International Journal of Electrical and Electronics Engineering Studies, 9 (2), 61-74, 2023

ISSN 2056-581X (Print),

ISSN 2056-5828(Online)

Website: https://www.eajournals.org/

Publication of the European Centre for Research Training and Development -UK

5-Line-to-32 Line-Decoder/Demultiplexer using IC 74AC11138 i.e. 3:8 Decoder/Demultiplexer

Nikhil
Electronics and Communication
National Institute of Technology
Jamshedpur,INDIA

doi: https://doi.org/10.37745/ijeees.13/vol9n26174 Published December 07, 2023

Citation: Nikhil (2023) 5-Line-to-32 Line-Decoder/Demultiplexer using IC 74AC11138 i.e. 3:8Decoder/Demultiplexer, *International Journal of Electrical and Electronics Engineering Studies*, 9 (2), 61-74

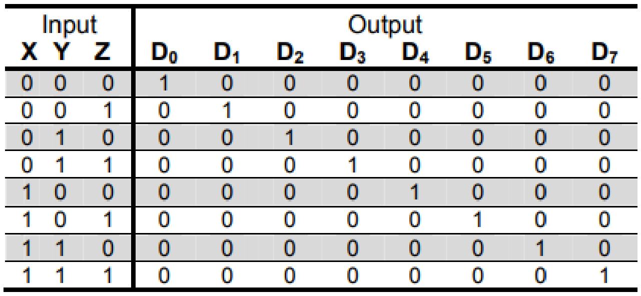
ABSTRACT: *The circuit is designed to be used in high performance memory-decoding or data-routing applications requiring very short propagation delay times. In highperformance memory systems, this decoder can be used to minimize the effects of system decoding. When employed with high-speed memories utilizing a fast enable circuit, the delay times of this decoder and the enable time of the memory are usually less than the typical access time of the memory. This means that the effective system delay introduced by the decoder is negligible.*

KEYWORDS*:* decoder, CMOS inverter, practical, IC 74AC11138, 4:16 bit decoder design

INTRODUCTION

Decoder Discrete quantities of information are represented in digital system with binary codes. A binary code of n bits is capable of representing up to distinct elements of the coded information. A decoder is a combinational circuit that converts binary information from n input lines to a maximum of 2^n unique output lines. If the n-bit decoded information has unused or don’t-care combinations, the decoder output will have less then 2^n outputs [2].In Digital Electronics, discrete quantities of information are represented by binary codes. A binary code of n bits is capable of representing up to 2^n distinct elements of coded information. The name “Decoder” means to translate or decode coded information from one format into another, so a digital decoder transforms a set of digital input signals into an equivalent decimal code at its output. A decoder is a combinational circuit that converts binary information from n input lines to a maximum of 2^n unique output lines. Binary Decoder – Binary Decoders are another type of digital logic device that has inputs of 2-bit, 3-bit or 4-bit codes depending upon the number of data input lines, so a decoder that has a set of two or more bits will be defined as having an n-bit code, and therefore it will be possible to represent 2^n possible values. If a binary decoder receives n inputs it activates one and only one of its 2^n outputs based on that input with all other outputs deactivated. If the n -bit coded information has unused combinations, the decoder may have fewer than 2^n outputs. Example, an inverter (NOT-gate) can be classified as a 1-to-2 binary decoder as 1-input and 2-outputs is possible. i.e an input A can give either A or A complement as the output. Then we can say that a standard combinational logic decoder is an n-to-m decoder, where m <= 2^n, and whose output, Q is dependent only on its present input states. Their purpose is to generate the 2^n (or fewer) minterms of n input variables. Each combination of inputs will assert a unique output. A Binary Decoder converts coded inputs into coded outputs, where the input and output codes are different and decoders are available to "decode" either a Binary or BCD (8421 code) input pattern to typically a Decimal output code. Practical "binary decoder" circuits include 2-to-4, 3-to-8 and 4-to-16 line configurations.

TRUTH TABLE OF 3:8 DECODER



Input			Output							
X	Y	Z	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

Table 1. Truth table of 3-to-8 line decoder.

The combinational circuit that change the binary information into 2^N output lines is known as Decoders. The binary information is passed in the form of N input lines. The output lines define the 2N-bit code for the binary information. In simple words, the Decoder performs the reverse operation of the Encoder. At a time, only one input line is activated for simplicity. The produced 2^N-bit output code is equivalent to the binary information.

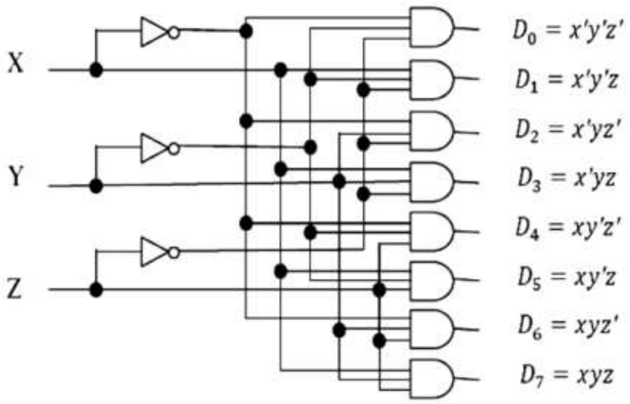
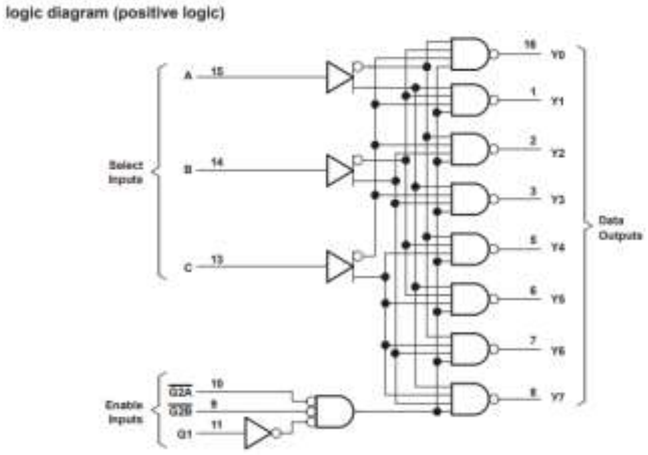


Figure 1. 3-to-8 line decoder.

3 to 8 line decoder: The 3 to 8 line decoder is also known as Binary to Octal Decoder. In a 3 to 8 line decoder, there is a total of eight outputs, i.e., Y0, Y1, Y2, Y3, Y4, Y5, Y6, and Y7 and three outputs, i.e., A0, A1, and A2. This circuit has an enable input 'E'. Just like 2 to 4 line decoder, when enable 'E' is set to 1, one of these four outputs will be 1. The block diagram and the truth table of the 3 to 8 line encoder are given below.



A DIAGRAM FOR 32 BIT DECODING SCHEME

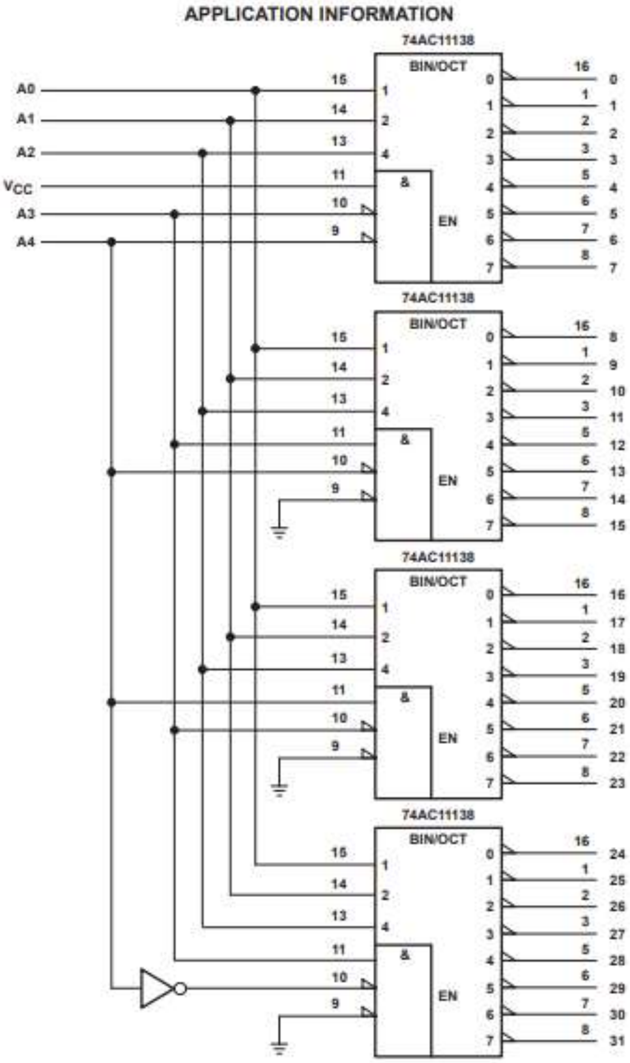
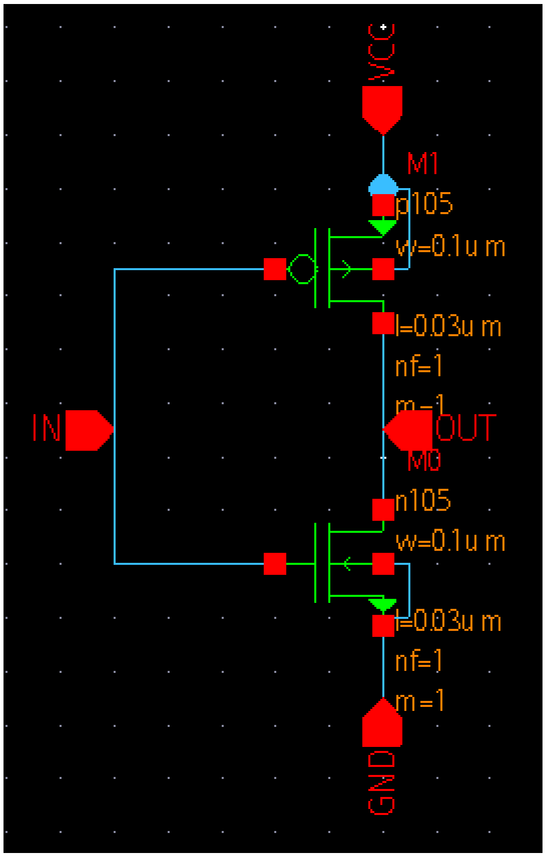
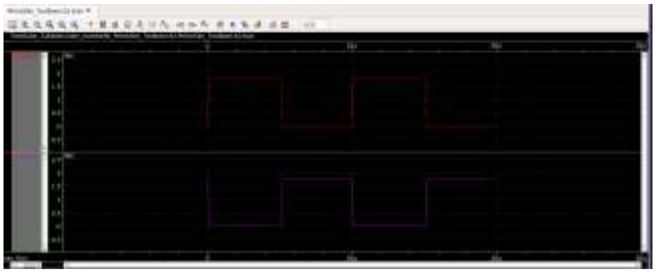


Figure 3. 32-Bit Decoding Scheme

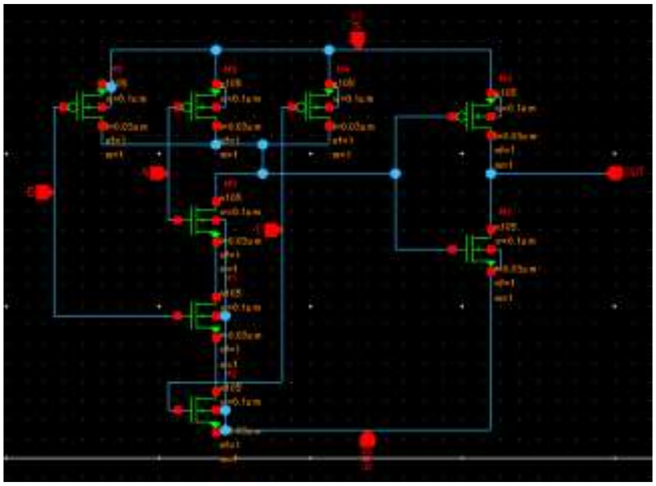
CIRCUITS AND WAVEFORMS

Inveter

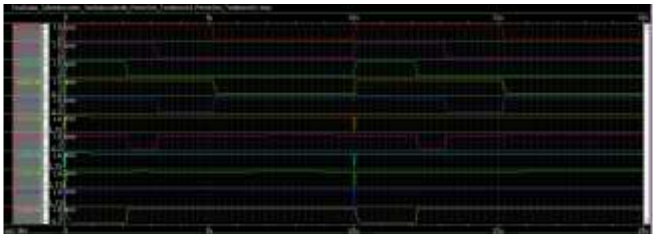




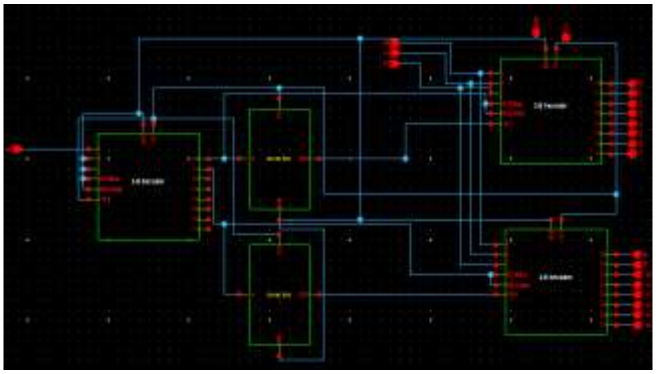
3 Input AND



3-8 decoder waveform (NOTE : Active LOW)

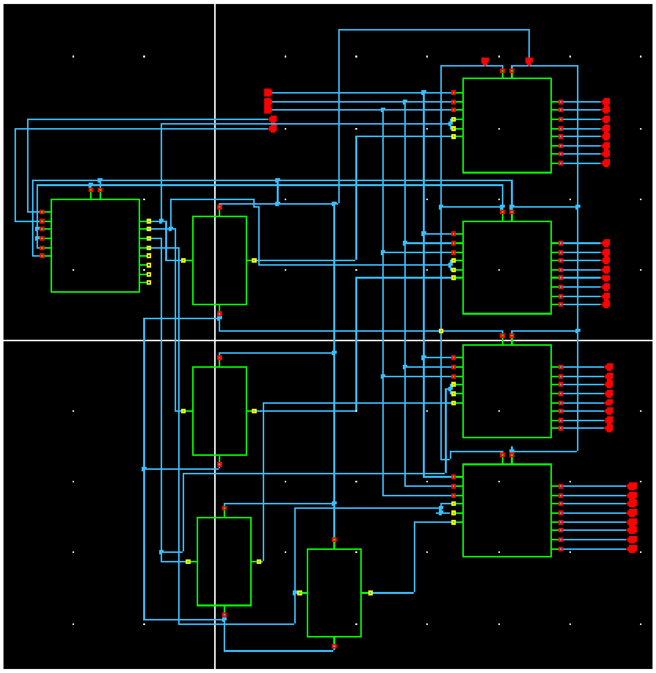


4-16 bit decoder using 3-8 Decoder

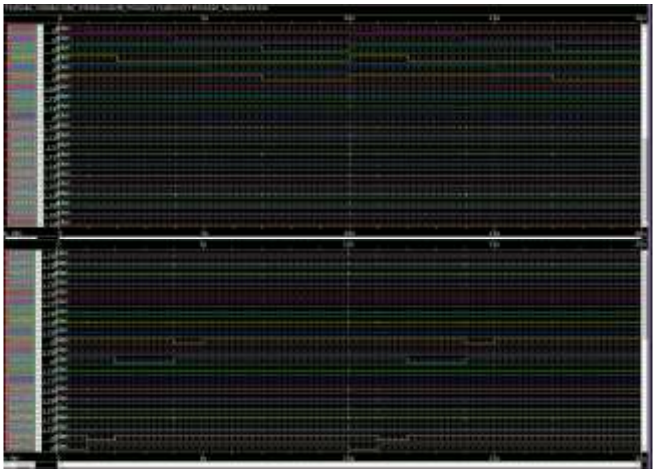


Simulation Implementation

In this Circuit diagram 5 x 3:8 decoder and 4 inverters are used



Simulation/Testbench Circuit



NETLIST

```
*  Generated for: PrimeSim
*  Design library name: 32bitdecoder
*  Design cell name: 32bitdecodertb
*  Design view name: schematic
.lib '/PDK/SAED_PDK32nm/hspice/saed32nm.lib' TT

*Custom Compiler Version S-2021.09
*Sat Feb 26 14:49:02 2022

.global gnd!
************************************************************************************************
* Library          : 32bitdecoder
* Cell             : 4inputnand
* View             : schematic
* View Search List : hspice hspiceD schematic spice veriloga
* View Stop List   : hspice hspiceD
************************************************************************************************
.subckt _4inputnand a b c d gnd_1 vcc y
xm3 net13 d gnd_1 gnd_1 n105 w=0.1u l=0.03u nf=1 m=1
xm2 net9 c net13 gnd_1 n105 w=0.1u l=0.03u nf=1 m=1
xm1 net5 a net9 gnd_1 n105 w=0.1u l=0.03u nf=1 m=1
xm0 y b net5 gnd_1 n105 w=0.1u l=0.03u nf=1 m=1
xm7 y d vcc vcc p105 w=0.1u l=0.03u nf=1 m=1
xm6 y c vcc vcc p105 w=0.1u l=0.03u nf=1 m=1
xm5 y b vcc vcc p105 w=0.1u l=0.03u nf=1 m=1
xm4 y a vcc vcc p105 w=0.1u l=0.03u nf=1 m=1
.ends _4inputnand

************************************************************************************************
* Library          : 32bitdecoder
* Cell             : inverter
* View             : schematic
* View Search List : hspice hspiceD schematic spice veriloga
* View Stop List   : hspice hspiceD
************************************************************************************************
```

```
.subckt inverter gnd_1 in out vcc
xm0 out in gnd_1 gnd_1 n105 w=0.1u l=0.03u nf=1 m=1
xm1 out in vcc vcc p105 w=0.1u l=0.03u nf=1 m=1
.ends inverter

********************************************************************************
* Library         : 32bitdecoder
* Cell            : 3inputand
* View            : schematic
* View Search List : hspice hspiceD schematic spice veriloga
* View Stop List   : hspice hspiceD
********************************************************************************
.subckt _3inputand a b c gnd_1 out vcc
xm8 out net43 gnd_1 gnd_1 n105 w=0.1u l=0.03u nf=1 m=1
xm2 net9 c gnd_1 gnd_1 n105 w=0.1u l=0.03u nf=1 m=1
xm1 net5 b net9 gnd_1 n105 w=0.1u l=0.03u nf=1 m=1
xm0 net43 a net5 gnd_1 n105 w=0.1u l=0.03u nf=1 m=1
xm9 out net43 vcc vcc p105 w=0.1u l=0.03u nf=1 m=1
xm5 net43 b vcc vcc p105 w=0.1u l=0.03u nf=1 m=1
xm4 net43 c vcc vcc p105 w=0.1u l=0.03u nf=1 m=1
xm3 net43 a vcc vcc p105 w=0.1u l=0.03u nf=1 m=1
.ends _3inputand

********************************************************************************
* Library         : 32bitdecoder
* Cell            : 3to8decoder
* View            : schematic
* View Search List : hspice hspiceD schematic spice veriloga
* View Stop List   : hspice hspiceD
********************************************************************************
.subckt _3to8decoder 9g2bbar 10g2abar 11g1 a b c gnd_1 vcc y0 y1 y2 y3 y4 y5 y6
+ y7
xi19 c b a net66 gnd_1 vcc y7 _4inputnand
xi6 c b net50 net66 gnd_1 vcc y6 _4inputnand
xi5 c net82 a net66 gnd_1 vcc y5 _4inputnand
xi4 c net82 net50 net66 gnd_1 vcc y4 _4inputnand
```

```
xi3 net58 b a net66 gnd_1 vcc y3 _4inputnand
xi2 b net50 net58 net66 gnd_1 vcc y2 _4inputnand
xi1 net58 a net82 net66 gnd_1 vcc y1 _4inputnand
xi0 net58 net82 net50 net66 gnd_1 vcc y0 _4inputnand
xi12 gnd_1 9g2bbar net72 vcc inverter
xi10 gnd_1 10g2abar net69 vcc inverter
xi9 gnd_1 c net58 vcc inverter
xi8 gnd_1 b net82 vcc inverter
xi7 gnd_1 a net50 vcc inverter
xi11 net69 net72 11g1 gnd_1 net66 vcc _3inputand
.ends _3to8decoder

********************************************************************************
* Library         : 32bitdecoder
* Cell            : 32bitdecoder
* View            : schematic
* View Search List : hspice hspiceD schematic spice veriloga
* View Stop List   : hspice hspiceD
********************************************************************************
.subckt _32bitdecoder 0_1 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21
+ 22 23 24 25 26 27 28 29 30 31 a0 a1 a2 a3 a4 gnd_1 vcc
xi9 gnd_1 gnd_1 vcc a3 a4 gnd_1 gnd_1 vcc net137 net139 net149 net157 net108
+ net109 net110 net111 _3to8decoder
xi3 net157 net151 net152 a0 a1 a2 gnd_1 vcc 24 25 26 27 28 29 30 31 _3to8decoder
xi2 net149 net149 net150 a0 a1 a2 gnd_1 vcc 16 17 18 19 20 21 22 23 _3to8decoder
xi1 net139 net139 net148 a0 a1 a2 gnd_1 vcc 8 9 10 11 12 13 14 15 _3to8decoder
xi0 net137 net137 net146 a0 a1 a2 gnd_1 vcc 0_1 1 2 3 4 5 6 7 _3to8decoder
xi13 gnd_1 net157 net152 vcc inverter
xi12 gnd_1 net149 net150 vcc inverter
xi11 gnd_1 net139 net148 vcc inverter
xi10 gnd_1 net137 net146 vcc inverter
.ends _32bitdecoder

********************************************************************************
* Library         : 32bitdecoder
* Cell            : 32bitdecodertb
* View            : schematic
* View Search List : hspice hspiceD schematic spice veriloga
* View Stop List   : hspice hspiceD
****************************************************************************************
xi0 y0 y1 y2 y3 y4 y5 y6 y7 y8 y9 y10 y11 y12 y13 y14 y15 y16 y17 y18 y19 y20
+ y21 y22 y23 y24 y25 y26 y27 y28 y29 y30 y31 a0 a1 a2 a3 a4 gnd! net40
+ _32bitdecoder
v1 net40 gnd! dc=1.8
v41 a3 gnd! dc=0 pulse ( 0 1.8 0 0.1p 0.1p 2u 10u )
v42 a2 gnd! dc=0 pulse ( 0 1.8 0 0.1p 0.1p 7u 10u )
v40 a4 gnd! dc=0 pulse ( 0 1.8 0 0.1p 0.1p 5u 10u )
v43 a1 gnd! dc=0 pulse ( 0 1.8 0 0.1p 0.1p 4u 10u )
v44 a0 gnd! dc=0 pulse ( 0 1.8 0 0.1p 0.1p 1u 10u )
c39 y31 gnd! c=1f
c38 y30 gnd! c=1f
c37 y29 gnd! c=1f
c36 y28 gnd! c=1f
c35 y27 gnd! c=1f
c34 y26 gnd! c=1f
c33 y25 gnd! c=1f
c32 y24 gnd! c=1f
c31 y23 gnd! c=1f
c30 y22 gnd! c=1f
c29 y21 gnd! c=1f
c28 y20 gnd! c=1f
c27 y19 gnd! c=1f
c26 y18 gnd! c=1f
c25 y17 gnd! c=1f
c24 y16 gnd! c=1f
c23 y15 gnd! c=1f
c22 y14 gnd! c=1f
c21 y13 gnd! c=1f
c20 y12 gnd! c=1f
c19 y11 gnd! c=1f
c18 y10 gnd! c=1f
c17 y9 gnd! c=1f
```

```
c16 y8 gnd! c=1f
c15 y7 gnd! c=1f
c14 y6 gnd! c=1f
c13 y5 gnd! c=1f
c12 y4 gnd! c=1f
c11 y3 gnd! c=1f
c10 y2 gnd! c=1f
c9 y1 gnd! c=1f
c8 y0 gnd! c=1f


.tran '1u' '20u' name=tran

.option primesim_remove_probe_prefix = 0
.probe v(*) i(*) level=1
.probe tran v(a0) v(a1) v(a2) v(a3) v(a4) v(y0) v(y1) v(y10) v(y11) v(y12)
+ v(y13) v(y14) v(y15) v(y16) v(y17) v(y18) v(y19) v(y2) v(y20) v(y21) v(y22)
+ v(y23) v(y24) v(y25) v(y26) v(y27) v(y28) v(y29) v(y3) v(y30) v(y31) v(y4)
+ v(y5) v(y6) v(y7) v(y8) v(y9)

.temp 25



.option primesim_output=wdf


.option parhier = LOCAL


.end
```

CONCLUSION

The 5:32 bit Decoder can be made successfully at Transistor level using Synopsys CAD Tool. A GitHub Post was successfully made::https://github.com/laptopcomputermistri/Nikhil

AUTHOR

The author name is NIKHIL. He is currently trainee Engineer in Qualcomm India Pvt. Ltd. For any query send eMail to laptopcomputermistri@gmail.com .

ACKNOWLEDGMENT

The author wants to acknowledge yourself for glancing the paper and the author is thankful to People of NIT Jamshedpur. The author is Thankful to IIT Hyderabad for organizing Cloud Based Analog IC Design Hackathon 2022 ,where the access to SYNOPSYS CAD Tool was provided through LINUX Servers. Thanks! to Kunal Ghosh, Co-founder, VSD Corp. Pvt. Ltd. .

REFERENCES.

[1] Texas Instrument 3:8 decoder/demultiplexer using IC 74AC11138

[2] M. M. Mano, Digital Design, Prentice Hall, 1984, pp. 167-175

[3] Design of a Qubit and a Decoder in Quantum Computing Based on a Spin Field Effect A. A. Suratgar1 , S. Rafiei*2 , A. A. Taherpour3 , A.Babaei4