Name: MAHFUZA KHATUN

ID : 2112376111

Course code: CSE - 1212

Course Title: Introduction to Digital Electronics Lab

Session: 2020-21

Department name: Computer Science and Engineering

Design And Implementation of Code Converter:

AIM:

To design and implement 4-bit
i) Binary to Gray Code Converter
ii) Gray to Binary Code Converter
iii) Binary to Excess-3 Code Converter
iv) Excess-3 to binary code converter

Theory

The availability of large variety of codes for the discrete elements of information results in the use of different codes by different systems. A Conversion circuit must be inserted between the two systems if each uses different codes for same information. Thus, Code Converter is a circuit that makes the two systems compitable even though each uses different binary code.

The bit combination assigned to binary code to gray code. Since each code uses four bits to represent a decimal digit. There are four inputs and four outputs. Gray code is a non-weighted code. The input variable are designed as B_3, B_2, B_1, B_0 and the output variables are designated as G_3, G_2, G_1, G_0 from the truth table, combinational circuit is designed. The boolean functions are obtained from K-map for each output variable.

A code converter is a circuit that makes the two systems compatible even though each uses a different binary code. To convert from binary to excess-3 code, the input line must supply the bit combination of elements as specified by code and the output lines generate the corresponding bit combination of code.

A two-level logic diagram may be obtained directly from the Boolean expressio derived by maps. These are various other possibilities for a logic diagram that implements this circuit. Now the OR gate whose output is C+D has been used to implement partially each of three outputs.

Apparatus:

1. XOR gate IC-7486
2. AND gate IC - 7408
3. OR gate IC- 7432
4. NOT gat IC - 7404
5. IC Trainer kit
6. Bread board
7. wires

LOGIC DIAGRAM :

1) Binary to Gray code converter

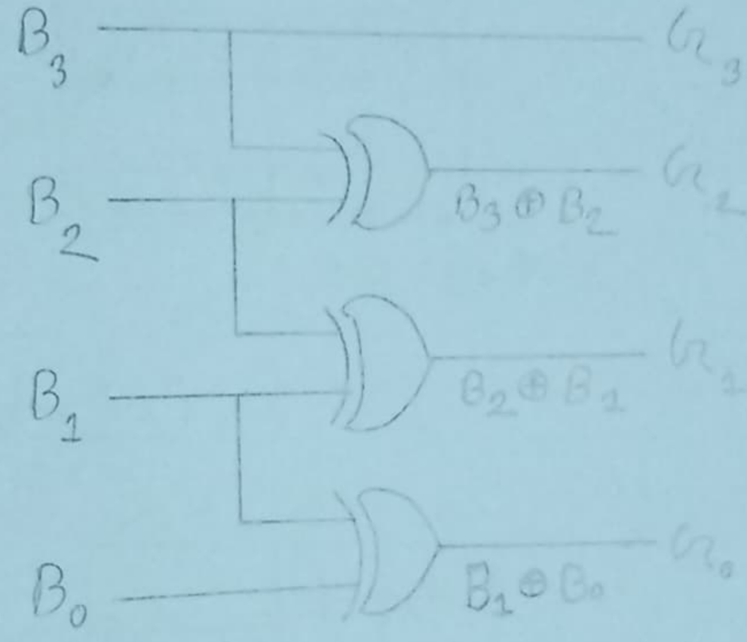

Fig: Logic gate for Binary to Gray

K-Map for G_3 :

B_3B_2 \ B_1B_0	00	01	11	10
00				
01				
11	1	1	1	1
10	1	1	1	1

$G3 = B3$

K-map for G2:

B_3B_2 \ B_1B_0	00	01	11	10
00				
01	1	1	1	1
11				
10	1	1	1	1

$G_2 = B_3 \oplus B_2$

K map for G1:

B_3B_2 \ B_1B_0	00	01	11	10
00			1	1
01	1	1		
11	1	1		
10			1	1

$G_1 = B_1 \oplus B_2$

K map for G0:

B_3B_2 \ B_1B_0	00	01	11	10
00		1		1
01		1		1
11		1		1
10		1		1

$G_0 = B_1 \oplus B_0$

Truth Table:

Binary Input				Gray Code output			
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

Logic Diagram:

Gray code to Binary converter

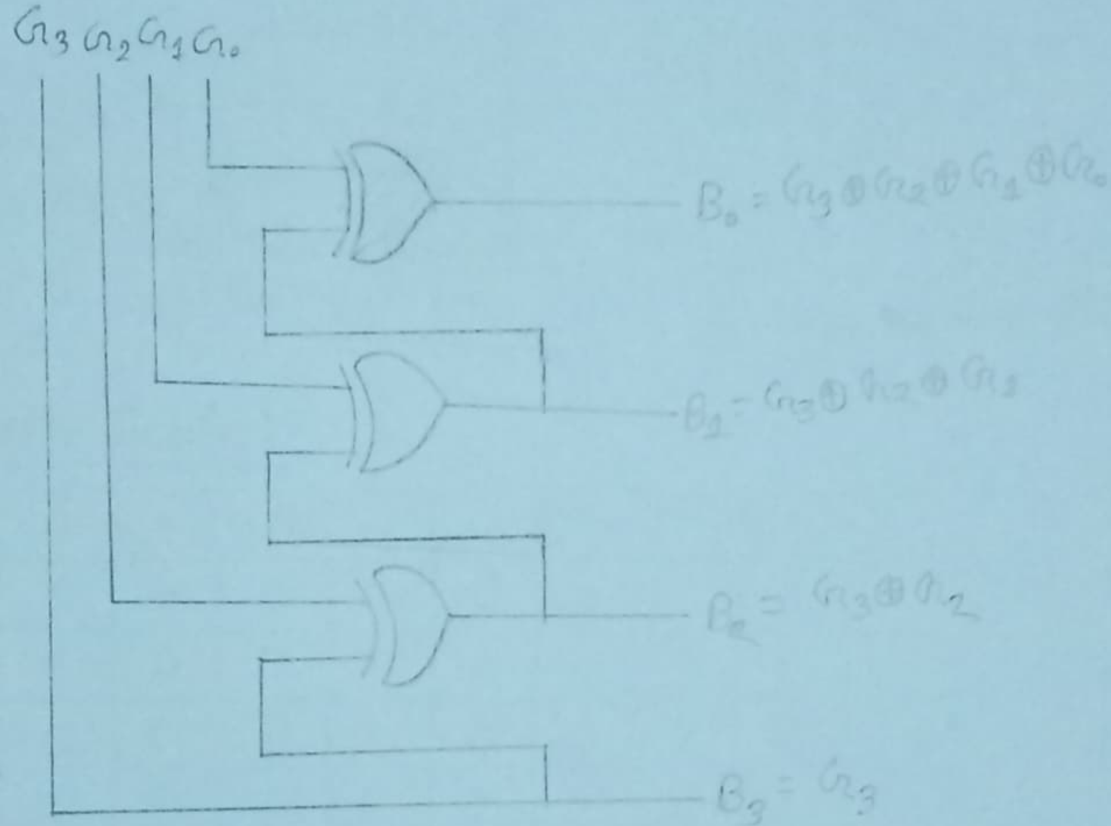

Fig: Gray to binary

K-map for B_3:

G_3G_2 \ G_1G_0	00	01	11	10
00				
01				
11	1	1	1	1
10	1	1	1	1

$B_3 = G_3$

K map for B_2:

G_3G_2 \ G_1G_0	00	01	11	10
00				
01	1	1	1	1
11				
10	1	1	1	1

$B_2 = G_3 \oplus G_2$

K-Map for B_1:

$G_3G_2 \backslash G_1G_0$	00	01	11	10
00			$\boxed{1}$	$\boxed{1}$
01	$\boxed{1}$	$\boxed{1}$		
11			$\boxed{1}$	$\boxed{1}$
10	$\boxed{1}$	$\boxed{1}$		

$B_1 = G_3 \oplus G_2 \oplus G_1$

K-Map for B_0:

$G_3G_2 \backslash G_1G_0$	00	01	11	10
00		1		1
01	1		1	
11		1		1
10	1		1	

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

Truth Table:

Gray Code				Binary Code			
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	1	0	0	1	0
0	0	1	0	0	0	1	1
0	1	1	0	0	1	0	0
0	1	1	1	0	1	0	1
0	1	0	1	0	1	1	0
0	1	0	0	0	1	1	1
1	1	0	0	1	0	0	0
1	1	0	1	1	0	0	1
1	1	1	1	1	0	1	0
1	1	1	0	1	0	1	1
1	0	1	0	1	1	0	0
1	0	1	1	1	1	0	1
1	0	0	1	1	1	1	0
1	0	0	0	1	1	1	1

Logic Diagram:

i) Binary to Excess-3 Code Converter

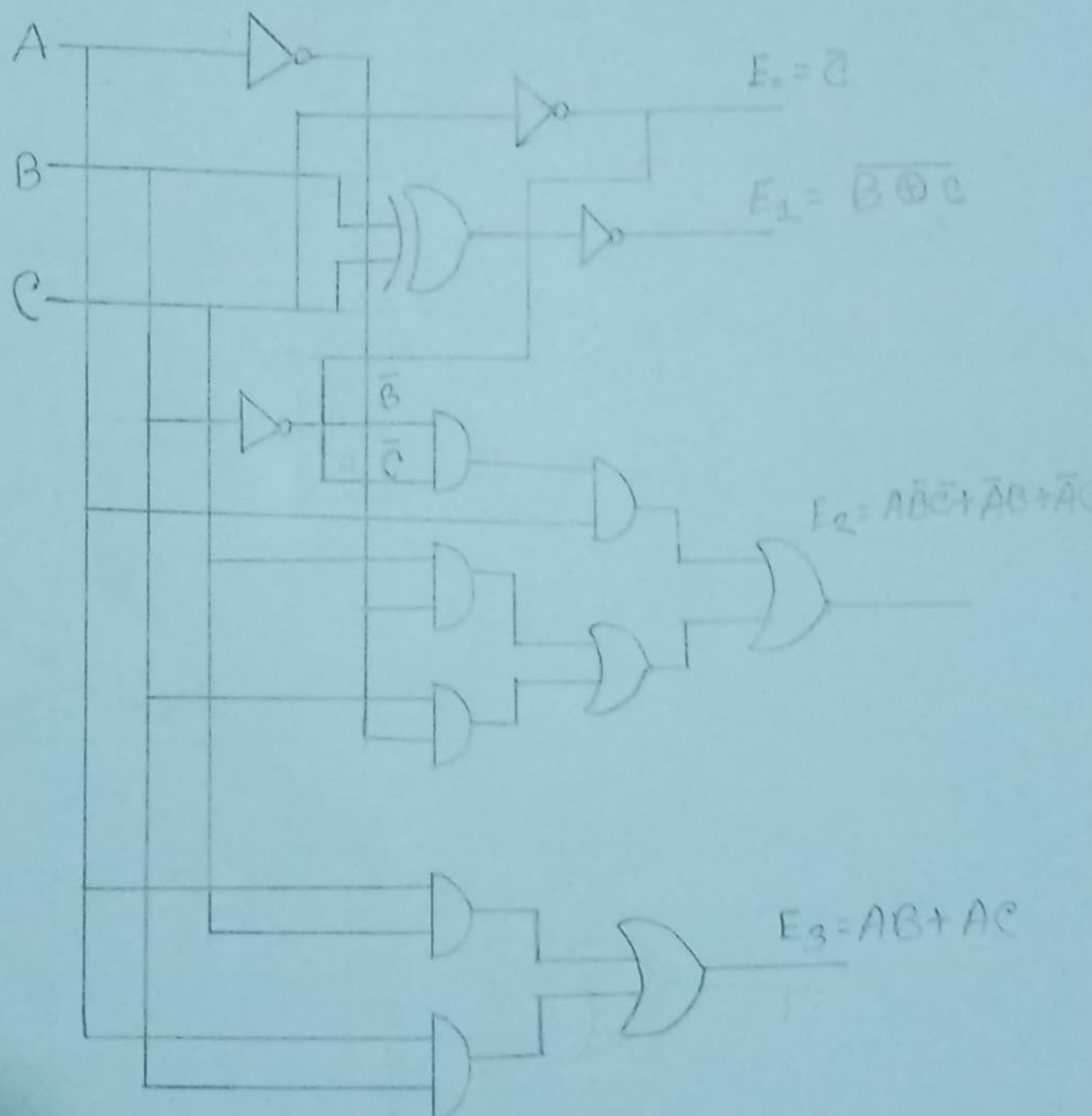

Fig: 3-bit binary to excess+3 Code Converter

K-map for F_0 :

A \ BC	00	01	11	10
0	1			1
1	1			1

$= \bar{C}$

K-map for E_1 :

A \ BC	00	01	11	10
0	1		1	
1	1		1	

$= \bar{B}\bar{C} + BC$

$= \overline{B \oplus C}$

K-map for E_2 :

A \ BC	00	01	11	10
0		1	1	1
1	1			

$= A\bar{B}\bar{C} + \bar{A}B + \bar{A}C$

K-map for E_3 :

A \ BC	00	01	11	10
0				
1		1	1	1

$= AB + AC$

Truth table:

3-bit Binary			Excess-3 Code			
A	B	C	E_3	E_2	E_1	E_0
0	0	0	0	0	1	1
0	0	1	0	1	0	0
0	1	0	0	1	0	1
0	1	1	0	1	1	0
1	0	0	0	1	1	1
1	0	1	1	0	0	0
1	1	0	1	0	0	1
1	1	1	1	0	1	0

Working Procedure:

1. Binary to Gray Converter:

i) Firstly we checked the components and all IC (logic gates) for experiment.

ii) We separated XOR gate (IC 7486) for binary to Gray.

iii) We connected the wires according to the logic circuit.

iv) Finally, we checked all inputs according to the truth table of binary to Gray.

2) Gray to Binary Converter:

i) Firstly we checked the all components and all IC (logic gates) for experiment.

ii) We separated XOR gate (IC 7486) for Gray to binary converter.

iii) We connected the wires according to the logic circuit.

iv) Finally, we checked all inputs according to the truth table of Gray to binary converter.

3) <u>3-bit binary to Excess+3 Code:</u>

1) Firstly we checked the components and logic gate for examined the binary to Excess+3 code converter.

II) We separated XOR gate (IC 7486), NOT gate (IC 7404), AND gate (IC 7408). OR gate (IC 7432).

III) we connected the wires according to the binary to excess+3 logic circuit.

IV) Finally, we checked all inputs according to the truth table of binary to excess+3.

4) <u>Excess+3 to binary code converter:</u>

1) Firstly we checked the components and logic gate for experiment.

II) We separated XOR gate, AND gate, OR gate, NOT gate.

III) We connected the wires according to the Excess+3 to binary converter.

IV) Finally, we checked all inputs according to the truth table of Excess+3 code converter.

Results:

1. **Binary to Gray converter:**

1) We obtained the results using the K-map method —

$G_3 = B_3, \quad G_2 = B_3 \oplus B_2, \quad G_1 = B_1 \oplus B_2, \quad G_0 = B_1 \oplus B_0$

2) **Gray to binary converter:**

1) We obtained the results using the K-map method —

$B_3 = G_3, \quad B_2 = G_3 \oplus G_2, \quad B_1 = G_3 \oplus G_2 \oplus G_1,$

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

3) **3-bit binary to Excess+3 converter:**

1) We obtained the results using the K-map method —

$E_0 = \bar{C}, \quad E_1 = \overline{B \oplus C}$ or, $E_1 = B\bar{C} + BC,$

$E_2 = A\bar{B}\bar{C} + \bar{A}C + \bar{A}B, \quad E_3 = AB + AC$

Precausion:

i) Check the connection according to the logic circuit.

ii) The connection should be properly.

iii) Check the equipment befor starting the experiment.