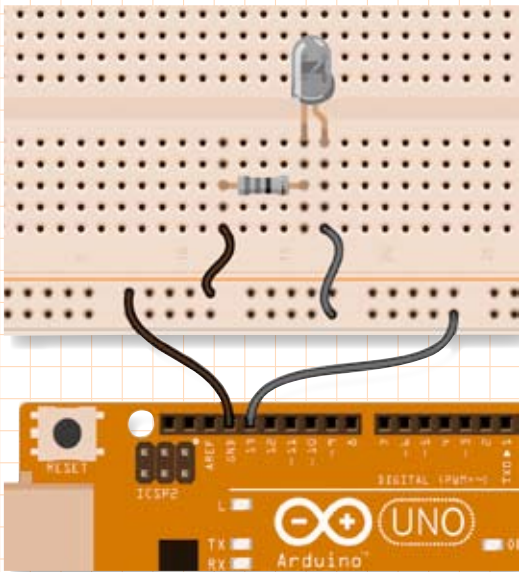


# 第 2 章

## Arduino を動かしてみよう



ブレッドボードを使ったLEDと抵抗の接続例

本章では、さっそく Arduino を使って何かを動かしてみましょ。ここでは、すでに IDE に用意してあるサンプル・スケッチを PC 上に呼び出し、それを Arduino 上で実行させてみます。

スケッチの中に記述されている内容を紹介しながら、プログラミングとは何かを紹介していきます。まずは基本的なプログラミングの中のコメントや変数、それに関数などの理解を深めていきましょう。またその処理の流れも見ていきましょう。

ソフトウェア開発の経験者は、Arduino との連携がどうなっているかのポイントだけ注意して見ていってください。未経験者は、プログラムに含まれる用語とその意味、さらに処理の流れを理解するようにしていってください。

この章の説明に沿って Arduino を動かすだけで、Arduino およびソフトウェア開発、どちらの初心者でも、「いかに Arduino が簡単か」が理解できるようになっています。さらに、この章の内容を理解していくことで、つぎの第3章や第4章へ入る準備にもなっています。

準備するものは、前半では IDE がインストールされた PC と Arduino を USB ケーブルでつなぐだけです。後半はブレッドボードとワイヤーの使い方などを学んでいきます。それと、Arduino 上の電子部品との関係で、互いの通信についても基本的なものを紹介します。



2.1

## PC と Arduino との USB ケーブル接続確認と注意事項

デバイスドライバの設定については、「1.6 統合開発環境 (IDE) の準備」の「PC と Arduino の接続のためのドライバ設定とその確認」(P.23) でも説明したとおりです。Arduino のドライバが PC 上にインストールされていれば、USB ケーブルで PC と Arduino をつなぐだけで、IDE 上のメニューバーの「ツール」にある「シリアルポート」に Arduino が認識できるようになります。

図2.1 Arduino のシリアルポート (COM の認識) のメニュー

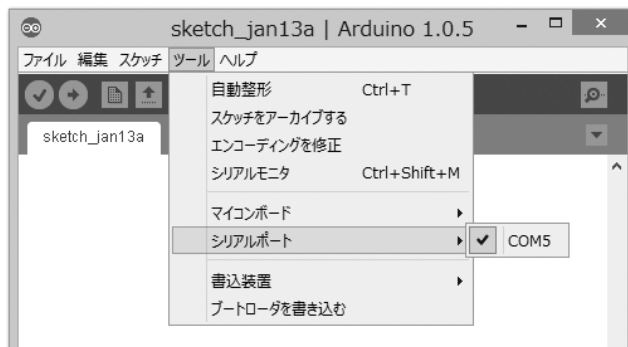


図2.1に示すように、表示された Arduino の COM 番号を選択して、PC と Arduino とを接続しま

す。ただし、複数のCOM番号が表示される場合には、第1章で説明した図1.22 (P.24)にあるデバイスマネージャー画面でArduinoのCOMがどれなのかを確認してください。もしくは、一度、PCからArduinoを切り離し、どれが切り離されたかを確認し、再度Arduinoをつないで、どのCOM番号が追加されたかを再確認してください。

なお、USBケーブルを接続する際は、Arduinoのデジタル入出力ポートのD0やD1を使用する電子部品や拡張ボードは、あらかじめ取り外しておいてください。デジタル入出力ポートのD0やD1を使用する電子部品や拡張ボードは、ソフトウェアの組込みが終わった時点でつなぐようにしてください。もし取り外さないまま、PCとArduinoのUSBケーブルを使ってシリアル通信(D0とD1を利用)を行おうとすると、Arduinoの書き込みで次のような通信エラーが発生します。このエラーは、初心者の多くが陥りやすいものの1つです。

```
avrdude: stk500_getsync(): not in sync: resp=0x00
```



## 2.2 サンプル・スケッチを動かしてみよう

それでは、これから具体的にサンプル・スケッチを使って、その動かし方を見て行きましょう。

最初の事例は、Arduino上での電子部品の組み立て(配線)はまったく不要です。またスケッチも、すでにIDEに組み込まれているサンプルの事例を使って動かしてみます。

### (1)スケッチの作成(サンプル・スケッチの読み込み)

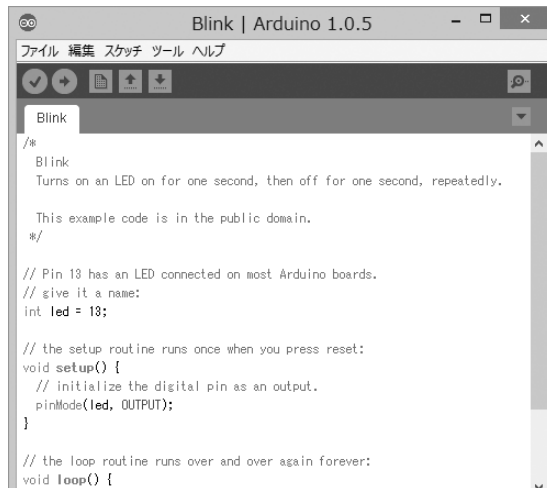
まずPC上のArduinoを起動し、図2.2のようにメニューバー「ファイル」から、「スケッチの例」を選択します。つぎに表示されるメニューの中の最上位の「01. Basics」を選択し、その中の「Blink」を選択します。

図2.2 サンプル・スケッチ「Blink」の読み込み



すると、図2.3のように、「Blink」のタブ名が表示され、スケッチ・エディタには、プログラムが表示されます。スケッチ2.1には、プログラミングの内容について簡単な説明を記載しておきます。

図2.3 サンプル・スケッチ「Blink.ino」



```

/*
  Blink
  Turns on an LED on for one second, then off for one second, repeatedly.
  This example code is in the public domain.
  */

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000); // wait for a second
  digitalWrite(led, LOW); // turn the LED off by making the voltage LOW
  delay(1000); // wait for a second
}

```

Annotations in the image:

- Comments (コメント) are indicated by a bracket on the right side of the first block of code.
- Global variable declaration (グローバル変数設定) is indicated by a callout pointing to `int led = 13;`.
- Initialization function (初期設定関数) is indicated by a callout pointing to the `void setup()` block.
- Comments (コメント) are indicated by callouts pointing to `void setup()` and `void loop()`.
- Loop function (繰り返し関数) is indicated by a callout pointing to the `void loop()` block.

## (2) Arduinoの実行(コンパイルとスケッチ書き込み、実行)

次に読み込んだスケッチをコンパイルし、Arduinoに書き込んでみます。


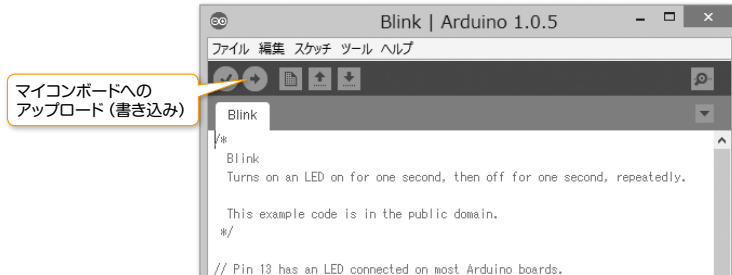
IDE上にサンプル・スケッチ「Blink.ino」(Arduinoの拡張子は「ino」)を読み込んだ状態で、図2.4に示すように、IDEのツールバーにある右矢印のアイコン(コンパイルとArduinoへの書き込み)「」を選択すれば、スケッチのコンパイルとArduinoボード上への書き込みが行われます。

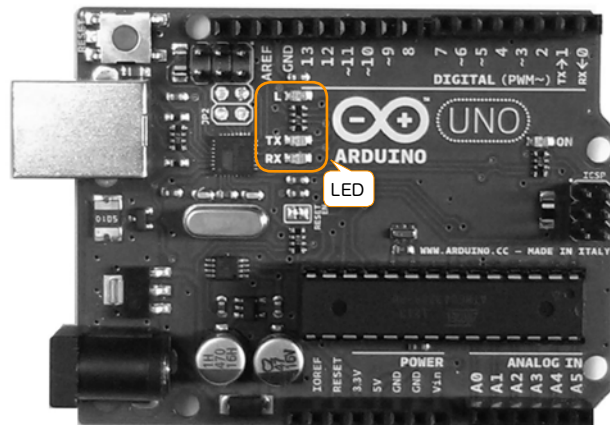
図2.4 スケッチのコンパイルとマイコンボードへの書き込みを行うアイコン選択



Arduinoへ書き込みが行われている間、Arduino上の3つのLEDが点滅します。これは、前述したように、デジタル入出力ポートのD0とD1を使ってシリアル通信を行っていて、アップロード(書き込み)の通信状態であることを示しています。

それでは、アップロードが終了した後、Arduino上のLEDで何が実行されているかを注意深く見てください(図2.5)。「L」のLEDが1秒間点灯し、1秒間消灯する繰り返しが確認できたらプログラム「Blink(明滅する)」は成功です。

図2.5 Arduino上の3つのLED



このように、PCとArduino、それにUSBケーブルだけあれば、動かすことができる簡単な例で、ほかの電子部品などは一切不要です。