Title of the Case Study :

EXCESS 3 TO BCD CONVERSION

Roll No & Student name :

BU21EECE0100387 MOHAN PS

BU21EECE0100486 D.GOWHASRI

Semester and Year :

6TH Semester & 3rd year

Subject Code & Subject Name :

EECE3051 VLSI Design

Introduction :

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. Excess-3 binary code is an unweighted self-complementary BCD code.

Truth Table, Equation and Logical Diagram :

Truth Table :

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

Equation :

The equation are derived from truth table and with help of K-map

THE DERIVED EQUATION ARE:

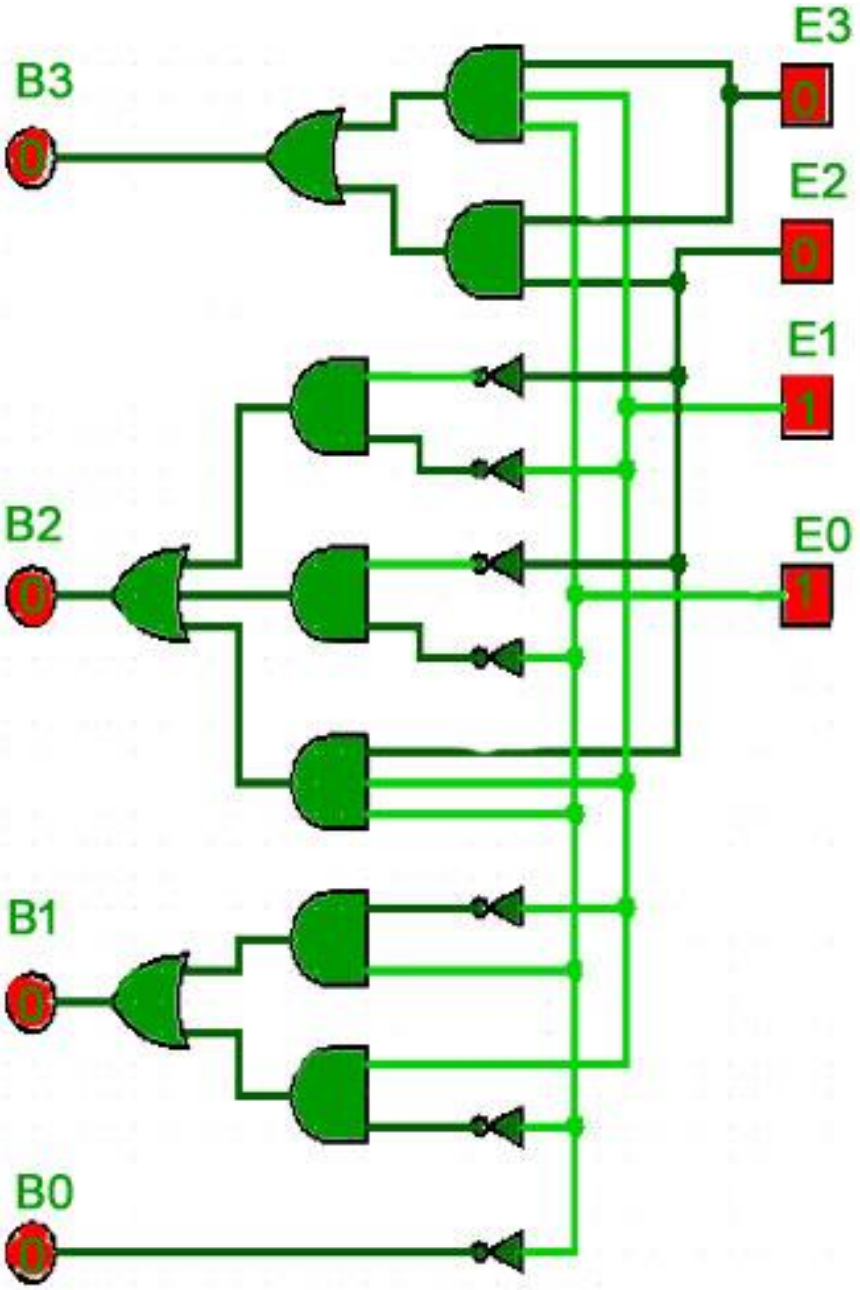
M = AB+ACD

X = B0D0+B0C0+BCD

Y= C XOR D

Z = D0

Logical Diagram :



Working :

- **Step 1** – Take each Excess-3 code.
- **Step 2** – Subtract 3 from each Excess-3 code. The result will be the equivalent BCD code.
- **Step 3** – Combine all the BCD codes equivalent to each Excess-3 code to obtain the final result in BCD representation.

Application :

Arithmetic-Friendly Nature

Early Adoption in Computing Devices

Conversion Process:

Video Drive link :

https://drive.google.com/file/d/1amUFEzUy2Yixt_jUQ_1SEz5ZBi-UO8lX/view?usp=drivesdk