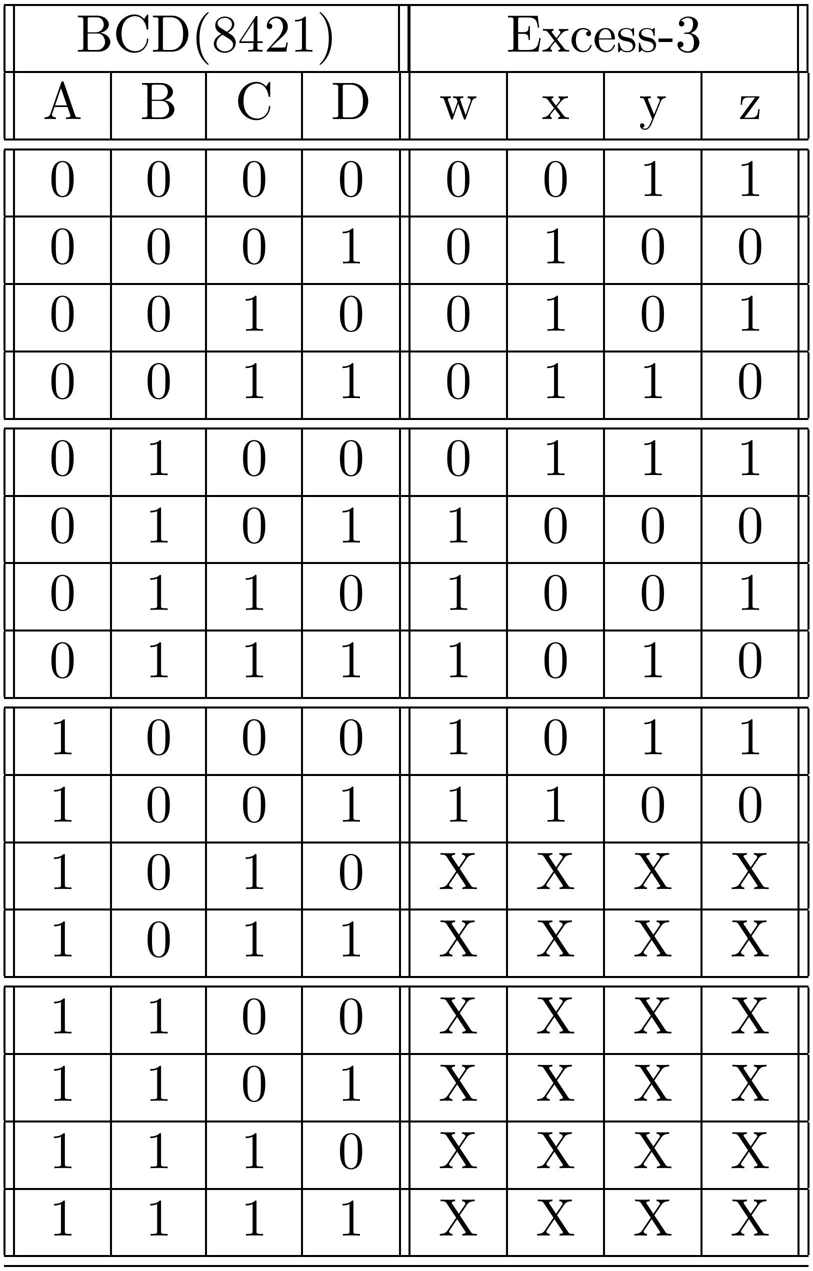
Digital Logic | Code Converters – BCD(8421) to/from Excess-3

Prerequisite – Number System and base conversions

Excess-3 binary code is a **unweighted self-complementary** BCD code.
Self-Complementary property means that the 1's complement of an excess-3 number is the excess-3 code of the 9's complement of the corresponding decimal number. This property is useful since a decimal number can be nines' complemented (for subtraction) as easily as a binary number can be ones' complemented; just by inverting all bits.
For example, the excess-3 code for 3(0011) is 0110 and to find the excess-3 code of the complement of 3, we just need to find the 1's complement of 0110 -> 1001, which is also the excess-3 code for the 9's complement of 3 -> (9-3) = 6.

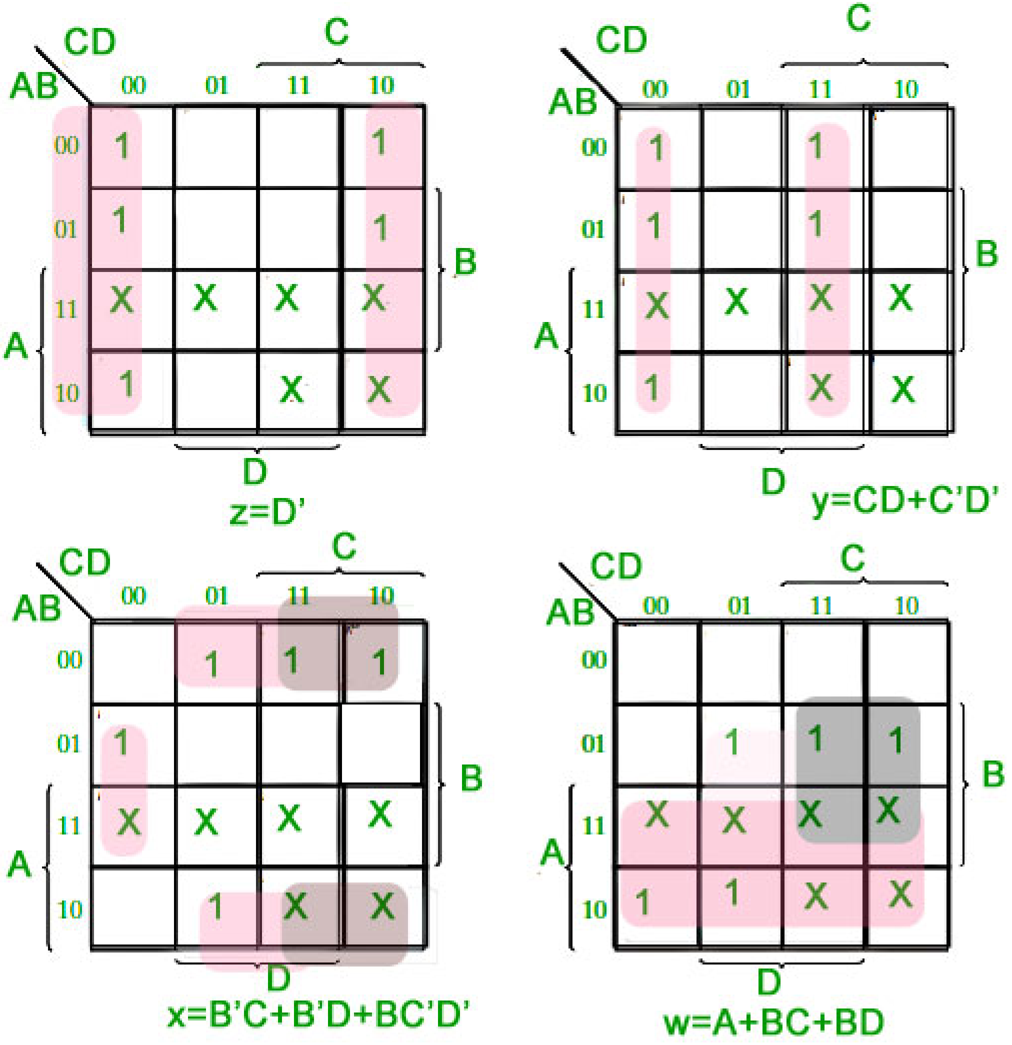
Converting BCD(8421) to Excess-3 –

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

To find the corresponding digital circuit, we will use the K-Map technique for each of the Excess-3 code bits as output with all of the bits of the BCD number as input.



Corresponding minimized Boolean expressions for Excess-3 code bits –

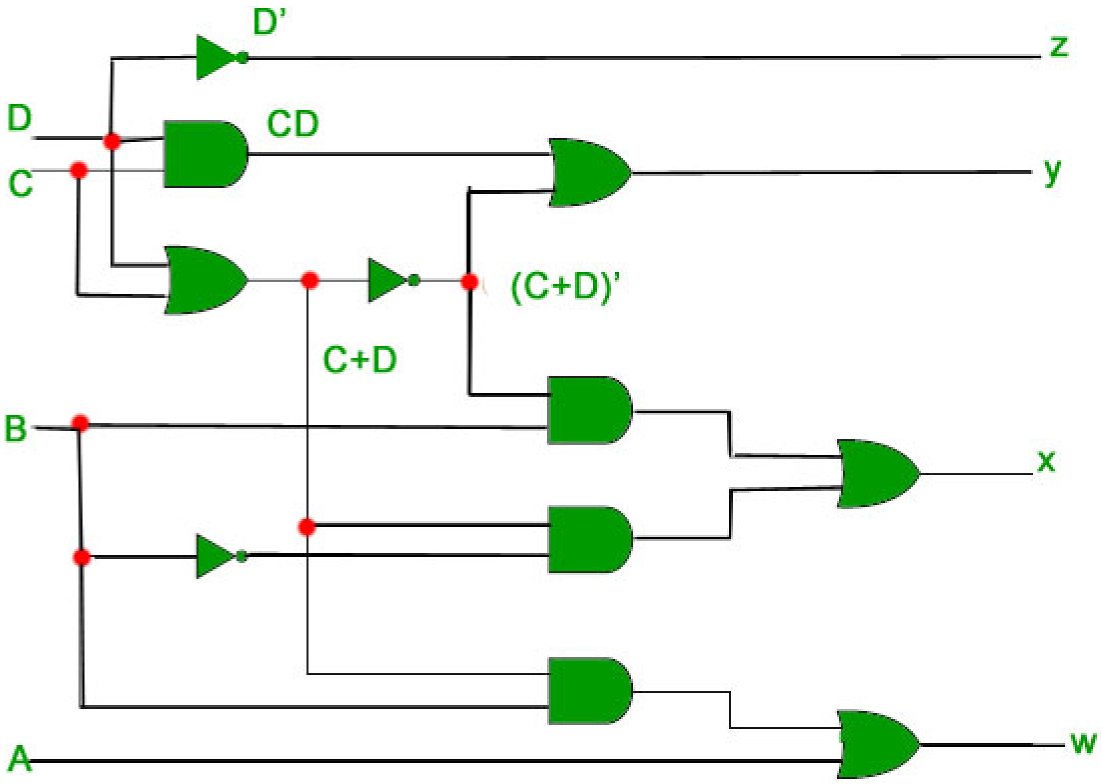
$$w = A + BC + BD$$

$$x = B'C + B'D + BC'D'$$

$$y = CD + C'D'$$

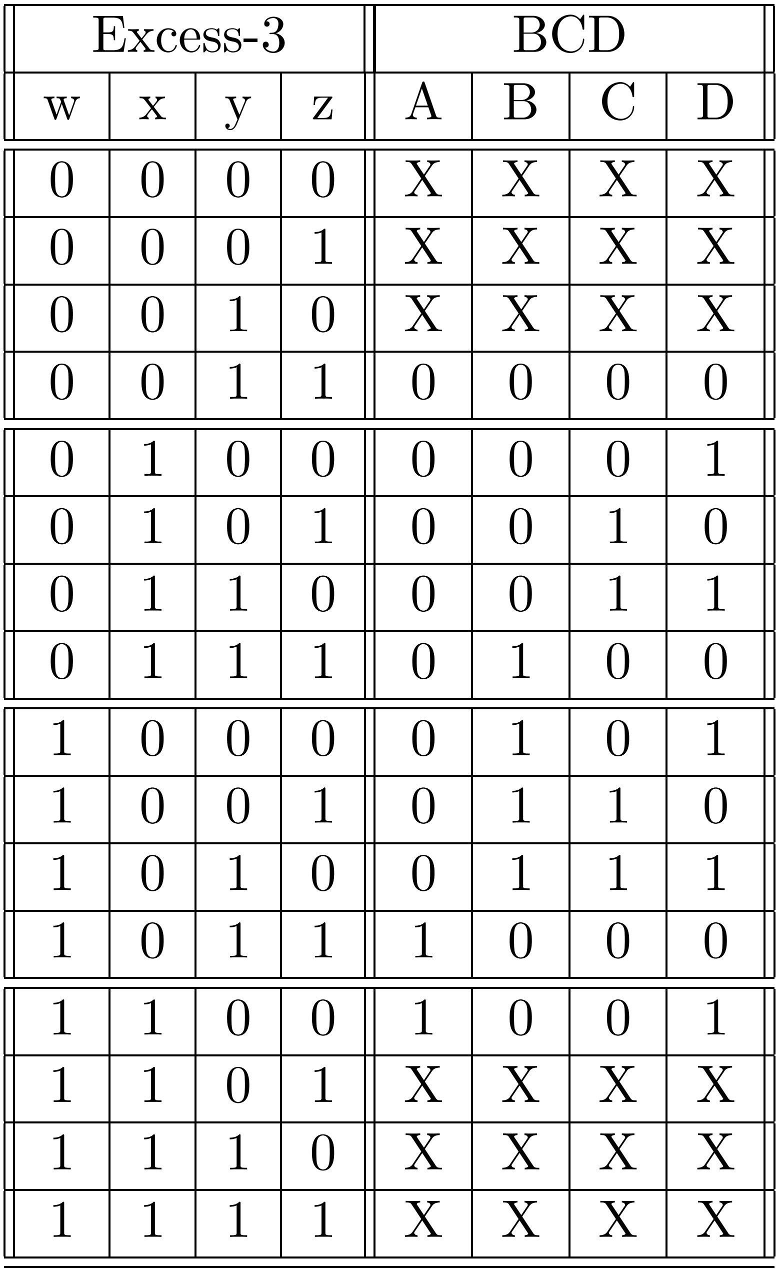
$$z = D'$$

The corresponding digital circuit-



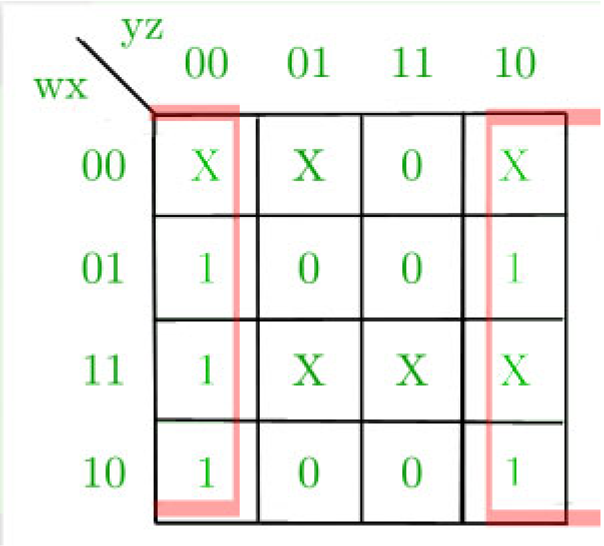
Converting Excess-3 to BCD(8421) –

Excess-3 code can be converted back to BCD in the same manner.
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.



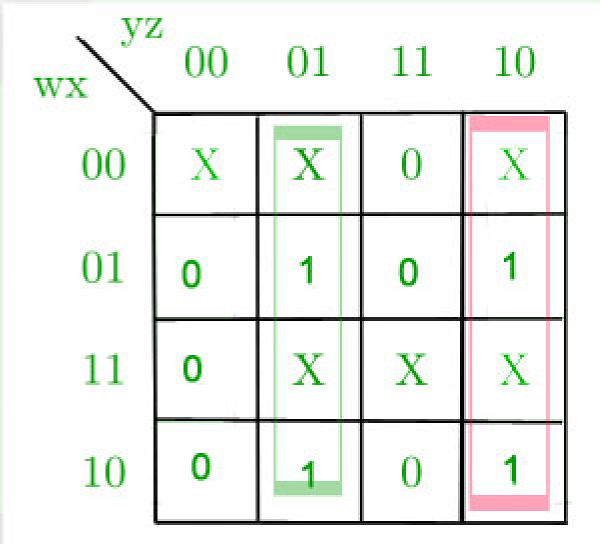
Excess-3				BCD			
w	x	y	z	A	B	C	D
0	0	0	0	X	X	X	X
0	0	0	1	X	X	X	X
0	0	1	0	X	X	X	X
0	0	1	1	0	0	0	0
0	1	0	0	0	0	0	1
0	1	0	1	0	0	1	0
0	1	1	0	0	0	1	1
0	1	1	1	0	1	0	0
1	0	0	0	0	1	0	1
1	0	0	1	0	1	1	0
1	0	1	0	0	1	1	1
1	0	1	1	1	0	0	0
1	1	0	0	1	0	0	1
1	1	0	1	X	X	X	X
1	1	1	0	X	X	X	X
1	1	1	1	X	X	X	X

K-Map for D-



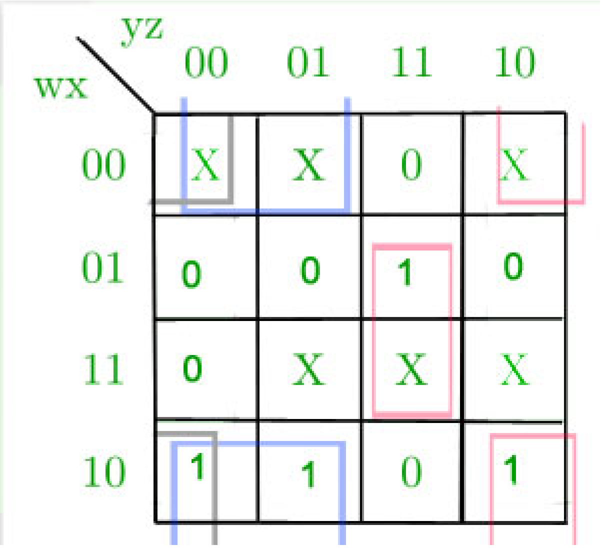
wx \ yz	00	01	11	10
00	X	X	0	X
01	1	0	0	1
11	1	X	X	X
10	1	0	0	1

K-Map for C-



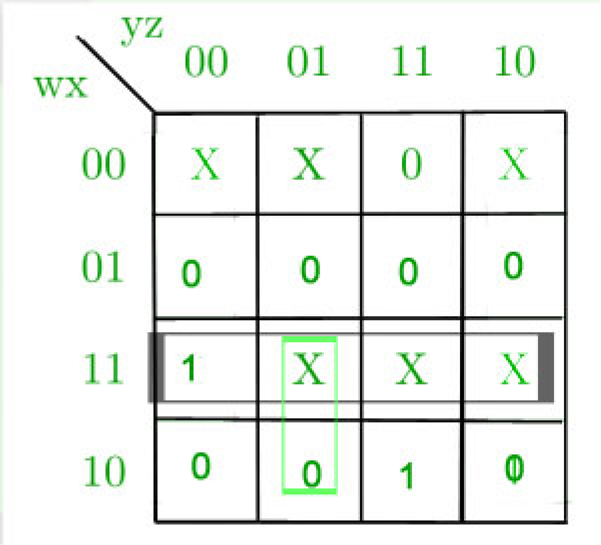
wx \ yz	00	01	11	10
00	X	X	0	X
01	0	1	0	1
11	0	X	X	X
10	0	1	0	1

K-Map for B-



wx \ yz	00	01	11	10
00	X	X	0	X
01	0	0	1	0
11	0	X	X	X
10	1	1	0	1

K-Map for A-



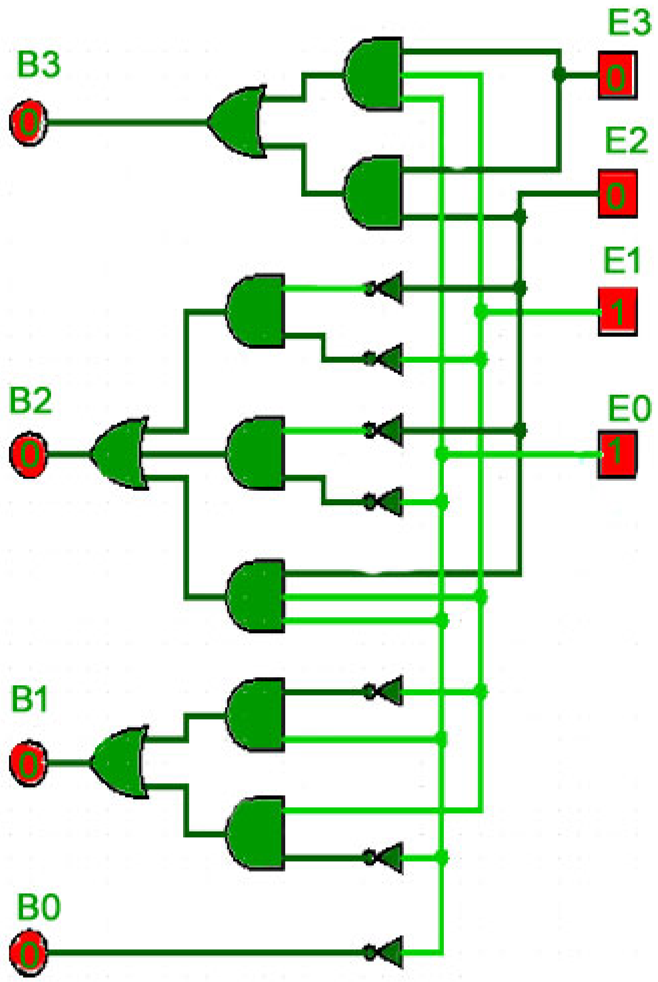
wx \ yz	00	01	11	10
00	X	X	0	X
01	0	0	0	0
11	1	X	X	X
10	0	0	1	0

Corresponding minimized boolean expressions for Excess-3 code bits –

$$A = wx + wyz$$
$$B = x'y' + x'z' + xyz$$
$$C = y^{+}yz'$$
$$D = z'$$

The corresponding digital circuit –

Here $E_3,\ E_2,\ E_1,\ and\ E_0$ correspond to $w,\ x,\ y,\ and\ z$ and $B_3,\ B_2,\ B_1,\ and\ B_0$ correspond to $A,\ B,\ C,\ and\ D$.



References-

Digital Design, 5th edition by Morris Mano and Michael Ciletti
Excess-3 – Wikipedia

This article is contributed by **Chirag Manwani**. If you like GeeksforGeeks and would like to contribute, you can also write an article using contribute.geeksforgeeks.org or mail your article to contribute@geeksforgeeks.org. See your article appearing on the GeeksforGeeks main page and help other Geeks.

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.

Digital Electronics & Logic Design GATE CS

Login to Improve this Article

Please write to us at contribute@geeksforgeeks.org to report any issue with the above content.

▲

Recommended Posts:

- Digital logic | Code Converters – Binary to/from Gray Code
- Number System and base conversions | Digital Electronics
- Program for hexadecimal to decimal
- Digital logic | Binary Decoder
- Last Minute Notes – Computer Networks
- Digital Logic | Implicants in K-Map
- Non-Restoring Division For Unsigned Integer
- Restoring Division Algorithm For Unsigned Integer
- Amortized analysis for increment in counter
- Digital Electronics | Operational Amplifier (op-amp)

(Login to Rate)

0 Average Difficulty : **0/5.0**
No votes yet.

Basic Easy Medium Hard Expert

□ Add to TODO List

□ Mark as DONE

Writing code in comment? Please use ide.geeksforgeeks.org, generate link and share the link here.

Share this post!

2 Comments **GeeksforGeeks** 1 **Login**

♡ **Recommend** **Share** **Sort by Newest**

Join the discussion…

LOG IN WITH

OR SIGN UP WITH DISQUS ?

Name

Shoikot • a month ago

In Converting BCD(8421) to Excess-3:
K-Map for X is worng... IN 0110 there will be no 1.

^ | v • Reply • Share ›

Mithlesh Upadhyay Mod ➦ Shoikot • a month ago

Thanks for pointing out, corrected.

^ | v • Reply • Share ›

Subscribe **Add Disqus to your siteAdd DisqusAdd** **Privacy**

A computer science portal for geeks

710-B, Advant Navis Business Park,
Sector-142, Noida, Uttar Pradesh - 201305
feedback@geeksforgeeks.org

COMPANY

About Us
Careers
Privacy Policy
Contact Us

LEARN

Algorithms
Data Structures
Languages
CS Subjects
Video Tutorials

PRACTICE

Company-wise
Topic-wise
Contests
Subjective Questions

CONTRIBUTE

Write an Article
GBlog
Videos

▲