ISSN: 2395-1680 (ONLINE)
DOI: 10.21917/ijme.2021.0200

THE EFFICIENT IMPLEMENTATION TO OPTIMIZE POWER AND DELAY USING DATA SELECTOR

Jogi Prakash, Biroju Ravi Kiran and Parvatham Sathish
Department of Electronics and Communication Engineering, Sree Dattha Group of Institutions, India

Abstract

The objective of this paper is designing a 16:1 multiplexer using logic gates and CMOS logic. In this research, we have investigated the delay and power modulations of 16:1MUX. This demonstrates that the CMOS technique takes lead as it uses decreased number of transistors, have less capacitances and faster than others. In this research a comparative work is done and made the simulated results and it illustrates the superior nature of CMOS logic design and it dissipates very decreased power and delay. The simulations for the proposed model are done by using Synopsys tool HSPICE under 32 nm BSIM 4 model card for bulk CMOS technology of PTM model and examined the results with varying voltages. The minimum and maximum delay and power dissipation results are 68.82ps, 92.16ps and 103.96µW, 1471.4µW respectively. The overall transistor count we got in the Multiplexer is 282 and this is simulated and we got output waveforms of the MUX by using the advanced tool called HSPICE and they are represented in the results section.

Keywords:

Multiplexer, 2×1 Multiplexer, 4×1 Multiplexer, 8×1 Multiplexer, 16×1 Multiplexer, Delay, Power Dissipation

1. INTRODUCTION

Currently multiplexer has major role in the field of the communication. So, there is a need of designing a less area, low power and high-speed multiplexer. Multiplexer is a digital component which is used as a data selector [1]. Multiplexer is used for building a combinational circuit [2]. Multiplexer is also termed as MUX [3]. Multiplexer generates one of the inputs as output based on the given selection line. Multiplexer is a digital component transmits multiple data on a common transmission medium.

Multiplexer has many inputs with single output [4]. Multiplexer is used as time division multiplexer; the selected data is transmitted at a particular instant of the time. Multiplexer has 2^n inputs and n selection lines. Based on the selection lines it provides one output from given 2^n inputs [9]. Multiplexer is used to transmit audio and video signals. Telephone network uses multiplexer to transmit the desire audio to the receiver. It connects many users on single transmission channel [14]. Based on the requirement of the low power and high-speed device in many applications, a 16×1 multiplexer is designed.

The research paper is a general overview on advanced system of multiplexer 16:1. This research paper has a brief introduction about an electronic gadget called multiplexer. A MUX or data selector is a device that is having the ability of taking two or more data lines and changing them, into a unique data line for transmission to another resource [15]. 16×1 multiplexer is introduced by using the basic multiplexer concepts: For example, if we start with the 2:1 MUX circuit. It has one output, two inputs and one selection input. The way we are selecting the particular input logic is maintained by a set of defined lines. This defined line determines which of the input bit is transferred to the output [1]-[4]. Here in this modification, we are implementing a 16:1 MUX with logic gates and when we have analysed the output with some power, delay and PDP, the count of the transistors came out as 282.

This paper is presented as: Section 2 is literature survey, section 3 is the explanation of the different types of the multiplexers, section 4 is the proposed design, section 5 is simulation results and section 6 is conclusion of the paper.

2. LITERATURE SURVEY

2×1 multiplexer is a two-input block with one output, one selection line and one enable signal which makes the block to work when it is at logic 1. The operation of the 2×1 is understood by observing the truth table of 2×1 multiplexer [1]. 2×1 multiplexer is designed with 10 transistors and simulated on HSPICE with 1V supply power under various technologies [2]. 2×1 multiplexer is designed by using 2-transistor logic where 2 transistors are used to design a 2×1 multiplexer. It is an optimized 2×1 multiplexer with fewer transistors [3]. 2×1 multiplexer is designed by using NAND logic, NOR logic and CMOS logic. CMOS logic based 2×1 multiplexer is better in terms of power and area [4]. 2×1 multiplexer is designed with two AND gates, one Inverter and One OR gate.

4×1 multiplexer is a block with four inputs with two selection lines and single output. Output for different combination of inputs based on the selection lines can be observed [5]. 4×1 multiplexer designed by using four AND gates, two inverters and single OR gate [6]. 4×1 multiplexer is designed by using CMOS logic with the help of PMOS and NMOS transistors [7]. CMOS based 4×1 multiplexer is designed and simulated in tanner tool with 1V supply voltage [8]. Complete 4×1 block, operation of the 4×1 multiplexer, gate level circuit design of 4×1 multiplexer and CMOS based 4×1 is explained. Power dissipation and delay parameters are explained. Simulations are done on cadence tool [9]. 2×1 multiplexer is designed in schematic window and symbol is created for it. 4×1 multiplexer is designed and simulated using EDA software and VHDL language [10].

8×1 multiplexer is with eight inputs, three selection lines and single output [11]. 8×1 multiplexer is designed in gate level by using 12 AND gates, 5 OR gates and 6 inverters. CMOS based 8×1 multiplexer is designed with 126 transistors [12]. 8×1 multiplexer is designed using 2× multiplexer simulated in tanner tool [13]. Multiplexer is used in many applications like communications. In telephone networks multiple calls are connected in a single line by using multiplexing. It uses the satellite communication for transferring the data [14]. Basics of the logic gates, operation of the logic gates and the operation of the multiplexer are shown in [15].

3. MULTIPLEXER

A multiplexer is a digital logic circuit which selects output data from given n sources. A selected input determines which data is appeared to the output. A multiplexer of 2*N* inputs has *N* selection lines; these are considered to select which input data as the output.

3.1 2×1 MULTIPLEXER

A 2x1 multiplexer in Fig.1 shows 2-inputs I_0 to I_1 are inputs terminals. Selection lines S to select a particular input data. The multiplexer block diagram of 2:1 mux is in the Fig.1 [1].

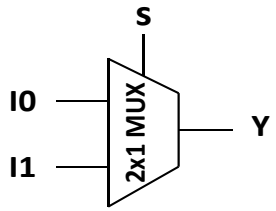


Fig.1. Multiplexer block diagram of 2:1

The equation for a 2:1 multiplexer is Eq. (1)

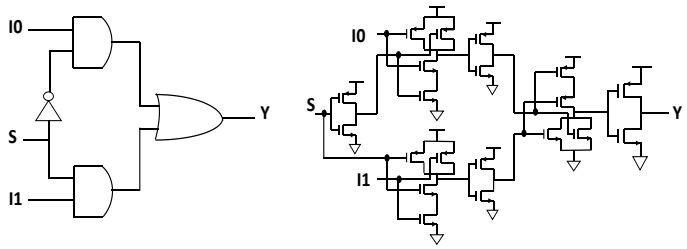
$$Y = I_0S'+I_1S \quad (1)$$

Inputs are I_0, I_1 and S is the selection line and the single Y (output).

Table.1. Truth Table of 2×1 multiplexer

S	*Y*
0	I_0
1	I_1

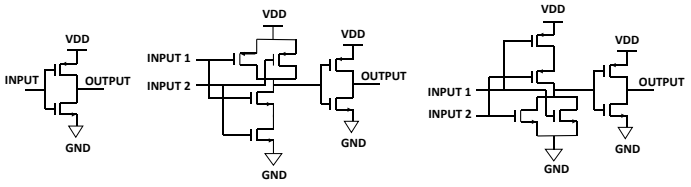
As given Table.1 while 0 is the selection line, the output is the value of the I_0 and while 1is the selection line the output is the value of the I_1. The gate level diagram of the 2×1 multiplexer is given in Fig.2. The gate level 2×1 multiplexer as two AND gates, one inverter and one OR gate.



(a) (b)

Fig.2. (a) Gate level diagram of the 2x1 multiplexer (b) CMOS representation of 2x1 multiplexer

One AND gate given with input I_0 and inversion of the selection line(s). Other AND gate is given with inputs I_1 and selection line(s). The output of the two AND gate is given as input to OR gate. The OR gate gives the output *Y*. 2×1 multiplexer is designed by using CMOS with PMOS and NMOS transistors as shown in Fig.2(b).



(a) (b) (c)

Fig.3. (a) CMOS inverter (b) CMOS two input AND gate (c) CMOS two input OR gate

CMOS Inverter is designed with one PMOS and one NMOS transistors. CMOS inverter attached at the output of the CMOS NAND gate this gives a CMOS AND gate. CMOS NAND gate is formed by sequence connection of the NMOS transistors and side by side connection of the PMOS transistors. CMOS OR gate is designed with one CMOS NOR gate with a CMOS inverter connection at the output of the CMOS NOR gate this gives a CMOS OR gate. NOR gate of CMOS is formed by series connection of PMOS transistors and parallel connection of the NMOS transistors as shown in *Fig.*3.

3.2 4×1 MULTIPLEXER

A 4 to 1 multiplexer is given with inputs I_0, I_1, I_2 and I_3 as shown in Fig.4. It as selection line S_0 and $S_1$and output Y.

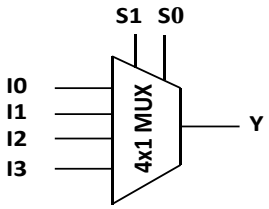
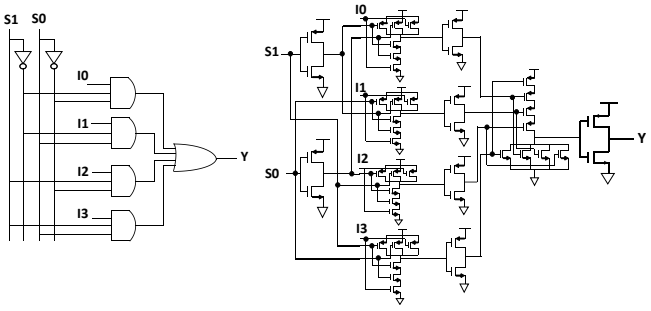


Fig.4. 4×1 multiplexer Block Diagram

The equation for a 4:1 multiplexer is Eq. (2)

$$Y=I_0S_1' S_0'+I_1S_1' S_0+I_2S_1 S_0'+I_3S_1 S_0 \quad (2)$$

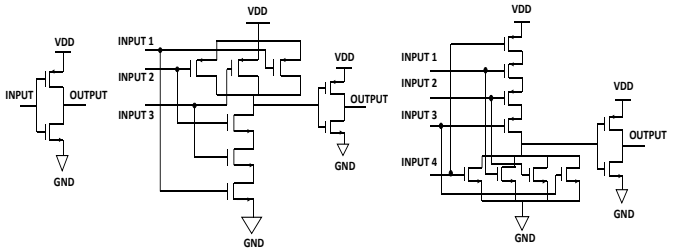
The 4×1 multiplexer is designed with two inverters, four AND gates and one OR gate as shown in Fig.5. The inputs to the AND gates are given with inputs I_0, I_1, I_2 and I_3 and selection lines S_0 and S_1. First AND gate is given with inputs I_0, S_0', and S_1'. Second AND gate is given with inputs I_1, S_1', and S_0. Third, AND gate is given with I_2, S_1, and S_0'. Fourth AND gate is given with I_3, S_0, and S_1. OR gate is given with output of AND gates and OR gate provides the Y output.



(a) (b)

Fig.5.(a) Gate level design of 4×1 multiplexer (b) CMOS representation of 4×1 multiplexer

A 4×1 multiplexer CMOS based is given in Fig.5(b) is designed by using CMOS inverter, CMOS AND gate and CMOS OR gate. Four 3-input CMOS AND gates, two CMOS inverters and one 4-input CMOS OR gate is used for 4×1 multiplexer. CMOS inverter attached at the output sections of the CMOS NAND gate gives a CMOS AND gate. CMOS NAND gate is formed by sequence connection of the 3 NMOS transistors and side by side connection of the 3 PMOS transistors. CMOS OR gate is designed with one CMOS NOR gate with a CMOS inverter connection at the output of the CMOS NOR gate this gives a CMOS OR gate. NOR gate of CMOS is formed by series connection of 4 PMOS transistors and parallel connection of the 4 NMOS transistors as shown in Fig.6. CMOS inverter is designed with one PMOS and one NMOS transistors.



(a) (b) (c)

Fig.6. (a) CMOS inverter (b) CMOS three input AND gate (c) CMOS four input OR gate

Table.2 is given with different input combinations with selection lines S_0 and S_1 and output Y. when S_1 is 0 and S_0 is 0, output (Y) is I_0. When S_1 is 0 and S_0 is 1, output (Y) is I_1. When S_1 is 1 and S_0 is 0, output is I_2. When S_1 is 1 and S_0 is 1, output is I_3.

Table.2. Truth Table of the 4×1 multiplexer

S_1	S_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

3.3 8×1 MULTIPLEXER

A multiplexer with 8 inputs, 3 selection lines and 1 output are an 8×1 multiplexer. As shown in Fig.7 inputs are I_0, I_1, I_2, I_3, I_4, I_5, I_6, and I_7 are the given to the multiplexer. Selection lines are S_0, S_1, and S_2 and output is the Y.

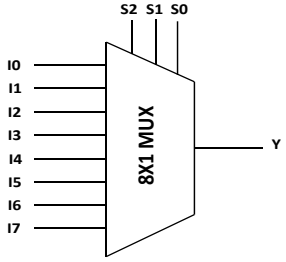


Fig.7. 8×1 multiplexer block diagram

The equation for an 8:1 multiplexer is Eq. (3)

$$Y=I_0S_2'S_1'S_0'+I_1S_2'S_1'S_0+I_2\,S_2'S_1S_0'+I_3\,S_2S_1'S_0+ I_4S_2S_1'S_0'+I_5\,S_2S_1'S_0+\,I_6\,S_2S_1S_0'+\,I_7\,S_2S_1S_0 \quad (3)$$

As shown in Fig.8, 8×1 multiplexer is designed with 8 AND gates, three inverters and one OR gate. First AND gate is given with inputs I_0, S_2', S_1', S_0' and output obtained is given as input to OR gate. Second AND gate is given with inputs I_1, S_2', S_1', S_0 and output obtained is given as input to OR gate. Third AND gate is given with inputs I_2, S_2', S_1, S_0' and output obtained is given as input to OR gate. Fourth AND gate is given with inputs I_3, S_2', S_1, S_0 and output obtained is given as input to OR gate. Fifth AND gate is given with inputs I_4, S_2, S_1', S_0' and output obtained is given as input to OR gate. Sixth AND gate is given with inputs I_5, S_2, S_1', S_2 and output obtained is given as input to OR gate. Seventh AND gate is given with inputs I_6, S_2, S_1, S_0' and output obtained is given as input to OR gate. Eight AND gate is given with inputs I_7, S_2, S_1, S_0 and output obtained is given as input to OR gate. The OR gate is given with input from the output of AND gates and gives output Y.

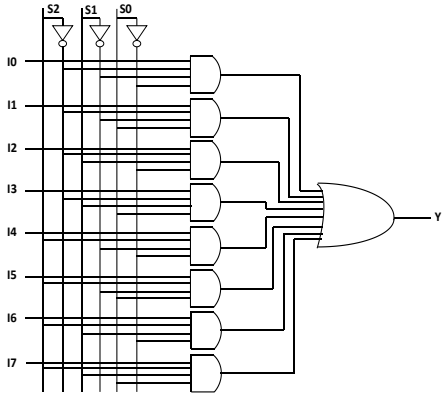
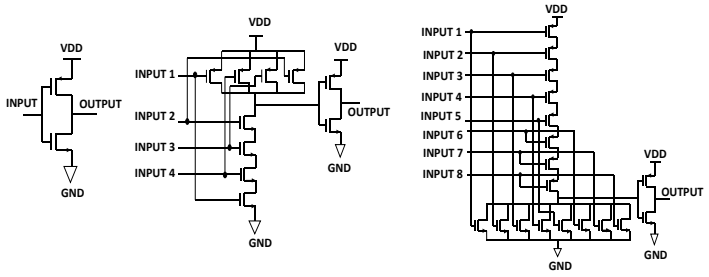


Fig.8. Gate level diagram of the 8x1 multiplexer

CMOS Inverter is designed with one PMOS and one NMOS transistors. CMOS AND gates are designed with one CMOS NAND gate with an CMOS inverter connection at the output of the CMOS NAND gate this gives an CMOS AND gate. CMOS NAND gate is formed by sequence connection of the 4 NMOS transistors and side by side connection of the 4 PMOS transistors. CMOS OR gate is designed with one CMOS NOR gate with a CMOS inverter connection at the output of the CMOS NOR gate this gives a CMOS OR gate. CMOS NOR gate is formed by series connection of 8 PMOS transistors and parallel connection of the 8 NMOS transistors as shown in Fig.9.



(a) (b) (c)

Fig.9. (a)CMOS inverter (b)CMOS four input AND gate (c)CMOS eight input OR gate

The operation for the 8×1 multiplexer is explained as followed by the Table.3 When the selection lines S_0 is 0, S_1 is 0, S_2 is 0, the output value of the I_0. When the selection lines S_0 is 0, S_1 is 0, S_2 is 1, the output value of the I_1. When the selection lines S_0 is 0, S_1 is 1, S_2 is 0, the output value of the I_2. When the selection lines S_0 is 0, S_1 is 1, S_2 is 1, the output value of the I_3. When the selection lines S_0 is 1, S_1 is 0, S_2 is 0, the output value of the I_4. When the selection lines S_0 is 1, S_1 is 0, S_2 is 1, the output value of the I_5. While the selection lines S_0 is 1, S_1 is 1, S_2 is 0, the output value of the I_6. While the selection lines S_0 is 1, S_1 is 1, S_2 is 1, the output value of the I_7.

Table.3. Truth Table of 8×1 multiplexer

S_2	S_1	S_0	Y
0	0	0	I_0
0	0	1	I_1
0	1	0	I_2
0	1	1	I_3
1	0	0	I_4
1	0	1	I_5
1	1	0	I_6
1	1	1	I_7

4. PROPOSED 16×1 MULTIPLEXER

The represented Block diagram is a MUX i.e., 16:1. In this we have given 16 digital logic inputs from LOGIC-0 to LOGIC-15, in the diagram it is represented in the form of I_0 to I_{15}. Here is the important terminal we have is ENABLE (En), the important point about enable is here we have to put always high that means Logic 1 for the enable. And beside that we have four selection lines S_0, S_1, S_2, S_3 through this selection lines we can select the data directly by giving different logic in digital format.

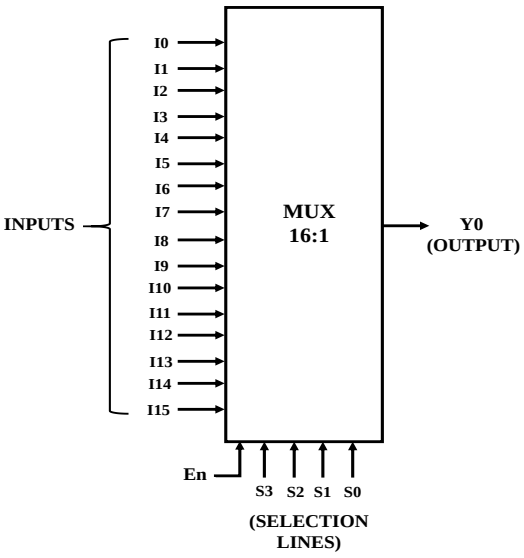


Fig.10. Block diagram of 16:1 MUX

The equation for a 16:1 multiplexer is Eq. (4)

$$Y=EnI_0S_0'S_1'S_2'S_3'+EnI_1S_0S_1'S_2'S_3'+EnI_2S_0'S_1S_2'S_3'+EnI_3S_0S_1S_2'S_3'+EnI_4S_0'S_1'S_2S_3'+EnI_5S_0S_1'S_2S_3'+EnI_6S_0'S_1S_2S_3'+EnI_7S_0S_1S_2S_3'+EnI_8S_0'S_1'S_2'S_3+EnI_9S_0S_1'S_2'S_3+EnI_{10}S_0'S_1S_2'S_3+EnI_{11}S_0S_1S_2'S_3+EnI_{12}S_0'S_1'S_2S_3+EnI_{13}S_0S_1'S_2S_3+EnI_{14}S_0'S_1S_2S_3+EnI_{15}S_0S_1S_2S_3 \quad (4)$$

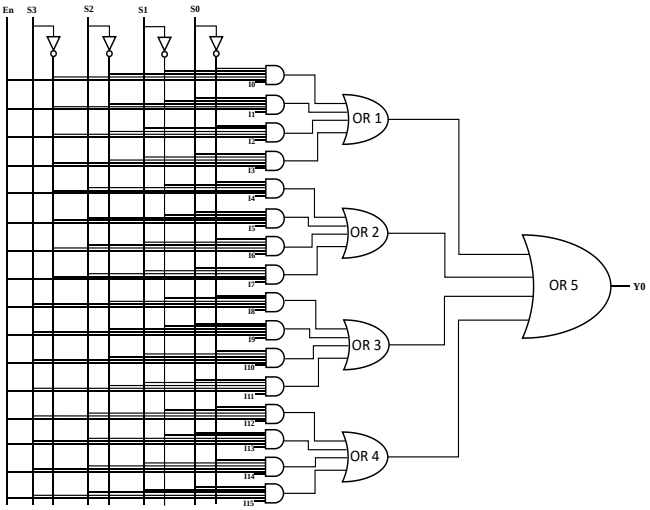


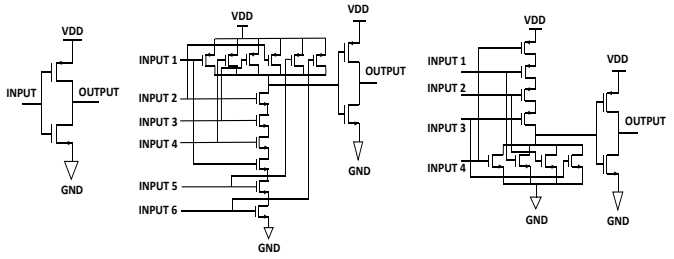
Fig.11.Gate level diagram of 16:1 Multiplexer

The diagram is designed with 4 inverters, 16 AND logic gates and 5 OR gates through which we are getting the output on the bases of selection lines, and another important input is enable and we have to give a logic 1 always to this Enable (*En*). AND gate as 6 inputs and one output, whereas OR gate as 4 inputs and one output. By observing Fig.11:

- The first AND gate inputs are *En*, I_0, S_0', S_1', S_2', S_3', the obtained output is input to the OR 1 gate.
- The second AND gate inputs are *En*, I_1, S_0, S_1', S_2', S_3', the obtained output is input to the OR 1 gate.
- The third AND gate inputs are *En*, I_2, S_0', S_1, S_2', S_3', the obtained output is input to the OR 1 gate.
- The fourth AND gate inputs are *En*, I_3, S_0, S_1, S_2', S_3', the obtained output is input to the OR 1 gate.
- The fifth AND gate inputs are *En*, I_4, S_0', S_1', S_2, S_3', the obtained output is input to the OR 2 gate.
- The sixth AND gate inputs are *En*, I_5, S_0, S_1', S_2, S_3', the obtained output is input to the OR 2 gate.
- The seventh AND gate inputs are *En*, I_6, S_0', S_1, S_2, S_3', the obtained output is input to the OR 2 gate.
- The eighth AND gate inputs are En, I_7, S_0, S_1, S_2, S_3', the obtained output is input to the OR 2 gate.
- The ninth AND gate inputs are *En*, I_8, S_0', S_1', S_2', S_3, the obtained output is input to the OR 3 gate.
- The tenth AND gate inputs are *En*, I_9, S_0, S_1', S_2', S_3, the obtained output is input to the OR 3 gate.
- The eleventh AND gate inputs are *En*, I_{10}, S_0', S_1, S_2', S_3, the obtained output is input to the OR 3 gate.
- The twelfth AND gate inputs are *En*, I_{11}, S_0, S_1, S_2', S_3, the obtained output is input to the OR 3 gate.
- The thirteenth AND gate inputs are *En*, I_{12}, S_0', S_1', S_2, S_3, the obtained output is input to the OR 4 gate.

- The fourteenth AND gate inputs are *En*, I_{13}, S_0, S_1', S_2, S_3, the obtained output is input to the OR 4 gate.
- The fifteenth AND gate inputs are *En*, I_{14}, S_0', S_1, S_2, S_3, the obtained output is input to the OR 4 gate.
- The sixteenth AND gate inputs are *En*, I_{15}, S_0, S_1, S_2, S_3, the obtained output is input to the OR 4 gate.

Outputs of all the 16 AND gates are given as input to 4 OR gates again the output of the 4 OR gates are given as input to 5th OR gate which provides the output Y_0.



(a) (b) (c)

Fig.12. (a) CMOS inverter (b) CMOS six input AND gate (c) CMOS four input OR gate

CMOS NAND gate is designed with 6 PMOS connected in side by side and 6 NMOS connected in sequence with 6 inputs and one output. By connecting CMOS inverter to the output of the NAND gate gives CMOS AND gate as shown in Fig.12(b). CMOS NOR gate is designed by using 4 PMOS transistors connected in sequence and 4 NMOS transistors connected side by side with 4 inputs and one output. By connecting CMOS inverter at the output of the NOR gate gives the CMOS OR gate as shown in Fig.12 (c). CMOS inverter is designed with one PMOS and one NMOS as shown in Fig.12(a). The connections of the CMOS AND gate, CMOS inverter, CMOS OR gate is done as shown in Fig.11 to design a 16×1 multiplexer. One of the points about the designed 16x1 MUX is, here we have used 282 transistors.

Table.4. Truth Table of 16:1 MUX

Input	**Enable (*En*)**	S_3	S_2	S_1	S_0	**Output (Y_0)**
I_0	1	0	0	0	0	I_0
I_1	1	0	0	0	1	I_1
I_2	1	0	0	1	0	I_2
I_3	1	0	0	1	1	I_3
I_4	1	0	1	0	0	I_4
I_5	1	0	1	0	1	I_5
I_6	1	0	1	1	0	I_6
I_7	1	0	1	1	1	I_7
I_8	1	1	0	0	0	I_8
I_9	1	1	0	0	1	I_9
I_{10}	1	1	0	1	0	I_{10}
I_{11}	1	1	0	1	1	I_{11}
I_{12}	1	1	1	0	0	I_{12}
I_{13}	1	1	1	0	1	I_{13}
I_{14}	1	1	1	1	0	I_{14}
I_{15}	1	1	1	1	1	I_{15}

The Table.4 represents the inputs as (I_0 to I_{15}) and a single output (Y_0), here we have given inputs as selection lines from S_0 to S_3 by different logic values (0,1) and taking enable (*En*) as always high i.e.(logic1), and by analysis we got single output and we can take it from the truth table Y0.

5. SIMULATIONS AND RESULTS

Simulations are performed on Synopsys tool HSPICE under 32 nm BSIM 4 model card for bulk CMOS technology of PTM model. The sizing of the transistor is taken as L_P=32 nm, W_P=3200 nm, L_N=32 nm, W_N=1600 nm where L_P=length of the PMOS, W_P=width of the PMOS, L_N=length of the NMOS and W_N=width of the NMOS. The Fig.13 represents HSPICE simulations results of the input and output of the 16:1 Mux. Basically, here we have taken input as I_0 to I_{15} and output we are getting through Y_0 depending on the selection lines logic that we have given.

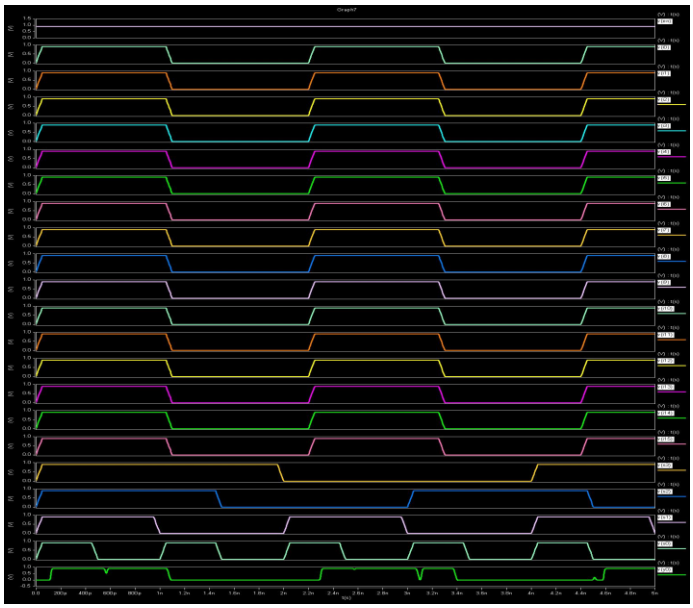


Fig.13. 16:1 multiplexer output waveform

- When the inputs S_3 is 1, S_2 is 1, S_1 is 1, S_0 is 1, output Y_0 is 1 same as given input I_{15}.
- When the inputs S_3 is 1, S_2 is 1, S_1 is 1, S_0 is 0, output Y_0 is 1 same as given input I_{14}.
- When the inputs S_3 is 1, S_2 is 1, S_1 is 0, S_0 is 1, output Y_0 is 0 same as given input I_{13}.
- When the inputs S_3 is 1, S_2 is 0, S_1 is 0, S_0 is 0, output Y_0 is 0 same as given input I_8.
- When the inputs S_3 is 0, S_2 is 0, S_1 is 1, S_0 is 1, output Y_0 is 0 same as given input I_3.
- When the inputs S_3 is 0, S_2 is 0, S_1 is 1, S_0 is 0, output Y_0 is 1 same as given input I_2.
- When the inputs S_3 is 0, S_2 is 1, S_1 is 0, S_0 is 1, output Y_0 is 1 same as given input I_5.
- When the inputs S_3 is 0, S_2 is 1, S_1 is 0, S_0 is 0, output Y_0 is 0 same as given input I_4.

- When the inputs S_3 is 1, S_2 is 1, S_1 is 1, S_0 is 1, output Y_0 is 0 same as given input $I_{15.}$
- When the inputs S_3 is 1, S_2 is 0, S_1 is 1, S_0 is 0, output Y_0 is 1 same as given input $I_{10.}$

The simulation waveform shown the exact output verified by using the Table.4 with different input combinations. Always Enable (*En*) is keep at logic high (1).

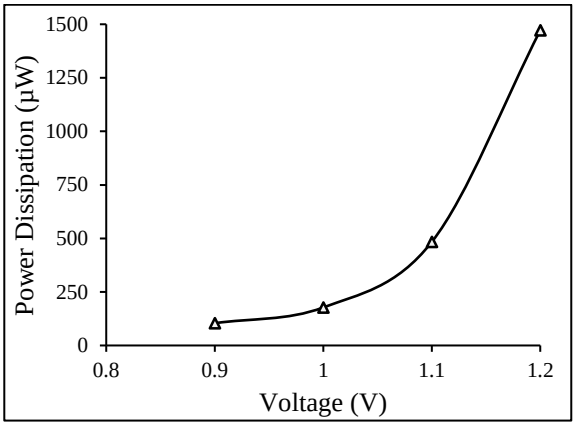


Fig.14. Power dissipation (µW) vs Voltage (V)

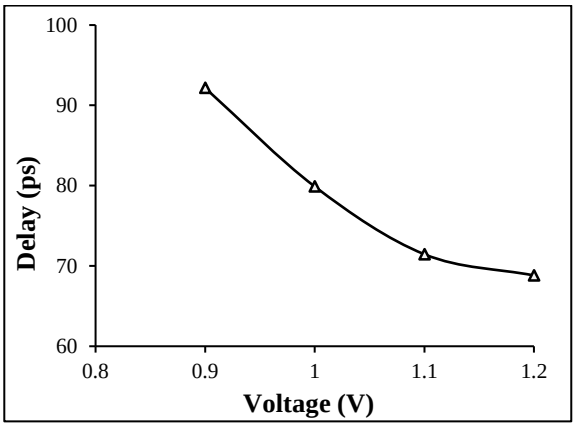


Fig.15. Delay (ps) vs Voltage (V)

By observing Fig.14 the variations in the power dissipation can be seen at different voltage for the 16x1 multiplexer and for Fig.15 the variations in the delay can be seen for different voltage for the 16×1 multiplexer. The arrow in the graph shows the changing of the power dissipation and delay for different voltage for the 16×1 multiplexer.

Table.5. Comparison table for the 16×1 multiplexer at different voltages

MUX Circuit	Transistor Count	Voltage (V)	Delay (ps)	Power Dissipation (µW)	PDP $\times 10^{-16}$ (J)
16:1	282	0.9	92.16	103.96	95.80
16:1	282	1	79.89	177.53	141.82
16:1	282	1.1	71.46	483.32	345.38
16:1	282	1.2	68.82	1471.4	1012.61

The Table.5 representation shows the different comparisons of 16:1 Mux with different power dissipation (µW), delay (ps) and power delay product (J) values. Total number of the transistors used is 282 for the 16x1 multiplexer. The minimum delay is 68.82 ps at 1.2 V, maximum delay is 92.16 ps at 0.9 V, minimum power dissipation is 103.96 µW at 0.9 V and maximum power dissipation is 1471.4 µW at 1.2 V for the 16x1 multiplexer given in the Table.5.

6. CONCLUSION

This analysis is made to propose the architecture of 16:1 multiplexer for the optimization of power and delay values. 16×1 multiplexer is designed with lesser transistors and also to decrease the power dissipation and delay. CMOS logic is the well one to introduce 16:1 multiplexer. The above given table representation shows the comparison of transistors count used and also the power dissipation and delay for the four different voltage levels. Furthermore, the delay and power dissipation analysis also achieved very less delay and power dissipation. The minimum and maximum delay and power dissipation for Synopsys tool HSPICE under 32 nm BSIM 4 model card for bulk CMOS technology of the PTM model simulation results are 68.82 ps, 92.16 ps and 103.96 µW, 1471.4 µW respectively. Hence, it is concluded that the proposed model power dissipation and delay of CMOS 16:1 multiplexer logic is analysed. Future scope of the work is to design 32×1 multiplexer.

REFERENCES

[1] Anjum Aara and Sushant Jain, "Design and Implementation of CMOS and CNT based 2:1 Multiplexer at 32nm Technology", *International Research Journal of Engineering and Technology*, Vol. 6, No. 8, pp. 548-552, 2019.

[2] Tripti Dua and Anju Rajput, "2:1 Multiplexer using Different Design Styles: Comparative Analysis", *STM Journal of Advancements in Robotics*, Vol. 7, No. 3, pp. 5-13, 2020.

[3] N. Sreenivasa Rao, Y. Vishnuvardhan Reddy, G. Shivamanikanta and B. Vijaysree, "Design the 2x1 Mux with 2T Logic and Comparing the Power Dissipation and Area with Different Logics", *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, Vol. 4, No. 3, pp. 1284-1290, 2015

[4] V. Vaishali, S. Raja Rajeshwari and C. Saravana Kumar, "A Study on Low Power Implementation of Multiplexer", *International journal of Emerging Technology in Computer Science and Electronics*, Vol. 25, No. 5, pp. 316-319, 2018.

[5] Aishwarya Tambe, Snehal Bhakre and Sankit Karsa, "Design and Analysis of (2x1) and (4x1) Multiplexer Circuit in Quantum Dot Cellular Automata Approach", *International Journal of Innovative Technology and Exploring Engineering*, Vol. 8, No. 6, pp. 277-281, 2019.

[6] Rajinder Tiwari, Anil Kumar and Preeta Sharan, "Design and Implementation of 4:1 Multiplexer for Reversible ALU using QCA", *Proceedings of International Conference on Micro-Electronics and Telecommunication Engineering*, pp 191-196, 2018.

[7] Debika Chaudhuri, Atanu Nag, Sukanta Bose, Suchismita Mitra and Hemanta Ghosh, "Power and Delay Analysis of a 4 to 1 Multiplexer Implemented in Different Logic Style",

International Journal of Innovative Research in Science, Engineering and Technology, Vol. 4, No. 9, pp. 118-123, 2015.

[8] Yalla Haresh and Yelithoti Sravana Kumar, "Design and Operation of 4x1 Lower Power Multiplexer using Different Logics", *International Journal of Emerging Science and Engineering*, Vol. 4, No. 8, pp. 36-39, 2017.

[9] Vikash Kumar Sharma and Shyam Akashe, "Power Efficient Design of 4:1 Multiplexer using Low Power Techniques", *International Journal of Modern Engineering and Management Research*, Vol. 4, No. 4, pp. 44-49, 2016.

[10] Li Hua Jiang and Chuan Huang, "Design and Realization of Multiplexer Based on Schematic and VHDL Language", *Proceedings of IEEE International Conference on Optics, Photonics and Energy Engineering*, pp. 273-276, 2010.

[11] Guvvapotula Sreekanth and B. Jeevan Rao, "Design of 8x1 Low Power Multiplexer by using Transmission Gates", *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, Vol. 8, No. 11, pp. 2465-2470, 2019.

[12] Abhishek Dixit, Saurabh Khandelwal and Shyam Akashe, "Design Low Power High Performance 8:1 Mux using Transmission Gate Logic (TGL)", *International Journal of Modern Engineering and Management Research*, Vol 2, No. 2, pp. 14-20, 2014.

[13] Deepak Mittal and Amit Niranjan, "Designing of Multiplexer and de-Multiplexer using different Adiabatic Logic in 90nm Technology", *Proceedings of International Conference on Computing, Communication and Networking Technologies*, pp. 1-9, 2018.

[14] Ankit Dalal and Ankur Atri, "A General Overview of Multiplexer and Demultiplexer", *International Journal of Research*, Vol. 1, No. 10, pp. 1293-1298, 2014.

[15] M. Morris Mano and Michael D. Ciletti, "*Digital Design*", 5th Edition, Pearson Education, 2011.