

# 情報工学実験報告書

制作テーマ：マイコン演習1

提出期限：令和5年11月20日

情報工学科 3年 3番

氏名：荒木 渉志

指導教官：前田 弘文

教官コメント：	受付日
	評価

## 1. 実験目的

本実験ではまず、Arduino Uno について学ぶ。そこから Arduino 基礎学習シールドを用いて Arduino Uno の基本操作を覚える。最後に Arduino Uno を用いて音楽を奏でる。以上 3 ステップを達成することを本実験の目的とする。

## 2. Arduino について

Arduino とはワンボードマイコン(マイクロコンピュータ)の一種である。ワンボードマイコンとは必要最低限の電子部品と入出力装置を搭載したものである。また、Arduino はオープンソースハードウェア(設計図が公開されたハードウェア)のため改変や製造、配布や販売ができる。Arduino IDE にて Arduino で実行するコードを作成する。

これは北イタリアのイヴレーアで「Massimo Banzi」「David Cuartielles」「Tom Igoe」「Gianluca」「David Mellis」の 5 人によって開発された。「もっとシンプルに、もっと安価に、技術者でない学生でもデジタルなものを作ることができるようにする」ことを目的に開発され、「Massimo Banzi」が行きつけだったバー「Bar di Re Arduino(アルドゥイーノ王のバー)」を名前の由来とした。アルドゥイーノ・ディヴレーア(アルドゥイーノ王)はイタリアの都市イヴレーアの辺境伯、さらにはイタリア王を務めた。

## 3. 課題

### 課題1. クロスコンパイル

ここでは、Arduino を実験で使用方法について記述する。Arduino を使うにはまず、パソコンに USB で接続する。その後、Arduino IDE を起動し、設定を行う。「ツール」タブから「ポート」を選択し、「Arduino UNO」を指定する。同じく「ツール」タブから「シリアルポート」を指定する。次に、パソコンにてソースコードを作成、保存をしてコンパイルを行う。最後に Arduino にそのファイルをダウンロードし、Arduino にて実行する。このコーディングから実行までの動作を「クロスコンパイル」という。

ここで実行したソースコードを図 1 に示す。これを実行すると図 2 のように L4(13 番ポート)が点灯した。

1	void setup() {
2	pinMode(13,OUTPUT);
3	}
4	
5	void loop() {
6	digitalWrite(13,HIGH);
7	}

図 1 : LED の点灯

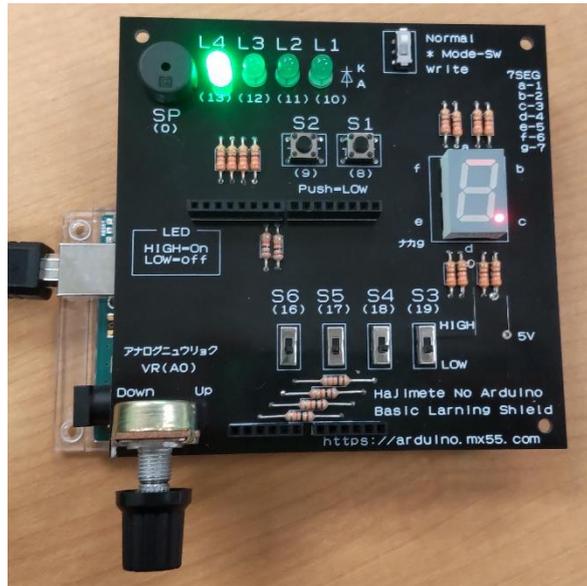


図 2 : LED の点灯

## 課題2. Lチカ

ここでは LED(L4)を 0.5 秒ごとに点滅させるコードを作成する。ソースコードを図 3 に示す。1~3 行目は `setup` 関数内なので一度だけ実行し、13 番ポートの動作を出力に初期化している。6 行目で 13 番ポートを点灯させ、7 行目で 0.5 秒間プログラムを一時停止させる。こうすることで L4 の点灯時間 0.5 秒を実現している。その後、8 行目で 13 番ポートを消灯させ、9 行目でプログラムを 0.5 秒間停止させる。5~10 行目は `void loop()`関数なのでループし続ける。これを実行すると、図 2 のように L4 が 0.5 秒間点灯したのち、0.5 秒間消灯し、これを繰り返した

1	<code>void setup() {</code>
2	<code>  pinMode(13,OUTPUT);</code>
3	<code>}</code>
4	
5	<code>void loop() {</code>
6	<code>  digitalWrite(13,HIGH);</code>
7	<code>  delay(500);</code>
8	<code>  digitalWrite(13,LOW);</code>
9	<code>  delay(500);</code>
10	<code>}</code>

図 3 : LED を点灯させるコード

### 課題3. スイッチ

ここではスイッチ(S3~6(19~16 番ポート))を押すと、そのスイッチに対応する LED (L1~4(10~13 番ポート))を点灯させるコードを作成する。ソースコードを図4に示す。2~5 行目で LED を出力に、6~9 行目でスイッチを入力に初期化する。13 行目で 16 番ポート (S6)が HIGH の時 13 番ポート(L4)を点灯させる。14 行目で 0.5 秒プログラムを一時停止し、15 行目で消灯させ、16 行目で 0.5 秒一時停止する。以降これと同様に各スイッチと LED を連動させて点灯させる。これを実行すると、スイッチ S6 を HIGH にしたとき、図 2 のように L4 が点灯した。そのほかのピンにおいてもスイッチに対応する LED が点灯した。

1	void setup() {
2	pinMode(13,OUTPUT);
3	pinMode(12,OUTPUT);
4	pinMode(11,OUTPUT);
5	pinMode(10,OUTPUT);
6	pinMode(19,INPUT);
7	pinMode(18,INPUT);
8	pinMode(17,INPUT);
9	pinMode(16,INPUT);
10	}
11	
12	void loop() {
13	digitalWrite(13,digitalRead(16));
14	delay(500);
15	digitalWrite(13,LOW);
16	delay(500);
17	
18	digitalWrite(12,digitalRead(17));
19	delay(500);
20	digitalWrite(12,LOW);
21	delay(500);
22	
23	digitalWrite(11,digitalRead(18));
24	delay(500);
25	digitalWrite(11,LOW);
26	delay(500);
27	
28	digitalWrite(10,digitalRead(19));
29	delay(500);
30	digitalWrite(10,LOW);
31	delay(500);
32	}

図 4 : スイッチ連動の LED 点灯

#### 課題4. シリアル通信

ここでは PC から Arduino に苗字を送信し, Arduino から PC へ名前を返信させるコードを作成する. ソースコードを図5に示す. 1~4行目で各変数を定義し, 6行目でシリアル通信のデータ転送レートを 9600bps に初期化する. これは1秒間に 1200文字送れる速度である. 10行目で val にシリアルバッファにあるデータのバイト数(文字数)を代入し, これが 0 でなければ 12~18行目の処理をおこなう. 12行目では data にシリアルバッファにあるデータの一文字目を代入して, それが, 配列 name のインデックスが count の時の文字と等しければ count をインクリメントする, 異なれば count を 0 に戻し, 苗字の最初からまた読み取れるようにする. また, 15行目でその count が5になった時名前を出力し, count を 0 に戻す. これを実行し, PC で苗字の「ARAKI」を入力すると図6のように名前の「TAKAYUKI」が出力された.

1	int val;
2	char data;
3	int count=0;
4	char name[]="ARAKI";
5	void setup() {
6	Serial.begin(9600);
7	}
8	
9	void loop() {
10	val=Serial.available();
11	if(val!=0){
12	data=Serial.read();
13	if(data==name[count]){
14	count++;
15	if(count==5){
16	Serial.println("TAKAYUKI");
17	count=0;
18	}
19	}else count=0;
20	}
21	}

図5：シリアル通信のコード

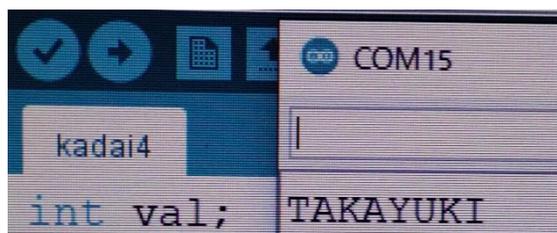


図6：シリアル通信コードの実行結果

## 課題5. 7セグメント LED

ここでは1秒ごとに7セグメントLEDを0~9までカウントアップさせ、その後また0にリセットさせるコードを作成する。7セグメントLEDのポートは図7のようになっている。ソースコードを図8に示す。2行目のfor文内で1~7番ポートを出力に初期化している。7行目以降、0~9までの数字を順番に表示するために、各数字に対応するポートを点灯するようにし、1秒停止した後、消灯させている。0の場合、7番ポート以外を点灯させ、1秒維持したのち、消灯させている。

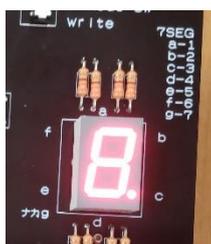


図7：7セグメントLEDのピン配置

1	void setup() {
2	for(int i=1;i++;i<8)
3	pinMode(i,OUTPUT);
4	}
5	
6	void loop() {
7	digitalWrite(1,HIGH);
8	digitalWrite(2,HIGH);
9	digitalWrite(3,HIGH);
10	digitalWrite(4,HIGH);
11	digitalWrite(6,HIGH);
12	digitalWrite(5,HIGH);
13	delay(1000);
14	digitalWrite(1,LOW);
15	digitalWrite(2,LOW);
16	digitalWrite(3,LOW);
17	digitalWrite(4,LOW);
18	digitalWrite(6,LOW);
19	digitalWrite(5,LOW);
20	delay(1000);
21	
22	digitalWrite(2,HIGH);
23	digitalWrite(3,HIGH);
24	delay(1000);
25	digitalWrite(2,LOW);
26	digitalWrite(3,LOW);
27	delay(1000);
28	
29	digitalWrite(1,HIGH);

30	digitalWrite(2,HIGH);
31	digitalWrite(4,HIGH);
32	digitalWrite(5,HIGH);
33	digitalWrite(7,HIGH);
34	delay(1000);
35	digitalWrite(1,LOW);
36	digitalWrite(2,LOW);
37	digitalWrite(4,LOW);
38	digitalWrite(5,LOW);
39	digitalWrite(7,LOW);
40	delay(1000);
41	
42	digitalWrite(1,HIGH);
43	digitalWrite(2,HIGH);
44	digitalWrite(4,HIGH);
45	digitalWrite(3,HIGH);
46	digitalWrite(7,HIGH);
47	delay(1000);
48	digitalWrite(1,LOW);
49	digitalWrite(2,LOW);
50	digitalWrite(4,LOW);
51	digitalWrite(3,LOW);
52	digitalWrite(7,LOW);
53	delay(1000);
54	
55	digitalWrite(2,HIGH);
56	digitalWrite(6,HIGH);
57	digitalWrite(3,HIGH);
58	digitalWrite(7,HIGH);
59	delay(1000);
60	digitalWrite(2,LOW);
61	digitalWrite(6,LOW);
62	digitalWrite(3,LOW);
63	digitalWrite(7,LOW);
64	delay(1000);
65	
66	digitalWrite(1,HIGH);
67	digitalWrite(3,HIGH);
68	digitalWrite(4,HIGH);
69	digitalWrite(6,HIGH);
70	digitalWrite(7,HIGH);
71	delay(1000);
72	digitalWrite(1,LOW);
73	digitalWrite(3,LOW);
74	digitalWrite(4,LOW);
75	digitalWrite(6,LOW);
76	digitalWrite(7,LOW);
77	delay(1000);
78	
79	digitalWrite(1,HIGH);

80	digitalWrite(3,HIGH);
81	digitalWrite(4,HIGH);
82	digitalWrite(6,HIGH);
83	digitalWrite(7,HIGH);
84	digitalWrite(5,HIGH);
85	delay(1000);
86	digitalWrite(1,LOW);
87	digitalWrite(3,LOW);
88	digitalWrite(4,LOW);
89	digitalWrite(6,LOW);
90	digitalWrite(7,LOW);
91	digitalWrite(5,LOW);
92	delay(1000);
93	
94	digitalWrite(1,HIGH);
95	digitalWrite(3,HIGH);
96	digitalWrite(6,HIGH);
97	digitalWrite(2,HIGH);
98	delay(1000);
99	digitalWrite(1,LOW);
100	digitalWrite(2,LOW);
101	digitalWrite(3,LOW);
102	digitalWrite(6,LOW);
103	delay(1000);
104	
105	digitalWrite(1,HIGH);
106	digitalWrite(2,HIGH);
107	digitalWrite(3,HIGH);
108	digitalWrite(4,HIGH);
109	digitalWrite(6,HIGH);
110	digitalWrite(7,HIGH);
111	digitalWrite(5,HIGH);
112	delay(1000);
113	digitalWrite(1,LOW);
114	digitalWrite(2,LOW);
115	digitalWrite(3,LOW);
116	digitalWrite(4,LOW);
117	digitalWrite(6,LOW);
118	digitalWrite(7,LOW);
119	digitalWrite(5,LOW);
120	delay(1000);
121	
122	digitalWrite(1,HIGH);
123	digitalWrite(2,HIGH);
124	digitalWrite(3,HIGH);
125	digitalWrite(4,HIGH);
126	digitalWrite(6,HIGH);
127	digitalWrite(7,HIGH);
128	delay(1000);
129	digitalWrite(1,LOW);

130	<code>digitalWrite(2,LOW);</code>
131	<code>digitalWrite(3,LOW);</code>
132	<code>digitalWrite(4,LOW);</code>
133	<code>digitalWrite(6,LOW);</code>
134	<code>digitalWrite(7,LOW);</code>
135	<code>delay(1000);</code>
136	<code>}</code>

図 8 : 7セグメント LED

これを実行すると 1~9 まで昇順で数字が表示され、図 9 のように「9」を表示したら、また 0 に戻って繰り返した。

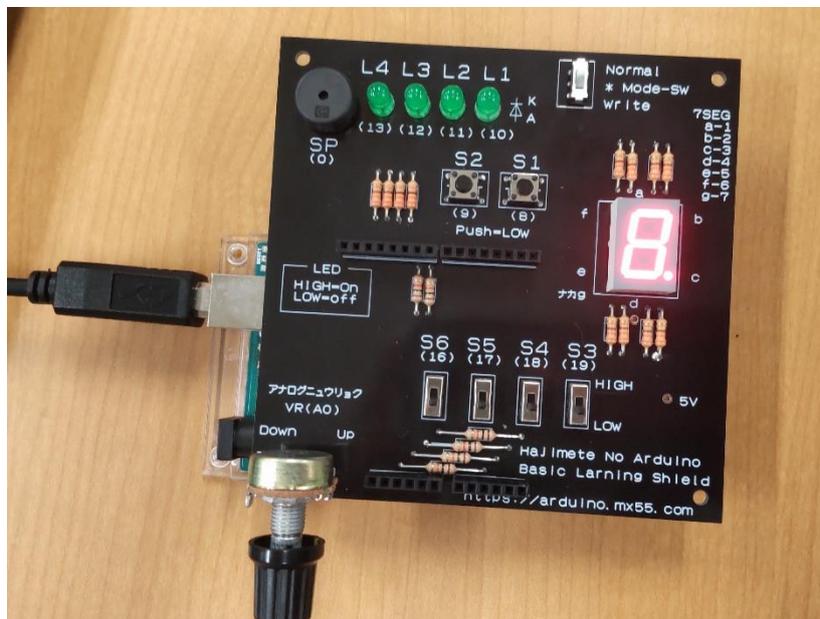


図 9 : 7セグメント LED の実行結果

## 課題6. ボリューム

ここではボリュームを回して、7セグメントLEDを0~9まで均等に時計回りに数字を増加させる。コードを図10に示す。2, 3行目は図9と同様に1~7ポートを出力に初期化している。ボリュームは0~1023まであるためこれを10で割って、102ごとに数字を昇順で変化させる。以降は図9と同様に各数字に見えるようLEDを点灯させるコードが続くため省略している。これを実行すると図9のように数字が表示され、つまみを右に回すと数字が大きくなり、左に回すと小さくなった。

1	void setup() {
2	for(int i=1;i++;i<8)
3	pinMode(i,OUTPUT);
4	}
5	
6	void loop() {
7	if(analogRead(A0)<=102){
8	digitalWrite(1,HIGH);
9	digitalWrite(2,HIGH);
10	digitalWrite(3,HIGH);
11	digitalWrite(4,HIGH);
12	digitalWrite(6,HIGH);
13	digitalWrite(5,HIGH);
14	delay(1000);
15	digitalWrite(1,LOW);
16	digitalWrite(2,LOW);
17	digitalWrite(3,LOW);
18	digitalWrite(4,LOW);
19	digitalWrite(6,LOW);
20	digitalWrite(5,LOW);
21	delay(1000);
22	
23	}else if(analogRead(A0)<=204){

図10：ボリュームで変化させる7セグメントLED

## 課題7. スピーカー

ここではスイッチ(S1)を押し、スピーカーから C3~C4(ド 3~ド 4)を一秒間隔で音を鳴らせるコードを作成する。コードを図 11 に示す。8 番ポートの S1 を入力に、0 番ポートのスピーカーを出力に初期化している。8 行目の if 文でスイッチが押されたかどうかを判断している。なお、S1 は初期状態が HIGH になっているため、入力があると LOW になることから、LOW の時出力するようにしている。以降 0 番ポートから 1 秒間任意の音を鳴らし、その間プログラムを停止させ、また次の音を出力している。なお、音は周波数(131~262)で表している。これを実行すると S1 を押したとき、ド 3 からレ、ミ、と音が高くなっていきド 4 になったら音が止まった。

1	void setup() {
2	pinMode(8,INPUT);
3	pinMode(0,OUTPUT);
4	}
5	
6	void loop() {
7	// put your main code here, to run repeatedly:
8	if(digitalRead(8)==LOW){
9	tone(0,131,1000);//c3
10	delay(1000);
11	tone(0,147,1000);//d
12	delay(1000);
13	tone(0,165,1000);//e
14	delay(1000);
15	tone(0,175,1000);//f
16	delay(1000);
17	tone(0,196,1000);//g
18	delay(1000);
19	tone(0,220,1000);//a
20	delay(1000);
21	tone(0,247,1000);//b
22	delay(1000);
23	tone(0,262,1000);c4
24	delay(1000);
25	}
26	}

図 11 : 音声出力のコード

## 課題8. 音楽

ここでは S1 を押し、スピーカーから音楽を鳴らさせるコードを作成する。なお、ここでは図 12 の「ドラゴンクエスト 序曲」を鳴らす。コードを図 13 に示す。図 7 同様に初期化をし、S1 が押されたとき各音を順番にならして音楽が鳴るようになっている。なお、音符の時間の長さは「 $60000(\text{ms})/\text{BPM}=\text{四分音符の長さ}(\text{ms})$ 」を基に計算した。この曲の BPM は 125 なので四分音符を 480(ms)として、二分音符は二倍、八分音符は $\frac{1}{2}$ 倍、十六分音符は $\frac{1}{4}$ 倍すれば長さが求められる。譜点が付くものには不随している音符の長さを 1.5 倍すればよい。これを実行すると、スイッチを押したときに音源と同様のリズムで音楽が流れた。



図 12 ; 「ドラゴンクエスト 序曲」の楽譜(抜粋)

1	void setup() {
2	pinMode(8,INPUT);
3	pinMode(0,OUTPUT);
4	}
5	
6	void loop() {
7	if(digitalRead(8)==LOW){
8	//1
9	tone(0,262,360);//c4.
10	delay(360);
11	tone(0,262,120);//c4-
12	delay(120);
13	
14	//2
15	tone(0,349,480);//f4
16	delay(480);
17	tone(0,392,480);//g4
18	delay(480);
19	tone(0,440,480);//a4
20	delay(480);
21	tone(0,466,480);//b4
22	delay(480);
23	
24	//3
25	tone(0,523,480);//c5
26	delay(480);
27	tone(0,698,960);//f5
28	delay(960);
29	tone(0,659,360);//e5

30	delay(360);
31	tone(0,587,240);//d5
32	delay(240);
33	
34	//4
35	tone(0,587,720);//d5
36	delay(720);
37	tone(0,523,240);//c5
38	delay(240);
39	delay(240);
40	tone(0,494,240);//b4
41	delay(240);
42	tone(0,494,240);//b4
43	delay(240);
44	tone(0,587,240);//d5
45	delay(240);
46	
47	//5
48	tone(0,523,480);//c5
49	delay(480);
50	tone(0,440,720);//a4
51	delay(720);
52	}
53	}

図 12 : 「ドラゴンクエスト 序曲」を鳴らすコード