Simulating the Effect of the Sampling Time in Different Discrete Transformations for the First Order System Using MATLAB Software

AHMED S. ELGAREWI  
Operation Management  
Waha Oil Company /Libya  
ahmed_algrewi@hotmail.com

MAHMOUD M. EL -FANDI  
Electrical and Electronic Dept.  
University of Tripoli/Libya  
m_elfandi@hotmail.com

MUFTAH E. ALMOHER  
Operation Management  
Waha Oil Company /Libya  
muftah_almher@yahoo.ca

ABSTRACT

The basic model type of continuous-time dynamic systems is the differential equation. Analogously, the basis model type of discrete-time dynamic systems is the difference equation. Numerical analysis has great importance in the field of Engineering, Science and Technology etc. In numerical analysis, we get the result in numerical form by computing methods of given data. The base of numerical analysis is calculus of finite difference which deals with the changes in the dependent variable due to changes in the independent variable. This paper studies mathematically with examples different sampling times in different transformations (Difference equations ,Direct z-transform and Zero Order Hold transform ).Also in this paper, continuous and these discrete transformations were compared and simulated using MATLAB package. Linear Time Invariant systems LTI with backward, forward, and bilinear approximation methods, direct z-transform and Zero Order Hold were simulated to study the effect of different sampling times in open loop.

KEYWORDS


1- INTRODUCTION

A mathematical description of the input-output relation of a system can be formulated either in the time domain ,in the frequency domain or in z domain , and since the z-transform, describe a system in terms of its response to the individual frequency constituents of the input signal. [1]Another important term in discrete is the sampling time, the sampling for discrete systems should be take under consideration .One of the main principles for sampling is Nyquist Sampling Theorem.

The Nyquist Sampling Theorem : If a time-varying signal is periodically sampled at a rate of at least twice the frequency of the highest-frequency sinusoidal component contained within the signal, then the original time-varying signal can be exactly recovered from the periodic samples .Thus, if we are to have any chance of not losing information, we must restrict ourselves to functions that are zero in the frequency domain except in a window of width less than the sampling frequency, centered at the origin. The lowest frequency that cannot be in the data so as to avoid aliasing—one-half of the sampling frequency—is called the Nyquist frequency[2]

The Numerical approximation of differential equations (Difference equation) is one of the methods to approximate an analogue control filter and system with a digital and to convert the analogue system or controller’s transfer function to differential equations and then obtain a numerical equation. This approach is approximation of numerical differential equations and called finite-difference approximation of derivatives [3] [4]Another method for discretization is Z.O.H , the Zero Order Hold the simplest extrapolation method introduced when is samples are held constant until the next sampling instant.
2- **EULER’S BACKWARD METHOD APPROXIMATION OF DIFFERENTIAL EQUATIONS**

The Euler method is the simplest algorithm for numerical solution of differential equations. The first order-derivative and second order-derivative approximated as:

- **First order derivative**

\[
\frac{dx}{dt} = \frac{X(k) - X(k - 1)}{T} \tag{1}
\]

Where
- \(X(k)\) is the value of the \(X(t)\) at the time “\(k\)"
- \(X(k-1)\) is the value of the \(X(t)\) at the time “\(k-1\)”
- \(T\) is an integration time

2.1 **Discrete Model of the First Order System by Using Backward Approximation**

i. **Difference Equation Transformation**

We consider the general form of the first order system of single input single output linear-time invariant [4] given in figure (1) below

\[
G(s) = \frac{y(s)}{u(s)} = \frac{K}{\tau s + 1} \tag{2}
\]

Where:
- \(K\) is the system gain,
- \(\tau\) is the system time constant.

From equation (3)

\[
\tau sy(s) + y(s) = Ku(s) \tag{3}
\]

By taking inverse Laplace we get

\[
\tau \frac{dy}{dt} + y(t) = Ku(t) \tag{4}
\]

By using backward equation (1), equation (4) becomes

\[
\tau \left( \frac{y(k) - y(k - 1)}{T} \right) + y(k) = Ku(k)
\]

\[
\tau (y(k) - y(k - 1)) + Ty(k) = KT u(k)
\]

\[
[\tau + T]y(k) = KT u(k) + \tau y(k - 1)
\]

\[
y(k) = \frac{\tau}{(\tau + T)}y(k - 1) + \frac{KT}{(\tau + T)} u(k) \tag{5}
\]

Where
- \(\alpha_1 = \frac{\tau}{(\tau + T)}\)
- \(\alpha_2 = \frac{KT}{(\tau + T)}\)

Equation (5) is a recursive of the first order system using backward approximation

ii. **Z-Transformation**

\[
y(k) = \frac{\tau}{(\tau + T)}y(k - 1) + \frac{KT}{(\tau + T)} u(k)
\]

\(y(k) \rightarrow Y(z); \quad y(k - 1) \rightarrow z^{-1}Y(z);\)

\(u(k) \rightarrow U(z)\)

From equation (5)

\[
\tau Y(z) - \tau z^{-1}Y(z) + TY(z) = KTU(z)
\]

\[
Y(z)[\tau - \tau z^{-1} + T] = KTU(z)
\]

\[
\frac{Y(z)}{U(z)} = \frac{KT}{\tau - \tau z^{-1} + T} \tag{7}
\]

\[
\frac{Y(z^{-1})}{U(z^{-1})} = \frac{KT/\tau}{(\tau + T) - z^{-1}} \tag{8}
\]

Where
\[ a = \frac{KT}{\tau}, \]
\[ b = \frac{\tau}{(\tau + T)} \]

**Example 1**

For the first order system giving by the following transfer function, the value of \( K = 12.8 \) and \( \tau = 16.7 \)

\[ \frac{y(s)}{u(s)} = \frac{12.8}{16.7s + 1} \tag{9} \]

\[ \alpha_1 = \frac{\tau}{(\tau + T)}, \quad \alpha_2 = \frac{KT}{(\tau + T)} \]

\[ y(k) = \frac{16.7}{(16.7 + T)}y(k-1) + \frac{12.8T}{(16.7 + T)}u(k) \tag{10} \]

Using equation (8) we can transfer equation (10)

where \( a = \frac{KT}{\tau} = 0.7664T \)

and \( b = \frac{(\tau + T)}{\tau} = 1.7664 \)

\[ \frac{Y(z^{-1})}{U(z^{-1})} = \frac{0.7664T}{1.7664 - z^{-1}} \tag{11} \]

**3- FORWARD METHOD APPROXIMATION OF DIFFERENTIAL EQUATIONS**

The first order- derivative and second order-derivative of the forward approximation [4][5] as

- **First order derivative**

\[ \frac{dx}{dt} = \frac{X(k+1) - X(k)}{T} \tag{12} \]

Where \( X(k) \) is the value of the \( X(t) \) at the time “k”

\( X(k+1) \) is the value of the \( X(t) \) at the time “k+1”

\( T \) is an integration time

**3.1- Discrete Model of the First Order System by Using Forward Approximation**

**i. Difference Equation Transformation**

We consider the general form of the first order system of equation (2) of single input single output linear-time invariant given in the figure (1)

\[ G(s) = \frac{y(s)}{u(s)} = \frac{K}{\tau s + 1} \]

\[ \tau s y(s) + y(s) = Ku(s) \]

By taking inverse Laplace for the above equation

\[ \frac{dy}{dt} + y(t) = Ku(t) \]

In forward approximation using equation (12)

\[ \frac{dy}{dt} = \frac{y(k+1) - y(k)}{T} \]

\[ \tau \left( \frac{y(k+1) - y(k)}{T} \right) + y(k) = Ku(k) \]

\[ \tau y(k+1) - \tau y(k) + Ty(k) = KT u(k) \]

\[ \tau y(k+1) = KT u(k) + [\tau - T]y(k) \]

\[ y(k+1) = \left[ 1 - \frac{T}{\tau} \right] y(k) + \frac{KT}{\tau} u(k) \tag{13} \]

Equation (13) is a recursive equation of the first order system by using forward approximation

**ii. Z-Transformation**

The Z. transform of model equation (13)

\[ y(k+1) = \left[ 1 - \frac{T}{\tau} \right] y(k) + \frac{KT}{\tau} u(k) \]

\[ y(k) \rightarrow Y(z), y(k+1) \rightarrow zY(z), \]

\[ u(k) \rightarrow U(z) \]
\[ \tau y(k + 1) - \tau y(k) + Ty(k) = KT u(k) \]
\[ \tau zY(z) - \tau Y(z) + TY(z) = KTU(z) \]
\[ Y(z)[\tau z - \tau + T] = KTU(z) \]

\[ Y(z) = \frac{KT}{z - \frac{(\tau + T)}{\tau}} \]  \hspace{1cm} (14)
\[ Y(z) = \frac{a}{z - b} \]  \hspace{1cm} (15)

Where

\[ a = \frac{KT}{\tau} \frac{(\tau + T)}{\tau} \]
\[ b = \frac{1}{T} \]

Example 2

For the first order system transfer function in example (1) find the forward approximation

\[ \frac{y(s)}{u(s)} = \frac{12.8}{16.7s + 1} \]
\[ y(k + 1) = \left[ 1 - \frac{T}{16.7} \right] y(k) + \frac{12.8T}{16.7} u(k) \]  \hspace{1cm} (15)

To z transform using equation (14)

\[ \frac{Y(z)}{U(z)} = \frac{0.7664T}{z - \frac{(16.7 + T)}{16.7}} \]  \hspace{1cm} (16)

4- BILINEAR APPROXIMATION METHOD OF DIFFERENTIAL EQUATIONS

This is also known as trapezoidal or center, approximation method Euler’s Forward and Backward approximation methods are called first order because they use one sample during each sampling interval, when more than one sample is used to update the approximation analogue model can be improved over that of the simpler approximation given by Euler’s forward or backward methods [6]

In central method the first derivative is approximated by the average of the first derivatives at half a time interval before and after time “k”

- First order derivative

\[ \frac{dx}{dt} = \frac{1}{2} \left[ X \left( k + \frac{1}{2} \right) + X \left( k - \frac{1}{2} \right) \right] \]
\[ \frac{dx}{dt} = \frac{1}{2} \left[ \frac{X(k + 1) - X(k)}{T} - \frac{X(k) - X(k - 1)}{T} \right] \]
\[ \frac{dx}{dt} = \frac{1}{2} \left( \frac{X(k + 1) - X(k) + X(k) - X(k - 1)}{T} \right) \]
\[ \frac{dx}{dt} = \frac{X(k + 1) - X(k - 1)}{2T} \]  \hspace{1cm} (17)

4.1- Discrete Model of The First Order System by Using Bilinear Approximation

i. Difference Equation Transformation

We consider the general form of the first order system in equation (2) of single input single output linear-time invariant given in the figure (.)

\[ G(s) = \frac{y(s)}{u(s)} = \frac{K}{\tau s + 1} \]
\[ \tau y(s) + y(s) = Ku(s) \]

By taking inverse Laplace

\[ \tau \frac{dy}{dt} + y(t) = Ku(t) \]

In Bilinear approximation using equation (17)

\[ \frac{dy}{dt} = \frac{y(k + 1) - y(k - 1)}{2T} \]
\[ \tau \left( \frac{y(k + 1) - y(k - 1)}{2T} \right) + y(k) = Ku(k) \]
\[ \tau (y(k + 1) - y(k - 1)) + 2Ty(k) = 2KTu(k) \]
Equation (18) is a recursive equation of the first order system using bilinear approximation

ii. **Z-Transformation**

The Z-transform model of equation (18)

\[
y(k + 1) = y(k - 1) - \frac{2T}{\tau}y(k) + \frac{2KT}{\tau}u(k)
\]

\[
y(k) \rightarrow Y(z), y(k + 1) \rightarrow zY(z)
\]

\[
, y(k - 1) \rightarrow z^{-1}u(k) \rightarrow U(z)
\]

\[
\tau(y(k + 1) - y(k - 1)) + 2Ty(k) = 2KTu(k)
\]

\[
tzY(z) - \tau z^{-1}Y(z) + 2TY(z) = 2KTU(z)
\]

\[
Y(z)[\tau z - \tau z^{-1} + 2T] = 2KTU(z)
\]

\[
\frac{Y(z)}{U(z)} = \frac{2KT}{\tau z - \tau z^{-1} + 2T}
\]

\[
\frac{Y(z)}{U(z)} = \frac{2KTz}{\tau z^2 - \tau + 2Tz}
\]

(19)

**Example 3**

For the first order system transfer function in example (1) of equation (9) find the bilinear approximation

\[
\frac{y(s)}{u(s)} = \frac{12.8}{16.7s + 1}
\]

Using equation (18) of the first order bilinear approximation we get

\[
y(k + 1) = y(k - 1) - \frac{2T}{16.7}y(k) + \frac{25.6T}{16.7}u(k)
\]

(20)

The z-transformation using equation (19)

\[
\frac{Y(z)}{U(z)} = \frac{25.6Tz}{16.7z^2 - 16.7 + 2Tz}
\]

(21)

**Table 1 Finite difference approximation of the first order system.**

<table>
<thead>
<tr>
<th>Continuous system</th>
</tr>
</thead>
<tbody>
<tr>
<td>( G(s) = \frac{k}{\tau s + 1} )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Backward approximation</th>
</tr>
</thead>
<tbody>
<tr>
<td>( y(k) = \frac{\tau}{(\tau + T)}y(k - 1) + \frac{KT}{(\tau + T)}u(k) )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Forward approximation</th>
</tr>
</thead>
<tbody>
<tr>
<td>( y(k + 1) = \left[1 - \frac{T}{\tau}\right]y(k) + \frac{KT}{\tau}u(k) )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bilinear approximation</th>
</tr>
</thead>
<tbody>
<tr>
<td>( y(k + 1) = y(k - 1) - \frac{2T}{\tau}y(k) + \frac{2KT}{\tau}u(k) )</td>
</tr>
</tbody>
</table>

Where:

- \( K \) is the system Gain
- \( \tau \) is the system time constant
- \( T \) is the integration time

**5- DIRECT Z-TRANSFORMATION METHOD**

The direct method is a very logical and straightforward process. \( y(t) \) is considered to be the inverse Laplace transforms of linear system with transfer function \( G(s) \). To determine \( G(z) \), we evaluate \( g(t) \) at the sampling time and take the Z-transform of this resultant sequence [7]:

That is

\[
H(z) = \sum_{k=0}^{\infty} h(kT)z^{-k}
\]

(22)

Where

\[
h(t) = l^{-1}[H(s)]
\]

(23)
Example 4
For the general transfer function of the first order system below
\[ G(s) = \frac{b}{s + a} \] (24)

Using equations (22), (23)
\[ g(t) = l^{-1}\left( \frac{b}{s + a} \right) = be^{-at} \]

So that
\[ h(kT) = be^{-akT} \quad \text{for} \quad k > 0 \]
\[ h(kT) = 0 \quad \text{for} \quad k < 0 \]

The z-transformation of this sequence is given by
\[ H(z) = \sum_{k=0}^{\infty} be^{-akT}z^{-k} \] (25)
\[ H(z) = b + be^{-T}z^{-1} + be^{-2T}z^{-2} \ldots \ldots \] (26)

5.1 - Discrete of the first order system by using Direct Z transform

The general form of the first order system with time constant \((\tau)\) and gain \((k)\) given in equation (2)
\[ G(s) = \frac{k}{\tau s + 1} \]

The z-transformation of this system by using direct method is given by
\[ G(z) = \frac{K}{1 - e^{-T}z^{-1}} = \frac{b_0}{1 - a_1z^{-1}} \] (27)

Where
\[ b_0 = \frac{k}{\tau} \]
\[ a_1 = e^{-T/\tau} \]

Example 5
For the first order system transfer function of equation (2.10), find the direct Z-transform
\[ G(s) = \frac{12.8}{16.7s + 1} \]
\[ k = 12.8, \quad \tau = 16.7 \]

First \(G(s)\) can be modified to be in pole-zero form:
\[ G(s) = \frac{0.7664}{s + 0.0598} \]

The value of \(b_0 = 0.7664\) and \(a_1 = e^{-0.0598T}\)

Using equation (27)
\[ G(z) = \frac{0.7664}{1 - e^{-0.0598T}z^{-1}} \]
\[ G(z) = \frac{0.7664z}{z - e^{-0.0598T}} \] (28)

6 - TRANSFORMATION WITH ZERO ORDER HOLD (Z.O.H)

The Zero Order Hold (ZOH) is a mathematical model of the practical signal reconstruction done by a conventional digital – to - analog converter (DAC). That is, it describes the effect of converting a discrete-time signal to a continuous-time signal by holding each sample value for one sample interval. It has several applications in digital control systems and digital communications [8].

The z-transform with Z.O.H of a linear system with transfer function \(G(s)\) is given by
\[ G_{Z.O.H}(z) = (1 - z^{-1})Z\left(\frac{G(s)}{s}\right) \] (29)

6.1 - Discrete model of the first order system using Z.O.H transformation

The general form of the first order system of equation (2) is given Transfer function
By rearranging the above formula we get

\[ G(s) = \frac{K}{\tau s + 1} \]

By rearranging the above formula we get

\[ G(s) = \frac{1}{\tau} \frac{K}{s + \frac{1}{\tau}} \]  \hspace{1cm} (30)

Using equation (30) to transfer equation (2) to Z.O.H

\[ G(z) = (1 - z^{-1})Z \left( \frac{1}{\tau} \right) \left( \frac{1}{s + \frac{1}{\tau}} \right) \]

\[ G(z) = K(1 - z^{-1})Z \left( \frac{1}{s(s + \frac{1}{\tau})} \right) \]  \hspace{1cm} (31)

For the first order system transfer function of equation (9), find the Zero Order Hold

\[ G(s) = \frac{12.8}{16.7s + 1} \]

The z-transform with Z.O.H of the system becomes

\[ G(z) = (1 - z^{-1})Z \left( \frac{12.816}{s + 0.0598} \right) \]

\[ G(z) = (12.816)(1 - z^{-1})Z \left( \frac{0.0598}{s(s + 0.0598)} \right) \]

Using equation (33)

\[ G(z) = \frac{(12.816)(1 - e^{-0.0598T})}{(z - e^{-0.0598T})} \]  \hspace{1cm} (36)

\[
\begin{array}{|c|c|}
\hline
\text{Continuous system} & \text{G(s)} = \frac{k}{\tau s + 1} \\
\hline
\text{Backward Transformation} & \text{G(z)} = \left( \frac{KT}{T + T} \right) \left( \frac{z}{z - \frac{T}{T} + T} \right) \\
\hline
\text{Forward Transformation} & \text{G(z)} = \left( \frac{KT}{T} \right) \left( \frac{z}{z + \frac{T}{T} - 1} \right) \\
\hline
\text{Bilinear Transformation} & \text{G(z)} = \left( \frac{2KT}{T} \right) \left( \frac{z}{z^2 + \left( \frac{2T}{T} \right)z - 1} \right) \\
\hline
\text{Direct Z Transformation} & \text{G(z)} = \left( \frac{KT}{T} \right) \left( \frac{z}{z - \frac{T}{T}} \right) \\
\hline
\text{Z.O.H Transformation} & \text{G(z)} = \left( \frac{K(1 - e^{-T})}{(z - e^{-T})} \right) \\
\hline
\end{array}
\]

Where

- T is the sampling time
- K is the system Gain
- \( \tau \) is the system time constant
### 7- SIMULATION OF 1ST ORDER SYSTEMS WITH DIFFERENT TRANSFORMATION AND SAMPLING TIME

Different systems were simulated as shown in figures (2-a) and (2-b) in open loop, in both continuous and discrete with step input. Using five methods of digital z-transformation, forward, backward and bilinear approximation methods, Direct z-transformation and Zero Order Hold transformation simulated with three sampling time to check the open loop response and the effect of each transformation. All transformations are simulated using MATLAB code.

![Figure 2-a: Blocks of continuous and digital systems in open loop](image)

![Figure 2-b: Blocks of continuous and digital systems in open loop](image)

For the first order SISO time invariant system in equation (37) and the digital z-transformation as shown in **TABLE 3** below.

**TABLE 3** All discrete transformations equations with different sampling time of the 1st system of equation (37)

<table>
<thead>
<tr>
<th>Z-Transform Method</th>
<th>Sampling Time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>G(z)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>G(s)</strong></td>
<td></td>
</tr>
<tr>
<td>0.001 sec</td>
<td></td>
</tr>
<tr>
<td>Forward approximation</td>
<td>( \frac{0.001}{z - 0.999} )</td>
</tr>
<tr>
<td>Backward approximation</td>
<td>( \frac{0.000999z}{z - 0.999} )</td>
</tr>
<tr>
<td>Bilinear approximation</td>
<td>( \frac{0.00049z + 0.00049}{z - 0.999} )</td>
</tr>
<tr>
<td>Direct method</td>
<td>( \frac{z}{z - 0.999} )</td>
</tr>
<tr>
<td>Zero Order Hold</td>
<td>( \frac{0.0009995}{z - 0.999} )</td>
</tr>
<tr>
<td>0.1 sec</td>
<td></td>
</tr>
<tr>
<td>Forward approximation</td>
<td>( \frac{0.1}{z - 0.9} )</td>
</tr>
<tr>
<td>Backward approximation</td>
<td>( \frac{0.0909z}{z - 0.909} )</td>
</tr>
<tr>
<td>Bilinear approximation</td>
<td>( \frac{0.0476z + 0.0476}{z - 0.9047} )</td>
</tr>
<tr>
<td>Direct method</td>
<td>( \frac{z}{z - 0.9048} )</td>
</tr>
<tr>
<td>Zero Order Hold</td>
<td>( \frac{0.095161}{z - 0.9048} )</td>
</tr>
<tr>
<td>0.5 sec</td>
<td></td>
</tr>
<tr>
<td>Forward approximation</td>
<td>( \frac{0.5}{z - 0.5} )</td>
</tr>
<tr>
<td>Backward approximation</td>
<td>( \frac{0.333z}{z - 0.666} )</td>
</tr>
<tr>
<td>Bilinear approximation</td>
<td>( \frac{0.2z + 0.2}{z - 0.6} )</td>
</tr>
<tr>
<td>Direct method</td>
<td>( \frac{z}{z - 0.6065} )</td>
</tr>
<tr>
<td>Zero Order Hold</td>
<td>( \frac{0.3935}{z - 0.6065} )</td>
</tr>
</tbody>
</table>
the results of the simulation using MATLAB program of equation:

\[ G(s) = \frac{1}{s + 1} \]  

(Figure 3)

The open loop response of first order system Eq. (37)

Sampling time 0.001 sec

(Figure 4)

System of equation (37) in continuous, Forward, Backward and Bilinear approximations and Zero Order hold transform with sampling time 0.001 sec

For Figure 4 the all responses of continuous, forward, backward, bilinear and ZOH are the same for the sampling time is 0.001 seconds, so the z-transform is totally the same as continuous signal.

Sampling time 0.1 sec

(Figure 6)

System of equation (37) in continuous, Forward, Backward and Bilinear approximations and Zero Order hold transform with sampling time 0.1 sec

For the sampling time 0.001 second the direct method z transformation has larger steady state error than the continuous system.
when sampling time is increased to 0.1 second all transformations responses are close to the response of continuous system except that the sampling time as shown, and for the direct method still has larger steady state error than the continues but smaller than the one with sampling time 0.001 second.

**Sampling time 0.1 sec**

With sampling time 0.5 seconds, the steps in the transient time are shown which considered as distortion in some applications, still all transformations responses in open loop are close to the continuous system step response and the direct z-transform response with smaller steady state error than 0.001 & 0.1 seconds

**8- CONCLUSION**

Simulations using software like MATLAB package gives good results before real time implementations. Transformations from continuous to discrete can be used in different methods, and choosing desired sampling time depends on the application. And some discrete transformations and sampling times change the characteristics of some continuous transfer functions and leads to instability. Discrete open loops systems mostly have the same transient response of continuous open loops systems in all transformations except direct method z-transform.

**9- MATLAB CODE**

%CODE A-1
% This Program is used to transform continues S-domain transfer function to different discrete forms , Forward, Backward and Bilinear approximations, Direct method z-transform and zero order hold.
%This program uses different colors for each transform
% 1- When using a forward, Backward or Bilinear MAKE SURE TO CHANGE LETTER “s” to LETTER “x” in the transfer function due to programming issues.
% 2- After completing all transformation make sure to close the figure to let the code open a new figure and work probably
clc
s=tf('s');
fprintf('This programme changes to Forward, Backward and Bilinear approximation. Direct method and Zero order Hold\n');
H1=input('Enter Your TF Function of S: \n')%Transfer Function needed to be Transferred
h=figure(1);
hold;
step(H1) %Step response of open-loop system in continuous
T=input('Enter the sampling Time: \n')% Sampling Time
for r=1:5 % TO choose one of five different discrete conversion method
    z=tf('z',T);
    O=input('Enter the Method of Approximation: \n1 Forward Method \n2-Backward Method \n3-Bilinear Method \n4- Direct z transform \n5-Zero Order Hold\n');
    if O==1
        x=((z-1)/(T));% Forward method form
        s=sym('x')
        H=input('Enter Your TF Function using x: \n')%Transfer Function needed to be Changed to Forward
        disp('The Forward Method TF above\n')
        step(H)
    elseif O==2
        x=((z-1)/(z*T));% Backward method form
        s=sym('x')
        H=input('Enter Your TF Function of x: \n')%Transfer Function needed to be Changed to Backward
        disp('The Backward Method TF above\n')
        step(H)
    elseif O==3
        x=[2*(z-1)]/([T*(z+1)]);% Bilinear method form
        s=sym('x')
        H=input('Enter Your TF Function of x: \n')%Transfer Function needed to be Changed to Bilinear
        disp('The Bilinear Method TF above\n')
        step(H)
    elseif O==4
        H=c2d(H1,T,'impulse')% Direct Method transformation
        disp('DirectZ transform Method TF above\n')
        step(H)
    elseif O==5
        H=c2d(H1,T,'zoh')% Zero Order Hold transform.
        disp('The Zero Order Hold Method TF above\n')
        step(H)
end
legend('Continuous', 'Forward', 'Backward', 'Bilinear', 'Direct', 'Z. O.H')%To use different colors and names for each transform
else
    button = questdlg('Ready to quit?', ...% switch button
    'Exit Dialog', 'Yes', 'No', 'No');
end