

```

/*-----*/
/* ものづくりコンテスト2019 山形県大会 */
/*-----*/
/* 事前公開課題2 */
/* 初期状態 */
/* ・トグルスイッチTSWはOFF、2つのタクトスイッチSW1、SW2はOFFとする。 */
/* ・フルカラーLEDは消灯している。 */
/* ・圧電ブザーからは何も音がでていない(OFF)。 */
/* ・7SEG2は「1.」を7SEG1は「5」を表示している。 */
/* ・DCモータおよびステッピングモータは停止している。 */
/*-----*/
/* 動作説明 */
/* ① 測距モジュールの測定距離を2つの7セグメントLEDに表示させること。ただし表示する数値は、以下 */
/* の表の通りとする。 7SEG2 7SEG1 */
/* 0cm ≤ 距離 ≤ 10cm 1. 0 */
/* 10cm < 距離 ≤ 15cm 1. 5 */
/* 15cm < 距離 ≤ 20cm 2. 0 */
/* 20cm < 距離 ≤ 25cm 2. 5 */
/* 25cm < 距離 ≤ 30cm 3. 0 */
/* 30cm < 距離 ≤ 35cm 3. 5 */
/* 35cm < 距離 ≤ 40cm 4. 0 */
/* 40cm < 距離 ≤ 45cm 4. 5 */
/* 45cm < 距離 ≤ 50cm 5. 0 */
/* 50cm < 距離 8. 8 */
/* ② トグルスイッチTSWがONのとき、7SEG2の「.」（ドット、点）を消灯させること。 */
/* 再び、トグルスイッチTSWがOFFになったら、7SEG2の「.」（ドット、点）を点灯させること。 */
/* 「.」の消灯/点灯は、どのタイミングでも行えるようにすること。 */
/*-----*/

/*-----*/
/* ※この課題ではトグルスイッチがONのときに「.」点灯、OFFのときは「.」消灯と理解できる。 */
/* 2個の7セグメントLEDはダイナミックドライブ方式での点灯を行うので、割り込みをつかう。 */
/* PSDセンサはフリーランニング方式でAD変換する。 */
/*-----*/

/*-----*/
/* インクルード */
/*-----*/
#include "sfr_r835a.h" /* R8C/35A SFRの定義ファイル */
#include "contobj_lib.h" /* 山形工高の定義ファイル */

/*-----*/
/* ポート仕様 */
/*-----*/
/* p0_0 PSDセンサ 入力端子(アナログ) */
/* p0_1 タクトスイッチ1 入力端子(デジタル) */
/* p0_2 タクトスイッチ2 入力端子(デジタル) */
/* p0_3 トグルスイッチ 入力端子(デジタル) */
/* p0_4~p0_7 未使用 */
/*-----*/
/* p2_0 右7セグメントLED(1の位) コモンアノード端子 */
/* p2_1 左7セグメントLED(10の位) コモンアノード端子 */
/* p2_2 フルカラーLED赤 */
/* p2_3 フルカラーLED青 */
/* p2_4 フルカラーLED緑 */
/* p2_5 ブザー端子 */
/* p2_6 未使用 */
/* p2_7 74HC574CLK */
/*-----*/
/* p6_0 ステッピングモータφ1 / 7セグメントLED Dp */
/* p6_1 ステッピングモータφ2 / " g */
/* p6_2 ステッピングモータφ3 / " f */
/* p6_3 ステッピングモータφ4 / " e */
/* p6_4 / " d */
/* p6_5 / " c */
/* p6_6 DCモータIC FIN / " b */
/* p6_7 DCモータIC RIN / " a */
/*-----*/

/*-----*/
/* シンボル定義(この部分はcontobj_lib.hへ移行予定) */
/*-----*/
#define PSD p0_0 /* PSDセンサ 入力端子(アナログ) */
#define SW1 p0_1 /* タクトスイッチ1 入力端子(デジタル) */
#define SW2 p0_2 /* タクトスイッチ2 入力端子(デジタル) */
#define TSW p0_3 /* トグルスイッチ 入力端子(デジタル) */

```

```

#define L7SEG2_COM p2_0 /* 右7セグメントLED(1の位) コモンアノード端子 */
#define R7SEG1_COM p2_1 /* 左7セグメントLED(10の位) コモンアノード端子 */

#define BUZZER p2_5 /* ブザー端子 */
/* (p2_6 未接続) */
#define CLK_574 p2_7 /* 74HC574CLK */

#define FIN p6_0 /* DCモータ IN1端子 */
#define RIN p6_1 /* DCモータ IN2端子

#define CN2 p2
#define CN3 p6

/*-----*/
/* OFF 1入力 ON 0入力 */
/*-----*/
#define ON 0
#define OFF 1

/*=====*/
/* プロトタイプ宣言 */
/*=====*/
void init( void ); /* ポートイニシャライズ, 割り込み処理設定等 */ /* 全課題使用 */
void timer( unsigned long timer_set ); /* ソフトウェアタイマー */ /* " */
void full_color_led( unsigned char led ); /* フルカラーLED点灯(自作) */ /* 課題1,2 */
int get_ad7( void ); /* A/D変換 PSD値取得 */ /* 課題2 */

/*=====*/
/* グローバル変数の宣言 */
/*=====*/
unsigned long cnt_rb; /* タイマRB用 */
unsigned long cnt10; /* ダイナミックドライブ点灯用タイマー */

unsigned int l7seg2_value = 1; /* 左7セグメントLEDに表示する値 */
unsigned int r7seg1_value = 5; /* 右7セグメントLEDに表示する値 */
unsigned int lr7seg_select = 0; /* 左右どちらの7セグメントLEDに表示するか 1:左 0:右 */

/* 7セグメントLED表示用データ */
/* 表示なしは[11]を出力するかコモン共通端子0出力 */
/* 0 1 2 3 4 5 6 7 8 9 */
unsigned char disp[12]={ 0xfc, 0x60, 0xda, 0xf2, 0x66, 0xb6, 0xbe, 0xe0, 0xfe, 0xe6, 0x01, 0x00 };

/*-----*/
/* タイマ本体 */
/* 引数 タイマ値 1=1ms */
/*-----*/
void timer( unsigned long timer_set )
{
    int i;
    do {
        for( i=0; i<1240; i++ );
    } while( timer_set-- );
}

/*-----*/
/* タイマRB 割り込み処理 */
/* 1ms毎に実行される */
/*-----*/
#pragma interrupt intTRB(vect=24)
void intTRB( void )
{
    unsigned char data;

    cnt_rb++;
    cnt10++;

    /*-----*/
    /* 7セグメントLED表示(ダイナミックドライブ方式) */
    /*-----*/
    if( cnt10>10 ) {
        /* 7セグメントLED表示 */
        if( lr7seg_select==1 ) {

            /* 左7セグメントLED表示 */
            //data = CN2 & 0xfe; /* 1111 1100 */
            //CN2 = data | 0x02;

```

```

        L7SEG2_COM = 0;
        R7SEG1_COM = 1;
        if( TSW==0N ) CN3 = disp[ l7seg2_value ] | 0x01 ; /* 0x01:0000 0001 */
        else          CN3 = disp[ l7seg2_value ] & 0xfe ; /* 0xfe:1111 1110 */
        lr7seg_select = 0; /* 次は右7SEG */

    } else {

        /* 右7セグメントLED表示 */
        //data = CN2 & 0xfe; /* 1111 1100 */
        //CN2 = data | 0x01;
        L7SEG2_COM = 1;
        R7SEG1_COM = 0;

        CN3 = disp[ r7seg1_value ] ;
        lr7seg_select = 1; /* 次は左7SEG */

    } /* if( RL7SEG_SELECT==1 ) */

    cnt10=0;

    } /* if( cnt10>10 ) */
}

/*-----*/
/* A/D値読み込み (AN7) */
/* 引数 なし */
/* 戻り値 A/D値 0~1023 */
/*-----*/
int get_ad7( void )
{
    int i;
    /* 繰り返しモード0は、自動的に繰り返すので、結果を読み込むだけ */
    i = ad7;

    return i;
}

/*-----*/
/* フルカラーLED制御 */
/* 引数 点灯する色の値 */
/* color_noと色点灯について */
/* full_color_led(0); 消灯 0:000 */
/* full_color_led(1); 赤点灯 1:001 */
/* full_color_led(2); 青点灯 2:010 */
/* full_color_led(3); 紫点灯 3:011 */
/* full_color_led(4); 緑点灯 4:100 */
/* full_color_led(5); 黄緑点灯 5:101 */
/* full_color_led(6); 水色点灯 6:110 */
/* full_color_led(7); 白点灯 7:111 */
/* 1:点灯 0:消灯 */
/*-----*/
void full_color_led( unsigned char led )
{
    unsigned char data;

    led <<= 2;
    data = CN2 & 0xe3; /* 1110 0011 */
    CN2 = data | led;
}

/*-----*/
/* メイン処理 */
/*-----*/
void main(void)
{
    unsigned char i;
    int ad;
    int color_no; /* カラーNO */

    init(); /* r8c/35aポート設定 */

    /*-----*/
    /* 動作確認用にLED点滅(なくともよい) */
    /*-----*/
    for( i=0; i<2; i++ ) {
        p1_3=1; p1_2=1; p1_1=1; p1_0=1;
        timer(200);
    }
}

```

```

        p1_3=0; p1_2=0; p1_1=0; p1_0=0;
        timer(200);
    }

asm(" fset I "); /* 全体の割り込み許可 */

color_no = 0;
full_color_led( color_no ); /* 初期状態 フルカラーLEDは消灯している */

timer(2000); /* 7セグメントLED初期表示「1.5」を表示させるため */

while(1) {
    /* PSD値読み込み */
    ad = get_ad7(); /* 0~1023 */ /* AN0(P0_0) */

    /* PSD値を距離に変換(事前計測) */
    /* 7セグメント表示値設定 */
    /* 7セグメントLED表示は割り込みにて処理 */
    if( ad>420 ) {
        l7seg2_value=1; r7seg1_value=0;
    } else if( ad>380 ) {
        l7seg2_value=1; r7seg1_value=5;
    } else if( ad>325 ) {
        l7seg2_value=2; r7seg1_value=0;
    } else if( ad>265 ) {
        l7seg2_value=2; r7seg1_value=5;
    } else if( ad>220 ) {
        l7seg2_value=3; r7seg1_value=0;
    } else if( ad>185 ) {
        l7seg2_value=3; r7seg1_value=5;
    } else if( ad>175 ) {
        l7seg2_value=4; r7seg1_value=0;
    } else if( ad>155 ) {
        l7seg2_value=4; r7seg1_value=5;
    } else if( ad>144 ) {
        l7seg2_value=5; r7seg1_value=0;
    } else {
        l7seg2_value=8; r7seg1_value=8;
    } /* end of if else */

} /* end of while(1) */

} /* end of main */

```