

## テーマ：イベントフラグによる通信

```
//user.c

/*****
*** reset process      ***
*****/
void uinit(void)
{
    /* initialize */
    clock_switch();
    lcd_init();
    uart1_init();
    trb_init();
    asm("FSET I");
}

/*****
*** tasks              ***
*****/
void tsk01(void) {
    W flgcd;
    while(1){
        flgcd = wai_flg(1,0x00ff);
        flgcd = flgcd & 0x00ff;
        lcd_xy(0,0);
        lcd_char((UB)flgcd);
    }
}

void tsk02(void) {
    while(1) {
        slp_tsk();
    }
}

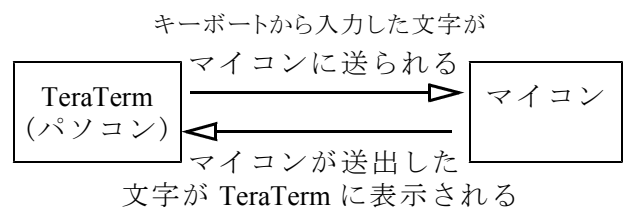
void tsk03(void) {
    while(1) {
        slp_tsk();
    }
}
```

これは、シリアル通信で受信したデータを LCD に表示するプログラムである。

**ポイント 1** シリアル通信でもこの方法を使ってデータを受信している。

**ポイント 2** TeraTerm に表示される文字は、TeraTerm がシリアル通信で受信した文字である。よって、TeraTerm に文字を表示させるためには、マイコンが受信した値を再びパソコンに送り返す。この処理のことをエコーバックという。

イベントフラグを利用して  
シリアル通信からデータが来るのを待つ



【実習 1】 実習 1 のプログラムを TeraTerm の画面にも入力したデータが表示されるように変更しなさい。

ヒント：put\_uart1()

```
//si_uart1.c

/*
* SCI interruption
*/
void si_int(H mode)
{
    UB data; // 受信データ

    if(get_uart1(&data) == 1){
        iset_flg(1, data);
    }
    disp(mode);
}
```

☆シリアル通信割り込み  
シリアル通信でデータを受信すると  
割り込みが発生する

イベントフラグを利用して  
シリアル通信のデータを tsk01  
に届ける

```
//user.c

/*****
*** reset process *****/
/*****/
void uinit(void)
{
    /* initialize */
    clock_switch();
    bz_init();
    lcd_init();
    uart1_init();
    trb_init();
    asm("FSET I");
}

/*****/
/*****/
void tsk01(void) {
    while(1) {
        bz_on();
        tslp_tsk(2);
        bz_off();
        tslp_tsk(2);
    }
}

void tsk02(void) {
    UB c;
    lcd_clr();
    sus_tsk(1);
    while(1) {
        c = wai_flg(1,0x00ff);
        if(c=='¥r'){
            lcd_clr();
            put_uart1('¥n');
        }
        else{
            lcd_char(c);
        }
        put_uart1(c);

        switch(c){
            case 'b':
                rsm_tsk(1);
                break;
            case 's':
                sus_tsk(1);
                break;
        }
    }
}

void tsk03(void) {
    UB k;
    while(1) {
        slp_tsk();
    }
}
```

これは、キーボードから **b** を入力したときにブザーを鳴らし、キーボードから **s** を入力したときにブザーを止め、さらに、入力した文字を LCD とターミナルに表示するプログラムである。

【実習 2】 右のプログラムをキーボードから **b** を入力したときに LCD に “ブザー ON” と表示し、**s** を入力したときには、“ブザー OFF” と表示するように改良しなさい。

ヒント：

```
void tsk02(void) {
    UB f;
    while(1) {
        f=wai_flg(2,0x0003);
        lcd_clr();
        if(f==0x0001)lcd_str((B*)"ブザー - ON");
        if(f==0x0002)lcd_str((B*)"ブザー - OFF");
    }
}
```

【実習 3】 実習 2 のプログラムをターミナルにも “ブザー ON”、“ブザー OFF” と表示するように改良しなさい。

ヒント： PutStr("ブザー-ON¥r¥n")  
PutStr("ブザー-OFF¥r¥n")

si\_uart1. c について

void uart1_init(void)	SI を初期化する
H put_uart1(UB r)	SCI に 1 文字送信する ch 文字コード
H get_uart1(UB *s)	SCI から 1 文字受信する