White, Michael | Lab #2, Activity #1 ME 421L

- $V_a(t) = R_a \cdot i_a(t) + L\,\frac{di_a(t)}{dt} + V_e(t)$

$V_e(t) = K_b \cdot \theta_m$

$T(t) = J_m \cdot \ddot{\theta}_m + B_m \cdot \dot{\theta}_m$; $T(t) = K_t \cdot i_a(t)$

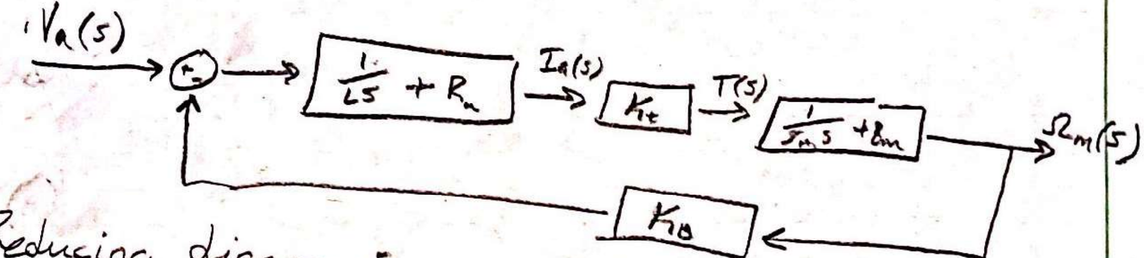
<u>Laplace:</u> $V_a(s) = R_a i_a(s) + L s i_a(s) + V_e(s)$

$V_a(s) = (Ls + R_a) i_a(s) + V_e(s)$; $V_e(s) = K_b \cdot \Omega_m(s)$

$T(s) = (J_m s + B_m)\Omega_m(s)$; $T(s) = K_b \cdot i_a(s)$

$\Omega_m(s) = \frac{T(s)}{J_m s + B_m}$

- <u>Model:</u>

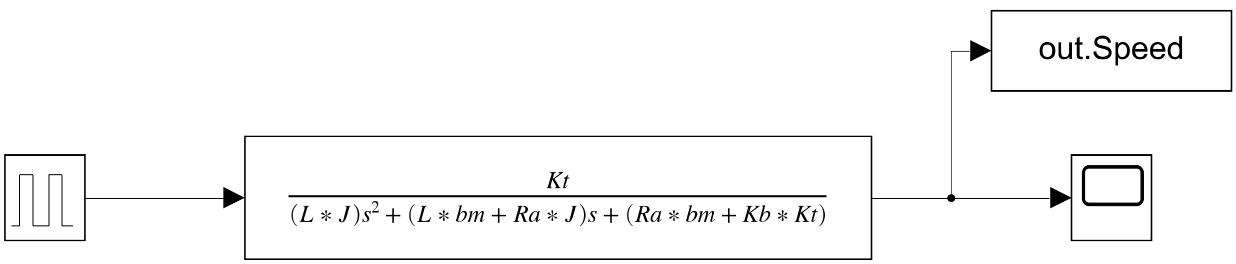


Reducing diagram:

$G(s) = \left[\frac{1}{Ls + R_a}\right] \cdot K_t \cdot \left[\frac{1}{J_m s + B_m}\right]$ $H(s) = K_b$

$$\frac{\Omega(s)}{V_a(s)} = \frac{\frac{K_t}{(Ls + R_a)(J_m s + B_m)}}{1 + \frac{K_t K_B}{(Ls + R_a)(J_m s + B_m)}} = \frac{K_t}{(Ls + R_a)(J_m s + B_m) + K_t K_b}$$

- $$\frac{\Omega(s)}{V_a(s)} = \frac{K_t}{(L \cdot J_m)s^2 + (R_a \cdot J_m + B_m \cdot L)s + (K_t K_b + R_a B_m)}$$

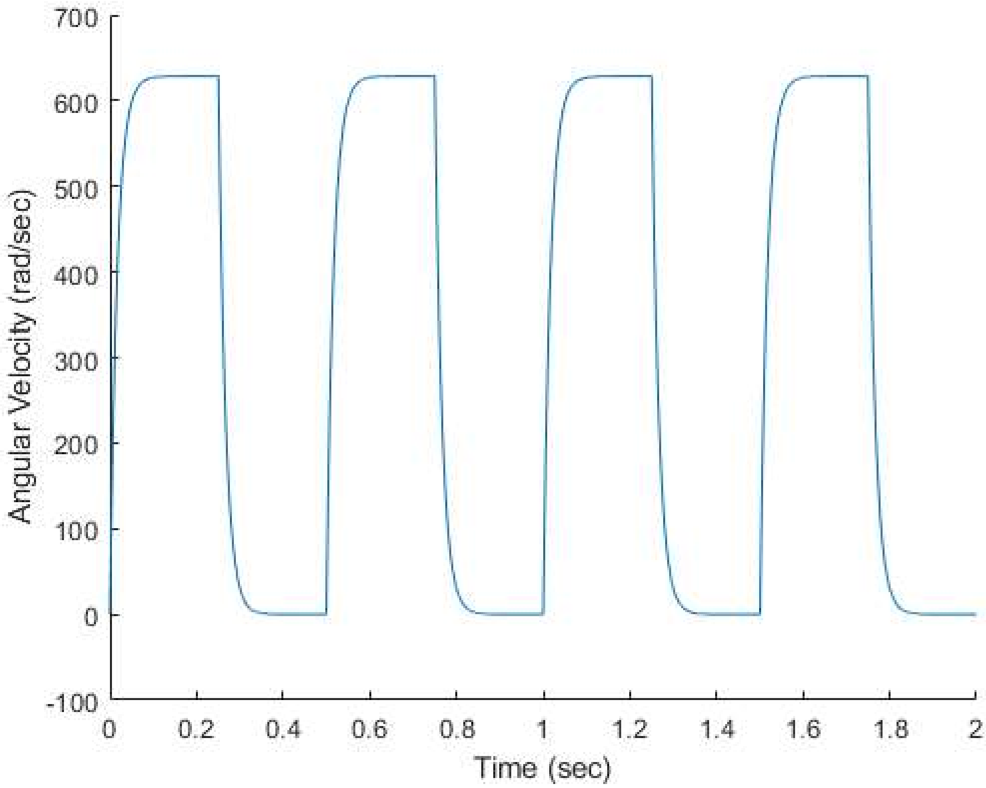

out.Speed
$$\frac{Kt}{(L*J)s^2+(L*bm+Ra*J)s+(Ra*bm+Kb*Kt)}$$

Activity 2

3/25/21 Michael White Section 3 / Online

```
close all;
clear all;
clc;

% Define parameters to be used
Kt = 7.68e-3; % Nm/A
Kb = 7.68e-3; % V/(rad/sec)
L = 0.18e-3; % H
J = 3.9e-7; % kg*m^2
bm = 8.148e-7; % Nm/(rad/sec)
Ra = 2.6; % Ohm

% Run the simulation
simout = sim('DC_Motor_Simulation_Act2');

% Generate figure and plot results
figure;
hold on;
plot(simout.Speed);
xlabel('Time (sec)');
ylabel('Angular Velocity (rad/sec)');

% WRITTEN RESPONSES:

% This system is clearly a first order system response, as it is simply
% rising up to the maximum value in each trial, and not overshooting.
% In addition, the governing function is a first order equation.

% This activity asks for the max RPM that the motor reaches, but the output
% for this system is in radians per second. This conversion and output can
% be seen below, with the result being written to the command window.

% Calculating the max RPM (including conversion from rad/sec)
maxRPM = max(simout.Speed.Data)*9.5492965;
disp(strcat("The max RPM reached by the motor is ",num2str(maxRPM)));

% Isolating the data to the first curve (contained in first 0.2 seconds)
calculationData = simout.Speed.Data(simout.Speed.Time <= 0.2);
calculationTime = simout.Speed.Time(simout.Speed.Time <= 0.2);

% Finding the time constant using the 63.2% method
maxValue = max(calculationData);
timeConstantValue = maxValue*0.632;
absDiffList = abs(calculationData-timeConstantValue);
timeConstantPoint = ...
    [calculationTime(absDiffList == min(absDiffList)),...
     calculationData(absDiffList == min(absDiffList))];

% Display the calculated time constant
disp(strcat("The time constant of the system is ",num2str(timeConstantPoint(1))));
```

```
The max RPM reached by the motor is 6001.5095
The time constant of the system is 0.016463
```



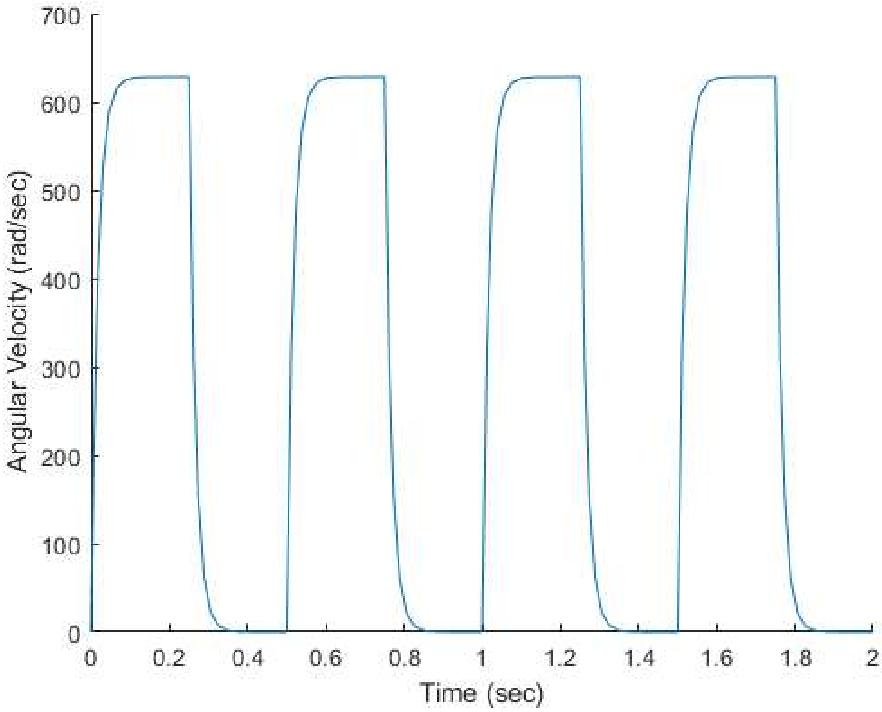
Published with MATLAB® R2020b

Activity 3

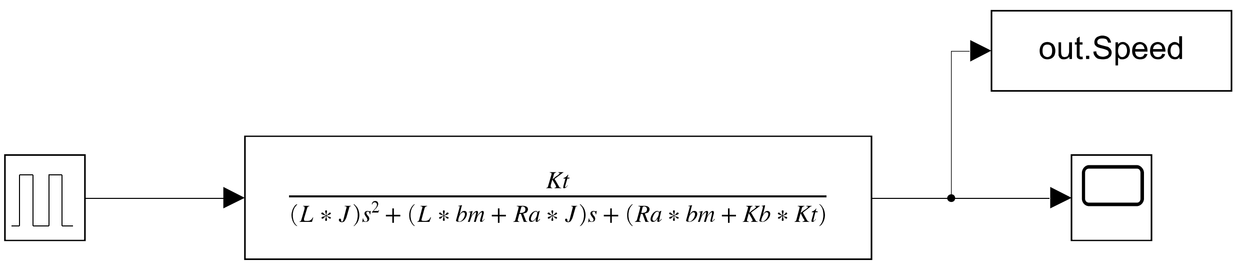
3/25/21 Michael White Section 3 / Online

```
close all;
clear all;
clc;

% Define parameters to be used
Kt = 7.68e-3; % Nm/A
Kb = 7.68e-3; % V/(rad/sec)
L = 0; % H
J = 3.9e-7; % kg*m^2
bm = 8.148e-7; % Nm/(rad/sec)
Ra = 2.6; % Ohm

% Run the simulation
simout = sim('DC_Motor_Sim_Act3');

% Generate figure and plot results
figure;
hold on;
plot(simout.Speed);
xlabel('Time (sec)');
ylabel('Angular Velocity (rad/sec)');

% Isolating the data to the first curve (contained in first 0.2 seconds)
calculationData = simout.Speed.Data(simout.Speed.Time <= 0.2);
calculationTime = simout.Speed.Time(simout.Speed.Time <= 0.2);

% Finding the time constant using the 63.2% method
maxValue = max(calculationData);
timeConstantValue = maxValue*0.632;
absDiffList = abs(calculationData-timeConstantValue);
timeConstantPoint = ...
    [calculationTime(absDiffList == min(absDiffList)),...
     calculationData(absDiffList == min(absDiffList))];

% Compare this time constant to Activity 2
timeConstant_Act2 = 0.016463;
percentDiff = 100*(abs(timeConstantPoint(1)-timeConstant_Act2)/mean([timeConstant_Act2,timeConstantPoint(1)]));
disp(strcat("The percent difference in time constants between Activity 2 and Activity 3 is ",num2str(percentDiff),"%"));
```

```
The percent difference in time constants between Activity 2 and Activity 3 is 4.184%
```



Angular Velocity (rad/sec)
Time (sec)
0
100
200
300
400
500
600
700
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2



$\frac{Kt}{(L*J)s^2+(L*bm+Ra*J)s+(Ra*bm+Kb*Kt)}$
out.Speed

White, Michael | Lab #2, Activity #4 | ME 421 L

① $\omega_0 = 0 \quad t_0 = 0 \rightarrow \omega_0 = C_2 e^{-t/\tau} + C_1$

$0 = C_2 e^0 + C_1 \rightarrow C_1 = -C_2 \rightarrow \omega_0 = -C_1 e^{-t/\tau} + C_1$

so... $\omega_0 = C_1(1 - e^{-t/\tau})$

$J_{eq}(\dot{\omega}_m) + b_{eq}(\omega_m) = \frac{K_t}{R_a} V_a(s)$

↓ Laplace transform

$J_{eq}(sW(s) - W(0)) + b_{eq} W(s) = \frac{K_t}{R_a} V_a(s) \rightarrow V_a(s) = \frac{1}{s}$ (step)

$W(s) = \frac{\frac{K_t}{R_a}}{J_{eq}s + b_{eq}} \cdot \frac{1}{s} = \frac{A}{s} + \frac{B}{J_{eq}s + b_{eq}}$

$\frac{K_t}{R_a} = A_{b_{eq}} \rightarrow A = \frac{K_t}{R_A b_{eq}}$; $B = -\frac{K_t J_{eq}}{R_A b_{eq}}$

$\frac{\frac{K_t}{R_A b_{eq}}}{s} - \frac{\frac{K_t J_{eq}}{R_i b_{eq}}}{J_{eq}s + b_{eq}} \rightarrow \frac{K_t}{R_a b_{eq}}\left(\frac{1}{s + \frac{b_{eq}}{J_{eq}}}\right)$

Laplace Inverse:

$\frac{K_t}{R_a b_{eq}} - \frac{K_t}{R_a b_{eq}} e^{-t\frac{b_{eq}}{J_{eq}}} \rightarrow$ Max Velocity $= \frac{K_t}{R_a b_{eq}} = C_1$; $\tau = \frac{J_{eq}}{b_{eq}}$

$(1 - e^{-t/\tau})\left(\frac{K_t}{R_a b_{eq}}\right) = \omega$ @ 63.2% where $t \approx \tau$

$b_{eq} = (1 - e^{-1})\left(\frac{K_t}{R_a b_{eq}}\right)$

Scanned with CamScanner

Activity 4

3/25/21 Michael White Section 3 / Online

```
close all;
clear all;
clc;

% Define parameters to be used
Kt = 7.68e-3; % Nm/A
Kb = 7.68e-3; % V/(rad/sec)
L = 0.18e-3; % H
J = 3.9e-7; % kg*m^2
bm = 8.148e-7; % Nm/(rad/sec)
Ra = 2.6; % Ohm
Va = 4; % V

% Import data from other activities
w = 397.1749;
timeConstant_Act2 = 0.016463;
timeConstant_Act3 = 0.0177;

% Calculate beq from derived equation
beq = (1-exp(-1))*(Kt/(Ra*w));

% Calculate jeqs using time constant - beq relationship
jeq2 = timeConstant_Act2*beq;
jeq3 = timeConstant_Act3*beq;

% Display results
disp(strcat("The value of Beq is ",num2str(beq)));
disp(strcat("The value of Jeq for Activity 2 is ",num2str(jeq2)));
disp(strcat("The value of Jeq for Activity 3 is ",num2str(jeq3)));
```

```
The value of Beq is 4.7012e-06
The value of Jeq for Activity 2 is 7.7395e-08
The value of Jeq for Activity 3 is 8.3211e-08
```

Published with MATLAB® R2020b

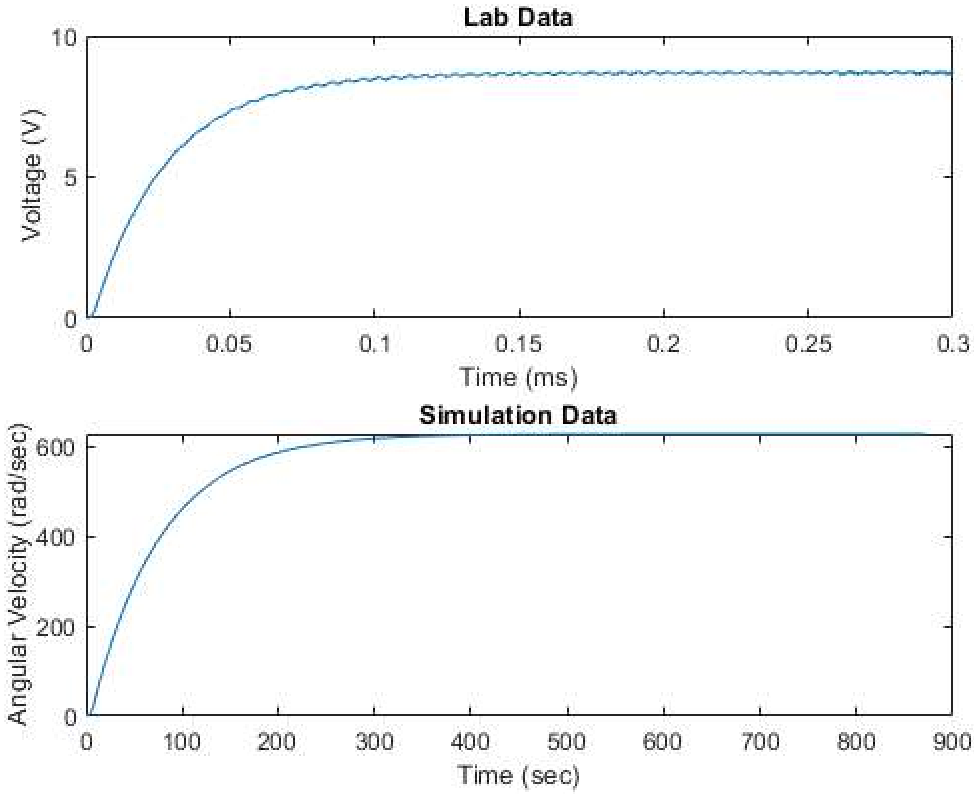
Activity 5

3/25/21 Michael White Section 3 / Online

```
close all;
clear all;
clc;

% I created a simplified excel table that includes the times and averaged
% values as two, simple columns, and import it here with xlsread.
data = xlsread('SimplifiedData.xlsx');
time = data(:,1); voltage = data(:,2);

% Set parameters for simulation
Kt = 7.68e-3; % Nm/A
Kb = 7.68e-3; % V/(rad/sec)
L = 0.18e-3; % H
J = 3.9e-7; % kg*m^2
bm = 8.148e-7; % Nm/(rad/sec)
Ra = 2.6; % Ohm

% Run simulation file from Activity 2
simout = sim('DC_Motor_Simulation_Act2');

% Isolating the data to the first curve (contained in first 0.2 seconds)
calculationData = voltage(time <= 0.2);
calculationTime = time(time <= 0.2);

% Finding the time constant using the 63.2% method
maxValue = max(calculationData);
timeConstantValue = maxValue*0.632;
absDiffList = abs(calculationData-timeConstantValue);
timeConstantPoint = ...
    [calculationTime(absDiffList == min(absDiffList)),...
     calculationData(absDiffList == min(absDiffList))];

% Define parameters to be used
Kt = 7.68e-3; % Nm/A
Kb = 7.68e-3; % V/(rad/sec)
L = 0.18e-3; % H
J = 3.9e-7; % kg*m^2
bm = 8.148e-7; % Nm/(rad/sec)
Ra = 2.6; % Ohm
Va = 4; % V

% Import data from other activities
w = 397.1749;
timeConstant_Act2 = 0.016463;
timeConstant_Act3 = 0.0177;

% Calculate beq from derived equation
beq = (1-exp(-1))*(Kt/(Ra*w));

% Calculate jeqs using time constant - beq relationship
jeq2 = timeConstant_Act2*beq;
jeq3 = timeConstant_Act3*beq;
jeqLab = timeConstantPoint(1)*beq;
```

```
% Display results
disp(strcat("The value of Beq is ",num2str(beq)));
disp(strcat("The value of Jeq for Activity 2 is ",num2str(jeq2)));
disp(strcat("The value of Jeq for Activity 3 is ",num2str(jeq3)));
disp(strcat("The value of Jeq from the lab data is ",num2str(jeqLab)));

% Graph voltage data from lab
subplot(2,1,1);
plot(time, voltage);
xlabel('Time (ms)');
ylabel('Voltage (V)');
title('Lab Data');

% Graph data from simulations
subplot(2,1,2);
plot(simout.Speed.Data(simout.Speed.Time <= 0.2));
xlabel('Time (sec)');
ylabel('Angular Velocity (rad/sec)');
title('Simulation Data');

% WRITTEN RESPONSES:
% The simulated plot looks similar in shape to the experimental data,
% though obviously there is some difference mathematically due to the
% difference in time constants.

% The calculated values are clearly different from the lab results. This
% arises obviously from the difference in time constants. The time
% constant from the lab is obviously greater than the calculated values.
% This makes since as the lab setup is not going to be ideal and will
% ultimately operate slower for potentially many different reasons. The
% torque on the motor could have been greater than calculated. The friction
% on the motor could have affected this, or the weight of the rod. Many
% things could have affected this outcome.
```

```
The value of Beq is 4.7012e-06
The value of Jeq for Activity 2 is 7.7395e-08
The value of Jeq for Activity 3 is 8.3211e-08
The value of Jeq from the lab data is 1.3398e-07
```



Published with MATLAB® R2020b