

Title: code converterObjective:

- 1> Design & implement 4-bit binary to Gray-code converter using minimum number of gates & vice-versa.
- 2> Design & implement Excess-3 to BCD code converter using minimum number of logic gates & vice versa.

Apparatus

Digital Board, GP-4 patch chords, IC74LS86, IC74LS32, IC74LS08/IC74LS04 & required logic gates if any.

Theory:

Code converter is combinational logic circuits which can be used to convert one number system to another.

Binary code is a code having base-2.

Gray code is a code in which one bit change is obtained.

BCD code is 4-bit binary code but it is valid from 0-9.

Excess-3 code is binary code (4-bit) which can be obtained by adding 3 to each binary.

Excess-3 is non-weighted code.

Pin diagram

1A	V	Vcc
1B	PIN DIAGRAM	4B
1Y	for	4A
2A	IC 74LS00/	4Y
2B	08/32/86	3B
2Y		3A
GND		3Y

Procedure

- 1> Make the connections as per the logic circuit of 4-bit

binary to 4-bit Gray code converter & vice-versa & verify its truth-table.

2) Make the connections as per the logic circuits of 4-bit BCD to 4-bit Excess-3 code converter & vice-versa & verify its truth-table.

Design of 4-bit binary to gray code converter:

Dec Equ.	Binary Code Input				Gray Code Output			
	B ₃	B ₂	B ₁	B ₀	G ₃	G ₂	G ₁	G ₀
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1
2	0	0	1	0	0	0	1	1
3	0	0	1	1	0	0	1	0
4	0	1	0	0	0	1	1	1
5	0	1	0	1	0	1	1	0
6	0	1	1	0	0	1	0	0
7	0	1	1	1	0	1	0	1
8	1	0	0	0	1	1	0	1
9	1	0	0	1	1	1	1	0
10	1	0	1	0	1	1	0	0
11	1	0	1	1	1	1	0	1
12	1	1	0	0	1	0	0	0
13	1	1	0	1	1	0	0	1
14	1	1	1	0	1	0	1	1
15	1	1	1	1	1	0	1	0

k-map simplification:

$$G_0 \Rightarrow$$

		B ₁ B ₀				
		00	01	11	10	
B ₂ B ₃		00	0	1	0	1
00	01	0	1	0	1	
01	10	1	0	1	0	
11	10	0	1	1	1	
10	01	0	1	0	1	

$$G_1 \Rightarrow$$

		B ₁ B ₀				
		00	01	11	10	
B ₂ B ₃		00	0	0	1	1
00	01	0	1	1	0	0
01	11	1	1	1	0	0
11	10	1	0	1	0	1
10	00	0	0	1	1	1

21/18 DATE

$$G_0 = B_0 \bar{B}_1 + B_1 \bar{B}_0 \\ = B_1 \oplus B_0$$

$$G_1 = B_2 \bar{B}_1 + B_1 \bar{B}_2 \\ = B_1 \oplus B_2$$

		$G_2 \Rightarrow$					
		$B_3 B_2$	$B_1 B_0$	00	01	11	10
$B_3 B_2$	00	0	0	0	0	0	0
01	1	1	1	1	1	1	1
11	0	0	0	0	0	1	1
10	1	1	1	1	1	1	1

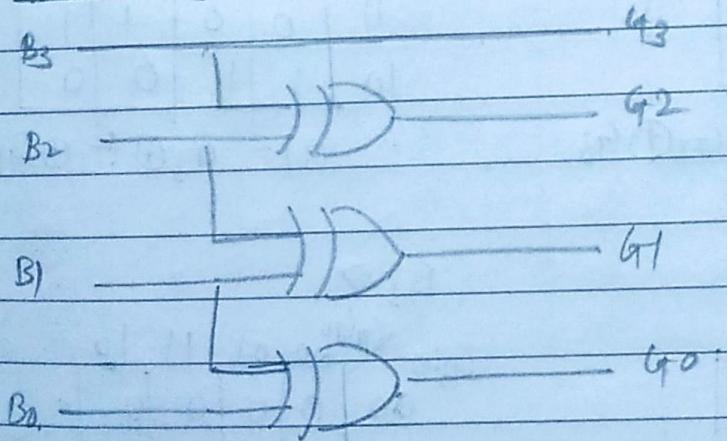
		$G_3 \Rightarrow$					
		$B_3 B_2$	$B_1 B_0$	00	01	11	10
$B_3 B_2$	00	0	0	0	0	0	0
01	0	0	0	0	0	0	0
11	1	1	1	1	1	1	1
10	1	1	1	1	1	1	1

$$G_3 = B_3$$

$$G_2 = B_2 \bar{B}_3 + B_3 \bar{B}_2$$

$$= B_2 \oplus B_3$$

Logic diagram:



Design of 4-bit gray to binary code converter:

Dec Eqn.	Gray Binary Code Input				Binary Gray Code Output			
	G_3	G_2	G_1	G_0	B_3	B_2	B_1	B_0
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1
2	0	0	1	0	0	0	1	1
3	0	0	1	1	0	0	1	0
4	0	1	0	0	0	1	1	1
5	0	1	0	1	0	1	1	0
6	0	1	1	0	0	1	1	0
7	0	1	1	1	0	1	0	1

8	1	0	0	0	1	1	1	1	1
9	1	0	0	1	1	1	1	1	0
10	1	0	1	0	1	1	0	0	0
11	1	0	1	1	1	1	0	1	
12	1	1	0	0	1	0	0	0	0
13	1	1	0	1	1	0	0	1	
14	1	1	1	0	1	0	1	1	
15	1	1	1	1	1	0	1	0	

K-map simplification:

$B_0 \Rightarrow$

		G ₁ G ₀			
		00	01	11	10
G ₃ G ₂	00	0	1	0	1
	01	1	0	1	0
	11	0	1	0	1
	10	1	0	1	0

$$B_0 = G_3 \oplus G_2 \oplus G_1 \oplus G_0$$

$B_1 \Rightarrow$

		G ₁ G ₀			
		00	01	11	10
G ₃ G ₂	00	0	0	1	1
	01	1	1	1	0
	11	0	0	0	1
	10	1	1	0	0

$$B_1 = G_3 \oplus G_2 \oplus G_1$$

$B_2 \Rightarrow$

		G ₁ G ₀			
		00	01	11	10
G ₃ G ₂	00	0	0	0	0
	01	1	1	1	1
	11	0	0	0	0
	10	1	1	1	1

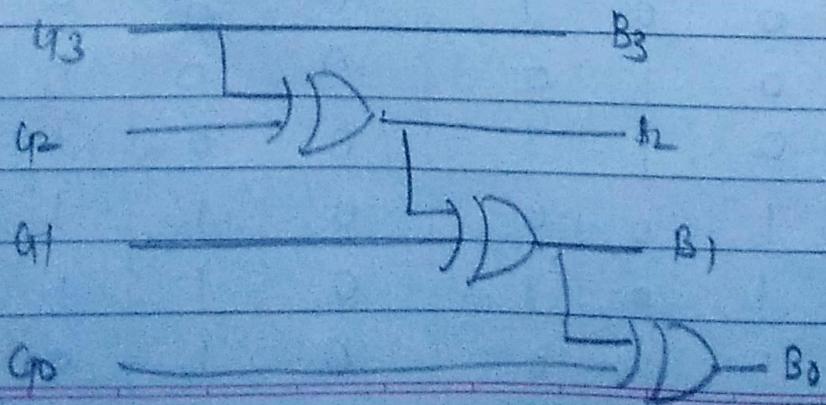
$$B_2 = G_3 \oplus G_2$$

$B_3 \Rightarrow$

		G ₁ G ₀			
		00	01	11	10
G ₃ G ₂	00	0	0	0	0
	01	0	0	0	0
	11	1	1	1	1
	10	1	1	1	1

$$B_3 = G_3$$

Logic diagram



Design of BCD code to Excess-3 code converter:

Dec. Eqn.	BCD code Input				Excess-3 code Output			
	B ₃	B ₂	B ₁	B ₀	E ₃	E ₂	E ₁	E ₀
0	0	0	0	0	0	0	1	1
1	0	0	0	1	0	1	0	0
2	0	0	1	0	0	1	0	1
3	0	0	1	1	0	1	1	0
4	0	1	0	0	0	1	1	1
5	0	1	0	1	1	0	0	0
6	0	1	1	0	1	0	0	1
7	0	1	1	1	1	0	1	0
8	1	0	0	0	1	0	1	1
9	1	0	0	1	1	1	0	0
10	1	0	1	0	x	x	x	x
11	1	0	1	1	x	x	x	x
12	1	1	0	0	x	x	x	x
13	1	1	0	1	x	x	x	x
14	1	1	1	0	x	x	x	x
15	1	1	1	1	x	x	x	x

k-map simplification:

		B ₃ ⇒			
		B ₂	B ₁	B ₀	
		00	01	11	10
00		1			
01		1			
11		x	x	x	x
10		x	x	x	x

$E_3 = \overline{B_0}$.

		E ₂ ⇒			
		B ₂	B ₁	B ₀	
		00	01	11	10
00		1			
01		1			
11		x	x	x	x
10		1			x

$$E_2 = B_1 B_0 + \overline{B_1} \overline{B_0} = B_1 \oplus B_0$$

		E ₁ ⇒			
		B ₂	B ₁	B ₀	
		00	01	11	10
00					
01					
11		x	x	x	x
10		x	x	x	x

		E ₀ ⇒			
		B ₂	B ₁	B ₀	
		00	01	11	10
00					
01					
11		x	x	x	x
10		x	x	x	x

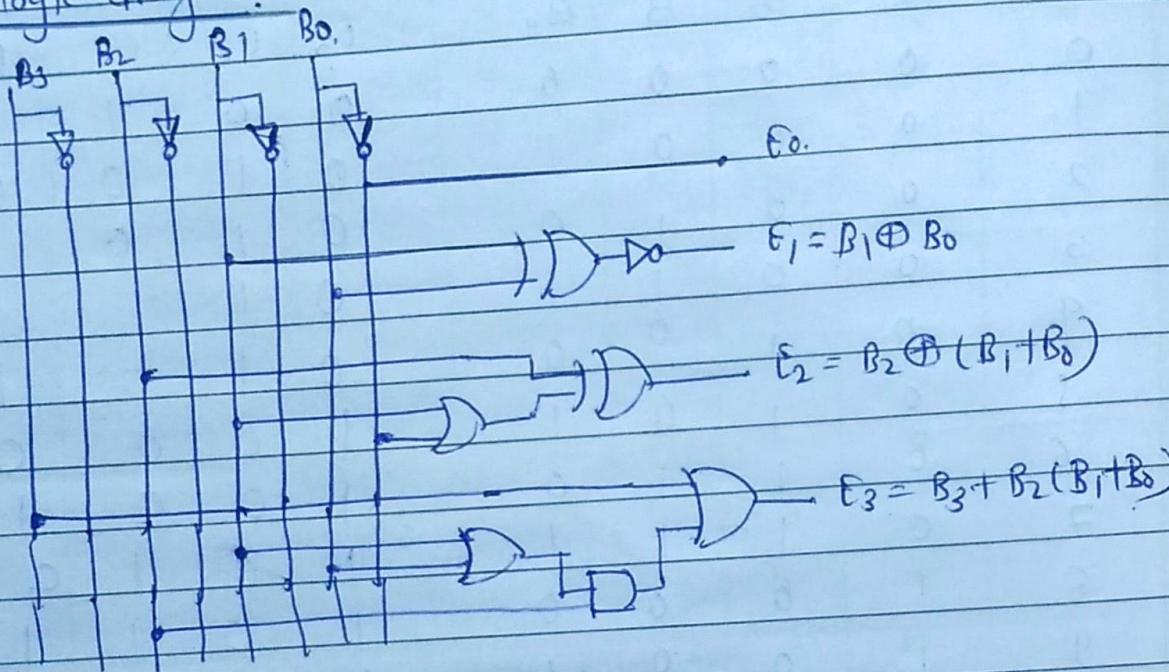
$$E_2 = \overline{B_2} B_1 + \overline{B_2} B_0 + B_2 \overline{B_1} \overline{B_0}$$

$$= B_2 \oplus (B_1 + B_0)$$

$$E_3 = B_3 + B_2 B_0 + B_2 B_1$$

$$= B_3 + B_2 (B_1 + B_0)$$

logic diagram.



Design of Excess-3 to BCD code converter.

Dec Eq?	Excess-3 code input				BCD code output			
	E ₃	E ₂	E ₁	E ₀	B ₃	B ₂	B ₁	B ₀
0	0	0	0	0	X	X	X	X
1	0	0	0	1	X	X	X	X
2	0	0	1	0	X	X	X	X
3	0	0	1	1	0	0	0	0
4	0	1	0	0	0	0	0	1
5	0	1	0	1	0	0	1	0
6	0	1	1	0	0	0	1	0
7	0	1	1	1	0	1	0	0
8	1	0	0	0	0	1	0	1
9	1	0	0	1	0	1	1	0
10	1	0	1	0	1	0	0	0
11	1	0	1	1	1	0	0	0
12	1	1	0	0	1	0	0	1
13	1	1	0	1	X	X	X	X
14	1	1	1	0	X	X	X	X
15	1	1	1	1	X	X	X	X

K-map simplification:

$B_0 \Rightarrow$

		00	01	11	10
		$E_1 E_0$	X	X	
		00	1	1	
		01			
		11	X	X	X
		10	X	X	X

$$B_0 = \bar{E}_0$$

$B_1 \Rightarrow$

		00	01	11	10
		$E_1 E_0$	X	X	X
		00		1	1
		01			
		11	X	X	X
		10	X	X	X

$$B_1 = \bar{E}_1 E_0 + E_1 \bar{E}_0$$

$$= E_1 \oplus E_0.$$

$B_2 \Rightarrow$

		00	01	11	10
		$E_2 E_0$	X	X	X
		00		1	1
		01		1	1
		11	X	X	X
		10	1	1	1

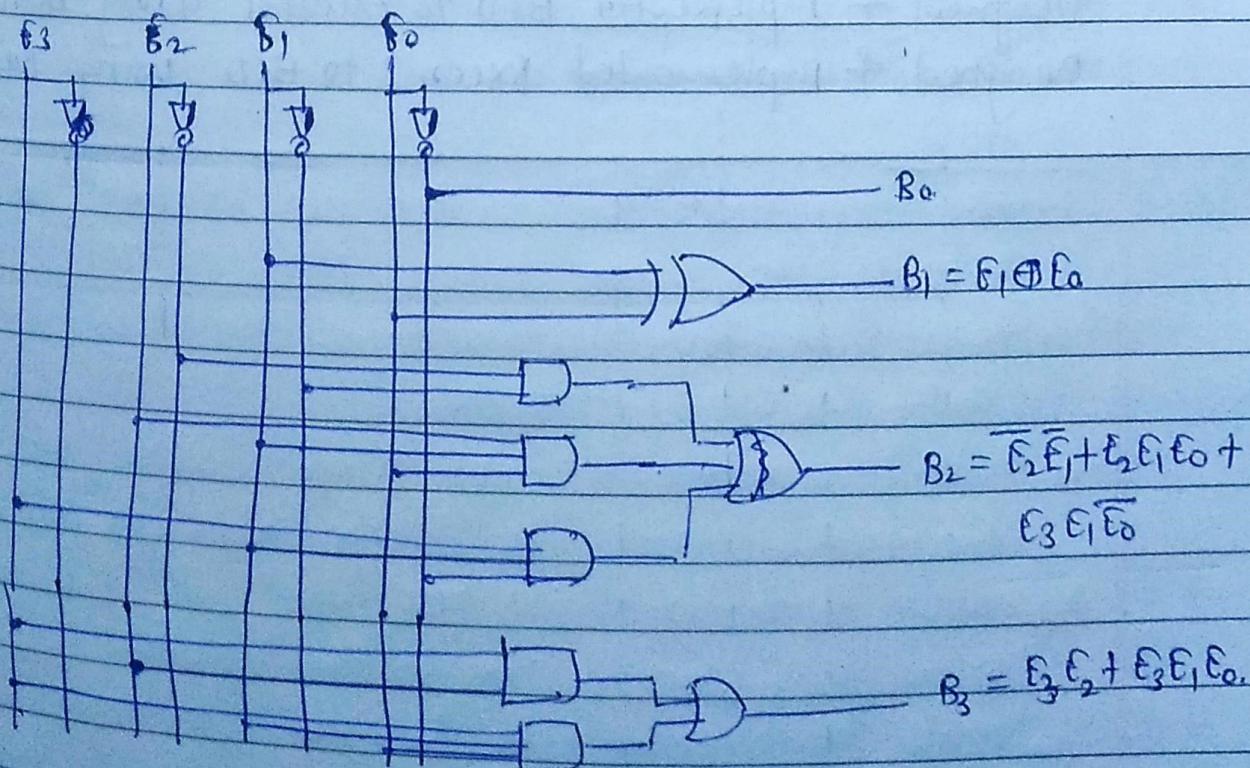
$$B_2 = \bar{E}_2 \bar{E}_1 + E_2 E_1 \bar{E}_0 + E_2 \bar{E}_1 \bar{E}_0$$

$B_3 \Rightarrow$

		00	01	11	10
		$E_3 E_0$	X	X	X
		00		1	1
		01		1	1
		11	1	X	X
		10	1	1	X

$$B_3 = E_3 E_2 + \bar{E}_3 E_1 \bar{E}_0$$

Logic diagram



Logic gates / MSI device required for implementation

S.No.	Title	Name of the IC	Number of Gates	IC required
01	Binary to Gray Code	Quad 2-input XOR 7486	3	1
	Gray to Binary Code	Quad - 2-input XOR - 7486	3	1
02	BCD to Excess 3 Code	HEX INVERTER 7404	3	1
		Quad 2-input AND 7408	4	1
		Quad-2-input OR 7432	4	1
	Excess-3 to BCD	HEX INVERTER 7404	3	1
		Quad-2-input AND 7408	9	3
		Quad-2-input-OR - 7432	4	1

Conclusion

Designed & Implemented BCD to Graycode using XNOR gates.

Designed & implemented Graycode to BCD using XNOR gates

Designed & implemented BCD to excess-3 using basic gates

Designed & implemented Excess-3 to BCD using Basic gates