

3E コンピュータ基礎 H8マイコンを使ったアセンブラ演習

1. 目的

実際にアセンブラでプログラムを作成し、授業内容の理解を深める。

2. 今回利用するH8マイコンについて

秋月電子通商の3664タイナーマイコンボードセット(キット)を利用する。搭載されているマイコンはH8/3664Fである。教科書で説明されているマイコンH8/3048Fとは型番は異なるが、コアとなるCPUには同じH8/300Hが使用されている。3048Fと3664Fの比較を行うと表1のようになる。

表1 3048Fと3664Fの性能比較

	3048F	3664F
フラッシュROM	128kB	32kB
RAM	4kB	2kB
総ピン数	100	42
入出力ポート	11個(78ピン)	6個(31ピン)

図1に今回使用するマイコンボードの外観を示す。図1では配線などで実際に使う部分のみ表している。CN1、CN2側にそれぞれ26ピンずつ端子があるが、ブレッドボード上で動作させるための構造的な都合上、図2のようにしている。

- ・内側の偶数番号のピン:メス型ソケットになっており、ジャンパー線を差して使用する。
- ・外側の奇数番号のピン:オス型ソケットをブレッドボードに差して使用する。

SW1とSW2はパソコンからH8マイコンにプログラムを書き込む場合にはジャンパーピンを差す必要があります。このとき、図3の禁止事項を必ず守ること。これを間違えると回路が短絡(ショート)して、素子が破損する場合があります。

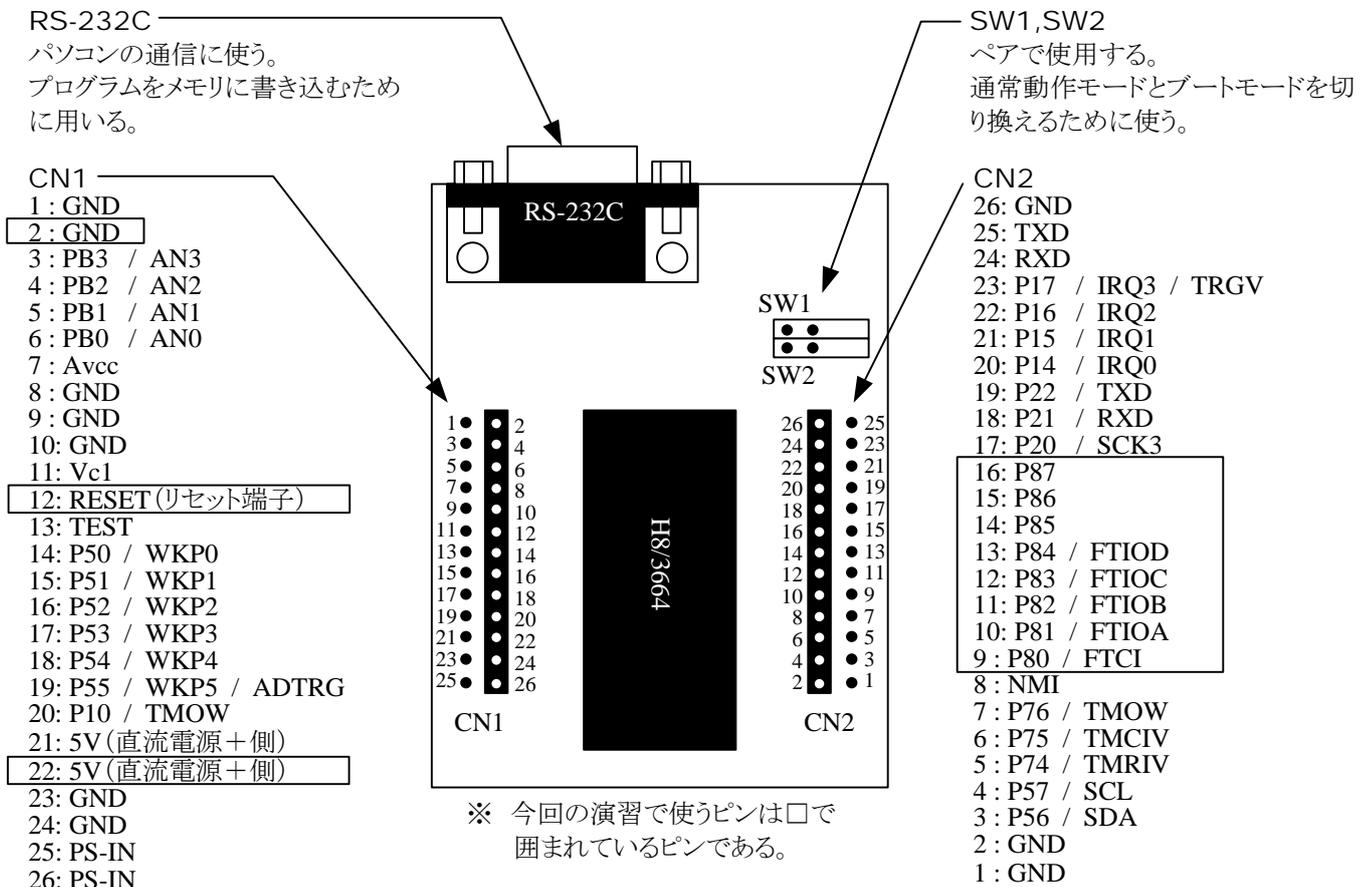


図1 マイコンボードの外観

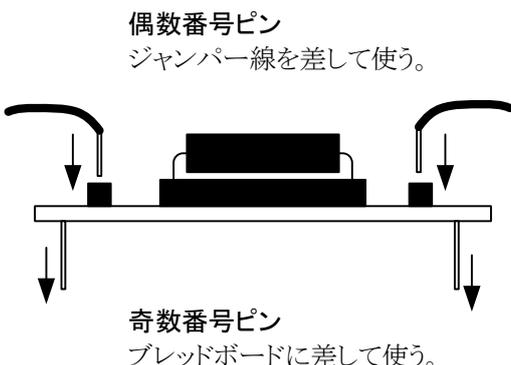


図2 端子の位置

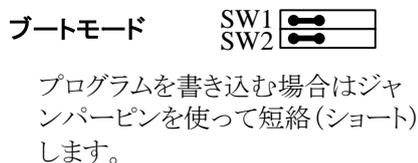
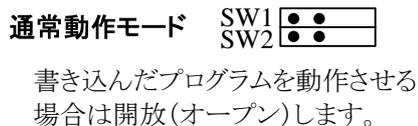


図3 SW1、SW2の使い方

禁止事項



ジャンパーピンを間違っていると電源とGNDがショートして素子を破損します。

3. 回路製作と動作確認

今回使用するH8マイコンにはすでにLEDを点滅させるプログラムを書き込んでいる。まずは回路を製作して動作させてみましょう。使用する部品などは以下の通りです。

- ・ブレッドボード+H8マイコンボードキット(マイコンボードはブレッドボードに実装済である。抜かないこと。)
- ・ACアダプタ(5V)または直流安定化電源(5Vで使用する。)
- ・ジャンパー線セット
- ・抵抗 3.3kΩ 4本
- ・LED 4個
- ・タクトスイッチ 1個

(1) 電源とGNDラインの製作

図4のようにブレッドボードに電源ライン(赤)とGNDライン(青)を構成する。電源ラインは+5Vを供給する。Va、Vb、Vc端子のいずれかをACアダプタまたは直流安定化電源の+側を接続する端子とすること(図ではVcとしている)。

図4のようにジャンパー線を接続すれば、全ての赤ラインに+5Vが供給され、青ラインがGNDになっている。

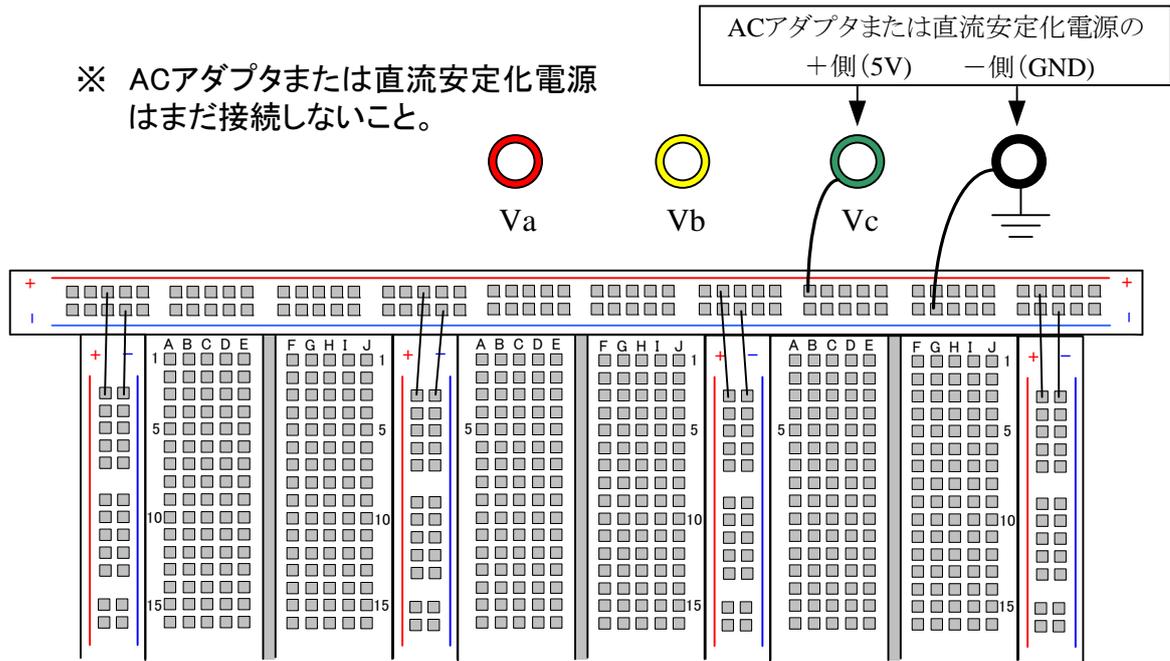


図4 電源ラインとGNDライン

(2) マイコンへの接続

図5のように回路を構成する。GNDに接続するピンはGNDであればどこでもよい(例えば、1番ピンを使ってもよい)。9番ピンはブレッドボードに差さっているので、ブレッドボードから接続すること。

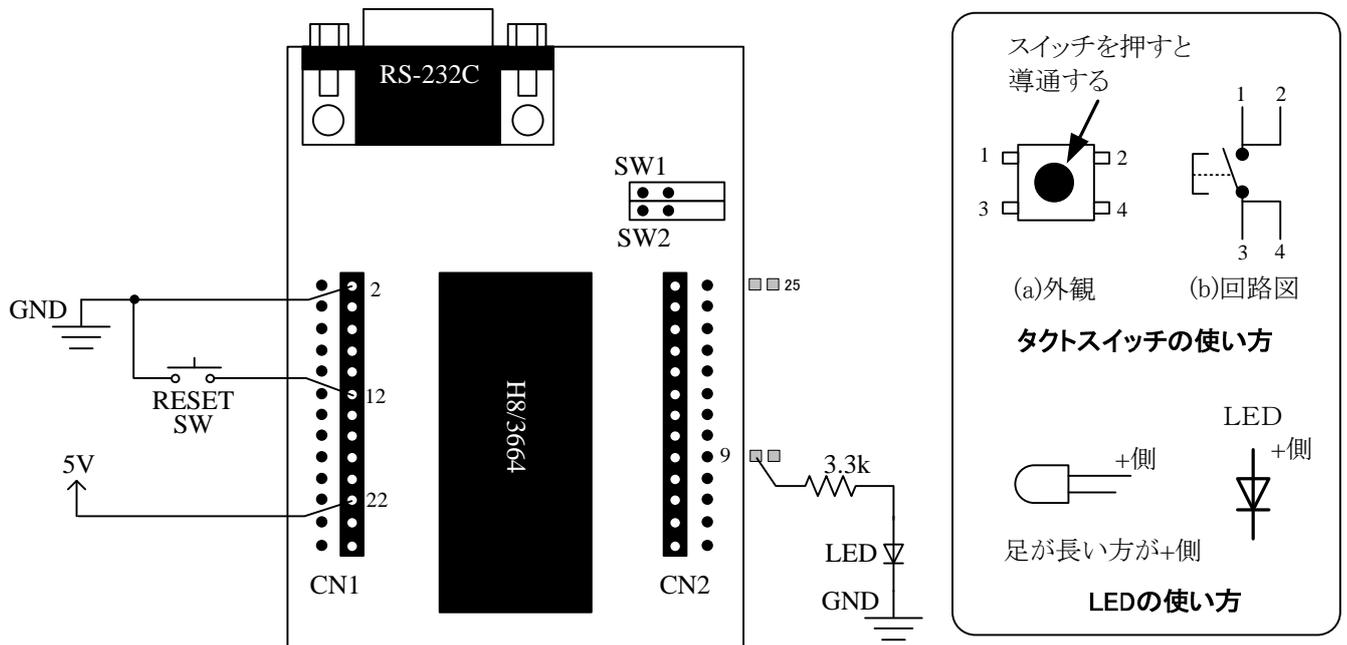


図5 回路図

(3) SW1とSW2の設定

H8マイコンを動作させるためにはSW1とSW2を開放(ジャンパーピンを外す)して**通常動作モード**にする。間違わないように図3を再度示すので参考にする。

(4) 直流電源の接続

ACアダプタまたは直流安定化電源をブレッドボードに接続して**電源をON**にする。きちんと接続されていればLEDが点滅するはずである。

通常動作モード SW1  SW2 

書き込んだプログラムを動作させる場合は開放(オープン)します。

ブートモード SW1  SW2 

プログラムを書き込む場合はジャンパーピンを使って短絡(ショート)します。

禁止事項

 SW1 SW2

ジャンパーピンを間違っ
て図のように使うと電源と
GNDがショートして素子を
破損します。

図3 SW1、SW2の使い方

4. プログラムのH8マイコンへの書き込み

回路の動作が確認できたら次にH8マイコンへプログラムの書き込みを行う。以下の話はWindowsでパソコンが起動していることが前提となる。既にC:\¥H8フォルダにサンプルプログラムが置いてあるので、これを利用してプログラムの転送方法を説明する。

以下の内容は山本先生のホームページから転載している。山本先生に謝意を表するとともに、学生のみなさんは山本先生のホームページも参考にする。

(1) ファイルの確認

マイコンコンピュータなどを利用してC:\¥H8フォルダにサンプルプログラムsample.asmがあることを確認すること。プログラムの内容はテキストエディタ(例えばEmEditorなど)で確認することができる。

(2) コマンドプロンプトの起動

[スタート]→[ファイル名を指定して実行]をクリックして(図6)、「ファイル名を指定して実行」ウィンドウを起動する(図7)。図7のように「名前:」の欄に“cmd”と入力し、OKボタンをクリックすると図8のコマンドプロンプトが起動する。プログラムの転送はコマンドプロンプトから行う。

コマンドプロンプト上での操作例を参考にして、コマンドプロンプト上でフォルダをC:\¥H8>に移動すること。



図6

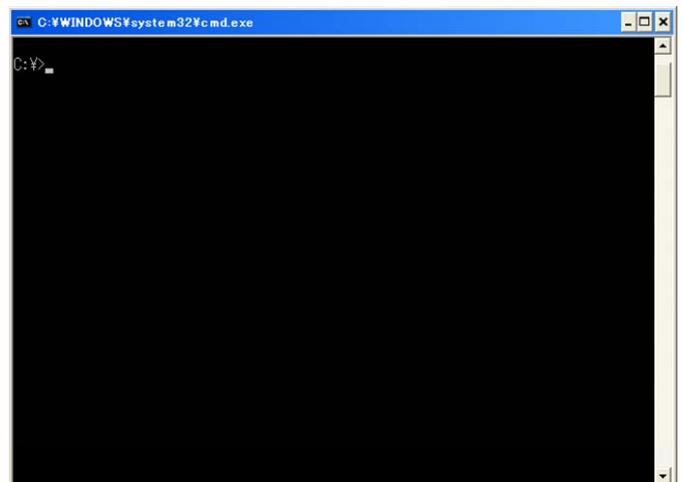


図8 コマンドプロンプト画面

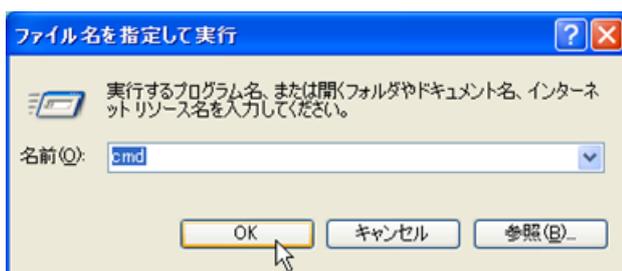


図7

コマンドプロンプト上での操作例

- C:\¥>にする方法
U:\¥>c: ←
- フォルダを移動する方法
C:\¥>cd H8 ←
とすると、C:\¥H8> に移動する。
ちなみに、C:\¥H8>cd .. ←
とすると、一つ上のフォルダに戻る。
- フォルダ内のファイルの表示方法
C:\¥H8>dir ←

(3) プログラムのアセンブルとリンク

サンプルプログラムsample.asmをアセンブル(コンパイルすること)して、オブジェクトファイルsample.objを作成する。

```
C:\H8>asm38 sample.asm ←
```

asm38(.exe)はアセンブルするためのexeファイルである。エラーが表示されたら、スペル等を確認して再度実行すること。

次にオブジェクトファイルsample.objをリンクして、アブソリュートファイルsample.absを作成する。

```
C:\H8>lnk sample.obj ←
```

lnk(.exe)はリンクするためのexeファイルである。エラーが表示されたら、スペル等を確認して再度実行すること。

以上で転送するためのファイルの準備は整ったので、次にH8マイコンのハードウェアの設定を行う。

(4) H8マイコンの設定

H8マイコンに書き込むために、SW1とSW2を**ブートモード**にする必要がある。以下の手順で行うこと。

- ・電源をOFFにする。
- ・SW1、SW2にジャンパーピンを差し込みブートモードにする。**方向を間違えないこと**(図3参照)。
- ・RS-232Cケーブルを接続する。
- ・以上を確認したら、**電源をON**にすること。

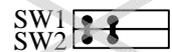
通常動作モード 

書き込んだプログラムを動作させる場合は開放(オープン)します。

ブートモード 

プログラムを書き込む場合はジャンパーピンを使って短絡(ショート)します。

禁止事項



ジャンパーピンを間違っ
図のように使うと電源と
GNDがショートして素子を
破損します。

図3 SW1、SW2の使い方

(5) プログラムの転送

プログラムをH8マイコンに転送するためにはhterm.exeを利用する。まずは次のようにしてhtermを起動する。

```
C:\H8>hterm ←
```

以下の手順通りに進めること。

- ・[ctrl]キーを押しながら[F]キーを押す。
- ・「Set Boot and Hit Any key.」と表示されるので、キーボードのどのキーでもよいので押すこと。
- ・「Input Control Program Name :」と表示されるので、

```
C:\3664\WRITER\3664.MOT ←
```

と入力する。

- ・「Input Program File Name :」と表示されるので、アブソリュートファイル名を入力する。今回の場合は以下のようになる。

```
sample.abs ←
```

- ・「Program Completed」と表示されれば、転送終了である。
- ・最後に[Esc]キーを押して、htermを終了する。

もし、エラーなど表示される場合はスペル等を確認して、再度実行すること。

(6) プログラムの実行

プログラムの転送ができれば、サンプルプログラムを動作させてみましょう。

その前にSW1、SW2を設定し、**通常動作モード**にする必要があります。以下の手順で行うこと。

- ・電源をOFFにする。
- ・SW1、SW2にジャンパーピンを外し、通常動作モードにする(図3参照)。
- ・以上を確認したら、**電源をON**にすること。

電源をONにした瞬間、H8マイコンが動作する。プログラムが転送されていればLEDの点滅間隔が変わっているはずである。変わらない場合はもう一度最初から実行してみる。

5. サンプルプログラムの変更

ここまでの説明でH8マイコンの一通りの使い方は説明した。ここからがアセンブラ言語のプログラム演習となる。まずはサンプルプログラムを書き換えてみよう。書き換えたプログラムはH8マイコンに転送して、動作確認すること。

(1) サンプルプログラムの内容

テキストエディタ(例えばEmEditorなど)を利用してサンプルファイルを開くこと。内容は図9のようにになっているはずである。行番号は見やすくするために表示しているが、プログラムには無関係である。図9で授業内容と示された範囲は既に授業で学習した命令を使っている。プログラム中の空白を利用して、自分が理解できるように各命令の説明を書き加えること。

```

01:      .CPU 300HN
02:      .SECTION VECTOR,DATA,LOCATE=H'0
03:      .DATA.L START
04:      .SECTION PROG,CODE,LOCATE=H'34
-----
05: START: MOV.L #H'0000FF80,SP
06:      MOV.B #H'01,R0L
07:      MOV.B R0L,@PCR8
08: LP:   MOV.B #H'01,R0L
09:      MOV.B R0L,@PDR8
10:      MOV.L #H'00400000,ER0
11: ON:   DEC.L #H'00000001,ER0
12:      BNE ON
13:      MOV.B #H'00,R0L
14:      MOV.B R0L,@PDR8
15:      MOV.L #H'00400000,ER0
16: OFF:  DEC.L #H'00000001,ER0
17:      BNE OFF
18:      BRA LP
-----
19: PCR8: .EQU H'FFEB
20: PDR8: .EQU H'FFDB
21:      .END
    
```

図9 サンプルプログラム

(2) サンプルプログラムの変更

10行目と15行目の数値を変更すると点滅時間が変わるはずである。まずは適当な値でもよいので数字を変えてみる。プログラムを変更したときはテキストエディタで上書き保存するか、ファイル名を変えて保存すること。後は前述の手順を踏んで実行すると、変更したとおりの動作をするはずである。プログラムの意味が分かってきたら、自分の思うように動作させてみるとよい。

(3) 課題プログラム

LEDを複数個、同時に点滅するプログラムを作成すること。プログラムに合わせて回路も変更すること。ただし、LEDは最大で4個までとする。ヒントを参考にして、図9のサンプルプログラムを変更すること。授業内容の範囲以外は変更しないこと。

ヒント

今回はH8マイコンのポート8を入出力ポートとして使用している(図1のCN2側の9~16番ピン。P80~PB7の8ビット出力)。入出力ポートの設定はDDR(データディレクションレジスタ)とDR(データレジスタ)で行う必要がある。それぞれのレジスタの説明は教科書、ノートを参考にすること。図9のプログラムにおいて19、20行目でDDRとDRを表2のように定義している。

表2 DDRとDRの定義

	メモリアドレスシンボル	メモリアドレス(16進)
DDR	PCR8	FFEB
DR	PDR8	FFDB



あとはプログラムの7、9、14行目のように、メモリアドレスシンボルを利用してDDRとDRを書き換えればよいことが分かる。

謝辞

H8マイコンのアセンブラ演習を行うために3E有志学生の奮闘があったことをここに記し、謝辞に代えさせていただきます。この演習は、はんだ付け作業から開発環境の構築まで、全ての準備は学生により行われました。構想段階から数えると約3ヶ月間、放課後を利用しての作業でしたが、みなさんのがんばりに感謝いたします。ありがとうございました。