Derek Nhan

CPE301 – SPRING 2016

Design Assignment 3

**DO NOT REMOVE THIS PAGE DURING SUBMISSION:**

The student understands that all required components should be submitted in complete for grading of this assignment.

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **SUBMISSION ITEM** | **COMPLETED (Y/N)** | **MARKS**  **(/MAX)** |
| 0. | COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS |  |  |
| 1. | INITIAL CODE OF TASK 1/A |  |  |
| 2. | SCHEMATICS |  |  |
| 3. | SCREENSHOTS OF EACH TASK OUTPUT |  |  |
| 4. | SCREENSHOT OF EACH DEMO |  |  |
| 5. | FLOW CHART |  |  |
| 6. | VIDEO LINKS OF EACH DEMO |  |  |
| 7. | GOOGLECODE LINK OF THE DA |  |  |
| 8. |  |  |  |
| 9. |  |  |  |
| 10. |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 0. | COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS |  |  |

-LM34 Temperature Sensor

-ATmega32p Xplained

-FTDI232 USB-to-Serial Chip

|  |  |  |  |
| --- | --- | --- | --- |
| 1. | INITIAL CODE OF TASK 1/A |  |  |

#define *F\_CPU* 8000000UL

#include <avr/io.h> //standard AVR header

#include <util/delay.h> //delay header

#include <avr/interrupt.h>

void usart\_init (void)

{

UCSR0B = (1<<TXEN0); //enable transmitt only

UCSR0C = ((1<<UCSZ01)|(1<<UCSZ00)); //removed (1<<UMSEL00) because it sets in synchronous mode, we are using asynchronous

UBRR0H = 0x00; //high value of baud rate

UBRR0L = 0x33; //baud rate of 9600

}

volatile unsigned int ADCvalue; //adc value buffer

void init\_328(void)

//subroutine to initialize all counter/adc used in the application

{

ADMUX|=(1<<REFS0); //use Vcc as ref (0.1uF cap attached)

ADCSRA|=(1<<ADSC)|(1<<ADEN)|(1<<ADIE)|(1<<ADPS2)|(1<<ADPS1)|(1<<ADPS0)|(1<<ADATE);

//start conversion; enable adc; enable interrupts; select 128 prescaler;

//auto trigger enable; ADC0 used

ADCSRB=0;

//adc in free running mode

}

void usart\_send(unsigned char ch)

{

while (!(UCSR0A & (1<<UDRE0))); //wait here until the buffer done writing

UDR0=ch; //set ch to UDR0 buffer to send to terminal

}

ISR (ADC\_vect)

{

char temperature[2] = "00"; //temperature string array

char temp; //temp place holder

char header[13]= "Temperature: "; //header for temperature

ADCvalue=ADC; //reads all 16-bits of ADCH:ADCL

ADCvalue=(ADCvalue)\*(500.0/1024.0); //convert the read ADCvalue to temperature

//500.0=>(Vref \* 100)=>(5V \* 100)

//divide by 1024, the max for the ADC values (0-1024)

temp=(ADCvalue/10); //divide by 10 to get the 10's place digit

temperature[0]=temp+48; //add ascii '0' to display on LCD

temp=(ADCvalue%10); //modulo by 10 to get 1's place digit

temperature[1]=temp+ 48; //add ascii '0' to get display on LCD

for (int i; i<13; i++)

usart\_send(header[i]); //loop to send the header "Temperature: "

usart\_send(temperature[0]); //sends 10s space of temperatue

usart\_send(temperature[1]); //sends 1 space of temperature

usart\_send('°'); //sends degree sign

usart\_send('F'); //sends F

usart\_send('\n'); //send line feed

*\_delay\_ms*(1000); //wait 1 seconds before retreiving data again

}

int main(void)

{

usart\_init(); //initialize usart

init\_328(); //initialize the adc

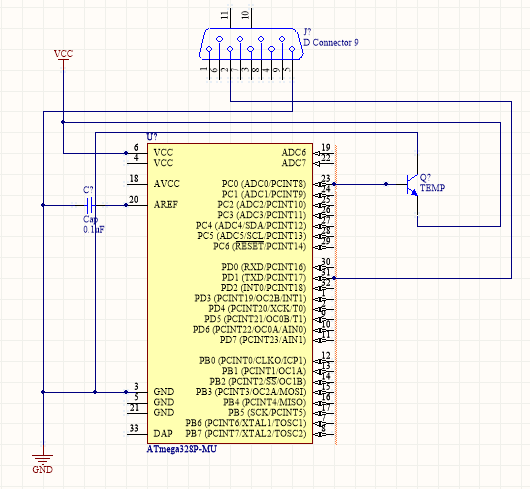
sei(); //enable interrupts

while(1);

return 0;

}

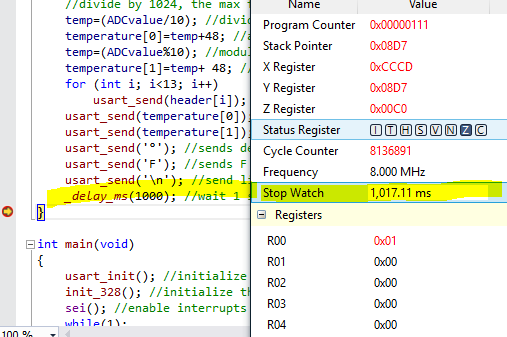
|  |  |  |  |
| --- | --- | --- | --- |
| 2. | SCHEMATICS |  |  |



|  |  |  |  |
| --- | --- | --- | --- |
| 3. | SCREENSHOTS OF EACH TASK OUTPUT |  |  |

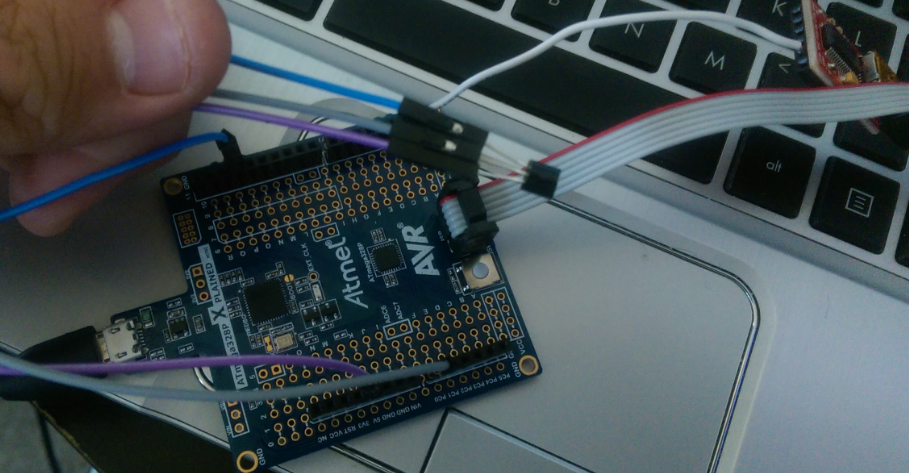
TASK 1/A:

After 1 second delay, we check the temperature again and update it to the terminal

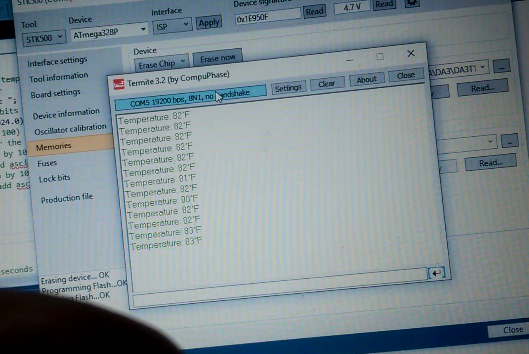


|  |  |  |  |
| --- | --- | --- | --- |
| 4. | SCREENSHOT OF EACH DEMO |  |  |

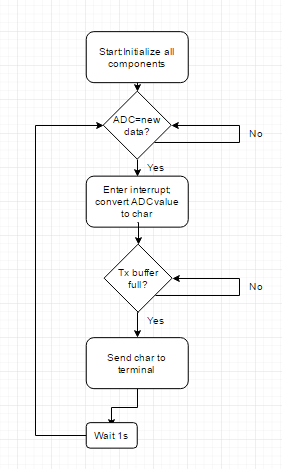
The LM34 connected directly to the ATmega328p Xplained. The FTDI232 USB to serial is used to communicate with the PC.



Terminal showing changes in the temperature every 1 second.



|  |  |  |  |
| --- | --- | --- | --- |
| 5. | FLOW CHART |  |  |



|  |  |  |  |
| --- | --- | --- | --- |
| 6. | VIDEO LINKS OF EACH DEMO |  |  |
| https://youtu.be/0Gl--mWpvP0 | | | |
| 7. | GOOGLECODE LINK OF THE DA |  |  |
| https://github.com/nhand2/CPE301S16 | | | |

**Student Academic Misconduct Policy**

<http://studentconduct.unlv.edu/misconduct/policy.html>

“This assignment submission is my own, original work”.

Derek Nhan