Digital Logic Design

UNIT- I

1. Digital Systems:-

A Digital system is a system in which signals have a finite number of discrete values.
Examples of discrete sets are 10 decimal digits, 26 letters etc.,

Signals:- Discrete elements of information are represented in a digital system by physical quantities called signals.

The signals in most present day electronic digital systems use just two discrete values and they are said to be binary.

A binary digit called a bit has two values either "0 or 1".

Discrete elements of information are represented with groups of bits called "Binary codes".

Thus a digital system is a system that manipulates discrete elements of information represented internally in binary form.

For ex.; the decimal digit 0 through 9 are represented in a digital system with a code of four bits. The number 7 is represented by 0111.

How a pattern of bits is interpreted as a number depends on the code system in which it resides.

↳ $(0111)_2$ to indicate the pattern is to be in binary system.

↳ $(0111)_{10}$ to indicate the pattern is to be in decimal system.

The users are allowed to specify and change the program or the data according to the specific need. Because of this flexibility,

General purpose digital computers can perform a variety of Information processing tasks that range over a wide spectrum of appln's.

Ex:- (1) A payroll schedule is an discrete process, which has discrete data values such as letters e

On the other hand,

(2) A Research scholar may observe a continuous process, but record only specific quantities in tabular form.

In many cases, the quantization of a process can be performed automatically by an Analog-to-digital converter.

Analog-to-Digital Converter:- A Device that forms a digital (discrete) representation of a Analog (continuos) quantity.

The general purpose digital computer is the best known ex., of digital system. The major parts of computer are memory unit, central processing unit, Input & Output.

A digital computer is a powerful instrument that can perform not only arithmetic computations, but also logical operations.

↳ **Advantages of Digital Systems:-**

(i) Ease of programmability.
(ii) Reduction in cost of hardware
(iii) High speed
(iv) High Reliability
(v) Design is easy
(vi) Results can be reproduced easily.

1.1 NUMBER SYSTEM:-

It defines a set of values used to rep., quantity. It is a language of digital system consisting of set of symbols called digits with rules defined for their addition, multipln and other mathematical operations. There a 2 types

(i) Positional Number system

(ii) Non positional Number system.

↳ **Positional Number System:-**

The position of each digit of a number has some positional weight. It is most widely used. Ex:- Decimal numbers.

↳ **Non - Positional:-** In this a digit of a number does not indicate any significance in position and weight. Ex:- Roman numericals. It is very difficult to use because zero is not present.

A number is constructed by a collection of digits and it has Integer and Fraction, both are seperated by a Radix point.

General format of a Number:-

$$N_r = a_{n-1} r^{n-1} + a_{n-2} r^{n-2} + \dots + a_1 r^1 + a_0 r^0 + a_{-1} r^{-1} + \dots + a_{-m} r^{-m}$$

a_{n-1}	a_{n-2}	$---a_1$	a_0	.	a_{-1}	a_{-2}	a_{-3} ---	a_{-m}

MSD — Integer Part — Radix point — Fraction part — LSD

$N_r \rightarrow$ Number with base r

$r \rightarrow$ radix or base

$a \rightarrow$ integer in the range $0 \leq a_i \leq (r-1)$.

Digital Logic Design.

1.2 Decimal Number System:-

Decimal number system has 10 Symbols
The Symbols are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.
So the radix or base of this number System is "10".
The position of each digit in a decimal number indicates with the magnitude of the quantity represented and can be assigned a weight.
Hence it is called Positional Weight Number System.

---	10^2	10^1	10^0	.	10^{-1}	10^{-2}	---

Decimal point.

In the decimal number system we can express any decimal number in units, tens, etc.,

Decimal Number:-

7392, it can be written as

$7\times10^3 + 3\times10^2 + 9\times10^1 + 2\times10^0$

The Thousands, hundreds etc., are powers of 10 implied by position of the co-efficients.

In general, A number with a decimal point is rep., by a series of coefficients as follows

$\boxed{a_5 a_4 a_3 a_2 a_1 a_0 . a_{-1} a_{-2} a_{-3}}$

The decimal number system is said to be of base or radix 10 because it uses 10 digit and the co-efficients are multiplied by powers of 10.

Ex:- $10^5 a_5 + 10^4 a_4 + 10^3 a_3 + 10^2 a_2 + 10^1 a_1 + 10^0 a_0 + 10^{-1} a_{-1} + 10^{-2} a_{-2} + 10^{-3} a_{-3}$.

2. BINARY NUMBERS :-

The Coefficients of binary number system have only two values "0 & 1".

bit (0 or 1)
↓
4 bits (Nibble)
↓
8 bits (Byte)
↓
16 bits (word)
↓
32 bit (Double word)

0 + 0 = 0
0 + 1 = 1
1 + 1 = 10

$$\begin{array}{r} 1\,1 \\ 1 \\ \hline 100 \end{array}$$

For ex:, The decimal equivalent of the binary number 11010.11 is 26.75

Multiplication of the co-eff by powers of 2.

$$1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} = 26.75$$

In general, a number expressed in a base - r system has coeff multiplied by powers of r.

$$a_n \cdot r^n + a_{n-1} \cdot r^{n-1} + \cdots + a_2 \cdot r^2 + a_1 \cdot r + a_0 + a_{-1} \cdot r^{-1} + a_{-2} \cdot r^{-2} + \cdots + a_{-m} \cdot r^{-m}$$

To distinguish between numbers of different bases we enclose the coefficients in parenthesis and write a subscript equal to the base used.

Ex:- Base - 5 number $(4021.2)_5$

$$= 4 \times 5^3 + 0 \times 5^2 + 2 \times 5^1 + 1 \times 5^0 + 2 \times 5^{-1} = (511.4)_{10}$$

The coeff values for base '5' can be only 0, 1, 2, 3 & 4.

3. OCTAL NUMBER:-

The Octal number system is a base - 8 system that has eight digits i.e., 0, 1, 2, 3, 4, 5, 6, 7.

Ex:- $(127.4)_8 = 1 \times 8^2 + 2 \times 8^1 + 7 \times 8^0 + 4 \times 8^{-1} = (87.5)_{10}$

Note:- The digits 8 & 9 cannot appear in an Octal number. By adding each digit of an octal number in a powers of 8 we can find the decimal equivalent of octal number.

4. HEXA DECIMAL NUMBER:-

It is a base 16 number system. The first ten digits are borrowed from the decimal system. i.e., 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. It is another number system that is particularly useful for human communications with a computer.

Ex:- $(B65F)_{16} = 11 \times 16^3 + 6 \times 16^2 + 5 \times 16^1 + 15 \times 16^0 = (46,687)_{10}$

The digits in a binary number are called bits. When a bit is equal to 0, it does not contribute to the sum during conversion.

Therefore the conversion from binary to decimal can be obtained by adding only the numbers with powers of two corresponding to the bits that are equal to 1.

Ex:- $(110101)_2 = 32 + 16 + 4 + 1 = (53)_{10}$

↳ 2^{10} is referred as Kilo (K)
↳ 2^{20} is referred as Mega (M)
↳ 2^{30} is referred as Giga (G)
↳ 2^{40} is referred as tera (T)

1KB = 1024 bytes
1MB = 1024 kB
1GB = 1024 MB
1TB = 1024 GB

∴ so $4K = 2^{12} = 4096$
$16M = 2^{24} = 16,777,216$.

↳ A computer hard disk with four gigabytes of storage has a capacity of $4G = 2^{32}$ bytes (approx., 4 billion bytes)
↳ A terabyte is 1024 gigabytes (approx, 1 trillion bytes)

Extra
Examples of Binary numbers:-

(i) Addition

augend: 101101
addend: + 100111
sum: 1010100

(ii) Subtraction

minuend: 101101
subtrahend - 100111
difference: 000110

(iii) multiplication.

multiplicand: 1011
multiplier × 101
partial product
1011
0000
10111
product: 110111

Number system	Base	First digit	Last digit	All digits / characters.
Binary	2	0	1	0, 1
Octal	8	0	7	0, 1, 2, 3, 4, 5, 6, 7
Decimal	10	0	9	0, 1, 2, 3, 4, 5, 6, 7, 8, 9
Hexadecimal	16	0	F	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F.

Ex:- Relatⁿ b/w Binary decimal etc.,

Decimal	Binary	Octal	Hexadecimal.
0	0000	0	0
4	0100	4	4
8	1000	10	8
11	1011	13	B
15	1111	17	F

NUMBER BASE CONVERSIONS :-

The human beings use decimal number system while computer uses binary number system. Therefore it is necessary to convert decimal number into its equivalent binary number while feeding to computer and again convert to decimal ~~which~~ while displaying result.

- Binary to Octal
- Octal to Binary
- Binary to Hexadecimal
- Hexadecimal to Binary
- Octal to Hexadecimal
- Hexadecimal to Octal.
- Decimal to Binary
- Decimal to Octal.

(i) Convert decimal 41 to binary.

To convert decimal number to a binary, divide decimal number by 2 successively & its reminders form binary number & the final remainder becomes the most significant digit.

Ex:- $(41)_{10} =$

$$
\begin{array}{r|l l}
2 & 41 & \\
\hline
2 & 20 & -1 \\
\hline
2 & 10 & -0 \\
\hline
2 & 5 & -0 \\
\hline
2 & 2 & -1 \\
\hline
2 & 1 & -0 \\
\hline
 & 0 & -1
\end{array}
$$

$\therefore (41)_{10} = (\underset{a_5}{1}\underset{a_4}{0}\underset{a_3}{1}\underset{a_2}{0}\underset{a_1}{0}\underset{a_0}{1})_2$

If decimal number contains Fraction part, then it requires successive multiplication of the fractional part by 2, with the integer part after each multiplication becomes the next lower digit in the binary fraction.

$(41.375)_{10}$ — Integer part

	Integer part
$0.375 \times 2 = 0.75$	0
$0.75 \times 2 = 1.5$	1
$0.5 \times 2 = 1$	1

$\therefore (41.375)_{10} = (101001.011)_2$

(ii) Convert binary to decimal.

To convert binary to decimal, coefficients of binary number are multiplied by powers of 2.

Ex:- $(1001)_2 = 1\times 2^3 + 0\times 2^2 + 0\times 2^1 + 1\times 2^0 = (9)_{10}$

(iii) Convert Binary to Octal

The base for octal number is 8 & the base for binary number is 2. The base for octal number is the third power of the base of binary numbers. So, by grouping 3 digits of binary numbers & then converting each group digit to its octal equivalent we can convert binary number to octal equivalent.

Ex:- $(10101101 \cdot 0111)_2$ to octal.

(i) Make group of 3 bits starting from LSB for Integer part & MSB for fractional part by adding 0's at the end, if required.

010	101	101	.	011	100	Binary
2	5	5	.	3	4	Octal

$\therefore (10101101 \cdot 0111)_2 = (255.34)_8$

Adding 0's at MSB to make group of 3-bits

Adding 0's at LSB to make group of 3 bits.

(iv) Convert Octal to Binary.

Each digit of the octal number is individually converted to its binary equivalent to get octal to binary conversion of the number.

Ex:- $(125.62)_8$

↳ Remove any leading or trailing zeros.

1	2	5	.	6	2
001	010	101	.	110	010

1 0 1 0 1 0 1 . 1 1 0 0 1

$(125.62)_8 = (1010101.11001)_2$.

(v) Convert Binary to Hexadecimal

The base for hexadecimal is the fourth power of the base for binary numbers. Therefore by grouping 4 digits of binary numbers & then converting each group digit to its hexadecimal equivalent we can convert binary number to its hexadecimal equivalent.

Ex:- $(1101101110.1001101)_2$

0011	0110	1110	.	1001	1010
3	6	E	.	9	A

Adding 0's to make group of 4 bits.

$\therefore (1101101110.1001101)_2$

$= (36E.9A)_{16}$

(vi) Hexadecimal to Binary Conversion.

Ex:- $(8A9.B4)_{16}$

8	A	9	.	B	4
1000	1010	1001	.	1011	0100

$\therefore (8A9.B4)_{16} = (100010101001.101101)_2$

(vii) Convert decimal to octal.

Eg:- $(0.513)_{10}$

$0.513 \times 8 = 4.104$

$0.104 \times 8 = 0.832$

$0.832 \times 8 = 6.656$

$0.656 \times 8 = 5.248$

$0.248 \times 8 = 1.984$

$0.984 \times 8 = 7.872$

$\therefore (0.513)_{10} = (0.406517..)_8$

(viii) Convert octal to Hexadecimal.

Step-1:- Convert octal to ~~hexadecimal~~ binary equivalent.

Step-2:- Binary to its hexadecimal equivalent.

Eg:- $(615.25)_8$ to Hexadecimal.

6	1	5	.	2	5
110	001	101	.	010	101

0001	1000	1101	.	0101	0100
1	8	D	.	5	4

$(615.25)_8 = (18D.54)_{16}$

(ix) Convert Hexadecimal to octal.

- Convert Hexadecimal to Binary
- Convert Binary to octal.

Eg:- $(BC66.AF)_{16}$ to octal

B	C	6	6	.	A	F
1011	1100	0110	0110	.	1010	1111

1	011	110	001	100	110	.	101	011	110
1	3	6	1	4	6	.	5	3	6

(i) $(367)_8 = (x)_2$ (ii) $(378.93)_{10} = (x)_8$

(iii) $(B9F.AE)_{16} = (x)_8$ (iv) $(16)_{10} = (100)_x$. [∵ Dec-16 6Mks]

(i) $(367)_8 = (x)_2$

3	6	7	octal
011	110	111	

$\therefore (367)_8 = (11110111)_2$

(ii) $(378.93)_{10} = (x)_8$

8 | 378
8 | 47 - 2
5 - 7

$0.93 \times 8 = 7.44 \rightarrow 7$
$0.44 \times 8 = 3.52 \rightarrow 3$
$0.52 \times 8 = 4.16 \rightarrow 4$
$0.16 \times 8 = 1.28 \rightarrow 1$

$(572.734)_8$

(iii) B 9 F . A E

| 1011 | 1001 | 1111 | . | 1010 | 11100 |

$(5637.534)_8$

(iv) $(16)_{10} = (100)_x$

$16 = 1 \times x^2 + 0 \times x^1 + 0 \times x^0$

$16 = x^2$

$\rightarrow x = 4$

$\therefore (16)_{10} = (100)_4$

** Convert $(163.875)_{10}$ to Binary, octal Hexadecimal. [Dec-16 4Mks]

16 | 163
10 - 3
(A)

$\therefore (163)_{16} = (A3)_{16}$

$0.875 \times 16 = 14.0$

$\therefore (A3.E)_{16}$

A	3	.	E
01010	0011	.	1110
2 4	3	.	7

$\therefore (163.875)_{10} = (A3.E)_{16} = (10100011.1110)_2$
$= (243.7)_8$.

** Determine the Value of 'b' for the following.

(i) $(292)_{10} = (1204)_b$

$\therefore (292)_{10} = 1 \times b^3 + 2 \times b^2 + 0 \times b^1 + 4 \times b^0$

$= b^3 + 2b^2 + 4$

$\therefore b = 6$

(ii) $(16)_{10} = (100)_x$

$1 \times x^2 + 0 \times x^1 + 0 \times x^0$

$= x^2 = 16$

$x = 4$

Complements are used in digital computers to simplify the Subtraction Operation and for logical manipulation.

There are 2 types of complements for base number system.

(i) r's Compliment (or) Radix complement

(ii) (r-1)'s compliment (or) Diminished Radix complement.

↳ For decimal no's, the complements are termed as

(i) 10's complement

(ii) 9's complement

↳ For binary number system

(i) 2's complement

(ii) 1's complement.

↳ For Octal number system

(i) 8's complement

(ii) 7's complement.

↳ For Hexadecimal number system

(i) 16's complement

(ii) 15's complement.

	r's comp	(r-1)'s comp
r=10	10's comp	9's comp
r=2	2's comp	1's comp
r=8	8's comp	7's comp
r=16	16's comp	F's comp

∴ r's comp = $r^n - N$

(r-1)'s comp = $r^n - N - 1$

(r-1)'s comp = r's comp - 1

$\boxed{\text{(r-1)'s comp} + 1 = \text{r's comp}}$ (∵ By adding 1 on both sides)

No borrow — borrow.

$r^n - N$

6.1 * r's complement [Radix Complement]

The r's complement of an n-digit number N in base r is defined as $r^n - N$ for $N \neq 0$ and as 0 for $N = 0$.

↳ r's complement is obtained by adding 1 to the (r-1)'s complement.