

コマンド自動入力アケコンの自作



東京理科大学 I 部無線研究部
坂本 悟

1. はじめに

こんにちは、JO4FDC の坂本です。みなさんは CAPCOM が開発した格闘ゲームの金字塔、ストリートファイターシリーズをご存知でしょうか？その歴史は長く、1987 年のアーケード型から始まり、今では 6 シリーズ目となるストリートファイター6 が PC やプレステで遊ぶことができます。そんなストリートファイターシリーズですが、昔から一貫して変わっていない、初心者苦しめてきた機能があります。それが「コマンド入力」です。コマンド入力とは、特定の入力操作をすることにより必殺技を繰り出すことができる機能です。後ろ長押しからパンチを出すだけで必殺技が出せるキャラから、十字キーを一回転入力してからパンチを出さなければ必殺技が出ないといったような無茶苦茶なキャラまでいます。コマンド入力はとにかく難しいのです。そんなコマンド入力を持ち合わせの技術でなんとかしていこうというのが本プロジェクトとなります。

2. 制作

今回僕が目をつけたのは、コマンド入力の難易度が中難易度、といっても慣れるまで時間がかかるキャラである「リュウ」と「ケン」です。リュウとケンには波動拳と昇龍拳という技を出すことができます。ハードはスーパーファミコン（以下 SFC）、ソフトは多分どれでも変わらないと思います。波動拳と昇龍拳のコマンドは次の通りにボタンの入力をする必要があります。

波動拳 ↓↘→ (X または Y)

昇龍拳 →↓↘ (X または Y)

入力の猶予フレームが 4 フレーム（多分。スケッチ書いてる途中で気付きました。）なので指を十字キー上で素早く滑らせる必要があります、長時間プレイするとすご〜く指が疲れます。たまたま家に Arduino があったので、これを使えば自動入力できるんじゃないか？！と思い、早速部品を調達し、制作に取り掛かりました。

部品

- | | |
|-------------------|----------------|
| • Arduino UNO | • ブラケット入 LED |
| • ジョイスティックレバー | • トグルスイッチ |
| • プッシュスイッチ各種 | • 筐体（工具箱） |
| • トランジスタ（2SC1815） | • 9V 電池&電池ボックス |
| • 抵抗器（30Ω, 10kΩ） | |
| • ユニバーサル基板 | |

2.1 スケッチ

まずはプログラミング（もどき）から。C 言語なんて知ったこっちゃないのでネットで調べて試行錯誤しました。こちらがそのスケッチになります（図 1）。

```
1 void setup(){
2   pinMode(13, OUTPUT); //DOWN
3   pinMode(12, OUTPUT); //RIGHT
4   pinMode(10, OUTPUT); //Y パンチ
5   pinMode(7, INPUT_PULLUP);
6   pinMode(6, INPUT_PULLUP);
7
8 }
9
10 void loop(){
11
12   if(digitalRead(7) == LOW){
13     digitalWrite(13, HIGH); //下入力
14     delay(70);
15     digitalWrite(12, HIGH); //右入力
16     delay(70);
17     digitalWrite(13, LOW); //下入力終了
18     delay(70);
19     digitalWrite(10, HIGH); //パンチ入力
20     delay(70);
21     digitalWrite(12, LOW); //右入力終了
22     digitalWrite(10, LOW); //パンチ入力終了
23     delay(100);
24
25   }else if(digitalRead(6) == LOW){
26     digitalWrite(12, HIGH); //右入力
27     delay(70);
28     digitalWrite(12, LOW); //右入力終了
29     digitalWrite(13, HIGH); //下入力
30     delay(70);
31     digitalWrite(12, HIGH); //右入力
32     delay(70);
33     digitalWrite(10, HIGH); //パンチ入力
34     delay(70);
35     digitalWrite(10, LOW);
36     digitalWrite(12, LOW);
37     digitalWrite(13, LOW);
38     delay(100);
39
40   }else{
41     delay(70);
42   }
43   delay(10);
44 }
```

void setup(){} の項では Arduino のピンの設定をしています。13 番ピンを下入力、12 番ピンを右入力、10 番ピンを Y 入力に設定。7 番、6 番ピンは入力待機をする設定をしました。

Void loop(){} の項では、「ボタンが押されたらコマンド入力を実行する」というプログラムが書かれています。テスト段階ではどれくらいの時間間隔で入力を行えば高精度で必殺技が出せるのだろうと試行錯誤していました。その結果、70 ミリ秒間隔でボタン入力をすれば良いという結論に至りました。

なんかこう、もう少しきれいに書けないのけね...

図 1 Arduino に書き込んだスケッチ

2.2 スイッチング回路

2.1 で書いたスケッチだけではコントローラーの導通を確保することはできません。**Arduino** から出力される電圧を元にスイッチングをする必要があります。リレーを使うか、トランジスタを使うか、はたまたフォトカプラを使うか、手法はたくさんあります。今回選んだのはトランジスタを用いたスイッチング回路です。コマンドが出されるたびにガチャガチャ音が鳴るのもなんだかなぁと思い、トランジスタを選びました。以下に回路図を示します。(図 2)

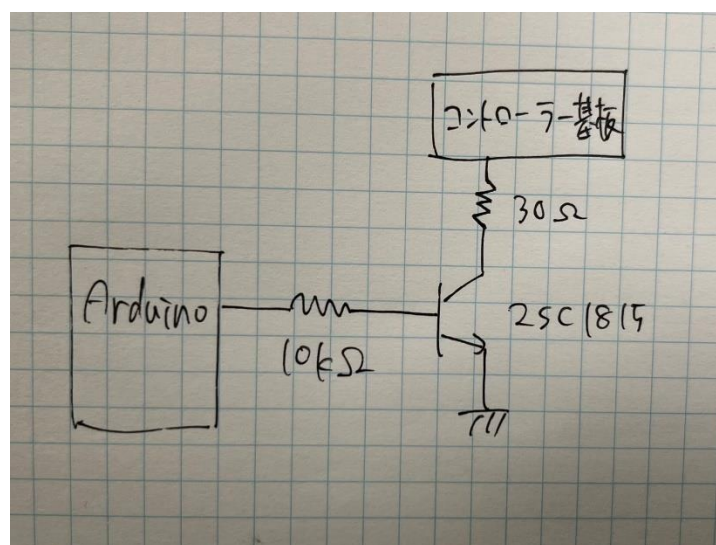


図 2 スイッチング回路の回路図

コントローラーの基板からは常時コレクタに電圧がかかっており、ベースに電圧がかかったらコントローラーの導通が確保できるといった仕組みになっています。多分。コレクタにつながっている 30Ω はコントローラーのボタンが押されたと判定される(?) 抵抗です(図 3 参照)。



図 3 ボタンの抵抗値を計測している様子

2.3 筐体加工

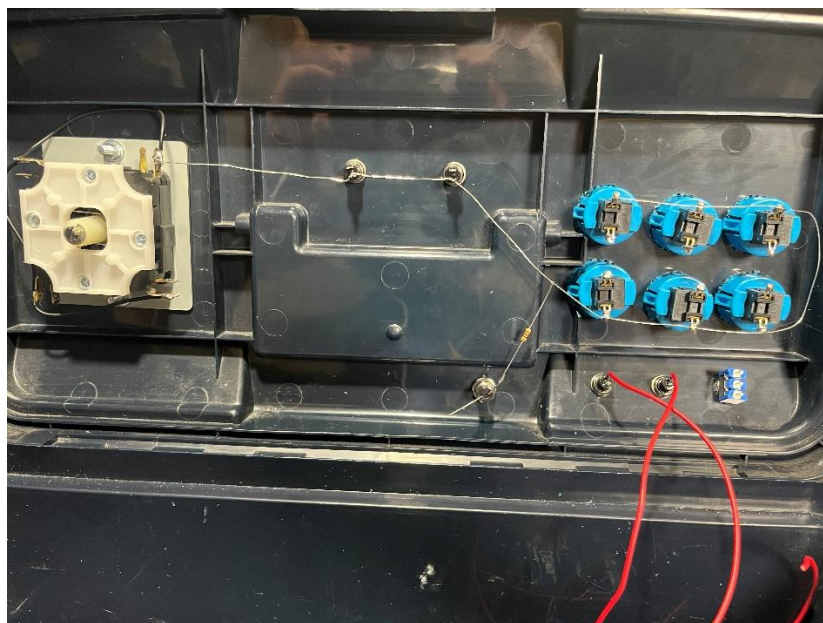


図 4 部品取り付けの様子

工具箱に穴を開けてジョイスティックレバー、プッシュスイッチ、LED、トグルスイッチを取り付けた様子です。この辺は適当に、自分が使いやすいようにしました。ここで見えている配線は GND です。あと、コントローラー基板、Arduino、諸々を取り付けるためのネジ穴も底面にあけておきました。

2.4 全体像を掴む

さてさて、アケコンを完成させるにあたって必要になりそうな要素は揃ってきました。組み上げる前にここで一度全体像を確認しましょう。

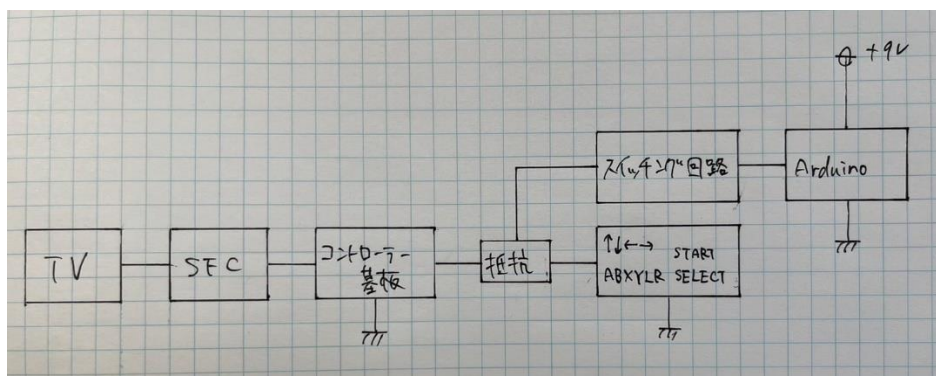


図 5 系統図

系統図は完成させた後に書いたものですが、中身は多分こんな感じになっています。筐体に取り付けたボタンやレバーから直接抵抗器を伸ばして空中配線するのは、短絡の恐れがあるので一つの基板の上に集約しました（図5中“抵抗”）

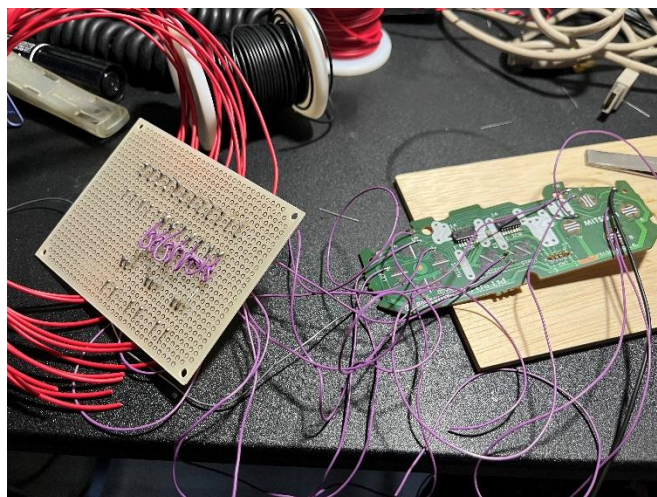


図6 “抵抗”基板にコントローラー基板をつなげる様子

2.5 組み上げ

お疲れ様でした、あとは組み上げるだけです。部品を壊さないよう注意して組み上げましょう。

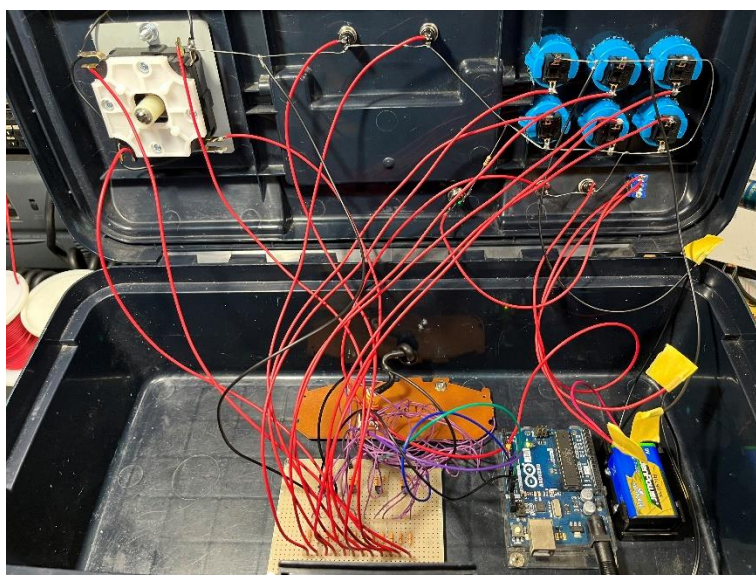


図7 完成

3. 終わりに

いかがだったでしょうか？当方、電気の「で」の字も知らない化学徒ですが、ほんの少しの好奇心とインターネットがあればこんなものが作れてしまうのです。スイッチング回路の項では偉そうに動作原理らしきものを語っていますが、まっっっったく理解なんてしていません（笑）。実際プレイしてみるとわかるのですが、結局スト 2 の強さというのは必殺技をいかに上手に出せるか、ではなく基礎的な技をいかに正確に出せるか、ということに尽きるのです。現にこのアケコンは 1P しか対応しておらず、2P 側に回った途端必殺技ボタンが使えなくなってしまうという致命的な欠陥があるのです。ゲームのスキルというのは技術で解決されるものではなく、努力あってこそそのモノなのだと痛感させられる工作でした。興味がある方はぜひ作ってみてください！