Logic Project

project Name:

BCD(8421) to/from Excess-3

Team:	**section 6**
Mahmoud Ezzat El-agami	
Nada Hashem Mohamed	
Mostafa Eslam Jeweily	
Yossef Samy El-Beheiry	
Mostafa Ezz	

project Overview

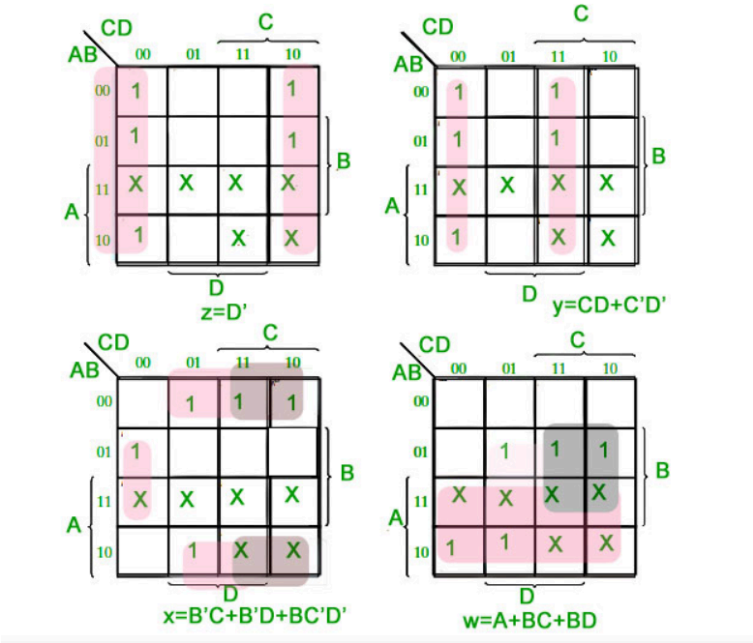
Excess-3 binary code is an **unweighted self-complementary** BCD code.
Self-Complementary property means that the 1's complement of an excess-3 number is the excess-3 code of the 9's complement of the corresponding decimal number. This property is useful since a decimal number can be nines' complemented (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 its corresponding Excess-3 code by simply adding 3 to it. Since we have only 10 digits(0 to 9) in decimal, we don't care about the rest and marked them with a cross(X).
Let A,B,C and D be the bits representing the binary numbers, where is the LSB and is the MSB, and
Let w,x,y and Z be the bits representing the gray code of the binary numbers, where is the LSB and is the MSB.
The truth table for the conversion is given below. The X's mark is don't care condition.

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

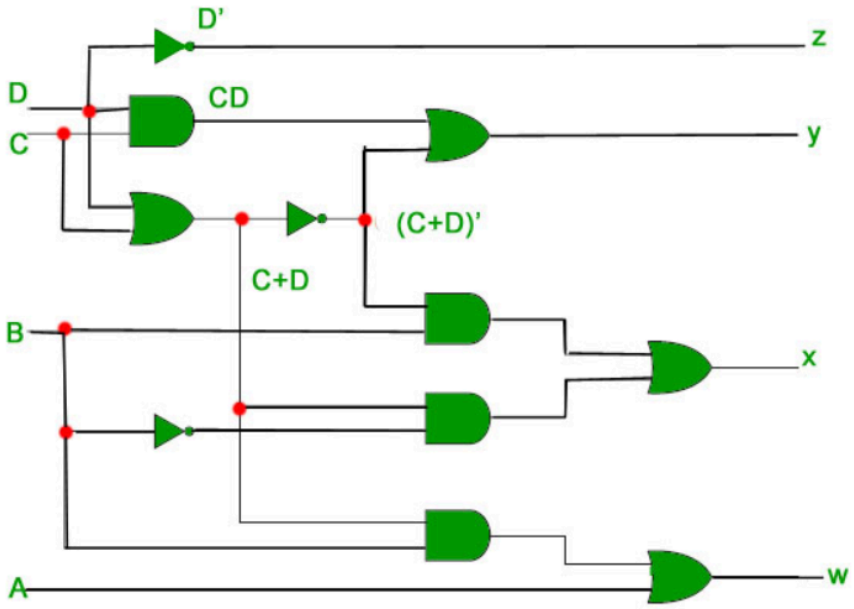
To find the corresponding digital circuit, we will use the K-Map technique for each of the Excess-3 code bits as output with all of the bits of the BCD number as input.



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)-

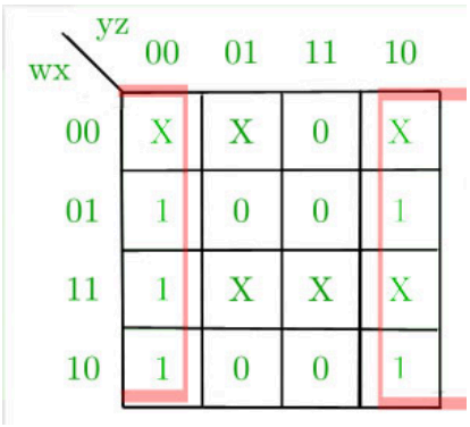
Excess-3 code can be converted back to BCD in the same manner.

Let A,B,C and D be the bits representing the binary numbers, where is the LSB and is the MSB, and Let w,x,y and z be the bits representing the gray code of the binary numbers, where D is the LSB and w is the MSB.

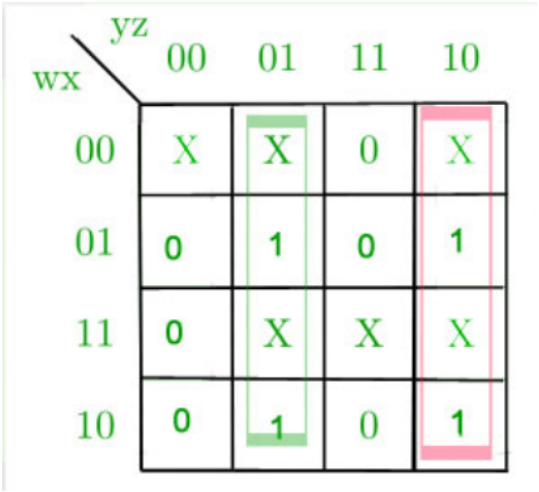
The truth table for the conversion is given below. The X's mark is don't care condition.

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

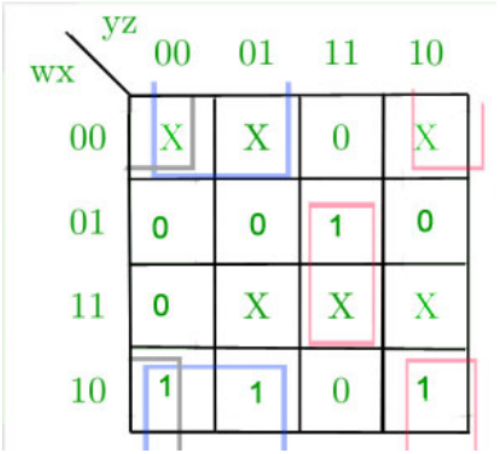
K-Map for D-



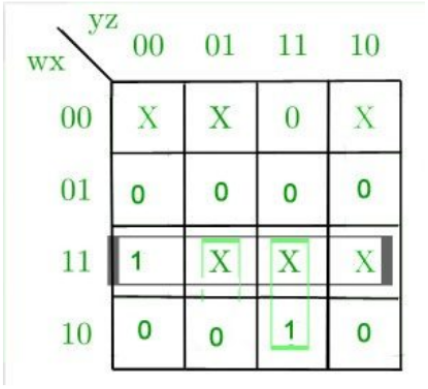
K-Map for C-



K-Map for B-



K-Map for A-



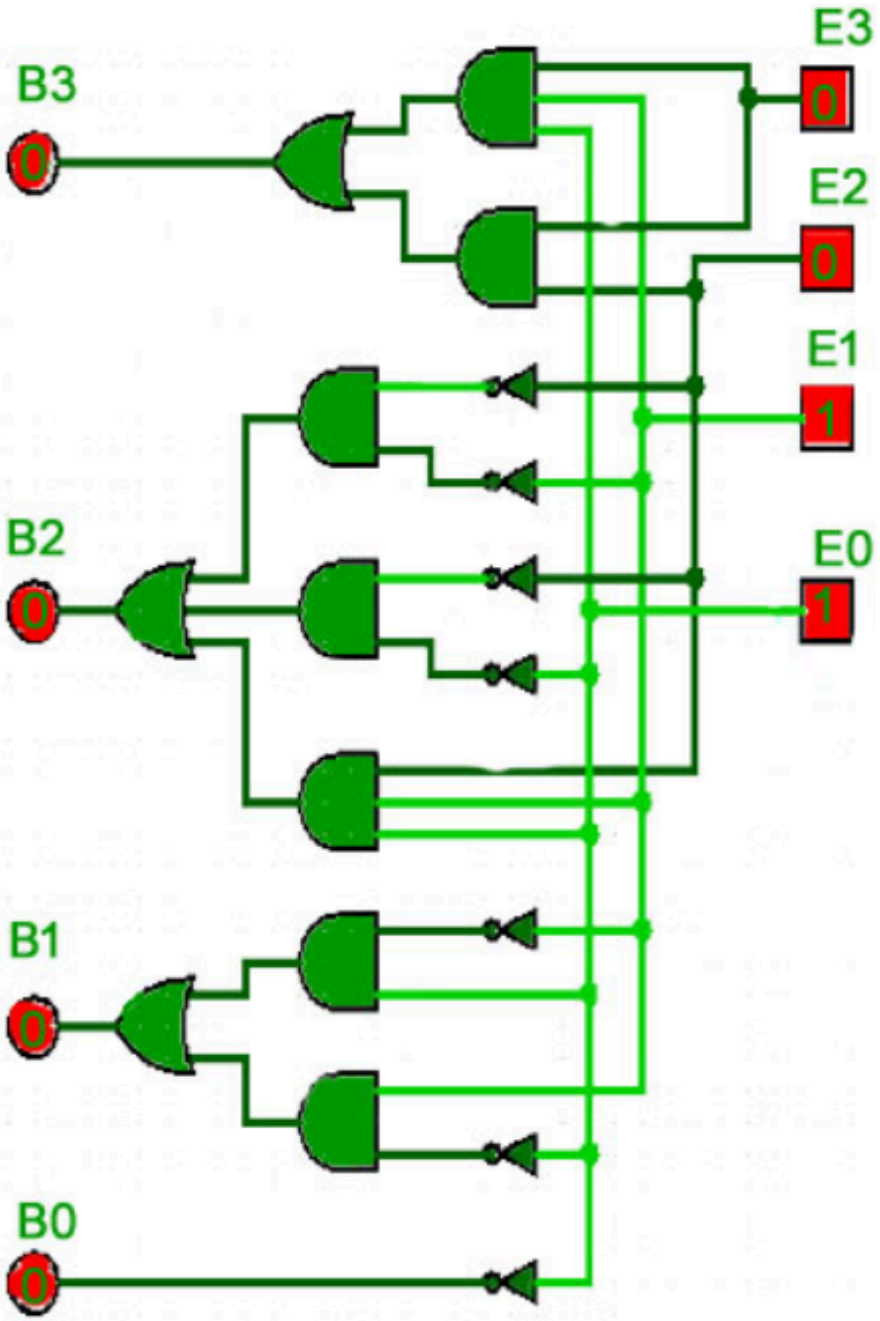
wx \ yz	00	01	11	10
00	X	X	0	X
01	0	0	0	0
11	1	X	X	X
10	0	0	1	0

Corresponding minimized boolean expressions for Excess-3 code bits –

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

The corresponding digital circuit –

Here E3 ,E2,E1 and E0 correspond to w,x,y and Z and B3 ,B2,B1 and B0 correspond to . A,B,C and D


E3
B3
0
E2
0
E1
1
B2
0
E0
1
B1
0
B0
0