UNIT -1 NUMBER SYSTEMS AND CODES (covers 20 marks)

Difference between Analog and Digital Signal

Sl. No.	Analog Signals	Digital Signals
1	Continuous signals	Discrete signals
2	Represented by sine waves	Represented by square waves
3	Human voice, natural sound, analog electronic devices are a few examples	Computers, optical drives, and other electronic devices
4	Continuous range of values	Discontinuous values
5	Records sound waves as they are	Converts into a binary waveform
6	Only used in analog devices	Suited for digital electronics like computers, mobiles and more

Types of number system. (List the types of number system)

In general, there are four different number system they are:

1) Decimal Number System
2) Binary Number System
3) Octal Number System
4) Hexadecimal Number System

DECIMAL NUMBER SYSTEM: (Explain the Decimal number system with example)

- Decimal number system has 10 different digits/ symbols 0, 1, 2, 3, 4, 5, 6, 7, 8 and 9.
- Base of decimal number system is 10.
- Example: $193_{(10)}$, $1256.78_{(10)}$, $3.142_{(10)}$ etc.

The weights of digits in decimal number is as shown below:

Weight:	10^3	10^2	10^1	10^0		10^{-1}	10^{-2}
Digits:	1	2	5	6	.	7	8
Value:	1×10^3	2×10^2	5×10^1	6×10^0	.	7×10^{-1}	8×10^{-2}

MSD LSD

BINARY NUMBER SYSTEM: (Explain the Binary number system with example)

- Binary number system has 2 different digits/ symbols 0 and 1.
- Base of binary number system is 2.
- Each digit in binary number system is known as "***bit***".
- Example: $101_{(2)}$, $11011_{(2)}$, $110.101_{(2)}$ etc.

The weights of digits in binary number is as shown below:

Weight:	2^3	2^2	2^1	2^0		2^{-1}	2^{-2}
Digits:	1	0	1	1	.	0	1
Value:	1×2^3	0×2^2	1×2^1	1×2^0	.	0×2^{-1}	1×2^{-2}

MSB LSB

OCTAL NUMBER SYSTEM: (Explain the Octal number system with example)

- Octal number system has 8 different digits/ symbols 0, 1, 2, 3, 4, 5, 6 and 7.
- Base of octal number system is 8.
- Example: $256_{(8)}$, $432.35_{(8)}$, $250.06_{(8)}$, $125.56_{(8)}$ etc.

The weights of digits in octal number is as shown below:

Weight:	8^2	8^1	8^0		8^{-1}	8^{-2}
Digits:	4	3	2	.	3	5
Value:	4×8^2	3×8^1	2×8^0	.	3×8^{-1}	5×8^{-2}

MSD LSD

HEXADECIMAL NUMBER SYSTEM: (Explain the Hexadecimal number system with example)

- Hexadecimal number system has 16 different digits 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E and F.
- Base of hexadecimal number system is 16.
- Each digit in hexadecimal number system is known as "***nibble***".
- Example: $256_{(16)}$, $432.35_{(16)}$, $AB50.06_{(16)}$ etc.

The weights of digits in hexadecimal number is as shown below:

Weight:	16^2	16^1	16^0		16^{-1}	16^{-2}
Digits:	A	9	3	.	2	5
Value:	10×16^2	9×16^1	3×16^0	.	2×16^{-1}	5×16^{-2}

MSD LSD

Write steps to convert binary number to decimal number with example:

Each digit must be multiplied by its weight and the resulting products are added.

Problem1: Convert $10111.110_{(2)}$ into decimal

Answer: $10111.110_{(2)} = (1X2^4) + (0X2^3) + (1X2^2) + (1X2^1) + (1X2^0) + (1X2^{-1}) + (1X2^{-2}) + 0$

$= (1 X 16) + (0 X 8) + (1 X 4) + (1 X 2) + (1X1) + (1 X 0.5) + (1X0.25)$

$= 16+0+4+2+1+0.5+0.25$

$\mathbf{10111.110_{(2)} = 23.75_{(10)}}$

Problem2: Convert $110111.11_{(2)}$ into decimal

Answer: $110111.110_{(2)} = (1X2^5) + (1X2^4) + (0X2^3) + (1X2^2) + (1X2^1) + (1X2^0) + (1X2^{-1}) + (1X2^{-2})$

$= (1 X 32) + (1 X 16) + 0 + (1 X 4) + (1 X 2) + (1X1) + (1 X 0.5) + (1X0.25)$

$= 32+16+0+4+2+1+0.5+0.25$

$\mathbf{10111.110_{(2)} = 55.75_{(10)}}$

Table showing relationship between decimal, binary, octal and 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
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

Write steps to convert octal number to decimal number:

Each digit must be multiplied by its weight and the resulting products are added.

Problem1: Convert $1523_{(8)}$ into decimal

Answer: $1523_{(8)} = (1X8^3) + (5X8^2) + (2X8^1) + (3X8^0)$

$= (1 X 512) + (5 X 64) + (2 X8) + (3 X1)$

$= 512+320+16+3$

$\mathbf{1523_{(8)} = 851_{(10)}}$

Problem2: Convert $237.56_{(8)}$ into decimal

Answer: $237.56_{(8)} = (2X8^2) + (3X8^1) + (7X8^0) + (5X8^{-1}) + (6X8^{-2})$

$= (2 X 64) + (3 X 8) + (7 X1) + (5 X 0.125) + (6 X0.0156)$

$= 128 + 24 + 7 +0.625 + 0.0936$

$\mathbf{237.56_{(8)} = 159.7186_{(10)}}$

Write steps to convert Hexadecimal number to decimal number:

Each digit must be multiplied by its weight and the resulting products are added.

Problem1: Convert $256_{(16)}$ into decimal

Answer: $256_{(16)} = (2X16^2) + (5X16^1) + (6X16^0)$

$= (2 X 256) + (5 X16) + (6 X 1)$

$= 512 + 80 + 6$

$\mathbf{256_{(16)} =598_{(10)}}$

Problem 2: Convert 7AC. $5_{(16)}$ into decimal

Answer: $7AC.5_{(16)} = (7 X 16^2) + (10 X 16^1) + (12 X 16^0) + (5 X 16^{-1})$

$= (7 X 256) + (10 X16) + (12 X 1) + (5 X .0625)$

$= 1792 + 160 + 12 +.03125$

$\mathbf{7AC.5_{(16)} =1964.3125_{(10)}}$

Write steps to convert binary number to octal number:

1) **Make a group of 3-bits starting from LSB for integer part and from MSB for fractional part.**
2) **Add zeroes at the end to make group f 3-bits, if required.**
3) **Write octal equivalent for each group of 3-bits.**

Problem1: Convert $11011011.01101_{(2)}$ into octal

Answer:

11011011 . 01101

011|011|011 . 011|010

Add Zeroes

By witting

Add Zeroes to make group of 3-bits

Octal equivalent for each group of 3-bit to get octal equivalent of given binary number

$11011011.01101_{(2)} = 333.32_{(8)}$

Problem 2: Convert $101110.0111_{(2)}$ into octal

Answer:

$101110 . 0111_{(2)}$

101|110 . 011|100

Add Zeroes to make group of 3-bits

By witting octal equivalent for each group of 3-bit to get octal equivalent of given binary number

$101110.0111_{(2)} = 56.34_{(8)}$

Write steps to convert binary number to hexadecimal number:

1) **Make a group of 4-bits starting from LSB for integer part and from MSB for fractional part.**
2) **Add zeroes at the end to make group of 4-bits, if required.**
3) **Write hexadecimal equivalent for each group of 4-bits.**

Problem1: Convert $11011011.01101_{(2)}$ into hexadecimal

Answer:

11011011 . 01101

1101|1011 . 0110|1000

Add Zeroes to make group of 4-bits

By witting hexadecimal

equivalent for each group of 4-bit to get required hexadecimal number

$11011011.01101_{(2)} = DB.68_{(16)}$

Problem 2: Convert $1101011011011_{(2)}$ into hexadecimal

Answer:

$1101011011011_{(2)}$

Add Zeroes to make group of 4-bits

0001|1010|1101|1011 $_{(2)}$

By witting hexadecimal equivalent for each group of 4-bit to get required hexadecimal number

$1101011011011_{(2)} = 1ADB_{(16)}$

Write steps to convert hexadecimal number to binary number:

1) **Write 4-bit binary equivalent for each hexadecimal digit.**
2) **Remove zeroes at the end.**

Problem 1: Convert $13AF_{(16)}$ into binary.

Answer: **1 3 A F**

0001 0011 1010 1111

Remove extra Zeroes

$13AF_{(16)} = 1001110101111_{(2)}$

Problem 2: Convert $7AC.59_{(16)}$ into binary.

Answer: **7 A C. 5 9**

0111 1010 1100 . 0101 1001

$7AC.59_{(16)} = 11110101100.0101001_{(2)}$

Write steps to convert octal number to binary number:

1) **Write 3-bit binary equivalent for each octal digit.**
2) **Remove zeroes at the end.**

Problem 1: Convert $1367_{(8)}$ into binary.

Answer: **1 3 6 7**

001 011 110 111

Remove extra Zeroes

$1367_{(8)} = 1011110111_{(2)}$

Problem 2: Convert $256.2_{(8)}$ into binary.

Answer: **2 5 6 . 2**

010 101 110 011 010 Remove extra Zeroes

Remove extra Zeroes

$256.12_{(8)} = 10101110.01101_{(2)}$

Write steps to convert octal number to hexadecimal number:

1) **Write binary equivalent for each octal digit.**
2) **Convert binary number obtained in step 1 to hexadecimal number.**

Problem 1: Convert $256.2_{(8)}$ into hexadecimal.

Answer: **2 5 6 . 2**

010 101 110 011 010 Remove extra Zeroes

Remove extra Zeroes

$256.12_{(8)}$ = $10101110.01101_{(2)}$, convert this binary number to hexadecimal number.

1010|1110.0110|1000 Add extra Zeroes to make 4bit group

By witting hexadecimal equivalent for each group of 4-bit to get required hexadecimal number

$256.12_{(8)}$ = AE . $38_{(16)}$

Write steps to convert hexadecimal number to octal number:

1) **Write binary equivalent for each hexadecimal digit.**
2) **Convert binary number obtained in step 1 to octal number.**

Problem 1: Convert AE . 38 $_{(16)}$ into octal.

Answer: **A E . 3 8**

1010 1110 0011 1000 Remove extra Zeroes

AE . 38 $_{(16)}$ = $10101110.00111_{(2)}$, convert this binary number to octal number.

010|101|110.001|110 Add extra Zeroes to make 4bit group

By witting octal equivalent for each group of 3-bit to get required octal number

$AE.38_{(16)}$ =256.16 $_{(8)}$

Write steps to convert decimal number to any other number system:

For integer part (Successive division method)

1) **Divide the integer part by base of the required number system. Record the Quotient and remainder.**
2) **Consider Quotient as new integer part and repeat step 1 until Quotient becomes 0.**
3) **List the remainders in upward direction.**

For fractional part (Successive multiplication method)

1) **Multiply the fractional part by base of the required number system. Record integer part as carry.**
2) **Consider fractional part as new fractional part and repeat step 1 until required number of digits are obtained.**
3) **List the carry in downward direction.**

Problem 1: Convert 12.125 decimal number to binary

Answer: Integer part conversion

2	12	
2	6	0
2	3	0
	1	1

$12 = 1100_{(2)}$

Fractional part conversion

$0.125 \times 2 = 0.25$	0 (MSB)
$0.25 \times 2 = 0.5$	0
$0.5 \times 2 = 1.0$	1 (LSB)

$0.125 = 0.001_{(2)}$

Therefore, $\boxed{\mathbf{12.125 = 1100.001_{(2)}}}$

Problem 2: Convert 12.125 decimal number to octal.

Answer: Integer part conversion

8	12	
	1	4

$12 = 14_{(8)}$

Therefore, $\boxed{\mathbf{12.125 = 14.1_{(8)}}}$

Fractional part conversion

$0.125 \times 8 = 1.0$	1

$0.125 = 0.1_{(8)}$

Problem 3: Convert 125.125 decimal number to hexadecimal.

Answer: Integer part conversion

16	125	
	7	13(D)

$125 = 7D_{(16)}$

Fractional part conversion

$0.125 \times 16 = 2.0$	2

$0.125 = 0.2_{(16)}$

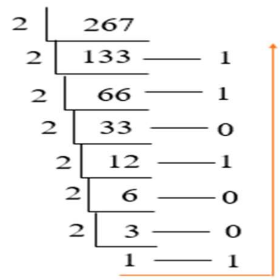
Therefore, $\boxed{\mathbf{125.125 = 7D.2_{(16)}}}$

Problem 4: Convert 267.25 decimal number to (i) binary, (ii) octal and (iii) hexadecimal.

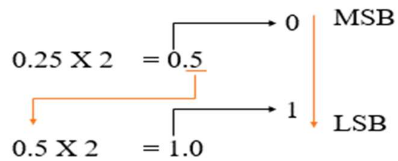
Answer:

(i) BINARY CONVERSION

Integer part conversion



Fractional part conversion



Therefore , 267.25 =11001011.01$_{(2)}$

(ii) OCTAL CONVERSION

267.25 =11001011.01$_{(2)}$ =011 001 011.010$_{(2)}$

$\boxed{\textbf{267.25 =613.2}_{(8)}}$

(ii) HEXADECIMAL CONVERSION

267.25 =11001011.01$_{(2)}$ =1100 1011.0100$_{(2)}$

$\boxed{\textbf{267.25 =CB.6}_{(16)}}$

BINARY ADDITION

The binary number system uses only two digits 0 and 1. The four basic rules for binary addition are

1) 0+0=0
2) 0+1=1
3) 1+0=1
4) 1+1=10

Perform addition for following numbers:

1) 11101 and 11011.

Solution:

```
  1 1 1 1      ←carry
  1 1 1 0 1
  1 1 0 1 1
1 1 1 0 0 0
```

2) 10101 and 110110.

Solution:

```
   1   1        ←carry
    1 0 1 0 1
  1 1 0 1 1 0
1 0 0 1 0 1 1
```

BINARY SUBTRACTION

Rules for binary subtraction are

1) 0 – 0 = 0
2) 1 – 0 = 1
3) 1 – 1 = 0
4) 0 – 1 = 1 with barrow 1

Subtract the following numbers:

1) 101 from 1001

Solution:

```
1 0 0 1
  1 0 1
  1 0 0
```

2) 111 from 1000

Solution:

```
1 0 0 0
  1 1 1
0 0 0 1
```

3) 1001 from 1000

Solution:

```
   1 0 0 0
   1 0 0 1
11 1 1 1
```

BINARY MULPLICATION

1) 10001× 101

```
  10001× 101
  10001
 00000
10001
1010101
```

BINARY DIVISION

11010 ÷ 101

```
101)11010(101
    101
    00110
      101
      001
```

Quotient = 101

Remainder = 001

1's and 2's complement of binary number:

1's and 2's complement of binary numbers are used to represent signed binary numbers.

1's complement of binary number:

The 's complement of binary number is obtained by changing all 1s to 0s and all 0s to 1s as shown below:

1 0 1 1 0 0 1 0 Binary number

0 1 0 0 1 1 0 1 1's complement

2's complement of binary number:

The 2's complement of binary number is obtained by adding 1 to the LSB of 1's complement.

2's complement = (1's complement) +1

Problem 1: Find 2's complement of 10110010

Answer: 1 0 1 1 0 0 1 0 Binary number

0 1 0 0 1 1 0 1 1's complement

\+ 1 Add 1

0 1 0 0 1 1 1 0 2's complement

Application of 1's complement and 2's complement:

1's and 2's complement of binary number are used to represent signed binary numbers.

Representation of signed binary numbers using 1's and 2's complement:

Binary number	1's complement value	2's complement value
000	0	0
001	1	1
010	2	2
011	3	3
100	-3	-4
101	-2	-3
110	-1	-2
111	-0	-1

Binary Subtraction using 1's complement addition:

Steps to perform (A-B):

1. First take 1's complement of B
2. Then add 1's complement of B to A.
3. If there is a carry, then result is positive add carry to result to get final result.
4. If there is no carry, then result is negative and take 1's complement of the result.

---

1. Perform 110110 – 1011 using 1's complement addition

Answer:

Step 1: Make both the numbers equal in number of bits i.e., 110110 - 010110

Step 2: Take 1's complement of 2nd number 010110

1's complement of 010110 is→101001

Step 3: Add 1st number and 1's complement of 2nd number

```
  1 1 0 1 1 0
  1 0 1 0 0 1
1 0 1 1 1 1 1
```

Step 4: Add carry to result → +1

1 0 0 0 0 0

110110 – 1011 = 100000(2)

---

2. Perform 1100 – 10110 using 1's complement addition

Answer:

Step 1: Make both the numbers equal in number of bits i.e., 01100 -10110

Step 2: Take 1's complement of subtrahend 10110

1's complement of 10110 is → 01001

Step 3: Add 1st number and 1's complement of 2nd number

```
1         ←carry
0 1 1 0 0
0 1 0 0 1
1 0 1 0 1
```

Step 4: There is no carry, hence result is negative. Take 1's complement of result

1's complement of 10101→ 01010

1100-10110 = 01010(2)

Binary Subtraction using 2's complement addition:

Steps to perform (A-B):

1. First take 2's complement of B
2. Then add 2's complement of B to A.
3. If there is a carry, then drop the carry and the result is positive.
4. If there is no carry, then result is negative and take 2's complement of the sum.

---

1. Perform 110110 – 10110 using 2's complement addition

Answer:

Step 1: Make both the numbers equal in number of bits i.e., 110110 - 010110

Step 2: Take 2's complement of 2nd number 010110

1's complement of 010110 is → 101001

+1

2's complement of 010110 is → 101010

Step 3: add 1st number and 2's complement of 2nd number

```
 1 1  1 1      ←carry
   1 1 0 1 1 0
   1 0 1 0 1 0
 1 1 0 0 0 0 0
```

Step 4: there is carry, hence neglect the carry. **110110-010110=100000**

---

2. Perform 1100 – 10110 using 2's complement addition

Step 1: Make both the numbers equal in number of bits i.e., 01100 -10110

Step 2: Take 2's complement of subtrahend 10110

1's complement of 10110 is → 01001

+1

2's complement of 010110 is → 01010

Step 3: Add minuend and 2's complement of subtrahend

```
1          ←carry
0 1 1 0 0
0 1 0 1 0
1 0 1 1 0
```

Step 4: There is no carry, hence result is negative. Take 2's complement of result

1's complement of 10110 → 01001

+1

2's complement of 10110 → **01010**

1100 – 10110 = $01010_{(2)}$