

Design Assignment 4B

Student Name: Francisco Mata Carlos

Student #: 1012593607

Student Email: matacarl@unlv.nevada.edu

Primary Github address: https://github.com/chicosisco/da_sub.git

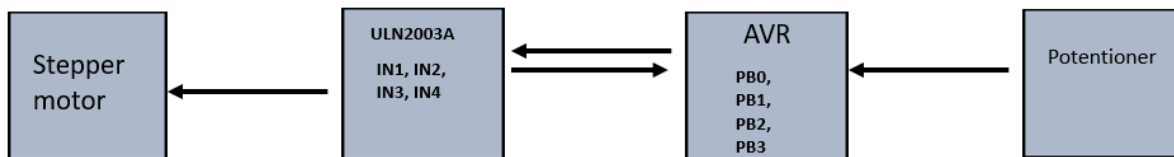
Directory: repository/cpe301/DesignAssignments/DA4B

1. COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS

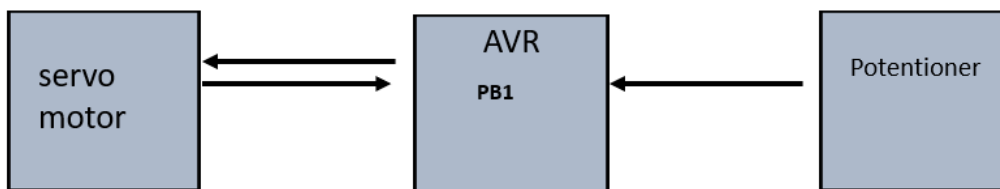
The components used for this assignment are the next:

- Atmega328p Xplained Mini
- Multi-functional Shield
- Atmel Studio 7
- Sepper Motor(Bi-polar)
- Servo Motor
- Breadboard

Block diagram with pins used in the Atmega328P and stepper motor



Block diagram with pins used in the Atmega328P and servo motor



2. INITIAL/MODIFIED/DEVELOPED CODE OF TASK 1/A

1. Task 1

Write an AVR C program to control the speed of the Stepper Motor using a potentiometer connected to PC0. Use a timer in CTC mode to control the delay.

```
#define F_CPU 16000000UL
#include <avr/io.h>
#include <util/delay.h>

// Function Prototypes
void initial(void); // function setting Timer/Counter control Register 0
void adc_init(void); // function declared for ADC set up

int main(void)
{
    int period; // integer used to set the period

    while (1)
    {
        initial(); // calling function to initialize the setting Timer/Counter
        control Register 0
        adc_init(); // calling function for ADC set up
        DDRB=0x0F;
        OCR0A = ADC;
        // setting the period to a value of 2, fast frequency
        if ((OCR0A>=0)&&(OCR0A<=20))
        {
            period =2 ;
            PORTB = 0x09;
            _delay_ms(period);
            PORTB = 0x03;
            _delay_ms(period);
            PORTB = 0x06;
            _delay_ms(period);
            PORTB = 0x0c;
            _delay_ms(period);
        }
        // setting the period to a value of 10
        else if((OCR0A>20)&&(OCR0A<=40))
        {
            period=10;
            PORTB = 0x09;
            _delay_ms(period);
            PORTB = 0x03;
            _delay_ms(period);
            PORTB = 0x06;
            _delay_ms(period);
            PORTB = 0x0c;
            _delay_ms(period);
        }
        // setting the period to a value of 20
        else if((OCR0A>40)&&(OCR0A<=80))
        {
            period=20;
```

```

        PORTB = 0x09;
        _delay_ms(period);
        PORTB = 0x03;
        _delay_ms(period);
        PORTB = 0x06;
        _delay_ms(period);
        PORTB = 0x0c;
        _delay_ms(period);
    }
    // setting the period to a value of 30
    else if((OCR0A>80)&&(OCR0A<=120))
    {
        period=30;
        PORTB = 0x09;
        _delay_ms(period);
        PORTB = 0x03;
        _delay_ms(period);
        PORTB = 0x06;
        _delay_ms(period);
        PORTB = 0x0c;
        _delay_ms(period);
    }

    else if((OCR0A>120)&&(OCR0A<=150))
    {
        period=40;
        PORTB = 0x09;
        _delay_ms(period);
        PORTB = 0x03;
        _delay_ms(period);
        PORTB = 0x06;
        _delay_ms(period);
        PORTB = 0x0c;
        _delay_ms(period);
    }
    // setting the period to a value of 50, fast frequency
    else if((OCR0A>150)&&(OCR0A<=225))
    {
        period=50;
        PORTB = 0x09;
        _delay_ms(period);
        PORTB = 0x03;
        _delay_ms(period);
        PORTB = 0x06;
        _delay_ms(period);
        PORTB = 0x0c;
        _delay_ms(period);
    }

    else if(OCR0A>225)
    {
        PORTB=0;
    }
}

}

void initial(void)
{

```

```

//clear on compare for OCOA
//CTC mode
//pre-scaler = 256
TCCR0A |= (1<<COM0A1)|(0<<COM0A0)|(0<<COM0B1)|(0<<COM0B0)|(1<<WGM01)|(0<<WGM00);
TCCR0B |= (1<<CS02)|(0<<CS01)|(0<<CS00);

}

// function to initialize ADC set up
void adc_init(void)
{
    ADCSRA |= ((1<<ADEN)|(1<<ADSC)|(0<<ADPS2)|(1<<ADPS1)|(1<<ADPS0));
    ADMUX |= (1<<REFS0);
}

```

2. Task 2

Write an AVR C program to control the position of the Servo Motor using a potentiometer connected to PC0. When pot value is 0 the servo is at position 0 deg. and when pot value is max (approx. 5V) the servo is at position 180 deg

```

#define F_CPU 16000000UL
#include <avr/io.h>
#include <util/delay.h>

// Function Prototypes
void initial(void); // function setting Timer/Counter control Register 0
void adc_init(void); // function declared for ADC set up

int main(void)
{
    while (1)
    {
        initial(); // calling function to initialize the setting Timer/Counter
        control Register 0
        adc_init(); // calling function for ADC set up
        ICR1=4999;
        DDRB|=(1<<PB1);
        OCR1A=ADC; // reading from potentiometer to control servo angle
        _delay_ms(10);
    }
}

// Clear OC1A/OC1B on compare match (set output to low level)
// Mode = 14, Fast PWM
//pre-scaler = 64
void initial(void)
{
    TCCR1A |= (1<<COM1A1)|(1<<COM1B1)|(1<<WGM11);
    TCCR1B |= (1<<WGM13)|(1<<WGM12)|(1<<CS11)|(1<<CS10);
}

```

```
// function to initialize ADC set up
void adc_init(void)
{
    ADCSRA |= ((1<<ADEN)|(1<<ADSC)|(0<<ADPS2)|(1<<ADPS1)|(1<<ADPS0));
    ADMUX |= (1<<REFS0);
}

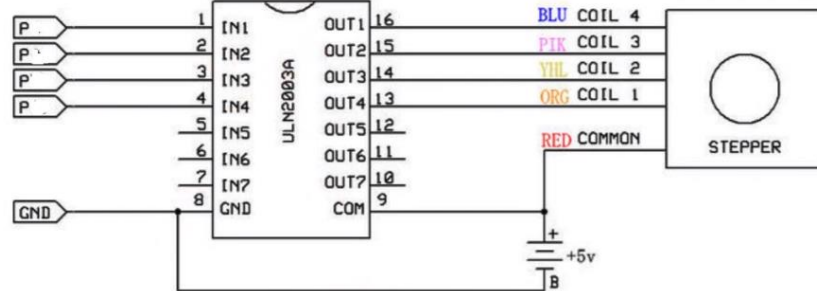
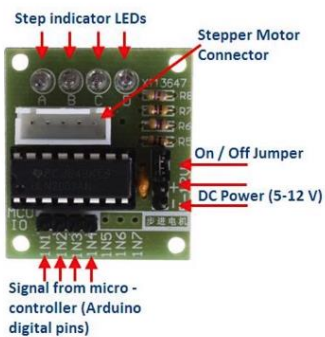
```

3. DEVELOPED MODIFIED CODE OF TASK 2/A from TASK 1/A

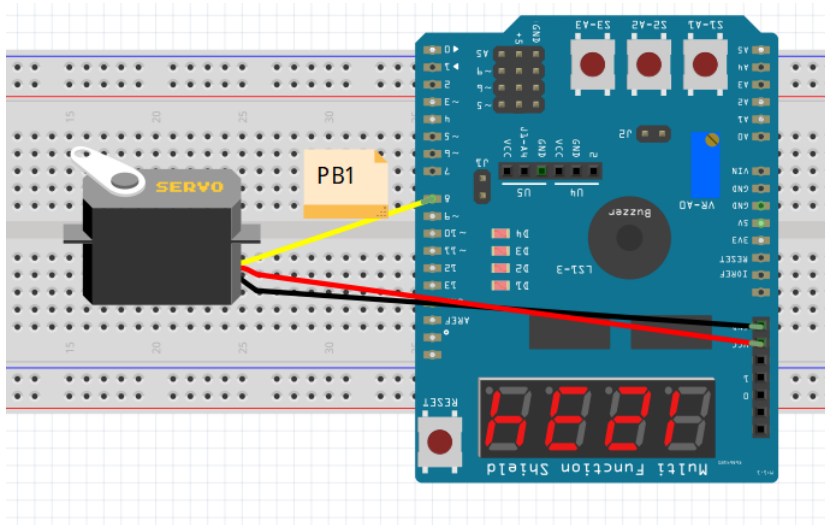
Same as above

4. SCHEMATICS

Task 1



Task 2



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

TASK 1

```

#define F_CPU 16000000UL
#include <avr/io.h>
#include <util/delay.h>

// Function Prototypes
void initial(void); // function setting Timer/Counter control Register 0
void adc_init(void); // function declared for ADC set up

int main(void)
{
    int period; // integer used to set the period

    while (1)
    {
        initial(); // calling function to initialize the setting Timer/Counter control Register 0
        adc_init(); // calling function for ADC set up
        DDRCB=0x0F;
        OCR0A = ADC;
        // setting the period to a value of 2, fast frequency
        if ((OCR0A>0)&&(OCR0A<20))
        {
            period =2;
            PORTB = 0x09;
            _delay_ms(period);
            PORTB = 0x03;
            _delay_ms(period);
            PORTB = 0x06;
            _delay_ms(period);
            PORTB = 0x0c;
            _delay_ms(period);
        }
        // setting the period to a value of 10
        else if((OCR0A>20)&&(OCR0A<40))
        {
            period=10;
            PORTB = 0x09;
            _delay_ms(period);
            PORTB = 0x03;
            _delay_ms(period);
        }
    }
}

```

[illegible]

Task 2

```
#define F_CPU 16000000UL
#include <avr/io.h>
#include <util/delay.h>

// Function Prototypes
void initial(void); // function setting Timer/Counter control Register 0
void adc_init(void); // function declared for ADC set up

int main(void)
{
    while (1)
    {
        initial(); // calling function to initialize the setting Timer/Counter control Register 0
        adc_init(); // calling function for ADC set up
        ICR1=4999;
        DDRA=(1<<PB1);
        OCR1A=ADC; // reading from potentiometer to control servo angle
        _delay_ms(10);
    }

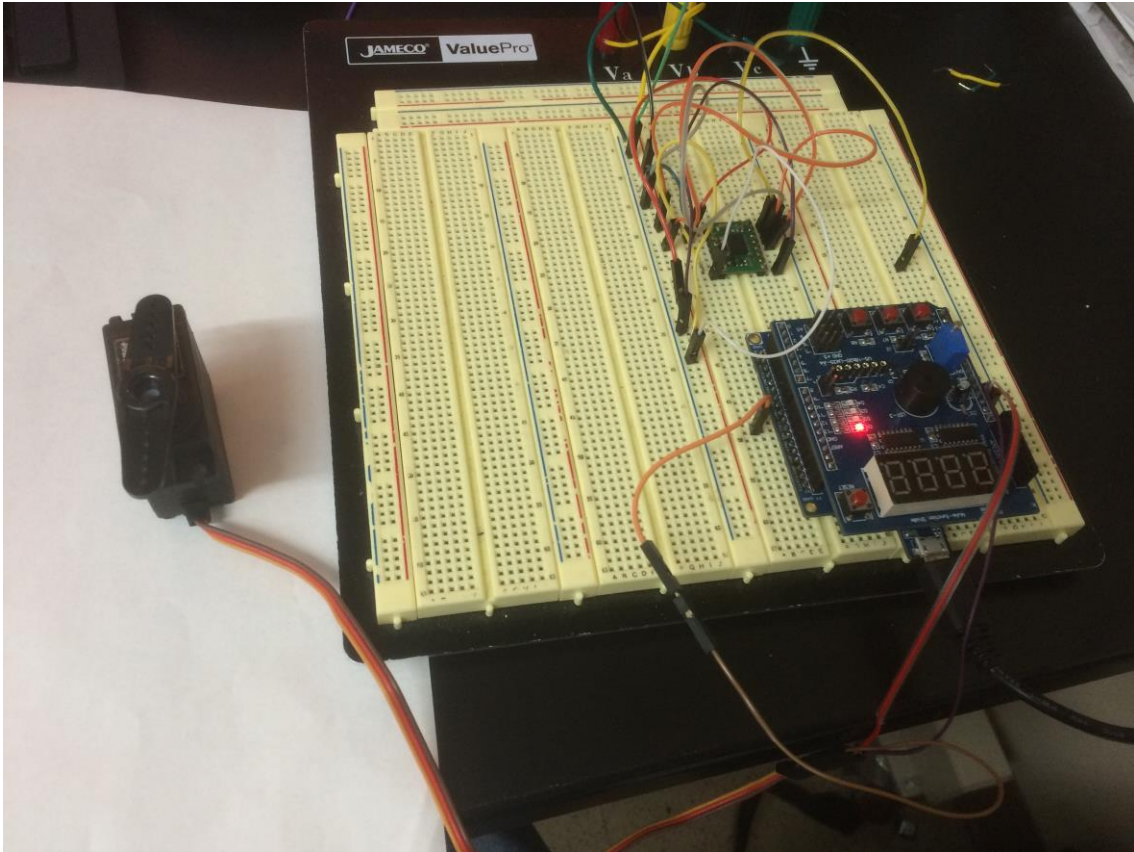
    // Clear OC1A/OC1B on compare match (set output to low lever1)
    // Mode = 14, Fast PWM
    //pre-scaler = 64

    void initial(void)
    {
        TCCR1A |= (1<<COM1A1)|(1<<COM1B1)|(1<<WGM11);
        TCCR1B |= (1<<WGM13)|(1<<WGM12)|(1<<CS11)|(1<<CS10);
    }

    // function to initialize ADC set up
    void adc_init(void)
    {
        ADSCRA |= ((1<<ADEN)|(1<<ADSC)|(0<<ADPS2)|(1<<ADPS1)|(1<<ADPS0));
        ADMUX |= (1<<REFS0);
    }
}
```

[illegible]

6. SCREENSHOT OF EACH DEMO (BOARD SETUP)



7. VIDEO LINKS OF EACH DEMO

Task 1 video:

<https://youtu.be/sqZjKxtxrRg>

Task 2 video:

<https://youtu.be/0HaUv2sjbuU>

8. GITHUB LINK OF THIS DA

https://github.com/chicosisco/da_sub.git

Student Academic Misconduct Policy

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

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

Francisco Mata Carlos