

```

/*-----*/
* ID      :なし
* 関数名  : main
* 機能    : 緑色、赤色LEDを点滅させる
* 引数    :なし
* 戻り値  :なし
* 使用関数:なし
*-----*/

/* インクルードファイル */
#include "sfr_r815.h" //SR8C15CPのI/Oレジスタが定義してあるSFRヘッダ (C言語用)
#include <stdio.h>
#include <string.h>

/* プロトタイプ宣言 */

/* I/Oの定義*/

/* 使用するI/Oレジスタの定義
 * LEDは高電流駆動能力が選択できるPORT 1 に配線されている
 */

#define LED_PORT          p1                // p1

#define LED_GREEN        p1_2              // 緑LEDはP1_1 (Pin15) に配線されている
#define LED_RED          p1_1              // 赤LEDはP1_0 (Pin17) に配線されている

#define LED_GREEN_DIR    pd1_2             // 緑LEDのポートの方向レジスタ (1を設定すると出力ポートに設定される)
#define LED_RED_DIR      pd1_1             // 赤LEDのポートの方向レジスタ (1を設定すると出力ポートに設定される)

#define LED_GREEN_DRV    drr2              // 使用するI/Oポートの駆動能力を設定するbit
#define LED_RED_DRV      drr1              // 1を設定すると駆動能力Highになり、LEDが駆動できる

#define TIM_PORT         tzmr              // タイマーZ
#define PRS_PORT         prez              // プリスケアラ
#define SEC_PORT         tzsc              // セカンダリ
#define PRI_PORT         tzpr              // プライマリ
#define OUT_PORT         tzoc              // 出力制御
#define WAV_PORT         pum                // 波形制御
#define CNT_PORT         tcss              // カウントソース

#define L_LOOP           86                 // Low toneループ
#define H_LOOP           45                 // High toneループ
#define M_LOOP           730               // muteループ

#define LO_S_0           0b00010000        // セカンダリ位相0
#define LO_P_0           0b00010000        // プライマリ位相0
#define LO_S_1           0b00010100        // セカンダリ位相1
#define LO_P_1           0b00001100        // プライマリ位相1
#define LO_S_2           0b00010110        // セカンダリ位相2
#define LO_P_2           0b00001010        // プライマリ位相2
#define LO_S_3           0b00010100        // セカンダリ位相3
#define LO_P_3           0b00001100        // プライマリ位相3
#define LO_S_4           0b00010000        // セカンダリ位相4
#define LO_P_4           0b00010000        // プライマリ位相4
#define LO_S_5           0b00001100        // セカンダリ位相5
#define LO_P_5           0b00010100        // プライマリ位相5
#define LO_S_6           0b00001010        // セカンダリ位相6
#define LO_P_6           0b00010110        // プライマリ位相6
#define LO_S_7           0b00001100        // セカンダリ位相7
#define LO_P_7           0b00010100        // プライマリ位相7

#define HI_S_0           0b00010000        // セカンダリ位相0
#define HI_P_0           0b00010000        // プライマリ位相0
#define HI_S_1           0b00010110        // セカンダリ位相1
#define HI_P_1           0b00001010        // プライマリ位相1
#define HI_S_2           0b00011000        // セカンダリ位相2
#define HI_P_2           0b00001000        // プライマリ位相2
#define HI_S_3           0b00010110        // セカンダリ位相3
#define HI_P_3           0b00001010        // プライマリ位相3
#define HI_S_4           0b00010000        // セカンダリ位相4
#define HI_P_4           0b00010000        // プライマリ位相4
#define HI_S_5           0b00001010        // セカンダリ位相5
#define HI_P_5           0b00010110        // プライマリ位相5
#define HI_S_6           0b00001000        // セカンダリ位相6
#define HI_P_6           0b00011000        // プライマリ位相6
#define HI_S_7           0b00001010        // セカンダリ位相7
#define HI_P_7           0b00010110        // プライマリ位相7

/*-----*/
* 関数名  : mute

```

```

* 機能      :
* 引数      :なし
* 戻り値    :なし
* 使用関数  :なし
* ----- */
void mute(void)
{
    int          w45d;          // 45degループカウンタ
    SEC_PORT     = LO_S_0;     // セカンダリ
    PRI_PORT     = LO_P_0;     // プライマリ
    for( w45d=0; w45d<M_LOOP; w45d++){ // 45deg ループ
//Rerurn
    return;
}
* ----- */
* 関数名    : h/ltone
* 機能      : 1200/2400Hzトーン
* 引数      :なし
* 戻り値    :なし
* 使用関数  :なし
* ----- */
void ltone(void)
{
    int          w45d;          // 45degループカウンタ
//Phase0
    SEC_PORT     = LO_S_0;     // セカンダリ
    PRI_PORT     = LO_P_0;     // プライマリ
    for( w45d=0; w45d<L_LOOP; w45d++){ // 45deg ループ
//Phase1
    SEC_PORT     = LO_S_1;     // セカンダリ
    PRI_PORT     = LO_P_1;     // プライマリ
    for( w45d=0; w45d<L_LOOP; w45d++){ // 45deg ループ
//Phase2
    SEC_PORT     = LO_S_2;     // セカンダリ
    PRI_PORT     = LO_P_2;     // プライマリ
    for( w45d=0; w45d<L_LOOP; w45d++){ // 45deg ループ
//Phase3
    SEC_PORT     = LO_S_3;     // セカンダリ
    PRI_PORT     = LO_P_3;     // プライマリ
    for( w45d=0; w45d<L_LOOP; w45d++){ // 45deg ループ
//Phase4
    SEC_PORT     = LO_S_4;     // セカンダリ
    PRI_PORT     = LO_P_4;     // プライマリ
    for( w45d=0; w45d<L_LOOP; w45d++){ // 45deg ループ
//Phase5
    SEC_PORT     = LO_S_5;     // セカンダリ
    PRI_PORT     = LO_P_5;     // プライマリ
    for( w45d=0; w45d<L_LOOP; w45d++){ // 45deg ループ
//Phase6
    SEC_PORT     = LO_S_6;     // セカンダリ
    PRI_PORT     = LO_P_6;     // プライマリ
    for( w45d=0; w45d<L_LOOP; w45d++){ // 45deg ループ
//Phase7
    SEC_PORT     = LO_S_7;     // セカンダリ
    PRI_PORT     = LO_P_7;     // プライマリ
    for( w45d=0; w45d<L_LOOP; w45d++){ // 45deg ループ
//Rerurn
    return;
}
* ----- */
* 関数名    : h/htone
* 機能      : 1200/2400Hzトーン
* 引数      :なし
* 戻り値    :なし
* 使用関数  :なし
* ----- */
void htone(void)
{
    int          w45d;          // 45degループカウンタ
//Phase0
    SEC_PORT     = HI_S_0;     // セカンダリ
    PRI_PORT     = HI_P_0;     // プライマリ
    for( w45d=0; w45d<H_LOOP; w45d++){ // 45deg ループ
//Phase1
    SEC_PORT     = HI_S_1;     // セカンダリ
    PRI_PORT     = HI_P_1;     // プライマリ
    for( w45d=0; w45d<H_LOOP; w45d++){ // 45deg ループ
//Phase2
    SEC_PORT     = HI_S_2;     // セカンダリ

```

```

    PRI_PORT      = HI_P_2;      // プライマリ
    for( w45d=0; w45d<H_LOOP; w45d++){ // 45deg ループ
//Phase3
    SEC_PORT      = HI_S_3;      // セカンダリ
    PRI_PORT      = HI_P_3;      // プライマリ
    for( w45d=0; w45d<H_LOOP; w45d++){ // 45deg ループ
//Phase4
    SEC_PORT      = HI_S_4;      // セカンダリ
    PRI_PORT      = HI_P_4;      // プライマリ
    for( w45d=0; w45d<H_LOOP; w45d++){ // 45deg ループ
//Phase5
    SEC_PORT      = HI_S_5;      // セカンダリ
    PRI_PORT      = HI_P_5;      // プライマリ
    for( w45d=0; w45d<H_LOOP; w45d++){ // 45deg ループ
//Phase6
    SEC_PORT      = HI_S_6;      // セカンダリ
    PRI_PORT      = HI_P_6;      // プライマリ
    for( w45d=0; w45d<H_LOOP; w45d++){ // 45deg ループ
//Phase7
    SEC_PORT      = HI_S_7;      // セカンダリ
    PRI_PORT      = HI_P_7;      // プライマリ
    for( w45d=0; w45d<H_LOOP; w45d++){ // 45deg ループ

//Phase0
    SEC_PORT      = HI_S_0;      // セカンダリ
    PRI_PORT      = HI_P_0;      // プライマリ
    for( w45d=0; w45d<H_LOOP; w45d++){ // 45deg ループ
//Phase1
    SEC_PORT      = HI_S_1;      // セカンダリ
    PRI_PORT      = HI_P_1;      // プライマリ
    for( w45d=0; w45d<H_LOOP; w45d++){ // 45deg ループ
//Phase2
    SEC_PORT      = HI_S_2;      // セカンダリ
    PRI_PORT      = HI_P_2;      // プライマリ
    for( w45d=0; w45d<H_LOOP; w45d++){ // 45deg ループ
//Phase3
    SEC_PORT      = HI_S_3;      // セカンダリ
    PRI_PORT      = HI_P_3;      // プライマリ
    for( w45d=0; w45d<H_LOOP; w45d++){ // 45deg ループ
//Phase4
    SEC_PORT      = HI_S_4;      // セカンダリ
    PRI_PORT      = HI_P_4;      // プライマリ
    for( w45d=0; w45d<H_LOOP; w45d++){ // 45deg ループ
//Phase5
    SEC_PORT      = HI_S_5;      // セカンダリ
    PRI_PORT      = HI_P_5;      // プライマリ
    for( w45d=0; w45d<H_LOOP; w45d++){ // 45deg ループ
//Phase6
    SEC_PORT      = HI_S_6;      // セカンダリ
    PRI_PORT      = HI_P_6;      // プライマリ
    for( w45d=0; w45d<H_LOOP; w45d++){ // 45deg ループ
//Phase7
    SEC_PORT      = HI_S_7;      // セカンダリ
    PRI_PORT      = HI_P_7;      // プライマリ
    for( w45d=0; w45d<H_LOOP; w45d++){ // 45deg ループ

//Rerurn
    return;
}

```

```

/*-----
* 関数名 : psend
* 機能   : 1200/2400Hz トーン
* 引数   : 送信データ
* 戻り値 : なし
* 使用関数 : なし
*-----*/

```

```

void bsend(char outa){
    int bout=0;
    int bsft;

    for(bsft=8; bsft>0; bsft--){
        bout = (outa & 0x80);
        if(bout != 0){
            ltone(); //mark
        }
        else{
            htone(); //space
        }
    }
}

```

```

        outa = outa << 1;
    }
}

/*-----
 * 関数名 : psend
 * 機能   : 1200/2400Hzトーン
 * 引数   : 送信データ
 * 戻り値 : なし
 * 使用関数: なし
 *-----*/

void psend(char *outa_p){
//Mark 1200Hz--0x01 Space 2200Hz--0x02

    char outa;
    int txdly;

        LED_GREEN    = 0;                // 赤LEDが点灯する
        LED_RED      = 1;

//Flag
    for(txdly=100; txdly>0; txdly--){
        outa = 0x7F;                    //01111111(NRZI)
        bsend(outa);
    }

//data
    while((outa = *outa_p) != '¥0'){
        bsend(outa);
        outa_p++;                        //Next byte
    }

        LED_GREEN    = 1;                // 緑LEDが点灯する
        LED_RED      = 0;

//Mute
    for(txdly=1000; txdly>0; txdly--){
        mute();
    }
}

/*-----
 * 関数名 : main
 * 機能   : MARK/SPACEを発生させる
 * 引数   : なし
 * 戻り値 : なし
 * 使用関数: なし
 *-----*/

main(void)
{
    /* I/Oポート初期化*/
    LED_GREEN_DRV = 1;                    // LEDを駆動できるように、1を設定する
    LED_RED_DRV   = 1;                    // この設定はP1_0~P1_3でBit単位で設定できる
    LED_PORT      = 0b00001000;         //
LEDの初期状態(赤 消灯、青 点灯)をポートレジスタに設定する この時点ではまだ入力ポートの設定になっている
    LED_GREEN_DIR = 1;                    // LEDの接続されているI/Oポートを出力に設定する
    LED_RED_DIR   = 1;
    LED_GREEN     = 1;                    // 両方のLEDを点灯する
    LED_RED       = 1;

    /* PWMポート初期化*/
    TIM_PORT      = 0b10010000;         // タイマーZ
    PRS_PORT      = 0b00000000;         // プリスケアラ
    SEC_PORT      = 0b01000000;         // セカンダリ
    PRI_PORT      = 0b10000000;         // プライマリ
    OUT_PORT      = 0b00000000;         // 出力制御
    WAV_PORT      = 0b00000000;         // 波形制御
    CNT_PORT      = 0b00000000;         // カウントソース

    /* 試験回路 */
    while(0){
        int loop;

        LED_GREEN    = 0;                // 赤LEDが点滅する
        LED_RED      = 1;
        for(loop=100; loop>0; loop--){
            ltone();
            htone();
            htone();

```

```

        htone();
        htone();
        htone();
        htone();
        htone();
    }
    LED_GREEN      = 1;          // 緑LEDが点滅する
    LED_RED        = 0;
    for(loop=300; loop>0; loop--){
        mute();
    }
}

/* 擬似回路 */
while(1){
    char data[32] ; //data
    char *outa_p;    //pointer

    outa_p = &data[0];

//NRZI capture data
//char *data0 = "01111111 01111111 01111111 01111111 01111111 01111111 01111111 01111111 ¥0";
//char *data1 = "01110101 10010110 01010100 10101011 01010100 10101011 01010111 01001101 ¥0";
//char *data2 = "10001101 10010111 01100010 01011101 10010001 11010111 11101010 10101000 ¥0";
//char *data3 = "00010001 01000010 10101101 00011111 11011111 11 ¥0";

//trace data
//000: 86A24040 40406094 96629A98 B2E103F0 CQ OJK1MLYp.x ..@@@`..b.....
//010: 0D
//JK1MLY>CQ <UI R>:

    data[ 0] = 0x07F;
    data[ 1] = 0x07F;
    data[ 2] = 0x07F;
    data[ 3] = 0x07F;
    data[ 4] = 0x07F;
    data[ 5] = 0x07F;
    data[ 6] = 0x07F;
    data[ 7] = 0x07F;
    data[ 8] = 0x075;
    data[ 9] = 0x096;
    data[10] = 0x054;
    data[11] = 0x0AB;
    data[12] = 0x054;
    data[13] = 0x0AB;
    data[14] = 0x057;
    data[15] = 0x04D;
    data[16] = 0x08D;
    data[17] = 0x097;
    data[18] = 0x062;
    data[19] = 0x05D;
    data[20] = 0x091;
    data[21] = 0x0D7;
    data[22] = 0x0EA;
    data[23] = 0x0A8;
    data[24] = 0x011;
    data[25] = 0x042;
    data[26] = 0x0AD;
    data[27] = 0x01F;
    data[28] = 0x0DF;
    data[29] = 0x0DF;
    data[30] = 0x0DF;
    data[31] = 0x000;

    outa_p = &data[0];

    psend(outa_p);

    LED_GREEN      = 0;          // LEDが消灯する
    LED_RED        = 0;
}
}

```