

調光器

03

03-1

動作説明

可変抵抗器を使い LED の明るさを調整する

内蔵の**アナログ・コンパレータ**と切り替え可能な**基準電圧**を利用し、4ビットのAD変換を行う。

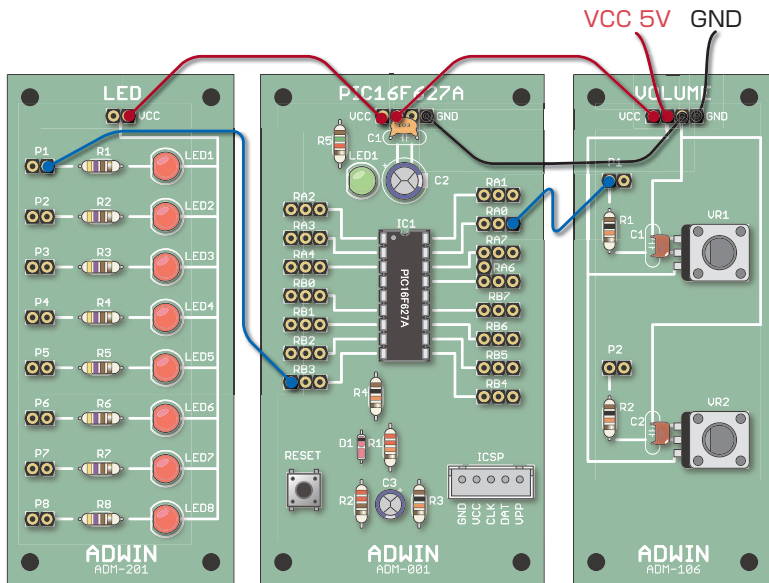
AD変換は**タイマ0**を使い、約10m秒おきに行う。

CCP1モジュールの**PWMモード**を使いLEDの明るさを調整する。

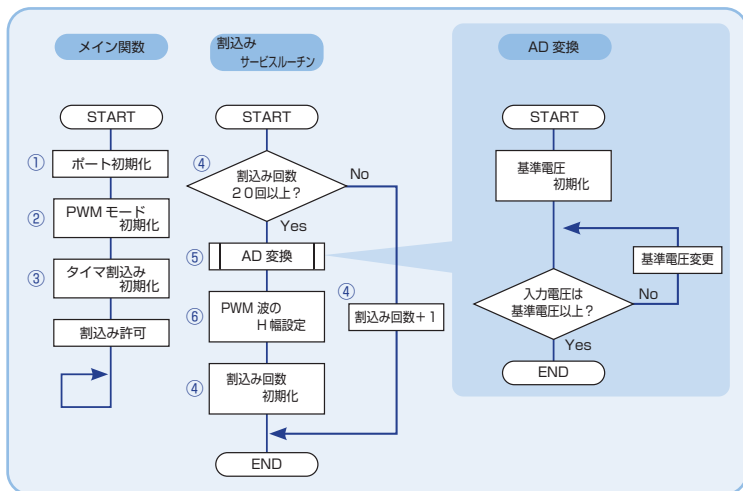
03-2

配線

使用ボードNo. ADM-001・ADM-106・ADM-201



03-3 概略フローチャート



03-4 概略フローチャート解説

① 17 番ピンを ANO ピンとして利用する

② CCP1 モジュールの PWM モードを使い、CCP1 ピン (9 番ピン) から PWM 波を出力
 <設定内容>

CCP モジュール

- ・モード：PWM モード
- ・周期：1000 μ 秒
- ・H 幅：AD 変換結果により可変

タイマ2

- ・クロック源：内部クロック
- ・プリスケアラ：4

- ③ タイマ 0 のタイマ割り込みにより、定期的に AD 変換を行う

<設定内容>

- ・プリスケアラ : 2
- ・クロック源 : 内部クロック

※ 512 μ 秒おきに割り込み発生

- ④ 約 10m 秒おきに AD 変換を行う

$512 \mu\text{秒} \times 20 \text{回} \approx 10\text{m 秒}$

- ⑤ AD 変換

ANO ピンから入力された電圧と基準電圧を比較し AD 変換を行う。

詳細は後述。

- ⑥ ⑤の AD 変換結果を元に CCP1L レジスタの値を設定

03-5 PIC16F627A における AD 変換

PIC16F627A では、内蔵の **アナログ・コンパレータ** と切り替え可能な **基準電圧** を利用することで、4 ビットの AD 変換を行うことができる。

基準電圧は VRCON レジスタの下位 4 ビットで設定が可能であり、ANO ピンから入力された電圧が、基準電圧を超えると、**CMCON レジスタの C1OUT ビット** が 1 となる。

■ 基準電圧 ※ V_{DD} : 4.8V の場合

例) VRCON 下位 4 ビットに 1101 を設定した場合

ANO ピンの入力電圧 : 2.4V より低い
C1OUT ビット : 0

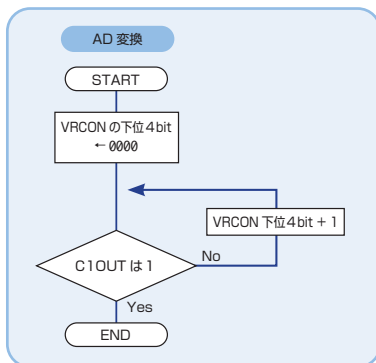
ANO ピンの入力電圧 : 2.4V 以上
C1OUT ビット : 1

基準電圧	VRCON 下位 4 ビット			
3.0V ~	—			
2.8 ~ 3.0V	1	1	1	1
2.6 ~ 2.8V	1	1	1	0
2.4 ~ 2.6V	1	1	0	1
2.2 ~ 2.4V	1	1	0	0
2.0 ~ 2.2V	1	0	1	1
1.8 ~ 2.0V	1	0	1	0
1.6 ~ 1.8V	1	0	0	1
1.4 ~ 1.6V	1	0	0	0
1.2 ~ 1.4V	0	1	1	1
1.0 ~ 1.2V	0	1	1	0
0.8 ~ 1.0V	0	1	0	1
0.6 ~ 0.8V	0	1	0	0
0.4 ~ 0.6V	0	0	1	1
0.2 ~ 0.4V	0	0	1	0
0.0 ~ 0.2V	0	0	0	1
~ 0.0V	0	0	0	0

具体的には、C1OUT ビットが 1 になるまで VRCON の下位 4 ビットを 0 ~ 15 まで 1 ずつプラスする。

C1OUT ビットが 1 になった時点の基準電圧が ANO ピンの入力電圧である。

例えば、VRCON の値が、12 (2 進数 : 1100) の時に C1OUT ビットが 1 となれば、ANO ピンからの入力電圧は 2.2 ~ 2.4V である。



03-6 レジスタ

CMCON レジスタ

C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0
					0	1	0

1,2,17,18 番端子をどの機能で使用するか設定
 010 : AN0 ~ AN3 をアナログ・コンパレーへの
 の入力端子として使用

1 : AN0 ピンからの入力電圧 > 基準電圧
 0 : AN0 ピンからの入力電圧 < 基準電圧

VRCON

V _{REN}	V _{ROE}	V _{RR}	V _{R3}	V _{R2}	V _{R1}	V _{R0}

基準電圧の設定 (前述表参照)

$$1 : V_{ref} = \left(\frac{V_R<3:0>}{24} \right) \times V_{DD}$$

$$0 : V_{ref} = \left(V_{DD} \times \frac{1}{4} \right) + \left(\frac{V_R<3:0>}{32} \right) \times V_{DD}$$

1 : 出力電圧 RA2 に接続
 0 : 出力電圧 RA2 に接続しない

1 : V_{REF} 回路電源 ON
 0 : V_{REF} 回路電源 OFF

03-7 ソースコード

掲載ソースは弊社 HP で公開中のエレモサンプルソースからの抜粋です。
(URL : https://www.adwin.com/image/support/ADM_SampleSource.zip)

AD 変換部

```
// AD 変換の結果により CCPR1L に代入する値を配列
// 要素変更により LED 明るさ調整可
const unsigned int a_ccpr1l[16] = {
    240, 200, 180, 160, 140, 120, 100, 80,
    70, 60, 50, 40, 30, 20, 10, 5
};

/*
AD 変換
戻り値 : VRCON の下位 4 ビット
*/
int getAnalogV( void ){

    VRCON = 0b10100000; // VRCON の下位 4 ビットを初期化 : 基準電圧 0V

    // C1OUT が 1 になるまでループ
    // -> AN0 の電圧が基準電圧以上になるまでループ
    while ( C1OUT != 1 ){
        VRCON++; // 基準電圧変更
    }

    return( VRCON&0x0F );
}
```

■ 割込みサービスルーチン

```
/*
  割込みサービスルーチン
  512us ごとにタイマ割込みお渡し整理
*/
static void interrupt isr( ){
    static int cnt = 0;      // 割込み回数カウント
    int ad;                  // AD 変換結果格納

    // 割込みが 20 回ごとに AD 変換
    // 512us × 20 ≒ 10ms
    if ( cnt > 20 ){

        ad = getAnalogV( );  // AD 変換

        CCP1L = a_ccpr1l[ad]; // PWM H 幅を AD 変換結果によって設定
        CCP1X = 0;
        CCP1Y = 0;

        cnt = 0;             // 割込み回数 初期化
    }
    else{
        cnt++;               // 割込み回数 + 1
    }

    T0IF = 0;              // タイマ 0 割込みフラグをクリア
}
}
```