Faculty of Engineering

Computer and Systems Engineering Department

CSE 371: Control Systems (1)

Instructor: Prof. Wahied Gharieb Ali

Micro-Project

PID Controller

Submitted by

Eslam Samir Ali Abu El-Ala [ID: 1200259]

Mohamed Ahmed Anwer Abdelhalim [ID: 1101803]

Nourhan Essam Ahmed Shiba El-Hamd [ID: 1201605]

Shaza Ismail Kaoud [ID: 1200717]

Fall (2015-2016)

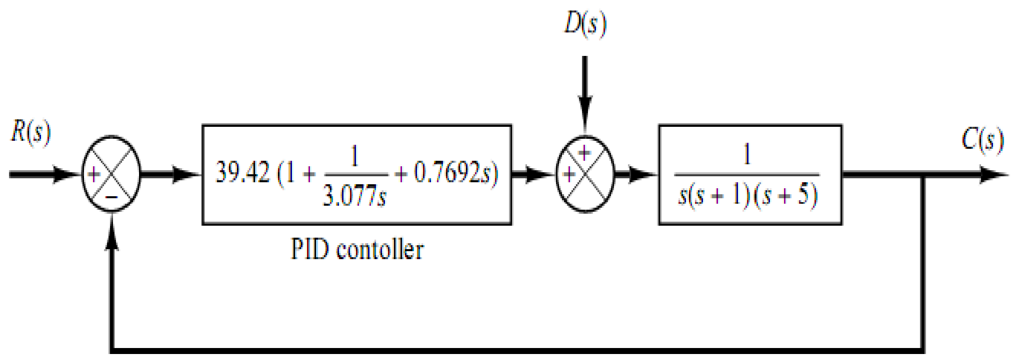
Table of Contents

LIST OF FIGURES

LIST OF TABLES

1. PROBLEM FORMULATION

For the given system,



a) Simulate the closed loop system using the MATLAB/SIMULINK for a unit step input u(t)

b) Explain, which parameter in the PID block has to be modified to decrease the settling time.

c) Study the effect of a constant disturbance input = ±0.1u(t) on the system response.

d) Compute the gain margin and phase margin for the open loop without PID controller and closed loop system with PID controller.

2. BACKGROUND

The PID is the most popular feedback controller algorithm used. It is a robust easily understood algorithm that can provide excellent control performance despite the varied dynamic characteristics of processes.

The PID algorithm consists of three basic modes: the Proportional mode, the Integral mode & the Derivative mode.

In the *s*-domain, the PID controller may be represented as:

$$\mathrm{U(s)} = (\mathrm{K_p} + \frac{\mathrm{K_i}}{\mathrm{s}} + \mathrm{K_d s})\mathrm{E(s)}$$

In the time domain:

$$\mathrm{u(t)} = \mathrm{K_p e(t)} + \mathrm{K_i}\int_0^t e(t)dt + \mathrm{K_d}\frac{\mathrm{de(t)}}{dt})$$

u(t): The PID output

e(t): The error between the output and the set point.

K_p: Proportional Gain

K_i: Integral Gain

K_d: Derivative Gain

While varying each element of the PID controller, the system responses differently. So, the effect of each of these elements on the system must be analyzed in order to get the desired response of the system using the PID controller.

The proportional controller's main responsibility is to force the system to go to the desired value fast, hence, decreases the rise time, but overshoot and settling time increases. Also, steady-state error never reach zero.

When adding integral controller's effect, its main advantage is eliminating the steady-state error. On the other hand, it increases overshoot and settling time also.

When the effect of derivative controller is added, overshoot and settling time decrease while rise time increases.

To sum up, to control the system, P, I, and D values must be chosen with care.

Table 1 The Effect of the PID Controllers on the System

	Rise time	Maximum overshoot	Settling time	Steady-state error
P	**Decrease**	**Decrease**	Small change	**Decrease**
I	**Decrease**	Increase	Increase	**Eliminate**
D	Small change	**Decrease**	**Decrease**	Small

3. SIMULATION

3.1. Closed Loop System Simulation

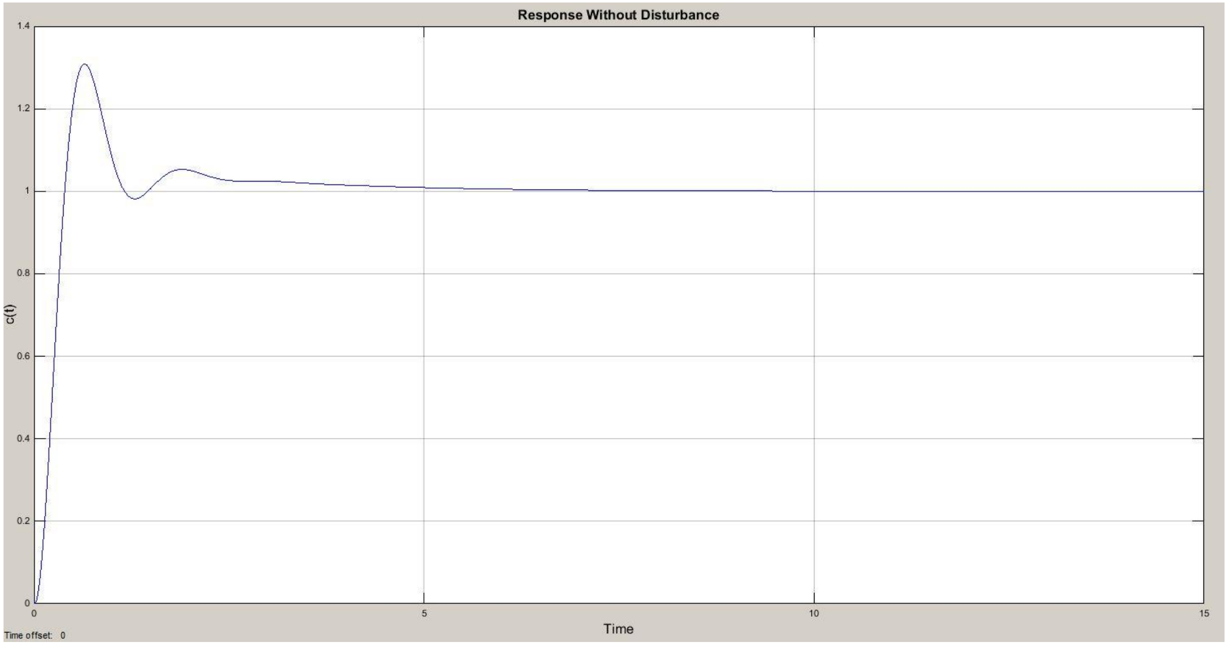


Figure 1 Closed loop system response

3.2. Decreasing Settling Time

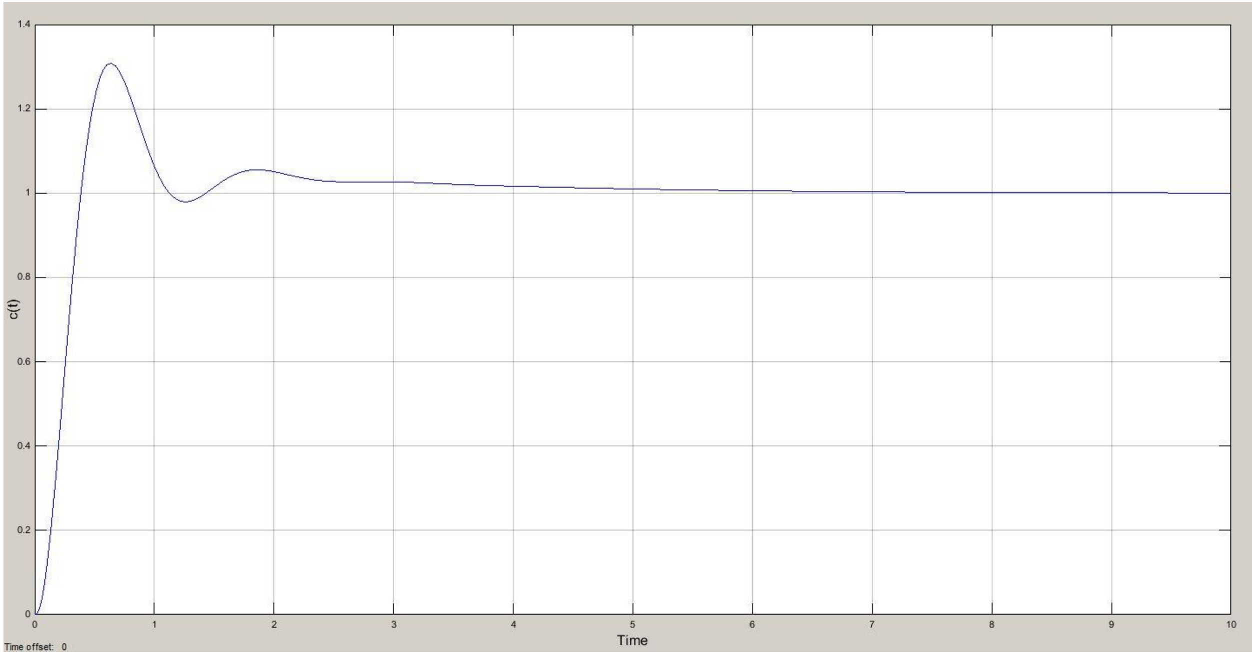


Figure 2 System response before changing the parameters (zoomed in)

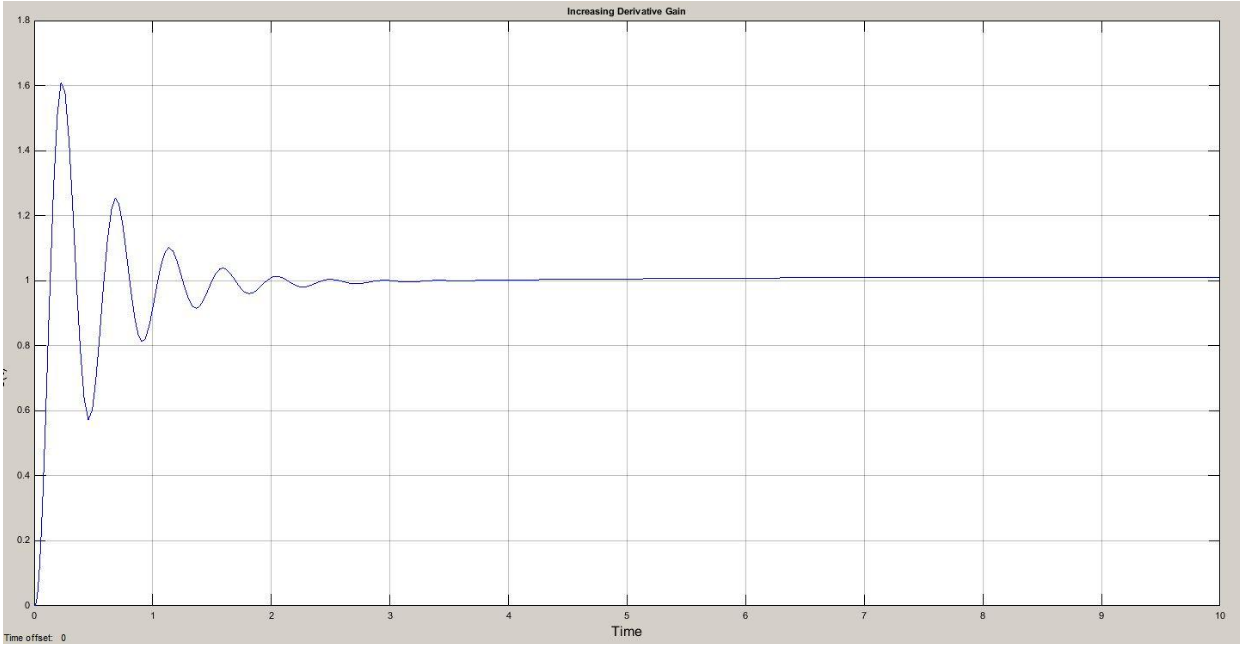


Figure 3 System response after increasing K_d value

- Note: Increasing the Derivative controller gain decreases the settling time for this system but increases the maximum overshoot.
- Explanation: derivative controller gets the derivative of the error, form the derivative law:
  (New error – Old error) / Δt
  And as long as the output is approaching the desired value, the new error is decreasing, so the value of the derivative is always negative, since new error is always less than old error.
  So derivative is always trying to brace the system. So overshoot decreases, hence, settling time decreases.

3.3. Disturbance Effect on the System

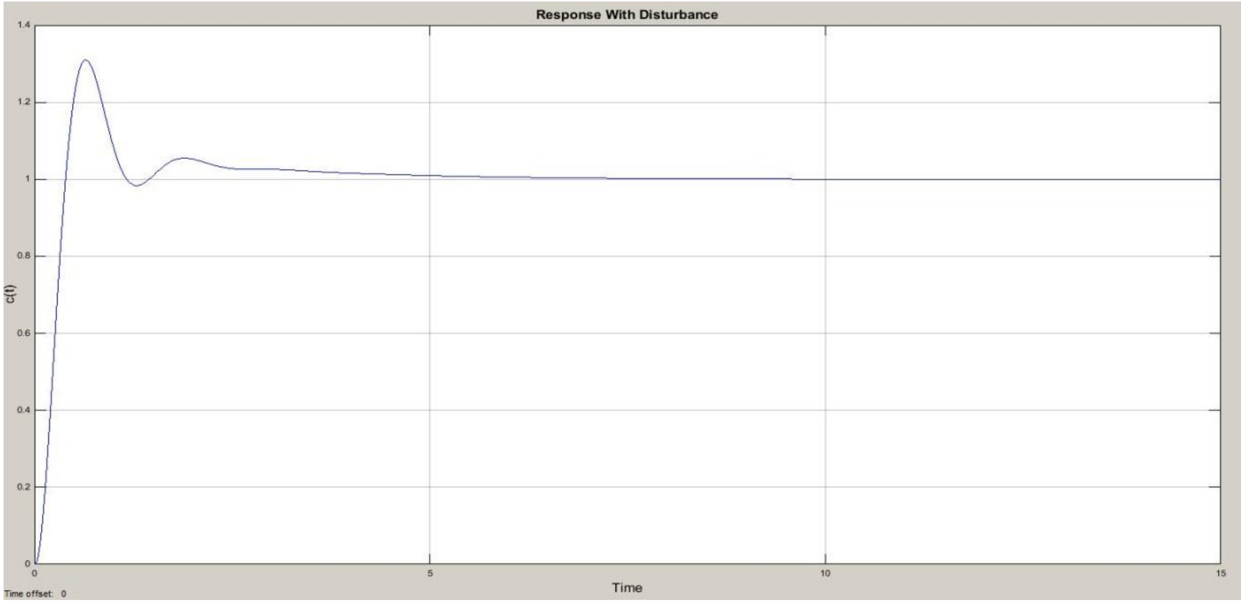


Figure 4 Closed loop system response with disturbance

- Note: Disturbance does not affect the system's response as the PID controller is very powerful that it omits the effect of the disturbance on the system.
  When we simulate the system without PID controller, we can observe disturbance's effect on the system.

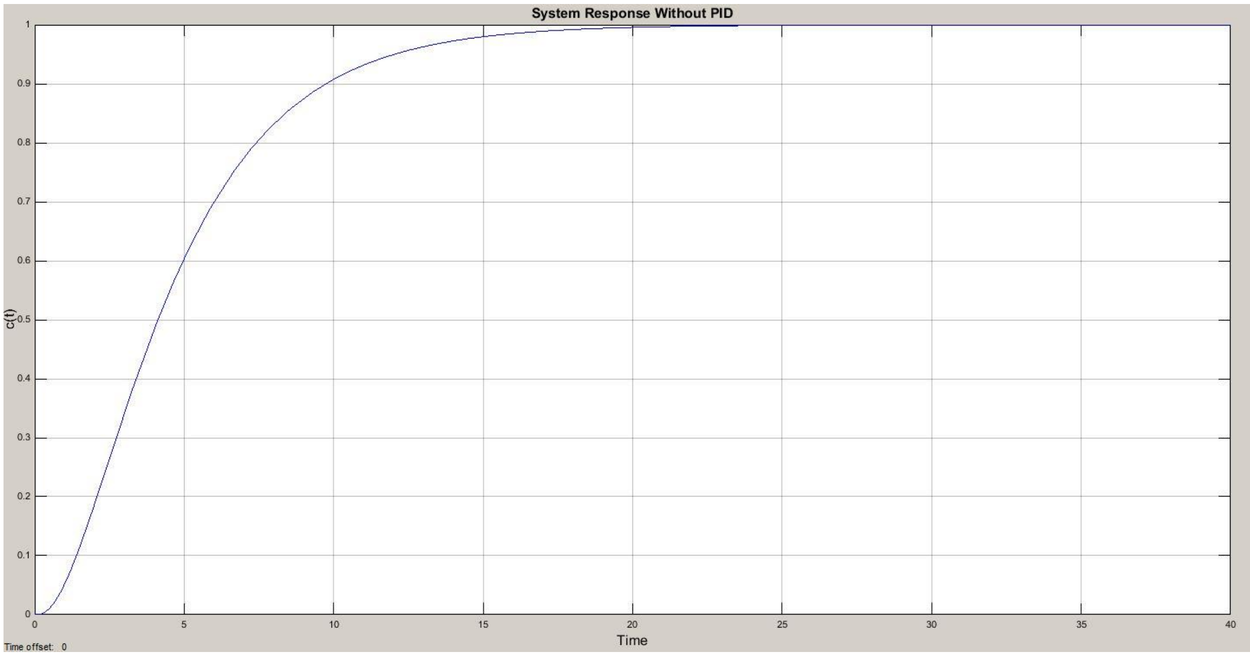


Figure 5 Closed loop system response without disturbance (without PID)

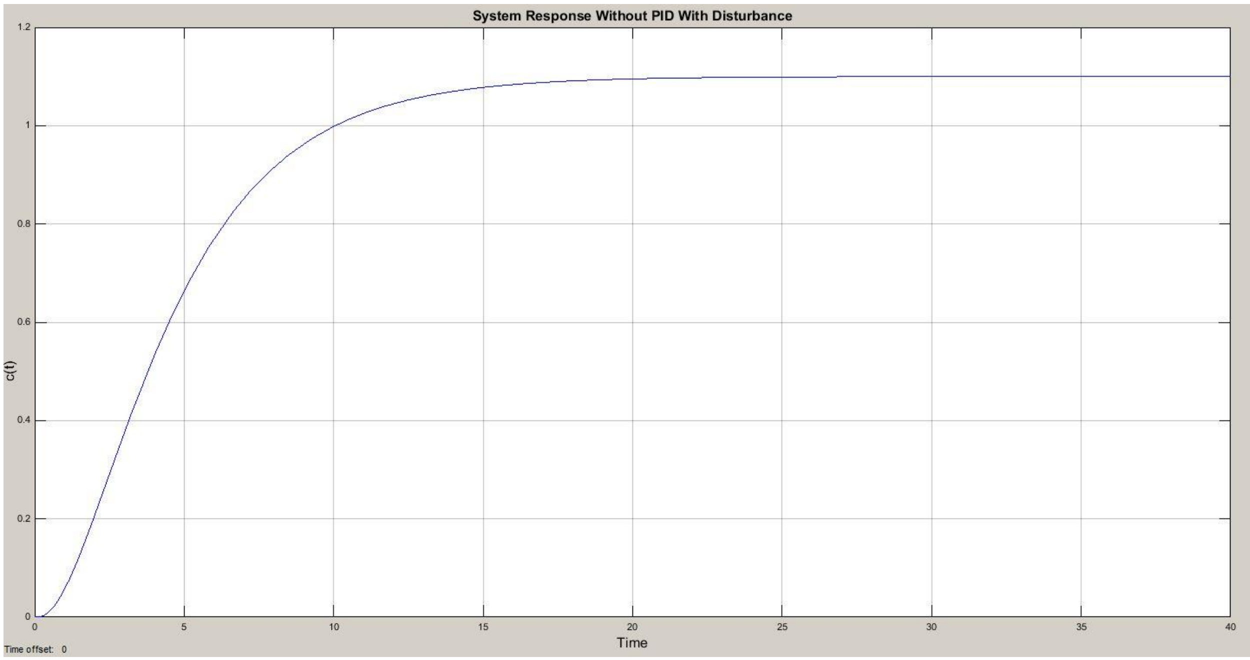


Figure 6 Closed loop system response with disturbance (without PID)

3.4. Phase and Gain Margin for Open Loop System

```
x1 = tf([1],[1 0]);
x2 = tf([1],[1 1]);
x3 = tf([1],[1 5]);
sys = x1*x2*x3
bode(sys)
grid on;
[Gm1,Pm1,Wgm1,Wpm1] = margin(sys)
```

Figure 7 Using Matlab to calculate Pm and Gm of the open loop system

```
sys =

          1
  -----------------
  s^3 + 6 s^2 + 5 s

Continuous-time transfer function.


Gm1 =

    30


Pm1 =

   76.6603


Wgm1 =

    2.2361


Wpm1 =

    0.1961
```

Figure 6 Matlab results of the open loop system

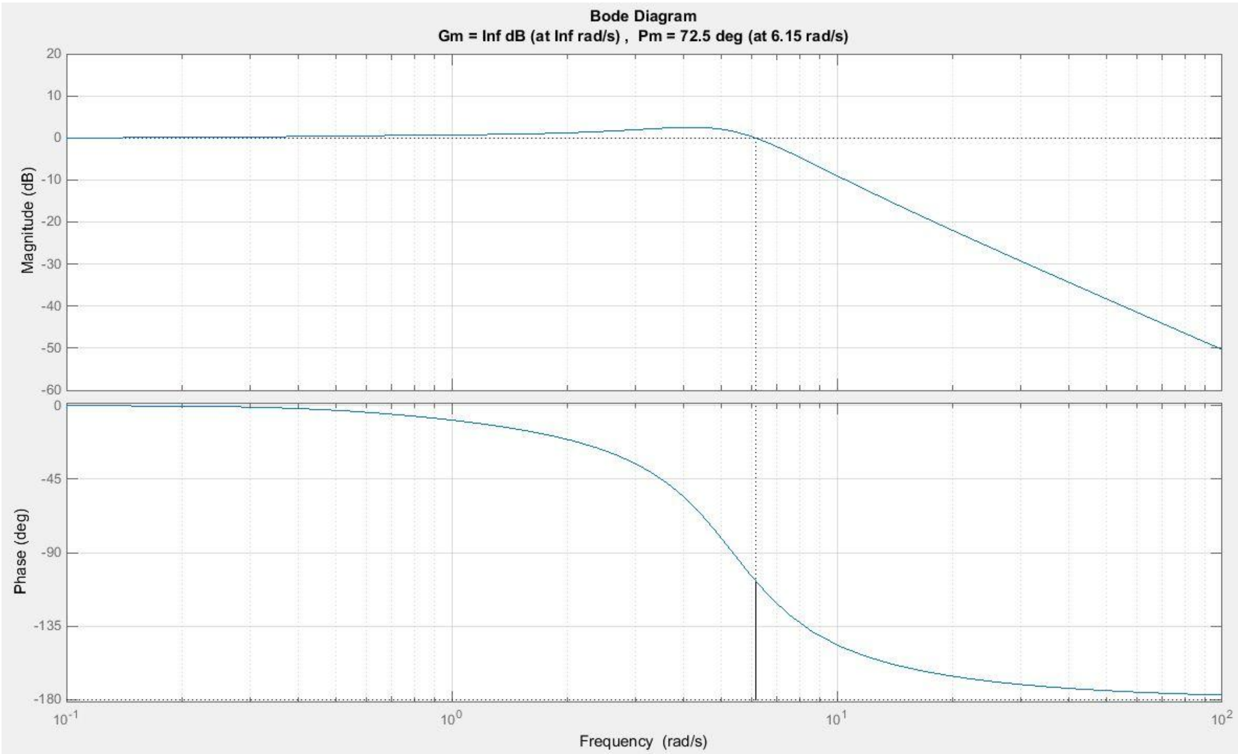


Figure 8 Bode plot of the open loop system

3.5. Phase and Gain Margin for Closed Loop System

```
y1 = 39.42/(3.077*tf('s'));
y2 = 39.42*0.7692*tf('s');
y3 = 39.42;
PID = y1+y2+y3;
G = PID*sys;
closedLoopSys = feedback(G,1)
bode(closedLoopSys)
margin(closedLoopSys)
grid on;
[Gm2,Pm2,Wgm2,Wpm2] = margin(closedLoopSys)
```

Figure 9 Using Matlab to calculate Pm and Gm of the closed loop system

```
closedLoopSys =

               93.3 s^2 + 121.3 s + 39.42
  ---------------------------------------------------
  3.077 s^4 + 18.46 s^3 + 108.7 s^2 + 121.3 s + 39.42

Continuous-time transfer function.


Gm2 =

   Inf


Pm2 =

   72.5314


Wgm2 =

   Inf


Wpm2 =

    6.1542
```

Figure 10 Matlab results of the closed loop system

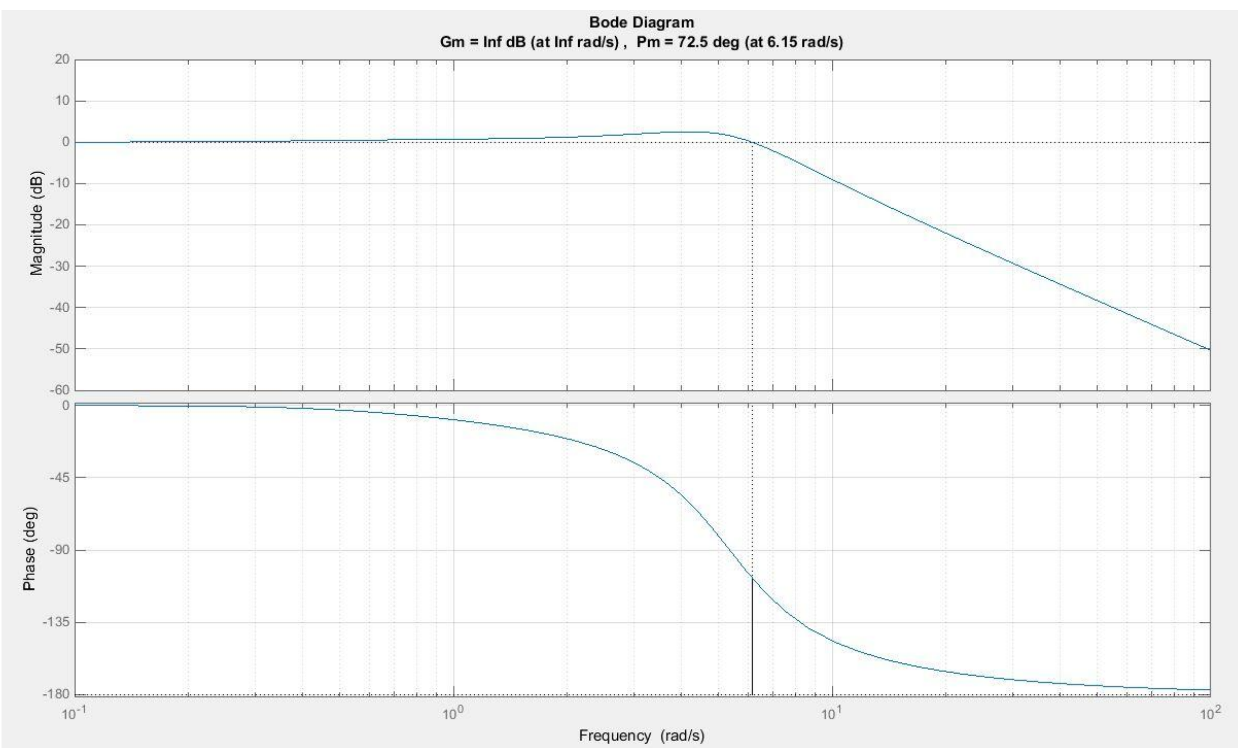


Figure 11 Bode plot of the closed loop system

4. CONCLUSION

PID is a very powerful controller. One must choose the values for proportional, integral, and derivative controllers with care to have the desired performance of the system as each of them has positive and negative impact on the system.

5. REFERENCES

[1] "Modern Control Engineering", 5th edition, Katsuhiko Ogata.

[2] Lecture Slides: "Topic#11 PID Controller Design", Dr. Wahid Gharib, 2015.

Appendix A: MATLAB Code

```
x1 = tf([1],[1 0]);
x2 = tf([1],[1 1]);
x3 = tf([1],[1 5]);
sys = x1*x2*x3;
bode(sys)
margin(sys)
grid on;
[Gm1,Pm1,Wgm1,Wpm1] = margin(sys)

y1 = 39.42/(3.077*tf('s'));
y2 = 39.42*0.7692*tf('s');
y3 = 39.42;
PID = y1+y2+y3;
G = PID*sys;
closedLoopSys = feedback(G,1)
bode(closedLoopSys)
margin(closedLoopSys)
grid on;
[Gm2,Pm2,Wgm2,Wpm2] = margin(closedLoopSys)
```

Appendix B: SIMULINK Model

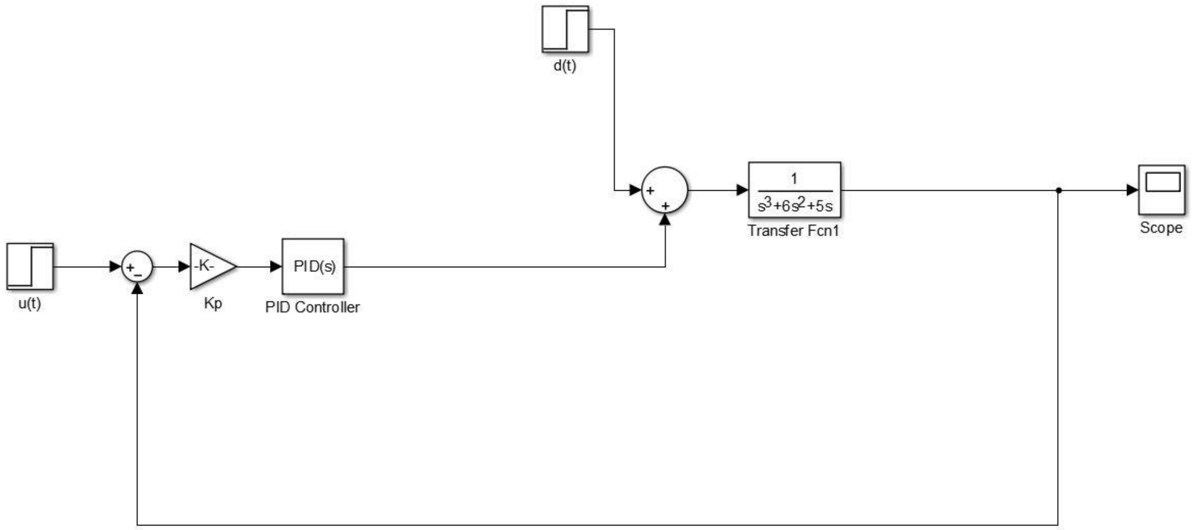


Figure 12 SIMULINK Model

- Note: because MATLAB has no s block and tf can't have a higher order numerator than the denominator we used the PID Controller Block but it has a derivative filter coefficient of default value 100