

LEVEL 5 ASSESSMENT SPECIFICATION

Student name:	Paul Miller	Student ID number:	
Programme:	HND/HNC Electrical/Electronic Engineering, HNC Instrumentation Engineering & HNC Power System Engineering		
Module:	Electrical Principles, Methods and Simulation		
Module code:	PSDE5009	Contribution to Overall Module Assessment (%):	50%
Lecturer:	Jason Pointer	Internal Verifier:	Steve Roberts
Assignment Title:	Control Systems	Word count (or equivalent):	2000
Submission deadline:	16/05/19	Return date of provisional marks & written feedback:	
Submission method:	All written assessments, where practical and possible, must be submitted via Turnitin unless otherwise instructed by the Lecturer. (Please DO NOT put this assessment specification into Turnitin or it will match many similarities with other students' submissions.) Alternative submission method (if applicable): Late submission of the assessment will result in a late penalty mark. Penalties for late submission: Up to one week late, maximum mark of 40%. Over one week late, 0%. Only the Extenuating Circumstances Panel may approve a change to submission dates.		
Academic honesty / referencing:	Academic honesty is required. In the main body of your submission you must give credit to authors on whose research and ideas your work is based. Append to your submission a reference list that indicates the books, articles, etc. that you have used, cited or quoted in order to complete this assessment.		

Module Learning Outcomes (from module syllabus)

1. Analyse and evaluate electrical circuits, electronic systems and **control systems**.
2. Simulate standard electrical circuits, electronic systems and **control systems**.

STUDENT FEEDBACK FORM

This section details the *extent* to which the assessment criteria are demonstrated by you, which in turn determines your mark. The marks available for each category of skill are shown. Lecturers will use the space provided to comment on the achievement of the task(s), including those areas in which you have performed well and areas that would benefit from development/improvement.

Generic Assessment Criteria	Marks available	Marks awarded
1. Engagement with Literature Skills	0	0
2. Knowledge and Understanding Skills	30	16.4
Achieved 45% for Task 1. Achieved 70% for Task 2. Achieved 38% for Task 3. Achieved 70% for Task 4. Achieved 50% for Task 5. Knowledge is reasonably detailed and accurate in most parts. Show a good understanding of some concepts and principles covered by this assignment but not all of the elements were covered for some of the task.		
3. Cognitive and Intellectual Skills	10	3
Largely descriptive work in the analysis of the test results.		
4. Practical Application Skills	50	28.7
Achieved 50% for Task 1. Achieved 75% for Task 2. Achieved 42% for Task 3. Achieved 70% for Task 4. Achieved 50% for Task 5. A good and appropriate application of established methods in the use of Matlab, Multisim and Simulink but in places not all of the required work was produced.		
5. Transferable Skills for Life and Professional Practice	10	5
Mostly coherent, organised and accurate work, in a suitable structure and is for the most part clearly expressed.		

Assessment Mark (Assessment marks are subject to ratification at the Exam Board. These comments and marks are to give feedback on module work and are for guidance only until they are confirmed.)

Late Submission Penalties (tick if appropriate)

Up to 1 week late (40% Max)

Over 1 week late (0%)

53%

Jason Pointer

Task 1 – RC Circuit First Order System

Figure 1 shows a RC circuit that can be analysis using time and frequency response.

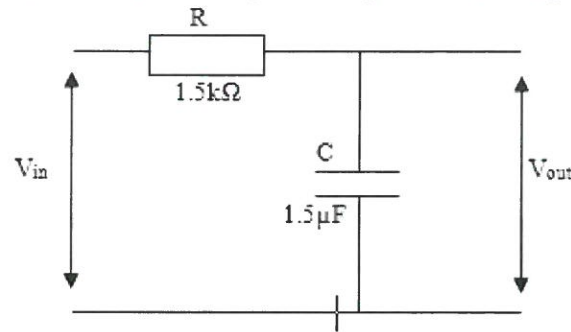


Figure 1

- A) Determine the transfer function for the circuit and using this produce the time response equation.

Using Kirchhoff's Voltage law

$$Ri(t) + \frac{1}{C} \int i(t) dt = V_{in}(t)$$

Use the first derivatives

$$R \frac{di(t)}{dt} + \frac{1}{C} i(t) = \frac{dV_{in}(t)}{dt}$$

Using Laplace transform

$$sRI(s) + \frac{I(s)}{C} = sV_{in}(s)$$

$$I(s) \left[R + \frac{1}{Cs} \right] = V_{in}(s)$$

$$V_{in}(s) = I(s) \left[R + \frac{1}{Cs} \right] \quad (1)$$

And

$$V_{out} = \frac{1}{C} \int i(t) dt$$

$$\frac{dV_{out}}{dt} = \frac{i(t)}{C}$$

Using Laplace transform

$$sV_{out}(s) = \frac{I(s)}{C}$$

$$I(s) = sV_{out}(s) \times C$$

Sub into (1)

$$V_{in}(s) = sV_{out}(s) \times C \left[R + \frac{1}{Cs} \right]$$

$$\frac{V_{in}(s)}{V_{out}(s)} = sC \left[R + \frac{1}{Cs} \right] = CRs + 1$$

$$\frac{V_{out}(s)}{V_{in}(s)} = \frac{1}{CRs + 1}$$

$$\frac{V_{out}(s)}{V_{in}(s)} = \frac{1/CR}{s + 1/CR}$$

Take inverse Laplace transforms:

$$\frac{v_{out}(t)}{v_{in}(t)} = \frac{1}{CR} \times e^{-\frac{t}{CR}}$$

$$v_{out}(t) = v_{in}(t) \times \frac{1}{CR} e^{-\frac{t}{CR}}$$

B) Use Matlab to produce the time response plot when 20V dc voltage is applied at Vin and also to calculate the time constant, delay time and rise time. Confirm the plot by using Simulink.

Figure 1 illustrates the time response, this was produced by writing a code which has the calculations and values entered, and the graph is then produced by using the 'plot' command.

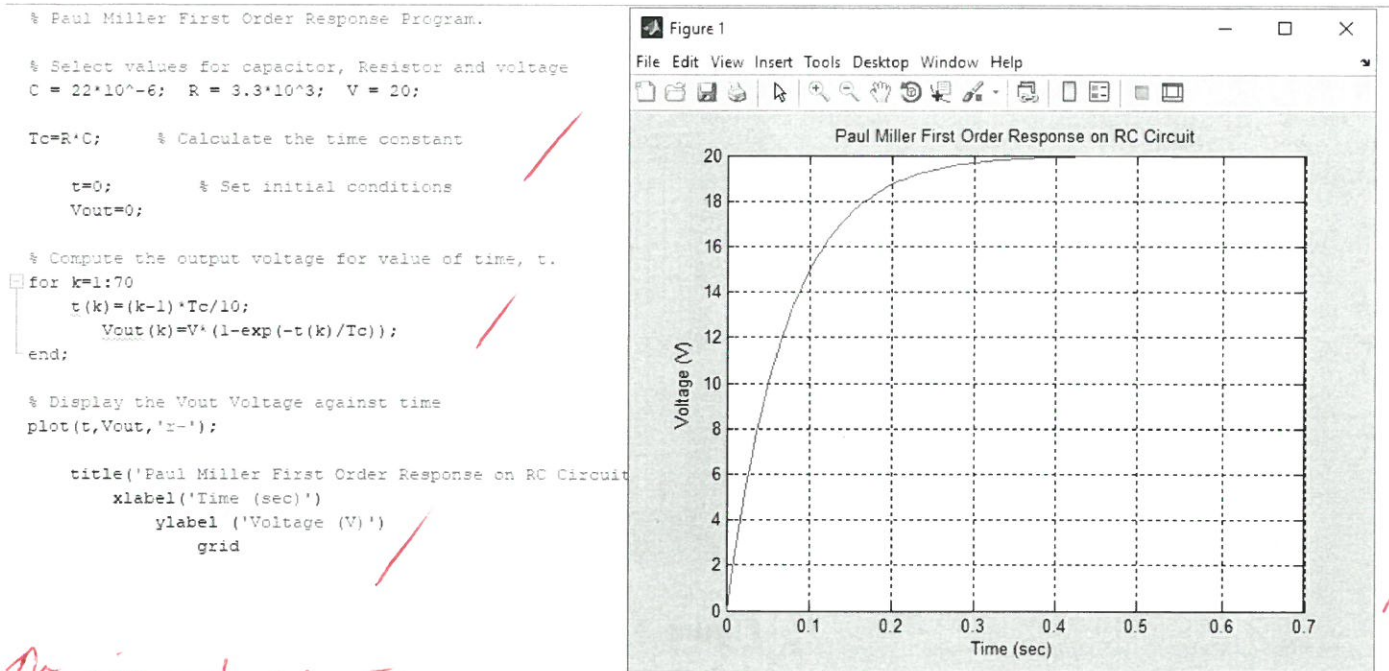


Figure 1

Figure 2 illustrates the Simulink model which was produced to confirm the results of the program above, as **Figure 2** shows, both graphs are the same which confirms the results.

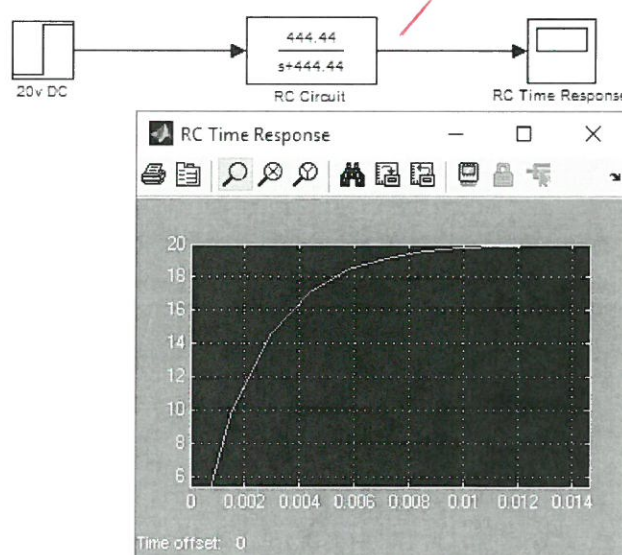


Figure 2

C) If a sine wave is now applied to the input the frequency response of the circuit can be determined. Use a 1V peak to peak sine wave and Matlab to produce the frequency response for the magnitude also determine the cut-off frequency.

Figure 3 illustrates the code which was produced to find the frequency response and shows the cut off point for this Circuit, the cut off point for this circuit is **0.77**, and **Figure 4** shows this.



Figure 3

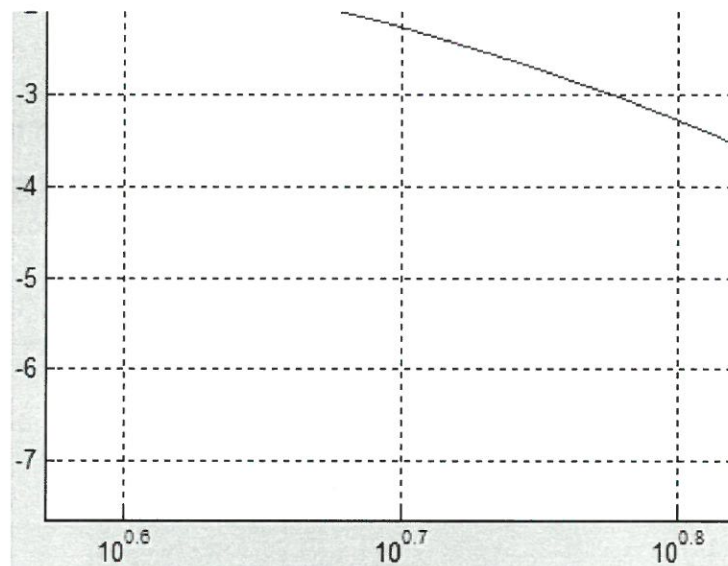


Figure 4

I assume this is to show the cut off frequency.

- D) The frequency response can also be represented using a Bode plot which will include magnitude and phase plots. Use Matlab to produce a Bode plot and confirm the results using Multisim.

Figure 5 illustrates the graphs which was produced using multisim, the graphs produced show a bode plot, which includes the magnitude and phase plots.



Where is the code to produce the Bode plot.

← ?

Figure 5

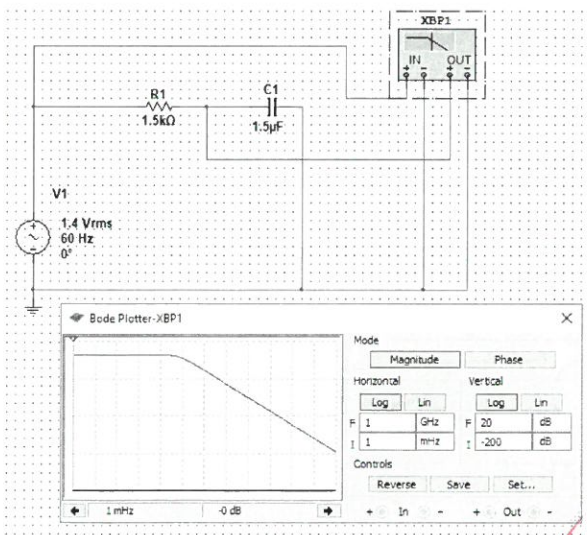


Figure 6

Could have measured the cut-off.

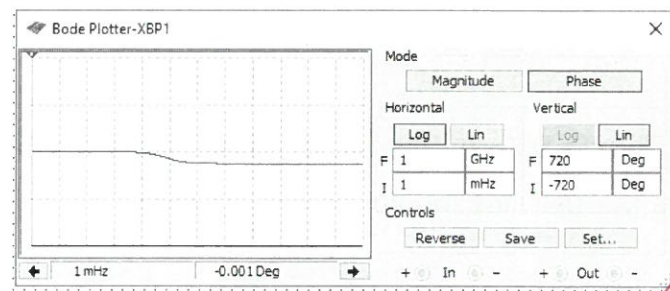


Figure 7

Figures 6 and 7 illustrates the results being confirmed on multisim, it shows the circuit which was built to test, it also shows the magnitude and the phase.

Task 2 - DC Motor Model

A) Using Matlab write an m-file to find the motor constant K_M and to plot the speed-torque

Figure 7 illustrates the code which was produced to find the constant K_M , **Figure 8** illustrates the graph which was produced using the code.

```
%Paul Miller
%Task 2 Ass 2

% DC Motor Programme
Ra=1.4; Va=12; Ts=0.21453; % Values for motor

km=(Ts*Ra)/Va; % Motor Constant Calculation

% Max Speed
W0=Va/km;

% Speed Range
W=0:2:2000;
T=km*(Va-km*W)/Ra;

% Show Results
disp('Motor Constant Equals, km')
disp(km)
plot(T,W,'b-')
xlabel('Torque Equals (Nm)')
ylabel('Speed Equals (Rads/s)')
title('Speed and Torque Comparison')
```

Figure 7

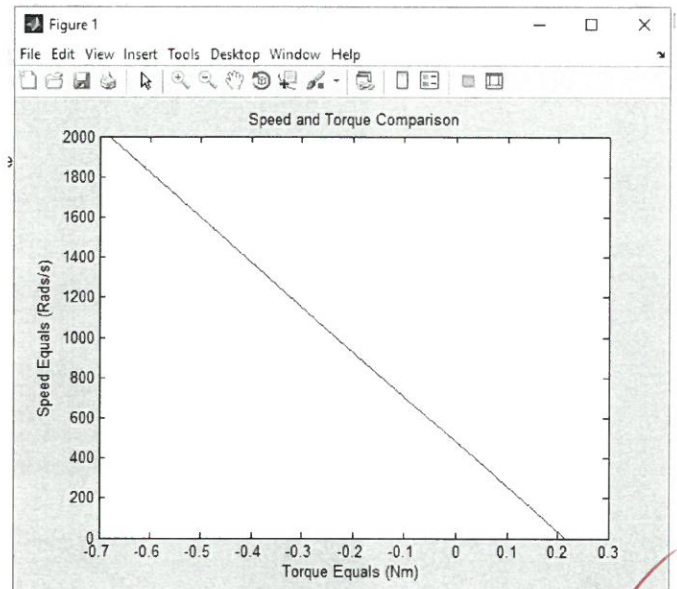


Figure 8

B) Build a Simulink model for the armature-voltage controlled DC motor. The motor parameters are $R_a = 1.5\Omega$, $L_a = 0.14H$, $K_m = 0.03Nm/A$, $J = 0.0004kgm^2$, $b = 3.4 \times 10^{-6}Nms/rad$ and $V_a = 6V$.

Figure 9 illustrates the Simulink model which was produced for the armature voltage, the graph illustrates the speed against time.

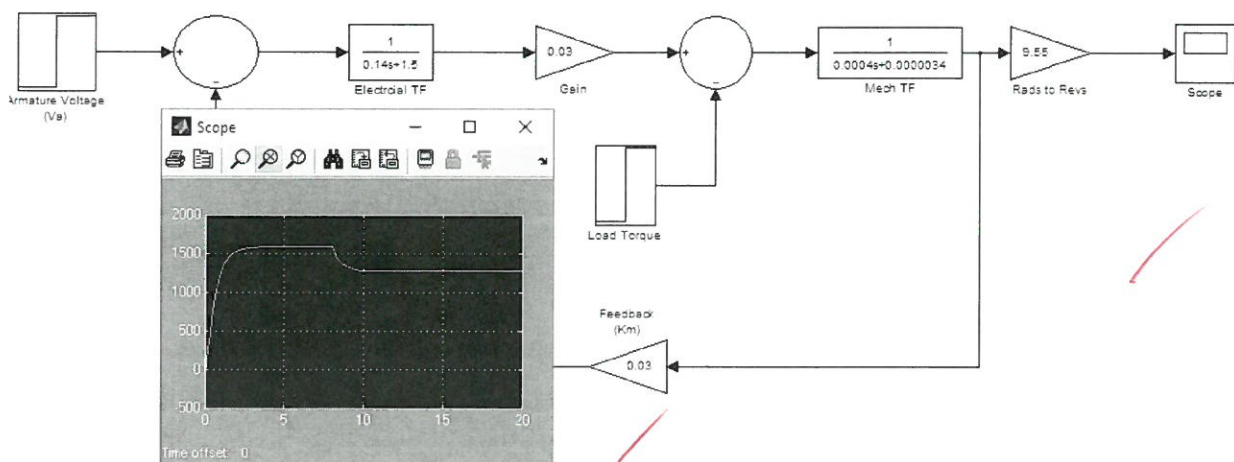


Figure 9

*↑
You have covered up part of the model.*

- C) Run the simulation model and plot the speed response (in rpm) for the first 16 seconds when the load of a DC motor changes between 0.02Nm and 0.04Nm with frequency 0.125Hz.

Figure 10 illustrates the speed response for the first 16 seconds, the graph shows the motor overcoming the initial friction and then settling into a steady state.

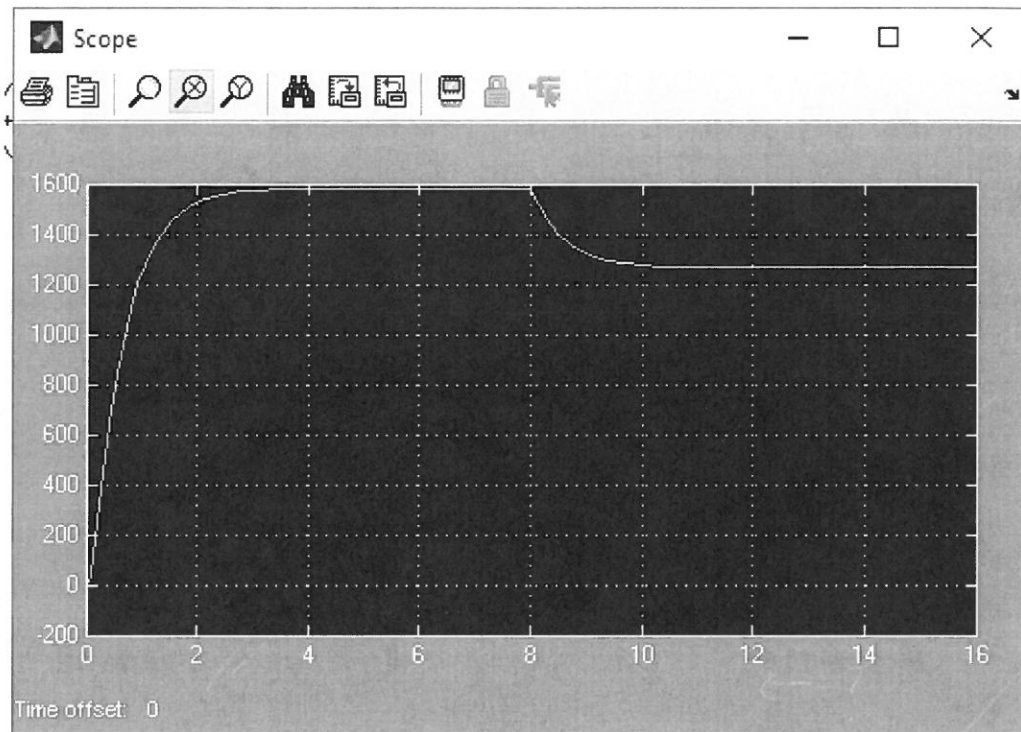


Figure 10

D) Find out the steady state angular speeds (in rpm) at load 0.02Nm and 0.04Nm, respectively.

Figure 11 illustrates the motor set at 0.02Nm, the steady state for this is approximately 1600 rpm, **Figure 12** illustrates the motor set at 0.04Nm, and the steady state for this is approximately 1300 rpm.

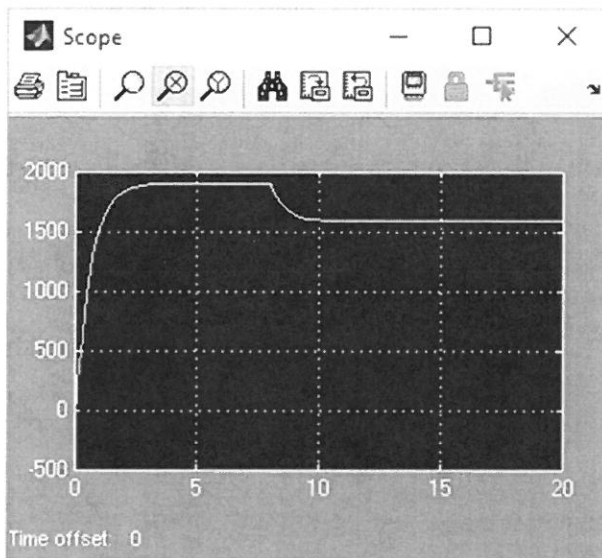


Figure 11

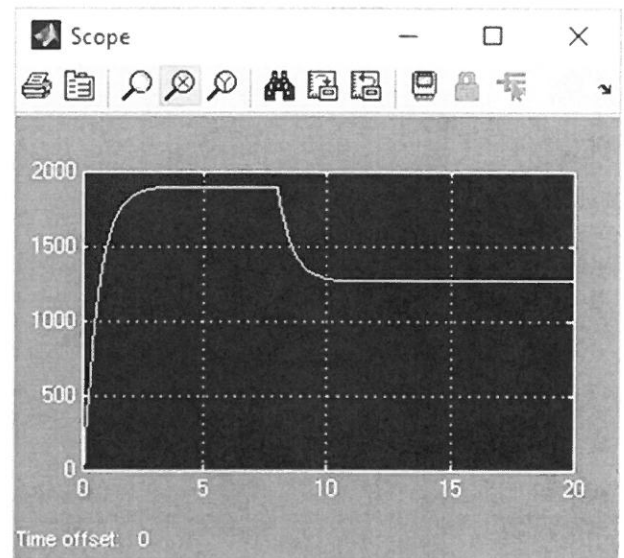


Figure 12

Task 3 - Position Control Servo

A) Sketch a block diagram for the system.

Figure 13 illustrates a simple block diagram for the circuit.

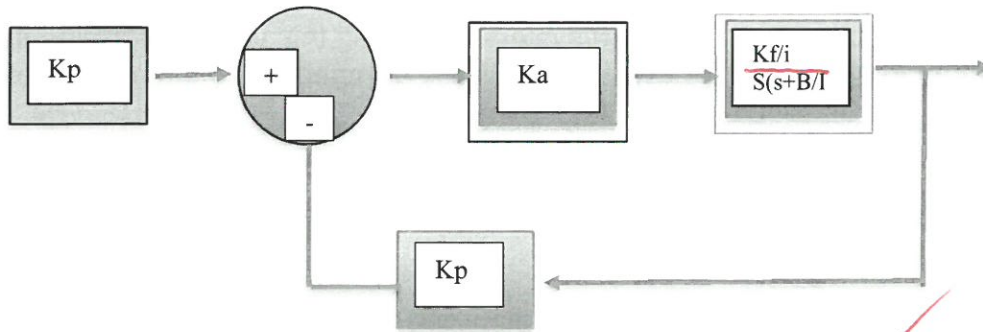


Figure 13

B) From your block diagram show that the transfer function for the system is:

$$\frac{\theta_o(s)}{\theta_i(s)} = \frac{K_A K_P K_f / I}{s^2 + \frac{sB}{I} + \frac{K_A K_P K_f}{I}}$$

The transfer functions for negative feedback system.

$$\frac{\theta_o}{\theta_i} = \frac{G(s)}{1 + G(s)H(s)}$$

$$G(s) = K_A \times \frac{K_f / I}{s(s + B/I)}$$

$$H(s) = K_P$$

The overall Transfer function

$$\frac{\theta_o(s)}{\theta_i(s)} = \frac{K_P \times \frac{K_A K_F / I}{s(s + B/I)}}{1 + K_P \times \frac{K_A K_F / I}{s(s + B/I)}}$$

$$\frac{\theta_o(s)}{\theta_i(s)} = \frac{\frac{K_P K_A K_F / I}{s^2 + Bs/I}}{1 + \frac{K_P K_A K_F / I}{s^2 + Bs/I}}$$

So

$$\frac{\theta_o(s)}{\theta_i(s)} = \frac{\frac{K_P K_A K_F}{I}}{s^2 + \frac{Bs}{I} + \frac{K_P K_A K_F}{I}}$$

C) A unit step input is to be applied to the position control servomechanism and the following parameters are applied:

- $K_A = 1.0$
- $K_P = 5Vr-1$
- $K_f = 8mAV-1$
- $I = 2 \times 10^{-4} Kgm^2$

Find the value for B to create a critically damped system and using this value determine the poles of the system. Use Matlab to show that the system is critically damped by obtaining the pole plot and also to show stability.

Figure 14 illustrates the formula which was derived to find the value of B

$$\frac{\theta_o(s)}{\theta_i(s)} = \frac{\frac{K_p K_A K_F}{I}}{s^2 + \frac{Bs}{I} + \frac{K_p K_A K_F}{I}}$$

$$\frac{\theta_o(s)}{\theta_i(s)} = k_{ss} \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

$$\frac{\theta_o(s)}{\theta_i(s)} = \frac{\frac{5 \times 1 \times 8 \times 10^{-3}}{2 \times 10^{-4}}}{s^2 + \frac{Bs}{2 \times 10^{-4}} + \frac{5 \times 1 \times 8 \times 10^{-3}}{2 \times 10^{-4}}} = \frac{200}{s^2 + 5000Bs + 200}$$

$$\omega_n^2 = 200$$

So Steady state value = $k_{ss} = 1$

Natural frequency = $\omega_n = \sqrt{200} = 14.14 \text{ rad/sec}$

$$2\zeta\omega_n = 5000B$$

For critically damping $\zeta = 1$

So $2 \times 1 \times 14.14 = 5000B$

$$B = \frac{28.28}{5000} = 5.656 \times 10^{-3} \text{ N/ms}^{-1}$$

To determine the poles for critically damped

$$(s^2 + 2\omega_n s + \omega_n^2) = 0$$

$$s = -\omega_n, -\omega_n = -14.14$$

Figure 14

Figure 15 illustrates the code which was written in matlab to produce a graph which shows the pole plot and stability of the circuit.

```
% Create the numerator
num=[125];

% Create the denominator
den=[1 22.36 125];

% Transfer Function
sys=tf(num,den)

% Poles
pole(sys)

% Plotting the Poles
subplot(211), pzmap(sys)
axis([-50 1 -15 15])

% Plotting the response for an unit step
subplot(212), step(sys)
```

Figure 15

Figure 16 illustrates the graph which was produced, it shows the system as being critically damped as it has no oscillation with the fastest response.

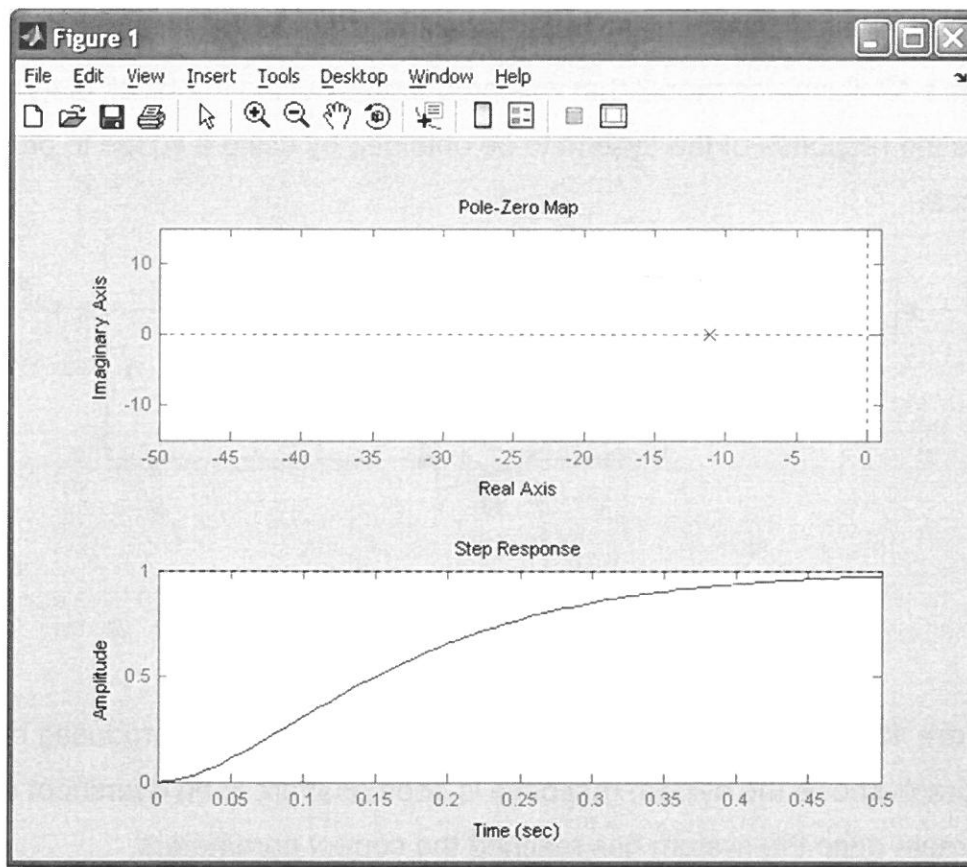


Figure 16

D) Obtain the time response of the system using Matlab and compare its performance by modelling it in Simulink.

Figure 17 illustrates model that was produced by using the block diagram in **part A**, this model allows the response of the system to be obtained by using a scope to produce a graph of the performance.

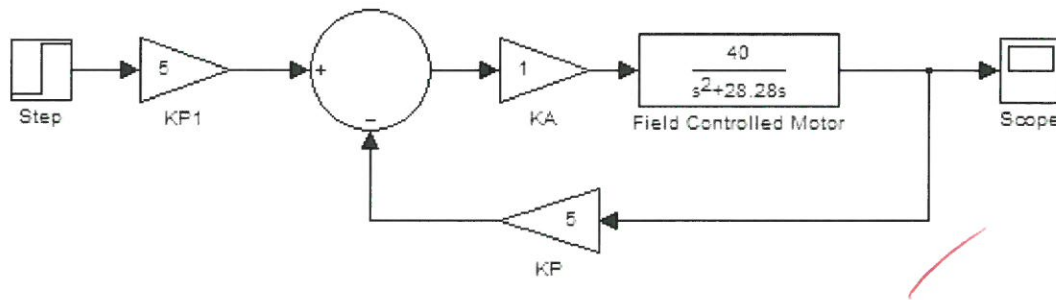


Figure 17

Figure 18 illustrates the response of the system which was produced by the scope in the model, this graph shows the system response is good as there is no overshoot and shows a steady power stream once the system has reached the correct parameters.

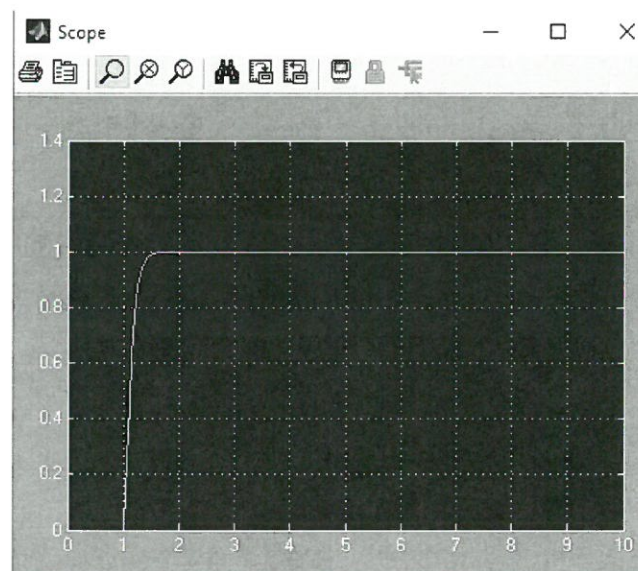


Figure 18

Should have started at zero, you have not changed the default value for the step.

You have not completed the task.

Task 4

An electrical system modelled by

$$G(s) = \frac{197.54}{s^2 - 2.05s + 3.96}$$

Using the trial and error methods, find out parameters K_p , K_i and K_d of the PID controller in Simulink so that the closed-loop system can achieve overshoot $\sigma = 0\%$, rise time t_r less than 0.18 second and zero steady-state error. Plot the unit step response of this closed-loop system for the first 1 second

Figure 19 illustrates the system which was produced in Simulink, it shows the system and how stable it is before adding PID

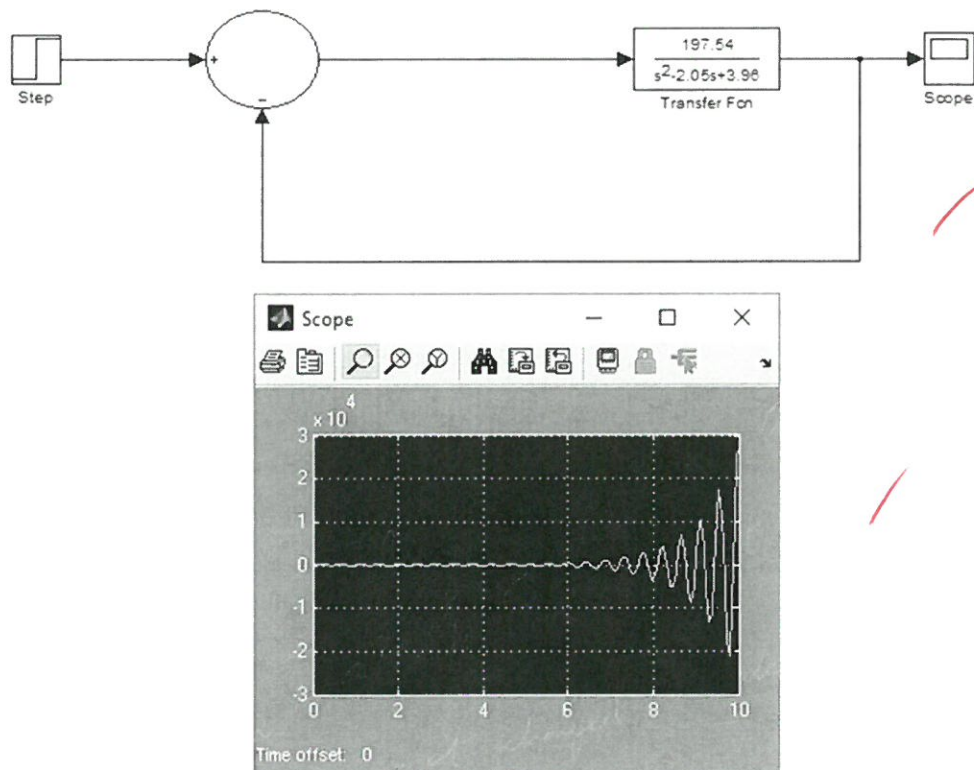


Figure 19

Figure 20 illustrates the system after adding PID but without making any changes to the values, it also shows the graph of how the system stability has changed.

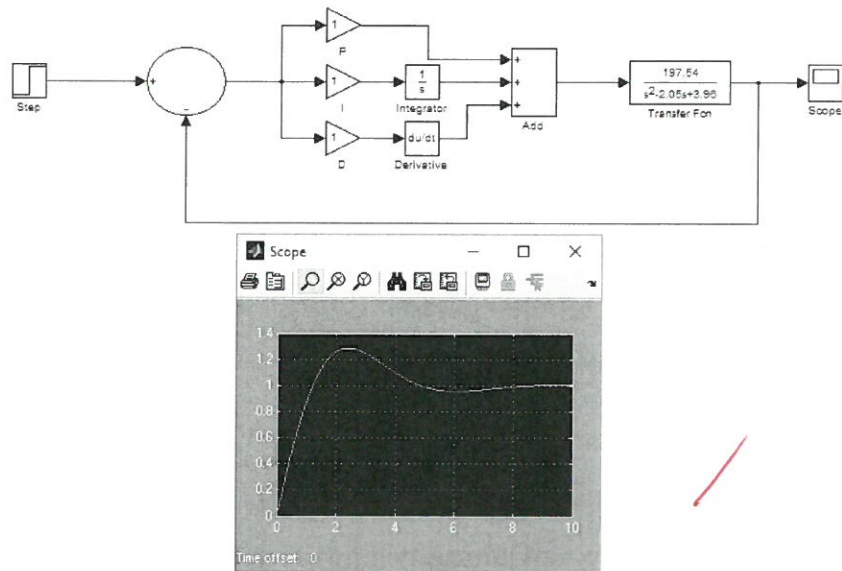


Figure 20

Figure 21 illustrates the system after adding PID and adding a higher value into the integrator of the PID system. This figure shows the integrator value is set at 5 and shows how unstable the system has become.

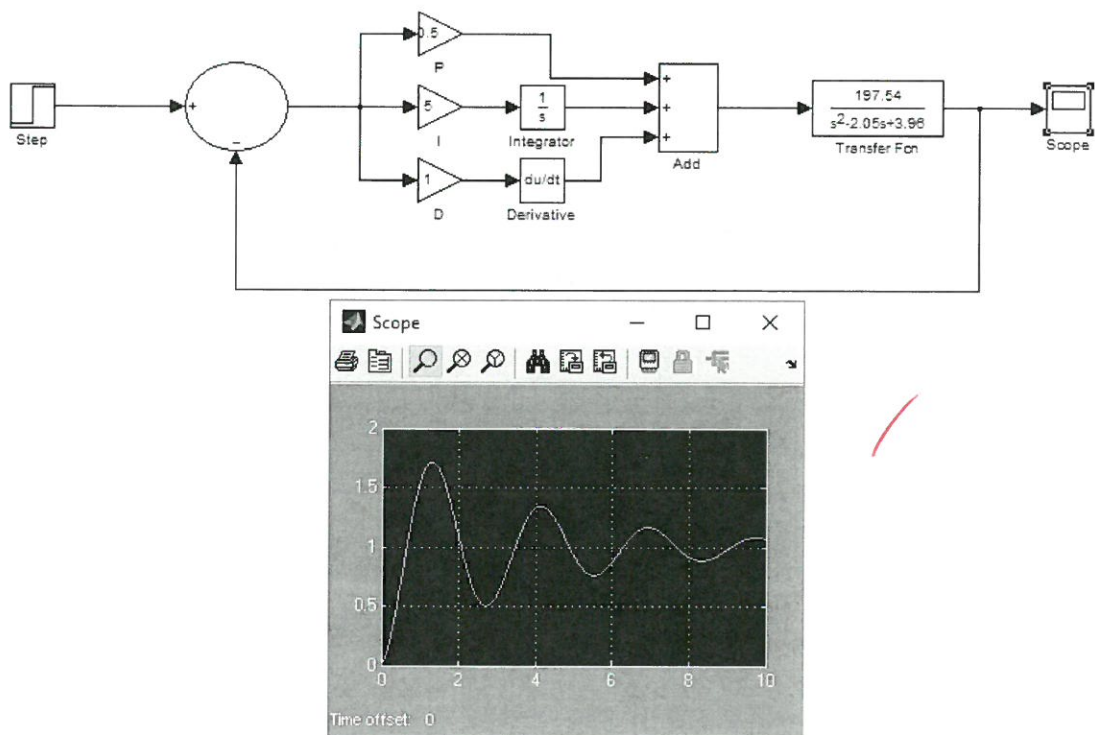


Figure 21

Figure 22 illustrates the graph of the system with the integrator value set at 0.5, this shows a more stable system as the curve produced is a lot more linear.

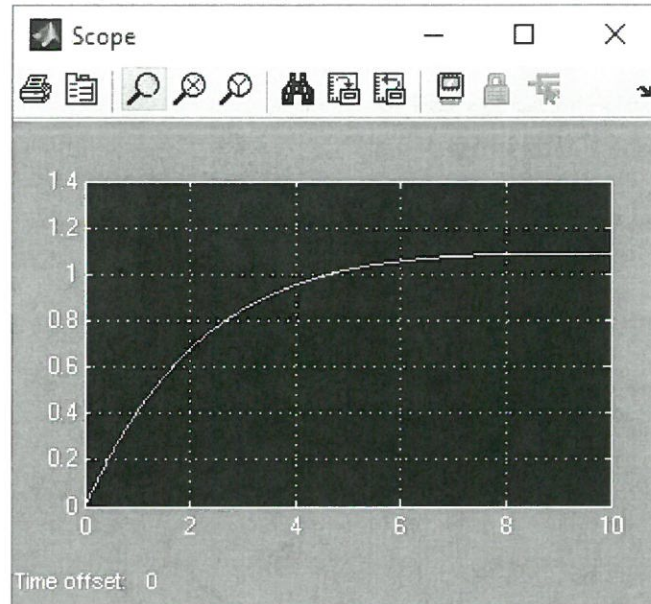


Figure 22

Figure 23 illustrates the system once all values on the PID system have been changed to make the system as stable as possible, this set up on the PID system meets the rudiments which are stated in **Q4**.

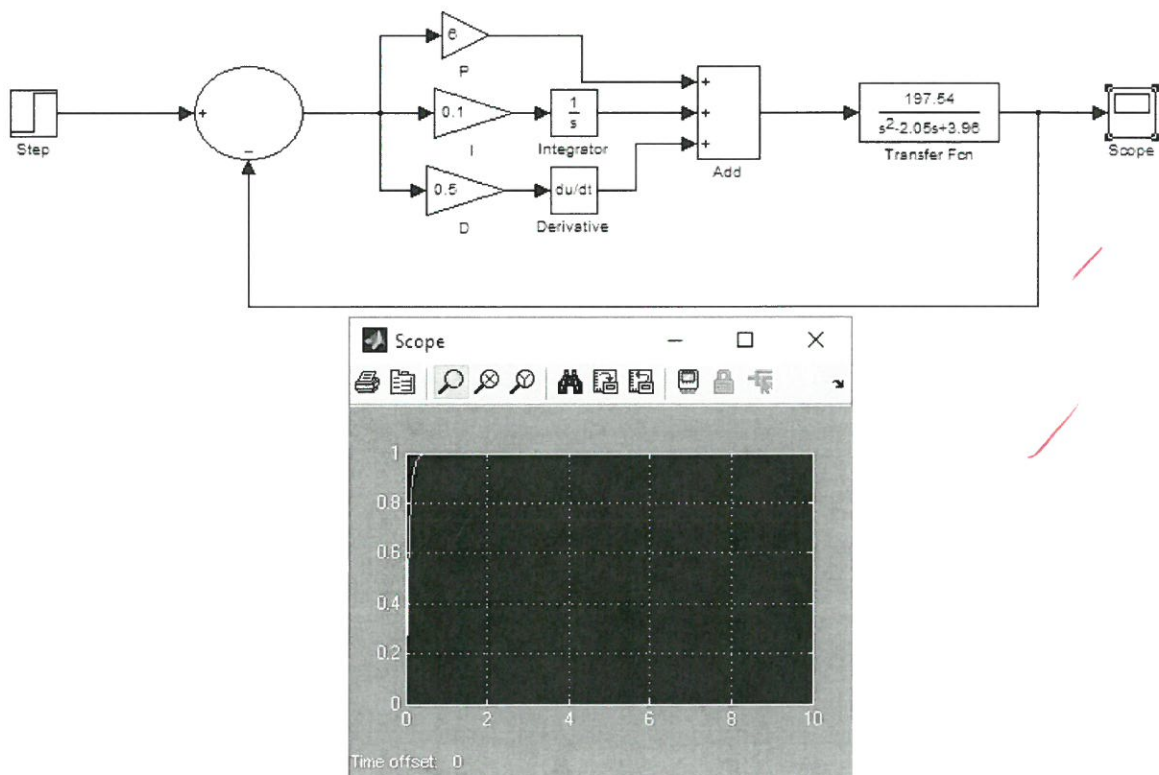


Figure 23

Task 5

- a. Write a Matlab file to plot the Nyquist diagram for the open loop transfer function of a servo control rig given below.

$$L(s) = \frac{0.03868s^3 - 5.369s^2 - 97.34s + 13450}{s^3 + 169.8s^2 + 1730s + 159}$$

Figure 24 illustrates the code which was produced to plot a Nyquist diagram for the open loop transfer function.

```
numerator=[0.03868 -5.369 -97.34 13450];  
denominator=[1 169.8 1730 159 0.003868 -5.369 -97.34 13450];  
  
RIG=tf([numerator],[denominator]); % Produce transfer function  
  
nyquist(RIG) % Produce the Nyquist response Graph
```

Figure 24

Figure 25 illustrates the Nyquist plot which was produced for the open loop transfer function.

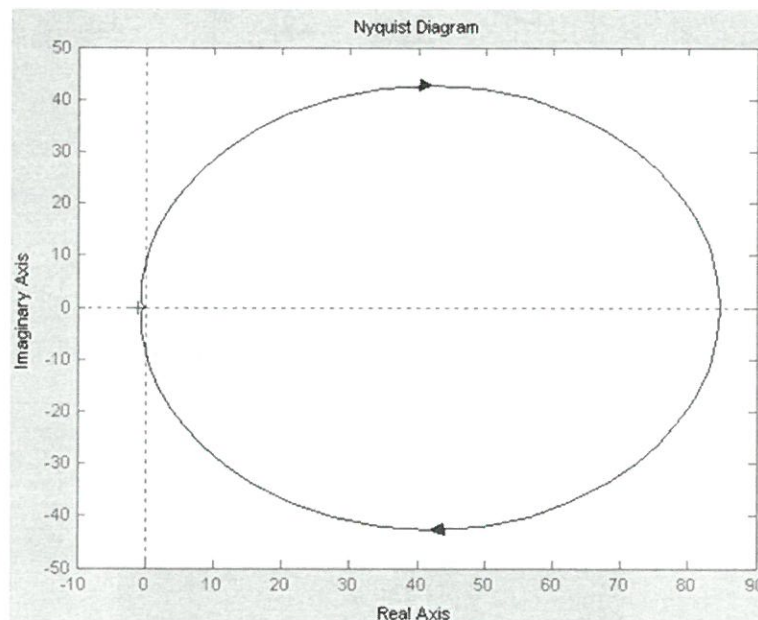


Figure 25

b. Is the closed-loop system $L(s)/(1+L(s))$ stable? If yes, find the stability margins.

Figure 26 illustrates the code which was produced for the closed loop system on the transfer function.

```
numerator=[0.03868 -5.369 -97.34 13450];  
denominator=[1.03868 164.431 1632.66 13609 ];  
  
RIG=tf([numerator],[denominator]); % Produce transfer function  
  
nyquist(RIG) % Produce the Nyquist response  
pause  
  
bode(RIG) % Produce the Bode plot  
grid
```

Figure 26

Figure 27 illustrates the Nyquist diagram and the bode plot which was produced for the closed loop system.

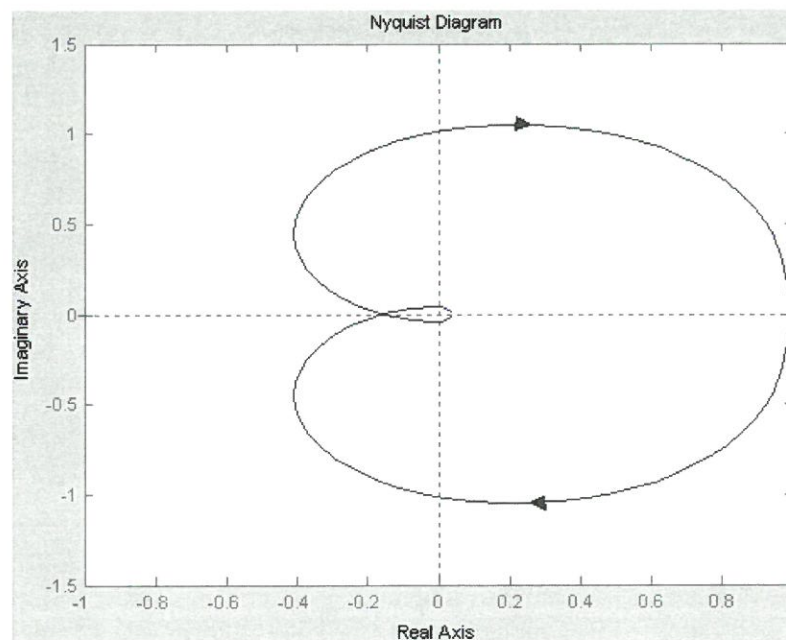


Figure 27

C) Plot the Bode plot of the closed-loop transfer function $G_c(s) = \frac{L(s)}{1+L(s)}$ and find the peak

Figure 28 illustrates the bode plot of the closed loop transfer function.

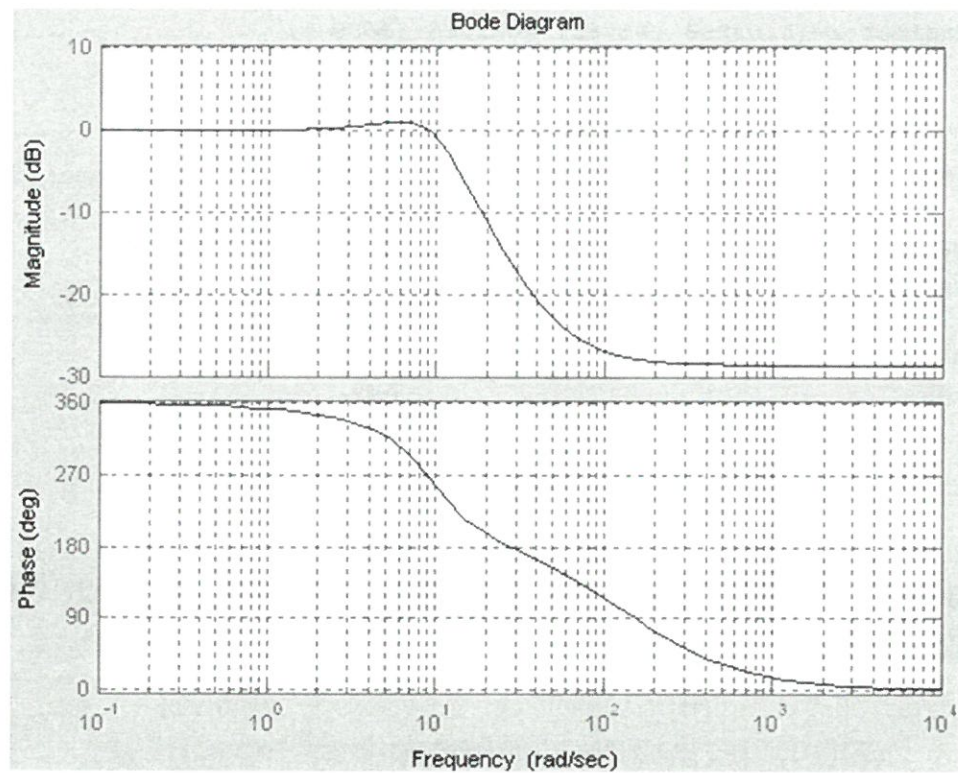


Figure 28

Was the system stable?

- D)** Build the system model with a PI controller gain and initial gains $K_i = 0.1$ and $K_p = 0.5$ in Simulink and then run the simulation. Plot the position response curve for the first 10 seconds if the reference position angle is a square wave with amplitude 40 degrees and frequency 0.2Hz. Manually tune the PI controller's gains in the Servo Control Rig so that the closed-loop system can achieve overshoot $\sigma < 5\%$ and settling time $t_s < 1.0\text{sec}$. Describe the tuning procedure in detail.

Figure 29 illustrates the PI controller which was produced on Simulink, the values for P and I were set at the initial values stated in the question, the graph which is down on **figure 28** shows the system is not within the tolerances given, so tuning will have to be done to the P and I to insure it is in tolerance.

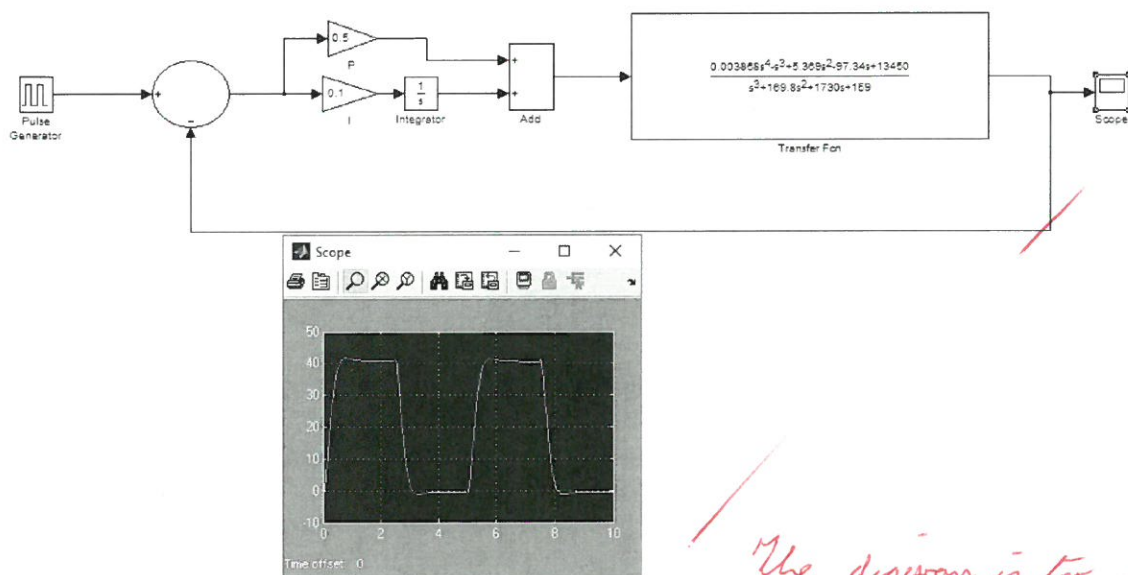


Figure 28

*The diagram is too small
does not clearly show
the improved response.*

Figure 29 illustrates the PI controller once the values had been changed to ensure it was within the tolerances given, the Proportional was first changed to remove oscillation and bring it down to a steady level. The Integrator was then changed to a value which reduces the settle in time, this allows it be within the 2 second time limit given. The P values is now set at 0.4, and the I value is now set at 0.02, as the graph on figure 29 shows, it is within all tolerances.

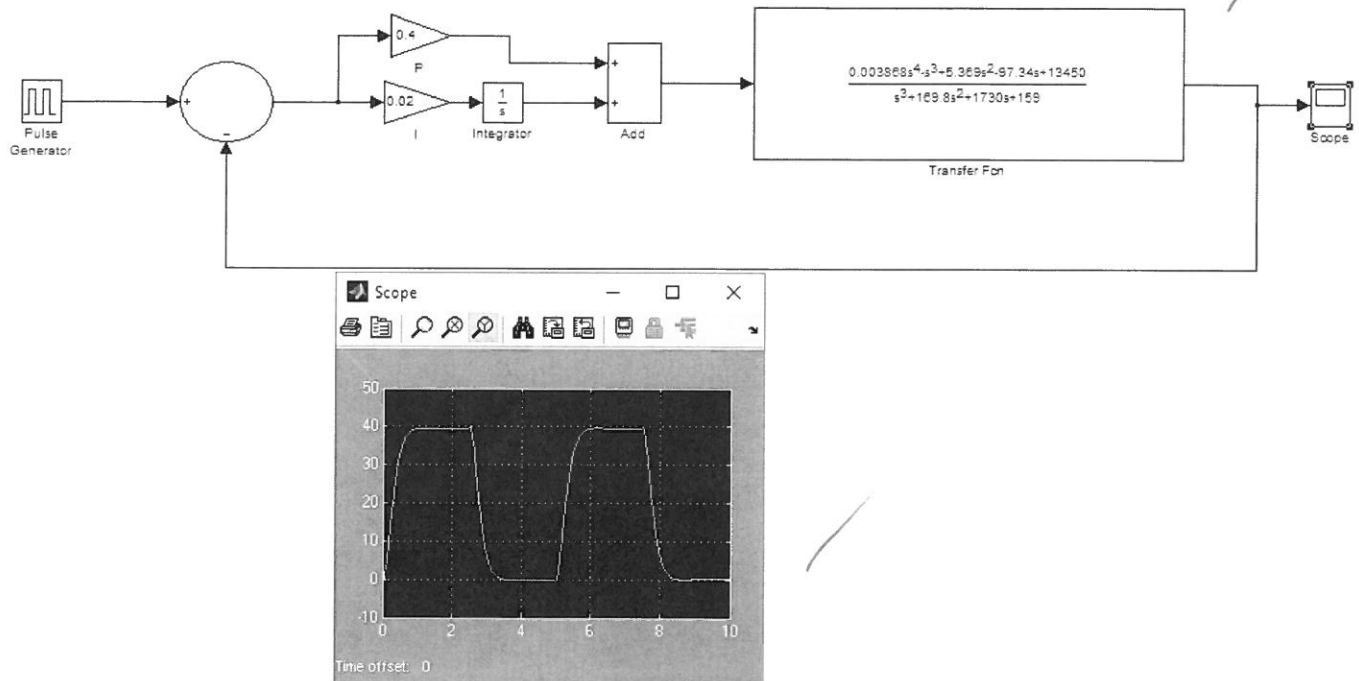


Figure 29