# ALU Implementation Using CMOS Technology

Sharana Basavaraj B[1], Renuka S[2], Kavitha K[3], Krishna P[4]
*Electronics and Communication, RYMEC, Cantonment Ballari, Karnataka, India*

***Abstract*—In designing ALUs, many techniques have been followed. The functional units of an ALU has been realized using conventional transistors and pass transistor gates. In this paper, the design of a 2 bit ALU is done using CMOS technology. The ALU can perform arithmetic operations and logical operations. The design is implemented using 3:8 decoder to select any one of the 8 operations. Designing an ALU which performs addition, subtraction, Logical operations like and, nand , or,nor, xor & xnor and shift operations like left shift and right shift and 2bit multiplier operation using CMOS technology.**

***Keywords*—CMOS, ALU, AND, XOR, 3:8 DECODER, NAND, NOR**

## I. INTRODUCTION

CPU is the heart of computer and ALU is the core of CPU. Design of a simple ALU is done using CMOS technology. CMOS technology has both N logic and P logic. N logic is connected to ground (pull down network) and P logic is connected to Vdd and acts like pull up network. Increase in electric field that lead to degradation of device performance . An enhancement in the basic function of a transistor, allowed for designs to be implemented using fewer transistors and reduced interconnections.

### *A. DESIGN OF OVERVIEW*

In this paper CMOS technology, both N-type and P-type transistors are used to realize logic functions. The N type and P type transistor are used for turns on and turn off a transistor. The design used simple switches. In this paper CMOS logic gates a collection of n-type MOSFETs is arranged in a pull down network (Vss or ground). In place of load resistor of NMOS logic gates, CMOS logic gates have a collection of p-type MOSFETs in a pull-up network , higher voltage rail (often named Vdd)[1]. Thus, if both a p-type and n-type transistor have their gates connected to the input, the p-type MOSFET will be on when the n-type MOSFET is off, and vice-versa as shown in the fig. 1.1.

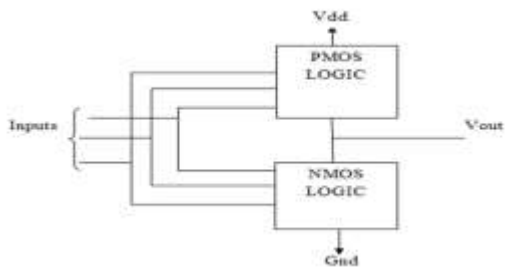


Fig1.1: Block Diagram of CMOS Logic

### *B. ALU IMPLEMENTATION*

ALU has been designed for operation in which, the full adder design has been implemented using MIFG CMOS inverters. The ALU performs the following three arithmetic operations, ADD, SUBTRACT, MULTIPLIER. The four logical operations are performed, EXOR, EXNOR, AND and OR. A set of three input signals to decoder has been incorporated in the design to determine the operation being performed and the inputs and outputs being selected .The block diagram as shown in fig.1.2.

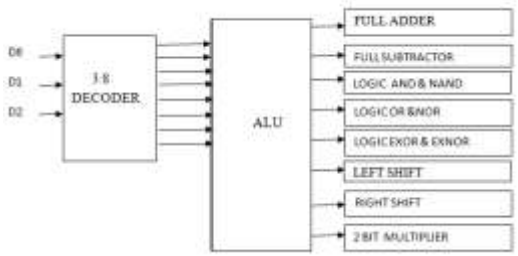


Fig 1.2: Block Diagram of ALU

The following blocks are used in ALU block:

1. **3:8 Decoder:** For selecting any particular operation in an ALU. Totally 8 functionalities can be implemented.
2. **ALU:** A typical ALU should perform addition, Subtraction, multiplication, shift operations and logical operations.

The following 8 functionalities are implemented using ALU:

- Full Adder
- Full Subtractor
- Right shifter.
- Left Shifter.
- Logical AND & NAND.
- Logical OR & NOR.
- Logical EXOR & EXNOR
- 8.2 Bit Multiplier

*Software used:*

Export Dsch for drawing & verifying CMOS Schematic and Microwind for drawing & verifying CMOS schematic [3].

*C. Logic Symbol, CMOS Schematic and symbol diagram of inverter:*

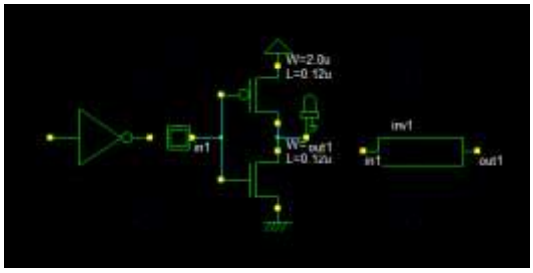


Fig.1.3 Logic Symbol, CMOS Schematic and symbol diagram of inverter

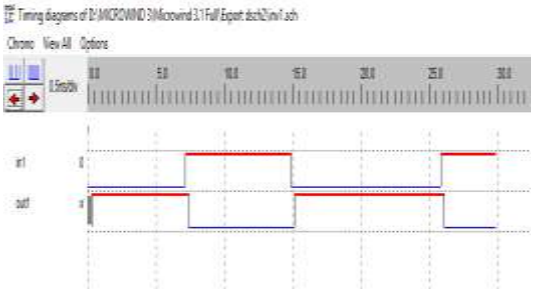

Fig.1.4 timing diagram of CMOS inverter

Table 1. truth table of inverter

| Input | Output |
|---|---|
| 0 | 1 |
| 1 | 0 |

Fig.1.3shows the schematic of INVERTER**.**NOT is the most simple logic gate. which takes the input that is either ON or OFF and it invert the opposite. For a 0 it will give a 1, and for a 1it will give a o. Another name for a NOT gate is inverter, because it inverts the input.The result shown in fig.1.4 timing diagram of CMOS inverter[2].

*D.CMOS Schematic of 3:8 decoder :*

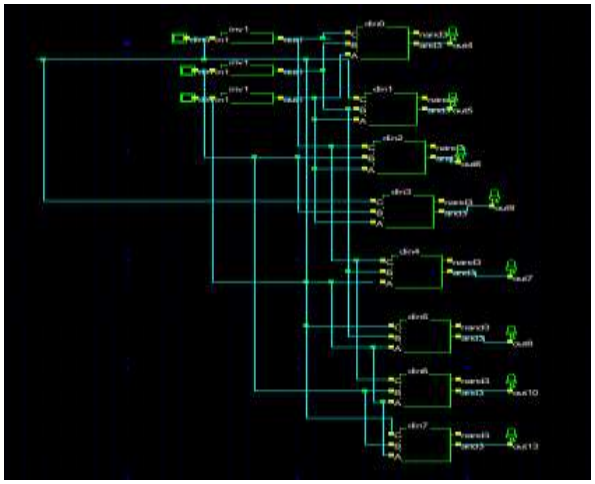

Fig.1.6CMOS schematic of 3:8 decoder

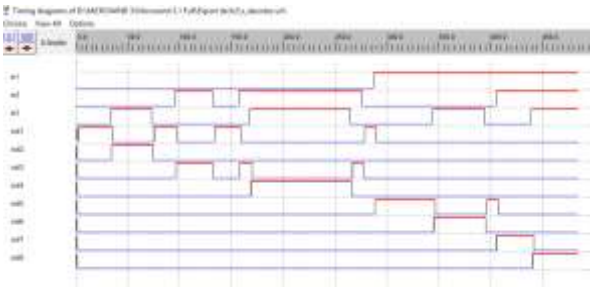

Fig.1.7 timing diagram of 3:8 DECODER

The fig.1.5 shows three to eight decoder, It has three inputs and eight outputs. A0 is the least significant bit variable, while A2 is the most significant bit variable. The three inputs are decoded and produces eight outputs. That is, binary vALUes at the input form a combination, and based on this combination, the selected output line is activated. The result shown in fig.1.6 timing diagram of 3:8 DECODER.

*E. SCHEMATIC OF .FULL ADDER*

Full Adder is a combinational logic circuit that performs the addition of three bits (two significant bits and previous carry).

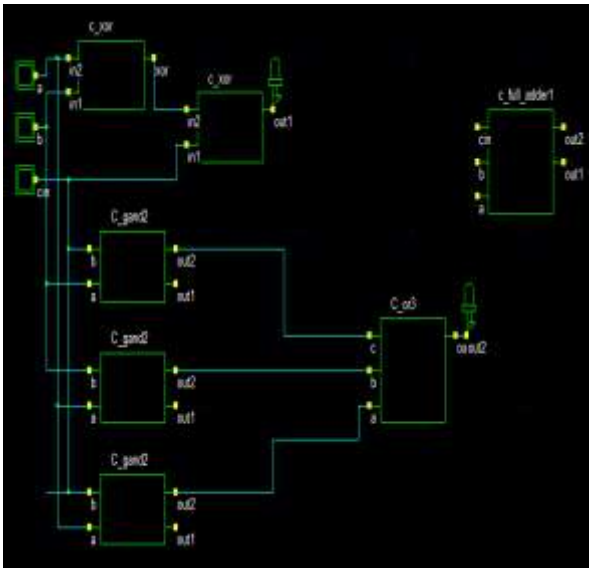


Fig.1.8 CMOS Schematic of Full Adder

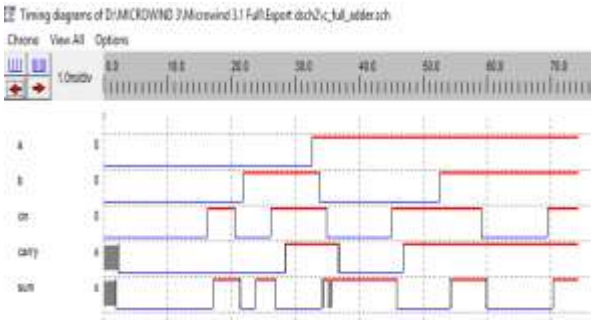

Fig.1.9timing diagram of FULL ADDER

| A | B | Carry in | Sum | Carry out |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

Table 2.truth table of full adder

The fig.1.8 shows the schematic of full adder. The full adder consists of three inputs and two outputs, two inputs are the bits to be added, the third input represents the carry bit form the previous position. The full adder is usually a component in a cascade of adders, which add 8 bits, 16 bits, etc, binary numbers .The output is equal to 1 when only one input is equal to 1 or when all three inputs are equal to 1 .The output has a carry bit is 1 if two or three inputs are equal to 1 . The result shown in fig.1.9 timing diagram of FULL ADDER.

*F. Implementation of ALU:*

Fig.1.8 schematic of ALU

| A | B | C | Decoder_output | ALU_block_selected |
|---|---|---|---|---|
| 0 | 0 | 0 | Y0 | Full subtractor |
| 0 | 0 | 1 | Y1 | Full adder |
| 0 | 1 | 0 | Y2 | 3 input and gate |
| 0 | 1 | 1 | Y3 | 3 input or gate |
| 1 | 0 | 0 | Y4 | 3 input xor gate |
| 1 | 0 | 1 | Y5 | Left shift |
| 1 | 1 | 0 | Y6 | Right shift |
| 1 | 1 | 1 | Y7 | 2 bit Multiplier |

Table 3. truth table of ALU

The fIg.1.8 shows the schematic of ALU ,it consist of 8 operations as shown in the table.5.1 .. An ALU loads data from input registers, executes the operation and stores the result into output registers. The design and function of an ALU may vary between different processors. For example, some arithmetic and logical units only perform integer calculations, The ALU is designed, its primary job is to handle integer operations. Therefore, a computer's integer performance is tied directly to processing speed of ALU . ALUs can performs Integer arithmetic operations (addition, subtraction and multiplication).Bitwise logic operations (AND,NAND, OR,NOR XOR,XNOR).Bit-shifting operations (shifting a word by a specified number of bits to the left or right).

## II. CONCLUSION

In this paper, the design and analysis of Arithmetic and Logical Unit (ALU) circuit designed using CMOS technology as well as speed performances, but the fundamental aspects of many designs must remain the same ,using the 3:8 decoder to select the 8 operations and make use of pass transistor. The NMOS pass transistor is used for enabling the corresponding functionality depending on the decoder inputs.

## REFERENCES

[1]. Arvind Kumar ,Dept. of Electronics & Communication, UIET, Punjab University, Chandigarh, UT, India march 2012.
[2]. Douglas A. Pucknell & Kamran Eshraghian, **"Basic VLSI Design"** PHI 3rd Edition (original Edition – 1994), 2005.
[3]. MICROWIND & DSCH V3.1 - LITE USER'S MANUAL by Etienne Sicard,august 2006.