

```

/* *****
PhotoTR sensing of PIC12F683
    By nobcha all right reserved

Ver 1.   09/16/2011 for i2c LCD TEST
Ver 1.1  09/17/2011 for photoTR sensing
Ver 1.2  09/19/2011 for ULPWUE implemented

PIC12F683
PIN Assign  #7 GP100:ULPWUE
             #6 GP101:LED drive
             #5 GP102:photo monitor
             #3 GP104:debug:SCL
             #2 GP105:debug:SDA

OSC INT 4MHz

Development Circumstance
MPLAB IDE V8.73  HiTECH C V9.81

*****
*/

#define _LEGACY_HEADERS
#define _XTAL_FREQ 4000000

#define DEBUG

#define led_port GPI01

#include <htc.h>
#include "delay.h"
#include "lcd_i2c_func.h"

__CONFIG(BORDIS & UNPROTECT & PWRRTEN & WDTDIS & MCLREN & INTIO);

//
unsigned char Msg1[17] = "SLEEP TEST ";
unsigned char Msg2[7] = "DARKEN";
unsigned char Msg3[7] = "QUIET ";
short bright_ave=10;

// Photo TR check darken:0 or quiet:1
unsigned char photo_tr( void ){
    short bright;
    unsigned char photo_tr;
    GODONE=1; // ADC start
    __delay_us(100);
    while(GODONE){ // wait for ADC function
        bright=ADRESH<<8 | ADRESL; // Get ADC value ADRESL

        bright_ave=(bright_ave*3+ bright)/4; // averaging
        if(((bright_ave-bright)>2) & bright<5 ){
            photo_tr=0; // darken
            bright_ave=2;
        }
        else{ photo_tr=1; } // bright or still dark

#ifdef DEBUG
// debug display

        lcd_goto(0x40);
        lcd_data( bright/100%10 | 0x30 ); // Display digit
        lcd_data( bright/10%10 | 0x30 ); // Display digit
        lcd_data( bright%10 | 0x30 ); // Display digit
        lcd_data(0x20);

        lcd_data(bright_ave/100%10 | 0x30); // Display digit
        lcd_data(bright_ave/10%10 | 0x30); // Display digit
        lcd_data(bright_ave%10 | 0x30); // Display digit
        lcd_goto(0x48);
//

```

```

#endif
        return(photo_tr);
    }
// LED drive
void led_chop(char on_off){
    unsigned char i; // period about 10us
    led_port=on_off; // if switch is 1
    led_port=0; // inductor drive ON
    __delay_us(4); // inductor drive OFF
    // 4us waiting
}

void led_on(char bright){
    unsigned char j; // dimming by bright value
    for (j=50;j>0;j--){ // if bright is more than 50, turn on LED
        if(bright>50){ led_chop(1); } // 10uS*50=1mS
        else{ led_chop(0); } // LED on
        bright--; // LED off
    } // PPM bright control
}

// Ultra low power wake up
void ulpwu(void){
    GPIO0 = 1; // GPIO0:1
    TRIS100 = 0; // Charge start
    __delay_ms(1); // Wait 2*PI*C*R*10

    INTCON = 0b10001000; // GIE:1, GPIE:1
    TRIS100 = 1; // Start discharge

    asm("SLEEP");
    asm("NOP");
}

// interrupt by
void interrupt wakeup(void) {
    if ( GPIF ) {
        GPIE = 0; // interrupt on change
    }
}

void main(void) {
    unsigned char i, j;

    OSCCON = 0b01100000; // Set 4MHz
    GPIO = 0b00110001; // GPIO4, I05 High
    TRIS10 = 0b00110100; // GPIO0, 02, 04, 05 INPUT

    ANSEL = 0b01010100; // Enable AN2
    ADCON0 = 0b10001001; // AN3
    ADFM=1; // 8bit left

    CMCON0 = 0b00000111; // No using compalator
    PCON = 0b00100000; // ULPWUE enabled
    I0C = 0b000000001; // Int on change I0C0

#ifdef DEBUG
    __delay_ms(100);

    lcd_init(); // LCD initialize

    lcd_goto(0x00); // Move cursor 1st line
    lcd_str(Msg1); // Display test message
#endif

    while(1) {
        ulpwu(); // Ultra low power wake up
    }
}

```

```

683_i2c_lcd_sleep_main
if( photo_tr() == 0 ){
    lcd_str(Msg2); // darken?
    for(i=100; i>0; i--){ // Yes dark
        for(j=100; j>0; j--){ // 5sec
            led_on(i+50); // 50ms
            led_on(i+50); // 0.5ms bright control
        }
    }
}

#ifdef DEBUG
else{lcd_str(Msg3);} // QUIET
#endif
}
}

```