

AVR入門 明るさ測定 レポート

- 実験日：2024年1月16日
- 実験者：2144 吉高 僚真

実験目的

- マイコンによるLEDマトリクス制御の制御ができる

ワーク1 LEDマトリクスの点

- マイコンボードのPB0～PB6をLEDマトリクスボードのP1～P7に、マイコンボードのPD0～PD6をLEDマトリクスのP9～P15に配線し、LEDが点灯するプログラムを作成せよ。なお、ここでは、8行目、8列目は使用せずに7×7のLEDマトリクスとして利用する。

プログラム

```
.include "tn2313def.inc"
.CSEG
START:
    LDI R17, 0b11111111
    OUT DDRD, R17
    LDI R17, 0b11111111
    OUT DDRB, R17
    OUT PORTB, R17
LIGHT:
    LDI R17, 0b00000000
    OUT PORTD, R17
```

説明

- PORTBをすべてHIGHに設定する
- PORTDをすべてLOWに設定する

ワーク2

- ワーク1のプログラムを変更し、1行おき（1列おき）にLEDが点灯するプログラムを作成せよ

プログラム(1行おき)

```
.include "tn2313def.inc"
.CSEG
START:
    LDI R17, 0b11111111
    OUT DDRD, R17
    LDI R17, 0b11111111
    OUT DDRB, R17
```

```

    OUT PORTB, R17
LIGHT:
    LDI R17, 0b01010101 ;変更
    OUT PORTD, R17

```

プログラム(1列おき)

```

.CSEG
START:
    LDI R17, 0b11111111
    OUT DDRD, R17
    LDI R17, 0b11111111
    OUT DDRB, R17
    LDI R17, 0b01010101 ;追加
    OUT PORTB, R17
LIGHT:
    LDI R17, 0b00000000
    OUT PORTD, R17

```

説明

- すべてのLEDを点灯させるプログラムから、上記のように変更する
- 行の場合はP9~P16につながっているのでPORTDを0,1を交互に繰り返すようにする
- 列の場合はP1~P8につながっているのでPORTBを0,1を交互に繰り返すようにする

ワーク3

- LEDマトリクスを用いて、KOSENと順に表示せよ。なお、点灯間隔は1秒程度する。ダイナミック制御が必要

プログラム

```

.include "tn2313def.inc"
.CSEG
START:
    LDI R17, 0b11111111
    OUT DDRD, R17
    LDI R17, 0b11111111
    OUT DDRB, R17
LOOP:
K:
    LDI R16, 100
_K:
    ;1行目
    LDI R17, 0b00100010
    OUT PORTB, R17
    LDI R17, 0b11111110
    OUT PORTD, R17

```

```
OUT T10MS
```

```
;2行目
```

```
LDI R17, 0b00010010
```

```
OUT PORTB, R17
```

```
LDI R17, 0b11111101
```

```
OUT PORTD, R17
```

```
OUT T10MS
```

```
;3行目
```

```
LDI R17, 0b00001010
```

```
OUT PORTB, R17
```

```
LDI R17, 0b11111011
```

```
OUT PORTD, R17
```

```
OUT T10MS
```

```
;4行目
```

```
LDI R17, 0b00000110
```

```
OUT PORTB, R17
```

```
LDI R17, 0b11110111
```

```
OUT PORTD, R17
```

```
OUT T10MS
```

```
;5行目
```

```
LDI R17, 0b00001010
```

```
OUT PORTB, R17
```

```
LDI R17, 0b11101111
```

```
OUT PORTD, R17
```

```
OUT T10MS
```

```
;6行目
```

```
LDI R17, 0b00010010
```

```
OUT PORTB, R17
```

```
LDI R17, 0b11011111
```

```
OUT PORTD, R17
```

```
OUT T10MS
```

```
;7行目
```

```
LDI R17, 0b00100010
```

```
OUT PORTB, R17
```

```
LDI R17, 0b10111111
```

```
OUT PORTD, R17
```

```
OUT T10MS
```

```
SUBI R16, 1
```

```
BRNE _K
```

```
0:
```

```
LDI R16, 100
```

```
_0:
```

```
;1行目
```

```
LDI R17, 0b0011000
```

```
OUT PORTB, R17
```

```
LDI R17, 0b11111110
OUT PORTD, R17
OUT T10MS
```

```
;2行目
```

```
LDI R17, 0b0100100
OUT PORTB, R17
LDI R17, 0b11111101
OUT PORTD, R17
OUT T10MS
```

```
;3行目
```

```
LDI R17, 0b1000010
OUT PORTB, R17
LDI R17, 0b11111011
OUT PORTD, R17
OUT T10MS
```

```
;4行目
```

```
LDI R17, 0b11110111
OUT PORTD, R17
OUT T10MS
```

```
;5行目
```

```
LDI R17, 0b11011111
OUT PORTD, R17
OUT T10MS
```

```
;6行目
```

```
LDI R17, 0b0100100
OUT PORTB, R17
LDI R17, 0b10111111
OUT PORTD, R17
OUT T10MS
```

```
;7行目
```

```
LDI R17, 0b0011000
OUT PORTB, R17
LDI R17, 0b01111111
OUT PORTD, R17
OUT T10MS
```

```
SUBI R16, 1
BRNE _0
```

```
S:
```

```
LDI R16, 100
```

```
_S:
```

```
;1行目
```

```
LDI R17, 0b00111100
OUT PORTB, R17
LDI R17, 0b11111110
OUT PORTD, R17
OUT T10MS
```

```
;2行目
LDI R17, 0b0100010
OUT PORTB, R17
LDI R17, 0b1111101
OUT PORTD, R17
OUT T10MS
```

```
;3行目
LDI R17, 0b0000100
OUT PORTB, R17
LDI R17, 0b1111011
OUT PORTD, R17
OUT T10MS
```

```
;4行目
LDI R17, 0b0001000
OUT PORTB, R17
LDI R17, 0b1110111
OUT PORTD, R17
OUT T10MS
```

```
;5行目
LDI R17, 0b0010000
OUT PORTB, R17
LDI R17, 0b1101111
OUT PORTD, R17
OUT T10MS
```

```
;6行目
LDI R17, 0b0100010
OUT PORTB, R17
LDI R17, 0b1011111
OUT PORTD, R17
OUT T10MS
```

```
;7行目
LDI R17, 0b0011100
OUT PORTB, R17
LDI R17, 0b0111111
OUT PORTD, R17
OUT T10MS
```

```
SUBI R16, 1
BRNE _S
```

E:

```
LDI R16, 100
```

_E:

```
;1行目
LDI R17, 0b0111110
OUT PORTB, R17
LDI R17, 0b1111110
OUT PORTD, R17
OUT T10MS
```

```
;2行目
LDI R17, 0b00000010
OUT portb, R17
LDI R17, 0b1111101
OUT PORTD, R17
OUT T10MS
```

```
;3行目
LDI R17, 0b11111011
OUT PORTD, R17
OUT T10MS
```

```
;4行目
LDI R17, 0b01111110
OUT portb, R17
LDI R17, 0b1110111
OUT PORTD, R17
OUT T10MS
```

```
;5行目
LDI R17, 0b00000010
OUT portb, R17
LDI R17, 0b1101111
OUT PORTD, R17
OUT T10MS
```

```
;6行目
LDI R17, 0b1011111
OUT PORTD, R17
OUT T10MS
```

```
;7行目
LDI R17, 0b01111110
OUT portb, R17
LDI R17, 0b0111111
OUT PORTD, R17
OUT T10MS
```

```
SUBI R16, 1
BRNE _E
```

N:

```
LDI R16, 100
```

_N:

```
;1行目
LDI R17, 0b01000010
OUT PORTB, R17
LDI R17, 0b11111110
OUT PORTD, R17
OUT T10MS
```

```
;2行目
LDI R17, 0b01001110
```

```
OUT PORTB, R17
LDI R17, 0b1111101
OUT PORTD, R17
OUT T10MS
```

```
;3行目
```

```
LDI R17, 0b0101010
OUT PORTB, R17
LDI R17, 0b1111011
OUT PORTD, R17
OUT T10MS
```

```
;4行目
```

```
LDI R17, 0b0110010
OUT PORTB, R17
LDI R17, 0b1110111
OUT PORTD, R17
OUT T10MS
```

```
;5行目
```

```
LDI R17, 0b0100010
OUT PORTB, R17
LDI R17, 0b1101111
OUT PORTD, R17
OUT T10MS
```

```
SUBI R16, 1
BRNE _N
```

```
RJMP LOOP
```

```
T10MS:
```

```
LDI R22, 10
```

```
_T100US:
```

```
OUT T100US
SUBI R22, 1
BRNE _T100US
RET
```

```
T100US:
```

```
LDI R21, 249
```

```
_TUS:
```

```
NOP
NOP
NOP
NOP
NOP
SUBI R21, 1
BRNE _TUS
NOP
RET
```

説明

- ここではKを表示させる例を用いて説明する。

```
K:
    LDI R16, 100
_K:
    ;1行目
    LDI R17, 0b00100010
    OUT PORTB, R17
    LDI R17, 0b11111110
    OUT PORTD, R17
    OUT T10MS

    ///
    ///省略
    ///

    SUBI R16, 1
    BRNE _K
```

1文字を表示させる手順

- R16に100を入れて初期化する
- 一行ごとに表示させていき、ダイナミック駆動を行う
 - 1行目用にPORTB, PORTDを設定する
 - PORTBはいい感じに設定する
 - PORTDは1行目を表示するときは1ビット目を0に設定する
 - 10ミリ秒待つ
 - 2行目用のPORTB, PORTDを設定する
 - 10ミリ秒待つ
 - これを7行目まで繰り返す
- R16から1を引く
- ゼロフラグが立っていなかったらBRNE _Kで_Kに戻る
- ゼロフラグが立っていたら次の文字に進む