CPE301 – SPRING 2021

Design Assignment 3A

Student Name:	Elmer Mejia
Student #:	5003824808
Student Email:	mejiae4@unlv.nevada.edu
Primary Github address:	https://github.com/cpemejia/design_assignments.git
Directory:	design_assignments

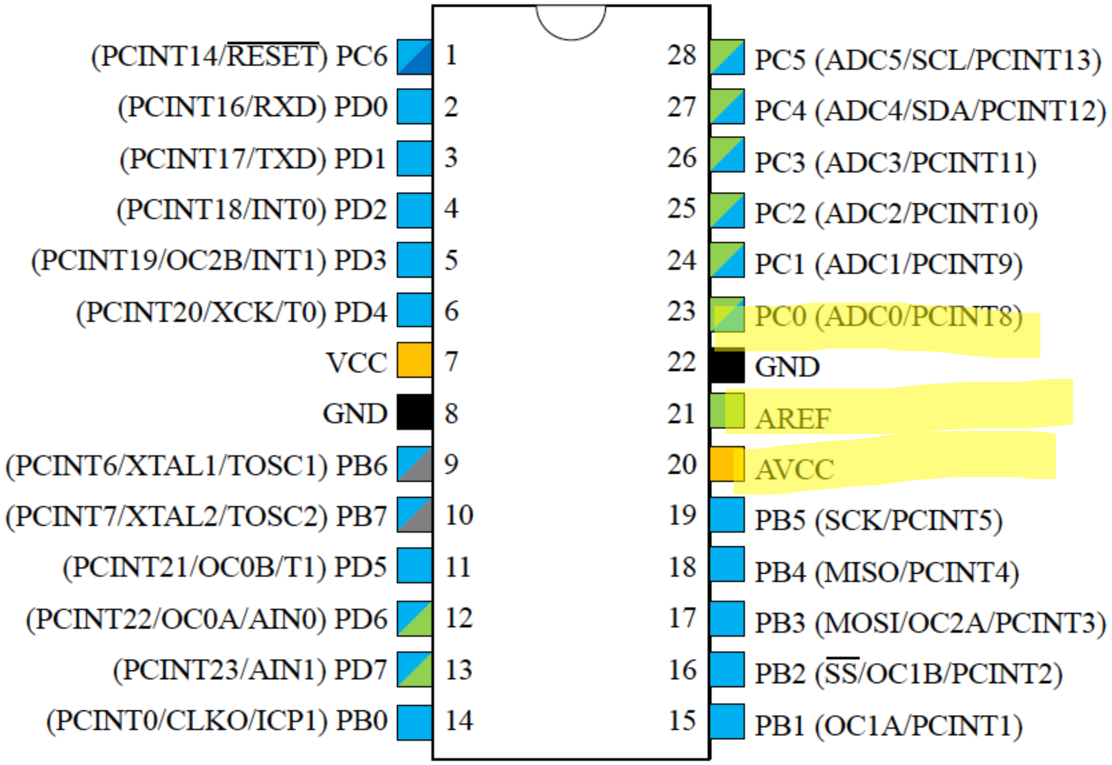
1. COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS

-Atmega328pb

-Microchip Studio

- Kicad

-LM35

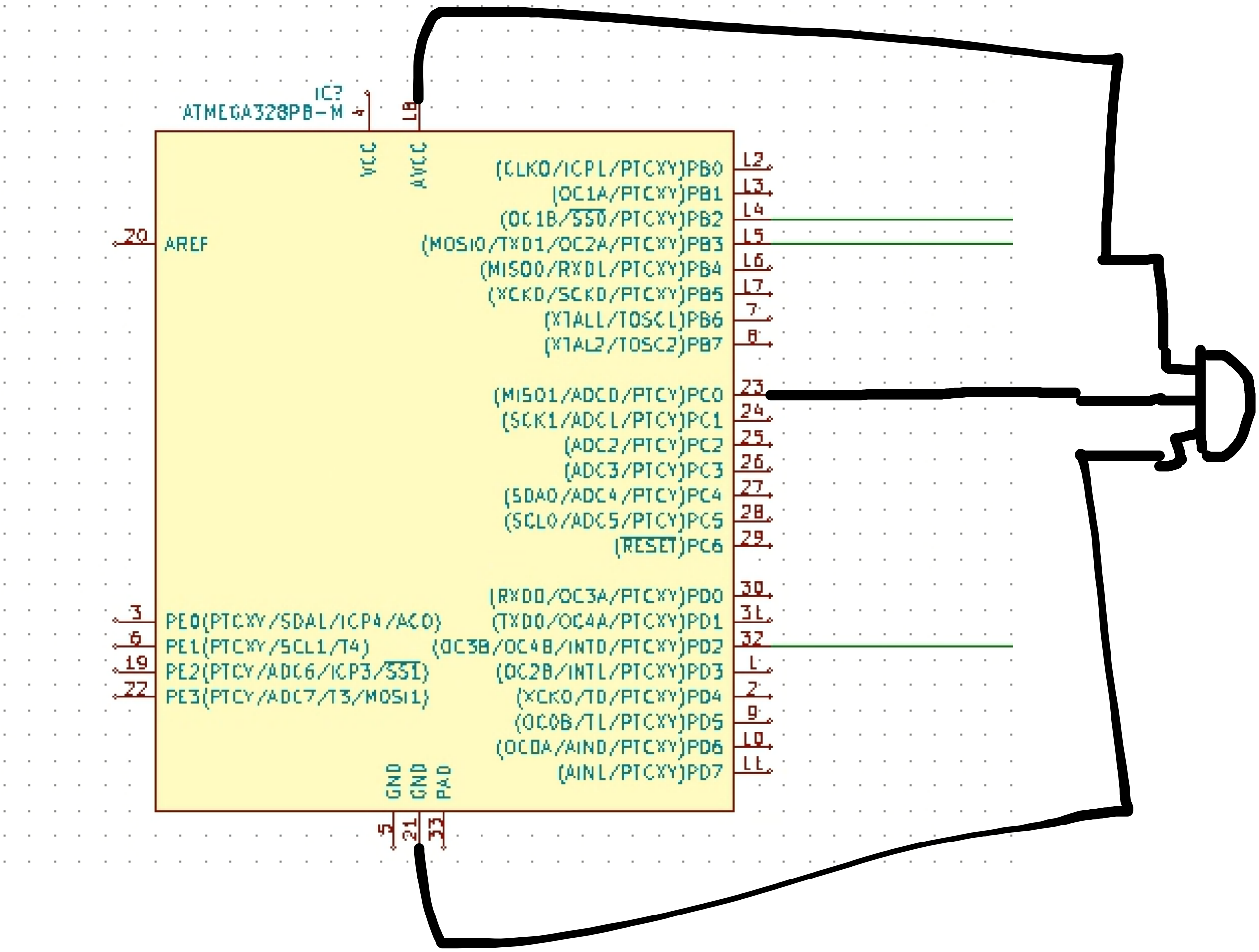


```
/*
 * 3AUART.c
 *
 * Created: 3/12/2021 9:42:49 PM
 * Author : ElmerOMejia
 */
#define F_CPU 16000000UL
#include <stdio.h>
#include <avr/io.h>
#include <avr/interrupt.h>
#define USART_BAUDRATE 9600
#define UBRR_VALUE (((F_CPU / (USART_BAUDRATE * 16UL))) -1)
volatile uint8_t counter = 0;

void USART0Init(void)
{
       // Set baud rate
       UBRR0H = (uint8_t)(UBRR_VALUE >> 8);
       UBRR0L = (uint8_t)UBRR_VALUE;
       // Set frame format to 8 data bits, no parity, 1 stop bit
       UCSR0C |= (1 << UCSZ01) | (1 << UCSZ00);
       //enable transmission and reception
       UCSR0B |= (1 << RXEN0) | (1 << TXEN0);
}

void InitTimer1(void)
{
       //Set Initial Timer value
       TCNT1 = 0;
       //TOP set to 6249 for 0.05s delay at compare match
       OCR1A = 6249;
       //Set CTC mode
       TCCR1B |= (1<<WGM12);
    TIMSK1 |= (1<<OCIE1A); // enable compare interrupt

}



int USART_send(char u8Data)
{
       //wait while previous byte is completed
       while (!(UCSR0A & (1 << UDRE0))) {};
              // Transmit data
              UDR0 = u8Data;
              return 0;
}

//start timer 1 with prescaller8
void StartTimer1(void)
{
       //Set prescaller 64, and start timer
       TCCR1B|= (1<<CS11)|(1<<CS10);

}
```

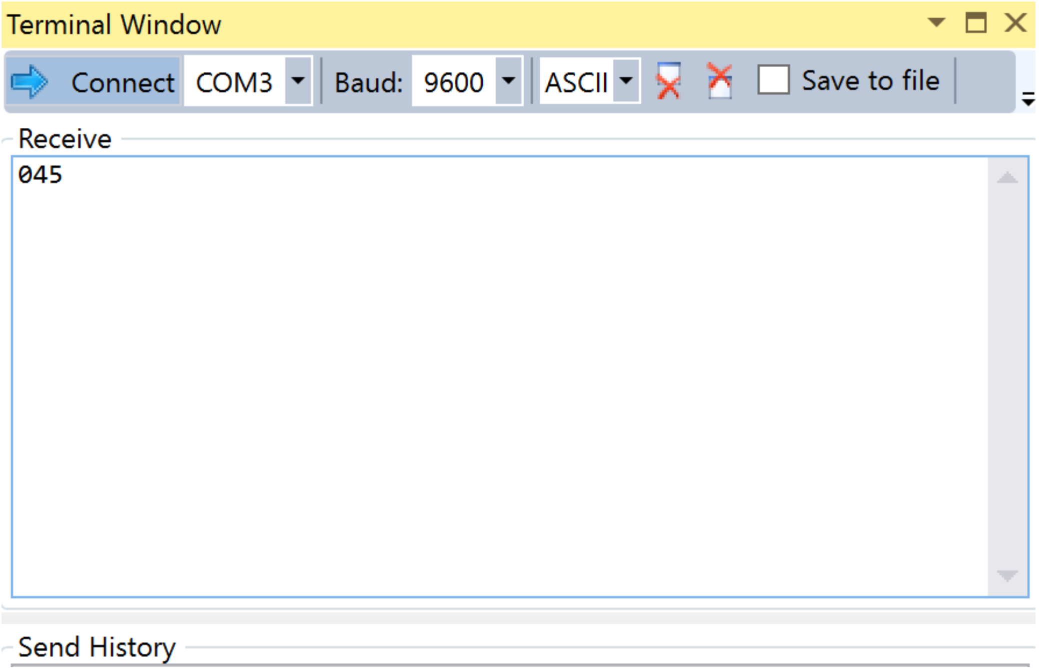
```
void InitADC()
{
       // Select Vref=Avcc
       ADMUX |= (1 << REFS0);
       //set prescaler to 32, enable ADC interrupt, and enable ADC
       ADCSRA |= (1 << ADPS2) | (1 << ADPS0) | (1 << ADIE) | (1 << ADEN);
}

void SetADCChannel(uint8_t ADCchannel)
{
       //select ADC channel with safety mask
       ADMUX = (ADMUX & 0xF0) | (ADCchannel& 0x0F);
}

void StartADC(void)
{
       ADCSRA |= (1 << ADSC); // set ADSC to begin ADC
}


ISR (TIMER1_COMPA_vect)
{
       counter++;
       if (counter == 5)
       {
              // start conversion
              StartADC();
       }
}

//ADC conversion complete ISR
ISR(ADC_vect)
{
       counter = 0;                // set count back to 0
       int a = ADCL;              // get adc
       a = a | (ADCH<<8);
       a = (a/1024.0) * 5000/10;  // begin converting adc value and send to usart
       a = (a * (9/5)) + 32;      // celsius to fahrenheit
    USART_send((a/100)+'0');
       a = a % 100;
       USART_send((a/10)+'0');
       a = a % 10;
       USART_send((a)+'0');
       USART_send('\r');
}

void InitPort()
{
       DDRC |= (1<<0);     // set PC0 as output
       PORTC &= ~((1<<0));
}
```

```
int main()
{
	//Initialize USART0
	USART0Init();
	//initialize ports
	InitPort();
	//initialize ADC
	InitADC();
	// ADC channel is ADC0
	SetADCChannel(0);
	//initialize timer1
	InitTimer1();
	//start timer1
	StartTimer1();
	//enable global interrupts
	sei();

    while (1)
	{

	}
	return 0;
}
```

2. SCHEMATICS



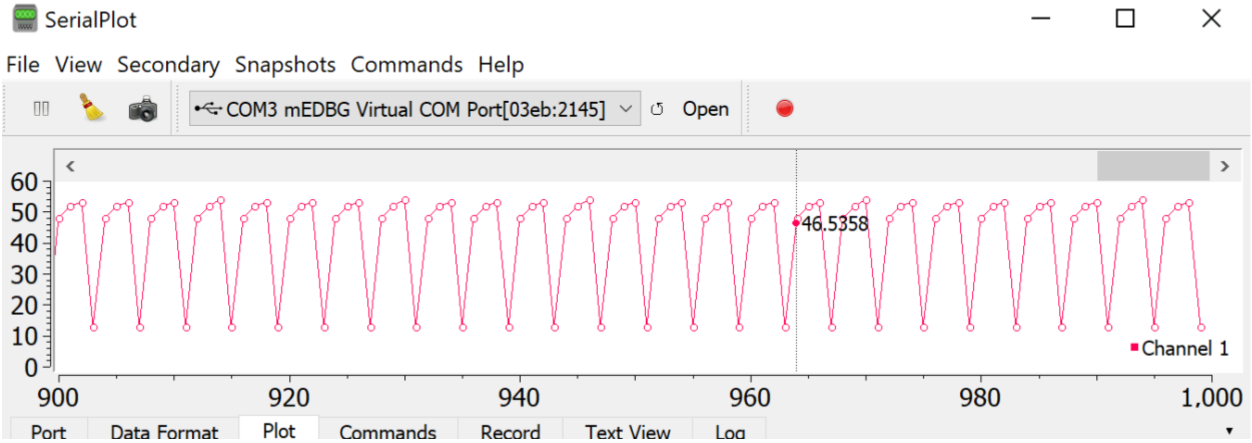
3. SCREENSHOTS OF EACH TASK OUTPUT (ATMEL STUDIO OUTPUT)



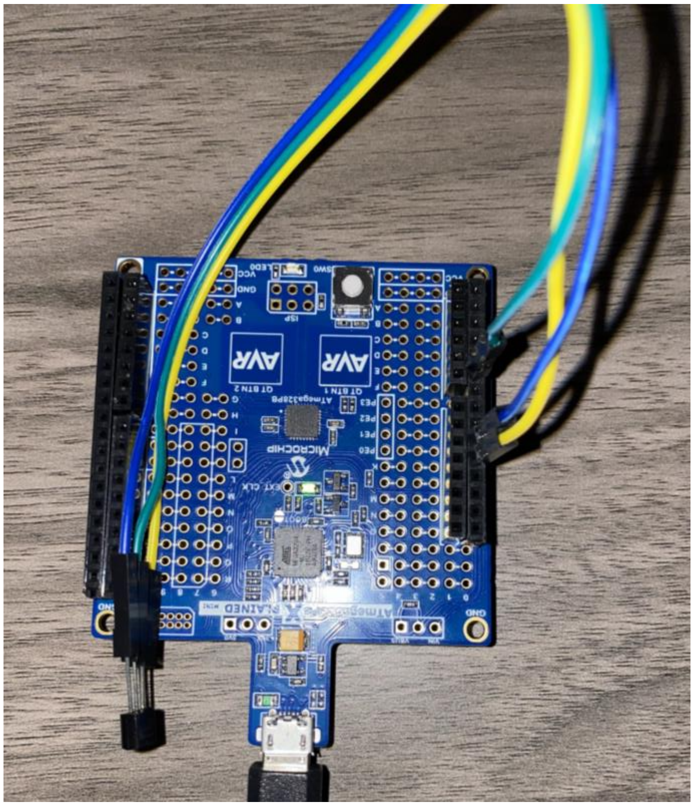
COM3 Baud: 9600 ASCII □ Save to file

Receive

```
045
```



4. SCREENSHOT OF EACH DEMO (BOARD SETUP)



5. VIDEO LINKS OF EACH DEMO

Design_Assignments - YouTube

6. GITHUB LINK OF THIS DA

DA_3A

Student Academic Misconduct Policy
http://studentconduct.unlv.edu/misconduct/policy.html

"This assignment submission is my own, original work".
Elmer Mejia