

PID controllers are an important class of lead-lag compensators that are widely used in industry^[1].

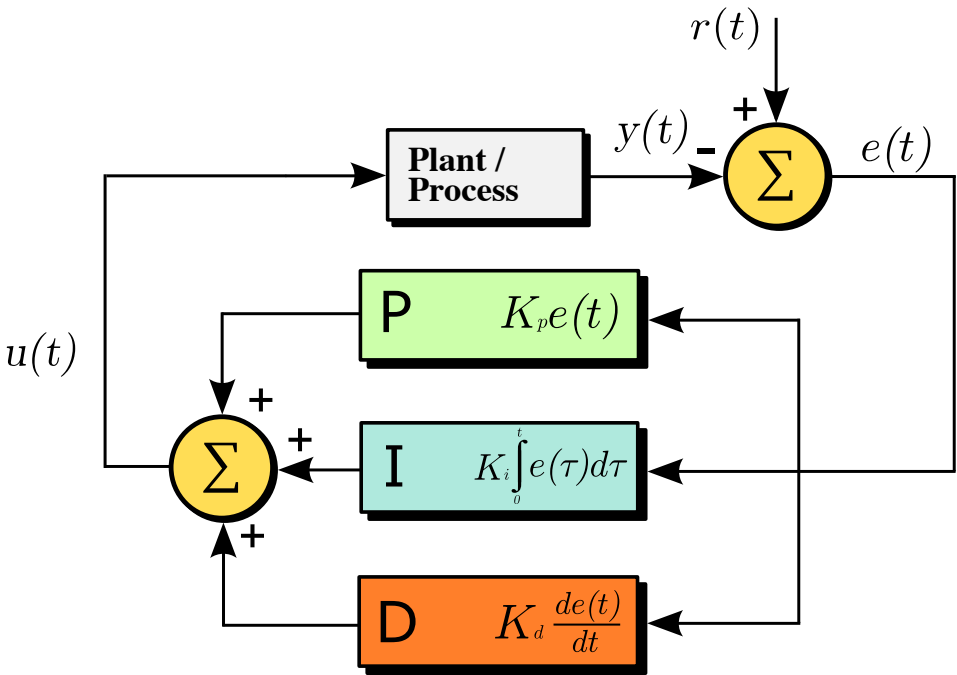


Fig 1. A block diagram of a PID controller in a

feedback loop. $r(t)$ is the desired process value or setpoint (SP), and $y(t)$ is the measured process value (PV). (Source Wikipedia: [PID Controller](#). Image: Arturo Urquiza [\[CC BY-SA 3.0\]](#), [via Wikimedia Commons](#)).

In this chapter we will give an example of tuning a PID controller using Simulink and the Control System Toolbox. For more detail you may wish to refer to the Mathworks Webinar **PID Control Made Easy** below.

The notes give further examples including a definition of the [PID Controller](#) an example of [Zeigler Nichols tuning](#) of manual tuning of a [PID controller](#), and [Analytic Tuning method](#), and Autotuning a PID Compensator with MATLAB.

If we present PID in the exam, you will be expected to use the root-locus methods to locate the poles and zeros.

4.1. PID Control made easy

PID control is ubiquitous. While simple in theory, design and tuning of PID controllers can be difficult and time consuming in practice. The webinar will start with a quick theory primer on PID control. MathWorks engineer Arkadiy Turevskiy will then introduce a simple and straightforward way to quickly design, tune, and implement PID controllers. Through demonstrations, you will learn how to tune PID controllers for control systems in various industries, and how to address various practical considerations such as conversion from continuous to discrete-time domain, output saturation, anti-windup protection, gain-scheduling, and tuning for disturbance rejection.

You may need to Register with the Mathworks to view the video.

[Mathworks Webinar](#)

Files: [Matlab Exchange](#)

4.2. Reference

[1] PID controller, Wikipedia article, URL: http://en.wikipedia.org/wiki/PID_controller. Accessed 25 Feb. 2015.