

Prelab	Participation	Lab
√+ √ √- 0	√+ √ √- 0	√+ √ √- 0

Name: _____

8 Lab: Motor Control

The purpose of control system design is to determine an appropriate input to an *actuator* (e.g., voltage to a *motor*) to obtain a desired (or nominal) output (e.g., motor speed). Control systems help satellites to track distant stars, airplanes to follow a specified trajectory, cars to travel at a designated speed, disk-drives to spin at designated angular speeds, and humans to walk, hear, and regulate body temperature. In this lab, you will use system identification techniques to determine a motor's relevant physical parameters. You will implement a control law to control the orientation and angular speed of a motor. You will look at the step response of the motor to various inputs.

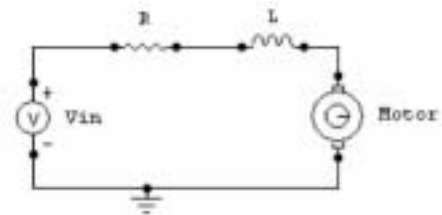
8.1 PreLab: Working Model and brainstorming

1. Download the following Working Model simulations from the class website:
MotorControlWithOnOffAndDeadBand.wm2d
MotorControlWithKpKi.wm2d
2. Run the Working Model simulations.
Record results on the Working Model PreLab (see back of the book).
3. Provide an example of a control system encountered in engineering:

4. Provide an example of a control system used by a human, animal, or plant:

8.2 Experimental

The system to be controlled is a motor whose rotor (shaft) is attached to a rod. Using Newton's laws and circuit analysis, the equation relating v_i to motor orientation is as follows (this should be familiar from the homework):



$$\frac{L(J_m + J_{rod})}{k_m} \ddot{\theta} + \frac{Lb_m + R(J_m + J_{rod})}{k_m} \dot{\theta} + \left(\frac{Rb_m}{k_m} + k_v \right) \dot{\theta} = \tilde{v}_i$$

8.2.1 Estimation with an Encoder

We use several pieces of equipment to measure and record the motor's angular speed, namely, we use an encoder, a microprocessor, a transceiver, and a computer.¹¹ This equipment is briefly described below.

- **Encoder:**

Our optical quadrature encoder determines our motor's rotational speed by detecting alternating light and dark patterns on a disk (see figure on right). A quadrature encoder has the ability to detect both angular speed **and** direction. Our encoder counts $1000 \frac{\text{tics}}{\text{rev}}$.



- **AVR ATmega168 microprocessor:**

The microprocessor receives a digital signal from the encoder (i.e., bits of one and zero) and counts the number of times the encoder changes from light to dark in a 2 ms interval.

- **DS275 transceiver:**

The transceiver receives bits (ones and zeros) from the microprocessor and translates it to the standard RS-232 serial port communications protocol. This transceiver is somewhat specific to the microprocessor, the serial port, and the communications protocol. For practical purposes, information received by the transceiver is instantaneously translated to the serial port.

- **Computer:**

The computer receives bits from its serial port and uses the software program Teraterm to translate the bits to integer numbers which are then printed to the screen. The numbers displayed on the computer screen are in units of $\frac{\text{tics}}{2 \text{ ms}}$.

The following equation converts from the units displayed on the computer screen to the motor's angular speed in RPM.

$$\frac{1 \text{ tic}}{1 \text{ interval}} * \frac{1 \text{ interval}}{2 \text{ ms}} * \frac{1000 \text{ ms}}{1 \text{ second}} * \frac{60 \text{ sec}}{1 \text{ minute}} * \frac{1 \text{ rev}}{1000 \text{ tics}} = \frac{30 \text{ rev}}{1 \text{ minute}}$$

8.2.2 Collecting data for system identification

1. Login to the computer. Username: me161student. Password: 1euler1
2. Make sure domain says ENGR
3. Create a folder on the desktop for your lab group (put all your files in it every week)
4. From the desktop, open TeraTerm Pro (ttermpro) (or Start menu -> Programs -> ttermpro)
5. Click the button for serial connection Com1
6. Go to File→Log and make a new log file in your group folder
7. Turn on the power supply. Make sure the variable power is at a low voltage.

¹¹Most motors do not come attached to a rotary encoder and assembled with a encoder, microprocessor, transceiver, and computer.

8. On the terminal screen a menu should appear.
Choose the “Motor Control” option.
9. Enter values for k_p , k_i , and k_d .
10. Observe the motor and rod setup.
11. Plot the data (e.g., using Excel, Matlab, or MGPlot) and determine ζ , ω_n , etc.
For the yAxis, convert encoder counts to radians by multiplying by $\frac{2\pi}{1000}$
For the xAxis, convert to seconds by dividing by 200 (i.e., each tick is 5 ms)
Note: Use the oscilloscope to ensure $V_{cc} = 12$ volts.
12. Email the data files and/or graphs to yourself and your group members.

Given a position PD control law of $\tilde{v}_i = -k_p\tilde{\theta} - k_d\dot{\tilde{\theta}}$ with $k_p = 1$ and $k_d = 0.01$, experimentally determine the values for ω_n and ζ from a motor step response of 10 rad. (Assume $L \approx 0$ from here on.)

$$\omega_n = \boxed{} \frac{\text{rad}}{\text{sec}} \qquad \zeta = \boxed{} \text{ no units}$$

In the equation above, whenever moment of inertia shows up it is a combined term of J_m and J_{rod} . Find the value of the effective inertia, J_{eff} . (The mass of the rod is about 0.072 kg and you can measure its length.)

$$J_{rod} \gg J_m \quad \Rightarrow \quad J_{eff} = J_{rod} + J_m \approx J_{rod}$$

The mass of the rod $m_{rod} \approx 0.072$ kg, and the length of the rod, $L_{rod} \approx 0.40$ m, hence

$$J_{rod} = \frac{1}{12}m_{rod}L_{rod}^2 \approx \underline{\hspace{2cm}}$$

Using the governing ODE and PD control law, write the equations for ω_n and ζ as functions of k_d and k_p (do this symbolically - without numbers).

$$\omega_n = \qquad \zeta =$$

Determine the value for the motor resistance R and motor's damping b_m for the motor with a rod attached? (Note: The last lab showed $k_m = k_v \approx 0.041$ N m/Amp.)

$$R = \boxed{} \Omega \qquad b_m = \boxed{} \text{ N m sec}$$

8.2.3 Position Control

Analytically calculate values for k_p and k_d that give an overshoot of 0.257 and a settling time of 1.75 sec. How confident are you that this will work? (1=doubtful, 5=very confident) [1 2 3 4 5]

$$k_p = \underline{\hspace{2cm}}$$

$$k_d = \underline{\hspace{2cm}}$$

Try these values for k_p and k_d on the actual system. Graph the data and calculate the experimental overshoot M_p and settling time t_{settle} . Show one of the TA's your graph when you have this. What do you think of the model?

$$M_p = \underline{\hspace{2cm}}$$

$$t_{settle} = \underline{\hspace{2cm}}$$

What is the fastest settling time you can get with critical damping? What are your values for k_p and k_d ? (Hint: Pick a k_p value (5-10) and solve for k_d so the system is critically damped. Then keep trying!)

$$t_{settle} = \underline{\hspace{2cm}}$$

$$k_p = \underline{\hspace{2cm}}$$

$$k_d = \underline{\hspace{2cm}}$$



Why can't you get a faster settling time?

Do you get any steady-state error? If so, how much and what do you think causes it?

Write out a new PID control law (with integral control) and then try various values for k_i on the actual system. What happens? How does integral control eliminate steady-state error? Do you have to be careful with your value for k_i ?

$$\tilde{v}_i =$$

8.2.4 Velocity Control

Note: The velocity control on the microcontroller rounds the target velocity you give it to increments of $\frac{2\pi i}{5} \approx 1.26 \text{ rad/s}$. This number arises from the combination of running the control software at 200 Hz and having 500 gaps in the encoder.

Implement velocity PI control. What is the governing equation when using PI control?

$$= 0$$

Find k_p and k_i to give a peak of $30 \frac{\text{rad}}{\text{sec}}$ at $t = 2 \text{ sec}$ to a $24 \frac{\text{rad}}{\text{sec}}$ step. Try to get the motor to spin at the desired rate of $24 \frac{\text{rad}}{\text{sec}}$.

$$k_p =$$

$$k_d =$$