Lab 4: Motor Control

In this lab you will characterize a DC motor and implement the speed controller from homework 3 with real hardware and demonstrate that a brushed DC motor can run at a constant speed under varying load. Make sure to pull the latest starter code from the class Github repository.

Assigned: October 16, 2017

Due: Week of October 23, 2017
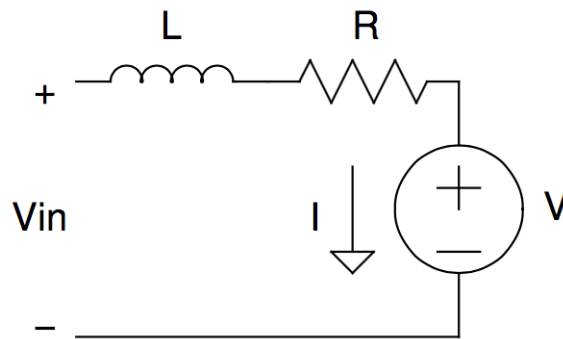
Part 1

Motor Model

Figure 1: Motor model

A brushed DC motor can be modeled as shown in Figure 1. The inductance and resistance are properties of the motors coils. The current I through the motor is proportional to the torque exerted by the shaft, and the voltage V is proportional to the rotational speed ω.

Definitions:

R	Resistance in ohms
L	Inductance in henries
V	Back-EMF (voltage produced by rotation) in volts
I	Current through the motor
K_e	Back-EMF constant, $\frac{V}{rad/s}$
K_T	Torque constant, $\frac{N \cdot m}{A}$
T	Torque, $N \cdot m$
J	Load inertia, $\frac{kg \cdot m^2}{rad^2}$

Using the units shown above, $K_e = K_T$. Motors are often specified with a speed constant K_V in $\frac{RPM}{V}$.

Motor behavior:

$V = K_e\omega$
$T = K_T I$
$\dot{\omega} = \frac{T}{J}$
$V_{in} = K_e\omega + IR$

Safety

All probes on the oscilloscope share a ground connection. Dont connect probe grounds to anything other than circuit ground, because you could create a path which carries a large current and melts the probes. Be careful!

Lab power supplies can be used to test for basic functionality, but it is not safe to operate the motor and an H-bridge without a parallel load attached. The H-bridge and motor become a boost converter (this is called regenerative braking), converting the speed-induced voltage in the motor to a higher voltage, which can easily exceed the maximum VDS of the MOSFETs and possibly damage the power supply. Lab power supplies cannot absorb power, and they will not prevent an excessive voltage from developing on their output (some have a down-programmer which can absorb a small amount of power, and it could be destroyed by regeneration). A dryer load across the power supply terminals provides another path to dissipate the energy from regenerative braking and protects the lab power supply.

Keep your wiring clean. Unintended connections can cause damage, such as 12V touching any pin on the microcontroller.

Motor and Load Assembly

The motor is a Jundi 555-2580.

The pinout for the encoder is GND, SIG, +5V.

The encoder wheel has twenty white and twenty black sectors, so the frequency on the encoder output is twenty times the rotational frequency. The forty sectors may not be exactly the same size, so averaging over many periods may improve your speed measurements.

Part 2

Electronics

A schematic diagram of this lab is shown in figure 2. The input capacitor is located on the component board. Take care to wire everything cleanly to minimize noise when measuring the speed of the motor.

The rotary encoder comprises of a LED and photodiode pair which measures the reflectance of the spinning black and white disk on the motor assembly. This generates a square wave at twenty times the frequency of the motor and is what we will measure to calculate the speed of the motor. Because the photodiode outputs

an analog voltage corresponding to the reflected light, we need to add a comparator to make sure the voltage fully swings between the supplies. The schematic for a basic comparator is shown in figure 3. The compare value is set by the voltage divider. Start off with 2.5V.

You may also find that you need to add bypass capacitors between the sensor supply line and ground to minimize noise that is coupled to the supply due to the commutation of the motors.

The half-bridge board contains two MOSFETs and a gate driver chip. You need to provide 12V to power the gate driver. The two inputs (Lin and Hin) control the two MOSFETs and should be tied together. To prevent shoot-through, the gate driver chip will insert an appropriate dead time when the inputs change so that both MOSFETs are not on at the same time.

The half-bridge board contains these parts:

1. IRLB3036-This is the MOSFET used in the half-bridge modules. This determines the maximum voltage and current that can be safely supplied to the half-bridge board. Note that both are N-channel devices.

2. IRS21834- The MOSFET driver used in the half-bridge modules. This takes PWM signals from the microcontroller and drives the relatively large gate capacitance of the MOSFETs. It shifts the high-side gate drive signal so that it is a fixed voltage above the high-side MOSFETs source terminal.

The design of the half-bridge board largely follows the **Typical Connection** figure in the IRS21834 datasheet.
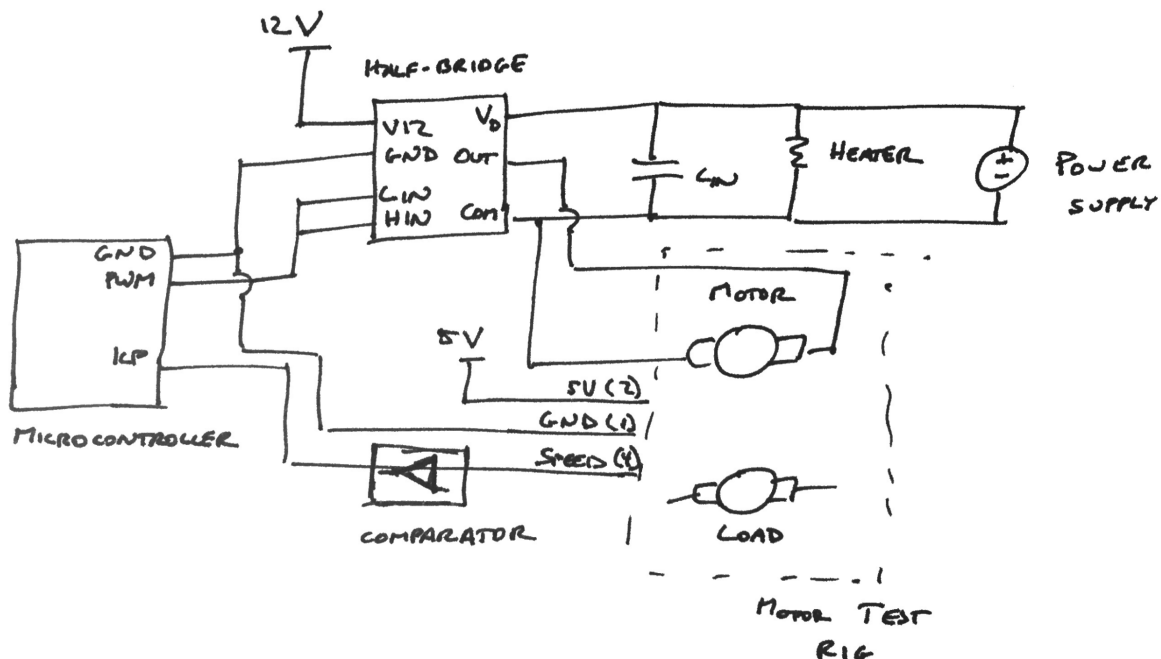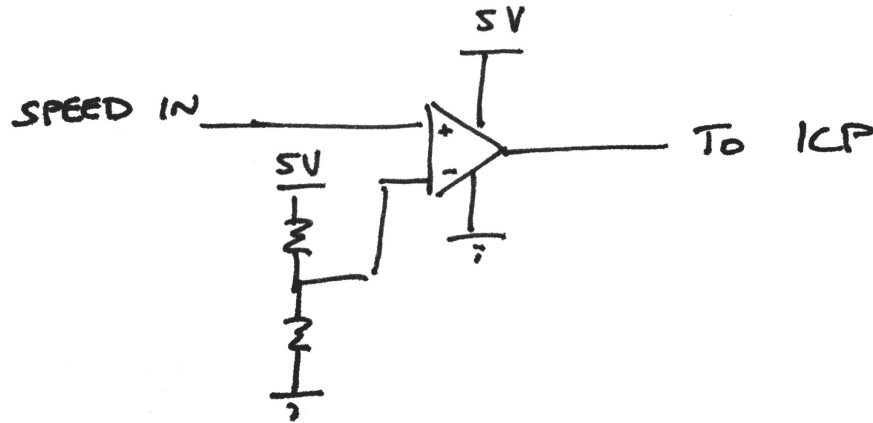

Figure 2: Motor test setup.

Figure 3: Motor test setup.

Regeneration

The 10Ω dryer heater load resistor is used to absorb power from the motor when it is braking. When the motor brakes, the motor and the H-bridge form a boost converter which drives power back towards the power supply. This behavior is called regenerative braking, or regeneration, and is a useful feature in systems like electric vehicles which can store the regenerated energy. The bench power supplies cannot absorb any power, so without the resistor the supply voltage will rise (potentially to dangerous levels) and the power supply or other electronics could be damaged. As long as the current due to regeneration is less than what the load resistor on its own would draw from the power supply, the supply voltage will be controlled by the power supply.

Some applications, like machine tools and subway cars (due to poor third-rail connections), can't always store regenerated energy and must dissipate it in resistors. The setup used in this lab is very inefficient because the resistor always dissipates power. What would be a more efficient design?

Part 3

Controller Implementation

Based on your MATLAB simulations, write a program that holds the motor speed at an approximately constant, adjustable value, even when the load is varied. The motor speed should not oscillate indefinitely and should exhibit little, if any, overshoot. The motor should reach its target speed in about two seconds or less.

Your program should use buttons to change the speed setpoint up and down, to change the setpoint to a high value immediately (step input), and to safely and quickly stop the motor. It should display the current speed and speed setpoint.

You should expect some variation from the simulation to the real hardware. How much did you have to change the controller design or parameters to get acceptable performance on real hardware?

Regardless of the user inputs, the controller should operate the motor in a way that does not produce excessive oscillation and that does not exceed the power supply's current limit.

If your controller uses motor current, do something similar to lab 2/3 with a return side sensing resistor fed to the controller's ADC through the breakout board.

The starter code for lab 4 can be used to measure the speed of the motor by measuring the period of the signal from the encoder. The starter code implements a basic median filter to digitally filter the speed of the motor. Depending on your setup, you may need to be even more aggressive in your filtering. NOTE: The current starter code has one small bug - find it!

Testing

Use the motor on the side of the test rig near the sensor connector as the load. The other motor is the one your code will drive.

The load motor has a rheostat across it which can be adjusted to change the load seen by the controlled motor. The rheostat is connected when the toggle switch is on, so the step response to load changes can be easily observed. Other electrical loads can be connected across the banana jacks.

Note that the rheostat is rated for 25W, but you may be able to deliver more power than this to the load motor. It is suggested that you only have the rheostat connected for short periods of time (around ten seconds) if it will be dissipating significantly more than 25W.

Signoffs

1. Demonstrate the motor reaching a set speed and staying there with little or no oscillation.

2. Show that the motor settles at the same speed after a load is applied and again after it is removed (within the limits of the power supply and motor).

3. Explain how the behavior of your simulation matches and differs from the real hardware.