

3 Lab 3: DC Motor Voltage to Speed Transfer Function Estimation by Step Response

3.1 Introduction

This lab introduces new methods for estimating the transfer function of a plant (in our case, the DC motor). Recall that in Lab 2 the transfer function of the motor was obtained by measuring the various physical parameters of the motor and applying them to the known mathematical model. The method to be used in this lab, called step-response methods, is indirect and can be used even if a good mathematical model of the plant is unavailable.

3.2 Pre-lab

1. For a first order system, how can the transfer function be estimated from the step response, i.e. how are the DC-gain and time constant found?
2. Determine the transfer function from the step response graph in Figure 1.

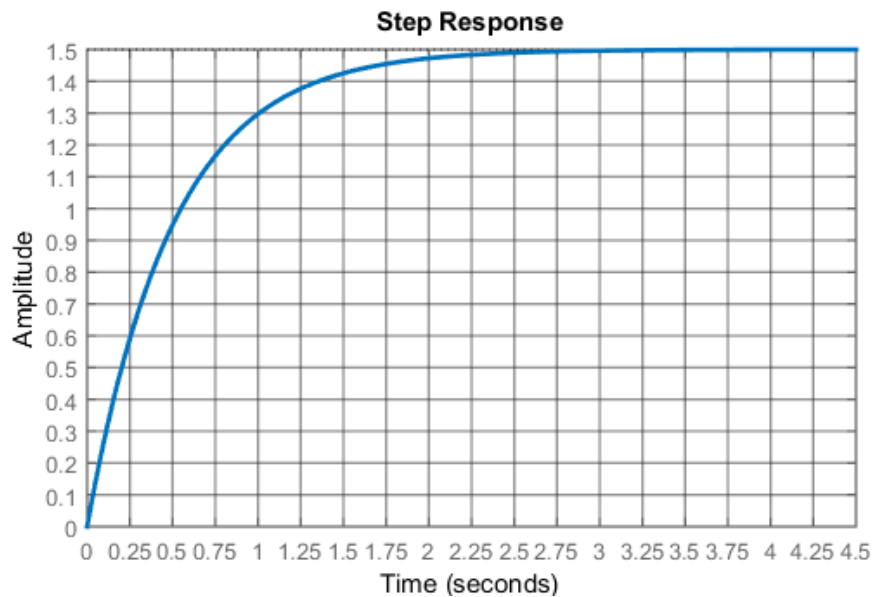


Figure 1 – Step Response of unknown system

3.3 Objectives

After this lab, you will be able to:

- Identify the important parameters in a step response
- Determine the transfer functions from a step response.

3.4 References

Feedback Control Systems (2011), Fifth ed. by Charles L. Phillips and John M. Parr.

Note: the same information is in *Feedback Control Systems* (2000), Fourth ed. by Charles L. Phillips and Royce D. Harbor.

- Step Response: see Chapter 4

3.5 Simulation of Motor Transfer Functions in Simulink.

First, your TA will give you a short lecture on how to use Simulink in MATLAB. You will then build a Simulink simulation to compare the estimated transfer function from Lab 2 to the actual output data to the motor. In order to compare this transfer function to the one you will determine in this lab, you will have to add the gain of the amplifier, $K_{Amp} = 2.4$, and the gain of the tachometer, $K_{Tach} = 0.03$. The transfer function that will be identified in this lab is

$$\frac{V_{Tach}(s)}{V_{input_to_Amp}(s)} = \frac{K_{Amp} K_m K_{Tach}}{s\tau_m + 1}$$

Note: This transfer function does not take into account the static (coulomb) friction acting on the motor's rotor. We will identify this coulomb friction as a part of this lab but not do too much with it in the analysis portion of this lab. Make sure to record this value though, as it will be important in simulations for Lab 5.

Start by simulating the motor transfer function found in the post lab questions of Lab 2. Use Figure 2 and the steps below to build a Simulink simulation of the motor.

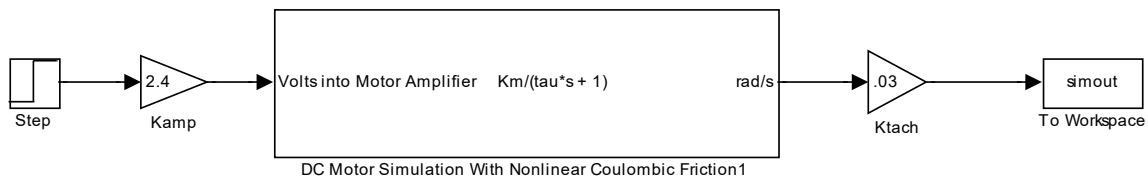


Figure 2 - Simulink Model for DC Motor Simulation

Creating a SIMULINK simulation

- 1) Start MATLAB.
- 2) Load Simulink by typing "simulink" on MATLAB's prompt.
- 3) In the Simulink window, click the file folder button to open a file. Open **SE320_lab3starter** from [N:/labs/se320/exp3](#). This will open a starter shell for your simulations. You will notice that it also includes a "special" motor block that simulates the motor's transfer function along with the coulomb friction acting on the motor.

- 4) Using a unique name, save this Simulink file to a folder that you create on the c:\ drive of the lab PC where the folder name is your NetID c:\YourNETID directory.
- 5) To set the simulation time and other simulation parameters, right click in any white area of your Simulink window and select **Model Configuration Parameters**. Under “Time Simulation,” change the value of “Stop Time” to 5 sec.
- 6) From the Simulink Library Browser window select the **Commonly Used Blocks** category.
- 7) Drag a **Gain** block into the model.
- 8) Go back to the Simulink Library Browser window and look for the category **Sinks**
- 9) Drag a **simout - To Workspace** block to the model. This box will output your measurements as variables in MATLAB (simout = Voltage; tout = time).
- 10) Go back to the Simulink Library Browser window and look for **Sources**.
- 11) Drag a **Step** block source to the model.
- 12) Now connect the boxes so they match Figure 2. To connect them use the small arrows on the edges of each block. Left click on them and drag the connection to the next block.
- 13) Double click on each box and change the values so they match Figure 2 and these items below:
 - a. In the DC Motor block initially set the motor gain and time constant, K_m and τ_m , to the values you identified in lab 2 and set coulomb friction to 0. Later in this lab you will change these values to ones identified in this lab.
 - b. Make sure the **Step** block is configured with the following parameters:
 - i. Step Time - **1** (the source is turned on after 1 second)
 - ii. Initial Value - **1.25** (This is the initial value you input on VEE)
 - iii. Final Value - **4.5**
 - iv. Sample time - **0**
 - c. Make sure your **simout** block has “Save Format” and set it to “Array”.
- 14) After setting up every block, hit the play symbol (“Start simulation”) on the bar, or go to **Simulation -> Start**.
- 15) Graph your response using the following MATLAB command prompt command:
plot(tout,simout)
- 16) Show your work to your TA.
- 17) Print and label this step response and your simulation block diagram to be included in the post lab. You may find the following commands useful:
 - a. title('You can enter your graph title here')
 - b. xlabel('You can enter your x-axis label here')
 - c. ylabel('You can enter your y-axis label here')

3.6 Transfer Function Estimation by Step Response

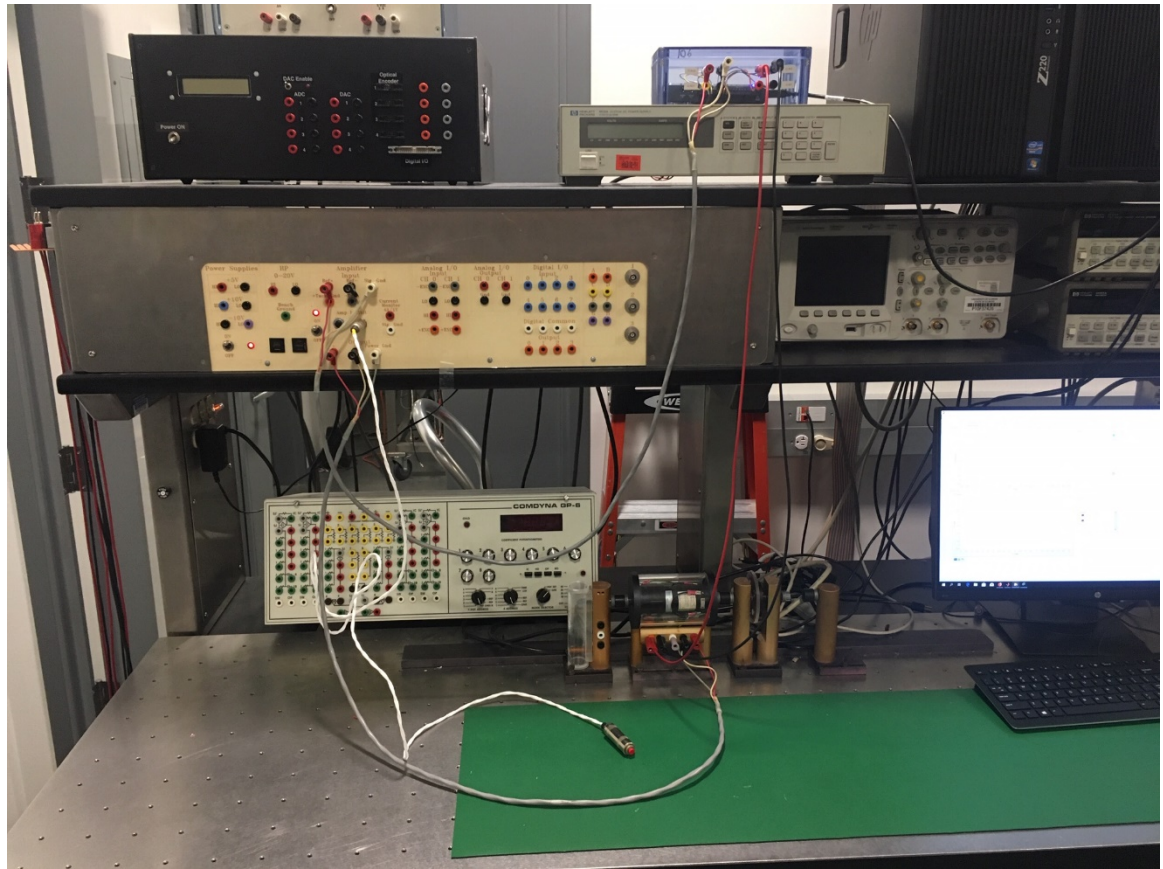
The transfer function of an unknown plant can be obtained by analyzing its step response. As seen in Lab 2, the transfer function of motor Voltage to Angular Velocity can be approximated by a first order transfer function.

$$\frac{\Omega(s)}{E_a(s)} = \frac{K_m}{\tau_m s + 1}$$

The goal of this section will be to estimate the values for K_m and τ_m by analyzing the response of the motor to a step input voltage. You will be implementing the block diagram in Figure 2 with your actual motor. A step function will be input to the motor, and the signal from the tachometer will be observed. Remember that the voltage of the tachometer is proportional to the angular velocity of the motor. You will also use this step response data to estimate the coulomb friction acting on the motor.

Testing Procedure:

- The myRIO analog output can only generate outputs between ± 10 V with a maximum current of 10 mA. We will use the amplifier built in the patch panel to make sure there is a large enough voltage and current to overcome the deadzone of the motor. This amplifier has a gain of $K_{\text{Amp}} = 2.4$. Connect the output of the myRIO to the input of the amplifier. Then, connect the output of the amplifier to the motor. Match the color of the ports with the color of the jacks.
- The myRIO also has analog inputs. The analog 0 input channel will be used to measure the tachometer voltage. Connect the tachometer to the analog 0 input to the myRIO.
- Now that the motor is connected to the myRIO, we will build a program in LabVIEW to generate the input to the motor and collect the data from the myRIO.



- Open the LabVIEW starter file for the myRIO program:
 - (a) In the C drive, if you have not yet, create a folder with your NetID. Then, open the myrio folder from N:\labs\se320\exp3 and copy the two files to your newly created folder in C drive.
 - (b) Open LabVIEW and select **Open Existing**. Open the “Lab3.lvproj” project file. After this, the project explorer window should appear.
 - (c) Expand the NI-myRIO-1900-COESCLXYZ and open the starter file “Step_response.vi”. This will open Front Panel of the VI. Press “ctrl+E” to open the Block diagram window. Also at this time, ask your TA how to change your LabView program to run on the MyRIO located at your bench. Each bench has a different numbered MyRIO.
- You will be building a program in LabVIEW to generate a step response input to the motor and record the voltage data from the tachometer. The final Block Diagram and Front Panel are provided in Figures 4 and 5, respectively. Most of the blocks have already been created for you. You will only need to add the ones listed in red.
 - a. **Control & Simulation Loop:** Control & Simulation -> Simulation
 - b. **Step Signal:** Control & Simulation -> Simulation -> Signal Generation
 - c. **Simulation Time:** Control & Simulation -> Simulation -> Utilities
 - d. **SimTime Waveform:** Control & Simulation -> Simulation -> Graph Utilities. This will place both the Waveform Chart and the Simulation Time Waveform in the block diagram window.

- e. **Buffer XY Graph:** Control & Simulation -> Simulation -> Graph Utilities. This will place two blocks in the block diagram: Buffer XY Graph and XY Graph. You will need to make a second copy of just the Buffer XY Graph block to complete the block diagram.
 - f. **myRIO Analog Output:** myRIO. In the Configuration window that automatically opens, select C/AO0 as the Channel.
 - g. **myRIO Analog Input:** myRIO. In the Configuration window that automatically opens, select C/AI0 as the Channel.
 - h. **Build Array:** Array -> Build Array. Then resize from the bottom to get two inputs.
 - i. **Bundle (x2):** Cluster, Class, & Variant. Then resize from the bottom to get two inputs.
 - j. **Build Cluster Array:** Cluster, Class, & Variant. Then resize from the bottom to get two inputs.
- Wire the blocks up to match Figure 4.
 - Right click on the box in the attached to the upper left corner of the **Control & Simulation Loop** and select properties. Make sure it matches the values in Figure 6 and Figure 7. In the Simulation Parameters tab, change the solver to **Runge-Kutta 1** and the Step Size to **0.01**. In the Timing Parameters tab check the **Synchronize Loop to Timing Source** box.

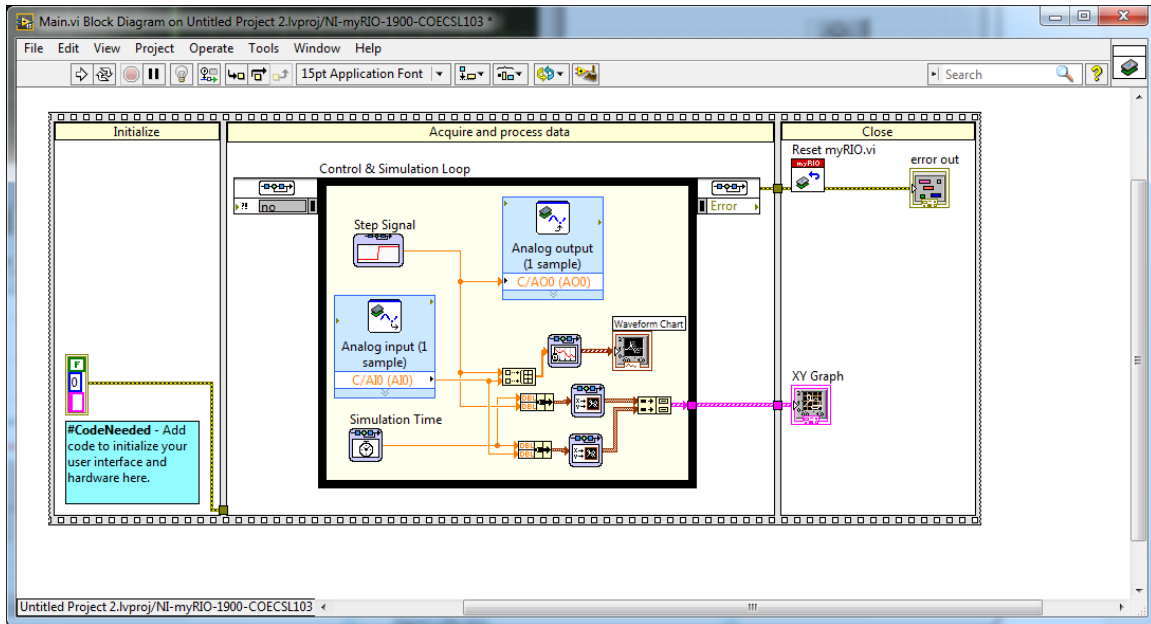


Figure 3 - Block Diagram for Step Response Program

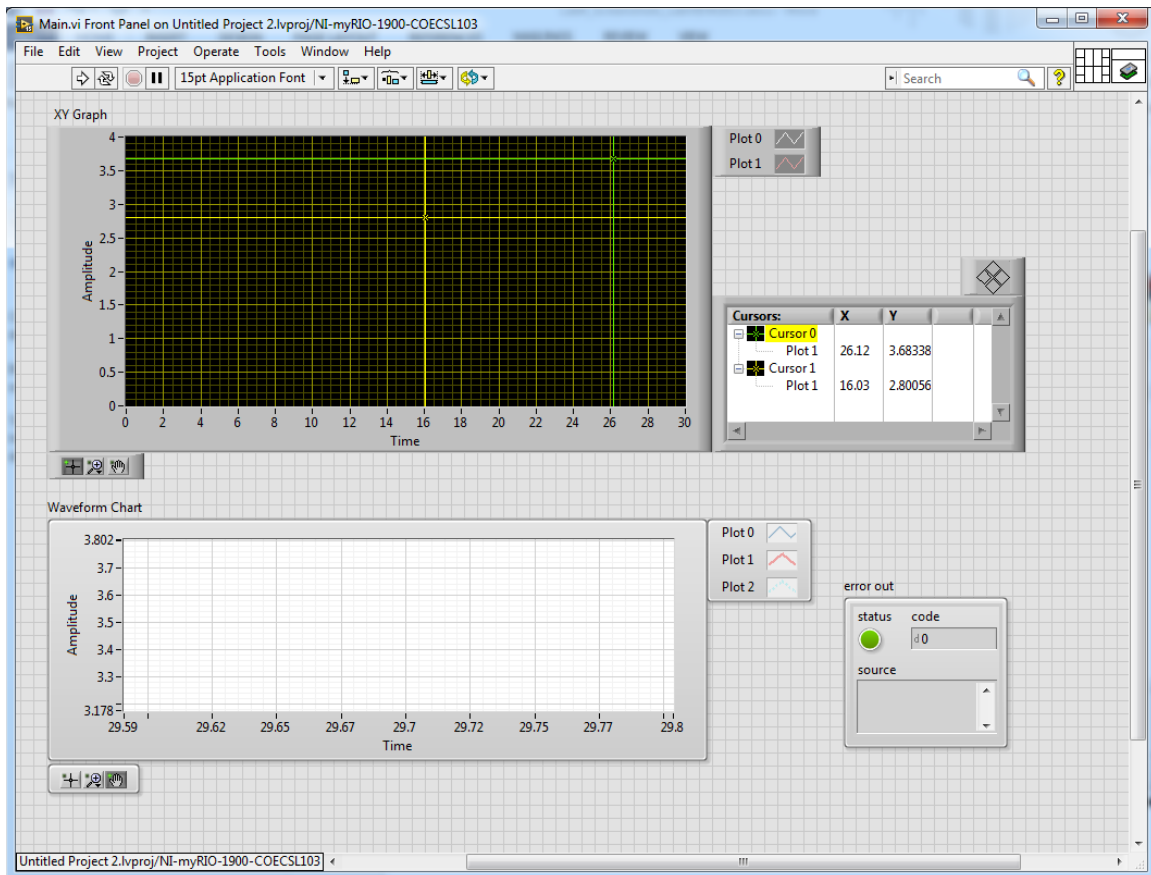



Figure 4 - Front Panel for the myRIO program

 **Configure Simulation Parameters** X

Simulation Parameters **Timing Parameters**

Simulation Time

Initial Time (s) Final Time (s)

0 10

Solver Method

ODE Solver

Runge-Kutta 1 (Euler) ☐ Nan/Inf Check

Continuous Time Step and Tolerance

Step Size (s)

0.01

Minimum Step Size (s) Maximum Step Size (s)

1E-10 0.1

Relative Tolerance Absolute Tolerance

0.001 1E-7

Discrete Time Step

Discrete Step Size (s)

0.01 ☒ Auto Discrete Time

Signal Collection

Decimation

0

OK Cancel Help

Figure 5 - Configure Simulation Parameters dialog box - Simulation Parameter tab

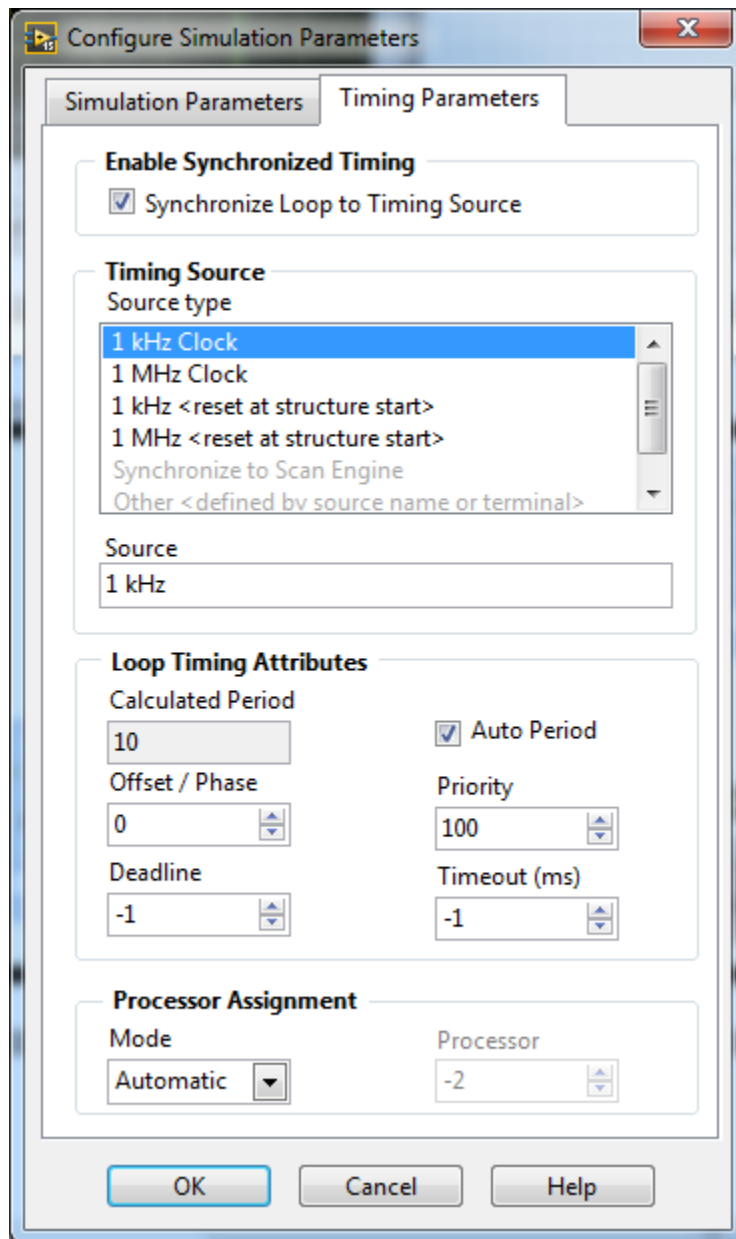


Figure 6 - Configure Simulation Parameters dialog box - Timing Parameters tab

- Double click on the **Step Signal** block to set the parameters of the step response.
 - Initial Value - **1.25**
 - Final Value - **2.5**
 - Step Time - **5 sec**
- Click the arrow button to run the program on the myRIO.
- *This step is already done for you but given here if you want to create more cursers for measurements.* There are some extra tools that you can display on the Front Panel to help with data collection. Right click on the XY Graph and add the **Graph Palette** and **Cursor Legend** as Visible Items. Then right click inside the Cursor

Legend and Create two Single-Plot cursors. The cursors will allow you move the cursor along each plot and give you the X and Y values of the plot where the cursor is located. The Single-plot cursor is also created for you in the starter file.

- Use this data to calculate K_m and τ_m and record these values in your data sheet. Remember that $K_{Amp} = 2.4$ and $K_{Tach} = 0.03$.
- Repeat the Step Response more times always starting with the voltage at 1.25V and incrementing your final voltages from 3.0 to 4.5 volts in 0.5 volt increments. Calculate K_m and τ_m for each step response and take an average of the five values found to use as your identified K_m and τ_m .
- Repeat the last step input from 1.25V to 4.5V to perform additional steps to compare your actual response to the Simulink simulation and estimate coulomb friction.
 - a. In your Simulink simulation, double click on the “special” motor simulation block. Enter the values your just identified for K_m and τ_m and initially enter 0 for the coulomb friction.
 - b. Run the Simulink simulation and compare your simulated step response with your actual data. You should notice some similarities and some differences. If you ignore the starting and ending voltages of both responses you should find that the rise time and settling time match along with the voltage distance stepped. This should make sense because the gain identification you performed above only took into account the ratio of the difference between the ending and starting output and input values.
 - c. The differences in the starting and ending voltages are due to your Simulink simulation not taking into account the coulomb friction acting on the motor. To determine the coulomb friction value, change the friction value in the motor simulation block until the starting and ending voltages of the simulation match closely with actual step response data. Start with a value of 0.25V and adjust it up by 0.02 or so until you find the correct value. Record this value in your data sheet and make sure to bring this data sheet to lab 5 as you will need this value and your K_m and τ_m to run the lab 5 simulations.

3.7 Post Lab

Include the answers to the following questions in your lab report.

1. Calculate the transfer function for the motor with the data obtained from the step-response experiment.
2. Comment on the similarity/differences between the first order transfer function estimated in the step response experiment with the transfer function obtained from governing physical equations and measured parameters in Lab 2. Remember to account for the gain of the amplifier and the gain of the tachometer.
3. Using the first order transfer function estimated in the step response experiment, design an open loop control system that will spin the motor at 100 rad/sec, and 200 rad/sec. That is, you will have to determine the voltage that needs to be applied to the motor to spin the motor at the required speeds. Approximately how long should we have to wait to achieve the required speed to within 2%?
4. Include the block diagram from your Simulink simulations (step responses). Just include two step responses that compare the transfer function found in Lab 2 to one of the transfer functions found with the step response method.

3.8 Data Sheet

STEP RESPONSE OF A DC MOTOR

Delta Vin Step size Input From To	Tachometer Avg. Starting Voltage	Tachometer Avg. Ending Voltage	Delta Vout Tachometer (Ending – Starting)	Time constant (sec)	Motor Gain (K_m)
Step of 1.25V 1.25V – 2.5V					
Step of 1.75V 1.25V – 3.0V					
Step of 2.25V 1.25V – 3.5V					
Step of 2.75V 1.25V – 4V					
Step of 3.25V 1.25V – 4.5V					

COULOMB FRICTION VALUE: