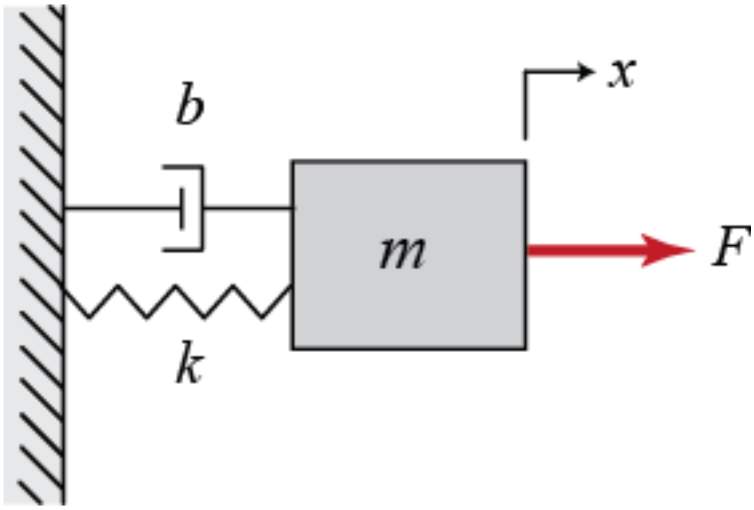
1. Analyse a simple mechanical system

- Consider the mass-spring-damper system shown below:



- Here we have a mass M connected to a fixed object by a spring with coefficient k and damper with coefficient b. The displacement from equilibrium position is given by distance x(t) and it can be moved by applying a force F(t) to the mass.
- Write down the differential equations of motion of the system.
- From your differential equation derive its Laplace transformation.
- Rearrange the equation for derived the transfer function of output position given input force F(s):

$$P(s) = X(s)/F(s)$$

- Use the following parameter values in the transfer function:

```
M = 1 kg
b = 4 N s/m
k = 5 N/m
F = 1 N
```

2. Using initial and final value theorem

- Use the Laplace initial and final value theorems to calculate:
    - The initial response to a step input
    - The final response to a step input

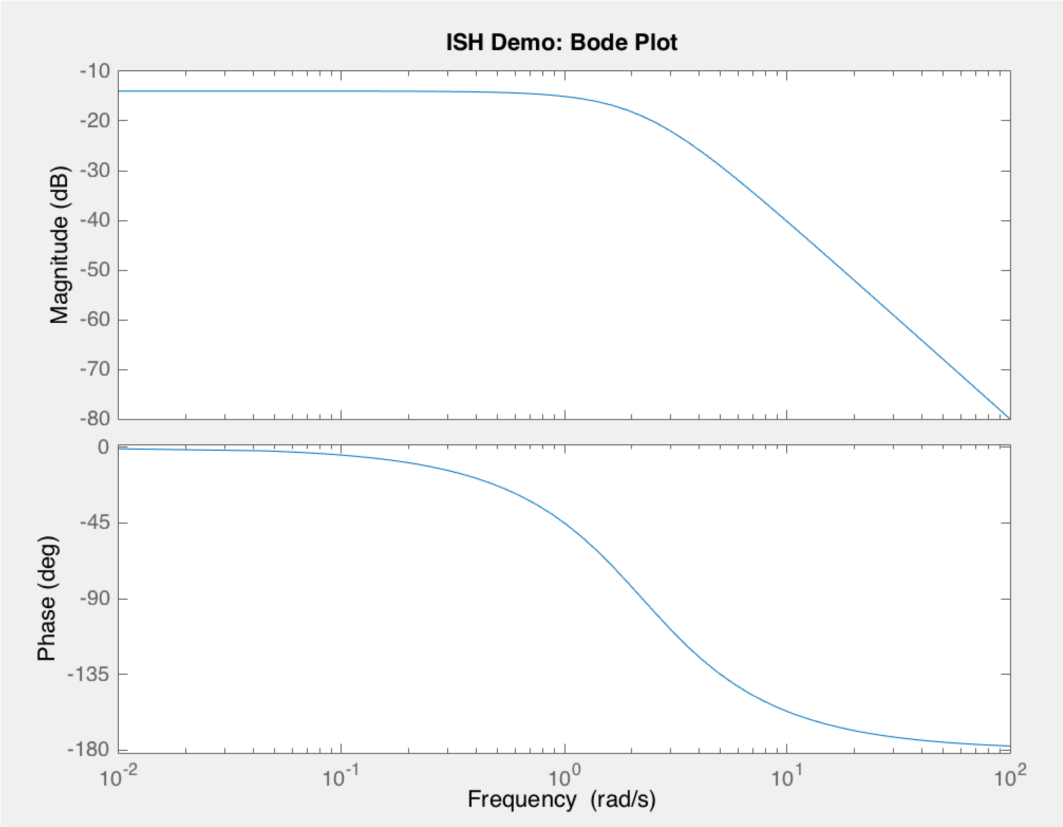
- HINT: Remember the factor s arising from the theorem and the factor 1/s arising from the step function!
- Show the algebraic steps in your calculation of both of these quantities.

3. Bode plot of a transfer function

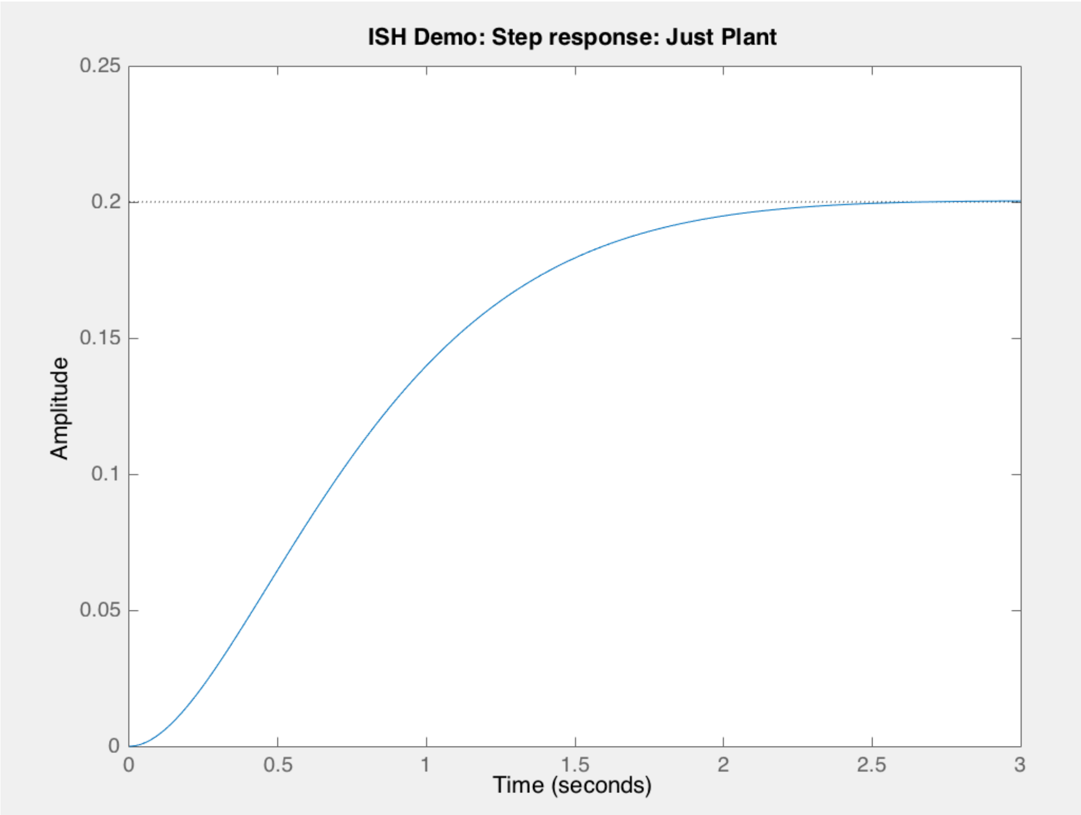
- Use the Matlab tf function to generate a Matlab transfer function for the open loop system and place this in a variable P.
- If you display the contexts of transfer function variable P you should get the following output:

```
P =

        1
  -------------
  s^2 + 4 s + 5

Continuous-time transfer function.
```

- Use Matlab to generate a Bode plot of the transfer function of the mechanical system.
- You should get a plot that looks like this:



- Estimate the bandwidth of the transfer system from the Bode plot
- Use the Matlab step function to plot the step response of the system.
- Look at the Matlab help for step and also pass it a time vector that gives time values between 0 and 3 seconds in steps of 0.01 seconds.
- The output should look something like the plot below:



- Comment on the initial and steady state value of system.
- Do these values match those from the Laplace initial and final value theorems?

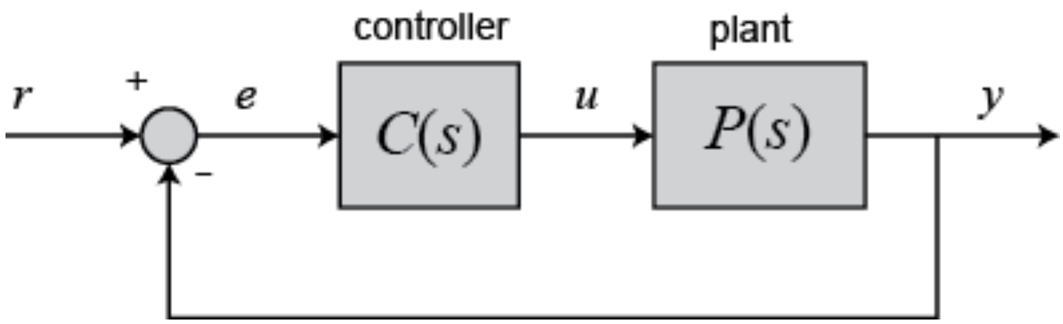
By looking at your plot

- What is the rise time of the response?
- What is the settle time of the response?

4. PID controller

Ensuring that a response quickly reaches unity without overshoot will be the goal of the PID control section in this laboratory task.

- You will now attempt to control the mass-spring-damper system using a PID controller.
- The action of a well-designed control system is to ensure we reach a target value quickly, without overshoot and with low steady state error.
- One method to achieve control uses a PID.
- This involves placing a PID controller C(s) in series with the plant P(s) and making use of negative feedback. This is illustrated below:



- The PID controller consists of three parallel elements:
- ❖ A proportional term that directly relates to the current error value
    - The proportional gain is given as K_p
    - Proportional gain can improve rise time
    - If K_p too high the system can become unstable
- ❖ An integral term that is proportional to both the magnitude of the error and its duration
    - The integrator gain is given as K_i
    - Integral term eliminates residual steady-state error
- ❖ A derivative term that is proportional to the slope of the error over time
    - The differentiator gain is given as K_d
    - Derivative term improves settling time and stability of the system

- The relationship between input error e(t) and output control signal u(t) can be captured by the differential equation:

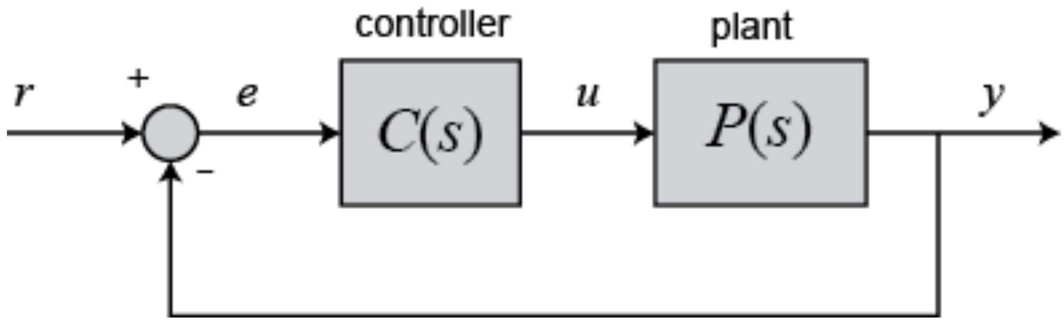
$$u(t) = K_p e(t) + K_i \int e(t)\,dt + K_d \frac{de}{dt}$$

- Use Laplace transforms to write down the transfer function u(s)/(e(s) for this PID controller
- Write down the mathematical expression for the open loop response of the serial combination of the PID controller and plant

5. Using feedback control with only proportional gain

- You will now investigate the effect of just using a large proportional gain in controlling the mass-spring-damper system.
- Open loop performance of our mass-spring-damper plant is poor since the output from the system rises slowly and doesn’t reach unity.
- Set the proportional gain K_{ip} to some relatively high value – e.g. try a value 300
- Set both **integral and differential** gains to **zero**
- Use Matlab code to express the transfer function of the open-loop gain of the system and show all the stages in you calculations in your Matlab code
- Now compute the closed-loop transfer function using the relationship for a feedback system

- HINT: remember for following system:



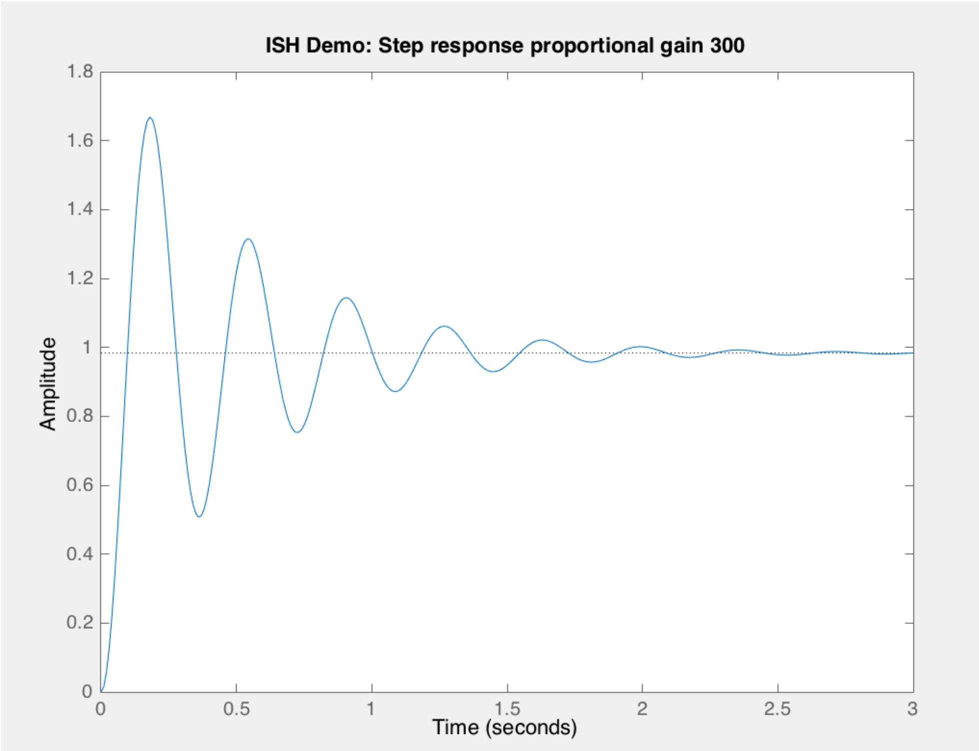
$$\frac{Y(s)}{R(s)} = \frac{C(s)P(s)}{1+C(s)P(s)}$$

where C(s) is the transfer function of the PID controller and P(s) is the transfer function of the mass spring damper that you derived in section 1 and analysed in section 3.

- Using only a proportional gain of 300 you should end up with a closed-loop Matlab transfer function that looks like this:

```
FB =

        300 s^2 + 1200 s + 1500
  ------------------------------------
  s^4 + 8 s^3 + 326 s^2 + 1240 s + 1525

Continuous-time transfer function.
```

- Plot the step response of this transfer function using the same time points as used before for the open loop system.
- You should end up with a plot that looks something like this:

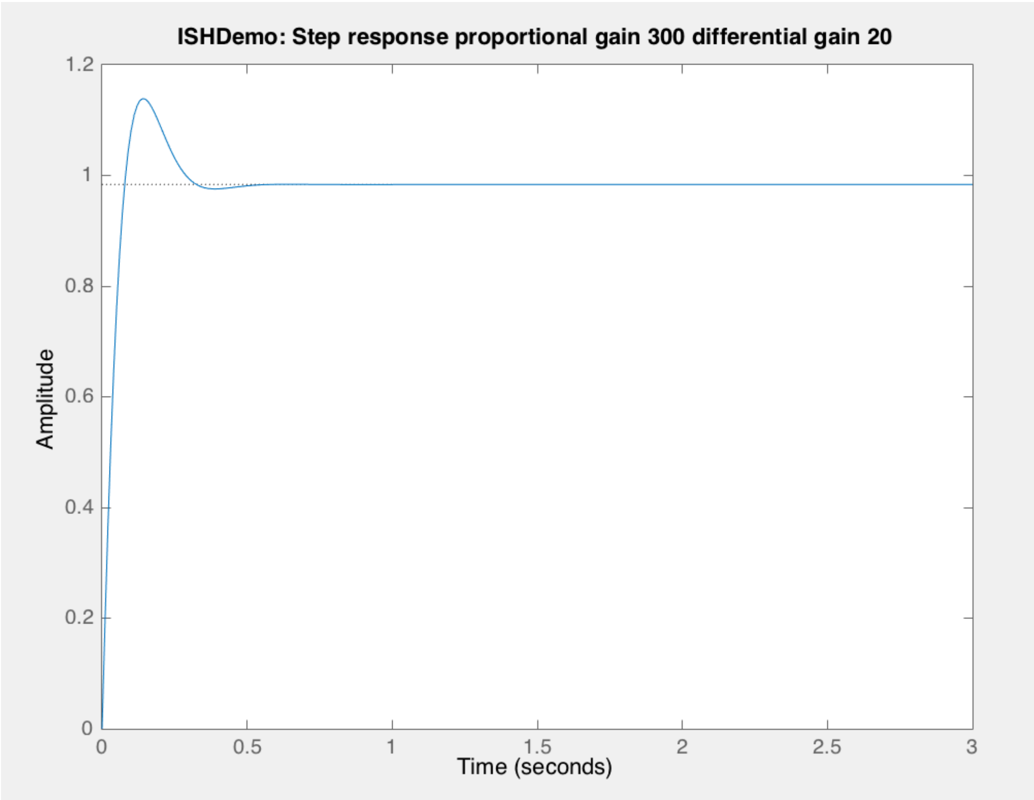


- What can you say about the response of the feedback system using a proportional gain of about 300?
- Consider the time behaviour mesaures of the system: risetime, overshoot, peak time, settle time and steady state error in your analysis. Use Matlab to check the value you see on the plot.
- HINT: The Matlab step function has options to display these values

6. Using feedback control with proportional and differential gain

- Now investigate the effect of using differential gain as well as proportional gain in controlling the mass-spring-damper system.
- The differential term should improve overshoot performance.

- Use proportional gain K_p of 300
- Also include a differential gain K_g of 20
- Once again compute the closed-loop transfer function using the relationship for a feedback system
- Plot the step response of this transfer function as before.
- You should end up with a plot that looks something like this:
-



- What can you say about the response of the feedback system now?
- Again report the time behaviour mesaures of the system.

7. Using feedback control with full PID

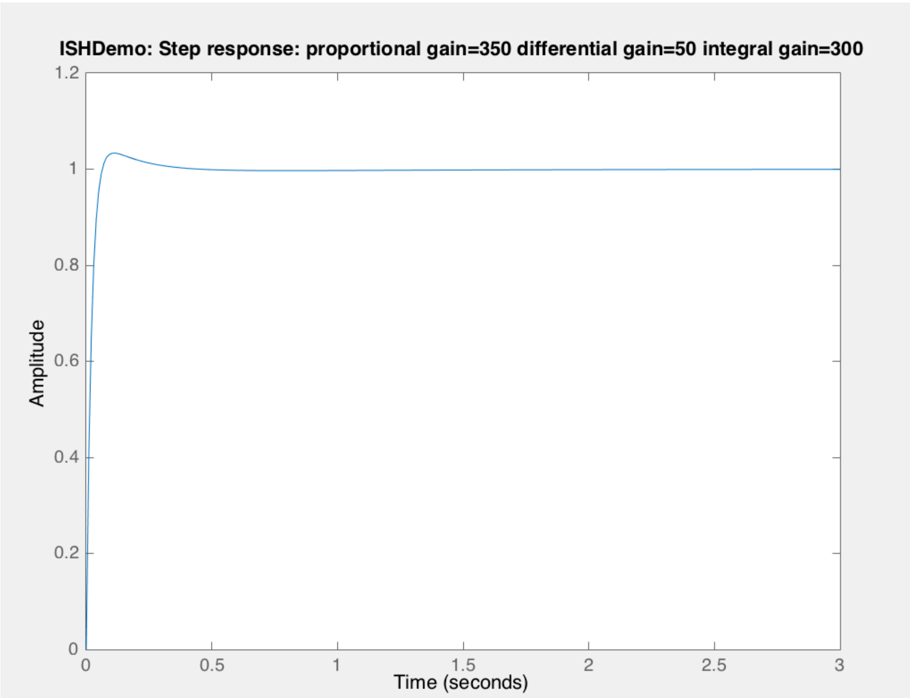
- Investigate the effect of using full PID in controlling the mass-spring-damper system. Use:

  - Proportional gain K_p 350
  - Differential gain K_g of 50
  - Integral gain K_i of 300

- You should end up with a closed-loop Matlab transfer function that looks like this:

```
FB =

     50 s^5 + 550 s^4 + 1950 s^3 + 2950 s^2 + 1500 s
  -----------------------------------------------------
  s^6 + 58 s^5 + 576 s^4 + 1990 s^3 + 2975 s^2 + 1500 s

Continuous-time transfer function.
```

- Plot the step response of this transfer function as before.
- You should end up with a plot that looks something like this:



- Again report the time behaviour mesaures of the system.

- What can you say about the response of this feedback system?
- How does response compare with the open-loop response at the start of this practical session?
- What does this tell you about using PID control and feedback?