

# The PID Control of the Electromechanical System

Spartak Poçari <sup>1</sup>,

Eng. PhD Student, Faculty of Mechanical Engineering, Polytechnic University of Tirana, Albania

**ABSTRACT:** PID (Proportional-Integral-Derivative) control is the most common control algorithm used in industry and has been universally accepted in industrial control.

With goal to understand how each of  $K_p$ ,  $K_i$  and  $K_d$  coefficients contributes to obtain fast rise time, minimum overshoot, and no steady-state error. We have build a simulation program in MatLab to analyze the time response of system for: Open-Loop, Proportional control, Proportional-Derivative Control, Proportional-Integral Control and Proportional-Integral-Derivative Control (PID). To concretize, we have taken into consideration an electromechanical system that is widely used in industry. Numerical results tells that that the proportional control reduced both the rise time and the steady-state error, increased the overshoot, and decreased the settling time by small amount, derivative controller reduced both the overshoot and the settling time, integral controller eliminated the steady-state error, Proportional-Integral-Derivative Control (PID) obtain a closed-loop system with no overshoot, fast rise time, and no steady-state error. And the mathematical model is accurate and can be used to stabilize these kind of systems in industry.

**KEYWORDS:** PID control, Electromechanical system, Mathematical Model, Simulation, Pneumatics

## I. INTRODUCTION

PID algorithm consists of three basic coefficients; proportional, integral and derivative which are varied to get optimal response. The basic idea behind a PID controller is to read a sensor, then compute the desired actuator output by calculating proportional, integral, and derivative responses and summing those three components to compute the output as show the figure 1.

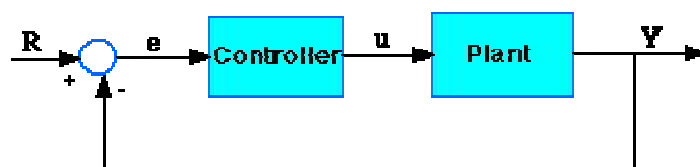


Figure.1. Closed Loop Control

**Plant:** A system to be controlled

**Controller:** Provides the excitation for the plant, designed to control the overall system behavior

First, let's take a look at how the PID controller works in a closed-loop system using the schematic shown above. The variable (e) represents the tracking error, the difference between the desired input value (R) and the actual output (Y). This error signal (e) will be sent to the PID controller, and the controller computes both the derivative and the integral of this error signal. The signal (u) just past the controller is now equal to the proportional gain ( $K_p$ ) times the

## International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 9, September 2017

magnitude of the error plus the integral gain ( $K_i$ ) times the integral of the error plus the derivative gain ( $K_d$ ) times the derivative of the error.

$$u = K_p e + K_i \int e dt + K_d \frac{de}{dt} \tag{1}$$

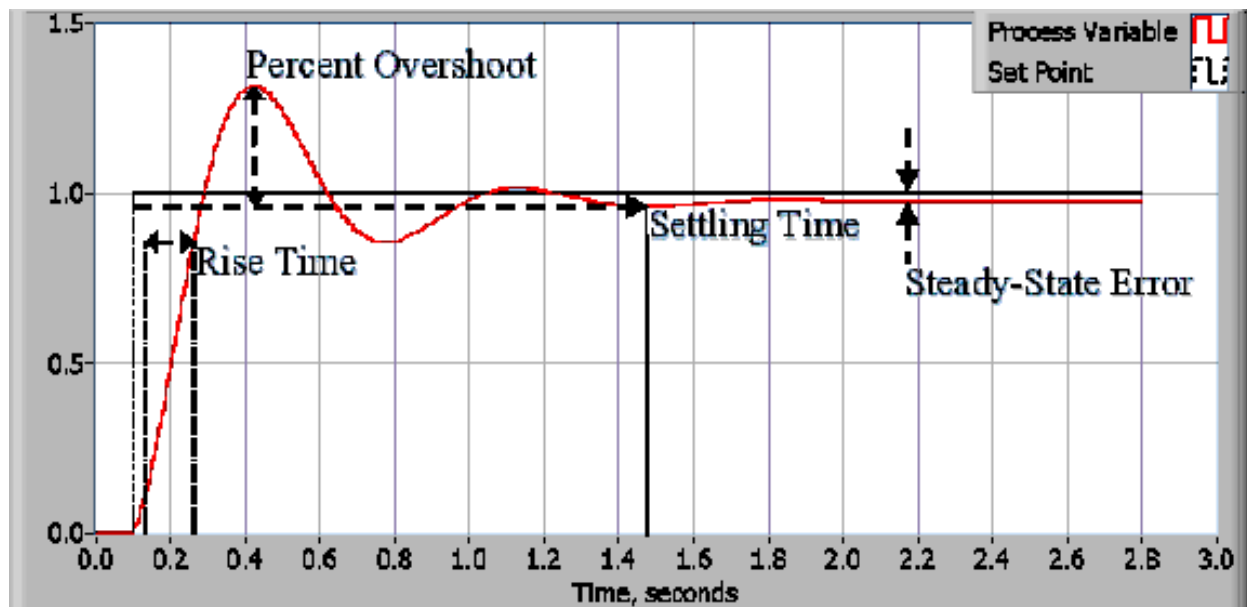
This signal ( $u$ ) will be sent to the plant, and the new output ( $Y$ ) will be obtained. This new output ( $Y$ ) will be sent back to the sensor again to find the new error signal ( $e$ ). The controller takes this new error signal and computes its derivative and its integral again. This process goes on and on.

The transfer function of the PID controller looks like the following:

$$K_p + \frac{K_i}{s} + K_d s = \frac{K_d s^2 + K_p s + K_i}{s} \tag{2}$$

Where,  $K_p$  is proportional gain,  $K_i$  is integral gain,  $K_d$  is derivative gain.

A proportional controller ( $K_p$ ) will have the effect of reducing the rise time and will reduce but never eliminate the steady-state error. An integral control ( $K_i$ ) will have the effect of eliminating the steady-state error, but it may make the transient response worse. A derivative control ( $K_d$ ) will have the effect of increasing the stability of the system, reducing the overshoot, and improving the transient response. Effects of each of controllers  $K_p$ ,  $K_d$ , and  $K_i$  on a closed-loop system are summarized in the table shown below.



CL RESPONSE	RISE TIME	OVERSHOOT	SETTLING TIME	S-S ERROR
$K_p$	Decrease	Increase	Small Change	Decrease
$K_i$	Decrease	Increase	Increase	Eliminate
$K_d$	Small Change	Decrease	Decrease	Small Change

**Figure.2.** Effect of PID Controllers on Closed-Loop System

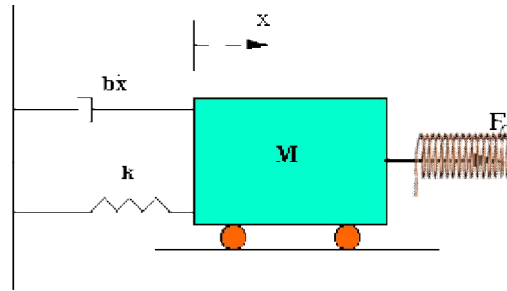
## International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 9, September 2017

For our study, we have take in consideration a simple electromechanical system composed by a coil, a mass, a spring, and a damper as in figure below.



**Figure.3.**Coil-mass-spring-dumper System

The modeling equation of this system is :

$$M\ddot{x} + b\dot{x} + kx = F_c \tag{3}$$

Where:

M- is the mass of system

b-is the damping coefficient

k- is the spring coefficient

F<sub>c</sub>- is the coil force

For the actuating force of the electromagnet when passing a given intensity of current we can write:

$$F_c = 2\pi a N \beta i \tag{4}$$

Where (a) is the coil radius, (N) is the coil number , (β) is the magnetic flux, ( i<sub>c</sub> ) is the currents feeding the coil.

Taking the Laplace transform of the modeling equation, we get:

$$Ms^2X(s) + bsX(s) + kX(s) = F(s) \tag{5}$$

The transfer function between the displacement X(s) and the input F(s) then becomes:

$$\frac{X(s)}{F(s)} = \frac{1}{Ms^2 + bs + k} \tag{6}$$

Let M = 1kg, b = 20 N.s/m, k = 40 N/m, and F(s) = 1. If we use these values in the above transfer function, the result is:

$$\frac{X(s)}{F(s)} = \frac{1}{s^2 + 20s + 40} \tag{7}$$

With goal to show you how each of K<sub>p</sub>, K<sub>i</sub> and K<sub>d</sub> contributes to obtain fast rise time, minimum overshoot, and no steady-state error. We have build a simulation program in MatLab to analyze the time response of system for: Open-Loop, Proportional control, Proportional-Derivative Control, Proportional-Integral Control and Proportional-Integral-Derivative Control (PID).

# International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 9, September 2017

## II. NUMERICAL RESULTS

### Open-Loop Control

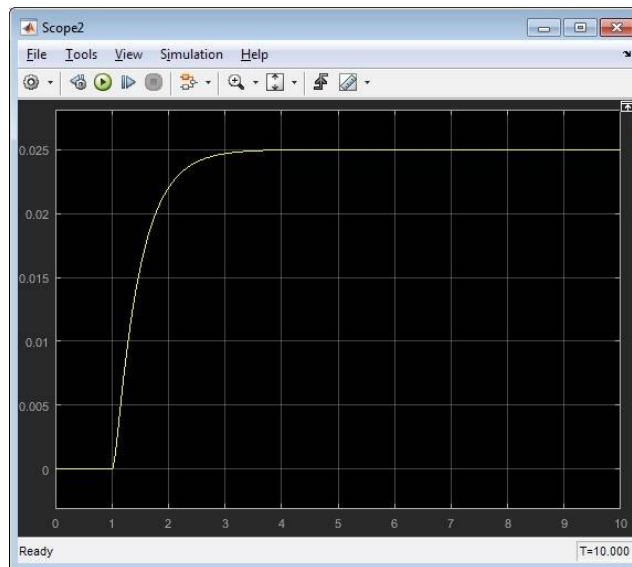


Figure.4. Open-Loop Step Response

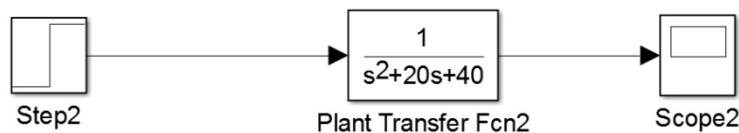


Figure.5. Block Diagram for Open-Loop control in Simulink

The DC gain of the plant transfer function is  $1/40$ , so  $0.025$  is the final value of the output to a unit step input. This corresponds to the steady-state error of  $0.975$ , quite large indeed. Furthermore, the rise time is about  $2.2$  seconds, and the settling time is about  $3$  seconds. Let's design a controller that will reduce the rise time, reduce the settling time, and eliminates the steady-state error.

### Proportional Control

From the table in Figure 2, we see that the proportional controller ( $K_p$ ) reduces the rise time, increases the overshoot, and reduces the steady-state error. The closed-loop transfer function of the above system with a proportional controller is:

$$\frac{X(s)}{F(s)} = \frac{K_p}{s^2 + 20s + (40 + K_p)} \quad (8)$$

Let the proportional gain ( $K_p$ ) equal  $3600$ .

# International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 9, September 2017

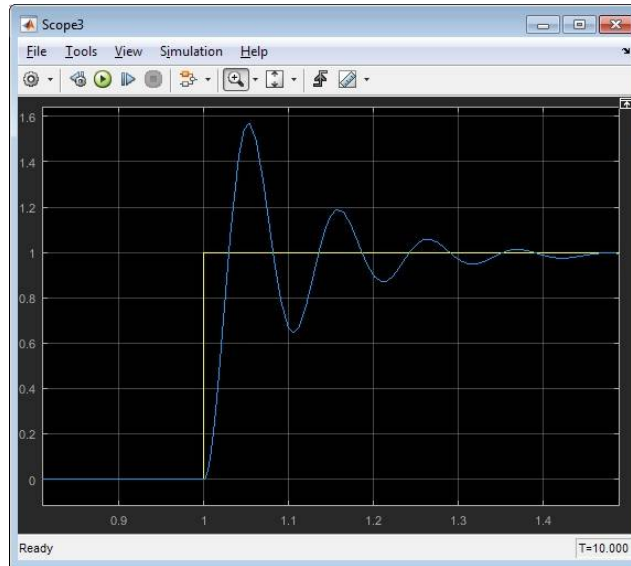


Figure.6. Proportional Control

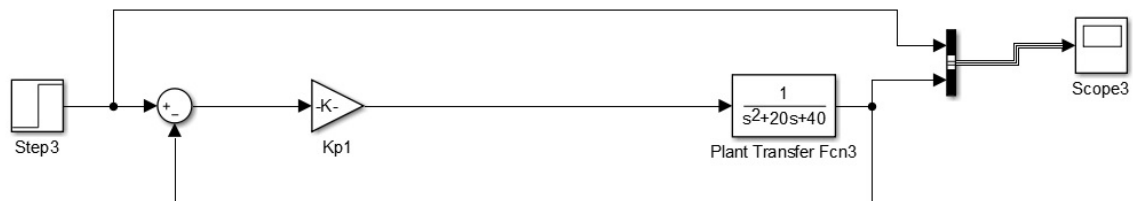


Figure.7. Proportional Control block Diagram in Simulink

The graph shows that the proportional controller reduced both the rise time and the steady-state error, increased the overshoot, and decreased the settling time by small amount.

## Proportional-Derivative Control

From the table in Figure 2, we see that the derivative controller ( $K_d$ ) reduces both the overshoot and the settling time. The closed-loop transfer function of the given system with a PD controller is:

$$\frac{X(s)}{F(s)} = \frac{K_D s + K_P}{s^2 + (20 + K_D)s + (40 + K_P)} \quad (9)$$

Let  $K_p = 2659$  and  $K_d = 48.37$ .

# International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 9, September 2017

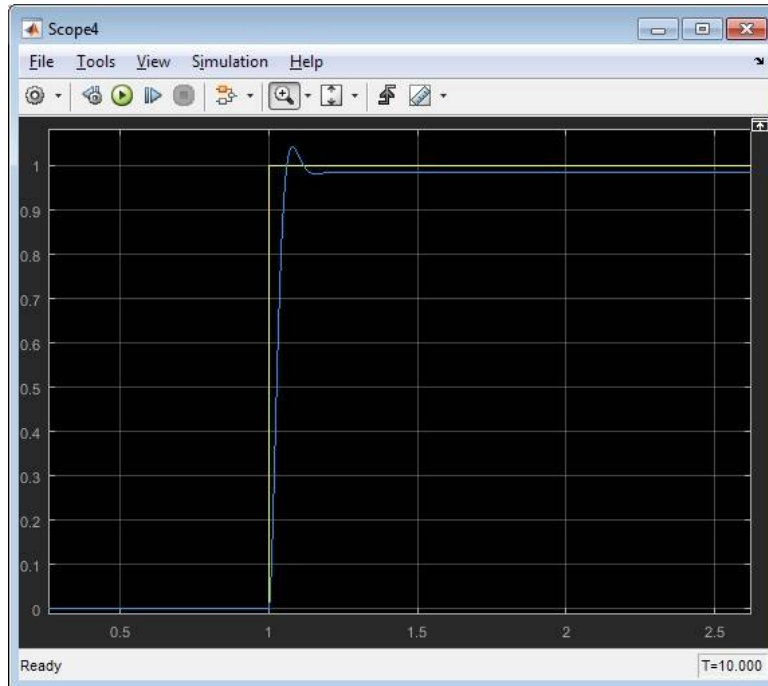


Figure.8. Proportional-Derivative Control

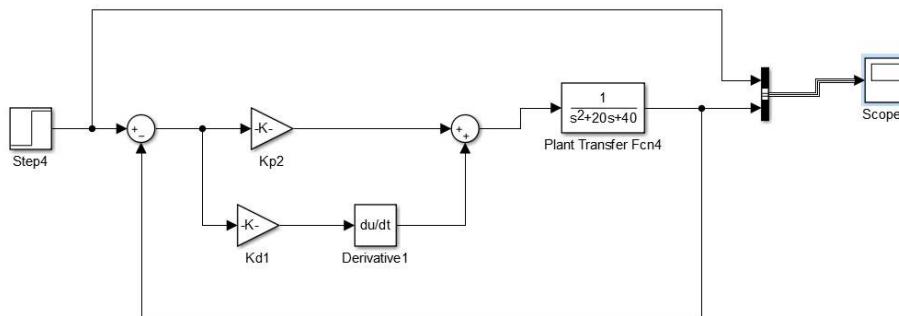


Figure.9. Proportional-Derivative Control block Diagram in Simulink

Compare the graph in Figure 8 to the graph in Figure 6. The step response plot shows that the derivative controller reduced both the overshoot and the settling time, and had a small effect on the rise time and the steady-state error.

## Proportional-Integral Control

From the table in Figure 2, we see that an integral controller ( $K_i$ ) decreases the rise time, increases both the overshoot and the settling time, and eliminates the steady-state error. For the given system, the closed-loop transfer function with a PI control is:

$$\frac{X(s)}{F(s)} = \frac{K_p s + K_i}{s^3 + 20s^2 + (40 + K_p)s + K_i} \quad (10)$$

Let  $K_p = 56.8$  and  $K_i = 241.6$ .

# International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 9, September 2017

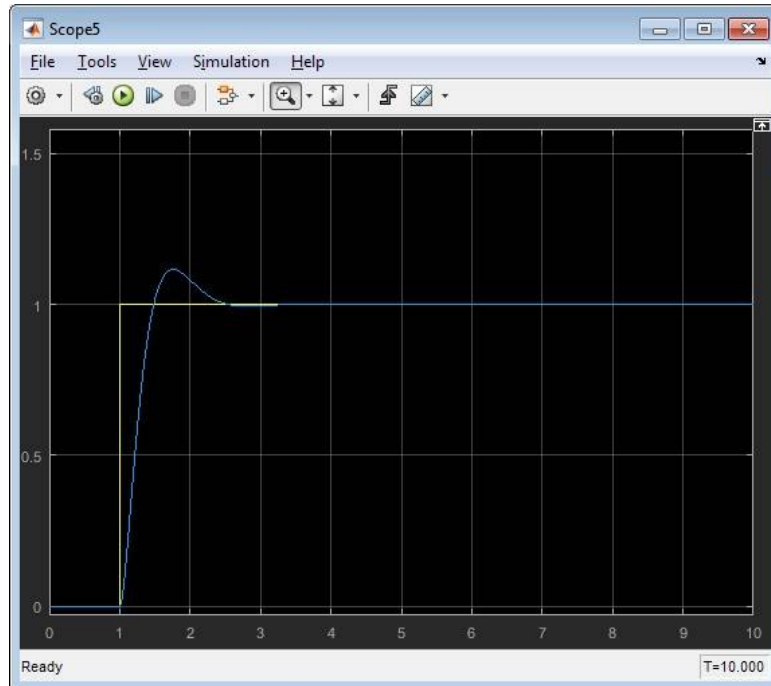


Figure.10. Proportional-Integral Control

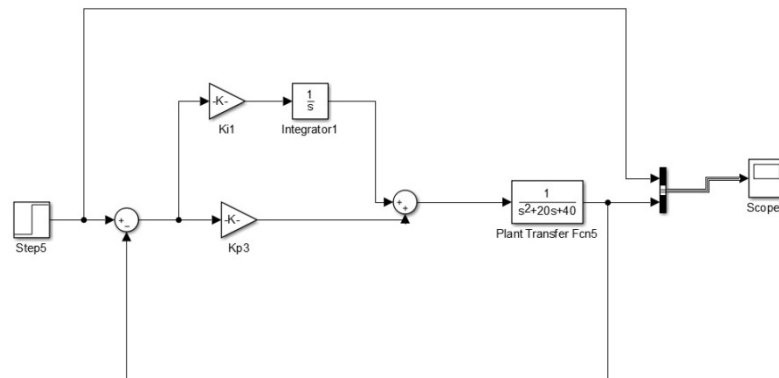


Figure.11. Proportional-Integral Control block Diagram in Simulink

We have reduced the proportional gain ( $K_p$ ) because the integral controller also reduces the rise time and increases the overshoot as the proportional controller does (double effect). The above response in Figure 11 shows that the integral controller eliminated the steady-state error.

### Proportional-Integral-Derivative Control (PID)

Now, let's take a look at a PID controller. The closed-loop transfer function of the given system with a PID controller is:

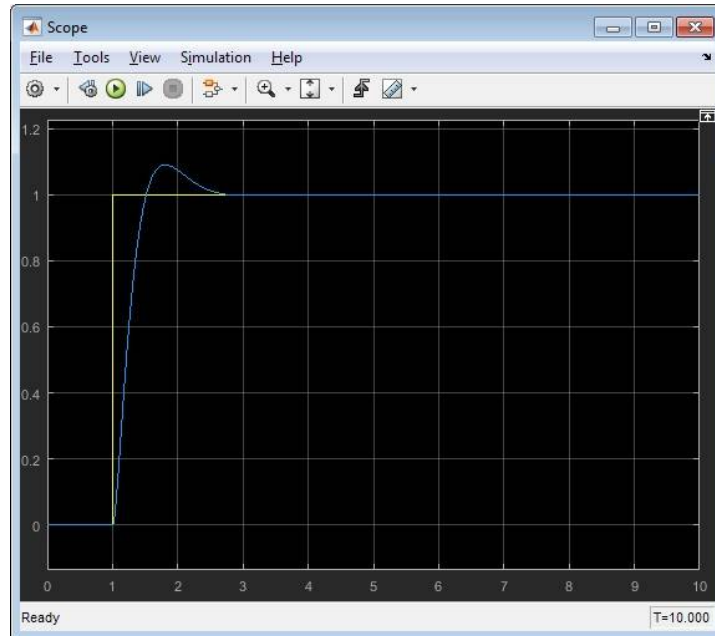
$$\frac{X(s)}{F(s)} = \frac{K_D s^2 + K_P s + K_I}{s^3 + (20 + K_D) s^2 + (40 + K_P) s + K_I} \quad (11)$$

# International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

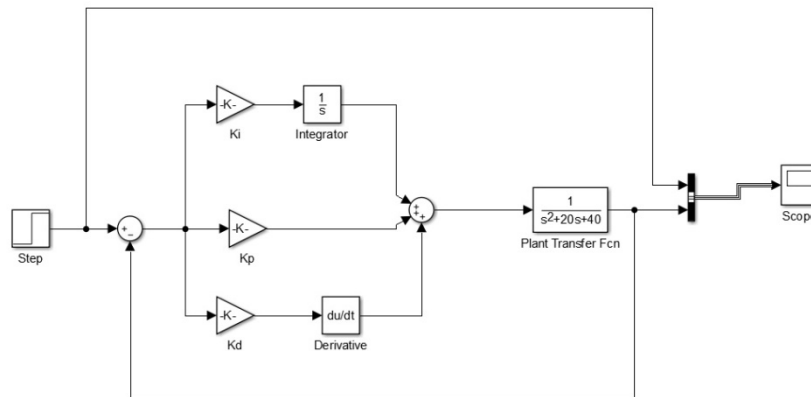
Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 9, September 2017



**Figure.11.** Proportional-Integral-Derivative Control

The gains  $K_p=65.56$ ,  $K_i=231.926$ , and  $K_d= 1.811$  provided the desired response.



**Figure.12.** (PID) Control block Diagram in Simulink

Now, we have obtained a closed-loop system with no overshoot, fast rise time, and no steady-state error.

### III. CONCLUSION

The Open-Loop control transfer function is  $1/40$ , so  $0.025$  is the final value of the output to a unit step input. This corresponds to the steady-state error of  $0.975$ , quite large indeed. Furthermore, the rise time is about  $2.2$  seconds, and the settling time is about  $3$  seconds.

Proportional controller reduced both the rise time and the steady-state error, increased the overshoot, and decreased the settling time by small amount.

Proportional-Derivative controller reduced both the overshoot and the settling time, and had a small effect on the rise time and the steady-state error as you can see Comparing the graph in Figure 8 with the graph in Figure 6.



# International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 9, September 2017

Proportional - Integral controller eliminated the steady-state error as you can see in Figure 11.

Proportional-Integral-Derivative Control (PID) obtain a closed-loop system with no overshoot, fast rise time, and no steady-state error.

To obtain a PID controller for a given system, follow the steps shown below to obtain a desired response.

1. Obtain an open-loop response and determine what needs to be improved
2. Add a proportional control to improve the rise time
3. Add a derivative control to improve the overshoot
4. Add an integral control to eliminate the steady-state error

We do not need to implement all three controllers (proportional, derivative, and integral) into a single system, if not necessary. For example, if a PI controller gives a good enough response (like the above example), then you don't need to implement a derivative controller on the system.

## REFERENCES

- [1] A. Londo "Transmission Hidropneumatike" 2008 Sh.B.L.U Tiranë 2008
- [2] John Cave Electromechanical Systems London 2005
- [3] Andrew Parr "Hydraulics and Pneumatics" Third edition 2011
- [4] Holly Moore "MATLAB for Engineers"
- [5] Brian H. Hahn Daniel T. Valentine "Essential MATLAB for Engineers and Scientists"
- [6] Stephen J. Chapman "MATLAB Programming for Engineers"
- [7] Michael A. Johnson and Mohammad H. Moradi "PID Control" London 2005
- [8] I. Ninka. "Metoda Numerike" Tiranë: 1983
- [9] F. Hoxha "Analiza Numerike" Tiranë: Sh.B.U. 1997
- [10] Nuovo Colombo. "Manuale dell'Ingegnere" Torino: 1985
- [11] Karl Johan Astrom "Control System Design" 2002 (Chapter 6)
- [12] R.C. Hibbeler "Engineering Mechanics: Dynamics" 14th Edition 2015.