People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research
University of M'hamed Bougara- Boumerdes

Faculty of Technology
Department of Electrical Systems Engineering

Course handout

BASICS IN DESIGN AND ANALYSIS OF DISCRETE TIME CONTROL SYSTEMS

Dr. Yakhelef Yassine

–2024–

Preface

The present material consists of a content that I have taught for several years. It is entitled as "Basics in design and analysis of discrete feedback control systems"; a title which is chosen to reflect the content in which the focus is to cover the basics and the fundamental knowledge required to be known and understood by the students and engineers of automatics and system's design and control in general but for particularly those interesting with sampled data or digital feedback control systems is a mandatory prerequisite to carry out their study, design and performance analysis of these types of feedback control systems.

Therefore, the content of this handout is principally addressed to all students of both undergraduate program (**Licence**) and graduate program (**Master**) with the intention to not target only the students of automatics but those of Electronics, "Electrotechnique" and other fields can also benefit of this course material as a basic and reference document as long as the sampled data or discrete closed loop control system design and analysis is concerned.

The aim is to give a reference academic material that allows particularly the students to gain the basic and fundamental notions and concepts encountered when studying discrete time control systems with a simple and methodological manner. On the other hand, the student can easily understand, master and implement the different basic methods used to design and analyze the accuracy and stability performance of sampled data (discrete) linear feedback control systems.

In order to attain these objectives, the present handout covers the most topics needed by the students of different study programs and fields and which are discussed according to the following chapters:

Chapter 1: ***General Structure of Discrete Time Control Systems***
Chapter 2: ***Z-Transform of Sampled Signals***
Chapter 3: ***Modeling and Representation of Sampled Data Linear Control Systems***
Chapter 4: ***Performance Analysis of Discrete Linear Feedback Control Systems***
Chapter 5: ***Discrete Linear Time Invariant System Controller Design***

Chapter 1: General Structure of Discrete Time Control System

Digital systems such as computers operate on digital signals; accordingly, the need for handling digital signals is also increased. Due to high-speed processing capabilities of modern digital systems, wide range of applications make use of digital signals, which further accelerate the development of the use of digital signals. Hence, digital control systems have gradually become more prominent in today's industries.

In this chapter we will give the main definitions relating to sampled and discrete signals. We also explore the operation and procedure of obtaining discrete and sampled signals from the corresponding continuous time signals. We also give both mathematical and graphical representation of the basic and standard discrete signals which are widely employed in designing and analyzing a sampled data control systems. We finish the chapter by describing the general structure and block diagram of a typical discrete time feedback control system.

1. Basic Definitions

1.1. What is a signal?

A signal can be defined as a physical quantity that is generated by the evolution of an engineering process. It carries information and measurement about the behavior of that process. For example: the temperature, pressure, voltage, current, sound, etc. It is however important to know that any signal as a physical quantity evolves in time with a variable amplitude. Hence it can be represented as a function where the amplitude takes different values at different corresponding time instants.

1.2. Continuous time (Analog) signal

A continuous time signal is defined as a signal which is both continuous in time and amplitude. Sometimes, a continuous signal is also called analog signal and both names are used interchangeably. The continuous (analog) signal is originally found in nature and is usually measured and generated by the different sensors.

1.3. Discrete time (digital) signal

In any data acquisition system, the measured values of the physical quantities are taken at regular periods of time. Therefore, the time is never being considered as continuous but discrete. As such, a discrete time signal is defined as that obtained by converting a continuous signal at discrete instants of time. This is called discretization of continuous signal; hence the main property of a discrete signal is that it takes finite amplitude values at discrete instants of time.

1.4. Sampled signal

A sampled signal is defined as a continuous signal which is discretized at a regular time step and interval called sampling period. We will come back to explain in more details in the foregoing sections of this chapter.

1.5. Quantification

The quantification is defined as the process and operation of attributing to each amplitude value of the sampled signal a binary number; that is a pattern of bits 0s and 1s.

1.6. Digital signal

A digital signal can be defined as a sampled signal with quantified amplitude.

1.7. Causal signal

A causal signal is defined to be of zero values for negative instants of time. This signal is of paramount importance for the design and analysis of control systems in general. Therefore, regarding the scope of course, we will be interesting only with causal signals and causal systems.

2. Laplace transform: A Review

2.1. Definition

Laplace transform is a mathematical tool used to fundamentally solve linear differential equations. In the context of control system design and analysis, this tool is employed to develop and derive a continuous input-output modeling and representation of the system (process) to be controlled or the whole control system. The use of Laplace transform also makes ease a direct and qualitative analysis of the effect of different environmental variables and parameters of the system's behavior and performance.

Mathematically, the mapping of the continuous time function $f(t)$ of the variable 't' to the frequency domain function $F(s)$ of the complex variable 's' is called Laplace transform. It is defined as [1]:

$$F(s) = \mathcal{L}\{f(t)\} = \int_{-\infty}^{\infty} f(t).e^{-st}\,dt \tag{1.1}$$

For $f(t)$ causal signal, that is: $f(t) = 0,\ for\ \ t < 0$, Laplace transform definition becomes:

$$F(s) = \mathcal{L}\{f(t)\} = \int_{0}^{\infty} f(t).e^{-st}\,dt \tag{1.2}$$

Where:

$\mathcal{L}\{.\}$: is the symbol given to Laplace transform operator.

Expressions (1.1) and (1.2) define respectively two-sided and one-sided Laplace transforms.

In both expressions, we read that Laplace transform of the continuous time signal $f(t)$ is $F(s)$.

Example:

Let $f(t) = e^{-at}$ be continuous and causal signal.

Calculate its Laplace transform.

Answer:

We apply the definition (1.2) on the signal $f(t)$, we get :

$$F(s) = \mathcal{L}\{f(t)\} = \int_{0}^{\infty} f(t).e^{-st}\,dt = \int_{0}^{\infty} e^{-at}.e^{-st}\,dt = \int_{0}^{\infty} e^{-(s+a)t}\,dt$$

$$F(s) = \frac{1}{-(s+a)}\left[e^{-(s+a)t}\right]_0^{\infty}$$

Therefore:

$$\mathcal{L}\{f(t) = e^{-at}\} = F(s) = \frac{1}{(s+a)}$$

2.2. Properties of Laplace Transform

Many important properties are characterizing Laplace transform. If we have:

$$F(s) = \mathcal{L}\{f(t)\} \tag{1.3}$$

We summarize the main of these properties in **Table 1.1** as follows:

Table 1.1 Main properties of Laplace transform

N°	Property	Time signal	Laplace Transform
1	Linearity	$\alpha f_1(t) \pm \beta f_2(t)$	$\alpha F_1(s) \pm \beta F_2(s)$
2	Time Delay	$f(t-m)$	$e^{-ms}.F(s)$
3	Time Advance	$f(t+m)$	$e^{ms}.F(s)$
4	Complex Translation	$e^{+at}.f(t)$	$F(s-m)$
		$e^{-at}.f(t)$	$F(s+m)$
5	Time Scaling	$f(t/a)$	$aF(as)$
6	Derivative	$\dfrac{df(t)}{dt}$	$sF(s) - f(0^+)$
7	Derivative of Order n	$\dfrac{d^{(n)}f(t)}{dt^n}$	$s^nF(s) - \sum_{k=0}^{n-1} s^{n-1-k}.\left.\dfrac{d^{(k)}f(t)}{dt^k}\right\|_{t=0^+}$
8	Integral	$\int_0^t f(\tau)\,d\tau$	$\dfrac{F(s)}{s} + \dfrac{f^{-1}(0^+)}{s}$
9	Initial value theorem	$f(0^+) = \lim_{t\to 0^+} f(t)$	$\lim_{s\to +\infty} sF(s)$
10	Final value theorem	$f(\infty) = \lim_{t\to\infty} f(t)$	$\lim_{s\to 0} sF(s)$

2.3. Laplace transform of basic and familiar signals

In the following table (**Table 1.2**), we mention and summarize the Laplace transform of the most basic signals and widely employed in the design and analysis of continuous time control systems.

Table 1.2 Laplace transform of basic signals

N°	Time domain function : $f(t)$	Laplace Transform: $F(s)$
1	$\delta(t)$	1
2	$u(t)$	$\frac{1}{s}$
3	$t.u(t)$	$\frac{1}{s^2}$
4	$t^n.u(t)$	$\frac{n!}{s^{n+1}}$
5	$e^{-at}.u(t)$	$\frac{1}{s+a}$
6	$t^n.e^{-at}.u(t)$	$\frac{1}{(s+a)^{n+1}}$
7	$\sin(\omega t).u(t)$	$\frac{\omega}{s^2+\omega^2}$
8	$\cos(\omega t).u(t)$	$\frac{s}{s^2+\omega^2}$
9	$e^{-at}\sin(\omega t).u(t)$	$\frac{\omega}{(s+a)^2+\omega^2}$
10	$e^{-at}\cos(\omega t).u(t)$	$\frac{s+a}{(s+a)^2+\omega^2}$

Where:

$\delta(t)$: is the unit impulse (Dirac) signal (function).

$u(t)$: is the unit step signal (function).

3. Sampling of Continuous (Analog) signals

3.1. Definition of Sampling

The sampling can be defined as the process and operation of converting a continuous time (analog) signal $f(t)$ into discrete time signal consisting of a series of impulses of amplitudes determined as the values of the continuous time signal $f(t)$ at the sampling instants [2]. Consequently, the sampling operation produces a sequence of samples denoted by $\{f(kT_s)\}$ from the given analog signal $f(t)$. We express this sequence of samples as:

$$\{f(kT_s)\} = \{f(0), f(1T_s), f(2T_s), \dots., f(kT_s)\} \tag{1.4}$$

We denote the sampled signal as;

$$f^*(t) = f(kT_s) = \{f(kT_s)\}$$
$$= \{f(0), f(1T_s), f(2T_s), \dots, f(kT_s)\} = \{f_0, f_1, f_2, \dots, f_k\} \tag{1.5}$$

Where:

k: is an integer ; $k \in \mathbb{N}$.

T_s: is the sampling period ($T_s > 0$).

kT_s: are defined to be the sampling time instants.

$f(kT_s) = f_k = f^*(t)$: is the notation used to refer the sampled signal obtained to the amplitudes of the continuous time signal $f(t)$ at the corresponding sampling time instants kT_s.

3.2. Principle of Operation of Sampling

To understand the principle of operation of sampling, we need to define the following two important functions:

3.2.1. Dirac (Impulse) Function: $\delta(t)$

The Dirac function $\delta(t)$, also known as Kronecker function, is a non-practical signal. Mathematically, it is defined to be a rectangular signal for which the time width tends to zero and the amplitude (length) tends to infinity with an area equals one. This is expressed as [3]:

$$\delta(t) = \begin{cases} \infty, & if \ \ t = 0 \\ 0, & \forall \ t \neq 0 \\ \int_{-\infty}^{\infty} \delta(t) dt = 1 \end{cases} \tag{1.6}$$

As a shifted version of $\delta(t)$ at the time instant t_0, $\delta(t-t_0)$ is defined by the following:

$$\delta(t-t_0) = \begin{cases} \infty, & if\ \ t = t_0 \\ 0, & \forall\ t \neq t_0 \\ \int_{-\infty}^{\infty} \delta(t)dt = 1 \end{cases} \tag{1.7}$$

3.2.2. Train of Impulses: $\delta_T(t)$

The train of impulses, denoted as $\delta_T(t)$**,** is known to be the sampling function or the sampler in the field of sampling. Mathematically it is defined as:

$$P(t) = \delta_{T_s}(t) = \sum_{k=0}^{\infty} \delta(t - kT_s) \tag{1.8}$$

Graphically, the sampling function is represented, using Matlab, as it is shown in the figure below:

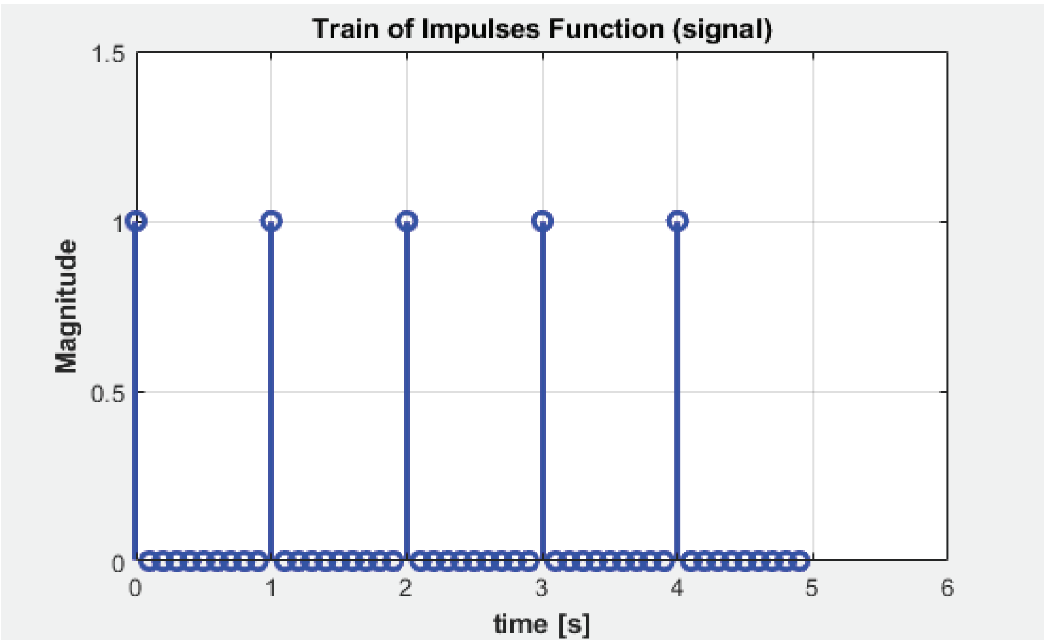


Fig.1.1 Graphical representation of train of impulses signal

Based on the above definitions of the Dirac and the train of impulses function as basic signals, the sampling of an analog signal $f(t)$ can be mathematically described as the product and multiplication of the continuous signal to be sampled by the train of impulses (sampler), $P(t)$.

For $t > 0$, we can write:

$$f^*(t) = f(kT) = f(k) = f(t) \times P(t) = f(t) \times \delta_T(t)$$

$$f^*(t) = f(t) \times \sum_{k=0}^{\infty} \delta(t - kT) = \sum_{k=0}^{\infty} f(kT) \times \delta(t - kT) \qquad (1.9)$$

We can illustrate the sampling operation as it is shown in the following figure:

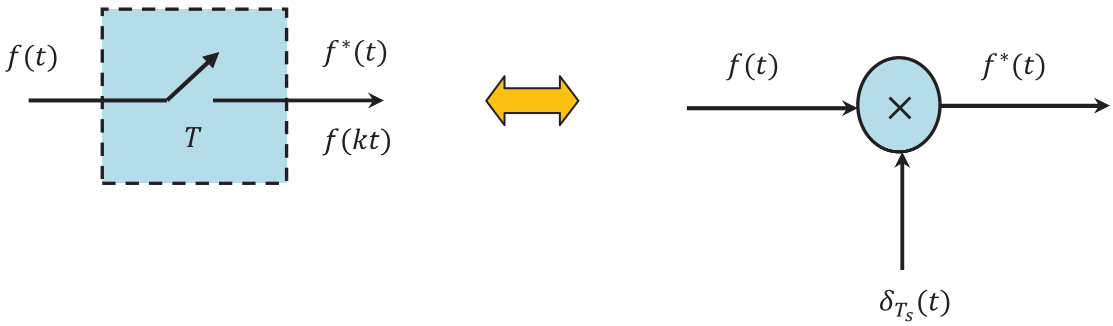


Fig.1.2 Schematic illustration of sampling

3.2.3. Modeling and Representation of a Sampler

For the sake of simplifying the study and analysis of sampled data control systems, the sampling mechanism depicted and illustrated in **Fig.1.2** above is conveniently modeled and represented by an ideal switch that closes at each sampling instant for an infinitesimal time duration ($t \rightarrow 0$) and keep open thereafter for a time duration corresponding to the value of the sampling period T_s. This model given to the sampler is shown in the following figure (**Fig.1.3**).

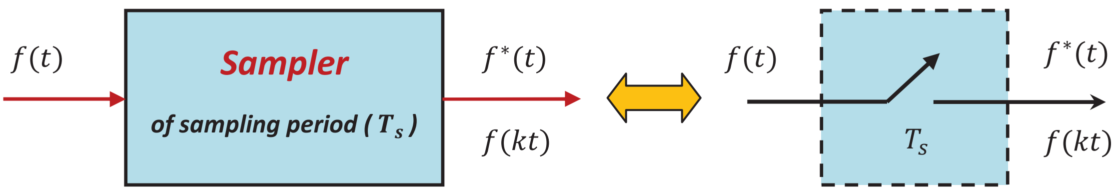


Fig. 1.3 Modeling of a sampler using an ideal switch

3.2.4. Typical standard sampled signals

The outcome of the sampling operation described above is the sampled signal or a discrete time signal. To end up this section, we give some examples of the most familiar and widely used discrete time signals as long as the sampled data control systems are concerned. It is important to notice that only causal signals are being described.

3.2.4.1. Unit impulse discrete time signal: $\delta(kT_s)$

The sampled or discrete time of the impulse or Dirac signal is defined as:

$$\delta(k) = \begin{cases} 1, & if \;\; k = 0 \\ 0, & \forall \;\; k \neq 0 \end{cases} \tag{1.10}$$

However, the discrete time unit impulse signal shifted in time at k_0 is defined as:

$$\delta(k - k_0) = \begin{cases} 1, & if \;\; k = k_0 \\ 0, & \forall \;\; k \neq k_0 \end{cases} \tag{1.11}$$

Graphically, both discrete time unit impulse signal and its time shift version are represented according to the figure below:

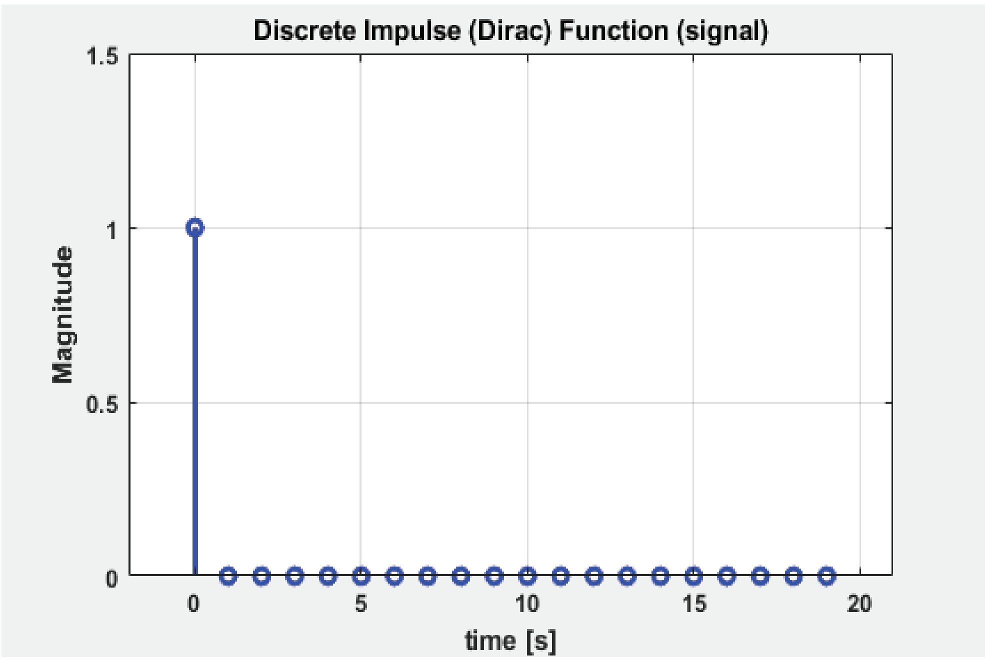


Fig.1.4 Graphical representation of discrete impulse signal

3.2.4.2. Unit step discrete time signal: $u(k)$

When sampled, the unit step signal is defined as:

$$u(k) = \begin{cases} 1, & \forall\ k \geq 0 \\ 0, & \forall\ k < 0 \end{cases} \quad (1.12)$$

The sequence definition of the discrete time (sampled) unit step signal is given as:

$$u(k) = u(kT_s) = u^*(t) = \Big\{ \underbrace{1}_{\uparrow k=0}\ , 1,1,1,1, \dots 1 \Big\} \quad (1.13)$$

Similarly, we can define a time shift at k_0 of the above unit step signal as:

$$u(k - k_0) = \begin{cases} 1, & \forall\ k \geq k_0 \\ 0, & \forall\ k < k_0 \end{cases} \quad (1.14)$$

In the following figure, we show the graphical representation of the sampled unit step signal as well as its time shifted version, where the sampling period is taken arbitrary as: $T_s = 1\ sec.$

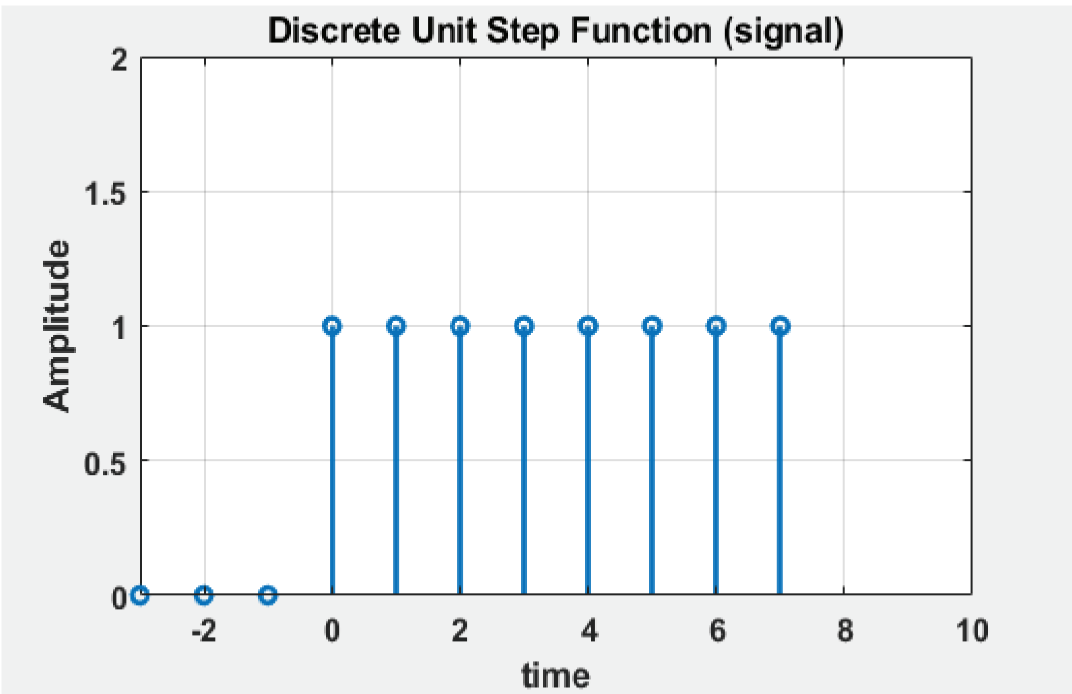


Fig.1.5 Graphical representation of discrete unit step signal

3.2.4.3. Unit Ramp discrete time signal: $r(k)$

The unit ramp sampled signal is defined according to the following description:

$$r(k) = \begin{cases} k, & \forall\ k \geq 0 \\ 0, & \forall\ k < 0 \end{cases} \quad or: \quad r(k) = \begin{cases} kT_s, & \forall\ k \geq 0 \\ 0, & \forall\ k < 0 \end{cases} \tag{1.15}$$

As a sequence, it is defined as:

$$r(k) = r(kT_s) = r^*(t) = \sum_{k=0}^{\infty} r(kT_s)\delta(t - kT_s)$$

$$r(k) = \sum_{k=0}^{\infty} (kT_s)\delta(t - kT_s) = \left\{ \underbrace{0}_{\uparrow k=0}, 1,2,3,4, \dots k \right\} \tag{1.16}$$

When it is time shifted at k_0, it is defined as :

$$r(k - k_0) = \begin{cases} k, & \forall\ k \geq k_0 \\ 0, & \forall\ k < k_0 \end{cases} \tag{1.17}$$

Using Matlab, the graphical representation of a typical discrete ramp signal is shown in **Fig.1.6** below, where the sampling period is taken to be: $T_s = 1\ sec$.

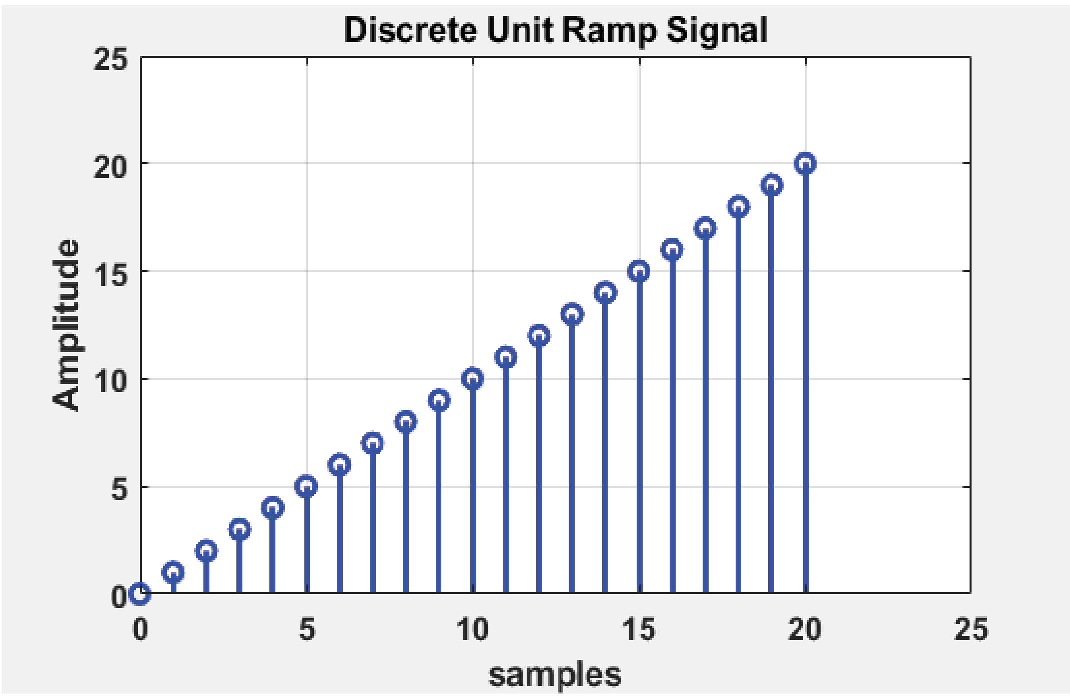


Fig.1.6 Graphical representation of discrete ramp signal

3.2.4.4. Sinusoidal Discrete time signal

If we consider $f(t)$ being unity amplitude continuous time and causal sinusoidal signal defined as:

$$f(t) = sin(t), \forall t \geq 0 \quad (1.18)$$

When sampled at a regular sampling period T_s, the obtained discrete time signal is defined as:

$$f^*(t) = f(kT_s) = f(k) = \sum_{k=0}^{\infty} f(kT_s)\delta(t - kT_s)$$

$$f^*(t) = \sum_{k=0}^{\infty} \sin(\omega kT_s)\delta(t - kT_s) = \begin{cases} \sin(\omega kT_s), & \forall\ k \geq 0 \\ 0, & \forall\ k < 0 \end{cases} \quad (1.19)$$

Using MATLAB, the sampled sinusoidal signal can be depicted in **Fig.1.7**. A comparison is done with the corresponding continuous time sinusoidal signal. We also notice that the sampling period is taken in this case to be: $T_s = 1\ sec$.

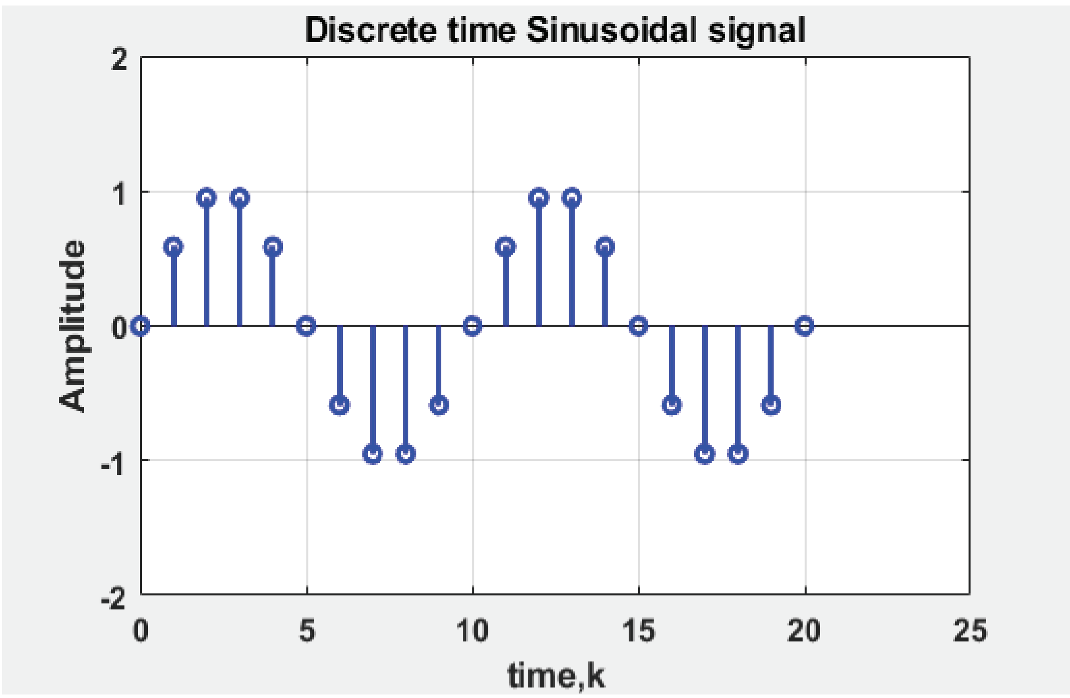


Fig.1.7 Graphical representation of discrete time sinusoidal signal

4. Selecting the Appropriate Sampling Period

It is known in digital signal processing domain that applying sampling operation to convert continuous time (analog) signal into discrete time (digital) signal is accompanied by an inherent error which is attributed to the choice of the sampling period T_s. Unfortunately, this error causes an important loss of information contained in the signal if the value of the sampling period is not conveniently chosen. This problem is also encountered when the design and analysis of digital control system is concerned. In order to solve this problem and achieving a good sampling, Shannon theorem is applied to choose the desired and convenient the sampling period. Using this theorem, it is ensured that the sampling operation is performed with minimum loss of information.

4.1. Shannon theorem

Shannon theorem states that for a continuous time signal to be built and regenerated from a given sequence of samples with a sampling period T_s, the sampling angular frequency defined as $\omega_s = \frac{2\pi}{T_s} = 2\pi f_s$ must be at least two times greater than the angular frequency of the of the continuous time signal. This statement can be interpreted as:

If we assume $f(t)$ to be a continuous time signal of finite energy, hence having a calculated Fourier transform denoted by $F(\omega)$ and illustrated as shown in **Fig.1.8** below**:**

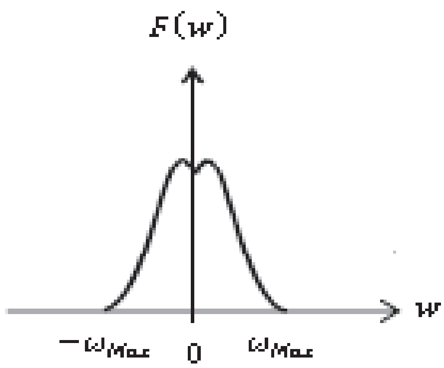


Fig.1.8 Frequency spectral of a given continuous time signal $f(t)$

Where**:**

ω: is the angular frequency of the continuous time signal.

ω_{Max}: is the greatest angular frequency contained in the frequency spectral $F(\omega)$.

Then the Fourier transform $F^*(\omega)$ of the sampled signal $f^*(t)$ can be defined as:

$$F^*(\omega) = \frac{1}{T_s}\sum_{k=-\infty}^{\infty} F(\omega - k\omega_s) \qquad (1.20)$$

We notice that $F^*(\omega)$ is an infinite frequency spectrum obtained from the periodic spectrum of $F(\omega)$ around the sampling angular frequency ω_s.

By applying Shannon theorem, a perfect sampling operation is obtained if the sampling period is chosen such that the following relationship is satisfied:

$$\omega_s \geq 2\omega_{Max} \qquad (1.21)$$

The illustration of the frequency spectrum of the sampled signal $F^*(\omega)$ is shown in the following **Fig. 1.9**.

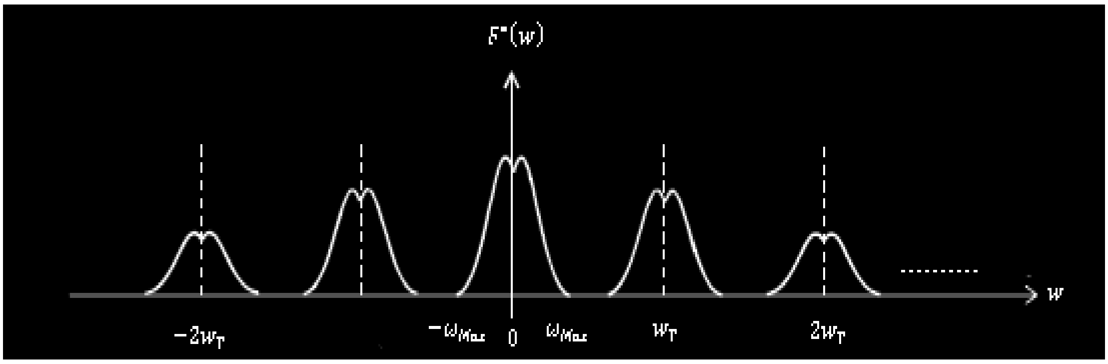


Fig. 1.9 Case of frequency spectrum $F^*(\omega)$ without aliasing phenomenon

The effect of sampling period on the performance and operation of the digital control system can be seen from the following practical issues [2]:

- ☞ When performing sampling, it is required to apply and respect the Shannon theorem.
- ☞ If the sampling period T_s is too small, the robustness and disturbance rejection performance of the feedback control system is highly deteriorated.
- ☞ If on the other hand the sampling period T_s is sufficiently high, high memory storage is meaninglessly needed.

In the following table (**Table 1.3**), it is given the appropriate values of the sampling period regarding some industrial processes subjected to sampling operation [8].

Table 1.3 Typical sampling period values for some processes

Signal	*Recommended sampling period* T_s
Current in Electric Drives	$50 < T_s < 100\ (\mu s)$
Position in Robotics	$0.2 < T_s < 1\ (ms)$
Position in Machine Tools	$0.5 < T_s < 10\ (ms)$
Rate Signal	$1 < T_s < 3\ (s)$
Level Signal	$5 < T_s < 10\ (s)$
Pressure	$1 < T_s < 5\ (s)$
Temperature	$10 < T_s < 45\ (s)$

5. Typical block diagram of discrete time feedback control system

Throughout the previous sections and subsections of this chapter, the main and basic notions and concepts which are used in the context of discrete time (sampled data) control systems are defined and described. However, the crucial idea in all of these concepts is the sampling operation which allows the conversion of analog (continuous time) signals into discrete time signals. Due to the fact that the majority the controlled processes are of analog nature, the use and implementation of these signal converters are principal and mandatory to be able of performing the desired control tasks.

In order to accommodate the functional and operational requirements in terms of signal type processing, any discrete time (digital) control system should incorporate these signal conversion devices. These are called respectively Analog-to-Digital and Digital-to-Analog Converters, which are respectively abbreviated as ADC and DAC. Consequently, the general structure of the block diagram representing a typical discrete time feedback control system is as illustrated in **Fig.1.10**.

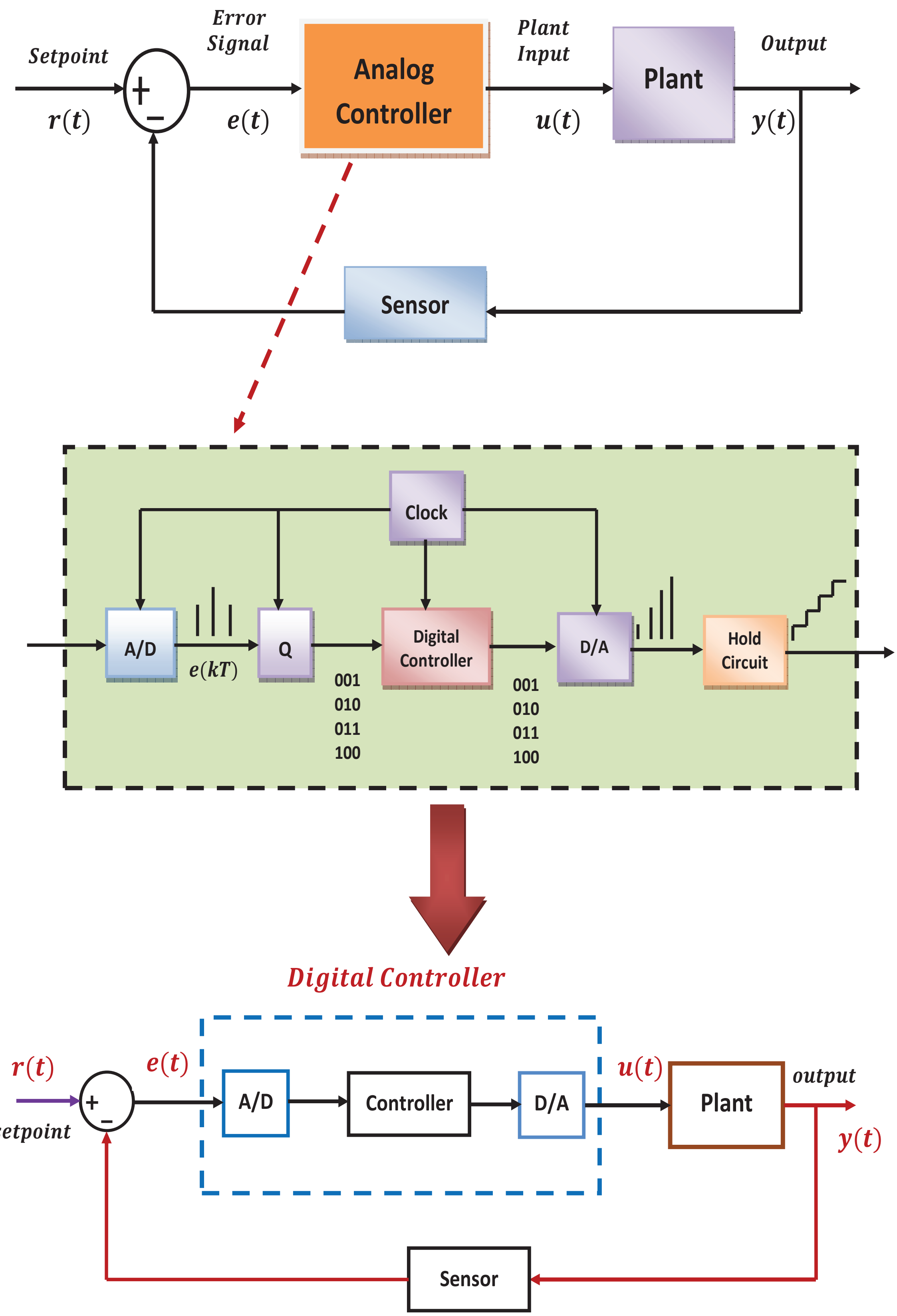


Fig.1.10 Typical block diagram of discrete time feedback control system

5.1. Description of Basic Components of Digital Control system

The operation of the digital control system which is represented by the above typical block diagram is fully accomplished through the following three (03) functional blocks:

- **Analog to Digital Converter (A/D)**:
  This block consists of an electronic device that converts the continuous time (analog) signal to a digital (binary) signal in order to be compatible with the digital controller (micro-processor or microcontroller or any digital-like device) operation. Because, for the computer to respond to any outside event, the data representing this event must be converted into digital form (**1 to 5 volts**), which is being suitably processed and wired to the computer's processor [4].
- **Digital to Analog Converter (D/A)**:
  Digital to Analog Converter (D/A) is an electronic device responsible of converting the output of the digital controller (**pattern of 0s and 1s**) into analog (continuous time) signal to be compatible with the "plant" input (continuous time signal).
- **Digital controller**:
  is the heart of the digital control system, which is an algorithm (software program) implemented to process digital data and generates the appropriate control signal applied at the input of the controlled plant.

The content of the subsequent chapters will be based on the above typical and general structure of discrete time feedback control system.

5.2. Digital Control Systems vs. Analog Control System

Regarding discrete time (digital) control systems, the following advantages over continuous time (analog) control systems can be pointed out:

- As the hardware implementation of analog control systems is a circuit composed of passive and active electronics devices where their properties are highly affected by external factors such as temperature. Hence, the performance of the control system is strongly influenced. However, digital control systems are software based implemented, which means that their dynamic performances are far from the influence of such external affecting factors.

- Since digital control systems are software implemented, consequently, their size is too smaller compared with that of analog control system.
- One of the most important property and advantage of a digital control system over the analog control system consists in its high reproducibility to fulfill the targeted application requirements and specifications. This is because there is an unlimited means of programming. This greatest advantage makes the digital technology more flexible in case of any required modification
- Another valuable advantage of digital control systems consists in their ease of troubleshooting the faults and defected operating conditions compared to the analog control system for which the troubleshooting process is more difficult and laborious.

Despite these great advantages of implementing discrete time (digital) control systems, they however presents some disadvantages such as:

- The mathematical analysis and design of a discrete-time control system is more complex and tedious as compared to continuous-time control system development. This is because of the additional analysis and design parameter, particularly that of the sampling period.
- Because the A/D converter, D/A converter, and the digital computer in reality delay the control signal input (sampling period is not zero), the performance objectives can be more difficult to achieve since the theoretical design approaches usually do not model this small delay.

Chapter 2: Z-Transform of sampled signals

1. Introduction

As it was stated in the previous chapter that a signal is a physical quantity that contains and carries data and information. Similarly, a system is a mechanism that establishes and responds to a relationship between the different affecting input signals. These signals need to be processed in order to study and analyze the system's behavior. As it is the case of continuous time systems, the Laplace transform represents the powerful mathematical tool that allows these study and analysis, in discrete time case, Z transform is instead used.

2. Mathematical Definition of Z Transform

If we assume $f^*(t) = f(kT_s) = f(k)$ to be a given sampled (discrete time) signal, its Z transform denoted as $F(z)$ is defined as a function of the complex variable 'z' by the following mathematical expression:

$$F(z) = \mathcal{Z}\{f(kT_s)\} = \mathcal{Z}\{f^*(t)\} = \sum_{k=\infty}^{\infty} f(kT_s)z^{-k} \qquad (2.1)$$

The expression (2.1) defines what is called two sided Z transform for any discrete time signal. As a particular case which concerns the causal discrete signals and systems, the one sided Z transform is defined as:

$$F(z) = \mathcal{Z}\{f(kT_s)\} = \mathcal{Z}\{f^*(t)\} = \sum_{k=0}^{\infty} f(kT_s)z^{-k} \qquad (2.2)$$

In both definitions, the symbol $\mathcal{Z}$ is used to denote the Z transform operator.

3. Derivation of Z transform

Before tackling the mathematical derivation of Z transform expression, we introduce the following definition of Laplace transform of discrete time (sampled) signal.

3.1. Laplace Transform of sampled signal

We consider $f^*(t)$ being the sampled signal of the continuous time signal $f(t)$, we accept without proof the definition of Laplace transform of the signal $f^*(t)$ denoted by $F^*(s)$ and given by:

$$F^*(s) = L\{f^*(t)\} = \sum_{k=0}^{\infty} f(kT_s)e^{-skT_s} \tag{2.3}$$

$F^*(s)$: is also called stared Laplace transform.

In the following figure (**Fig.2.1**), we give an illustration of how the Laplace transform $F^*(s)$ is calculated:

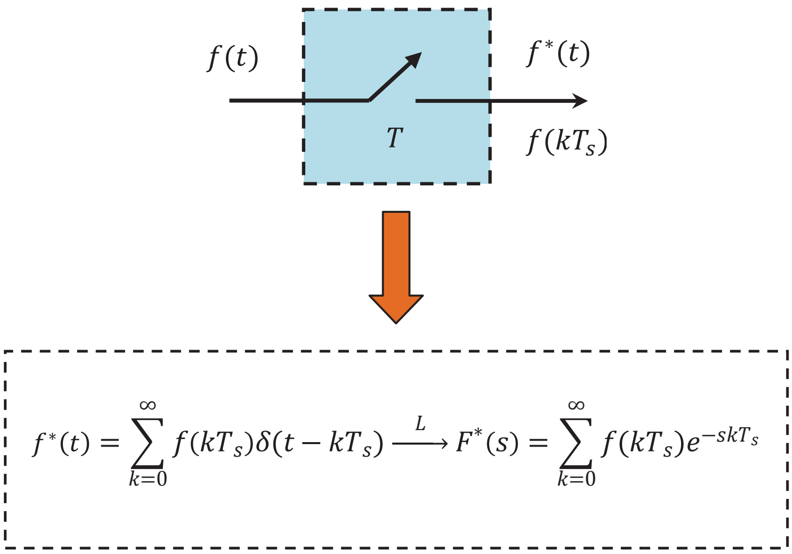


Fig. 2.1 Illustration showing the calculation of stared Laplace transform $F^*(s)$

After we have introduced this concept of Laplace transform of sampled signal, we are at the level of deriving the expression defining the Z transform.

Let us consider $f^*(t)$ being the causal sampled signal expressed as:

$$f^*(t) = f(kT_s) = f(t).\delta_{T_s}(t) = f(t).\sum_{k=0}^{\infty} \delta(t-kT_s)$$

$$f^*(t) = f(kT_s) = \sum_{k=0}^{\infty} f(kT_s).\delta(t-kT_s) \tag{2.4}$$

By applying stared Laplace transform given by (2.3) on expression (2.4), we get:

$$F^*(s) = L\{f^*(t)\} = \sum_{k=0}^{\infty}\left[\sum_{k=0}^{\infty} f(kT_s).\delta(t-kT_s))\right]e^{-skT_s}$$

$$F^*(s) = \sum_{k=0}^{\infty} f(kT_s)e^{-skT_s} \tag{2.5}$$

Then, we use the variable transformation given by:

$$z = e^{sT_s} \tag{2.6}$$

The equation (2.5) becomes:

$$F^*(s) = \sum_{k=0}^{\infty} f(kT_s)z^{-k} = F(z) \tag{2.7}$$

Expression (2.7) is exactly the definition of one sided Z transform given previously by (2.2).

4. Some Properties of Z transform

Z transform, as Laplace transform, possess many properties, however in this section, we will focus on the most and widely used ones in the field of designing and analyzing discrete time (sampled data) control systems.

4.1. Linearity

The linearity property of Z transform is stated as follows:

Consider $f_1(k)$ and $f_2(k)$ being two discrete time (sampled) signals, for which:

$$\begin{cases} F_1(z) = Z\{f_1(k)\} \\ F_2(z) = Z\{f_2(k)\} \end{cases}$$

If α, β are real numbers, then:

$$Z\{\alpha f_1(k) \pm \beta f_2(k)\} = \alpha Z\{f_1(k)\} \pm \beta Z\{f_2(k)\} = \alpha F_1(z) \pm \beta F_2(z) \tag{2.8}$$

4.2. Time Translation

If $f(k)$ is discrete time signal and $F(z)$ its Z transform, let $m \in \mathbb{N}$ and $T_s > 0$, then two types of time translation on $f(k)$ are distinguishable:

4.2.1. Time delay

$$Z\{f(k-mT_s)\} = z^{-m}Z\{f(k)\} = z^{-m}.F(z) \tag{2.9}$$

4.2.2. Time Advance

$$Z\{f(k+mT_s)\} = z^m\left[Z\{f(k)\} - \sum_{k=0}^{m-1} f(k).z^{-k}\right]$$

$$Z\{f(k+mT_s)\} = z^m\left[F(z) - \sum_{k=0}^{m-1} f(k).z^{-k}\right] \tag{2.10}$$

4.3. Time Multiplication

If $f(k)$ is discrete time signal and $F(z)$ its Z transform, the Z transform of the product time and the signal $f(k)$ is obtained as follows:

$$Z\{kf(k)\} = -zT_s\frac{d}{dz}[Z\{f(k)\}] = -zT_s\frac{dF(z)}{dz} \tag{2.11}$$

4.4. Discrete Convolution Theorem

If $f(k)$ is discrete time signal and $F(z)$ its Z transform, the Z transform of the time convolution of the two signals $f_1(k)$ and $f_2(k)$ is calculated as [8]:

$$Z\{f_1(k) * f_1(k)\} = F_1(z).F_1(z) \tag{2.12}$$

4.5. Initial value theorem

Let $f(k) = f(kT) = f^*(t)$ be a discrete time (sampled) signal and $F(z)$ its Z transform. If we are in the frequency domain and we do not know the time domain expression of the discrete signal, the calculation of its initial value is performed using the initial value theorem which is stated as follows:

$$f(0) = \lim_{k\to 0} f(k) = \lim_{z\to\infty} F(z) \tag{2.13}$$

4.6. Final Value Theorem

Let $f(k) = f(kT) = f^*(t)$ be a discrete time (sampled) signal and $F(z)$ its Z transform. If we are in the frequency domain and we do not know the time domain expression of the discrete signal, the calculation of its final value is performed using the final value theorem which is stated as follows:

$$f(\infty) = \lim_{k\to\infty} f(kT) = \lim_{z\to 1}(1 - z^{-1})F(z) \qquad (2.14)$$

5. Z transform of the most familiar sampled signals

In the following table (**Table 2.1**), we summarize the Z transform of the most known and used discrete time signals regarding the design and analysis of sampled data control systems.

Table 2.1 Z transform of the most familiar discrete time signals

Continuous time signal: $f(t)$	**Discrete time signal: $f(kT_s)$**	**Z transform $F(z)$**
$\delta(t)$	$\delta(kT_s) = \delta(k)$	1
$u(t)$	$u(kT_s) = u(k)$	$\frac{z}{z-1}$
t	kT_s	$\frac{zT_s}{(z-1)^2}$
t^2	$(kT_s)^2$	$\frac{z(z+1){T_s}^2}{(z-1)^3}$
t^3	$(kT_s)^3$	$\frac{z(z^2+4z+1){T_s}^2}{(z-1)^4}$
e^{bt}	$e^{bkT_s} = a^k$	$\frac{z}{z-a} = \frac{z}{z-e^{bT_s}}$
a^t	a^{kT_s}	$\frac{z}{z-a^{T_s}}$
$1 - e^{\alpha t}$	$1 - a^k$	$\frac{(1-a)z}{(z-1)(z-a)}$
$e^{\alpha t} - e^{\beta t}$	$a^k - b^k$	$\frac{(a-b)z}{(z-a)(z-b)}$
$te^{-\alpha t}$	$kT_s a^k$	$\frac{azT_s}{(z-a)^2}$

$\sin(\omega_n t)$	$\sin(\omega_n kT_s)$	$\dfrac{\sin(\omega_n T_s)z}{z^2 - 2\cos(\omega_n T_s)\, z + 1}$
$\cos(\omega_n t)$	$\cos(\omega_n kT_s)$	$\dfrac{z[z - \cos(\omega_n T_s)]}{z^2 - 2\cos(\omega_n T_s)\, z + 1}$

6. Calculation Methods of Z transform

We distinguish two approaches to be used in calculating the Z transform of given signal. These are time domain and frequency domain approaches.

6.1. Time domain Approach

This approach considers the calculation of Z transform of a sampled signal directly using the definition of Z transform. That is, if $f(k) = f^*(t)$ is the sampled signal of the continuous time signal $f(t)$, its Z transform is calculated as:

$$F(z) = Z\{f(kT_s)\} = Z\{f^*(t)\} = L\{f^*(t)\}|_{z=e^{sT_s}}$$

$$F(z) = F^*(s)|_{z=e^{sT_s}} = \sum_{k=0}^{\infty} f(kT_s)z^{-k} \qquad (2.15)$$

This approach can be illustrated using the following diagram of **Fig.2.2**:

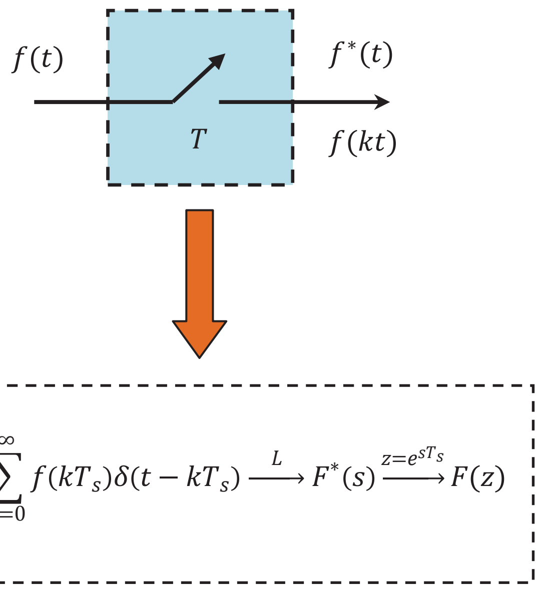


Fig.2.2 Diagram illustrating calculation of F(z) using time domain approach

Example:

Consider the sampled unit step signal $u(kT_s)$, we need to calculate its Z transfer function using the explained time domain approach.

Solution:

The use of time domain approach assumes the knowledge and availability of the sampled signal expression in the time domain.

The sampled unit step signal is defined as:

$$u(kT_s) = \begin{cases} 1, & \forall\ k \geq 0 \\ 0, & \forall\ k < 0 \end{cases}$$

Using (2.7), the Z transform of $u(k)$ is:

$$U(z) = Z\{u(kT_s)\} = U^*(s)|_{z=e^{sT_s}} = \sum_{k=0}^{\infty} u(kT_s).z^{-k} = \sum_{k=0}^{\infty} 1.z^{-k} = \sum_{k=0}^{\infty} z^{-k}$$

The latter expression is convergent geometric sequence of base: $r = z^{-1}$, it results:

$$U(z) = \frac{1}{1-z^{-1}} = \frac{z}{z-1}, \qquad |z^{-1}| < 1$$

6.2. Frequency domain Approach

This approach assume that we only know the Laplace transform $F(s)$ of the continuous time signal $f(t)$, then we can calculate the Z transform $F(z)$ of the discrete time signal $f^*(t) = f(kT_s) = f(k)$.

In order to do so, several methods can be found; the most widely used are:

- ☞ Residues Method.
- ☞ Partial Fraction expansion Method.
- ☞ Polynomial Division Method.

We will explain the use of all them due to their importance and extensively use in both signal processing and digital control system design and analysis.

6.2.1. Residues Method

The Residues method of calculating Z transform of any sampled signal described by the Laplace transform of the corresponding continuous time signal is used as follows:

Given $F(s) = \mathcal{L}\{f(t)\}$, then, Z transform of $f^*(t)$ is calculated by:

$$F(z) = \sum_{p_i} R_i = \sum_{p_i} \left[Residues\ of\ \frac{F(s)}{1 - e^{sT_s} z^{-1}} \right] \Bigg|_{s=p_i} \qquad (2.16)$$

Where:

p_i: are the poles of $F(s)$.

R_i: are the residues corresponding to the poles p_i.

According to expression (2.16), the use of Residues method depends on two observed cases:

❖ **Case 1: all the poles p_i are simple and real**

If $N(s)$ and $D(s)$ are respectively the Numerator and Denominator of $F(s)$, that is we can write:

$$F(s) = \frac{N(s)}{D(s)} \qquad (2.17)$$

When all the poles p_i of $F(s)$ are simple and real, the Residue corresponding to the i^{th} pole is calculated as:

$$R_i = \left[Residue\ of\ \frac{F(s)}{1 - e^{sT_s} z^{-1}} \right] \Bigg|_{s=p_i} = \frac{N(p_i)}{D'(p_i)} \cdot \frac{1}{1 - e^{sT_s} z^{-1}} \Bigg|_{s=p_i} \qquad (2.18)$$

Where:

$$D'(p_i) = \frac{d[D(s)]}{ds} \Bigg|_{s=p_i} \qquad (2.19)$$

Illustration:

We give the following example to illustrate the application of Residues method in this first case of simple and real poles of the known Laplace transform function.

Let: $F(s) = \frac{1}{s}$, we want to calculate $F(z)$.

Answer:

We can write: $F(s) = \frac{N(s)}{D(s)} = \frac{1}{s}$, this means that: $\begin{cases} N(s) = 1 \\ D(s) = s \end{cases}$

The poles of $F(s)$ are the roots of $D(s)$; that is:

$$D(s) = 0 \Longrightarrow s = 0 \Longrightarrow s = p_1 = 0.$$

It results that $F(s)$ has only one simple and real pole.

Therefore:

$$D'(s = p_1) = \left.\frac{d[D(s)]}{ds}\right|_{s=p_1} = 1$$

Using (2.18), we obtain:

$$R_1 = \frac{N(p_i)}{D'(p_i)} \cdot \frac{1}{1 - e^{sT_s}z^{-1}} = \frac{1}{1} \cdot \left.\frac{1}{1 - e^{sT_s}z^{-1}}\right|_{s=0} = \frac{1}{1 - z^{-1}} = \frac{z}{z - 1}$$

Since $F(s)$ has only one pole, it exists only one residue, consequently, the corresponding Z transform is:

$$F(z) = \sum_{p_i} R_i = R_1 = \frac{1}{1 - z^{-1}} = \frac{z}{z - 1}$$

❖ Case 2: Repeated poles

When the Laplace transform $F(s)$ has a repeated pole, say p_i, of multiplicity factor « m », the Residue corresponding to this pole is calculated using the following general formula:

$$R_i = \frac{1}{(m-1)!} \frac{d^{m-1}}{ds^{m-1}} \left[(s - p_1)^r \frac{F(s)}{1 - e^{sT_s}z^{-1}} \right] \Bigg|_{s=p_1} \qquad (2.20)$$

Illustration:

Calculate the Z transform corresponding to the Laplace function given by:

$$F(s) = \frac{1}{s^2(s + 1)}$$

Answer:

We have: $F(s) = \frac{1}{s^2(s+1)} = \frac{N(s)}{D(s)}$, which implies that : $\begin{cases} N(s) = 1 \\ D(s) = s^2(s+1) \end{cases}$

We start by determining the poles of $F(s)$; this corresponds to solve the equation:

$$D(s) = 0 \Longrightarrow s^2(s+1) = 0 \Longrightarrow \begin{cases} s^2 = 0 \\ s+1 = 0 \end{cases}$$

$$\Longrightarrow \begin{cases} s = p_1 = 0 \ ,(repeated\ pole\ of: m = 2) \\ s = p_2 = -1, \qquad simple\ and\ real\ pole \end{cases}$$

We calculate the residues corresponding to the repeated pole. Using (2.18) we get:

$$R_1 = \frac{d}{ds}\left[s^2 . \frac{1}{s^2(s+1)} . \frac{1}{(1 - e^{sT_s}z^{-1})}\right]\Bigg|_{s=0}$$

$$= \left[\frac{-(1 - e^{sT_s}z^{-1}) + (s+1)T_s e^{sT_s}z^{-1}}{(s+1)^2(1 - e^{sT_s}z^{-1})^2}\right]\Bigg|_{s=0}$$

$$R_1 = \frac{-(1 - z^{-1}) + T_s z^{-1}}{(1 - z^{-1})^2} = -\frac{z}{z-1} + \frac{T_s z}{(z-1)^2}, \quad \forall\, T_s > 0$$

Using (2.18), the residue corresponding to the simple and real pole is:

$$R_2 = \frac{1}{D'(s)} \frac{1}{(1 - e^{sT_s}z^{-1})}\Bigg|_{s=-1} = \frac{1}{2s(s+1) + s^2} \frac{1}{(1 - e^{sT_s}z^{-1})}\Bigg|_{s=-1}$$

$$R_2 = \frac{1}{(1 - e^{-T_s}z^{-1})} = \frac{z}{z - e^{-T_s}}, \quad \forall\ T_s > 0$$

Regarding the two types of poles, the following end result can be obtained:

$$Z\left\{\frac{1}{s^2(s+1)}\right\} = R_1 + R_2 = -\frac{z}{z-1} + \frac{T_s z}{(z-1)^2} + \frac{z}{z - e^{-T_s}}, \quad \forall\, T_s > 0$$

6.2.2. Partial Fraction Expansion Method

This method is the most familiar and widely used particularly in the field of control system design and analysis. It consists of decomposing the Laplace transform function into simple fractions of known and easy determined Z transform. After we use the Z transform properties, it is possible to calculate Z transform of the originally given Laplace transform.

Assuming that Laplace transform function possesses both types of simple real poles and repeated poles, the following general formula is used to expand it into simple partial fractions.

$$F(s)=\frac{N(s)}{D(s)}=\frac{N(s)}{(s-p_1)^m.\prod_{j=m+1}^{n}(s-p_j)}$$

$$F(s)=\underbrace{\sum_{i=1}^{m}\frac{C_{1i}}{(s-p_1)^{m+1-i}}}_{\substack{\text{Simple Fractions}\\ \text{Corresponding to}\\ \text{Repeated poles}}}+\underbrace{\sum_{j=m+1}^{n}\frac{C_j}{(s-p_j)}}_{\substack{\text{Simple Fractions}\\ \text{Corresponding to}\\ \text{Simple Real poles}}} \tag{2.21}$$

Where:

$F(s)$: represents Laplace transform fractional function of order “n”.

p_1: is the repeated pole of $F(s)$ of multiplicity factor “m”.

p_j: is the j^{th} simple and real pole of $F(s)$.

C_{1i} and C_j are unknown coefficients to be determined.

C_{1i} are the coefficients corresponding to the repeated poles; these are calculated using the following formula:

$$C_{1i}=\frac{1}{(i-1)!}\frac{d^{(i-1)}}{ds^{(i-1)}}(s-p_1)^m F(s)|_{s=p_1}\ ,\quad for: i=1,2,3,\dots.,m \tag{2.22}$$

C_j: are the coefficients corresponding to the simple real poles, these are calculated using the following formula:

$$C_j=(s-p_j)F(s)\big|_{s=p_j} \tag{2.22}$$

Example:

Let’s take the Laplace transform function of the previous illustration and use partial fraction expansion to calculate its Z transform.

Answer:

By observing the s-function $F(s)$, it has the pole $p_1 = 0$ which repeated twice and one single pole $p_3 = -1$, which is simple and real. Using partial fraction expansion defined in general by (2.19), we obtain:

$$F(s) = \frac{1}{s^2(s+1)} = \frac{C_{11}}{s} + \frac{C_{12}}{s^2} + \frac{C_3}{(s+1)}$$

Using (2.20) and (2.21), the coefficients C_{11}, C_{12} and C_3 are respectively calculated as:

$$C_{11} = \frac{d}{ds}[s^2\mathrm{F(s)}]|_{s=0} = \frac{d}{ds}\left[\frac{1}{s+1}\right]\Big|_{s=0} = -1$$

$$C_{12} = [s^2\mathrm{F(s)}]|_{s=0} = \left[\frac{1}{s+1}\right]\Big|_{s=0} = 1$$

$$C_3 = [(s+1)\mathrm{F(s)}]|_{s=-1} = \left[\frac{1}{s^2}\right]\Big|_{s=-1} = 1$$

Hence :

$$F(s) = \frac{1}{s^2(s+1)} = \frac{-1}{s} + \frac{1}{s^2} + \frac{1}{(s+1)}$$

By applying Z transform, we have:

$$F(z) = Z[F(s)] = Z\left[\frac{-1}{s} + \frac{1}{s^2} + \frac{1}{(s+1)}\right] = -TZ\left[\frac{1}{p}\right] + TZ\left[\frac{1}{p^2}\right] + TZ\left[\frac{1}{(p+1)}\right]$$

Using linearity property of Z transform, it turns out that:

$$F(z) = -Z\left[\frac{1}{s}\right] + Z\left[\frac{1}{s^2}\right] + Z\left[\frac{1}{(s+1)}\right]$$

To finish, we still only to use the correspondence table between Laplace transform and Z transform of the basic signals to find:

$$F(z) = -\frac{z}{z-1} + \frac{T_s z}{(z-1)^2} + \frac{z}{z-e^{-T_s}}, \quad \forall\, T_s > 0$$

Obviously, identical results are obtained using either Residues or Partial fraction expansion methods.

7. Inverse Z Transform

7.1. Definition

While Z transform represents a mathematical tool applied on discrete time (sampled) signal which allows us to transform the work domain from time domain into frequency domain, inverse Z transform is the reverse operation; in other words, it is a transformation that allows us to obtain the discrete time domain representation of a sampled signal from the knowledge of its frequency domain representation.

The inverse Z transform mechanism, denoted by the symbol $\mathcal{Z}^{-1}$, is described as:

$$if: \;\; F(z) = \mathcal{Z}\{f^*(t)\} = \mathcal{Z}\{f(kT)\}$$

$$then:$$

$$f^*(t) = f(kT) = \mathcal{Z}^{-1}[F(z)] \tag{2.23}$$

7.2. Calculation of Inverse Z transform

The same methods which are explained earlier and used to calculate Z transform of discrete time signals are also employed in calculating the inverse Z transform with only a little bit difference in their formulation and application procedure. These are as follows:

7.2.1. Residues Method

If it is known the Z transform $F(z) = Z\{f^*(t)\}$, the sampled (discrete time) signal $f^*(t) = f(kT)$ is obtained back using residues method according to the following formula:

$$f^*(t) = f(kT) = \sum_{p_i} R_i = \sum_{p_i} [Residue\ de\ z^{k-1}.F(z)]|_{z=p_i} \tag{2.24}$$

Where:

p_i: are the poles of the Z transform function $F(z) = \frac{N(z)}{D(z)}$

R_i: is the residue corresponding to the pole p_i of multiplicity factor "m" , which is calculated using the following general formula :

$$R_i = \frac{1}{(m-1)!}\frac{d^{(m-1)}}{dz^{(m-1)}}[(z-p_i)^m\, z^{k-1}.F(z)]|_{z=p_i} \qquad (2.24)$$

To illustrate the use of this method, we consider the following example.

Example:

Consider the Z transform defined by the function $F(z) = \frac{T_s.z}{(z-1)^2}$, $\forall\, T_s > 0$.

Calculate the inverse Z transform of $F(z)$?

Answer:

First of all, we need to determine the poles of $F(z)$. we have:

$$(z-1)^2 = 0 \Longrightarrow z = p_1 = 1: repeated\ pole\ of\ m = 2$$

By applying the Residues defined by (2.24), we get:

$$f^*(t) = f(kT) = \sum\nolimits_{p_i} R_i = R_1$$

With :

$$R_1 = [Residue\ de\ z^{k-1}.F(z)]|_{z=p_i} = \frac{1}{(m-1)!}\frac{d^{(m-1)}}{dz^{(m-1)}}[(z-p_i)^m\, z^{k-1}.F(z)]|_{z=p_i}$$

For $m = 2$:

$$R_1 = \frac{1}{(2-1)!}\frac{d^{(2-1)}}{dz^{(2-1)}}[(z-1)^2\, z^{k-1}.F(z)]|_{z=1}$$

$$R_1 = \frac{d}{dz}\left[(z-1)^2\, z^{k-1}.\frac{T_s z}{(z-1)^2}\right]\Bigg|_{z=1} = \frac{d}{dz}[T_s z^k]|_{z=1} = kT_s[z^{k-1}]|_{z=1} = kT_s$$

Therefore :

$$f^*(t) = f(kT_s) = \mathcal{Z}^{-1}[F(z)] = R_1 = kT_s, \quad \forall\, T_s > 0$$

Which represents the sampled unit ramp signal.

7.2.2. Partial Fraction Expansion Method

This method, as it is previously explained, is based on decomposing the fractional z-function into simple elementary fractions where each partial fraction is a Z transform of one of the basic and familiar sampled signals. However, the use of this method in calculating the inverse Z transform of $F(z)$ is done according to the following procedure:

Step 1: we construct the function: $\frac{F(z)}{z}$

Step 2: use partial fraction expansion to expand $\frac{F(z)}{z}$ into simple elementary fractions.

Step 3: obtain again the original function $F(z) = \frac{F(z)}{z} \times z$

Step 4: apply the inverse Z transform definition and use Z transform table to obtain the discrete time signal $f(kT_s)$.

Illustration 2.3

Given the z- function described by: $F(z) = \frac{z}{4z^2-5z+1}$.

Use the partial fraction expansion method to calculate the sampled signal $f(kT_s)$.

Answer:

We shall follow the indicated procedure according to the shown steps.

Step 1: construction of the function $\frac{F(z)}{z}$

We have :

$$\frac{F(z)}{z} = \frac{z}{4z^2 - 5z + 1} \times \frac{1}{z} = \frac{1}{4z^2 - 5z + 1} = \frac{1}{(z-1)\left(z-\frac{1}{4}\right)}$$

Step 2 : expansion of $\frac{F(z)}{z}$ into simple fractions

$$\frac{F(z)}{z} = \frac{1}{(z-1)\left(z-\frac{1}{4}\right)} = \frac{C_1}{(z-1)} + \frac{C_2}{\left(z-\frac{1}{4}\right)}$$

Where :

$$C_1 = \left[(z-1)\frac{\mathrm{F(z)}}{\mathrm{z}}\right]\Bigg|_{z=1} = \left[(z-1)\frac{1}{(z-1)(z-\frac{1}{4})}\right]\Bigg|_{z=1} = \left[\frac{1}{(z-\frac{1}{4})}\right]\Bigg|_{z=1} = \frac{4}{3}$$

$$C_2 = \left[\left(z-\frac{1}{4}\right)\frac{\mathrm{F(z)}}{\mathrm{z}}\right]\Bigg|_{z=\frac{1}{4}} = \left[\left(z-\frac{1}{4}\right)\frac{1}{(z-1)(z-\frac{1}{4})}\right]\Bigg|_{z=\frac{1}{4}} = \left[\frac{1}{(z-1)}\right]\Bigg|_{z=\frac{1}{4}} = -\frac{4}{3}$$

Therefore:

$$\frac{F(z)}{z} = \frac{1}{(z-1)\left(z-\frac{1}{4}\right)} = \frac{\frac{4}{3}}{(z-1)} - \frac{\frac{4}{3}}{\left(z-\frac{1}{4}\right)}$$

Step 3: obtaining $F(z)$

$$F(z) = \frac{F(z)}{z} \times z = \frac{\frac{4}{3}z}{(z-1)} - \frac{\frac{4}{3}z}{\left(z-\frac{1}{4}\right)}$$

Then:

$$f(k) = f(kT) = \mathcal{Z}^{-1}\{F(z)\} = \mathcal{Z}^{-1}\left\{\frac{\frac{4}{3}z}{(z-1)} - \frac{\frac{4}{3}z}{\left(z-\frac{1}{4}\right)}\right\}$$

$$f(k) = \frac{4}{3}\mathcal{Z}^{-1}\left\{\frac{z}{(z-1)}\right\} - \frac{4}{3}\mathcal{Z}^{-1}\left\{\frac{z}{\left(z-\frac{1}{4}\right)}\right\}$$

Step 4: using Z transform table of standard signals

By referring to Z transform table of standard sampled signals we find:

$$\mathcal{Z}^{-1}\left\{\frac{z}{(z-1)}\right\} = u(kT_s)\,, \qquad \forall\, T_s > 0$$

$$\mathcal{Z}^{-1}\left\{\frac{z}{\left(z-\frac{1}{4}\right)}\right\} = \left(\frac{1}{4}\right)^{kT_s}, \qquad \forall\, T_s > 0$$

It results:

$$f(k) = f(kT) = \frac{4}{3}u(kT_s) - \frac{4}{3}\left(\frac{1}{4}\right)^{kT_s}, \qquad \forall\, T_s > 0$$

7.2.3. Polynomial Division Method

Polynomial division, also called long division, is based on one sided definition of Z transform given by (2.2) obtained after performing polynomial division of the Numerator $N(z)$ over Denominator $D(z)$ of the function $F(z)$. The result of this division gives rise the time sequence defining the sampled signal; that is if:

$$F(z) = \mathcal{Z}\{f(kT_s)\} = \mathcal{Z}\{f^*(t)\} = \sum_{k=0}^{\infty} f(kT_s)z^{-k}$$

$$\Rightarrow f(kT_s) = \mathcal{Z}^{-1}\{F(z)\} = \{f(0), f(1T_s), f(2T_s), f(3T_s), \dots \dots . .\}$$

The number of samples in the obtained time sequence is determined in such a way sufficient data points are reached.

Example:

Consider the function defined as:

$$F(z) = \frac{2z + 3}{z^2 - 0.4z + 0.2}$$

We want to calculate the inverse Z transform using the polynomial division method.

Solution:

$$2z + 3 \;\Big|\; \underbrace{z^2 - 0.4z + 0.2}_{2z^{-1} + 3.8z^{-2} + 1.12z^{-3} + \cdots \dots}$$

The result of division yields:

$$F(z) = \mathbf{2}z^{-1} + \mathbf{3.8}z^{-2} + \mathbf{1.12}z^{-3} + \cdots \dots$$

From which the inverse Z transform is:

$$f(kT_s) = \mathcal{Z}^{-1}\{F(z)\} = \{f(0), f(1T_s), f(2T_s), f(3T_s), \dots \dots . .\}$$

$$f(kT_s) = \{0, 2, 3.8, 1.12, \dots \dots . .\}$$

Chapter 3: Modeling and Representation of Sampled Data Linear Control Systems

Representation and modeling is a mandatory task throughout the process of designing any feedback control system and analyzing its performance. It allows defining and establishing the relationship between mainly the input and output signal as well as other signals affecting the operating performance of the control system. In this chapter we will be interesting of explaining the different methods and approaches used to mathematically represent and model the behavior of the sampled data (discrete time) control system.

1. Basic Notions and Definitions

Before we tackle the subject of this chapter stated earlier, it is important to present some preliminary definitions which need to be known.

1.1. Automatic System

Automatic system is the field and domain that concerns the design methods and approaches which ensure the control of a physical system (process) behavior without the intervention of human being. Consequently, automatics deals with aspects like: modeling, identification and control system dynamics.

1.2. System Modeling

Modeling process can be defined as being the establishment and representation of the relationship between the input and the output variables as well as other affecting signals for a given physical system. The objective is to emphasize how a target output variable(s) can be controlled regarding the influence of input design or disturbance variables; hence mimicking the real behavior of the physical system.

1.3. System Identification

Identification of systems is mutually linked to its modeling, indeed it is the process of determining the designation of different variables describing the system's model as well as the values of the associated parameters. Modeling and identification are two necessary phases before any work of control that can be done for an automatic system.

1.4. System Control

Controlling a system is defined to be the use and application of a control law or mechanism to affect the variation of its output response with respect to the applied desired behavior and taking into account the influence of surrounding environment. The objective of system control is to ensure the desired performance of stability, robustness and other properties.

Regarding system control objectives, the general case is to make the system's output response follow and track the variable profile imposed at its input reference whatever the surrounding influencing conditions (such as: noise, disturbance,…etc.). Particular case is when the aim is to keep and maintain the output response constant and fixe at that value preset at the input reference in all operating conditions; this control system is known, in the field of automatics, as regulatory control (automatic) system.

Either general control system or regulatory control system, the following two types can be distinguished and implemented; we speak about open loop and closed loop control systems.

1.4.1. Open loop control system

An open loop control system is defined as the system when the control law does not depend on the value of the output but only on the value imposed at the input reference. In other word, an open loop control system there is no feedback from the output to the input.

For sampled data control system, a typical block diagram of open loop system is shown in **Fig.3.1**.

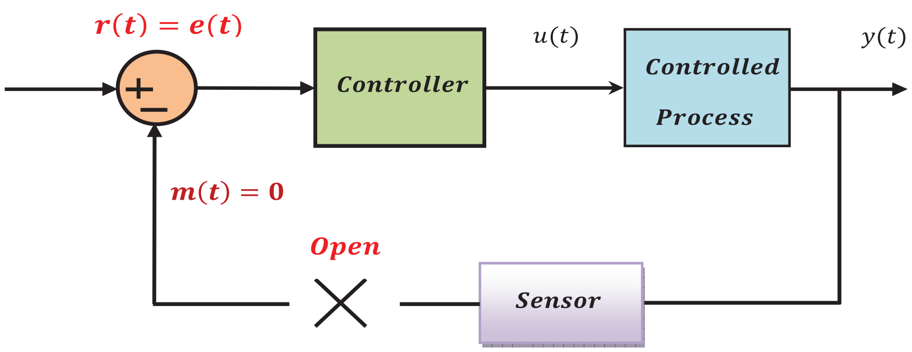


Fig.3.1 typical block diagram of open loop control system

1.4.2. Closed (feedback) loop control system

Unlike open loop control system, closed loop (also called feedback) control system is characterized by the fact that the control law depends on the actual value of the output response. This is implemented by feeding back the system's output to be compared with the imposed reference signal, which based on the comparison value the control and actuating signal is issued.

For sampled data control system, a typical block diagram of closed loop system is shown in **Fig.3.2**.

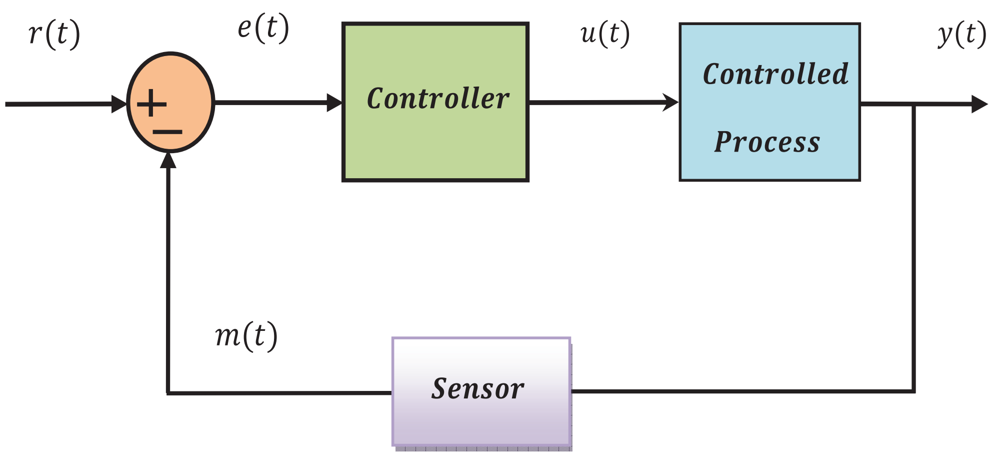


Fig.3.2 typical block diagram of closed loop control system

1.5. Continuous control system

A control system is said to be continuous (or analog) if both its input and output signals are continuous time or analog.

1.6. Sampled date control system

A control system is said to be sampled data, also called discrete or digital, if when it receives sampled (discrete or digital) signal it responds with a sampled or discrete time signal.

It is important to state that, in automatic control system domain, the terms sampled data, discrete and digital are equivalently and interchangeably used and employed.

1.7. Linear control system

A linear control system is defined as a system where its input-output representation or modeling can be described by a linear differential equation (for continuous time case) and by a linear difference equation (for discrete time case). This means that, for linear system, the system parameters do not vary as a function of signal level [3].

1.8. Time invariant control system

Generally we define a time invariant system as being the system for which when a time shifted input signal is applied, the corresponding produced and generated output response signal is characterized by the same amount of time shift.

1.9. Linear time invariant (LTI) control system

A control system is said to be linear time invariant (LTI) when it is both linear and time invariant. From the mathematical point of view, an LTI system can be represented and modeled using linear differential equation of constant coefficients (for the case of continuous LTI system) or using difference equations (for the case of discrete LTI system).

1.10. SISO and MIMO control systems

Single input single output (SISO) control systems are characterized by only one signal as input and one controlled signal as output, however, multi-input multi-output (MIMO) control systems are those systems which are characterized by several variable signals as inputs and several controlled variable signals as outputs. It is important to point out that the same control methodology can be applied for both SISO and MIMO systems, the difference resides in the modeling and identification phases as well as the fact that MIMO systems encompass many output variables need to be controlled in response to the applied inputs.

2. General block diagram of sampled data control system

A sampled data (discrete) control system is implemented around a digital controller instead of the old existing analog controller; that is we may need only to change the analog controller and keep all other parts of the control system such as the controlled process and the used sensors. However, for the digital controller, as a digital system, to work and operate properly, additional signal conversion devices need to be inserted at its input and output for signal conditioning and adaptation purposes.

Among these devices to be used with the digital controller are respectively the Analog-to-Digital Converter (ADC) at the controller's input and the Digital-to-Analog Converter (DAC) placed at the controller's output.

With these two conversion devices inserted additionally with the digital controller, the general structure of sampled data (discrete) control system becomes as it is shown in **Fig.3.3**.

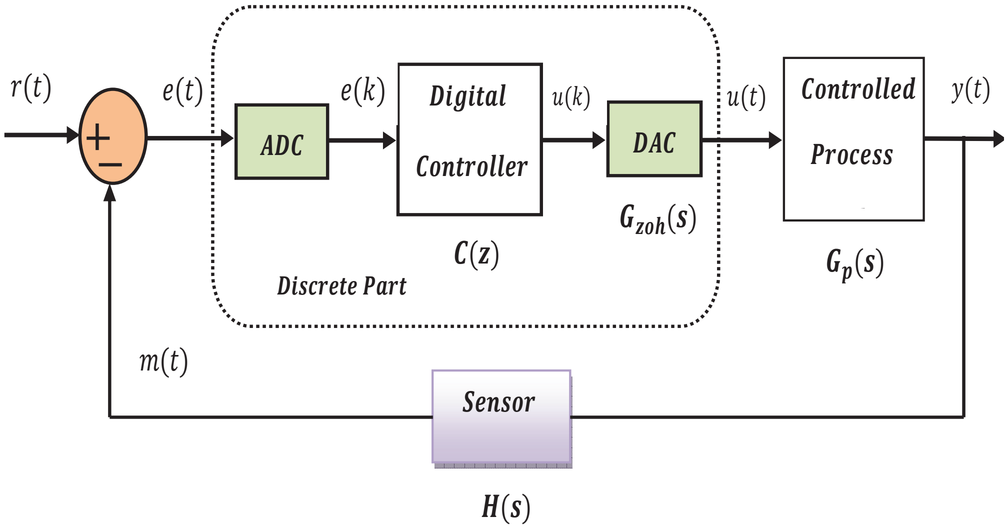


Fig.3.3 General block diagram of sampled data (discrete) control system

2.1. Component description of typical sampled data control system

From the general and typical block diagram representing the sampled data (discrete) control system, the following components and parts are generally found.

- **Analog Process or Plant** ($G_p(s)$): this represents the process to be controlled; that is it involves the output variable(s) being controlled or regulated. The controlled process is often preceded an actuating element or actuator.
- **Digital Controller** ($C(z)$)**:** this is the main component in a feedback control system which is compulsorily being discrete or digital. The digital controller is usually represented by a discrete transfer function denoted as: $C(z)$.

- **Sensor** ($H(s)$): for feedback control system we always need to measure the actual value of the output variable subjected to control. This is achieved using the sensor, which is generally represented by the s-transfer function $H(s)$.
- **Analog-to-Digital Converter (ADC):** in sampled data control system, the DAC is necessary to convert the analog signal into digital one and hence make it pertinent and adequate as input to the digital controller of the system.
- **Digital-to-Analog Converter (DAC):** the output of the digital controller is discrete (digital) which is applied as input to control the output variable of the analog process. As an analog system, the controlled process requires, for its proper operation and functioning, to be actuated by an analog input. Due to that fact, a DAC is inserted after the digital controller to convert its output digital signal into analog signal suitable as input to the process.

Besides these components which are inherent when implementing any sampled data (or digital) control system, the following signal nomenclature is also necessarily to be known. We mean:

- **Setpoint signal ($r(t)$):** it represents the reference real valued quantity to be reached or followed by the controlled output variable of the control system. For general sampled data control system, this signal is assumed continuous time (analog).
- **Controlled output signal ($y(t)$):** this is the output variable of the control system; more precisely, it is the output of the process targeted by the control or regulation. In major cases of implemented sampled data control system, the controlled process is of type analog; hence is its controlled output variable.
- **Measured output signal** ($m(t)$): this signal gives us a quantitative measure of the actual value of the controlled output variable $y(t)$ which is issued by used sensor. The measured output signal may be discrete time or continuous time depending on the implementing method of the control system;

however, in our particular case of system's block diagram shown in **Fig.3.3**, it is assumed analog-type signal.

- **Error signal** ($\boldsymbol{e(t)}$): this signal represents instantly the difference between the setpoint and the measured controlled output signal values. Mathematically, we define the continuous time type error signal as:

$$e(t) = r(t) - m(t) \tag{3.1}$$

For a particular case of unity feedback control system, we have $m(t) = y(t)$, expression (3.1) becomes:

$$e(t) = r(t) - y(t) \tag{3.2}$$

Obviously, the type of the error signal whether it is discrete time or continuous time depends on both the sensor's technology used (analog or digital) and the implemented sampled data control system.

- **Control (actuating) signal ($\boldsymbol{u(k)}$):** it is the most important quantity in any feedback control system (continuous or discrete), which handles the action taken by the controller and being applied on the controlled process to alter the output response. Of course this signal is originally discrete (digital) because it is outputted by the used digital controller; it is being instantly converted into continuous time quantity $u(t)$ using the inserted DAC to best suit the proper operation and functioning of the controlled process.

Regarding its importance in the control system's general block diagram and in order to include it in further analysis of the control system performance and characteristics, a modeled and representative component of the ADC converter is instead used. We will adopt the model represented and shown in **Fig.3.4**.

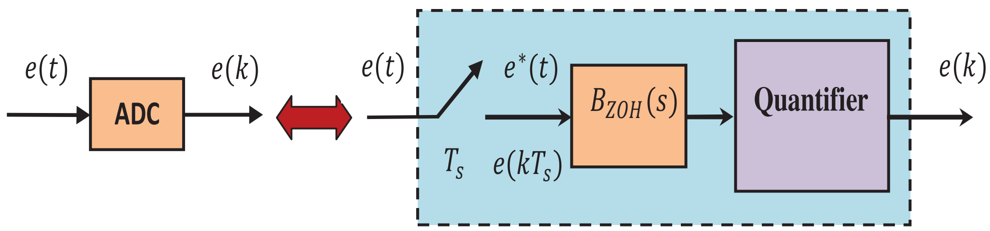


Fig.3.4 Typical model representation of ADC (sampler)

As the case of DAC, the ADC converter should also be given a model such that it can be included in the analysis and design of the sampled data control system. a typical model and representation of this component is shown in **Fig.3.5**.

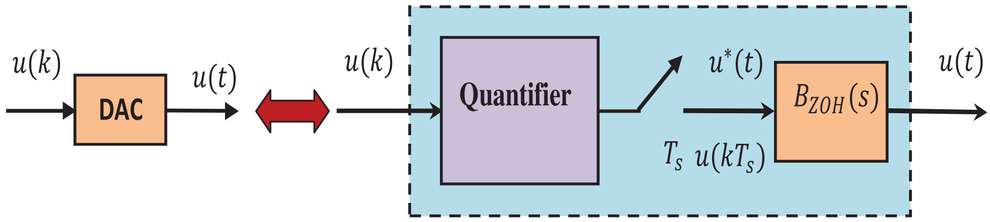


Fig.3.5 Typical model representation of DAC

Regarding the design and analysis of a typical sampled data (digital) control system, which is our fundamental focus and interest in this material, the quantifier in both ADC and DAC elements can be disregarded without affecting the targeted objectives. As such, the final models that can be given for these two components are represented in the following **Fig.3.6** and **Fig.3.7** respectively.

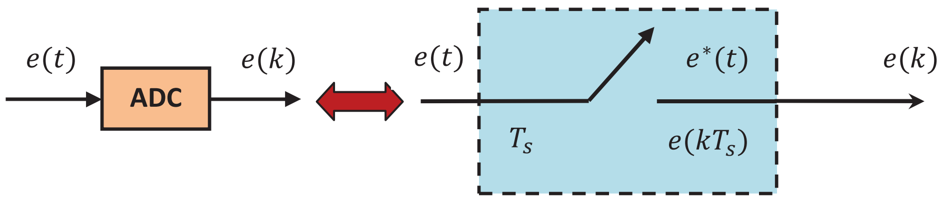


Fig.3.6 Final model representation of ADC (sampler)

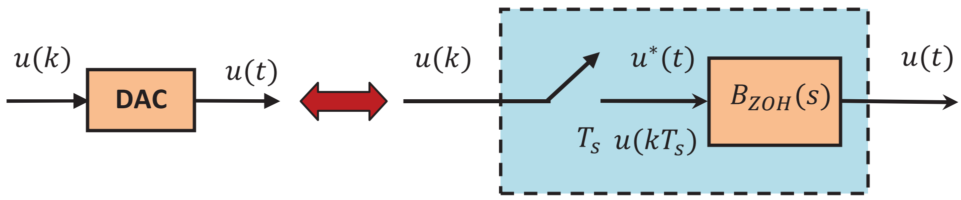


Fig.3.7 Final model representation of DAC

Consequently, the final general block diagram representing a sampled data control system shown earlier in Fig.3.3 is typically represented by the block diagram of **Fig.3.8**.

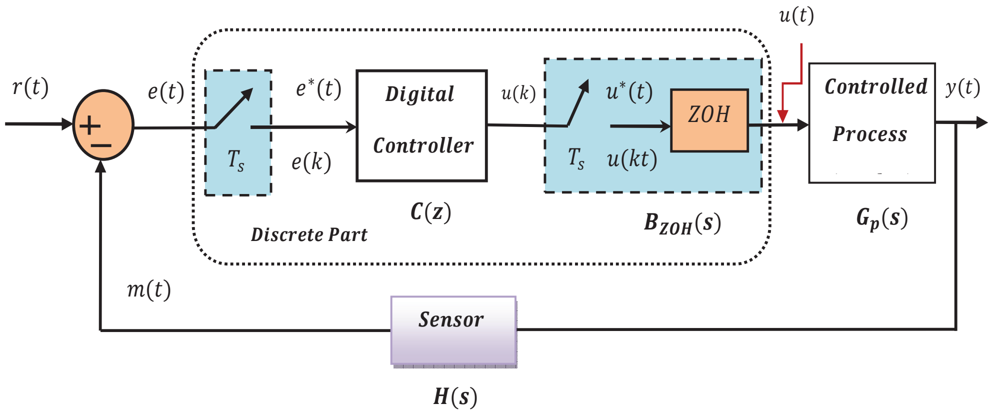


Fig.3.8 Typical general block diagram representation of sampled data control system

3. Sampled data linear time invariant (LTI) control system modeling and representation

3.1. Modeling of Linear time invariant system

For any sampled data linear time invariant (LTI) system which is characterized by a single input and single output (SISO), a general input-output representation can be given by the following diagram.

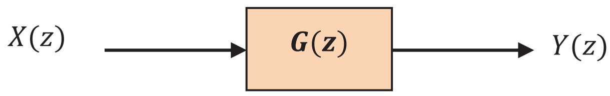


Fig. 3.9 General block diagram representation of LTI SISO and sampled data system

With:

$X(z), Y(z)$: represent, respectively the systems' input and output.

$G(z)$: denotes the transfer function describing the system's dynamic behaviour.

To model and represent the dynamic behavior of a sampled data (discrete time) system, we distinguish three methods:

- Discrete transfer function.

- Difference equations.
- Block diagram representation.

The following sections of this chapter will devoted to explain and describe how a discrete time system can be modeled and represented using these methods, where the relationship between the different representations is mentioned as well.

3.2.1 Modeling using discrete transfer function

As in the case of continuous time system, the discrete time system can also be represented by a discrete transfer function. Mathematically, it is defined as the ratio between the z transform of the input and the output of the system. the discrete transfer function is denoted by $G(z)$ and is expressed as:

$$G(z) = \frac{Y(z)}{X(z)} \tag{3.3}$$

With:

$X(z)$ and $Y(z)$ represent respectively the z transform of the input and the z transform of the output.

Analytically, the discrete transfer function of a sampled data system can be explicitly expressed under three forms:

- Polynomial ratio as a function of the variable 'z'.
- Polynomial ratio as a function of the variable 'z^{-1}'.
- Gain-pole-zero form.

(1) Polynomial ratio as a function of the variable 'z'

A general form and expression of the discrete transfer function using the ratio of two polynomials as a function of the complex variable 'z' is given as follows:

$$G(z) = \frac{N(z)}{D(z)} = \frac{(b_0 z^m + b_1 z^{m-1} + \cdots + b_m z^0)}{(a_0 z^n + a_1 z^{n-1} + \cdots + a_n z^0)} \tag{3.4}$$

With:

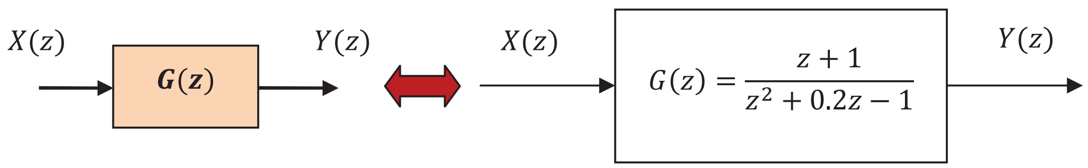
$N(z)$ and $D(z)$ are respectively called the numerator and denominator of the transfer function $G(z)$.

b_j and a_i $(i = 0,1,2,3, \dots, n, j = 0,1,2,3, \dots, m)$ are known coefficients corresponding respectively to the numerator $N(z)$ and denominator $D(z)$.

m and n are respectively the degrees of the numerator and denominator of the transfer function.

Example:

As an example of this representation form of a discrete transfer function, we consider the following:



In this example, we have: $N(z) = z + 1$, $D(z) = z^2 + 0.2z - 1$, where we notice that both polynomials are function of the complex variable 'z'.

We can use Matlab environment to implement this form of the discrete transfer function using the Matlab user defined function '***tf***'. this is done as follows:

```
num_G =[1 1];
den_G =[1 0.2 -1];
Ts = 1;
G = tf(num_G, den_G,Ts,'variable','z')
```

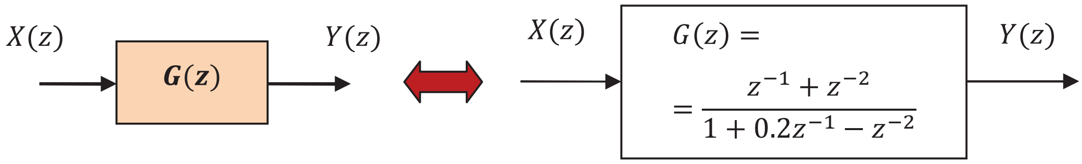
(2) Polynomial ratio as a function of the variable 'z^{-1}'

In some cases and for the sake of making easy the analytical manipulation and analysis of the discrete (sampled data) control system, we often require that the complex variable be 'z^{-1}' instead of 'z'. Therefore, by expressing the numerator $N(z)$ and the denominator $D(z)$ as a function of the complex variable 'z^{-1}', the discrete transfer function can also be given the following form;

$$G(z) = \frac{N(z)}{D(z)} = \frac{(b_0 + b_1 z^{-1} + \cdots + b_m z^{-m})}{(a_0 + a_1 z^{-1} + \cdots + a_n z^{-n})} \tag{3.5}$$

Example;

As an example to the discrete transfer function expressed under this form we give:



For which, $N(z) = z^{-1} + z^{-2}$, $D(z) = 1 + 0.2z^{-1} - z^{-2}$, and both of the polynomials are function of 'z^{-1}'.

In this case also, it is possible to implement the discrete transfer function in Matlab environment using the following set of instruction with Matlab function '**tf**':

```
num_G=[0 1 1];
den_G=[1 0.2 -1];
Ts= 1;
G = tf(num_G, den_G,Ts,'variable','z^-1')
```

(3) Gain-pole-zero form of a discrete transfer function

Sometimes, it is helpful to factorize the numerator $N(z)$ and denominator $D(z)$ with respect to their roots. The result will be a new form of the discrete transfer function known as Gain-pole-zero form that is expressed as follows:

$$G(z) = \frac{N(z)}{D(z)} = \frac{b_0}{a_0}\frac{(z - z_1)(z - z_2)\dots(z - z_m)}{(z - p_1)(z - p_2)\dots(z - p_n)} = \frac{b_0}{a_0}\frac{\prod_{j=1}^{m}(z - z_j)}{\prod_{i=1}^{n}(z - p_i)} \qquad (3.6)$$

$$G(z) = \frac{N(z)}{D(z)} = \frac{b_0}{a_0}\frac{(1 - z_1 z^{-1})\dots(1 - z_m z^{-1})}{(1 - p_1 z^{-1})\dots(1 - p_n z^{-1})} = \frac{b_0}{a_0}\frac{\prod_{j=1}^{m}(1 - z_j z^{-1})}{\prod_{i=1}^{n}(1 - p_i z^{-1})} \qquad (3.7)$$

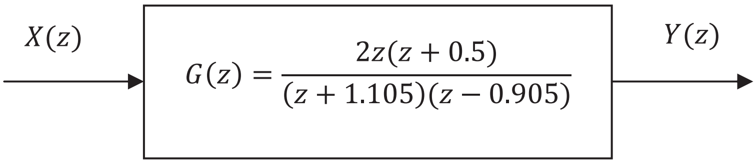
With:

$\frac{b_0}{a_0}$: represents the gain of the discrete system.

p_i and z_j (with $i = 1,2,3, \dots, n$, $j = 1,2,3, \dots, m$) represent respectively the roots of the polynomials $D(z)$ and $N(z)$, which are respectively known to be the poles and zeros of the discrete transfer function representing the discrete system.

Example:

As an example of the discrete transfer function written under the form gain-pole-zero, we give:



Obviously, in order to express a discrete transfer function under the form gain-pole-zero, from the polynomial ratio, we follow the steps in below:

- Determine and calculate the gain of the transfer function using its definition as:

$$K = \frac{b_0}{a_0}$$

- Determine the poles p_i and zeros z_j of the transfer function which are respectively the roots of the polynomials $D(z)$ and $N(z)$.
- Finally, the gain-pole-zero of the transfer function is either of the forms (3.6) or (3.7).

We can also generate the gain-pole-zero form of the discrete transfer function using MATLAB environment via the set of the following instructions.

```
Zeros_G= [0 -0.5]' ; Poles_G= [-1.105 0.905]';
K= 2; Ts= 1;
G = zpk(Zeros_G ,Poles_G ,K ,Ts)
```

In this Matlab script, the gain of the discrete transfer function is determined to be equals $K = \frac{b_0}{a_0} = 2$.

On the other hand, the poles and zeros of the transfer function are respectively given by the two row vectors as follows:

$$p_i = [-1.105 \quad 0.905]^{\mathrm{t}} = \begin{bmatrix} -0.105 \\ \\ 0.905 \end{bmatrix} \quad ; \qquad z_j = [0 \quad -0.5]^{\mathrm{t}} = \begin{bmatrix} 0 \\ \\ -0.5 \end{bmatrix}$$

By using the MATLAB user-defined function '**zpk**', we generate the gain-pole-zero form of the transfer function.

3.1.1.1. System Characteristics from Discrete Transfer Function

Regardless the form under which the discrete transfer function can be expressed, the following properties and characteristics of the digital control system are being extracted and deduced.

(i) The roots of the numerator $N(z)$ are called the **zeros** of the transfer function (or of the system). They are denoted as: $z_j, j = 1,2,3, \dots, m$.

(ii) The roots of the denominator $D(z)$ are called the **poles** of the transfer function (or of the system). They are denoted as: $p_i, i = 1,2,3, \dots, n$.

(iii) The denominator $D(z)$ is called the **characteristic polynomial** of the discrete transfer function.

(iv) Correspondingly, the equation defined as $D(z) = 0$ is known as the **characteristic equation** of the discrete system.

(v) The **degree** of the polynomial $D(z)$ is equal to **the order** of transfer function (or the discrete system).

(vi) Lastly, for the discrete system (transfer function) to be **realizable**, it should satisfy the condition of $n \geq m$

3.2.2 Modeling using Difference Equation

The difference (also called recurrent) equation for linear time invariant (LTI) sampled data (discrete time) system is analogous to differential equation used to describe and model the LTI analog (continuous time) system.

In general form, the dynamics of an LTI discrete system can be described by the difference equation given as:

$$\sum_{j=0}^{m} b_j x(k-j) = \sum_{i=0}^{n} a_i y(k-i) \tag{3.8}$$

With :

$x(k-j)$ and $y(k-i)$; $i = 0,1,,2, \dots, n$; $j = 0,1,2,3, \dots, m$ are respectively the $(m+1)$ input samples and $(n+1)$ output samples of the discrete system.

b_j, a_i: are assumed to be known real coefficients.

Therefore, using the difference equation, we can dynamically model the behaviour of the discrete system via the time occurrence of the input and output samples.

As a particular case, when $a_i = 1,\ i = 0$, the equation (3.8) is also written as:

$$y(k) = \sum_{j=0}^{m} b_j x(k-j) - \sum_{i=1}^{n} a_i y(k-i) \qquad (3.9)$$

The equation (3.9) is particularly simpler when implementing the discrete time (digital) system.

3.2.2.1 Relationship between transfer function and difference equation representation

The two representations of transfer function and difference equation used to model a discrete control system are in fact interrelated; that is we can go from one representation to another, also the reverse operation of obtaining one representation from the other is possible.

In the following, we will describe, in general manner, how this relationship is established.

We consider that we have a discrete system represented by its z transfer function given generally as:

$$G(z) = \frac{Y(z)}{U(z)} = \frac{N(z)}{D(z)} = \frac{(b_0 + b_1 z^{-1} + \cdots + b_m z^{-m})}{(a_0 + a_1 z^{-1} + \cdots + a_n z^{-n})} \qquad (3.10)$$

With:

$Y(z)$ and $U(z)$ are respectively the Z transforms of the input and output of the discrete system.

For the sake of simplifying things, the two polynomials of the transfer function $N(z)$ and $D(z)$ are expressed as a function of z^{-1}.

By performing cross multiplication of (3.10), we obtain:

$$\frac{Y(z)}{U(z)} = \frac{(b_0 + b_1 z^{-1} + \cdots + b_m z^{-m})}{(a_0 + a_1 z^{-1} + \cdots + a_n z^{-n})}$$

$$\Longrightarrow\ (a_0 + a_1 z^{-1} + \cdots + a_n z^{-n})\, Y(z) = (b_0 + b_1 z^{-1} + \cdots + b_m z^{-m})\, U(z)$$

$$\Longrightarrow\ \left(a_0 Y(z) + a_1 z^{-1} Y(z) + \cdots + a_n z^{-n} Y(z)\right) =$$

$$= \left(b_0 U(z) + b_1 z^{-1} U(z) + \cdots + b_m z^{-m} U(z)\right) \quad (3.11)$$

We apply next the inverse Z transform on both sides of expression (3.11), we get:

$$Z^{-1}\{a_0 Y(z) + a_1 z^{-1} Y(z) + \cdots + a_n z^{-n} Y(z)\} =$$

$$= Z^{-1}\{b_0 U(z) + b_1 z^{-1} U(z) + \cdots + b_m z^{-m} U(z)\}$$

Using Z transform and inverse Z transform properties, which are identical, particularly the two properties of linearity and Z transform of time delay, it results:

$$a_0 Z^{-1}\{Y(z)\} + a_1 Z^{-1}\{z^{-1} Y(z)\} + \cdots + a_n Z^{-1}\{z^{-n} Y(z)\} =$$

$$= b_0 Z^{-1}\{U(z)\} + b_1 Z^{-1}\{z^{-1} U(z)\} + \cdots + b_m Z^{-1}\{z^{-m} U(z)\}$$

$$\Longrightarrow a_0 y(k) + a_1 y(k-1) + \cdots + a_n y(k-n)$$

$$= b_0 u(k) + b_1 u(k-1) + \cdots + b_m u(k-m)$$

Which is the corresponding difference equation representing the discrete (sampled data) system as it is obtained from the transfer function representation. This difference equation can be rewritten in more general form as:

$$\sum_{j=0}^{m} b_j x(k-j) = \sum_{i=0}^{n} a_i y(k-i) \quad (3.12)$$

Obviously, equation (3.12) is equivalent to equation (3.8).

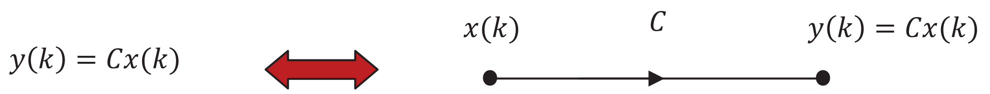
3.2.3 Modeling discrete system using block diagram

The representation and modeling of LTI discrete system using block diagram, three basic elements are indeed required to be introduced. These are:

- Scalar multiplication.
- Algebraic sum
- Discrete time delay.

3.2.3.1. Scalar Multiplication

The question is how to represent graphically a discrete signal multiplied by a scalar C. This is done according to the following illustration (**Fig.3.10**):



We can also do this representation as:

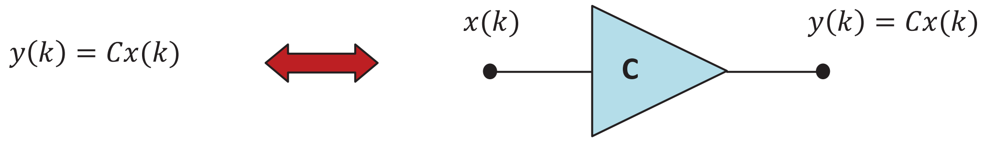


Fig. 3.10 Block diagram representation of scalar multiplication operator

3.2.3.2. Algebraic Sum

The algebraic sum allows us to represent graphically several signals when they are summed up to produce an output signal.

If $x_1(k)$, $x_2(k)$ and $x_3(k)$ are supposed to be three signals scaled respectively by the gains C_1 , C_2 and C_3 which are applied at the input of an LTI discrete system to produce the output signal $y(k)$. The block diagram representation of this sum of signals is shown as in **Fig.3.11**:

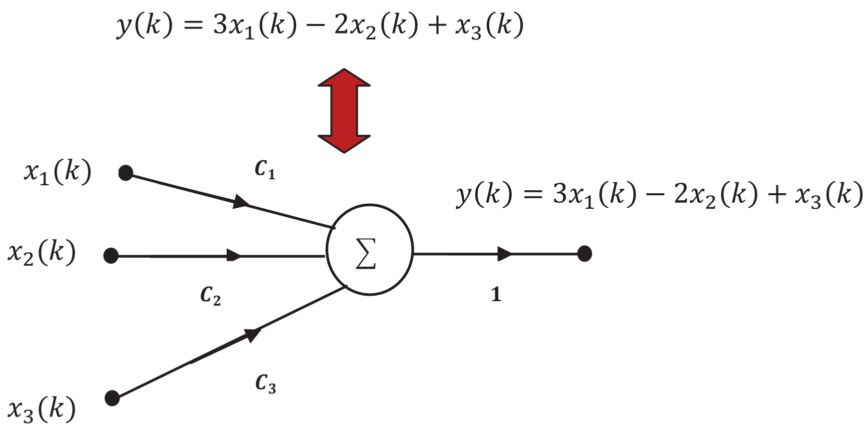


Fig. 3.11 Block diagram representation of algebraic sum

3.2.3.3 Discrete Time Delay

In discrete time systems, time delay is represented by the operator z^{-1}. The block diagram representation of the general case of time delay is given in **Fig.3.12**.

Fig. 3.12 Block diagram representation of discrete time delay

Illustrative example:

Using block diagram representation, a typical LTI discrete (sampled data) system is shown in **Fig.3.13**.

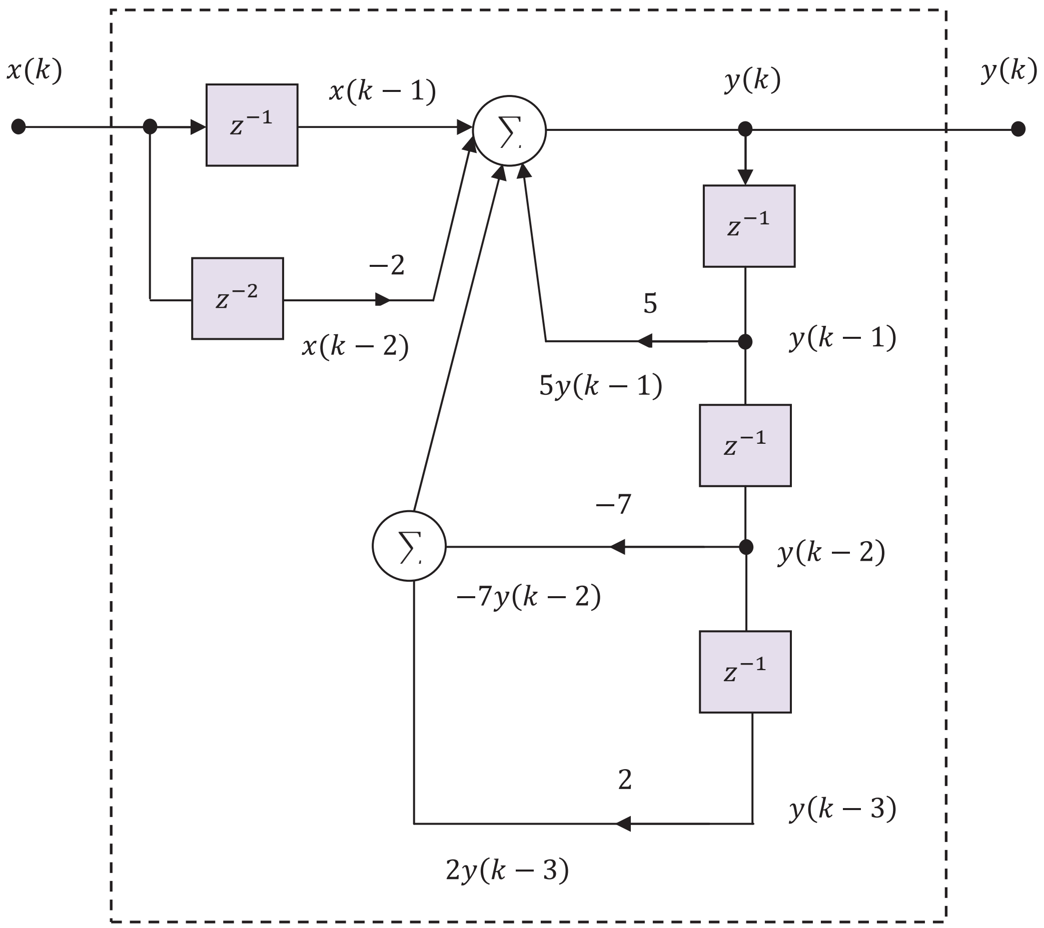


Fig.3. 13 Typical block diagram representation of LTI discrete system

2.3.4. From block diagram to discrete transfer function representation

We can obtain the discrete transfer function representing the behavior of any discrete system given its block diagram representation. The discrete transfer function is generated by passing through the difference equation determination from the block diagram. In the following illustrative example, we show the step-by-step procedure of so doing.

Example:

Consider the discrete (digital) system represented by the block diagram shown in **Fig.3.13; d**erive the corresponding discrete transfer function of the system?

Solution:

From the block diagram representation of **Fig.3.13,** we can easily write the corresponding difference equation describing the behavior of the digital system as:

$$y(k) = x(k-1) - 2x(k-2) + 5y(k-1) - 7y(k-2) + 2y(k-3) \qquad (3.13)$$

By applying Z transform on both sides of eq.(3.13), we get:

$$Z\{y(k)\} = Z\{x(k-1) - 2x(k-2) + 5y(k-1) - 7y(k-2) + 2y(k-3)\}$$

Using Z transform properties particularly that of linearity and time delay, it results:

$$Y(z) = z^{-1}X(z) - 2z^{-2}X(z) + 5z^{-1}Y(z) - 7z^{-2}Y(z) + 2z^{-3}Y(z)$$

We arrange the input samples in one side and the output samples in one side, it becomes:

$$Y(z) - 5z^{-1}Y(z) + 7z^{-2}Y(z) - 2z^{-3}Y(z) = z^{-1}X(z) - 2z^{-2}X(z)$$

$$\Rightarrow (1 - 5z^{-1} + 7z^{-2} - 2z^{-3})Y(z) = (z^{-1} - 2z^{-2})X(z)$$

We apply the definition of discrete transfer function, we write:

$$G(z) = \frac{Y(z)}{X(z)} = \frac{z^{-1} - 2z^{-2}}{1 - 5z^{-1} + 7z^{-2} - 2z^{-3}}$$

We can express $G(z)$ under the form of ratio of polynomials as function of z as:

$$G(z) = \frac{Y(z)}{X(z)} = \frac{z^{-1} - 2z^{-2}}{1 - 5z^{-1} + 7z^{-2} - 2z^{-3}} \times \frac{z^3}{z^3} = \frac{z^2 - 2z}{z^3 - 5z^2 + 7z - 2}$$

Consequently, we have explained the relationship between the different representations of any given discrete or digital system. we have also mention how to move from one representation to another by just applying either Z transform or its inverse with their appropriate properties.

4. Discretizing LTI continuous time systems

We have seen in the first chapter the use of sampler to convert a continuous time (analog) signal into discrete time (digital) signal. The operation was called sampling or discretization and the electronic device performing this conversion is the analog to digital converter (ADC) which is modeled by an ideal switch as long as the analysis and design of digital control system is concerned.

On the other hand, the design and analysis of sampled data control system often requires the discretization of a continuous time system. Usually this continuous time system to be discretized is represented by the Laplace transfer function.

In this section we explore the different methods widely used and employed to obtain discrete transfer function from the existing Laplace transfer function, hence discrete system representation is obtained from that of continuous system. In this vein, the following discretizing methods can be distinguished.

(1) Discretization using **Zero Order Hold** (ZOH) method.

(2) Discretization using **One Order Hold** (1OH) method.

(3) Discretization using **Tustin approximation** method.

(4) Discretization using **Euler approximation** method.

4.1. Discretization using Zero Order Hold (ZOH) method

The discretization of a continuous time system represented by an s-transfer function using the method of zero order hold (ZOH) is done by preceding the continuous transfer function $G(s)$ with the ZOH block (transfer function) $G_{ZOH}(s)$. This is illustrated by the following block diagram of **Fig.3.14**.

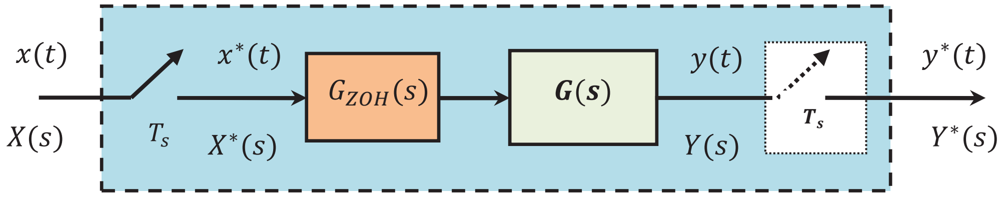


Fig.3. 14 Block diagram representation of continuous system discretization using ZOH method

The discrete LTI system obtained using ZOH discretization method is represented by the following block diagram (**Fig.3.15**).

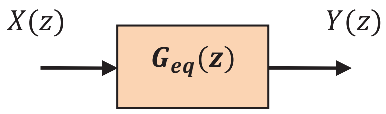


Fig.3. 15 Block diagram of the equivalent discretized LTI system

With:

$G_{eq}(z)$: is the equivalent z transfer function of the resulted discrete system and is calculated as:

$$G_{eq}(z) = \frac{Y(z)}{X(z)} = (1 - z^{-1})\mathcal{Z}\left[\frac{G(s)}{s}\right] \tag{3.14}$$

4.2. Discretization using One Order Hold (1OH) method

The one order hold (1OH) can also be used as a discretization method to obtain a discrete (digital) system from a continuous system. It is similar to ZOH method with the use of 1OH block instead of ZOH block. Hence, using this method, the block diagram illustration is depicted in **Fig.3.16**:

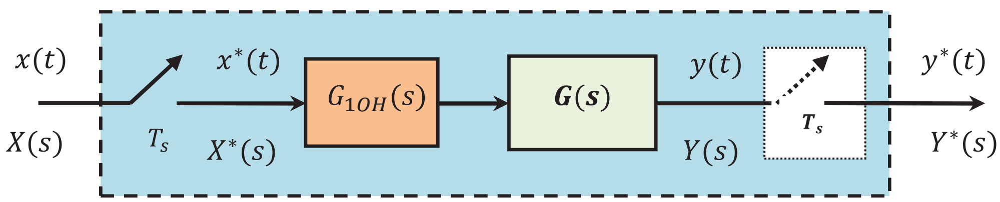


Fig.3. 16 Block diagram representation of continuous system discretization using 1OH method

The discrete LTI system obtained using 1OH discretization method is represented by the following block diagram of **Fig.3.17**.

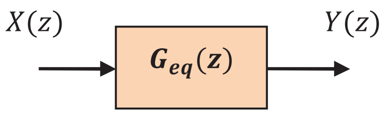


Fig.3. 17 Block diagram of the equivalent discretized LTI system using 1OH method

With:

T_s: is the sampling period.

$G_{eq}(z)$: is the equivalent z transfer function of the resulted discrete system and is calculated as:

$$G_{eq}(z) = \frac{Y(z)}{X(z)} = (1 - z^{-1})^2 \mathcal{Z}\left[\frac{1 + T_s\, s}{T_s\, s^2} G(s)\right] \tag{3.15}$$

4.3. Discretization using Tustin approximation method

The Tustin approximation, also known as Tustin Bilinear approximation, is used to convert a continuous time system into discrete time system. The equivalent z transfer function of resulted discrete system is obtained by performing in the s transfer function the following direct replacement:

$$s = \frac{2}{T_s}\frac{(z-1)}{(z+1)} \tag{3.16}$$

With :

T_s: is the sampling period.

Hence, the equivalent z transfer function representing the discrete system is generated and written:

$$G_{eq}(z) = G(s)\big|_{s=\frac{2\,(z-1)}{T_s(z+1)}} \tag{3.17}$$

We can illustrate Tustin approximation method of discretizing a continuous time system by the following block diagram of **Fig.3.18**.

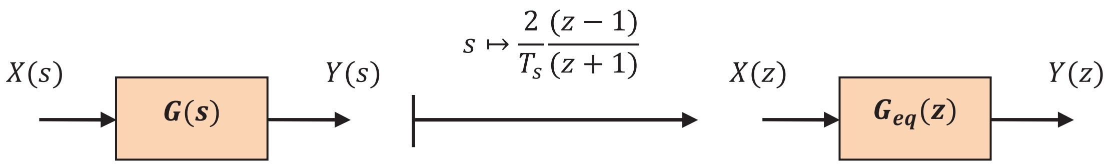


Fig.3. 18 Block diagram representation of continuous system discretization using **Tustin** method

4.4. Discretization using Euler approximation method

The Euler's method of discretizing a continuous time system relies on Euler's principle of approximating the first derivative of a continuous function between two consecutive sampling instants. In fact, Euler's method contains two approximate versions known respectively as forward and backward approximations.

4.4.1. Discretization using Euler Backward approximation method

Euler's backward approximation used to discretize a continuous system as it is represented by an s-transfer function consists of direct replacing the complex variable 's' by: $\frac{(z-1)}{T_s z}$. The illustration of this approximation is depicted in **Fig.3.19**.

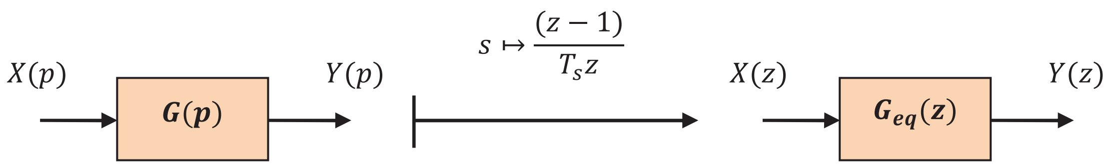


Fig.3. 19 Block diagram representation of continuous system discretization using **Backward Euler's** method

Using this method, the resulting equivalent z (discrete) transfer function can be obtained as:

$$G_{eq}(z) = G(s)\big|_{s=\frac{(z-1)}{T_s z}} \qquad (3.18)$$

Of course, in all the mentioned approximations, T_s is the sampling period.

4.4.2. Discretization using Euler Forward approximation method

The Forward version of Euler's approximation method for discretization is based on replacing the complex variable 's' by: $\frac{(z-1)}{z}$. Therefore, using this approximation, the equivalent discrete transfer function obtained from the s transfer function is generated as:

$$G_{eq}(z) = G(s)|_{s=\frac{(z-1)}{z}} \tag{3.19}$$

The block diagram representation of the discretized system using Euler's Forward approximation is illustrated in the following figure (**Fig.3.20**).

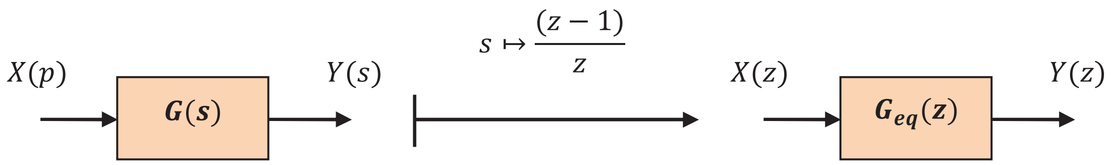


Fig.3. 20 Block diagram representation of continuous system discretization using **Forward Euler's** method

Example:

At the end of this section, we consider this example of discretizing a continuous time system modeled by the Laplace transfer function.

Assume a continuous time system given by the s transfer function as:

$$G(s) = \frac{2}{(3+5s)}.$$

Q: / using the aforementioned discretization methods, obtain the equivalent and corresponding z transfer function $G(z)$?

Answer:

1) Using Zero Order Hold (ZOH)

Using (3.13), the equivalent z transfer function is:

$$G(z) = \frac{Y(z)}{X(z)} = (1-z^{-1})\mathcal{Z}\left[\frac{G(s)}{s}\right] = (1-z^{-1})\mathcal{Z}\left[\frac{2/5}{s\left(s+\frac{3}{5}\right)}\right]$$

Applying partial fraction expansion on $\frac{2/5}{s\left(s+\frac{3}{5}\right)}$, we can write :

$$\frac{2/5}{s\left(s+\frac{3}{5}\right)}=\frac{A_1}{s}+\frac{A_2}{s+\frac{3}{5}}$$

With:

$$A_1=s.\frac{2/5}{s\left(s+\frac{3}{5}\right)}\Bigg|_{s=0}=\frac{2/5}{\left(s+\frac{3}{5}\right)}\Bigg|_{s=0}=\frac{2}{3}$$

$$A_2=\left(s+\frac{3}{5}\right).\frac{2/5}{s\left(s+\frac{3}{5}\right)}\Bigg|_{s=0}=\frac{2/5}{s}\Big|_{s=-\frac{3}{5}}=-\frac{2}{3}$$

That is :

$$\frac{2/5}{s\left(s+\frac{3}{5}\right)}=\frac{\frac{2}{3}}{s}-\frac{\frac{2}{3}}{s+\frac{3}{5}}$$

It results:

$$\mathcal{Z}\left[\frac{2/5}{s\left(s+\frac{3}{5}\right)}\right]=\mathcal{Z}\left[\frac{\frac{2}{3}}{s}-\frac{\frac{2}{3}}{s+\frac{3}{5}}\right]=\frac{2}{3}\mathcal{Z}\left[\frac{1}{s}\right]-\frac{2}{3}\mathcal{Z}\left[\frac{1}{s+\frac{3}{5}}\right]$$

By using the correspondence table between Laplace transform and Z transform, we get:

$$\mathcal{Z}\left[\frac{2/5}{s\left(s+\frac{3}{5}\right)}\right]=\frac{2}{3}\frac{z}{z-1}-\frac{2}{3}\frac{z}{z-e^{\left(-\frac{3}{5}T_s\right)}}$$

Taking: $T_s=1\ sec$, then :

$$\mathcal{Z}\left[\frac{2/5}{s\left(s+\frac{3}{5}\right)}\right]=\frac{2}{3}\frac{z}{z-1}-\frac{2}{3}\frac{z}{z-e^{\left(-\frac{3}{5}\right)}}=\frac{2}{3}\frac{z}{z-1}-\frac{2}{3}\frac{z}{z-0.5488}$$

Finally:

$$G(z) = (1 - z^{-1})\left[\frac{2}{3}\frac{z}{z-1} - \frac{2}{3}\frac{z}{z-0.5488}\right] = \frac{z-1}{z}\left[\frac{2}{3}\frac{z}{z-1} - \frac{2}{3}\frac{z}{z-0.5488}\right]$$

$$G(z) = \frac{2}{3} - \frac{2}{3}\left(\frac{z-1}{z-0.5488}\right) = \frac{2}{3}\left(\frac{z-0.5488-z+1}{z-0.5488}\right) = \frac{2}{3}\left(\frac{0.4512}{z-0.5488}\right)$$

2) Using One Order Hold (OOH) Discretization Method

Using this method, the equivalent z transfer function of the given s transfer function is obtained as follows:

$$G(z) = \frac{Y(z)}{X(z)} = (1 - z^{-1})^2 \mathcal{Z}\left[\frac{1 + T_s\, s}{T_s\, s^2} G(s)\right]$$

Taking: $T_s = 1\, sec$, then :

$$G(z) = \frac{Y(z)}{X(z)} = (1 - z^{-1})^2 \mathcal{Z}\left[\frac{1+\ s}{s^2} G(s)\right] = \frac{2}{5}(1 - z^{-1})^2 \mathcal{Z}\left[\frac{(1+\ s)}{s^2\left(s+\frac{3}{5}\right)}\right]$$

Applying partial fraction expansion on the s transfer function: $\frac{(1+\,s)}{s^2\left(s+\frac{3}{5}\right)}$, we have:

$$\frac{(1+\ s)}{s^2\left(s+\frac{3}{5}\right)} = \frac{A_{11}}{s^2} + \frac{A_{12}}{s} + \frac{A_3}{s+\frac{3}{5}}$$

Where the coefficients are determined as:

$$A_{11} = \frac{1}{0!}\frac{d^{(0)}}{ds^{(0)}}\left[s^2\frac{(1+\ s)}{s^2\left(s+\frac{3}{5}\right)}\right]\Bigg|_{s=0} = \left[\frac{(1+\ s)}{\left(s+\frac{3}{5}\right)}\right]\Bigg|_{s=0} = \frac{5}{3}$$

$$A_{12} = \frac{1}{1!}\frac{d^{(1)}}{ds^{(1)}}\left[s^2\frac{(1+\ s)}{s^2\left(s+\frac{3}{5}\right)}\right]\Bigg|_{s=0} = \frac{d^{(1)}}{ds^{(1)}}\left[\frac{(1+\ s)}{\left(s+\frac{3}{5}\right)}\right]\Bigg|_{s=0}$$

$$= \left[\frac{(1+\ s) - \left(s+\frac{3}{5}\right)}{\left(s+\frac{3}{5}\right)^2}\right]\Bigg|_{s=0} = \frac{2}{3}$$

$$A_3 = \left[\left(s+\frac{3}{5}\right)\frac{(1+\ s)}{s^2\left(s+\frac{3}{5}\right)}\right]\Bigg|_{s=-\frac{3}{5}} = \left[\frac{(1+\ s)}{s^2}\right]\Bigg|_{s=-\frac{3}{5}} = \frac{10}{9}$$

Hence:

$$\frac{(1+\ s)}{s^2\left(s+\frac{3}{5}\right)} = \frac{\frac{5}{3}}{s^2} + \frac{\frac{2}{3}}{s} + \frac{\frac{10}{9}}{s+\frac{3}{5}}$$

It results after using Z transform properties :

$$\mathcal{Z}\left[\frac{(1+\ s)}{s^2\left(s+\frac{3}{5}\right)}\right] = \mathcal{Z}\left[\frac{\frac{5}{3}}{s^2} + \frac{\frac{2}{3}}{s} + \frac{\frac{10}{9}}{s+\frac{3}{5}}\right] = \frac{5}{3}\mathcal{Z}\left[\frac{1}{s^2}\right] + \frac{2}{3}\mathcal{Z}\left[\frac{1}{s}\right] + \frac{10}{9}\mathcal{Z}\left[\frac{1}{s+\frac{3}{5}}\right]$$

Taking $T_s = 1\ sec$ and using the table of correspondence between z and s transforms, we get:

$$\mathcal{Z}\left[\frac{(1+\ s)}{s^2\left(s+\frac{3}{5}\right)}\right] = \frac{5}{3}\frac{z}{(z-1)^2} + \frac{2}{3}\frac{z}{(z-1)} + \frac{10}{9}\frac{z}{(z-0.5488)}$$

Finally :

$$G(z) = \frac{2}{5}(1-z^{-1})^2\mathcal{Z}\left[\frac{(1+\ s)}{s^2\left(s+\frac{3}{5}\right)}\right]$$

$$= \frac{2}{5}(1-z^{-1})^2\left[\frac{5}{3}\frac{z}{(z-1)^2} + \frac{2}{3}\frac{z}{(z-1)} + \frac{10}{9}\frac{z}{(z-0.5488)}\right]$$

$$G(z) = \frac{2}{5}\frac{(z-1)^2}{z^2}\left[\frac{5}{3}\frac{z}{(z-1)^2} + \frac{2}{3}\frac{z}{(z-1)} + \frac{10}{9}\frac{z}{(z-0.5488)}\right]$$

$$G(z) = \left[\frac{2}{3}\frac{1}{z} + \frac{4}{15}\frac{(z-1)}{z} + \frac{4}{9}\frac{(z-1)^2}{z(z-0.5488)}\right]$$

$$= \frac{2}{45}\left[\frac{3(z-0.5488)(3+2z) + 10(z-1)^2}{z(z-0.5488)}\right]$$

3) Using « Tustin » approximation Method :

By direct use of the formula (3.15), and taking $T_s = 1\,sec$ for simplicity, the equivalent discrete transfer function is:

$$G_{eq}(z) = G(s)|_{s=\frac{2\,(z-1)}{T_s(z+1)}} = \frac{2}{\left(3 + 10\frac{(z-1)}{(z+1)}\right)} = \frac{2}{\frac{3(z+1)+10(z-1)}{(z+1)}}$$

$$G_{eq}(z) = \frac{2(z+1)}{13z-7}$$

4) Using Euler's Backward Method

From the formula (3.16), the discrete transfer function using this method version of Euler is:

$$G_{eq}(z) = G(s)|_{s=\frac{(z-1)}{T_s z}} = \frac{2}{\left(3 + 5\frac{(z-1)}{z}\right)} = \frac{2}{\frac{3z+5z-5}{z}} = \frac{2z}{8z-5}$$

5) Using Euler's Forward Method

From the formula (3.16), the discrete transfer function using this method version of Euler is:

$$G_{eq}(z) = G(s)|_{s=\frac{(z-1)}{z}} = \frac{2}{\left(3 + 5\frac{(z-1)}{z}\right)} = \frac{2}{\frac{3z+5z-5}{z}} = \frac{2z}{8z-5}$$

This reveals that when the sampling period equals 1, the **Backward** and **Forward** versions of Euler are identical.

5. Equivalent transfer function of complex discrete control system

In this section, we consider the calculation and determination of the equivalent and global discrete transfer function of a sampled data control system consisted of an interconnection of elementary sub-systems. Any digital control system is found under two main structures depending on the way the elementary sub-systems are connected. These two structures are:

- **Open loop discrete control system**: when the sub-systems are connected in series (or in cascade).

- **Closed loop discrete control system**: when the sub-systems are connected in both cascade and parallel structure.

In developing the content of this subject, we will discover that the final expression of the equivalent discrete transfer function of the sampled data control system depends on the following two factors:

(1) The number of the samplers used in the control system block diagram.

(2) The position of the samplers regarding the elementary sub-systems.

5.1. Case of Open Loop discrete control system

In order to well understanding how to determine the equivalent discrete transfer function for given open loop discrete control system, we distinguish the following cases of how both the subsystems and the samplers are interconnected.

5.1.1 Case of the continuous time system is between two samplers

This situation can be depicted as in **Fig. 3. 21**.

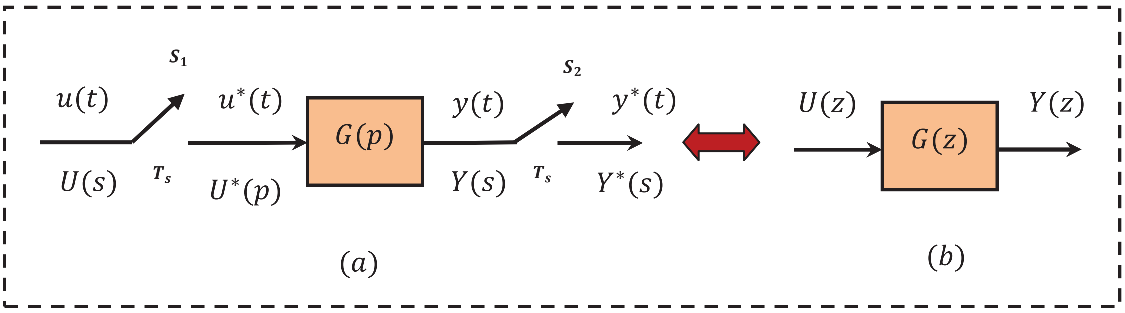


Fig.3. 21: **(a)** Block diagram of continuous time open loop system surrounded by two samplers, **(b)** Block diagram of the equivalent discrete system

In this case, the equivalent transfer function of the discrete system is determined according to the following procedure:

- Before the sampler S_1, we have: $u(t) \mapsto \overset{L}{\rightarrow} U(s)$
- After the sampler S_1, we have: $u(t) \mapsto \overset{S_1}{\rightarrow} u^*(t) \quad \Rightarrow U(s) \mapsto \overset{S_1}{\rightarrow} U^*(s)$
- Before the sampler S_2, we have: $y(t) \mapsto \overset{L}{\rightarrow} Y(s) = U^*(s).G(s)$
- After the sampler S_2, we have: $y(t) \mapsto \overset{S_2}{\rightarrow} y^*(t) \quad \Rightarrow Y(s) \mapsto \overset{S_2}{\rightarrow} Y^*(s)$

Where L denotes the Laplace transform operator.

Using the operation principles applied on the given block diagram of **Fig.3. 21 (a)**, we can perform the following:

$$Y^*(s) = [Y(s)]^* = [U^*(s).G(s)]^* = U^*(s)[G(s)]^* = U^*(s)G^*(s)$$

$$\Rightarrow \begin{cases} Y^*(s) \mapsto \xrightarrow{z=e^{s.T_S}} Y(z) \\ U^*(s) \mapsto \xrightarrow{z=e^{s.T_S}} U(z) \\ G^*(s) \mapsto \xrightarrow{z=e^{s.T_S}} G(z) \end{cases}$$

From which, we obtain :

$$Y(z) = U(z)G(z)$$

Consequently :

$$Y(z) = U(z)G(z) \Rightarrow G(z) = \frac{Y(z)}{U(z)} = \frac{\mathcal{Z}[Y(s)]}{\mathcal{Z}[U(s)]} = \mathcal{Z}[G(s)]$$

Hence, we can state and apply the following rule which is used to determine the equivalent discrete transfer function corresponding to a continuous time system situated between two samplers.

$$G(z) = \frac{Y(z)}{U(z)} = \frac{\mathcal{Z}[Y(s)]}{\mathcal{Z}[U(s)]} = \mathcal{Z}[G(s)] \quad (3.20)$$

With $\mathcal{Z}$ denotes the z transform operator.

5.1.2 Case of two cascaded continuous time systems with an intermediate sampler

This case is illustrated by the following **Fig.3. 22**.

To determine the equivalent discrete transfer function for this case, we can follow similar steps as in the previous case and we arrive to the following result:

$$G(z) = \frac{Y(z)}{U(z)} = \frac{\mathcal{Z}[Y(p)]}{\mathcal{Z}[U(p)]} = \mathcal{Z}[G_1(s)]\mathcal{Z}[G_2(s)] \quad (3.21)$$

With $\mathcal{Z}$ denotes the z transform operator.

Expression (3.21) can also be written as:

$$G(z) = G_1(z)G_2(z) \quad (3.22)$$

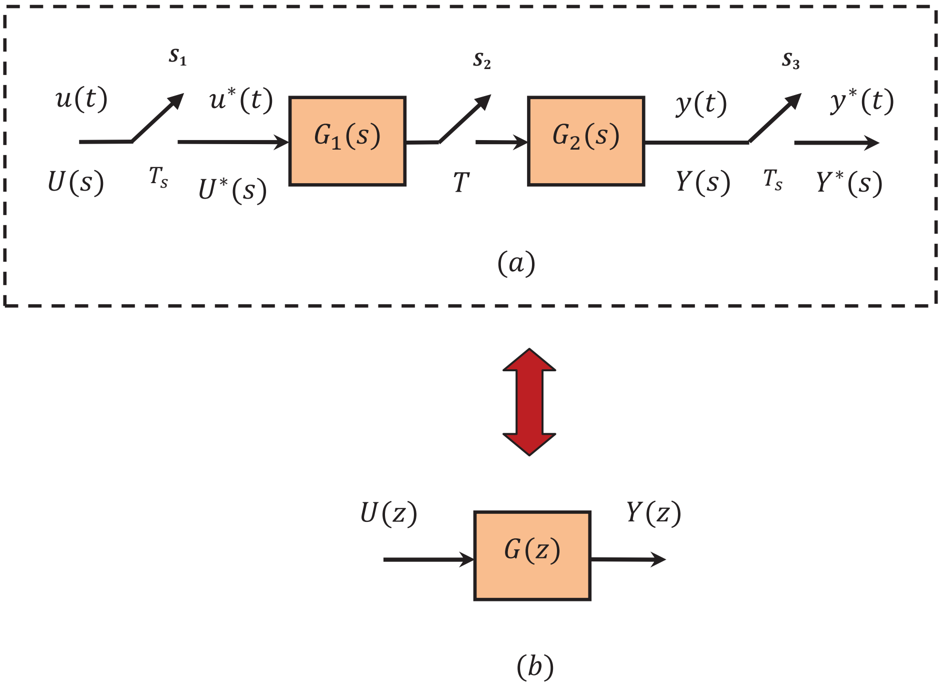


Fig.3. 22: **(a)** Block diagram of two cascade continuous time system with an intermediate sampler

(b) Block diagram of the equivalent discrete system

5.1.3 Case of two cascaded continuous systems without an intermediate sampler

This can be illustrated as in **Fig.3.23**

By applying a similar procedure, we determine the equivalent discrete transfer function for this case and we end up to the following result:

$$G(z) = \frac{Y(z)}{U(z)} = \frac{\mathcal{Z}[Y(s)]}{\mathcal{Z}[U(s)]} = \mathcal{Z}[G_1(s)G_2(s)] \qquad (2.23)$$

With $\mathcal{Z}$ denotes the z transform operator.

Expression (3.23) can also be written as:

$$G(z) = \frac{Y(z)}{U(z)} = [G_1G_2](z) \qquad (3.24)$$

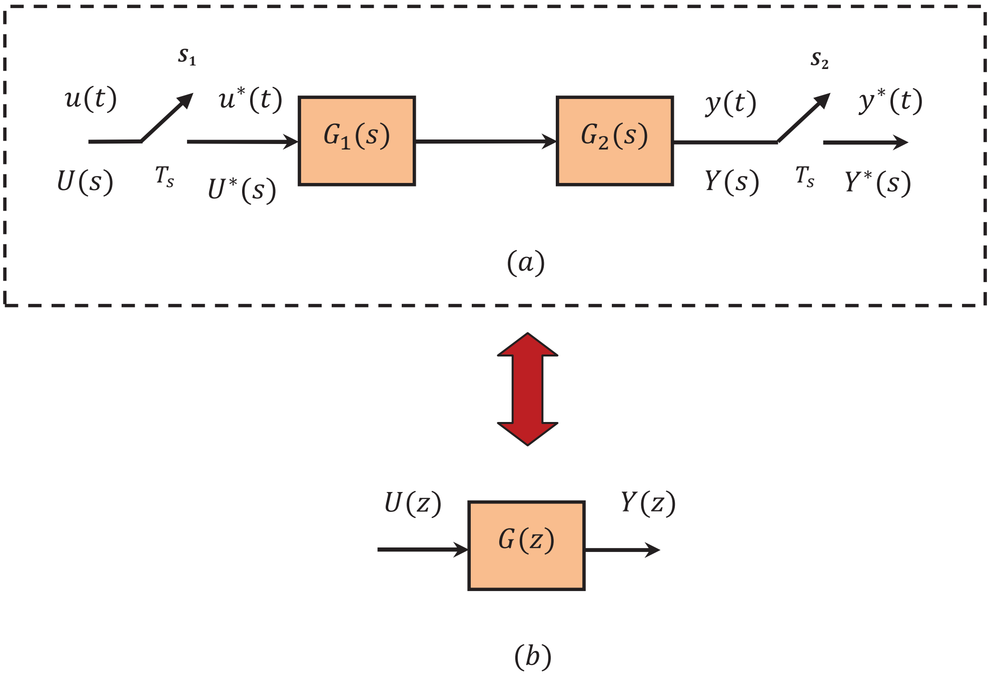


Fig.3. 23: (a) Block diagram of two cascade continuous time system without an intermediate sampler

(b) Block diagram of the equivalent discrete system

5.2. Case of Closed Loop discrete control system

In fact the determination of the equivalent discrete transfer function of a closed loop discrete (sampled data) control system is not unique even for the same structure and topology, because this fundamentally depends on both the number of samplers used and their locations within the system's structure. For the sake of illustration, we will work out a typical closed loop discrete control system represented by the following block diagram of **Fig.3.24**.

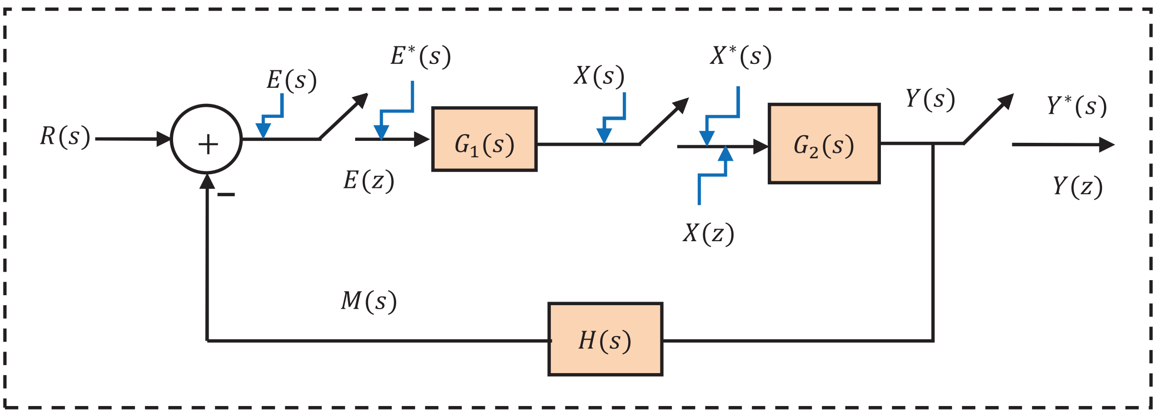


Fig.3. 24: Block diagram of typical sampled data closed loop control system

To determine the equivalent discrete transfer function of the closed loop sampled data control system described by the block diagram of **Fig.3.24**, we proceed as follows:

We have:

$$E^*(s) = [E(s)]^* = [R(s) - M(s)]^* = [R(s) - H(s)Y(s)]^* \tag{3.25}$$

By applying a virtual sampler at the input, we get:

$$[R(s) - H(s)Y(s)]^* = R^*(s) - [H(s)Y(s)]^* \tag{3.26}$$

From (3.18) and (3.19), we can write:

$$E^*(s) = [E(s)]^* = R^*(s) - [H(s)G_2(s){G_1}^*(s)E^*(s)]^* \tag{3.27}$$

At the sampling time instant, (3.27) gives:

$$E^*(s) = [E(s)]^* = R^*(s) - [H(s)G_2(s)]^*{G_1}^*(s)E^*(s) \tag{3.28}$$

From which we can write:

$$\{1 + [H(s)G_2(s)]^*{G_1}^*(s)\}E^*(s) = R^*(s) \tag{3.29}$$

On the other hand:

$$Y^*(s) = [Y(s)]^* = [G_2(s)X^*(s)]^* = {G_2}^*(s)X^*(s) = {G_2}^*(s){G_1}^*(s)E^*(s) \tag{3.30}$$

From (3.30), we can write:

$$E^*(s) = \frac{Y^*(s)}{{G_2}^*(s){G_1}^*(s)} \tag{3.31}$$

By substituting (3.31) into (3.29), we obtain:

$$\{1 + [H(s)G_2(s)]^*{G_1}^*(s)\}\frac{Y^*(s)}{{G_2}^*(s){G_1}^*(s)} = R^*(s) \tag{3.32}$$

After some manipulating steps and simplification, we end up to the following result:

$$\frac{Y^*(s)}{R^*(s)} = \frac{{G_2}^*(s){G_1}^*(s)}{1 + [H(s)G_2(s)]^*{G_1}^*(s)}R^*(s) \tag{3.33}$$

Considering the sampling time instant, we apply Z transform on both sides of **(3.33)** to get:

$$Z\left\{\frac{Y^*(s)}{R^*(s)}\right\} = Z\{G^*(s)\} = Z\left\{\frac{{G_2}^*(s){G_1}^*(s)}{1+[H(s)G_2(s)]^*{G_1}^*(s)}R^*(s)\right\} \tag{3.34}$$

Using the previous notions on Z transform application, the following final result represents the determined equivalent discrete transfer function that corresponds to the typical closed loop sampled data control system described by the block diagram of **Fig.3.24**.

$$G(z) = \frac{Y(z)}{R(z)} = \frac{G_1(z)G_2(z)}{1+[HG_2](z)G_1(z)} \tag{3.35}$$

Where:

$$\begin{cases} G(z) = Z\{G^*(s)\} \\ Y(z) = Z\{Y^*(s)\} \\ R(z) = Z\{R^*(s)\} \\ G_1(z)G_2(z) = Z\{{G_1}^*(s){G_2}^*(s)\} \\ G_1(z) = Z\{{G_1}^*(s)\} \\ [HG_2](z) = Z\{[H(s)G_2(s)]^*\} \end{cases} \tag{3.36}$$

The above development and analysis can be illustrated as it is shown in **Fig.3.25**

Based on the aforementioned demonstration and development used to determine the equivalent discrete transfer function of the typical closed loop sampled data control system, and due to the fact that this transfer function changes from one structure to another depending on the number of samplers and their locations within the system, the following closed loop sampled data control systems are further given and can be encountered. The reader can apply the same procedure to determine their corresponding equivalent discrete transfer functions.

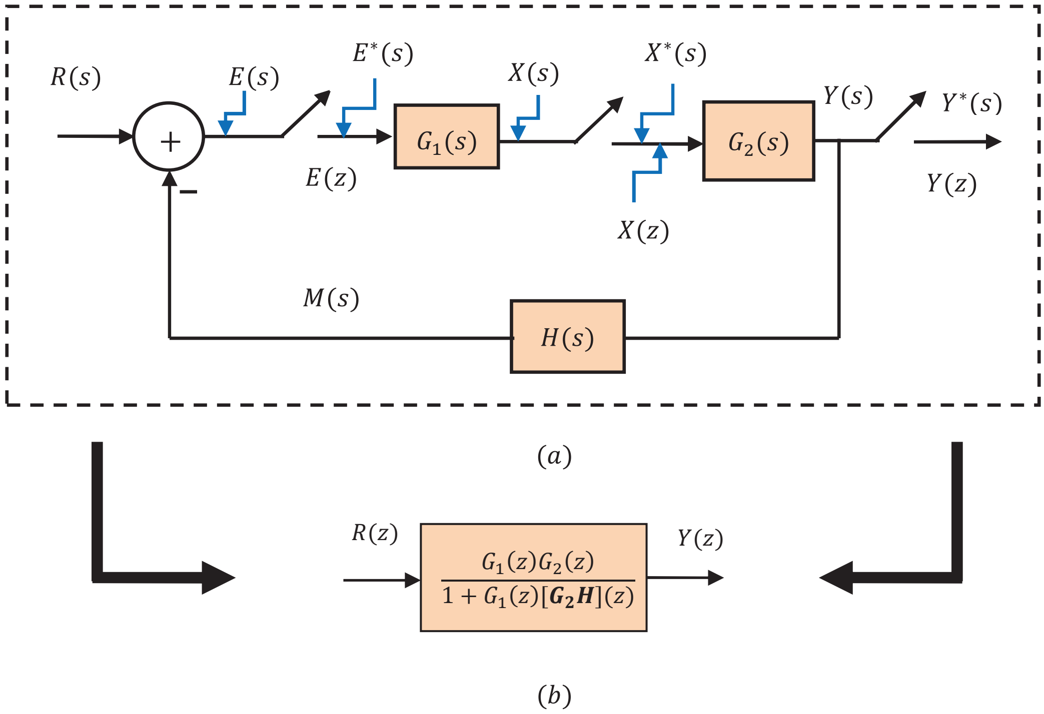
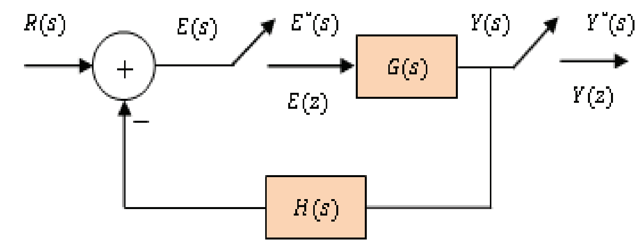
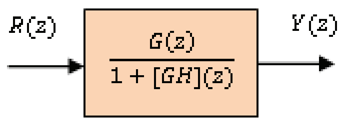
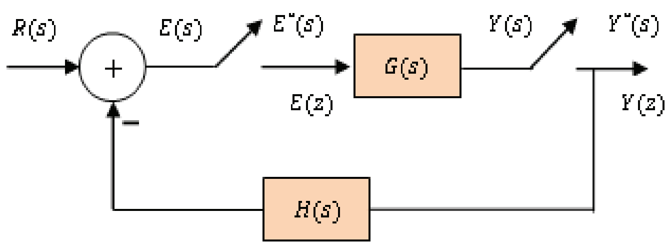
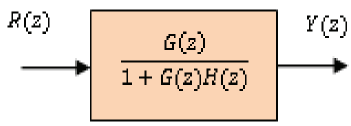
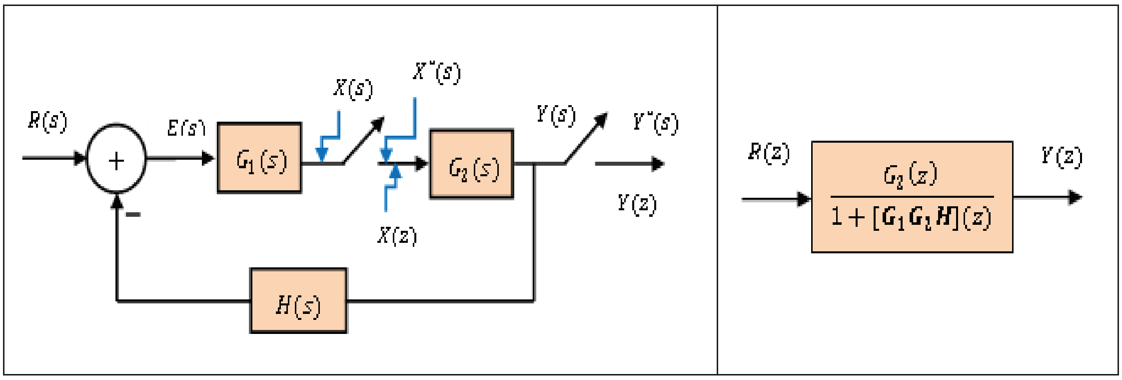


Fig.3.25: (a) Block diagram of typical sampled data closed loop control system

(b) Block diagram of the equivalent discrete control system

Table 3.1 Possible closed loop structures of sampled data control system and their Equivalent discrete transfer functions

Possible closed loop structure of sampled data control system	**Equivalent discrete transfer function**
	
	



6. Equivalent transfer function of Associated LTI discrete systems

Linear Time Invariant (LTI) discrete systems represented by their corresponding discrete transfer functions can be associated using three topologies which are:

- Serial structure association.
- Parallel structure association.
- Feedback structure association.

In this section, we will show how to determine the equivalent discrete transfer function of discrete systems when connected under these structures.

6.1. Case of two discrete systems associated in series

The equivalent discrete transfer function of two discrete systems connected in series (also called two systems in cascade) is determined as it is illustrated in the following **Fig.3.26**.

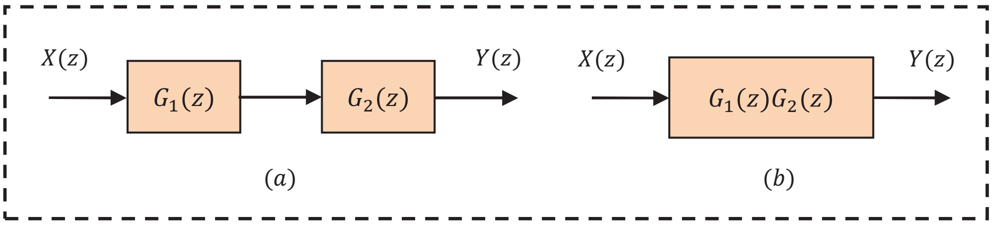


Fig.3.26 Association of two discrete systems in series, **(a)** original block diagram, **(b)** The equivalent block diagram and the corresponding discrete transfer function

6.2. Case of two discrete systems associated in parallel

In the following figure (**Fig.3.27**) we illustrate how the equivalent discrete transfer function corresponding to two discrete systems connected in parallel is determined.

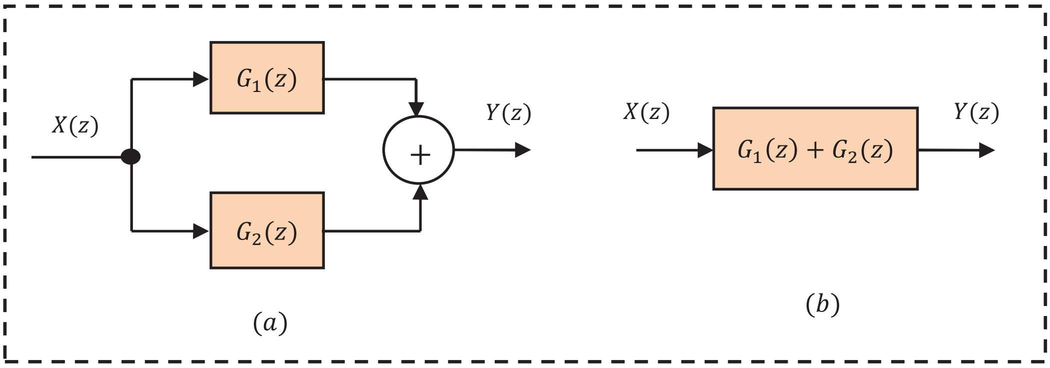


Fig.3.27 Association of two discrete systems in parallel, **(a)** original block diagram, **(b)** The equivalent block diagram and the corresponding discrete transfer function

6.3. Case of two discrete systems associated in feedback structure

When two discrete systems are associated in a feedback structure, the equivalent block diagram with the corresponding equivalent discrete transfer function is determined as it is shown and illustrated in the following **Fig.3.28**.

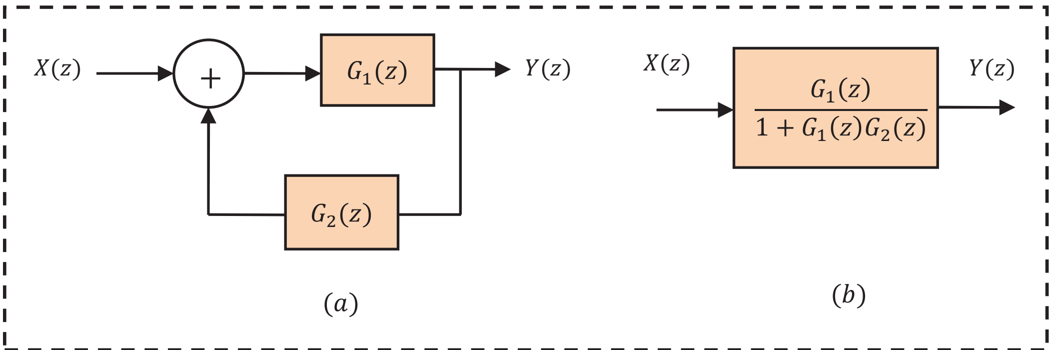


Fig.3.28 Association of two discrete systems in feedback structure, **(a)** original block diagram, **(b)** the equivalent block diagram and the corresponding discrete transfer function

Chapter 4: Performance Analysis of discrete linear feedback control systems

The main objective of designing any feedback control system is to ensure its stability in general conditions but also this stability is required to be robust against some environmental properties and influences. This chapter will be fundamentally devoted to tackle in study and analysis the main performance properties of discrete and linear feedback control systems where the behavior of each system is represented by the corresponding discrete (or Z) transfer function. Mainly we will focus on how to study the stability and accuracy performance indices.

1. Stability of discrete control systems

In general, the stability of any closed loop control system is related to the value of the amplitude of the steady state response. In other words, the feedback control system can be viewed stable if the amplitude of its output response has a finite value as time tends to infinity. Conversely, the response amplitude value is unbounded, the control system is unstable.

In this chapter our interest will be focused on defining and discussing the stability of linear time invariant discrete systems from two different perspectives; that is the stability regarding the steady state magnitude of the system response, where the asymptotic and marginal stability are to be defined. Similarly, the stability regarding the input and output characteristics which is called the stability in the sense of bounded input and bounded output (BIBO) will also be defined. Whereas the second perspective will be that of the stability based on pole locations of the corresponding discrete transfer function used to describe the behavior of the system.

1.1. Asymptotic and Marginal Stability

The definition of asymptotic and marginal stability is mainly related to the way and manner the system's output response responds to the initial conditions. If it decays asymptotically to zero corresponding to the applied initial conditions, then we are talking about the asymptotic stability. We mathematically express this as:

$$\lim_{k\to\infty} y(k) = 0 \tag{4.1}$$

Where:

$y(k)$: is the output response of the discrete system.

If however, the system's response does not decay to zero due to the applied initial condition and stays bounded, in this case we are talking about marginal stability. We can mathematically express this as:

$$\lim_{k \to \infty} y(k) = C\,, \quad C \neq 0 \tag{4.2}$$

Where:

C: is any constant.

1.2. Bounded Input-Bounded Output Stability

The bounded input bounded output stability is defined based on the forced response with respect to the applied bounded input. To understand this definition, we introduce and define mathematically the bounded discrete input as follows:

if $u(k)$ be a discrete signal with: $k = 0,1,2,3, \dots \dots .\,, \infty$. then $u(k)$ is said to be bounded if its samples are upper limited; in other words, there exists a real number $M = B_u > 0$, such that:

$$\forall\, k \in \mathbb{N}, |u(k)| < M \tag{4.3}$$

Consequently and based on the steady state amplitude or the magnitude of the output response, we can mathematically define the bounded input, bounded output stability as:

$$|u(k)| < b_u, \quad \begin{cases} k = 0,1,2,, \dots \dots \\ 0 < b_u < \infty \end{cases} \Rightarrow |y(k)| < b_y, \quad \begin{cases} k = 0,1,2,, \dots \dots \\ 0 < b_y < \infty \end{cases} \tag{4.4}$$

With:

$u(k), y(k)$: Are respectively the input and output of the discrete system.

b_u, b_y: Are respectively the input and output upper bounds.

1.3. Stability Definition in the Complex Z plane

1.3.1. Mapping between s and z complex planes

What is more important regarding the stability analysis and study of any linear time invariant closed loop control system whatever whether it is continuous time or discrete time (or sampled data) is the absolute and relative stability which are also known respectively as asymptotic and marginal stability. These two types of stability are hopefully determined by observing the location of the poles of the system's closed loop transfer function in a complex plane formed by the two perpendicular axes called respectively as real axis and imaginary axis.

Concerning the discrete feedback control system stability analysis, a mapping between the continuous time complex plane (s-plane) and the discrete time complex plane (z-plane) is generally established. As it is known, a continuous time closed loop control system stability is studied and analyzed according to the location of its poles in the complex s-plane in such a way the system is said to be absolutely stable when all its poles are located on the left hand side (LHS) of the s-plane, but if at least one pole is found on the Right Hand Side (RHS) of the s-plane, the system then is said to be unstable. Analogously, we can apply the same approach to determine the stability state of the discrete time control system but in the z-plane which is jointly related to the s-plane according to the following demonstration.

We have that the complex variables s and z are related by:

$$z = e^{sT_s} \tag{4.5}$$

With:

T_s: is the sapling period.

s: is the Laplace complex variable.

If we consider that:

$$s = \sigma + j\omega \tag{4.6}$$

Where:

σ and ω are respectively the real part value and the imaginary part value of the complex variable.

Substituting (4.6) in (4.5), we obtain:

$$z = e^{(\sigma + j\omega)T_s} = e^{(\sigma T_s + j\omega T_s)} = e^{(\sigma T_s)}.e^{(j\omega T_s)} \qquad (4.7)$$

Using Euler's formula, we can write (4.7) as:

$$z = e^{(\sigma T_s)}.[\cos(\omega T_s) + j\sin(\omega T_s)] = e^{T_s\sigma}\lfloor\omega T_s = |z|\lfloor\omega T_s \qquad (4.8)$$

We can distinguish three different cases regarding the value of the parameter σ. These are:

- **Case of $\boldsymbol{\sigma} < 0$**, which corresponds to the left hand side of the s-plane, and this gives:

  $$|z| = e^{T_s\sigma} < 1$$

  That is the left hand side of s-plane corresponds to the inside the unity circle of the z-plane.

- **Case of $\boldsymbol{\sigma} = \mathbf{0}$**, which corresponds to the imaginary axis of the s-plane, and this gives:

  $$|z| = e^{T_s\sigma} = 1$$

  That is the imaginary axis of s-plane corresponds to the boundary of the unity circle of the z-plane.

- **Case of $\boldsymbol{\sigma} > 0$**, which corresponds the right hand side of s-plane, and this gives:

  $$|z| = e^{T_s\sigma} > 1$$

  That is the right hand side of s-plane corresponds to the outside of the unity circle of the z-plane.

Note:

It is important to notice that the unit circle is defined in the complex z-plane as the circle of radius equals one ($r = 1$) and centered at the point (0,0).

From this demonstration we can establish a mapping between s-plane and z-plane. This mapping is illustrated in **Fig.4.1**.

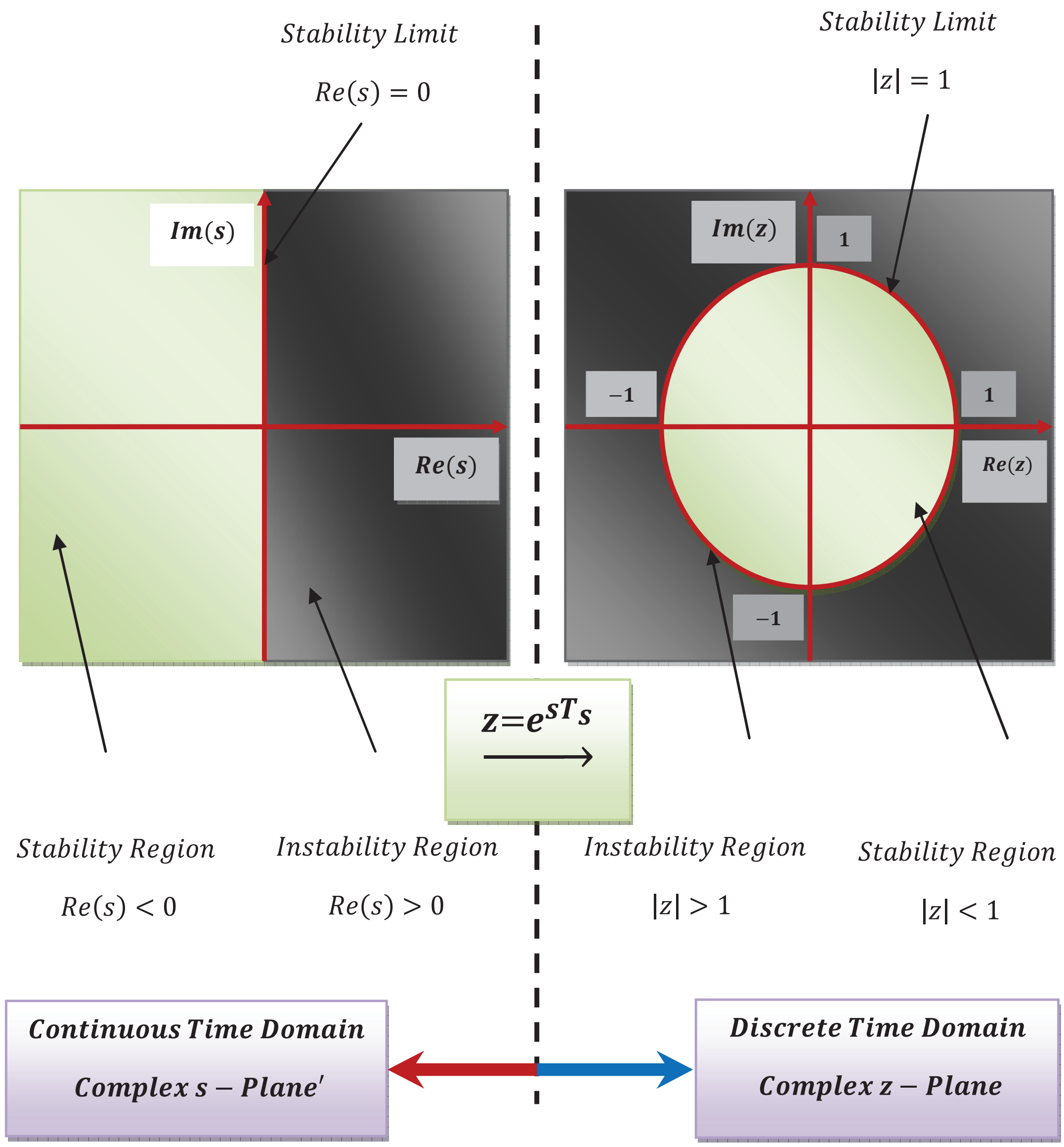


Fig.4.1 Mapping between s-plane and z-plane

1.3.2. Stability Theorem of discrete control system

From this mapping and correspondence between s-plane and z-plane, and by analogy with the analog feedback control system, the stability of linear time invariant discrete (sampled data) control system can be studied and analyzed according to the following results:

- The discrete feedback control system is said to be stable when all the poles of the corresponding discrete or z transfer function are located inside the unit circle of z-plane. In other words, for a discrete control system of order n, if

$p_i, i = 1,2,3, \dots \dots, n$ are its poles, then we say that this system is **stable** if and only if the following condition is satisfied:

$$\forall\, i, |p_i| < 1, \qquad i = 1,2,3, \dots \dots ., n$$

- The discrete feedback control system is said to be unstable when there exists at least one pole of its corresponding z transfer function is located outside the unit circle of the z-plane. For a discrete control system of order n, we express this as:

$$if\ \exists\ p_i, such\ that\ |p_i| > 1\ \Leftrightarrow the\ discrete\ system\ is\ unstable$$

1.3.3. Stability conditions based on discrete transfer function and pole location

The following stability conditions of discrete feedback control system which is represented by its z transfer function can be stated. The analysis and study of the different stability types can be easily done using these conditions and in accordance to the pole location.

1.3.3.1. Asymptotic and Marginal Stability

In order to determine whether a given discrete closed loop control system, which is represented by its discrete (also called) impulse transfer function is asymptotically or marginally stable, we accept without proof the following theorems.

Theorem 4.1: **Asymptotic and Marginal Stability**

In the absence of pole-zero cancellation, a Linear Time Invariant discrete (sampled data) control system is ***asymptotically stable*** if ***all*** of its z transfer function ***poles*** are located inside the unit circle of z-plane. If it exists ***at least one pole*** which is located on the boundary of the unit circle of z-plane, the so-called system is said to be ***marginally stable***. In the following example, we illustrate how to apply this theorem to test the asymptotic and marginal (critical) stability of discrete control systems.

1.3.3.2. Bounded Input Bounded Output (BIBO) Stability

Regarding the discrete control system which is represented by its z transfer function and based on the principle of pole location in the z-plane, we can also test whether it is bounded input bounded output stable. To do so, the following theorem states the necessary and sufficient condition for BIBO stability.

Theorem 4.2: Stability in the sense of BIBO

If there is a pole-zero cancellation, a discrete LTI closed loop control system is ***Bounded input-Bounded output*** stable if and only if ***all the poles*** of its corresponding z transfer function are located inside the unit circle.

1.3.3.3. Some Results and Discussion

The following notes can be pointed out in respect of the stability analysis of discrete control system.

- In the case of there is no pole-zero cancellation in the z transfer function representing the LTI discrete control system, ***asymptotic stability*** is identical to ***Bounded Input-Bounded Output stability***.
- These two theorems applied to determine the stability of discrete closed loop control system is also valid for the same purpose of discrete open loop control system.

Based on the above remarks and by combining the two theorems (4.1) and (4.2), we can conclude that even if there is a pole-zero cancellation, the asymptotic or marginal stability of the system can be concluded provided that the cancelled pole(s) is (are) stable; in other words, it (they) is (are located) inside or on the boundary of the unit circle of z-plane. This is true because stability determination is in fact the look for unstable poles. With the system declared stable, it means none is found. However, stable but hidden poles don’t lead to wrong conclusion about the stability. On the other hand, hidden and unstable poles do lead to draw wrong conclusion. We can notice this important result in the following example.

Example 4.1

Consider the linear time invariant control systems represented by the discrete transfer functions as below:

1) $G_1(z) = \frac{4(z-2)}{(z-2)(z-0.1)}$

2) $G_2(z) = \frac{4(z-0.2)}{(z-0.2)(z-0.1)}$

3) $G_3(z) = \frac{5(z-0.3)}{(z-0.2)(z-0.1)}$

4) $G_4(z) = \frac{8(z-0.2)}{(z-1)(z-0.1)}$

We would like to determine whether these discrete transfer functions are asymptotically, marginally or Bounded Input-Bounded Output stable.

Answer 4.1

By referring to the two aforementioned theorems, the transfer functions $G_1(z)$ and $G_2(z)$ both of them present a pole-zero cancellation and their remaining poles are located inside the unit circle ($p_1 = 0.1, p_1 = 0.1\ and\ |p_1 = 0.1| < 1$), hence these transfer functions are both BIBO stable.

Concerning the discrete control systems represented respectively by the transfer functions $G_3(z)$ and $G_4(z)$, it is obvious that there is no pole-zero cancellation. Since $G_3(z)$ have two poles $p_1 = 0.2, p_2 = 0.1\ and\ |p_1 = 0.2| < 1, |p_2 = 0.1| < 1$; that is both of them are located inside the unit circle and $G_3(z)$ is ***asymptotically stable*** (theorem 4.1).

For the transfer function $G_4(z)$, it has two poles $p_1 = 1, p_2 = 0.1\ and\ |p_1 = 1| = 1, |p_2 = 0.1| < 1$; that is one pole is located on the boundary of the unit circle which indicates that it is ***marginally*** (***critically***) **stable** (theorem 4.1).

As we can notice, the transfer function $G_1(z)$ although it contains one unstable hidden pole ($p_1 = 2$), it is however stable in sense of BIBO.

Consequently, BIBO stability is not always enough to judge and conclude about the absolute stability of discrete control system.

2. Stability Criteria applied to Discrete control systems

In this section, we will present and discuss some criteria used to judge and determine whether a discrete (digital) control system is stable or not by viewing and analyzing its open loop or closed loop z transfer function.

Several criteria (also called tests) are available and can be used to determine and test the stability of a discrete control system. Mostly used are:

- Jury stability criterion
- Routh Hurwitz stability criterion
- Nyquist stability criterion

2.1. Jury Criterion of Stability

Jury criterion of stability is an algebraic method which is used to evaluate and test the stability of a discrete LTI control system whose discrete transfer function is known. Determining the stability of the system using this criterion is based solely on the known coefficients of the characteristic polynomial of either the discrete open loop or closed loop transfer function.

If we consider that the discrete transfer function of the control system is expressed as a ration of the numerator and denominator polynomials as:

$$G(z) = \frac{Y(z)}{R(z)} = \frac{N(z)}{D(z)} = \frac{b_0 z^m + b_1 z^{m-1} + \cdots . + b_{m-1} z^1 + b_m}{a_0 z^n + a_1 z^{n-1} + \cdots . + a_{n-1} z^1 + a_n}, \quad n \geq m, \qquad (4.9)$$

With:

$Y(z), R(z)$: are respectively the z transform of the input and output of the control system.

$a_0, a_1, \ldots, a_{n-1}, a_n; \quad b_0, b_1, \ldots, b_{m-1}, b_m$: are known real coefficients, where $a_0 > 0$ is a necessary condition for the use of Jury criterion.

In this transfer function, the characteristic polynomial is determined to be:

$$D(z) = a_0 z^n + a_1 z^{n-1} + \cdots . + a_{n-1} z^1 + a_n \qquad (4.10)$$

2.1.1. Jury Table

The first step in using Jury criterion to evaluate and determine the stability of the control system is by constructing the so-called Jury table. This is done as mentioned below:

Table 4.1 Jury Table

line	z^0	z^1	z^2	z^3	………..	z^{n-2}	z^{n-1}	z^n
1	a_n	a_{n-1}	a_{n-2}	a_{n-3}	………..	a_2	a_1	a_0
2	a_0	a_1	a_2	a_3	………..	a_{n-2}	a_{n-1}	a_n
3	B_{n-1}	B_{n-2}	B_{n-3}	B_{n-4}	………..	B_1	B_0	
4	B_0	B_1	B_2	B_3	………..	B_{n-2}	B_{n-1}	
5	C_{n-2}	C_{n-3}	C_{n-4}	C_{n-5}	………..	C_0		
6	C_0	C_1	C_2	C_3	………..	C_{n-2}		
. . .	. . .	. . .	. . .	. . .	. . .			
$\mathbf{2n-5}$	P_3	P_2	P_1	P_0				
$\mathbf{2n-4}$	P_0	P_1	P_2	P_3				
$\mathbf{2n-3}$	Q_2	Q_1	Q_0					

Where:

B_i, C_i, P_i, Q_i: are inserted real coefficients and are determined as:

$$B_i = \begin{vmatrix} a_n & a_{n-1-i} \\ a_0 & a_{i+1} \end{vmatrix}, \qquad i = 0,1,2,3, \dots \dots ., n-1$$

$$C_i = \begin{vmatrix} B_{n-1} & B_{n-2-i} \\ B_0 & B_{i+1} \end{vmatrix}, \qquad i = 0,1,2,3, \dots \dots ., n-2$$

.
.
.

$$Q_i = \begin{vmatrix} P_3 & P_{2-i} \\ P_0 & P_{i+1} \end{vmatrix}, \qquad i = 0,1,2$$

2.1.2. Jury criterion statement

The use of Jury criterion as a tool to determine and direct test of the stability of the discrete control system is based on the satisfaction of the following corresponding conditions.

Consequently, a discrete control system which is characterized by its characteristic polynomial:

$$D(z) = a_0 z^n + a_1 z^{n-1} + \cdots . + a_{n-1} z^1 + a_n \quad , \quad a_0 > 0$$

Is said to be stable if and only if all of the following conditions are simultaneously satisfied:

(1) $|a_n| < a_0$

(2) $D(1) > 0$

(3) $(-1)^n D(-1) > 0$

(4) $|B_{n-1}| > |B_0|$

(5) $|C_{n-2}| > |C_0|$

.
.
.

$|Q_2| > |Q_0|$

2.1.3. Application procedure

Jury criterion of stability is a powerful tool for studying the stability of control systems. It allows us, however, to only determine whether the discrete open or closed loop control system is stable or not stable and no more information and details about, for instance, the pole locations. In order to correctly use and apply this tool, a basic procedure is indeed required.

First of all, we need to determine number of lines to be used in the Jury table. This depends on the order of the discrete control system under study; where we use the general formula given as:

$$2n - 3 \tag{4.11}$$

With $'n'$ represents the order of the control system.

After that, we construct the table as it is shown in Table 4.1, where the different unknown coefficients are calculated as it is indicated.

The last step is to check and verify the Jury conditions for stability; if all these conditions are simultaneously satisfied, the control system of interest is stable. But if at least one condition is not satisfied, the system will be judged unstable.

To illustrate the used of this procedure, we consider the following simple example.

Example 4.2

We assume that the discrete control system has the characteristic polynomial given by:

$$D(z) = z^2 + 9z + 8$$

Use Jury stability criterion and determine the stability of the system.

Answer 4.2

Firstly, we determine the number of lines of the Jury table.

We have:

$$n = 2, \qquad a_0 = 1, \qquad a_1 = 9, \qquad a_n = a_2 = 8$$

Which gives:

$$The\ number\ of\ lines = 2n - 3 = 1$$

Consequently, the Jury table corresponding to our discrete control system is as shown below:

$ligne$	z^0	z^1	z^2
1	a_2	a_1	a_0

After we have constructed the Jury table, we move to check and verify the stability conditions, we have:

(1) $|a_2 = 8| < 1$

(2) $D(1) = 1 + 9 + 8 = 18 > 0$

(3) $(-1)^2 D(-1) = 1 - 9 + 8 = 0 > 0$

We notice that conditions (1) and (3) are not satisfied, which lead us to conclude the instability of the given discrete control system.

Notice that:

In the above explanation of using Jury stability criterion, we have assumed that:

$$a_0 > 0$$

In case when $a_0 < 0$, the idea is :

- To obtain a new characteristic polynomial: $P(z) = -D(z)$
- Then, we apply Jury stability criterion with the polynomial $P(z)$.

2.2. Routh Hurwitz Stability Criterion

Routh Hurwitz method is another criterion widely used to determine and test the stability feedback control systems. However, this method is most familiar to be directly applied for the stability study and determination of continuous time control systems. In this vein, to be able of using this method for discrete control system, we need first to explain its use for exploring the stability of continuous time system.

Consider the continuous time feedback control system defined and described by transfer function in the s plane as:

$$G(s) = \frac{Y(s)}{R(s)} = \frac{b_m s^m + b_{m-1} s^{m-1} + \cdots . + b_1 s^1 + b_0}{a_n s^n + a_{n-1} s^{n-1} + \cdots . + a_1 s^1 + a_0}, \quad n \geq m, a_n > 0 \qquad (4.12)$$

With:

$R(s), Y(s)$: are respectively are the Laplace transform of the input and the output of the control system.

We define the characteristic polynomial of the system's transfer function to be:

$$D(s) = a_n s^n + a_{n-1} s^{n-1} + \cdots . + a_1 s^1 + a_0$$

The Routh Hurwitz criterion of stability uses the parameters of the characteristic polynomial to firstly construct the so-called Routh table as shown in **Table 4.2**:

Table 4.2 Routh Hurwitz stability table

s^n	a_n	a_{n-2}	a_{n-4}	a_{n-6}	… … … ..
s^{n-1}	a_{n-1}	a_{n-3}	a_{n-5}	a_{n-7}	… … … ..
s^{n-2}	b_1	b_2	b_3	… … … ..	
s^{n-3}	c_1	c_2	c_3	… … … ..	
s^{n-4}	. . .	. . .	. . .		
s^1	j_1				
s^0	k_1				

The first two rows of the Routh table are formed by just listing the coefficients of the characteristic polynomial as it is shown.

For the new coefficients b_i and c_i which are inserted and entered in the subsequent rows of the table, they are calculated as follows:

$$b_1 = \frac{\begin{vmatrix} a_{n-1} & a_{n-3} \\ a_n & a_{n-2} \end{vmatrix}}{a_{n-1}}$$

$$b_2 = \frac{\begin{vmatrix} a_{n-1} & a_{n-5} \\ a_n & a_{n-4} \end{vmatrix}}{a_{n-1}}$$

$$b_3 = \frac{\begin{vmatrix} a_{n-1} & a_{n-7} \\ a_n & a_{n-6} \end{vmatrix}}{a_{n-1}}$$

The process of calculating the coefficients b_i should continue until the remaining values of $b's$ are all zero. When it is the case, we proceed to calculate the other inserted coefficient, that is:

$$c_1 = \frac{\begin{vmatrix} b_1 & b_2 \\ a_{n-1} & a_{n-3} \end{vmatrix}}{b_1}$$

$$c_2 = \frac{\begin{vmatrix} b_1 & b_3 \\ a_{n-1} & a_{n-5} \end{vmatrix}}{b_1}$$

$$c_3 = \frac{\begin{vmatrix} b_1 & b_4 \\ a_{n-1} & a_{n-7} \end{vmatrix}}{b_1}$$

Similarly, the process of calculating the coefficients c_i should continue until the remaining values of $c's$ are all zero.

The construction of the Routh table is continued in a similar manner until we finish with all the inserted and introduced coefficients where the Routh array will always be terminated with the two rows: s^1, s^0 which contain only respectively the two elements: j_1, k_1.

2.2.1 Statement of Routh Hurwitz Stability Criterion

Given a continuous time control system which is defined and represented by its transfer function $G(s)$, therefore, Routh criterion of stability states that ***all the roots of the characteristic polynomial of the transfer function (poles) have negative real parts if and only if the coefficients (elements) of the first column of the Routh table have the same sign***. ***On the other hand, the number of roots of the characteristic polynomial (poles of*** $G(s)$***) with positive real parts is equal to the number of sign changes occurred on the coefficients (elements) of the first column of Routh table***.

Consequently, the control system is stable if and only if all the coefficients (elements) of the first column of Routh table are of the same sign.

From the above statement, it is obvious that Routh-Hurwitz criterion allows us to just determine the stability of the control system by indicating whether all the poles of transfer function are stable; that is they are all located on the left hand side of the s-plane or some poles instead have positive real part (located on the right hand side of the s-plane).

This stability condition however cannot directly be applied to investigate the stability of discrete-time control systems. Fortunately and thanks of using the so-called the bilinear transformation, it is possible to explore the stability of discrete control system which is represented by its z transfer function, by making transformation from s-plane (also named as w-plane) to z-plane and vice-versa, hence we can apply Routh Hurwitz stability conditions. This bilinear transformation is defined by the following relationship between the continuous complex variable w and the discrete complex variable z:

$$z = \frac{1+w}{1-w} \Leftrightarrow w = \frac{z-1}{z+1} \tag{4.13}$$

By performing this transformation, the inside of unit circle of z-plane is transformed to the left hand side (LHS) of the w-plane and the outside of the unit circle in z-plane is transformed to the right hand side (RHS) of w-plane. This transformation is illustrated in the following drawing of Fig.4.2.

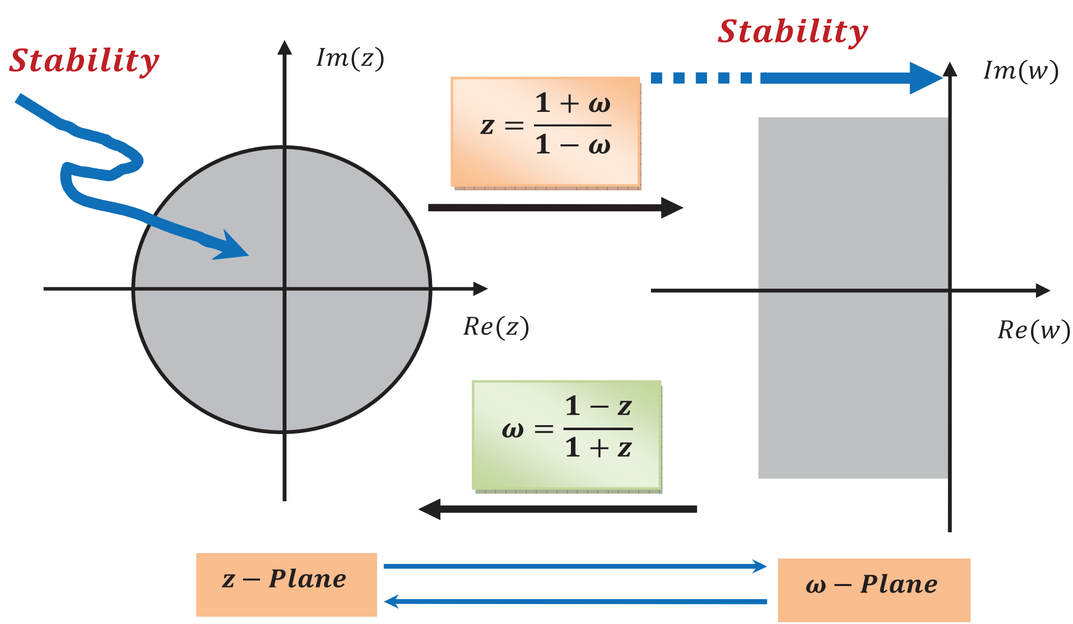


Fig. 4.2 Transforming z-plane into w-plane and vice-versa

Due to this transformation between the two planes, Routh Hurwitz criterion of stability is applied on the discrete control system as it is transformed from z-plane to w-plane. Therefore, the stability conclusion found in w-plane will be valid to judge the stability of the discrete control system in z-plane.

For the sake of illustration, we consider the following example.

2.2.2 Illustrative Example

Using Routh Hurwitz stability criterion, determine whether or not the system described by the following z transfer function is stable.

$$G(z) = \frac{2z+1}{z^3+2z^2+4z+7}$$

Solution:

In order to be able of applying Routh Hurwitz stability criterion on the above discrete control system represented by the z transfer function $G(z)$ for its stability investigation, we first use the bilinear transformation and transforming the study from z-plane to nw-plane.

$$z = \frac{w+1}{w-1}$$

By substituting in the transfer function, we obtain:

$$G(w) = \frac{2\left(\frac{w+1}{w-1}\right)+1}{\left(\frac{w+1}{w-1}\right)^3+2\left(\frac{w+1}{w-1}\right)^2+4\left(\frac{w+1}{w-1}\right)+7}$$

$$G(w) = \frac{\frac{2w+2+w-1}{w-1}}{\frac{(w+1)^3+2(w+1)^2(w-1)+4(w+1)(w-1)^2+7(w-1)^3}{(w-1)^3}}$$

$$G(w) = \frac{\frac{3w+1}{w-1}}{\frac{(w+1)[(w+1)^2+2(w+1)(w-1)+4(w-1)^2]+7(w-1)^3}{(w-1)^3}}$$

$$G(w)=\frac{(3w+1)(w-1)^2}{(w+1)[w^2+2w+1+2w^2-2+4w^2-8w+4]+7(w-1)^3}$$

$$G(w)=\frac{(3w+1)(w-1)^2}{(w+1)[7w^2-6w+3]+7(w-1)(w^2-2w+1)}$$

$$G(w)=\frac{(3w+1)(w-1)^2}{[7w^3-6w^2+3w+7w^2-6w+3]+[7w^3-21w^2+21w-7]}$$

$$G(w)=\frac{(3w+1)(w-1)^2}{14w^3-20w^2+18w-4}$$

Now, we construct the Routh table of the new transfer function $G(w)$ which is characterized by the polynomial:

$$D(w)=14w^3-20w^2+18w-4$$

This gives:

w^3	14	18	0
w^2	-20	-4	0
w^1	b_1	b_2	b_3
w^0	c_1	c_2	c_3

$$b_1=\frac{\begin{vmatrix}-20 & -4\\ 14 & 18\end{vmatrix}}{-20}=\frac{-360+56}{-20}=\frac{-304}{-20}=\frac{76}{5}$$

$$b_2=\frac{\begin{vmatrix}-20 & 0\\ 14 & 0\end{vmatrix}}{-20}=0$$

$$c_1 = \frac{\begin{vmatrix} \frac{76}{5} & 0 \\ 14 & 18 \end{vmatrix}}{\frac{76}{5}} = \frac{\frac{76}{5}.18}{\frac{76}{5}} = 18$$

$$c_2 = \frac{\begin{vmatrix} \frac{76}{5} & 0 \\ 14 & 0 \end{vmatrix}}{\frac{76}{5}} = 0$$

It results the following final Routh table:

w^3	14	18	0
w^2	-20	-4	0
w^1	$\frac{76}{5}$	0	0
w^0	18	0	0

By observing only the first column of the table we notice that:

- Not all the elements (coefficients) of the first column of the array have the same sign. This leads us to conclude that the discrete control system is unstable.
- The number of sign changes among the coefficients of the first column equals two (02), which means that two (02) poles of the discrete transfer function are unstable.

2.2.3 Properties of Routh Hurwitz Stability criterion

From the theoretical and practical point of view of using Routh Hurwitz criterion to determine and investigate the stability of feedback control systems, we can point out the following properties:

1) The use of this stability criterion is completely independent of the order of the system's transfer function. In other words, it is applicable whatever the order of the system but it is particularly more useful for higher order systems.

2) This stability criterion can only tell us whether the control system is stable or not by observing the coefficients of first column of the table. Therefore, no information about the degree of stability (stability margins) can be drawn and concluded.
3) This criterion is not applicable to investigate the stability of feedback control systems which involve time delays.
4) As the last property of Routh Hurwitz stability criterion is that a necessary but not sufficient condition for the poles of the control system transfer function to be all with negative real parts is that all the coefficients of the characteristic polynomial $D(w)$ are of the same sign (all negative or all positive).

As a conclusion to this section and regarding the study and direct determination of the stability of feedback discrete (sampled data) control systems which are represented by a z transfer function, we have explored and discussed the two stability criteria of Jury and Routh Hurwitz as most familiar and widely used methods by the automatic control systems designers and engineers. Also these two methods (criteria) are simple and of straight forward use. Nevertheless, we can find other methods and criteria to test the stability of discrete control systems such as Nyquist method, Bode plot method and others. These methods however are design methods not just direct test of the stability of feedback control systems.

3. Accuracy Analysis of Linear sampled data control systems

The accuracy property is an important performance measure regarding the design and analysis of any feedback control system in general, and in particular for the discrete time control systems due the inherent characteristics caused after the sampling operation and the sampling period [6]. in a given feedback control system, the accuracy performance is measured and analyzed according to the value of the tracking error of the system's response, which is defined, in the time domain, as the difference between the reference signal (desired response), $r(k)$, and the actual measured output, $y(k)$. mathematically, we express this error as:

$$e(k) = r(k) - y(k) \qquad (4.14)$$

Where:

$e(k)$: denotes the tracking error.

Regarding the accuracy of a feedback control system, two types of accuracy performance can be distinguished; namely:

- **Transient (dynamic) accuracy**: which characterize the accuracy behavior of the discrete time control system within the transient state of the system's response.
- **Steady state accuracy**: This describes the accuracy behavior of the discrete time control system at the steady state of the system's response.

We will be interesting of steady state accuracy due to its importance of giving a clear idea about the stability of the control system.

The relationship between the accuracy and the stability of discrete time feedback control system can be described by the following results:

- $\boldsymbol{The\ Control\ System\ is\ Accurate} \Rightarrow \boldsymbol{it\ is\ Stable} \Leftrightarrow \forall\, \boldsymbol{p_i} / \, |\boldsymbol{p_i}| < 1$
- $\boldsymbol{The\ Control\ System\ is\ Unstable} \Leftrightarrow \exists\, \boldsymbol{p_i} / \, |\boldsymbol{p_i}| > 1 \Rightarrow \boldsymbol{it\ is\ inaccurate}$

With:

$p_i: (i = 1,2,3, \ldots, n)$, are the 'n' poles of the z transfer function.

3.1. Steady state Accuracy

The steady state accuracy corresponds to steady state error of the feedback control system. In order to calculate the steady state error, we need to work out the following typical block diagram of a general discrete time unity feedback control system shown in **Fig.4.3**.

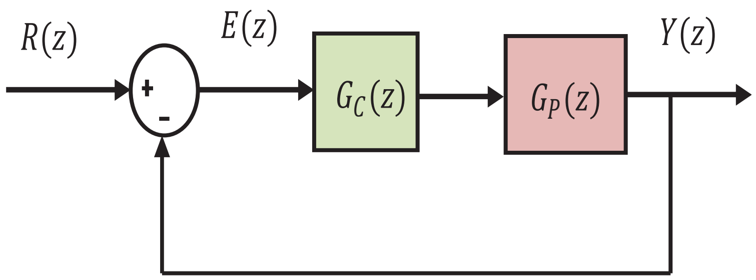


Fig.4.3 Typical discrete unity feedback control system

With:

$R(z), Y(z)$ and $E(z)$ are respectively the reference signal, the output signal and the tracking error signal, all expressed in frequency domain.

$G_C(z)$ and $G_P(z)$ are respectively the controller and the controlled process transfer functions.

We now proceed to calculate the tracking error as follows:

We have:

$$E(z) = R(z) - Y(z) = R(z) - G_C(z)G_P(z)E(z) \Longrightarrow (1 + G_C(z)G_P(z))E(z) = R(z)$$

That is:

$$E(z) = \frac{1}{1 + G_C(z)G_P(z)} R(z) \tag{4.15}$$

The steady state error corresponds to the error $e(k)$ as $k \to \infty$. Using final value theorem discussed in chapter 2, we can write:

$$e_{ss} = e(\infty) \lim_{k \to \infty} e(k) = \lim_{z \to 1} (1 - z^{-1}) E(z) \tag{4.16}$$

Substituting (4.15) in (4.16), we get:

$$e_{ss} = \lim_{z \to 1} (1 - z^{-1}) \frac{1}{1 + G_C(z)G_P(z)} R(z) \tag{4.17}$$

If we define:

$$\frac{1}{1 + G_C(z)G_P(z)} = \frac{(z-1)^\alpha}{(z-1)^\alpha + K} \tag{4.18}$$

Where:

α and K represent respectively the order of astatism (also known as the class and the type of the feedback control system) and open loop static gain of the system.

Using (4.18), (4.17) can be rewritten as:

$$e_{ss} = \lim_{z \to 1} \left(\frac{z-1}{z}\right) \frac{(z-1)^{\alpha}}{(z-1)^{\alpha} + K} R(z) \tag{4.19}$$

From (4.19), it is obvious that the steady state error and hence the accuracy of the control system depends on the following parameters:

- The reference (Setpoint) signal, $R(z)$.
- The order of Astatism (or the type) of the control system, α.
- The open loop gain, K.

All these three parameters can be used to analyze and enhance the accuracy performance of the digital control system as it will be mentioned in the subsequent subsections.

3.2. Steady state accuracy due to input reference signal

The accuracy of the discrete feedback control system depends on the type of the input reference signal. As a result, it can be studied and analyzed according to three standard and basic reference signals as follows:

Table 4.3 different types of steady state errors corresponding to the types of input reference signals

Type of input reference signal		Generated Steady State Error
Step	$R(z) = \frac{z}{z-1}$	Position steady state error $(e_p(\infty))$
Ramp	$R(z) = \frac{zT_s}{(z-1)^2}$	Velocity steady state error $(e_v(\infty))$
Parabola	$R(z) = \frac{T^2 z(z+1)}{(z-1)^3}$	Acceleration steady state error $(e_a(\infty))$

With: T_s is the sampling time period (second).

The following table (**Table 4.4**) summarizes the values of the steady state errors and consequently the accuracy performance study of the discrete time control system regarding the three parameters; namely the type of the input reference signal as well as the type and the open loop static gain of the control system.

Table 4.4 Values of steady state error depending on control system's order of Astatism, input reference signal and open loop gain

		Order of Astatism of the Control System			
		$\alpha = 0$	$\alpha = 1$	$\alpha = 2$	$\alpha > 0$
Steady State Errors (e_{ss})	$e_p(\infty)$	$\frac{1}{1+K}$	0	0	0
	$e_v(\infty)$	∞	$\frac{T_s}{K}$	0	0
	$e_a(\infty)$	∞	∞	$\frac{{T_s}^2}{K}$	0

Chapter 5: Discrete Linear time invariant system controller design

1. Introduction

In the previous chapters we have discussed in details mainly the representation and modeling of discrete (sampled data) control systems, where the input-output or discrete transfer function representation is considered as the most important method to describe the behavior of such systems. This is because the knowledge of the transfer function is fundamental and basic for designing the appropriate controller which ensures the operation of the whole feedback control system with the required and desired performance.

This chapter will be devoted to discuss the discrete controller for the sampled data control system and we will be particularly interesting with the design of discrete proportional-integral-derivative (shortly named as PID) controller (regulator). We will focus on this type of controllers due to the fact that it is simple in structure and operation, familiar and extensively used everywhere.

2. Continuous time PID controller

Before tackling the subject of designing the discrete PID controller for a sampled data system, it is convenient to make brief review on continuous time PID controller.

The continuous time PID controller can be defined as an algorithm used to control or regulate a given physical quantity (variable) for a given process or system. It is the type of controller which is widely used in industrial processes, engineering and other fields. PID controller, also called three terms controller, is implemented using three actions:

- Proportional action (P).
- Integral action (I).
- Derivative action (D).

Using these three actions of control, different types and structures of control can be built such as: P, PI, PD and PID controllers.

In the subsequent sections, we consider the general input-output block diagram representation at the controller level as it is depicted in **Fig.5.1**.

Fig.5.1 General block diagram of the controller input-output relation

With $e(t)$ and $u(t)$ are respectively the tracking error and the control signals.
We recall from the previous chapters that in time domain, the tracking error signal is defined in a unity feedback control system as the difference between the reference signal $r(t)$ and the actual output signal $y(t)$. Mathematically, this is defined by:

$$e(t) = r(t) - y(t) \tag{5.1}$$

2.1 Proportional (P) controller

The control law generated using the proportional (P) controller is defined by the following relation:

$$u(t) = K_p e(t) \tag{5.2}$$

With K_p is called the proportional constant (parameter) which is specific designation to the proportional-type controller.
In frequency domain, the transfer function representing P-type controller can be obtained by applying Laplace transform on both sides of the relation (5.2). The result is mentioned by:

$$G_P(s) = \frac{U(s)}{E(s)} = K_p \tag{5.3}$$

With :
$C_P(s)$: is the P-type controller transfer function.
$E(s)$ and $U(s)$ are respectively Laplace transform of the input and output of the controller. We can also represent the P-controller by the following general block diagram depicted in **Fig.5.2**.

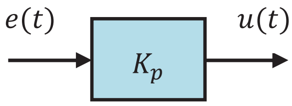


Fig.5.2 General block diagram representation of P-type controller

2.2 Proportional-Integral (PI) controller

The proportional-integral, which is abbreviated by PI and known as PI-type controller is a control algorithm based on two simultaneous actions; proportional action and integral action. Using this controller, the control signal is generated according to the following law (relation):

$$u(t) = K_p e(t) + K_I \int_0^t e(\tau)\, d\tau \qquad (5.4)$$

With:

K_p: is the proportional action constant (coefficient).

K_I: is the integral action constant (coefficient).

In some times (depending on the application), the PI control action is governed by:

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(\tau)\, d\tau \right] \qquad (5.5)$$

With:

T_i: is called the integral time constant and is related to the integral constant by :

$$K_i = \frac{K_p}{T_i} \qquad (5.6)$$

We can obtain the transfer function of PI controller by simply applying Laplace transform on both sides of either (5.4) or (5.5). In other words, we can have two transfer function configurations.

Using (5.4), the corresponding PI controller transfer function is given to be:

$$G_{PI}(s) = \frac{U(s)}{E(s)} = K_p + \frac{K_i}{s} \tag{5.7}$$

Regarding the relation (5.6), another form of PI controller transfer function can also be obtained and given by:

$$G_{PI}(s) = \frac{U(s)}{E(s)} = K_p . \frac{1 + T_i s}{T_i s} \tag{5.8}$$

Accordingly, two block diagram representations can be given to the PI controller respective to the couple of the above relations. These are given respectively in **Fig.5.3 (a)** and **Fig.5.3 (b)**.

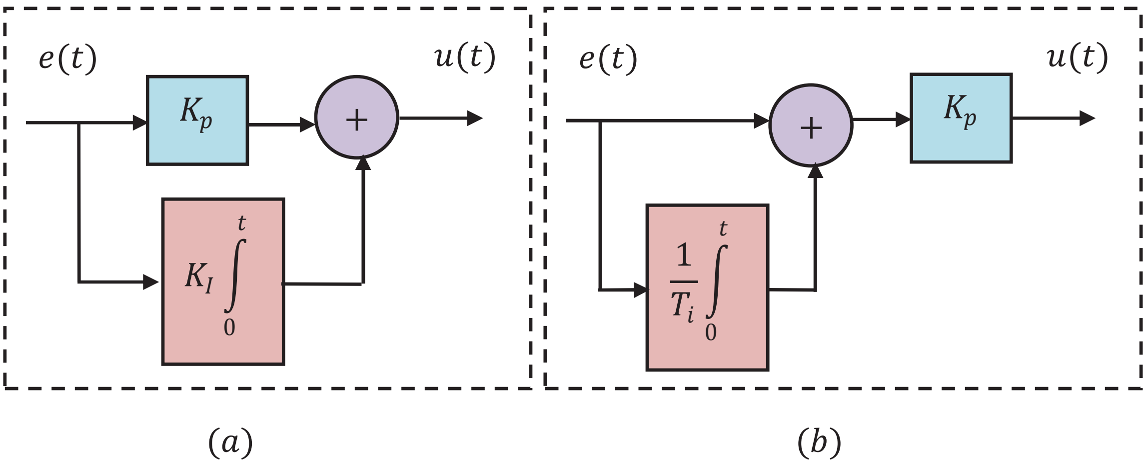


Fig.5.3 PI controller block diagram two possible representations

2.3 Proportional-Derivative (PD) controller

Among the possible combination between the three control actions P, I and D is that of combining the proportional action (P) with the derivative action (D) to form the Proportional-Derivative (PD) controller. For this controller type, the control law (signal) is generated using the following relationship between the tracking error, $e(t)$, as input to the controller and the control (actuating) signal, $u(t)$, as the output of the controller:

$$u(t) = K_p e(t) + K_d \frac{d[e(t)]}{dt} \tag{5.9}$$

with:

K_p and K_d are respectively the proportional action constant and the derivative action constant.

The expression (5.9) can also be rewritten under the following form:

$$u(t) = K_p \left[e(t) + T_d \frac{de(t)}{dt} \right] \tag{5.10}$$

With :

T_d : is called the derivative time constant, which is defined according to the following relation:

$$T_d = \frac{K_d}{K_p} \tag{5.11}$$

Using transfer function representation, PD controller can be input-output modeled by the following respective transfer functions:

$$G_{PD}(s) = \frac{U(s)}{E(s)} = \left(K_p + K_d s \right) \tag{5.12}$$

And:

$$G_{PD}(s) = \frac{U(s)}{E(s)} = K_p(1 + T_d s) \tag{5.13}$$

With :

$$T_d = \frac{K_d}{K_p} \tag{5.14}$$

Is called the derivative time constant.

When using block diagram representation, PD controller operation is also illustrated as shown in **Fig.5.4.**

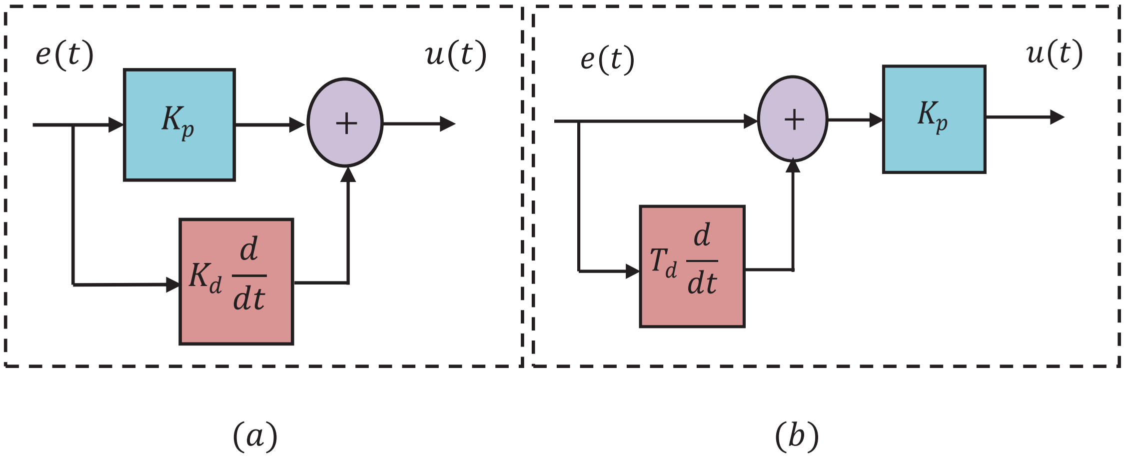


Fig.5.4 PD controller block diagram two possible representations

2.4 Proportional-Integral-Derivative (PID) controller

Proportional-integral-Derivative, abbreviated by PID, controller is the control algorithm that involves the contribution of all three control actions, P, I and D. as for the previous types, PID controller can be implemented using several and different structures and architectures. In this section, we shall briefly discuss three of them, namely:

- Series PID configuration.
- Parallel PID configuration.
- Mixed PID configuration.

2.4.1 Series PID controller configuration

Series PID controller configuration is the structure of the controller where the proportional, integral and derivative actions are connected in series as it is mentioned and represented by the following block diagram of **Fig.5.5**.

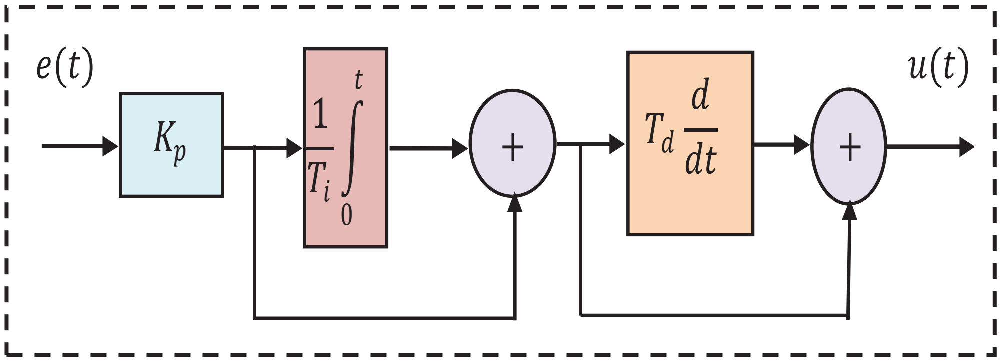


Fig.5.5 block diagram representation for series configuration of PID controller

The time domain relationship between the tracking error as input and the actuating (control) signal as the output of the series configuration of PID controller is given as follows:

$$u(t) = K_p e(t) + K_I \int_0^t e(\tau)\, d\tau + K_d \frac{de(t)}{dt}$$

$$= K_p \left[\frac{T_i + T_d}{T_i} e(t) + \frac{1}{T_i}\int_0^t e(\tau)\, d\tau + T_d \frac{de(t)}{dt}\right] \qquad (5.15)$$

By applying Laplace transform on both sides of (5.14), the frequency domain representation describing the behavior of the PID controller under its series structure implementation is given by the following continuous time transfer function:

$$G_{PID_s}(s) = \frac{U(s)}{E(s)} = K_p\left(1 + \frac{1}{T_i s}\right)(1 + T_d s) \qquad (5.16)$$

2.4.2 Parallel PID controller configuration

The PID controller can also be implemented using a parallel structure where the three actions Proportional, integral and derivative of the controller are connected in parallel structure as it is depicted in **Fig.5.6**.

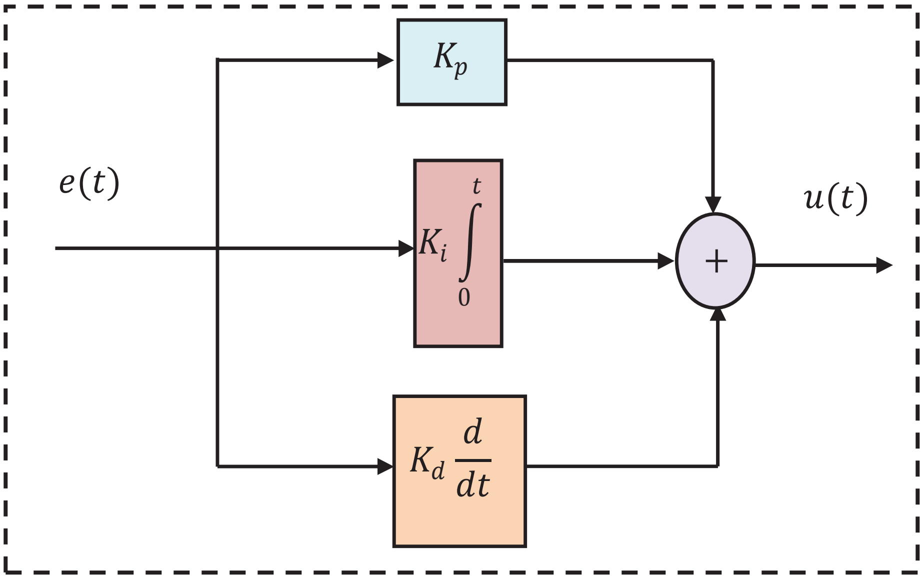


Fig.5.6 block diagram representation for parallel configuration of PID controller

The control law that governs the operation of parallel configuration of PID controller is expressed in the time domain by the following relationship between the input and the output of the controller:

$$u(t) = K_p e(t) + K_I \int_0^t e(\tau)\, d\tau + K_d \frac{de(t)}{dt} \tag{5.17}$$

From (5.16) and using Laplace transform, the corresponding transfer function representing and describing the behavior of the controller is obtained and given by:

$$G_{PID_p}(s) = \frac{U(s)}{E(s)} = K_p + K_i \frac{1}{s} + K_d s \tag{5.18}$$

Obviously, the parallel structure of PID controller is simpler and direct compared with the series configuration. Therefore, as long as the implementation of the controller is considered, the parallel structure is preferred in most of the industrial cases [7].

2.4.3 Mixed PID controller configuration

The mixed structure used to implement PID controller combines between both series and parallel connection of the proportional, integral and derivative control actions. A typical block diagram of this PID structure is shown in **Fig.5.7**.

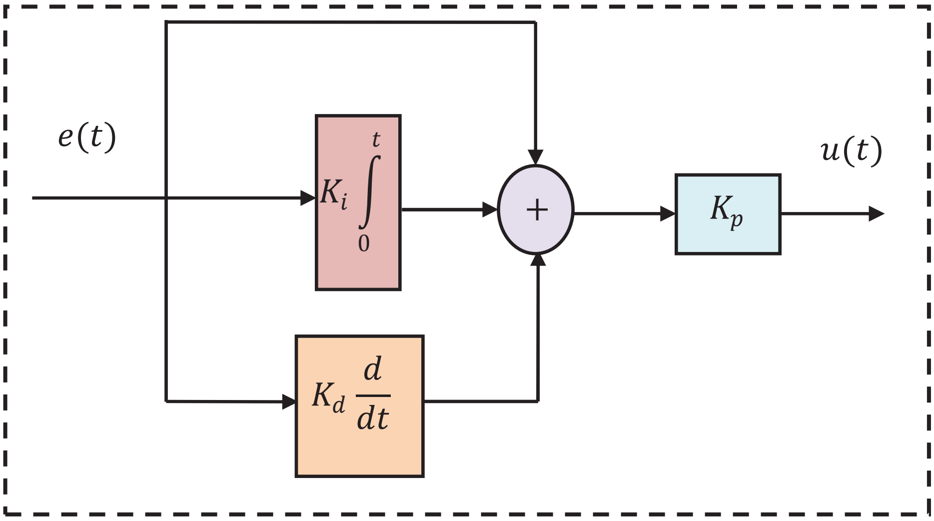


Fig.5.7 block diagram representation for mixed configuration of PID controller

The input-output law that describes the generation of the control signal by the mixed configuration of PID controller is expressed in the time domain by:

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(\tau)\, d\tau + T_d \frac{de(t)}{dt} \right] \tag{5.19}$$

In the frequency domain, the continuous time transfer function which is representing the behavior of mixed structure PID controller is obtained by using Laplace transform mathematical tool to be written as:

$$G_{PID_{mix}}(s) = \frac{U(s)}{E(s)} = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \tag{5.20}$$

3. Discrete PID controller design

Among the direct and straightforward method of designing a discrete (digital) PID controller is to discretize the already implemented continuous time PID controller by using one the discretization methods discussed previously in chapter 3. The general mechanism can be explained and illustrated by the following diagram of **Fig.5.8**.

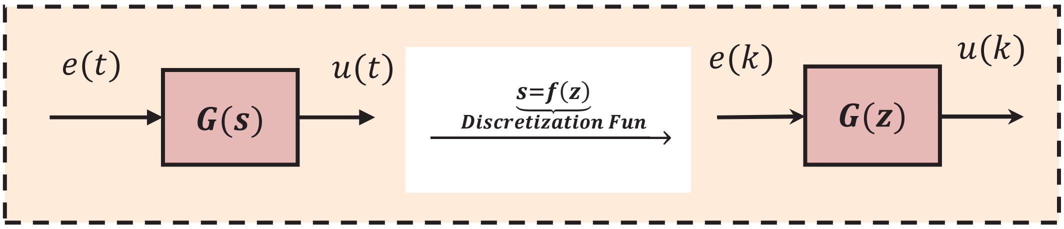


Fig.5.8 Block diagram representation showing discretization process

Where:

$G(s)$ and $G(z)$ represent respectively the continuous time and discrete transfer function of the corresponding controllers.

$s = f(z)$: is the used discretizing function, which corresponds to one of the discretization methods.

3.1 Discretization methods of continuous time derivative and integral

The basic principle employed to design the discrete (digital) PID controller by discretizing a continuous time controller consists of approximating the integral and derivative terms. These approximations consist, in fact, of converting the analog integral and derivative actions into discrete functions with respect to a predefined sampling period T_s.

In this section, we shall focus on using the more useful approximation (discretization) methods to obtain the discrete versions of the analog (I) and (D). Namely, this concern:

- Forward Euler's approximation method.
- Backward Euler's approximation method.
- Tustin approximation method.

In the following table (**Table 5.1**), we summarize the use of these approximation methods to obtain the equivalent z (discrete) transfer functions of integral (I) and derivative (D) parts of the PID controller.

Table 5.1 Discretization of integral (I) and derivative (D) of PID controller [8]

	Integral term (I)		**Derivative term (D)**	
	Continuous	**discrete**	**Continuous**	**discrete**
Forward Euler's approximation method	$\frac{1}{s}$	$\frac{z}{(z-1)}$	s	$\frac{(z-1)}{z}$
Backward Euler's approximation method	$\frac{1}{s}$	$\frac{T_s z}{(z-1)}$	s	$\frac{(z-1)}{T_s z}$
Tustin approximation method	$\frac{1}{s}$	$\frac{T_s}{2}\frac{(z+1)}{(z-1)}$	s	$\frac{2}{T_s}\frac{(z-1)}{(z+1)}$
With : T_s represents the used sampling time period				

4. Discrete (Digital) PID controller

Now it is easier to design and implement any variant of the different discrete PID controller types by just choosing and applying one of the above discretization methods. In the subsequent sections, we will be interesting of using the approximation method of forward Euler.

4.1 Equivalent Digital Proportional (P) controller

The digital P-type controller is fully defined and described by its discrete transfer function. Since the behavior of continuous time P-type controller is represented by the transfer function:

$$G_P(s) = \frac{U(s)}{E(s)} = K_p \tag{5.21}$$

The equivalent z transfer function is obtained by taking Z transform of $G(s)$, therefore:

$$G_P(z) = Z\{G_P(s)\} = \frac{Z\{U(s)\}}{Z\{E(s)\}} = \frac{U(z)}{E(z)} = Z\{K_p\} = K_p \tag{5.22}$$

Obviously, analog and digital P-type controllers have the same transfer function as no approximation can be applied to discretize the proportional (P) term of the controller.

In time domain representation, the digital proportional controller algorithm is written according to the following relationship between the input and output samples:

$$u(k) = K_p e(k) \tag{5.23}$$

If the discrete transfer function of the controller is known, the digital P-type controller can also be represented by the following functional block diagram shown in **Fig.5.9**.

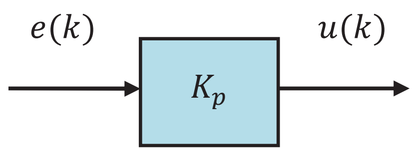


Fig.5.9 General block diagram representation of digital P-type controller

4.2 Equivalent Digital Proportional-Integral (PI) controller

If we consider that the continuous time PI controller is described by its transfer function mentioned in (5.8), which is rewritten for convenience as:

$$G_{PI}(s) = \frac{U(s)}{E(s)} = K_p.\frac{1 + T_i s}{T_i s} \tag{5.24}$$

By applying Euler's forward discretization formula, we can get:

$$G_{PI}(z) = G_{PI}(s)|_{s=\frac{(z-1)}{z}} = \frac{U(z)}{E(z)} = K_p.\left(1 + \frac{1}{T_i\left[\frac{(z-1)}{z}\right]}\right)$$

Finally, the equivalent z transfer function of digital PI-type controller is expressed as:

$$G_{PI}(z) = K_p.\left(1 + \frac{1}{T_i}\frac{z}{z-1}\right) \tag{5.25}$$

Accordingly, the control law (signal) generated by the controller at the time instant 'k' is expressed by the following time domain relationship.

$$u(k) = u(k-1) + K_p.\left(1 + \frac{1}{T_i}\right)e(k) - K_p e(k-1) \tag{5.26}$$

On the other hand, if the analog PI controller is represented by the second variant of transfer function given as :

$$G_{PI}(s) = \frac{U(s)}{E(s)} = K_p + \frac{K_i}{s} \tag{5.27}$$

In this case, when discretizing using forward Euler' formula, the digital PI controller is obtained to be represented by the z transfer function given by:

$$G_{PI}(z) = G_{PI}(s)\big|_{s=\frac{(z-1)}{z}} = K_p + K_i\left(\frac{z}{z-1}\right) \tag{5.28}$$

The corresponding two possible block diagram representations of the above z transfer functions of a typical digital PI controller are shown in **Fig.5.10**.

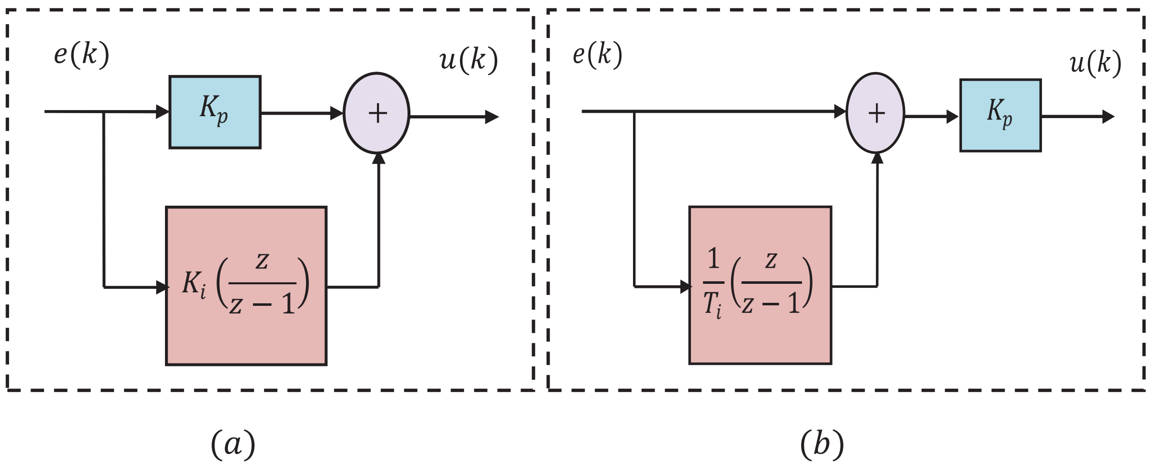


Fig.5.10 Digital PI controller block diagram two possible representations

4.3 Equivalent Digital Proportional-Derivative (PD) controller

We have described earlier the continuous time PD controller by two forms of transfer function which we rewrite here for convenience as:

$$G_{PD}(s) = \frac{U(s)}{E(s)} = (K_p + K_d s) \tag{5.29}$$

And:

$$G_{PD}(s) = \frac{U(s)}{E(s)} = K_p(1 + T_d s) \tag{5.30}$$

By discretizing using forward Euler's formula, the corresponding equivalent two possible discrete transfer function of a typical digital PD controller can be obtained to be:

$$G_{PD}(z) = \frac{U(z)}{E(z)} = G_{PD}(s)\Big|_{s=\frac{(z-1)}{z}} = \left(K_p + K_d\frac{(z-1)}{z}\right) \quad (5.31)$$

And:

$$G_{PD}(z) = \frac{U(z)}{E(z)} = G_{PD}(s)\Big|_{s=\frac{(z-1)}{z}} = K_p\left(1 + T_d\frac{(z-1)}{z}\right) \quad (5.32)$$

We can express the control law used to generate the control signal of the above two structures of the digital PD controller by just applying inverse z transform on the respective transfer functions (5.31) and (5.32). The result will be respectively as below:

$$u(k) = (K_p + K_d)e(k) - K_d e(k-1) \quad (5.33)$$

And :

$$u(k) = (K_p + K_pT_d)e(k) - K_pT_d e(k-1) \quad (5.34)$$

Using block diagram representation, these two configurations of digital PD controller can represented as it is depicted in **Fig.5.11.**

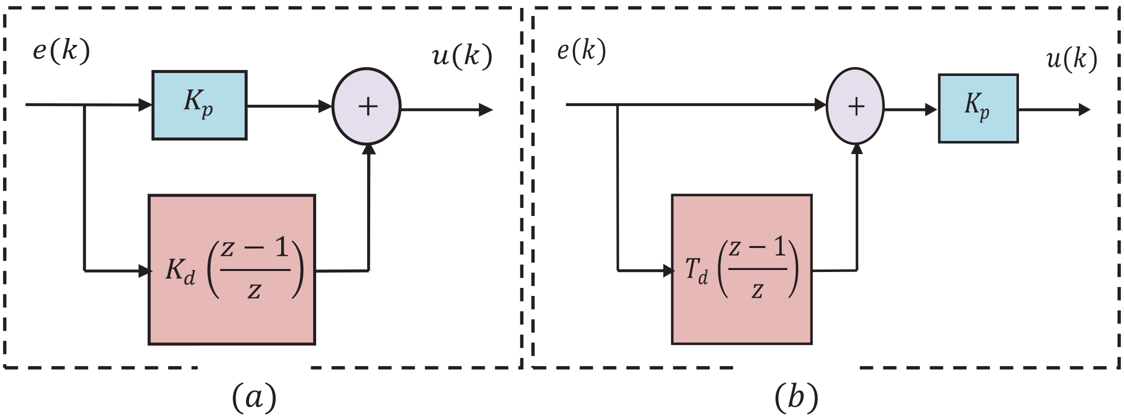


Fig.5.11 Digital PD controller block diagram two possible representations

4.4 Equivalent Digital Proportional-Integral-Derivative (PID) controller

As it was previously discussed in section 2.4, continuous time PID controller can be implemented using three different structures. Similarly, equivalent digital PID controller is also being designed under the same of these configurations; namely:

- Series digital PID configuration.
- Parallel digital PID configuration.
- Mixed digital PID configuration.

In the following subsections, we shall interest to deriving the different time domain, frequency domain and block diagram representations corresponding to each configuration.

4.4.1 Equivalent Series configuration of digital PID controller

For convenience, we rewrite the transfer function of the series structure of the analog PID controller as:

$$G_{PID_s}(s) = \frac{U(s)}{E(s)} = K_p\left(1 + \frac{1}{T_i s}\right)(1 + T_d s) \qquad (5.35)$$

Applying forward Euler's discretization formula, the discrete series configuration PID controller transfer function is obtained as:

$$G_{PID_s}(z) = \frac{U(z)}{E(z)} = G_{PID_s}(s)\Big|_{s=\frac{(z-1)}{z}}$$

$$G_{PID_s}(z) = K_p\left[1 + \frac{1}{T_i\frac{(z-1)}{z}}\right]\left[1 + T_d\frac{(z-1)}{z}\right] \qquad (5.36)$$

As long as the control law is concerned, the application of inverse z transform tool allows us to express the control signal generated at the discrete time instant 'k' by the following relationship between the input and the output samples of the controller.

$$u(k) = u(k-1) + C_0 e(k) + C_1 e(k-1) + C_2 e(k-2) \qquad (5.37)$$

Where :

C_0, C_1, C_2: are known constant coefficients and are respectively defined by :

$$\begin{cases} C_0 = \dfrac{K_p}{T_i}[(T_i + T_d) + 1 + T_i T_d] \\ C_1 = -\dfrac{K_p}{T_i}[(T_i + T_d) + 2T_i T_d] \\ C_2 = K_p T_d \end{cases}$$

We can also give the block diagram representation of this digital PID structure by direct implementation of the respective z transfer function expression (5.36). this is shown in **Fig.5.12**.

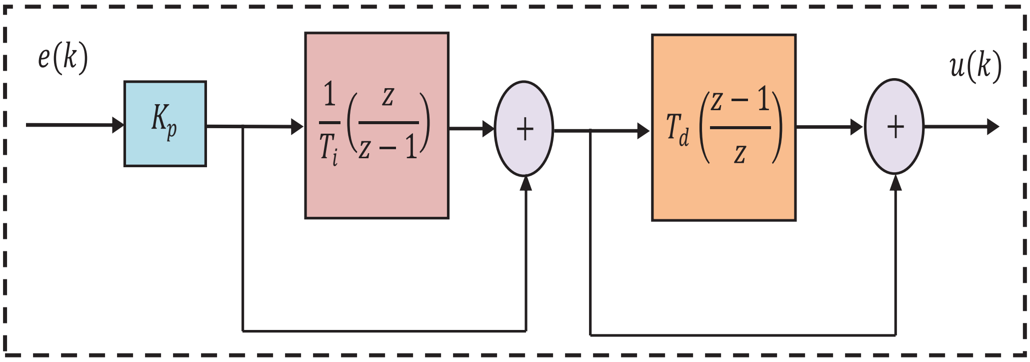


Fig.5.12 Block diagram representation for series configuration digital PID controller

4.4.2 Equivalent parallel configuration of digital PID controller

By referring to transfer function expression derived for parallel structure analog PID controller in section 2.4.2, and using the discretization formula of forward Euler's method, the equivalent z transfer function of this PID structure can be obtained as follows:

$$G_{PID_p}(z) = \frac{U(z)}{E(z)} = G_{PID_p}(s)\Big|_{s=\frac{(z-1)}{z}}$$

$$G_{PID_p}(z) = K_p + K_i\left(\frac{z}{z-1}\right) + K_d\left(\frac{z-1}{z}\right) \quad (5.38)$$

The corresponding control law responsible of generating the control signal can be derived from the expression of the discrete transfer function (5.38) using the inverse z transform. After some algebraic manipulations, this control law is described by the following relationship.

$$u(k) = u(k-1) + [K_p + K_i + K_d]e(k)$$
$$-[K_p + 2K_d]e(k-1) + K_d e(k-2) \qquad (5.39)$$

The control law described by (5.39) is also called the difference equation representation of the parallel structure digital PID controller.

The block diagram representing the operation and behavior of this PID controller structure can be drawn from the controller's z transfer function (5.38) and it is shown in **Fig.5.13**.

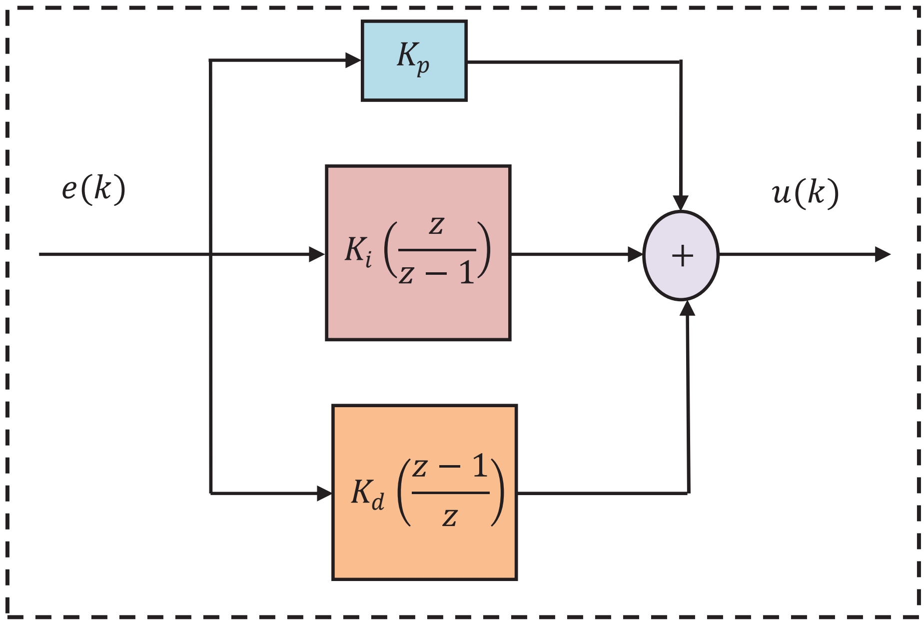


Fig.5.13 Block diagram representation for parallel configuration digital PID controller

4.4.3 Equivalent mixed configuration of digital PID controller

The mixed structure of digital PID controller combines the series and parallel structures exactly as it is done for the case of analog PID controller. Consequently, the equivalent discrete transfer function representing the implementation of this controller structure is directly obtained by discretization. Using the forward Euler's formula of discretization, the equivalent z transfer function of mixed structure digital PID controller is given as:

$$G_{PID_{mix}}(z) = \frac{U(z)}{E(z)} = G_{PID_{mix}}(s)\Big|_{s=\frac{(z-1)}{z}}$$

$$G_{PID_{mix}}(z) = K_p\left[1 + \frac{1}{T_i}\left(\frac{z}{z-1}\right) + T_d\left(\frac{z-1}{z}\right)\right] \quad (5.40)$$

Using inverse z transform applied on transfer function described by (5.40), the corresponding difference equation representing the controller in the time domain can be obtained and written as:

$$u(k) = u(k-1) + C_0 e(k) + C_1 e(k-1) + C_2 e(k-2) \quad (5.41)$$

Where C_0, C_1, C_2: are known constant coefficients and are respectively defined by :

$$\begin{cases} C_0 = K_p\left(1 + \frac{1}{T_i} + T_d\right) \\ C_1 = -K_p(1 + 2T_d) \\ C_2 = K_p T_d \end{cases}$$

5. Effect PID controller constants on the control system's performance

We end up this discussion about the design of the equivalent discrete (digital) PID controller by stating how the performance of the discrete (sampled data) feedback control system is affected by the choice of the parameters (constants), K_p, K_i and K_d, which characterize respectively the proportional, integral and derivative actions of PID controller. The effect of these constants is individually summarized in the following table (**Table 5.2**).

Table 5.2 Effect of Changing Independently PID Parameters on System Response

Closed Loop Response	***Rise Time***	***Overshoot***	***Settling Time***	***Steady State Error***	***Stability***
Increasing K_P	decrease	Increase	Small Increase	Decrease	Degrade
Increasing K_I	Small Decrease	Increase	Increase	Large Decrease	Degrade
Increasing K_D	Small Decrease	decrease	Decrease	Minor Change	Improve

References

1. John J. D'Azzo, Constantine H. Houpis, Stuart N. Sheldon, « ***Linear control System analysis and design with MATLAB***", Fifth Edition, Marcel Dekker, Inc., NewYork, U.S.A., 2003.

2. Charles L. Phillips, H. Troy Nagle, "***Digital Control System Analysis and Design***", 3rd edition, Prentice Hall, Englewood Cliffs, New Jersey, 1995.

3. Rao V. Dukkipati "***Analysis and Design of Control Systems using MATLAB***", New Age International (P) Ltd., Publishers, New Delhi, 2006.

4. M. Gopal, "***Digital Control and State Variable Methods- Conventional and Neuro-Fuzzy Control Systems***", 2nd edition, Tata McGraw Hill Publishing Company Limited, New Delhi, 2006.

5. Gregory P. Starr, « ***Introduction to Applied Digital Control*** », Second Edition, Department of Mechanical Engineering, The University of New Mexico, November 2006.

6. M. Sami Fadali, Antonio Visioli, « ***Digital Control Engineering, Analysis and Design***", Second Edition, Elsevier Inc., U.S.A., 2013.

7. David I. Wilson, **« *Advanced Control using* *MATLAB: or Stabilizing the Unstabilisable* »,** Auckland University of Technology, New Zealand, July 27, 2016.

8. Katsuhiko Ogata, "***Discrete time control systems***", 2nd Edition, Prentice Hall, Englewood Cliffs, New Jersey, 1995.