**CPE301 – SPRING 2019**

# Design Assignment 4A

Student Name: Itzel Becerril
Student #: 2000478001
Student Email: becerri2@unlv.nevada.edu
Primary Github address: hadidbuilds
Directory: DA4A

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

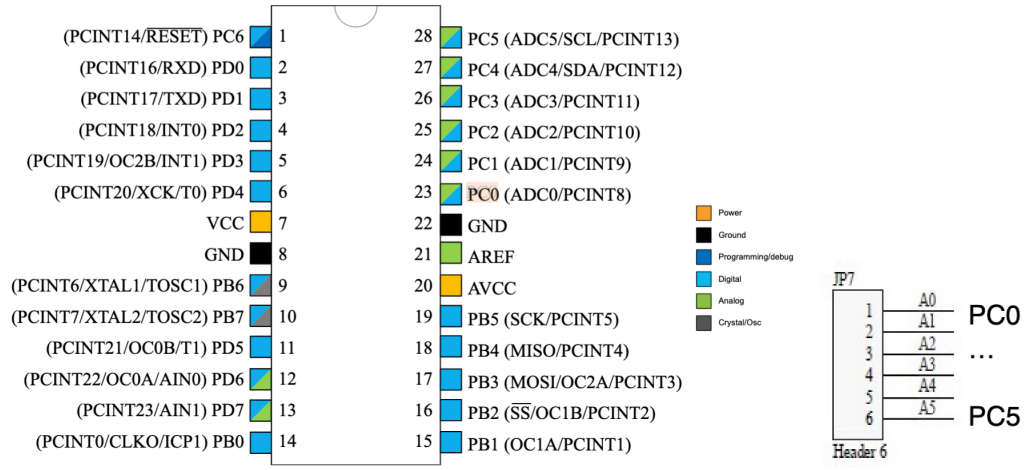
List of Components used

- Atmega 328P
- Xplained mini
- Microusb board
- Breadboard
- Male/female wires
- Driver
- Potentiometer
- Multifunctional shield
- Dc motor

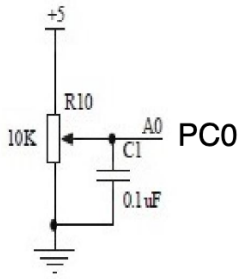
Block diagram with pins used in the Atmega328P

**Pin-out**

**Figure 5-1. 28-pin PDIP**



For assignments in PWM use ~5/PD5 (T0B), ~6/PD6 (T0A), ~

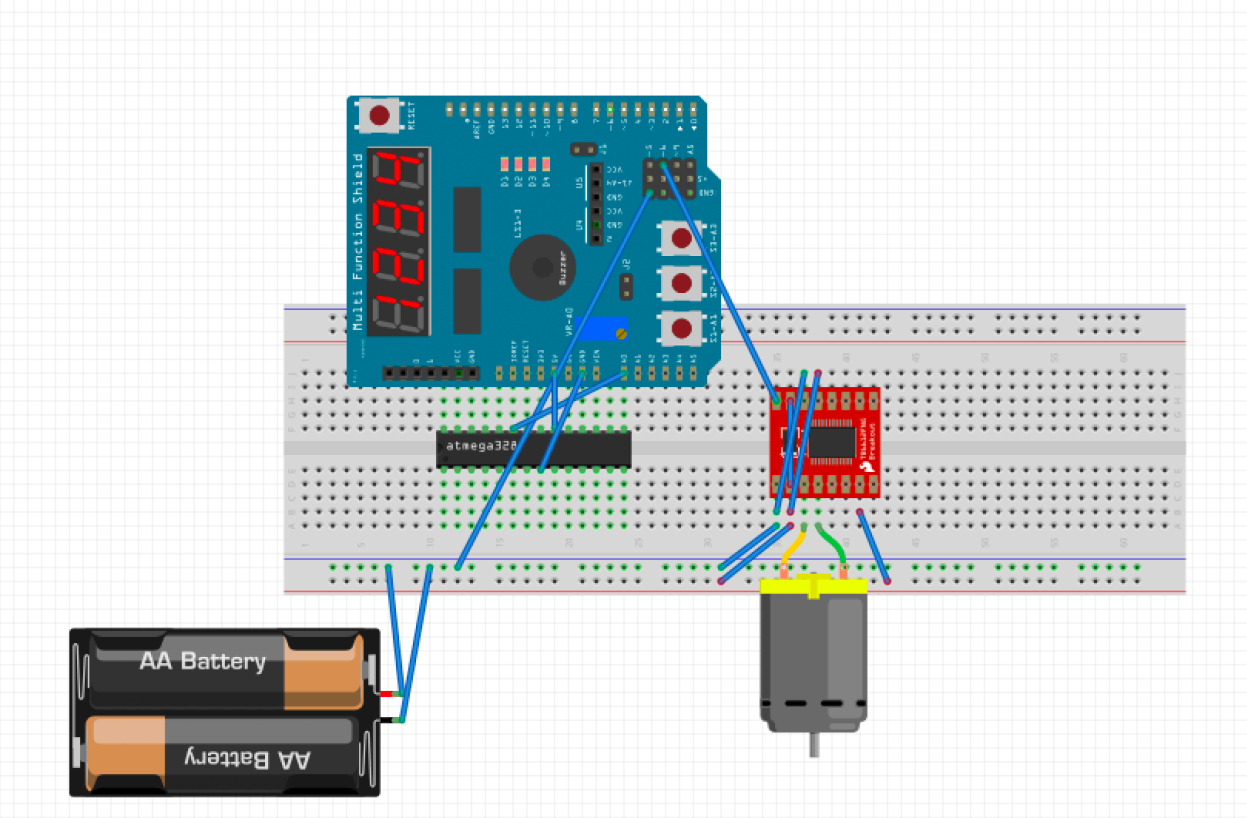



## 2. DEVELOPED C CODE

```
#define F_CPU 16000000UL // clock is 16MHz
#include <avr/io.h>
#include <avr/interrupt.h>
#include <util/delay.h>

void init_adc(void);
int potencon; // controlling the motor with potentiometer
int toggle = 0; // to toggle the motor with the switch

int main()
{
	DDRD = 0x40; //enable port D
	DDRC = 0x02; // set Port C as outputs
	PORTC |= (1<<1); // enable pull-up pin
	TCCR0A=0x83; // set fast PWM & clear OCR0A on MATCH
	TCCR0B=0x05; // set prescaler to 1024
	PCICR = 0x02; // 0x02 is PCIE1, that is, enable PCIE1 for PCMSK1 to work
	PCMSK1 = 0x02; // enable pin changes on PCINT8 (PC0)

	sei(); // enable interrupt
	init_adc(); //call init_adc

	while (1)
	{


	}
}
void init_adc(void) // Initiate ADC function
{
	ADMUX = (1<<REFS0); // Reference voltage at Aref
	ADCSRA = (1<<ADEN)|(1<<ADSC)|(1<<ADATE)|(1<<ADPS2)|(1<<ADPS1)|(1<<ADPS0);
	// from ADCRSA we Enable ADC, Start Conversion, Set prescalar as 128
}

ISR(PCINT1_vect)
{
	if(!(PINC & (1<<PINC1))) // if button is pressed
	{
		if(toggle == 0) // if 0
		{
```

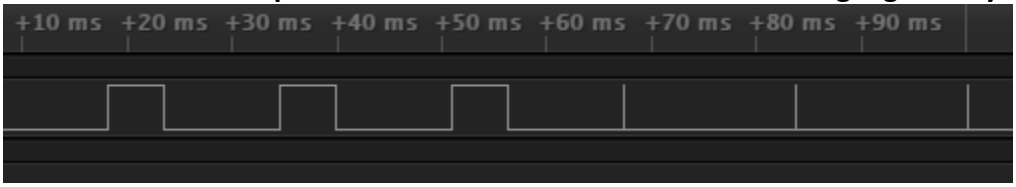
```
                OCR0A = 0; // reset
                _delay_ms(1000);
            }
            if (toggle == 1) // if 1
            {
                while((ADCSRA&(1<<ADIF))==0); // wait for conversion

                potencon = ADC/5; // ADC Conversion
                OCR0A = potencon; // Output to converted value to 0CR0A
                _delay_ms(1000);
            }
            toggle ^= 1; //update state of motor to on
        }
}
```

## 3. SCHEMATICS



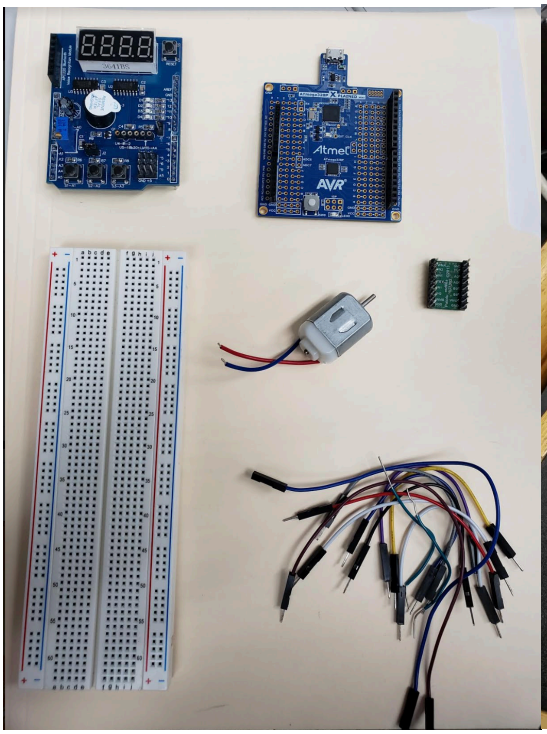

## 4. SCREENSHOTS OF EACH TASK OUTPUT (ATMEL STUDIO OUTPUT)

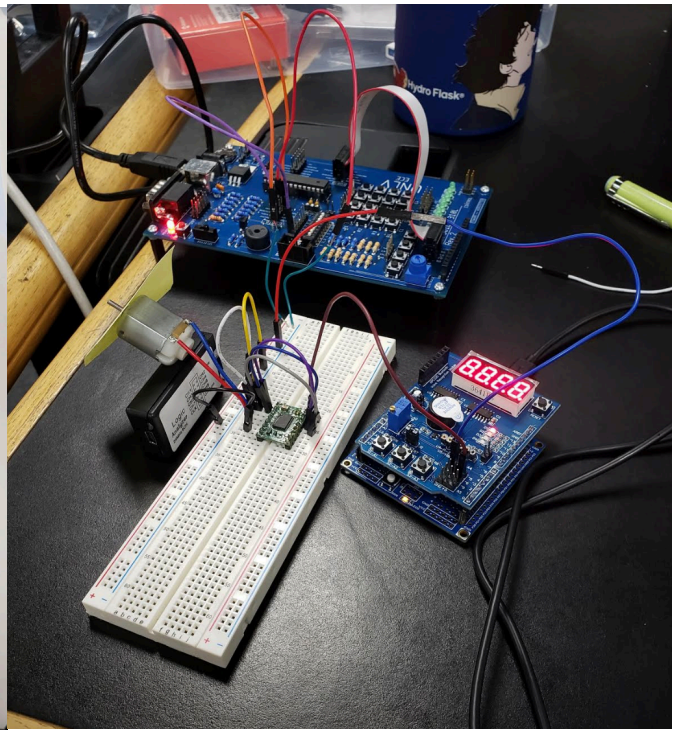
**No Atmel Studio output but this is the wave form of PWM using logic analyzer.**



## 5. SCREENSHOT OF EACH DEMO (BOARD SETUP)

**Before:** **After:**




## 6. VIDEO LINKS OF EACH DEMO

https://www.youtube.com/watch?v=8K3a-kCnGnk

## 7. GITHUB LINK OF THIS DA

https://github.com/HadidBuilds/hw_sub_da1

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

"*This assignment submission is my own, original work*".
Itzel Becerril