IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)
e-ISSN: 2278-2834,p- ISSN: 2278-8735. Volume 7, Issue 4 (Sep. - Oct. 2013), PP 14-17
www.iosrjournals.org

VHDL Implementation Of 64-bit ALU

P Bhanusree[1], G Bhargav Sai[2], Y Ashwanth Kumar[3], K Sravan Kumar[3]

1. *Assistant Professor, Department of Electronics and Instrumentation Engineering, Vignan Institute of Technology & Science, Deshmukhi, Nalgonda Dist,- 508284, India.*
2. *Supply Chain Operation Trainee (SCOT), Hindustan Coca-Cola Beverages Pvt Ltd, Hyderabad, A.P, India*
3. *U.G Students, Department of Electronics and Instrumentation Engineering, Vignan Institute of Technology & Science, Deshmukhi, Nalgonda Dist,- 508284, India.*

Abstract: *In this paper VHDL implementation of 64-bit arithmetic logic unit (ALU) is presented. The design was implemented using VHDL Xilinx Synthesis tool ISE 9.1 and targeted for Spartan device. ALU was designed to perform arithmetic operation and logical operations such as addition , subtraction using 64-bit fast adder, logical operations such as AND, OR, XOR and NOT operations, 1'scomplement, rotate operations and compare. ALU consist of two input registers to hold the data during operation, one output register to hold the result of operation, 64-bit fast adder with 2's complement circuit to perform subtraction and logic gates to perform logical operation. The maximum propagation delay is 13.588ns and power dissipation is 38mW.*
Keywords: *ALU, Fast Adder, XILINX, VHDL*

I. INTRODUCTION

Digital hardware plays a prominent role in many electrical and computer engineering products today. This is principally due to the rapid increase in transistor densities and speed of integrated circuits and steep decline in their cost caused by the advance in micro-electronic implementation technologies. This trend is likely to continue in the foreseeable future. The "computer revolution" has affected every aspect of society and many problems viewed as being intractable can now be solved. Modern digital design relies on engineering groups made up of individuals that have an understanding of all aspects of the problem, from the top to the bottom in the hierarchical chain, with expertise in one or two areas. Present industry practice has created a high demand for systems designers with knowledge and experience in using programmable logic in the form of CPLDs and FPGAs in addition to hardware description languages.

This paper entitles 64 bit ALU design using VHDL. For modern digital design, VHDL is one of the most popular design applications used by designers. ALU's in microprocessors are the important part of the field of electronics engineering. This is a very interesting project because processors are not as flexible as programmable logic. The ability to emulate alu on a programmable chip can lead to cheaper, more efficient and more flexible performance.

II. ARITHMETIC AND LOGIC UNIT

ALU was designed to perform arithmetic and logical operations. Arithmetic operations performed are 64-bit addition, subtraction, multiplication, increment and decrement. Logical operations performed are AND, OR, XOR, NOT, NAND, NOR, SHIFT LEFT, SHIFT RIGHT, ROTATE LEFT, ROTATE RIGHT. ALU also calculates 1's complement for the 64-bit input and compares the two inputs using 64-bit comparator. ALU also consist of two input 64-bit registers to hold that data during operation and output register to hold result of operation. Fig. 1 shows the entity for ALU.

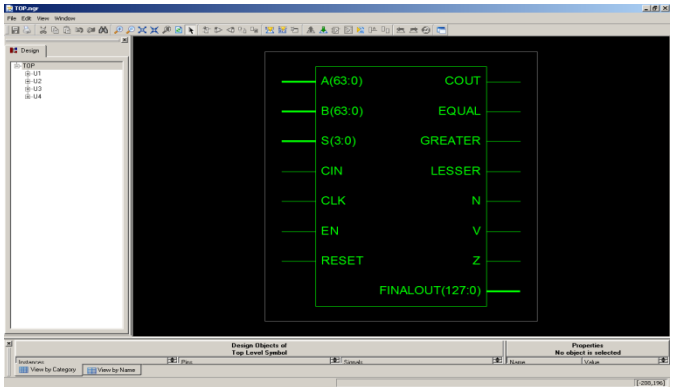

Figure 1. Entity ALU

Arithmetic Operations –

***2.1. Addition* - The addition function may be implemented using a cascaded combination of simple full adders.**

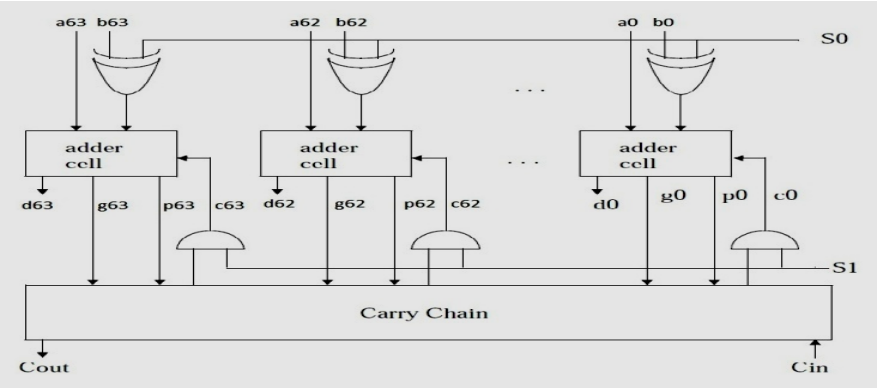

Figure 2. 64-bit Cascaded Full Adder

Figure 3.

2.2.Subtraction- The subtraction is performed using adder module by noticing that a - b = a + (-b) where -b is formed (in the 2's complement representation) by inverting all of the b-bits and adding one to the lease significant bit. The circuit for subtractor consists of an adder with inverters placed between each data input b and the corresponding input of the full adder.The input carry C0 must be equal to 1 when subtraction is performed. The result of subtraction will also be obtained in two's complement's form.

2.3.Multiplication - The multiplication is done through repeated additions, wherein the sum of all partial products are taken. The partial products depend on the multiplier bit being considered (partial product equals multiplicand if multiplier bit is 1 and partial product equals 0 if multippie bit is zero). Several schemes are available for implementing this algorithm out of which a refined version is chosen and implemented.

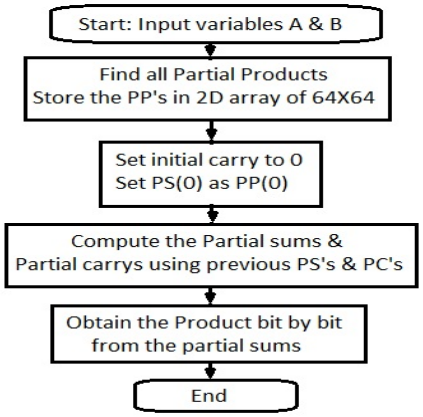

Figure 4. Multiplication Algorithm

2.4.Comparator - 64-bitcomparator compares the two inputs and generates the high signal for A greater than B, A equal to B and A less than B.

Logical Operations –

64-bit logical operations were performed on the data bitwise. This block simply consists of parallel gates connected to perform desired operation. Finally the output of the desired operation is selected by select lines. Table -1 shows the status of select lines and operations performed by ALU.

TABLE -1 OPERATIONS PERFORMED CORRESPONDING TO SELECTION LINES OF MUX

FUNCTION	SELECTION LINES				OPERATION
	S3	S2	S1	S0	
INC	0	0	0	0	A+1
ADC	0	0	0	1	A+B+Cin
SBB	0	0	1	0	A-B-Cin
DEC	0	0	1	1	A-1

AND	**0**	**1**	**0**	**0**	**A and B**
OR	**0**	**1**	**0**	**1**	**A or B**
XOR	**0**	**1**	**1**	**0**	**A xor B**
NOT	**0**	**1**	**1**	**1**	**Not A**
SHR	**1**	**0**	**0**	**0**	**Shr A**
SHL	**1**	**0**	**0**	**1**	**Shr A**
ROR	**1**	**0**	**1**	**0**	**Ror A**
ROL	**1**	**0**	**1**	**1**	**Rol A**
MUL	**1**	**1**	**0**	**0**	**A*B**
A>B/A=B/A<B	**1**	**1**	**0**	**1**	**COMPARATOR**
NAND	**1**	**1**	**1**	**0**	**A nand B**
NOR	**1**	**1**	**1**	**1**	**A nor B**

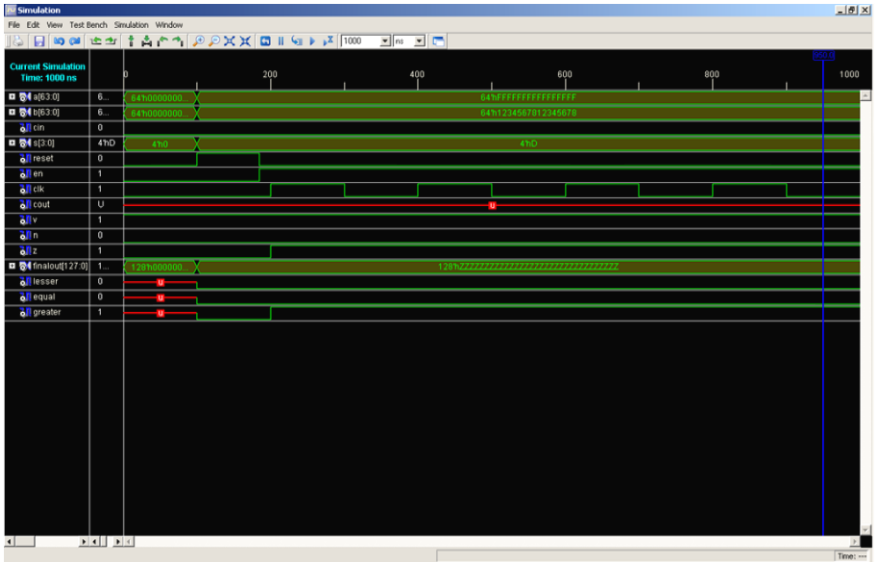

Figure 5. Simulation Results

III. RESULT

VHDL implementation of the ALU was done using Xilinx Synthesis tool 9.2 and targeted for Spartan device. Detail synthesis report is presented in table-2. Fig. 4 shows the simulation results of ALU.

TABLE-2 ADVANCED HDL SYNTHESIS REPORT

Sno	Blocks	Utilized
1	Adders/Subtractors	3
	64-bit adder carry in	1
	64-bit subtractor	1
	65-bit adder carry in	1
2	Registers	3
	Flip-Flops	3
3	Latches	7
	1-bit latch	3
	64-bit latch	1
	64-bit latch	1
	65-bit latch	1
	65-bit latch	1
4	Comparators	6
	64-bit comparator equal	2
	64-bit comparator great equal	1
	64-bit comparator greater	1
	64-bit comparator less equal	1
	64-bit comparator not equal	1
5	Xors	4035
	1-bit xor2	66
	1-bit xor3	3968
	64-bit xor2	1

IV. Conclusion

The basic structure, and design procedure of VHDL are studied. Behavioral modeling and structural modeling are used for the implementation of 64 bit ALU. The design considerations of an ALU are also studied. Logical operations are bit by bit operations and are implemented using simple gates which operate independent of each other. All the mathematical operations in the ALU are performed by means of repeated additions. Along with the basic operations of ALU multiplication and comparison are incorporated and designed as a single unit. The design consists of three modules whose outputs are combined using a multiplexer at the top most level. The project is designed and implemented using VHDL and is simulated using Xilinx9.2i ISE.

References

[1] Suchitha Kamble, Prof. N.N. Mhala, "VHDL IMPLEMENTATION OF ALU", IOSRJECE, VOL.1, ISSUE 1(May-June 2012). ISSN: 2278-2834.

[2] M.Linder,M.Schmid, "PROCESSOR IMPLEMENTATION IN VHDL",PROJECT REPORT-2007,UNIVERSITY OF ULSTER AT JORDANSTOWN.

[3] Muhammad Aizuddin Bin Che Soh, "DESIGN OF 8 BIT CPU IMPLEMENTED ON FPGA", PROJECT REPORT-April 2007, Universiti Teknikal Malaysia Melaka.

[4] David A. Patterson and John L.Henessy, "COMPUTER ORGANISATION & DESIGN"