Noor blum

Design Assignment DA3

**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. | C CODE WITH COMMENTS FOR TASK | Y |  |
| 1. | SCHEMATIC | Y |  |
| 2. | SNAPSHOT OF THE BOARD WITH CONNECTED COMPONENTS | Y |  |
| 3. | SNAPSHOT OF THE COMPUTER SCREEN WITH TERMINAL | Y |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 0. | C CODE WITH COMMENTS FOR TASK |  |  |

/\*

\* HomeAuto.c

\*

\* Created: 22/02/2013 3:03:37 PM

\* Author: PC

\*/

#include <avr/io.h>

#include <avr/interrupt.h>

#include <stdio.h>

#define *F\_CPU* 16000000UL

#include <util/delay.h>

#define ADC\_CH 1;

#define USART\_BAUDRATE 9600

#define BAUD\_PRESCALE (((*F\_CPU*/(USART\_BAUDRATE\*16UL)))-1)

*uint16\_t* counter;

////////////////////////////////////////////////////////This is USART////////////////////////////////////////////////////////////

void USARTInit(unsigned int ubrr\_value)

{

/\*Set baud rate \*/

UBRR0H = (unsigned char)(ubrr\_value>>8);

UBRR0L = (unsigned char)ubrr\_value;

/\* Set frame format: 8data, 2stop bit \*/

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

/\*Enable receiver and transmitter \*/

UCSR0B = (1<<RXEN0)|(1<<TXEN0);

}

void USARTTransmitChar(unsigned char data)

{

/\* Wait for empty transmit buffer \*/

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

/\* Put data into buffer, sends the data \*/

UDR0 = data;

}

///////////////////////////////////////////////////////////////USART END/////////////////////////////////////////////////////////////

//////////////////////////////////////////////////////////////ADC START//////////////////////////////////////////////////////////////

void adc\_Init()

{

//DDRC=0x00;

ADMUX|=(1<<REFS0);

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

}

*uint16\_t* read\_adc(*uint8\_t* ch)

{

ADMUX = (ADMUX & 0xF0) | (ch & 0x0F);

ADCSRA|=(1<<ADSC);

while(!(ADCSRA&(1<<ADSC)));

return ADC;

}

///////////////////////////////////////////////////////////////ADC END////////////////////////////////////////////////////////////////////////////////

////////////////////////////////////////////////////////////TImer START//////////////////////////////////////////////////////////////////////////////

void timer\_init(){

TCCR0A = 0x00;

TCCR0B = 0x05; //clk/1024

TIMSK0 = 0x01;

}

ISR(TIMER0\_OVF\_vect){

if(counter == 61)

{

counter = 0;

*uint8\_t* data = read\_adc(0);

volatile float f = (((float)data/1024.0f)\*500.0f);

char ch[20];

*sprintf*(ch,"%d",(*uint8\_t*)f);

USARTTransmitChar(ch[0]);

USARTTransmitChar(ch[1]);

USARTTransmitChar('\n');

}

counter++;

}

////////////////////////////////////////////////////////////Timer End////////////////////////////////////////////////////////////////////////////////

int main(void)

{

USARTInit(BAUD\_PRESCALE);

USARTTransmitChar('g');

adc\_Init();

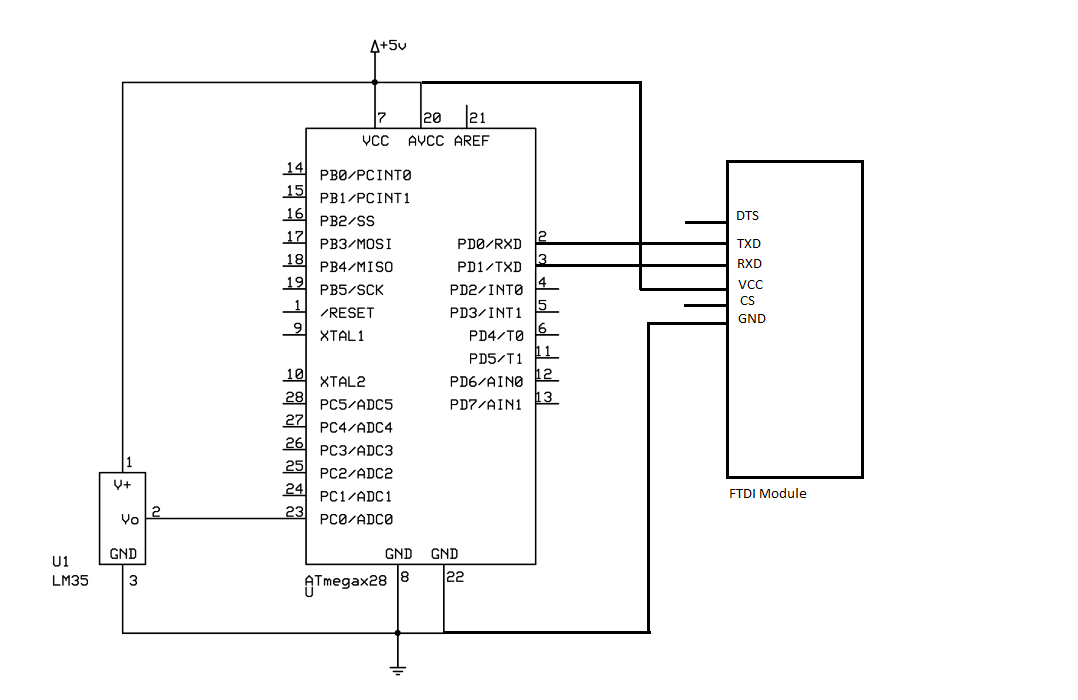
timer\_init();

sei();

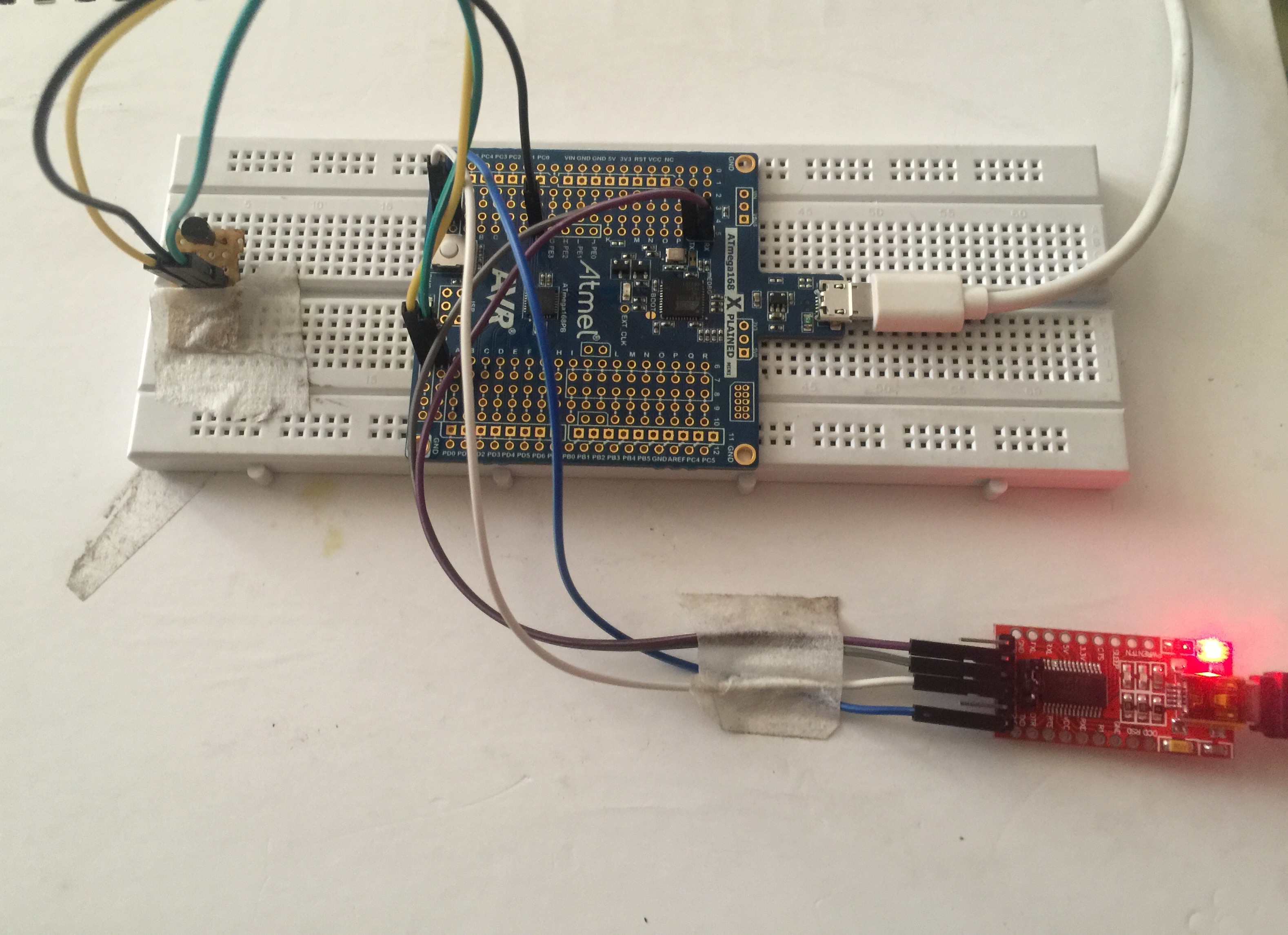
while(1){}

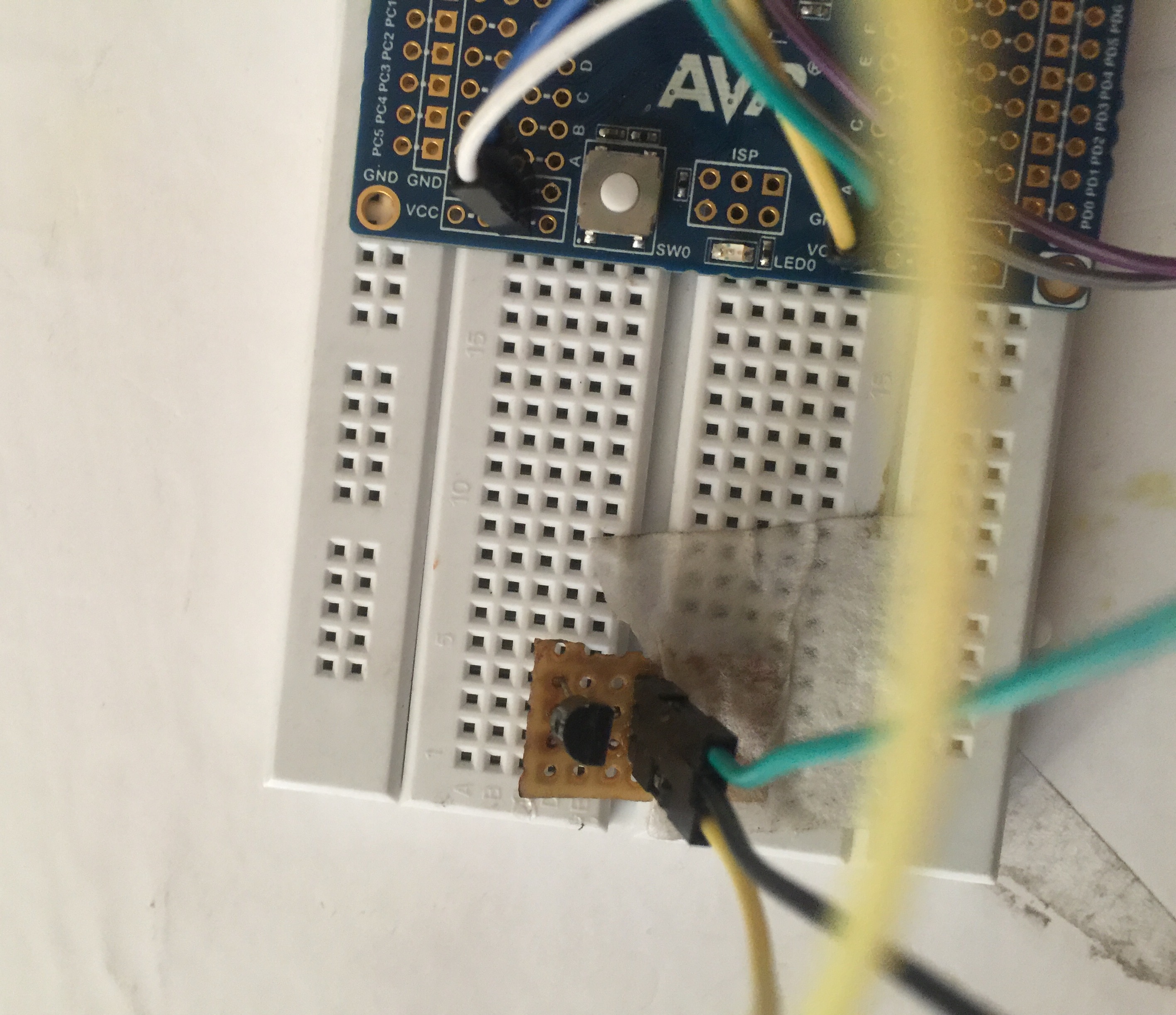
}

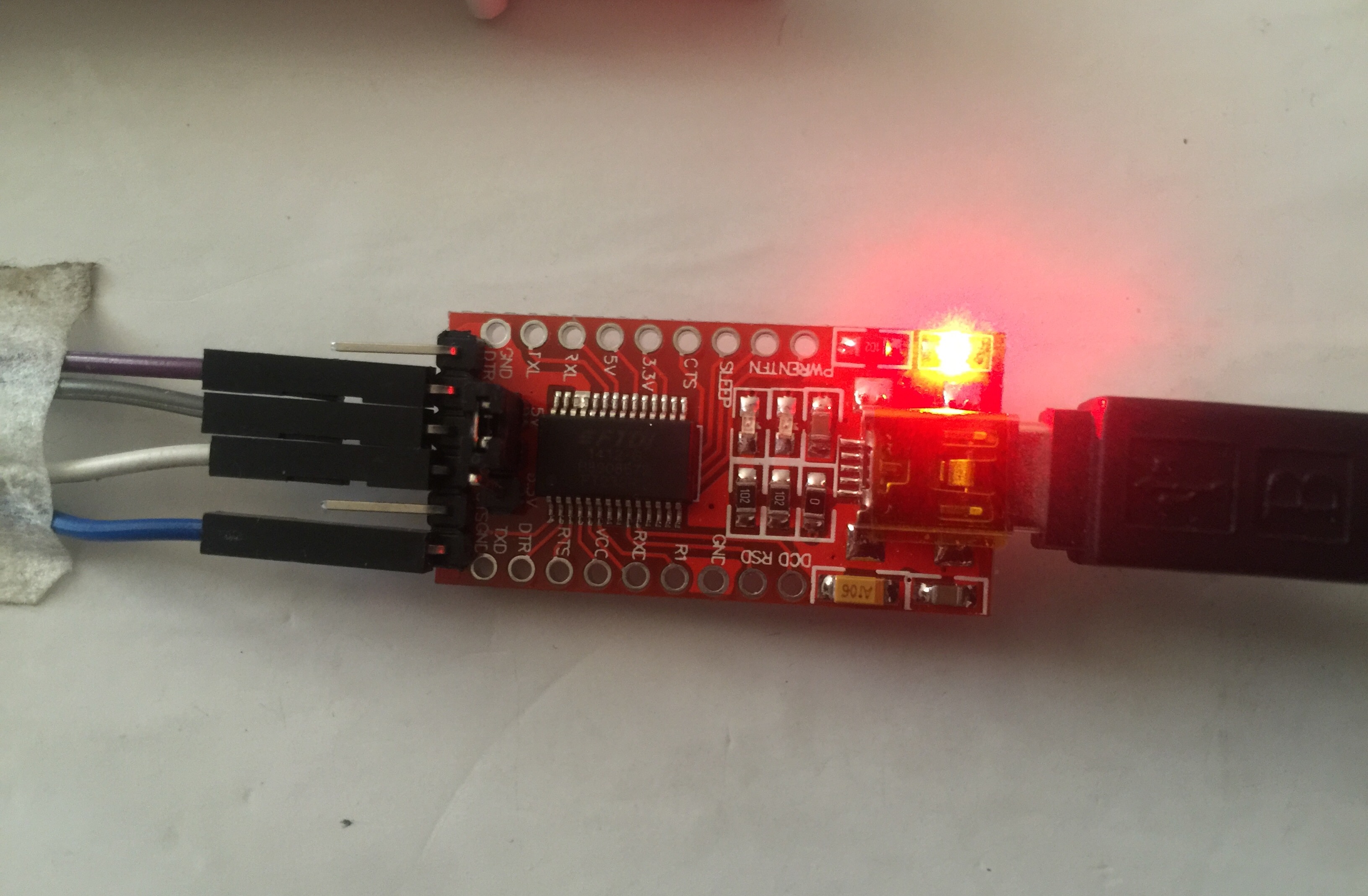
|  |  |  |  |
| --- | --- | --- | --- |
| 1. | SCHEMATIC |  |  |

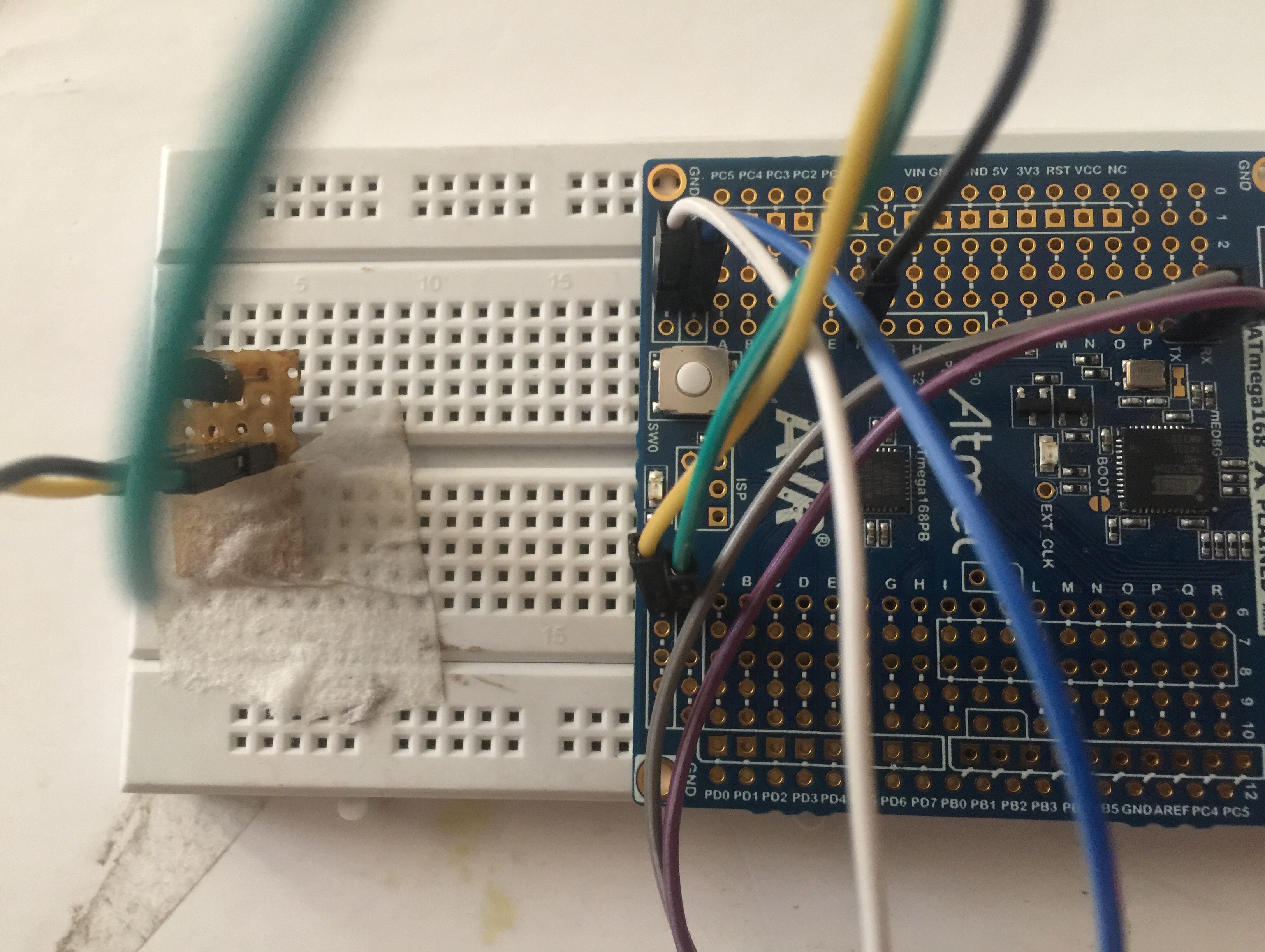


|  |  |  |  |
| --- | --- | --- | --- |
| 2. | SNAPSHOT OF THE BOARD WITH CONNECTED COMPONENTS |  |  |

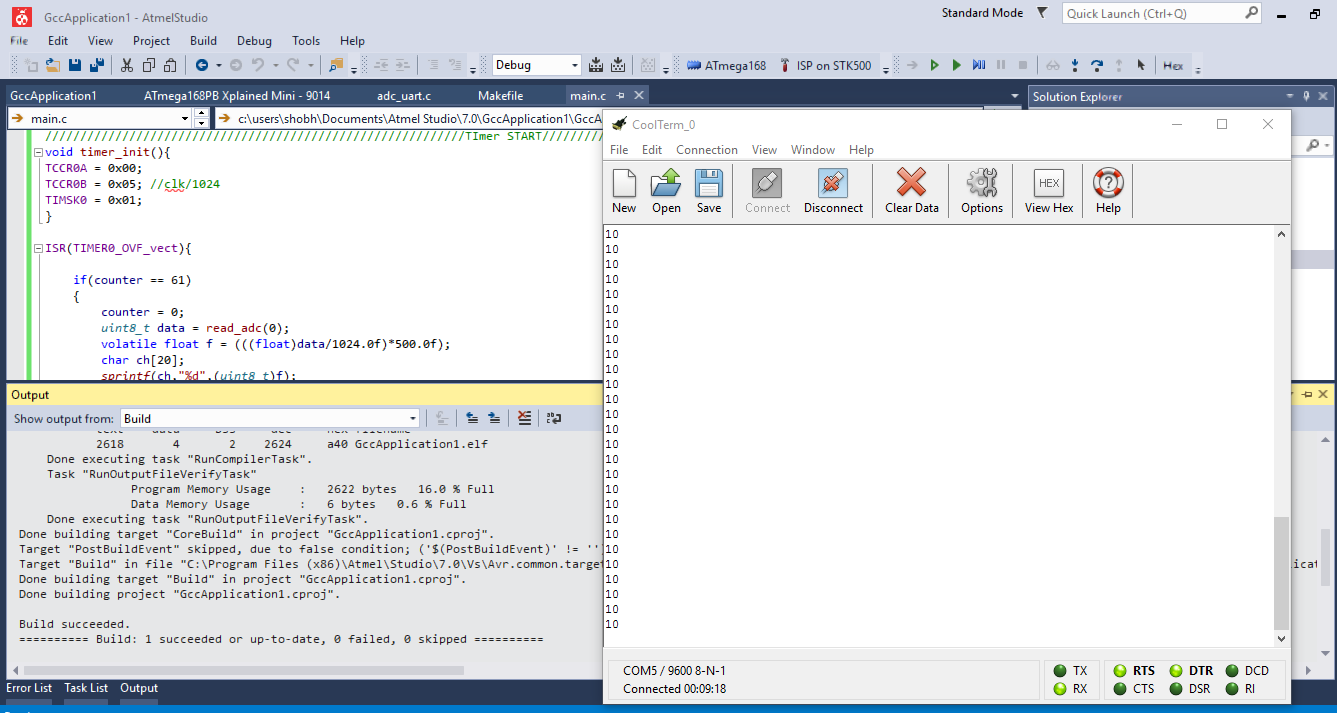








|  |  |  |  |
| --- | --- | --- | --- |
| 3. | SNAPSHOT OF THE COMPUTER SCREEN WITH TERMINAL |  |  |



**Student Academic Misconduct Policy**

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

Noor Blum