

Εργαστήριο Μικροπολογιστών

Παπαδόπουλος Κωνσταντίνος el20152
Σκούρτης Παύλος el20052

ΑΣΚΗΣΗ 7.2

#define F_CPU 16000000UL	
#include<avr/io.h>	
#include<avr/interrupt.h>	
#include<util/delay.h>	
#include <stddef.h>	
#include <string.h>	
#include "lcd.h"	Επικοινωνία με την LCD μέσω PIND
#include "twi.h"	
uint16_t one_wire_reset(){	Οι ακόλουθες συναρτήσεις είναι η υλοποίηση της assembly σε c. Είναι σημαντικό να αναφέρουμε ότι οι χρόνοι καθυστέρησης (delay),πρέπει να είναι ακριβείς, διαφορετικά δημιουργείται πρόβλημα
DDRD =0x10;	
PORTD &=0b11101111;	
_delay_ms(0.48);	
DDRD &=0b11101111;	
PORTD &=0b11101111;	
_delay_ms(0.1);	
uint16_t temp = PIND;	
_delay_ms(0.38);	
temp &=0b00010000;	
if(temp==0b00000000){	
return 0x0001;	
}	
else{	
return 0x8000;	
}	
}	
uint16_t one_wire_receive_bit(){	

DDRD =0x10;	
PORTD &=0b11101111;	
_delay_ms(0.002);	
DDRD &=0b11101111;	
PORTD &=0b11101111;	
_delay_ms(0.01);	
uint16_t temp=PIND;	
_delay_ms(0.38);	
temp &= 0b00010000;	
if(temp==0b00010000){	
 temp=1;	
}	
else{	
 temp=0;	
}	
_delay_ms(0.049);	
return temp;	
}	
void one_wire_transmit_bit(uint16_t data){	
 uint16_t temp=data;	
 DDRD =0x10;	
 PORTD &=0b11101111;	
 _delay_ms(0.002);	
 temp &=0x0001;	
 temp=temp<<4;	
 PORTD=temp;	
 _delay_ms(0.058);	
 DDRD &=0b11101111;	

PORTD &=0b11101111;	
_delay_ms(0.01);	
}	
uint16_t one_wire_receive_byte(){	
uint16_t temp,result;	
result=0x0000;	
for(int i=0;i<16;i++){	
temp=one_wire_receive_bit();	
result=result>>1;	
if(temp==1){	
result =0x8000;	
}	
}	
return result;	
}	
void one_wire_transmit_byte(uint16_t data){	
uint16_t temp=data;	
uint16_t temp2=0x01;	
for(int i=0;i<8;i++){	
if((temp 0b11111110)==0b11111111){	
one_wire_transmit_bit(temp2);	
}	
else one_wire_transmit_bit(0x00);	
temp=temp>>1;	
}	
}	

int presicion (int a){	
while (a%10 ==0 && a!=0)a/=10;	
return a;	
}	
int main(void){	
twi_init();	
PCA9555_O_write(REG_CONFIGURATION_O, 0x00);	
uint16_t init1,init2,result,temp;	
lcd_init();	
while(1){	Συνεχής λειτουργία του προγράμματος
init1=one_wire_reset();	Έλεγχος ύπαρξης συσκευής
lcd_clear_display();	
if(init1==0x0001){	
one_wire_transmit_byte(0xCC);	
one_wire_transmit_byte(0x44);	Έναρξη μέτρησης θερμοκρασίας
while(one_wire_receive_bit()==0);	Αναμονή μέτρησης
}	
init2=one_wire_reset();	Εκ νέου αρχικοποίηση της συσκευής
if(init2==0x0001){	Έλεγχος ύπαρξης συσκευής
one_wire_transmit_byte(0xCC);	
one_wire_transmit_byte(0xBE);	Ανάγνωση θερμοκρασίας
result=one_wire_receive_byte();	result—> περιέχει τη τιμή της θερμοκρασίας που επέστρεψε η συσκευή

temp=result;	
float deci;	
char string1[5],string2[5];	
if(result > 0x0800){	
 result =~ result;	Συμπλήρωμα ως προς 2 για σωστή ανάγνωση τς
 result +=1;	
 }	
deci=(float)(result & 0x0F)/16.0;	Το δεκαδικό μέρος ως float
int value2 = presicion((int)(deci*1000));	Το δεκαδικό μέρος ως ακέραιος
result = result>>4;	
sprintf(string1, "%d", result);	
sprintf(string2, "%d", value2);	
char data[10];	
strcpy(data, string1);	
strcat(data, ".");	
strcat(data, string2);	data—> η θερμοκρασία ως string
if(result > 0x0800){	
 lcd_data('-');	Εάν η θερμοκρασία είναι αρνητική
 }	
else{	
 lcd_data('+');	
 }	
lcd_data_string(data);	Εκτύπωση της θερμοκρασίας
lcd_data((char)223);	Σύμβολο °
lcd_data('C');	
_delay_ms(1000);	
}	

else if(init2==0x8000){	Εάν δεν υπάρχει συνδεδεμένη συσκευή
 lcd_data_string("No Device");	
 _delay_ms(1000);	
 }	
 }	
}	