

10.Arduino を使用した電子ドラムの作成

村田 憲貴

指導教員 飯坂ちひろ

1. はじめに

私は普段音楽活動をしており、音楽関係のシステムを作成してみたいと思った事や、産技短展での情報技術科のコーナーを人が興味を持ちやすい物にしたいと思った事を踏まえ、Arduino を利用して電子ドラムを自分で作成してみたいと思い、テーマを設定した。

2. 研究概要

2.1 目的

産技短展にて、多くの人に興味を持ってもらえるような作品を作る。また、使用したことのない Arduino を使い、組み込みシステムのプログラミングに関する知識を深める。

2.2 開発環境

本研究における開発環境は、以下に示す通りである。

- | | |
|----------|------------------|
| • OS | Windows 7 |
| • ハードウェア | Arduino UNO |
| • 使用言語 | ArduinoIDE-1.0.4 |



2.3 開発手順

開発手順は以下の順番で行う。

- Arduino IDE のインストール
- パッドの作成
- フォーンプラグ作成
- 配線
- プログラミング
- 実稼働

2.4 全体の完成予想図

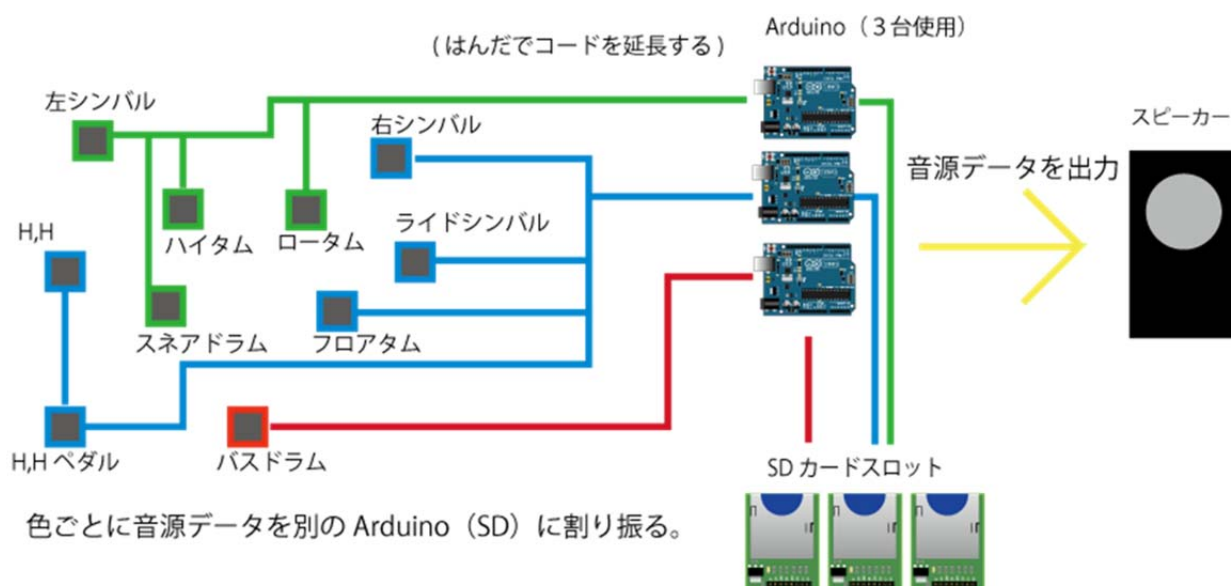


図 1.完成予想図

2.5 Arduino UNO について

AVR マイコン, 入出力ポートを備えた基盤, C++ 風の Arduino 言語とその統合開発環境から構成されるシステム. オープンソースハードウェアなのでハードウェア設計情報のファイルは無料で公開されており, 誰でも自分の手で Arduino を組み立てることができる.本研究では, 書籍などでよく取り扱われている一般的な Arduino UNO を使用する.

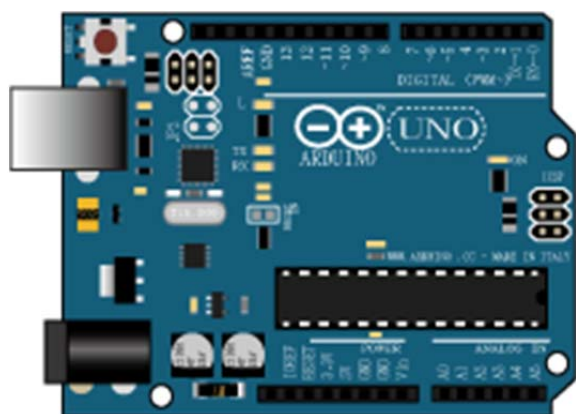


図 2.Arduino UNO

2.6 Arduino IDE について

Arduino の統合開発環境はクロスプラットフォームの Java アプリケーションであり,エディター, コンパイラ,基板へのファームウェア転送機能などを含む.その内部では C 言語のコンパイラ gccやアップロードプログラム avrdudeが使用されている.

開発環境は Processing ベースで,ソフトウェア開発に不慣れなアーティストでも容易にプログラミングできるよう設計されている.プログラミング言語は Wiring から派生したもので,C 言語風の構文で制限の多い基板向けに最適化されている.Arduino ではプログラムをスケッチと呼ぶ.

本研究では, Arduino IDE-1.0.4 を使用する.

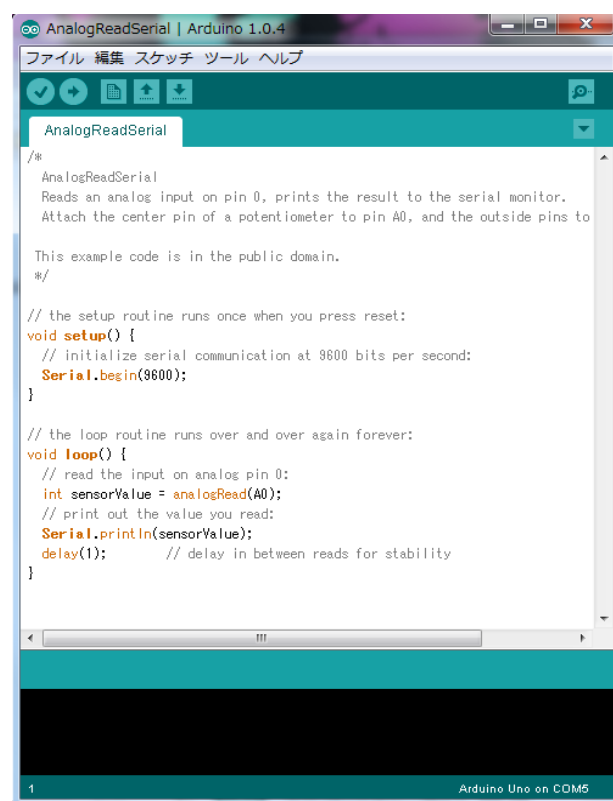


図 3.Arduino IDE

3. 進捗状況

表 1.進捗状況

項目	進捗度(%)
環境設定	100%
動作確認	70%
プログラム	50%
本体作成	0%
レポート	0%

3.1 環境設定

上記2.6で説明した Arduino IDE をインストールは完了,サンプルプログラム等を使用して正常にインストールされたことを確認した.

3.2 動作確認

3.2.1 圧電素子

35mm φ の円盤型圧電素子にリード線が付いたタイプ.発音体やセンサ、発電など、いろんな用途がある.

動作確認プログラムとして.圧電素子が振動を感知した場合,LED が点灯するプログラムを作成し,実証した.

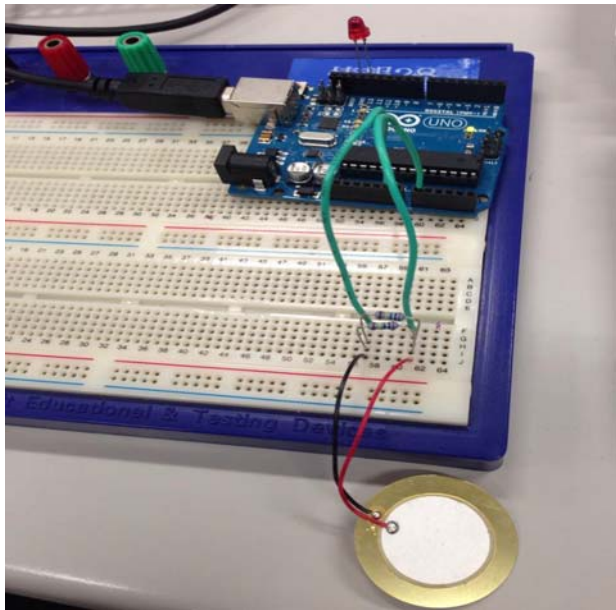


図 4.圧電素子の動作確認用配線
プログラムは以下の通りである。

```
//デジタル 13 番ピンを LED ピンと定義
#define LED_PIN 13

// 変数やピンモードの初期化
void setup(){
    //デジタルピンを出力に設定
    pinMode(LED_PIN, OUTPUT);
}

void loop(){
    //アナログ 0 番ピンが振動を感知した場合
    if(analogRead(0) == HIGH){
        //LED を点灯
        digitalWrite(LED_PIN, HIGH);
    }
    //それ以外（通常状態）
    else{
        //LED を消灯
        digitalWrite(LED_PIN, LOW);
    }
}
```

3.2.2 SD カードスロット化モジュール

SD カードを,このモジュールで開発や実験で手軽に導入できるようになる。

2. 5 4 mm ピッチのピンが実装されているので,カードスロットの実装や配線の必要がない。動作確認用のプログラムとして,SD カード内の wav ファイルを再生するプログラムを作成した。しかし,ノイズがひどく,再生されているか確認できない為,今後改善策を考える必要がある。

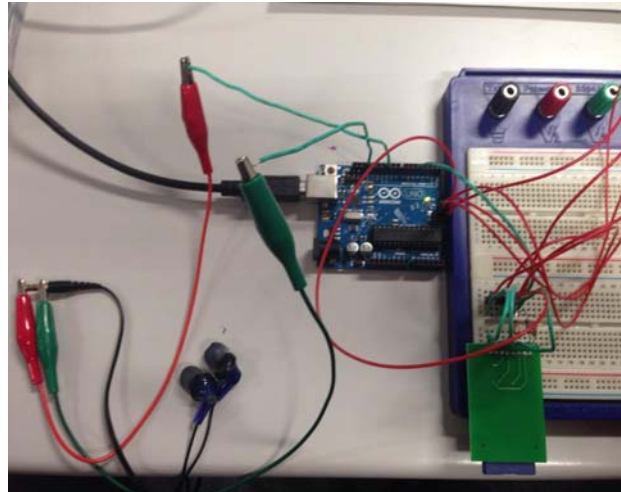


図 5.SD カードスロット化モジュールの
動作確認用配線図
プログラムは以下の通りである。(一部抜粋)

```
#include <SD.h>
#define BUF_SIZE 384 // バッファ・サイズ
volatile uint8_t // グローバル変数
    buf[2][BUF_SIZE], // バッファ
    buf_page, // バッファ・ページ
    buf_flg; // バッファ読み込みフラグ
volatile uint16_t buf_index; // バッファ位置
volatile uint16_t read_size[2]; // バッファ読み込みサイズ
void setup() {
    pinMode(10,OUTPUT); // SD ライブラリ使用
    while(!SD.begin(10)); // ライブラリと SD カードを初期化
}
```

```

void loop() {
    File dataFile;
    if ( !(dataFile = SD.open( "crash1.wav" )) ) return;
    buf_index = 44;  buf_page = 0;  buf_flg = 1;
    // パラメータ設定 (44byte はヘッダ)
    read_size[buf_page]=dataFile.read( (uint8_t*)buf[buf_
    page], BUF_SIZE );
    TIMSK2 |= _BV(TOIE2);
    while(TIMSK2 & _BV(TOIE2)) {
        // データ読み込み指令のフラグが立ったら読み
        込む
        if(buf_flg) {
            read_size[buf_page^1]=dataFile.read( (uint8_t*)buf[b
            uf_page ^ 1], BUF_SIZE );
            buf_flg = 0;  // 読み込んだらフラグを下ろす
        }
    }
    OCR2B = 0;
    dataFile.close();
    delay(500);
}

```

3.3 プログラム

動作確認とほぼ同時進行で作成中。

先述の通り,SD カードスロット化モジュールの動作確認段階で止まっている.スイッチの ON/OFF により再生ファイルが変わるハイハットのみ,IF 文による分岐が追加する.想定している完成プログラムのフローチャートを以下に示す.

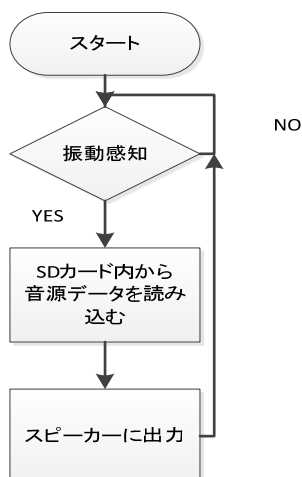


図 6. プログラムフローチャート

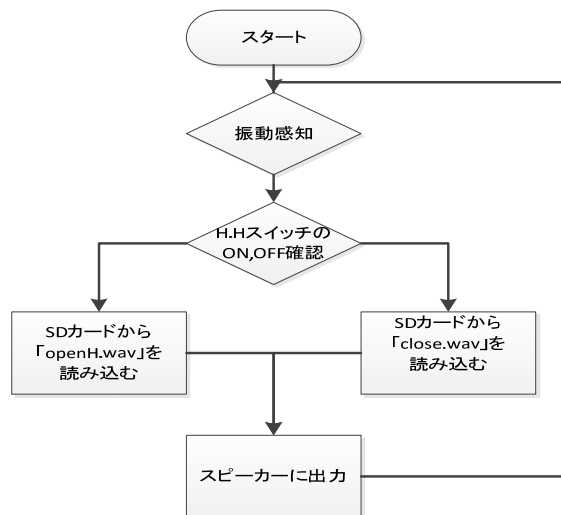


図 7. プログラムフローチャート
(ハイハットの場合)

4. 今後のスケジュール

表 2. 今後のスケジュール

期間		予定
1 月	上旬	プログラム統合, テスト
	下旬	パッド作成, テスト
2 月	上旬	実装, 発表準備
	下旬	レポート作成
3 月		

5. おわりに

作成していたプログラムが上手く動作せず, 勉強に時間を取られてしまった。

3.2.2 で説明した SD カードスロット化モジュールでのノイズ問題を解決し, 次の段階に進まなければ, スケジュール通りに進まないで, 早めに切り上げて外装に取り掛かりたい。

参考文献

[1] ボクの Arduino 工作ノート

[2] Arduino で始める電子工作