CHAPTER TWO
PROPORTIONAL-INTEGRAL-DERIVATIVE (PID) CONTROLLER

2.1 Introduction

Feedback control is a control mechanism that uses information from measurements. In a feedback control system, the output is sensed. There are two main types of feedback control systems:

- Positive feedback.
- Negative feedback.

The positive feedback is used to increase the size of the input but in a negative feedback, the feedback is used to decrease the size of the input. The negative systems are usually stable.

A PID is widely used in feedback control of industrial processes on the market in 1939 and has remained the most widely used controller in process control until today. Thus, the PID controller can be understood as a controller that takes the present, the past, and the future of the error into consideration. After digital implementation was introduced, a certain change of the structure of the control system was proposed and has been adopted in many applications. But that change does not influence the essential part of the analysis and design of PID controllers.[1]

PID control is an important ingredient of a distributed control system. The controllers are also embedded in many special purpose control systems. PID control is often combined with logic, sequential functions, selectors, and simple function blocks to build the complicated automation systems used for energy production, transportation, and manufacturing. Many sophisticated control strategies, such as model predictive control, are also organized hierarchically. PID control is used at the lowest level; the multivariable controller gives the set points to the controllers at the lower level. The PID controller can thus be said to be the "bread and butter 't' of control engineering. It is an important component in every control engineer's tool box.

PID controllers have survived many changes in technology, from mechanics and pneumatics to microprocessors via electronic tubes, transistors, integrated circuits. The microprocessor has had a dramatic influence on the PID controller. Practically all PID controllers made today are based on microprocessors. This has given opportunities to provide additional features like automatic tuning, gain scheduling, and continuous adaptation.[2]

2.2 PID Controller Structure

The controller is used in a closed loop unity feedback system according to Fig. 2.1.

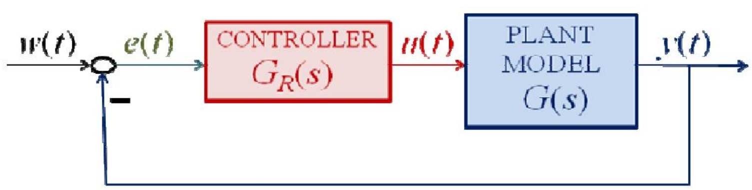

Figure 2.1: Block scheme of closed loop control system

The variable *e* denotes the tracking error, which is sent to the PID controller, *w* is reference variable and *y* is controlled (output) variable.

A proportional–integral–derivative controller (PID controller) is a method of the control loop feedback. This method is composing of three controllers [1]:

1. Proportional controller (PC)
2. Integral controller (IC)
3. Derivative controller (DC)

2.3 PID Representation

The PID controller is quite sophisticated and three different representations can be given. First, there is a symbolic representation (Figure 2.2(a)), where each of the three terms can be selected to achieve different control actions. Secondly, there is a time domain operator form (Figure 2.2(b)), and finally, there is a Laplace transform version of the PID controller (Figure2.2(c)). This gives the controller an s-domain operator interpretation and allows the link between the time domain and the frequency domain to enter the discussion of PID controller performance.

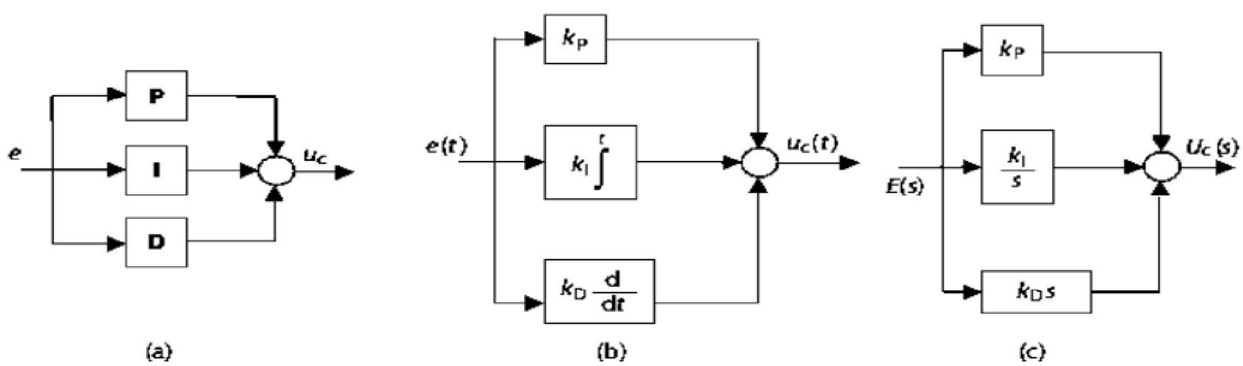

Figure 2.2: PID controller representation

2.4 The Algorithm

We will start by summarizing the key features of the PID controller. The "textbook" version of the PID algorithm is described by:

$$u(t) = K\left(e(t) + \frac{1}{T_i}\int_0^t e(\tau)d\tau + T_d\frac{de(t)}{dt}\right) \quad (2.1)$$

where y is the measured process variable, r the reference variable, u is the control signal and e is the control error ($e = y_{sp} - y$). The reference variable is often called the set point. The control signal is thus a sum of three terms: the P term (which is proportional to the error), the I term (which is proportional to the integral of the error), and the D term (which is proportional to the derivative of the error). The controller parameters are proportional gain K, integral time Ti, and derivative time T_d

2.5 PID Control of Plants

Figure 2-2 shows a PID control of a plant. If a mathematical model of the plant can be derived, then it is possible to apply various design techniques for determining parameters of the controller that will meet the transient and steady-state specifications of the closed-loop system. However, if the plant is so complicated that its mathematical model cannot be easily obtained, then an analytical approach to the design of a PID controller is not possible. Then we must resort to experimental approaches to the tuning of PID controllers. The process of selecting the controller parameters to meet given performance specifications is known as controller tuning. Ziegler and Nichols suggested rules for tuning PID controllers (meaning to set values *Kp, T,*and *T)* based on experimental step responses or based on the value of *K,* that results in marginal stability when only proportional control action is used. Ziegler-Nichols rules, which are briefly presented in

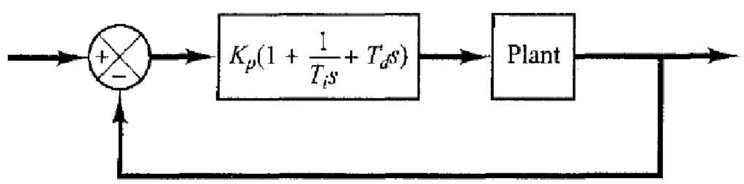

Figure 2.3: PID control of **a** plant.

2.6 PID Terms

2.6.1 Proportional term

The proportional term (*P*) gives a system control input proportional with the error.

Using only *P* control gives a stationary error in all cases except when the system control input is zero and the system process value equals the desired value. In Figure 2.4 the stationary error in the system process value appears after a change in the desired value (ref). Using a too large *P* term gives an unstable system.

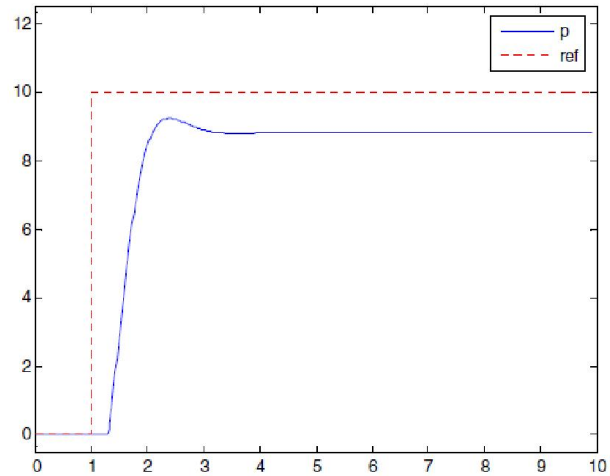

Figure 2.4: Step response *P* controller

2.6.2 Integral term

The integral term (*I*) gives an addition from the sum of the previous errors to the system control input. The summing of the error will continue until the system process value equals the desired value, and this results in no stationary error when the reference is stable. The most common use of the *I* term is normally together with the *P* term, called a *PI* controller. Using only the *I* term gives slow response and often an oscillating system. Figure 2.5 shows the step responses to a *I* and *PI* controller. As seen the *PI* controller response have no stationary error and the *I* controller response is very slow.

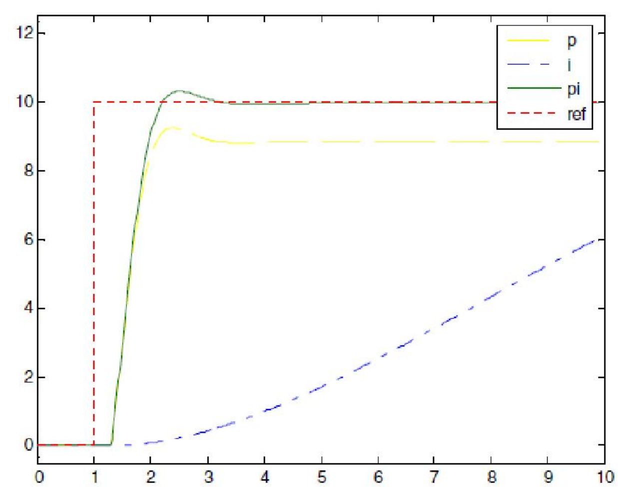

Figure 2.5: Step response I, P and *PI* controller

2.6.3 Derivative term

The derivative term (*D*) gives an addition from the rate of change in the error to the system control input. A rapid change in the error will give an addition to the system control input. This improves the response to a sudden change in the system state or reference value. The *D* term is typically used with the *P* or *PI* as a *PD* or *PID* controller. A to large *D* term usually gives an unstable system. Figure 2.6 shows *D* and *PD* controller responses. The response of the *PD* controller gives a faster rising system process value than the *P* controller. Note that the *D* term essentially behaves as a high pass filter on the error signal and thus easily introduces instability in a system and make it more sensitive to noise.

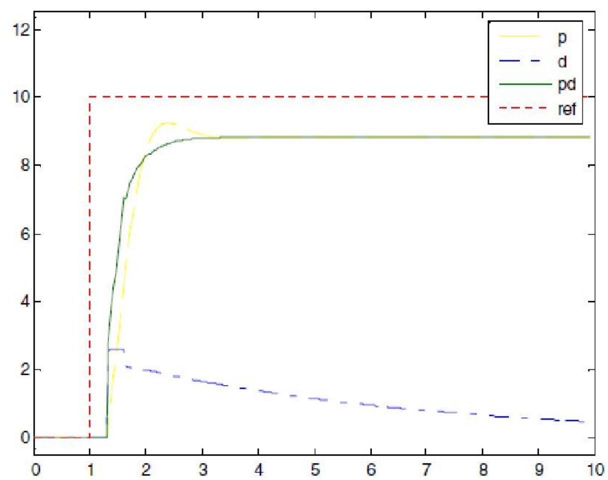

Figure 2.6: Step response *D* and *PD* controller

2.7 Effects of Proportional, Integral and Derivative Action

Proportional control is illustrated in figure 2.7 the controller is given by (2.7) with $T_i = \infty$ and $T_d = 0$. The figure shows that there is always a steady state error in proportional control. The error will decrease with increasing gain, but the tendency towards oscillation will also increase.

Figure 2.8. illustrates the effects of adding integral. It follows from(2.7) that the strength of integral action increases with decreasing integral time T_i. The figure shows that the steady state error disappears when integral action is used. The properties of derivative action are illustrated in figure 2.9 figure 2.9

illustrates the effects of adding derivative action. The parameters K and T_i are chosen so that the closed-loop system is oscillatory. Damping increases with increasing derivative time, but decreases again when derivative time becomes too large. Recall that derivative action can be interpreted as providing prediction by linear extrapolation over the time T_d. Using this interpretation it is easy to understand that derivative action does not help if the prediction time T_d is too large. In Figure 2.9 the period of oscillation is about 6 s for the system without derivative action. Derivative action ceases to be effective when Td is larger than a 1s (one sixth of the period). Also notice that the period of oscillation increases when derivative time is increased.

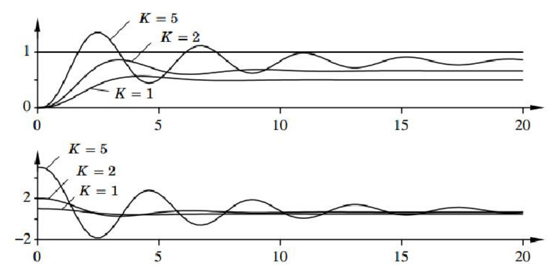

Figure 2.7: Simulation of a closed-loop system with proportional control.

The process transfer function is P(s)=1/(s+1)^3

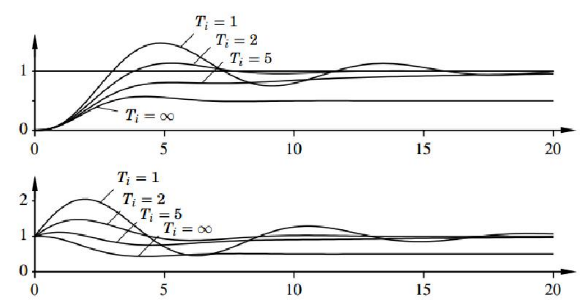

Figure 2.8: Simulation of a closed-loop system with proportional and integral control.

The process transfer function is $P(s)1/(s+1)^3$, and the controller gain is K=1

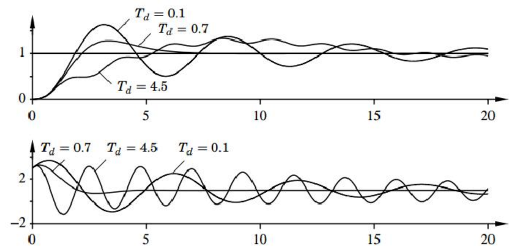

Figure 2.9: Simulation of a closed-loop system with proportional, integral and derivative control.

The process transfer function is $(s)1/(s+1)^3$, the controller gain is K=3, and the integral time is Ti =2.

2.8 PID Controller Tuning Methods

In the previous chapter the basic PID control schemes were discussed. This chapter will present some tuning methods for PID controller. It is interesting to note that more than half of the industrial controllers used today utilize PID or modified PID control schemes. Analog PID controllers are mostly hydraulic, pneumatic, electric, and electronic types or their combinations.

Currently, many of these controllers are transformed into digital forms through the use of microprocessors. Because most of PID controllers are adjusted in field, many different types of tuning methods have been proposed in the literature.

Using these tuning methods delicate and fine tuning of PID controllers can be made in field industry. Also automatic tuning methods of the PID controller have been developed and some of the PID controllers may possess on line automatic tuning capabilities.

Modified forms of PID control, such as I-PD control and two degrees of freedom PID control, are currently in use in industry. Many practical methods for switching (from manual to automatic operation) and gain scheduling are commercially available.

The usefulness of PID controllers lies in their general applicability to most control systems; in the field of process controls systems, it is a well-known fact that the basic and modified PID control schemes have proved their usefulness in providing satisfactory control, although they may not provide optimal control in many given situations.

All general methods for control design can be applied to PID control. A number of special methods that are tailor made for PID control have also been developed these methods are often called tuning methods. Irrespective of the method used, it is essential to always consider the key elements of control, load disturbances, sensor noise, process uncertainty and reference signals. To obtain rational methods for designing controllers it is necessary to define the main purpose of the control system, and the design methods differ with respect to the knowledge of the process dynamics they require. A PI controller is described by two parameters (K and T_i) and a PID controller by three or four parameters (K, T_i, T_d, and N).

2.8.1 Ziegler-Nichols Rules for Tuning PID Controllers

Ziegler and Nichols proposed rules for determining values of the proportional gain K, integral time *T,* and derivative time *T,* based on the transient response characteristics of a given plant. Such determination of the parameters of PID controllers or tuning of PID controllers can be made by engineers on-site by experiments on the plant. (Numerous tuning rules for PID controllers have been proposed since the Ziegler-Nichols proposal. They are available in the literature and from the manufacturers of such controllers).

There are two methods called Ziegler-Nichols tuning rules: the first method and the second method. We shall give a brief presentation of these two methods. **First Method**: In the first method, we obtain experimentally the response of the plant to a unit-step input, as shown in Figure 2.10. If the plant involves neither integrator(~n) or dominant complex-conjugate poles, then such a unit-step response curve look S-shaped, as shown in Figure 2.11.This method applies if the response to may a step input exhibits an S-shaped curve. Such step-response curves may be generated experimentally or from a dynamic simulation of the plant.

The S-shaped curve may be characterized by two constants, delay time L and time constant *T*. The delay time and time constant are determined by drawing a tangent line at the inflection point of the S-shaped curve and determining the intersections of the tangent line with the time axis and line *c(t)* = K, as shown in Figure 2.10. The transfer function C(s)/U(s) may then be approximated by a first-order system with a transport lag as follows:

$$\frac{C(s)}{U(s)} = \frac{Ke^{-ls}}{Ts+1} \qquad (2.2)$$

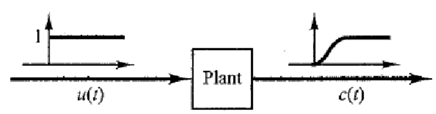

Figure 2.10 Unit-step response of a plan

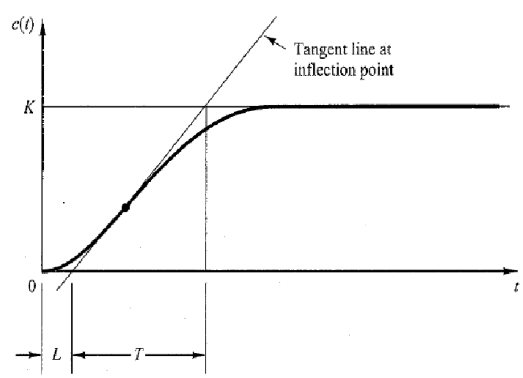

Figure 2.11: S-shaped response curve

Table 2.1**:** Ziegler-Nichols Tuning Rule Based on Step Response of Plant (First Method)

Type of Controller	KP	*Ti*	***Td***
P	T/L	*Inf*	0
PI	0.9T/L	L/0.3	0
PID	1.2T/L	2L	0.5L

Ziegler and Nichols suggested to set the values of *Kp,Ti* and *Td* according to the formula shown in Table 2.1.

Notice that the PID controller tuned by the first method of Ziegler-Nichols rules gives

$$G_c(s) = -K_p\left(1 + \frac{1}{sT_i} + sT_d\right)$$

$$= 1.2\frac{T}{L}(1 + \frac{1}{2Ls} + .5Ls$$

$$= 0.6 * \frac{(s+\frac{1}{L})^2}{s} \qquad (2.3)$$

Thus, the PID controller has a pole at the origin and double zeros at **s** = -1/L,

Second Method: In the second method, we first set T_i = *inf* and T_d = 0. Using the proportional control action only (see Figure 2.12), increase *K,* from 0 to a critical value K_{cr} at which the output first exhibits sustained oscillations.(If the output does not exhibit sustained oscillations for whatever value *K,* may take, then this method does not apply.) Thus, the critical gain *Kc,* and the corresponding period ***Pcr,*** are experimentally determined. Ziegler and Nichols suggested that we set the values of the parameters *K, T,* and T_d according to the formula shown in Table 2.1

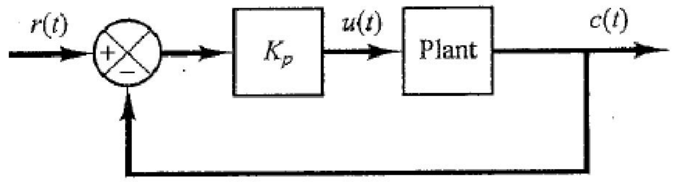

Figure 2.12 Closed-loop system with a proportional controller.

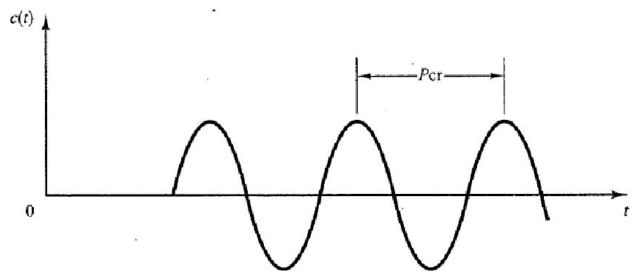

Figure 2.13 Sustained oscillation with period *Pcr.*

2.9 Filtering

Differentiation is always sensitive to noise. This is clearly seen from the Transfer function G(s)=s of a differentiator which goes to infinity for larges. The following example is also illuminating.

EXAMPLE (2.1):

Differentiation amplifies high frequency noise

Consider the signal is:

$$y(t) = \sin t + n(t) = \sin t + a_n \sin\omega_n t \qquad (2.4)$$

where the noise is sinusoidal noise with frequency ω. The derivative of the signal is:

$$\frac{dy(t)}{dt} = \cos t + n(t) = \cos t + a_n \omega \cos \omega_n \mathrm{t} \qquad (2.5)$$

The signal to noise ratio for the original signal is$1/a_n$ but the signal to noise ratio of the differentiated signal is ω/a_n . This ratio can be arbitrarily high if ω is large.

In a practical controller with derivative action it is therefor necessary to limit the high frequency gain of the derivative term. This can be done by implementing the derivative term as

$$D = -\frac{sKT_d}{1+sT_d/N}Y \quad (2.6)$$

Instead of $D = sT_dY$ The approximation given by (2.6) can be interpreted as the ideal derivative sT_d filtered by a first-order system with the time constant T_d/N. The approximation acts as a derivative for low-frequency signal components. The gain, however, is limited to KN. This means that high-frequency measurement noise is amplified at most by a factor KN. Typical values of N are 8 to 20.

Further limitation of the high-frequency gain

The transfer function from measurement y to controller output u of a PID controller with the approximate derivative is

$$C(s) = -K\left(1 + \frac{1}{sT_i} + \frac{sT_d}{1+sT_d/N}\right) \quad (2.7)$$

This controller has constant gain at high frequencies

$$\lim_{s\to\infty} C(s) = -K(1+N) \quad (2.8)$$

It is highly desirable to roll-off the controller gain at high frequencies. This can be achieved by additional low pass filtering of the control signal by

$$F(s) = \frac{1}{\left(1+sT_f\right)^n} \quad (2.9)$$

Where T_f is the filter time constant and n is the order of the filter. The choice of T_f is a compromise between filtering capacity and performance.

The value of T_f can be coupled to the controller time constants in the same way as for the derivative filter above. If the derivative time is used,

$T_f = T_d/N$ is a suitable choice. If the controller is only PI, $T_f = T_i/N$ may be suitable.

The controller can also be implemented as

$$C(s) = -K(1 + \frac{1}{sT_i} + sT_d)\frac{1}{(1+sT_d/N)^2} \quad (2.9)$$

This structure has the advantage that we can develop the design methods for an ideal PID controller and use an iterative design procedure. The controller is first designed for the process $P(s)$. The design gives the controller parameter T_d. An ideal controller for the process $P(s)\frac{1}{(1+sT_d/N)^2}$ is then designed giving a new value of T_d etc. Such a procedure will also give a clear picture of the trade-off between performance and filtering [2].

2.10 Digital PID Controller

Digital PID is commonly used because it is more suitable to design for a complex system for the purpose of reducing cost, and is more immune to noise than an analog PID.

digital controllers are being used in many large and small-scale control systems, replacing the analog controllers. It is now a common practice to implement PID controllers in its digital version, which means that they operate in discrete time domain and deal with analog signals quantized in a limited number of levels.

The trend toward digital rather than analog controls is mainly due to the availability of low-cost digital computers. This controller has been widely used in many different areas such as aerospace, process control, manufacturing, robotics, automation and transportation system. It is one of the most powerful and efficient controller. process measurements are usually of an analog nature: the temperature of the furnace, the rate of flow through a pipe, the pressure of a fluid, etc. These are all analog quantities, infinitely variables not discrete (e.g. counting the number of units passed by on a conveyor belt), the majority of measurements in the industry world are analog.

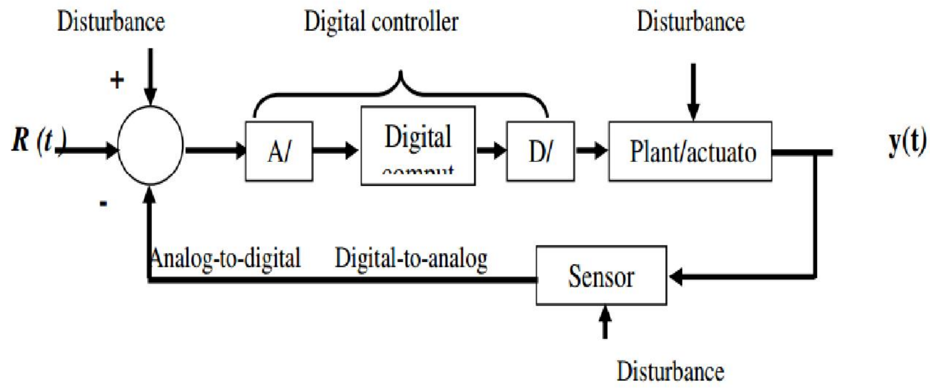

Figure 2.14 a simple digital controller

2.10.1 Analog to digital converter (ADC) and digital to analog converter (DAC)

In order for any digital device to successfully interface with an analog signal, that signal must be digitized by means of analog-to-digital converter or ADC. This section will not endeavor to explore the intricate details of ADC circuitry, but merely to discuss ADC performance in the context of process measurements.

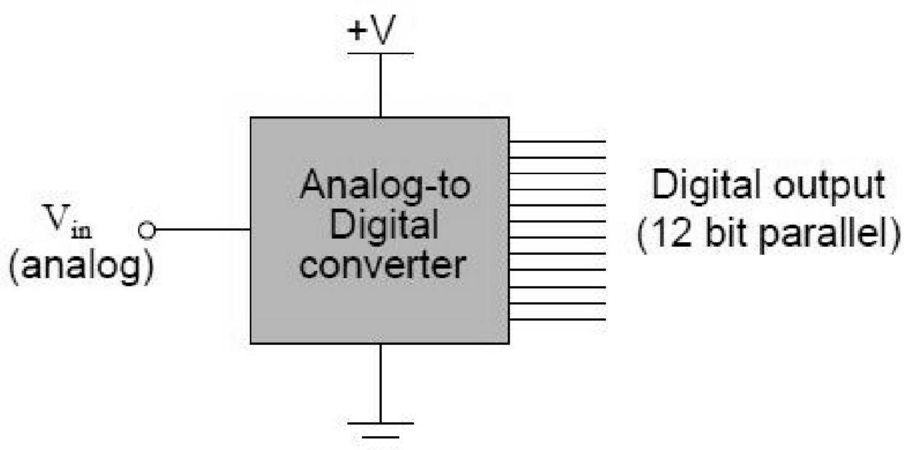

Figure 2.15 ADC

The digital-to-analog converters or DACs, are generally used to produce the analog drive signal required to final control elements.

Digital PID based control system consist of ADC at the input side of the converter comparator based generated error signal in to digital signal.

2.10.2 Digital PID design

Several methods can be used to design a digital PID. One of the methods is to design an analog PID first, then convert the s-domain into the z-domain with appropriate approximation. A digital PID can also be directly designed by the root locus and direct response methods.

- **Conversion from Analog to Digital PID**

The conversion from s-domain into z-domain is quick and easy. The conversion can be done by using difference approximation, ZOH (zero-order hold), bilinear transformation or first-order hold. In this section, the difference approximation equation is derived.

The proportional term in PID can be approximated as:

$$K_p e(k) \qquad (2.10)$$

The backward rectangular rule approximation of integral term in PID:

$$K_I T e(k-1) \qquad (2.11)$$

Also, the backward difference approximation of derivative term in PID:

$$\frac{K_D}{T}[e(k)-e(k-1)\,] \qquad (2.12)$$

However, the integral term requires previous information. Thus, the summation of the three terms becomes, where T denotes the sample period:

$$u(k)=K_P e(k)+a(k)+\frac{K_D}{T}[e(k)-e(k-1) \qquad (2.13)$$

$$a(k)=a(k-1)+K_I T e(k-1) \qquad (2.14)$$

Equation 3.13 is the position algorithm of the present control output. The velocity algorithm for the PID is:

$$u(k-1)=K_P e(k-1)+a(k-1)+\frac{K_D}{T}[e(k-1)-e(k-2) \qquad (2.15)$$

$$a(k-1)=a(k-2)+K_I T e(k-2) \qquad (2.16)$$

By subtracting Equation 3.14 from Equation 3.13, the digital PID is approximated as:

$$u(k) - u(k-1) = K_P[e(k) - e(k-1)] + K_I T e(k-1) + \frac{K_D}{T}[e(k) - 2e(k-1) + e(k-2)] \quad (2.17)$$

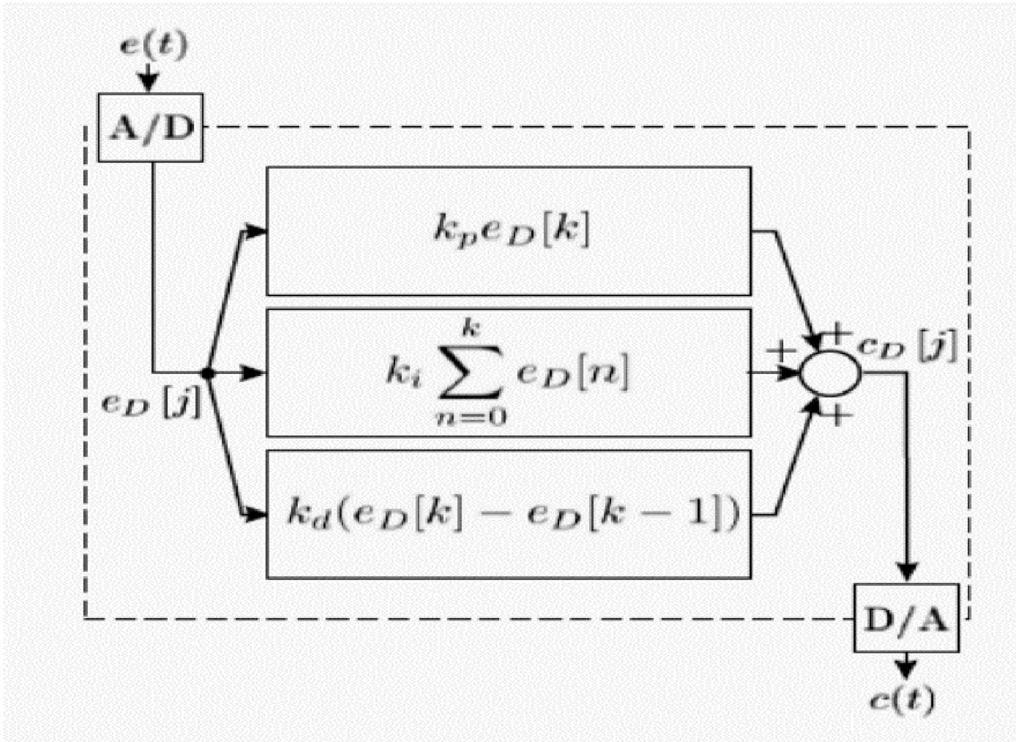

Figure 2.16: Digital PID Controller

➢ Direct root locus design

Root locus design for a digital PID is similar to an analog PID. Basically, the rules for drawing the root locus for both are the same except that stability, frequency and damping ratio are changed.

In terms of stability, it is suggested that the poles be placed in the right-hand plane, and inside the unit circle. The closer the poles are to the origin, the faster the settling time will be. The procedure to design a digital PID is exactly the same as an analog PID, where the poles and zeros work together to shape the root loci to the desired location.

Even though there is no need to physically build a controller algorithm as the analog PID, one needs to consider whether the digital PID is realizable (i.e. the controller does not requires future variables). If the controller is not programmable, the digital PID needs to be redesigned. Modification such as adding another pole inside the unit circle can possibly make the controller realizable.

- **Direct frequency design**

Direct frequency design is useful especially in deadbeat control, a method to make the system meet commands one sample time later than the desired time.

Using direct frequency design, system requirements are first considered, and written in the form of a transfer function. The controller and system transfer function is set equal to the desired transfer function. Then, the proportional, integral and derivative terms can be solved. This is illustrated in Equation below:

$$T(z) = \frac{C(z)}{R(z)} = \frac{D(z)G(z)}{1+D(z)G(z)} \tag{2.18}$$

2.10.3 Tuning for digital PID

The procedure of Ziegler-Nichols tuning for a digital PID is the same as tuning an analog PID. The main difference between them is the sampling time, if the sampling time designed for the digital PID is small compared to system response, an analog tuning method like Ziegler-Nichols works well in a digital PID. However, if the sampling time is larger than the system response, the tuning becomes inaccurate. Thus, it is important to select and design the sampling time wisely, in order to achieve optimum performance.

Table 2.2: Comparison of Digital and Analog PID Controllers

Digital PID Controllers	Analog PID Controllers
More economical because of cheap components and the simple design algorithm	Comparatively expensive due to the complexity of the design algorithm.
Fully integrated and compact.	A large number of operational amplifiers and other components are needed.
High noise immunity.	Noise susceptibility is high.
More flexibility because of the ability to program and reprogram our chip.	Redesigning is required for any change in the system parameters.
High accuracy with faster processing and low power consumption.	Less accurate with more processing time and power consumption is higher.