Sri SAI RAM ENGINEERING COLLEGE
INSTITUTE OF TECHNOLOGY
West Tambaram, Chennai - 44

SAIRAM DIGITAL RESOURCES

YEAR III

SEM VI

EN8592

WATER SUPPLY ENGINEERING

UNIT NO 1

PLANNING AND DESIGN OF SEWERAGE

1.5 Plumbing systems in buildings

CIVIL ENGINEERING

COURSE : ***WASTEWATER ENGINEERING***

COURSE CODE: **EN8592**

YEAR: **III YEAR**

SEMESTER: **VI SEMESTER**

UNIT I - PLANNING AND DESIGN OF SEWERAGE

1.1 Characteristics and composition of sewage

1.2 Sanitary sewage flow estimation & Hydraulics of flow in sanitary sewers

1.3 Sewer materials & Sewer appurtenances

1.4 Corrosion in sewers & its prevention and control

1.5 Plumbing systems in buildings

1.6 Rain Water ting

House Drainage System:-

The foul matter should be quickly removed away from the sanitary fixtures so as to avoid the putrefaction and production of bad smelling gases.

The drainage system should be able to prevent the entry of gases, vermin etc from the sewer into the building.

Requirements of Good Drainage System in Buildings:

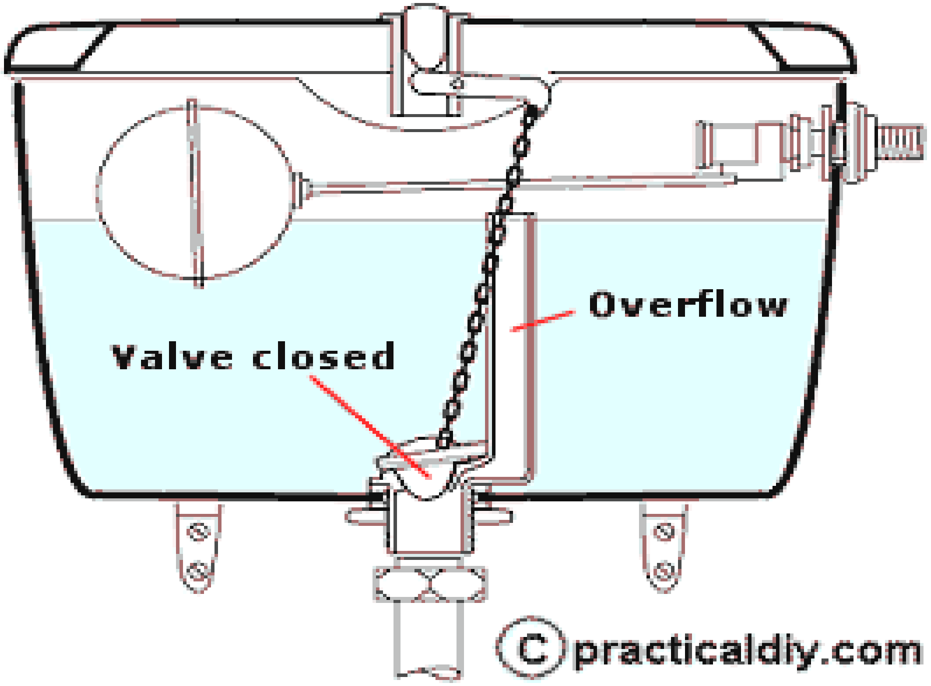
1) The drainage pipes should be strong and durable.
2) The pipes should be of non-absorbent materials.
3) The pipes and joints should be airtight to prevent the leakage of waste water or gases.
4) The network of pipes should have sufficient accessibility for inspection & cleaning.
5) The levels of building, sewer and other points of outlet should be fixed accurately.
6) As per as possible drains should not pass under the buildings.
7) The drains should be given proper ventilation to avoid air locks and syphonage.
8) The system should have traps at all necessary points.

Sanitary Fittings:-

The sanitary fittings are required in house drainage for the efficient collection and removal of waste water from the house to house drain.

The following are some of the sanitary fittings.

1) Traps.
2) Water closets.
3) Flushing cisterns.
4) Urinals.
5) Inspection chambers.
6) Wash basins.
7) Sinks.
8) Bath tubs etc.



Traps:- -

A trap is a depressed or bent sanitary fitting which always remains full of water (water seal).

The function of a trap is to prevent the entry of bad smelling gases into the house.

The effectiveness depends upon the depth of water seal. (25 to 75mm)

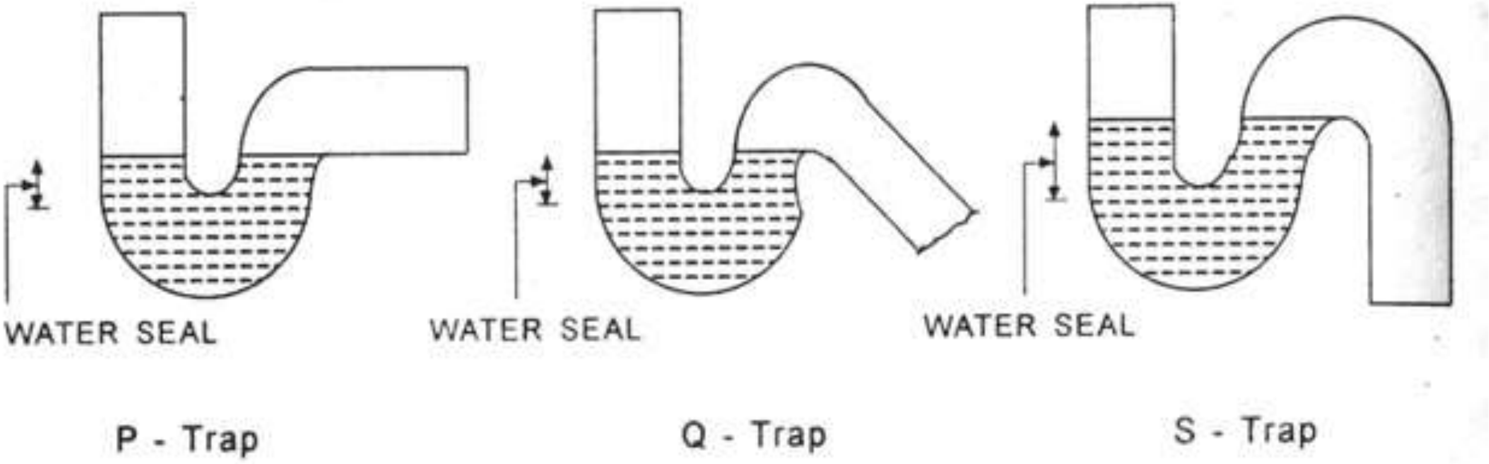
Requirements of Good Trap:-

1) It should be capable of being easily cleaned.
2) It should be easily fixed with the drain.
3) It should be of simple construction.
4) It should possess self-cleansing property.
5) It should posses adequate water seal to fulfill the purpose of installation.
6) The internal and external surfaces should be of smooth finish.
7) It should be free from any inside projection which are likely to obstruct the passage of flow of sewage.

Types of Traps:-

Classification according to shape:-

1) P-Trap.
2) Q-Trap.
3) S-Trap.



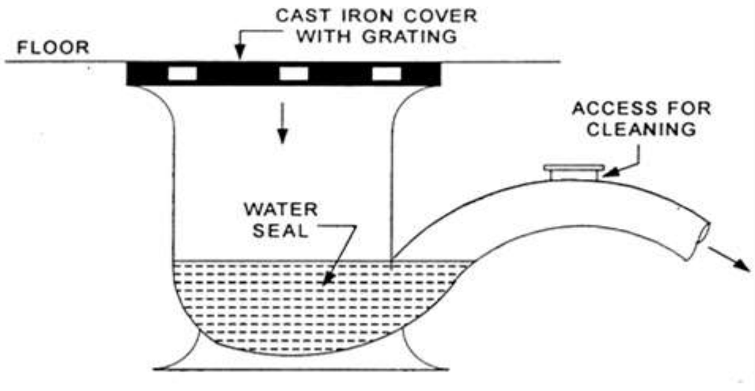
Traps according to shapes

Types of Traps:-

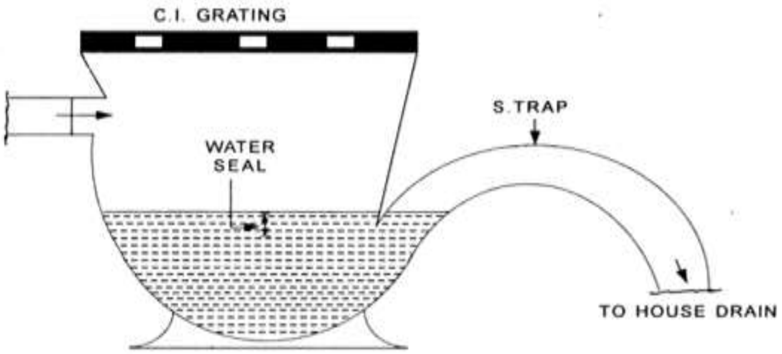
Classification according to use:-

1) **Floor Trap:-** Made of cast iron with grating at top and used to for admitting wastewater from floor of bath & kitchen. This trap is also known as Nahni Trap

2) **Gully Trap:-** Made of stoneware & caste iron. Wastewater from sinks, bath, etc., enter through back inlet & unfoul water from the sweeping of rooms, courtyards, etc. enter from the top, where coarse screen is fitted to check the solid matter.



FLOOR TRAP



GULLY TRAP

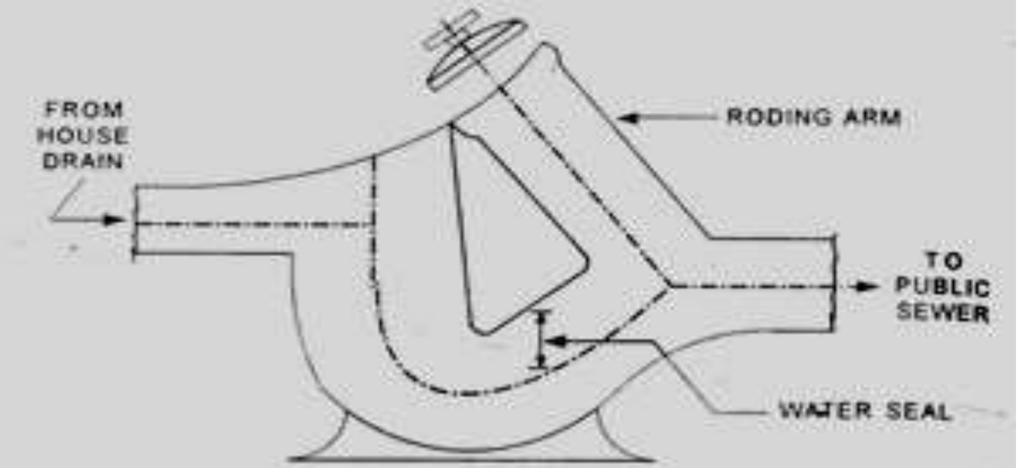
Types of Traps:-

Classification according to use:-

1) **Floor Trap:-**

2) **Gully Trap:-**

3) **Intercepting Trap:-** It is provided in inspection chamber outside the house to prevent the entry of foul gases from the street sewers to the houses through the house drains. This trap has water seal of about 100mm.



Floor trap ,Gully trap & Intercepting trap

www.alamy.com - CYFDR0

Types of Pipes Used in House Drainage:-

1) **Soil Pipe:-** Important section of house drainage system in house.
2) **Vent Pipe:-** Essential for ventilation purpose.
3) **Rainwater Pipe:- C**ollect the rain water from the roof.
4) **Fresh Air Inlet Pipe:-** Provide for fresh air in the last manhole which connects the house drain with public sewer.
5) **Anti-siphonage Pipe:-** This type of pipe is provided not to be happen the siphonic action.

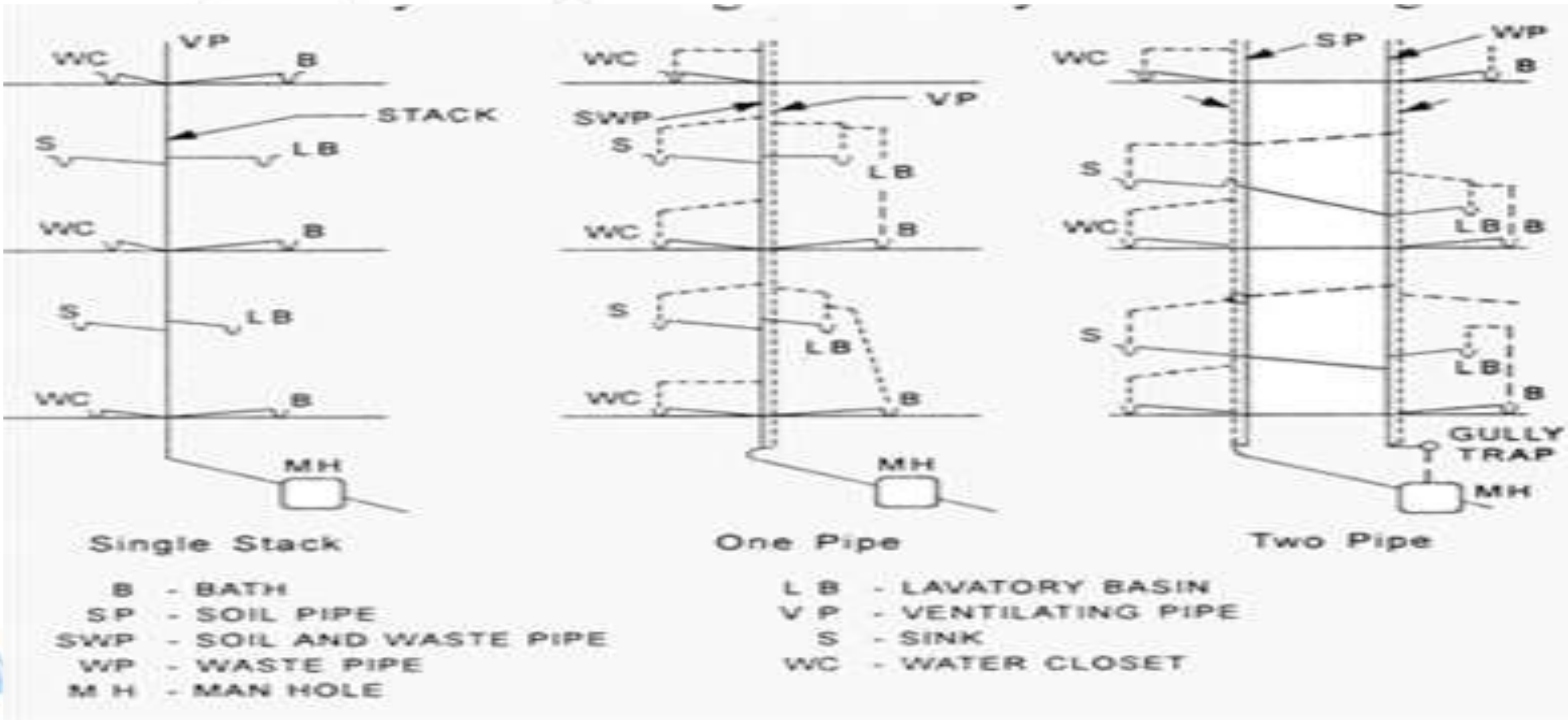
Safety Equipments:-

1) Gas Mask:-
2) Oxygen Breathing Apparatus:-
3) Portable Lightening Equipments:-
4) Portable Air Blow:-
5) Inhalators :-
6) Safety Belt:-

System of House Plumbing:-

Following are three major system adopted in plumbing of drainage work in a building.

1.) Single Stack System:- A single vertical soil pipe is fixed & all the waste matter from baths, kitchens, water closet etc. is discharged into it. It also act as a vent pipe. The single stack system is economical but it is entirely effective in the depth of water seal only.



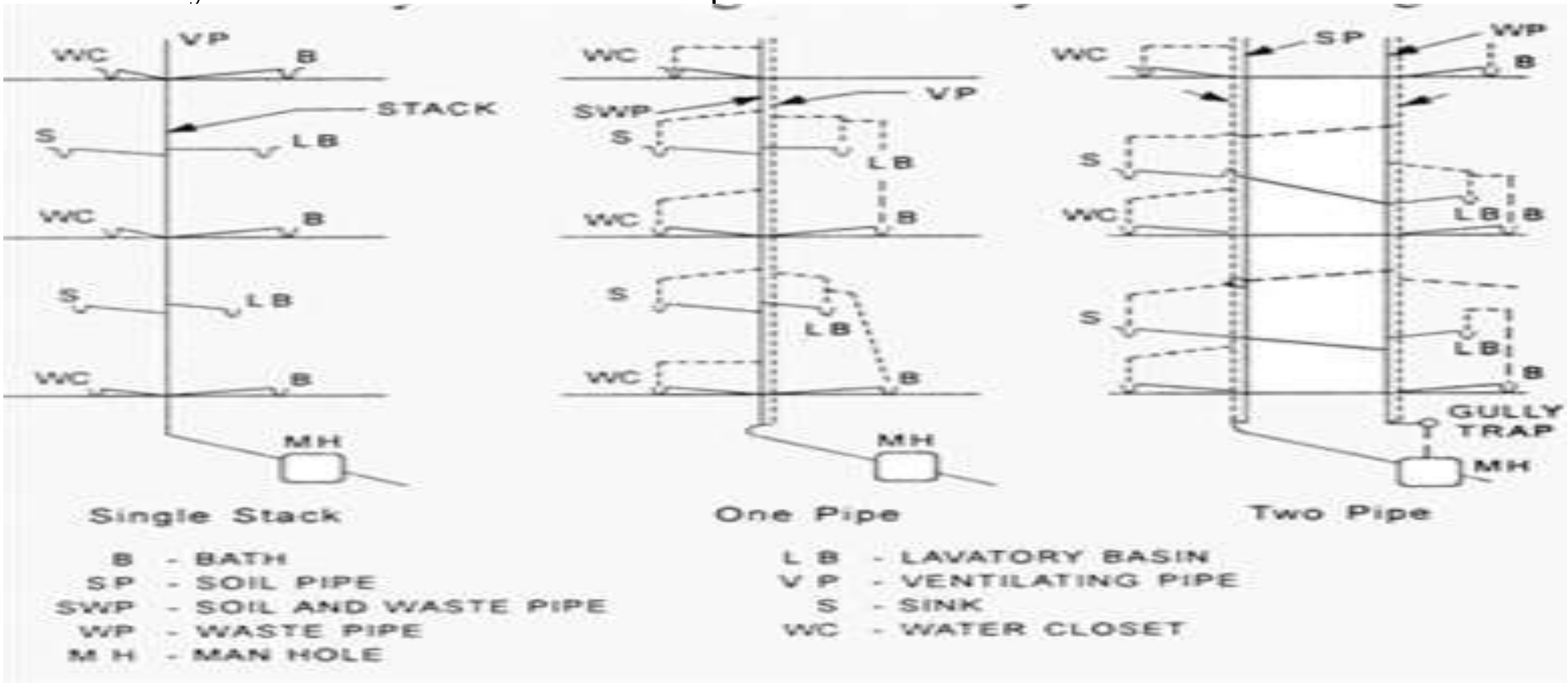
Single Stack One Pipe Two Pipe

B	- BATH
SP	- SOIL PIPE
SWP	- SOIL AND WASTE PIPE
WP	- WASTE PIPE
M H	- MAN HOLE
L B	- LAVATORY BASIN
V P	- VENTILATING PIPE
S	- SINK
WC	- WATER CLOSET

System of House Plumbing:-

Following are three major system adopted in plumbing of drainage work in a building.

2.) One Pipe System:- Only one pipe with a ventilation pipe is provided which collects both the foul soil waste as well as unfoul waste from the building. For multistory building, lavatory blocks of various floors are so placed one over the other.



Single Stack

One Pipe

Two Pipe

System of House Plumbing:-

Following are three major system adopted in plumbing of drainage work in a building.

3.) Two Pipe System:- Two sets of pipes are laid. The soil fixture such as urinal & WC are connected to the vertical soil pipe. Waste matter from baths, kitchen, etc. are connected by another vertical waste pipe. Soil pipe & waste pipe requires separate vent pipe hence requires four pipes & therefore becomes costly.

- Two pipe system is better than other system of plumbing for efficient conveyance of sanitary waste with minimum of risk.
- One pipes system has a more economical layout of pipes but requires sufficient safeguard, include proper ventilation, adequate water seal and direct connection between the sanitary fixture & soil pipe to make drainage effective.
- Now days single stack system mostly used on grounds of economy easy planning of plumbing system.

Pumping of Sewage:-

In sewerage system at some places the sewage can not flow under its gravitational force only & it requires its lifting.

Why it necessary to pump the sewage in sewerage system:-

1) In low laying areas sewage can not flow by gravity into mains & submains, therefore sewage need to pump from lower branch to main sewer.
2) When basement are provided in the buildings, the sewage is pumped to the sewer line.
3) When ground is flat and not getting self cleansing velocity.
4) At the treatment plant to rise it upto the plant for treatment.
5) To lay the sewer along the ridge of hill or obstruction so as to avoid the construction of a tunnel.
6) At the outfall, if level of water course is higher that level of outlet of sewer.

THANK YOU …….

Sri SAI RAM ENGINEERING COLLEGE
INSTITUTE OF TECHNOLOGY
West Tambaram, Chennai - 44

SAIRAM
DIGITAL RESOURCES

YEAR III

SEM V

EC8691

MICROPROCECSSORS AND MICROCONTROLLERS

UNIT NO 1

MICROPROCESSOR ARCHITECTURE

COMPUTER SCIENCE & ENGINEERING

8086 MICROPROCESSOR ARCHITECHTURE

Architecture of 8086

The 8086 CPU is divided into two independent functional units:

1. Bus Interface Unit (BIU)
2. Execution Unit (EU)

1. BIU consists of

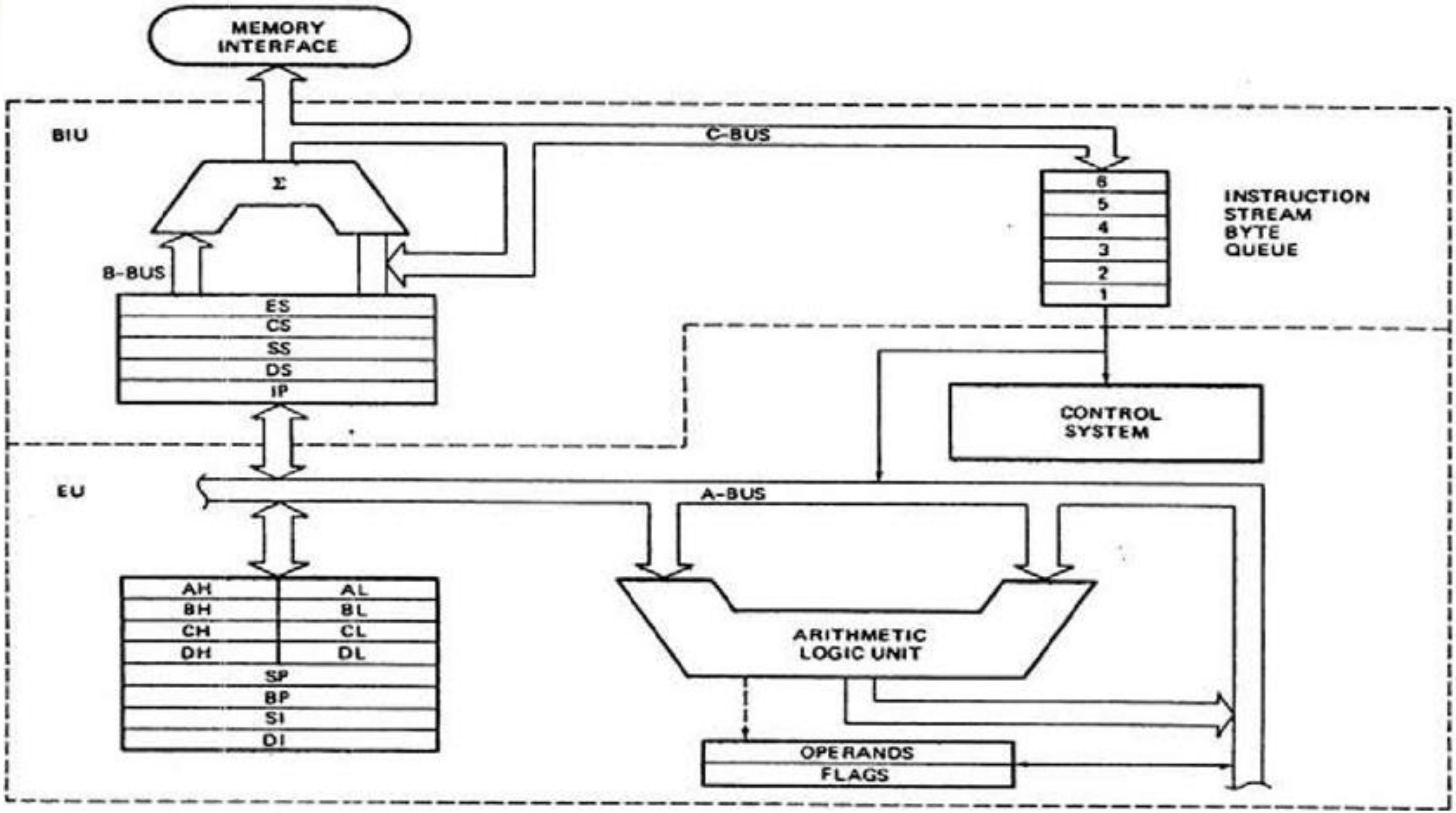
- Code Segment Register (CS)
- Data Segment Register (DS)
- Extra Segment Register (ES)
- Stack Segment Register (SS)
- and
- Instruction Pointer (IP)

The EU contains the following 8-bit registers:

- AH & AL (AX-16 BIT)
- BH & BL (BX-16 BIT)
- CH & CL (CX-16 BIT)
- DH & DL (DX-16 BIT)

It also includes the following 16 bit registers:

- STACK POINTER (SP)
- BASE POINTER (BP)
- SOURCE INDEX (SI)
- DESTINATION INDEX (DI)


MEMORY INTERFACE
BIU
C-BUS
Σ
6
5
4
3
2
1
INSTRUCTION STREAM BYTE QUEUE
B-BUS
ES
CS
SS
DS
IP
CONTROL SYSTEM
EU
A-BUS
AH
AL
BH
BL
CH
CL
DH
DL
SP
BP
SI
DI
ARITHMETIC LOGIC UNIT
OPERANDS
FLAGS

General Specification

- It is a 16 bit microprocessor.
- 8086 has a 20 bit address bus can access upto 2^{20} memory locations (1 MB) .
- It can support up to 64K I/O ports.
- It provides 14, 16-bit registers.
- It has multiplexed address and data bus AD0- AD15 and A16 – A19
- It requires single phase clock with 33% duty cycle to provide internal timing.
- 8086 is designed to operate in two modes, Minimum and Maximum.
- It can prefetch upto 6 instruction bytes from memory and queues them in order to speed up instruction execution.
- It requires +5V power supply.
- A 40 pin dual in line package

BIU (Bus Interface Unit)

The BIU Contains :

- **CODE SEGMENT REGISTER (CS)**

  It points to the starting address of the code segment.

- **DATA SEGMENT REGISTER (DS)**

  It points to the starting address of the data segment. The maximum capacity of a segment may be up to 64Kbytes. The starting address is divisible by 16.

- **EXTRA SEGMENT REGISTER (ES)**
- **STACK SEGMENT REGISTEER (SS)**

  It is used to compute the address of the stack location to be accessed.

- **INSTRUCTION POINTER (IP)**

  It points to the address of the next instruction to be executed.

Operation of BIU & EU

BIU Operation

- The function of BIU is to:

□ Fetch the instruction or data from memory.

□ Write the data to memory.

□ Write the data to the port.

□ Read data from the port.

EU Operation

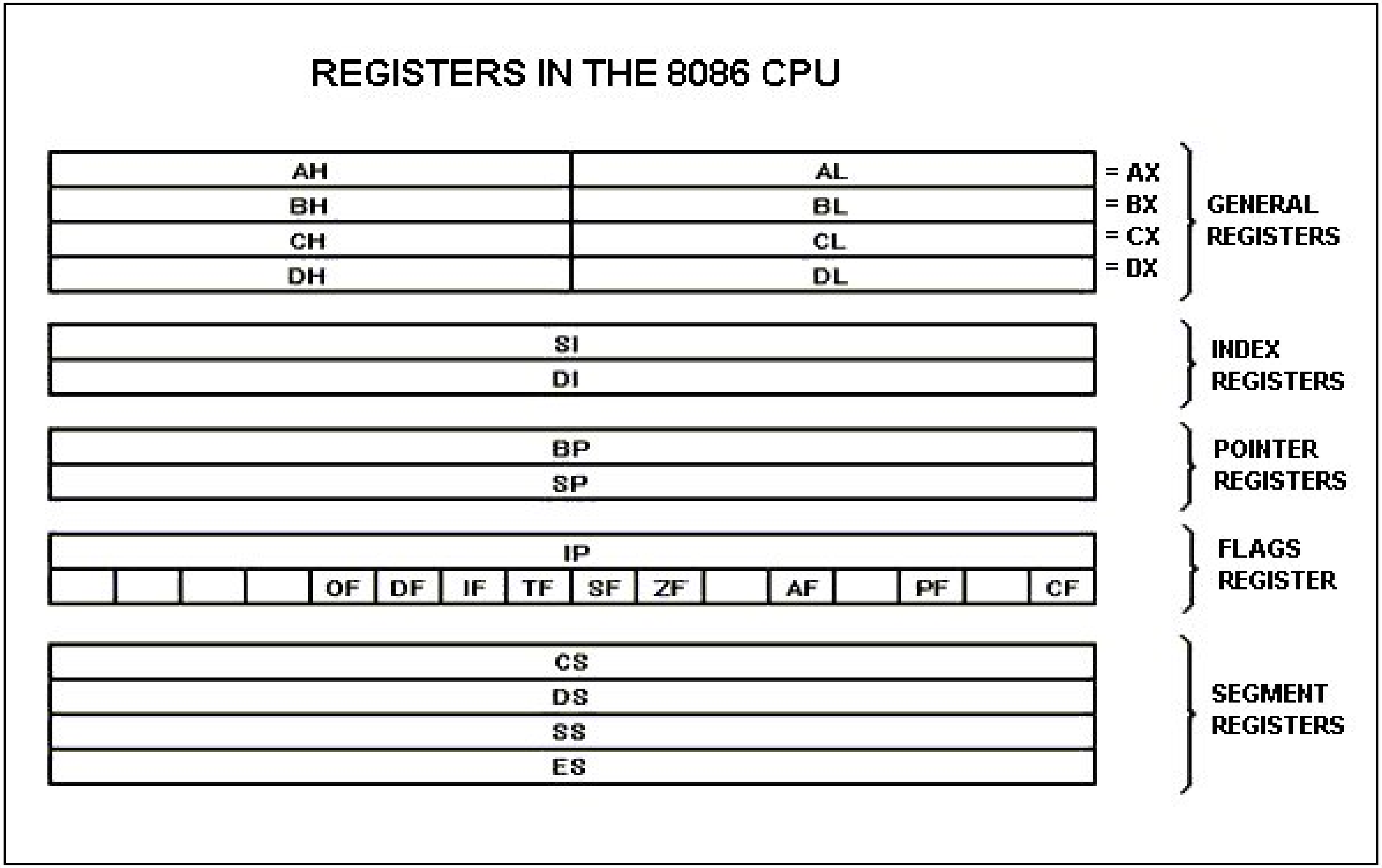
- The functions of execution unit are:

□ To tell BIU where to fetch the instructions or data from.

□ To decode the instructions.

□ To execute the instructions.

- The EU contains the control circuitry to perform various internal operations. A decoder in EU decodes the instruction fetched memory to generate different internal or external control signals required to perform the operation. EU has 16-bit ALU, which can perform arithmetic and logical operations on 8-bit as well as 16-bit



REGISTERS IN THE 8086 CPU

AH	AL	= AX	GENERAL REGISTERS
BH	BL	= BX	
CH	CL	= CX	
DH	DL	= DX	

SI	INDEX REGISTERS
DI	

BP	POINTER REGISTERS
SP	

IP

				OF	DF	IF	TF	SF	ZF		AF		PF		CF

FLAGS REGISTER

CS	SEGMENT REGISTERS
DS	
SS	
ES	

General Purpose Registers

8086 CPU has 8 general purpose registers

AX - the accumulator register (divided into **AH / AL**):

- ➢ Generates shortest machine code, Arithmetic, logic and data transfer
- ➢ One number must be in AL or AX
- ➢ Multiplication & Division
- ➢ Input & Output

BX - the base address register (divided into **BH / BL**)

CX - the count register(divided into **CH / CL**):

- ➢ Iterative code segments using the LOOP instruction
- ➢ Repetitive operations on strings with the REP command
- ➢ Count (in CL) of bits to shift and rotate

DX - the data register (divided into **DH / DL**):

- ➢ DX:AX concatenated into 32-bit register for some MUL and DIV operations
- ➢ Specifying ports in some IN and OUT operations

General Purpose Registers

SI - source index register:

- Can be used for pointer addressing of data and as source in some string processing instructions
- Offset address relative to DS

DI - destination index register:

- Can be used for pointer addressing of data and as destination in some string processing instructions
- Offset address relative to ES

BP - base pointer:

- Primarily used to access parameters passed via the stack
- Offset address relative to SS

SP - stack pointer:

- Always points to top item on the stack
- Offset address relative to SS
- Always points to word (byte at even address) and an empty stack will had SP = FFFEh

Segment Register

CS (Code segment)

- points at the segment containing the current program.

DS (Data Segment)

- generally points at segment where variables are defined.

ES (Extra Segment)

- extra segment register, it's up to a coder to define its usage.

SS (Stack Segment)

- points at the segment containing the stack.

Although it is possible to store any data in the segment registers, this is never a good idea. The segment registers have a very special purpose - pointing at accessible blocks of memory.

Segment Register

Segment registers work together with general purpose register to access any memory value.

For example if we would like to access memory at the physical address **12345h**(hexadecimal),

we could set the **DS = 1230h** and **SI = 0045h**. This way we can access much more memory than with a single register, which is limited to 16 bit values.

The CPU makes a calculation of the physical address by multiplying the segment register by 10h and adding the general purpose register to it (1230h * 10h + 45h = 12345h):

The address formed with 2 registers is called an **effective address**.

By default **BX, SI** and **DI** registers work with **DS** segment register;

BP and **SP** work with **SS** segment register.

Other general purpose registers cannot form an effective address.

Also, although **BX** can form an effective address, **BH** and **BL** cannot.

Special Purpose Register

- **IP** - the instruction pointer:
- Always points to next instruction to be executed
- Offset address relative to CS
- **IP** register always works together with **CS** segment register and it points to currently executing instruction.

Flag Register

Flags Register - determines the current state of the processor. They are modified automatically by CPU after mathematical operations, this allows to determine the type of the result, and to determine conditions to transfer control to other parts of the program.

Generally you cannot access these registers directly.

8086 has 9 flags and they are divided into two categories:

1.Conditional Flags &

2.Control Flags

				OF	DF	IF	TF	SF	ZF		AF		PF		CF

15 0

Conditional Flags

Conditional flags represent result of last arithmetic or logical instruction executed. Conditional flags are as follows:

1)Carry Flag (CF): This flag indicates an overflow condition for unsigned integer arithmetic. It is also used in multiple-precision arithmetic.

2)Auxiliary Flag (AF): If an operation performed in ALU generates a carry/barrow from lower nibble (i.e. D0 – D3) to upper nibble (i.e. D4 – D7), the AF flag is set i.e. carry given by D3 bit to D4 is AF flag. This is not a general-purpose flag, it is used internally by the processor to perform Binary to BCD conversion.

3)Parity Flag (PF): This flag is used to indicate the parity of result. If lower order 8-bits of the result contains even number of 1's, the Parity Flag is set and for odd number of 1's, the Parity Flag is reset.

4)Zero Flag (ZF): It is set; if the result of arithmetic or logical operation is zero else it is reset.

5) **Sign Flag (SF):** In sign magnitude format the sign of number is indicated by MSB bit. If the result of operation is negative, sign flag is set.

6)Overflow Flag (OF): It occurs when signed numbers are added or subtracted. An OF indicates that the result has exceeded the capacity of machine

Control Flags

Control flags are set or reset deliberately to control the operations of the execution unit.

Control flags are as follows:

1)Trap Flag (TP):

It is used for single step control.

It allows user to execute one instruction of a program at a time for debugging.

When trap flag is set, program can be run in single step mode.

2)Interrupt Flag (IF):

It is an interrupt enable/disable flag.

If it is set, the maskable interrupt of 8086 is enabled and if it is reset, the interrupt is disabled.

It can be set by executing instruction STI (SET INTERRUPT FLAG) and can be cleared by executing CLI (CLEAR INTERRUPT FLAG) instruction.

3)Direction Flag (DF):

It is used in string operation.

If it is set, string bytes are accessed from higher memory address to lower memory address.

When it is reset, the string bytes are accessed from lower memory address to higher memory address.

THANK YOU

Sri SAI RAM ENGINEERING COLLEGE
INSTITUTE OF TECHNOLOGY
West Tambaram, Chennai - 44

SAIRAM DIGITAL RESOURCES

UNIT NO 1

THE 8086 MICROPROCESSOR

1.3 Instruction set

YEAR III

SEM VI

20ECPC402

MICROCONTROLLERS AND EMBEDDED SYSTEMS

ELECTRONICS & COMMUNICATION ENGINEERING

INSTRUCTION SET OF 8086

- An instruction is a binary pattern designed inside a microprocessor to perform a specific function
- The sequence of commands used to instruct a microcomputer is called a program.
- Each command in a program is called an instruction.
- The entire group of instruction that microprocessor supports is called instruction set
- 8086 has more than 20,000instructions
- Instruction are command to microprocessor.
- **Eg.** A program written in machine language is referred to as machine code

ADD AX, BX

INSTRUCTION SET OF 8086

The 8086 microprocessor supports 8 types of instructions

- Data Transfer Instructions
- Arithmetic Instructions
- Bit Manipulation Instructions
- String Instructions
- Program Execution Transfer Instructions (Branch & Loop Instructions)
- Processor Control Instructions
- Control Instructions
- Interrupt Instructions

INSTRUCTION SET OF 8086

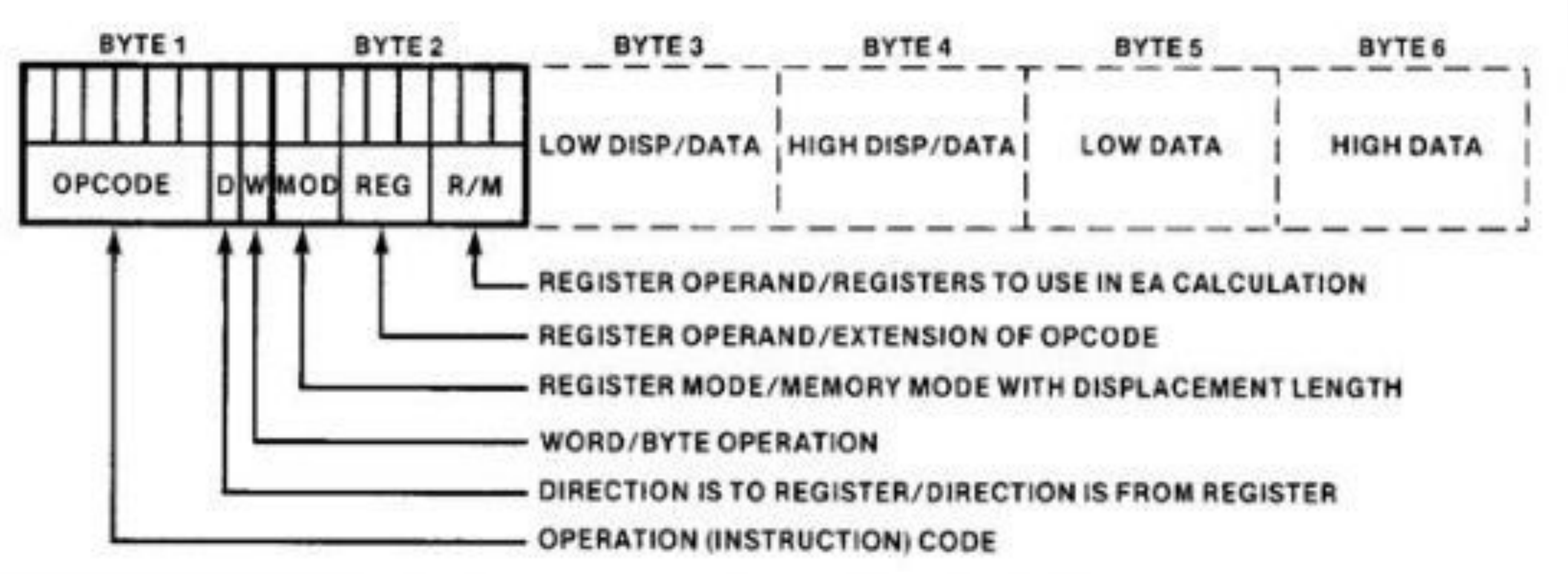
Data Transfer Instructions

These instructions are used to transfer the data from the source operand to the destination operand.

Instruction to transfer a word

- **MOV** − Used to copy the byte or word from the provided source to the provided destination.
- **PPUSH** − Used to put a word at the top of the stack.
- **POP** − Used to get a word from the top of the stack to the provided location.
- **PUSHA** − Used to put all the registers into the stack.
- **POPA** − Used to get words from the stack to all registers.
- **XCHG** − Used to exchange the data from two locations.
- **XLAT** − Used to translate a byte in AL using a table in the memory.

INSTRUCTION TEMPLATE



BYTE 1
BYTE 2
BYTE 3
BYTE 4
BYTE 5
BYTE 6

OPCODE
D
W
MOD
REG
R/M
LOW DISP/DATA
HIGH DISP/DATA
LOW DATA
HIGH DATA

REGISTER OPERAND/REGISTERS TO USE IN EA CALCULATION

REGISTER OPERAND/EXTENSION OF OPCODE

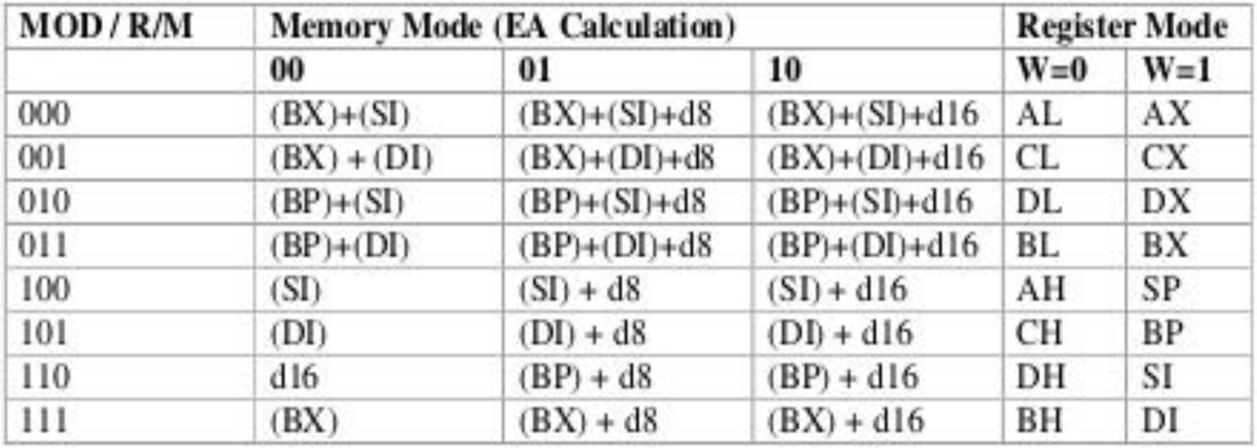
REGISTER MODE/MEMORY MODE WITH DISPLACEMENT LENGTH

WORD/BYTE OPERATION

DIRECTION IS TO REGISTER/DIRECTION IS FROM REGISTER

OPERATION (INSTRUCTION) CODE

Binary code MOV
=100010
MOV CH,BL
1000100011011101
88DDH



MOD / R/M	Memory Mode (EA Calculation)			Register Mode	
	00	01	10	W=0	W=1
000	(BX)+(SI)	(BX)+(SI)+d8	(BX)+(SI)+d16	AL	AX
001	(BX) + (DI)	(BX)+(DI)+d8	(BX)+(DI)+d16	CL	CX
010	(BP)+(SI)	(BP)+(SI)+d8	(BP)+(SI)+d16	DL	DX
011	(BP)+(DI)	(BP)+(DI)+d8	(BP)+(DI)+d16	BL	BX
100	(SI)	(SI) + d8	(SI) + d16	AH	SP
101	(DI)	(DI) + d8	(DI) + d16	CH	BP
110	d16	(BP) + d8	(BP) + d16	DH	SI
111	(BX)	(BX) + d8	(BX) + d16	BH	DI

Data transfer Instruction

Mnemonic	Meaning	Format	Operation		Flags affected
MOV	Move	Mov D,S	(S)	(D)	None

Destination	Source
Memory	Accumulator
Accumulator	Memory
Register	Register
Register	Memory
Memory	Register
Register	Immediate
Memory	Immediate
Seg reg	Reg 16
Seg reg	Mem 16
Reg 16	Seg reg
Memory	Seg reg

EX: MOV AL, BL

INSTRUCTION SET OF 8086

Instructions for input and output port transfer

IN − Used to read a byte or word from the provided port to the accumulator.

OUT − Used to send out a byte or word from the accumulator to the provided port.

Instructions to transfer the address

LEA − Used to load the address of operand into the provided register.

LDS − Used to load DS register and other provided register from the memory

LES − Used to load ES register and other provided register from the memory.

Instructions to transfer flag registers

LAHF − Used to load AH with the low byte of the flag register.

SAHF − Used to store AH register to low byte of the flag regi**PUSHF** − Used to copy the flag register at the top of the stack.

POPF − Used to copy a word at the top of the stack to the flag register.

Arithmetic Instructions

These instructions are used to perform arithmetic operations like addition, subtraction, multiplication, division, etc.

Instructions to perform addition

ADD − Used to add the provided byte to byte/word to word.

ADC − Used to add with carry.

INC − Used to increment the provided byte/word by 1.

AAA − Used to adjust ASCII after addition.

DAA − Used to adjust the decimal after the addition/subtraction operation.

Instructions to perform subtraction

SUB − Used to subtract the byte from byte/word from word.

SBB − Used to perform subtraction with borrow.

DEC − Used to decrement the provided byte/word by 1.

CMP − Used to compare 2 provided byte/word.

AAS − Used to adjust ASCII codes after subtraction.

DAS − Used to adjust decimal after subtraction.

Arithmetic Instructions

Instruction to perform multiplication

- **MUL** – Used to multiply unsigned byte by byte/word by word.
- **IMUL** – Used to multiply signed byte by byte/word by word.
- **AAM** – Used to adjust ASCII codes after multiplication.

Instructions to perform division

- **DIV** – Used to divide the unsigned word by byte or unsigned double word by word.
- **IDIV** – Used to divide the signed word by byte or signed double word by word.
- **AAD** – Used to adjust ASCII codes after division.
- **CBW** – Used to fill the upper byte of the word with the copies of sign bit of the lower byte.
- **CWD** – Used to fill the upper word of the double word with the sign bit of the lower word.

Bit Manipulation Instructions

These instructions are used to perform operations where data bits are involved, i.e. operations like logical, shift, etc.

Instructions to perform logical operation

- **NOT** – Used to invert each bit of a byte or word.
- **AND** – Used for adding each bit in a byte/word with the corresponding bit in another byte/word.
- **OR** – Used to multiply each bit in a byte/word with the corresponding bit in another byte/word.
- **XOR** – Used to perform Exclusive-OR operation over each bit in a byte/word with the corresponding bit in another byte/word.
- **TEST** – Used to add operands to update flags, without affecting operands.

Instructions to perform shift operations

- SHL/SAL – Used to shift bits of a byte/word towards left and put zero(S) in LSBs.
- SHR – Used to shift bits of a byte/word towards the right and put zero(S) in MSBs.
- SAR – Used to shift bits of a byte/word towards the right and copy the old MSB into the new MSB.

Instructions to perform rotate operations

- ROL – Used to rotate bits of byte/word towards the left, i.e. MSB to LSB and to Carry Flag [CF].
- ROR – Used to rotate bits of byte/word towards the right, i.e. LSB to MSB and to Carry Flag [CF].
- RCR – Used to rotate bits of byte/word towards the right, i.e. LSB to CF and CF to MSB.
- RCL – Used to rotate bits of byte/word towards the left, i.e. MSB to CF and CF to LSB.

THANK YOU ...

Sri SAI RAM ENGINEERING COLLEGE
INSTITUTE OF TECHNOLOGY
West Tambaram, Chennai - 44

SAIRAM DIGITAL RESOURCES

YEAR II

SEM III

EE8351

DIGITAL LOGIC CIRCUITS
Common to EEE,EIE,ICE

UNIT NO 1

NUMBER SYSTEMS AND DIGITAL LOGIC FAMILIES

- REVIEW OF NUMBER SYSTEMS

ELECTRICAL & ELECTRONICS ENGINEERING

Why number system is used?

Computer Numeral System (Number System in Computers)

When we type any letter or word, the computer translates them into numbers since computers can understand only numbers. A computer can understand only a few symbols called digits, and these symbols describe different values depending on the position they hold in the number. In general, the binary number system is used in computers. However, the octal, decimal and hexadecimal systems are also used sometimes.

Assume that the base or radix of a number system is ‘r’, (for example base 8)

The numbers present in that number system are ranging from zero to r-1. (0,1,2,3,4,5,6,7)

The total numbers present in that number system is ‘r’. (8)

So, we will get various number systems, by choosing the values of radix as greater than or equal to two.

Common Number Systems

Number System	Base	Representation
Decimal	10	0,1,2,3,4,5,6,7,8,9
Binary	2	0,1
Octal	8	0,1,2,3,4,5,6,7
Hexa Decimal	16	0,1,2,3,4,5,6,7,8,9, A,B,C,D,E,F

Example for Different Number Systems

Decimal	Binary	Octal	Hexadecimal
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8

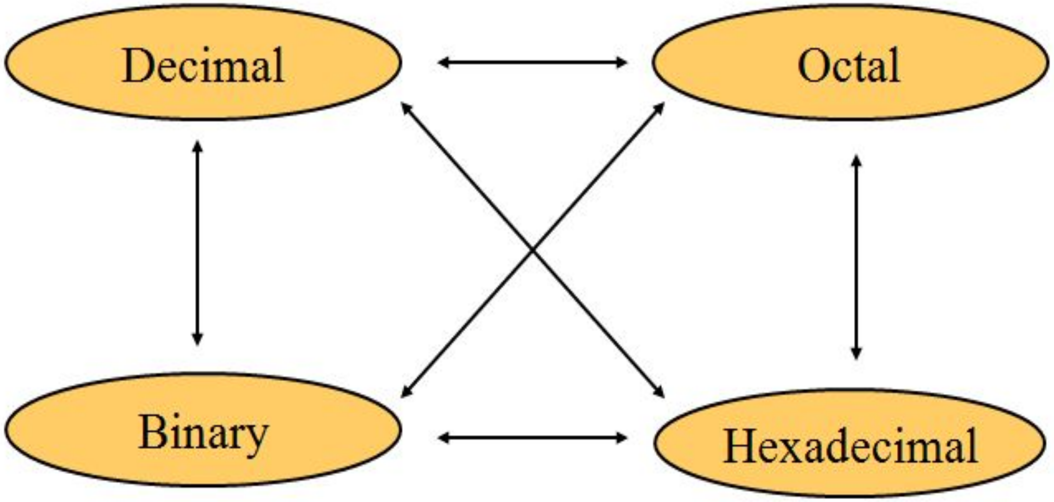
Example for Different Number Systems

Decimal	Binary	Octal	Hexadecimal
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Example for Different Number Systems

Decimal	Binary	Octal	Hexadecimal
16	10000	20	10
17	10001	21	11
18	10010	22	12
19	10011	23	13
20	10100	24	14
21	10101	25	15
22	10110	26	16

Conversion Possibilities



Base Conversion - (Decimal to binary)

Technique

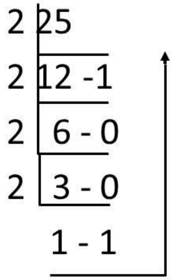
- ✔ Divide by two, keep track of the remainder
- ✔ First remainder is bit 0 (LSB, least-significant bit)
- ✔ Second remainder is bit 1
- ✔ Etc.

Decimal to binary

$$25_{10} = 11001_2 = 31_8 = 19_{16}$$

Base

Conversion of 25_{10} to Binary



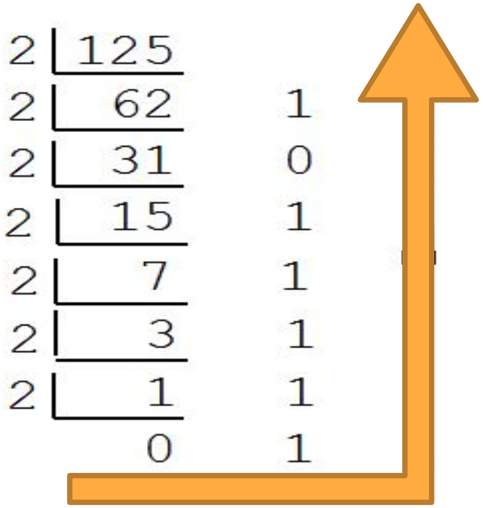
2 | 25
2 | 12 - 1
2 | 6 - 0
2 | 3 - 0
1 - 1

$25_{10}=11001_2$

For decimal to octal conversion divide 25 by ‘8’ instead of dividing 25 by ‘2’

Decimal to Binary

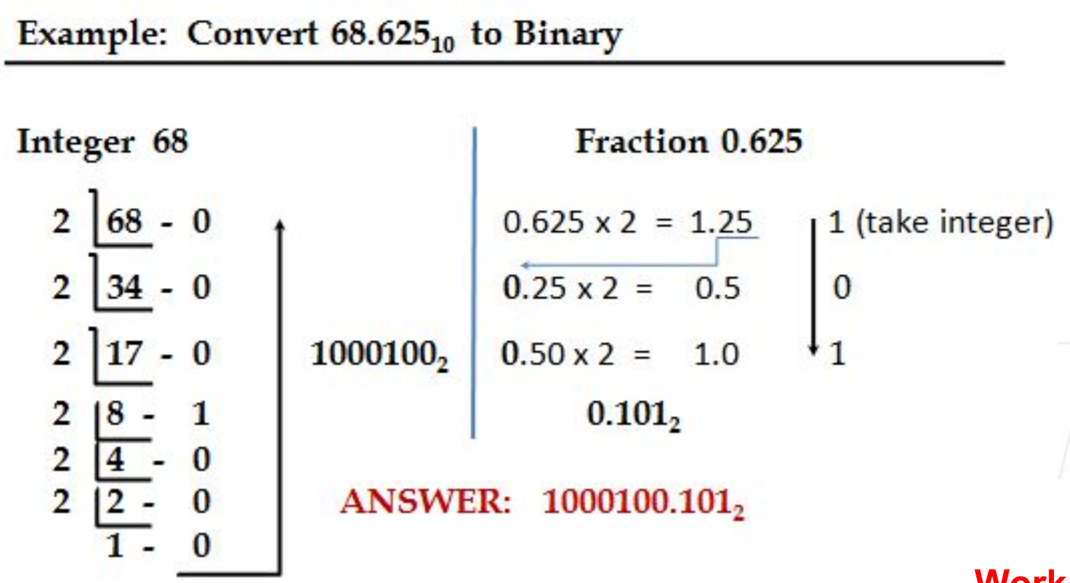
$125_{10} = ?_2$



2	125	
2	62	1
2	31	0
2	15	1
2	7	1
2	3	1
2	1	1
	0	1

$125_{10} = 1111101_2$

Decimal to Binary with decimal point



Example: Convert 68.625_{10} to Binary

Integer 68

2 | 68 - 0
2 | 34 - 0
2 | 17 - 0
2 | 8 - 1
2 | 4 - 0
2 | 2 - 0
1 - 0

1000100_2

Fraction 0.625

0.625 x 2 = 1.25 1 (take integer)
0.25 x 2 = 0.5 0
0.50 x 2 = 1.0 1

0.101_2

ANSWER: 1000100.101_2

Work out

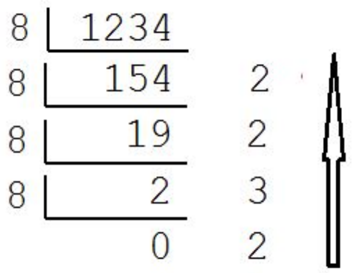
$(0.4375)_{10} = X_2$

2 x 0.4375 = 0.8750 => 0
2 x 0.8750 = 1.750 => 1
2 x 0. 750 = 1.5 => 1
2 x 0.5 = 1.0 => 1

$(0.4375)_{10} = (0.0111)_2$

Decimal to Octal

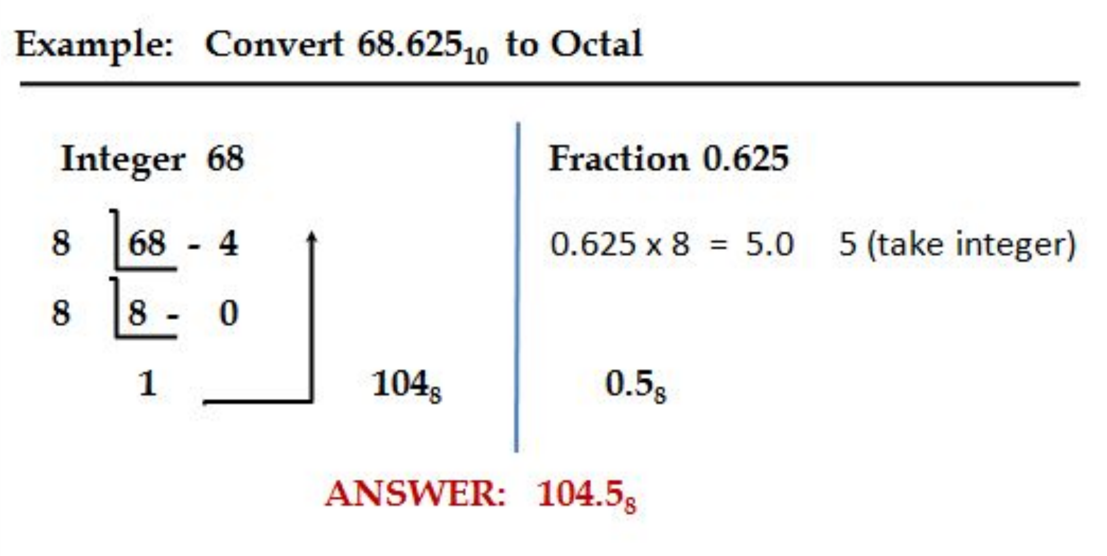
$1234_{10} = ?_8$



$$\begin{array}{r|r r}
8 & 1234 & \\
8 & 154 & 2 \\
8 & 19 & 2 \\
8 & 2 & 3 \\
 & 0 & 2
\end{array}$$

$1234_{10} = 2322_8$

Decimal to octal conversion: with decimal point



Example: Convert 68.625_{10} to Octal

Integer 68

$8 \,|\, 68$ - 4

$8 \,|\, 8$ - 0

1

104_8

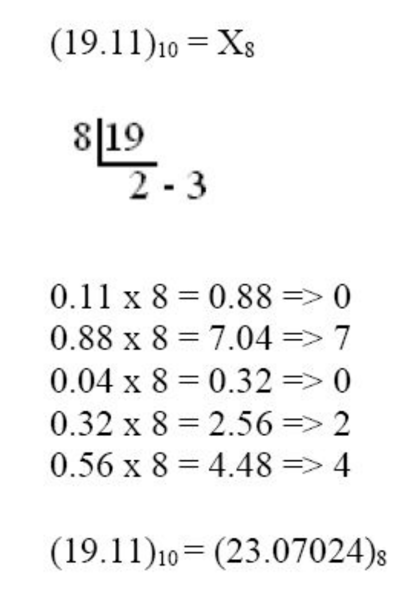
Fraction 0.625

0.625 x 8 = 5.0 5 (take integer)

0.5_8

ANSWER: 104.5_8

Work out



$(19.11)_{10} = X_8$

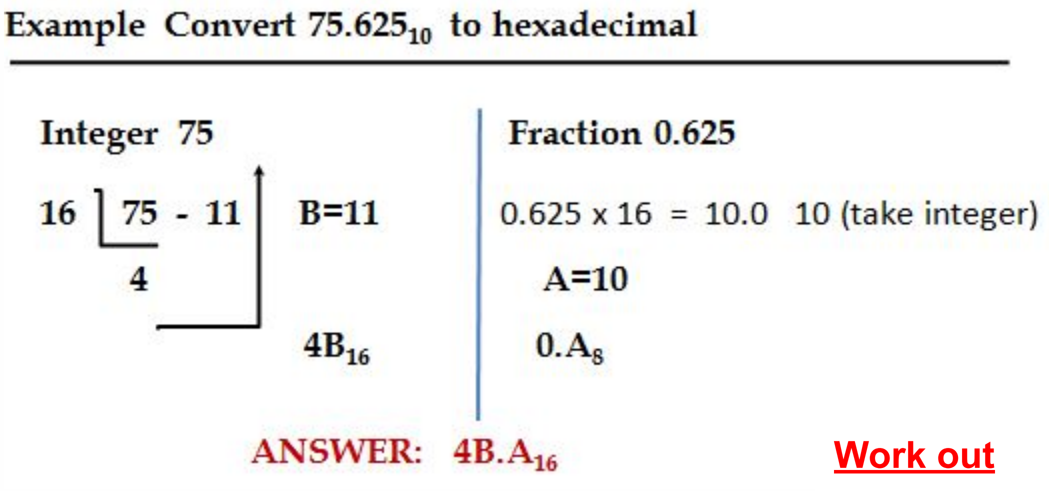
$8 \,|\, 19$

2 - 3

0.11 x 8 = 0.88 => 0
0.88 x 8 = 7.04 => 7
0.04 x 8 = 0.32 => 0
0.32 x 8 = 2.56 => 2
0.56 x 8 = 4.48 => 4

$(19.11)_{10} = (23.07024)_8$

Decimal to Hexadecimal- Example



Example Convert 75.625_{10} to hexadecimal

Integer 75

16 | 75 - 11 B=11

4

$4B_{16}$

Fraction 0.625

0.625 x 16 = 10.0 10 (take integer)

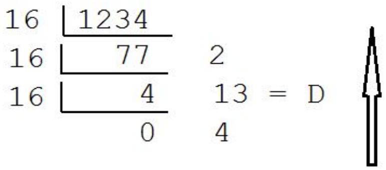
A=10

$0.A_8$

ANSWER: $4B.A_{16}$

Work out

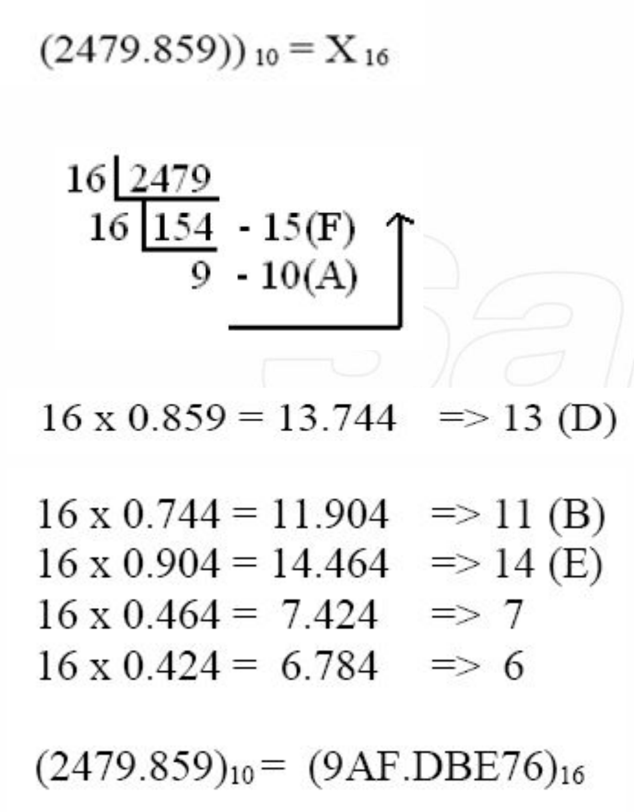
$1234_{10} = ?_{16}$



```
16 |1234
16 |  77     2
16 |   4    13 = D
       0     4
```

$1234_{10} = 4D2_{16}$

Decimal to hexadecimal conversion: with decimal point



$(2479.859))_{10} = X_{16}$

$$\begin{array}{r|l}
16 & 2479 \\
\hline
16 & 154 \quad - 15(F) \\
\hline
 & \;\;\; 9 \quad - 10(A)
\end{array}$$

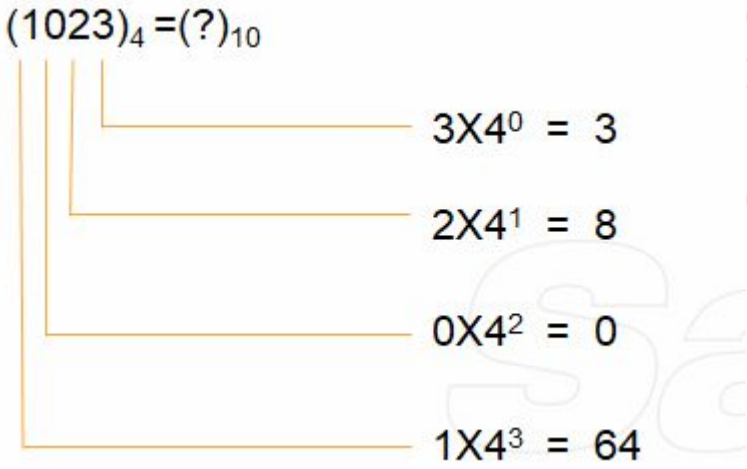
$16 \text{ x } 0.859 = 13.744 \quad => 13 \text{ (D)}$

$16 \text{ x } 0.744 = 11.904 \quad => 11 \text{ (B)}$
$16 \text{ x } 0.904 = 14.464 \quad => 14 \text{ (E)}$
$16 \text{ x } 0.464 = 7.424 \quad => 7$
$16 \text{ x } 0.424 = 6.784 \quad => 6$

$(2479.859)_{10} = (9AF.DBE76)_{16}$

Base 4 to decimal

$(1023)_4 = (?)_{10}$



$(1023)_4 = (?)_{10}$

$3X4^0 = 3$

$2X4^1 = 8$

$0X4^2 = 0$

$1X4^3 = 64$

Answer= 3+8+0+64= $(75)_{10}$

Practice Question

1. $(345.968)_{10} = (?)_2 = (?)_8 = (?)_{16}$

2. A system takes decimal input 154.9034 and encrypt the information in Octal form and send it to Module 1, in Binary form to Module 2 and Hexadecimal form to Module Show the information in Module 1, Module 2, and Module 3

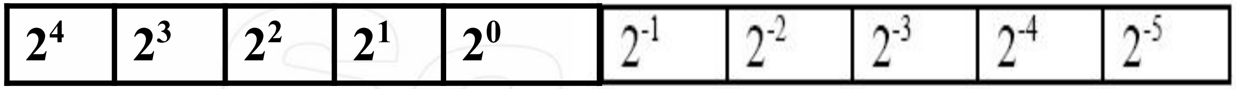
Base Conversions- Binary

Technique

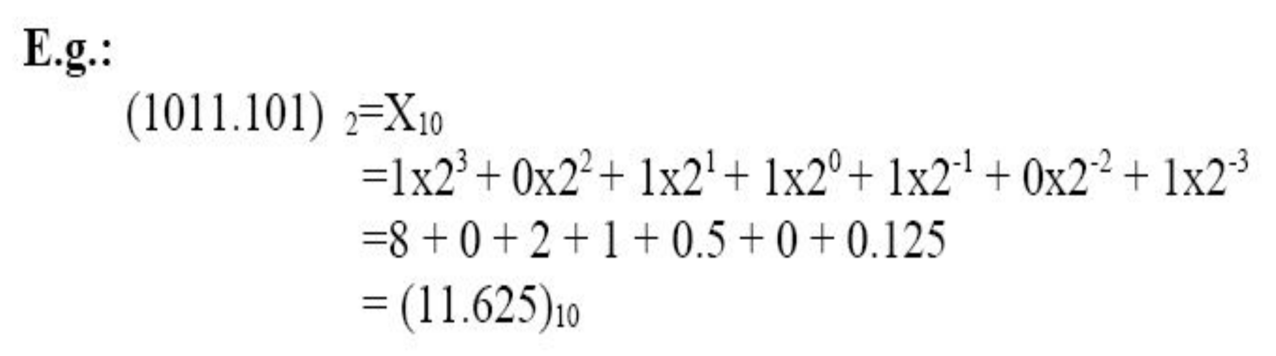
- ✔ Multiply each bit by 2^n, where *n* is the “weight” of the bit
- ✔ The weight is the position of the bit, starting from 0 on the right
- ✔ Add the results
- ✔ If any other base to decimal conversion needed then instead of 2, replace with corresponding base

Binary to decimal conversion:

The weights of Binary Number System are given as follows,



2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}



E.g.:

$$(1011.101)_2 = X_{10}$$

$$= 1x2^3 + 0x2^2 + 1x2^1 + 1x2^0 + 1x2^{-1} + 0x2^{-2} + 1x2^{-3}$$

$$= 8 + 0 + 2 + 1 + 0.5 + 0 + 0.125$$

$$= (11.625)_{10}$$

Binary to Decimal - Example

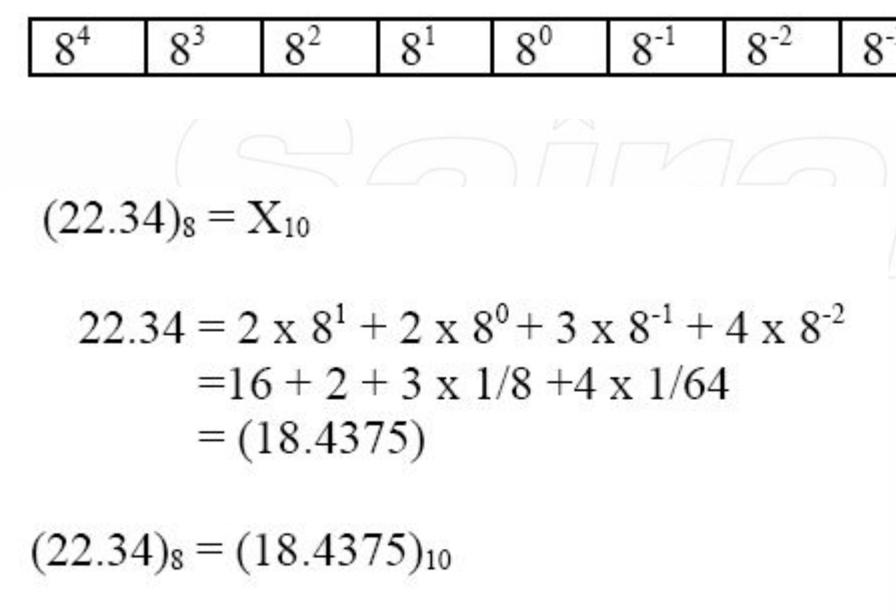
Weight 0

Base

$$
\begin{array}{lcccccc}
101011_2 \Rightarrow & 1 & \text{x} & 2^0 & = & 1 \\
 & 1 & \text{x} & 2^1 & = & 2 \\
 & 0 & \text{x} & 2^2 & = & 0 \\
 & 1 & \text{x} & 2^3 & = & 8 \\
 & 0 & \text{x} & 2^4 & = & 0 \\
 & 1 & \text{x} & 2^5 & = & 32 \\
\hline
 & & & & & 43_{10}
\end{array}
$$

Octal to decimal conversion:

The weight of digital position in octal number is as follows



8^4	8^3	8^2	8^1	8^0	8^{-1}	8^{-2}	8^{-3}

$(22.34)_8 = X_{10}$

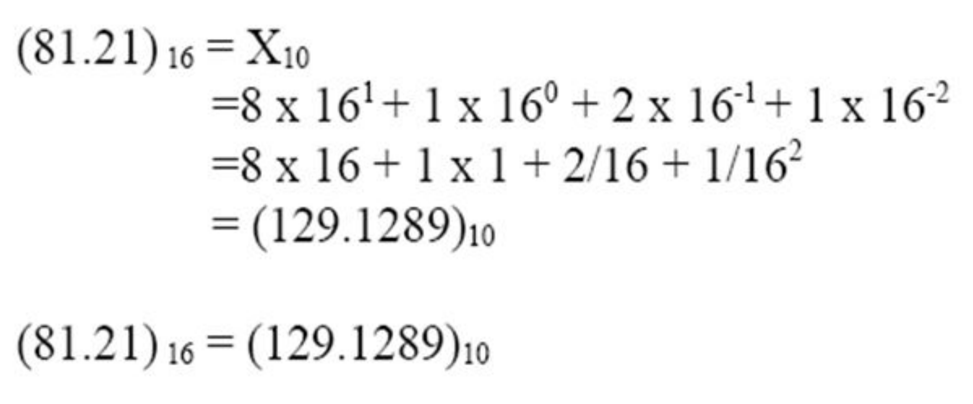
$$\begin{aligned} 22.34 &= 2 \text{ x } 8^1 + 2 \text{ x } 8^0 + 3 \text{ x } 8^{-1} + 4 \text{ x } 8^{-2} \\ &=16 + 2 + 3 \text{ x } 1/8 + 4 \text{ x } 1/64 \\ &= (18.4375) \end{aligned}$$

$(22.34)_8 = (18.4375)_{10}$

Octal to Decimal

$$
\begin{array}{lrcl}
724_8 \Rightarrow & 4 \times 8^0 & = & 4 \\
 & 2 \times 8^1 & = & 16 \\
 & 7 \times 8^2 & = & 448 \\
 & & & \text{---------} \\
 & & & 468_{10}
\end{array}
$$

Hexadecimal to decimal conversion



$$(81.21)_{16} = X_{10}$$

$$= 8 \text{ x } 16^{1} + 1 \text{ x } 16^{0} + 2 \text{ x } 16^{-1} + 1 \text{ x } 16^{-2}$$

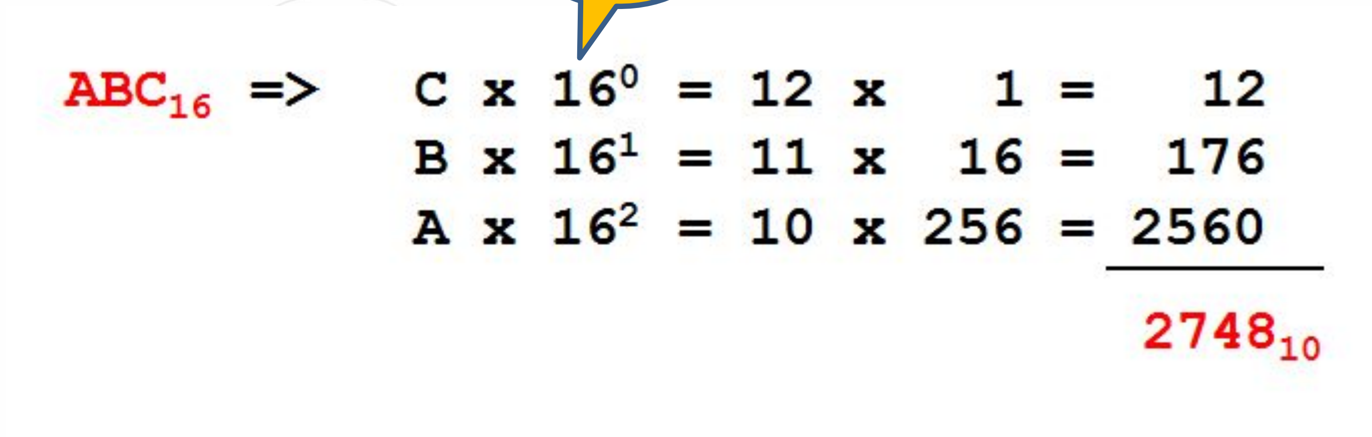
$$= 8 \text{ x } 16 + 1 \text{ x } 1 + 2/16 + 1/16^{2}$$

$$= (129.1289)_{10}$$

$$(81.21)_{16} = (129.1289)_{10}$$

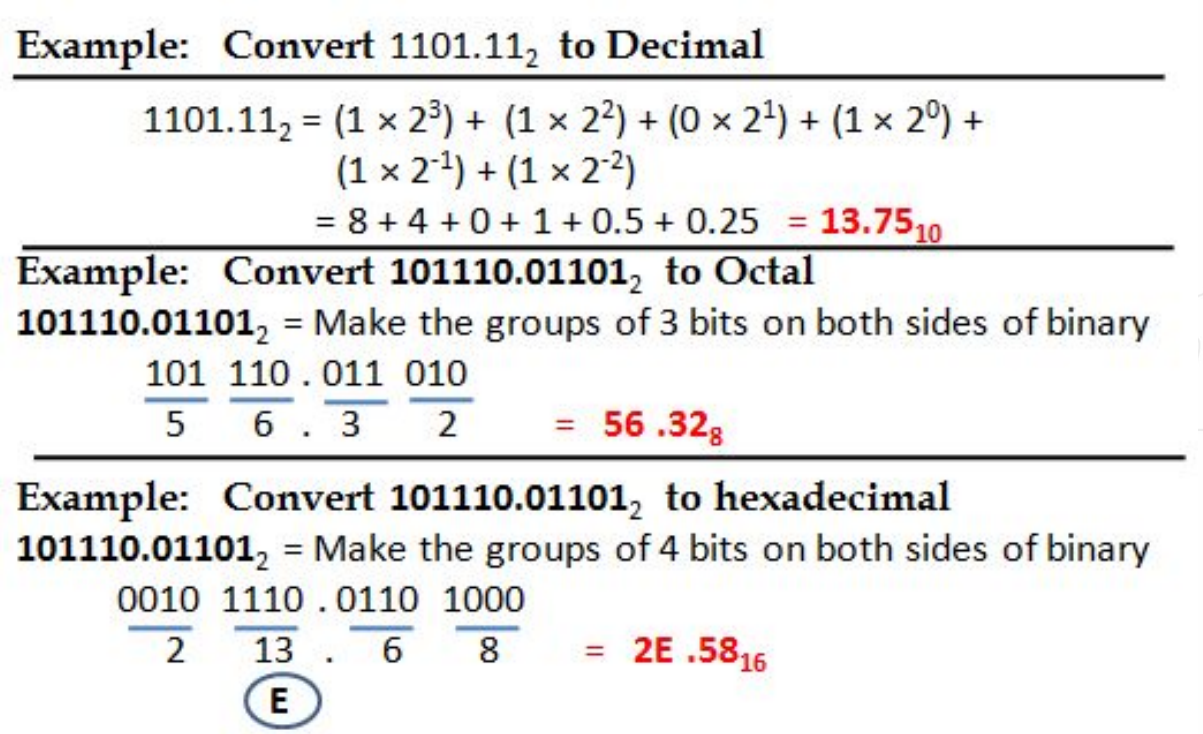
Hexadecimal to Decimal

Bas
e



$$
\begin{array}{lllllllll}
ABC_{16} \Rightarrow & C & \times & 16^0 & = 12 & \times & 1 & = & 12 \\
 & B & \times & 16^1 & = 11 & \times & 16 & = & 176 \\
 & A & \times & 16^2 & = 10 & \times & 256 & = & \underline{2560} \\
 & & & & & & & & 2748_{10}
\end{array}
$$

Summary with example



Example: Convert 1101.11_2 to Decimal

$$1101.11_2 = (1 \times 2^3) + (1 \times 2^2) + (0 \times 2^1) + (1 \times 2^0) + (1 \times 2^{-1}) + (1 \times 2^{-2})$$

$$= 8 + 4 + 0 + 1 + 0.5 + 0.25 = \mathbf{13.75_{10}}$$

Example: Convert 101110.01101_2 to Octal

$\mathbf{101110.01101_2}$ = Make the groups of 3 bits on both sides of binary

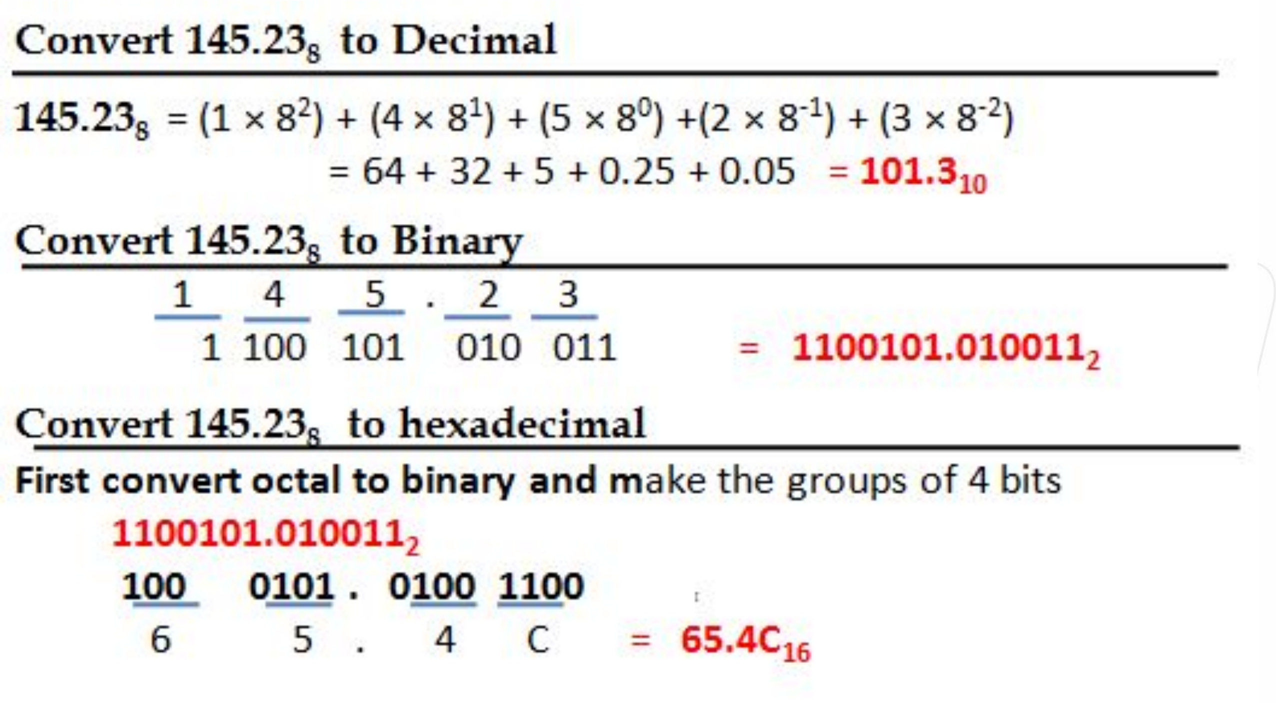
$$\underset{5}{\underline{101}}\ \ \underset{6}{\underline{110}}\ .\ \underset{3}{\underline{011}}\ \ \underset{2}{\underline{010}} \qquad = \mathbf{56\ .32_8}$$

Example: Convert 101110.01101_2 to hexadecimal

$\mathbf{101110.01101_2}$ = Make the groups of 4 bits on both sides of binary

$$\underset{2}{\underline{0010}}\ \ \underset{\overset{13}{\boxed{E}}}{\underline{1110}}\ .\ \underset{6}{\underline{0110}}\ \ \underset{8}{\underline{1000}} \qquad = \mathbf{2E\ .58_{16}}$$

Summary with example



Convert 145.23_8 to Decimal

$$145.23_8 = (1 \times 8^2) + (4 \times 8^1) + (5 \times 8^0) + (2 \times 8^{-1}) + (3 \times 8^{-2})$$

$$= 64 + 32 + 5 + 0.25 + 0.05 \quad = \mathbf{101.3_{10}}$$

Convert 145.23_8 to Binary

$$\underline{1} \quad \underline{4} \quad \underline{5} \;.\; \underline{2} \quad \underline{3}$$

$$1 \quad 100 \quad 101 \quad 010 \quad 011 \qquad = \mathbf{1100101.010011_2}$$

Convert 145.23_8 to hexadecimal

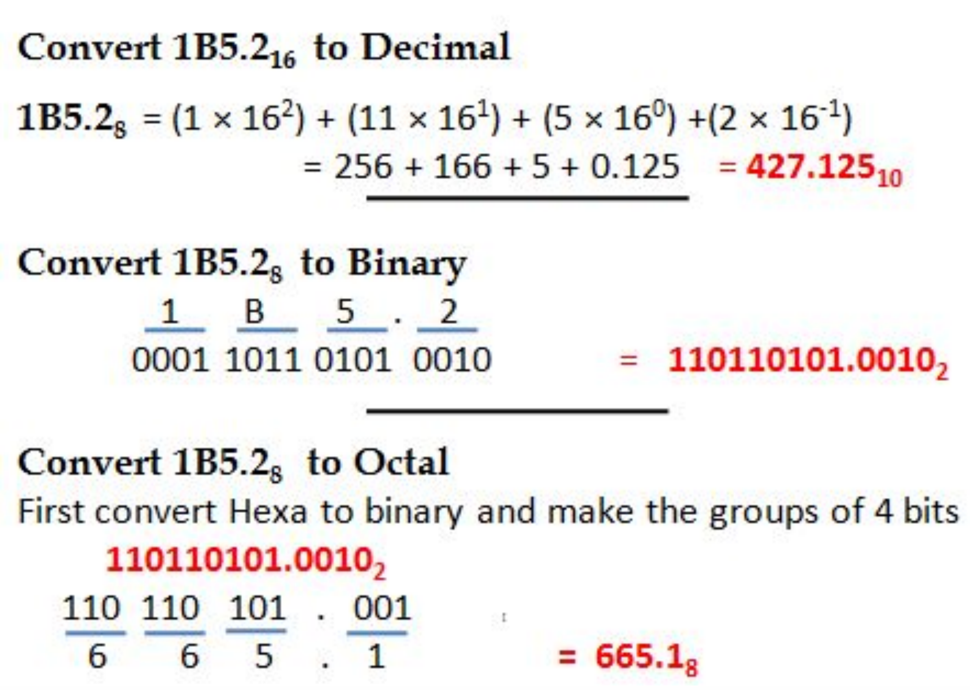
First convert octal to binary and make the groups of 4 bits

1100101.010011_2

$$\underline{100} \quad \underline{0101} \;.\; \underline{0100} \quad \underline{1100}$$

$$6 \quad 5 \;.\; 4 \quad C \qquad = \mathbf{65.4C_{16}}$$

Summary with example



Convert $1B5.2_{16}$ to Decimal

$1B5.2_8 = (1 \times 16^2) + (11 \times 16^1) + (5 \times 16^0) + (2 \times 16^{-1})$

$= 256 + 166 + 5 + 0.125 \quad = \mathbf{427.125_{10}}$

Convert $1B5.2_8$ to Binary

1 B 5 . 2

0001 1011 0101 0010 $= \mathbf{110110101.0010_2}$

Convert $1B5.2_8$ to Octal

First convert Hexa to binary and make the groups of 4 bits

$\mathbf{110110101.0010_2}$

110 110 101 . 001

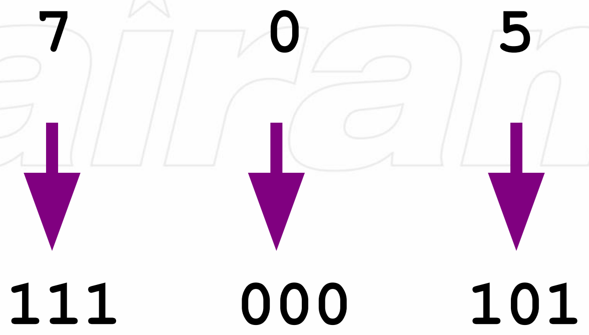
6 6 5 . 1 $= \mathbf{665.1_8}$

Exercise Problems

Octal to Binary

Convert each octal digit to a 3-bit equivalent binary representation

$705_8 = ?_2$


7
0
5
111
000
101

$705_8 = 111000101_2$

Exercise Problems

Octal to binary conversion:

$(57.127)_8 = X_2$

$$\begin{array}{cccccc} 5 & 7 & . & 1 & 2 & 7 \\ 101 & 111 & . & 001 & 010 & 111 \end{array}$$

$(57.127)_8 = (101111001010111)_2$

Binary to octal conversion:

$(1101101.11101)_2 = X_8$

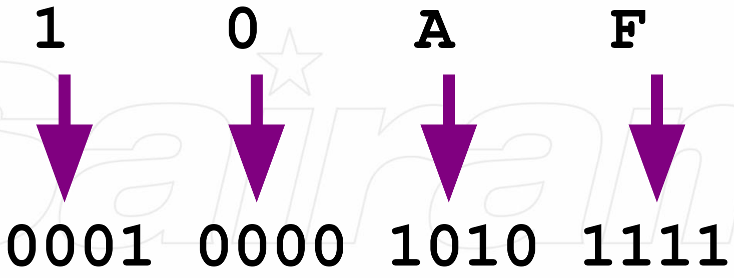
$$\begin{array}{cccccc} 001 & 101 & 101 & . & 111 & 010 \\ 1 & 5 & 5 & . & 7 & 2 \end{array}$$

$(1101101.11101)_2 = (155.72)_8$

Exercise Problems
Hexadecimal to Binary

Convert each hexadecimal digit to a 4-bit equivalent binary representation

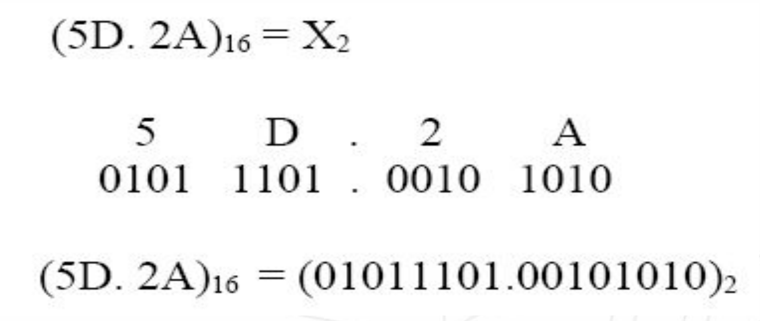
$10AF_{16} = ?_2$



$10AF_{16} = 0001000010101111_2$

Exercise Problems

Hexadecimal to binary conversion

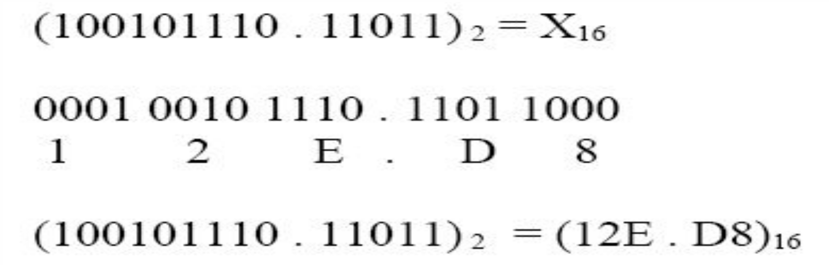


$$(5D.\ 2A)_{16} = X_2$$

$$\begin{array}{ccccc} 5 & D & . & 2 & A \\ 0101 & 1101 & . & 0010 & 1010 \end{array}$$

$$(5D.\ 2A)_{16} = (01011101.00101010)_2$$

Binary to hexadecimal conversion



$$(100101110\ .\ 11011)_2 = X_{16}$$

$$\begin{array}{cccccc} 0001 & 0010 & 1110 & . & 1101 & 1000 \\ 1 & 2 & E & . & D & 8 \end{array}$$

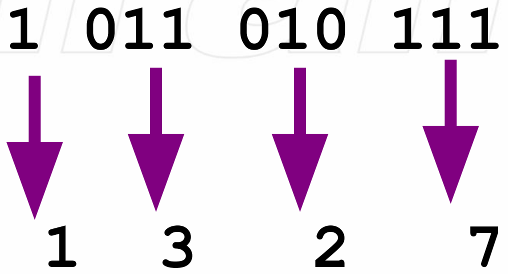
$$(100101110\ .\ 11011)_2 = (12E\ .\ D8)_{16}$$

Exercise Problems

Binary to Octal

- Technique
  - Group bits in threes, starting on right
  - Convert to octal digits

$$1011010111_2 = ?_8$$



1 011 010 111

1 3 2 7

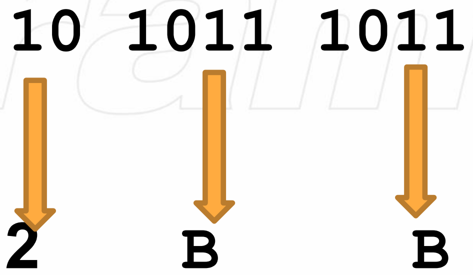
$$1011010111_2 = 1327_8$$

Binary to Hexadecimal

Technique

- Group bits in fours, starting on right
- Convert to hexadecimal digits

$1010111011_2 = ?_{16}$

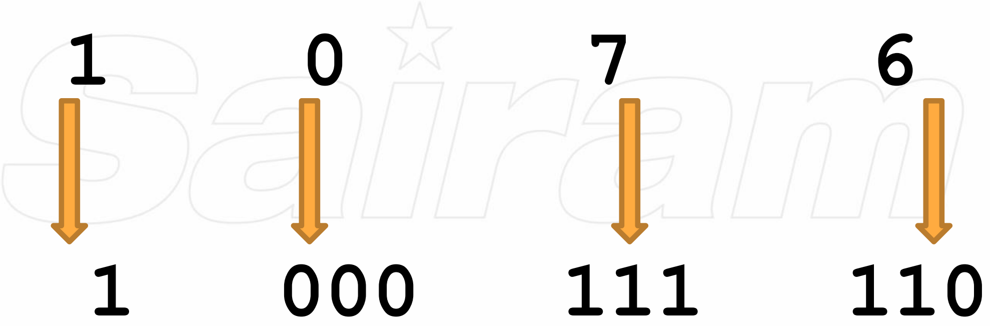


$1010111011_2 = 2BB_{16}$

Octal to Hexadecimal & Hexadecimal to Octal

<u>Technique</u> - Use binary as an intermediary

$1076_8 = ?_{16}$



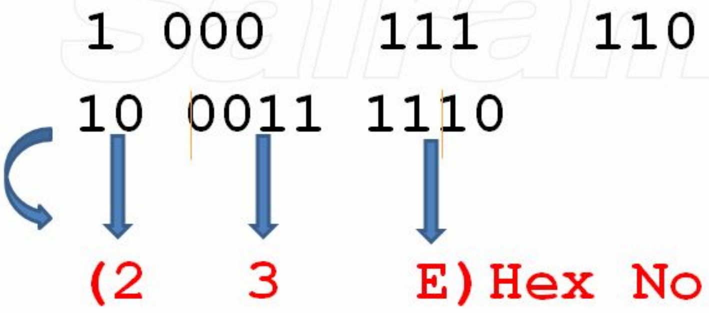
1 0 7 6

1 000 111 110

-> Write equivalent three digit binary values

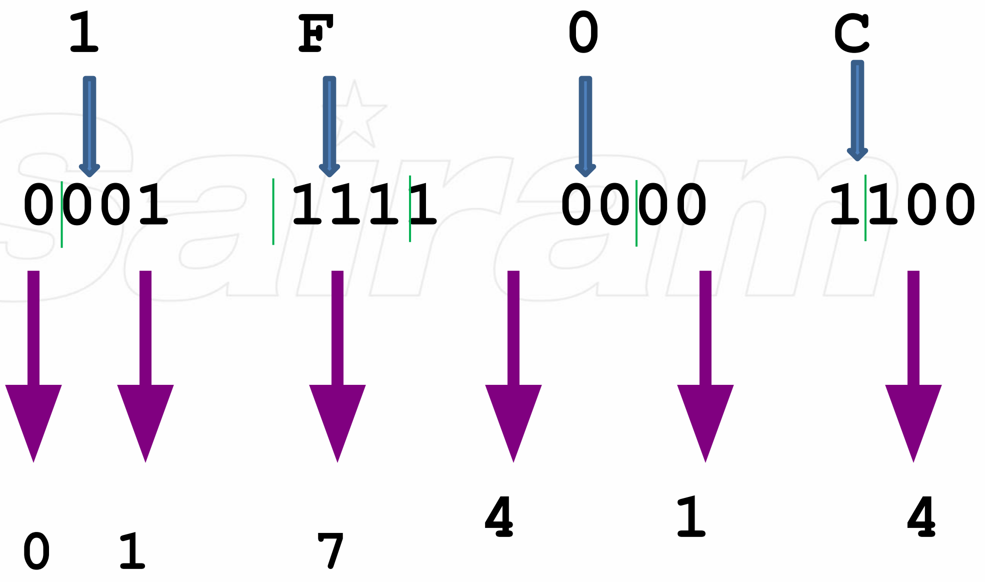
Continuation of $1076_8 = ?_{16}$

Group the binary numbers to 4 bits from right to left and write equivalent hex



```
1 000    111    110
10 0011 1110
(2   3     E)Hex No
```

Hexadecimal to Octal



1 F 0 C

0001 1111 0000 1100

0 1 7 4 1 4

$1F0C_{16} = 17414_8$

Practice Questions

$(10110110.00110101)_2=(?)_8$

$(10110110.10111)_2=(?)_{16}$

$(675234.3265)_8=(?)_{16}$

$(BABA.DAD)_{16}=(?)_8$

Test your knowledge

1. The largest decimal number that can be represented in binary with four bits □ 2^4 -1 = 15

2. The weight of the "1" in the binary number 1000 □ $2^3 = 8$

Arithmetic Operation on binary numbers

Computers perform arithmetic operations on binary numbers only.

- Addition
- Subtraction
- Multiplication
- Division

Binary Addition

A	B	Carry	Sum
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Example: 1011+1110

```
  1 1  1 ← Carry
    1 0 1 1
    1 1 1 0
   ----------
1   1  0  0 1 ← SUM
```

Practice questions

1. 10.001+11.110 =?
2. 1111+10010 =?

Answer: 1. 101.111

2. 100001

Binary Subtraction

A	B	Difference	Borrow
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Example: 1001-101

```
 1 0 0 1
   1 0  1
----------
   1 0  0
```

Difference

Practice questions

1. 10000-11 =?
2. 110.01-100.10 =?

Answer: 1. 1101
2. 1.11

Binary Multiplication

A	B	X
0	0	0
0	1	0
1	0	0
1	1	1

Practice questions

1. **1.01x10.1 =?**

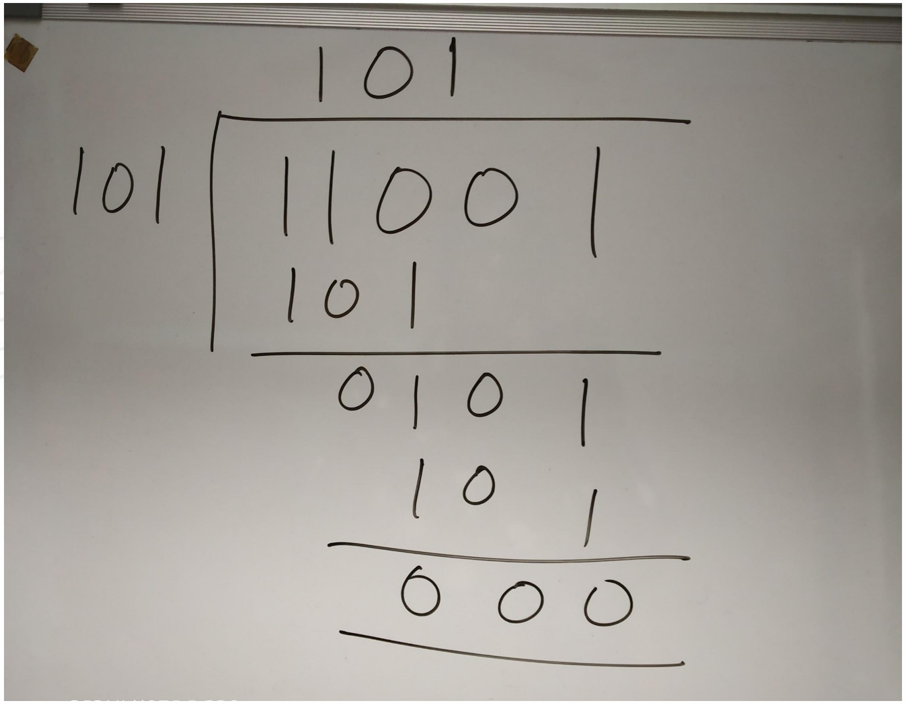
Ans: 11.001

Example: 1011x1101

```
          1 0 1 1
     x    1 1 0 1
          ----------
          1 0 1 1
       00 0 0
     1 0 1 1
   10 1 1
-----------------
1 0 0 01111
----------------
```

BINARY DIVISION

11001 /101



```
          101
      ________
  101 | 11001
        101
      _______
         0101
          101
        ______
          000
        ______
```

Complements:

1. **Radix complement**
   Example: i. For Binary 2’s complement
   ii. For decimal 10’s complement
1. **Diminished Radix complement**
   Example: i. For Binary 1’s complement
   ii. For decimal 9’s complement

Application:

Used For subtraction;

---> Instead of doing subtraction, we can find complement of subtrahend and add

1’s Complement:

For a binary number if all the bits are reversed that is 1’s are converted to 0’s and 0’s are converted to 1’s

For example 1’s complement of 1110001 is 0001110

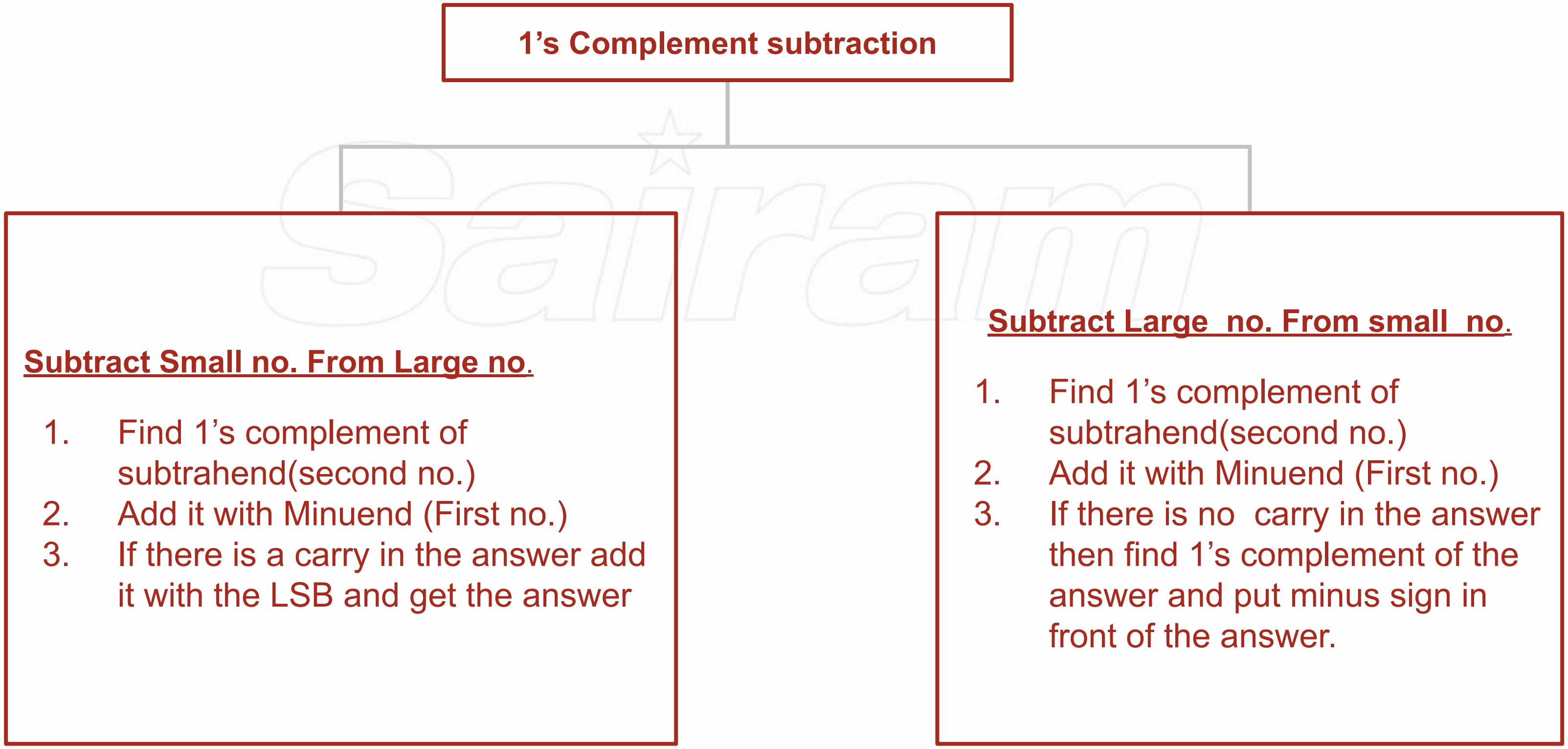
2’s Complement:

Find 1’s complement and add 1 to LSB

Example: 2’s complement of 1110001 is :

1’s complement of 1110001 is 0001110 + 1 = 0001111

1’s Complement Subtraction



1’s Complement subtraction

Subtract Small no. From Large no.

1. Find 1’s complement of subtrahend(second no.)
2. Add it with Minuend (First no.)
3. If there is a carry in the answer add it with the LSB and get the answer

Subtract Large no. From small no.

1. Find 1’s complement of subtrahend(second no.)
2. Add it with Minuend (First no.)
3. If there is no carry in the answer then find 1’s complement of the answer and put minus sign in front of the answer.

Example for 1’s complement Subtraction

Subtract $(11011)_2$ from $(11101)_2$ using 1’s complement. Also subtract $(11101)_2$ from $(11011)_2$

1 1 1 0 1
0 0 1 0 0 (+) ------> one’s complement of 11011
----
1 0 0 0 0 1
1-------> no carry
1
----
0 0 0 1 0 ----> Ans

Practice sums:

Do the following binary subtractions by 1’s complement method

1. 101101-110011
2. 110011-101101

1 1 0 1 1
0 0 0 1 0 (+)
----
1 1 1 0

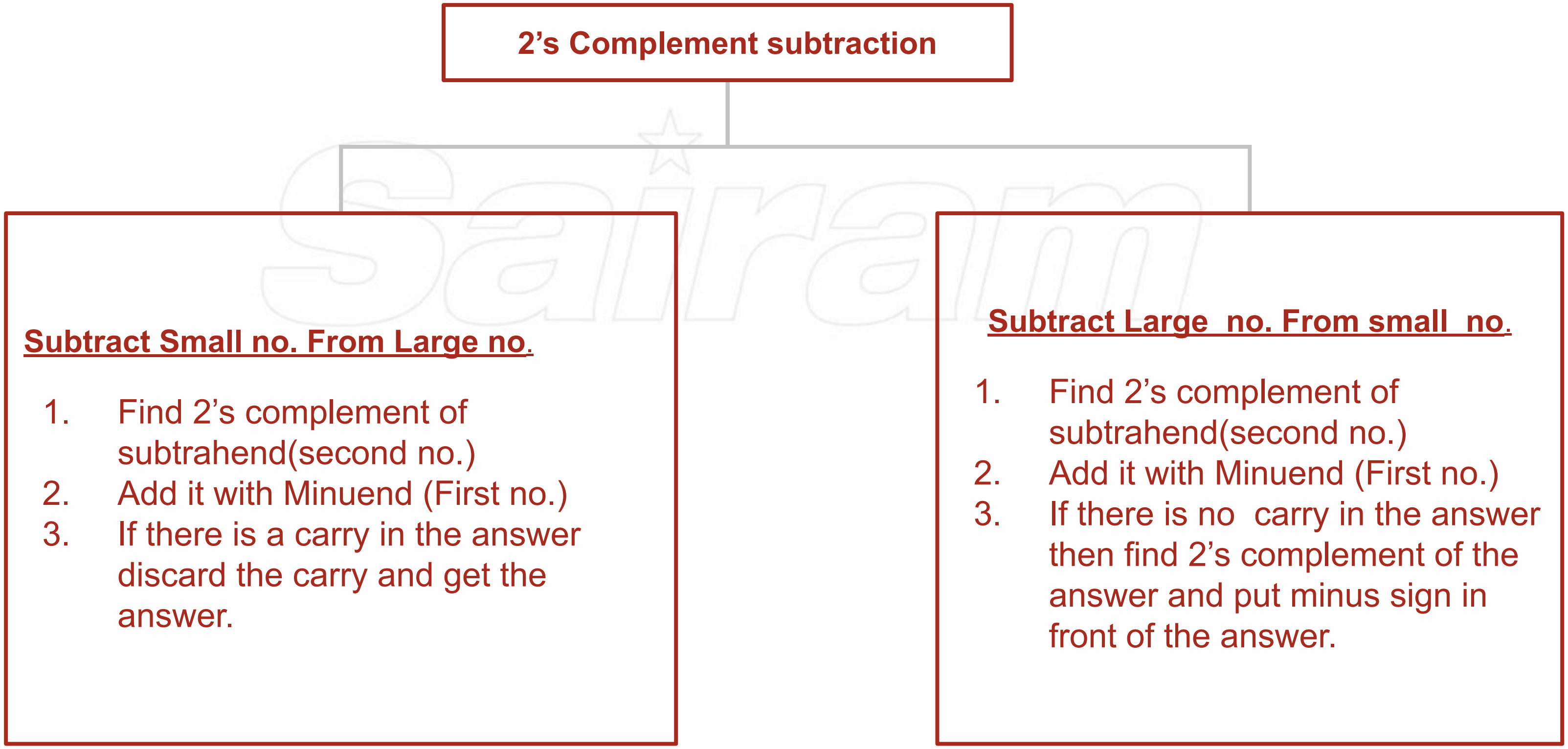
one’s complement

00010

‘-’ of (00010)

Ans: -(00010)

2’s Complement Subtraction



2’s Complement subtraction

Subtract Small no. From Large no.

1. Find 2’s complement of subtrahend(second no.)
2. Add it with Minuend (First no.)
3. If there is a carry in the answer discard the carry and get the answer.

Subtract Large no. From small no.

1. Find 2’s complement of subtrahend(second no.)
2. Add it with Minuend (First no.)
3. If there is no carry in the answer then find 2’s complement of the answer and put minus sign in front of the answer.

Example for 2’s complement Subtraction

Subtract (11011)$_2$ from (11101)$_2$ using 2’s complement. Also subtract (11101)$_2$ from (11011)$_2$

```
      1  1  1  0  1
      0  0  1  0  1  (+)  ------> two’s complement of 11011
     -----------------
   1  0  0  0  1  0  ----> Ans
carry
        Discard the
        carry
```

```
      1  1  0  1  1
      0  0  0  1  1    (+)
     ---------------
   1  1  1  1  0-------> no carry

        two’s complement

   00001+1=00010

     ‘-’ of (00010)

   Ans: -(00010)
```

Practice sums:

Do the following binary subtractions by 2’s complement method

1. 101101-110011
2. 110011-101101

9's Complement:

For a decimal number if all the digits are subtracted from 9 that is 9's complement of that decimal number

```
For example 9's complement of 23456 is   99999
                                         -23456
                                        ------------
                                          76543  ----> 9's complement of 23456
                                          ------------
```

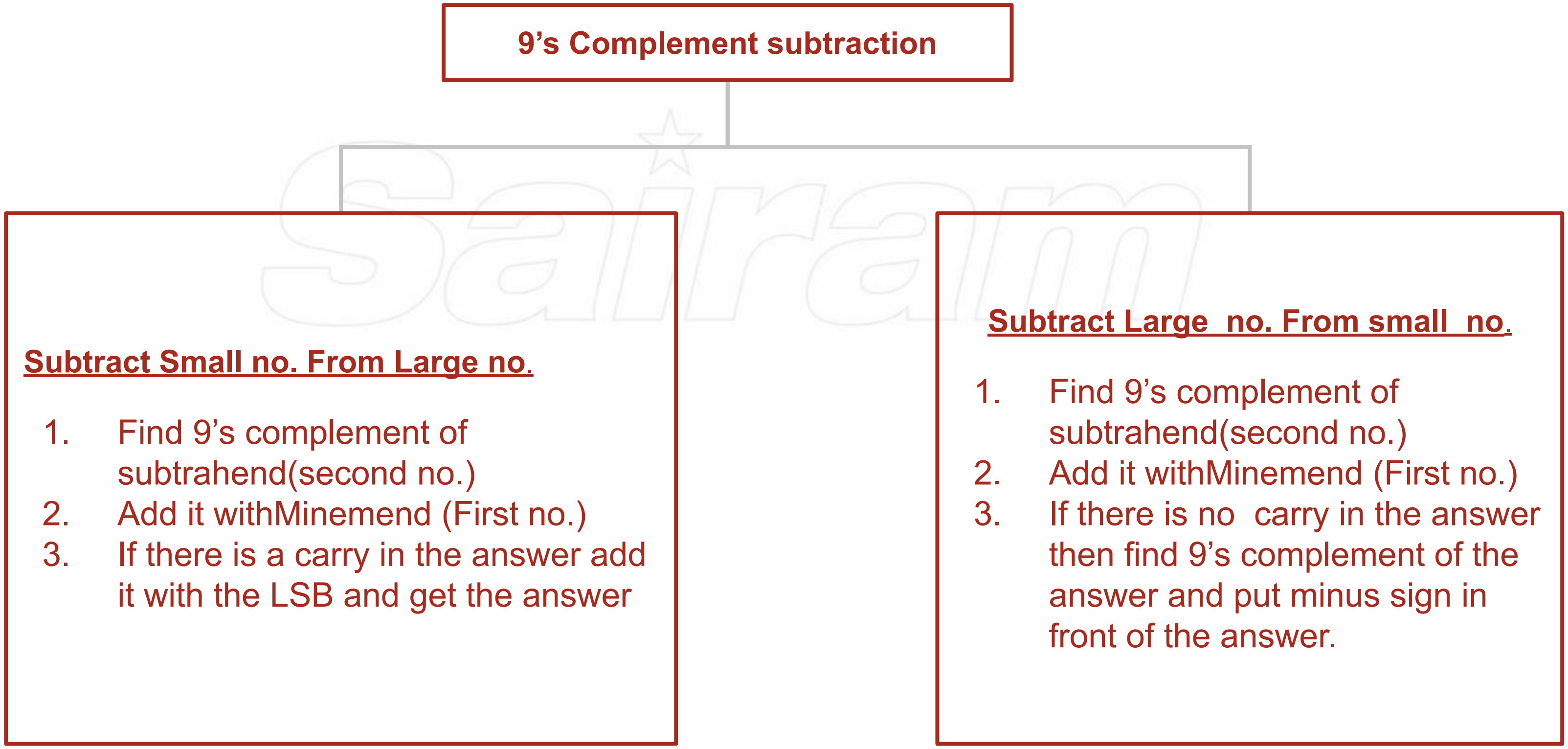
10's Complement:

Find 9's complement and add 1 to LSB

Example: 10's complement of 23456 is :

10's complement of 23456 is 76543 + 1 = 766544

9’s Complement Subtraction



9’s Complement subtraction

Subtract Small no. From Large no.

1. Find 9’s complement of subtrahend(second no.)
2. Add it withMinemend (First no.)
3. If there is a carry in the answer add it with the LSB and get the answer

Subtract Large no. From small no.

1. Find 9’s complement of subtrahend(second no.)
2. Add it withMinemend (First no.)
3. If there is no carry in the answer then find 9’s complement of the answer and put minus sign in front of the answer.

Example for 9’s complement Subtraction

Subtract $(265)_{10}$ from $(397)_{10}$ using 9’s complement. Also subtract $(397)_{10}$ from $(265)_{10}$

3 9 7 **2 6 5**

7 3 4 --->(999-265-->9’s compl of 265) **6 0 2**

----

1 1 3 1 9’s compl of 867 **8 6 7 → no carry**

1

----------- **1 3 2**

1 3 2 ---> Ans

-------- **-(132)**

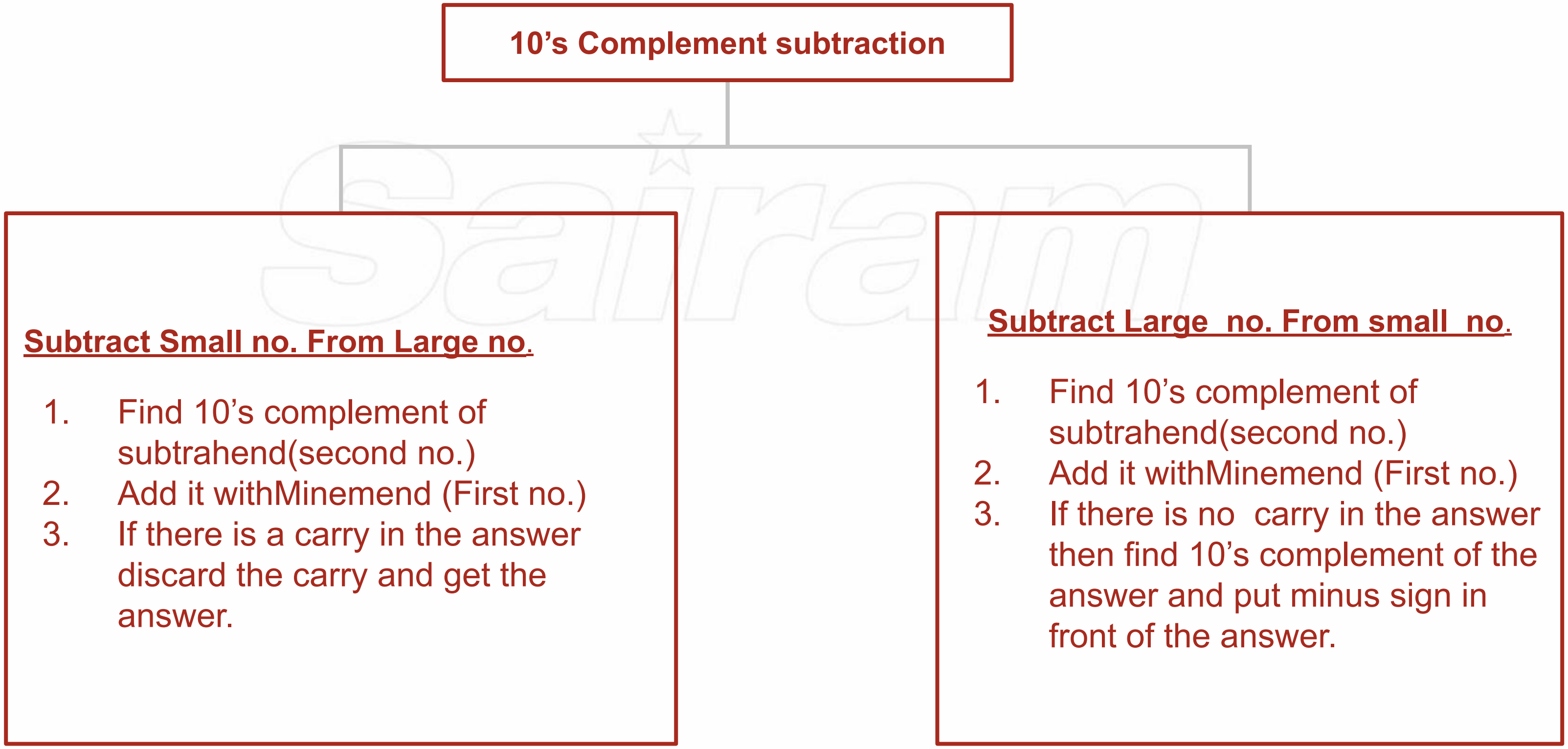
Ans

Practice sums:

Do the following decimal subtractions by 9’s complement method

1. 45689-387
2. 387-45689

10’s Complement Subtraction



10’s Complement subtraction

Subtract Small no. From Large no.

1. Find 10’s complement of subtrahend(second no.)
2. Add it withMinemend (First no.)
3. If there is a carry in the answer discard the carry and get the answer.

Subtract Large no. From small no.

1. Find 10’s complement of subtrahend(second no.)
2. Add it withMinemend (First no.)
3. If there is no carry in the answer then find 10’s complement of the answer and put minus sign in front of the answer.

Example for 10’s complement Subtraction

Subtract $(265)_{10}$ from $(397)_{10}$ using 10’s complement. Also subtract $(397)_{10}$ from $(265)_{10}$

3 9 7 **2 6 5**

7 3 5 --->(999-265-->10’s compl of 265) **6 0 3**

----

1 1 3 2 Discard the carry 10’s compl of 868 **8 6 8 → no carry**

----------- **1 3 1 +1**

1 3 2 ---> Ans

--------- **-(132) Ans**

Practice sums:

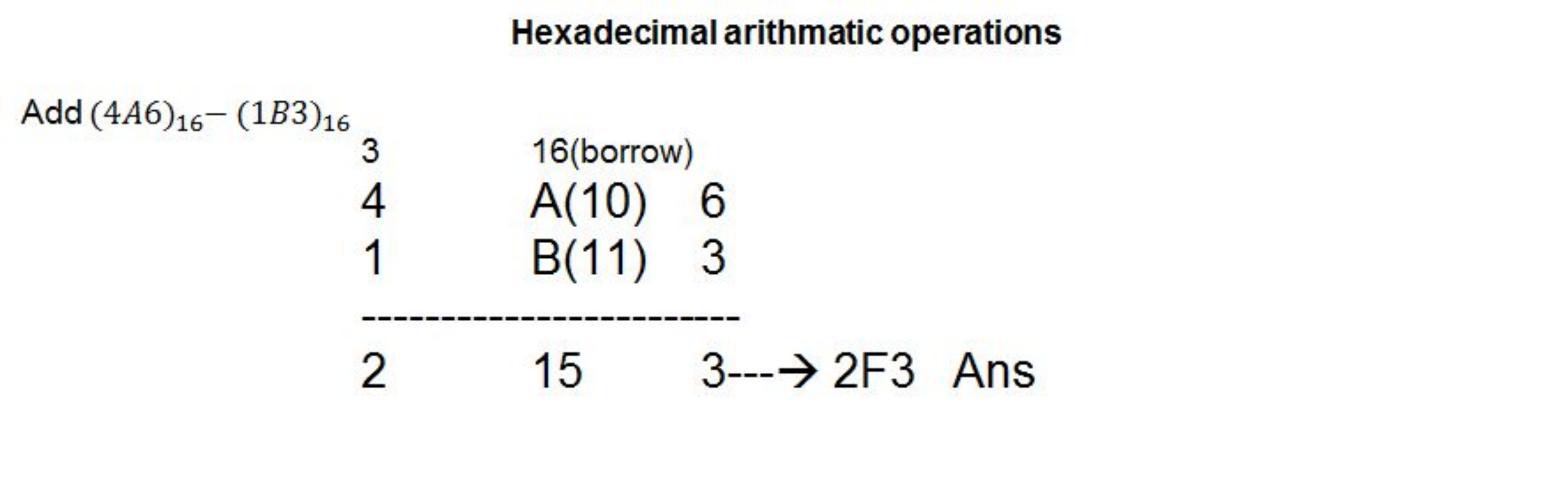
Do the following decimal subtractions by 10’s complement method

1. 45689-387
2. 387-45689

Hexadecimal arithmatic operations

Add $(4A6)_{16}$+ $(1B3)_{16}$

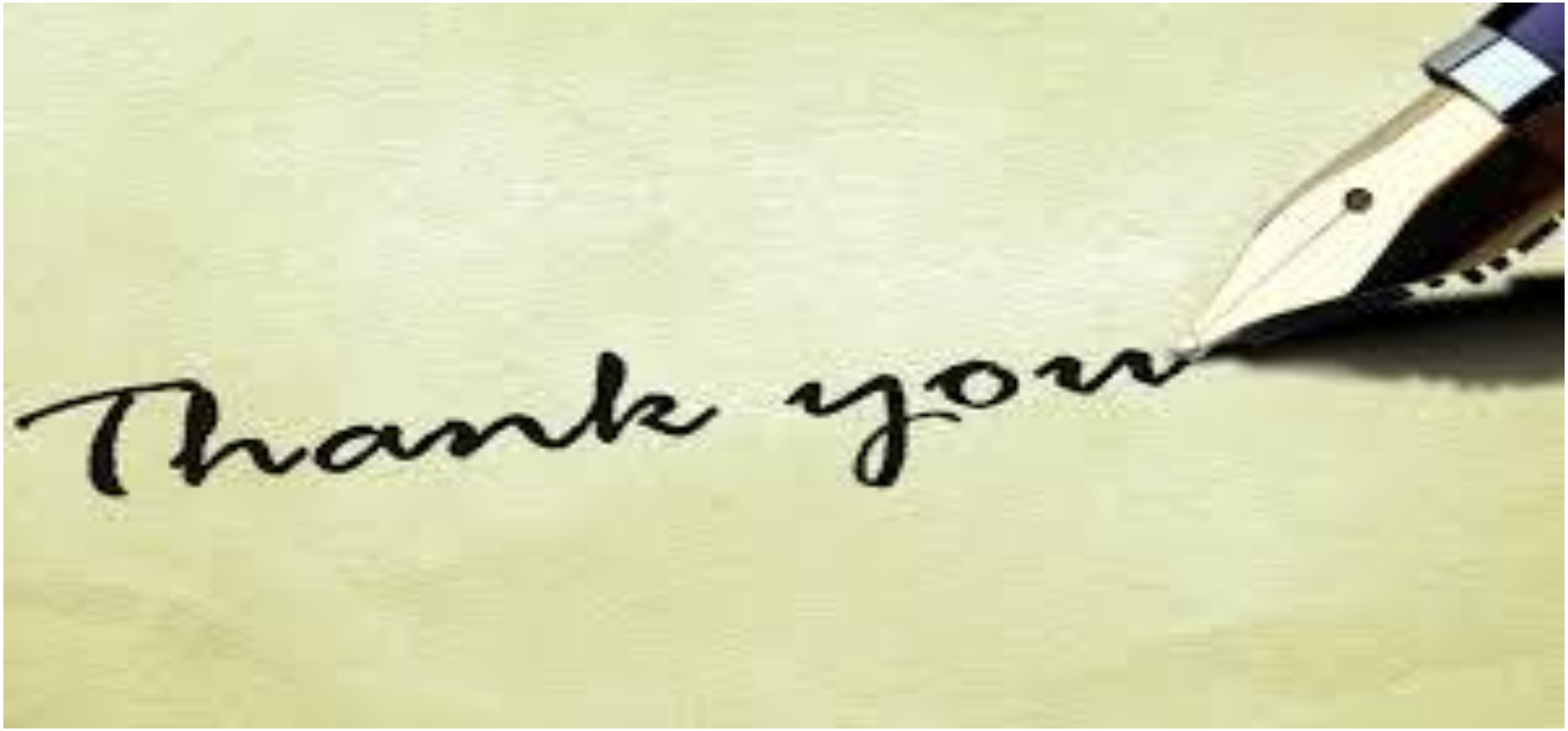
```
4        A(10)   6
1        B(11)   3
----------------------
5        21      9
+1       -16     9
---------------------
6        5       9------->ans
---------------------
```



Hexadecimal arithmatic operations

Add $(4A6)_{16} - (1B3)_{16}$

```
3          16(borrow)
4          A(10)   6
1          B(11)   3
------------------------
2          15      3---→ 2F3   Ans
```


Thank you

Sri SAI RAM ENGINEERING COLLEGE
INSTITUTE OF TECHNOLOGY
West Tambaram, Chennai - 44

SAIRAM DIGITAL RESOURCES

YEAR IV SEM VII

EI8751

INDUSTRIAL DATA NETWORKS
(COMMON TO EIE &ICE)

UNIT NO 1

DATA NETWORK FUNDAMENTALS

1.1 NETWORK HIERARCHY

ELECTRONICS & INSTRUMENTATION ENGINEERING

NETWORK

- A Network is a set of devices (node) connected by a media links.
- A node can be a computer, printer, or any other device capable of sending and receiving data generated by other nodes on the network.
- The links connecting the devices are often called communication channels.

Computer network

- An interconnected collection of autonomous computers.
- Two computers are said to be interconnected if they are able to exchange information.
- The connection may be a copper wire, fiber optics, microwaves, and communication satellites can also be used.
- A network is a large computer with remote printers and terminals.

A Distributed system

★ It is a software system build on top of a network.

★ In a distributed system, the existence of multiple autonomous computers is transparent to the user.

★ The user can type a command to run a program and it runs.

★ It gives a high degree of cohesiveness and transparency.

TOPIC -1 NETWORK HIERARCHY

To be effective and efficient, a network must meet the following criteria such as Performance, reliability and security.

Performance

It can be measured by transit time and response time. The performance of a network depends on a number of factors, including the number of users, the type of transmission medium, the capabilities of the connected hardware, and the efficiency of the software.

Reliability

Network reliability is measured by frequency of failure, the time it takes a link to recovery from a failure, and the network’s robustness in a catastrophe.

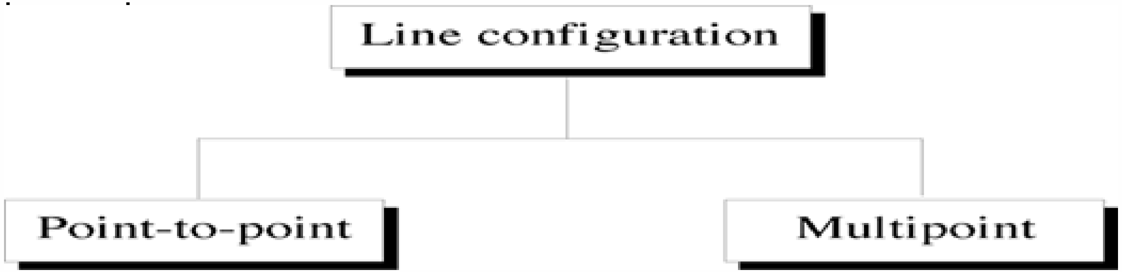
Security

Network security issues include protecting data from unauthorized access and viruses.

NETWORK ATTRIBUTES (Line configuration)

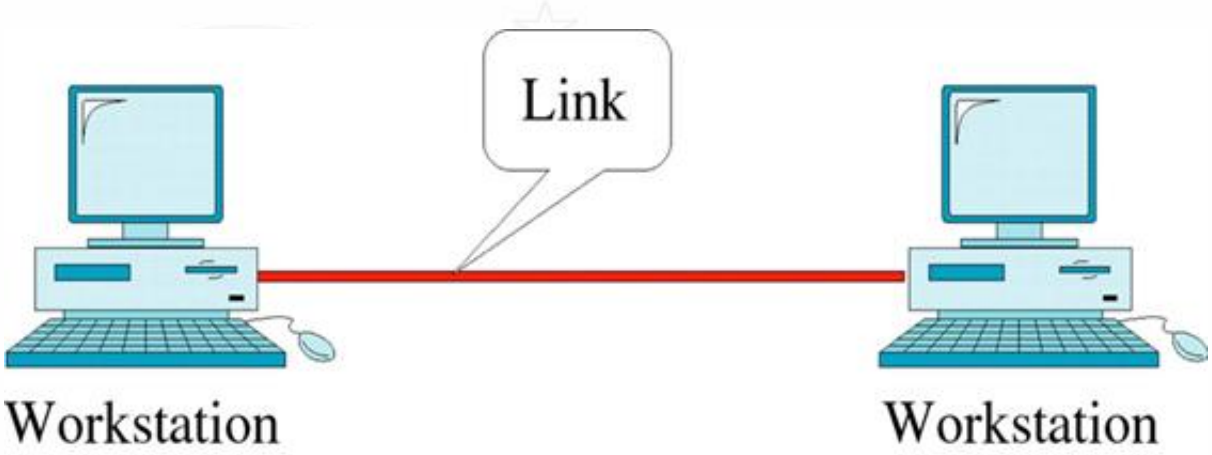
It is broadly classified into three types,
(i) Broadcast networks (ii)Point-to-point networks (iii)Multipoint networks.

- **Broadcast network** have a single communication channel that is shared by all the machines on the network.
- Short messages, called packets is sent by any machines are received by all the others.
- An address field within the packet specifies for whom it is intended.
- Upon receiving a packet, a machine checks the address field.
- If the packet is intended for itself, it processes the packet; if the packet is intended for some other machine, it is just



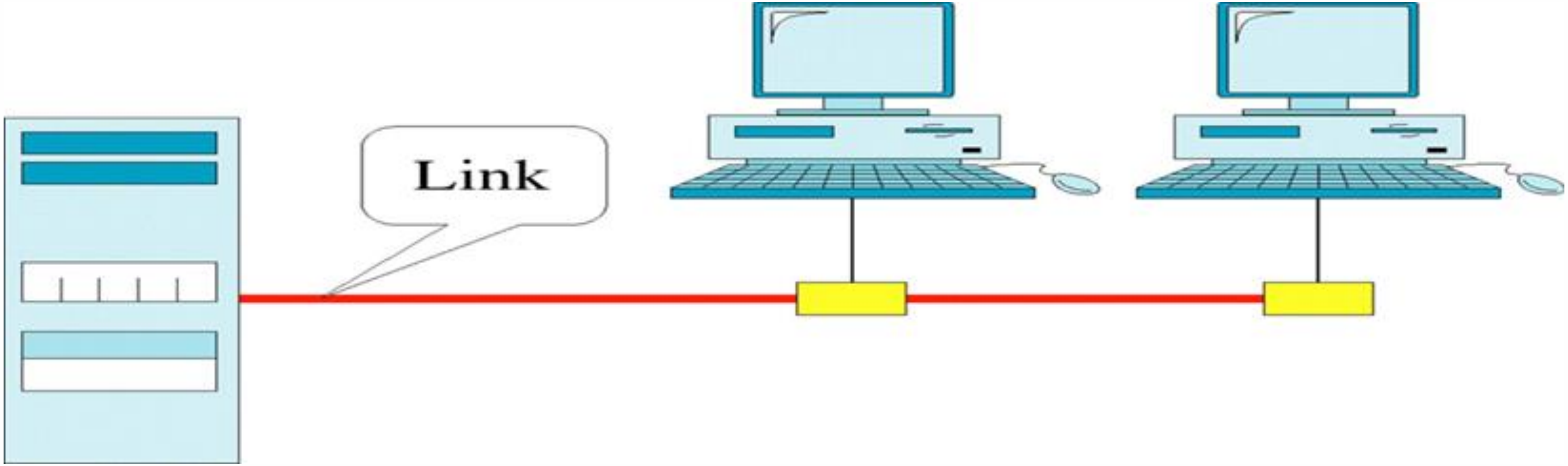
Point -to-point line configuration

It provides a dedicated link between two devices. The entire capacity of the channel is reserved for transmission between those two devices.



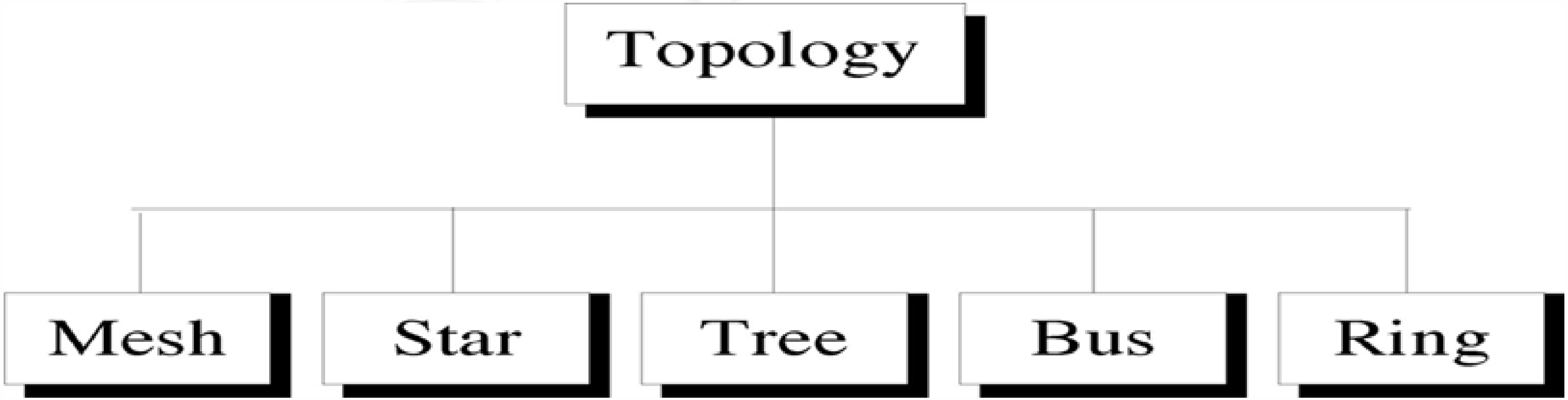
Multipoint line configuration

It is one in which more than two specific devices share a single link. In a multipoint environment, the capacity is shared, either spatially or temporally.



TOPOLOGY OF DATA NETWORKS

The topology of a network is the geometric representation of the relationship of all the links and linking devices to each other.



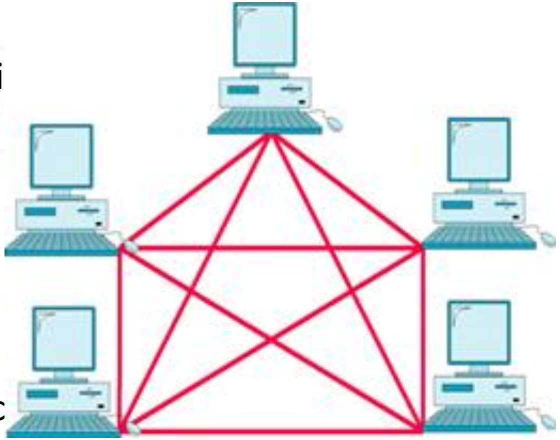
TOPOLOGY OF DATA NETWORKS

Mesh Topology

Every device has dedicated link carries traffic only between the two devices it connects. A fully connected mesh network has (n(n - l)/2)physical channels to link 'n' devices. To accommodate many links, every device on the network must have (n-1) input/output (I/O) ports.

Advantages

- ❖ Each connection can carry its own data load, thus eliminates the traffi
- ❖ A mesh topology is robust.
- ❖ it is privacy or security
- ❖ It makes fault identification and fault isolation easy.



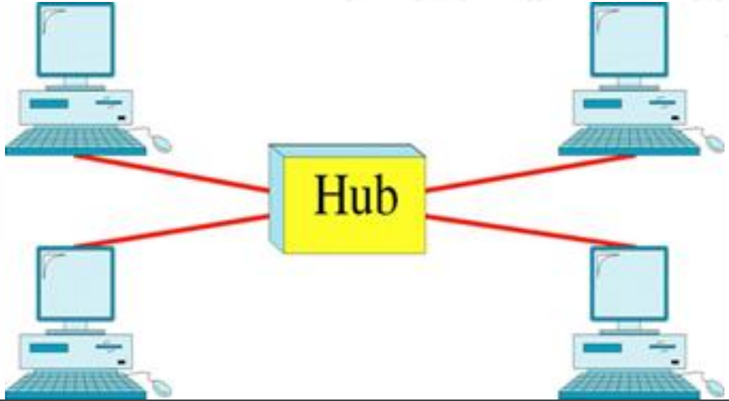
Disadvantages

- ❖ Installation and reconfiguration are difficult.
- ❖ The sheer bulk of the wiring can be greater than the available space c
- ❖ The hardware required to connect each link can be expensive

TOPOLOGY OF DATA NETWORKS

Star Topology

- Each device has a dedicated point-to-point link only to a central controller, usually called a hub.
- The devices are not directly linked to each other.
- The controller acts as an exchanger.
- If one device wants to send data to another, it sends the data to the controller, which then relays the data to the other connected device.



Advantages

- ❖ It is less expensive than a mesh topology.
- ❖ It is easy to install and reconfigure.
- ❖ A star topology is robust.
- ❖ It makes fault identification and fault isolation.

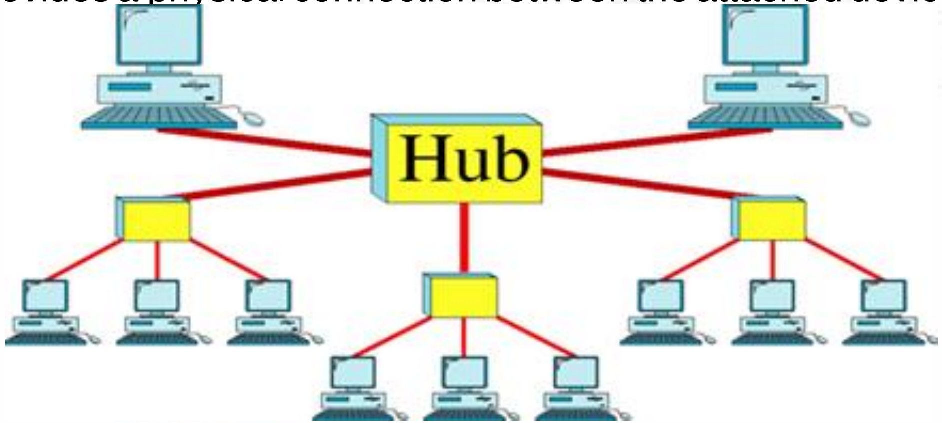
Disadvantage

- ❖ More cabling is required in a star topology.

TOPOLOGY OF DATA NETWORKS

Tree Topology

It is a variation of a star. As in a star, nodes in a tree are linked to a central hub that controls the traffic to the network. Only the majority of devices connect to a secondary hub that in turn is connected to the central hub. The central hub in the tree is an active hub. An active hub contains a repeater, which is a hardware device that regenerates the received bit patterns before sending them out. A passive hub provides a physical connection between the attached devices.



Advantages

- It allows more devices to be attached to a single central hub and can increase the distance a signal can travel between devices.
- It allows communications from different computers.

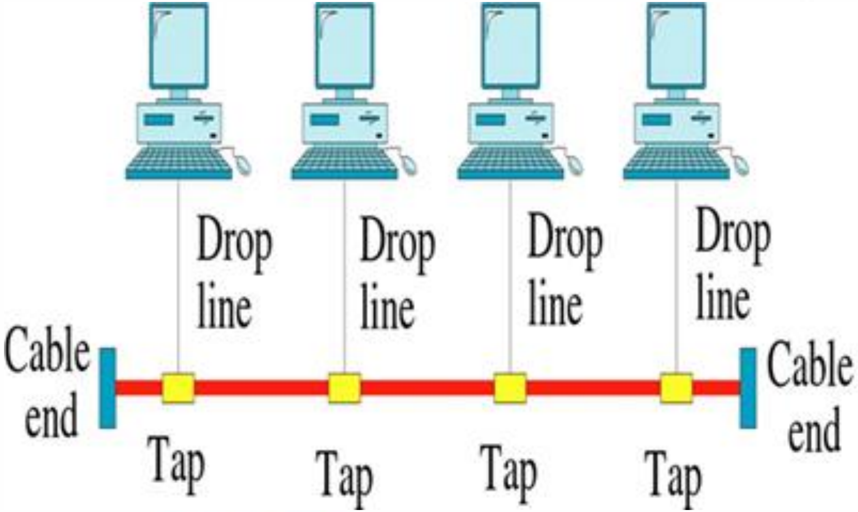
Disadvantages

It requires more cabling

TOPOLOGY OF DATA NETWORKS

Bus Topology

It is a multipoint topology. One long cable acts as a backbone to link all the devices in the network. Nodes are connected to the bus cable by drop lines and taps. A drop line is a connection running between the device and the main cable. A tap is a connector that either splices into the metallic core.



Advantages

- ❖ Easy to install
- ❖ It requires less cabling.
- ❖ In this topology redundancy is eliminated.

Disadvantages

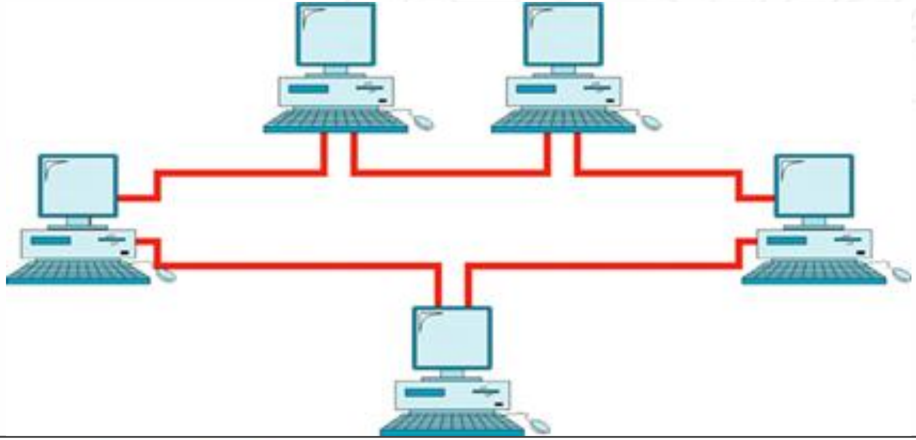
- ❖ It is difficult to reconfiguration and fault isolation. It is difficult to add new devices.
- ❖ A fault or break in the bus cable can stop all transmission.

  The damaged area reflects signals back in the direction of origin, creating noise in both directions.

TOPOLOGY OF DATA NETWORKS

Ring Topology

In a ring topology, each device has a dedicated point-to-point line configuration only with the two devices on either side of it. A signal is passed along the ring in one direction, from device to device, until it reaches its destination. Each device in the ring incorporates a repeater. When a device receives a signal intended for another device, its repeater regenerates the bits and passes them along the path.



Advantages

It is easy to install and reconfigure, Fault isolation is simplified.

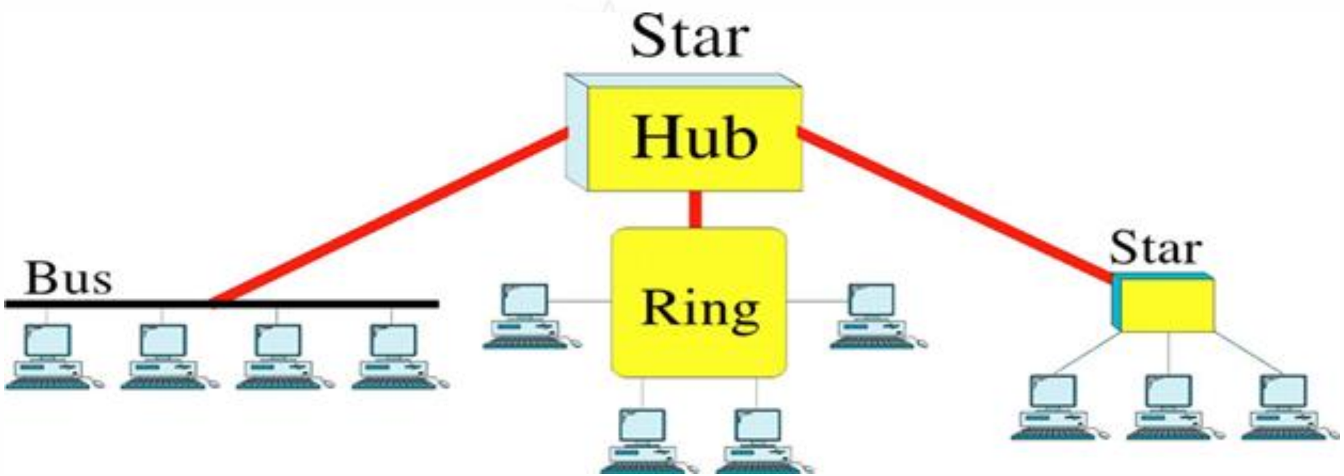
Disadvantages

It is a unidirectional traffic. A break in the ring can disable the entire network.

TOPOLOGY OF DATA NETWORKS

Hybrid topology

A network combines several topologies as sub networks linked together in a larger topology. One department of a business may have decided to use a bus topology while another department has a ring. The two can be connected to each other via a central controller in a star topology.

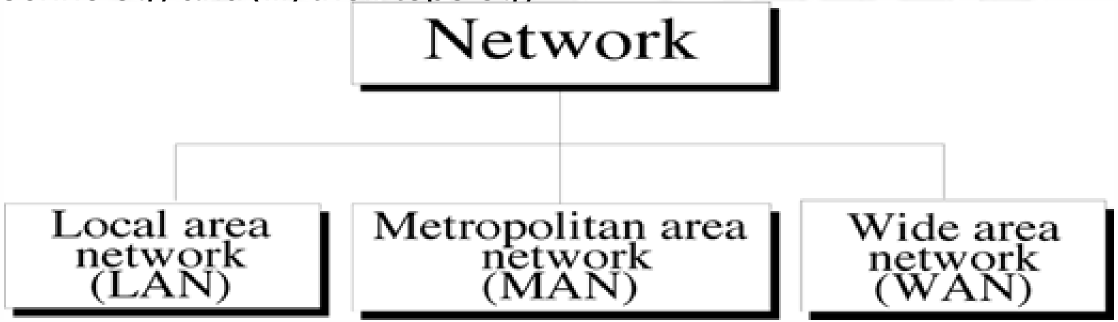


Network topology

https://youtu.be/zbqrNg4C98U

CATEGORIES OF NETWORK

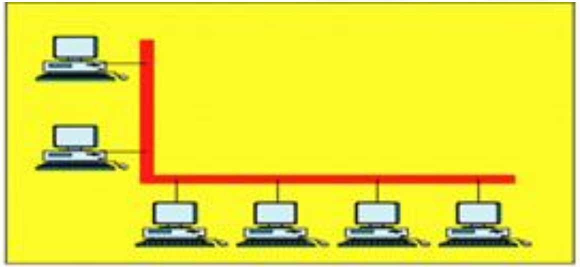
- It is usually privately owned network. Within a single building or campus of up to a few kilometers in size.
- They are widely used to connect personal computers and workstations in company offices and factories to share resources and exchange information.
- LAN's are distinguished from other kinds of network by three characteristics, (i) their size, (ii) their transmission technology and (iii) their topology.



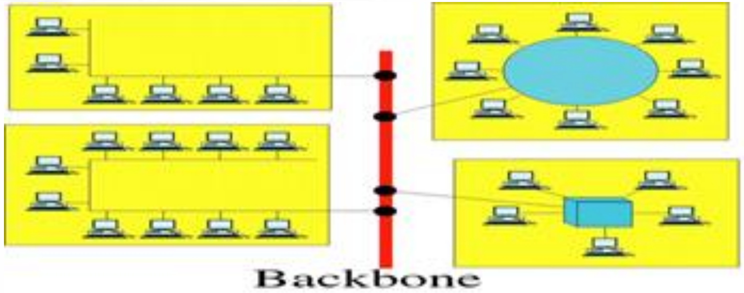
NETWORK –LAN

LAN’s are distinguished from other kinds of network by three characteristics, (i) their size, (ii) their transmission technology and (iii) their topology.

A common example is engineering workstations or accounting PCs. LAN’s often use a transmission Technology consisting of a single cable to which all the machines are attached, like the telephone company Party lines once used in rural areas. Traditionally, LAN’s have data rates 4 to 16 Mbps range. Today, However, speeds are increasing and can react 100 Mbps with gigabit systems in development. Various Topologies are possible for broadcast LAN’s. Most commonly used are bus, ring, and star.



Single building LAN

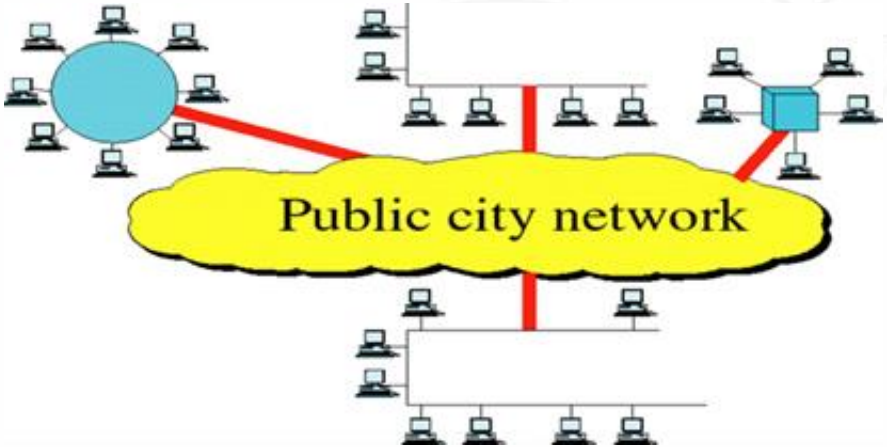


Multiple building LAN

NETWORK –MAN

Metropolitan Area Network (MAN)

It is basically a bigger version of a LAN and normally uses similar technology. It might cover a group of nearby corporate offices or city and might be either private or public.



- A MAN can support both data and voice, and might even be related to the local cable television network.
- A MAN just has one or two cables and many telephone companies provide a popular MAN service called switched multi- megabit data services (SMDS).

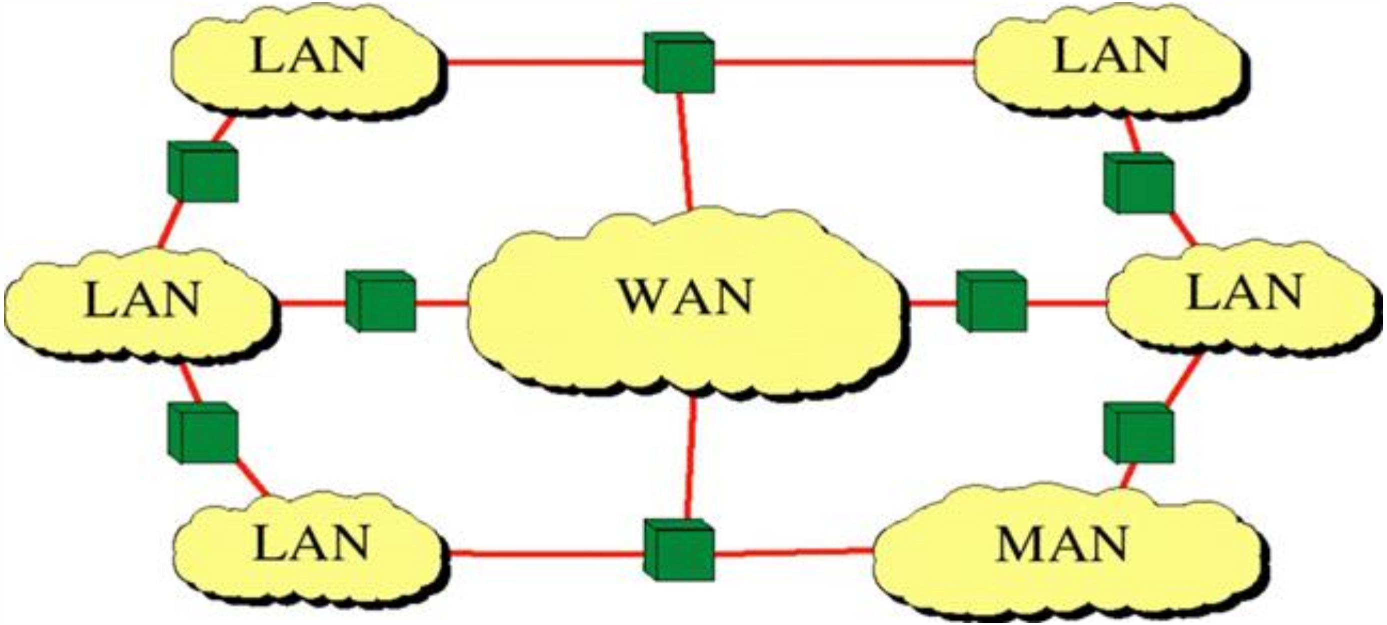
NETWORK –WAN

Wide Area Network (WAN)

It provides a long-distance transmission of data, voice, image and video information over large geographical areas that may comprise a country, or a continent or even the whole world.

It contains a collection of machines intended for running user programs.

Internetwork (Internet)



TYPES OF NETWORKS

https://youtu.be/4_zSIXb7tLQ

Service and protocols

A Service is a set of primitives (operation) that a layer provides to the layer above it. The service defines what operations the layer is prepared to perform on behalf of its users.

A protocol is a set of rules governing the format and meaning of the frames, packets or messages that are exchanged by the peer entities within a layer.

Entities use protocols in order to implement their service definitions.

They are free to change their protocols at will, provided they do not change the service visible to their users.

Key elements of the protocol

Syntax refers to the structure or format of the data, meaning the order in which they are presented.

Semantics refers to the meaning of each section of bits.

Timing refers to two characteristics: when data should be sent and how fast they can be sent

Standards

A Standard provides a model for development that makes it possible for a product to work regardless of the individual manufacturer. Standards are essential in creating and maintaining an open and competitive market for equipment manufacturers.

They provide guidelines to manufacturer, vendors, government agencies and other service providers to ensure the kind of interconnectivity necessary in today’s marketplace and in international communications.

Data communication standards fall into two categories, **de facto (by fact) and de jure (by law).**

- **Dejure** are those that have been legislated by an officially recognized body.
- **De facto** are often established originally by manufacturers to define the functionality of a new product or technology.

Standards Organisations

Standards are developed through the cooperation of standards creation committees, forums, and government regulatory agencies.

- International organization for standardization (ISO)
- American national standards institute (ANSI)
- Institute of electrical and electronics engineers (IEEE)
- Electronic Industries Association (EIA)

THANK YOU

slido

1.1.Choose the correct about the network

ⓘ Start presenting to display the poll results on this slide.

2.2.True or False: Line configuration refers to the way two or more communication devices attach to a link

ⓘ Start presenting to display the poll results on this slide.

3.Multipoint line configuration means

ⓘ Start presenting to display the poll results on this slide.

4.True or False: Two or more devices connected to a link and two or more links form a topology

ⓘ Start presenting to display the poll results on this slide.

slido

5.Which is true about Star topology?

ⓘ Start presenting to display the poll results on this slide.

slido

6.Which is true about Bus topology?

ⓘ Start presenting to display the poll results on this slide.

slido

7.Half duplex refers to

ⓘ Start presenting to display the poll results on this slide.

8.A MAN network is

ⓘ Start presenting to display the poll results on this slide.

slido

9.Simplex refers to

ⓘ Start presenting to display the poll results on this slide.

10.A LAN network is

ⓘ Start presenting to display the poll results on this slide.

Sri SAI RAM ENGINEERING COLLEGE
INSTITUTE OF TECHNOLOGY
West Tambaram, Chennai - 44

SAIRAM
DIGITAL RESOURCES

YEAR II

SEM IV

EI 8451

ELECTRICAL MACHINES
Common to EIE & ICE

UNIT NO 1 - DC MACHINES

- 1.3-Principle of operation and construction of motor,Various excitation schemes

INSTRUMENTATION & CONTROL ENGINEERING

Principle of operation and construction of motor,Various excitation schemes

Motor Principle

An electric motor is a machine which converts electrical energy into mechanical energy. Its action is based on the principle that when a current-carrying is placed in a magnetic field, the conductor experiences a mechanical force . The direction of this force is given by Fleming's left-hand rule and magnitude is given by

F=BI*l* (N)

B = flux density due to the flux produced by the field winding

I = Magnitude of current passing through the conductor

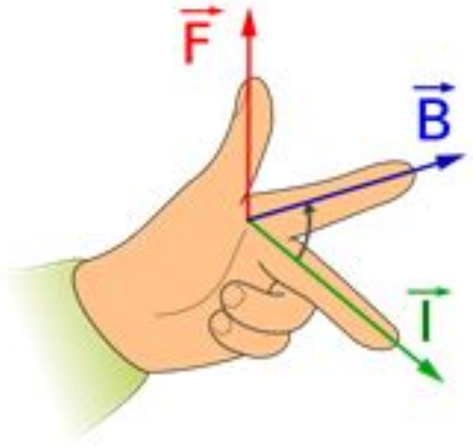
l = Active length of the conductor

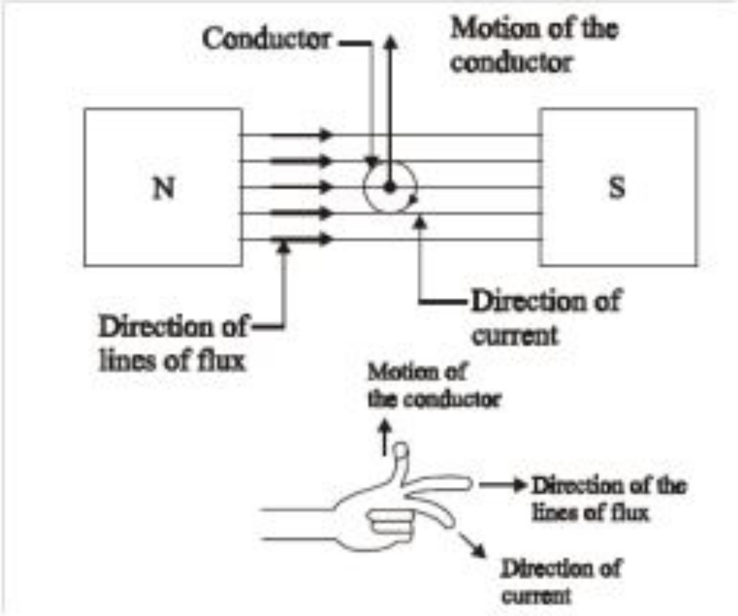
By construction, there is no basic difference between DC generator and DC motor. In fact, the same DC machine can be used interchangeably as a generator or as a motor. In generator the input is mechanical energy by a prime mover and develops a voltage, while in a DC motor the input is electrical energy and it develops a rotating torque.

Principle of operation and construction of motor,Various excitation schemes

Fleming's left hand rule

Keep the forefinger, middle finger and thumb of the left hand mutually perpendicular to one another. If the forefinger indicates the direction of the magnetic field (B) and the middle finger indicates the direction of current (I) in the conductor, then the thumb points to the direction of motion (F) of the conductor.



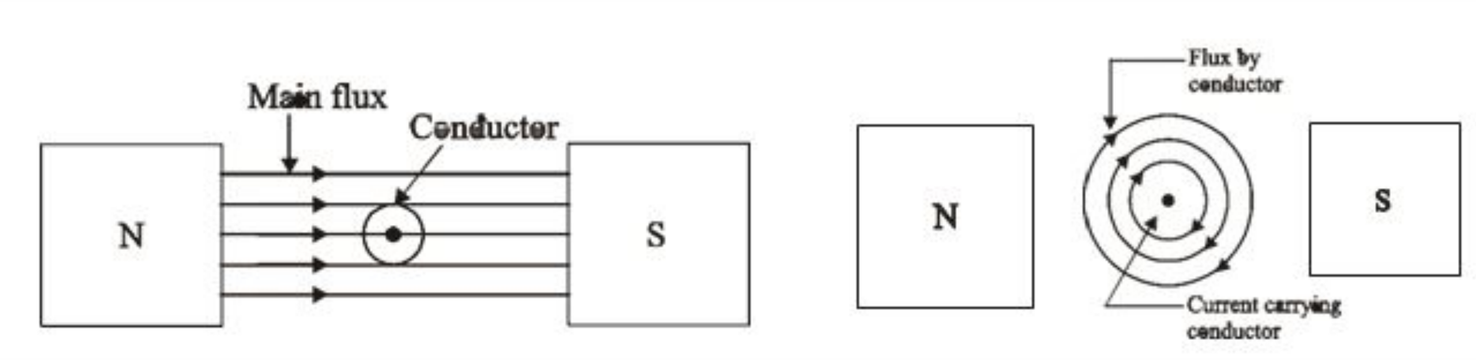


Principle of operation and construction of motor,Various excitation schemes

Principle of operation of DC Motor

A motor is a mechanism by which electrical energy is converted into mechanical energy. Both in principle and design, a DC motor is the reverse process of DC generator.

Consider a single conductor placed in a magnetic field as shown in fig 7.10, the two diagrams indicate the direction of magnetic fluxes considering separately.



Resultant Force developed

Principle of operation and construction of motor,Various excitation schemes

How the resultant rotation takes place in a DC motor is clearly understood from the following descriptions. The fig (7.11a)

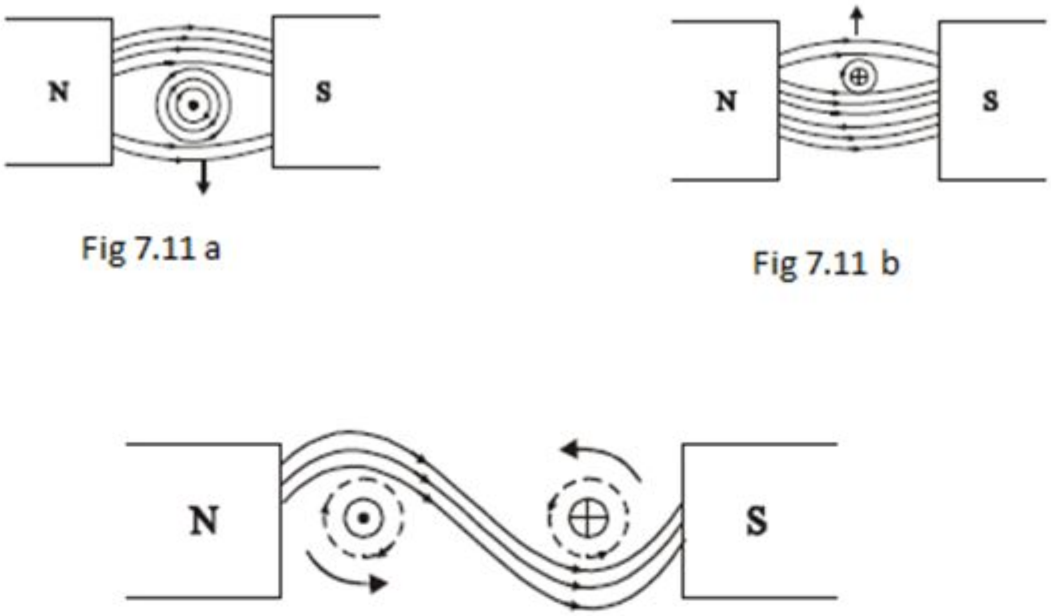


Fig 7.11 a

Fig 7.11 b

Fig 7.11 c

Torque development in DC motors

Principle of operation and construction of motor,Various excitation schemes

A steady current is passed through the armature coil from the commutator and the brushes are so arranged as to reverse the current every half revolution. When a coil, carrying a current is placed in magnetic field, it experiences forces, given by Fleming's left-hand rule, which turn it about in a direction perpendicular to both the field and the current.

The fig 7.11a shows on the upper side of the conductor, the magnetizing force of the field and due to the current are additive while on the lower side these are subtractive. The result is to increase the flux density into the region directly above the conductor and to reduce flux density below the conductor.

If the current is reversed in the conductor, the strengthening of flux lines occurs below the conductor and the conductor will be pushed upwards as shown in fig 7.11b.

As for the reasons above, one of the coil side 'A' will be forced to move downwards while 'B' moves upwards. These two forces are equal in magnitude and opposite to each other. (7.11c). These twisting force makes the armature to rotate through the two ends are supported by bearings.

But, practically a DC machine will have multiple conductors and each conductor will be experiencing a force F=BI l newton. These forces collectively produce a driving torque which sets the armature rotating. The machine is then said to be motoring.

Principle of operation and construction of motor,Various excitation schemes

BACK EMF OF DC MOTOR (E_B)

When the armature of a DC motor rotates in the magnetic field, the armature conductor cuts the magnetic flux. Hence emf will be induced in the conductor according to Faraday’s law of electromagnetic induction. This induced emf acts in opposite direction to the applied voltage and it is referred as the back emf or counter emf E_b.

The back emf is given by,

E_b=which is same as that emf induced in a DC generator.

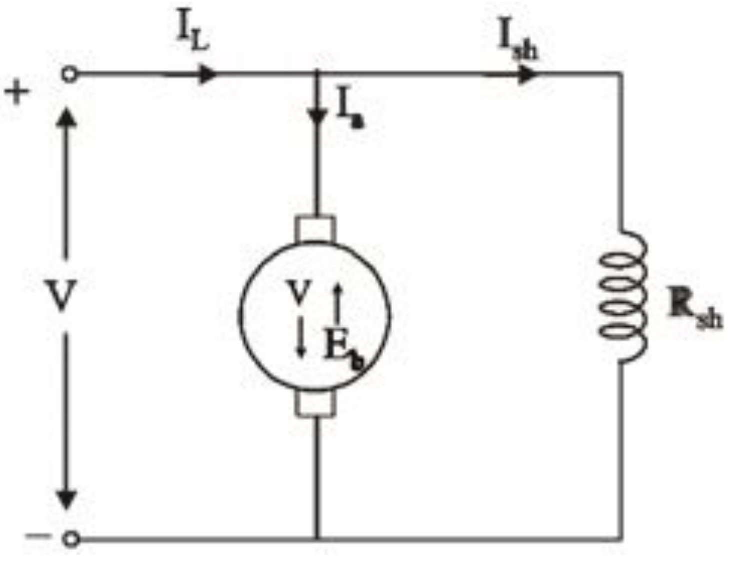
The relationship between the current, back emf and the applied voltage for a dc shunt motor is given by $E_b=V-I_aR_a$.

Where,

E_b=back emf in volts., V=supply voltage in volts

I_a=armature current in amperes. R_a=armature resistance in ohms.

The back emf E_b is always less than that of the applied voltage and hence the current is flowing against the direction of back emf.



DC motor circuit

SIGNIFICANCE OF BACK EMF

The presence of back e.m.f. makes the d.c. motor a self-regulating machine i.e., it makes the motor to draw as much armature current as is just sufficient to develop the torque required by the load.

$$\text{Armature current, } I_a = \frac{V - E_b}{R_a}$$

(i) When the motor is running on no load, small torque is required to overcome the friction and windage losses. Therefore, the armature current Ia is small and the back e.m.f. is nearly equal to the applied voltage.

(ii) If the motor is suddenly loaded, the first effect is to cause the armature to slow down. Therefore, the speed at which the armature conductors move through the field is reduced and hence the back e.m.f. Eb falls. The decreased back e.m.f. allows a larger current to flow through the armature and larger current means increased driving torque. Thus, the driving torque increases as the motor slows down. The motor will stop slowing down when the armature current is just sufficient to produce the increased torque required by the load.

SIGNIFICANCE OF BACK EMF

(iii) If the load on the motor is decreased, the driving torque is momentarily in excess of the requirement so that armature is accelerated. As the armature speed increases, the back e.m.f. Eb also increases and causes the armature current Ia to decrease. The motor will stop accelerating when the armature current is just sufficient to produce the reduced torque required by the load.

It follows, therefore, that back e.m.f. in a d.c. motor regulates the flow of armature current i.e., it automatically changes the armature current to meet the load requirement.

Principle of operation and construction of motor,Various excitation schemes

4.5 Voltage Equation of D.C. Motor

Let in a d.c. motor (See Fig. 4.3),

V = applied voltage

E_b = back e.m.f.

R_a = armature resistance

I_a = armature current

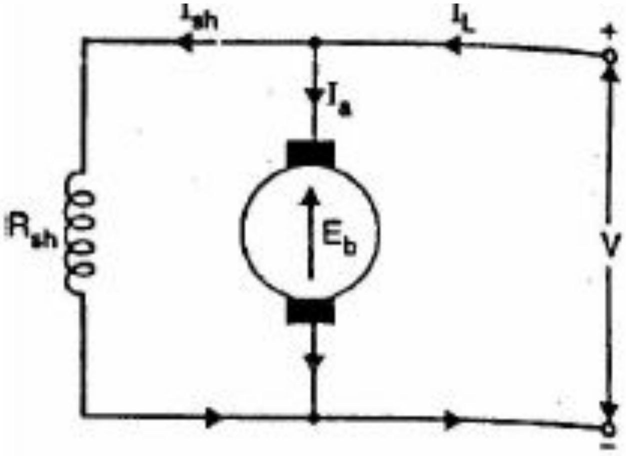


Fig. (4.3)

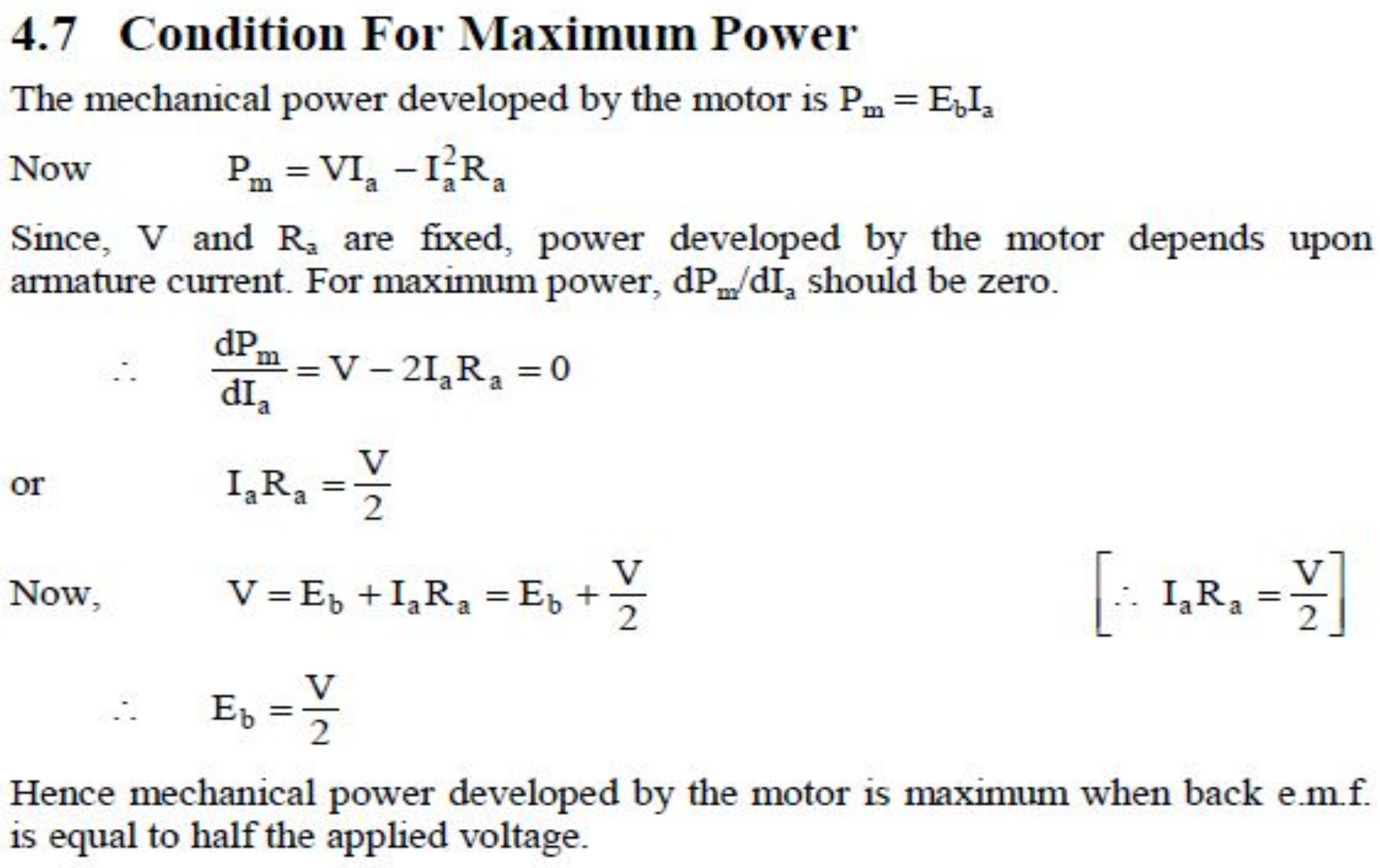
Since back e.m.f. E_b acts in opposition to the applied voltage V, the net voltage across the armature circuit is V− E_b. The armature current I_a is given by;

$$I_a = \frac{V - E_b}{R_a}$$

or $$V = E_b + I_a R_a \qquad (i)$$

This is known as voltage equation of the d.c. motor.

Condition for Maximum Power



4.7 Condition For Maximum Power

The mechanical power developed by the motor is $P_m = E_b I_a$

Now $$P_m = VI_a - I_a^2 R_a$$

Since, V and R_a are fixed, power developed by the motor depends upon armature current. For maximum power, dP_m/dI_a should be zero.

$$\therefore \quad \frac{dP_m}{dI_a} = V - 2I_a R_a = 0$$

or $$I_a R_a = \frac{V}{2}$$

Now, $$V = E_b + I_a R_a = E_b + \frac{V}{2} \qquad \left[\because \; I_a R_a = \frac{V}{2}\right]$$

$$\therefore \quad E_b = \frac{V}{2}$$

Hence mechanical power developed by the motor is maximum when back e.m.f. is equal to half the applied voltage.

Principle of operation and construction of motor,Various excitation schemes

Various excitation schemes

The DC motors are classified into three types based on the field winding connection with the armature.

i) DC shunt motor

ii) DC series motor

iii) DC compound motor

DC compound motors is classified as,

a) Long shunt compound motor

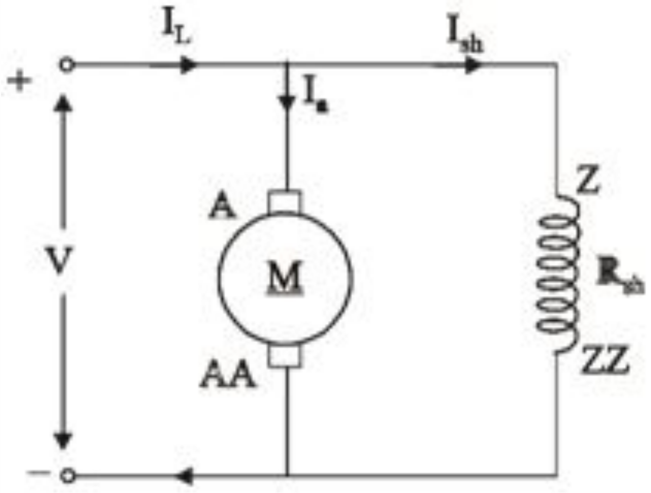
b) Short shunt compound motor.

Principle of operation and construction of motor,Various excitation schemes

Various excitation schemes

DC shunt motor

In DC shunt motor, the field winding is connected in parallel (shunt) with the armature as shown in figure The field winding has a large number of turns and relatively smaller cross sectional area. Therefore, the shunt field winding has high resistance and hence shunt field current is relatively small compared with armature current. The speed of DC shunt motor is almost constant during no-load to full-load.



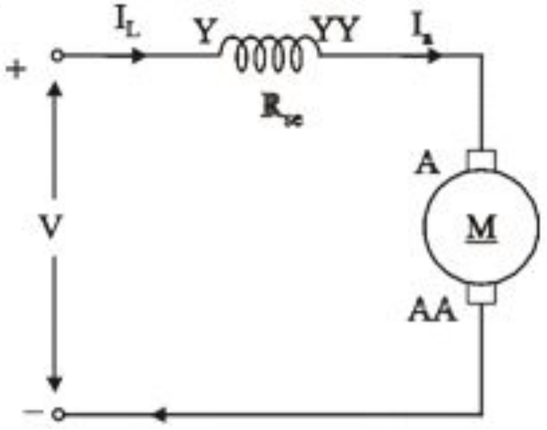
Equivalent circuit of DC shunt motor

Principle of operation and construction of motor,Various excitation schemes

Various excitation schemes

DC series motor

In DC series motor, the field winding is connected in series with the armature as shown in fig 7.14 The series field winding carries the input current. $I_L=I_a=I_{se}$. The series field winding has a relatively small in number   to give minimum value of resistance drop



Equivalent circuit of DC series motor

Principle of operation and construction of motor,Various excitation schemes

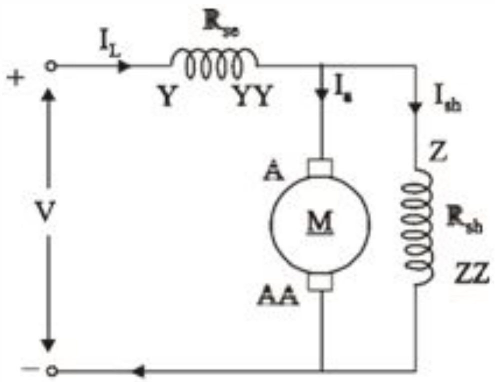
Various excitation schemes

DC compound motor

In compound motors both series and shunt windings are connected according to their name of long shunt and short shunt compound motor.

a) Short shunt compound motor

In short shunt compound motor, the series field winding is connected in series with the parallel combination of armature and shunt field winding. The connection diagram for this compound motor is described below.



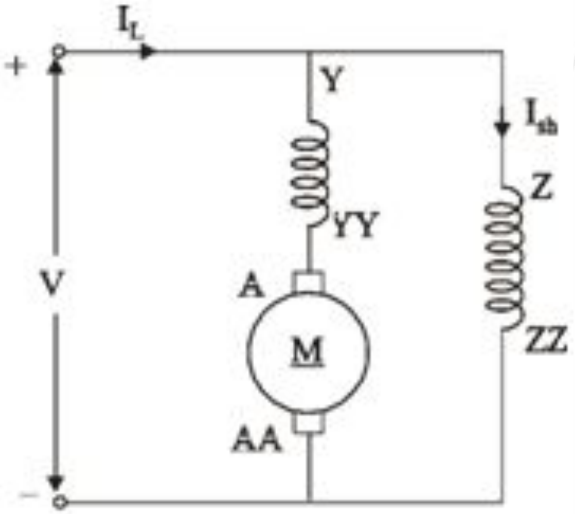
Equivalent circuit of DC short shunt compound motor

Principle of operation and construction of motor,Various excitation schemes

Various excitation schemes

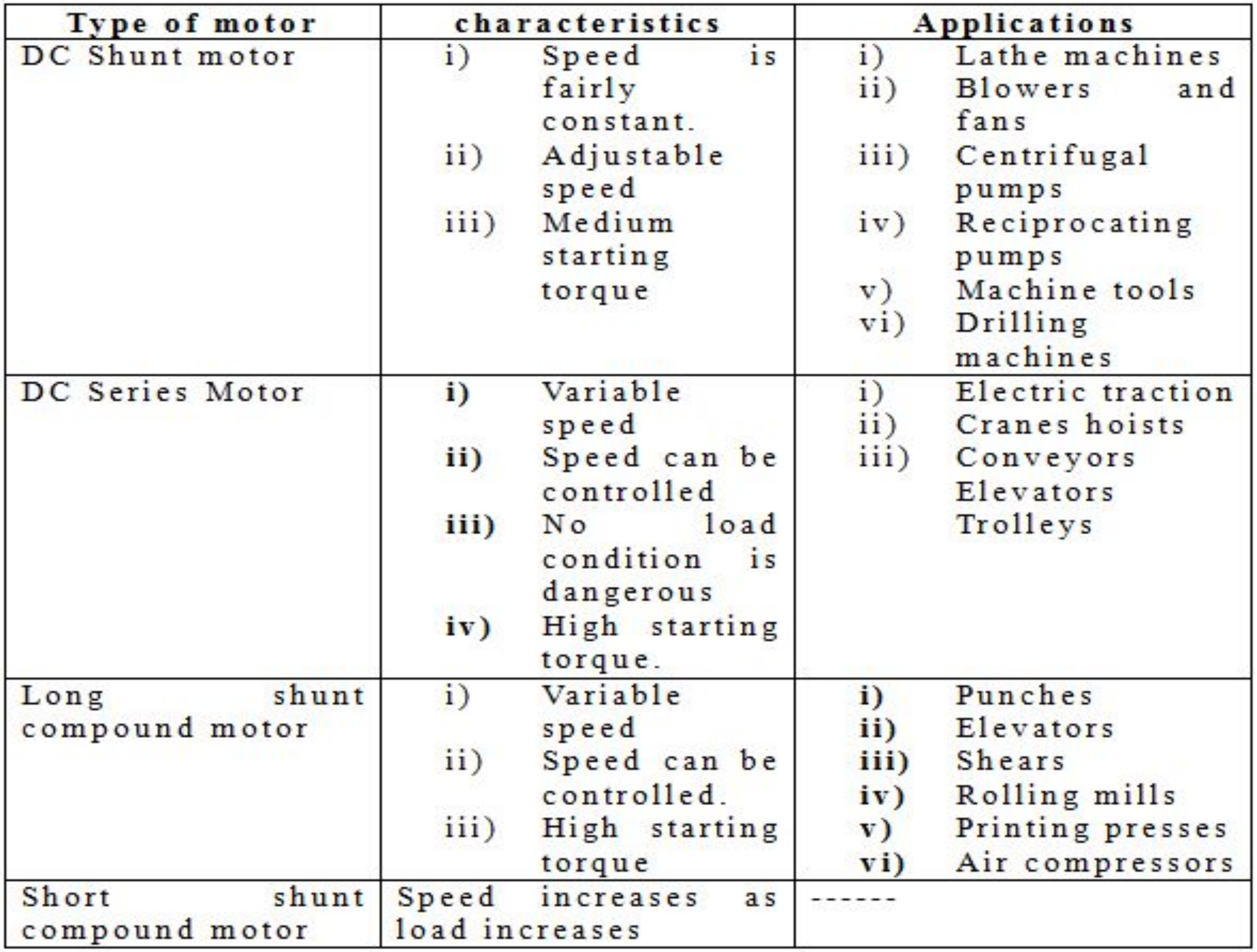
Long shunt compound motor

In long shunt compound motor, the shunt field winding is connected in parallel with the series combination of series field winding and armature winding.



Equivalent circuit of DC long shunt compound motor.

Appilcations of DC Motors



Type of motor	characteristics	Applications
DC Shunt motor	i) Speed is fairly constant. ii) Adjustable speed iii) Medium starting torque	i) Lathe machines ii) Blowers and fans iii) Centrifugal pumps iv) Reciprocating pumps v) Machine tools vi) Drilling machines
DC Series Motor	**i)** Variable speed **ii)** Speed can be controlled **iii)** No load condition is dangerous **iv)** High starting torque.	i) Electric traction ii) Cranes hoists iii) Conveyors Elevators Trolleys
Long shunt compound motor	i) Variable speed ii) Speed can be controlled. iii) High starting torque	**i)** Punches **ii)** Elevators **iii)** Shears **iv)** Rolling mills **v)** Printing presses **vi)** Air compressors
Short shunt compound motor	Speed increases as load increases	------

THANK YOU

Sri SAI RAM ENGINEERING COLLEGE

INSTITUTE OF TECHNOLOGY

West Tambaram, Chennai - 44

SAIRAM DIGITAL RESOURCES

YEAR IV

SEM VII

OCS752

INTRODUCTION TO C PROGRAMMING
(Common to CSE & IT)

UNIT NO V

STRUCTURES IN C

5.1.Structures – Declaration, Initialization, Accessing the members

INFORMATION TECHNOLOGY

UNIT V
STRUCTURES IN C

Introduction to structures

Introduction to structures

Structure is commonly referred to as user-defined data type. Structure is similar to an array but the only difference is that array is collection of similar data type on the other hand structure is collection of different data type. A structure can contain any data type including array and another structure as well. Each variable declared inside structure is called member of structure.

Structure declaration

Declaration of structure must start with the keyword struct followed by the structure name and structure's member variables are declared within braces.

Syntax for declaring structure

```
struct struct-name
{
      datatype var1;
      datatype var2;
      - - - - - - - - - - -
      - - - - - - - - - - -
      datatype varN;
};
```

Example for declaring structure

```
struct Employee
  {
         int Id;
         char Name[25];
         int Age;
         long Salary;
  };
```

Accessing the structure members

We have to create an object of structure to access its members. Object is a variable of type structure. Structure members are accessed using the dot operator(.) between structure's object and structure's member name.

Syntax for creating object

```
struct struct-name obj;
```

Example for creating object & accessing structure members

```
#include<stdio.h>
 struct Employee
 {
        int Id;
        char Name[25];
        int Age;
        long Salary;
 };
 void main()
 {

        struct Employee E;              //Statement  1

               printf("\nEnter Employee Id : ");
               scanf("%d",&E.Id);

               printf("\nEnter Employee Name : ");
               scanf("%s",&E.Name);
printf("\nEnter Employee Age : ");
                scanf("%d",&E.Age);

                printf("\nEnter Employee Salary : ");
                scanf("%ld",&E.Salary);
printf("\n\nEmployee Id : %d",E.Id);
                printf("\nEmployee Name : %s",E.Name);
                printf("\nEmployee Age : %d",E.Age);
                printf("\nEmployee Salary : %ld",E.Salary);
    }
```

Output :

```
Enter Employee Id : 1
Enter Employee Name : Kumar
Enter Employee Age : 29
Enter Employee Salary : 45000

Employee Id : 1
Employee Name : Kumar
Employee Age : 29
Employee Salary : 45000
```

Statement 1 is creating an object E of Employee type.

Initialization of structure

Like normal variable structures can be initialized at the time of declaration. Initialization of structure is almost similar to initializing array. The structure object is followed by equal sign and the list of values enclosed in braces and each value is separated with comma.

Example for declaring & initializing structure at same time

```
#include<stdio.h>

struct Employee

{

        int Id;

        char Name[25];

        int Age;

        long Salary;

};
```

```
void main()

{

        struct Employee E = {2,"Suresh",35,35000};



                printf("\n\nEmployee Id : %d",E.Id);

                printf("\nEmployee Name : %s",E.Name);

                printf("\nEmployee Age : %d",E.Age);

                printf("\nEmployee Salary : %ld",E.Salary);

}
```

Output :

```
Employee Id : 1

Employee Name : Kumar

Employee Age : 29

Employee Salary : 45000
```

Sri SAI RAM ENGINEERING COLLEGE
INSTITUTE OF TECHNOLOGY
West Tambaram, Chennai - 44

Sairam INSTITUTIONS

SAIRAM DIGITAL RESOURCES

YEAR II

SEM III

Unit V

Z-TRANSFORM AND DIFFERENCE EQUATIONS

5.4 FORMATION OF DIFFERENCE EQUATIONS

MA8353

TRANSFORMS AND PARTIAL DIFFERENTIAL EQUATIONS (common to EEE, CIVIL, PROD, E&I, MECH)

SCIENCE & HUMANITIES

4 QUALITY EDUCATION

FORMATION OF DIFFERENCE EQUATIONS

Definition: Difference equations: A differential equation is a relation between the differences of an unknown function at one or more general values of the argument.

Thus $\Delta y(n+1) + y(n) = 2 \qquad \ldots\ldots(1)$

And $\Delta y(n+1) + \Delta^2 y(n-1) = 1 \qquad \ldots\ldots(2)$ are difference equations.

1. Form the difference equation corresponding to the family of curves y = ax + bx^2

Solution:

$$y_x = ax + bx^2 \qquad \ldots\ldots(1)$$

$$put\ n = n+1 \Longrightarrow y_{x+1} = a(x+1) + b(x+1)^2 \ldots\ldots(2)$$

$$put\ n = n+2 \Longrightarrow y_{x+2} = a(x+2) + b(x+2)^2 \ldots\ldots(3)$$

Eliminating a and b, we get

$$\begin{vmatrix} y_x & x & x^2 \\ y_{x+1} & x+1 & (x+1)^2 \\ y_{x+2} & x+2 & (x+2)^2 \end{vmatrix} = 0$$

$$\Rightarrow y_x[(x+1)(x+2)^2-(x+2)(x+1)^2]-y_{x+1}[(x)(x+2)^2-(x+2)(x)^2] + y_{x+2}[(x)(x+1)^2-(x+1)(x)^2]=0$$

$$\Rightarrow y_x\ (x+1)(x+2)[x+2-x-1]-y_{x+1}[(x)(x+2)][(x+2-x)] + y_{x+2}[(x)(x+1)][(x+1-x)]=0$$

$$\Rightarrow y_x\ (x+1)(x+2)-y_{x+1}2x(x+2)+y_{x+2}[x(x+1)]=0$$

$$\therefore (x^2+3x+2)y_x-2(x^2+2x)y_{x+1}+(x^2+x)y_{x+2}=0$$

2. Form the difference equation from $y_n = a + b3^n$

Solution:

Given: $y_n = a + b3^n$----(1)

$put\ n = n+1$ in (1)

Therefore we get $y_{n+1} = a + b3^{n+1}$-----(2)

$put\ n = n+2$ in (1)

We get, $y_{n+2} = a + b3^{n+2}$-----(3)

From equations (1), (2), and (3) Eliminating a, b we get

$$\begin{vmatrix} y_n & 1 & 1 \\ y_{n+1} & 1 & 3 \\ y_{n+2} & 1 & 3^2 \end{vmatrix} = y_{n+2} - 4y_{n+1} + 3y_n = 0$$

3. Derive the difference equation from $y_n = (A + nB)2^n$

Solution:

Given: $y_n = (A + nB)2^n$

$$y_n = 2^n A + Bn2^n ----(1)$$

$$put\ n = n + 1 \Rightarrow y_{n+1} = 2^{n+1}A + B(n+1)2^{n+1}$$
$$= 2.2^n.A + (n+1)B2^n ----(2)$$

$$put\ n = n + 2 \Rightarrow y_{n+2} = 2^{n+2}A + B(n+2)2^{n+2}$$
$$= 4.2^n A + 4(n+2)B2^n ----(3)$$

$From\ 1, 2\ and\ 3\ equations$, Eliminating A, B we get

$$\begin{vmatrix} y_n & 1 & n \\ y_{n+1} & 2 & 2(n+1) \\ y_{n+2} & 4 & 4(n+2) \end{vmatrix} = 0$$

$$\Rightarrow y_n[8(n+2) - 8(n+1)] - 1[4(n+2)y_{n+1} - 2(n+1)y_{n+2}] + n[4y_{n+1} - 2y_{n+2}] = 0$$

$$\Rightarrow y_n[8n + 16 - 8n - 8] - 1[(4n+8)y_{n+1} - (2n+2)y_{n+2}] + n4y_{n+1} - 2ny_{n+2} = 0$$

$$\Rightarrow y_{n+2} - 4y_{n+1} + 4y_n = 0$$

4. From $y_n = a2^n + b(-2)^n$ derive a difference equation by eliminating the constants.

Solution: Given $y_n = a2^n + b(-2)^n$

$$put\ n = n+1 \Rightarrow \quad y_{n+1} = a2^{n+1} + b(-2)^{n+1}$$

$$= 2a2^n - 2b(-2)^n$$

$put\ n = n + 2 \Longrightarrow$ $\quad y_{n+2} = a2^{n+2} + b(-2)^{n+2}$

$$= 4a2^n - 4b(-2)^n$$

Eliminating a,b we get,

$$\begin{vmatrix} y_n & 1 & n \\ y_{n+1} & 2 & -2 \\ y_{n+2} & 4 & 4 \end{vmatrix} = 0$$

$$\Longrightarrow y_n[8+8] - 1[4y_{n+1} + 2y_{n+2}] + 1[4y_{n+1} - 2y_{n+2}] = 0$$

$$\Longrightarrow y_n[16] - 4y_{n+1} - 2y_{n+2} + 4y_{n+1} - 2y_{n+2} = 0$$

$$\Longrightarrow 16y_n - 4y_{n+2} = 0$$

$$\Longrightarrow y_{n+2} - 4y_n = 0.$$

5. Derive the difference equation from $y_n = A2^n + nB$

Solution:

Given: $y_n = A2^n + nB$

$$y_n = 2^n A + Bn - - - -(1)$$

$$put\ n = n+1 \Rightarrow y_{n+1} = 2^{n+1}A + B(n+1)$$

$$= 2.2^n.A + (n+1)B - - - -(2)$$

$$put\ n = n+2 \Rightarrow y_{n+2} = 2^{n+2}A + B(n+2)$$

$$= 4.2^n A + (n+2)B - - - -(3)$$

$From\ 1, 2\ and\ 3\ equations$, Eliminating A, B we get

$$\begin{vmatrix} y_n & 1 & n \\ y_{n+1} & 2 & (n+1) \\ y_{n+2} & 4 & (n+2) \end{vmatrix} = 0$$

$$\Rightarrow y_n[2(n+2) - 4(n+1)] - 1[(n+2)y_{n+1} - (n+1)y_{n+2}] + n[4y_{n+1} - 2y_{n+2}] = 0$$

Vedio link:

https://youtu.be/vw6fzRd-kvs

Sri SAI RAM ENGINEERING COLLEGE
INSTITUTE OF TECHNOLOGY
West Tambaram, Chennai - 44

SAIRAM DIGITAL RESOURCES

YEAR IV SEM VII

ME 8792

POWER PLANT ENGINEERING

UNIT NO 1 - COAL BASED POWER LANTS

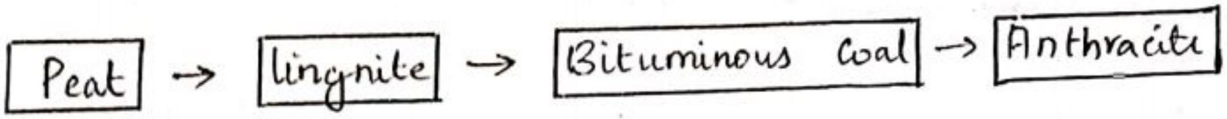
- 1.2. Fuel and Ash Handling

MECHANICAL ENGINEERING

1.2.1 COAL HANDLING:

Coal:

It is a fossil fuel that forms when dead plants are buried under earth surface for millions of years



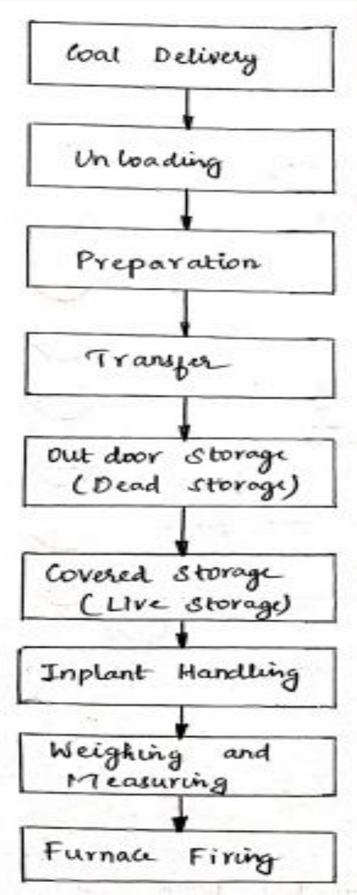
- Lingnite is low grade coal with high as content
- Anthracite is high grade coal with low ash content and Calorific Value
- In thermal power plant 50% to 60% of total operating cost is spent for coal and ash handling.

I. Stages of Coal Handling:

- **Coal Delivery:** Delivery stations near sea or river will use navigation facilities. Stations that cannot use navigation will use rail or truck.

- **Unloading:** If coal is transported through sea, the following unloading equipment are used:
  (i) Conveyors (ii) Unloading Bridges (iii) Self unloading boats
  (iv) Coal towers
- **Note:** If trucks are used for unloading no need for unloading devices

- **Out door storage (Dead storage):** The gross calorific value (CGV) of coal gets reduced due to air pollution, rainfall and moisture in outdoor storage.
- **Covered Storage (Live Storage):** Sufficient amount of coal to meet one day requirements of boiler is stored here

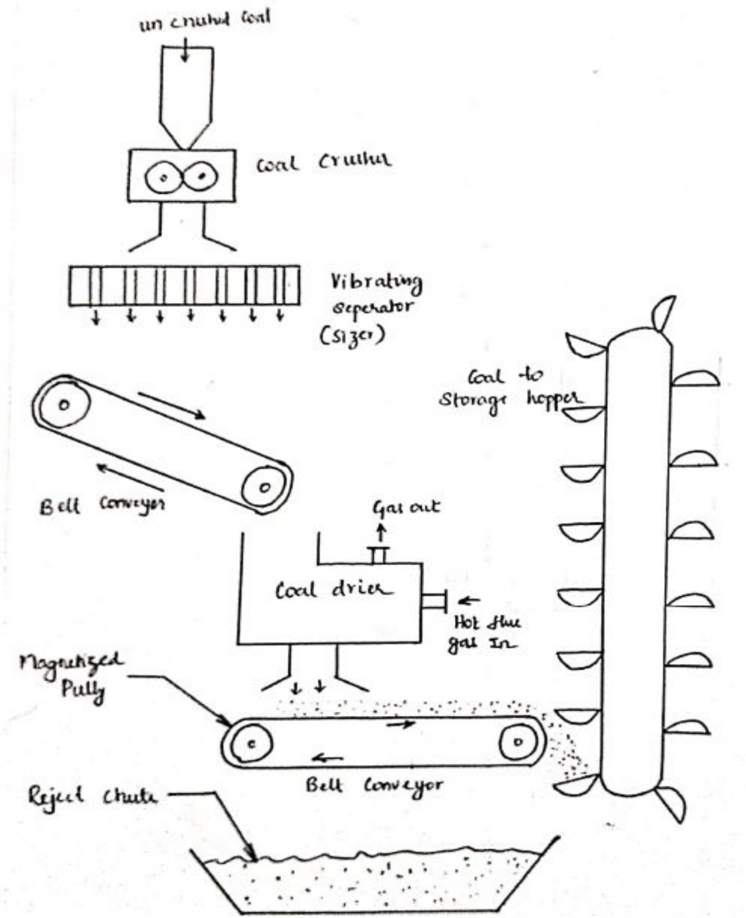


1.2.2 COAL PREPARATION:

Carried out in three main stages:

(i) Crusher (ii) Sizer (iii) Drier

- **Crusher:** Crushes the coal to required size approximately 10mm in size.
- **Sizer:** It is a kind of vibrating separator used to filter coal of large size
- **Coal drier:** Removes excess moisture from coal using flue gas that comes out from boiler.
- **Magnetic Separator:** It is a belt conveyor fixed with magnetized pulley to remove iron particles from coal.



1.2.3 TRANSFER:

- It is the coal handling between “unloading point” to “storage point” from where coal is discharged for firing.

Types of Equipment used:

(i) Belt Conveyors

(ii) Screw Conveyors

(iii) Bucket Elevators

(iv) Pivoted bucket Conveyors

(v) Grab bucket Conveyors

(vi) Flight Conveyors

(vii) Skip Hoist

(viii) Mass flow Conveyors

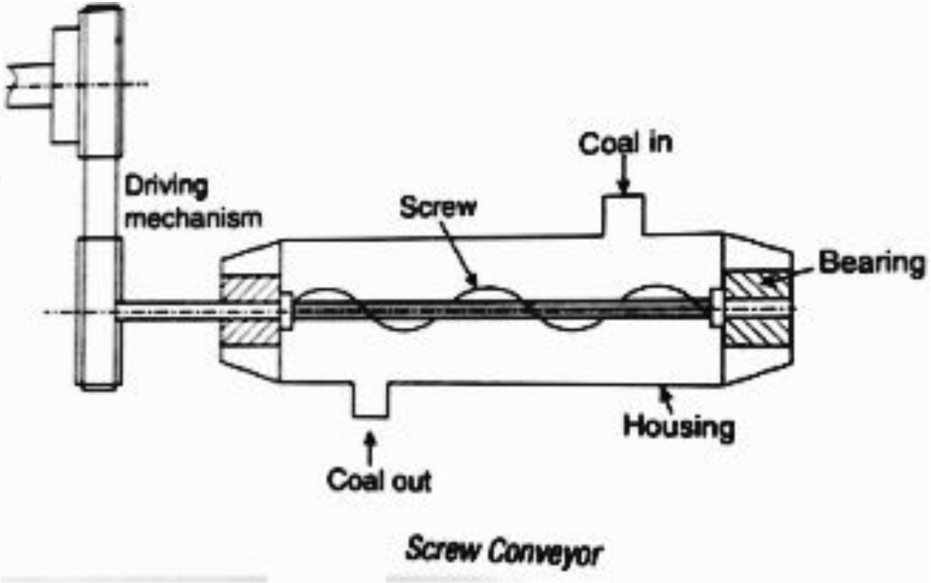
(ix) Chutes

I. Screw Conveyor:

- It is used for short distance
- It is mainly used in metering of coal and closed from atmosphere
- Can transfer coal at high inclinations
- One end of the shaft is attached with driving mechanism and the side with bearing

Specification:

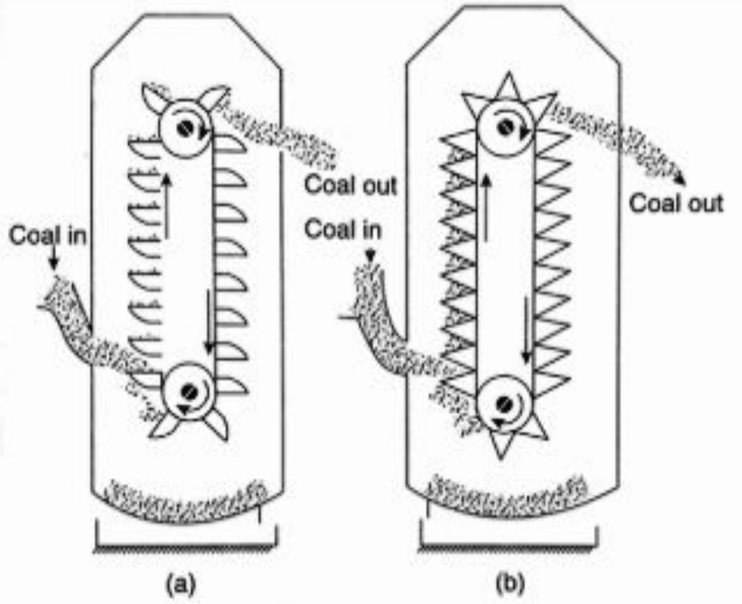
- Screw Diameter: 15 to 50cm
- Speed : 70 rpm to 120 rpm
- Length : Max. upto 30m
- Capacity : upto 125 tonnes/hour



Screw Conveyor

II. Bucket Elevator:

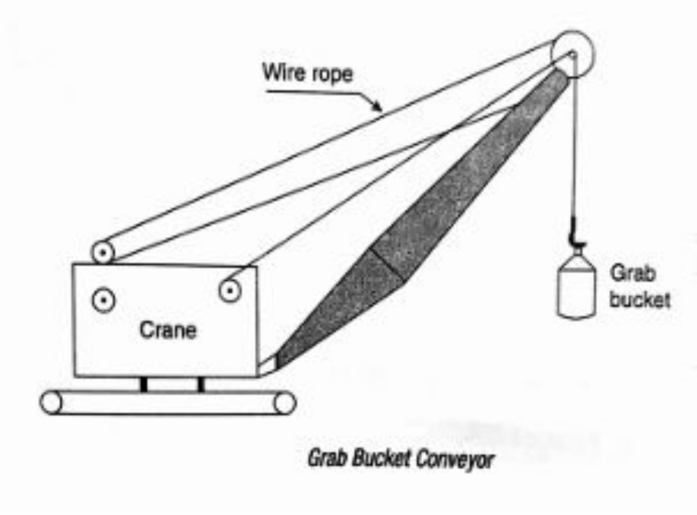
- It is used for vertical lifts
- Buckets are fitted to the chain
- Buckets are loaded at bottom and discharged at top.



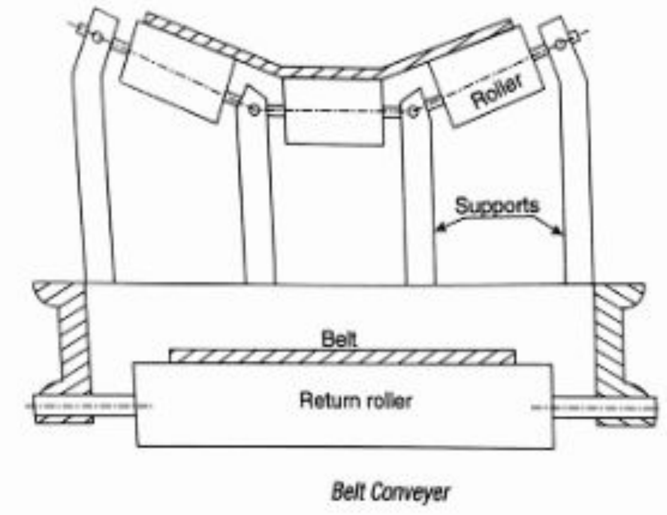
Bucket Elevators: (a) Centrifugal; (b) Continuous

Specification:

- Max. Height : 30.5m
- Max. Inclination : 30°
- Chain Speed : 75 m/min
- Capacity : 60 tonnes/hour



Grab Bucket Conveyor



Belt Conveyer

* It can grab as well as convey coal from one place to another

* It is used to transport large quantities of Coal over long distances

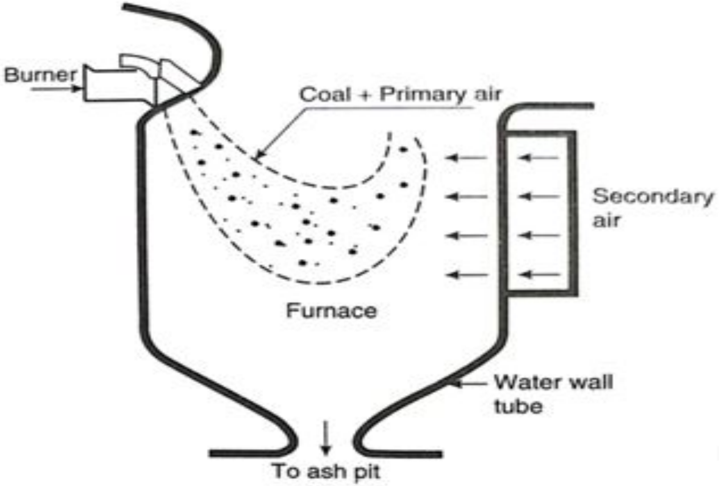
1.2.4. COAL BURNERS

Types of Burners:

(i) U – Flame Burner (ii) Tangential Burner (iii) Turbulent Burner (iv) Cyclone Burner

I. U – Flame Burner:

- In this burner, air and coal mixture travels a considerable distance thus providing sufficient time for complete combustion.



II. Tangential Burner:

- Fired in such a way that four flames are tangential to an imaginary circle formed at the centre.
- Heat released rate is very high and almost complete combustion can be achieved.

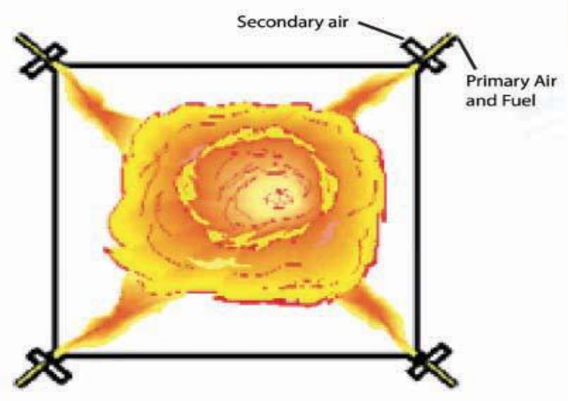
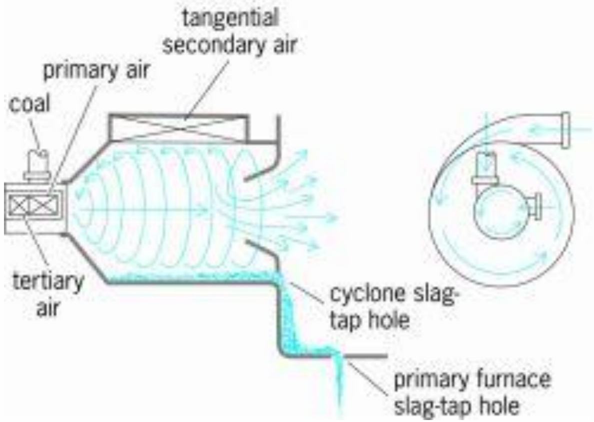


Fig: Tangential Firing for Pulverized Fuel

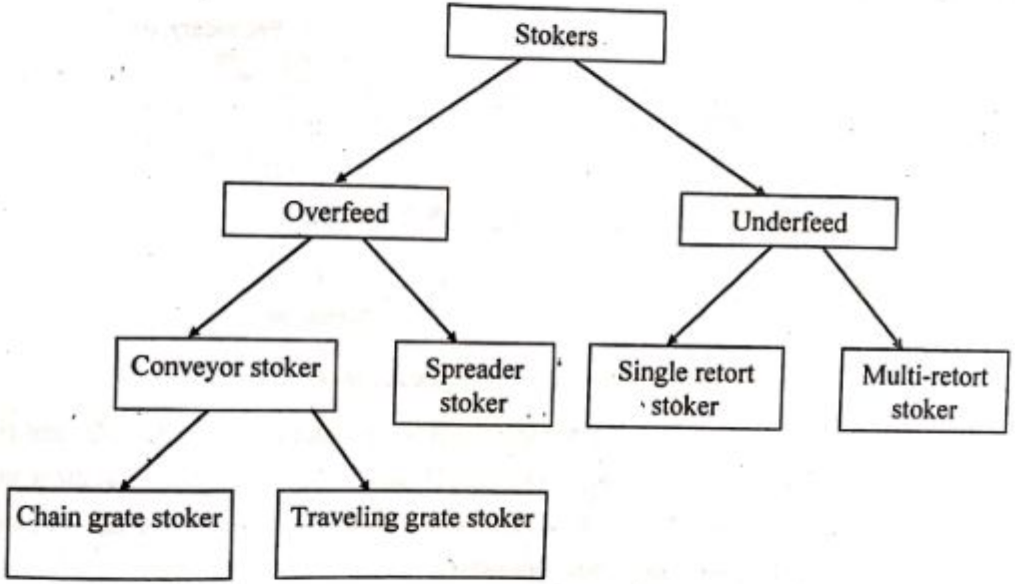
III. Cyclone Burner:

- The incoming coal particles are thrown to the walls by centrifugal force, and scrubbed by the high velocity tangential secondary air.
- Since water cooled surface is provided, more heat loss occurs through molten slag.



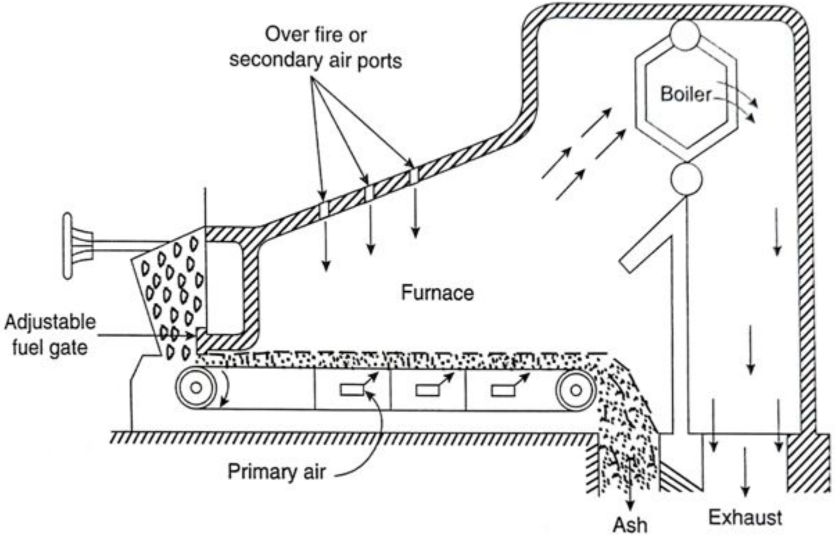
1.2.5. MECHANICAL STOKERS

- Stokers are used to feed solid fuels in to the furnace.
- Helps in achieving uniform operation condition, thereby reaching a higher burning rate.



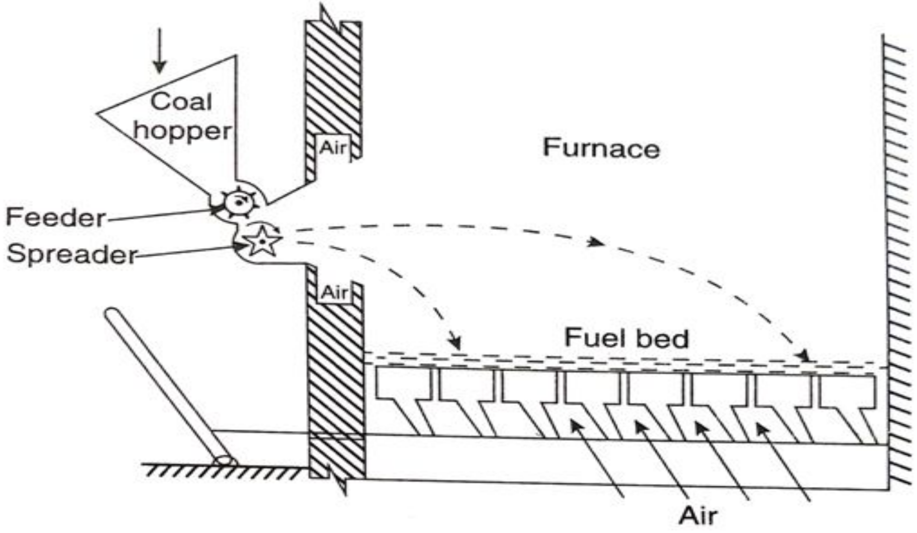
Types of mechanical stokers.

I. Chain Grate Stoker



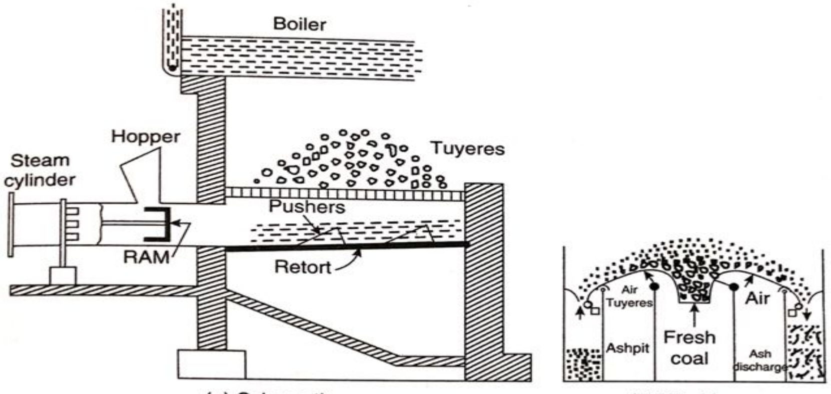
- Consist of Endless chain – supports fuel bed
- Chain travels over two sprockets (Front and rear)
- Front sprocket connected to driving mechanism.
- Coal fed hopper is located in front of the stoker.

II. Spreader Stoker



- Coal from hopper is fed on to the feeder.
- Feeder is a rotating drum (slow speed) fitted with blades.
- Spreader distributes coal evenly over entire grate area.
- Spreader consist of rapidly rotating shaft carrying blades
- Size of the coal selected should be 6 to 36 cm.
- Used for 10 – 30 MW capacity plant.

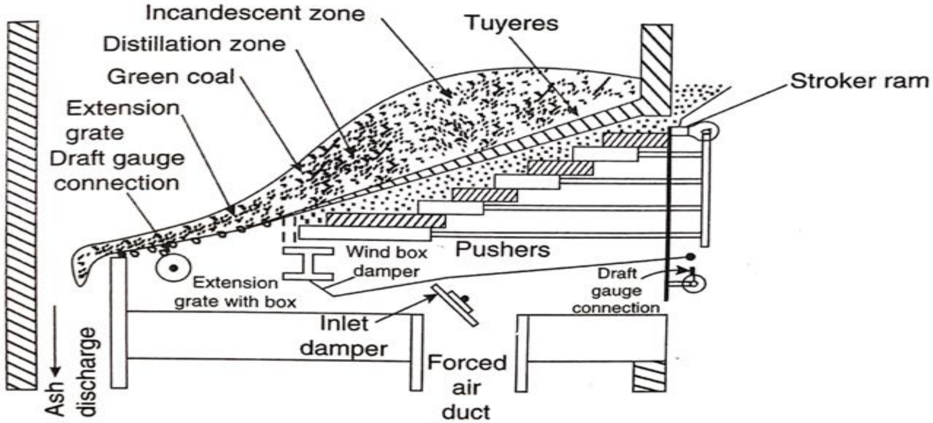
Single Retort Stoker



(a) Schematic

(b) Working

Multi Retort Stoker



- It consist of retorts and tuyere boxes for supply of air. Each retort is fitted with a reciprocating ram for feeding and pusher plates for uniform distribution of coal. The number of retorts may vary from 2 to 20 with burning capacity from 300 to 2000kg/h/retort

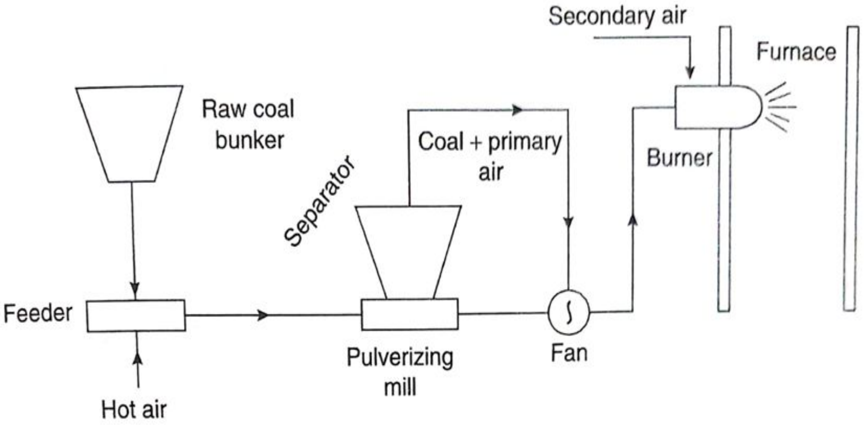
1.2.6. PULVARISED COAL FIRING

Divided into two systems:

(i) Unit or Direct System (ii) Bin or Central System

I. Unit or Direct System:

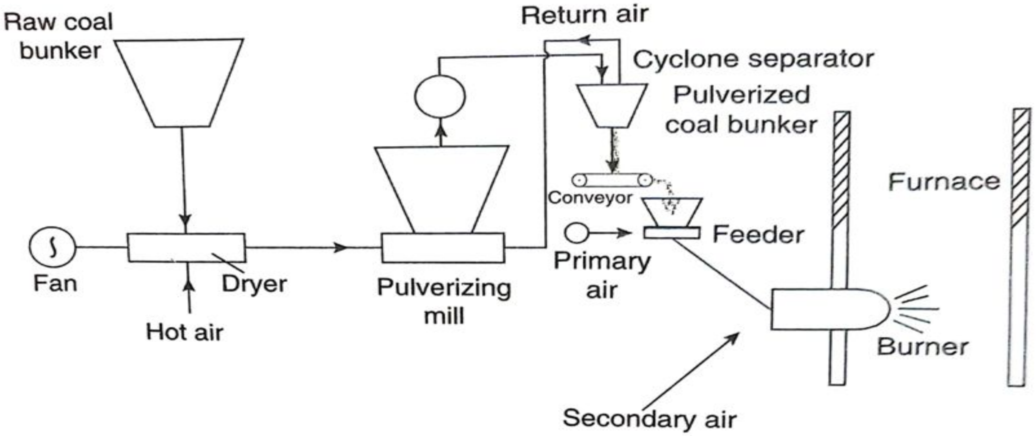
- The raw coal from bunker falls into feeder and their it is dried by supplying hot air. Coal then transferred to pulverizing mill, there it is pulverized.
- The mixture of pulverized coal and primary air then flows to burner where secondary air is added.



II. Bin or Central System:

- Coal from raw coal bunker is fed by gravity to a dryer where it is dried by hot air mill. The pulverized coal from the mill is transferred to pulverized coal bunker.

- The air from the coal is separated in the cyclone separator and returned to the mill. The primary air is mixed at the feeder and the mixture is supplied to the burner.



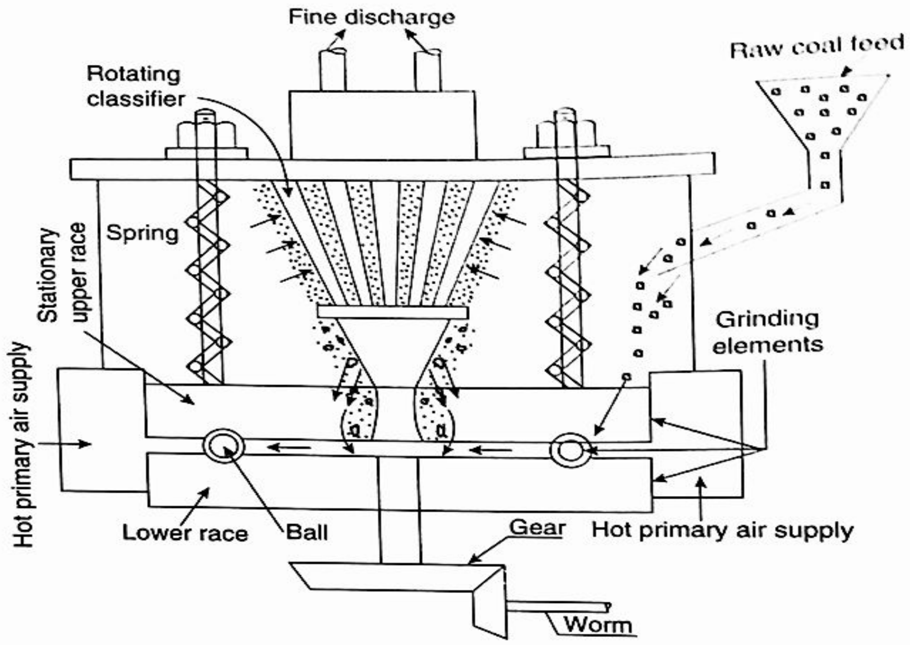
1.2.7. Pulverisers or Pulverising Mills

We have four major types of coal pulverising mills:

(i) Ball & race mill (ii) Bowl mill (iii) Impact mill (iv) Double classifier ball mill

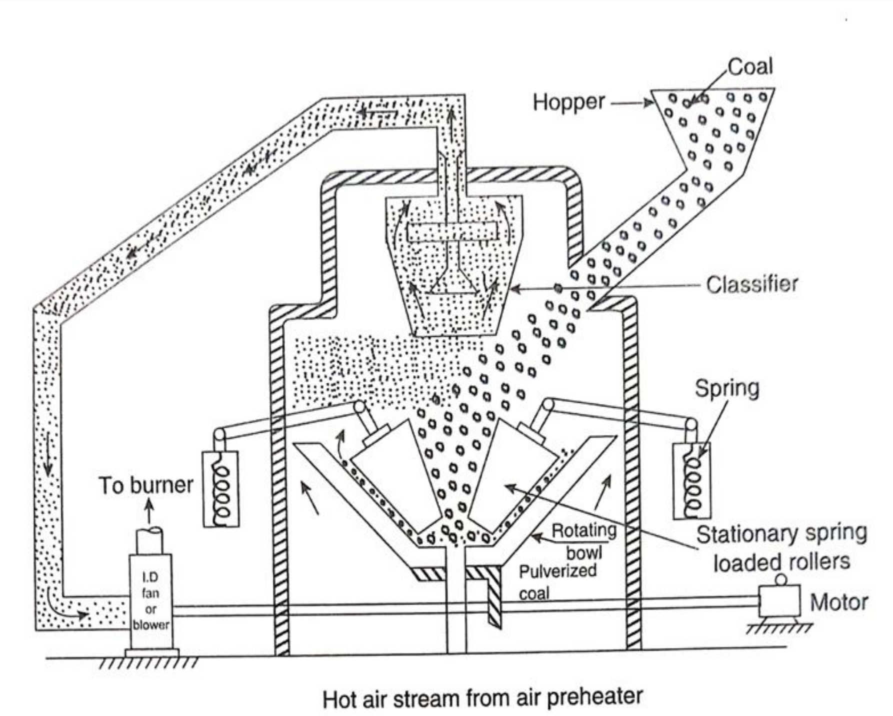
I. Ball and Race mill:

- ✔ The coal passes between the two rotating elements again and again until it has been pulverized to desired degree of finesses.
- ✔ After crushing the coal, it is sent to rotating classifier at where due to centrifugal force larger size coals stick to the walls and fine particles are discharged out through mesh.



II. Bowl mill:

- Bowl mills are employed to pulverize the pre-crushed coal.
- Crushed raw coal fed into the revolving bowl at a controlled rate
- The coal is speeded over the surface of the bowl uniformly due to the centrifugal force.
- The rollers will exert a required amount of centrifugal pressure with help of springs attached.
- The rollers do not touch the grinding bowl even when the mill is empty.

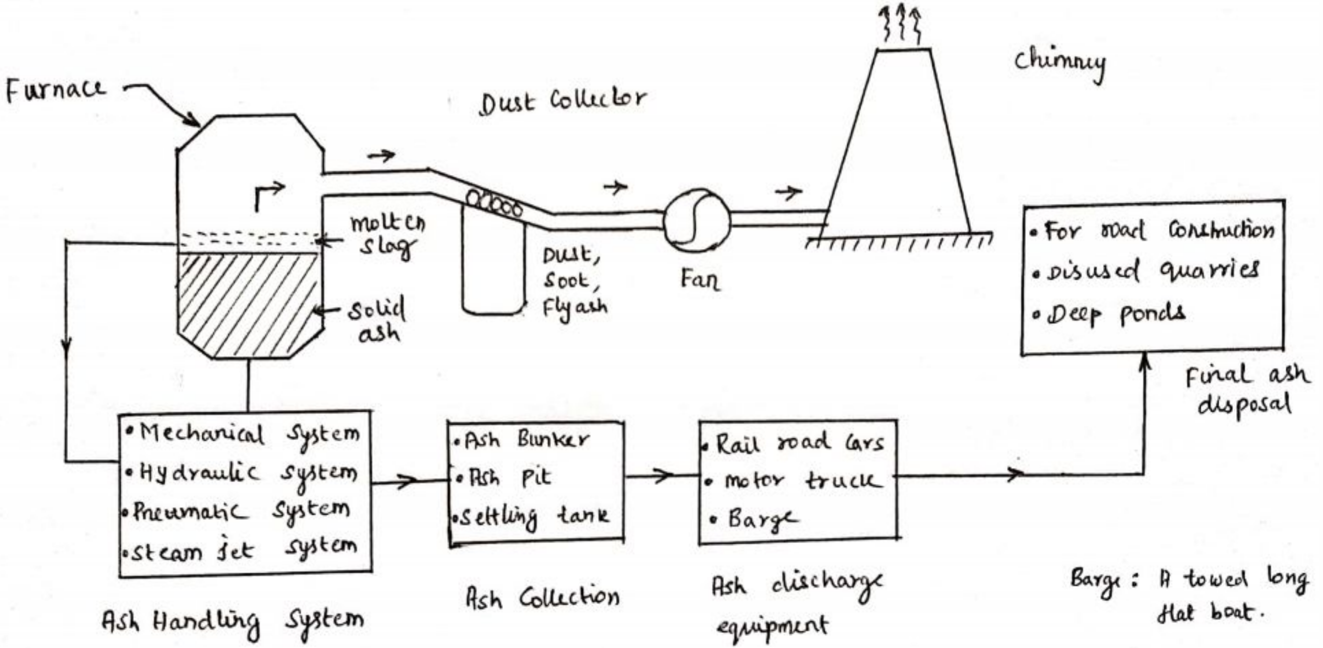


1.2.8. ASH HANDLING:

- ✔ About 10% to 20% of total quantity of coal burnt in a day is converted to ash
- ✔ Handling of ash is too hot, dusty, difficult to handle and emits poisonous gas.

Handling includes:

- Removal from furnace
- Loading on the conveyor
- Delivery to dump



Video Link:
https://www.youtube.com/wat

1.2.9. Ash Handling Systems

- **We have four major types of ash handling systems:**

(i) Mechanical ash handling system

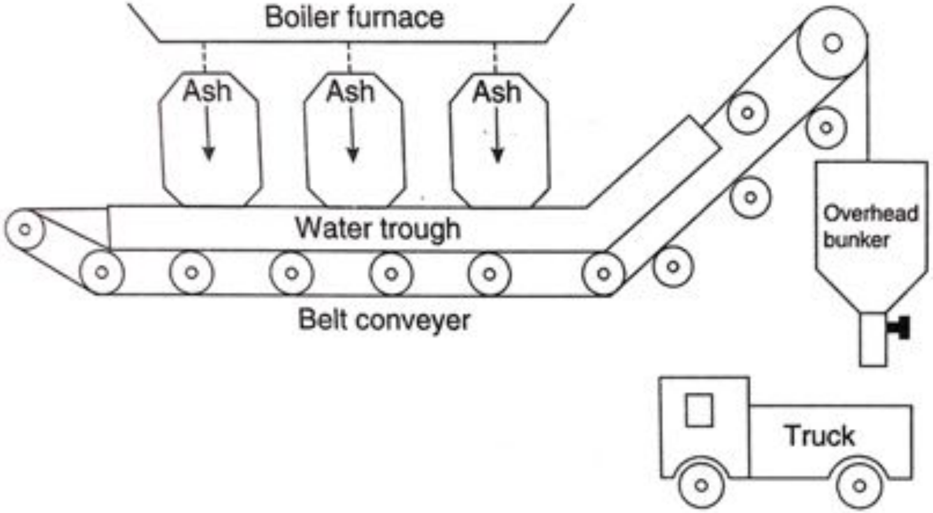
(ii) Hydraulic handling system

(iii) Pneumatic Handling system

(iv) Steam Jet system

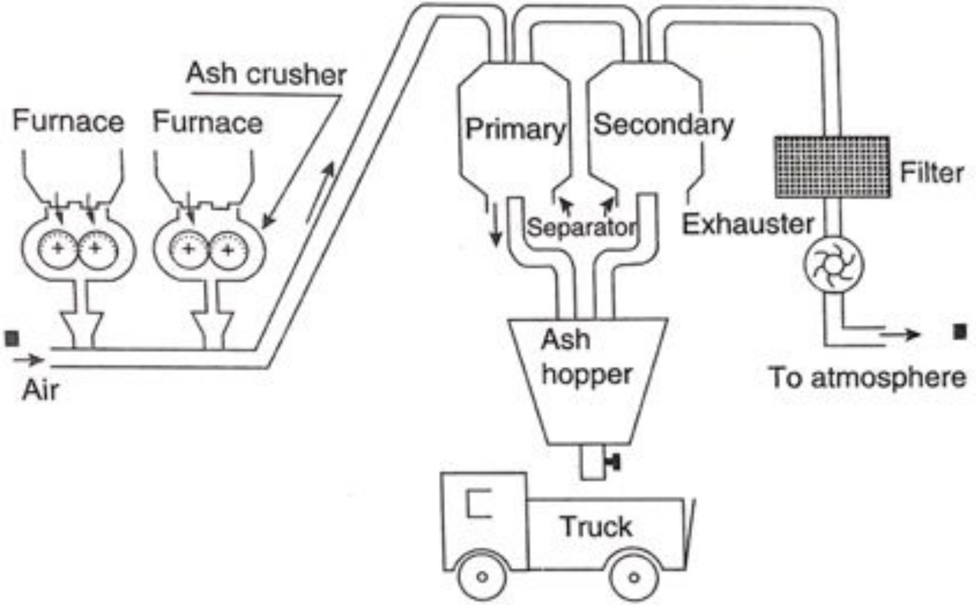
I. Mechanical Ash Handling System:

- Hot ash from Boiler furnace is fall through a water seal over Belt conveyor.
- The cooled ash is carried away by the belt to bunker and then to the dumping site.
- Only a limited amount ash can be handled by this system used especially in small capacity plants.



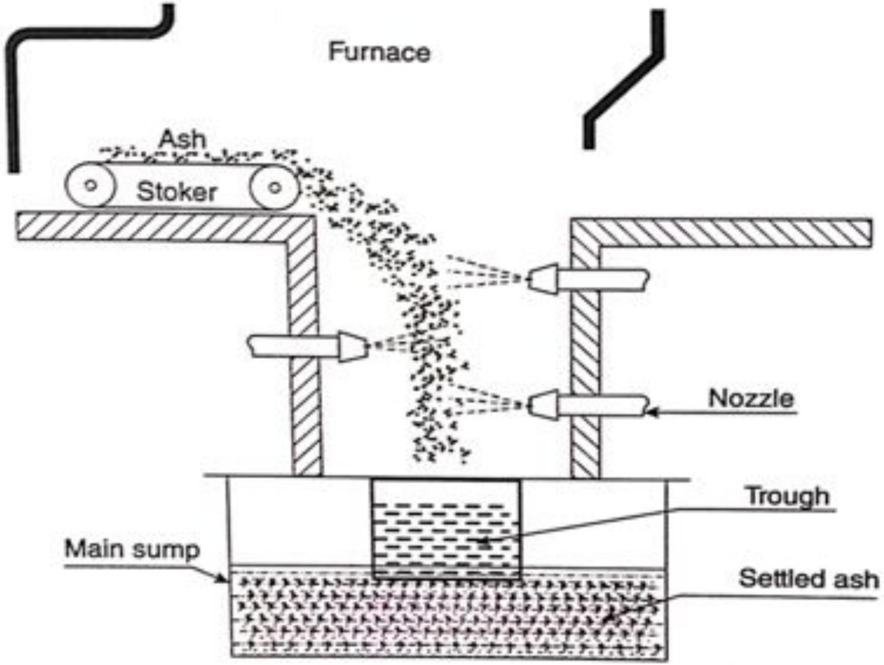
II. Pneumatic Ash Handling System

- Here air is used as a medium to carry ash over long distance at the rate of 5-30 t/hr
- The main parts of the systems consist of crushers, separators, hopper and ash filter.
- Heavier ash settles down in Hopper, tiny ash particles leaving secondary separator is further cleaned using Filter and exhausted into atmosphere.



III. Hydraulic System:

- Ash is quenched by top water nozzle spray and is carried away.
- The quenched ash flows along with water at high velocity and gets collected in sump.
- Water is separated from ash by passing it over screen and is re-circulated


Furnace
Ash
Stoker
Nozzle
Trough
Main sump
Settled ash