```
// thermo3_skel.c
// R. Traylor
// 11.15.2011 (revised 11.18.2013)

//Demonstrates basic functionality of the LM73 temperature sensor
//Uses the mega128 board and interrupt driven TWI.
//Display is the raw binary output from the LM73.
//PD0 is SCL, PD1 is SDA.

#include <avr/io.h>
#include <avr/interrupt.h>
#include <util/delay.h>
#include <string.h>
#include <stdlib.h>
#include "hd44780.h"
#include "lm73_functions.h"
#include "twi_master.h"
#include "uart_functions_m48.h"

char    lcd_string_array[16];  //holds a string to refresh the LCD
uint8_t i;                     //general purpose index

extern uint8_t lm73_wr_buf[2];
extern uint8_t lm73_rd_buf[2];

ISR(USART_RX_vect) {
    //uint8_t size = 17;

    //inputBuffer
}



/***********************************************************************/
/*                                  main                               */
/***********************************************************************/
int main ()
{
    uint16_t lm73_temp;  //a place to assemble the temperature from the lm73
    //uint8_t tempNum[1] = {0};
    char tempChar[3];
    const uint8_t address = 0b10010000;     // Model 0, pin floating

    DDRD |= (1<<PD1); // Port D 1 is transmit
    //DDRB |= 0xFF;
    //DDRD |= 0xFF;

    uart_init();

    init_twi(); //initalize TWI (twi_master.h)

    //set LM73 mode for reading temperature by loading pointer register
    lm73_wr_buf[0] = 0x00; //load lm73_wr_buf[0] with temperature pointer addres
s
    twi_start_wr(address, lm73_wr_buf, 2); //start the TWI write process
    _delay_ms(2);    //wait for the xfer to finish

    sei();

    while(1){           //main while loop
        //uart_putc('h');
        _delay_ms(10);
```

```

        twi_start_rd(address, lm73_rd_buf, 2); //read temperature data from LM73
 (2 bytes)
        _delay_ms(2);    //wait for it to finish
        lm73_temp = lm73_rd_buf[0]; //save high temperature byte into lm73_temp
        lm73_temp = lm73_temp << 8; //shift it into upper byte
        lm73_temp |= lm73_rd_buf[1]; //"OR" in the low temp byte to lm73_temp
        lm73_temp_convert(tempChar, lm73_temp, 1); //convert to string in array
with itoa() from avr-libc

        while(uart_getc() != ’t’ ) {}   // wait for ’t’
        _delay_ms(2);

        //uart_putc(’y’);
        //uart_putc(’y’);
        uart_putc(tempChar[0]);
        _delay_ms(1);
        uart_putc(tempChar[1]);
        //uart_putc(0);

    } //while
} //main
```