Hindbookcenter

GET IT ON Google Play

ALL NOTES BOOKS AVAILABLEALL STUDY MATERIAL AVAILABLE

COURIERS SERVICE AVAILABLE

MADE EASY , IES MASTER , ACE ACADEMY , KREATRYX

ESE , GATE, PSUs BEST QUALITY TOPPER HAND WRITTEN NOTES

MINIMUM PRICE AVAILABLE @ OUR WEBSITE

1. ELECTRONICS ENGINEERING
2. ELECTRICAL ENGINEERING
3. MECHANICAL ENGINEERING
4. CIVIL ENGINEERING
5. INSTRUMENTION ENGINEERING
6. COMPUTER SCIENCE

IES ,GATE , PSU TEST SERIES AVAILABLE @ OUR WEBSITE

- IES –PRELIMS & MAINS
- GATE

➢ NOTE;- ALL ENGINEERING BRANCHS

➢ ALL PSUs PREVIOUS YEAR QUESTION PAPER @ OUR WEBSITE

PUBLICATIONS BOOKS -

MADE EASY , IES MASTER ,ACE ACADEMY ,KREATRYX ,GATE ACADEMY, ARIHANT ,GK

RAKESH YADAV, KD CAMPUS , FOUNDATION , MC –GRAW HILL (TMH) ,PEARSON...OTHERS

HEAVY DISCOUNTS BOOKS AVAILABLE @ OUR WEBSITE

Shop No.7/8 Saidulajab Market Neb Sarai More, Saket, New Delhi-30	Shop No: 46 100 Futa M.G. Rd Near Made Easy Ghitorni, New Delhi-30	F518 Near Kali Maa Mandir Lado Sarai New Delhi-110030	Shop No.7/8 Saidulajab Market Neb Sarai More, Saket, New Delhi-30

Website: www.hindbookcenter.com

Contact Us: 9711475393

Computer Organization

10 marks

Syllabus:

Module 1: computer architecture

Module 2: computer organization.

Ref Books: 1. computer architecture & organigization.
- Morris Mano. (Hardware design)

2. computer orgn.
- william stallings.

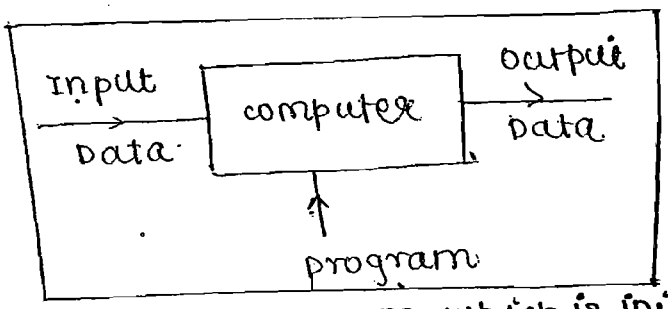
Faculty: Pingili Sagar.

email: sagar262003@yahoo.co.in.

Keywords:

computer:

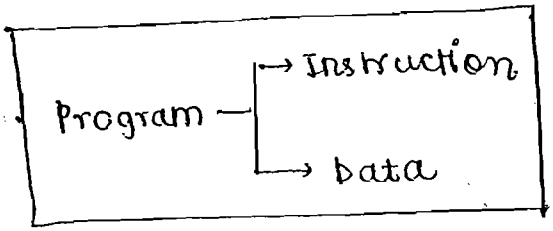
computer is a computational Machine used to process the data under the control of a application program. Therefore computer system functionality is program execution.



(program which is initiated by user)

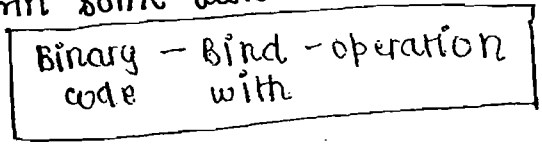
program:

Program is a sequence of instructions along with the data.



Instruction:

Instruction is a Binary code which is designed inside the processor to confirm some task.



Eg: If CPU – 'x' supports 8 different operation then opcode = $\log_2 8$ = 3bit.

Binary code (opcode)	operation
000	+
001	–
010	*
⋮	⋮
111	AND

Decided by the Designer

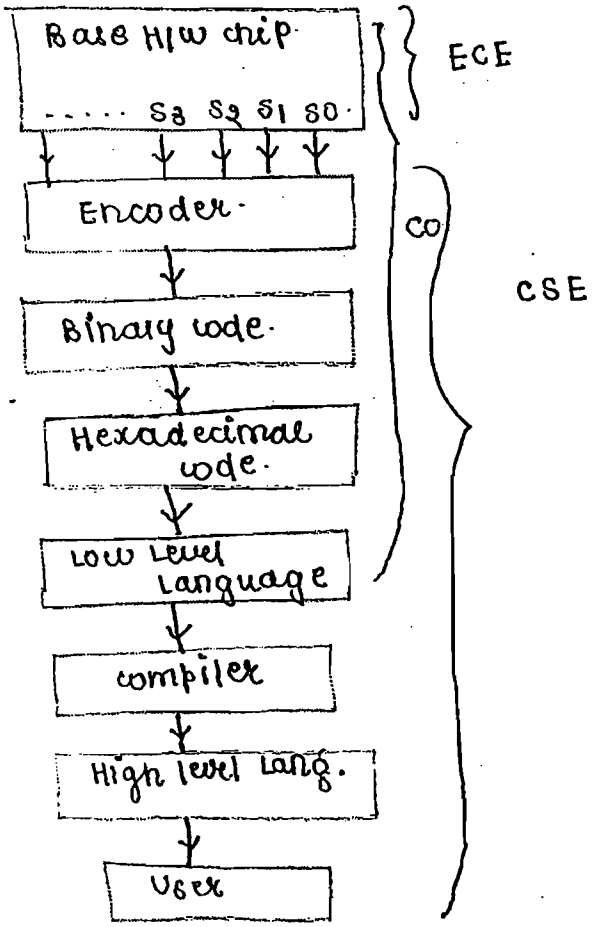
ROM.

control unit.

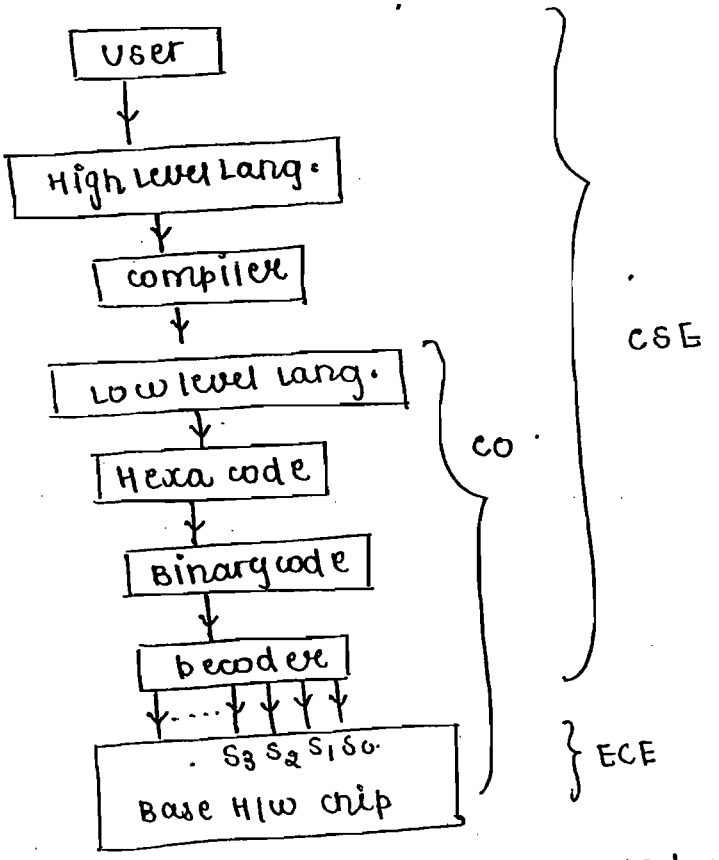
Encoding process: n signals given How many bits required to process signals $\log_2 n$.

Decoding process: n bits are given, How many operation can be performed by computer : '2^n' operation.

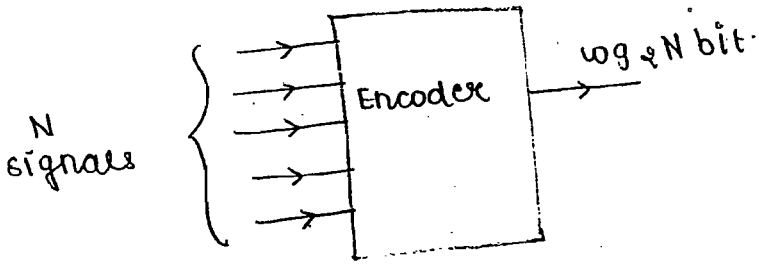
Designer View:



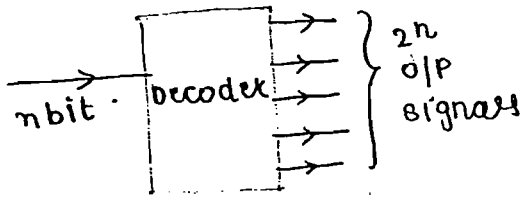
User View:



Encoding: In this process 'N' signals are represented using $\log_2 N$ bit format.



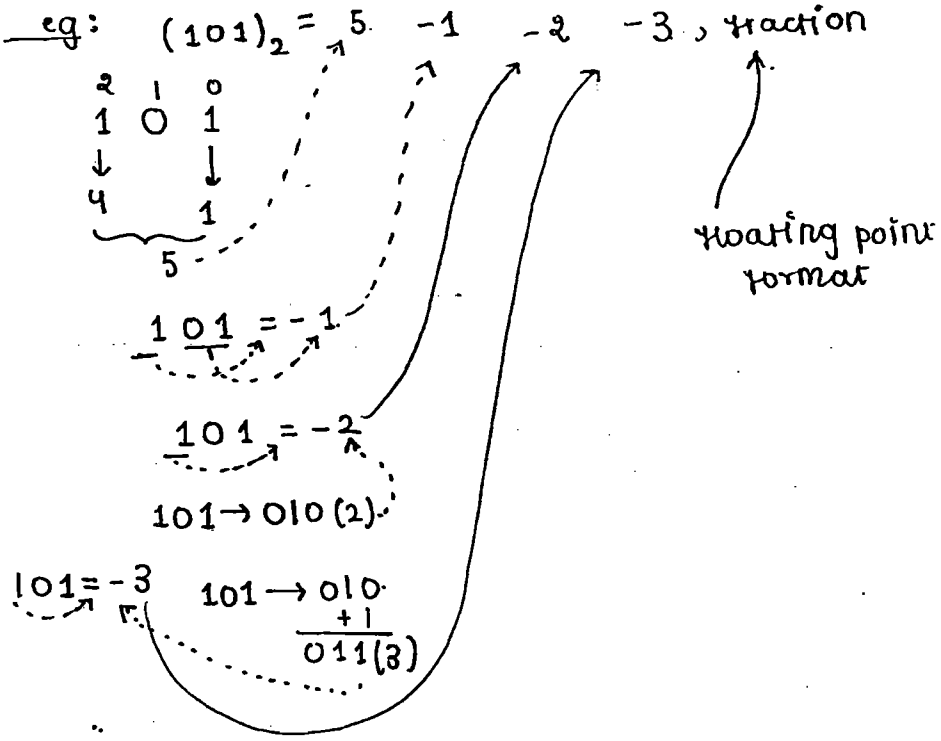
Decoding: In this process, n bit decoder produces 2^n output signals.



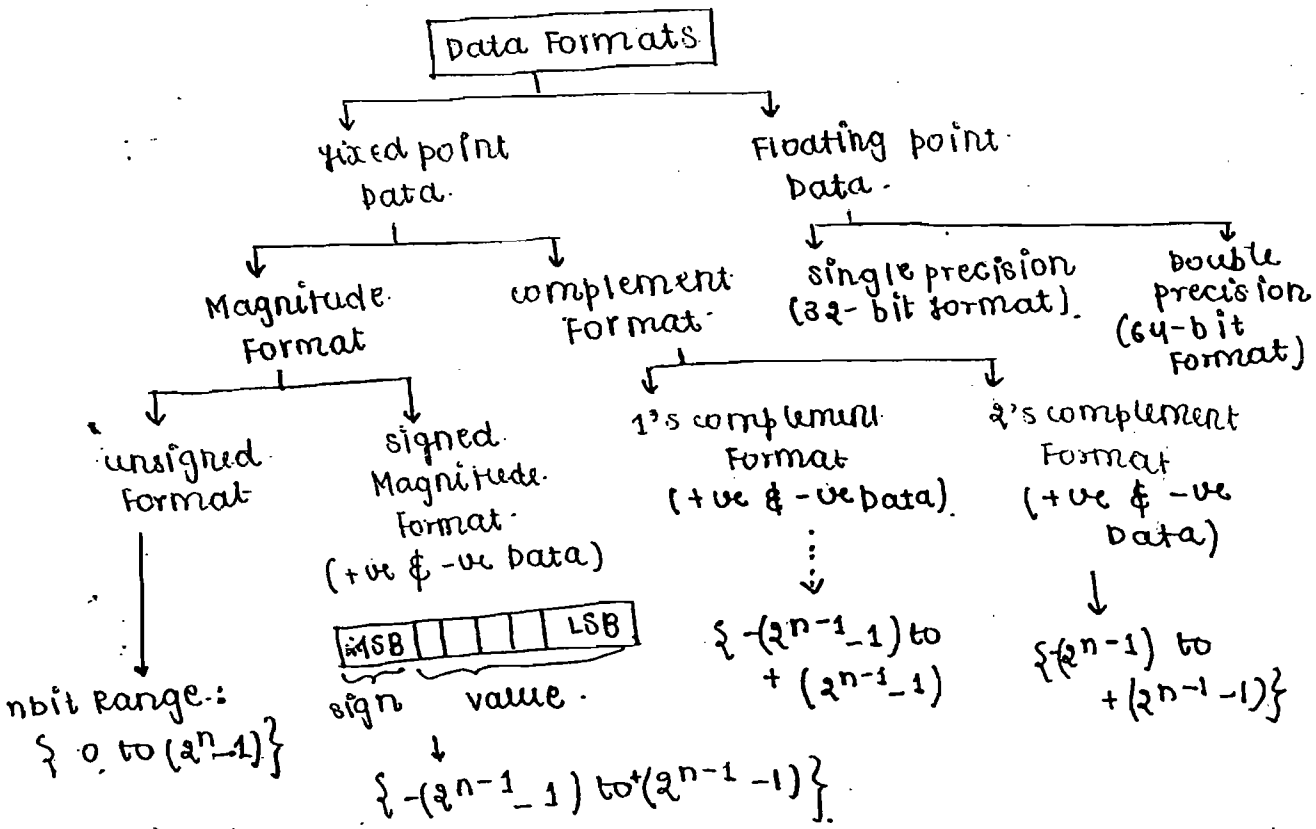
④ Data: It is a Binary code which is associated with a value based on the Data format.

Binary code - Bind with - value

eg: $(101)_2$ = 5 -1 -2 -3, fraction



Data Representation:



Fixed point Data

4 Bit Binary.	Unsigned Data.	Sign Magnitude	1's complement	2's complement
0000	0	+0	+0	+0
0001	1	+1	+1	+1
0010	2	+2	+2	+2
0011	3	+3	+3	+3
0100	4	+4	+4	+4
0101	5	+5	+5	+5
0110	6	+6	+6	+6
0111	7	+7	+7	+7
1000		-0	-7	-8
1001	8	-1	-6	-7
1010	9	-2	-5	-6
1011	10	-3	-4	-5
1100	11	-4	-3	-4
1101	12	-5	-2	-3
1110	13	-6	-1	-2
1111	14	-7	-0	-1
	15			

Data Redundancy Problem. 'NOT in USE"

'NOT in USE"

1's complement

1000 = -7

111 (7)

1001 = -6

110 (6)

2's complement

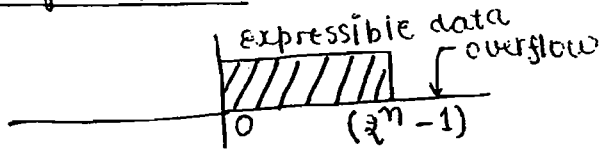
1000 = -8

000
111
+ 1
1000 (8)

1001 = -7

110
+1
111 (7)

unsigned data



eg: 4bit data {0 to 15}

	①	1	1	1
15	:	1	1	1	1
+15	:	1	1	1	1
30		1	1	1	0

↓ overflow.

Test with 5 bit Data : {0 to 31}

NOTE:

(n-bit) + (n bit) = (n+1) bit.

↓

1 bit storage space required.

↓

1 Flip flop

↓

Flag.

↓

carry flag.

condition : "Is there an extra bit out of MSB" (or) "Is Borrow required into the MSB"

- T = set = 1 = C
- F = reset = 0 = NC

eg:

6	:	×	1	1	0
		0	1	1	0
⊕ 7		0	1	1	1
13.		1	1	0	1
CY: 0		CY: 0			

Justify PSW (program status word)

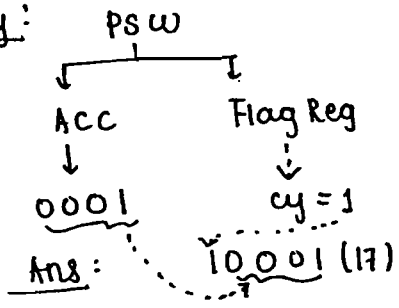
- ACC → 1101
- Flag Register → CY = 0

Ans: 1101 (13)

Eg:

8	1000
9	1001
17	0001
cy = 1.	cy = 1

Justify:



Multiplication:

- Multiplication process is controlled by a Multiplier.
- Two actions are present in the Multiplication.
  (1) generation of partial product
  (2) summation of partial product
- partial product is generated based on the Multiplier bits. i.e when the multiplier bit is '1' partial product is Multiplicant otherwise partial product is zero. (0).
- After the generation of a partial product, provide the solution to produce the final product.

Multiplicant * Multiplier.

1 1 1 1 * 1 1 1 1 ← LSB

			2	1				
		3	1	1	1	1		} partial products.
	3	1	1	1	1	x		
2	1	1	1	1	x	x		
1	1	1	1	x	x	x		
1	1	1	0	0	0	0	1	} final product

2 → 1 0 → carry (1).

4 → 1 0 0 → carry (2).

6 = 1 1 0 → carry (3)

5 = 1 0 1 → carry (2).