Lab 8

Servo Motor: Angular Position Control

Florida Polytechnic University

EEL 4746C / EEL 5756C: Microcomputer

Fall 2018

Student Name:

Student ID:

Section:

Contents

1 Introduction

In this lab students will gain hands-on experience controlling the angle of a servo motor using Pulse Width Modulation (PWM).

2 Background

2.1 Duty Cycle

For a periodic control signal, the duty cycle is the percent of time that your signal is active with respect to the period of the signal. The duty cycle for a signal is given as:

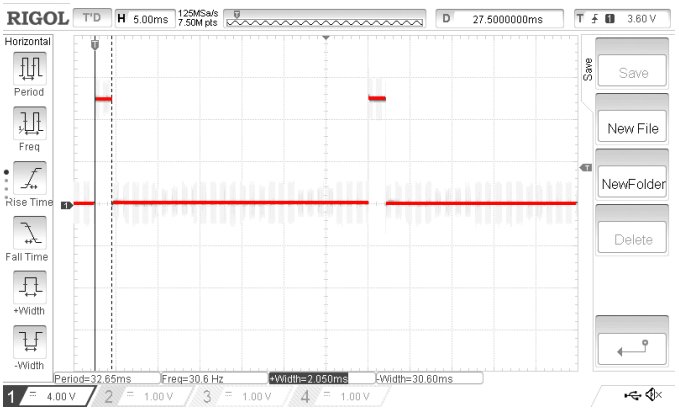
$$D = \frac{T_{high}}{T_{high} + T_{low}} \times 100\% = \frac{T_{high}}{T} \times 100\%$$

For example, for the signal shown in figure 1a, the duty cycle is calculated as:

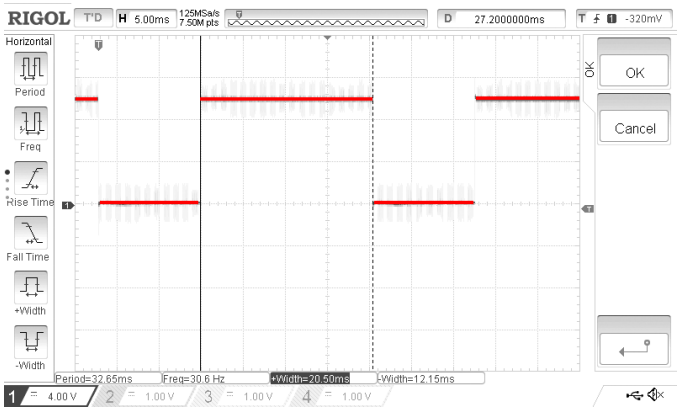
$$D = \frac{T_{high}}{T} \times 100\% = \frac{2.05 \times 10^{-3}}{32.85 \times 10^{-3}} \times 100\% = 6.24\%$$

Another example, the signal shown in figure 1b, the duty cycle for this signal is calculated as:

$$D = \frac{T_{high}}{T} \times 100\% = \frac{20.5 \times 10^{-3}}{32.85 \times 10^{-3}} \times 100\% = 62.4\%$$



(a) Period 32.85ms and Pulse width=2.05ms



(b) Period 32.85ms and Pulse width=20.50ms

Figure 1: PWM with different pulse widths

2.2 Pulse Width Modulation (PWM)

Pulse Width Modulation (PWM) is an encoding technique for digital signals where the information is represented using the pulse widths. PWM is used in multiple applications:

- Control of DC Servo Motors:
  - Position control
  - Speed / direction control of continuous rotation servos
- Brightness control

- Heat control
- Audio systems
- Power

One way to think of PWM is as a way to control analog devices with digital signals. The effect of turning the power on an off on the analog device will differ depending on the physical nature of that device.

2.3 DC Motors

A DC motor is an electrical device that converts the electrical energy of a DC voltage to rotational mechanical energy. One way to categorize DC motors is based on the field:

- Permanent Magnet
- Series
- Shunt
- Compound (Mix of the two above)

The regular DC motor has two wires to be supplied with power. By applying the PWM signal to the regular DC motor, the pulse will determine the speed of rotation for the motor. To change the direction of rotation in the DC motor, the applied voltage needs to reverse polarity, i.e., change the direction of the current.

Question: What about the direction?

2.4 Servo Motor

A Servo motor is a DC motor with three extra components for control:

- Gearing set.
- Control circuit.
- Position sensor (example: Potentio-meter).

Unlike the DC motor, the Servo has three wires, two for power and one for control. When a pulse is received on the control wire, the control circuit will apply power to the DC motor, until the position sensor detects the required position.

Note 1: A Servo motor requires a PWM signal width fixed period to control. Most of recent, Radio Controlled (RC) Servo motors operates on 20ms period (or 50Hz frequency of pulses) with the pulse widths from almost 0.5ms and up to almost 2ms. Most servo motors have a limit on the angle between 120° and 270°. Some of the new RC Servo motors don't depend on the length of the period, i.e., duty cycle, but only on the width of the pulse.

Note 2: When a servo motor is modified and the mechanical tab (break) is broken from the gear, the DC motor is physically allowed to turn beyond the original angle limit. With another modification by not allowing the position sensor to track the position and just have a fixed position, the Servo motor is transferred to a different kind of Servo motor, that is known as the continuous rotation Servo.

Question: What will happen when the position sensor stops tracking the position? What about the direction of rotation?

2.5 Stepper Motor

A stepper motor can be thought of as an enhanced servo motor. Instead of the DC motor and the controller, the motor is designed to divide the full rotation of the motor into a certain number of equal steps. The steps are defined by electomagnets. The stepper motor is using open-loop control instead of the closed loop feedback control of the servo motor.

2.6 ATMega328p Timer/Counter 0

The TC0 in ATMega328p is an 8-bit timer/counter that we will use in this experiment. TC0 has four modes of operations. Two of these modes are for PWM. The first one is the Fast PWM and the second one is the Phase Correct PWM. In this experiment you are going to use the Phase Correct PWM. Refer to the datasheet for TC0 modes of operation.

3 Components used in the lab

3.1 Arduino UNO

3.2 Standard RC Servo Motor

Figure 2: Standard RC Servo-motor

[Source: https://github.com/SeeedDocument/Grove-Servo/raw/master/img/Grove%E2%80%94Servo.jpg]

3.3 Resistors

3.4 Jumper Wires

4 Experiment

4.1 Part I: finding the the approximate pulse widths for the 0° and the 180° angles

Use listing 1 to determine the number pulse widths for both the 0° and the 180° angles. If your motor has a limit of 120° max, then find the pulse width for the 120°. Modify the pulse width by modifying the value of `OCR0A`. Also, register the values of `OCR0A` that sets the motor angle to 0° and to 180°. We will refer to these values as L and H respectively.

4.2 Part 2: Control the angle of the servo

Create three switches using the jumper wires and the resistors. Let your MCU reads the value of the three switches as a 3-bit number, i.e., each switch is a bit in the number. Write a code to read the number n from the switches

and based on the read value (n), the MCU must control the angle of the motor by setting the value of $OCR0A$ using the following formula:

$$OCR0A = L + (H - L) \times \lfloor \frac{n}{7} \rfloor$$

4.3 Part 3: Control in C

Rewrite the code of (`Part 2`) in C.

5 Deliverables

For each part, the following are the deliverables:

- Circuit
- Code
- Recorded values / Observations
- Short Conclusion

6 Listings

Listing 1: Correct Phase PWM

```
                .include "eel4746Cv3.inc"
                initstack
init:
                sbi DDRD,6              ; Set pin 6 of Port D to output (OC0A)
                sbi DDRB,0              ; Set pin 0 of Port B to output
                ldi r16,127             ; Select Compare Value
                out OCR0A,r16           ; Set Output Compare Register A
                ldi r16,0x81            ; Select PWM Correct Phase mode
                out TCCR0A,r16          ; Set PWM Correct Phase mode
                ldi r16,0x03
                out TCCR0B,r16          ; Set Clock Prescale
                ldi r17,0
loop:
                cpi r17,2
                brsh skip
                sbi PINB,0              ; Toggle the value of bit 0
skip:
                call poll
                inc r17
                cpi r17,20
                brne loop
                ldi r17,0
                rjmp loop
poll:
                in   r18, TIFR0
                andi r18, 0x02          ; Mask for Output Compare A flag
                breq poll               ; Check if Output Compare A flag is set
                ldi  r18,0x02
                out  TIFR0,r18          ; Clear Output Compare A flag
                ret
```