

AVR マイコンを用いた交通信号機制御

佐藤 哲也 麓 裕太
妹尾 淳暉

1. 研究概要

近年、様々なワンチップマイコンが登場しており、電子工作を行う上で身近な存在となっている。私たちはその中の1つである、AVRマイコンの活用について研究し、道路にある交通信号機模型の製作を通じて AVR マイコンについて理解を深めることを目的とした。

2. 研究内容

私たちは次の3つのことについて研究した。

- (ア) AVR についての学習と理解
- (イ) 信号機模型の製作
- (ウ) AVR マイコンプログラミング開発環境と制御プログラムの作成

1. ワンチップマイコンと AVR マイコン

ワンチップマイコンとは、IC という小さい部品の中に CPU, ROM, RAM の他、I/O ポートコントローラを内蔵しているもので、特定の動作を行うプログラムを組み込むことができ、主に電化製品の様々な制御に用いられる。その他にも自動車の燃料供給制御、携帯電話などに使われており我々の生活を便利に、快適にしてくれるものである。その中でホビーユーザーが使いやすいものとして、PIC, AVR, H8 等があげられる。

ワンチップマイコンは発売当初マスク ROM や PROM が中心であったが現在は EEPROM が主流となり大変使いやすいものとなった。また、CPU のクロック周波数や向上により大型になり、大きなメモリが自由に利用できるようになった。また 8bit だったものが 32bit まで拡張されている。その中で今回は、AVR について研究を行った。

AVR はアメリカの Atmel 社が 1997 年にアーキテクチャを発表したワンチップマイコンで、PIC と同じようにプログラムを書き込むことで制御する。AVR は PIC のライバルマイコンとして PIC の欠点を補う構成である。そのなかでも PIC より AVR の利点は主に 2 つあり、

- (1) AVR は入出力に対して完全な read (読み込み) -modify (修正) -write (書き込み) が出来ること

AVR は特定のビットにのみ ON-OFF 制御する命令があり、他のポートと関係なく独立した動作ができる。

- (2) 1 クロックで 1 つの命令を実行できること

PIC は AVR と違い 4 クロックで 1 つの命令を実行する。そのため AVR と PIC を比べると 5~8 倍差が出る。この 2 つの利点を生かし AVR での開発に取り組んだ。

現在、Web ページを見ると PIC と AVR の情報量の差は PIC のほうが歴史が長いから、日本では断然 PIC に分がある。しかし、AVR マイコンの資料が増えてきた今、PIC に負けることも劣らない性能を持っていると言えるだろう。

2. 信号機回路の製作

まず、信号機の動作を確かめるための試験回路を製作した。主な部品としては、インバーター、信号機の点灯には LED を使用した。AVR は ATtiny2313 を使い、マイコンの ON-OFF 制御で道路にある交差点が再現できるようにプログラミングをしていく。



図 1 交通信号機実験回路

図 1 に示すように、右上に AVR、左側の LED は上にある 4 つが歩行者用、その他が普通自動車用信号である。

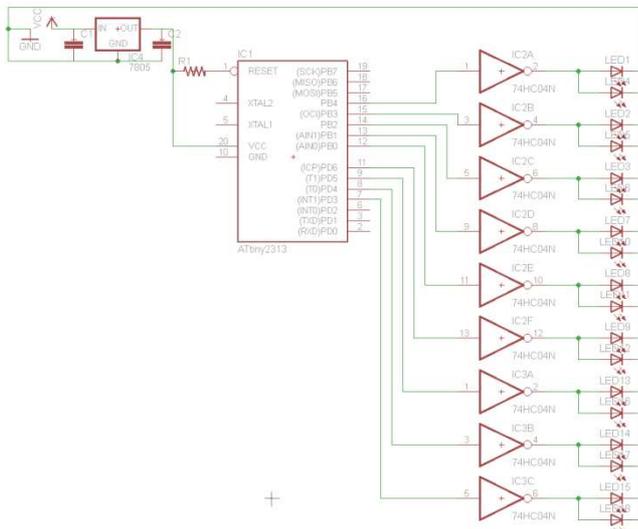


図 2 回路図

3. 開発環境

AVR マイコンのプログラムを作成する方法として一般的には AVR Atmel mkII ライタを使用し AVR Studio というソフトでプログラムを作成する。

しかし今回は、なるべく簡単に行うために、ja_hidapio 通称「ヒダピオ学習回路」を用いてプログラミングを行った。このプログラミングツールは中学校の技術家庭科の制御・計測教材として主に利用されており、株式会社 トップマンより販売されている教材のライタをつかってプログラミングを行う。

この「ヒダピオ学習回路」を使ったのは、今回私たちがはじめてプログラミング環境を開発することにあたり、初めてでもわかりやすいものにしようと考えたとき、この「ヒダピオ学習回路」を発見した。またこのソフトに対応したライタが自作できると分かり、ぜひ利用してみようと考えた。

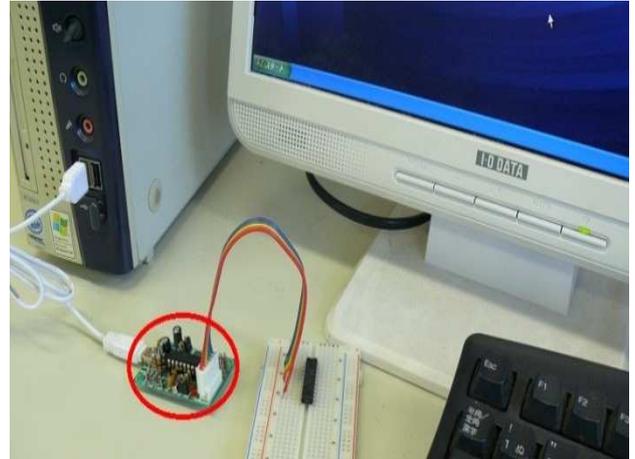


図 3 HIDaspX と書き込みの様子

このライタは HIDaspX といい、HIDaspX の HID はヒューマンインターフェースデバイスの略で、付属の USB ケーブルをパソコンにつなげるだけでドライバをインストールしなくても、コンピュータが自動認識することで、ja_hidapio から AVR に書き込むことができる。

このライタはライタ動作用に HIDaspX のファームウェアを書き込んだ AVR をライタ回路に接続している。つまりライタ動作させるために AVR を使用する。図 4 の回路図でライタと書き込む用の AVR をつないでいる。

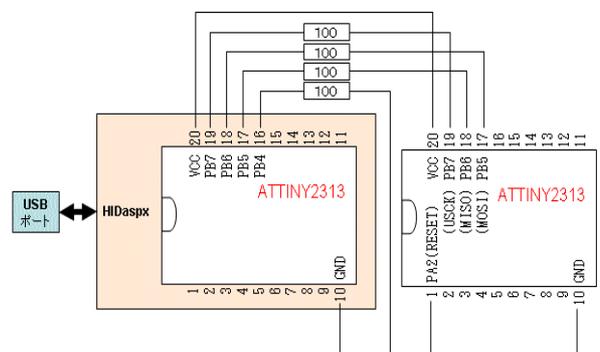


図 4 HIDaspX 接続回路図

図 3 に示したように今回は回路図の右側の AVR はブレッドボードに接続した。

4. プログラム



図5 ヒダピオ学習回路

この簡易プログラムによるプログラム作成手順を説明する。



図6 簡易プログラム入力画面

図6に示すところに、プログラムを入力していく。

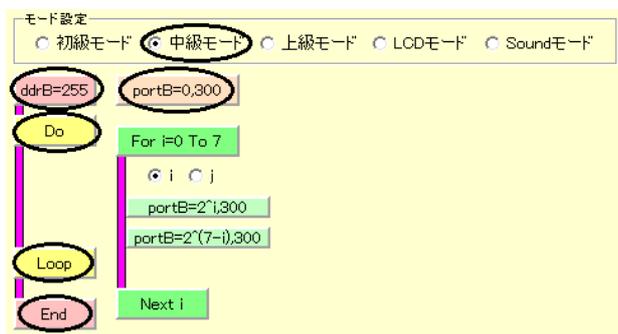


図7 GUI入力フォーム (中級モード)

図7はプログラムに必要な要素を選択するものである。今回は、中級モードの丸で囲んだ部分の要素を使った。この図の上部にあるモード設定を選ぶとレベルによって使用できる命令アイコンが表示される。次にプログラムの命令アイコンをクリックしていくと、対

応したプログラム命令が図8のようにプログラムエリアに挿入されることで、プログラムを作成していく。



図8 プログラム挿入

図7で選択した要素は、上記のように上のテキストボックスに格納される。そこで必要な修正を加え、追加・挿入することでプログラムを書いていく。ちなみに、このソフトの使用言語はBASICで初心者にもわかりやすいものとなっている。プログラム完成後、図9にあるファイル名を指定し、「コンパイラへ」をクリック。コンパイルと、指定したbasファイルを「bascomp1.bas」として保存する。



図9 コンパイラ部 (20個まで保存可能)



図10 basファイル保存完了画面

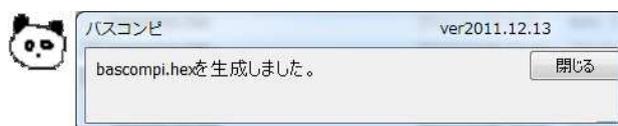


図11 バスコンピ実行

同じ ja_hidapio の中に付属されているバ

スコンビというソフトをダブルクリックし、先ほどの bas ファイルを hex ファイルに変換する。(この hex ファイルを AVR に書き込む。)



図 12 HAG_mini (書き込みソフト)

AVR への書き込み方法はまず、HIDaspx が USB ポートに接続されているかを確認し「読み出し」ボタンをクリック。このとき、デバイス名に「ATtiny2313」と表示されれば良い。その後、プログラム欄に書き込む hex ファイルを指定する。(「...」は参照ボタンである。)それから「プログラム書き込み」ボタンをクリックすれば AVR に書き込める。この書き込んだ AVR を図 1 の信号機回路にセットして動作を確認する。

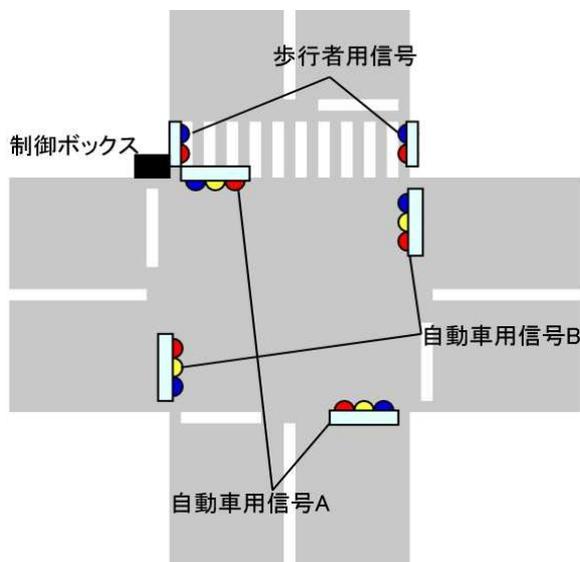


図 13 今回再現した交差点のイメージ
実際の簡易プログラム例

```
ddrB=255
```

```
Do
```

```
portB=9, 1000
```

```
portB=33, 4000
```

```
portB=17, 2000
```

```
portB=9, 1000
```

```
portB=12, 4000
```

```
portB=10, 2000
```

```
Loop
```

```
End
```

説明 portB=数 1, 数 2 とは

数 1

AVR の PB0 から PB7 を使い, PB0 を最下位ビットとして 10 進数で表している。

数 2

表示させる時間を ms (ミリ秒) であらわす。

その動作条件を表などに書き出し, 点灯点滅を行うよう LED につながる端子に 1 を設定する。

3. 研究のまとめ

ワンチップマイコンである AVR について理解を深め, その AVR を使ったのものづくりの 1 つとして, ごく一般的な交差点を今回は AVR で実現させることを目標として研究を行った。この交通信号機はシーケンス制御といい, LED が点灯する順番, 時間を制御するものである。それを今まで使ったことのない AVR を用いて書き込み環境を開発し, 制御を行った。

今回構築した開発環境は電子工作をやったことのない人でも簡単にあつかうことのできるものである。この研究で学んだことや, ものづくりの大切さ, 楽しさを電子工作のやったことのない人に伝えていければと思う。

参考文献

ヒダピオ学習回路

<http://hidapio.jp/>

いまして PIC をやめて

AVR に移行すべき 10 の理由

<http://d.hatena.ne.jp/yaneurao/20080228>

交通安全教室用信号機

<http://siva.cc.hirosaki-u.ac.jp/usr/koyama/signal/>

株式会社 トップマン

<http://www.topman.co.jp/>

1 2月7日（金）～15：00提出

文書の清書はWordを使用し、紙に出力したものと電子ファイルの両方を提出

1月22日（火）

情報技術科課題研究発表会

（書式）

余白 上下左右全て20mm

段組 2段組，段幅80mm

間隔 10mm

書体 10.5ポイント，MS明朝体

段落 20文字×40行

サイズ 英数字は原則として半角

1桁の数字の場合は全角

半角カナは使用しない

文字 機種依存文字は使用しない。

株，①，②，③，ミリなど

写真 解像度150dpi以上。

図 図記号（図の下側）

表 表番（表の上側）

写真 写真番号（写真の下側）

語尾 「～である。」，「～した。」

「1. 研究概要」

「2. 研究の具体的内容」

「3. 研究のまとめ」

「参考文献」

の各項目名はそのまま使用する。