

Arduino によるコンピュータプログラミング A を教わった後の ハードウェア入門 2

土肥 紳一*1

Email: dohi@chiba.dendai.ac.jp

*1: 東京電機大学 情報環境学部 情報環境学科

◎Key Words Arduino, モチベーション, プログラミング入門教育, ハードウェア入門

1. はじめに

エクステンションプログラムで開講している「コンピュータプログラミング A を教わった後のハードウェア入門」は、今年で 3 回目の開催となった。昨年と同様に 3 日間の集中講義で、1 月 25 日から 1 月 27 日にかけて開講した。時間帯は、14:30~16:20(途中 10 分の休憩)である。受講者は「コンピュータプログラミング A」の授業で Java 言語を学んだ知識を活用しながら、変数、繰返し、メソッド等の概念を、ハードウェア (Arduino) を通じて理解を深めることを狙っている[1]。3 回目の開催となるとマンネリ化が生じる。これを避けるために、今年は新たなハードウェアとしてサーボモータを取り入れた。サーボモータはパルス幅で回転角度を制御でき、Arduino には角度を与えると適切なパルス幅を出力する Servo ライブラリに write 関数が用意されている。これを活用することによって「コンピュータプログラミング A」を教わった知識の延長線上で、サーボモータを容易に制御できる。本論文では、過去の開催と比較しながら、3 回目の実施結果を述べる。

2. Arduino について

Arduino はマイクロコンピュータである。「Arduino をはじめようキット」が販売されており、本テーマを実施するために必要な部品が一通り揃っている[2]。教材提示装置上の Arduino、テキスト、新たに導入したサーボモータ等の様子を図 1 に示す。このキットの価格は 4 千円程度、サーボモータは 500 円程度である。学生諸君のお小遣いで十分手に入る価格である。



図 1 教材提示装置の様子

筆者は、東京電機大学生協同組合から購入した。情報環境学部は学部開設当初から BYOD(Bring Your Own Device)を進めている。今年利用した Arduino の統合環境のバージョンは 1.6.7 で、受講者のノート PC にインストールした。Arduino は Processing に似ており、さらに Processing は Java に似ていることから、「コンピュータプログラミング A」を教わった後の受講者にとって馴染みやすい。テキストは、「Arduino をはじめよう 第 2 版」を使っており、本テーマを実施する上で適切な内容である[3]。テキストは第 5 章までの内容と、これに関連する付録と Arduino 公式リファレンスを利用している。

3. 実施スケジュールと受講者の様子について

3.1 スケジュール

今年開講した「コンピュータプログラミング A を教わった後のハードウェア入門」の実施内容とスケジュールは、表 1 に示す。昨年と比較して大きく変わった内容は、3 日目の RGBLED アレイに置き換えて、サーボモータを取り入れたことである[4]。

表 1 スケジュール

日程	内容
1 月 25 日	統合環境 Arduino のインストール, LED の表示
1 月 26 日	スイッチを使った LED の表示
1 月 27 日	光センサーを使った LED の表示, サーボモータの回転(昨年は RGBLED アレイの表示)

3.2 ハードウェアならではの体験

「コンピュータプログラミング A」の授業では体験できない、ハードウェア固有の興味深い問題が生じる。一例であるが、テキストではタクトスイッチを 1 回押すと LED が点灯し、もう 1 回押すと消灯する例が紹介されている。単純な例であるが、「コンピュータプログラミング A」では体験できない、ハードウェア固有の問題、すなわちバウンスが隠れている。バウンスはタクトスイッチを押した瞬間に、内部の接点が短時間に振動し、スイッチの ON と OFF を繰り返す現象である。この現象を避けるために、delay 関数を使って 50ms 程度の遅延を入れる。受講者は、ハードウェア固有の問題をソフトウェアで補えることに気付く。

スイッチは単に ON と OFF を行うだけであるが、バウンス対策を施したプログラムは、さらにタクト

スイッチを押している時間の長さによって、ふるまいを変更でき、そのふるまいはソフトウェアによって実現できることを体験する。テキストでは、「1つの回路、千のふるまい」と述べられてプログラムが紹介されている。「コンピュータプログラミング A」を教わった後に体験する例として、最適な内容である。

4. サーボモータの導入

マンネリ化を避けるために、今年は新たなハードウェアとしてサーボモータ(SG-90)を導入した。サーボモータの様子を図 2 に示す。



<http://akizukidenshi.com/img/goods/L/M-08761.jpg> より引用

図 2 サーボモータの様子

サーボモータからは、赤色、茶色、橙色の 3 本のケーブルが出ており、各ケーブルの働きを図 3 に示す。茶色が GND、赤色が Vcc、橙色が PWM(Pulse Width Modulation)になる。パルス幅は 20ms の周期の中で、0.5ms の場合が 0 度、2.4ms の場合が 180 度の回転角度に対応しており、0.5ms~2.4ms の間でパルス幅を変更することによって、0 度から 180 度までの回転角度を制御できる。プログラムでは Arduino の Servo ライブラリを利用することによって、角度を与えると必要なパルス幅を生成してくれる write 関数が用意されており、これを利用することによって、サーボモータを容易に制御できる。

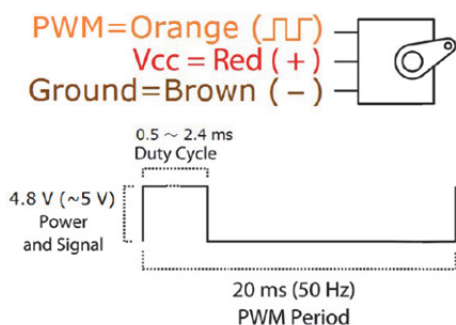


図 3 サーボモータの回路図

http://akizukidenshi.com/download/ds/towerpro/SG90_a.pdf より抜粋

Arduino とサーボモータの配線は、茶色が Arduino の GND に、赤色が 5V に、橙色は 9 番ポートに接続した。この様子を図 4 に示す。使用した SG-90 は消費電流が少ないため、ノート PC の USB で供給される電源で十分動作させることができた。「コンピュータプログラミング A」では、プログラム分割によって他のクラスにあるメソッドを利用することを教わっている。受講者の様子を観察している限り、Servo ライブラリの

include は、Java の import と読み替えることによって、「コンピュータプログラミング A」の延長線上で理解でき、良い復習になっている。

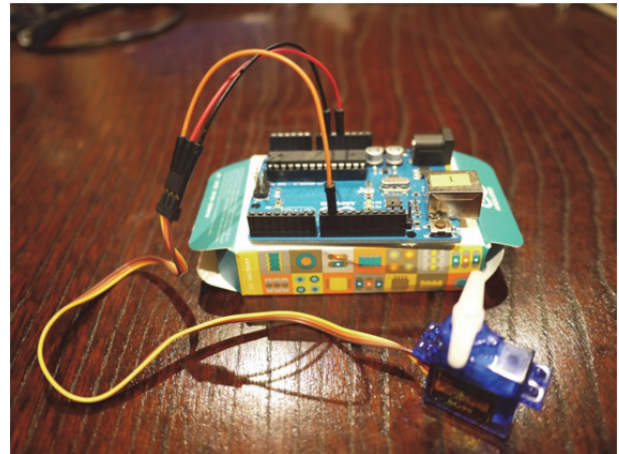


図 4 Arduino とサーボモータの接続の様子

サーボモータを 0 度から 180 度まで回転し、逆に 180 度から 0 度まで回転するプログラムを図 5 に示す。①は Servo クラスを利用するために、Servo.h をインクルードする。②は変数 servo を定義し、この変数が参照しているオブジェクトを使う。③はポートを指定するための定数である。④は最小のパルス幅を指定するための定数である。使用した SG-90 は 0.5ms になっており us 単位で 500 を代入している。⑤は最大のパルス幅を指定するための定数である。使用した SG-90 は 2.4ms になっており us 単位で 2400 を代入している。

⑥は setup 関数の定義で、プログラム実行時に一度だけ実行される。⑦は attach 関数で Arduino のポート、パルス幅の最小値、最大値を引数で割り当てる。ポート番号だけを与えた場合、パルス幅の最小値は 544、最大値は 2400 がデフォルトとして設定される。

```
#include <Servo.h> // ①

Servo servo; // ②
const int port = 9; // ③
const int min = 500; // ④
const int max = 2400; // ⑤

void setup() { // ⑥
  servo.attach(port, min, max); // ⑦
}

void loop() { // ⑧
  for(int i = 0; i < 181; i++) { // ⑨
    servo.write(i); // ⑩
    delay(30); // ⑪
  }
  for(int i = 0; i < 181; i++) { // ⑫
    servo.write(180 - i); // ⑬
    delay(30); // ⑭
  }
}
```

図 5 サーボモータを回転するプログラムの例

⑧は loop 関数の定義で、この中に記述されたプログラムが無限に繰り返される。⑨はサーボモータを 0 度から 180 度まで 1 度ずつ回転する繰返しである。⑩は write 関数を呼び出し、引数に角度の値を与えることによってサーボモータを回転することができる。サーボモータはパルス幅を制御する事によって回転角度を制御でき、write 関数の引数で受け取った角度によって適切なパルス幅になる様に Servo クラスが変換してくれる。⑪は delay 関数を使って 30ms の遅延を入れ、サーボモータの回転速度を調整している。⑫はサーボモータを 180 度から 0 度まで 1 度ずつ回転する繰返しである。⑬は write 関数を呼び出し、サーボモータを逆回転する。「コンピュータプログラミング A」の授業では、繰返しの制御変数 i は初期値を 0 に、制御変数は 1 ずつ増える記述を習慣づけている。for(int i = 0; i < 181; i++)の記述は、必要な 181 回の繰返しが明確になる。i が増える中で、180 から 1 ずつ減る仕組みを引数の部分で実現している。for(int i = 180; i >= 0; i--)等の記述を行うことによって、引数での補正を防ぐ記述もできるが、繰返す回数に分かり難くなるため勧めていない。この辺りが、「コンピュータプログラミング A を教わった後のハードウェア入門」の拘りである。⑭は delay 関数を使って 30ms の遅延を入れ、サーボモータの回転速度を調整している。

5. アンケート調査結果について

「コンピュータプログラミング A」と同様に、毎回の授業毎に理解度調査を兼ねたアンケート調査を実施した。過去の開催と比較しながら、結果を述べる。

5.1 初日のアンケート調査結果

初日のアンケート調査結果は表 2 に示す。半田付けの経験者が 100.0%であった。Arduino について知っている人は 25.0%であり、少しずつその存在が知られるようになっていくことが窺えるが、まだまだ少ない状況である。「コンピュータプログラミング A」の知識は全員役に立ったと回答した。初日の受講者の感想の中には、「簡単だけど面白いです。」があった。本テーマの開催が受講者の興味を喚起していることが窺える。

表 2 アンケート調査結果 (初日)

質問項目	はい (%)		
	2014	2015	2016
半田付けを行った事はありますか。	68.4	66.7	100.0
電子工作を行ったことはありますか。	31.6	33.3	25.0
Arduino を知っていましたか。	10.5	20.0	25.0
Arduino のセットアップは理解できましたか。	100.0	100.0	100.0
Arduino とパソコンの関係は理解できましたか。	100.0	100.0	75.0
コンピュータプログラミング A の知識は、役立ちましたか。	100.0	93.3	100.0
LED の電極 (アノード, カソード) は、理解できましたか。	100.0	100.0	100.0

5.2 二日目のアンケート調査結果

二日目のアンケート調査結果を表 3 に示す。すべての調査項目について、全員の理解が得られた。

表 3 アンケート調査結果 (二日目)

質問項目	はい (%)		
	2014	2015	2016
const の意味は、理解できましたか。	100.0	100.0	100.0
変数のスコープは、理解できましたか。	100.0	100.0	100.0
ブレッドボードの仕組みは、理解できましたか。	100.0	100.0	100.0
抵抗のカラーコードの読み方は、理解できましたか。	94.7	92.3	100.0
ジャンパは、理解できましたか。	100.0	92.3	100.0
モメンタリ型プッシュボタンスイッチ (タクトスイッチ) は、理解できましたか。	100.0	100.0	100.0
if 文でスイッチの状態を判断する仕組みは、理解できましたか。	100.0	100.0	100.0
バウンスの意味は、理解できましたか。	94.7	100.0	100.0
バウンスの対策に delay 関数を使う方法は、理解できましたか。	89.5	84.6	100.0

5.3 三日目のアンケート調査結果

三日目のアンケート調査結果を表 4 に示す。今年は LED のフェードインとフェードアウトの仕組みの理解が若干低下したが、MOSFET の理解が 100.0%と過去最高となった。サーボモータを取り入れたことによって、大きなものを動かす場合には多くの電力が必要になることを補足説明し、MOSFET で制御することの理解が向上したためと考えられる。最大の狙いである、今後「Arduino を購入し、ハードウェアの工作を行いますか」は、残念ながら 25.0%と過去最低となった。

表 4 アンケート調査結果 (三日目)

質問項目	はい (%)		
	2014	2015	2016
パルス幅変調 (PWM) は、理解できましたか。	100.0	91.7	100.0
LED のフェードインとフェードアウトの仕組みは、理解できましたか。	100.0	100.0	75.0
光センサ (CdS セル) の働きは、理解できましたか。	100.0	100.0	100.0
アナログ出力 (analogWrite) は、理解できましたか。	94.4	100.0	100.0
アナログ入力 (analogRead) は、理解できましたか。	94.4	100.0	100.0
シリアル通信は、理解できましたか。	94.4	100.0	100.0
MOSFET の働きは、理解できましたか。	88.9	75.0	100.0
10 ポイント RGBLED アレイの点灯の方法は、理解できましたか。	-----	83.3	-----
サーボモータの回転方法は、理解できましたか。	-----	-----	75.0
今後 Arduino を購入し、ハードウェアの工作を行いますか。	50.0	83.3	25.0

6. 受講者のモチベーションの推移

「コンピュータプログラミング A」の授業では、受講者のモチベーションの向上を目的に SIEM (School of Information Environment Method: ジーム) を実践している。「コンピュータプログラミング A を教わった後のハードウェア入門」用の SIEM アセスメント尺度が存在しないために要因分析は行えないが、モチベーシ

ンは数値化できる。受講者のモチベーションを1回目と3回目の授業で測定した。調査項目は、重要度「ハードウェアを学習することは重要だと思いますか」、期待度「もっとハードウェアの知識や技術を高めたいと思いますか」を5段階のリッカート尺度で回答を求め、その積の平均で1~25に定量化する。

前期(1回目の授業)のモチベーションは22.8, 後期(3回目の授業)は23.8と、前期から後期にかけて1.0上昇した。この様子を表5に示す。前期から後期にかけて大きく上昇したことは、エクステンションによってハードウェアに対する興味が喚起されたためであると考えられる。「コンピュータプログラミングA」のモチベーションの推移と比較した結果を図6に示した。エクステンションの実施形態が集中講義形式であること、また前期と後期の期間が短いこともあり、単純な比較は難しいが、本テーマの後期の値が高い傾向が窺える。

表5 モチベーションの推移

	前期	後期
2014年	21.3	20.3
2015年	20.8	22.3
2016年	22.8	23.8

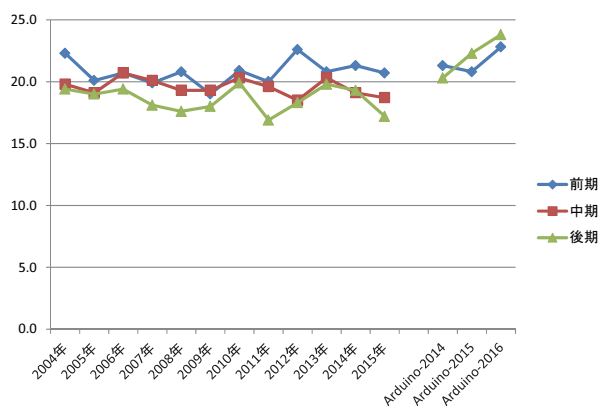


図6 モチベーションの推移

7. 次年度の開催に向けて

マンネリ化を防止するために、毎回、新しいハードウェアを一つ取り入れて行きたいと考えている。次年度の開催では、超音波距離センサー(HC-SR04)を使う予定である。このセンサーの測定可能な距離は、2cmから180cmであり、大学生協を通じて500円程度で購入できる。超音波距離センサーを図7に示す。センサーには左からVcc, Trig, Echo, GNDの端子がある。

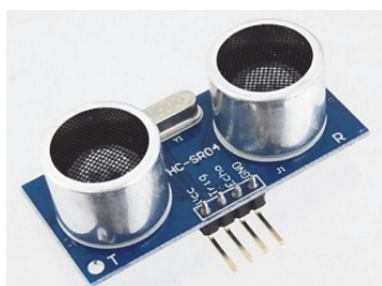


図7 超音波距離センサー(HC-SR04)の様子

<http://akizukidenshi.com/img/goods/3/M-08762.jpg> より引用

Trig端子を10uSの間HIGHにすると(トリガを与える)と、超音波がモジュールから出力される。測定対象の物体から反射波が戻ってくるとEcho端子がHIGHになり、pulseIn関数でHIGHになっている時間を計測できる。この時間の半分が、測定対象までの時間となる。音速が分かれば、この積で距離が求まる。このタイムチャートを図8に示す。Arduinoと超音波距離センサーを接続した様子を図9に示す。

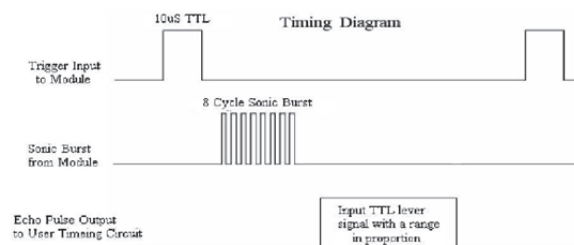


図8 超音波センサーのタイムチャート

http://akizukidenshi.com/download/ds/sainsmar/HCSR04_a_2~180cm.pdf より

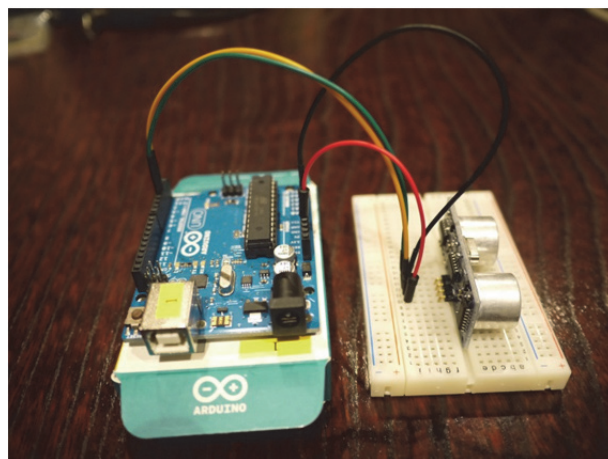


図9 超音波センサーの接続の様子

8. おわりに

「コンピュータプログラミングAを教わった後のハードウェア入門」は、3回目の開催となった。新しく取り入れたサーボモータは、プログラムの実行結果が物の動きとして表現でき、このことが後期モチベーションを向上できた要因であると考えている。次年度は、超音波センサーを取り入れ、マンネリ化を避けたい。複数のセンサーを組み合わせることも、考えていきたい。

参考文献

- (1) 土肥紳一, ArduinoによるコンピュータプログラミングAを教わった後のハードウェア入門, コンピュータ利用教育学会, PCカンファレンス講演論文集, p313-316, 2015
- (2) Arduinoをはじめようキット, <http://www.switch-science.com/catalog/181/>
- (3) Massimo Banzi 著, 船田巧訳, Arduinoをはじめよう 第2版, オライリー・ジャパン, 2012
- (4) コンピュータプログラミングAを教わった後のハードウェア入門, エクステンションプログラム, <http://dohi.chiba.dendai.ac.jp/~dohi/ext-arduino/>