RESEARCHERID
THOMSON REUTERS

[Gupta, 5(7): July 2018]
DOI: 10.5281/zenodo.1325039

ISSN 2348 – 8034
Impact Factor- 4.022

GLOBAL JOURNAL OF ENGINEERING SCIENCE AND RESEARCHES

PHYSICAL DESIGN, LAYOUT AND SIMULATION USING C5 PROCESS TECHNOLOGY OF 8 BIT ARITHMETIC AND LOGIC UNIT

Aparna Gupta[*1], Dr. Rita Jain[2] & Dr. R. P. Singh[3]

[*1]Research Scholar, Department of Electronics and Communication Engineering, AISECT University, Bhopal, India

[2]Prof. and Head, Department of Electronics and Communication Engineering, Lakshmi Narain College of Technology, Bhopal, India

[3]Vice Chancellor, Sri Satya Sai University, Bhopal, India

ABSTRACT

A critical component of the microprocessor, the core component of central processing unit, Arithmetic and Logical Unit (ALU) comprises of the combinational logic that implements logic operations such as AND and OR, and arithmetic operations such as addition, subtraction, and multiplication. In this proposed work, a 8-bit ALU is designed, implemented and simulated using the Electric CAD and SPICE software. The proposed design is a 8-bit ALU that can perform operations like: A AND B, A OR B, A + B (addition), and A - B (subtraction) and all possible arithmetic and logical operations. Physical design of every sub module is carried out using C5 process 300 nm process technology. ALUs can be built in so many ways with wide specifications and since the objective of the proposed work is to learn the basics of VLSI design mainly digital VLSI Design, the specifications of the ALU were relaxed.

Key words: *ALU, CPU, 8 bit, VLSI CAD, Simulation, C5 process, SPICE*

I. INTRODUCTION

Digital design is an amazing and very broad field. The applications of digital design are present in our daily life, including computers, calculators, video cameras etc. In fact, there will always be need for high speed and low power digital products which make digital design a future growing business. Arithmetic Logic Unit (ALU) is a critical component of a microprocessor and is the core component of Central Processing Unit (CPU)[2]. Furthermore, it is the heart of the instruction execution part of every computer. ALU consist of the combinational logic that implements logic operations, such as AND, OR and arithmetic operations, such as ADD and SUBTRACT[3]. The designed ALU comprises of NAND, NOR, AND, OR, NOT gates. The designed system would provide output to respective inputs for the required operation.

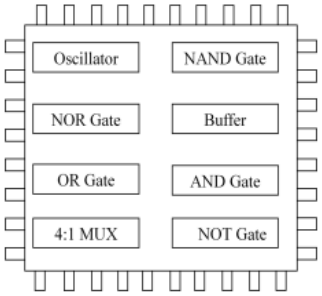


Fig.1. Pad frame for integrated ALU with sub circuit topologies

The sub modules are placed on a single pad frame to provide the desired system as shown in the fig. 1. The design of the layouts is done with the help of Electric CAD and the simulation of those designs is performed by LTSpice IV.

[Gupta, 5(7): July 2018]
DOI: 10.5281/zenodo.1325039
ISSN 2348 – 8034
Impact Factor- 4.022

II. CIRCUIT DESIGN

Because ALUs can be built in so many ways with wide specifications and since the objective of the design is to learn the basic of VLSI chip design, the specifications of the ALU were relaxed. The main objective of the project is to have a working ALU that performs different arithmetic and logic functions for all possible combinations of the inputs[4]. The speed of ALU was not an issue and we wanted it to run at low power. The high level circuit diagram of the four-bit ALU is as shown in fig 2.

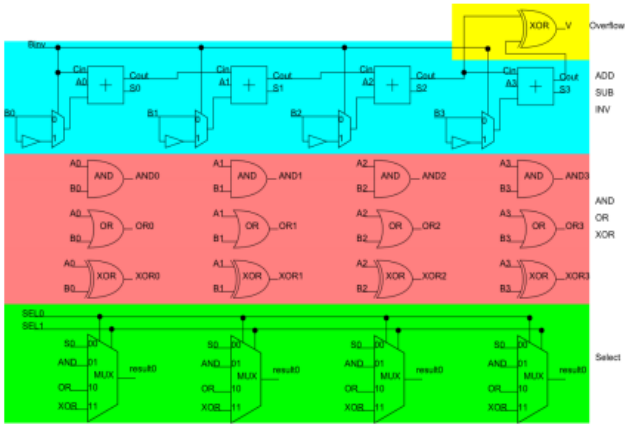


Fig. 2. High level circuit diagram of the four-bit ALU

As we can see in fig. 2, the diagram can be roughly divided into four sections. These are arithmetic section (blue), overflow section (yellow), logical section (red) and selection section (green). The arithmetic section is a chain of full adders. It is responsible for addition (ADD), subtraction (SUB) and invert (INV) operations. The ADD operation requires Binv set to low, such that it functions like a ripple adder. For SUB operation, it uses 2's compliment and thus, A-B becomes A + (-B). The usage of Binv has two folds. Firstly, when Binv is high, B will be inverted (through the inverter and MUX) before going into the full adder. Secondly, Binv also serves as the initial carry-in for 2's complement. To perform bitwise INV, input A needs to be set to zero and Binv set to high to invert input B. As a result, B first got invert to -B and the operation becomes 0 + (-B) = -B. The yellow section reports if there's any overflow occurs throughout an operation. Overflow happens when adding two positive numbers but the result is negative (the most significant bit is 1). For instance, 0101 + 0011 = 1000, the resulting value 1000 is negative in 2's compliment, so overflow has occurred. We want to set the overflow bit (V) to high when such situation happens. The carry out from last two full adders can be XOR together. The result would be the overflow bit. The red section performs bitwise logical operations, including AND, OR and XOR. Since each gate handles only one bit, in order to handle four-bit inputs, we need to place four gates of the same kind in parallel. The green section is select section, which determines which operation results go to the output. There are two bit select line (SEL0, SEL1) that select different operations.

The ALU proposed will implement the AND, OR, Addition, and Subtraction functions for the 8-bit A and B input buses and the result will be output to the 8-bit bus.

ISSN 2348 – 8034
Impact Factor- 4.022

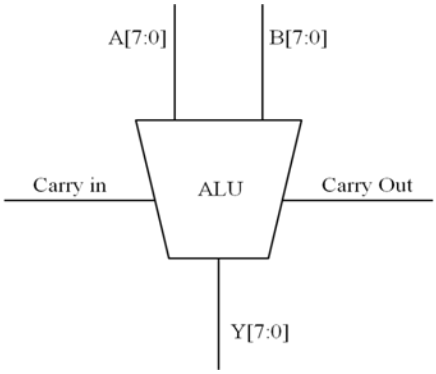


Fig. 3. Block Diagram of 8 bit ALU

III. DESIGN METHODOLOGY

Defining the requirements and setting the specifications is an important aspect in any VLSI Design. Design of the proposed 8 bit ALU will be according to the Tool flow (EDA based). The proposed methodology will start with the design of the test circuits like basic gates, for example NOT Gate, NAND gate, NOR gate etc followed by simulating the test results and optimization of transient and DC characteristics for the sub circuits. A pad frame is also proposed that will accommodate all ALU sub circuits and will be approximately 40 – 60 pins. The process technology used here is C5 process provided by MOSIS.

The technology used in Electric CAD is C5 process, 300 nm and is also used for fabrication is with respect to MOSIS design rules. This process has two layers of polysilicon to make a poly1 - poly2 capacitor, 3 layers of metal, and a hi-res layer to Design of Digital ALU block the implant, and thus decrease in resistance, of poly2 to fabricate higher-value (than what we would get with poly1- poly2 resistors[15]. Further, this process uses the MOSIS Scalable CMOS (SCMOS) submicron design rules. The system in whole is designed in various steps. Firstly, the basic sub modules of the system are designed. The design procedure of these sub modules involves obtaining the Boolean expression for the operation to be performed. When the Boolean expression is obtained, the CMOS schematic is prepared for the same. The sources and drains are marked for the PMOS and NMOS.

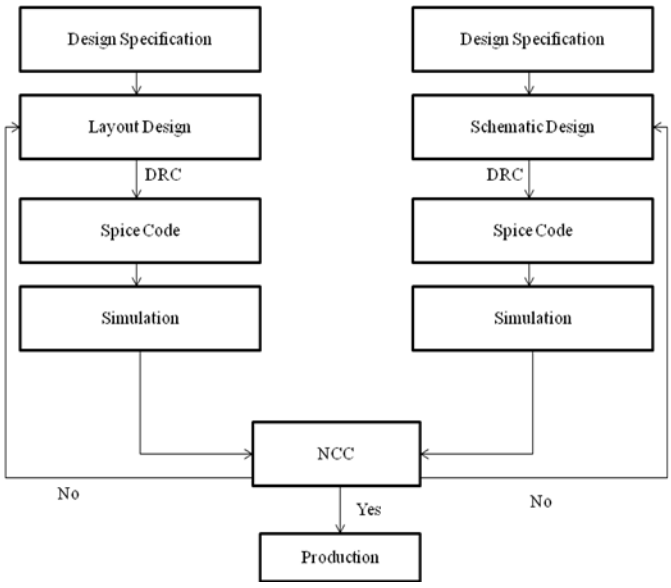


Fig.4. Design using Parallel Approach

ISSN 2348 – 8034
Impact Factor- 4.022

The step following the schematic design is the layout design. Layout design is very important part of the pad frame digital design. The design uses Euler's Rule for finding the best arrangement of input gates and obtaining minimum number of interconnects.

Following is the flow graph that illustrates the design of mixed analog integrated circuits.

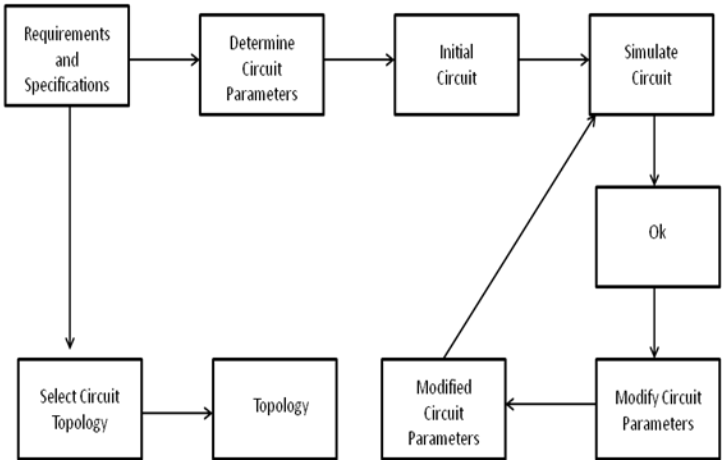


Fig.5. Flow graph for design of 8 Bit ALU

The design of CMOS combinational circuits starts with the very basic design of NMOS and PMOS. The active regions for the MOS, the MOS itself and the wells are positioned based on the design rules and requirements. The design rule checks need to be applied after every step so that errors, if occur, can be removed.

IV. DESIGN – STEP BY STEP

The system in whole is designed in various steps. The technology used in Electric is C5 process, 300 nm and is also used for fabrication is with respect to MOSIS design rules[1]. Firstly, the basic submodules of the system are designed. The design procedure of these submodules involves obtaining the Boolean expression for the operation to be performed. When the Boolean expression is obtained, the CMOS schematic is prepared for the same. The sources and drains are marked for the PMOS and NMOS. The step following the schematic design is the layout design. Layout design is very important part of the digital design. The design uses Euler's Rule for finding the best arrangement of input gates and obtaining minimum number of interconnects. The design of CMOS combinational circuits starts with the very basic design of NMOS and PMOS [15]. The active regions for the MOS, the MOS itself and the wells are positioned based on the design rules and requirements. The design rule checks need to be applied after every step so that errors, if occur, can be removed. When all the sub modules have been designed, they need to be placed on a single pad frame which shall be fabricated to provide us with an integrated chip. The designed ALU can perform an operation and give its output for respective inputs.

Each and every component has an operation of its own. Its working can be checked by writing spice code for the design of the component. Spice codes are commands given to the spice simulator for simulating the circuit designed.

V. SIMULATION RESULTS AND DESIGN REPORTS

The project aims to design the combinational circuits and simulate the designed circuits. The designed schematic views and layout views along with the simulation results are shown below.

ISSN 2348 – 8034
Impact Factor- 4.022

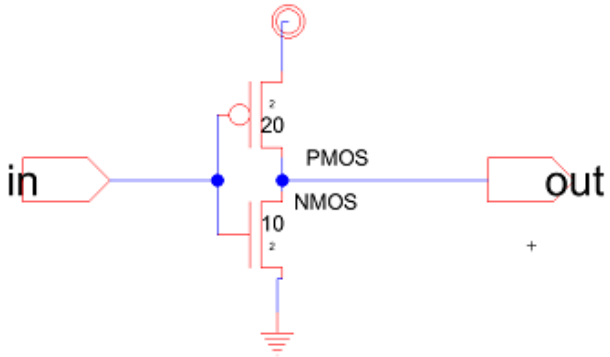


Fig.6. Schematic design for inverter (NOT gate)

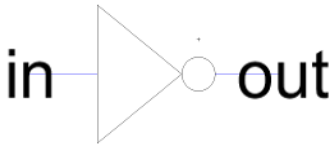


Fig.7. Icon view for inverter (NOT gate)

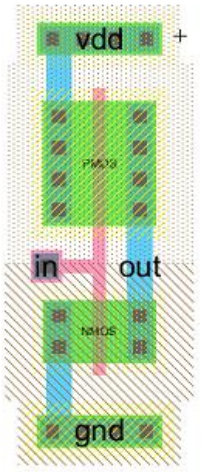
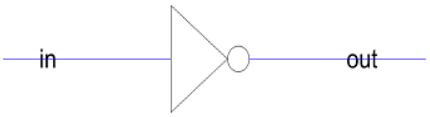


Fig.8. Layout design for inverter (NOT gate)

```
Checking schematic cell 'Inv_20_10{sch}'
   No errors found
0 errors and 0 warnings found (took 0.01 secs)
============================1============================
0 errors and 0 warnings found (took 0.0 secs)
============================2============================
Running DRC with area bit on, extension bit on, Mosis bit
Checking again hierarchy .... (0.0 secs)
Found 7 networks
0 errors and 0 warnings found (took 0.03 secs)
============================3============================
Hierarchical NCC every cell in the design: cell 'Inv_20_10{sch}'  cell 'Inv_20_10{lay}'
Comparing: 2npmos_1V:Inv_20_10{sch} with: 2npmos_1V:Inv_20_10{lay}
  exports match, topologies match, sizes not checked in 0.05 seconds.
Summary for all cells: exports match, topologies match, sizes not checked
NCC command completed in: 0.06 seconds.
```

Fig.9. Results for DRC and NCC for inverter (NOT Gate)

ISSN 2348 – 8034
Impact Factor- 4.022




```
vdd vdd 0 DC 5
vin in 0 DC 0
.dc vin 0 5 1m
.include C:\Electric\C5_models.txt
```

Fig.10.Spice simulation for DC analysis for inverter (NOT gate)

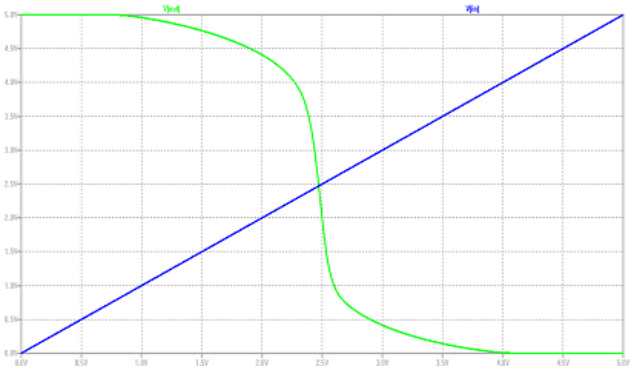


Fig.11. DC Analysis of CMOS Inverter

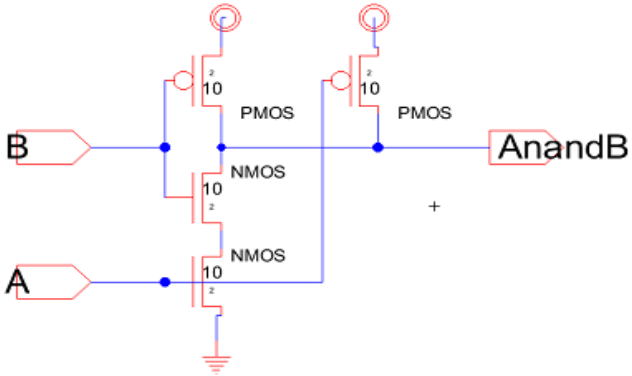


Fig.12. Schematic design of NAND gate

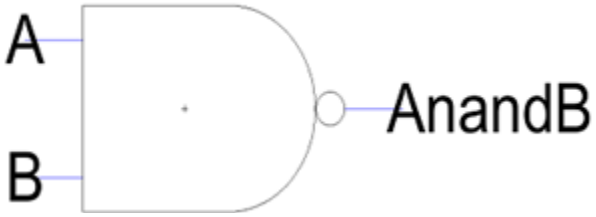


Fig.13. Icon view for NAND gate

[Gupta, 5(7): July 2018]
DOI: 10.5281/zenodo.1325039

ISSN 2348 – 8034
Impact Factor- 4.022

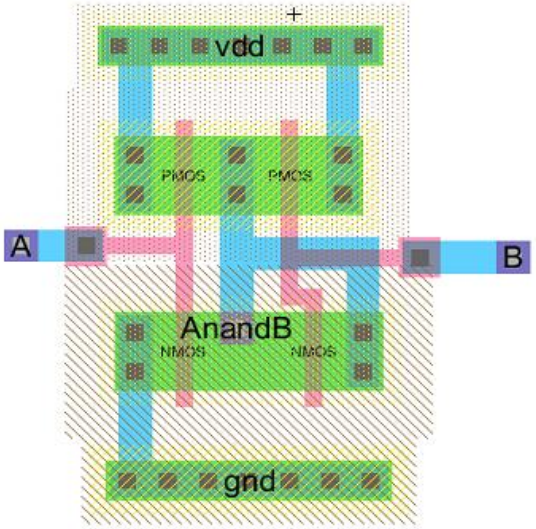


Fig.14.Layout design for NAND gate

```
Checking schematic cell 'NAND_2{sch}'
   No errors found
0 errors and 0 warnings found (took 0.0 secs)
============================]Te============================
Running DRC with area bit on, extension bit on, Mosis bit
Checking again hierarchy .... (0.0 secs)
Found 11 networks
0 errors and 0 warnings found (took 0.01 secs)
============================]E============================
Hierarchical NCC every cell in the design: cell 'NAND_2{sch}'  cell 'NAND_2{lay}'
Comparing: 2npmos_IV:NAND_2{sch} with: 2npmos_IV:NAND_2{lay}
  exports match, topologies match, sizes not checked in 0.05 seconds.
Summary for all cells: exports match, topologies match, sizes not checked
NCC command completed in: 0.05 seconds.
```

Fig.15.Results of DRC and NCC for NAND gate design

Fig.16. Spice simulation for DC analysis for NAND gate

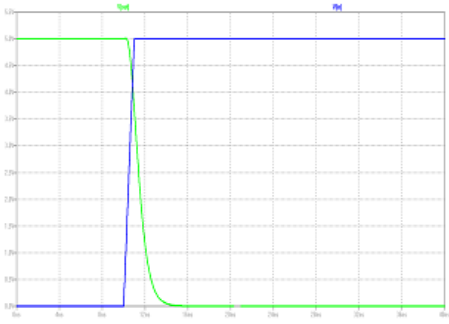


Fig.17. Simulation results for NAND gate design

ISSN 2348 – 8034
Impact Factor- 4.022

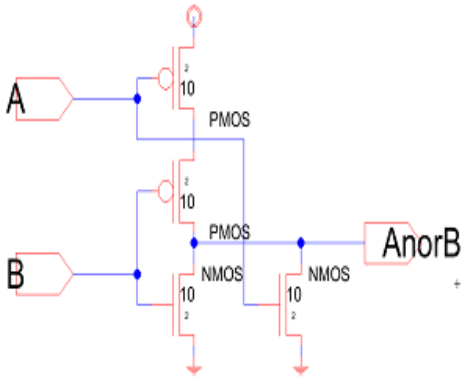


Fig.18.Schematic design of NOR gate

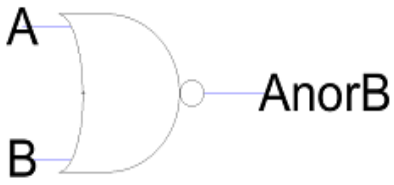


Fig.19. Icon view of NOR gate

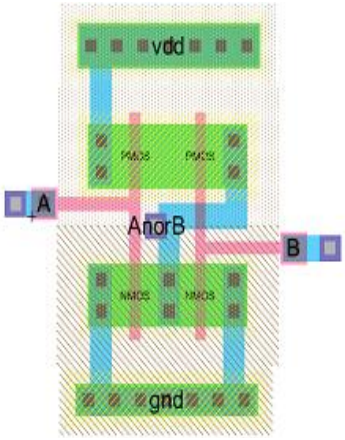


Fig.20. Layout design for NOR gate

```
Checking schematic cell 'NOR{sch}'
   No errors found
0 errors and 0 warnings found (took 0.01 secs)
=================================27=================================
Running DRC with area bit on, extension bit on, Mosis bit
Checking again hierarchy .... (0.0 secs)
Found 11 networks
Checking cell 'NOR{lay}'
        No errors/warnings found
0 errors and 0 warnings found (took 0.27 secs)
=================================28=================================
Hierarchical NCC every cell in the design: cell 'NOR{sch}'  cell 'NOR{lay}'
Comparing: 2npmos_IV:NOR{sch} with: 2npmos_IV:NOR{lay}
  exports match, topologies match, sizes not checked in 0.01 seconds.
Summary for all cells: exports match, topologies match, sizes not checked
NCC command completed in: 0.01 seconds.
```

Fig.21. Results of DRC and NCC for NOR gate design

ISSN 2348 – 8034
Impact Factor- 4.022

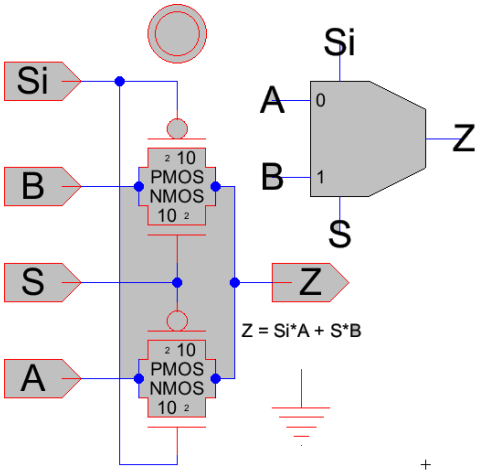


Fig.22. Schematic design for 2x1 Multiplexer

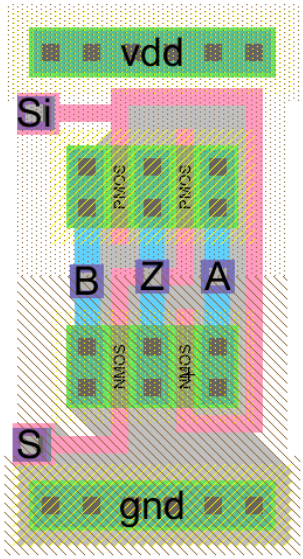


Fig.23. Layout design for2x1 Multiplexer

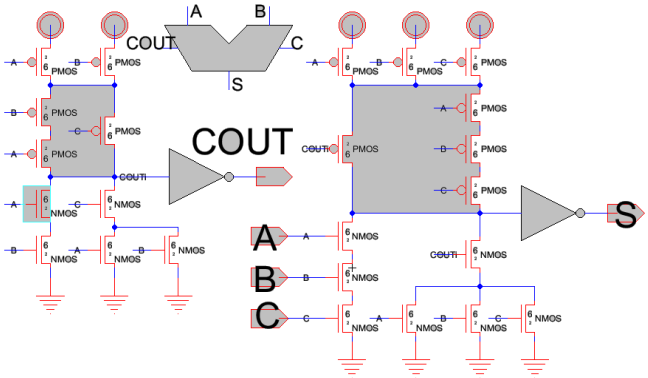


Fig.24. Schematic Design for Full Adder

[Gupta, 5(7): July 2018]
DOI: 10.5281/zenodo.1325039

ISSN 2348 – 8034
Impact Factor- 4.022

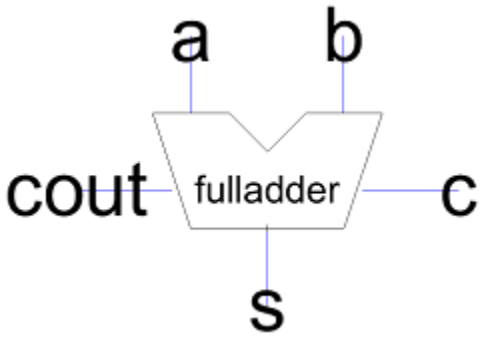


Fig.25. Icon View for Full Adder

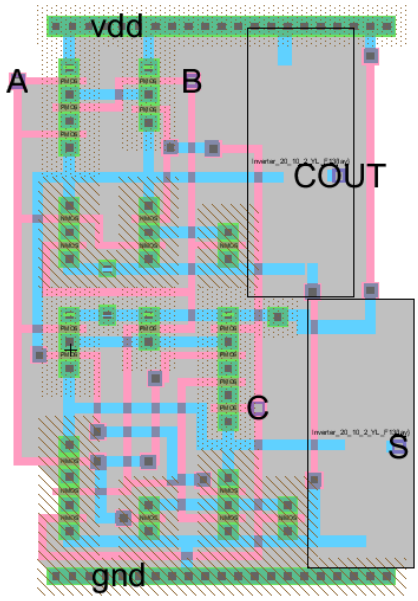


Fig.26. Layout design for Full Adder

```
Checking schematic cell 'mosbLib07:fulladder{sch}'
  No errors found
0 errors and 0 warnings found (took 0.017 secs)
================================================================================
Running DRC with area bit on, extension bit on, Manis bit
Checking again hierarchy .... (0.000 secs)
Found 52 networks
Checking cell 'mosbLib07:fulladder{lay}'
  No errors found
0 errors and 0 warnings found (took 0.067 secs)
================================================================================
Hierarchical NCC every cell in the design: cell 'mosbLib07:fulladder{sch}'  cell 'mosbLib07:fulladder{lay}'
Comparing: mosbLib07:fulladder{sch} with: mosbLib07:fulladder{lay}
  exports match, topologies match, sizes not checked in 0.000 seconds.
Summary for all cells: exports match, topologies match, sizes not checked
NCC command completed in: 0.067 seconds.
```

Fig 27.Results of DRC and NCC for Full Adder

ISSN 2348 – 8034
Impact Factor- 4.022

VI. CONCLUSIONS

The schematics and layouts for the components of the padframe have been designed. The designs have been simulated for the respective spice codes. The design can be used for fabrication or extended to make a newer system. Since the complete project is based on software, in case any changes are required they can be made very easily. So the system to design provides flexibility. The Design meets all the proposed specifications. This design concept can be a building block for higher bit ALU ex. 16-bit, 32-bit. As the open source software is used for designing of 8bit ALU the speed was not an issue and we wanted it to perform basic operations.

Satisfying design rules with all design rule checks and related network consistency checks were done.

REFERENCES

1. *Kumbhalkar, mr snehal, and sanjay tembhurne. "a novel arithmatic logic unit design for delay & area optimization." Journal of Iinformation, knowledge and research in electronics and communication January 2016.*
2. *Dubey, Vivechana, and Ravimohan Sairam. "An arithmetic and logic unit optimized for area and power." 2014 Fourth International Conference on Advanced Computing & Communication Technologies. IEEE, 2014.*
3. *Kumar, Arvind. "Comparative Study of 4-Bit ALU using CMOS and BiCMOS for 200nm Technology." Dept. of Electronics and communication, UIET, Punjab University, Chandigarh (2012).*
4. *Garima Rawat, Khyati Rathore, Siddharth Goyal, Shefali Kala''Design and Analysis of ALU: Vedic Mathematics Approach''International Conference on Computing, Communication and Automation (ICCCA2015), PP1372 1376,IEEE2015.*
5. *Begum, J. Thameema, S. Harshavardhan Naidu, N. Vaishnavi, G. Sakana, and N. Prabhakaran. "Design and Implementation of Reconfigurable ALU for Signal Processing Applications." Indian Journal of Science and Technology9, no. 2 (2016).*
6. *M Senthil kumar,Dr S Rajjalakshmi,''High Efficient Modified MixColumns in Advanced Encryption Standard using Vedic multiplier.''Current trends in Engineering and Technology,International conference on.pp462-467,IEEE2014.*
7. *Swetha, K. Lakshmi, K. Kalpana, K. Lakshmi Swetha, K. Kalpana, and M. Tech–ECE. "An Efficient Design and Implementation of ALU using Gated Diffusion Index ." International Journal Of Engineering And Computer Science ISSN:2319-7242 Volume 4 Issue , 2015,pp 12210-12215, May (2015).*
8. *Ravi Kishore Kodali, Lakshmi Boppana and Sai Sourabh Yenamachintala.'' FPGA Implementation of Vedic Floating Point Multiplier.'' Conference on Advanced technologies,pp1 -4,IEEE2014*
9. *Pandey, Bishwajeet, Jyotsana Yadav, Yatendra Kumar Singh, Ravindra Kumar, and Surabhi Patel. "Energy efficient design and implementation of ALU on 40nm FPGA." In Energy Efficient Technologies for Sustainability (ICEETS), 2013 International Conference on, pp. 45-50. IEEE, 2013.*
10. *Jana, Biswabandhu, Anindya Jana, Subhramita Basak, Jamuna Kanta Sing, and Subir Kumar Sarkar. "Design and performance analysis of reversible logic based ALU using hybrid single electron transistor." In Engineering and Computational Sciences (RAECS), 2014 Recent Advances in, pp. 1 -4. IEEE, 2014.*
11. *Bhandari, Nikhil, and Shubhajit Roy Chowdhury. "FPGA based High Performance Asynchronous ALU based on Modified 4 Phase Handshaking Protocol with Tapered Buffers." In Devices, Circuits and Communications (ICDCCom), 2014 International Conference on, pp. 1 -6. IEEE, 2014.*
12. *Nautiyal, Priyanka, Pitchaiah Madduri, and Sonam Negi. "Implementation of an ALU using modified carry select adder for low power and area-efficient applications." In Computer and Computational Sciences (ICCCS), International Conference on,pp22-25,IEEE2015*
13. *Kunal adav, Aditya vibute and Shyam Iyer Novel. "Vedic Mathematics Based ALU using application specific reversibility''. In Intelligent Systems and Control (ISCO),(2015)International Conference on ,pp.1 -5.IEEE,2015.*
14. *Abhyarthana bisoyil,Mitu Baratl,Manoja Kumar Senapati.''Compaarison of 32 bit multiplier with a conventional binary multiplier.''In Advanced Communication Control and Computing Technologies(ICACCCT),2014 International conference on,pp1758-1760.IEEE,2014.*

ISSN 2348 – 8034
Impact Factor- 4.022

15. *Randal E. Bryant, Kwang-TingCheng , Andrew B Kahang, Kurt Kreutzer, Wojciech Maly,Richard Newton, Lawerance Pileggi, Jan M Rabaey, Alberto Saniovanni- Vincentelli, "Limitations and Challenges of CAD technology for CMOS VLSI"*
16. *http://nptel.iitm.ac.in/courses/IIT-MADRAS/CAD_for_VLSI_Design_I/index.php*
17. *https://www.coursera.org/course/vlsicad/*
18. *http://www.vlsiacademy.org/open-source-cad-tools.html*
19. *Static Free Software. July 2004 http://www.staticfreesoft.com*
20. *Custom IC Design, http://www.cadence.com/products/cic*
21. *BSIM3- Introduction, www.device.eees.berkeley.edu/~bsim3*