# CHAPTER 11

# INTERFACING WITH THE ANALOG WORLD

# OUTLINE

- 11-1 Review of Digital Versus Analog
- 11-2 Digital-to-Analog Conversion
- **11-3** D/A-Converter Circuitry
- **11-4** DAC Specifications
- 11-5 An Integrated-Circuit DAC
- **11-6** DAC Applications
- 11-7 Troubleshooting DACs
- 11-8 Analog-to-Digital Conversion
- 11-9 Digital-Ramp ADC

- 11-10 Data Acquisition
- 11-11 Successive-Approximation ADC
- 11-12 Flash ADCs
- 11-13 Other A/D Conversion Methods
- 11-14 Sample-and-Hold Circuits
- 11-15 Multiplexing
- 11-16 Digital Storage Oscilloscope
- 11-17 Digital Signal Processing (DSP)

# **OBJECTIVES**

Upon completion of this chapter, you will be able to:

- Understand the theory of operation and the circuit limitations of several types of digital-to-analog converters (DACs).
- Read and understand the various DAC manufacturer specifications.
- Use different test procedures to troubleshoot DAC circuits.
- Compare the advantages and disadvantages among the digital-ramp analog-to-digital converter (ADC), successive-approximation ADC, and flash ADC.
- Analyze the process by which a computer, in conjunction with an ADC, digitizes an analog signal and then reconstructs that analog signal from the digital data.
- Describe the basic operation of a digital voltmeter.
- Understand the need for using sample-and-hold circuits in conjunction with ADCs.
- Describe the operation of an analog multiplexing system.
- Understand the features and basic operation of a digital storage oscilloscope.
- Understand the basic concepts of digital signal processing.

# 11-1 REVIEW OF DIGITAL VERSUS ANALOG

A **digital quantity** has a value that is specified as one of two possibilities, such as 0 or 1, LOW or HIGH, true or false, and so on. In practice, a digital quantity such as a voltage may actually have a value that is anywhere within specified ranges, and we define values within a given range to have the same digital value. For example, for TTL logic, we know that

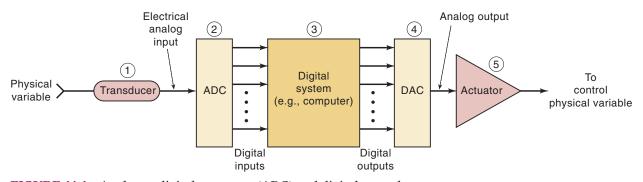
> 0 V to 0.8 V = logic 0 2 V to 5 V = logic 1

Any voltage falling in the range from 0 to 0.8 V is given the digital value 0, and any voltage in the range 2 to 5 V is assigned the digital value 1. The exact voltage values are not significant because the digital circuits respond in the same way to all voltage values within a given range.

By contrast, an **analog quantity** can take on any value over a continuous range of values and, most important, its exact value is significant. For example, the output of an analog temperature-to-voltage converter might be measured as 2.76 V, which may represent a specific temperature of 27.6°C. If the voltage were measured as something different, such as 2.34 V or 3.78 V, this would represent a completely different temperature. In other words, each possible value of an analog quantity has a different meaning. Another example of this is the output voltage from an audio amplifier into a speaker. This voltage is an analog quantity because each of its possible values produces a different response in the speaker.

Most physical variables are analog in nature and can take on any value within a continuous range of values. Examples include temperature, pressure, light intensity, audio signals, position, rotational speed, and flow rate. Digital systems perform all of their internal operations using digital circuitry and digital operations. Any information that must be input to a digital system must first be put into digital form. Similarly, the outputs from a digital system are always in digital form. When a digital system such as a computer is to be used to monitor and/or control a physical process, we must deal with the difference between the digital nature of the computer and the analog nature of the process variables. Figure 11-1 illustrates the situation. This diagram shows the five elements that are involved when a computer is monitoring and controlling a physical variable that is assumed to be analog:

- 1. Transducer. The physical variable is normally a nonelectrical quantity. A transducer is a device that converts the physical variable to an electrical variable. Some common transducers include thermistors, photocells, photodiodes, flow meters, pressure transducers, and tachometers. The electrical output of the transducer is an analog current or voltage that is proportional to the physical variable that it is monitoring. For example, the physical variable could be the temperature of water in a large tank that is being filled from cold and hot water pipes. Let's say that the water temperature varies from 80 to 150°F and that a thermistor and its associated circuitry convert this water temperature to a voltage ranging from 800 to 1500 mV. Note that the transducer's output is directly proportional to temperature such that each 1°F produces a 10-mV output. This proportionality factor was chosen for convenience.
- 2. Analog-to-digital converter (ADC). The transducer's electrical analog output serves as the analog input to the analog-to-digital converter (ADC). The ADC converts this analog input to a digital output. This digital output consists of a number of bits that represent the value of the analog input. For example, the ADC might convert the transducer's 800- to 1500-mV analog values to binary values ranging from 01010000 (80) to 10010110 (150). Note that the binary output from the ADC is proportional to the analog input voltage so that each unit of the digital output represents 10 mV.



**FIGURE 11-1** Analog-to-digital converter (ADC) and digital-to-analog converter (DAC) are used to interface a computer to the analog world so that the computer can monitor and control a physical variable.

- 3. **Computer.** The digital representation of the process variable is transmitted from the ADC to the digital computer, which stores the digital value and processes it according to a program of instructions that it is executing. The program might perform calculations or other operations on this digital representation of temperature to come up with a digital output that will eventually be used to control the temperature.
- 4. **Digital-to-analog converter (DAC).** This digital output from the computer is connected to a **digital-to-analog converter (DAC)**, which converts it to a proportional analog voltage or current. For example, the computer might produce a digital output ranging from 00000000 to 11111111, which the DAC converts to a voltage ranging from 0 to 10 V.
- 5. Actuator. The analog signal from the DAC is often connected to some device or circuit that serves as an actuator to control the physical variable. For our water temperature example, the actuator might be an electrically controlled valve that regulates the flow of hot water into the tank in accordance with the analog voltage from the DAC. The flow rate would vary in proportion to this analog voltage, with 0 V producing no flow and 10 V producing the maximum flow.

Thus, we see that ADCs and DACs function as *interfaces* between a completely digital system, such as a computer, and the analog world. This function has become increasingly more important as inexpensive microcomputers have moved into areas of process control where computer control was previously not feasible.

<b>REVIEW QUESTIONS</b>	1. What is the function of a transducer?
	2. What is the function of an ADC?
	3. What does a computer often do with the data that it receives from an ADC?
	4. What function does a DAC perform?
	5. What is the function of an actuator?

# 11-2 DIGITAL-TO-ANALOG CONVERSION

We will now begin our study of digital-to-analog (D/A) and analog-to-digital (A/D) conversion. Many A/D conversion methods utilize the D/A conversion process, so we will examine D/A conversion first.

Basically, D/A *conversion* is the process of taking a value represented in *digital* code (such as straight binary or BCD) and converting it to a voltage or current that is proportional to the digital value. Figure 11-2(a) shows the symbol for a typical four-bit D/A converter. We will not concern ourselves with the internal circuitry until later. For now, we will examine the various input/output relationships.

Notice that there is an input for a voltage reference,  $V_{ref}$ . This input is used to determine the **full-scale output** or maximum value that the D/A converter can produce. The digital inputs *D*, *C*, *B*, and *A* are usually derived from the output register of a digital system. The  $2^4 = 16$  different binary numbers represented by these four bits are listed in Figure 11-2(b). For each input number, the D/A converter output voltage is a unique value. In fact, for this case, the analog output voltage  $V_{OUT}$  is equal in volts to the binary number. It

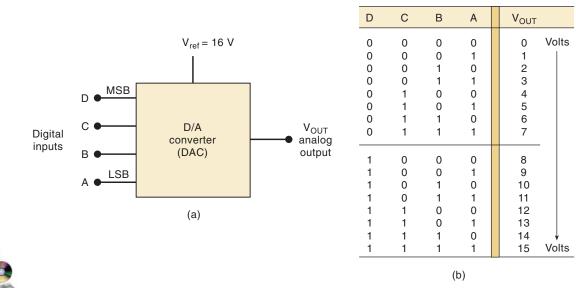


FIGURE 11-2 Four-bit DAC with voltage output.

could also have been twice the binary number or some other proportionality factor. The same idea would hold true if the D/A output were a current  $I_{\rm OUT}$ . In general,

analog output = 
$$K \times$$
 digital input (11-1)

where *K* is the proportionality factor and is a constant value for a given DAC connected to a fixed reference voltage. The analog output can, of course, be a voltage or a current. When it is a voltage, *K* will be in voltage units, and when the output is a current, *K* will be in current units. For the DAC of Figure 11-2, K = 1 V, so that

$$V_{\rm OUT} = (1 \, {\rm V}) \times {\rm digital input}$$

We can use this to calculate  $V_{OUT}$  for any value of digital input. For example, with a digital input of  $1100_2 = 12_{10}$ , we obtain

$$V_{\rm OUT} = 1 \, {\rm V} \times 12 = 12 \, {\rm V}$$

### **EXAMPLE 11-1A**

A five-bit DAC has a current output. For a digital input of 10100, an output current of 10 mA is produced. What will  $I_{OUT}$  be for a digital input of 11101?

#### Solution

The digital input  $10100_2$  is equal to decimal 20. Because  $I_{OUT} = 10$  mA for this case, the proportionality factor must be 0.5 mA. Thus, we can find  $I_{OUT}$  for any digital input such as  $11101_2 = 29_{10}$  as follows:

$$I_{\rm OUT} = (0.5 \text{ mA}) \times 29$$
  
= 14.5 mA

Remember, the proportionality factor, *K*, varies from one DAC to another and depends on the reference voltage.

**EXAMPLE 11-1B** 

What is the largest value of output voltage from an eight-bit DAC that produces 1.0 V for a digital input of 00110010?

#### **Solution**

Therefore,

 $00110010_2 = 50_{10} \\ 1.0 \text{ V} = K \times 50$ 

K = 20 mV

The largest output will occur for an input of  $11111111_2 = 255_{10}$ .

 $V_{\rm OUT}(\rm max) = 20 \ mV \times 255$  $= 5.10 \ V$ 

# **Analog Output**

The output of a DAC is technically not an analog quantity because it can take on only specific values, such as the 16 possible voltage levels for  $V_{OUT}$  in Figure 11-2, as long as  $V_{ref}$  is constant. Thus, in that sense, it is actually digital. As we will see, however, the number of different possible output values can be increased and the difference between successive values decreased by increasing the number of input bits. This will allow us to produce an output that is more and more like an analog quantity that varies continuously over a range of values. In other words, the DAC output is a "pseudo-analog" quantity. We will continue to refer to it as analog, keeping in mind that it is an approximation to a pure analog quantity.

## **Input Weights**

For the DAC of Figure 11-2, note that each digital input contributes a different amount to the analog output. This is easily seen if we examine the cases where only one input is HIGH (Table 11-1). The contributions of each digital input are *weighted* according to their position in the binary number. Thus, A, which is the LSB, has a *weight* of 1 V; B has a weight of 2 V; C has a weight of 4 V; and D, the MSB, has the largest weight, 8 V. The weights are successively doubled for each bit, beginning with the LSB. Thus, we can consider  $V_{OUT}$  to be the weighted sum of the digital inputs. For instance, to find  $V_{OUT}$  for the digital input 0111, we can add the weights of the C, B, and A bits to obtain 4 V + 2 V + 1 V = 7 V.

TA	A B	LE	1	1	-1

D	С	В	Α		V <sub>OUT</sub> (V)
0	0	0	1	$\rightarrow$	1
0	0	1	0	$\rightarrow$	2
0	1	0	0	$\rightarrow$	4
1	0	0	0	$\rightarrow$	8

## EXAMPLE 11-2

A five-bit D/A converter produces  $V_{OUT} = 0.2$  V for a digital input of 00001. Find the value of  $V_{OUT}$  for an input of 11111.

#### Solution

Obviously, 0.2 V is the weight of the LSB. Thus, the weights of the other bits must be 0.4 V, 0.8 V, 1.6 V, and 3.2 V, respectively. For a digital input of 11111, then, the value of  $V_{\text{OUT}}$  will be 3.2 V + 1.6 V + 0.8 V + 0.4 V + 0.2 V = 6.2 V.

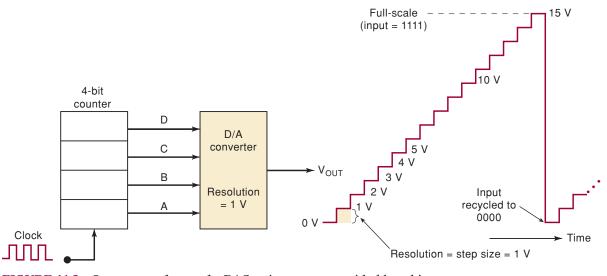
### **Resolution** (Step Size)

**Resolution** of a D/A converter is defined as the smallest change that can occur in the analog output as a result of a change in the digital input. Referring to the table in Figure 11-2, we can see that the resolution is 1 V because  $V_{OUT}$  can change by no less than 1 V when the digital input value is changed. The resolution is always equal to the weight of the LSB and is also referred to as the **step size** because it is the amount that  $V_{OUT}$  will change as the digital input value is changed from one step to the next. This is illustrated better in Figure 11-3, where the outputs from a four-bit binary counter provide the inputs to our DAC. As the counter is being continually cycled through its 16 states by the clock signal, the DAC output is a **staircase** waveform that goes up 1 V per step. When the counter is at 1111, the DAC output is at its maximum value of 15 V; this is its full-scale output. When the counter recycles to 0000, the DAC output returns to 0 V. The resolution (or step size) is the size of the jumps in the staircase waveform; in this case, each step is 1 V.

Note that the staircase has 16 levels corresponding to the 16 input states, but there are only 15 steps or jumps between the 0-V level and full-scale. In general, for an *N*-bit DAC the number of different levels will be  $2^N$ , and the number of steps will be  $2^N-1$ .

You may have already figured out that resolution (step size) is the same as the proportionality factor in the DAC input/output relationship:

analog output =  $K \times$  digital input



**FIGURE 11-3** Output waveforms of a DAC as inputs are provided by a binary counter.

A new interpretation of this expression would be that the digital input is equal to the number of steps, K is the amount of voltage (or current) per step, and the analog output is the product of the two. We now have a convenient way of calculating the value of K for the D/A:

resolution = 
$$K = \frac{A_{\rm fs}}{(2^n - 1)}$$
 (11-2)

where  $A_{\rm fs}$  is the analog full-scale output and *n* is the number of bits.

**EXAMPLE 11-3A** What is the resolution (step size) of the DAC of Example 11-2? Describe the staircase signal out of this DAC.

#### Solution

The LSB for this converter has a weight of 0.2 V. This is the resolution or step size. A staircase waveform can be generated by connecting a five-bit counter to the DAC inputs. The staircase will have 32 levels, from 0 V up to a full-scale output of 6.2 V, and 31 steps of 0.2 V each.

### **EXAMPLE 11-3B**

For the DAC of Example 11-2, determine  $V_{\text{OUT}}$  for a digital input of 10001.

### Solution

The step size is 0.2 V, which is the proportionality factor *K*. The digital input is  $10001 = 17_{10}$ . Thus, we have

 $V_{\rm OUT} = (0.2 \text{ V}) \times 17$ = 3.4 V

### **Percentage Resolution**

Although resolution can be expressed as the amount of voltage or current per step, it is also useful to express it as a percentage of the *full-scale output*. To illustrate, the DAC of Figure 11-3 has a maximum full-scale output of 15 V (when the digital input is 1111). The step size is 1 V, which gives a percentage resolution of

% resolution = 
$$\frac{\text{step size}}{\text{full scale (F.S.)}} \times 100\%$$
 (11-3)  
=  $\frac{1 \text{ V}}{15 \text{ V}} \times 100\% = 6.67\%$ 

**EXAMPLE 11-4** 

A 10-bit DAC has a step size of 10 mV. Determine the full-scale output voltage and the percentage resolution.

#### **Solution**

With 10 bits, there will be  $2^{10} - 1 = 1023$  steps of 10 mV each. The full-scale output will therefore be 10 mV  $\times$  1023 = 10.23 V, and

% resolution = 
$$\frac{10 \text{ mV}}{10.23 \text{ V}} \times 100\% \approx 0.1\%$$

Example 11-4 helps to illustrate the fact that the percentage resolution becomes smaller as the number of input bits is increased. In fact, the percentage resolution can also be calculated from

% resolution = 
$$\frac{1}{\text{total number of steps}} \times 100\%$$
 (11-4)

For an *N*-bit binary input code, the total number of steps is  $2^N - 1$ . Thus, for the previous example,

% resolution = 
$$\frac{1}{2^{10} - 1} \times 100\%$$
  
=  $\frac{1}{1023} \times 100\%$   
 $\approx 0.1\%$ 

This means that it is *only the number of bits* that determines the *percentage* resolution. Increasing the number of bits increases the number of steps to reach full scale, so that each step is a smaller part of the full-scale voltage. Most DAC manufacturers specify resolution as the number of bits.

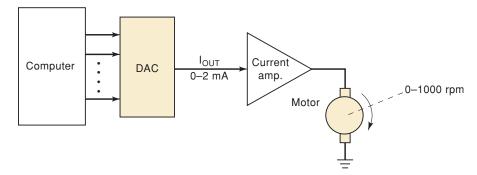
### What Does Resolution Mean?

A DAC cannot produce a continuous range of output values and so, strictly speaking, its output is not truly analog. A DAC produces a finite set of output values. In our water temperature example of Section 11-1, the computer generates a digital output to provide an analog voltage between 0 and 10 V to an electrically controlled valve. The DAC's resolution (number of bits) determines how many possible voltage values the computer can send to the valve. If a six-bit DAC is used, there will be 63 possible steps of 0.159 V between 0 and 10 V. When an eight-bit DAC is used, there will be 255 possible steps of 0.039 V between 0 and 10 V. The greater the number of bits, the finer the resolution (the smaller the step size).

The system designer must decide what resolution is needed on the basis of the required system performance. The resolution limits how close the DAC output can come to a given analog value. Generally, the cost of DACs increases with the number of bits, and so the designer will use only as many bits as necessary.

### **EXAMPLE 11-5**

Figure 11-4 shows a computer controlling the speed of a motor. The 0- to 2-mA analog current from the DAC is amplified to produce motor speeds from 0 to 1000 rpm (revolutions per minute). How many bits should be used if the computer is to be able to produce a motor speed that is within 2 rpm of the desired speed?



**FIGURE 11-4** Example 11-5.

### Solution

The motor speed will range from 0 to 1000 rpm as the DAC goes from zero to full scale. Each step in the DAC output will produce a step in the motor speed. We want the step size to be no greater than 2 rpm. Thus, we need at least 500 steps (1000/2). Now we must determine how many bits are required so that there are at least 500 steps from zero to full scale. We know that the number of steps is  $2^N - 1$ , and so we can say

$$2^N - 1 \ge 500$$

or

 $2^{N} \ge 501$ 

Since  $2^8 = 256$  and  $2^9 = 512$ , the smallest number of bits that will produce at least 500 steps is *nine*. We could use more than nine bits, but this might add to the cost of the DAC.

### **EXAMPLE 11-6**

Using nine bits, how close to 326 rpm can the motor speed be adjusted?

#### Solution

With nine bits, there will be 511 steps  $(2^9 - 1)$ . Thus, the motor speed will go up in steps of 1000 rpm/511 = 1.957 rpm. The number of steps needed to reach 326 rpm is 326/1.957 = 166.58. This is not a whole number of steps, and so we will round it to 167. The actual motor speed on the 167th step will be  $167 \times 1.957 = 326.8$  rpm. Thus, the computer must output the nine-bit binary equivalent of  $167_{10}$  to produce the desired motor speed within the resolution of the system.

In all of our examples, we have assumed that the DACs have been perfectly accurate in producing an analog output that is directly proportional to the binary input, and that the resolution is the only thing that limits how close we can come to a desired analog value. This, of course, is unrealistic because all devices contain inaccuracies. We will examine the causes and effects of DAC inaccuracy in Sections 11-3 and 11-4.

# **Bipolar DACs**

Up to this point we have assumed that the binary input to a DAC has been an unsigned number and the DAC output has been a positive voltage or current. Many DACs can also produce negative voltages by making slight changes to the analog circuitry on the output of the DAC. In this case the range of binary inputs (e.g., 00000000 to 11111111) spans a range of  $-V_{ref}$  to approximately  $+V_{ref}$ . The value of 10000000 converts to 0 V out. The output of a signed 2's complement digital system can drive this type of DAC by inverting the MSB, which converts the signed binary numbers to the proper values for the DAC as shown in Table 11-2.

### **TABLE 11-2**

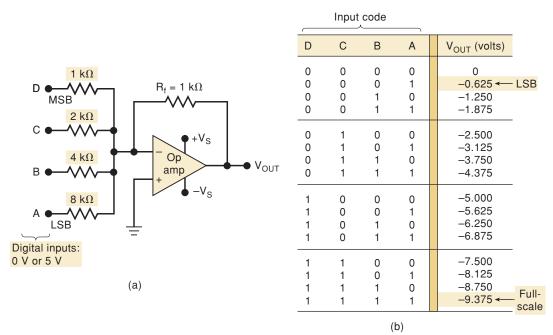
	Signed 2's Complement	DAC Inputs	DAC V <sub>out</sub>
Most positive	<mark>0</mark> 1111111	<b>1</b> 1111111	$\sim + V_{\rm ref}$
Zero	0000000	<b>1</b> 0000000	0 V
Most negative	1000000	00000000	$-V_{\rm ref}$

Other DACs may have the extra circuitry built in and accept 2's complement signed numbers as inputs. For example, suppose that we have a six-bit bipolar DAC that uses the 2's-complement system and has a resolution of 0.2 V. The binary input values range from 100000 (-32) to 011111 (+31) to produce analog outputs in the range from -6.4 to +6.2 V. There are 63 steps ( $2^6 - 1$ ) of 0.2 V between these negative and positive limits.

REVIEW QUESTIONS	1. An eight-bit DAC has an output of 3.92 mA for an input of 01100010. What are the DAC's resolution and full-scale output?
	2. What is the weight of the MSB of the DAC of question 1?
	3. What is the percentage resolution of an eight-bit DAC?
	4. How many different output voltages can a 12-bit DAC produce?
	5. For the system of Figure 11-4, how many bits should be used if the computer is to control the motor speed within 0.4 rpm?
	6. <i>True or false:</i> The percentage resolution of a DAC depends <i>only</i> on the number of bits.
	7. What is the advantage of a smaller (finer) resolution?
	11-3 D/A-CONVERTER CIRCUITRY
	There are several methods and circuits for producing the D/A operation that has

There are several methods and circuits for producing the D/A operation that has been described. We shall examine several of the basic schemes to gain an insight into the ideas used. It is not important to be familiar with all of the various circuit schemes because D/A converters are available as ICs or as encapsulated packages that do not require any circuit knowledge. Instead, it is important to know the significant performance characteristics of DACs, in general, so that they can be used intelligently. These will be covered in Section 11-4.

Figure 11-5(a) shows the basic circuit for one type of four-bit DAC. The inputs A, B, C, and D are binary inputs that are assumed to have values of either 0 or 5 V. The *operational amplifier* is employed as a summing amplifier,





**FIGURE 11-5** Simple DAC using an op-amp summing amplifier with binary-weighted resistors.

which produces the weighted sum of these input voltages. Recall that the summing amplifier multiplies each input voltage by the ratio of the feedback resistor  $R_{\rm F}$  to the corresponding input resistor  $R_{\rm IN}$ . In this circuit  $R_{\rm F} = 1 \,\mathrm{k}\Omega$ , and the input resistors range from 1 to 8 k $\Omega$ . The *D* input has  $R_{\rm IN} = 1 \,\mathrm{k}\Omega$ , so the summing amplifier passes the voltage at *D* with no attenuation. The *C* input has  $R_{\rm IN} = 2 \,\mathrm{k}\Omega$ , so that it will be attenuated by  $\frac{1}{2}$ . Similarly, the *B* input will be attenuated by  $\frac{1}{4}$ , and the *A* input by  $\frac{1}{8}$ . The amplifier output can thus be expressed as

$$V_{\rm OUT} = -(V_D + \frac{1}{2}V_C + \frac{1}{4}V_B + \frac{1}{8}V_A)$$
(11-5)

The negative sign is present because the summing amplifier is a polarityinverting amplifier, but it will not concern us here.

Clearly, the summing amplifier output is an analog voltage that represents a weighted sum of the digital inputs, as shown by the table in Figure 11-5(b). This table lists all of the possible input conditions and the resultant amplifier output voltage. The output is evaluated for any input condition by setting the appropriate inputs to either 0 or 5 V. For example, if the digital input is 1010, then  $V_D = V_B = 5$  V and  $V_C = V_A = 0$  V. Thus, using equation (11-5),

$$V_{\text{OUT}} = -(5 \text{ V} + 0 \text{ V} + \frac{1}{4} \times 5 \text{ V} + 0 \text{ V})$$
  
= -6.25 V

The resolution of this D/A converter is equal to the weighting of the LSB, which is  $\frac{1}{8} \times 5$  V = 0.625 V. As shown in the table, the analog output increases by 0.625 V as the binary input number advances one step.

(a) Determine the weight of each input bit of Figure 11-5(a).
(b) Change R<sub>F</sub> to 250 Ω and determine the full-scale output.

### Solution

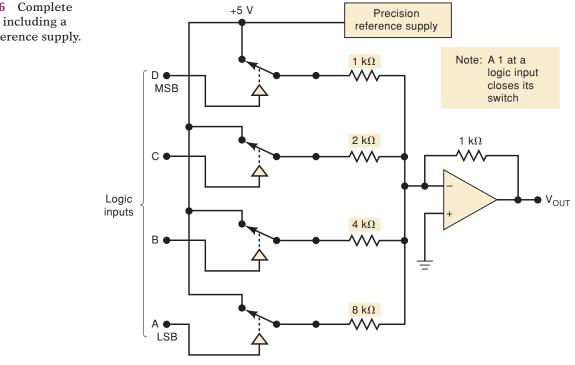
(a) The MSB passes with gain = 1, so its weight in the output is 5 V. Thus,

- $MSB \rightarrow 5 V$ 2nd MSB  $\rightarrow$  2.5 V  $3rd MSB \rightarrow 1.25 V$ 4th MSB = LSB  $\rightarrow 0.625$  V
- (b) If  $R_{\rm F}$  is reduced by a factor of 4, to 250  $\Omega$ , each input weight will be four times smaller than the values above. Thus, the full-scale output will be reduced by this same factor and becomes -9.375/4 = -2.344 V.

If we look at the input resistor values in Figure 11-5, it should come as no surprise that they are *binarily weighted*. In other words, starting with the MSB resistor, the resistor values increase by a factor of 2. This, of course, produces the desired weighting in the voltage output.

### **Conversion Accuracy**

The table in Figure 11-5(b) gives the *ideal* values of  $V_{OUT}$  for the various input cases. How close the circuit comes to producing these values depends primarily on two factors: (1) the precision of the input and feedback resistors and (2) the precision of the input voltage levels. The resistors can be made very accurate (within 0.01 percent of the desired values) by trimming, but the input voltage levels must be handled differently. It should be clear that the digital inputs cannot be taken directly from the outputs of FFs or logic gates because the output logic levels of these devices are not precise values like 0 V and 5 V but vary within given ranges. For this reason, it is necessary to add some more circuitry between each digital input and its input resistor to the summing amplifier, as shown in Figure 11-6.

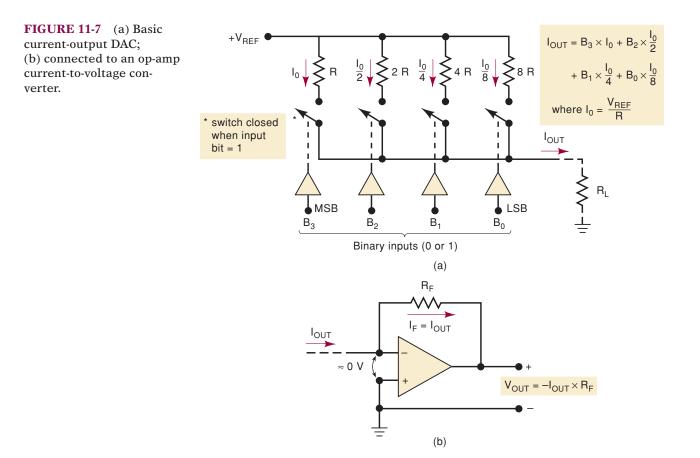




Each digital input controls a semiconductor switch such as the CMOS transmission gate we studied in Chapter 8. When the input is HIGH, the switch closes and connects a *precision reference supply* to the input resistor; when the input is LOW, the switch is open. The reference supply produces a very stable, precise voltage needed to generate an accurate analog output.

# **DAC** with Current Output

Figure 11-7(a) shows one basic scheme for generating an analog output current proportional to a binary input. The circuit shown is a four-bit DAC using binarily weighted resistors. The circuit uses four parallel current paths, each controlled by a semiconductor switch such as the CMOS transmission gate. The state of each switch is controlled by logic levels at the binary inputs. The current through each path is determined by an accurate reference voltage,  $V_{\text{REF}}$ , and a precision resistor in the path. The resistors are binarily weighted so that the various currents will be binarily weighted, and the total current,  $I_{\text{OUT}}$ , will be the sum of the individual currents. The MSB path has the smallest resistor, R; the next path has a resistor of twice the value; and so on. The output current can be made to flow through a load  $R_{\rm L}$  that is much smaller than R, so that it has no effect on the value of current. Ideally,  $R_{\rm L}$  should be a short to ground.



**EXAMPLE 11-8** 

Assume that  $V_{\text{REF}} = 10 \text{ V}$  and  $R = 10 \text{ k}\Omega$ . Determine the resolution and the full-scale output for this DAC. Assume that  $R_{\text{L}}$  is much smaller than R.

#### Solution

 $I_{\text{OUT}} = V_{\text{REF}}/R = 1$  mA. This is the weight of the MSB. The other three currents will be 0.5, 0.25, and 0.125 mA. The LSB is 0.125 mA, which is also the resolution.

The full-scale output will occur when the binary inputs are all HIGH so that each current switch is closed and

$$I_{\text{OUT}} = 1 + 0.5 + 0.25 + 0.125 = 1.875 \text{ mA}$$

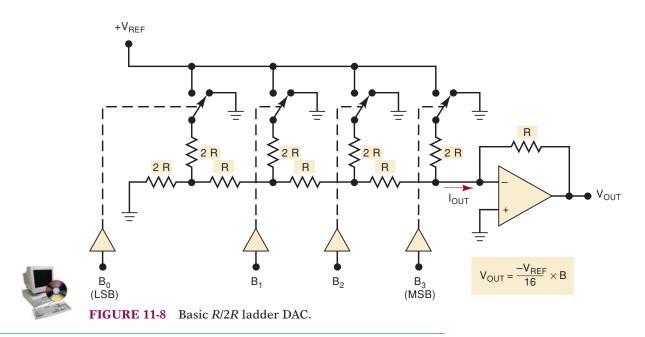
Note that the output current is proportional to  $V_{\text{REF}}$ . If  $V_{\text{REF}}$  is increased or decreased, the resolution and the full-scale output will change proportionally.

For  $I_{\text{OUT}}$  to be accurate,  $R_{\text{L}}$  should be a short to ground. One common way to accomplish this is to use an op-amp as a current-to-voltage converter, as shown in Figure 11-7(b). Here, the  $I_{\text{OUT}}$  from the DAC is connected to the op-amp's "–" input, which is virtually at ground. The op-amp negative feedback forces a current equal to  $I_{\text{OUT}}$  to flow through  $R_{\text{F}}$  to produce  $V_{\text{OUT}} = -I_{\text{OUT}} \times R_{\text{F}}$ . Thus,  $V_{\text{OUT}}$  will be an analog voltage that is proportional to the binary input to the DAC. This analog output can drive a wide range of loads without being loaded down.

### R/2R Ladder

The DAC circuits we have looked at thus far use binary-weighted resistors to produce the proper weighting of each bit. Whereas this method works in theory, it has some practical limitations. The biggest problem is the large difference in resistor values between the LSB and the MSB, especially in high-resolution DACs (i.e., many bits). For example, if the MSB resistor is  $1 \text{ k}\Omega$  in a 12-bit DAC, the LSB resistor will be over  $2 \text{ M}\Omega$ . With the current IC fabrication technology, it is very difficult to produce resistance values over a wide resistance range that maintain an accurate ratio, especially with variations in temperature.

For this reason, it is preferable to have a circuit that uses resistances that are fairly close in value. One of the most widely used DAC circuits that satisfies this requirement is the *R*/2*R* ladder network, where the resistance values span a range of only 2 to 1. One such DAC is shown in Figure 11-8.



Note how the resistors are arranged, and especially note that only two different values are used, R and 2R. The current  $I_{OUT}$  depends on the positions of the four switches, and the binary inputs  $B_3B_2B_1B_0$  control the states of the switches. This current is allowed to flow through an op-amp current-to-voltage converter to develop  $V_{OUT}$ . We will not perform a detailed analysis of this circuit here, but it can be shown that the value of  $V_{OUT}$  is given by the expression

$$V_{\rm OUT} = \frac{-V_{\rm REF}}{16} \times B \tag{11-6}$$

where *B* is the value of the binary input, which can range from 0000(0) to 1111(15).

### **EXAMPLE 11-9**

**REVIEW QUESTIONS** 

Assume that  $V_{\text{REF}} = 10 \text{ V}$  for the DAC in Figure 11-8. What are the resolution and full-scale output of this converter?

### Solution

The resolution is equal to the weight of the LSB, which we can determine by setting B = 0001 = 1 in equation (11-6):

resolution =  $\frac{-10 \text{ V} \times 1}{16}$ = -0.625 V

The full-scale output occurs for  $B = 1111 = 15_{10}$ . Again using equation (11-6),

$$full-scale = \frac{-10 \text{ V} \times 15}{16}$$
$$= -9.375 \text{ V}$$

- 1. What is the advantage of *R*/2*R* ladder DACs over those that use binaryweighted resistors?
- 2. A certain six-bit DAC uses binary-weighted resistors. If the MSB resistor is 20 k $\Omega$ , what is the LSB resistor?
- 3. What will the resolution be if the value of  $R_{\rm F}$  in Figure 11-5 is changed to 800  $\Omega$ ?
- 4. What will happen to both resolution and full-scale output when  $V_{\text{REF}}$  is increased by 20 percent?

# 11-4 DAC SPECIFICATIONS

A wide variety of DACs are currently available as ICs or as self-contained, encapsulated packages. One should be familiar with the more important manufacturers' specifications in order to evaluate a DAC for a particular application.

### Resolution

As mentioned earlier, the percentage resolution of a DAC depends solely on the number of bits. For this reason, manufacturers usually specify a DAC resolution as the number of bits. A 10-bit DAC has a finer (smaller) resolution than an eight-bit DAC.

### Accuracy

DAC manufacturers have several ways of specifying accuracy. The two most common are called **full-scale error** and **linearity error**, which are normally expressed as a percentage of the converter's full-scale output (% F.S.).

Full-scale error is the maximum deviation of the DAC's output from its expected (ideal) value, expressed as a percentage of full scale. For example, assume that the DAC of Figure 11-5 has an accuracy of  $\pm 0.01\%$  F.S. Because this converter has a full-scale output of 9.375 V, this percentage converts to

 $\pm 0.01\% \times 9.375 \text{ V} = \pm 0.9375 \text{ mV}$ 

This means that the output of this DAC can, at any time, be off by as much as 0.9375 mV from its expected value.

Linearity error is the maximum deviation in step size from the ideal step size. For example, the DAC of Figure 11-5 has an expected step size of 0.625 V. If this converter has a linearity error of  $\pm 0.01\%$  F.S., this would mean that the actual *step size* could be off by as much as 0.9375 mV.

It is important to understand that accuracy and resolution of a DAC must be compatible. It is illogical to have a resolution of, say, 1 percent and an accuracy of 0.1 percent, or vice versa. To illustrate, a DAC with a resolution of 1 percent and an F.S. output of 10 V can produce an output analog voltage within 0.1 V of any desired value, assuming perfect accuracy. It makes no sense to have a costly accuracy of 0.01% F.S. (or 1 mV) if the resolution already limits the closeness of the desired value to 0.1 V. The same can be said for having a resolution that is very small (many bits) while the accuracy is poor; it is a waste of input bits.

#### **EXAMPLE 11-10**

A certain eight-bit DAC has a full-scale output of 2 mA and a full-scale error of  $\pm 0.5\%$  F.S. What is the range of possible outputs for an input of 10000000?

### Solution

The step size is 2 mA/255 = 7.84  $\mu$ A. Since 1000000 = 128<sub>10</sub>, the ideal output should be 128  $\times$  7.84  $\mu$ A = 1004  $\mu$ A. The error can be as much as

$$\pm 0.5\% \times 2 \text{ mA} = \pm 10 \,\mu\text{A}$$

Thus, the actual output can deviate by this amount from the ideal 1004  $\mu$ A, so the actual output can be anywhere from 994 to 1014  $\mu$ A.

### **Offset Error**

Ideally, the output of a DAC will be zero volts when the binary input is all 0s. In practice, however, there will be a very small output voltage for this situation; this is called **offset error**. This offset error, if not corrected, will be added to the expected DAC output for *all* input cases. For example, let's say that a four-bit DAC has an offset error of +2 mV and a *perfect* step size of 100 mV. Table 11-3 shows the ideal and the actual DAC output for several input cases. Note that the actual output is 2 mV greater than expected; this is due to the offset error. Offset error can be negative as well as positive.

		L T.	11	1 2	
- 4	٩К				
				-5	

Input Code	Ideal Output (mV)	Actual Output (mV)
0000	0	2
0001	100	102
1000	800	802
1111	1500	1502

Many DACs have an external offset adjustment that allows you to zero the offset. This is usually accomplished by applying all 0s to the DAC input and monitoring the output while an *offset adjustment potentiometer* is adjusted until the output is as close to 0 V as required.

# Settling Time

The operating speed of a DAC is usually specified by giving its **settling time**, which is the time required for the DAC output to go from zero to full scale as the binary input is changed from all 0s to all 1s. Actually, the settling time is measured as the time for the DAC output to settle within  $\pm \frac{1}{2}$  step size (resolution) of its final value. For example, if a DAC has a resolution of 10 mV, settling time is measured as the time it takes the output to settle within 5 mV of its full-scale value.

Typical values for settling time range from 50 ns to  $10 \,\mu$ s. Generally speaking, DACs with a current output will have shorter settling times than those with voltage outputs. The main reason for this difference is the response time of the op-amp that is used as the current-to-voltage converter.

# Monotonicity

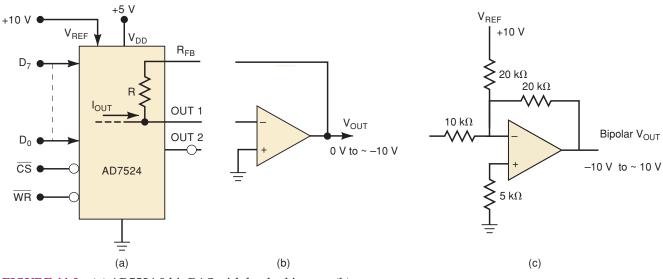
A DAC is **monotonic** if its output increases as the binary input is incremented from one value to the next. Another way to describe this is that the staircase output will have no downward steps as the binary input is incremented from zero to full scale.

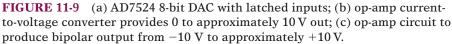
### **REVIEW QUESTIONS**

- 1. Define full-scale error.
- 2. What is settling time?
- 3. Describe offset error and its effect on a DAC output.
- 4. Why are voltage DACs generally slower than current DACs?

# 11-5 AN INTEGRATED-CIRCUIT DAC

The AD7524, a CMOS IC available from several IC manufacturers, is an eight-bit D/A converter that uses an R/2R ladder network. Its block symbol is given in Figure 11-9(a). This DAC has an eight-bit input that can be latched internally under the control of the Chip Select ( $\overline{CS}$ ) and WRITE ( $\overline{WR}$ ) inputs. When both of these control inputs are LOW, the digital data inputs  $D_7-D_0$  produce the analog output current *OUT 1* (the *OUT 2* terminal is normally grounded). When either control input goes HIGH, the digital input data are latched, and the analog output remains at the level corresponding to that latched digital data. Subsequent changes in the digital inputs will have no effect on *OUT 1* in this latched state.





The maximum settling time for the AD7524 is typically 100 ns, and its full-range accuracy is rated at  $\pm 0.2\%$  F.S. The  $V_{\text{REF}}$  can range over both negative and positive voltages from 0 to 25 V, so that analog output currents of both polarities can be produced. The output current can be converted to a voltage using an op-amp connected as in Figure 11-9(b). Note that the op-amp's feedback resistor is already on the DAC chip. The op-amp circuit shown in Figure 11-9(c) can be added to produce a bipolar output that ranges from  $-V_{\text{ref}}$  (when input = 0000000) to almost  $+V_{\text{ref}}$  (when input = 1111111).

# 11-6 DAC APPLICATIONS

DACs are used whenever the output of a digital circuit must provide an analog voltage or current to drive an analog device. Some of the most common applications are described in the following paragraphs.

# Control

The digital output from a computer can be converted to an analog control signal to adjust the speed of a motor or the temperature of a furnace, or to control almost any physical variable.

# **Automatic Testing**

Computers can be programmed to generate the analog signals (through a DAC) needed to test analog circuitry. The test circuit's analog output response will normally be converted to a digital value by an ADC and fed into the computer to be stored, displayed, and sometimes analyzed.

# Signal Reconstruction

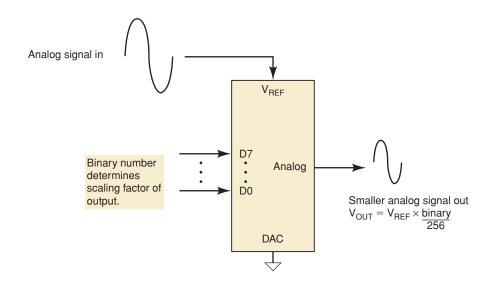
In many applications, an analog signal is **digitized**; that is, successive points on the signal are converted to their digital equivalents and stored in memory. This conversion is performed by an analog-to-digital converter (ADC). A DAC can then be used to convert the stored digitized data back to analog one point at a time—thereby reconstructing the original signal. This combination of digitizing and reconstructing is used in digital storage oscilloscopes, audio compact disk systems, and digital audio and video recording. We will look at this further after we learn about ADCs.

# A/D Conversion

Several types of ADCs use DACs as part of their circuitry, as we shall see in Section 11-8.

# **Digital Amplitude Control**

DACs can also be used to reduce the amplitude of an analog signal by connecting the analog signal to the  $V_{\text{REF}}$  input as shown in Figure 11-10. The binary input scales the signal on  $V_{\text{REF}}$ :  $V_{\text{OUT}} = V_{\text{REF}} \times \text{binary in}/2^N$ . When the maximum binary input value is applied, the output is nearly the same as the  $V_{\text{REF}}$  input. However, when a value that represents about half of the maximum (e.g., 1000 000 for a unipolar eight-bit converter) is applied to the inputs, the output is about half of  $V_{\text{REF}}$ . If  $V_{\text{REF}}$  is a signal (e.g., a sine wave) that varies within the range of the reference voltage, the output will be the same fully analog wave shape whose amplitude depends on the digital number applied to the DAC. In this way a digital system can control things such as the volume of an audio system or the amplitude of a function generator.



# Serial DACs

**FIGURE 11-10** A DAC used to control the amplitude of an analog

signal.

Many of these DAC applications involve a microprocessor. The main problem with using the parallel-data DACs that have been described thus far is that they occupy so many port bits of the microcomputer. In cases where speed of data transfer is of little concern, a microprocessor can output the digital value to a DAC over a serial interface. Serial DACs are now readily available with a built-in serial in/parallel out shift register. Many of these devices have more than one DAC on the same chip. The digital data, along with a code that specifies which DAC you want, are sent to the chip, one bit at a time. As each bit is presented on the DAC input, a pulse is applied to the serial clock input to shift the bit in. After the proper number of clock pulses, the data value is latched and converted to its analog value.

# 11-7 TROUBLESHOOTING DACs

DACs are both digital and analog. Logic probes and pulsers can be used on the digital inputs, but a meter or an oscilloscope must be used on the analog output. There are basically two ways to test a DAC's operation: a *static accuracy test* and a *staircase test*.

The static test involves setting the binary input to a fixed value and measuring the analog output with a high-accuracy meter. This test is used to check that the output value falls within the expected range consistent with the DAC's specified accuracy. If it does not, there can be several possible causes. Here are some of them:

- Drift in the DAC's internal component values (e.g., resistor values) caused by temperature, aging, or some other factors. This condition can easily produce output values outside the expected accuracy range.
- Open connections or shorts in any of the binary inputs. This could either prevent an input from adding its weight to the analog output or cause its weight to be permanently present in the output. This situation is especially hard to detect when the fault is in the less significant inputs.
- A faulty voltage reference. Because the analog output depends directly on  $V_{\text{REF}}$ , this could produce results that are way off. For DACs that use external reference sources, the reference voltage can be checked easily with a digital voltmeter. Many DACs have internal reference voltages that cannot be checked, except on some DACs that bring the reference voltage out to a pin of the IC.
- Excessive offset error caused by component aging or temperature. This would produce outputs that are off by a fixed amount. If the DAC has an external offset adjustment capability, this type of error can initially be zeroed out, but changes in operating temperature can cause the offset error to reappear.

The staircase test is used to check the monotonicity of the DAC; that is, it checks to see that the output increases step by step as the binary input is incremented as in Figure 11-3. The steps on the staircase must be of the same size, and there should be no missing steps or downward steps until full scale is reached. This test can help detect internal or external faults that cause an input to have either no contribution or a permanent contribution to the analog output. The following example will illustrate.

### **EXAMPLE 11-11**

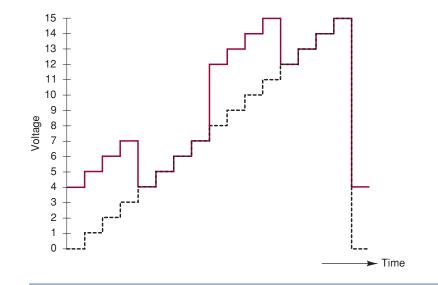
How would the staircase waveform appear if the *C* input to the DAC of Figure 11-3 is open? Assume that the DAC inputs are TTL-compatible.

#### Solution

An open connection at C will be interpreted as a constant logic 1 by the DAC. Thus, this will contribute a constant 4 V to the DAC output so that the DAC output waveform will appear as shown in Figure 11-11. The dotted lines are the staircase as it would appear if the DAC were working correctly. Note that the faulty output waveform matches the correct one during those times when the bit C input would normally be HIGH.

**FIGURE 11-11** 

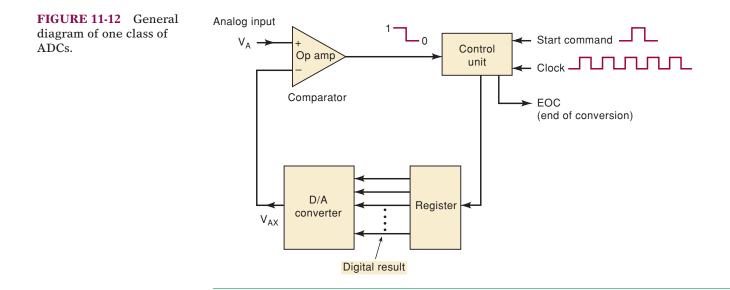
Example 11-11.



# 11-8 ANALOG-TO-DIGITAL CONVERSION

An analog-to-digital converter takes an analog input voltage and, after a certain amount of time, produces a digital output code that represents the analog input. The A/D conversion process is generally more complex and timeconsuming than the D/A process, and many different methods have been developed and used. We shall examine several of these methods in detail, even though it may never be necessary to design or construct ADCs (they are available as completely packaged units). However, the techniques that are used provide an insight into what factors determine an ADC's performance.

Several important types of ADCs utilize a DAC as part of their circuitry. Figure 11-12 is a general block diagram for this class of ADC. The timing for the operation is provided by the input clock signal. The control unit contains the logic circuitry for generating the proper sequence of operations in response to the START COMMAND, which initiates the conversion process. The op-amp comparator has two *analog* inputs and a *digital* output that switches states, depending on which analog input is greater.



The basic operation of ADCs of this type consists of the following steps:

- 1. The START COMMAND pulse initiates the operation.
- 2. At a rate determined by the clock, the control unit continually modifies the binary number that is stored in the register.
- 3. The binary number in the register is converted to an analog voltage,  $V_{AX}$ , by the DAC.
- 4. The comparator compares  $V_{AX}$  with the analog input  $V_A$ . As long as  $V_{AX} < V_A$ , the comparator output stays HIGH. When  $V_{AX}$  exceeds  $V_A$  by at least an amount equal to  $V_T$  (threshold voltage), the comparator output goes LOW and stops the process of modifying the register number. At this point,  $V_{AX}$  is a close approximation to  $V_A$ . The digital number in the register, which is the digital equivalent of  $V_{AX}$ , is also the approximate digital equivalent of  $V_A$ , within the resolution and accuracy of the system.
- 5. The control logic activates the end-of-conversion signal, *EOC*, when the conversion is complete.

The several variations of this A/D conversion scheme differ mainly in the manner in which the control section continually modifies the numbers in the register. Otherwise, the basic idea is the same, with the register holding the required digital output when the conversion process is complete.

### **REVIEW QUESTIONS**

1. What is the function of the comparator in the ADC?

- 2. Where is the approximate digital equivalent of  $V_A$  when the conversion is complete?
- 3. What is the function of the *EOC* signal?

# 11-9 DIGITAL-RAMP ADC

One of the simplest versions of the general ADC of Figure 11-12 uses a binary counter as the register and allows the clock to increment the counter one step at a time until  $V_{AX} \ge V_A$ . It is called a **digital-ramp ADC** because the waveform at  $V_{AX}$  is a step-by-step ramp (actually a staircase) like the one shown in Figure 11-3. It is also referred to as a *counter-type* ADC.

Figure 11-13 is the diagram for a digital-ramp ADC. It contains a counter, a DAC, an analog comparator, and a control AND gate. The comparator output serves as the active-LOW end-of-conversion signal  $\overline{EOC}$ . If we assume that  $V_A$ , the analog voltage to be converted, is positive, the operation proceeds as follows:

- 1. A START pulse is applied to reset the counter to 0. The HIGH at START also inhibits clock pulses from passing through the AND gate into the counter.
- 2. With all 0s at its input, the DAC's output will be  $V_{AX} = 0$  V.
- 3. Because  $V_A > V_{AX}$ , the comparator output,  $\overline{EOC}$ , will be HIGH.
- 4. When START returns LOW, the AND gate is enabled and clock pulses get through to the counter.

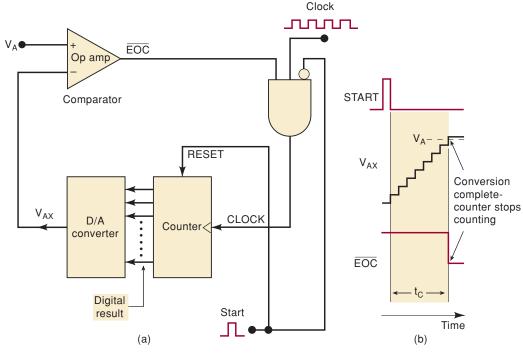


FIGURE 11-13 Digital-ramp ADC.

- 5. As the counter advances, the DAC output,  $V_{AX}$ , increases one step at a time, as shown in Figure 11-13(b).
- 6. This process continues until  $V_{AX}$  reaches a step that exceeds  $V_A$  by an amount equal to or greater than  $V_T$  (typically 10 to 100  $\mu$ V). At this point,  $\overline{EOC}$  will go LOW and inhibit the flow of pulses into the counter, and the counter will stop counting.
- 7. The conversion process is now complete, as signaled by the HIGH-to-LOW transition at  $\overline{EOC}$ , and the contents of the counter are the digital representation of  $V_A$ .
- 8. The counter will hold the digital value until the next START pulse initiates a new conversion.

#### **EXAMPLE 11-12**

Assume the following values for the ADC of Figure 11-13: clock frequency = 1 MHz;  $V_{\rm T}$  = 0.1 mV; DAC has F.S. output = 10.23 V and a 10-bit input. Determine the following values.

- (a) The digital equivalent obtained for  $V_A = 3.728 \text{ V}$
- (b) The conversion time
- (c) The resolution of this converter

### Solution

(a) The DAC has a 10-bit input and a 10.23-V F.S. output. Thus, the number of total possible steps is  $2^{10} - 1 = 1023$ , and so the step size is

$$\frac{10.23 \text{ V}}{1023} = 10 \text{ mV}$$

This means that  $V_{AX}$  increases in steps of 10 mV as the counter counts up from 0. Because  $V_A = 3.728$  V and  $V_T = 0.1$  mV,  $V_{AX}$  must reach 3.7281 V or more before the comparator switches LOW. This will require

$$\frac{3.7281 \text{ V}}{10 \text{ mV}} = 372.81 = 373 \text{ steps}$$

At the end of the conversion, then, the counter will hold the binary equivalent of 373, which is 0101110101. This is the desired digital equivalent of  $V_A = 3.728$  V, as produced by this ADC.

- (b) Three hundred seventy-three steps were required to complete the conversion. Thus, 373 clock pulses occurred at the rate of one per microsecond. This gives a total conversion time of 373  $\mu$ s.
- (c) The resolution of this converter is equal to the step size of the DAC, which is 10 mV. Expressed as a percentage, it is  $1/1023 \times 100\% \approx 0.1\%$ .

### **EXAMPLE 11-13**

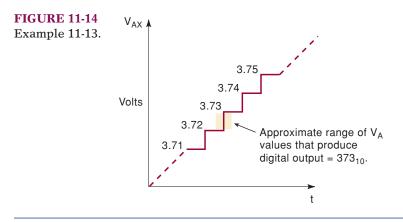
For the same ADC of Example 11-12, determine the approximate range of analog input voltages that will produce the same digital result of  $0101110101_2 = 373_{10}$ .

### Solution

Table 11-4 shows the ideal DAC output voltage,  $V_{AX}$ , for several of the steps on and around the 373rd. If  $V_A$  is slightly smaller than 3.72 V (by an amount  $< V_T$ ), then  $\overline{EOC}$  won't go LOW when  $V_{AX}$  reaches the 3.72-V step, but it will go LOW on the 3.73-V step. If  $V_A$  is slightly smaller than 3.73 V (by an amount  $< V_T$ ), then  $\overline{EOC}$  won't go LOW until  $V_{AX}$  reaches the 3.74-V step. Thus, as long as  $V_A$  is between approximately 3.72 and 3.73 V,  $\overline{EOC}$  will go LOW when  $V_{AX}$  reaches the 3.73-V step. The exact range of  $V_A$  values is

 $3.72 \text{ V} - V_{\text{T}}$  to  $3.73 \text{ V} - V_{\text{T}}$ 

but because  $V_{\rm T}$  is so small, we can simply say that the range is approximately 3.72 to 3.73 V—a range equal to 10 mV, the DAC's resolution. This is illustrated in Figure 11-14.



### A/D Resolution and Accuracy

It is very important to understand the errors associated with making any kind of measurements. An unavoidable source of error in the digital-ramp

Step	$V_{AX}(V)$
371	3.71
372	3.72
373	3.73
374	3.74
375	3.75

method is that the step size or resolution of the internal DAC is the smallest unit of measure. Imagine trying to measure basketball players' heights by standing them next to a staircase with 12-in steps and assigning them the height of the first step higher than their head. Anyone over 6 ft would be measured as 7 ft tall! Likewise, the output voltage  $V_{AX}$  is a staircase waveform that goes up in discrete steps until it exceeds the input voltage,  $V_A$ . By making the step size smaller, we can reduce the potential error, but there will always be a difference between the actual (analog) quantity and the digital value assigned to it. This is called **quantization error**. Thus,  $V_{AX}$  is an approximation to the value of  $V_A$ , and the best we can expect is that  $V_{AX}$  is within 10 mV of  $V_A$  if the resolution (step size) is 10 mV. This quantization error, which can be reduced by increasing the number of bits in the counter and the DAC, is sometimes specified as an error of +1 LSB, indicating that the result could be off by as much as the weight of the LSB.

A more common practice is to make the quantization error symmetrical around an integer multiple of the resolution to make the quantization error  $\pm \frac{1}{2}$  LSB. This is done by making sure the output changes at  $\frac{1}{2}$  resolution unit below and above the nominal input voltage. For example, if the resolution is 10 mV, then the A/D output will ideally switch from 0 to 1 at 5 mV and from 1 to 2 at 15 mV. The nominal value (10 mV), which is represented by the digital value of 1, is ideally always within 5 mV ( $\frac{1}{2}$  LSB) of the actual input voltage. Problem 11-28 explores a method to accomplish this. In any case, there is a small range of input voltages that will produce the same digital output.

The accuracy specification reflects the fact that the output of every ADC does not switch from one binary value to the next at the exact prescribed input voltage. Some change at slightly higher voltage than expected, and some at slightly lower. The inaccuracy and inconsistency is due to imperfect components such as precision resistors, comparators, current switches, and so on. Accuracy can be expressed as % full-scale, just as for the DAC, but it is more commonly specified as  $\pm n$  LSB, where *n* is a fractional value or 1. For example, if the accuracy is specified as  $\pm \frac{1}{4}$  LSB with a resolution of 10 mV, and assuming the output should ideally switch from 0 to 1 at 5 mV, then we know that the output could change from 0 to 1 at any input voltage between 2.5 and 7.5 mV. In this case we would be assured that any voltage between 7.5 and 12.5 mV would definitely produce the value 1. However, in the worst case, the output of binary 1 could be representing a nominal value of 10 mV with an actual applied voltage of 2.5 mV, an error of  $\frac{3}{4}$  bit which is the sum of the quantization error and the accuracy.

### **EXAMPLE 11-14**

A certain eight-bit ADC, similar to Figure 11-13, has a full-scale input of 2.55 V (i.e.,  $V_A = 2.55$  V produces a digital output of 11111111). It has a specified error of  $\pm \frac{1}{4}$  LSB. Determine the maximum amount of error in the measurement.

#### Solution

The step size is 2.55 V/( $2^8 - 1$ ), which is exactly 10 mV. This means that even if the DAC has no inaccuracies, the  $V_{AX}$  output could be off by as much as 10 mV because  $V_{AX}$  can change only in 10-mV steps; this is the quantization error. The specified error of  $\pm 1/4$  LSB is  $1/4 \times 10$  mV = 2.5 mV. This means that the  $V_{AX}$  value can be off by as much as 2.5 mV because of component inaccuracies. Thus, the total possible error could be as much as 10 mV + 2.5 mV = 12.5 mV. For example, suppose that the analog input was 1.268 V. If the DAC output were perfectly accurate, the staircase would stop at the 127th step (1.27 V). But let's say that  $V_{AX}$  was off by -2 mV, so it was 1.268 V at the 127th step. This would not be large enough to stop the conversion; it would stop at the 128th step. Thus, the digital output would be  $10000000_2 = 128_{10}$  (representing 12.8 V) for an analog input of 1.268 V, an error of 12 mV.

# Conversion Time, *t*<sub>C</sub>

The conversion time is shown in Figure 11-13(b) as the time interval between the end of the START pulse and the activation of the  $\overline{EOC}$  output. The counter starts counting from 0 and counts up until  $V_{AX}$  exceeds  $V_A$ , at which point  $\overline{EOC}$  goes LOW to end the conversion process. It should be clear that the value of the conversion time,  $t_C$ , depends on  $V_A$ . A larger value will require more steps before the staircase voltage exceeds  $V_A$ .

The maximum conversion time will occur when  $V_A$  is just below full scale so that  $V_{AX}$  must go to the last step to activate  $\overline{EOC}$ . For an *N*-bit converter, this will be

$$t_{\rm C}({\rm max}) = (2^N - 1)$$
 clock cycles

For example, the ADC in Example 11-12 would have a maximum conversion time of

$$t_{\rm C}({\rm max}) = (2^{10} - 1) \times 1 \,\mu{\rm s} = 1023 \,\mu{\rm s}$$

Sometimes, average conversion time is specified; it is half of the maximum conversion time. For the digital-ramp converter, this would be

$$t_{\rm C}({\rm avg}) = \frac{t_{\rm C}({\rm max})}{2} \approx 2^{N-1} \,{\rm clock \ cycles}$$

The major disadvantage of the digital-ramp method is that conversion time essentially doubles for each bit that is added to the counter, so that resolution can be improved only at the cost of a longer  $t_{\rm C}$ . This makes this type of ADC unsuitable for applications where repetitive A/D conversions of a fast-changing analog signal must be made. For low-speed applications, however, the relative simplicity of the digital-ramp converter is an advantage over the more complex, higher-speed ADCs.

### EXAMPLE 11-15

What will happen to the operation of a digital-ramp ADC if the analog input  $V_A$  is greater than the full-scale value?

### Solution

From Figure 11-13, it should be clear that the comparator output will never go LOW because the staircase voltage can never exceed  $V_A$ . Thus, pulses will be continually applied to the counter, so that the counter will repetitively count up from 0 to maximum, recycle back to 0, count up, and so on. This will produce repetitive staircase waveforms at  $V_{AX}$  going from 0 to full scale, and this will continue until  $V_A$  is decreased below full scale.

<b>REVIEW QUESTIONS</b>	1. Describe the basic operation of the digital-ramp ADC.
	2. Explain <i>quantization error</i> .
	3. Why does conversion time increase with the value of the analog input voltage?
	4. <i>True or false:</i> Everything else being equal, a 10-bit digital-ramp ADC will have a better resolution, but a longer conversion time, than an eight-bit ADC.
	5. Give one advantage and one disadvantage of a digital-ramp ADC.
	6. For the converter of Example 11-12, determine the digital output for $V_A = 1.345$ V. Repeat for $V_A = 1.342$ V.

# 11-10 DATA ACQUISITION

There are many applications in which analog data must be *digitized* (converted to digital) and transferred into a computer's memory. The process by which the computer acquires these digitized analog data is referred to as *data acquisition*. Acquiring a single data point's value is referred to as *sampling* the analog signal, and that data point is often called a *sample*. The computer can do several different things with the data, depending on the application. In a storage application, such as digital audio recording, video recording, or a digital oscilloscope, the internal microcomputer will store the data and then transfer them to a DAC at a later time to reproduce the original analog signal. In a process control application, the computer can examine the data or perform computations on them to determine what control outputs to generate.

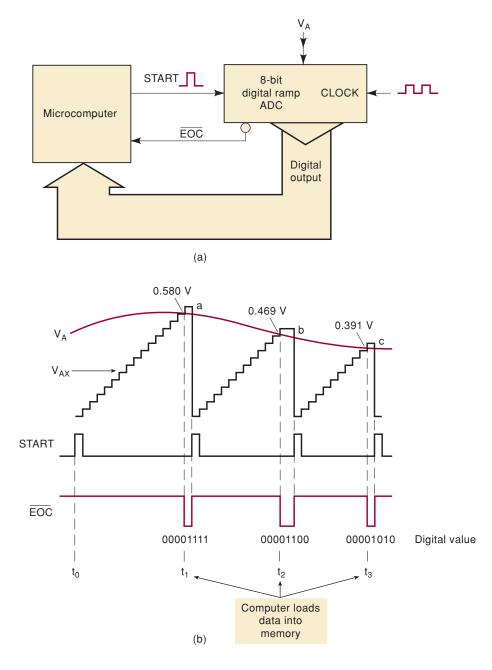
Figure 11-15(a) shows how a microcomputer is connected to a digital-ramp ADC for the purpose of data acquisition. The computer generates the START pulses that initiate each new A/D conversion. The  $\overline{EOC}$  (end-of-conversion) signal from the ADC is fed to the computer. The computer monitors  $\overline{EOC}$  to find out when the current A/D conversion is complete; then it transfers the digital data from the ADC output into its memory.

The waveforms in Figure 11-15(b) illustrate how the computer acquires a digital version of the analog signal,  $V_A$ . The  $V_{AX}$  staircase waveform that is generated internal to the ADC is shown superimposed on the  $V_A$  waveform for purposes of illustration. The process begins at  $t_0$ , when the computer generates a START pulse to start an A/D conversion cycle. The conversion is completed at  $t_1$ , when the staircase first exceeds  $V_A$ , and  $\overline{EOC}$  goes LOW. This NGT at  $\overline{EOC}$  signals the computer that the ADC has a digital output that now represents the value of  $V_A$  at point a, and the computer will load these data into its memory.

The computer generates a new START pulse shortly after  $t_1$  to initiate a second conversion cycle. Note that this resets the staircase to 0 and  $\overline{EOC}$  back HIGH because the START pulse resets the counter in the ADC. The second conversion ends at  $t_2$  when the staircase again exceeds  $V_A$ . The computer then loads the digital data corresponding to point b into its memory. These steps are repeated at  $t_3$ ,  $t_4$ , and so on.

The process whereby the computer generates a START pulse, monitors  $\overline{EOC}$ , and loads ADC data into memory is done under the control of a program that the computer is executing. This data acquisition program will determine how many data points from the analog signal will be stored in the computer memory.

**FIGURE 11-15** (a) Typical computer data acquisition system; (b) waveforms showing how the computer initiates each new conversion cycle and then loads the digital data into memory at the end of conversion.



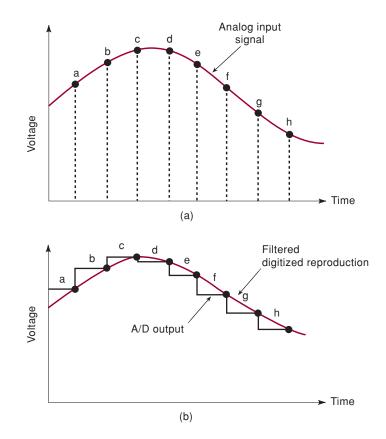
# **Reconstructing a Digitized Signal**

In Figure 11-15(b), the ADC is operating at its maximum speed because a new START pulse is generated immediately after the computer acquires the ADC output data from the previous conversion. Note that the conversion times are not constant because the analog input value is changing. The problem with this method of storing a waveform is that in order to reconstruct the waveform, we would need to know the point in time that each data value is to be plotted. Normally, when storing a digitized waveform, the samples are taken at fixed intervals at a rate that is at least two times greater than the highest frequency in the analog signal. The digital system will store the waveform as a list of sample data values. Table 11-5 shows the list of data that would be stored if the signal in Figure 11-16(a) were digitized.

<b>TABLE 11-5</b> Digitizeddata samples.	Point	Actual Voltage (V)	Digital Equivalent
	а	1.22	01111010
	b	1.47	10010011
	С	1.74	10101110
	d	1.70	10101010
	е	1.35	10000111
	f	1.12	01110000
	g	0.91	01011011
	h	0.82	01010010

#### **FIGURE 11-16**

(a) Digitizing an analog signal; (b) reconstructing the signal from the digital data.

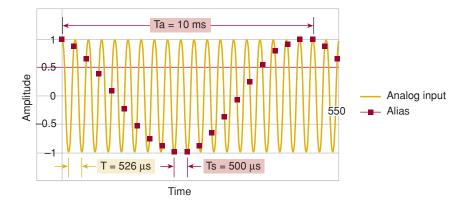


In Figure 11-16(a), we see how the ADC continually performs conversions to digitize the analog signal at points a, b, c, d, and so on. If these digital data are used to reconstruct the signal, the result will look like that in Figure 11-16(b). The black line represents the voltage waveform that would actually come out of the D/A converter. The red line would be the result of passing the signal through a simple low-pass RC filter. We can see that it is a fairly good reproduction of the original analog signal because the analog signal does not make any rapid changes between digitized points. If the analog signal contained higher-frequency variations, the ADC would not be able to follow the variations, and the reproduced version would be much less accurate.

### Aliasing

The obvious goal in signal reconstruction is to make the reconstruction nearly identical to the original analog signal. In order to avoid loss of information, as has been proven by a man named Harry Nyquist, the incoming signal must be sampled at a rate greater than two times the highest-frequency component in the incoming signal. For example, if you are pretty sure that the highest frequency in an audio system will be less than 10 kHz, you must sample the audio signal at 20,000 samples per second in order to be able to reconstruct the signal. The frequency at which samples are taken is referred to as the sampling frequency,  $F_{\rm S}$ . What do you think would happen if for some reason a 12-kHz tone is present in the input signal? Unfortunately, the system would not simply ignore it because it is too high! Rather, a phenomenon called *aliasing* would occur. A signal alias is produced by sampling the signal at a rate less than the minimum rate identified by Nyquist (twice the highest incoming frequency). In this case, any frequency over 10 kHz will produce an alias frequency. The alias frequency is always the difference between any integer multiple of the sampling frequency  $F_{\rm S}$  (20 kHz) and the incoming frequency that is being digitized (12 kHz). Instead of hearing a 12-kHz tone in the reconstructed signal, you would hear an 8-kHz tone that was not in the original signal.

To see how aliasing can happen, consider the sine wave in Figure 11-17. Its frequency is 1.9 kHz. The dots show where the waveform is sampled every 500  $\mu$ s ( $F_{\rm S} = 2$  kHz). If we connect the dots that make up the sampled waveform, we discover that they form a cosine wave that has a period of 10 ms and a frequency of 100 Hz. This demonstrates that the alias frequency is equal to the difference between the sample frequency and the incoming frequency. If we could hear the output that results from this data acquisition, it would not sound like 1.9 kHz; it would sound like 100 Hz.



The problem with **undersampling** ( $F_{\rm S} < 2 F_{\rm in}$  max) is that the digital system has no idea that there was actually a higher frequency at the input. It simply samples the input and stores the data. When it reconstructs the signal, the alias frequency (100 Hz) is present, the original 1.9 kHz is missing, and the reconstructed signal does not sound the same. This is why a data acquisition system must not allow frequencies greater than half of  $F_{\rm S}$  to be placed on the input.

<b>REVIEW QUESTIONS</b>	1. What is <i>digitizing a signal</i> ?
	2. Describe the steps in a computer data acquisition process.
	3. What is the minimum sample frequency needed to reconstruct an analog signal?
	4. What occurs if the signal is sampled at less than the minimum frequency determined in question 3?

**FIGURE 11-17** An alias signal due to undersampling.

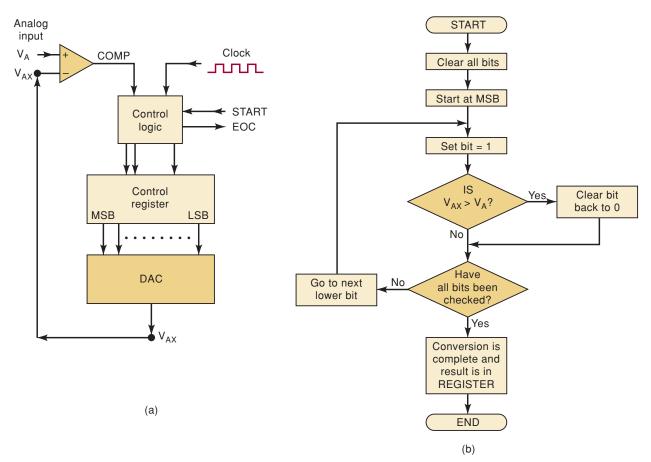
# 11-11 SUCCESSIVE-APPROXIMATION ADC

The **successive-approximation converter** is one of the most widely used types of ADC. It has more complex circuitry than the digital-ramp ADC but a much shorter conversion time. In addition, successive-approximation converters (SACs) have a fixed value of conversion time that is not dependent on the value of the analog input.

The basic arrangement, shown in Figure 11-18(a), is similar to that of the digital-ramp ADC. The SAC, however, does not use a counter to provide the input to the DAC block but uses a register instead. The control logic modifies the contents of the register bit by bit until the register data are the digital equivalent of the analog input  $V_A$  within the resolution of the converter. The basic sequence of operation is given by the flowchart in Figure 11-18(b). We will follow this flowchart as we go through the example illustrated in Figure 11-19.

For this example, we have chosen a simple four-bit converter with a step size of 1 V. Even though most practical SACs would have more bits and smaller resolution than our example, the operation will be exactly the same. At this point, you should be able to determine that the four register bits feeding the DAC have weights of 8, 4, 2, and 1 V, respectively.

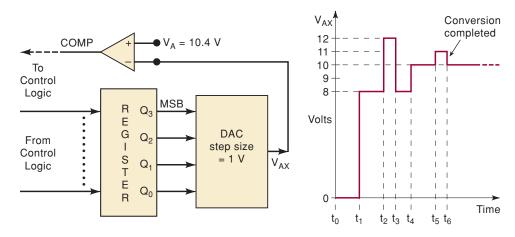
Let's assume that the analog input is  $V_A = 10.4$  V. The operation begins with the control logic clearing all of the register bits to 0 so that  $Q_3 = Q_2 = Q_1 = Q_0 = 0$ . We will express this as [Q] = 0000. This makes the DAC output  $V_{AX} = 0$  V, as indicated at time  $t_0$  on the timing diagram in Figure 11-19. With  $V_{AX} < V_A$ , the comparator output is HIGH.



**FIGURE 11-18** Successive-approximation ADC: (a) simplified block diagram; (b) flowchart of operation.

#### **FIGURE 11-19**

Illustration of four-bit SAC operation using a DAC step size of 1 V and  $V_A = 10.4$  V.



At the next step (time  $t_1$ ), the control logic sets the MSB of the register to 1 so that [Q] = 1000. This produces  $V_{AX} = 8$  V. Because  $V_{AX} < V_A$ , the COMP output is still HIGH. This HIGH tells the control logic that the setting of the MSB did not make  $V_{AX}$  exceed  $V_A$ , so that the MSB is kept at 1.

The control logic now proceeds to the next lower bit,  $Q_2$ . It sets  $Q_2$  to 1 to produce [Q] = 1100 and  $V_{AX} = 12$  V at time  $t_2$ . Because  $V_{AX} > V_A$ , the COMP output goes LOW. This LOW signals the control logic that the value of  $V_{AX}$  is too large, and the control logic then clears  $Q_2$  back to 0 at  $t_3$ . Thus, at  $t_3$ , the register contents are back to 1000 and  $V_{AX}$  is back to 8 V.

The next step occurs at  $t_4$ , where the control logic sets the next lower bit  $Q_1$  so that [Q] = 1010 and  $V_{AX} = 10$  V. With  $V_{AX} < V_A$ , COMP is HIGH and tells the control logic to keep  $Q_1$  set at 1.

The final step occurs at  $t_5$ , where the control logic sets the next lower bit  $Q_0$  so that [Q] = 1011 and  $V_{AX} = 11$  V. Because  $V_{AX} > V_A$ , COMP goes LOW to signal that  $V_{AX}$  is too large, and the control logic clears  $Q_0$  back to 0 at  $t_6$ .

At this point, all of the register bits have been processed, the conversion is complete, and the control logic activates its  $\overline{EOC}$  output to signal that the digital equivalent of  $V_A$  is now in the register. For this example, digital output for  $V_A = 10.4$  V is [Q] = 1010. Notice that 1010 is actually equivalent to 10 V, which is *less than* the analog input; this is a characteristic of the successiveapproximation method. Recall that in the digital-ramp method, the digital output was always equivalent to a voltage that was on the step above  $V_A$ .

### **EXAMPLE 11-16**

An eight-bit SAC has a resolution of 20 mV. What will its digital output be for an analog input of 2.17 V?

### Solution

$$2.17 \text{ V}/20 \text{ mV} = 108.5$$

so that step 108 would produce  $V_{AX} = 2.16$  V and step 109 would produce 2.18 V. The SAC always produces a final  $V_{AX}$  that is at the step *below*  $V_A$ . Therefore, for the case of  $V_A = 2.17$  V, the digital result would be  $108_{10} = 01101100_2$ .

### **Conversion Time**

In the operation just described, the control logic goes to each register bit, sets it to 1, decides whether or not to keep it at 1, and goes on to the next bit.

The processing of each bit takes one clock cycle, so that the total conversion time for an *N*-bit SAC will be *N* clock cycles. That is,

$$t_{\rm C}$$
 for SAC =  $N \times 1$  clock cycle

This conversion time will be the same *regardless of the value of*  $V_A$  because the control logic must process each bit to see whether or not a 1 is needed.

**EXAMPLE 11-17** 

Compare the maximum conversion times of a 10-bit digital-ramp ADC and a 10-bit successive-approximation ADC if both utilize a 500-kHz clock frequency.

#### **Solution**

For the digital-ramp converter, the maximum conversion time is

 $(2^N - 1) \times (1 \text{ clock cycle}) = 1023 \times 2 \ \mu \text{s} = 2046 \ \mu \text{s}$ 

For a 10-bit successive-approximation converter, the conversion time is always 10 clock periods or

$$10 \times 2 \ \mu s = 20 \ \mu s$$

Thus, it is about 100 times faster than the digital-ramp converter.

Because SACs have relatively fast conversion times, their use in data acquisition applications will permit more data values to be acquired in a given time interval. This feature can be very important when the analog data are changing at a relatively fast rate.

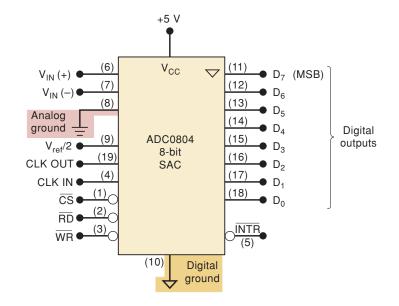
Because many SACs are available as ICs, it is rarely necessary to design the control logic circuitry, and so we will not cover it here. For those who are interested in the details of the control logic, many manufacturers' data books should provide sufficient detail.

# An Actual IC: The ADC0804 Successive-Approximation ADC

ADCs are available from several IC manufacturers with a wide range of operating characteristics and features. We will take a look at one of the more popular devices to get an idea of what is actually used in system applications. Figure 11-20 is the pin layout for the ADC0804, which is a 20-pin CMOS IC that performs A/D conversion using the successive-approximation method. Some of its important characteristics are as follows:

- It has two analog inputs,  $V_{IN}(+)$  and  $V_{IN}(-)$ , to allow differential inputs. In other words, the actual analog input,  $V_{IN}$ , is the difference in the voltages applied to these pins [analog  $V_{IN} = V_{IN}(+) - V_{IN}(-)$ ]. In single-ended measurements, the analog input is applied to  $V_{IN}(+)$ , while  $V_{IN}(-)$ , is connected to analog ground. During normal operation, the converter uses  $V_{CC} = +5$  V as its reference voltage, and the analog input can range from 0 to 5 V.
- It converts the differential analog input voltage to an eight-bit tristate buffered digital output. The internal circuitry is slightly more complex

FIGURE 11-20 ADC0804 eight-bit successiveapproximation ADC with tristate outputs. The numbers in parentheses are the IC's pin numbers.



than that described in Figure 11-19 in order to make transitions between output values occur at the nominal value  $\pm \frac{1}{2}$  LSB. For example, with 10-mV resolution, the A/D output would switch from 0 to 1 at 5 mV, from 1 to 2 at 15 mV, and so on. For this converter the resolution is calculated as  $V_{\text{REF}}/256$ ; with  $V_{\text{REF}} = 5.00$  V, the resolution is 19.53 mV. The nominal full-scale input is  $255 \times 19.53 = 4.98$  V, which should produce an output of 1111111. This converter will output 1111111 for any analog input between approximately 4.971 and 4.990 V.

- It has an internal clock generator circuit that produces a frequency of f = 1/(1.1RC), where *R* and *C* are values of externally connected components. A typical clock frequency is 606 kHz using  $R = 10 \text{ k}\Omega$  and C = 150 pF. An external clock signal can be used, if desired, by connecting it to the CLK IN pin.
- Using a 606-kHz clock frequency, the conversion time is approximately 100 μs.
- It has separate ground connections for digital and analog voltages. Pin 8 is the analog ground that is connected to the common reference point of the analog circuit that is generating the analog voltage. Pin 10 is the digital ground that is the one used by all of the digital devices in the system. (Note the different symbols used for the different grounds.) The digital ground is inherently noisy because of the rapid current changes that occur as digital devices change states. Although it is not necessary to use a separate analog ground, doing so ensures that the noise from digital ground is prevented from causing premature switching of the analog comparator inside the ADC.

This IC is designed to be easily interfaced to a microprocessor data bus. For this reason, the names of some of the ADC0804 inputs and outputs are based on functions that are common to microprocessor-based systems. The functions of these inputs and outputs are defined as follows:

•  $\overline{CS}$  (Chip Select). This input must be in its active-LOW state for  $\overline{RD}$  or  $\overline{WR}$  inputs to have any effect. With  $\overline{CS}$  HIGH, the digital outputs are in the Hi-Z state, and no conversions can take place.

- $\overline{RD}$  (READ). This input is used to enable the digital output buffers. With  $\overline{CS} = \overline{RD} = \text{LOW}$ , the digital output pins will have logic levels representing the results of the *last* A/D conversion. The microcomputer can then *read* (fetch) this digital data value over the system data bus.
- WR (WRITE). A LOW pulse is applied to this input to signal the start of a new conversion. This is actually a start conversion input. It is called a WRITE input because in a typical application, the microcomputer generates a WRITE pulse (similar to one used for writing to memory) that drives this input.
- INTR (INTERRUPT). This output signal will go HIGH at the start of a conversion and will return LOW to signal the end of conversion. This is actually an end-of-conversion output signal, but it is called INTERRUPT because in a typical situation, it is sent to a microprocessor's interrupt input to get the microprocessor's attention and let it know that the ADC's data are ready to be read.
- $V_{\text{ref}}/2$ . This is an optional input that can be used to reduce the internal reference voltage and thereby change the analog input range that the converter can handle. When this input is unconnected, it sits at 2.5 V  $(V_{CC}/2)$  because  $V_{CC}$  is being used as the reference voltage. By connecting an external voltage to this pin, the internal reference is changed to twice that voltage, and the analog input range is changed accordingly (see Table 11-6).
- CLK OUT. A resistor is connected to this pin to use the internal clock. The clock signal appears on this pin.

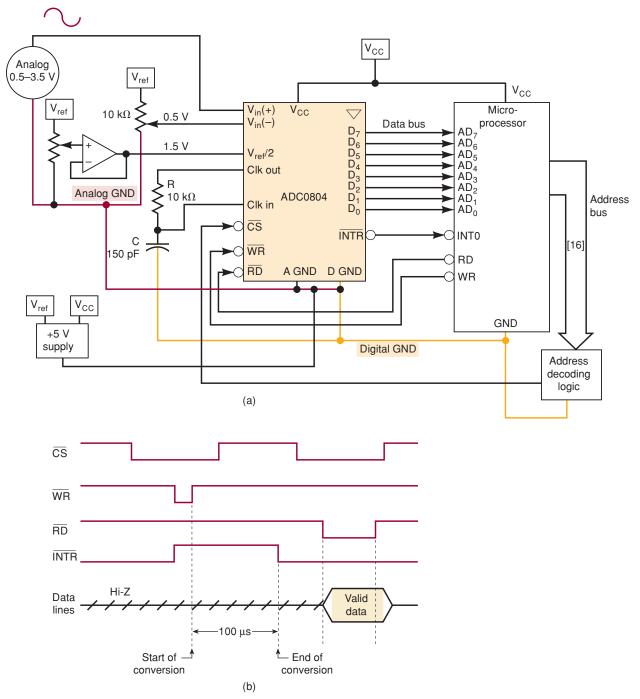
V <sub>ref</sub> /2	Analog Input Range (V)	Resolution (mV)
Open	0–5	19.5
2.25	0-4.5	17.6
2.0	0–4	15.6
1.5	0–3	11.7

**TABLE 11-6** 

CLK IN. Used for external clock input, or for a capacitor connection when the internal clock is used.

Figure 11-21(a) shows a typical connection of the ADC0804 to a microcomputer in a data acquisition application. The microcomputer controls when a conversion is to take place by generating the  $\overline{CS}$  and  $\overline{WR}$  signals. It then acquires the ADC output data by generating the  $\overline{CS}$  and  $\overline{RD}$  signals after detecting an NGT at  $\overline{INTR}$ , indicating the end of conversion. The waveforms in Figure 11-21(b) show the signal activity during the data acquisition process. Note that  $\overline{INTR}$  goes HIGH when  $\overline{CS}$  and  $\overline{WR}$  are LOW, but the conversion process does not begin until  $\overline{WR}$  returns HIGH. Also note that the ADC output data lines are in their Hi-Z state until the microcomputer activates  $\overline{CS}$  and  $\overline{RD}$ ; at that point the ADC's data buffers are enabled so that the ADC data are sent to the microcomputer over the data bus. The data lines return to the Hi-Z state when either  $\overline{CS}$  or  $\overline{RD}$  is returned HIGH.

In this application of the ADC0804, the input signal is varying over a range of 0.5 to 3.5 V. In order to make full use of the eight-bit resolution, the A/D must be matched to the analog signal specifications. In this case, the full-scale range is 3.0 V. However, it is offset from ground by 0.5 V. The offset of 0.5 V is applied to the negative input  $V_{\rm IN}(-)$ , establishing this as the 0 value reference. The range of 3.0 V is set by applying 1.5 V to  $V_{\rm ref}/2$ , which establishes  $V_{\rm ref}$ 



**FIGURE 11-21** (a) An application of an ADC0804; (b) typical timing signals during data acquisition.

as 3.0 V. An input of 0.5 V will produce a digital value of 00000000, and an input of 3.5 V (or any value over 3.482) will produce 11111111.

Another major concern when interfacing digital and analog signals is *noise*. Notice that the digital and analog ground paths are separated. The two grounds are tied together at a point that is very close to the A/D converter. A very low-resistance path ties this point directly to the negative terminal of the power supply. It is also wise to route the positive supply lines separately

to digital and analog devices and make extensive use of decoupling capacitors (0.01  $\mu$ F) from very near each chip's supply connection to ground.

<b>REVIEW QUESTIONS</b>	1. What is the main advantage of a SAC over a digital-ramp ADC?
	2. What is its principal disadvantage compared with the digital-ramp converter?
	3. <i>True or false:</i> The conversion time for a SAC increases as the analog voltage increases.
	4. Answer the following concerning the ADC0804.
	(a) What is its resolution in bits?
	(b) What is the normal analog input voltage range?
	(c) Describe the functions of the $\overline{CS}$ , $\overline{WR}$ , and $\overline{RD}$ inputs.
	(d) What is the function of the $\overline{INTR}$ output?
	(e) Why does it have two separate grounds?
	(f) What is the purpose of $V_{IN}(-)$ ?

## 11-12 FLASH ADCs

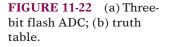
The **flash converter** is the highest-speed ADC available, but it requires much more circuitry than the other types. For example, a six-bit flash ADC requires 63 analog comparators, while an eight-bit unit requires 255 comparators, and a ten-bit converter requires 1023 comparators. The large number of comparators has limited the size of flash converters. IC flash converters are commonly available in two- to eight-bit units, and most manufacturers offer nine- and ten-bit units as well.

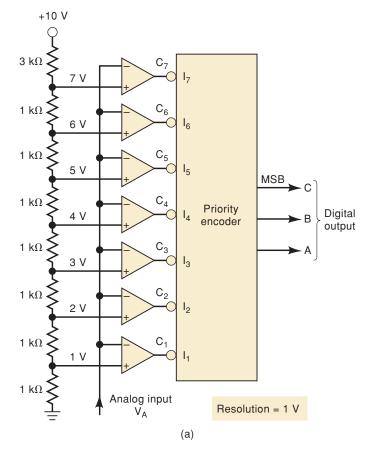
The principle of operation will be described for a three-bit flash converter in order to keep the circuitry at a workable level. Once the three-bit converter is understood, it should be easy to extend the basic idea to higherbit flash converters.

The flash converter in Figure 11-22(a) has a three-bit resolution and a step size of 1 V. The voltage divider sets up reference levels for each comparator so that there are seven levels corresponding to 1 V (weight of LSB), 2V, 3V, . . . , and 7 V (full scale). The analog input,  $V_A$ , is connected to the other input of each comparator.

With  $V_A < 1$  V, all of the comparator outputs  $C_1$  through  $C_7$  will be HIGH. With  $V_A > 1$  V, one or more of the comparator outputs will be LOW. The comparator outputs are fed into an active-LOW priority encoder that generates a binary output corresponding to the highest-numbered comparator output that is LOW. For example, when  $V_A$  is between 3 and 4 V, outputs  $C_1$ ,  $C_2$ , and  $C_3$  will be LOW and all others will be HIGH. The priority encoder will respond only to the LOW at  $C_3$  and will produce a binary output *CBA* = 011, which represents the digital equivalent of  $V_A$ , within the resolution of 1 V. When  $V_A$  is greater than 7 V,  $C_1$  to  $C_7$  will all be LOW, and the encoder will produce *CBA* = 111 as the digital equivalent of  $V_A$ . The table in Figure 11-22(b) shows the responses for all possible values of analog input.

The flash ADC of Figure 11-22 has a resolution of 1 V because the analog input must change by 1 V in order to bring the digital output to its next step. To achieve finer resolutions, we would have to increase the number of input voltage levels (i.e., use more voltage-divider resistors) and the number of





Analog in		Comparator outputs								puts
V <sub>A</sub>	C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>	$C_4$	$C_5$	C <sub>6</sub>	C <sub>7</sub>	С	В	А
0–1 V	1	1	1	1	1	1	1	0	0	0
1–2 V	0	1	1	1	1	1	1	0	0	1
2–3 V	0	0	1	1	1	1	1	0	1	0
3–4 V	0	0	0	1	1	1	1	0	1	1
4–5 V	0	0	0	0	1	1	1	1	0	0
5–6 V	0	0	0	0	0	1	1	1	0	1
6–7 V	0	0	0	0	0	0	1	1	1	0
> 7 V	0	0	0	0	0	0	0	1	1	1
(b)										

comparators. For example, an eight-bit flash converter would require  $2^8 = 256$  voltage levels, including 0 V. This would require 256 resistors and 255 comparators (there is no comparator for the 0-V level). The 255 comparator outputs would feed a priority encoder circuit that would produce an eight-bit code corresponding to the highest-order comparator output that is LOW. In general, an *N*-bit flash converter would require  $2^N - 1$  comparators,  $2^N$  resistors, and the necessary encoder logic.

## **Conversion Time**

The flash converter uses no clock signal because no timing or sequencing is required. The conversion takes place continuously. When the value of analog input changes, the comparator outputs will change, thereby causing the encoder outputs to change. The conversion time is the time it takes for a new digital output to appear in response to a change in  $V_A$ , and it depends only on the propagation delays of the comparators and encoder logic. For this reason, flash converters have extremely short conversion times. For example, the Analog Devices AD9020 is a 10-bit flash converter with a conversion time under 17 ns.

#### **REVIEW QUESTIONS**

- 1. True or false: A flash ADC does not contain a DAC.
- 2. How many comparators would a 12-bit flash converter require? How many resistors?
- 3. State the major advantage and disadvantage of a flash converter.

## 11-13 OTHER A/D CONVERSION METHODS

Several other methods of A/D conversion have been in use for some time, each with its relative advantages and disadvantages. We will briefly describe some of them now.

## Up/Down Digital-Ramp ADC (Tracking ADC)

As we have seen, the digital-ramp ADC is relatively slow because the counter is reset to 0 at the start of each new conversion. The staircase always begins at 0 V and steps its way up to the "switching point" where  $V_{AX}$  exceeds  $V_A$  and the comparator output switches LOW. The time it takes the staircase to reset to 0 and step back up to the new switching point is really wasted. The **up/down digital-ramp ADC** uses an up/down counter to reduce this wasted time.

The up/down counter replaces the up counter that feeds the DAC. It is designed to count up whenever the comparator output indicates that  $V_{AX} < V_A$  and to count down whenever  $V_{AX} > V_A$ . Thus, the DAC output is always being stepped in the direction of the  $V_A$  value. Each time the comparator output switches states, it indicates that  $V_{AX}$  has "crossed" the  $V_A$  value, the digital equivalent of  $V_A$  is in the counter, and the conversion is complete.

When a new conversion is to begin, the counter is *not reset to 0* but begins counting either up or down from its last value, depending on the comparator output. It will count until the staircase crosses  $V_A$  again to end the conversion. The  $V_{AX}$  waveform, then, will contain both positive-going and negative-going staircase signals that "track" the  $V_A$  signal. For this reason, this ADC is often called a *tracking ADC*.

Clearly, the conversion times will generally be reduced because the counter does not start over from 0 each time but simply counts up or down from its previous value. Of course, the value of  $t_{\rm C}$  will still depend on the value of  $V_A$ , and so it will not be constant.

## **Dual-Slope Integrating ADC**

The **dual-slope converter** has one of the slowest conversion times (typically 10 to 100 ms) but has the advantage of relatively low cost because it does not require precision components such as a DAC or a VCO. The basic operation of this converter involves the *linear* charging and discharging of a capacitor using constant currents. First, the capacitor is charged up for a fixed time interval from a constant current derived from the analog input voltage,  $V_A$ . Thus, at the end of this fixed charging interval, the capacitor voltage will be

proportional to  $V_A$ . At that point, the capacitor is linearly discharged from a constant current derived from a precise reference voltage,  $V_{ref}$ . When the capacitor voltage reaches 0, the linear discharging is terminated. During the discharge interval, a digital reference frequency is fed to a counter and counted. The duration of the discharge interval will be proportional to the initial capacitor voltage. Thus, at the end of the discharge interval, the counter will hold a count proportional to the initial capacitor voltage, which, as we said, is proportional to  $V_A$ .

In addition to its low cost, another advantage of the dual-slope ADC is its low sensitivity to noise and to variations in its component values caused by temperature changes. Because of its slow conversion times, the dual-slope ADC is not used in any data acquisition applications. The slow conversion times, however, are not a problem in applications such as digital voltmeters or multimeters, and this is where they find their major application.

## **Voltage-to-Frequency ADC**

The **voltage-to-frequency ADC** is simpler than other ADCs because it does not use a DAC. Instead it uses a *linear voltage-controlled oscillator (VCO)* that produces an output frequency proportional to its input voltage. The analog voltage that is to be converted is applied to the VCO to generate an output frequency. This frequency is fed to a counter to be counted for a fixed time interval. The final count is proportional to the value of the analog voltage.

To illustrate, suppose that the VCO generates a frequency of 10 kHz for each volt of input (i.e., 1 V produces 10 kHz, 1.5 V produces 15 kHz, 2.73 V produces 27.3 kHz). If the analog input voltage is 4.54 V, then the VCO output will be a 45.4-kHz signal that clocks a counter for, say, 10 ms. After the 10-ms counting interval, the counter will hold the count of 454, which is the digital representation of 4.54 V.

Although this is a simple method of conversion, it is difficult to achieve a high degree of accuracy because of the difficulty in designing VCOs with accuracies of better than 0.1 percent.

One of the main applications of this type of converter is in noisy industrial environments where small analog signals must be transmitted from transducer circuits to a control computer. The small analog signals can be drastically affected by noise if they are directly transmitted to the control computer. A better approach is to feed the analog signal to a VCO, which generates a digital signal whose output frequency changes according to the analog input. This digital signal is transmitted to the computer and will be much less affected by noise. Circuitry in the control computer will count the digital pulses (i.e., perform a frequency-counting function) to produce a digital value equivalent to the original analog input.

## Sigma/Delta Modulation\*

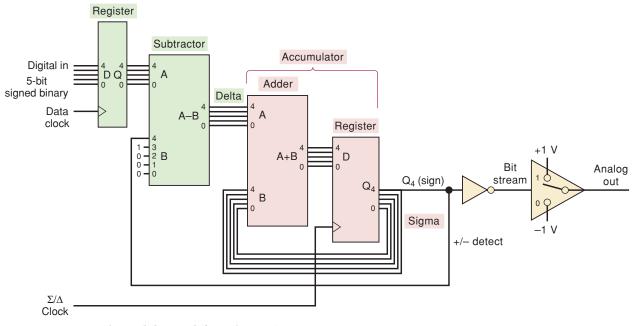
Another approach to representing analog information in digital form is called **sigma/delta modulation**. A sigma/delta A/D converter is an oversampling device, which means that it effectively samples the analog information more often than the minimum sample rate. The minimum sample rate is two times higher than the highest frequency in the incoming analog wave. The sigma/delta approach, like the voltage-to-frequency approach, does not directly produce a multibit

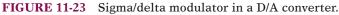
<sup>\*</sup>An excellent article published on the web by Jim Thompson, University of Washington, served as a basis for this description. Visit the *Digital Systems: Principles and Applications* Companion Web Site at http://www.prenhall.com/Tocci for the link to this article.

number for each sample. Instead, it represents the analog voltage by varying the density of logic 1s in a single-bit stream of serial data. To represent the positive portions of a waveform, a stream of bits with a high density of 1s is generated by the modulator (e.g., 01111101111101111101111). To represent the negative portions, a lower density of 1s (i.e., a higher density of 0s) is generated (e.g., 0001000100010001000).

Sigma/delta modulation is used in A/D as well as D/A conversion. One form of a sigma/delta modulator circuit is designed to convert a continuous analog signal into a modulated bit stream (A/D). The other form converts a sequence of digital samples into the modulated bit stream (D/A). We are coming from the perspective of digital systems, so it is easiest to understand the latter of these two circuits because it consists of all digital components that we have studied. Figure 11-23 shows a circuit that takes a five-bit signed digital value as its input and converts it into a sigma/delta bit stream. We will assume that the numbers that can be placed on this circuit's input range from -8 to +8. The first component is simply a subtractor (the delta section) similar to the one studied in Figure 6-14. The subtractor determines how far the input number is from its maximum or minimum value. This difference is often called the error signal. The second two components (the adder and the D register) form an accumulator very similar to the circuit in Figure 6-10 (the sigma section). For each sample that comes in, the accumulator adds the difference (error signal) to the running total. When the error is small, this running total (sigma) changes by small increments. When the error is large, the sigma changes by large increments. The last component compares the running total from the accumulator with a fixed threshold, which in this case is zero. In other words, it is simply determining if the total is positive or negative. This is accomplished by using the MSB (sign bit) of sigma. As soon as the total goes positive, the MSB goes LOW and feeds back to the delta section the maximum positive value (+8). When the MSB of sigma goes negative, it feeds back the maximum negative value (-8).

A spreadsheet is an excellent way to analyze a circuit like this. The tables in this section are taken from the spreadsheet that is included on the CD at





Sample ( <i>n</i> )	<b>Digital IN</b>	Delta	Sigma	Bit Stream Out	Analog OUT	Feedback
1	0	-8	0	1	1	8
2	0	8	-8	0	-1	-8
3	0	-8	0	1	1	8
4	0	8	-8	0	-1	-8
5	0	-8	0	1	1	8
6	0	8	-8	0	-1	-8
7	0	-8	0	1	1	8
8	0	8	-8	0	-1	-8

TABLE 11-8	Sigma/delta
modulator wit	th an input
of 4.	

**TABLE 11-7**Sigma/deltamodulator with an input

Sample ( <i>n</i> )	Digital IN	Delta	Sigma	Bit Stream Out	Analog OUT	Feedback
1	4	-4	4	1	1	8
2	4	-4	0	1	1	8
3	4	12	-4	0	-1	-8
4	4	-4	8	1	1	8
5	4	-4	4	1	1	8
6	4	-4	0	1	1	8
7	4	12	-4	0	-1	-8
8	4	-4	8	1	1	8

the back of this book. Table 11-7 shows the operation of the converter when a value of zero is the input. Notice that the bit stream output alternates between 1 and 0, and the average value of the analog output is 0 volts. Table 11-8 shows what happens when the digital input is 4. If we assume that 8 is full scale, this represents  $\frac{4}{8} = 0.5$ . The output is HIGH for three samples and LOW for one sample, a pattern that repeats every four samples. The average value of the analog output is (1 + 1 + 1 - 1)/4 = 0.5 V.

As a final example, let's use an input of -5, which represents -5/8 =-0.625. Table 11-9 shows the resulting output. The pattern in the bit stream is not periodic. From the sigma column, we can see that it takes 16 samples for the pattern to repeat. If we take the overall bit density, however, and calculate the average value of the analog output over 16 samples, we will find that it is equal to -0.625. Your CD player probably uses a sigma/delta D/A converter that operates in this fashion. The 16-bit digital numbers come off the CD serially; then they are formatted into parallel data patterns and clocked into a converter. As the changing numbers come into the converter, the average value of the analog out changes accordingly. Next, the analog output goes through a circuit called a low-pass filter that smoothes out the sudden changes and produces a smoothly changing voltage that is the average value of the bit stream. In your headphones, this changing analog signal sounds just like the original recording. A sigma/delta A/D converter works in a very similar way but converts the analog voltage into the modulated bit stream. To store the digitized data as a list of *N*-bit binary numbers, the average bit density of  $2^N$  bit-stream samples is calculated and stored.

of 0.

TABLE 11-9 Sigma/delta							
modulator with an input of	Sample (n)	<b>Digital IN</b>	Delta	Sigma	Bit Stream Out	Analog OUT	Feedback
-5.	1	-5	3	-5	0	-1	-8
	2	-5	3	-2	0	-1	-8
	3	-5	-13	1	1	1	8
	4	-5	3	-12	0	-1	-8
	5	-5	3	-9	0	-1	-8
	6	-5	3	-6	0	-1	-8
	7	-5	3	-3	0	-1	-8
	8	-5	-13	0	1	1	8
	9	-5	3	-13	0	-1	-8
	10	-5	3	-10	0	-1	-8
	11	-5	3	-7	0	-1	-8
	12	-5	3	-4	0	-1	-8
	13	-5	3	-1	0	-1	-8
	14	-5	-13	2	1	1	8
	15	-5	3	-11	0	-1	-8
	16	-5	3	-8	0	-1	-8
	17	-5	3	-5	0	-1	-8
	18	-5	3	-2	0	-1	-8

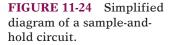
## **REVIEW QUESTIONS**

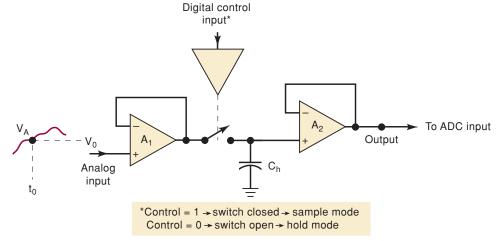
- 1. How does the up/down digital-ramp ADC improve on the digital-ramp ADC?
- 2. What is the main element of a voltage-to-frequency ADC?
- 3. Cite two advantages and one disadvantage of the dual-slope ADC.
- 4. Name three types of ADCs that do not use a DAC.
- 5. How many output data bits does a sigma/delta modulator use?

#### SAMPLE-AND-HOLD CIRCUITS 11-14

When an analog voltage is connected directly to the input of an ADC, the conversion process can be adversely affected if the analog voltage is changing during the conversion time. The stability of the conversion process can be improved by using a sample-and-hold (S/H) circuit to hold the analog voltage constant while the A/D conversion is taking place. A simplified diagram of a sample-and-hold (S/H) circuit is shown in Figure 11-24.

The S/H circuit contains a unity-gain buffer amplifier  $A_1$  that presents a high impedance to the analog signal and has a low output impedance that can rapidly charge the hold capacitor,  $C_{\rm h}$ . The capacitor will be connected to the output of  $A_1$  when the digitally controlled switch is closed. This is called the sample operation. The switch will be closed long enough for  $C_{\rm h}$  to charge to the present value of the analog input. For example, if the switch is closed at time  $t_0$ , the  $A_1$  output will quickly charge  $C_h$  up to a voltage  $V_0$ . When the switch opens,  $C_{\rm h}$  will hold this voltage so that the output of  $A_2$  will apply this voltage to the ADC. The unity-gain buffer amplifier  $A_2$  presents a high input impedance that will not discharge the capacitor voltage appreciably during





the conversion time of the ADC, and so the ADC will essentially receive a dc input voltage  $V_0$ .

In a computer-controlled data acquisition system such as the one discussed earlier, the sample-and-hold switch would be controlled by a digital signal from the computer. The computer signal would close the switch in order to charge  $C_h$  to a new sample of the analog voltage; the amount of time the switch would have to remain closed is called the **acquisition time**, and it depends on the value of  $C_h$  and the characteristics of the S/H circuit. The computer signal would then open the switch to allow  $C_h$  to hold its value and provide a relatively constant analog voltage at the  $A_2$  output.

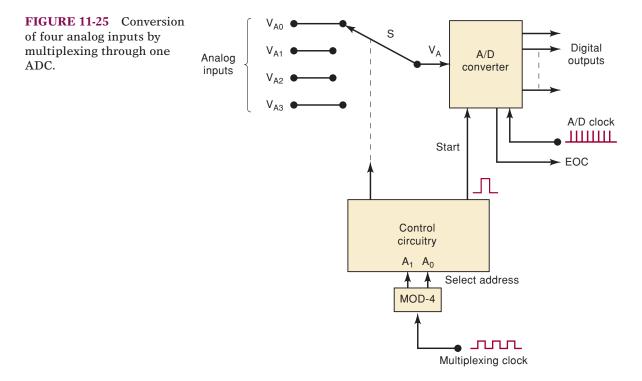
The AD1154 is a sample-and-hold integrated circuit that has an internal hold capacitor with an acquisition time of 3.5  $\mu$ s. During the hold time, the capacitor voltage will droop (discharge) at a rate of only 0.1  $\mu$ V/ $\mu$ s. The voltage droop within the sampling interval should be less than the weight of the LSB. For example, a 10-bit converter with a full-scale range of 10 V would have an LSB weight of approximately 10 mV. It would take 100 ms before the capacitor droop would equal the weight of the ADC's LSB. It is not likely, however, that it would ever be necessary to hold the sample for such a long time in the conversion process.

#### **REVIEW QUESTIONS**

- 1. Describe the function of a sample-and-hold circuit.
- 2. *True or false:* The amplifiers in an S/H circuit are used to provide voltage amplification.

## 11-15 MULTIPLEXING

When analog inputs from several sources are to be converted, a multiplexing technique can be used so that one ADC may be time-shared. The basic scheme is illustrated in Figure 11-25 for a four-channel acquisition system. Rotary switch *S* is used to switch each analog signal to the input of the ADC, one at a time in sequence. The control circuitry controls the switch position according to the *select address* bits,  $A_1$ ,  $A_0$ , from the MOD-4 counter. For example, with  $A_1A_0 = 00$ , the switch connects  $V_{A0}$  to the ADC input;  $A_1A_0 = 01$  connects  $V_{A1}$  to the ADC input; and so on. Each input channel has a specific address code that, when present, connects that channel to the ADC.



The operation proceeds as follows:

- 1. With select address = 00,  $V_{A0}$  is connected to the ADC input.
- 2. The control circuit generates a START pulse to initiate the conversion of  $V_{A0}$  to its digital equivalent.
- 3. When the conversion is complete, *EOC* signals that the ADC output data are ready. Typically, these data will be transferred to a computer over a data bus.
- 4. The multiplexing clock increments the select address to 01, which connects  $V_{A1}$  to the ADC.
- 5. Steps 2 and 3 are repeated with the digital equivalent of  $V_{A1}$  now present at the ADC outputs.
- 6. The multiplexing clock increments the select address to 10, and  $V_{A2}$  is connected to the ADC.
- 7. Steps 2 and 3 are repeated with the digital equivalent of  $V_{A2}$  now present at the ADC outputs.
- 8. The multiplexing clock increments the select address to 11, and  $V_{A3}$  is connected to the ADC.
- 9. Steps 2 and 3 are repeated with the digital equivalent of  $V_{A3}$  now present at the ADC outputs.

The multiplexing clock controls the rate at which the analog signals are sequentially switched into the ADC. The maximum rate is determined by the delay time of the switches and the conversion time of the ADC. The switch delay time can be minimized by using semiconductor switches such as the CMOS bilateral switch described in Chapter 8. It may be necessary to connect a sample-and-hold circuit at the input of the ADC if the analog inputs will change significantly during the ADC conversion time.

Many integrated ADCs contain the multiplexing circuitry on the same chip as the ADC. The ADC0808, for example, can multiplex eight different analog inputs into one ADC. It uses a three-bit select input code to determine which analog input is connected to the ADC.

### **REVIEW QUESTIONS**

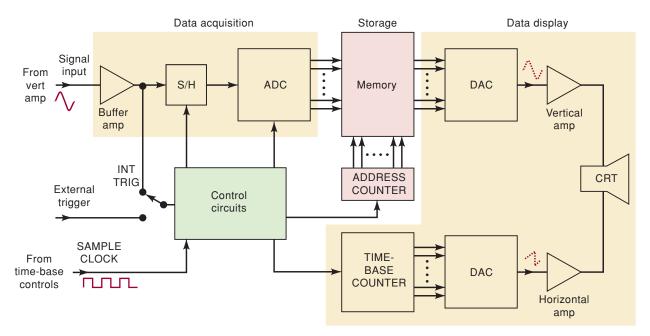
- 1. What is the advantage of this multiplexing scheme?
- 2. How would the address counter be changed if there were eight analog inputs?

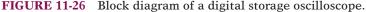
## 11-16 DIGITAL STORAGE OSCILLOSCOPE

As a final example of the application of D/A and A/D converters, we will take a brief look at the digital storage oscilloscope (DSO). The DSO uses both of these devices to digitize, store, and display analog waveforms.

A block diagram of a DSO is shown in Figure 11-26. The overall operation is controlled and synchronized by the circuits in the CONTROL block, which usually contains a microprocessor executing a control program stored in ROM (read-only memory). The data acquisition portion of the system contains a sample-and-hold (S/H) and an ADC that repetitively samples and digitizes the input signal at a rate determined by the SAMPLE CLOCK and then sends the digitized data to memory for storage. The CONTROL block makes sure that successive data points are stored in successive memory locations by continually updating the memory's ADDRESS COUNTER.

When memory is full, the next data point from the ADC is stored in the first memory location, writing over the old data, and so on, for successive data points. This data acquisition and storage process continues until the CONTROL block receives a trigger signal from either the input waveform (INTERNAL trigger) or an EXTERNAL trigger source. When the trigger





occurs, the system stops acquiring new data and enters the display mode of operation, in which all or part of the memory data is repetitively displayed on the CRT.

The display operation uses two DACs to provide the vertical and horizontal deflection voltages for the CRT. Data from memory produce the vertical deflection of the electron beam, while the TIME-BASE COUNTER provides the horizontal deflection in the form of a staircase sweep signal. The CONTROL block synchronizes the display operation by incrementing the memory ADDRESS COUNTER and the TIME-BASE COUNTER at the same time so that each horizontal step of the electron beam is accompanied by a new data value from memory to the vertical DAC. The counters are continuously recycled so that the stored data points are repetitively replotted on the CRT screen. The screen display consists of discrete dots representing the various data points, but the number of dots is usually so large (typically 1000 or more) that they tend to blend together and appear to be a smooth, continuous waveform. The display operation is terminated when the operator presses a front-panel button that commands the DSO to begin a new data acquisition cycle.

## **Related Applications**

The same sequence of operations performed in a DSO—data acquisition/digitizing/storage/data outputting—is used in other applications of DACs and ADCs. For example, heart monitors that can be found in any hospital are similar to DSOs but are constantly displaying a waveform showing the patient's heart activity over the past several seconds. As another example, digital video cameras digitize an image one picture element (pixel) at a time and store the information on magnetic tape or DVD. Digital still cameras digitize each pixel and store the data on a solid-state memory card. The data can later be transferred digitally and then output to a display device, where the data is converted to an analog "brightness" signal for each pixel and reassembled to form an image on the display.

#### **REVIEW QUESTIONS**

- 1. Look at Figure 11-26. How are waveforms "stored" in a DSO?
- 2. Describe the functions of the ADC and DACs that are part of the DSO.

## 11-17 DIGITAL SIGNAL PROCESSING (DSP)

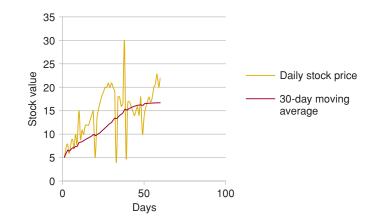
One of the most dynamic areas of digital systems today is in the field of **digital signal processing (DSP)**. A DSP is a very specialized form of microprocessor that has been optimized to perform repetitive calculations on streams of digitized data. The digitized data are usually being fed to the DSP from an A/D converter. It is beyond the scope of this text to explain the mathematics that allow a DSP to process these data values, but suffice it to say that for each new data point that comes in, a calculation is performed (very quickly). This calculation involves the most recent data point as well as several of the preceding data samples. The result of the calculation produces a new output data point, which is usually sent to a D/A converter. A DSP system is similar to the block diagram shown in Figure 11-1. The main difference is in the specialized hardware contained in the computer section.

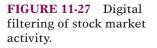
A major application for DSP is in filtering and conditioning of analog signals. As a very simple example, a DSP can be programmed to take in an analog waveform, such as the output from an audio preamplifier, and pass to the output only those frequency components that are below a certain frequency. All higher frequencies are attenuated by the filter. Perhaps you recall from your study of analog circuits that the same thing can be accomplished by a simple low-pass filter made from a resistor and capacitor. The advantage of DSP over resistors and capacitors is the flexibility of being able to change the critical frequency without switching any components. Instead, numbers are simply changed in the calculations to adapt the dynamic response of the filter. Have you ever been in an auditorium when the PA system started to squeal? This can be prevented if the degenerative feedback frequency can be filtered out. Unfortunately, the frequency that causes the squeal changes with the number of people in the room, the clothes they wear, and many other factors. With a DSP-based audio equalizer, the oscillation frequency can be detected and the filters dynamically adjusted to tune it out.

## **Digital Filtering**

To help you understand digital filtering, imagine you are buying and selling stock. To decide when to buy and sell, you need to know what the market is doing. You want to ignore sudden, short-term (high-frequency) changes but respond to the overall trends (30-day averages). Every day you read the newspaper, take a sample of the closing price for your stock, and write it down. Then you use a formula to calculate the average of the last 30 days' prices. This average value is plotted as shown in Figure 11-27, and the resulting graph is used to make decisions. This is a way of filtering the digital signal (sequence of data samples) that represents the stock market activity.

Now imagine that instead of sampling stock prices, a digital system is sampling an audio (analog) signal from a microphone using an A/D converter. Instead of taking a sample once a day, it takes a sample 20,000 times each second (every 50  $\mu$ s). For each sample, a weighted averaging calculation is performed using the last 256 data samples and produces a single output data point. A **weighted average** means that some of the data points are considered more important than others. Each of the samples is multiplied by a fractional number (between 0 and 1) before adding them together. This averaging calculation is processing (filtering) the audio signal.





The most difficult part of this form of DSP is determining the correct weighting constants for the averaging calculation in order to achieve the desired filter characteristics. Fortunately, there is readily available software for PCs that makes this very easy. The special DSP hardware must perform the following operations:

Read the newest sample (one new number) from A/D.

Replace the oldest sample (of 256) with the new one from A/D.

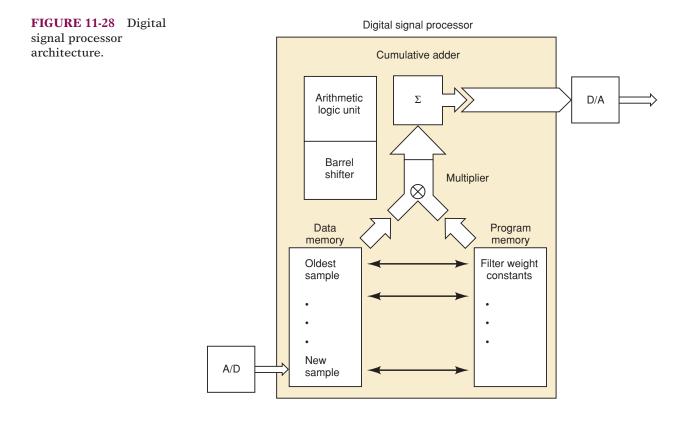
Multiply each of the 256 samples by their corresponding weight constant.

Add all of these products.

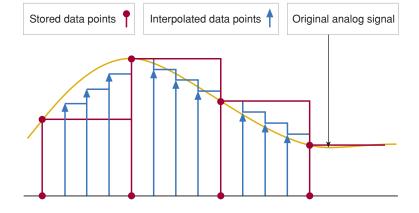
Output the resulting sum of products (1 number) to the D/A.

Figure 11-28 shows the basic architecture of a DSP. The multiply and accumulate (MAC) section is central to all DSPs and is used in most applications. Special hardware, like you will study in Chapter 12, is used to implement the memory system that stores the data samples and weight values. The arithmetic logic unit and barrel shifter (shift register) provide the necessary support to deal with the binary number system while processing signals.

Another useful application of DSP is called **oversampling** or **interpolation filtering**. As you recall, the reconstructed waveform is always an approximation of the original due to quantization error. The sudden step changes from one data point to the next also introduce high-frequency noise into the reconstructed signal. A DSP can insert interpolated data points into the digital signal. Figure 11-29 shows how 4X oversampling interpolation filtering smoothes out the waveform and makes final filtering possible



**FIGURE 11-29** Inserting an interpolated data point into a digital signal to reduce noise.



with simpler analog circuitry. DSP performs this role in your CD player to provide an excellent audio reproduction. The round dots represent the digitally recorded data on your CD. The triangles represent the interpolated data points that the digital filter in your CD player inserts before the final analog output filter.

Many of the important concepts that you need to understand in order to move on to DSP have been presented in this and previous chapters. A/D and D/A conversion methods and hardware along with data acquisition and sampling concepts are vital. Topics such as signed binary number representations (including fractions), signed binary addition and multiplication (covered in Chapter 6), and shift registers (Chapter 7) are necessary to understand the hardware and programming of a DSP. Memory system concepts, which will be presented in the next chapter, will also be important.

DSP is being integrated into many common systems that you are familiar with. CD players use DSP to filter the digital data being read from the disk to minimize the quantization noise that is unavoidably caused by digitizing the music. Telephone systems use DSP to cancel echoes on the phone lines. The high-speed modems that are standard on PCs have been made possible and affordable by DSP. Special effects boxes for guitars and other instruments perform echo, reverb, phasing, and other effects using DSP. Applications of DSP are growing right now at the same rate that microprocessor applications grew in the early 1980s. They provide a digital solution to many traditionally analog problems. Some other examples of applications include speech recognition, telecommunications data encryption, fast Fourier transforms, image processing in digital television, ultrasonic beam forming in medical electronics, and noise cancellation in industrial controls. As this trend continues, you can expect to see nearly all electronic systems containing digital signal processing circuitry.

REVIEW QUESTIONS	1. What is a major application of DSP?
	2. What is the typical source of digital data for a DSP to process?
	3. What advantage does a DSP filter have over an analog filter circuit?
	4. What is the central hardware feature of a DSP?
	5. How many interpolated data points are inserted between samples when performing 4X oversampled digital filtering? How many for 8X oversampling?

# **SUMMARY**

- 1. Physical variables that we want to measure, such as temperature, pressure, humidity, distance velocity, and so on, are continuously variable quantities. A transducer can be used to translate these quantities into an electrical signal of voltage or current that fluctuates in proportion to the physical variable. These continuously variable voltage or current signals are called *analog* signals.
- 2. To measure a physical variable, a digital system must assign a binary number to the analog value that is present at that instant. This is accomplished by an A/D converter. To generate variable voltages or current values that can control physical processes, a digital system must translate binary numbers into a voltage or current magnitude. This is accomplished by a D/A converter.
- 3. A D/A converter with n bits divides a range of analog values (voltage or current) into  $2^n 1$  pieces. The size or magnitude of each piece is the analog equivalent weight of the least significant bit. This is called the *resolution* or *step size*.
- 4. Most D/A converters use resistor networks that can cause weighted amounts of current to flow when any of its binary inputs are activated. The amount of current that flows is proportional to the binary weight of each input bit. These weighted currents are summed to create the analog signal out.
- 5. An A/D converter must assign a binary number to an analog (continuously variable) quantity. The precision with which an A/D converter can perform this conversion depends on how many different numbers it can assign and how wide the analog range is. The smallest change in analog value that an A/D can measure is called its *resolution*, the weight of its least significant bit.
- 6. By repeatedly sampling the incoming analog signal, converting it to digital, and storing the digital values in a memory device, an analog waveform can be captured. To reconstruct the signal, the digital values are read from the memory device at the same rate at which they were stored, and then they are fed into a D/A converter. The output of the D/A is then filtered to smooth the stair steps and re-create the original waveform. The bandwidth of sampled signals is limited to  $\frac{1}{2} F_S$ . Incoming frequencies greater than  $\frac{1}{2} F_S$  create an *alias* that has a frequency equal to the difference between the nearest integer multiple of  $F_S$  and the incoming frequency. This difference will always be less than  $\frac{1}{2} F_S$ .
- 7. A digital-ramp A/D is the simplest to understand but it is not often used due to its variable conversion time. A successive-approximation converter has a constant conversion time and is probably the most common general-purpose converter.
- 8. Flash converters use analog comparators and a priority encoder to assign a digital value to the analog input. These are the fastest converters because the only delays involved are propagation delays.
- 9. Other popular methods of A/D include up/down tracking, integrating, voltage-to-frequency conversion, and sigma/delta conversion. Each type of converter has its own niche of applications.
- 10. Any D/A converter can be used with other circuitry such as analog multiplexers that select one of several analog signals to be converted, one at a time. Sample-and-hold circuits can be used to "freeze" a rapidly changing analog signal while the conversion is taking place.

11. Digital signal processing is an exciting new growth field in electronics. These devices allow calculations to be performed quickly in order to emulate the operation of many analog filter circuits digitally. The primary architectural feature of a DSP is a hardware multiplier and adder circuit that can multiply pairs of numbers together and accumulate the running total (sum) of these products. This circuitry is used to perform efficiently the weighted moving average calculations that are used to implement digital filters and other DSP functions. DSP is responsible for many of the recent advances in high-fidelity audio, high-definition TV, and telecommunications.

## **IMPORTANT TERMS**

digital quantity digital-ramp ADC voltage-to-frequency analog quantity quantization error ADC transducer sampling sigma/delta analog-to-digital sampling frequency, modulation converter (ADC)  $F_{\rm S}$ sample-and-hold digital-to-analog alias (S/H) circuit converter (DAC) undersampling acquisition time full-scale output successivedigital signal resolution approximation processing (DSP) step size ADC weighted average staircase differential inputs MAC arithmetic logic unit full-scale error WRITE flash ADC barrel shifter linearity error up/down digital-ramp offset error oversampling settling time ADC interpolation monotonicity dual-slope ADC filtering digitization

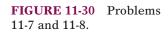
## PROBLEMS

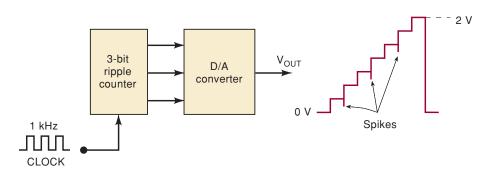
#### **SECTIONS 11-1 AND 11-2**

- B 11-1. DRILL QUESTION
  - (a) What is the expression relating the output and inputs of a DAC?
  - (b) Define *step size* of a DAC.
  - (c) Define *resolution* of a DAC.
  - (d) Define *full scale*.
  - (e) Define *percentage resolution*.
  - (f)\**True or false:* A 10-bit DAC will have a smaller resolution than a 12bit DAC for the same full-scale output.
  - (g)\**True or false:* A 10-bit DAC with full-scale output of 10 V has a smaller percentage resolution than a 10-bit DAC with 12 V full scale.
- **B** 11-2. An eight-bit DAC produces an output voltage of 2.0 V for an input code of 01100100. What will the value of  $V_{\text{OUT}}$  be for an input code of 10110011?
- **B** 11-3.\*Determine the weight of each input bit for the DAC of Problem 11-2.

<sup>\*</sup>Answers to problems marked with an asterisk can be found in the back of the text.

- **B** 11-4. What is the resolution of the DAC of Problem 11-2? Express it in volts and as a percentage.
- **B** 11-5.\*What is the resolution in volts of a 10-bit DAC whose F.S. output is 5 V?
- **B** 11-6. How many bits are required for a DAC so that its F.S. output is 10 mA and its resolution is less than 40  $\mu$ A?
- **B** 11-7.\*What is the percentage resolution of the DAC of Figure 11-30? What is the step size if the top step is 2 V?





- C 11-8. What is the cause of the negative-going spikes on the  $V_{\text{OUT}}$  waveform of Figure 11-30? (*Hint:* Note that the counter is a ripple counter and that the spikes occur on every other step.)
- **B** 11-9.\*Assuming a 12-bit DAC with perfect accuracy, how close to 250 rpm can the motor speed be adjusted in Figure 11-4?
  - 11-10. A 12-bit DAC has a full-scale output of 15.0 V. Determine the step size, the percentage resolution, and the value of  $V_{\text{OUT}}$  for an input code of 011010010101.
  - 11-11.\*A microcontroller has an eight-bit output port that is to be used to drive a DAC. The DAC that is available has 10 input bits and has a full-scale output of 10 V. The application requires a voltage that ranges between 0 and 10 V in steps of 50 mV or smaller. Which eight bits of the 10-bit DAC will be connected to the output port?
  - 11-12. You need a DAC that can span 12 V with a resolution of 20 mV or less. How many bits are needed?

#### **SECTION 11-3**

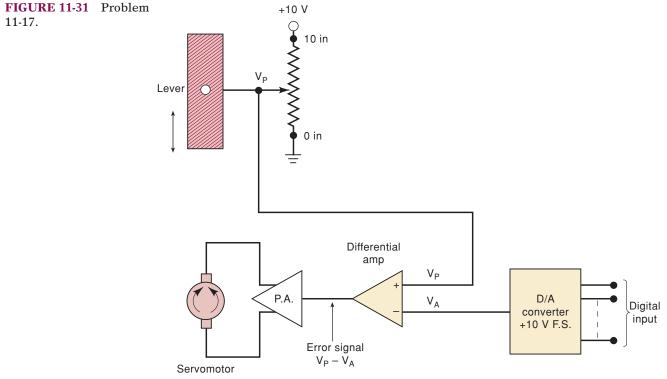
- **D** 11-13.\*The step size of the DAC of Figure 11-5 can be changed by changing the value of  $R_{\rm F}$ . Determine the required value of  $R_{\rm F}$  for a step size of 0.5 V. Will the new value of  $R_{\rm F}$  change the percentage resolution?
- D 11-14. Assume that the output of the DAC in Figure 11-7(a) is connected to the op-amp of Figure 11-7(b).
  - (a) With  $V_{\text{REF}} = 5 \text{ V}$ ,  $R = 20 \text{ k}\Omega$ , and  $R_{\text{F}} = 10 \text{ k}\Omega$ , determine the step size and the full-scale voltage at  $V_{\text{OUT}}$ .
  - (b) Change the value of  $R_{\rm F}$  so that the full-scale voltage at  $V_{\rm OUT}$  is -2 V.
  - (c) Use this new value of  $R_{\rm F}$ , and determine the proportionality factor, K, in the relationship  $V_{\rm OUT} = K(V_{\rm REF} \times B)$ .
  - 11-15.\* What is the advantage of the DAC of Figure 11-8 over that of Figure 11-7, especially for a larger number of input bits?

#### **SECTIONS 11-4 TO 11-6**

- 11-16. An eight-bit DAC has a full-scale error of 0.2% F.S. If the DAC has a full-scale output of 10 mA, what is the most that it can be in error for any digital input? If the D/A output reads 50  $\mu$ A for a digital input of 00000001, is this within the specified range of accuracy? (Assume no offset error.)
- C, N 11-17.\*The control of a positioning device may be achieved using a *servomotor*, which is a motor designed to drive a mechanical device as long as an error signal exists. Figure 11-31 shows a simple servo-controlled system that is controlled by a digital input that could be coming directly from a computer or from an output medium such as magnetic tape. The lever arm is moved vertically by the servomotor. The motor rotates clockwise or counterclockwise, depending on whether the voltage from the power amplifier (P.A.) is positive or negative. The motor stops when the P.A. output is 0.

The mechanical position of the lever is converted to a dc voltage by the potentiometer arrangement shown. When the lever is at its 0 reference point,  $V_P = 0$  V. The value of  $V_P$  increases at the rate of 1 V/inch until the lever is at its highest point (10 inches) and  $V_{\rm P} = 10$  V. The desired position of the lever is provided as a digital code from the computer and is then fed to a DAC, producing  $V_A$ . The difference between  $V_P$ and  $V_A$  (called *error*) is produced by the *differential* amplifier and is amplified by the P.A. to drive the motor in the direction that causes the error signal to decrease to 0—that is, moves the lever until  $V_{\rm P} = V_{\rm A}$ .

- (a) If the lever must be positioned within a resolution of 0.1 in, what is the number of bits needed in the digital input code?
- (b) In actual operation, the lever arm might oscillate slightly around the desired position, especially if a wire-wound potentiometer is used. Can you explain why?



## B 11-18. DRILL QUESTION

- (a) Define *binary-weighted resistor network*.
- (b) Define *R*/2*R* ladder network.
- (c) Define *DAC settling time*.
- (d) Define *full-scale error*.
- (e) Define offset error.
- 11-19.\*A particular six-bit DAC has a full-scale output rated at 1.260 V. Its accuracy is specified as  $\pm 0.1\%$  F.S., and it has an offset error of  $\pm 1$  mV. Assume that the offset error has not been zeroed out. Consider the measurements made on this DAC (Table 11-10), and determine which of them are not within the device's specifications. (*Hint:* The offset error is added to the error caused by component inaccuracies.)

<b>TABLE 11-10</b>	Input Code	Output
	000010	41.5 mV
	000111	140.2 mV
	001100	242.5 mV
	111111	1.258 V

#### **SECTION 11-7**

T 11-20. A certain DAC has the following specifications: eight-bit resolution, full scale = 2.55 V, offset  $\leq 2 \text{ mV}$ ; accuracy =  $\pm 0.1\%$  F.S. A static test on this DAC produces the results shown in Table 11-11. What is the probable cause of the malfunction?

<b>TABLE 11-11</b>	Input Code	Output
	0000000	8 mV
	0000001	18.2 mV
	0000010	28.5 mV
	00000100	48.3 mV
	00001111	158.3 mV
	1000000	1.289 V

**T** 11-21.\*Repeat Problem 11-20 using the measured data given in Table 11-12.

<b>TABLE 11-12</b>	Input Code	Output
	0000000	20.5 mV
	0000001	30.5 mV
	0000010	20.5 mV
	00000100	60.6 mV
	00001111	150.6 mV
	1000000	1.300 V

 T 11-22.\*A technician connects a counter to the DAC of Figure 11-3 to perform a staircase test using a 1-kHz clock. The result is shown in Figure 11-32. What is the probable cause of the incorrect staircase signal?

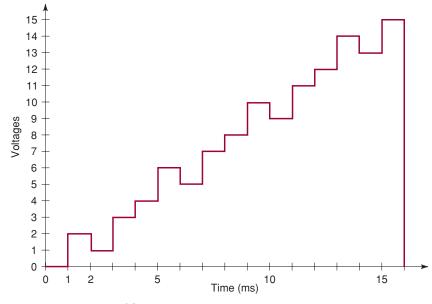


FIGURE 11-32 Problem 11-22.

#### **SECTIONS 11-8 AND 11-9**

#### 11-23. DRILL QUESTION

Fill in the blanks in the following description of the ADC of Figure 11-13. Each blank may be one or more words.

A START pulse is applied to \_\_\_\_\_ the counter and to keep \_\_\_\_\_ from passing through the AND gate into the \_\_\_\_\_. At this point, the DAC output,  $V_{AX}$ , is \_\_\_\_\_ and  $\overline{EOC}$  is \_\_\_\_\_.

When START returns \_\_\_\_\_, the AND gate is \_\_\_\_\_, and the counter is allowed to \_\_\_\_\_. The  $V_{AX}$  signal is increased one \_\_\_\_\_ at a time until it \_\_\_\_\_  $V_A$ . At that point, \_\_\_\_\_ goes LOW to \_\_\_\_\_ further pulses from \_\_\_\_\_. This signals the end of conversion, and the digital equivalent of  $V_A$  is present at the \_\_\_\_\_.

- **B** 11-24. An eight-bit digital-ramp ADC with a 40-mV resolution uses a clock frequency of 2.5 MHz and a comparator with  $V_{\rm T} = 1$  mV. Determine the following values.
  - (a)\*The digital output for  $V_A = 6.000$  V
  - (b) The digital output for 6.035 V
  - (c) The maximum and average conversion times for this ADC
- **B** 11-25. Why were the digital outputs the same for parts (a) and (b) of Problem 11-24?
- **D** 11-26. What would happen in the ADC of Problem 11-24 if an analog voltage of  $V_A = 10.853$  V were applied to the input? What waveform would appear at the D/A output? Incorporate the necessary logic in this ADC

so that an "overscale" indication will be generated whenever  $V_A$  is too large.

- B 11-27\* An ADC has the following characteristics: resolution, 12 bits; fullscale error, 0.03% F.S.; full scale output, +5 V.
  - (a) What is the quantization error in volts?
  - (b) What is the total possible error in volts?
- **C**, **N** 11-28. The quantization error of an ADC such as the one in Figure 11-13 is always positive because the  $V_{AX}$  value must exceed  $V_A$  in order for the comparator output to switch states. This means that the value of  $V_{AX}$  could be as much as 1 LSB greater than  $V_A$ . This quantization error can be modified so that  $V_{AX}$  would be within  $\pm \frac{1}{2}$  LSB of  $V_A$ . This can be done by adding a fixed voltage equal to  $\frac{1}{2}$  LSB ( $\frac{1}{2}$  step) to the value of  $V_A$ . Figure 11-33 shows this symbolically for a converter that has a resolution of 10 mV/step. A fixed voltage of +5 mV is added to the D/A output in the summing amplifier, and the result,  $V_{AY}$ , is fed to the comparator, which has  $V_T = 1$  mV.

For this modified converter, determine the digital output for the following  $V_A$  values.

$$(a)^* V_A = 5.022 V$$

(b)  $V_A = 50.28 \,\mathrm{V}$ 

Determine the quantization error in each case by comparing  $V_{AX}$  and  $V_A$ . Note that the error is positive in one case and negative in the other.

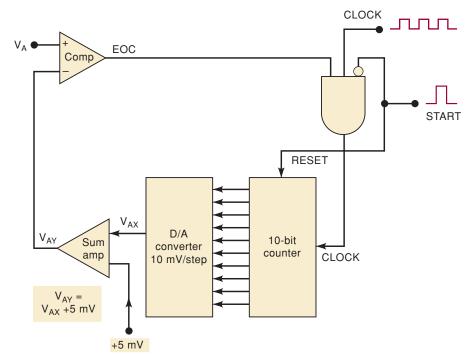
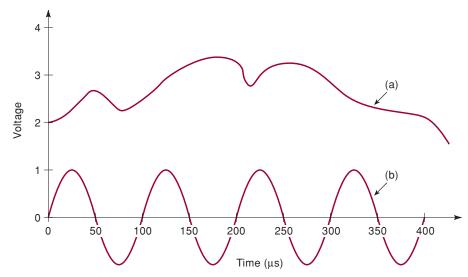


FIGURE 11-33 Problems 11-28 and 11-29.

**C** 11-29. For the ADC of Figure 11-33, determine the range of analog input values that will produce a digital output of 0100011100.

#### **SECTION 11-10**

**N** 11-30. Assume that the analog signal in Figure 11-34(a) is to be digitized by performing continuous A/D conversions using an eight-bit digital-ramp converter whose staircase rises at the rate of 1 V every 25  $\mu$ s. Sketch the reconstructed signal using the data obtained during the digitizing process. Compare it with the original signal, and discuss what could be done to make it a more accurate representation.



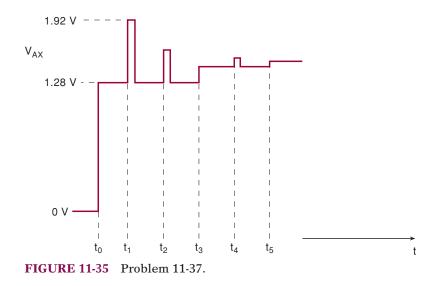
**FIGURE 11-34** Problems 11-30, 11-31, and 11-41.

- **C** 11-31\*On the sine wave of Figure 11-34(b), mark the points where samples are taken by a flash A/D converter at intervals of 75  $\mu$ s (starting at the origin). Then draw the reconstructed output from the D/A converter (connect the sample points with straight lines to show filtering). Calculate the sample frequency, the sine input frequency, and the difference between them. Then compare to the resulting reconstructed waveform frequency.
  - 11-32. A sampled data acquisition system is being used to digitize an audio signal. Assume the sample frequency  $F_S$  is 20 kHz. Determine the output frequency that will be heard for each of the following input frequencies.
    - (a)\* Input signal = 5 kHz
    - (b)\* Input signal = 10.1 kHz
    - (c) Input signal = 10.2 kHz
    - (d) Input signal = 15 kHz
    - (e) Input signal = 19.1 kHz
    - (f) Input signal = 19.2 kHz

#### SECTION 11-11

- B 11-33.\*DRILL QUESTION
  - Indicate whether each of the following statements refers to the digital-ramp ADC, the successive-approximation ADC, or both.

- (a) Produces a staircase signal at its DAC output
- (b) Has a constant conversion time independent of  $V_A$
- (c) Has a shorter average conversion time
- (d) Uses an analog comparator
- (e) Uses a DAC
- (f) Uses a counter
- (g) Has complex control logic
- (h) Has an  $\overline{EOC}$  output
- 11-34. Draw the waveform for  $V_{AX}$  as the SAC of Figure 11-19 converts  $V_A = 6.7$  V.
- 11-35. Repeat Problem 11-34 for  $V_A = 16$  V.
- **B** 11-36.\*A certain eight-bit successive-approximation converter has 2.55 V full scale. The conversion time for  $V_A = 1$  V is 80  $\mu$ s. What will be the conversion time for  $V_A = 1.5$  V?
  - 11-37. Figure 11-35 shows the waveform at  $V_{AX}$  for a six-bit SAC with a step size of 40 mV during a complete conversion cycle. Examine this waveform and describe what is occurring at times  $t_0$  to  $t_5$ . Then determine the resultant digital output.



- **B** 11-38.\*Refer to Figure 11-21. What is the approximate value of the analog input if the microcomputer's data bus is at 10010111 when  $\overline{RD}$  is pulsed LOW?
- **D** 11-39. Connect a 2.0-V reference source to  $V_{ref}/2$ , and repeat Problem 11-38.
- **C**, **D** 11-40\* Design the ADC interface to a digital thermostat using an LM34 temperature sensor and the ADC0804. Your system must measure accurately ( $\pm 0.2^{\circ}$ F) from 50 to 101°F. The LM34 puts out 0.01 V per degree F (0°F = 0 V).
  - (a) What should the digital value for 50°F be for the best resolution?
  - (b) What voltage must be applied to  $V_{in}(-)$ ?
  - (c) What is the full-scale range of voltage that will come in?
  - (d) What voltage must be applied to  $V_{ref}/2$ ?
  - (e) What binary value will represent 72°F?
  - (f) What is the resolution in °F? In volts?

#### **SECTION 11-12**

- **B** 11-41. Discuss how a flash ADC with a conversion time of  $1 \mu s$  would work for the situation of Problem 11-30.
- **D** 11-42. Draw the circuit diagram for a four-bit flash converter with BCD output and a resolution of 0.1 V. Assume that a +5 V precision supply voltage is available.

#### **DRILL QUESTION**

- B 11-43. For each of the following statements, indicate which type of ADC—digital-ramp, SAC, or flash—is being described.
  - (a) Fastest method of conversion
  - (b) Needs a START pulse
  - (c) Requires the most circuitry
  - (d) Does not use a DAC
  - (e) Generates a staircase signal
  - (f) Uses an analog comparator
  - (g) Has a relatively fixed conversion time independent of  $V_A$

#### SECTION 11-13

#### **B** 11-44. **DRILL QUESTION**

For each statement, indicate what type(s) of ADC is (are) being described.

- (a) Uses a counter that is never reset to 0
- (b) Uses a large number of comparators
- (c) Uses a VCO
- (d) Is used in noisy industrial environments
- (e) Uses a capacitor
- (f) Is relatively insensitive to temperature

#### **SECTIONS 11-14 AND 11-15**

- **T** 11-45\* Refer to the sample-and-hold circuit of Figure 11-24. What circuit fault would result in  $V_{\text{OUT}}$  looking exactly like  $V_A$ ? What fault would cause  $V_{\text{OUT}}$  to be stuck at 0?
- **C, D** 11-46. Use the CMOS 4016 IC (Section 8-16) to implement the switching in Figure 11-25, and design the necessary control logic so that each analog input is converted to its digital equivalent in sequence. The ADC is a 10-bit, successive-approximation type using a 50-kHz clock signal, and it requires a  $10-\mu$ s-duration start pulse to begin each conversion. The digital outputs are to remain stable for  $100 \ \mu$ s after the conversion is complete before switching to the next analog input. Choose an appropriate multiplexing clock frequency.

#### **MICROCOMPUTER APPLICATION**

**C**, **N**, **D** 11-47\* Figure 11-21 shows how the ADC0804 is interfaced to a microcomputer. It shows three control signals,  $\overline{CS}$ ,  $\overline{RD}$ , and  $\overline{WR}$ , that come from the microcomputer to the ADC. These signals are used to start each new A/D conversion and to read (transfer) the ADC data output into the microcomputer over the data bus.

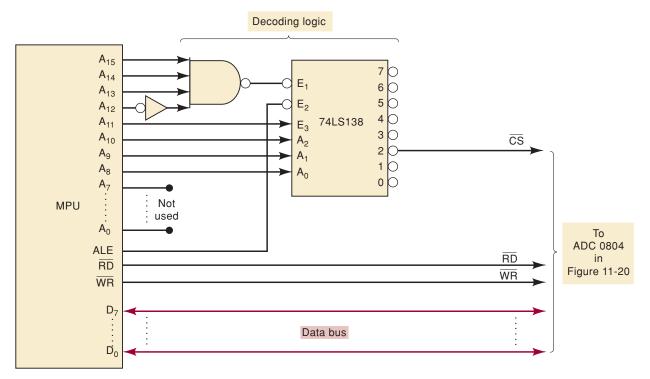


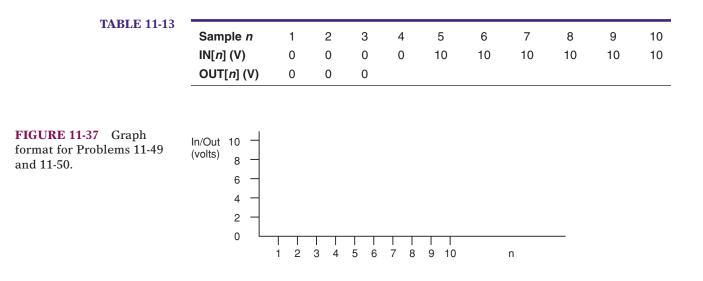
FIGURE 11-36 Problem 11-47: MPU interfaced to the ADC0804 of Figure 11-20.

Figure 11-36 shows one way the address decoding logic could be implemented. The  $\overline{CS}$  signal that activates the ADC0804 is developed from the eight high-order address lines of the MPU address bus. Whenever the MPU wants to communicate with the ADC0804, it places the address of the ADC0804 onto the address bus, and the decoding logic drives the  $\overline{CS}$  signal LOW. Notice that in addition to the address lines, a timing and control signal (*ALE*) is connected to the  $\overline{E_2}$  enable input. Whenever *ALE* is HIGH, it means that the address is potentially in transition, so the decoder should be disabled until *ALE* goes LOW (at which time the address will be valid and stable). This serves a purpose for timing but has no effect on the address of the ADC.

- (a) Determine the address of the ADC0804.
- (b) Modify the diagram of Figure 11-36 to place the ADC0804 at address E8XX hex.
- (c) Modify the diagram of Figure 11-36 to place the ADC0804 at address FFXX hex.
- D 11-48. You have available a 10-bit SAC A/D converter (AD 573), but your system requires only eight bits of resolution and you have only eight port bits available on your microprocessor. Can you use this A/D converter, and if so, which of the 10 data lines will you attach to the port?

#### **SECTION 11-17**

11-49. The data in Table 11-13 are input samples taken by an A/D converter. Notice that if the input data were plotted, it would represent a simple step function like the rising edge of a digital signal. Calculate the simple average of the four most recent data points, starting with OUT[4] and proceeding through OUT[10]. Plot the values for IN and OUT against the sample number n as shown in Figure 11-37.



Sample calculations:

OUT[n] = (IN[n -	3] + IN[ $n - 2$	2] + IN[n - ]	1] + IN[n])/4 = 0
OUT[4] = (IN[1])	+ IN[2]	+ IN[3]	+ IN[4])/4 = 0
OUT[5] = (IN[2])	+ IN[3]	+ IN[4]	+ IN[5])/4 = 2.5

(Notice that this calculation is equivalent to multiplying each sample by  $\frac{1}{4}$  and summing.)

11-50. Repeat the previous problem using a weighted average of the last four samples. The weights in this case are placing greater emphasis on recent samples and less emphasis on older samples. Use the weights 0.1, 0.2, 0.3, and 0.4.

 $\begin{aligned} & \text{OUT}[n] = 0.1(\text{IN}[n-3]) + 0.2(\text{IN}[n-2]) + 0.3(\text{IN}[n-1]) + 0.4(\text{IN}[n]) \\ & \text{OUT}[5] = 0.1(\text{IN}[2]) + 0.2(\text{IN}[3] + 0.3(\text{IN}[4] + 0.4(\text{IN}[5]) = 4 \end{aligned}$ 

11-51. What does the term MAC stand for?

#### 11-52.\*DRILL QUESTIONS

True or false:

- (a) A digital signal is a continuously changing voltage.
- (b) A digital signal is a sequence of numbers that represent an analog signal.

When processing an analog signal, the output may be distorted due to:

- (a) Quantization error when converting analog to digital
- (b) Not sampling the original signal frequently enough
- (c) Temperature variation in the processor components
- (d) The high-frequency components associated with sudden changes in voltage out of the DAC
- (e) Electrical noise on the power supply
- (f) Alias signals introduced by the digital system

## ANSWERS TO SECTION REVIEW QUESTIONS

#### **SECTION 11-1**

1. Converts a nonelectrical physical quantity to an electrical quantity2. Convertsan analog voltage or current to a digital representation3. Stores it; performscalculation or some other operation on it4. Converts digital data to theiranalog representation5. Controls a physical variable according to an electricalinput signal

#### **SECTION 11-2**

1. 40  $\mu$ A; 10.2 mA 2. 5.12 mA 3. 0.39 percent 4. 4096 5. 12 6. True 7. It produces a greater number of possible analog outputs between 0 and full scale.

#### SECTION 11-3

1. It uses only two different sizes of resistors. 2. 640 k $\Omega$  3. 0.5 V 4. Increases by 20 percent

#### **SECTION 11-4**

Maximum deviation of DAC output from its ideal value, expressed as percentage of full scale
 Time it takes DAC output to settle to within <sup>1</sup>/<sub>2</sub> step size of its full-scale value when the digital input changes from 0 to full scale
 Offset error adds a small constant positive or negative value to the expected analog output for any digital input.
 Because of the response time of the op-amp current-to-voltage converter

#### **SECTION 11-8**

1. Tells control logic when the DAC output exceeds the analog input2. At outputs of register3. Tells us when conversion is complete and digital equivalentof  $V_A$  is at register outputs

#### SECTION 11-9

1. The digital input to a DAC is incremented until the DAC staircase output exceeds the analog input. 2. The built-in error caused by the fact that  $V_{AX}$  does not continuously increase but goes up in steps equal to the DAC's resolution. The final  $V_{AX}$ can be different from  $V_A$  by as much as one step size. 3. If  $V_A$  increases, it will take more steps before  $V_{AX}$  can reach the step that first exceeds  $V_A$ . 4. True 5. Simple circuit; relatively long conversion time that changes with  $V_A$ 6. 0010000111<sub>2</sub> = 135<sub>10</sub> for both cases

#### **SECTION 11-10**

 Process of converting different points on an analog signal to digital and storing the digital data for later use
 Computer generates START signal to begin an A/D conversion of the analog signal. When EOC goes LOW, it signals the computer that the conversion is complete. The computer then loads the ADC output into memory. The process is repeated for the next point on the analog signal.
 Twice the highest frequency in the input signal
 An alias frequency will be present in the output.

#### **SECTION 11-11**

1. The SAC has a shorter conversion time that doesn't change with  $V_A$ . 2. It has more complex control logic. 3. False 4. (a) 8 (b) 0–5 V (c)  $\overline{CS}$  controls the effect of the  $\overline{RD}$  and  $\overline{WR}$  signals;  $\overline{WR}$  is used to start a new conversion;  $\overline{RD}$  enables the output buffers. (d) When LOW, it signals the end of a conversion. (e) It separates the usually noisy digital ground from the analog ground so as not to contaminate the analog input signal. (f) All analog voltages on  $V_{in}(+)$  are measured with reference to this pin. This allows the input range to be offset from ground.

#### **SECTION 11-12**

1. True 2. 4095 comparators and 4096 resistors 3. Major advantage is its conversion speed; disadvantage is the number of required circuit components for a practical resolution.

#### **SECTION 11-13**

1. It reduces the conversion time by using an up/down counter that allows  $V_{AX}$  to<br/>track  $V_A$  without starting from 0 for each conversion.2. A VCO3. Advantages: low cost, temperature immunity; disadvantage: slow conversion time<br/>4. Flash ADC, voltage-to-frequency ADC, and dual-slope ADC5. One

#### **SECTION 11-14**

It takes a sample of an analog voltage signal and stores it on a capacitor.
 False; they are unity-gain buffers with high input impedance and low output impedance.

#### **SECTION 11-15**

1. Uses a single ADC 2. It would become a MOD-8 counter.

#### **SECTION 11-16**

1. Digitized waveforms are stored in the memory block. 2. The ADC digitizes the points on the input waveform for storage in memory; the vertical DAC converts the stored data points back to analog voltages to produce the vertical deflection of the electron beam; the horizontal DAC produces a staircase sweep voltage that provides the horizontal deflection of the electron beam.

#### **SECTION 11-17**

1. Filtering analog signals2. An A/D converter3. To change their dynamicresponse, you simply change the numbers in the software program, not the hardwarecomponents.4. The Multiply and Accumulate (MAC) unit5. 3; 7