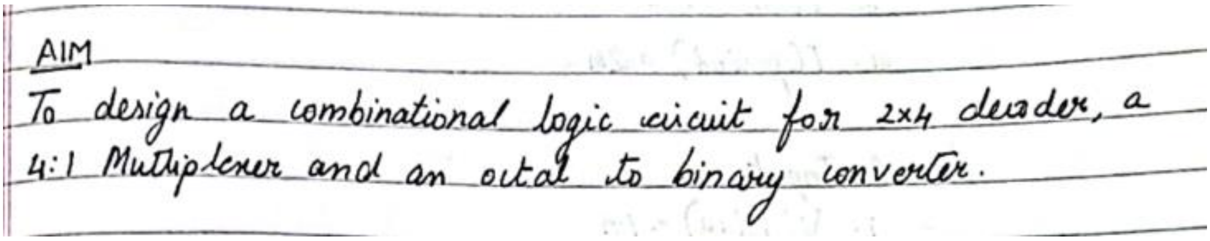
EXPERIMENT - IV

ANALYSIS OF COMBINATIONAL LOGIC CIRCUITS FOR 2X4 DECODER, 4:1 MULTIPLEXER AND OCTAL TO BINARY CONVERTER

SLOT: L37 + L38

HARSH AVINASH
20BDS0190
Instructor: Mrs. Rajalakshmi S

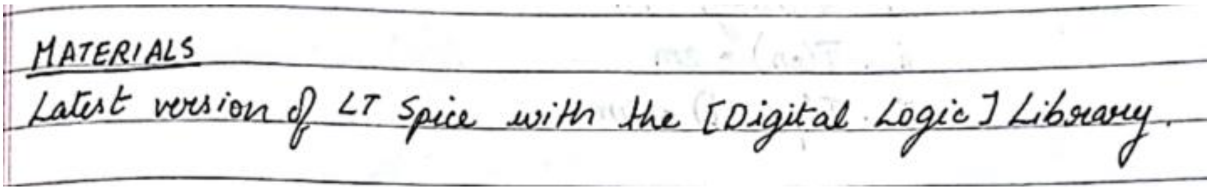
AIM



AIM

To design a combinational logic circuit for 2x4 decoder, a 4:1 Multiplexer and an octal to binary converter.

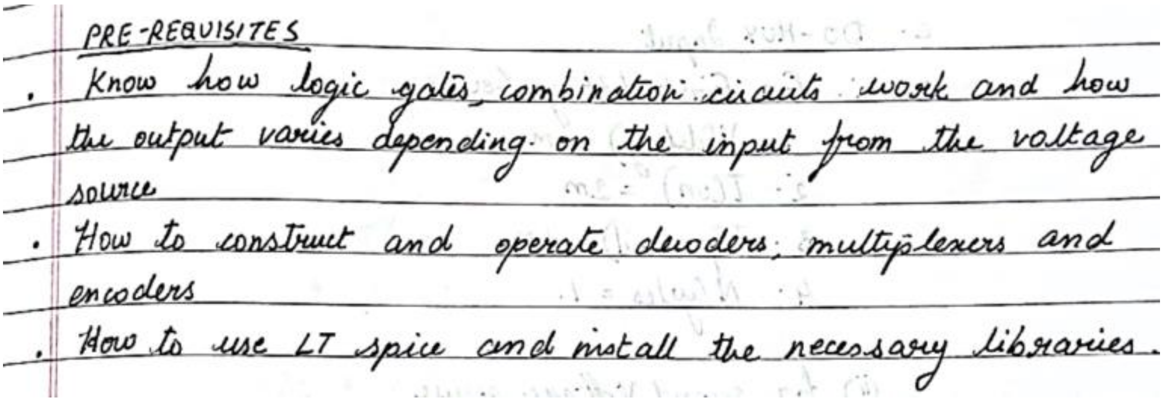
MATERIALS



MATERIALS

Latest version of LT Spice with the [Digital Logic] Library.

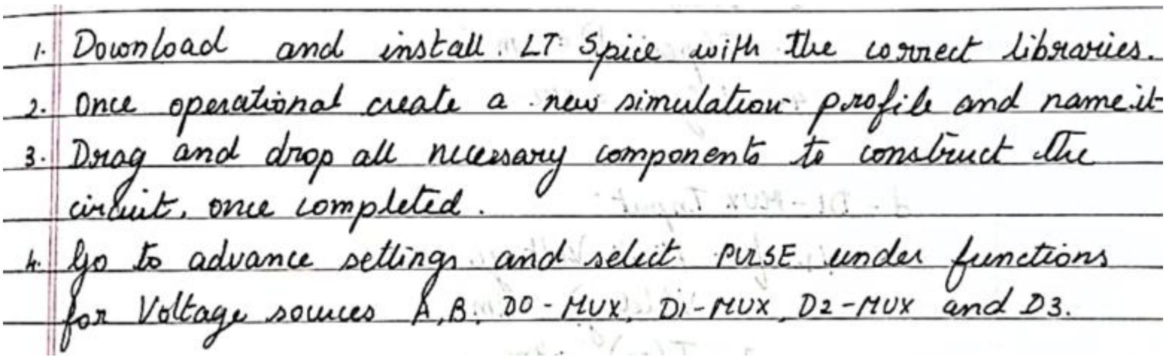
PRE-REQUISITES



PRE-REQUISITES

- Know how logic gates, combination circuits work and how the output varies depending on the input from the voltage source
- How to construct and operate decoders, multiplexers and encoders
- How to use LT spice and install the necessary libraries.

PROCEDURE



1. Download and install LT Spice with the correct libraries.
2. Once operational create a new simulation profile and name it
3. Drag and drop all necessary components to construct the circuit, once completed.
4. Go to advance settings and select PULSE under functions for Voltage sources A,B, D0-MUX, D1-MUX, D2-MUX and D3.

I. V(initial) = 0

II. V(on) = 5

III. T(rise) = 1n

IV. T(fall) = 1n

V. And depending on name of inputs.

a. A Input.

i. V(delay) = 1m

ii. T(on) = 1m

iii. T(period) = 2m.

b. B Input

i. V(delay) = 1m

ii. T(on) = 2m

iii. T(period) = 4m

c. DO-MUX Input

i. For First Voltage Source.

1. V(delay) = 1m
2. T(on) = 2m
3. T(period) = 4m
4. N cycles = 1.

(ii) for second Voltage source

1. V(delay) = 4m
2. T(on) = 1m
3. T(period) = 2m
4. N cycles = NIL

d. D1-MUX Input.

(i). foor First Voltage souce

1. V(delay) = 1m
2. T(on) = 3m
3. T(period) = 6m
4. N Cycles = 1

ii. For second Voltage souce.

1. V(delay) = 6m
2. T(on) = 1m
3. T(period) = 2m
4. NCycles = NIL.

e. D2-MUX Input.

i. For First Voltage source.

1. V(delay) = 0m
2. T(on) = 3m
3. T(period) = 6m
4. N Cycles = 1.

ii. For second Voltage souce.

1. V(delay) = 4m
2. T(on) = 2m
3. T(period) = 4m
4. N cycles = 1.

f. D3 · MUX Input ·

i. For First Vottage souce.

1. V(delay) = 4m
2. T(on) = 2m
3. T(period) = 4m
4. N Cycles = 1.

ii. For second Voltage source

1. V(delay) = 7m
2. T(on) = 1m
3. T(period) = 2m
4. N Cycles = 1.

g. 1-4 DEMUX input.

i. For First Voltage source

1. V(delay) = 0m
2. T(on) = 2m
3. T(period) = 4m
4. N Cycles = 1.

ii. For second Voltage source.

1. V(delay) = 4m
2. T(on) = 1m
3. T(period) = 2m
4. N Cycles = 1

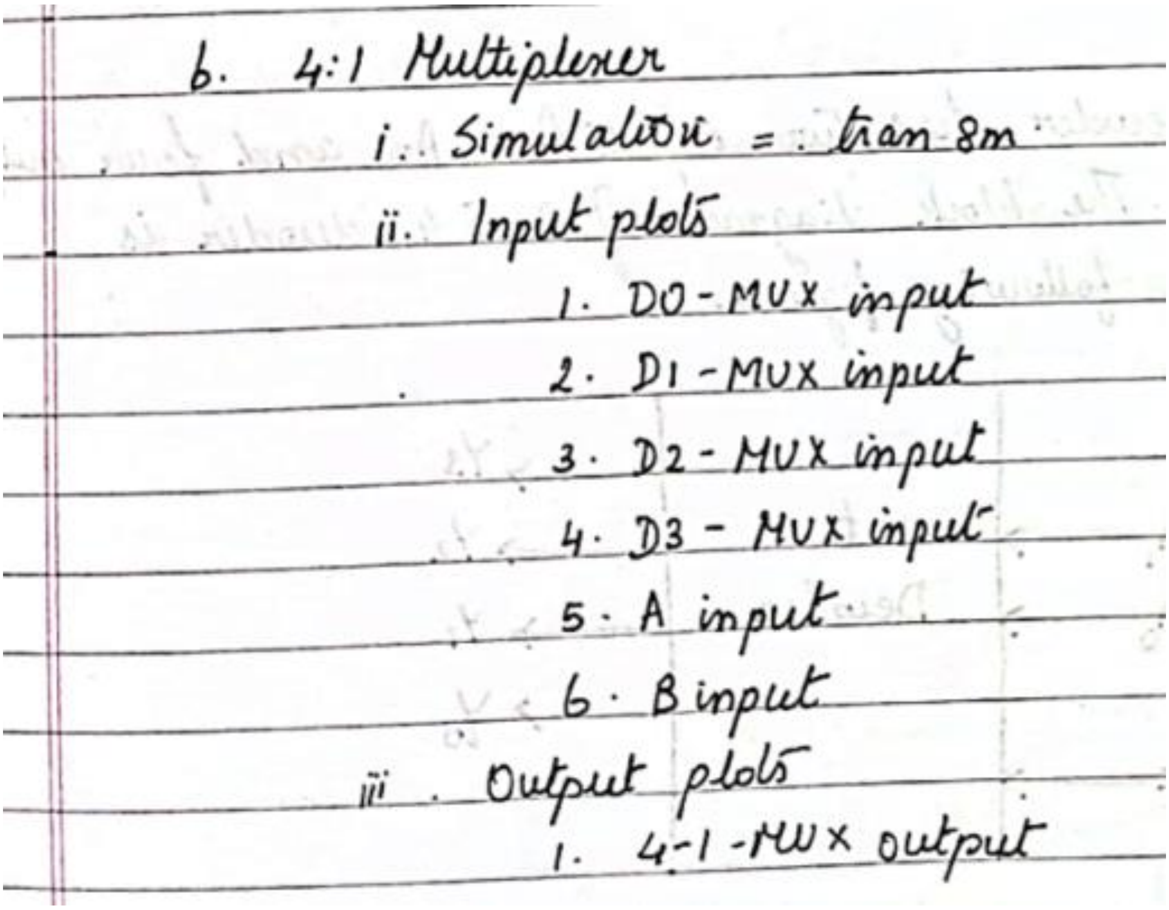
iii. For third Voltage souce.

1. V(delay) = 6m
2. T(on) = 2m
3. T(period) = 4m
4. N Cycles = 1

vi. Connect all of them accordingly using the appropriat labels and make sure to ground all negative ends of the voltage sources.

5. For Voltage sources x0 to x7
   a. V(initial) = 0.
   b. V(on) = 5
   c. T(rise) = 1n
   d. T(fall) = 1n
   e. T(on) = 1m
   f. T(period) = 2m
   g. N cycles = 1
   h. And depending on the name vary the T(delay)
      i. X0 - T(delay) = 0m
      ii. X1 - T(delay) = 1m
      iii X2 - T(delay) = 2m
      iv. X3 - T(delay) = 3m
      v. X4 - T(delay) = 4m
      vi. X5 - T(delay) = 5m.
      vii X6 - T(delay) = 6m
      viii X7 - T(delay) = 7m.

6. Once done run the simulation under trans for the respective amounts of time and plot the output waveforms in three seperate plots.
   a. 2x4 Decoder
      i. Simulation = .tran 8m
      ii Input plots
         1. 1-4-DEMUX input
         2. A input
         3. B input.
      iii. Output plots
         1. D0-DEMUX output
         2. D1-DEMUX output
         3. D2-DEMUX output
         4. D3-DEMUX output



b. 4:1 Multiplexer

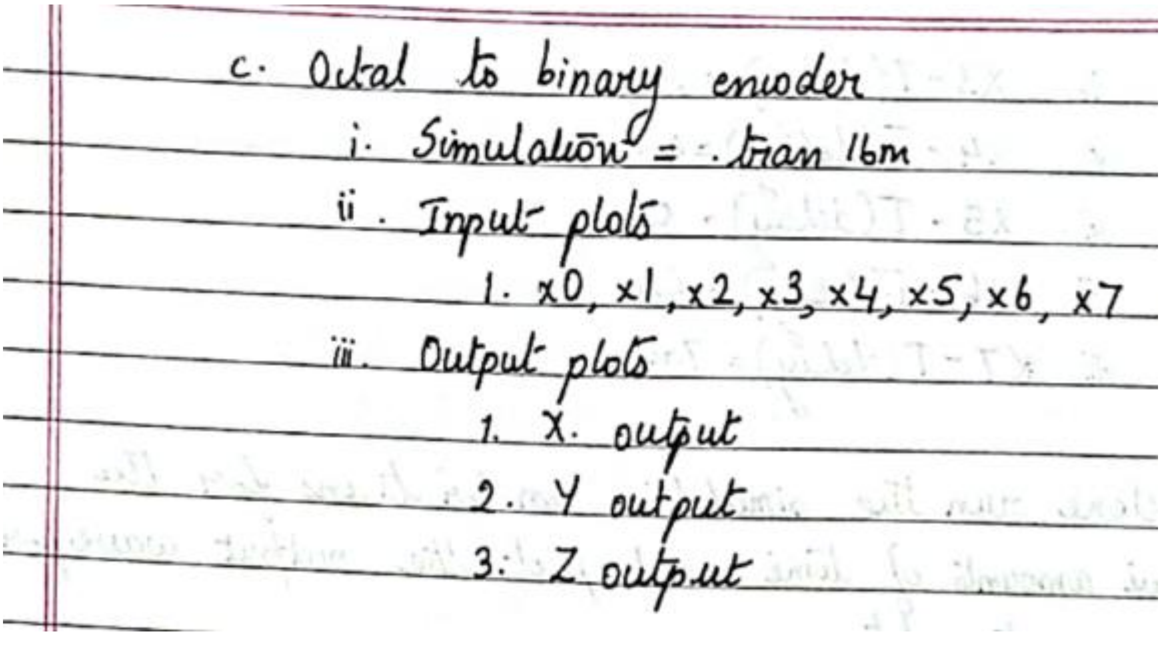
i. Simulation = .tran 8m

ii. Input plots

1. D0-MUX input
2. D1-MUX input
3. D2-MUX input
4. D3-MUX input
5. A input
6. B input

iii. Output plots

1. 4-1-MUX output



c. Octal to binary encoder

i. Simulation = .tran 16m

ii. Input plots

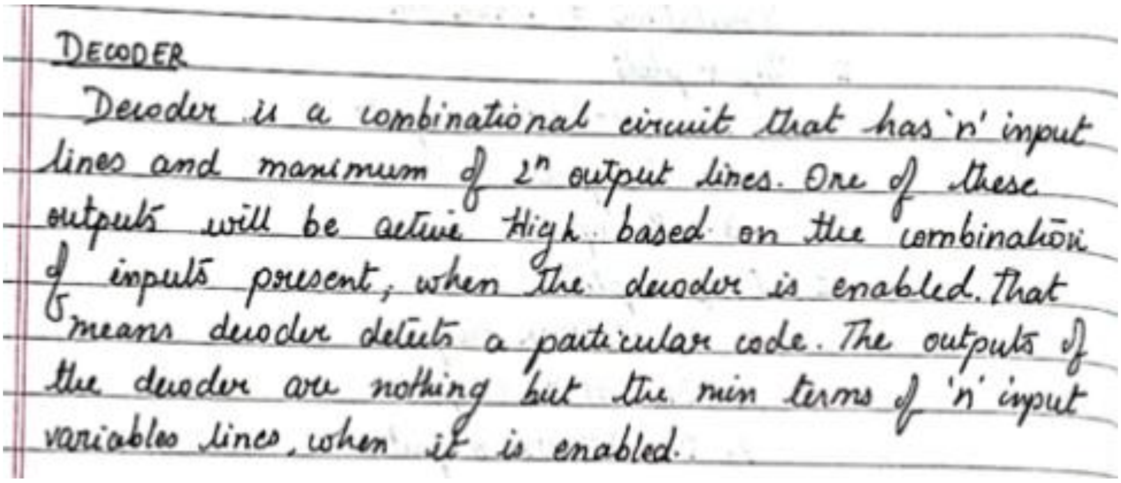
1. x0, x1, x2, x3, x4, x5, x6, x7

iii. Output plots

1. X. output
2. Y output
3. Z output

THEORY

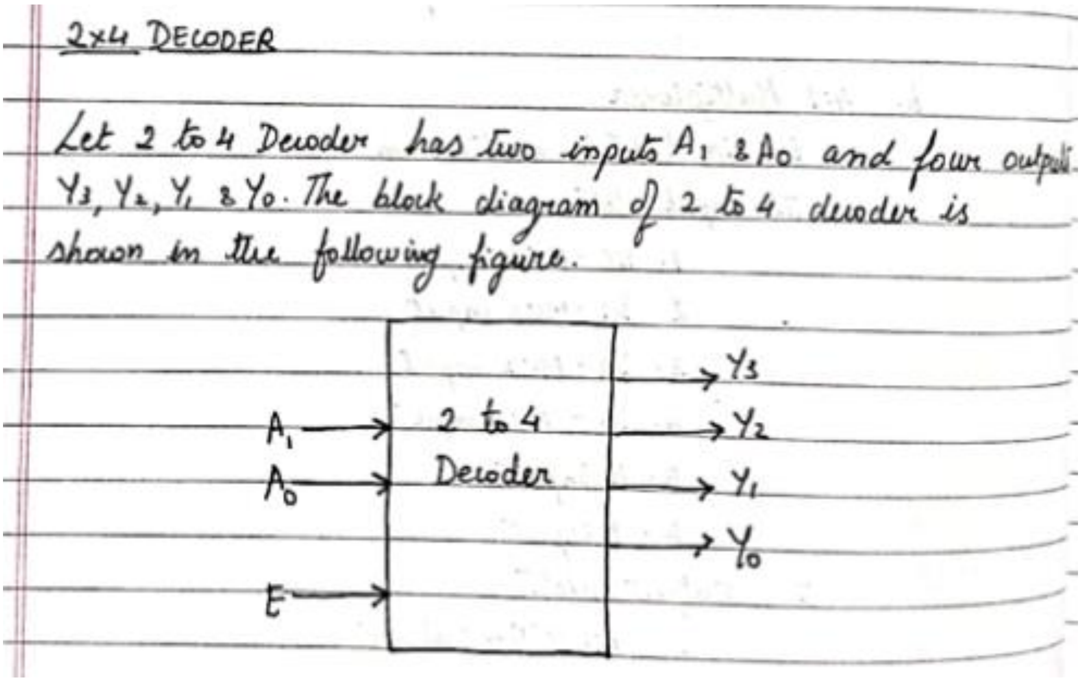
DECODER



DECODER

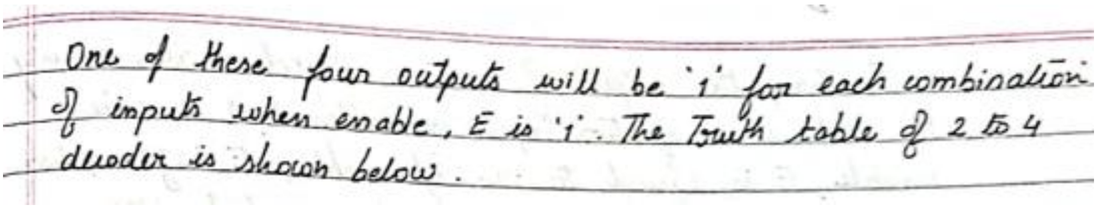
Decoder is a combinational circuit that has 'n' input lines and maximum of 2^n output lines. One of these outputs will be active High based on the combination of inputs present, when the decoder is enabled. That means decoder detects a particular code. The outputs of the decoder are nothing but the min terms of 'n' input variables lines, when it is enabled.

2X4 DECODER



2x4 DECODER

Let 2 to 4 Decoder has two inputs A_1 & A_0 and four outputs Y_3, Y_2, Y_1 & Y_0. The block diagram of 2 to 4 decoder is shown in the following figure.



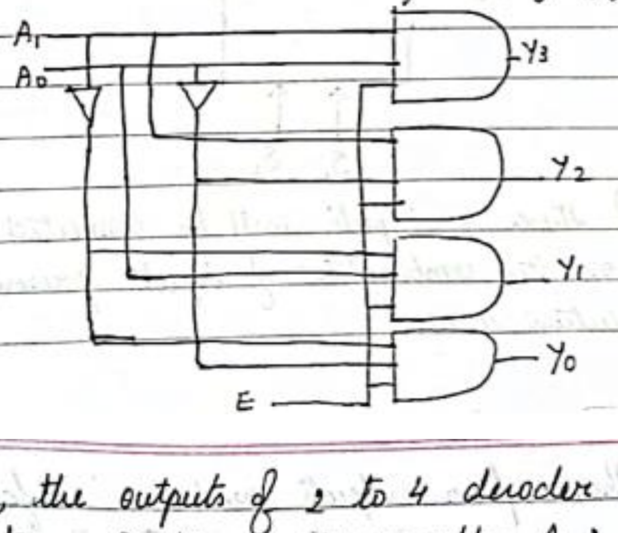
One of these four outputs will be '1' for each combination of inputs when enable, E is '1'. The Truth table of 2 to 4 decoder is shown below.

Enable	Inputs		Outputs			
E	A_1	A_0	Y_3	Y_2	Y_1	Y_0
0	x	x	0	0	0	0
1	0	0	0	0	0	1
1	0	1	0	0	1	0
1	1	0	0	1	0	0
1	1	1	1	0	0	0

From Truth table, we can write the Boolean functions for each output as.

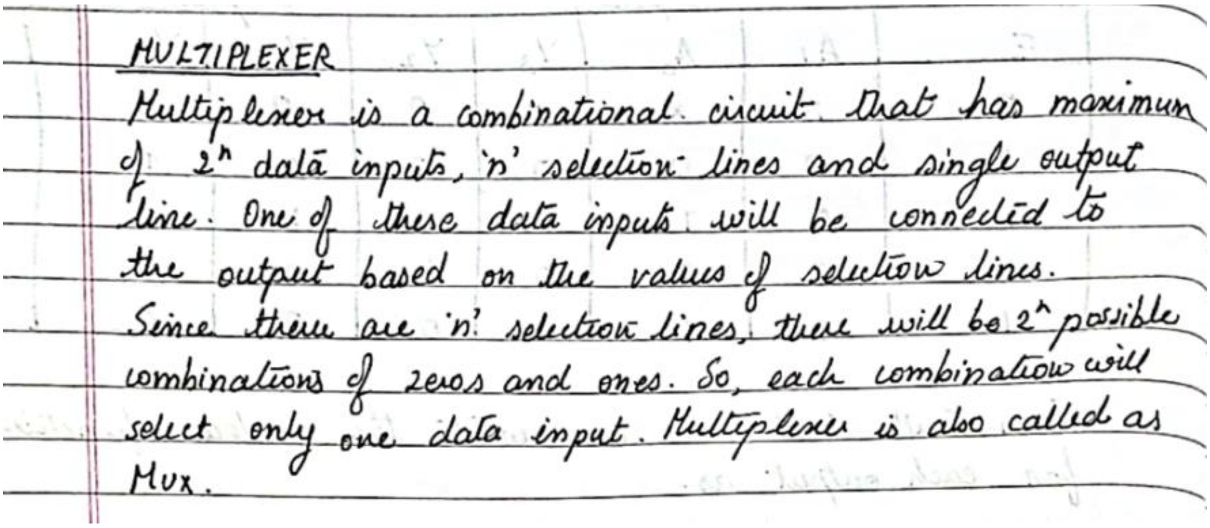
$$Y_3 = E \cdot A_1 \cdot A_0$$
$$Y_2 = E \cdot A_1 \cdot A_0'$$
$$Y_1 = E \cdot A_1' \cdot A_0$$
$$Y_0 = E \cdot A_1' \cdot A_0'$$

Each output is having one product term. So, there are four product terms in total. We can implement these four product terms by using four AND gates having three inputs each and Two inverters. The circuit diagram of 2 to 4 decoder is shown in the following figure.



Therefore, the outputs of 2 to 4 decoder are nothing but the min terms of two input variables A_1 & A_0 when enable, E is equal to one. If enable, E iszero, then all the outputs of decoder will be equal to zero.

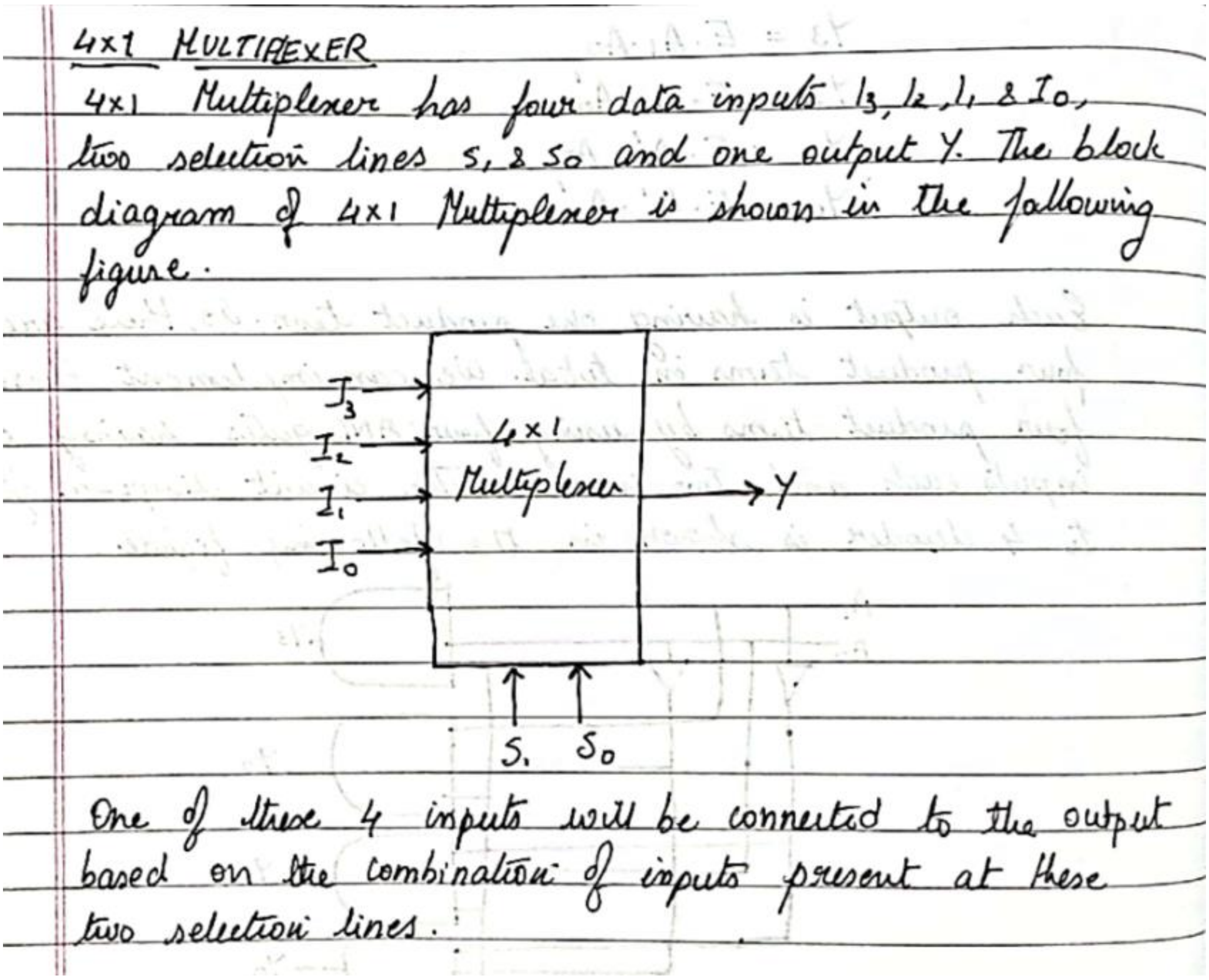
MULTIPLEXER



MULTIPLEXER

Multiplexer is a combinational circuit that has maximum of 2^n data inputs, 'n' selection lines and single output line. One of these data inputs will be connected to the output based on the values of selection lines. Since there are 'n' selection lines, there will be 2^n possible combinations of zeros and ones. So, each combination will select only one data input. Multiplexer is also called as Mux.

4X1 MULTIPLEXER



4x1 MULTIPLEXER

4x1 Multiplexer has four data inputs I_3, I_2, I_1 & I_0, two selection lines S_1 & S_0 and one output Y. The block diagram of 4x1 Multiplexer is shown in the following figure.

I_3
I_2
I_1
I_0
4x1
Multiplexer
Y
S_1
S_0

One of these 4 inputs will be connected to the output based on the combination of inputs present at these two selection lines.

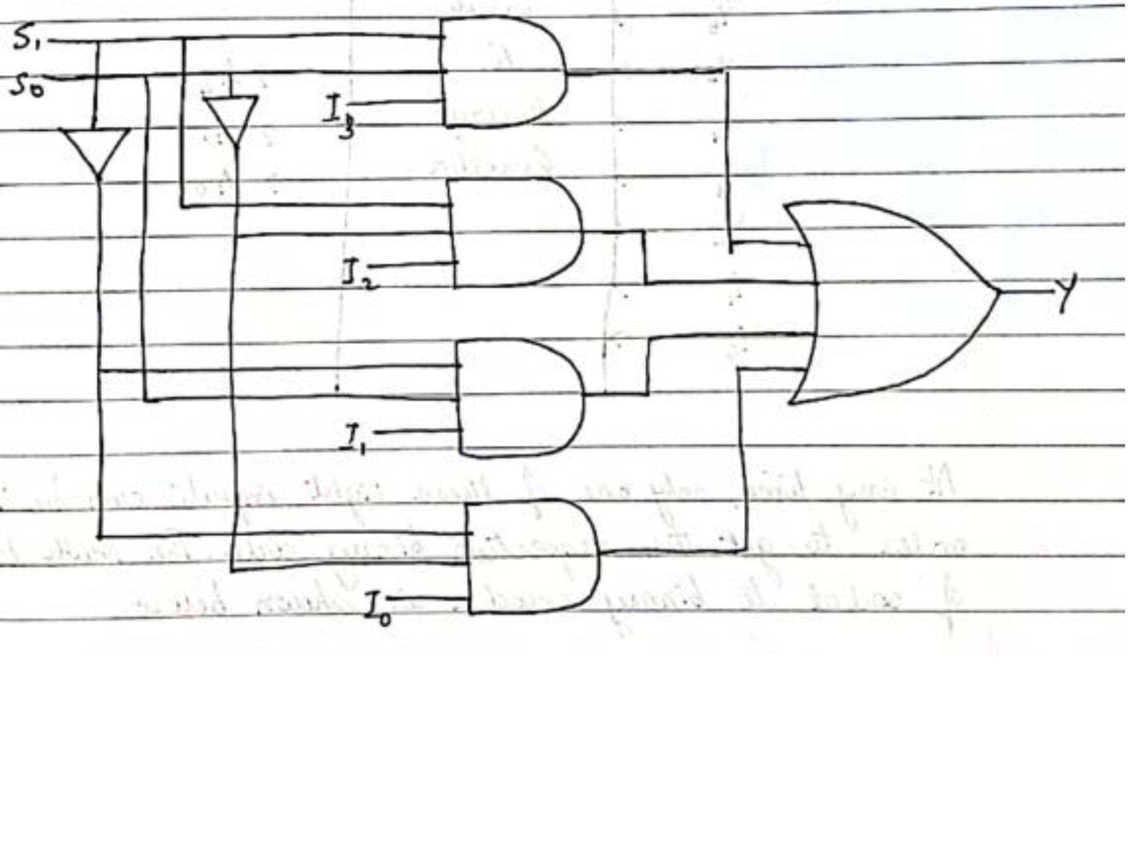
Truth table of 4x1 Multiplexer is shown below.

Selection Lines		Output
S_1	S_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

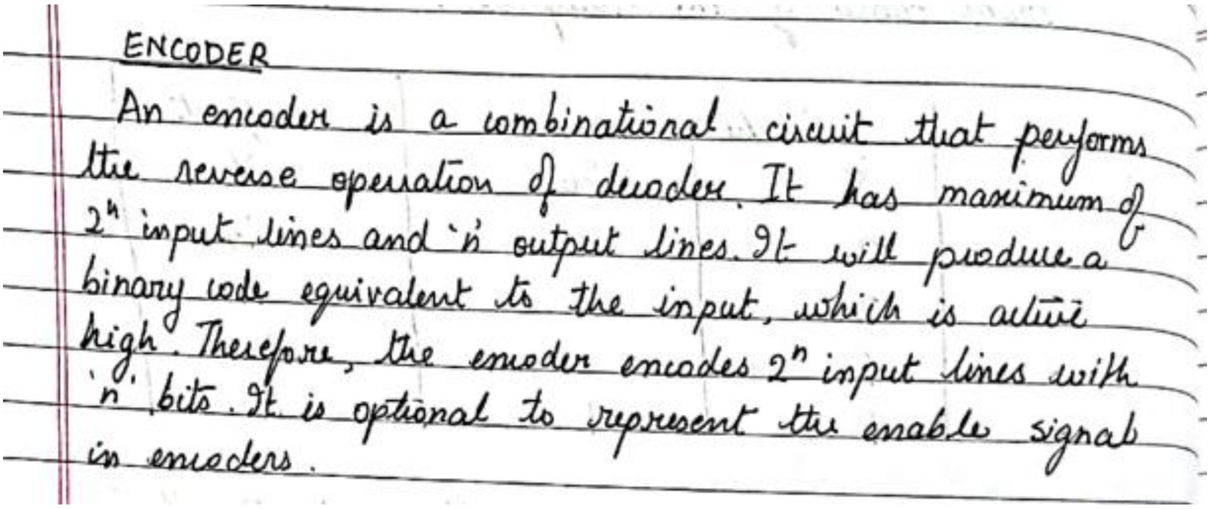
From Truth table, we can directly write the Boolean function for output, Y as.

$$Y = S_1'S_0'I_0 + S_1'S_0I_1 + S_1S_0'I_2 + S_1S_0I_3$$

We can implement this Boolean function using invertors. AND gates & OR gate. The circuit diagram of 4x1 multiplexer is shown in the following figure



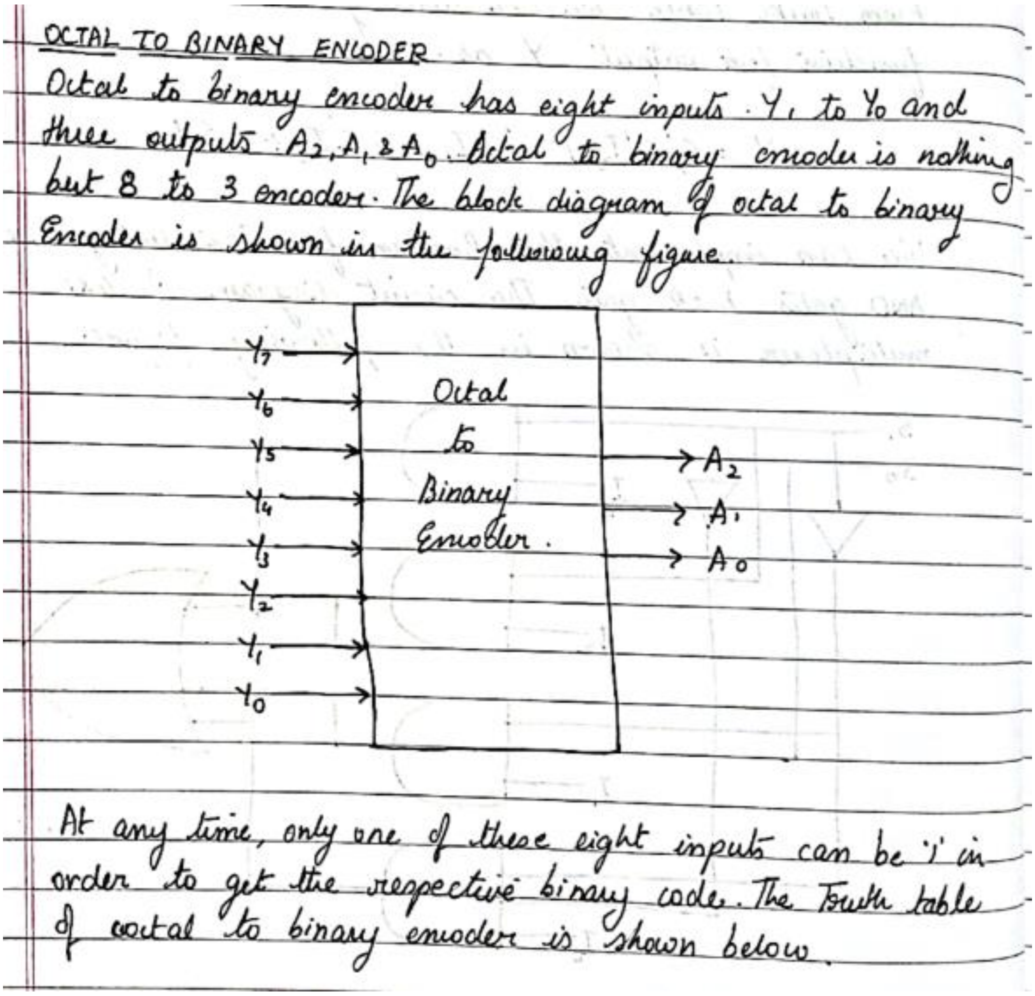
ENCODER



ENCODER

An encoder is a combinational circuit that performs the reverse operation of decoder. It has maximum of 2^n input lines and 'n' output lines. It will produce a binary code equivalent to the input, which is active high. Therefore, the encoder encodes 2^n input lines with 'n' bits. It is optional to represent the enable signal in encoders.

OCTAL TO BINARY ENCODER



OCTAL TO BINARY ENCODER

Octal to binary encoder has eight inputs. Y_1 to Y_0 and three outputs A_2, A_1 & A_0. Octal to binary encoder is nothing but 8 to 3 encoder. The block diagram of octal to binary Encoder is shown in the following figure.

Y_7
Y_6
Y_5
Y_4
Y_3
Y_2
Y_1
Y_0
Octal to Binary Encoder.
A_2
A_1
A_0

At any time, only one of these eight inputs can be '1' in order to get the respective binary code. The Truth table of octal to binary encoder is shown below.

Inputs								Outputs		
Y_7	Y_6	Y_5	Y_4	Y_3	Y_2	Y_1	Y_0	A_2	A_1	A_0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	1
0	1	0	0	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	1	1

From Truth table, we can write the Boolean functions for each output as.

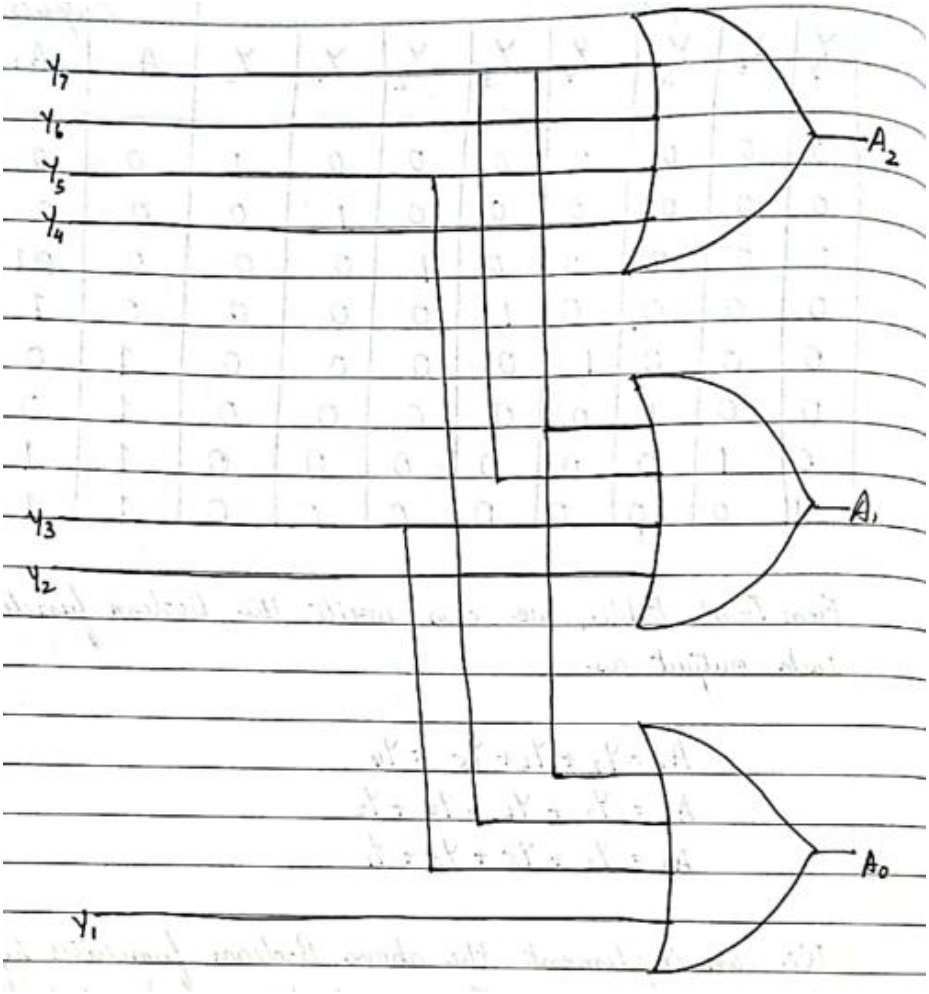
$$A_2 = Y_7 + Y_6 + Y_5 + Y_4$$

$$A_1 = Y_7 + Y_6 + Y_3 + Y_2$$

$$A_0 = Y_7 + Y_5 + Y_3 + Y_1$$

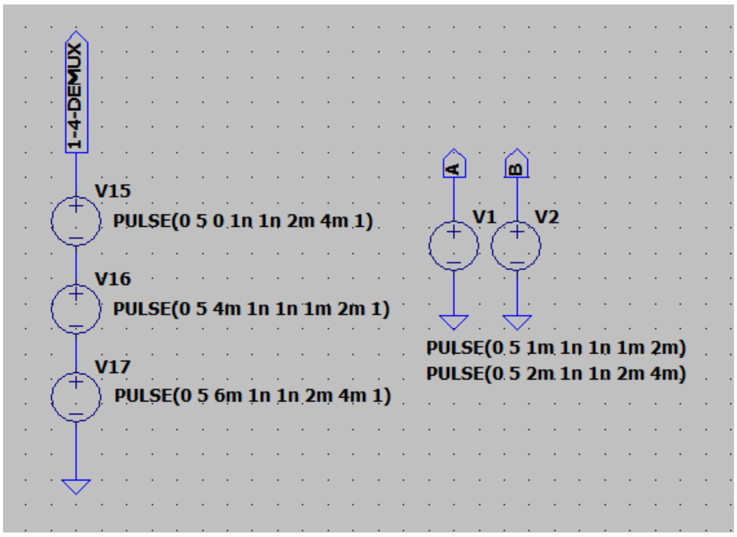
We can implement the above Boolean functions by using four input OR gates. The circuit diagram of octal to binary encoder is shown in the following figure.

The above circuit diagram contains three 4-input OR gates. These OR gates encode the eight inputs with three bits.



SCHEMATIC DIAGRAM

2X4 DECODER



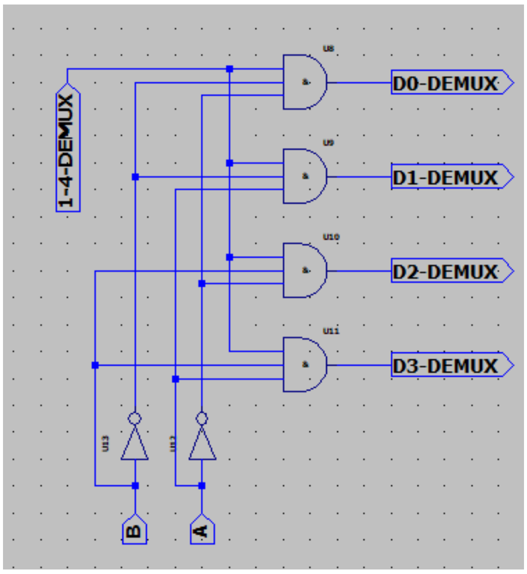
V15
PULSE(0 5 0 1n 1n 2m 4m 1)

V16
PULSE(0 5 4m 1n 1n 1m 2m 1)

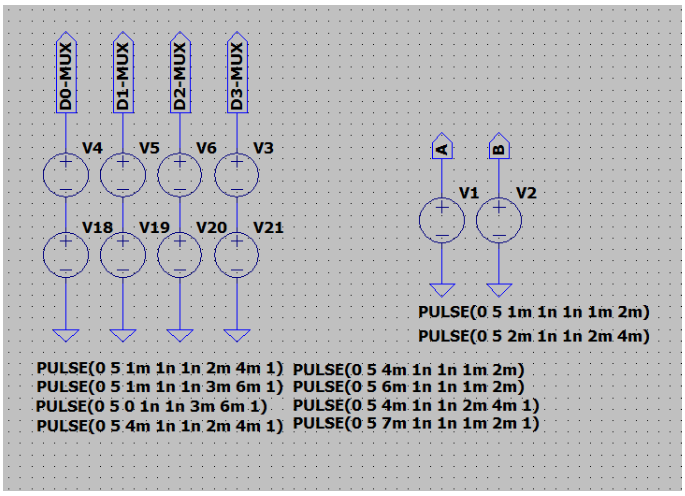
V17
PULSE(0 5 6m 1n 1n 2m 4m 1)

V1
PULSE(0 5 1m 1n 1n 1m 2m)

V2
PULSE(0 5 2m 1n 1n 2m 4m)

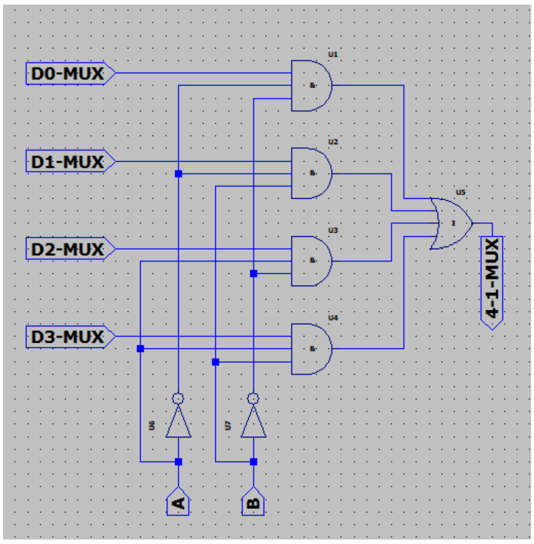


4:1 MULTIPLEXER

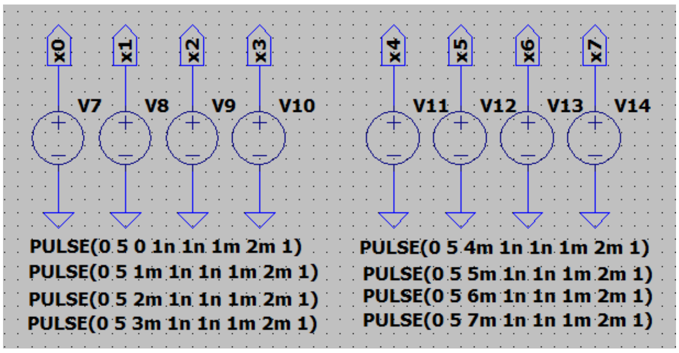


```
PULSE(0 5 1m 1n 1n 1m 2m)
PULSE(0 5 2m 1n 1n 2m 4m)
```

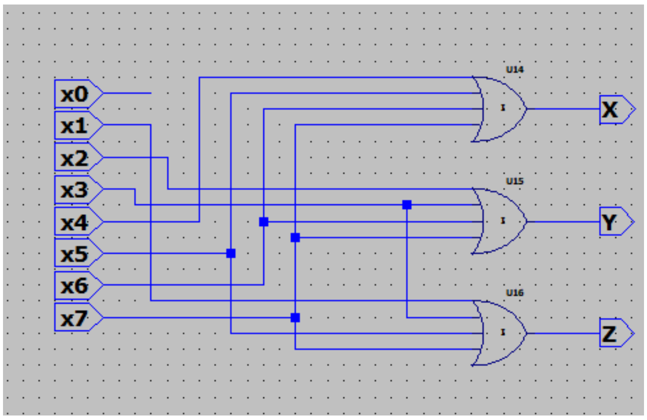
```
PULSE(0 5 1m 1n 1n 2m 4m 1)  PULSE(0 5 4m 1n 1n 1m 2m)
PULSE(0 5 1m 1n 1n 3m 6m 1)  PULSE(0 5 6m 1n 1n 1m 2m)
PULSE(0 5 0 1n 1n 3m 6m 1)   PULSE(0 5 4m 1n 1n 2m 4m 1)
PULSE(0 5 4m 1n 1n 2m 4m 1)  PULSE(0 5 7m 1n 1n 1m 2m 1)
```



OCTAL TO DECIMAL ENCODER

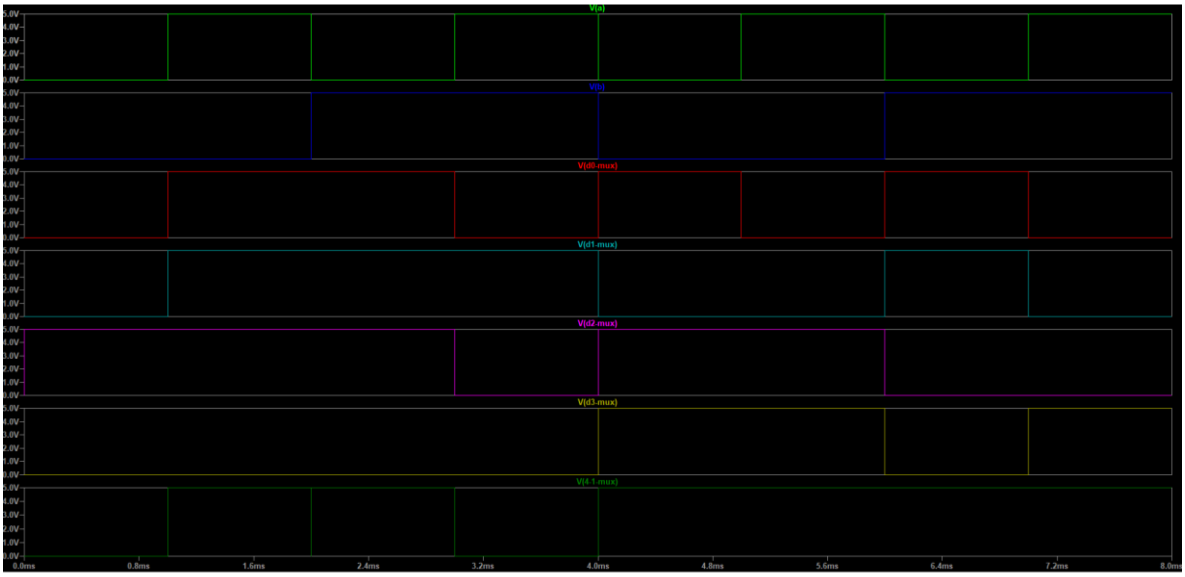


PULSE(0 5 0 1n 1n 1m 2m 1)
PULSE(0 5 1m 1n 1n 1m 2m 1)
PULSE(0 5 2m 1n 1n 1m 2m 1)
PULSE(0 5 3m 1n 1n 1m 2m 1)
PULSE(0 5 4m 1n 1n 1m 2m 1)
PULSE(0 5 5m 1n 1n 1m 2m 1)
PULSE(0 5 6m 1n 1n 1m 2m 1)
PULSE(0 5 7m 1n 1n 1m 2m 1)

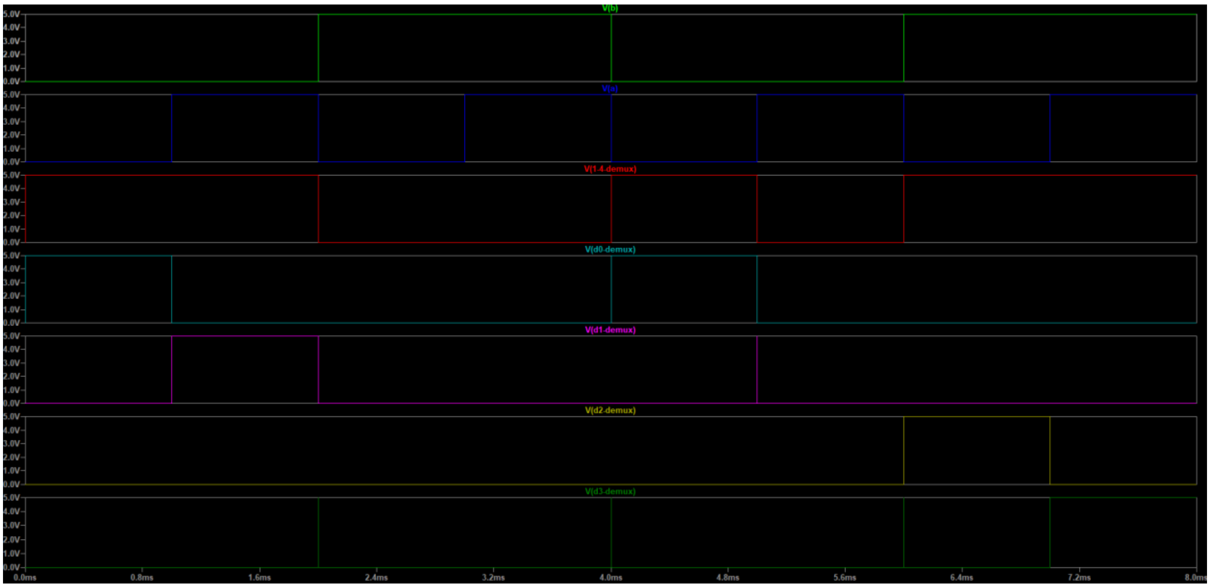


OUTPUT WAVEFORMS

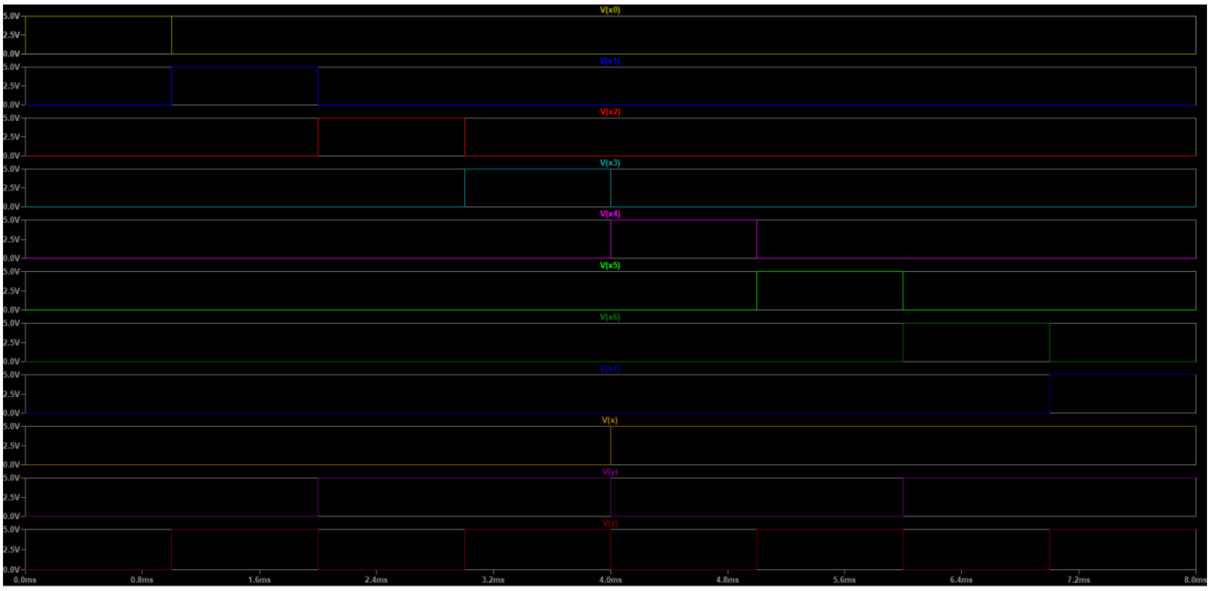
2X4 DECODER



4:1 MULTIPLEXER



OCTAL TO DECIMAL ENCODER



RESULTS AND INFERENCE:

The results are verified with the help of truth tables and LT SPICE results and hence the 2x4 Decoder, 4:1 Multiplexer and Octal to Decimal Encoder are verified.