

製作手順

下記の手順を参考に作品の製作を行って下さい。

1. LEDを取付けるボードの加工
2. LEDの取付け
3. 基板配線(はんだ付け)
 - 3-1. 基板にPICマイコン用のICソケットを取付ける
 - 3-2. 1k Ω 抵抗や0.1 μ Fコンデンサを取付ける
 - 3-3. LED取り付けボードの配線
 - 3-4. PICマイコンの基板とLEDボードの配線
4. PICマイコンのプログラミング
5. テスト

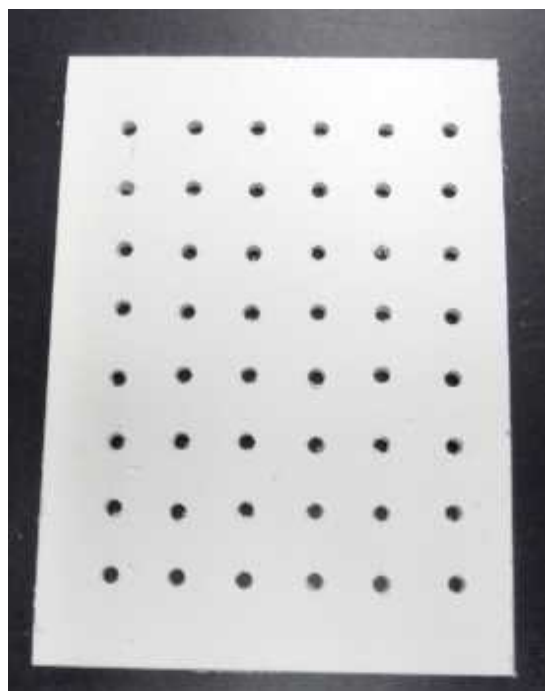
製作手順

1. LEDを取付けるボードの加工

LEDの取付けにボードを使用する場合は、その目的の大きさに合わせてノコギリで切ったり、LEDの取付け穴をドリルであけたりします。薄くて柔らかい資材の場合は、用意されてるドライバー・タイプのドリル(穴径3mm)を使用してください。



ドライバータイプのドリルでLEDの取り付け穴をあける

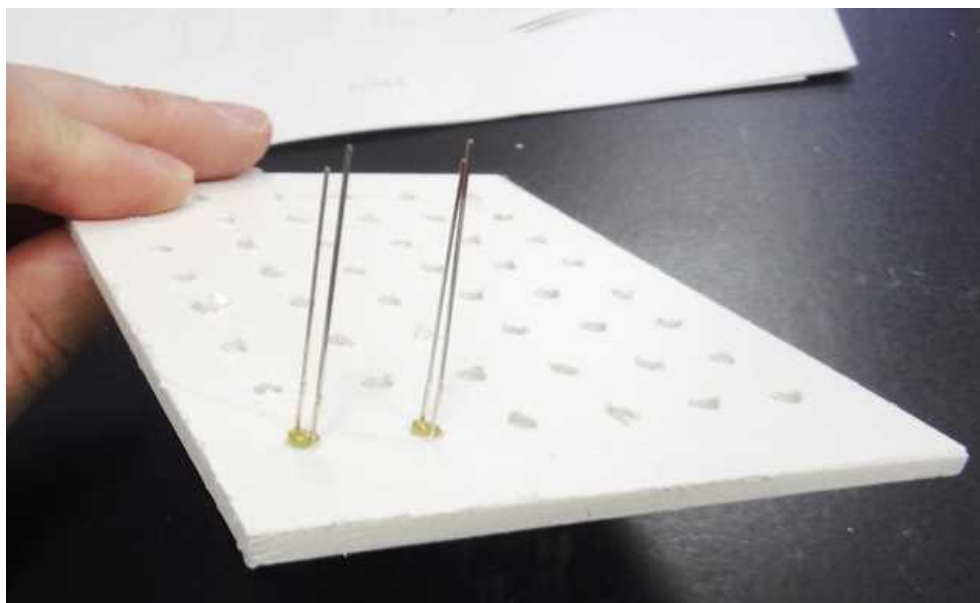


アクリル・ボードに穴を開けた後

製作手順

2. LEDの取付け

ある程度の硬さがある素材で径3mmの穴をあけると、接着材なしでLEDが取り付け可能です。またLEDは、足が長いアノード(+側に接続)と足が短いカソード(-側に接続)を区別しておきます。




LED2個を取り付け穴に挿した状態

製作手順

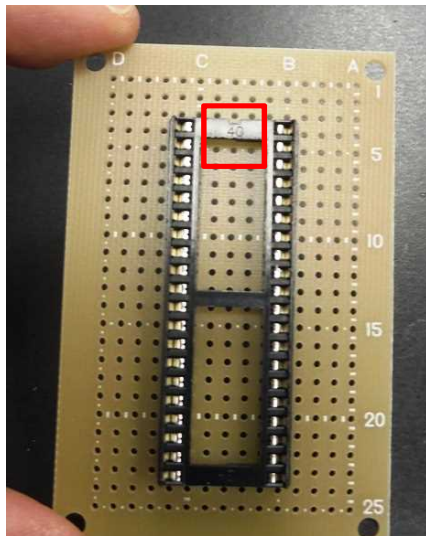
3. 基板配線(はんだ付け)

3-1. 基板にPICマイコン用のICソケットを取付ける

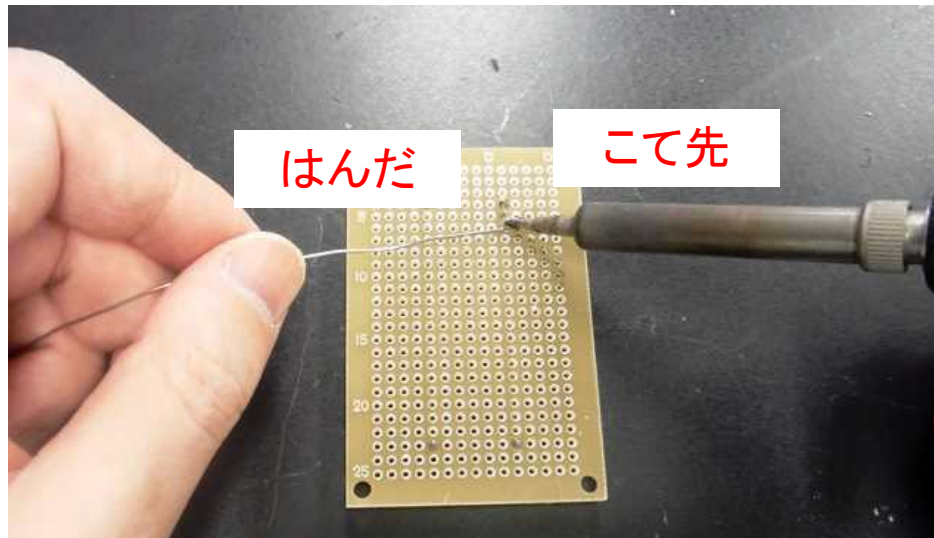
ソケットには上下があります。切欠きを上にして基板の上に載せ、基板と共に裏返して半田付けします。

はんだ付けを行う手順は、

- ①「こて先」で装着部(基板から出たソケットの足)を温める
- ②「はんだ」を近づけて融かす
- ③はんだが融けたら装着部から「はんだ」と「こて先」を離す



基板にソケットを載せる



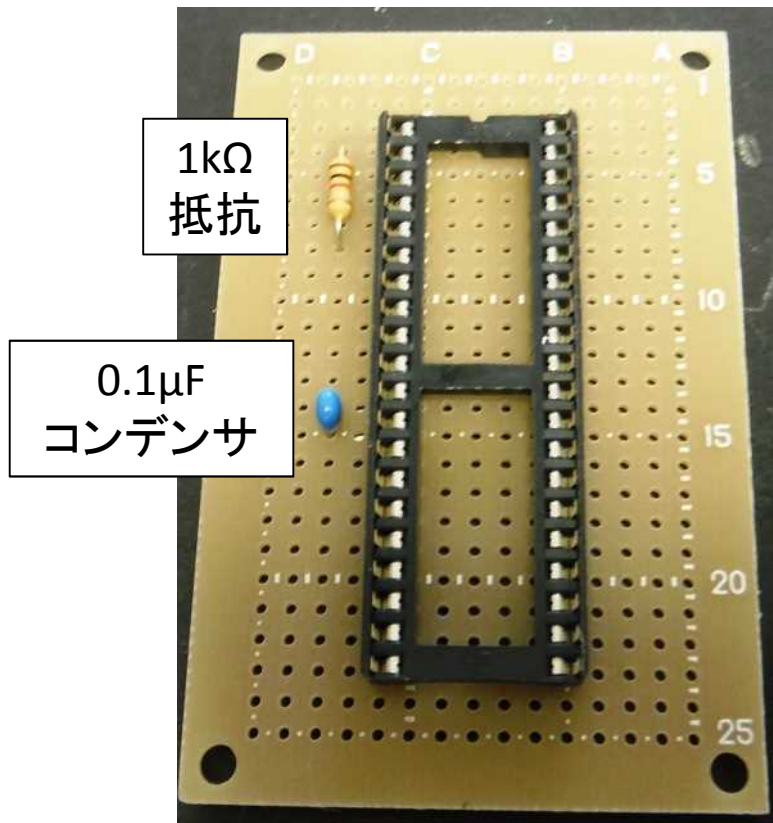
はんだ付けの作業

製作手順

3. 基板配線(はんだ付け)

3-2. 1k Ω 抵抗や0.1 μ Fコンデンサを取付ける

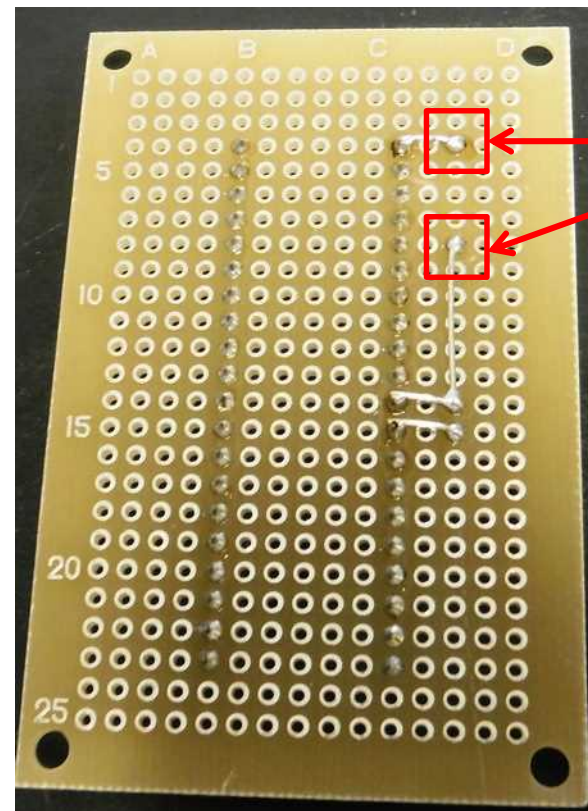
1k Ω 抵抗は、1番(Vpp/MCLR/RE3)と11番(VDD)ピンの間に接続します。0.1 μ Fのコンデンサは、11番と12番(Vss)の間に接続します。



1k Ω
抵抗

0.1 μ F
コンデンサ

基板表側



抵抗の足を曲げて、余分な足の長さをニッパで切り、はんだ付けします。

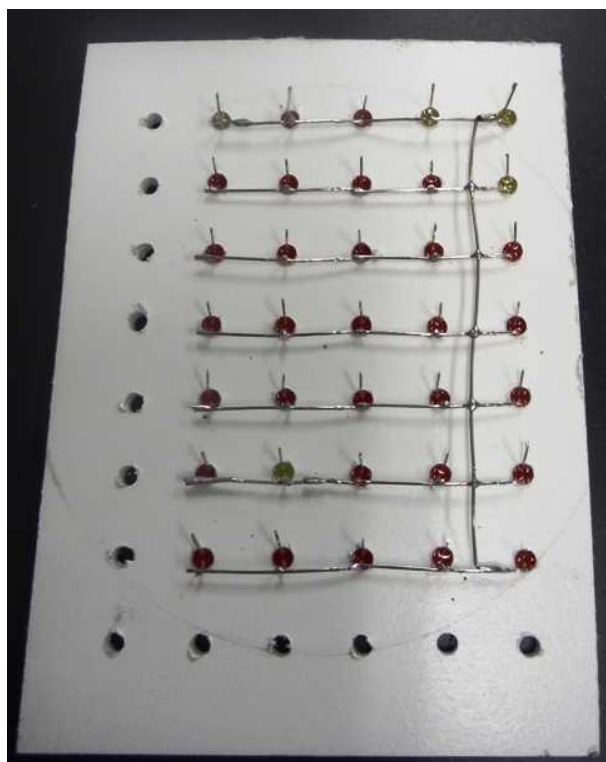
基板うら側

製作手順

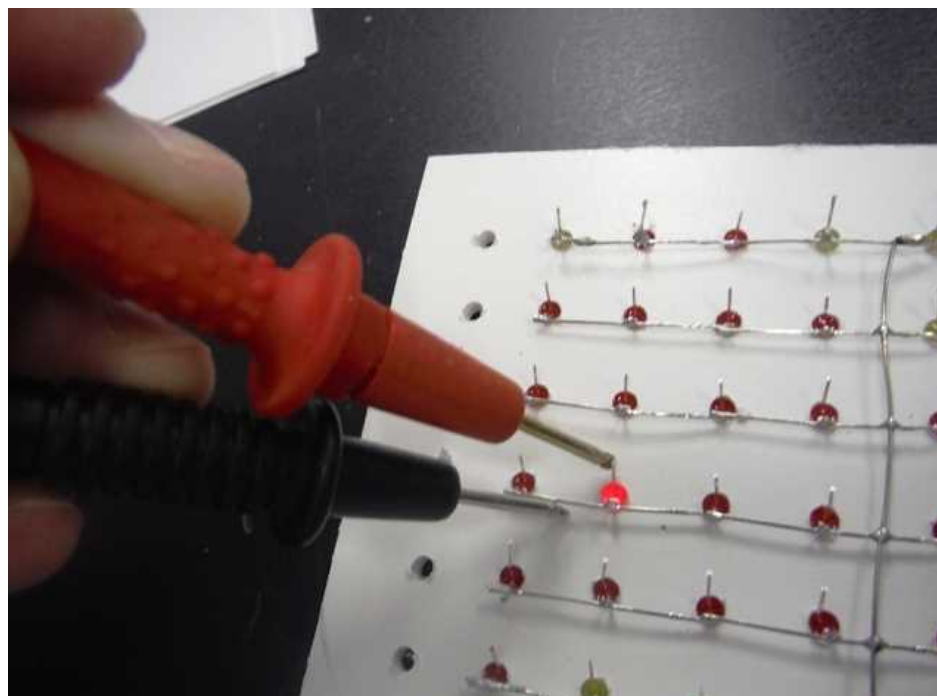
3. 基板配線(はんだ付け)

3-3. LED取り付けボードの配線

LED取り付けのボードでカソード(一側に接続)がまとめられると、PICマイコン基板からの配線が少なくてすみます。例えば、下記がLEDのカソードをボード上で結線した例です。



LEDのカソードを結線



結線後は、テスタを使用してLEDの点灯を
チェックします

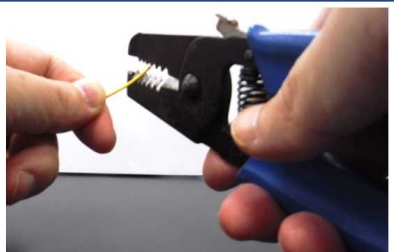
製作手順

3. 基板配線(はんだ付け)

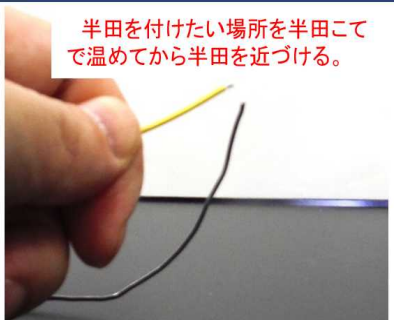
3-4. PICマイコンの基板とLEDボードの配線

LEDのアノード側に導線を取り付け、基板うら側に出ているソケットの端子と結線する。導線を取り付ける際は、ワイヤーストリッパーを使用して先端3mm程度被覆をはぎ、その先端を半田でメッキしておくともスムーズに付く。

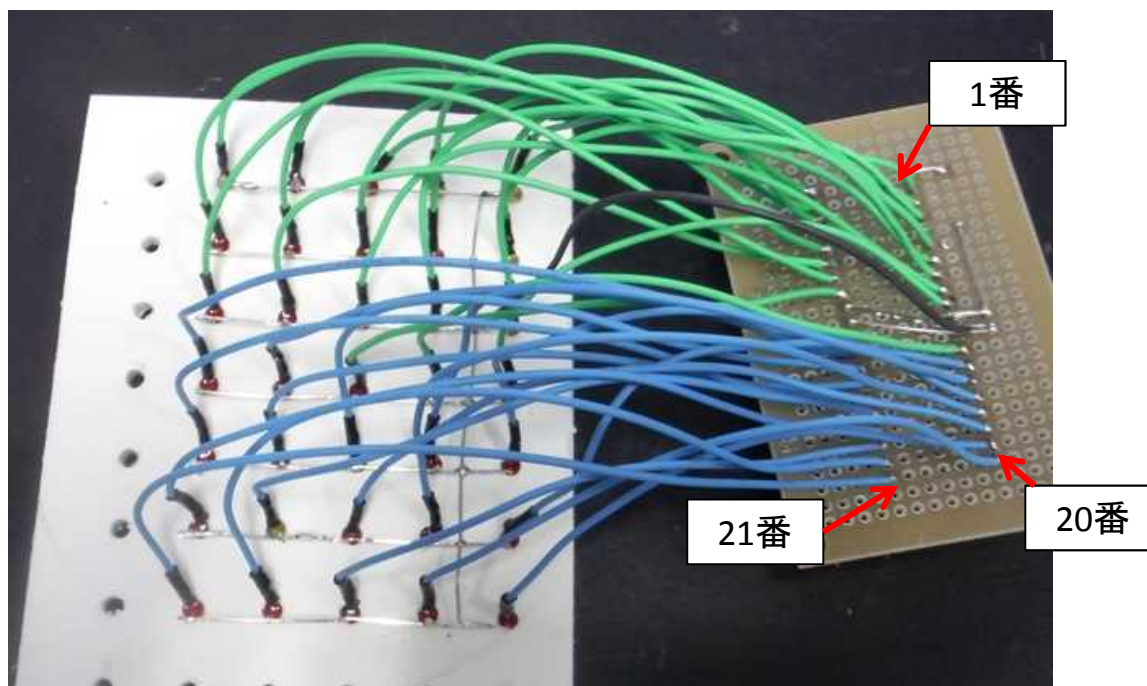
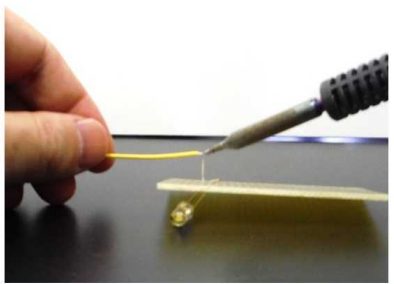
- ①ワイヤーストリッパー
を使って、導線の
先端3mm長さで
被覆をはぐ。



- ②導線の先端を
半田こてで温め、
半田を近づけて
コーティングする。
また、LEDのアノ
ード端子も同様に
コーティングして
おく。



- ③半田でコーティ
ングした導線の先端
とLEDのアノード端
子を近づけ、半田
こてで温める。



PICマイコンの基板が裏返しになっている時、
ピン番号の向きに注意すること！！

4. PICマイコンのプログラミング

4-1. パソコンの起動(ログインパスワードは実験室のホワイトボードに記載)

4-2. サンプルプログラムのダウンロード

Webブラウザを起動して <http://esato.net/ex/micom/> にアクセスし、「資料・サンプル」の「c.PICマイコンプログラムサンプル2;ライト注意灯」上でマウスの右ボタンを押す。「リンク先を名前を付けて保存」等とすると、ダウンロードしたプログラムデータがPCに保存される。

4-3. MPLABの起動

Webブラウザを起動して <http://esato.net/ex/micom/> にアクセスし、「プログラミングMPLABの使い方」を参照してください。
先にダウンロードしたサンプルプログラムは、「3. ソースコード(プログラム)の作成」の「②テキストエディタに入力」の際に貼り付けを行い、次のページを参考に書き換えや削除を行って下さい。

4-5. PICライターによるプログラムの書き込み

Webブラウザを起動して <http://esato.net/ex/micom/> にアクセスし、「PICライターの使い方」を参照してください。

「c.PICマイコンプログラムサンプル2;ライト注意灯」の補足説明

- このサンプルプログラムの動作フロー図は、第1回の資料(ワークシート・サンプル p.5)にあります。
- ラベル「LOOP」の手前まで(次ページの①そのまま利用する)は、初期設定を行っています。I/Oポートは全て出力の設定がなされており、そのまま利用可能です。
- ラベル「LOOP」から命令「GOTO LOOP」まで(②変更する)は、目的に合わせ変更しなければなりません。
- サンプルでは、「ホ」の表示パターンを作るために、以下の10行を記述しています。ワークシートの「各ポートの出力パターン」の表を参照してPORTAからPORTEに出力するデータに書き換えてください。(●点灯が"1" ○消灯が"0"とする)

```
MOVLW B'00100100' ←PORTAに出力する8ビットのデータ
MOVWF PORTA
MOVLW B'00100111' ←PORTBに出力する8ビットのデータ
MOVWF PORTB
MOVLW B'00111010' ←PORTCに出力する8ビットのデータ
MOVWF PORTC
MOVLW B'00100101' ←PORTDに出力する8ビットのデータ
MOVWF PORTD
MOVLW B'00000011' ←PORTEに出力する8ビットのデータ
MOVWF PORTE
```

```

;
; SAMPLE 02 - ほたる鑑賞で役立つ「ライト注意灯」
;
;--- デバイスの選択, コンフィグレーションの設定 ---
LIST P=PIC16F1939
INCLUDE "P16F1939.INC"

                CONFIG CONFIG1,
_FOSC_INTOSC& _WDTE_OFF& _PWRTE_OFF& _MCLRE_ON& _CP_OFF&
_CPD_OFF& _BOREN_OFF& _CLKOUTEN_OFF& _IESO_OFF& _FCMEN_OFF
                CONFIG CONFIG2, _WRT_OFF& _VCPEN_OFF&
_PLLEN_ON& _STVREN_OFF& _BORV_19& _LVP_OFF

;--- ユーザエリア -----
#define UA      30H
#define UB      31H
#define UC      32H
;--- リセット, 割り込みベクタ ---
ORG 0
GOTO MAIN
ORG 4
GOTO MAIN
;===== メインルーチン =====
MAIN
BCF  INTCON,7    ;割り込み不許可
BSF  BSR,BSRO   ;バンク1に切り替え
MOVLW B'00000000'
MOVWF TRISA     ;ポートAのデータ方向(入出力)設定
MOVLW B'00000000'
MOVWF TRISB     ;ポートBのデータ方向(入出力)設定
MOVLW B'00000000'
MOVWF TRISC     ;ポートCのデータ方向(入出力)設定
MOVLW B'00000000'
MOVWF TRISD     ;ポートDのデータ方向(入出力)設定
MOVLW B'00000000'
MOVWF TRISE     ;ポートEのデータ方向(入出力)設定
MOVLW B'01101010'
MOVWF OSCCON    ;内部発生クロック4MHz選択
;-----
MOVLW 080H      ;オプションレジスタの設定
MOVWF OPTION_REG
BCF  BSR,BSRO   ;バンク0に切り替え
NOP

```

①そのまま利用する

```

LOOP

MOVLW B'00100100' ;Time 0
MOVWF PORTA
MOVLW B'00100111'
MOVWF PORTB
MOVLW B'00111010'
MOVWF PORTC
MOVLW B'00100101'
MOVWF PORTD
MOVLW B'00000011'
MOVWF PORTE
CALL  WAIT0

;Time 1
MOVLW B'00010100'
MOVWF PORTA
MOVLW B'10101011'
MOVWF PORTB
MOVLW B'00001000'
MOVWF PORTC
MOVLW B'11010010'
MOVWF PORTD
MOVLW B'00000101'
MOVWF PORTE
CALL  WAIT0

```

Time0「ホ」の表示パターン

②変更する

```

;
; (省略)
;
                ;Time 13
MOVLW B'00000001'
MOVWF PORTA
CALL  WAIT0

                ;Time 14
MOVLW B'00000000'
MOVWF PORTA
CALL  WAIT0

                ;Time 15
CALL  WAIT0
CALL  WAIT0

GOTO LOOP

```

②変更する

```

;-----
; 1SEC ;1秒カウント(待機)するためのサブルーチン
;-----
WAIT0
MOVLW D'10' ;
MOVWF UC    ;UC=10とする
WAIT0A
CALL  WAIT1 ;WAIT1をサブルーチンコール
DECFSZ UC,1 ;UCの値を-1し、ゼロならばGOTO命令をスキップ
GOTO  WAIT0A ;ラベルWAIT0Aへジャンプ
RETURN

;-----
; 100mSEC ;100ミリ秒カウントするサブルーチン
;-----
WAIT1
MOVLW D'150'
MOVWF UA    ;UA=150
WAIT1A
MOVLW D'223'
MOVWF UB    ;UB=223
WAIT1B
DECFSZ UB,1 ;UBを-1し、ゼロならばGOTO命令をスキップ
GOTO  WAIT1B ;ラベルWAIT1Bへジャンプ
DECFSZ UA,1 ;UAを-1し、ゼロならばGOTO命令をスキップ
GOTO  WAIT1A ;ラベルWAIT1Aへジャンプ
RETURN

END ;ソースファイルの終わりを宣言

```

③必要に応じて変更する

※「1秒待ち」を同様に使用する場合は、変更の必要がありません。「0.5秒待ち」を使用する場合は、ラベル「WAIT0」の次の行を「MOVLW D'5'」に変更します。