Sheikh Hasina University, Netrokona
Department of Computer Science and Engineering
CSE-2205: Introduction to Mechatronics

Lec-4: DAC and ADC

Md. Ariful Islam
Assistant Professor
Dept. of Robotics and Mechatronics Engineering
University of Dhaka
&
Adjunct Faculty
Sheikh Hasina University, Netrokona
Department of Computer Science and Engineering

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 or true or false.

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*.

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.

$$0\text{ V to }0.8\text{ V} = \text{logic }0$$
$$2\text{ V to }5\text{ V} = \text{logic }1$$

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.

Most physical variables are analog in nature and can take on any value within a continuous range of values.

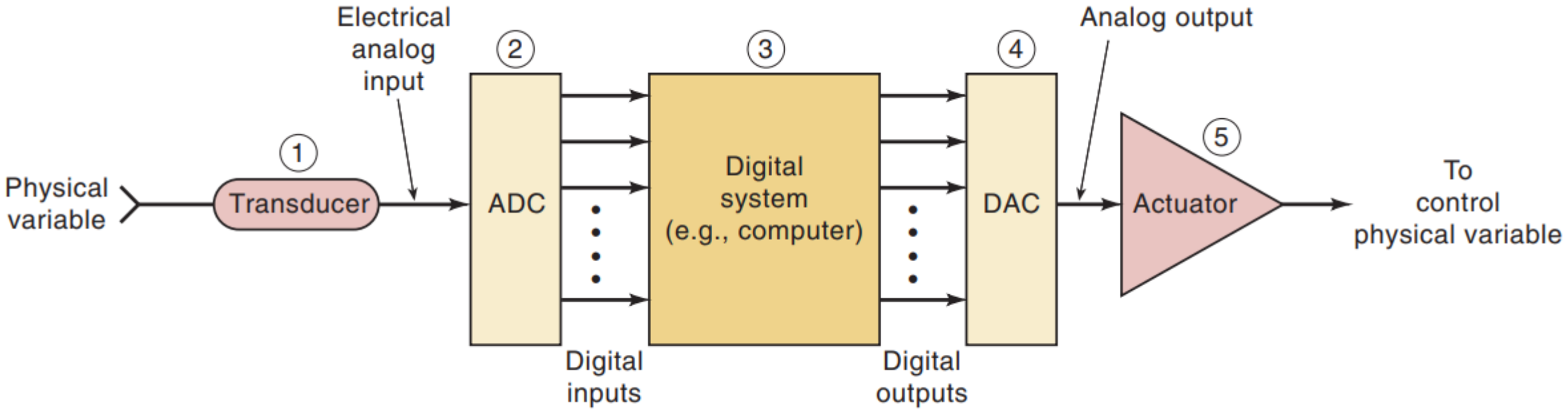
Examples include temperature, pressure, light intensity, audio signals, position, rotational speed, and flow rate.

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.

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:



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.

Analog-to-digital converter (ADC): The transducer's electrical analog output serves as the analog input to the analog-to-digital converter (ADC).

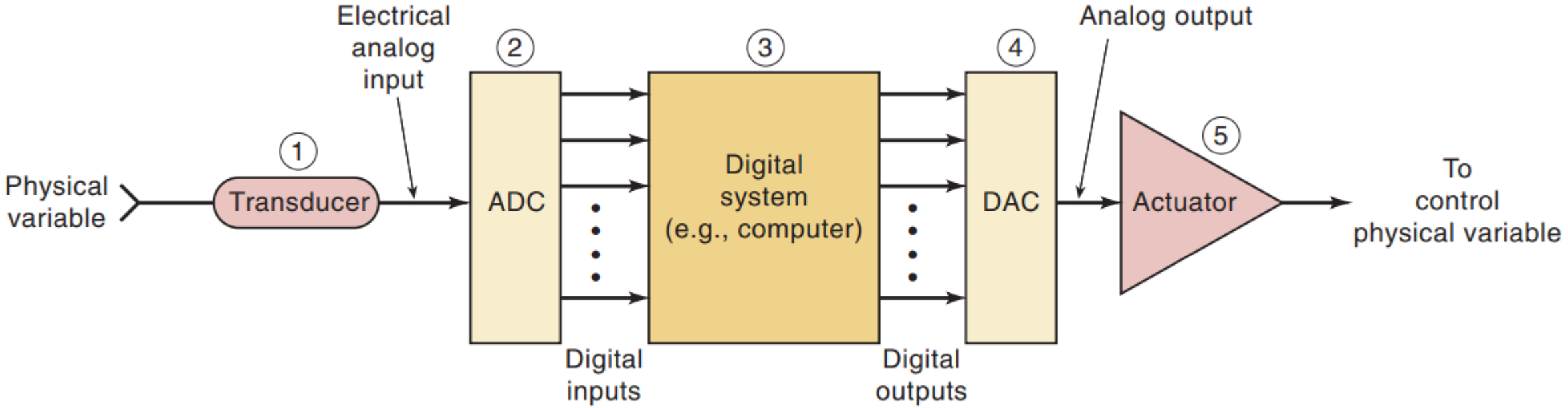
The ADC converts this analog input to a digital output. This digital output consists of a number of bits that represent the value of the analog input.

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.

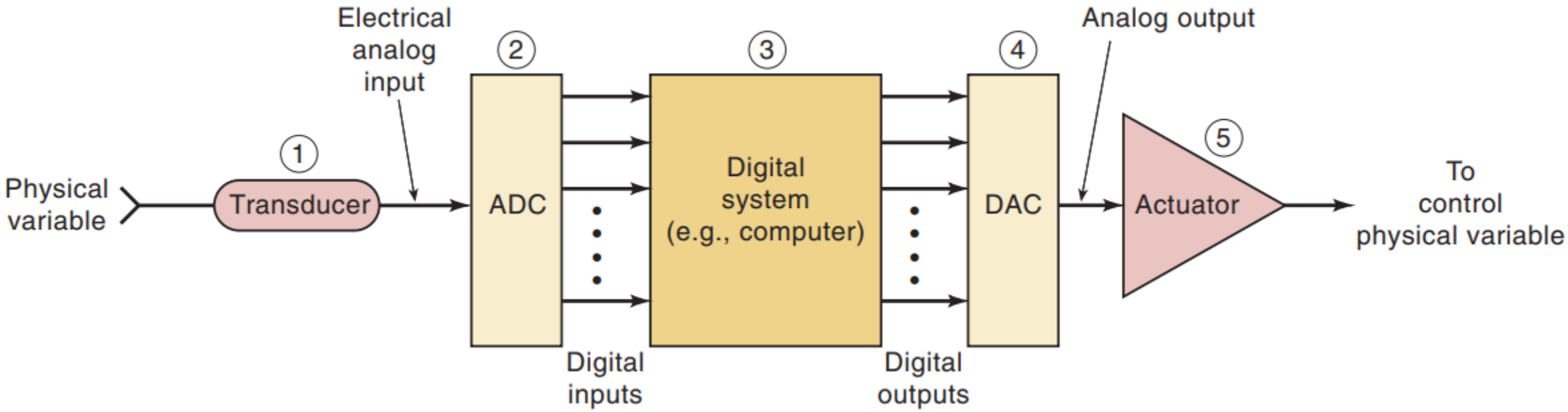
Computer: The digital representation of the process variable is transmitted from the ADC to the digital computer, which stores the digital value and processes it *according to a program of instructions* that it is executing.

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.



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.

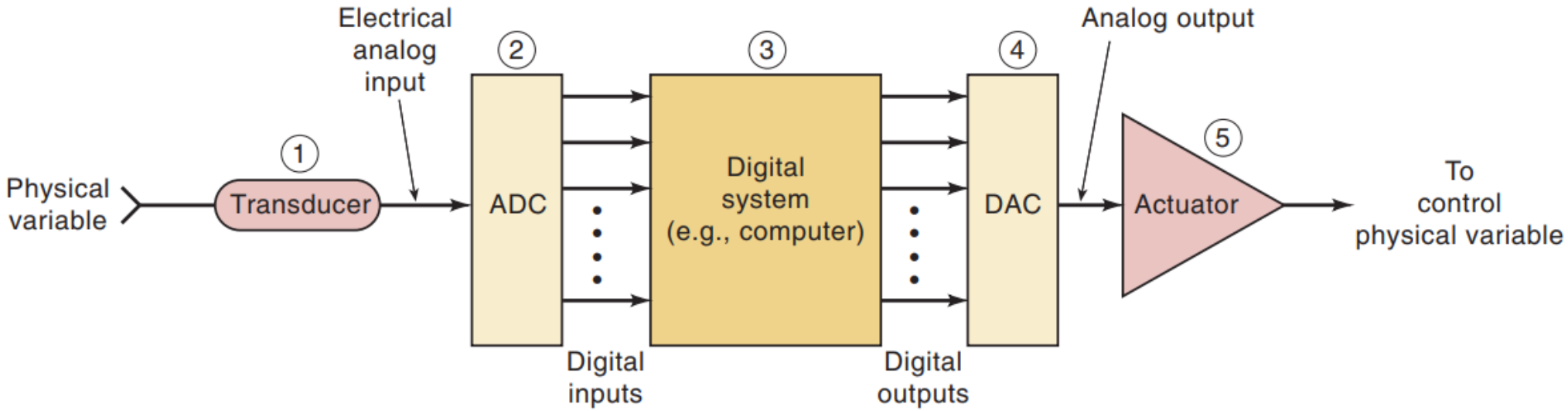
For example, the computer might produce a digital output ranging from 00000000 to 11111111, which the DAC converts to a voltage ranging from 0 to 10 V.



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.

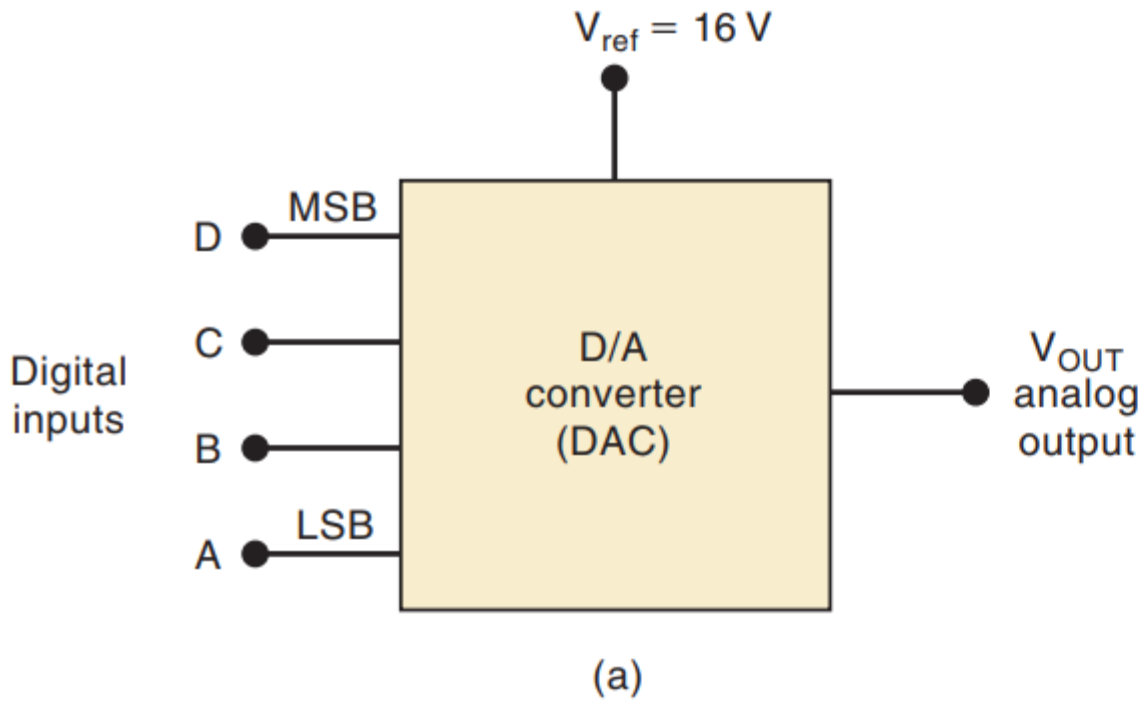
For our water temperature example, the actuator might be an electrically controlled valve that regulates the flow of hot water into the tank in accordance with the analog voltage from the DAC.

The flow rate would vary in proportion to this analog voltage, with 0 V producing no flow and 10 V producing the maximum flow.



DIGITAL-TO-ANALOG CONVERSION

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.



(a)

D	C	B	A	V_{OUT}	
0	0	0	0	0	Volts
0	0	0	1	1	
0	0	1	0	2	
0	0	1	1	3	
0	1	0	0	4	
0	1	0	1	5	
0	1	1	0	6	
0	1	1	1	7	
1	0	0	0	8	
1	0	0	1	9	
1	0	1	0	10	
1	0	1	1	11	
1	1	0	0	12	
1	1	0	1	13	
1	1	1	0	14	
1	1	1	1	15	Volts

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.

For each input number, the D/A converter output voltage is a unique value.

$$\text{analog output} = K \times \text{digital input}$$

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 $I_{OUT} = 10\,\text{mA}$ 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:

$$\begin{aligned} I_{OUT} &= (0.5\,\text{mA}) \times 29 \\ &= 14.5\,\text{mA} \end{aligned}$$

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

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

Therefore,

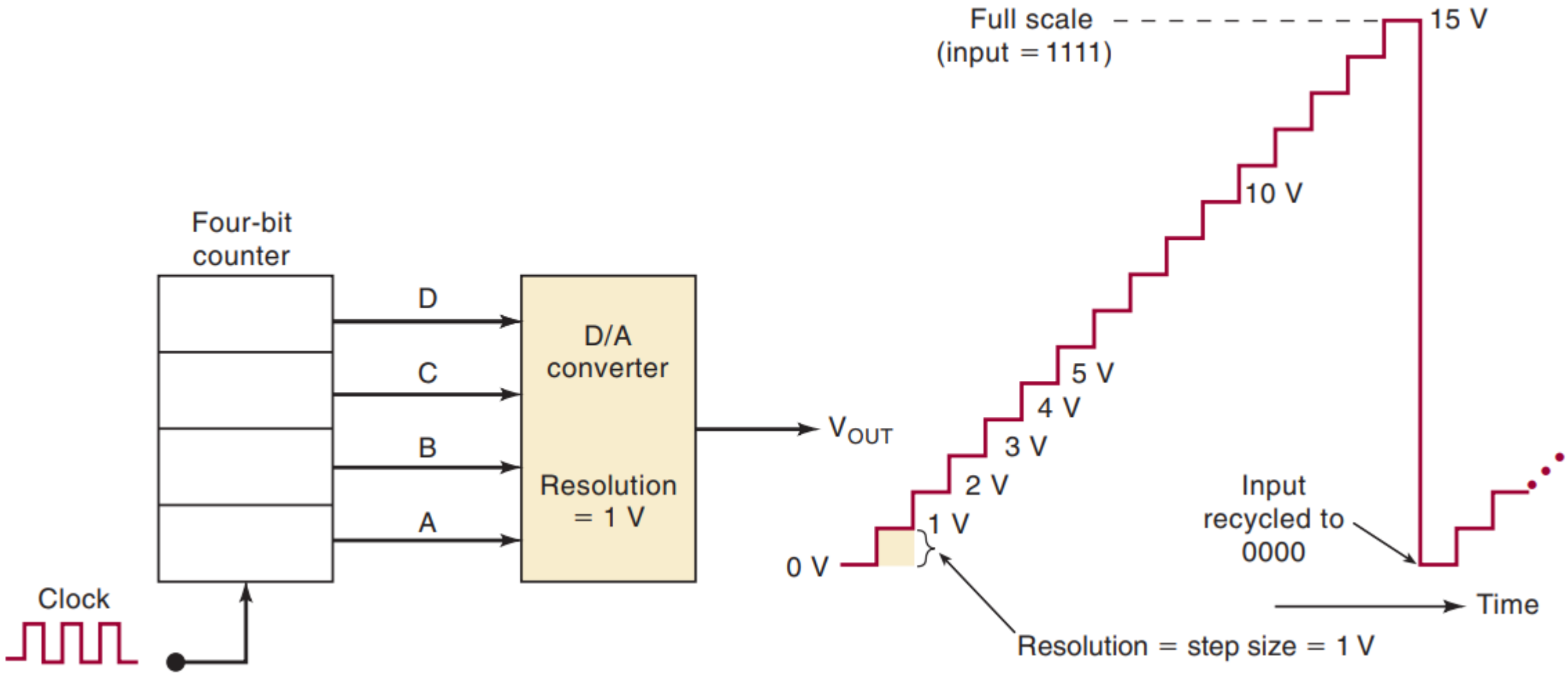
$$K = 20\,\text{mV}$$

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

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

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.



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.

$$\text{resolution} = K = \frac{A_{\text{fs}}}{(2^N - 1)}$$

where A_{fs} is the analog full-scale output and N is the number of bits.

$$\%\,\text{resolution} = \frac{\text{step size}}{\text{full scale (F.S.)}} \times 100\%$$

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

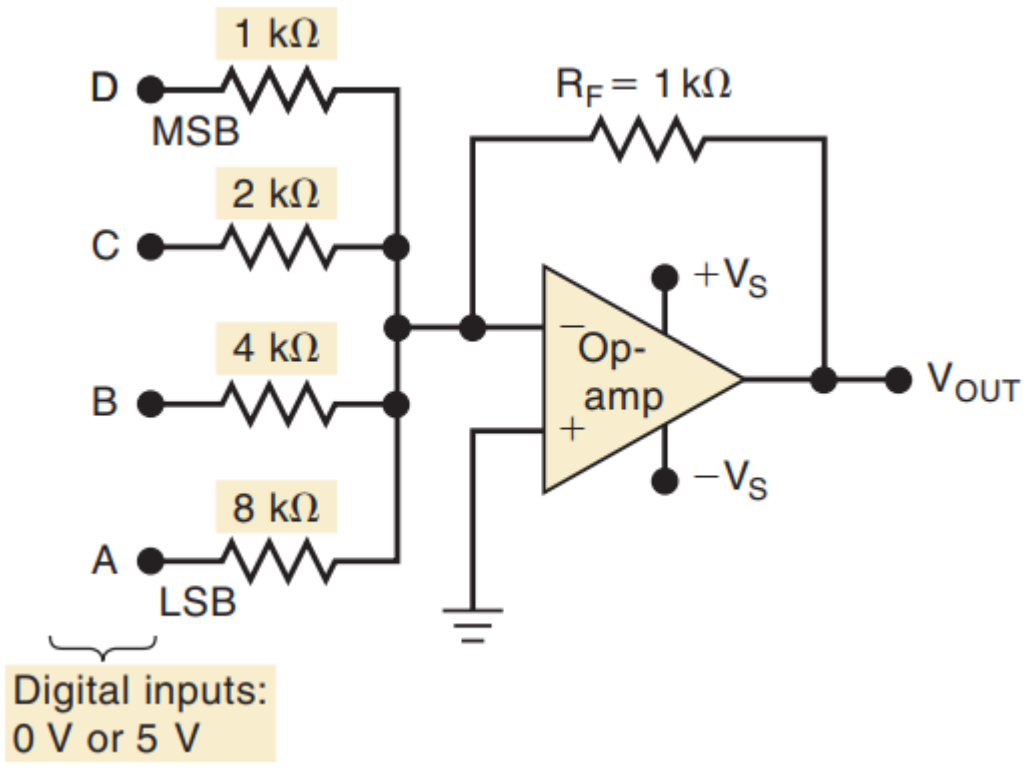
Solution

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

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

DAC CIRCUITRY

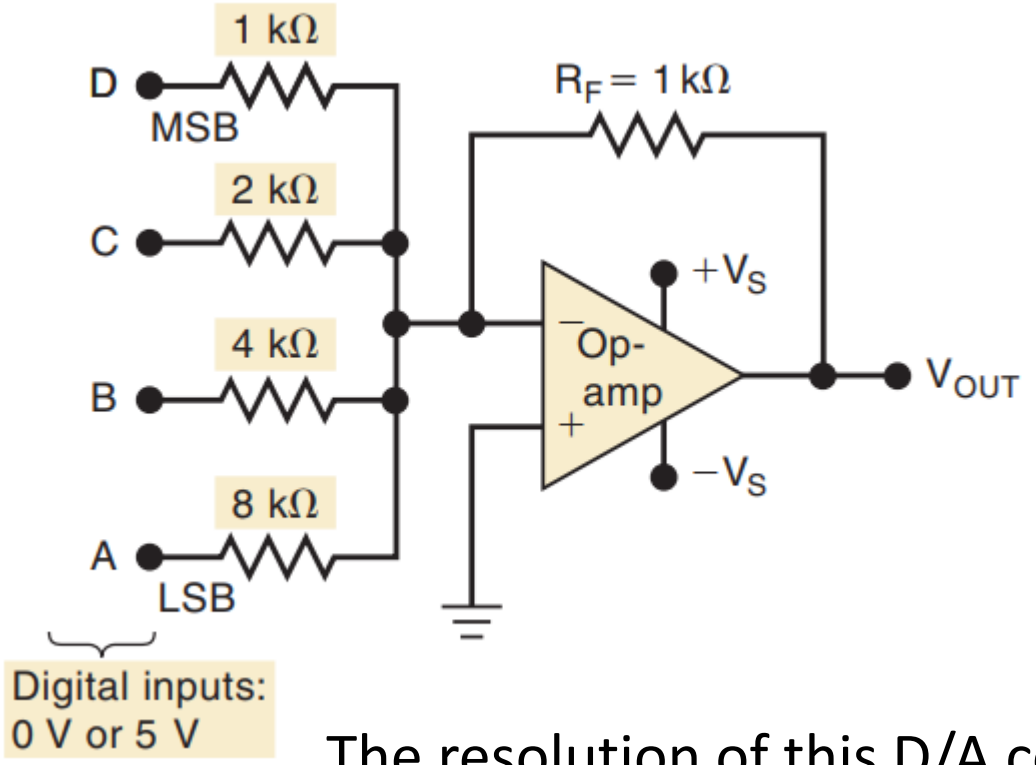
Binary weighted resistors



Input code				
D	C	B	A	V_{OUT} (volts)
0	0	0	0	0
0	0	0	1	−0.625 ← LSB
0	0	1	0	−1.250
0	0	1	1	−1.875
0	1	0	0	−2.500
0	1	0	1	−3.125
0	1	1	0	−3.750
0	1	1	1	−4.375
1	0	0	0	−5.000
1	0	0	1	−5.625
1	0	1	0	−6.250
1	0	1	1	−6.875
1	1	0	0	−7.500
1	1	0	1	−8.125
1	1	1	0	−8.750
1	1	1	1	−9.375 ← Full-scale

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, which produces the weighted sum of these input voltages.



$$V_{OUT} = -(V_D + \tfrac{1}{2}V_C + \tfrac{1}{4}V_B + \tfrac{1}{8}V_A)$$

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.

$$\begin{aligned} V_{OUT} &= -(5\text{ V} + 0\text{ V} + \tfrac{1}{4} \times 5\text{ V} + 0\text{ V}) \\ &= -6.25\text{ V} \end{aligned}$$

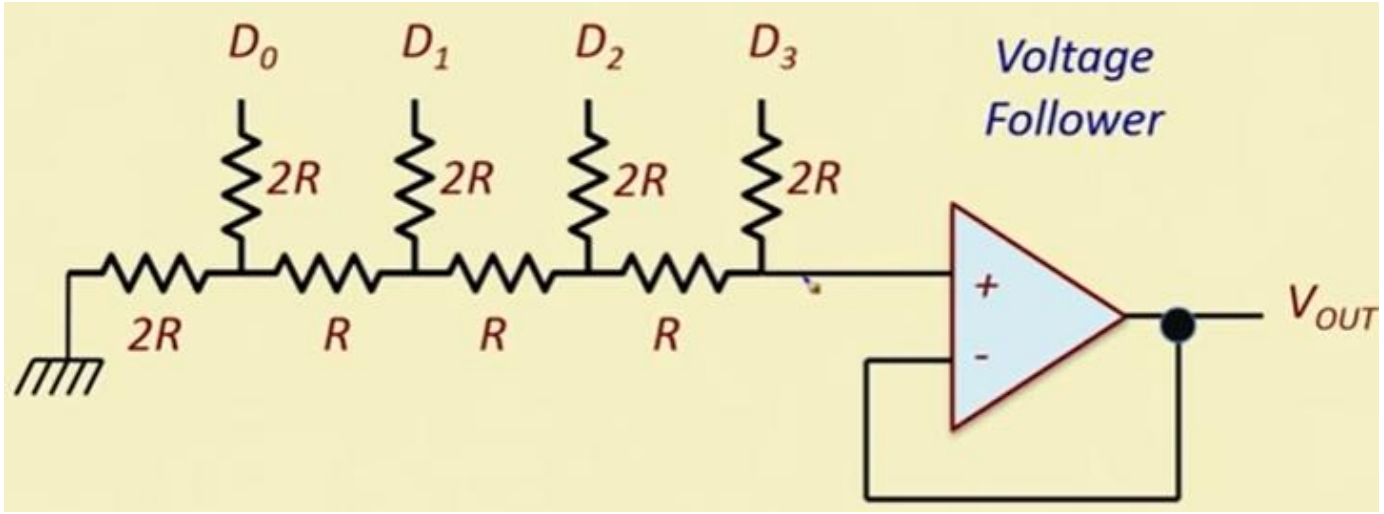
The resolution of this D/A converter is equal to the weighting of the LSB, which is 1/8 * 5 V = 0.625 V.

R/2R Ladder

The biggest problem of binary weighted method 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 kΩ in a 12-bit DAC, the LSB resistor will be over 2 MΩ.

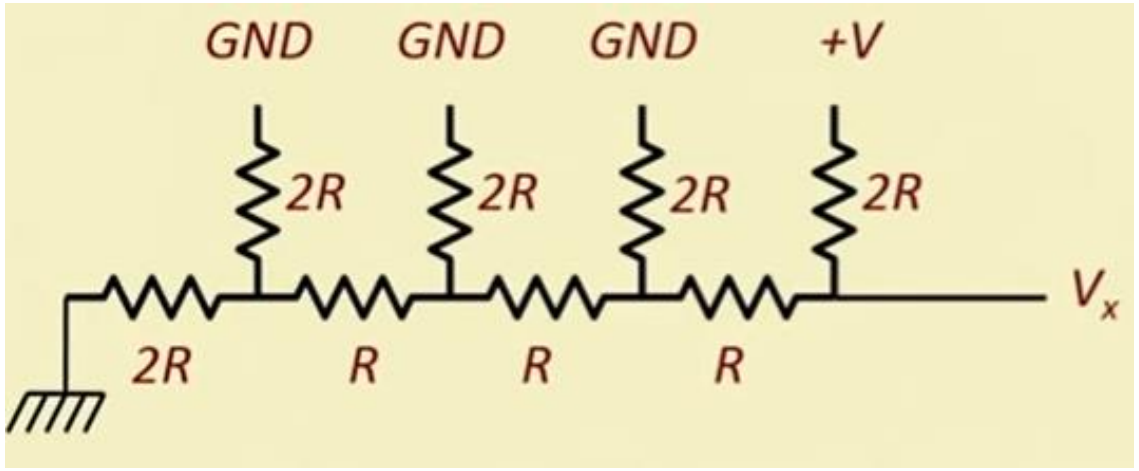
One of the most widely used DAC circuits that satisfies this requirement is the R/2R ladder network, where the resistance values span a range of only 2 to 1. One such DAC is shown in Figure.

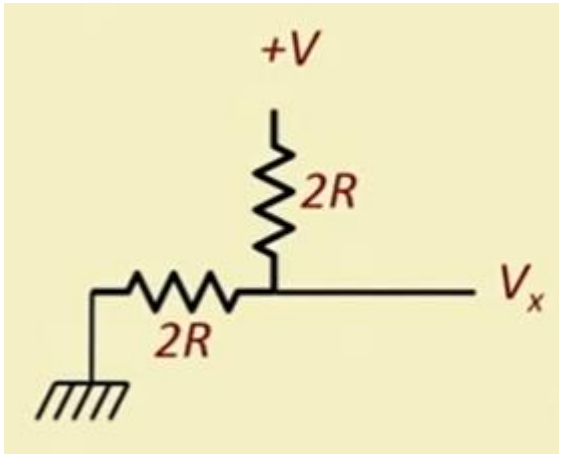
Note how the resistors are arranged, and especially note that only two different values are used, R and 2R.



Let us calculate the voltages at the op-amp input when exactly one of the Di inputs is at 1 (say, +V volts)

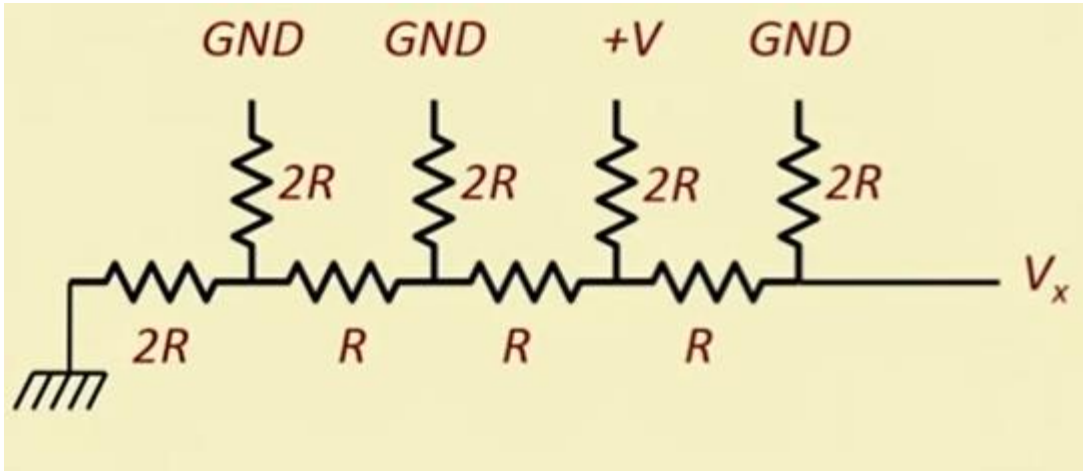
Case 1: Input is 1000

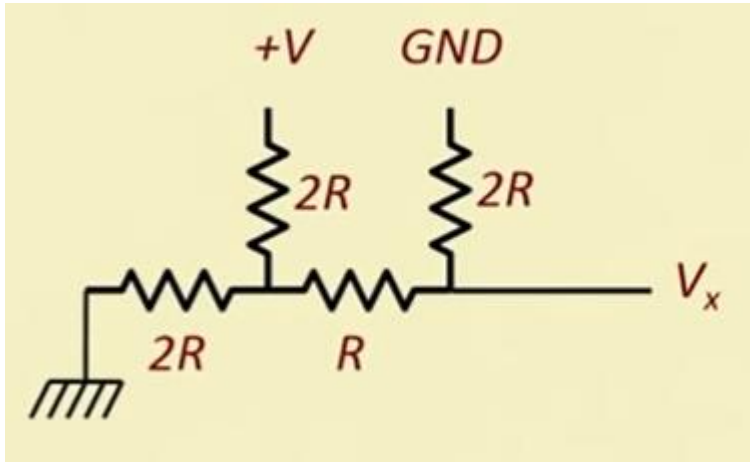


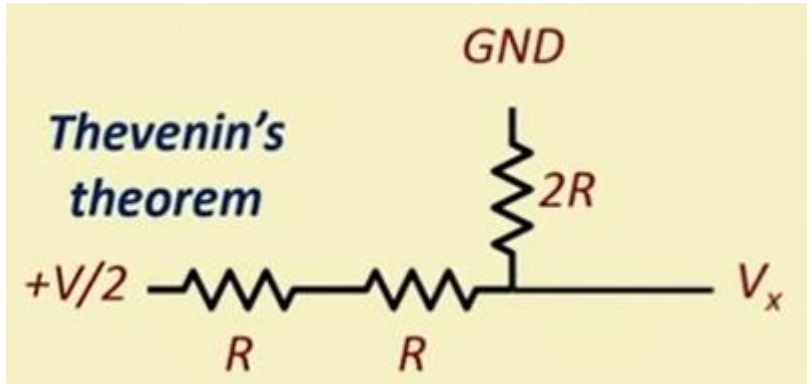


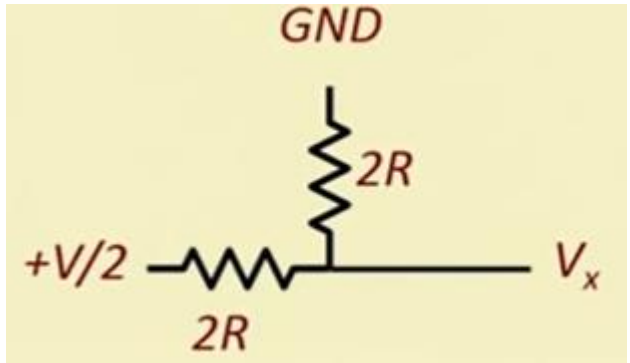
$$V_x = V / 2$$

Case 2: Input is 0100



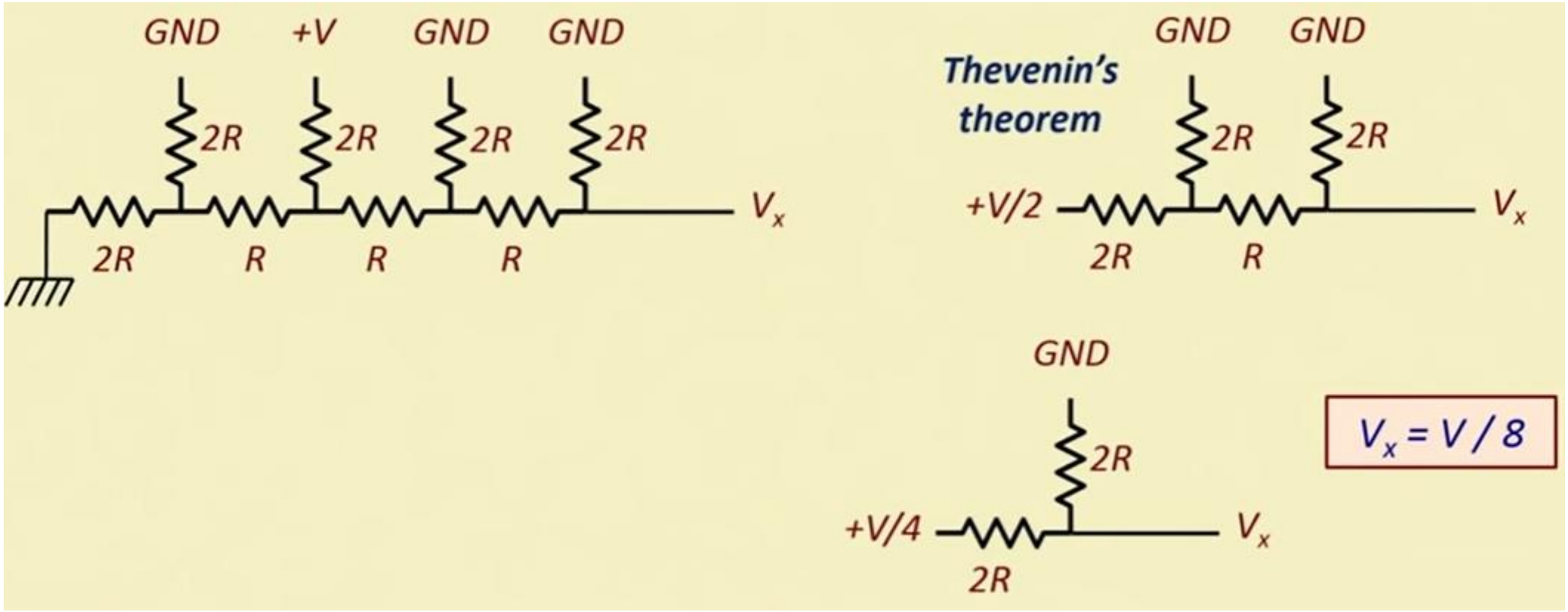




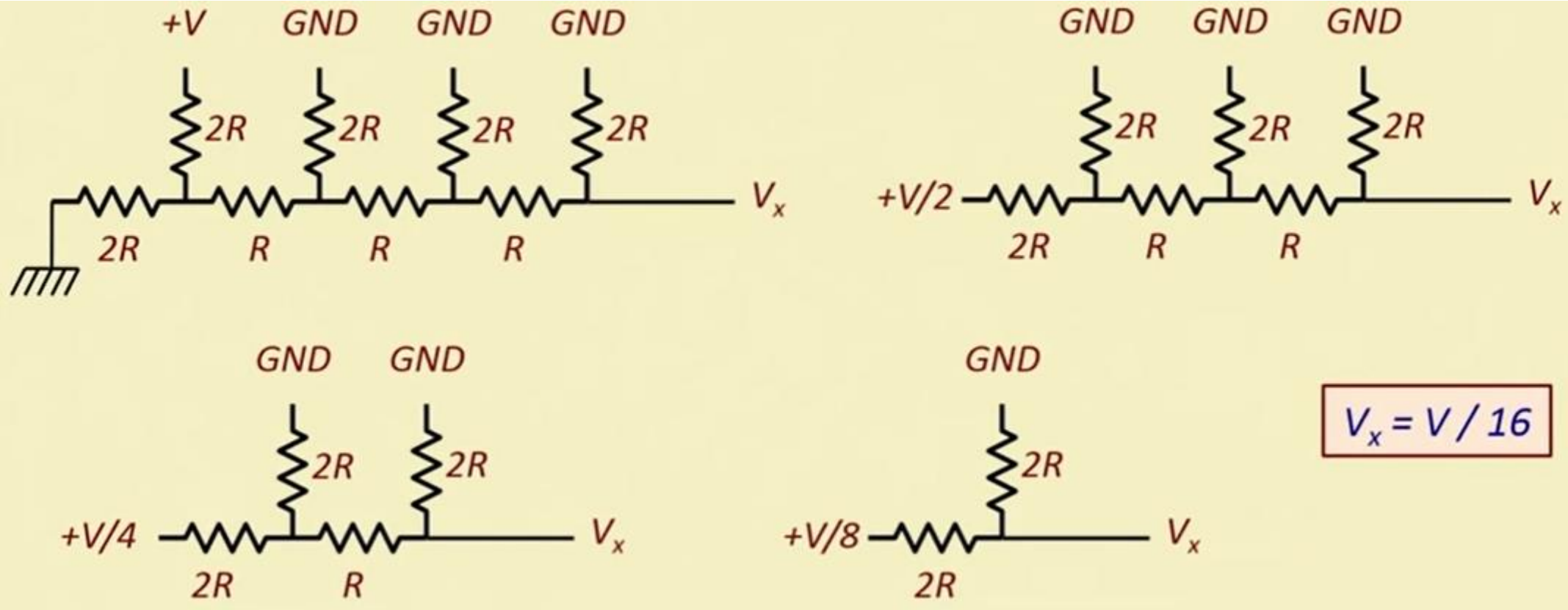


$V_x = V / 4$

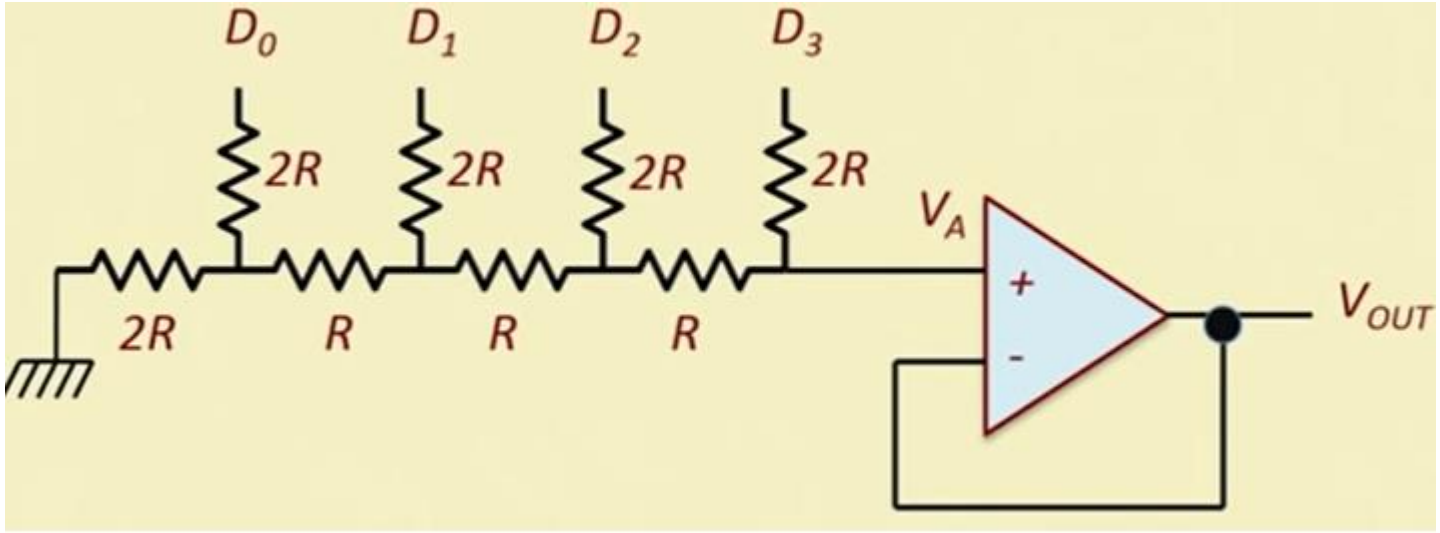
Case 2: Input is 0010



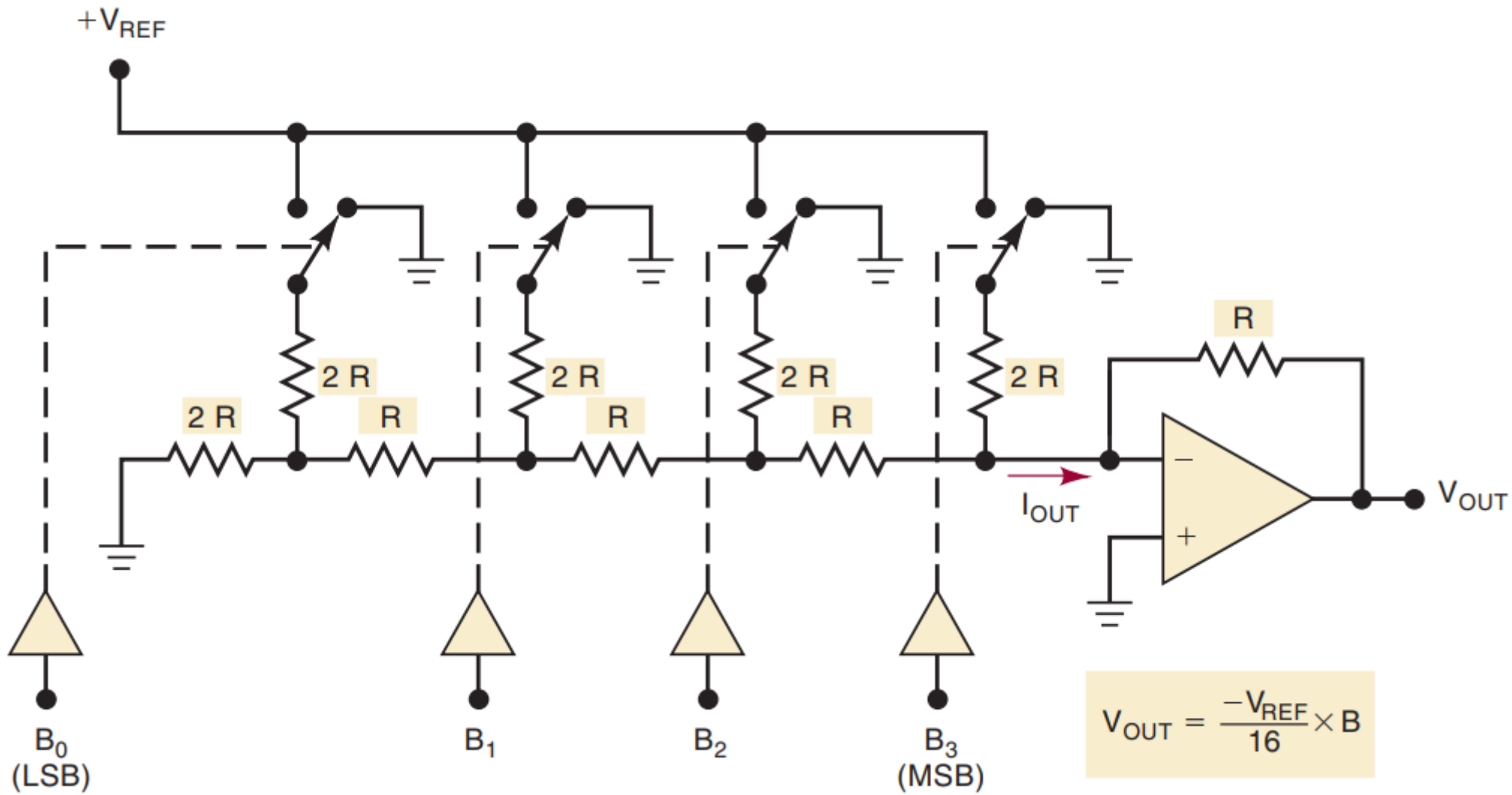
Case 2: Input is 0001



When all the four inputs D=$D_3D_2D_1D_0$ are applied (where D_i=GND or +V volts), we can apply the principle of superposition to compute the final output voltage V_A



$$V_A = [D_3.(V/2) + D_2.(V/4) + D_1.(V/8) + D_0.(V/16)]$$
$$= [8D_3 + 4D_2 + 2D_1 + D_0].(V/16)$$
$$= D.(V/16)$$


+VREF
2 R
R
2 R
R
2 R
R
2 R
2 R
R
IOUT
VOUT
B0
(LSB)
B1
B2
B3
(MSB)
VOUT = −VREF/16 × B

Assume that $V_{REF} = 10\,V$ 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):

$$\begin{aligned} \text{resolution} &= \frac{-10\text{ V} \times 1}{16} \\ &= -0.625\text{ V} \end{aligned}$$

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

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

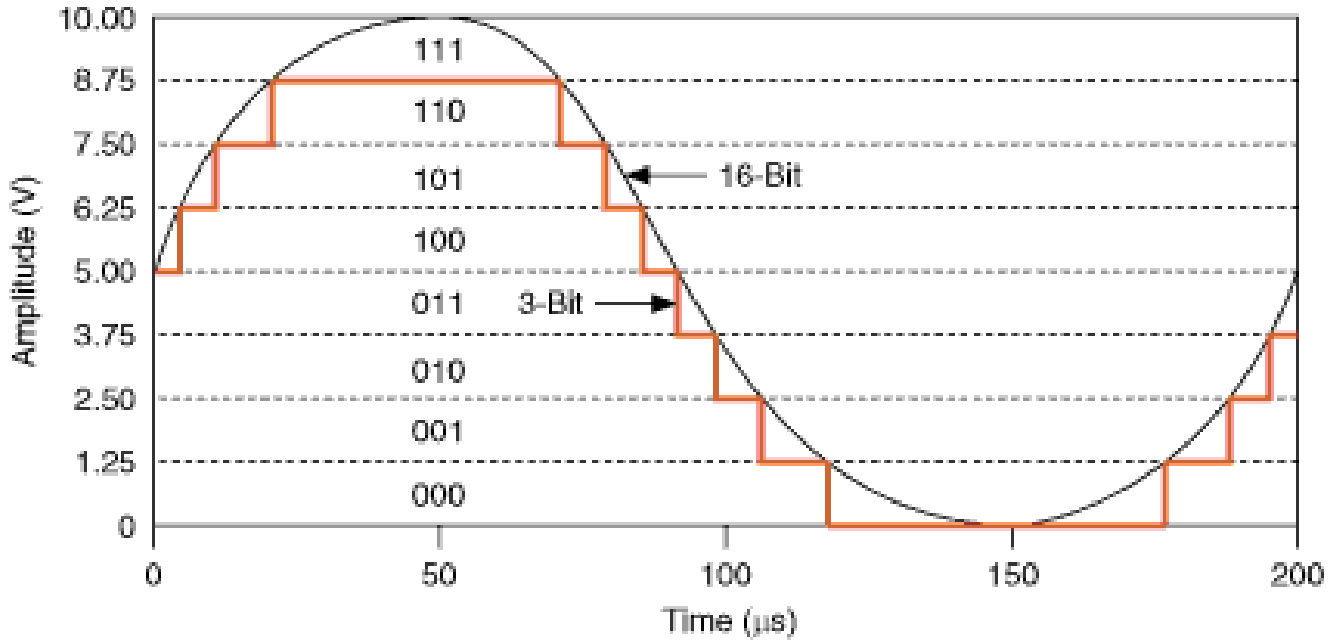
DAC SPECIFICATIONS

Resolution

The percentage resolution of a DAC depends solely on the number of bits.

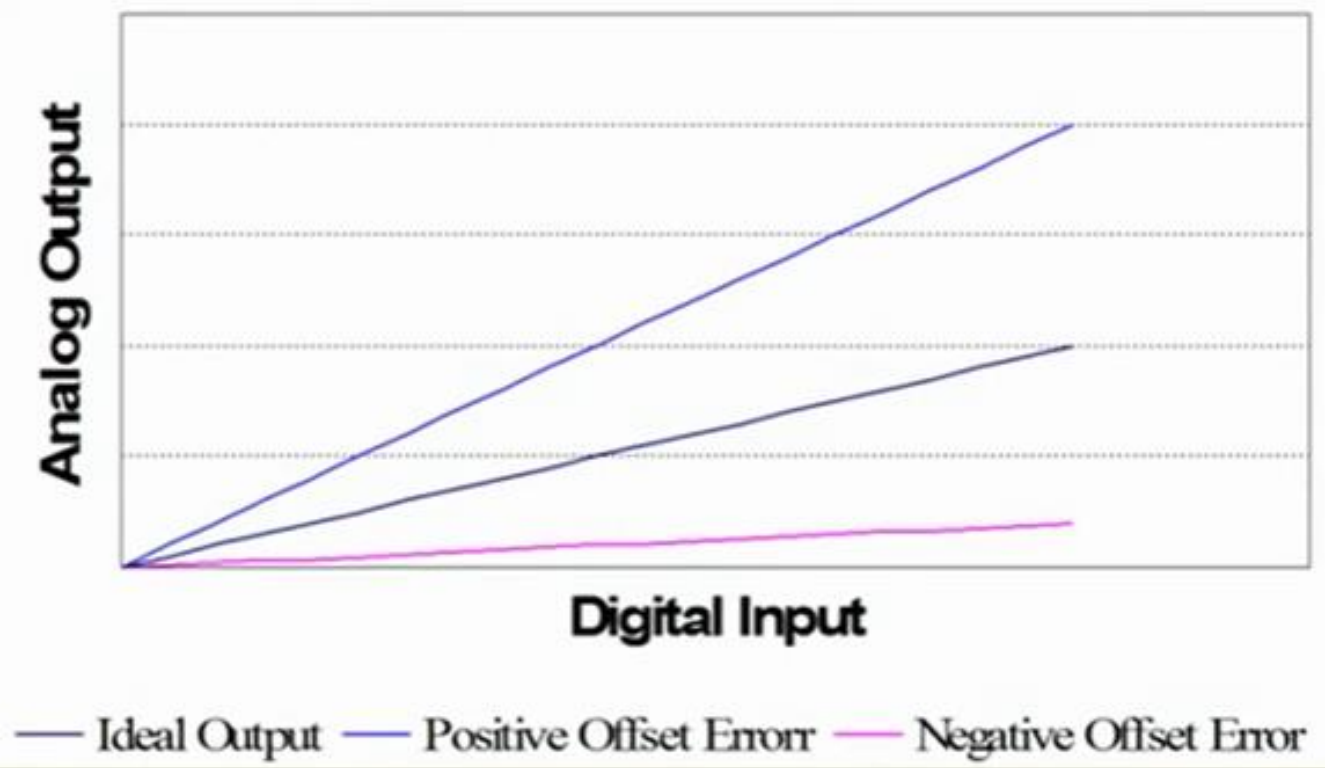
For this reason, manufacturers usually specify a DAC resolution as the number of bits.

A 10-bit DAC has a finer (smaller) resolution than an eight-bit DAC.



Gain Error

Occurs when the slope of the actual output deviates from the slope of the ideal output.



The slope of this straight line will change, if the value of K changes. This is called gain error

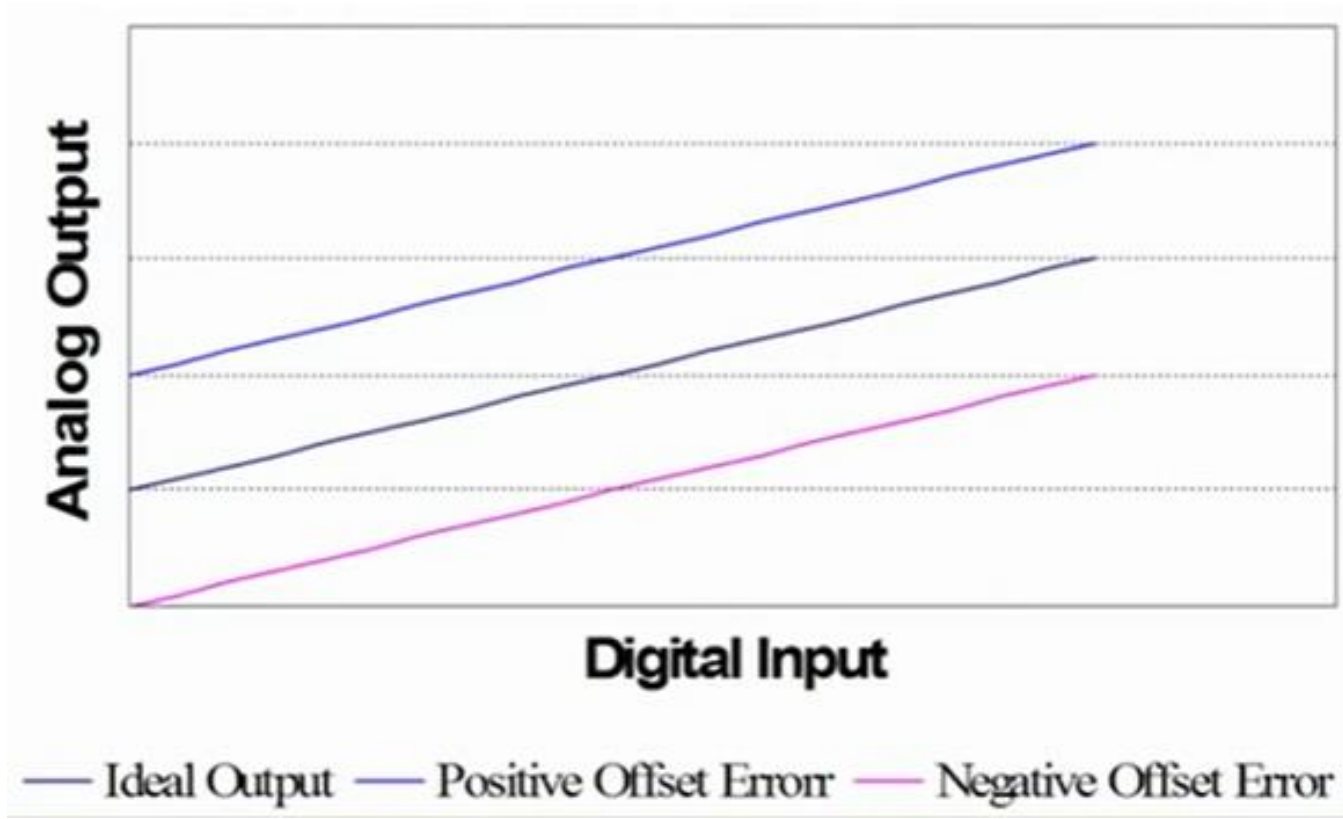
Offset Error

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.

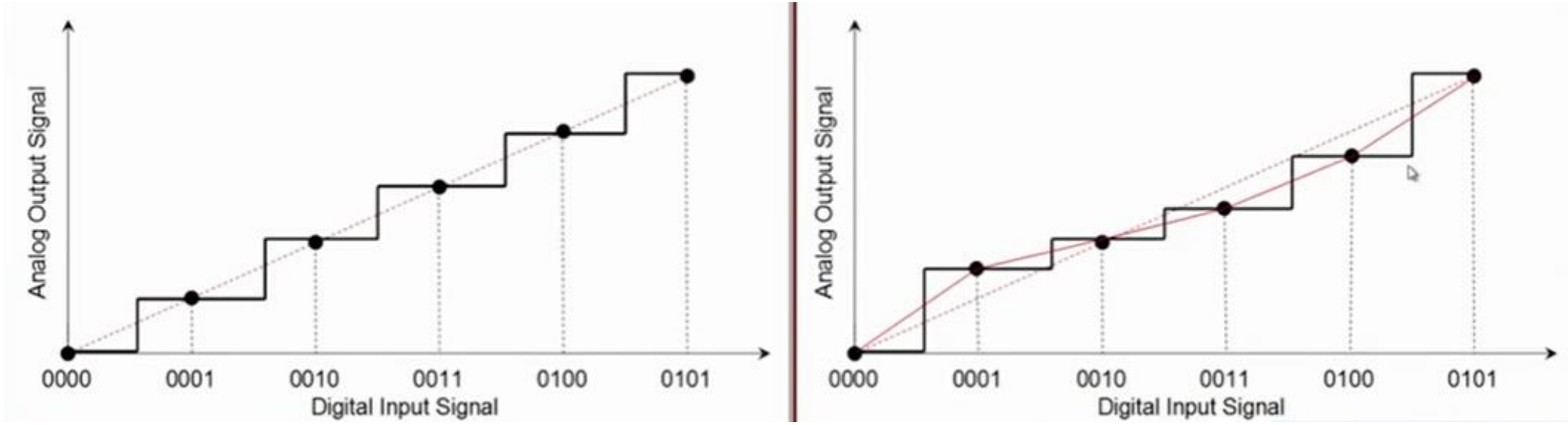
Input Code	Ideal Output (mV)	Actual Output (mV)
0000	0	2
0001	100	102
1000	800	802
1111	1500	1502


Analog Output
Digital Input
Ideal Output
Positive Offset Errorr
Negative Offset Error

Non-linearity Error

Occurs when analog output of signal is non-linear.

All step heights are exactly same, but DA converters can be designed using resistances. Now, if the value of some resistances changes due to fabrication defect or because of some other error design error. So, what might happen these step heights may become different.

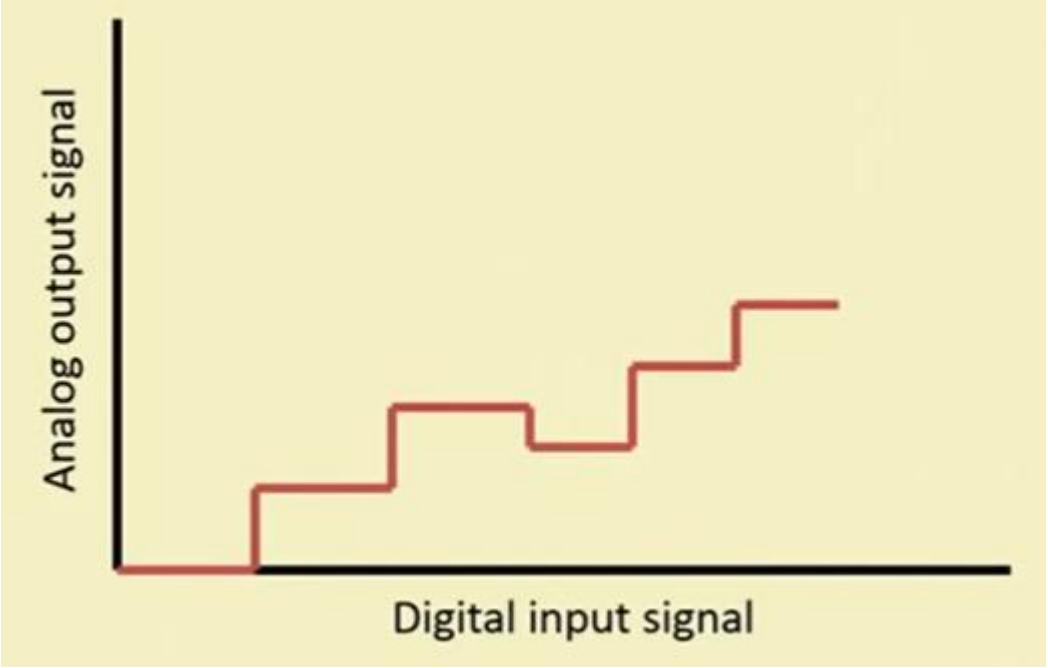


Linearity **Non-Linearity**

Monotonicity Error

Occurs when an increase in digital input results in a decrease in analog output.

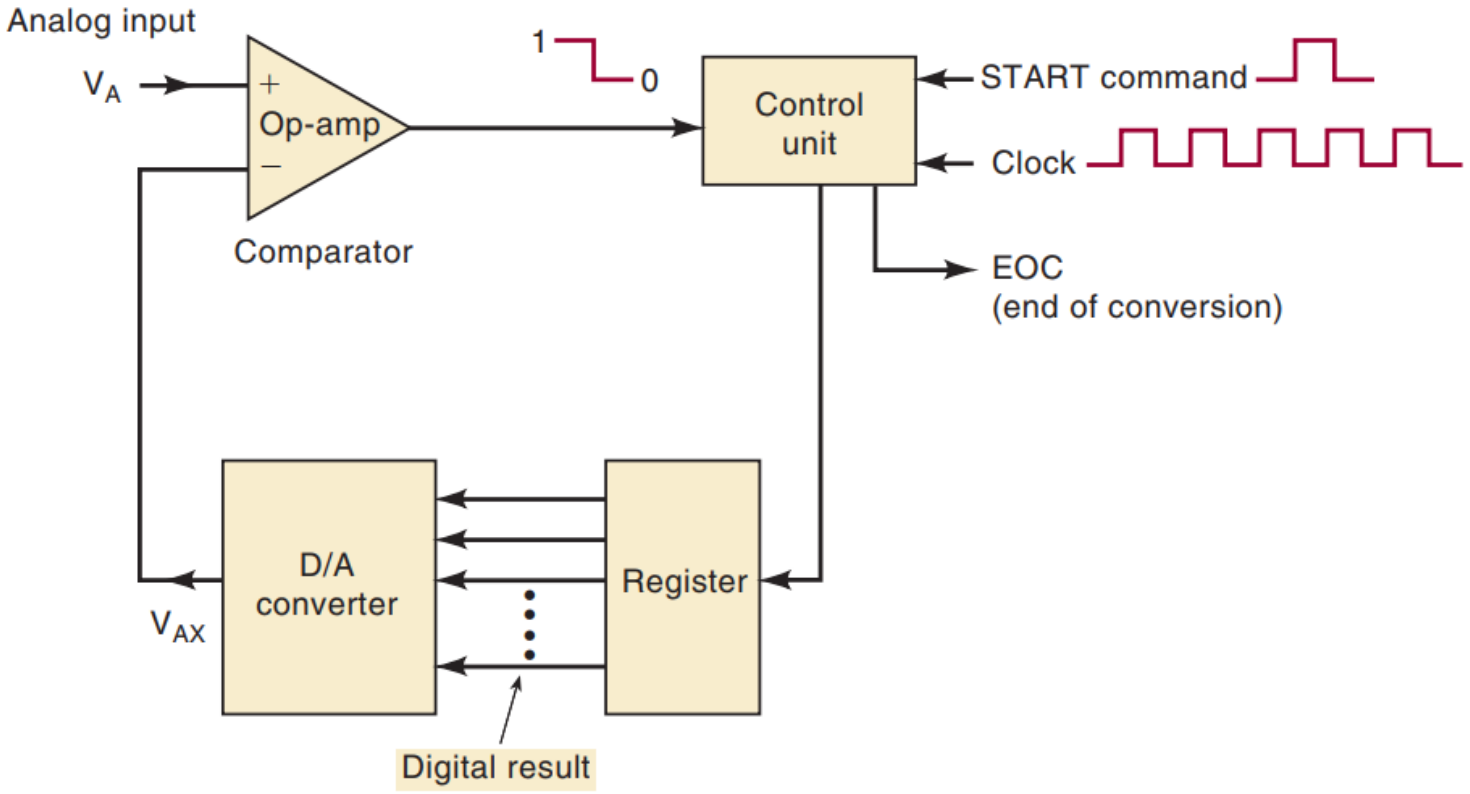
Suppose while you are designing and making the connections, this D_0 connection you have forgot to make or there is a disconnection, this is disconnected.



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 *time-consuming* than the D/A process, and many different methods have been developed and used.

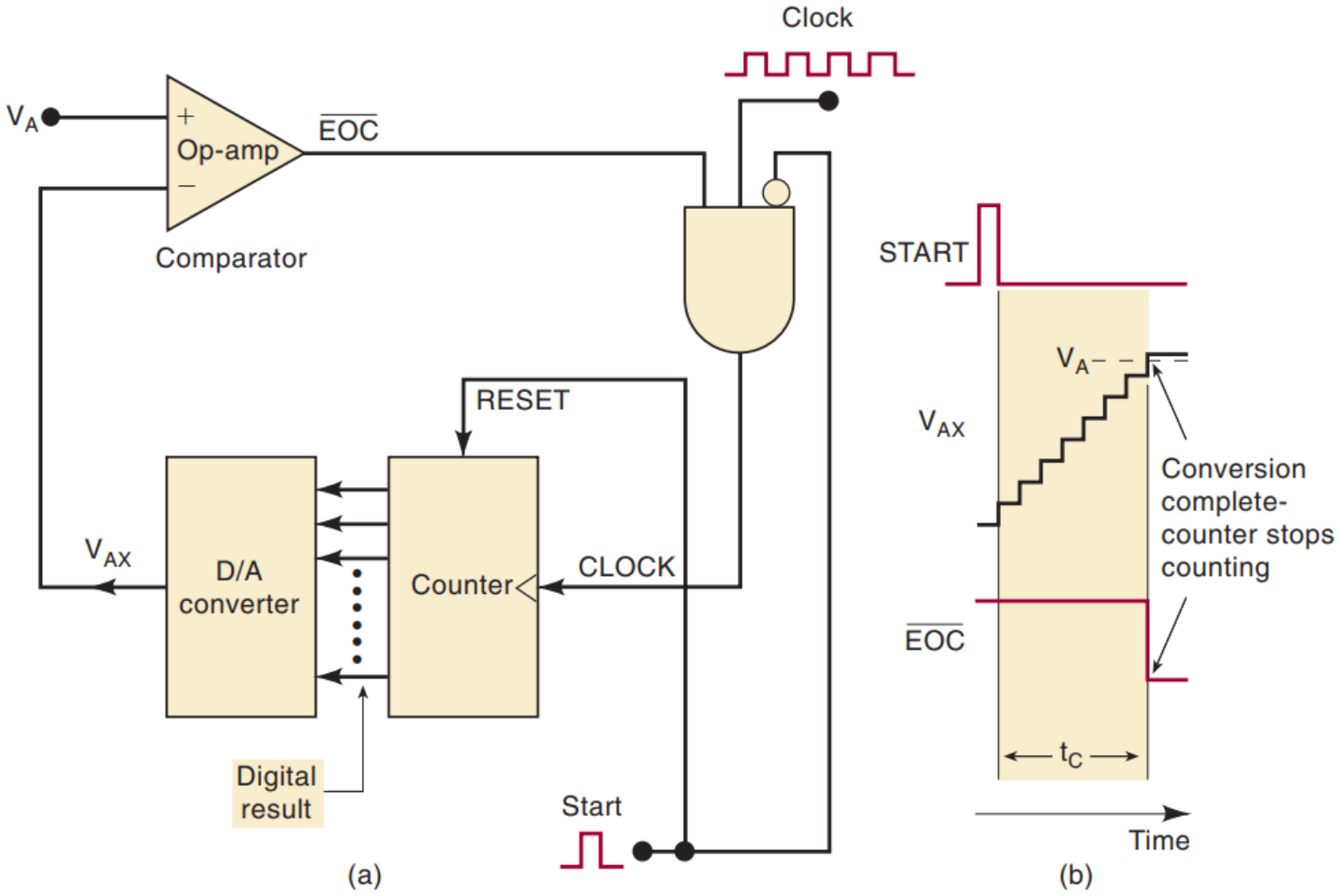


The basic operation of ADCs of this type consists of the following steps:

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.

DIGITAL-RAMP ADC

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. It is also referred to as a ***counter-type ADC***.



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 $V_{AX} = 0\,\text{V}$.
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.
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 μV). 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.

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

(a) The digital equivalent obtained for $V_A = 3.728$ 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 $V_A = 3.278$ 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 μs.

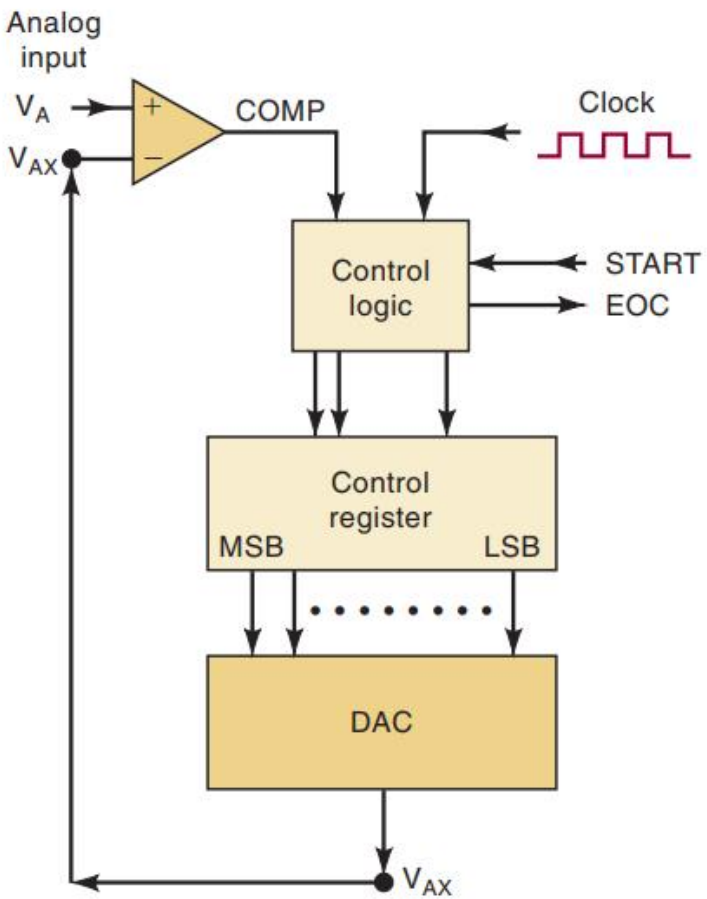
(c) The resolution of this converter is equal to the step size of the DAC, which is 10 mV. Expressed as a percentage, it is $1/1023 \times 100\% \approx 0.1\%$.

SUCCESSIVE-APPROXIMATION ADC

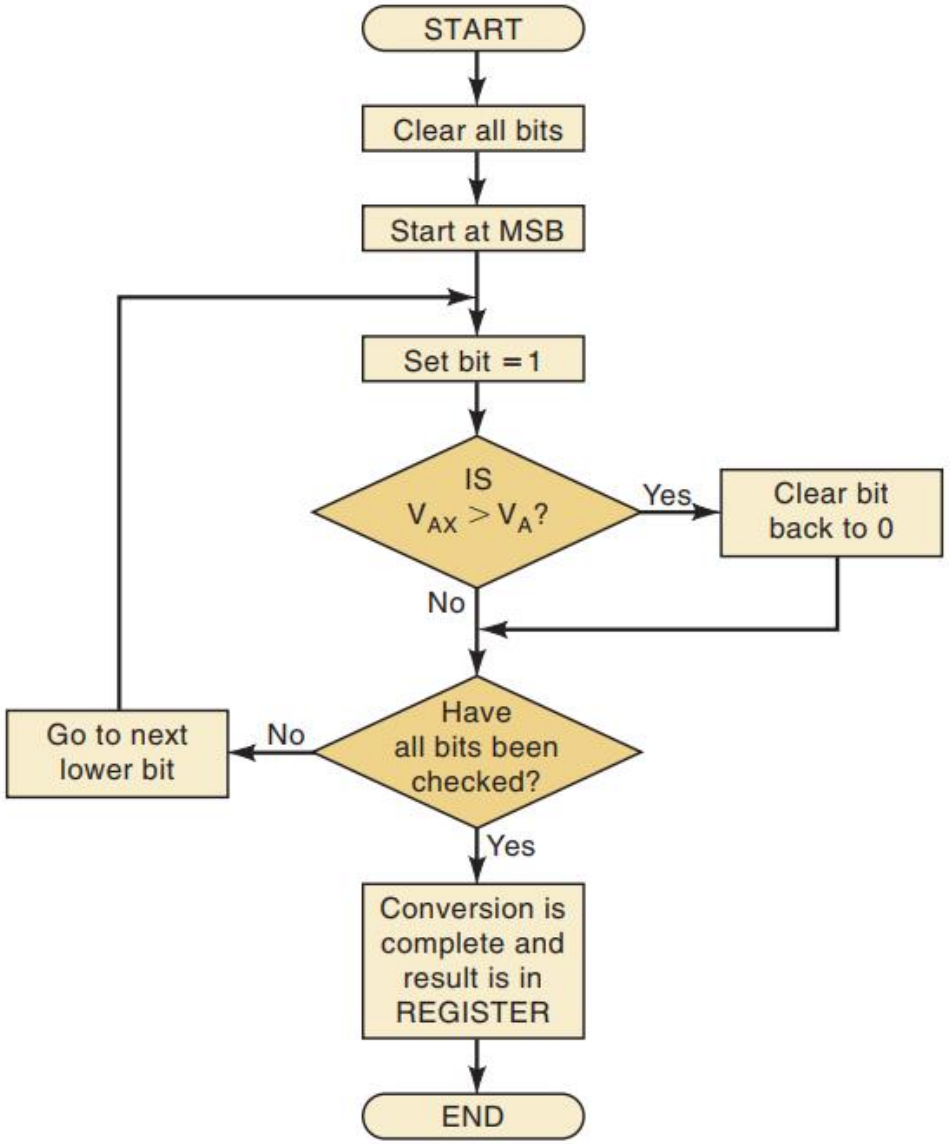
The basic arrangement is similar to that of the digital-ramp ADC.

The SAC, however, does not use a counter to provide the input to the DAC block but uses a register instead.

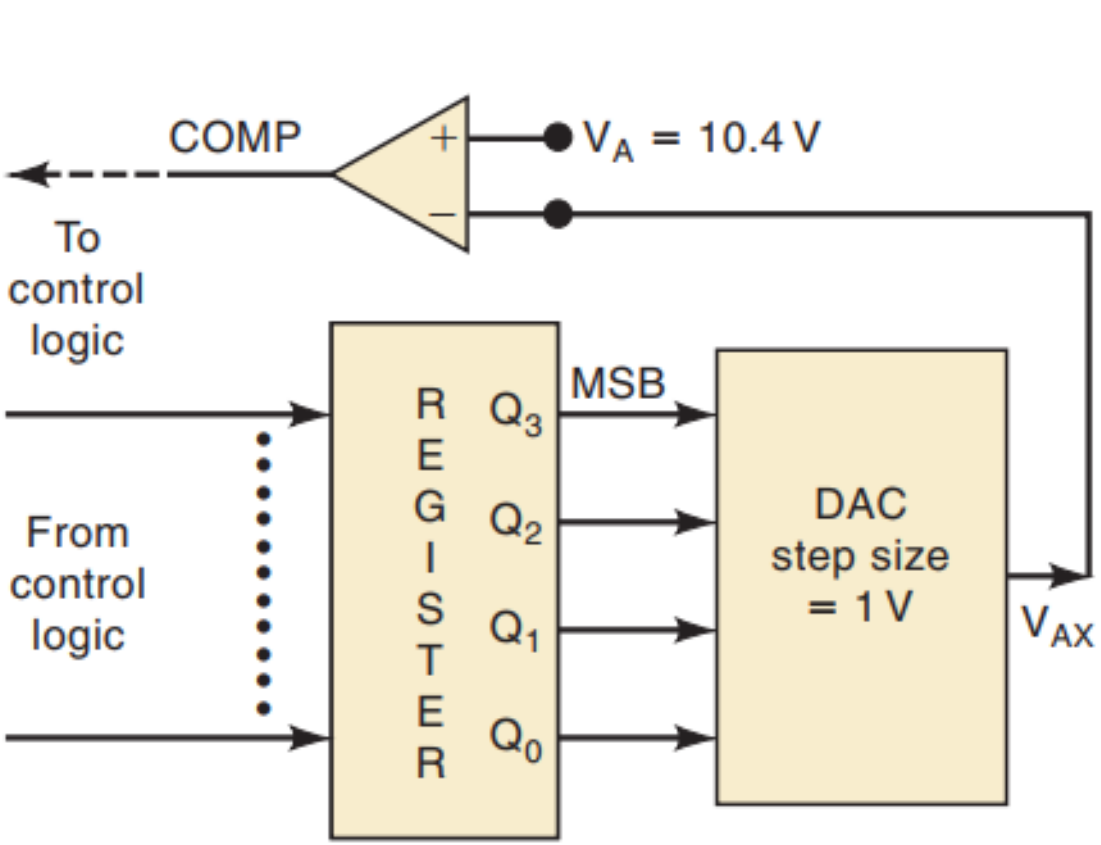
The control logic modifies the contents of the register bit by bit until the register data are the digital equivalent of the analog input V_A within the resolution of the converter.

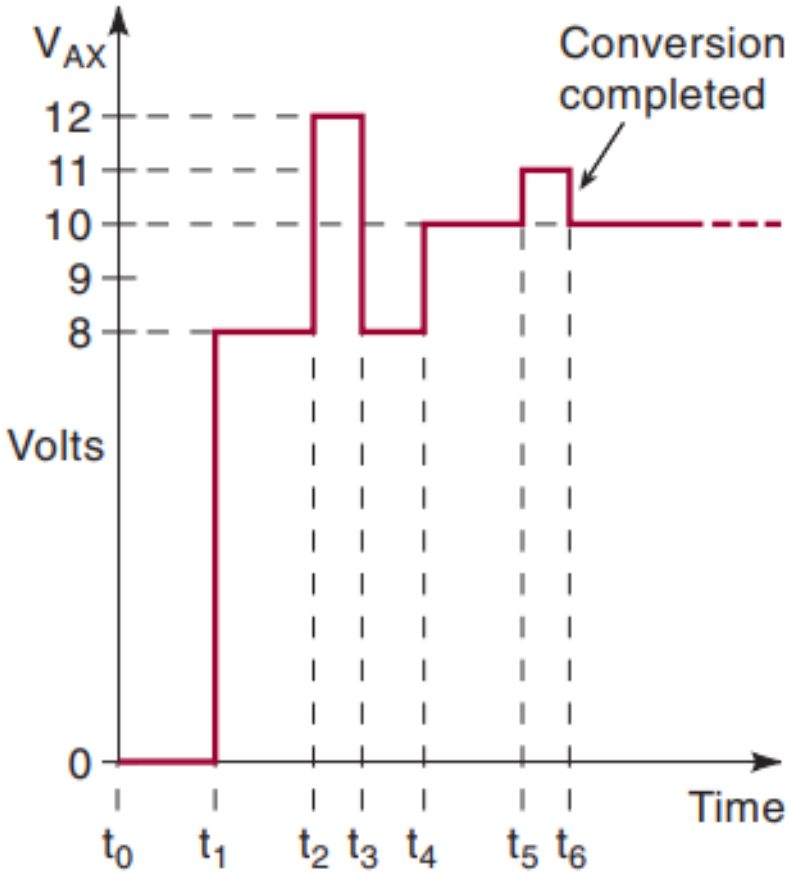


(a)



(b)


COMP
+
−
V_A = 10.4 V
To control logic
From control logic
REGISTER
Q_3
Q_2
Q_1
Q_0
MSB
DAC step size = 1 V
V_{AX}


V_{AX}
Conversion completed
12
11
10
9
8
Volts
0
Time
t_0
t_1
t_2
t_3
t_4
t_5
t_6

Let's assume that the analog input is $V_A = 10.4\,\text{V}$. 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 $V_{AX} = 0\,\text{V}$, as indicated at time t_0 on the timing diagram in Figure 11-19. With $V_{AX} < V_A$, the comparator output is HIGH.

At the next step (time t_1), the control logic sets the MSB of the register to 1 so that $[Q] = 1000$. This produces $V_{AX} = 8\,\text{V}$. 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 $V_{AX} = 12\,\text{V}$ 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 $V_{AX} = 10\,\text{V}$. 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 $V_{AX} = 11\,\text{V}$. 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.

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

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

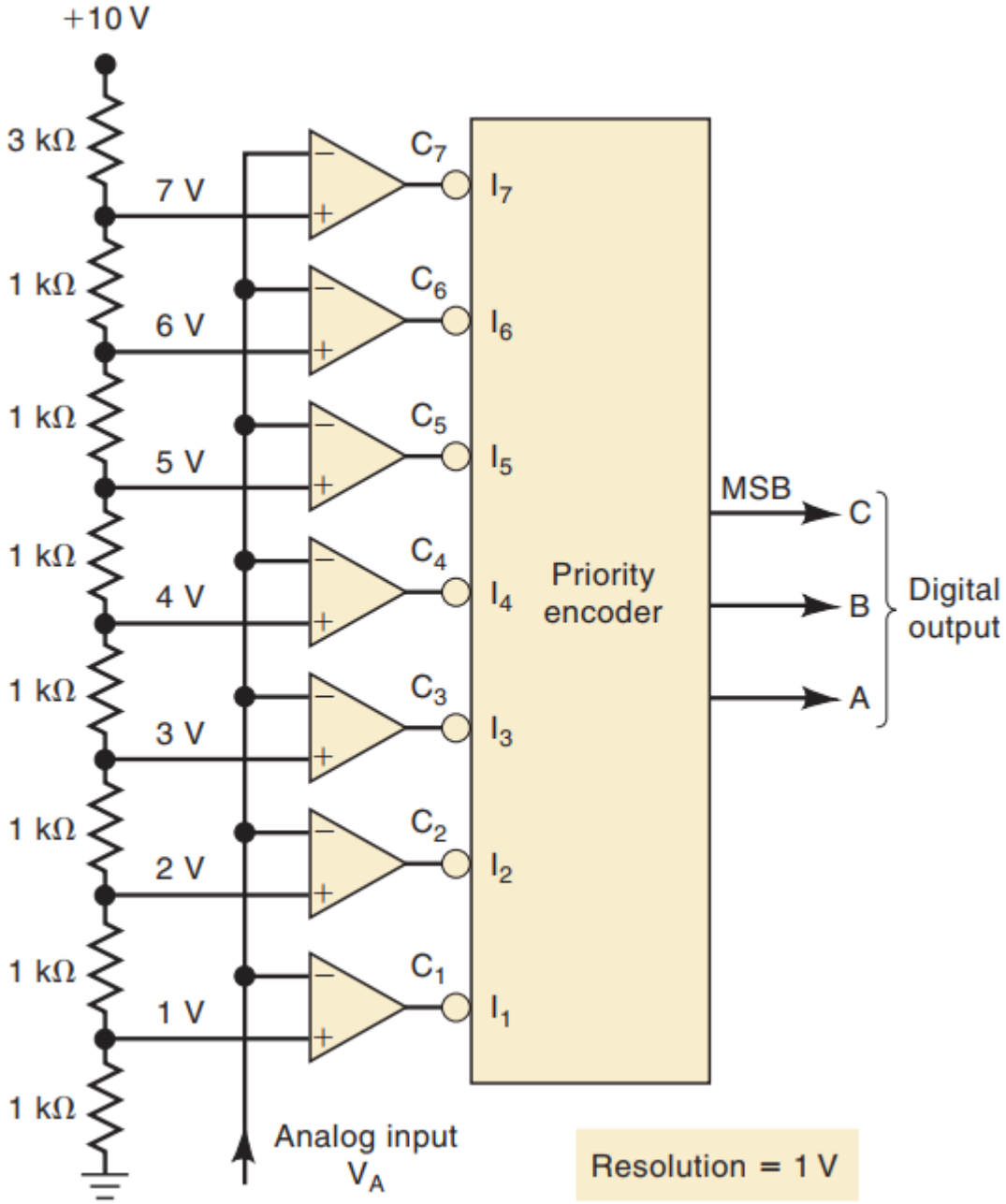
so that step 108 would produce $V_{AX} = 2.16\,\text{V}$ and step 109 would produce 2.18 V. The SAC always produces a final V_{AX} that is at the step *below* V_A. Therefore, for the case of $V_A = 2.17\,\text{V}$, the digital result would be $108_{10} = 01101100_2$.

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.

Analog in	Comparator outputs							Digital outputs		
V_A	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C	B	A
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



With $V_A < 1\,\text{V}$, all of the comparator outputs C_1 through C_7 will be HIGH. With $V_A > 1\,\text{V}$, 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.