# Digital Logic

## Unit 1: Binary Systems

### 1.1 Digital System

System :- consisting of input, process, output

→ System is the combination of input, process & output. It takes input from environment, processes the information and provides output in desired form.

→ [Input] → [Process] → [Output] →

Fig :- Block diagram of system

→ Systems can be broadly classified into two types
i) Digital System
ii) Analog System

(*) Digital System :-

→ Digital System takes discrete information, processes it and provides output in digital form.
→ It works with digital signals consisting of two binary values, viz. 0 and 1.

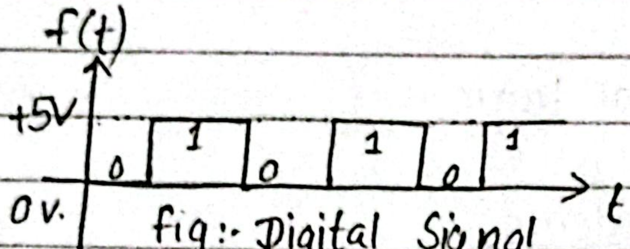
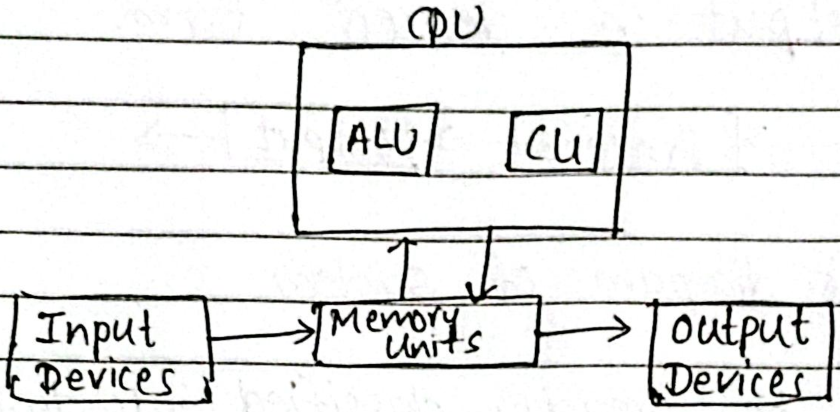


fig:- Digital Signal

⇒ Examples of digital system are:
- digital Computers
- Digital Watch
- Digital Thermometer

# Block Diagram of Digital Computer




Register:- CPU's direct memory
Cache:- Intermediate between MU and CPU.

In digital computers, input devices provide information to memory units such as RAM. Memory Unit stores data and provides to processing unit when required. Processing unit, consisting of ALU and CU, performs arithmetic and logical operations as per the instructions from control unit. Processing Unit stores final output in memory unit and output devices read the information from memory unit and make it available to user in desired format.

## * Analog System

→ Analog System works with continuous signal, processes it and provides output to the user as continuous wave forms or signals.

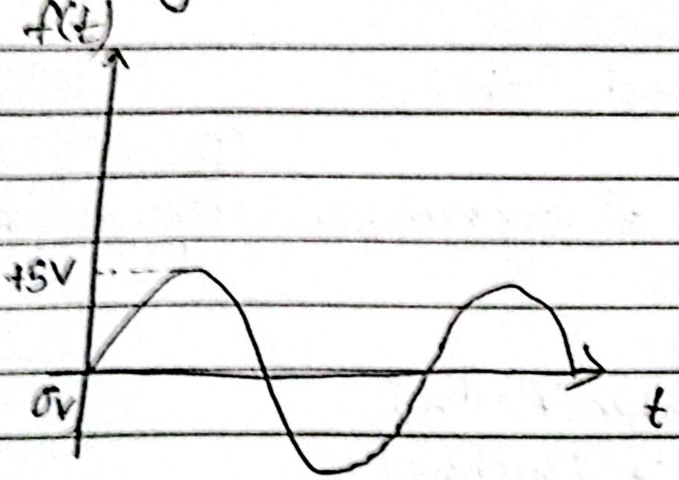



Fig:- Analog Signal

→ Examples of analog signals are:
Speakers, analog thermometers, speedometers.

* Advantages of digital systems over analog systems:-

i) Processing of information is easier and efficient
ii) Digital systems are more immune to noise.
iii) Digital signals can be easily stored and transmitted from one place to another.

Date ____
Page ____

* Disadvantages of digital systems over analog:-
i) Digital systems are more costly and complex.
ii) In digital systems, all components need to work on same timing (synchronized) otherwise even a single bit delay can change the meaning of information.

# Number systems

* Numbers are used for counting.
$(Base)^n - 1$ ← number of digits.
* The count (value) of any number depends on three factors:-
i) Digits used
ii) Base of number (Radix)
iii) Position of Digit (Weight)

In general, a number 'N' with base 'r' has ~~value~~ written as

$$(N)_r = A_{n-1} A_{n-2} A_{n-3} \cdots A_0 \bullet A_{-1} A_{-2} A_{-3} \cdots A_{-m}$$

Here, $A_{n-1}, A_{n-2}, \cdots A_0$ are n digits in integer part
$A_{-1}, A_{-2}, \cdots A_{-m}$ are m digits in decimal part
• is radix point.

$A_{n-1}$ = Most Significant Digit (MSD)
$A_{-m}$ = Least Significant Digit (LSD)

Value of $(N)_r$ is given as

$$(N)_r = r^{n-1} \times A_{n-1} + r^{n-2} \times A_{n-2} + \cdots + r^0 \times A_0 + r^{-1} \times A_{-1} + \cdots + r^{-m} A_{-m}$$

Find the value of (1243.56) considering it as decimal, octal and hex number.

As decimal,

$$(1243.56)_{10} = 1 \times 10^3 + 2 \times 10^2 + 4 \times 10^1 + 3 \times 10^0 + 5 \times 10^{-1} + 6 \times 10^{-2}$$

$$= 1000 + 2 \times 100 + 4 \times 10 + 3 \times 1 + \frac{5}{10} + \frac{6}{100}$$

$$= 1000 + 200 + 40 + 3 + 0.5 + 0.06$$

$$= (1243.56)_{10}$$

As for $(1243.56)_8$,

$$(1243.56)_8 = 8^3 \times 1 + 8^2 \times 2 + 8^1 \times 4 + 8^0 \times 3 + 8^{-1} \times 5 + 8^{-2} \times 6$$

$$= 512 + 64 \times 2 + 8 \times 4 + 3 + \frac{5}{8} + \frac{6}{64}$$

$$= (675.71875)_{10}$$

For $(1243.56)_H$

$$(1243.56)_H = 16^3 \times 1 + 16^2 \times 2 + 16^1 \times 4 + 16^0 \times 3 + 16^{-1} \times 5 + 16^{-2} \times 6$$

$$= (4675.3359)_{10}$$

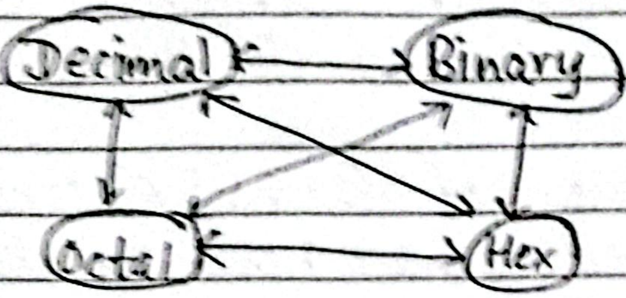
Types of number system

On the basis of different bases, commonly used number systems,

| Number system | Base (r) | Digits |
|---|---|---|
| Decimal | 10 | 0-9 |
| Binary | 2 | 0, 1 (Bits) |
| Octal | 8 (O) | 0-7 |
| Hexadecimal | 16 (H) | 0-9, A-F |

## Number System Conversion.

#




### (i) From Any base to decimal

# Find the equivalent using expansion.

i) $(ADD.BABA)_H$
ii) $(101101.10)_8$
iii) $(11100.011)_2$

i) $(ADD.BABA)_H = 16^2 \times 10 + 16^1 \times 13 + 16^0 \times 13 + 11 \times 16^{-1} + 16^{-2} \times 10$
$+ 16^{-3} \times 11 + 16^{-4} \times 10$
$= (2781.729901)_{10}$

ii) $(101101.10)_8 = 1 \times 8^5 + 0 \times 8^4 + 1 \times 8^3 + 1 \times 8^2 + 0 \times 8^1 + 1 \times 8^0 + 1 \times 8^{-1} + 0 \times 8^{-2}$
$= (33345.125)_{10}$

iii) $(11100.011)_2 = 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 + 0 \times 2^{-1}$
$+ 1 \times 2^{-2} + 1 \times 2^{-3}$
$= (28.375)_{10}$

From decimal to any base :

For integer part,

# Repeatedly divide integer part by r unless quotient is zero and consider last remainder as most significant digit.

For fractional part,

# Repeatedly multiply fractional part by 'r' unless result is complete integer or till 4 digits and consider last remainder as LSD.

Example:- Convert $(4321.8765)_{10}$ to binary, octal and hex.

$(4321.8765)_{10} \longrightarrow (?)_2$

For integer part,

| | | |
|---|---|---|
| 2 | 4321 | 1 |
| 2 | 2160 | 0 |
| 2 | 1080 | 0 |
| 2 | 540 | 0 |
| 2 | 270 | 0 |
| 2 | 135 | 1 |
| 2 | 67 | 1 |
| 2 | 33 | 1 |
| 2 | 16 | 0 |
| 2 | 8 | 0 |
| 2 | 4 | 0 |
| 2 | 2 | 0 |
| | 1 | |

For decimal part.

| | | |
|---|---|---|
| 2 | 8765 | 1 |
| 2 | 4382 | 0 |
| 2 | 2191 | 1 |
| 2 | 1095 | 1 |
| 2 | 547 | 1 |
| 2 | 273 | 1 |
| 2 | 136 | 0 |
| 2 | 68 | 0 |
| 2 | 34 | 0 |
| 2 | 17 | 1 |
| 2 | 8 | 0 |
| 2 | 4 | 0 |
| 2 | 2 | 0 |
| | 1 | |

~~$(4321.8765)_{10} = 1000011100001.1011$~~

- ~~(4321.8765~~

| | Integr |
|---|---|
| ~~$0.8765 \times 8 = 7.012$~~ | ~~7~~ |
| ~~$0.012 \times 8 = 6.096$~~ | ~~0~~ |
| ~~$0.096 \times 8 = 0.768$~~ | ~~0~~ |
| $0.$~~$768 \times 8 = 6.144$~~ | ~~6~~ |

| | Integer |
|---|---|
| $0.765 \times 2 = 1.753$ | 1 |
| $0.753 \times 2 = 1.506$ | 1 |
| $0.506 \times 2 = 1.012$ | 1 |
| $0.012 \times 2 = 0.024$ | 0 |

$(4321.8765)_{10} = (1000011100001.10110)_2$

- $(4321.8765) \rightarrow (?)_8$

For integer part,

| | | |
|---|---|---|
| 8 | 4321 | 1 |
| 8 | 540 | 4 |
| 8 | 67 | 3 |
| 8 | 8 | 0 |
| 8 | 1 | 1 |
| | 0 | |

For decimal part,

| | Integer |
|---|---|
| $0.8765 \times 8 = 7.012$ | 7 |
| $0.012 \times 8 = 0.096$ | 0 |
| $0.096 \times 8 = 0.768$ | 0 |
| $0.768 \times 8 = 6.144$ | 6 |

$(4321.8765)_{10} = (10341.7006)_8$

- $(4321.8765)_{10} \rightarrow (?)_{16}$

For integer part,

| | | |
|---|---|---|
| 16 | 4321 | 1 |
| 16 | 270 | E |
| 16 | 16 | 0 |
| 16 | 1 | 1 |
| | 0 | |

For fractional part,

| | Integer |
|---|---|
| $0.8765 \times 16 = 14.024$ | E |
| $0.024 \times 16 = 0.384$ | 0 |
| $0.384 \times 16 = 6.144$ | 6 |
| $0.144 \times 16 = 2.304$ | 2 |

$\therefore (4321.8765)_{10} = (10E1.E062)_{16}$

Assignment :- Write from 0 to 15 in decimal, binary, octal & Hex

| Decimal | Binary | Octal | Hex |
|---|---|---|---|
| 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 |
| 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 |

Date
Page

iii) From binary to octal and octal to binary.

For binary to octal:-
i) Group the given binary in 3 bits starting from left of binary point for integer part and right of binary point for fractional part

Note:- You can add 0's in front of MSB and back of LSB to make group of 3 bit.

ii) Write equivalent octal value of each 3 bit group.
eg:- 2 for 010, 5 for 101 & 7 for 111

For octal to binary:-
⇒ Replace each octal digit with 3 bit binary equivalent.

Example:-
convert (i) $(10110.01101)_2$ to octal.

→ Given binary number = $(10110.01101)_2$
Grouping into 3 bits.

| | | | | |
|---|---|---|---|---|
| | 010 | 110 . | 011 | 010 |
| Octal equivalent | 2 | 6 | 3 | 2 |

∴ $(10110.01101)_2 = (26.32)_8$

$(76.23)_8$ to binary

76.23 Octal number

111 110 010 011 Binary equivalent

$(76.23)_8 = (111110.010011)_2$

from binary to hex and hex to binary.

For binary to hex :-
i) Group the given binary in 4bits starting from left of binary point for ~~b~~ integer part and from right of binary point for fractional part.

Note:- You can 0's infront of MSB and back of LSB to make group of 4 bits.

ii) Write equivalent ~~octal~~ hex value of each 4 bit group. Eg 2 for 0010, 5 for 0101 & E for 1110

For hex to binary:-
⇒ Replace each hex digit with 4 bit binary equivalent

Example:- Convert:- i) $(1011010111.101)_2$ to Hex

Binary no:- $(1011010111.101)_2$

Group of 4 bits :- 0010 1101 0111 · 1010

Equivalent hex :- 2 D 7 A

$(1011010111.101)_2 = (2D7.A)_{16}$

ii) $(3AD.4BC)_H$ to Binary

Hex number :: 3AD·4BC

Equivalent binary bits 0011 1010 1101 · 0100 10111100

∴ $(3AD.4BC)_H = (1110101101.0100101111)_2$

(v) From octal to hex and hex to octal.

For octal to Hex

i) First convert octal to binary with 3 bit group and then make 4 bit group in binary number to find equivalent hex

For hex to octal

i) First convert hex to binary with 4 bit group and then make 3 bit group in binary number to find equivalent octal.

Example...

Convert:

(i) $(7DE.ABC)_H$ to octal

Given hex number :- 7DE . ABC

Binary equivalent ~~numbe~~ bits : 0111 1101 1110 . 1010 1011 1100

Group of 3 bits : 011 111 011 110 . 101 010 111 100

octal equivalent : 3 7 3 6 . 5 2 7 4

$\therefore (7DE.ABC)_H = (3736.5274)_8$

ii) $(444.666)_o$ to hex

Given octal number : $(444.666)_o$

Binary equivalent bits - 100100100 . 110110 110

Group of 4 bit = 0001 0010 0100 . 1101 1011 0000

Hex equivalent = 124 . DB0

$\therefore (444.666)_o = (124.DB)_H$

## 1.2 Complement of a Number

→ Complement is another form of a number used to simplify subtraction and represent negative numbers.

→ There are two types of complement. They are:

i) Radix Complement (r's complement):
For any number N with base r,
r's complement = $r^n - N$
where n = no of digits / bits in integer part.

ii) Diminished Radix complement ((r-1)'s)
For any given number, N with base r
(r-1)'s complement = $r^n - r^{-m} - N$
n is no. of digits in integer part.
m is no. of digits in fractional part.

Example:
i) Find 10's and 9's complement of $(4567.89)_{10}$

ii) Find 7's and 8's complement of $(423.56)_8$
iii) Find 2's and 1's complement of $(101101.011)_2$.

Date
Page

i) => Sol$^n$...

a) 10's complement of $(4567.89)_{10}$
$(4567.89)_{10}$

→ Sol$^n$..

$N = (4567.89)_{10}$
$n = 4$
$r = 10$

Then 10's (r's) complement $= r^n - N$
$= (10)^4 - (4567.89)_{10}$
$= 10000 - (4567.89)_{10}$
$= (5432.11)_{10}$

b) 9's complement of $(4567.89)_{10}$
$N = (4567.89)_{10}$
$n = 4$
$r = 10$ $m = 2$
$(r-1) = 9$

Then 9's (r-1's) complement $: 10^4 - 10^{-2} - (4567.89)_{10}$
$= (5432.1)_{10}$

ii) => Sol$^n$..

8's complement of $(423.56)_8$
Given,
$N = (423.56)_8$
$n = 3$
$r = 8$

Then 8's (r's) complement $= r^n - N$
$= 8^3 - (423.56)_8$
$= 850.22$

7's complement of $(423.56)_8$

Given,

$N = (423.56)_8$

$r = 8$

$n = 3$

$N = (423.56)_8$

$r - 1 = 7$

$m = 2$

The 7's (r-1's) complement $= r^n - r^{-m} - N$

$= (8^3 - 8^{-2})_{10} - (423.56)_8$

$= (511.98)_{10} - (423.56)_8$

$= (777.76)_8 - (423.56)_8$

$= (354.20)_8$

Here, Integer part

| | | |
|---|---|---|
| 8 | 511 | 7 |
| 8 | 63 | 7 |
| 8 | 7 | 7 |
| | 0 | |

For fractional part

$0.98 \times 8 = 7.84$ 7

$0.84 \times 8 = 6.72$ 6

2's complement of $(101101.011)_2$

$N = (101101.011)_2$

$n = 6$

$m = 3$

$r = 2$

Now,

2's complement (r's complement) $= r^n - N$

$= 2^6 - (101101.011)_2$

$= (64)_{10} - (101101.011)_2$

$= (1000000)_2 - (101101.011)_2$

$= (10010.101)_2$

$$\begin{array}{r} 1000000.000 \\ -\ 101101.011 \\ \hline 010010.101 \end{array}$$

1's complement of $(101101.011)_2$

$N = (101101.011)_2$

$n = 6$

$m = 3$

$r = 2$

$r - 1 = 1$

Now,

1's complement (r-1's complement) $= r^n - r^{-m} - N$

$= (2)^6 - (2)^{-3} - (101101.011)_2$

$= (2^6 - 2^{-3})_{10} - (101101.011)_2$

$= (2^6 - 2^{-3})_{10} - (10101.011)_2$

$= (63.875)_{10} - (10101.011)_2$

$= (111111.111)_2 - (101101.011)_2$

$= (10010.100)_2$

$$\begin{array}{r} 111111.111 \\ -\ 101101.011 \\ \hline 010010.100 \end{array}$$

Note:- shortcut for 1's complement

→ Invert all bits i.e replace '1' by '0' & '0' by '1'

Short cut for 2's complement
→ Add '1' in LSB of 1's complement
→ Starting from LSB, copy same bits till ~~to~~ first 1 and after that invert all bits.

Shortcut for 9's complement.
Subtract each digits from 9.

Shortcut for 10's complement.
Add 1 in LSD of 9's complement.

i) Subtraction using r's complement:

To subtract (M−N)
i) Add M with r's complement of N taking equal digits / bits

ii) If end carry occurs, discard it and remaining will be true result.

If end carry does not occur then answer is negative and is in r's complement form. True result is obtained by taking r's complement.

Example:.

i) If $A = (54)_{10}$ and $B = (35)_{10}$ then find $A-B$ and $B-A$ using a) 10's complement
b) 2's complement.

Subtraction using r's complement.

Soln..

Given, $A = (54)_{10}$
$B = (35)_{10}$

a) using 10's complement.

For $A-B$,

$A = 54$

10's complement of $B = \begin{array}{r} 99 \\ -35 \\ \hline 64 \\ +1 \\ \hline 65 \end{array}$

$A = 54$

10's complement of $B = 65$

$$\begin{array}{r} 54 \\ +65 \\ \hline \boxed{1}19 \end{array}$$

$\therefore 54 - 35 = (19)_{10}$

$\therefore A - B = (19)_{10}$

For $B-A$:

$B = 35$

10's complement of $A = \begin{array}{r} 99 \\ -54 \\ \hline 45 \\ +1 \\ \hline 46 \end{array}$

Date ______
Page ______

$$\begin{array}{r} 35 \\ +46 \\ \hline 81 \end{array}$$

Since there is no end carry, result is
- (10's complement of 81)
= -19

$\therefore (35 - 54) = -19$

b) 2's complement

For A - B
A = $(54)_{10}$ = $(110110)_2$
B = $(35)_{10}$ = $(100011)_2$
1's complement of B = $(011100)_2$

2's complement of B = 011101

For A - B

$$\begin{array}{rr} A = & 110110 \\ \text{2's comp of } B = + & 011101 \\ \hline & 1010011 \end{array}$$

$\therefore (A - B) = (010011)_2$
$= (19)_{10}$

For B - A
A = $(54)_{10}$ $(110110)_2$
B = $(35)_{10}$ = $(100011)_2$
2's complement of A = $(001010)_2$

Now,

$B-A$

$B = (100011)_2$

2's comp of $A = (001010)_2$

$101101$

Now

$(B-A) = -(010011)_2$

$= (-19)_{10}$

Eg:

If $A=(20)_{10}$ and $B=(10)_{10}$

then find $A-B$ and $B-A$

Using a) 10's complement

b) 2's complement

~~If A= 20~~

Using 10's complement

for $A-B$

$A=(20)_{10}$

$B=(10)_{10}$

10's complement of $B =$

$$\begin{array}{r} 99 \\ -10 \\ \hline 89 \\ +1 \\ \hline 90 \end{array}$$

$A = (20)_{10}$

10's comp of $B = (90)_{10}$

$\cancel{1}\,10$

$\therefore (20)_{10} - (10)_{10} = (10)_{10}$

Date ______
Page ______

For B - A,

$$\begin{array}{r} 99 \\ \text{10's complement of } A = -20 \\ \hline 79 \\ +1 \\ \hline 80 \end{array}$$

Now, $B = (10)_{10}$

10's complement of $A = +(80)_{10}$

$$90$$

10's complement of result $= 99$

$$\begin{array}{r} -90 \\ \hline 9 \\ +1 \\ \hline 10 \end{array}$$

$\therefore (10)_{10} - (20)_{10} = (-10)_{10}$.

b) 2's complement

A - B

$A = (20)_{10} = (10100)_2$

$B = (10)_{10} = (01010)_2$

For A - B,

1's complement of $B = (10101)_2$

2's complement of $B = (10110)_2$

Now, $A = (10100)_2$

2's comp of $B = (10110)_2$

$$\cancel{1}01010$$

$\therefore (A - B) = (01010)_2$

$= (10)_{10}$

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

Again, $A = (20)_{10} = (010100)_2$
$B = (10)_{10} = (01010)_2$

For $B - A$:

2's comp of A = 01100

$$\begin{array}{r} B = 01010 \\ \text{2's comp of } A = 01100 \\ \hline 10110 \end{array}$$

$B - A = -(01010)_2$
$= (-10)_{10}$

Subtraction using (r-1)'s complement.

For M - N

# Add mineuent (M) with (r-1)'s complement with N.

# Inspect for end carry.

# LSD of the result => If end carry occur, add it to the LSD of the result. The difference will be in true form

# If end carry does not occur then result is negative and is in (r-1)'s complement form.

Example:

1) If A = -20 and B = 10 then find A - B and B - A using 9's and 1's complement

$$\begin{array}{r} 99 \\ -20 \\ \hline 79 \end{array}$$

→ Soln.

Given,

$A = -20 = 79 = (01011)_2$

$B = 10 = (01010)_2$

Now,

For A-B

$$\begin{array}{r} 99 \\ -10 \\ \hline 89 \end{array}$$

Using 9's complement

$A = 79$

9's complement of $B = +89$

$$\begin{array}{r} \cancel{1}\,68 \\ +1 \\ \hline 69 \end{array}$$

Since there is no end carry, result is negative.

Then $A - B = -(9\text{'s complement of } 69)$

$= -(30)_{10}$

$$\begin{array}{r} 99 \\ -69 \\ \hline 30 \end{array}$$

1's Complement (A-B)

$A = -20 = (01011)_2$

$B = 10 = (01010)_2$

Now,

For A-B,

Using 1's complement,

$$\begin{array}{lr} & 1\;\;11\,1 \\ A = & (0\,1\,0\,1\,1)_2 \\ 1\text{'s complement of } B = & (1\,0\,1\,0\,1)_2 \\ \hline & \cancel{1}\,0\,0\,0\,0\,0 \\ & 1 \end{array}$$

Now,

$\therefore$ A - B = - (1's complement of 00001)

= - (11110)₂

= - (30)₁₀

9's complement (B - A)

A = -20 = 79 = (01011)₂

B = 10

For B - A

B = 10

9's complement of A = 99 - 79 = 20

Now,

B = 10

9's comp of A = 20

+

30

$$\begin{array}{r} 99 \\ -69 \\ \hline 30 \end{array}$$

$\therefore$ B - A = - (9's comp of 30)

= - (69)

= + 30

$$\begin{array}{r} 99 \\ -30 \\ \hline 69 \end{array}$$

B - A (1's complement)

A = -20 = (01011)₂

B = 10 = (01010)₂

1's complement of A = (10100)₂

B-A,

$$B = (01010)_2$$

1s comp of $A = (10100)_2$

$$(11110)_2$$

Now

$B-A = -(1\text{'s comp of } (11110)_2)$

$= -(00001)_2$

$= (11110)_2$

$= 30$

Eg:-

ii) If $A = -12$ and $B = -11$ then find A-B and B-A using 1's complement.

Given,

$A = (-12)_{10} = (0011)_2$

$B = (-11)_{10} = (0100)_2$

Now,

A-B

$A = (-12)_{10} = (0011)_2$

$\therefore$ ~~A~~ 1's complement of $B = (1011)_2$

$(1110)_2$

$\therefore A-B = -(1\text{'s complement of } (1110)_2)$

$= -(0001)_2$

B - A :-

$B = (0100)_2$

2's comp of A $= (1100)_2$

$\boxed{1}\,0000$

$\quad +1$

$0001$

$\therefore A - B = (0001)_2$

Representing Signed Numbers:-

Signed number means numbers with magnitude, i.e either positive or negative.

→ In binary system, signed numbers can be represented ~~to~~ using one of the following methods.

Sign Magnitude Form:-

Uses 8/16 bits representation where MSB is sign bit and remaining bit shows quantity.

If MSB = 1 ⇒ -ve number

If MSB = 0 ⇒ +ve number.

2's complement form:-

It also uses 8/16 bits representation where +ve numbers are represented by using true form and -ve numbers are represented using 2's complement of its true form.

Eg:- BCD of 1 = 0001
BCD of 9 = 1001

Invalid BCD codes :- 1010, 1011, 1100, 1101, 1110, 1111

b) Excess -3 code:-
It is a non-weighted code used to represent decimal digits by adding '0011' to given BCD codes.

Excess-3 code of 1 = 0100
Excess -3 code of 9 : 1100

Invalid codes:- 0000, 0001, 0010, 1101, 1110, 1111

| Decimal | Binary | BCD | Excess-3 |
|---|---|---|---|
| 0 | 0000 | 0000 | 0011 |
| 1 | 0001 | 0001 | 0100 |
| 2 | 0010 | 0010 | 0101 |
| 3 | 0011 | 0011 | 0110 |
| 4 | 0100 | 0100 | 0111 |
| 5 | 0101 | 0101 | 1000 |
| 6 | 0110 | 0110 | 1001 |
| 7 | 0111 | 0111 | 1010 |
| 8 | 1000 | 1000 | 1011 |
| 9 | 1001 | 1001 | 1100 |
| 10 | 001010 | 00010000 | 0100 0011 |
| 16 | 10000 | 00010110 | 0100 1001 |

Binary Codes:-
8421, 2421 and 84-2-1

complementing codes

| Decimal | 8421 | 2421 | 84-2-1 |
| --- | --- | --- | --- |
| 0 | 0000 | 0000 | 0000 |
| 1 | 0001 | 0001 | 0111 |
| 2 | 0010 | 1000 | 0110 |
| 3 | 0011 | 1001 | 0101 |
| 4 | 0100 | 0100 | 0100 |
| 5 | 0101 | 1011 | 1011 |
| 6 | 0110 | 0110 | 1010 |
| 7 | 0111 | 0111 | 1001 |
| 8 | 1000 | 1110 | 1000 |
| 9 | 1001 | 1111 | 1111 |

Gray code:-
- It is a non-weighted code and has refle properties.

| n=1 | n=2 | n=3 | n=4 | Decimal |
| --- | --- | --- | --- | --- |
| 0 | 00 | 000 | 0000 | 0 |
| 1 | 01 | 001 | 0001 | 1 |
| | 11 | 011 | 0011 | 2 |
| | 10 | 010 | 0010 | |
| | | 110 | 0110 | |
| | | 111 | 0111 | |
| | | 101 | 0101 | |
| | | 100 | 0100 | |
| | | | 1100 | |
| | | | 1101 | |
| | | | 1111 | |
| | | | 1110 | |
| | | | 1010 | |
| | | | 1011 | |
| | | | 1001 | |
| | | | 1000 | |

| n=1 | n=2 | n=3 | n=4 | Decimal |
|---|---|---|---|---|
| 0 | 00 | 000 | 0000 | 0 |
| 1 | 01 | 001 | 0001 | 1 |
| | 11 | 011 | 0011 | 2 |
| | 10 | 010 | 0010 | 3 |
| | | 110 | 0110 | 4 |
| | | 111 | 0111 | 5 |
| | | 101 | 0101 | 6 |
| | | 100 | 0100 | 7 |
| | | | 1100 | 8 |
| | | | 1101 | 9 |
| | | | 1111 | 10 |
| | | | 1110 | 11 |
| | | | 1010 | 12 |
| | | | 1011 | 13 |
| | | | 1001 | 14 |
| | | | 1000 | 15 |

→ Gray code can be obtained from binary values ~~from~~ as below:

Binary value : 1+0+1+1+0+1+0

Gray code : 1 1 1 0 1 1 1

→ We can also find equivalent binary value from gray code.

Gray code : 1 1 1 0 1 1 1

Binary code : 1 0 1 1 0 1 0

→ There is only one bit change in gray code of consecutive numbers.

# Find gray code of 1101110 and also find equivalent binary of obtained gray code

Binary code : 1 1 0 1 1 1 0
Gray code : 1 0 1 1 0 0 1

Gray code : 1011001
Binary code : 1101110

e) Error detecting codes

→ Error detecting codes are used to detect errors occured due to change of bits during transmission. Some of the examples of error detecting codes are:- parity bit, hamming code, check sum, etc.

i) Parity Bit:-

→ It is an additional bit added to the given binary number to make total number of 1's either even or odd. If we make total number of '1' odd, it is called odd parity and if you make total number of '1' even, it is called even parity.

| Message | Odd Parity bit | Even-parity bit |
|---|---|---|
| 010 | 0010 | 1010 |
| 011 | 1011 | 0011 |
| 101 | 1101 | 0101 |
| 000 | 1000 | 0000 |

Drawback

⇒ Even parity bit cannot detect even number of errors.

f) Alphanumeric Codes

→ Alphanumeric Codes are used to represent number, alphabets and symbols.
Two common binary alphanumeric codes are:-

i) ASCII Code (American Standard Code for Information Interchange):-

→ It uses 7 bit binary number from 0000000 1111111 (hex equivalent 00 to 7F) representing 128 different numbers/digits, symbols, alphabets.

→ It is still used in keyboards to communicate with processor. For example:- ASCII code for

$a = (61)_H = (1100001)_2 = (97)_{10}$

$A : (41)_H = (1000001)_2 = (65)_{10}$

i) EBCDIC Code (Extended Binary Coded Decimal Interchange Code):-

• In EBCDIC, 8 bit binary number from 00000000 to 11111111 (hex equivalent 00 to FF) are used which can represent 256 different alphabets number and symbols, enlarging the scope of ASCII code. For example,

$a = (81)_H = (129)_{10} = (10000001)_2$

$A = (61)_H = (97)_{10} = (01100001)_2$

BCD addition

In BCD addition, we convert each decimal digits to its equivalent BCD and add in a group of 4 bits using binary addition rule.

The sum of above additions will be invalid if the sum represents invalid BCD codes or generates end carry. In such cases, we convert such 4 bit invalid sum to valid sum by adding '0110' (6)

If carry is generated while adding '110', it is not considered invalid.

```
  8      1000
+ 1    + 0001
  9      1001 = 9 ✓

  7        0111
+ 9     + 1001
 16       10000
           0110
          10110
          1  6

  8      1000
+ 7    + 0111
 15      1111
         0110
        10101
        1  5
```

Add 853 and 256 using BCD addition.

| | | | | |
|---|---|---|---|---|
| 853 | = | 1000 | 0101 | 0011 |
| + 256 | = | 0010 | 0101 | 0110 |
| 1109 | | $1010^{+1}$ | 1010 | 1001 |
| | | 0110 | 0110 | |
| | 1 | 0001 | 0000 | 1001 |
| | 1 | 1 | 0 | 9 |

$\therefore 853 + 256 = 1109$