Umm Al-Qura University
College of Engineering & Islamic Architecture
Mechanical Engineering Department

Course #: 804465–Automatic Control Lab #6

PID Controller Design

Objective

- Introduction to PID Controller.
- To determine a mathematical model of PID controller.
- To obtain transfer function of the Controller
- Use Simulink to Implement PID Controller to get close loop response of the system (Lab # 3).
- Tune PID controller to get acceptable response, i.e.,
  - ✓ Fast rise time
  - ✓ Minimum overshoot
  - ✓ No steady-state error

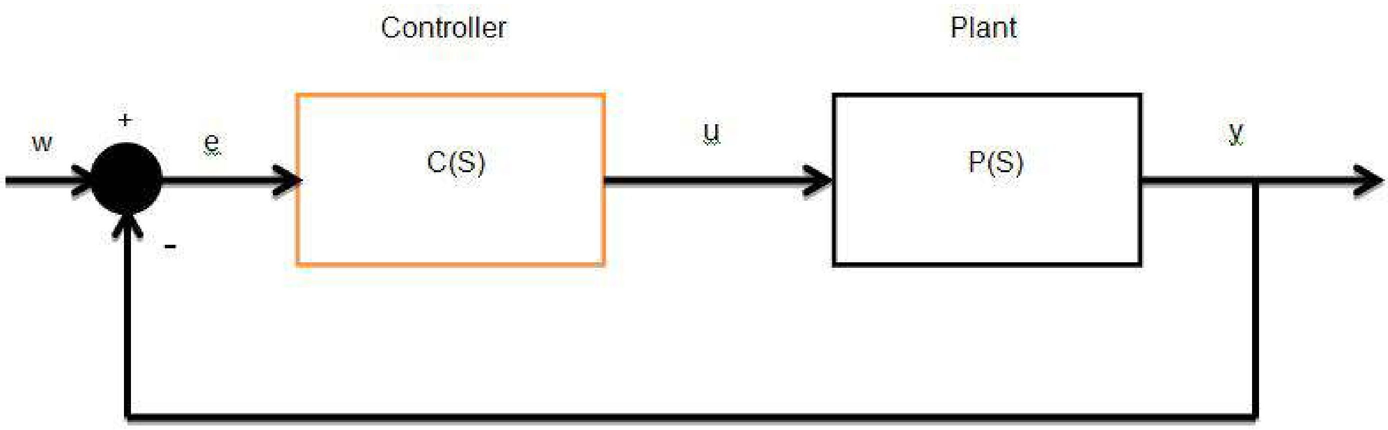
Introduction to PID Controller

In this lab tutorial we will study a simple but very useful feedback compensator structure, the Proportional-Integral-Derivative (PID) controller. It is currently one of the most frequently used controllers in the process industry. We will observe the effect of each of the PID parameters on the closed-loop dynamics and demonstrate how to use a PID controller to improve the system performance.
Ina PID controller the control variable is generated from a term proportional to the error, a term which is the integral of the error, and a term which is the derivative of the error.
Proportional**:** the error is multiplied by a gain *Kp*. A very high gain may cause instability, and a very low gain may cause the system to drift away.
Integral**:** the integral of the error is taken and multiplied by a gain *Ki*. The gain can be adjusted to drive the error to zero in the required time. A too high gain may cause oscillations and a too low gain may result in a sluggish response.
Derivative**:** The derivative of the error is multiplied by a gain *Kd*. Again, if the gain is too high the system may oscillate and if the gain is too low the response may be sluggish.
A proportional controller (*Kp*) will have the effect of reducing the rise time and will reduce but never eliminate the steady-state error. An integral control (*Ki*) will have the effect of eliminating the steady state error for a constant or step input, but it may make the transient response slower. A derivative control (*Kd*) will have the effect of increasing the stability of the system, reducing the overshoot, and improving the transient response.
The effects of each of controller parameters, , , and on a closed-loop system are summarized in the table below.

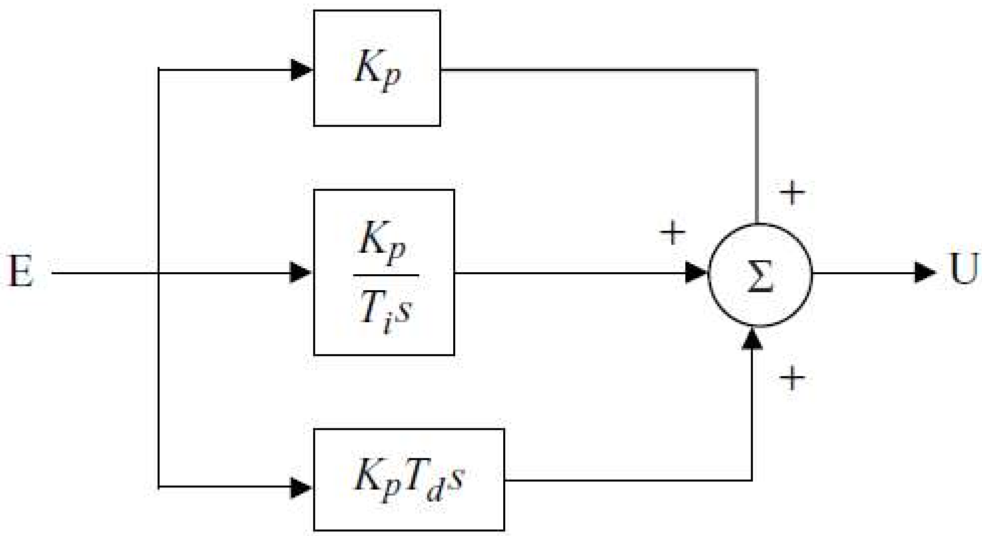
CL RESPONSE	RISE TIME	OVERSHOOT	SETTLING TIME	S-S ERROR
Kp	Decrease	Increase	Small Change	Decrease
Ki	Decrease	Increase	Increase	Eliminate
Kd	Small Change	Decrease	Decrease	No Change

Mathematical Model of PID Controller

Consider the following unity feedback system:



The input–output relationship of a PID controller can be expressed as:



$$u(t) = K_p\left[e(t) + \frac{1}{T_i}\int_0^t e(t)dt + T_d\frac{de(t)}{dt}\right]$$

where u(t) is the output from the controller and e(t)= w(t)− y(t), in which w(t) is the desired set-point(reference input) and y(t) is the plant output. Ti and Td are known as the integral and derivative action time, respectively. The above equation can also be written as:

$$u(t) = K_p e(t) + K_i\int_0^t e(t)dt + \ K_d\frac{de(t)}{dt} + u_0$$

Where,

$$K_i = \frac{K_p}{T_i}\ , K_d = K_p T_d$$

By taking Laplace we can write the transfer function of a continuous-time PID as:

$$\frac{U_{(s)}}{E_{(s)}} = K_p + \frac{K_p}{T_i S} + K_p T_d S$$

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