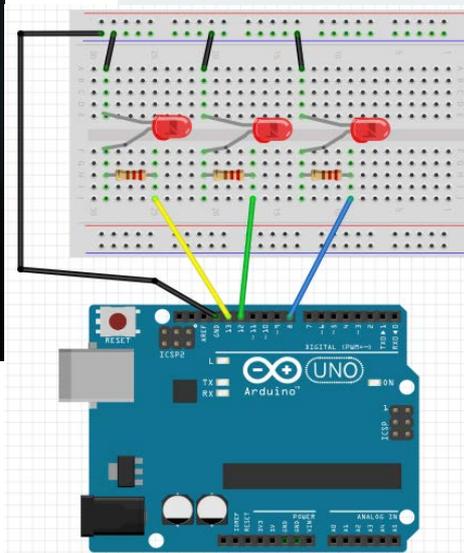
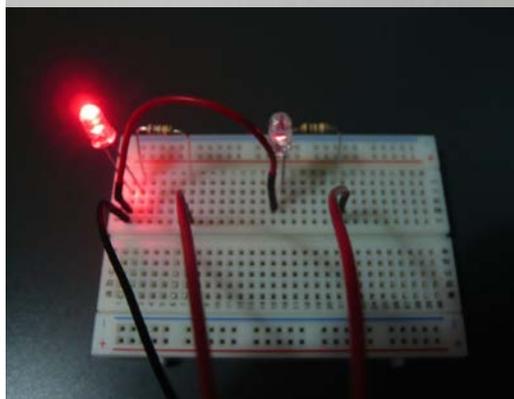
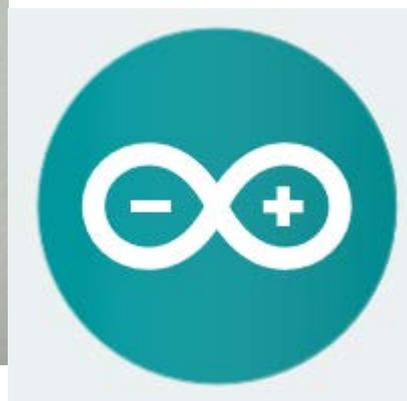


Arduino 編



目次

1	Arduino とは	1
2	Arduino のハードウェアについて	2
3	必要なものは？	4
4	動かしてみよう！	7
例題 1	Arduino ボード上のランプ L (LED) を 0.5 秒毎に点滅するプログラムを作成しよう。	7
例題 2	ブレッドボードを使用して LED を点灯させよう。	11
例題 3	2 個の LED を同時に 0.5 秒で点滅させよう。	13
例題 4	2 つの LED の明るさを制御させよう。	15
例題 5	PWM を使って明るさを制御しよう。	16
例題 6	PWM の応用を確認しよう。LED が徐々に明るくなる (フェードイン) を作成する。	17
例題 7	タクトスイッチで LED を点灯させよう。	18
例題 8	30 秒で LED がすべて消える「30 秒タイマー」を作ろう。	19
例題 9	フルカラー LED を制御してみよう。	21
例題 10	フルカラー LED の明るさを制御してみよう。	23
例題 11-1	圧電スピーカーを使って音を出してみよう。	24
例題 11-2	圧電スピーカーを使って音階を出してみよう。	25
例題 11-3	圧電スピーカーの音をだんだん高くしてみよう。	26
例題 11-4	圧電スピーカーをセンサにしてみよう。	27
例題 12-1	感圧センサで圧力を測定しよう。	28
例題 12-2	感圧センサと圧電スピーカー (1)	29
例題 12-3	感圧センサと圧電スピーカー (2)	31
例題 13-1	光センサ (Cds) で明るさを調べよう。	31
例題 13-2	光センサで LED の点灯を変化させよう (1)	32
例題 13-3	光センサで LED の点灯を変化させよう (2)	34
例題 14-1	温度センサで温度を測定しよう。	35
例題 14-2	温度センサで温度を測定しよう。	36

例題 15-1	距離センサを使ってみよう (1)	38
例題 15-2	距離センサを使ってみよう (2)	40
例題 15-3	距離センサを使ってみよう (3)	41
例題 15-4	人感センサを使ってみよう	43
例題 16-1	サーボモータを指定した角度に回転させよう。	44
例題 16-2	サーボモータを指定した角度に動かそう。	45
例題 16-3	サーボモータを指定した角度で往復しよう。	46
例題 16-4	サーボモータを可変抵抗から動かそう。	46
5	Processing とは	47
例題 17-1	Processing で図形描画 1 ～長方形～	50
例題 17-2	Processing で図形描画 2 ～円～	50
例題 18	Processing で図形描画 3 ～数を描画、塗りつぶし～	50
例題 19-1	Processing で図形描画 4 ～動く図形 1～	51
例題 19-2	Processing で図形描画 4 ～動く図形 2～	51
例題 19-3	Processing で図形描画 4 ～動く図形 3～	52
例題 19-4	Processing で図形描画 4 ～動く図形 4～	52
例題 20-1	Arduino-Processing でシリアル通信 1	53
例題 20-2	Arduino-Processing でシリアル通信 2	54
例題 20-3	Arduino-Processing でシリアル通信 3	56

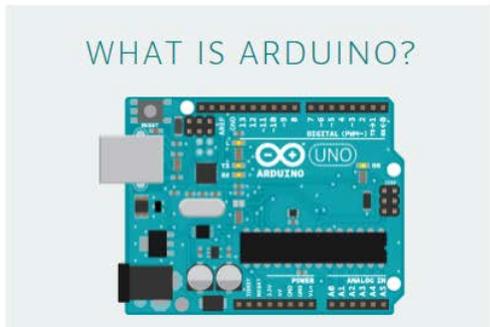
3 必要なものは？

Arduino を使うときに必要になるのは、次の通りである。

【部品リスト】

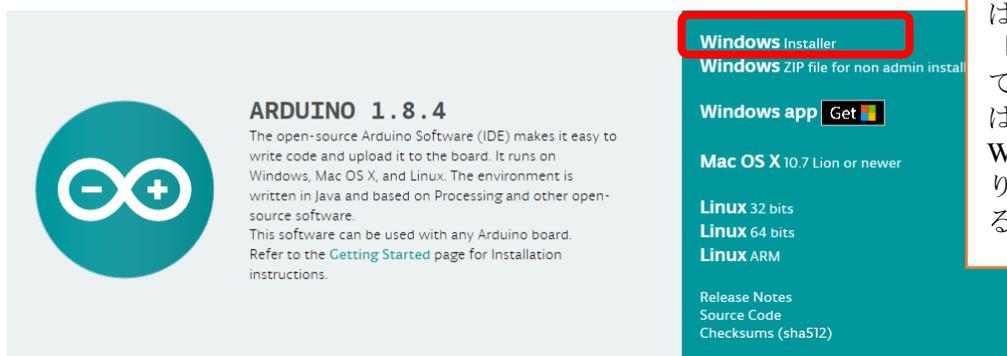
部品	参考価格	備考
Arduino Uno	3,000 円程度	最も一般的な Arduino。互換性のあるもので 500 円を切る価格で同等の性能の互換機もある。
USB ケーブル	130 円	A-B タイプのもの

- (1) Arduino ソフトウェアをインストール(無料)するためのサイトにアクセスする。
<http://arduino.cc> からダウンロードする。
- (2) Windows を選択し、自分のコンピュータにダウンロードする。



<http://arduino.cc> の「SOFTWARE」よりダウンロードできる。

Download the Arduino IDE



ARDUINO 1.8.4
The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software. This software can be used with any Arduino board. Refer to the [Getting Started](#) page for installation instructions.

Windows Installer
Windows ZIP file for non admin install

Windows app 

Mac OS X 10.7 Lion or newer

Linux 32 bits
Linux 64 bits
Linux ARM

Release Notes
Source Code
Checksums (sha512)

左のバージョンは、「ARDUINO1.8.4」で、windows の場合は、WindowsInstaller よりダウンロードする。

Previous IDE Releases

ARDUINO 1.8.2

ARDUINO 1.0.6

「ARDUINO 1.8.2」をダウンロードする場合

「ARDUINO 1.0.6」をダウンロードする場合。

これまでにリリースされた色々なバージョンをダウンロードすることができる。

最新版の 1.x.x または、1.0.x をダウンロードする。

複数の Arduino IDE がある理由の 1 つに、これまで Arduino 創業者陣営である Arduino LLC (arduino.cc) と Arduino Sr (arduino.org) が、2 つに分裂していたことがあげられる。分裂前は Arduino IDE 1.0.X だったが、分裂後は Arduino IDE 1.6X (arduino.cc) と Arduino IDE 1.7.X (arduino.org) が混在することになり、それぞれどちらか一方の Arduino IDE でしか開発できない場合もあった。(分裂以前からあった Arduino IDE 1.0X はどちらでも使える)

その両者が 2016 年末に統合することになり、Arduino IDE 1.8.X を発表した。それまでの Arduino IDE である 1.6.X と 1.7.X では、メニュー構成が微妙に異なるなどの要素もあるが、これまで分かれていた Arduino IDE が、一本化されたことになり、Arduino IDE 1.6.X または 1.7.X を使わなければ開発が行えないなどの混乱が解消されることが期待される。

Contribute to the Arduino Software

Consider supporting the Arduino Software by contributing to its development. (US tax payers, please note this contribution is not tax deductible). [Learn more on how your contribution will be used.](#)

SINCE MARCH 2015, THE ARDUINO IDE HAS BEEN DOWNLOADED **18,363,340** TIMES. (IMPRESSIVE!) NO LONGER JUST FOR ARDUINO AND GENUINO BOARDS, HUNDREDS OF COMPANIES AROUND THE WORLD ARE USING THE IDE TO PROGRAM THEIR DEVICES, INCLUDING COMPATIBLES, CLONES, AND EVEN COUNTERFEITS. HELP ACCELERATE ITS DEVELOPMENT WITH A SMALL CONTRIBUTION! REMEMBER: OPEN SOURCE IS LOVE!

\$3 \$5 \$10 \$25 \$50 OTHER

JUST DOWNLOAD CONTRIBUTE & DOWNLOAD

デスクトップ上などに、ZIP ファイルを保存する。

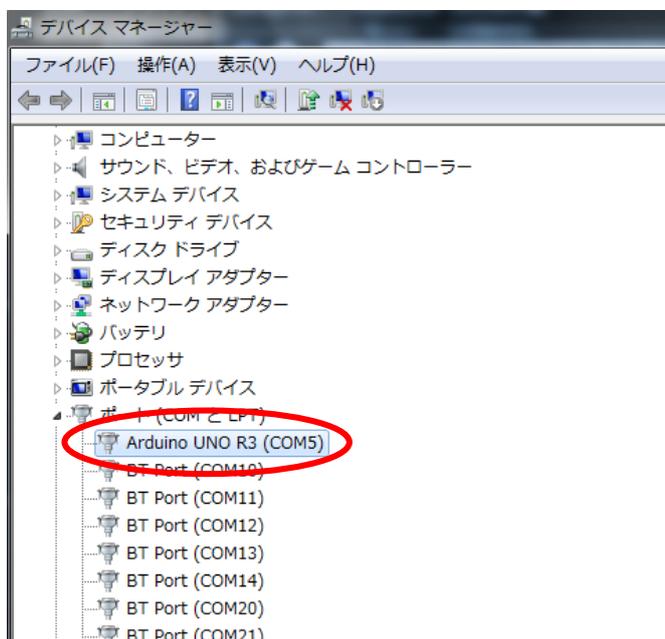
- (3) 解凍作業を行い、「Arduino.exe」が実行ファイルである。ダブルクリックする。

名前	更新日時	種類
__MACOSX	2012/05/23 4:45	ファイル フォル...
drivers	2012/05/23 4:45	ファイル フォル...
examples	2012/05/23 4:45	ファイル フォル...
hardware	2012/05/23 4:45	ファイル フォル...
java	2012/05/23 4:47	ファイル フォル...
lib	2012/05/23 4:48	ファイル フォル...
libraries	2012/05/23 4:45	ファイル フォル...
reference	2012/05/23 4:48	ファイル フォル...
tools	2012/05/23 4:48	ファイル フォル...
arduino.exe	2011/11/28 19:32	アプリケーション
cygiconv_2.dll	2011/11/28 19:31	アプリケーショ...
cygwin1.dll	2011/11/28 19:31	アプリケーショ...
libusb0.dll	2011/11/28 19:31	アプリケーショ...
revisions.txt	2011/11/28 19:31	テキスト ドキュ...
rxTxSerial.dll	2011/11/28 19:31	アプリケーショ...

- (4) USBケーブルで Arduino ボードとパソコンを接続する。



- (5) Windows7 では「コントロールパネル」→「ハードウェアとサウンド」→「デバイスマネージャー」
Windows10 では左下の「スタート」→「W」欄の「Windows システムツール」→「コントロールパネル」
あとは Windows7 と同じ。



図の場合は、Arduino ボードが「com5」に接続していることを示す。

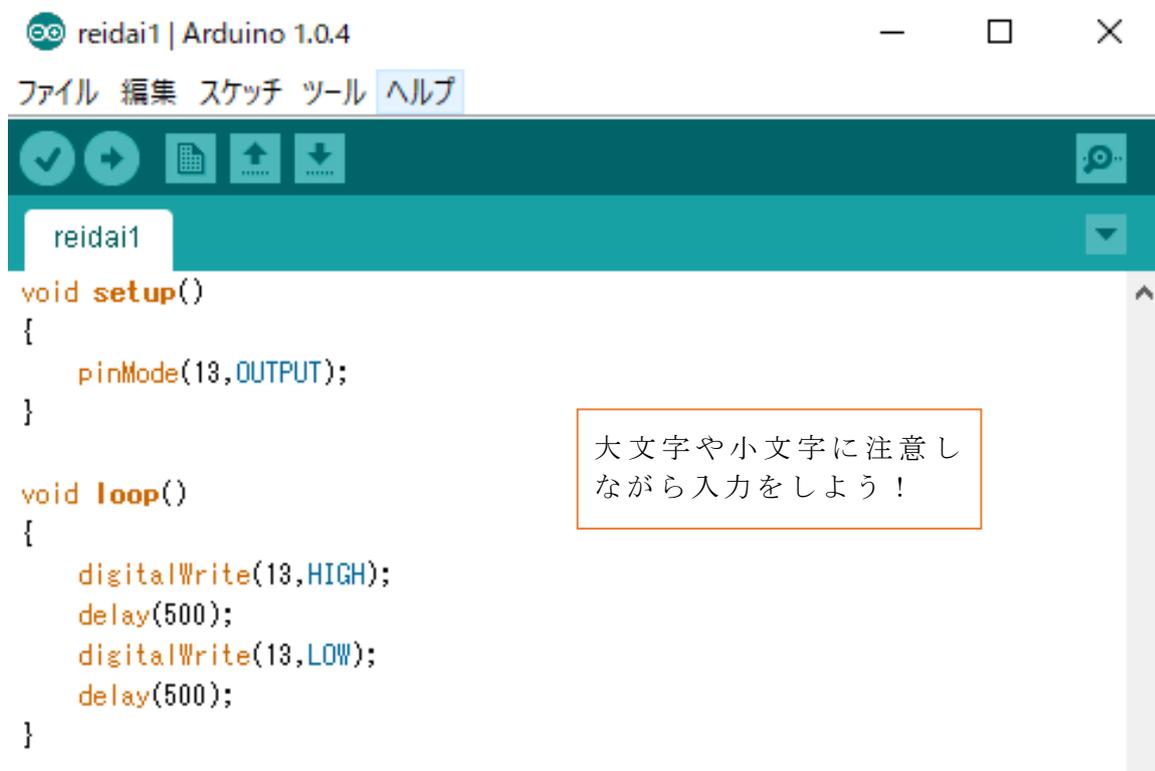
もし、認識されていない場合は、直接解凍したフォルダーの「drivers」フォルダーを指定して認識させる。

4 動かしてみよう！

例題 1 Arduino ボード上のランプ L (LED) を 0.5 秒毎に点滅するプログラムを作成しよう。

【例題 1 部品リスト】

部品	参考価格	備考
Arduino Uno	3,000 円程度	最も一般的な Arduino。互換性のあるもので 500 円を切る価格で同等の性能の互換機もある。
USB ケーブル	130 円	A-B タイプのもの

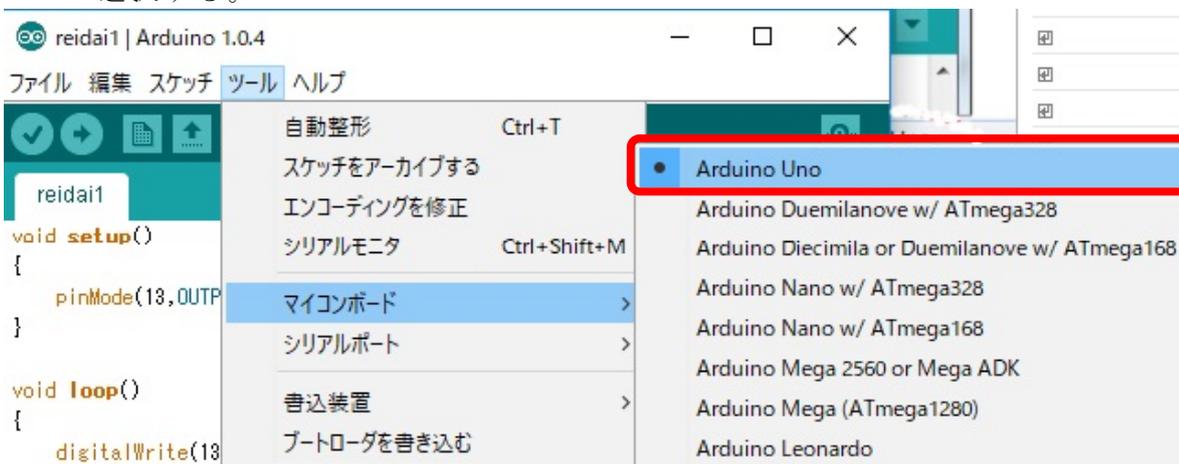


```
void setup()
{
  pinMode(13,OUTPUT);
}

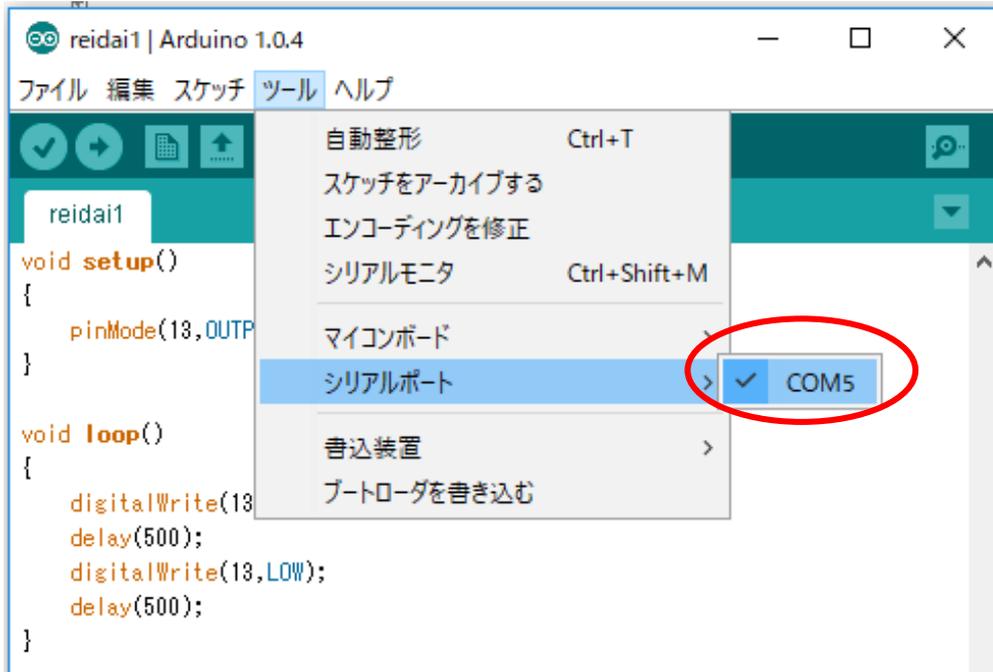
void loop()
{
  digitalWrite(13,HIGH);
  delay(500);
  digitalWrite(13,LOW);
  delay(500);
}
```

大文字や小文字に注意しながら入力しよう！

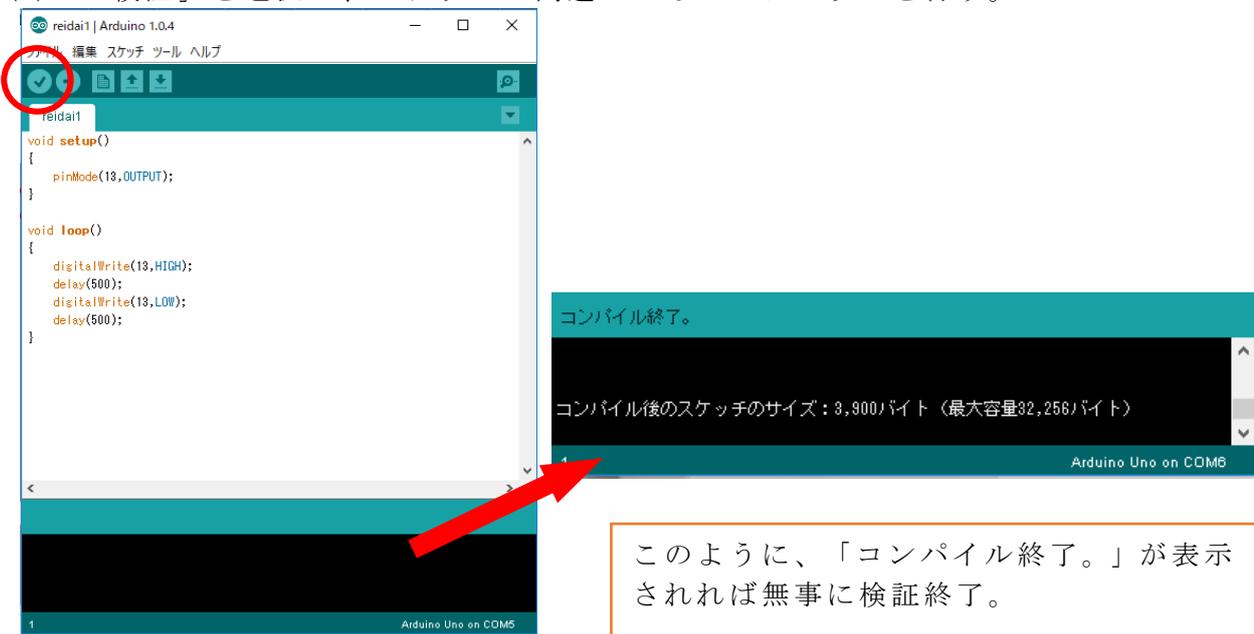
- (1) プログラムを入力したら、「ツール」→「マイコンボード」→「Arduino Uno」を選択する。



(2) 3の(5)で確認したポート番号を選択する。



(3) 「検証」を選択し、スケッチに間違いがないかチェックを行う。



(4) 「マイコンボードに書き込む」を選択する。

```
void setup()
{
  pinMode(13,OUTPUT);
}

void loop()
{
  digitalWrite(13,HIGH);
}
```

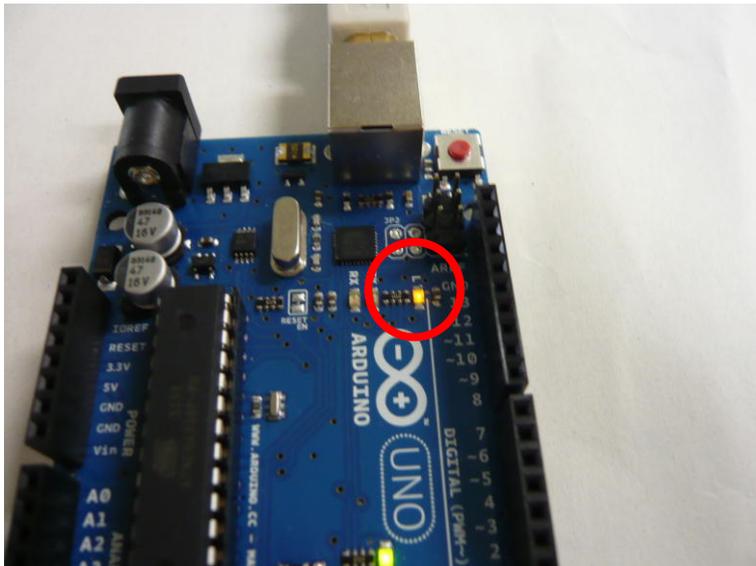
このように「ボードへの書き込みが完了しました。」表示されれば書き込み完了。

ボードへの書き込みが完了しました。

最大32256バイトのフラッシュメモリのうち、スケッチが3058バイト (9%) を使っています。
最大2048バイトのRAMのうち、グローバル変数が258バイト (12%) を使っていて、ローカル変数で1789バイト

COM4のArduino/Genuino Uno

(5) 0.5 秒間隔で点滅する。点滅すれば成功！



(6) スケッチの説明

```
void setup()
{
  pinMode(13,OUTPUT);
}
```

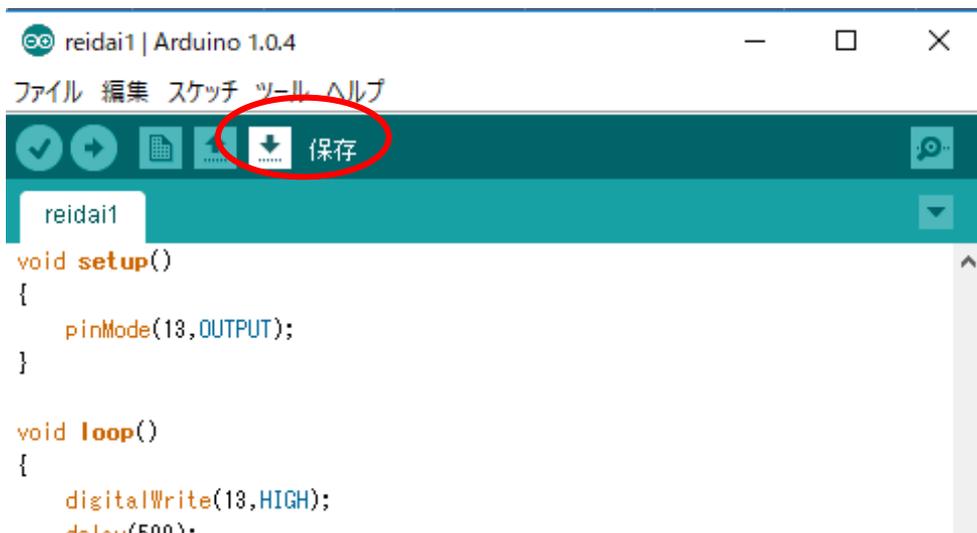
初期化の処理を書く。
13番のピンを出力として設定している。

```
void loop()
{
  digitalWrite(13,HIGH);
  delay(500);
  digitalWrite(13,LOW);
  delay(500);
}
```

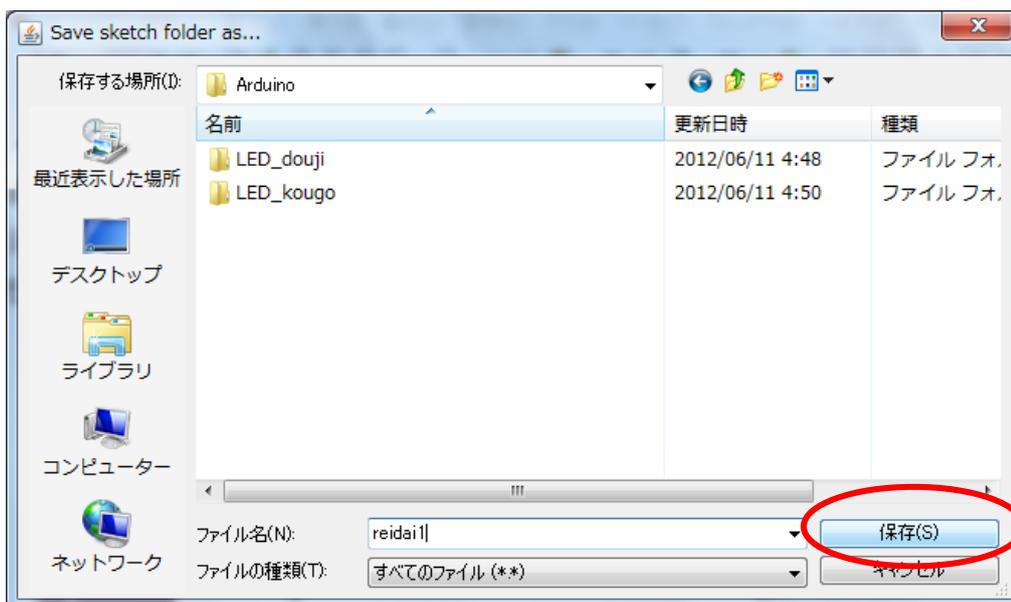
void loop() {} は、繰り返される処理を {} に書く。

13番ピンを「点灯(HIGH)」させる。
0.5秒(500ミリ秒)カウントする。
13番ピンを「消灯(LOW)」させる。
0.5秒(500ミリ秒)カウントする。

(7) スケッチの保存



(8) 適当な保存場所に保存する。



問題 1 - 1 Arduino ボード上のランプ L (LED) を任意の時間で点滅させよう。

問題 1 - 2 Arduino ボードの 13 番ピンではなく、12 番ピンで点滅させるためにはどうすればよいか考えてみよう。

例題 1 のスケッチの書き方を変更したもの

```
#define LED 13
void setup()
{
  pinMode(LED,OUTPUT);
}
void loop()
{
  digitalWrite(LED,HIGH);
  delay(500);
  digitalWrite(LED,LOW);
  delay(500);
}
```

#define を使って 13 番のピンに「LED」と名前を付ける。

「LED」をその後のスケッチで使用できる。ピンに名前があると、スケッチの動作をつかみやすく、特に長い行数の場合は便利である。

※ 1 行目 LED と 13 の間にスペースが必要！

例題 2 ブレッドボードを使用して LED を点灯させよう。

半田付けなしで回路を簡単に組み立てるのがブレッドボードであり、回路や部品の変更が容易にできるので便利である。

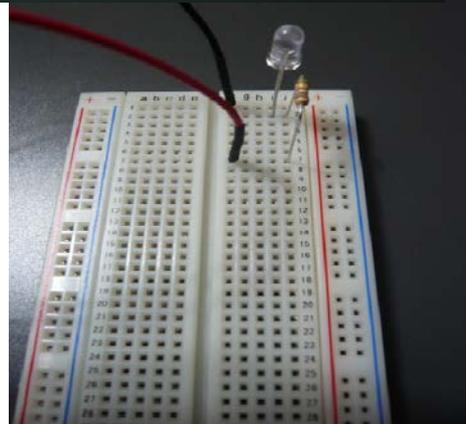
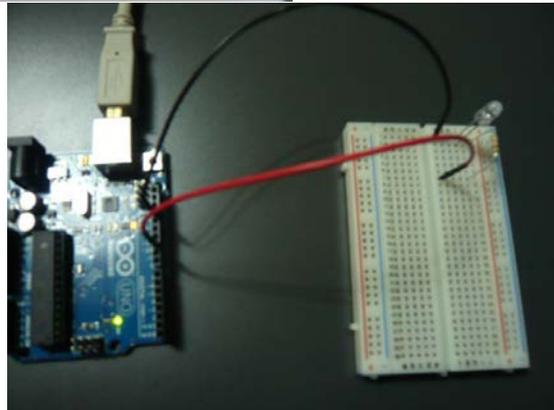
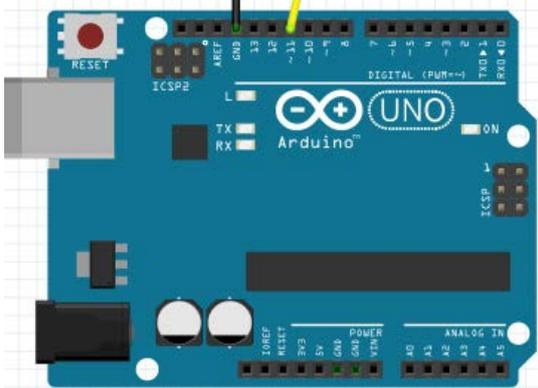
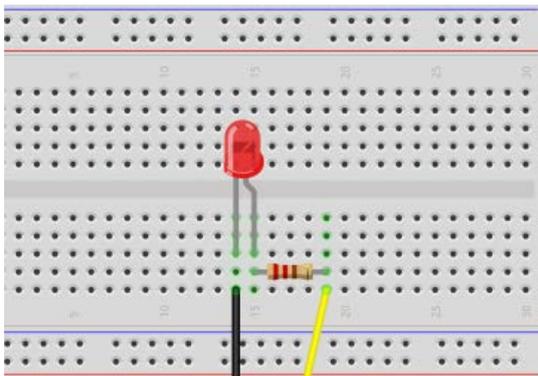
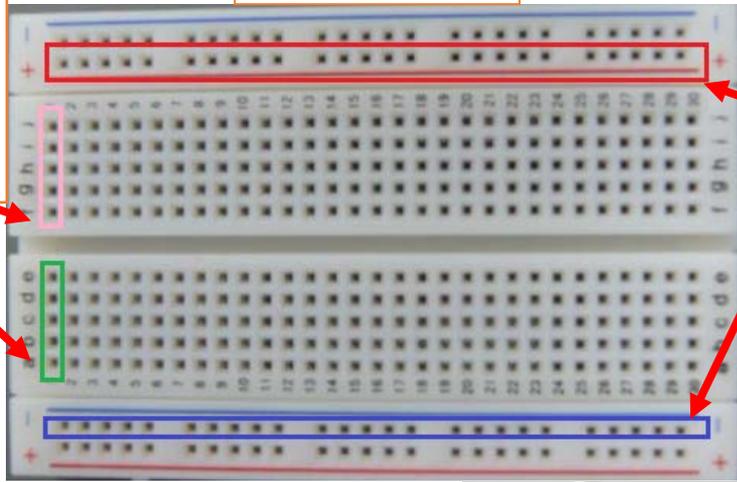
【例題 2 部品リスト】

部品	参考価格	備考
Arduino Uno	3,000 円程度	最も一般的な Arduino。 500 円を切る価格で同等の性能の互換機もある。
LED	300 円 (50 個入) 程度	LED の直径 (5 mm など) や色により価格は様々である。抵抗があらかじめ内蔵されている LED もある。
USB ケーブル	130 円程度	A-B タイプのもの。
ブレッドボード	200 円程度	下の写真よりもサイズが小さく安価 (50 円程度) なものもある。
抵抗	100 円 (100 個入) 程度	1 / 4 W 220 Ω
ジャンパーワイヤ	150 円 (20 本入) 程度	

ピンクの「列」や緑の「列」に部品を指せばはんだ付けしなくても接続される。

ブレットボード

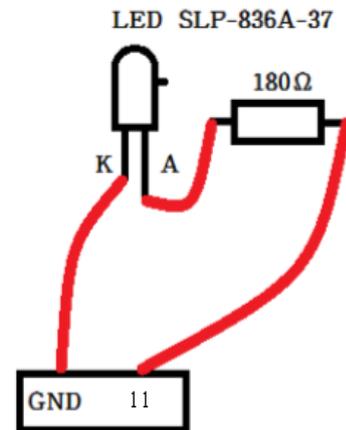
赤の「行」や青の「行」に部品をさせば、半田付けしなくても接続される。



```

#define LED 11
void setup()
{
  pinMode(LED,OUTPUT);
}
void loop()
{
  digitalWrite(LED,HIGH); //LED を HIGH
  delay(500); //0.5 秒待つ
  digitalWrite(LED,LOW); //LED を LOW
  delay(500); //0.5 秒待つ
}

```



スケッチしたら動作を確認しよう。0.5 秒毎に点滅すれば成功である。できたら点滅時間を変えてみよう。

問題 2-1 ブレットボードを使う利点について考えてみよう。

問題 2-2 使用しているブレッドボードの中央の溝や、上下にある赤い線と青い線の行の使い方について考えてみよう。

例題 3 2 個の LED を同時に 0.5 秒で点滅させよう。

任意の間隔で 2 個の LED を点灯させてみる。

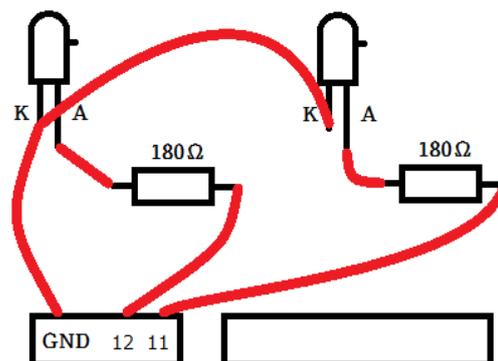
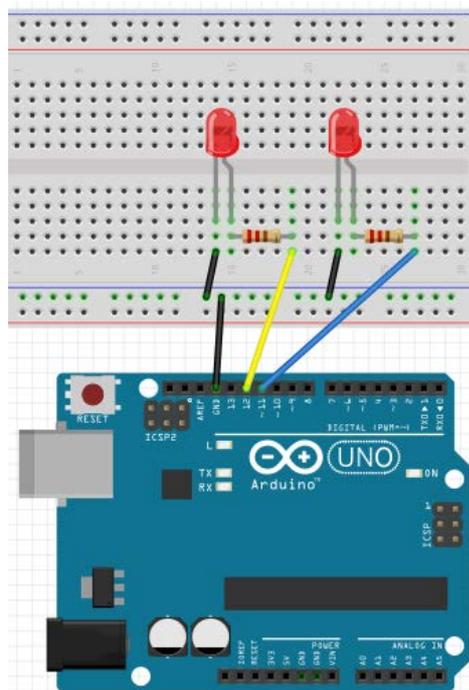
【例題 3 部品リスト】

部品	参考価格	備考
Arduino Uno	3,000 円程度	最も一般的な Arduino。 500 円を切る価格で同等の性能の互換機もある。
USB ケーブル	130 円程度	A-B タイプのもの。
ブレッドボード	200 円程度	下の写真よりもサイズが小さく安価（50 円程度）なものもある。
LED	300 円（50 個入）程度	LED の直径（5 mm など）や色により価格は様々である。抵抗があらかじめ内蔵されている LED もある。
抵抗	100 円（100 個入）程度	1 / 4 W 220 Ω
ジャンパーワイヤ	150 円（20 本入）程度	

```

//L E Dを繋ぐピンを定義
#define LED_1 11
#define LED_2 12
//初期化
void setup()
{
  //L E Dのピンを出力に
  pinMode(LED_1,OUTPUT);
  pinMode(LED_2,OUTPUT);
}
//メインループ
void loop()
{
  //LED_1 を点灯
  digitalWrite(LED_1,HIGH);
  //LED_2 を点灯
  digitalWrite(LED_2,HIGH);
  //500 ミリ秒遅延
  delay(500);
  //LED_1 を消灯
  digitalWrite(LED_1,LOW);
  //LED_2 を消灯
  digitalWrite(LED_2,LOW);
  //500 ミリ秒遅延
  delay(500);
}

```



「//」を使ってスケッチにコメントを入力することができる。コメントはスケッチのうえでは何も行わない。スケッチの中にコメントを入れることによって、スケッチを解説して誰が見てもわかりやすくするためのものである。「//」より右側がコメントとして認識される。また、複数行をコメントとして入力したい場合や、実行したくないスケッチの一部を指定する場合は、「//」を各行に入れる他にも、「/*」と「*/」で囲む方法もある。

例)

```

/* ここから
  delay(100);
  コメントと同様にスケッチ上は何もしない。ここまで
*/

```