ECE 206: Motor Proportional Speed Control

Closed loop feedback control is important to delivering solutions to many products and processes. This lab introduces closed loop feedback control using the STEMTera to control the Sparkfun Hobby Motor's speed.

In the conceptual system (Figure 1.), the user sets the desired speed (aka "reference" speed) in the microcontroller's instructions (the Arduino "sketch"). The microcontroller compares the motor's speed with the reference speed. The difference between the reference speed and the actual speed is referred to as the "error". The microcontroller updates the control signal to reduce error – ideally reaching zero error (i.e. motor's set and actual speed match).

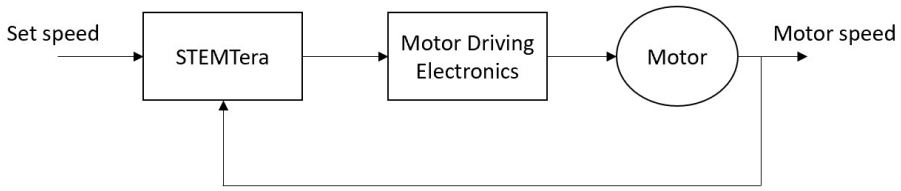

Figure 1. Closed loop motor speed control concept.

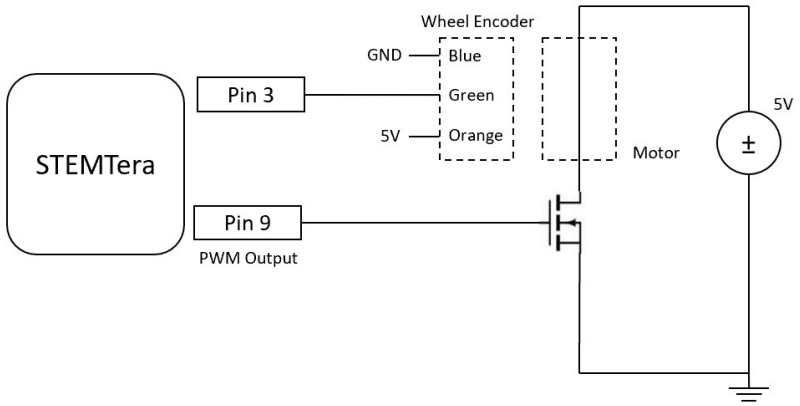

Figure 2. Motor closed loop proportional speed control circuit using STEMTera, DC motor and encoder (4 magnet disk and Hall effect sensor).

PID is a common control scheme that updates the control signal based upon error magnitude (proportional), accumulated error (integral), and error rate of change (derivative) – hence Proportional-Integral-Derivative (PID) control. In this introduction to feedback control, we will modify the control signal in "proportion" to the error's magnitude.

The lab builds upon previous Encoder, DC Motor, and Pulse-Width-Modulation (PWM) laboratories.

Challenges:

- Build the motor circuit (Fig. 2) used in the MOSFET lab exercise. You don't need to include a low-pass filter this time. Recall the MOSFET controls gate voltage to amplify the STEMTera output current to drive your DC motor.

- Write, compile, and implement microcontroller code implementing proportional feedback control to maintain motor shaft speed at speed that you have set. Recall that PWM was used to control motor speed using values ranging from 0 to 255. Your objective is to use encoder derived information to adjust your PWM value. The adjustment should be proportional to the difference between set speed and actual speed (error, e), i.e. your adjustment should be $K_p * e$. (K_p = .5 is a reasonable starting value; you may adjust as needed).

    - Most of the code is already provided to you in the website. You only need to change three lines of 'update_speed()' function to complete this lab.

- Demonstrate your TA that your DC motor can update the speed to a given reference speed.

References:

- What PIDs do and how they do it, https://www.youtube.com/watch?v=0vqWyramGy8 accessed 17 Jul 17.

Prelab Deliverables:

- Read the provided code in the website. What does 'attachInterrupt' function do? Also describe how to calculate currRPM, error, and currPWM inside the 'update_speed()' function.

Report Deliverables:

- Provide descriptions with accompanying diagrams of your circuits and experimental setup; include annotated photographs of your experimental setups.

- Include an image of Serial Plotter, which shows refRPM and currRPM at three different reference speeds.