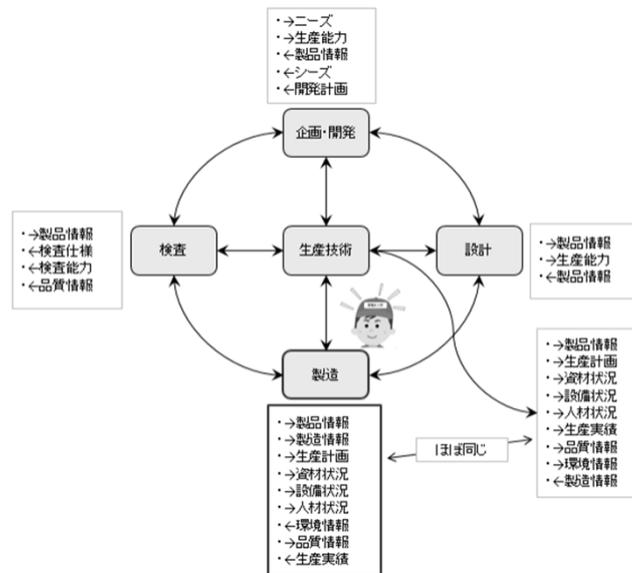
◇生産技術部門では、現場や設備、物的・人的資源などを客観的に見る目(観察力)が育つので、生産技術部門を経験すると【問題点の切り分けができる】技術者になります。さらに、IoT技術を身に付ければオールマイティな人材となります。

※このカリキュラムで学ぶ内容は、生産技術者としてのスキルを身に付ける内容となっています。

(※1)工数: 「工数」とは、ある作業を行うのに必要とされる「人数 × 時間」のことを指します。人日や人月などの単位で表現することがあります。他に、製造工程数を表す言葉として用いられることもありますが、本書では、前者の用途で使っています。

(※2)LCA : Low Cost Automation。費用を掛けずに、簡単に行う省力化や自動化の工夫。

## 製造部門



18

◇製造部門は、モノづくり企業の最重要部門です。生産技術部門と強い連携を持ちながら、高精度で安定した製品を作り出す部門です。製品製造のためには、次のような情報が必要と考えられます。

- ①. 製品情報: 設計部門から伝えられる製品のスペック情報
- ②. 製造情報: 生産技術部門から出力される、製品製造のノウハウを良く理解して、製造現場スタッフに伝える。また、設備やロボットなどに必要なデータの入力やプログラミングを製造部門で行うこともあります。
- ③. 生産計画: 生産管理部門が作製した計画に従い、量産品であれば一定のタクト(※1)で、一品モノ(※2)であれば予定の日時に製品を仕上げるための予定。
- ④. 資材状況: 製造現場では、原材料や部材、パーツなどが必要になります。生産計画通りに製造を行える資材が必要な分確保されているか、または順調に入荷しているかに配慮が必要です。
- ⑤. 設備状況: 設備は、機械製品であることが多く、稼動すれば消耗し、加工設備であれば加工精度も変化します。短期間の設備状況は現場で把握して、最小限の部品交換や消耗品(オイルなど)の補充などを製造部門で行います。また、モノづくりの内容と生産計画によっては、1日に数回製造する製品が切り替わり、段取り替えが発生します。この時に必要な工具などの準備も必要です。
- ⑥. 人材状況: 現場スタッフの人数や勤務時間の管理は現場で行います。製造部門が複数ある場合は、他の製造部門と進み具合の調整をしたり、応援をしたりするための管理も必要です。
- ⑦. 環境情報: 製造部門にある設備やロボット、現場に設置したセンサーなどから取得できる情報をすべて取り入れ、それらの内容から直接・間接的に製造現場の状況を把握しようとする取り組みに利用されます。自動的に常時取得される

情報なので、Big Dataとなります。生産技術部門と共有して統計・分析する。IoTを取り入れやすい対象です。

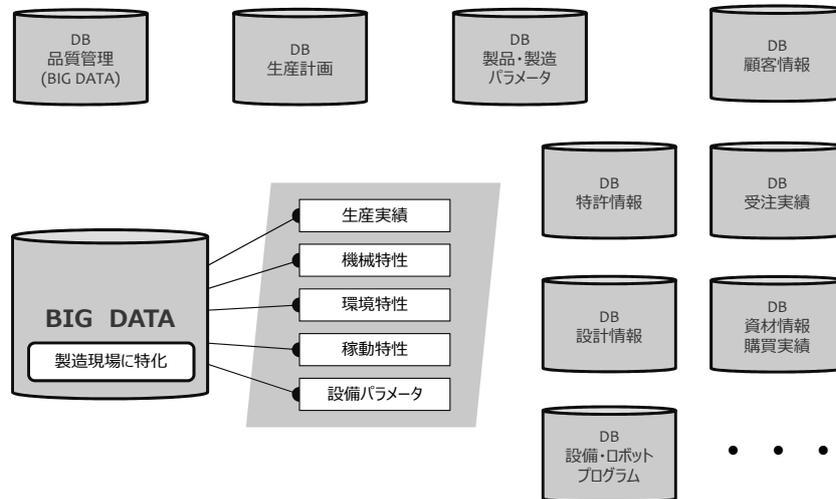
- ⑧. 品質情報: 検査部門から得られる品質情報を理解して、工具や刃物の摩耗、消耗品の不足、設備の故障などにつながる兆候がないか配慮します。
- ⑨. 生産実績: 生産計画に対して進み、遅れがないかに気を配ります。

(※1)タクト: 製造工程から製品が出来上がってくる時間間隔をタクトと言います。ある工程の作業1サイクルにかかる時間を指すことも有ります。一般に製造と検査では、かかる時間が異なります。例えば、検査工程が10分必要で、製造工程のタクトが5分であれば、製造工程にタクトを合わせるためには、単純に検査ラインを2つ設ける必要があります。生産管理部門はこのタクトと在庫状況(=保管スペース)などを考慮して生産計画を作成しなければなりません。工程数が多い場合には、工程ごとにタクトを決めて、ライン全体の同期がずれないように配慮も必要です。

(※2)一品モノ: 量産しない、文字通り1つしか作らないものを指します。大きな建造物や船、航空機、ロケットなど、ダムや橋梁の部材などは、ほとんどが一品モノですね。

## Step3 部門内情報

◇部門内情報 → 何処に、どのような情報が必要か



19

## Step3 部門内情報

◇部門内情報

◇モノづくり企業の情報には、

- ①. 部門ごとに必要な情報
- ②. 部門間で共有して役立つ情報

などがあります。

◇図はStep2で説明した部門ごとの情報ですが、Step3では、それらを詳しく調べます。

★以後、部門ごとに情報内容を分類整理していますが、同じ情報を複数の部門で共有することが多く、それらは重複して記載しています。

★以後の図では、データ項目ごとに一般的なデータ型(コンピュータ上で取り扱うデータの形式:数値、文字列など)も示します。この事は、今後データベースを取り扱う際に役立ちます。

## 営業部門で取り扱う情報 その1



・製品情報：お客様に知らせるべき製品の特長や価格など

- ①. 品番 → 文字・数字
- ②. 価格 → 数値（見積価格・定価・売価いろいろある）
- ③. 納期 → 日付・時間（時刻と同義）
- ④. 仕様 → 文字・数字・数値・画像

・シーズ：「このような技術なども持っています」という案内

- ①. 説明 → 文章・画像・他
- ②. 公開日 → 日付・時間

・ニーズ：お客様の企業活動に必要なモノ・こと・・・お客様からヒアリング

- ①. 品番 → 文字・数字
- ②. 価格 → 数値（購入希望価格）
- ③. 内容 → 文章・画像・その他



※図左側は、他部門の情報参照  
図右側はこの部門で作成する情報  
以後の図も 同様。

20

◇営業部門で取り扱う情報の代表的な例を示します。

（ここに上げる例は、あくまでもモデルであり、実際には、はるかに多くの情報を取り扱っています。以後も同じです。）

・製品情報：お客様に知らせるべき、自社製品の特長や価格を網羅する内容です。質問・問い合わせに答えられるようにデータベース化（DB化）されて、容易に検索できる必要があります。クラウド化されたDBシステムで、社外からのアクセスが可能になっている場合もあります。

- ①. 品番：製品を識別する番号や名称など。
- ②. 価格：取引数などにより、売価が変わる場合もあるので、見積価格や売価、定価など、別々にする場合があります。
- ③. 納期：製品をお客様に収められるまでの期間（リードタイムという場合もあります）
- ④. 仕様：製品の用途、規格、外観、性能、・・・など。企画・開発部門や設計部門で決められた仕様全てをお客様に公開するわけではありません。

★品番は、型番や製品名称を含んだ情報として考えています。

★文字・数字は、記号も含みます。

★時間は、時刻と時間の両方の意味で使用しています。

★上記は、以後の説明でも同様です。

・受注実績：いつ・誰から・何を・どれだけ・いくらで・いつまでに納める注文があったという記録。

- ①. 受注日: 注文のあった日(契約日)
- ②. 顧客番号: お客様を識別する番号(得意先番号、得意先コードなどとも言う)
- ③. 品番: 製品を識別する番号や名称など。
- ④. 数量: 注文数。具体的な数値に加えて、単位を別に記録する場合もあります。
- ⑤. 価格: 売価や契約価格。消費税などは別に記録されることが多い。
- ⑥. 納期: 受注日から何日後に納品するか。具体的な納品予定日付と時刻を記録する場合もあります。

・シーズ(※1): 自社の持つ技術や能力、人材、設備や商品など。営業活動の材料の一つ。

- ①. 内容: 文章・画像・他
- ②. 公開日: 情報を提供した日付

・ニーズ: お客様の企業活動に必要なモノ・こと。お客様からヒアリングして記録する。

- ①. 品番: 具体的な製品が明確であればそれを記録。
- ②. 価格: お客様が考えている購買価格帯。
- ③. 内容: 明確な製品がまだない場合、このような仕様の製品が欲しいなどの具体的な内容。

(※1)シーズ : 研究開発や新規事業創出を推進していく上で必要となる発明(技術)や能力、人材、設備などのこと。「ニーズ」という用語と対比で用いることもあり、その場合はニーズが客の要望によって必要とされるものである一方で、シーズはメーカー・企業などが必要に応じて提供する技術や商品などを意味する。

## 営業部門で取り扱う情報 その2

・顧客情報：お客様の企業活動の内容など

- ①. 顧客番号 → 文字・数字
- ②. 名称 → 文字・数字
- ③. 企業内容 → 文字
- ④. 取引開始日 → 日付・時間
- ⑤. 納品場所 → 文字

・受注実績：受注を受け納品した履歴

- ①. 受注番号 → 文字・数字
- ②. 顧客番号 → 文字・数字
- ③. 品番 → 文字・数字
- ④. ロット番号 → 文字・数字
- ⑤. 数量 → 数値
- ⑥. 価格 → 数値
- ⑦. 受注日 → 日付・時間
- ⑧. 納品予定日 → 日付・時間
- ⑨. 納品日 → 日付・時間

・営業実績：お客様に営業アプローチした履歴

- ①. 日付 → 日付・時間
- ②. 顧客番号 → 文字・数字
- ③. 担当者 → 文字・数字（社内の担当者番号・氏名など）
- ④. 応対者 → 文字（お客様側の応対者氏名など）
- ⑤. 営業内容 → 文章・画像・その他（具体的な活動内容）



21

・顧客情報：お客様の企業活動の内容、保有技術、特許や具体的な製品などの情報。購買対応窓口の具体的な担当者、連絡先など。クラウド化されたDBシステムで、社外からのアクセスが可能になっている場合もあります。

- ①. 顧客番号：お客様を識別する番号など。
- ②. 名称：お客様の名称、住所、購買担当者の氏名、連絡先なども含まれます。
- ③. 企業内容：お客様の企業活動の内容、保有技術、特許や具体的な製品などの情報。
- ④. 取引開始日：初回注文受注日・契約日。
- ⑤. 納品場所：注文品を納品する具体的な場所。購買・受入担当窓口の所在地や連絡先。

・受注実績：いつ・誰から・何を・どれだけ・いくらでいつまでに納める、注文があったという記録。納品時には、その実績を記録して完結する（消し込む）。

- ①. 受注番号：受注データの番号。製品出庫の際、受注番号と関連付けます。
- ②. 顧客番号：お客様を識別する番号など。得意先番号、得意先コードなどとも言います。
- ③. 品番：製品を識別する番号や名称など。
- ④. ロット番号：ロット生産している場合。後の追跡調査などに役立ちます。
- ⑤. 数量：注文数。具体的な数値とは別に単位を記録する場合があります。
- ⑥. 価格：契約価格。（消費税の取り扱い）
- ⑦. 受注日：注文を受付けた日。契約日。

- ⑧. 納品予定日: 納品予定日を元に生産計画が作製される。時刻も記録する場合があります。
  - ⑨. 納品日: 実際に納品した日。
- ・営業実績: 営業活動の記録。この記録を部門内で共有すれば、複数の担当者がアプローチできます。営業日報という形式で記録する場合があります。
- ①. 日付: お客様へのアプローチの日付。
  - ②. 顧客番号: お客様を識別する番号など。
  - ③. 担当者: 自社の営業担当者。(社員番号や社員コード)
  - ④. 応対者: お客様側の対応担当者。
  - ⑤. 営業内容: 具体的な活動内容。〇〇製品の説明。今後の注文の見通し打合せ。製品や試作品DEMOの様子など。
  - ⑥. 記録番号: 図にはないが、番号を付与し、受注実績と連携して、どのような活動が受注に結び付いたかなどを見る場合もあります。その場合には、受注実績にも同じ記録番号で紐づけします。

以上が、営業部門で取り扱われる情報ですが、ここで説明したもので完全かといえば、そうとは言い切れません。例えば、営業活動の中で、お客様から製品の在庫状況はどうですか？と尋ねられたとしたら、在庫状況も取り扱う情報に含めるべきです。仮に、ここに含めていない情報が必要になった場合は、企業内各部門と連携を図り、情報提供を受けることになります。

## 生産管理部門で取り扱う情報



・在庫状況：製品ごとの在庫数や在庫場所など

- ①. 品番 → 文字・数字
- ②. ロット番号 → 文字・数字
- ③. 数量 → 数値
- ④. ロケーション → 文字・数字
- ⑤. 在庫更新日時 → 日付・時間

・生産状況：生産実績ともいう

- ①. 品番 → 文字・数字
- ②. ロット番号 → 文字・数字
- ③. 数量 → 数値
- ④. 計画番号 → 文字・数字（生産計画との関連付け）
- ⑤. 実績更新日時 → 日付・時間

・生産能力：製造部門の能力。

- ①. 設備能力 → 文字・数字・数値
- ②. 作業人数 → 数値
- ③. ライン能力 → 数値

・資材状況：生産に必要な資材に不足があると、製造工程が進まない。

- ①. 資材番号 → 文字・数字
- ②. 在庫数量 → 数値
- ③. 発注点 → 数値

22

◇生産管理部門で取り扱う情報の代表的な例を示します。

・在庫状況：製品ごとの在庫数や在庫場所など。

- ①. 品番：製品を識別する番号や名称など。
- ②. ロット番号：ロット生産している場合。後の追跡調査などに役立ちます。
- ③. 数量：具体的な数量と単位。
- ④. ロケーション：在庫を保管している場所。同じ品番・ロット番号でも数量によっては、分割して保管される場合もあります。その場合、何処にあるのかは、大切な情報です。
- ⑤. 在庫更新日時：ここでは、生産部門、或いは検査部門からの情報によって常時新鮮な在庫情報が記録されていると考えています。そのため、何時の情報かを明確にします。

・生産状況(実績)(※2)：生産実績として、リアルタイムに完成した製品の数量が記録されます。本来、製品検査を通過したものが正味の生産実績となりますが、ここでは製造部門で記録される数量を対象としています。

- ①. 品番：製品を識別する番号や名称など。
- ②. ロット番号：ロット生産している場合。後の追跡調査などに役立ちます。
- ③. 数量：具体的な数量と単位。
- ④. 計画番号：生産計画に基づいて生産するので生産計画と関連付けます。
- ⑤. 実績更新日時：生産が進むにつれ実績数量が増加します。何時の時点かを明確に記録します。

【重要】:生産状況を計画完了した時点で記録するか、それとも生産が進む様子を把握できるように1点ずつ記録するかは、IoTシステムには重要な問題です。1点ごとに記録する場合、製品が出来上がるごとに記録するようIoTシステムを設計しなければなりません。  
(リアルタイムな状況把握)

- ・生産能力:製造設備の性能や台数、作業者の人数など多くの情報が含まれます。この情報は、通常変化することはありません。例えば日勤、2交代制、3交代制の変更や新しい設備導入、ライン変更時などには、変化が生じます。生産計画を作成する場合は、担当製造部門の生産能力に対する余裕を持った計画を立てなければ、計画は守れません。生産能力を設備毎、人員毎、ライン毎、あるいはサブ部門毎に正確に把握するのは難しいことです。

- ①. 設備能力:設備性能(対象製品・精度・単位時間当たりの製造能力など)設備ごとに管理するならば、設備の識別番号などが必要です。
- ②. 作業人数:個々人の熟練度や能力は、測ることが難しいことです製造部門のサブグループごとの実績を記録して、平均値を算出するなどの処理をします。
  - ★10人のグループで、日産100個→10個/日・人 という具合。
  - ★詳細に分析するならば、個人の能力を記録することとなるので、個人番号(社員番号など)が必要です。
- ③. ライン能力:設備+人員での能力。ラインごとに管理するならば、ラインを構成する、設備と人員の紐づけを行います。

★考えられる指標は、他にも沢山ありますがここでは、省略します。

- ・資材状況 : 計画を遂行するために必要な資材が、必要な量を毎日確保できるかどうか計画段階で把握しておく必要があります。もし、不足の可能性があれば、計画を変更して製造可能な他の製品を先に製造し、資材が準備できた段階で、該当の生産計画を実行するなどの対応が必要です。

- ①. 資材番号:資材を識別するコード。代わりに使える資材が分かり易いようにコード設計(体系を決める)をすることもあります。
- ②. 在庫数量:在庫の量。単位を別に記録する場合があります。(ガロン缶、リッター缶など)
- ③. 発注点:在庫管理の手法の中で、モノを仕入れる際の判断手法の一つに定点発注方式というのがあります。在庫数が発注点を下回ったとき発注する方式。←→定期発注方式(※1)

★資材状況も、製品の在庫管理と同じことを行います。対象が製品か資材か、消費される場所がお客様か現場かという違いです。  
★他にも管理できる情報はたくさんあるが、ここでは省略。資材を適正に管理すれば、無駄が減り利益の向上につながる場合もあります。

(※1)定期発注方式:一定の期間ごとに一定の量を発注する方式。定点発注方式も、

定期発注方式も、発注後、モノが納入されるまでの期間に、資材が消費される分を考慮する。発注数量は固定とする方式や、それまでの消費を加味して数量を変化させる方式など、さまざまなやり方があります。

(※2)生産実績(状況):何を見たい(把握したい)かによって項目が変わります。

例:製品ごとの出来高に注目 → 品番

設備ごとの出来高に注目 → 設備番号・ショップ

見たい項目は、記録しなければ見ることはできません。生産管理システムに後から項目を追加するのは大変手間がかかるので、初期の詳細な検討が大切です。

## 生産管理部門で取り扱う情報

・生産計画：製造部門の予定表

- ①. 計画番号 → 文字・数字
- ②. 品番 → 文字・数字
- ③. ロット番号 → 文字・数字
- ④. 数量 → 数値
- ⑤. 着手予定日 → 日付・時間
- ⑥. 完了予定日 → 日付・時間
- ⑦. 計画作成日 → 日付・時間



23

・生産計画：製造部門の予定表。この内容に従って、生産が行われます。生産計画は、製造部門の状態だけで決められるのではなく、お客様の意向や、受注の全体量、資材の状況など様々な要因を考慮したうえで作成されますが、大量生産品などは機械的な計算で計画を決めて、生産実績をフィードバックして計画を追加・訂正しながら生産が行われる場合もあります。

- ①. 計画番号：生産計画識別番号。生産実績と関連付けます。
- ②. 品番：製品を識別する番号や名称など。
- ③. ロット番号：ロット生産している場合。後の追跡調査などに役立ちます。
- ④. 数量：具体的な数量と単位。
- ⑤. 着手予定日：計画を実行開始する日付・時刻。
- ⑥. 完了予定日：製造完了予定日付・時刻。
- ⑦. 計画作成日：計画を作成した日付・時刻。

※計画変更の可能性もあるので、更新の都度記録します。

## 企画・開発部門で取り扱う情報



・ニーズ：お客様の企業活動に必要なモノ・こと・・・お客様からヒアリング

- ①. 品番 → 文字・数字
- ②. 価格 → 数値 (購入希望価格)
- ③. 内容 → 文章・画像・その他

・生産能力

- ①. 設備能力 → 文字・数字・数値
- ②. 作業人数 → 数値
- ③. ライン能力 → 数値

・製品情報

- ①. 品番 → 文字・数字
- ②. 仕様 → 文字・数字・数値・画像
- ③. 価格 → 数値
- ④. 生産

・シーズ：「このような技術なども持っています」という案内

- ①. 説明 → 文章・画像・他
- ②. 公開日 → 日付・時間



24

◇企画・開発部門で取り扱う情報の代表的な例を示します。

・ニーズ：お客様が望んでいる、自社の製品につながる、要求事項など。企業活動に必要なモノ・こと。主にお客様からのヒアリングによって、収集します。

- ①. 品番：製品を識別する番号や名称など。
- ②. 価格：購入希望価格。
- ③. 内容：具体的な要求の内容を記録する。必要に応じて、画像や動画などを利用して理解しやすい情報にします。

・生産能力：製造設備の性能や台数、作業者の人数など多くの情報が含まれます。この情報は、通常変化することはありません。例えば日勤、2交代制、3交代制の変更や新しい設備導入、ライン変更時などには、変化が生じます。生産計画を作成する場合は、担当製造部門の生産能力に対する余裕を持った計画を立てなければ、計画は守れません。生産能力を設備毎、人員毎、ライン毎、あるいはサブ部門毎に正確に把握するのは難しいことです。

- ①. 設備能力：設備性能(対象製品・精度・単位時間当たりの製造能力など)とも言います。設備毎に管理するならば、設備の識別番号などが必要です。
- ②. 作業人数：個々人の熟練度や能力は、測ることが難しいものです。製造部門のサブグループごとの実績を記録して、平均値を算出するなどしておきます。

- ★10人のグループで、日産100個→10個/日・人 という具合。
- ★詳細に分析するならば、個人の能力を記録することとなるので、

個人番号(社員番号など)が必要です。

- ③. ライン能力:設備+人員での能力。ラインごとに管理するならば、ラインを構成する、設備と人員の紐づけを行います。

★考えられる指標は、沢山ありますが、ここでは省略します。

・製品情報:企画・開発部門のアウトプットです。企画・開発段階の情報は、設計部門を経て、お客様に知らせるべき、自社製品の特長や価格が網羅されます。

- ①. 品番:製品を識別する番号や名称など。
- ②. 価格:取引数などにより、売価が変わる場合もあるので、見積価格や売価、定価など、別々にする場合があります。
- ③. 納期:製品をお客様に収められるまでの期間(リードタイムという場合もあります。)
- ④. 仕様:製品の用途、規格、外観、性能など。企画・開発部門や設計部門で決められた仕様全てをお客様に公開するわけではありません。

・シーズ(※1):自社の持つ技術や能力、人材、設備や商品など。営業活動に提供する材料。

- ①. 内容:文章・画像・他
- ②. 公開日:情報を提供した日付

(※1)シーズ:シーズとは、研究開発や新規事業創出を推進していく上で必要となる発明(技術)や能力、人材、設備などのこと。「ニーズ」という用語と対比で用いることもあり、その場合はニーズが客の要望によって必要とされるものである一方で、シーズはメーカー・企業などが必要に応じて提供する技術や商品などを意味します。

## 企画・開発部門で取り扱う情報



- ・開発計画：今後の製品ラインナップの充実化の内容
- ①. 開発番号 → 文字・数字
  - ②. 開発内容 → 文章・画像・その他
  - ③. 開発予定日 → 日付け・時間（開発時期）

- ・特許情報：保有する特許情報
- ①. 特許番号 → 文字・数字
  - ②. 種類 → 文字・数字
  - ③. 件名 → 文章
  - ④. 内容 → 文章・画像・その他
  - ⑤. 出願日 → 日付・時間
  - ⑥. 存続期限 → 日付・時間
  - ⑦. 出願者 → 文字・数字



25

・開発計画：自社製品を、今後どのような方向に発展・充実させていくかを示す情報。

- ①. 開発番号：開発計画の識別番号。
- ②. 開発内容：具体的な内容。目標。仕様など。
- ③. 開発予定日：開発を行う時期。

★これらを要約した情報を外部に公開することもあります。

・特許情報：保有している特許の内容など。

- ①. 特許番号：特許権番号。
- ②. 種類：特許の種類。特許・実用新案・意匠・商標の4種類。これらをまとめて工業所有権と呼びます。
- ③. 件名：特許の名称
- ④. 内容：特許の範囲や箇所、具体的な方法の説明や図面、画像など。
- ⑤. 出願日：出願した日付。
- ⑥. 存続期限：特許権が存続する期限。
- ⑦. 出願者：特許を出願した人。社員番号など。

## 設計部門で取り扱う情報



・製品情報：企画・開発部門が作製する新規製品情報（原始製品情報）

- ①. 品番 → 文字・数字
- ②. 仕様 → 文字・数字・数値・画像
- ③. 価格 → 数値
- ④. 総生産数 → 数値

・生産能力：製造部門の能力

- ①. 設備能力 → 文字・数字・数値
- ②. 作業人数 → 数値
- ③. ライン能力 → 数値

・製品情報：設計部門のアウトプット

- ①. 品番 → 文字・数字
- ②. 仕様 → 文字・数字・数値・画像
- ③. 価格 → 数値
- ④. 総生産数 → 数値

・特許情報：保有する特許情報

- ①. 特許番号 → 文字・数字
- ②. 種類 → 文字・数字
- ③. 件名 → 文章
- ④. 内容 → 文章・画像・その他
- ⑤. 出願日 → 日付・時間
- ⑥. 存続期限 → 日付・時間
- ⑦. 出願者 → 文字・数字



26

◇設計部門で取り扱う情報の代表的な例を示します。

・製品情報：企画・開発部門が作製する製品情報です。原始製品情報として取り扱い、設計部門では、これを詳細にします。

- ①. 品番：製品識別番号。
- ②. 仕様：製品の原始パラメータ。開発品として試作品が作れるレベルの内容。試作品そのもの場合もあります。（画像等も有り）
- ③. 価格：想定価格（原価、売価）
- ④. 総生産数：全体でどれ程生産するのか、それによって資材の調達規模や新規設備導入予算、人員計画などにも影響が出ることがあります。

・生産能力：製造部門の能力を定義するのは大変難しいが、原始製品情報の総生産数を決定する要因にもなる指標です。

- ①. 設備能力：設備性能（対象製品・精度・単位時間当たりの製造能力など）設備ごとに管理するならば、設備の識別番号などが必要です。
- ②. 作業人数：個々人の熟練度や能力は、測ることが難しいものです。製造部門のサブグループごとの平均した能力を記録して平均値を算出しておきます。  
★10人のグループで、日産100個→10個/日・人 という具合。  
★詳細に分析するならば、個人の能力を記録することとなるので、個人番号（社員番号など）が必要。
- ③. ライン能力：設備＋人員での能力。ラインごとに管理するならば、ライン

を構成する設備と人員の紐づけを行います。

- ・製品情報:設計部門のアウトプット。原始製品情報を詳細にして、生産技術部門・製造部門・検査部門でモノづくりが始められる情報です。

- ①. 品番 → 文字・数字
- ②. 仕様 → 文字・数字・数値・画像
- ③. 価格 → 数値
- ④. 総生産数 → 数値

- ・特許情報:保有する特許情報。外部の特許を情報として持つ企業もあります。社外特許に抵触しないように、また社内特許は積極的に活用して設計します。設計内容が特許出願できるかの判断材料にもなります。特許権が認められれば、それ自体が企業の商品にも成り得ます。

- ①. 特許番号:特許権の番号。
- ②. 種類:特許の種類。
- ③. 件名:件名。
- ④. 内容:特許の具体的な内容。
- ⑤. 出願日:出願日。
- ⑥. 存続期限:特許権の有効期限。
- ⑦. 出願者:社内であれば、社員番号。社外特許であれば、権利者名など。

## 検査部門で取り扱う情報



※以後、図の左側にあるのが入力情報、右側にあるのが出力情報である。

・製品情報：設計部門のアウトプット

- ①. 品番 → 文字・数字
- ②. 仕様 → 文字・数字・数値・画像
- ③. 価格 → 数値
- ④. 総生産数 → 数値

・検査仕様：検査部門のアウトプット（作業指示書：実作業をこれで行う）

- ①. 品番 → 文字・数字
- ②. 検査番号 → 文字・数字
- ③. 検査内容 → 文章・画像・その他
- ④. 検査規格 → 文字・数字
- ⑤. 検査時間 → 日付・時間

・検査能力：検査部門のアウトプット。検査部門の能力（正味の能力）

- ①. 品番 → 文字・数字
- ②. 検査番号 → 文字・数字
- ③. 正味時間 → 日付・時間

・品質情報：検査部門のアウトプット。検査実績

- ①. 品番 → 文字・数字
- ②. ロット番号 → 文字・数字
- ③. 検体番号 → 文字・数字
- ④. 検査番号 → 文字・数字
- ⑤. 検査結果 → 文字・数字
- ⑥. 測定値 → 数値



27

◇検査部門で取り扱う情報の代表的な例を示します。

・製品情報：企画・開発部門が作製する製品情報で、原始製品情報として取り扱い、設計部門では、これを詳細にします。

- ①. 品番：製品識別番号。
- ②. 仕様：製品の原始パラメータ。開発品として試作品が作れるレベルの内容。試作品そのものの場合もある。（画像等も有り）
- ③. 価格：想定価格（原価、売価）
- ④. 総生産数：全体でどれ程生産するのか、それによって資材の調達規模や新規設備導入予算、人員計画などにも影響が出ることがあります。

・検査仕様：検査部門のアウトプット。検査業務の作業指示書となり、実際の検査はこの仕様に基づいて行われます。

- ①. 品番：検査対象製品番号。
- ②. 検査番号：検査作業の識別番号。
- ③. 検査内容：具体的な検査の内容。手順など。
- ④. 検査規格：測定や検査の精度や判断基準など。
- ⑤. 検査時間：検査に必要な標準の時間。

・検査能力：検査部門のアウトプット。検査部門の正味の能力。

- ①. 品番：検査対象製品番号。

- ②. 検査番号:検査作業の識別番号。
- ③. 正味時間:検査に必要な最短の時間。これは能力の指標にもなります。この項目を検査仕様に加えて情報をまとめることもできます。

・品質情報:検査部門のアウトプット。検査実績。

- ①. 品番:検査対象製品番号。生産実績。
- ②. ロット番号:検査対象ロット番号。
- ③. 検体番号:ロットをさらに分解して、全数検査など行う場合は、シーケンス番号などを付けて記録します。
- ④. 検査番号:検査仕様の番号。どの様に検査したのかが追跡できます。
- ⑤. 検査結果:検査の具体的な結果。
- ⑥. 測定値:測定・計測の結果。

## 生産技術部門で取り扱う情報



・→製品情報：設計部門のアウトプット

- ①. 品番 → 文字・数字
- ②. 仕様 → 文字・数字・数値・画像
- ③. 価格 → 数値
- ④. 総生産数 → 数値

・→生産計画：製造部門の予定表

- ①. 計画番号 → 文字・数字
- ②. 品番 → 文字・数字
- ③. ロット番号 → 文字・数字
- ④. 数量 → 数値
- ⑤. 着手予定日 → 日付・時間
- ⑥. 完了予定日 → 日付・時間
- ⑦. 計画作成日 → 日付・時間

・→資材状況：生産に必要な資材に不足があると、製造工程が進まない。

- ①. 資材番号 → 文字・数字
- ②. 在庫数量 → 数値
- ③. 発注点 → 数値

・→設備状況：製造部門のアウトプット

- ①. 設備番号 → 文字・数字
- ②. 日付 → 日付・時間
- ③. 記録区分 → 文字・数字
- ④. 時間 → 数値
- ⑤. 記録日 → 日付・時間
- ⑥. 内容 → 文章・画像・その他

28

◇生産技術部門で取り扱う情報の代表的な例を示します。

・製品情報：設計部門で詳細に設計された製品パラメータ。

- ①. 品番：製品識別番号。
- ②. 仕様：製品製造パラメータ。量産品が作れる内容・精度。
- ③. 価格：原価・売価
- ④. 総生産数：全体でどれ程生産するのか、それによって資材の調達規模や新規設備導入予算、人員計画などにも影響が出ることがあります。

・生産計画：製造部門の予定表。

- ①. 計画番号：生産計画識別番号。生産実績と関連付けます。
- ②. 品番：製品を識別する番号や名称など。
- ③. ロット番号：ロット生産している場合。後の追跡調査などに役立ちます。
- ④. 数量：具体的な数量と単位。
- ⑤. 着手予定日：計画を実行開始する日付・時刻。
- ⑥. 完了予定日：製造完了予定日付・時刻。
- ⑦. 計画作成日：計画を作成した日付・時刻。

・資材状況：計画を全うするために必要な資材が、必要な量、毎日確保できるかどうか計画段階で把握しておく必要もあります。もし不足の可能性があれば、計画を変更し製造可能な他の製品を先に製造し、資材が準備できた段階で、該当の生産計画を実行するなどの対応が必要です。

- ①. 資材番号: 資材を識別するコード。代わりに使える資材が分かり易いようにコード設計(体系を決める)をすることもあります。
- ②. 在庫数量: 在庫の量。単位を別に記録する場合もあります。(ガロン缶、リッター缶など)
- ③. 発注点: 在庫管理の手法の中で、モノを仕入れる際の判断手法の一つに定点発注方式というのがあり、在庫が発注点を下回ったとき発注する方式。  
←→定期発注方式(※1)

★資材状況も、製品の在庫管理と同じことを行います。対象が製品か資材か、消費される場所がお客様か現場かという違いです。

★他にも管理できる情報はたくさんありますが、ここでは省略します。資材を適正に管理すれば、無駄が減り利益の向上につながる場合もあります。

(※1)定期発注方式: 一定の期間ごとに一定の量を発注する方式。定点発注方式も、定期発注方式も、発注後、モノが納入されるまでの期間に、資材が消費される分を考慮する。発注数量は固定とする方式や、それまでの消費を加味して数量を変化させる方式など、さまざまなやり方があります。

・設備状況: 設備の状態は、稼働していなくても日々変化します。設備の状況が見えるように(※2)常に最良の状態にするために、日々変化する設備の状態を記録します。ここでは、設備の状況を把握するために、一つのまとまった設備状況というデータで管理するモデルを考えています、実際には設備台帳、稼働履歴、保守履歴などに分けて管理することも多い。

(※2)見えるようにする=見える化です。これは、IoT活用の分かり易いポイントです。

- ①. 設備番号: 設備の識別番号。
- ②. 日付: 下記の記録区分の示す事象が発生した日。
- ③. 記録区分: 記録の種別。導入時期、稼働時間、総稼働時間、保守作業、保守時間、故障など。
- ④. 時間: 記録区分の事象にかかった時間。導入時期などの場合は、空欄か0。
- ⑤. 記録日: 記録した日時。
- ⑥. 内容: 保守内容や故障の原因などの具体的な説明。記録。設備の稼働率(※3)の元になる情報でもあります。

(※3)稼働率: 稼働率は、システムが正常に稼働している割合を示す指標。システムや人員が効果的に管理されているか、生産性、効率性を示す指標。

$$\begin{aligned} \text{稼働率} &= \text{MTBF} \div (\text{MTBF} + \text{MTTR}) \\ &= \text{実働時間(拘束時間 - 休憩時間)} \div \text{拘束時間} \end{aligned}$$

例: 勤務時間9時~18時(休憩1時間・実働8時間)社員の稼働率

$$\text{稼働率} = (9\text{時間} - 1\text{時間}) \div 9\text{時間} = 0.888\cdots (\text{約}89\%)$$

MTBF: 平均故障間隔。故障と故障の間の時間。  
MTTR: 平均修理時間。復旧・修理にかかる時間

## 生産技術部門で取り扱う情報



- ・人材状況：製造部門のアウトプット。
  - ①. 社員番号 → 文字・数字
  - ②. 経歴 → 文章・他
  - ③. 熟練度 → 文字・数字・数値
- ・生産実績
  - ①. 品番 → 文字・数字
  - ②. ロット番号 → 文字・数字
  - ③. 数量 → 数値
  - ④. 計画番号 → 文字・数字（生産計画との関連付け）
  - ⑤. 実績更新日時 → 日付・時間
- ・品質情報：検査部門のアウトプット。検査実績
  - ①. 品番 → 文字・数字
  - ②. ロット番号 → 文字・数字
  - ③. 検体番号 → 文字・数字
  - ④. 検査番号 → 文字・数字
  - ⑤. 検査結果 → 文字・数字
  - ⑥. 測定値 → 数値
- ・環境情報：製造現場の環境を記録。
  - ①. 部門コード → 文字・数字
  - ②. 測定位置 → 文字・数字
  - ③. 測定日時 → 日付・時間
  - ④. 測定区分 → 文字・数字
  - ⑤. 測定値 → 数値
- ・製造情報：生産技術部門のアウトプット。モノの作り方。
  - ①. 品番 → 文字・数字
  - ②. 手順番号 → 文字・数字
  - ③. 作業区分 → 文字・数字
  - ④. 作業内容 → 文章・画像・他



29

・人材状況：製造部門のアウトプット。モノの作り方を検討するための参考データです。

- ①. 社員番号：作業スタッフの識別番号
- ②. 経歴：これまでの作業経験
- ③. 熟練度：製造作業への熟練度。この指標を高精度で求めるのは大変難しいことです。例えば、保持資格、社内評価試験、作業コンテストなどの結果などから、企業独自の指標で表す場合もあります。この指標を元に、ラインレイアウト、人員配置、作業分担などを決めることもあります。

・生産実績：製造工程を通過した製品の数量が記録されています。

- ①. 品番：製品識別番号。
- ②. ロット番号：同、ロット識別番号。
- ③. 数量：生産数量。製造部門の稼動状況を見る場合と、生産計画に対しての到達度を見る場合があります。
- ④. 計画番号：生産計画識別番号。生産計画との関連付け。
- ⑤. 実績更新日時：実績データを記録・更新した日時。

★数量1単位ずつ更新するか、ロット単位で更新するかにより、IoTシステムの設計が変わります。

・品質情報：検査部門のアウトプット。検査実績

- ①. 品番:検査対象製品番号。生産実績。
- ②. ロット番号:検査対象ロット番号。
- ③. 検体番号:ロットをさらに分解して、全数検査など行う場合は、シーケンス番号などを付けて記録します。
- ④. 検査番号:検査仕様の番号。どの様に検査したのかが追跡できます。
- ⑤. 検査結果:検査の具体的な結果。
- ⑥. 測定値:測定・計測の結果。

・環境情報:製造現場の環境を記録。環境と生産実績、設備状況や製品の品質との相関を見出す目的で収集している場合が多い。

- ①. 部門コード:部門の識別番号。場所が分かります。
- ②. 測定位置:測定の位置。
- ③. 測定日時:計測日時。
- ④. 測定区分:計測の内容。温度、湿度、CO2濃度、振動、騒音レベル、照度、気圧・・・などの区別。
- ⑤. 測定値:実際の値。

★このデータは、設備や人材の稼働率に影響する可能性があるので、多くの企業でIoTシステムを構築し、自動計測・記録が行われています。IoTデバイスがコンパクトになり、無線でデータ取得ができるようになって来たので、記録されるデータの量が大きくなり、Big Data化する傾向にあります。このデータは定期的に分析を行う目的でも記録されることがあります。

・製造情報:生産技術部門のアウトプット。モノの作り方。

- ①. 品番:製品の識別番号。
- ②. 手順番号:作業の順番を表すシーケンス番号。
- ③. 作業区分:作業の名称。機械加工、ねじ止め、テープ貼り、組立、自動機、ロボット・・・etc。
- ④. 作業内容:作業の具体的な内容説明。この部分が作業指示となります。

★作業区分が自動機やロボットの場合は、プログラムを直接記録する場合もあります。自動機やロボットのプログラムを独立したデータとして管理する場合は、それらに関連付けるプログラム番号などを記録することもあります。今、自動機・ロボットがネットワークに接続することは当たり前になっており、生産計画をそれらが自力で取得して、人間が自動機などに表示される材料の供給や加工済みの部材の排出・搬送など、指示を受けて作業をする場合も多くなっています。

## 製造部門で取り扱う情報



### ・製品情報：設計部門のアウトプット

- ①. 品番 → 文字・数字
- ②. 仕様 → 文字・数字・数値・画像
- ③. 価格 → 数値
- ④. 総生産数 → 数値

### ・製造情報：生産技術部門のアウトプット。モノの作り方。

- ①. 品番 → 文字・数字
- ②. 手順番号 → 文字・数字
- ③. 作業区分 → 文字・数字
- ④. 作業内容 → 文章・画像・他

### ・生産計画：製造部門の予定表

- ①. 計画番号 → 文字・数字
- ②. 品番 → 文字・数字
- ③. ロット番号 → 文字・数字
- ④. 数量 → 数値
- ⑤. 着手予定日 → 日付・時間
- ⑥. 完了予定日 → 日付・時間
- ⑦. 計画作成日 → 日付・時間

### ・資材状況：生産に必要な資材に不足があると、製造工程が進まない。

- ①. 資材番号 → 文字・数字
- ②. 在庫数量 → 数値
- ③. 発注点 → 数値

30

◇製造部門で取り扱う情報の代表的な例を示します。

### ・製品情報：設計部門のアウトプット

- ①. 品番：製品識別番号。
- ②. 仕様：製品の原始パラメータ。開発品として試作品が作れるレベルの内容。試作品そのものの場合もある。（画像等もあり）
- ③. 価格：想定価格（原価、売価）。
- ④. 総生産数：全体でどれ程生産するのか。それによって、製造部門で該当製品が流れる期間がおおよそ判断できます。人員配置や設備の振分けの参考データとなります。

### ・製造情報

- ①. 品番：製品の識別番号。
- ②. 手順番号：作業の順番を表すシーケンス番号。
- ③. 作業区分：作業の名称。機械加工、ねじ止め、テープ貼り、組立、自動機、ロボット・・・etc。
- ④. 作業内容：作業の具体的な内容説明。この部分が作業指示となります。  
★作業区分が自動機やロボットの場合は、プログラムを直接記録する場合があります。プログラムを必要とする作業の場合、それらの作成や入力を製造部門で行いますが、場合によっては生産技術部門で行うこともあります。通常、量産体制のものは製造部門で処理します。

- ・生産計画:製造部門の予定表。この内容に従って生産が行われます。生産計画は、製造部門の状態だけで決められるのではなく、お客様の意向や、受注の全体量、資材の状況など様々な要因を考慮したうえで決められますが、大量生産品などは機械的な計算で計画を決めて、生産実績をフィードバックして計画を追加・訂正しながら生産が行われる場合もあります。

- ①. 計画番号:生産計画識別番号。生産実績と関連付けます。
- ②. 品番:製品を識別する番号や名称など。
- ③. ロット番号:ロット生産している場合。後の追跡調査などに役立ちます。
- ④. 数量:具体的な数量と単位。
- ⑤. 着手予定日:計画を実行開始する日付・時刻。
- ⑥. 完了予定日:製造完了予定日付・時刻。
- ⑦. 計画作成日:計画を作成した日付・時刻。

★計画変更の可能性もあるので、更新の都度記録します。

- ・資材状況:生産に必要な資材に不足があると製造工程が進みません。
- ・資材状況:計画を全うするために必要な資材が、必要な量、毎日確保できるかどうか計画段階で把握しておく必要があります。もし不足の可能性があれば、計画を変更して製造可能な他の製品を先に製造し、資材が準備できた段階で、該当の生産計画を実行するなどの対応が必要です。

- ①. 資材番号:資材を識別するコード。代わりに使える資材が分かり易いようにコード設計(体系を決める)をすることもあります。
- ②. 在庫数量:在庫の量。単位を別に記録する場合があります。(ガロン缶、リッター缶など)
- ③. 発注点:在庫管理の手法の中で、モノを仕入れる際の判断手法の一つに定点発注方式というのがあり、在庫が発注点を下回ったとき発注する方式。  
←→定期発注方式(※1)

★資材状況も、製品の在庫管理と同じことを行います。対象が製品か資材か、消費される場所がお客様か現場かという違いです。

★他にも管理できる情報はたくさんありますが、ここでは省略します。資材を適正に管理すれば、無駄が減り利益の向上につながる場合もあります。

(※1)定期発注方式:一定の期間ごとに一定の量を発注する方式。定点発注方式も、定期発注方式も、発注後、モノが納入されるまでの期間に、資材が消費される分を考慮する。発注数量は固定とする方式や、それまでの消費を加味して数量を変化させる方式など、さまざまなやり方があります。

## 製造部門で取り扱う情報



### ・設備状況

- ①. 設備番号 → 文字・数字
- ②. 日付 → 日付・時間
- ③. 記録区分 → 文字・数字
- ④. 時間 → 数値
- ⑤. 記録日 → 日付・時間
- ⑥. 内容 → 文章・画像・その他

### ・人材状況：製造部門のアウトプット

- ①. 社員番号 → 文字・数字
- ②. 経歴 → 文章・他
- ③. 熟練度 → 文字・数字・数値

### ・環境情報：製造現場の環境を記録。

- ①. 部門コード → 文字・数字
- ②. 測定位置 → 文字・数字
- ③. 測定日時 → 日付・時間
- ④. 測定区分 → 文字・数字
- ⑤. 測定値 → 数値

### ・品質情報：検査部門のアウトプット。検査実績

- ①. 品番 → 文字・数字
- ②. ロット番号 → 文字・数字
- ③. 検体番号 → 文字・数字
- ④. 検査番号 → 文字・数字
- ⑤. 検査結果 → 文字・数字
- ⑥. 測定値 → 数値

### ・生産実績

- ①. 品番 → 文字・数字
- ②. ロット番号 → 文字・数字
- ③. 数量 → 数値
- ④. 計画番号 → 文字・数字（生産計画との関連付け）
- ⑤. 実績更新日時 → 日付・時間



31

・設備状況：設備の状態は、稼動していなくても日々変化します。設備の状況が見えるように(※1)常に最良の状態にするため、日々変化する設備の状態を記録します。ここでは、設備の状況を把握するために、一つのまとまった設備状況というデータで管理するモデルを考えていますが、実際には設備台帳、稼動履歴、保守履歴などに分けて管理することもあります。

(※1)見えるようにする＝見える化 です。これは、IoT活用の分かり易いポイントです。

- ①. 設備番号：設備の識別番号。
- ②. 日付：下記の記録区分の示す事象が発生した日。
- ③. 記録区分：記録の種別。導入時期、稼動時間、総稼働時間、保守作業、保守時間、故障など。
- ④. 時間：記録区分の事象にかかった時間。導入時期などの場合は、空欄か0。
- ⑤. 記録日：記録した日時。
- ⑥. 内容：保守内容や故障の原因などの具体的な説明、記録。

これらは設備の稼働率(※2)の元になる情報でもあります。

(※2)稼働率：稼働率は、システムが正常に稼働している割合を示す指標。システムや人員が効果的に管理されているか、生産性、効率性を示す指標でもありません。

$$\begin{aligned} \text{稼働率} &= \text{MTBF} \div (\text{MTBF} + \text{MTTR}) \\ &= \text{実働時間}(\text{拘束時間} - \text{休憩時間}) \div \text{拘束時間} \end{aligned}$$

例: 勤務時間9時～18時(休憩1時間・実働8時間)社員の稼働率

$$\text{稼働率} = (9\text{時間} - 1\text{時間}) \div 9\text{時間} = 0.888\cdots (\text{約}89\%)$$

MTBF: 平均故障間隔。故障と故障の間の時間。

MTTR: 平均修理時間。復旧・修理にかかる時間

・人材状況: 製造部門のアウトプット。モノの作り方を検討するための参考データ。

- ①. 社員番号: 作業スタッフの識別番号。
- ②. 経歴: これまでの作業経験。
- ③. 熟練度: 製造作業への熟練度。この指標を高精度で求めるのは大変難しいことです。例えば、保持資格、社内評価試験、作業コンテストなどの結果などから、企業独自の指標で表すことがあります。この指標を元にラインレイアウト、人員配置、作業分担などを決めることがあります。

・環境情報: 製造現場の環境を記録します。環境と生産実績、設備状況や製品の品質との相関を見出す目的で収集している場合もあります。

- ①. 部門コード: 部門の識別番号。場所が分かる。
- ②. 測定位置: 測定の位置。
- ③. 測定日時: 計測日時。
- ④. 測定区分: 計測の内容。温度、湿度、CO<sub>2</sub>濃度、振動、騒音レベル、照度、気圧・・・などの区別。
- ⑤. 測定値: 実際の値。

★このデータは、設備や人材の稼働率に影響する可能性があるので、多くの企業でIoTシステムを構築し、自動計測・記録が行われています。IoTデバイスがコンパクトになり、無線でデータ取得ができるようになって来たので、記録されるデータ量が大きくなり、Big Data化する傾向にあります。このデータは定期的に分析を行う目的で記録されることもあります。

・品質情報: 検査部門のアウトプット。検査実績。

- ①. 品番: 検査対象製品番号。生産実績。
- ②. ロット番号: 検査対象ロット番号。
- ③. 検体番号: ロットをさらに分解して、全数検査など行う場合は、シーケンス番号などを付けて記録します。
- ④. 検査番号: 検査仕様の番号。どの様に検査したのかが追跡できます。
- ⑤. 検査結果: 検査の具体的な結果。
- ⑥. 測定値: 測定・計測の結果。

・生産実績: 製造工程を通過した製品の数量が記録されています。

- ①. 品番: 製品識別番号。
- ②. ロット番号: 同、ロット識別番号。
- ③. 数量: 生産数量。製造部門の稼動状況を見る場合と、生産計画に対しての遅れ進みを見る場合があります。
- ④. 計画番号: 生産計画識別番号。生産計画との関連付けです。
- ⑤. 実績更新日時: 実績データを記録・更新した日時。

★数量1単位ずつ更新するか、ロット単位で更新するかにより、IoTシステムの設計も変わります。

## その他の情報



・入出庫実績：在庫場所での製品の出入り情報。

- ①. 入出庫番号 → 文字・数字
- ②. 日付 → 日付・時間
- ③. 品番 → 文字・数字
- ④. ロット番号 → 文字・数字
- ⑤. 数量 → 数値
- ⑥. 入出庫区分 → 文字・数字（または数値）
- ⑦. ロケーション → 文字・数字
- ⑧. 受注番号 → 文字・数字

・売掛情報：納品した製品の請求情報を記録する。

- ①. 伝票番号 → 文字・数字
- ②. 日付 → 日付・時間
- ③. 請求日 → 日付・時間
- ④. 受注番号 → 文字・数字
- ⑤. 請求金額 → 数値
- ⑥. 入金予定日 → 日付・時間
- ⑦. 入金日 → 日付・時間
- ⑧. 入金額 → 数値

32

◇その他にも企業内で取り扱う情報は沢山ありますが、次の2つを追加しておきます。

・入出庫実績：製品検査も終わり、納品前の在庫として存在する製品を管理する情報。データを品番ごとにまとめると、総在庫量が分かり、原価を掛ければ総在庫金額が分かる。

- ①. 入出庫番号：入庫・出庫毎の記録のインデックス。
- ②. 日付：入庫・出庫の日付。
- ③. 品番：該当の製品番号。
- ④. ロット番号：ロット番号。
- ⑤. 数量：入庫・出庫の数量。
- ⑥. 入出庫区分：入庫・出庫の識別。0/1の数値を用いる場合もある。
- ⑦. ロケーション：在庫場所を示す番号や、名称など。
- ⑧. 受注番号：受注による入出庫であれば、該当する受注番号。注文はないが、在庫が目的の入庫の場合は、空欄。

・売掛情報：納品した製品の請求情報を記録する。企業の売上を計上する(全体の数値に組み入れる)タイミングは、契約時(受注時)、製品完成時、納品時、試運転完了時、請求時など、いろいろある。

- ①. 伝票番号：請求書の番号。
- ②. 日付：売上として計上する日付。
- ③. 請求日：請求した日。

- ④. 受注番号:どの注文に該当するかを明示。
- ⑤. 請求金額:請求した金額(消費税の取り扱い)
- ⑥. 入金予定日:お客様からの支払い予定日。
- ⑦. 入金日:支払いが行われた日。
- ⑧. 入金額:支払われた金額。

★請求額と入金額が異なる場合がある。経理上は、その差額を支払手数料等(経費)で処理することも有る。経費で処理すると、売上は減少しない。値引きとして処理すると、売上が減少する。企業としては重要事項。



やカンマ(,)区切りデータと言います。カンマ区切りデータは、広く利用されていてカンマで区切るのでCSV(Comma Separated Value)形式やCSVファイルと呼ばれています。CSVファイルは、Text Fileです。上の例をCSVで表現すると、スペースをカンマに置き換えて、

2018, 1116, 13, 30, A123, 0090

となります。更に、Text Fileで記録する際に、日付は2018/11/16とスラッシュ(/)を入れ、時間は13:30とコロン(:)を入れておくと、表計算ソフトやデータベースとデータの交換が行い易くなります。

例：2018 1116 13 30 A123 0090  
2018 1116 15 50 A124 0120  
2018 1116 16 50 B010 0070  
...

例：2018/11/16, 13:30, A123, 0090  
2018/11/16, 15:50, A124, 0120  
2018/11/16, 16:50, B010, 0070  
...

★データが文字の場合に、両端に半角ダブルクォーテーション(“)を付けて表現している場合もあります。

Text Fileで情報を記録する場合は、情報のまとまり毎にファイルを作成します。例えば、上の図で示す生産実績と機械特性は別のファイルになります。また、同じ生産実績でも、含まれるデータの種類が異なる場合は、別のファイルとして管理するのが一般的です。

Text Fileは1文字ずつ文字コード(※1)で表現された値が、ファイルに保存されます。だからText Fileなのですが、通常同じシステムで取り扱うText Fileの同じ文字コードは統一して、1つの文字コードを使用します。そのようにしないと、ファイルによって文字コードの変換が必要になって、処理が煩雑になってしまいます。

図、下の例では、データをデータベース(DB)化しています。データベースは、データを項目の値(文字・数値・時間など)毎に、最適な形の入れ物に記録する方法を採用しています。記録したデータを後で検索しやすいように、検索のキーとなる項目を決めてやると、その項目独自の索引が自動的に作られるので、検索の速度が速くなります。また、複数の種類のデータを関連付けてまとまりのある1つのデータとして取り扱うことができる、という長所もあります。膨大な情報を記録することができますが、使用するデータベースシステムの種類が異なると互換性がないので、利用するデータベースシステムの選択は、慎重に行う必要があります。

既に利用しているデータベースシステムを異なる種類のシステムに変更する場合には、既存データを変換する作業が必要になり、Big Dataがある場合は、作業工数(※2)が多くなります。専用変換プログラムを開発することもあります。

データベースと同様に、広く使われている表計算ソフトの表を利用する方法もあり

ます。表の場合は、入れ物の形は同じでも種類の異なるデータを記録できます。Microsoft Excel(2013)の場合、一つの入れ物をセルと呼び、セルが配置されているものをワークシート(sheet)と呼んでいます。シートには縦・横にセルが配置されていて、縦方向を列、横方向を行と呼びます。列はAから始まりB,C,D,...Zになり、次はAA, AB,AC,...ZZ、AAA,...XFDまで続きます。なにも記録されていないワークシートの場合、コントロールキー(Ctrlキー)を押下しながら右向き矢印キーを押下(これを、Ctrl+→ と表現する場合があります)すると、XFDの列に移動できます。同様にコントロールキーを押下しながら左向き矢印キーを押下する(Ctrl+←)とA列に戻れます。また、行も同様にコントロールキー押下+下向き矢印キー押下(Ctrl+↓)すると1048576行目に移動します。Ctrl+↑で、1行目に戻ります。行は理解しやすいですが、列はいくつあるのでしょうか？

問題:1枚のワークシートにあるセルの数はいくつか？

表計算ソフトは、Text Fileとのデータ交換が容易なので、その点が評価されてデータの記録に利用されている場合もあります。

問題:上の例に示したText Fileデータをそのまま、メモ帳ソフトなどで、Text Fileとして保存したものを、Excelで開いてみなさい。また、Excelで名前を付けて保存のファイル形式を、CSVファイルとして指定して保存したものをメモ帳で開いてみなさい。

【注意】表計算ソフトのワークシートには、膨大な数のセルがありますが、だからと言って全てのセルにデータを記録して良いわけではありません。大量のデータを記録したワークシートは、画面の表示やスクロール、データの編集などに長い時間が掛かり、仕事に支障をきたすので、ほどほどに利用するのが良いとされています。

(※1)文字コード:文字の見た目(イメージ)を容易に伝えるのは難しいので、例えば、「あ」は1、「い」は2というように、該当の文字を表現するためにつけられた文字に対応する値と、その体系。多くの文字コードがあります。ASCIIコードは、覚えておくと便利です。

(※2)工数:仕事を行う場合、どのくらいの時間が掛かるか(掛かったか)を時間と仕事をする(した)人数で表すことがあります。1人で1ヶ月掛かる仕事は、1人月の仕事といいます。

## 情報の保存方法

### ◇保存の具体的方法

- どのように Text File 化するか？
- どのように 表 化するか？
- どのように D B 化するか？

- ①. 具体的な道具
- ②. 具体的な道具の使い方
- ③. 具体的な情報の記録方法
- ④. 具体的な情報の取り出し方
- ⑤. 具体的な情報の関連付け方

これらを、自動化する方法を解説する。  
【Appendix A：Excel VBA によるプログラミング】で詳細に解説。

34

## 情報の保存方法

◇Text File・表・DB それぞれの場合の具体的な利用方法について説明します。

◇Text File はPCのメモ帳ソフトやテキストエディタ(※1)と呼ばれるソフトで、作成・編集ができます。表は表計算ソフト、DBはDBソフトを利用すれば、同様に作成・編集ができるし、検索もできます。しかし、企業内の全ての情報を人力でこれらのソフトウェアを使用して記録していたのでは、工数(※2)が多く掛かってしまいます。この一連の講座では、IoTを主題にしているので、ある程度の自動化を考えるのが当然です。ここでいう自動化には、プログラミングの知識が必要ですが、どのようにすればプログラムが作れて、自動化できるのかをこのStepで説明します。

(※1)テキストエディタ：フリーで公開されているもので優れたソフトが多くあります。今後プログラミングを行うことを考えて、あれこれ試して自分の気に入ったものを決めて習熟した方が良いでしょう。

(※2)工数：仕事を行う場合、どのくらいの時間が掛かるか(掛かったか)を時間と仕事をする(した)人数で表すことがあります。1人で1ヶ月掛かる仕事は、1人月の仕事と言います。

•Text File → メモ帳、Note Pad や テキストエディタ(※1)と呼ばれる、テキスト編集用ソフトが使える。それらを使い、文字情報をファイル化します。ファイル化したデータの取り扱いは、様々なプログラミング言語で処理する他に、シェ

ルスクリプトと呼ばれるUnixやLinux等のコマンドをText Fileに記述したものを、あたかもプログラムのように連続実行させてデータ処理する方法もあります。Windowsでのバッチ処理も同じように、コマンドプロンプトで実行できる機能を連続して記述したバッチファイルを実行することで、データ処理を行うのです。しかしコマンドプロンプトに含まれる機能が少ないので、IoTシステムと連携する目的で使用されることはないでしょう。

- ・表 → 表計算ソフトといえば Microsoft Excel を指すと考えられるほど、広く用いられていて、ほとんどの Windows PC にインストールされています。Open Office(または Apache Open Office)というフリーのオフィスソフトもあり、その中にExcel互換の Calc という表計算ソフトが含まれています。それらのアプリケーションを自動化するためのOpenOffice.org Basicという言語もあります。(※3)
- ・DB(データベース) → Excel 同様に Office ソフトの仲間の Microsoft Access、SQL Server、Oracle、MySQL、PostgreSQLなどが有名です。

IoTで活用するために、これらを自動化する方法はたくさんありますが、ここではExcelに含まれている、VBA(Visual Basic for Applications)の機能を使用する方法を採用します。VBAの使用方法を理解すれば、Text File、表(Excel WorkSheet)、DB(Accessや、Accessを通じて他のDBを使う方法)を使ったデータ処理が自由にできるようになり、IoTシステムと連携した情報処理システムを自前で構築することができます。実際にVBAで開発された沢山のシステムが多くの企業で稼動しています。

その方法は、【Appendix A:Excel VBA によるプログラミング】で詳細に解説しているので、参照してください。

(※3)OpenOffice.org Basic: Apache Open Officeで利用できるプログラミング言語でVBAとよく似ている。【VBAによるプログラミングが行えるのであれば、すぐにOpenOffice.org Basic をマスターできる。】とWebで紹介されている。

## Step5 部門間ネットワークの情報

◇部門間ネットワークの情報



・部門間**で**必要な情報 と



・他部門**が**必要な情報 がある。

データは、1か所で利用するのではない。データを作成した部門以外で利用されるデータが多い。

データを複数の場所から利用する方法が必要

【共有】機能を使うが、問題が生じる場合もある。

★最善の策は何か？

35

## Step5 部門間ネットワークの情報

◇部門間ネットワークの情報

このStepでは、部門を跨いで利用されるデータをどのようにして部門間で共有利用出来るようにするかということを考えましょう。

◇例えば、生産管理部門が作った生産計画を製造部門で見たい場合や、製造部門での製品完成が計画に対してどのような進み具合かを、生産管理部門から見たい場合などがこれに該当します。

この解決方法は文字通り【情報共有】です。各部門で作成・蓄積したデータをコンピュータ管理の下で、ファイルに保存し、そのファイルを共有します。これ以外にはありません。単にファイルを共有するだけであれば、難しいことは何もせずに、LANで接続したPC同士で、フォルダを共有設定すれば良く、そのフォルダに、内容の分かるファイル名で情報を保存すれば部門を跨いでデータが参照できるようになります。

Windowsであれば、フォルダプロパティの共有タブで設定できます。容易に実現できますが、安易に考えていると、次のような問題が起こる場合があります。

◇まず簡単にメモ帳で生産計画ファイル(名前はなんでもよい)を作成します。その中に、生産予定数(100)と生産実績数(今は0)を格納して保存します。

①. 生産予定数=100個で生産計画が作製され、製造部門が計画に従い生産

開始したとします。

②. 半分まで生産が進んだので、中間報告として生産実績数を50個と入力している最中に、計画変更があり、生産予定数=200個に変更しました。

③. 生産部門も中間報告を入力して保存しました。

★さて、生産計画はいくつになっているでしょうか？

ファイルを保存するタイミングで結果が異なることが注意点です。

①. 生産管理部門が後にファイルを保存した場合は、

生産予定数=200個      生産実績数=0個

②. 製造部門が後にファイルを保存した場合は、

生産予定数=100個      生産実績数=50個

となり、後に保存した情報で、ファイルが上書きされてしまいます。

このようなことを防ぐために【排他制御】が行われます。

◇排他制御：1つのファイルを誰かが開いている場合、参照は出来るが編集・書込ができない様になります。上の例では、メモ帳を利用したが、同じことをExcelで行うとどうなるでしょうか。既に開いているExcelファイルを他のPCで開くと、【他のPCで該当のファイルが開かれているので、参照モードで開く】という内容のメッセージが表示されます。後から開いた方は、参照はできますが書込みは出来ません。

これで、万事うまくいくかというと、そうでもありません。1つのファイルには複数種類のデータが保存されるでしょう。(先ほどの生産予定数と生産実績数など)。

通常、生産計画時には、計画そのものがまだ存在していないので、他の部門がその計画を見る事は無いから問題なさそうに思えますが、1つのファイルに1計画だけ保存することはないでしょう。沢山の計画が1ファイルに収まっているのが普通です。計画変更などは良くある事だし、生産実績は刻々と変化するので、頻りにファイルへのアクセスがあるはずで、この時、Excelのような排他制御が行われると、どうでしょうか。一瞬安心に思いましたが、誰かが、ただ参照する目的でファイルを開いていると、その間データの変更ができないので、リアルタイムに情報を管理するには向いていません。(★但し、ExcelファイルをExcelで開かず単なるファイルとして開き、高速で該当のワークシートの該当セルのデータだけ書き換える方法もあり、これは実際のシステムでも行われています。)

◇解決策の一つにデータベース(DB)の利用があります。データベースは、非常に多くのデータを効率的に管理するシステムです。他制御機能を含んだデータベースを巧く使えば、上のような問題は【ほぼ】解決します。【ほぼ】と書いたのは、実際にデータベースシステムを利用しているユーザーでも、システム設計に注意しない

と、時々巧く行かないことも現実には起こっているからです。しかし、現時点ではデータベースを利用することが最善の策でしょう。

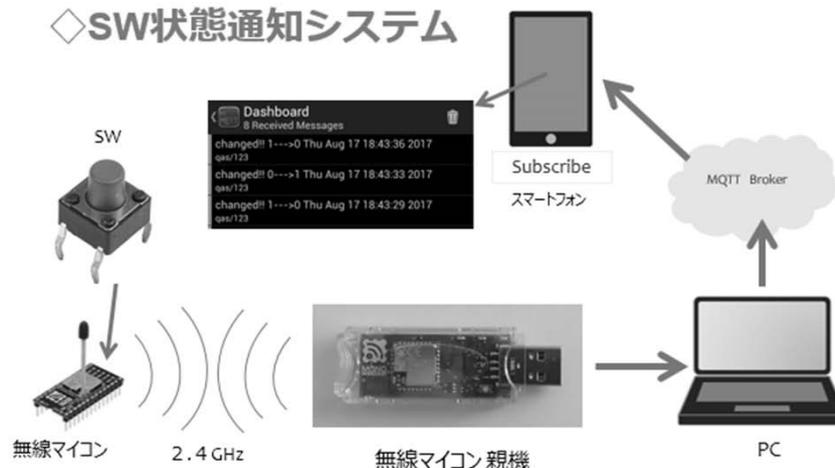
◇データベースの利用は、言葉で言うのは簡単ですが、情報管理を巧く行うためには沢山のプログラム(ソフトウェア)を作らなければなりません。しかし、それは自動でシステムティックに行う場合であって、手作業でデータベースにアクセスする場合は、基本的な使い方が分かれば十分です。

**★データベースの使い方については、別の機会に解説します。**

## Step6 部門間IoT導入ポイント

例：SWの状態をWEBに通知、スマートフォンで遠隔監視する。

### ◇SW状態通知システム



36

## Step6 部門間IoT導入ポイント

◇部門間でIoTを導入すると、モノがインターネットに繋がるので、情報の自動取得や状況把握を行い易くなります。ここで言う【モノ】はIoTデバイスそのものを指しています。

IoTデバイスは、様々な入力を持っているので、製造現場にある機械の状況や製品の出来上りを1個ずつWEBサービスに通知することなどが容易に実現できます。WEBサービス上の情報は、どこからでもアクセスができるので、部門間で情報を共有するためのWEBサービスの利用は、最良の方法と言えます。

◇IoTデバイスは、どの様にしてインターネットに情報を通知するかというと、メールのような仕組みと理解してもよいでしょう。

例えば、多くのIoTデバイスが各々の受け持ちの情報をWEB上のサービスに見出しを付けたメールで通知すれば、見出しでデータの種類が判別できます。

WEBサービスは、沢山の相手 (IoTデバイス) から送信されたメールを受信します。メールを利用したことも多いと思いますが、受信した順に見出しが並び(もちろん並び替えもできます)、見出しによって内容を判断して処理できます。IoTでは処理要求は、多くのIoTデバイスから沢山送信されますが、受信の段階で、1列に並べられ基本的に受信した順に処理されます。処理を行うのは単独のWEBサービスなので排他制御を心配する必要がありません。(メールの受信時間順に処理を行うというルールで

す。)

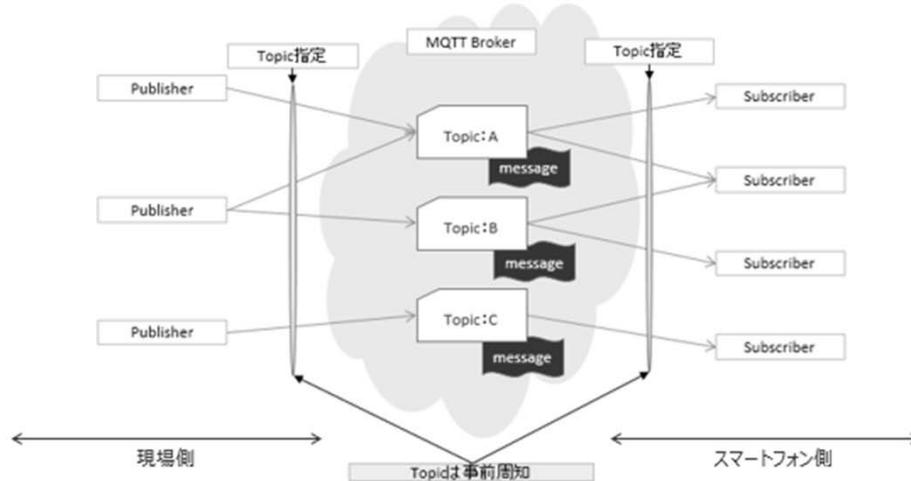
図に示すのは、SWの状態をWEB経由で通知するシステムの例です。SWのON/OFF状態を持ちますが、必要なのは、ON→OFFまたは、OFF→ONの変化があったことです。この変化を検出するのはマイコンの得意な仕事で、それをインターネット経由でMQTT(図中で雲の中にある)というWEBサービスに送っています。

MQTTと言うのは Message Queue Telemetry Transport という名称の頭文字です。これは、短いメッセージ(文字情報)に特化したWEBサービスで、送信者が見出し(Topicと言う)を付けてメッセージを送ると、同じ見出しで読み出し登録をしているユーザー側でメッセージが読めるというWEBサービスです。多くのコンピュータ言語用のライブラリが開発されているので、PCはもちろんタブレット、スマートフォン、マイコンでもこのサービスを利用することができます。(詳細は次の図で説明します。)

上図左側にあるSWを製造部門の機械に取り付けられた生産カウンタ用のSWと考えれば、このSWがOFF→ON→OFFと変化した時に製品が一つ完成したので、それをこのシステムでWEBに通知すると、時々刻々変化する生産実績を、複数個所で見ることができます。つまり、WEBを利用した現場の見える化が実現できます。このメッセージは、複数個所(何か所でもOK)で読み出しできるので、あるシステムでは、時々刻々変化する生産の様子をグラフでダイナミックに表示し、別のシステムでは、データベースにアクセスして、生産実績数を更新し、さらに他のシステムでは、生産予定数までの残数を現場に表示し・・・etc と、広がりのあるシステムが構築できます。

## MQTT : 短いメッセージの発行と購読

### ◇ Message Queue Telemetry Transport



◇MQTTはMessage Queue Telemetry Transportの略で、短いメッセージを頻繁にやり取りすることに特化したサービスです。

◇図はMQTTの利用方法を描いています。図の左側が現場(IoTデバイス)、右側がスマートフォン等です。中央の雲の中が、IoTデバイスが発行するメッセージを仲介するWEBサービス(MQTT Broker)です。

1. まず、メッセージの発行者側(Publisher:IoTデバイス)とそのメッセージの読者側(Subscriber:スマートフォン等)で、特定のメッセージを交換するためにTopicというキーワードを決めます。
2. 読者側は購読したいTopicをあらかじめMQTTサービスに登録しておきます。
3. メッセージ発行者がメッセージをTopicと共にMQTTサービス(MQTT Broker)に送ると、MQTT Brokerは、そのTopicのメッセージ購読希望者に対してメッセージを配信するというものです。

◇実際には、MQTT Brokerが登録されている全てのTopicと購読者に対してメッセージを送るのではなく、購読者側の端末プログラムがMQTT Brokerに問い合わせをかけて、新しいメッセージが発行されると、それを読み出す処理をしています。

◇購読者が何人いても良く、これまで実験したところではタイムラグもほとんど感じません。そして、ユーザー登録など必要のない無料利用できるMQTT Brokerも数多くあります。短いメッセージに特化しているからできることなのでしょう。

◇購読者側 (Subscriber) のシステムは、タブレットやスマートフォンアプリの利用も可能です。もちろんプログラミングを行って、独自システムを開発することも可能です。また、MQTT Brokerは、WEB上の外部サービスを利用する事もできますが、例えば社内サーバーなどでの運用も可能です。

## Step7 情報通信の方法

### ◇情報通信の方法

- ・インターネット通信  
↓  
TCP/IPプロトコル → OSに依存しない  
↓  
ソケットライブラリ（各種言語向け）  
↓  
Python による例示
- ・その他の通信  
↓
  - ①. パラレル通信
  - ②. シリアル通信 【Appendix Bに詳細解説】

38

## Step7 情報通信の方法

◇Step6で解説したMQTTのようなWEBサービスに、メッセージを送るようなインターネット通信は、TCP/IPプロトコルを利用しています。TCP/IPプロトコルは、OSに関係しないので、Windows、Linux、UNIX、MAC、Android間でも通信が行えます。このプロトコルを実装したマイコンがあれば、PCとマイコンをインターネットで接続できて、双方向で通信ができるのです。

◇これらのいろいろなOSやマイコン間のTCP/IP通信を行うには、作成するプログラムでTCP/IPを容易に使えるようにした【ソケットライブラリ】を利用することが多く、このライブラリを使えば複雑なTCP/IPプロトコルを使ったシステムが誰にでも開発できます。サーバーとクライアントとして、ソケットライブラリを使用したPythonのプログラム例を以下に示します。

- ◇サーバー側のプログラムが行うべきことは、
1. socketでソケットを作成
  2. bindでアドレスとポート番号を指定
  3. listenでクライアントの接続を待つ
  4. acceptでクライアントの接続を受け付ける
  5. sendやrecvを使ってクライアントのデータの送受信を行う
  6. closeでソケットを閉じる

の順で処理します。

サーバー側のPythonソースコード例:

```
#-----  
from _future_ import print_function  
import socket  
from contextlib import closing  
  
def main():  
    host = '127.0.0.1'  
    port = 4000  
    backlog = 10  
    bufsize = 4096  
  
    sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)  
    with closing(sock):  
        sock.bind((host, port))  
        sock.listen(backlog)  
        while True:  
            conn, address = sock.accept()  
            with closing(conn):  
                msg = conn.recv(bufsize)  
                print(msg)  
                conn.send(msg)  
    return  
  
if __name__ == '__main__':  
    main()  
#-----
```

◇クライアント側のプログラムが行うべきことは、

クライアント側では、

1. ソケットを作成
2. connectを使ってサーバに接続
3. 通信完了後にclose

です。

クライアント側のPythonソースコード例:

```
#-----  
from _future_ import print_function  
import socket  
from contextlib import closing  
  
def main():  
    host = '127.0.0.1'
```

```

port = 4000
bufsize = 4096

sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
with closing(sock):
    sock.connect((host, port))
    sock.send(b'Hello world')
    print(sock.recv(bufsize))
return

if __name__ == '__main__':
    main()
#-----

```

◇上記のプログラム例では、IPアドレスがローカルアドレスになっていますが、遠隔地とソケット通信を行う場合は、グローバルアドレスを指定します。

◇セキュリティを考慮した遠隔地とのソケット通信では、外部からのアクセスができないVPNを利用すれば、社内の一般的なローカル環境をそのままインターネットに乗せることができます。

◇上記のプログラム例にある `import socket` で取り込んでいるソケットライブラリは、他の言語用にも開発されています。言語が異なる環境であっても同じアルゴリズムが使えます。

★Appendix Aで説明したVBAにもソケットライブラリがあるので、Excelからでもインターネット通信が出来ます。

◇Pythonを利用した実際のインターネット通信は、演習編で行います。

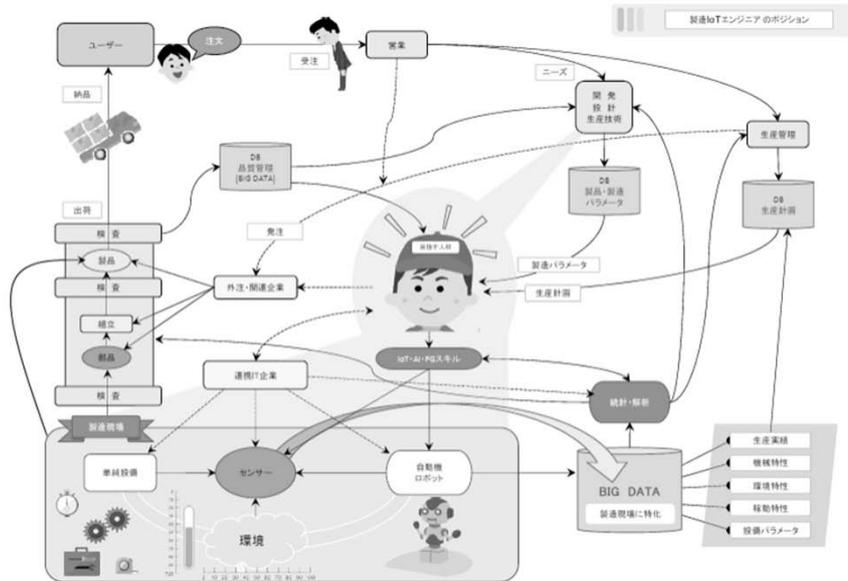
◇その他の通信：

初めの方のStepでIoTデバイスを分解・整理した際に解説した【その他の通信】には、大別するとパラレル通信とシリアル通信の2つの方式があります。

それらは、【Appendix B:シリアル通信】で説明しました。特にシリアル通信はいろいろな機器・コンピュータとの通信の基礎となる技術で、昔から広く利用されているので、是非参照しておいてください。

◇ここでは、詳しく解説しませんが、I2C通信 (I2Cインターフェース) やSPI通信 (SPIインターフェース) というシリアル通信の仲間は、マイコンとセンサーや表示器等、周辺デバイスとの間の通信で広く利用されています。これらの実際の使い方も、演習編で説明し、実習します。

## Step8 目指すIoT人材の立場



39

## Step8 目指すIoT人材の立場

◇Step2で示した図を再掲します。

この一連の講座も中盤を迎えました。ここで学ぶ方々の将来像を再確認しておきましょう。

◇中央に掲げられている人こそ、この教材で学んでいる皆さんです。

高度情報化社会の環境にあり、「モノづくりIoT」というキーワードで研鑽を続けているエンジニアは、IT・ICT・IoT技術を持つのはもちろんですが、製造の現場だけではなく、他の多くの部門と連携を図り、さまざまなデータを使いこなして、モノづくりを進めるのが理想です。現場で活躍できるためには、ある程度の実践が必要不可欠です。そのためにもこの一連の講座で全体を把握して、IoTに関する基礎スキルを身に付けていただきたいと思います。

◇【目指す人材の立場】として目指しているのは、「モノづくり現場」で活躍する【生産技術系の人材】を目標としています。その方の所属は生産技術部門であったり、企業によっては製造部門所属や開発部門所属のことも有り得ます。

◇勘違いしないで欲しいのは、所属が大切なのではなく、その人の行う仕事の内容と範囲をよく理解することが大切なのだということです。また、図では営業部門の方だけが、お客様(人)とコンタクトしているように見えますが、これは図の都合によるもので、実際には、人の周りには人だらけであり、多くの部門の人たち(時には社外のエンジニアやお客様)との人的な関係も当然必要となります。

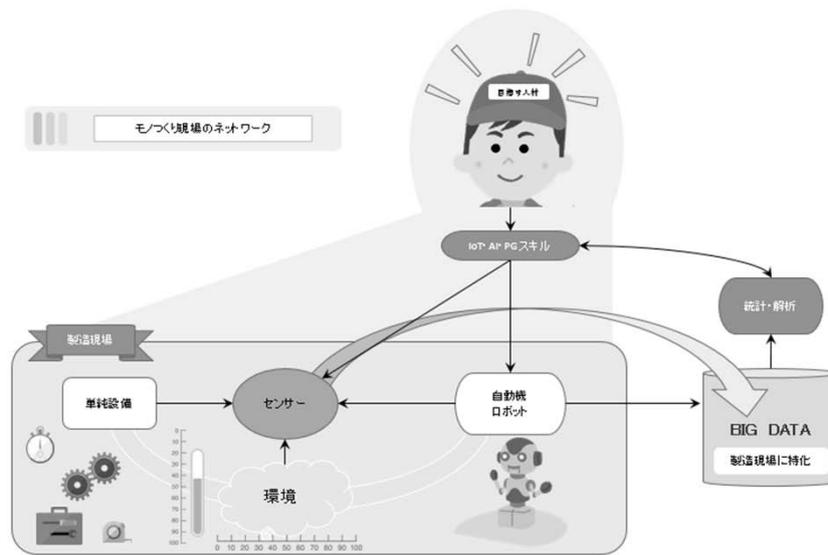
◇生産技術部を担うエンジニアは、実は人間臭い部分も多くあるのだと記憶しておいてください。

◇日本能率協会では、生産技術者マネジメントスキル認定資格というものを設定して、一定レベルの知識をもつエンジニアの認定を行っています。

(<https://jma-cpe.jp/outline/sample/>)

◇上記URLには、試験問題のサンプルが数問公開されています。この認定試験の試験問題サンプルは、回答例も含めて【Appendix C: 生産技術者マネジメントスキル認定試験サンプル】に掲載しました。この資格認定は、実務レベルで経験5年以上の技術者を想定しています。試験内容の幅は広いですが、資料などが入手できるので、ホームページを参照すると良いと思います。

## Step9 モノづくり現場のネットワーク



40

## Step9 モノづくり現場のネットワーク

◇図は、モノづくりの現場を見るエンジニアを描いています。現場には、プログラムの必要な自動機・ロボットやその他の、例えばコンベアやパーツフィーダ（部品搬送装置）など、比較的単純な設備も多く設置されています。またこれらが複合的に組み合わせられてできている設備もあるでしょう。

◇現場のエンジニアには単純設備にセンサーなどを取り付けて設備の状態などを取得できることが望まれます。取得したデータは、今後のためにデータベース化して記録されれば、なお良いでしょう。これらのことを行うためには、センサーの使い方やデータベース化の方法、そのデータを通信する方法などが理解されていて、【マイコンを活用したIoTデバイス】が自力開発できると、非常に効率の良い現場把握【見える化】システムができます。

★演習編も含めて、全て学習すると、IoTデバイス開発の基礎が身に付き、IoTシステムへの拡張の糸口が見えるようになるでしょう。

◇また、自動機・ロボットの様な、初めから各種センサーを内蔵していて通信機能を持っている装置からの情報も合わせて見ながら、製造現場全体の状態を把握しなければならないので、難しい仕事ではありますが、やりがいもあり企業もそのようなエンジニアに期待しています。

◇現場の環境を把握するためには、各種センサーを利用した【作業環境】への配

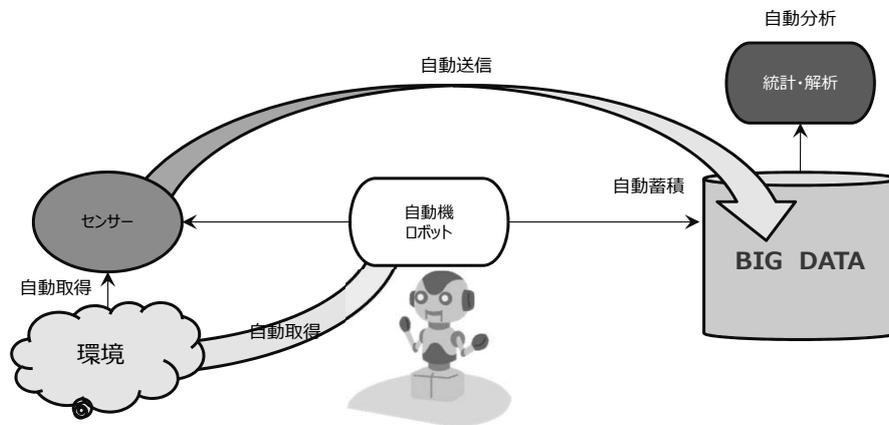
慮も必要です。ここで言う【環境】とは、【人への配慮】として気温や湿度、明るさ、コンベアスピードやCO2濃度・・・etcを指します。

◇気温や湿度、CO2濃度が高いと集中力が低下するなどの影響が出るので、そのような場合は生産している製品への影響が出ない範囲で、換気したり空調を調整するなどの人への配慮が要ります。また、コンベアスピードは一日を通して同じ速さではなく、作業開始時から少しの間は遅めに設定し、作業者の体が馴染んで作業が順調に進むようになった時点で通常の速さに調整する、などの配慮が必要になります。(これらは現場の担当者と協力して行うことが多い。)

◇これらのことも、【マイコンを活用したIoTデバイス】が作れると現場把握に対する負担が減少し、詳細なデータが蓄積できるようになります。この蓄積されるデータを統計・分析すれば、時間的に長い期間の設備の傾向や環境に対する人の馴染み具合などが把握できる可能性があります。このように、データ取得→蓄積→統計・分析の流れによって、設備が故障する前に保全作業(メンテナンス)を行う手法を予防保全と呼び、設備が故障しない様に稼働時間を長く保つ(稼働率を向上させる)ことが行われています。

## Step9 モノづくり現場のネットワーク

### ◇モノづくり現場のネットワーク



41

◇前の図で解説したようなデータは、自動的に取得・蓄積されるので、多数の設備を有する企業でなくとも、自然にデータ量が増えます。

大きくなったデータを言葉の通り【Big Data】と呼んで、それを統計・分析する手法などが開発されています。前述の予防保全に対して、AIの技術など使い、設備の傾向を分析・把握し故障を予知し保全する【予知保全】などの技術も実用化されています。このような、Big Data の処理を、現場にあるIoTデバイスの中で行うことはまだ難しいのですが、WEB上に配置されたホストコンピュータが受け取った情報をデータベース化しながら、定期的に統計・分析を行い、その結果を現場のIoTデバイスに通知し、メンテナンスの指示を出すことは可能になっていて、導入事例も多くなっています。

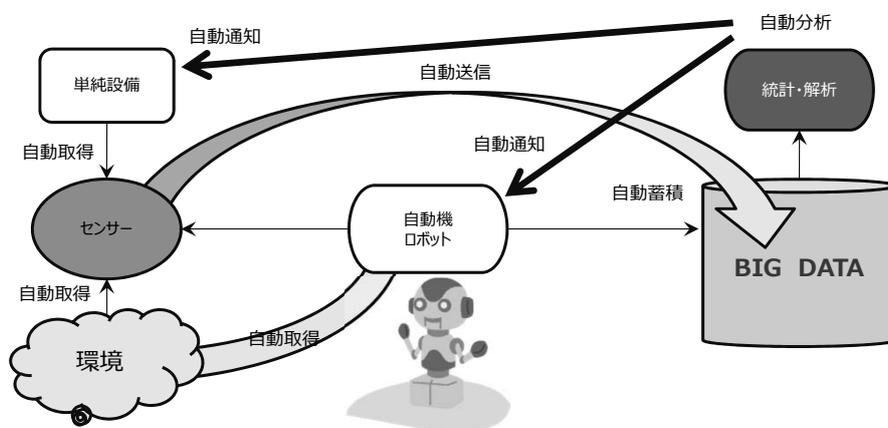
【統計・分析・AI】などのキーワードで表現されている知識を持つエンジニアは、高く評価されています。

これらのことを実現するためにはネットワークが不可欠であり、その通信手法として、TCP/IP等によるインターネット通信やシリアル通信があり、その実現方法として有線通信と無線通信があります。製造現場では、既に稼働している設備のある工場内などで、無線通信の利用が今後多くなっていくでしょう。

以後のStepで解説しますが、単独の基板内に無線機能やWiFi機能を持つCPUを使ったマイコンが開発されています。

## Step10 モノづくり現場のIoT導入ポイント

◇モノづくり現場のIoT導入ポイント



42

## Step10 モノづくり現場のIoT導入ポイント

◇単純設備が稼働しているか、停止しているかを判断するには、いろいろな方法があります。

- ①. 開始(電源ON)SWの状態を監視する。
  - ②. 主軸の回転を監視する。
  - ③. 設備全体の振動を監視する。
  - ④. 流れる電流を監視する。
- ・・・etc

その内容は、

- ①. SWの状態 → 接点が空いていれば(★1)それを利用してマイコンへの入力回路を構成してDIで読み込める。もし空きが無い場合は、開始(電源ON)の回路にリレーを追加して、そのリレーの接点でマイコンへの入力回路を構成して、DIで読む。
- ②. 主軸の回転 → 回転軸に近接センサーや反射センサーで非接触に検出し、信号が一定時間発生しない場合は停止と判断する。同様にスリットを設けた円盤を回転軸に取り付けて、スリットを挟む形で通過型光センサーを設置して、円盤の回転を検出する方法もある。
- ③. 設備の振動 → 加速度センサーやジャイロセンサーで、設備の揺れを検出する

ことができる。

- ④. 電流を検出 → 設備が稼働中の電流を検出する電流センサーを利用して検出できる。

(★1)SWやリレーには、複数回路の接点を持つものがあります。設備の運転開始(電源ON)のSWが複数回路ならば、未使用の回路の接点を使い、マイコンへの入力を行いません。通常マイコンの電源電圧はDC3.3V~5Vであり、設備はAC100~200Vなので、両方の回路を直接接続することはできませんので、この様にして回路を分離します。

◇設備や環境の温度を計測するにはどうすれば良いでしょうか。

単純に温度センサーを利用すればよいのですが、計測する温度範囲を決めて、その被測定温度の範囲(レンジ)と精度を持つ温度センサーを選択します。温度センサーには、アナログセンサーとデジタルセンサーがあります。アナログセンサーは温度に対応する電圧が出力されるものが多いのですが、その電圧をマイコンで計測するにはADC(A/D変換器)を利用して、電圧をデジタル値に変換しなくてはなりません。変換した電圧を基に温度換算して温度を知ることができます。

◇デジタルセンサーの場合は、センサーごとに設計されている通信コマンドにより、温度読込を行います。温度は直読できますが、センサー毎に異なるコマンドを理解して利用する必要があります。

◇アナログ・デジタルどちらのセンサーを使うにしても、センサーを開発したメーカーが発行するデータシートを良く理解して既定の定格値の範囲内で使用します。

◇湿度、変位、圧力・・・など多くのセンサーが開発されているので、利用の可能性のあるセンサーの情報を普段から収集し、機会があれば試用(あらかじめ試す)してみることがイザというときにとても役立ちます。

◇生産実績を自動カウントするには、どうすれば良いでしょうか。

最も簡単な方法は、SWを利用する方法です。例えば、設備から製品が排出される経路にSWを置いて、製品の通過でSWが押されるようにします。設備の製品排出動作でSWが押されるようにしても良いでしょう。マイコンでSWの状態を監視すれば、製品が排出される度にSWの信号はON/OFFを繰り返すので、これをカウントすれば良いでしょう。カウントして一定時間ごとに、ホストコンピュータ(WEB・クラウド)に通知するのか、排出される度に通知するのかは、システム全体の仕様によります。SWの代わりに、通過型光センサーや近接センサー、反射型のセンサーなど多くのセンサーが使えます。

◇多くの設備や人が処理している場所で、1個の製品が完成するためには、その材料が必要になります。自動機などでは、1個ずつワーク(作業の対象物)や材料を取り付ける場合もありますが、パーツフィーダなどで自動投入される場合もあります。この時、その場所(ショップとも言う)にある材料が不足しそうな事を検出し、前もって材料を供給できれば、設備を止めずに生産が継続出来て、設備の稼働率が向上します。これも、IoT導入の良いポイントです。パーツフィーダなどは材料が残り少なくなると信号を発するものもあるし、部材だまり(材料や部材を投

入しておく部分)に触れるSWを設けて、その枯渇を検出できます。その情報をIoTデバイスで通知すれば稼働率も上がり、一人で複数の設備を管理できる可能性も出てきます。このSWの情報を蓄積すれば、現場の設備担当者がどれくらい忙しいかも把握できます。このデータを統計・分析すれば、各設備毎にあらかじめ必要な材料や部材の数量が把握できます。可能であれば、ある程度まとまった数量を設備前に、或いは材料だまりに準備できれば、頻繁な供給作業は、その回数を減らせることになり、その製品に対する作業工数が減少して原価低減につながります。

◇これまで説明したように、検出した情報は、IoTデバイスで記憶しただけでは利用しにくいので、ホストコンピュータへ通知することが必要です。ホストコンピュータはローカルに現場付近にあるかもしれないし、WEB上にあるかもしれません(クラウド)。何らかの形で通信を行う必要が有ります。ホストコンピュータで統計・分析された結果を現場設備に通知するのも同様です。この時、有線でも無線でも通信できる手段をエンジニアが提供できることが必要です。図で示す自動送信・自動通知がそれに該当します。

◇IoT導入ポイントをまとめると、

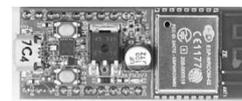
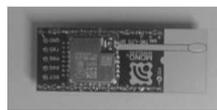
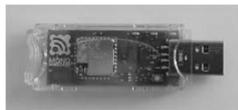
1. 自動設備・ロボットも含む生産設備とその付近
2. SWで検出できる事象が基本
3. 非接触のセンサーで検出できる事象
4. 配線可能なら有線で。無理なら無線で。無線を使う方法は演習編で、詳しく解説する。
5. 環境情報で特殊なものに向けて、常に新しいデバイスなどの情報を収集する。
6. ホストコンピュータからの通知を受け取り分かり易く知らせる。  
(表示・音・・・etc)

**【Appendix D: SWの情報を検出する2つの方法】で、マイコンへのSW接続の方法を説明しています。**

## Step11 IoTシステム開発の方法

### ◇IoTシステム開発の方法

1. IoTデバイスからIoTシステムへ
2. IoTデバイス開発
3. IoTシステム開発



43

## Step11 IoTシステム開発の方法

### ◇IoTデバイスからIoTシステムへ

IoTは、情報の可視化に有効との考え方から、現場での情報取得とホストコンピュータへの送信にも利用されています。それを行うのがIoTデバイスです。ホストコンピュータでは、受け取ったデータを蓄積しながら、必要な分析を行い、その結果を現場にフィードバックします。ホストコンピュータ周辺ではフィードバック情報も【見える化】されているでしょう。

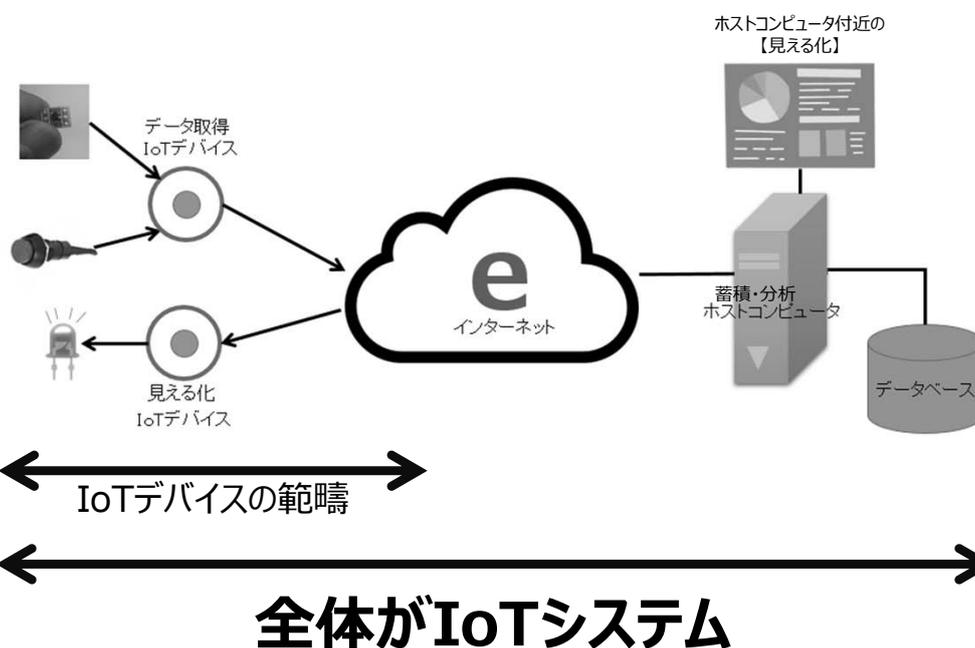
そのとき現場でフィードバックを受ける仕掛けもIoTデバイスの役目となります。受取ったフィードバック情報は、表示装置などを利用して人に通知されます。それが【現場での見える化】となるのです。自動機・ロボットなどの場合は、それらが直接フィードバックを受ける場合もあり、そうなれば、ホストコンピュータが多数（群）の設備をコントロールする群管理が実現できるのです。

整理すると、

- ①. IoTデバイスは、現場の生データを取得してホストコンピュータに送信。
- ②. ホストコンピュータは、受取ったデータを蓄積しつつ、分析する。
- ③. 分析結果を現場のIoTデバイスに送信する。
- ④. IoTデバイスは、ホストから受取ったデータを現場に通知する。

これを図にしたものを次に示します。

## IoTデバイスとIoTシステム



44

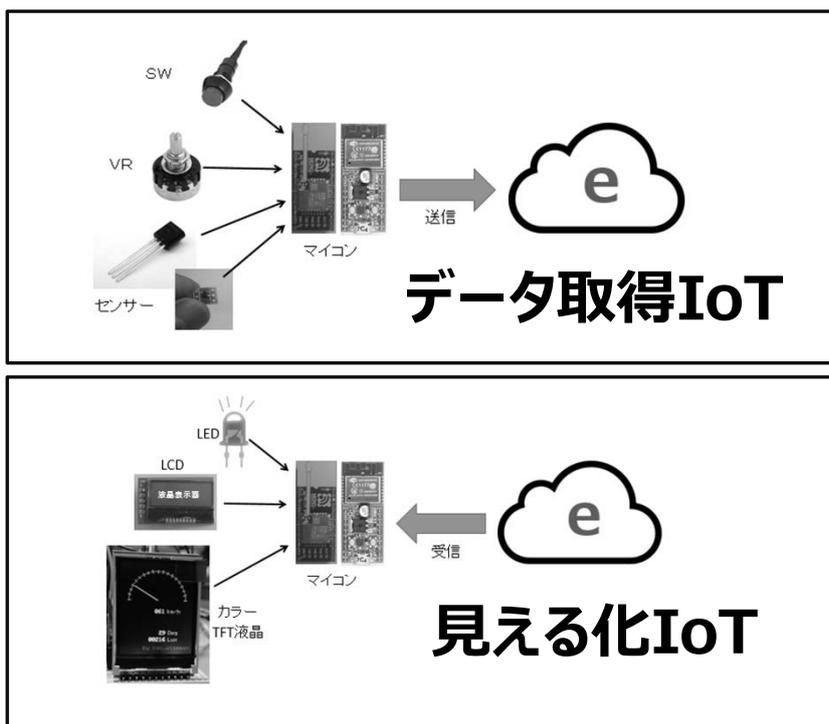
◇言い換えると、IoTデバイスは、【生データ取得&見える化端末】で、IoTシステムは、【IoTデバイス+ネットワーク+ホストコンピュータ+データベース全体】と考えられます。

◇図のインターネットより左側をIoTデバイスの範疇と定義し、演習はこの範囲を中心に行いますが、システムとして必要になるデータ蓄積についても触れています。

◇Step7で解説したソケット通信を用いれば、Pythonを用いてIoTデバイスとの通信が可能なので、ホストコンピュータ側で同様のプログラムを開発することもできます。また、Appendix:Aで示したVBAを用いて、ソケット通信を行えばIoTデバイスから得たデータを簡単なテキストファイル形式で蓄積することもできます。これでIoTシステム開発の道筋はおおむね理解できるでしょう。不足しているとすれば、情報のデータベース化の方法と、欲を言えば Big Data の統計・分析ですね。

★データベース取り扱いについては、演習編で説明します。Big Data については、別途教材を開発する予定になっています。

## IoTデバイス開発



45

◇IoTデバイス開発は、

1. データ取得IoTデバイス開発
2. 見える化IoTデバイス開発

の2つに分類できます。

◇最近の、電子回路設計・製造技術の進歩でマイコンの性能を向上させているので、データ取得と見える化を同時に行い、それに加えて設備の制御を行うシステムを開発することも可能になっています。自動機やロボットがそれに該当します。これらはIoTデバイスとは別のものとして取り扱います。

◇ここでは整理するために上の2つに分けましたが、両者の違いは入力【DI】か出力【DO】かの違いだけで、本質的には何も変わりません。

◇それぞれの図の中央にある2つのマイコンは、無線機能を内蔵しています。左側は、同種のマイコン間で無線通信ができるものです。右側に示したマイコンは、単独でWiFiに接続できます。

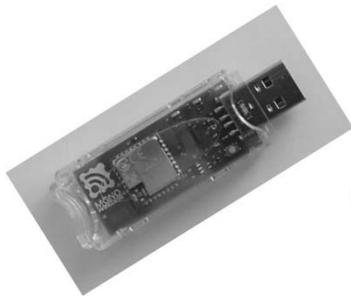
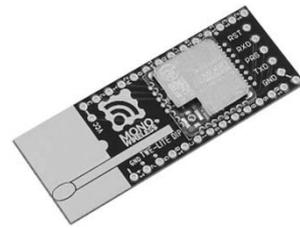
何故2つ示したかという、左側のものは無線通信の可能な距離は長い(およそ1000m)がWiFiへの接続は出来ず、右側はWiFi接続(インターネット接続)はできるが距離は短い。と、それぞれ長所・短所があります。これら両方の長所を備えたコンピュータが出現するのも時間の問題だと思われませんが、それまで待つことと、その際の費用を考えると、上の2機種のマコンを組み合わせる利用できるスキルを身

に付けた方が、IoTデバイス開発を身に付ける早道だと思うからです。

◇2つのマイコンは、他にもそれぞれ特徴があります、それは後のStepで解説することになります。

## Step12 無線IoT入門

- ◇TWE-Lite – トワイライト
- ◇標準アプリ内蔵
- ◇配線するだけで使える
- ◇マイコンに独自アプリを書き込める
- ◇中継機能で距離を延ばせる



46

## Step12 無線IoT入門

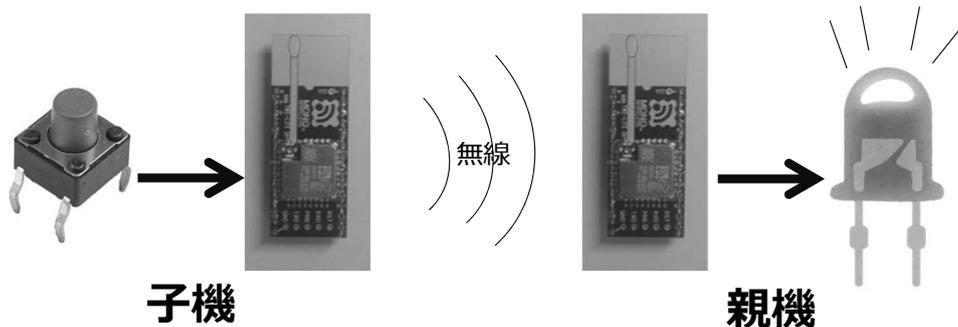
- ◇Step12は、【無線機能を持つマイコン】を使ったIoTの入門解説です。
- ◇図に示しているいくつかの写真は、いずれもTWE-Liteと書いてトワイライトと呼ぶ無線機能内蔵のマイコンです。以後、このマイコンを指して、【無線マイコン】と呼ぶことがあります。
- ◇マイコンは、目的の動作を行うプログラムを開発してメモリに書込んで動作させるのが一般的ですが、この無線マイコンは工場出荷時に標準動作を行うアプリケーションが既書き込まれているので、そのアプリ機能の範囲内であれば、プログラムを作ることなく回路を作るだけで動作させることができます。
- ◇もちろんプログラムを作成して新しいシステム開発も可能です。
- ◇アプリケーション開発環境は、ソースプログラムなど全て公開されています。メーカーWEBページからダウンロードできるので、新規システム開発の参考になります。開発言語はC、C++です。IDEはEclipseです。
- ◇電源はDC3Vで、乾電池駆動が可能です。
- ◇無線到達距離は、およそ～1000m未満ですが、設定により中継機能を有効にしたモジュールを経由すれば、全体の通信距離を延ばすことができます。

- ◇図左下に示す写真のモジュールは、USBコネクタが付いていてPCにそのまま接続できるものです。PCに接続するとCOMポートとして認識され、これを經由してシリアル通信が可能になります。以後このモジュールを指して【USBモジュール】と呼ぶことがあります。シリアル通信の相手には、図の右上、右下、中央下のモジュールが対応できます。これらも、工場出荷時の標準アプリの機能の範囲です。
- ◇フィールド(現場)にセンサーを取り付けた無線マイコンを置きPCにUSBモジュールを接続すれば、無線通信によるデータセンシングが出来るようになります。しかも、プログラミングレスです。
- ◇このUSBモジュールは、USBホスト変換アダプタを利用するとPCだけでなく、タブレットやスマートフォンにも接続できます。PC、タブレット、スマートフォン用のアプリケーションも公開されているので、無線を利用したセンシングが容易に実現できるようになっています。
- ◇PC、タブレット、スマートフォン側では無線を意識することなく、通常のシリアル通信を行うプログラムを開発すれば自由に無線マイコンと通信ができます。PC、タブレット、スマートフォンで現場に置いた無線マイコン(IoTデバイス)からデータを受信しインターネット通信すれば、IoTの目的は達せられます。
- ◇シリアル通信のプログラム開発は難しくありません。演習では、Pythonを使用したシリアル通信で、USBモジュールと通信するシステムを開発します。
- ◇【Appendix E】にこの無線マイコンモジュールのメーカーデータシートの抜粋を掲載しています。IoTデバイス開発では、色々なデバイスのデータシートを読むことになるので、参照してどのような特徴があるのか理解しておいてください。

## LED点灯の遠隔操作・・・

### <<無線を使う>>

・・・配線するだけです・・・



47

◇上図はこのTWE-Liteを使用して、LEDを遠隔操作で点灯させるシステムの概要です。分かり易く送信側(子機:SW担当)と受信側(親機:LED担当)に分けて描いています。

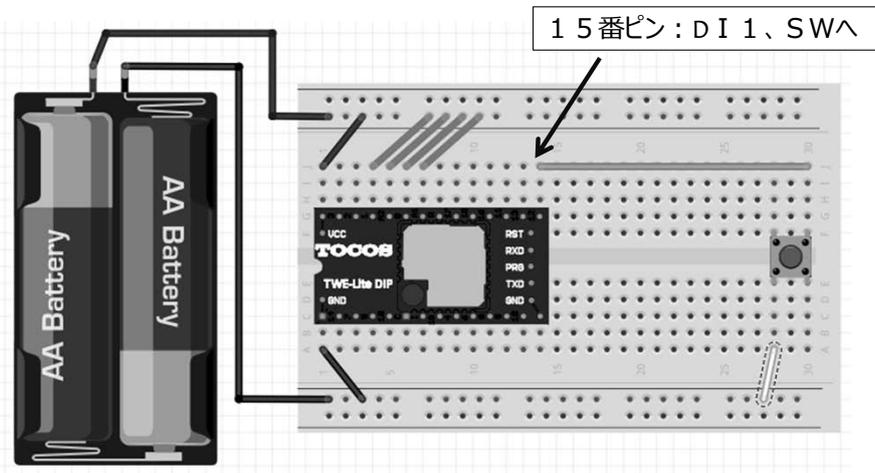
◇SWは基本的に2つの接点のON/OFFを行うものなので、配線数は2本です。

◇LEDは、ピン数が2本なので、配線数はやはり2本です。

◇その他に電源の+/-を接続するので、親機と子機合わせて4本。その他数本の配線を行えば、LEDの遠隔点灯制御が実現できるので、そのための配線は、10本程度です。

## 子機の例

### ◇子機の配線



48

◇前の図で示したSWを使う側の、配線の様子を図示しています。ご覧の様に10本の配線が必要ですが、図にしてみると大変シンプルですね。

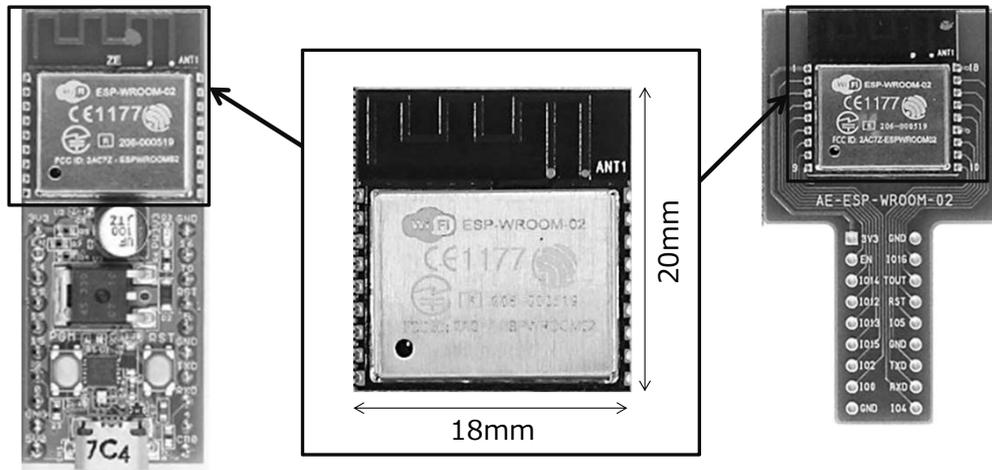
◇これらの回路を含めて演習編では、多くの回路を実際に作り無線通信のIoT応用を体験していただきます。

◇入門の初期で電子回路作成が未経験でも、回を重ねるうちに短時間で作成できるようになります。

◇演習編は【IoTトレーニングドリル】とさせていただければ良いでしょう。

## Step13 マイコンIoT入門

### ESP-WROOM-02



49

## Step13 マイコンIoT入門

◇Step13は、【WiFi機能を持つマイコン】を使ったIoT入門解説です。

◇図に示す写真は、いずれも ESP-WROOM-02 というマイコンを搭載したモジュールです。

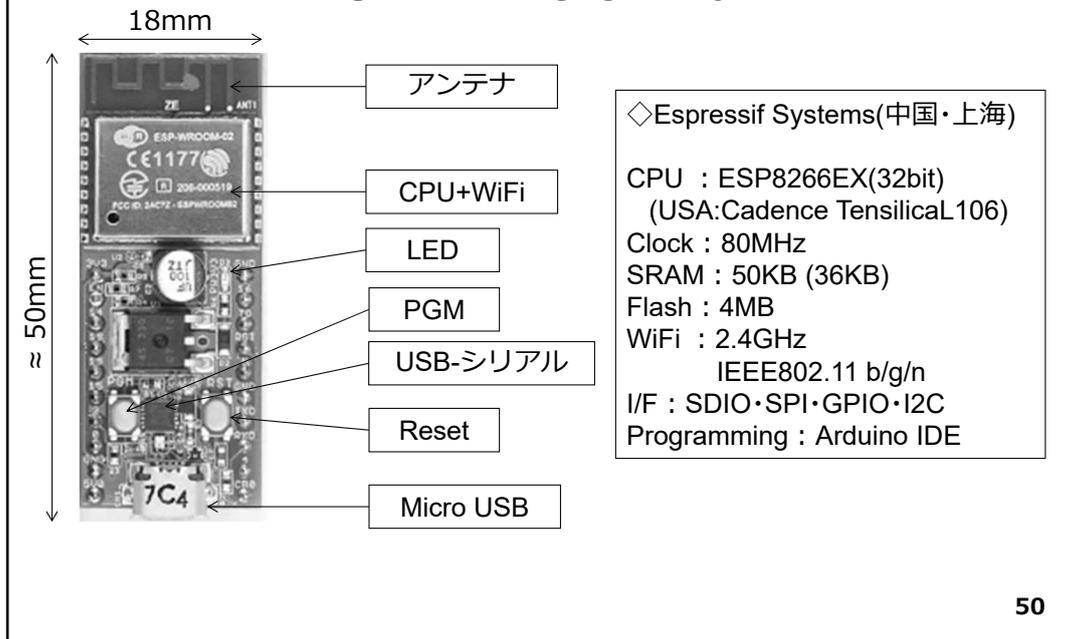
◇中央は、そのCPU基板部分です。このCPU部分が左右の基板上に配置されています（赤枠部分）。左のモジュールは、外部の回路が付属していて、USBケーブルで電源供給ができて、単独でプログラムの書込みやシリアル通信を行うことができるので、便利です。

◇このモジュールは、WiFi機能を内蔵しているので、単独でインターネット接続ができます。

◇以後、このモジュールを指してWiFiマイコンと呼ぶことがあります。

## WiFiマイコン

# ESP-WROOM-02



◇図に、すべての周辺回路が1枚の基板に収められたモジュールの特徴を示します。演習ではこのモジュールを使用する予定です。

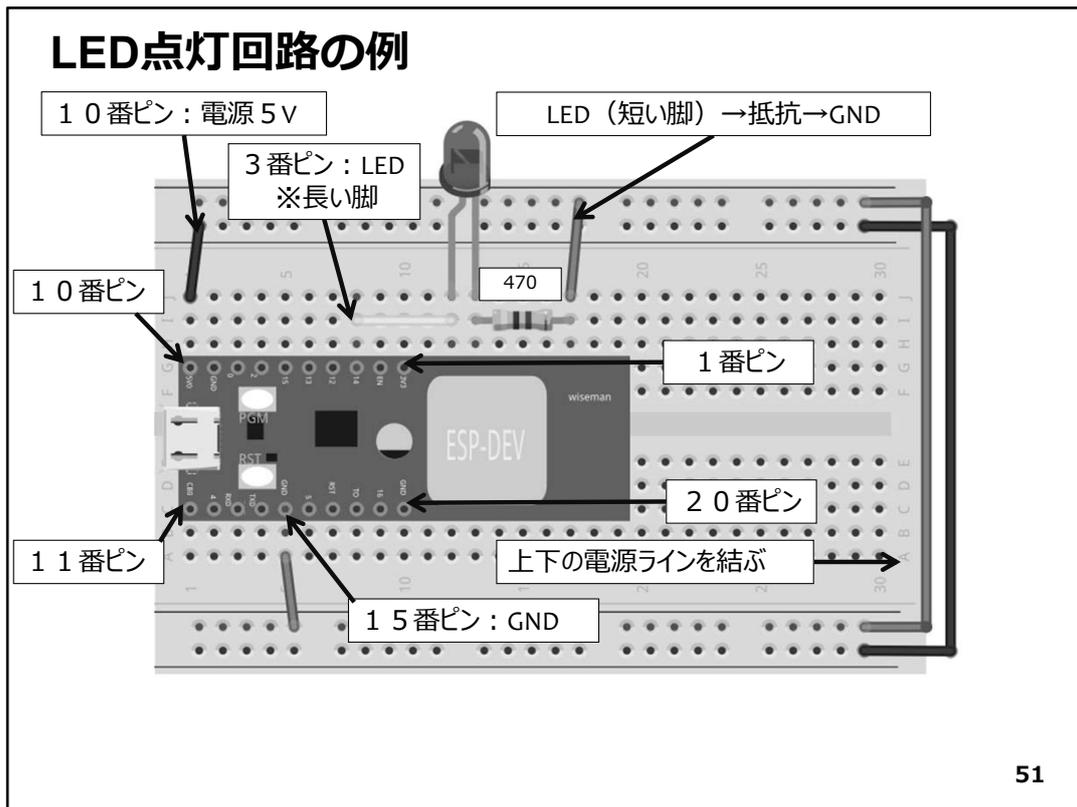
◇このWiFiマイコンは、Espressif Systems社(上海)が開発したもので、その最大の特徴はWiFi制御ブロックを内蔵し、単独でインターネット通信ができることです。 (WiFiは、2.4GHz IEEE802.11/b/g/n 対応となっています。)

◇従来のワンチップマイコン(1つのパッケージに、CPU,RAM,Flashメモリ、IO、I/Fなどを備えるマイコン)のクロック数と比べると数倍速くなっています。

★マイコンの処理速度基準の一つとしてクロック数が挙げられます。  
→ クロック数が大きければ、概ね処理速度も速い。

◇インターフェース: SDIO(SDカードなどに利用される)・SPI・GPIO・I2Cなど、IoTデバイスには、価値のあるインターフェースを備えています。

◇プログラム開発には、Arduino IDEが利用できます。



◇図にWiFiマイコン(ESP8266)を使用した、LED点灯の回路図を示します。無線マイコン(TWE-Lite)と比較してどうでしょうか。

◇配線数を数えると6本です。電源は、図中のWiFiマイコン右端にあるマイクロUSBコネクタから供給できます。プログラム開発時はPCとUSBケーブルで接続するので、電源はPCから供給されて、そのままUSB経由でPCとシリアル通信ができます。

◇PCと切り離して駆動する場合には、携帯電話などの充電用バッテリーからUSBケーブルでの給電や、USBを経由せずに基板上の電源ピンで、乾電池からの給電もできます。

◇この状態で、WiFiを利用した外部からの通知によるLEDのON/OFFが出来ます。  
★ただしそれには、プログラムを作る必要があります。

プログラム開発の環境は、次の図で説明します。

## Arduino IDE

### ◇Arduino 統合開発環境 IDE



52

◇Arduino IDEは、Arduino(アーデュイーノやアルディーノと発音するようです)マイコンボード向けの統合開発環境が利用できます。

このIDEは、インターネットに接続した環境であれば、ライブラリマネージャやボードマネージャによって、容易に各種ライブラリの導入や、ボードに対応したコンパイル環境が整備できます。高速プロトタイピングに向けて開発された環境で、Arduinoボードと呼ばれるマイコンが無期限保証で安価に提供されたことに加え、その仕様も公開された為、世界中で利用されています。このIDEは、対応しているマイコンの種類も豊富で、今も対応機種が増えています。

◇導入するのは容易です。【Arduino】で検索すると、<https://www.arduino.cc/> がヒットします。このWEBページから各種OS用のIDEをダウンロードして、ファイルをダブルクリックすればインストールができます。

★【Appendix G: Arduino IDEのインストール】にその手順を示しました。

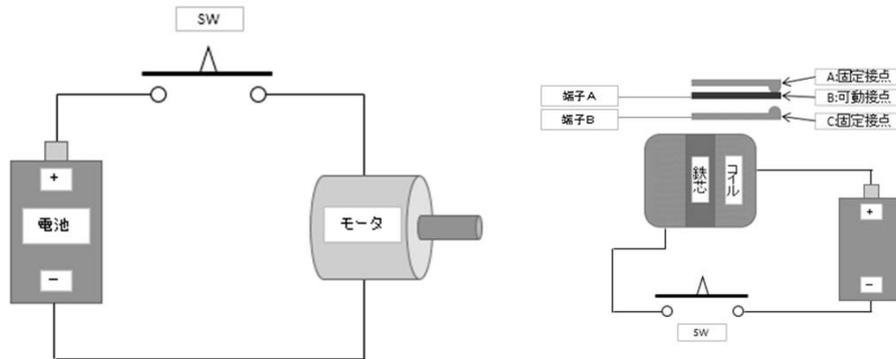
◇開発言語は C/C++をベースにしており、C言語のすべての構造と、いくつかのC++の機能をサポートしています。

◇IDEを起動すると開く、上図右のウインドウにソースコードを入力してプログラム開発を行います。

◇Arduino IDEの詳しい使い方は演習編で説明します。

## Step14 PLCによる制御とIoT

### SWと電磁石



53

## Step14 PLCによる制御とIoT

### 1. SWと電磁石

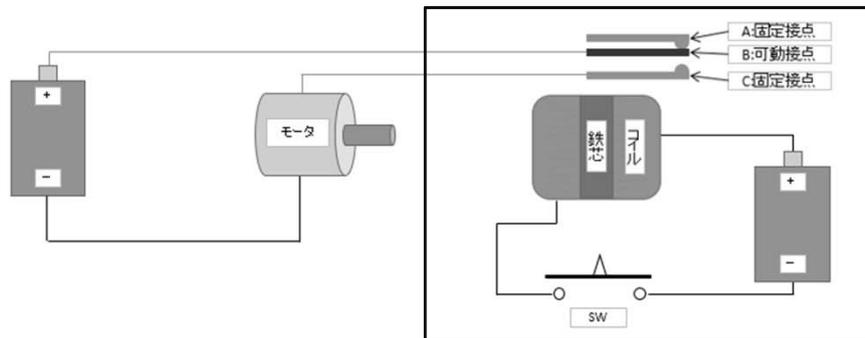
◇図左は、モーターに電池とSWを配線した図です。SWを押せばモーターは回転します。

◇SWは、電気の流れを止めたり(回路を開く)流したり(回路を閉じる)する役目があります。

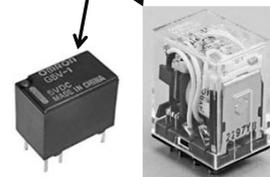
◇図右は、SWを押すと電気がコイルに流れて磁力が発生して【A:可動接点】を引き寄せます。すると、端子Aと端子Bが繋がります。これはSWと同じことで、コイルと鉄心でできた電磁石の働きによるものです。

この両方を組み合わせたものが次の図です。

## リレー回路



- ①. SWを押下すると、コイルに電気が流れます。
- ②. コイルと鉄心でできた電磁石に磁力が発生。
- ③. 【B:稼動接点】が引き寄せられる。
- ④. 【B:稼動接点】と【C:固定接点】が接触する。
- ⑤. モータに電気が流れる。
- ⑥. モータが回る。



54

## 2. リレー回路

◇この回路の動作は、

- ①. 図赤枠内下のSWを押下すると、コイルに電気が流れます。
- ②. コイルと鉄心でできた電磁石に磁力が発生。
- ③. 【B:稼動接点】が引き寄せられる。
- ④. 【B:稼動接点】と【C:固定接点】が接触する。
- ⑤. モータに電気が流れる。
- ⑥. モータが回る。

となって、SW押下 → モータ回転 となります。

★モータ電源とリレー電源が別々になっていることは重要です。

この図の赤枠内は、前の図のSWの代役を務めていることとなりますが、このようなことを行う産業機器としてリレー(右下写真)があります。

★写真左は基板直付け用、写真右は制御盤内で用いられるものです。

このようにリレーを用いた回路を【リレー回路】と呼びます。

## 3. 何故リレーを使うか

◇SWを使えば事足りるのに、なぜわざわざリレーを使い、ワンクッション置くような

ことをするのでしょか。

図のモータが、実は流れ作業用の長〜いベルトコンベアの駆動に使われていると考えましょう。コンベアの駆動に1個のモーターで力が足りない場合には複数のモーターを使いますが、モーターの数だけSWを使った回路を作っても、3個以上になると人が全て同時にSWを押すことが難しくなります。

◇他に、複数のモーターを1個のSWにつなぐという方法も思い浮かびますが、モーターの数が多くなると、1個のSWに沢山の電流が流れるので、安全性の面から問題です。

◇そのような場合にリレーを用いて、リレーの接点でSWの代わりを行えば、接点で流せる電流の最大値(最大定格)までは、モーターを増やせます。しかしこれも最大値まで使うのは、回路の信頼性の面で問題があります。

その解決策は次の様にする事です。

◇実は、ほとんどのリレーは同時に作動する複数の接点を持っていて、1個のリレーで2回路以上のSWの代わりを務めてくれます。リレーが作動する(ONすると言うこともある)と複数の回路の接点が同時に接続されるので、SW 1個の操作を複数のSWの操作に拡張してくれます。数が足りないとき、リレーの接点で他のリレーをONすることにすれば、【走るリレーのボタンタッチの様に(※1)】次々とリレーをONすることができて便利なので、工場などで数多く使われています。

(※1)このように順序良く事が進むので、リレー回路を順序回路や【シーケンス回路】と呼ぶこともあります。

◇リレーの稼働接点で構成する電気回路は、【リレーの電源とは切り離されている】ので、低い電源電圧のリレーで高い電圧のリレーをON/OFFしたり、直流電源のリレーで交流をON/OFFすることもできます。

◇リレーの接点はSWと同じなので、SWでON/OFF出来る物なら何でも制御できるのです。

#### 4. SWとリレーによる動きのある設備の制御

◇SWを押したらコンベアが動き始めましたが、SWから手を放すとコンベアは止まってしまう。そのままでは、ずっとSWを押し続けていなくてははいけません。このような場合もリレーの接点を使い、ONしているリレーの接点で自らのコイルに電流を流すような回路(自己保持回路)を構成すれば、一度だけSWを押すことでコンベアが動き続けるようにできます。

SW ON → リレー ON → コンベア稼働

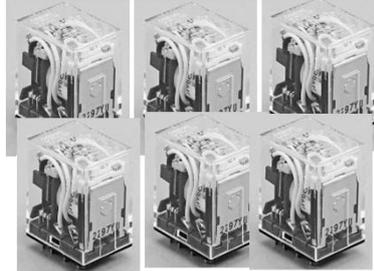
◇では、どうやって止めましょうか？

問題 : SWを押すとコンベアが動き出し、その上に載っている物を運びます。この

一定の場所まで対象物を運んでコンベアを停止したいときには、どうすれば良いか考えてみて下さい。

## 複雑な制御とPLC

複雑化→SW・リレーがどんどん増える！！



◎ ◎ ◎ **そこでPLCの登場！！** ◎ ◎ ◎



55

### 5. PLCの登場

◇条件が複数あるような複雑な制御を行うとSWやリレーなどがどんどん増えてしまい、これらを配線して納める制御盤の中も混雑してきます。そのような状態では、少しの追加・改善を行う場合でも、正確な配線が出来なかったり、リレーや配線が制御盤内に収まらないなどの問題が出てきます。制御盤を新たに作り直し回路も組み直し・・・などと言う事になると、掛かる時間も費用も工数も増加してしまいます。

◇そこで、そのような複雑な制御回路をコンパクトにまとめられる機器としてPLC (Programable Logic Controller) が使われています。

◇PLCの中身は実はマイコンです。SWやモータなど入力対象と出力対象は、そのままPLCに配線し、リレー回路の設計図と同じ回路図(ラダー図)をパソコンで描いてPLCに転送すれば、PLC内部でシーケンス回路がシミュレーションされて動作してくれます。変更したいときは、パソコンで回路図を変更すればOKです。制御対象の入力と出力の間にある中間的なリレーが、PLC内部のマイコンのメモリで実現されるので、制御盤内のリレーの数が激減します。回路図はパソコンのファイルとして保存されるので、管理が容易です。USBメモリやSDカードなどで現場のPLCに転送することも可能になっています。配線作業も軽減されるので、工場などでは多くのPLCが使われています。

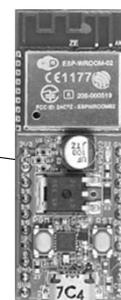
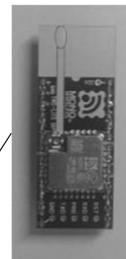
◇IoTデバイスは、この様に既に生産現場に多く存在しているPLCのシステムとも協調しながら、IoTシステムを構築する必要もあるのです。その場合PLCで稼動し

ている回路図のどの接点から情報を取得するか検討する必要も出てきます。リレー回路の基礎を理解していると大変役立ちます。

## Step15 WEB連携IoTの開発

### ◇WEBサービスと連携するIoTモデル

- ①. 無線マイコン、WiFiマイコン、PLC  
メッセージ交換WEBサービス等を利用
- ②. IoTシステムモデルの設計
- ③. 必要なものを明らかにする
- ④. 演習のための準備



56

## Step15 WEB連携 IoTの開発

◇このStepでは、IoTデバイスとPLCなどを利用したWEBサービス連携IoTシステムのモデルを説明します。それぞれのモデルシステムを構成する機器や環境の組み合わせについて解説し、各モデルで何が必要かを明らかにしておきます。演習編で実際に開発実習するための準備Stepと考えてください。

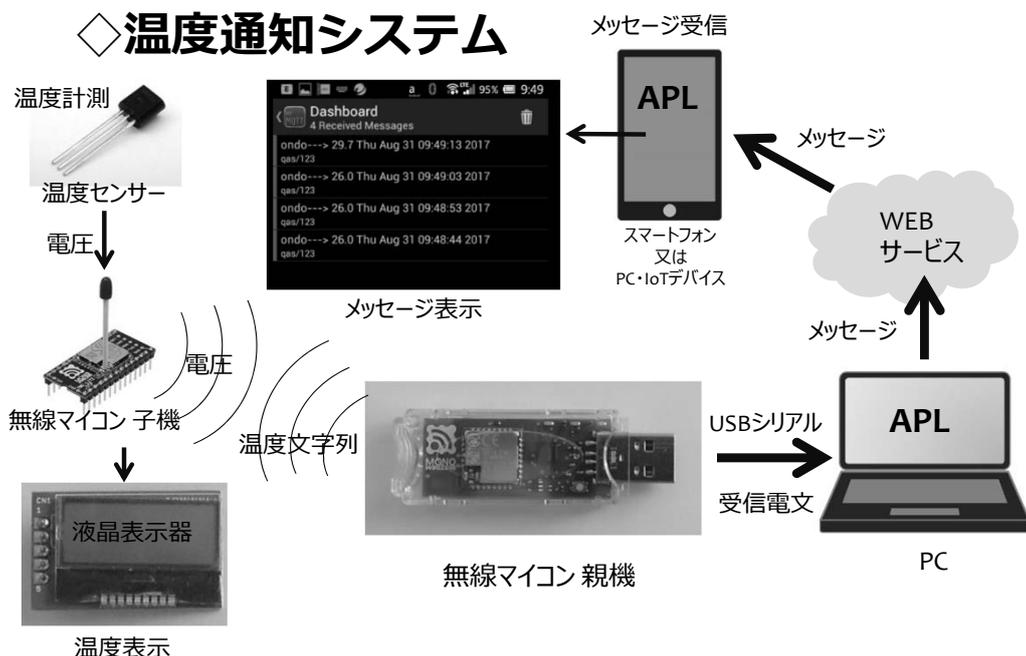
◇開発実習では、ここで説明するようなモデルと同様のものをいくつか開発します。

### ◇IoTシステムモデル

1. 無線マイコン(TWE-Lite)を利用した、センサーデータのWEB連携IoTシステム
2. 無線マイコン(TWE-Lite)を利用した、生産実績収集IoTシステム
3. WiFiマイコン(ESP8266)を利用した、センサーデータのWEB連携IoTシステム
4. WiFiマイコン(ESP8266)を利用した、生産実績収集IoTシステム
5. PLC+無線マイコンを利用した、センサーデータのWEB連携IoTシステム
6. PLC+無線マイコンを利用した、生産実績収集IoTシステム

## センサーデータのWEB連携（無線マイコン）

### ◇温度通知システム



57

### 1. 無線マイコン(TWE-Lite)を利用した、センサーデータのWEB連携IoTシステム

◇図は、無線マイコン(TWE-Lite)を用いた温度通知を行うシステムです。このシステムは、次の様に稼動します。

- ①. 温度センサーの出力を電圧として無線マイコン子機(図左中)が計測します。
- ②. 子機は、電圧を無線マイコン親機(図下中)に無線で送信します。(固定長文字列電文方式)
- ③. 親機は、受信した電文をそのままシリアル通信(USB経由)でPCに送信します。
- ④. PC内部のアプリケーションが電文を受け取り、解析して電圧を温度に変換します。  
★以下⑦までがPC内での処理です。
- ⑤. 温度は、液晶表示用とWEBサービスに通知するメッセージ用に文字列編集を行います。
- ⑥. 液晶表示用文字列をシリアル通信で親機に送ると、親機はそれを無線通信で子機に送ります。
- ⑦. 同様にメッセージ用文字列をWEBサービスに送ります。
- ⑧. スマートフォン(または、PCや他のIoTデバイス)内部のアプリケーションがWEBサービスからメッセージを受信します。
- ⑨. 受信したメッセージを表示します。

以後、①～⑨を繰り返します。

◇このシステムは、PC内のプログラムとスマートフォン(またはPCや他のIoTデバイス)のプログラム(図でPCとスマートフォンのAPL)を作成する必要があります。

★スマートフォンやPCのアプリケーションは、フリーソフトとして公開されているものを利用することが可能です。

★無線マイコン内部のアプリケーションプログラムは、出荷時に書き込まれている標準アプリケーションをそのまま利用します。

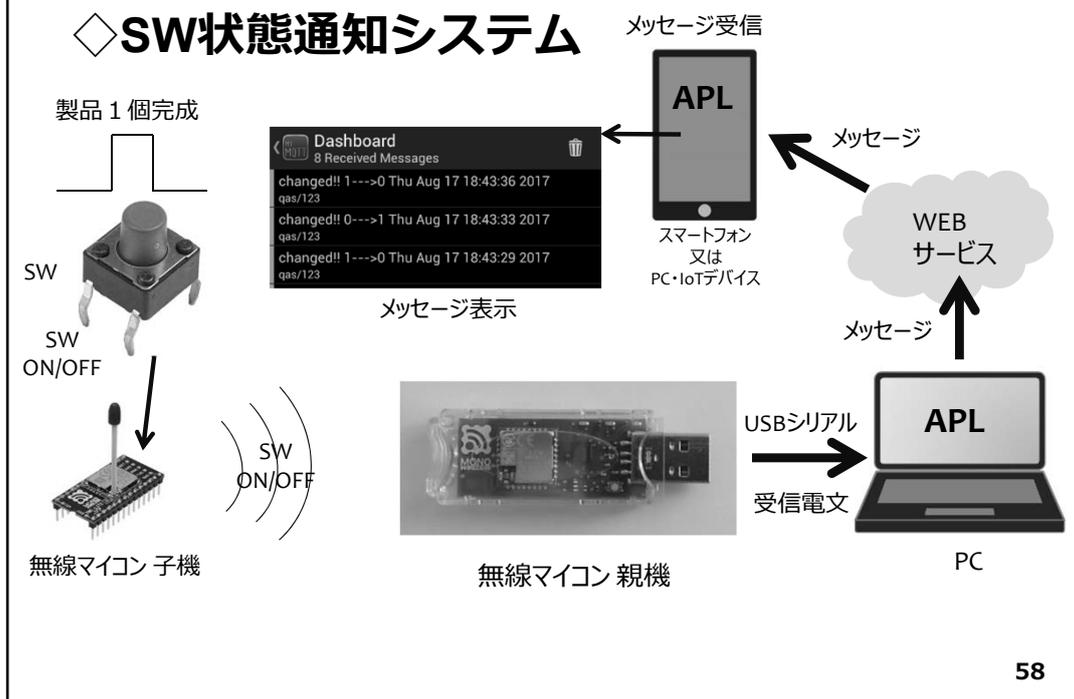
◇使用する機器などは次の通りです。

1. 温度センサー
2. 液晶表示器
3. 無線マイコン(子機・親機)
4. PC(内部にアプリケーションプログラム)
5. WEB上メッセージ交換サービス
6. スマートフォンまたはPCや他のIoTデバイス。  
(内部にアプリケーションプログラム)
7. Internet接続環境(PC側、スマートフォン側)

◇PCで温度変換したデータを蓄積すれば、ローカル環境でデータベースを構築できます。また、スマートフォン(メッセージを受け取る)側でデータを蓄積すれば、広く共用できるデータベースを構築できます。このデータベースを分析して、その結果をWEBサービスを通じて送信すれば、整理された情報の広範囲での共有が実現します。

## 生産実績収集IoTシステム

### ◇SW状態通知システム



## 2. 無線マイコン(TWE-Lite)を利用した、生産実績収集IoTシステム

◇図は、無線マイコン(TWE-Lite)を用いた生産実績収集を行うシステムです。

★図左のSWは生産設備などに取り付けられていて、製品が1個排出されるごとに、状態はOFF→ON→OFFと変化します。

このシステムは、次の様に稼働します。

- ①. 無線マイコン子機がSWの状態変化(ON/OFF)を検出します。
- ②. 子機は、変化後の状態(ONかOFFか)を無線マイコン親機に無線で送信します。(固定長文字列電文方式)
- ③. 親機は、受信した電文をそのままシリアル通信(USB経由)でPCに送信します。
- ④. PC内部のアプリケーションが電文を受け取り、解析してSWの状態変化を確認します。  
★以下⑦までがPC内での処理です。
- ⑤. SWの状態がOFF→ON→OFFと変化したとき、製品が1個完成したとPCアプリケーションが認識します。
- ⑥. 製品が1個完成したことを通知する、生産実績通知メッセージを作製します。
- ⑦. 生産実績通知メッセージをWEBサービスに送ります。
- ⑧. スマートフォン(または、PCや他のIoTデバイス)内部のアプリケーションがWEBサービスからメッセージを受信します。
- ⑨. 受信したメッセージを表示します。

以後、①～⑨を繰り返します。

◇このシステムは、PC内のプログラムとスマートフォン(またはPCや他のIoTデバイス)のプログラム(図でPCとスマートフォンのAPL)を作成する必要があります。

★スマートフォンやPCのアプリケーションは、フリーソフトとして公開されているものを利用することが可能です。

★無線マイコン内部のアプリケーションプログラムは、出荷時に書き込まれている標準アプリケーションをそのまま利用します。

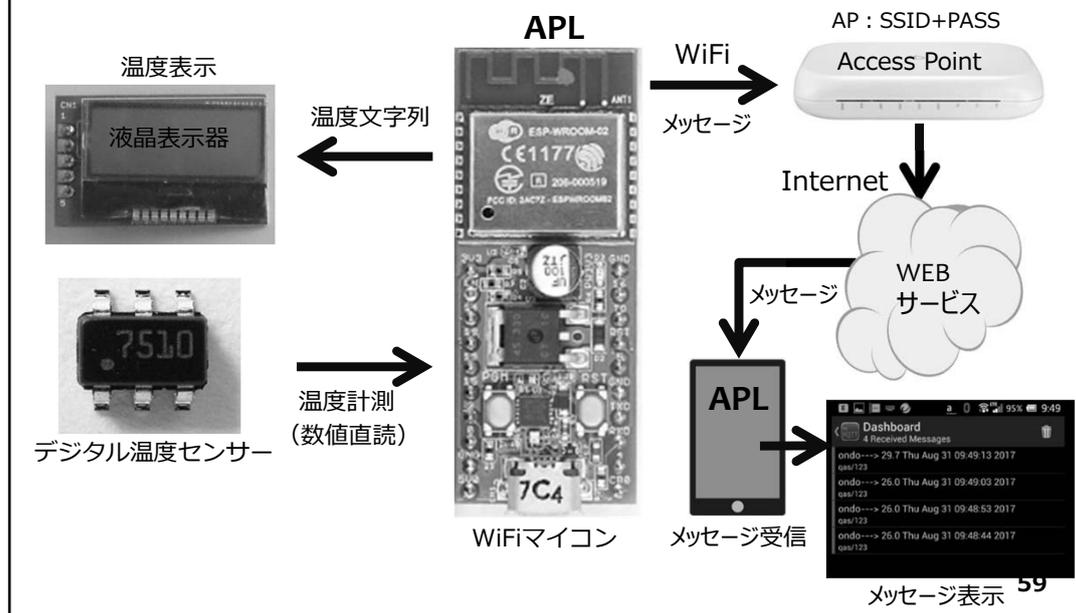
◇使用する機器などは次の通りです。

1. SW(その他の実績を検出できるセンサー)
2. 無線マイコン(子機・親機)
3. PC(内部にアプリケーションプログラム)
4. WEB上メッセージ交換サービス
5. スマートフォン(またはPCや他のIoTデバイス。内部にアプリケーションプログラム)
6. Internet接続環境(PC側、スマートフォン側)

◇PCで製品が1個完成したと認識した時にデータを蓄積すれば、ローカル環境で生産実績データベースを構築できます。また、スマートフォン(メッセージを受け取る)側でデータを蓄積すれば、広く共用できる生産実績データベースを構築できます。このデータベースを分析して、その結果をWEBサービスを通じて送信すれば、整理された生産状況の広範囲での共有が実現します。

## センサーデータのWEB連携IoTシステム

### ◇温度通知システム



### 3. WiFiマイコン(ESP8266)を利用した、センサーデータのWEB連携IoTシステム

◇図は、WiFiマイコン(ESP8266)を用いた温度通知システムです。

このシステムは、次の様に稼動します。

- ①. WiFiマイコン(内部のアプリケーション)がデジタル温度センサーの出力を温度直読値として読取ります。  
★以下④までがWiFiマイコン内部の処理です。
- ②. 温度を液晶表示用とWEBサービスに通知するメッセージ用に文字列編集を行います。
- ③. 液晶表示用文字列を表示します。
- ④. 同様にメッセージ用文字列をWEBサービスに送ります。
- ⑤. スマートフォン(または、PCや他のIoTデバイス)内部のアプリケーションがWEBサービスからメッセージを受信します。
- ⑥. 受信したメッセージを表示します。

以後、①～⑥を繰り返します。

◇このシステムは、WiFiマイコン内のプログラムとスマートフォン(またはPCや他のIoTデバイス)のプログラムを作成する必要があります。

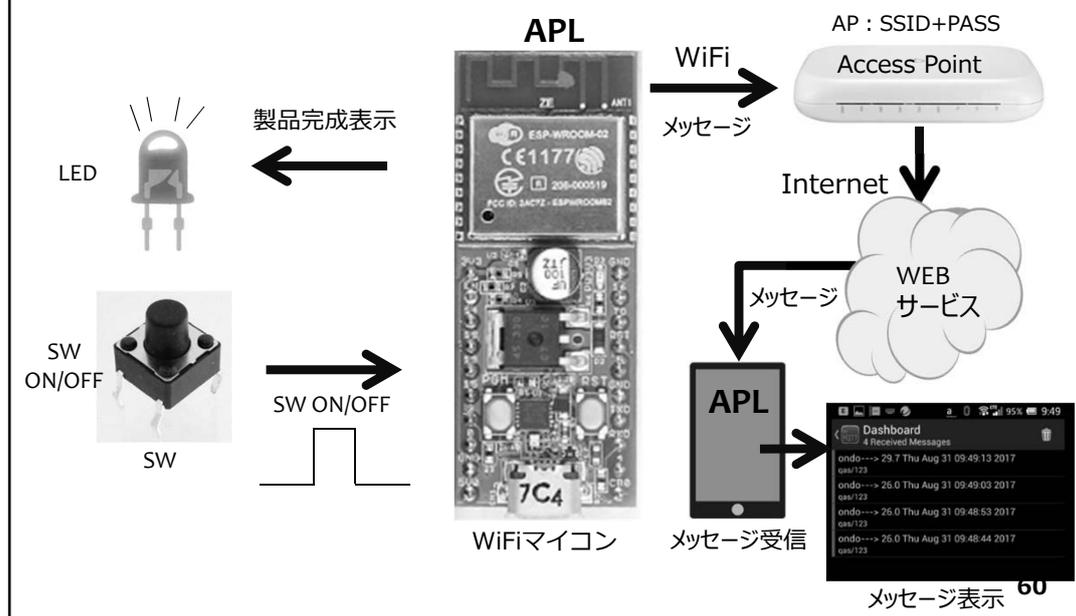
◇使用する機器などは次の通りです。

1. 温度センサー(デジタル)
2. 液晶表示器
3. WiFiマイコン
4. WiFi Access Point
5. WEB上メッセージ交換サービス
6. スマートフォン(またはPCや他のIoTデバイス。内部にアプリケーションプログラム)
7. Internet接続環境(WiFiマイコン側、スマートフォン側)

◇このシステムでは、WiFiマイコン側(IoTデバイス側)でのデータ蓄積ができませんが、スマートフォン(メッセージを受け取る)側でデータを蓄積すれば、広く共用できるデータベースが構築できます。このデータベースを分析して、その結果をWEBサービスを通じて送信すれば、整理された情報の広範囲での共有が実現します。

# 生産実績収集IoTシステム

## ◇SW状態通知システム



### 4. WiFiマイコン(ESP8266)を利用した、生産実績収集IoTシステム

◇図は、WiFiマイコン(ESP8266)を用いた生産実績収集を行うシステムです。

★図左のSWの生産設備などに取り付けられていて、製品が1個排出されるごとに、状態はOFF→ON→OFFと変化します。

このシステムは、次の様に稼働します。

- ①. 無線マイコン子機がSWの状態変化(ON/OFF)を検出します。
- ②. 子機は、変化後の状態(ONかOFFか)を無線マイコン親機に無線で送信します。(固定長文字列電文方式)
- ③. 親機は、受信した電文をそのままシリアル通信(USB経由)でPCに送信します。
- ④. PC内部のアプリケーションが電文を受け取り、解析してSWの状態変化を確認します。  
★以下⑦までがPC内での処理です。
- ⑤. SWの状態がOFF→ON→OFFと変化したとき、製品が1個完成したとPCアプリケーションが認識します。その時、LEDを1回点滅します。  
★液晶表示器を追加してローカルにカウントした生産実績を表示することもできます。
- ⑥. 製品が1個完成したことを通知する、生産実績通知メッセージを作製します。
- ⑦. 生産実績通知メッセージをWEBサービスに送ります。

- ⑧. スマートフォン(または、PCや他のIoTデバイス)内部のアプリケーションがWEBサービスからメッセージを受信します。
- ⑨. 受信したメッセージを表示します。

以後、①～⑨を繰り返します。

◇このシステムは、WiFiマイコンのプログラムとスマートフォン(またはPCや他のIoTデバイス)のプログラムを作成する必要があります。

★スマートフォンやPCのアプリケーションは、フリーソフトとして公開されているものを利用することが可能です。

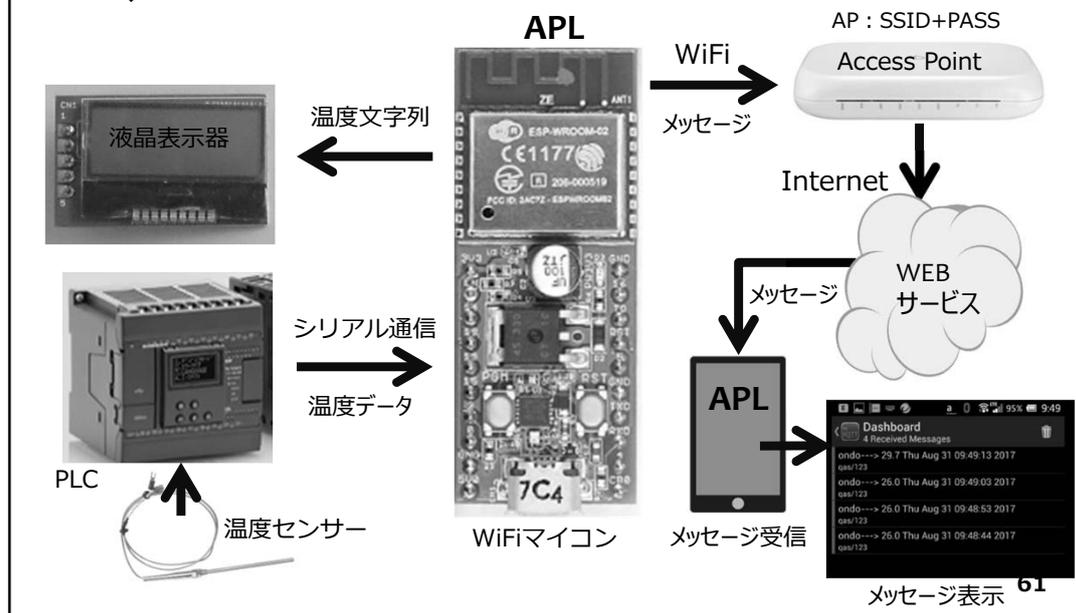
◇使用する機器などは次の通りです。

1. SW(その他の実績を検出できるセンサー)
2. WiFiマイコン(子機・親機)
3. WEB上メッセージ交換サービス
4. スマートフォン(またはPCや他のIoTデバイス。内部にアプリケーションプログラム)
5. Internet接続環境(PC側、スマートフォン側)

◇このシステムでは、WiFiマイコン側(IoTデバイス側)でのデータ蓄積ができませんが、スマートフォン(メッセージを受け取る)側でデータを蓄積すれば、広く共用できる生産実績データベースを構築できます。このデータベースを分析して、その結果をWEBサービスを通じて送信すれば、整理された生産状況の広範囲での共有が実現します。

# センサーデータWEB連携IoTシステム (PLC+WiFiマイコン)

## ◇温度通知システム



### 5. PLC+WiFiマイコンを利用した、センサーデータのWEB連携IoTシステム

◇図は、PLC+WiFiマイコン(ESP8266)を用いた温度通知システムです。

このシステムは、次の様に稼働します。

- ①. PLCが温度センサーの出力を読み取ります。この際、センサーはアナログですが、PLC内部で温度換算して計測値をPLC内部メモリに保存します。
- ②. PLCは、温度データをWiFiマイコンにシリアル通信で送信します。  
★この際、温度データは、数値(バイナリ)、文字列どちらでもPLC内部で変換が可能です。
- ③. WiFiマイコン(内部のアプリケーション)が温度を受信します。  
★以下⑥までがWiFiマイコン内部の処理です。
- ④. 温度を液晶表示用とWEBサービスに通知するメッセージ用に文字列編集を行います。
- ⑤. 液晶表示用文字列を表示します。
- ⑥. 同様にメッセージ用文字列をWEBサービスに送ります。
- ⑦. スマートフォン(または、PCや他のIoTデバイス)内部のアプリケーションがWEBサービスからメッセージを受信します。
- ⑧. 受信したメッセージを表示します。

以後、①～⑧を繰り返します。

◇このシステムは、PLCおよびWiFiマイコン内のプログラムとスマートフォン(また

はPCや他のIoTデバイス)のプログラムを作成する必要があります。

◇使用する機器などは次の通りです。

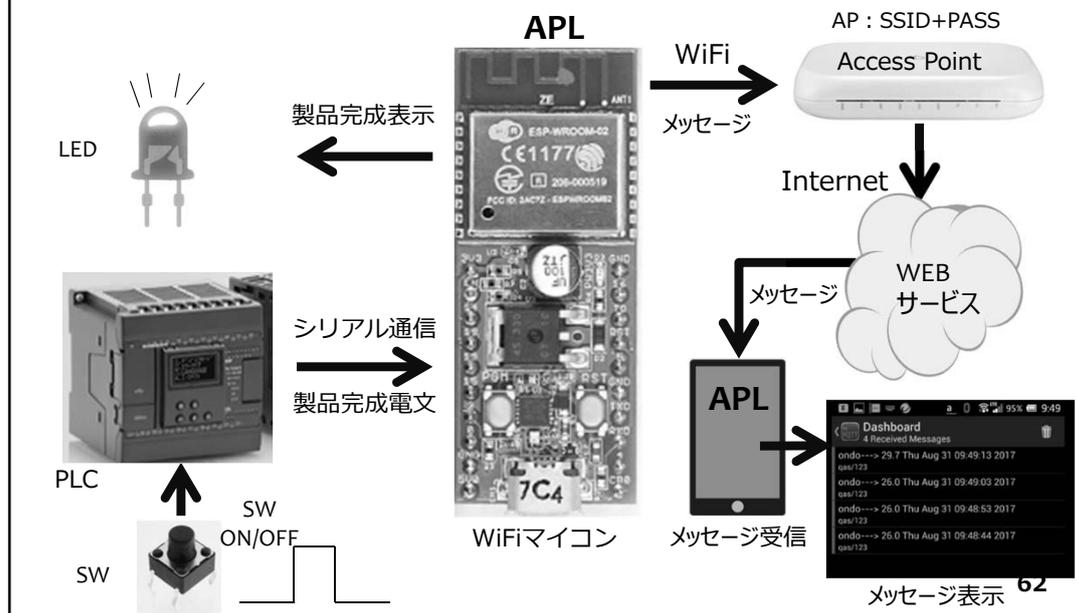
1. 温度センサー
2. PLC(内部にアプリケーションプログラム)
2. 液晶表示器
3. WiFiマイコン(内部にアプリケーションプログラム)
4. WiFi Access Point
5. WEB上メッセージ交換サービス
6. スマートフォン(またはPCや他のIoTデバイス。内部にアプリケーションプログラム)
7. Internet接続環境(WiFiマイコン側、スマートフォン側)

◇このシステムでは、WiFiマイコン側(IoTデバイス側)でのデータ蓄積ができませんが、スマートフォン(メッセージを受け取る)側でデータを蓄積すれば、広く共用できるデータベースが構築できます。このデータベースを分析して、その結果をWEBサービスを通じて送信すれば、整理された情報の広範囲での共有が実現します。

◇PLC内部のアプリケーションの機能で、大きなサイズの外部メモリにデータを蓄積することも可能です。

## 生産実績収集IoTシステム

### ◇SW状態通知システム



## 6. PLC+WiFiマイコン(ESP8266)を利用した、生産実績収集IoTシステム

◇図はPLC+WiFiマイコン(ESP8266)を用いた生産実績収集を行うシステムです。

- ★図左のSWは生産設備などに取り付けられていて、製品が1個排出されるごとに、状態はOFF→ON→OFFと変化します。
- ★図では、末端の製品完成信号をSWから取得するように作図していますが、実際には、ほとんどのPLC設備制御システムでは、PLC内部のアプリケーションプログラムで、製品完成を認識できます。

このシステムは、次の様に稼働します。

- ①. PLCがSWの状態変化(OFF→ON→OFF)を検出し、製品が1個完成したことを認識します。
- ②. PLCは、製品が1個完成したことをシリアル通信で、WiFiマイコンに通知します。
- ③. WiFiマイコンは、PLCから製品1個完成電文を受け取ったことをLED点滅で知らせます。
  - ★液晶表示器を追加してローカルにカウントした生産実績を表示することもできます。
  - ★以下⑤までがWiFiマイコン内での処理です。
- ④. 製品が1個完成したことを通知する、生産実績通知メッセージを作製します。
- ⑤. 生産実績通知メッセージをWEBサービスに送ります。
- ⑥. スマートフォン(または、PCや他のIoTデバイス)内部のアプリケーションがWEBサービスからメッセージを受信します。

⑦. 受信したメッセージを表示します。

以後、①～⑦を繰り返します。

◇このシステムは、PLC、WiFiマイコン、スマートフォン(またはPCや他のIoTデバイス)のプログラムを作成する必要があります。

★スマートフォンやPCのアプリケーションは、フリーソフトとして公開されているものを利用することが可能です。

◇使用する機器などは次の通りです。

1. SW(その他の実績を検出できるセンサー)
2. PLC(内部にアプリケーションプログラム)
3. WiFiマイコン(内部にアプリケーションプログラム)
4. WEB上メッセージ交換サービス
5. スマートフォン(またはPCや他のIoTデバイス。内部にアプリケーションプログラム)
6. Internet接続環境(PC側、スマートフォン側)

◇このシステムでは、WiFiマイコン側(IoTデバイス側)でのデータ蓄積ができませんが、スマートフォン(メッセージを受け取る)側でデータを蓄積すれば、広く共用できる生産実績データベースを構築できます。このデータベースを分析して、その結果をWEBサービスを通じて送信すれば、整理された生産状況の広範囲での共有が実現します。

◇PLC内部のアプリケーションの機能で、大きなサイズの外部メモリにデータを蓄積することも可能です。

☆これで【基礎概論編】は終了です。

お疲れさまでした！！

**終わり**

# Appendix 【基礎概論】

# Appendix

---

**A:Excel VBA によるプログラミング**

**B:シリアル通信(RS-232c)**

**C:生産技術者マネジメントスキル**

**認定試験サンプル**

**D:SW の情報を検出する2つの方法**

**E:TWE-Lite Data Sheet(抜粋)**

**F:ESP-8266 Data Sheet(抜粋)**

**G:Arduino IDE のインストール**

# A:Excel VBA によるプログラミング

表計算ソフト（Microsoft Excel）は、広く利用されていて、ほとんどの Windows PC にインストールされています。Excel が持つ機能の一つで、データの取り扱いを自動化できる VBA の使い方について説明します。VBA とは Visual Basic for Applications を指しています。

Visual Basic という言語は、PC システム開発では広く利用されている言語で、その機能をオフィスソフト（Applications、ここでは Excel）に特化させたものが VBA です。VBA だけで相当に使えるアプリケーションが開発できるので、今後折に触れて利用していただきたいと考えています。時間を割いても決して無駄にはなりません。Text File と表（Excel）によるデータの取扱いは、VBA で同じように実現できるので、ここでまとめて解説することにします。

## 1 準備



図 1

図は Excel を開いたときのリボン（貼り付け、ハサミのアイコンなどが並んでいる横に広がった帯のこと）の様子です。（Excel のバージョンは Excel 2013）赤枠で示すところに【開発】というタブがあり、

これを利用して VBA で Excel を活用したシステム開発を行うことができます。【開発タブ】はデフォルトでは表示されないので、次の様にして表示させます。

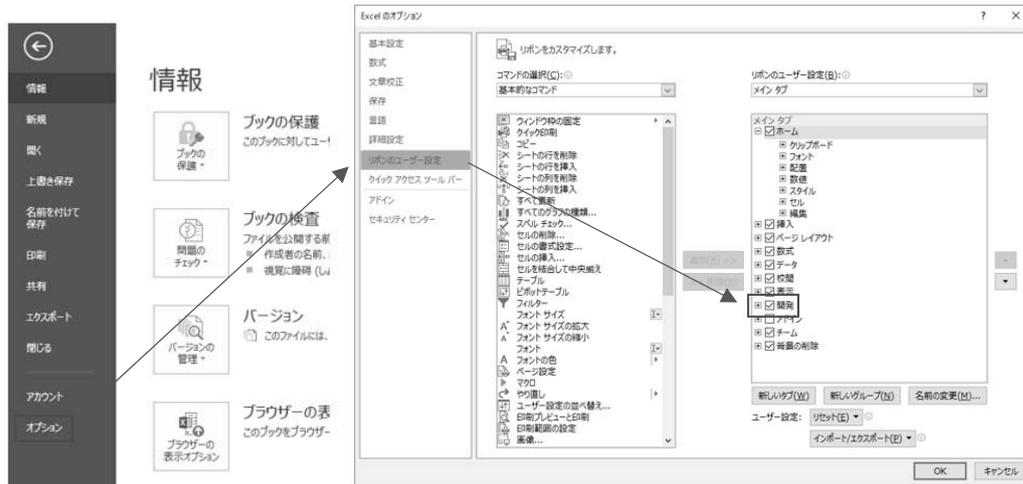


図 2

【ファイル】タブをクリックし、【オプション】→【リボンのユーザー設定】と辿ります。図 2 に示す Excel のオプションウィンドウが開くので、ウィンドウ右側にある【開発】と書かれているチェックボックスにチェックを入れて OK ボタンをクリックすると【開発】タブが表示されます。

## 2 ボタンの作成

開発タブを選択するとリボンが図 3 の様に変化します。



図 3

図 3 の左側にある、Visual Basic と書かれたアイコンをクリックすれば、VBA プログラムの作成・編集などができますが、それは後で使うことにして、ここでは少し簡易な方法で進めます。図 3 中央にある挿入と書かれたアイコンをクリックすると【フォームコントロール】というツールボックスが表示されます。(図 4)



図 4

その左上にある長方形のマーク（ボタン）をクリックして、マウスカーソルをワークシート上に移動すると、カーソルの形が+に変化します。ワークシート上に適当な大きさの長方形を描くと【マクロの登録】ウインドウが開く（図 5）ので【新規作成】をクリックします。

【Microsoft Visual Basic for Applications】ウインドウ（以後 VBA ウインドウと呼ぶ）が開き、そこには既にボタン 1\_Click() という名前の VBA プログラム（マクロと言うこともある）のひな形が作られています。(図 6)

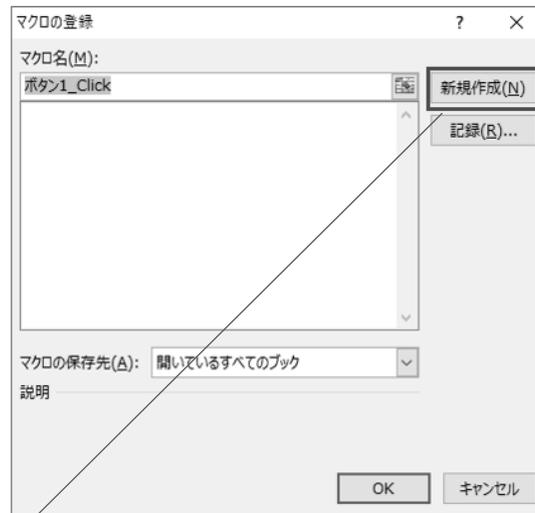


図 5

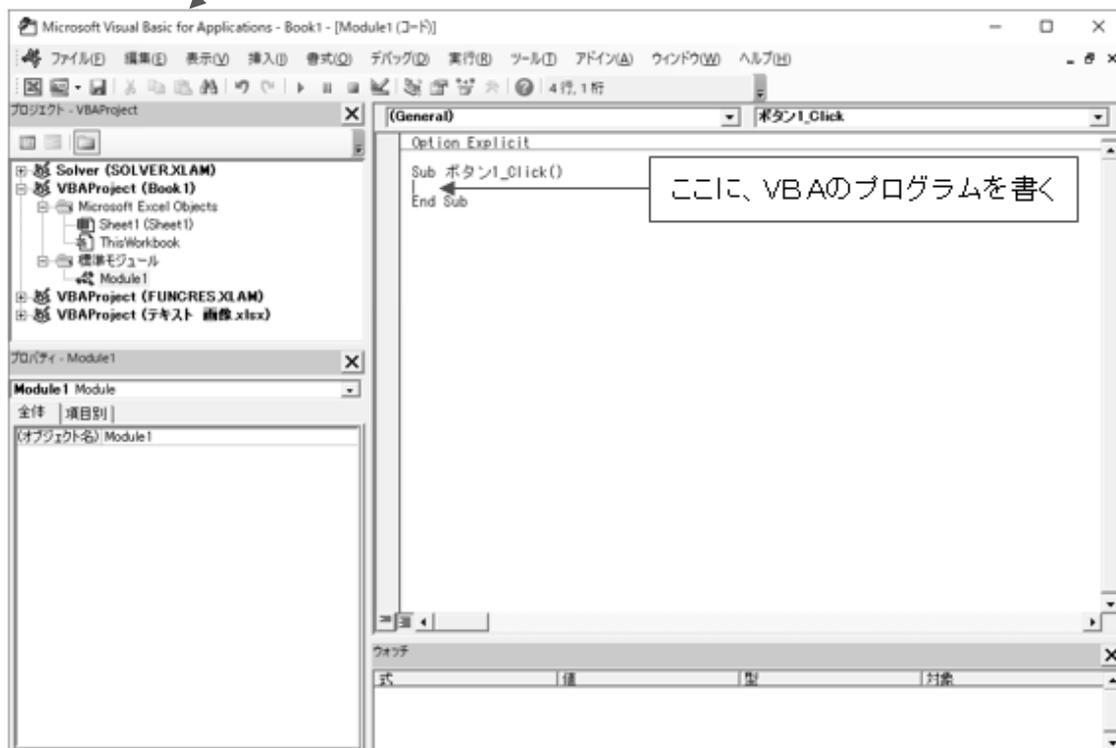


図 6

### 3 VBA プログラムの作成

図 6 で示すように、【Sub ボタン 1\_Click()】と【End Sub】の間の空白行先頭にカーソルがあるので、そこから VBA プログラムを書き始めます。

手始めに、リスト 1 の様にプログラム（ソース、ソースコードとも言います）を書いてみましょう。プログラム中のアポストロフィー【'】の右側は、コメントを意味します。[]の前の空白は、TAB キーを用いて入力すると各行の出だしが揃い、見やすい（読みやすい）プログラムになります。

```
Sub ボタン 1_Click()  
    [a1] = 1 'A1 のセルに 1 を格納  
End Sub
```

#### リスト 1

ソースコードを入力したら、VBA ウィンドウはそのままにしておき、ワークシートのウィンドウを選択します。そこに今作ったボタン 1 があります。【マウスカーソルをボタンの上に移動する】と【カーソルが指の形】に変化します。ボタン 1 をクリックすると、【A1セル】に【1】が入力されます。プログラム上でセルを指定する場合は、括弧[]の間にセルアドレスを書いて指定します。セルアドレスとは、セルの列と行を並べて書いた文字列のことです。

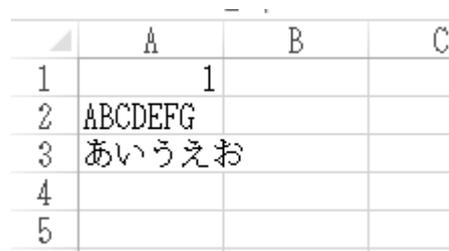
ポイント：セルアドレスは、[A1]、[B7]の様に括弧[]で挟んで指定できます。他の指定方法もありますが、それは、後で説明します。

次に、VBA ウィンドウに切り替えてプログラムを次の様に変更します。

```
Sub ボタン 1_Click()  
    [a1] = 1 'A1 のセルに 1 を格納  
    [a2] = "ABCDEFGG"  
    [a3] = "あいうえお"  
End Sub
```

### リスト 2

先ほどと同様に、ボタン 1 をクリックするとワークシートには、指定の数値や文字列が入力されます（図 8）。文字列はダブルクォーテーション【””】で挟んで表現します。



	A	B	C
1	1		
2	ABCDEFGG		
3	あいうえお		
4			
5			

図 7

```
Sub ボタン 1_Click()  
    [a1] = 1 'A1 のセルに 1 を格納  
    [a2] = "ABCDEFGG"  
    [a3] = "あいうえお"  
    [a4] = [a1] + 999  
End Sub
```

### リスト 3

リスト 3 の様に記述すれば計算もできます。VBA には複雑な計算を行うための関数が豊富に準備されているので、それらを活用して統計・分析などを行うことができます。ここでは、難しい計算は行いませんが、よく利用するいくつかの関数を使ってみます。

ポイント：計算式は、数学の計算式の様に記述します。C 言語や Java や Python 等とほとんど同じです。（全く同じかもしれない。）

## 4 MsgBox 関数

```
Sub ボタン 1_Click()  
    [a1] = 1  
    [a2] = "ABCDEFGFG"  
    [a3] = "あいうえお"  
    [a4] = [a1] + 999  
    MsgBox [a1]  
End Sub
```

リスト 4

リスト 4 の様にソースコードを変更して、ボタンをクリックすると、

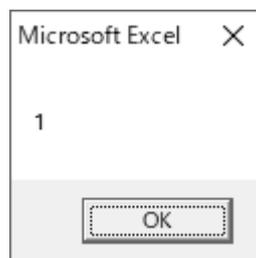


図 8

メッセージボックスが表示されます（図 8）。[A1]セルの内容【1】がメッセージとして表示されています。右上の×または OK ボタンをクリックしてメッセージボックスを閉じます。※メッセージボックスを表示したままでは VBA プログラムの編集ができないので、必ず OK ボタン、または×でメッセージボックスを閉じてください。

メッセージボックスは、もう少し気の利いた表示ができます。

```
Sub ボタン 1_Click()  
    [a1] = 1  
    [a2] = "ABCDEFGFG"  
    [a3] = "あいうえお"  
    [a4] = [a1] + 999  
    MsgBox "[A1]は、[" & [a1] & "]", vbOKOnly, "IoT 基礎"  
End Sub
```

#### リスト 5

【&】は、文字列の連結です。キーワード【vbOKOnly】は、OK ボタンだけの表示を指示しています。その後の"IoT 基礎"というのは、メッセージボックスのタイトル指定です。これらの関数に与える引数は、カンマ【,】で区切って指定します。リスト 5 を記述してボタンをクリックすると、次のようなメッセージボックスが表示されます。

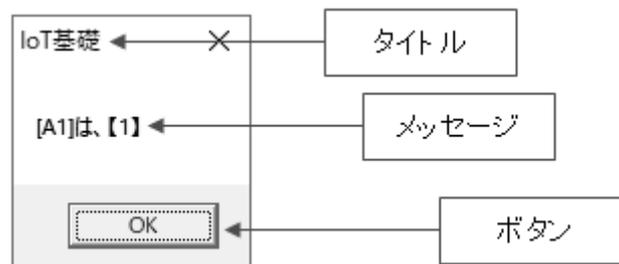


図 9

## 5 InputBox () 関数

```
Sub ボタン 1_Click()  
    [A1] = InputBox("数値は？", "計測値入力", 9.9)  
End Sub
```

リスト 6

リスト 6 のプログラムを入力して実行（ボタンをクリックしてプログラムを動かす）すると図 10 のような画面表示があり、ソースコードで指定したデフォルト値が初めから入力されています。そのまま OK ボタンをクリック（または Enter キー押下）すると、A1 セルに数値が入ります。数値を入力して OK ボタンをクリックすれば、入力した数値が入ります。



図 10

ソースコードに書いた【9.9】はデフォルトの入力値です。この数値の代わりに、文字列を入力して OK しても、入力ができます。文字列か数値かの判断が必要であれば、VBA のソースコードをそのように作成します。入力された情報が、文字か数字かを判断する関数 `IsNumeric()` や `VarType()` を利用します。それらの使い方は、Web を検索すると容易に分かるでしょう。

もう気付いたかもしれませんが、`MsgBox` 関数には括弧【()】を記述し

ませんが、InputBox()関数には、括弧を記述しています。

ポイント：戻り値のある関数には、括弧【()】を付ける。関数から処理の結果戻される値を戻り値と言います。

MsgBox 関数は、システムが操作者へ通知を行うのに使用できます。  
InputBox 関数は、操作者とのやり取りができるマン・マシンインタフェース関数です。

## 6 名前を付けて保存

この辺りでそろそろ、現状の Excel ファイル (Book) を保存しておきましょう。

既にご存知の方も多いと思いますが、Excel ファイルには【xlsx】という拡張子が付きます。これは、通常の Excel ファイルであり、VBA プログラムを追加したファイルは、特別な形式で保存しなければいけません。

【ファイル】→【名前を付けて保存】と辿ると、ファイル名の入力が必要ですが、その下にある【ファイルの種類】プルダウンで、【Excel マクロ有効ブック (\*.xlsm)】という形式を選んでください。

ファイル名(N):	Book1.xlsx
ファイルの種類(T):	Excel ブック (*.xlsx)
作成者:	Excel ブック (*.xlsx)
	Excel マクロ有効ブック (*.xlsm)
	Excel バイナリブック (*.xlsb)
	Excel 97-2003 ブック (*.xls)
	XML データ (*.xml)
フォルダーの非表示	単一ファイル Web ページ (*.mht;*.mhtml)
	Web ページ (*.htm;*.html)
	Excel テンプレート (*.xltx)
リジョン	Excel マクロ有効テンプレート (*.xltm)
	Excel 97-2003 テンプレート (*.xlt)
	テキスト (タブ区切り) (*.txt)
	Unicode テキスト (*.txt)
	XML スプレッドシート 2003 (*.xml)
	Microsoft Excel 5.0/95 ブック (*.xls)
	CSV (カンマ区切り) (*.csv)
	テキスト (スペース区切り) (*.prn)

図 11

ポイント：VBA プログラム付きの Excel ファイルは、マクロ有効ブック (\*.xlsm) 形式で保存する。

## 7 保存したマクロ有効ブックファイルを開き、VBA を編集する

一度保存した VBA ソース有りの Book を開いて、ソースコードの編集を行うには、ワークシートに配置した【ボタンを右クリック】して【マクロの登録】を選択すると、【マクロの登録】ウインドウが表示されます。

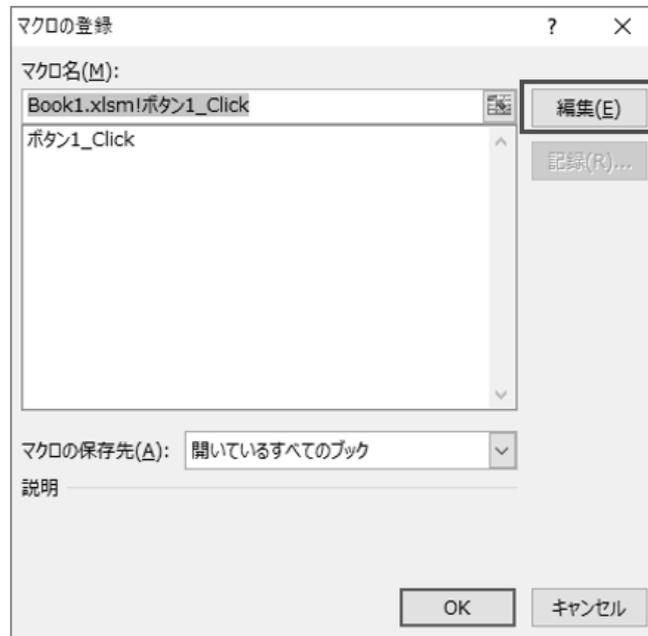


図 12

そこで【編集】ボタンをクリックします。ボタンを複数配置したときは、編集したいボタンを右クリックで選択して【編集】ボタンをクリックすると VBA ウィンドウが開きます。

## 8 もう一つのセルへのアクセス方法（重要）

これまでは、[A1]などと記述してセルにアクセスしていましたが、大量のデータを取り扱う際の必須の記述方法が他にあります。

```
Sub ボタン 1_Click()  
    Worksheets("Sheet1").Cells(1, 1) = 1  
End Sub
```

### リスト 7

リスト 7 の様にソースコードを入力して実行すると、A1 セルに、数値 1 が入力されます。

```
Worksheets("Sheet1").Cells(1, 1) = 1
```

- ・ Sheet1 はワークシート
- ・ Cells の ( ) 内の最初の 1 は、行を示す ( 1 = 1 行)
- ・ Cells の ( ) 内の最後の 1 は、列を示す ( 1 = A 列)

※ワークシート名の指定もできるので、この記述方法でセルを指定すれば、複数のワークシートのセルにアクセスできます。

※**Worksheets** の左にさらに、**Book** 名を指定することもできますが、ここでは省略します。その方法も **Web** を検索すれば、容易に見つかるでしょう。

```
Sub ボタン 1_Click()  
    Dim row As Integer    'row という名前の整数型の変数  
    Dim col As Integer    'col という名前の整数型の変数  
    row = 2              '行 2 行目  
    col = 3              '列 3 列目 = C 列  
    Worksheets("Sheet1").Cells(row, col) = 1  
End Sub
```

### リスト 8

リスト 8 を入力して実行すると、C2 セルに数値 1 が入力されます。  
行と列を変数で指定して、そのセルにデータを格納するプログラムです。

**ポイント：**変数で行と列が指定できる。変数の内容を変化させれば、連続したセルのデータ処理が可能となり、大量データ処理向けのプログラムが作れます。

プログラム各行の説明は、コメントを参照して下さい。

以下、必要最小限の VBA コードのステートメント（一つ一つの手続きや命令、宣言などのこと）を説明します。

## 9 繰り返し処理 (For～Next)

```
Sub ボタン 1_Click()  
    Dim row As Integer 'row という名の行カウント用変数  
    Dim col As Integer 'col という名の列用変数  
    row = 1 '行 1 行目  
    col = 3 '列 3 列目 = C 列  
    For row = 1 To 10 'row を 1 から 10 まで繰り返す  
        Worksheets("Sheet1").Cells(row, col) = row 'row 格納  
    Next row 'row を +1 して繰り返す  
End Sub
```

リスト 9

リスト 9 のプログラムを入力して、実行すると、C1 から C10 のセルに、1～10 の数値が入ります。

	A	B	C	D	E	F
1			1			
2			2			
3			3			
4			4			
5			5			
6			6			
7			7			
8			8			
9			9			
10			10			
11						



図 13

では、次の様にプログラムを追加すると、どうでしょう。

```
Sub ボタン 1_Click()  
  
    Dim row As Integer 'row という名の行カウント用変数  
  
    Dim col As Integer 'col という名の列用変数  
  
    row = 1 '行 1 行目  
  
    col = 3 '列 3 列目 = C 列  
  
    For row = 1 To 10 'row を 1 から 10 まで繰り返す  
  
        Worksheets("Sheet1").Cells(row, col) = row  
  
        Worksheets("Sheet1").Cells(row, col - 1) = Chr(Asc("A") + row)  
  
    Next row  
  
End Sub
```

#### リスト 10

これを実行すると、図 14 のようになります。

	A	B	C	D	E	F
1		B		1		
2		C		2		
3		D		3		
4		E		4		
5		F		5		
6		G		6		
7		H		7		
8		I		8		
9		J		9		
10		K		10		
11						



図 14

```
Worksheets("Sheet1").Cells(row, col - 1) = Chr(Asc("A") + row)
```

は、文字【A】の ASCII コード（16 進で 41）に row の内容を加算して、文字に変換したものを B 列のセルに格納しています。

ASCII コードでは、A~Z の文字の値が 1 ずつ増加しているので、文字 A の ASCII コードに+1、+2、・・・と加算すると、その値が表す文字コードは B から順に C,D,E,F、・・・と進むことになるのです。もし、初めの文字が数字の 0（ゼロ）なら、B 列には 1、2、3、・・・となりますが、10 番目はどうなるのかな？

問題：上記の様に、初めの文字を数字の 0（ゼロ）にして、10 番目の文字がどうなるか確かめて、その結果を説明しなさい。

## 10 条件分岐

```
Sub ボタン 1_Click()  
    Dim row As Integer  
    Dim col As Integer  
    row = 1      '行 1 行目  
    col = 3     '列 3 列目 = C 列  
    Do  
        If Worksheets("Sheet1").Cells(row, col) = "" Then  
            Exit Do  
        End If  
        row = row + 1  
    Loop  
    MsgBox "データ数 = [" & (row - 1) & "]", vbOKOnly + vbInformation  
End Sub
```

リスト 11

前の結果をそのままにして、リスト 11 を入力・実行すると、図 15 のメッセージボックスが表示されます。データ数は 10 です。



図 15

赤枠で囲んだ部分で、row と col で指定のセルが空であるか調べています。その結果、セルが空であれば Exit Do で Do~Loop の無限繰り返しから脱出しています。脱出に失敗すると、無限ループになってプログラムが止まりません。そのときは、Ctrl+Break を押下します。

※Break キーが無い PC もありますが、その場合どのキーを押せばよいかは、事前に調べておくと安心です。筆者のノート PC (DELL) では、Fn + Ctrl+B でした。

**問題：** MsgBox に与えている引数の中で **[(row - 1)]** の部分は、何をしているのか説明しなさい。

## 1 1 テキストファイル出力

さて、ここでやっとテキストファイルへの出力を行います。前（リスト 11）の結果をそのままにして、次のリストを入力・実行してください。

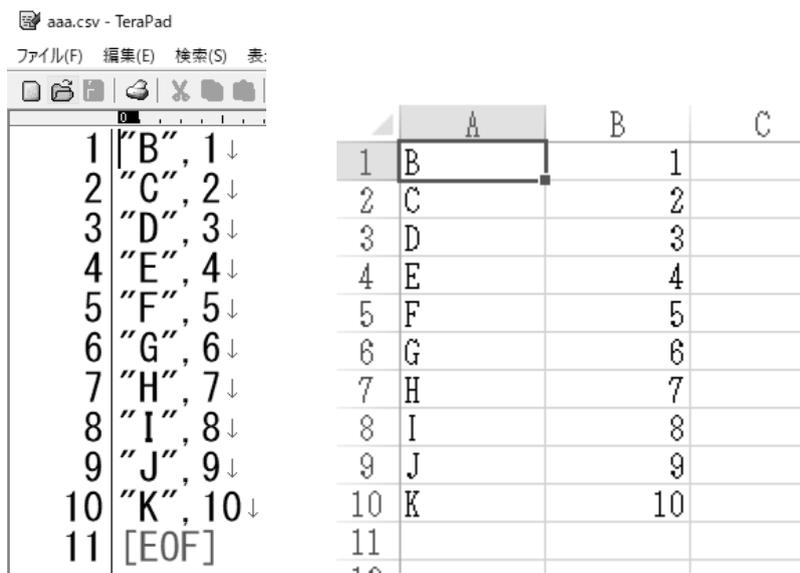
```
Sub ボタン 1_Click()  
  
    Dim row As Integer  
  
    Dim col As Integer  
  
    Dim CellB As String    'B列のセルのデータを入れる変数（文字列）  
    Dim CellC As Integer   'C列のセルのデータを入れる変数（整数）  
  
    row = 1                '行 1行目  
  
    col = 3                '列 3列目 = C列  
  
    '1番でファイルを開く  
  
    Open "C:¥aaa.csv" For Output As #1  
  
    For row = 1 To 10  
  
        CellC = Worksheets("Sheet1").Cells(row, col)  
  
        CellB = Worksheets("Sheet1").Cells(row, col - 1)  
  
        Write #1, CellB, CellC        '1行ファイルに書き出す  
  
    Next row  
  
    Close #1                'ファイルを閉じる  
  
End Sub
```

### リスト 12

```
Open "C:¥aaa.txt" For Output As #1
```

で指定している場所に、aaa.txt という名前のファイルが出力されます。

他の場所に出力したい場合、C:¥ とファイル名の間パスを追加します（例：C:¥Users¥ken¥Documents¥aaa.txt）。出来上がったファイルをメモ帳やテキストエディタで開くと図 16 左のようにセルのデータが記録されています。同じファイルを Excel で開くと、図 16 右のように表示されます。



	A	B	C
1	B		1
2	C		2
3	D		3
4	E		4
5	F		5
6	G		6
7	H		7
8	I		8
9	J		9
10	K		10
11			

図 16

これは、CSV（カンマ区切りファイル）としてファイルを格納したからで、CSV ファイルは、Excel で開き、自由に編集ができます。

このように、特に VBA に頼らなくても CSV ファイルは Excel で読み込めますが、次はあえて VBA でファイルをワークシートに読み込んでみます。

## 12 テキストファイル読込

```
Sub ボタン 1_Click()  
    Dim row As Integer  
    Dim col As Integer  
    Dim CellB As String  
    Dim CellC As Integer  
    row = 1      '行 1 行目  
    col = 3      '列 3 列目 = C 列  
    '1 番でファイルを開く  
    Open "C:¥Users¥ken¥Documents¥aaa.csv" For Input As #1  
    Do Until EOF(1)      'ファイルの最後(※)まで繰り返す  
        Input #1, CellB, CellC  '1 行読込、変数に格納  
        '変数からセルに格納  
        Worksheets("Sheet1").Cells(row, col - 1) = CellB  
        Worksheets("Sheet1").Cells(row, col) = CellC  
        row = row + 1  '行番号を+1  
    Loop  '繰り返す  
    Close #1      'ファイルを閉じる  
End Sub
```

### リスト 13

まず、ワークシートを綺麗に空にしてリスト 13 を入力・実行すると、ファイルのデータが読み込まれて、図 17 のようにワークシートにデー

タが格納されます。図 16 では、A 列、B 列にデータが入っていましたが、今回の結果は B 列と C 列にデータが格納されています。それは次の様にセルの列を指定したからです。【col ,col - 1】の部分。

```
Worksheets("Sheet1").Cells(row, col - 1) = CellB
```

```
Worksheets("Sheet1").Cells(row, col) = CellC
```

	A	B	C	D	E	F
1		B	1			
2		C	2			
3		D	3			
4		E	4			
5		F	5			
6		G	6			
7		H	7			
8		I	8			
9		J	9			
10		K	10			
11						



図 17

このように、Excel VBA を使えば、テキストファイルと Excel ワークシートを自由に利用できて、文字・文字列・数字・数値などを大量に扱うことができます。格納するファイル名もサンプルでは、固定ファイル名を使いましたが、変数に入れたり InputBox() 等で指定したり、ファイル指定ダイアログ (Application.FileDialog(msoFileDialogOpen)) を利用すれば容易に指定できます。これらの使用方法も Web で容易に見つかるでしょう。

ここまで VBA で Text File と Excel ワークシートを使う方法を解説しましたが、元になるデータは、プログラムで発生させたり入力したりし

たものでした。IoT システムでは、数値などのデータを手入力するのではなく、IoT デバイス (IoT システム) から何らかの方法で受取る必要があります。この外部からの【受取】方法は、別の部分で解説しますが、要は `InputBox()` のような関数が、IoT システムと通信などを行って、データを受取り、戻り値として VBA プログラムに通知すると考えておけば良いのです。

# B:シリアル通信(RS-232c)

---

## 1. シリアル通信とは

パソコン（マイコン）と周辺機器、あるいはパソコン（マイコン）とパソコン（マイコン）間などで通信を行う技術は様々に存在していますが、最も多用されている通信技術はシリアル通信です。

コンピュータの世界では、扱うデータは全て2進数に変換して処理をします。通信を行うためには、この2進数のデータをどのようにして送信・受信するのでしょうか。

その方法は、大きくシリアル通信とパラレル通信に分けられます。

シリアル通信は1本の信号線を使用して1ビットずつ順番にデータを伝送する方法のことを言います。



図 18 シリアル通信

シリアル通信は1本の信号線を使用して1ビットずつ順番にデータを伝送する方法です。それに対して、パラレル通信は複数の信号線を使用して複数ビットを一度に伝送する方法のことです。

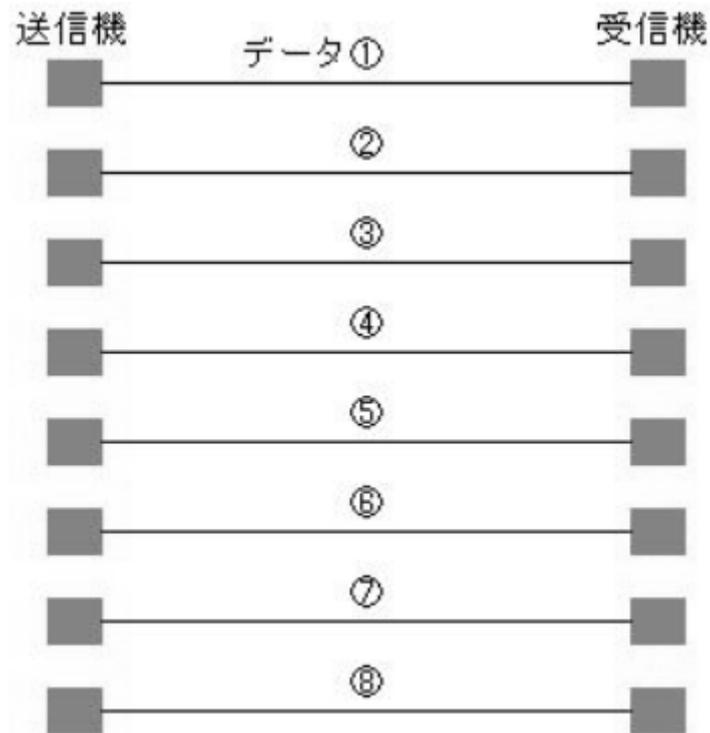


図 19 パラレル通信

パラレル通信を実現するためには何本もの信号線が必要になり、距離が離れれば離れるほど、コストが膨らんでしまいます。また、ケーブルも太くなってしまうので、きれいにまとめることが難しくなります。通信異常がケーブルに起因して発生したような場合を想定すると、シリアル通信の方が保守に関わるコストも抑えることができるのは、容易に想像できます。

パラレル通信の場合は、一度に複数ビットを伝送するということが送信タイミングのずれを誘発させる可能性があり、通信の精度についても品質が低下してしまいます。

その点、シリアル通信であれば、信号線が一本で済むためコストも場所も取らず、通信タイミングのずれも発生しません。

## 2. RS-232c とは

シリアル通信は、様々な用途に使われるようになりましたが、その際、仕様の互換性を持たせるために、規格統一が必要となり、アメリカの EIA (Electronic Industries Association) で 1969 年にシリアルインターフェースの規格が制定されました。その規格が RS-232c です。

RS-232c では、パソコンなどの DTE (データ端末装置) と、モデムなどの DCE (データ回線終端装置) とを接続してデータ通信を行うための電氣的・機械的な特性が定義されています。(DTE、DCE という用語は RS-232c のなかで定義されたインターフェース用語です。)

## 3. RS-232c の信号レベル

RS-232c はいくつかの信号線で使用しますが、その中に送信線、受信線、接地線が存在します。

コンピュータで扱うデータは全て 2 進数に変換して処理されますが、この 2 進数のデータを電圧 (信号レベル) で表します。信号線が  $-3V \sim -25V$  のときが "1" であり、 $+3V \sim +25V$  のときが "0" と取り決められまし

た。また、データを伝送していないときは”1”の状態になるように定められています。

状態	Low	High
電圧範囲	-3V~-25V	+3~+25V
値	”1”	”0”
名称	マーク	スペース

図 20 RS-232c の信号レベル

一見、データ伝送していないときは”0”の状態にする方が正解のように思えるかもしれませんが。これはシリアルインターフェースのルーツがモールス信号にあることに由来します。モールス信号では、データ伝送していないとき電流が流れていなければ、何らかの故障か信号線の切断としていました。その名残が RS-232c 規格に残っているのです。

#### 4. シリアル通信のイメージ

シリアル通信のイメージは、糸電話にたとえることができます。一方は喋る人、もう一方は聴く人です。お互いに話したいときは2本の糸電話が必要です。ただし、同時に使用することはできません。

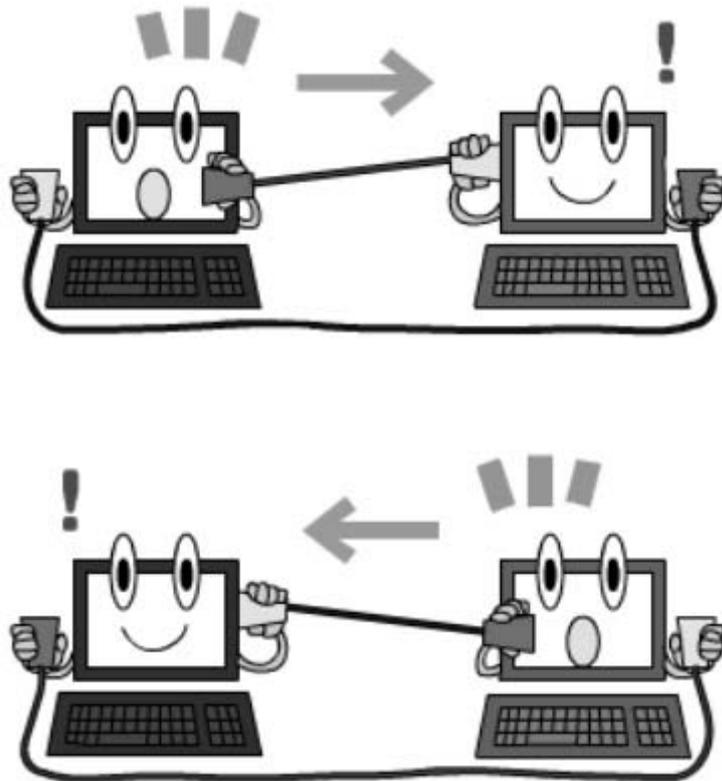


図 21 シリアル通信のイメージ（出典：CodeZine）

一方から「お元気ですか」と送ると、返事として「元気ですよ」と返ってきます。一本の糸電話で「お元気ですか」の文字列が送られます。この通信方法がシリアル通信です。パソコンの場合、文字は1文字（1バイト）ずつ送ります。

相手に受け取り準備をしてもらうため、「送りますよ」「1文字目」「終わりました」「送りますよ」「2文字目」「終わりました」…と1文字ごと、送り初めと終わりを知らせます。

送る文字の内容は送る側と受け取る側で独自に決めることができます。

こちらから送りたい相手への線は 1 本、相手からこちらへの線 1 本、それに接地線の 3 本で通信できるということになります。

この 3 本線が一つのケーブルに入ればシリアルケーブルの出来上がりですが、実際のシリアルケーブルには、他の信号線も含まれているので、もう少し信号線の数が多くなっています。

## 5. 信号の種類とコネクタ

RS-232c は、使用するコネクタや信号配置も規定され標準化されています。図に D-SUB9 ピンの信号配置と信号線の種類を示します。

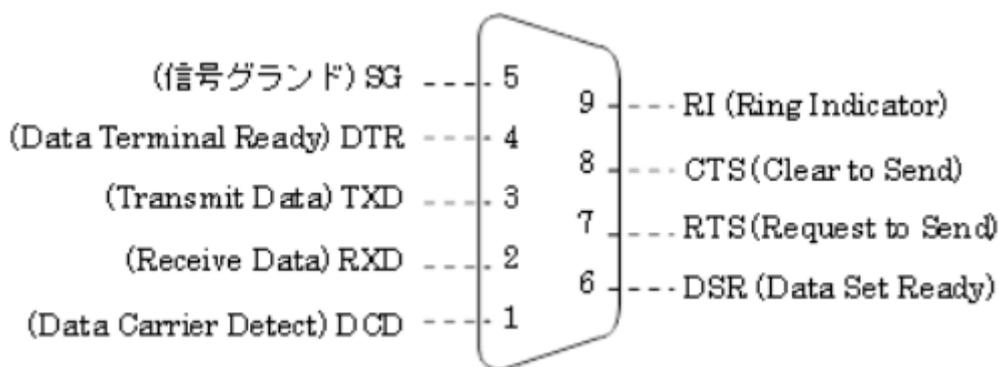


図 22 信号線の配置 (D-SUB9 コネクタ)

ピン番号	信号名	説明	
1	DCD	Date Carrier Detect	キャリア検出
2	RxD	Recieved Data	受信データ
3	TxD	Transmitted Data	送信データ
4	DTR	Data Terminal Ready	データ端末レディ
5	SG	Signal Ground	信号用接地または共通帰線
6	DSR	Data Set Ready	データセットレディ
7	RTS	Request To Send	送信要求
8	CTS	Clear To Send	送信許可
9	RI	Ring Indicator	被呼表示
CASE	FG	Frame Ground	保守用接地またはアース

図 23 信号の種類（出典：CONTEC）

上の図は、D-SUB9 ピンコネクタの信号線ですが、D-SUB25 ピンコネクタも存在します。現在では、パソコンにシリアルポートのコネクタが実装されていないものが多くなり、USB を仮想シリアルポートとして利用する「USB-Serial 変換ケーブル」というものが市販されています。この「USB-Serial 変換ケーブル」では、USB の規格に従って、データの送受信を行う信号は、D+と D-の 2 本しかありません。（線間差動信号と言います。）このために、上の図で示す RxD、TxD と接地信号以外は、使用されません。

シリアル通信の接続対象にもよりますが、糸電話で説明したように、シリアル通信では通信用の信号と接地信号の 3 本の信号があれば、通信ができるので、いろいろなところに使用されてきたわけです。

## 6. 通信手順と実際の接続

下の図は、2台の PC がシリアル通信を行う手順です。

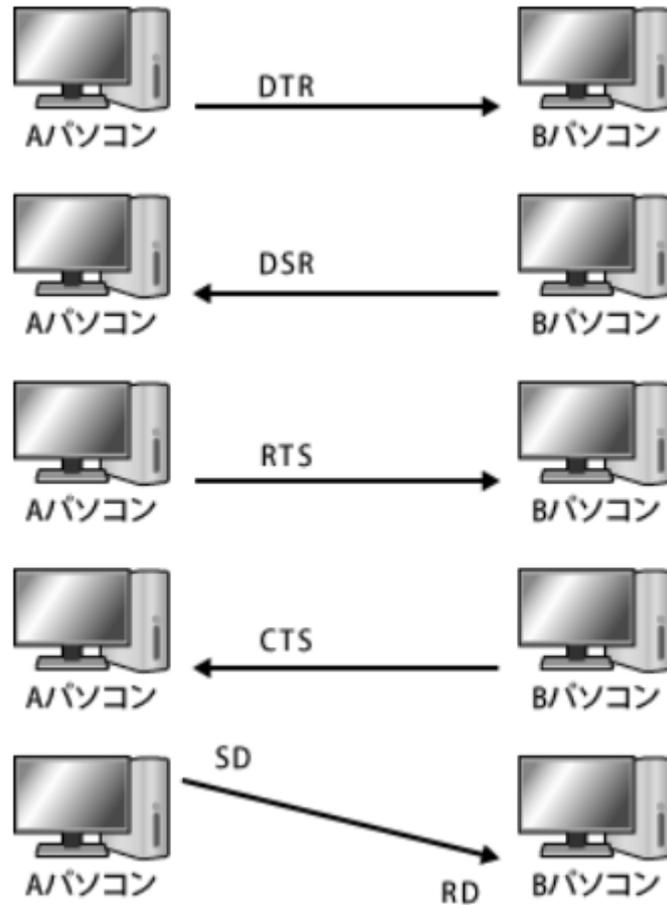


図 24 正式な通信手順（出典：CodeZine）

- ① まず A パソコンから DTR（信号を送る通路が使えます）を ON にします。B パソコンは DSR（私も準備 OK です）を ON にします。
- ② 次に A パソコンは RTS（データを送ってもよいですか）を ON にします。B パソコンは CTS（送ってもらって大丈夫です）を

を ON にします。

- ③ これで準備が整いました。A パソコンは TxD から信号を送ります。B パソコンは RxD から信号を受け取ります。

以上のような手順を行いながら、対向機（通信相手）との接続信号数を減少出来る接続方法が次の図です。

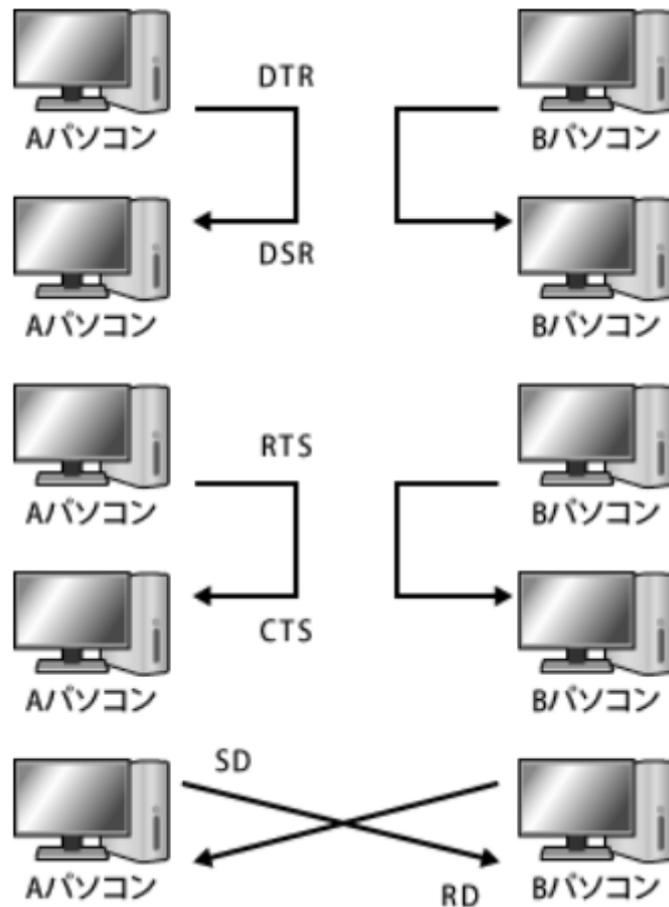


図 25 簡易な接続（出典：CodeZine）

もし A パソコンが DTR を出して DSR 待ちの時、自分が出した信号線がそのまま DSR につながっていると A パソコンは DSR 信号を相手から貰っ

たと思い次のステップに進みます。RTS も同様に自分が出した信号をそのまま CTS につなぐと送信可能と思い次のステップへ進みます。相手からの通信待ちで止まってしまうことなく接続開始ができるようになります。

そこで、ケーブルを作る時、4:DTR と 6:DSR のピンを自分のコネクタの中で接続します。7:RTS と 8:CTS も接続します。これで接続開始の処理は自分のパソコン側だけで行えるようになりました。そうすると実際に必要な信号は 2:RxD、3:TxD、5:GND の 3 本になります。

A パソコンの 2:RxD と B パソコンの 3:TxD をつなぎ、A パソコンの 3:TxD と B パソコンの 2:RxD をつなぐので信号線をクロスさせることからクロスケーブルと言います。

## 7. 信号線の中のデータ

送受信信号の中を伝わるデータはどのようなものでしょうか。

図に 1 バイトのデータが送られる様子を示します。

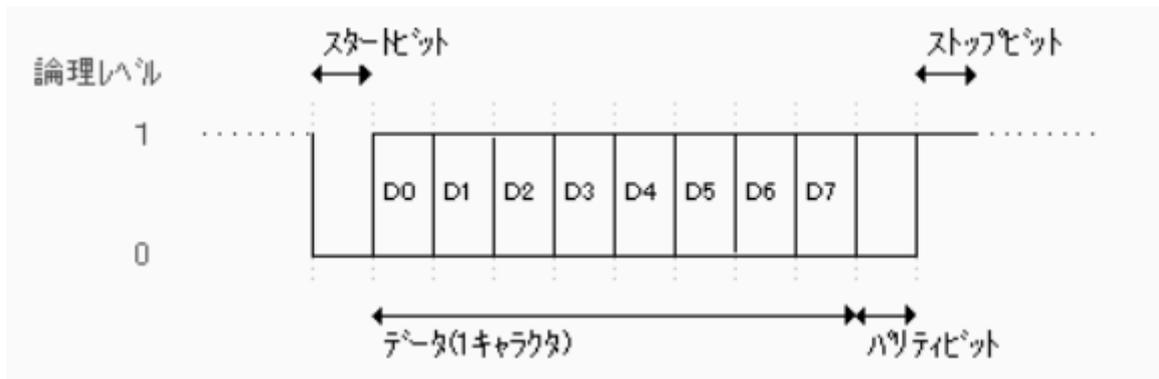


図 26 1 バイトの送信データ (出典 : Softech)

図では、左側に向かってデータが送られる様子を示しています。

インターフェース IC の送信レジスタに 1 バイトのデータを書き込むと、最初にスタートビットが送られて、相手に「これから通信データが来る」ことを知らせます。続いて、8bit のデータが送信され、後に誤り検出のためのパリティビットが続きます。最後にストップビットが送られて 1 バイトのデータの送信が終わります。

スタートビットは、1 ビットですが、ストップビットはパリティビットの有無などにより、1 または 2 ビットになります。パリティビットは偶数パリティ・奇数パリティ・パリティ無しが設定できます。パリティ無しに設定した場合は、ストップビットを 2bit にします。

シリアルインターフェースのメーカーや使用する IC の種類によっても、設定の内容が異なるので、直接このような設定を行うときは、IC のデータシートを良く読むことが必要です。

様々な家電製品や制御機器などに内蔵されるシリアル通信部分を開発する場合は、このような制御設定を細かく行うことにより、通信精度・通信品質を高めます。

通信速度はボーレート (Baud Rate) と呼び、単位は bps (Bits Per Second) で表します。

1 秒間に送信できるビット数を意味します。このボーレートは、スタートビット・データ・パリティビット・ストップビットのすべてを含めた送信速度なので、データ自体の通信速度は、ボーレートより遅くなることに注意してください。

D0	D1	D2	D3	D4	D5	D6	D7	P
1	0	0	0	0	0	1	0	0

図 27 'A'のデータ (偶数パリティ)

上図に半角の A (16 進数の H' 41) を偶数パリティで送信する場合を示します。

この時の実際の通信信号をロジックアナライザで見ると、次のようになります。

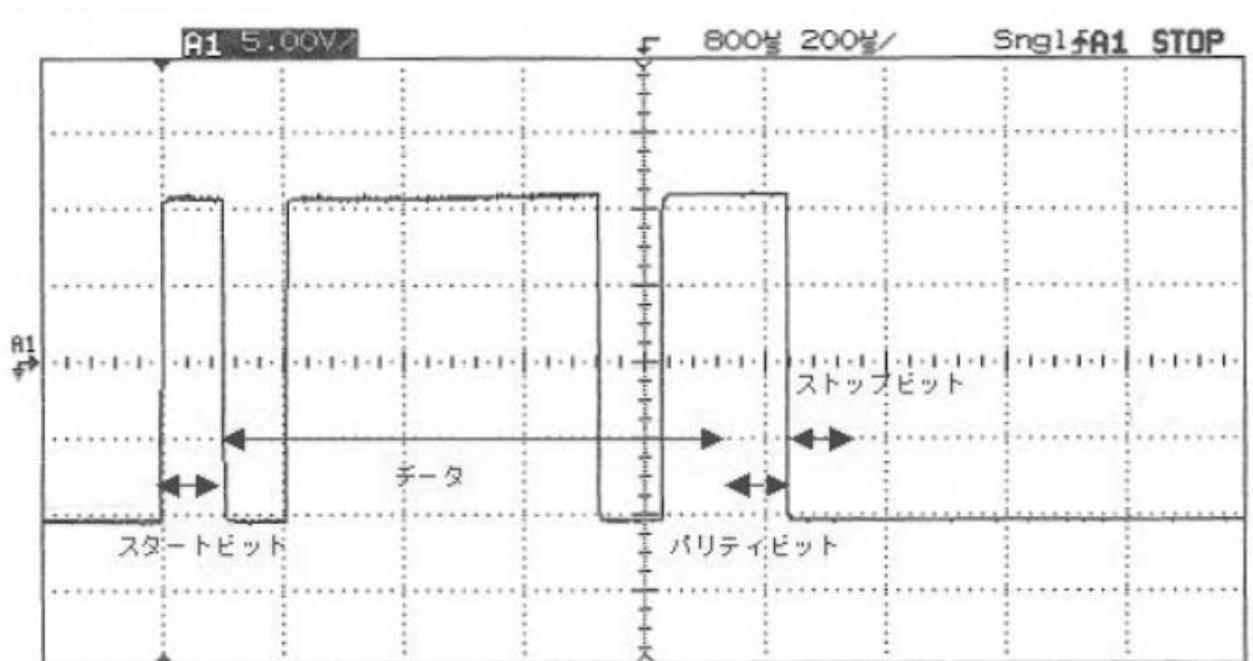


図 28 実際の送信信号

(9600bps H'41='A'・1ストップビット・偶数パリティ)

組み込みソフト開発ではソフトウェアの開発とともにハードウェアの開発をすることが良くあるため、ハードウェア側に不具合が存在する可能性もあります。実際の波形を見てどのように通信が行なわれている

かを理解することができれば、いろいろな側面からの問題解決が出来るようになります。

# C:生産技術者マネジメントスキル

## 認定試験サンプル

公開されている試験問題のサンプルと解答例を示します。

量産工場の組み立てラインや機械加工において、稼働時間や当該ライン、設備のデータを使って、1日の生産数を算出したい。以下の式について空欄に当てはまる適切な用語を下記語群から選びなさい。

$$\text{1日の生産数} = \frac{\text{稼働時間}}{\text{ピッチタイム}} \times (1 - \text{不良率}) \times \boxed{\phantom{000}}$$

語群

- a. 生産性
- b. 稼働率
- c. 作業員数
- d. ライン数

試験問題サンプル

図 29

量産工場の組み立てラインや機械加工において、稼働時間や当該ライン、設備のデータを使って、1日の生産数を算出したい。以下の式について空欄に当てはまる適切な用語を下記語群から選びなさい。

$$\text{1日の生産数} = \frac{\text{稼働時間}}{\text{ピッチタイム}} \times (1 - \text{不良率}) \times \boxed{\phantom{\text{生産性}}}$$

語群

- a. 生産性
- b. 稼働率
- c. 作業員数
- d. ライン数

図 30

以下の式は、VEにおいて、「価値」を定義する際に、使用される式である。

$$\text{価値 (V)} = \frac{\boxed{\phantom{\text{技術 (T)}}}}{\text{原価 (C)}}$$

空欄に当てはまる用語を語群から1つ選びなさい。

語群

- a. 技術 (T)
- b. 効果 (P)
- c. 付加価値 (Q)
- d. 機能 (F)

図 31

以下の式は、VEにおいて、「価値」を定義する際に、使用される式である。

$$\text{価値 (V)} = \frac{\text{[空欄]}}{\text{原価 (C)}}$$

空欄に当てはまる用語を語群から1つ選びなさい。

語群

- a. 技術 (T)
- b. 効果 (P)
- c. 付加価値 (Q)
- d. 機能 (F)

図 32

機械加工系の現場では、作業者と設備の作業タイミングを管理することで、生産性が大きく左右される。こうした職場で活用される、現状を把握し、生産性を向上させる手法として最適なものを、下記語群から1つ選択しなさい。

語群

- a. ワークメジャーメント
- b. ワークサンプリング
- c. マンマシン・チャート
- d. ピッチダイヤグラム

図 33

機械加工系の現場では、作業者と設備の作業タイミングを管理することで、生産性が大きく左右される。こうした職場で活用される、現状を把握し、生産性を向上させる手法として最適なものを、下記語群から1つ選択しなさい。

語群

- a. ワークメジャーメント
- b. ワークサンプリング
- c. マンマシン・チャート
- d. ピッチダイヤグラム

図 34

商品開発から量産に至るまでに、何段階かで試作が行われる。各段階における試作のねらいとして、最適なものを選択しなさい。

- |           |                             |
|-----------|-----------------------------|
| 1. 情報収集段階 | A                           |
| 2. 先行研究段階 | B                           |
| 3. 開発設計段階 | C                           |
| 4. 生産準備段階 | D                           |
| 5. 生産段階   | 主として改善案などを採用する際に新規設備等を検討する、 |

語群

- a.  商品の機能・性能をはじめとした商品性を検討する
- b.  商品開発を進めるかどうかを検討する
- c.  設備の加工・組立・メンテナンス性……などを検討する
- d.  アイデアが商品開発テーマになるかどうかを検討する

図 35

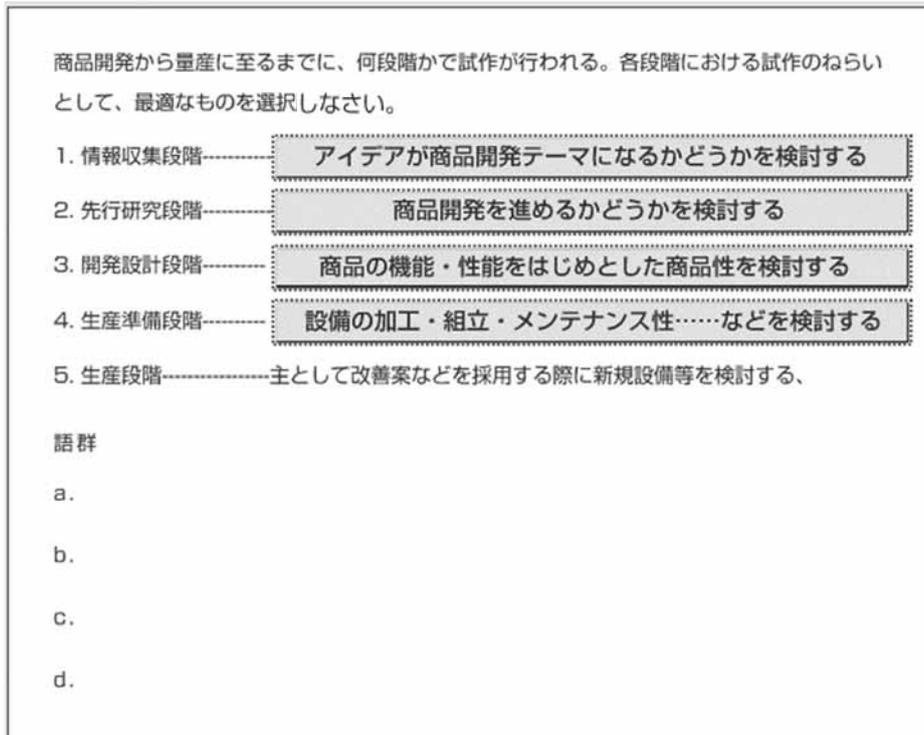


図 36

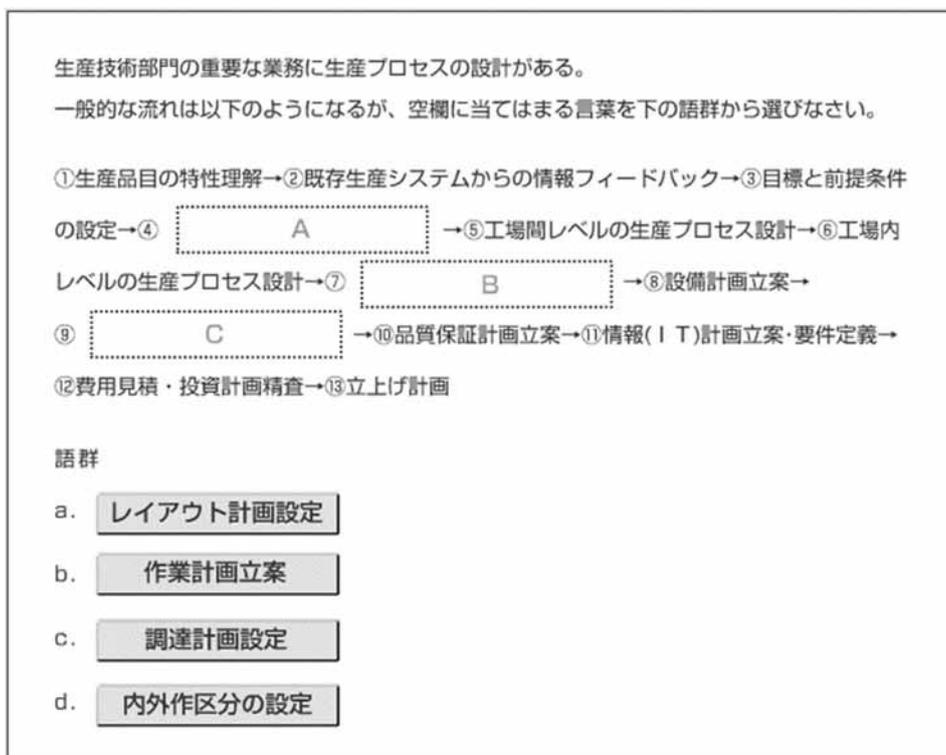


図 37

生産技術部門の重要な業務に生産プロセスの設計がある。  
 一般的な流れは以下ようになるが、空欄に当てはまる言葉を下の語群から選びなさい。

①生産品目の特性理解→②既存生産システムからの情報フィードバック→③目標と前提条件  
 の設定→④ **内外作区分の設定** →⑤工場間レベルの生産プロセス設計→⑥工場内  
 レベルの生産プロセス設計→⑦ **レイアウト計画設定** →⑧設備計画立案→  
 ⑨ **作業計画立案** →⑩品質保証計画立案→⑪情報(I T)計画立案・要件定義→  
 ⑫費用見積・投資計画精査→⑬立上げ計画

語群

a.

b.

c. **調達計画設定**

d.

図 38

(出典：日本能率協会)

## D:SW の情報を検出する2つの方法

コンピュータはデジタルのデータを処理できます。マイコン外部にある SW の ON/OFF をどのようにコンピュータに入力すれば、処理できるかを解説します。基本は、1 か 0、High か Low からです。1 は電圧（マイコンでは普通、3.3V か 5.0V）が掛かっている状態。0 は電圧が掛かっていない（あるいは何も繋がっていない、または接地されている、または GND に繋がっている）状態をプログラムから読み込んだときのデータです。

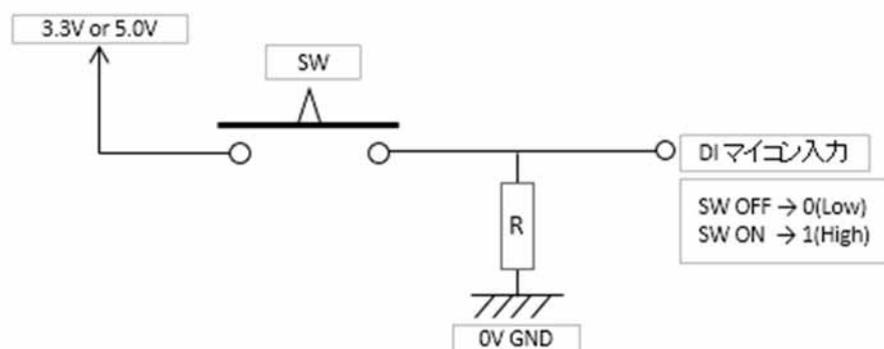


図 39

上図は、SW が押されていないとき 1(High)となる回路です。図で R とあるのは抵抗器で、この場合マイコンの入力端子 DI の電圧を通常 0V にしています。SW を押すと回路が閉じて、DI 端子に電圧(3.3V または 5.0V)が掛かり、そのときマイコン内部のプログラムで状態を読み込むとデータは 1(High)になっています。この場合の抵抗器を Pull Down 抵抗と言います。

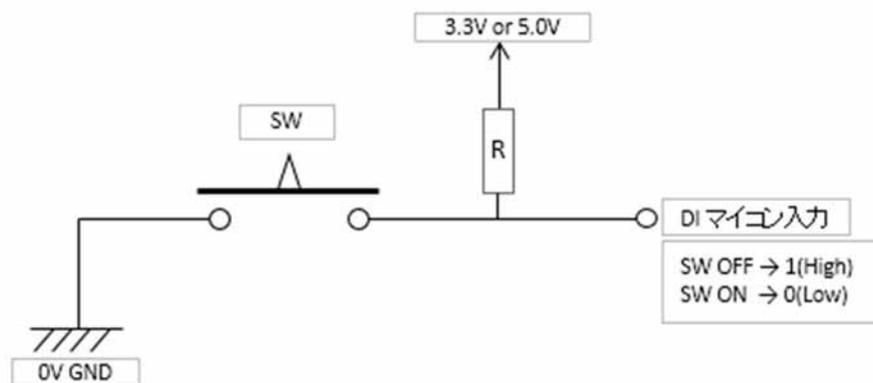


図 40

上図は、逆に、通常 1(High)、SW を押しているとき、0(Low)となる回路です。この場合の抵抗器を Pull Up 抵抗と言います。

この回路では、マイコンの入力端子 DI は通常電源に接続されているので、抵抗を通じて電圧(3.3V または 5.0V)が掛かっています。マイコン内部でこれを読み込むと、データは 1(High)となります。SW が押されて回路が閉じると、DI 端子は、0V(GND)接続されるので DI 端子の電圧も 0V になり、これをマイコン内部で読み込めば 0(Low)となります。

# E:TWE-Lite Data Sheet(抜粋)

---

TWE-Lite のデータシートの一部を掲載します。

## 製品概要

---

TWELITE DIP BLUE (認証型式:TWE-001 Lite)/TWELITE DIP RED (認証型式:TWELITE RED) は扱いやすい DIP 形状にした TWELITE で、超低消費電力かつ高性能なマイコン、フラッシュメモリ、IEEE802.15.4 準拠の高性能無線を備えます。

電源とセンサーなどを接続し、フラッシュメモリ、EEPROM にプログラムを格納することで動作させられます。SPI、I2C、UART をサポートしており、様々なセンサーやマイコンなどと接続可能です。

日本国内での認証を取得しておりますので、すぐに製品化が可能です。

図 41

## 主な特徴

---

- 世界標準規格である IEEE802.15.4 に準拠
- 弊社独自のプロトコルスタック“TWELITE NET”が利用可能
- 2.54mm ピッチ 28 ピン(600mil) DIP 型 IC の形状
- チップ性能を最大限に引き出す基板設計により長距離でも安定した通信が可能
- 32KB の RAM、160KB/512KB のフラッシュメモリを備え高性能な通信用アプリケーションソフトウェアの動作が可能
- 待機時の電流が  $0.1 \mu\text{A}$  (RAMOFF スリープ時)と非常に少ないため電池寿命を延ばすことが可能
- 4個または6個の AD コンバータ、1 個のコンパレータ、20 個の汎用入出力ポートといった豊富な I/O を内蔵しセンサー等を直接接続可能
- フラッシュメモリを内蔵しておりファームウェアの変更が可能
- 無償で入手可能な GNU および eclipse ベースの開発環境によりファームウェア開発が可能
- 強力な 128-bit AES 暗号化技術によりセキュリティを保つことが可能
- 日本国内の ARIB STD-T66 工事設計認証(技適)を取得済みであるため免許や新たな申請の必要なく使用が可能
- RoHS 対応により新環境基準に準拠

図 42

(出典：モノワイヤレス(株))

# F:ESP-8266 Data Sheet(抜粋)

ESP-8266 のデータシートの一部を掲載します。

Categories	Items	Values
WiFi Parameters	Certificates	FCC/CE/TELEC
	WiFi Protocols	802.11 b/g/n
	Frequency Range	2.4GHz-2.5GHz (2400M-2483.5M)
Hardware Parameters	Peripheral Bus	UART/HSPI/I2C/I2S/Ir Remote Control
		GPIO/PWM
	Operating Voltage	3.0~3.6V
	Operating Current	Average value: 80mA
	Operating Temperature Range	-40°~125°
	Ambient Temperature Range	Normal temperature
	Package Size	18mm*20mm*3mm
	External Interface	N/A
Software Parameters	Wi-Fi mode	station/softAP/SoftAP+station
	Security	WPA/WPA2
	Encryption	WEP/TKIP/AES
	Firmware Upgrade	UART Download / OTA (via network) / download and write firmware via host
	Software Development	Supports Cloud Server Development / SDK for custom firmware development
	Network Protocols	IPv4, TCP/UDP/HTTP/FTP
	User Configuration	AT Instruction Set, Cloud Server, Android/iOS App

(出典 : Espressif Systems(社))

図 43

赤枠内、Peripheral Bus では、入出力 (GPIO、PWM)、外部との通信に利用できるインターフェース (UART、HSPI、I2C、I2S、Ir) が数多く準備され、電源電圧も電池駆動の範囲内となっています。動作温度範囲も屋外で使用することを想定しても、十分な範囲です。

NO.	Pin Name	Function
1	3V3	3.3V power supply (VDD)
2	EN	Chip enable pin. Active high.
3	IO14	GPIO14; HSPI_CLK
4	IO12	GPIO12; HSPI_MISO
5	IO13	GPIO13; HSPI_MOSI; UART0_CTS
6	IO15	GPIO15; MTDO; HSPICS; UART0_RTS
7	IO2	GPIO2; UART1_TXD
8	IO0	GPIO0
9	GND	GND
10	IO4	GPIO4
11	RXD	UART0_RXD; GPIO3
12	TXD	UART0_TXD; GPIO1
13	GND	GND
14	IO5	GPIO5
15	RST	Reset the module
16	TOUT	It can be used to test the power-supply voltage of VDD3P3 (Pin3 and Pin4) and the input power voltage of TOUT (Pin 6). However, these two functions cannot be used simultaneously.
17	IO16	GPIO16; can be used to wake up the chipset from deep sleep mode.
18	GND	GND

(出典 : Espressif Systems(社))

図 44

上図に ESP-8266 の信号とピンの割り当てを示します。少ないピン数で多用途に利用できるように、信号の種類がバランスよく割り当てられています。

# G: Arduino IDE のインストール

---

## Arduino IDE

---



図 45

ソフトウェア開発には Arduino IDE を使用します。WEB で <https://www.arduino.cc/> を指定すると上のページが見つかります。ページ上部の SOFTWARE をクリックすると次のページに移動します。



図 46



図 47

さらに、すこし下に移動すると上図のページになります。ここで、Windows Installer をクリックすると、下図のページに移動します。

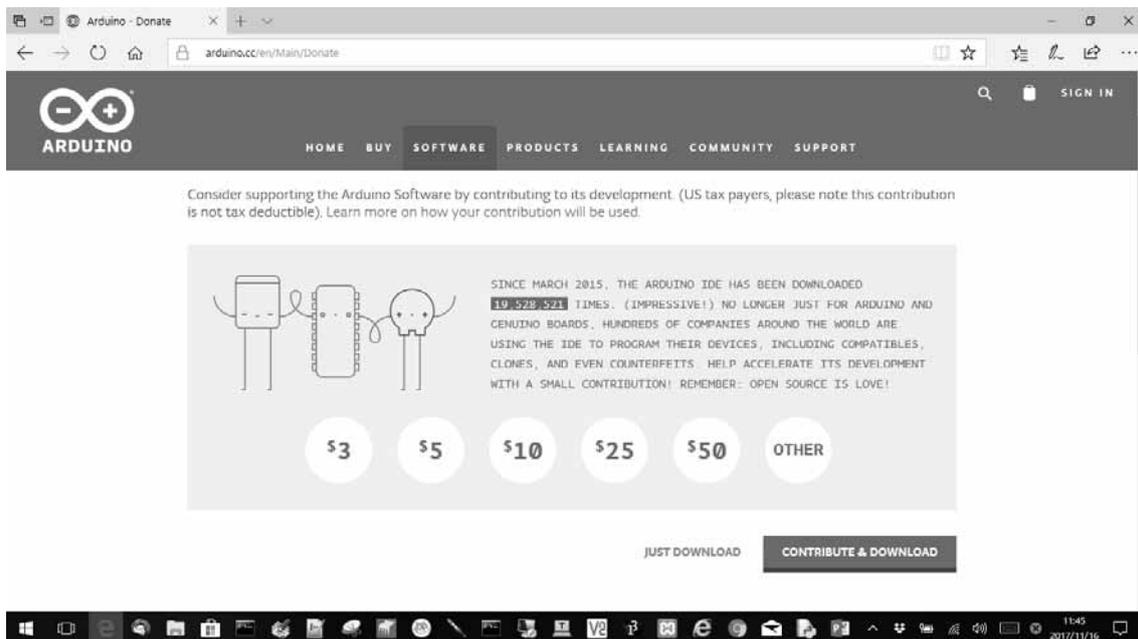


図 48

ここで、寄付される方は **CONTRIBUTE & DOWNLOAD** を、そのままダウンロードを続ける方は **JUST DOWNLOAD** を選択します。

適当なフォルダにダウンロードファイルを置き、そのファイルをダブルクリックして **Install** します。途中で **USB-シリアルドライバ** などの **Install** 確認メッセージが表示されることがありますが、全て **Install** してください。また **Java** が通信するという確認のメッセージが表示されることがありますが、全ての通信を **OK** (またはチェック) してください。Install が終わるとデスクトップに下図左のような眼鏡マークのショートカットができています。それをダブルクリックすることで、下図右のような **Arduino IDE** が起動されます。プログラムソースコードは、ウインドウ中央の白い部分に記述します。

## ◇Arduino 統合開発環境 IDE



◇ショートカットができます。  
実行するとIDEが開きます。

図 49

### IDEの環境設定

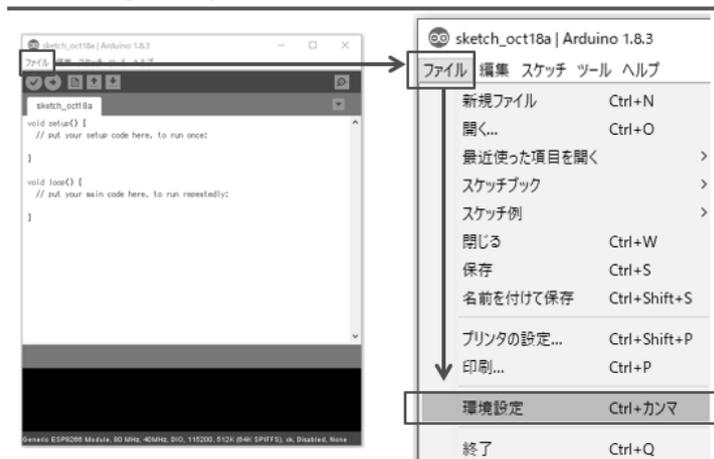


図 50

これから使用する WiFi マイコンモジュール専用のライブラリなどを準備します。上図で、【ファイル→環境設定】と辿ると、次のウィンドウが開きます。



図 51

ウィンドウ下部にある【追加のボードマネージャの URL】に次の URL を入力して、OK ボタンを押してください。

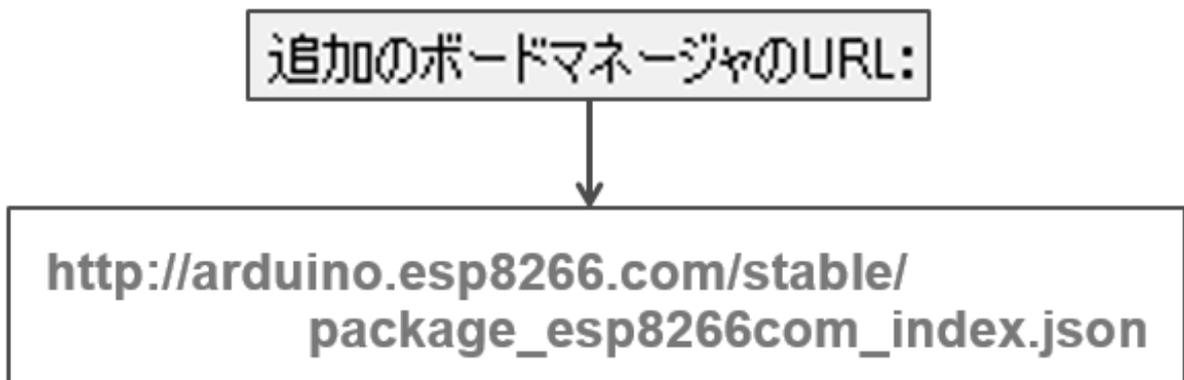


図 52

## ボードマネージャ

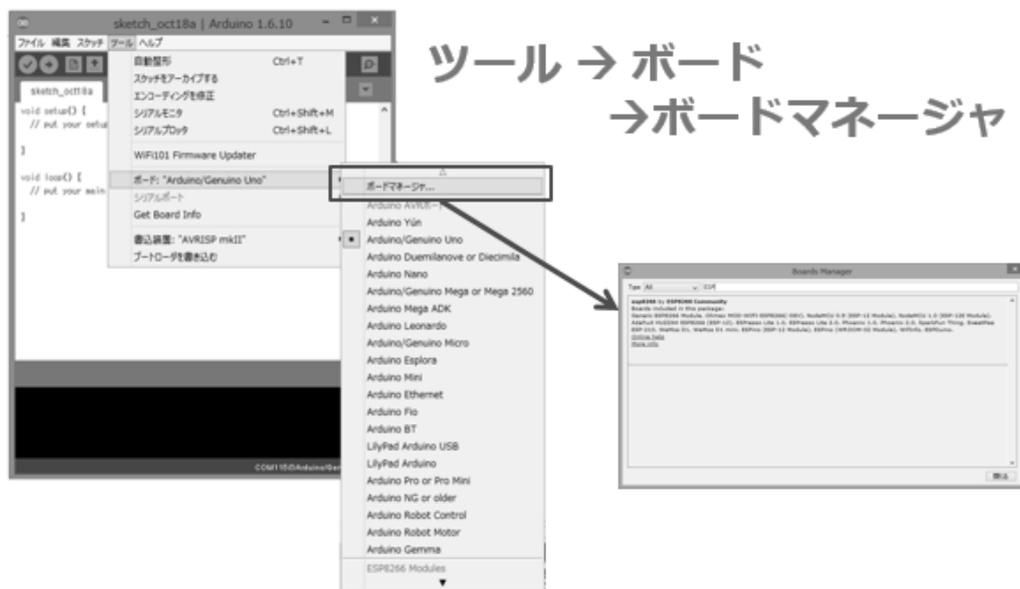


図 53

【ボード→ボードマネージャ】と辿り、ボードマネージャのウィンドウを開きます（下図）。

◇ 検索に **【ESP】** と入力します



図 54

右上にある検索フィールドに ESP と入力し、表示される esp8266 by ESP8266 Community を選択し、Install します。(下図)

## ◇ esp8266 by ESP8266 Community を選択



図 55

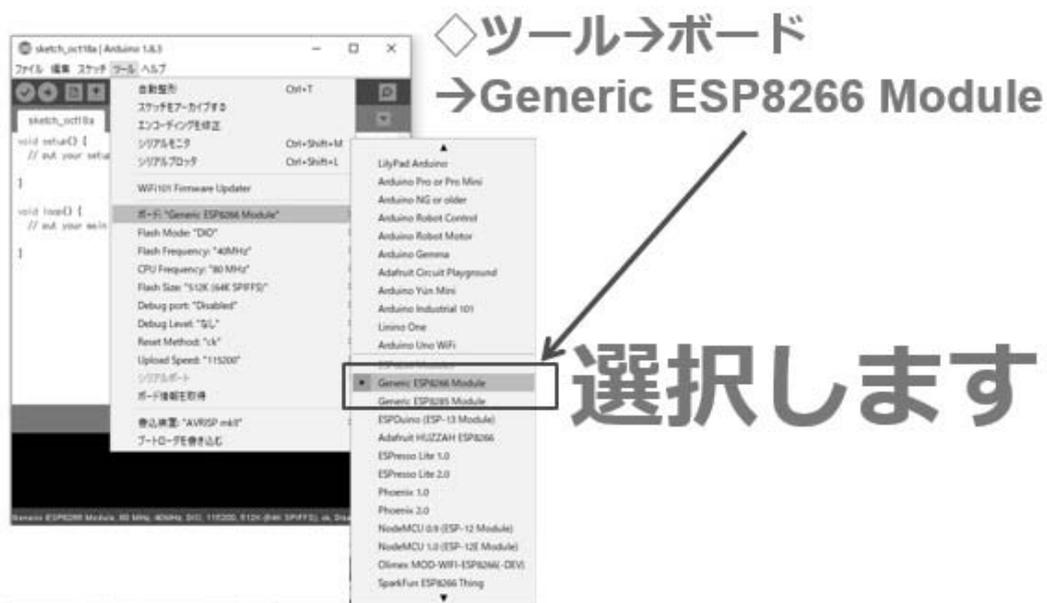


図 56

元のウィンドウに戻り、【ツール→ボード】と辿り、一覧の中から、Generic ESP8266 Module を選択します。

◇ ツールを選択、表示を確認してください。

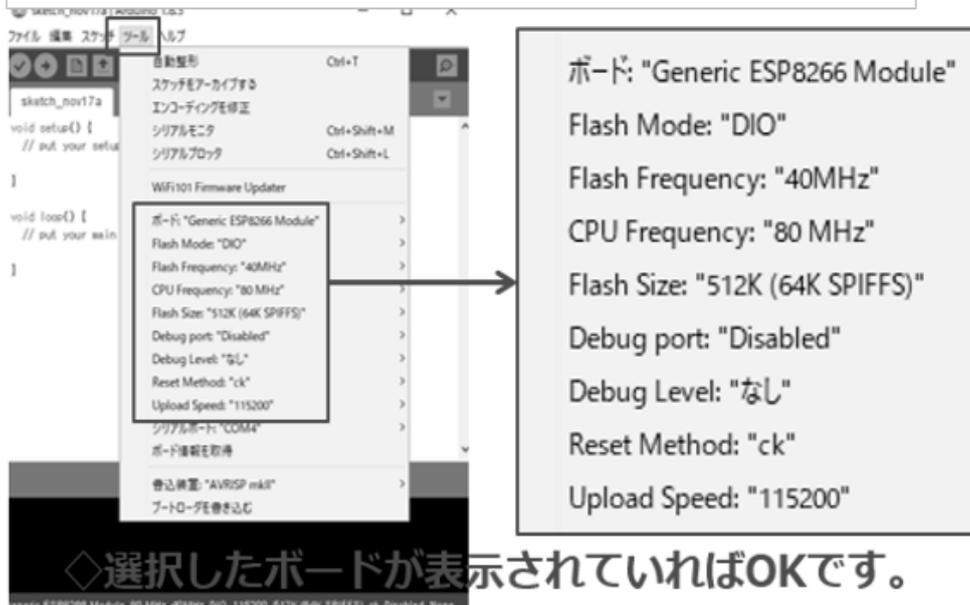


図 57

ツールで選択したボードが表示されていれば OK です。(上図)

## プログラムを書く

```

ESP_2101_brink_LED

#define LED_PIN 14 //<--- GPIO14 for LED
void setup() {
  pinMode(LED_PIN, OUTPUT);
}
void loop() {
  digitalWrite(LED_PIN, HIGH);
  delay(500);
  digitalWrite(LED_PIN, LOW);
  delay(500);
}
  
```

① LEDをGPIO14に接続した。

② setup()は、初期化処理。

③ GPIO14を出力に設定。

④ loop()は、繰り返し実行される。

⑤ LED HIGH=点灯 LOW=消灯。

⑥ 500ms(=0.5秒)待つ。

**C言語**  
 ※ファイル→名前を付けて保存

図 58

IDE ウィンドウ中央の背景の白い部分がソースコードエディタになっていますので、そこにプログラムを記述します。

**【重要】**

基本的に **Arduino** マイコンと同様の言語体系となっています。  
**Arduino** 言語は **C/C++**をベースにして、**C** 言語のすべての構造と、いくつかの **C++**の機能をサポートしています。

平成 30 年度「専修学校による地域産業中核的人材養成事業」

(Society5.0 等対応カリキュラムの開発・実証)

富山県をモデルとした「モノづくり」現場に

IoT を導入する中核的人材育成

製造 IoT 基礎概論 テキスト教材

平成 3 1 年 3 月発行

学校法人浦山学園 富山情報ビジネス専門学校

〒939-0341 富山県射水市三ヶ 5 7 6

TEL : 0766-55-1420 Fax : 0766-55-0757