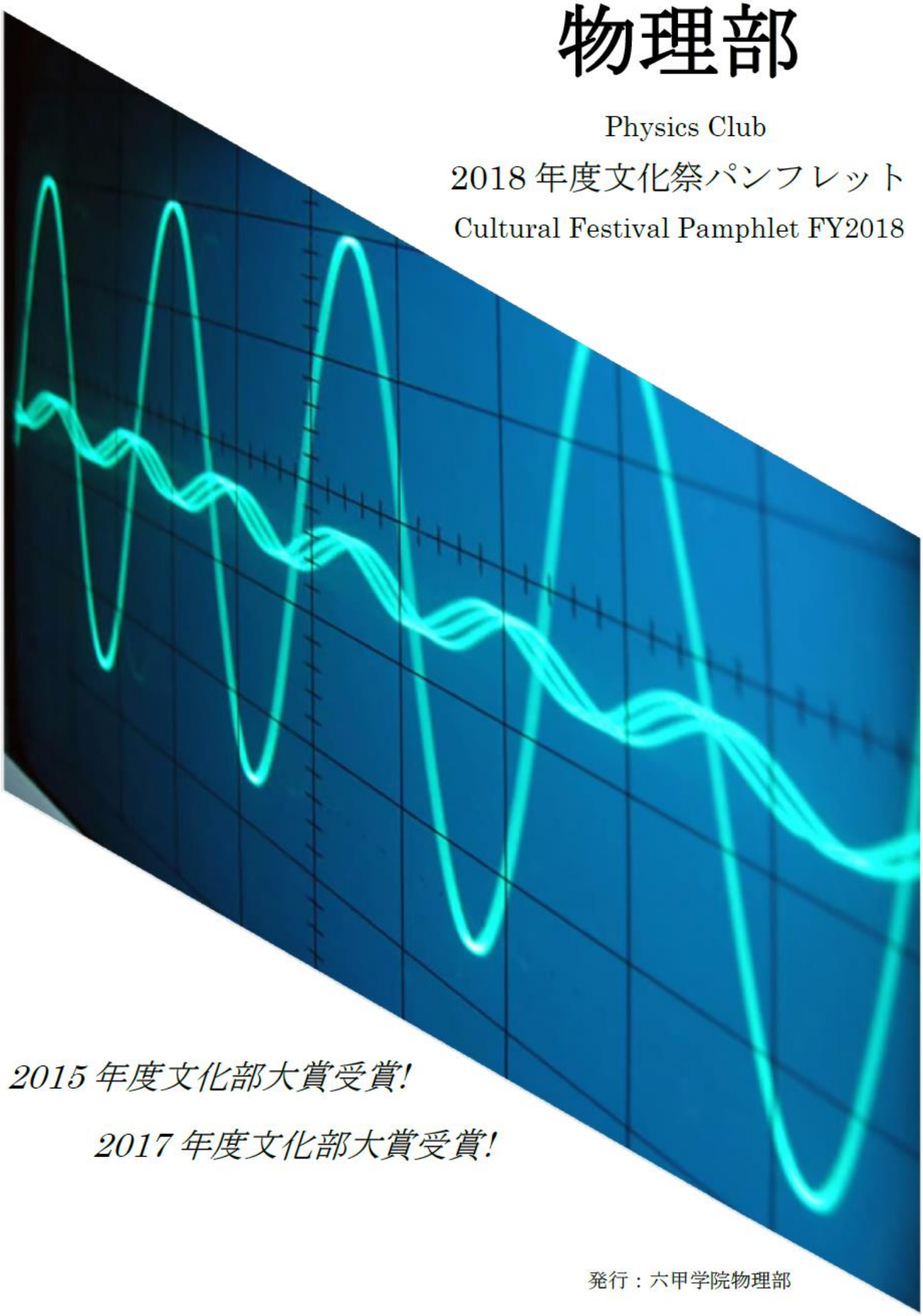
物理部

Physics Club

2018 年度文化祭パンフレット

Cultural Festival Pamphlet FY2018



2015 年度文化部大賞受賞!

2017 年度文化部大賞受賞!

発行：六甲学院物理部

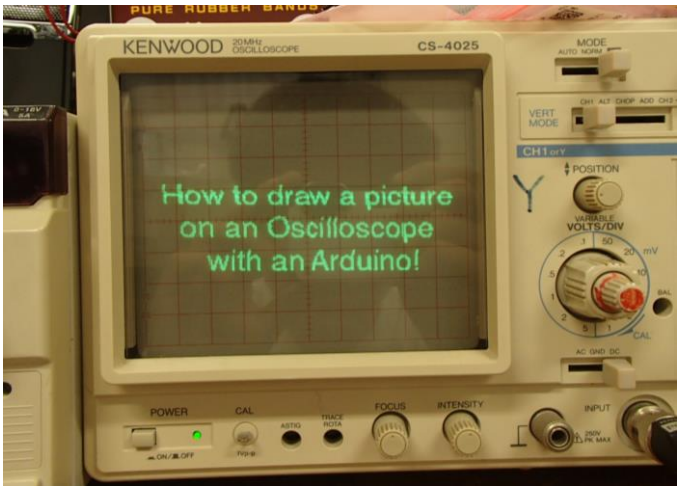
目次

How to draw a picture on an Oscilloscope with an Arduino!

78 期　越智　一稀

①何をするものなのか？

これは Arduino を使って**アナログオシロスコープ**に 256×256 の点描画を表示するものである。



②オシロスコープとは？Arduino とは？

オシロスコープとは、電気的な振動をスクリーンに表示する装置である。

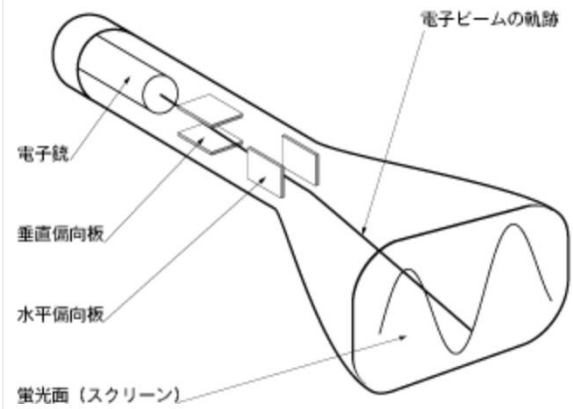
（ウィキペディアより）

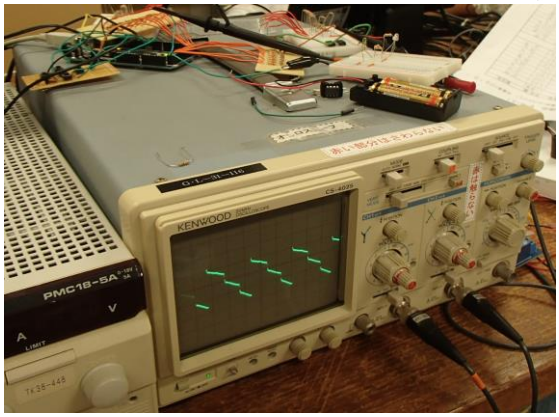
通常、オシロスコープは横軸を時間軸、縦軸を電圧軸として、電気信号の波形を表示する。たいていのオシロスコープにはチャンネルが２つある。
オシロスコープには主に、デジタルオシロスコープとアナログオシロスコープが存在する。この二つの区別は、実は曖昧なのだが、ここではブラウン管オシロスコープをアナログオシロスコープとして説明する。

まず電子銃から電子ビームが出る。この後、電子は偏向板とよばれる２組の金属板の間を通る。上下に配置された偏向板にはチャンネル１または２から電気信号が入力されており、垂直に電界を与えている。このとき、電界が正だと電子ビームは上に、負だと下に曲がる。左右に配置された偏向板にはノコギリ波が入力されており、これにより電子ビームは周期的に左から右に曲がっていく。これを掃引という。電子ビームが蛍光板にあたると光る。この点を輝点という。この輝点の軌跡が波形として表示されるのである。

ちなみに、掃引にはトリガー掃引方式など、少し違う方式もあるのだが、この企画には全く関係ないので説明しない（後述）。

図解オシロスコープの仕組み→





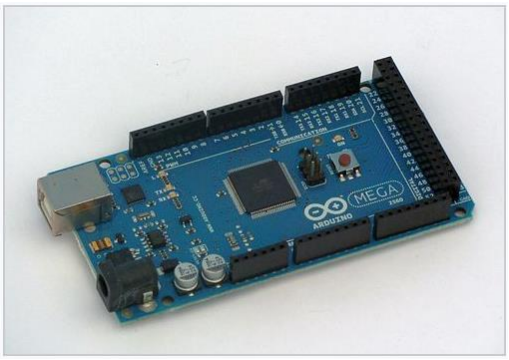
物理部のオシロスコープ
KENWOOD 製　帯域 20MHz
型番：CS-4025

※上にのっている物は気にしないで下さい

Arduino とは、AVR マイコン、入出力ポートを備えた基板、C 言語風の Arduino 言語とそれの統合開発環境から構成されるシステム。（ウィキペディアより）

要するに、プログラムによって入出力ピンの動作を制御したりコンピューターと通信したりする事のできるマイコンである。
主にイタリアで開発され、最近では IoT の定番機器として扱われ、世界的な人気はあの Raspberry Pi と双璧をなしている。（Raspberry Pi に関しては 2017 年度物理部パンフレットを参照）Arduino には、Arduino Uno や Arduino Mega など複数の種類があり、今回は Arduino Mega を使用した。

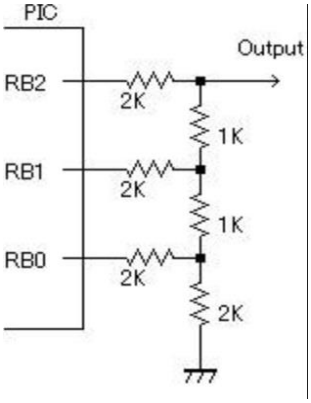
Arduino Mega　物理部のは色違い→



③仕組み

前述では掃引について長々と説明したが、この企画には残念ながら掃引は関係ない。なぜなら時間軸は関係ないからである。実は、たいていのオシロスコープにはもう一つモードが存在する。それが**X-Y モード**である。この X-Y モードは、チャンネル１の電圧を Y 軸に、チャンネル２の電圧を X 軸に表示するものである。本来はリサージュ図形などを使う事で２つの電気信号の位相関係などを測定するものなのだが、ここではリサージュ図形は関係ない。

いま、X-Y モードでチャンネル１に V1、チャンネル２に V2 という電圧が入力されたとする。すると、輝点は X 軸、Y 軸のそれぞれが入力された電圧を示す点になる。
このように、オシロスコープ上の任意の座標に点を打ちたいとき、X-Y モードでチャンネル１、チャンネル２にそれぞれある電圧を加える事で表示する事ができる。
ここで、Arduino の登場だ。
今回は、オシロスコープへの電圧の入力は Arduino を使った。
しかしながら、Arduino は実は任意の電圧を直接出す事はできないのだ。Arduino は基本的に 5V しか出せない（電源電圧 5V の場合）。一応 **analogWrite(pin, value)**という関数はあるのだが、これは、PWM 波（パルスの幅を変化させる事でアナログの電圧を出力するもの、詳しくは 2017 年度物理部パンフレット参照）を使うもので実際には 0V と 5V しか出ていない。これでもモーターの回転数や LED の明るさを変えるには十分だが、オシロスコープにはこのような方法は通用しない。もちろん平滑回路を通せばそのようなこともないのだが、Arduino から出力される PWM 波は 490Hz 程であり、絵を描くには遅すぎると判断したため、**D/A コンバーター**を使う別の方式を採用した（後述）。
この方式は使用するピンの数が増える、回路が面倒くさいなどのデメリットもあるが、スピードは PWM 波より速いというメリットがある。
D/A コンバーターには、**ラダー抵抗式**を採用した。（左、回路図）

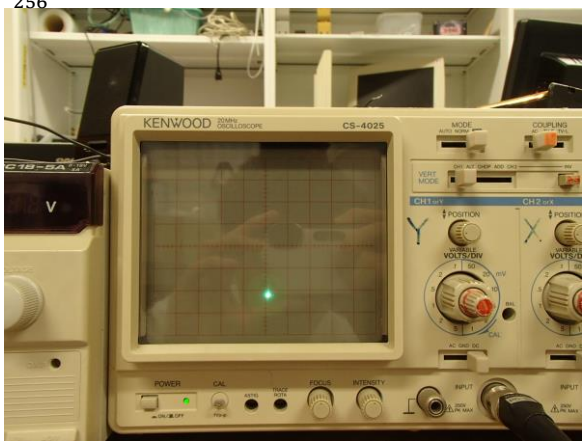


今回は入力ピンを８つずつ用意し、それぞれ 8bit の D/A コンバーターをつくった。この D/A コンバーターでは、入力ピンそれぞれが２進数の桁に対応しており、電圧が入力されたか否かの２進数の値を１０進数の値へ変換して出力する。具体的には、この回路で 0V から 5V まで 256 段階で出力できる。

Output ピンとオシロスコープのチャンネル１、チャンネル２の入力をつなぎ、Input ピンはそれぞれ X 軸側（チャンネル２）が Arduino Mega の pin22~29、Y 軸側（チャンネル１）

が pin49~42 に接続されている。（後述）

今、座標(0,0)を画面の左下に、座標(255,255)を右上に設定する。
例えば座標(128,64)に点を打ちたいとき、チャンネル１には$5 \times \frac{128}{256} = 2.5V$、
チャンネル２には$5 \times \frac{64}{256} = 1.25V$が入力されれば良い。（次ページ図）



（オシロスコープに表示された点。オシロスコープの画面そのものが縦：横が４：５になっているため、画像は横に 1.25 倍引き延ばされる）

プログラム側では、Point(x,y)という関数を定義して、この x と y にそれぞれ任意の座標を入力すると、その場所に点が打たれる。この x と y は１０進数で入力されプログラムの中で２進数に変換され、各桁が１の場合は対応するピンから 5V を出力、０の場合はなにも出力しない。

こうして、点をたくさん打つ事で、Arduino でオシロスコープに絵を描く事が可能になるのである。
パソコン上で作った 256×256 の画像データを Processing で作られた imageconvert_light によって表示する部分の座標をテキストに表示し、それをコピーして Arduino のプログラムの配列に格納する事で、たくさんの点を表示する。
また、備え付けのボタンで絵を描く事もできる。
線や、円を描くコマンドも用意したが、計算の過程で遅れが生じる（後述）のであまりお勧めしない。

④思考錯誤・課題

しかしながら、単純に上記の方法だけではうまくいかない。少し大きな絵を表示しようと思うと描画に時間がかかりすぎる。Arduino が非常に遅いためだ。Arduino を高速化させなければならない。

そこで、まずピンを HIGH（電圧を出力する）、LOW（OFF にする）にする方法を変えてみた。
元は次のような書き方である。（pin8 を HIGH、LOW にするコマンド）
digitalWrite(8, HIGH);
digitalWrite(8, LOW);
しかし、これは処理に４４サイクル必要になる。そこで次のような書き方をする事で処理を３サイクルにまで減らす事ができる。（Arduino Megaの場合）
PORTH |= _BV(5) | _BV(6);
PORTH &= ~(_BV(5) | _BV(6));
これは、PORTH の５番ピン、６番ピンを HIGH、LOW にするという意味で、Arduino Mega では、pin8 は PORTH の５番ピンである事からこのような書き方をする（PORTH の６番ピンは pin9）。Arduino Mega には PORTA から PORTL まであり、それぞれに８つのピンが所属している。
さらに、digitalWrite では、同時に１つのピンしか作動させられないのに対し、この書き方だと同じポートのピンは同時に動かす事ができるのでその分も高速化につながる。
そこで、それぞれの D/A コンバーターに２つのポートを使って入力するようにした。（ピンの数からすると８bit と８つのピンなのでポートは１つで事足りると思うかもしれないが、いつも必ず８つのピンを動かすので更新したくないピンの変わりに動かす何も接続していないピンを用意するため２つのポートとが必要になった）

しかし、これでも遅い。なんと Arduino は計算が遅いのだ。かけ算なら問題ないのだが、割り算はとにかく遅い。そのため、１０進数を２進数に変換する過程で遅れが生じているのだった。
これに関しては、また別のプログラムによって上記の問題に関しても一気に解決する事ができた。
実は、上記の高速化プログラムにはもう一つ種類がある。それが次のような書き方だ。
PORTB = B11000000;
これは、PORTB の０番ピン、１番ピンを HIGH に、２～７番ピンを LOW にするという意味である。この B11000000 は、２進数で 192 を意味している。逆に、
PORTB = 192;
と書けば上と同じ意味になるのである。前述で、Point(x,y);の x、y の場所に点を打つと書いたが、この書き方を使えばわざわざ２進数に変換しなくても１０進数の数字をそのまま入力すればいいのだ。
よって(x,y)に点を打つだけの場合は次のように書けばいい。

```
PORTA = x;
PORTL = y;
```

前述で X 軸側が pin22~29、Y 軸側が pin49~42 に接続されているとあったが、これは Arduino Mega の PORTA が pin22~29、PORTL が pin49~42 を指しているからである。

このようにして、Arduino を高速化した。
上記の『How to~Arduino!』の画像は 1 秒間に約 190 回描画されている、はずである。

また、Arduino はメモリ（SRAM）の容量が小さく、たくさんの配列を格納できない（2000 個程度が限度）という問題があったが、より容量の大きいフラッシュメモリ側に書き込むプログラムによって解決した。

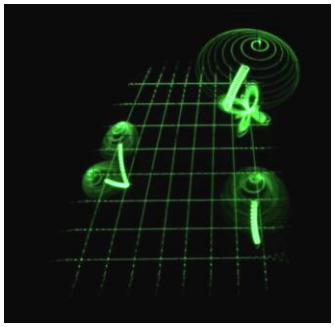
画像が一部歪んでしまっているが、残念ながら直す事はできなかった。

⑤ちなみに・・・

- 今回の物理部のロゴはこの仕組みによって描かれた。
- 実験に使用したアナログオシロスコープは、神戸大学の元学生実験用のもののうちの一台であり、2016 年に物理部へ来た。

この手の『遊び』をやっている人はそれなりにいるらしく、ネット上に上がっていた他の方法もご紹介する。

- 前述で、PWM 波を使うのは遅すぎると書いたが、中には PWM 波を使っている人もいた。どうやら、PWM 波の周波数は設定を変えられるらしく、この場合は 31kHz まで周波数を上げていた、が、やはり描画は遅いようだ。
- ラスタースキャン方式の人もいた。これは、X-Y モードは使用しない。ブラウン管テレビの走査線と似ているが、この場合は縦に線を走らせ、同期信号を出してトリガーを調節し、輝度変調信号で線の明るさを変えることで絵を表示している。
  最初はこの方式も考えたが、難しそうだったので止めた。
- 電圧の入力を Arduino などのマイコンではなく音声信号を使っているものもあった。
  おそらく、シンセサイザーなどで音声信号を合成し、それをチャンネル１、２に入力しているのだと思う。この場合、それぞれの波形の位相関係なども考慮しなければならず、かなり難しい方法ではあるが、YouTube には『How To Draw Mushrooms On An Oscilloscope With Sound』という動画が投稿されている。



←オシロスコープに表示された様々なキノコ

これらの実験はいくつか用意が必要だが、興味がある人などは是非やってみてはいかがだろうか？

⑥謝辞

今回の企画にあたっては、顧問の溝内先生、Processing で imageconvert_light を作ってくれた 77 期の伊藤先輩にこの場を借りて御礼申し上げます。ありがとうございました。

C++によるオセロ AI

78 期　阿比留　波音

①オセロ AI について

オセロ AI というのはいろいろな種類があります。

①Min-Max 法

Min-Max 法は、後の自分の駒の状況（角をとれるか、自分の駒の位置、数など）のうち、考えうる最悪の状況が最もましになるものを探査する方法です。今回の AI ではこの方法を使っています。

②NegaMax 法

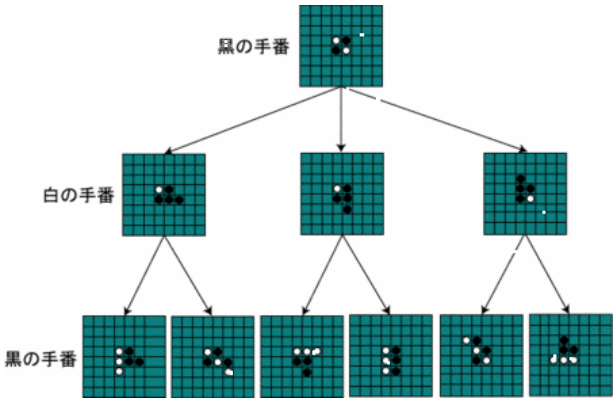
NegaMax 法は、双方のプレイヤー（人間と CPU）がそれぞれにとっての最善手を打つのを想定して最も良い手を探査する方法です。

③α β法

基本的なプログラムは①の Min-Max 法と同じですが、こちらはゲーム木において計算する必要のない部分を刈り取るという方法です。

＝＝ゲーム木とは＝＝

自分と相手の後の手を樹形図状に表したもの（図１）です。



←図 1

②動作環境

Microsoft Visual Studio 2017 C++

③今回の反省

・GUI の表記を漢字にすることができなかった。

・プログラムを走らせている間に応答なしになることが多い。

音楽ゲーム製作

78 期　越 智　　一 稀
78 期　出原　健太郎

・ このゲームの特徴

この音楽ゲームは、主に「ハードウェア部分」と「ソフトウェア部分」の二つで構成されていて、二人の部員で分業をして製作しました。

「このゲームが動く仕組み」を簡単に説明すると、

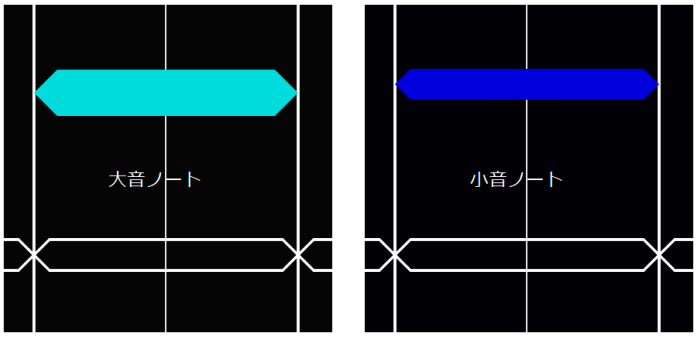
1.プレイヤーがパネルに触れる。

2.「パネルに触れた」という情報をマイコンが処理して PC に伝達する。

3.PC 側のプログラムでタイミングの判定などをする。

…という感じです。

このゲームの特徴として「大音」と「小音」があり、パネルの特性である「パネルに触れている面積によって反応が変わる」というものを生かしたゲーム仕様となっています(図 1)。



(図 1)左が大音ノートで右が小音ノート。大音ノートは手のひらで触れることで高得点が得られる。

それでは詳しい説明に移っていきましょう。まずは 78 期越智製作のタッチパネルの説明です。越智に筆を譲ります。

・パネルシステム-LPF タッチセンサー

このパネルは静電容量方式のタッチセンサーを４つ並べる事で構成されている。
このパネルの特徴は、２段階判定（大音・小音）ができる事である。

このパネルの回路は Arduino の pin8 から抵抗を通して金属板が接続されており、各金属板は１番パネルが pin9、２番パネルが pin10、３番パネルが pin11、４番パネルが pin12 に接続されている。金属板には製本テープを貼る事で手と金属板が直接触れる事のないようにしている。

pin8 は OUTPUT に、pin9~12 は INPUT に設定されている。pin8 からはパルスを出しており pin9~12 では入力される電圧の立ち上がり速度を計測している。
これらはそれぞれ、ある程度以下の電圧は LOW、ある程度以上の電圧は HIGH として計測される。
これが LOW のときの時間を計測し、それがある程度以上の時間に達したとき、パネルは触れていると判定される（小音判定)。そして、さらにそれよりもある程度大きな時間が計測されたときそれが大音判定となるのである。

- 触れていないとき、パルスは抵抗を通ってそのまま入力ピンに入るので、電圧はすぐに立ち上がる。そのため立ち上がり速度は速い。

- 触れているとき、人間と金属板がそれぞれコンデンサを構成する。これによってこのパネルの回路は抵抗と金属板と人間で CR ローパスフィルタ（LPF）を構成する。ローパスフィルタにパルスが入力されると立ち上がりが遅くなる。この時間を時定数という。時定数はコンデンサの容量と抵抗値で決定される。よって手と金属板の距離が近い程静電容量が大きくなるので時定数も大きくなる。つまりパネルに手を触れることで静電容量が大きくなり、センサーは反応する。さらに、手とパネルの触れる面積が増える事でより容量が大きくなり時定数は増大する。これによって大音判定が可能になる。

金属板と手が直接触れてしまうと、時定数が増大しすぎて触れる面積の変化に対する時定数の変化量の割合が小さくなってしまい、大音・小音の判定が難しくなる。そのため製本テープを利用して金属板をカバーした。

パネルの判定はシリアル通信を利用してコンピューターに送られる。
それぞれ小音は 1 番パネルから 4 番パネルまで順に 1,2,3,4 の数字が送信される。
大音は 1 番パネルから順に 5,6,7,8 が送信される。送信速度は 2000000bps である。

以上がこのパネルシステムの説明である。

・PC 側のプログラム

簡潔明瞭な説明ありがとうございました。どうも再登場した出原です。今回製作したプログラムの簡単な説明をしていきたいと思います。

まず、このゲームは”Java”というプログラミング言語で書かれています。しかし、プログラミングに詳しい人なら知っているかもしれませんが、本来 Java はゲームプロ

グラミングに適した言語ではありません。よく使われるのは C++とか C#とかですね。じゃあ何故 Java で書いたのかって?それは僕が Java しか扱えないからです(オイ)

こんな茶番はさておき、プログラムの仕組み(アルゴリズム)を覗いてみましょう。

まず、パネル側に入力された情報は、Arduino を通してシリアル通信で PC に送られます。それを、シリアル通信を読み取る”RXTXcomm”というクラス(機能がまとめられたもの)を使い、0,1,2 (順に、何も押されていない状態、普通に押した状態、手のひらで押した状態)の信号に変換します。

次に、その信号を使って、「パネルを押した瞬間」の信号を取り出し、それをゲームのタイミング判定や、メニュー画面での選択に利用します。

このゲームは 1 秒あたり 60 回の画面更新をしていて、一番得点が高い Perfect の許容誤差が 2/60 秒(約 0.03 秒)、Good が 4/60 秒(約 0.07 秒)、Fair が 7/60 秒(約 0.12 秒)以内となっています。数字だけで見るとかなり厳しいような気がしますが、実際にプレイしてみるとまあそんなもんです。

ここまでゲーム内の処理についていろいろ書いてきましたが、実は実際的な処理はソースコードの 15%程度しかありません。

残りの 85%は何を書いているのかというと、「プレイヤーに快適なゲームプレイを提供する」ために必要なコードを書いています。具体的に言えば、「見た目」です。

例えば、起動と同時に何やら複雑そうな画面(図 2)が出てきて、その画面にコマンドを打ち込むと曲が始まり、画面は変わらないままタイミングよくたたいたら点数が増える、というのでは何も楽しくありませんよね(実際何も設定しない状態だとこうなるし、そっちのが楽です)。

(図 2)こんなではお世辞にもゲーム画面とは呼べない。

だから、ゲームプログラミングには、プレイヤーに楽しんでもらうために、実際のゲームプレイに必ずしも必要でない「見た目」を追求する必要があるのです。

・・・裏話的になりますが、メニュー画面のアニメーションを作るのには丸三日くらいかかりました。

・反省、感想

・越智

このパネルの製作に取り掛かりだしたのは実際ソフトウェアより 1 ヶ月程前（2017 年 10 月）のことだった。最初、うまくいく確信がなかったためである。その結果当初予定された 3 段階判定は 2 段階判定になりその後も反応の鈍さや誤判定などいくつもの問題に直面した結果、パネルの完成は文化祭直前にまで伸びてしまった。しかし、まだ完全形とはいえず当初の読みの甘さが最後まで響いたように思う。

・出原

開発に時間がかかりすぎてしまい、肝心の譜面作成に割ける時間が少なくなってしまったのが痛い。本当は 20 曲くらい収録したかったところ。JavaFX とかいう GUI システムを使いこなすための勉強期間が 5 ヵ月とか時間かけすぎやろ…。

でもまあ正直なところ、作り始めた段階ではここまでクオリティを上げられるとは思っていなかったので、そこに関してはよかったと思う。

初めて総文字数 100,000 文字達成したから誰か褒めて(ボソッ

剛体三重振り子のカオス

78 期　出原　健太郎

①はじめに

「剛体三重振り子」とはなにやら聞き慣れないゴツそうな言葉ですが、簡単に言うと「変形しないもの(≒剛体)をつなぎ合わせて、振り子状にしたもの」のことです。

今回の研究は、この三重振り子を自作し、その挙動について研究しました。

②使用物、作業工程

使用物：

アルミニウム角材 30×300×1000[mm]

アルミニウム角材 30×200×100[mm]

アルミニウム丸棒材 φ4×100[mm]

ベアリング d=4,D=7,B=2.5[mm]

土台用鉄材 適量

作業工程：

1. まずアルミ材を加工し、長さ 300mm×2、200mm×2、100mm×1 に切断します。
2. 次にアルミ材に穴を開け、アルミ材自体が振り子の働きをするようにベアリングをはめ込みます。
3. ベアリングに軸としてアルミ丸棒材を挿入し、別のアルミ材に固定します。
4. 三重振り子の形になったら、一番上のアルミ材に主要回転軸を通し、土台に固定して完成です。

③挙動

この三重振り子には、おもしろい性質があります。

三重振り子の下端を掴んである程度の高さまで引き上げ、手を離すと言うまでもなく振り子は振動を始めます。しかし、まもなく振り子がおもしろい挙動を見せ始めます。先端部分の振り子が不規則に高速回転をしたり、思い出したかのように止まり、また逆回転を始めたりします。全く予測不可能な動きを見せ始めるのです。

この、振り子が意志を持って暴れているかのような動きは、「カオス」ないしは「決定論的カオス」と呼ばれるものです。今回は、この決定論的カオスについて解説していきます。

この三重振り子は、「決定論的」カオスというだけあって、初期状態を定めればt秒後の位置と速度はただ一通りに決定します。具体的に言うと、「ラグランジュ運動方程式」というめちゃんこ難しい式を解くことで状態を求められます(→補足)。数学でも習う、所謂「関数」というやつですね。

ここが決定論的カオスにおける最も重要な特徴の一つなのですが、初期状態が少しでも(振り子の分子配列が一つ違っただけでも！)変動すると、その後の挙動は完全に別のものになります。この性質のことを、初期値に鋭敏に反応するという意味で「初期値鋭敏性」(もしくは「長期予測不可能性」)といいます。

この他にも、決定論的カオスには「非線形性」(行列で定義されず、微分方程式が線形性を持たない)「有界性」(返値が有限範囲内に離散する)「非周期性」(周期的に繰り返されることはない)という特徴があるのですが、ここでは割愛します。

④応用

前項で述べたとおり、決定論的カオスは、初期値を決定すれば任意秒後の状態をシミュレーション計算することができます。しかも初期値鋭敏性により、似たような初

期値からでも全く別の結果を得ることができます。この性質を応用して、コンピュータ上で疑似乱数を生成するときなどに、決定論的カオスが用いられます。

コンピュータ上で生成される疑似乱数は、現代一般的に用いられている暗号化技術である「公開鍵暗号」などにおける必須の要素であり、少し話を飛躍させると、「決定論的カオスが我々のセキュリティを守っている」と言っても過言ではないでしょう。

決定論的カオスが我々のセキュリティを守っている

はい。

⑤反省

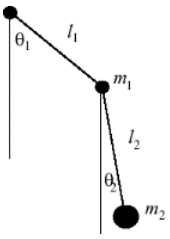
今回製作した三重振り子は、エネルギーロスを減らすためにベアリングを用いるなどいろいろな工夫をしたが、加工精度の問題でやはりエネルギーロスは大きくなってしまった。

もちろんロスの削減には限界はあるが、もう少し減らせたのではないかと思う。

また、今回工作には加工が簡単なアルミ材を使用したが、もっと比重の大きな金属を使うなどの試みもしてみたい。

⑥補足

この場合のラグランジュ運動方程式は、画像のようにパラメーターを設定すると以下のようになる。



$$(m_1+4m_2)l_1\ddot{\theta}_1+2m_2l_2\ddot{\theta}_2\cos(\theta_1-\theta_2)+2m_2l_2\dot{\theta}_2^2\sin(\theta_1-\theta_2)+(m_1+2m_2)g\sin\theta_1=0$$

$$l_2\ddot{\theta}_2+2l_1\ddot{\theta}_1\cos(\theta_1-\theta_2)-2l_1\dot{\theta}_1^2\sin(\theta_1-\theta_2)+g\sin\theta_2=0$$

⑦参考文献、web サイト

https://ja.wikipedia.org/wiki/%E5%85%AC%E9%96%8B%E9%8D%B5%E6%9A%97%E5%8F%B7 公開鍵暗号 - Wikipedia

https://ja.wikipedia.org/wiki/%E4%BA%8C%E9%87%8D%E6%8C%AF%E3%82%8A%E5%AD%90 二重振り子 - Wikipedia

https://ja.wikipedia.org/wiki/%E7%B7%9A%E5%BD%A2%E5%8A%9B%E5%AD%A6%E7%B3%BB 線形力学系 - Wikipedia

ネット対戦ゲーム

80 期　楠田　力基

1 ゲームの概要

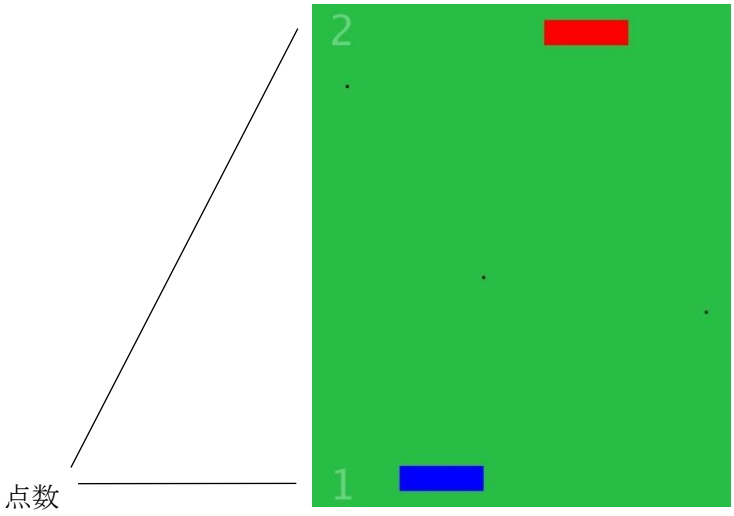
インターネットを通じて、２台のパソコン同士でエアホッケーゲーム？ができる。

2 使用したもの

- Processing3
- Network ライブラリ

3 内容

- 下のバーが自分、上のバーが相手。
- 矢印キーの‘←’・‘→’ で操作。‘R’キーでリトライ。
- そのままでは、あまりにも難易度が低かったので、ボールは３つある。
- ボールが自分側の画面下につけば相手に１点。相手がミスすると自分に１点。
- 相手のパソコンには、逆転した画面が映る。



（サーバー側の画面）

4 反省など

インターネットを使うことに気を取られて、ゲームとしての完成度が低くなってしまった。その上、通信が安定しないときに画面が点滅したり、ボールがバーを通り抜けてしまったりするので、改善したかった。

来年の文化祭では、今回の反省点を生かして余裕をもって準備をしたい。また、“物

理部”なので、ゲーム以外のものにしようと思う。

<参考>

- http://snakamura.org/teach/fms/programming2_08.pdf
- http://www2.kobe-u.ac.jp/~tnishida/misc/processing-net.html

ブロックキャッチゲーム

80 期　高橋　佑典

①はじめに

　このブロックキャッチゲームは、僕自身ほとんどプログラミングについて知らないところから作り始めたものです。シンプルなゲームで、とても簡単なつくりなので、一度気軽に遊んでみてください。

②ルール、遊び方について

　名前の通り下にある黒色のバーで、ランダムに落下してくるブロックをキャッチするゲームです。

青：触れると+10 点　触れられずに下に落ちてしまうと HP－1

赤：触れると HP－1

最初に 5 ある HP が無くなってしまうとゲームオーバーです。

また、左下にあるメーターが 1 増えるごとにブロックの落下速度が速くなります。

より高い点数を目指しましょう。

③大変だった点

　最初にも書きましたが、本当に基本的なプログラミングの用語だけ覚えただけのところからの制作だったので、全体的に進度が悪かったこと。

また、ブロックの落下の速度を変えるプログラムの作り方がどうすれば良いかわからず、少し時間がかかってしまいました

④反省点

もう少し時間があれば、HP を回復させるブロックを作ったり、落下してくるブロックの速さをランダムにしたりするなど、もっと面白いゲームにできたと思うので、もう少し余裕をもって作られれば良かったと思いました。

⑤参考文献

http://r-dimension.xsrv.jp/classes_j/animation/
https://detail.chiebukuro.yahoo.co.jp/qa/question_detail/q1380003888

ライントレーサーカー

80 期　高橋　佑典

①はじめに

ライントレーサーって何のこと？と、なる人はあまりいないと思いますが、実際の動作原理を知っている人は少ないかもしれません。現物を見て、実際に動かしてみてこの機会に、なぜ線の上を走るのかを覚えて帰ってください。

ライントレーサー：白い紙の上を、黒い線に沿って自動的に走る車。

（やばい…あと三日しかないのに…二回目の作り直し…文化祭に展示できているのか…）

②使用した部品

部品の名前	型番	個数
抵抗器	1kΩ	2
	75Ω	1
トランジスター	2SC1815	2
	2SD1828	2
フォトトランジスター	NJL7505L	2

LED	赤色（2.0V　20mA）	1
（LED 光拡散キャップ）	白色	1
可変抵抗器	100kΩ	2
積層セラミックコンデンサー	0.1μF	2
スイッチ	小型スライドスイッチ	1
ユニバーサル基盤	15×15 穴	1
電池ボックス	単 3 形×2 本	1
電池	単三形	2
ビニール線	12cm	8
駆動系	タミヤツインモーターギヤーボックス	1 セット
	タミヤトラックタイヤセット	1 セット（タイヤ 2 個分）
	タミヤボールキャスター	1 セット
	タミヤユニバーサルプレート	1

③動作の仕組み

車体の前にある LED の光を、その左右に取り付けたフォトトランジスターが受け、床の反射で地面が白色か黒色かを見分け、線の有無を感知しながら進みます。

このライントレーサーは黒色を感知するため、例えば右カーブに差し掛かった時、右のフォトトランジスターが黒色を感知して、右のタイヤだけを止めます。そうすると、左のタイヤは回り続けるため、車体は右に曲がります。

④反省点・課題

恐らくはんだ付けの時に失敗してしまい、二回作り直しになってしまった点。
→はんだ付けを一回で決める。精度を上げる。
→せめて、動かない原因を自分で見つける。

⑤参考文献

『電子工作パーフェクトガイド』/ 伊藤尚未 / 新光社

人数カウンター

80 期　中村　勇貴

・簡単な説明

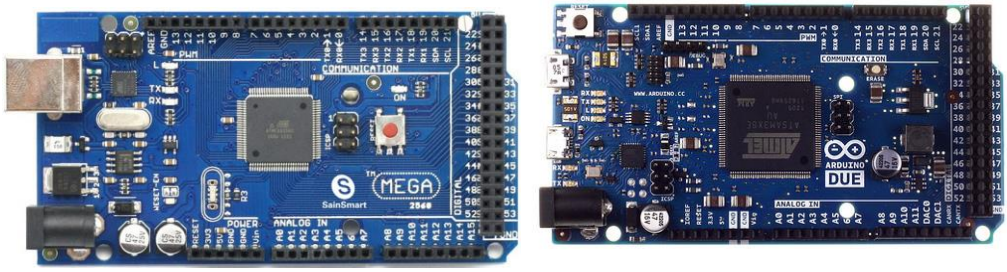
超音波距離センサーを使ってドアを通る人の人数をカウントし、液晶に表示しています。

（文化祭の来場者数を数えています。）

・使用した部品と部品説明

Arduino MEGA（互換品）（右）

Arduino DUE（互換品）（左）



Arduino とは？

パソコンに USB で繋ぎ、C++に近い"Arduino 言語"を使ってプログラミングすることで、

自由自在に電子部品を動かせるものです。

例えば、

```
void setup(){
    pinmode(13, OUTPUT);
}
void loop(){
    digitalWrite(led, HIGH);
    delay(1000);
    digitalWrite(led, LOW);
    delay(1000);
}
```

と書くと、基板上の LED が 1 秒ごとに点滅します。

詳しい説明は、"Arduino"と検索して下さい。

超音波距離センサー ×2 （右: HC-SR04, 左: HC-SRF05）

仕組み

超音波を出し、それが返ってくるまでの時間を計ってそこから距離を求めています。

（今回は距離が短くなるのを感知し、人が通ったことを検知しています。

LED ×2 (赤、緑)

抵抗(20Ω)

ケーブル

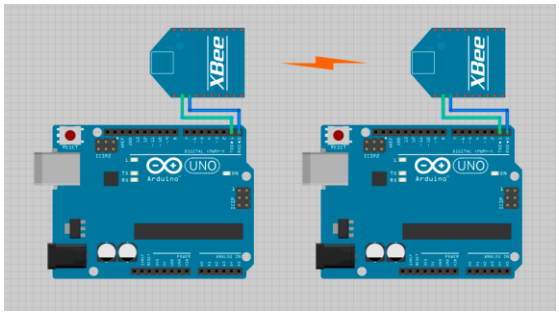
EPSON S5U13781R01C100 (グラフィックコントローラー)

(Arduino DUE に接続し、液晶に表示しています。)

FG040346DSSWBG03 (液晶)

XBee R3 ×2

AE-XBee-REG-DIP ×2 (XBee とブレッドボードの穴の大きさを合わせるための変換基板)



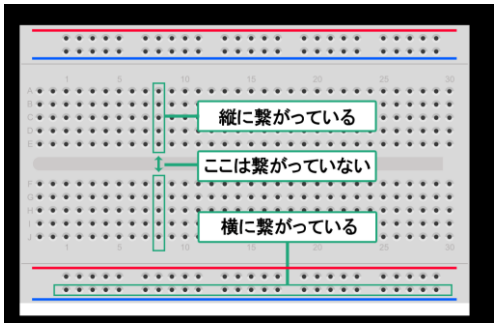
XBee とは?

XBee は無線モジュールの一つで、使い方の一つとして Arduino 同士でデータを

やり取りする”シリアル通信”を無線で行うことができます。(上の画像)

他の使い方や詳しい説明は、僕には無理なので検索してください。

ブレッドボード ×2 (ケーブルを差し込むだけで簡単に回路を作れる基盤(?)です)



・仕組み

この人数カウンターは超音波センサーでドアを通った人数を計測し、それを XBee で机の上の Arduino に人数を送信し、その Arduino で液晶に人数表示しています。
超音波距離センサーは外と中に設置していて、人数を計測する仕組みは、

人が入る時： 1 外のセンサーの距離の値が減少する。
2 中のセンサーの距離の値が減少するまで待つ。
3 中のセンサーの距離の値が元に戻るまで待つ。

人が出る時： 1 中のセンサーの距離の値が減少する。
2 外のセンサーの距離の値が減少するまで待つ。
3 外のセンサーの距離の値が元に戻るまで待つ。

これを Arduino のプログラムで検知し、人が入ったときには人数を 1 増やし、人が出たときにはそのまま続けます。仕組みはそれほど難しくないかわりに、精度は…(苦)
ということで、余りあてにはしないでください…
この人数を XBee でドア近くの Arduino から机の上にある表示用の Arduino に送信し、液晶にプログラムどおりに表示させる、という仕組みです。

p.s. 文字ばかりですいません。

・反省

去年の文化祭で入場者数をカウントできたらいいなと思ったのがきっかけです。精度がいまいちで統計には使えないような精度に仕上がってしまったのが惜しいです。また、表示の液晶部分の制作が遅くなってしまったので、次は余裕を持って取り組みたいです。

電子工作体験

平成３０年度

担当者　78 期　越智　一稀

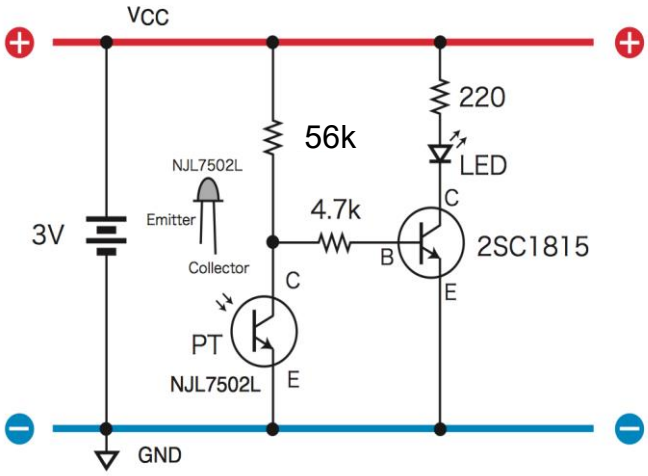
今回の電子工作体験では、フォトトランジスタを使って周囲が暗くなると LED が点灯する回路を製作する。

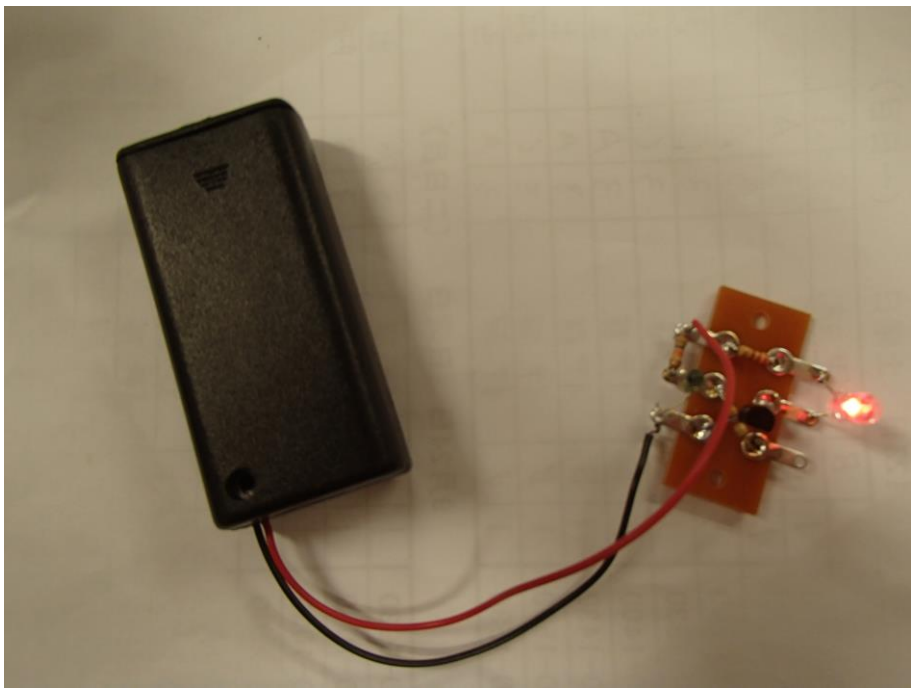
回路図は、下記の通り。

(http://cms.db.tokushima-u.ac.jp/DAV//person/S10723/LED を使いこなそう/公開講座 06-2012SS.pdf)より。抵抗値を一部変更。

部品

- トランジスタ 2SC1815Y
- フォトトランジス NJL7502L
- 高輝度赤色 LED
- 抵抗 220Ω
- 抵抗 4.7kΩ
- 抵抗 56kΩ
- ラグ板
- 電池ケース





各部品の動作

トランジスタ 2SC1815Y

NPN トランジスタのうちの一つで、ベース-エミッタ間の電流の約 100 倍の電流が、コレクタ-エミッタ間に流れる。また、ベース-エミッタ間には 0.6V 以上でなければ電気が流れないようになっており、リレースイッチの代わりに利用できる。

フォトトランジスタ NJL7502L

光信号によって電流を制御するトランジスタ。ベース端子はなく、二本の端子のうち長い方がコレクタ、短い端子がエミッタである。光線信号がベース電流を代用する。

LED

発光ダイオードとも呼ばれる。順方向に電圧を加えると発光する半導体素子。２つの端子のうち長い方が＋端子。

抵抗 220Ω

電気を流しにくくする。この抵抗は、LED に流れる電気を調節する。

抵抗 4.7kΩ

電気を流しにくくする。この抵抗は、トランジスタ 2SC1815Y へ流れる電気を調節する。

抵抗 56kΩ

電気を流しにくくする。この抵抗は、トランジスタ 2SC1815Y、フォトトランジスタ NJL7502L へ流れる電気を調節する。

その他の部品の説明は省略

この回路の仕組み

・ 暗いとき

暗いとき、フォトトランジスタのコレクタ-エミッタ間にはほとんど電気が流れないので、ここでは電流は流れていないものとする。すると電流は、56kΩと 4.7kΩの抵抗を通してトランジスタのベース-エミッタ間に流れ、その約 100 倍までの電流がトランジスタのコレクタ-エミッタ間に流れる事ができる。それによって抵抗 220Ωを通して LED に電流が流れ、点灯する。

- 明るいとき

明るいとき、フォトトランジスタは暗い時に比べより多くの電流がコレクタ-エミッタ間に流れる。この電流がある程度以上のとき、並列につながっているトランジスタのベース-エミッタ間に電流が流れないほど（0.6V 以下）になることで、コレクタ-エミッタ間に電流が流れなくなり、LED は消灯する。

コメント

今年でこの電子工作体験は 4 年目を迎えた。僕が担当するのは、今年で 3 年目であるが毎年違う回路にする事を一つの目標にしている（数年以内に過去の回路に戻る可能性あり）。実は、この手の回路はフォトトランジスタより CdS セルを使う事が多い（と思う）。ところがその回路は、2015 年度の電子工作体験（第 1 回）で既に採用してしまっているうえ、近所の某有名私立高校の電子工作体験の今年の回路がどうもその回路なのではないかと思われ、あえてフォトトランジスタを使う回路を採用した。
当初、回路を試作する段階で間違えて赤外フォトトランジスタを使っていたというミスから大幅に予定に遅れが生じていたが、こうして間に合って良かったと思う。
ちなみに、CdS セルは、光の強さによって抵抗値の変化する素子である。

展示のみだった作品の紹介

・VR 的当て(77th 伊藤)

VR を用いた没入性の高い弓矢的当てゲーム
諸事情により文化祭に参加できないため、やむなく展示企画に。

・テルミン(78th 越智)

手を触れることなく演奏する電子楽器
2 つの発振器の周波数を 0.3MHz から 1.1MHz にまで引き上げるなど回路を作り替えることで音域を飛躍的に広げた。
昨年と内容がかぶるため、パンフレットでの解説は割愛。

2018 年度

部員紹介

計 25 人

期	学年	役職	名前	コメント
		顧問	溝内　千尋先生	*Special Thanks*
		副顧問	中村　優先生	
77	高２	部長	伊藤　大貴	PIC はじめました。
78	高１		阿比留　波音	(-_-)/~~~ピシー!ピシー!
78	高１	副部長	越智　一稀	物理部は地上の楽園。僕が最高指導者。
78	高１	会計	出原　健太郎	へんじがない。ただのしかばねのようだ。
79	中３		奥村　迅	好きな食べ物はサンドイッチです。
79	中３		久戸瀬　太一	来年から本気出す
79	中３		馬場　識有	気分屋
79	中３		森　壮太	403 Forbidden - コメントの取得に失敗しました
80	中２		岩佐　高志	_(´ｍ`」 ∠)_
80	中２		加古　隆井	バックアップって大切だね。
80	中２		亀山　航雅	次はもっとすごいのを作ります。
80	中２		楠田　力基	来年は何をしよう
80	中２		曽我部　京佑	まともじゃないです。
80	中２		高橋　佑典	ヤバい
80	中２		中村　勇貴	煙草を「ケムリクサ」と読みたがる人
80	中２		名定　玖晃	ゲームが大好きです。
80	中２		藤原　旺祐	403 Forbidden - コメントの取得に失敗しました
80	中２		藤原　康太	部活あんまり行ってないです。
81	中１		佐藤　隆太郎	403 Forbidden - コメントの取得に失敗しました
81	中１		佐野　巧磨	403 Forbidden - コメントの取得に失敗しました
81	中１		芝　倫太郎	緩いです
81	中１		神農　大喜	403 Forbidden - コメントの取得に失敗しました
81	中１		土井　柾人	犬と猫なら猫派。
81	中１		菱矢　遥生	何も思いつかない。
81	中１		深尾　唯斗	よく木槌を持ち歩く