Audie Escala

CPE301 – SPRING 2018

Design Assignment 03

**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)** |
| 1 | COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS |  |  |
| 2. | INITIAL CODE OF TASK 1 |  |  |
| 3. | TASK 2 |  |  |
| 4. | SCREENSHOT OF BREADBOAD w/ ATmega328P |  |  |
| 6. | YOUTUBE & GITHUBECODE LINK OF THE DA3 |  |  |
|  |  |  |  |
|  |  |  |  |

1. **COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS**

List of Components used:

LM34

FDTI Sparkfun Breakout board

Block diagram with pins used in the Atmega328P

1. **INITIAL/DEVELOPED CODE IN ASSEMBLY OF TASK 1**

; ==============================================================================

; TASK 1

;===============================================================================

#define FOSC 16000000 // Clock Speed

#define BAUD 9600

#define MYUBRR FOSC/16/BAUD-1

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

#include <stdint.h> // need for uint8\_t

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

#include <avr/interrupt.h>

void USARTinit (void)

{

/\* set baud rate \*/

UBRR0H = (MYUBRR >> 8); //high value of baud rate

UBRR0L = MYUBRR; // low value of baud rate

UCSR0B |= (1 << RXEN0) | (1<<TXEN0); //enable receiver and transmitter

UCSR0B |= (1 << RXCIE0); // enable receiver input

UCSR0C = ((1<<UCSZ01)|(1<<UCSZ00)); //asynchronous

}

volatile *uint8\_t* ADCvalue; // Global Variable of the returned value from ADC

void ADCinit(void) // function for Initializing ADC

{

ADMUX |= (1 << REFS0); //use AVcc as ref

ADCSRA |= (1 << ADPS2) | (1 << ADPS1) | (1 << ADPS0); // 128 Prescale for 16MHz

ADCSRA |= (1 << ADATE); // Set ADC Auto Trigger Enable

ADCSRB = 0; // Free running mode

ADCSRA |= (1 << ADEN); // Enable the ADC

ADCSRA |= (1 << ADIE); // Enable interrupts

ADCSRA |= (1 << ADSC); // start conversion

}

void USARTsend(unsigned char Data) // function for sending data to the stream

{

while (!(UCSR0A & (1<<UDRE0)));

UDR0=Data;

}

ISR (ADC\_vect)

{

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

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

char temp; //temporary place holder

/\*

convert the read ADCvalue to temperature

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

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

\*/

ADCvalue = ADC;

ADCvalue = (ADCvalue)\*(500.0/1024.0);

/\* converts value into ascii \*/

temp = (ADCvalue/10); //divide by 10 for tenths place digit

temperature[0] = temp + 48; //add ASCII '0'

temp= (ADCvalue%10); //modulo by 10 for ones place digit

temperature[1] = temp + 48; //add ASCII '0'

/\* Displaying onto terminal \*/

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

USARTsend(repeatTemp[i]); //loop to send the header "Temperature: "

}

USARTsend(temperature[0]); //sends 10s space of temperature

USARTsend(temperature[1]); //sends 1 space of temperature

USARTsend('°'); //sends degree sign

USARTsend('F'); //sends F

USARTsend('\n'); //send line feed

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

}

int main(void)

{

USARTinit(); //initialize USART

ADCinit(); //initialize the ADC

sei(); //enable interrupts

while(1);

return 0;

}

1. **ATMEL Studio Data Visualizer Terminal Screenshot for TASK 2**



1. **Screenshot of breadboard w/ ATmega 328P Xplained Mini**

FDTI connected to the 328P Xplained Mini

Connected DTR to RST for resetting

Connected TVO to PD0 for transmitting

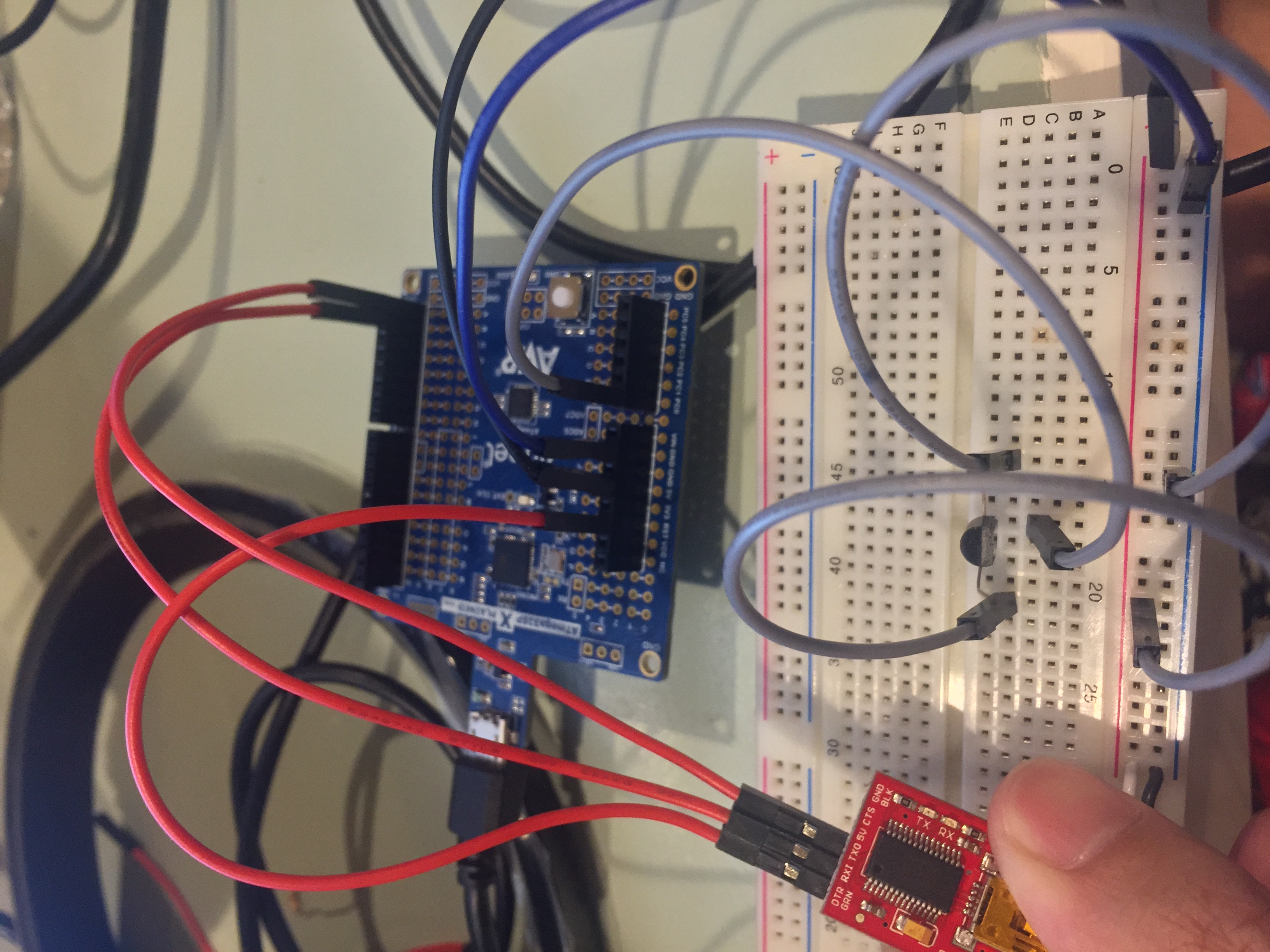
Connected RXI to PD1 for receiving

LM34 on the breadboard connected to the 328P Xplained Mini

Connected Input to 5V

Connected Output to PC0/ADC0

Connected GND to GND



1. **DEVELEPED CODE IN C TASK 4**

https://github.com/escalaa/DA03

**Student Academic Misconduct Policy**

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

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

Audie Escala