UNIT :I
NUMBER SYSTEM

- BY

- **BHANUDAS NARWADE**
  - ASST. PROF.
  - DEGLOOR COLLEGE,DEGLOOR

NUMBER SYSTEMS:

- 1. **Decimal Number System:**
- **Appreciated by Laplace**
- **Follows natural symbols 0,1,2,3,4,…….8,9 with position**
- **Base or Radix is number of different digits**
- **Decimal number system has 10 different digits**
- **Base or Radix is 10**
- **Higher numbers are written with weight of digits or place value of digit**

NUMBER SYSTEMS:

- 1. **Decimal Number System:**
  - **Weight of digits**
  - **Weight –** 10^{-4} 10^{-3} 10^{-2} 10^{-1} . 10^{0} 10^{1} 10^{2} 10^{3} 10^{4}
- **Weight value** $\frac{1}{10^4}$ $\frac{1}{10^3}$ $\frac{1}{10^2}$ $\frac{1}{10^1}$ **Decimal point** 1 10^{1} 10^{2} 10^{3} 10^{4}
- **0.0001 0.001 0.01 0.1 . 1 10 100 1000 10000**

NUMBER SYSTEMS:

- 1. **Decimal Number System:**
- **Find the value of digit in following decimal Number**
- **8632**

Number-	**8**	**6**	**3**	**2**
Weight –	**10^3**	**10^2**	**10^1**	**10^0**
Weight value	**1000**	**100**	**10**	**1**
	8× 1000	**6× 100**	**3× 10**	**2 × 1**
	8000 +	**600 +**	**30 +**	**2**

- **= 8632**

NUMBER SYSTEMS:

- 1. **Decimal Number System:**
- **Find the value of digit 4 in following decimal Number**
- **9428**

Number-	**9**	**4**	**2**	**8**
Weight –	10^3	10^2	10^1	10^0
Weight value	**1000**	**100**	**10**	**1**

- **Weight of digit 4 is 100**

NUMBER SYSTEMS:

- **9's Complement:**
  - **The 9' s complement of number is obtained by subtracting the given decimal number from digit 9**
  -

Digit N	0	1	2	3	4	5	6	7	8	9
9's Compleme nt (9-N)	9	8	7	6	5	4	3	2	1	0

NUMBER SYSTEMS:

- **9's Complement Subtraction:**
- **Rules:**
- **1) Keep minuend number as it is**
- **2) 9's compliments of subtrahend number**
- **3) Add two numbers**
- **4) Is carry? Yes -answer is positive add end round carry(ERC)**
  - **No - Answer is negative and in 9's complement form**

NUMBER SYSTEMS:

- **9's Complement Subtraction:**
- **Example:**

```
  7  ⟶    7
 -3  ⟶  + 6   9's complement
---------   -----
  4       1  3
             +1
            -----
              4
```

```
   4          4
  -7        + 2   9's complement
---------   -----
  -3          6  ans is –ve and in 9's
complement form
  9's completement of 6  is 3
```

NUMBER SYSTEMS:

- **9's Complement Subtraction:**
- **Example:**

```
  27 ⟶         27
 -19 ⟶       + 80   9's complement
---------    ------
   08          107
             +  1
             -----
                08
```

NUMBER SYSTEMS:

- **10’s Complement:The 10’ s complement of number is obtained by adding 1 in 9’s complement**

Digit N	0	1	2	3	4	5	6	7	8	9
9’s Complement	9	8	7	6	5	4	3	2	1	0
10’complement 9’s comp+1	10	9	8	7	6	5	4	3	2	1

NUMBER SYSTEMS:

- **10’s Complement Subtraction:**
- **Rules:**
- **1) Keep minuend number as it is**
- **2) 10’s compliments of subtrahend number**
- **3) Add two numbers**
- **4) Is carry? Yes -answer is positive discard end round carry(ERC)**
  - **No - Answer is negative and in 10’s complement form**

NUMBER SYSTEMS:

- **10’s Complement Subtraction:**
- **Example:**

```
   7  ──→    7
  -3  ──→  + 7   10's complement
---------  -----
   4        1 4  discard carry
             -----
               4
```

discard carry

```
   4            4
  -7          + 3   10's complement
---------     -----
  -3            7  ans is –ve and in
10's complement form
   10's completement of 7  is 3
```

BINARY NUMBER SYSTEM:

- 2. **Binary Number System:**
- **Number system with base two called binary number system**
- **It uses only two digits 0 and 1**
- **In digital electronics voltage level +5v =1 and 0v =0**
- **Sometimes 1 as HIGH and 0 as LOW**
- **Binary digits are called BIT**
- **Group of 4 bits is Nibble, 8 bit= Byte**
- **Position of 1 or 0 indicates its weight and increasing with power of 2**

BINARY NUMBER SYSTEM:

- 1. **Binary Number System:**

- **Weight –** 2^{-4} 2^{-3} 2^{-2} 2^{-1} . 2^{0} 2^{1} 2^{2} 2^{3} 2^{4} 2^{5} 2^{6}
- **Weight value** $\frac{1}{2^4}$ $\frac{1}{2^3}$ $\frac{1}{2^2}$ $\frac{1}{2^1}$ **Binary point** **1 2 4 8 16 32 64**

- **0.0625 0.125 0.25 0.5 . 1 2 4 8 16 32 64**

BINARY NUMBER SYSTEM:

- 1. **Binary Number System:**

Decimal	0	1	2	3	4	5	6	7	8
Binary	0000	0001	0010	0011	0100	0101	0110	0111	1000

Decimal	9	10	11	12	13	14	15	16	17
Binary	1001	1010	1011	1100	1101	1110	1111	10000	10001

BINARY ARITHMATICS:

Binary Addition:

- Performed in same manner as decimal

A	B	SUM	CARRY
0	0	0+0=0	0
0	1	0+1=1	0
1	0	1+0=1	0
1	1	1+1=0	1

In last case when binary 1 is added with 1 yields binary (10) i.e. sum=0 and carry=1 since 1 is largest digit in binary system

BINARY ARITHMATICS:

Three Bit Binary Addition:

A	B	C	SUM	CARRY
0	0	0	0+0+0=0	0
0	0	1	0+0+1=1	0
0	1	0	0+1+0=1	0
0	1	1	0+1+1=0	1
1	0	0	1+0+0=1	0
1	0	1	1+0+1=0	1
1	1	0	1+1+0=0	1
1	1	1	1+1+1=1	1

BINARY ARITHMATICS:

Examples:

1)

5	0 1 0 1
+ 2	+ 0 0 1 0
7	0 1 1 1

0 1 1 1	
00111	7
10101	21
11100	= 28

	1 0 0 1 0
+	1 0 0 1
	1 1 0 1 1

		1 1 1 1 1
$(85)_{10}$		01010101
$+(181)_{10}$	+	10110101
$(266)_{10}$		100001010

BINARY ARITHMATICS:

Binary Subtraction:

- Subtraction is inverse operation of addition

A	B	DIFF	BORROW
0	0	0-0=0	0
0	1	0-1=1	1
1	0	1-0=1	0
1	1	1-1=0	0

When 1 is subtracted from 0 , the reminder is 1 with borrow 1 from next column to the left

BINARY ARITHMATICS:

Three Bit Binary Subtraction:

A	B	C	DIFF	BORROW
0	0	0	0-0-0=0	0
0	0	1	0-0-1=1	1
0	1	0	0-1-0=1	1
0	1	1	0-1-1=0	1
1	0	0	1-0-0=1	0
1	0	1	1-0-1=0	0
1	1	0	1-1-0=0	0
1	1	1	1-1-1=1	1

BINARY ARITHMATICS(SUBTRACTION:

Examples:

1) 5 0 1 0 1

- 2 - 0 0 1 0

. ----------- ------------------

3 0 0 1 1

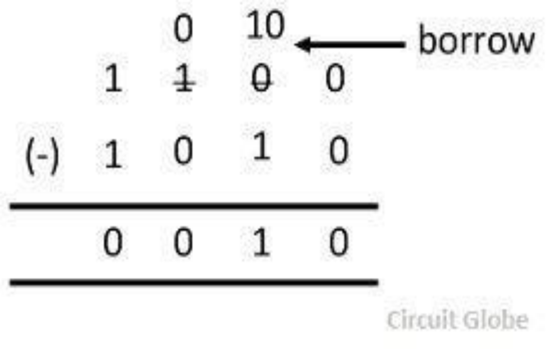
4 5 1 0 1 1 0 1

-39 - 1 0 0 1 1 1

----------- ---------------------

0 0 1 1 ⇐ borrow

06 (0 0 0 1 1 0)



BINARY ARITHMATICS:

Binary Multiplication:

A	B	MULTIPLCATION
0	0	0×0=0
0	1	0×1=0
1	0	1×0=0
1	1	1×1=1

BINARY ARITHMATICS(MULTIPLCATION:

```
x     100
      011
   -------
      100
     100    +
   -------
     1100
```

```
      1001
    x  101
   -------
      1001
     0000
+   1001
   -------
    101101
```

```
1100×11
   1100
    ×11
 -------
   1100
 1100×
 -------
  10100
```

```
         1011.01
×         110.1
 ---------------
          101101
         0000000
        10110100
       101101000
 ---------------
     1001001.001
```

BINARY ARITHMATICS:

Binary Division:

- Division follows same pattern in binary as in decimal
  - **Examples:**
  - **25 ÷ 5 = 5**

```
        1 0 1        Quotient
101 | 1 1 0 0 1
    | - 1 0 1
    | -----------
         0 0 1 0 1
       -     1 0 1
        ------------
             0 0 0     Reminder
```

BINARY ARITHMATICS:

Binary Division:

- Division follows same pattern in binary as in decimal
  - **Examples:**
  - **12 ÷ 2 = 6**

```
          1 1 0      Quotient
     10 | 1 1 0 0
        - 1 0
        -----------
              1 0
          -   1 0
          ------------
              0 0 0    Reminder
```

1'S AND 2'S COMPLEMENT:

1's complement of a given binary number is obtained by replacing 1 instead of 0 and 0 instead of 1

Find out 1's complement of binary number

101101 1's complement is 010010

111001 1's complement is 000110

01010110 1's complement is 10101001

2's complement of a given binary number is obtained by adding1 to 1's completement

Find out 2's complement of binary number

101101 1's complement is 010010 2's complement is 010010+1=010011

111001 1's complement is 000110 2's complement is 000110+1=000111

01010110 1's complement is 10101001 2's comp.. is 10101001+1=10101010

BINARY SUBTRACTION1'S AND 2'S COMPLEMENT:

Binary subtraction using 1's complement

Rules

1. Keep minuend number as it is
2. Make 1's complement of negative number or number to be subtracted
3. Add these two numbers
4. Is carry ? Yes answer is positive , add end round carry
5. No Answer is negative and in 1's complement form

BINARY SUBTRACTION 1'S AND 2'S COMPLEMENT:

Example:

	binary	subtraction using 1,s completement	
12	1100	1100	**keep** minuend number as it is
- 07	- 0111	+ 1000	make 1's complement of subtrahend no.
-----------	-------	---------	
05	0101	1 010 0	Add end round carry
		+ 1	

		0101	

BINARY SUBTRACTION 1'S AND 2'S COMPLEMENT:

Example:

	binary		subtraction using 1's completement	
14	1110		1110	**keep** minuend number as it is
- 11	- 1011		+ 0100	make 1's complement of subtrahend no.
-----------	--------		---------	
03	0011	1	0010	Add end round carry
			+ 1	

			0011	

BINARY SUBTRACTION 1'S AND 2'S COMPLEMENT:

Example:

	binary	subtraction using 1's completement	
25	11001	11001	**keep** minuend number as it is
- 30	- 11110	+00001	make 1's complement of subtrahend no.
-----------	--------	---------	
-05	-00101	11010	No end round carry answer is negative and in 1's complement form

-00101 = -5 in Decimal

BINARY SUBTRACTION 1'S AND 2'S COMPLEMENT:

Examples for practice:

subtraction using 1's completement

1) 17 2) 53 3) 48

- 08 -44 -60

----------- ------ -----

09 09 -12

If both numbers are negative then make 1's complement of both no. and add that ,if carry add that carry and answer is in 1's complement form

-5 ⟶ 0101 1's complement 1010

-7 ⟶ 0111 1's complement+1000

---- ---------

-12 1 0010 + 1 = 0011 1's comp. -1100

BINARY SUBTRACTION1'S AND 2'S COMPLEMENT:

Binary subtraction using 2's complement

Rules

1. Keep minuend number as it is
2. Make 1's complement of negative number or number to be subtracted add 1 to 1's complement to make 2's complement
3. Add these two numbers
4. Is carry ? Yes answer is positive , discard end round carry
5. No Answer is negative and in 2's complement form

BINARY SUBTRACTION 1'S AND 2'S COMPLEMENT:

Example:

	binary		subtraction using 1,s completement
12	1100	1100	**keep** minuend number as it is
- 07	- 0111	+ 1001	make 2's complement of subtrahend no.
-----------	--------	---------	
05	0101	1 010 1	discard end round carry

		0101	

BINARY SUBTRACTION 1'S AND 2'S COMPLEMENT:

Example:

	binary	subtraction using 1's completement	
14	1110	1110	**keep** minuend number as it is
- 11	- 1011	+ 0101	make 2's complement of subtrahend no.
-----------	--------	---------	
03	0011	1 0011	discard end round carry

		0011	

BINARY SUBTRACTION 1'S AND 2'S COMPLEMENT:

Example:

	binary	subtraction using 1's completement	
25	11001	11001	**keep** minuend number as it is
- 30	- 11110	+00010	make 2's complement of subtrahend no.
-----------	-------	---------	
-05	-00101	11011	No end round carry answer is negative and in 2's complement form

00100
+1

0101 =-5

OCTAL NUMBER SYSTEM

3) Octal number system:

Base or radix : 8

Digits: 0-7

3. **Octal Number System:**

   Weight of digits

Weight –	8^{-2}	8^{-1}	.	8^{0}	8^{1}	8^{2}	8^{3}	8^{4}
Weight value	$\frac{1}{8^2}$	$\frac{1}{8^1}$	Octal point	1	8^{1}	8^{2}	8^{3}	8^{4}
	0.01562	0 .125	.	1	8	64	512	4096

OCTAL NUMBER SYSTEM

DECIMAL	BINARY	OCTAL
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	**0111**	**7**
8	1000	10

DECIMAL	BINARY	OCTAL
9	1001	11
10	1010	12
11	1011	13
12	1100	14
13	1101	15
14	1110	16
15	1111	17
16	**10000**	**20**
17	10001	21

DECIMAL	BINARY	OCTAL
18	10010	22
19	10011	23
20	10100	24
21	10101	25
22	10110	26
23	10111	27
24	11000	30
25	**11001**	**31**
26	11010	32

OCTAL ADDITION AND SUBTRACTION:

Example:

1) $(7)_8 + (4)_8$ = $(13)_8$

2) $(17)_8 + (25)_8$ = $(44)_8$

3) $(15)_8 - (06)_8$ = $(07)_8$

4) $(42)_8 - (16)_8$ = $(24)_8$

5) $(43)_8 + (55)_8$ = $(120)_8$

6) $(12)_8 + (71)_8$ = $(103)_8$

7) $(35)_8 + (27)_8$ = $(64)_8$

8) $(24)_8 - (17)_8$ = $(05)_8$

9) $(20)_8 - (15)_8$ = $(03)_8$

10) $(37)_8 + (76)_8$ = $(135)_8$

HEXADECIMAL NUMBER SYSTEM

4) Hexadecimal number system:

Base or radix : 16

Digits: 0-9 and character A,B,C,D,E,F

3. **Hexadecimal Number System:**

Weight of digits

Weight –	16^{-2}	16^{-1}	.	16^{0}	16^{1}	16^{2}	16^{3}
Weight value	$\frac{1}{16^2}$	$\frac{1}{16^1}$	Hexadecimal point	1	16^{1}	16^{2}	16^{3}
	0.0039	0 .0625	.	1	16	256	4096

HEXADECIMAL NUMBER SYSTEM

DECIMAL	BINARY	OCTAL	HEXADECIMAL
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7
8	1000	10	8

DECIMAL	BINARY	OCTAL	HEXADECIMAL
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10
17	10001	21	11

DECIMAL	BINARY	OCTAL	Hexadecimal
18	10010	22	12
19	10011	23	13
20	10100	24	14
21	10101	25	15
22	10110	26	16
23	10111	27	17
24	11000	30	18
25	11001	31	19
26	11010	32	1A

HEXADECIMAL NUMBER SYSTEM

DECIMAL	BINARY	OCTAL	HEXADECIMAL
27	11011	33	1B
28	11100	34	1C
29	11101	35	1D
30	11110	36	1E
31	11111	37	1F
32	100000	40	20
33	100001	41	21
34	100010	42	22

HEXADECIMAL ADDITION AND SUBTRACTION:

Example:

1) $(07)_{16}$ + $(17)_{16}$ ------------- $(1E)_{16}$

2) $(A7)_{16}$ + $(25)_{16}$ --------- $(CC)_{16}$

3) $(1E)_{16}$ + $(06)_{16}$ ------------ $(24)_{16}$

4) $(B9)_{16}$ + $(C8)_{16}$ ---------- $(181)_{16}$

5) $(63)_{16}$ + $(55)_{16}$ --------- $(B8)_{16}$

6) $(A2)_{16}$ - $(71)_{16}$ ------------- $(31)_{16}$

7) $(65)_{16}$ - $(27)_{16}$ --------- $(3D)_{16}$

8) $(24)_{16}$ - $(19)_{16}$ ------------ $(0B)_{16}$

9) $(2F)_{16}$ - $(15)_{16}$ ---------- $(1A)_{16}$

10) $(A7)_{16}$ - $(7D)_{16}$ --------- $(2A)_{16}$

CONVERSION:
1) DECIMAL TO BINARY:

Decimal to Binary

There are two methods

1) SUM OF WEIGHT METHOD
2) DOUBLE DOUBLE METHOD

SUM OF WEIGHT METHOD:

In this method ,a given decimal number is to determine the set of binary weight values whose sum is equal to decimal number

Example

$(9)_{10} = 8+1 = 8+0+0+1$

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

$= 8 + 0 + 0 + 1$

$(9)_{10} = (1001)_2$

CONVERSION:
1) DECIMAL TO BINARY:

Decimal to Binary

Example

2) $(25)_{10} = 16 + 8 + 1 = 16+8+0+0+1$

$= 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$

$= 16 + 8 + 0 + 0 + 1$

$(25)_{10} = (11001)_2$

3) $(58)_{10} = 32+16 + 8 + 2 = 32+16+8+0+2+0$

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

$= 32+16+ 8 + 0 + 2 + 0$

$(58)_{10} = (111010)_2$

CONVERSION:
1) DECIMAL TO BINARY:

Decimal to Binary

Example

4) $(0.625)_{10}$= 0.5+0.125= 0.5+0.125

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

= 0.5+0+0.125

$(0.625)_{10} = (0.1010)_2$

5) $(0.375)_{10}$= 0.25+0.125=

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

= 0.25+0.125

$(58)_{10} = (0.0110)_2$

CONVERSION:
1) DECIMAL TO BINARY:

Double –Double Method:

More popular and convenient

Decimal number is repeatedly divided by the base of binary (2) and reminder of these division are to form binary number.

The reminder of first division as LSB an last division as MSB of binary

Example

$(9)_{10} = (?)_2$

9/2 = 4 with reminder of 1 (LSB)

4/2 = 2 with reminder of 0

2/2 = 1 with reminder of 0

1/2 = 0 with reminder of 1(MSB)

$(9)_{10} = (1001)_2$

CONVERSION:
1) DECIMAL TO BINARY:

Double –Double Method:

Example

$(29)_{10} = (?)_2$

29/2	=	14 with reminder of 1 (LSB)
14/2	=	7 with reminder of 0
7/2	=	3 with reminder of 1
3/2	=	1 with reminder of 1
1/2	=	0 with reminder of 1(MSB)
$(29)_{10}$	=	$(11101)_2$

CONVERSION:
1) DECIMAL TO BINARY:

Double –Double Method:

Example

$(23)_{10} = (?)_2$

23/2	=	11 with reminder of 1 (LSB)
11/2	=	5 with reminder of 1
5/2	=	2 with reminder of 1
2/2	=	1 with reminder of 0
1/2	=	0 with reminder of 1(MSB)
$(23)_{10}$	=	$(10111)_2$

CONVERSION:
1) DECIMAL TO BINARY:

Fractional Decimal to Fractional binary conversion:

In this case successively multiply by base (2). When carry occurs tabulate it separately but do not use in next multiplication

Continue multiplying the fractional part reminder zero

First carry generated is MSB and last carry is LSB

Example 1

$(0.625)_{10} = (?)_2$

$0.625 \times 2 = 1.250$ with carry 1 (MSB)

$0.250 \times 2 = 0.500 =$ with carry 0

$0.500 \times 2 = 1.000$ with carry 1(LSB)

$(0.625)_{10} = (0.101)_2$

CONVERSION:
1) DECIMAL TO BINARY:

Fraction decimal to fraction binary:

Example 2

$(0.3)_{10} = (?)_2$

0.300×2	$= 0.600$	with carry 0 (MSB)
0.600×2	$= 1.200$	with carry 1
0.200×2	$= 0.400$	with carry 0
0.400×2	$= 0.800$	with carry 0
0.800×2	$= 1.600$	With carry 1
0.600×2	$= 1.200$	With carry 1(LSB)

$(0.3)_{10} = (0.010011)_2$

CONVERSION:
2) BINARY TO DECIMAL:

Binary to Decimal:

Sum of weight Method

Digit position of binary number following decimal weights

2^5	2^4	2^3	2^2	2^1	2^0
32	16	8	4	2	1

For conversion

When 1 in digit position add weight of position

When 0 in digit position discard weight of position

Example 1

$(1011)_2 = (?)_{10}$

$(1011)_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$

$= 1 \times 8 + 0 \times 4 + 1 \times 2 + 1 \times 1 = 8+0+2+1 = (11)_{10}$

CONVERSION:
3) DECIMAL TO OCTAL:

Double –Double Method:

More popular and convenient

Decimal number is repeatedly divided by the base of octal (8) and reminder of these division are to form octal number.

The reminder of first division as LSB an last division as MSB of octal

Example

$(654)_{10} = (?)_8$

654/8	=	81 with reminder of 6 (LSB)
81/8	=	10 with reminder of 1
10/8	=	1 with reminder of 2
1/8	=	0 with reminder of 1(MSB)
$(654)_{10}$	=	$(1216)_8$

CONVERSION:
3) DECIMAL TO OCTAL:

Double –Double Method:

Example

$(583)_{10} = (?)_8$

583/8 = 72 with reminder of 7 (LSB)

72/8 = 9 with reminder of 0

9/8 = 1 with reminder of 1

1/8 = 0 with reminder of 1(MSB)

$(583)_{10} = (1107)_8$

CONVERSION:
3) DECIMAL TO OCTAL:

Double –Double Method:

Example

$(27.125)_{10}$ = $(?)_8$

27/8 = 3 with reminder of 3 (LSB)

3/8 = 0 with reminder of 3(MSB)

0.125×8 = 1.00 with carry 1(MSB)

$(27.125)_{10}$ = $(33.1)_8$

CONVERSION:
3) DECIMAL TO OCTAL:

Double –Double Method:

Example

$(342.44)_{10}$ = $(?)_8$

342/8 = 42 with reminder of 6 (LSB)

42/8 = 5 with reminder of 2

5/8 = 0 with reminder 5 (MSB)

0.44 × 8 = 3.52 with carry 3(MSB)

0.52 × 8 = 4.16 with carry 4

0.16 × 8 = 1.28 with carry 1(LSB)

$(342.44)_{10}$ = $(526.341)_8$

CONVERSION:
3) DECIMAL TO OCTAL:

Perform the following conversion:

$(0.875)_{10} = (?)_8$

$(94)_{10} = (?)_8$

$(855.24)_{10} = (?)_8$

$(49)_{10} = (?)_8$

$(108.92)_{10} = (?)_8$

$(497.85)_{10} = (?)_8$

CONVERSION:
4) BINARY TO OCTAL:

Binary to octal conversion

Rules:

1 Group the bits in “ three” starting from binary point

2 For grouping the bits in three, move towards left from binary point for whole integral part

3 For fractional part move towards right

4 In the case if only one or two bits are left add zero to complete group

5 Replace each group of three bits by equivalent octal number

Example

$(100110)_2 = (?)_8$

1 0 0 1 1 0 = $(46)_8$

4 6

CONVERSION:
4) BINARY TO OCTAL:

Example

$(1101.10)_2 \quad =(?)_8$

1 101 . 10

0 0 1 1 0 1 . 1 0 0 = (15.4)8

$(11100.1101)_2 \quad =(?)_8$

1 1 1 0 0 . 1 1 0 1

0 0 1 1 0 0 . 1 1 0 1 0 0 =(14.64)

CONVERSION:
5) OCTAL TO BINARY:

Reverse of binary to octal

One octal digit is converted into its three bit binary equivalent

Example 2

$(57)_8 \quad =(?)_2$

5 7

1 0 1 1 1 1

$(57)_8 \quad =(101111)_2$

(3) $(0.42)_8 \quad =(?)_2$

4 2

1 0 0 0 1 0

$(0.42)_8 \quad =(0.100010)_2$

CONVERSION:
6) OCTAL TO DECIMAL:

In octal each digit corresponds to power of 8

Multiply each octal by its weight and add the resultant product

Example

$(654)_8 \quad =(?)_{10}$

$654 \quad = 6 \times 8^2 + 5 \times 8^1 + 4 \times 8^0$

$= 6 \times 64 + 5 \times 8 + 4 \times 1$

$= 384+40+4 \quad =428$

$(654)_8 \quad =(428)_{10}$

CONVERSION:
7) DECIMAL TO HEXADECIMAL :

Double –Double Method:

More popular and convenient

Decimal number is repeatedly divided by the base of hexadecimal (16) and reminder of these division are to form hexadecimal number.

The reminder of first division as LSB an last division as MSB of hexadecimal

Example

$(654)_{10}$ = $(?)_{16}$

654/16 = 40 with reminder of 14 i.e.(E) (LSB)

40/16 = 2 with reminder of 8

2/16 = 0 with reminder of 2 (MSB)

$(654)_{10}$ = $(28E)_{16}$

CONVERSION:
7) DECIMAL TO HEXADECIMAL:

Double –Double Method:

Example 2

$(498)_{10}$ = $(?)_{16}$

498/16 = 31 with reminder of 2 (LSB)

31/16 = 1 with reminder of 15 i.e.(F)

1/16 = 0 with reminder of 1(MSB)

$(498)_{10}$ = $(1F2)_{16}$

Example 3

$(98)_{10}$ = $(?)_{16}$

98/16 = 6 with reminder of 2 (LSB)

6/16 = 0 with reminder of 6 (MSB)

$(98)_{10}$ = $(62)_{16}$

CONVERSION:
7) DECIMAL TO HEXADECIMAL:

Double –Double Method:

Example 4

$(0.635)_{10}$ = $(?)_{16}$

0.635 × 16 = 10.160 with carry of 10 i.e. A (MSB)

0.160 × 16 = 2.56 with carry of 2

0.56 × 16 = 8.96 with carry of 8 (LSB)

$(0.635)_{10}$ = $(0.A28)_{16}$

Example 5

$(151)_{10}$ = $(?)_{16}$

151/16 = 9 with reminder of 7 (LSB)

9/16 = 0 with reminder of 9 (MSB)

$(151)_{10}$ = $(97)_{16}$

CONVERSION:
8) HEXADECIMAL TO DECIMAL:

In hexadecimal each digit corresponds to power of 16

Multiply each hexadecimal by its weight and add the resultant product

Example

$(FC8)_{16} = (?)_{10}$

$FC8 = F \times 16^2 + C \times 16^1 + 8 \times 16^0$

$= 15 \times 256 + 12 \times 16 + 8 \times 1$

$= 3840 + 192 + 8 = 4040$

$(FC8)_{16} = (4040)_{10}$

CONVERSION:
8) HEXADECIMAL TO DECIMAL:

Example 2

$(0.B2)_{16} = (?)_{10}$

$0.B2 = B \times 16^{-1} + 2 \times 16^{-2}$

$= 11 \times 1/16 + 2 \times 1/256$

$= 0.6875 + 0.00781 = 0.69531$

$(0.B2)_{16} = (0.69531)_{10}$

(4C.C2) =(?)10

$4C.\ C2 = 4 \times 16^1 + 12 \times 16^0 + 12 \times 1/16 + 2 \times 1/256$

$= 64 + 12 + 0.75 + 0.0078$

(4C.C2) =(76.7578)10

CONVERSION:
9) BINARY TO HEXADECIMAL:

Binary to hexadecimal conversion

Rules:

1 Group the bits in “ four” starting from binary point

2 For grouping the bits in four, move towards left from binary point for whole integral part

3 For fractional part move towards right

4 In the case if only one, two or three bits are left add zero to complete group

5 Replace each group of four bits by equivalent hexadecimal number

Example

$(11100110)_2 = (?)_{16}$

1110 0 1 1 0 $= (E6)_{16}$

E 6

CONVERSION:
9) BINARY TO HEXADECIMAL:

Example2

$(110111101)_2$ $=(?)_{16}$

0001 1011 1101 = (1BD)

$(110111101)_2$ $=(1BD)_{16}$

Example 3

$(0.11011)_2$ $=(?)_{16}$

1101 1

1101 1000 = D 8

$(0.11011)_2$ $=(0.D8)_{16}$

CONVERSION:
10) HEXADECIMAL TO BINARY:

Reverse of binary to hexadecimal

One hexadecimal digit is converted into its four bit binary equivalent

Example

$(C7)_{16} = (?)_2$

C 7

110 0 0 1 1 1

$(C7)_{16} = (11000111)_2$

(2) $(0.42)_{16} = (?)_2$

4 2

0 1 0 0 0 0 1 0

$(0.42)_{16} = (0.01000010)_2$

BINARY CODES:

Binary codes are divided into two categories

1 Weighted binary codes

2 Non weighted binary codes

BCD CODES :

Weighted binary codes

Each digit in code have specific weight

Also called 8421 code

BINARY CODES:
BCD CODES:

DECIMAL	BCD
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

Any decimal can be represented in BCD codes as

Decimal 521 can be represented in BCD code

5 → 0101   2 → 0010   1 → 0001

$(521)_{10} = (010100100001)_{8421}$

Convert following decimal into BCD codes

$(8)_{10}$ $(12)_{10}$ $(4.5)_{10}$ $(42.63)_{10}$

GRAY CODE:

Gray code is non weighted code
Not suitable for arithmetic operation
Useful for input/output devices and Analog to digital convertors
Decimal no. is represented in binary formatand only one bit will change in each time
Conversion of Binary number to gray code:
Keep MSB of binary number as it is
Add this bit to next position
Sum bit is as gray code
If carry ,discard it
Continue still last bit

GRAY CODE:

Conversion of Binary number to gray code:

convert 1011 into gray code

Binary	1	0	1	1
Gray code	1	1	1	0

$(1010)_{binary} = (1110)_{gray}$

GRAY CODE:

Conversion of Binary number to gray code:

convert 11011 into gray code

Binary	1	1	0	1	1
	↓	↓	↓	↓	↓
Gray code	1	0	1	1	0

$(11011)_{binary} = (10110)_{gray}$

GRAY CODE:

Conversion of Binary number to gray code:

convert 11011 into gray code

convert the following binary no. into gray code

1 (11111)

2 (101010)

3 (1100)

4 (0111)

5 (1110)

GRAY CODE:

Conversion of gray code to Binary number :

Keep MSB of gray as it is

Add this MSB bit diagonally to bit nearest

Ignore carry

Continue till LSB

Example

Convert (1100)gray to binary

Gray 1 1 0 0

Binary 1 0 0 0

(1100)Gray= (1000)binary

GRAY CODE:

Conversion of gray code to Binary number :

Convert (10101)gray to binary

Gray 1 0 1 0 1

Binary 1 1 0 0 1

$(1111)_{Gray} = (?)_{binary}$

$(1101)_{Gray} = (?)_{binary}$

EXCESS-3 CODE:

Non weighted code

Also known as XS-3 and used to express decimal number

To get decimal number into exess -3 form, add 3 to each decimal digit before converting to binary

XS-3is 3 more than 8421 code

Decimal to ex-3 code

To encode decimal number to ex-3 code, add three to each decimal digit

In this case if carry then don’t carry to next higher place

Convert each bot into 4 bit binary euivalent

Example

$(2)_{10}$ =2+3 =5= $(0101)_{EX\text{-}3}$

EXCESS-3 CODE:

Example

$(9)_{10}$ =9+3 =12= $(1100)_{EX\text{-}3}$

$(25)_{10}$ =2+3=5 and 5+3=8=$(01011000)_{EX\text{-}3}$

$(89)_{10}$ = $(1011\ 1100)_{EX\text{-}3}$

$(79)_{10}$ =$(?)_{EX\text{-}3}$