IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)
e-ISSN: 2278-2834,p- ISSN: 2278-8735.Volume 9, Issue 3, Ver. V (May - Jun. 2014), PP 36-43
www.iosrjournals.org

Design and Implementation of 4-Bit Arithmetic and Logic Unit Chip with the Constraint of Power Consumption

Priyanka Yadav, Gaurav Kumar, Sumita Gupta
Assistant Professor Department of Electronics & Communication HITM,Agra
Research Analyst ONE97 Communication LTD. Noida
Assistant Professor Department of Electronics & Communication HITM,Agra

Abstract: *By increasing the demand of enhancing the ability of processors to handle the more complex and challenging processors has resulted in the integration of a number of processor cores into one chip. Still the load of on the processor is not less in generic system. An Arithmetic Logic Unit (ALU) is the heart of all microprocessors. It is a combinational logic unit that performs its logical or arithmetic operations. ALU is getting smaller and more complex nowadays to enable the development of a more powerful but smaller computer.*

In this proposed paper a 4 bit ALU chip has been design to the benefits of all the computations are done in parallel and available simultaneously, so no clock resources are wasted. The MUX is then simply used to select the required output.

Index Terms: *ALU, CMOS, VLSI and TTL*

I. Introduction

ALU is getting smaller and more complex nowadays to enable the development of a more powerful but smaller computer. However there are a few limiting factors that slow down the development of smaller and more complex IC chip and they are IC fabrication technology, designer productivity and design cost. The increasing demand for high speed very large scale integration (VLSI) can be addressed at different design levels, such as the architectural, circuit, layout, and the process technology level. At the circuit design level, considerable potential for speed enhancement exists by means of proper choice of a logic style for implementing combinational circuits. This is because all the important parameters governing speed switching capacitance, transition activity, and short-circuit currents are strongly influenced by the chosen logic style. Depending on the application, the kind of circuit to be implemented, and the design technique used, different performance aspects become important. In the past, the parameters like low power dissipation, small area and low cost were the major areas of concern, whereas speed considerations are now gaining the attention of the scientific community associated with VLSI design.

Arithmetic Logical Unit is the very important subsystem in the digital system design. An Arithmetic logic Unit (ALU) is an integral part of a computer processor. It is a combinational logic unit that performs its arithmetic and logic operations. ALUs of various bit-widths are frequently required in very large-scale integrated circuits (VLSI) from processors to application specific integrated circuits (ASICs). ALU is getting smaller and more complex nowadays to enable the development of a more powerful but smaller computer.

The demand for low power & high speed processing has been increasing as a result of expanding computer and signal processing applications. Higher throughput arithmetic operations are important to achieve the desired performance in many real-time signal and image processing applications. The key arithmetic operations in such applications are multiplication, addition, division and subtraction.

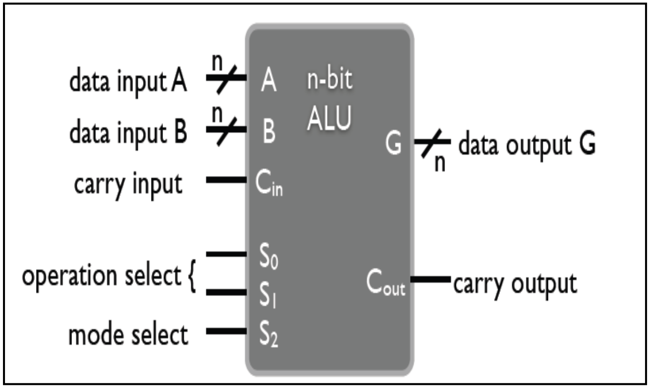


Figure 1.1 General block diagram of Arithmetic and Logic Unit [15]

In CMOS technology, Power dissipation is the most critical parameter for portability & mobility and it is classified into dynamic and static power dissipation. Dynamic power dissipation occurs when the circuit is operational, while static power dissipation becomes an issue when the circuit is inactive or is in a power-down mode. There are three major sources of power dissipation in digital CMOS circuits, which are summarized as

$$\mathbf{P_{avg} = P_{switching} + P_{shortcircuit} + P_{leakage}}$$

$$\mathbf{= \left(\alpha_{0\to1} \times C_1 \times V_{dd}^2 \times F_{clk}\right) + \left(I_{sc} \times V_{dd}\right) + \left(I_{leakage} \times V_{dd}\right)} \quad \mathbf{(1)}$$

the first term represents the switching component of power, where C_1 is the load capacitance, F_{clk} is the clock frequency and $\boldsymbol{\alpha}$ is the probability that a power consuming transition occurs (the activity factor). The second term is due to the direct-path short circuit current, I_{SC}, which arises when both the NMOS and PMOS transistors are simultaneously active, conducting current directly from supply to ground. Finally, leakage current, $I_{leakage}$, which can arise from substrate injection and sub-threshold effects, is primarily determined by fabrication technology considerations. The switching power in CMOS digital integrated circuits is a strong function of the power supply voltage. Therefore, reduction of V_{dd} emerges as a very effective means of limiting the power consumption. However, the saving in power consumption comes at a significant cost in terms increased circuit delay. Since the exact analysis of propagation delay is quite complex, a simple first order derivative can be used to show the relation between power supply and delay time

$$\mathbf{T_d \ \alpha \ C_1 V_{dd} / k(V_{dd} - V_{th})^{\alpha}} \qquad \mathbf{(2)}$$

k = transistor aspect ratio (W/L)
V_{th} = transistor threshold voltage
α = velocity saturation index which varies between 1 and 2.

Unfortunately, reducing the power supply voltage reduces power, but when the supply voltage is near to threshold voltage, the delay increases drastically.

The adder cell is the elementary unit of an ALU. The constraints the adder has to satisfy are area, power and speed requirements. By using CMOS technology, we design low power 4-bit ALU chip, also we have a care about high speed.

1.1 Features of ALU

We have designed the 4-Bit ALU which certain features as follows:

- Low-power CMOS Process Technology
- Total 16 arithmetic operations (add, subtract, plus, shift, plus 12 others)
- Total 16 logic operations (XOR, AND, NAND, NOR, OR, plus 11 others)
- Capable of active-high and active-low operation.
- Full carry look-ahead for high-speed arithmetic operation.
- Arithmetic operations expressed in 2s complement notation.
- 4 selection lines are to perform the operations on two 4 bit inputs in both the units.
- The 5th select line is used to select either one of the units.

1.2 Major Parts of ALU

• **Arithmetic block:** This block is used to perform arithmetic operations such as addition, subtraction and comparison. The core of the arithmetic block is an adder. In the architecture presented in Figure 4.1, the adder uses carry look-ahead and sum-select techniques.
• **Logic block:** This block is used to perform simple bitwise logic operations such as AND (masking), OR and XOR, XNOR, NAND, NOT and etc.

• **Multiplexers:** These blocks are used to select the appropriate inputs for the arithmetic and logic blocks. Usually more than two buses arrive at the inputs of the ALU. Sometimes these multiplexers are used to perform some simple logic operations. The 2:1 MUX can be programmed to invert one of the operands (this can be used to execute a subtraction using just an adder).

II. Related Work

In the past, ALU and full adder circuits have been implemented for optimum area and delay, each with their distinct features that bring about optimum area and delay. Some of them have been briefly described below to give us an idea of earlier work and shed light on different optimization techniques. Past work related to multiple-input floating gate CMOS applications has been reviewed to give us an idea about its operation, design and simulation issues.

But *et al* [1] have designed a low power 10-transistor full adder called Static Energy-Recovery Full-Adder (SERF) using 10 transistors. A novel set of XOR and XNOR gates in combination with existing ones have been used. The XOR and XNOR circuits designed by them do not directly connect to power and ground lines, respectively. It uses four transistors for XOR and XNOR gates in CMOS.

Wang *et al* [2] have shown an improved version of XOR and XNOR gates that make use of six transistors. In another design of CMOS 1-bit full adder cell, four transistor XOR and XNOR gates have been used [3]. The cell offers higher speed and lesser power consumption than standard 1-bit full adder cell. Radhakrishnan [4] has presented the design of low power CMOS full adder circuits using transmission function theory. This design uses six transistor CMOS XOR and XNOR gates. 16-bit, 2.4ns, 0.5nm CMOS ALU design [5], which consists of a logical and arithmetic unit (LAU), a magnitude comparator (CMP), an overflow detector (OVF) and zero flag detector (ZERO). The ALU employs a binary look-ahead carry (BLC) adder. All units in this design operate in parallel and high speed is achieved.

In the previous review 4-bit ALU design is based on multiple- input floating gate CMOS devices [6].

Harrison *et al* [7] propose an analog floating memory element for on-chip storage of bias voltages. A floating gate technology has been used to eliminate off-chip biasing voltages in the existing systems by providing these voltages on-chip, with arrays of programmable floating- gate voltages. These arrays can be individually programmed by digital controls. In [8], short-wave ultraviolet light has been applied to floating-gate devices to adjust threshold voltages for optimal performance in a circuit.

Linear amplifiers with rail-to-rail operation with supply voltages less than 1V have been designed in floating gate CMOS. Floating gate MOS circuits have the inherent ability to adapt to the incoming and outgoing signals by continuously enabling various programming mechanisms. Based on this property, single floating gate FETs, that emulate computational and adaptive properties of biological synoptic elements, were developed [12]. An example of continuously adapting floating- gate circuits is presented in [13]. The auto zeroing floating gate amplifier (AFGA) uses tunneling and pFET hot electron injection to adaptively set its dc operating point. No additional circuitry is required. Here a 4-bit ALU was designed which is operated at 3.0V, using multiple-input floating gate MOSFETs in standard 1.5μm CMOS process. The application of multi- input floating gate transistor, in designing the full adder, has enabled us to realize the full adder with fewer transistors, when compared to previous designs.

This paper describes circuit techniques for fabricating a high-speed adder using pass-transistor logic. Double pass-transistor logic (DPL) is shown to improve circuit performance at reduced supply voltage. Its symmetrical arrangement and double-transmission characteristics improve the gate speed without increasing the input capacitance. A carry propagation circuit technique called conditional carry selection (CCS) is shown to resolve the problem of series-connected pass transistors in the carry propagation path [22].

III. Proposed Methology

A 4-bit high-speed parallel Arithmetic Logic Unit (ALU). Controlled by the four Function Select inputs (S0–S3) and the Mode Control input (M), it can perform all the 16 possible logic operations or 16 different arithmetic operations on active-HIGH or active-LOW operands. The Function Table lists these operations.

When the Mode Control input (M) is HIGH, all internal carries are inhibited and the device performs logic operations on the individual bits as listed. When the Mode Control input is LOW, the carries are enabled and the device performs arithmetic operations on the two 4-bit words. The device incorporates full internal carry lookahead and provides for either ripple carry between devices using the Cn+4 output, or for carry look-ahead between packages using the signals P (Carry Propagate) and G (Carry Generate). In the Add mode, P indicates that F is 15 or more, while G indicates that F is 16 or more. In the Subtract mode, P indicates that F is zero or less, while G indicates that F is less than zero. P and G are not affected by carry in. When speed requirements are not stringent, it can be used in a simple Ripple Carry mode by connecting the Carry output (Cn+4) signal to the Carry input (Cn) of the next unit. For high-speed operation the device is used in conjunction with a carry lookahead circuit. One carry lookahead package is required for each group of four F181 devices. Carry lookahead can be provided at various levels and offers high-speed capability over extremely long word lengths.

The A = B output from the device goes HIGH when all four F outputs are HIGH and can be used to indicate logic equivalence over four bits when the unit is in the Subtract mode. The A = B output is open collector and can be wired-AND with other A = B outputs to give a comparison for more than four bits. The A = B signal can be used with the Cn+ 4 signal to indicate A > B and A < B. The Function Table lists the arithmetic operations that are performed without a carry in. An incoming carry adds a one to each operation. Thus, select code LHHL generates A minus B minus 1 (2s complement notation) without a carry in and generates A minus B when a carry is applied. Because subtraction is actually performed by complementary addition (1s complement), a carry out means borrow; thus a carry is generated when there is no underflow and no carry is generated when there is underflow. As indicated, this device can be used with either active-LOW inputs producing active-LOW outputs or with active-HIGH inputs producing active-HIGH outputs. For either case the table lists the operations that are performed to the operands labeled inside the logic symbol.

The benefit of this design is that all the computations are done in parallel and available simultaneously, so no clock resources are wasted. The MUX is then simply used to select the required output.

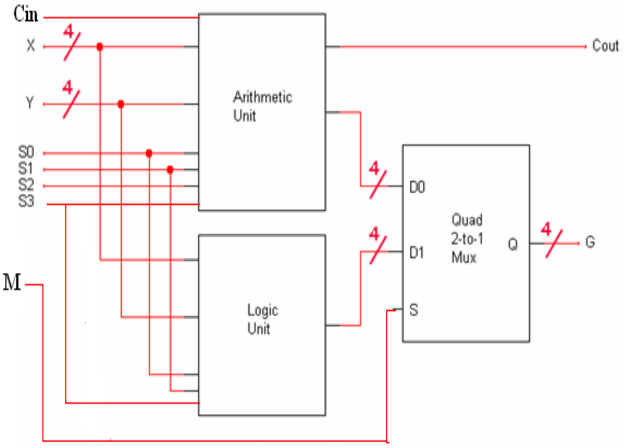
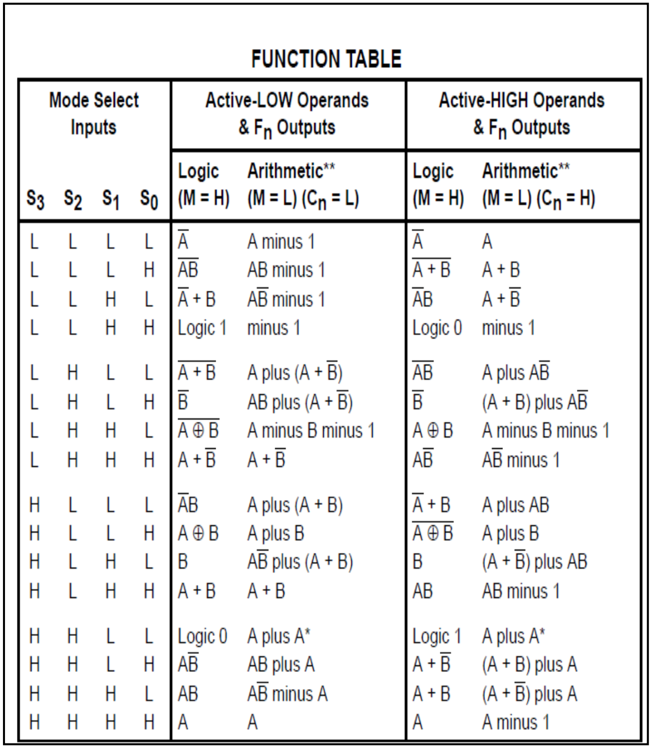


Figure 2 Block Diagram of ALU [15]

The functions performed by the ALU are specified in Table.1



FUNCTION TABLE

Mode Select Inputs				Active-LOW Operands & F_n Outputs		Active-HIGH Operands & F_n Outputs	
S_3	S_2	S_1	S_0	Logic (M = H)	Arithmetic** (M = L) (C_n = L)	Logic (M = H)	Arithmetic** (M = L) (C_n = H)
L	L	L	L	$\overline{A}$	A minus 1	$\overline{A}$	A
L	L	L	H	$\overline{AB}$	AB minus 1	$\overline{A + B}$	A + B
L	L	H	L	$\overline{A} + B$	$A\overline{B}$ minus 1	$\overline{A}B$	$A + \overline{B}$
L	L	H	H	Logic 1	minus 1	Logic 0	minus 1
L	H	L	L	$\overline{A + B}$	A plus $(A + \overline{B})$	$\overline{AB}$	A plus $A\overline{B}$
L	H	L	H	$\overline{B}$	AB plus $(A + \overline{B})$	$\overline{B}$	(A + B) plus $A\overline{B}$
L	H	H	L	$\overline{A \oplus B}$	A minus B minus 1	$A \oplus B$	A minus B minus 1
L	H	H	H	$A + \overline{B}$	$A + \overline{B}$	$A\overline{B}$	$A\overline{B}$ minus 1
H	L	L	L	$\overline{A}B$	A plus (A + B)	$\overline{A} + B$	A plus AB
H	L	L	H	$A \oplus B$	A plus B	$\overline{A \oplus B}$	A plus B
H	L	H	L	B	$A\overline{B}$ plus (A + B)	B	$(A + \overline{B})$ plus AB
H	L	H	H	A + B	A + B	AB	AB minus 1
H	H	L	L	Logic 0	A plus A*	Logic 1	A plus A*
H	H	L	H	$A\overline{B}$	AB plus A	$A + \overline{B}$	(A + B) plus A
H	H	H	L	AB	$A\overline{B}$ minus A	A + B	$(A + \overline{B})$ plus A
H	H	H	H	A	A	A	A minus 1

Table:- 1 Truth Table for 4-Bit ALU

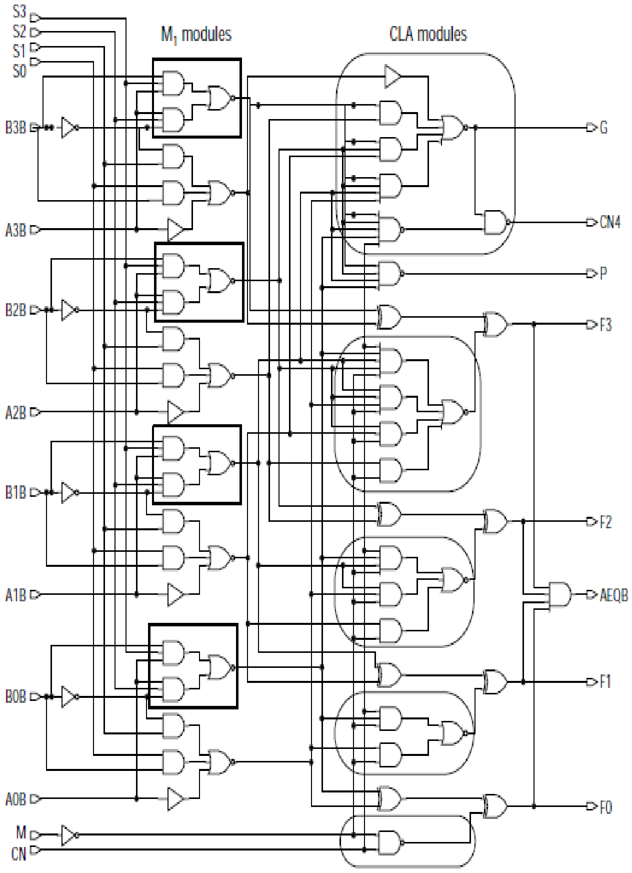


Figure 3. Logic Diagram of Proposed 4-Bit ALU

IV. Experimental Result And Discussion

TOOLS USED:

- XILINX ISE 9.2
- Mentor Graphics Tool Suite (Model Sim, ICStation, Eldo)
- Modelsim se 6.2
- Leonardo spectrum(MG)

4.1 RESULT OF LOGIC UNIT OF ALU

OUTPUT WAVEFORM OF LOGIC UNIT

In this given waveform we can see the simulation result of logic unit simulated using Modelsim 6.2. Input signals are applied at the input ports and at out port we can see the result.

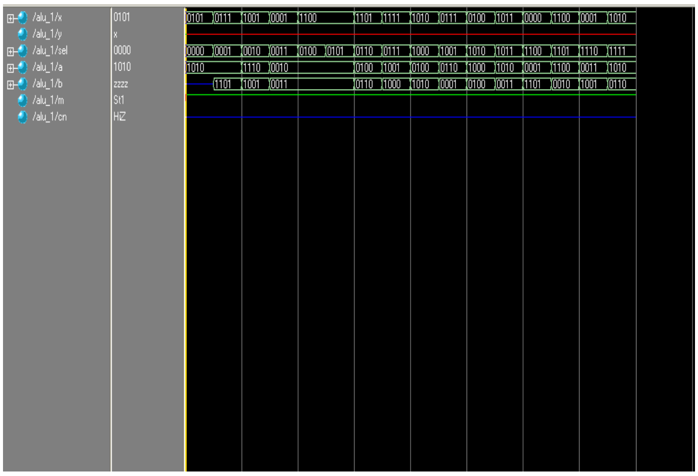


Figure 4.1 Waveform of logic unit of ALU

4.2 RESULT OF ARITHMETIC UNIT OF ALU

OUTPUT WAVEFORM OF ARITHMETIC UNIT OF ALU

In this given waveform we can see the simulation result of arithmetic unit simulated using Modelsim 6.2. Input signals are applied at the input ports and at out port we can see the result.

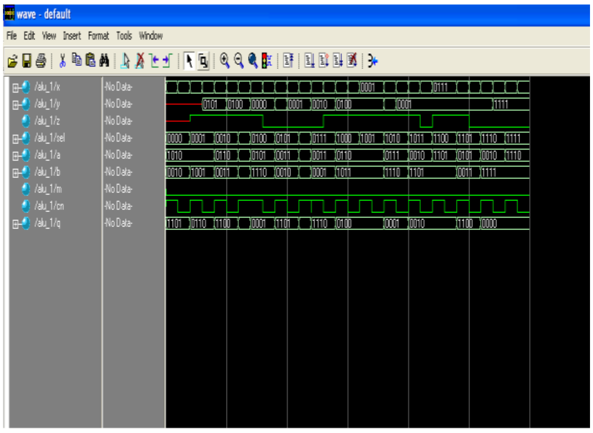


Figure 4.2 Waveform of arithmetic unit of ALU

4.3 IMPLEMENTATION OF SCHEMATIC OF ALU

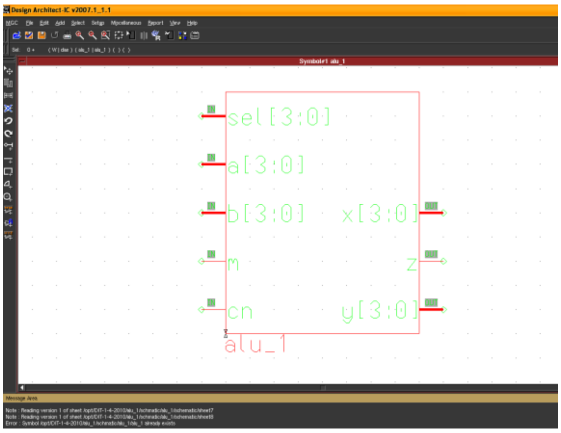


Figure 4.3 Symbol of ALU generated by MENTOR GRAPHICS (DA-IC)

The schematic for ALU generated by LEONARDO SPECTRUM is shown

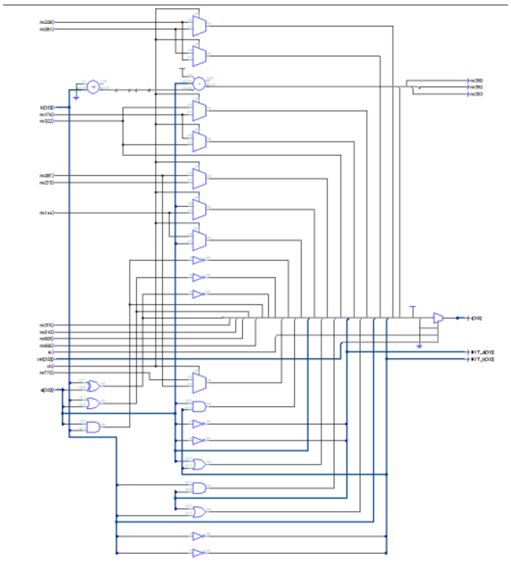


Figure 4.4 schematic for ALU generated by LEONARDO SPECTRUM

4.4 IMPLEMENTATION OF CRITICAL PATH FOR ALU

The CRITICAL PATH for ALU generated by LEONARDO SPECTRUM is shown

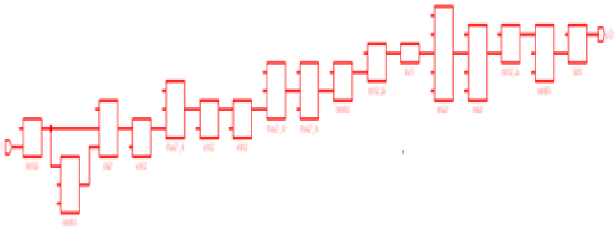


Figure 4.5 CRITICAL PATH for decoder generated by LEONARDO SPECTRUM

4.5 IMPLEMENTATION OF LAYOUT OF ALU

The layout for ALU generated by MENTOR GRAPHICS (ICStation) is shown

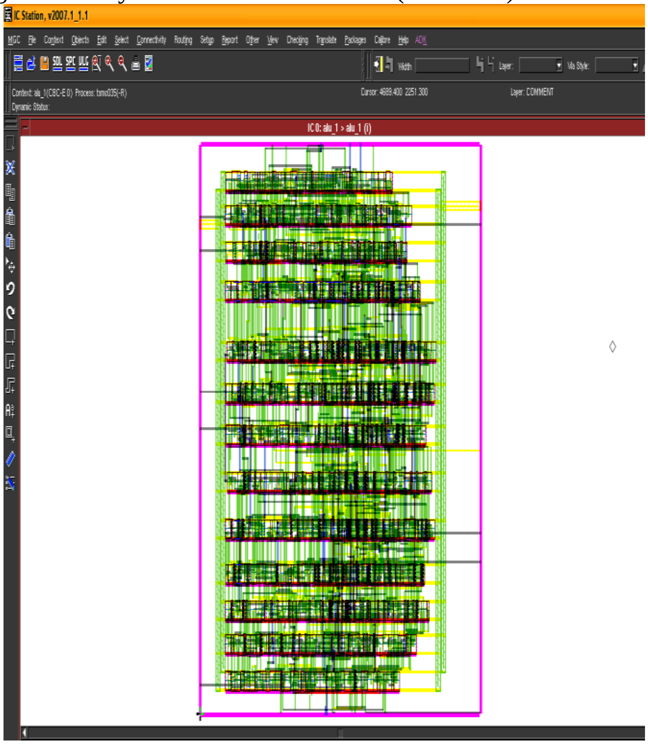
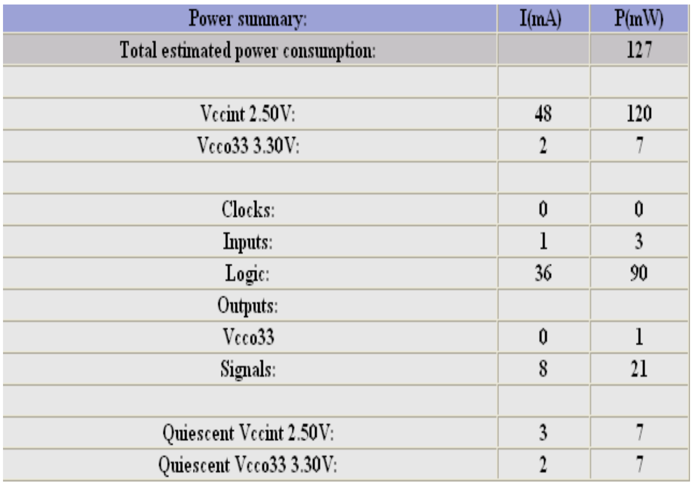


Figure 4.6 Layout for ALU generated by MENTOR GRAPHICS (IC Station)

4.6 POWER ANALYSIS OF ALU



Power summary:	I(mA)	P(mW)
Total estimated power consumption:		127
Vccint 2.50V:	48	120
Vcco33 3.30V:	2	7
Clocks:	0	0
Inputs:	1	3
Logic:	36	90
Outputs:		
Vcco33	0	1
Signals:	8	21
Quiescent Vccint 2.50V:	3	7
Quiescent Vcco33 3.30V:	2	7

V. Conclusion

To stay competitive in today's market, engineers must take a design from engineering through manufacturing with shorter design cycles and faster time to market. To be successful, we need a set of powerful, intuitive, and integrated tools that work seamlessly across the entire design flow.

We have studied well about arithmetic & logic unit. Here we have implemented 4-bit arithmetic logic unit successfully. The full integrated circuit is designed and simulated in standard 350 nm CMOS technology. Here the RTL coding is done first using a Verilog HDL and simulation is done by using ModelSim 6.2. Schematic layout simulation as well as the schematic versus layout comparison is done by using Mentor Graphics' EDA Tools are used for. The experimental waveforms, power measurements have also been presented.

The power consumption of ALU with CMOS technology is about 120mW. There is power reduction 34.34% in comparison with Schottky TTL.

References

[1] Saeid Moslehpour, Srikrishna Karatalapu" VLSI and SPICE modeling of ALU" Journal of Communication and Computer, ISSN 1548-7709, USA , Oct. 2009, Volume 6, No.10 (Serial No.59)

[2] H.T. Bui, Y. Wang and Y. Jiang, "Design and analysis of low-power 10-transistor full adders using novel XOR-XNOR gates", *IEEE Trans. On Circuits and Systems II: Analog and Digital Signal Processing*, vol. 49, pp. 25 –30, Jan. 2002.

[3] J.M. Wang, S.C. Fang and W.C. Fang, "New efficient designs for XOR and XNOR functions on transistor level", *IEEE J. of Solid State Circuits*, vol. 29, pp. 780-786, July 1994.

[4] A.M. Shams and M.A. Bayoumi, "A novel high-performance CMOS 1-bit full-adder cell", *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 47, pp. 478 –481, May 2000. 105

[5] D. Radhakrishnan, "Low-voltage low-power CMOS full adder", *IEE Proceedings-Circuits, Devices and Systems*, vol.148, pp. 19 - 24, Feb. 2001.

[6] K. Suzuki, M. Yamashina, J. Goto, Y. Inoue, T. Koseki, Y. Horiuchi, T. Hamatake, K. Kumagai, T. Enomoto and H. Yamada, "A 2.4- ns, 16-bit, 0.5- im CMOS arithmetic logic unit for microprogrammable video signal processor LSIs", *Proc. of the IEEE Custom Integrated Circuits Conference*, vol. 9, pp.12.4.1 -12.4.4, May 1993.

[7] A. Srivastava and C. Srinivasan, " ALU Design using Reconfigurable CMOS Logic", *Proc. of the 45th IEEE 2002 Midwest Symposium on Circuits and Systems*, vol.2, pp. 663-666, Aug. 2002.

[8] R.R. Harrison, J.A. Bragg, P. Hasler, B.A. Minch and S.P. Deweerth, "A CMOS programmable analog memory-cell array using floating- gate circuits" *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 48, pp. 4 –11, Jan. 2001.

[9] Y. Berg, T.S. Lande and O. Naess, "Programming floating-gate circuits with UV-activated conductances", *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 48, no. 1, pp. 12 –19, Jan. 2001.

[10] P. Hasler, "Continuous-time feedback in floating-gate MOS circuits", *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, vo l. 48, no. 1, pp. 56 –64, Jan. 2001.

[11] J.M. Rabaey, "Digital Integrated Circuits", Pearson Education, pp. 559-596, 2005.

[12] Suzuki, M. et al., "A 1.5-ns 32-b CMOS ALU in double pass-transistor logic" IEEE Journal of Solid-State Circuits, Vol 28, pp 1145-1151, 1993.

[13] Arvind Rajput, Anil Goyal, "DESIGN AND COMPARISON OF LOW POWER & HIGH SPEED 4-BIT ALU", *Proc. of 2nd National Conference on Challenges & Opportunities in Information Technology (COIT-2008),* RIMT-IET, Mandi Gobindgarh. March 29, 2008.

[14] A. Srivastava and C. Srinivasan, " ALU Design using Reconfigurable CMOS Logic", *Proc. of the 45th IEEE 2002 Midwest Symposium on Circuits and Systems*, vol.2, pp. 663-666, Aug. 2002.

[15] J. Wakerly, "Digital Design", 3rd Edition, Prentice Hall, Upper Saddle River, 2000.M. Mano and C. Kime, "Logic and Computer Design Fundamentals", 2nd Edition, Prentice Hall, Upper Saddle River, NJ, 2001

[16] D. Radhakrishnan, "Low-voltage low-power CMOS full adder", *IEE Proceedings- Circuits, Devices and Systems*, vol.148, pp. 19 - 24, Feb. 2001.

[17] N. Weste and K. Eshragian, Principles of CMOS VLSI Design: A Systems Perspective, Pearson/ Addison - Wesley Publishers, 2005.

[18] Chandrakasan, A., and Brodersen, Low Power Digital Design, Kluwer Academic Publishers, R., 1995

[19] www.datasheetcatalog.com

[20] J.P. Hayes, "Computer Architecture and Organization, Computer Organization", McGraw Hill, pp. 230-250, 1988.

[21] V.G. Mobdzija, "A Method for Speed Optimized Partial Product Reduction and Generation of Fast Parallel Multipliers Using an Algorithmic Approach", IEEE Transactions on Computers, vol. 45, No, 3, pp. 294-305,1996.

[22] Suzuki, M. et al., "A 1.5-ns 32-b CMOS ALU in double pass-transistor logic" IEEE Journal of Solid-State Circuits, Vol 28, pp 1145-1151, 1993.