



ロボット技術検定  
参考資料

○日時:2012年3月11日(日)13:00~16:00(受付は12時から)

○会場:学校法人岩崎学園情報科学専門学校(横浜駅きた西口徒歩1分)

横浜市神奈川区鶴屋町2-17 相鉄岩崎学園ビル

## ○持参物

受験者は以下の工具を用意して下さい。

会場では工具の貸出は行われませんので注意して下さい。

### 1) 必須なもの

- ・身分証明書(本人確認ができるもの)
- ・受験票(ロボット技術検定事務局より送付されたメール画面  
または印刷物)
- ・模型/電子工作用ニッパー
- ・ラジオペンチ
- ・+ (プラス) ドライバー#0、#1
- ・はさみ又はカッターナイフ

### 2) あると便利な工具・資料

- ・電子機器用テスター(導通、電圧計測)
- ・ワイヤーストリッパー
- ・両面テープ
- ・参考資料(当資料)

### 3) 事前申請が必要な工具

- ・プログラミング用パーソナルコンピュータ

※基本的には会場のコンピュータを使用しますが、モバイル/ノート等自己所有のコンピュータをプログラミング目的で持ち込み使用したい場合、受験申し込み時に別途申請が必要となります。

持ち込む際には、必ず今回の使用するプログラミングソフトをインストールした状態にしておいて下さい。

# ロボット 技術検定



## 4級対応 Arduino まとめシート

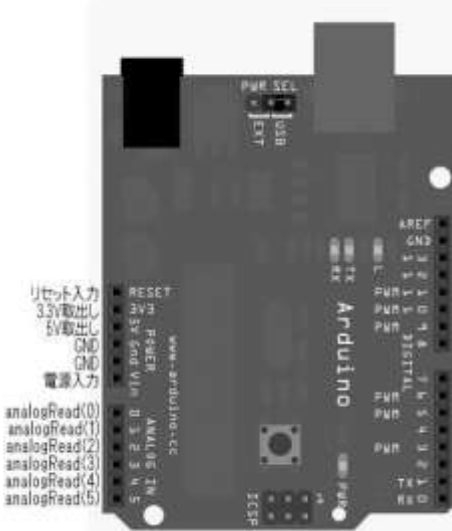
Ver.1.0

Arduino ボードを初めてマイコンに接続する  
※最新の Arduino を使用する場合



Windows の場合、自動セットアップではなく  
手動セットアップでフォルダを指定する。

### マイコンボードで使用するI/O命令



## Arduino Uno

アナログ基準電圧入力  
GND  
digitalRead(13)/digitalWrite(13)  
digitalRead(12)/digitalWrite(12)  
digitalRead(11)/digitalWrite(11)/analogWrite(11)  
digitalRead(10)/digitalWrite(10)/analogWrite(10)  
digitalRead(9)/digitalWrite(9)/analogWrite(9)  
digitalRead(8)/digitalWrite(8)  
digitalRead(7)/digitalWrite(7)  
digitalRead(6)/digitalWrite(6)/analogWrite(6)  
digitalRead(5)/digitalWrite(5)/analogWrite(5)  
digitalRead(4)/digitalWrite(4)  
digitalRead(3)/digitalWrite(3)/analogWrite(3)  
digitalRead(2)/digitalWrite(2)  
digitalRead(1)/digitalWrite(1)  
digitalRead(0)/digitalWrite(0)

digitalRead (ピン番号)  
digitalWrite (ピン番号,HIGH 又は LOW)  
analogRead (ピン番号)  
analogWrite (ピン番号,0~255)

### 基本的なスケッチ(プログラム)の形

```

/*****/
int 変数または定数設定
void setup(){ 事前設定項目
}
void loop(){ 繰り返し処理部分
}
/*****/
    
```

### 制御文の例

```

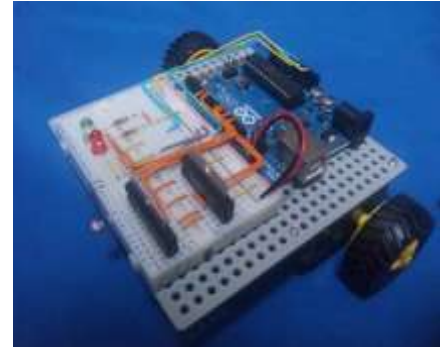
/*****/
if (analogRead(0)>=50) {
    // 動作させたいプログラムを記述。
}
/*****/
    
```

```

/*****/
int C
switch (C) {
    case 1:
        // 変数 C「1」の時
        break;
    case 2:
        // 変数 C「2」の時
        break;
    case 3:
        // 変数 C「3」の時
        break;
    default:
        // どの case にも一致しない
}
/*****/
    
```

### 応用スケッチ(プログラム)事例

光センサで床面を検出し、往復動作を繰り返すロボットカー



```

/*****/
int LED=13;
int Ct=0;
void setup() {
    pinMode(LED,OUTPUT);
}
void loop() {
    if(analogRead(0)>750) {
        if(Ct==0) // カウンタ変数1回目 {
            analogWrite(5,255);
            analogWrite(6,255);
            analogWrite(9,255);
            analogWrite(10,255);
            delay(500);
            analogWrite(5,100);
            analogWrite(6,0);
            analogWrite(9,0);
            analogWrite(10,100);
            delay(2000);
            Ct = 1; // 次は2回目
        }
        else if(Ct==1) // カウンタ変数2回目 {
            analogWrite(5,255);
            analogWrite(6,255);
            analogWrite(9,255);
            analogWrite(10,255);
            delay(500);
            analogWrite(5,0);
            analogWrite(6,100);
            analogWrite(9,100);
            analogWrite(10,0);
            delay(3000);
            Ct = 0; // カウンタ変数1回目に
        }
        else {
            analogWrite(5,0);
            analogWrite(6,100);
            analogWrite(9,0);
            analogWrite(10,100);
            delay(500);
        }
    }
}
/*****/
    
```

### モータドライバICの使い方

モータドライバIC TA7291P

- 東芝製モータドライバIC
- (正・逆切り替え可能ドライバ)
- VS (モーター用電源): 0~20V
- VCC (ロジック電源): 4.5V~20V
- 出力電流: 1A (ピーク2A)
- 10ピン
- 熱渡断面部・出力端子プロテクタ面内蔵
- 入力ヒステリシス回路、スタンバイ回路内蔵

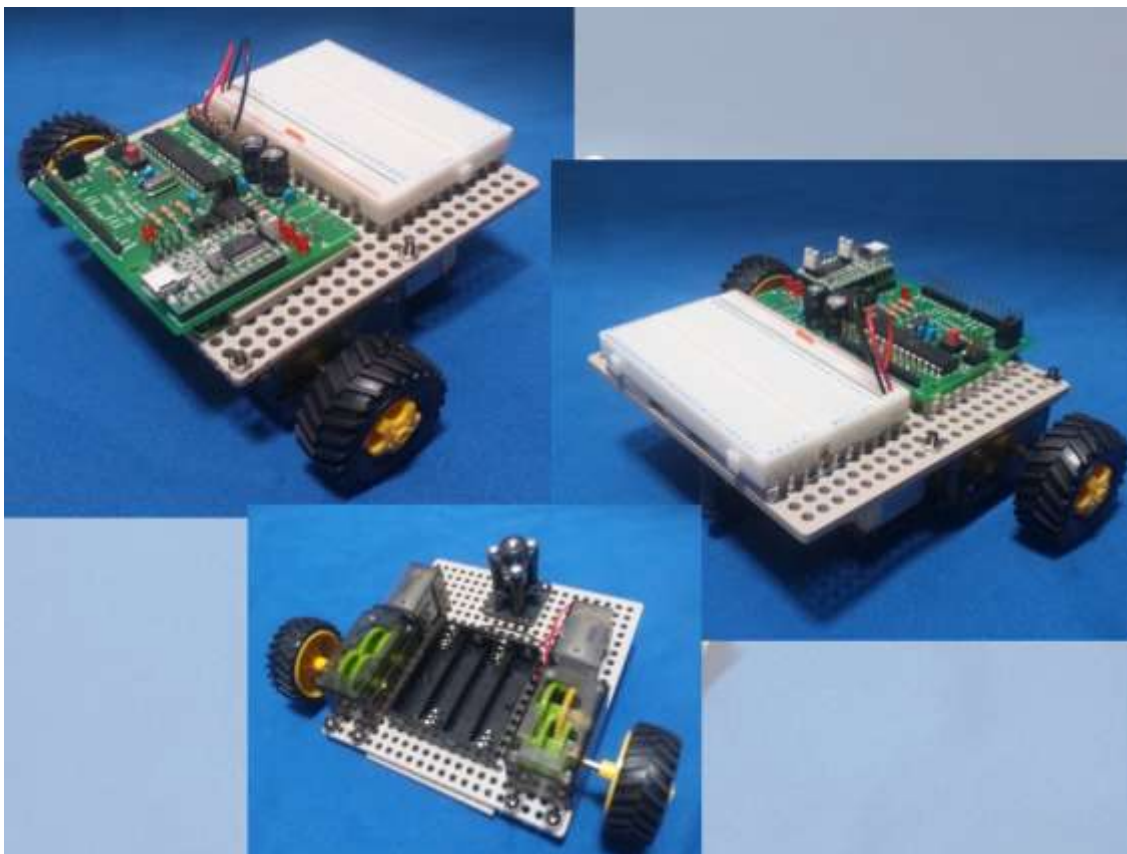
# 目次

1. システム概要
  - 1.1 車台サンプル
  - 1.2 使用キット構成
2. 各種部品使用方法
  - 2.1 光センサ(CdS)使用方法
  - 2.2 モータドライブIC使用方法
  - 2.3 ブレッドボード使用方法
3. 回路図
  - 3.1 参考回路図
4. 試験用マイコン使用方法
  - 4.1 Arduino の入手方法
  - 4.2 Arduino の起動
  - 4.3 ArduinoUNO マイコンボードを初めてPCに接続する
  - 4.4 基本設定
  - 4.5 プログラムの確認と転送
5. プログラムサンプル
  - 5.1 CdS センサを判別してLEDを点灯するプログラム
  - 5.2 2ビット制御でモータドライブIC×1個を制御するプログラム
  - 5.3 アナログポートに接続した光センサが暗さに反応する度に2つのLEDを交互に点灯させるプログラム
6. 試験フィールド説明

## 1. システム概要

	構成	概要
1	車台／移動部分	模型用のギアボックス＋市販パーツを組み立てて構成します。
2	制御回路部分	ブレッドボードを車台に取り付け、上部に回路を構築します。
3	制御マイコン部分	Arduino ボードを使用し、作成したプログラムを転送します。
4	動力源	車台に電池ボックスを乗せ、独立して動作できるようにします。

### 1. 1 車台サンプル



## 1. 2 使用キット構成

実技キット構成				
	名称	型番	メーカー	個数
1	自在ギアボックス	J1515	山崎教育	1
2	トラックタイヤセット	J1122	山崎教育	1
3	ナットレスプレート	N5591	山崎教育	1
4	ボールキャスター(2個セット)	J1776	山崎教育	1
5	電池ボックス(単4×4)	MP4-4	タカチ	1
6	サンハヤト ニューブレッドボードSAD-101	SAD-101	サンハヤト	1
7	ジャンプワイヤーセット	-	E-CALL ENTERPRISE	1
8	Arduino Uno R3	ARDUINO-000066	arduino	1
10	USBケーブル(Aオス-Bオス)			1
11	モータードライバ	TA7291P	東芝	1
12	M3タッピングネジ	F-0305E-B1	ウィルコ	10
13	CdSセンサ	-	Macron International Group	2
14	抵抗器(470Ω)	MF1/40C1000F MF1/40C4700F	KOA	5
15	抵抗器(1kΩ)		KOA	5
16	抵抗器(10kΩ)		KOA	5
17	絶縁型ラジアルリードタイプ積層セラミックコンデンサー0.1μ F50V 2.54mmピッチ		株式会社村田製作所	2
18	LED 3mm 弾丸型(赤)		OptoSupply	1
19	LED 3mm 弾丸型(黄緑)		ローム	1
20	FSR-41 ソケット(角ピン)2.54mmピッチ ライトアングル(1列)		廣杉計器	2

※採用メーカー及び型番は互換品を前提に変更される場合があります。

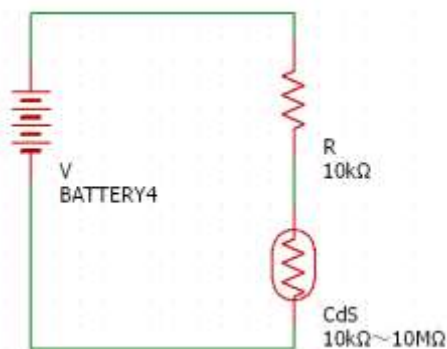
## 2. 各種部品使用方法

### 2.1 光センサ(CdS)使用方法

本試験に使用する光センサは CdS (硫化カドミウム)を用いたタイプを使用します。

このセンサは明るくなる(光が照射される)と抵抗値が少なくなり、暗くなる(光が遮断される)と抵抗値が大きくなる特性を持っています。

この性質を利用して、明るさの変化を電圧の変化として測定できる回路を作成しマイコンに接続、光スイッチとして利用します。



例えば左の回路において、CdS が明るい状態＝抵抗値が低い(この場合 10kΩ)時、回路中にある抵抗と CdS それぞれに加わる電圧は、等しく分圧される為

$$V_R = V_{CdS} = V/2$$

となります。

このCdSへの光を遮断(指でセンサ面を隠すなど)すると、抵抗値は増大する方向へ変化する為(この場合最大 10MΩ ですが、増加値 1MΩ と仮定します)

1MΩ = 1000kΩ ですので、

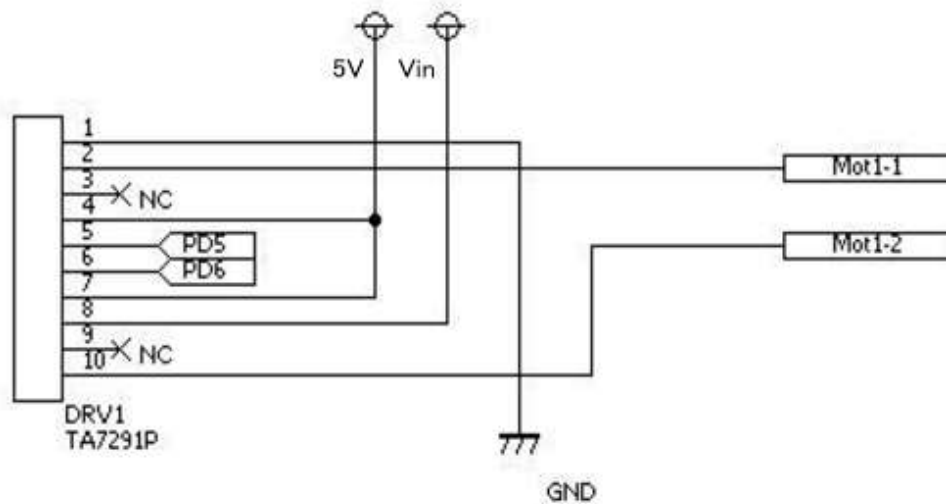
$$V_R = V \times 10k\Omega / (10k\Omega + 1000k\Omega) \doteq 0.01 \times V$$

$$V_{CdS} = V \times 1000k\Omega / (10k\Omega + 1000k\Omega) \doteq 0.99V$$

と変化します。この電圧の変化をマイコンのA/Dポートに入力して測定することで、明るさを測定するセンサとして利用する事が出来ます。

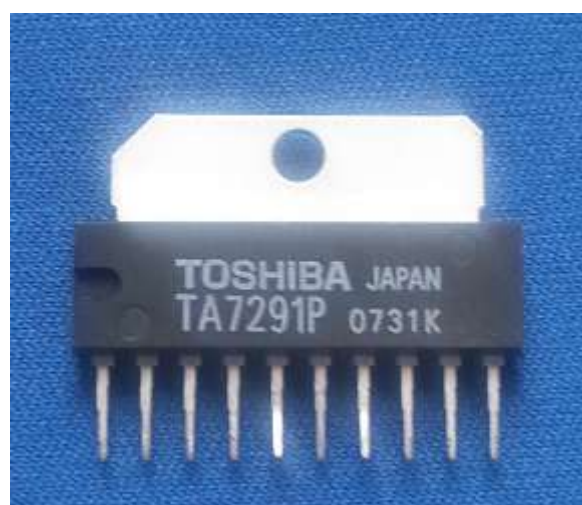
## 2.2 モータドライブIC使用方法

ロボットのモータ駆動部には、専用の「モータドライブ I C」を利用して配線します。一個のモータに対する回路図は下図の通りとなります。



- Motor\*\* : モータを接続します。
- 5V : マイコンからの 5 V を入力します (ロジック電源)
- Vin : 電池ボックスからの電源を入力します (駆動電源)
- PD\* : マイコンのデジタル出力信号を入力します。

図は東芝製「TA-7291P」について説明しています。

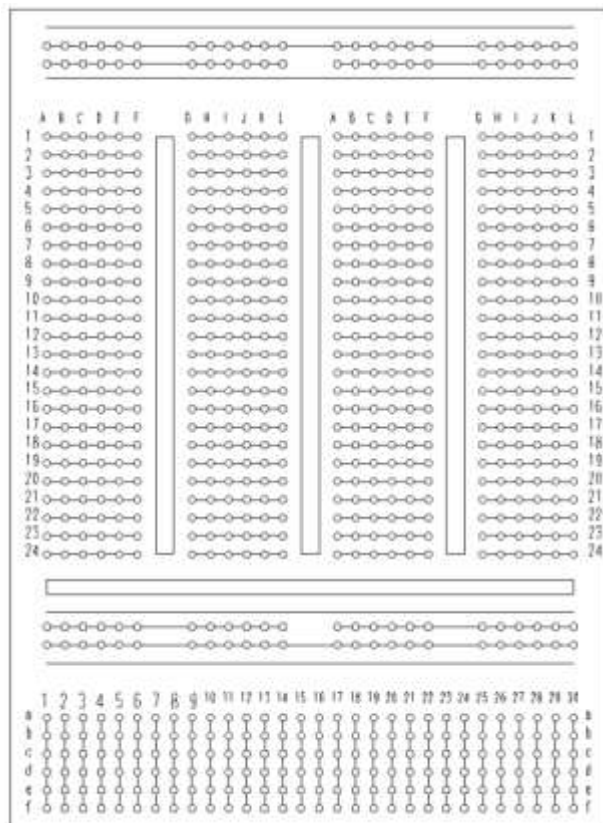


放熱板の左上が切り欠かれ、左端に凹マークがモールドされています。こちら側から 1 番~としてピンを数えます。



## 2.3 ブレッドボード使用方法

ブレッドボードは、下図に示す方向  
(○同士が線で繋がっている方向)  
に導通しています。



横方向に導通しています

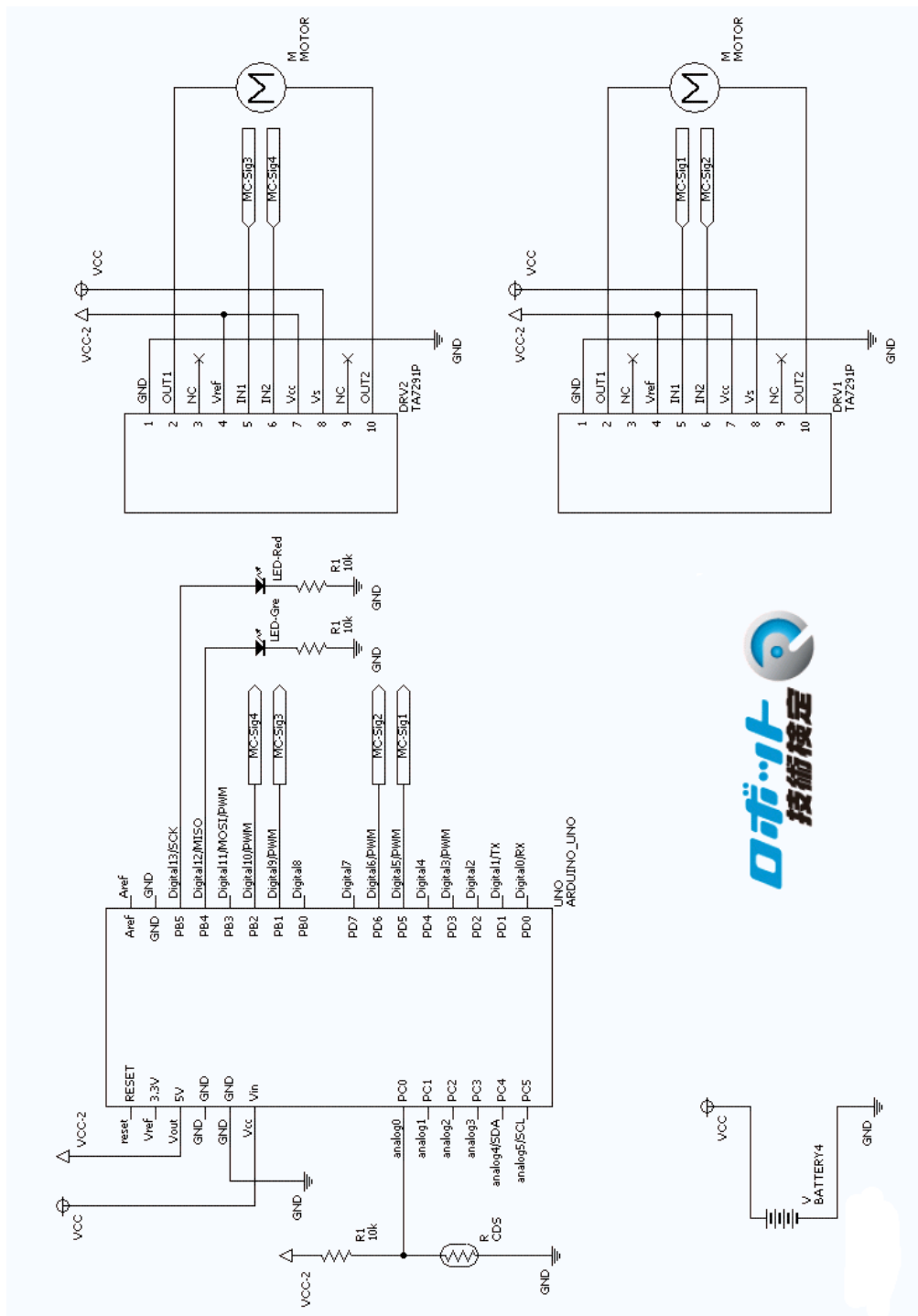
縦方向に導通しています。

キットに同梱されているブレッドボードは、背面が強力両面テープになっている場合があります。組立時は裏紙を全部剥がしてしまわず、カッターナイフ等を利用して必要な部分のみを剥がすようにしましょう。また、試験に使用されるブレッドボードは図示のものとは異なる場合があります。

### 3. 回路図

#### 3.1 参考回路図

図示の回路図からブレッドボード上に回路を構成できる技術水準が必要です。



## 4. 試験用マイコン使用方法

今回の試験では制御用マイコンシステムに「Arduino」を使用します。  
ツールの使用方法及びプログラミングの記述例は後述のサンプルプログラム及び別途配布される資料を参考にして下さい。

### 4. 1 Arduino の入手方法

公式ホームページより、マイコンのプログラミングに必要なツールは無償でダウンロードすることができます。

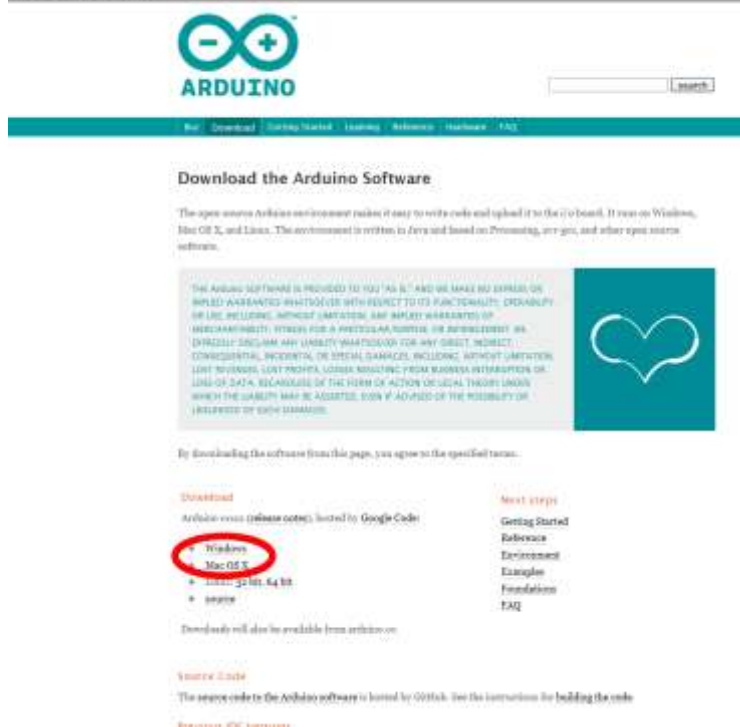
公式ホームページURL : <http://arduino.cc/>

ホームページが開いたら、  
右図赤○にある  
「Download」のリンクをクリックします。



リンク先のページを下にスクロールすると、Download のメニューがあります。使用したいOSの名称がそのままリンク先になっていますので、入手したいOSの種類をクリックします。

クリックする事で圧縮ファイルがダウンロードされますので、任意の場所に解凍してツールを利用可能な状態にします。

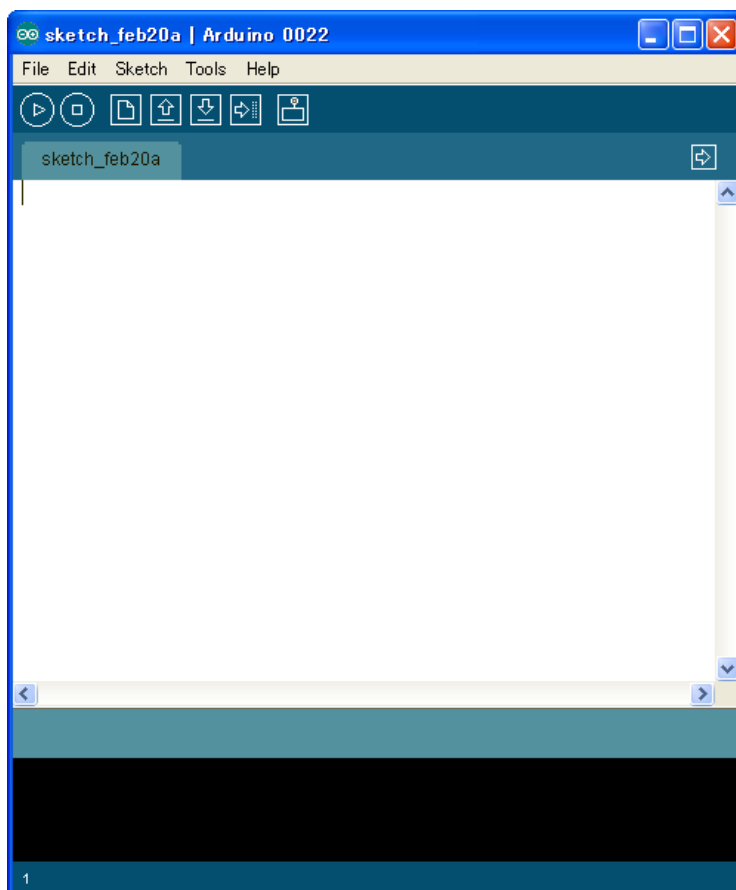


#### 4. 2 Arduino の起動

試験時は別途指示されたフォルダ内にある「Arduino.exe」ファイルをダブルクリックして起動します。



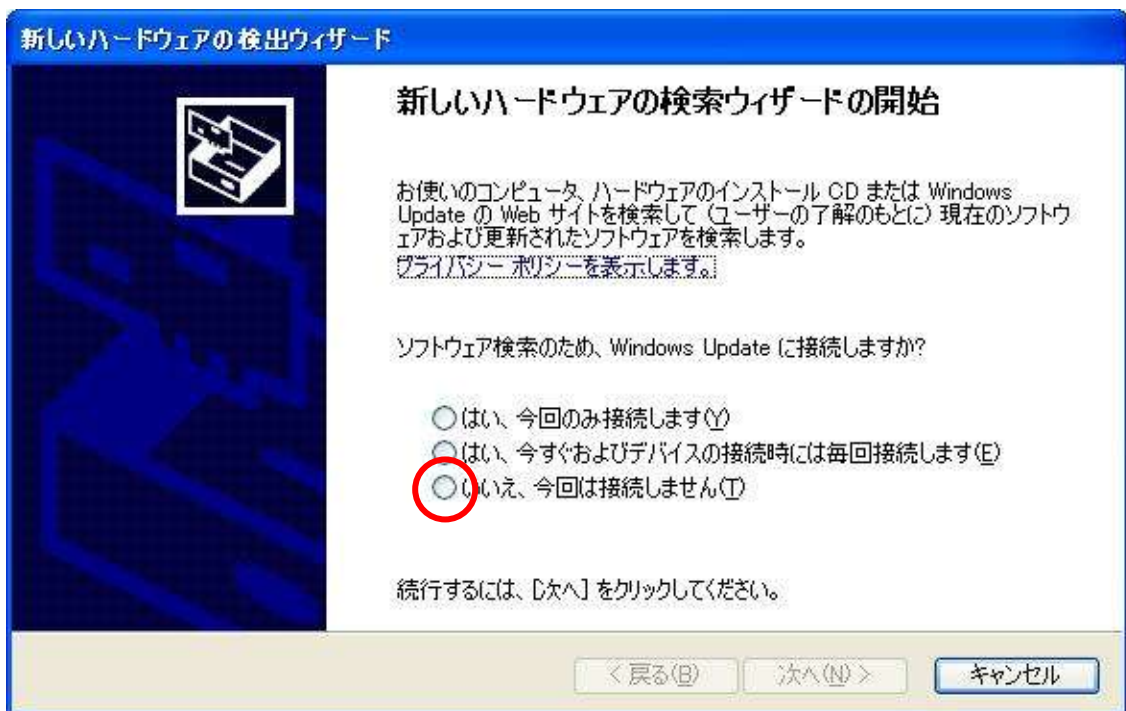
スケッチエディタが起動しますので、最初に基本設定を行います。



#### 4. 3 ArduinoUNO マイコンボードを初めてPCに接続する

プログラムを転送する為に、マイコンボードとPCをUSBケーブルで接続します。

初めてマイコンボードが接続されるPCからは「不明なデバイス」と表示されドライバのインストールウィザードが起動します。今後マイコンボードを利用する事が出来る様に以下の手順でドライバのセットアップを行って下さい。

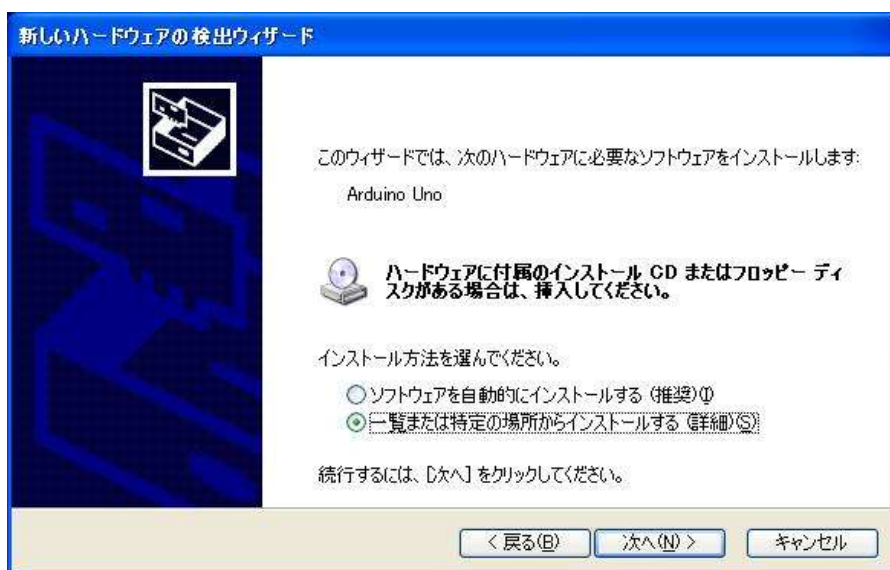


「新しいハードウェアの検索ウィザードの開始」が起動します。

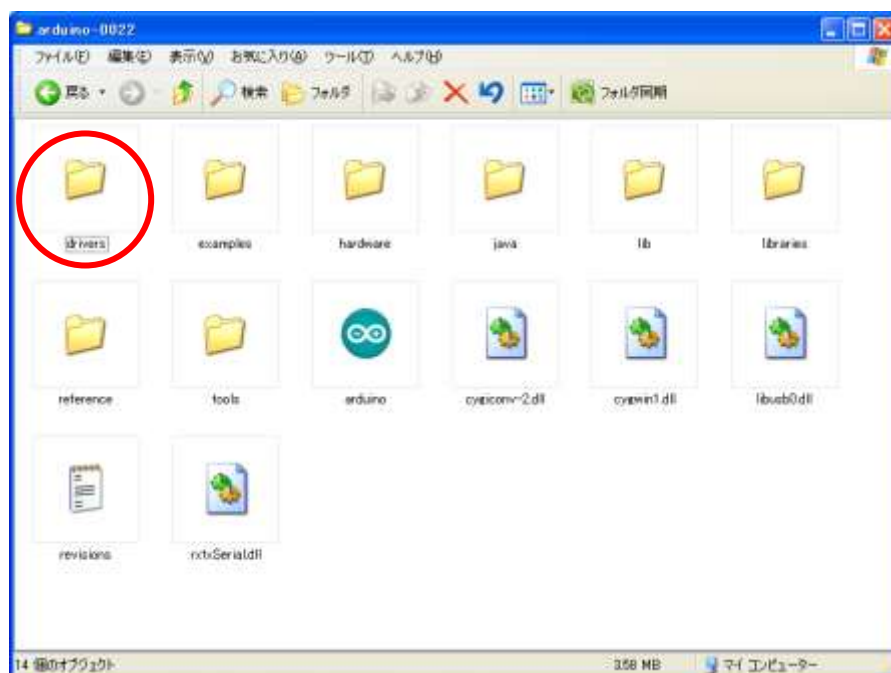
Windows Update に接続する必要はありませんので、一番下の選択肢

「いいえ、今回は接続しません」

を選んで次へ進んで下さい。



Arduino のドライバは、自動検索で検出する事が出来ません。先にダウンロードしたプログラミングツールのフォルダ内にドライバが用意されていますので、「一覧または特定の場所からインストールする」を選んで次に進んで下さい。



ドライバのある場所を指定する様に指示が表示されますので、ダウンロードした Arduino フォルダの中にある「drivers」フォルダを選択します。

フォルダ内にあるドライバファイルが読み込まれ、マイコンボードとの通信が可能になります。

#### 4. 4 基本設定

最初に、使用するマイコンの種類を決定します。

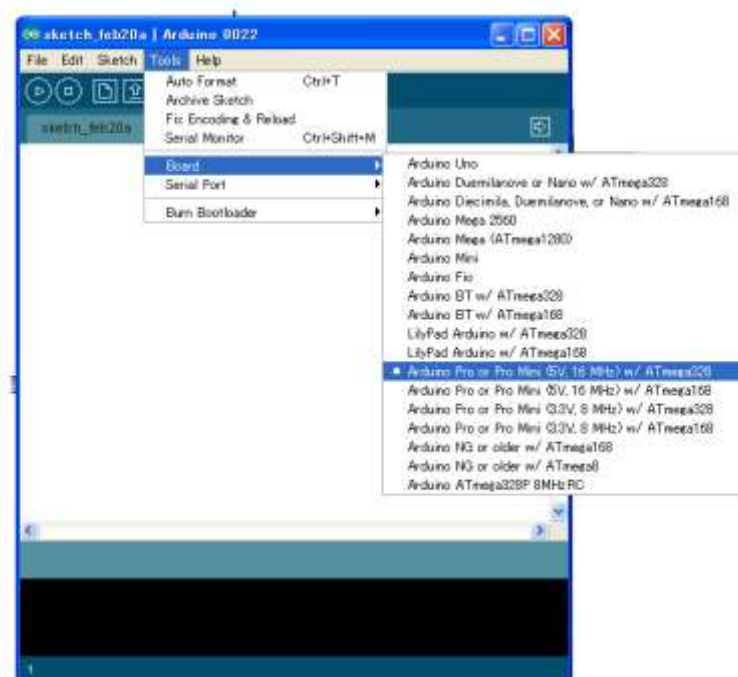
Arduino は多くのバリエーションを持ち、適切な設定を行わないとプログラムの転送を行う事が出来ません。

メニューバーから

「Tool」 - 「Board」の順で

クリックし、図示のマイコンタイプを選択します。

使用されるボードによって選択するモデルが異なりますので注意しましょう。



選択例：「Arduino Pro or Pro Mini (5V,16MHz) w/ATMega328」

次に、マイコンが接続されるUSBポートを設定します。

「Tool」 - 「Serial Port」の順にクリックし、対象のCOMポートを選びます。





#### 4. 5 プログラムの確認と転送

記述したプログラムに誤記が無いのみを確認する場合は、エディタの左側にある三角アイコン「Verify」を押します。

記述に問題が無ければ、右図の様に下段のメッセージ表示部に

「Done Compiling」と表示されます。



```
SampleSketch_01 | Arduino 0022
File Edit Sketch Tools Help
Verify
SampleSketch_01
/* ロボット技能検定プレテストサンプルプログラム01 */
/* サンプル1「LEDを光らせる*/

int led=13; //動作確認用LEDの接続ポート（13）に「led」ラベルを宣言
|
void setup() //1/0初期設定
{
  pinMode(led,OUTPUT); //動作確認用LEDのポートを出力に設定
}

void loop() //メイン処理部（ループで繰り返す）
{
  digitalWrite(led,HIGH); //LEDを消灯
  if(analogRead(0)>200) //アナログポート0に接続された光センサからの読み取り値
  {
    digitalWrite(led,LOW); //led点灯
    delay(1000); //1000mSec待機する。単位は1/1000秒
  }
}

Done compiling.

Binary sketch size: 1096 bytes (of a 30720 byte maximum)
5
```



```
SampleSketch_01 | Arduino 0022
File Edit Sketch Tools Help
SampleSketch_01
/* ロボット技能検定プレテストサンプルプログラム01 */
/* サンプル1「LEDを光らせる*/

int led=13; //動作確認用LEDの接続ポート（13）に「led」ラベルを宣言

void setup() //1/0初期設定
{
  pinMode(led,OUTPUT); //動作確認用LEDのポートを出力に設定
}

void loop() //メイン処理部（ループで繰り返す）
{
  digitalWrite(led,HIGH); //LEDを消灯
  if(analogRead(0)>200) //アナログポート0に接続された光センサからの読み取り値
  {
    digitalWrite(led,LOW); //led点灯
    delay(1000); //1000mSec待機する。単位は1/1000秒
  }
}

digitalWrite was not declared in this scope

SampleSketch_01.cpp: In function 'void loop()':
SampleSketch_01:12: error: 'digitalWrite' was not declared in this scope
12
```

記述間違い等のエラーがある場合、左図の様なメッセージと共にエラー部分の行が色つきで示されます。



マイコンへの転送時は「Upload」ボタンを押します。



マイコンが接続されていない、COMポートの指定が間違っている等の場合は先ほどと同様にエラーメッセージが表示されます。



転送に成功すると「Done Uploading」のメッセージが表示されます。



## 5. プログラムサンプル

### 5. 1 CdS センサを判別してLEDを点灯するプログラム

```
/* ロボット技能検定プレテストサンプルプログラム01 */
/* サンプル1「LEDを光らせる*/

int led=13;    //動作確認用LEDの接続ポート (13) に「led」ラベルを宣言

void setup()   //I/O初期設定
{
  pinMode(led,OUTPUT); //動作確認用LEDのポートを出力に設定
}

void loop()   //メイン処理部 (ループで繰り返す)
{
  digitalWrite(led,HIGH); //LEDを消灯
  if(analogRead(0)>1000) //アナログポート0に接続された光センサからの読み取り値が1000以上になったら動作開始
  {
    digitalWrite(led,LOW); //led点灯
    delay(1000); //1000mSec待機する。単位は1/1000秒
  }
}
```

Cds センサから入力されるアナログ値を判別し、条件が整ったら (指で陰になったら) LEDを点灯する様出力します。

## 5. 2 2ビット制御でモータドライバIC×1個を制御するプログラム

```
/* ロボット技能検定プレテストサンプルプログラム */
/* サンプル「モータドライバでモーターを制御する」*/

/* 2Bitの出力制御で、モータの回転方向を制御する */
/* 使用回路はプレテスト用配布資料に基づくものとする */

int led=13; //動作確認用LEDの接続ポート（13）に「led」ラベルを宣言
int Motor00=9; //モータ制御ビット0をデジタル出力9番ピンに設定
int Motor01=10; //モータ制御ビット1をデジタル出力10番ピンに設定

void setup() //I/O初期設定
{
  pinMode(led,OUTPUT); //動作確認用LEDのポートを出力に設定
  pinMode(Motor00,OUTPUT); //モータ制御ビットを出力に設定
  pinMode(Motor01,OUTPUT); //モータ制御ビットを出力に設定

  digitalWrite(Motor00,HIGH);
  digitalWrite(Motor01,HIGH); //モータを停止状態にしておく
}

void loop() //メイン処理部（ループで繰り返す）
{
  digitalWrite(led,HIGH); //LEDを消灯
  if(analogRead(0)>1000) //アナログポート0に接続された光センサからの読み取り値が1000以上になったら動作開始
  {
    digitalWrite(led,LOW); //led点灯
    digitalWrite(Motor00,HIGH);
    digitalWrite(Motor01,LOW); //モータを正転（CW）させる
    delay(3000); //3000mSec回転を続ける。

    digitalWrite(Motor00,HIGH);
    digitalWrite(Motor01,HIGH); //モータを停止状態にする
    delay(1000);

    digitalWrite(Motor00,LOW);
    digitalWrite(Motor01,HIGH); //モータを逆転（CCW）させる
    delay(3000);

    digitalWrite(Motor00,HIGH);
    digitalWrite(Motor01,HIGH); //モータを停止状態にする
    delay(1000);

    digitalWrite(led,HIGH); //動作終了を知らせる為にled消灯
  }
}
```

試験時はモータドライバの数が変わります。  
また、動作パターンも変わるので注意しましょう。

5. 3 アナログポートに接続した光センサが暗さに反応する度に2つのLEDを交互に点灯させるプログラム

```
/*~~~~~*/  
// 参考プログラム  
  
int LEDR=13;    // LED-R (赤) を13番ピンに接続  
int LEDG=12;    // LED-G (緑) を12番ピンに接続  
int Ct=0;       // 検出回数カウント用変数  
  
void setup()    //I/O 初期設定  
{  
  pinMode(LEDR,OUTPUT); //LED (赤) のポートを出力に設定  
  pinMode(LEDG,OUTPUT); //LED (緑) のポートを出力に設定  
}  
void loop()     //メイン処理部 (ループで繰り返す)  
{  
  if(analogRead(0)>750)  
    // アナログポート0番の値が750より大きい  
  {  
    if(Ct==0) // カウンタ変数の値が0 (=1回目)  
    {  
      digitalWrite(LEDR,LOW); //LED (赤) 点灯  
      digitalWrite(LEDG,HIGH); //LED (緑) 消灯  
      delay(1000);  
      Ct = 1; // カウンタ変数の値を「1」へ (次は2回目)  
    }  
    else if(Ct==1) // カウンタ変数の値が1 (=2回目)  
    {  
      digitalWrite(LEDR,HIGH); //LED (赤) 消灯  
      digitalWrite(LEDG,LOW); //LED (緑) 点灯  
      delay(1000);  
      Ct = 0; // カウンタ変数の値を「0」へ (1回目に戻す)  
    }  
  }  
}  
/*~~~~~*/
```

## 6. 試験フィールド説明

試験フィールドは、紙の上に印刷されたラインを利用して動作判定を行います。  
当日配布される問題に合わせたフィールドが用意されます。

<フィールドの例>

寸法などの記述は当日行われません。

