

第II部

基礎編



第1章において「システムは、入力と処理、それに出力からなる」と説明しました。また、Arduinoでの「入力」はセンサー類の電子部品で行われ、「出力」するデバイスとしてはスピーカーやLEDといったものがあるとも紹介しました。また第2章では、Arduino通信によるアナログ入出力とデジタル入出力を解説しました。さらに第3章では、「処理」を行うプログラミングについて説明しました。さて、いかがでしょうか。ここまでで、ある程度初歩的なArduinoのプログラミングについては理解できていませんか？

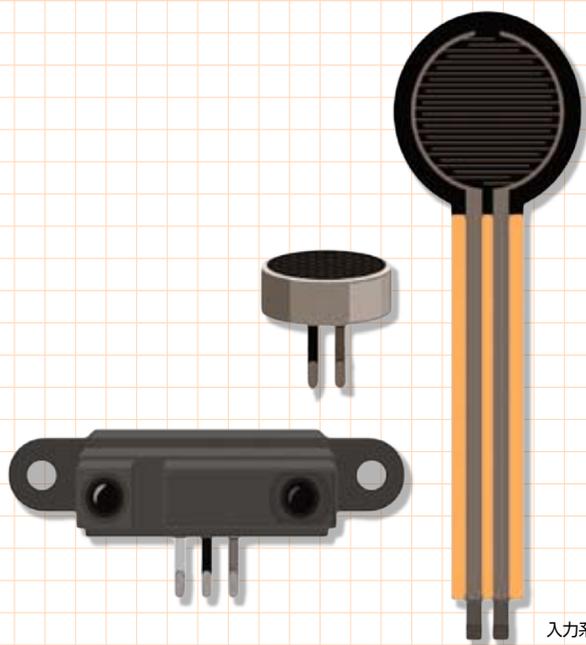
この基礎編では、Arduinoの入力と出力に関係する電子部品のデジタルとアナログに着目し、その違いと制御（プログラミング：処理）について解説します。

多くの電子部品は、その特性によって、入出力をアナログで行うかデジタルで行うかがあらかじめ決まっています。しかし、Arduinoでは、電子部品のアナログとデジタルの入出力を、それぞれ異なる関数で対応します。第4章と第5章では、これら基本となる関数を使って、アナログとデジタルの入出力を学んでいきます。

なお、第4章では入力系の電子部品のアナログ入力とデジタル入力を、第5章では出力系の電子部品のアナログ出力とデジタル出力について学びます。それぞれの違いをきちんと覚えるようにしましょう。

第 4 章

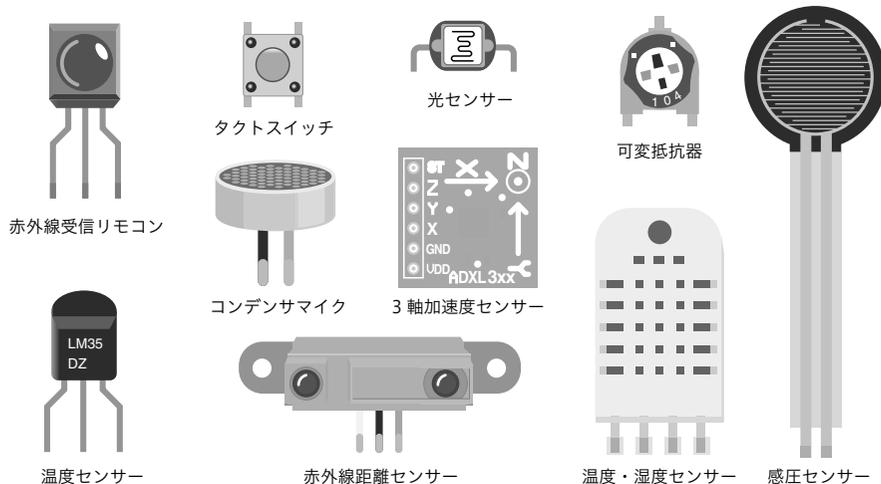
入力部品を使いこなそう



入力系の部品の一例

本章では、簡単な入力系の電子部品の使い方を説明します。入力系の電子部品には、スイッチやボリューム、それに多くのセンサー類があります。これらは、温度や湿度あるいは音など、外部環境の何らかの変化を読み取り、それを入力値としてArduino側に渡す役割を果たします。

図 4.1 入力系の電子部品群(主にセンサー類が豊富)



ここでは、デジタル入力系としては簡単なスイッチやボリューム、それとアナログ入力系として簡単に扱える可変抵抗器を取り上げて、入力系の電子部品の扱い方を学んでいきます。

扱う電子部品がアナログ入力なのかデジタル入力なのかによって、扱う関数も変わってきます。その区別をすることは重要です。アナログ入力とデジタル入力のそれぞれの関数を知り、また電子部品のどれがどの関数と関係するかも、一覧表を用意したので、それを見ながら覚えていきましょう。



4.1

アナログとデジタルの入力系を知る

入力系の電子部品には、センサー類をはじめ、マイクやカメラ、キーボード、マウスなど多くのものがあります。表4.1に、アナログとデジタルの入力部品と、それぞれが利用する関数を紹介します。

つまり、電子部品によって、アナログ入力として接続するのかデジタル入力として接続するかの注意が必要となり、使う関数が異なることになります。

表 4.1 アナログ入力とデジタル入力の関数と電子部品(一般)

入力方法	アナログ入力	デジタル入力
利用する電子部品	<ul style="list-style-type: none"> ・ボリューム ・可変抵抗器 ・光センサー ・温度センサー ・赤外線距離センサー ・加速度センサー 	<ul style="list-style-type: none"> ・スイッチ ・振動／傾斜センサー ・チルトセンサー ・磁気センサー ・人感センサー ・赤外線リモコン受信モジュール
利用する関数	analogRead関数を利用する	digitalRead関数を利用する

※高度なセンサーなどは、この対応表と異なる製品も存在します。

それと、第1章の図 1.10 と次の表 4.2 に示す Arduino ボード上のアナログとデジタルの入力ピンの位置の違いについても十分注意してください。

表 4.2 Arduino Uno 入力系のピン

信号	Arduino Uno 入力処理系のピン (関連関数)
アナログ入力	A0 ~ A5ピン対応 (analogRead関数を利用する)
デジタル入力	① D0 ~ D13 ピン対応 (digitalRead関数を利用する) (A0 から A5 までを D14 から D19 までで利用可能) ② その他シリアル通信 (UART, I2C, SPI)

以下では、Arduino が入力系電子部品の値を読み込むための、デジタルとアナログの入力関数、そしてデジタル入力でも重要となるプルアップ抵抗を紹介します。

(1)アナログ入力の関数

アナログ系の電子部品 (可変抵抗などのボリュームなど) を、Arduino に接続してデータ値を読み取るには、アナログ入力関数「analogRead」を使います。このアナログ入力関数「analogRead」の戻り値 (電子部品から読み取った値) は、整数の「0 から 1023 まで」の値となります。

表 4.3 アナログ入力に関する関数

関数名	説明
analogRead (ピン番号);	アナログ入力の値の読み込み 戻り値: 0 ~ 1023 で読み込まれる 0 は 0V で、1023 は 5V の意味

整数の「0 から 1023 まで」の値となるということは、アナログ値の幅を 1024 の分解能で取得できることになります。ただし、この取得した値を**単位** (距離: cm、温度: °C、湿度: % など) のある値に変換する場合は、それぞれの電子部品の特性に応じた変換 (計算) 式を使います。この変換式は、電子部品の仕様書に記載されたグラフや式などを用いてプログラミングする必要があります。

(2) デジタル入力の関数

続いてデジタル入力が必要となる関数群を表4.4にまとめました。デジタル入力の場合は、2つの関数「`pinMode`」と「`digitalRead`」を利用します。

表4.4 デジタル入力に関する関数

関数名	説明
<code>pinMode</code> (ピン番号, モード);※	ピンの動作を入力か出力に設定 ・ピン番号: 設定するピン番号 ・モード: 入力 (INPUTまたはINPUT_PULLUP)
<code>digitalRead</code> (ピン番号);	デジタル入力の値の読み込み ・戻り値: Onの状態 (HIGH) またはOffの状態 (LOW) ・ピン番号: D0からD13または、A0 (D14) からA5 (D19) まで利用できる

※モードが「INPUT」の場合は省略可

「`pinMode`」関数は、デジタル入出力の設定用として利用します。したがって、この関数の2番目の引数の「モード」に、「OUTPUT」もしくは「INPUT」、「INPUT_PULLUP」のいずれかを設定します。ただ「`pinMode`」関数の2番目のパラメータ「モード」が「INPUT」の場合には、この「`pinMode`」関数は省略できます。

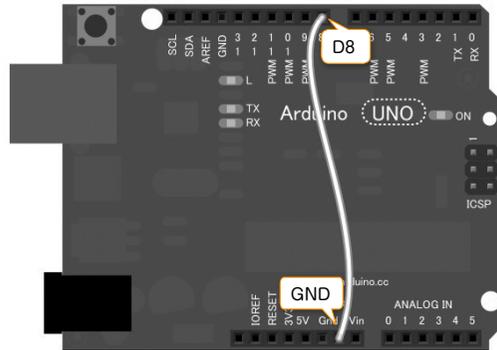
また「`digitalRead`」で読み取った値は、HIGH (=1) あるいは、LOW (=0) だけであり、超音波センサーなどの距離に変換する場合には、少し複雑な処理が必要となります。その詳細は第6章で紹介していきます。

(3) デジタル入力でのプルアップ抵抗について

電子回路の専門的な知識のひとつに**プルアップ抵抗**があります。一般にデジタル入力の場合、HIGHとLOWの中間電圧の状態では誤動作を起こします。Arduinoでは、この対策として、「`pinMode`」関数にプルアップ抵抗を取り入れ、この誤動作を防止しています。

デジタル入力関数「`digitalRead`」は、入力部品からの値の読み込み関数です。この関数による戻り値は、「`pinMode`」の「モード」が「INPUT」か、もしくは未設定の場合には、**HIGH** (約3.0V以上) か **LOW** (約2.0V以下) の値となります。しかし、思ったような値にならない場合があります。このことは、例えば図4.2にあるように単純にデジタル入出力ポートのD8番ピンとGNDをつなぎ合わせて、スケッチ4.1を起動すれば確認できます

図4.2 Arduinoのプルアップ抵抗のテストのための配線



スケッチ4.1 pinModeを使ったプルアップ抵抗のテスト

```

void setup(){
  Serial.begin(9600);
  pinMode(8,INPUT); // INPUT ⇒ INPUT_PULLUPに変更
}
void loop(){
  Serial.println(digitalRead(8)); // デジタルD8の値表示
  delay(1000);
}

```

「pinMode(8,INPUT);」は省略可能です

この状態で、ケーブルを接続したり切り離したりして、シリアルモニタ画面に出てくる値を確認してみてください。「pinMode」の2番目の引数が「INPUT」の場合は、必ずしも「0」か「1」の一定の値の状態にはならないと思います。

続いて、このスケッチの3行目の「モード」のパラメータ「INPUT」を「INPUT_PULLUP」に変更して実行してみてください。今度はうまく反応するはずですよ。

実際には、接続した場合「0」(=LOW)が表示され、切り離した場合「1」(=HIGH)と表示されるはずですよ。この「INPUT_PULLUP」が、プルアップ抵抗を宣言するためのもので、重要な知識となります。



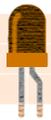
コラム

プルアップ抵抗のパラメータの「INPUT_PULLUP」変数は、初期のIDEではサポートされていませんでした。以前は、デジタル出力関数「digitalWrite」を使って、以下のように記述していました。もちろんこの方法は今でも利用できます。

```

pinMode(8,INPUT);
digitalWrite(8,HIGH); // デジタル出力関数でHIGHを宣言

```



4.2 アナログ入力(可変抵抗器と電圧測定)を知る

アナログ入力となる電子部品は、表4.3に示す「analogRead」関数を使って値を読み込みます。この「analogRead」による戻り値は、整数の「0から1023まで」となります。なお、Arduino Unoでは、この「0から1023まで」の値は電圧の0Vから5Vに相当します。

アナログ入力として使える電子部品は、光センサーや温度センサー、距離センサーなど多くのセンサー類が含まれます。簡単なものでは可変抵抗器（ボリュームなど）があります。これらをアナログ入力力で値を読み取ると、この「0から1023まで」の数値を返してくれます。具体的には、光センサーが明るさを感知したり、温度センサーが温かさを感知したりすると、これらの戻り値は小さくなるということです。

ただし、アナログの電子部品の一部には、測定した値に単位が関係することがあります。温度センサーの場合には、摂氏温度(°C)か華氏温度(°F)の単位があり、距離センサーの場合には、cm(センチメートル)やinch(インチ)の単位があります。つまり、「analogRead」が返してくる「0から1023まで」の値を、それぞれの単位に変更するための変換式が必要となります。

それでは、ここでの課題として、可変抵抗器（ボリュームなど）を使って抵抗(Ω：オーム)の値を読み取ることと、乾電池の電圧(V：ボルト)の測定について挑戦してみましょう。

(1)可変抵抗器とその配線

可変抵抗器には、3本のピン（リード線という）があります。まずは図4.3のように、それぞれArduino上のGNDと電源(5V)、それにアナログ入力ポートのA0番に接続します。可変抵抗器のピンには極性はありません。つまりプラスとマイナスの明確な区別はないので、両端をGNDと電源のいずれかに接続し、真ん中のピンをアナログ入力ポートのA0番に接続してください。ここでは10KΩの可変抵抗器を使いますが、それ以外の抵抗値を持つものでもかまいません。

図4.3 可変抵抗器を使った例

