DIGITALELECTRONICSCIRCUIT

3rdSEMESTER-ETC& CSE

CVRAMAN POLYTECHNIC

Prepared by
SUCHISMITA SATPATHY
DEPT. OF ETC
CVRP,BBSR

NUMBERSYSTEM AND CODES

INTRODUCTION:-

- Thetermdigitalreferstoaprocessthatisachievedbyusingdiscreteunit.
- In numbersystemthere are differentsymbols andeach symbolhasanabsolutevalueandalsohas place value.

RADIXORBASE:-

The radixor baseof anumber systemis definedas the number of differentdigits which canoccur in each position in the number system.

RADIXPOINT:-

Thegeneralizedformof adecimalpoint isknownas radix point. Inanypositionalnumber system the radix point divides the integer and fractional part.

N_r=[Integerpart.Fractionalpart]

↑

Radixpoint

NUMBERSYSTEM:-

Ingeneralanumberinasystemhavingbaseorradix 'r'canbewrittenas

$$a_n\ a_{n-1}a_{n-2}\ldots\ldots\ldots\ldots\ldots a_0.a_{-1}a_{-2}\ldots\ldots\ldots\ldots\ldots\ldots\ldots a_{-m}$$

Thiswillbeinterpretedas

$$Y= a_nxr^n+a_{n-1}xr^{n-1}+a_{n-2}xr^{n-2}+\ldots\ldots+a_0xr^0+a_{-1}xr^{-1}+a_{-2}xr^{-2}+\ldots\ldots\ldots\ldots\ldots\ldots+a_{-m}xr^{-m}$$

where Y=valueoftheentirenumber

a_n=the valueofthenthdigit r =

radix

TYPESOFNUMBERSYSTEM:-

Therearefourtypesofnumber systems.Theyare

1. Decimalnumbersystem
2. Binarynumbersystem
3. Octalnumbersystem
4. Hexadecimalnumbersystem

DECIMALNUMBERSYSTEM:-

- Thedecimalnumbersystemcontain tenunique symbols0,1,2,3,4,5,6,7,8and9. In
- decimal system 10 symbols are involved, so the base or radix is 10.
- Itisapositionalweighted system.
- Thevalueattachedtothesymboldependsonitslocation withrespecttothedecimalpoint.

Ingeneral,

$d_n \; d_{n-1}d_{n-2}\ldots\ldots\ldots\ldots d_0.d_{-1}d_{-2} \ldots\ldots\ldots\ldots d_{-m}$

isgiven by

$(d_n x\ 10^n)+(d_{n-1}x10^{n-1})+(d_{n-2}x10^{n-2})+\ldots + (d_0x10^0)+ (d_{-1}x10^{-1})+ (d_{-2}x10^{-2})+\ldots+(d_{-m}x10^{-m})$ **For example:-**

9256.26=9x1000+2x100+5x10+6x1+2x(1/10)+6x(1/100)

$=9x10^3+2x\ 10^2+5\ x10^1+6x10^0+2x10^{-1}+\ 6x10^{-2}$

BINARYNUMBERSYSTEM:-

- Thebinarynumbersystemisapositionalweightedsystem. The
- base or radix of this number system is 2.
- Ithastwoindependentsymbols.
- The symbols used are 0 and 1.
- A binary digit is called a bit.
- Thebinarypointseparatesthe integerandfractionparts.

Ingeneral,

$d_n \; d_{n-1}d_{n-2}\ldots\ldots\ldots\ldots d_0.d_{-1}d_{-2} \ldots\ldots\ldots\ldots d_{-k}$

isgiven by

$(d_nx2^n)+(d_{n-1}x2^{n-1})+(d_{n-2}x2^{n-2})+\ldots.+ (d_0x2^0)+(d_{-1}x2^{-1})+ (d_{-2}x2^{-2})+\ldots.+(d_{-k}x2^{-k})$

OCTALNUMBERSYSTEM:-

- Itisalsoapositionalweightedsystem. Its
- base or radix is 8.
- Ithas8 independentsymbols0,1,2,3,4,5,6and7.
- Itsbase8=2^3 ,every3-bitgroupofbinary canberepresentedbyanoctaldigit.

HEXADECIMALNUMBERSYSTEM:-

- Thehexadecimalnumbersystemisapositionalweightedsystem. The
- base or radix of this number system is 16.
- Thesymbolsusedare0,1,2,3,4,5,6,7,8,9,A,B,C,D,Eand F
- Thebase16 =24 ,every4- bit groupofbinarycanberepresentedbyanhexadecimaldigit.

CONVERSIONFROMONENUMBERSYSTEMTOANOTHER:-

1. **BINARYNUMBERSYSTEM:-**

(a) Binarytodecimalconversion:-

In this method,eachbinarydigitof the numberismultipliedbyitspositionalweightandthe productterms are added to obtain decimal number.

Forexample:

(i) Convert$(10101)_2$todecimal.

Solution :

(Positionalweight) $2^4 2^3 2^2 2^1 2^0$

Binarynumber 10101

$=(1x2^4)+(0\ x2^3)+(1x2^2)+(0\ x2^1)+(1\ x2^0)$

$=16+0+4+0+1$

$=(21)_{10}$

(ii) Convert$(111.101)_2$todecimal.

Solution:

$(111.101)_2$ $=(1x2^2)+(1\ x2^1)+(1x2^0)+(1x2^{-1})+(0x2^{-2})+(1x2^{-3})$

$=4+2+1+0.5+0+0.125$

$=(7.625)_{10}$

(b) BinarytoOctalconversion:-

For conversionbinary tooctal thebinarynumbersaredivided intogroupsof 3bitseach,startingatthe binary point and proceeding towards left and right.

Octal	Binary	Octal	Binary
0	000	4	100
1	001	5	101
2	010	6	110
3	011	7	111

Forexample:

(i) Convert$(101111010110.110110011)_2$intooctal.

Solution :

Groupof3bitsare	101	111	010	110.		110	110	011
Converteachgroupintooctal=	5	7	2	6	.	6	6	3

Theresultis**$(5726.663)_8$**

(ii) Convert$(10101111001.0111)_2$intooctal.

Solution :

Binarynumber	10	101	111	001.		011	1
Groupof3bitsare	=010	101	111	001.		011	100
Converteachgroupintooctal=	2	5	7	1	.	3	4

Theresultis**$(2571.34)_8$**

(c) BinarytoHexadecimalconversion:-

For conversionbinary tohexadecimal number thebinarynumbersstartingfrom thebinarypoint,groupsare made of 4 bits each, on either side of the binary point.

Hexadecimal	Binary	Hexadecimal	Binary
0	0000	8	1000
1	0001	9	1001
2	0010	A	1010
3	0011	B	1011
4	0100	C	1100
5	0101	D	1101
6	0110	E	1110
7	0111	F	1111

Forexample:
(i) Convert$(1011011011)_2$intohexadecimal.

Solution:

GivenBinary number 10 1101 1011

Groupof4bitsare 0010 1101 1011

Converteachgroupinto hex = 2 D B

Theresultis$(\mathbf{2DB})_{16}$

(ii) Convert$(01011111011.011111)_2$intohexadecimal.

Solution:

GivenBinary number 010 1111 1011 . 0111 11

Groupof3bitsare =0010 1111 1011 . 0111 1100

Converteachgroupintooctal= 2 F B . 7 C

Theresultis$(\mathbf{2FB.7C})_{16}$

2. DECIMALNUMBER SYSTEM:-

(a) Decimaltobinaryconversion:-

In theconversionthe integernumberare converted to thedesiredbaseusing successivedivisionbythe base or radix.

Forexample:
(i) Convert$(52)_{10}$intobinary.

Solution:

Divide the given decimal number successively by 2 read the integer part remainder upwards to get equivalentbinary number.Multiply thefractionpart by2. Keep theinteger in theproduct asit is andmultiply the new fraction in the product by 2. The process is continued and the integer are read in the products from top to bottom.

2|52
2|26 —0
2|13 —0
2|6 —1
2|3 —0
2|1 —1
0 —1

Resultof$(52)_{10}$is **$(110100)_2$**

(ii) Convert$(105.15)_{10}$intobinary.

Solution:

Integerpart		**Fraction part**
2|105		0.15x2=0.30
2|52	¯1	0.30x2=0.60
2|26	¯0	0.60x2=1.20
2|13	¯0	0.20x2=0.40
2|6	¯1	0.40x2=0.80
2|3	¯0	0.80x2=1.60
2|1	¯1	
0	¯1	

Resultof$(105.15)_{10}$is**$(1101001.001001)_2$**

(b) Decimaltooctalconversion:-

To convert the given decimal integer number to octal, successively divide the given number by 8 till the quotient is 0. To convert the given decimal fractions to octal successively multiply the decimal fraction and the subsequent decimal fractions by 8 till the product is 0 or till the required accuracy is obtained.

Forexample:

(i) Convert$(378.93)_{10}$intooctal.

Solution:

8|378		0.93x8=7.44
8|47	— 2	0.44x8=3.52
8|5	— 7	0.52x8=4.16
0	— 5	0.16x8=1.28

Resultof$(378.93)_{10}$is**$(572.7341)_8$**

(c) Decimaltohexadecimalconversion:-

Thedecimaltohexadecimalconversionissameasoctal.

Forexample:

(i) Convert$(2598.675)_{10}$intohexadecimal.

Solution:

	Remainder			**Hex**
	Decimal	**Hex**		
16|2598			0.675x16=10.8	A
16|162	—6	6	0.800x16=12.8	C
16|10	—2	2	0.800x16=12.8	C
0	— 10	A	0.800x16=12.8	C

Resultof$(2598.675)_{10}$is**$(A26.ACCC)_{16}$**

3. OCTAL NUMBER SYSTEM:-

(a) Octaltobinaryconversion:-

Toconvertagivenaoctalnumbertobinary,replaceeachoctaldigitby its3-bitbinaryequivalent.

Forexample:

Convert$(367.52)_8$intobinary.

Solution:

GivenOctalnumberis	3	6	7	.	5	2
Converteachgroupoctal to binary	=011	110111.101010				

Resultof$(367.52)_8$is**$(011110111.101010)_2$**

(b) Octaltodecimal conversion:-

Forconversionoctaltodecimalnumber, multiplyeachdigitin theoctalnumberby theweightof its position and add all the product terms

Forexample: -

Convert$(4057.06)_8$todecimal

Solution:

$(4057.06)_8 = 4x8^3+0x8^2+5x8^1+7x8^0+0x8^{-1}+6x8^{-2}$

$= 2048 +0 +40 +7 +0+0.0937$

$= (2095.0937)_{10}$

Resultis**$(2095.0937)_{10}$**

(c) Octaltohexadecimalconversion:-

Forconversionof octaltoHexadecimal,first convert thegivenoctalnumbertobinary andthen binary number to hexadecimal.

Forexample:-

Convert$(756.603)_8$tohexadecimal.

Solution:-

Givenoctal no.		7	5	6	.	6	0	3
Convert eachoctaldigittobinary	=	111	101	110	.	110	000	011
Group of 4bits are	=	0001	1110	1110	.	1100	0001	1000
Convert4bitsgrouptohex.	=	1	E	E	.	C	1	8

Resultis(**$1EE.C18)_{16}$**

(4)HEXADECIMALNUMBERSYSTEM:-

(a) Hexadecimaltobinaryconversion:-

Forconversionofhexadecimaltobinary,replacehexadecimaldigitbyits4bitbinarygroup.

Forexample:

Convert$(3A9E.B0D)_{16}$intobinary.

Solution:

GivenHexadecimalnumberis	3	A	9	E	.	B	0	D
Convert each hexadecimal 4 bitbinary	= 0011 101010011110.101100001101 digit to							

Resultof$(3A9E.B0D)_8$is**$(0011101010011110.101100001101)_2$**

(b) Hexadecimaltodecimalconversion:-

Forconversionof hexadecimaltodecimal, multiplyeachdigit in thehexadecimalnumberbyitsposition weight and add all those product terms.

Forexample: -
Convert$(A0F9.0EB)_{16}$todecimal

Solution:

$(A0F9.0EB)_{16} = (10x16^3)+(0x\ 16^2)+(15x16^1)+(9x\ 16^0)+(0x16^{-1})+(14x\ 16^{-2})+(11\ x16^{-3})$

$= 40960+0+240+9+0+0.0546+0.0026$

$= (41209.0572)_{10}$

Resultis**$(41209.0572)_{10}$**

(c) HexadecimaltoOctalconversion:-

Forconversionof hexadecimaltooctal,firstconvert the givenhexadecimalnumbertobinary andthen binary number to octal.

Forexample:-
Convert$(B9F.AE)_{16}$tooctal.

Solution:-

Givenhexadecimalno.is		B	9	F	.	A	E		
Converteachhex.digittobinary	=	1011	1001	1111	.	1010	1110		
Groupof3bitsare	=	101110		011111	.	101011100			
Convert3bitsgrouptooctal.	=	5	6	3	7	.	5	3	4

Result is **$(5637.534)_8$**

BINARYARITHEMATICOPERATION:-

1. BINARY ADDITION:-

Thebinaryaddition rulesareasfollows
0+0=0;0+1=1 ;1+0=1;1+1=10,i.e0witha carryof1

Forexample:-

Add$(100101)_2$and$(1101111)_2$.
Solution:-

```
   10 0 1 01
+  1 10 1 1 11
  10 0 1 01 00
```

Resultis**$(10010100)_2$**

2. BINARYSUBTRACTION:-

Thebinarysubtractionrulesareasfollows
0-0=0; 1-1=0;1-0=1; 0-1=1, witha borrowof1

Forexample:-
Substract$(111.111)_2$from$(1010.01)_2$.
Solution :-

```
    10 1 0.0 1 0
-   1 1 1. 1 1 1
    00 10.0 11
```

Resultis**$(0010.011)_2$**

3. BINARYMULTIPLICATION:-

Thebinarymultiplicationrulesareasfollows 0 x 0 = 0 ; 1 x 1 = 1 ; 1 x 0 = 0 ; 0 x 1 = 0

Forexample:-

Multiply$(1101)_2$by$(110)_2$.
Solution :-

```
      11 0 1
x     110
      00 0 0
     11 0 1
+  1 1 0 1
   10 0 1 11 0
```

Resultis**$(1001110)_2$**

4. BINARY DIVISION:-

Thebinarydivisionisverysimpleandsimilartodecimalnumbersystem.Thedivisionby‘0’ismeaningless. Sowehaveonly2rules

0÷1 =0

1÷1 =1

Forexample:-
Divide$(10110)_2$by$(110)_2$. Solution

:-

```
110 ) 101101(111.1
  -  110
      1010
      110
       1001
      110
        110
      110
      000
```

Resultis**$(111.1)_2$**

1'sCOMPLEMENTREPRESENTATION:-

The1'scomplementof abinarynumberisobtainedbychangingeach0to1andeach1to0.

Forexample:-

Find$(1100)_2$1'scomplement.

Solution:-

Given	1	1	0	0
1'scomplementis	0	0	1	1

Resultis**$(0011)_2$**

2'sCOMPLEMENTREPRESENTATION:-

The2'scomplementof abinarynumberisabinarynumberwhich isobtained byadding1tothe 1's complementofanumberi.e.

2'scomplement=1'scomplement +1

Forexample:-

Find$(1010)_2$2'scomplement.

Solution:-

Given		1	0	1	0
1'scomplementis		0	1	0	1
	+				1
2'scomplement		0	1	1	0

Resultis**$(0110)_2$**

SIGNEDNUMBER:-

In sign –magnitude form,additionalbit calledthesignbit isplaced infrontof thenumber. If the signbit is 0, the number is positive. If it is a 1, the number is negative.

Forexample:-

0 1 0 1 0 0 1= +41
↑
Signbit

1 1 0 1 0 0 1= -41
↑
Signbit

SUBSTRACTIONUSINGCOMPLEMENTMETHOD:-

1's COMPLEMENT:-

In 1's complement subtraction, add the 1's complement of subtrahend to the minuend. If there is a carry out, then the carry is added to the LSB. This is called end around carry. If the MSB is 0, the result is positive. If the MSB is 1, the result is negative and is in its 1's complement form. Then take its 1's complement to get the magnitude in binary.

Forexample:-

Subtract$(10000)_2$from$(11010)_2$using1'scomplement.

Solution:-

11 0 1 0 11 0 1 0 = 26

-10 0 0 0 => +0 1 11 1 (1's complement) =- 16

Carry → **1**0 1 0 01 + 10

\+ 1

01 0 1 0 =+10

Resultis**+10**

2's COMPLEMENT:-

In 2's complement subtraction, add the 2's complement of subtrahend to the minuend. If there is a carry out, ignoreit. Ifthe MSBis 0, theresult is positive.Ifthe MSBis 1, theresult is negative andis inits 2's complement form. Then take its 2's complement to get the magnitude in binary.

Forexample:-

Subtract$(1010100)_2$from$(1010100)_2$using2'scomplement.

Solution:-

10 1 0 10 0 10 1 0 10 0 = 84

\- 10 1 0 10 0 => +0 101 10 0(2'scomplement) = -84

10000000(Ignorethecarry) 0

= 0(result=0)

Hence MSB is0. Theanswerispositive.So itis+0000000 =**0**

DIGITALCODES:-

In practice the digital electronics requires to handle data which may be numeric, alphabets and special characters. This requires the conversion of the incoming data into binary format before it can be processed. There is various possible ways of doing this and this process is called encoding. To achieve the reverse of it,we use decoders.

WEIGHTEDANDNON-WEIGHTEDCODES:-

Therearetwotypesofbinary codes

1) Weightedbinary codes
2) Non-weightedbinary codes

Inweighted codes,foreach position (orbit),thereisspecificweightattached.

Forexample,inbinarynumber,each bitisassignedparticularweight2n where'n'isthebitnumberforn= 0,1,2,3,4theweightsare1,2,4,8,16 respectively.

Example :- BCD

Non-weightedcodesarecodeswhicharenotassigned withany weighttoeachdigitposition, i.e.,each digit position within the number is not assigned fixed value.

Example:-Excess-3(XS-3)codeandGraycodes

BINARYCODEDDECIMAL(BCD):-

BCD isa weighted code. In weighted codes, each successive digitfrom right to left represents weightsequalto some specified value and to get the equivalent decimal number add the products of the weights by the corresponding binary digit. 8421 is the most common because 8421 BCD is the most natural amongst theother possible codes.

Forexample:-

$(567)_{10}$isencodedinvarious4bitcodes.

Solution:-

Decimal	→	5	6	7
8421code	→	0101	0110	0111
6311code	→	0111	1000	1001
5421code	→	1000	0100	1010

BCDADDITION:-

Additionof BCD (8421) is performedbyaddingtwo digits of binary,startingfrom leastsignificant digit. In case if the result is an illegal code (greater than 9) or if there is a carry out of one then add 0110(6)and add the resulting carry to the next most significant.

Forexample:-

Add679.6from536.8usingBCDaddition.

Solution:-

67 9.6	011001111001.0110	(679.6inBCD)
+5 3 6.8	=>+01010011 0110.1000	(536.8 in BCD)
12 1 6.4	10111010 1111.1110	(Allareillegalcodes)
	+ 0110+0110+0110.+0110	(Add0110toeach)
	0001 0010 00010110.0100	
	1 2 1 6 . 4	(correctedsum =1216.4)

Resultis**1216.4**

BCDSUBTRACTION:-

The BCD subtraction is performed by subtracting the digits of each 4 –bit group of the subtrahend from corresponding 4 –bit group of the minuend in the binary starting from the LSD. If there is no borrow from the next higher group[then no correction is required. If there is a borrow from the next group, then 6_{10}(0110) is subtracted from the difference term of this group.

Forexample:-

Subtract147.8from206.7using8421BCDcode.

Solution:-

206.7	00100000 0110.0111	(206.7in BCD)
-147.8	=>-00010100 0111.1000	(147.8inBCD)
58.9	000010111110.1111	(Borrowsarepresent)
	-0110-0110 .-0110	
	01011000.1001	
	5 8 . 9	(correcteddifference=58.9)

Resultis**$(58.9)_{10}$**

EXCESSTHREE(XS-3)CODE:-

The Excess-3 code, also called XS-3, is a non- weighted BCD code. This derives it name from the fact thateach binary code word is the corresponding 8421 code word plus 0011(3). It is a sequential code. It is a self complementing code.

XS-3ADDITION:-

In XS-3 addition, add the XS-3 numbers by adding the 4 bit groups in each column starting from the LSD. If there is no carry out from the addition of any of the 4 bit groups, subtract 0011 from the sum term of those groups. If there is a carry out, add 0011 to the sum term of those groups

Forexample:-

Add37and28usingXS-3code.

Solution:-

37 01101010 (37inXS-3)

+28 => +01011011 (28inXS-3)

65 10111010 (Carry is generated)

\+ 1 (Propagatecarry)

1100 0101 (Add0110tocorrect0101and

-0011 +0011 subtract0011tocorrect1100)

1001 1000 (Correctedsum inXS-3=65_{10})

XS-3SUBTRACTION:-

To subtract in XS-3 number by subtracting each 4-bit group of the subtrahend from the corresponding 4-bit group of the minuend starting from the LSD. If there is no borrow from the next 4-bit group. add 0011 to the difference term of such groups. If there is a borrow, subtract 0011 from the difference term.

Forexample:-

. **Subtract175from267usingXS-3code**.

Solution :-`

267 01011010 1010 (267inXS-3)

-175 => -01001010 1000(175inXS-3)

092 000011110010 (Correct0010and0000byadding0011and

+0011-0011+0011 correct1111bysubtracting0011)

0011 11000101 (Correcteddifference inXS-3=92_{10})

ASCIICODE:-

The American Standard Code for Information Interchange (ASCII) pronounced as 'ASKEE' is widely used alphanumeric code. This is basically a 7 bit code. The number ofdifferent bit patterns thatcan be created with7 bits is 27 = 128 , the ASCII can be used to encode both the uppercase and lowercase characters of the alphabet(52symbols) andsomespecial symbols inadditiontothe10 decimal digits.Itisusedextensively for printers and terminals that interface with small computer systems. The table shown below shows the ASCII groups.

TheASCIIcode

LSBs	**MSBs**							
	000	001	010	011	100	101	110	111
0000	NUL	DEL	Space	0	@	P	P	
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	"	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	$	4	D	T	d	t

0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	‘	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	,	<	L	\	l	|
1101	CR	GS	-	=	M	]	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	_	o	DLE

EBCDICCODE:-

The Extended Binary Coded Decimal Interchange Code (EBCDIC) pronounced as ‘eb –si- dik’ is an 8 bit alphanumeric code.Since28= 256bitpatterns canbe formedwith8 bits.Itis used by mostlargecomputers to communicate in alphanumeric data. The table shown below shows the EBCDIC code.

TheEBCDICcode

LSD (Hex)	**MSD(Hex)**															
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	DLE	DS		SP	&							[	]	\	0
1	SOH	DC1	SOS				/		a	j	~		A	J		1
2	STX	DC2	FS	SYN					b	k	s		B	K	S	2
3	ETX	DC3							c	l	t		C	L	T	3
4	PF	RES	BYP	PN					d	m	u		D	M	U	4
5	HT	NL	LF	RS					e	n	v		E	N	V	5
6	LC	BS	EOB	YC					f	o	w		F	O	W	6
7	DEL	IL	PRE	EOT					g	p	x		G	P	X	7
8		CAN							h	q	y		H	Q	Y	8
9		EM							i	r	z		I	R	Z	9
A	SMM	CC	SM		Ø	!	I	:								
B	VT				.	$	,	#								
C	FF	IFS		DC4	<	*	%	@								
D	CR	IGS	ENQ	NAK	(	)	_	‘								
E	SO	IRS	ACK		+	;	>	=								
F	SI	IUS	BEL	SUB	I	‘	?	‘								

GRAYCODE:-

Thegraycode isanon-weighted code.It is notaBCDcode.It is cyclic code because successive words inthis differ in one bit position only i.e it is a unit distance code.

Gray code is used in instrumentation and data acquisition systems where linear orangulardisplacement is measured. They are also used in shaft encoders, I/O devices, A/D converters and other peripheral equipment.

BINARY-TO–GRAY CONVERSION:-

Ifann-bitbinarynumber isrepresentedbyB_nB_{n-1}-- -- -B_1anditsgraycodeequivalentbyG_nG_{n-1}----------G_1, where$B_n$and$G_n$are theMSBs ,thengraycodebits areobtainedfrom thebinary codeasfollows $G_n = B_n$

$G_{n-1}=B_n \oplus B_{n-1}$

.
.
.
.

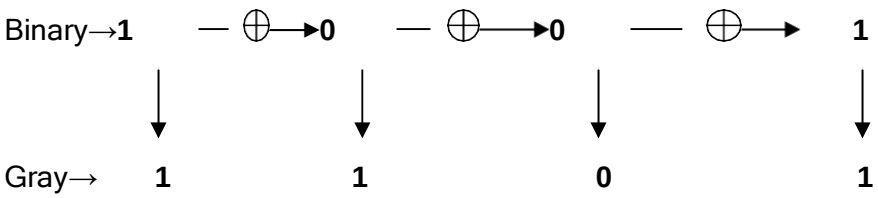
$G_1=B_2 \oplus B_1$

Wherethesymbol$\oplus$standsforExclusiveOR(X-OR)

Forexample:-

Convertthe binary1001totheGraycode.

Solution:-`



Thegraycodeis**1101**

GRAY-TO-BINARY CONVERSION:-

Ifann-bitgraynumberisrepresentedby G_nG_{n-1}------- G_1anditsbinary equivalentbyB_nB_{n-1}----------B_1, thenbinarybitsareobtainedfromGraybitsasfollows : B_n

$= G_n$

$B_{n-1}=B_n \oplus G_{n-1}$

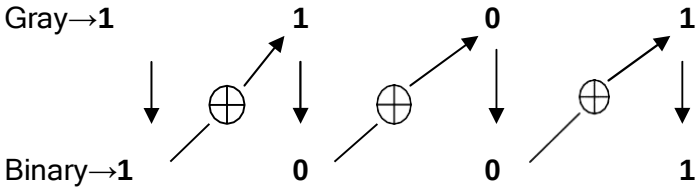
.
.
.
.

$B_1=B_2 \oplus G_1$

Forexample:-

ConverttheGraycode1101tothebinary.

Solution:-



Thebinary codeis**1001**

LOGICGATES

LOGICGATES:-

- Logicgatesarethefundamentalbuildingblocks ofdigitalsystems. There
- are 3 basic types of gates AND, OR and NOT.
- Logic gates are electronic circuits becausetheyare madeupof anumberof electronicdevicesand components.
- Inputsandoutputsof logicgatescan occuronly in2 levels. Thesetwo levelsaretermed HIGH and LOW, or TRUE and FALSE, or ON and OFF or simply 1 and0.
- Thetable which listsall thepossible combinationsof input variablesand thecorrespondingoutputsis called a truth table.

LEVEL LOGIC:-

Alogicin whichthe voltagelevelsrepresentslogic1and logic0.Levellogicmaybepositiveornegativelogic.

PositiveLogic:-

A positivelogicsystemis theone in which thehigherof thetwovoltage levels represents thelogic1and the lower of the two voltages level represents the logic 0.

NegativeLogic:-

A negativelogicsystemistheone in which thelowerof the two voltage levelsrepresentsthe logic1 andthe higher of the two voltages level represents the logic 0.

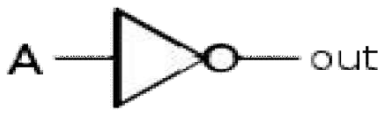
DIFFERENTTYPESOFLOGICGATES:-

NOT GATE (INVERTER):-

- A NOTgate,also calledand inverter, hasonlyone inputandoneoutput. It is
- a device whose output is always the complement of its input.
- Theoutputof a NOTgateis the logic1 state whenits inputis in logic0 stateand the logic0 state when its inputs is in logic 1 state.

ICNo.:-7404

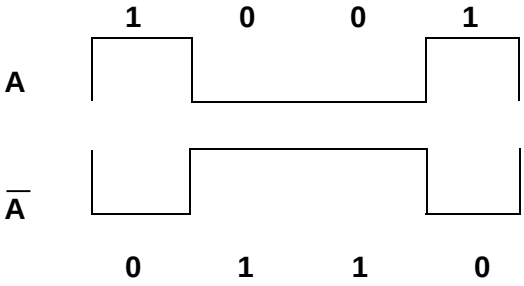
LogicSymbol



Truthtable

INPUT A	OUTPUT A
0	**1**
1	0

TimingDiagram

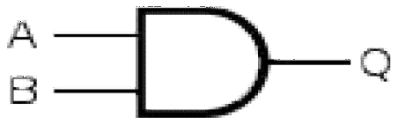


ANDGATE:-

- AnAND gatehastwoor moreinputsbutonlyoneoutput.
- Theoutput islogic1 stateonlywhen each one ofitsinputsisat logic1state. The
- output is logic 0 state even if one of its inputs is at logic 0 state.

ICNo.:-7408

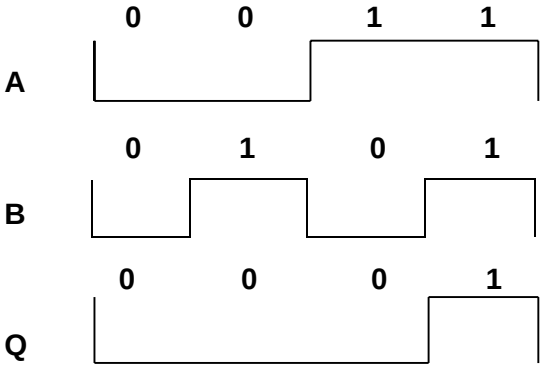
LogicSymbol



TruthTable

		OUTPUT
A	B	Q=A.B
0	0	0
0	1	0
1	0	0
1	1	1

TimingDiagram



ORGATE:-

- AnOR gatemayhavetwoor moreinputsbutonlyoneoutput.
- Theoutput islogic1 state,evenifoneofitsinputisin logic1state.
- Theoutputislogic0state,onlywheneachoneofitsinputsisinlogicstate.

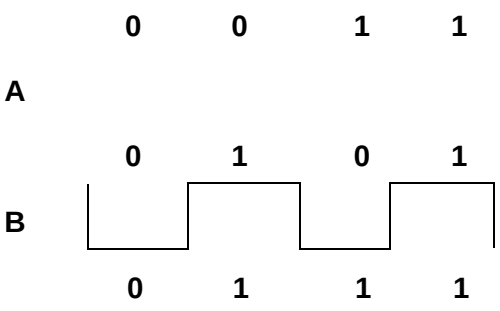
ICNo.:-7432

LogicSymbol

TruthTable

INPUT		OUTPUT
A	B	Q=A+B
0	0	0
0	1	1
1	0	1
1	1	1

TimingDiagram



NANDGATE:-

- NANDgateisacombinationofanANDgateandaNOTgate.
- Theoutput is logic0 when eachof the input is logic 1andfor anyothercombinationof inputs,the output is logic 1.

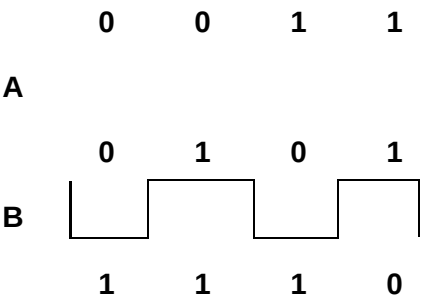
IC No.:- 7400 two input NAND gate
7410threeinputNANDgate
7420 four input NAND gate
7430eightinputNANDgate

LogicSymbol

TruthTable

INPUT		OUTPUT
A	B	$Q=\overline{A.B}$
0	0	1
0	1	1
1	0	1
1	1	0

TimingDiagram



NORGATE:-

- NORgateis acombinationofanORgateandaNOTgate.
- Theoutput is logic1,onlywheneachoneof itsinputis logic 0andforanyother combinationof inputs, the output is a logic 0 level.

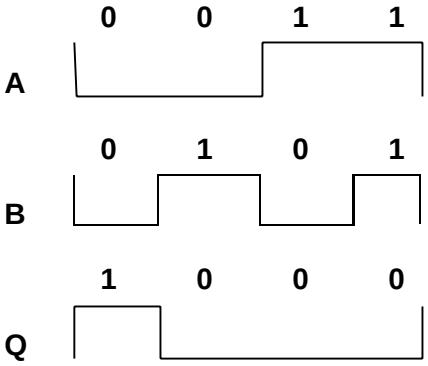
IC No.:- 7402 two input NOR gate
7427threeinputNORgate
7425 four input NOR gate

LogicSymbol

TruthTable

INPUT		OUTPUT
A	B	$Q=\overline{A+B}$
0	0	1
0	1	0
1	0	0
1	1	0

TimingDiagram



EXCLUSIVE–OR(X-OR)GATE:-

- AnX-ORgateisatwoinput,oneoutputlogiccircuit.
- Theoutput is logic1 when oneandonlyoneof itstwo inputs is logic1.Whenboth the inputs islogic0 or when both the inputs is logic 1, the output is logic 0.

ICNo.:-7486

LogicSymbol

INPUTSare**A**and**B**

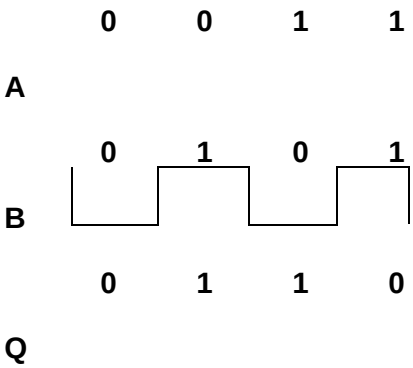
OUTPUTis**Q**=A⊕B

=A $\bar{B}$+A $\bar{B}$

TruthTable

INPUT		OUTPUT
A	B	Q=A⊕B
0	0	0
0	1	1
1	0	1
1	1	0

TimingDiagram

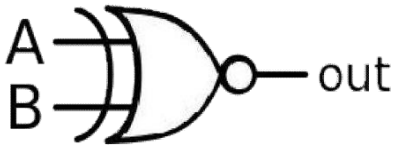


EXCLUSIVE–NOR(X-NOR)GATE:-

- AnX-NORgate isthecombination of anX-ORgate anda NOTgate. An
- X-NOR gate is a two input, one output logic circuit.
- Theoutput is logic1only whenboth theinputsarelogic0or whenboththe inputs is1. The
- output is logic 0 when one of the inputs is logic 0 and other is1.

IC No.:-74266

LogicSymbol

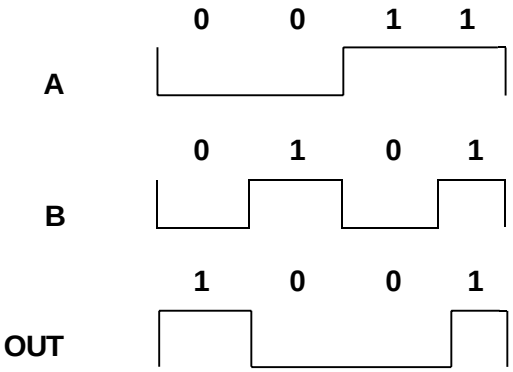


INPUT		OUTPUT
A	B	OUT=A XNORB
0	0	1
0	1	0
1	0	0
1	1	1

OUT=AB+$\overline{A}\overline{B}$

=AXNORB

TimingDiagram

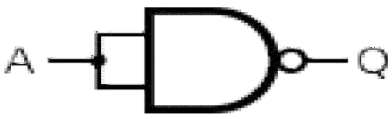


UNIVERSALGATES:-

There are 3 basic gates AND, OR and NOT, there are two universal gates NAND and NOR, each of which can realize logic circuits single handedly. The NAND and NOR gates are called universal building blocks. Both NAND and NOR gates can perform all logic functions i.e. AND, OR, NOT, EXOR and EXNOR.

NANDGATE:-

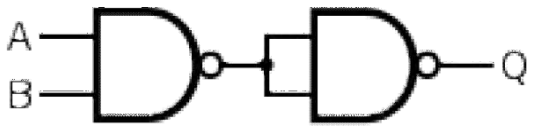
a) **InverterfromNANDgate**



Input **=A**
Output **Q=$\overline{A}$**

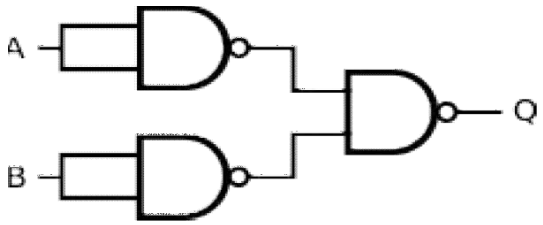
b) **ANDgatefromNANDgate**

Inputsare**A**and**B**
Output **Q=A.B**



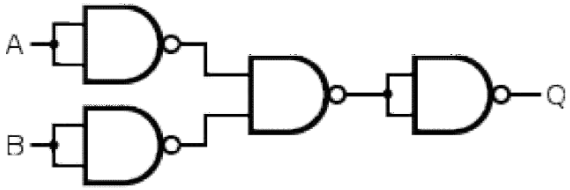
c) **ORgatefromNANDgate**

Inputsare**A**and**B**
Output **Q=A+B**



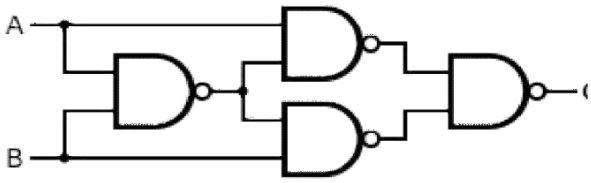
d) NORgatefromNANDgate

Inputsare**A**and**B**
Output **Q=A+B**



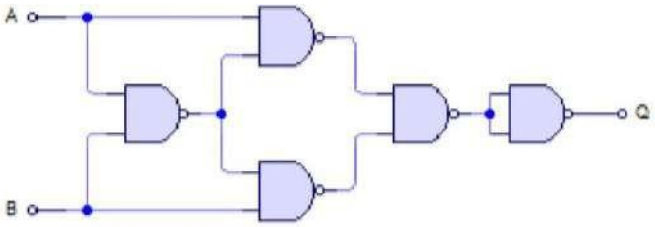
e) EX-ORgatefromNANDgate

Inputsare**A**and**B**
Output **Q=AB +AB**



f) EX-NORgateFromNANDgate

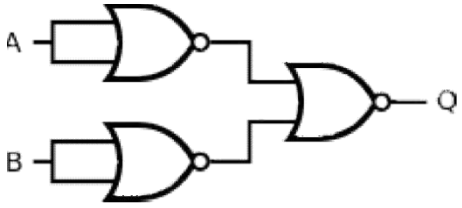
Inputs are **A** and **B**_
Output **Q=AB +A B**



NORGATE:-

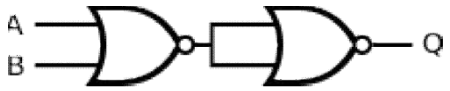
a) InverterfromNORgate
Input **=A**
Output **Q=A**

b) ANDgatefromNORgate Input s are **A** and **B**Output
Q = A.B



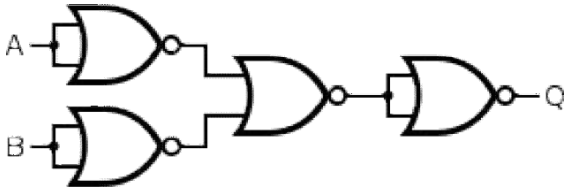
c) ORgatefromNORgate

InputsareAandB
Output **Q=A+B**



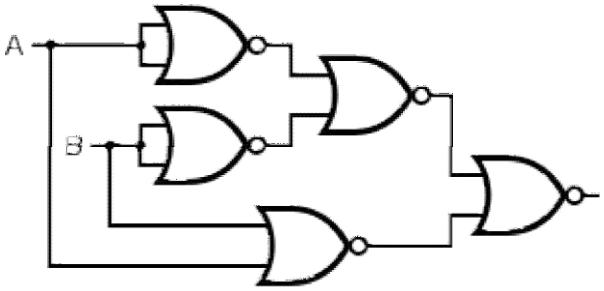
d) NANDgatefromNORgate

InputsareAandB
Output **Q=$\overline{A.B}$**



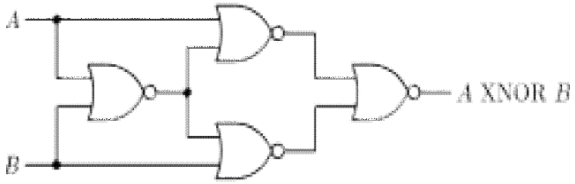
e) EX-ORgatefromNORgate

InputsareAandB
Output **Q=AB +AB**



f) EX-NORgateFromNORgate

Inputs are **A** and **B**_
Output **Q=AB +A B**



THRESHOLDLOGIC:-

INTRODUCTION:-

- Thethresholdelement,alsocalledthe thresholdgate(T-gate) isa much more powerful devicethanany of the conventional logic gates such as NAND, NOR and others.
- Complex,largeBooleanfunctionscanberealizedusingmuchfewerthresholdgates.
- Frequentlyasingle threshold gate can realizea verycomplexfunction whichotherwise mightrequire a large number of conventional gates.
- T-gate offersincomparablyeconomicalrealization; it hasnotfoundextensiveuse withthe digitalsystem designers mainly because of the following limitations.
  1. Itisverysensitivetoparametervariations.
  2. Itisdifficulttofabricate itinICform.

3. Thespeedofswitchingofthresholdelementsinmuchlowerthanthatofconventionalgates.

THETHRESHOLDELEMENTS:-

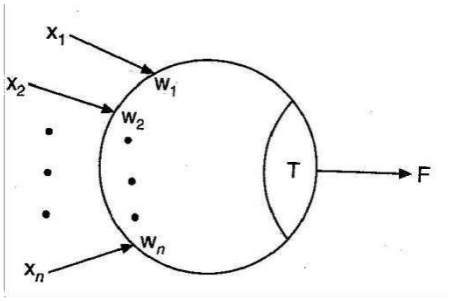
- A threshold element or gate has 'n' binary inputs x_1, x_2, ….., x_n; and a single binary output F. But in addition to those, it has two more parameters.
- Itsparametersareathreshold T and weights $w_1, w_2, \ldots, w_n$. The weights $w_1, w_2, \ldots, w_n$ are associated with the input variables x_1, x_2, …, x_n.
- The valueof thethreshold (T) and weightsmaybereal,positiveornegativenumber. The
- symbol of the threshold element is shown in fig.(a).
- Itisrepresentedbyacircle partitioned intotwoparts,onepart representstheweightsandother represents T.
- Itisdefinedas

$F(x_1, x_2, \ldots\ldots, x_n) = 1$ if and only if $\sum_{i=1}^{n} w_i x_i \geq T$

otherwise

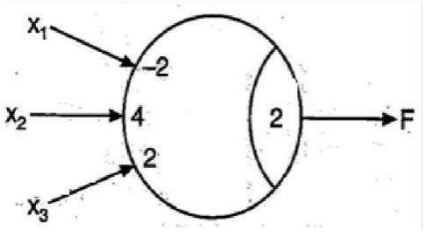
$F(x_1, x_2, \ldots\ldots, x_n) = 0$

- The sum and productoperationarenormalarithmetic operations and the sum $\sum_{i=1}^{n} w_i x_i \geq T$ iscalledthe weightedsumoftheelementorgate.



Example:-

ObtaintheminimalBooleanexpressionfromthethresholdgateshowninfigure.



Solution:-

The threshold gate with three inputs x_1, x_2, x_3 with weights $-2(w_1)$, $4(w_2)$ and $2(w_3)$ respectively. The value of threshold is 2(T). The table shown is the weighted sums and outputs for all input combinations. For this threshold gate, the weighted sum is

$$w = w_1x_1 + w_2x_2 + w_3x_3$$

$$= (-2)x_1 + (4)x_2 + (2)x_3$$

$$= -2x_1 + 4x_2 + 2x_3$$

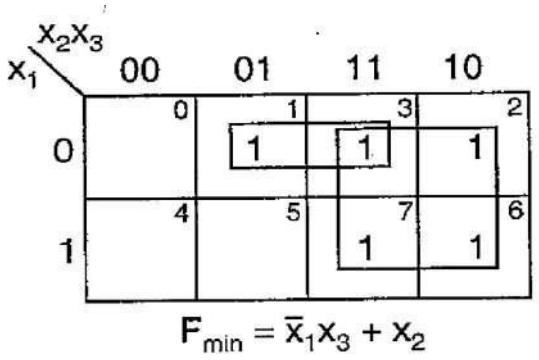
TheoutputFislogic1forw≥2anditis logic0forw<2

InputVariables			WeightedSum	Output
x_1	x_2	x_3	$w=-2x_1+4x_2+2x_3$	F
0	0	0	0	0
0	0	1	2	1
0	1	0	4	1
0	1	1	6	1
1	0	0	-2	0
1	0	1	0	0
1	1	0	2	1
1	1	1	4	1

From theinput-outputrelationisgiveninthetable,theBooleanexpressionfor theoutputis

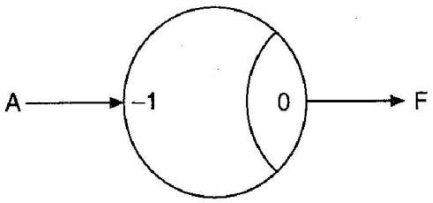
$$F=\sum m\,(1,2,3,6,7)$$

TheK-mapforF is



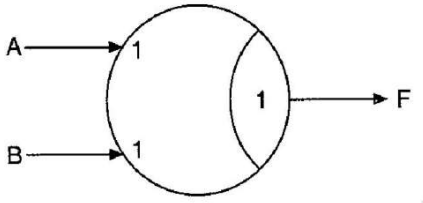
UNIVERSALITYOFAT-GATE:-

- A single T-gate can realizea large numberoffunctionsby merely changingeithertheweightsorthe threshold or both, which can be done by altering the value of the corresponding resistors.
- Sinceathresholdgate canrealize universalgates, i.e., NANDgatesand NORgates,athresholdgate is also a universal gate.
- SinglethresholdgatecannotrealizebyasingleT-gate
- RealizationoflogicgatesusingT-gatesisshownin thebelowfigure.



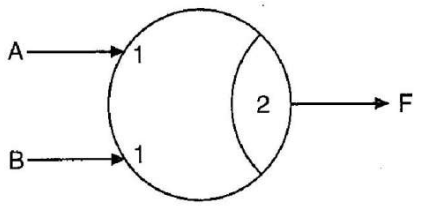
(a) NOT gate $F = \overline{A}$

Input	Weighted sum	Output
A	W = −A	F
0	0	1
1	−1	0



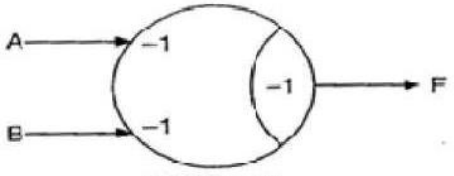
(b) OR gate F = A + B

Inputs		Weighted sum	Output
A	B	w = A + B	F
0	0	0	0
0	1	1	1
1	0	1	1
1	1	2	1



(c) AND gate F = AB

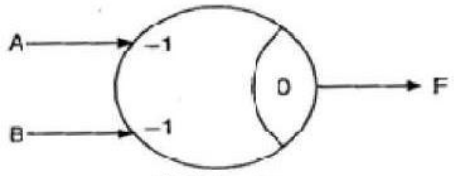
Inputs		Weighted sum	Output
A	B	w = A + B	F
0	0	0	0
0	1	1	0
1	0	1	0
1	1	2	1



NAND gate

(d) $F = \overline{A} + \overline{B} = \overline{AB}$

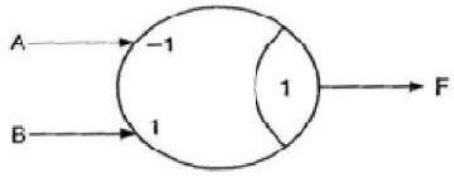
Inputs		Weighted sum	Output
A	B	w = −A − B	F
0	0	0	1
0	1	−1	1
1	0	−1	1
1	1	−2	0



NOR gate

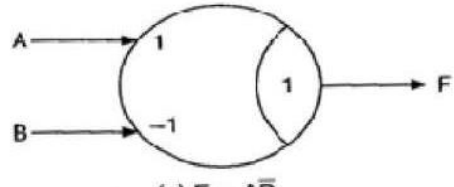
(e) $F = \overline{A} \cdot \overline{B} = \overline{A + B}$

Inputs		Weighted sum	Output
A	B	w = −A − B	F
0	0	0	1
0	1	−1	0
1	0	−1	0
1	1	−2	0



(f) $F = \overline{A}B$

Inputs		Weighted sum	Output
A	B	w = −A + B	F
0	0	0	0
0	1	1	1
1	0	−1	0
1	1	0	0



(g) $F = A\overline{B}$

Inputs		Weighted sum	Output
A	B	w = A − B	F
0	0	0	0
0	1	−1	0
1	0	1	1
1	1	0	0

BOOLEAN ALGEBRA

INTRODUCTION:-

- Switchingcircuitsarealsocalledlogiccircuits,gatescircuitsanddigitalcircuits.
- Switching algebra is also called Boolean algebra.
- Booleanalgebra isasystemof mathematicallogic. It isan algebraicsystemconsistingof the setof elements (0,1), two binary operators called OR and AND and unary operator called NOT.
- Itisthebasicmathematicaltoolintheanalysisand synthesisof switching circuits. It is a
- way to express logic functions algebraically.
- Anycomplex logic canbeexpressedbyaBooleanfunction.
- TheBooleanalgebraisgovernedbycertainwelldevelopedrulesandlaws.

AXIOMSANDLAWSOFBOOLEANALGEBRA:-

Axioms or postulates of Boolean algebra are set of logical expressions that are accepted without proof and upon which we can build a set of useful theorems. Actually, axioms are nothing more than the definitions of the three basic logic operations AND, OR and INVERTER. Each axiom can be interpreted as the outcome of an operation performed by a logic gate.

ANDoperation	**OR operation**	**NOToperation**
Axiom 1: 0. 0=0	Axiom5:0+0=0	Axiom 9:$\overline{1}$=0
Axiom 2:0.1=0	Axiom6:0+1=1	Axiom10:$\overline{0}$=1
Axiom 3:1.0=0	Axiom7:1+0=1	
Axiom 2:1.1=1	Axiom8:1+1=1	

1. ComplementationLaws:-

Theterm complementsimply meansto invert, i.e.to changes0s to 1s and1sto0s.Thefive lawsof complementation are as follows:

Law1: $0=\overline{1}$

Law2: $1=\overline{0}$

Law3: ifA =0,then$A=\overline{1}$

Law4: if$\overline{A}$=1,then$A=\overline{0}$

Law5: $\overline{\overline{A}}$=0(doublecomplementationlaw)

2. ORLaws:-

ThefourORlawsareasfollows

Law 1: A + 0 = 0(Null law)

Law2:A+ 1=1(Identitylaw)

Law 3: A + A = A

Law4: $A+\overline{A}=\overline{1}$

3. ANDLaws:-

The four AND laws are as follows

Law 1: A . 0 = 0(Null law)

Law2:A. 1=1(Identitylaw)

Law 3: A . A = A

Law4: $A.\overline{A}=0$

4. CommutativeLaws:-

CommutativelawsallowchangeinpositionofANDorORvariables.Therearetwocommutativelaws.

Law1: A +B =B+ A

Proof

A	B	A+B
0	0	0
0	1	1
1	0	1
1	1	1

=

B	A	B+A
0	0	0
0	1	1
1	0	1
1	1	1

Law2: A .B =B.A

Proof

A	B	A.B
0	0	0
0	1	0
1	0	0
1	1	1

=

B	A	B.A
0	0	0
0	1	0
1	0	0
1	1	1

Thislawcanbeextendedtoanynumberof variables. Forexample
A.B. C=B.C. A =C. A.B=B. A.C

5. AssociativeLaws:-

Theassociativelawsallowgroupingofvariables.Thereare2associative laws.

Law1: (A+B)+C=A+(B+C)

Proof

A	B	C	A+B	(A+B)+C
0	0	0	0	0
0	0	1	0	1
0	1	0	1	1
0	1	1	1	1
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

=

A	B	C	B+C	A+(B+C)
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	1
1	0	0	0	1
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

Law2:(A.B)C=A(B .C)

Proof

A	B	C	AB	(AB)C
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	0	0
1	0	0	0	0
1	0	1	0	0
1	1	0	1	0
1	1	1	1	1

=

A	B	C	B.C	A(B.C)
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	1	0
1	0	0	0	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Thislawcanbe extendedtoanynumberof variables. Forexample

A(BCD) = (ABC)D = (AB) (CD)

6. DistributiveLaws:-

Thedistributivelawsallowfactoringormultiplyingoutofexpressions.Therearetwodistributive laws.

Law1:A(B+C)=AB+AC

Proof

A	B	C	B+C	A(B+C)
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	0	0
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

=

A	B	C	AB	AC	A+(B+C)
0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	0	0	0
0	1	1	0	0	0
1	0	0	0	0	0
1	0	1	0	1	1
1	1	0	1	0	1
1	1	1	1	1	1

Law2: A +BC=(A+B)(A+C)

Proof **RHS**=(A+B)(A+C)

=AA + AC+ BA +BC

=A+AC+AB+BC

=A(1+C+B)+BC

=A. 1+BC (1+C+ B =1 +B=1)

=A+BC

=**LHS**

7. RedundantLiteralRule(RLR):-

Law 1: $A + \bar{A}B = A + B$

Proof

$A+A\bar{B} = (A+A)(\bar{A}+B)$

$= 1.(A+B)$

$= A+B$

Law2: $A(A+\bar{B}) = AB$ A(A +

Proof

B) $= AA + A\bar{B}$

$= 0 + AB$

$= AB$

8. IdempotenceLaws:-

Idempotencemeanssamevalue.

Law1: A.A=A

Proof

If A = 0, then A.A = 0.0 =0 = A If A=1, thenA.A =1.1 =1= A

ThislawstatesthatANDofavariablewithitselfisequaltothatvariableonly.

Law2: A+A = A

Proof

If A=0, thenA +A =0+ 0=0 =A

IfA=1,thenA+A=1+1=1=A

ThislawstatesthatORofavariable withitselfisequaltothatvariableonly.

9. AbsorptionLaws:-

Therearetwolaws:

Law1: A +A ·B= A

Proof

A+A·B=A(1+B)=A·1 =A

A	B	AB	A+AB
0	0	0	0
0	1	0	0
1	0	0	1
1	1	1	1

Law2: A(A+ B)= A

Proof

A(A +B)=A · A+A·B=A +AB=A(1 +B)=A·1 =A

A	B	A+B	A(A+B)
0	0	0	0
0	1	1	0
1	0	1	1
1	1	1	1

10. ConsensusTheorem(IncludedFactorTheorem):-

Theorem 1:

$AB+\bar{A}C+BC=AB+\bar{A}C$

Proof

LHS=$AB+\bar{A}C+BC$

$=AB+AC+BC(A+A)$

$=AB+A\bar{C}+B\bar{C}A+BCA$

$=AB(1+\bar{C})+AC(1+B)$

$=AB(1)+AC(1)$

$=AB+\bar{A}C$

$=$**RHS**

Theorem2:

$(A+B)(\bar{A}+C)(B+C)=(A+B)(\bar{A}+C)$

Proof

$LHS=(A+B)(\bar{A}+C)(B+\bar{C})$

$=(AA+AC+BA+\bar{B}C)(B+C)$

$=(AC+BC+AB)(\bar{B}+C)$

$=ABC+BC+AB+AC+BC+ABC$

$=AC+BC+AB$

$RHS=(A+B)(\bar{A}+\bar{C})$

$=A\bar{A}+AC+BC+\bar{A}B$

$=AC+BC+AB$

$=LHS$

11. TranspositionTheorem:-

Theorem:

$AB+\bar{A}C=(A+C)(A+B)$

Proof

RHS=$(A+C)(\bar{A}+B)$

$=A\bar{A}+CA+\bar{A}B+CB$

$=0+\bar{A}C+AB+BC$

$=\bar{A}C+AB+BC(A+\bar{A})$

$=AB+\bar{A}BC+AC+ABC$

$=AB+AC$

$=LHS$

12. DeMorgan'sTheorem:-

DeMorgan'stheoremrepresentstwolawsinBooleanalgebra.

Law1: $\overline{A+B}=\bar{A}\cdot\bar{B}$

Proof

A	B	A+B	$\overline{A+B}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

=

A	B	$\bar{A}$	$\bar{B}$	$\bar{A}\bar{B}$
0	0	1	1	1
0	1	1	0	0
1	0	0	1	0
1	1	0	0	0

Thislawstatesthat thecomplementof a sumof variablesisequalto theproductof theirindividual complements.

Law2: $A \cdot B = \overline{A} + \overline{B}$

Proof

A	B	A.B	$\overline{A.B}$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

=

A	B	$\overline{A}$	$\overline{B}$	$\overline{A}+\overline{B}$
0	0	1	1	1
0	1	1	0	1
1	0	0	1	1
1	1	0	0	0

This lawstates that thecomplementof aproductof variables isequaltothe sumof theirindividual complements.

DUALITY:-

The implicationof thedualityconcept isthatoncea theoremorstatement isproved, thedualalsothus stand proved. This is called the principle of duality.

$[f(A,B,C,\ldots..,0,1,+,\cdot)]_d = f(A,B,C,\ldots.,1,0,\cdot,+)$

Relationsbetweencomplementanddual

$f_c(A,B,C,\ldots..) = f(\overline{A},\overline{B},\overline{C},\ldots.) = f_d(A,B,C,\ldots)$

$f_d(A,B,C,\ldots..) = f(\overline{A},\overline{B},\overline{C},\ldots) = \overline{f_c(A,B,C,\ldots..)}$

Thefirstrelationstatesthatthecomplementofafunctionf(A,B,C,…)canbeobtainedbycomplementingall thevariables inthedualfunctionf_d(A,B, C,…..).

Thesecondrelationstatesthatthedualcanbeobtainedbycomplementing alltheliteralsin f(A, B,C, ….).

DUALS:-

	Givenexpression	***Dual***
1.	$\overline{0}=1$	$\overline{1}=0$
2.	$0 \cdot 1 = 0$	$1+0=1$
3.	$0 \cdot 0 = 0$	$1+1=1$
4.	$1 \cdot 1 = 1$	$0+0=0$
5.	$A \cdot 0 = 0$	$A+1=1$
6.	$A \cdot 1 = A$	$A+0=A$
7.	$A \cdot A = A$	$A+A=A$
8.	$A \cdot \overline{A} = 0$	$A+\overline{A}=1$
9.	$A \cdot B = B \cdot A$	$A+B=B+A$
10.	$A \cdot (B \cdot C) = (A \cdot B) \cdot C$	$A+(B+C)=(A+B)+C$
11.	$A \cdot (B+C) = AB+AC$	$A+BC=(A+B)(A+C)$
12.	$A(A+B)=A$	$A+AB=A$
13.	$A \cdot (A \cdot B) = A \cdot B$	$A+A+B=A+B$
14.	$AB=A+B$	$A+B=\overline{A}\,\overline{B}$
15.	$(A+\overline{B})(A+\overline{C})(B+\overline{C})=(A+B)(A+C)$	$AB+\overline{A}\overline{C}+BC=AB+AC$
16.	$A+BC=(A+B)(A+C)$	$A(B+C)=AB+AC$
17.	$(A+C)(\overline{A}+B)=AB+\overline{A}C$	$AC+\overline{A}B=(A+B)(\overline{A}+C)$
18.	$(A+B)(C+D)=AC+AD+BC+BD$	$(AB+CD)=(A+C)(A+D)(B+C)(B+D)$
19.	$A+B=AB+AB+AB$	$AB=(A+B)(A+B)(A+B)$
20.	$\overline{A}B+A\overline{B}+AB=0$	$\overline{A+B}\cdot A \cdot (A+B)=1$

SUM-OF-PRODUCTSFORM:-

- This is also calleddisjunctive CanonicalForm(DCF)orExpandedSumof Products FormorCanonical Sum of Products Form.
- In thisform, the function isthe sumof anumberof products terms where each producttermcontainsall variables of the function either in complemented or uncomplemented form.
- This can also be derivedfrom the truth table by finding the sum of all the terms that corresponds to thosecombinationsforwhich'f'assumesthevalue1.

  Forexample

  $f(A, B,C)=A\bar{B}+ BC$

  $=A\bar{B} (C+C)+B\bar{C}(A +A)$

  $=A BC+A\bar{B}C+AB\bar{C}+ABC$
- Theproducttermwhichcontainsallthe variablesof thefunctionseitherincomplementedor uncomplemented form is called a minterm.
- Themintermisdenotedasmo,m1,m2....
- An 'n'variablefunction canhave2nminterms.
- Anotherwayof representing thefunction in canonicalSOPformis theshowingthe sumof mintermsfor which the function equals to 1.

  For example

  $f(A, B,C)=m_1+m_2+m_3+m_5$

  or

  $f(A,B,C) =\sum m (1,2,3,5)$

  where$\sum$mrepresentsthe sumofallthemintermswhose decimalcodesaregiven theparenthesis.

PRODUCT-OF-SUMSFORM:-

- Thisform isalso called as Conjunctive CanonicalForm (CCF) or Expanded Product- of - Sums Form orCanonicalProductOfSumsForm.
- Thisisbyconsideringthe combinations forwhichf=0 Each
- term is a sum of all the variables.
- Thefunctionf(A, B,C)=$(A+\bar{B} +C\cdot C)+(\bar{A}+\bar{B}+C\cdot C)$

  $=(A+B +C)(A +B +C)(A+B+C)(A+B +C)$
- The sum term which contains each of the 'n' variables in eithercomplemented or uncomplemented form iscalledamaxterm.
- Maxterm is representedas M_0, M_1,M_2, ...

  Thus CCFof'f'maybewrittenas

  $f(A,B,C)=M_0\cdot M_4\cdot M_6\cdot M_7$

  or

  $f(A,B,C)=(0,4, 6, 7)$

  Whererepresentedtheproductofallmaxterms.

CONVERSIONBETWEENCANONICALFORM:-

The complement ofafunctionexpressedasthe sumof minterms equals the sumof minterms missingfromthe original function.

Example:-

$f(A,B,C) =\sum m(0,2,4,6,7)$

Thishasa complementthatcanbeexpressedas

$f(\bar{A},\bar{B},\bar{C})=\sum m(1,3,5)=m_1+m_3+m_5$

Ifwecomplementf byDe-Morgan's theoremweobtain'f'inaform.

$f=\overline{(m_1+m_3+m_5)}=\bar{m}_1.\bar{m}_3.\bar{m}_5$

$= M_1M_3M_5 = \prod M(1,3,5)$

Example:-

Expand $A(\bar{A}+B)(A+\bar{B}+C)$ to maxterms and minterms.

Solution:-

In POS form

$A(\bar{A}+B)(\bar{A}+B+C)\,A$

$= A + B\,B + C\bar{C}$

$=(A+B)(\bar{A}+B)+C\cdot\bar{C}$

$=(A+B+C\bar{C})(A+B+C\bar{C})$

$= (A+B+C)(A+B+\bar{C})(\bar{A}+B+\bar{C})(A+B+\bar{C})\,A+\bar{B}$

$B = A + \bar{B}+C\cdot\bar{C}$

$=(A+B+C)(A+\bar{B}+C)$

Therefore

$A(\bar{A}+B)(A+\bar{B}+C)$

$=(A+B+C)(A+B+C)(A+B+C)(A+B+C)(A+\bar{B}+C)(A+\bar{B}+C)$

$= (000)(001)(010)(011)(100)(101)$

$=M_0\cdot M_1\cdot M_2\cdot M_3\cdot M_4\cdot M_5$

$=\prod M(0,1,2,3,4,5)$

The maxterms M_6 and M_7 are missing in the POS form.

So, the SOP form will contain the minterms 6 and 7

KARNAUGH MAP OR K-MAP:-

- The K- map is a chart or a graph, composed of an arrangement of adjacent cells, each representing a particular combination of variables in sum or product form.
- The K- map is systematic method of simplifying the Boolean expression.

TWO VARIABLE K-MAP:-

A two variable expression can have $2^2=4$ possible combinations of the input variables A and B.

Mapping of SOP Expression:-

- The 2 variable K-map has $2^2=4$ squares. These squares are called cells.
- A '1' is placed in any square indicates that corresponding minterm is included in the output expression, and a 0 or no entry in any square indicates that the corresponding minterm does not appear in the expression for output.

A \ B	0	1
0	$A\bar{B}$	$\bar{A}B$
1	$\bar{A}\bar{B}$	AB

Example:-

Map expression $f=A\bar{B}+\bar{A}\bar{B}$

Solution:-

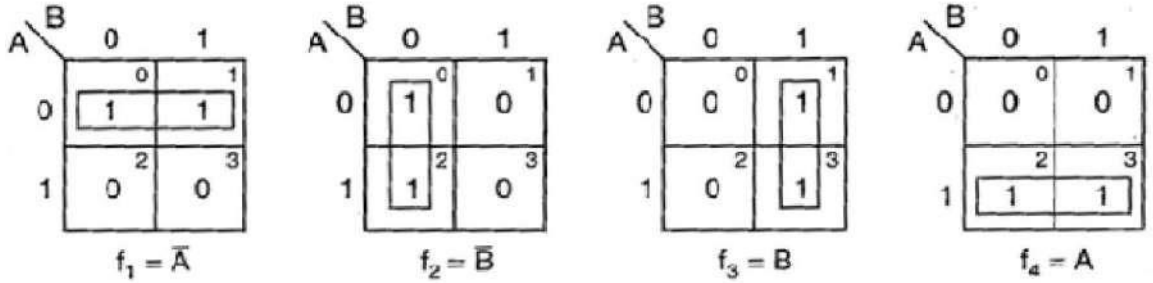
The expression minterms is F

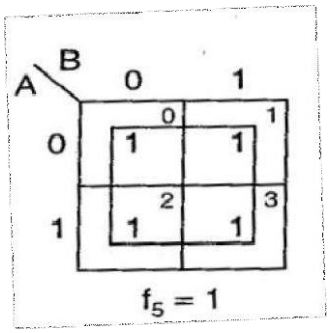
$= m_1 + m_2 = m(1, 2)$

A \ B	0	1
0	0 (0)	1 (1)
1	1 (2)	0 (3)

MinimizationofSOPExpression:-

To minimizea Booleanexpression givenin theSOPform byusingK- map, theadjacentsquares having1s, that is minterms adjacent to each other are combined to form larger squares to eliminate somevariables. Thepossibleintermgrouping inatwovariableK-mapare shownbelow





- Twominterms,which are adjacent to each other, can be combined to form a bigger square called 2–squareorapair.Thiseliminatesonevariablethatisnotcommontoboththeminterms.
- Two 2-squares adjacent to each other can be combined to form a 4- square. A 4- square eliminates 2 variables. A 4-square is called a quad.
- Consider only those variables which remain constant throughout the square, and ignore the variables which are varying. The non-complemented variable is the variable remaining constant as 1.The complemented variable is the variable remaining constant as a 0 and the variables are written as a product term.

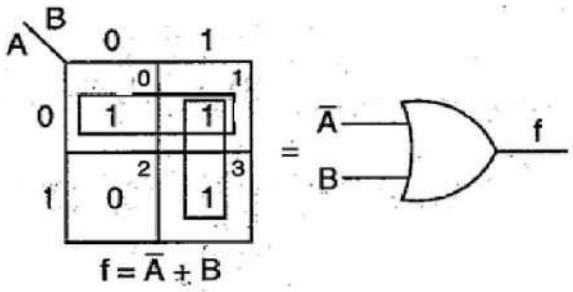
Example:-

Reducetheexpressionf=$A\bar{B}+\bar{A}\bar{B}+\bar{A}B$using mapping.

Solution:-

Expressedintermsof minterms,thegivenexpressionis f =

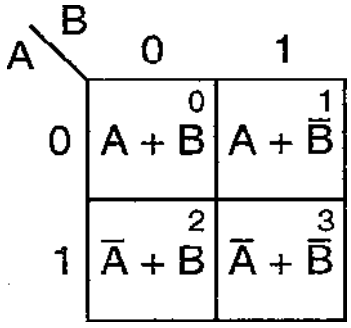
$m_0 + m_1 + m_3 = \sum m\ (0, 1, 3)$



F =$\bar{A}$+B

MappingofPOSExpression:-

Each sum term in the standard POS expression is called a Maxterm. A function in two variables (A,B) has 4 possible maxterms, $A + B$, $A + \overline{B}$, $\overline{A} + B$ and $\overline{A} + \overline{B}$. They are represented as M_0, M_1, $M_2$and M_3respectively.



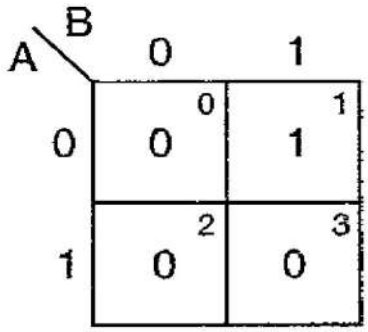
ThemaxtermofatwovariableK-map

Example:-

Plottheexpressionf=(A+B)(A+B)(A+B)

Solution:-

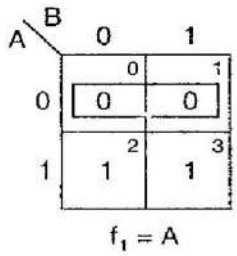
Expressionintermsofmaxtermsis f=∏M(0,2,3)



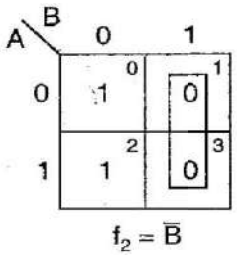
MinimizationofPOSExpressions:-

In POS form the adjacent 0s are combined into large square as possible. If the squares having complemented variablethen the value remain constant as a 1 and the non-complemented variable ifitsvalue remains constant as a 0 along the entire square and then their sum term iswritten.

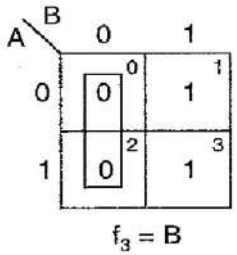
Thepossible maxtermsgroupinginatwo variableK-mapare shownbelow



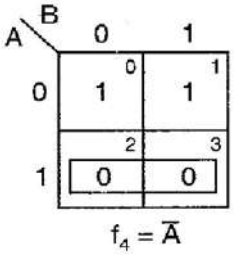
$f_1 = A$



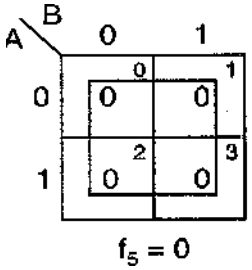
$f_2 = \overline{B}$



$f_3 = B$



$f_4 = \overline{A}$



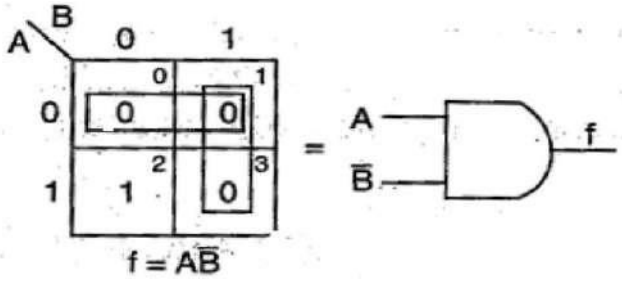
$f_5 = 0$

Example:-

Reducetheexpressionf=(A+B)(A+$\bar{B}$)(A+$\bar{B}$)using mapping

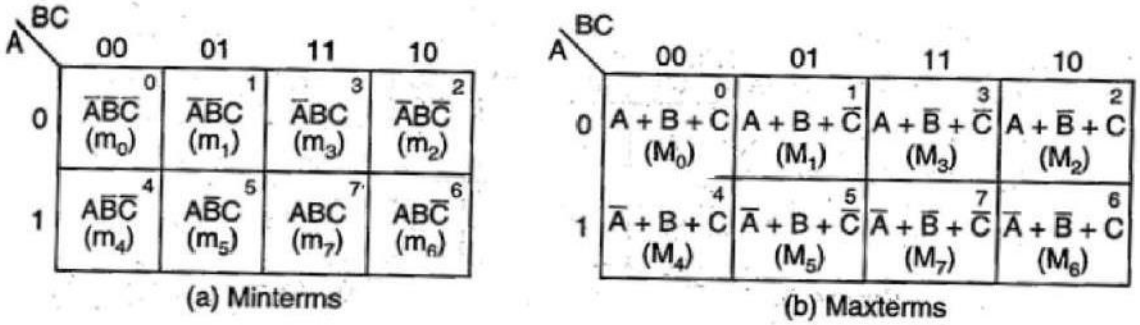
Solution:-

The givenexpressionintermsofmaxtermsis f=ΠM(0,1, 3)



THREEVARIABLEK-MAP:-

Afunctioninthreevariables(A,B,C)canbeexpressedinSOPandPOSformhavingeightpossible combination.A three variableK-maphave8 squaresorcellsandeachsquare onthemaprepresentsa minterm or maxterm is shown in the figure below.

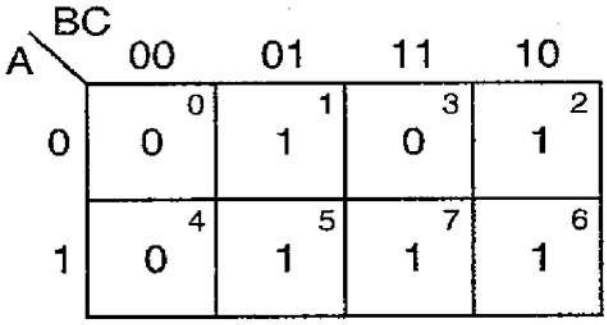


Example:-

Maptheexpressionf= A$\bar{B}\bar{C}$+AB$\bar{C}$+A$\bar{B}$C+A$\bar{B}\bar{C}$+AB$\bar{C}$

Solution:-

SointheSOPform theexpressionisf=∑m (1,5,2,6,7)

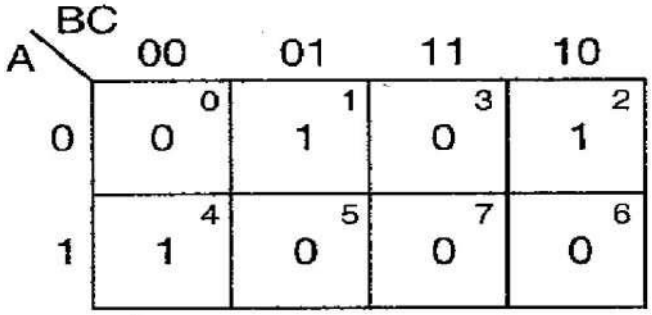


Example:-

Mapthe expressionf= (A+ B+ C)(A+ $\bar{B}$+C)(A+ $\bar{B}$+ $\bar{C}$)(A+ $\bar{B}$+ C)(A+ B+ $\bar{C}$)

Solution:-

Sointhe POSformtheexpression isf=ΠM(0,5,7, 3,6)

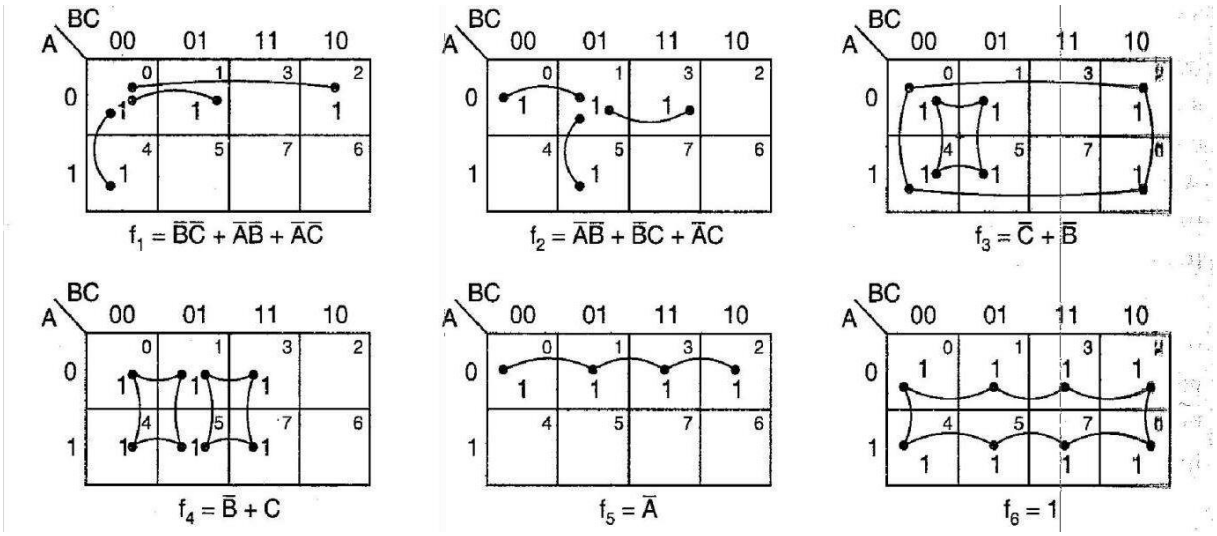


MinimizationofSOPandPOSExpressions:-

ForreducingtheBooleanexpressionsinSOP(POS)formthefollowingstepsaregivenbelow

- DrawtheK-mapandplace 1s(0s)corresponding totheminterms(maxterms)ofthe SOP(POS) expression.
- In themap1s(0s) whicharenotadjacenttoanyother1(0)are theisolated minterms (maxterms). They are to be read as they are because they cannot be combined even into a2-square.
- Forthose1s(0s)whichareadjacenttoonlyoneother1(0)makethempairs (2squares).
- For quads (4- squares)and octet (8squares)of adjacent1s (0s) evenif theycontain some1s (0s) which have already been combined. They must geometrically form a square or a rectangle.
- Forany1s(0s)that have notbeencombined yetthen combine themintobiggersquaresifpossible. Form the
- minimal expression by summing (multiplying) the product (sum) terms of all the groups.

SomeofthepossiblecombinationsofmintermsinSOPform

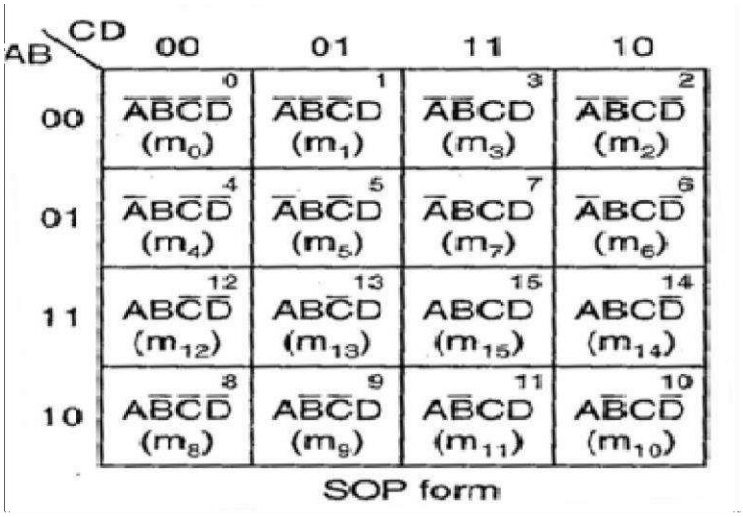


These possiblecombinationsarealsoforPOSbut1sarereplacedby0s.

FOURVARIABLEK-MAP:-

A four variable (A, B, C, D) expression can have 2^4 = 16 possible combinations of input variables. A four variable K-map has 2^4 = 16 squares or cells and each square on the map represents either a minterm or a maxterm as shown in thefigure below. Thebinarynumber designations of the rows and columnsare in the gray code. The binary numbers along the top of the map indicate the conditions of C and D along any column and binary numbers along left side indicate the conditions of A and B along any row. The numbers in the top right corners of the squares indicate the minterm or maxterm desginations.

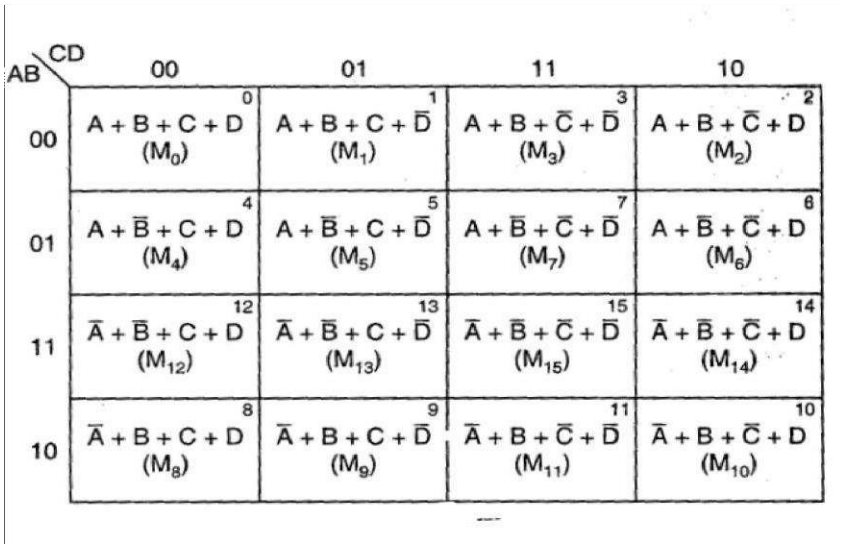
SOPFORM



AB \ CD	00	01	11	10
00	$\bar{A}\bar{B}\bar{C}\bar{D}$ (m_0) 0	$\bar{A}\bar{B}\bar{C}D$ (m_1) 1	$\bar{A}\bar{B}CD$ (m_3) 3	$\bar{A}\bar{B}C\bar{D}$ (m_2) 2
01	$\bar{A}B\bar{C}\bar{D}$ (m_4) 4	$\bar{A}B\bar{C}D$ (m_5) 5	$\bar{A}BCD$ (m_7) 7	$\bar{A}BC\bar{D}$ (m_6) 6
11	$AB\bar{C}\bar{D}$ (m_{12}) 12	$AB\bar{C}D$ (m_{13}) 13	$ABCD$ (m_{15}) 15	$ABC\bar{D}$ (m_{14}) 14
10	$A\bar{B}\bar{C}\bar{D}$ (m_8) 8	$A\bar{B}\bar{C}D$ (m_9) 9	$A\bar{B}CD$ (m_{11}) 11	$A\bar{B}C\bar{D}$ (m_{10}) 10

SOP form

POSFORM



AB \ CD	00	01	11	10
00	$A+B+C+D$ (M_0) 0	$A+B+C+\bar{D}$ (M_1) 1	$A+B+\bar{C}+\bar{D}$ (M_3) 3	$A+B+\bar{C}+D$ (M_2) 2
01	$A+\bar{B}+C+D$ (M_4) 4	$A+\bar{B}+C+\bar{D}$ (M_5) 5	$A+\bar{B}+\bar{C}+\bar{D}$ (M_7) 7	$A+\bar{B}+\bar{C}+D$ (M_6) 6
11	$\bar{A}+\bar{B}+C+D$ (M_{12}) 12	$\bar{A}+\bar{B}+C+\bar{D}$ (M_{13}) 13	$\bar{A}+\bar{B}+\bar{C}+\bar{D}$ (M_{15}) 15	$\bar{A}+\bar{B}+\bar{C}+D$ (M_{14}) 14
10	$\bar{A}+B+C+D$ (M_8) 8	$\bar{A}+B+C+\bar{D}$ (M_9) 9	$\bar{A}+B+\bar{C}+\bar{D}$ (M_{11}) 11	$\bar{A}+B+\bar{C}+D$ (M_{10}) 10

MinimizationofSOPandPOSExpressions:-

ForreducingtheBooleanexpressionsinSOP(POS)formthefollowingstepsaregivenbelow

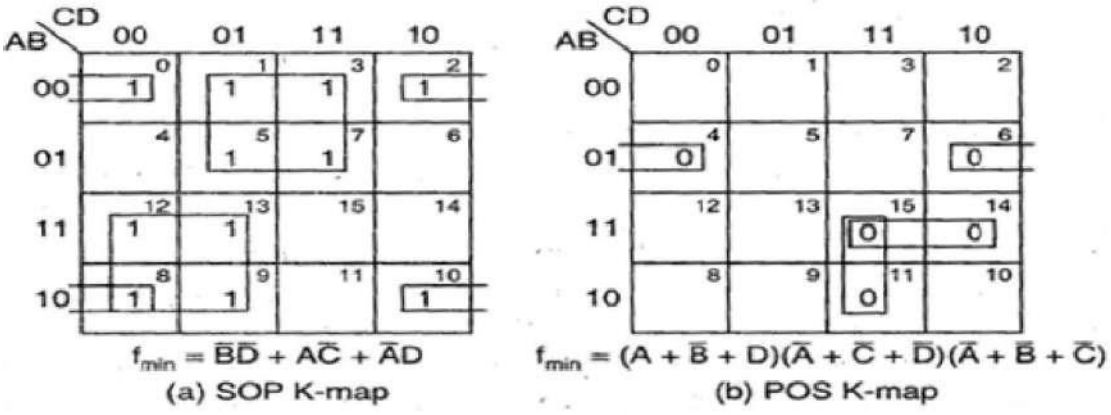
- Draw the K-mapandplace1s(0s) correspondingto theminterms (maxterms)of theSOP (POS) expression.
- In themap1s(0s) whicharenotadjacenttoanyother1(0)are theisolated minterms (maxterms). They are to be read as they are because they cannot be combined even into a2-square.
- For those 1s (0s) which are adjacent to only one other 1(0) make them pairs (2squares).
- For quads (4- squares)and octet (8 squares)of adjacent 1s (0s) even if they contain some 1s (0s) which have already been combined. They must geometrically form a square or a rectangle.
- Forany1s(0s)that have notbeencombined yetthen combine themintobiggersquares ifpossible. Form the
- minimal expression by summing (multiplying) the product (sum) terms of all the groups.

Example:-

Reduceusingmappingtheexpressionf=∑m(0, 1,2, 3, 5,7,8,9,10,12, 13)

Solution:-

The givenexpression inPOSformisf=ΠM(4, 6,11, 14,15)andin SOPformf=∑m(0,1, 2,3,5,7, 8,9, 10, 12, 13)



TheminimalSOPexpressionisf_{min}=$B\overline{D}+\overline{A}C+A\overline{D}$

Theminimal POSexpressionisf_{min}=(A+B+$\overline{D}$) (A+C+$\overline{D}$) ($\overline{A}$+B+C)

DON'TCARECOMBINATIONS:-

The combinationsforwhichthe valuesof theexpression arenot specified are called don'tcare combinations or optional combinations and such expression stand incompletely specified. The output is a don't care for these invalid combinations. The don't care terms are denoted by d or X. During the process of designing using SOP maps, each don't care is treated as 1 to reduce the map otherwise it is treated as 0 and left alone. During the process of designingusing POS maps,each don't care is treated as0 to reduce the map otherwise it is treated as 1 and left alone.

A standard SOP expression with don't cares can be converted into standard POS form by keeping the don't cares as they are, and the missing minterms of the SOP form are written as the maxterms of the POS form. Similarly, to convert a standard POS expression with don't cares can be converted into standard SOP form by keeping the don't cares as they are, and the missing maxterms of the POS form are written as the minterms of the SOP form.

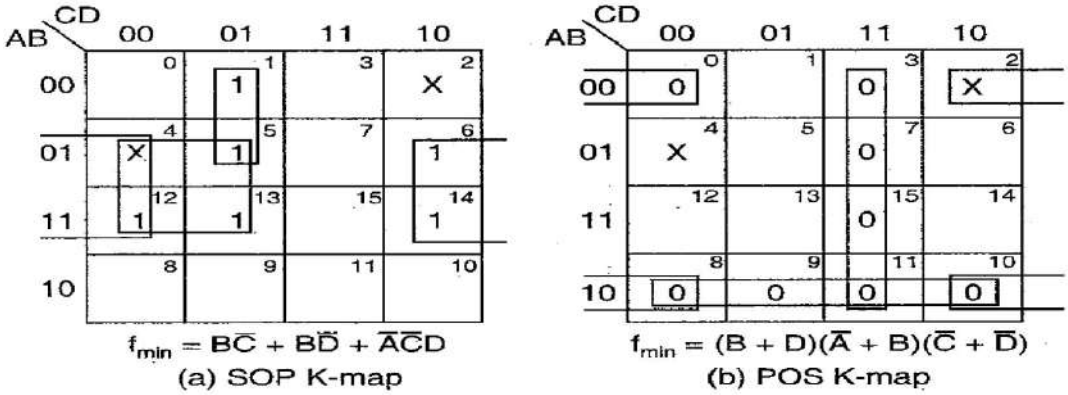
Example:-

Reducetheexpressionf=∑m(1, 5,6, 12,13, 14)+d(2, 4)usingK-map.

Solution:-

ThegivenexpressioninSOPform isf=∑m (1,5,6,12,13,14)+d(2,4)

The givenexpression inPOSformisf=ΠM(0,3,7, 8,9, 10,11,15)+ d(2, 4)



TheminimalofSOPexpressionisf_{min}=$BC+B\overline{D}+AC\overline{D}$

TheminimalofPOSexpressionisf_{min}=$(B+D)(A+B)(\overline{C}+D)$

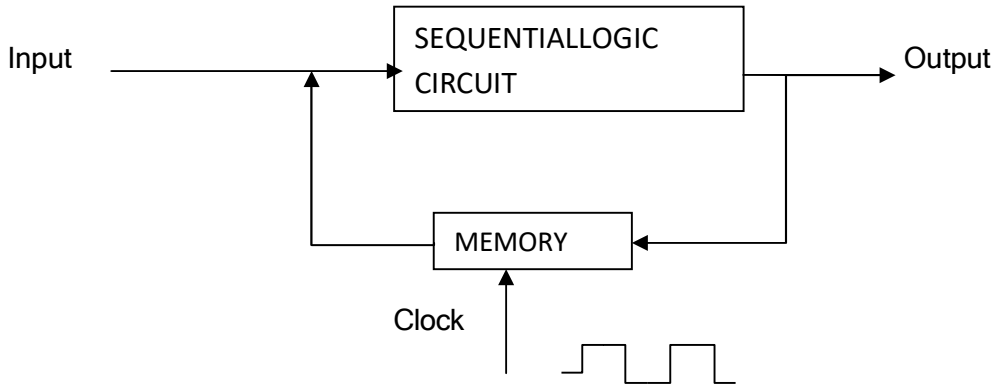
SEQUENTIALLOGICCIRCUIT

SEQUENTIALCIRCUIT:-

- It is a circuit whose output depends upon the present input, previous output and the sequence inwhich the inputs are applied.

HOW THESEQUENTIALCIRCUITISDIFFERENTFROMCOMBINATIONALCIRCUIT?:-

- In combinational circuit output depends upon present input at any instant of time and do notusememory. Hence previous input does not have any effect on the circuit. But sequential circuit hasmemory and depends upon present input and previous output.
- Sequential circuits are slower than combinational circuits and these sequential circuits are harder to design.



[BlockdiagramofSequentialLogicCircuit]

- Thedata storedby thememoryelementat anygiven instantof time iscalled the present state of sequential circuit.

TYPES:-

Sequentiallogiccircuits(SLC)areclassifiedas

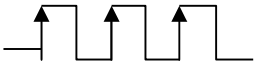
(i) SynchronousSLC
(ii) AsynchronousSLC

- TheSLCthatare controlled byclockarecalled synchronousSLCand those whicharenot controlled by a clock are asynchronous SLC.
- Clock:- A recurring pulse is called a clock.

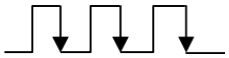
FLIP-FLOPANDLATCH:-

- Aflip-floporlatchisacircuit thathastwostablestatesandcanbeused tostoreinformation.
- Aflip-flop isabinarystoragedevice capableof storing onebitof information. Ina stable state,the output of a flip-flop is either 0 or 1.
- Latchisanon-clockedflip-flopanditisthebuilding blockfortheflip-flop.
- A storageelement in digitalcircuit can maintainabinarystate indefinitelyuntildirectedbyaninput signal to switch state.
- Storageelement that operatewithsignallevelarecalledlatchesandthoseoperate with clock transition are called as flip-flops.

- The circuit canbemade to change statebysignalsapplied to oneormore controlinputsandwill have one or two outputs.
- Aflip-flopiscalled so becauseitsoutputeitherflipsorflopsmeaning to switchbackandforth.
- Aflip-flop isalsocalleda bi-stable multi-vibratorasit hastwo stable states. Theinputsignalswhich command the flip-flop to change state are called excitations.
- Flip-flopsarestorage devicesandcan store1or0.
- Flip-flopsusing theclocksignalare calledclockedflip-flops. Controlsignalsareeffective only if theyare applied in synchronization with the clock signal.
- Clock-signalsmaybepositive-edgetriggeredornegative-edgetriggered.
- Positive-edgetriggeredflip-flopsarethose in which statetransitions takeplaceonly atpositive-going edge of the clock pulse.



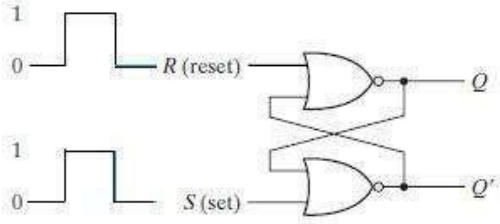
- Negative-edge triggeredflip-flopsarethose in which state transition takeplaceonly atnegative-going edge of the clock pulse.



- Some common type of flip-flops include
  a) SR(set-reset)F-F
  b) D(dataor delay)F-F
  c) T(toggle) F-Fand
  d) JKF-F

SRlatch:-

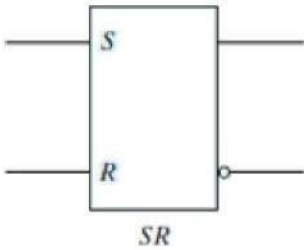
- TheSRlatch isa circuitwithtwocross-coupledNORgatesortwo cross-coupled NANDgates.
- It has twooutputs labeledQandQ'.TwoinputsaretherelabeledSforsetandRfoereset.
- Thelatchhastwousefulstates.WhenQ=0andQ'=1theconditioniscalledresetstateandwhenQ=1 andQ'=0theconditioniscalledsetstate.
- NormallyQand Q'are complementofeachother.
- The figure represents a SR latch with two cross-coupled NOR gates. The circuit has NOR gates and as we know ifany one ofthe input for a NOR gate is HIGH then its output will be LOW and ifboth the inputs are LOW then only the output will be HIGH.



- Under normal conditions, both inputs of the latch remain at 0 unless the state has to be changed. The application of a momentary 1 to the S input causes the latch to go to the set state. The S input must go back to 0 before any other changes take place, in order to avoid the occurrence of an undefined next state that results from the forbidden input condition.
- The first condition (S= 1, R = 0) is the action that must be taken by input Stobring thecircuit to theset state. Removing the active input from S leaves thecircuit in the same state. After bothinputs returnto0, it is then possible to shift to the reset state by momentary applying a 1 to the R input. The 1 can then be removed from R, whereupon the circuit remains in the reset state. When both inputs S and R are equal to 0, the latch can be in either the set or the reset state, depending on which input was most recently a 1.

- If a 1 is applied to both the S and R inputs of the latch, both outputs go to 0. This action produces an undefined next state, because the state that results from the input transitions depends on the order in which they return to 0. It also violates the requirement that outputs be the complement of each other. In normal operation, this condition is avoided by making sure that 1's are not applied to both inputs simultaneously.
- Truth table for SR latch designed with NOR gatesis shownbelow.

Input		Output				Comment
S	R	Q	Q'	Q_{Next}	Q'_{Next}	
0	0	0	1	0	1	Nochange
0	0	1	0	1	0	
0	1	0	1	0	1	Reset
0	1	1	0	0	1	
1	0	0	1	1	0	Set
1	0	1	0	1	0	
1	1	0	1	X	X	Prohibited state
1	1	1	0	X	X	



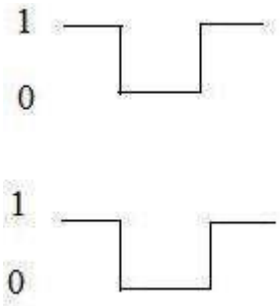
SymbolforSRNORLatch

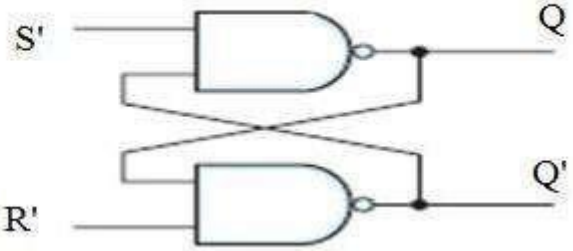
RacingCondition:-

In caseof aSRlatch whenS=R=1inputis givenboththeoutput willtry tobecome0. This is called Racing condition.

SRlatchusingNANDgate:-

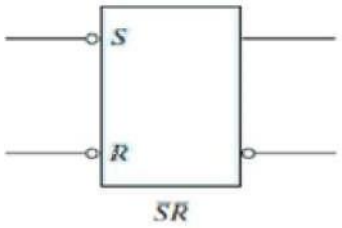
- The below figure represents a SR latch with two cross-coupled NAND gates. The circuit hasNAND gates and as weknow ifany one oftheinput for aNAND gateis LOWthenits outputwill beHIGH and if both the inputs are HIGH then only the output will be LOW.
- It operates with both inputs normally at 1, unless the state of the latch has to be changed.The application of 0 to the S input causes output Q to go to 1, putting the latch in the set state. When the S input goes back to 1, the circuit remains in the set state. After bothinputs go back to 1, we are allowedto change the state of the latch by placing a 0 in the R input. This action causes the circuit to go to the reset state and stay there even after both inputs return to 1.





- The condition that is forbidden for the NAND latch is both inputs being equal to 0 at the same time, an input combination that should be avoided.

Incomparing the NANDwiththe NOR latch, note that theinputsignals for theNAND require thecomplement of those values used for the NOR latch. Because the NAND latch requires a 0 signal to change its state, it is sometimes referredtoasanS'R' latch.The primes(or,sometimes,barsover theletters) designatethe fact that the inputs must be in their complement form to activate the circuit.



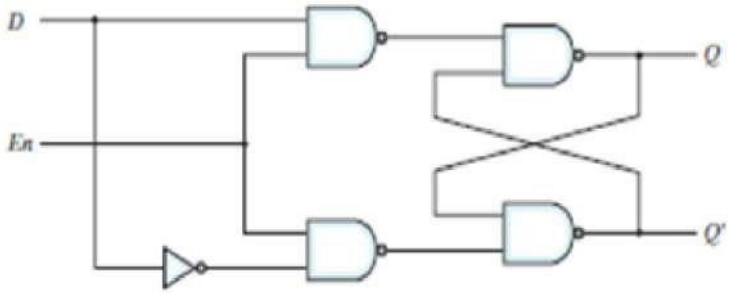
SR

Theaboverepresentsthe symbolfor invertedSR latchorSR latchusing NANDgate. Truth table for SR latch using NAND gate or Inverted SR latch

S	R	Q_{next}	Q'_{next}
0	0	Race	Race
0	1	0	1(Reset)
1	0	1	0(Set)
1	1	Q(Nochange)	Q' (Nochange)

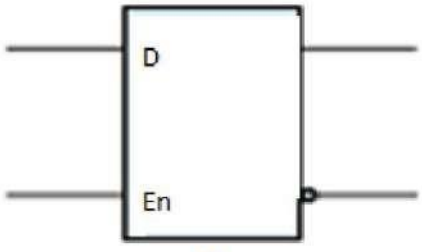
D LATCH:-

- Onewaytoeliminatethe undesirable conditionof theindeterminate state intheSRlatch istoensure that inputs S and R are never equal to 1 at the same time.



(a) Logic diagram

- This is done in the D latch. This latch has only two inputs: D (data) and En(enable).
- The Dinputgoesdirectlytothe Sinput, andits complement isappliedtothe Rinput.



D

(SymbolforD-Latch)

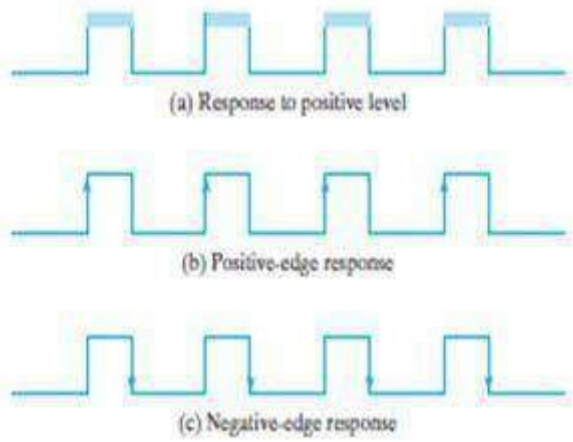
- As long as the enable input is at 0, the cross-coupled SR latch has both inputs at the 1 level and the circuitcan'tchangestateregardlessofthevalue ofD. The
- below represents the truth table for the D-latch.

En	D	Next StateofQ
0	X	Nochange
1	0	Q=0;ResetState
1	1	Q=1;Set State

- The D input is sampled when En = 1. IfD= 1, the Q output goes to 1, placing the circuit in the setstate. IfD = 0, output Q goes to 0, placing the circuit in the reset state. This situation provides a pathfrom input D to the output, and for this reason, the circuit is often called a TRANSPARENT latch.

TRIGGERINGMETHODS:-

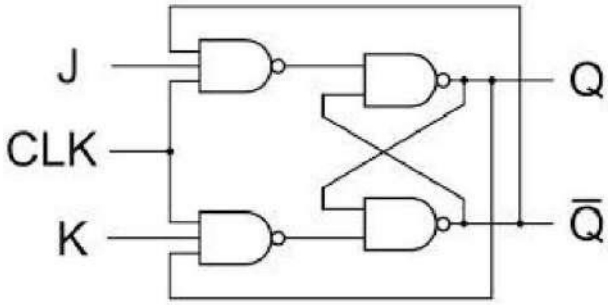
- The state of a latch or flip-flop is switched by a change in the control input. This momentary change is called a trigger, and the transition it causes is said to trigger theflip-flop.
- Flip-flopcircuits areconstructedinsuch away as tomake them operateproperly when they are partof a sequential circuit that employs a common clock.
- The problem with the latch is that it responds to a change in the level of a clock pulse. For proper operation of a flip-flop it should be triggered only during a signaltransition.
- This can be accomplished by eliminating the feedback path that is inherent in the operation of the sequential circuit using latches. A clock pulse goes through two transitions: from 0 to 1 and the return from 1 to 0.
- A ways that a latch can be modified to form a flip-flop is to produce a flip-flop that triggers only during a signaltransition (from 0 to 1 orfrom 1 to 0)of the synchronizing signal (clock) and is disabled during the rest of the clock pulse.



JKFLIP-FLOP:-

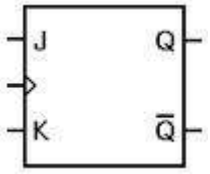
- The JK flip-flop can be constructed byusing basicSR latch and a clock. In this case the outputs Q and Q'arereturnedbackandconnectedto theinputsofNANDgates.
- This simpleJKflip Flop isthe most widely usedofalltheflip-flopdesigns andis considered to bea universal flip-flop circuit.
- The sequential operation of the JK flip flop is exactly the same as for the previous SR flip-flop with the same "Set"and"Reset"inputs.
- The difference this time is that the "JK flipflop" has no invalid orforbidden input states of the SR Latch even whenSandRarebothatlogic"1".

(ThebelowdiagramshowsthecircuitdiagramofaJKflip-flop)



- The JK flip flop is basically a gated SR Flip-flop with the addition of a clock input circuitry that prevents the illegal or invalid outputcondition that can occur when both inputs S andR are equal tologiclevel "1".
- Due to this additional clocked input, a JK flip-flop has four possible input combinations, "logic 1","logic 0", "no change" and "toggle".

- The symbolfora JK flip flop is similar to thatof an SR bistable latch except the clockinput.



(TheabovediagramshowsthesymbolofaJKflip-flop.)

- Boththe S and the R inputs oftheSR bi-stable have now beenreplaced by twoinputs calledthe Jand K inputs, respectively after its inventor Jack and Kilby. Then this equates to: J = S and K =R.
- Thetwo 2-input NAND gates ofthegatedSR bi-stable havenow beenreplaced by two3-inputNAND gates with the third input of each gate connected to the outputs at Q and Q'.
- ThiscrosscouplingoftheSR flip-flop allows thepreviouslyinvalidconditionofS="1"andR="1"state tobeusedtoproducea"toggleaction"asthetwoinputsarenowinterlocked.
- If the circuit is now "SET" the J input is inhibited by the "0" status of Q' through the lower NAND gate.If the circuit is "RESET" the K input is inhibited by the "0" status of Q through the upper NAND gate. As Q and Q' are always different we can use them to control the input.

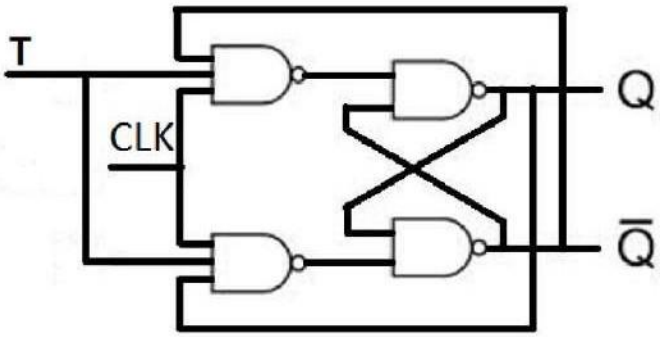
(Truth tableforJKflip-flop)

Input		Output		Comment
J	K	Q	Q_{next}	
0	0	0	0	Nochange
0	0	1	1	
0	1	0	0	Reset
0	1	1	0	
1	0	0	1	Set
1	0	1	1	
1	1	0	1	Toggle
1	1	1	0	

- WhenbothinputsJandKareequal tologic"1",theJKflipfloptoggles.

TFLIP-FLOP:-

- Toggleflip-floporcommonlyknownasTflip-flop.
- Thisflip-flophasthesimilaroperationasthat oftheJKflip-flopwithboththeinputsJandKareshorted i.e.botharegiventhecommoninput.



- Hence itstruth tableis sameas that of JKflip-flopwhenJ=K= 0andJ=K=1.Soits truthtable isas follows.

T	Q	Q_{next}	Comment
0	0	0	Nochange
	1	1	
1	0	1	Toggles
	1	0	

CHARACTERISTICTABLE:-

- A characteristic table definesthe logicalpropertiesof aflip-flopbydescribing itsoperationin tabular form. Thenext state isdefined asafunctionof the inputsand thepresentstate. Q (t)
- refers to the present state and Q (t + 1) is thenext.
- Thus,Q (t)denotes the state of theflip-flop immediately before the clock edge, and Q(t +1)denotes the state
- that results from the clock transition.
- The characteristic table for the JK flip-flop shows that the next state is equal to the present state when inputs J and K are both equal to 0. This condition can be expressed as Q (t + 1) = Q (t), indicating that the clock produces no change of state.

CharacteristicTableOfJKFlip-Flop

J	K	Q(t+1)
0	0	Q(t) Nochange
0	1	0 Reset
1	0	1 Set
1	1	Q'(t)Complement

- When K = 1 and J = 0, the clock resets the flip-flop and Q(t + 1) = 0. With J = 1 and K = 0, theflip-flop sets and Q(t + 1) = 1. When both J and K are equal to 1, the next state changes to the complement of the present state, a transition that can be expressed as Q(t + 1) = Q'(t).
- The characteristic equation for JK flip-flop is representedas

$$Q(t+1)=JQ'+K'Q$$

CharacteristicTableofDFlip-Flop

D	Q(t+1)
0	0
1	1

- Thenext stateof aDflip-flopisdependentonlyontheDinput andisindependent of thepresentstate.
- This canbeexpressed as Q (t+ 1) = D. It means that the next-state value is equal tothe valueof D. Note that the D flip-flop does not have a "no-change" condition and its characteristic equation is written as Q(t+1)=D.

CharacteristicTableofTFlip-Flop

T	Q(t+1)
0	Q(t) Nochange
1	Q'(t) Complement

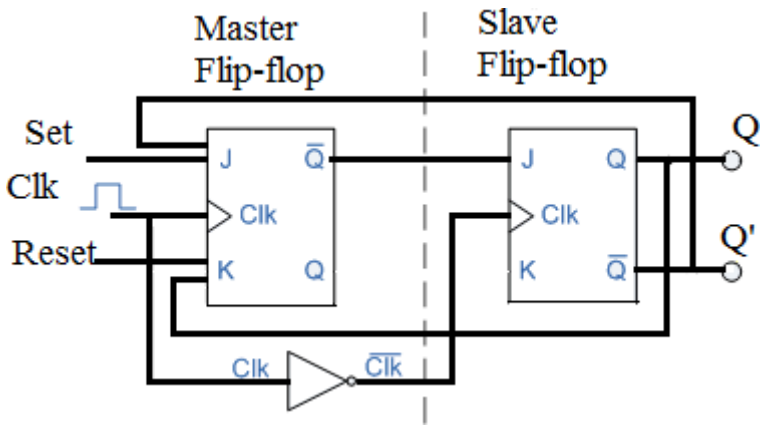
- The characteristic table of T flip-flop has only two conditions: When T = 0, the clock edge does not change the state; when T = 1, the clock edge complements the state of the flip-flop and thecharacteristic equation is

$$Q(t+1) = T \oplus Q = T'Q + TQ'$$

MASTER-SLAVEJKFLIP-FLOP:-

- The Master-SlaveFlip-Flopis basicallytwogatedSRflip-flops connected togetherinaseries configuration with the slave having an inverted clock pulse.
- Theoutputsfrom QandQ'fromthe "Slave" flip-flop arefedback to the inputsof the "Master" withthe outputs of the "Master" flip flop being connected to the two inputs of the "Slave" flipflop.
- Thisfeedback configurationfromthe slave'soutput tothe master's inputgivesthe characteristictoggle of the JK flip flop as shown below.

TheMaster-SlaveJKFlipFlop



- The input signals Jand Kareconnectedtothegated "master" SRflipflopwhich "locks" theinput condition while the clock (Clk) input is "HIGH" at logic level "1".
- As the clock inputof the"slave"flipflopis the inverse (complement)of the "master"clock input, the "slave" SR flip flop does not toggle.
- Theoutputsfrom the "master"flipflopareonly "seen"by thegated "slave"flipflop whenthe clock input goes "LOW" to logic level "0".
- When the clock is "LOW", the outputs from the "master"flip flop are latched and any additional changestoitsinputsareignored.
- The gated "slave" flip flop now responds to the state of its inputs passed over by the "master"section.
- Thenonthe"Low-to-High"transitionoftheclockpulsetheinputsofthe"master"flipfloparefed through to the gated inputs of the "slave" flip flop and on the "High-to-Low" transition the same inputsare reflected on the output of the "slave" making this type of flip flop edge orpulse-triggered.
- Then, the circuit accepts input data when the clock signal is "HIGH", and passes the data to theoutput on the falling-edge of the clock signal.
- In other words, the Master-Slave JK Flip flop is a "Synchronous" device as it only passes data with the timing of the clock signal.

FLIP-FLOPCONVERSIONS:-

SRFlip Flopto JKFlip Flop

For thisJ andK will begivenas external inputs toS andR. Asshownin thelogic diagrambelow,Sand R will be the outputs of the combinational circuit.

The truth tables for the flip flopconversion are given below. The present stateis represented by Qp and Qp+1 is the next state to be obtained when the J and K inputs are applied.

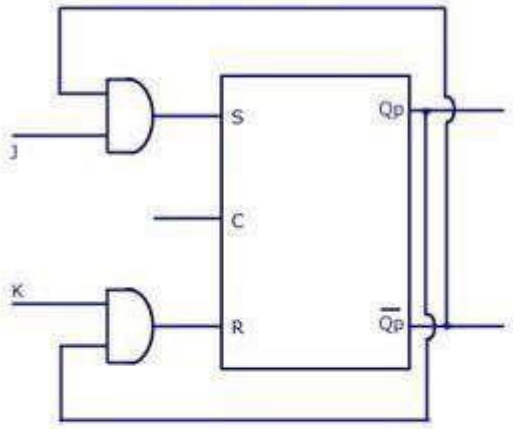
For two inputs J and K, there will be eight possible combinations. For each combination of J, K and Qp, the corresponding Qp+1 statesarefound.Qp+1 simply suggests thefuture values tobeobtained bytheJKflipflop after the value of Qp. The table is then completed by writing the values of S and R required to get each Qp+1 from the corresponding Qp. That is, the values of S and R that are required to change the state of the flip flop from Qp to Qp+1 are written.

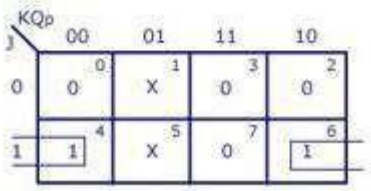
S-R Flip Flop to J-K Flip Flop

Conversion Table

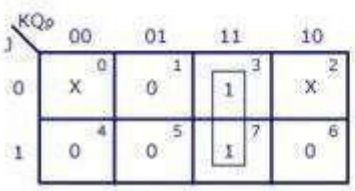
J-K Inputs		Outputs		S-R Inputs	
J	K	Qp	Qp+1	S	R
0	0	0	0	0	X
0	0	1	1	X	0
0	1	0	0	0	X
0	1	1	0	0	1
1	0	0	1	1	0
1	0	1	1	X	0
1	1	0	1	1	0
1	1	1	0	0	1

Logic Diagram





$S = \overline{J}Q_p$



$R = KQ_p$

K-Map

JK FlipFloptoSRFlipFlop

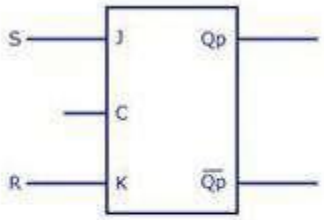
- This will be the reverse process ofthe above explained conversion. S and R will be the externalinputsto J and K. J and K will be the outputs of the combinational circuit. Thus, the values of J and K have to be obtained in terms of S, R and Qp.
- AconversiontableistobewrittenusingS,R,Qp,Qp+1, JandK.
- For two inputs, S and R, eight combinations are made. For each combination, the corresponding Qp+1 outputs are found out.
- Theoutputs forthecombinations ofS=1andR=1arenotpermittedfor anSR flipflop.Thustheoutputs are consideredinvalidandtheJandKvaluesaretakenas“don’tcares”.

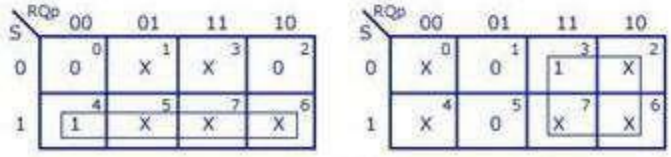
J-K Flip Flop to S-R Flip Flop

Conversion Table

S-R Inputs S	R	Outputs Qp	Qp+1	J-K Inputs J	K
0	0	0	0	0	X
0	0	1	1	X	0
0	1	0	0	0	X
0	1	1	0	X	1
1	0	0	1	1	X
1	0	1	1	X	0
1	1	Invalid		Dont care	
1	1	Invalid		Dont care	

Logic Diagram





SRFlip Flop toDFlip Flop

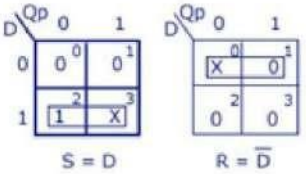
- S andRaretheactualinputsoftheflipflopandDis theexternalinputoftheflipflop.
- Thefour combinations, thelogic diagram,conversion table,and theK-mapfor Sand R in terms of D and Qp are shown below.

S-R Flip Flop to D Flip Flop

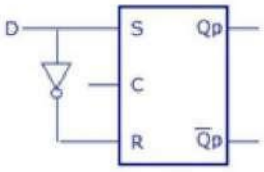
Conversion Table

D Input	Outputs Qp	Qp+1	S-R Inputs S	R
0	0	0	0	X
0	1	0	0	1
1	0	1	1	0
1	1	1	X	0

K-maps



Logic Diagram



DFlip Flopto SRFlip Flop

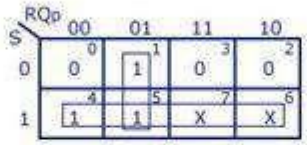
- Dis the actualinput of theflipflopandS and Rare theexternalinputs. Eightpossible combinationsare achieved from the external inputs S, R and Qp.
- But, since the combination of S=1 and R=1 are invalid, the valuesof Qp+1 and Dare considered as “don’tcares”.
- The logicdiagram showing theconversionfrom Dto SR, andtheK-mapfor Din terms of S, Rand Qp are shown below.

D Flip Flop to S-R Flip Flop

Conversion Table

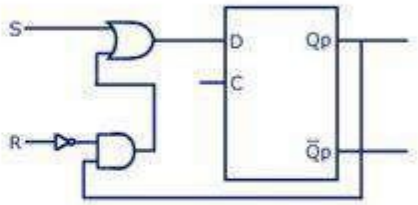
S-R Inputs S	R	Outputs Qp	Qp+1	D Input
0	0	0	0	0
0	0	1	1	1
0	1	0	0	0
0	1	1	0	0
1	0	0	1	1
1	0	1	1	1
1	1	Invalid		Don't care
1	1	Invalid		Don't care

K-map



$D = S + \bar{R}Q_n$

Logic Diagram



JK FlipFlopto TFlipFlop:-

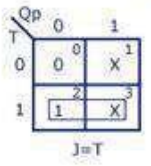
- J andK are theactualinputsof theflipflopand Tis takenasthe externalinputforconversion
- FourcombinationsareproducedwithTandQp. JandKareexpressedintermsofTandQp.
  - Theconversiontable,K-maps,andthelogicdiagramaregivenbelow.

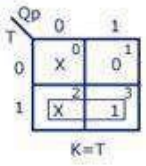
J-K Flip Flop to T Flip Flop

Conversion Table

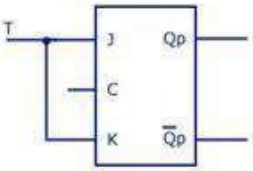
T Input	Outputs Qp	Qp+1	J-K Inputs J	K
0	0	0	0	X
0	1	1	X	0
1	0	1	1	X
1	1	0	X	1

K-maps





Logic Diagram



DFlipFloptoJKFlipFlop:-

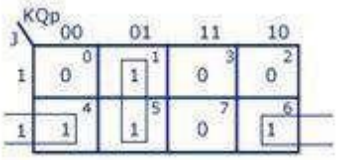
- In thisconversion, Distheactualinputto theflipflopandJandKare theexternalinputs.
- J, Kand Qp makeeightpossiblecombinations,asshown inthe conversion table below.Dis expressed in terms of J, K and Qp.
- The conversion table, theK-mapfor D intermsof J, Kand Qpand thelogicdiagram showingthe conversion from D to JK are given in the figurebelow.

D Flip Flop to J-K Flip Flop

Conversion Table

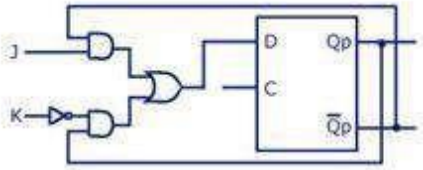
J-K Input J	K	Outputs Qp	Qp+1	D Input
0	0	0	0	0
0	0	1	1	1
0	1	0	0	0
0	1	1	0	0
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	0	0

K-map



$D = J\overline{Q}p + \overline{K}Qp$

Logic Diagram



JK FlipFlopto DFlipFlop:-

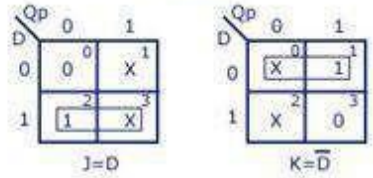
- D is the external inputand Jand Karetheactual inputsof theflipflop. Dand Qp makefour combinations. J and K are expressed in terms of D and Qp.
- Thefourcombinationconversiontable,theK-mapsforJandKintermsofDandQp.

J-K Flip Flop to D Flip Flop

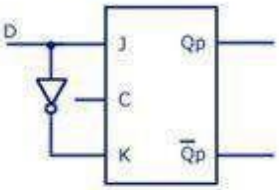
Conversion Table

D Input	Outputs Qp	Qp+1	J-K Inputs J	K
0	0	0	0	X
0	1	0	X	1
1	0	1	1	X
1	1	0	X	0

K-maps



Logic Diagram



COMBINATIONALLOGIC CIRCUIT

- A combinationalcircuit consistsof logicgateswhose outputsat anytime are determinedfromonlythe present combination of inputs.
- A combinationalcircuit performsan operationthatcanbespecified logicallybya setof Boolean functions.
- It consists of an interconnection of logic gates. Combinational logic gates react to the values ofthe signals at their inputs and produce the value of the output signal, transforming binary information from the given input data to a required output data.
- Ablockdiagramofacombinationalcircuitisshowninthebelowfigure.
- The n input binary variables come from an external source; the m output variables are produced by the internal combinational logic circuit and go to an external destination.
- Eachinput and output variable existsphysically as an analog signal whose valuesare interpreted tobe a binary signal that represents logic 1and logic 0.

BINARYADDER-SUBTRACTOR**:-**

- Digital computersperformavarietyof information-processingtasks.Amongthefunctionsencountered are the various arithmetic operations.
- The mostbasicarithmeticoperation istheadditionof twobinarydigits. Thissimple additionconsistsof four possible elementary operations: 0 + 0 = 0, 0 + 1 = 1, 1 + 0 = 1, and 1 + 1 =10.
- Thefirstthreeoperations producea sum ofone digit,butwhenbothaugend andaddendbits areequal to 1; the binary sum consists of two digits. The higher significant bit of this result is called acarry.
- Whentheaugendandaddendnumbers contain more significantdigits, thecarryobtainedfrom the addition of two bits is added to the next higher order pair of significant bits.
- Acombinationalcircuitthatperformstheadditionoftwo bitsiscalledahalfadder.
- Onethatperforms theadditionof threebits (two significantbitsandaprevious carry) is a fulladder.The namesof the circuits stemfrom the fact that two half adders can be employed to implement a fulladder.

HALFADDER:-

- Thiscircuitneedstwobinaryinputsand twobinaryoutputs.
- The input variables designate the augend and addend bits; the output variables produce the sumand carry.Symbols xandy are assignedtothetwo inputsandS (for sum)and C (for carry) to theoutputs. The
- truth table for the half adder is listed in the below table.
- The Coutput is1only whenboth inputsare1. The Soutput represents theleast significantbitof the sum.
- The simplified Booleanfunctionsfor the two outputs can be obtained directly from the truthtable.

x	y	D	B
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

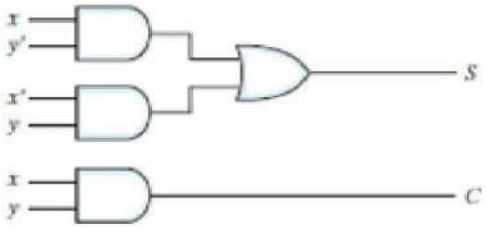
Truth Table

- The simplified sum-of-products expressions are

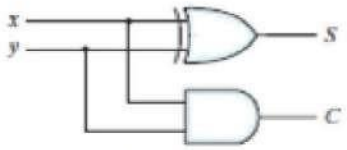
S=x'y+xy'

C=xy

- The logicdiagramof thehalf adderimplemented in sumof products is shown inthebelowfigure.It can be also implemented with an exclusive-OR and an AND gate.



(a) $S = xy' + x'y$
$C = xy$



(b) $S = x \oplus y$
$C = xy$

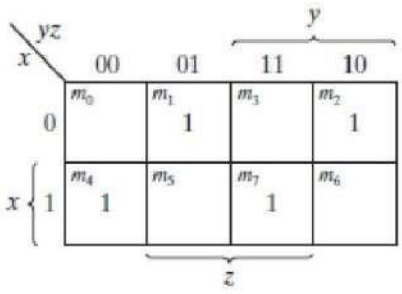
FULLADDER:-

- Afulladderisa combinationalcircuitthatformsthe arithmeticsumofthree bits.
- It consists of three inputs and two outputs. Two of the input variables, denoted by x and y ,represent the two significant bits to be added. The third input, z , represents the carry from the previous lower significant position.

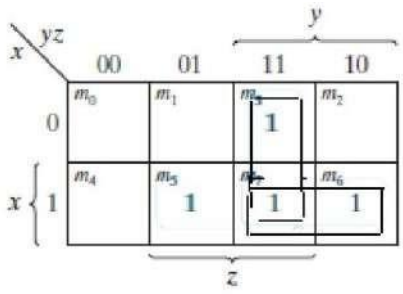
x	y	z	C	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Truth Table

- Two outputs are necessary because the arithmeticsum of three binary digits ranges in value from 0to3, and binary representation of 2 or 3 needs two bits. The two outputs are designated by the symbols S for sum and C for carry.



(a) $S = x'y'z + x'yz' + xy'z' + xyz$



(b) $C = xy + xz + yz$

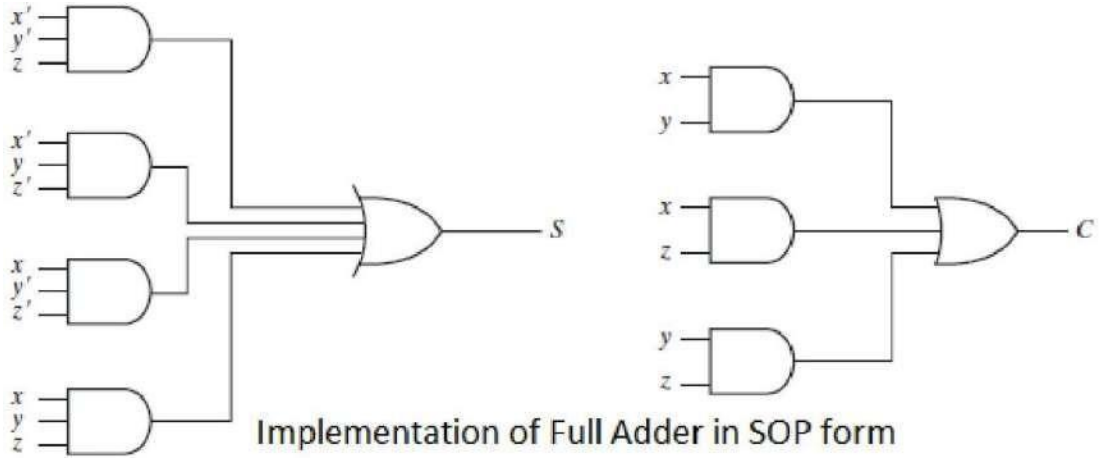
K-Map for full adder

- The binary variable S gives the value of the least significant bit of the sum. The binary variable Cgives the output carry formed by adding the input carry and the bits of thewords.
- The eight rows under the input variablesdesignate all possible combinations of the three variables. The output variables are determined from the arithmetic sum of the input bits. When all input bits are 0, the output is 0.
- The S output is equal to 1 when only one input is equal to 1 or when all three inputs are equal to 1.The C output has a carry of 1 if two or three inputs are equal to 1.
- Thesimplifiedexpressionsare

S=x'y'z+x'yz' +xy'z' +xyz

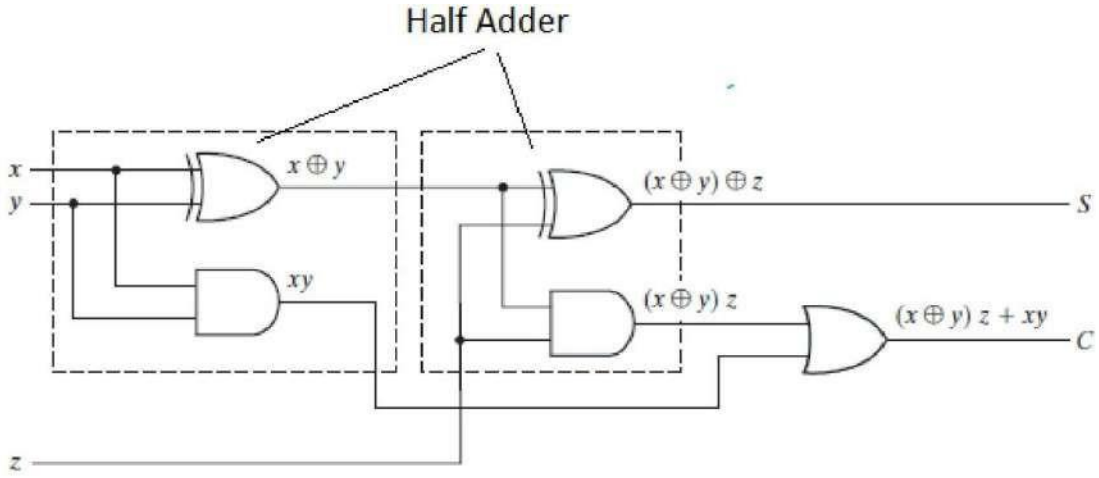
C=xy+xz+yz

- The logic diagramfor the full adder implemented in sum-of-products form is shown infigure.



Implementation of Full Adder in SOP form

- It canalsobeimplementedwithtwohalfaddersandoneORgateasshowninthefigure.



Implementation of Full Adder using Two Half Adders and an OR gate

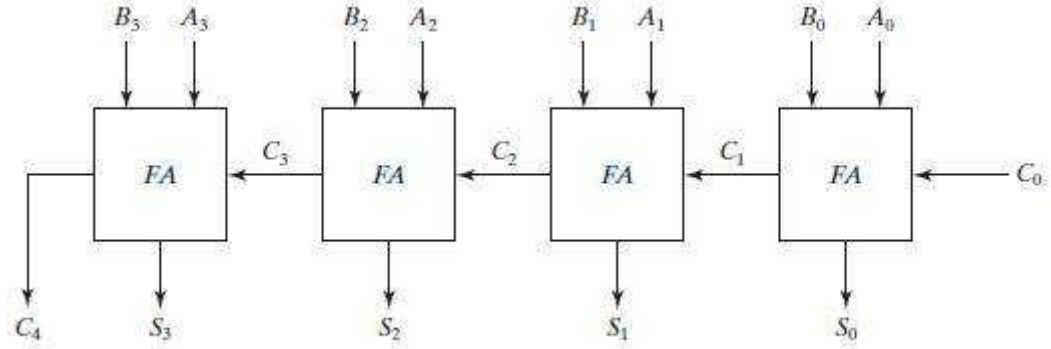
- Afulladderisa combinationalcircuitthatformsthe arithmeticsumofthreebits.

BINARYADDER:-

- Abinaryadderisadigitalcircuitthatproducesthearithmeticsumoftwobinarynumbers.
- It canbeconstructed withfulladdersconnected incascade, with theoutput carryfromeachfull adder connected to the input carry of the next full adder in thechain.
- Additionof n-bit numbers requiresa chainof nfulladdersor achainof one-half adder andn-1full adders. In the former case, the input carry to the least significant position is fixed at0.
- The interconnectionoffourfull-adder (FA) circuitstoprovideafour-bitbinary ripple carry adder is shown in the figure.
- Theaugendbitsof Aand theaddendbitsof B aredesignatedby subscriptnumbersfrom right to left, with subscript 0 denoting the least significant bit.
- The carriesare connected ina chain through thefulladders. The input carrytotheadderisC0,andit ripples through the full adders to the output carry C4. The S outputs generate the required sum bits.
- Ann -bitadder requiresnfull adders, witheach output carry connectedtothe input carry of thenext higher order full adder.
- Consider thetwobinary numbers A=1011andB= 0011. Their sum S= 1110 isformed with thefour- bit adder as follows:

Subscript i:	3	2	1	0	
Input carry	0	1	1	0	C_i
Augend	1	0	1	1	A_i
Addend	0	0	1	1	B_i
Sum	1	1	1	0	S_i
Output carry	0	0	1	1	C_{i+1}

- The bits are added with full adders, starting from the least significant position (subscript 0), to form the sum bit and carry bit. The input carry C_0 in the least significant position must be 0.
- The value of C_{i+1} in a given significant position is the output carry of the full adder. This value is transferred into the input carry of the full adder that adds the bits one higher significant position to the left.
- The sum bits are thus generated starting from the rightmost position and are available as soon as the corresponding previous carry bit is generated. All the carries must be generated for the correct sum bits to appear at the outputs.



Four Bit Binary Adder

HALF SUBTRACTOR:-

- This circuit needs two binary inputs and two binary outputs.
- Symbols x and y are assigned to the two inputs and D (for difference) and B (for borrow) to the outputs. The
- truth table for the half subtractor is listed in the below table.

x	y	D	B
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

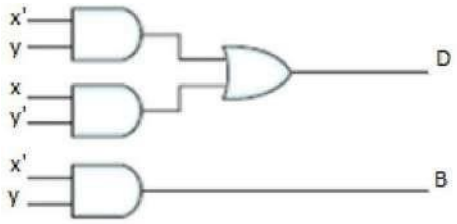
Truth Table

- The B output is 1 only when the inputs are 0 and 1. The D output represents the least significant bit of the subtraction.
- The subtraction operation is done by using the following rules as

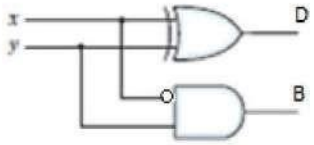
0-0=0;
0-1=1 with borrow 1;
1-0=1;
1-1=0.

- The simplified Boolean functions for the two outputs can be obtained directly from the truth table. The simplified sum-of-products expressions are

D=x'y+xy' and B=x'y



D = x'y+xy'
B = x'y



D = x ⊕ y
B=x'y

- Thelogicdiagramofthehalfadderimplementedinsumofproductsisshowninthefigure.Itcanbe also implemented with an exclusive-OR and an AND gate with one invertedinput.

FULLSUBTRACTOR:-

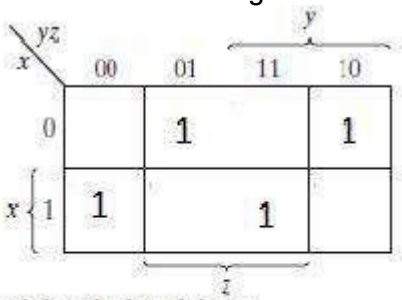
- Afullsubtractorisacombinationalcircuitthatformsthearithmeticsubtractionoperationofthreebits.
- It consists of three inputs and two outputs. Two of the input variables, denoted by x and y ,represent the two significant bits to be subtracted. The third input, z , is subtracted from the result 0f the first subtraction.

x	y	z	D	B
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

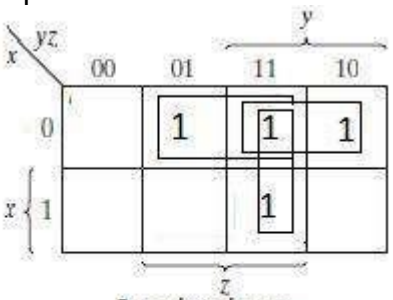
Truth Table

- Twooutputs arenecessary becausethe arithmeticsubtraction ofthree binarydigitsrangesinvalue from 0 to 3, and binary representation of 2 or 3 needs two bits. The two outputs are designated by the symbols D for difference and B for borrow.

- The binary variable D gives the value of the least significant bit of the difference. The binary variable B gives the output borrow formed during the subtraction process.



D = x'y'z+x'yz'+xy'z'+xyz



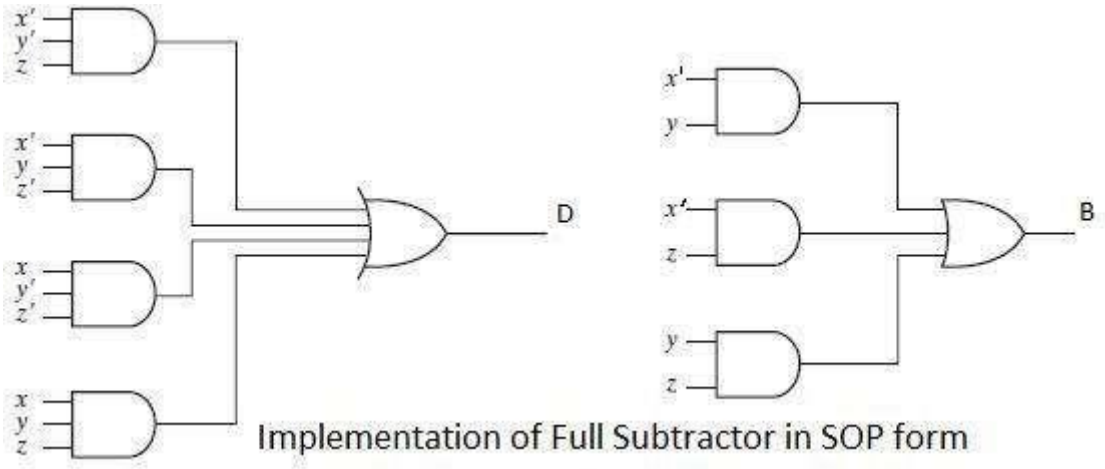
B = x'z+x'y+yz

K-Map for full Subtractor

- Theeight rowsunderthe input variablesdesignate allpossible combinationsof the three variables.The output variables are determined from the arithmetic subtraction of the input bits.
- Thedifference Dbecomes 1whenanyoneof the input is1orall three inputsareequalto1and the borrow B is 1 when the input combination is (0 0 1) or (0 1 0) or (0 1 1) or (1 1 1).
- Thesimplifiedexpressionsare

D= x'y'z+ x'yz'+ xy'z'+xyz B
= x'z + x'y + yz

- Thelogicdiagramforthefulladderimplemented insum-of-productsformisshowninfigure.



Implementation of Full Subtractor in SOP form

MAGNITUDECOMPARATOR:-

- A magnitude comparatorisacombinationalcircuitthat compares twonumbersA andBanddetermines their relative magnitudes.
- Thefollowingdescriptionisabouta2-bitmagnitudecomparatorcircuit.
- Theoutcome of thecomparison is specifiedbythreebinary variablesthat indicate whetherA<B,A= B, or A > B.
- Considertwo numbers,A andB,withtwodigitseach. Nowwritingthe coefficientsof the numbers in descending order of significance:

$$A=A_1A_0$$
$$B=B_1B_0$$

- Thetwonumbersareequalif allpairsof significant digits are equali.e. if andonlyif A1 =B1,and A0= B0.
- Whenthenumbersarebinary, thedigitsareeither1or 0,andtheequalityof eachpairofbits canbe expressed logically with an exclusive-NOR function as

$x1=A_1B_1+A_1'B_1'$

$$Andx0=A_0B_0+A_0'B_0'$$

- Theequalityof the twonumbersA andB isdisplayed in acombinationalcircuit byanoutput binary variable that we designate by the symbol (A = B).
- This binary variable isequalto 1 if the input numbers, A and B , are equal, andisequalto 0 otherwise.
- Forequality to exist,allxivariablesmustbe equalto1,a condition that dictatesanANDoperationof all variables:

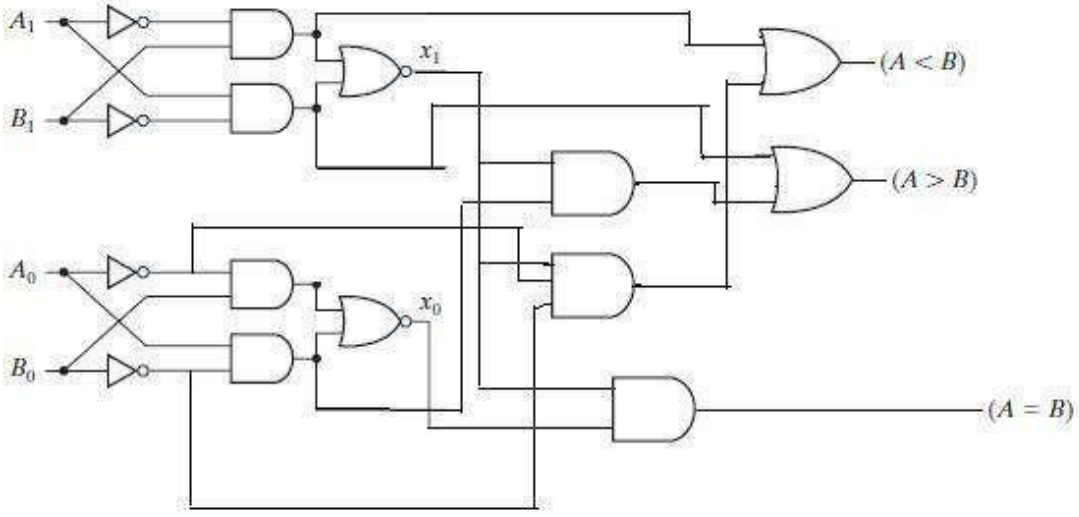
$$(A=B)=x_1x_0$$

- Thebinaryvariable(A=B)isequalto1onlyif allpairsofdigitsofthetwonumbersareequal.
- To determine whether A is greater or less than B, we inspect the relative magnitudes of pairs of significant digits, starting from the most significant position. If the two digits of a pair are equal, we compare the next lower significantpair of digits. If the corresponding digit of A is 1and thatof B is 0,we concludethat A> B. Ifthecorresponding digit ofA is0 andthatofB is1,we have A< B. The sequential comparison can be expressed logically by the two Boolean functions

$$(A > B) =$$
$$A_1B_1'+x_1A_0B'_0(A<B)=A_1'$$
$$B_1+x_1A_0'B_0'$$

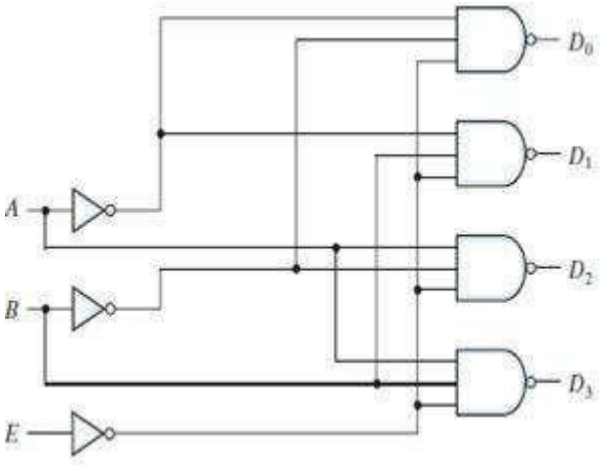
A_1	A_0	B_1	B_0	A>B	A<B	A=B
0	0	0	0	0	0	1
0	0	0	1	0	1	0
0	0	1	0	0	1	0
0	0	1	1	0	1	0
0	1	0	0	1	0	0
0	1	0	1	0	0	1
0	1	1	0	0	1	0
0	1	1	1	0	1	0
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	0	1
1	0	1	1	0	1	0
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	0	1

Truth Table



Logic Diagram of 2-bit Magnitude Comparator

DECODER:-



E	A	B	D_0	D_1	D_2	D_3
1	X	X	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	1

- A decoderisa combinationalcircuitthat convertsbinaryinformationfromn input linestoa maximum of 2^n unique output lines.
- If then -bitcoded informationhasunused combinations,the decodermayhavefewerthan2noutputs. The
- decoders presented here are called n -to- m -line decoders, where m ...2n.
- Theirpurposeistogenerate the2n (orfewer)mintermsof n inputvariables.
- Each combination ofinputswillasserta uniqueoutput.Thenamedecoderisalsoused inconjunction with other code converters, such as a BCD-to-seven-segment decoder.
- Consider thethree-to-eight-linedecoder circuit of threeinputsaredecoded intoeightoutputs,each representing one of the minterms of the three input variables.
- Thethree inverters providethe complement of theinputs,andeachoneof theeightANDgates generates one of the minterms.
- The input variables representabinary number,and theoutputs represent theeightdigitsof anumberin the octal number system.
- However,athree-to-eight-linedecoder canbe usedfor decoding any three-bit codetoprovideeight outputs, one for each element of the code.
- Atwo-to-four-linedecoderwithanenableinputconstructedwithNANDgatesisshowninFig.
- The circuit operates with complemented outputs and a complement enable input. The decoder is enabled when E is equal to 0 (i.e., active-low enable). As indicated by the truth table, only one output can be equal to 0 at any given time; all other outputs are equal to 1.
- The output whose value is equal to 0 represents the minterm selected by inputs A and B.
- The circuit isdisabledwhenE isequalto1, regardlessof the valuesof theothertwoinputs.
- Whenthe circuitis disabled,noneof theoutputsare equalto0andnoneof the mintermsareselected. In
- general, a decoder may operate with complemented or un-complemented outputs.
- Theenableinputmaybeactivatedwitha0orwitha1signal.
- Somedecodershave twoormoreenable inputs thatmustsatisfy agiven logiccondition in orderto enable the circuit.
- A decoder with enable input can function as a demultiplexer– a circuit that receives information from a single line and directs it to one of 2n possible output lines.
- Theselectionofaspecificoutputiscontrolledbythebitcombinationofnselectionlines.
- Thedecoder of Fig. canfunctionas aone-to-four-linedemultiplexer whenE istakenasadata input line and A and B are taken as the selection inputs.
- The single inputvariableEhasapathtoallfouroutputs,butthe inputinformation isdirectedtoonly one of the output lines, as specified by the binary combination of the two selection lines A and B. This feature
- can be verified from the truth table of the circuit.
- Forexample, if the selection linesAB= 10,outputD_2willbe thesameasthe input valueE,whileall other outputs are maintained at 1.
- Sincedecoderanddemultiplexeroperationsareobtainedfromthe same circuit,a decoderwithan enableinputisreferredtoasadecoder–demultiplexer.
- Aapplicationofthisdecoderisbinary-to-octalconversion.

ENCODER:-

- Anencoderisa digitalcircuit that performsthe inverseoperationof adecoder. An
- encoder has 2n (or fewer) input lines and n output lines.
- Theoutputlines,asanaggregate,generatethebinarycodecorrespondingtotheinputvalue.

Inputs								Outputs		
D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	x	y	z
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

- TheaboveEncoderhaseightinputs(onefor eachof the octaldigits)and threeoutputs that generate the corresponding binary number.
- Itis assumedthatonlyoneinputhasavalueof1atanygiven time.
- TheencodercanbeimplementedwithORgateswhose inputs are determined directlyfromthe truth table. Outputzisequalto1whentheinputoctal digitis1,3,5,or7.
- Output y is1 for octaldigits2, 3, 6, or7, and output x is1 fordigits4, 5, 6, or7.
- These conditions can be expressed by the following Boolean outputfunctions:
- $z=D_1+D_3+D_5+D_7$

  $y = D_2+D_3+D_6+$

  $D_7x=D_4+D_5+D_6+D_7$
- TheencodercanbeimplementedwiththreeORgates.
- Theencoderdefined abovehasthelimitationthatonlyoneinput canbeactiveatanygiventime.
- If twoinputsare active simultaneously,theoutputproducesanundefined combination.
- To resolvethisambiguity,encodercircuitsmustestablish an inputpriorityto ensure that onlyone input is encoded which is done in the Priority Encoder .

PRIORITYENCODER:-

- Aprioritыencoderisanencodercircuitthatincludesthepriorityfunction.
- Theoperationof thepriority encoder is such that if twoor moreinputsareequal to 1atthe sametime, the input having the highest priority will take precedence.

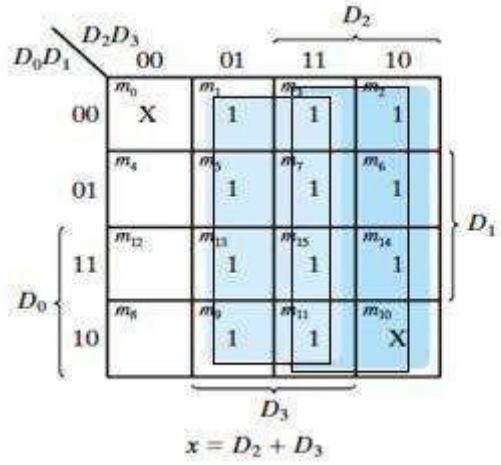
Inputs				Outputs		
D_0	D_1	D_2	D_3	x	y	V
0	0	0	0	X	X	0
1	0	0	0	0	0	1
X	1	0	0	0	1	1
X	X	1	0	1	0	1
X	X	X	1	1	1	1

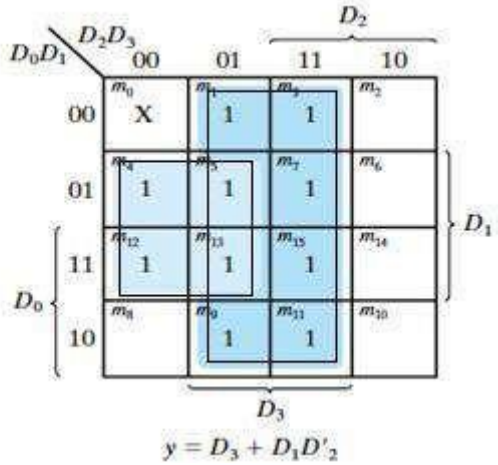
- In addition to thetwooutputs xandy ,the circuithas athird outputdesignatedbyV; this is avalidbit indicator that is set to 1 when one or moreinputsareequalto 1.

- If all inputs are 0, there is no valid input and V is equal to 0.
- The other two outputs are not inspected when V equals 0 and are specified as don't-care conditions.
- Here X's in output columns represent don't-care conditions, the X 's in the input columns are useful for representing a truth table in condensed form.

Inputs				Outputs		
D_0	D_1	D_2	D_3	x	y	V
0	0	0	0	X	X	0
1	0	0	0	0	0	1
X	1	0	0	0	1	1
X	X	1	0	1	0	1
X	X	X	1	1	1	1

- Higher the subscript number, the higher the priority of the input.
- Input D3 has the highest priority, so, regardless of the values of the other inputs, when this input is 1, the output for xy is 11 (binary 3).
- If D2=1, provided that D3 =0, regardless of the values of the other two lower priority inputs the output is 10.
- The output for D1 is generated only if higher priority inputs are 0, and so on down the priority levels.

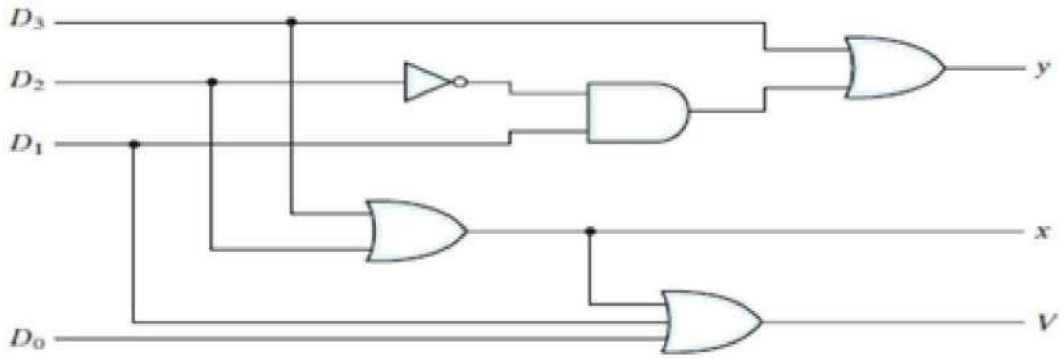




- The maps for simplifying outputs x and y are shown in above Fig.
- The minterms for the two functions are derived from its truth table.
- Although the table has only five rows, when each X in a row is replaced first by 0 and then by 1, we obtain all 16 possible input combinations.
- For example, the fourth row in the table, with inputs XX10, represents the four minterms 0010, 0110, 1010, and 1110. The simplified Boolean expressions for the priority encoder are obtained from the maps. The condition for output V is an OR function of all the input variables.
- The priority encoder is implemented according to the following Boolean functions: x =
- $D_2 + D_3$

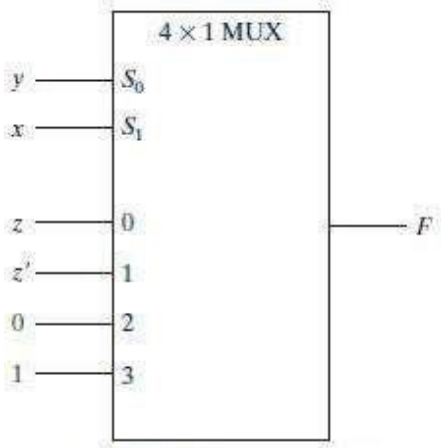
  $y = D_3 + D_1D'_2$

  $V = D_0 + D_1 + D_2 + D_3$

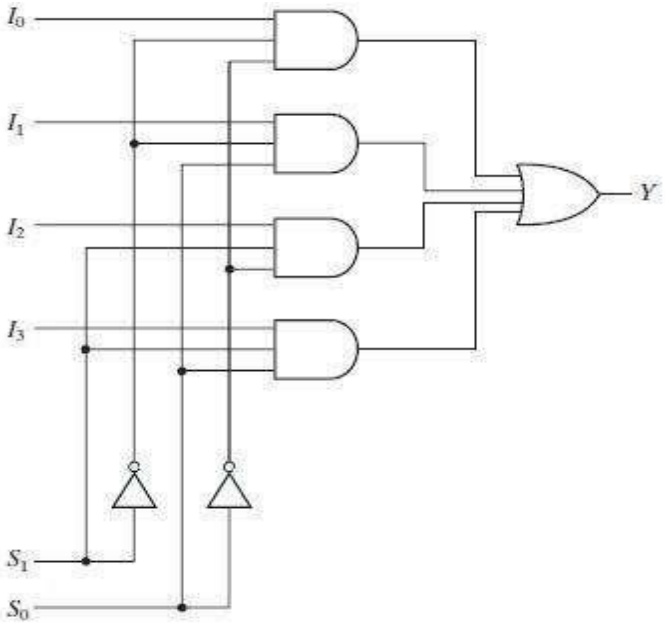


MULTIPLEXER:-

- A multiplexer is a combinational circuit that selects binary information from one of many input lines and directs it to a single output line.
- The selection of a particular input line is controlled by a set of selection lines.
- Normally, there are 2^n input lines and n selection lines whose bit combinations determine which input is selected.
- A four-to-one-line multiplexer is shown in the below figure. Each of the four inputs, I_0 through I_3, is applied to one input of an AND gate.
- Selection lines S_1 and S_0 are decoded to select a particular AND gate. The outputs of the AND gates are applied to a single OR gate that provides the one-line output.
- The function table lists the input that is passed to the output for each combination of the binary selection values.
- To demonstrate the operation of the circuit, consider the case when S_1S_0 = 10.
- The AND gate associated with input I_2 has two of its inputs equal to 1 and the third input connected to I_2.
- The other three AND gates have at least one input equal to 0, which makes their outputs equal to 0. The output of the OR gate is now equal to the value of I_2, providing a path from the selected input to the output.
- A multiplexer is also called a data selector, since it selects one of many inputs and steers the binary information to the output line.



(b) Multiplexer implementation



Logic diagram

S_1	S_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

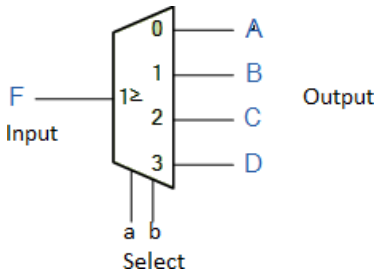
Truth table

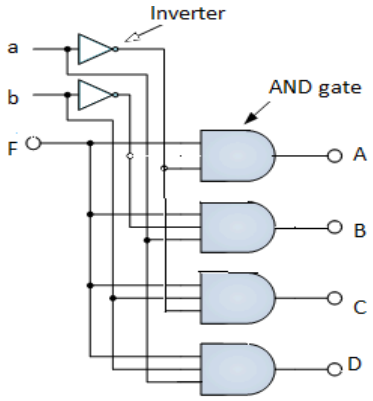
DEMULTIPLEXER:-

- The data distributor, known more commonly as a Demultiplexer or “Demux” for short, is the exact opposite of the Multiplexer.
- The demultiplexer takes one single input data line and then switches it to any one of a number of individual output lines one at a time. The demultiplexer converts a serial data signal at the input to a parallel data at its output lines as shown below.
- The Boolean expression for this 1-to-4 demultiplexerabove with outputs A to D and data select linesa,b is given as:

$$F = (ab)'A + a'bB + ab'C + abD$$

- The function ofthedemultiplexer is toswitchonecommondatainputlinetoany oneofthe 4output data lines A to D in our example above. As with the multiplexer the individual solid state switches are selected by the binary input address code on the output select pins “a” and “b” as shown.





Logic Diagram

- Unlike multiplexers which convert data froma single data line to multiple linesand demultiplexers which convert multiple lines to a single data line, there are devices available which convert data to and from multiple lines and in the next tutorial about combinational logic devices.
- Standard demultiplexerICpackages available arethe TTL 74LS138 1 to 8-output demultiplexer,the TTL 74LS139 Dual 1-to-4 output demultiplexer or the CMOS CD4514 1-to-16 outputdemultiplexer.

Output Select		Data output Selected
b	a	
0	0	A
0	1	B
1	0	C
1	1	D

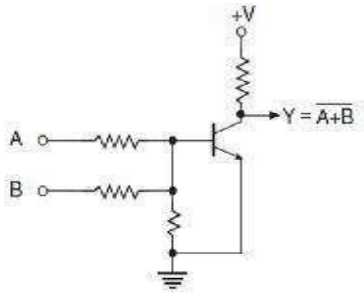
Truth Table

LOGIC FAMILIES

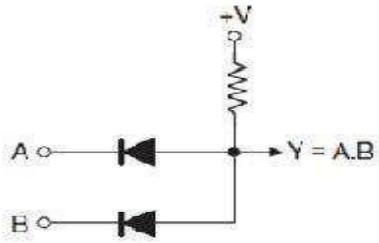
- A circuit configuration or approach used to produce a type of digital integrated circuit is called Logic Family.
- By using logic families we can generate different logic functions, when fabricated in the form of an IC with the same approach, or in other words belonging to the same logic family, will have identical electrical characteristics.
- These of digital ICs belonging to the same logic family are electrically compatible with each other.
- Some common Characteristics of the Same Logic Family include Supply voltage range, speed of response, power dissipation, input and output logic levels, current sourcing and sinking capability, fan-out, noise margin, etc.
- Choosing digital ICs from the same logic family guarantees that these ICs are compatible with respect to each other and that the system as a whole performs the intended logic function.

TYPES OF LOGIC FAMILY:-

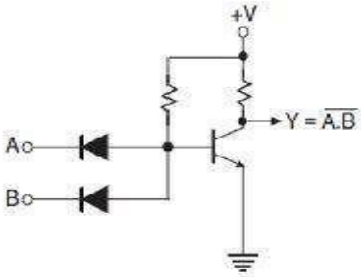
- The entire range of digital ICs is fabricated using either bipolar devices or MOS devices or a combination of the two.
- Bipolar families include:-
  - Diode logic (DL)
  - Resistor-Transistor logic (RTL)
  - Diode-transistor logic (DTL)
  - Transistor-Transistor logic (TTL)
  - Emitter Coupled Logic (ECL),
    - (also known as Current Mode Logic (CML))
  - Integrated Injection logic (I2L)
- The Bi-MOS logic family uses both bipolar and MOS devices.



Resistor -Transistor logic (RTL)



Diode Logic (DL)



Diode-Transistor logic (DTL)

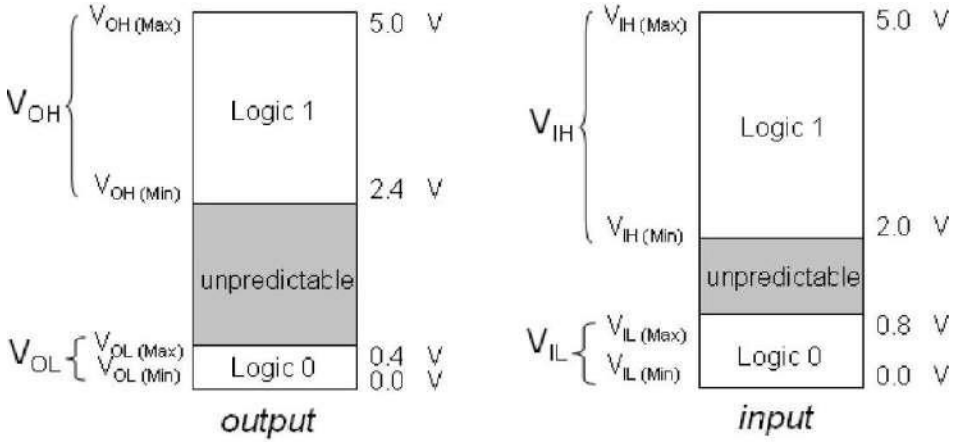
- Above are some example of DL, RTL and DTL.
- MOS families include:-
  - The PMOS family (using P-channel MOSFETs)
  - The NMOS family (using N-channel MOSFETs)
  - The CMOS family (using both N-and P-channel devices) SOME

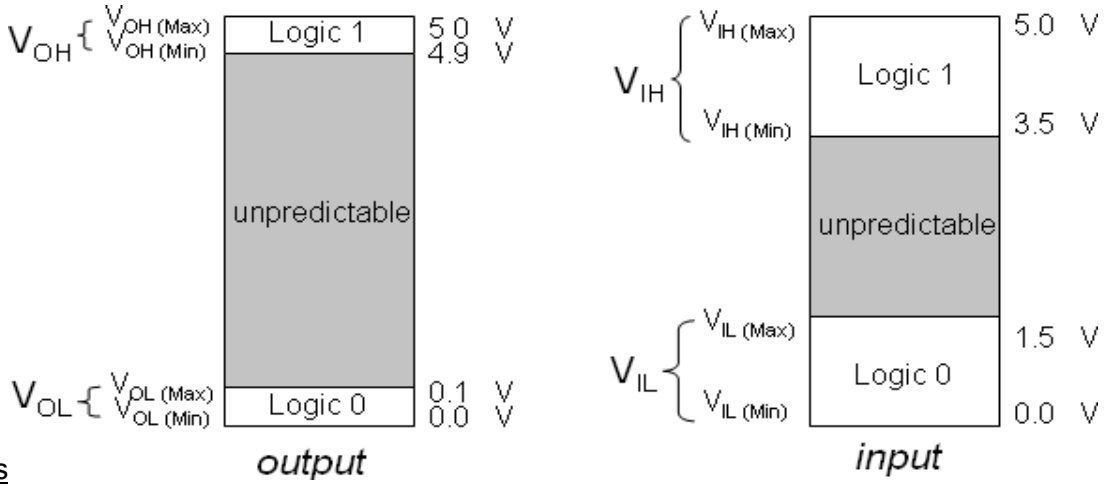
OPERATIONAL PROPERTIES OF LOGIC FAMILY:-

DC Supply Voltage:-

- The nominal value of the dc supply voltage for TTL (transisitor-transistor logic) and CMOS (complementary metal-oxide semiconductor) devices is +5V. Although ommitted from logic diagrams for simplicity, this voltage is connected to Vcc or VDD pin of an IC package and ground is connected to the GND pin.

TTLLogicLevels





CMOS LogicLevels

NoiseImmunity:-

- Noise is the unwanted voltage that is induced in electrical circuits and can present a threat to the poor operation of the circuit. In order not to be adversely effected bynoise, a logic circuit must have a certain amount of 'noise immunity'.
- This is the ability to tolerate a certain amount of unwanted voltage fluctuation on its inputs without changing its output state is called Noise Immunity.

NoiseMargin:-

- Ameasureofacircuit'snoiseimmunityiscalled'noisemargin'whichisexpressedinvolts.
- There are two values of noise margin specified for a given logic circuit: the HIGH (V_{NH}) and LOW (V_{NL}) noise margins.

Theseare definedbyfollowingequations:

$V_{NH}=V_{OH}(Min)-V_{IH}(Min)V_{NL}=V_{IL}(Max) - V_{OL}(Max)$

PowerDissipation:-

- A logic gate draws ICCH currentfromthe supplywhenthegate isin theHIGH output state,draws ICCL current from the supply in the LOW output state.
- Average power is

  PD=VCCICCwhereICC=(ICCH+ICCL)/2

PropagationDelaytime:-

- When asignal passes (propagates) through a logic circuit,it always experiences atime delayas shown below. A change in the output level always occurs a short time, called 'propagation delay time' , later than the change in the input level that caused it.

FanOutofGates:-

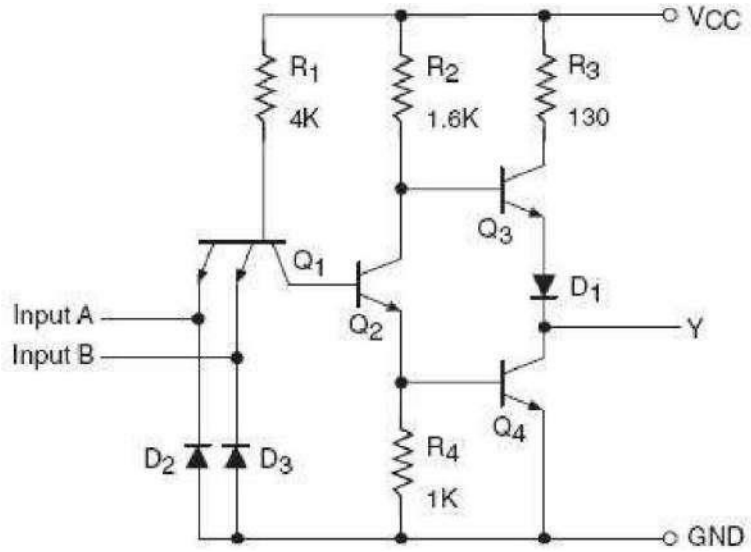
- When the output of a logic gate is connected to one or more inputs of other gates, a load on the driving gate is created. There is a limit to the number of load gates that a given gate can drive. This limit is called the 'Fan-Out' of the gate.

TRANSISTOR-TRANSISTORLOGIC:-

- InTransistor-TransistorlogicorjustTTL,logicgatesarebuiltonlyaroundtransistors.
- TTL wasdeveloped in 1965.Through theyears basic TTLhas been improved to meetperformance requirements. There are many versions or families of TTL.
- Forexample
  - Standard TTL
  - High SpeedTTL(twiceasfast,twiceasmuchpower)
  - LowPowerTTL (1/10 thespeed,1/10thepowerof “standard"TTL) Schhottky
  - TTL etc. (for high-frequency uses)
- All TTL logic families have three configurationsfor outputs
  1. Totempole output
  2. Opencollectoroutput
  3. Tristateoutput

Totempole output:-

- Additionofanactivepullup circuit in theoutputof a gate iscalledtotempole.
- To increase the switching speedof thegate whichislimiteddue to theparasiticcapacitanceat the output totem pole is used.
- The circuit of a totem-pole NAND gate is shown below, which has got three stages
  1. Input Stage
  2. PhaseSplitterStage
  3. Output Stage



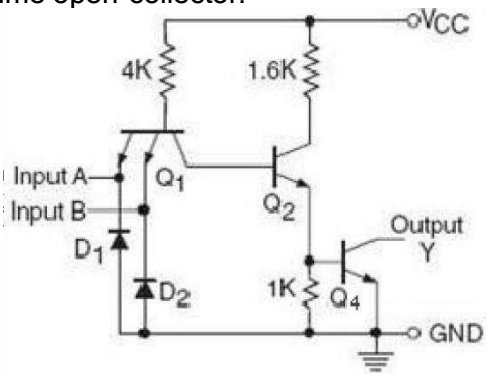
- Transistor Q1 is a two-emitter NPN transistor, which is equivalent two NPN transistors with their base and emitter terminals tied together.
- The two emitters are the two inputs ofthe NAND gate
  In TTL technology multiple emitter transistors are used for the input devices
  DiodesD2and D3 are protection diodesused to limit negative input voltages.
- When there is large negative voltage at input, the diode conducts and shorting it to the ground Q2 provides complementary voltages for the output transistors Q3 and Q4.
- The combination of Q3 and Q4 forms the output circuit often referred to as a totem polearrangement (Q4 is stacked on top of Q3). In such an arrangement, either Q3 or Q4 conducts at a time depending upon the logic status of the inputs
  Diode D1 ensuresthat Q4 willturnoff whenQ2 ison (HIGHinput)
  The output Y is taken from the top of Q3

AdvantagesofTotemPoleOutput:-

- The features of this arrangement are
  1. Lowpowerconsumption
  2. Fastswitching
  3. Lowoutputimpedance

OPEN COLLECTOR OUTPUT:-

- Figure below shows the circuit of a typical TTL gate with open-collector output Observe here that the circuit elements associated with Q3 in the totem-pole circuit are missing and the collector of Q4 is left open-circuited, hence the name open-collector.



- An open-collector output can present a logic LOWoutput. Since there is no internal path fromthe output Y to the supply voltage V_{CC}, the circuit cannot present a logic HIGH on itsown.

AdvantagesofOpenCollectorOutputs:-

- Open-collectoroutputscanbetieddirectlytogetherwhichresultsinthelogicalANDingoftheoutputs. Thusthe equivalentof anANDgatecanbeformedbysimplyconnectingtheoutputs.
- Increased current levels - Standard TTL gates with totem-pole outputs can only provide a HIGH current output of 0.4 mA and a LOW current of 1.6 mA. Many open-collector gates have increased current ratings.
- Different voltage levels - A wide variety of output HIGH voltages can be achieved using open-collector gates. This is useful in interfacing different logic families that have different voltage and current level requirements.

Disadvantageofopen-collectorgates:-

- They haveslow switchingspeed.Thisis because the value ofpull-upresistor isinkW,whichresultsin a relatively long time Constants

ComparisonofTotemPoleandOpenCollectorOutput:-

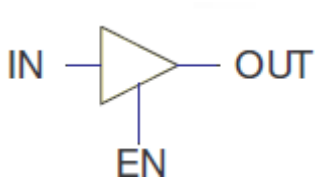
- The major advantage of using a totem-pole connection is that it offers low-output impedance in both the HIGH and LOW output states

Totem Pole	Open Collector
Output stage consists of pull-up transistor (Q3), diode resistor and pull-down transistor (Q4)	Output stage consists of only pull-down transistor
External pull-up resistor is not required	External pull-up resistor is required for proper operation of gate
Output of two gates cannot be tied together	Output of two gates can be tied together using wired AND technique
Operating speed is high	Operating speed is low

TRISTATE(THREE-STATE)LOGICOUPUT:-

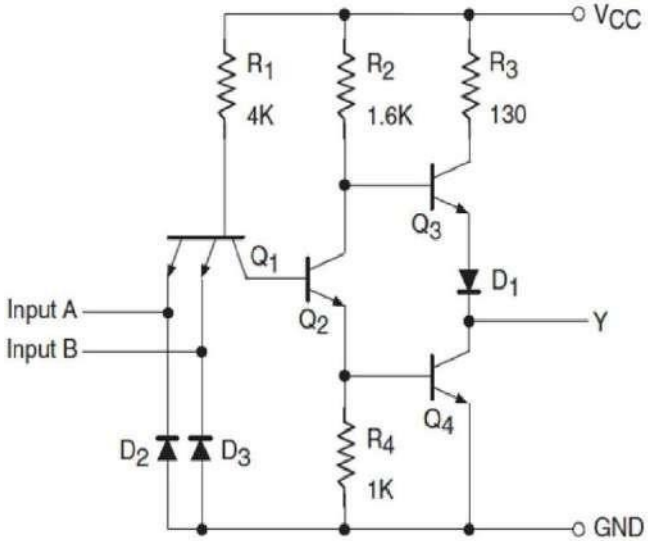
- Tristateoutput combinestheadvantagesof the totem-poleandopen collectorcircuits.
- Three output states are HIGH, LOW, and high impedance (Hi-Z).

EN	IN	OUT
0	X	HI-Z
1	0	0
1	1	1



- For the symbol and truth table, IN is the data input, and EN, the additional enable input for control.For EN = 0, regardless of the value on IN(denoted by X), the output value is Hi-Z. For EN = 1, the output value follows the input value.
- Data input, IN, can be inverted. Control input, EN, can be inverted by addition of "bubbles" to signals IN OUT EN.
- This requires two inputs: input and enable EN is to make output Hi-Z or followinput.

STANDARDTTLNANDGATE:



A	B	Q1	Q2	Q3	Q4	Y
L	L	sat	off	off	Off	H
L	H	sat	off	off	Off	H
H	L	sat	off	off	Off	H
H	H	ia m	sat	sat	On	L

CMOSTECHNOLOGY:-

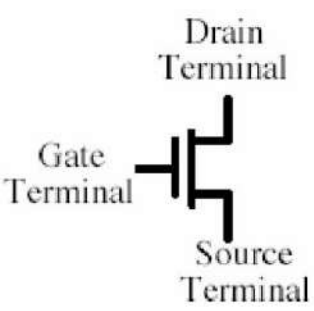
- MOS standsforMetalOxide Semiconductorand thistechnologyusesFETs.
- MOS can be classified into three sub-families:
  PMOS(P-channel)
  NMOS(N-channel)
  CMOS(ComplementaryMOS,mostcommon)
- The followingsimplifiedsymbols are used torepresentMOSFET transistorsinmost CMOS.Thegate of a MOS transistor controls the flow of the current between the drain and the source. The MOS transistor can be viewed as a simple ON/OFF switch.

AdvantagesofMOSDigitalICs:-

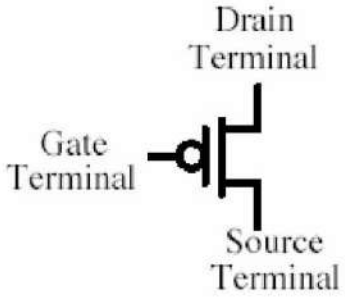
- Theyaresimpleandinexpensivetofabricate.
- CanbeusedforHigherintegrationandconsume littlepower.

Disadvantagesof MOSDigitalICs:-

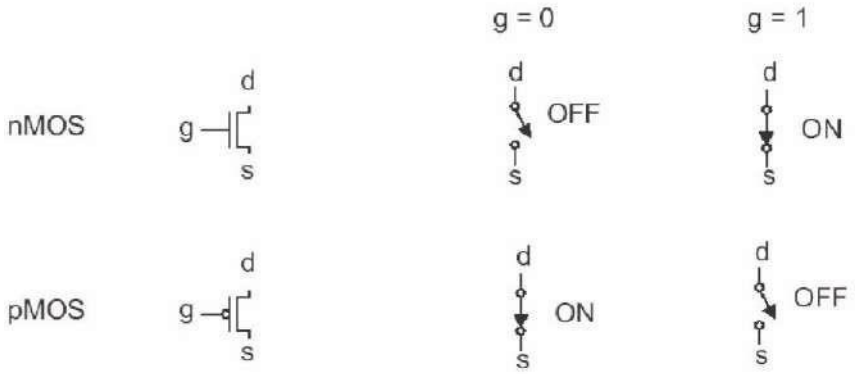
- ThereispossibilityforStatic-electricitydamage.
- They are slower than TTL.



N-Channel
MOSFET Symbol



P-Channel
MOSFET Symbol

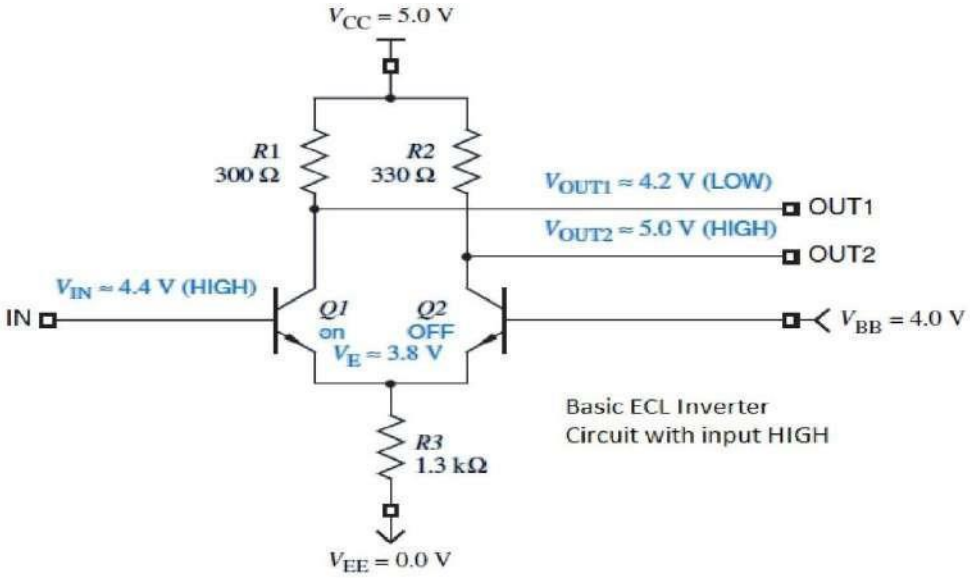


ECL:EMITTER-COUPLEDLOGIC:-

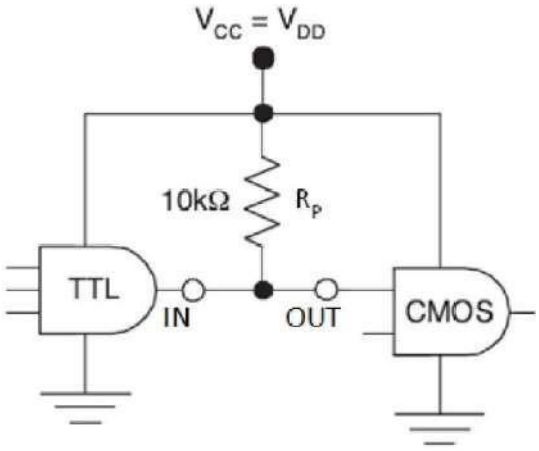
- The key to reduce propagation delay in a bipolar logic family is to prevent a gate's transistors from saturating. It is possible to prevent saturation by using a radically different circuit structure, called current-mode logic (CML) or emitter-coupled logic (ECL).
- Unlike the other logic families in this chapter, ECL does not produce a large voltage swing between the LOWand HIGHlevels butit has a smallvoltageswing, less than a volt, and it internallyswitches current between two possible paths, depending on the output state.

BasicECLCircuit

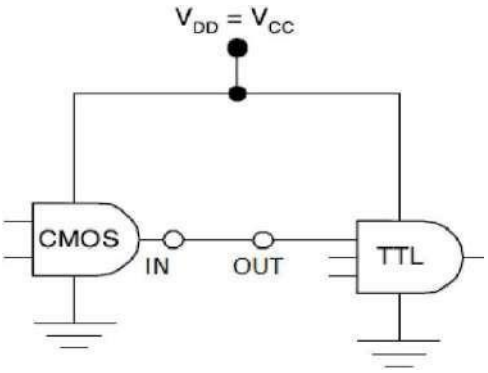
- The basic idea of current-mode logic is illustrated by the inverter/buffer circuit in the figure. This circuit has both an inverting output (OUT1) and a non-inverting output (OUT2).
- Twotransistorsareconnectedasadifferentialamplifierwithacommonemitterresistor.
- Thesupply voltages forthis exampleareVCC = 5.0, VBB= 4.0, andVEE= 0V, andtheinputLOW and HIGH levels are defined to be 3.6 and 4.4 V. This circuit actually produces output LOW and HIGH levels that are 0.6 V higher (4.2 and 5.0 V).



INTERFACINGOFTTLTO CMOS



INTERFACINGOFCMOSTOTTL



TTLvs. CMOS:-

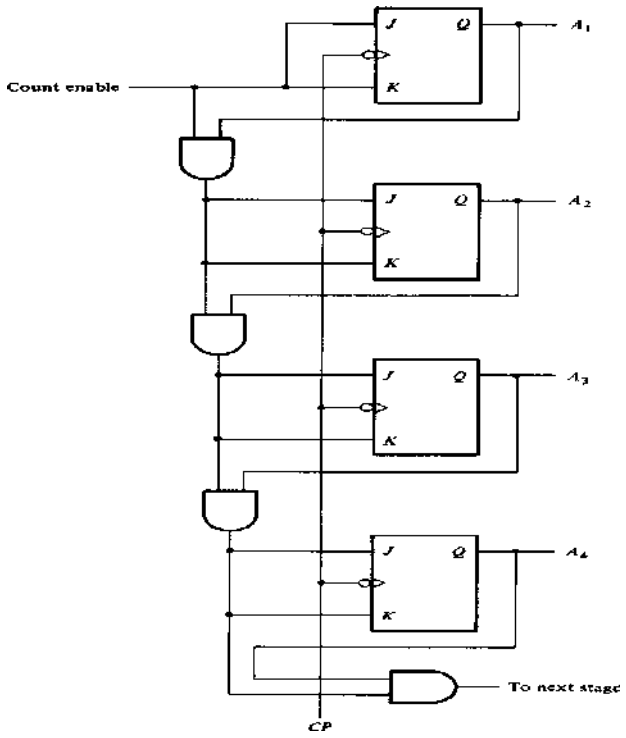
- TTLhaslesspropagationdelay than CMOS i.e.TTL isgood wherehigh speed isneeded. And
- CMOS 4000 is good for Battery equipment and where speed is not so important.
- CMOS requires lesspowerthan TTLi.e.powerdissipationandhencepowerconsumptionis lessfor CMOS.

COUNTER

- A counter is a device which stores (and sometimes displays) the number of times a particular event or process has occurred. In electronics, counters can be implemented quite easily using register-type circuits.
- There are different types of counters, viz.
    - Asynchronous(ripple)counter
    - Synchronouscounter
    - Decadecounter
    - Up/downcounter
    - Ringcounter
    - Johnsoncounter
    - Cascadedcounter
    - Modulus counter.

Synchronouscounter

- A4-bitsynchronouscounterusingJKflip-flopsisshown inthefigure.
- In synchronouscounters,the clockinputsof all theflip-flopsareconnectedtogetherandare triggered by the input pulses. Thus, all the flip-flops change state simultaneously (inparallel).

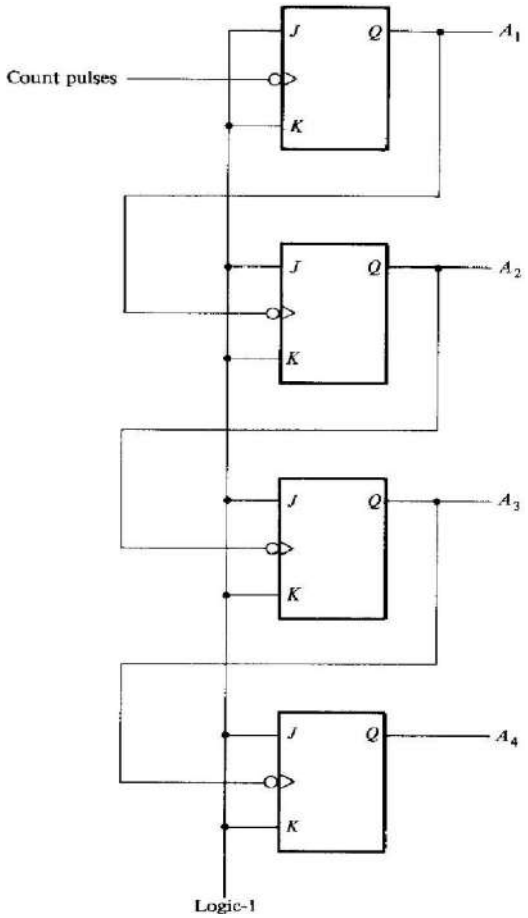


- The circuitbelowisa4-bitsynchronouscounter.
- The JandK inputs ofFF0 are connectedtoHIGH. FF1has its Jand Kinputsconnectedtotheoutput of FF0, and the J and K inputs of FF2 are connected to the output of an AND gate that is fed by the outputs of FF0 and FF1.
- Asimpleway ofimplementing thelogic for each bit ofan ascending counter (which iswhat isdepicted in the image to the right) is for each bit to toggle when all of the less significant bits are at a logic high state.
- For example, bit 1 toggles when bit 0 is logic high; bit 2 toggles when both bit 1 and bit 0 are logic high; bit 3 toggles when bit 2, bit 1 and bit 0 are all high; and so on.

- Synchronouscounterscanalso be implementedwith hardwarefinitestate machines,which are more complex but allow for smoother, more stable transitions.

AsynchronousCounter

- Anasynchronous(ripple)counterisasingled-typeflip-flop,withitsJ(data)inputfedfromitsown inverted output.
- This circuit can store one bit, and hence can count fromzero to one before it overflows (starts over from 0).



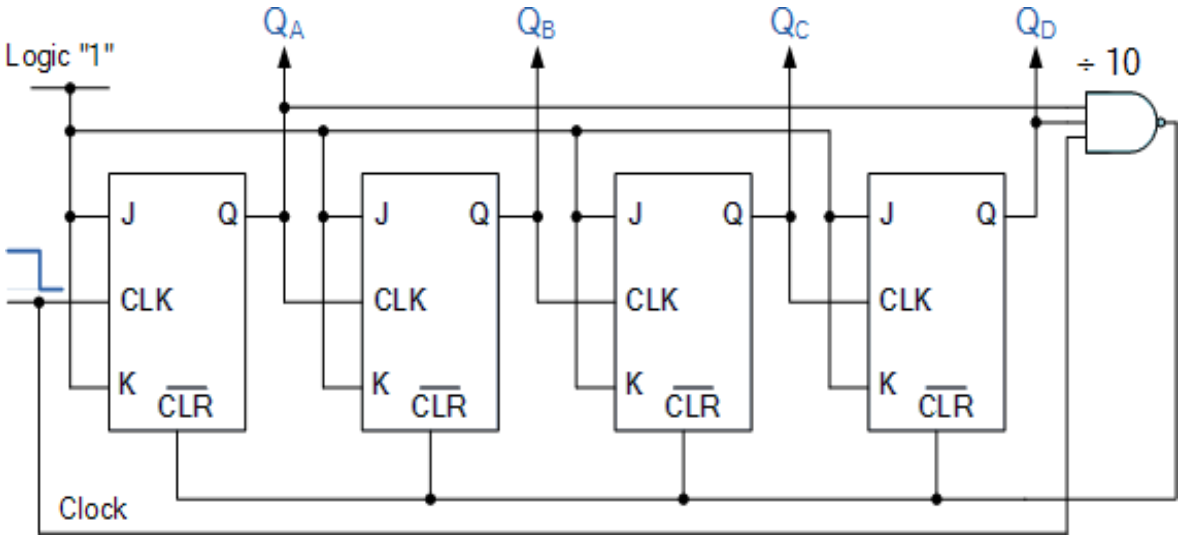
- This counter will increment once for every clock cycle and takes two clock cycles to overflow, so every cycle it will alternate between a transition from 0 to 1 and a transition from 1 to 0.
- Thiscreatesanewclockwitha50%dutycycleatexactlyhalfthefrequencyoftheinputclock.
- If this output is then used as the clock signal for a similarly arranged D flip-flop, remembering to invert the output to the input, one will get another 1 bit counter that counts half as fast. These together yield a two-bit counter.
- Additional flip-flops can be added, by always inverting the output to its own input, and using the output from thepreviousflip-flop asthe clock signal. Theresult is called aripple counter,whichcan count to $2^n - 1$, where n is the number of bits (flip-flop stages) in the counter.
- Ripple counters suffer from unstable outputs as the overflows "ripple" from stage to stage, but they find application as dividers for clock signals.

ModulusCounter

- A moduluscounter isthat whichproducesan outputpulseaftera certain number of inputpulses is applied. Inmoduluscounterthe totalcountpossibleisbasedonthenumberofstages,i.e.,digitpositions.
-

- Moduluscountersareusedindigitalcomputers.
- A binary modulo-8 counter with three flip-flops, i.e., three stages, will produce an output pulse, i.e., display an output one-digit, after eight input pulses have been counted, i.e., entered or applied. This assumes that the counter started in the zero-condition.

AsynchronousDecadeCounter



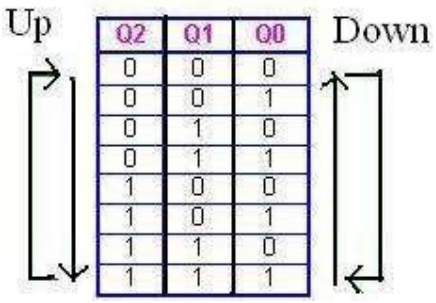
- Adecadecounter cancountfromBCD “0” toBCD“9”.
- A decadecounterrequiresresettingto zero whentheoutputcountreaches thedecimalvalueof 10,ie. when DCBA = 1010 and this condition is fed back to the reset input.
- A counter with a count sequence from binary “0000” (BCD = “0”) through to “1001” (BCD = “9”) is generallyreferred toasa BCDbinary-coded-decimalcounterbecauseitsten state sequenceis thatof a BCD code but binary decade counters are more common.
- This typeof asynchronous countercountsupwards oneachleadingedgeof the input clocksignal starting from 0000 until it reaches an output 1001 (decimal 9).
- Both outputs $Q_A$and $Q_D$are now equal to logic “1” and the output from the NAND gate changes state from logic“1” to alogic “0” leveland whoseoutputis alsoconnectedtothe CLEAR(CLR) inputsof all the J-K Flip-flops.
- This signal causesall of theQ outputs to be resetbacktobinary 0000onthe countof 10. OnceQAand QD are both equal to logic “0” the output of the NAND gate returns back to a logic level “1” and the counter restarts again from 0000. We now have a decade or Modulo-10counter.

DecadeCounterTruthTable

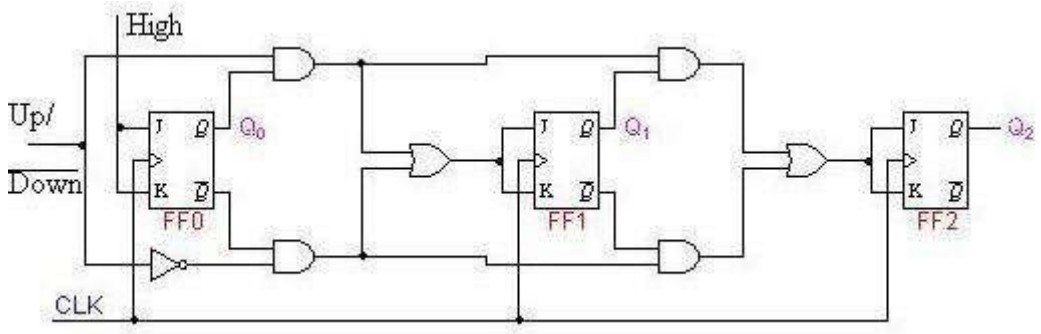
Clock Count	Output bit Pattern				Decimal Value
	QD	QC	QB	QA	
1	0	0	0	0	0
2	0	0	0	1	1
3	0	0	1	0	2
4	0	0	1	1	3
5	0	1	0	0	4
6	0	1	0	1	5
7	0	1	1	0	6
8	0	1	1	1	7
9	1	0	0	0	8
10	1	0	0	1	9
11	Counter Resets its Outputs back to Zero				

Up/DownCounter

- In a synchronousup-downbinary countertheflip-flop inthe lowest-orderpositionis complementedwith every pulse.
- Aflip-flop in anyotherposition iscomplemented with apulse,providedallthe lower-orderpulseequal to 0.
- Up/Down counter is used to control the direction of the counter through a certain sequence.



- From the sequence table we can observe that:
  - ForboththeUPandDOWNsequences,Q_0toggles oneachclockpulse.
  - FortheUPsequence,Q_1changesstateon thenext clockpulse whenQ_0=1.
  - FortheDOWNsequence,Q_1changesstateonthenextclockpulsewhenQ_0=0.
  - FortheUPsequence,Q_2changesstateon thenext clockpulse whenQ_0=Q_1=1.
  - FortheDOWNsequence,Q_2changesstateonthenextclockpulsewhenQ_0=Q_1=0.



- These characteristicsare implemented with theAND, OR&NOTlogicconnected asshownin thelogic diagram above.

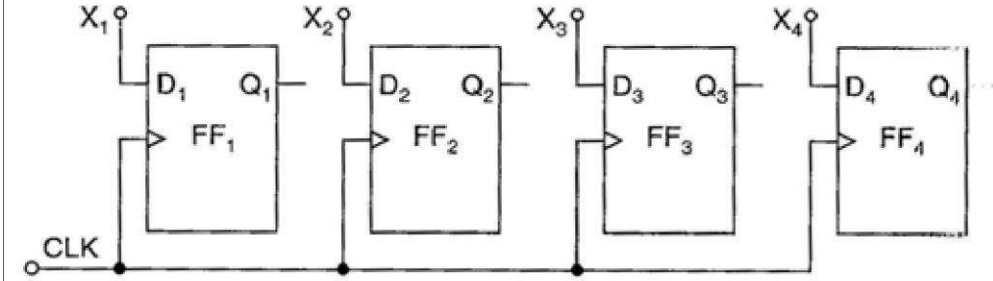
REGISTERS

INTRODUCTION:-

- Thesequentialcircuitsknownasregisterareveryimportant logicalblockin mostof thedigitalsystems. Registers
- are used for storage and transfer of binary information in a digital system.
- A registeris mostlyusedforthepurposeof storingand shiftingbinary dataenteredinto itfroman external source and has no characteristics internal sequence of states.
- The storagecapacity of aregisterisdefinedas the numberof bitsof digitaldata,it can storeorretain. These
- registers are normally used for temporary storage of data.

BUFFERREGISTER:-

- These are thesimplestregistersandareusedforsimply storinga binaryword. These
- may be controlled by Controlled Buffer Register.
- D flip -flops are used for constructing a buffer register orotherflip-flop can beused.
- Thefigureshownbelowisa4-bitbufferregister.



Logic diagram of a 4-bit buffer register.

- Thebinary wordtobestoredisappliedtothedataterminals.
- Whenthe clock pulseis applied,theoutput word becomes the sameasthe word appliedatthe input terminals, i.e. the input word is loaded into the register by the application of clockpulse.
- When the positive clock edge arrives, the stored word becomes:

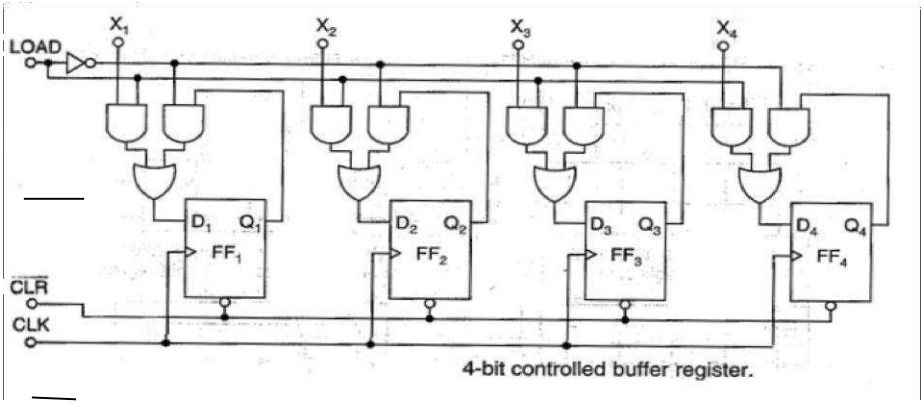
  Q4Q3Q2Q1=X4X3X2X1

  or Q=X .

  Thiscircuit istooprimitivetobeof anyuse.

CONTROLLEDBUFFERREGISTER:-

- The figure shows a controlled buffer register.



4-bit controlled buffer register.

- If CLRgoesLOW,all theflip-flopsare RESETand theoutputbecomes, Q=0000. When
- CLR is HIGH, the register is ready for action

- LOADiscontrol input.
- WhenLOADis HIGH, thedatabits XcanreachtheDinputsofFFs.
- Atthepositivegoingedgeofthenext clockpulse,theregisterisloaded,i.e.

  Q4Q3Q2Q1=X4X3X2X1

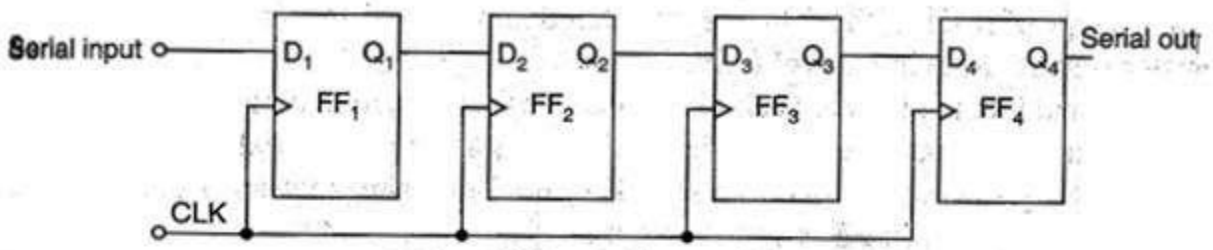
  or Q=X .
- WhenLOADisLOW,the XbitscannotreachtheFFs.Atthesametimetheinvertedsignal LOAD is HIGH. This forces each flip-flop output to feedback to its data input.
- Thereforedataiscirculated orretainedaseach clockpulsearrives.
- In otherwordsthecontentregisterremainsunchanged in spiteofthe clock pulses. Longer
- buffer registers can built by adding more FFs.

CONTROLLEDBUFFERREGISTER:-

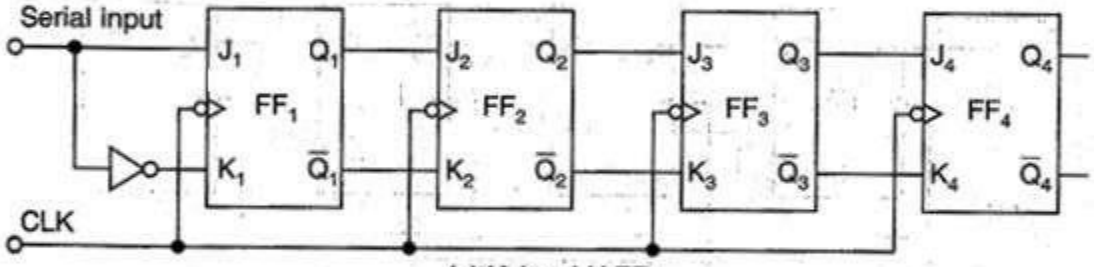
- A numberof FFs connected togethersuchthat data maybe shifted intoand shiftedoutof themiscalled a shift register.
- Data maybeshiftedintooroutof the registereitherin serialformorinparallelform. There
- are four basic types of shift registers
  1. Serialin,serialout
  2. Serialin,parallelout
  3. Parallelin,serialout
  4. Parallelin,parallel out

SERIALIN,SERIALOUTSHIFTREGISTER:-

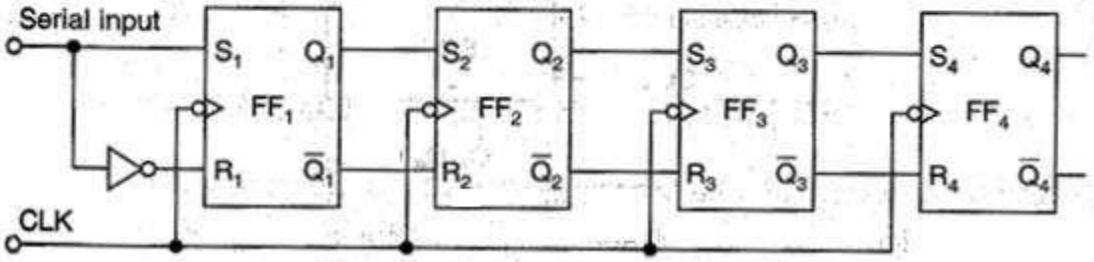
- This typeof shiftregisteracceptsdataserially, i.e., onebitata timeandalsooutputsdataserially. The
- logic diagram of a four bit serial in, serial out shift register is shown in belowfigure:
- In4stages i.e. with4FFs,theregistercanstoreupto4bitsof data.
- Serial data is applied at the D input of the first FF. The Q output ofthe first FF is connected to theD inputofthe secondFF, the outputofthesecond FFisconnectedtotheDinputofthe thirdFF and the Q output of the third FF is connected to the D input of the fourth FF. The data is outputted from the Q terminal of the last FF.
- When a serial data is transferred to a register, each new bit is clocked into the first FF at the positive going edge of each clock pulse.
- The bit that is previously stored by the first FF is transferred to the second FF.
- Thebit that is storedby thesecond FFistransferredtothe third FF,and soon. The
- bit that was stored by the last FF is shifted out.
- Ashift registercan alsobeconstructedusingJ-KFFsorS-RFFsasshownin thefigurebelow.



4-bit serial-in, serial-out, shift-right, shift register.



(a) Using J-K FFs

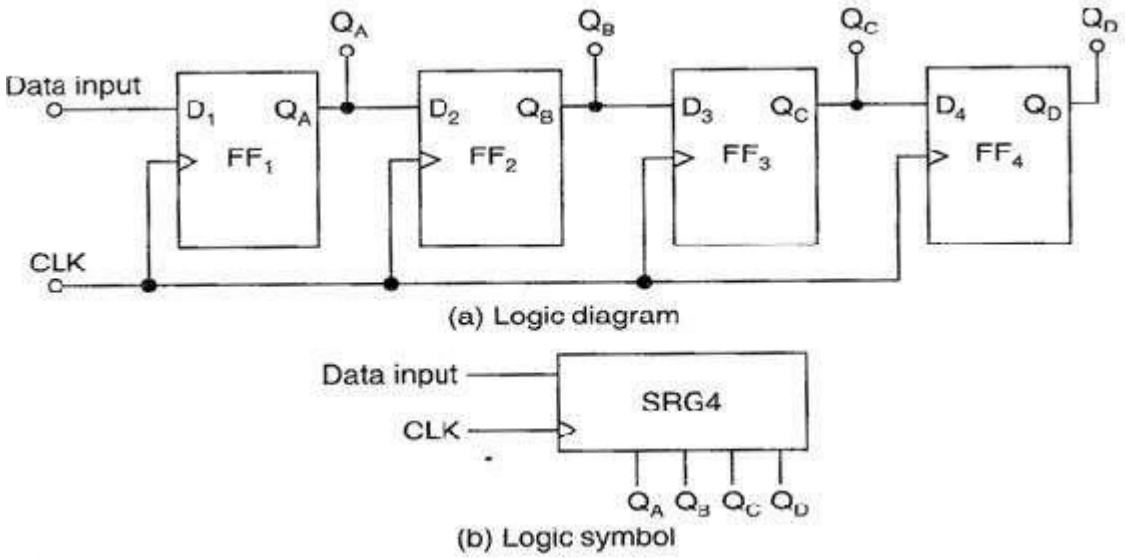


(b) Using S-R FFs

A 4-bit serial-in, serial-out, shift register.

SERIALIN,PARALLELOUTSHIFTREGISTER:-

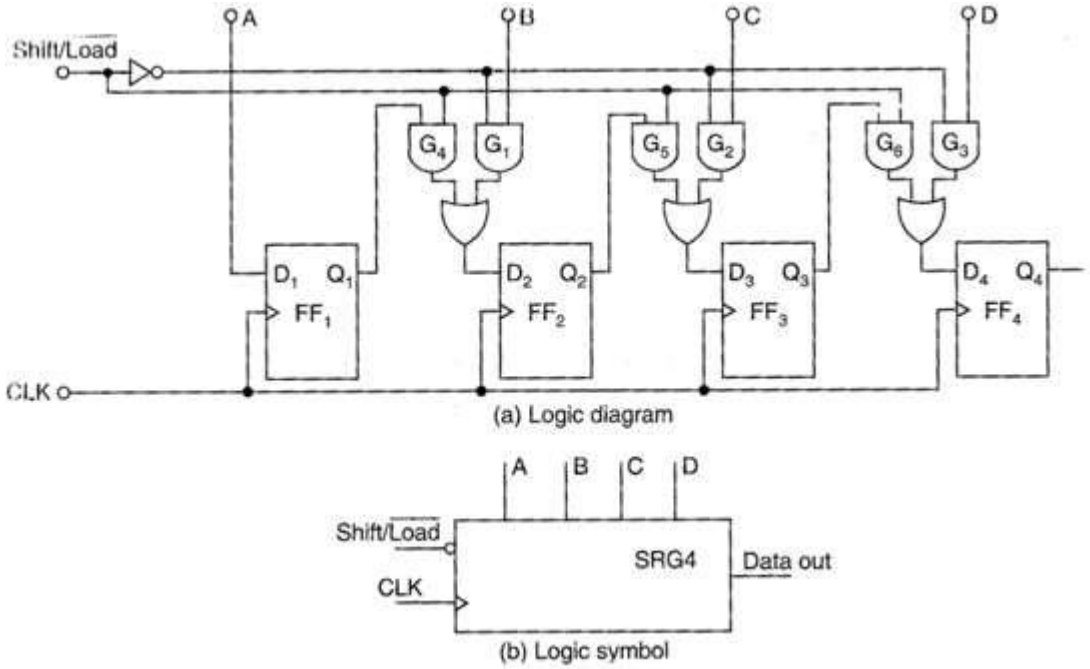
- In this typeof register,thedatabitsareentered into the registerserially,butthe data stored inthe register serially, but the stored in the register is shifted out in the parallel form.
- Whenthedatabitsare storedonce,eachbitsappearson its respectiveoutput lineand allbitsare availablesimultaneously,ratherthanbit–by–bitbasisasintheserialoutput.
- The serialin, parallelout shift registercanbeused asaserialin, serialout shiftregisterif theoutputis taken from the Q terminal of the last FF.
- Thelogicdiagramand logicsymbolofa 4bitserialin,parallelout shiftregisterisgivenbelow.



A4-bitserialin,paralleloutshiftregister

PARALLELIN,SERIALOUTSHIFTREGISTER:-

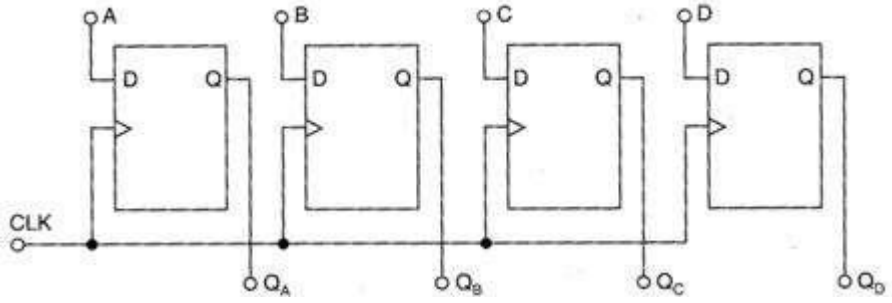
- For parallel in, serial out shift register the data bits are entered simultaneously into their respectivestages onparallel lines, rather thanonbit by bitbasis on one line as with serialdata inputs,butthedata bits are transferred out of the register serially, i.e.,on a bit by bit basis over a single line.
- Thelogicdiagramandlogicsymbolof4bitparallelin,serialoutshiftregisterusingDFFsisshown below.
- Therearefourdata lines A, B, Cand Dthrough which thedata isentered intothe registerin parallel form.
- The signal Shift /LOAD allows
  1. Thedatatobeenteredinparallelformintotheregisterand
  2. Thedatatobeshiftedoutseriallyfromterminal Q_4.
- WhenShift /LOADline is HIGH, gates G1,G2, and G3 are disabled, butgatesG4, G5and G6are enabled allowing the data bits to shift right from one stage to next.
- WhenShift /LOAD line is LOW,gatesG4,G5 andG6aredisabled, whereasgatesG1,G2andG3are enabled allowing the data input to appear at the D inputs of the respective FFs.
- Whenclockpulse isapplied, thesedatabitsare shifted to the Qoutputterminalsof the FFsand therefore the data is inputted in one step.
- TheORgate allows eitherthenormalshiftingoperation orthe paralleldataentry dependingon which AND gates are enabled by the level on the Shift /LOAD input.



A4-bitparallelin,serialoutshift register

PARALLELIN,PARALLELOUTSHIFTREGISTER:-

- In a parallel in, parallel out shift register, the data entered into the register in parallel form and alsothe data taken out of the register in parallel form. Immediately following the simultaneous entry of all data bits appear on the parallel outputs.
- Thefigure shownbelowisa4bit parallelinparallelout shift registerusingDFFs. Data
- applied to the D input terminals of the FFs.
- Whena clockpulse isappliedatthe positiveedgeof thatpulse, theDinputs are shifted intothe Q outputs of the FFs.
- The registernowstoresthe data.
- Thestoreddataisavailableinstantaneouslyforshiftingoutinparallelform.

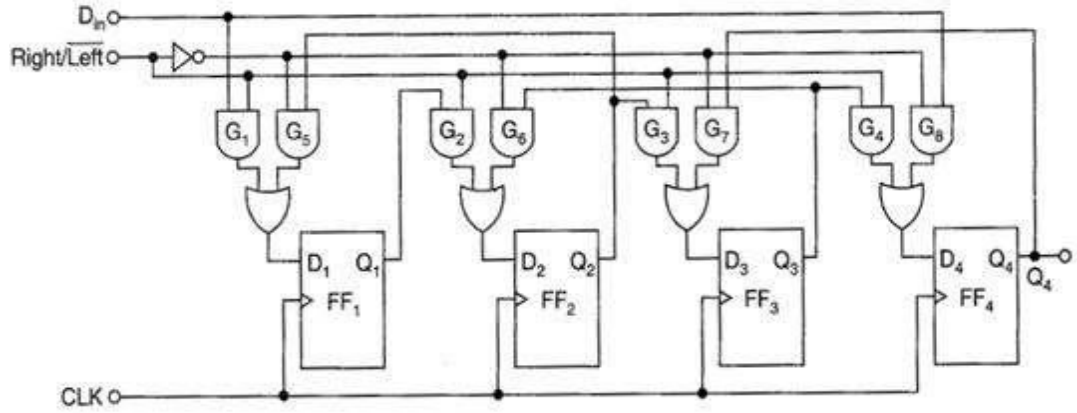


Logicdiagramofa4– bitparallelin,paralleloutshiftregister

BIDIRECTIONALSHIFTREGISTER:-

- In bidirectional shift register isone in which thedatabits canbeshiftedfromlefttoright orfrom rightto left.
- Thefigure shownbelowthe logic diagramof a4bit serialin, serialout,bidirectional(shift-left,shift- right) shift register.
- Right/Left isthe modesignal.When Right/Leftisa 1, the logiccircuit worksasa shift rightshift register. When Right /Left is a 0, the logic circuit works as a shift right shift register.

- The bidirectional is achieved by using the mode signal and two AND gates and one OR gate for each stage.
- A HIGH on the Right/$\overline{Left}$ control input enables the AND gates G_1, G_2, $G_3$and $G_4$and disables the AND gates G_5, G_6, $G_7$and $G_8$and the state ofQ output ofeachFF is passed through the gate to the D input of the following FF. When clock pulse occurs, the data bits are effectively shifted one place to theright.
- A LOWRight/$\overline{Left}$ control input enables the AND gates G_5, G_6, $G_7$and $G_8$and disables the AND gates G_1,G_2, $G_3$and G_4andthe Q outputofeachFF ispassed to theD inputofthe precedingFF. When clock pulse occurs the data bits are then effectively shifted one place to the left.
- So,thecircuitworksasabidirectionalshiftregister.



Logicdiagramof4-bitbidirectionalshiftregister

UNIVERSALSHIFTREGISTERS:-

- Theregisterwhichhasbothshifts andparallel loadcapabilities,itisreferredas auniversalshift register. So, universal shift register is a bidirectional register, whose input can be either in serial form or in parallel form and whose output also can be either in serial form or parallelform.
- Theuniversalshiftregistercanberealizedusing multiplexers.
- The figure shows the logic diagram of a 4 bit universal shift register that has all the capabilities of a general shift register.

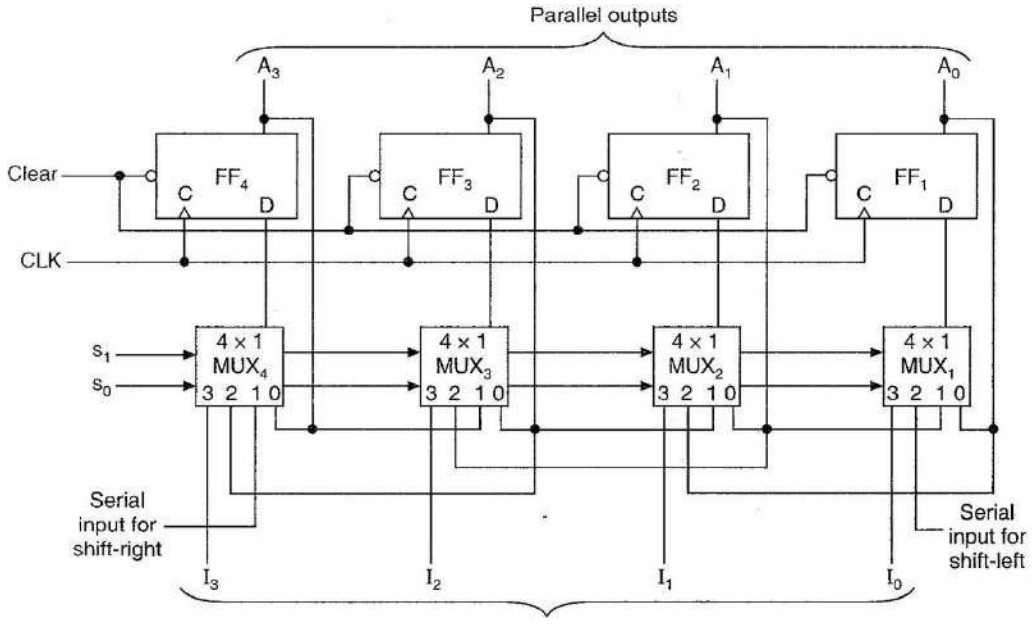


Fig-(a) 4bituniversalshiftregister

- It consists of four D flip-flops and four multiplexers.
- The four multiplexers have two common selection inputs S_1 and S_0.
- Input 0 in each multiplexer is selected when S_1S_0= 00, input 1 is selected when S_1S_0=01, and input 2 is selected when S_1S_0= 10 and input 3 is selected when S_1S_0= 11.
- The selection inputs control the mode of operation of the register is according to the function entries shown in the table.
- When S_1S_0=00 the present value of the register is applied to the D inputs of flip-flops. This condition forms a path from the output of each FF into the input of the same FF.
- The next clock edge transfers into each FF the binary value it held previously, and no change of state occurs.
- When S_1S_0=01, terminal 1 of the multiplexer inputs have a path of the D inputs of the flip-flops. This causes a shift right operation, with serial input transferred into FF_4.
- When S_1S_0=10 a shift left operation results with the other serial input going into the FF_1.
- Finally when S_1S_0=11, the binary information on the parallel input lines is transferred into the register simultaneously during the next clock edge.

Functional table for the register of fig – a:

Mode control		
S_1	S_0	Register operation
0	0	No change
0	1	Shift right
1	0	Shift left
1	1	Parallel load

APPLICATIONS OF SHIFT REGISTERS:-

1. **Time delays:**
   - In digital systems, it is necessary to delay the transfer of data until the operation of the other data have been completed, or to synchronize the arrival of data at a subsystem where it is processed with other data.
   - A shift register can be used to delay the arrival of serial data by a specific number of clock pulses, since the number of stages corresponds to the number of clock pulses required to shift each bit completely through the register.
   - The total time delay can be controlled by adjusting the clock frequency and by the number of stages in the register.
   - In practice, the clock frequency is fixed and the total delay can be adjusted only by controlling the number of stages through which the data is passed.
2. **Serial/Parallel data conversion:**
   - Transfer of data in parallel form is much faster than that in serial form.
   - Similarly the processing of data is much faster when all the data bits are available simultaneously. Thus in digital systems in which speed is important so to operate on data parallel form is used.
   - When large data is to be transmitted over long distances, transmitting data on parallel lines is costly and impracticable.
   - It is convenient and economical to transmit data in serial form, since serial data transmission requires only one line.

- Shift registers are used for converting serial data to parallel form, so that a serial input can be processed by a parallel system and for converting parallel data to serial form, so that paralleldata can be transmitted serially.
- A serial in, parallel out shift register can be used to perform serial-to parallel conversion, and a parallel in, serial out shift register can be used to perform parallel- to –serial conversion.
- A universal shift register can be used to perform both the serial- to–parallel and parallel-to-serial data conversion.
- A bidirectional shift register can be used to reverse the order of data.

RINGANDJOHNSONCOUNTER:-

- Ringcountersareconstructedbymodifyingtheserial-in,serial-out,shiftregister.
- Therearetwotypesofringcounters
  - i) Basicringcounter
  - ii) Johnsoncounter
- Thebasicringcounter canbeobtainedfroma serial-in serial-outshift registerby connecting theQ output of the last FF to the D input of the first FF.
- The Johnson counter canbeobtainedfromserial-in, serial- out, shiftregisterby connectingthe Q output of the last FF to the D input of the first FF.
- Ring counteroutputscanbeused asasequence ofsynchronizingpulses. The
- ring counter is a decimal counter.

D/AandA/DConverter

WeightedRegisterNetwork

Themostsignificantbit(MSB)resistanceisone-eighthoftheleastsignificantbit(LSB)resistance. $R_L$ismuchlargerthan 8R. The voltages $V_Æ$, V_B, $V_C$and $V_D$can be either equal to V (for logic 1) or 0 (for logical 0). Thus there are 2^4= 16 input combinations from 0000 to 1111. The output voltage V_O, given by Millman's theorem is

V_O=

Wheninput is 0001, $V_Æ$= V_B=V_C=0 and V_D= V and output is V/15. If input is 0010, $V_Æ$= V_B=V_D=0 and V_C=Vgiving an outputof 2V/15. If input is 0011, $V_Æ$= V_B=0and V_C=V_D= V giving an output of 3v/15. Thus, the output voltage varies from 0 to V in steps of V/15.

BinaryLadder Network

The weighted resistor network requires a range of resistor values. The binary ladder network requires only two resistancevalues.From node1,theresistance tothedigitalsourceis2Randresistance togroundisalso2R.Fromnode 2, the resistance to digital source is 2R and resistance to ground =

R+(2R)(2R)/(2R+2R)=2R

Thus,fromeachof thenodes 1,2,3,4,theresistanceto sourceand groundis2Reach.Adigitalinput0001 meansthatDis connected to V and A, B, C are grounded. The output voltage $V_O$is V/16. Thus as input varies from 0000 to 1111, the output varies from V/16 to V in steps of V/16.

Acompletedigital-to-analogconvertercircuitconsists ofanumberofladdernetworks(todealwithmorebitsofdata), operational amplifier, gates etc.

PerformanceCharacteristicsofD/Aconverters

TheperformancecharacteristicsofD/Aconvertersare resolution,accuracy,linearerrors,monotonicity,settingtimeand temperature sensitivity.

(a) **Resolution:**Itisthereciprocalofthenumberofdiscretestepsintheoutput.Evidentlyresolutiondepends on the number of bits. The percentage resolution is [1/ (2^N-1)] * 100 where N is the number of bits. The percentage resolution for different values of N is given intable.

(b) **Accuracy:**Itisameasureofthedifferencebetweenactualoutputandexpectedoutput.Itisexpressedasa percentageofthemaximumoutputvoltage.Ifthemaximumoutputvoltage(orfullscaledeflection)is5V and accuracyis±0.1%,thenthemaximumerroris $\frac{0.1}{100}$*~~5=~~0.005Vor5mV.Ideallytheaccuracyshouldbebetter than±0.5of LSB.Inan8bitconverter,LSB is1/256or 0.39%offullscale.Theaccuracy shouldbebetterthan 0.2%.

(c) **Setting Time:** When the input signal changes, it is desirable that analog output signal should immediately show the new output value. However in actual practice, the D/A converter takes some time to settle at the new positionoftheoutputvoltage.Settingtimeis definedasthetimetakenbythe D/Aconverterto settlewith±1/2 LSB of its final value when a change in input digital signal occurs. The final time taken to settle down to new value is due to the transients and oscillations in the output voltage. Figure shows the definition of settingtime.

Table

3 bit Binary word	Analog voltage
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

Fig 1

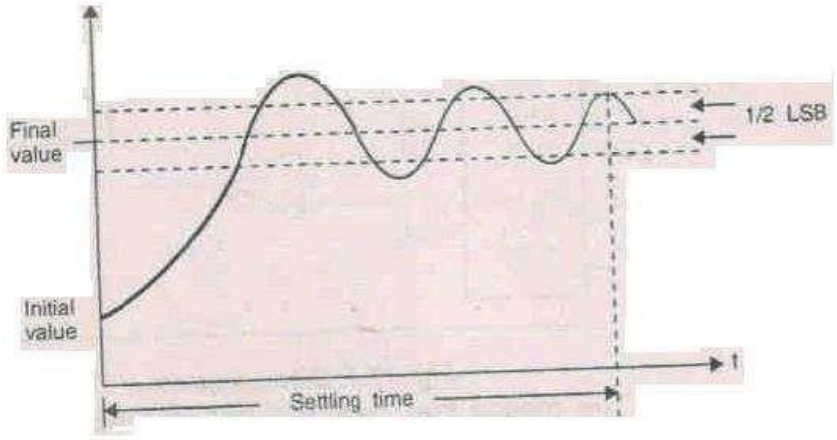
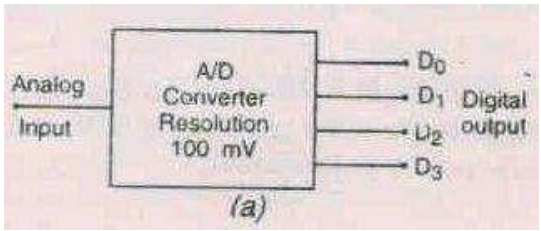


Fig. Settling time of D/A converter

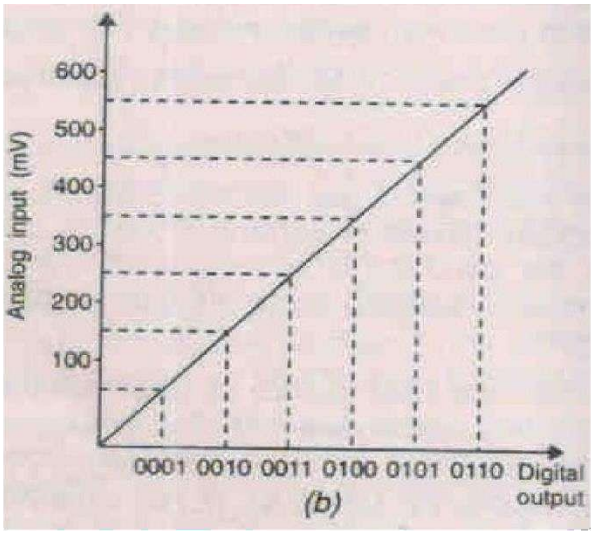
Quantizationerror:

An analog to digital converter changes analog signal into digital signal. It is important to note that in D/A converter the number of input is fixed. In 4 bit D/a converter there are 16 possible inputs and in 6 bit D/A converterthereare64 possibleinputs.However,inA/Dconvertertheanaloginputvoltagecanhaveanyvaluein the specified range but the digital output can have only 2^Ndiscrete levels (for N bit converter). This means that there is a certain range of input voltage which correspond to every discrete output level.

Consider a 4 bit A/D converter having a resolution of 1 count per 100 mV. Fig (b) shows the analog input and digital output. It is seen that for input voltage range of 50 mV to 150 mV, the output is same i.e. 0001, for input voltage range of 150 mV to 250 mV, the output is the same, i.e. 0010. Thus we have one digital output for each 100 mV input range. If the digital signal of 0010 is fed to a D/A converter, it will show an output of 200 V whereastheoriginalinputvoltagewasbetween150V and250 v.Thiserroriscalledquntisationerrorandinthis case this quntisation error can be ±50 mV and is equal to ±1/2 LSB.



Fig(a)A/DConverter



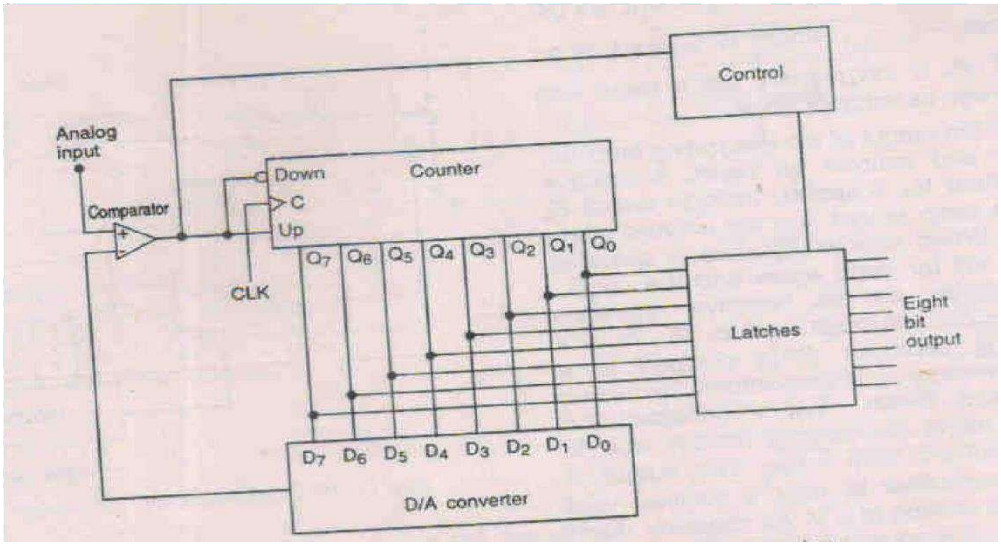
Fig(b)Quantisationerror

StairStepA/DConverter /RampA/D converter:

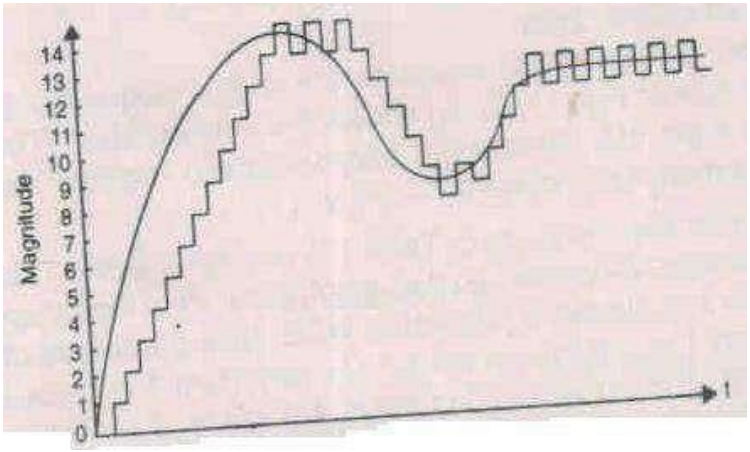
Thisconverterisalsocalleddigitalramporthecounter typeA/Dconverter.Figureshowsthe configurationfor8 bit converter. As seen in figure it uses a D/A converter and a binary counter to produce the digital number corresponding to analog input. The main components are comparator, AND gate, D/A converter, divide by 256 counter and latches. The analog input is given to non-inverting terminal of comparator. The D/A converter provides stair step reference voltage.

Let he counter be in reset state and output of D/A converter be zero. An analog input is given to non-inverting terminal of comparator. Since the reference input is 0, the comparator gives High output and enables the AND gate. The clock pulses cause advancing of counter through its binary states and stair step reference voltage is produced from D/A converter. As the counter keeps advancing, successively higher stair step output voltage is produced. When this stair step voltage reaches the level of analog input voltage, the comparator output goes LowanddisablestheANDgate.Theclockpulsesarecutoffand counterstops. Thestate ofcounteratthispoint is equal to the number of steps in reference voltage at which comparison occurs. The binary number corresponding to this number of steps is the value of the analog input voltage. The control logic causes this binary number to be loaded into the latches and counter is reset.

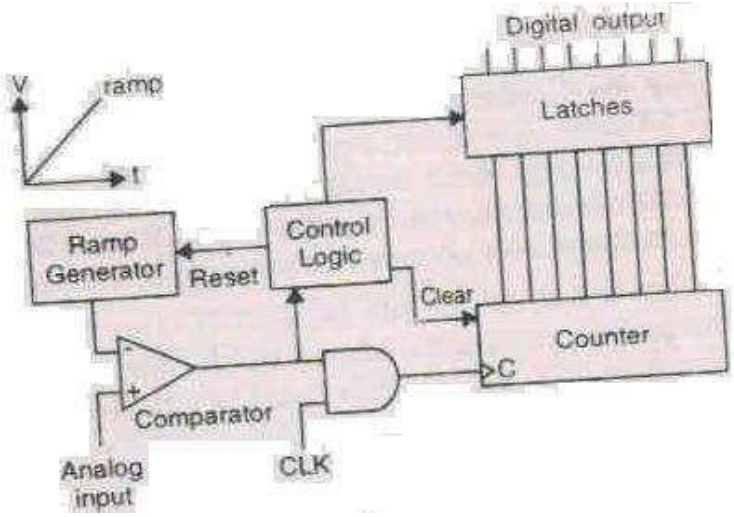
Thisconverterisratherslowinaction becausethe counterhasto passthroughthemaximumnumberofstates before a conversion takes place. For 8 bit device this means 256 counter states.



Fig(a)8bitup-downcountertypeA/Dconverter



Fig(b)Trackingactionofupdowncountertype A/D Converter



Fig(c)SingleslopeA/Dconverter

DualslopeA/Dconverter:

ThesingleslopeA/Dconverterissuscetible to noise. Thedualslopeconverterisfree fromthisproblem.Ituses anop-ampusedasintegretingamplifierforrampgenerator.Itisdualslopedevicebecauseitusesafixedslope ramp as well as variable slope ramp. Fig. Shows the configuration.

Itisseenthattheintegretingop-ampusesacapacitorinthefeedbackpath.

Outputvoltageofintegretingop-amp=$-\frac{1}{C}\int i\,dt=-\frac{1}{RC}\int V_{in}\,dt$

Thustheoutputvoltageisintegralofanaloginputvoltage. IfV_{in}isconstant,wegetanoutput-$\frac{V_{in}}{RC}t$whichis afixedsloperamp. If V_{in}isvaring wegetarampwithfixedaswell asvariableslope.

Lettheoutputoftheintegretingamplifierbezeroandcounterbereset.ApositiveanaloginputV_{in}isapplied through switch S, we get a ramp output and the counter starts working. When counter reaches a specified count, it will be reset again and the control logic switches on the negative reference voltage - V_{ref}(through switch S). At this instant the capacitor C is charged to a negative voltage - V proportional to analog input voltage.When-V_{ref}isconnectedthecapacitorstartsdischarginglinearlyduetoconstantcurrentfrom-V_{ref}.

Theoutputofintegretingamplifierisnowapositivefixedsloperampstartingat–V.Ascapacitordischarges,the counteradvancesfromtheresetstate.Whentheoutputofintegretorbecomeszero,thecomparatoroutput

becomes Low and disables the clock signal to the AND gate. The counter is therefore stopped and the binary counterislatched.This completesoneconversioncycle.Thebinarycountisproportionalto analoginput V_{in}.

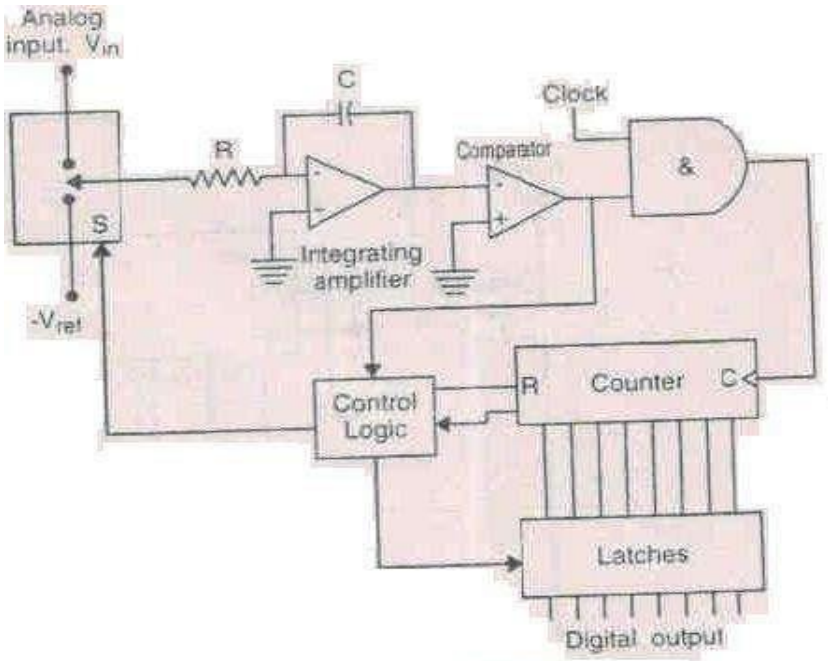


Fig. Dual slope A/D converter

SuccessiveApproximationA/DConverter:

This is the most widely used A/D converter. As the name suggests the digital output tends towards analog inputthrough successive approximations. Fig. Shows the configuration. The main components are op-amp comparator,controllogic,SA(successiveapproximation)registerandD/Aconverter.Asshownitisasixbit device using a maximum reference of 64 V.

Let the analog input be 26.1 v. The SA register is first set to zero. Then 1 is placed in MSB. This is fed to D/A converter whose output goes to comparator. Sincetheanalog input (26.1 V) isless than D/A output (i.e. 32V) the MSB is set to zero. Then 1is placedinbit next to MSB. Nowthe output of D/A is 16 V. Since analog input ismore than 16 V, this 1 is retained in this bit position. Next 1 is placed in third bit position. Now the D/A output is 24 V which is less than analog input. Therefore this 1 bit is retained and 1 is placed in the nextbit.Now the D/A output is 28 V, which is more than analog input. Therefore this 1 bit is set to zero and 1 is placed in 5^{th} bit position producing a D/Aoutput of 26 V. It islessthan analog input. Thereforethis1 bit isretained. Now1 isplacedinLSB producing a D/A output of 27 V which ismore than analog input. Therefore LSB is set to zero and the converter gives an output of 26 V.

ThesuccessiveapproximationmethodofA/Dconverterisveryfastandtakesonlyabout250ns/bit.

PerformanceCharacteristicsofA/Dconverters:

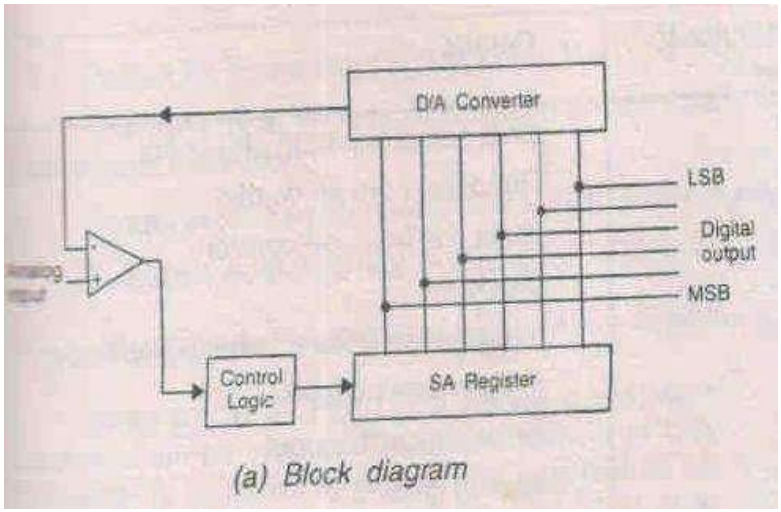
TheperformancecharacteristicsofA/Dconvertersareresolution,accuracy,A/D gainanddriftandA/Dspeed.

(a) Resolution: A/D rsolutionisthe change involtage inputnecessary for aone bit change inoutput.It can also be expressed as percent.

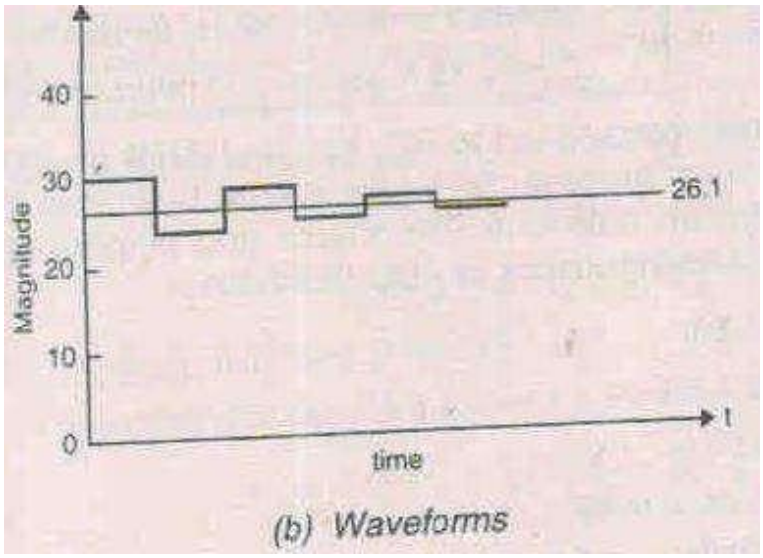
(b) A/D Accuracy:The accuracyof A/Dconversionislimited by the ±1/2 LSBdue to quantisation error andthe other errors of the system. It is defined as the maximum deviation of digital output from the ideal linear reference line. Ideally it aproaches ±1/2 LSB.

(c) A/D gainand Drift: A/D gainisthevoltage output isdevided bythevoltage input at thelinearityreferenceline. It can usually be zeroed out.
Driftmeanschangeincircuitparameterswithtime. Drifterrorsof upto±1/2LSB will causeamaximum errors of one LSB between the first and the last transition. Very low drift is quite difficult to achieve and increases cost of the device.

(d) A/D speed: It can bedefinedintwo ways, i.e.either thetime necessary to do one conversion ortheline between successive conversion at the highest rate possible. Speed depends on the settling time of components and the speed of the logic.



(a) Block diagram



(b) Waveforms