**ECEN315** **Lab Project**

# Control of a Motorised Pendulum
# Lab 4: Practical open loop control.

## 1. Previous – Open loop response of the system

In the first three labs we have modelled and experimentally measured the open loop response of our system. The pendulum position was controlled by our setting of the output voltage of the power supply – not a very practical way to set the control voltage in an automated control system. In this lab we will then incorporate a pulse width modulation (PWM) motor control scheme in order to facilitate automatic control.

## 2. Motor driver control.

By now you may have come across the technique of pulse width modulation (PWM) for motor speed control. This is far more efficient than a resistive voltage divider type control scheme. The PWM controller controls the duty cycle of the power supply by switching it on or off to create a square wave with a duty cycle D. If the amplitude of the voltage during the "on" cycle is $V_{\max}$ and during the "off" cycle is $V_{\min} = 0$, then the effective voltage that the motor experiences is given by:

$$V_{avg} = D \times V_{\max}$$

In our controller we use an analog input signal (0 – 5 V) from our Simulink program set the PWM duty cycle and thus the effective motor voltage.

## 3. Calibration

Calibrate the PWM unit by setting power supply to a constant 10 volts, drop a PWM output into your model and select pin 4. There is a resolution of 255 steps on the Arduino output, vary the output from 0 to 250 in steps of 25. Use a resistor as a load and for each of these input steps measure the PWM duty cycle on an oscilloscope and calculate what the effective DC voltage that the motor sees. When you have established this analog input – PWM output relationship, drive your motor and propeller with signals having different PWM duty cycles and measure the corresponding rotational velocities. How does the rotational velocity under a PWM signal compare to that previously obtained from a DC signal in Lab 3?

## 4. Testing open loop control

You should have calibrated your system earlier for an open loop control system, i.e. you should know what motor voltage to apply in order to reach a specified target angle for the pendulum arm. Change your Simulink model to incorporate the PWM unit and then calibrate the input-output characteristics again.

Use the calibration above to design an open loop controller that can be used to set an angle between $15^{o}$ and $30^{o}$ for the pendulum (the upper point may be constrained by the achievable performance of the motor). Test the open loop response to a setpoint of $20^{o}$. Measure the transient response characteristics as well as the steady state error. Determine all the relevant response characteristics.

What happens if the pendulum is disturbed (bumped) after reaching equilibrium?
Record the response to a disturbance.