Example 2.36 Obtain the minimal SOP expression for the function

$$Y = \Sigma_m(1,5,7,13,14,15,17,18,21,22,25,29) + \Sigma_d(6,9,19,23,30)$$

Solution The *K*-map for the given function is shown in Fig.E2.36.

DE \ ABC	000	001	011	010	110	111	101	100
00	0 (0)	4 0	12 0	8 0	24 0	28 0	20 0	16 0
01	1 (1)	5 1	13 1	9 d	25 1	29 1	21 1	17 1
11	3 0	7 1	15 1	11 0	27 0	31 0	23 d	19 d
10	2 0	6 d	14 1	10 0	26 0	30 d	22 1	18 1

Fig. E2.36

By combining the 1s and *d* s, one octet and two quads can be obtained as shown in Fig.E2.36. The simplified expression is

$$Y = \overline{D}E + \overline{A}CD + A\overline{B}D$$

2.8 QUINE–McCLUSKEY OR TABULAR METHOD OF MINIMIZATION OF LOGIC FUNCTIONS

The *K*-map is a very effective tool for minimization of logic functions with 4 or less variables. For logic expressions with more than 4variables, the visualization of adjacent cells and the drawing of the *K*-map become more difficult. The Quine–McCluskey method, also known as the tabular method, can be employed in such cases to minimize switching functions. This method employs a systematic, step-by-step procedure to produce a simplified standard form of expression for a function with any number of variables. The steps to be followed in the Quine–McCluskey method are:

Step 1 A set of all prime implicants of the function must be obtained.

Step 2 From the set of all prime implicants, a set of essential implicants must be determined by preparing a prime implicant chart.

Step 3 The minterms which are not covered by the essential implicants are taken into consideration and a minimum cover is obtained from the remaining prime implicants.

Selecting prime implicants The procedure for selecting prime implicants is given below:

(i) Each minterm should be expressed by its binary representation.

(ii) The minterms should be arranged according to increasing index (index can be defined as the number of 1s in a minterm). Separate each set of minterms possessing the same index by lines.

Scanned by CamScanner

(iii) Each term of index 'n' should be compared with each term of index $(n+1)$. For each pair of terms that can combine, the newly formed terms should be stated. If two minterms differ in only one variable, that variable should be removed and a dash placed at that position, thus a new term with one less literal is found. After all the pairs of terms with indices n and $(n+1)$ have been considered, a line should be drawn under the last term.

(iv) When the above process has been repeated for all the groups of terms, one stage of elimination will have been completed.

(v) The next stage of elimination or matching process should be repeated for the new terms. According to this stage, two terms can be combined only when they have dashes in the same positions.

(vi) The cycles have to be continued until no new list can be found (i.e. no further elimination of literals).

(vii) All terms which remain unchecked (do not match) during the process are considered to be prime implicants.

Prime implicant chart

(i) The prime implicants should be represented in rows and each minterm of the function in a column.

(ii) Crosses should be placed in each row to show the composition of minterms that makes the prime implicants.

(iii) A completed prime implicants table should be inspected for columns containing only a single cross. Prime implicants that cover minterms with a single cross in their column are called essential prime implicants.

Example 2.37 Find the minimal sum of products for the Boolean expression, $f = \Sigma(1, 2, 3, 7, 8, 9, 10, 11, 14, 15)$, using the Quine–McCluskey method.

Solution Firstly, these minterms are represented in the binary form as shown in Table E2.37(a). The above binary representation are grouped into a number of sections in terms of the number of 1s as shown in Table E2.37(b).

Table E2.37(a) Binary representation of minterms

Minterms	Variables			
	A	*B*	*C*	*D*
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
14	1	1	1	0
15	1	1	1	1

Scanned by CamScanner

Table E2.37(b) Group of minterms for different number of 1s

Number of 1s	Minterms	Variables			
		A	*B*	*C*	*D*
1	1 ✓	0	0	0	1
	2 ✓	0	0	1	0
	8 ✓	1	0	0	0
2	3 ✓	0	0	1	1
	9 ✓	1	0	0	1
	10 ✓	1	0	1	0
3	7 ✓	0	1	1	1
	11 ✓	1	0	1	1
	14 ✓	1	1	1	0
4	15 ✓	1	1	1	1

Any two numbers in these groups which differ from each other by only one variable can be chosen and combined, to get two-cell combinations, as shown in Table E2.37(c).

Table E2.37(c) 2-cell combinations

Combination	*A*	*B*	*C*	*D*
(1,3) ✓	0	0	–	1
(1,9) ✓	–	0	0	1
(2,3) ✓	0	0	1	–
(2,10) ✓	–	0	1	0
(8,9) ✓	1	0	0	–
(8,10) ✓	1	0	–	0
(3,7) ✓	0	–	1	1
(3,11) ✓	–	0	1	1
(9,11) ✓	1	0	–	1
(10,11) ✓	1	0	1	–
(10,14) ✓	1	–	1	0
(7,15) ✓	–	1	1	1
(11,15) ✓	1	–	1	1
(14,15) ✓	1	1	1	–

2^n

From the two-cell combinations, one variable and a dash in the same position can be combined to form 4-cell combinations as shown in Table E2.37(d)

Table E2.37(d) 4-cell combinations

Combination	*A*	*B*	*C*	*D*
(1, 3, 9, 11)	–	0	–	1
(2, 3, 10, 11)	–	0	1	–
(8, 9, 10, 11)	1	0	–	–
(3, 7, 11, 15)	–	–	1	1
(10, 11, 14, 15)	1	–	1	–

Note that the cells (1,3) and (9,11) form the same 4-cell combination as the cells (1,9) and (3,11). The order in which the cells are placed in a combination does not have any effect. Thus, the (1,3,9,11) combination may be given as (1,9,3,11). Using Table E2.37(d), the *prime implicants table* can be plotted as shown in Table E2.37(e).

Scanned by CamScanner

The columns having only one cross mark correspond to essential prime implicants. A tick mark is put against every column which has only one cross mark. A star mark is placed against every essential primary implicant. The sum of the prime implicants gives the function in its minimal SOP form, since all prime implicants are essential prime implicants.

Therefore, $f = \overline{B} \cdot D + \overline{B} \cdot C + A \cdot \overline{B} + C \cdot D + A \cdot C$

Table E2.37(e) *Prime implicants table*

Prime implicants	Minterms									
	1	2	3	7	8	9	10	11	14	15
(1, 3, 9, 11) *	X		X			X		X		
(2, 3, 10, 11) *		X	X				X	X		
(8, 9, 10, 11) *					X	X	X	X		
(3, 7, 11, 15) *			X	X				X		X
(10, 11, 14, 15) *							X	X	X	X
	✓	✓		✓	✓				✓	

Example 2.38 Find the minimal sum of products for the Boolean expression, $f(w,x,y,z) = \Sigma(1,3,4,5,9,10,11) + \Sigma\phi(6,8)$, using the Quine–McCluskey method.

Solution These minterms are firstly represented in binary form as shown in Table E2.38(a). The above binary representation is grouped into a number of sections in terms of a number of 1s as shown in Table E2.38(b).

Table E2.38(a) *Binary representation of minterms*

Minterms	Variables			
	w	*x*	*y*	*z*
1	0	0	0	1
3	0	1	1	0
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1

Table E2.38(b) *Group of minterms for different number of 1s*

Number of 1s	Minterms	Variables			
		w	*x*	*y*	*z*
1	1 ✓	0	0	0	1
	4 ✓	0	1	0	0
	8 ✓	1	0	0	0
2	3 ✓	0	0	1	1
	5 ✓	0	1	0	1
	6 ✓	0	1	1	0
	9 ✓	1	0	0	1
	10 ✓	1	0	1	0
3	11 ✓	1	0	1	1

Scanned by CamScanner

Any two numbers in these groups which differ from each other by only one variable can be chosen and combined to get two-cell combinations, as shown in Table E2.38(c).

Table E2.38(c) *2-cell combinations*

Combination	w	x	y	z
(1,3) ✓	0	0	–	1
(1,5)	0	–	0	1
(1,9) ✓	–	0	0	1
(4,5)	0	1	0	–
(4,6)	0	1	–	0
(8,9) ✓	1	0	0	–
(8,10) ✓	1	0	–	0
(3,11) ✓	–	0	1	1
(9,11) ✓	1	0	–	1
(10,11) ✓	1	0	1	–

From the two-cell combinations, one variable and a dash in the same position can be combined to form 4-cell combinations as shown in Table E2.38(d).

Table E2.38(d) *4-cell combinations*

Combination	w	x	y	z
(1, 3, 9, 11)	–	0	–	1
(8, 9, 10, 11)	1	0	–	–

Note that the cells (1,3) and (9,11) form the same 4-cell combination as the cells (1, 9) and (3,11). The order in which the cells are placed in a combination has no effect. Therefore, the (1,3,9,11) combination may be written as (1, 9, 3, 11). Using table E2.38(d), the prime implicants table can be plotted as shown in Table E2.38(e).

Don't care minterms cannot be listed as column headings in the chart because they do not have to be covered by a minimal expression.

Table E2.38(e) *Prime implicants table*

Prime implicants	Minterms						
	1	3	4	5	9	10	11
(1,5)	X			X			
(4,5)			X	X			
(4,6)			X				
(1,3,9,11) *	X	X			X		X
(8,9,10,11) *					X	X	X
		✓				✓	

The columns having only one cross mark correspond to the essential prime implicants. A tick mark is put against every column which has only one cross mark. A star mark is put against every essential primary implicant. The prime implicant which covers the minterm (1,3,9,11) is the essential prime implicant. Therefore, in order to cover the remaining minterms, the reduced prime implicant chart is formed as shown in Table E2.38(f).

Scanned by CamScanner

Table E2.38 (f) *Reduced prime implicants table*

Prime implicants	Minterms						
	1	3	4	5	9	10	11
(1, 5)	X			X			
(4, 5)			X	X			
(4, 6)			X				

To cover the minterms (4,5), the prime implicants (4,5) can be selected, in addition to the essential prime implicants, for obtaining the minimal expression of the given function.

Therefore, $f(w, x, y, z) = \bar{x}z + \bar{w}x\bar{y} + w\bar{x}$

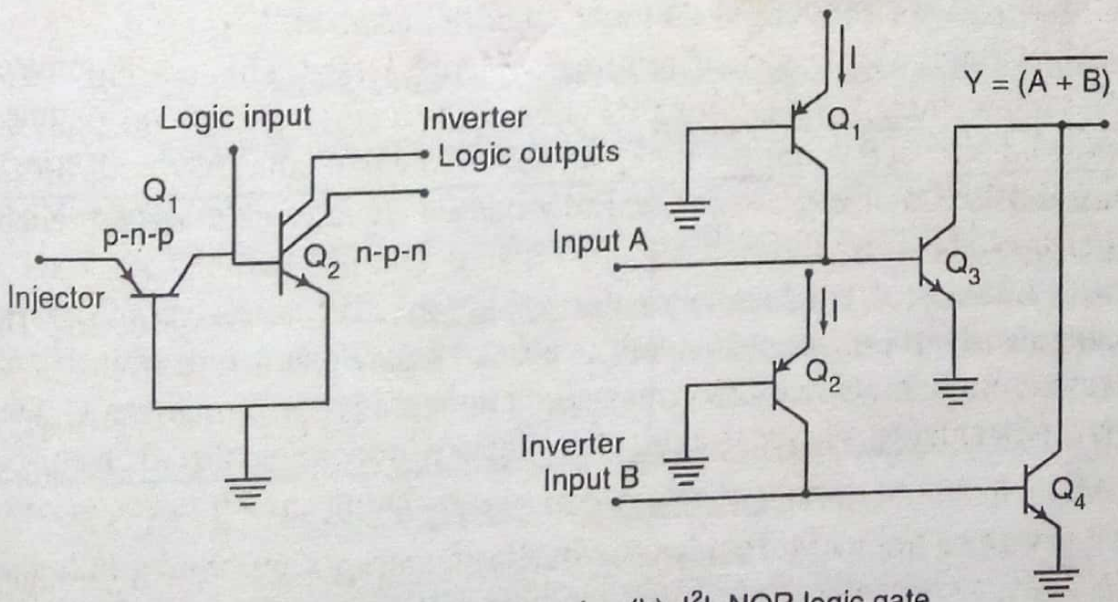
Scanned by CamScanner

4.11 INTEGRATED-INJECTION LOGIC (I^2L)

Integrated-Injection Logic, IIL or I^2L, is a new LSI technique also called Merged Transistor Logic (MTL) that uses both *n-p-n* and *p-n-p* bipolar junction transistors to form a large number of IC gates on a chip. It reduces the number of metal inter-connections. When operated at low speeds, I^2L dissipates less power (5mW) than any logic family including CMOS. At high speeds (5ns), it only dissipates 5mW per gate. Because of its high speed and less power dissipation, it is used in large computers.

Operation The basic I^2L inverter circuit is shown in Fig. 4.20(a). Due to the absence of resistors, the I^2L inverter occupies much smaller area than a T^2L inverter. The base of Q_1 and emitter of Q_2 are internally merged; the collector of Q_1 and the base of Q_2 are merged, and hence only four separate regions are required to form the two transistors. Thus, the entire I^2L gate takes only the space of a single TTL multiple-emitter transistor.

The *p-n-p* transistor Q_1 acts as a current source and active pull-up, and the multiple-collector *n-p-n* transistor Q_2 operates as an inverter. Most of the current leaving from the emitter of Q_1 is injected directly into the base of Q_2, and hence the emitter of Q_1 is known as the *injector*, and the integrated gate structure is called the Integrated Injection Logic.



(a) I^2L logic Inverter and (b) I^2L NOR logic gate

Fig. 4.20

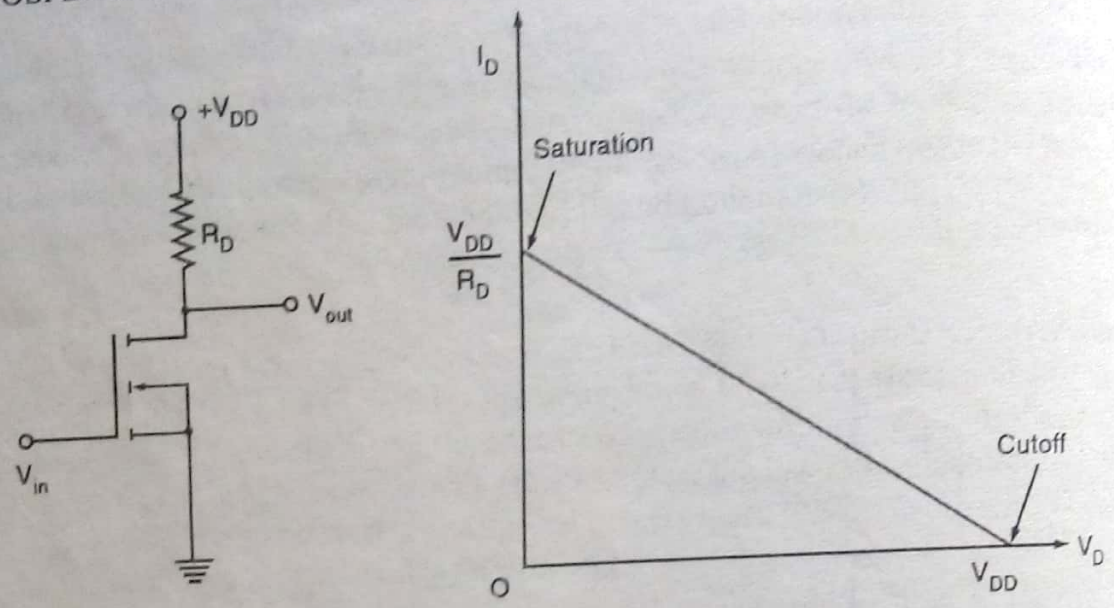
The base of Q_1 and emitter of Q_2 are internally merged and the collector of Q_1 and base of Q_2 are merged, the entire I^2L gate takes only the space of a single TTL multiple-emitter transistor.

Scanned by CamScanner

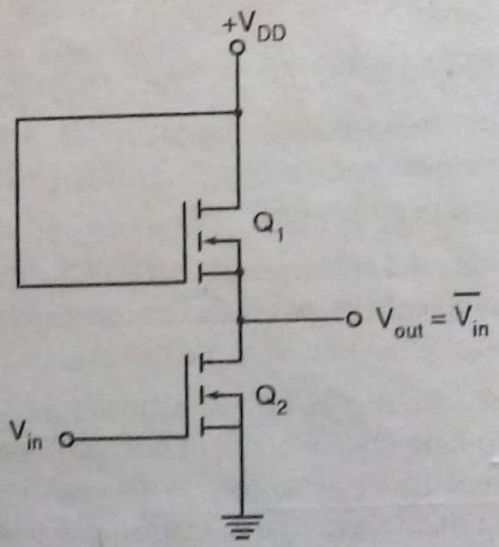
4.12.2 MOSFET Logic Circuits

For an *n*-channel MOS, supply voltage is of about +5V. The two voltage levels are a function of the threshold voltage V_T. The low level ranges anywhere from zero to V_T, and the high level ranges from V_T to V_{DD}. The *n*-channel gates usually employs positive logic.

Fig. 4.23 shows a MOSFET driver with a passive load resistor (R_D). MOSFET acts as a switch depending on V_{in}, either low or high. It switches between saturation and cutoff. When V_{in} is low, MOSFET is cutoff and hence V_{out} is high. When V_{in} is high, MOSFET is saturated and hence V_{out} is low.



(a) MOSFET driver with passive load and its load line



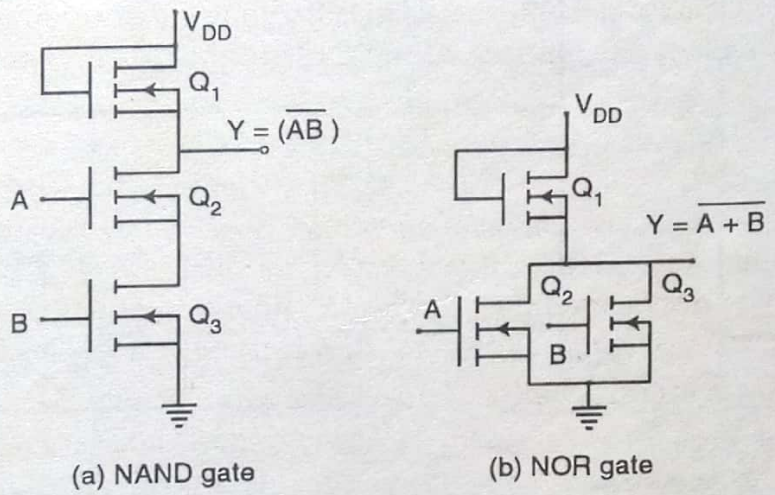
(b) MOSFET driver with active load

Fig. 4.23

In MOS technology, MOSFET fabrication is easier than resistors. For this reason, the resistor R_D in Fig. 4.23(a) is replaced by an *n*-channel MOSFET as shown in

Scanned by CamScanner

Fig. 4.23(b). The gate of the upper MOSFET is connected to the drain; this MOSFET always conducts and behaves like the resistor, R_D. The upper MOSFET has a resistance ten times greater than that of the lower MOSFET. When the input voltage is low (below V_T), Q_2 turns OFF. Since Q_1 is always ON, the output voltage is at about V_{DD}. When the input voltage is high (above V_T), Q_2 turns ON. Current flows from V_{DD} through the active load resistor, Q_1, and into Q_2. Thus, the circuit behaves as an inverter. The geometry of the two MOS devices must be such that the resistance of Q_2 when conducting is much less than the resistance of Q_1 to maintain the output Y at a voltage below V_T.



(a) NAND gate (b) NOR gate

Fig. 4.24 *n-channel MOS logic circuits*

The circuit of NMOS NAND gate that uses three transistors in series is shown in Fig. 4.24(a). Inputs A and B must be high for all transistors to conduct and cause the output to go LOW. If either of the inputs is LOW, the corresponding transistor is turned OFF and the output is HIGH. Again, the series resistance formed by the two active MOS devices must be much less than the resistance of the load of MOSFET.

The NOR gate that uses two NMOS transistors in parallel is shown in Fig. 4.24(b). If either of the inputs is HIGH, the corresponding MOS transistor conducts and the output is LOW. If the inputs are LOW, both MOSFETs are OFF, and hence the output is HIGH.

Fig. 4.25(a) illustrates a 3-input NMOS AND gate with an additional output inverter. The transistors Q_1 and Q_2 serve as loads. If any one input is at 0V, then the corresponding input transistor is in OFF state, resulting in conduction of Q_6. This leads to 0V at the output. If all inputs are at a high voltage, the MOSFETs Q_3, Q_4 and Q_5 will be ON and they offer a low resistance compared to the resistance of the load transistor Q_1, then the transistor Q_6 goes OFF. The output is now at a high level. Hence, the circuit acts as an AND gate.

A 3-input n-channel MOS OR gate with an output inverter is shown in Fig. 4.25(b). Transistors Q_1, Q_4, Q_5 and Q_6 form a NOR gate, and transistors Q_2 and Q_3 form an inverter. Transistors Q_1 and Q_2 act as loads. If the voltage is equal to V_{DD} at one of the inputs, the corresponding transistor is ON and allows the current to flow. The voltage at point K then drops close to zero, the transistor Q_3 switches OFF, and

Scanned by CamScanner

the output voltage approaches the level V_{DD}. If all the inputs are LOW (at 0V), the transistors Q_4, Q_5 and Q_6 turn OFF, and the voltage at point K comes close to V_{DD}. The transistor Q_3 then switches ON and the output drops to zero. Hence, this circuit performs the OR function.

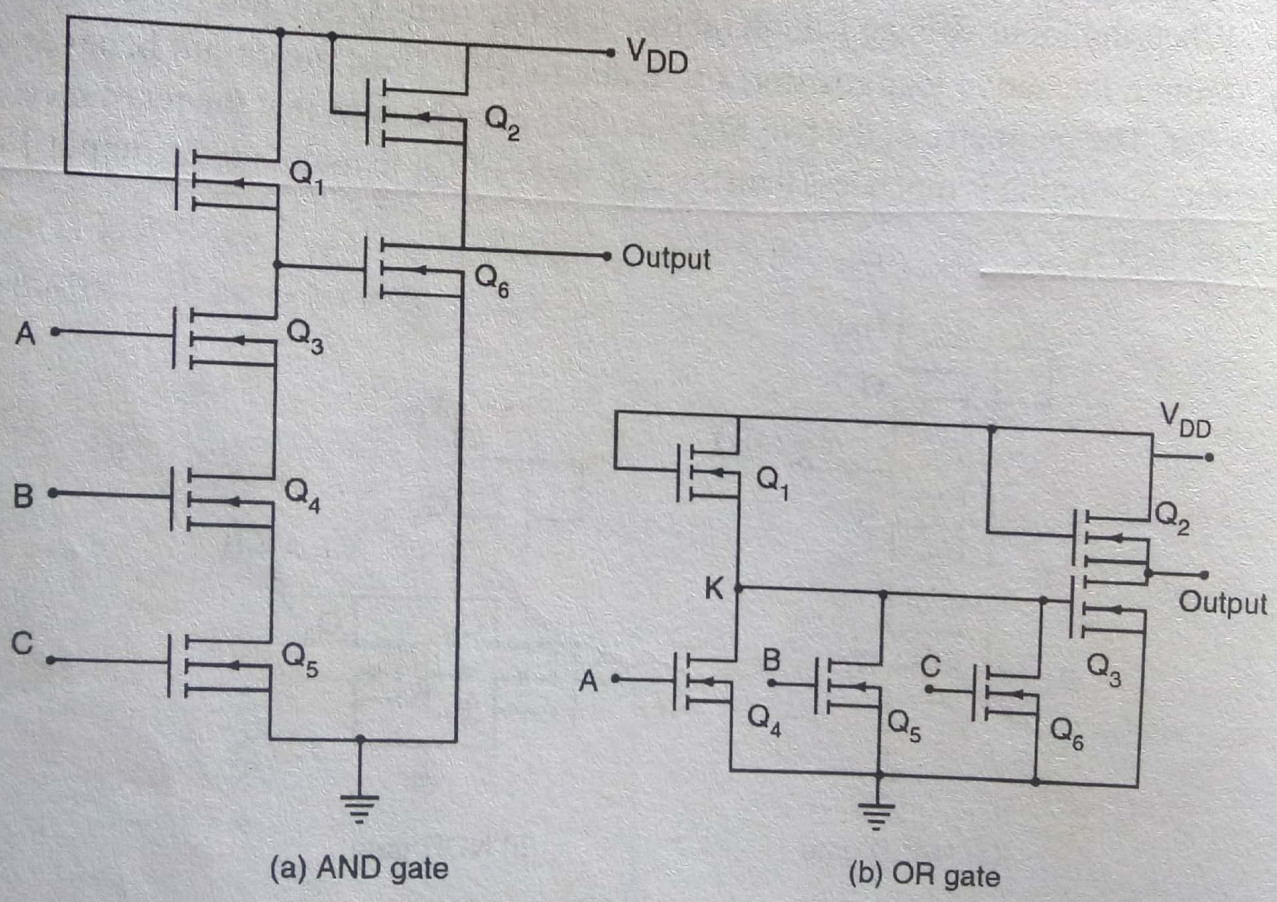


Fig. 4.25 *n-channel MOS logic circuits*

Scanned by CamScanner

Chapter

13

D/A and A/D Converters

13.1 INTRODUCTION

The data converters convert one form of data into another form. Real world processes produce analog signals which carry information pertaining to process variables such as voltage, current, charge, temperature and pressure. The rate of flow of such information may be very slow or very fast. It is difficult to store, manipulate, compare, calculate and retrieve such data with good accuracy using purely analog technology. But, computers can perform these operations quickly and efficiently using digital techniques. Therefore, it is necessary to convert the analog signals from various transducers into its equivalent digital data, which in turn act as the input for digital systems. Thus the requirement for converting analog signal into digital data emerged. The computers also need to communicate with people and physical processes through the use of analog signals, which necessitated the process of digital to analog conversion. This chapter discusses the most common D/A and A/D conversion techniques, D/A converter and A/D converter ICs, delta modulators and demodulators followed by sigma-delta converters and widely used ICs for D/A and A/D conversions.

13.2 ANALOG AND DIGITAL DATA CONVERSIONS

In the application of signal processing, the measurement and analysis of signals are very important to discover their characteristics. If the signal is unknown, the process of analysis begins with the acquisition of the signal. The most common technique of acquiring signals is by *sampling*. Sampling a signal is the process of acquiring its values only at discrete points in time.

The definitions related to the process of sampling and the subsequent analog and digital conversion processes are

(i) an *analog signal* is a signal that is defined over a continuous period of time, and in which the amplitude may assume a continuous range of values.

(ii) the term *quantization* refers to the process of representing a variable by a finite set of discrete values.

(iii) a *quantized variable* is the signal variable that can assume only finite distinct values.

Scanned by CamScanner

(iv) a *discrete time signal* is the one that is defined at particular points of time only. Therefore, the independent time variable is quantized. When the amplitude of a discrete-time signal is allowed to assume a continuous range of values, the function is called a *sampled-data* signal. A sampled data signal could result from sampling an analog signal at discrete points of time.

(v) a *digital signal* is a function, in which the time and amplitude are quantized. A digital signal is always represented by a sequence of *words*, where each word can contain a finite number of *bits* (binary digits).

A D/A converter (DAC) converts digital data into its equivalent analog data. The analog data is required to drive motors and other analog devices. An A/D converter (ADC) converts analog data into its equivalent digital data i.e., binary data. The A/D converter and D/A converter are also called *data converters* and they are also available as monolithic integrated circuits.

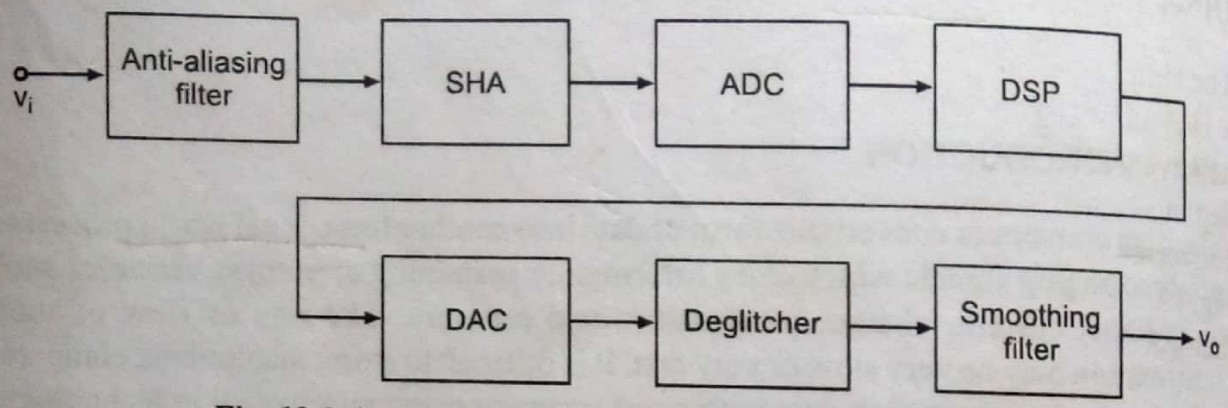


***Fig. 13.1** Sampled data system using A/D and D/A converters*

Figure 13.1 shows a typical application in which A/D and D/A conversions are employed. An analog input signal from a transducer is band-limited by *anti-aliasing* filter. The signal is then sampled at a frequency rate higher than twice the maximum frequency of the band-limited input signal. That is, when the A/D converter is operated at a rate of f_s samples/second, the highest frequency component of the input signal can be less than $f_s/2$. The A/D converters normally require the input to be held constant during the conversion process. Hence, a *Sample-and-Hold Amplifier* (SHA) is introduced in the loop as shown in Fig. 13.1 before A/D converter. The SHA freezes the band-limited signal just before the start of each conversion. The digital signal from A/D converter may be processed, transmitted and recorded in digital form by the digital signal processor (DSP) block. Then, the digital signal is converted into analog signal by D/A converter for use in analog form. The D/A converter is usually operated at the same frequency f_s as that of A/D converter. The output of D/A converter is commonly a staircase signal, which is passed through a smoothing filter to eliminate the quantization noise effects. A *deglitcher* may be introduced in the loop before the smoothing process, to remove any output glitches generated during input code variations.

The schematic structure shown in Fig. 13.1 is prevalent either in full or in part in numerous applications such as digital signal processing, direct signal control, digital audio mixing, music and video synthesis, pulse-code modulation (PCM) communication, data acquisition and digital microprocessor based instrumentation.

Scanned by CamScanner

13.3 SPECIFICATIONS OF D/A CONVERTER

The important specifications, namely, accuracy, offset voltage, monotonicity, resolution and settling time of D/A converter are discussed below.

Accuracy

The components in D/A converter circuits are prone to mismatches, drift, ageing, noise and other sources of errors. These factors lead to degradation in conversion performance.

Absolute accuracy defines the maximum deviation of the output from the ideal value and it is expressed in fractions of 1 LSB. The D/A converter manufacturers follow different ways of specifying accuracy. The D/A converter errors are classified as *static* and *dynamic errors*.

Offset Voltage

The simplest kind of static errors are *offset error* and *gain error*. Ideally, the output of a D/A converter is 0V when all the bits of binary input word are 0s. In practice, however, there is a very small output voltage called *offset voltage* or *offset error* as depicted in Fig. 13.2(a). The offset error is nullified by translating the actual A/D converter characteristics up or down so that it goes through the origin as shown in Fig. 13.2(b). The gain error shown in Fig. 13.2(b) is compensated by adjusting the scale factor *K*.

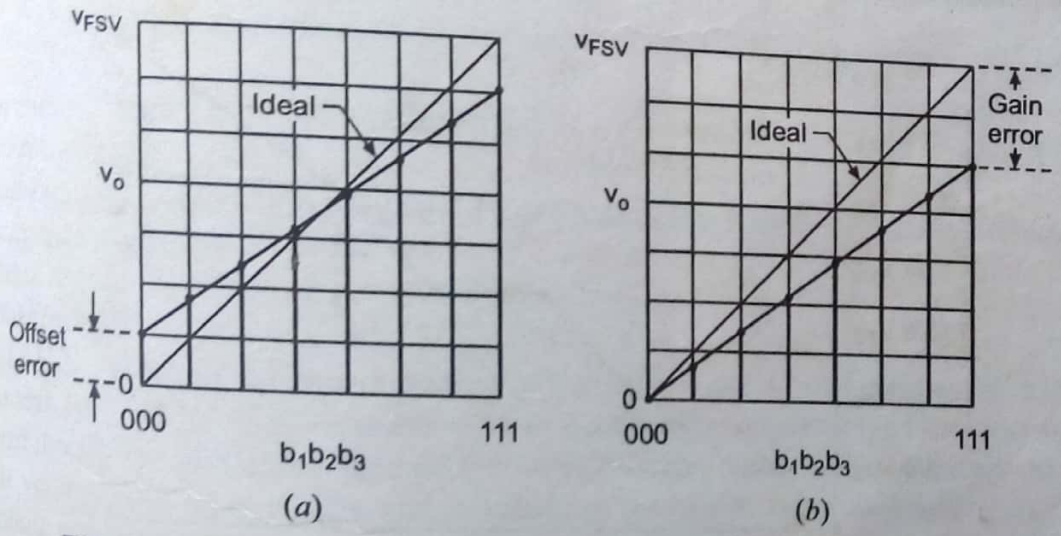


Fig. 13.2 (a) D/A converter offset and gain errors; (b) Nullifying the errors

Linearity

The most common dynamic errors are *full-scale error* and *linearity error*. *Full-scale error* is the maximum deviation of the output value from its expected or *ideal* value, expressed in percentage of full-scale. *Linearity error* is the maximum deviation in step size from the ideal step size. More expensive D/A converters have full-scale and linearity errors as low as 0.001% of full-scale. General purpose D/A converters have accuracies in the range of 0.01 to 0.1%.

Scanned by CamScanner

13.4 BASIC D/A CONVERSION TECHNIQUES

The D/A converter converts digital or binary data into its equivalent analog value. The input digital data for a D/A converter is an n-bit binary word D. The bit b_1 is called the most significant bit (MSB) and bit b_n the least significant bit (LSB). Then, the quantity D can be represented by $D = b_1 2^{-1} + b_2 2^{-2} + b_3 2^{-3} + \ldots + b_n 2^{-n}$. The D/A converter accepts the binary input D and produces an analog output, which is proportional to D using a reference voltage V_R. The converted analog value is either in voltage or current form. For a voltage output D/A converter, the conversion characteristic may be expressed as

$$V_o = KV_{FS} = KV_{FS}\left(b_1 2^{-1} + b_2 2^{-2} + b_3 2^{-3} + \ldots + b_n 2^{-n}\right) \tag{13.3}$$

where V_o = output voltage

V_{FS} = full-scale range of voltage

K = scaling factor, usually unity

$b_1 \ldots b_n$ = n-bit binary fractional word with binary point located at the left

b_1 = most significant bit (MSB) of weight $V_{FS}/2$

b_n = least significant bit (LSB) of weight $V_{FS}/2^n$

The symbolic representation of an n-bit D/A converter is shown in Fig. 13.4.

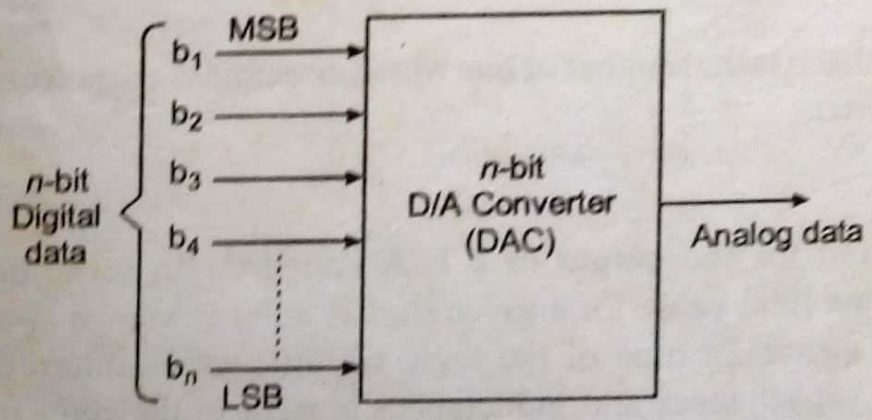


Fig. 13.4 n-bit D/A converter

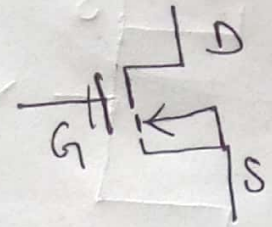
13.4.1 Weighted Resistor Type D/A Converter

Scanned by CamScanner

SSN

n-mos ON:

vgate ≥ vth (threshold)

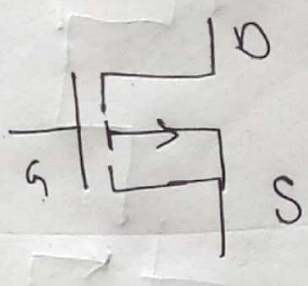


n-mos

P-mos ON:

vgate < vth.

0 ≤ vth ≤ vcc.



p-mos

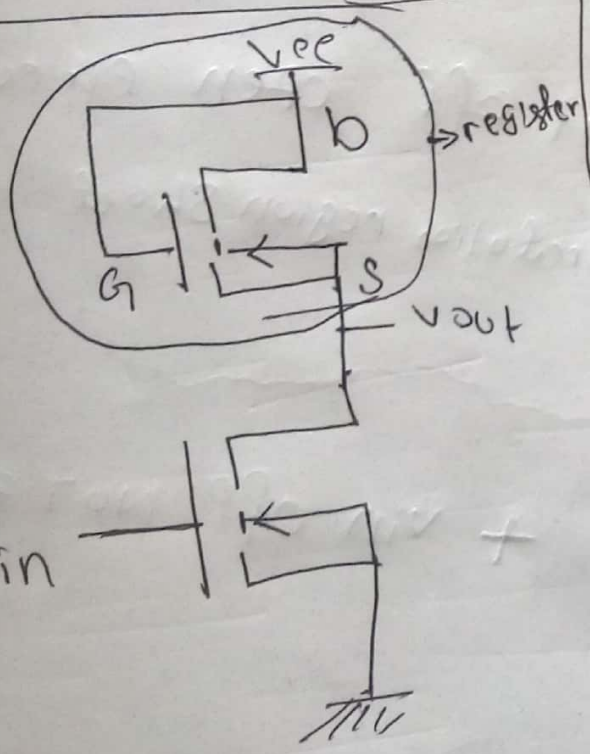
digital 0: 0 - vth

digital 1: vth - vcc

i) G = Gate

ii) D = Drain

iii) S = Source

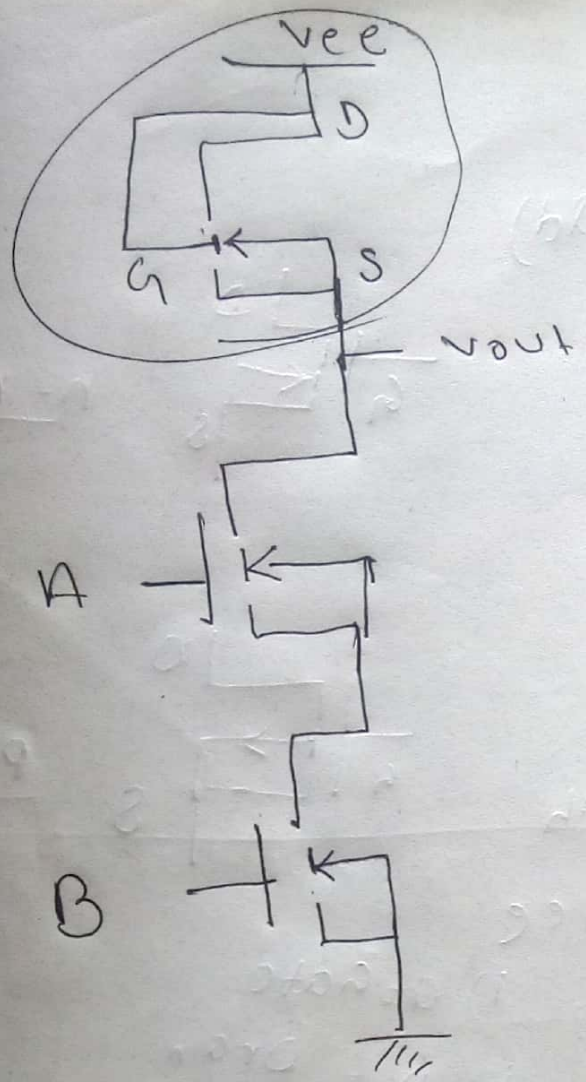


vin	vout
0	vcc
vcc	0 (because voltage connected)

gate voltage = cut off.

vin = 0, mos in ON, saturation region.

vcc = , vout directly is connected with gro...



NAND

A	B	Vout
0	0	Vee
0	Vee	Vee
Vee	0	Vee
Vee	Vee	0

0 = MOS off, cut off open circuit

Vee = MOS on, saturation region, short ckt

AND :-

NAND @ Vout + Vin off NOT gate

$(AB+C)'$

Scanned by CamScanner

4.3.5 Noise Immunity or Noise Margin

The term noise denotes an unwanted signal voltage, e.g., hum, transients and glitches.

Noise can sometimes cause the input voltage of a logic gate to drop below V_{IH} (min) or rise above V_{IL} (max), which leads to unreliable operation. *Noise immunity* is the maximum noise voltage that may appear at the input of a logic gate without changing the logical state of its output. A quantitative measure of noise immunity is called *noise margin*.

The difference between the operating input-logic voltage level and the threshold voltage is called the noise margin of the circuit. The manufacturer usually quotes the noise margin, which refers to the amplitude of the noise voltage that may cause the logic level to change. In the worst case, a TTL gate functions properly as long as the noise margin is kept less than 0.4V.

4.3.6 Operating Temperature

All IC gates are semiconductor devices that are temperature-sensitive by nature. The operating temperature ranges for an IC vary from 0°C to + 70°C for consumer and industrial applications and from–55°C to + 125°C for military applications.

4.3.7 Power Supply Requirements

The amount of power and supply voltage required by an IC are the main parameters to be taken into consideration while choosing a proper power supply.

4.3.8 Current and Voltage Parameters

The following currents and voltages are very important in designing digital systems. The values given below are for TTL gates only.

High-level input voltage (V_{IH}) [$V_{in(1)}$] It is the minimum voltage level required for a logical 1 at an input. Its minimum value is 2V.

Low-level input voltage (V_{IL}) [$V_{in(0)}$] It is the maximum input voltage required for a logical 0 (LOW) at an input. Its maximum value is 0.8V.

High-level output voltage (V_{OH}) [$V_{out(1)}$] It is the minimum voltage required for a logical 1 state at the output. Its minimum value is 2.4V.

Low-level output voltage (V_{OL}) [$V_{out(0)}$] It is the maximum voltage available at the circuit's output corresponding to the logical 0 state. Its maximum value is 0.4V.

High-level input current (I_{IH}) [$I_{in(1)}$] The current that flows through an input when a specified high-level voltage is applied to that input.

Low-level input current (I_{IL}) [$I_{in(0)}$] The current that flows through an input when a specified low-level voltage is applied to that input.

High-level output current (I_{OH}) [$I_{out(1)}$] The current that flows from an output in the logical 1 state under specified load conditions.

Low-level output current (I_{OL}) [$I_{out(0)}$] The current that flows from an output in the logical 0 state under specified load conditions.

Scanned by CamScanner

saturated and non-saturated, bipolar families are classified into

(i) saturated logic and

(ii) non-saturated logic.

The following are the saturated bipolar logic families:

1. Resistor–Transistor Logic (RTL)
2. Direct-coupled Transistor Logic (DCTL)
3. Diode–Transistor Logic (DTL)
4. High Threshold Logic (HTL)
5. Transistor–Transistor Logic (TTL)
6. Integrated-injection Logic (I^2L)

The following are the non-saturated logic families:

1. Schottky TTL
2. Emitter-coupled Logic (ECL)

4.2.2 MOS Families

The MOS families include

1. PMOS *p*-channel MOSFETs
2. NMOS *n*-channel MOSFETs
3. CMOS Complementary MOSFETs

4.3 CHARACTERISTICS OF DIGITAL ICs

Some of the important parameters or properties of various logic families are listed as follows:

1. Speed of operation (Propagation delays)
2. Power dissipation
3. Fan-in
4. Fan-out
5. Noise immunity
6. Operating temperature
7. Power supply requirements

The comparison of performance of digital ICs may be made with reference to the above properties.

4.3.1 Speed of Operation

The speed of operation of an IC is expressed in terms of propagation delay. *Propagation delay* is defined as the time taken for the output of a gate to change after the inputs have changed.

A logic signal always experiences a delay in going through a circuit. The two propagation delay times shown in Fig. 4.1 are defined as follows:

t_{PLH}: It is the propagation delay time in going from logical LOW (0 state) to logical HIGH (1 state).

Scanned by CamScanner

The linearity of a D/A converter is defined as the precision with which the digital input is converted into analog output. An ideal D/A converter produces equal increments in analog output for equal increments in digital input as shown in dotted line curve of the transfer characteristics of Fig. 13.3. However, in an actual D/A converter, the gain and offset errors due to resistors introduce non linearity as shown by the solid line of the transfer characteristics of Fig. 13.3.

The linearity error measures the deviation of the output from the fitted straight line which passes through the measured output points. It is represented by ϵ/Δ as shown in Fig. 13.3. Commonly, the linearity of D/A converter is specified as less than $\pm$ 1/2 *LSB* meaning that $|\epsilon| < (1/2)\Delta$.

Differential nonlinearity (DNL) error: For a D/A converter, the DNL error is the difference between the ideal and the measured output responses for successive D/A converter codes. An ideal D/A converter response would have analog output values exactly one code (1 LSB) apart (DNL = 0). A DNL specification of greater than or equal to 1 LSB guarantees monotonicity.

Integral nonlinearity (INL) error: For data converters, INL is the deviation of an actual transfer function from a straight line. After nullifying offset and gain errors, the straight line is either a best-fit straight line or a line drawn between the end points of the transfer function. The INL is often called *relative accuracy*.

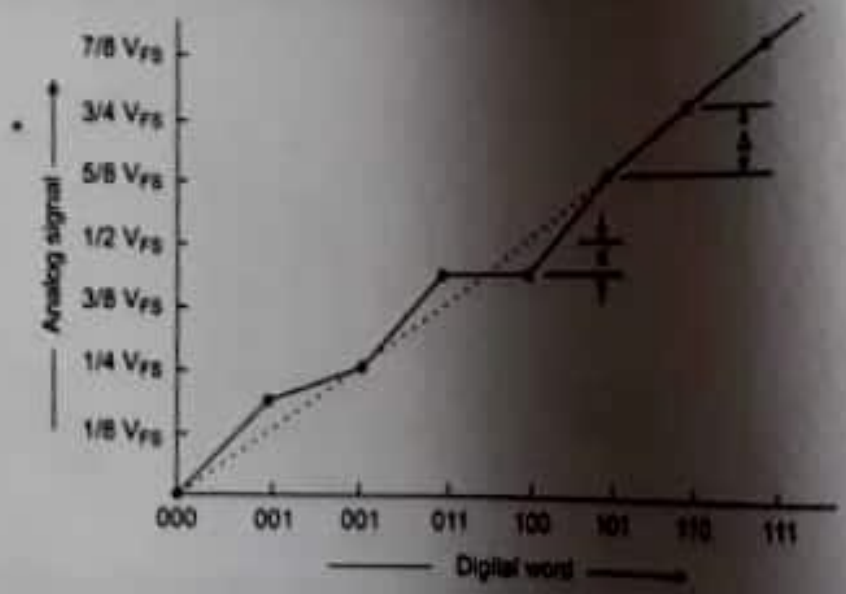


Fig.13.3 Linearity error of a 3-bit D/A converter

Monotonicity

A D/A converter is monotonic if its output value increases as the binary inputs are incremented from one value to the next. That is, the analogue output can have no downward step as the binary input is incremented. Figure 13.3 shows the transfer curve for a non-monotonic D/A converter. The output decreases when the input word changes from 011 to 100.

Scanned by CamScanner

(iv) a *discrete time signal* is the one that is defined at particular points of time only. Therefore, the independent time variable is quantized. When the amplitude of a discrete-time signal is allowed to assume a continuous range of values, the function is called a *sampled-data* signal. A sampled data signal could result from sampling an analog signal at discrete points of time.

(v) a *digital signal* is a function, in which the time and amplitude are quantized. A digital signal is always represented by a sequence of *words*, where each word can contain a finite number of *bits* (binary digits).

A D/A converter (DAC) converts digital data into its equivalent analog data. The analog data is required to drive motors and other analog devices. An A/D converter (ADC) converts analog data into its equivalent digital data i.e., binary data. The A/D converter and D/A converter are also called *data converters* and they are also available as monolithic integrated circuits.

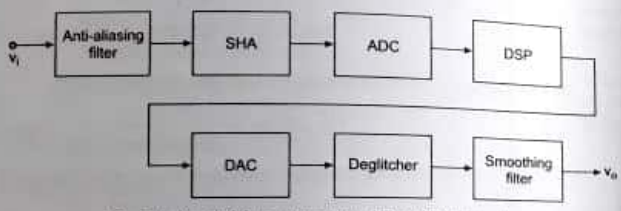


Fig. 13.1 Sampled data system using A/D and D/A converters

Figure 13.1 shows a typical application in which A/D and D/A conversions are employed. An analog input signal from a transducer is band-limited by *anti-aliasing* filter. The signal is then sampled at a frequency rate higher than twice the maximum frequency of the band-limited input signal. That is, when the A/D converter is operated at a rate of f_s samples/second, the highest frequency component of the input signal can be less than $f_s/2$. The A/D converters normally require the input to be held constant during the conversion process. Hence, a *Sample-and-Hold Amplifier* (SHA) is introduced in the loop as shown in Fig. 13.1 before A/D converter. The SHA freezes the band-limited signal just before the start of each conversion. The digital signal from A/D converter may be processed, transmitted and recorded in digital form by the digital signal processor (DSP) block. Then, the digital signal is converted into analog signal by D/A converter for use in analog form. The D/A converter is usually operated at the same frequency f_s as that of A/D converter. The output of D/A converter is commonly a staircase signal, which is passed through a smoothing filter to eliminate the quantization noise effects. A *deglitcher* may be introduced in the loop before the smoothing process, to remove any output glitches generated during input code variations.

The schematic structure shown in Fig. 13.1 is prevalent either in full or in part in numerous applications such as digital signal processing, direct signal control, digital audio mixing, music and video synthesis, pulse-code modulation (PCM) communication, data acquisition and digital microprocessor based instrumentation.

13.3 SPECIFICATIONS OF D/A CONVERTER

The important specifications, namely, accuracy, offset voltage, monotonicity, resolution and settling time of D/A converter are discussed below.

Accuracy

The components in D/A converter circuits are prone to mismatches, drift, ageing, noise and other sources of errors. These factors lead to degradation in conversion performance.

Absolute accuracy defines the maximum deviation of the output from the ideal value and it is expressed in fractions of 1 LSB. The D/A converter manufacturers follow different ways of specifying accuracy. The D/A converter errors are classified as *static* and *dynamic errors*.

Offset Voltage

The simplest kind of static errors are *offset error* and *gain error*. Ideally, the output of a D/A converter is 0V when all the bits of binary input word are 0s. In practice, however, there is a very small output voltage called *offset voltage* or *offset error* as depicted in Fig. 13.2(a). The offset error is nullified by translating the actual A/D converter characteristics up or down so that it goes through the origin as shown in Fig. 13.2(b). The gain error shown in Fig. 13.2(b) is compensated by adjusting the scale factor K.

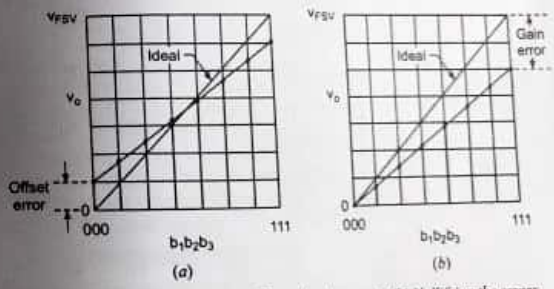


Fig. 13.2 (a) D/A converter offset and gain errors; (b) Nullifying the errors

Linearity

The most common dynamic errors are *full-scale error* and *linearity error*. *Full-scale error* is the maximum deviation of the output value from its expected or *ideal* value, expressed in percentage of full-scale. *Linearity error* is the maximum deviation in step size from the ideal step size. More expensive D/A converters have full-scale and linearity errors as low as 0.001% of full-scale. General purpose D/A converters have accuracies in the range of 0.01 to 0.1%.

Scanned by CamScanner

Chapter

13

D/A and A/D Converters

13.1 INTRODUCTION

The data converters convert one form of data into another form. Real world processes produce analog signals which carry information pertaining to process variables such as voltage, current, charge, temperature and pressure. The rate of flow of such information may be very slow or very fast. It is difficult to store, manipulate, compare, calculate and retrieve such data with good accuracy using purely analog technology. But, computers can perform these operations quickly and efficiently using digital techniques. Therefore, it is necessary to convert the analog signals from various transducers into its equivalent digital data, which in turn act as the input for digital systems. Thus the requirement for converting analog signal into digital data emerged. The computers also need to communicate with people and physical processes through the use of analog signals, which necessitated the process of digital to analog conversion. This chapter discusses the most common D/A and A/D conversion techniques, D/A converter and A/D converter ICs, delta modulators and demodulators followed by sigma-delta converters and widely used ICs for D/A and A/D conversions.

13.2 ANALOG AND DIGITAL DATA CONVERSIONS

In the application of signal processing, the measurement and analysis of signals are very important to discover their characteristics. If the signal is unknown, the process of analysis begins with the acquisition of the signal. The most common technique of acquiring signals is by *sampling*. Sampling a signal is the process of acquiring its values only at discrete points in time.

The definitions related to the process of sampling and the subsequent analog and digital conversion processes are

(i) an *analog signal* is a signal that is defined over a continuous period of time, and in which the amplitude may assume a continuous range of values.

(ii) the term *quantization* refers to the process of representing a variable by a finite set of discrete values.

(iii) a *quantized variable* is the signal variable that can assume only finite distinct values.

Scanned by CamScanner

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,

$$t_C \text{ for SAC} = N \times 1 \text{ clock cycle}$$

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.

Scanned by CamScanner

1st 2nd
1000 → 1000 → 1100 → 1000
↓ 3rd
1010 ← 1011 ← 1010
4th

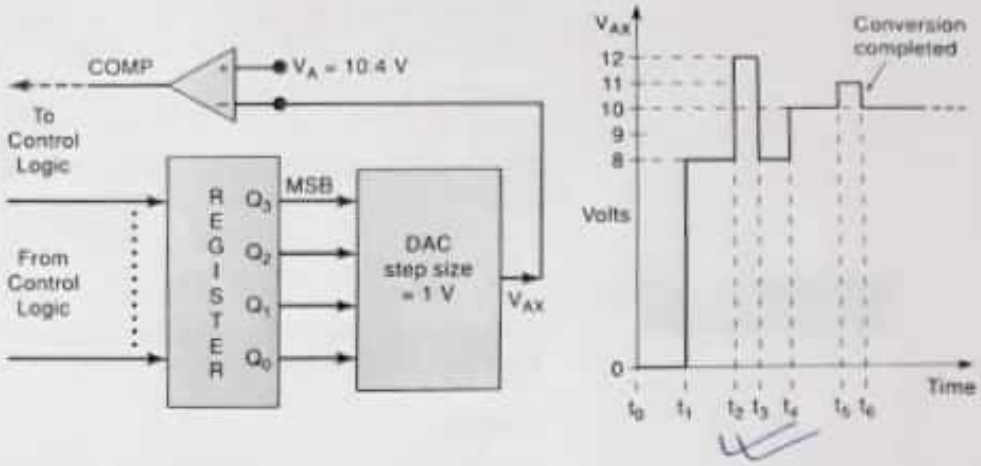


FIGURE 11-19 Illustration of four-bit SAC operation using a DAC step size of 1 V and V_A = 10.4 V.

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 *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 successive-approximation 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 $108_{10} = 01101100_2$.

Adv

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.

Scanned by CamScanner

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.

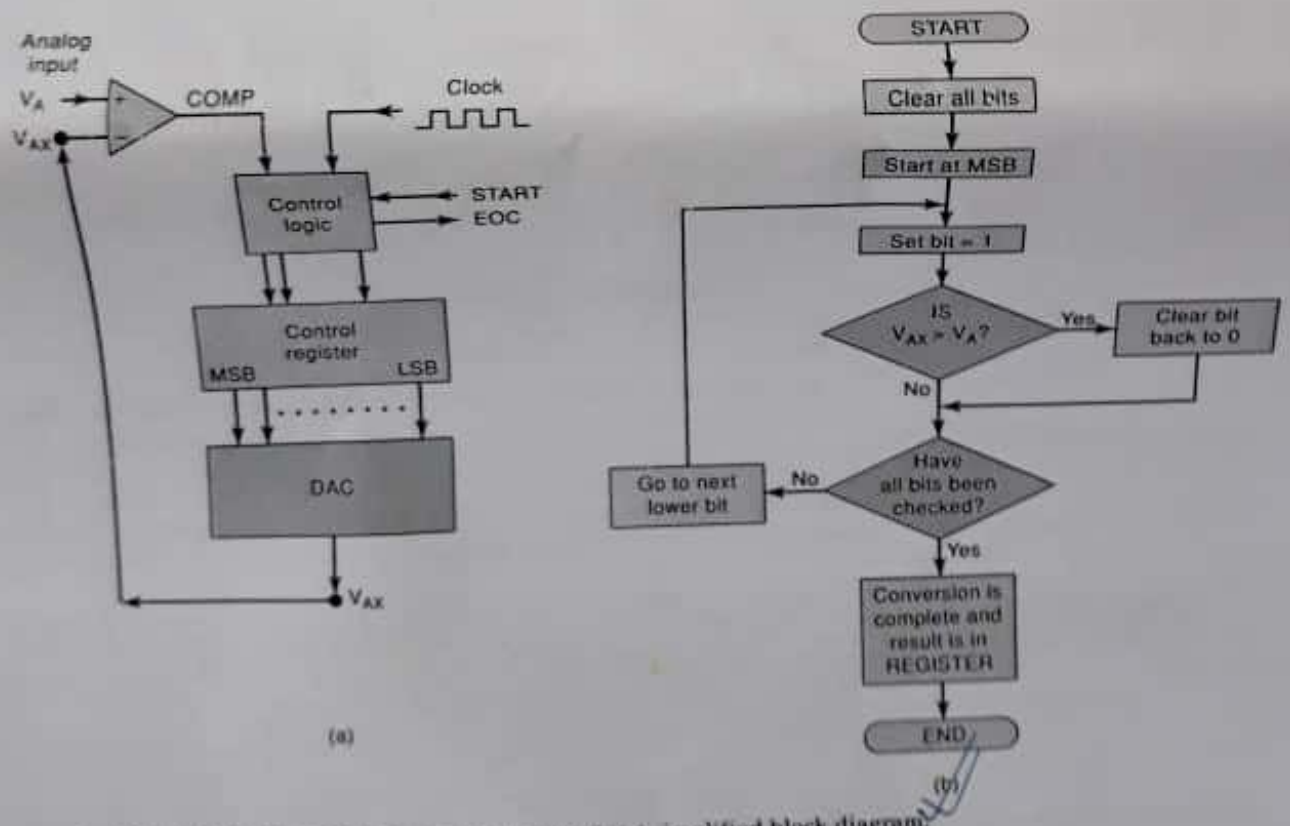


FIGURE 11-18 Successive-approximation ADC: (a) simplified block diagram; (b) flowchart of operation.

Scanned by CamScanner

From the Table 13.2, it is clear that the logic expressions for b_3, b_2 and b_1 are complex due to their dependence on seven input variables ($C_1, C_2 ... C_7$). Hence, the coding circuit is implemented using a priority encoder. The IC 74148 is an 8-to-3 priority

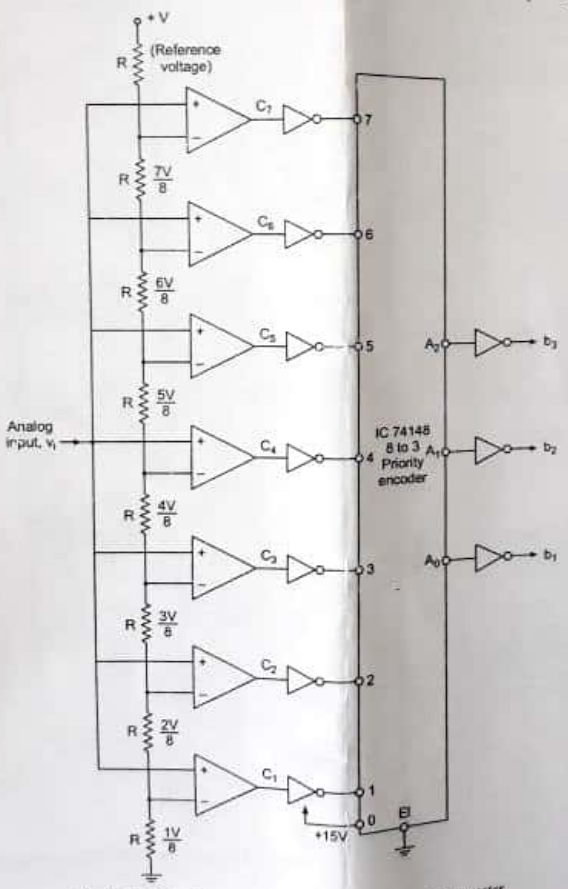


Fig. 13.18 Logic diagram of a 3-bit simultaneous type A/D converter

encoder with active LOW inputs and outputs. Since the comparator outputs are active HIGH, they are connected to the inputs of encoder through inverters and the outputs of encoder are inverted once again to get active HIGH digital outputs b_3, b_2 and b_1 as shown in Fig. 13.18.

Advantages

(i) Simultaneous type A/D converter is the fastest because A/D conversion is performed simultaneously through a set of comparators. Hence, it is also called *flash type A/D* converter. Typical conversion time is 100ns or less.

(ii) The construction is simple and easier to design.

Disadvantages

The simultaneous type A/D converter is not suitable for A/D conversion with more than 3 or 4 digital output bits. Then $(2^n - 1)$ comparators are required for an n-bit A/D converter and the number of comparators required doubles for each added bit.

13.9.2 Counter Type A/D Converter

The counter type A/D converter is constructed using only one comparator with a variable reference voltage. The variable reference voltage can be obtained by a sequence or binary counter and a D/A converter. The block diagram for an n-bit counter type A/D converter is shown in Fig. 13.19 (a).

The operation of the counter type A/D converter is as follows. The n-bit binary counter is initially set to 0 by the *Reset* switch which is normally active LOW. Therefore, the digital output is zero and the analog equivalent V_r is also 0. When Reset signal is released (HIGH), the clock pulses gated through the AND gate are counted by the binary counter. The D/A converter converts the digital output to an analog voltage and connects it as the inverting input to the comparator. The output of the comparator enables the AND gate to pass the clock. The number of counted pulses increases with time and the analog output V_r from DAC is a rising staircase waveform as shown in Fig. 13.19(b).

The counting will continue until the reference voltage V_r equals and just rises more than V_i. Then the comparator output becomes LOW and this disables the AND gate from passing the clock. The counting stops at the instance $V_i < V_r$ and at that instant the digital output of the comparator represents the analog input voltage V_i. Then the clock is inhibited, the counter stops its progress and the conversion is said to be complete. The numbers stored in the n-bit counter is the equivalent n-bit digital data for the given analog input voltage.

In this A/D converter, the counter advances by one count for every clock pulse, and therefore, the clock speed decides the conversion speed. For example, if a 100 kHz clock is used in an 8-bit A/D converter, the counter advances for every step and it will take 2.56 ms (i.e., $2^8 \times \frac{1}{100\text{ kHz}} = 2.56$ ms) to reach the full-scale digital output (i.e., $2^8 \times 10\ \mu s = 256$ ms). Normally, the time required to reach one half of the full-scale voltage is called *average conversion* time. Hence, the average conversion time of the above A/D converter is 1.28 ms.

Scanned by CamScanner

The output of a comparator is in *positive* saturation state when the voltage at the noninverting input terminal is more than the voltage at the inverting terminal and it is in *negative* saturation state otherwise. When the analog input voltage is less than $V/4$, the voltage at the noninverting terminals of the three comparators is less than their respective inverting input voltages, and hence, the comparator outputs are $C_1C_2C_3 = 000$. When the analog input is between $V/4$ and $V/2$, the comparator outputs are $C_1C_2C_3 = 100$. Table 13.1 shows the comparator outputs for different ranges of analog voltage and their corresponding digital outputs.

Table 13.1 : Comparator and digital outputs for a 2-bit simultaneous type A/D converter

Analog input voltage (V_i)	Comparator Outputs			Digital Outputs	
	C_1	C_2	C_3	b_2	b_1
$0 \le V_i \le V/4$	0	0	0	0	0
$V/4 \le V_i \le V/2$	1	0	0	0	1
$V/2 \le V_i \le 3V/4$	1	1	0	1	0
$3V/4 \le V_i \le V$	1	1	1	1	1

Since there are four ranges of analog input voltages, this can be coded using a 2 bit digital output (b_2, b_1) as shown in Table 13.1. The coding circuit for encoding the three comparator outputs into two digital outputs is shown inside the dotted square of Fig. 13.16 using the simplified expressions for b_1 and b_2 as discussed below.

From the Table 13.1, the logic expressions for b_2 and b_1 can be written as

$$b_2 = C_1C_2\overline{C_3} + C_1C_2C_3 = C_1C_2\left(\overline{C_3} + C_3\right) = C_1C_2 \tag{13.20}$$

$$b_1 = C_1\overline{C_2C_3} + C_1C_2C_3 = C_1\left(\overline{C_2 \oplus C_3}\right) \tag{13.21}$$

Similarly, a 3-bit A/D converter can be constructed using seven ($2^3 - 1$) comparators as shown in Fig. 13.17. The comparator and digital outputs for eight different ranges of analog input voltage are given in Table 13.2.

Table 13.2 : Comparator and digital outputs for a 3-bit simultaneous type A/D converter

Analog input voltage (V_i)	Comparator Outputs							Digital Outputs		
	C_1	C_2	C_3	C_4	C_5	C_6	C_7	b_3	b_2	b_1
$0 \le V_i \le V/8$	0	0	0	0	0	0	0	0	0	0
$V/8 \le V_i \le 2V/8$	1	0	0	0	0	0	0	0	0	1
$V/8 \le V_i \le 3V/8$	1	1	0	0	0	0	0	0	1	0
$V/8 \le V_i \le 4V/8$	1	1	1	0	0	0	0	0	1	1
$V/8 \le V_i \le 5V/8$	1	1	1	1	0	0	0	1	0	0
$V/8 \le V_i \le 6V/8$	1	1	1	1	1	0	0	1	0	1
$V/8 \le V_i \le 7V/8$	1	1	1	1	1	1	0	1	1	0
$V/8 \le V_i \le V$	1	1	1	1	1	1	1	1	1	1

Scanned by CamScanner

...ven analog signal with an internally ...ash (comparator) type A/D converter, Counter ...ecking or Servo operated A/D converter and Successive approximation A/D converter are direct type of A/D converters.

The integrating type of A/D converters performs the A/D conversion in an indirect manner. It is a special class of converter which uses either a reference voltage, or which integrates the signal during the conversion process. Therefore, they do not require S/H circuit at the input. The process of integrating the signals also improves the signal-to-noise ratio for certain type of analog signals. The integrating A/D converters are suitable only for very low frequency signals. For example, many of the digital voltmeters employ an integrating A/D converter in the circuit. The charge balancing type and dual slope A/D converters fall in this category.

13.9.1 Simultaneous Type (Flash type) A/D Converter

The simultaneous type A/D converter is based on comparing an unknown analog input voltage with a set of reference voltages. To convert an analog signal into a digital signal of n output bits $(2^n - 1)$ number of comparators are required. For example, a 2-bit A/D converter requires 3 or $(2^2 - 1)$ comparators, while a 3-bit converter needs 7 or $(2^3 - 1)$ comparators. The block diagram of a 2-bit simultaneous type A/D converter is shown in Fig. 13.16.

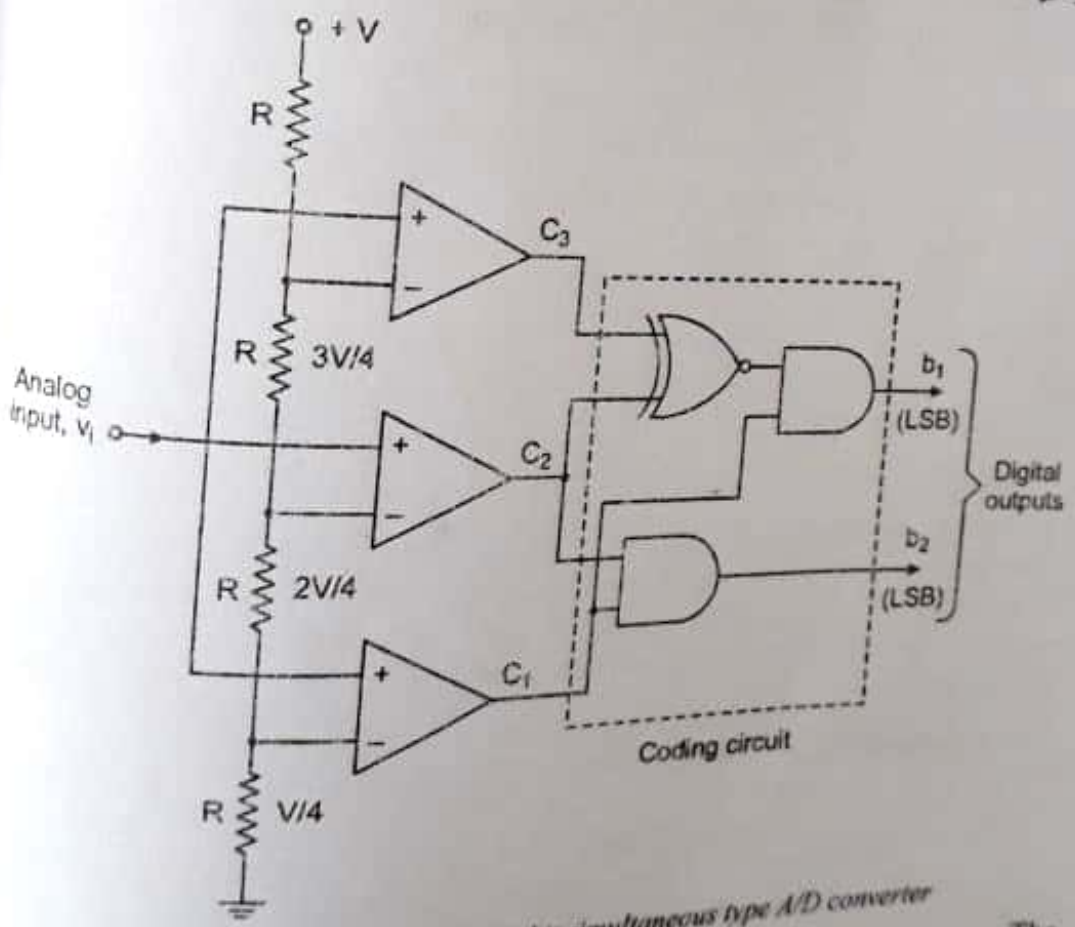


Fig. 13.16 Block diagram of a 2-bit simultaneous type A/D converter

As shown in Fig. 13.16, the three op-amps are used as comparators. The noninverting inputs of all the three comparators are connected to the analog input voltage. The inverting input terminal of the op-amps are connected to a set of reference voltages V/4, 2V/4 and 3V/4 respectively, which are obtained using a resistive divider network and power supply +V.

Scanned by CamScanner

13.4 BASIC D/A CONVERSION TECHNIQUES

The D/A converter converts digital or binary data into its equivalent analog value. The input digital data for a D/A converter is an n-bit binary word D. The bit b_1 is called the most significant bit (MSB) and bit b_n the least significant bit (LSB). Then, the quantity D can be represented by $D = b_1 2^{-1} + b_2 2^{-2} + b_3 2^{-3} + \ldots + b_n 2^{-n}$. The D/A converter accepts the binary input D and produces an analog output, which is proportional to D using a reference voltage V_R. The converted analog value is either in voltage or current form. For a voltage output D/A converter, the conversion characteristic may be expressed as

$$V_o = KV_{FS} = KV_{FS}\left(b_1 2^{-1} + b_2 2^{-2} + b_3 2^{-3} + \ldots + b_n 2^{-n}\right) \tag{13.3}$$

where V_o = output voltage

V_{FS} = full-scale range of voltage

K = scaling factor, usually unity

$b_1 \ldots b_n$ = n-bit binary fractional word with binary point located at the left

b_1 = most significant bit (MSB) of weight $V_{FS}/2$

b_n = least significant bit (LSB) of weight $V_{FS}/2^n$

The symbolic representation of an n-bit D/A converter is shown in Fig. 13.4.

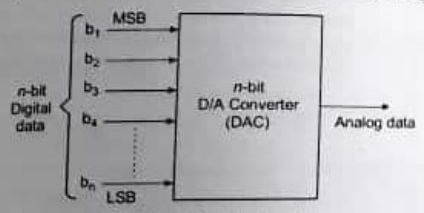


Fig. 13.4 n-bit D/A converter

13.4.1 Weighted Resistor Type D/A Converter

In the weighted resistor type D/A converter, each digital level is converted into an equivalent analog voltage or current. In a 4-bit D/A converter which accepts data from 0000 to 1111, there are 15 discrete levels of input, and hence it is convenient to divide the output analog signal into 15 levels. The LSB of the digital data causes a change in the analog output that is equal to 1/15th of the full-scale analog output voltage (V_R). Therefore, the weighted resistor network is designed in such a way that a 1 in LSB (2^0) position results in $V_R \times 1/15$ at the output. A 1 in the 2^1 bit position must cause a change in the analog output voltage that is equal to 2/15th of V_R (i.e., twice the size of the LSB). Similarly, a 1 in 2^2 and 2^3 bit positions must cause a change of $V_R \times 4/15$ V and $V_R \times 8/15$ V respectively as the analog output. It is important to note that the sum of the weights assigned to various bit positions of a 4-bit D/A converter must be equal to 1,

i.e., $(1/15 + 2/15 + 4/15 + 8/15 = 15/15)\ V_R = V_{FS}$. In general, the weight assigned to the LSB is $1/(2^n - 1)$, where n is the number of bits in the digital input.

Thus, the 4-bit weighted resistor network shown in Fig. 13.5 (a) performs the following D/A conversion:

1. The 2^0 bit is changed to 1/15th of V_R, 2^1 bit to 2/15th of V_R, 2^2 bit to 4/15th of V_R and 2^3 bit to 8/15th of V_R.
2. These four voltages are added together to form the analog output voltage using an op-amp summer circuit.

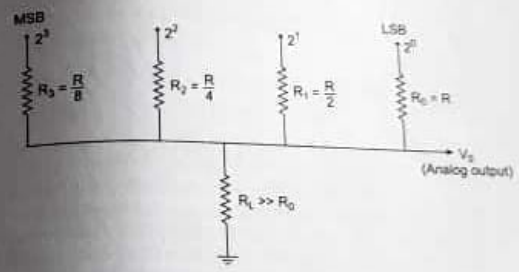


Fig. 13.5(a) Four-bit weighted resistor D/A converter

The resistor R_o, R_1, R_2 and R_3 form the voltage divider network connected with the op-amp and R_L is the load resistor which should be larger enough so as not to load the divider network. The LSB should be connected with the highest input resistance R_o while the 2^1 bit is connected with a resistance of half the value of LSB resistor, i.e. $R_o/2$. Therefore, its current contribution at the summing junction of op-amp will be twice that of LSB. The 2^2 bit is connected with a 1/4th of LSB resistance $R_o/4$. Similarly, the MSB is connected with 1/8th of the LSB resistance $R_o/8$. The output is the sum of these four attenuated voltages.

The operating principle of the circuit is explained with the following illustration.

Illustration: The equivalent circuit of Fig. 13.5 (a), when applied with the digital data of 0001, is shown in Fig. 13.5 (b). The analog output voltage V_o can be calculated using Millman's theorem, which states that the voltage at any node in a resistive network is equal to the sum of the currents entering the node divided by the sum of the conductance connected at the node. Then the output voltage is expressed as

$$V_o = \frac{\dfrac{V_R}{R_1} + \dfrac{V_R}{R_2} + \dfrac{V_R}{R_3} + \dfrac{V_R}{R_4}}{\dfrac{1}{R_1} + \dfrac{1}{R_2} + \dfrac{1}{R_3} + \dfrac{1}{R_4}} \tag{13.4}$$

Scanned by CamScanner

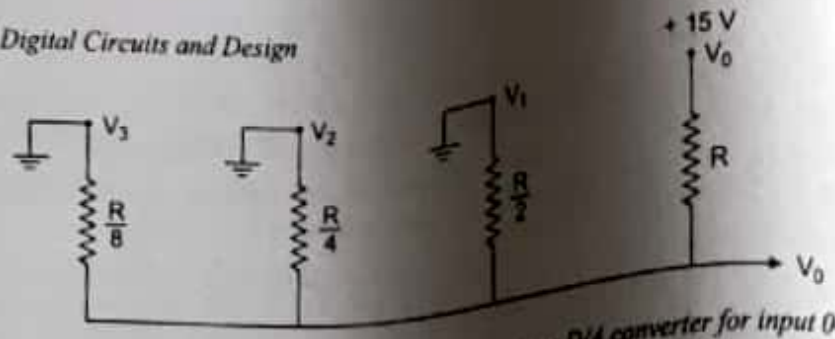


Fig. 13.5(b) Equivalent circuit of a 4-bit weighted resistor D/A converter for input 0001

For the weighted resistor network assuming $R_1 = R$, $R_2 = R/2$, $R_3 = R/4$ and $R_4 = R/8$, and applying the Millman's theorem to the circuit of Fig. 13.5 (b), we get

$$V_o = \frac{\frac{V_R}{R} + \frac{V_R}{R/2} + \frac{V_R}{R/4} + \frac{V_R}{R/8}}{\frac{1}{R} + \frac{1}{R/2} + \frac{1}{R/4} + \frac{1}{R/8}} \tag{13.5}$$

Figure 13.6 (a) shows the circuit of an n-bit D/A converter using op-amp as a summing amplifier. It employs a binary-weighted resistor network to generate the terms $b_i 2^{-i}$ where $i = 1, 2, \ldots n$. The circuit also uses n-electronic switches controlled by the binary input word $b_1, b_2, \ldots b_n$ and a reference voltage V_R. The switches are of single pole double throw (SPDT) type. If the binary input to a switch is 1, then the switch connects the resistance to the reference voltage $-V_R$. When the input bit to the switch is 0, it connects the resistor to ground.

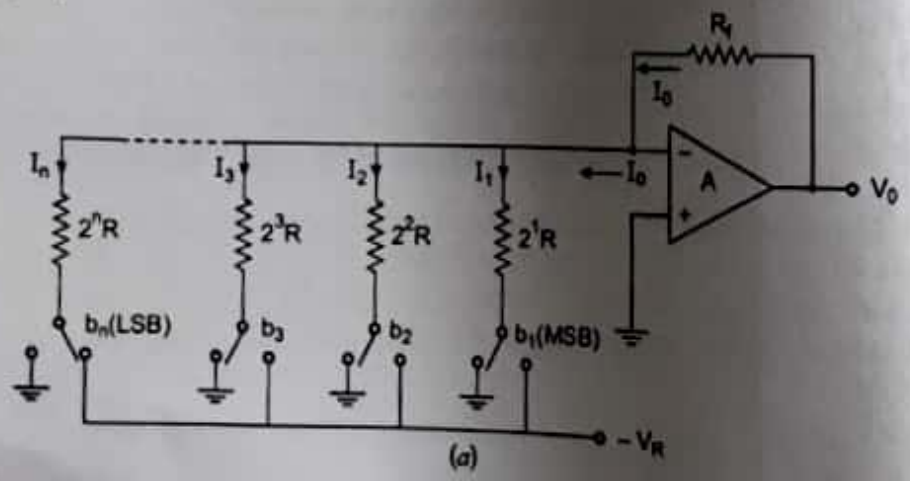
Considering an ideal amp A, the output current I_o is given by

$$I_o = I_1 + I_2 + \ldots + I_n = \frac{V_R}{2^1 R} b_1 + \frac{V_R}{2^2 R} b_2 + \ldots + \frac{V_R}{2^n R} b_n$$

$$= \frac{V_R}{R}[b_1 2^{-1} + b_2 2^{-2} + \ldots + b_n 2^{-n}]$$

Then the output voltage $V_o = I_o R_f = \frac{V_R}{R}[b_1 2^{-1} + b_2 2^{-2} + \ldots + b_n 2^{-n}]R_f$ (13.6)

Using Eqn. (13.3) and Eqn. (13.6), it can be seen that if $R_f = R$, then $K = 1$ and $V_{FS} = V_R$.



(a)

Scanned by CamScanner

The main disadvantage of binary weighted D/A converter is the requirement of wide range of resistor values. As the length of the binary word is increased, the range of resistor values needed also increases. For an 8-bit D/A converter, the resistor values to be connected with the bits are $2^0 R + 2^1 R + ... + 2^7 R$. Therefore, the largest resistor corresponding to bit b_8 is 128 times the value of the smallest resistor corresponding to bit b_1. The fabrication of such large value of resistors of the order of MΩ is not practically possible in monolithic circuit fabrication. In addition, the voltage drop variations across such high value resistors due to the bias currents affect the accuracy. Therefore, the limitations in achieving and maintaining resistor ratios restrict the use of weighted resistor D/A converters to below 8-bit of word length.

The R-2R ladder type D/A converter is a better choice for practical applications and it overcomes such drawbacks.

Example 13.1 A system uses a 12-bit word to represent the input signal. If the maximum peak-to-peak voltage at the output is set for 4V, find the resolution of the system and the dynamic range.

Solution The 12-bit word can represent 2^{12} or 4096 levels, which are equally spaced across the 4V range. Then, the step size $= \frac{4V}{4096} = 976\ \mu V$. Therefore the system can identify input changes as low as 976 μV.

The *dynamic range* gives the ratio of the largest value to the smallest value which can be converted. Therefore, the dynamic range $= \frac{4V}{976\mu V} = 4096$. The dynamic range in dB is given by $20 \log_{10} 4096 = 72$ dB.

Example 13.2 An 8-bit D/A converter has an output of voltage range 0 to 2.55V. Find the resolution of the system.

Scanned by CamScanner

4.3.5 Noise Immunity or Noise Margin

The term noise denotes an unwanted signal voltage, e.g., hum, transients and glitches.

Noise can sometimes cause the input voltage of a logic gate to drop below V_{IH} (min) or rise above V_{IL} (max), which leads to unreliable operation. *Noise immunity* is the maximum noise voltage that may appear at the input of a logic gate without changing the logical state of its output. A quantitative measure of noise immunity is called *noise margin*.

The difference between the operating input-logic voltage level and the threshold voltage is called the noise margin of the circuit. The manufacturer usually quotes the noise margin, which refers to the amplitude of the noise voltage that may cause the logic level to change. In the worst case, a TTL gate functions properly as long as the noise margin is kept less than 0.4V.

4.3.6 Operating Temperature

All IC gates are semiconductor devices that are temperature-sensitive by nature. The operating temperature ranges for an IC vary from 0°C to + 70°C for consumer and industrial applications and from–55°C to + 125°C for military applications.

4.3.7 Power Supply Requirements

The amount of power and supply voltage required by an IC are the main parameters to be taken into consideration while choosing a proper power supply.

4.3.8 Current and Voltage Parameters

The following currents and voltages are very important in designing digital systems. The values given below are for TTL gates only.

High-level input voltage (V_{IH}) [$V_{in(1)}$] It is the minimum voltage level required for a logical 1 at an input. Its minimum value is 2V.

Low-level input voltage (V_{IL}) [$V_{in(0)}$] It is the maximum input voltage required for a logical 0 (LOW) at an input. Its maximum value is 0.8V.

High-level output voltage (V_{OH}) [$V_{out(1)}$] It is the minimum voltage required for a logical 1 state at the output. Its minimum value is 2.4V.

Low-level output voltage (V_{OL}) [$V_{out(0)}$] It is the maximum voltage available at the circuit's output corresponding to the logical 0 state. Its maximum value is 0.4V.

High-level input current (I_{IH}) [$I_{in(1)}$] The current that flows through an input when a specified high-level voltage is applied to that input.

Low-level input current (I_{IL}) [$I_{in(0)}$] The current that flows through an input when a specified low-level voltage is applied to that input.

High-level output current (I_{OH}) [$I_{out(1)}$] The current that flows from an output in the logical 1 state under specified load conditions.

Low-level output current (I_{OL}) [$I_{out(0)}$] The current that flows from an output in the logical 0 state under specified load conditions.

Scanned by CamScanner

13.5 MULTIPLYING D/A CONVERTERS (MDACs)

An A/D converter which uses a varying analog reference voltage instead of a fixed reference voltage V_R is called a *multiplying* D/A converter. It is known from Eqn. (13.9) that the output is the product of the digital word and the analog reference voltage V_R. This arrangement can be used as a *programmable attenuator*, when the binary word is made to represent a value less than *unity*. Then the output is a fraction of the input reference voltage V_R.

The simplified functional diagram of a multiplying D/A converter IC is shown in Fig. 13.11. The input digital word D is applied to $b_1 ... b_n$ terminals. The reference voltage V_R is applied to circuit. The output currents I_o and I_o' are functions of the digital word D. The value of the output is thus programmable from 0 V to $(1 - 2^{-n})V_R$.

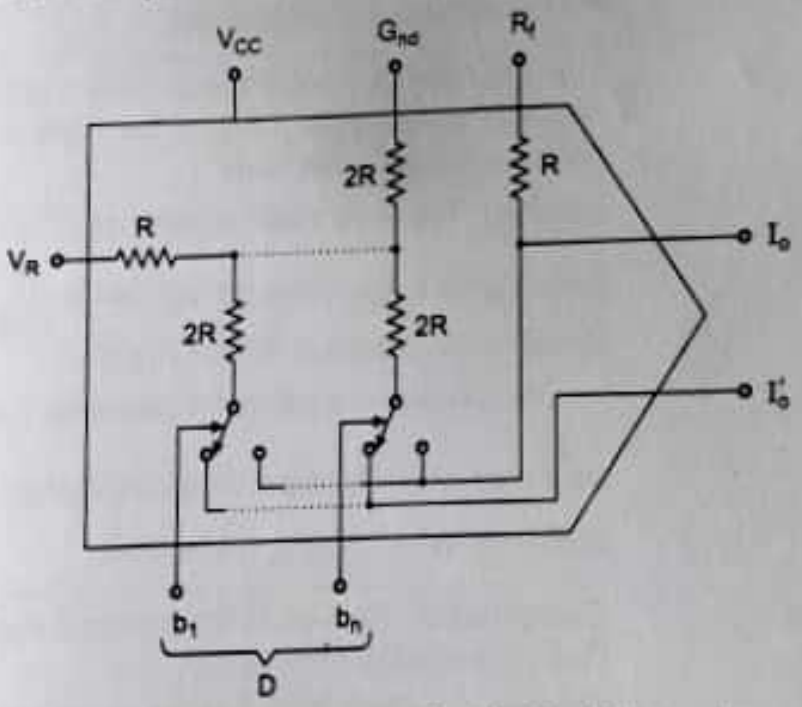


Fig. 13.11 Functional diagram of multiplying D/A converter

The reference voltage of a multiplying D/A converter can be varied over positive and negative values, including zero. This feature makes the multiplying D/A converters most suitable for digitally programmable applications such as programmable filters and oscillators.

13.6 SAMPLING PROCESS

Figure 13.12 shows the graphical representation of sampled input signal over time. The result of the sampling process is a series of the sampling instants and the amplitude of the signal at that instant of time. If the sampling is done at a constant rate, the resulting amplitude values may be used to reconstruct the signal. The accuracy of the conversion process will depend on two factors, namely, how frequently was the sampling done and the accuracy and resolution of the sample measurement.

Scanned by CamScanner

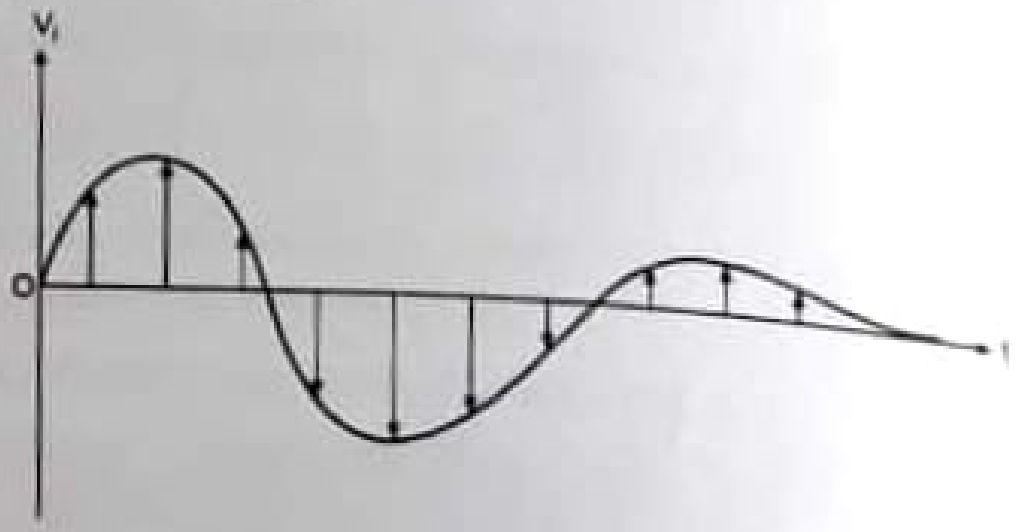


Fig. 13.12 Sampling the input signal over time

Shannon's sampling theorem

The minimum sampling rate used with any system bears great importance. Only a finite number of samples can be taken at any time, and a loss of information could result due to inadequate sampling frequency. The Shannon's sampling theorem establishes the theoretical basis for all the discrete sampling operations carried out on analog signals.

Shannon's sampling theorem states that if a signal is sampled for all time at a rate more than twice the highest frequency, it can be exactly reconstructed from the samples. If sampling is done at a rate higher than twice the highest frequency, the aliases do not overlap and the original signal can be recovered. If sampling is done at a slower rate than twice the frequency, then aliases would overlap. Assuming the signal has a spectrum with frequency components extending from dc to a maximum frequency of f_i Hz, the Shannon's sampling theorem states that the minimum possible sampling rate above which a bandpass signal can be recovered from the samples can be shown to be

$$f_s > 2f_i \text{ Hz}$$

For example, an audio signal ranging from dc to 20 kHz could theoretically be reconstructed by taking uniformly spaced samples at a rate of 40,000 samples/second. In practice, the sampling rate is always preferred to be higher than the theoretical minimum value, and normally, 3 to 4 times the highest frequency of the signal.

Example 13.7 A system employs a 16-bit word for representing the input signal. If the maximum output voltage is set for 2.1 V, [illegible] resolution of the system and its

Scanned by CamScanner

voltage with a ...
al of n output bits $(2^n - 1)$ number of comparators are required. For example, a 2-bit D converter requires 3 or $(2^2 - 1)$ comparators, while a 3-bit converter needs 7 or 2 - 1) comparators. The block diagram of a 2-bit simultaneous type A/D converter is wn in Fig. 13.16.

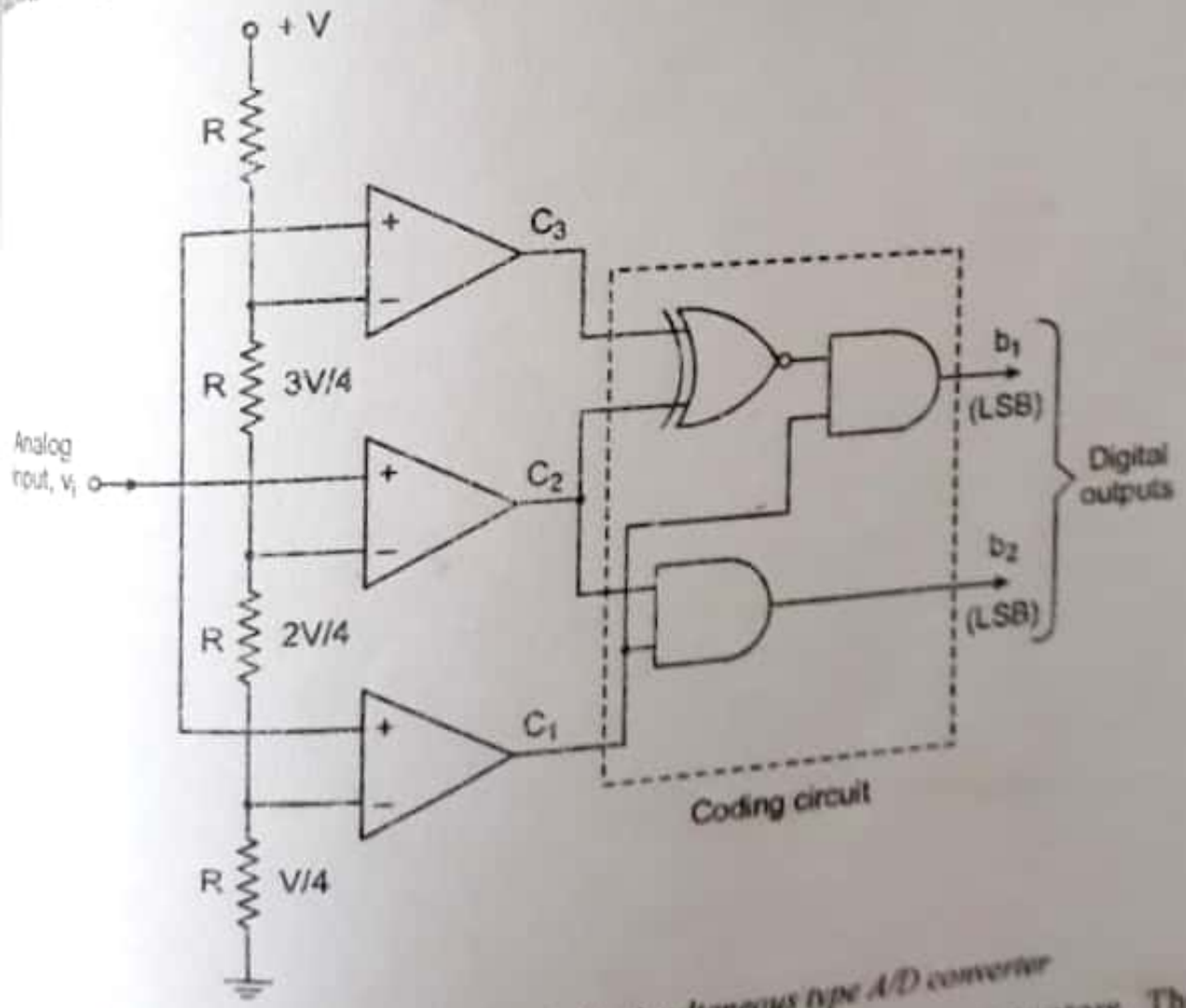


Fig. 13.16 Block diagram of a 2-bit simultaneous type A/D converter

As shown in Fig. 13.16, the three op-amps are used as comparators. The oninverting inputs of all the three comparators are connected to the analog input oltage. The inverting input terminal of the op-amps are connected to a set of reference oltages V/4, 2V/4 and 3V/4 respectively, which are obtained using a resistive divider etwork and power supply +V.

Scanned by CamScanner

(c) The digital output for an applied input voltage of 4.8V is given by

$$D = \frac{4.8V}{39.1mV} = 122.76 \cong 123$$

Converting this to binary gives the digital output for the 8-bit A/D converter to be 01111011.

Quantization Error

A digital error in an A/D converter is based on the resolution of the digital system. In A/D conversion, a continuous analog voltage is represented by an equivalent set of digital numbers. When the digital numbers are converted back to analog voltage by a D/A converter, the output is a staircase waveform, which is a discontinuous signal composed of a number of discrete steps. The smallest digital step is due to the LSB and it can be made smaller only by increasing the number of bits in the digital representation. This error is called *quantization error*, or *digitizing error* and it is commonly the bit.

As shown in Fig. 13.13 (b) the digital output is 011 for all values of $\frac{3}{8}V \pm \frac{1}{2}LSB$. Therefore there is an uncertainty about the exact value of v_i when the output is 011.

This uncertainty is called the *quantization error* and its value is $\pm\frac{1}{2}LSB$.

Increasing the number of bits of A/D converter results in finer resolution and smaller quantization error.

Analog Error

Analog error in an A/D converter is mainly due to variations in the dc switching point of the comparator. The variations in switching are mainly due to offset, gain and linearity error of the operational amplifier used in the comparator. The other sources of analog error are the resistors in the A/D converter, the reference voltage source and the ripple and noise introduced by the circuit components.

Linearity Error

This is an important measure of A/D converter performance. It is defined as a measure of the variation in voltage step size. This indicates the difference between the transitions for a minimum step of input voltage change. This is normally specified as a fraction of 1 LSB.

Differential Nonlinearity (DNL) Error

The analog input levels that trigger any two successive output codes should differ by 1 LSB (DNL = 0) for an A/D converter. Any deviation from 1 LSB value is defined as DNL error.

The counter type and continuous type A/D converters normally have better differential linearity than successive approximation type A/D converters.

Integral Nonlinearity (INL) Error

Figure 13.15 shows an actual A/D converter characteristic with a missing code. The dotted curve represents the locus of the midpoints of the actual input step voltage ranges. This line is called the *code center line*. The maximum deviation of the code center line from the straight line passing through the end points of the ideal characteristics after nulling the offset and gain errors is called *integral nonlinearity error* (INL).

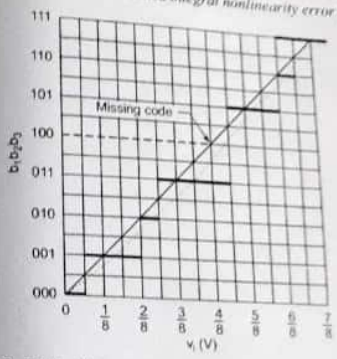


Fig. 13.15 *A/D converter characteristic with a missing code*

Dither

The performance of A/D converters can be improved using *dither* <*http://en.wikipedia.org/wiki/Dither*>. This is a very small amount of random noise (white noise <http://en.wikipedia.org/wiki/White_noise>) which is added to the input before A/D conversion. Its amplitude is set to half of the LSB value. Its effect is to cause the state of the LSB to randomly oscillate between 0 and 1 in the presence of very low levels of input, rather than sticking at a fixed value. Instead of the signal simply getting cut-off altogether at this low level (which is only being quantized to a resolution of 1 bit), it extends the effective range of signals that the A/D converter can convert, at the expense of a slight increase in noise. Thus, the quantization error is diffused across a series of noise values which is far less objectionable than a hard cut-off. The result is an accurate representation of the signal over time. A suitable filter at the output of the system can thus recover this small signal variation.

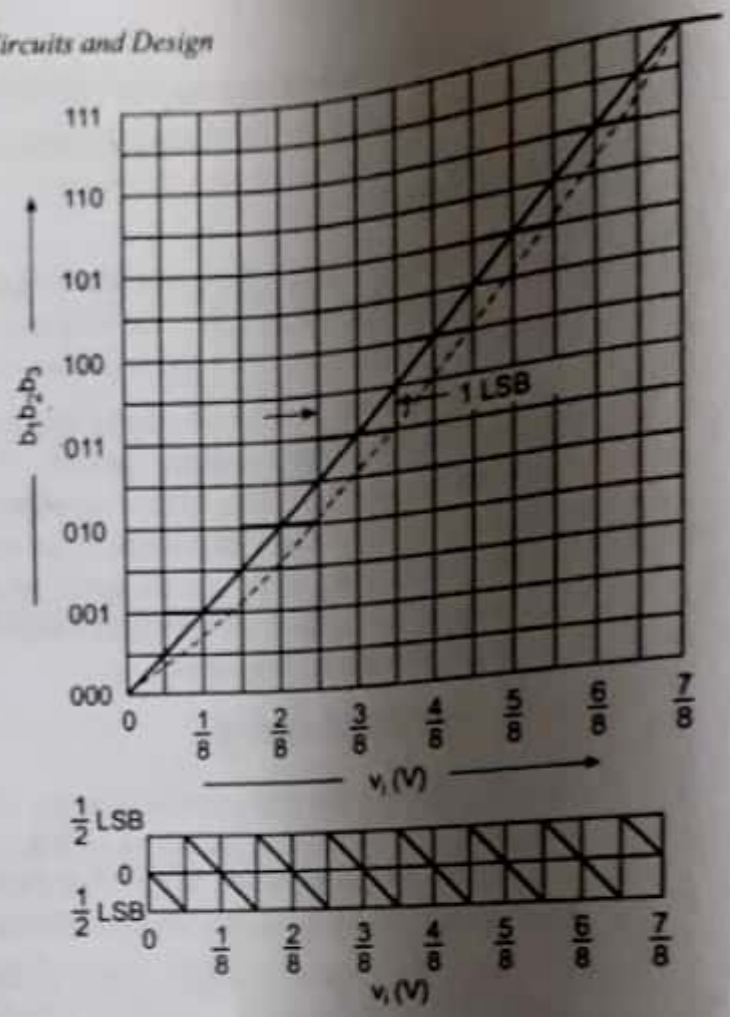
Conversion Time

The time required for an A/D converter to convert an analog input value into its equivalent digital data is called the *conversion time*.

Input Voltage Range

It is the range of voltage that an A/D converter can accept as its input without causing any overflow in the digital output.

Scanned by CamScanner



(b)

Fig. 13.13 A/D converter (a) Symbolic representation and (b) Ideal transfer characteristics and quantization noise for a 3-bit A/D converter

Figure 13.13 (b) shows the ideal characteristics of a 3-bit A/D converter with $V_{FS} = 1.0$ V where V_{FS} is the full-scale analog voltage. The A/D conversion process divides the analog input into 2^n intervals. These intervals are called *code ranges* and all the values of v_i falling within a code range are represented by the particular code. For instance, the code 110 corresponding to $V_i = 6/8\ V$ represents all inputs of value $6/8 + 1/16\ V$. Hence the output can err by $\pm (1/2)\ LSB$.

The general block diagram of an A/D converter is shown in Fig. 13.14. It consists of an antialiasing filter or prefilter, Sample-and-Hold amplifier, a quantizer and an encoder. The prefilter avoids the aliasing of high frequency signals. The Sample-and-Hold circuit holds the input analog signal into the A/D converter at a constant value during the conversion time. The quantizer segments the reference voltage signal into subranges. Typically, for an n-bit digital output code, there are 2^n subranges. The digital processor forms the encoder circuit which encodes the subrange into the corresponding digital bits. Therefore, the analog input signal is converted into an equivalent digital output code within the *conversion time*.

Scanned by CamScanner

saturated and non-saturated, bipolar families are classified into

(i) saturated logic and

(ii) non-saturated logic.

The following are the saturated bipolar logic families:

1. Resistor–Transistor Logic (RTL)
2. Direct-coupled Transistor Logic (DCTL)
3. Diode–Transistor Logic (DTL)
4. High Threshold Logic (HTL)
5. Transistor–Transistor Logic (TTL)
6. Integrated-injection Logic (I^2L)

The following are the non-saturated logic families:

1. Schottky TTL
2. Emitter-coupled Logic (ECL)

4.2.2 MOS Families

The MOS families include

1. PMOS *p*-channel MOSFETs
2. NMOS *n*-channel MOSFETs
3. CMOS Complementary MOSFETs

4.3 CHARACTERISTICS OF DIGITAL ICs

Some of the important parameters or properties of various logic families are listed as follows:

1. Speed of operation (Propagation delays)
2. Power dissipation
3. Fan-in
4. Fan-out
5. Noise immunity
6. Operating temperature
7. Power supply requirements

The comparison of performance of digital ICs may be made with reference to the above properties.

4.3.1 Speed of Operation

The speed of operation of an IC is expressed in terms of propagation delay. *Propagation delay* is defined as the time taken for the output of a gate to change after the inputs have changed.

A logic signal always experiences a delay in going through a circuit. The two propagation delay times shown in Fig. 4.1 are defined as follows:

t_{PLH}: It is the propagation delay time in going from logical LOW (0 state) to logical HIGH (1 state).

Scanned by CamScanner

8.2 ASYNCHRONOUS (RIPPLE or SERIAL) COUNTER

The asynchronous counter is the simplest in terms of logical operations, and is therefore the easiest to design. In this counter, all the flip-flops are not under the control of a single clock. Here, the clock pulse is applied to the first flip-flop, i.e. the least significant bit stage of the counter, and the successive flip-flop is triggered by the output of the previous flip-flop and thus the counter has a cumulative settling time. Hence, its speed of operation is limited. The first stage of the counter thus switches first on the

Scanned by CamScanner

application of a clock pulse to the first flip-flop and the successive stages change their states in turn causing a *ripple-through* effect of the count pulses. As the triggers move through the flip-flops like a ripple, it is called a *ripple counter*.

Fig. 8.1 shows a 4-bit binary ripple counter. As shown, a binary ripple counter is constructed using clocked *JK* flip-flops. The system clock, a square wave, drives flip-flop *A*. The output of *A* drives flip-flop *B*, the output of *B* drives flip-flop *C*, and the output of *C* drives flip-flop *D*. The overall propagation delay time of the counter is the sum of the individual delays of flip-flops. All the *J* and *K* inputs are connected to V_{CC} (1), which means that each flip-flop toggles on the negative edge of its clock input.

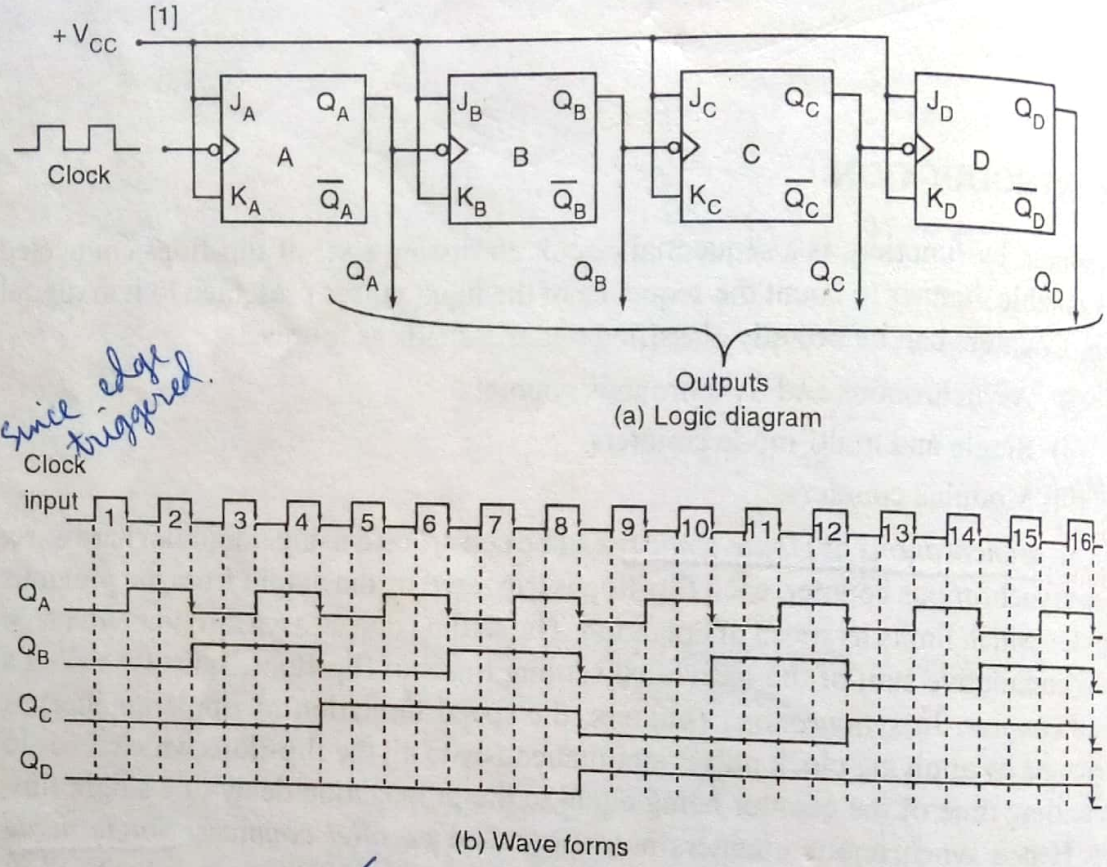


Fig. 8.1 *4-bit binary ripple counter*

Consider initially all flip-flops to be in the logical 0 state (i.e. $Q_A = Q_B = Q_C = Q_D = 0$). A negative transition (1 to 0) in clock input which drives flip-flop *A* causes Q_A to change from logical 0 to logical 1. Flip-flop *B* does not change its state since it also requires negative transition at its clock input, i.e. it requires its clock input (Q_A) to change from logical 1 to logical 0. With the arrival of the second clock pulse to flip-flop *A*, Q_A goes from 1 to 0. This change of state creates the negative going edge needed to trigger flip-flop *B*, and thus Q_B goes from 0 to 1. Before the arrival of the sixteenth clock pulse, all the flip-flops are in the logical 1 state. Clock pulse 16 causes Q_A, Q_B, Q_C and Q_D to go to logical 0 state in turn.

Table 8.1 shows the sequence of binary states that the flip-flops will follow as clock pulses are applied continuously. An *n*-bit binary counter repeats the counting sequence for every 2^n (n = number of flip-flops) clock pulses and has discrete states from 0 to $2^n - 1$.

Scanned by CamScanner

Table 8.1 *Truth table of 4-bit binary ripple counter*

State	Q_D	Q_C	Q_B	Q_A
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1
0	0	0	0	0

Scanned by CamScanner

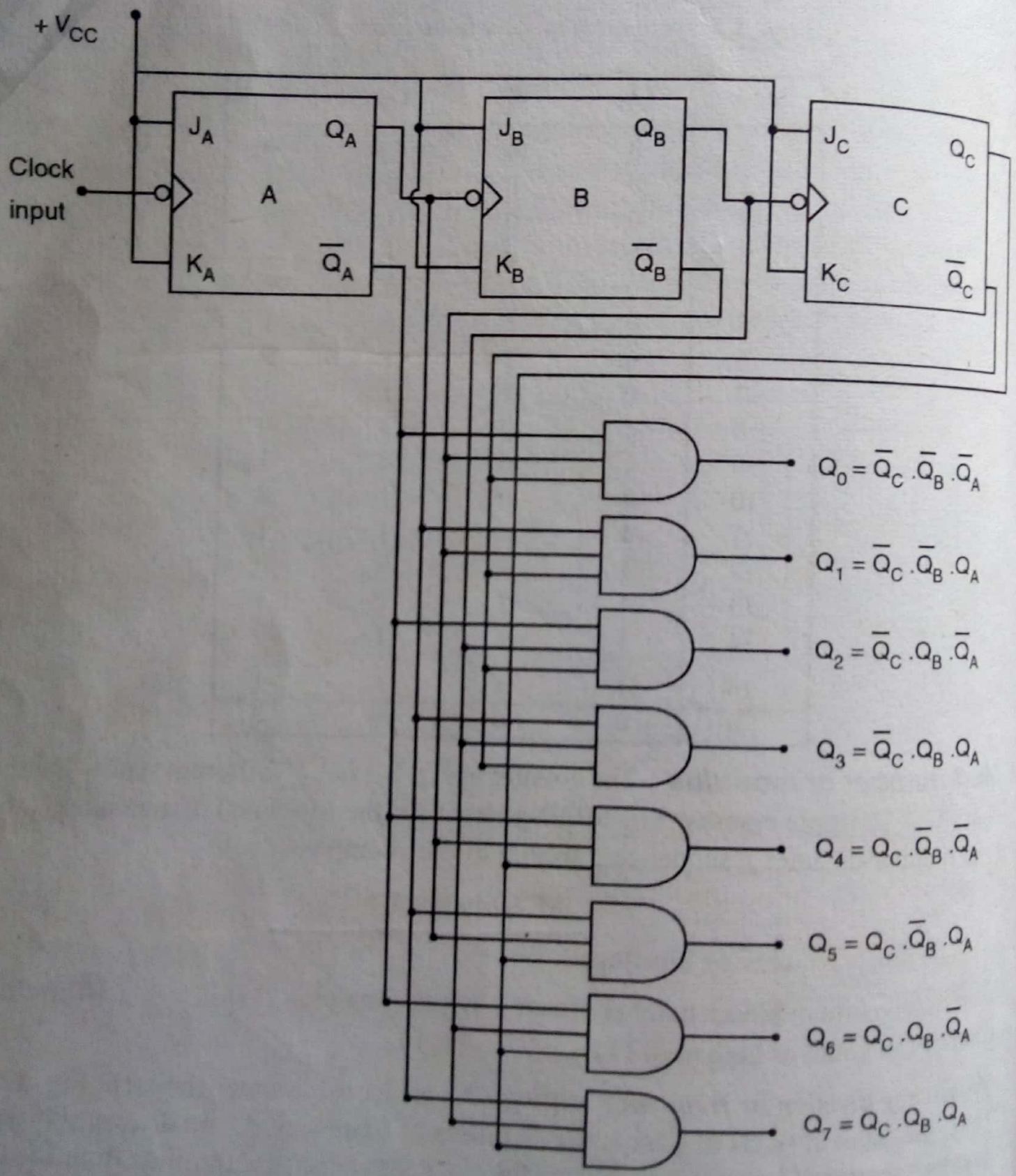


Fig. 8.2 *3-bit binary ripple counter with decoded outputs*

Scanned by CamScanner

8.11 SYNCHRONOUS DOWN COUNTER

A parallel down counter can be made to count down by using the inverted outputs of flip-flops to feed the various logic gates. The parallel counter shown in Fig. 8.11 can be converted to a down counter by connecting the $\overline{Q}_A$, $\overline{Q}_B$, and $(\overline{Q}_C)$ outputs to the AND gates in place of Q_A, Q_B and Q_C respectively as shown in Fig. 8.13. The counter will then proceed through the counting sequence (as shown in Table 8.7) when input pulses are applied.

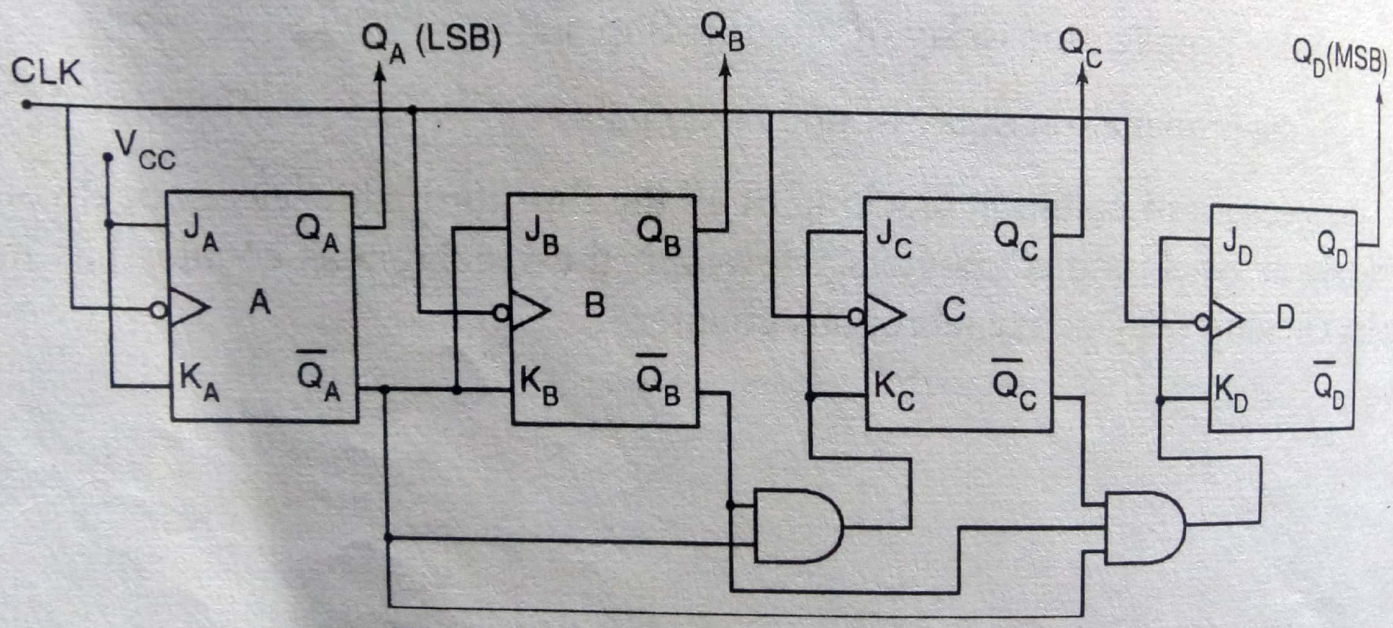


Fig. 8.13 *4-bit synchronous down counter*

Scanned by CamScanner

Table 8.7 Truth table of 4-bit synchronous down counter

State	Q_D	Q_C	Q_B	Q_A
15	1	1	1	1
14	1	1	1	0
13	1	1	0	1
12	1	1	0	0
11	1	0	1	1
10	1	0	1	0
9	1	0	0	1
8	1	0	0	0
7	0	1	1	1
6	0	1	1	0
5	0	1	0	1
4	0	1	0	0
3	0	0	1	1
2	0	0	1	0
1	0	0	0	1
0	0	0	0	0
15	1	1	1	1

8.12 SYNCHRONOUS UP / DOWN COUNTER

To form a parallel UP/DOWN counter, the control inputs (COUNT-UP and COUNT-DOWN) are used to allow either the normal output or the inverted output of one flip-flop to the J and K inputs of the following flip-flop. A MOD-8 UP/DOWN counter which will count from 000 to 111 when the COUNT-UP = 1 and COUNT-DOWN = 0, or from 111 to 000 when the COUNT-DOWN = 1 and COUNT-UP = 0, is shown in Fig. 8.14.

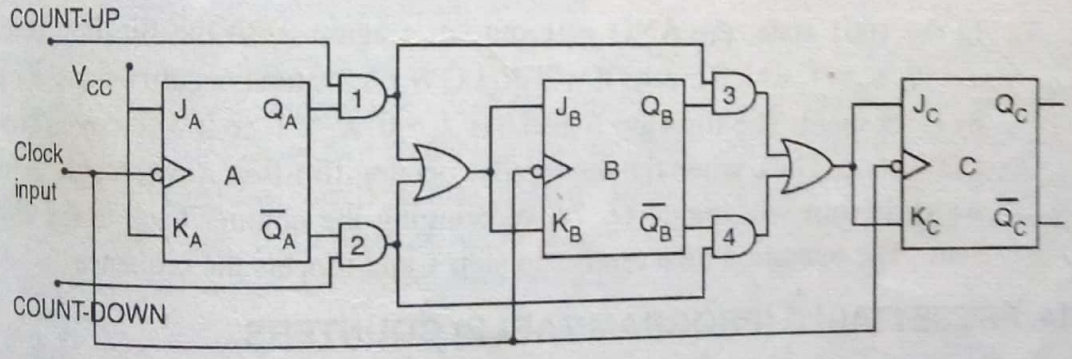


***Fig. 8.14** Parallel UP/DOWN counter (MOD-8)*

A logical 1 on the COUNT-UP line while COUNT-DOWN = 0 enables AND gates 1 and 3 and disables gates 2 and 4. This allows the Q_A and Q_B outputs through the AND gates to the J and K inputs of the following flip-flops, so that the counter counts up as pulses are applied. The reverse action takes place when COUNT-UP = 0 and COUNT-DOWN = 1.

8.13 SYNCHRONOUS/ASYNCHRONOUS COUNTER

The synchronous/asynchronous counter, constructed by combining the synchronous and asynchronous counters, represents a compromise between the speed of synchronous

Scanned by CamScanner

...plained as fol-

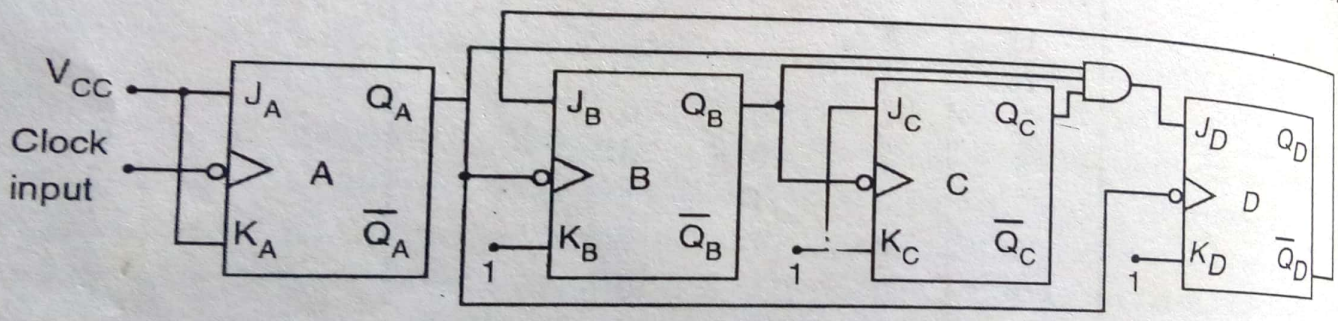


Fig. 8.15 *Synchronous / asynchronous decade counter*

Scanned by CamScanner

8.15.1 Design of MOD-3 Counter

MOD-3 counter is one with three states. To design a counter with three states, the number of flip-flops required can be found using the equation $2^n \geq N \geq 2^{n-1}$, where n is the number of flip-flops required and N is the number of states present in the counter. For $N = 3$, from the above equation, $n = 2$, i.e. two flip-flops are required. Assume that the MOD-3 counter has three states a, b and c and its sequence is given by $a \rightarrow b \rightarrow c \rightarrow a \ldots$

Step 1 State diagram Now, the state diagram for the MOD-3 counter can be drawn as shown in Fig. 8.17. Here, it is assumed that the state transition from one state to another takes place when the clock pulse is asserted; when the clock is unasserted, the counter remains in the present state.

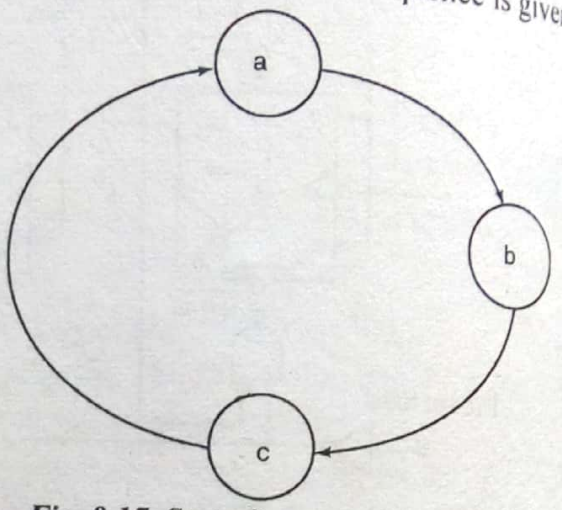


Fig. 8.17 *State diagram of MOD-3 counter*

Step 2 State table From the above state diagram, one can draw PS-NS table (Table 8.8).

Table 8.8 *PS-NS table for MOD-3 counter*

Present state (PS)	Next state (NS)
a	b
b	c
c	a

The state table given in Table 8.8 has no redundant state because no two states are equivalent. Hence, there is no modification required in the given state table.

Step 3 State assignment Let us assign two state variables to states a, b and c as follows: $a = 00$, $b = 01$ and $c = 10$. Then, the PS-NS table gets modified as shown in Table 8.9.

Table 8.9 *PS-NS table for MOD-3 counter*

Present state (PS)		Next state (NS)	
q_1	q_0	Q_1	Q_0
0	0	0	1
0	1	1	0
1	0	0	0
.....		.....	
1	1	d	d

Step 4 Excitation table Although any one of the four flip-flops, i.e. SR, JK, T and D, can be used , the selection of J-K flip-flop will result in a simplified circuit for synchronous counters. The excitation table having entries for flip-flop inputs

Scanned by CamScanner

(J_1K_1 and J_0K_0) can be drawn from the above PS-NS table (and using the application table of JK Flip-flop given in Table 7.11) as shown in Table 8.10.

Table 8.10 *Excitation table for MOD-3 counter*

PS		NS		Excitation inputs			
q_1	q_0	Q_1	Q_0	J_1	K_1	J_0	K_0
0	0	0	1	0	d	1	d
0	1	1	0	1	d	d	1
1	0	0	0	d	1	0	d
.....		.....		.....		.....	
1	1	d	d	d	d	d	d

In the first row of the above table, for the flip-flop 2 of the counter to change from present state ($q_1 = 0$) to next state ($Q_1 = 0$), the J_1K_1 inputs required are $0d$; for flip-flop 1 to change from $q_0 = 0$ to $Q_0 = 1$, the J_0K_0 inputs required are $1d$. Similarly, other entries are also made using the application table.

Step 5 Excitation maps The excitation maps for J_1, K_1, J_0 and K_0 inputs of the counter can be drawn as shown in Fig. 8.18 from the excitation table (Table 8.10).

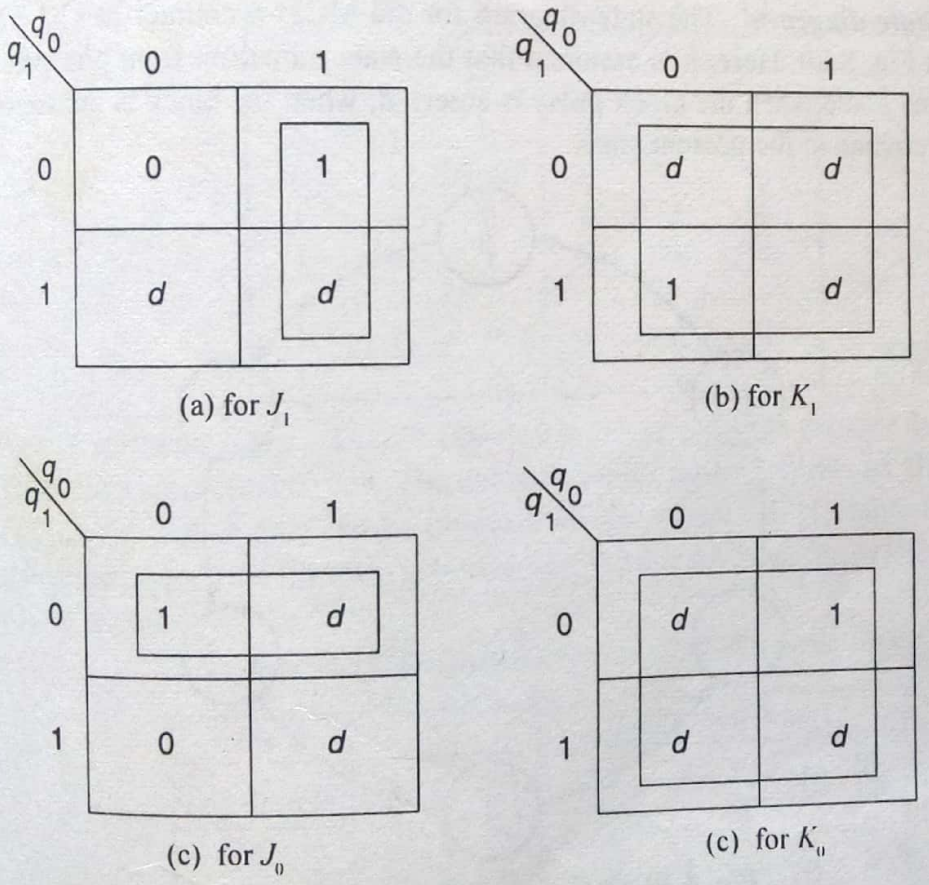


Fig. 8.18 *Excitation maps*

From the above excitation maps, the simplified excitation functions are :

$$J_1 = q_0, K_1 = 1; J_0 = \bar{q}_1 \text{ and } K_0 = 1$$

Step 6 Schematic diagram Using the above excitation equations, the circuit diagram for the MOD-3 counter can be drawn as shown in Fig. 8.19.

Scanned by CamScanner

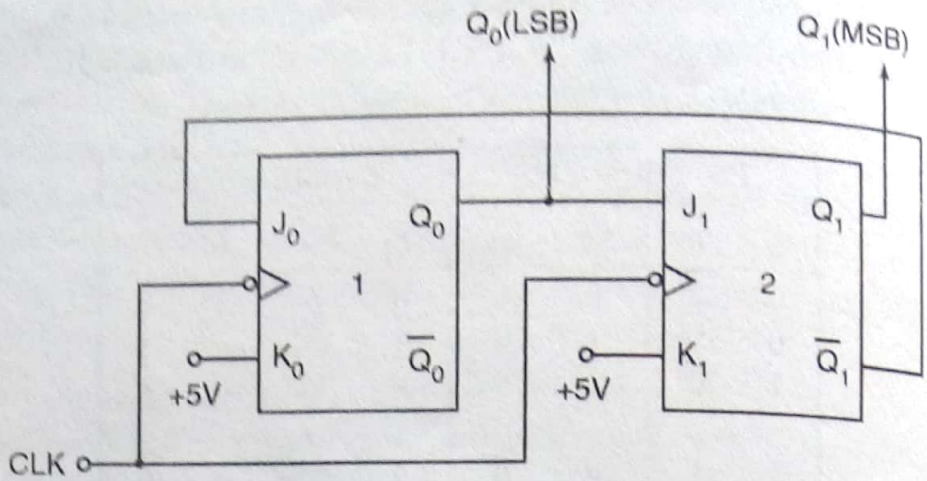


Fig. 8.19 Circuit diagram for MOD-3 synchronous counter

8.15.2 Design of MOD-6 Counter

In order to design a MOD-6 counter with six states, the number of flip-flops required is three. This is found from the equation $2^n \geq N \geq 2^{n-1}$, where $N = 6$, the number of states present in the MOD-6 counter. Let us assume that the MOD-6 counter has six states, viz. *a, b, c, d, e* and *f*.

Step 1 State diagram The state diagram for the MOD-6 counter can be drawn as shown in Fig. 8.20. Here, it is assumed that the state transition from one state to another takes place when the clock pulse is asserted; when the clock is unasserted, the counter remains in the present state.

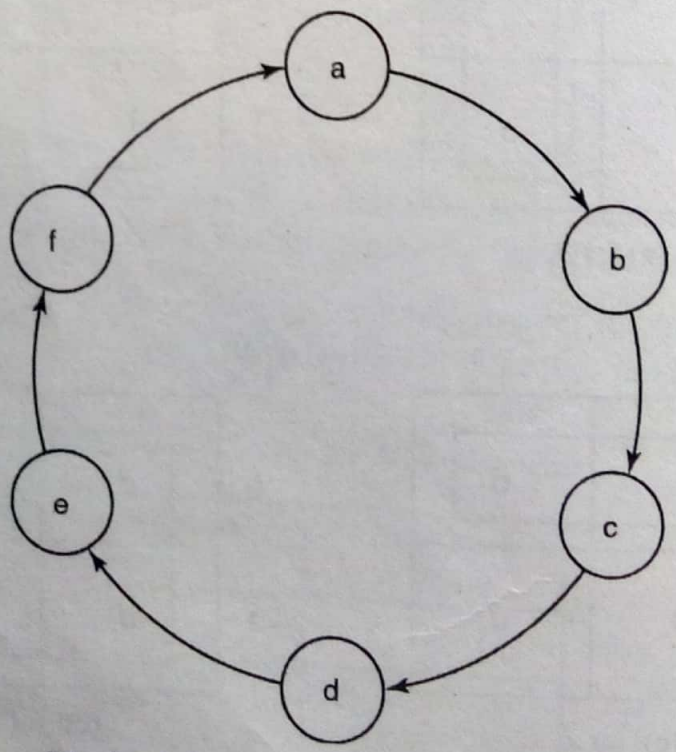


Fig. 8.20 State diagram of MOD-6 counter

Step 2 State table From the above state diagram, one can draw PS–NS table as shown in Table 8.11.

Scanned by CamScanner

Present state (PS)			Next state (NS)		
q_2	q_1	q_0	Q_2	Q_1	Q_0
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	0	0	0
.....			.....		
1	1	0	d	d	d
1	1	1	d	d	d

Scanned by CamScanner

Step 5 Excitation maps The excitation maps for J_2, K_2, J_1, K_1, J_0 and K_0 inputs of the counter can be drawn as shown in Fig. 8.21 from the excitation table (Table 8.13).

q_0 \ q_2q_1	00	01	11	10
0	0	0	d	d
1	0	1	d	d

(a) for J_2
$J_2 = q_1 q_0$

q_0 \ q_2q_1	00	01	11	10
0	d	d	d	0
1	d	d	d	1

(b) for K_2
$K_2 = q_0$

q_0 \ q_2q_1	00	01	11	10
0	0	d	d	0
1	1	d	d	0

(c) for J_1
$J_1 = \overline{q}_2 q_0$

q_0 \ q_2q_1	00	01	11	10
0	d	0	d	d
1	d	1	d	d

(d) for K_1
$K_1 = q_0$

q_0 \ q_2q_1	00	01	11	10
0	1	1	d	1
1	d	d	d	d

(e) for J_0
$J_0 = 1$

q_0 \ q_2q_1	00	01	11	10
0	d	d	d	d
1	1	1	d	1

(f) for K_0
$K_0 = 1$

Fig. 8.21 *Excitation maps for MOD-6 counter*

Step 6 Schematic diagram Using the above excitation equations, the circuit diagram for the MOD-6 counter can be drawn as shown in Fig. 8.22.

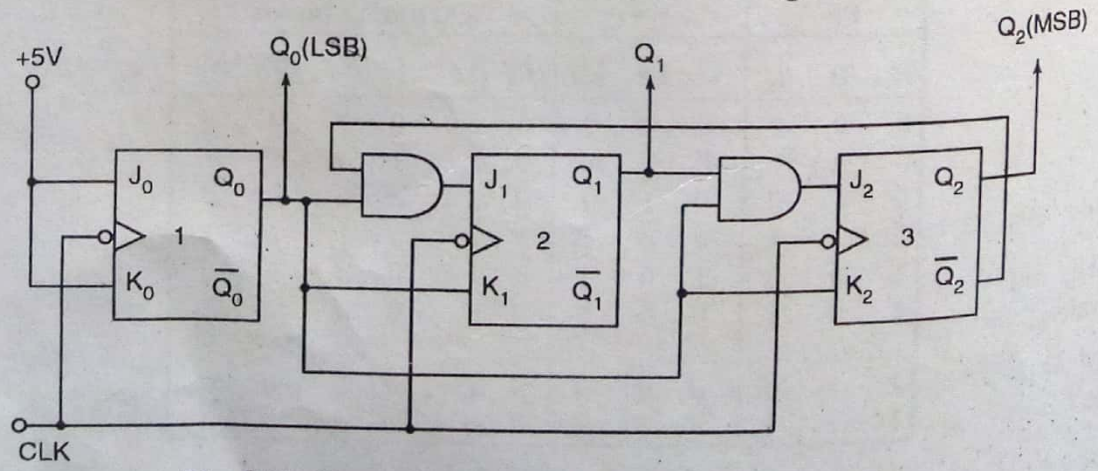


Fig. 8.22 *Circuit diagram for MOD-6 synchronous counter*

Scanned by CamScanner