

```

/*
 * File:   kadai07.c
 * Author: mono2kuri
 * 反射板を移動させ、dsSSの出力電圧を（7SEG）に表示する。
 * 注：XC8では、デジ`ル値をアナ`ル値へ変換する式で整数同士の乗除算は、結果が整数になる
 *     ようで、そのまま左辺に浮動小数点の変数を配置しても、整数の計算結果しか代入
 *     されないから、小数点以下が必ず0となり、その後の表示処理では、小数点以下の値を
 *     取得しても0としかならない。
 *     回避策は、左辺へ向け浮動小数点の結果として渡したい場合、整数同士の乗除算におい
 *     て、何か変数を浮動小数点の型にキャストする「(double)」ことで可能となる。
 * Created on 2019/06/12, 12:52
 */

#include <xc.h>

// クロック周波数指定
// ( __delay_ms(), __delay_us())関数が必要としているため
#define    _XTAL_FREQ    1000000

//記号文字定数：入出力
#define TGSW    (RA3)    // tgSW：トグルスイッチ
#define TCSW_Y    (RA5)    // tsSW：タクトスイッチ(黄色)
#define TCSW_B    (RE1)    // tsSW：タクトスイッチ(青色)

// 外部参照変数の宣言
extern char Seg7Data[18];

// 外部参照関数の宣言
extern unsigned int adConv();

//関数の宣言
void kadai07(void);

/* 関数名：kadai07()
 * 引 数：なし
 * 戻り値：なし
 * 概 要：
 *   入力：RA1/AN1＝測距センサの入力
 *   出力：7セグLEDの点灯＝AN1のAD変換結果を電圧に換算し、表示する。
 */
void kadai07(void)
{
    // 0-加変数の定義
    unsigned int iLower, iUpper;    // AD変換後の電圧値の1位と小数点第1位の値を格納する
    double fValue;    // AD変換後の電圧値

    // pic16f874aの端子定義/初期化
    PORTA = 0x00;    // PORTA = All 0 = Low
    PORTE = 0x00;    // PORTE = All 0 = Low
    PORTB = 0x00;    // PORTB = All 0 = Low
    PORTC = 0x00;    // PORTC = All 0 = Low
    PORTD = 0x00;    // PORTD = All 0 = Low
    TRISA = 0xFF;    // PORTA = 全て入力(AnalogもDigitalも)
    TRISE = 0x07;    // PORTE = 全て入力(RE0?RE3のみの設定、AnalogもDigitalも)
    TRISB=0x00;    // PORTB を出力に設定
    TRISC=0x00;    // PORTC を出力に設定
    TRISD=0x00;    // PORTD を出力に設定
    ADCON1 = 0b00000110;    // PORTA/E = Digital I/O, not A/D inputs

    //処理本体

```

```
while(i) {
    unsigned int value = adConv();
    if( 0 <= value && 64 > value ) PORTB = 0x01;
    else if( 64 <= value && 128 > value ) PORTB = 0x02;
    else if( 128 <= value && 192 > value ) PORTB = 0x04;
    else if( 192 <= value && 256 > value ) PORTB = 0x08;
    else if( 256 <= value && 320 > value ) PORTB = 0x10;
    else if( 320 <= value && 384 > value ) PORTB = 0x20;
    else if( 384 <= value && 448 > value ) PORTB = 0x40;
    else PORTB = 0x80;
    fValue = (double)value * 5 / 1023;
    iUpper = (unsigned int)(fValue);
    iLower = (unsigned int)(fValue * 10 + 0.5);
    iLower = iLower % 10;
    PORTD = Seg7Data[iUpper] + 0x80;
    RC6 = 1;
    RC7 = 0;
    __delay_ms(10);
    PORTD = Seg7Data[iLower];
    RC6 = 0;
    RC7 = 1;
    __delay_ms(10);
}
return;
}
```

```
// AD変換結果の取得
// RB0のみHight=LED点灯
// RB1のみHight=LED点灯
// RB2のみHight=LED点灯
// RB3のみHight=LED点灯
// RB4のみHight=LED点灯
// RB5のみHight=LED点灯
// RB6のみHight=LED点灯
// RB7のみHight=LED点灯
```