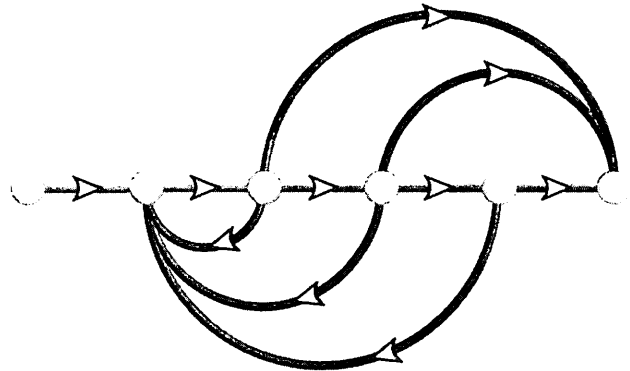


Digital Control Systems

13



Chapter Learning Outcomes

After completing this chapter the student will be able to:

- Model the digital computer in a feedback system (Sections 13.1–13.2)
- Find z - and inverse z -transforms of time and Laplace functions (Section 13.3)
- Find sampled-data transfer functions (Section 13.4)
- Reduce an interconnection of sampled-data transfer functions to a single sampled-data transfer function (Section 13.5)
- Determine whether a sampled-data system is stable and determine sampling rates for stability (Section 13.6)
- Design digital systems to meet steady-state error specification (Section 13.7)
- Design digital systems to meet transient response specifications using gain adjustment (Sections 13.8–13.9)
- Design cascade compensation for digital systems (Sections 13.10–13.11)

Case Study Learning Outcomes

You will be able to demonstrate your knowledge of the chapter objectives with a case study as follows:

- Given the analog antenna azimuth position control system shown on the front endpapers and in Figure 13.1(a), you will be able to convert the system to a digital system as shown in Figure 13.1(b) and then design the gain to meet a transient response specification.

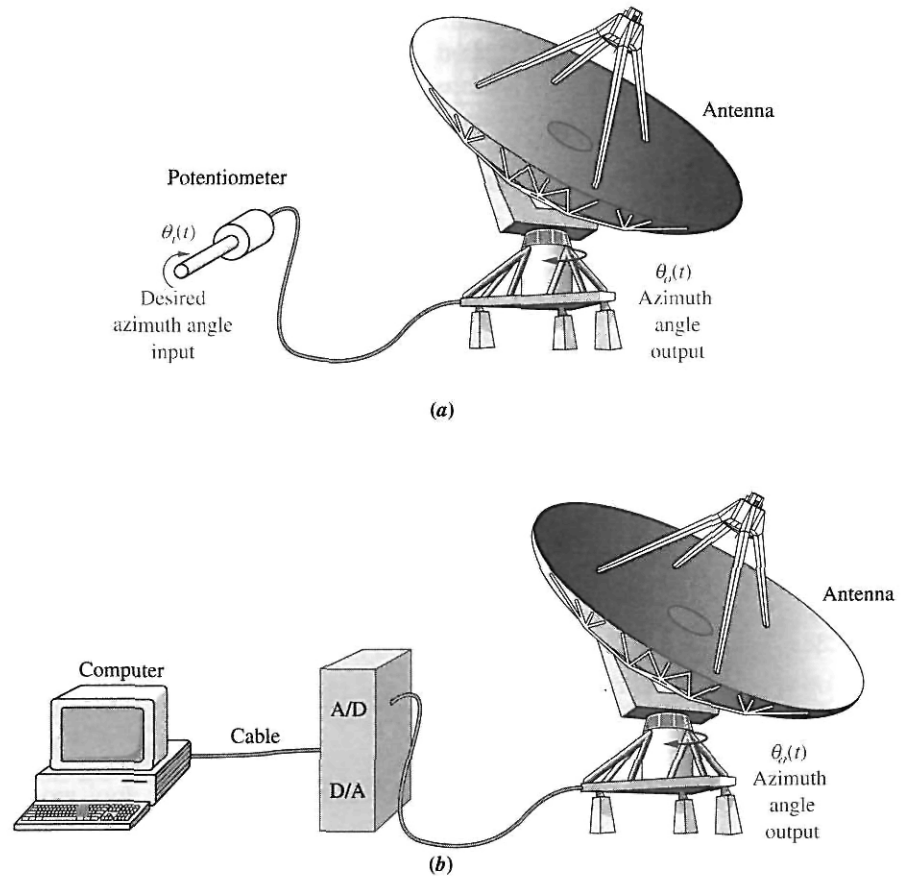


FIGURE 13.1 Conversion of antenna azimuth position control system from **a.** analog control to **b.** digital control

- Given the digital antenna azimuth position control system shown in Figure 13.1(b), you will be able to design a digital cascade compensator to improve the transient response.

13.1 Introduction

This chapter is an introduction to digital control systems and will cover only frequency-domain analysis and design. You are encouraged to pursue the study of state-space techniques in an advanced course in sampled-data control systems. In this chapter, we introduce analysis and design of stability, steady-state error, and transient response for computer-controlled systems.

With the development of the minicomputer in the mid-1960s and the microcomputer in the mid-1970s, physical systems need no longer be controlled by expensive mainframe computers. For example, milling operations that required mainframe computers in the past can now be controlled by a personal computer.

The digital computer can perform two functions: (1) supervisory—external to the feedback loop; and (2) control—internal to the feedback loop. Examples of

supervisory functions consist of scheduling tasks, monitoring parameters and variables for out-of-range values, or initiating safety shutdown. Control functions are of primary interest to us, since a computer that performs within the feedback loop replaces the methods of compensation heretofore discussed. Examples of control functions are lead and lag compensation.

Transfer functions, representing compensators built with analog components, are now replaced with a digital computer that performs calculations that emulate the physical compensator. What advantages are there to replacing analog components with a digital computer?

Advantages of Digital Computers

The use of digital computers in the loop yields the following advantages over analog systems: (1) reduced cost, (2) flexibility in response to design changes, and (3) noise immunity. Modern control systems require control of numerous loops at the same time—pressure, position, velocity, and tension, for example. In the steel industry, a single digital computer can replace numerous analog controllers with a subsequent reduction in cost. Where analog controllers implied numerous adjustments and resulting hardware, digital systems are now installed. Banks of equipment, meters, and knobs are replaced with computer terminals, where information about settings and performance is obtained through menus and screen displays. Digital computers in the loop can yield a degree of flexibility in response to changes in design. Any changes or modifications that are required in the future can be implemented with simple software changes rather than expensive hardware modifications. Finally, digital systems exhibit more noise immunity than analog systems by virtue of the methods of implementation.

Where then is the computer placed in the loop? Remember that the digital computer is controlling numerous loops; thus, its position in the loop depends upon the function it performs. Typically, the computer replaces the cascade compensator and is thus positioned at the place shown in Figure 13.2(a).

The signals r , e , f , and c shown in Figure 13.2(a) can take on two forms: digital or analog. Up to this point we have used analog signals exclusively. Digital signals, which consist of a sequence of binary numbers, can be found in loops containing digital computers.

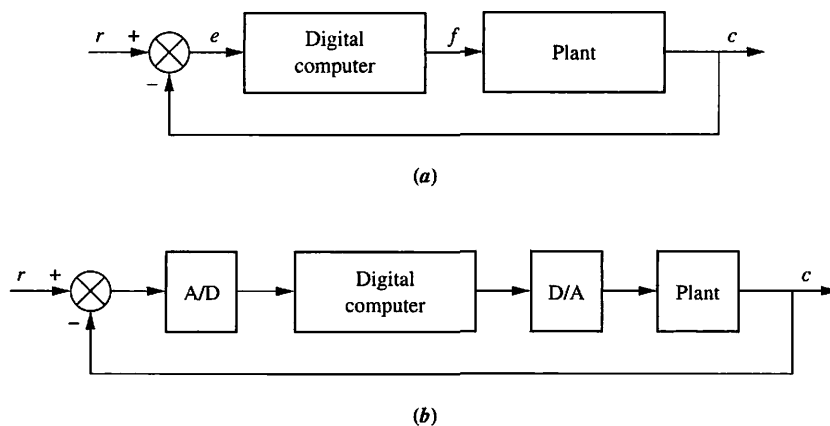


FIGURE 13.2 a. Placement of the digital computer within the loop; b. detailed block diagram showing placement of A/D and D/A converters

Loops containing both analog and digital signals must provide a means for conversion from one form to the other as required by each subsystem. A device that converts analog signals to digital signals is called an *analog-to-digital (A/D) converter*. Conversely, a device that converts digital signals to analog signals is called a *digital-to-analog (D/A) converter*. For example, in Figure 13.2(b), if the plant output, c , and the system input, r , are analog signals, then an analog-to-digital converter must be provided at the input to the digital computer. Also, if the plant input, f , is an analog signal, then a digital-to-analog converter must be provided at the output of the digital computer.

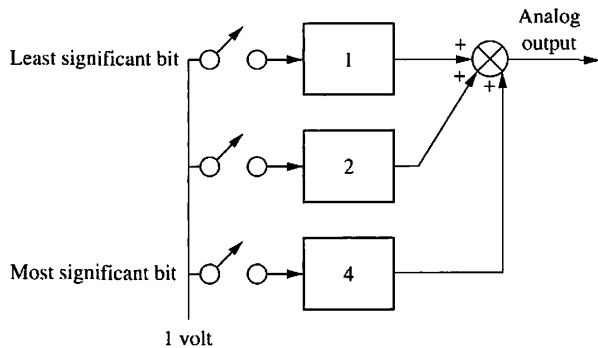


FIGURE 13.3 Digital-to-analog converter

Digital-to-Analog Conversion

Digital-to-analog conversion is simple and effectively instantaneous. Properly weighted voltages are summed together to yield the analog output. For example, in Figure 13.3, three weighted voltages are summed. The three-bit binary code is represented by the switches. Thus, if the binary number is 110_2 , the center and bottom switches are on, and the analog output is 6 volts. In actual use, the switches are electronic and are set by the input binary code.

Analog-to-Digital Conversion

Analog-to-digital conversion, on the other hand, is a two-step process and is not instantaneous. There is a delay between the input analog voltage and the output digital word. In an analog-to-digital converter, the analog signal is first converted to a sampled signal and then converted to a sequence of binary numbers, the digital signal.

The sampling rate must be at least twice the bandwidth of the signal, or else there will be distortion. This minimum sampling frequency is called the *Nyquist sampling rate*.¹

In Figure 13.4(a), we start with the analog signal. In Figure 13.4(b), we see the analog signal sampled at periodic intervals and held over the sampling interval by a device called a *zero-order sample-and-hold (z.o.h.)* that yields a staircase approximation to the analog signal. Higher-order holds, such as a first-order hold, generate more complex and more accurate waveshapes between samples. For example, a first-order hold generates a ramp between the samples. Samples are held before being digitized because the analog-to-digital converter converts the voltage to a digital number via a digital counter, which takes time to reach the correct digital number. Hence, the constant analog voltage must be present during the conversion process.

After sampling and holding, the analog-to-digital converter converts the sample to a digital number (as shown in Figure 13.4(c)), which is arrived at in the following manner. The dynamic range of the analog signal's voltage is divided into discrete levels, and each level is assigned a digital number. For example, in Figure 13.4(b), the analog signal is divided into eight levels. A three-bit digital number can represent each of the eight levels as shown in the figure. Thus, the difference between quantization levels is $M/8$ volts, where M is the maximum analog voltage. In general, for any system, this difference is $M/2^n$ volts, where n is the number of binary bits used for the analog-to-digital conversion.

Looking at Figure 13.4(b), we can see that there will be an associated error for each digitized analog value except the voltages at the boundaries such as $M/8$ and $2M/8$. We call this error the *quantization error*. Assuming that the quantization process rounds off the analog voltage to the next higher or lower level, the maximum

¹ See Ogata (1987: 170–177) for a detailed discussion.

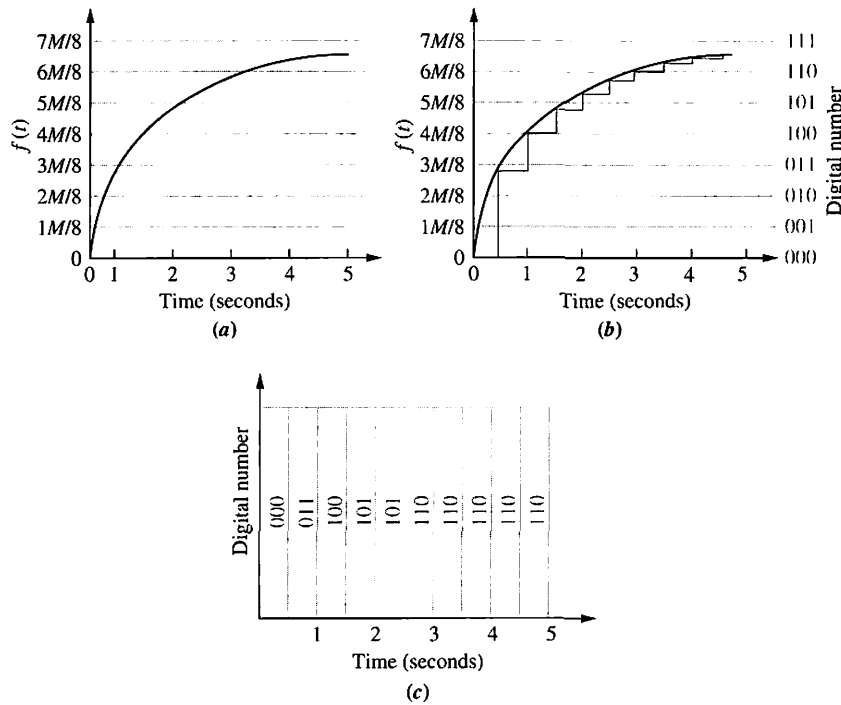


FIGURE 13.4 Steps in analog-to-digital conversion: **a.** analog signal; **b.** analog signal after sample-and-hold; **c.** conversion of samples to digital numbers

value of the quantization error is $1/2$ the difference between quantization levels in the range of analog voltages from 0 to $15M/16$. In general, for any system using roundoff, the quantization error will be $(1/2)(M/2^n) = M/2^{n+1}$.

We have now covered the basic concepts of digital systems. We found out why they are used, where the digital computer is placed in the loop, and how to convert between analog and digital signals. Since the computer can replace the compensator, we have to realize that the computer is working with a quantized amplitude representation of the analog signal formed from values of the analog signal at discrete intervals of time. Ignoring the quantization error, we see that the computer performs just as the compensator does, except that signals pass through the computer only at the sampled intervals of time. We will find that the sampling of data has an unusual effect upon the performance of a closed-loop feedback system, since stability and transient response are now dependent upon the sampling rate; if it is too slow, the system can be unstable since the values are not being updated rapidly enough. If we are to analyze and design feedback control systems with digital computers in the loop, we must be able to model the digital computer and associated digital-to-analog and analog-to-digital converters. The modeling of the digital computer along with associated converters is covered in the next section.

13.2 Modeling the Digital Computer

If we think about it, the form of the signals in a loop is not as important as what happens to them. For example, if analog-to-digital conversion could happen instantaneously, and time samples occurred at intervals of time that approached zero, there would be no need to differentiate between the digital signals and the analog signals. Thus, previous analysis and design techniques would be valid regardless of the presence of the digital computer.

The fact that signals are sampled at specified intervals and held causes the system performance to change with changes in sampling rate. Basically, then, the computer's effect upon the signal comes from this sampling and holding. Thus, in order to model digital control systems, we must come up with a mathematical representation of this sample-and-hold process.

Modeling the Sampler

Our objective at this point is to derive a mathematical model for the digital computer as represented by a sampler and zero-order hold. Our goal is to represent the computer as a transfer function similar to that for any subsystem. When signals are sampled, however, the Laplace transform that we have dealt with becomes a bit unwieldy. The Laplace transform can be replaced by another related transform called the *z-transform*. The *z-transform* will arise naturally from our development of the mathematical representation of the computer.

Consider the models for sampling shown in Figure 13.5. The model in Figure 13.5(a) is a switch turning on and off at a uniform sampling rate. In Figure 13.5(b), sampling can also be considered to be the product of the time waveform to be sampled, $f(t)$, and a sampling function, $s(t)$. If $s(t)$ is a sequence of pulses of width T_w , constant amplitude, and uniform rate as shown, the sampled output, $f_{T_w}^*(t)$, will consist of a sequence of sections of $f(t)$ at regular intervals. This view is equivalent to the switch model of Figure 13.5(a).

We can now write the time equation of the sampled waveform, $f_{T_w}^*(t)$. Using the model shown in Figure 13.5(b), we have

$$f_{T_w}^*(t) = f(t)s(t) = f(t) \sum_{k=-\infty}^{\infty} u(t - kT) - u(t - kT - T_w) \quad (13.1)$$

where k is an integer between $-\infty$ and $+\infty$, T is the period of the pulse train, and T_w is the pulse width.

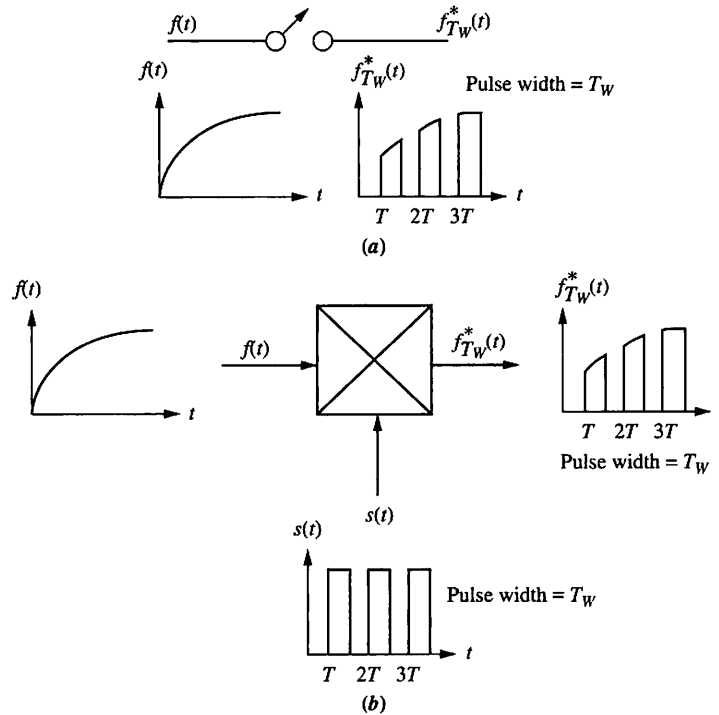


FIGURE 13.5 Two views of uniform-rate sampling:
a. switch opening and closing;
b. product of time waveform and sampling waveform

Since Eq. (13.1) is the product of two time functions, taking the Laplace transform in order to find a transfer function is not simple. A simplification can be made if we assume that the pulse width, T_w , is small in comparison to the period, T , such that $f(t)$ can be considered constant during the sampling interval. Over the sampling interval, then, $f(t) = f(kT)$. Hence,

$$f_{T_w}^*(t) = \sum_{k=-\infty}^{\infty} f(kT)[u(t - kT) - u(t - kT - T_w)] \quad (13.2)$$

for small T_w .

Equation (13.2) can be further simplified through insight provided by the Laplace transform. Taking the Laplace transform of Eq. (13.2), we have

$$F_{T_w}^*(s) = \sum_{k=-\infty}^{\infty} f(kT) \left[\frac{e^{-kTs}}{s} - \frac{e^{-kTs - T_w s}}{s} \right] = \sum_{k=-\infty}^{\infty} f(kT) \left[\frac{1 - e^{-T_w s}}{s} \right] e^{-kTs} \quad (13.3)$$

Replacing $e^{-T_w s}$ with its series expansion, we obtain

$$F_{T_w}^*(s) = \sum_{k=-\infty}^{\infty} f(kT) \left[\frac{1 - \left\{ 1 - T_w s + \frac{(T_w s)^2}{2!} - \dots \right\}}{s} \right] e^{-kTs} \quad (13.4)$$

For small T_w , Eq. (13.4) becomes

$$F_{T_w}^*(s) = \sum_{k=-\infty}^{\infty} f(kT) \left[\frac{T_w s}{s} \right] e^{-kTs} = \sum_{k=-\infty}^{\infty} f(kT) T_w e^{-kTs} \quad (13.5)$$

Finally, converting back to the time domain, we have

$$f_{T_w}^*(t) = T_w \sum_{k=-\infty}^{\infty} f(kT) \delta(t - kT) \quad (13.6)$$

where $\delta(t - kT)$ are Dirac delta functions.

Thus, the result of sampling with rectangular pulses can be thought of as a series of delta functions whose area is the product of the rectangular pulse width and the amplitude of the sampled waveform, or $T_w f(kT)$.

Equation (13.6) is portrayed in Figure 13.6. The sampler is divided into two parts: (1) an ideal sampler described by the portion of Eq. (13.6) that is not dependent upon the sampling waveform characteristics,

$$f^*(t) \sum_{k=-\infty}^{\infty} f(kT) \delta(t - kT) \quad (13.7)$$

and (2) the portion dependent upon the sampling waveform's characteristics, T_w .

Modeling the Zero-Order Hold

The final step in modeling the digital computer is modeling the zero-order hold that follows the sampler. Figure 13.7 summarizes the function of the zero-order hold,

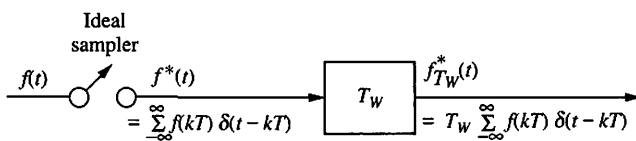


FIGURE 13.6 Model of sampling with a uniform rectangular pulse train

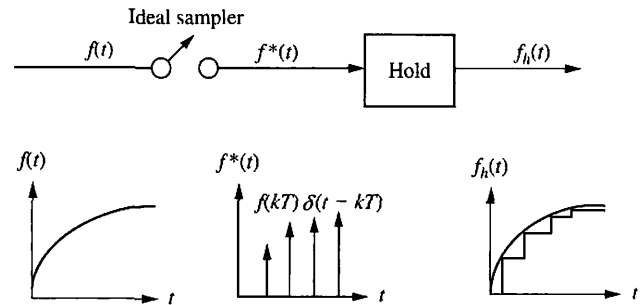


FIGURE 13.7 Ideal sampling and the zero-order hold

which is to hold the last sampled value of $f(t)$. If we assume an ideal sampler (equivalent to setting $T_w = 1$), then $f^*(t)$ is represented by a sequence of delta functions. The zero-order hold yields a staircase approximation to $f(t)$. Hence, the output from the hold is a sequence of step functions whose amplitude is $f(t)$ at the sampling instant, or $f(kT)$. We have previously seen that the transfer function of any linear system is identical to the Laplace transform of the impulse response since the Laplace transform of a unit impulse or delta function input is unity. Since a single impulse from the sampler yields a step over the sampling interval, the Laplace transform of this step, $G_h(s)$, which is the impulse response of the zero-order hold, is the transfer function of the zero-order hold. Using an impulse at zero time, the transform of the resulting step that starts at $t = 0$ and ends at $t = T$ is

$$G_h(s) = \frac{1 - e^{-Ts}}{s} \quad (13.8)$$

In a physical system, samples of the input time waveform, $f(kT)$, are held over the sampling interval. We can see from Eq. (13.8) that the hold circuit integrates the input and holds its value over the sampling interval. Since the area under the delta functions coming from the ideal sampler is $f(kT)$, we can then integrate the ideal sampled waveform and obtain the same result as for the physical system. In other words, if the ideal sampled signal, $f^*(t)$, is followed by a hold, we can use the ideal sampled waveform as the input, rather than $f_{T_w}^*(t)$.

In this section, we modeled the digital computer by cascading two elements: (1) an ideal sampler and (2) a zero-order hold. Together, the model is known as a *zero-order sample-and-hold*. The ideal sampler is modeled by Eq. (13.7), and the zero-order hold is modeled by Eq. (13.8). In the next section, we start to create a transform approach to digital systems by introducing the z -transform.

13.3 The z -Transform

The effect of sampling within a system is pronounced. Whereas the stability and transient response of analog systems depend upon gain and component values, sampled-data system stability and transient response also depend upon sampling rate. Our goal is to develop a transform that contains the information of sampling from which sampled-data systems can be modeled with transfer functions, analyzed, and designed with the ease and insight we enjoyed with the Laplace transform. We now develop such a transform and use the information from the last section to obtain sampled-data transfer functions for physical systems.

Equation (13.7) is the ideal sampled waveform. Taking the Laplace transform of this sampled time waveform, we obtain

$$F^*(s) = \sum_{k=0}^{\infty} f(kT)e^{-kTs} \quad (13.9)$$

Now, letting $z = e^{Ts}$, Eq. (13.9) can be written as

$$F(z) = \sum_{k=0}^{\infty} f(kT)z^{-k} \quad (13.10)$$

Equation (13.10) defines the *z-transform*. That is, an $F(z)$ can be transformed to $f(kT)$, or an $f(kT)$ can be transformed to $F(z)$. Alternately, we can write

$$f(kT) \xleftrightarrow{\quad} F(z) \quad (13.11)$$

Paralleling the development of the Laplace transform, we can form a table relating $f(kT)$, the value of the sampled time function at the sampling instants, to $F(z)$. Let us look at an example.

Example 13.1

z-Transform of a Time Function

PROBLEM: Find the z-transform of a sampled unit ramp.

SOLUTION: For a unit ramp, $f(kT) = kT$. Hence the ideal sampled step can be written from Eq. (13.7) as

$$f^*t = \sum_{k=0}^{\infty} kt\delta(t - kT) \quad (13.12)$$

Taking the Laplace transform, we obtain

$$F^*(s) = \sum_{k=0}^{\infty} kTe^{-kTs} \quad (13.13)$$

Converting to the z-transform by letting $e^{-kTs} = z^{-k}$, we have

$$F(z) = \sum_{k=0}^{\infty} kTz^{-k} = T \sum_{k=0}^{\infty} kz^{-k} = T(z^{-1} + 2z^{-2} + 3z^{-3} + \dots) \quad (13.14)$$

Equation (13.14) can be converted to a closed form by forming the series for $zF(z)$ and subtracting $F(z)$. Multiplying Eq. (13.14) by z , we get

$$zF(z) = T(1 + 2z^{-1} + 3z^{-2} + \dots) \quad (13.15)$$

Subtracting Eq. (13.14) from Eq. (13.15), we obtain

$$zF(z) - F(z) = (z - 1)F(z) = T(1 + z^{-1} + z^{-2} + \dots) \quad (13.16)$$

But

$$\frac{1}{1 - z^{-1}} = 1 + z^{-1} + z^{-2} + z^{-3} + \dots \quad (13.17)$$

which can be verified by performing the indicated division. Substituting Eq. (13.17) into (13.16) and solving for $F(z)$ yields

$$F(z) = T \frac{z}{(z-1)^2} \quad (13.18)$$

as the z -transform of $f(kT) = kT$.

Symbolic Math

SM

Students who are performing the MATLAB exercises and want to explore the added capability of MATLAB's Symbolic Math Toolbox should now run `ch13sp1` in Appendix F located at www.wiley.com/college/nise. You will learn how to find the z -transform of time functions. Example 13.1 will be solved using MATLAB and the Symbolic Math Toolbox.

The example demonstrates that any function of s , $F^*(s)$, that represents a sampled time waveform can be transformed into a function of z , $F(z)$. The final result, $F(z) = Tz/(z-1)^2$, is in a closed form, unlike $F^*(s)$. If this is the case for numerous other sampled time waveforms, then we have the convenient transform that we were looking for. In a similar way, z -transforms for other waveforms can be obtained that parallel the table of Laplace transforms in Chapter 2. A partial table of z -transforms is shown in Table 13.1, and a partial table of z -transform theorems is

TABLE 13.1 Partial table of z - and s -transforms

	$f(t)$	$F(s)$	$F(z)$	$f(kT)$
1.	$u(t)$	$\frac{1}{s}$	$\frac{z}{z-1}$	$u(kT)$
2.	t	$\frac{1}{s^2}$	$\frac{Tz}{(z-1)^2}$	kT
3.	t^n	$\frac{n!}{s^{n+1}}$	$\lim_{a \rightarrow 0} (-1)^n \frac{d^n}{da^n} \left[\frac{z}{z - e^{-aT}} \right]$	$(kT)^n$
4.	e^{-at}	$\frac{1}{s+a}$	$\frac{z}{z - e^{-aT}}$	e^{-akT}
5.	$t^n e^{-at}$	$\frac{n!}{(s+a)^{n+1}}$	$(-1)^n \frac{d^n}{da^n} \left[\frac{z}{z - e^{-aT}} \right]$	$(kT)^n e^{-akT}$
6.	$\sin \omega t$	$\frac{\omega}{s^2 + \omega^2}$	$\frac{z \sin \omega T}{z^2 - 2z \cos \omega T + 1}$	$\sin \omega kT$
7.	$\cos \omega t$	$\frac{s}{s^2 + \omega^2}$	$\frac{z(z - \cos \omega T)}{z^2 - 2z \cos \omega T + 1}$	$\cos \omega kT$
8.	$e^{-at} \sin \omega t$	$\frac{\omega}{(s+a)^2 + \omega^2}$	$\frac{ze^{-aT} \sin \omega T}{z^2 - 2ze^{-aT} \cos \omega T + e^{-2aT}}$	$e^{-akT} \sin \omega kT$
9.	$e^{-at} \cos \omega t$	$\frac{s+a}{(s+a)^2 + \omega^2}$	$\frac{z^2 - ze^{-aT} \cos \omega T}{z^2 - 2ze^{-aT} \cos \omega T + e^{-2aT}}$	$e^{-akT} \cos \omega kT$

TABLE 13.2 z-transform theorems

	Theorem	Name
1.	$z\{af(t)\} = aF(z)$	Linearity theorem
2.	$z\{f_1(t) + f_2(t)\} = F_1(z) + F_2(z)$	Linearity theorem
3.	$z\{e^{-aT}f(t)\} = F(e^{aT}z)$	Complex differentiation
4.	$z\{f(t - nT)\} = z^{-n}F(z)$	Real translation
5.	$z\{tf(t)\} = -Tz \frac{dF(z)}{dz}$	Complex differentiation
6.	$f(0) = \lim_{z \rightarrow \infty} F(z)$	Initial value theorem
7.	$f(\infty) = \lim_{z \rightarrow 1} (1 - z^{-1})F(z)$	Final value theorem

Note: kT may be substituted for t in the table.

shown in Table 13.2. For functions not in the table, we must perform an inverse z-transform calculation similar to the inverse Laplace transform by partial-fraction expansion. Let us now see how we can work in the reverse direction and find the time function from its z-transform.

The Inverse z-Transform

Two methods for finding the inverse z-transform (the sampled time function from its z-transform) will be described: (1) partial-fraction expansion and (2) the power series method. Regardless of the method used, remember that since the z-transform came from the sampled waveform, the inverse z-transform will yield only the values of the time function at the sampling instants. Keep this in mind as we proceed, because even as we obtain closed-form time functions as results, they are valid only at sampling instants.

Inverse z-Transforms via Partial-Fraction Expansion Recall that the Laplace transform consists of a partial fraction that yields a sum of terms leading to exponentials, that is, $A/(s + a)$. Taking this lead and looking at Table 13.1, we find that sampled exponential time functions are related to their z-transforms as follows:

$$e^{-akT} \longleftrightarrow \frac{z}{z - e^{aT}} \tag{13.19}$$

We thus predict that a partial-fraction expansion should be of the following form:

$$F(z) = \frac{Az}{z - z_1} + \frac{Bz}{z - z_2} + \dots \tag{13.20}$$

Since our partial-fraction expansion of $F(s)$ did not contain terms with s in the numerator of the partial fractions, we first form $F(z)/z$ to eliminate the z terms in the numerator, perform a partial-fraction expansion of $F(z)/z$, and finally multiply the result by z to replace the z 's in the numerator. An example follows.

Example 13.2

Inverse z-Transform via Partial-Fraction Expansion

PROBLEM: Given the function in Eq. (13.21), find the sampled time function.

$$F(z) = \frac{0.5z}{(z - 0.5)(z - 0.7)} \tag{13.21}$$

SOLUTION: Begin by dividing Eq. (13.21) by z and performing a partial-fraction expansion.

$$\frac{F(z)}{z} = \frac{0.5}{(z-0.5)(z-0.7)} = \frac{A}{z-0.5} + \frac{B}{z-0.7} = \frac{-2.5}{z-0.5} + \frac{2.5}{z-0.7} \quad (13.22)$$

Next, multiply through by z .

$$F(z) = \frac{0.5z}{(z-0.5)(z-0.7)} = \frac{-2.5z}{z-0.5} + \frac{2.5z}{z-0.7} \quad (13.23)$$

Using Table 13.1, we find the inverse z -transform of each partial fraction. Hence, the value of the time function at the sampling instants is

$$f(kT) = -2.5(0.5)^k + 2.5(0.7)^k \quad (13.24)$$

Also, from Eqs. (13.7) and (13.24), the ideal sampled time function is

$$f^*(t) = \sum_{k=-\infty}^{\infty} f(kT)\delta(t-kT) = \sum_{k=-\infty}^{\infty} [-2.5(0.5)^k + 2.5(0.7)^k]\delta(t-kT) \quad (13.25)$$

If we substitute $k = 0, 1, 2,$ and $3,$ we can find the first four samples of the ideal sampled time waveform. Hence,

$$f^*(t) = 0\delta(t) + 0.5\delta(t-T) + 0.6\delta(t-2T) + 0.545\delta(t-3T) \quad (13.26)$$

Symbolic Math

SM

Students who are performing the MATLAB exercises and want to explore the added capability of MATLAB's Symbolic Math Toolbox should now run `ch13sp2` in Appendix F located at www.wiley.com/college/nise. You will learn how to find the inverse z -transform of sampled time functions. Example 13.2 will be solved using MATLAB and the Symbolic Math Toolbox.

Inverse z -Transform via the Power Series Method The values of the sampled time waveform can also be found directly from $F(z)$. Although this method does not yield closed-form expressions for $f(kT)$, it can be used for plotting. The method consists of performing the indicated division, which yields a power series for $F(z)$. The power series can then be easily transformed into $F^*(s)$ and $f^*(t)$.

Example 13.3

Inverse z -Transform via Power Series

PROBLEM: Given the function in Eq. (13.21), find the sampled time function.

SOLUTION: Begin by converting the numerator and denominator of $F(z)$ to polynomials in z .

$$F(z) = \frac{0.5z}{(z-0.5)(z-0.7)} = \frac{0.5z}{z^2 - 1.2z + 0.35} \quad (13.27)$$

Now perform the indicated division.

$$\begin{array}{r}
 0.5z^{-1} + 0.6z^{-2} + 0.545z^{-3} \\
 z^2 - 1.2z + 0.35 \overline{)0.5z} \\
 \underline{0.5z - 0.6 + 0.175z^{-1}} \\
 0.6 - 0.175z^{-1} \\
 \underline{0.6 - 0.720z^{-1} + 0.21} \\
 0.545z^{-1} - 0.21
 \end{array} \quad (13.28)$$

Using the numerator and the definition of z , we obtain

$$F^*(s) = 0.5e^{-Ts} + 0.6e^{-2Ts} + 0.545e^{-3Ts} + \dots \quad (13.29)$$

from which

$$f^*(t) = 0.5\delta(t - T) + 0.6\delta(t - 2T) + 0.545\delta(t - 3T) + \dots \quad (13.30)$$

You should compare Eq. (13.30) with Eq. (13.26), the result obtained via partial expansion.

Skill-Assessment Exercise 13.1

PROBLEM: Derive the z -transform for $f(t) = \sin \omega t u(t)$.

ANSWER: $F(z) = \frac{z^{-1} \sin(\omega T)}{1 - 2z^{-1} \cos(\omega T) + z^{-2}}$

WileyPLUS
WPCS
Control Solutions

The complete solution is located at www.wiley.com/college/nise.

Skill-Assessment Exercise 13.2

PROBLEM: Find $f(kT)$ if $F(z) = \frac{z(z+1)(z+2)}{(z-0.5)(z-0.7)(z-0.9)}$.

ANSWER: $f(kT) = 46.875(0.5)^k - 114.75(0.7)^k + 68.875(0.9)^k$

The complete solution is located at www.wiley.com/college/nise.

13.4 Transfer Functions

Now that we have established the z -transform, let us apply it to physical systems by finding transfer functions of sampled-data systems. Consider the continuous system shown in Figure 13.8 (a). If the input is sampled as shown in Figure 13.8(b), the output is still a continuous signal. If, however, we are satisfied with finding the output at the sampling instants and not in between, the representation of the sampled-data system

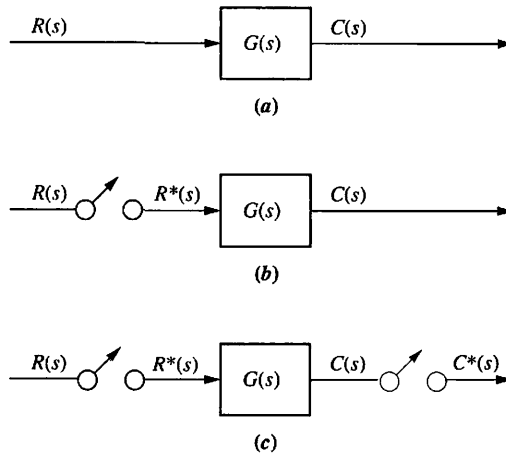


FIGURE 13.8 Sampled-data systems: **a.** continuous; **b.** sampled input; **c.** sampled input and output

Note: Phantom sampler is shown in color.

can be greatly simplified. Our assumption is visually described in Figure 13.8(c), where the output is conceptually sampled in synchronization with the input by a phantom sampler. Using the concept described in Figure 13.8(c), we derive the pulse transfer function of $G(s)$.

Derivation of the Pulse Transfer Function

Using Eq. (13.7), we find that the sampled input, $r^*(t)$, to the system of Figure 13.8(c) is

$$r^*(t) = \sum_{n=0}^{\infty} r(nT)\delta(t - nT) \quad (13.31)$$

which is a sum of impulses. Since the impulse response of a system, $G(s)$, is $g(t)$, we can write the time output of $G(s)$ as the sum of impulse responses generated by the input, Eq. (13.31). Thus,

$$c(t) = \sum_{n=0}^{\infty} r(nT)g(t - nT) \quad (13.32)$$

From Eq. (13.10),

$$C(z) = \sum_{k=0}^{\infty} c(kT)z^{-k} \quad (13.33)$$

Using Eq. (13.32) with $t = kT$, we obtain

$$c(kT) = \sum_{n=0}^{\infty} r(nT)g(kT - nT) \quad (13.34)$$

Substituting Eq. (13.34) into Eq. (13.33), we obtain

$$C(z) = \sum_{k=0}^{\infty} \sum_{n=0}^{\infty} r(nT)g[(k - n)T]z^{-k} \quad (13.35)$$

Letting $m = k - n$, we find

$$\begin{aligned} C(z) &= \sum_{m+n=0}^{\infty} \sum_{n=0}^{\infty} r(nT)g(mT)z^{-(m+n)} \\ &= \left\{ \sum_{m=0}^{\infty} g(mT)z^{-m} \right\} \left\{ \sum_{n=0}^{\infty} r(nT)z^{-n} \right\} \end{aligned} \quad (13.36)$$

where the lower limit, $m + n$, was changed to m . The reasoning is that $m + n = 0$ yields negative values of m for all $n > 0$. But, since $g(mT) = 0$ for all $m < 0$, m is not less than zero. Alternately, $g(t) = 0$ for $t < 0$. Thus, $n = 0$ in the first sum's lower limit.

Using the definition of the z -transform, Eq. (13.36) becomes

$$C(z) = \sum_{m=0}^{\infty} g(mT)z^{-m} \sum_{n=0}^{\infty} r(nT)z^{-n} = G(z)R(z) \quad (13.37)$$

Equation (13.37) is a very important result, since it shows that the transform of the sampled output is the product of the transforms of the sampled input and the pulse transfer function of the system. Remember that although the output of the system is a continuous function, we had to make an assumption of a sampled output (phantom sampler) in order to arrive at the compact result of Eq. (13.37).

One way of finding the pulse transfer function, $G(z)$, is to start with $G(s)$, find $g(t)$, and then use Table 13.1 to find $G(z)$. Let us look at an example.

Example 13.4

Converting $G_1(s)$ in Cascade with z.o.h. to $G(z)$

PROBLEM: Given a z.o.h. in cascade with $G_1(s) = (s + 2)/(s + 1)$ or

$$G(s) = \frac{1 - e^{-Ts}}{s} \frac{(s + 2)}{(s + 1)} \quad (13.38)$$

find the sampled-data transfer function, $G(z)$, if the sampling time, T , is 0.5 second.

SOLUTION: Equation (13.38) represents a common occurrence in digital control systems, namely a transfer function in cascade with a zero-order hold. Specifically, $G_1(s) = (s + 2)/(s + 1)$ is in cascade with a zero-order hold, $(1 - e^{-Ts})/s$. We can formulate a general solution to this type of problem by moving the s in the denominator of the zero-order hold to $G_1(s)$, yielding

$$G(s) = (1 - e^{-Ts}) \frac{G_1(s)}{s} \quad (13.39)$$

from which

$$G(z) = (1 - z^{-1})z \left\{ \frac{G_1(s)}{s} \right\} = \frac{z - 1}{z} z \left\{ \frac{G_1(s)}{s} \right\} \quad (13.40)$$

Thus, begin the solution by finding the impulse response (inverse Laplace transform) of $G_1(s)/s$. Hence,

$$G_2(s) = \frac{G_1(s)}{s} = \frac{s + 2}{s(s + 1)} = \frac{A}{s} + \frac{B}{s + 1} = \frac{2}{s} - \frac{1}{s + 1} \quad (13.41)$$

TryIt 13.1

Use MATLAB, the Control System Toolbox, and the following statements to find $G_1(s)$ in Example 13.4 given $G(z)$ in Eq. (13.46)

```
num=0.213;
den=0.607;
k=1;
T=0.5;
Gz=zpk(num,den,K,T)
Gs=d2c(Gz,'zoh')
```

MATLAB

ML

Symbolic Math

SM

Taking the inverse Laplace transform, we get

$$g_2(t) = 2 - e^{-t} \quad (13.42)$$

from which

$$g_2(kT) = 2 - e^{-kt} \quad (13.43)$$

Using Table 13.1, we find

$$G_2(z) = \frac{2z}{z-1} - \frac{z}{z-e^{-T}} \quad (13.44)$$

Substituting $T = 0.5$ yields

$$G_2(z) = z \left\{ \frac{G_1(s)}{s} \right\} = \frac{2z}{z-1} - \frac{z}{z-0.607} = \frac{z^2 - 0.213z}{(z-1)(z-0.607)} \quad (13.45)$$

From Eq. (13.40),

$$G(z) = \frac{z-1}{z} G_2(z) = \frac{z-0.213}{z-0.607} \quad (13.46)$$

Students who are using MATLAB should now run ch13p1 in Appendix B. You will learn how to use MATLAB to convert $G_1(s)$ in cascade with a zero-order hold to $G(z)$. This exercise solves Example 13.4 using MATLAB.

Students who are performing the MATLAB exercises and want to explore the added capability of MATLAB's Symbolic Math Toolbox should now run ch13sp3 in Appendix F located at www.wiley.com/college/nise. MATLAB's Symbolic Math Toolbox yields an alternative method of finding the z-transform of a transfer function in cascade with a zero-order hold. Example 13.4 will be solved using MATLAB and the Symbolic Math Toolbox with a method that follows closely the hand calculation shown in that example.

MATLAB

ML

Students who are using MATLAB should now run ch13p2 in Appendix B. You will learn how to use MATLAB to convert $G(s)$ to $G(z)$ when $G(s)$ is not in cascade with a zero-order hold. This is the same as finding the z-transform of $G(s)$.

MATLAB

ML

Students who are using MATLAB should now run ch13p3 in Appendix B. You will learn how to create digital transfer functions directly.

MATLAB

ML

Students who are using MATLAB should now run ch13p4 in Appendix B. You will learn how to use MATLAB to convert $G(z)$ to $G(s)$ when $G(s)$ is not in cascade with a zero-order hold. This is the same as finding the Laplace transform of $G(z)$.

Skill-Assessment Exercise 13.3**TryIt 13.2**

Use MATLAB, the Control System Toolbox, and the following statements to solve Skill-Assessment Exercise 13.3.

```
Gs=zpk([],-4,8)
Gz=c2d(Gs,0.25,'zoh')
```

PROBLEM: Find $G(z)$ for $G(s) = 8/(s+4)$ in cascade with a zero-order sample and hold. The sampling period is 0.25 second.

ANSWER: $G(z) = 1.264/(z - 0.3679)$

The complete solution is located at www.wiley.com/college/nise.

WileyPLUS

WPCS

Control Solutions

The major discovery in this section is that once the pulse transfer function, $G(z)$, of a system is obtained, the transform of the sampled output response, $C(z)$, for a given sampled input can be evaluated using the relationship $C(z) = R(z)G(z)$. Finally, the time function can be found by taking the inverse z -transform, as covered in Section 13.3. In the next section, we look at block diagram reduction for digital systems.

13.5 Block Diagram Reduction

Up to this point, we have defined the z -transform and the sampled-data system transfer function and have shown how to obtain the sampled response. Basically, we are paralleling our discussions of the Laplace transform in Chapters 2 and 4. We now draw a parallel with some of the objectives of Chapter 5, namely block diagram reduction. Our objective here is to be able to find the closed-loop sampled-data transfer function of an arrangement of subsystems that have a computer in the loop.

When manipulating block diagrams for sampled-data systems, you must be careful to remember the definition of the sampled-data system transfer function (derived in the last section) to avoid mistakes. For example, $z\{G_1(s)G_2(s)\} \neq G_1(z)G_2(z)$, where $z\{G_1(s)G_2(s)\}$ denotes the z -transform. The s -domain functions have to be multiplied together before taking the z -transform. In the ensuing discussion, we use the notation $G_1G_2(s)$ to denote a single function that is $G_1(s)G_2(s)$ after evaluating the product. Hence, $z\{G_1(s)G_2(s)\} = z\{G_1G_2(s)\} = G_1G_2(z) \neq G_1(z)G_2(z)$.

Let us look at the sampled-data systems shown in Figure 13.9. The sampled-data systems are shown under the column marked s . Their z -transforms are shown under the column marked z . The standard system that we derived earlier is shown in Figure 13.9(a), where the transform of the output, $C(z)$, is equal to $R(z)G(z)$. This system forms the basis for the other entries in Figure 13.9.

In Figure 13.9(b), there is no sampler between $G_1(s)$ and $G_2(s)$. Thus, we can think of a single function, $G_1(s)G_2(s)$, denoted $G_1G_2(s)$, existing between the two samplers and yielding a single transfer function, as shown in Figure 13.9(a). Hence, the pulse transfer function is $z\{G_1G_2(s)\} = G_1G_2(z)$. The transform of the output, $C(z) = R(z)G_1G_2(z)$.

In Figure 13.9(c), we have the cascaded two subsystems of the type shown in Figure 13.9(a). For this case, then, the z -transform is the product of the two z -transforms, or $G_2(z)G_1(z)$. Hence the transform of the output $C(z) = R(z)G_2(z)G_1(z)$.

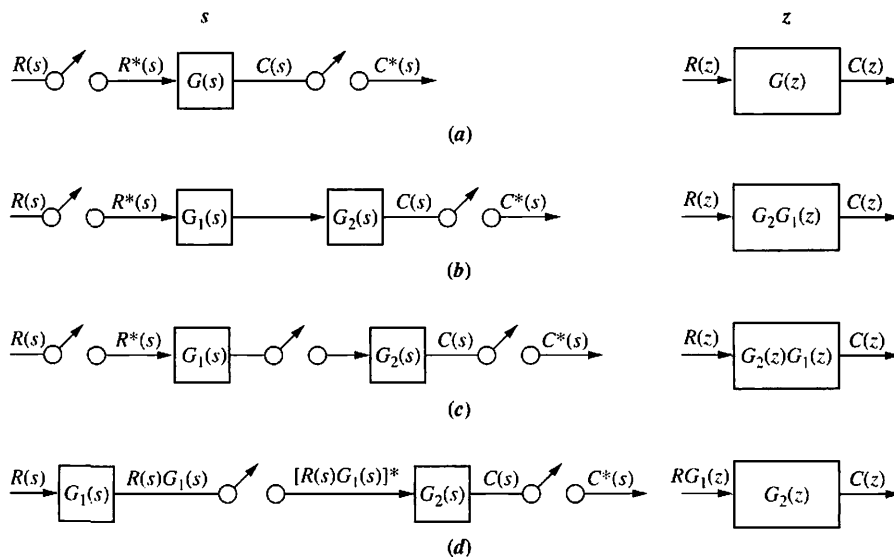


FIGURE 13.9 Sampled-data systems and their z -transforms

Finally, in Figure 13.9(d), we see that the continuous signal entering the sampler is $R(s)G_1(s)$. Thus, the model is the same as Figure 13.9(a) with $R(s)$ replaced by $R(s)G_1(s)$, and $G_2(s)$ in Figure 13.9(d) replacing $G(s)$ in Figure 13.9(a). The z -transform of the input to $G_2(s)$ is $z\{R(s)G_1(s)\} = z\{RG_1(s)\} = RG_1(z)$. The pulse transfer function for the system $G_2(s)$ is $G_2(z)$. Hence, the output $C(z) = RG_1(z)G_2(z)$.

Using the basic forms shown in Figure 13.9, we can now find the z -transform of feedback control systems. We have shown that any system, $G(s)$, with sampled input and sampled output, such as that shown in Figure 13.9(a), can be represented as a sampled-data transfer function, $G(z)$. Thus, we want to perform block diagram manipulations that result in subsystems, as well as the entire feedback system, that have sampled inputs and sampled outputs. Then we can make the transformation to sampled-data transfer functions. An example follows.

Example 13.5

Pulse Transfer Function of a Feedback System

PROBLEM: Find the z -transform of the system shown in Figure 13.10(a).

SOLUTION: The objective of the problem is to proceed in an orderly fashion, starting with the block diagram of Figure 13.10(a) and reducing it to the one shown in Figure 13.10(f).

One operation we can always perform is to place a phantom sampler at the output of any subsystem that has a sampled input, provided that the nature of the signal sent to any other subsystem is not changed. For example in Figure 13.10(b), phantom sampler S_4 can be added. The justification for this, of course, is that the

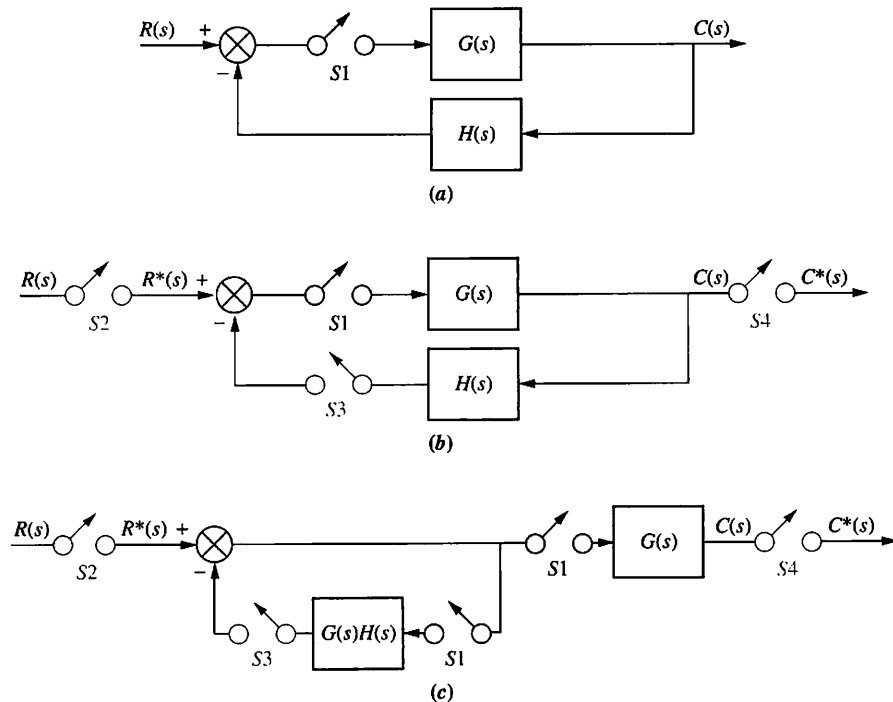


FIGURE 13.10 Steps in block diagram reduction of a sampled-data system (figure continues)

Note: Phantom samplers are shown in color.

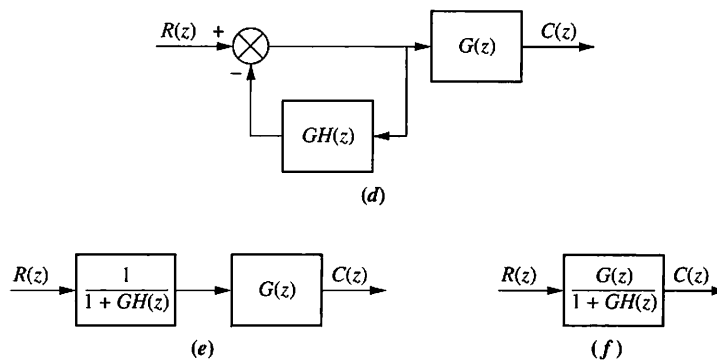


FIGURE 13.10 (Continued)

output of a sampled-data system can only be found at the sampling instants anyway, and the signal is not an input to any other block.

Another operation that can be performed is to add phantom samplers $S2$ and $S3$ at the input to a summing junction whose output is sampled. The justification for this operation is that the sampled sum is equivalent to the sum of the sampled inputs, provided, of course, that all samplers are synchronized.

Next, move sampler $S1$ and $G(s)$ to the right past the pickoff point, as shown in Figure 13.10(c). The motivation for this move is to yield a sampler at the input of $G(s)H(s)$ to match Figure 13.9(b). Also, $G(s)$ with sampler $S1$ at the input and sampler $S4$ at the output matches Figure 13.9(a). The closed-loop system now has a sampled input and a sampled output.

$G(s)H(s)$ with samplers $S1$ and $S3$ becomes $GH(z)$, and $G(s)$ with samplers $S1$ and $S4$ becomes $G(z)$, as shown in Figure 13.10(d). Also, converting $R^*(s)$ to $R(z)$ and $C^*(s)$ to $C(z)$, we now have the system represented totally in the z -domain.

The equations derived in Chapter 5 for transfer functions represented with the Laplace transform can be used for sampled-data transfer functions with only a change in variables from s to z . Thus, using the feedback formula, we obtain the first block of Figure 13.10(e). Finally, multiplication of the cascaded sampled-data systems yields the final result shown in Figure 13.10(f).

Skill-Assessment Exercise 13.4

PROBLEM: Find $T(z) = C(z)/R(z)$ for the system shown in Figure 13.11.

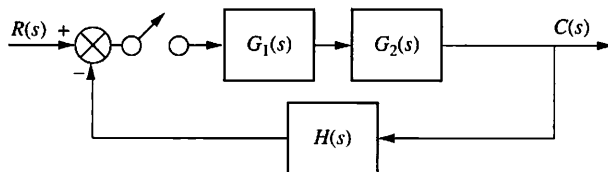


FIGURE 13.11 Digital system for Skill-Assessment Exercise 13.4

ANSWER:
$$T(z) = \frac{G_1 G_2(z)}{1 + H G_1 G_2(z)}$$

The complete solution is located at www.wiley.com/college/nise.

This section paralleled Chapter 5 by showing how to obtain the closed-loop, sampled-data transfer function for a collection of subsystems. The next section parallels the discussion of stability in Chapter 6.

13.6 Stability

The glaring difference between analog feedback control systems and digital feedback control systems, such as the one shown in Figure 13.12, is the effect that the sampling rate has on the transient response. Changes in sampling rate not only change the nature of the response from overdamped to underdamped, but also can turn a stable system into an unstable one. As we proceed with our discussion, these effects will become apparent. You are encouraged to be on the lookout.

We now discuss the stability of digital systems from two perspectives: (1) z -plane and (2) s -plane. We will see that the Routh-Hurwitz criterion can be used only if we perform our analysis and design on the s -plane.

Digital System Stability via the z -Plane

In the s -plane, the region of stability is the left half-plane. If the transfer function, $G(s)$, is transformed into a sampled-data transfer function, $G(z)$, the region of stability on the z -plane can be evaluated from the definition, $z = e^{Ts}$. Letting $s = \alpha + j\omega$, we obtain

$$\begin{aligned} z = e^{Ts} &= e^{T(\alpha + j\omega)} = e^{\alpha T} e^{j\omega T} \\ &= e^{\alpha T} (\cos \omega T + j \sin \omega T) \\ &= e^{\alpha T} \angle \omega T \end{aligned} \quad (13.47)$$

since $(\cos \omega T + j \sin \omega T) = 1 \angle \omega T$.

Each region of the s -plane can be mapped into a corresponding region on the z -plane (see Figure 13.13). Points that have positive values of α are in the right half



FIGURE 13.12 A lathe using digital numerical control (© David J. Green—Industry/Alamy)

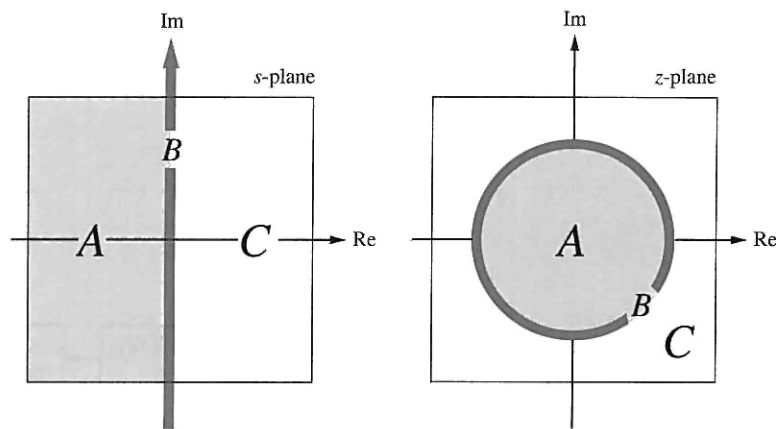


FIGURE 13.13 Mapping regions of the s -plane onto the z -plane

of the s -plane, region C . From Eq. (13.47), the magnitudes of the mapped points are $e^{\alpha T} > 1$. Thus points in the right half of the s -plane map into points outside the unit circle on the z -plane.

Points on the $j\omega$ -axis, region B , have zero values of α and yield points on the z -plane with magnitude = 1, the unit circle. Hence, points on the $j\omega$ -axis in the s -plane map into points on the unit circle on the z -plane.

Finally, points on the s -plane that yield negative values of α (left-half-plane roots, region A) map into the inside of the unit circle on the z -plane.

Thus, a digital control system is (1) stable if all poles of the closed-loop transfer function, $T(z)$, are inside the unit circle on the z -plane, (2) unstable if any pole is outside the unit circle and/or there are poles of multiplicity greater than one on the unit circle, and (3) marginally stable if poles of multiplicity one are on the unit circle and all other poles are inside the unit circle. Let us look at an example.

Example 13.6

Modeling and Stability

PROBLEM: The missile shown in Figure 13.14(a) can be aerodynamically controlled by torques created by the deflection of control surfaces on the missile's body. The commands to deflect these control surfaces come from a computer that uses tracking data along with programmed guidance equations to determine whether the missile is on track. The information from the guidance equations is used to develop flight-control commands for the missile. A simplified model is shown in Figure 13.14(b). Here the computer performs the function of controller by using tracking information to develop input commands to the missile. An accelerometer in the missile detects the actual acceleration, which is fed back to the computer. Find the closed-loop digital transfer function for this system and determine if the system is stable for $K = 20$ and $K = 100$ with a sampling interval of $T = 0.1$ second.

SOLUTION: The input to the control system is an acceleration command developed by the computer. The computer can be modeled by a sample-and-hold. The s -plane model is shown in Figure 13.14(c). The first step in finding the z -plane model is to find $G(z)$, the forward-path transfer function. From Figure 13.14(c) or (d),

$$G(s) = \frac{1 - e^{-Ts}}{s} \frac{Ka}{s(s+a)} \quad (13.48)$$

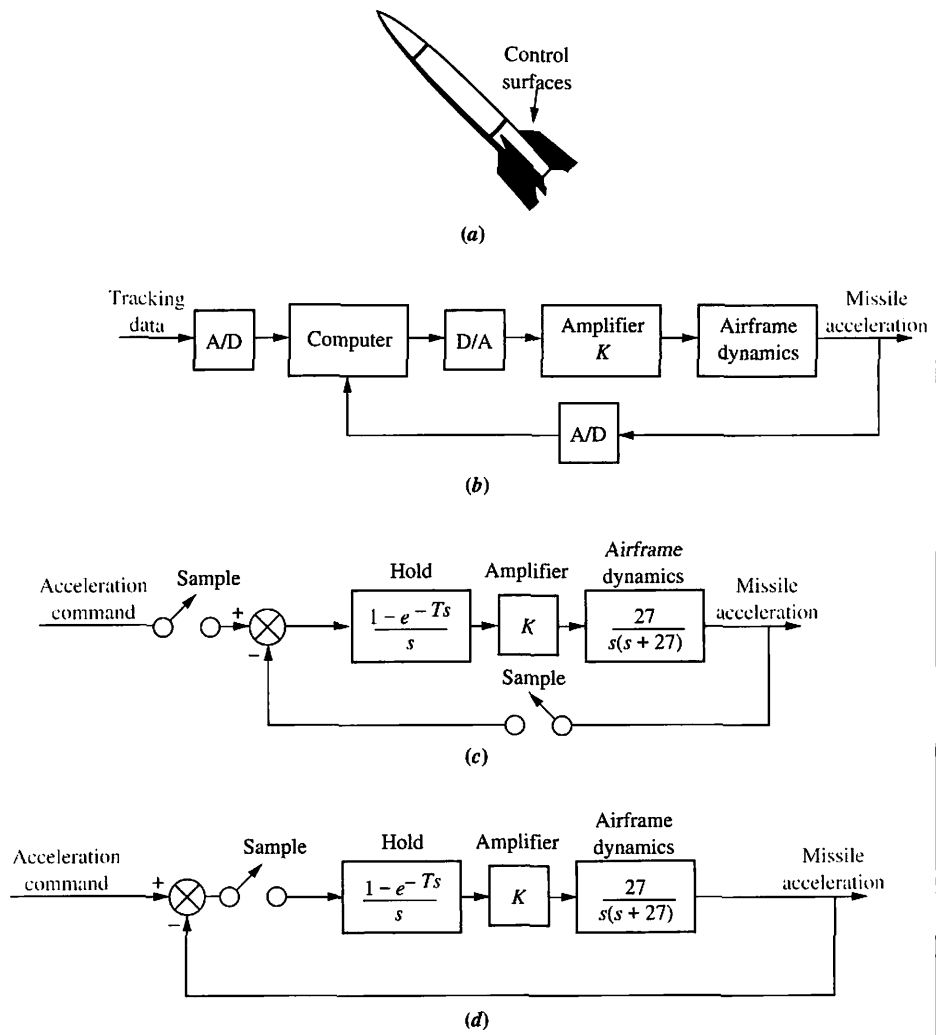


FIGURE 13.14 Finding stability of a missile control system: **a.** missile; **b.** conceptual block diagram; **c.** block diagram; **d.** block diagram with equivalent single sampler

where $a = 27$. The z -transform, $G(z)$, is $(1 - z^{-1})z\{Ka/[s^2(s + a)]\}$.

The term $Ka/[s^2(s + a)]$ is first expanded by partial fractions, after which we find the z -transform of each term from Table 13.1. Hence,

$$\begin{aligned} z\left\{\frac{Ka}{s^2(s+a)}\right\} &= Kz\left\{\frac{a}{s^2(s+a)}\right\} = Kz\left\{\frac{1}{s^2} - \frac{1/a}{s} + \frac{1/a}{s+a}\right\} \\ &= K\left\{\frac{Tz}{(z-1)^2} - \frac{z/a}{z-1} + \frac{z/a}{z-e^{-aT}}\right\} \\ &= K\left\{\frac{Tz}{(z-1)^2} - \frac{(1-e^{-aT})z}{a(z-1)(z-e^{-aT})}\right\} \end{aligned} \tag{13.49}$$

Thus,

$$G(z) = K\left\{\frac{T(z-e^{-aT}) - (z-1)\left(\frac{1-e^{-aT}}{a}\right)}{(z-1)(z-e^{-aT})}\right\} \tag{13.50}$$

Letting $T = 0.1$ and $a = 27$, we have

$$G(z) = \frac{K(0.0655z + 0.02783)}{(z - 1)(z - 0.0672)} \quad (13.51)$$

Finally, we find the closed-loop transfer function, $T(z)$, for a unity feedback system:

$$T(z) = \frac{G(z)}{1 + G(z)} = \frac{K(0.0655z + 0.02783)}{z^2 + (0.0655K - 1.0672)z + (0.02783K + 0.0672)} \quad (13.52)$$

The stability of the system is found by finding the roots of the denominator. For $K = 20$, the roots of the denominator are $0.12 \pm j0.78$. The system is thus stable for $K = 20$, since the poles are inside the unit circle. For $K = 100$, the poles are at -0.58 and -4.9 . Since one of the poles is outside the unit circle, the system is unstable for $K = 100$.

Students who are using MATLAB should now run `ch13p5` in Appendix B. You will learn how to use MATLAB to determine the range of K for stability in a digital system. This exercise solves Example 13.6 using MATLAB.

MATLAB
ML

In the case of continuous systems, the determination of stability hinges upon our ability to determine whether the roots of the denominator of the closed-loop transfer function are in the stable region of the s -plane. The problem for high-order systems is complicated by the fact that the closed-loop transfer function denominator is in polynomial form, not factored form. The same problem surfaces with closed-loop sampled-data transfer functions.

Tabular methods for determining stability, such as the Routh-Hurwitz method used for higher-order continuous systems, exist for sampled-data systems. These methods, which are not covered in this introductory chapter to digital control systems, can be used to determine stability in higher-order digital systems. If you wish to go further into the area of digital system stability, you are encouraged to look at Raible's tabular method or Jury's stability test for determining the number of a sampled-data system's closed-loop poles that exist outside the unit circle and thus indicate instability.²

The following example demonstrates the effect of sampling rate on the stability of a closed-loop feedback control system. All parameters are constant except for the sampling interval, T . We will see that varying T will lead us through regions of stability and instability just as though we were varying the forward-path gain, K .

Example 13.7

Range of T for Stability

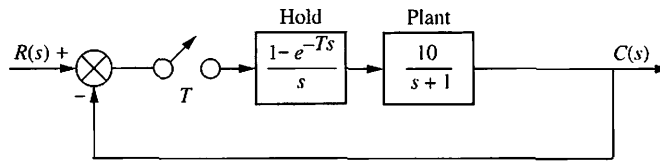
PROBLEM: Determine the range of sampling interval, T , that will make the system shown in Figure 13.15 stable, and the range that will make it unstable.

SOLUTION: Since $H(s) = 1$, the z -transform of the closed-loop system, $T(z)$, is found from Figure 13.10 to be

$$T(z) = \frac{G(z)}{1 + G(z)} \quad (13.53)$$

² A discussion of Raible's tabular method and Jury's stability test can be found in Kuo (1980: 278–286).

FIGURE 13.15 Digital system for Example 13.7



To find $G(z)$, first find the partial-fraction expansion of $G(s)$.

$$G(s) = 10 \frac{1 - e^{-Ts}}{s(s+1)} = 10(1 - e^{-Ts}) \left(\frac{1}{s} - \frac{1}{s+1} \right) \quad (13.54)$$

Taking the z -transform, we obtain

$$G(z) = \frac{10(z-1)}{z} \left[\frac{z}{z-1} - \frac{z}{z-e^{-T}} \right] = 10 \frac{(1 - e^{-T})}{(z - e^{-T})} \quad (13.55)$$

Substituting Eq. (13.55) into (13.53) yields

$$T(z) = \frac{10(1 - e^{-T})}{z - (11e^{-T} - 10)} \quad (13.56)$$

The pole of Eq. (13.56), $(11e^{-T} - 10)$, monotonically decreases from $+1$ to -1 for $0 < T < 0.2$. For $0.2 < T < \infty$, $(11e^{-T} - 10)$ monotonically decreases from -1 to -10 . Thus, the pole of $T(z)$ will be inside the unit circle, and the system will be stable if $0 < T < 0.2$. In terms of frequency, where $f = 1/T$, the system will be stable as long as the sampling frequency is $1/0.2 = 5$ hertz or greater.

We now have found, via the z -plane, that sampled systems are stable if their poles are inside the unit circle. Unfortunately, this stability criterion precludes the use of the Routh-Hurwitz criterion, which detects roots in the right half-plane rather than outside the unit circle. However, another method exists that allows us to use the familiar s -plane and the Routh-Hurwitz criterion to determine the stability of a sampled system. Let us introduce this topic.

Bilinear Transformations

Bilinear transformations give us the ability to apply our s -plane analysis and design techniques to digital systems. We can analyze and design on the s -plane as we have done in Chapters 8 and 9 and then, using these transformations, convert the results to a digital system that contains the same properties. Let us look further into this topic.

We can consider $z = e^{Ts}$ and its inverse, $s = (1/T) \ln z$, as the exact transformations between z and s . Thus, if we have $G(z)$ and substitute $z = e^{Ts}$, we obtain $G(e^{Ts})$ as the result of converting to s . Similarly, if we have $G(s)$ and substitute $s = (1/T) \ln z$, we obtain $G((1/T) \ln z)$ as the result of converting to z . Unfortunately, both transformations yield transcendental functions, which we of course take care of through the rather complicated z -transform.

What we would like is a simple transformation that would yield linear arguments when transforming in both directions (bilinear) through direct substitution and without the complicated z -transform.

Bilinear transformations of the form

$$z = \frac{as + b}{cs + d} \quad (13.57)$$

and its inverse,

$$s = \frac{-dz + b}{cz - a} \quad (13.58)$$

have been derived to yield linear variables in s and z . Different values of a , b , c , and d have been derived for particular applications and yield various degrees of accuracy when comparing properties of the continuous and sampled functions.

For example, in the next subsection we will see that a particular choice of coefficients will take points on the unit circle and map them into points on the $j\omega$ -axis. Points outside the unit circle will be mapped into the right half-plane, and points inside the unit circle will be mapped into the left half-plane. Thus, we will be able to make a simple transformation from the z -plane to the s -plane and obtain stability information about the digital system by working in the s -plane.

Since the transformations are not exact, only the property for which they are designed can be relied upon. For the stability transformation just discussed, we cannot expect the resulting $G(s)$ to have the same transient response as $G(z)$. Another transformation will be covered that will retain that property.

Digital System Stability via the s -Plane

In this subsection, we look at a bilinear transformation that maps $j\omega$ -axis points on the s -plane to unit-circle points on the z -plane. Further, the transformation maps right-half-plane points on the s -plane to points outside the unit circle on the z -plane. Finally, the transformation maps left-half-plane points on the s -plane to points inside the unit circle on the z -plane. Thus, we are able to transform the denominator of the pulsed transfer function, $D(z)$, to the denominator of a continuous transfer function, $D(s)$, and use the Routh-Hurwitz criterion to determine stability.

The bilinear transformation

$$s = \frac{z + 1}{z - 1} \quad (13.59)$$

and its inverse

$$z = \frac{s + 1}{s - 1} \quad (13.60)$$

perform the required transformation (Kuo, 1995). We can show this fact as follows: Letting $s = \alpha + j\omega$ and substituting into Eq. (13.60),

$$z = \frac{(\alpha + 1) + j\omega}{(\alpha - 1) + j\omega} \quad (13.61)$$

from which

$$|z| = \frac{\sqrt{(\alpha + 1)^2 + \omega^2}}{\sqrt{(\alpha - 1)^2 + \omega^2}} \quad (13.62)$$

Thus,

$$|z| < 1 \quad \text{when } \alpha < 0 \quad (13.63a)$$

$$|z| > 1 \quad \text{when } \alpha > 0 \quad (13.63b)$$

and

$$|z| = 1 \quad \text{when } \alpha = 0 \tag{13.63c}$$

Let us look at an example that shows how the stability of sampled systems can be found using this bilinear transformation and the Routh-Hurwitz criterion.

Example 13.8

Stability via Routh-Hurwitz

PROBLEM: Given $T(z) = N(z)/D(z)$, where $D(z) = z^3 - z^2 - 0.2z + 0.1$, use the Routh-Hurwitz criterion to find the number of z -plane poles of $T(z)$ inside, outside, and on the unit circle. Is the system stable?

SOLUTION: Substitute Eq. (13.60) into $D(z) = 0$ and obtain³

$$s^3 - 19s^2 - 45s - 17 = 0 \tag{13.64}$$

The Routh table for Eq. (13.64), Table 13.3, shows one root in the right-half-plane and two roots in the left-half-plane. Hence, $T(z)$ has one pole outside the unit circle, no poles on the unit circle, and two poles inside the unit circle. The system is unstable because of the pole outside the unit circle.

TABLE 13.3 Routh table for Example 13.8

s^3	1	-45
s^2	19	-17
s^1	-45.89	0
s^0	-17	0

Skill-Assessment Exercise 13.5

WileyPLUS
WPCS
 Control Solutions

PROBLEM: Determine the range of sampling interval, T , that will make the system shown in Figure 13.16 stable.

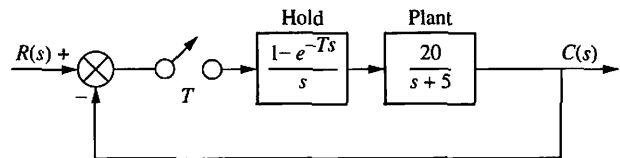


FIGURE 13.16 Digital system for Skill-Assessment Exercise 13.5

ANSWER: $0 < T < 0.1022$ second

The complete solution is located at www.wiley.com/college/nise.

³Symbolic math software, such as MATLAB's Symbolic Math Toolbox, is recommended to reduce the labor required to perform the transformation.

Skill-Assessment Exercise 13.6

PROBLEM: Given $T(z) = N(z)/D(z)$, where $D(z) = z^3 - z^2 - 0.5z + 0.3$, use the Routh-Hurwitz criterion to find the number of z -plane poles of $T(z)$ inside, outside, and on the unit circle. Is the system stable?

ANSWER: $T(z)$ has one pole outside the unit circle, no poles on the unit circle, and two poles inside the unit circle. The system is unstable.

The complete solution is located at www.wiley.com/college/nise.

In this section, we covered the concepts of stability for digital systems. Both z - and s -plane perspectives were discussed. Using a bilinear transformation, we are able to use the Routh-Hurwitz criterion to determine stability.

The highlight of the section is that sampling rate (along with system parameters, such as gain and component values) helps to determine or destroy the stability of a digital system. In general, if the sampling rate is too slow, the closed-loop digital system will be unstable. We now move from stability to steady-state errors, paralleling our previous discussion of steady-state errors in analog systems.

13.7 Steady-State Errors

We now examine the effect of sampling upon the steady-state error for digital systems. Any general conclusion about the steady-state error is difficult because of the dependence of those conclusions upon the placement of the sampler in the loop. Remember that the position of the sampler could change the open-loop transfer function. In the discussion of analog systems, there was only one open-loop transfer function, $G(s)$, upon which the general theory of steady-state error was based and from which came the standard definitions of static error constants. For digital systems, however, the placement of the sampler changes the open-loop transfer function and thus precludes any general conclusions. In this section, we assume the typical placement of the sampler after the error and in the position of the cascade controller, and we derive our conclusions accordingly about the steady-state error of digital systems.

Consider the digital system in Figure 13.17(a), where the digital computer is represented by the sampler and zero-order hold. The transfer function of the plant is represented by $G_1(s)$ and the transfer function of the z.o.h. by $(1 - e^{-Ts})/s$. Letting $G(s)$ equal the product of the z.o.h. and $G_1(s)$, and using the block diagram reduction techniques for sampled-data systems, we can find the sampled error, $E^*(s) = E(z)$. Adding synchronous samplers at the input and the feedback, we obtain Figure 13.17(b). Pushing $G(s)$ and its input sampler to the right past the pickoff point yields Figure 13.17(c). Using Figure 13.9(a), we can convert each block to its z -transform, resulting in Figure 13.17(d).

From this figure, $E(z) = R(z) - E(z)G(z)$, or

$$E(z) = \frac{R(z)}{1 + G(z)} \quad (13.65)$$

The final value theorem for discrete signals states that

$$e^*(\infty) = \lim_{z \rightarrow 1} (1 - z^{-1})E(z) \quad (13.66)$$

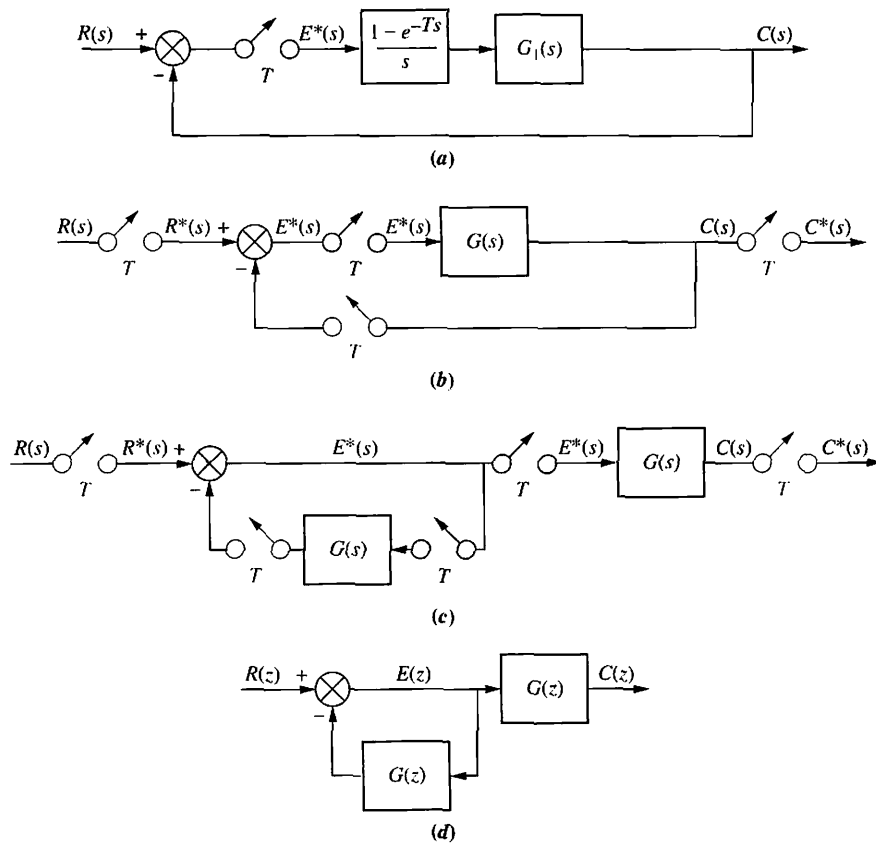


FIGURE 13.17 a. Digital feedback control system for evaluation of steady-state errors; b. phantom samplers added; c. pushing $G(s)$ and its samplers to the right past the pickoff point; d. z-transform equivalent system

Note: Phantom samplers are shown in color.

where $e^*(\infty)$ is the final sampled value of $e(t)$, or (alternatively) the final value of $e(kT)$.⁴

Using the final value theorem on Eq. (13.65), we find that the sampled steady-state error, $e^*(\infty)$, for unity negative-feedback systems is

$$e^*(\infty) = \lim_{z \rightarrow 1} (1 - z^{-1})E(z) = \lim_{z \rightarrow 1} (1 - z^{-1}) \frac{R(z)}{1 + G(z)} \quad (13.67)$$

Equation (13.67) must now be evaluated for each input: step, ramp, and parabola.

Unit Step Input

For a unit step input, $R(s) = 1/s$. From Table 13.1,

$$R(z) = \frac{z}{z - 1} \quad (13.68)$$

Substituting Eq. (13.68) into Eq. (13.67), we have

$$e^*(\infty) = \frac{1}{1 + \lim_{z \rightarrow 1} G(z)} \quad (13.69)$$

⁴ See Ogata (1987: 59) for a derivation.

Defining the static error constant, K_p , as

$$K_p = \lim_{z \rightarrow 1} G(z) \quad (13.70)$$

we rewrite Eq. (13.69) as

$$e^*(\infty) = \frac{1}{1 + K_p} \quad (13.71)$$

Unit Ramp Input

For a unit ramp input, $R(z) = Tz/(z - 1)^2$. Following the procedure for the step input, you can derive the fact that

$$e^*(\infty) = \frac{1}{K_v} \quad (13.72)$$

where

$$K_v = \frac{1}{T} \lim_{z \rightarrow 1} (z - 1)G(z) \quad (13.73)$$

Unit Parabolic Input

For a unit parabolic input, $R(z) = T^2z/(z + 1)/[2(z - 1)^3]$. Similarly,

$$e^*(\infty) = \frac{1}{K_a} \quad (13.74)$$

where

$$K_a = \frac{1}{T^2} \lim_{z \rightarrow 1} (z - 1)^2 G(z) \quad (13.75)$$

Summary of Steady-State Errors

The equations developed above for $e^*(\infty)$, K_p , K_v , and K_a are similar to the equations developed for analog systems. Whereas multiple pole placement at the origin of the s -plane reduced steady-state errors to zero in the analog case, we can see that multiple pole placement at $z = 1$ reduces the steady-state error to zero for digital systems of the type discussed in this section. This conclusion makes sense when one considers that $s = 0$ maps into $z = 1$ under $z = e^{Ts}$.

For example, for a step input, we see that if $G(z)$ in Eq. (13.69) has one pole at $z = 1$, the limit will become infinite, and the steady-state error will reduce to zero.

For a ramp input, if $G(z)$ in Eq. (13.73) has two poles at $z = 1$, the limit will become infinite, and the error will reduce to zero.

Similar conclusions can be drawn for the parabolic input and Eq. (13.75). Here, $G(z)$ needs three poles at $z = 1$ in order for the steady-state error to be zero. Let us look at an example.

Example 13.9

Finding Steady-State Error

PROBLEM: For step, ramp, and parabolic inputs, find the steady-state error for the feedback control system shown in Figure 13.17(a) if

$$G_1(s) = \frac{10}{s(s+1)} \quad (13.76)$$

SOLUTION: First find $G(s)$, the product of the z.o.h. and the plant.

$$G(s) = \frac{10(1 - e^{-Ts})}{s^2(s+1)} = 10(1 - e^{-Ts}) \left[\frac{1}{s^2} - \frac{1}{s} + \frac{1}{s+1} \right] \quad (13.77)$$

The z-transform is then

$$\begin{aligned} G(z) &= 10(1 - z^{-1}) \left[\frac{Tz}{(z-1)^2} - \frac{z}{z-1} + \frac{z}{z - e^{-T}} \right] \\ &= 10 \left[\frac{T}{z-1} - 1 + \frac{z-1}{z - e^{-T}} \right] \end{aligned} \quad (13.78)$$

For a step input,

$$K_p = \lim_{z \rightarrow 1} G(z) = \infty; \quad e^*(\infty) = \frac{1}{1 + K_p} = 0 \quad (13.79)$$

For a ramp input,

$$K_v = \frac{1}{T} \lim_{z \rightarrow 1} (z-1)G(z) = 10; \quad e^*(\infty) = \frac{1}{K_v} = 0.1 \quad (13.80)$$

For a parabolic input,

$$K_a = \frac{1}{T^2} \lim_{z \rightarrow 1} (z-1)^2 G(z) = 0; \quad e^*(\infty) = \frac{1}{K_a} = \infty \quad (13.81)$$

You will notice that the answers obtained are the same as the results obtained for the analog system. However, since stability depends upon the sampling interval, be sure to check the stability of the system after a sampling interval is established before making steady-state error calculations.

MATLAB
ML

Students who are using MATLAB should now run ch13p6 in Appendix B. You will learn how to use MATLAB to determine K_p , K_v , and K_a in a digital system as well as check the stability. This exercise solves Example 13.9 using MATLAB.

Skill-Assessment Exercise 13.7

WileyPLUS
WPCS
Control Solutions

PROBLEM: For step, ramp, and parabolic inputs, find the steady-state error for the feedback control system shown in Figure 13.17(a) if

$$G_1(s) = \frac{20(s+3)}{(s+4)(s+5)}$$

Let $T = 0.1$ second. Repeat for $T = 0.5$ second.

ANSWER: For $T = 0.1$ second, $K_p = 3$, $K_v = 0$, and $K_a = 0$; for $T = 0.5$ second, the system is unstable.

The complete solution is located at www.wiley.com/college/nise.

In this section, we discussed and evaluated the steady-state error of digital systems for step, ramp, and parabolic inputs. The equations for steady-state error parallel those for analog systems. Even the definitions of the static error constants were similar. Poles at the origin of the s -plane for analog systems were replaced with poles at $+1$ on the z -plane to improve the steady-state error. We continue our parallel discussion by moving into a discussion of transient response and the root locus for digital systems.

13.8 Transient Response on the z-Plane

Recall that for analog systems a transient response requirement was specified by selecting a closed-loop, s -plane pole. In Chapter 8, the closed-loop pole was on the existing root locus, and the design consisted of a simple gain adjustment. If the closed-loop pole was not on the existing root locus, then a cascade compensator was designed to reshape the original root locus to go through the desired closed-loop pole. A gain adjustment then completed the design.

In the next two sections, we want to parallel the described analog methods and apply similar techniques to digital systems. For this introductory chapter, we will parallel the discussion through design via gain adjustment. The design of compensation is left to you to pursue in an advanced course.

Chapter 4 established the relationships between transient response and the s -plane. We saw that vertical lines on the s -plane were lines of constant settling time, horizontal lines were lines of constant peak time, and radial lines were lines of constant percent overshoot. In order to draw equivalent conclusions on the z -plane, we now map those lines through $z = e^{sT}$.

The vertical lines on the s -plane are lines of constant settling time and are characterized by the equation $s = \sigma_1 + j\omega$, where the real part, $\sigma_1 = -4/T_s$, is constant and is in the left-half-plane for stability. Substituting this into $z = e^{sT}$, we obtain

$$z = e^{\sigma_1 T} e^{j\omega T} = r_1 e^{j\omega T} \quad (13.82)$$

Equation (13.82) denotes concentric circles of radius r_1 . If σ_1 is positive, the circle has a larger radius than the unit circle. On the other hand, if σ_1 is negative, the circle has a smaller radius than the unit circle. The circles of constant settling time, normalized to the sampling interval, are shown in Figure 13.18 with radius $e^{\sigma_1 T} = e^{-4/(T_s/T)}$. Also, $T_s/T = -4/\ln(r)$, where r is the radius of the circle of constant settling time.

The horizontal lines are lines of constant peak time. The lines are characterized by the equation $s = \sigma + j\omega_1$, where the imaginary part, $\omega_1 = \pi/T_p$, is constant. Substituting this into $z = e^{sT}$, we obtain

$$z = e^{\sigma T} e^{j\omega_1 T} = e^{\sigma T} e^{j\theta_1} \quad (13.83)$$

Equation (13.83) represents radial lines at an angle of θ_1 . If σ is negative, that section of the radial line lies inside the unit circle. If σ is positive, that section of the radial line lies outside the unit circle. The lines of constant peak time normalized to the sampling interval are shown in Figure 13.18. The angle of each radial line is $\omega_1 T = \theta_1 = \pi/(T_p/T)$, from which $T_p/T = \pi/\theta_1$.

Finally, we map the radial lines of the s -plane onto the z -plane. Remember, these radial lines are lines of constant percent overshoot on the s -plane. From Figure 13.19, these radial lines are represented by

$$\frac{\sigma}{\omega} = -\tan(\sin^{-1}\zeta) = -\frac{\zeta}{\sqrt{1-\zeta^2}} \quad (13.84)$$

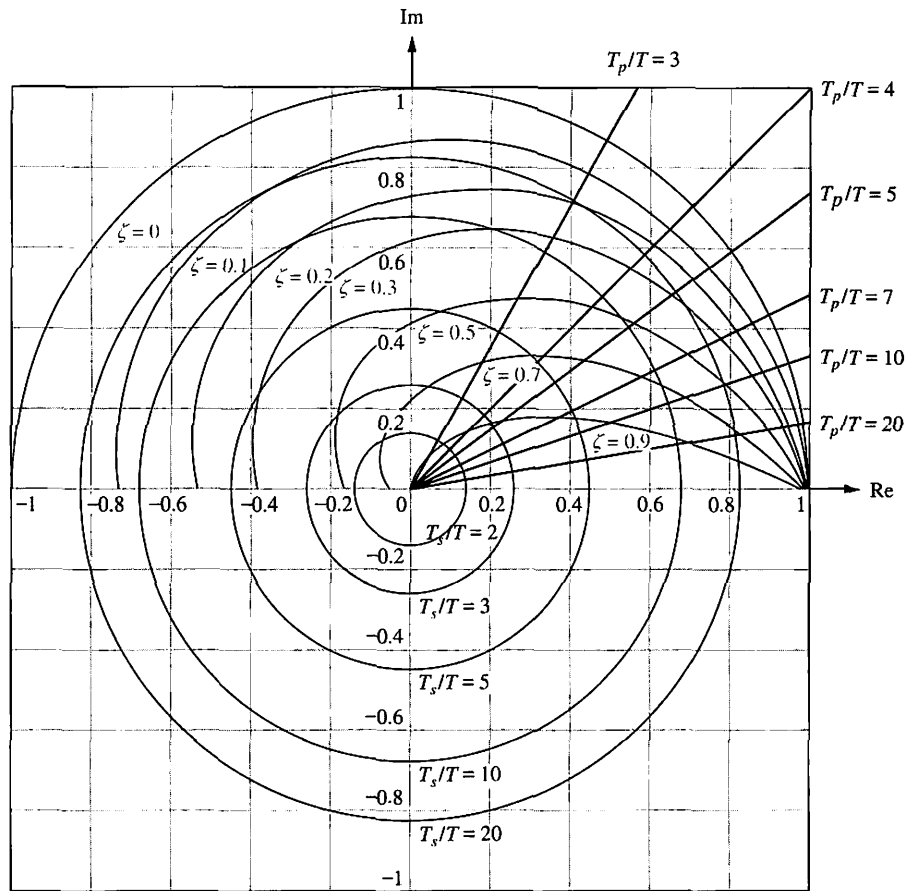


FIGURE 13.18 Constant damping ratio, normalized settling time, and normalized peak time plots on the z-plane

Hence,

$$s = \sigma + j\omega = -\omega \frac{\zeta}{\sqrt{1 - \zeta^2}} + j\omega \quad (13.85)$$

Transforming Eq. (13.85) to the z-plane yields

$$z = e^{sT} = e^{-\omega T (\zeta / \sqrt{1 - \zeta^2})} e^{j\omega T} = e^{-\omega T (\zeta / \sqrt{1 - \zeta^2})} \angle \omega T \quad (13.86)$$

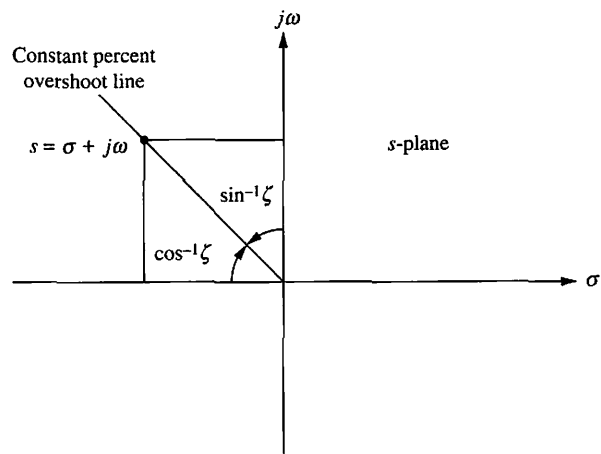


FIGURE 13.19 The s-plane sketch of constant percent overshoot line

Thus, given a desired damping ratio, ζ , Eq. (13.86) can be plotted on the z-plane through a range of ωT as shown in Figure 13.18. These curves can then be used as constant percent overshoot curves on the z-plane.

This section has set the stage for the analysis and design of transient response for digital systems. In the next section, we apply the results to digital systems using the root locus.

13.9 Gain Design on the z-Plane

In this section, we plot root loci and determine the gain required for stability as well as the gain required to meet a transient response requirement. Since the open-loop and closed-loop transfer functions for the generic digital system shown in Figure 13.20 are identical to the continuous system except for a change in variables from s to z , we can use the same rules for plotting a root locus.

However, from our previous discussion, the region of stability on the z-plane is within the unit circle and not the left half-plane. Thus, in order to determine stability, we must search for the intersection of the root locus with the unit circle rather than the imaginary axis.

In the last section, we derived the curves of constant settling time, peak time, and damping ratio. In order to design a digital system for transient response, we find the intersection of the root locus with the appropriate curves as they appear on the z-plane in Figure 13.18. Let us look at the following example.

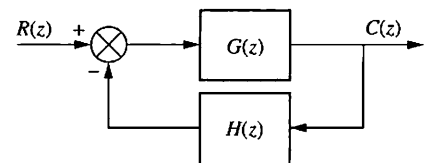


FIGURE 13.20 Generic digital feedback control system

Example 13.10

Stability Design via Root Locus

PROBLEM: Sketch the root locus for the system shown in Figure 13.21. Also, determine the range of gain, K , for stability from the root locus plot.

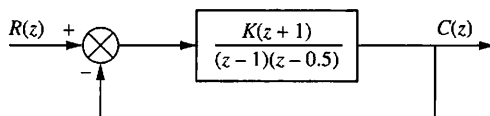


FIGURE 13.21 Digital feedback control for Example 13.10

SOLUTION: Treat the system as if z were s , and sketch the root locus. The result is shown in Figure 13.22. Using the root locus program discussed in Appendix H.2 at www.wiley.com/college/nise, search along the unit circle for 180° . Identification of the gain, K , at this point yields the range of gain for stability. Using the program, we find that the intersection of the root locus with the unit circle is $1\angle 60^\circ$. The gain at this point is 0.5. Hence, the range of gain for stability is $0 < K < 0.5$.

Students who are using MATLAB should now run ch13p7 in Appendix B. You will learn how to use MATLAB to plot a root locus on the z-plane as well as superimpose the unit circle. You will learn how to select interactively the intersection of the root locus and the unit circle to obtain the value of gain for stability. This exercise solves Example 13.10 using MATLAB.

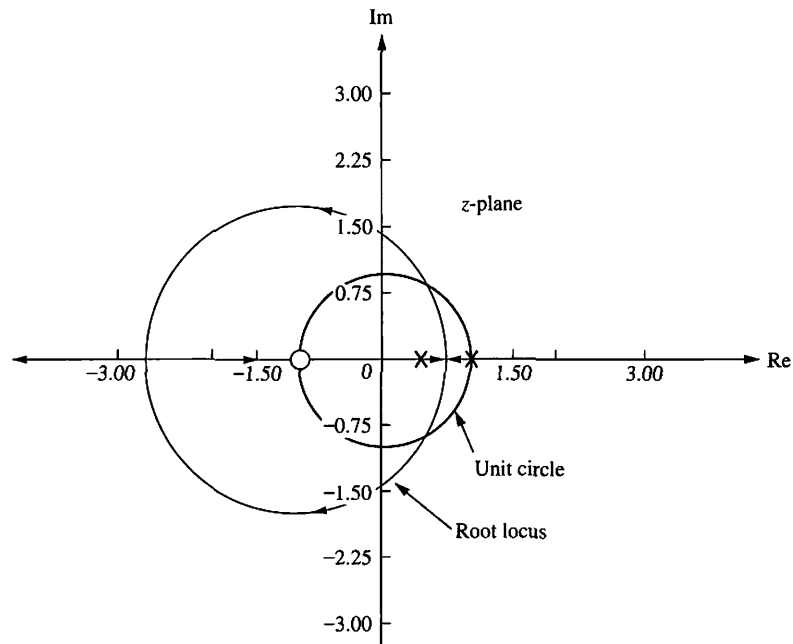


FIGURE 13.22 Root locus for the system of Figure 13.21

In the next example, we design the value of gain, K , in Figure 13.21 to meet a transient response specification. The problem is handled similarly to the analog system design, where we found the gain at the point where the root locus crossed the specified damping ratio, settling time, or peak time curve. In digital systems, these curves are as shown in Figure 13.18. In summary, then, draw the root locus of the digital system and superimpose the curves of Figure 13.18. Then find out where the root locus intersects the desired damping ratio, settling time, or peak time curve and evaluate the gain at that point. In order to simplify the calculations and obtain more accurate results, draw a radial line through the point where the root locus intersects the appropriate curve. Measure the angle of this line and use the root locus program in Appendix H.2 at www.wiley.com/college/nise to search along this radial line for the point of intersection with the root locus.

Example 13.11

Transient Response Design via Gain Adjustment

PROBLEM: For the system of Figure 13.21, find the value of gain, K , to yield a damping ratio of 0.7.

SOLUTION: Figure 13.23 shows the constant damping ratio curves superimposed over the root locus for the system as determined from the last example. Draw a radial line from the origin to the intersection of the root locus with the 0.7 damping ratio curve (a 16.62° line). The root locus program discussed in Appendix H.2 at www.wiley.com/college/nise can now be used to obtain the gain by searching along a 16.62° line for 180° , the intersection with the root locus. The results of the program show that the gain, K , is 0.0627 at $0.719 + j0.215$, the point where the 0.7 damping ratio curve intersects the root locus.

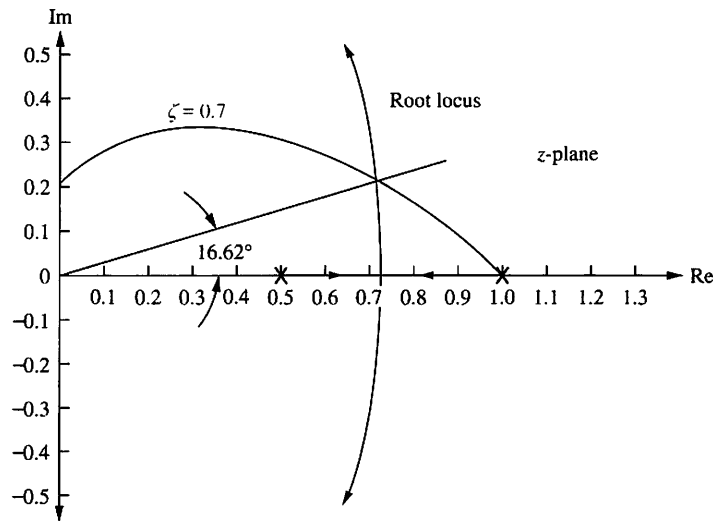


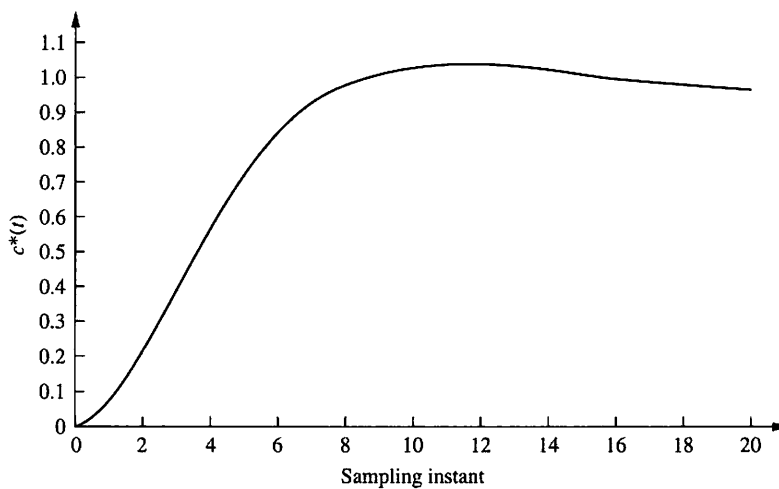
FIGURE 13.23 Root locus for the system of Figure 13.21 with constant 0.7 damping ratio curve

We can now check our design by finding the unit sampled step response of the system of Figure 13.21. Using our design, $K = 0.0627$, along with $R(z) = z/(z - 1)$, a sampled step input, we find the sampled output to be

$$C(z) = \frac{R(z)G(z)}{1 + G(z)} = \frac{0.0627z^2 + 0.0627z}{z^3 - 2.4373z^2 + 2z - 0.5627} \quad (13.87)$$

Performing the indicated division, we obtain the output valid at the sampling instants, as shown in Figure 13.24. Since the overshoot is approximately 5%, the requirement of a 0.7 damping ratio has been met. You should remember, however, that the plot is valid only at integer values of the sampling instants.

Students who are using MATLAB should now run ch13p8 in Appendix B. You will learn how to use MATLAB to plot a root locus on the z-plane as well as superimpose a grid of damping ratio curves. You will learn how to obtain the gain and a closed-loop step response of a digital system after interactively selecting the operating point on the root locus. This exercise solves Example 13.11 using MATLAB.



Note: Valid only at integer values of sampling instant

FIGURE 13.24 Sampled step response of the system of Figure 13.21 with $K = 0.0627$

Skill-Assessment Exercise 13.8

TryIt 13.3

Use MATLAB, the Control System Toolbox, and the following statements to solve Skill-Assessment Exercise 13.8.

```
Gz=zpk(-0.5,[0.25,0.75],...
1,[])
rlocus(Gz)
zgrid(0.5,[])
[K,p]=rlocfind(Gz)
```

Note: When the root locus appears, click on the intersection of the 0.5 damping ratio curve and the root locus to calculate the gain.

PROBLEM: For the system of Figure 13.20 where $H(z) = 1$ and

$$G(z) = \frac{K(z + 0.5)}{(z - 0.25)(z - 0.75)}$$

find the value of gain, K , to yield a damping ratio of 0.5.

ANSWER: $K = 0.31$

The complete solution is at www.wiley.com/college/nise.

Simulink

SL

MATLAB's Simulink provides an alternative method of simulating digital systems to obtain the time response. Students who are performing the MATLAB exercises and want to explore the added capability of Simulink should now consult Appendix C, MATLAB's Simulink Tutorial. Example C.4 in the tutorial shows how to use Simulink to simulate digital systems.

Gui Tool

GUIT

MATLAB's LTI Viewer provides another method of simulating digital systems to obtain the time response. Students who are performing the MATLAB exercises and want to explore the added capability of MATLAB's LTI Viewer should now consult Appendix E at www.wiley.com/college/nise, which contains a tutorial on the LTI Viewer as well as some examples. One of the illustrative examples, Example E.5, finds the closed-loop step response of a digital system using the LTI Viewer.

In this section, we used the root locus and gain adjustment to design the transient response of a digital system. This method suffers the same drawbacks as when it was applied to analog systems; namely, if the root locus does not intersect a desired design point, then a simple gain adjustment will not accomplish the design objective. Techniques to design compensation for digital systems can then be applied.

13.10 Cascade Compensation via the s -Plane

In previous sections of this chapter, we analyzed and designed digital systems directly in the z -domain up to and including design via gain adjustment. We are now ready to design digital compensators, such as those covered in Chapters 9 and 11. Rather than continuing on this path of design directly in the z -domain, we depart by covering analysis and design techniques that allow us to make use of previous chapters by designing on the s -plane and then transforming our s -plane design to a

digital implementation. We covered one aspect of s -plane analysis in Section 13.6, where we used a bilinear transformation to analyze stability. We now continue with s -plane analysis and design by applying it to cascade compensator design. Direct design of compensators on the z -plane is left for a dedicated course in digital control systems.

Cascade Compensation

In order to perform design in the s -plane and then convert the continuous compensator to a digital compensator, we need a bilinear transformation that will preserve, at the sampling instants, the response of the continuous compensator. The bilinear transformation covered in Section 13.6 will not meet that requirement. A bilinear transformation that can be performed with hand calculations and yields a digital transfer function whose output response at the sampling instants is approximately the same as the equivalent analog transfer function is called the *Tustin transformation*. This transformation is used to transform the continuous compensator, $G_c(s)$, to the digital compensator, $G_c(z)$. The Tustin transformation is given by⁵

$$s = \frac{2(z-1)}{T(z+1)} \quad (13.88)$$

and its inverse by

$$z = \frac{-\left(s + \frac{2}{T}\right) \left(1 + \frac{T}{2}s\right)}{\left(s - \frac{2}{T}\right) \left(1 - \frac{T}{2}s\right)} \quad (13.89)$$

As the sampling interval, T , gets smaller (higher sampling rate), the designed digital compensator's output yields a closer match to the analog compensator. If the sampling rate is not high enough, there is a discrepancy at higher frequencies between the digital and analog filters' frequency responses. Methods are available to correct the discrepancy, but they are beyond the scope of our discussion. The interested reader should investigate the topic of *prewarping*, covered in books dedicated to digital control and listed in the Bibliography at the end of this chapter.

Astrom and Wittenmark (1984) have developed a guideline for selecting the sampling interval, T . Their conclusion is that the value of T in seconds should be in the range $0.15/\omega_{\Phi_M}$ to $0.5/\omega_{\Phi_M}$, where ω_{Φ_M} is the zero dB frequency (rad/s) of the magnitude frequency response curve for the cascaded analog compensator and plant.

In the following example, we will design a compensator, $G_c(s)$, to meet the required performance specifications. We will then use the Tustin transformation to obtain the model for an equivalent digital controller. In the next section, we will show how to implement the digital controller.

⁵ See *Ogata (1987: 315–318)* for a derivation.

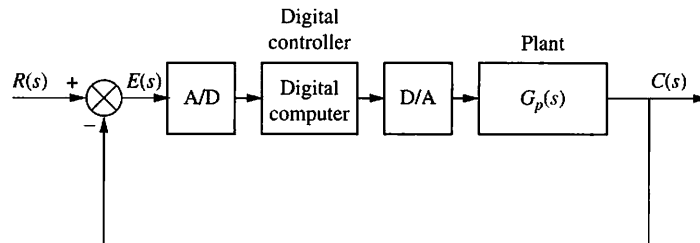
Example 13.12

Digital Cascade Compensator Design

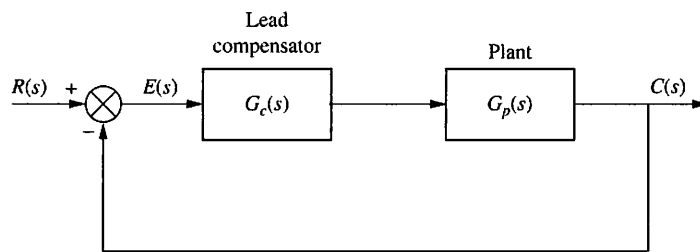
PROBLEM: For the digital control system of Figure 13.25(a), where

$$G_p(s) = \frac{1}{s(s+6)(s+10)} \quad (13.90)$$

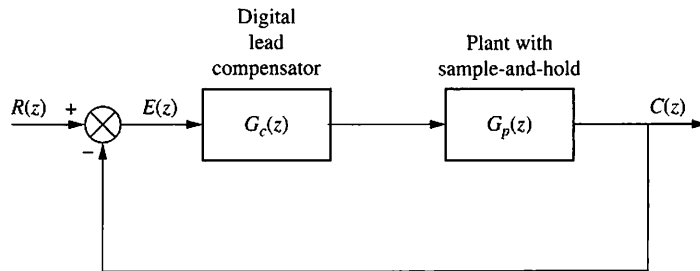
design a digital lead compensator, $G_c(z)$, as shown in Figure 13.25(c), so that the system will operate with 20% overshoot and a settling time of 1.1 seconds. Create your design in the s -domain and transform the compensator to the z -domain.



(a)



(b)



(c)

FIGURE 13.25 a. Digital control system showing the digital computer performing compensation; b. continuous system used for design; c. transformed digital system

SOLUTION: Using Figure 13.25(b), design a lead compensator using the techniques described in Chapter 9 or 11. The design was created as part of Example 9.6, where we found that the lead compensator was

$$G_c(s) = \frac{1977(s+6)}{(s+29.1)} \quad (13.91)$$

Using Eqs. (13.90) and (13.91), we find that the zero dB frequency, ω_{Φ_M} , for $G_p(s)G_c(s)$ is 5.8 rad/s. Using the guideline described by *Astrom and Wittenmark (1984)*, the lowest value of T should be in the range $0.15/\omega_{\Phi_M} = 0.026$ to $0.5/\omega_{\Phi_M} = 0.086$ second. Let us use $T = 0.01$ second.

Substituting Eq. (13.88) into Eq. (13.91) with $T = 0.01$ second yields

$$G_c(z) = \frac{1778z - 1674}{z - 0.746} \quad (13.92)$$

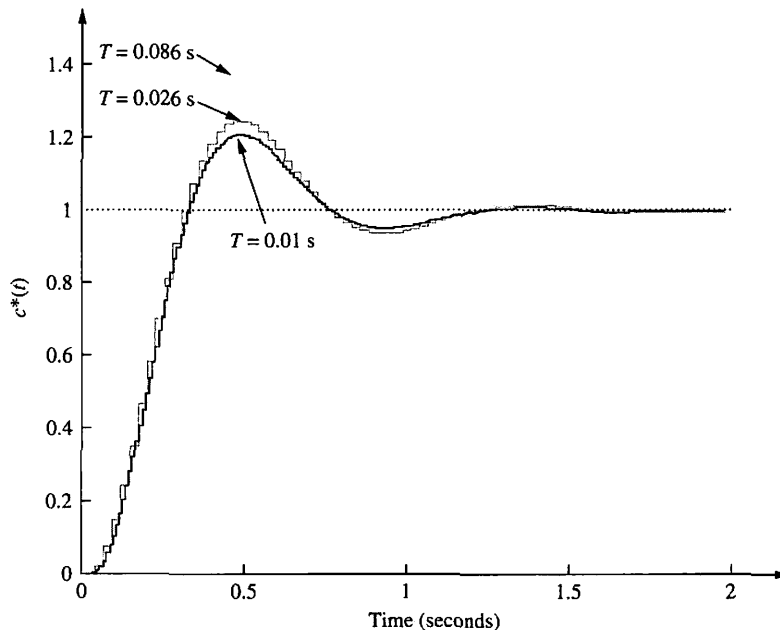
The z -transform of the plant and zero-order hold, found by the method discussed in Section 13.4 with $T = 0.01$ second, is

$$G_p(z) = \frac{(1.602 \times 10^{-7}z^2) + (6.156 \times 10^{-7}z) + (1.478 \times 10^{-7})}{z^3 - 2.847z^2 + 2.699z - 0.8521} \quad (13.93)$$

The time response in Figure 13.26 ($T = 0.01$ s) shows that the compensated closed-loop system meets the transient response requirements. The figure also shows the response for a compensator designed with sampling times at the extremes of Astrom and Wittenmark's guideline.

Students who are using MATLAB should now run ch13p9 in Appendix B. You will learn how to use MATLAB to design a digital lead compensator using the Tustin transformation. This exercise solves Example 13.12 using MATLAB.

MATLAB
ML



Note: Valid only at integer values of sampling instant

FIGURE 13.26 Closed-loop response for the compensated system of Example 13.12 showing effect of three different sampling frequencies

Skill-Assessment Exercise 13.9

WileyPLUS

WPCS

Control Solutions

PROBLEM: In Example 11.3, a lead compensator was designed for a unity feedback system whose plant was

$$G(s) = \frac{100K}{s(s+36)(s+100)}$$

The design specifications were as follows: percent overshoot = 20%, peak time 0.1 second, and $K_v = 40$. In order to meet the requirements, the design yielded $K = 1440$ and a lead compensator,

$$G_c(s) = 2.38 \frac{s + 25.3}{s + 60.2}$$

If the system is to be computer controlled, find the digital controller, $G_c(z)$.

ANSWER: $G_c(z) = 2.34 \frac{z - 0.975}{z - 0.9416}$, $T = 0.001$ second

The complete solution is at www.wiley.com/college/nise.

Now that we have learned how to design a digital cascade compensator, $G_c(z)$, the next section will teach us how to use the digital computer to implement it.

13.11 Implementing the Digital Compensator

The controller, $G_c(z)$, can be implemented directly via calculations within the digital computer in the forward path as shown in Figure 13.27. Let us now derive a numerical algorithm that the computer can use to emulate the compensator. We will find an expression for the computer's sampled output, $x^*(t)$, whose transforms are shown in Figure 13.27 as $X(z)$. We will see that this expression can be used to program the digital computer to emulate the compensator.

Consider a second-order compensator, $G_c(z)$,

$$G_c(z) = \frac{X(z)}{E(z)} = \frac{a_3z^3 + a_2z^2 + a_1z + a_0}{b_2z^2 + b_1z + b_0} \quad (13.94)$$

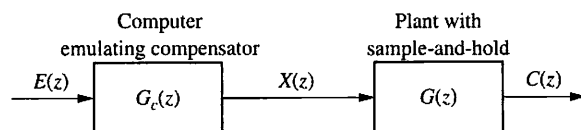
Cross-multiplying,

$$(b_2z^2 + b_1z + b_0)X(z) = (a_3z^3 + a_2z^2 + a_1z + a_0)E(z) \quad (13.95)$$

Solving for the term with the highest power of z operating on the output, $X(z)$,

$$b_2z^2X(z) = (a_3z^3 + a_2z^2 + a_1z + a_0)E(z) - (b_1z + b_0)X(z) \quad (13.96)$$

FIGURE 13.27 Block diagram showing computer emulation of a digital compensator



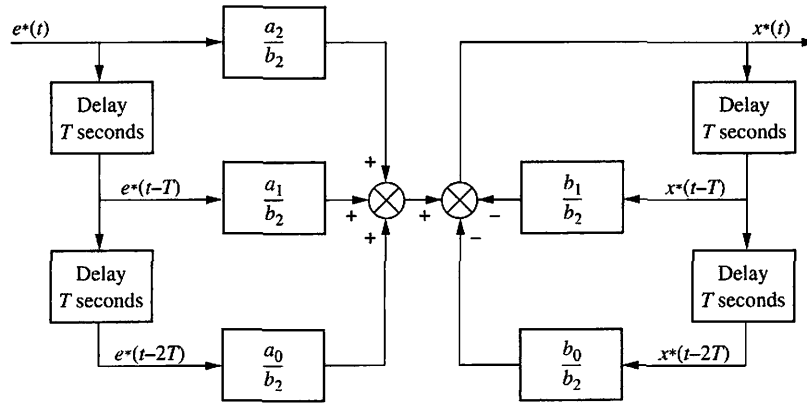


FIGURE 13.28 Flowchart for a second-order digital compensator (Reprinted with permission of John Wiley & Sons, Inc.)

Dividing by the coefficient of $X(z)$ on the left-hand side of Eq. (13.96) yields

$$X(z) = \left(\frac{a_3}{b_2} z + \frac{a_2}{b_2} + \frac{a_1}{b_2} z^{-1} + \frac{a_0}{b_2} z^{-2} \right) E(z) - \left(\frac{b_1}{b_2} z^{-1} + \frac{b_0}{b_2} z^{-2} \right) X(z) \quad (13.97)$$

Finally, taking the inverse z -transform,

$$x^*(t) = \frac{a_3}{b_2} e^*(t+T) + \frac{a_2}{b_2} e^*(t) + \frac{a_1}{b_2} e^*(t-T) + \frac{a_0}{b_2} e^*(t-2T) - \frac{b_1}{b_2} x^*(t-T) - \frac{b_0}{b_2} x^*(t-2T) \quad (13.98)$$

We can see from this equation that the present sample of the compensator output, $x^*(t)$, is a function of future ($e^*(t+T)$) present ($e^*(t)$) and past ($e^*(t-T)$ and $e^*(t-2T)$) samples of $e(t)$, along with past values of the output, $x^*(t-T)$ and $x^*(t-2T)$. Obviously, if we are to physically realize this compensator, the output sample cannot be dependent upon future values of the input. Hence, to be physically realizable, a_3 must equal zero for the future value of $e(t)$ to be zero. We conclude that the numerator of the compensator's transfer function must be of equal or lower order than the denominator in order that the compensator be physically realizable.

Now assume that a_3 does indeed equal zero. Equation (13.98) now becomes

$$x^*(t) = \frac{a_2}{b_2} e^*(t) + \frac{a_1}{b_2} e^*(t-T) + \frac{a_0}{b_2} e^*(t-2T) - \frac{b_1}{b_2} x^*(t-T) - \frac{b_0}{b_2} x^*(t-2T) \quad (13.99)$$

Hence, the output sample is a function of current and past input samples of the input as well as past samples of the output. Figure 13.28 shows the flowchart of the compensator from which a program can be written for the digital computer.⁶ The figure shows that the compensator can be implemented by storing several successive values of the input and output. The output is then formed by a weighted linear combination of these stored variables. Let us now look at a numerical example.

⁶For an excellent discussion on basic flowcharts to represent digital compensators, including the representation shown in Figure 13.28 and alternative flowcharts with half as many delays, see *Chassaing (1999, pp. 135-143)*.

Example 13.13

Digital Cascade Compensator Implementation

PROBLEM: Develop a flowchart for the digital compensator defined by Eq. (13.100).

$$G_c(z) = \frac{X(z)}{E(z)} = \frac{z + 0.5}{z^2 - 0.5z + 0.7} \quad (13.100)$$

SOLUTION: Cross-multiply and obtain

$$(z^2 - 0.5z + 0.7)X(z) = (z + 0.5)E(z) \quad (13.101)$$

Solve for the highest power of z operating on the output, $X(z)$,

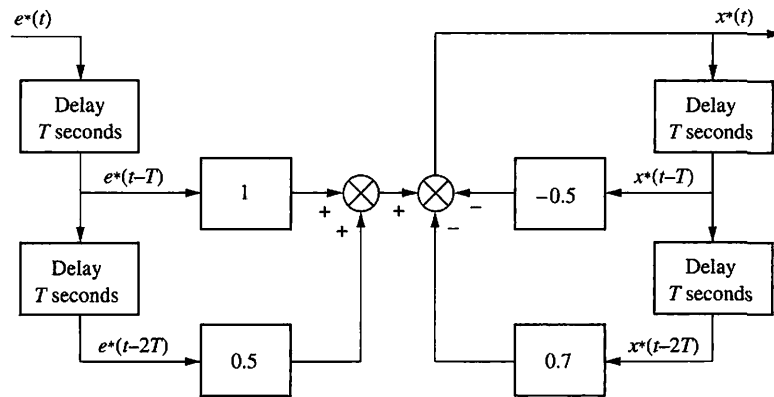
$$z^2X(z) = (z + 0.5)E(z) - (-0.5z + 0.7)X(z) \quad (13.102)$$

Solving for $X(z)$ on the left-hand side,

$$X(z) = (z^{-1} + 0.5z^{-2})E(z) - (-0.5z^{-1} + 0.7z^{-2})X(z) \quad (13.103)$$

Implementing Eq. (13.103) with the flowchart of Figure 13.29 completes the design.

FIGURE 13.29 Flowchart to implement $G_c(z) = \frac{z + 0.5}{z^2 - 0.5z + 0.7}$
(Reprinted with permission of John Wiley & Sons, Inc.)



Skill-Assessment Exercise 13.10

PROBLEM: Draw a flowchart from which the compensator

$$G_c(z) = \frac{1899z^2 - 3761z + 1861}{z^2 - 1.908z + 0.9075}$$

can be programmed if the sampling interval is 0.1 second.

ANSWER: The complete solution is at www.wiley.com/college/nise.

In this section, we learned how to implement a digital compensator. The resulting flowchart can serve as the design of a digital computer program for the computer in the loop. The design consists of delays that can be thought of as storage for each sampled value of input and output. The stored values are weighted and added. The engineer then can implement the design with a computer program.

In the next section, we will put together the concepts of this chapter as we apply the principles of digital control system design to our antenna azimuth control system.

Case Studies

Antenna Control: Transient Design via Gain

Design
D

We now demonstrate the objectives of this chapter by turning to our ongoing antenna azimuth position control system. We will show where the computer is inserted in the loop, model the system, and design the gain to meet a transient response requirement. Later, we will design a digital cascade compensator.

The computer will perform two functions in the loop. First, the computer will be used as the input device. It will receive digital signals from the keyboard in the form of commands, and digital signals from the output for closed-loop control. The keyboard will replace the input potentiometer, and an analog-to-digital (A/D) converter along with a unity gain feedback transducer will replace the output potentiometer.

Figure 13.30(a) shows the original analog system, and Figure 13.30(b) shows the system with the computer in the loop. Here the computer is receiving digital signals from two sources: (1) the input via the keyboard or other tracking commands and (2) the output via an A/D converter. The plant is receiving signals from the digital computer via a digital-to-analog (D/A) converter and the sample-and-hold.

Figure 13.30(b) shows some simplifying assumptions we have made. The power amplifier's pole is assumed to be far enough away from the motor's pole that we can represent the power amplifier as a pure gain equal to its dc gain of unity. Also, we have absorbed any preamplifier and potentiometer gain in the computer and its associated D/A converter.

PROBLEM: Design the gain for the antenna azimuth position control system shown in Figure 13.30(b) to yield a closed-loop damping ratio of 0.5. Assume a sampling interval of $T = 0.1$ second.

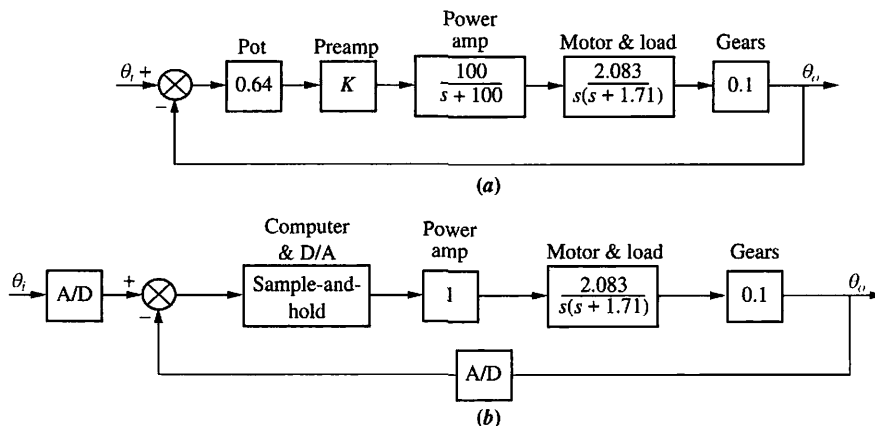


FIGURE 13.30 Antenna control system: **a.** analog implementation; **b.** digital implementation

SOLUTION: Modeling the System: Our first objective is to model the system in the z -domain. The forward transfer function, $G(s)$, which includes the sample-and-hold, power amplifier, motor and load, and the gears, is

$$G(s) = \frac{1 - e^{-Ts}}{s} \frac{0.2083}{s(s+a)} = \frac{0.2083}{a} (1 - e^{-Ts}) \frac{a}{s^2(s+a)} \tag{13.104}$$

where $a = 1.71$, and $T = 0.1$.

Since the z -transform of $(1 - e^{-Ts})$ is $(1 - z^{-1})$ and, from Example 13.6, the z -transform of $a/[s^2(s+a)]$ is

$$z \left\{ \frac{a}{s^2(s+a)} \right\} = \left[\frac{Tz}{(z-1)^2} - \frac{(1 - e^{-aT})z}{a(z-1)(z - e^{-aT})} \right] \tag{13.105}$$

the z -transform of the plant, $G(z)$, is

$$\begin{aligned} G(z) &= \frac{0.2083}{a} (1 - z^{-1})z \left\{ \frac{a}{s^2(s+a)} \right\} \\ &= \frac{0.2083}{a^2} \left[\frac{[aT - (1 - e^{-aT})]z + [(1 - e^{-aT}) - aTe^{-aT}]}{(z-1)(z - e^{-aT})} \right] \end{aligned} \tag{13.106}$$

Substituting the values for a and T , we obtain

$$G(z) = \frac{9.846 \times 10^{-4}(z + 0.945)}{(z - 1)(z - 0.843)} \tag{13.107}$$

Figure 13.31 shows the computer and plant as part of the digital feedback control system.

Designing for Transient Response: Now that the modeling in the z -domain is complete, we can begin to design the system for the required transient response. We superimpose the root locus over the constant damping ratio curves in the z -plane, as shown in Figure 13.32. A line drawn from the origin to the intersection forms an 8.58° angle. Searching along this line for 180° , we find the intersection to be $(0.915 + j0.138)$, with a loop gain, $9.846 \times 10^{-4}K$, of 0.0135. Hence, $K = 13.71$.

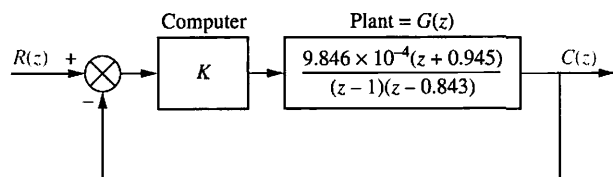
Checking the design by finding the unit sampled step response of the closed-loop system yields the plot of Figure 13.33, which exhibits 20% overshoot ($\zeta = 0.456$).

CHALLENGE: We now give you a case study to test your knowledge of this chapter's objectives: You are given the antenna azimuth position control system shown on the front endpapers, Configuration 2. Do the following:

- Convert the system into a digital system with $T = 0.1$ second. For the purposes of the conversion, assume that the potentiometers are replaced with unity gain transducers. Neglect power amplifier dynamics.
- Design the gain, K , for 16.3% overshoot.
- For your designed value of gain, find the steady-state error for a unit ramp input.
- Repeat Part **b** using MATLAB.

MATLAB
ML

FIGURE 13.31 Analog antenna azimuth position control system converted to a digital system



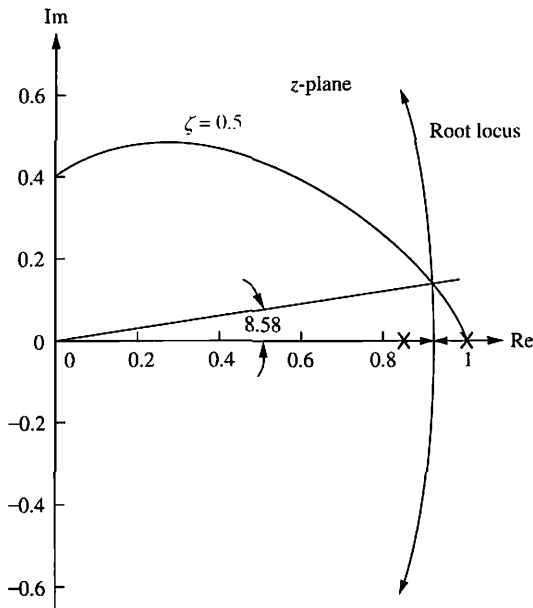
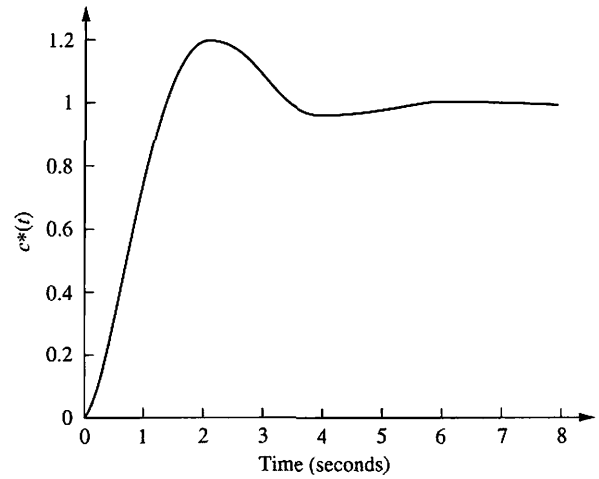


FIGURE 13.32 Root locus superimposed over constant damping ratio curve



Note: Valid only at integer values of sampling instant

FIGURE 13.33 Sampled step response of the antenna azimuth position control system

Antenna Control: Digital Cascade Compensator Design

Design
D

PROBLEM: Design a digital lead compensator to reduce the settling time by a factor of 2.5 from that obtained for the antenna azimuth control system in the previous Case Study problem in this chapter.

SOLUTION: Figure 13.34 shows a simplified block diagram of the continuous system, neglecting power amplifier dynamics and assuming that the potentiometers are replaced with unity gain transducers as previously explained.

We begin with an s -plane design. From Figure 13.33, the settling time is about 5 seconds. Thus, our design requirements are a settling time of 2 seconds and a damping ratio of 0.5. The natural frequency is $\omega_n = 4/(\zeta T_s) = 4$ rad/s. The compensated dominant poles are located at $-\zeta\omega_n \pm j\omega_n\sqrt{1 - \zeta^2} = -2 \pm j3.464$.

Designing a lead compensator zero to cancel the plant pole on the s -plane at -1.71 yields a lead compensator pole at -4 . Hence the lead compensator is given by

$$G_c(s) = \frac{s + 1.71}{s + 4} \tag{13.108}$$

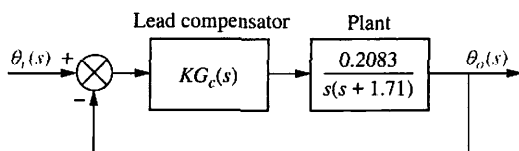


FIGURE 13.34 Simplified block diagram of antenna azimuth control system

Using root locus to evaluate the gain, K , at the design point yields $0.2083K = 16$, or $K = 76.81$.

We now select an appropriate sampling frequency as described in Section 13.10. Using the cascaded compensator,

$$KG_c(s) = \frac{76.81(s + 1.71)}{(s + 4)} \quad (13.109)$$

and plant,

$$G_p(s) = \frac{0.2083}{s(s + 1.71)} \quad (13.110)$$

the equivalent forward-path transfer function, $G_e(s) = KG_c(s)G_p(s)$, is

$$G_e(s) = \frac{16}{s(s + 4)} \quad (13.111)$$

The magnitude frequency response of Eq. (13.111) is 0 dB at 3.1 rad/s. Thus, from Section 13.10, the value of the sampling interval, T , should be in the range $0.15/\omega_{\Phi_M} = 0.05$ to $0.5/\omega_{\Phi_M} = 0.16$ second. Let us choose a smaller value, say $T = 0.025$ second.

Substituting Eq. (13.88) into Eq. (13.111), where $T = 0.025$, yields the digital compensator

$$KG_c(z) = \frac{74.72z - 71.59}{z - 0.9048} \quad (13.112)$$

In order to simulate the digital system, we calculate the z -transform of the plant in Figure 13.34 in cascade with a zero-order sample-and-hold. The z -transform of the sampled plant is evaluated by the method discussed in Section 13.4 using $T = 0.025$. The result is

$$G_p(z) = \frac{6.418 \times 10^{-5}z + 6.327 \times 10^{-5}}{z^2 - 1.958z + 0.9582} \quad (13.113)$$

The step response in Figure 13.35 shows approximately 20% overshoot and a settling time of 2.1 seconds for the closed-loop digital system.

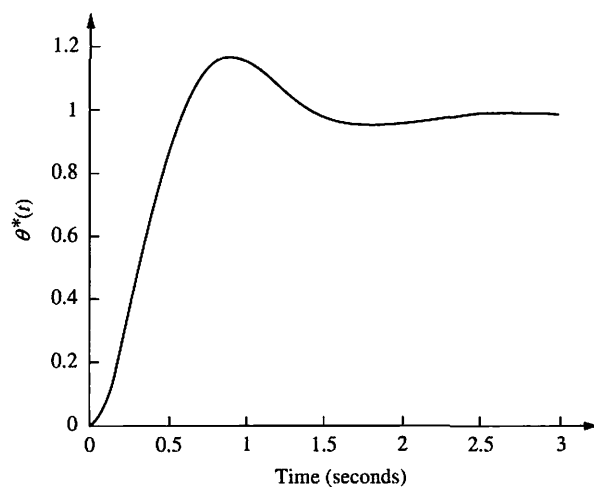


FIGURE 13.35 Closed-loop digital step response for antenna control system with a lead compensator

Note: Valid only at integer values of sampling instant

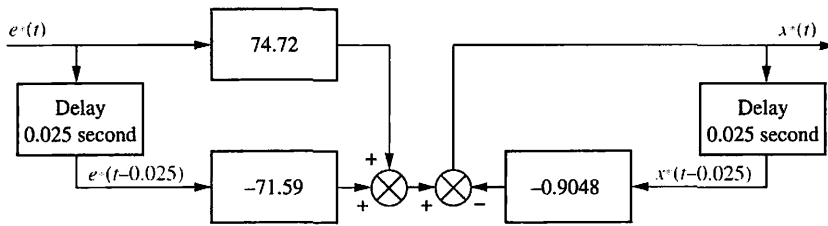


FIGURE 13.36 Flowchart for digital lead compensator. (Reprinted with permission of John Wiley & Sons, Inc.)

We conclude the design by obtaining a flowchart for the digital compensator. Using Eq. (13.112), where we define $KG_c(z) = X(z)/E(z)$, and cross-multiplying yields

$$(z - 0.9048)X(z) = (74.72z - 71.59)E(z) \quad (13.114)$$

Solving for the highest power of z operating on $X(z)$,

$$zX(z) = (74.72z - 71.59)E(z) + 0.9048X(z) \quad (13.115)$$

Solving for $X(z)$,

$$X(z) = (74.72 - 71.59z^{-1})E(z) + 0.9048z^{-1}X(z) \quad (13.116)$$

Implementing Eq. (13.116) as a flowchart yields Figure 13.36.

CHALLENGE: You are now given a case study to test your knowledge of this chapter's objectives. You are given the antenna azimuth position control system shown on the front endpapers, Configuration 2. Replace the potentiometers with unity gain transducers, neglect power amplifier dynamics, and do the following:

- Design a digital lead compensator to yield 10% overshoot with a 1-second peak time. Design in the s -plane and use the Tustin transformation to specify and implement a digital compensator. Choose an appropriate sampling interval.
- Draw a flowchart for your digital lead compensator.
- Repeat Part **a** using MATLAB.

MATLAB
ML

Summary

In this chapter, we covered the design of digital systems using classical methods. State-space techniques were not covered. However, you are encouraged to pursue this topic in a course dedicated to sampled-data control systems.

We looked at the advantages of digital control systems. These systems can control numerous loops at reduced cost. System modifications can be implemented with software changes rather than hardware changes.

Typically, the digital computer is placed in the forward path preceding the plant. Digital-to-analog and analog-to-digital conversion is required within the system to ensure compatibility of the analog and digital signals throughout the system. The digital computer in the loop is modeled as a sample-and-hold network along with any compensation that it performs.

Throughout the chapter, we saw direct parallels to the methods used for s -plane analysis of transients, steady-state errors, and the stability of analog systems.

The parallel is made possible by the z -transform, which replaces the Laplace transform as the transform of choice for analyzing sampled-data systems. The z -transform allows us to represent sampled waveforms at the sampling instants. We can handle sampled systems as easily as continuous systems, including block diagram reduction, since both signals and systems can be represented in the z -domain and manipulated algebraically. Complex systems can be reduced to a single block through techniques that parallel those used with the s -plane. Time responses can be obtained through division of the numerator by the denominator without the partial-fraction expansion required in the s -domain.

Digital systems analysis parallels the s -plane techniques in the area of stability. The unit circle becomes the boundary of stability, replacing the imaginary axis.

We also found that the concepts of root locus and transient response are easily carried into the z -plane. The rules for sketching the root locus do not change. We can map points on the s -plane into points on the z -plane and attach transient response characteristics to the points. Evaluating a sampled-data system shows that the sampling rate, in addition to gain and load, determines the transient response.

Cascade compensators also can be designed for digital systems. One method is to first design the compensator on the s -plane or via frequency response techniques described in Chapters 9 and 11, respectively. Then the resulting design is transformed to a digital compensator using the Tustin transformation. Designing cascade compensation directly on the z -plane is an alternative method that can be used. However, these techniques are beyond the scope of this book.

This introductory control systems course is now complete. You have learned how to analyze and design linear control systems using frequency-domain and state-space techniques. This course is only a beginning. You may consider furthering your study of control systems by taking advanced courses in digital, nonlinear, and optimal control, where you will learn new techniques for analyzing and designing classes of systems not covered in this book. We hope we have whetted your appetite to continue your education in control systems engineering.

Review Questions

1. Name two functions that the digital computer can perform when used with feedback control systems.
2. Name three advantages of using digital computers in the loop.
3. Name two important considerations in analog-to-digital conversion that yield errors.
4. Of what does the block diagram model for a computer consist?
5. What is the z -transform?
6. What does the inverse z -transform of a time waveform actually yield?
7. Name two methods of finding the inverse z -transform.
8. What method for finding the inverse z -transform yields a closed-form expression for the time function?
9. What method for finding the inverse z -transform immediately yields the values of the time waveform at the sampling instants?
10. In order to find the z -transform of a $G(s)$, what must be true of the input and the output?
11. If input $R(z)$ to system $G(z)$ yields output $C(z)$, what is the nature of $c(t)$?

12. If a time waveform, $c(t)$, at the output of system $G(z)$ is plotted using the inverse z -transform, and a typical second-order response with damping ratio = 0.5 results, can we say that the system is stable?
13. What must exist in order for cascaded sampled-data systems to be represented by the product of their pulse transfer functions, $G(z)$?
14. Where is the region for stability on the z -plane?
15. What methods for finding the stability of digital systems can replace the Routh-Hurwitz criterion for analog systems?
16. To drive steady-state errors in analog systems to zero, a pole can be placed at the origin of the s -plane. Where on the z -plane should a pole be placed to drive the steady-state error of a sampled system to zero?
17. How do the rules for sketching the root locus on the z -plane differ from those for sketching the root locus on the s -plane?
18. Given a point on the z -plane, how can one determine the associated percent overshoot, settling time, and peak time?
19. Given a desired percent overshoot and settling time, how can one tell which point on the z -plane is the design point?
20. Describe how digital compensators can be designed on the s -plane.
21. What characteristic is common between a cascade compensator designed on the s -plane and the digital compensator to which it is converted?

Problems

1. Derive the z -transforms for the time functions listed below. Do not use any z -transform tables. Use the plan $f(t) \rightarrow f^*(t) \rightarrow F^*(s) \rightarrow F(z)$, followed by converting $F(z)$ into closed form making use of the fact that $1/(1-z^{-1}) = 1 + z^{-1} + z^{-2} + z^{-3} + \dots$. Assume ideal sampling. [Section: 13.3]
 - a. $e^{-at}u(t)$
 - b. $u(t)$
 - c. $t^2e^{-at}u(t)$
 - d. $\cos \omega t u(t)$
2. Repeat all parts of Problem 1 using MATLAB and MATLAB's Symbolic Math Toolbox. Symbolic Math
SM
3. For each $F(z)$, find $f(kT)$ using partial-fraction expansion. [Section: 13.3]
 - a. $F(z) = \frac{z(z+3)(z+5)}{(z-0.4)(z-0.6)(z-0.8)}$
 - b. $F(z) = \frac{(z+0.2)(z+0.4)}{(z-0.1)(z-0.5)(z-0.9)}$
 - c. $F(z) = \frac{(z+1)(z+0.3)(z+0.4)}{z(z-0.2)(z-0.5)(z-0.7)}$
4. Repeat all parts of Problem 3 using MATLAB and MATLAB's Symbolic Math Toolbox. Symbolic Math
SM
5. For each $F(z)$ in Problem 3, do the following: [Section: 13.3]
 - a. Find $f(kT)$ using the power series expansion.
 - b. Check your results against your answers from Problem 3.
6. Using partial-fraction expansion and Table 13.1, find the z -transform for each $G(s)$ shown below if $T = 0.5$ second. [Section: 13.3]
 - a. $G(s) = \frac{(s+4)}{(s+2)(s+5)}$
 - b. $G(s) = \frac{(s+1)(s+2)}{s(s+3)(s+4)}$
 - c. $G(s) = \frac{20}{(s+3)(s^2+6s+25)}$
 - d. $G(s) = \frac{15}{s(s+1)(s^2+10s+81)}$WileyPLUS
WPCS
Control Solutions

7. Repeat all parts of Problem 6 using MATLAB and MATLAB's Symbolic Math Toolbox.
8. Find $G(z) = C(z)/R(z)$ for each of the block diagrams shown in Figure P13.1 if $T = 0.3$ second. [Section: 13.4]
9. Find $T(z) = C(z)/R(z)$ for each of the systems shown in Figure P13.2. [Section: 13.5]

Symbolic Math
SM

10. Find $C(z)$ in general terms for the digital system shown in Figure P13.3. [Section: 13.5]
11. Find the closed-loop transfer function, $T(z) = C(z)/R(z)$, for the system shown in Figure P13.4. [Section: 13.5]
12. Given the system in Figure P13.5, find the range of sampling interval, T , that will keep the system stable. [Section: 13.6]

WileyPLUS
WPCS
Control Solutions

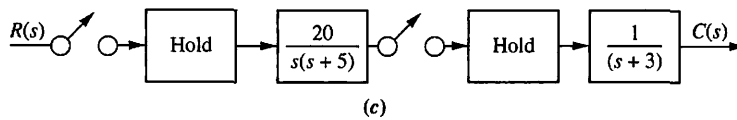
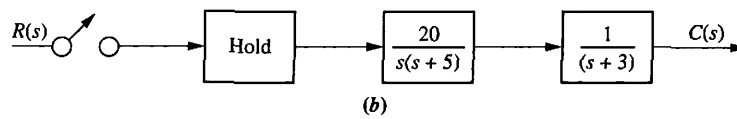
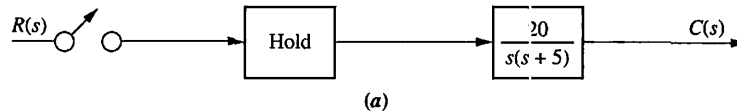


FIGURE P13.1

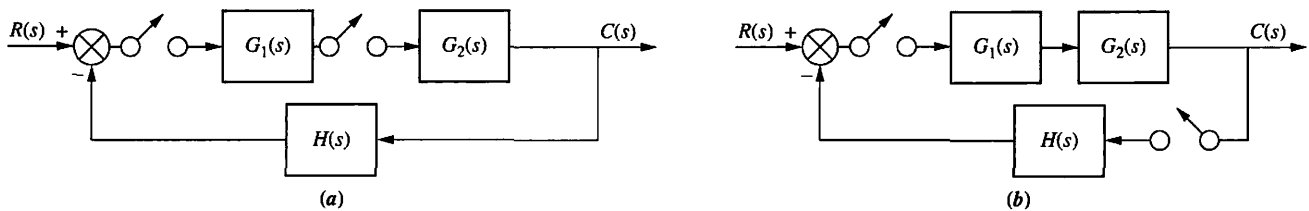


FIGURE P13.2

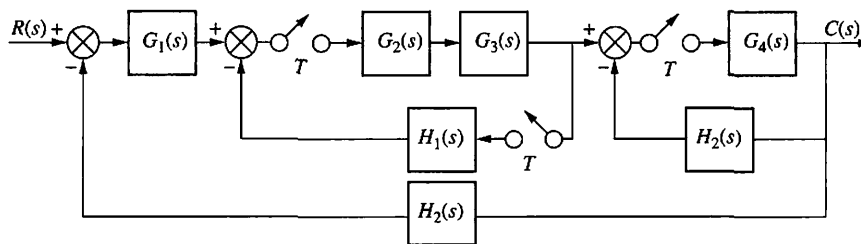


FIGURE P13.3

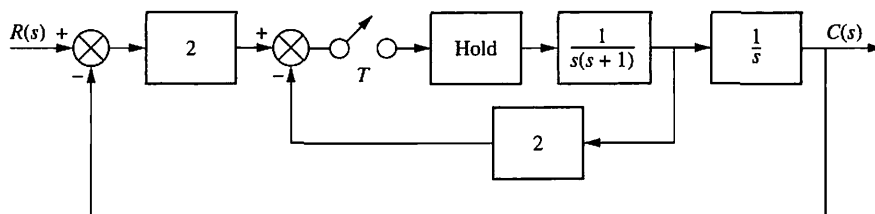


FIGURE P13.4

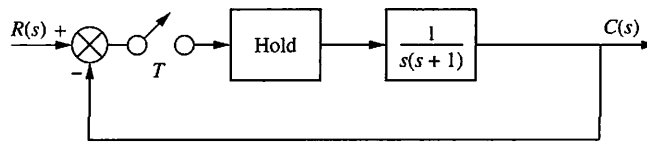


FIGURE P13.5

13. Write a MATLAB program that can be used to find the range of sampling time, T , for stability. The program will be used for systems of the type represented in Figure P13.6 and should meet the following requirements:

MATLAB ML

- MATLAB will convert $G_1(s)$ cascaded with a sample-and-hold to $G(z)$.
- The program will calculate the z -plane roots of the closed-loop system for a range of T and determine the value of T , if any, below which the system will be stable. MATLAB will display this value of T along with the z -plane poles of the closed-loop transfer function.

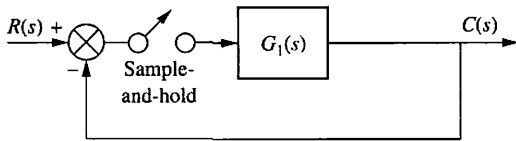


FIGURE P13.6

Test the program on

$$G_1(s) = \frac{10(s+7)}{(s+1)(s+3)(s+4)(s+5)}$$

14. Find the range of gain, K , to make the system shown in Figure P13.7 stable. [Section: 13.6]

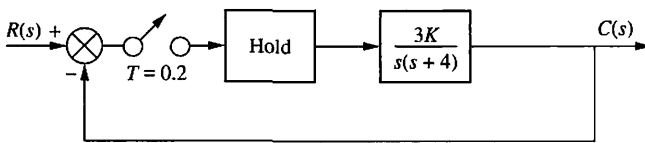
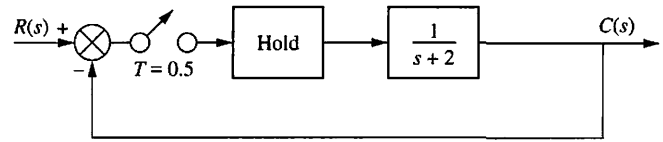


FIGURE P13.7

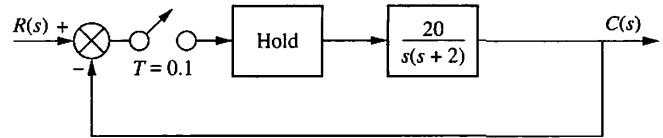
15. Find the static error constants and the steady-state error for each of the digital systems shown in Figure P13.8 if the inputs are [Section: 13.7]

WileyPLUS
WPCS
Control Solutions

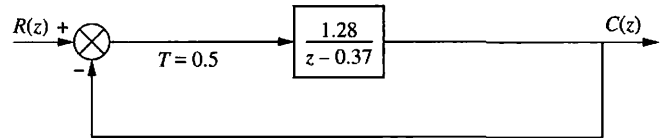
- $u(t)$
- $tu(t)$
- $\frac{1}{2}t^2u(t)$



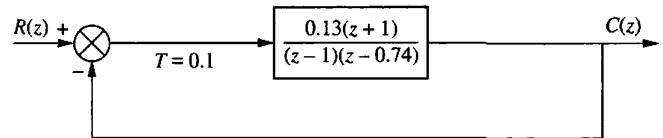
System 1



System 2



System 3



System 4

FIGURE P13.8

16. Write a MATLAB program that can be used to find K_p , K_v , and K_a for digital systems. The program will be used for systems of the type represented in Figure P13.6. Test your program for

MATLAB ML

$$G(z) = \frac{0.04406z^3 - 0.03624z^2 - 0.03284z + 0.02857}{z^4 - 3.394z^3 + 4.29z^2 - 2.393z + 0.4966}$$

where $G(z)$ is the pulse transfer function for $G_1(s)$ in cascade with the z.o.h. and $T = 0.1$ second.

17. For the digital system shown in Figure P13.6, where $G_1(s) = K/[(s+1)(s+4)]$, find the value of K to yield a 16.3% overshoot. Also find the range of K for stability. Let $T = 0.1$ second. [Section: 13.9]

18. Use Simulink to simulate the step response for the system of Problem 17. Set the value of gain, K , to that designed in Problem 17 for 16.3% overshoot.

Simulink
SL

19. Use MATLAB's LTI Viewer to determine the peak time and settling time of the closed-loop step response for System 4 in Figure P13.8

Gui Tool

GUIT

20. Write a MATLAB program that can be used to design the gain of a digital control system to meet a percent overshoot requirement. The program will be used for systems of the type represented in Figure P13.6 and meet the following requirements:

MATLAB

ML

- The user will input the desired percent overshoot.
- MATLAB will convert $G_1(s)$ cascaded with the sample-and-hold to $G(z)$.
- MATLAB will display the root locus on the z -plane along with an overlay of the percent overshoot curve.
- The user will click with the mouse at the intersection of the root locus and percent overshoot overlay and MATLAB will respond with the value of gain followed by a display of the step response of the closed-loop system.

Apply your program to Problem 17 and compare results.

21. For the digital system shown in Figure P13.6, where $G_1(s) = K/[s(s+1)]$, find the value of K to yield a peak time of 2 seconds if the sampling interval, T , is 0.1 second. Also, find the range of K for stability. [Section: 13.9]

22. For the digital system shown in Figure P13.6, where $G_1(s) = K/[s(s+1)(s+3)]$, find the value of K to yield a 20% overshoot if the sampling interval, T , is 0.1 second. Also, find the range of K for stability. [Section: 13.9]

WileyPLUS

WPCS

Control Solutions

23. For the digital system shown in Figure P13.6, where $G_1(s) = K(s+2) \div [s(s+1)(s+3)]$, find the value of K to yield a settling time of 15 seconds if the sampling interval, T , is 1 second. Also, find the range of K for stability. [Section: 13.9]

24. A PID controller was designed in Example 9.5 for a continuous system with unity feedback. The system's plant was

$$G(s) = \frac{(s+8)}{(s+3)(s+6)(s+10)}$$

The designed PID controller was

$$G_c(s) = 4.6 \frac{(s+55.92)(s+0.5)}{s}$$

Find the digital transfer function, $G_c(z)$, of the PID controller in order for the system to be computer controlled if the sampling interval, T , is 0.01 second. [Section: 13.10]

25. A continuous unity feedback system has a forward transfer function of

WileyPLUS

WPCS

Control Solutions

$$G(s) = \frac{1}{s(s+5)(s+8)}$$

The system is to be computer controlled with the following specifications:

Percent overshoot: 10%

Settling time: 2 seconds

Sampling interval: 0.01 second

Design a lead compensator for the digital system to meet the specifications. [Section: 13.10]

26. Repeat Problem 25 using MATLAB.

MATLAB

ML

DESIGN PROBLEMS

27. a. Convert the heading control for the UFSS vehicle shown on the back endpapers (*Johnson, 1980*) into a digitally controlled system.

b. Find the closed-loop pulse transfer function, $T(z)$, if $T = 0.1$ second.

c. Find the range of heading gain to keep the digital system stable.

28. A robot equipped to perform arc welding was discussed in Problem 45, Chapter 8. The robot was compensated by feeding back pressure and velocity signals as shown in Figure P8.13(b). Eliminating these feedback paths yields the block diagram shown in Figure P13.9 (*Hardy, 1967*).

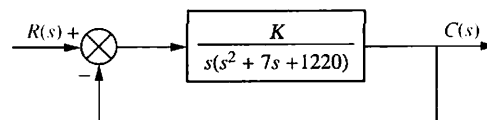


FIGURE P13.9 Simplified block diagram for robot swing motion

a. Convert the robot to a digital control system. Use a sampling time of 0.1 second.

b. Sketch the root locus.

- c. Find the range of gain, K , to keep the digital system stable.
- d. Repeat all previous parts using MATLAB
ML

29. The floppy disk drive of Problem 57, Chapter 8 is to be digitally controlled. If the analog system is as shown in Figure P13.10, do the following:

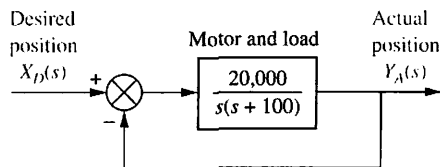


FIGURE P13.10 Simplified block diagram of a floppy disk drive

- a. Convert the disk drive to a digital system. Use a sampling time of 0.01 second.
- b. Find the range of digital controller gain to keep the system stable.
- c. Find the value of digital controller gain to yield 15% overshoot for a digital step response.
- d. Repeat all previous parts and obtain the step response for MATLAB
ML Part **c** using MATLAB.
30. Scanning probe microscopes are used to visualize samples in the sub-micron dimensional range. They typically use a silica-based probe to physically track the sample topography to create a viable image. However, these devices are very sensitive to external disturbance and vibrations. An approach called *inherent disturbance suppression* tries to alleviate the disturbance problem through the addition of a laser interferometer that is used to measure the probe-sample interaction and compensate for undesired probe movements. The technique was implemented in a tapping mode atomic force microscope measuring single DNA molecules. It was shown that for a significant range of frequencies the open-loop transmission from the probe's voltage input to probe's displacement is (Sparks, 2006)

$$G_1(s) = \frac{20000}{s}$$

Assuming the probe is digitally controlled in a loop, as shown in Figure P13.6, calculate the sampling period range that will result in a stable closed-loop system.

31. Problem 35, Chapter 9 described a two-tank system where the objective was to maintain a constant liquid level in one of the tanks via control of an inflow valve. Assume for this problem that the transfer function relating liquid-level output, $Y(s)$, to flow rate input $F_\theta(s)$, for the lower tank is (Romagnoli, 2006)

$$G(s) = \frac{Y(s)}{F_\theta(s)} = \frac{0.0187}{s^2 + 0.237s + 0.00908}$$

Assume that the system will be controlled in closed loop by means of a digital computer system with a sample period $T = 1$ second, as shown in Figure P13.6, with $G_1(s) = KG(s)$. Use the bilinear transformation and the Routh-Hurwitz method to find the range of K that will result in a stable closed-loop system.

32. Assume that the two-tank system of Problem 31 is controlled by a digital computer in the configuration of Figure P13.6, where $G_1(s) = KG(s)$. If a sampling period of $T = 1$ second is used, do the following (Romagnoli, 2006).
- a. Use MATLAB to draw the root MATLAB
ML locus.
- b. Find the value of K that will result in a stable system with a damping factor of $\zeta = 0.7$.
- c. Use the root locus of Part **b** to predict the step-response settling time, T_s , and peak time, T_p .
- d. Calculate the final value of the closed-loop system to a unit step input.
- e. Obtain the step response of Simulink
SL the system using Simulink. Verify the predictions you made in Parts **c** and **d**.
33. In Problem 48, Chapter 9, and Problem 39, Chapter 10, we considered the radial pickup position control of a DVD player. A controller was designed and placed in cascade with the plant in a unit feedback configuration to stabilize the system. The controller was given by

$$M(s) = \frac{0.5(s + 1.63)}{s(s + 0.27)}$$

and the plant by (Bittanti, 2002)

$$P(s) = \frac{0.63}{\left(1 + \frac{0.36}{305.4}s + \frac{s^2}{305.4^2}\right) \left(1 + \frac{0.04}{248.2}s + \frac{s^2}{248.2^2}\right)}$$

It is desired to replace the continuous system by an equivalent discrete system without appreciably affecting the system performance.

- Find an appropriate sampling frequency for the discretization.
- Using the chosen sampling frequency, translate the continuous compensator into a discrete compensator.
- Use Simulink to simulate the continuous and discrete systems on the same graph. Simulink
SL

Assume a unit step input. Are there significant differences in the system's performance?

34. In Problem 25, Chapter 11, we discussed an EVAD, a device that works in parallel with the human heart to help pump blood in patients with cardiac conditions. The device has a transfer function

$$G(s) = \frac{P_{ao}(s)}{E_m(s)} = \frac{1361}{s^2 + 69s + 70.85}$$

where $E_m(s)$ is the motor's armature voltage, and $P_{ao}(s)$ is the aortic blood pressure (Tasch, 1990). Using continuous techniques, a cascaded compensator is designed in a unity feedback configuration with a transfer function

$$G_c(s) = \frac{0.5(s+1)}{s+0.05}$$

Selecting to control the device using a microcontroller, a discrete equivalent has to be found for $G_c(s)$. Do the following.

- Find an appropriate sampling frequency for the discretization.
 - Translate the continuous compensator into a discrete compensator using the sampling frequency found in Part a.
 - Use Simulink to simulate the continuous and discrete systems on the same graph for a unit step input. There should be little difference between the compensated continuous and discrete systems. Simulink
SL
35. In Problem 46, Chapter 9, a steam-driven turbine-governor system was implemented by a unity feedback system with a forward-path transfer function (Khodabakhshian, 2005)

$$G(s) = \frac{K}{(s+0.08)(s+2)(s+5)}$$

- Use a sampling period of $T = 0.5$ s and find a discrete equivalent for this system.
 - Use MATLAB to draw the root locus. MATLAB
ML
 - Find the value of K that will result in a stable system with a damping factor of $\zeta = 0.7$.
 - Use the root locus found in Part a to predict the step-response settling time, T_s , and peak time, T_p .
 - Calculate the final value of the closed-loop system unit step response.
 - Obtain the step response of the system using Simulink. Simulink
SL Verify the predictions you made in Parts c and d.
36. If you have not already done so, do Problem 45 in Chapter 9. In this problem, you design a PID controller for a temperature control system. Digitize your PID design and draw a flowchart from which the PID controller can be implemented. WileyPLUS
WPCS
Control Solutions

37. Discrete time controlled systems can exhibit unique characteristics not available in continuous controllers. For example, assuming a specific input and some conditions, it is possible to design a system to achieve steady state within one single time sample without overshoot. This scheme is well known and referred to as *deadbeat control*. We illustrate deadbeat control design with a simple example. For a more comprehensive treatment see (Ogata, 1987).

Assume in Figure 13.25(a) that $G_p(s) = \frac{1}{s+1}$.

The purpose of the design will be to find a compensator, $G_c(z)$, such that for a step input the system achieves steady state within one sample. We start by translating the system into the discrete domain to obtain the equivalent of Figure 13.25(c). The pulse

transfer function, $G_p(z) = \frac{(1-e^{-T})z^{-1}}{1-e^{-T}z^{-1}}$, is found

using Eq. (13.40), since it is assumed that the compensator will be followed by a zero-order hold. In Figure 13.25(c), the closed-loop transfer function is

given by $\frac{C(z)}{R(z)} = T(z) = \frac{G_c(z)G_p(z)}{1+G_c(z)G_p(z)}$, or, solving

for the compensator, we get $G_c(z) = \frac{1}{G_p(z)} \frac{T(z)}{1-T(z)}$.

The desired system output is a unit step delayed by one unit sample. Thus, $C(z) = \frac{z}{z-1} z^{-1} = \frac{1}{z-1}$. Since

the input is a unit step, $R(z) = \frac{z}{z-1}$; the desired closed-loop transfer function is $\frac{C(z)}{R(z)} = T(z) = z^{-1}$, and the resulting compensator, found by direct substitution, is given by $G_c(z) = \frac{1}{1-e^{-T}} \frac{(z-e^{-T})}{z-1}$.

Assume now that the plant is given by $G_p(s) = \frac{1}{s}$, and a sampling period of $T=0.05$ second is used.

- Design a deadbeat compensator to reach steady state within one time sample for a step input.
- Calculate the resulting steady-state error for a unit-slope ramp input.
- Simulate your system using **Simulink** (Hint: Following **SL** Figure 13.25, the forward path will consist of the cascading of $G_c(z)$, a zero-order hold, and $G_p(s)$.) Show that the system reaches steady state after one sample. Also verify your steady-state error ramp result.

38. Given

$$G(s) = \frac{8}{s+4}$$

LabVIEW
LV

Use the LabVIEW Control Design and Simulation Module to (1) convert $G(s)$ to a digital transfer function using a sampling rate of 0.25 second; and (2) plot the step responses of the discrete and the continuous transfer functions.

39. Given

$$G(z) = \frac{K(z+0.5)}{(z-0.25)(z-0.75)}$$

LabVIEW
LV

Use the LabVIEW Control Design and Simulation Module and the MathScript RT Module to (1) obtain the value of K that will yield a damping ratio of 0.5 for the closed-loop system in Figure 13.20, where $H(z)=1$; and (2) display the step response of the closed-loop system in Figure 13.20 where $H(z)=1$. Compare your results with those of Skill-Assessment Exercise 13.8.

MATLAB
ML

PROGRESSIVE ANALYSIS AND DESIGN PROBLEMS

40. **High-speed rail pantograph.** Problem 21 in Chapter 1 discusses active control of a pantograph mechanism for high-speed rail systems (O'Connor, 1997). In

Problem 79(a), Chapter 5, you found the block diagram for the active pantograph control system. In Chapter 9, you designed a PID controller to yield a settling time of 0.3 second with zero steady-state error. Assuming that the active control system is to be computer controlled, do the following:

- Convert the PID controller designed in Problem 55, Chapter 9, to a digital controller by specifying its sampled transfer function, $G_c(z)$. Assume that the potentiometers are replaced by a keyboard, A/D converters, and unity gain transducers.
- Draw a flowchart from which the PID controller can be implemented.
- Use **MATLAB** to simulate the step response of the digital active control system. **ML**

41. **Control of HIV/AIDS.** In Chapter 11, a continuous cascaded compensator for a unity feedback system was designed for the treatment of the HIV-infected patient treated with RTIs (Craig, 2004). The transfer function of the designed compensator was

$$G_c(s) = \frac{-2 \times 10^{-4}(s^2 + 0.04s + 0.0048)}{s(s + 0.02)}$$

The linearized plant was given by

$$P(s) = \frac{Y(s)}{U_1(s)} = \frac{-520s - 10.3844}{s^3 + 2.6817s^2 + 0.11s + 0.0126}$$

The compensated system is overdamped with an approximate settling time of 100 seconds. This system must be discretized for practical reasons: (1) HIV patient cannot be monitored continuously and (2) medicine dosage cannot be adjusted continuously.

- Show that a reasonable sampling period for this system is $T = 8$ days (medicine dosage will be updated on a weekly basis).
- Use Tustin's method and $T = 8$ days to find a discrete equivalent to $G_c(S)$.
- Use **Simulink** to simulate the continuous and discrete compensated systems for a unit step input. Plot both responses on the same graph. **SL**

42. **Hybrid vehicle.** In Problem 7.69 (Figure P7.34), the block diagram of a cascade scheme for the speed control of an HEV (Preitl, 2007) was represented as a unity feedback system. **MATLAB**
ML

In that diagram the output of the system is the speed transducer's output voltage, $C(s) = K_{ss}V(s)$. In Part **b** of Problem 11.35, where a compensator was designed for this problem, we discussed the feasibility of achieving full pole-zero cancellation when we place a PI speed controller's zero, Z_c , on top of the uncompensated system's real pole, closest to the origin (located at -0.0163). Noting that perfect pole-zero cancellation may not be maintained, we studied a case, in which the PI-controller's zero changed by +20%, moving to -0.01304 . In that case, the transfer function of the plant with a PI speed controller, which has a proportional gain $= K$, was given by:

$$G(s) = \frac{K(s + 0.6)(s + 0.01304)}{s(s + 0.0163)(s + 0.5858)}$$

Assuming that $G_1(s)$ in Figure P13.6 equals the transfer function, $G(s)$, given above for the vehicle with the speed controller:

- a. Develop a MATLAB M-file that would allow you to do the following: [Hint: Refer to the M-files you developed for Problems 13 and 20 of this chapter]
 - (1) Convert $G_1(s)$ cascaded with a sample-and-hold to $G(z)$;
 - (2) Search over the range $0 < T < 5$ seconds for the largest sampling period T_{max} below which the system is stable. Calculate the z -plane roots of the closed-loop system for the whole range of the sampling time, T . Subsequently set $T = 0.75T_{max}$;
- (3) Design the gain of a digital control system to meet a percent overshoot requirement, %OS, allowing the user to input the value of the desired %OS and the value of the PI speed controller's proportional gain, K ;
- (4) Plot the step response of that digital system (in per unit, p. u., vs. time in seconds)
- b. Run the M-file you developed in Part **a** and enter the values of the desired percent overshoot, %OS=0, and the PI speed controller's proportional gain, $K=61$;
- c. Select a point in the graphics window displaying the root locus, such that all poles of the closed-loop transfer function, T_z , are inside the unit circle.
- d. Write the sampled-data transfer functions obtained, G_z and T_z , indicating the corresponding value of the sampling time, T , and all poles, r , of the closed-loop transfer function, T_z ;
- e. Plot the step response of that digital system (in per unit, p. u., vs. time in seconds) noting the following characteristics: final value, rise time, and settling time.

Cyber Exploration Laboratory

Experiment 13.1

Objective To design the gain of a digital control system to meet a transient response requirement; to simulate a digital control system to test a design; to see the effect of sampling rate upon the time response of a digital system.

Minimum Required Software Packages MATLAB, Simulink, and the Control System Toolbox

Prelab

1. Given the antenna azimuth control system shown on the front endpapers, use Configuration 2 to find the discrete transfer function of the plant. Neglect the dynamics of the power amplifier and include the preamplifier, motor, gears, and load. Assume a zero-order hold and a sampling interval of 0.01 second.

2. Using the digital plant found in Prelab 1, find the preamplifier gain required for a closed-loop digital system response with 10% overshoot and a sampling interval of 0.01 second. What is the peak time?
3. Given the antenna azimuth control system shown on the front endpapers, use Configuration 2 to find the preamplifier gain required for the continuous system to yield a closed-loop step response with 10% overshoot. Consider the open-loop system to be the preamplifier, motor, gears, and load. Neglect the dynamics of the power amplifier.

Lab

1. Verify your value of preamplifier gain found in Prelab 2 using the SISO Design Tool to generate the root locus for the digital open-loop transfer function found in Prelab 1. Use the Design Constraints capability to generate the 10% overshoot curve and place your closed-loop poles at this boundary. Obtain a plot of the root locus and the design boundary. Record the value of gain for 10% overshoot. Also, obtain a plot of the closed-loop step response using the LTI Viewer and record the values of percent overshoot and peak time. Use the same tool to find the range of gain for stability.
2. Using Simulink set up the closed-loop digital system whose plant was found in Prelab 1. Make two diagrams: one with the digital transfer function for the plant and another using the continuous transfer function for the plant preceded by a zero-order sample-and-hold. Use the same step input for both diagrams and obtain the step response of each. Measure the percent overshoot and peak time.
3. Using Simulink, set up both the digital and continuous systems calculated in Prelabs 2 and Prelab 3, respectively, to yield 10% overshoot. Build the digital system with a sample-and-hold rather than the z -transform function. Plot the step response of each system and record the percent overshoot and the peak time.
4. For one of the digital systems built in Lab 2, vary the sampling interval and record the responses for a few values of sampling interval above 0.01 second. Record sampling interval, percent overshoot, and peak time. Also, find the value of sampling interval that makes the system unstable.

Postlab

1. Make a table containing the percent overshoot, peak time, and gain for each of the following closed-loop responses: the digital system using the SISO Design Tool; the digital system using Simulink and the digital transfer functions; the digital system using Simulink and the continuous transfer functions with the zero-order sample-and-hold; and the continuous system using Simulink.
2. Using the data from Lab 4, make a table containing sampling interval, percent overshoot, and peak time. Also, state the sampling interval that makes the system unstable.
3. Compare the responses of all of the digital systems with a sampling interval of 0.01 second and the continuous system. Explain any discrepancies.
4. Compare the responses of the digital system at different sampling intervals with the continuous system. Explain the differences.
5. Draw some conclusions about the effect of sampling.

Experiment 13.2

Objective To use the various functions from the LabVIEW Control Design and Simulation Module for the analysis of digital control systems.

Minimum Required Software Packages LabVIEW with the Control Design and Simulation Module and the MathScript RT Module; MATLAB with the Control Systems Toolbox.

Prelab You are given Figure P8.28 and the parameters listed in the Prelab of Cyber Exploration Laboratory Experiment 8.2 for the open-loop NASA eight-axis ARMII (Advanced Research Manipulator II) electromechanical shoulder joint/link, actuated by an armature-controlled dc servomotor.

1. Obtain the open-loop transfer function of the shoulder joint/link, $G(s) = \frac{\theta_L(s)}{V_{ref}(s)}$, or use your calculation from Cyber Exploration Laboratory Experiment 8.2.
2. Use MATLAB and design a digital compensator to yield a closed-loop response with zero steady-state error and a damping ratio of 0.7. If you already have performed Cyber Exploration Laboratory Experiment 8.2, modify your M-file from that experiment. Test your design using MATLAB.

Lab Simulate your Prelab design using a Simulation Loop from the LabVIEW Control Design and Simulation Module. Plot the step response of two loops as follows: (1) a unity feedback with the forward path consisting of the continuous system transfer function preceded by a zero-order hold, and (2) a unity feedback with the forward path consisting of the equivalent discrete transfer function of your compensator in cascade with the open-loop plant.

Postlab Compare the results obtained with those from your prelab MATLAB program. Comment on time-performance specifications.

(Bibliography)

- Astrom, K. J., and Wittenmark, B. *Computer Controlled Systems*. Prentice Hall, Upper Saddle River, NJ, 1984.
- Bittanti, S., Dell'Orto, F., Di Carlo, A., and Savaresi, S. M. Notch Filtering and Multirate Control for Radial Tracking in High Speed DVD-Players. *IEEE Transactions on Consumer Electronics*, vol. 48, 2002, pp. 56–62.
- Boyd, M., and Yingst, J. C. PC-Based Operator Control Station Simplifies Process, Saves Time. *Chilton's I & CS*, September 1988, pp. 99–101.
- Chassaing, R. *Digital Signal Processing*. Wiley, New York, 1999.
- Craig, I. K., Xia, X., and Venter, J. W. Introducing HIV/AIDS Education into the Electrical Engineering Curriculum at the University of Pretoria. *IEEE Transactions on Education*, vol. 47, no. 1, February 2004, pp. 65–73.
- Craig, J. J. *Introduction to Robotics. Mechanics and Control*, 3d ed. Prentice Hall, Upper Saddle River, NJ, 2005.
- Hardy, H. L. Multi-Loop Servo Controls Programmed Robot. *Instruments and Control Systems*, June 1967, pp. 105–111.
- Hostetter, G. H. *Digital Control System Design*. Holt, Rinehart & Winston, New York, 1988.
- Johnson, H. et al. *Unmanned Free-Swimming Submersible (UFSS) System Description*. NRL Memorandum Report 4393. Naval Research Laboratory, Washington, D.C., 1980.

- Katz, P. *Digital Control Using Microprocessors*. Prentice Hall, Upper Saddle River, NJ, 1981.
- Khodabakhshian, A., and Golbon, N. Design of a New Load Frequency PID Controller Using QFT. *Proceedings of the 13th Mediterranean Conference on Control and Automation*, 2005, pp. 970–975.
- Kuo, B. C. *Automatic Control Systems*, 7th ed. Prentice Hall, Upper Saddle River, NJ, 1995.
- Kuo, B. C. *Digital Control Systems*. Holt, Rinehart & Winston, New York, 1980.
- Nyzen, R. J. *Analysis and Control of an Eight-Degree-of-Freedom Manipulator*, Ohio University Masters Thesis, Mechanical Engineering, Dr. Robert L. Williams II, Advisor, August 1999.
- O'Connor, D. N., Eppinger, S. D., Seering, W. P., and Wormly, D. N. Active Control of a High-Speed Pantograph. *Journal of Dynamic Systems, Measurements, and Control*, vol. 119, March 1997, pp. 1–4.
- Ogata, K. *Discrete-Time Control Systems*. Prentice Hall, Upper Saddle River, NJ, 1987.
- Phillips, C. L., and Nagle, H. T., Jr., *Digital Control System Analysis and Design*. Prentice Hall, Upper Saddle River, NJ, 1984.
- Preitl, Z., Bauer, P., and Bokor, J. A Simple Control Solution for Traction Motor Used in Hybrid Vehicles. *4th International Symposium on Applied Computational Intelligence and Informatics*. IEEE, 2007.
- Romagnoli, J. A., and Palazoglu, A. *Introduction to Process Control*. CRC Press, Boca Raton, FL, 2006.
- Smith, C. L. *Digital Computer Process Control*. Intext Educational Publishers, NY, 1972.
- Sparks, A. W., and Manalis, S. R. Atomic Force Microscopy with Inherent Disturbance Suppression for Nanostructure Imaging. *Nanotechnology*, vol. 17, 2006, pp. 1574–1579.
- Tasch, U., Koontz, J. W., Ignatoski, M. A., and Geselowitz, D. B. An Adaptive Aortic Pressure Observer for the Penn State Electric Ventricular Assist Device. *IEEE Transactions on Biomedical Engineering*, vol. 37, 1990, pp. 374–383.
- Tou, J. *Digital and Sampled-Data Control Systems*. McGraw-Hill, New York, 1959.
- Williams, R. L. II. Local Performance Optimization for a Class of Redundant Eight-Degree-of-Freedom Manipulators. *NASA Technical Paper 3417*, NASA Langley Research Center, Hampton VA, March 1994.

