Power Dissipation Of ALU Implementation Of GCD Processor With And Without BIST Among Various Xilinx Families

[1]Sachin D.Kohale
Post Graduate Student
[2]Ratnaprabha W.Jasutkar
Assistant Professor
Department of Computer Science and Engineering,
G.H.Raisoni College of Engineering,
Hingna Road, Nagpur, Maharashtra, India

Abstract

Power Dissipation is an important factor which is to be considered while designing any processor or digital logic circuits. The Performance of any logical circuit mainly depends upon how much power it is consuming and also depends upon the time factor in which it gives output. In this paper, we are designing Arithmetic and Logic Unit(ALU) which is a part of gcd processor, which performs not only arithmetic addition, subtraction, but also calculate greatest common divisor(gcd) of two non-negative integers using two algorithms via. Euclid's and Stein's Algorithm. Also, we are comparing the power dissipation of ALU among various Xilinx Families by applying with and without BIST technique. Selection of Xilinx Family basically depends upon lowest power consumption of the ALU.

Keywords : Arithmetic and Logic Unit(ALU) , Built in Self Test(BIST) , Euclid's algorithm, Greatest Common Divisor(GCD),Linear Feedback Shift Register(LFSR), Power Consumption, Stein's Algorithm.

1. Introduction

Low Power Electronics has become increasingly important with the advent of portable electronic devices such as laptop computers and cellular phones. Power dissipation[2] is also important in VLSI designs such as microprocessors that contain very large numbers of very small devices. Also, the power consumption problem has been raised considerable attention nowadays in design technologies. The power consumption can be reduced by decreasing the supply voltage, load capacitance and frequency. Some propose by using gated clocks to reduce the switching activity of logic in redundant cycles[4].

An arithmetic and logic unit (ALU)[1] is a digital circuit that performs arithmetic and logical operations. The ALU is a fundamental building block of the central processing unit of a computer, and even the simplest microprocessors contain one for purposes such as maintaining timers. The processors found inside modern CPUs and graphics processing units (GPUs) accommodate very powerful and very complex ALUs; a single component may contain a number of ALUs. Most of a processor's operations are performed by one or more ALUs. An ALU loads data from input registers, an external Control Unit then tells the ALU what operation to perform on that data, and then the ALU[4] stores its result into an output register. The Control Unit is responsible for moving the processed data between these registers, ALU and memory.

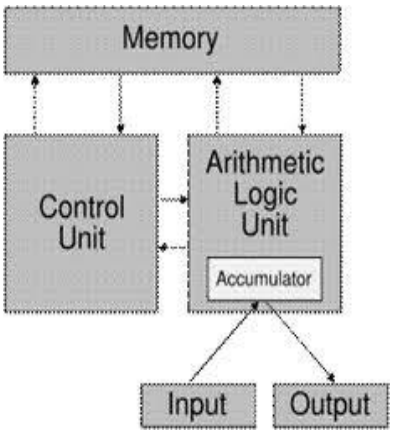

Figure 1. Block diagram of ALU

2 Built in Self Test(BIST)

Built-In Self Test(BIST) is a technique of integrating the functionality of an automatic test system onto a chip. It is a Design for Test technique in which testing (test generation and test

application) is accomplished through built in hardware features.

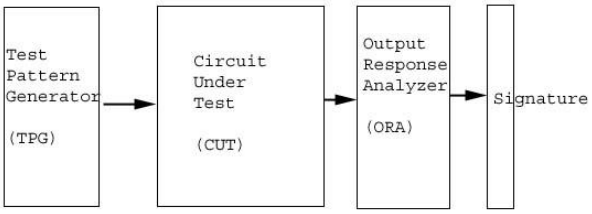

Figure 2. Block diagram of Built in Self Test(BIST)

2.1. On-line BIST

It refers to testing which occurs during normal operation of the IC. Examples of this kind of BIST often have to do with functional testing such as Error Detecting/Error Correcting (ED/EC) codes or on chip electrical monitoring.

2.2. Off-line BIST

Its operation occurs during a specified period when the Circuit Under Test(CUT) is idle. This operation occurs often over the period of multiple clock cycles and is usually intended to operate during a dedicated testing period.

3. Greatest Common Divisor(GCD)

In mathematics, the greatest common divisor(gcd),also known as the greatest common factor(gcf), or highest common factor(hcf), of two or more non-zero integers, is the largest positive integers that divides the numbers without a remainder. For example, the GCD of 48 and 180 is 12 shown in figure 3 below.

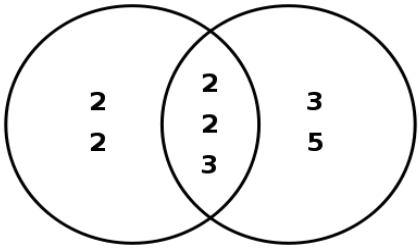

Figure 3. Greatest Common Divisor(gcd) calculations of two non-negative integers

3.1. Euclid's Algorithm

In Mathematics, the Euclidean algorithm or Euclid's algorithm[3], is an efficient method of computing the greatest common divisor(gcd) of two integers, also known as greatest common factor(gcf) or highest common factor(hcf). It is named after the Greek Mathematician , Euclid. In its simplest form, Euclid's algorithm[3] starts with a pair of positive integers and forms a new pair that consists of the smaller number and the difference between the smaller and larger numbers. The process repeats until the numbers are equal. That number then is the greatest common divisor of the original pair.

Basically Euclid algorithm can be described as

$$gcd(a , 0) = a \quad (1)$$

$$gcd(a , b) = gcd(b , a \bmod b) \quad (2)$$

If arguments are both greater than zero, then

$$gcd(a , a) = a \quad (3)$$

$$gcd(a , b) = gcd(a - b , b) ; \text{if } b < a \quad (4)$$

$$gcd(a , b) = gcd(a , b - a) ; \text{if } a < b \quad (5)$$

For ex. gcd(20 , 0) is 20 [1]. Similarly, gcd(20 , 10) [4] is same as gcd((20-10) , 10) = gcd(10 , 10) = 10.

3.2. Stein's Algorithm

This algorithm is also known as binary gcd algorithm. It is algorithm that computes the greatest common divisor of two nonnegative integers. It gains a measure of efficiency over the ancient Euclidean algorithm by replacing divisions and multiplications with shifts, which are cheaper when operating on the binary representation used by modern computers. This is particularly critical on embedded platforms that have no direct processor support for calculations of division. Basically Stein's algorithm[6] can be described as

$$gcd(0 , v) = v \quad (6)$$

$$gcd(u , 0) = u \quad (7)$$

$$gcd(0 , 0) = 0 \quad (8)$$

If u and v are both **even**, then

$$gcd(u , v) = 2.gcd(u/2 , v/2) \quad (9)$$

If u is **even** and v is **odd,** then

$$gcd(u , v) = gcd(u/2 , v) \quad (10)$$

Similarly u is **odd** and v is **even** then

$$gcd(u , v) = gcd(u , v/2) \quad (11)$$

If u and v are both **odd** and **u is ≥ v**, then

$$\gcd(u, v) = \gcd((u - v)/2, v) \quad (12)$$

If both are **odd** and **u < v** , then

$$\gcd(u, v) = \gcd((v - u)/2, u) \quad (13)$$

For ex. gcd(0 , 22) is 22 [6]. Also, gcd(33 , 0) is 33 [7]. Similarly , gcd(21 , 22) is same as gcd(21 , 11) [11]. Also, gcd(21 , 41) is same as gcd((41 - 21) / 2 , 21) is again same as gcd(10 , 21)[13].

4.Linear Feedback Shift Register(LFSR)

An LFSR[5] is a shift register that, when clocked, advances the signal through the register from one bit to the next most-significant bit (see Figure 4) below. Some of the outputs are combined in exclusive-OR configuration to form a feedback mechanism. A linear feedback shift register can be formed by performing exclusive-OR on the outputs of two or more of the flip-flops together and feeding those outputs back into the input of one of the flip-flops as shown in Figure 5 below.

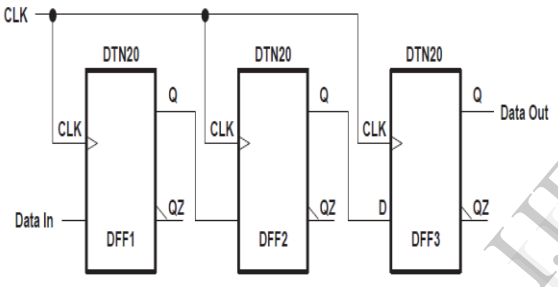

Figure 4. A 3-bit Shift Registers

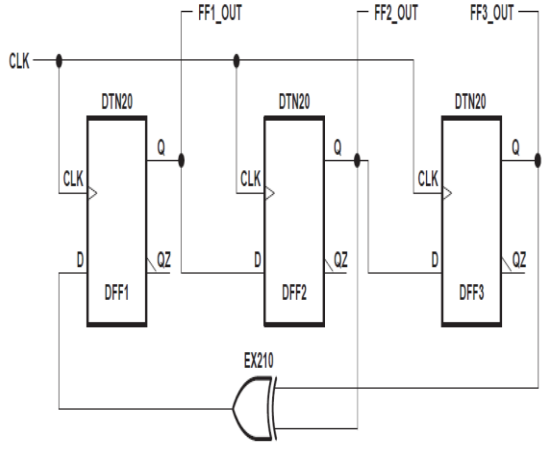

Figure 5. Linear Feedback Shift Registers

Linear feedback shift registers make extremely good pseudorandom pattern generators. When the outputs of the flip-flops are loaded with a seed value (anything except all 0s, which would cause the LFSR to produce all 0 patterns) and when the LFSR is clocked, it will generate a pseudorandom pattern of 1s and 0s. Note that the only signal necessary to generate the test patterns is the clock.

5.Power Dissipated by ALU of GCD Processor With and Without BIST

The Graphs shown in Figure 6 and Figure 7 below indicates the power dissipated by Arithmetic and Logic Unit(ALU) of GCD Processor among various Xilinx Families.

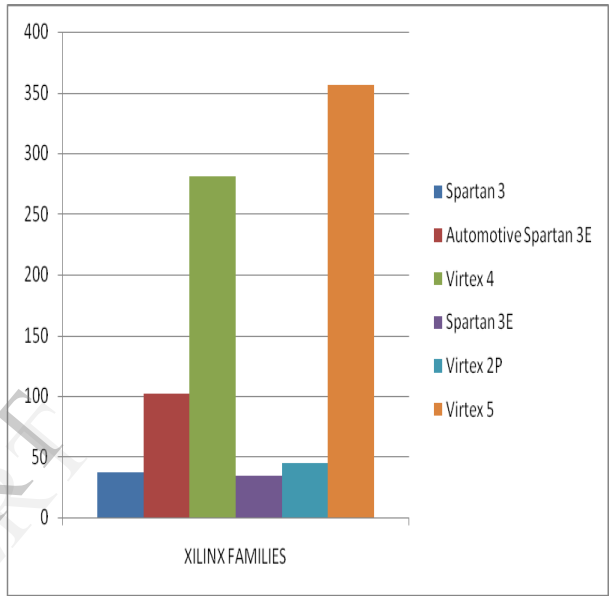

Figure 6. Power Dissipated by ALU of GCD Processor with BIST technique considering 8-bit input data's

From the Figure 6 above, it is been observed that the less power is dissipated by FPGA family named as "Spartan 3E". The calculated value of power dissipation is shown in Table 1 below.

Table 1. Power Dissipation with BIST Technique

FPGA Family	Power (in Milli-Watts)
Spartan 3E	34
Spartan 3	37
Virtex 2P	45
Automotive Spartan 3E	102
Virtex 4	281
Virtex 5	356

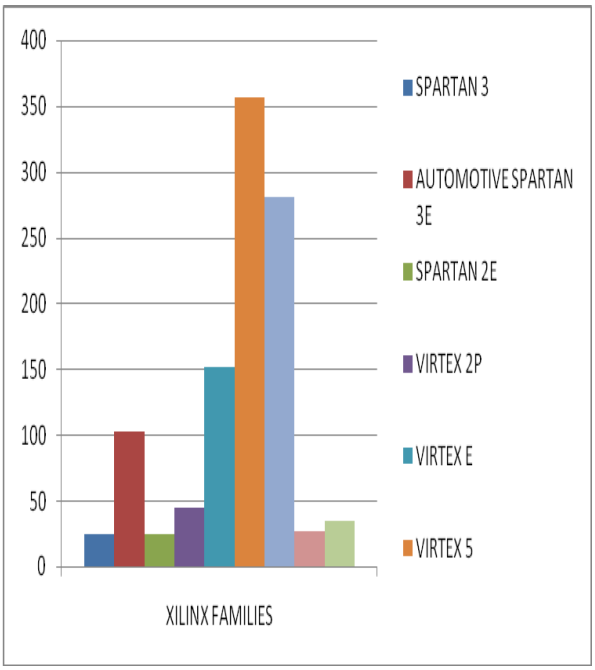

Figure 7. Power Dissipated by ALU of GCD Processor without BIST technique considering 8-bit input data's

From the Table 2 shown below, it is been observed that the power dissipation is minimum for Spartan 3 FPGA Xilinx Family and it is maximum for Virtex 5 Family.

Table 2. Power Dissipation without BIST Technique

FPGA Family	Power (in Milli-Watts)
Spartan 3	24
Spartan 2E	25
Virtex	27
Spartan 3E	34
Virtex 2P	45
Automotive Spartan 3E	102
Virtex E	151
Virtex 4	281
Virtex 5	356

From the Figure 7 above, it is been observed that the less power is dissipated by FPGA family named as "Spartan 3". The calculated value of power dissipation is shown in Table 2 below.

6.Operations Performed by ALU of GCD Processor With & Without BIST

Table 3. ALU operations with & without BIST Technique

Operation	Opcode	Description
gcd(a,b)	00	gcd of two inputs a and b is calculated using Euclid's Algorithm.
gcd(a,b)	01	gcd of two inputs a and b is calculated using Stein's Algorithm.
add(a,b)	10	Addition of two inputs a and b
sub(a,b)	11	Subtraction of two inputs a and b

Table 3 Shown above shows different operations performed by Arithmetic and Logic Unit(ALU) of GCD Processor. The operations performed by ALU with and without BIST technique is same, except that in case of ALU with BIST technique, random inputs are generated by using Linear Feedback Shift Register(LFSR) circuitry. Select lines are used for selecting operations. The output waveforms of ALU implementation of gcd processor without BIST is shown in Fig. 8 below.

Similarly, output waveforms of ALU implementations of gcd processor is shown in Fig.9 below.

8.Simulation Results

8.1 Without BIST

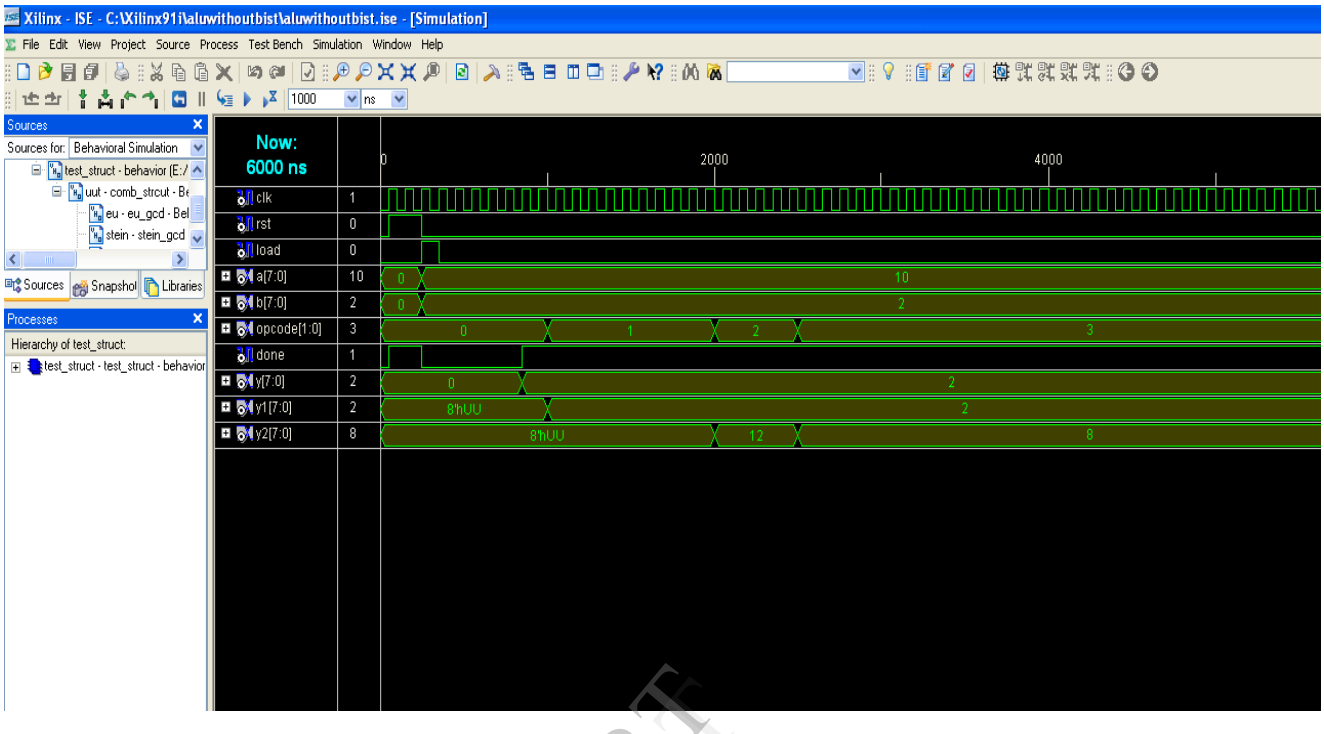

Figure 8. ALU Implementation of GCD Processor without Build in Self Test(BIST) Technique

8.2 With BIST

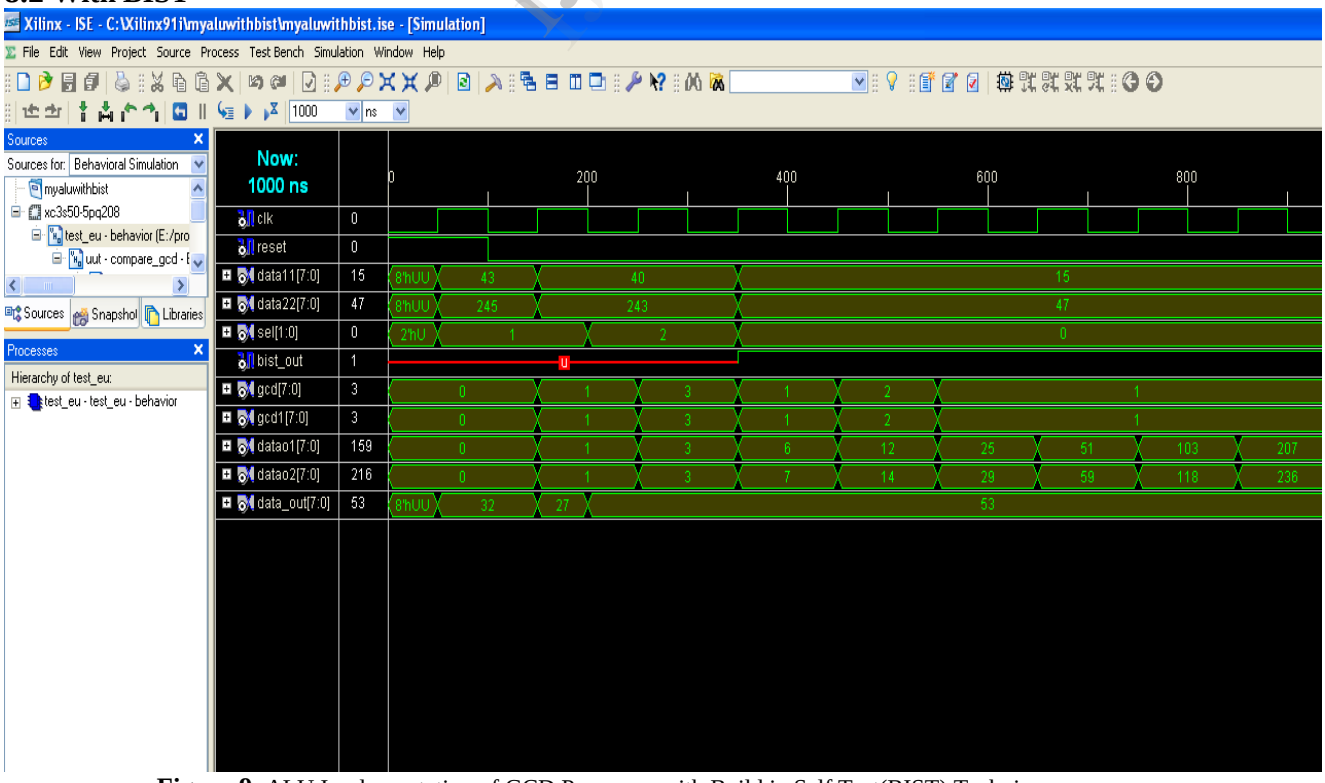

Figure 9. ALU Implementation of GCD Processor with Build in Self Test(BIST) Technique

9.RTL View

9.1 Without BIST

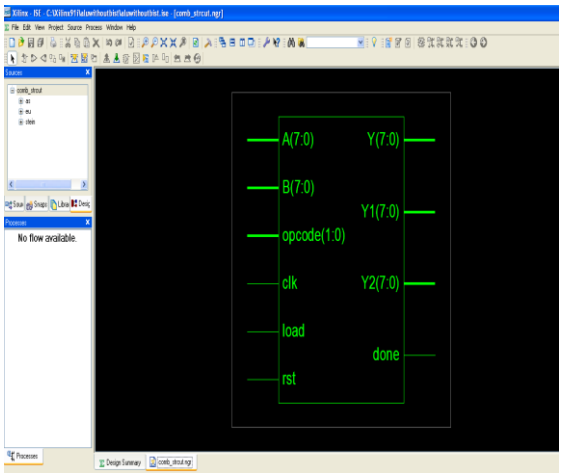

Figure 10. RTL View of ALU Implementation of GCD Processor without Build in Self Test(BIST) Technique

A = first 8-bit input.

B = second 8-bit input.

Opcode = 00 / 01 /10 /11 for gcd using Euclid's algorithm / gcd using Stein's algorithm / addition / subtraction.

clk = generating clock.

load= if '1' , loads input 8-bit data in A and B.

rst = for resetting operation.

done= if '1' , when completed ALU operations.

Y = gcd output using Euclid's algorithm.

Y=gcd(A,B)

Y1= gcd output using Stein's algorithm.

Y1=gcd(A,B)

Y2= add / subtract two 8-bit input A & B.

Y2=A+B / A-B

9.2 With BIST

The RTL view of ALU implementation of GCD Processor with BIST technique is shown in Figure 11 below.

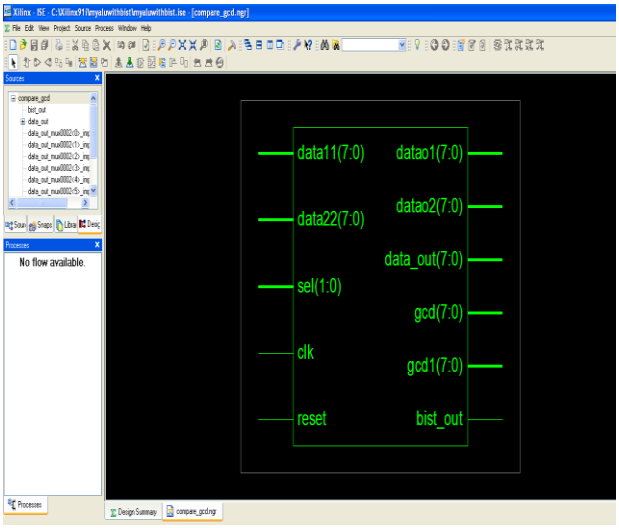

Figure 11. RTL View of ALU Implementation of GCD Processor with Build in Self Test(BIST) Technique

data01 = 8-bit random first input generated using LFSR Circuitry.

data02 = 8-bit random second input generated using LFSR Circuitry.

sel = 00 / 01 /10 /11 for gcd using Euclid's algorithm / gcd using Stein's algorithm / addition / subtraction.

10.Conclusion

In this paper, we have designed Arithmetic and Logic Unit(ALU) of GCD Processor with BIST and without BIST technique using two algorithms. The ALU operations and outputs are shown in figure 8 and 9 above(with BIST and without BIST technique).

Spartan 3E FPGA family can be preferable for designing GCD Processor, if we are employing BIST features into it. And, the power dissipated by ALU implemented using Spartan 3E family is 34 mw which is less.

Also, if we are not employing BIST features while designing ALU part of GCD Processor, so Spartan 3 is more preferable and power dissipated by it is 24 mw.

So, it can be conclude that power dissipated by implemented device is an important factor and this factor should be kept in mind while designing any processor. Xilinx Family should be selected in such a way that it can dissipate less power while performing ALU operations inside the processor. thus improved the performance of the processor. Although Power dissipation is not the only factor.

References

[1] Rajeev Kumar, Manpreet Kaur,"Design & Implementation of 64-bit ALU for Instruction set Architecture & Comparison between Speed/Power Consumption on FPGA",*International Journal of Advanced Research in Computer Engineering & Technology*, vol. 1,issue 4,June-2012,pp.186-193.

[2] Nadir Z. Basturkmen ,Sudhakar M. Reddy , Irith Pomeranz , A Low Power Pseudo-Random BIST Technique, Research supported in part by NSFGrant No. CCR-0097905 and in part by SRC Grant No. 2001-TJ-949 and Research supported in part by NSFGrant No. CCR-0098091 and in part by SRC Grant No. 2001-TJ-950.

[3 Navdeep Kaur, Neeru Malhotra, Balwinder Singh, " VHDL Implementation of ALU with Built In Self Test Technique", *International Journal of Engineering Research and Development* ,vol.5,issue 1,Nov.2012,pp.14-17.

[4 P.Manikandan, B.D.Liu, L. Y. Chiou, G.Sundar, C. R. Mandal, "Asynchronous Design Methodology for an Efficient Implementation of Low power ALU ",pp.590-593 *APCCAS 2006 IEEE*.

[5] John Koeter, What's an LFSR?,Texas Instrument, SCTA036A December 1996,1-12.

[6] George Purdy, Carla Purdy , Kiran Vedantam, "Two Binary Algorithms for Calculating the Jacobi Symbol and a Fast Systolic Implementation in Hardware",@2006IEEE.

[7] Pavel Emeliyanenko," High-performance Polynomial GCD Computations on Graphics Processors",pp.215-224, 978-1-61284-383-4/11/$26.00 ©2011 IEEE.

[8] Prashanth B.u.v , P.Anil Kumai, .G Sreenivasulu," Design & Implementation of Floating point ALU on a FPGA Processor", 2012 International Conference on Computing, Electronics and Electrical Technologies [ICCEET],pp. 772-776.

[9] Shikha Khurana, Kanika Kaur,"Implementation of ALU using FPGA", *International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)*, vol. 1, iss. 2, July-August 2012,pp.146-149.

Authors

Sachin D. Kohale received Bachelor of Engineering (B.E.) degree in Electronics from K.D.K.College of Engineering ,Nagpur, Maharashtra, India in 2008. He is pursuing Master of Engineering(M.E.) in Embedded System and Computing(ESC) from G.H.Raisoni College of Engineering, Nagpur, Maharashtra, India. His research area includes designing gcd processor , performance comparison of Algorithms used for designing gcd processor and also designing digital logic circuit.

Ratnaprabha W.Jasutkar received Master of Technology(M.Tech.) degree in VLSI Design from R.K.N.E.C.,Nagpur,Maharashtra, India . Presently, she is pursuing Phd. in the field of Mixed Signal VLSI. She is also working as Assistant Professor in Computer Science and Engineering Dept. in G.H. Raisoni College of Engineering, Nagpur. Her research area includes Mixed Signal VLSI , VLSI Testing , Embedded System & Wireless Networks etc.