BCD Code → Binary Coded Decimal

In BCD code, each digit of a decimal no is represented by its binary equivalent instead of converting the entire decimal value to a binary number. This makes the conversion process easier.

BCD 4-bit →

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

In BCD 4-bits, all decimal digits are represented by 4 bits.

for Eg:-

42_{10} = 0100 (4) 0010 (2)

= 01000010 in BCD

When four bits are used, 2^4 (16) combinations are possible, but only first 10 of these are used to represent decimal digits.

BCD coding does not use remaining six arrangements. That is, 1010 does not represent 10_{10} in BCD.

$10_{10} = \underset{1}{0001}\ \underset{0}{0000} = 00010000$ in BCD

4 Bit BCD coding system can represent only decimal numbers because four bits are not sufficient to represent the various characters used by a computer.

6-Bit BCD → Instead of using four bits with only 16 possible characters, computer designers use 6 bits BCD code to represent characters.

In 6-Bit BCD code, the 4 BCD numeric place positions are same & 2 additional zone positions are added. With 6 bits, 2^6 (64) different characters can be represented. which is sufficient to code the decimal digits (10), alphabets (26) & other special characters (28).

Eg:- Write the binary digits used to record the word BASE in BCD.

B = 110010

A = 110001

S = 010010

E = 110101

Hence - B A S E

110010 110001 010010 110101.

EBCDIC → The major Problem with BCD code is that it can represent only 64 (2^6) different characters, which is not sufficient for decimal numbers (10), lowercase (small) letters (26), uppercase (capital) letters (26) + a large number of other special characters (28+).

Hence BCD code is extended from a 6-bit code to 8-bit code. The zone of 2 bits is expanded to 4 bits. The resulting code is known as Extended Binary-Coded Decimal Interchange code (EBCDIC).

In this code, 2^8 (256) characters are possible to represent.

Since EBCDIC is an 8-bit code, we can divide it easily into two 4-bit groups and use one hexadecimal digit for representing

each of these.

Eg:- Write the EBCDIC coding for the word 'BIT' in both binary and hexadecimal notations.

Sol^{n}.

B - 1100 0010 .

I - 1100 1001

T - 1110 0011

Hence in binary notations EBCDIC code of 'BIT' will be -

11000010	11001001	11100011
B	I	T

In hexadecimal -

C2	C9	E3
B	I	T

ASCII → ASCII is widely used computer code. ASCII stands for 'American Standard code for Information Interchange'.

ANSI (American National Standards Institute) published ASCII Standards in 1963.

ASCII is of two types - ASCII-7 and ASCII-8.

ASCII-7 is a 7 bit code that can represent 2^7 (128) different characters

ASCII-8 is an extended version of ASCII-7. It is an 8-bit code that can represent 2^8 (256) different characters. It adds the additional bit to left of the 7th bit (leftmost bit) of ASCII-7 codes.

Eg:- Write the ASCII-7 coding for the word 'GIRL' in both binary and hexadecimal notations.

Sol^n - In ASCII-7

G - 1000111

I - 1001001

R - 1010010

L - 1001100

Hence the ASCII-7 coding in Binary -

$\underbrace{1000111}_{G}$ $\underbrace{1001001}_{I}$ $\underbrace{1010010}_{R}$ $\underbrace{1001100}_{L}$

In hexadecimal -

49	49	52	4C
G	I	R	L

Eg:- Write the ASCII-8 coding for the word 'John' in both binary and hexadecimal notations.

Sol^n - In ASCII-8 Binary notations -

J - 0100 1010
o - 0110 1110
h - 0110 1000
n - 0110 1110

Hence, the ASCII-8 coding is -

01001010	01101110	01101000	01101110
J	o	h	n

In hexadecimal -

4A	6F	68	6E
J	o	h	n.