

## 電気自動車の制御研究

システム科学技術学部 知能メカトロニクス学科  
1年 B20N015 奥住 公祐  
システム科学技術学部 機械工学科  
1年 B20M013 小野塚 陸  
指導教員 システム科学技術学部 知能メカトロニクス学科  
准教授 戸花 照雄

### 1. 目的

現代社会において、私たちの生活は自動車や電化製品、産業機械によって成り立っている。これらの製品の多くにはマイコン(マイクロコンピュータ, マイクロコントローラ)が入っており、このマイコンによって機械は電氣的に制御され動いている。そこで、私たちは自分でマイコンを用いて機械、特に電気自動車を制御したいと思い、本研究を始めた。

### 2. 使用素子と原理, 要求される知識

#### 2.1. PIC マイコン

PICとは、Peripheral Interface Controllerの略称で、マクロチップ・テクノロジー社が製造しているマイクロコントローラ(制御用IC, 略称:マイコン)製品群の総称である。今回用いたPICはピンが計28本あるPIC18F2320である。PIC誕生当初は、使用可能な言語はアセンブラのみであった。しかしその後、MPLAB Xという統合開発環境の出現により、現在はC言語およびC++言語でも使用可能になっている。そこで私たちは、可読性が高くプログラム容易なC言語を用いてPICのプログラムを作成することにした。また、PICは回路構成が容易かつ安価であり、ほかのマイクロコントローラと比べて書籍やインターネット上の情報が充実しているといった利点もある。

#### 2.2. LCD モジュール

LCDとは、Liquid Crystal Displayの略で、液晶ディスプレイのことである。PICからの出力により扱えるモジュールとして、今回はSunlike Display Tech社のSC1602BS\*Bを採用した。昨年度の自主研究において使われていることから、その使い方を踏襲することでプログラムと電子配線を両方を学習できることからこのモジュールを採用した。このLCDモジュールはPICkit3から供給される電力を電源として、文字列を表示することが可能である。

#### 2.3. LED

PICから直接供給される電力で利用可能であり、配線が直列的で簡潔にでき、プログラムが容易、といった特徴があり制御しやすい素子である。PICが正しく動作しているか、プログラムに不具合がないか、をLEDを発光させることにより判断するために用いた。

#### 2.4. DC モータ (直流モータ)

LCDモジュールとLEDの配線、制御が成功したのちに電気自動車の駆動制御の理解を深めようと考え採用した。なお、この素子はPICやPICkit3からの電力供給では駆動するのに必要なエネルギーが足りないため、外部電源を必要とする。また、この素子はPICでは直接ON/OFF制御することができないので、トランジスタを介して回路レベルで制御する。

#### 2.5. 要求される知識

マイコンやトランジスタなどの電子素子の配線、C言語およびC++によるプログラミング、開発環境(MPLAB X)の構築

### 3. 研究内容

#### 3.1. 使用機器・ソフト

マイコン(PIC18F2320), LCDモジュール(SC1602BS\*B), LED(赤色, 黄色), DCモータ, PICプログラム(PICkit3), 抵抗やトランジスタなどの電子素子, テスター, はんだごて等の工作機器, MPLAB X IDE, BSch3V

#### 3.2. マイコン制御機械の基板の製作

昨年度の電子時計実機および書籍を参考にしながら、各素子をはんだ付けしながら配線して

いった. 何度も試行錯誤と修正を繰り返しながら最終的に実機が完成した. 今回は実際に回路の配線を行うことにより回路の理解を深めた後に回路図を作成したため, 通常と作業工程が逆転している. 今回作った回路基板の写真を図1, 回路図を図2に示す.

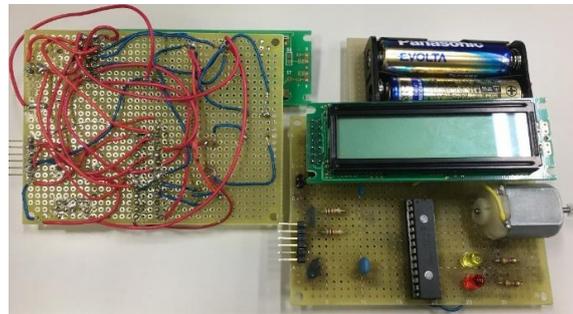


図1. 実機の回路基板の配線の様子

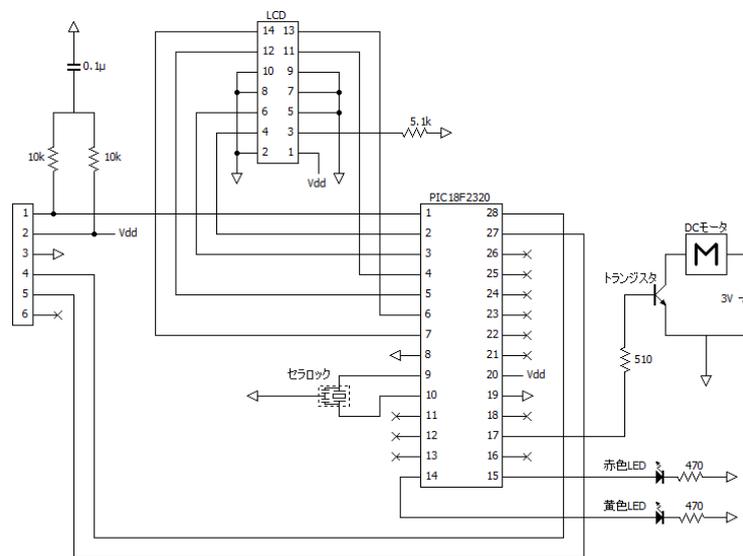


図2. マイコン制御式機械の回路図

### 3.3. C言語によるプログラミング

実機の回路の配線が完了後, プログラムの作成した. 私たちはPICやLCDモジュールなどの出力素子, またC言語プログラミングの知識が十分でなかったため, 本研究室において昨年度の自主研究で作成したプログラムを元にして, 中身を書き換える形でプログラミングを行った. 以下の動作をするプログラムをC言語により作成し, PICKit3を用いてPIC18F2320に書き込み, それぞれの動作を確認する.

#### 3.3.1. LEDの点滅

2.3節で書いたように, 電子素子の中でLEDは比較的プログラムで制御しやすいため, PICのピンに正しく配線することで, 比較的早い段階で点滅させることに成功した. PICから直接LEDに電流を流す, 止めるといった制御を行い, 2つあるLEDを一定のタイミングで点滅させた. これは以降の実験でプログラムおよびPICが動作しているか確認するために使用する. 秒カウンタに合わせて1秒ごとに点滅するようにした.

#### 3.3.2. LCDモジュール上における文字の表示

PICからの信号で指定された文字列を表示する. 電力供給はPICKit3からピンヘッドを介して行う. 今回使用したLCDモジュールのハードウェアインターフェースは, ピン配置とパルス幅に注意すれば正常に動作する. ディスプレイ光度は電流の大きさに制御されるが, 基本的にPICKit3からの供給電圧は固定されているため(今回は5Vに設定), 電力供給回路に直列に抵抗を挿入することで, 電流の規格範囲内で明るさを調整することが可能である. 実際に行った制御は, まずPIC上に残る以前の文字情報を初期化し, 表示させたい半角英数字・記号をア

スキーコードで指定してから PIC に送る. その後, PIC から LCD モジュールあてに文字コードの信号を送るといったものである.

### 3.3.3. DC モータの動作

LED と LCD モジュールの配線・制御が成功した後に追加したこの DC モータは昨年度の自主研究で使われていない素子であり, 指導教官による指導の元で制御プログラムを作成した. しかし 2.4 節で述べたように, PIC や PICkit3 からの出力回路では, DC モータを駆動させるために必要な電流量が不足している. そこで DC モータを外部電源 (今回は単 3 乾電池 2 本を直列につないだ 3V 電源) につなぎ, DC モータと外部電源の回路の間にあるトランジスタを PIC からの信号で ON/OFF 制御することで制御した.

### 3.3.4. 制御フローチャート

実際に作成したプログラムによって実機を制御したプロセスを, フローチャートで下記に示す.

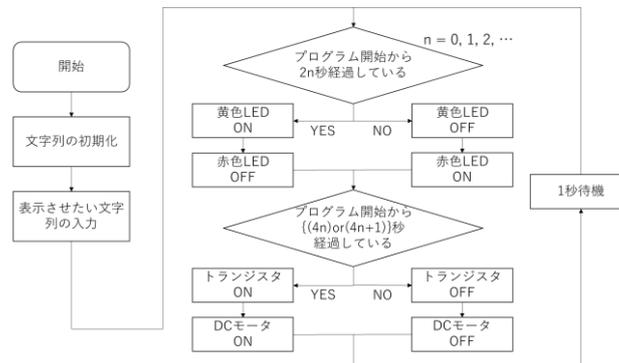


図 3. マイコン制御式機械のフローチャート

### 3.3.5. MPLAB X でのプログラム実装画面

以下に実装したプログラムと使用した開発環境 (MPLAB X) の写真を載せる. この画面以外に書かれているコードのほとんどは指導教官が作成したものであるが, このコードが記載されている main 関数内の処理は自分たちで作成した.

```

178 while(1){
179
180     LCD_init();
181     LCD_write(1h);
182     LCD_write(2h);
183     LCD_write(3h);
184     LCD_write(4h);
185     LCD_write(5h);
186     LCD_write(6h);
187     LCD_write(7h);
188     LCD_write(8h);
189
190     /* LED制御
191     * 初期化 control の時
192     * 最初だけ黄色LED, 赤色LED
193     * 以降は黄色LED, 赤色LED*/
194     int ic = 0;
195     ic = 2;
196     if(ic == 0){
197         PORTbits.RC0 = 1;
198         PORTbits.RC4 = 0;
199     }
200     else{
201         PORTbits.RC0 = 0;
202         PORTbits.RC4 = 1;
203     }
204 }

```

図 4. 使用した開発環境と記述したコードの一部

## 4. 結果・考察

昨年度の自主研究で作られた電子時計の実機をもとに自分たちで配線を行い, プログラムを書き込んだが, はじめは動作しなかった. 結果と結果に対しての考察を下記に記す.

(1) LED, LCD モジュール両方が動作しなかった.

原因: プログラムコードは昨年度使用されたものを用いたが, 配線は独自で参考書のサンプルを用いたため, プログラムと製作した回路が適合しなかった.

対策: プログラムと PIC プログラマのピン機能を確認・理解し, 昨年度の電子時計の配線につなぎ変えた.

(2) (1)の対策の後、LEDは動作したが、LCDモジュールは動作しなかった。

原因：プログラムコードにおいて、アナログ信号をデジタル信号に変換するための ADCON 命令文を入力しておらず、LCDモジュールへの信号入力ができなかった。

対策：昨年のコードと比較することにより、上記の原因を突き止めた。その後、アナログ信号・デジタル信号に関して理解した。また、去年の研究で作られた電子時計には可変抵抗が付けられていたが、私たちは使わなかったため、テスターを用いて液晶がはっきり映るように固定抵抗を付け替えて光度を調整した。

以上で昨年度の自主研究の結果を踏襲することができた。そこで私たちは研究を発展し、電気自動車を制御する第一歩として、DCモータを回路に追加し、プログラムを作成しモータの制御を行った。以下にその結果を補足する。

(3)追加したDCモータが動作しなかった

原因：最初に追加したモータを、PICからの電流供給で駆動させようとしていたが、PICからの出力では電流が小さすぎた。

対策：トランジスタを用いて回路を制御する方法を用いた。ここで、トランジスタについて学習し、外部電源からDCモータに供給される電流を制御する回路を作成した。なおトランジスタにはエミッタ、コネクタ、ベースの三極があるが、三極の配線を間違えたり、ベースとPICの間の抵抗を接続し忘れるなどのミスにより、DCモータの制御を成功まで時間を要した。

以上の対策の結果、すべてのデバイスが正常に動作するようになった。下記に実際に動作した本機の写真を載せる。



図5. 本機械が実際に動作している様子

## 5. まとめ

プログラミングやマイクロコンピュータ、電子配線や駆動機構などは工学の大事な要素であり、その工学は現代社会を支える重要な基盤産業の一つである。システム工学初学者である私たちは本年度、それらの知識と技術を広範囲に学ぼうとしてきた。しかし広い範囲の知識を1年という限られた期間で学ぶことは困難であり、若干の知識がついても実際に回路とプログラムを作成する段階になると、プログラムのエラーや配線のミスといったソフトとハードの小さな欠陥により機械が動かないことも多々あった。また今年度の自主研究では、残念ながら当初目的としていた電気自動車の制御まで到達しなかった。それは工学についてほとんど何も知らない状態で研究を開始したため、基礎知識の理解と技術の習得に時間を要したことが原因だと考えられる。

しかし、今回の自主研究を通じて、コンピューターを用いて機械を制御することの難しさを体験することができた。また、その体験から得られた知識や経験は非常に多かったと思う。学んだ知識と経験を生かし、次年度の自主研究や卒業研究を、より飛躍したものになりたい。

## 参考文献

- [1]後閑哲也, 改訂版電子工作のためのPIC18F 本格活用ガイド, 技術評論社, 2015
- [2]PICマイコンの基礎, 後閑 哲也, 株式会社毎日コミュニケーションズ, 2011
- [3]柴田望洋, 新明解C言語入門編, SBCreative, 2014
- [4]MMGames, 苦しんで覚えるC言語, 秀和システム, 2017