# Lecture session 5_ UNIT-3_Continuation

# Unit-3-COMBINATIONAL LOGIC
# CODE CONVERSION

**By**
**V.GEETHA**
**ASSISTANT PROFESSOR/EEE**
**SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY**
**CHENNAI-119**

# Code Converter

# Binary to Gray Code Converter

The Gray Code is unweighted and is not an arithmetic code: that is, there are no specific weights asssigned to the bit positions. The important feature of the Gray Code is that it exhibits only a single bit change from one code word to the next in sequence. This property is important in many applications, such as shaft position encoders, where error susceptibility increases with the number of bit changes between adjacent numbers in a sequence.

To convert a binary number to a Gray Code number, the following rules apply.

1. The most significant digit ( Left Most Bit ) in the Gray Code is the same as the corresponding digit in the binary number.

2. Going from left to right, add each adjacent pair of binary digits to get the next Gray code digit, regradless carries.

**For instance** - Let us convert the binary number 1010 to Gray Code.

**Step 1** - The left most Gray digit is the same as the left most binary

| | | | | |
|---|---|---|---|---|
| 1 | 0 | 1 | 0 | Binary |
| ↓ | | | | |
| 1 | | | | Gray |

**Step 2** - Add the left most binary digit to the adjacent one.

| | | | | |
|---|---|---|---|---|
| 1 | + 0 | 1 | 0 | Binary |
| | ↓ | | | |
| 1 | 1 | | | Gray |

**Step 3** - Add the next adjacent pair

| | | | | |
|---|---|---|---|---|
| 1 | 0 | + 1 | 0 | Binary |
| | | ↓ | | |
| 1 | 1 | 1 | | Gray |

**Step 4** -Add the last adjacent pair

| | | | | |
|---|---|---|---|---|
| 1 | 0 | 1 | + 0 | Binary |
| | | | ↓ | |
| 1 | 1 | 1 | 1 | Gray |

The conversion is now complete and the Gray Code is 1111.

# Steps to design the converter

1. Design a converter by the following procedures:

a. Write down the truth table of both input and output bits of the converter.

b. Apply Karnaugh Map to look for the minimized logic expression for the output bits.

c. Implement the logic gates by using Circuit Maker.

Example:

For Binary to Gray Code Converter, binary bits are input and gray code bits are output. So first write the truth table for binary bits and gray code. Then k-map for the all bits of gray code, find the simplified expression for each bit of gray code. Then design the logical circuit.

# Truth Table

| Natural-binary code | | | | Gray code | | | |
|---|---|---|---|---|---|---|---|
| B3 | B2 | B1 | B0 | G3 | G2 | G1 | G0 |
| | | | | | | | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |

# K-Map for each bit of Gray code

### K-map for $G_0$

| $B_3B_2$ \ $B_1B_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 1 | 0 | 1 |
| 01 | 0 | 1 | 0 | 1 |
| 11 | 0 | 1 | 0 | 1 |
| 10 | 0 | 1 | 0 | 1 |

$$G_0 = B_1'B_0 + B_1B_0'$$

$$G_0 = B_0 \oplus B_1$$

### K-map for $G_1$

| $B_3B_2$ \ $B_1B_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 0 | 1 | 1 |
| 01 | 1 | 1 | 0 | 0 |
| 11 | 1 | 1 | 0 | 0 |
| 10 | 0 | 0 | 1 | 1 |

$$G_1 = B_1'B_2 + B_1B_2'$$

$$G_2 = B_1 \oplus B_2$$

K-map for $G_2$

| $B_3 B_2$ \ $B_1 B_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 0 | 0 | 0 |
| 01 | 1 | 1 | 1 | 1 |
| 11 | 0 | 0 | 0 | 0 |
| 10 | 1 | 1 | 1 | 1 |

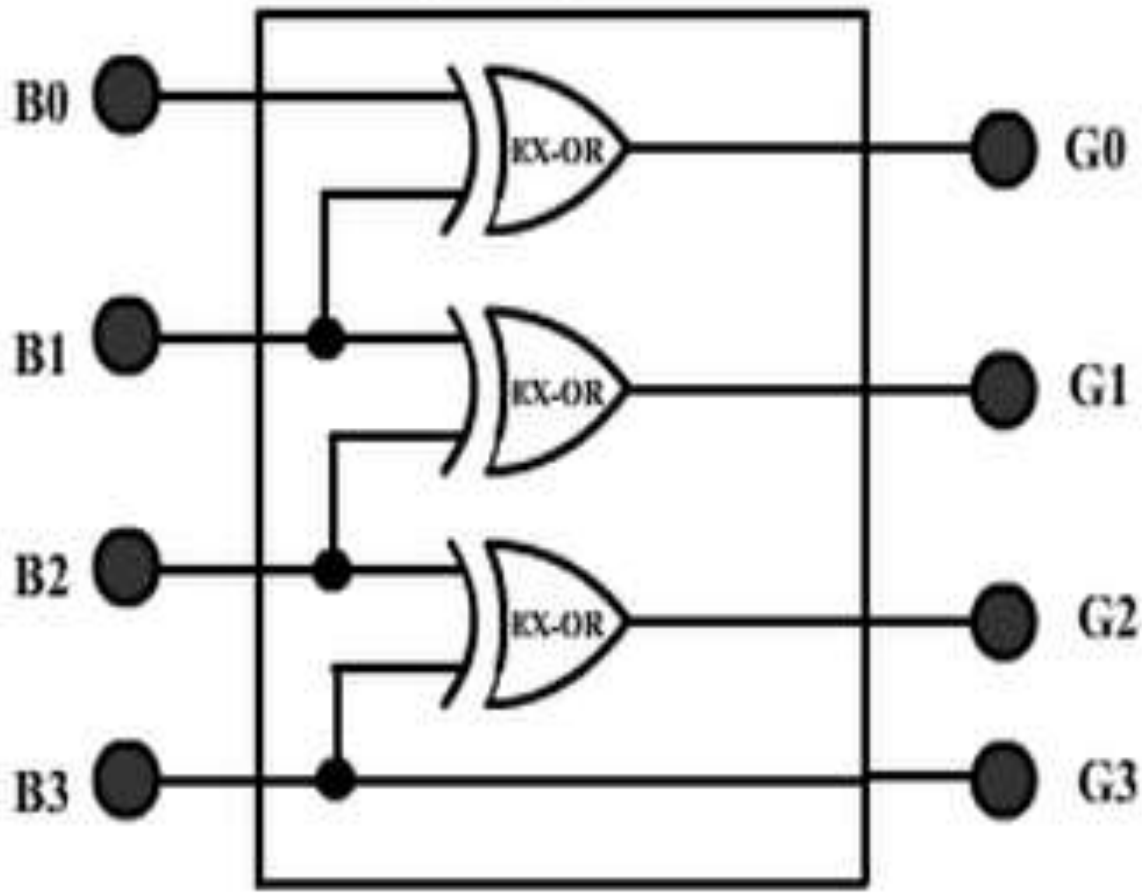
And

G3 = B3

$$G_2 = B_3' B_2 + B_3 B_2'$$

$$G_2 = B_2 \oplus B_3$$

**Binary to Gray Converter**



# Gray to Binary Converter

## Truth Table

| Gray code | | | | Natural-binary code | | | |
|---|---|---|---|---|---|---|---|
| G3 | G2 | G1 | G0 | B3 | B2 | B1 | B0 |
| | | | | | | | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 |

## K-map for $B_0$

| $G_3G_2$ \ $G_1G_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 1 | 0 | 1 |
| 01 | 1 | 0 | 1 | 0 |
| 11 | 1 | 0 | 1 | 0 |
| 10 | 0 | 1 | 0 | 1 |

## K-map for $B_1$

| $G_3G_2$ \ $G_1G_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 0 | 1 | 1 |
| 01 | 1 | 1 | 0 | 0 |
| 11 | 0 | 0 | 1 | 1 |
| 10 | 1 | 1 | 0 | 0 |

$$B_0 = G_2G_1'G_0' + G_2'G_1G_0' + G_2'G_1'G_0 + G_2G_1G_0$$

$$= G_0'(G_1'G_2 + G_1G_2') + G_0(G_1G_2 + G_1'G_2')$$

$$= G_0'(G \oplus G_2) + G_0(G_1 \oplus G_2)' = G_0 \oplus G_1 \oplus G_2$$

$$B_1 = G_3'G_2'G_1 + G_3'G_2G_1' + G_3G_2G_1 + G_3G_2'G_1'$$

$$= G_3'(G_2 \oplus G_1) + G_3(G_2 \oplus G_1)'$$

$$= G_1 \oplus G_2 \oplus G_3$$

**K-map for $B_2$**

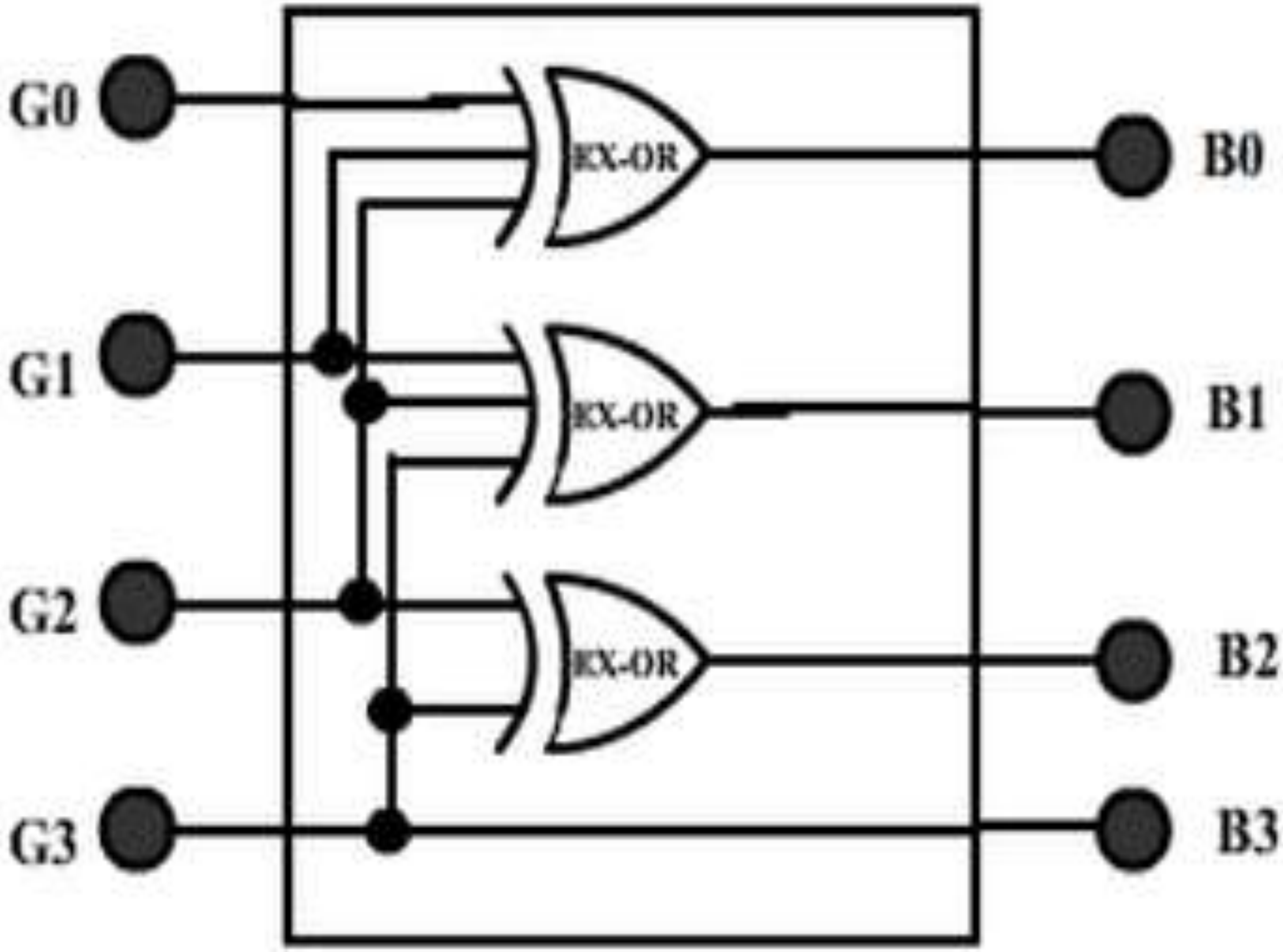
| $G_3G_2 \backslash G_1G_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 0 | 0 | 0 |
| 01 | 1 | 1 | 1 | 1 |
| 11 | 0 | 0 | 0 | 0 |
| 10 | 1 | 1 | 1 | 1 |

$$B_2 = G_3'G_2 + G_3G_2'$$

$$= G_3 \oplus G_2$$

**And** **B3 = G3**

## Gray to Binary Converter



# BCD TO EXCESS 3

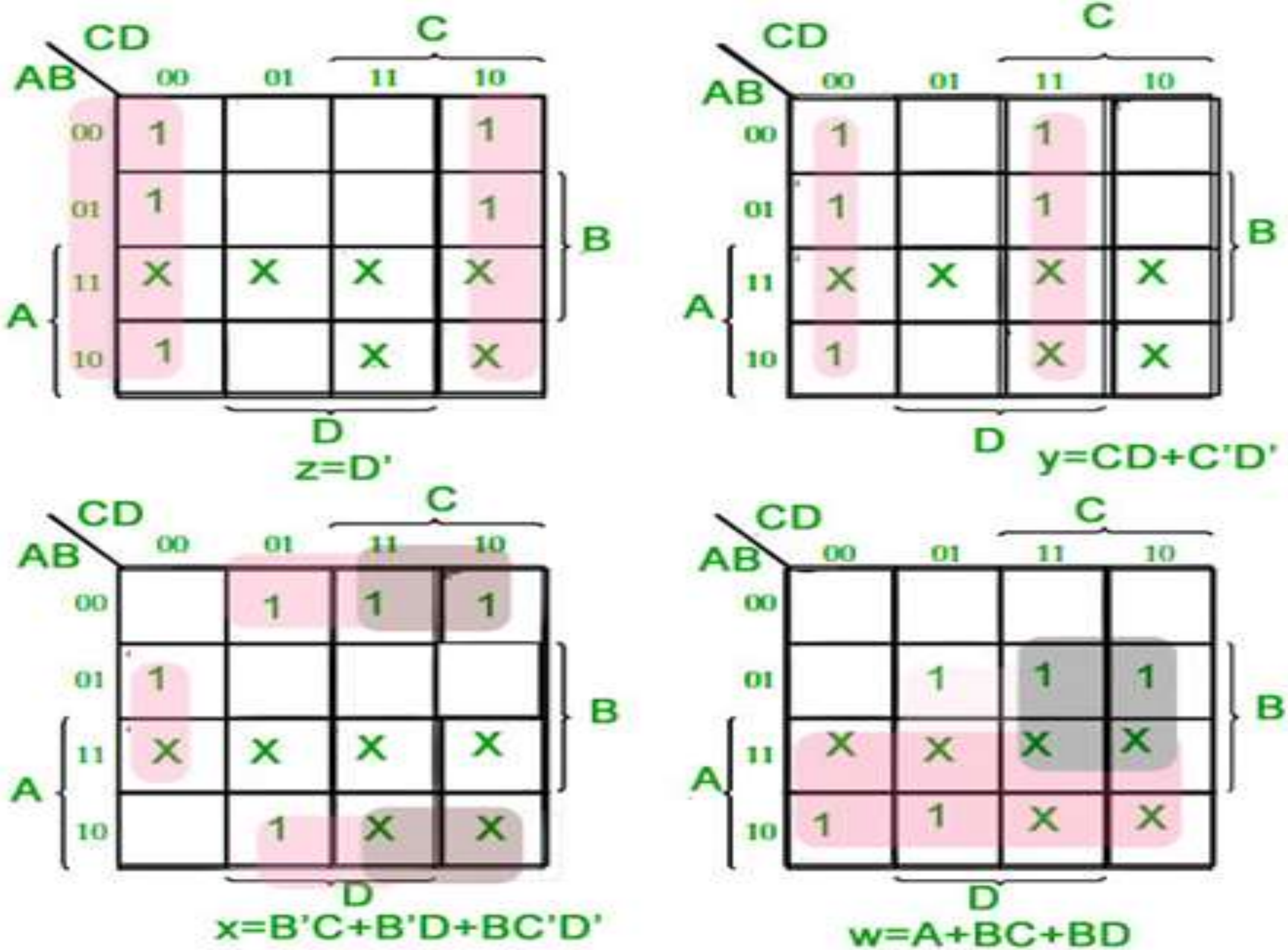
As is clear by the name, a BCD digit can be converted to it's corresponding Excess-3 code by simply adding 3 to it.

Let $A,\ B,\ C,\ and\ D$ be the bits representing the binary numbers, where $D$ is the LSB and $A$ is the MSB, and

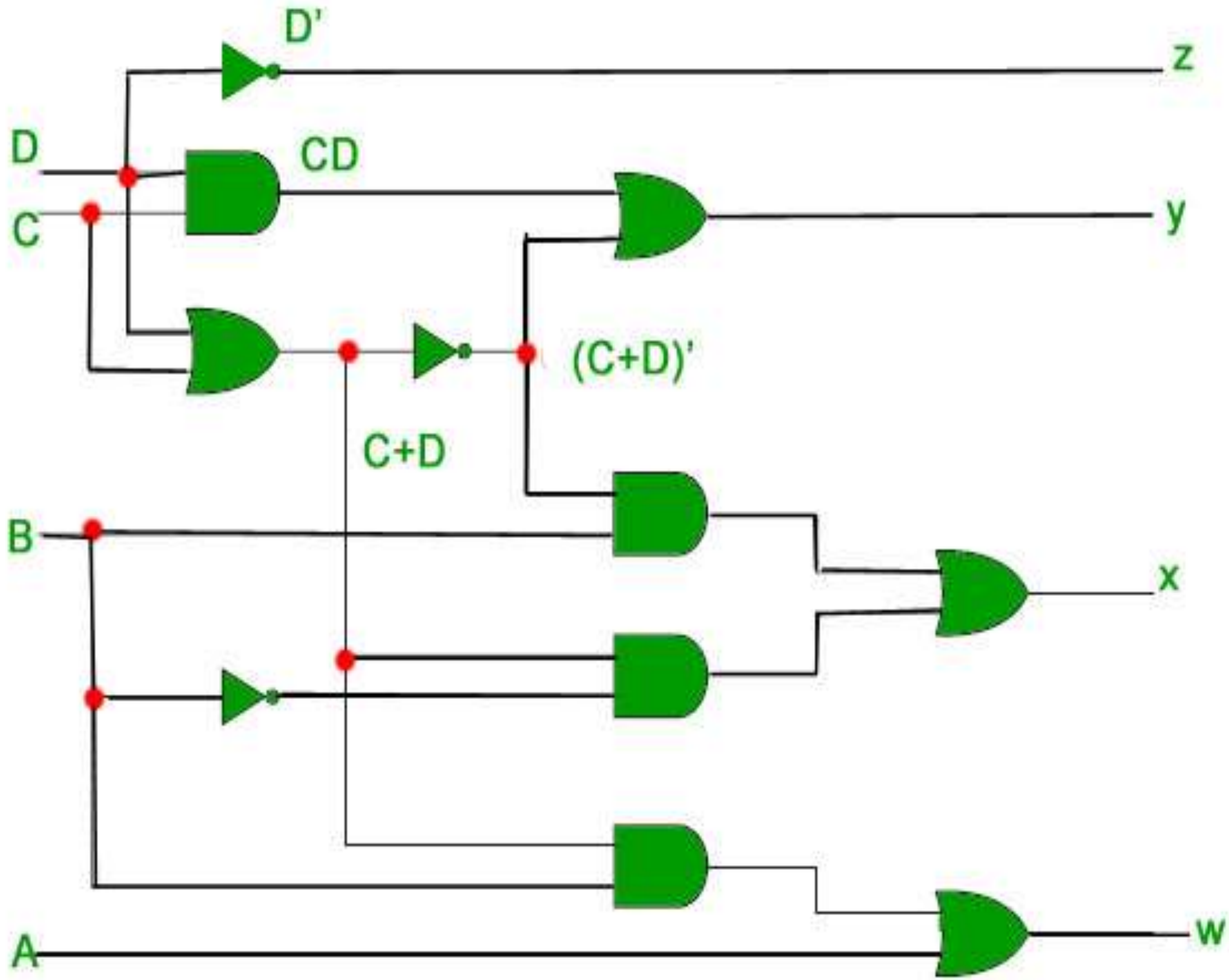
Let $w,\ x,\ y,\ and\ z$ be the bits representing the gray code of the binary numbers, where $z$ is the LSB and $w$ is the MSB.

The truth table for the conversion is given below. The X's mark don't care conditions.

| BCD(8421) | | | | Excess-3 | | | |
|---|---|---|---|---|---|---|---|
| A | B | C | D | w | x | y | z |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | X | X | X | X |
| 1 | 0 | 1 | 1 | X | X | X | X |
| 1 | 1 | 0 | 0 | X | X | X | X |
| 1 | 1 | 0 | 1 | X | X | X | X |
| 1 | 1 | 1 | 0 | X | X | X | X |
| 1 | 1 | 1 | 1 | X | X | X | X |



z=D'
y=CD+C'D'
x=B'C+B'D+BC'D'
w=A+BC+BD



D'
z
D
CD
C
y
(C+D)'
C+D
B
x
A
w

# BCD TO SEVEN SEGMENT DISPLAY

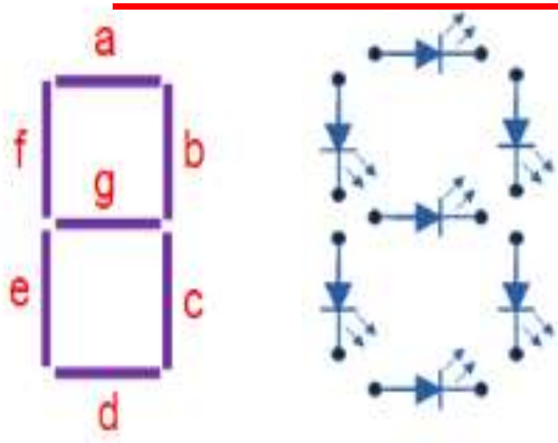



Figure1 Seven segment display

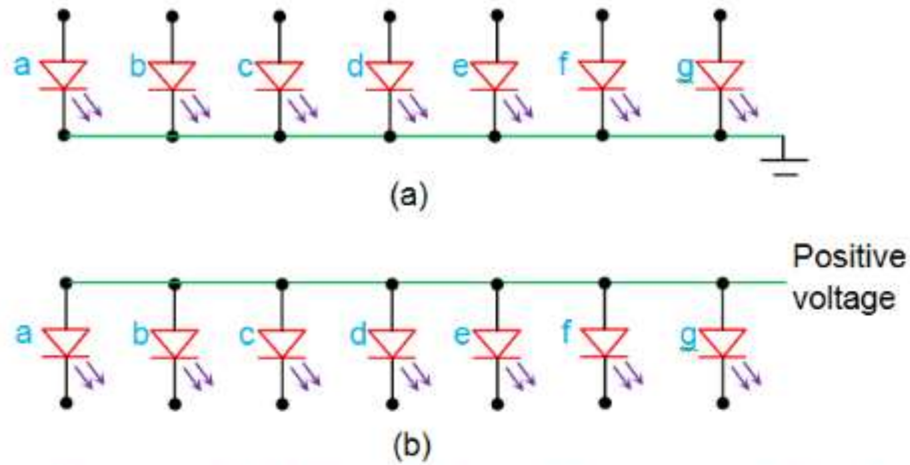



Figure 2 (a) Common cathode type display (b) Common anode type display

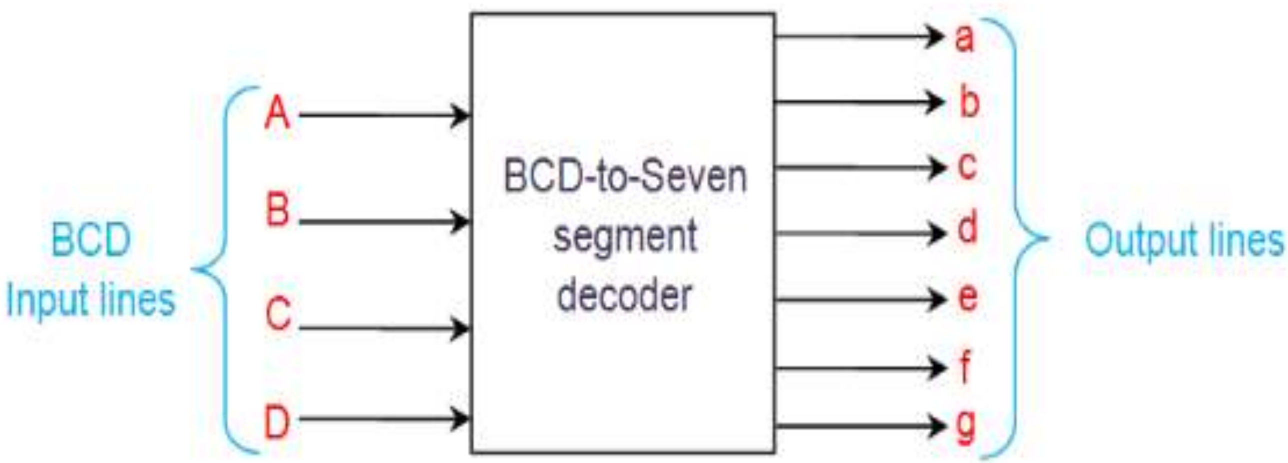
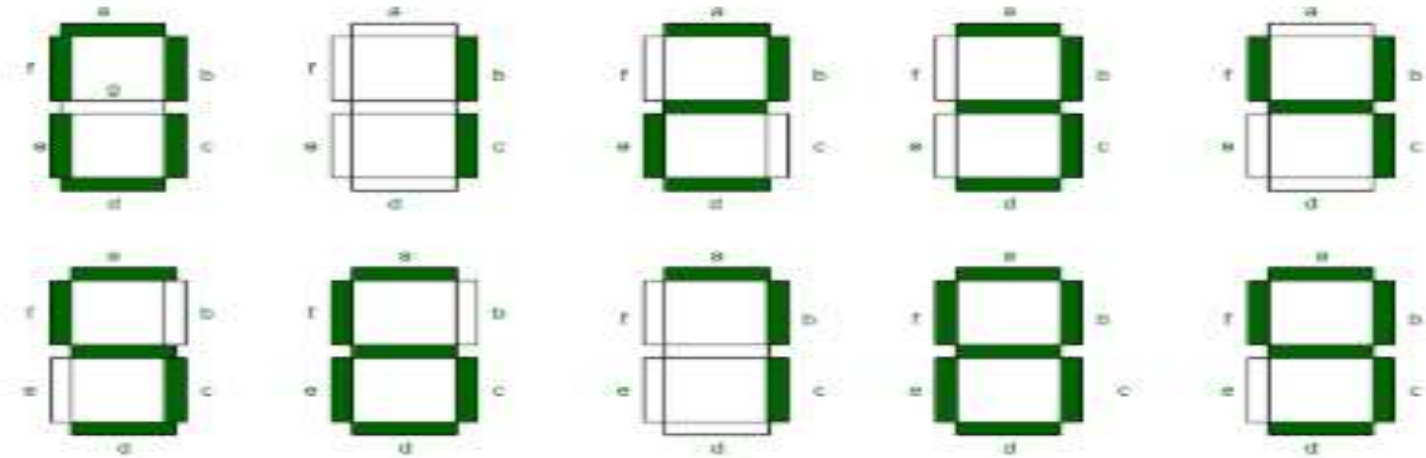
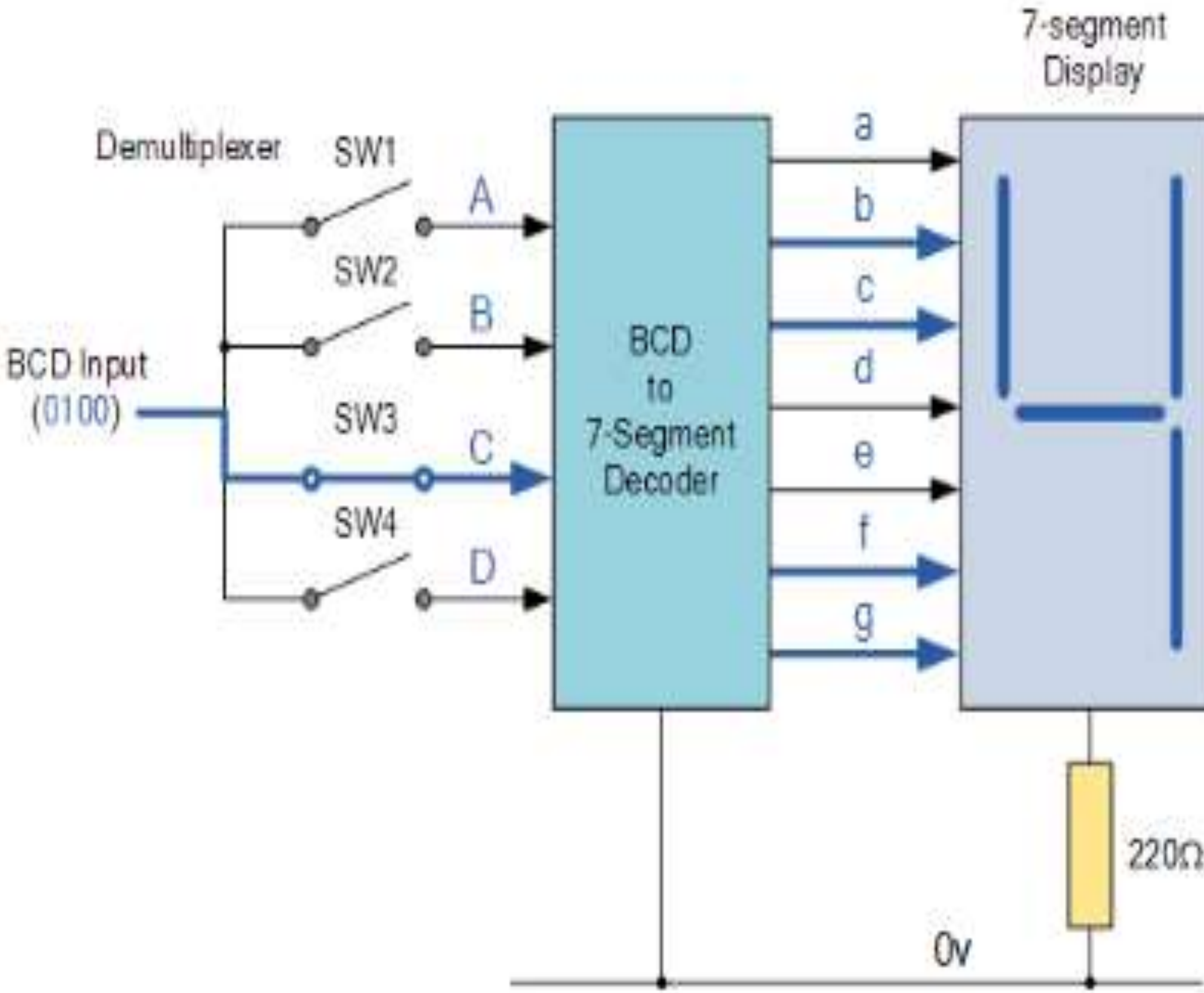



Figure 3 BCD-to-Seven segment decoder



| Decimal Digit | Input lines | | | | Output lines | | | | | | | Display pattern |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | A | B | C | D | a | b | c | d | e | f | g | |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| 2 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 2 |
| 3 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 3 |
| 4 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 4 |
| 5 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 5 |
| 6 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 6 |
| 7 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 7 |
| 8 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 8 |
| 9 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 9 |


7-segment
Display
Demultiplexer
SW1
SW2
SW3
SW4
A
B
C
D
BCD Input
(0100)
BCD
to
7-Segment
Decoder
a
b
c
d
e
f
g
220Ω
0v

| AB\CD | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 1 | 0 | 1 | 1 |
| 01 | 0 | 1 | 1 | 1 |
| 11 | X | X | X | X |
| 10 | 1 | 1 | X | X |

**FOR a: F(A,B,C,D)=B’D’+C+BD+A**

| AB\CD | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 1 | 1 | 1 | 1 |
| 01 | 1 | 0 | 1 | 0 |
| 11 | X | X | X | X |
| 10 | 1 | 1 | X | X |

**For b: F(A,B,C,D)=B’+C’D’+CD**

| AB\CD | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 1 | 1 | 1 | 0 |
| 01 | 1 | 1 | 1 | 1 |
| 11 | X | X | X | X |
| 10 | 1 | 1 | X | X |

**FOR c: F(A,B,C,D)= C’+D+B**

| AB\CD | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 1 | 0 | 1 | 1 |
| 01 | 0 | 1 | 0 | 1 |
| 11 | X | X | X | X |
| 10 | 1 | 1 | X | X |

**FOR d: F(A,B,C,D)=**
**B’D’+B’C+BC’D+CD’+A**

| AB\CD | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 1 | 0 | 0 | 1 |
| 01 | 0 | 0 | 0 | 1 |
| 11 | X | X | X | X |
| 10 | 1 | 0 | X | X |

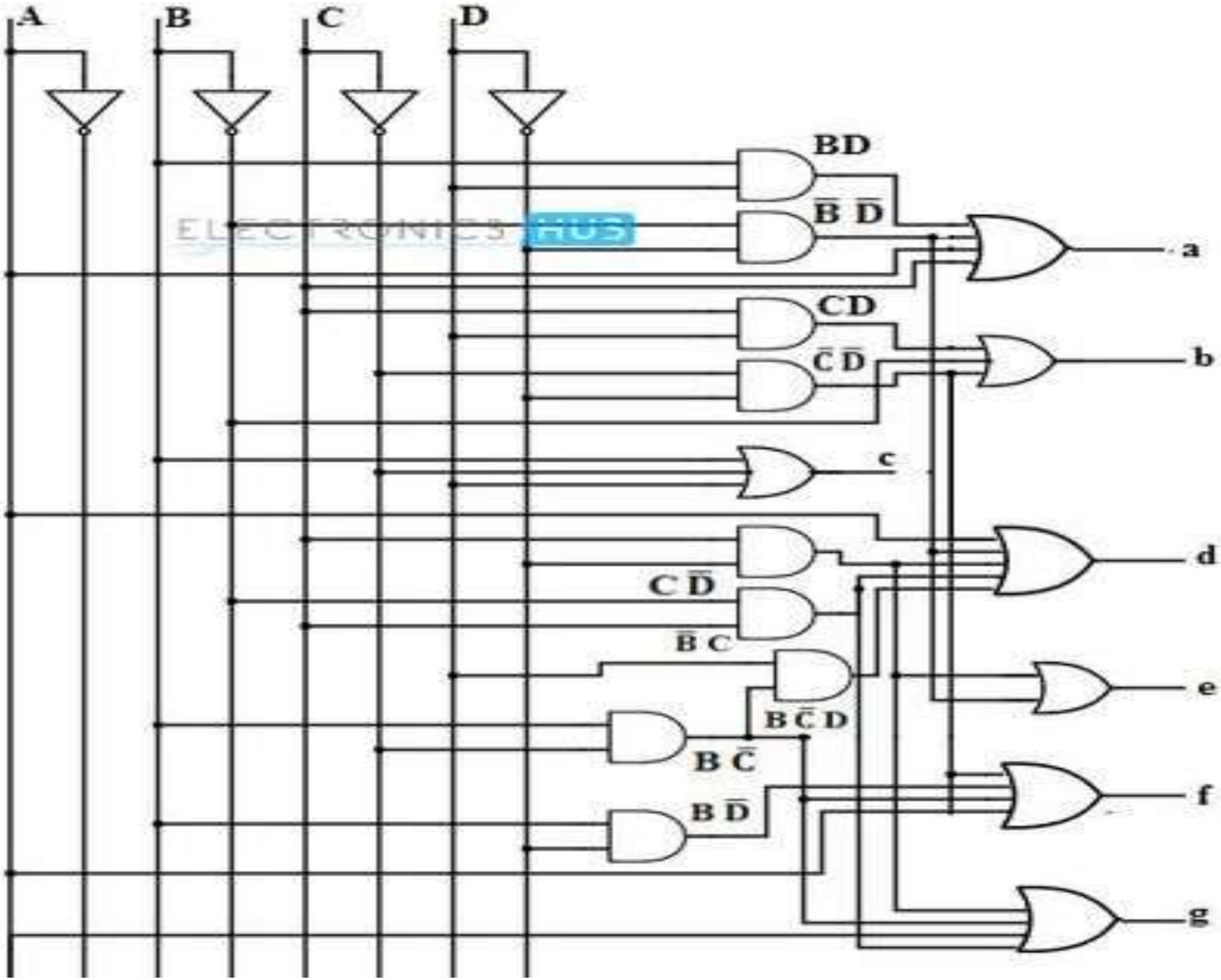
**FOR e: F(A,B,C,D)= B’D’+CD’**

| AB\CD | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 1 | 0 | 0 | 0 |
| 01 | 1 | 1 | 0 | 1 |
| 11 | X | X | X | X |
| 10 | 1 | 1 | X | X |

**FOR f: F(A,B,C,D)= C’D’+BC’+BD’+A**

| AB\CD | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 0 | 1 | 1 |
| 01 | 1 | 1 | 0 | 1 |
| 11 | X | X | X | X |
| 10 | 1 | 1 | X | X |

**FOR g: F(A,B,C,D)= B’C+BC’+A+BD’**


A
B
C
D
BD
$\bar{B}\,\bar{D}$
a
CD
$\bar{C}\,\bar{D}$
b
c
d
$C\,\bar{D}$
$\bar{B}\,C$
e
$B\,\bar{C}\,D$
$B\,\bar{C}$
f
$B\,\bar{D}$
g
ELECTRONICS HUB

**Thank you**