```
/**********************************************
 *  LCD表示関数：I2Cインターフェースを呼出し使用
 *    By nobcha (c)2011,2012
 *
 *    STRAWBERRY  LINUX社LCD アドレス 0x7c
 *      AN001とかST7032iのデータシート参照
 *      Key-pad support read
 *
 *      PIC16F1827対応に修正  2012.09.25  LCD+Key
 *
 *    パラレル4ビット接続のときの関数に合わせる
 *    i2c_sl_lcd_init(void)     ----- 初期化
 *    i2c_sl_lcd_cmd(command)   ----- コマンド出力
 *    i2c_sl_lcd_data(byte)     ----- １文字表示
 *    i2c_sl_lcd_str(string)    ----- 文字列表示
 *    i2c_sl_lcd_clear(void)    ----- LCD画面全消去
 *    i2c_sl_key_get(void) ------------- キー入力取得
 **********************************************/
#define _LEGACY_HEADERS
#define LCD_AD  0x7C
#define _XTAL_FREQ 8000000
#define PIC_CLOCK  8000000

#include <htc.h>
#include "mssp_i2c.h"
#include "lcd_key_i2c_sl_mssp.h"
#include "lcd.h"
/*************************************
 *  コントラスト調整用定数  電圧依存
 *************************************/
// #define CONTRAST     0x20    // for 5.0V
 #define CONTRAST       0x2F    // for 3.3V
// #define CONTRAST     0x3F    // for 2.6V
/*********************************
 *  記号アイコン表示定義
 *********************************/
unsigned char ICON[14][2]={
        {0x00, 0x10},           // アンテナ
        {0x02, 0x10},           // 電話
        {0x04, 0x10},           // 無線
        {0x06, 0x10},           // ジャック
        {0x07, 0x10},           // △
        {0x07, 0x08},           // ▽
        {0x07, 0x18},           // △▽
        {0x09, 0x10},           // 鍵
        {0x0B, 0x10},           // ピン
        {0x0D, 0x02},           // 電池無し
        {0x0D, 0x12},           // 容量少
        {0x0D, 0x1A},           // 容量中
        {0x0D, 0x1E},           // 容量多
        {0x0F, 0x10}            // 丸
};

/******************************
 *  LCDへ表示データ1文字出力
 ******************************/
void i2c_sl_lcd_data(unsigned char data){
        i2c_writeto(LCD_AD);            // LCDアドレスをOPEN

        i2c_putbyte(0x40);              // CO:0 (STOP) ,RS:1  データなので

        i2c_putbyte(data);              // 表示バイトを送る

        i2c_stop();                     // stop コンディション
        __delay_us(50);                 // 10usec待ち
        }

/******************************
 * 液晶へ１コマンド出力
 ******************************/
void i2c_sl_lcd_cmd(unsigned char cmd){
        i2c_writeto(LCD_AD);            // LCDアドレスをOPEN

        i2c_putbyte(0x00);              // CO:0 (STOP) ,RS:0  コマンドです
```

```
        i2c_putbyte(cmd);                       // コマンドバイトを送る

        i2c_stop();                                     // stop

        if((cmd == 0x01)||(cmd == 0x02))        // ClearかHomeかで待ち時間調整
                __delay_ms(2);                  // Clearなら応答時間は2msec待ち
        else
                __delay_us(50);                 // それ以外コマンドなら50μsec待ち
        }

/******************************
 * 液晶画面出力カーソル位置指定
 ******************************/
void i2c_sl_lcd_goto(unsigned char cursor){     // カーソルアドレスは00～0F,40～4F
        i2c_sl_lcd_cmd(0x80 | cursor );         // DRAMアドレス設定#7：1にして7ビット
        __delay_us(100);                                        // 100μsec待ち
        }

/******************************
 *  LCD初期化処理
 ******************************/
void i2c_sl_lcd_init(void) {                    // ST7032iマニュアル掲載お約束
        __delay_ms(100);                        // 100mS delay  電源安定まで待つ
        i2c_sl_lcd_cmd(0x38);                   // 8bit 2line Normal mode
        i2c_sl_lcd_cmd(0x39);                   // 8bit 2line IS mode
        i2c_sl_lcd_cmd(0x14);                   // OSC 183Hz BIAS 1/5
        i2c_sl_lcd_cmd(0x70 + (CONTRAST & 0x0F));       // コントラスト設定
        i2c_sl_lcd_cmd(0x5C + (CONTRAST >> 4));
//      i2c_sl_lcd_cmd(0x6A);                   // Follower for 5.0Vの場合
        i2c_sl_lcd_cmd(0x6B);                   // Ffollwer for 3.3Vの場合
        __delay_ms(100);                        // 400mS待ち
        __delay_ms(100);
        __delay_ms(100);
        __delay_ms(100);
        i2c_sl_lcd_cmd(0x38);                   // Set back to  Normal mode
        i2c_sl_lcd_cmd(0x0C);                   // Display On
        i2c_sl_lcd_cmd(0x01);                   // Clear Display
        }

/******************************
 * LCD画面全消去
 ******************************/
void i2c_sl_lcd_clear(void) {
        i2c_sl_lcd_cmd(0x01);                   //初期化コマンド0ｘ01出力
        }

/******************************
 * 文字列出力
 ******************************/
void i2c_sl_lcd_str(char* ptr) {
        while(*ptr != 0){                       //  0ｘ00があるまで文字取り出し
                i2c_sl_lcd_data(*ptr++);}       //  文字表示して、次の文字へ

        }

/******************************
 *  オプションアイコン表示
 ******************************/
 void i2c_sl_lcd_icon(unsigned char num, unsigned char onoff){
        char cmd;
        i2c_sl_lcd_cmd(0x39);                   // Extend modeにします
        cmd=0x40 | (ICON[num][0]);              // アイコン表示のお約束
        i2c_sl_lcd_cmd(cmd);                    // アイコンアドレス指定
        if(onoff){
                cmd= (ICON[num][1]);
                i2c_sl_lcd_data(cmd);           // アイコンオンデータ
                }
        else                                            // 消す時は0ｘ00かく
                i2c_sl_lcd_data(0x00);          // アイコンオフデータ
        i2c_sl_lcd_cmd(0x38);                   // Normal Mode
}

/******************************
 *  キーパッド INPUT
 ******************************/
```

```
unsigned char i2c_sl_key_get(){
        unsigned char key_data;
        i2c_readfrom(LCD_AD);

        key_data=i2c_read();
        i2c_sendack(1);
        i2c_stop();
        __delay_us(50);                         // 10usec待ち
        return key_data;
}
void i2c_sl_key_run(unsigned char status){
        i2c_sl_lcd_cmd(0x10|status);            // 0x11:start,0x10:stop
}
```