

THEORY & OBJECTIVE

# DIGITAL ELECTRONICS

*By  
Team of  
Engineers Academy*

- Covers ESE, GATE, IAS, PSUs, DRDO, ISRO & other Technical Exams.
- Covers complete syllabus & all topics of ESE & GATE examination.
- Thoroughly revised, fully solved & error free book.
- Concise, concept oriented and topicwise presentation & authentic solutions.



**ENGINEERS ACADEMY<sup>®</sup>**

Your GATEway to Professional Excellence

**IES • GATE • PSUs • JTO • IAS • NET**

**CORPORATE OFFICE**

100-102, Ram Nagar, Bambala Puliya, Tonk Road, Pratap Nagar, Jaipur-302033

**Ph.: + 91-8094441777**

**Website : [www.engineersacademy.org](http://www.engineersacademy.org) | Email: [info@engineersacademy.org](mailto:info@engineersacademy.org)**

Ajmer | Jaipur | Kota | Jodhpur | Bhilwara | Delhi | Patna | Lucknow | LPU | Ludhiana | Jalandhar | Kanpur | Allahabad



# NUMBER SYSTEMS & BINARY ARITHMETIC

## THEORY

### 1.1 INTRODUCTION

A decimal number system is said to be base 10 because it uses 10 digits (0 to 9).

A binary number system has base 2 because it uses 2 digits (0 and 1).

Any number to the base 'r' uses r digits from 0 to (r - 1). If the value of base  $r \leq 10$ , r digits are taken from decimal number system. If the value of base  $r > 10$  the letters of alphabets are used in addition to 10 decimal digits (0 to 9).

**Ex.:** Hexadecimal number system have base 16 and it requires 16 digits. Out of sixteen digits, 10 digits are taken from decimal number system while remaining 6 digits (10 to 15) are represented by letters (A to F) respectively.

### 1.2 TYPE OF NUMBER SYSTEMS

- Binary number system (Base-2)
- Octal number system (Base-8)
- Decimal number system (Base-10)
- Hexadecimal number system (Base-16)

#### 1.2.1 Binary Number System

This number system uses 0 and 1 as digits. Its radix or base is 2. Binary number system consists of only two digits i.e. 0 and 1. In binary number system, each digit is called a bit, the group of four bits is called a nibble and group of 8 bits is called a byte. The highest decimal number represented by n digit number is  $2^n - 1$ . For example, with 4 bit binary number, highest decimal number represented is

$$2^4 - 1 = 15$$

**(i) Decimal to Binary Conversion :** A decimal number can be converted into binary number dividing the decimal number by two progressively until quotient zero is obtained. Binary number is obtained by taking remainder of each division in reverse order with first remainder as LSB and last remainder as MSB.

If decimal number is a fraction, its binary equivalent is obtained by multiplying the number progressively by 2 with carry generated as binary digit. First carry is taken as MSB and last carry as LSB.

# CONTENTS

S.No.	TOPIC	PAGE No.
1.	NUMBER SYSTEMS AND BINARY ARITHMETIC.....	1 – 15
	PRACTICE SHEET .....	16 – 24
2.	BOOLEAN FUNCTIONS .....	25 – 35
	PRACTICE SHEET .....	36 – 46
3.	LOGIC GATES .....	47 – 57
	PRACTICE SHEET .....	58 – 69
4.	COMBINATIONAL CIRCUITS .....	70 – 86
	PRACTICE SHEET .....	87 – 106
5.	SEQUENTIAL CIRCUITS .....	107 – 136
	PRACTICE SHEET .....	137 – 158
6.	MEMORIES .....	159 – 164
	PRACTICE SHEET .....	165 – 170
7.	LOGIC FAMILIES .....	171 – 184
	PRACTICE SHEET .....	185 – 193
8.	ADCs AND DACs .....	194 – 202
	PRACTICE SHEET .....	203 – 207
9.	MISCELLANEOUS .....	208 – 212
	PRACTICE SHEET .....	213 – 215

**Example 1 :** Convert the  $(117)_{10}$  into binary.

**Solution :**

	117	Quotient	Remainder	
2	117			
2	58	1	1	LSB
2	29	0	0	↑
2	14	1	1	↑
2	7	0	0	↑
2	3	1	1	↑
2	1	1	1	↑
2	0	1	1	MSB

$$(117)_{10} = (1110101)_2$$

**Example 2 :** Convert the  $(0.61)_{10}$  into binary.

**Solution :**

Conversion of 0.61 in to Binary

0.61 × 2 = 1.22	(MSB)	1
0.22 × 2 = 0.44		0
0.44 × 2 = 0.88		0
0.88 × 2 = 1.76		1

only upto 4 or 5 digits

$$\Rightarrow (0.61)_{10} = (0.1001)_2$$

**(ii) Binary to Decimal Conversion :**  $(\dots X_4 X_3 X_2 X_1 X_0 X_{-1} X_{-2} X_{-3} \dots)$

**Decimal equation :**  $[(\dots X_3 r^3 + X_2 r^2 + X_1 r^1 + X_0 r^0) \cdot (X_{-1} r^{-1} + X_{-2} r^{-2} + X_{-3} r^{-3} + \dots)]$

For binary number system,  $r = 2$

**Ex.:**  $(101101.001)_2 = (X)_{10}$

$$(1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0) \cdot (0 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3})$$

$$[1 + 0 + 4 + 8 + 0 + 32] \cdot \left[\frac{1}{8}\right] = (45.125)_{10}$$

### 1.2.2 Octal Number System

This number system uses 0, 1, 2, 3, 4, 5, 6 and 7 as digits. Its radix or base is 8.

**(i) Conversion from Decimal to Octal :** Absolute (Integral value) : An absolute decimal number is converted into octal number by dividing the number by 8 progressively as in binary system.

**Fractional number :** A fractional number be converted to octal number by multiplying the number by 8 progressively as in case of binary system.

**(ii) Conversion from Octal to Decimal :** A number with radix or base 'r' can be converted to its decimal equivalent by multiplying each digit by its respective weight and adding the products.

**Ex.:**  $(467.32)_8 = (X)_{10}$

For octal number system,  $r = 8$

$$X = (7 \times 8^0 + 6 \times 8^1 + 4 \times 8^2) \cdot (3 \times 8^{-1} + 2 \times 8^{-2})$$

$$= (7 + 48 + 256) \cdot \left(\frac{3}{8} + \frac{2}{64}\right)$$

$$= 311 \cdot (0.375 + 0.03125)$$

$$= (311.406)_{10}$$

### 1.2.3 Decimal Number System

Decimal number system consists of 10 digits

i.e. 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.

The base or radix of this number system is 10.

Digits = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 and base = 10.

### 1.2.4 Hexadecimal Number System

Hexadecimal number system consists of 16 digits

i.e. 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E and F.

The base or radix of this number system is 16.

Digits = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F and base = 16.

#### (i) Conversion from Decimal to Hexadecimal

(a) **Integral Part** : An absolute decimal number is converted to hexadecimal number by dividing the number by 16 progressively in similar manner as in binary number system.

(b) **Decimal Part / Fractional Part** : A fractional decimal number is converted to hexadecimal number by multiplying the number by 16 progressively in similar manner as in binary number system.

(ii) **Conversion from Hexadecimal to Decimal** : A hexadecimal number can be converted to its decimal equivalent by multiplying each digit by its respective weight and adding all the products.

**Example 3** :  $(B7AF.3E)_{16} = (X)_{10}$  find the value of X.

**Solution** :

$$\begin{aligned} X &= (11 \times 16^3 + 7 \times 16^2 + 10 \times 16^1 + 15 \times 16^0) \left( \frac{3}{16} + \frac{14}{16^2} \right) \\ &= (11 \times 4096 + 7 \times 256 + 10 \times 16 + 15 \times 1) \left( \frac{3}{16} + \frac{14}{256} \right) \\ &= (45056 + 1792 + 160 + 15) \cdot (0.1875 + 0.0546) \\ &= (47023.242)_{10} \end{aligned}$$

## 1.3 CONVERSION OF NUMBER SYSTEM

### 1.3.1 Binary to Hexadecimal

Binary to Hexadecimal conversion is obtained by making group of four bits to left and right of decimal point. Then each group is replaced by its hexadecimal equivalent number.

### 1.3.2 Hexadecimal to Binary

Hexadecimal to binary conversion is obtained by replacing each digit by its binary equivalent of four bits.

$$\text{Ex.: } (1A3.BA)_{16} = \left( \underbrace{0001}_1 \underbrace{1010}_A \underbrace{0011}_3 \underbrace{1011}_B \underbrace{1010}_A \right)_2$$

### 1.3.3 Binary to Octal Conversion

A binary number can be converted to its octal equivalent by making group of three bits to left and right of decimal point. Then each group is replaced by its octal equivalent number.

Ex.:  $(101110010.0011)_2 = (X)_8$

$\underbrace{101}_5$

$\underbrace{110}_6$

$\underbrace{010}_2$

$\underbrace{001}_1$

$\underbrace{11}_4$

$X = 562.14$

### 1.3.4 Octal to Binary Conversion

Octal to binary conversion is obtained by replacing each digit by its binary equivalent of three bits.

Ex.:  $(462.71)_8 = (X)_2$

$X = 100\ 110\ 010\ .\ 111\ 001$

### 1.3.5 Hexadecimal to Octal Conversion

Hexadecimal to octal conversion is obtained by converting hexadecimal to binary and then from binary to octal number.

### 1.3.6 Octal to Hexadecimal Conversion

Octal to hexadecimal conversion is obtained by converting octal to binary and then from binary to hexadecimal number.

**Table of Different Number System**

Decimal	Binary	Octal	Hexadecimal
0	0000	00	0
1	0001	01	1
2	0010	02	2
3	0011	03	3
4	0100	04	4
5	0101	05	5
6	0110	06	6
7	0111	07	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

### 1.3.7 Arithmetic Operation

Binary addition, subtraction, multiplication :

**(i) Binary Addition**

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

Ex.: 
$$\begin{array}{r} 101011 \\ +101100 \\ \hline 1010111 \end{array}$$

**(ii) Binary Subtraction**

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

Ex.: 
$$\begin{array}{r} 11011 \\ -10110 \\ \hline 00101 \end{array}$$

**(iii) Binary multiplication**

Ex.: 
$$\begin{array}{r} 1011 \\ 101 \\ \hline 1011 \\ 0000 \\ 1011 \\ \hline 11011 \end{array}$$

**Octal addition, subtraction :**

- 0 + 0 = 0
- 0 + 3 = 3
- 1 + 4 = 5
- 1 + 7 = 10
- 7 + 3 = 12

$[7 + 1 = (8)_{10} = (10)_8]$

Ex.: Octal sum

243	564
+322	+323
<u>565</u>	<u>1107</u>



Octal subtract

$$\begin{array}{r} 8 \\ 734 \\ -362 \\ \hline 352 \end{array}$$

### Hexadecimal (addition, subtraction)

$$1 + 1 = 2$$

$$1 + 7 = 8$$

$$2 + 9 = B$$

$$A + A = 14$$

$$[A + A = (20)_{10} = (14)_{16}]$$

Ex.: Hexadecimal sum

8689	DAD
<u>3032</u>	<u>ADD</u>
<u>B6BB</u>	<u>188A</u>

Hexadecimal subtract

<u>1616</u>	6287
956C	-2145
<u>-274D</u>	<u>4142</u>
6E1F	

## 1.4 CODES

Codes are symbolic representation of discrete information, which may be present in the form of number, letters or physical quantities. These symbols are used to communicate information to computers.

Note :

- (i) Maximum number of distinct quantities using  $n$  bits =  $2^n$ .
- (ii) Each digit of decimal number system is represented by a four digit code in weighted and non weighted codes. Except Gray code.

### 1.4.1 Weighted Binary Codes

These are the codes in which each symbol position is assigned a weight. In weighted binary code, each bit is multiplied by its weight indicated in code to get the decimal equivalent of the code.

e.g. BCD (8421), 7421, 5421, 5211, 4221, 3321, 2421,  $84\bar{2}1$ ,  $74\bar{2}1$ .

**Example 4 :** Convert  $(8)_{10}$  in 7421, 5421, 5211, 4221, 3321, 2421,  $84\bar{2}1$ ,  $74\bar{2}1$  codes.

**Solution :**

- (a)  $(8)_{10} = 7 + 1 = (1001)_{7421}$
- (b)  $(8)_{10} = 5 + 2 + 1 = (1011)_{5421}$
- (c)  $(8)_{10} = 5 + 2 + 1 = (1101)_{5211} = (1110)_{5211}$
- (d)  $(8)_{10} = 4 + 2 + 2 = (1110)_{4221}$
- (e)  $(8)_{10} = 3 + 3 + 2 = (1110)_{3321}$
- (f)  $(8)_{10} = 2 + 4 + 2 = (1110)_{2421}$

(g)  $(8)_{10} = 8 + 0 + 0 + 0 = (1000)_{8421}$

(h)  $(8)_{10} = 7 + 4 - 2 - 1 = (1111)_{7421}$

**Note : Self complement :** A code is called self complementing code if its complement of the coded number yield 9's complement of number itself i.e., '0' is complement of '9', 1 is complement of '8', 2 is of 7, 3 is of 6, 4 is of 5.

Example of self complement codes is 5211, 4221, 3321, 2421,  $84\bar{2}1$ .

Ex.: Representation of  $(8)_{10}$  in 5211 code

- $(1110)_{5211} = (8)_{10}$

- $(1101)_{5211} = (8)_{10}$

Both are correct.

- For self complementary, '1' should be complement of code of '8'.

### 1.4.2 Non Weighted Codes

Non-weighted codes are codes which are not positionally weighted, excess-3 code and gray codes are example of non-weighted codes.

(i) **Excess-3 :** Excess-3 code is obtained by adding 0011 to each BCD code of decimal numbers.

e.g.

decimal	BCD	Excess-3
0	0000	0011
1	0001	0100
2	0010	0101
:	:	:
9	1001	1100

**Note :** Excess-3 code is self complementing code.

(ii) **Gray code :** In gray code, every new code differs from the previous code only by a single bit. That is why, it is also called Reflected code. It can be used for measurement of shaft speed.

**Note :** It is also called unit distance code or minimum change code.

### 1.4.3 Alpha Numeric Codes

- ASCII (American Standard code for Information Interchange) :** It is a 7 bit code. It is most commonly used in microcomputers. This code represents a character with seven bits, which can be stored as one byte with one bit used.
- ISCII (Indian Standard code for Information Interchange).**
- EBCDIC (ebb. see-dic) :** It uses eight bits for character and ninth bit for parity.
- Hollerith Code :** It is used in punch cards used for storage of information.
- Morse Code :** It is used in Telegraph.
- Tele Type Writer (TTW).**

### 1.4.4 Error Detecting and Correcting Codes

- (i) **Parity Check Code** : This code detects single bit error which changes parity of information. Errors with more than one bit cannot be detected with this code.
- (ii) **Hamming Code** : This code detects as well as corrects the error in an information. This code transmits additional parity bits in addition to information bits. These parity bits are used to detect and correct the error.

### 1.4.5 Error Detecting Code

(i) **Even Parity Method** : In this method, the value of parity is added so that total number of 1's in code group (including parity bit) is even.

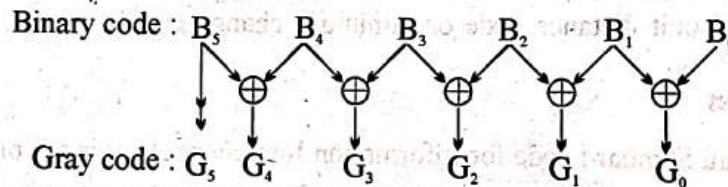
$$\text{Parity} = 0 \text{ or } 1$$

(ii) **Odd Parity Method** : In this method, the value of parity is added so that total number of 1's in code group (including parity bit) is odd.

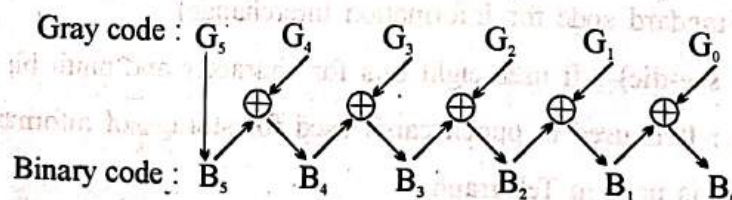
Ex.: (1011) → odd  
(11000) → even

## 1.5 CODE CONVERSIONS

- (i) **Decimal to BCD conversion** : Decimal to BCD conversion is obtained by replacing each digit of decimal number by its binary equivalent.
- (ii) **Binary to BCD conversion** : Binary to BCD code conversion is possible by converting binary to decimal and then from decimal to its BCD equivalent.
- (iii) **BCD to Binary conversion** : BCD to binary conversion is obtained by converting the BCD to decimal and then decimal to its binary equivalent.
- (iv) **BCD to Excess-3 conversion** : BCD to excess-3 code conversion is obtained by adding 0011 to each BCD coded digit.
- (v) **Excess-3 to BCD conversion** : Each (Excess-3 code) - 0011 = BCD  
Excess-3 to BCD code conversion is obtained by subtracting 0011 from excess-3 code of each digit.
- (vi) **Binary to gray code conversion** : Binary to gray code conversion is obtained by using rules as shown under.



(vii) **Gray to binary code conversion** : Gray to binary code conversion is obtained by using rules as shown under.



**1.6 BCD AND EXCESS-3 ARITHMETIC**

**1.6.1 BCD**

(a) BCD Addition :

Rules :

- (i) If result of addition contains invalid BCD, add 0110 to that BCD.
- (ii) If carry is passed from lower BCD to next BCD add 0110 to that BCD.

**Example 5 :**  $(80)_{BCD} + (80)_{BCD} = ?$

**Solution :**

$$\begin{array}{r}
 1000 \quad 0000 \\
 1000 \quad 0000 \\
 \hline
 \text{I} \quad 0000 \quad 0000 \\
 \quad 0110 \\
 \hline
 1 \quad 0110 \quad 0000 \\
 \hline
 1 \quad 6 \quad 0
 \end{array}$$

$(80)_{BCD} + (80)_{BCD} = (160)_{BCD}$

**Example 6 :**  $(88)_{BCD} + (88)_{BCD} = ?$

**Solution :**

$$\begin{array}{r}
 1000 \quad 1000 \\
 1000 \quad 1000 \\
 \hline
 \text{I} \quad 0001 \quad 0000 \\
 \quad 0110 \quad 0110 \\
 \hline
 1 \quad 0111 \quad 0110 \\
 \hline
 1 \quad 7 \quad 6
 \end{array}$$

$(88)_{BCD} + (88)_{BCD} = (176)_{BCD}$

**Example 7 :**  $(23)_{BCD} + (88)_{BCD} = ?$

**Solution :**

$$\begin{array}{r}
 0010 \quad 0011 \\
 1000 \quad 1000 \\
 \hline
 1010 \quad 1011 \\
 \text{invalid BCD} \quad \text{invalid BCD} \\
 \hline
 \text{Step-I} \quad \text{I} \quad 0110 \\
 \hline
 1011 \quad 0001 \\
 \hline
 \text{Step-II} \quad 0110 \\
 \hline
 1 \quad 0001 \quad 0001 \\
 \hline
 1 \quad 1 \quad 1
 \end{array}$$

$(23)_{BCD} + (88)_{BCD} = (111)_{BCD}$



**Example 8 :**  $(76)_{BCD} + (86)_{BCD} = ?$

**Solution :**

$$\begin{array}{r}
 \begin{array}{cc}
 0111 & 0110 \\
 1000 & 0110 \\
 \hline
 1111 & 1100 \\
 \text{invalid BCD} & \text{invalid BCD} \\
 \text{Step-I} & \boxed{1} \quad 0110 \\
 \hline
 0000 & 0001 \\
 \text{Step-II} & 0110 \\
 \hline
 1 & 0110 & 0010 \\
 \hline
 1 & 6 & 2
 \end{array}
 \end{array}$$

$$(76)_{BCD} + (86)_{BCD} = (162)_{BCD}$$

**(b) BCD Subtraction :** Subtraction BCD is performed either by 9's complement or by 10's complement. Rules are similar to binary subtraction with 1's and 2's complement.

### 1.6.2 Excess-3 Arithmetic

**(a) Excess-3 addition :**

**Rules :**

- (i) Convert the given number to Excess-3 format.
- (ii) Now add Excess-3 numbers.
- (iii) If carry is generated by Excess-3 code of 2 digits, add  $(0011)_2$  to the sum of digits.
- (iv) If carry is not generated, subtract  $0011$  from sum of two digits.
- (v) Subtracting  $(0011)_2$  is equivalent to adding  $(1101)_2$  or  $(13)_{10}$ .

**Example 9 :**  $(38)_{10} + (44)_{10} = ?$

**Solution :**

$$\begin{array}{r}
 \begin{array}{cc}
 0110 & 1011 \\
 0111 & 0111 \\
 \hline
 \boxed{0} & \boxed{1} \\
 1110 & 0010 \\
 -0011 & +0011 \\
 \hline
 \text{ignored} & \boxed{1} \quad 1011 & 0101 \\
 & (8) & (2)_{10}
 \end{array}
 \end{array}$$

**(b) Excess-3 subtraction :** Excess-3 subtraction is done with 9's complement or 10's complements.

## 1.7 NUMBER REPRESENTATION IN BINARY NUMBER SYSTEM

### 1.7.1 Sign Magnitude Representation

Left most bit represents sign.

For example  $\underbrace{B_{n-1}}_{\text{Sign}} \underbrace{B_{n-2} B_{n-3} \dots B_2 B_1 B_0}_{\text{Magnitude}}$

**Note :** Range =  $-(2^{n-1} - 1)$  to  $+(2^{n-1} - 1)$

**Example 10 :** Find sign Magnitude representation of  $(-20)_{10}$  and  $(+20)_{10}$ .

**Solution :**

$$-20 \Rightarrow \overset{1}{\downarrow} \underbrace{10100}_{20}$$

$$+20 \Rightarrow \overset{0}{\downarrow} \underbrace{10100}_{20}$$

**Example 11 :** Find sign magnitude representation of  $(-0)_{10}$  and  $(+0)_{10}$ .

**Solution :**

$$-0 \Rightarrow 1000$$

$$+0 \Rightarrow 0000$$

**Note :** Zero has two distinct representation in sign magnitude representation.

### 1.7.2 One's Complement Representation

When each bit of number is complemented, resulting number is one's complement of original number. One's complement of a number gives its negative number.

**Note :** Range =  $-(2^{n-1} - 1)$  to  $+(2^{n-1} - 1)$ .

**Example 12 :** Find the one's complement of  $(-20)_{10}$ .

**Solution :**

$$(+20)_{10} \Rightarrow (00010100)_2 \text{ original number}$$

$$(-20)_{10} \Rightarrow (11101011)_2 \text{ one's complement}$$

**Example 13 :** Find the one's complement of  $(-0)_{10}$  and  $(+0)_{10}$ .

**Solution :**

$$(+0)_{10} \Rightarrow (0000)_2$$

$$(-0)_{10} \Rightarrow (1000)_2$$

Two distinct representation of zero.

**Note :**

- 1's complement representation of zero has two distinct representations.
- 1's complement of a number gives original number.

### 1.7.3 Two's Complement Representation

2's complement of a number = 1's complement of the number + 1

**Note :** 2's complement has unique representation of zero.

$$\begin{array}{r} \text{Ex.:} \quad (0)_{10} = (0000)_2 \\ \quad \quad \quad 1111 \\ \quad \quad \quad +1 \\ \hline \end{array}$$

Final carry ignored  $\rightarrow \boxed{1}0000$  2's complement

**Note :** Range  $\rightarrow -2^{n-1}$  to  $(2^{n-1} - 1)$ .

**Example 14 :** Find 2's complement representation of  $(-37)_{10}$ .

**Solution :**

$$\begin{aligned} (+37)_{10} &= (00100101)_2 \\ \text{2's complement representation of } (-37)_{10} &= \text{1's complement of } (00100101)_2 + 1 \\ &= (11011010)_2 + 1 \\ &= (11011011)_2 \end{aligned}$$

**Table of number representation**

Binary	Sign magnitude	1's	2's
0000	+0	+0	+0
0001	+1	+1	+1
0010	+2	+2	+2
0011	+3	+3	+3
0100	+4	+4	+4
0101	+5	+5	+5
0110	+6	+6	+6
0111	+7	+7	+7
1000	-0	-7	-8
1001	-1	-6	-7
1010	-2	-5	-6
1011	-3	-4	-5
1100	-4	-3	-4
1101	-5	-2	-3
1110	-6	-1	-2
1111	-7	-0	-1

### 1.8 r's COMPLEMENT & (r-1)'s COMPLEMENT

(i) **r's complement :** r's complement of a positive number 'N' of n digits in integral part =  $r^n - N$

Where, r = radix or base

n = number of digits in integral part of the number.

**Ex.:**

- (1) 10's complement of  $(52520)_{10}$  is  
 $10^5 - 52520 = 47480$  here n = 5 and r = 10
- (2) 10's complement of  $(0.3267)_{10}$  is  
 $(10^0 - 0.3267) = 0.6733$  here n = 0 and r = 10
- (3) 2's complement of  $(101100)_2$  is  
 $(2^6)_{10} - (101100)_2 = (1000000)_2 - (101100)_2 = 010100$
- (4) 2's complement of  $(0.0110)_2$  is  
 $2^0 - 0.0110 = 1 - 0.0110 = (0.1010)_2$



## OBJECTIVE QUESTIONS

1. The binary number 00001011 when represented in BCD format, is given by  
 (a) 00001011      (b) 10111011  
 (c) 00010001      (d) 10001000
2. When two numbers are added in excess-3 code and the sum is less than 9, then in order to get the correct answer it is necessary to  
 (a) subtract 0011 from the sum  
 (b) add 0011 to the sum  
 (c) subtract 0110 from the sum  
 (d) add 0110 to the sum
3. Gray code of the decimal number 13 is  
 (a) 1010      (b) 1101  
 (c) 1011      (d) 1001
4. The binary representation 100110 is numerically equivalent to  
 (a) the decimal representation 46  
 (b) the octal representation 46  
 (c) the octal representation 26  
 (d) the decimal representation 26
5. The number 7F in Hexadecimal number system is equivalent to the decimal number  
 (a) 255      (b) 256  
 (c) 128      (d) 127
6. Decimal 43 in Hexadecimal and BCD number system is respectively  
 (a) B2, 0100 0011  
 (b) 2B, 0100 0011  
 (c) 2B, 0011 0100  
 (d) B2, 0100 0100
7. The decimal value 0.25  
 (a) is equivalent to the binary value 0.1  
 (b) is equivalent to the binary value 0.01  
 (c) is equivalent to the binary value 0.00111  
 (d) cannot be represented precisely in binary
8. Which of the following operations do result in  $(EA)_{16}$  ?  
 1.  $(AB)_{16} - (3F)_{16}$   
 2.  $(BC)_{16} - (CB)_{16}$   
 3.  $(FE)_{16} - (14)_{16}$   
 Select the correct answer using the code given below  
 (a) 1 and 2      (b) 1 and 3  
 (c) 2 and 3      (d) 1, 2 and 3
9. What is the Gray code word for the binary number 101011 ?  
 (a) 101011      (b) 110101  
 (c) 011111      (d) 111110
10. How many 1's are present in the binary representation of  
 $(4 \times 4096) + (9 \times 256) + (7 \times 16) + 5$   
 (a) 8      (b) 9  
 (c) 10      (d) 11
11. Match List-I (Binary) with List-II (Decimal) and select the correct answer using the codes given below the lists  

List-I	List-II
A. 10101010	1. 128
B. 11110000	2. 240
C. 10001000	3. 170
D. 10000000	4. 136

Codes :

	A	B	C	D
(a)	3	2	4	1
(b)	2	3	1	4
(c)	2	4	1	3
(d)	3	1	2	4
12. What is the octal equivalent of decimal 0.3125?  
 (a) 0.42      (b) 0.3125  
 (c) 0.24      (d) 0.12

13. Match List-I (Octal) with List-II (Binary) and select the correct answer using the codes given below the lists:

List-I	List-II
A. 75	1. 010110
B. 65	2. 110101
C. 37	3. 111101
D. 26	4. 011111

Codes :

	A	B	C	D
(a)	3	1	4	2
(b)	2	1	3	4
(c)	3	2	4	1
(d)	4	2	3	1

14. In hexadecimal arithmetic, the result of  $(77)_{16} - (3B)_{16}$  is equal to

- (a)  $3D_{16}$  (b)  $3C_{16}$   
(c)  $60_{16}$  (d)  $73_{16}$

15. The decimal equivalent of hexadecimal number 2A0F is

- (a) 17670 (b) 17607  
(c) 17067 (d) 10767

16. The binary equivalent of hexadecimal number 4F2D is

- (a) 0101 1111 0010 1100  
(b) 0100 1111 0010 1100  
(c) 0100 1110 0010 1101  
(d) 0100 1111 0010 1101

17. Match list-I (Hexadecimal) with list-II (Octal) and select the correct answer by using the codes given below the lists

List-I	List-II
A. 68	1. 150
B. 8C	2. 214
C. 4F	3. 117
D. 5D	4. 135

Codes :

	A	B	C	D
(a)	3	1	4	2
(b)	1	2	3	4
(c)	3	2	4	1
(d)	4	1	2	3

18. F's complement of  $(2BFD)_{16}$  is

- (a) E 304 (b) D 403  
(c) D 402 (d) C 403

19. Given  $(125)_R = (203)_5$ , the value of radix R will be

- (a) 16 (b) 10  
(c) 8 (d) 6

20. If  $(327)_9 = (X)_5$ , then the value of X is given by

- (a) 327 (b) 268  
(c) 2033 (d) 3302

21. Which of the following is a self-complementing code?

- (a) 8421 code (b) Excess 3 code  
(c) Pure binary code (d) Gray code

22. The parity bit is added for (purpose)

- (a) coding (b) indexing  
(c) error-detection (d) controlling

23. Which one of the following is a non-valid BCD code?

- (a) 0111 1001 (b) 0101 1011  
(c) 0100 (d) 0100 1001

24. A three-input NAND gate is to be used as an inverter. Which one of the following measures will achieve better results?

- (a) The two inputs not used are kept open  
(b) The two inputs not used are connected to ground ('0' level)  
(c) The two inputs not used are connected to logic ('1' level)  
(d) None of the above

25. How is inversion achieved using Ex-OR gate?

- (a) Giving input signal to the two input lines of the gate tied together  
(b) Giving input to one input line and logic zero to the other line  
(c) Giving input to one input line and logic one to the other line  
(d) Inversion cannot be achieved using Ex-OR gate

(b)  $(11110000)_2$  then its decimal equivalent will be.

$$N = 1 \times 2^7 + 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$$

$$\Rightarrow N = 1 \times 2^7 + 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4$$

$$\Rightarrow N = (240)_{10}$$

(c)  $(10001000)_2$  then its decimal equivalent will be

$$N = 1 \times 2^7 + 0 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$$

$$N = 1 \times 128 + 1 \times 8$$

$$= 128 + 8$$

$$= (136)_{10}$$

(c)  $(10000000)_2$  then its decimal equivalent will be

$$N = 1 \times 2^7 + 0 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$$

$$N = 1 \times 2^7$$

$$= (128)_{10}$$

12. **Ans. (c)**

A fractional decimal number is converted to its equivalent octal number by successive multiplication by 8 as follows

	Generated Integer
$0.3125 \times 8 = 2.5000$	2
$.5000 \times 8 = 4.0000$	4
$.0000 \times 8 = 0.0000$	0

The process is terminated when zeros or significant digits are obtained.

Thus octal equivalent of  $(0.3125)_{10}$  is  $(0.24)_8$

13. **Ans. (c)**

For obtaining the binary equivalent of an octal number, significant digit in the given number replaced by its 3 bit binary equivalent.

(a)  $(75)_8 = 75 = (111101)_2$

(b)  $(65)_8 = 65 = (110101)_2$

(c)  $(37)_8 = 37 = (011111)_2$

(d)  $(26)_8 = 26 = (010110)_2$

14. **Ans. (b)**

Method I :

$$(77)_{16} = 7 \times 16^1 + 7 \times 16^0$$

$$= 112 + 7$$

$$= (119)_{10}$$

$$(3B)_{16} = 3 \times 16^1 + B \times 16^0$$

$$= 3 \times 16^1 + 11 \times 16^0$$

$$= 48 + 11$$

$$= (59)_{10}$$

$$(77)_{16} - (3B)_{16} = (119)_{10} - (59)_{10}$$

$$= (60)_{10}$$

Now we can convert 60 from decimal to hexadecimal by dividing it by 16.

16	60	
16	3	12
		3

$\uparrow = (3C)_{16}$

Method II :

$$\begin{array}{r} \rightarrow 16 \\ 77 \\ - 3B \\ \hline 3C \end{array}$$

$$16 + 7 = 23$$

$$7 - B = 23 - B$$

$$= 23 - 11 = 12 = C$$

$$\therefore (77)_{16} - (3B)_{16} = (3C)_{16}$$

Note : The operation is same as that in decimal except that here 16 is used instead of 10.

15. **Ans. (d)**

The conversion from hexadecimal to a decimal number can be carried out by multiplying each significant digit of the hexadecimal by its respective weight and adding the products

$$2 \times 16^3 + A \times 16^2 + 0 \times 16^1 + 15 \times 16^0$$

$$= 2 \times 16^3 + A \times 16^2 + 0 \times 16^1 + F \times 16^0$$

$$= 8192 + 2560 + 0 + 15$$

$$= (10767)_{10}$$

16. **Ans. (d)**

To convert a hexadecimal number into its binary equivalent, each significant digit in the given number is replaced by its 4 bit binary equivalent

$$(4F2D)_{16} \rightarrow (x)_2$$

4	F	2	D
0100	1111	0010	1101