DATE: __/__/__
PAGE ____

Fundamentals of Information Technology

Unit 1 - Introduction to Computers
Unit 2 - Interaction with computers
Unit 3 - Computer Number System
Unit 4 - Computer Network

Unit - 1 INTRODUCTION TO COMPUTER

Computer → It is an electronic device, which takes the data, stores it, processes it, and gives the output.
Also called "Data Processor"

Q. Why called data processor?
Ans. It takes in data as raw facts, observations & processes it into useful, meaning ful information.
→ Information → Processed Data

Data Processing Task

Data → 1. Store the data
2. Manipulate / Provide some operations (Processing)
3. Output the results
↓
Information (Output / Result)
↓
Knowledge → Wisdom

Knowledge → Application of information towards different fields.

FIT

DATE: __/__/

PAGE

Applications of computer system

→ Education System
→ Military
→ Industry
→ Research
→ Medical

Basic Organisation of computer System.

↳ How every component of computer system is connected to each other.

5 types of tasks

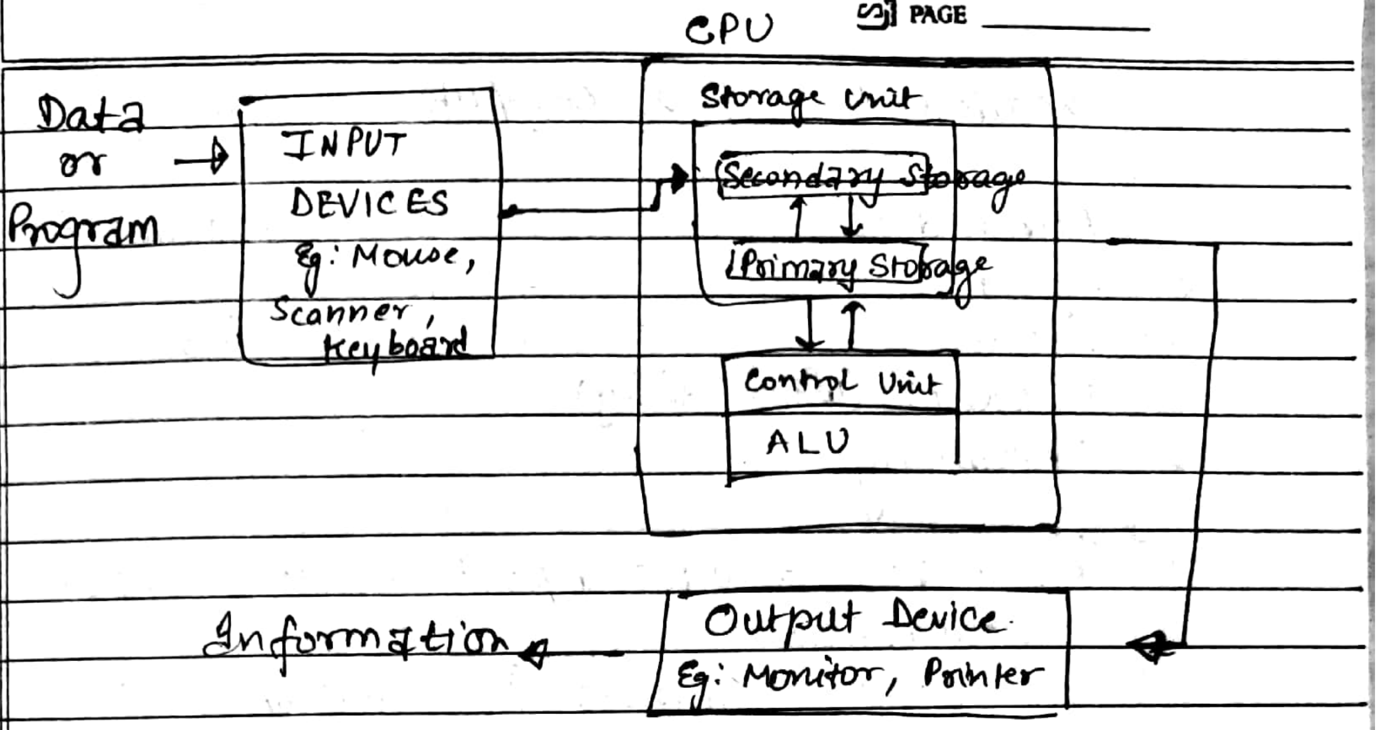
1. Inputting the data as program through input devices
2. Storing data into memory
3. Processing / Manipulating
4. Controlling
5. Outputting the results through output devices

CPU (main part)

↓ Control Unit (CU)

→ Arithmatic & Logical Unit (ALU)

↓ to supervise the different components that are connected to the computer system.



~~Input devices~~

	Main / Primary Memory	Secondary Memory
1.	Volatile	Non-volatile
2.	very fast becoz made of semi conductor devices	Not very fast
3.	Very expensive	less expensive
4.	~~more~~ less storage ~~and~~ space	more storage space
5.	Eg: RAM	Eg: Hard disk, Pendrive

Output Device

→ coverts the digital form into human readable form.

Input Device

→ takes raw facts as data from the outside world by default through the keyboard.

Charactersties of a computer System

1. Automatic
2. Speed
3. Accuracy
4. Diligence (without tiring)
5. Versatility
6. Power of Remembering
7. No IQ
8. No Feelings

→ Accuracy always depends upon computer architecture.
GIGO → Garbage In Garbage Out

Generation of Computer
↳ (Gap in Technology) → Gap in hardware + software)

→ It provides a framework for growth of computer industry.

Five Generations of Computer

First Generation

Period -: 1942 - 1955
Hardware - vaccum tube
→ single program based: untill one program gets completed & executed, we can't work on another program
For input / input device - Punch Card
Memory - Electromagnetic Relay Memory
→ programming was done using machine coding (using symbols 0 & 1)

→ Assembly language was also used.

Features :- 1. Bulky in size
2. Unreliable
3. More / Large heat emission
4. large room for vaccum tube

Examples :- ENIAC, EDVAC, UNIVAC, IBM701

ENIAC - Electronic numerator, integrator and computer

Second Generation

Period -: 1955 - 1964

→ Introduction to transistors

Memory - Magnetic Disk, Magnetic Tape

Softwares -: Introduction to FORTRAN, COBOL, ALGOL

COBOL - Common Business Oriented language

Features :-

Examples :- IBM7030, Honey well, CDC

Third Generation

Period -: 1964 - 1975

Use of Integrated Circuit (IC) Technology which is made up of different no. of components.

IC Technology

SSI

MSI

Small Scale Integrated Circuit

10 to 20 components (100 components)

(Transistors, diodes, etc)

Software Technology → High Level Language (HLL)

Features :- 1. Faster
2. More reliable
3. More Storage
4. cheaper

Examples:- IBM 360/370

Fourth Generation

Period :- 1975 - 1989

Hardware Used :- 1. IC's with LSI & VLSI technology

large scale (30,000 components were inbuilt into a single chip)

very large scale (1 million components)

2. Ram, Cache Memory used for primary memory
3. Large Sized Secondary storage devices
4. Floppy Disk, Magnetic Tapes were used for portable media.

Software Used :- * Operating system is Multi~~tasking~~ processing, Multiasking of MS windows
* UNIX OS
* MS - DOS
* High speed computer networking software (LAN, WAN) technologies.
* C language became very popular
* Development of Graphical User Interface (GUI)

- Personal computers
- Super computers with parallel processors

→ Hardware

Examples :- ~~[illegible]~~ IBM PC, VAX 9000, CRAY-1, CRAY-2

Fifth Generation

Period :- 1989 - Present
Hardware Used :- * Introduction to ULSI (Ultra Large Scale IC's)
→ (10 million electronic components)

* Increase in capacity in Main Memory, hard disk
* Optical disk introduction (CD-ROM)
Compact Disk - Read Only Memory
* Powerful Desktops, PC's & workstations
* Powerful mainframes
* Internal ~~feasibit~~ facility.

DATE: __/__/____
PAGE ________

Software Used :- www, e-commerce, mail, multimedia applications

Examples :- IBM notebook, Pentium PC's, SUN Microsystem, PARAM 10000

Classification of Computers

Based on size and performance wise
1. micro computers
   - PCs
   - laptops
   - Workstations
   - PDA (Personal Digital Assistant)
2. Mini Computers
3. Mainframes
4. Super Computers

Technology wise
1. Digital Computers
2. Analog computers
3. Hybrid Computers

Purpose wise
1. Specific purpose
2. General purpose

★ MODEM (Modulator Demodulator)
Modulator → Analog to Digital
Demodulator → Digital to Analog

★ LCD → Liquid Crystal Display

Size & Performance wise Digital computers are classified into the following four types:-

★ **Micro Computer** → A micro computer is a computer whose CPU is a micro processor. It is a processor where all ~~components~~ elements (electronic components) are inbuilt into a single IC chip. These are also called desktop machine as it is a single user system.
Eg: IBM PC's, DELL PC's, etc...

→ Micro computers are further divided into following types:-

1. **Personal Computer** :- It is a desktop machine. It is not very expensive and easy to use in any organization as well as at home.
Eg:- ~~Pentium~~ Pentium III, Pentium IV, etc.
2. **Laptop** :- Laptop Computers are called mobile computers. It is single user system powered with battery also. Mostly it uses similar types of harware as of PC but for displaying purpose it uses LCD instead of video monitor.

3. **Workstations** :- They are also called desktop machine but are more powerful and speed is about 10 times faster than the PC. These machines are largely used by engineers, architect and another professionals who needs detailed graphic
Eg: SUN Microsystem, IBM DEC

DATE: __/__/____
PAGE ______

4. Personal Digital Assistants : They are much smaller than PC's and Laptops and can be held in palm. It is basically used for Scheduling system and address book. It doesn't have a disk drive. It has limited memory and is less powerful.
Eg: Tablets, Palm top

★ Mini computers → These are also called mid range servers. They are more powerful than microcomputers in terms of processing power and capabilities. They are multiuser systems where many users (4-20) simultaneously work on that system. Eg:- PDP-11, VAX

★ Mainframe computers → The term 'mainframe' is used for large and very powerful system. These are multiuser, multiprocessing and high performing computers. It has very large storage as compared to mini-computers. They are used in Banks, railway reservations, etc.
Eg:- IBM 4381, CDC 6600, VAX 8842

★ Super Computers :- They are the most powerful computers among all the digital computers. They have high processing speed as compared to other systems. They ~~have~~ use parallel processors and do complex tasks.

The speed of super computers ~~are~~ is generally measured in ~~f~~ FLOPS (Floating Point operation per second)

Eg :- 1 Tera FLOPS = 2^{40} FLOPS

speed

eg :- (PARAM, ANURAG, PARAM PADMA)

↳ C-DAC (Pune)

Purpose wise :-

1. Specific Purpose → Not versatile
2. General Purpose → versatile

DATE: __/__/____
PAGE ______

Unit-3 Data Representation

How to represent data in computer memory?

Data in the form of :-

1. Numeric → 0 to 9 digits
2. Alphabets → A to Z and a to z
3. Alphanumeric → combination of alphabets + numerics
4. Special Character → +, -, %, /, *, ?

IMPORTANT TERMS

Bit → Basic Information stored in computer memory is called bit (Binary Digit)

Byte → collection of 8-bits

Nibble → collection of 4-bits

Word Size → size of the register

Eg: $\underbrace{\text{16 bits / word}}_{\text{register size}}$

Number System → A way to represent numbers into a computer system to store and process.

Every no. system has minimum value = 0 & maximum value = Base - 1

GOOD WRITE

Scanned by CamScanner

Number System

Non-Positional No. System

Eg: Roman Numerals

Positional No. System

Eg: Decimal No. System

$(256)_{10}$

$2\times10^2 + 5\times10^1 + 6\times10^0$

$200 + 50 + 6 = (256)_{10}$

$\boxed{6 \times 10^{0}}$ → Position

The digit itself

Base

Binary No. System

→ Binary to Decimal conversion

$(1\,0\,1\,0\,1)_2 \rightarrow 1\times2^4 + 0\times2^3 + 1\times2^2 + 0\times2^1 + 1\times2^0$

$= 16 + 4 + 1 = (21)_{10}$

Octal No. System

→ ~~Octal to Decimal Conversion~~

$(125)_8 \rightarrow$ Octal.

$(852)_8 \rightarrow$ Not Octal becoz it contains 8 as its element whereas maximum value can be 7 only.

Positional Number System

Decimal	Binary	Octal	HexaDecimal
Base = 10	Base = 2	Base = 8	Base = 16
Total no.s → 0-9	Total No → 0 & 1	Total No → 0 to 7	No. used are 0 to 9
Min = 0	Min = 0	Min = 0	A, B, C, D, E, F
Max = 9	Max = 1	Max = 7	10, 11, 12, 13, 14, 15

Q. Convert hexadecimal to decimal.

1. $(25)_{16}$

$$= 2\times16^1 + 5\times16^0 = 37$$

<u>Ans</u> $(37)_{10}$

2. $(1B6)_{16}$

$$= 1\times16^2 + B\times16^1 + 6\times16^0$$
$$= 256 + 11\times16 + 6\times1$$
$$= 256 + 176 + 6 = 438$$

<u>Ans.</u> $(438)_{10}$

Binary = Base 2 = 2^1 → n-bit
Octal = Base 8 = 2^3
Hexadecimal = Base 16 = 2^4

$$\underbrace{10^4 \quad 10^3 \quad 10^2 \quad 10^1 \quad 10^0}_{\text{integer part}} \;.\; \underbrace{10^{-1} \quad 10^{-2} \quad 10^{-3}}_{\text{Fraction part}}$$

Conversion

It plays an important role in data representation as we use decimal number system externally, but internally we use binary number ~~system~~ equivalent (i.e. in terms of 0 & 1)

TYPE 1 Conversion from Decimal to Binary, Octal and Hexadecimal Number System

OR

Conversion from Decimal to Another Number System

Case 1 To convert decimal integer into Binary, Octal and Hexadecimal

Example 1 Convert decimal 25 to Binary, Octal & HD

a) $(25)_{10} = (?)_2$

To Base	Number	Remainder
2	25	1
2	12	0
2	6	0
2	3	1
2	1	1
	0	

Ans. $(11001)_2$

b) $(25)_{10} = (?)_8$

8	25	1
8	3	3
	0	

Ans. $(31)_8$

c) $(25)_{10} = (?)_{16}$

To Base	Number	Remainder
16	25	9
16	1	1
	0	

Ans. $(19)_{16}$

Q. $(16)_{10} = (?)_2$

2	16	0
2	8	0
2	4	0
2	2	0
2	1	1
	0	

Ans. $(10000)_2$

Q $(94)_{10} = (?)_{16}$

16	94	14=E
16	5	5
	0	

Ans. $(5E)_{16}$

Steps to follow

Step 1 → Divide the decimal Integer to
(a) Binary – then divide by 2
(b) Octal – then divide by 8
(C) Hexadecimal – then divide by 16

Step 2 → Record the remainder in the table
Step 3 → Repeat step 1 with the quotient and then perform step 2 until the quotient becomes 0 or less than the base to be converted.

Step 4 → Write the result on the order starting from the last remainder to the first one i.e. from bottom to top.

Case 2 Converting decimal fraction into Binary, Octal and Hexadecimal

Q: $(0.625)_{10} = (?)_2$
$(0.625)_{10} = (?)_8$
$(0.625)_{10} = (?)_{16}$

a) $(.625)_{10} = (?)_2$

Fractional Number	Base		Integer Part (Result)
0.625	× 2	= 1.250	1
0.250	× 2	= 0.500	0
0.500	× 2	= 1.000	1

Ans. $(.101)_2$

b) $(.625)_{10} = (?)_8$ $\quad 0.625 \times 8 = 5.000 \quad 5$

Ans $= (.5)_8$

c) $(.625)_{10} = (?)_{16}$ $\quad 0.625 \times 16 = 10.000 \quad 10 = A$

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

Q: $(25.625)_{10} = (?)_2$

Case 1 → $(25)_{10} \rightarrow (11001)_2$

Case 2 → $(0.625)_{10} \rightarrow (.101)_2$

Ans. $(11001.101)_2$

DATE : __/__/____
PAGE ______

TYPE 2 Conversion of any base (Binary, Octal, H.D., base 4, base 6, etc) into Decimal Number System

Eg:- ① $(1011)_2 = (?)_{10}$

② $(1011.11)_2 = (?)_{10}$

③ $(123)_8 = (?)_{10}$

④ $(123.35)_8 = (?)_{10}$

⑤ $(1A5.6B)_{16} = (?)_{10}$

⑥ $(121)_4 = (?)_{10}$

⑦ $(512)_6 = (?)_{10}$

Ans ①

$$\begin{matrix} 1 & 0 & 1 & 1 \\ 2^3 & 2^2 & 2^1 & 2^0 \end{matrix}$$

$= 2^3 \times 1 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$

$= 8 + 0 + 2 + 1 = 11$

Ans. $(1011)_2 = (11)_{10}$

② $(1011.11)_2$

$$\begin{matrix} 1 & 0 & 1 & 1 & . & 1 & 1 \\ 2^3 & 2^2 & 2^1 & 2^0 & & 2^{-1} & 2^{-2} \end{matrix}$$

$= 2^3 \times 1 + 2^2 \times 0 + 2^1 \times 1 + 2^0 \times 1 + 1 \times 2^{-1} + 2^{-2} \times 1$

$= 8 + 0 + 2 + 1 + \cancel{\tfrac{1}{2} + \tfrac{1}{4}}\ 0.5 + 0.25$

$= (11.75)_{10}$

(3) $(123)_8$

$$\begin{array}{ccc} 1 & 2 & 3 \\ 8^2 & 8^1 & 8^0 \end{array}$$

$= 8^2 \times 1 + 8^1 \times 2 + 8^0 \times 3$
$= 64 + 16 + 3$
$= (83)_{10}$

(4) $(123.35)_8$

$$\begin{array}{cccccc} 1 & 2 & 3 & . & 3 & 5 \\ 8^2 & 8^1 & 8^0 & & 8^{-1} & 8^{-2} \end{array}$$

$= 8^2 \times 1 + 8^1 \times 2 + 8^0 \times 3 + 8^{-1} \times 3 + 8^{-2} \times 5$
$= 64 + 16 + 3 + \frac{3}{8} + \frac{5}{64}$
$= 83 + 0.375 + 0.078$
$= (83.453)_{10}$

(5) $(1A5.6B)_{16}$

$$\begin{array}{cccccc} 1 & A & 5 & . & 6 & B \\ 16^2 & 16^1 & 16^0 & & 16^{-1} & 16^{-2} \end{array}$$

$= 16^2 \times 1 + 10 \times 16 + 5 \times 16^0 + 16^{-1} \times 6 + 16^{-2} \times 11$
$= 256 + 160 + 5 + 0.375 + 0.042$
$= (421.417)_{10}$

(6) $(121)_4$

$$\begin{array}{ccc} 1 & 2 & 1 \\ 4^2 & 4^1 & 4^0 \end{array}$$

$= 4^2 \times 1 + 4^1 \times 2 + 4^0 \times 1$
$= 16 + 8 + 1$
$= (25)_{10}$

(7) $(512)_6$

$$\begin{array}{ccc} 5 & 1 & 2 \\ 6^2 & 6^1 & 6^0 \end{array}$$

$= 5 \times 6^2 + 1 \times 6^1 + 2 \times 6^0$

$= 180 + 6 + 2$

$= (188)_{10}$

$36 \times 5 = 180$

TYPE 3 Binary to Octal and vica-versa

Eg:- 1. $(101)_2 = (?)_8$

$\leftarrow$ Right to Left

$(101)_2 = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$

$= 4 + 0 + 1$

$= (5)_8$ ↳ Ans.

Table

0	0	0	→ 0
0	0	1	→ 1
0	1	0	→ 2
0	1	1	→ 3
1	0	0	→ 4
1	0	1	→ 5
1	1	0	→ 6
1	1	1	→ 7

2. $(10111101)_2 = (?)_8$

$$\underbrace{0\ 1\ 0}_{2}\ \ \underbrace{1\ 1\ 1}_{7}\ \ \underbrace{1\ 0\ 1}_{5}$$

Ans. $(275)_8$

* each octal equivalent can be converted into 3 bits

DATE: __/__/____ PAGE ______

3. $(5)_8 = (?)_2$

2	5	1
2	2	0
2	1	1
	0	

Ans. $(101)_2$

4. $(275)_8 = (?)_2$

2 → 010, 7 → 111, 5 → 101

~~2~~	~~275~~	~~1~~	~~1~~
~~2~~	~~137~~	~~6~~	~~0~~
~~2~~	~~68~~	~~1~~	~~1~~
~~2~~	~~34~~	~~1~~	~~1~~
~~2~~	~~17~~	~~1~~	~~1~~
~~2~~	~~8~~	~~1~~	~~1~~
~~2~~	~~4~~	~~0~~	~~0~~
~~2~~	~~2~~	~~4~~	~~1~~
~~2~~	~~1~~		~~0~~
	~~0~~		

Ans. $(010\ 111\ 101)_2$

5. $(523)_8 = (?)_2$

5 → 101, 2 → 010, 3 → 011

Ans. $(101\ 010\ 011)_2$

Type 4 Binary to H.D and vica-versa

$16 = 2^4$ → 4 bits are needed for making a group.

GOOD WRITE

DATE: __/__/____
PAGE ______

Table

0 0 0 0	→ 0
0 0 0 1	→ 1
0 0 1 0	→ 2
0 0 1 1	→ 3
0 1 0 0	→ 4
0 1 0 1	→ 5
0 1 1 0	→ 6
0 1 1 1	→ 7
1 0 0 0	→ 8
1 0 0 1	→ 9
1 0 1 0	→ A = 10
1 0 1 1	→ B = 11
1 1 0 0	→ C = 12
1 1 0 1	→ D = 13
1 1 1 0	→ E = 14
1 1 1 1	→ F = 15

Eg:- ① $(1011110101)_2 = (?)_{16}$

→ Right to left

→ Make a group of 4 bits

→ Write equivalent HD no.

$\underbrace{0010}_{2}\ \underbrace{1111}_{F}\ \underbrace{0101}_{5}$

Ans. $(2F5)_{16}$

② $(2F5)_{16} = (?)_2$

2 → 0010, F → 1111, 5 → 0101

Ans. $(001011110101)_2$

③ $(01011110.11)_2 = (?)_{16}$

$\underbrace{0101}_{5}\ \underbrace{1110}_{E} . \underbrace{1100}_{C}$

GOOD WRITE

Ans: (5E·C)16

Binary Arithmatic.

When arithmatic operations (eg: Addition, Subtraction, Multiplication & Division) applied to binary numbers, then it is called Binary Arithmatic.

- Computer actually doing Binary arithmatic operations
- Two most important operations are :-
 1) Binary Addition &
 11) Binary Subtraction

Rules for Binary addition of 2 digits

a	b	sum	carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

→ Truth Table

$$\begin{array}{r} 1 \\ +\ 1 \\ \hline 2 \end{array}$$

2 → 10 in binary form

1 0
↓ carry ↓ sum

Eg:
$$\begin{array}{r} {}^{1}\,{}^{1}\ \ \ \ \\ 1\ 1\ 0 \\ 0\ 1\ 1 \\ \hline 1\ 0\ 0\ 1 \end{array}$$

★ carry is always added to the left side

Rules for Binary addition of 3 digits

Decimal	a	b	c	Sum	Carry
0	0	0	0	0	0
1	0	0	1	1	0
2	0	1	0	1	0
3	0	1	1	0	1
4	1	0	0	1	0
5	1	0	1	0	1
6	1	1	0	0	1
7	1	1	1	1	1

→ Truth table

Carry: 1 1

Eg:
```
 1100
 0110
-----
10010
```

Eg: (Verify, decimal form)
```
 11            carry
 10    →   2
  1    →   1
  1    →   1
----      ---
100    →   4
```

Examples

a)
```
  1       0       0      1 1
+ 0     + 0     + 1    +   1
---     ---     ---    -----
  1       0       1     1 0
```

b)
```
  1
  10 →  2          11 →  3
+ 11 → +3        + 00 → +0
----   --        ----   --
 101 →  5          11 →  3
```

c)
```
  11
  11 →  3
+ 11 → +3
----   --
 110 →  6
```

Q. Perform Binary addition for the following:-

a)
```
  1100 → 12
+ 1010 → 10
 10110 → 22
```

b)
```
  1001 → 9
+ 1111 → 15
 11000 → 24
```

c)
```
  0110 → 6
+ 1110 → 14
 10100 → 20
```

d)
```
   111100 → 60
+ 0011011 → 27
  1010111 → 87
```

Binary Subtraction

Rules for Binary subtraction of 2 digits

a	b	Difference	Borrow
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

→ Truth Table

Eg:-
```
  11 → 3
- 10 → 2
  01 → 1
```

Rules for subtraction of 3 digits

Decimal	a	b	c	Difference	Borrow
0	0	0	0	0	0
1	0	0	1	1	1
2	0	1	0	1	1
3	0	1	1	0	1
4	1	0	0	1	0
5	1	0	1	0	0
6	1	1	0	0	0
7	1	1	1	1	1

→ Truth Table

Eg:-

```
   1   → First we subtract 1 from 1.
 - 1     Ans = 0
 - 1     Then again we have to subtract 1 from 0
 ---     Now, we'll borrow.
  11     0 becomes 10 i.e 2 in decimal
         Hence 2-1 = 1  OR 10-1 = 1
```

Ques Perform Binary subtraction for following numbers:-

a)
```
   1 1 1 1 → 15
 - 1 0 0 1 → 9
 ---------
   0 1 1 0 → 6
```

c)
```
   0 1 1 0 → 6
 - 0 0 1 0 → -2
 ---------
   0 1 0 0 → 4
```

b)
```
     0
   1 X 0 0 → 12
 - 1 0 1 0 → -10
 ---------
   0 0 1 0 → 2
```

d)
```
   1 1 1 0 0 0 → 56
 - 0 1 1 0 1 0 → -26
 -------------
     1 1 1 1 0 → 30
```

Binary Subtraction using Additive Method (also called complementary method)

Q. What is complement of a number?

→ $a-b = a+(-b)$

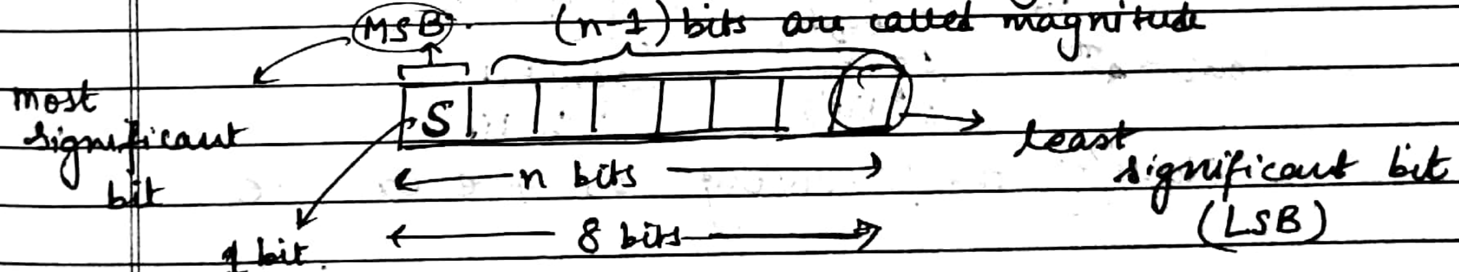
+12, +15, +0 } → unsigned integers

−12, −15, −0, +12, +15, +0 } signed integers

+12.50, +15.20, +0.0 } → unsigned real / floating no.

−12.50, −15.20, −0.0 } signed real no.s

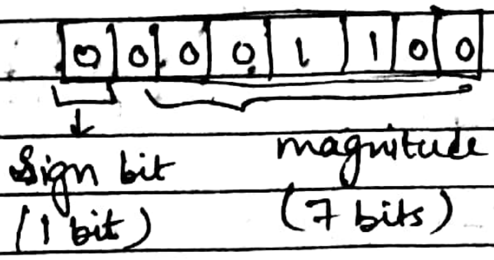
8 bits representation of a binary number



Binary No = Sign Bit + Magnitude

Represent $(+12)_{10}$ and $(-12)_{10}$ into 8 bits register

→ $(+12)_{10}$



NOTE

if S = 0 +ve no.
S = 1 −ve no.

→ But the negative binary number can be represented by using the following method :-

1. Signed magnitude method
2. 1's complement
3. 2's complement.

I. Signed Magnitude Method.

The MSB is used for signed bit (1-bit) and ~~are~~ occupied with value 1, the rest of the bits (n-1) are represented for the actual number (magnitude)

$(-12)_{10}$ → | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |

becoz -ve no.
Sign bit (1-bit)
magnitude (7 bits)

Q. Identify the following as +ve or -ve

1. 10010001 → -ve becoz sign bit is 1.
2. 00011110 → +ve becoz sign bit is 0.

II 1's Complement

Q Represent $(-12)_{10}$ into 1's complement

$(+12)_{10}$ = 00001100

1's complement 11110011

↳ change 1 into 0 and vica-versa

III 2's Complement

1. Find 1's complement of the number
2. Add 1 at L.S.B (least significant bit)

Q. Find $(-12)_{10}$ into 2's complement from using 8 bits.

$(+12)_{10}$ → 00001100

1's complement 11110011

+1

2's complement 11110100

Ques 0-4
= 0+(-4) -0 → 10000000

Eg:- +0 00000000

1's complement → 11111111

+1

2's complement (1) 00000000

neglect becoz 8 bit representation.

NOTE :- a-b
a+(-b)
(-b) should be represented in 3 forms -
(I) Signed magnitude
(II) 1's complement
(III) 2's complement.

DATE: __/__/____
PAGE ______

4+(-0)

Q. 4-0 using signed magnitude

$(+4)_{10} \Rightarrow (00000100)_2$

$+0 \rightarrow$ 00000000

$-0 \rightarrow$ 10000000

4 → 00000100
+ -0 → 10000000

10000100

-4

The answer is not accurate as it should come +4.

Q4 4-0 using 1's complement

+4 → 00000100

~~+0 → 00000000~~
1st ~~complement of +0 → 11111111~~

+0 → 00000000

1's complement (-0) → 11111111

DATE: __/__/____
PAGE ______

+4	00000100
+ −0	11111111
	1 00000011

⟹ This ans. is 3] It is not precise and accurate.

↳ neglect becoz 8 bit

Ques 4−0 using 2's complement

+4 → 00000100

+0 → 00000000

1's complement → 11111111
+1

2's complement → 1 00000000
↳ neglect

+4 →	00000100
+ −0 →	00000000
	00000100

→ This ans. is 4] It gives accurate ans. Hence, this method is used.

Complement Method

↓

r's complement | (r-1)'s complement

where r = Base of No. System

$r = 2$ Binary have 2's and 1's complement
$r = 10$ Decimals have 10's and 9's complement
$r = 8$ Octal have 8's and 7's complement
$r = 16$ Hexadecimal have 16's and 15's complement
$r = 4$ Base 4 have 4's and 3's complement

For (r-1)'s complement system,
Complement of a no. $(C) = B^n - 1 - N$

where B = base
n = total no. of digits
N = the actual no.

Q. To find out complements of binary no. $(101)_2$

Ans. $C = 2^3 - 1 - (101)_2$
$= 7 - (101)_2 = (111)_2 - (101)_2$

$$\begin{array}{r} 111 \\ -\,101 \\ \hline 010 \end{array}$$

$C = 010$ i.e 2 Ans.

Q. Find the complement of $(37)_{10}$

Ans $C = B^n - 1 - N$

$= 10^2 - 1 - (37)_{10}$

~~100 - 1~~

$= 99 - 37 = 62$ Ans.

Q. Find the complement of $(6)_8$

Ans $C = B^n - 1 - N$

$C = 8^1 - 1 - (6)_8$

$C = 7 - 6 = 1$ Ans.

GOOD WRITE

Scanned by CamScanner

Rules for subtraction using (r-1)'s complement Method

Step 1 Find the complement of a number you want to subtract (subtrahend)

Step 2 Add this to the no. (minuend)

Step 3 1) if there is carry of 1, add it to the result.

2) if there is no carry i.e. carry = 0,

a) recomplement the result
b) Put a -ve symbol to obtain the final result

Eg:- a - b = Result

a → minuend
b → subtrahend

$\Rightarrow a + (-b)$

Q1. Subtract $(56)_{10}$ from $(92)_{10}$ using complementary method. [Decimal system]

Ans

92 - 56 → we'll find the complement of 56

step 1 $C = 10^2 - 1 - 56$
$= 99 - 56 = (43)_{10}$

Step 2 $92 + 43$

$$\begin{array}{r} 92 \\ +\ 43 \\ \hline 1\ 35 \\ +1 \\ \hline 36 \end{array}$$

carry is added to the LSB

Step 3 $(36)_{10}$

Q2 Subtract $(35)_{10}$ from $(18)_{10}$

Ans

$18 - 35$

Step 1 $C = 10^2 - 1 - 35$
$C = 99 - 35 = (64)_{10}$

Step 2 $18 + 64$

$$\begin{array}{r} 18 \\ +\ 64 \\ \hline 82 \end{array}$$

i.e no carry generated.

Step 3 (ii) a)

$C = 10^2 - 1 - 82$
$= 99 - 82$
$= 17$

b) Putting -ve sign to the result.

Result $= (-17)_{10}$

Q. Subtract a) $(6)_8$ from $(3)_8$
b) $(16)_8$ from $(30)_8$ } octal system.

Ans a) 1. $3 - 6$

$C = 8^1 - 1 - 6 = 1$

2. $3 + 1 = 4$

i.e no carry.

3. a) complement of 4

$C = 8^1 - 1 - 4 = 3$

b) Put -ve sign

Ans = -3

b)

$30 - 16$

1. $C = 8^2 - 1 - 16 = 64 - 17 = 47$

2.
$$\begin{array}{r} 30 \\ +\ 47 \\ \hline 77 \end{array}$$

i.e no carry

3. a) complement of 77

$C = 8^2 - 1 - 77 = 64 - 78 = -14$

3 b) Put -ve sign i.e $-(-14) = (14)_8$

DATE: __/__/____
PAGE ______

Q. Subtract $(011)_2$ from $(101)_2$

Ans. 101 − 011

Step 1 Complement of $(011)_2$ = $(100)_2$

→ C can be directly found by converting 0 to 1 and vica versa

Step 2
$$\begin{array}{r} 101 \\ +\,100 \\ \hline 1\,001 \\ +1 \end{array}$$

Step 3 010 → Ans.

Verify :- 101 − 011 = 010
5 − 3 = 2

Q. Subtract $(101)_2$ from $(011)_2$
Ans. $(011)_2$ − $(101)_2$

1. Complement of $(101)_2$ = $(010)_2$

2.
$$\begin{array}{r} 011 \\ +\,010 \\ \hline 101 \end{array}$$

ie. no carry

3 a) Complent of $(101)_2$ → $(010)_2$

b) Ans. $(-010)_2$

GOOD WRITE

Verify :- 011 - 101 = -010
3 - 5 = -2

Q.

a) $(1011100)_2 - (0111000)_2$
b) $(010010)_2 - (100011)_2$

a) $(1011100)_2 - (0111000)_2$

1. complement of 0111000 is : $(1000111)_2$

```
   1 1 1
2.  1 0 1 1 1 0 0
  + 1 0 0 0 1 1 1
  ---------------
  1 0 1 0 0 0 1 1
               + 1
  ---------------
3.  0 1 0 0 1 0 0
```

Ans. $(0100100)_2$

verify : 1011100 - 0111000 = 0100100
92 - 56 = 36

b) $(010010)_2 - (100011)_2$

1. Complement of 100011 is : $(011100)_2$

```
    1
2.  0 1 0 0 1 0
  + 0 1 1 1 0 0
  -------------
    1 0 1 1 1 0
```

i.e no carry

3a. Complement of 101110 is : $(010001)_2$

3b) Put -ve sign : $(-010001)_2$ Ans.

verify : 010010 - 100011 = -010001
18 - 35 = -17

Computer Codes / Character Coding Schemes

Computer codes are used to represent data in memory or internal representation of data.

Data means,
numeric data → 0 to 9 = 10 values
Character = Uppercase & lowercase alphabets.
(26+26) = 52 values/alphabets.

Alphanumerical = Number + Alphabets + Special Characters

⇒ Computer codes using binary coding scheme because computer only understands binary digits i.e 0 & 1.

Computer codes are :-

- BCD (Binary Coded Decimal)
- EBCDIC (Extended Binary Coded Decimal Interchange Code)
- ASCII (American Standard Code for Information Interchange)
- Grey Code
- ~~Gray~~ WRITE CODE

BCD or 8421 Code

→ It uses 4-bits/6-bits to represent a symbol.

→ For 4-bits, total combinations are $2^4 = 16$ symbols.

→ For 6-bits, total combinations are $2^6 = 64$ symbols

→ Basically used to represent decimal values i.e 0 to 9

→ This is one of the early computer coding scheme.

→ Also called weighted code

Table → Representation of BCD using 4 bits

Decimal	Equivalent BCD using 4-bits
0	0 0 0 0
1	0 0 0 1
2	0 0 1 0
3	0 0 1 1
4	0 1 0 0
5	0 1 0 1
6	0 1 1 0
7	0 1 1 1
8	1 0 0 0
9	1 0 0 1

GOOD WRITE

DATE: __/__/____
PAGE ______

eg:- $(25)_{10} = (0010\ 0101)_{BCD}$

$(10)_{10} = (00010000)_{BCD}$

$(625)_{10} = (0110\ 0010\ 0101)_{BCD}$

BCD using 6-bits

Refer ch-5
(PK Sinha)
→ computer code

~~Decimal~~	BCD code 6-bits ZONE (2bits)	BCD (4 bits)
0	00	0000
1	00	0001
2	00	0010
3	00	0011
4	00	0100
5	00	0101
6	00	0110
7	00	0111
8	00	1000
9	00	1001
A-I	11	0001
	11	0010
	"	⋮
	11	1001
J-R	10	0001
	"	⋮
	10	1001
S-Z	01	0001
	"	⋮
	01	1001

GOOD WRITE

DATE: __/__/____
PAGE ______

Q. Represent 'HELLO' in BCD Code.

Ans

H	E	L	L	O
111000	110101	100011	100011	100110

For Octal conversion

$2^3 = 8$ ~~bits~~

So we will make pairs of 3.

H	E	L	L	O
$\underbrace{111}_{7}\,\underbrace{000}_{0}$	$\underbrace{110}_{6}\,\underbrace{101}_{5}$	$\underbrace{100}\,\underbrace{011}$	100011	100110
$(70)_8$	$(65)_8$			

EBCDIC

→ Extended Binary Coded Decimal Interchange Code

→ Uses 8-bits to represent a character | symbol in the data.

→ 2^8 combination i.e $2^8 = 256$ characters | symbols which include decimal digit (0-9), lower case (a-z), uppercase (A to Z), special characters some non-printable characters, etc.

→ Mostly used in mainframe computers.

EBCDIC code

ZONE (4 bits)	Digit (4 bits).

← 8 bits →

→ Fig 4.3 Pg 41 (P.K. Sinha)

Eg:-

EBCDIC	character	Hexadecimal Equivalent
1100 0001	A	C1
1100 0010	B	C2

Gray Code (Not in P.K. Sinha)

→ It is non-weighted code.

→ Not suitable for arithmatic operation but used in analog to digital conversion.

A. Conversion from Binary no. to gray code

$(\text{Given})_2 = (\ ?\ )_{\text{Gray code}}$

Step 1 → The first bit (MSB) of the gray code is same as the first bit of the binary number.

Step 2 → The second bit (next to MSB) of the gray code is equal to the EX-OR (exclusive OR) of second and first bit of the binary number.

Step 3 The third bit of the gray code is the EX-OR of the second and third bit of the binary code and so on.

A	B	$A \oplus B = AB' + BA'$ (EX-OR)
0	0	$0 \cdot 1 + 0 \cdot 1 = 0$
0	1	1
1	0	1
1	1	0

Eg: $(0100)_2 = (\ ?\ )$ Gray Code

Binary	Gray Code
0100	0 1 1 0

∴ **Ans.** (0110) Gray Code

Q. Convert $(1001)_2 = (\ ?\ )$ Gray code

Binary	Gray Code
1001	1 1 0 1

Ans. (1101) Gray Code

Binary	Gray Code
0 0 0 0	0 0 0 0
0 0 0 1	0 0 0 1
0 0 1 0	0 0 1 1
0 0 1 1	0 0 1 0
0 1 0 0	0 1 1 0
0 1 0 1	0 1 1 1
0 1 1 0	0 1 0 1
0 1 1 1	0 1 0 0
1 0 0 0	1 1 0 0
1 0 0 1	1 1 0 1
1 0 1 0	1 1 1 1
1 0 1 1	1 1 1 0
1 1 0 0	1 0 1 0
1 1 0 1	1 0 1 1
1 1 1 0	1 0 0 1
1 1 1 1	1 0 0 0

B. Conversion from Gray code to Binary

$$(\text{Given})_{\text{Gray code}} = (\ ?\ )_2$$

Step 1 The first binary bit (MSB) is same as the first bit (MSB) of the gray code.

Step 2 a) If the second bit of the gray code is 0, then the second bit of the binary ~~to~~ will be same as that of first ~~Gray~~ binary bit.

DATE: __/__/____
PAGE ______

Step 2 b) If the second bit of the gray code is 1 then the second binary bit will be the inverse / complement of the first ~~gray~~ binary code.

Step 3 → Step 2 is repeated for each succeeding bit.

Eg: ① $(0010)_{\text{Gray Code}} = (?)_2$

$(0010)_{\text{Gray Code}} = (\underline{0}\ \underline{0}\ \underline{1}\ \underline{1})_2$

② $(1010)_{\text{Gray Code}} = (?)_2$

Ans. $(\underline{1}\ \underline{1}\ \underline{1}\ \underline{1})_2$

③ $(1001)_{\text{Gray Code}} = (?)_2$

$(1001)_{\text{Gray code}} = (\underline{1}\ \underline{1}\ \underline{0}\ \underline{1})_2$

COMPUTER CODES

① ASCII character coding Scheme

→ American Standard Code for Information Interchange
→ Popular and widely used in all types of computer systems.

GOOD WRITE

Scanned by CamScanner

→ Used for data communication networks.

→ It is of 2 types.

2 types

→ ASCII-7

→ ASCII-8

ASCII-7

b_6 b_5 b_4 (3 bits)	b_3 b_2 b_1 b_0
ZONE	4 bit Digit

← 7 bits →

Possible combinations $= 2^7$

$= 128$ characters

0	000	0000
⋮	⋮	⋮
127	111	1111

ASCII-8

b_7 b_6 b_5 b_4 (4 bits)	b_3 b_2 b_1 b_0 (4 bits)
ZONE	Digit

← 8 bits →

Possible combinations $= 2^8$

$= 256$ characters

0	0000	0000
⋮	⋮	⋮
255	1111	1111

ASCII characters → Pg 44 Fig 4.6 PK Sinha

(2) UNICODE

→ Universal character coding

→ Used for multiple languages like Hindi, Chineese, Japanese, etc

→ Main advantage is that it is compatible with ASCII-8 code

→ It uses 32-bits to represent the data.
→ First 256 characters are taken for ASCII-8 code
→ Total possible combinations $= 2^{32}$

Represent A in unicode

0 .... 00 0 1 0 0 0 0 0 1

31 ———————————— 0

$A = (65)_{10}$

↳ convert in binary

0 1 0 0 0 0 0 1

3-Encoding formats for Unicode

1. UTF-8
(Unicode Transformation Format-8)
2. UTF-16
(Unicode Transformation Format-16)
3. UTF-32
(Unicode Transformation Format-32)

UTF-8

1B	1B	1B	1B

← 32 bits →

Minimum → 1 Byte
Max → 4 Bytes

→ Also called Byte-formatted

UTF-16

1 word	1 word

← 32 bits →

16 bits = 1 word

⇒ Also called word formatted

UTF-32

1D

← 32 bits →

32 bits = 1 double

⇒ Also called Double-word formatted

Q. Do the following binary arithmatic addition and subtraction of binary real numbers.

(a) 110100.101 + 10011.10

```
  1 1 1 1 1
   110100.101
+   10011.100
-------------
  1001000.001
```

(b) 10100.101 − 10011.10

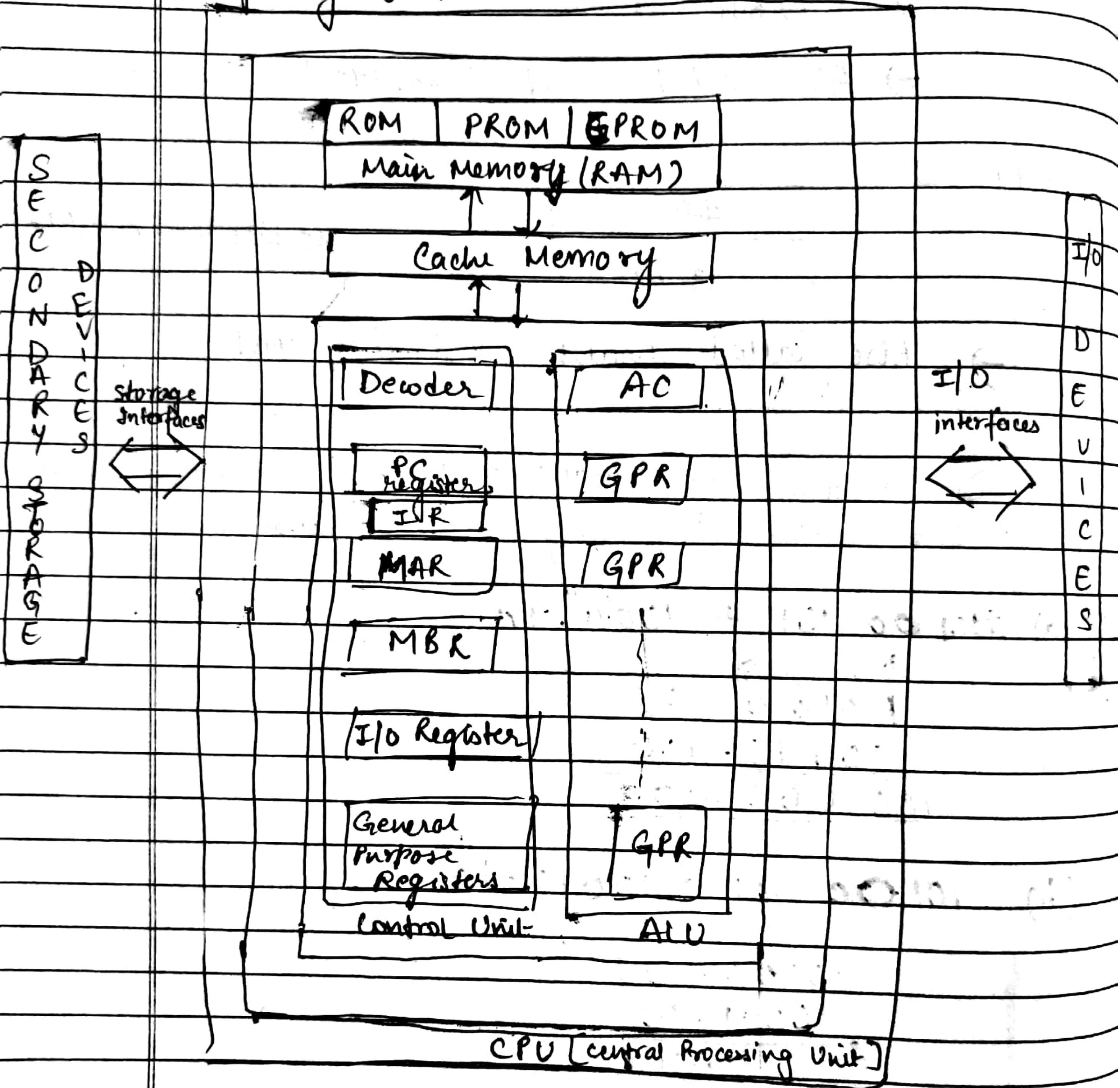
```
   10100.101
 − 10011.100
------------
   00001.001
```

(c) 11100.110 + 1110.10

(d) 11100.110 − 1110.10

PROCESS AND MEMORY ARCHITECTURE OF A COMP. SYSTEM

Refer Pg 105 PK Sinha



where PC → Program control Register
IR → Instruction Register
MAR → Memory Address Register
MBR → Memory buffer register

DATE: __/__/____
PAGE ______

I/O Register → Input / Output Register
AC → Accumulator Register
GPR → General Purpose Register

Computer Memory

↳ storage unit of the computer system

→ A memory unit is a collection of storage cells together with associated circuits need to transfer data / information / program in and out of storage.

→ Basic storage element is called "Bit"

BINARY DIGIT = BIT

clock cycle on / enable → $\boxed{1}$ Flip-Flop
or
clock cycle off / disable → $\boxed{0}$

1 Flip-flop takes 1 bit
So to represent sybols and words, we use registers.

b7						b1	b0

Register →

8 bits register
or
word

1 Byte = 8 bits

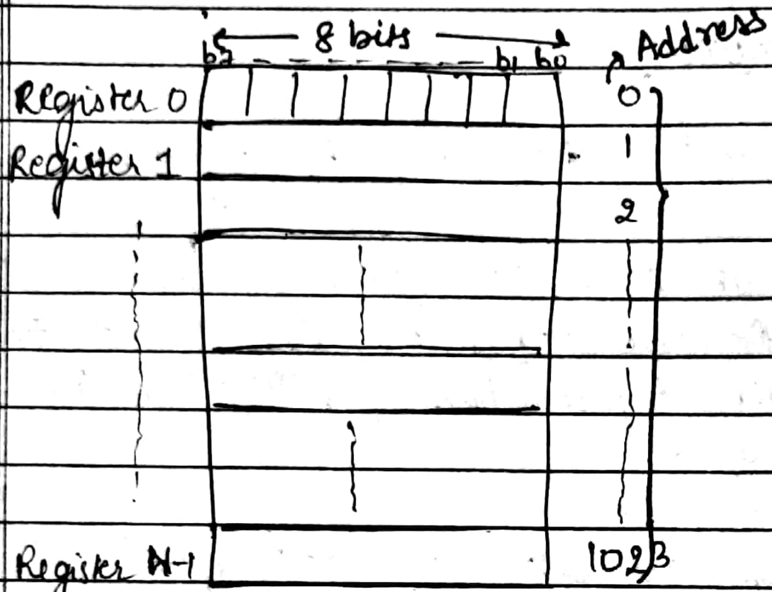
Two ways of accessing memory.
→ Read From
→ Write Into

GOOD WRITE

Scanned by CamScanner

Main Memory Organisation

↳ Identifies the structure that determines how data is accessed.

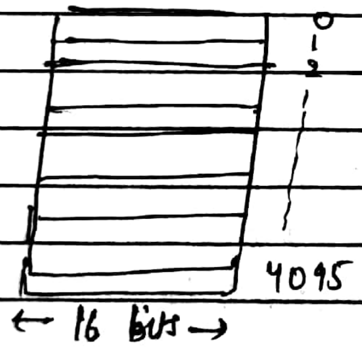


Main Memory

Suppose 1024 registers

Total size = 1024 × 8
= 8192 bits

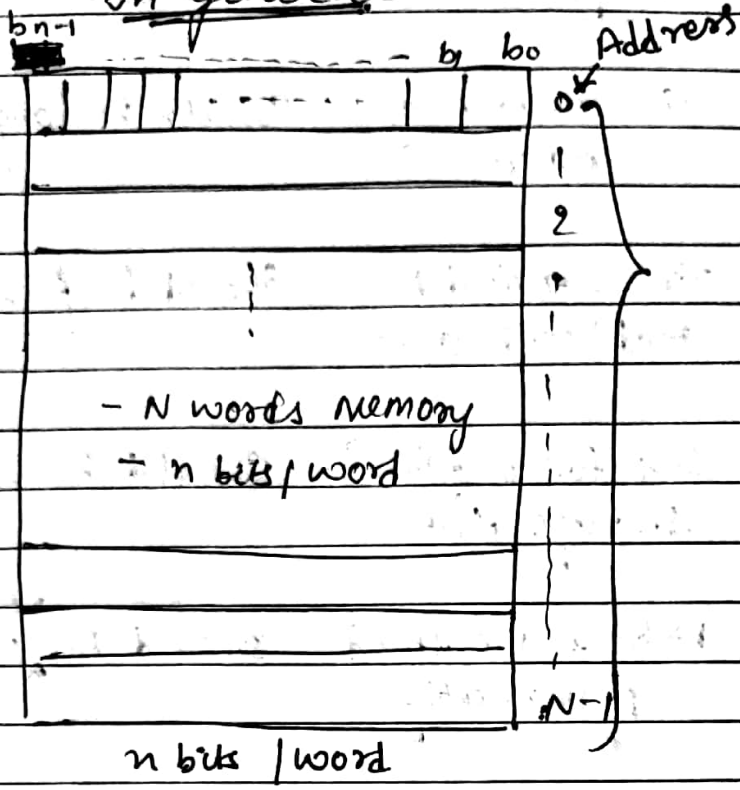
Eg:-



Total size = 4096 × 16
= ____ bits

NOTE: 1 B = 8 bits
1 KB = 1024 B = 2^{10} B
1 MB = 1024 × 1024 B = 2^{20} B
1 GB = 1024 × 1024 × 1024 B = 2^{30} B
1 TB = 2^{40} B

In general



Total N words
where $N = 2^k$
K = address bits

Eg:

← 8 bits →

3 flip flops

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

$8 = 2^{3} \rightarrow K$

first address → 0

last address → 7

Storage Evaluation criteria

→ Any storage unit can be evaluated using the following criteria:

	Criteria	Purpose	Desirable
1.	Storage Capacity	Total amount of data	Should be large
2.	Access time	Time required to read / write data	Should be fast
3.	Cost per ~~unit~~ bit storage	Cost for a given storage unit	Should be lower
4.	Volatility	Data is not stored permanently	Should be non volatile.
5.	Access Mechanism	2 types of access mechanism: (1) Random Access (2) Sequential Access	~~Should be~~ Random access should be desirable.

Primary Memory (Main Memory)

Characterstics
- made up of semiconductor devices
- very expensive
- volatile in nature
- Speed is very fast
- computer system can't run without primary memory.

Q. Computer system cannot run without primary memory. Why?

Ans.

V.V. Imp.

Primary Memory :

- RAM (Random Access Memory)
  - SRAM (Static RAM)
    - → It is used as cache memory due to the fast speed.
    - → Access speed = ~~10~~ 2 to 10 nsec
  - DRAM (Dynamic RAM)
    - → Main Memory
    - → Access speed = 50 - 100 nsec
- ROM (Read Only Memory)
  - Manufactured ROM
    - a) MROM
  - User programmed ROM
    - PROM
    - EPROM
      - UVEPROM (Ultra violet EPROM)
      - EEPROM (Electrical EPROM)

Q. Difference b/w SRAM and DRAM?
Q. Difference b/w PROM and EPROM?
Q. Difference b/w primary and secondary memory?

* BIOS Program stored in ROM (Bootable Program)
↳ Basic Input Output System

Memory Hierarchy

Access Time	Cost	Capacity
high	high	low
low	low	high

Registers

Cache Memory

Primary / Main Memory

Semi-conductor memory or Primary Storage

Magnetic Disk / Optical Disk

Magnetic Tape

Secondary Storage

REVISION FIT

→ Do the following

1. Write BCD equivalent of the following decimals:-
   a) $(125)_{10}$
   b) $(256)_{10}$
   c) $(45)_{10}$

2. Convert $(111011100)_{\text{Gray code}}$ to $(\quad)_2$

3. Add 10111 + 11000 + 111
4. 1100 − 1010

5. Represent $(-20)_{10}$ as signed magnitude, 1's complement and 2's complement form.

6. $(94.76)_{10} = (\quad)_2, (\quad)_8, (\quad)_{16}$

7. Five basic functions of computer system.
8. Five characteostics of computer system

9. $(234.334)_{10} = (\quad)_2, (\quad)_8, (\quad)_{16}$

Answers.

Ans 7
1. Inputting
2. Storing
3. Processing
4. Controlling
5. Outputting

DATE: __/__/____
PAGE ________

Ans 2 $(11101100)_{Gray\ code} = (\cancel{10011010})_2$

$= (10110111)_2$

Ans 1

① $(125)_{10}$

$= (0001\ 0010\ 0101)_{BCD}$

② $(256)_{10}$

$= (0010\ 0101\ 0110)_{BCD}$

③ $(45)_{10}$

$= (0100\ 0101)_{BCD}$

0 0 0 0	→ 0
0 0 0 1	→ 1
0 0 1 0	→ 2
0 0 1 1	→ 3
0 1 0 0	→ 4
0 1 0 1	→ 5
0 1 1 0	→ 6
0 1 1 1	→ 7
1 0 0 0	→ 8
1 0 0 1	→ 9

Ans 3

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

Ans 4

```
 1100 → 12
-1010 → 10
 0010 → 2
```

Now subtract using complement method.

complement of 1010 → 0101

Now add 0101 + 1100 = 0010

```
  1
  0101
+ 1100
 10001
  ↳ +1
  0010
```

GOOD WRITE

Ans 5. $(-20)_{10}$

Signed Magnitude

1	0	0	1	0	1	0	0

signed bit

1's complement

Step 1 : Convert $(+20)_{10}$ two binary form

$(+20)_{10} \rightarrow \boxed{00010100}$

1's complement $\boxed{111\ 01011}$

2's Complement

$(+20)_{10} \rightarrow \boxed{00010100}$

1's complement $\rightarrow \boxed{11101011}$

$+1$

Add 1 to LSB $\rightarrow \boxed{11101100}$

Ans 6 $(94.76)_{10}$

To Binary.

2	94	0
2	47	1
2	23	1
2	11	1
2	5	1
2	2	0
2	1	1
	0	

		Integer
0.76 × 2	= 1.52	1
0.52 × 2	= 1.04	1
0.04 × 2	= 0.08	0
0.08 × 2	= 0.16	0
0.16 × 2	= 0.32	0
0.32 × 2	= 0.64	0
0.64 × 2	= 1.28	1

Ans. $(1011110.1100001)_2$

To Octal.

6	0.76 × 8 = 6.08
0	0.08 × 8 = 0.64
5	0.64 × 8 = 5.12
0	0.12 × 8 = 0.96
7	0.96 × 8 = 7.68
5	0.68 × 8 = 5.44

8	94	6
8	11	3
8	1	1
	0	

Ans. $(136.605075)_8$

To hexadecimal.

12 = C	0.76 × 16 = 12.16
2	0.16 × 16 = 2.56
8	0.56 × 16 = 8.96
15 = F	0.96 × 16 = 15.36
5	0.36 × 16 = 5.76
12 = C	0.76 × 16 = 12.16

16	94	14 = E
16	5	5
	0	

Ans. $(5E.C28F5C)_{16}$

Ans 8 ① Speed
② Accuracy
③ Versatile
④ NO IQ
⑤ Diligence

Ans 9 $(234.334)_{10}$

To Binary.

0	$0.334 \times 2 = 0.668$
1	$0.668 \times 2 = 1.336$
0	$0.336 \times 2 = 0.672$
1	$0.672 \times 2 = 1.344$
0	$0.344 \times 2 = 0.688$
1	$0.688 \times 2 = 1.376$

2	234	0
2	117	1
2	58	0
2	29	1
2	14	0
2	7	1
2	3	1
2	1	1
	0	

Ans. $(11101010.010101)_2$

To Octal.

8	234	2
8	29	5
8	3	3
	0	

2	$0.334 \times 8 = 2.672$
5	$0.672 \times 8 = 5.376$
3	$0.376 \times 8 = 3.008$
0	$0.008 \times 8 = 0.064$
0	$0.064 \times 8 = 0.512$

Ans. $(352.25300)_8$

DATE: __/__/__
PAGE ____

To hexadecimal

16	234
16	14
	0

10 = A
14 = E

5 $0.334 \times 16 = 5.344$
5 $0.344 \times 16 = 5.504$
8 $0.504 \times 16 = 8.064$
1 $0.064 \times 16 = 1.024$
0 $0.024 \times 16 = 0.384$

Ans. $(EA.55810)_{16}$

GOOD WRITE