International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 03 | March - 2024 SJIF Rating: 8.176 ISSN: 2582-3930

2:1 Multiplexer, 1:2 De-multiplexer,2:4 Decoder and 4:2 Encoder Circuit Design with CMOS Technology Implementing with Artificial Neural Network with Verilog HDL Code for Output

Rupam Sardar

Budge Budge Institute of Technology, Kolkata-700137

Abstract:

The goal of this project is to create a De-multiplexer gate using a complementary metal oxide semiconductor (CMOS) and an artificial neural network.When designing any COMS circuit, we always keep in mind that the lowest possible cost should be the aim.In this work, the circuit was designed using a multilayer artificial neural network. We utilize weights to alter the value and treat negative values as inverters and neurons as transistors in our work.We are also developing Verilog-HdL code to easily apply the De-multiplexer for experimenting with an artificial neural network and set weights to acquire the desired results.

The purpose of this project is to use an artificial neural network and a complementary metal oxide semiconductor (CMOS) to develop a decoder gate.We constantly keep in mind that the goal should be to construct a COMS circuit at the lowest feasible cost.An artificial neural network with multiple layers was used to create the circuit in this work. In our study, we treat negative values as inverters and neurons as transistors, and we modify the value using weights.Additionally, we are creating Verilog-HdL code that will make it simple to use the Decoder to experiment with artificial neural networks and adjust weights to get the desired outcomes.

The purpose of this project is to use an artificial neural network and a complementary metal oxide semiconductor (CMOS) to develop a decoder gate.We constantly keep in mind that the goal should be to construct a COMS circuit at the lowest feasible cost.An artificial neural network with multiple layers was used to create the circuit in this work. In our study, we treat negative values as inverters and neurons as transistors, and we modify the value using weights.Additionally, we are creating Verilog-HdL code that will make it simple to use the Decoder to experiment with artificial neural networks and adjust weights to get the desired outcomes. The purpose of this research is to use an artificial neural network and a complementary metal oxide semiconductor (CMOS) to produce a multiplexer gate.We always keep in mind that the lowest feasible cost should be the goal while designing any COMS circuit.This work used a multilayer artificial neural network to design the circuit. In our work, we treat neurons as transistors and negative values as inverters, and we use weights to adjust the value.In order to effortlessly apply the Multiplexer for experimenting with an artificial neural network and set weights to obtain the desired results, we are also building Verilog-HdL code.Keywords: Multiplexer, CMOS,ANN,Verilog-HDL.

Keywords: Multiplexer De-multiplexer, Decoder, Encoder Verilog-HDL, CMOS, ANN, Verilog HDL

2:1 Multiplexer:

Introduction:

A multiplexer, also called a data selector, is an electrical device that picks one or more analog or digital input signals and sends the chosen signal to a single output line. It is also frequently referred to as a mux. Select lines are a different set of digital inputs that control the selection.A complete snake is more adaptable than a half viper because it can add three information bits. By connecting several complete adders, it can also be used to add multi-bit values. Because the complete viper has a convey input, it may chain many adders together and achieve multi-bit number expansion.

The process of delivering one or more digital or analog signals over a single transmission line at various times or speeds is known as multiplexing, and the equipment we use to accomplish this is known as a multiplexer.

The multiplexer, often known as a "MPX" or "MUX," is a combinational logic circuit that uses a control signal to switch one or more input lines to a single common output line. Multiplexers function similarly to extremely quick-acting rotary switches with several positions, linking or managing several input lines, referred to as "channels," one at a time to the output.

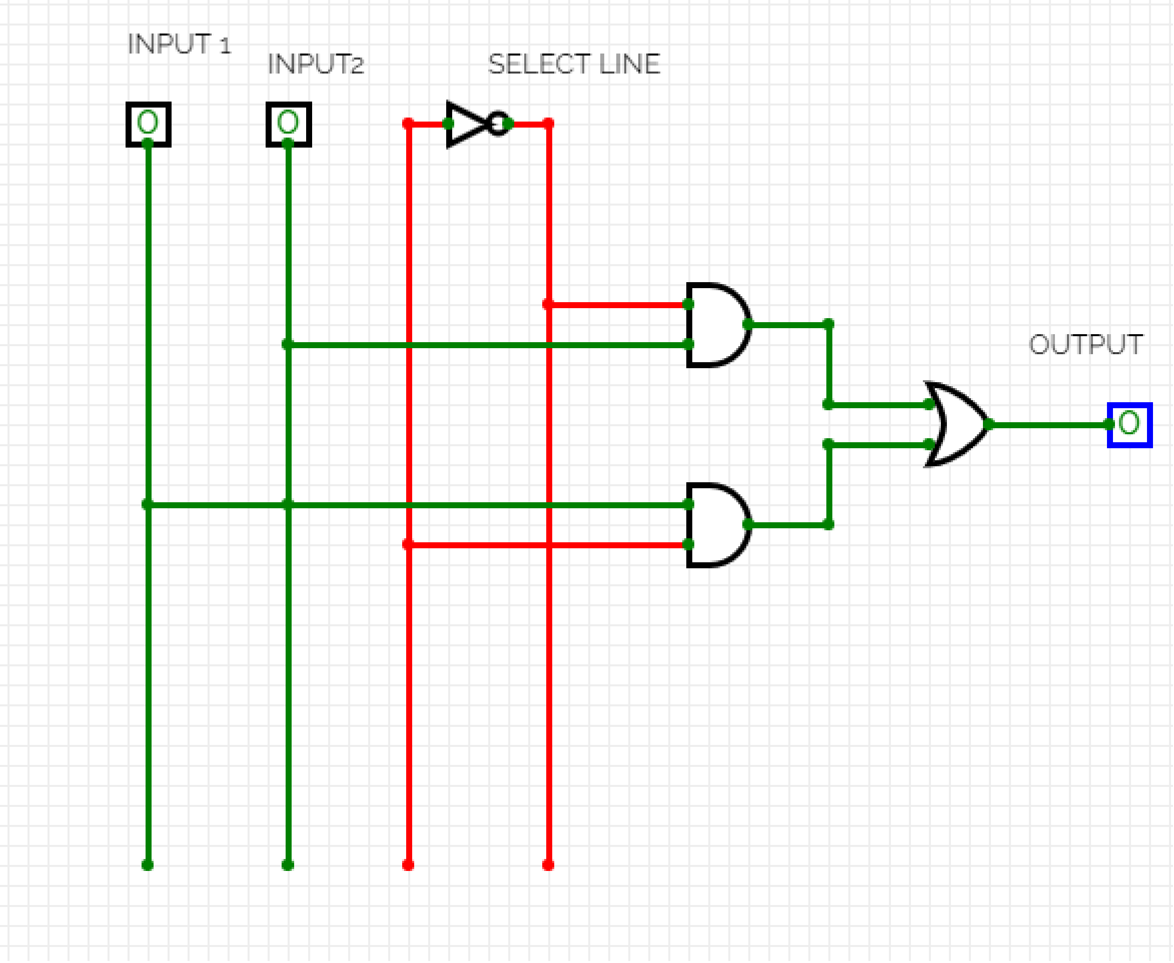


Figure 1: Circuit Diagram of Multiplexer

Truth Table for 2:1 Multiplexer

Input	Select Line(S)	Output
I1,I2	0	I1
I1,I2	1	I2

So, the equation for 2:1 Multiplexer is

Y=I1*s'+I2s

Methodology for CMOS Design:

The design of complementary metal-oxide-semiconductor (CMOS) VLSI (very-large-scale integration) semiconductors has enabled the massive scaling of a variety of semiconductor devices. Together, VLSI and the CMOS technology have reduced package sizes without sacrificing affordability. Despite not being as compact and dense as FinFET technology, CMOS technology is still significant in older technology nodes that are not expected to be phased out anytime soon. If you're constructing a new integrated circuit, SoC, or other component, CMOS VLSI design will almost certainly be required.

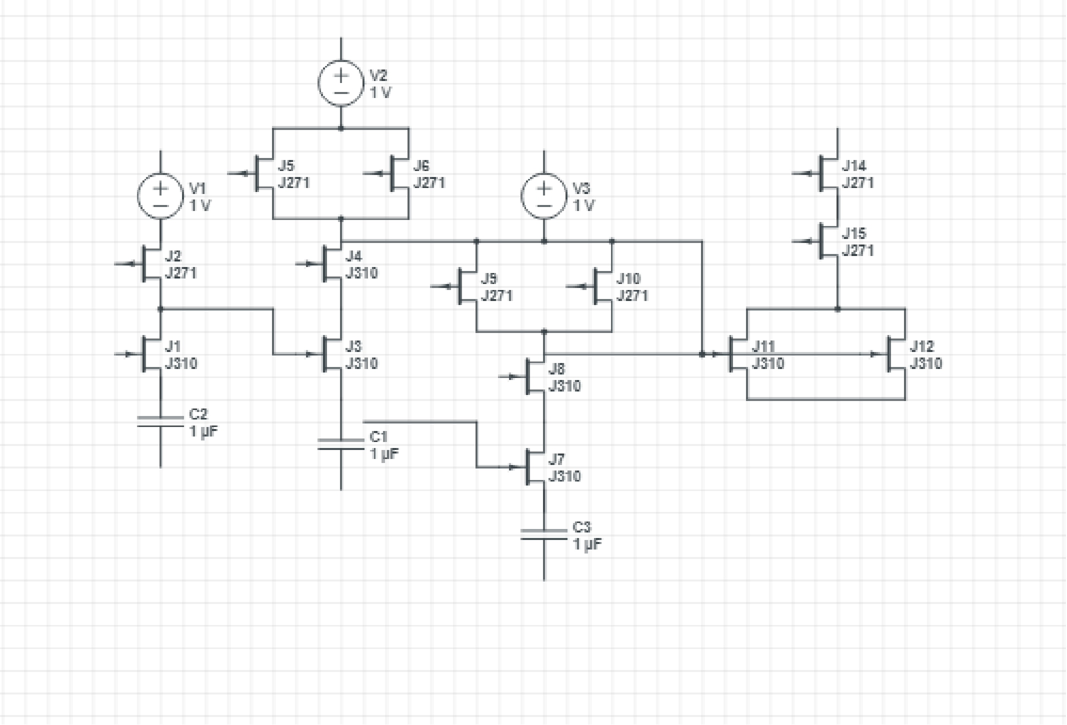


Figure 2: Transistor level design of 2_1 Multiplexer

Artificial Neural Network Design for CMOS of SUM and Carry:

Artificial neural networks are made up of units, also referred to as artificial neurons. These units are arranged in a series of layers to form the Artificial Neural Network of a system. Several dozen to millions of units can make up a layer, depending on how many intricate neural networks are required to find the hidden patterns in the dataset. The typical architecture of artificial neural networks includes input, output, and hidden layers. The neural network receives external data for analysis or training at the input layer. Subsequently, the data passes through one or more hidden layers that transform the input into information that is beneficial for the output layer.

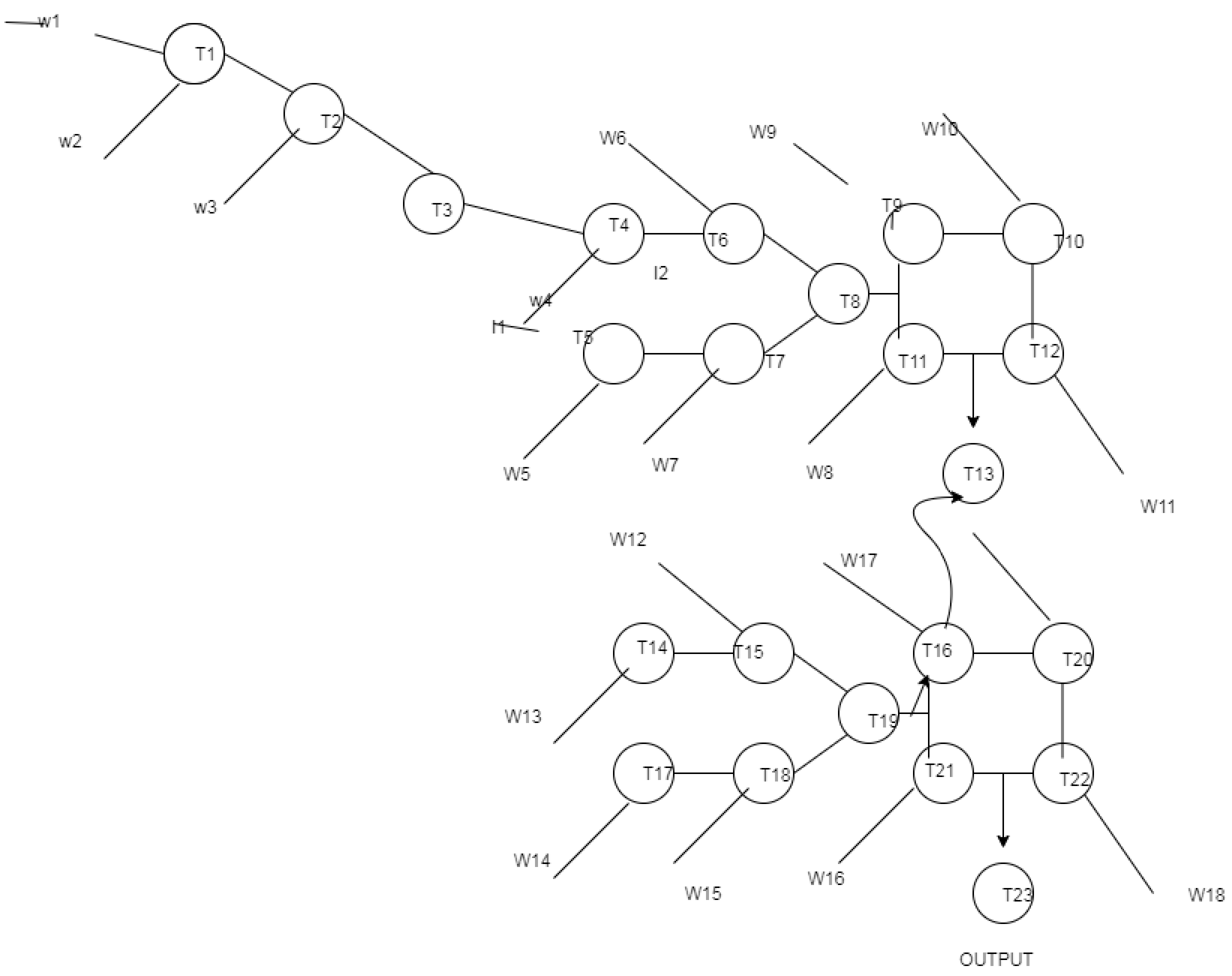
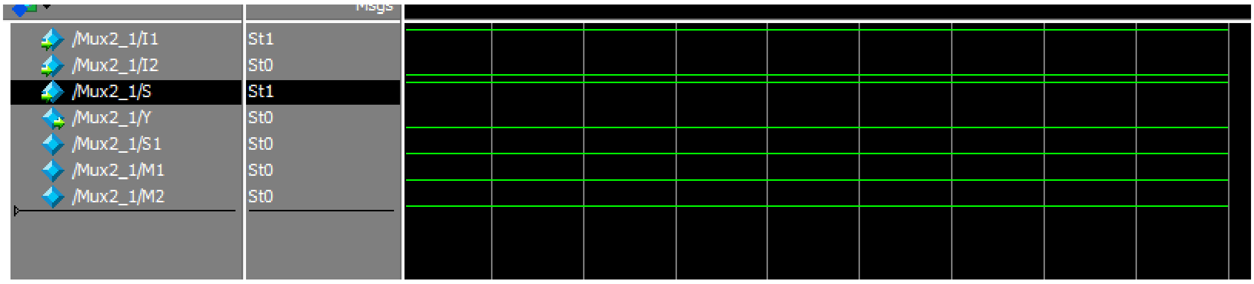


Fig 3: CMOS Level Design implementing Artificial Neural Network

Result in Verilog-HDL and Weight Calculation:

The term "Multiplexer" refers to an adder that takes three inputs and outputs two. A and B are the initial two inputs, while an input carry designated as C-IN is the third input. The normal output, denoted as S, is SUM, while the output carry is indicated as C-OUT.

Since the complete adder is a combinational circuit, Verilog can be used to simulate it. Below is the logical expression for the two outputs, sum and carry. For two-bit binary values, A and B are the input variables, Cin is the carry input, and Cout is the output variable for sum and carry.



```
                Msgs
/Mux2_1/I1      St1
/Mux2_1/I2      St0
/Mux2_1/S       St1
/Mux2_1/Y       St0
/Mux2_1/S1      St0
/Mux2_1/M1      St0
/Mux2_1/M2      St0
```

Fig6:Verilog Output for Multiplexer

Weight Calculation for Multiplexer;

T1=W1*DO;----(1)

T2=d0*W2-----(2)

T3=T1*T2;(3)

T4=W3*I1;----(4)

T5=W4*I2;----(5)

T6=T3*T4;----(6)

T7=T5*T3;---(7)

This is for pull down circuit pull up circuit for opposite transistor that is pmos and wightes are also applicable for all the transistors.

Finally out put

T23=T16*T20+T21+T22;---(8)

1:2 De-multiplexer

Introduction:

Combinational circuits with only one input line and two or more output lines are called de-multiplexers. The multiplexer is essentially a combinational circuit with a single input and many outputs. The output line receives the data that was sent from the single input lines. The input will be connected to one of these outputs based on the values of the selection lines. In opposition to the multiplexer is the de-multiplexer.

There are n selection lines and 2n outputs, as opposed to an encoder and a decoder. Thus, there are a total of 2n potential input combinations. De-mux is another term for demultiplexer.

1×2 De-multiplexer: This type of demultiplexer has one selection line (S0), one input (A), and only two outputs (Y0 and Y1). The input will be connected to one of the outputs based on the selected value. Below are the 1x2 multiplexer's block diagram and truth table.

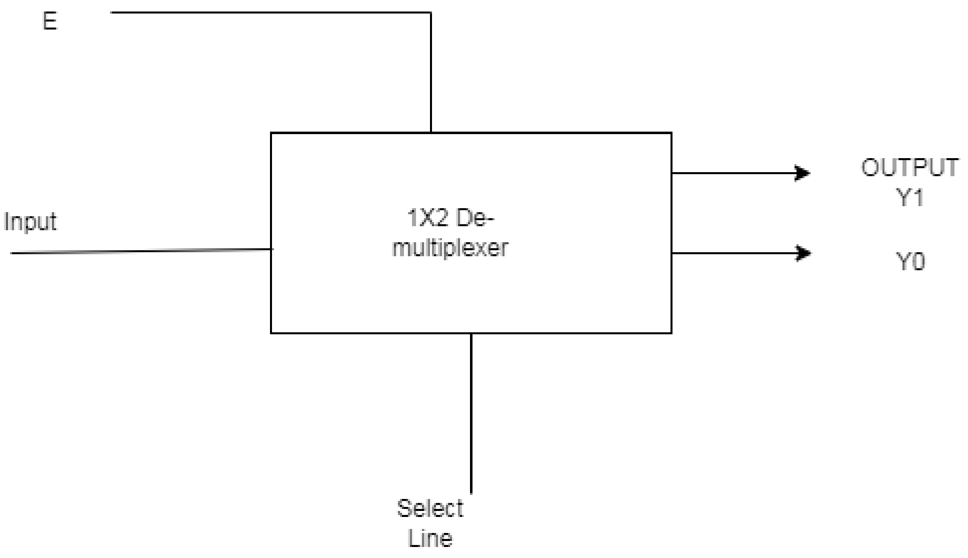


Figure 1: Block Diagram of De-multiplexer

Truth Table for 1_2 De-multiplexer

Inputs	Output	
S	Y0	Y1
0	0	A
1	A	0

So, the equation for 1_2 De-multiplexer is

Y0=S’A

Y1=SA1

Methodology for CMOS Design:

The design of complementary metal-oxide-semiconductor (CMOS) VLSI (very-large-scale integration) semiconductors has enabled the massive scaling of a variety of semiconductor devices. Together, VLSI and the CMOS technology have reduced package sizes without sacrificing affordability. Despite not being as compact and dense as FinFET technology, CMOS technology is still significant in older technology nodes that are not expected to be phased out anytime soon. If you're constructing a new integrated circuit, SoC, or other component, CMOS VLSI design will almost certainly be required.

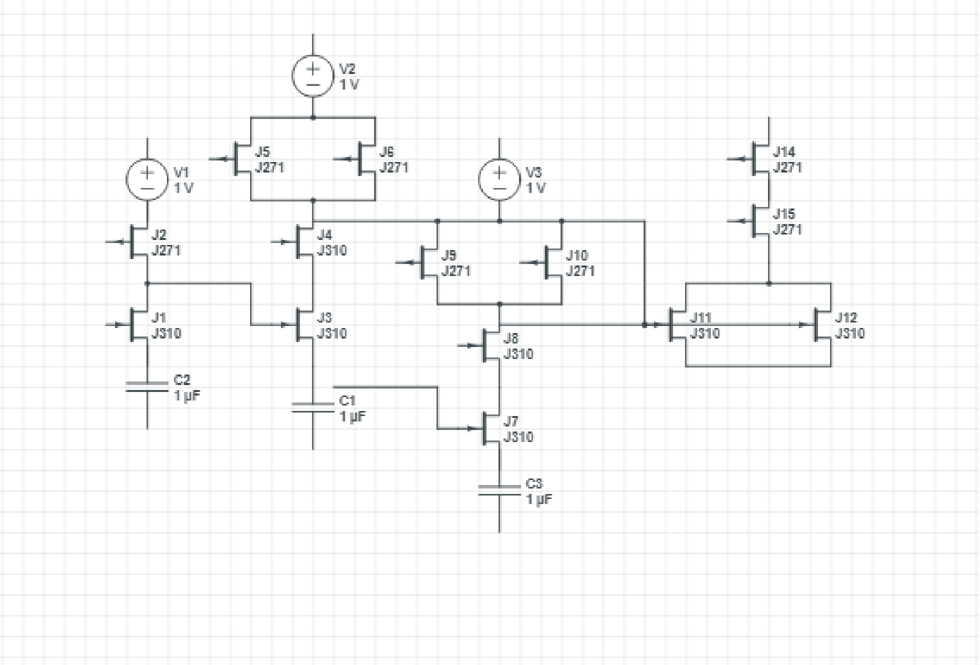


Figure 2: Transistor level design of 2_1 De-multiplexer

Artificial Neural Network Design for CMOS of SUM and Carry:

Artificial neural networks are made up of units, also referred to as artificial neurons. These units are arranged in a series of layers to form the Artificial Neural Network of a system. Several dozen to millions of units can make up a layer, depending on how many intricate neural networks are required to find the hidden patterns in the dataset. The typical architecture of artificial neural networks includes input, output, and hidden layers. The neural network receives external data for analysis or training at the input layer. Subsequently, the data passes through one or more hidden layers that transform the input into information that is beneficial for the output layer.

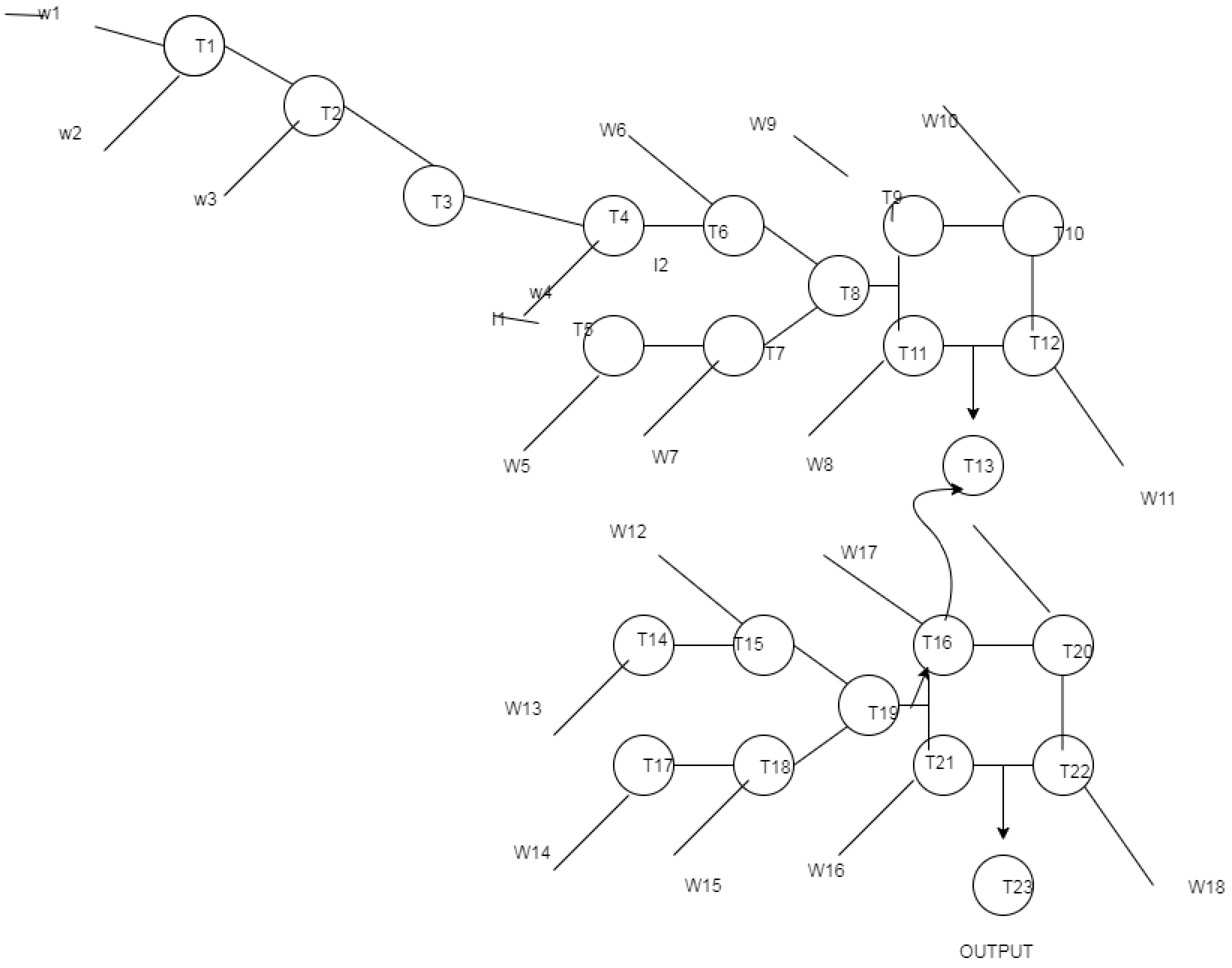
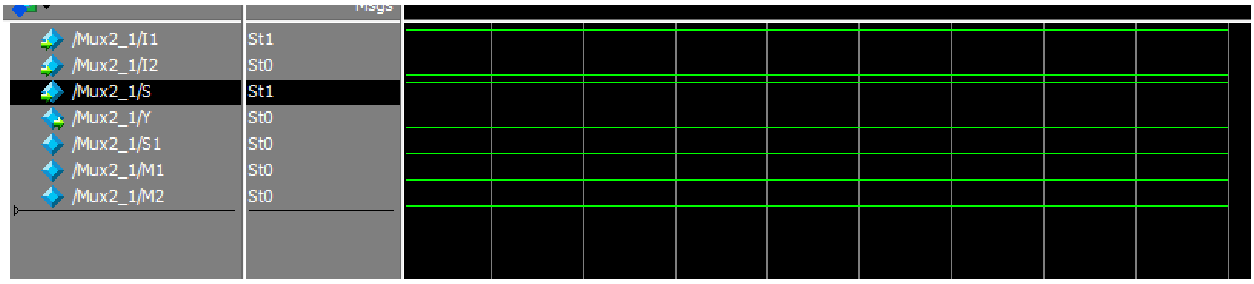


Fig 3: CMOS Level Design implementing Artificial Neural Network

Result in Verilog-HDL and Weight Calculation:

The term "De-multiplexer" refers to an adder that takes three inputs and outputs two. A and B are the initial two inputs, while an input carry designated as C-IN is the third input. The normal output, denoted as S, is SUM, while the output carry is indicated as C-OUT.

Since the complete adder is a combinational circuit, Verilog can be used to simulate it. Below is the logical expression for the two outputs, sum and carry. For two-bit binary values, A and B are the input variables, Cin is the carry input, and Cout is the output variable for sum and carry.



	Msgs
/Mux2_1/I1	St1
/Mux2_1/I2	St0
/Mux2_1/S	St1
/Mux2_1/Y	St0
/Mux2_1/S1	St0
/Mux2_1/M1	St0
/Mux2_1/M2	St0

Fig6:Verilog Output for De-multiplexer

Weight Calculation for De-multiplexer;

T1=W1*DO;----(1)

T2=d0*W2-----(2)

T3=T1*T2;(3)

T4=W3*I1;----(4)

T5=W4*I2;----(5)

T6=T3*T4;----(6)

T7=T5*T3;---(7)

This is for pull down circuit pull up circuit for opposite transistor that is pmos and wightes are also applicable for all the transistors.

Finally out put

T23=T16*T20+T21+T22;---(8)

2:4 Decoder

Introduction:

A 2 to 4 decoder is a type of combinational logic circuit that produces four output lines, commonly named Y0, Y1, Y2, and Y3, from two input lines, usually labeled A and B. After examining the input combination, the decoder turns on the matching output line. A distinct combination of the input lines is represented by each output line.

Because it contains two input lines (A and B) and four output lines (Y0, Y1, Y2, and Y3), the 2 to 4 decoder is referred to as a "2 to 4" decoder. Using the formula 2^n, where n is the number of input lines, the number of output lines is determined by the number of input lines.

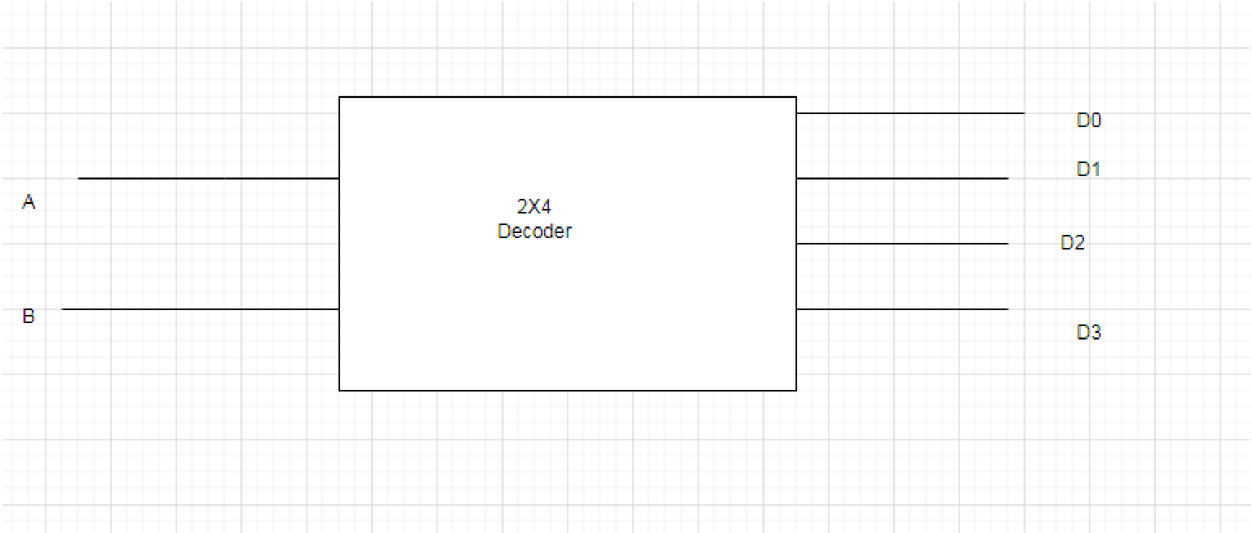


Figure 1: Block Diagram of Decoder

Truth Table for 2:4 Decoder

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

So, the equation for 2:4 Decoder is

D0=A’B’

D1=A’B

D2=AB’

D3=AB

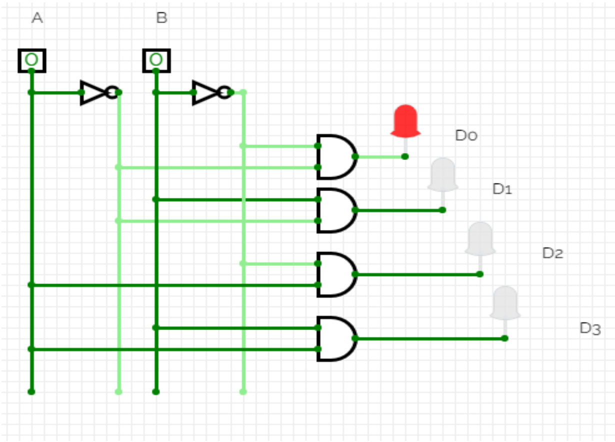


Figure 2: D0

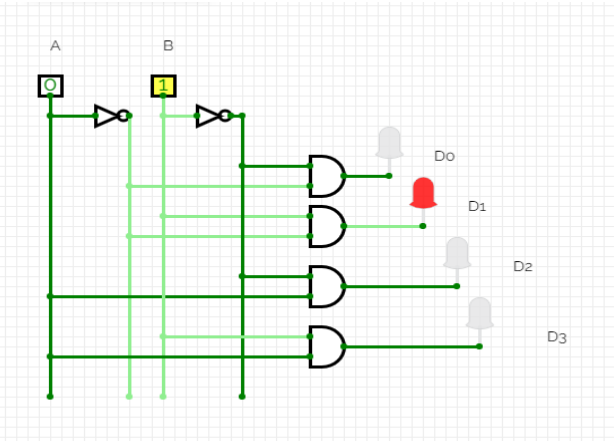


Figure 3:D1

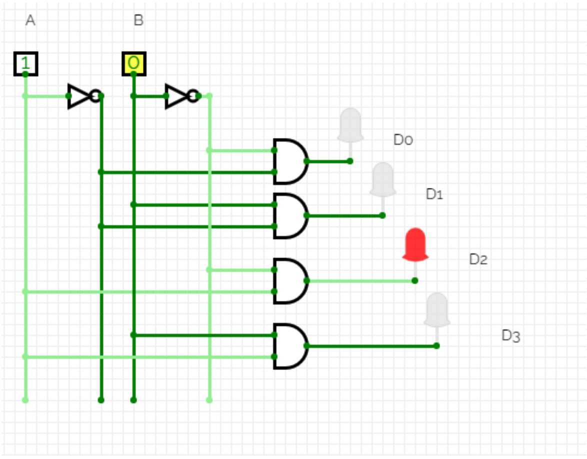


Figure 4:D2

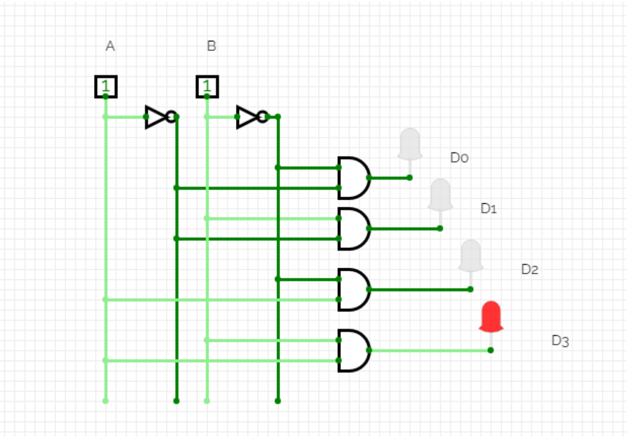


Figure 5:D3

Methodology for CMOS Design:

For "Complementary Metal Oxide Semiconductor," use the acronym CMOS. One of the most widely utilized technologies in the computer chip design business today, it forms integrated circuits for a wide range of applications. This technique is used in modern computer memory, CPUs, and cell phones because it has a number of significant benefits. Both P channel and N channel semiconductor devices are used in this technique. Complementary MOS, sometimes known as CMOS technology, is one of the most widely used MOSFET technologies on the market today. For microprocessors, microcontroller chips, memory such as RAM, ROM, and EEPROM, as well as application-specific integrated circuits (ASICs), this is the predominant semiconductor technology.

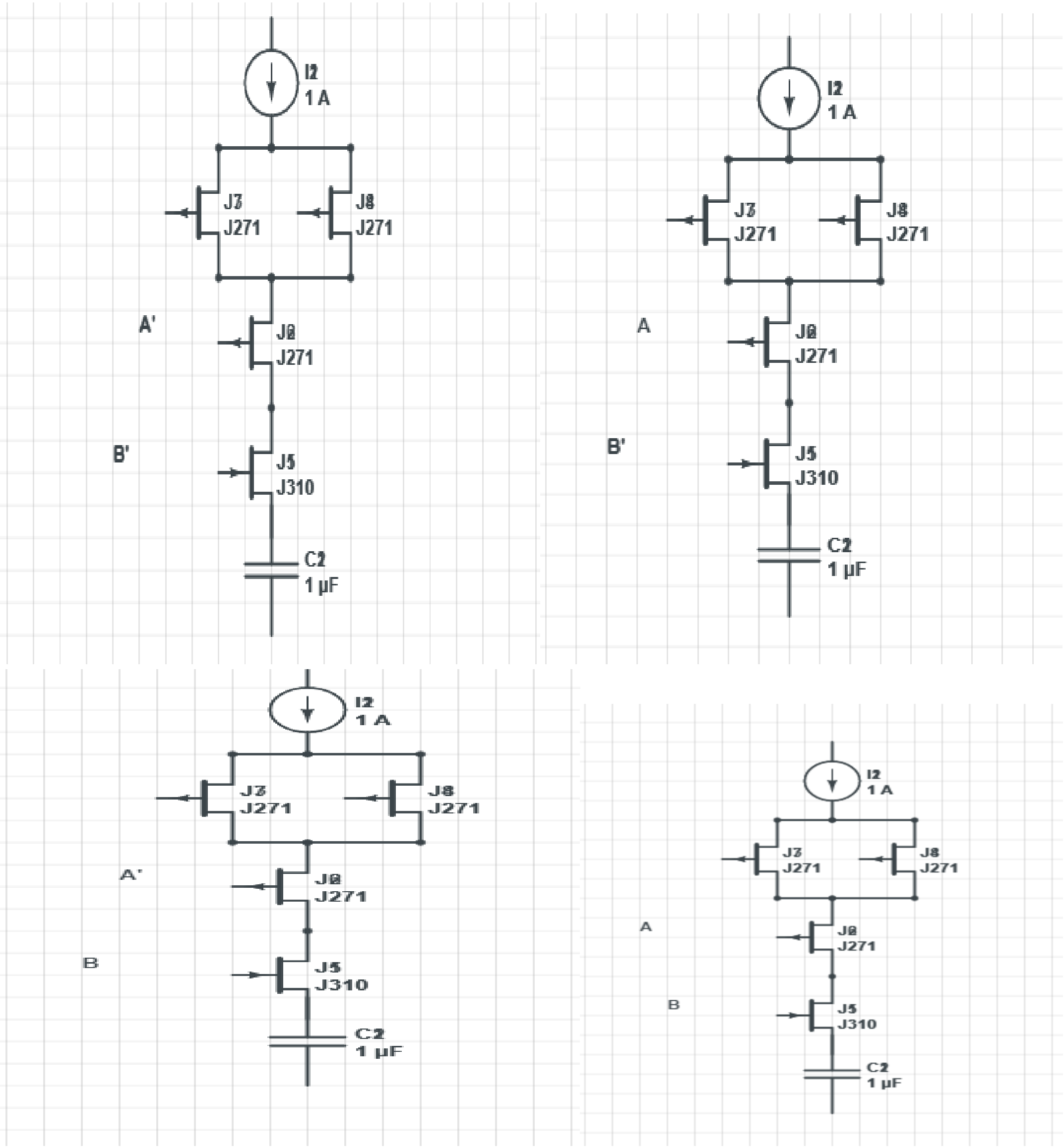


Figure 6(a) Transistor Level D0 (b) D1 (c) D2 (d) D3

Artificial Neural Network Design for CMOS Decoder:

Artificial neural networks are constructed using units, also referred to as artificial neurons. These modules are stacked one on top of the other to form the Artificial Neural Network of a system. A layer can have anywhere from a few hundred to millions of units in it, depending on how many sophisticated neural networks are required to find the underlying patterns in the dataset. The architecture of artificial neural networks frequently includes input, output, and hidden layers. The neural network receives external data at the input layer for training or analysis. After passing through one or more hidden levels, the data becomes knowledge that the output layer can employ.

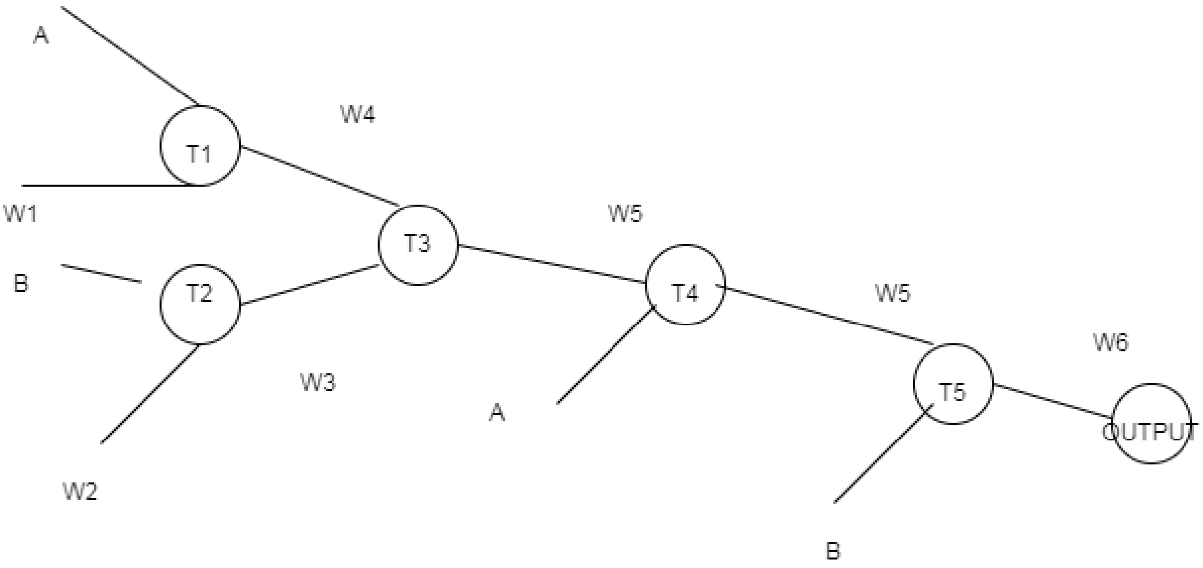
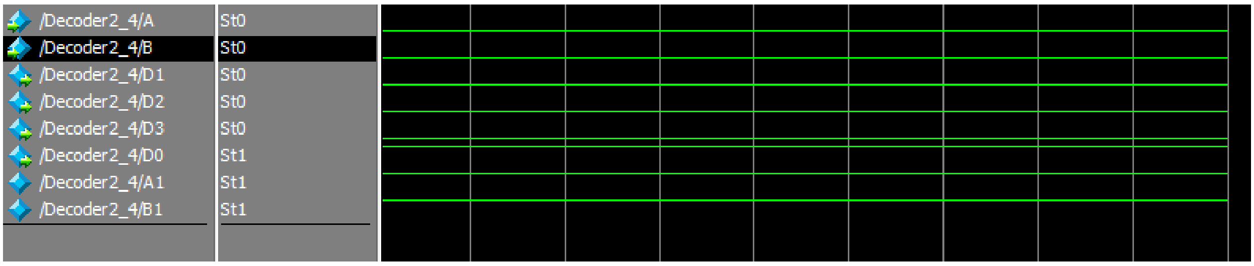


Fig 7: CMOS Level Design implementing Artificial Neural Network for D0

Result in Verilog-HDL and Weight Calculation.

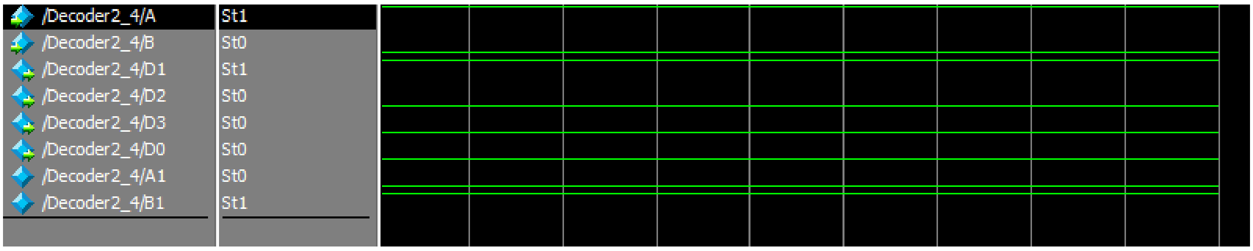
Behavioral modeling represents the circuit at a high level of abstraction. The syntax of Behavioral modeling resembles that of C programming. We can implement conditions statements like if, case statement, looping, and structure procedures(initial and always) within the Verilog design block using Behavioral modeling.

```
module Decoder2_4(A,B,D0,D1,D2,D3);

  input A,B;

  output D1,D2,D3,D0;

  wire A1,B1;

  not g1(A1,A);

  not g2(B1,B);

  and g3(D0,A1,B1);

  and g4(D1,A,B1);
  and g5(D2,A1,B);

  and g6(D3,A,B);

endmodule
```



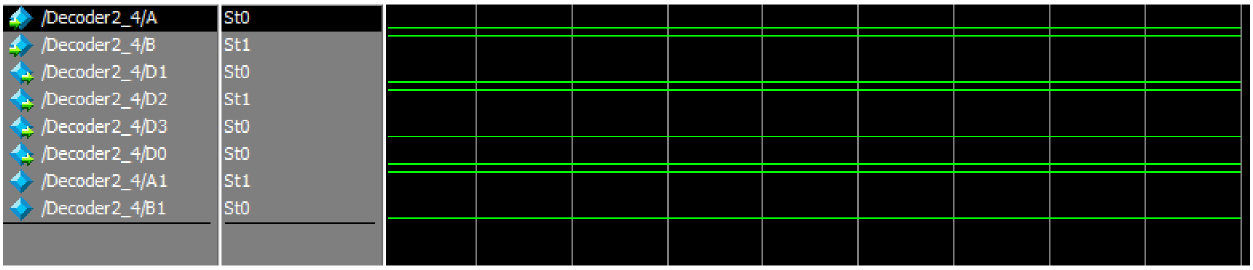
```
/Decoder2_4/A    St0
/Decoder2_4/B    St0
/Decoder2_4/D1   St0
/Decoder2_4/D2   St0
/Decoder2_4/D3   St0
/Decoder2_4/D0   St1
/Decoder2_4/A1   St1
/Decoder2_4/B1   St1
```

Fig 8:Verilog Output for Decoder D0



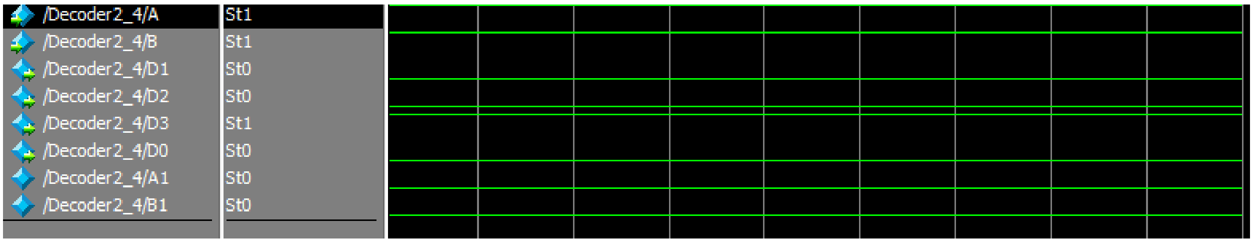
```
/Decoder2_4/A    St1
/Decoder2_4/B    St0
/Decoder2_4/D1   St1
/Decoder2_4/D2   St0
/Decoder2_4/D3   St0
/Decoder2_4/D0   St0
/Decoder2_4/A1   St0
/Decoder2_4/B1   St1
```

Fig 9:Verilog Output for Decoder D1



```
/Decoder2_4/A    St0
/Decoder2_4/B    St1
/Decoder2_4/D1   St0
/Decoder2_4/D2   St1
/Decoder2_4/D3   St0
/Decoder2_4/D0   St0
/Decoder2_4/A1   St1
/Decoder2_4/B1   St0
```

Fig 10:Verilog Output for Decoder D2



```
/Decoder2_4/A    St1
/Decoder2_4/B    St1
/Decoder2_4/D1   St0
/Decoder2_4/D2   St0
/Decoder2_4/D3   St1
/Decoder2_4/D0   St0
/Decoder2_4/A1   St0
/Decoder2_4/B1   St0
```

Fig 11:Verilog Output for Decoder D3

4:2 Encoder

Introduction:

An electromechanical device called an encoder is used to change the format or code of information. An electrical signal that indicates position, speed, and direction of motion is produced when linear or rotating motion is converted by a position encoder, such those manufactured by Renishaw.

A digital circuit known as an encoder transforms a series of binary inputs into a single binary code. The input's position is represented by the binary code, which is utilized to pinpoint the precise input that is active. In digital systems, encoders are frequently employed to translate a parallel collection of inputs into a serial code.

An encoder's fundamental function is to provide a distinct binary code to every potential input. A 2-to-4 line encoder, for instance, has two input lines, four output lines, and gives each of the two^two = four possible input combinations a distinct 4-bit binary code. An encoder's output is typically active low, which means that only one output is ever active (low) and the other outputs are always inactive (high). The binary code that is assigned to the active input determines which is the active low output.

There are various kinds of encoders, such as binary-weighted encoders that employ a binary weighting system to assign binary codes to inputs and priority encoders that give each input a priority. To put it briefly, an encoder is a type of digital circuit that takes a series of binary inputs and creates a distinct binary code that indicates the input's position. In digital systems, encoders are commonly employed to translate parallel inputs into serial codes.

A combinational circuit called an encoder carries out a decoder's opposite function. It encodes the data from 2^n inputs into an n-bit code since it can have a maximum of 2^n input lines and "n" output lines. It will output an equivalent binary code, which is active High, to the input. As a result, the encoder uses "n" bits to encode 2^n input lines.

The several categories of encoders are listed below.

4:2 Encoder

Eight to Three Octal to Binary Encoder

Decoder from Decimal to BCD

First-Class Encoder

4:2 Encoder

Four inputs (Y3, Y2, Y1 & Y0) and two outputs (A1 & A0) make up the 4 to 2 encoder. Only one of these four inputs can ever be '1' at a time in order for the output to have the appropriate binary code. The 4 to 2 encoder's logic symbol is displayed in the image below.

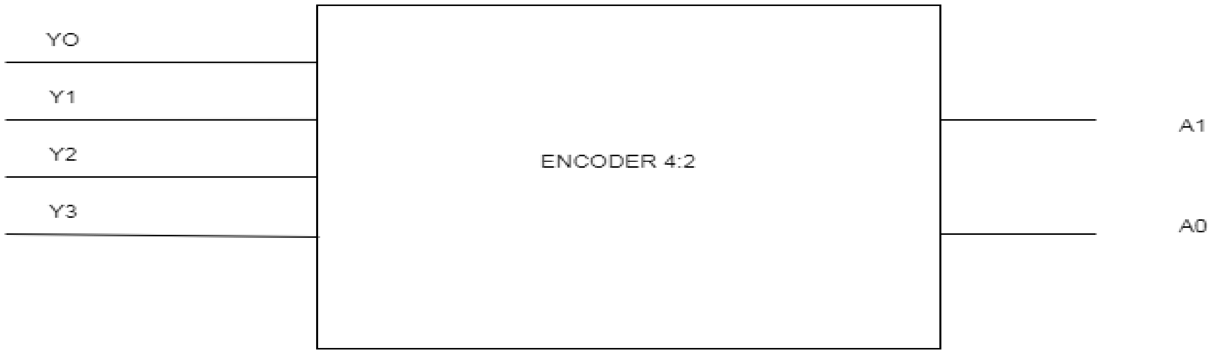


Figure 1: Block Diagram of Encoder

Truth Table for 4:2 Encoder

INPUT				**OUTPUT**	
Y3	**Y2**	**Y1**	**Y0**	**A1**	**A0**
0	**0**	**0**	**1**	**0**	**0**
0	**0**	**1**	**0**	**0**	**1**
0	**1**	**0**	**0**	**1**	**0**
1	**0**	**0**	**0**	**1**	**1**

So, the equation for 2:4 Decoder is

A0=Y1+Y3

A1=Y2+Y3

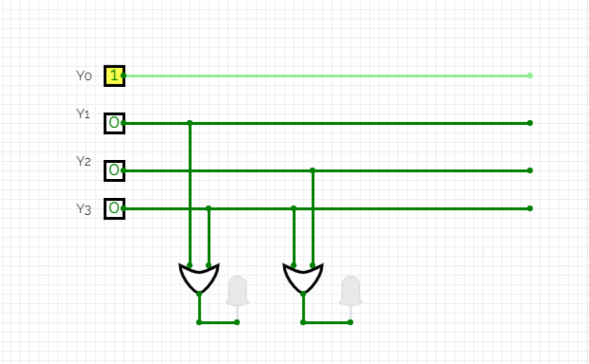


Figure 2:

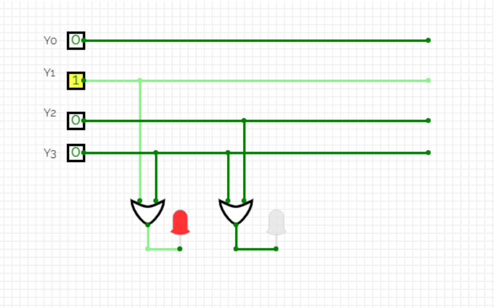


Figure 3:

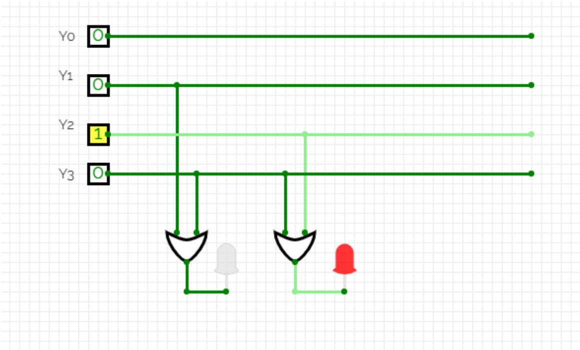


Figure 4:

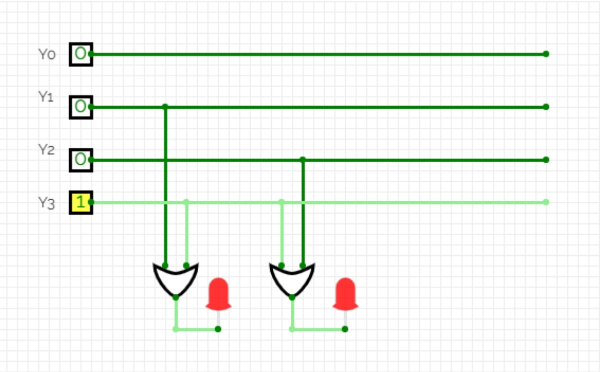


Figure 5

Methodology for CMOS Design:

The majority of integrated circuits (ICs), sometimes referred to as chips or microchips, nowadays are made with complementary metal-oxide semiconductors (CMOS) technology. Metal-oxide semiconductor field-effect transistor (MOSFET) technology is the foundation of CMOS transistors.Combining NMOS with PMOS transistors, CMOS, or complementary metal oxide semiconductor, functions when an applied electrical field is present. CMOS was first designed with high density and low power logic gates in mind.Metal Oxide Semiconductor Field Effect Transistors (MOSFETs) come in two varieties: NMOS and PMOS. Applications for CMOS transistors include integrated circuit chips, microprocessors, switching circuits, logic circuits, amplifiers, and more.Because of its low operating current and low power dissipation, CMOS is crucial to semiconductor technology. In contrast to bipolar junction transistors, it takes fewer steps in the production process.As was previously said, CMOS transistors combine NMOS and PMOS technology. To start with CMOS, let's take a brief look at the NMOS and PMOS transistors.

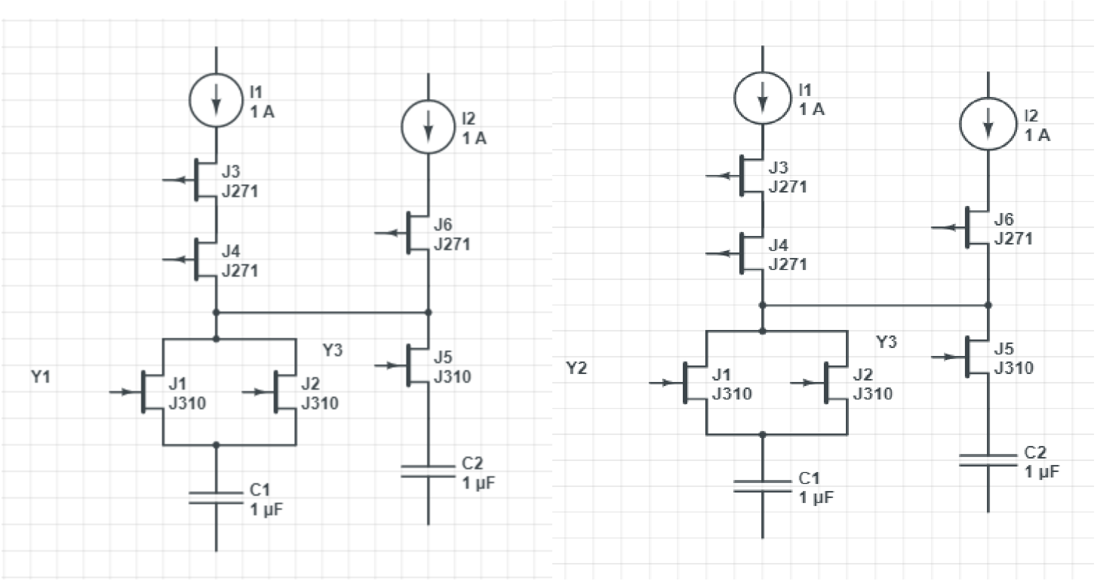


Figure 6(a) Transistor Level Design of A0 and A1

Artificial Neural Network Design for CMOS Encoder coder:

Artificial neural networks are constructed using units, also referred to as artificial neurons. These modules are stacked one on top of the other to form the Artificial Neural Network of a system. A layer can have anywhere from a few hundred to millions of units in it, depending on how many sophisticated neural networks are required to find the underlying patterns in the dataset. The architecture of artificial neural networks frequently includes input, output, and hidden layers. The neural network receives external data at the input layer for training or analysis. After passing through one or more hidden levels, the data becomes knowledge that the output layer can employ.

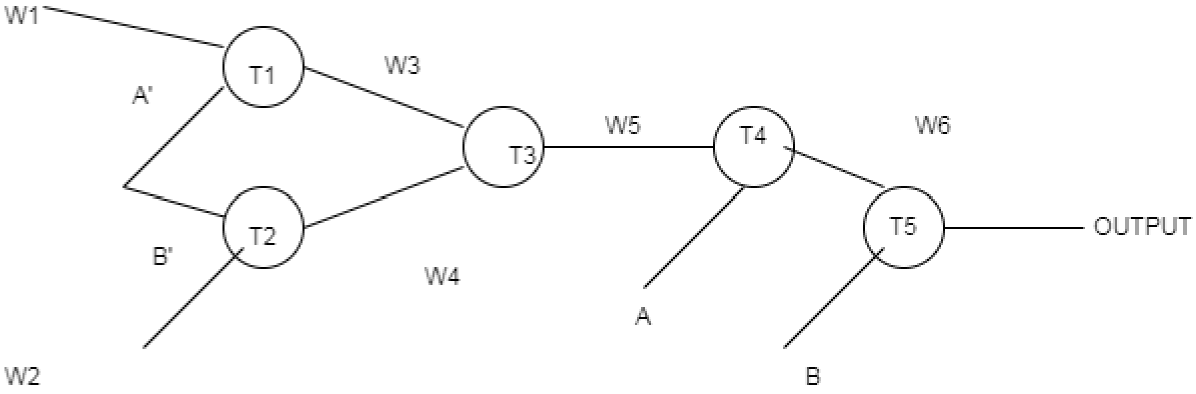
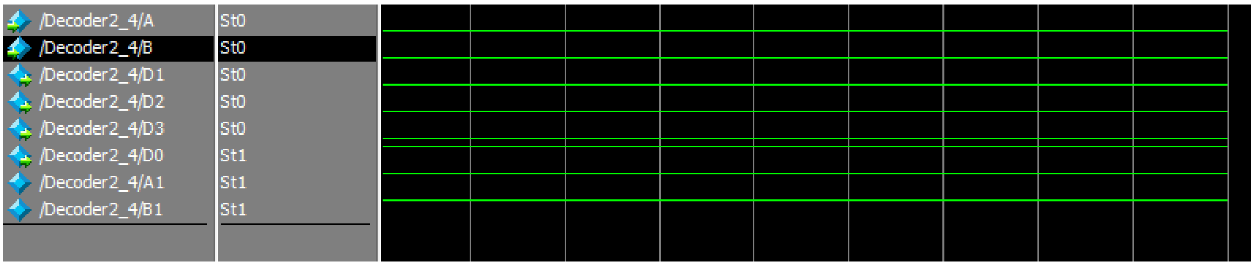


Fig 7: CMOS Level Design implementing Artificial Neural Network for Encoder Circuit for A0

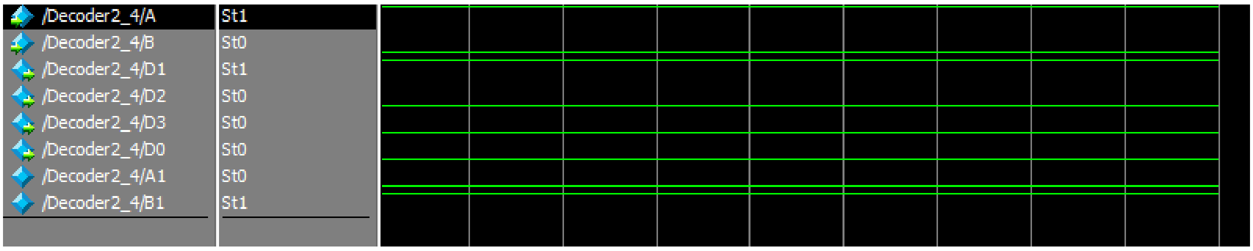
Result in Verilog-HDL and Weight Calculation.

A more abstract representation of the circuit is provided via behavioral modeling. Behavioral modeling uses a syntax similar to C programming. By employing behavioral modeling, we may incorporate conditions statements such as if, case statements, looping, and structural procedures (initial and always) inside the Verilog design block.



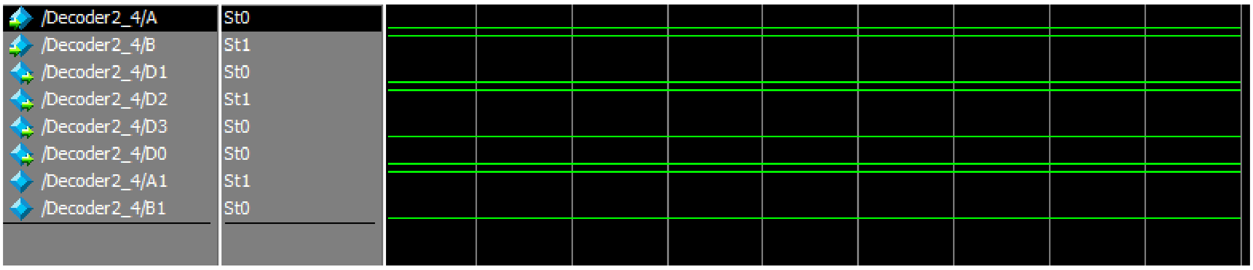
```
/Decoder2_4/A   St0
/Decoder2_4/B   St0
/Decoder2_4/D1  St0
/Decoder2_4/D2  St0
/Decoder2_4/D3  St0
/Decoder2_4/D0  St1
/Decoder2_4/A1  St1
/Decoder2_4/B1  St1
```

Fig 8:Verilog Output for Decoder D0



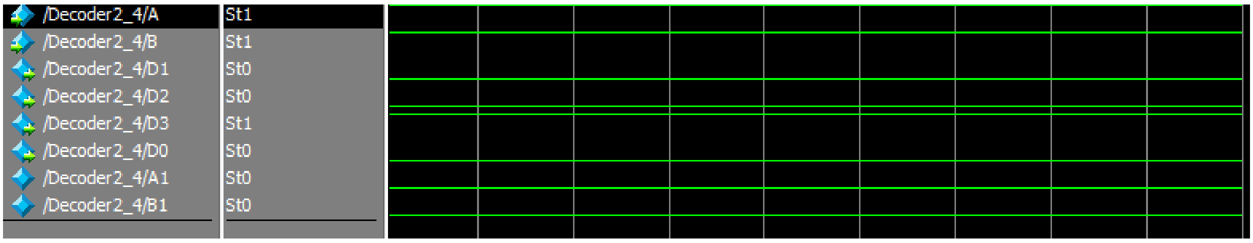
```
/Decoder2_4/A   St1
/Decoder2_4/B   St0
/Decoder2_4/D1  St1
/Decoder2_4/D2  St0
/Decoder2_4/D3  St0
/Decoder2_4/D0  St0
/Decoder2_4/A1  St0
/Decoder2_4/B1  St1
```

Fig 9:Verilog Output for Decoder D1



```
/Decoder2_4/A   St0
/Decoder2_4/B   St1
/Decoder2_4/D1  St0
/Decoder2_4/D2  St1
/Decoder2_4/D3  St0
/Decoder2_4/D0  St0
/Decoder2_4/A1  St1
/Decoder2_4/B1  St0
```

Fig 10:Verilog Output for Decoder D2



```
/Decoder2_4/A   St1
/Decoder2_4/B   St1
/Decoder2_4/D1  St0
/Decoder2_4/D2  St0
/Decoder2_4/D3  St1
/Decoder2_4/D0  St0
/Decoder2_4/A1  St0
/Decoder2_4/B1  St0
```

Fig 11:Verilog Output for Decoder D3

Conclusion:

The goal of this research is to create a 2:1 De-multiplexer circuit using a complementary metal oxide semiconductor (CMOS) and an artificial neural network.When designing any COMS circuit, we always keep in mind that the lowest possible cost should be the aim.In this work, the circuit was designed using a multilayer artificial neural network. We utilize weights to alter the value and treat negative values as inverters and neurons as transistors in our work.Furthermore, we're developing Verilog-HdL code that may be used to the complete adder to facilitate artificial neural network experiments and the allocation of weights to yield pertinent outcomes.

The goal of this project is to create a 2:4 Decoder circuit using a complementary metal oxide semiconductor (CMOS) and an artificial neural network.We never forget that building a COMS circuit at the lowest practical cost should be the ultimate goal.The circuit in this work was constructed using a multilayer artificial neural network. We use weights to adjust the value in our study, treating neurons as transistors and negative values as inverters.Furthermore, we are developing Verilog-HdL code that can be applied to the complete adder to facilitate the testing of artificial neural networks and the optimization of weight distribution to yield pertinent outcomes.

The goal of this project is to create a 4:2 Encoder circuit using a complementary metal oxide semiconductor (CMOS) and an artificial neural network.We never forget that building a COMS circuit at the lowest practical cost should be the ultimate goal.The circuit in this work was constructed using a multilayer artificial neural network. We use weights to adjust the value in our study, treating neurons as transistors and negative values as inverters.Furthermore, we are developing Verilog-HdL code that can be applied to the complete adder to facilitate the testing of artificial neural networks and the optimization of weight distribution to yield pertinent outcomes.

Reference:

1] R. K. Mandal, Design of a CMOS “OR Gate” using Artificial Neural Networks (ANNs), AMSE JOURNALS-2016-Series:
Advances D; Vol. 21; N°1; pp 66-77
[2.] Ashutosh Aggarwal, Rajneesh Rani, RenuDhir, “Handwritten Devanagri Character Recognition using Gradient Features”, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 5, May 2012, ISSN 2277 128X, pp. 85-90.

[3.] Sandeep, Saha, Nabarag Paul, Sayam Kumar Das, Sandip Kundu, “Optical Character Recognition using 40-point
Feature Extraction and Artificial Neural Network”, International Journal of Advanced Research in Computer Science
and Software Engineering, Volume 3, Issue 4, April 2013, ISSN 2277 128X, pp. 495-502.
[4.] Ali Borji, Mandana Hamidi, Fariborz Mahmoudi, “Robust Handwritten Character Recognition with Features Inspired by Visual Ventral Stream”, © Springer Science+Business Media, LLC. 2008, published online (31 August 2008), pp. 97-111.
[5.] Y Perwej and A Chaturvedi, “Neural Networks for Handwritten English Alphabet Recognition”, International Journal of Computer Applications, Volume 20, No. 7, pp. 1-5, 2011.
[6.] Frye R C, Rietman E A, and Wong C C, “Back-propagation learning and non idealities in analog neural network
hardware,” Neural Networks, IEEE Transactions on, vol. 2, no. 1, pp. 110–117, 1991.

[7.] Jung S and Kim S S, "Hardware implementation of a real-time neural network controller with a dsp and an fpga
for nonlinear systems," Industrial Electronics, IEEE Transactions on, vol. 54, no. 1, pp. 265–271, 2007.
[8.] Hikawa H, "{FPGA} implementation of self organizing map with digital phase locked loops", Neural Networks, vol.
18, no. 56, pp. 514 – 522, 2005, {IJCNN} 2005. Available Online:
http://www.sciencedirect.com/science/article/pii/S0893608005001103
[9.] Merolla P A, Arthur J V, Alvarez-Icaza R, Cassidy A S, Sawada J, Akopyan F, Jackson B L, Imam N, Guo C, Nakamura
Y, Brezzo B, Vo I, Esser S K, Appuswamy R, Taba B, Amir A, Flickner M D, Risk W P, Manohar R, and Modha D S, "A
million spiking-neuron integrated circuit with a scalable communication network and interface", Science, Vol. 345, No. 6197, pp 668–673, 2014.
[10.] Forssell M, "Hardware Implementation of Artificial Neural Networks", 18-859E INFORMATION FLOW IN NETWORKS, pp 1-4, "Available: http://users.ece.cmu.edu/~pggrover/teaching/files/NeuromorphicComputing.pdf",
(Accessed : 2016)
[11.] Yellamraju S, Kumari Swati, Girolkar S, Chourasia S and Tete A D, "Design of Various Logic Gates in Neural
Networks", Annual IEEE India Conference (INDICON), 2013, Mumbai, India.
[12.] Hawas N M, Rekaby B K A, "ANN Based On Learning Rule Of Neuron Activation Function Using Electronic
Devices", International Journal of Advanced Computer Technology (IJACT), Vol 4, No. 3, pp 19-22, 2015.
[14.] Kale N B, Padole V B, "Compression and Decompression of Signal Using CMOS Technology...A Review", International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 4, Issue 3, pp 53-55,
2014.