Digital Logic Design :

Lecture 8

Binary Coded Decimal (BCD) code :

BCD code mean each decimal digit, 0 through 9, is represented by a binary code of four bits.

Convert decimal number '35' to BCD :

$$\begin{array}{cc} 3 & 5 \\ \downarrow & \downarrow \\ = 0011 & 0101 \end{array}$$

BCD code for decimal '98' :

$$\begin{array}{cc} 9 & 8 \\ \downarrow & \downarrow \\ = 1001 & 1000 \end{array}$$

Converting BCD codes to decimal

$$\begin{array}{cccc} \underbrace{1001} & \underbrace{0100} & \underbrace{0111} & \underbrace{0000} \\ \downarrow & \downarrow & \downarrow & \downarrow \\ 9 & 4 & 7 & 0 \end{array}$$

Invalid codes in BCD codes

** Six combinations : 1010, 1011, 1100, 1101, 1110, 1111.

BCD addition

a) 0011 + 0100

$$\begin{array}{r} 0011 \\ 0100 \\ \hline 0111 \end{array} \qquad \begin{array}{r} 3 \\ +\ 4 \\ \hline 7 \end{array}$$

b) 0010 0011 + 0001 0101

$$\begin{array}{rr} 0010 & 0011 \\ 0001 & 0101 \\ \hline 0011 & 1000 \end{array} \qquad \begin{array}{r} 23 \\ +\ 15 \\ \hline 38 \end{array}$$

c) 1001 + 0100

$$\begin{array}{r} 1001 \\ +\ 0100 \\ \hline 1101 \\ +\ 0110 \\ \hline \underbrace{0001}_{1}\ \underbrace{0011}_{3} \end{array} \qquad \begin{array}{r} 9 \\ +4 \\ \hline 13 \end{array}$$

1101 → Invalid BCD number (>9)

+ 0110 Add 6

d) 1001 + 1001

$$\begin{array}{r} 1001 \\ +\ 1001 \\ \hline 10010 \\ +\ 0110 \\ \hline \underbrace{0001}_{1}\ \underbrace{1000}_{8} \end{array} \qquad \begin{array}{r} 9 \\ +9 \\ \hline 18 \end{array}$$

10010 → Invalid because carry generated.

+ 0110 Add 6

e) 00010110 + 00010101

$$\begin{array}{rl} & 00010110 \\ + & 00010101 \\ \hline & 0010\underbrace{1011}_{} \end{array} \qquad \begin{array}{r} 16 \\ +\ 15 \\ \hline 31 \end{array}$$

$\longrightarrow$ (invalid BCD >9)

$$\begin{array}{rl} + & 0110 \\ \hline & \underbrace{0011}_{3}\ \underbrace{0001}_{1} \end{array}$$

(add 6)

f) 0110 0111 + 0101 0011

$$\begin{array}{r} 0110\ 0111 \\ 0101\ 0011 \\ \hline \underbrace{1011}_{}\ \underbrace{1010}_{} \\ +\ 0110 + 0110 \\ \hline \underbrace{0001}_{1}\ \underbrace{0010}_{2}\ \underbrace{0000}_{0} \end{array} \qquad \begin{array}{r} 67 \\ +\ 53 \\ \hline 120 \end{array}$$

$\Longrightarrow$ Invalid >9

g)

$$\begin{array}{r} 1001\ \ 1001 \\ 1000\ \ 1001 \\ \hline 0001\ \underbrace{0010}_{}\ \underbrace{0010}_{} \\ +\ 0110 + 0110 \\ \hline \underbrace{0001}_{\cancel{3}\ 1}\ \underbrace{1000}_{8}\ \underbrace{1000}_{8} \end{array} \qquad \begin{array}{r} 99 \\ +\ 89 \\ \hline 188 \end{array}$$

$\Longrightarrow$ Invalid because carry generated

The excess-3 code :

Excess-3 is a digital code related to BCD that is derived by adding 3 to each decimal digit and then converting the result of that addition to 4-bit binary.

The excess-3 code for decimal 5 is,

$$\begin{array}{r} 5 \\ +\ 3 \\ \hline 8 \end{array} \longrightarrow 1000$$

The excess 3 code for decimal 7 is,

$$\begin{array}{r} 7 \\ +\ 3 \\ \hline 10 \end{array} \longrightarrow 1010$$

Decimal	BCD	Excess-3
0	0000	0011
1	0001	0100
2	0010	0101
3	0011	0110
4	0100	0111
5	0101	1000
6	0110	1001
7	0111	1010
8	1000	1011
9	1001	1100

six invalid combinations in excess-3 codes are 0000, 0001, 0010, 1101, 1110, 1111.

Convert decimal 928 to excess 3

$$\begin{array}{ccc} 9 & 2 & 8 \\ +3 & +3 & +3 \\ \hline 12 & 5 & 11 \\ \downarrow & \downarrow & \downarrow \\ 1100 & 0101 & 1011 \end{array} \quad \text{Excess - 3}$$

Self complementing property of excess-3 code :

9's complement of 7 is 9-7 = 2.

Excess - 3 code for 7 is 1010.

1's complement of 1010 is 0101.

Excess - 3 code for 2 is 0101.

usefulness of 9's compliment :

9's complement of 28 is 99 - 28 = 71

now, 51 - 28 = 23

$$\begin{array}{r} 51 \\ +71 \\ \hline 122 \\ +1 \\ \hline 23 \end{array}$$

+71 → 9's complement of 28

dis-carded