Lab03 - Motor Control

MTRN3100 - UNSW School of Mechanical and Manufacturing Engineering

Task 1 - DC Motor

Wire up one motor and motor controller. Attach a wheel to the output shaft of the motor such that you can easily backdrive it in future tasks. Follow the wiring guide from the lab slides. Use the 5V Arduino output as the voltage input to the motor controller. Complete the stub functions in the class "Motor.hpp". Write an Arduino program to turn the output shaft one rotation, then pause for 3 seconds and loop. As this is open-loop control, do not spend too long getting the timing to be exactly one rotation.

Additional information for the motor driver board can be found: here

Questions

- **Question 1:** Try different speeds starting from zero, is there a minimum PWM signal that gets the motor spinning? Is this different for each direction? (Don't spend too long on this task)
- **Question 2:** Explain why open-loop control is generally undesired for robotics. Give an example concerning motor speed control.
- **Question 3:** What is the advantage of using a motor controller? Hint: What if we used the single MOSFET from lab 1?

Task 2 - Encoder

Connect the encoder to the Arduino. Remember to connect only one of the encoder wires to the Arduino interrupt pin. The other encoder pin can be placed on any digital input port. Complete the stub functions in the class "Encoder.hpp". Print out the encoder count using the serial monitor or serial plotter.

The reference sheet for the Arduino interrupt function can be found: here

Questions

- **Question 4:** How many encoder counts are there per revolution of the motor? You can find this value by running the code from task 1 or hand-driving the motor one rotation.
- **Question 5:** If a second interrupt was used, what additional information could be gained?

Task 3 - getRotation()

If not completed, finish the getter function "getRotation()". It should return the current rotation in radians when called.

Questions

There are no questions for this task; however, this is a useful function for the next tasks. Verify that this code works through trials. Make sure it works for clockwise and anticlockwise motor speeds. Your function must return the value as a float.

Task 4 - Bang Bang Controller

Complete the provided BangBangController class. It should take in a desired position and move to the desired angle. Your code will have two hyperparameters, speed and deadband. For the sake of this experiment, set the deadband to zero. Adjust the speed to explore the results.

Questions

- **Question 6:** Set the controller deadband to zero and the target position as one full rotation. What is roughly the maximum PWM signal that does not cause oscillation?
- **Question 7:** Set the controller deadband to zero and the target position as one full rotation. Observe the oscillation when the PWM signal is set to a maximum value.
- **Question 8:** The oscillations can be counteracted with a deadband. What is the trade-off in using a larger deadband?

Task 5 - P Controller

Complete the provided PIDController class. It should take in a desired position and move to the desired angle. Your code will have 3 hyperparameters, kp, ki, kd. For the sake of this task, set $ki, kd = 0$. Set the target as 0.5 rotations and explore different kp values.

Questions

- **Question 9:** What can happen if the kp term is too large (ie $kp >= 8000$)?
- **Question 10:** What can happen if the kp term is too small (ie $kp <= 100$)?
- **Question 11:** In one sentence, explain how the proportional part of the controller works.
- **Question 12:** Backdrive the motor, how does the controller respond?

Extension 1 - PID Controller

Implement the integral and derivative control terms if you have not already. Set the target as 0.5 rotations, and explore different kp, kd, and ki values.

Questions - NOT MARKED

- **Question 13:** What can happen if the kd term is too large (ie $kd >= 1000$)?
- **Question 14:** What can happen if the kd term is too small (ie $kd <= 10$)?
- **Question 15:** What can happen if the ki term is too large (ie $ki >= 100$)?
- **Question 16:** What can happen if the ki term is too small (ie $ki <= 5$)?
- **Question 17:** In one sentence, explain how the derivative part of the controller works.
- **Question 18:** In one sentence, explain how the integral part of the controller works.

Extension 2 - Open discussion and extension

This task is simply an open discussion between yourself, your classmates, and/or a demonstrator.

Questions - NOT MARKED

- **Question 19:** From your experience in this lab, do you think bang bang or PID control is necessarily better for the micro-mouse challenge?
- **Question 20:** In the case of using PID, are all the terms required?
- **Question 21:** How would different motors with higher/lower gear ratios affect your answers to the previous two questions?
- **Question 22:** What is the reason for the zeroAndSetTarget() functions in both controllers? After reaching the setpoint how would you move to the next setpoint? This will be covered more in Lab04.