Table of Contents

Objectives 3

Apparatus 3

Components 3

Theory Overview 3

DECODER 3

ENCODER 4

Logic Diagram for Encoder 4

Encoder Designing in Logisim 5

Truth Table for Encoder 5

Logic Diagram for Decoder 6

Decoder Designing in Logisim 6

- When Enabler is ON: 6
- When Enabler is OFF 7

Decoder Implementation on BreadBoard 7

Truth Table for Decoder 8

Procedure 8

Review Questions 9

Objectives

After completing this experiment, you will be able to:

- Design and construct Decoder and Encoder
- Verify their truth tables using logic gates

Apparatus

1. Power Supply
2. Breadboard

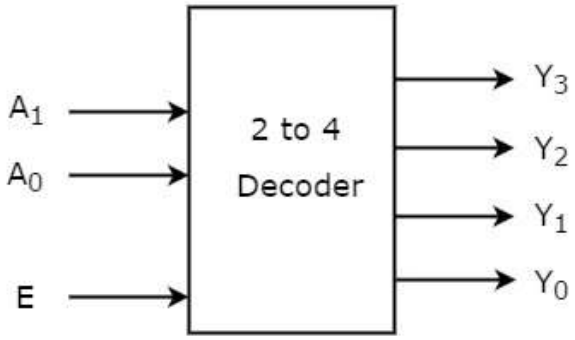
Components

1. Two 7410, 3 I/P NAND gate
2. Three 7432, 2 I/P OR gate
3. 7404 hex inverter

Theory Overview

DECODER

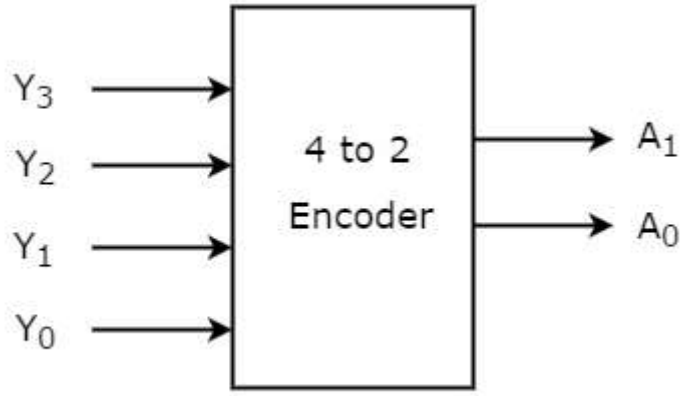
A Decoder is a multiple input multiple output logic circuit which converts coded input into coded output where input and output codes are different. The input code generally has fewer bits than the output code. Each input code word produces a different output code word i.e there is one to one mapping can be expressed in truth table. In the block diagram of Decoder circuit the encoded information is present as n input producing 2 n possible outputs. 2 n output values are



from 0 through out 2n – 1. One of these four outputs will be '1' for each combination of inputs when enable, E is '1'.

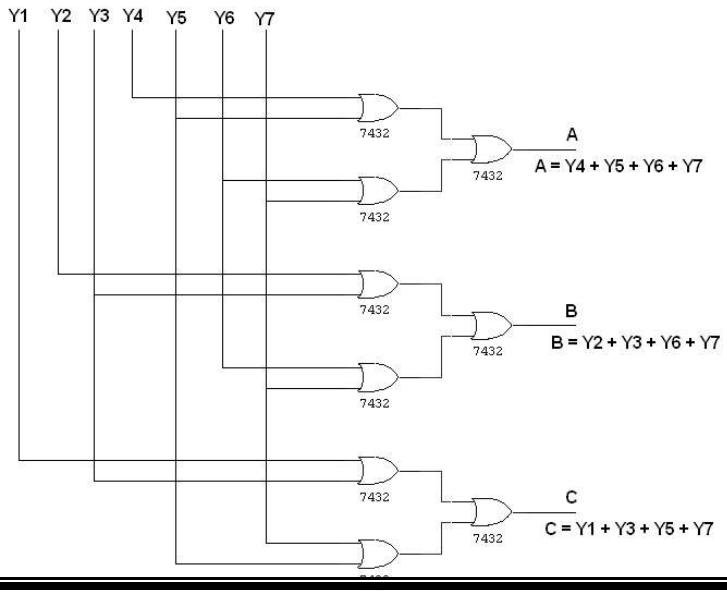
ENCODER

An Encoder is a digital circuit that perform inverse operation of a Decoder. An Encoder has 2 n input lines and n output lines. In Encoder the output lines generate the binary code corresponding to the input value. In octal to binary Encoder it has eight inputs, one for each octal digit and three outputs that generate the corresponding binary code. In Encoder it is assumed that only one input has a value of one at any given time otherwise the circuit is meaningless. It has an Abigayle that when all inputs are zero the outputs are zero. The zero outputs can also be generated when D0 = 1.



At any time, only one of these 4 inputs can be '1' in order to get the respective binary code at the output.

Logic Diagram for Encoder



Encoder Designing in Logisim

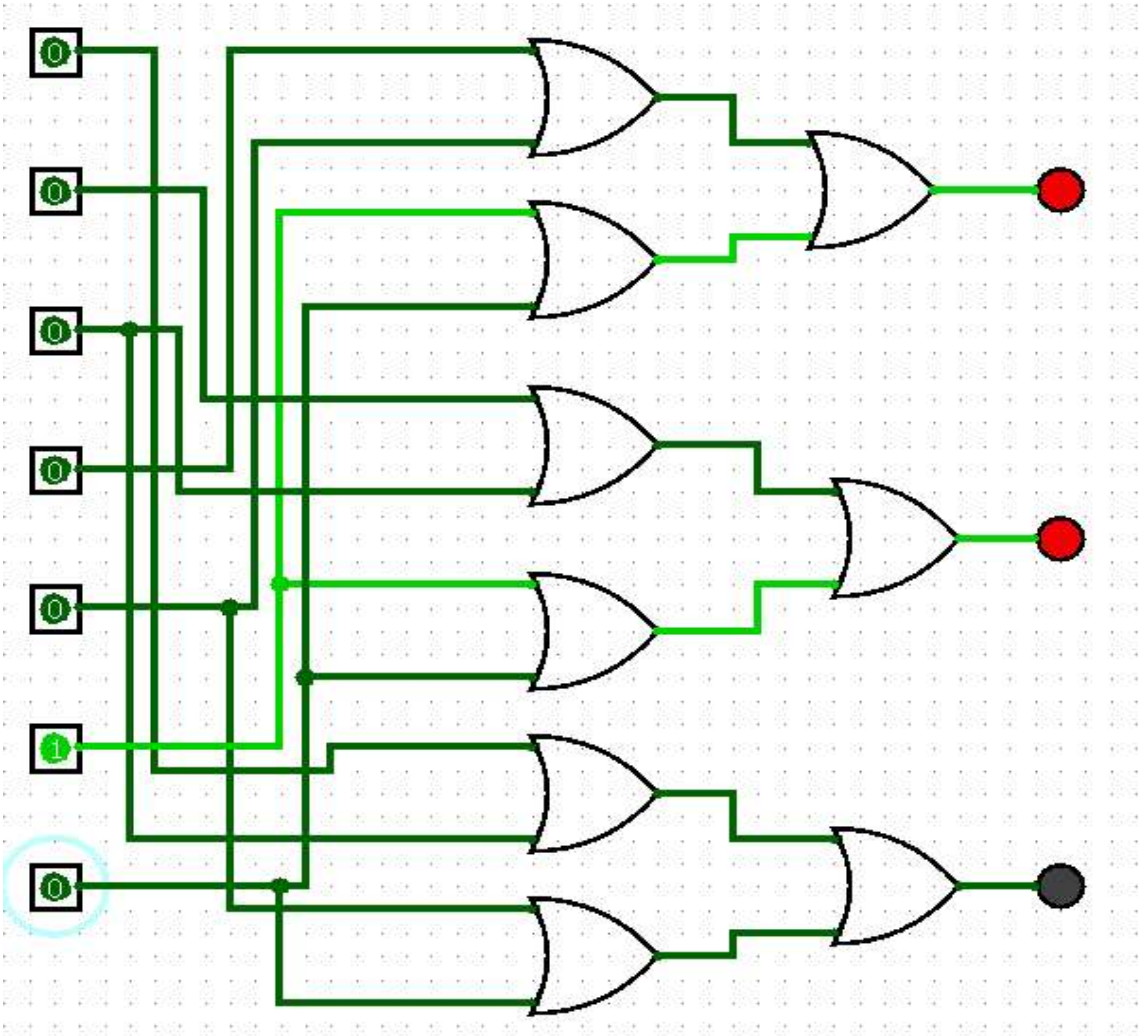
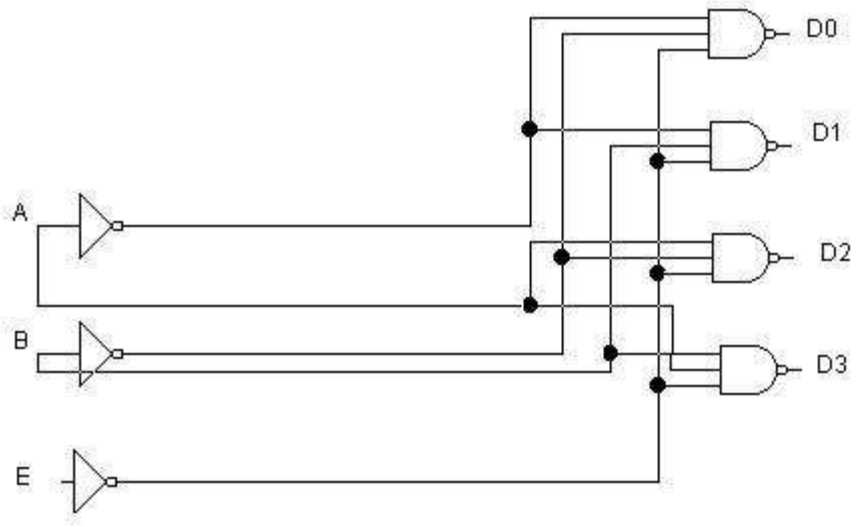


Figure : Encoder

Truth Table for Encoder

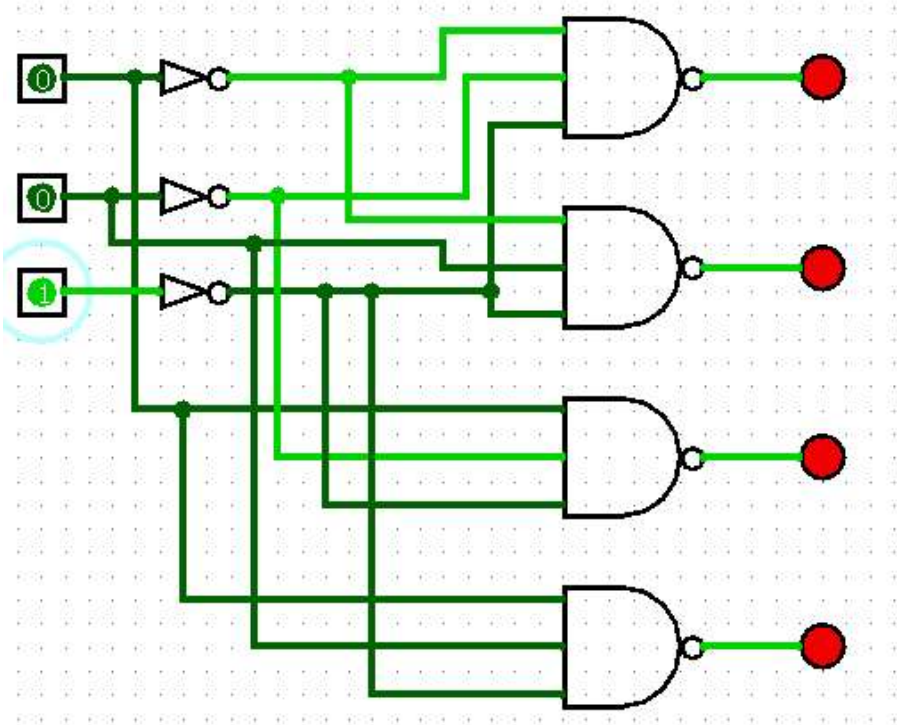
INPUT							OUTPUT		
Y1	**Y2**	**Y3**	**Y4**	**Y5**	**Y6**	**Y7**	**A**	**B**	**C**
1	**0**	**0**	**0**	**0**	**0**	**0**	0	0	1
0	**1**	**0**	**0**	**0**	**0**	**0**	0	1	0
0	**0**	**1**	**0**	**0**	**0**	**0**	0	1	1
0	**0**	**0**	**1**	**0**	**0**	**0**	1	0	0
0	**0**	**0**	**0**	**1**	**0**	**0**	1	0	1
0	**0**	**0**	**0**	**0**	**1**	**0**	1	1	0
0	**0**	**0**	**0**	**0**	**0**	**1**	1	1	1

Logic Diagram for Decoder

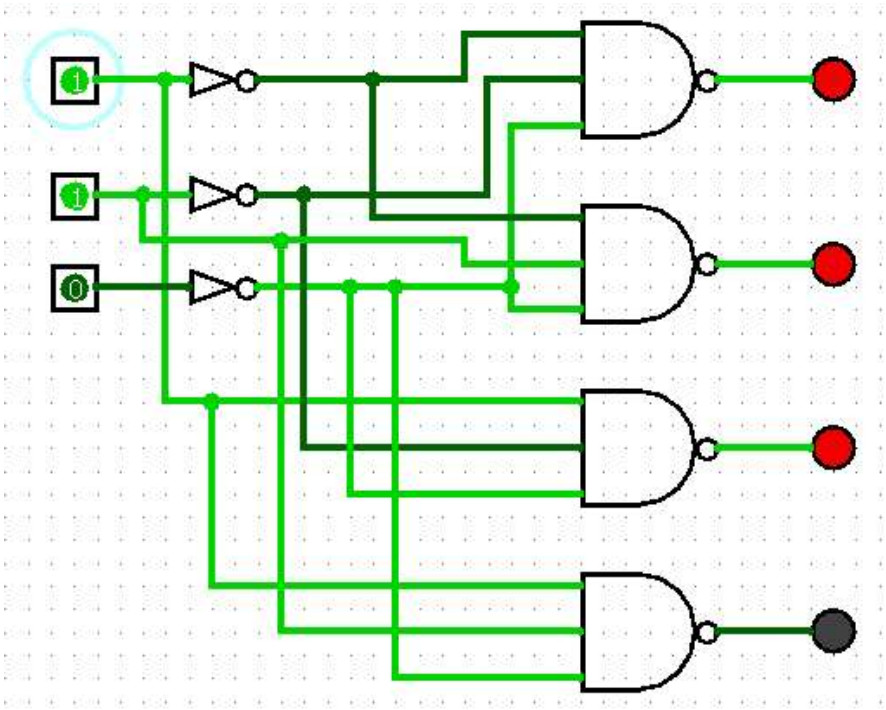


Decoder Designing in Logisim

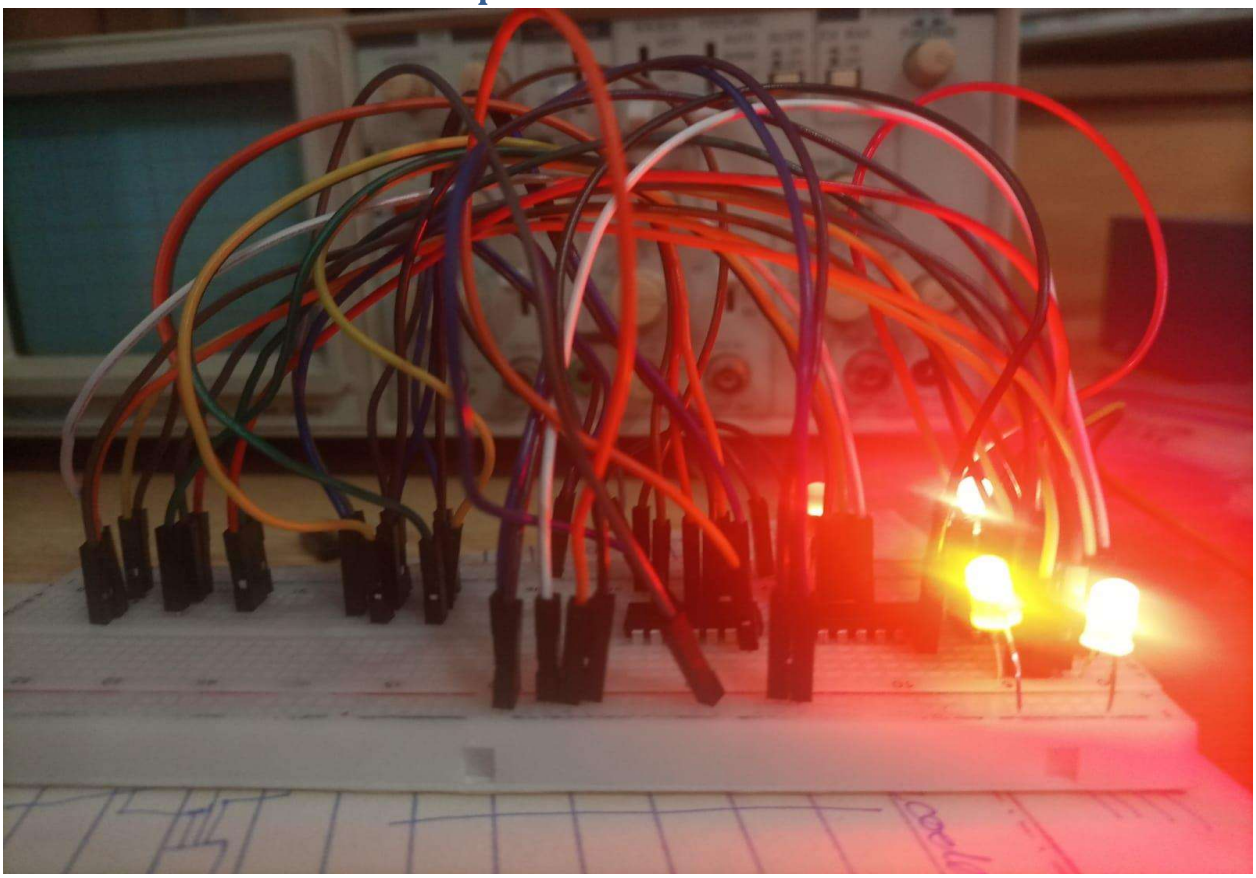
- When Enabler is ON:



- When Enabler is OFF



Decoder Implementation on BreadBoard



Truth Table for Decoder

INPUT			OUTPUT			
E	A	B	D0	D1	D2	D3
1	x	x	1	1	1	1
0	0	0	**0**	1	1	1
0	0	1	1	**0**	1	1
0	1	0	1	1	**0**	1
0	1	1	1	1	1	**0**

Procedure

1. Connections are given as per circuit diagram.
2. Logical inputs are given as per circuit diagram.
3. Observe the output and verify the truth table.

REVIEW QUESTIONS ARE NEXT PAGE

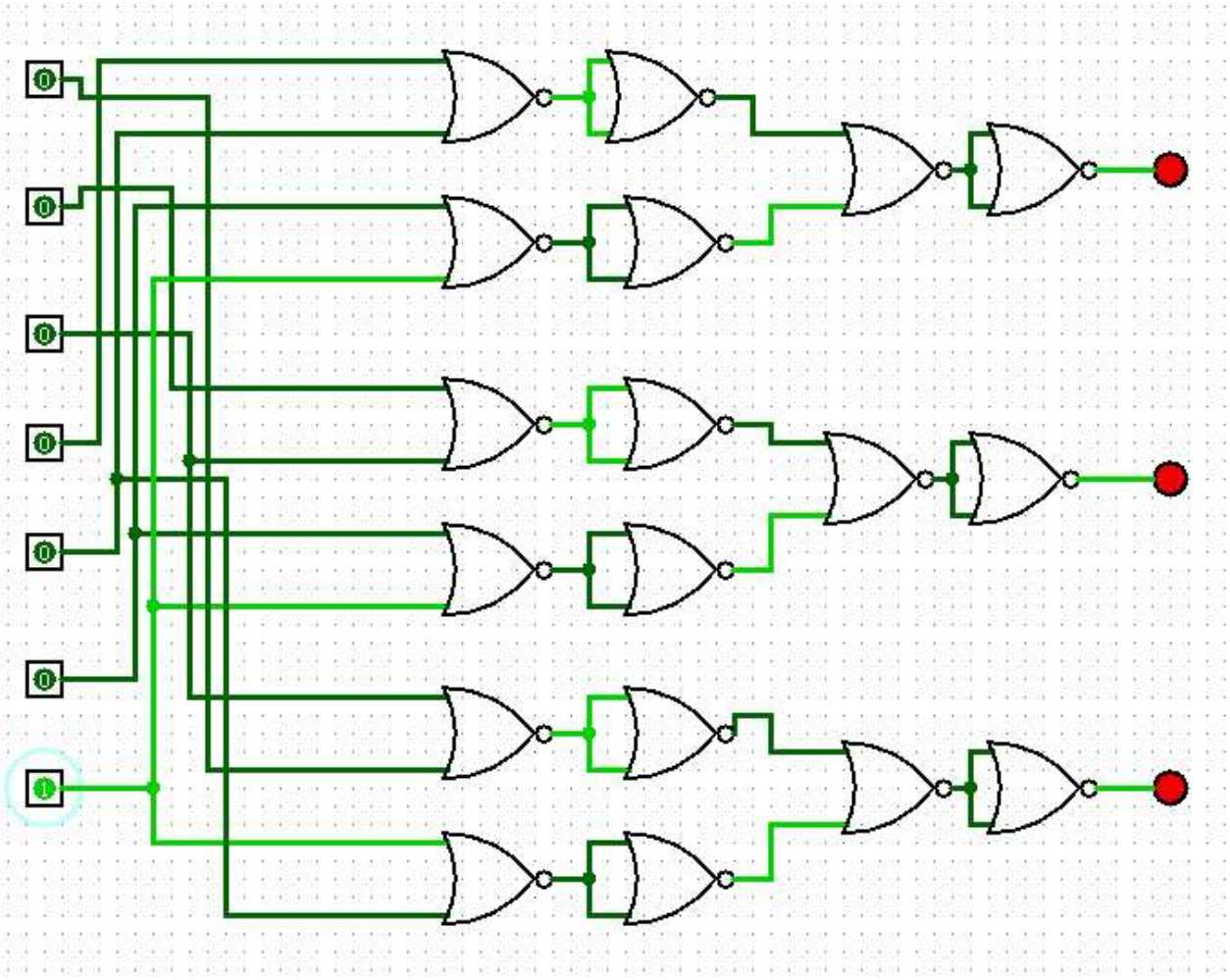
Review Questions

1. Design an Encoder using NOR gates only.

Truth Table

INPUT							OUTPUT		
Y1	**Y2**	**Y3**	**Y4**	**Y5**	**Y6**	**Y7**	**A**	**B**	**C**
1	**0**	**0**	**0**	**0**	**0**	**0**	0	0	1
0	**1**	**0**	**0**	**0**	**0**	**0**	0	1	0
0	**0**	**1**	**0**	**0**	**0**	**0**	0	1	1
0	**0**	**0**	**1**	**0**	**0**	**0**	1	0	0
0	**0**	**0**	**0**	**1**	**0**	**0**	1	0	1
0	**0**	**0**	**0**	**0**	**1**	**0**	1	1	0
0	**0**	**0**	**0**	**0**	**0**	**1**	1	1	1

Encoder Implemented using NOR Gates Only



2. What will be the output of the Decoder circuit if NAND gates are replaced by AND gates?

Truth Table

INPUT			OUTPUT			
E	A	B	D0	D1	D2	D3
1	x	x	0	0	0	0
0	0	0	**1**	0	0	0
0	0	1	0	**1**	0	0
0	1	0	0	0	**1**	0
0	1	1	0	0	0	**1**

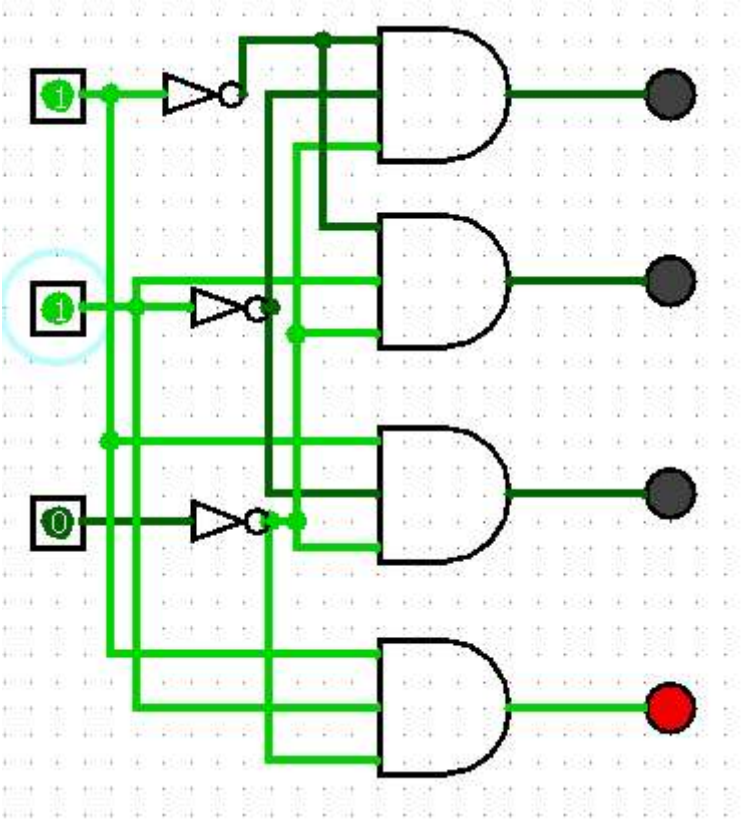


Figure : NAND gate replaced by AND Gates

3. What is the purpose of enable input in Decoder?

Enable Input:

The Enable input is used for Enabling the Decoder. It allows the decoder to be either enabled or disabled. In case, the decoder is not enabled, no matter whichever combination is given at the inputs of the decoder, it won't work.

The standard commercial practice is to design a decoder that is active–low and enabled–low. The enable input allows the decoder to be either enabled or disabled. For a decoder that is enabled low (Enable = 0 activates it) we have the following:

-Enable = 1 None of the outputs of the decoder are active.

-Enable = 0 Only the selected output is active; all others are inactive.

4. Design a 3x8 Decoder using two 2x4 Decoders (74LS139).

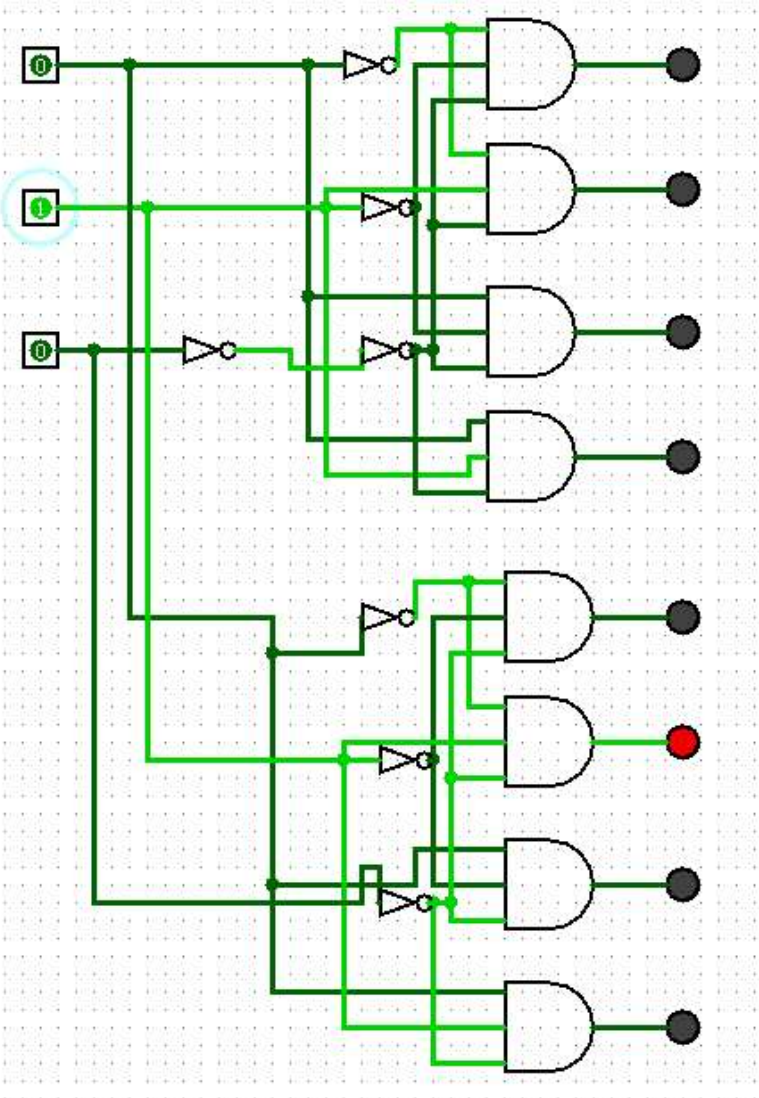


Figure :3x8 Decoder using two 2x4 Decoder