D.Y. PATIL INTERNATIONAL UNIVERSITY
B.TECH CSE FY SEM-2
A.Y. 2022-2023

Name: Suryakant Upadhyay
PRN: 20220802043
Subject: DIGITAL LOGIC AND DESIGN
Batch: A1

EXPERIMENT-3 (A)

Aim:
Design and implement Code converter- Binary to Gray.

Requirements:

NI LabVIEW tool

Theory:

Binary Code:
A symbolic representation of data/ information is called code. The base or radix of the binary number is 2. Hence, it has two independent symbols. The symbols used are 0 and 1. A binary digit is called as a bit. A binary number consists of sequence of bits, each of which is either a 0 or 1. Each bit carries a weight based on its position relative to the binary point. The weight of each bit position is one power of 2 greater than the weight of the position to its immediate right. e. g. of binary number is 100011 which is equivalent to decimal number 35.

BCD CODE:

Numeric codes represent numeric information i.e. only numbers as a series of 0's and 1's. Numeric codes used to represent decimal digits are called Binary Coded Decimal (BCD) codes. A BCD code is one, in which the digits of a decimal number are encoded-one at a time into group of four binary digits. There are a large number of BCD codes in order to represent decimal digits0, 1, 2,....,9, it is necessary to use a sequence of at least four binary digits. Such a sequence of binary digits which represents a decimal digit is called code word.

Gray Code:
It is a non-weighted code; therefore, it is not a suitable for arithmetic operations. It is a cyclic code because successive code words in this code differ in one bit position only i.e. it is a unit distance code.

Circuit Diagram:

Circuit Diagram

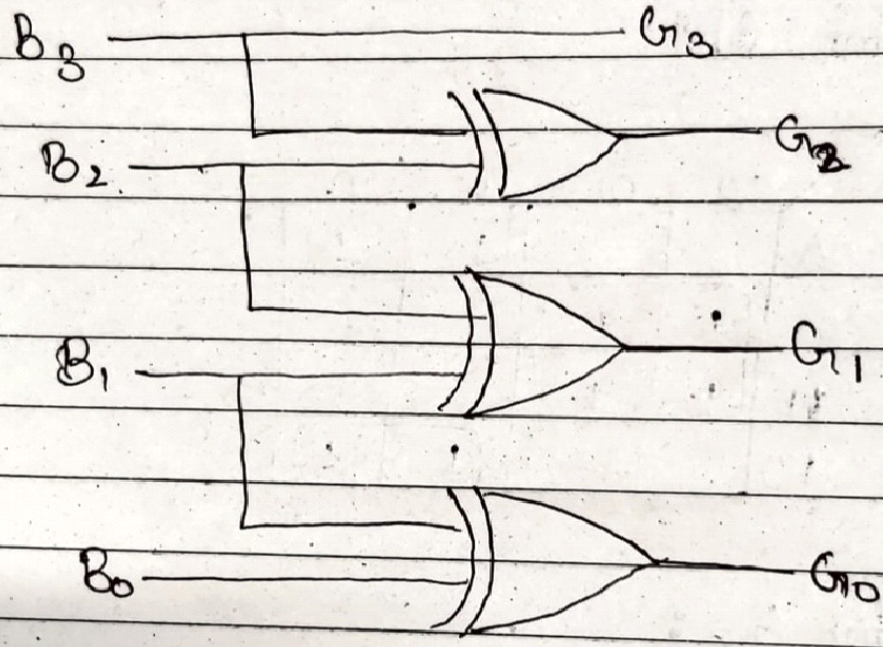


Fig : Binary to Gray converter.

Mathematical Equation.

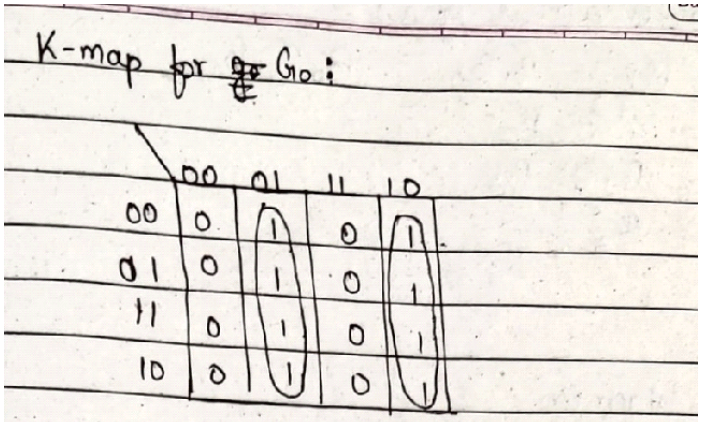
$G_0 = B_0B_1' + B_1B_0' = B_0 \oplus B_1$

$G_1 = B_1 \oplus B_2$

$G_2 = B_2 \oplus B_3$

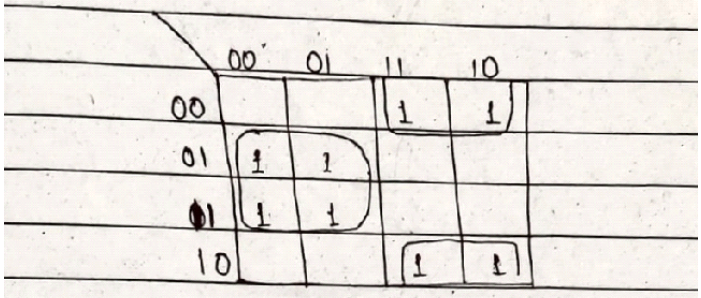
$G_3 = B_3$

K-map for G_0:



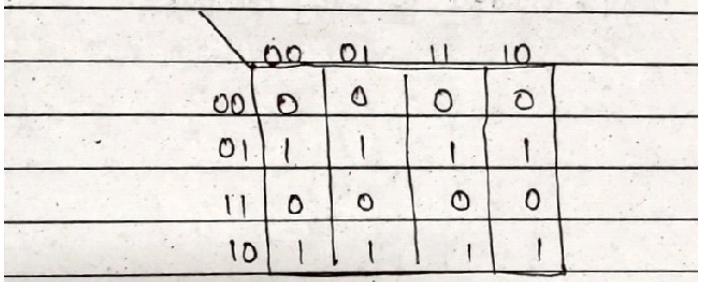
	00	01	11	10
00	0	1	0	1
01	0	1	0	1
11	0	1	0	1
10	0	1	0	1

K-map for G_1:



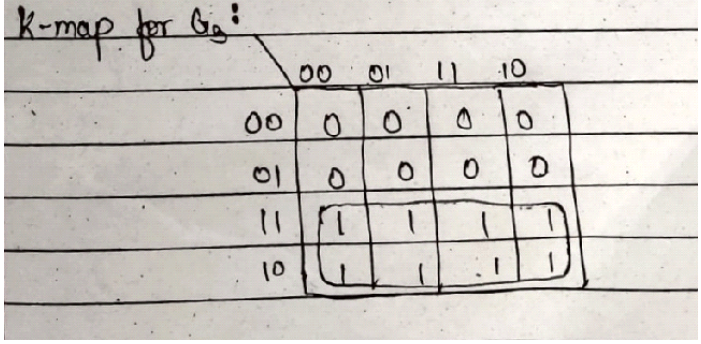
	00	01	11	10
00			1	1
01	1	1		
11	1	1		
10			1	1

K-map for G_2:



	00	01	11	10
00	0	0	0	0
01	1	1	1	1
11	0	0	0	0
10	1	1	1	1

K-map for G_3:



	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	1	1	1	1
10	1	1	1	1

Observation Table:

Input Variables- B_3, B_2, B_1, B_0

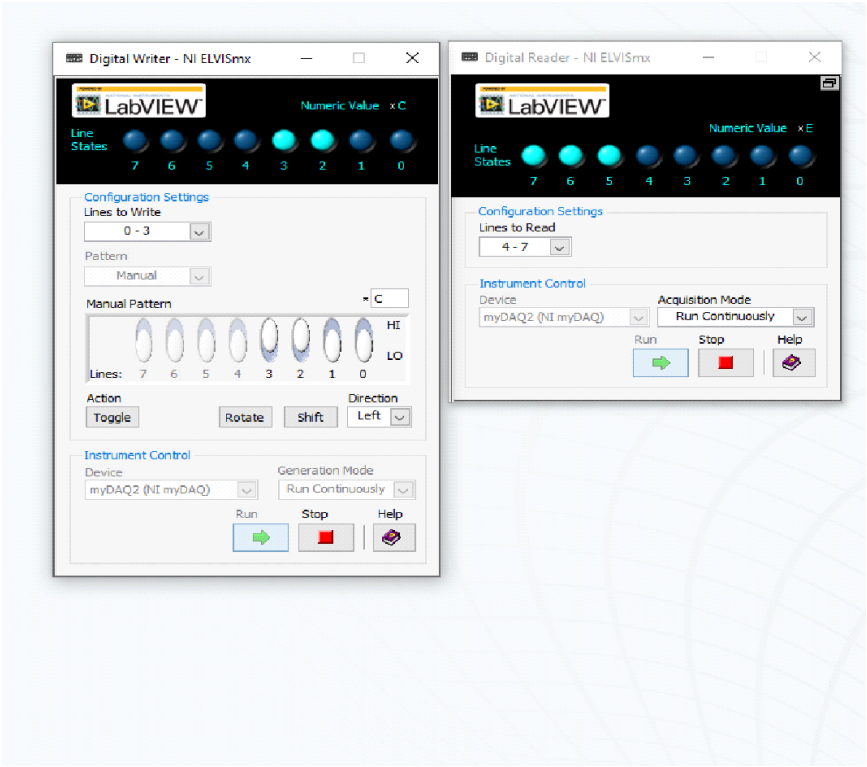
Output Variables- G_3, G_2, G_1, G_0

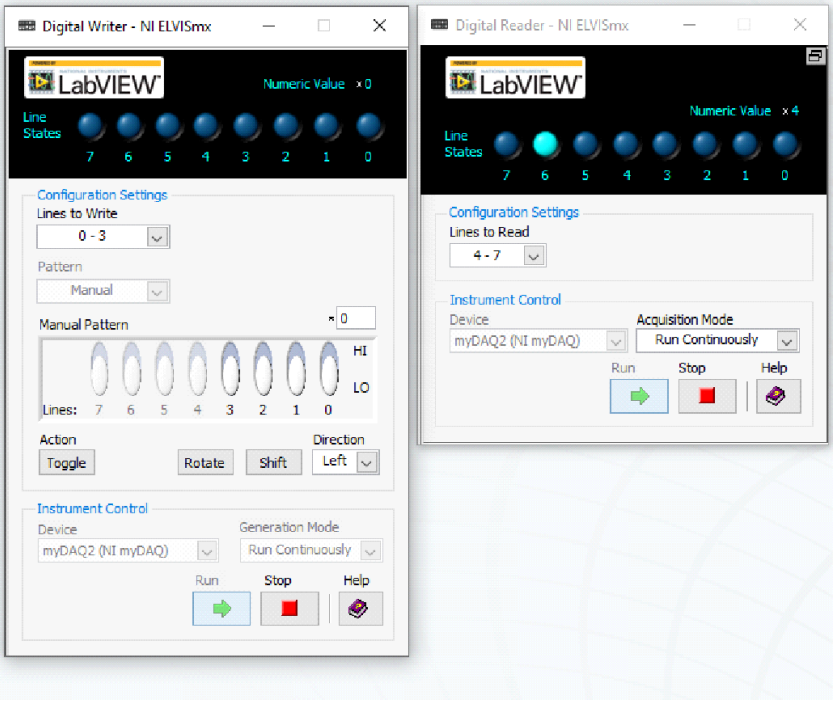
Where B_3, B_2, B_1, B_0 represents the binary bits,

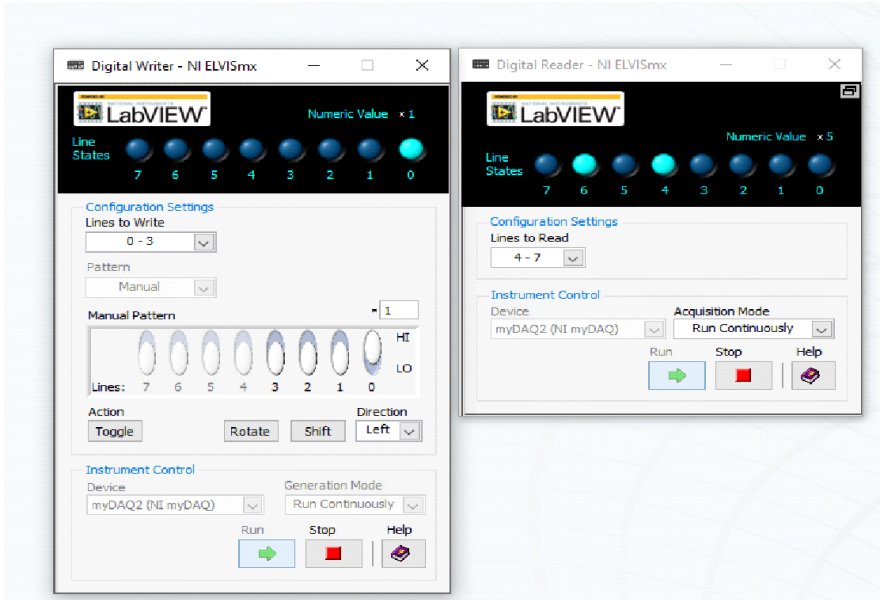
G_3, G_2, G_1, G_0 represents the gray code bits of binary number.

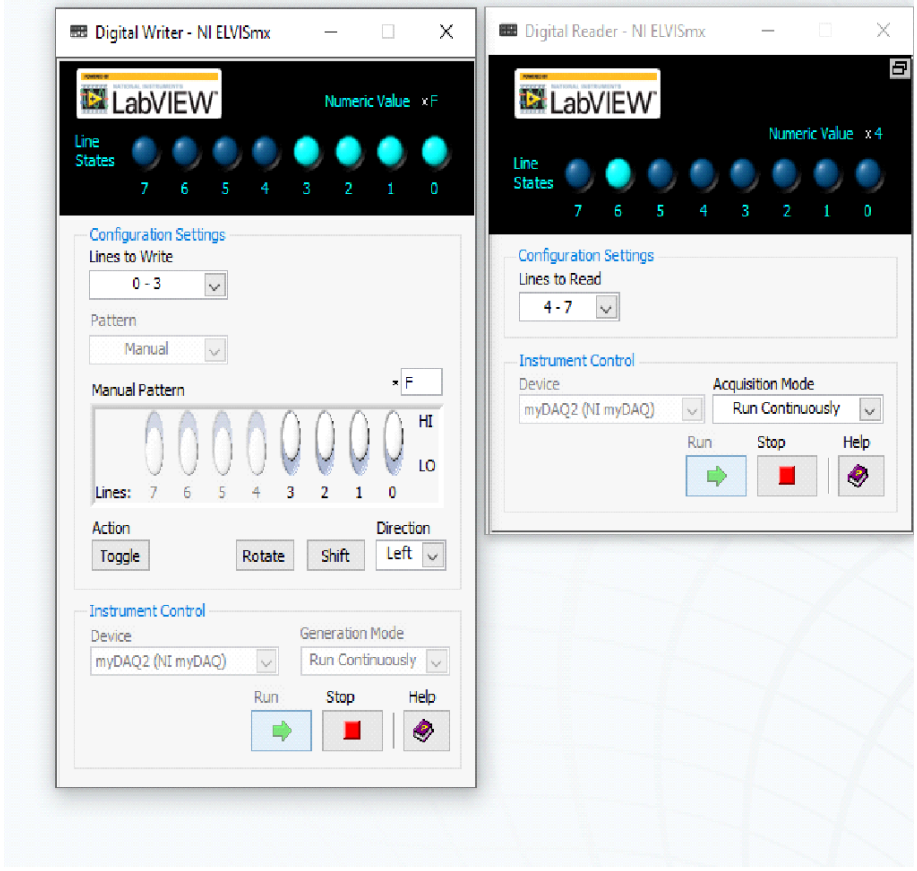
Input (Binary Code)				**Output (Gray Code)**			
B_3	**B_2**	**B_1**	**B_0**	**G_3**	**G_2**	**G_1**	**G_0**
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	0	0	1
0	1	1	1	0	0	0	0
1	0	0	0	1	0	0	0
1	0	0	1	1	0	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	1	1	0
1	1	0	1	1	1	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	0

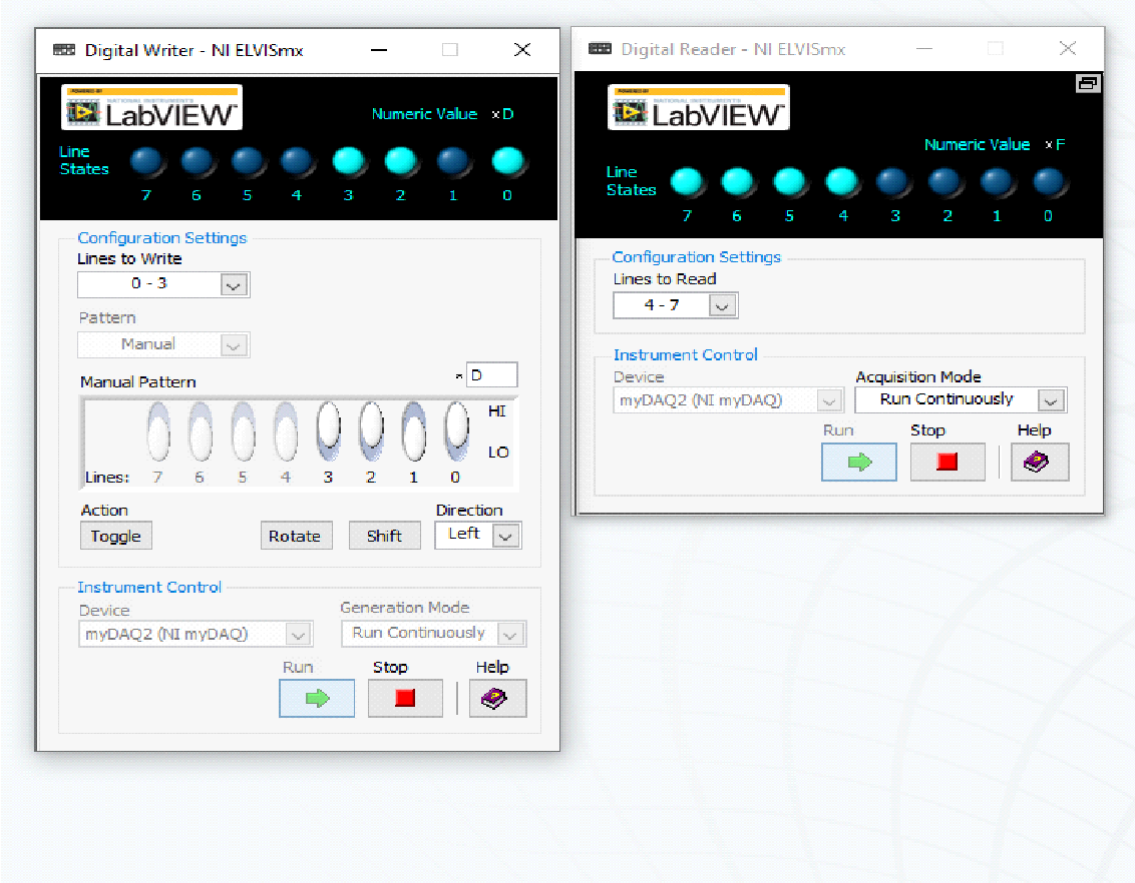
Output:


Digital Writer - NI ELVISmx
LabVIEW
Numeric Value x C
Line States
7 6 5 4 3 2 1 0
Configuration Settings
Lines to Write
0 - 3
Pattern
Manual
Manual Pattern
x C
HI
LO
Lines: 7 6 5 4 3 2 1 0
Action
Toggle
Rotate
Shift
Direction
Left
Instrument Control
Device
myDAQ2 (NI myDAQ)
Generation Mode
Run Continuously
Run
Stop
Help
Digital Reader - NI ELVISmx
LabVIEW
Numeric Value x E
Line States
7 6 5 4 3 2 1 0
Configuration Settings
Lines to Read
4 - 7
Instrument Control
Device
myDAQ2 (NI myDAQ)
Acquisition Mode
Run Continuously
Run
Stop
Help


Digital Writer - NI ELVISmx
LabVIEW
Numeric Value x 0
Line States
7 6 5 4 3 2 1 0
Configuration Settings
Lines to Write
0 - 3
Pattern
Manual
Manual Pattern
x 0
HI
LO
Lines: 7 6 5 4 3 2 1 0
Action
Toggle
Rotate
Shift
Direction
Left
Instrument Control
Device
myDAQ2 (NI myDAQ)
Generation Mode
Run Continuously
Run
Stop
Help
Digital Reader - NI ELVISmx
LabVIEW
Numeric Value x 4
Line States
7 6 5 4 3 2 1 0
Configuration Settings
Lines to Read
4 - 7
Instrument Control
Device
myDAQ2 (NI myDAQ)
Acquisition Mode
Run Continuously
Run
Stop
Help


Digital Writer - NI ELVISmx
LabVIEW
Numeric Value x 1
Line States
7 6 5 4 3 2 1 0
Configuration Settings
Lines to Write
0 - 3
Pattern
Manual
Manual Pattern
HI
LO
Lines: 7 6 5 4 3 2 1 0
Action
Toggle
Rotate
Shift
Direction
Left
Instrument Control
Device
myDAQ2 (NI myDAQ)
Generation Mode
Run Continuously
Run
Stop
Help
Digital Reader - NI ELVISmx
LabVIEW
Numeric Value x 5
Line States
7 6 5 4 3 2 1 0
Configuration Settings
Lines to Read
4 - 7
Instrument Control
Device
myDAQ2 (NI myDAQ)
Acquisition Mode
Run Continuously
Run
Stop
Help


Digital Writer - NI ELVISmx
LabVIEW
Numeric Value xF
Line States
7 6 5 4 3 2 1 0
Configuration Settings
Lines to Write
0 - 3
Pattern
Manual
Manual Pattern
HI
LO
Lines: 7 6 5 4 3 2 1 0
Action
Toggle
Rotate
Shift
Direction
Left
Instrument Control
Device
myDAQ2 (NI myDAQ)
Generation Mode
Run Continuously
Run
Stop
Help
Digital Reader - NI ELVISmx
LabVIEW
Numeric Value x4
Line States
7 6 5 4 3 2 1 0
Configuration Settings
Lines to Read
4 - 7
Instrument Control
Device
myDAQ2 (NI myDAQ)
Acquisition Mode
Run Continuously
Run
Stop
Help



Result:

The realization of Binary to Gray Code Converter is done.

EXPERIMENT-3 (B)

Aim:

Design and implement Code converter- BCD to Excess-3

Requirements:
NI LabVIEW tool

Theory:
Describe the Binary Coded Decimal (BCD) number system and Excess-3 code in brief. And explain the method to convert the BCD to its equivalent Excess-3 code with an example. Also draw truth table for the same.

Binary Coded Decimal (BCD):
Numeric codes represent numeric information i.e. only numbers as a series of 0's and 1's. Numeric codes used to represent decimal digits are called Binary Coded Decimal (BCD) codes. A BCD code is one, in which the digits of a decimal number are encoded-one at a time into group of four binary digits. There are a large number of BCD codes in order to represent decimal digits0, 1, 2,....,9, it is necessary to use a sequence of at least four binary digits. Such a sequence of binary digits which represents a decimal digit is called code word.

Excess-3 Code:
It is a non-weighted code. It is also a self-complementing BCD code used in decimal arithmetic units. . The Excess-3 code for the decimal number is performed in the same manner as BCD except that decimal number 3 is added to the each decimal unit before encoding it to binary.

BCD to Excess-3:
To convert from binary code A to binary code B, the input lines must supply the bit combination of elements as specified by code A and the output lines must generate the corresponding bit combination of code B. A combinational circuit performs this transformation by means of logic gates. As we want to design 4-bit code, we must use four input variables and four output variables. Designate the four input binary variables by the symbols B0, B1, B2 and B3, and the four output variables by E0, E1, E2 and E3. The truth table relating the input and output variables is as shown. A two-level logic diagram may be obtained directly from the Boolean expressions derived by the maps. The expressions obtained may be manipulated for the purpose of using common gates for two or more outputs. This manipulation illustrates flexibility obtained with multiple-output systems when implemented with three or more levels of gates.

Circuit Diagram:

Mathematical Equation.

$E_0 = B_0$

$E_1 = B_1B_0 + B_1'B_0' = B_1B_0(B_1B_0)'$

$E_2 = B_2'B_1 + B_2'B_0 + B_2B_1'B_0'$

$= B_2'(B_1 + B_0)$

$E_2 = B_2'(B_1 + B_0) + B_2(B_1 + B_0)'$

$E_3 = B_2'(B_1 + B_0) + B_2(B_1 + B_0)'$

$= B_3 + B_2B_1 + B_2B_0.$

$= B_3 + B_2(B_1 + B_0)$

Circuit Diagram:

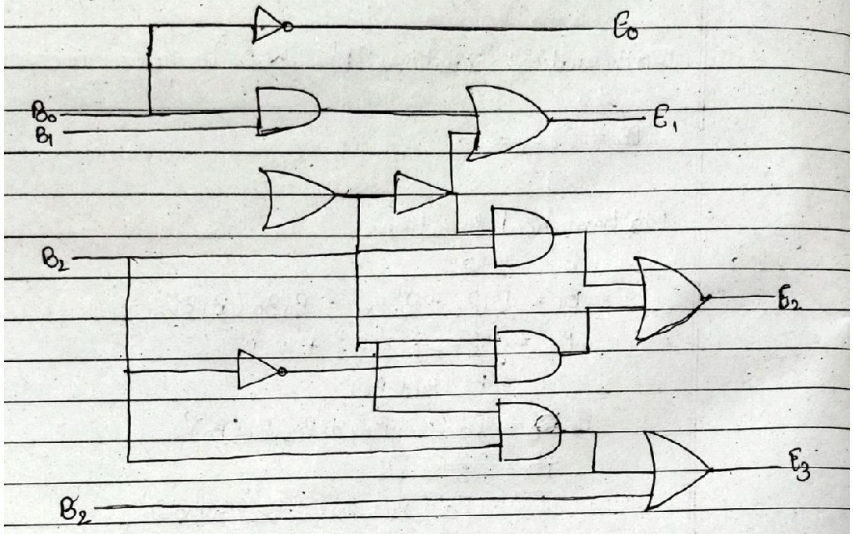
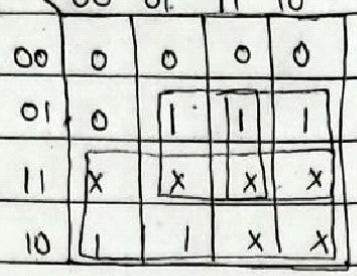


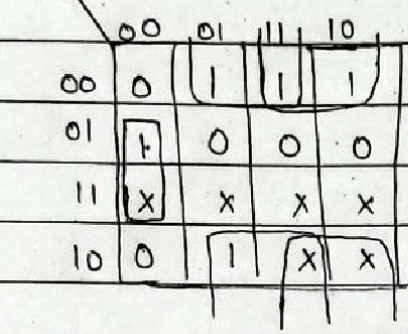
Fig: BCD TO EXCESS-3code converter.

K-map for E_3



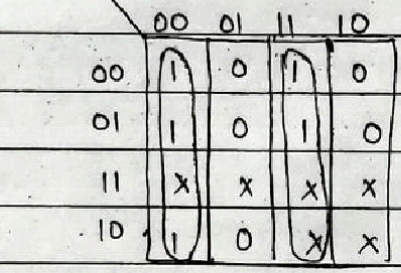
	00	01	11	10
00	0	0	0	0
01	0	1	1	1
11	x	x	x	x
10	1	1	x	x

K-map for E_2



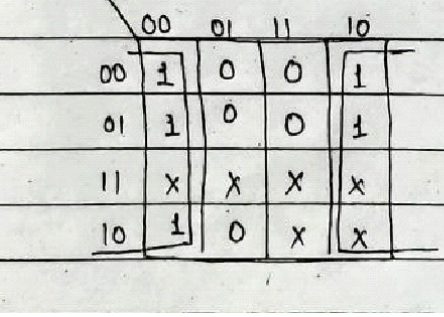
	00	01	11	10
00	0	1	1	1
01	1	0	0	0
11	x	x	x	x
10	0	1	x	x

K-map for E_1



	00	01	11	10
00	1	0	1	0
01	1	0	1	0
11	x	x	x	x
10	1	0	x	x

K-map for E_0



	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	x	x	x	x
10	1	0	x	x

Observation Table:

Input Variables- B_3, B_2, B_1, B_0

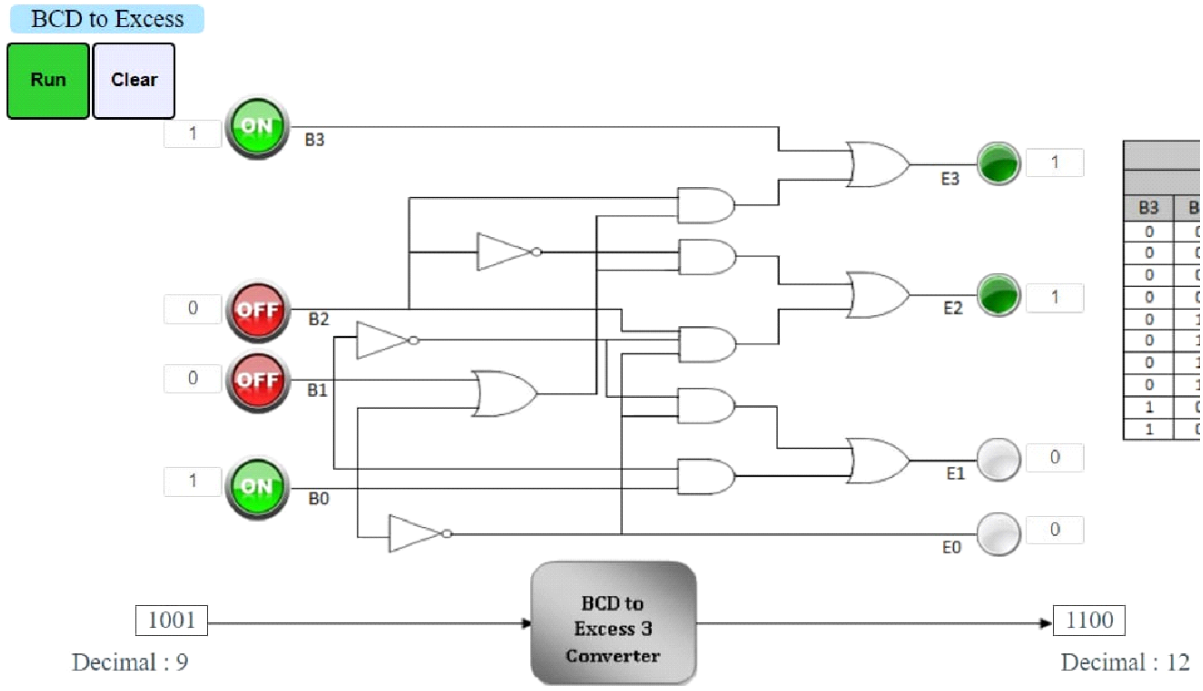
Output Variables- E_3, E_2, E_1, E_0

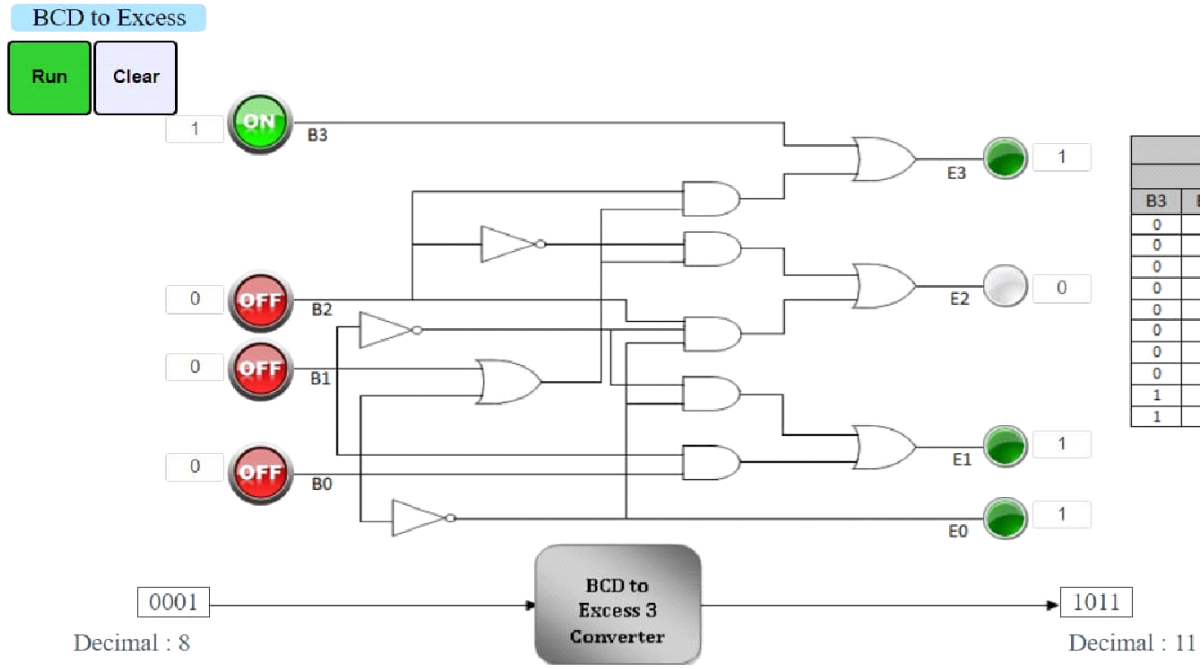
Where B_3, B_2, B_1, B_0 represents the bits of BCD code,

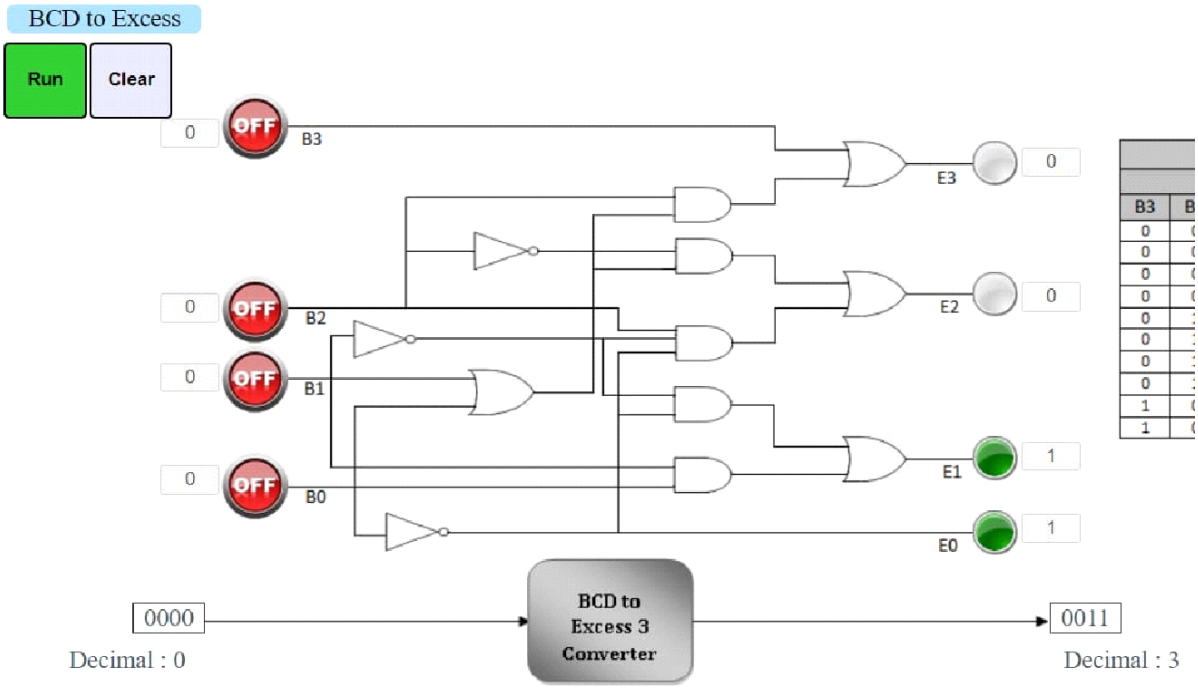
E_3, E_2, E_1, E_0 represents the bits of Excess-3 code.

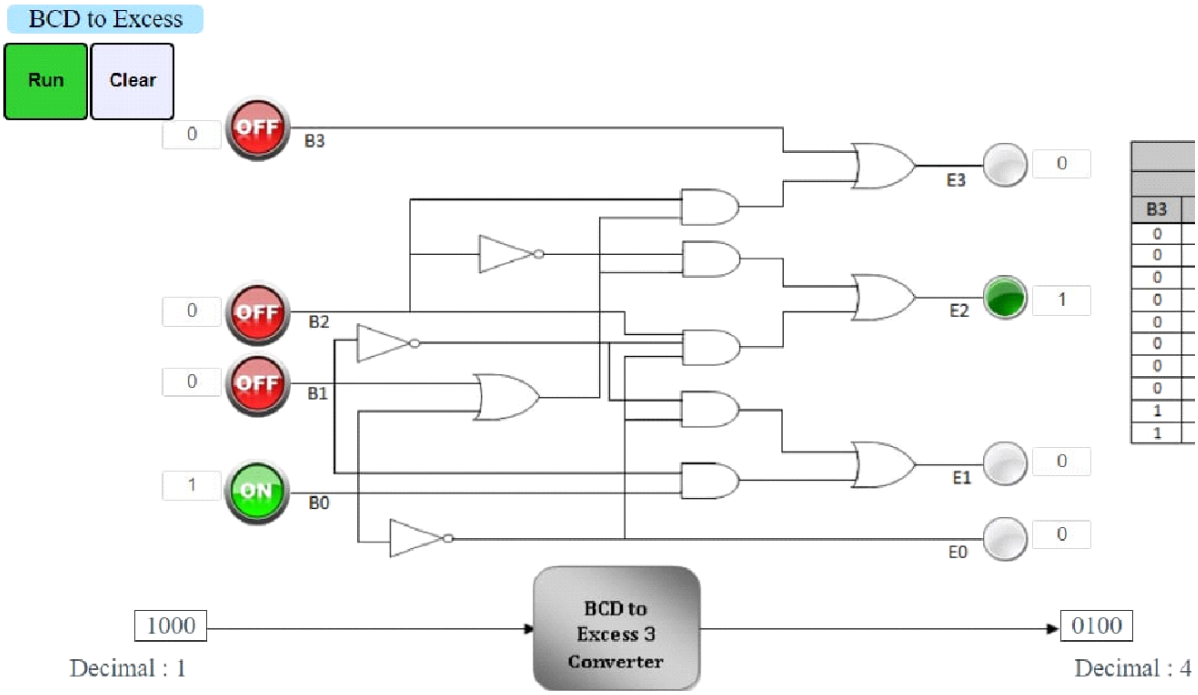
Input (BCD Code)				**Output (Excess-3 Code)**			
B_3	**B_2**	**B_1**	**B_0**	**E_3**	**E_2**	**E_1**	**E_0**
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0
1	0	1	0	X	X	X	X
1	0	1	1	X	X	X	X
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

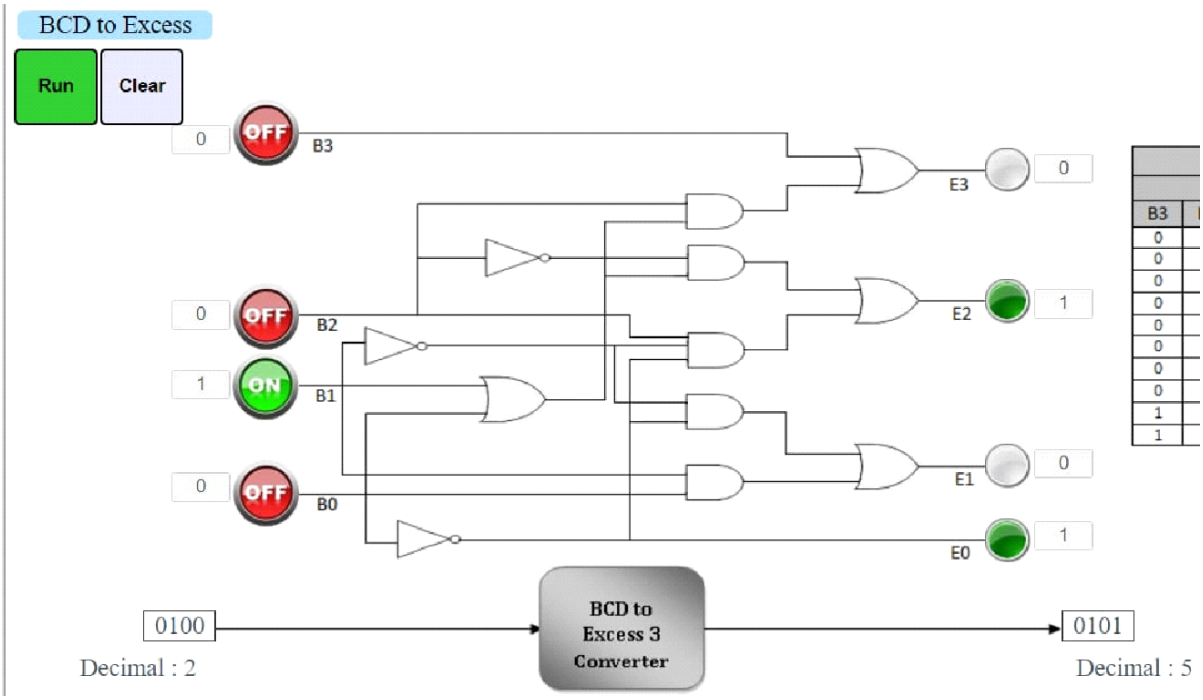
Output:


BCD to Excess
Run
Clear
1
ON
B3
0
OFF
B2
0
OFF
B1
1
ON
B0
E3
1
E2
1
E1
0
E0
0
B3
0
0
0
0
0
0
0
0
1
1
1001
BCD to Excess 3 Converter
1100
Decimal : 9
Decimal : 12


BCD to Excess
Run
Clear
1
ON
B3
0
OFF
B2
0
OFF
B1
0
OFF
B0
E3
1
E2
0
E1
1
E0
1
B3
0
0
0
0
0
0
0
0
1
1
0001
BCD to Excess 3 Converter
1011
Decimal : 8
Decimal : 11


BCD to Excess
Run
Clear
0
OFF
B3
E3
0
0
OFF
B2
E2
0
0
OFF
B1
E1
1
0
OFF
B0
E0
1
B3
0
0
0
0
0
0
0
0
1
1
0000
BCD to Excess 3 Converter
0011
Decimal : 0
Decimal : 3


BCD to Excess
Run
Clear
0
OFF
B3
E3
0
0
OFF
B2
E2
1
0
OFF
B1
E1
0
1
ON
B0
E0
0
B3
0
0
0
0
0
0
0
0
1
1
1000
BCD to Excess 3 Converter
0100
Decimal : 1
Decimal : 4



Result:

The realization of BCD to Excess-3 Code Converter is done.

Note: The whole documentation is to be converted to pdf/word format before uploading it on the Moodle. Only the pdf/word format will be acceptable. The Submission has to be done on time. After the deadline no submission of the particular experiment will be acceptable.