CONTENTS

Date 21·10·2022
Page 01

BOOLEAN ALGEBRA

1. TRUTH TABLE
2. CIRCUIT DIAGRAM
3. ALGEBRIC EXPRESSION
4. K·MAP (Simplification)

Boolean Algebra:

Introduction: Boolean algebra is a kind of algebra that is fundamental to computer operation and frequently used in digital electronics. It is also based in switching function. in all digital systems. There are only two values that is input/output. In order to obtained closed form and expression blw input and output algebra known as switching algebra. that also developed by 'Claude E.' An electrical engineer he suggested that this algebra could be applied to solve the problems arising in telephone switching circuits. This switching algebra is a subsact of more genral algebra known as 'Boolean-algebra'.

P·T·O·

Boolean algebra was developed by 'George boole (1815-1864)'. He defined his principle in his book "An investigation of law law of thoughts" 1854.

Basic Technology of boolean algebra

① LOGICAL STATEMENT: The statements which can take only two values that is either true or false are called logical statement.

② LOGICAL CONSTANT: The truth table / value of logical statements are called logical statements. There are two logical constants which are true or false.

③ BINARY VALUED VARIABLE: The variables which can take only two values are called 'Binary valued variables'. That is 0 and 1. where True is equal to 1. and False equal to 0. True is called open switch and false is called closed switch.

Date 21.10.2022
Page 03

BOOLEAN OPERATORS AND STATEMENTS

① Compound statement : Logical statements which are combined with the help of logical operators to form compound statements.

② Logical operator : Operators which are used to form compound statements are called logical operators they are as follows : NOT , AND and OR.

Date 03.11.2022

There are three types of logical operators. They are as follows :

(i) NOT or (Negation operator)
(ii) AND (conjuction operator)
(iii) OR (Disjunction operators)

P. T. O.

NOT: This operator is unarray operator because it operates on single variable. This operator is also called 'complement' and the symbol for this operator is 'N' or '-' or '/' is same similarly x, $\bar{x}$, x'. Thus it will complement if it apply in the circuit.

Circuit diagram of not operator

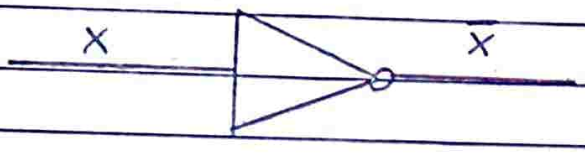
Eg: we can define the not operator in the truth table.

Single variable truth table in NOT operator.

TRUTH TABLE

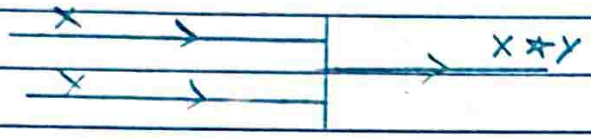
X	$\bar{X}$
0	1
1	0

CIRCUIT DIAGRAM



AND: This operator operates on two or more operants, AND operators is used to perform logical multiplication and this symbol for AND operator is DOT (·). sign.

CIRCUIT DIAGRAM



$2^n = 2^2 = 4$ rows

TRUTH TABLE FOR AND OPERATOR

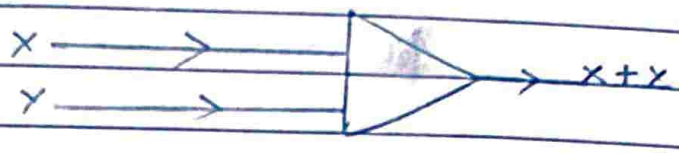
X	Y	XY
0	0	0
0	1	0
1	0	0
1	1	1

P.T.O.

Date 03.11.2022
Page 06

OR :- OR operators is used to perform logical addition and the symbol for OR operator is (+) sign.
Eg :- X + Y

CIRCUIT DIAGRAM



TRUTH TABLE

X	Y	X+Y
0	0	0
0	1	1
1	0	1
1	1	1

Q. What is Boolean algebra?

The algebra with deals with only two quantities that is either true or false are called boolean algebra. The truth table value true of boolean algebra denoted by 1 and the false value of boolean algebra is denoted by 0.

Date 03.11.2022
Page 07

Q. What is truth table?

Truth table is the tabular representation of all possible values of logical variables with all the possible outcomes for the given combination.

A Truth table consist of two major parts they are Row and Column.

ROW:

The Horizontal lines of truth table are called rows the number of rows depends upon the number of input variables.

e.g. if we have two input values then there will be 2^n formula.

Where 'n' is the number of variable, Therefore $2^2 = 2^2 = 4$ rows.

COLOUMN:

The verticle lines in a truth table are called columns.

X	Y	F
0	0	0
	1	
	0	

→ Rows

↓ Column

03.11.2021
08

Calculation of truth table :

Rules :

(i) Find out the values in the expressions.

(ii) Once the value got put them into the formula.

(iii) Arrange them with all its possible outcomes.

Q. Prove the following using truth table method.

(i) $x + (x \cdot y) = x$

(ii) $(x')' = x$

(iii) $(x \cdot y)' = x' + y'$

(iv) $x + x'y = (x+x')(x+y)$

(i) $x + (x \cdot y) = x$

x	y	$x \cdot y$	$x + x \cdot y$	x
0	0	0	0	0
0	1	0	0	0
1	0	0	1	1
1	1	1	1	1

RHS = LHS Proved

Date 03.11.2022
Page 09

(zii) $(x')^x = x$

x	$(x')'$	x
0	0	0
0	0	0
1	1	1
1	1	1

RHS = LHS Proved

(ziii) $(x \cdot x)' = x' + x'$

x	y	$x \cdot x$	$x' + x'$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

RHS ≠ LHS proved

(ziv) $x + x'y = (x + x')(x + y)$

x	y	$x + x'y$	$(x + x')(x + y)$
0	0	1	0
0	1	1	0
1	0	1	1
1	1	1	1

RHS = LHS proved

Date 10.11.2022
Page 10

BASIC LOGIC GATES:

Logic gates are the basic electronic circuits which perform the logical operations. The logic gates receives one or more input and produce a single output only after performing desired logical operation on the input.

"Many inputes but single output".

To make a complete circuit we have to use diode and transistor which makes the utilization of input and output of the electric current.

The gates are of two types

(i) Fundamental gates
(ii) Universal gates (deserved gate)

(i) Fundamental gates:

These gates are the logic gates with the help of which all other gates are created. Mainly their are three basic gates -

Date 10.11.2022
Page 11

(i) NOT GATE
(ii) AND GATE
(iii) OR GATE

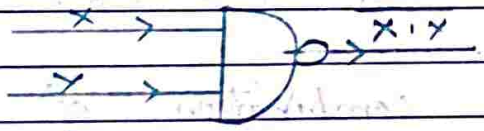
(ii) UNIVERSAL GATE :

These are the gates which combine with two logic gates to perform the desired output basically we use "NAND and NOR" Gates as are universal gate. In which a modified outcomes will be displayed.

(a) NAND GATE :

This gate is the combination of AND and NOT gate. The NAND gate has Two or more input but only one output. it produces Output "1." when any of the input is 0.

CIRCUIT DIAGRAM



TRUTH TABLE

X	Y	X.Y	$\overline{X.Y}$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

Date 10.11.2022
Page 12

NOR GATE:-

This gate is the combination of 'NOT and OR' gate. The NOR gate has two or more input but only one output. It produces output 1 when all inputs are 0.

CIRCUIT DIAGRAM

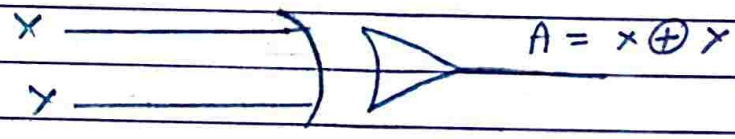
TRUTH TABLE

X	Y	X+Y	$\overline{X+Y}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

EXCLUSIVE OR (X-OR) - A⊕B :

It is the combination of OR AND exclusive that means a additional space is provided to complete the circuit.

CIRCUIT DIAGRAM



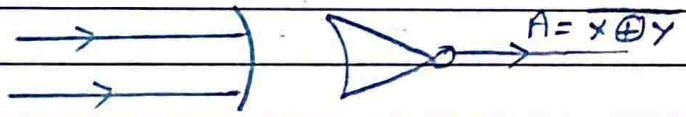
TRUTH TABLE

X	Y	A = X+Y
0	0	0
0	1	1
1	0	1
1	1	0

Exclusive NOR = X - NOR $(\overline{A \oplus B})$

In this circuit we find that if the same value present in the circuit will does not supply the output current.

CIRCUIT DIAGRAM



TRUTH TABLE

X	Y	$A = \overline{X+Y}$
0	0	1
0	1	0
1	0	0
1	1	1

Terminology of Boolean algebra

A Boolean expression can be described by following terms:

(i) Product term:

It is also known as (Minterm) a product term is a term of boolean expression in which two or more logical variables are connected by AND operator.

Ex. $X \cdot Y \cdot Z$. It is defined by a symbol "Σ" sign. (Sumbation).

TABLE OF MINTERM:

X	Y	Z	Minterm	Designation
0	0	0	$\bar{x} \cdot \bar{y} \cdot \bar{z}$	0
0	0	1	$\bar{x} \cdot \bar{y} \cdot z$	1
0	1	0	$\bar{x} \cdot y \cdot \bar{z}$	2
0	1	1	$\bar{x} \cdot y \cdot z$	3
1	0	0	$x \cdot \bar{y} \cdot \bar{z}$	4
1	0	1	$x \cdot \bar{y} \cdot z$	5
1	1	0	$x \cdot y \cdot \bar{z}$	6
1	1	1	$x \cdot y \cdot z$	7

Date 29.11.2022
Page 17

SUM TERM:

It is also known as ('MAx term'). A sum term is a term of boolean expression, In which two or more logical variables are connected by 'OR' operator.

Ex. $x+y+z$ It is defined by a symbol π sign.

TABLE OF SUM TERM:

x	y	z	SUM TERM	DESIGNATION
0	0	0	$x+y+z$	0
0	0	1	$x+y+\bar{z}$	1
0	1	0	$x+\bar{y}+z$	2
0	1	1	$x+\bar{y}+\bar{z}$	3
1	0	0	$\bar{x}+y+z$	4
1	0	1	$\bar{x}+y+\bar{z}$	5
1	1	0	$\bar{x}+\bar{y}+z$	6
1	1	1	$\bar{x}+\bar{y}+\bar{z}$	7

Date 30.11.2022
Page 18

K - MAP - KARNAUGH MAP

A Karnaugh map method is a graphical technique for simplifying boolean function. It is a two dimensional representation of a truth table its provides a simple method for minimizing logic expression. This method is very much suitable upto four variables, No need to applying the logics and theorems for simplifying. execution can be defined in two forms.

(i) Mean Terms (SOP)
(ii) Max Term (POS)

A boolean function can be represent by a formula 2^n to get the squary.

Ex. If we have n number of variables will have 2^n squares thus, a row of a truth table corespondes to a square in K- MAP.

Date 30.11.2022
Page 19

METHOD TO SIMPLIFY THE K-MAP EQUATION

① TWO VARIABLES in K-MAP :

The Two variables K. map consist of Four cells with formula (2^n).

x \ y	y'	y
x'	x'y'	x'y
x	xy'	xy

	0	1
0	0 0	0 1
1	1 0	1 1

FIG OF 2 VARIABLE

② THREE VARIABLE K. MAP :

The Three variable K. map consist of eight cells by the formula 2^n to get the squares.

x \ yz	y'z'	y'z	yz	y'z
x'	x'y'z'$_0$	x'y'z$_1$	x'yz$_3$	x'y'z$_2$
x	xy'z'$_4$	xy'z$_5$	xyz$_7$	xy'z$_6$

	00	01	11	01
0	000$_0$	001$_1$	011$_3$	001$_2$
1	100$_4$	101$_5$	111$_7$	101$_6$

FIG OF 3 VARIABLES

Date 30.11.2022
Page 20

FOUR VARIABLE K·MAP :

Four variable K map consist of 16th cell by the formula 2^n to get the squares.

XY \ ZW	z'w'	z'w	zw	zw'
x'y'	0000 (0)	0001 (1)	0011 (3)	0010 (2)
x'y	0100 (4)	0101 (5)	0111 (7)	0110 (6)
xy	1100 (12)	1101 (13)	1111 (15)	1110 (14)
xy'	1000 (8)	1001 (9)	1111 (11)	1010 (10)

FIG OF 4 VARIABLES

Date 1.12.2022

RULES TO SIMPLIFICATION OF K·MAP

① Draw K·map from the boolean function:

② Make groups of ones (1)

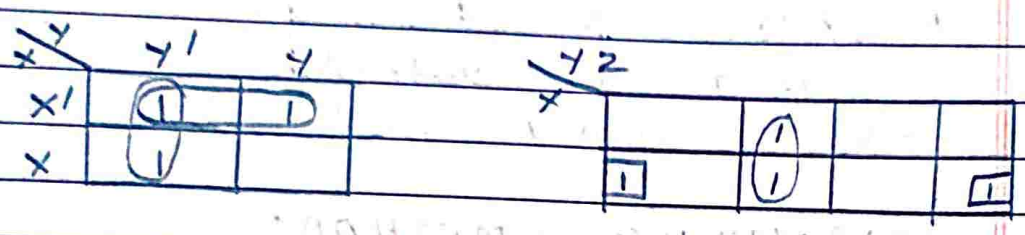
③ Write terms for each groups.

④ Write the final minimized expressions.

GROUPING METHOD:

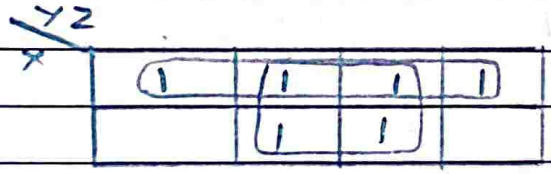
① PAIR METHOD:

This is a group of two adjusent ones in a single row or column e.g.



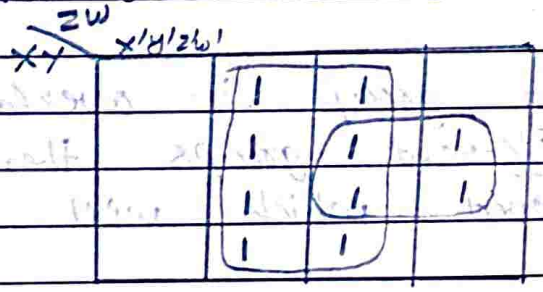
② QUARD METHOD:

Group of four continous ones in a single row or column either in horizontal or verticle form.



③ OCTAL METHOD:

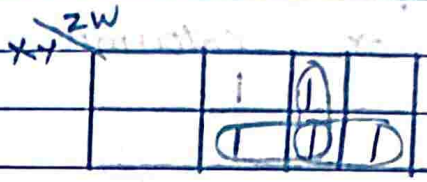
Group of eight ones is called Octal method.



Date 1·12·2022
Page 22

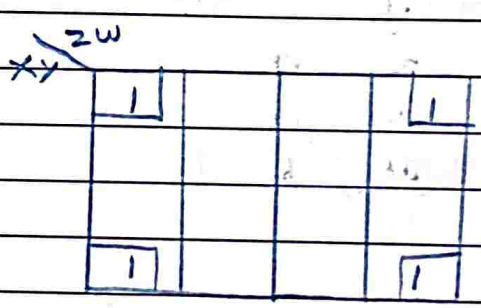
④ OVERLAPPING METHOD:-

While making groups we must for biggest groups. We can overlap the same ones or group for over lapping.



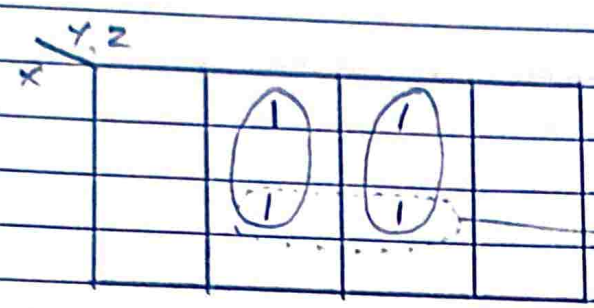
⑤ MAPROLLING METHOD:

When four ones are present at the edge of at edge of the cells or make the group by rolling.



⑥ REDUNDENT METHOD:-

When any group is overlapped with the two different groups then that group will be removed which will overlap the others.



Q. Draw the K-map for following exp:

If $(a,b,c,d) = \Sigma(0,4,8,12)$ and simplify the equation.

ab \ cd	c'd'	c'd	cd	cd'
a'b'	0	1	3	2
a'b	4	5	7	6
ab	12	13	15	14
ab'	8	9	11	10

$(a\ b\ (D) = \Sigma(a'b'c'D' + a'b\,c'd' + ab'cd' + ab'\,c'd'$
$= \Sigma(c'd')$

Q. Simplify $F(x,y,z,w) = \Sigma(0,2,7,8,10,15)$

P.T.O.

Date 1.12.2022
Page 24

0, 2, 7, 8, 10, 15

xy \ zw	00	01	11	10
00	1 (0)	1	1	1 (2)
01	4	5	1 (7)	6
11	12	13	1 (15)	14
10	1 (8)	9	11	1 (10)

$F(XYZW) = \Sigma(X'Y'Z'W' + XY'Z'W' + XY'Z'W' + XYZW)$

$= Y'W'Y'W'Y'W'Y'W'$

1

$F(XYZW) = \Sigma(X'YZW + XYZW)$

$= \Sigma(YZW)$ Ans pair

(3) Obtain a simplified form. $F \Sigma(U,V,W,Z) = \Sigma(0,1,3,4,5,7,9,10,11,13,15)$

UV \ WZ	W'Z'	W'Z	WZ	WZ'
U'V'	1 (0)	1 (1)	1 (3)	2
U'V	1 (4)	1 (5)	1 (7)	6
UV	12	1 (13)	1 (15)	14
UV'	8	1 (9)	1 (11)	1 (10)

$F(UVZW) = \Sigma(U'V'W'Z' + U'V'W'Z + U'VW'Z' + U'VW'Z + U'V'WZ + U'V'WZ + U'VWZ + UVW'Z + UV'W'Z + U'V'WZ + UVWZ + UVWZ + UV'WZ + UV'WZ + UV'W'Z'$

Ans $= U'U'U'W'ZZU'ZU'ZU'ZU'ZZVW'ZU'ZZZZZV'$

Q. $F(a,b,c,d) = \Sigma(0,2,4,5,7,8,10,12,13,15$

ab \ cd	c'd'	c'd	cd	cd'
a'b'	1 0	1	3	1 2
a'b	1 4	1 5	1 7	6
ab	1 12	1 13	1 15	14
ab'	8	9	11	1 10

$F(abcd) = \Sigma\ a'b'c'd' + a'bc'd' + a'bc'd' + a'bc'd + ab c'd' + ab c'd + a'bc'd + a'bcd + ab c'd + abcd + a'b'cd' + ab'cd'$

$F(abcd) = \Sigma\ c'c'a'bbba'd'd'$ Ans

Q. $F(x,y,z,w) = \Sigma(1,3,4,5,7,11,12,13,15)$ and also write circuit diagram.

xy \ zw	z'w'	z'w	zw	zw'
x'y'	0	1 1	1 9	2
x'y	1 4	1 5	1 7	6
xy	1 12	1 13	1 15	14
xy'	8	9	1 11	10

$F(xyzw) = \Sigma\ x'y'z'w + x'y'zw + x'y'zw + x'yzw + xyzw + xy'zw + x'yz'w' + x'yz'w + xyz'w' + xyz'w + x'yz'w + x'yzw + xyz'w + xyzw$

$F(xyzw) = \Sigma\ w\ ww\ ww\ wy\ x'yw\ w\ xw\ x'yzw\ yw$ Ans

Date 10.12.2022
Page 86

BINARY ADDITION

(i) $1101111 + 110011$

$$
\begin{array}{r}
1\ 1\ 0\ 1\ 1\ 1\ 1 \\
+\ 1\ 1\ 0\ 0\ 1\ 1 \\
\hline
1\ 0\ 1\ 0\ 0\ 0\ 1\ 0
\end{array}
$$

(ii) $10100\ 1111 + 1110\ 110$

$$
\begin{array}{r}
1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 1 \\
+\ \ \ 1\ 1\ 1\ 0\ 1\ 1\ 0 \\
\hline
1\ 1\ 1\ 0\ 0\ 0\ 1\ 0\ 1
\end{array}
$$

(iii) $1011011 \cdot 1101 + 10011 \cdot 011$

$$
\begin{array}{r}
1\ 0\ 1\ 1\ 0\ 1\ 1 \\
+\ \ \ 1\ 0\ 0\ 1\ 1 \\
\hline
1\ 1\ 0\ 1\ 1\ 1\ 0
\end{array}
\qquad
\begin{array}{r}
1\ 1\ 0\ 1 \\
0\ 1\ 1 \\
\hline
1\ 0\ 0\ 0\ 0
\end{array}
$$

$1101110 \cdot 10000$

Date 10.12.2022
Page 27

BINARY SUBSTRACTION

RULES :

X	Y	Difference	Borrow
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

(i) 111001 - 1101

```
 111001
-  1101
------
 101100
```

(ii) 100111 - 10110

```
 100111
- 10110
------
 010001
```

Q. Convert decimal number $(53.39)_{10}$ into Hexadecimal.

A. $(35.6307)_{16}$ Ans

16	53	R
	3	5

FRACTION PART

$0.39 \times 16 = 6.24 = 6$
$0.24 \times 16 = 3.84 = 3$
$0.84 \times 16 = 13.44 = 13$
$0.44 \times 16 = 7.04 = 7$
6 3 13 7
6 3 D 7

Q. Convert $(0.52)_{10}$ to its binary equivation (8th Places).

⇒ Int 0 Exp .52 $= (0.10000101)_2$

FRACTION PART

$0.52 \times 2 = 1.04 = 1$
$0.04 \times 2 = 0.08 = 0$
$0.08 \times 2 = 0.16 = 0$
$0.16 \times 2 = 0.32 = 0$
$0.32 \times 2 = 0.64 = 0$
$0.64 \times 2 = 1.28 = 1$
$0.28 \times 2 = 0.56 = 0$
$0.56 \times 2 = 1.12 = 1$

Q. Convert (0.125) to its binary equivalent.

Int	Exp	= $(0.0100)_2$
0	.125	

FRACTION PART

$0.125 \times 2 = 0.250 = 0$
$0.250 \times 2 = 0.500 = 0$
$0.500 \times 2 = 1.000 = 1$
$0.000 \times 2 = 0 = 0$

Q Convert $(48.65)_{10}$ to its octal conversion.

Int	Exp	= 6.5146
48	.65	

8	48	
	6	0

FRACTIONAL PART

$0.65 \times 8 = 5.2 = 5$

$0.2 \times 8 = 1.6 = 1$

$0.6 \times 8 = 4.8 = 4$

$0.8 \times 8 = 6.4 = 6$

Date 16.12.2022

BINARY MULTIPLICATION :

RULES FOR MULTIPLICATION :

X	Y	RESULT
0	0	0
0	1	0
1	0	0
1	1	1

X	Y	SUM	CARRY
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Date 16.12.2022
Page 30

NUMBER SYSTEM :

A NUMBER system is a writing system for expressing numbers that is a mathematical notation for representing numbers of a given set.

There are four types of number system.

(i) Decimal number system
(ii) Binary number system
(iii) Octal number system
(iv) Hexa decimal number system

(i) DECIMAL NUMBER SYSTEM :

In a decimal number system 10 digits from 0 to 9.

The base of this number is 10^{th}.

e.g. 2345 $2 \times 10^3 + 3 \times 10^2 + 4 \times 10^1 + 5 \times 10^0$

1000^{th} position, 100^{th} position, 10^{th} position, Unit position

Here the powers of 10 indicates the relative position of the digit of the number from "right to left". They may be called the place value of the digit.

A Decimal fraction may be also expressed

e.g. $45.23 = 4\times10^{1} + 5\times10^{0} + 2\times10^{-1} + 3\times10^{-2}$

Tenth position | Unit position | 1/10th position | 1/100th position

(ii) BINARY NUMBER SYSTEM :

The binary number system has two digits that is 0 and 1. computer use binary system because the electrical devices can understand only on (1) and off (0) condition.

Thus, binary number can be constructed just like decimal number except that the base is (2)

e.g. $(101) = 1\times2^{2} + 0\times2^{1} + 1\times2^{0}$
$= 4 + 0 + 1$
$= 5$

Q. BINARY TO DECIMAL CONVERSION

(i) $(75)_{10}$

$\Rightarrow$ 1 0 10 11

(ii) $(135)_{10}$

$\Rightarrow$ 1000 111

(iii) $(222)_{10}$

$\Rightarrow$ 11 0 1111 0

(iv) $(1111\ 001)_2$

$\Rightarrow (121)_{10}$

(iii) The OCtal NUMBER SYSTEM :

In Octal number system the base is (8).

Thus, eight (8) Symbol can be informed by the numbers by (0-7).

e.g. $(25)_8 = (21)_{10}$

(iv) HEXA DECIMAL NUMBER SYSTEM:

The hexa decimal number system has a base '(16)' and the combination of the digits are from (0-9) and 10, 11, 12, 13, 14, 15 as per the hexa decimal but in place of 10 we have to write (A), 11-B, 12-C, 13-D, 14-E, 15-F respectively.

eg: $(3V)_{16} = (\ )_{10}$
$= (49)_{10}$

NUMBER BASE CONVERSION

(i) DECIMAL TO BINARY CONVERSION:

(a) Convert the decimal $(81)_{10}$ to binary.

$(81)_{10} - (\ )_2$

$\Rightarrow 1\ 0\ 1\ 0\ 0\ 0\ 1$

(b.) Convert $(1100101)_2$ into decimal number.

$(1100101)_2 - (\ )_{10}$

$\Rightarrow (101)_{10}$

(c.) Convert $(11011O)_2$ into decimal number.

$(11011O)_2 - (\ )_{10}$

$\Rightarrow (54)_{10}$

(d.) BINARY FRACTIONS

Convert $(11.111)_2 - (\ )_{10}$

$(11.111)_2 - (\ )_{10}$

$\Rightarrow$ 3 (0.875)

Convert (1011.1101) into decimal

$(1011.1101) - (\ )_{10}$

$= 10.9375$ (11.8125)

Date 08.12.2022
Page 35

BINARY TO OCTAL CONVERSION

(a) $(110\,111)_2 - (\;)_8$

1st step → Convert into decimal

2nd step → AFTER converting divided into 3 to convert into Octal.

SHORTCUT

1. Made a group of 1's and 0's of three from left to right.

$(110\;\;111)_2 \qquad (\;)_8$

110 - 6
111 - 7

$\Rightarrow (67)_8$

1s compliment no.

(i) BINARY ADDITION
(ii) BINARY SUBSTRACTION
(iii) BINARY MULTIPLICATION
(iv) BINARY DIVISION

BINARY ADDITION :

Binary Addition is very much similar and simpler than decimal addition only four possible addition of Bit are possible to calculate the equation as shown in the table :

When we add we start with the least significant bit depending upon the values x and y of bits to be added : We would have a sum and may have a carry for the next stage of addition.

TABLE OF BINARY ADDITION (HAIF ADDER)

X	Y	SUM	CARRY
1	0	1	0
0	0	0	0
1	1	1	0
0	1	1	1

(a) 7+5 = 12

111 + 101

```
  111
  101
 1100
```

1100 = 12

(b) 1101 + 10001

```
 10001
  1101
 11110
```

11110 = 30

(c) 110111 + 101100

```
 110111
 101100
1100011
```

1100011 = 99

Date 08.12.2022
Page 38

CONVERT 1st COMPLIMENT NO

In the first compliment system the representation of persitive numbers is identified to theal used in the sign magnitude system (That is express a number by a sign forming by the magnitude). It consist of a sign bit in the left most position followed by the magnitude bits. In this conversion the Ones is converted into 0. and zero is converted into 1.

$(15)_{10}$ $(\ )_2$

$(1111)_2 \rightarrow (0000)_2$

2S COMPLIMENT :

The 2s compliment representation of a negative number is obtained by adding 1 to the 1s compliment of that number thus we can get the 2's compliment number by followed the rules of addition.

$(15)_{10} - (1111)_2 \rightarrow (0000)_2 \rightarrow (0001)_2$

BINARY DIVISION :

RULES

$0 \div 0 = 0$
$1 \div 1 = 1$

Q. Convert the binary division of 101101 by 101.

```
      1001
101 | 101101
       101
      -----
        0010
         101
        ----
          000
```

⇒ 1001

Q. Convert the binary division of 11011 by 101.

```
      101
101 | 11011
       101
      -----
        0011
         101
        ----
          010
```

⇒ 101

Date 20.12.2021
Page 42

DIGITAL CIRCUIT :

DIGITAL CIRCUIT :

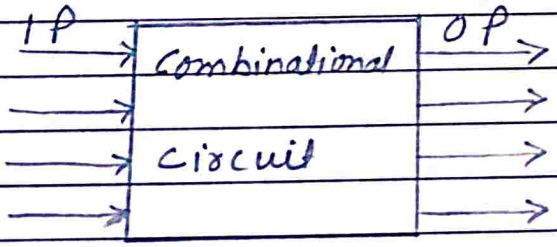
It is a circuit by which electronic current will supply to work the electronic device. It is divided into two parts.

(i) Combinational circuit
(ii) Sequential circuit

COMBINATIONAL CIRCUIT

Its output is depend only on present input and memory units are used to store the current. That is it produce the output same as they input provided.

BLOCK DIAGRAM



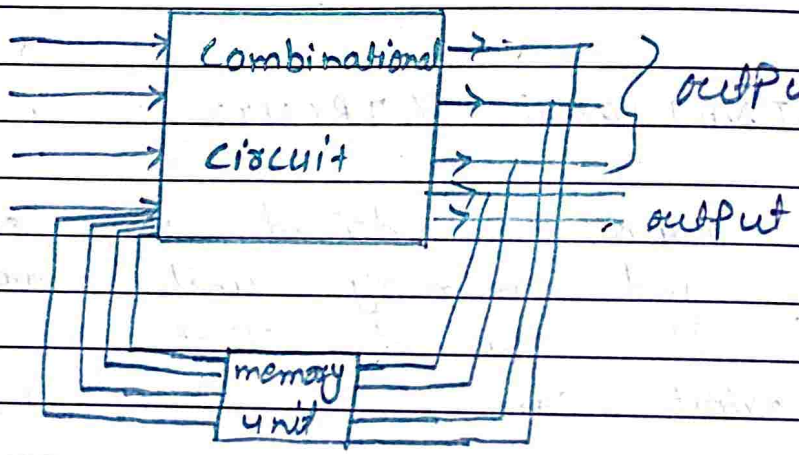
Eg. Adder, Encoder, decoder, multiplexer, demultiplexer, comparator.

Date 20.12.2022
Page 43

SEQUENTAL CIRCUIT:

These are the circuit which have designed with the help of combinational circuit and added a memory unit to provide the flip-flop motion into the circuit. Output of sequental circuit depends upon processed output as well as previous output.

Block diagram



Eg. flip-flop
Counters
Registers

Half Adder :

What is adder ?

An adder is a device that can add two binary digits or more it is a type of digital circuit that perform the operations of addition. It is mainly designed for the addition of binary number but they can be used in various others applications like. Binary code decimal and address decoding.

There are two types of adderss:

One is Half adder and another one is known as Full adder.

(i) Half adder :

There are two inputs and two outputs in a half adder. inputs are named as A and B and the output are named as sum and carry. The sum is denoted as "X-OR" of the input A and B. Carry is denoted as "AND" of the input "A and B" with the help of Half adder, We can design circuit that is

P.T.O.

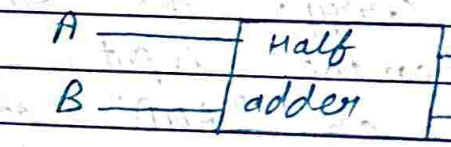
Capable of performing simple addition with the help of performing simple addition with the help of logic gates.

Table of Half adder :

A	B	SUM	CARRY
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

These are the least possible signal bit combination but the result will always shown by the help of sum and carry. Therefore, we have to combine two values to get the result of Half adder.

Basic diagram of Half adder :

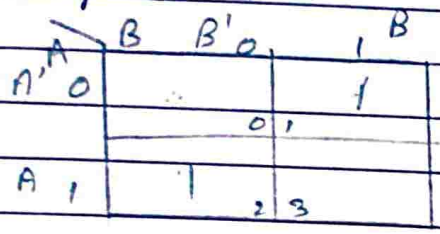


Date 18.1.2023
Page 48

Truth table :

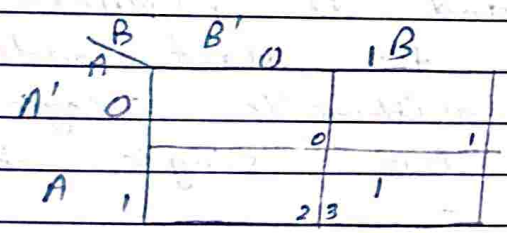
A	B	SUM	CARRY
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

K- map :



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

Fig of Sum is X - OR



CARRY = AB

Fig of carry

P. T. O.

Circuit Diagram :

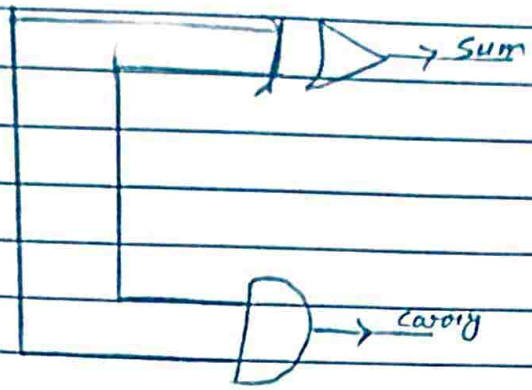


Fig of circuit diagram

Date - 19.01.2023

Full adder :

The Full adder is the co-ordination of three variables and it produces sum and carry as like the Half adder. Full adder is an arithmetic logic circuit design to add two single bit with a carry.

The Full adder is a little more difficult to impliment than a Half adder. The main difference between a Half adder and full adder is that the Full adder has three input and two output. The Full adder is the two input A and

and third input is a carry input. It is denoted by Cin. The output carry is desinated as Cout and the normal output is desinated as S (sum).

The truth table of the Full adder circuit is as shown:

A	B	Cin	sum	Carry	
0	0	0	0	0	
0	1	0	1	0	
0	0	1	1	0	
0	1	1	0	1	
1	0	0	1	0	
1	1	0	0	1	
1	0	1	0	1	
1	1	1	1	1	invalid output

The output sum is an x-OR between the input A and the Half adder sum output B. -

The C out will be true only if ~~any~~ of the two inputs out of the three are high or at logic one. Thus, a Full adder circuit can be implemented with the help of two added circuit. The First Half

P.T.O.

adder circuit will be used to add A and B. produce A and B to produce a partial sum. The second Half adder logic can be used to add Cin to the sum produced by the first Half adder circuit. Finally, the output S is obtained.

Logic circuit :

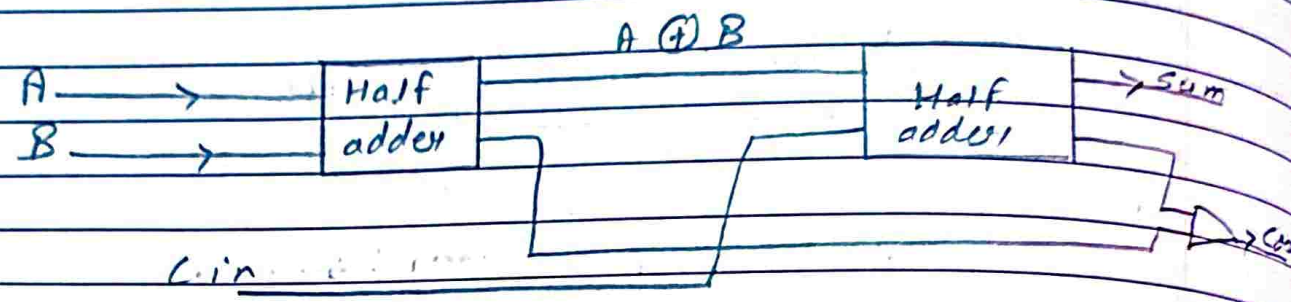


Fig. Block Diagram

The three variable K-map:

A \ B Cin	00	01	11	10
0	0	1	3	2
1	4	5	7	6

A ⊕ B ⊕ Cin

Date 19-1-2023
Page 50

A \ B Cin	00	01	11	10
0	0	1	3	2
1	4	5	7	6

$= B\,C_{in} + AB + C_{in}$

Circuit diagram :

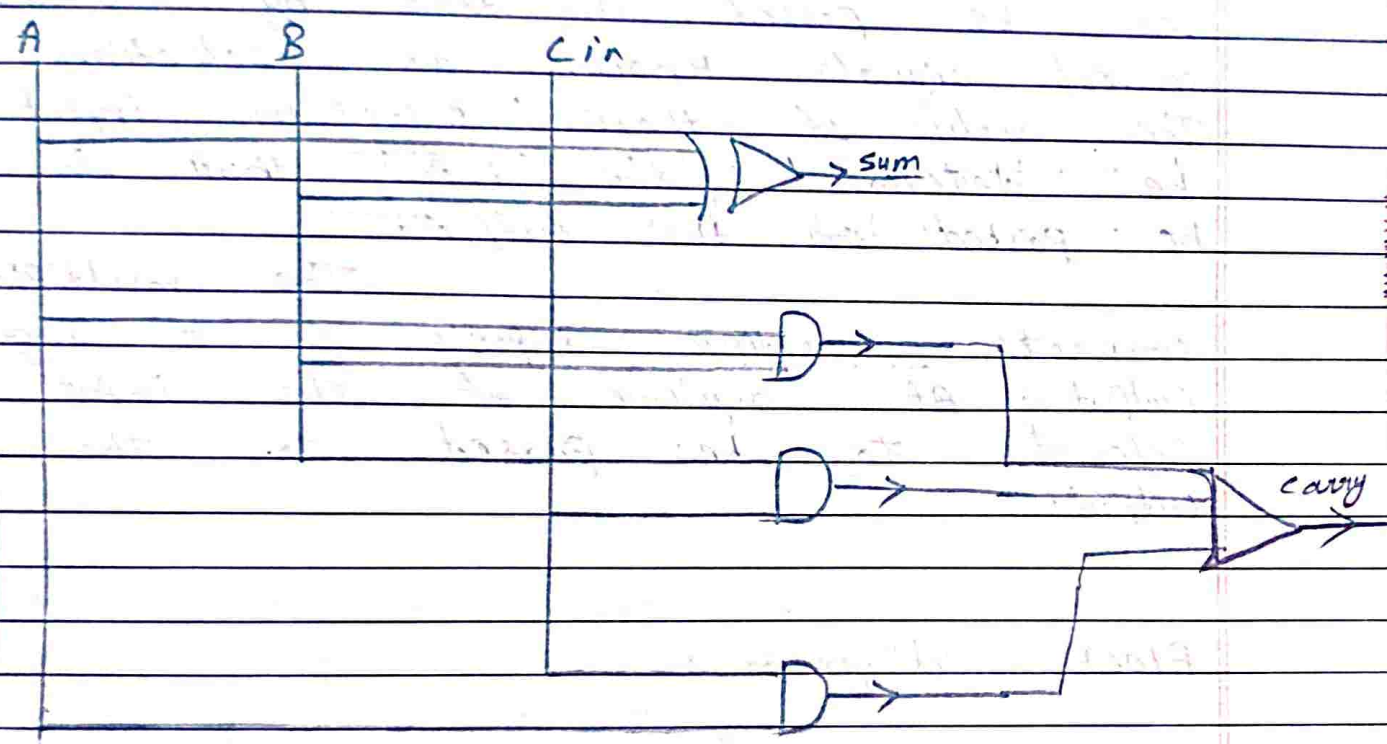


Fig of circuit diagram

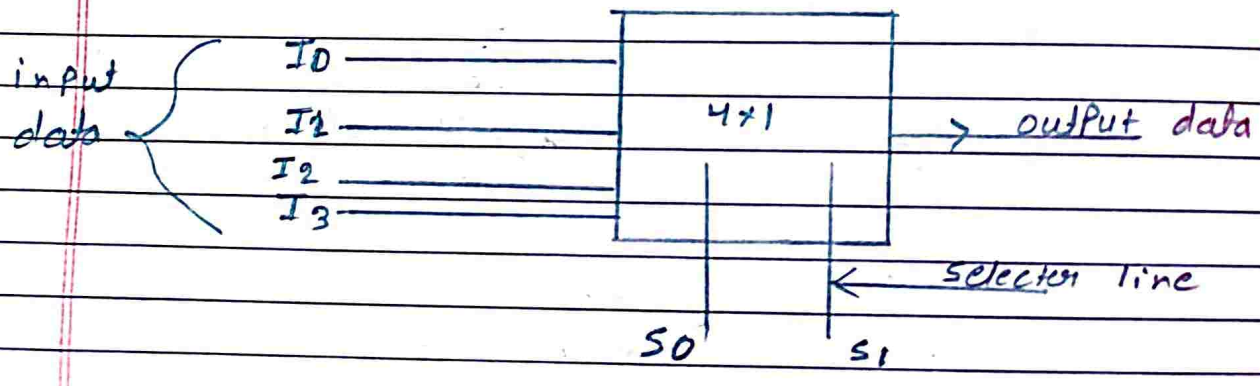
Date 20.1.2023
Page 51

Multiplexer / Multiselecter (MUX) (4×1):

A multiplexer is a combinational logic circuit which accepts many inputs but selected only one 1 input to be passed on the output. It is sometimes refered to as a data selection or selecter since it selects only one of the inputs. the selection of the input to be passed is done by a set of control signals known as selection input. The value of these selection input will be determind the input that is to be passed on the output.

The multiplexer connects multiple inputs to a single output. At anytime of the input is selected to be passed to the output.

Block diagram:



This represents 4x1 multiplexer. There are four input lines labeled as I_0, I_1, I_2, I_3. One of these line is selected to provide to the output signal 0 (zero). To select one of the four possible inputs a 2 bits selection code is needed and this is implemented as two selection lines labeled as S_0 and S_1.

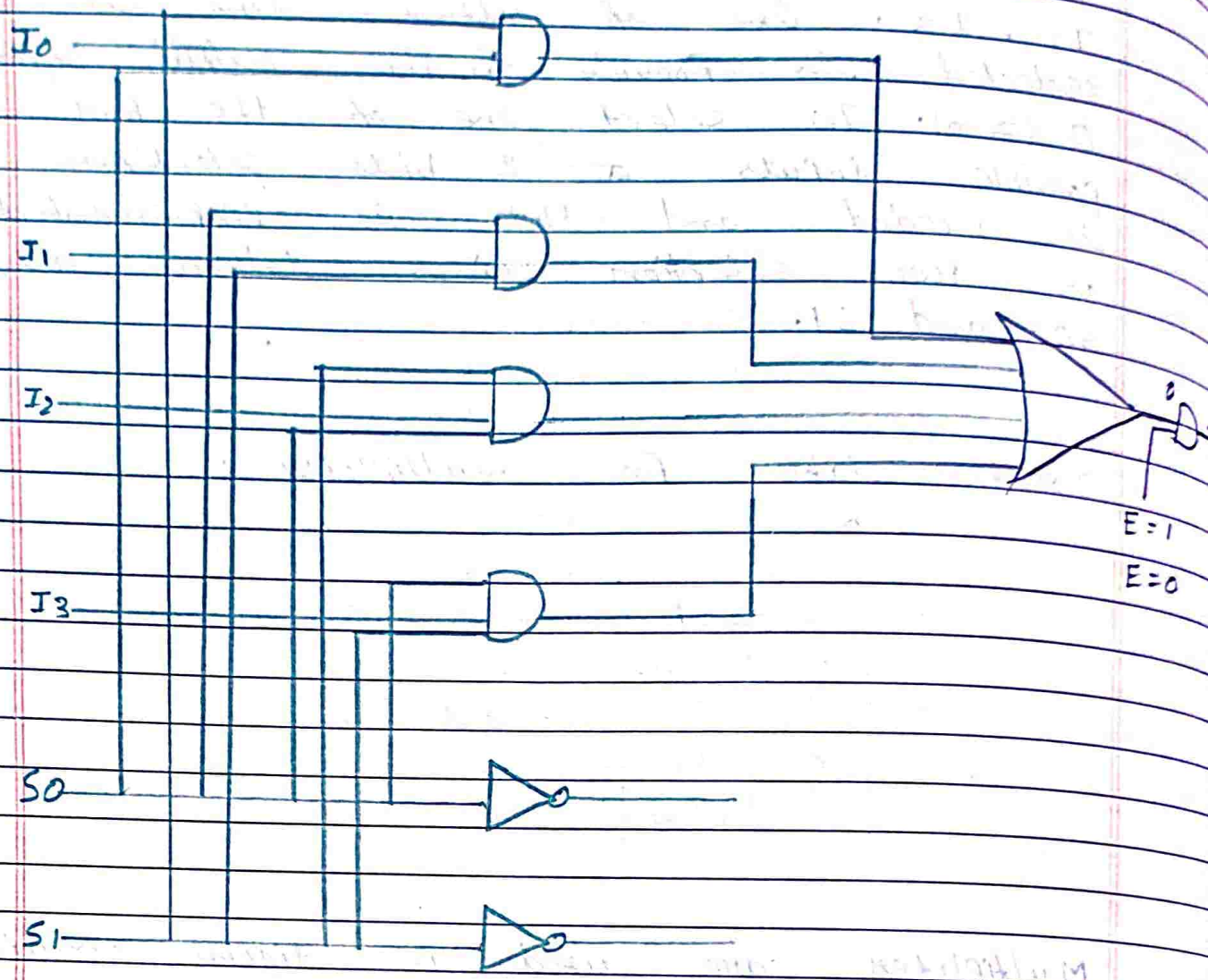
Truth table for multiplexer :

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

Multiplexer are used in digital circuit to control signals and data routing with the help of truth table we finally define the logical diagram of the multiplexer the current which is flowing always depend on the enable lines.

Date 20.1.2023
Page 53

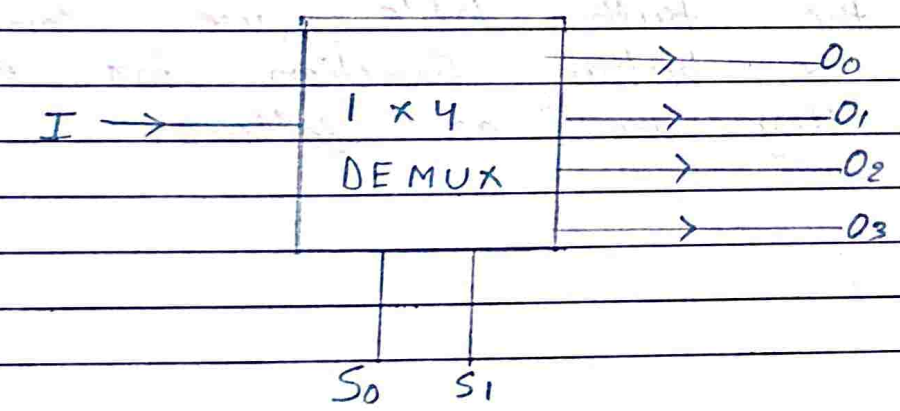
Logical Diagram:



DEMULTIPLEXER :

A Demultiplexer is a combinational circuit that performs the reverse operation of multiplexer. It takes one single input and passes this to the output number specified by the select input since, there are 'n' selection lines these will '2^n' possible '1or' possible combination of zeros and ones so each combination can select only one output. It is also called DE-MUX or Data Distributor.

We are trying to design '1x4' Demultiplexer in which it has 1 input 'I', TWO (2) selection lines as S0 & S1 and four outputs O_0, O_1, O_2, O_3 which shows the outputs of each stages, The Block diagram are as shown:



The single output 'I' will be connected to one of the four outputs from O_0 to O_3 based on the values of selection lines of S_0 and S_1.

The truth table of 1×4 DE-MUX as shown:

S_0	S_1	Input	Output
0	0	1	O_0
0	1	1	O_1
1	0	1	O_2
1	1	1	O_3

OR

S_0	S_1	Input	OUTPUT			
			O_0	O_1	O_2	O_3
0	0	1	1	0	0	0
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	1

From the truth table we can directly write the boolean function for each output they are as follows:

Date 31.1.2023
Page 56

$O_0 = S_0' S_1' I$
$O_1 = S_0' S_1 I$
$O_2 = S_0 S_1' I$
$O_3 = S_0 S_1 I$

We can implement these boolean function using inverters and input as AND gates the circuit diagram of 1x4 'DE - MUX' is as shown:

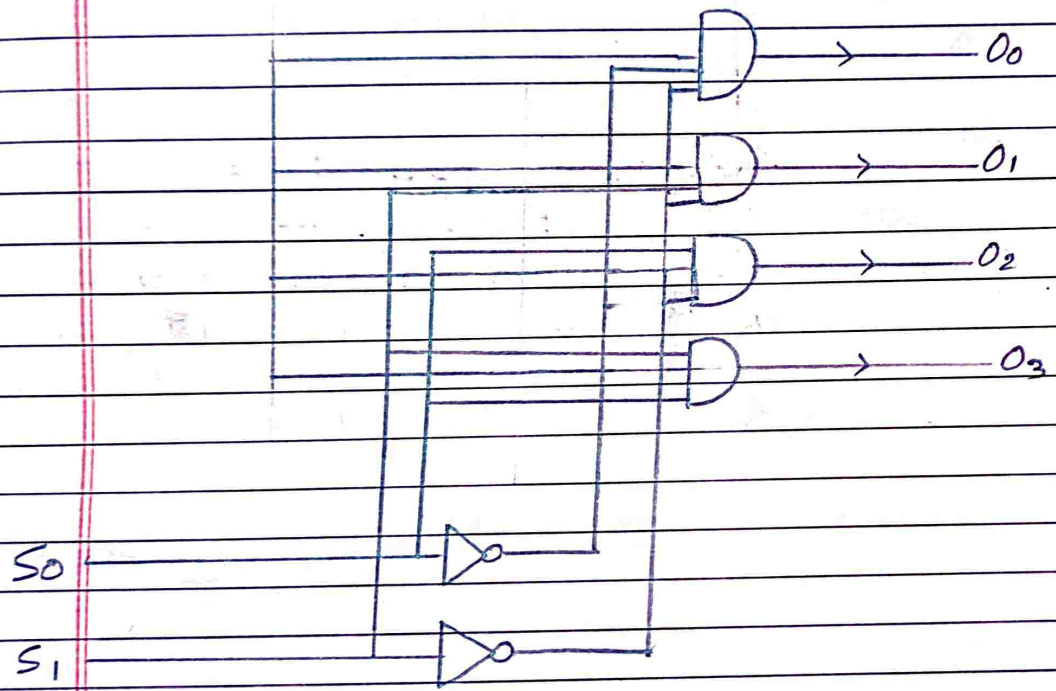
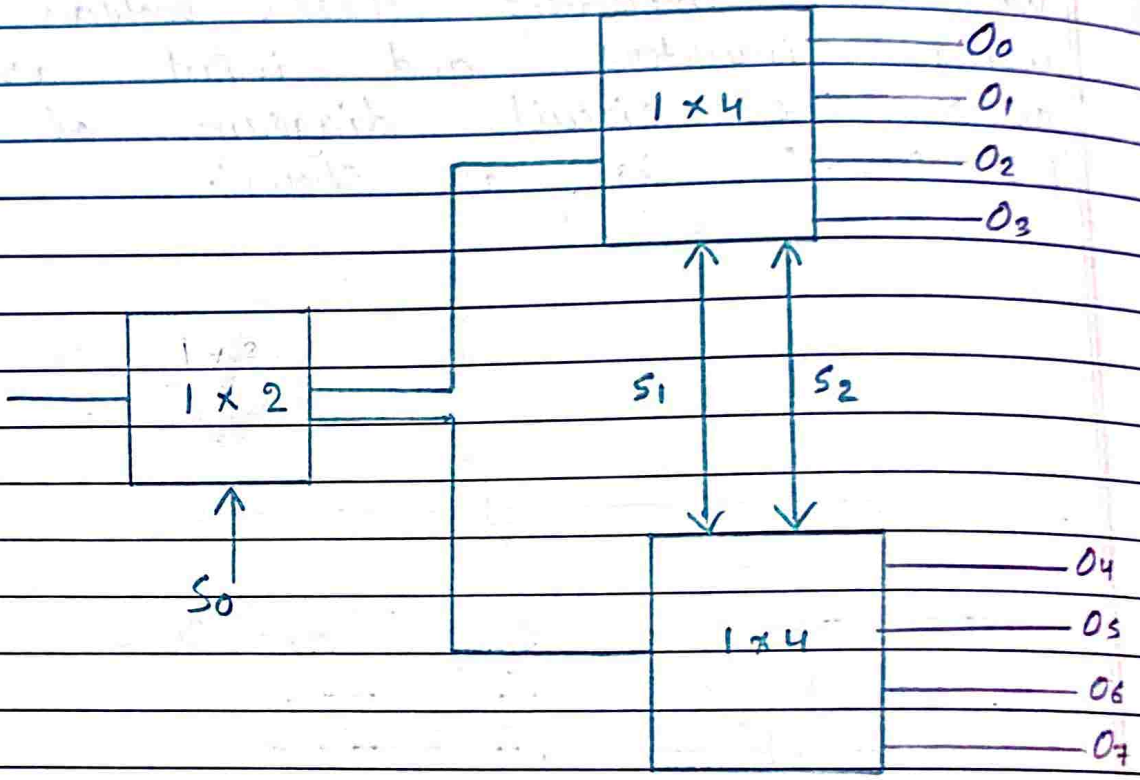


Fig of Circuit Diagram

Date 31-1-2023
Page 57

High order of DE-MUX :

1 x 8 } High - DE-MUX
1 x 16 }



Date 01.02.2023
Page 58

DE-Coder and En coder:

A De-coder is a combinational logic circuit that accets a set of inputs that represent a binary number and activates that output which corresponding to the input binary number. A De-coder 'n' inputs and (One single line) and '2^n' Output lines.

Let 2 (two) to 4 (four) De-coder has two input I_0 & I_1. and four output such as O_0, O_1, O_2, O_3. The Block diagram of 2 to 4 De-coder is as shown:

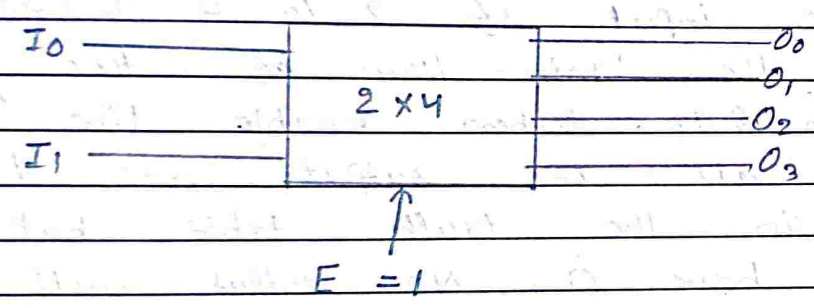


Fig of Block-Diagram

As we Know that current will flow when enable line has higher value that is I. It can not flow the current if it has a value 0, that means it is in

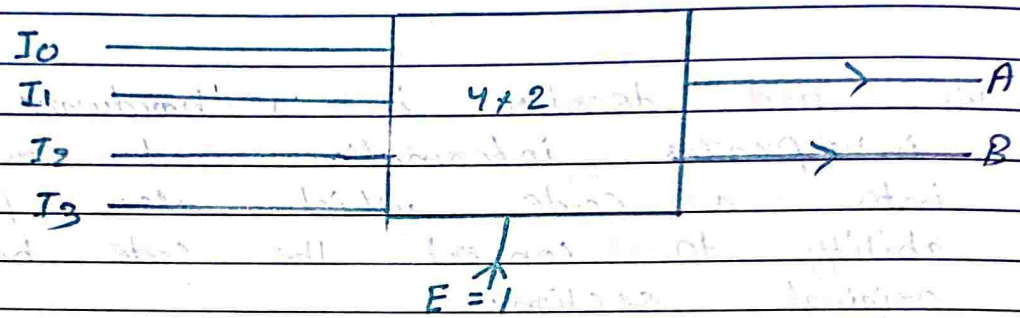
P.T.O.

which is activated at a time. At the output it display a value corresponding to the activated input.

An encoder has '2^n' input lines and 'n' output lines.

let us consider 4 to 2 Encoder which has 4 inputs, such as I_0, I_1, I_2, I_3 and 2 output A & B.

The Block diagram are as shown:



Define the truth table of the Block diagram which is as known:

INPUT				OUTPUT	
I_0	I_1	I_2	I_3	A	B
1	0	0	0	0	0
0	1	0	0	0	1
0	0	1	0	1	0
0	0	0	1	1	1

A '4×2' priority encoder has four input as I_0, I_1, I_2, I_3 and two outputs A and B. Here, the input I_3 has the highest priority where as the input I_0 has the lowest priority. In this case even if more than one input as '1' at the same time the output will be the binary code corresponding to the input which is having higher priority.

Equations of these output are as follows:

$$A = I_0' I_1' I_2 I_3' + I_0' I_1' I_2' I_3$$

$$B = I_0' I_1 I_2' I_3' + I_0' I_1' I_2' I_3$$

With the help of the equation we are going to design a circuit diagram for encoder:

P.T.O.

CLASSIFICATION OF SEQENTAL CIRCUIT

It is of two types :

(i) Syn chronious sequental circuit
(ii) ASyn chronious sequental circuit

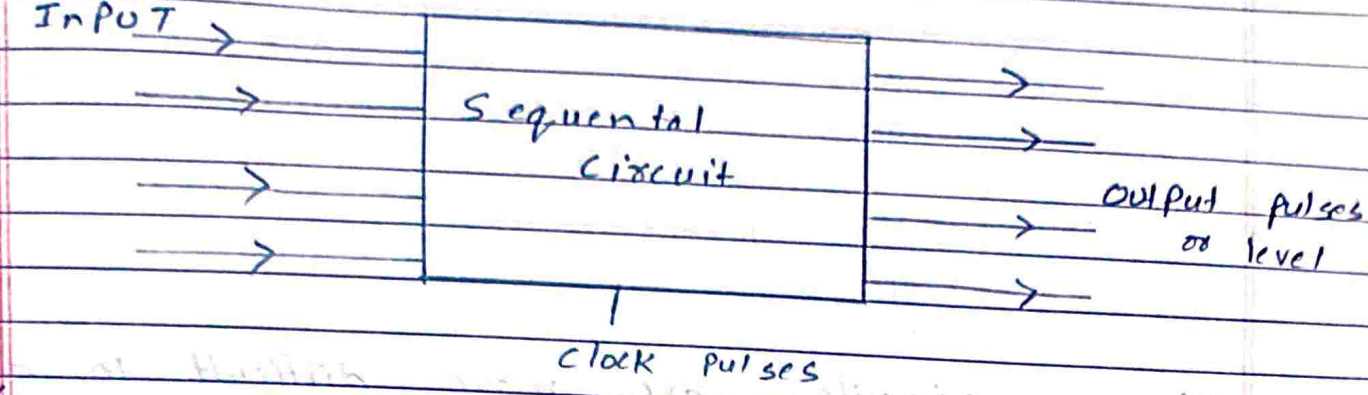
(i) SYNCHRONIOUS SEQENTAL CIRCUIT :

These circuits uses clock signal and label input (pulses). The output pulse is the same duration as the clock pulse for the clocked sequental circuit since they wait for the next pulse to arrive to perform the next clock pulse to arrive to perform next operation. So, these circuits are bit slower compared to asynchronious. Label output changes state at the & an input pulse and remains in that unit until the next input or clock pulses arrived.

We use syn chronious sequental circuits in synchronious counters, flip-flop. and in the design of (MOORE - MEALY' state managment machine.

Date 17.02.2023
Page 66

BLOCK DIAGRAM:



DATE 25.02.2023

ASYNCHRONOUS SEQENTAL CIRCUIT:

These circuits do not use in clock signal but uses the pulses of the input. These circuits are faster than Synchronous Sequental circuit because there is clock pulses and change their state immediately when there is a change in the input signal. We use asynchronous sequental circuits when speed of operation is important and independent of internal clock pulses.

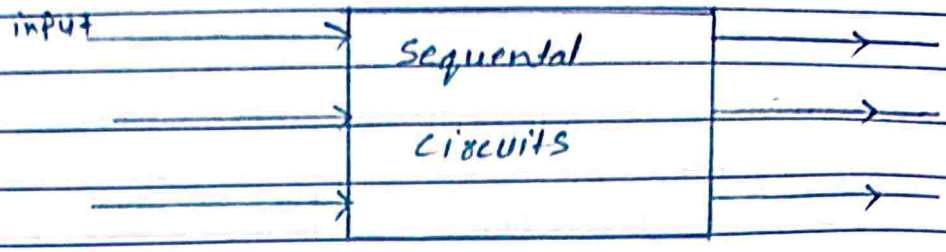


Fig of Asynchronous S.C.

These circuits are more difficult to design and output is uncertain.

FLIP - FLOP :

Flip - flop is a memory element which is capable of storing 1 bit of information and is in clocked sequental circuit.

A flip - flop has two outputs one for normal value of and other for compliment values of the bit stored in it.

A flip - flop can maintain a binary state indefinetly (as long as power is delivered to the circuit) until directed by input signal to switch. A flip- flop is also known as "by stable multivibrator".

Date 25.02.2023
Page 63

The basic flip-flop or latch are the flowing condition state of flip-flop.

A basic flip-flop circuit can be constructed from two NAND GATE OR TWO NOR GATE.

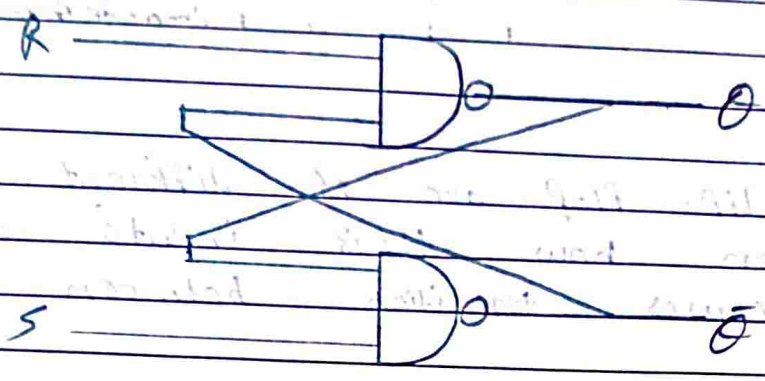
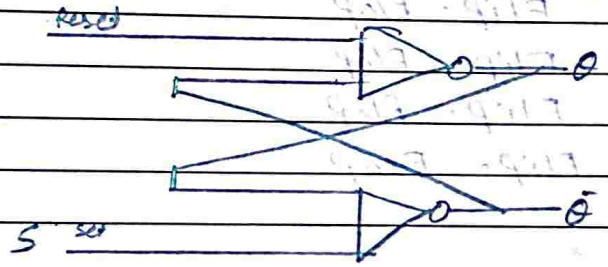
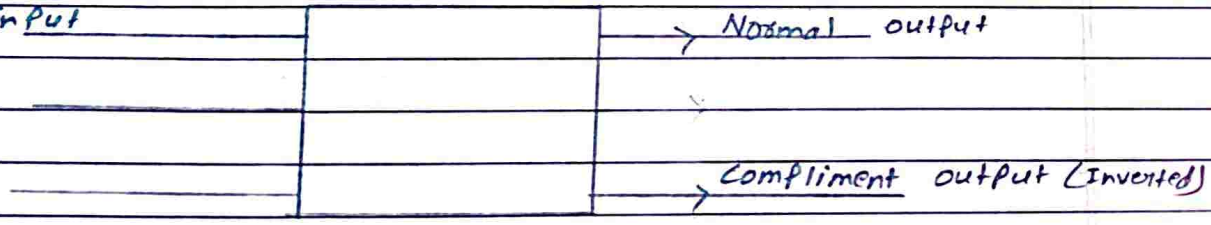


Fig of Flip-Flop circuit of NAND GATE

Fig. of flip-flop circuit of NOR GATE



BLOCK DIAGRAM:



Date 25.02.2023
Page 69

The truth table of flip-flop

R	S	Q	$\bar{Q}$	CONVERSION
0	0	x	x	No change
0	1	1	0	set
1	0	0	1	Reset
1	1	?	?	undetermined

Flip-flop are of different types depending on how their inputs and clock pulses causes transition between two states.

There are four Basic types of flip-flop

(i) SR Flip-Flop
(ii) JK Flip-Flop
(iii) D Flip-Flop
(iv) T Flip-Flop

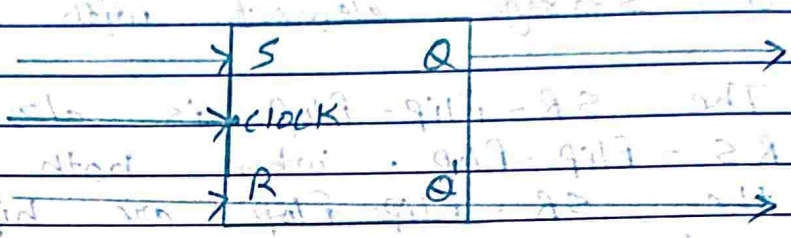
Date 28.02.2023
Page 70

D - FLIP - FLOP :

A D - flip - flop is a clocked flip - flop with a single digital input 'D'. Each time a D - flip - flop is clocked its output

The circuit is similar to the SR latch except for the clock signal and two AND gates. The SR - flip - flop circuit responds to the positive edge of the clock pulses to the inputs S and R. SR - flip - flop is used as a storage device for a single data bit. Its Block diagram will be shown as like:

BLOCK DIAGRAM OF SR- FLIP- FLOP



As we consider a Block diagram we can generate a truth table also:

SR - FLIP - FLOP TRUTH TABLE:

S and R are the two inputs to the SR - flip- flop. The Flip - flop Qn represents the state of the SR - flip- flop before applying the input and Qn+1 represents the state of the SR - flip - flop as output.

The truth table for SR-flip-flop is as shown:

SR- FLIP- FLOP TRUTH TABLE:

CLOCK	S	R	Qn+1	STATE
0	x	x	Qn	x
1	0	0	Qn	Hold
1	0	1	0	Reset
1	1	0	1	set
1	1	1	1	Invalid

Date 28.02.2023

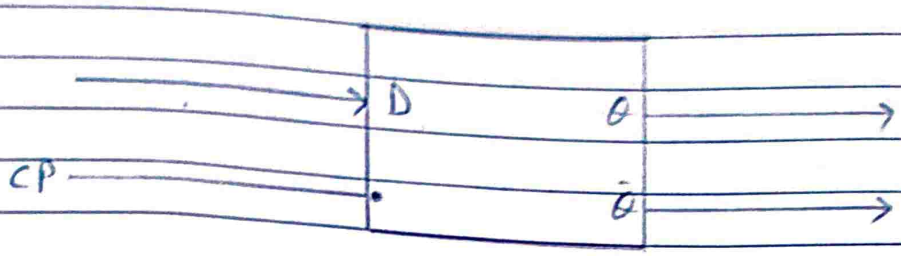
D - FLIP- FLOP:

A D-Flip-flop is a clocked flip-flop with a single digital input 'D'. Each time a D-flip-flop is clocked its output follows the state of D. The D flip-flop has only two inputs 'D' and 'CP'. The D input go percisly to the S input and its complement is used to R input (Reset input) which is at 0 the outputs of gates are at the one level and the circuit can not convert

P.T.O.

state regardless of the value of D. The D input is sampled when CP is equal to 1. If D is one the Q output goes to 1 locating the circuit in the set state.

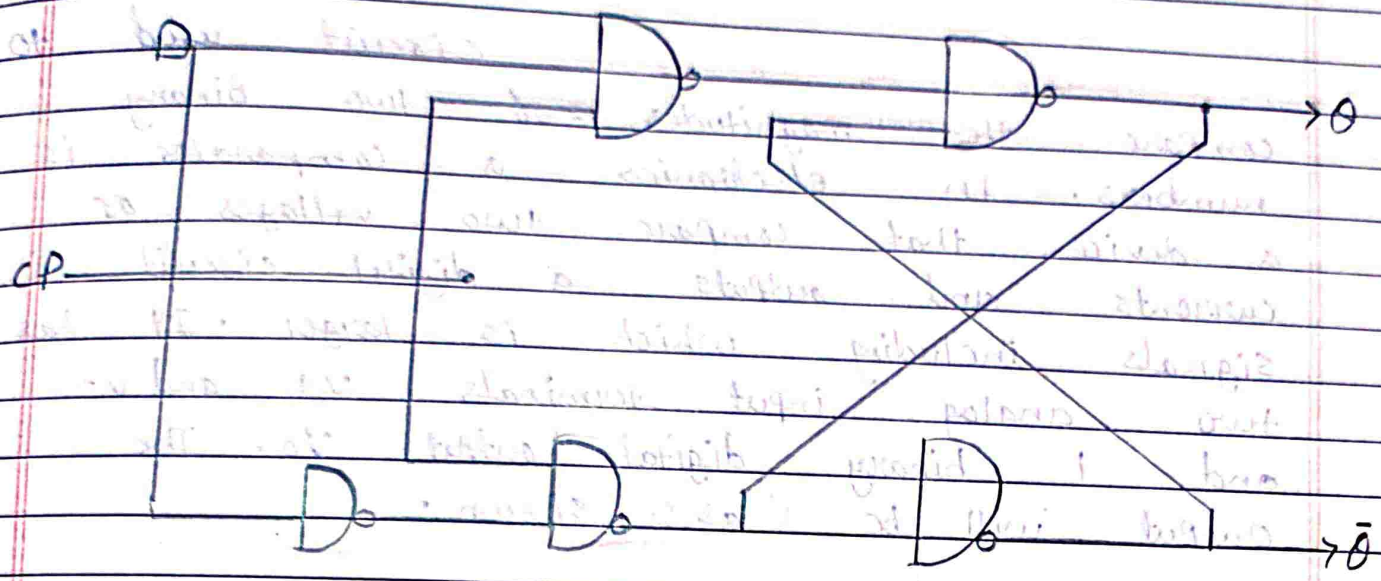
BLOCK DIAGRAM OF D-FLIP-FLOP



The D-flip-flop obtains the destination capacity to manage data into its internal storage. These types of flip-flop is known as a gated D-latch. The CP input is provided given the destination to denote that these input allows the gated latch to create applicable date entry into the circuit.

Date 28.02.2023
Page 76

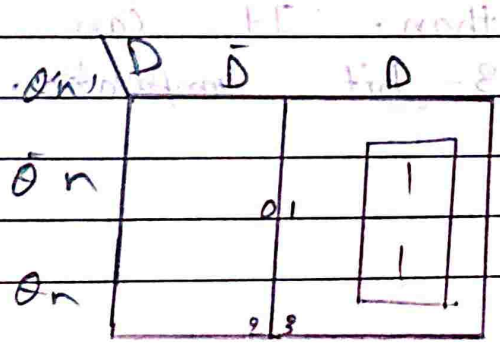
LOGICAL CIRCUIT OF D-FLIP-FLOP



TRUTH TABLE OF D-FLIP-FLOP:

CP	Q(n)	D	Q(n+1)
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

K-MAP:



$Q(n+1) = D$

COMPARATOR :

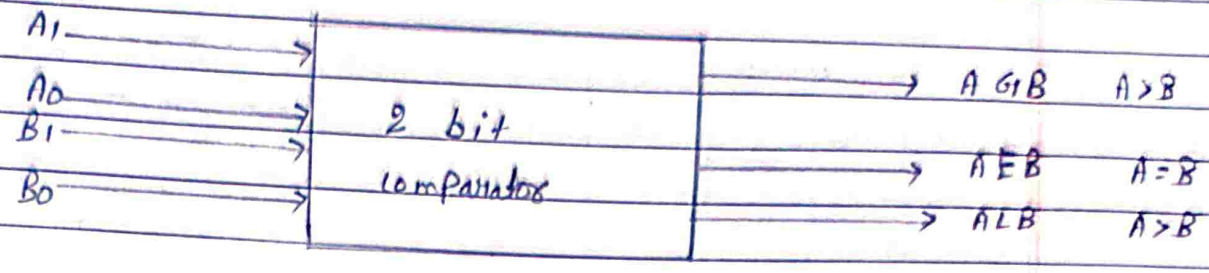
A comparator is a logic circuit used Binary compare the magnitudes of two numbers. In electronics a comparator is a device that compare two voltages or currents and outputs a digital circuit signals including which is larger. It has two analog input terminals V+ and V- and 1 binary digital output Vo. The output will be as shown:

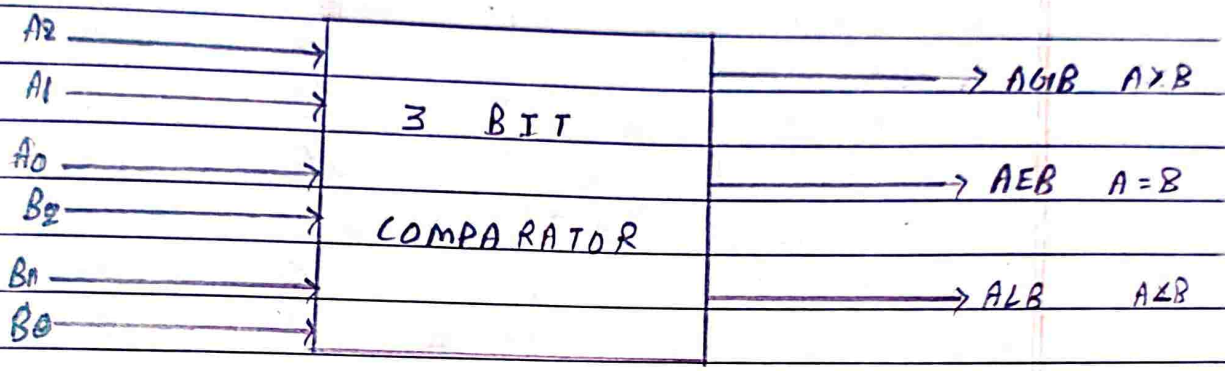
$$V_o = \begin{cases} 1 & \text{if } V+ > V- \\ 0 & \text{if } V+ < V- \end{cases}$$

A comparator consist of a specialized high-gain differential amplifier. They are commonly used in devices that measure and digitalized analog signals, such as analog to digital converts (A.D.C.s) as well as "relaxation oscillators". The comparator has three output: equal to, greater than, less than. It can be 2-bit comparator or 3-bit comparator.

Date 01-03-2023
Page 78

BLOCK - DIAGRAM OF COMPARATOR :





eg. LM 339 IC is a dedicated comparator chip

Which is design to interface with a digital ~~circuit~~ logic interface. The output is a binary state often used to interface real world signals to digital circuit.

P.T.O.

Date 01.03.2023
Page 79

TRUTH TABLE FOR 2 BIT COMPARATOR:

INPUT				OUTPUT			
A_1	A_0	B_1	B_0	AGB	AEB	ALB	
0	0	0	0	0	1	0	0
0	0	0	1	0	0	1	1
0	0	1	0	0	0	1	2
0	0	1	1	0	0	1	3
0	1	0	0	1	0	0	4
0	1	0	1	0	1	0	5
0	1	1	0	0	0	1	6
0	1	1	1	0	0	1	7
1	0	0	0	1	0	0	8
1	0	0	1	1	0	0	9
1	0	1	0	0	1	0	10
1	0	1	1	0	0	1	11
1	1	0	0	1	0	0	12
1	1	0	1	1	0	0	13
1	1	1	0	1	0	0	14
1	1	1	1	0	1	0	15

EQUATION FOR THE MAGNITUDE:

$A > B = \Sigma m(4, 8, 9, 12, 13, 14)$

$A = B = \Sigma m(0, 5, 10, 15)$

$A < B = \Sigma m(1, 2, 3, 6, 7, 11)$

Date 01.03.2023
Page 80

K. MAP FOR THE MAGNETIC EQUATION:

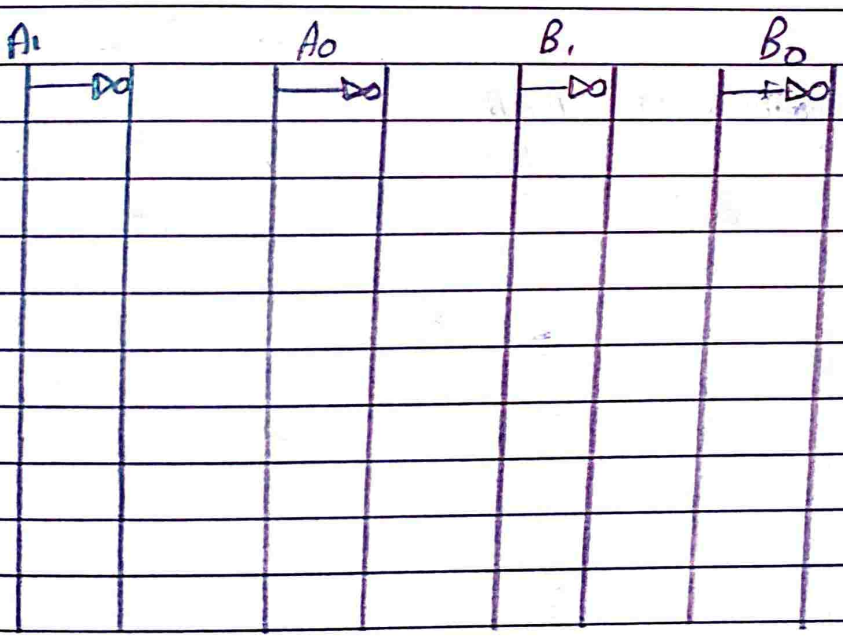
$B'B_0$ \ $A'A_0$	$\bar{A}_1\bar{A}_0$	$\bar{A}_1A_0$	A_1A_0	$A_1\bar{A}_0$
$\bar{B}_1\bar{B}_0$	0	1	3	2
$\bar{B}_1B_0$	1 4	5	7	6
B_1B_0	1 12	1 13	15	1 14
$B_1\bar{B}_0$	1 8	1 9	11	10

Fig of AGB MAGNETIC EQUATION

$$Em = (\cancel{B_1}\,\cancel{B_0}\,\cancel{\bar{A}_1}\,A_0' + B_1\,\cancel{B_0}\,\cancel{A_1}\,\cancel{A_0}' + \cancel{B_1}B_0\,A_1'A_0' + B_1B_0\,\cancel{A_1'}\,\cancel{A_0'} + B_1B_0\,A_1'\,\cancel{A_0'} + B_1\,\cancel{B_0'}\,A_1'\,A_0' + B_1\,\cancel{B_0}\,A_1'\,\cancel{A_0}.$$

$$Em = (A_0'B_1B_0A_1' + A_0'B_1B_0A_1' + B_1B_0A_1'B_1A_1'A_0' + B_1A_1'$$

CIRCUIT DIAGRAM OF AGB:



Date 1.3.2023
Page 81

K-MAP FOR MAGNETIC EQUATION

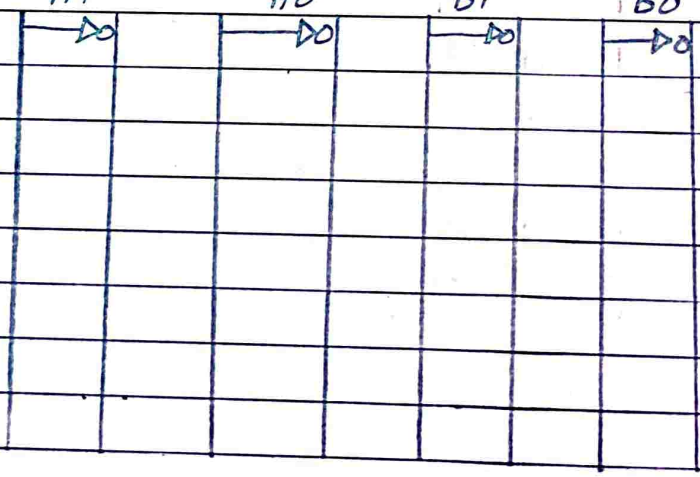
B_1B_0 \ A_1A_0	A_1A_0'	A_1A_0'	A_1A_0	A_1A_0'
$B_1'B_0'$	1 0	1	3	2
$B_1'B_0$	4	1 5	7	6
B_1B_0	12	13	1 15	14
B_1B_0'	8	9	11	1 10

Fig of AEB Magnetic equation

$$Em = (\cancel{B_1'B_0'A_1'A_0'} + \cancel{B_1'B_0}A_1\cancel{A}A_0' + \cancel{B_1B_0}\cancel{A_1A_0} + \cancel{B_1'B_0'}A_1A_0')$$

$$= A_1A_0'B_0B_0'A_1A_0'$$

CIRCUIT DIAGRAM OF AEB



Date 1.9.2023
Page 82

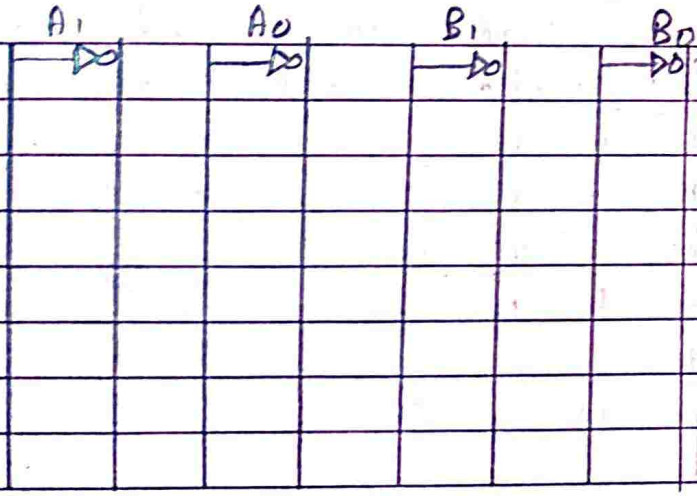
K. MAP FOR MAGNETIC EQUATION :

B_1B_0 \ A_1A_0	$A_1'A_0'$	$A_1'A_0$	A_1A_0	A_1A_0'
$B_1'B_0'$	0	1 1	3 1	2 1
$B_1'B_0$	4	5	7 1	6 1
B_1B_0	12	13	15	14
B_1B_0'	8	9	11 1	10

Fig of ALB Magnetic equation

$$\Sigma m = (B_1'B_0'A_1'A_0 + B_1'B_0'A_1A_0 + B_1'B_0'A_1A_0 + B_1'B_0'A_1A_0' + B_1'B_0A_1A_0 + B_1'B_0A_1A_0' + \cdot$$

CIRCUIT DIAGRAM OF ALB



Date 02.9.2023
Page 83

RESISTOR :

It is the part of control unit that are used for storing the instruction code from the memory. Resistor are used to quickly accept, store and transfer data and instructions that are being used immediately by the CPU. There are various types of Resistors those are used for various purpose these resistors are used for various operations while we are working on the system then these resistors are used by the CPU for performing the operations. When we gives some input to the system then the input will be stored into the resistor and when the system will gives us the result after processing then the result will also be from the resistor, Therefore we can say the resistor are the part of control unit that are used for storing the instruction code from the memory.

Basically there are many - Resistors, some of them are as follows:

REGISTOR SYMBOL	NO. OF BITS	REGISTER NAME	FUNCTION
DR	16 bits	Data Register	Holds the memory operands
AR	16 bits	Address Register	Holds the address of memory
AC	16 bits	Accumulator	Process registor
IR	16 bits	Instruction Registor	Holds the instruction code
PC	12 bits	Programme Centre	Holds the address of instruction code
PR	16 bits	Temporary Register	Holds temporary data
INPR	8 bits	Input Register	Holds the input character
OUTR	8 bits	Output Registes	Holds the output character

Date 02.3.2023
Page 83

The maximum bits hold by the register is 16 bits. The memory unit can hold 4096 word or character.

$\therefore$ 4096 words $= 2^{12} = 16$ bits

In other words we can say that a register is a group of flip-flop that stores group of flip-flop.

Flip-flop can stores 1 bit of information

2 bit $\longrightarrow$ 2 flip-flop
3 bit $\longrightarrow$ 3 flip-flop
4 bit $\longrightarrow$ 4 flip-flop

group of bit (group of flip-flop)

A register is capable of shifting its binary information to left is called shift-register. It can be represent by the block-diagram.

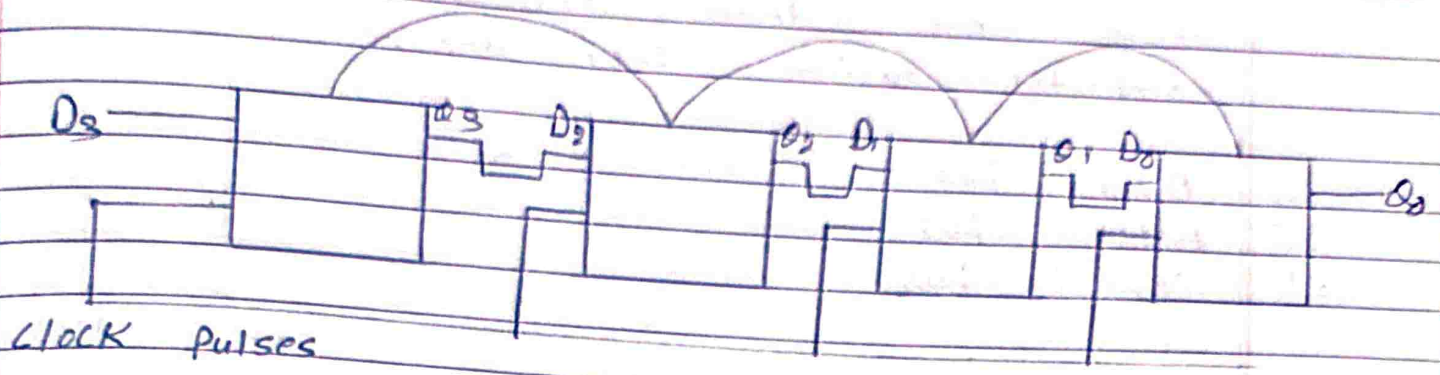


Fig of 4 bit (D- flip- flop)

Date
13.03.2023

BUS SYSTEM IN COMPUTER ARCHITECTURE

BUS SYSTEM:

The CPU must be able to communicate with all devices. The devices are connected together by a communication channel called a bus. A bus is composed of set of communication line or wires.

There are three different buses in computer system they are:

(i) Data bus
(ii) Address bus
(iii) Control bus

(i) DATA BUS:

A bus that connects major components such as: processor, memory, input/output devices to share the information. There are fifty to hundred of seprate lines and each line assigns some special function.

The most common bus is the data bus. A data bus carries data. It is an electrical path that connects the C.P.U., memory, i/o devices and secondary storage devices. The bus contain parallel group of lines. The number of lines in bus affects the speed at which the data travells b/w different components. It consist of 8, 32, 64, 128 and more seprate lines. The number of lines reffered as width of data bus. It is a bydirectional component.

13.3.2023
88

(ii) ADDRESS BUS :

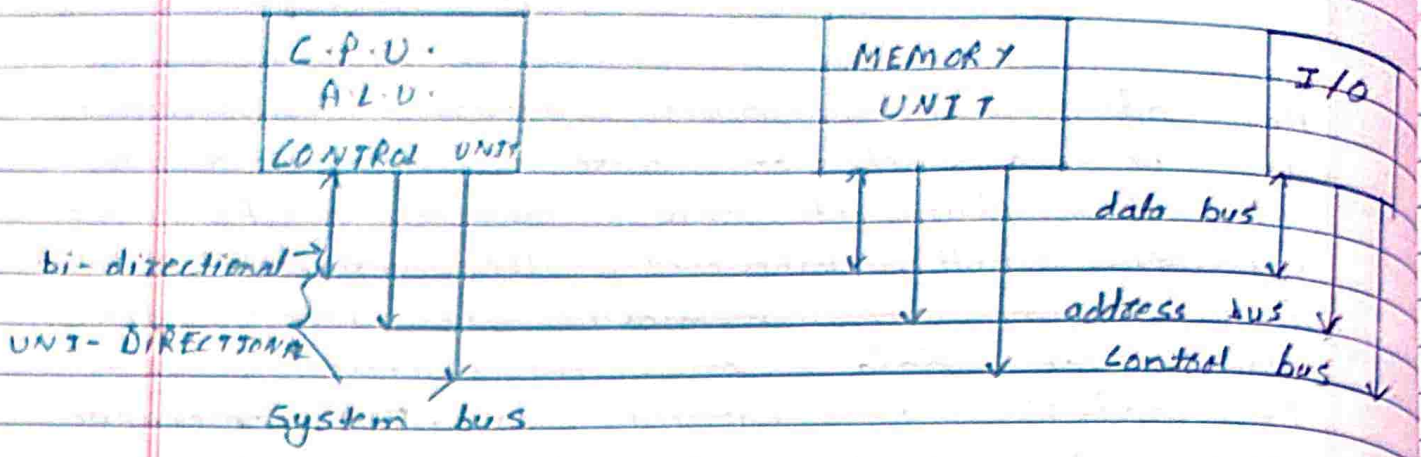
An address bus carries address information. It is a set of wire similar to the data bus but it only connect CPU and memory unit, Whenever the processor needs data from the memory it places the address of data on the address bus. The address is carried to the memory where the data from the requested address is fetched and placed on the data bus. The data bus carried to CPU. It is used to identify the source of destination of data. It is uni-directional source of data.

(iii) CONTROL BUS :

The control bus carries control information from control unit to the other units. The control information is used for directing the activities to all units. The control unit control the functioning other units such as i/o devices, secondary storage etc. It controls and timing information correctly.

P.T.O.

13.3.2023



BLOCK DIAGRAM OF DATA BUS

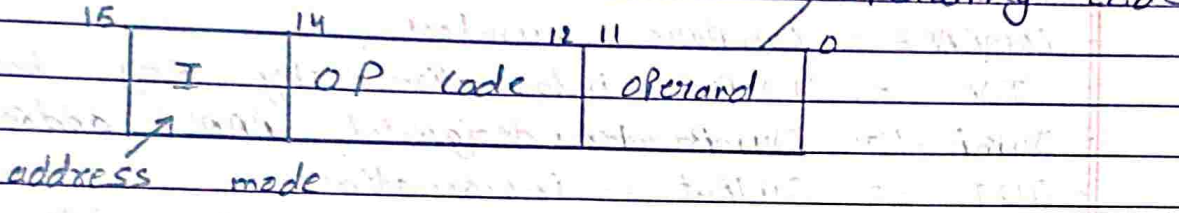
Date
14.3.2023

INSTRUCTION SET

The instruction set also called "I S A" (instruction set architecture) is part of a computer that pretains to programming, which is basically machine language. The instruction set provides commands to tell the processor to tell it what it need to do. The instruction set consist of addressing mode, instruction, native data types, resistors, memory architecture,

interupt and exception handling and external input / output. Basic computer instruction format is of three types -

(i) Memory
(ii) Refference
(iii) Instruction



I=0 = direct address
I=1 = Indirect address

BLOCK DIAGRAM

e.g. an instruction is the 0x86 instruction set, which is eg (8080 mic. and 8086 processor) common to find on computer. Different computer processor can use almost use same instruction set while still having very different internal design Both the intel pentnium and Amd (Athanol processor) use nearly the same x-86 instruction set. An instruction set can be build into the hardware of the

P.T.O.

Date 14.3.2023
Page 91

processor or it can used by the software using the interpriator the speed is depend on the designed of the processor.

Example of instruction set

- ADD - Two numbers
- COMPARE - Compare numbers
- IN - INPUT information by Key-board
- Jump - Jump to designated RAM address
- OUT - Output information
- STORE - STORE information to RAM.

Date 15.3.2023
Page 92

MEMORY ORGANISATION

(i) Associative Memory
(ii) Cache memory
(iii) Mapping

(i) ASSOCIATIVE MEMORY :

A memory unit accessed by contents is called an associative memory. All contents addressable memory. This type of memory is accessed simultionally and in parallel on the basis of data content rather than byspecific address or location. It is a set of storage location but it works on contain rather than its address in memory. Associative storage or content addressable memory in a computer system is a computer memory used in certain very high speed searching application. Instead of looking up a storage location by its address it looks up a storage location by its contents.

e.g. search engine (google), In RDBMS.

P.T.O.

15.3.2023
93

There are two types of operation of Associative memory:

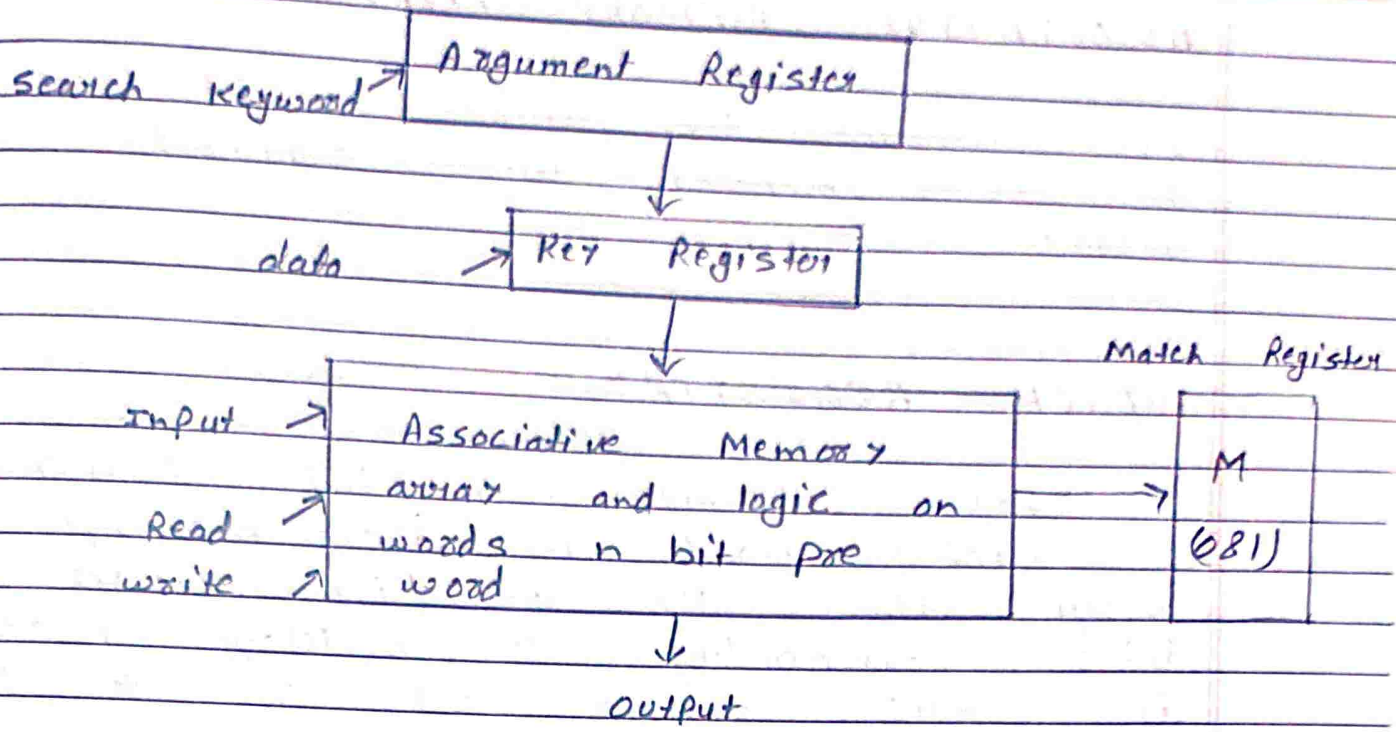
(i) Write operation
(ii) Read operation

(i) WRITE OPERATION:

When a word is written in an associative memory, no address is given the memory is capable of finding an unused location to store the word.

(ii) READ OPERATION:

When a word is to be read from an associative memory the contents of the word or a part of the word is specified the memory locates all the words which match the specified contents and marks them for reading.



(i) ARGUMENT REGISTER :

It contains the word to be searched it has 'n' bits (ONE for each bit of the word).

(ii) KEY REGISTER :

It provide marks for choosing a particular field or e in the argument word. It also has 'n' bits

P.T.O.

15.3.2023
95

ASSOCIATIVE MEMORY ARRAY:

It contains the words which are to be compared with argument word.

MATCH REGISTER:

It has 'n' bits, n bits corresponding to each word in the memory array after the matching causes the bits corresponding to matching words in match register are set to be 1.

ADVANTAGE OF ASSOCIATIVE MEMORY:

It is a fast allocation memory and data can be searched by a similiar words.

DISADVANTAGE OF ASSOCIATIVE MEMORY

Only one disadvantage is that it is very costly and used in the big company.

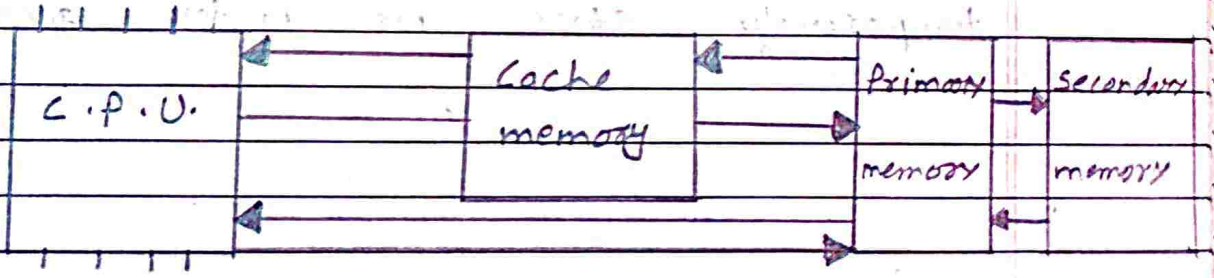
Date 21.03.2023
Page 96

CACHE MEMORY :

Cache memory is a special & very high speed memory. It is used to speed up and synconize with high speed CPU. Cache memory is an exteremly fast memory type that's acts as a buffer between RAM and CPU. It holds frequently requested data and instruction so that they can immediately available to the CPU, when needed. Cache memory is used to reduce the average time to access data from the main-memory. The cache is a smaller and faster memory that stores copies of the data from frequently used main memory location.

There are various different independent cache in CPU and which store instruction and data.

BLOCK DIAGRAM:



The data have to travel in different levels :

(i) Level 1 or Register
(ii) Level 2 or Cache memory
(iii) Level 3 or Main memory
(iv) Level 4 or Secondary memory

(i) Level 1 or Register :

It is a type of memory in which data is stored and accepted that are immediately store in CPU.

e.g. accumulator, program register / counter, address register etc.

(ii) Level 2 or Cache memory :

It is the faster memory which has faster access time where data is temporarely store for faster access.

(iii) Level 3 or Main memory:

It is a memory on which computer works currently. It is small in size and ones power off, data no longer stays in this memory.

(iv) Level 4 or Secondary memory:

It is external memory which is not as fast as main memory but see data stays permanently in this memory.

Date 22.3.2023
Page 99

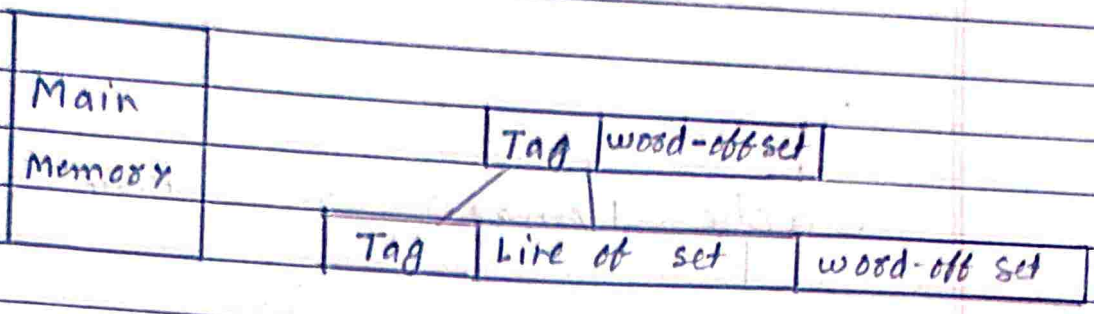
CACHE MEMORY :

There are three different types of mapping used for the purpose of cache memory, which is as follows :

(i) Direct mapping
(ii) Associative mapping
(iii) Set - Associative mapping

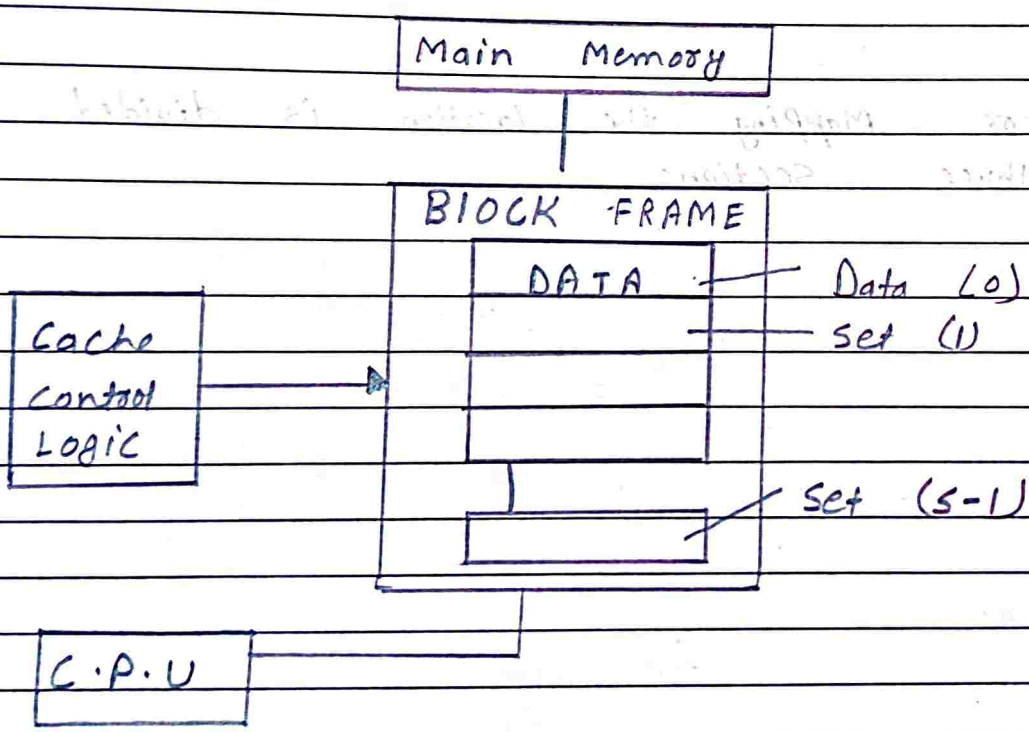
(i) DIRECT MAPPING :

The simplest technique known as direct mapping, Map each block of main - memory into only one possible Cache - line or In Direct mapping assign each memory block to a specific line in the cache. If a line is previously taken up by a memory block then a new block needs to be loaded the old block is trashed, an address space is split into two parts index field and a tag field. The Cache is used to store the tag field, whereas the rest is store in the main mapping memory. Direct mapping performance is directly proportional to the hit ratio.



As we find that cache memory of Direct mapping divide the tag into two forms for the execution the file.

BLOCK DIAGRAM:



Date 28.3.2023
Page 101

Cache Memory

L_0	B_0	B_4	B_8	:
L_1	B_1	B_5	B_9	:
L_2	B_2	B_6	B_{10}	.
L_3	B_3	B_7	B_{11}	B_{128}

Main memory

W_0	W_1	W_2	W_3	B_0
W_4	W_5	W_6	W_7	B_1
W_8	W_9	W_{10}	W_{11}	B_2
W_{12}	W_{13}	W_{14}	W_{15}	B_3
..	...	W_{127}	W_{128}	B_{31}

16 bits or 16 words

$\frac{16}{4} = 4$ size of line.

For Mapping the location is divided into three section.

← 7 →

3	2	2
Tag	Line no.	Block offset

Eg.

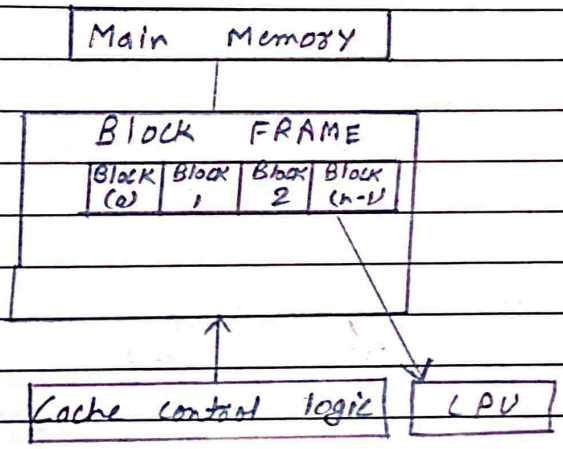
Tag	Line	Block
001	01	00

Date 28.3.2023
Page 102

ASSOCIATED MAPPING :

In this type of mapping the associative memory is used to store contents and address of the memory word. Any block can go into any line of the cache memory. This means that the word (TD) bits are used to identify which word in the block is needed but the tag becomes all of the remaining bits. This enables the placement of any word at any place in the cache memory. It is considered to be fastest and the most flexible mapping form.

The block diagram are as follows:



Date 28.3.2023
Page 109

B_0	B_1	B_2	B_3	L_0	W_0	W_1	W_2	W_3	B_0
B_0	B_1	B_2	B_3	L_1	W_4	W_5	W_6	W_7	B_1
B_0	B_1	B_2	B_3	L_2	W_8	W_9	W_{10}	W_{11}	B_2
B_0	B_1	B_2	B_3	L_3	W_{12}	W_{13}	W_{14}	W_{15}	B_3
					W_{16}	W_{17}	W_{18}	W_{19}	B_4
								W_{128}	B_{31}

MULTIPROCESSOR

Physical Address

5	2

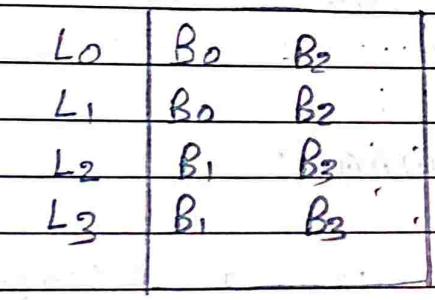
Block — Block offset

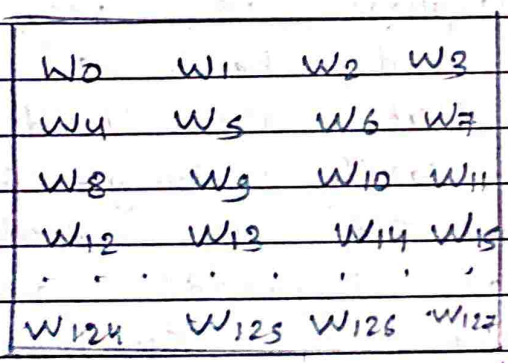
Date 13.04.23
Page 104

SET ASSOCIATIVE

Set associative mapping combines direct mapping with fully associative mapping by arrangment lines of a cache into sets. The sets are perceiving using a direct mapping scheme. However the lines with in each sets are created as a small fully associative cache where any block that can be same in the set can be stored to any line inside the set.

BLOCK DIAGRAM:





Date 13.4.2023
Page 105

Language of a Computer

Low - level - language (Machine language) → Middle - level - language (Assembly language) → High - level - language (Similar to human

Machine Language:

Low - level - language is the only language which can be understand by the computer. Low - level - language is also known as machine language consist only two symbols that is 0 and 1. All the instructions of machine language are written in the form of binary number ones and zeros. A computer can directly understand the machine language (8 bits = 1 byte)

High - LEVEL - LANGUAGE:

High - level - language is a computer language which can be understood by the user. The high - level - language is very similar to human languages and has a set of grammer rules that are used to make instructions more easily. Every - high - level language has a set of predefined words known as keywords and a set of rules known as Syntax to create

instruction. The high-level-language is easier to understand. It needs to be converted into the low-level-language to make it understable by the computer. we use compilier and interpretor to convert high level language to low-level-language.

ASSEMBLY-LEVEL-LANGUAGE

Assembly language mid-level-language is a computer-language in which the instructions are created using symbol such as letters digits and special characters. Assembly language is an example of middle-level-language. In assembly language we use predefined word is called mnemonics. Binary code instruction in low-level-language are replaced with memories mnemonics and operands in middle-level-language but the computer cannot understand memories mnemonics, so we used a translator called assembler to translate mnemonics into machine language.

Assembler is a translator which takes assembly code as input and produces machine code as output.

Syntax / structure of Assembler language

[label:] macromanics [operand] [;]

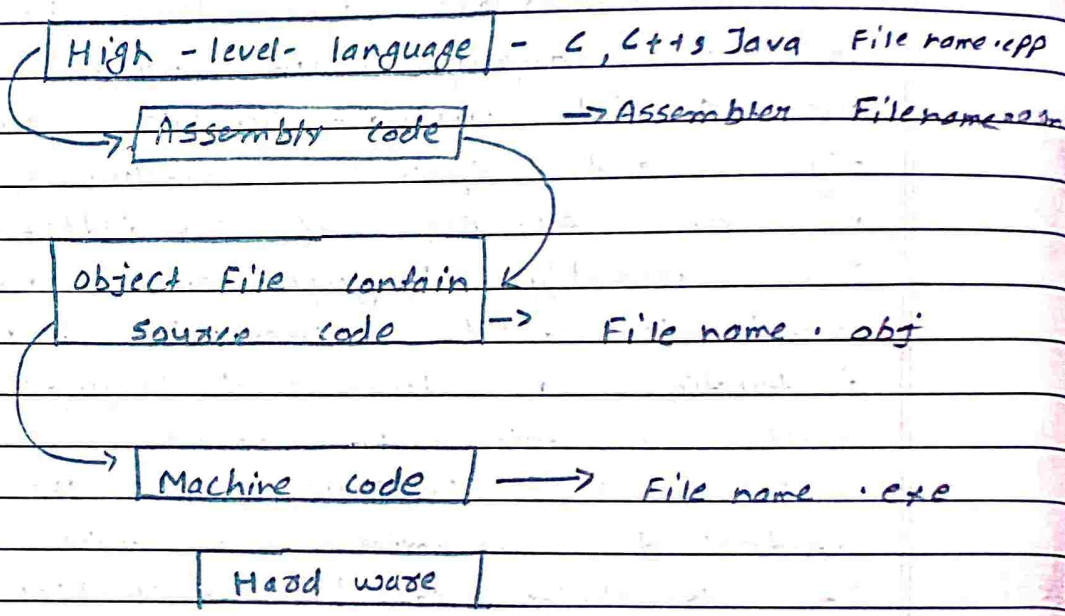
: add R1, R2 [Add the value].

Syntax : [label] : Mnemonics [operand] [;]

- [label] ↓ Symbolic name for memory locating
- Mnemonics ↓ operation to be perform
- [operand] ↓ data for

e.g. L1 Add R1, R2 ;

STRUCTURE OF ASSEMBLY LANGUAGE APPLICATION



Q. What is Assembly language application.

Assembly language helps programmers to write human readable code that is almost similar to machine language. Machine language is difficult to understand

and read as it is just a series of number. Assembly language helps in providing full control of what task are computer is to perform.

Q. Why should we learn assembly language.

The learning of assembly language is still useful for programmers. It helps in taking complete control over the system and it's resources by learning assembly language the programmer can write the code to acess and retrive the memory addresss of pointers and values. It mainly helps in speed optimition, that increases efficeincy and performance.

Assembly label language helps in understanding the processor and memory function. If the programmer is writting any program that means the programmers should have a complete understanding of the processor. It is cryptic and symbolised the language.

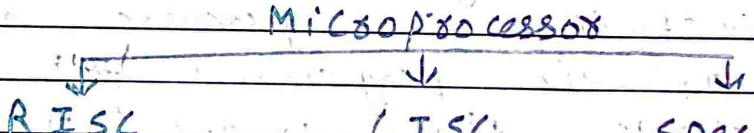
It also helps in connecting the hardware directory. This language is mainly based on computer, archeitecture and it recognise certain type of processor and its different for different C.P.U.

Date 04.05.2023
Page 109

Q. What is microprocessor ?

A microprocessor is a computer processor that combines the functions of a central processing unit on a single integrated circuit. The microprocessor is thus a multy processor, multiprocess, clock driven, resistor based, digital integrated circuit that takes binary data as input, process it as per the instruction.

Microprocessor is of three type.

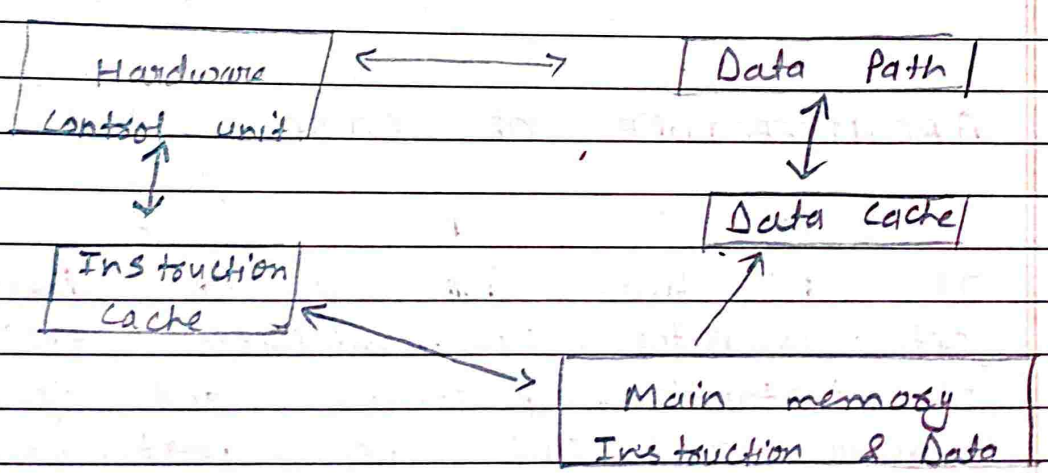


Date 11.05.2023
Page 110

ARCHITECTURE OF RISC

(i) It is stand for Reduced instruction set computer. It is designed to reduce the execution time by simplifying the instruction set. Each instruction requires only 1 clock cycle (fetch, decode and execute etc.) It uses highly optimised set of instruction. It is used in tablets, mobiles, smartphones, portable devices.

Diagram



Date 11.5.2023
Page 111

CHARCTERSTIC OF RISC

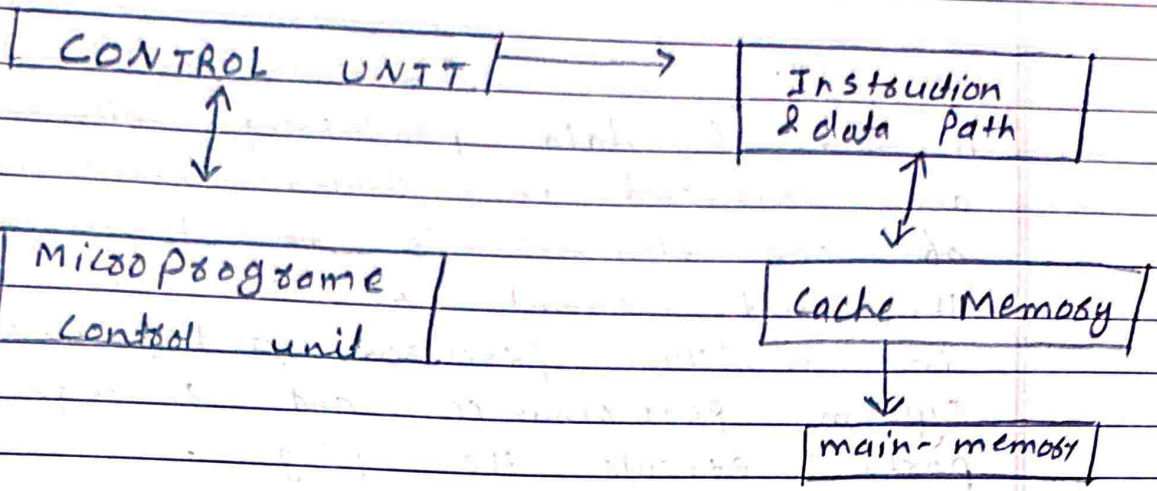
(i) Relatively few instruction set.
(ii) Relatively few addressing mode.
(iii) limited memory acess to load and save instruction.
(iv) All operation are done with in the resistors of the C.P.U.
(v) Fixed length and easily decode ~~inst~~ instruction format.
(vi) Single cycle instruction execution.
(vii) Hardware required rather than microprogramme controlled.
(viii) low cost.

ARCHITECTURE OF CISC

It is stands for complex instruction set computer, To minimize the number of instruction, programme and ignoring the number of cycle per instruction. It is used in Desktop, laptop and other mechanical devices. It has less and other mechanical devices. It has less memory (ram) is required to store instructions.

Date 11.5.2023
Page 112

DIAGRAM



CHRACTERSTIC OF CISC

(i) Many addressing mode is used.

(ii) Large number of instruction set.

(iii) Variable length of instruction format.

(iv) Several cycles may be required to execute one instruction.

(v) Instruction decoding logic is complex.

(vi) Cost is high.

(vii) Memory is used for work.

Date: 12.5.2023
Page: 113

PIPELINING PROCESSING

A set of data processing elements which are connected in series hence, the output of one elements is in the input of the next element of the instruction used in modern processors and increase the system performance and through (codes) ports execute the program.

There are 5 different phases to complete / execute a program : they are : (i) Instruction fetch (ii) Instruction decode (iii) Operand fetch (iv) Operand execute (v) operand store.

IF	ID	OF	OE	OS

So, that every program have to finish these five phases for execution.

The process of pipelining is of two types :-

(i) Non- Pipeline process
(ii) Pipe line - Process

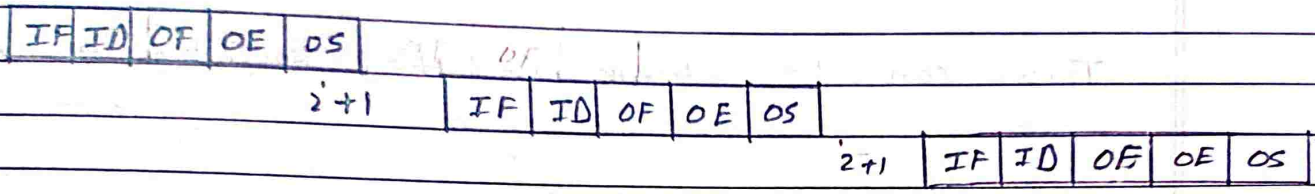
Date 12.5.2023
Page 114

NON-PIPELINE PROCESS:

In this process all the five phases are to be complete for the new execution of the programe. So, that a programe have to wait unless and until completed the process.

Block diagram are shown:

(i)



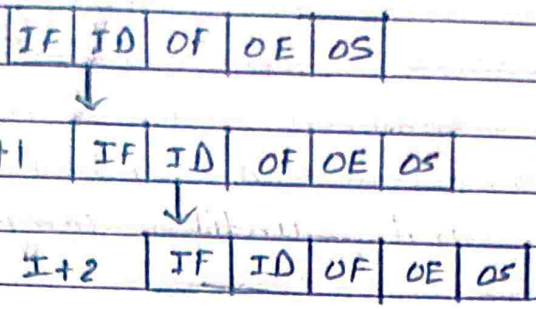
∴ total phase (tp) = Phase duration (ts) { Total instruction duration }

(ii) PIPELINE PROCESS:

It is a process in which programs does not have to ~~unit~~ wait for the first job to complete it moves to the another block so, that next job will be entered as a input for the processing.

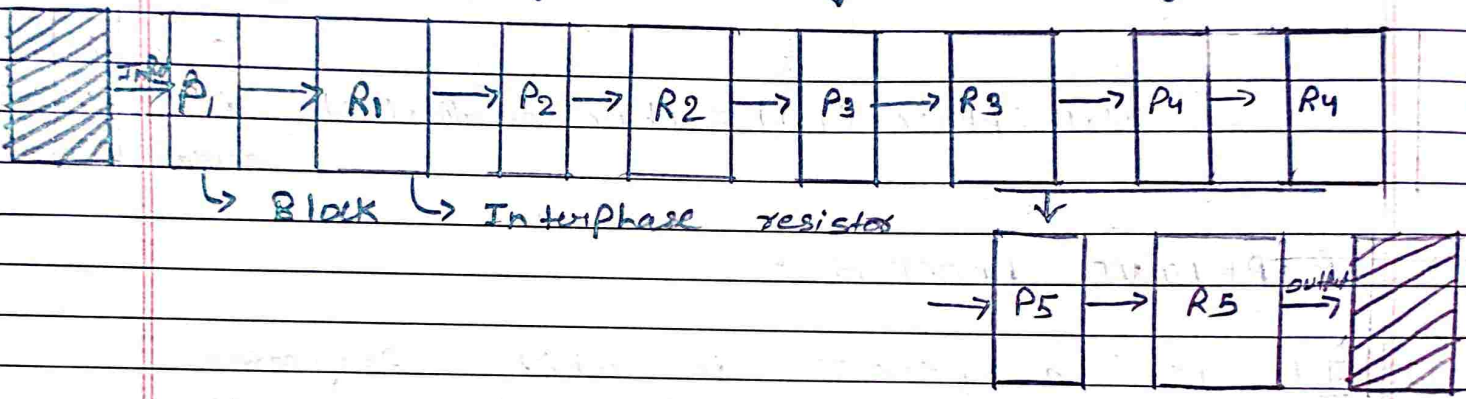
P.T.O.

BLOCK DIAGRAM :



1P = phase duration = (1)

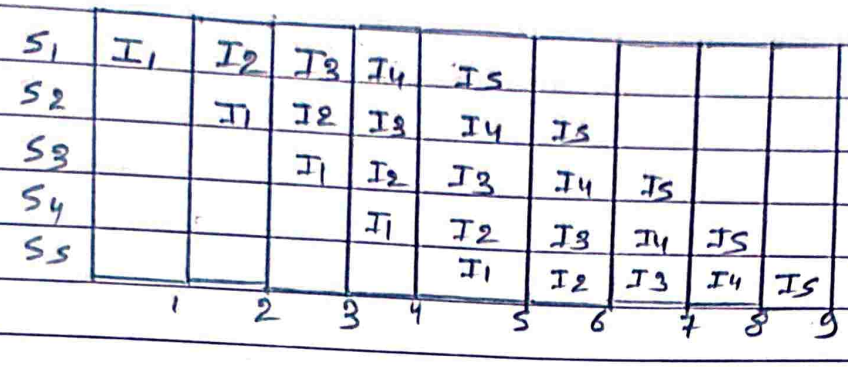
This can be shown in the help of a ex.



It allows storing and executing instruction in a orderly process it is also known as pipeline processing. Pipelining is a technique where multiple instruction are overlapped during execution. Pipeline is divided in - two stages and

these stages are connected with one another to form a pipeline structure.

OR



Execute or store