ISSN: 2277-9655
Impact Factor: 4.116

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

SPAA AWARE ERROR TOLERANT 32 BIT ARITHMETIC AND LOGICAL UNIT FOR GRAPHICS PROCESSOR UNIT

Kaushal Kumar Sahu*, Nitin Jain
Department of Digital electronics,CSVTU Bhilai,Chhattisgarh
Department of electronics & telecommunication,CSVTU Bhilai,Chhattisgarh

DOI: 10.5281/zenodo.56917

ABSTRACT

Arithmetic and logical unit are responsible for all computationally intensive task which determines the speed and reliability of a processor. In other word we can say ALU is the brain of a processor. Nowadays every portable devices are battery operated so primary concern of those devices are low power consumption. But at the same time we want higher performance also so that there should not be any lag while using those devices. Graphically intensive application demands more resources and at the same time demand more power. Optimization between speed of operation and power consumption is the key challenge in design paradigm. The performance increase can be achieved by increasing clock frequency, but it leads to some other issues such as overheating, leakage etc. Instead, We approach to exploiting parallelism at the architecture level, which significantly increase throughput without compromising on power. Instead, using single, powerful CPU, we can add a cluster of less powerful CPU on a single chip, which combinatorically gives better performance. In proposing work we present architecture of 32 bit ALU for graphics processors. We have designed and implemented multicore processor based on 8bit ALU unit, which is specifically designed for low power consumption. The synthesized architecture is implemented in Verilog HDL. Analysis is performing on FPGA Artix-7 (Field Programmable Gate Array) level.

KEYWORDS: ALU, Verilog, FPGA, Artix-7, System design

INTRODUCTION

In recent years, the demand for power-sensitive designs has grown significantly. This is because of the increase in demand of mobile computing devices. To get high performance, semiconductor devices are shrinking to small sizes aggressively. Which leads to increasing density of transistors on a die, higher frequency of operation and causing high power consumption. To reduce power consumption, we have to scale supply voltage to maintain it within limits. But scaling of supply voltage is limited due to performance consideration. To address this issue we have to work on circuit techniques and system level design.

The ALU is most busy and workhorse of any processor. If we can optimize activities on a ALU then power optimization issue can be solved. There are two typical ALU design structures: tree structure and chain structure. An example is shown, where the ALU contains four functional components in addition, logic and, logic XOR, and logic OR.

LITERATURE SURVEY

Booth[1] in 1951 proposed a signed binary recording scheme, which is used for reduction in number of partial products in multiplier. Later Wallace [2] in his work improved integer operation. In [3] present a 32, 64 and 128 bit mode FMA unit with SIMD support. To the best of our knowledge, our proposed architecture is the first true dynamic precision architecture targeting both fixed point and floating-point ALUs. Now a the days in current research there is also a very usefull technique is VEDIC mathematics using this logic there is improvement in timing complexity, area power [4].

There is some more latest research on the arithmetic logical unit, in [5] author presents a ALU which contains two sub modules lower bound module and upper bound module. This design performs add, subtract, multiply and set operation of union, divide is performed by shifting Lower and upper bound module is selected by flag generation. The drawback of this approach is Hardware size and power consumption is increases for division only shifting property is used.

In [6] author proposed a ALU unit for 8 bit microcontroller according to that approach proposed ALU contains three sub modules, ARITHMETIC, LOGIC, and BIT operation. In [7] author proposed a approach which have two type of ALU structure first one is tree and the second one is chain. In tree approach require less area as compare to other design. In chain approach latency of design is fast and controlling on operation is simple. Problem with this approach is there is no any control signal, when any two inputs A & B are generated so that input is computed by every computation unit. Require extra hardware for Chain Structure. Input, Output Pins are increases in chain structure. Tree structure increase latency according to operation.

There is some another logic is developed in present era which is known as Approximation [8, 9] according to that approach there is many applications which are error tolerant means the error which is not identified by human eyes. Similar for power reduction there is one useful module which is known as Clock gating [10] according to that it will reduce the dynamic power of the whole system.

PROPOSED SYSTEM

As we know in present era every multimedia and general purpose application demands fast and ultra-low power system. In current stage every device is operated on battery power supply and as we know for battery there is some kind of limitation with their power issues and battery size. We also know for any processing unit there is the most important part is an Arithmetical logical unit (ALU). Due to heavy arithmetic and logical operation generally ALU requires a heavy amount of power as compare to complete system. Due to these reasons we think for a generation of the approximate ALU unit. Which is a combination of four ALU which are:

1. Eight Bit Pure Accurate ALU Unit
2. Eight Bit Semi Accurate ALU unit
3. Eight Bit Semi Approximate ALU Unit
4. Eight Bit Pure Approximate ALU Unit.

In our proposed approach we also use logic of clock gating to reduce the clock power and dynamic power. Here our proposed design will work on total 16 instruction those are followings:

Table 1. ALU operation

Arithmetical Operation	Logical Operation
1. Addition	**1.** AND Operation
2. Subtraction	**2.** OR Operation
3. Multiplication	**3.** NOR Operation
4. Division	**4.** NAND operation
5. Square	**5.** XOR Operation
6. Modules	**6.** XNOR Operation **7.** 1's Complement **8.** 2's Complement **9.** Right Shifting **10.** Left Shifting

ISSN: 2277-9655
Impact Factor: 4.116

A. Proposed Architecture

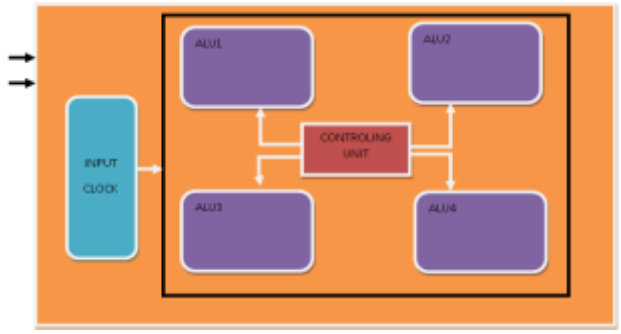

Figure. 1 Proposed architecture

Detailed Description about Proposed architecture:

Pure Accurate ALU: In this architecture we are using all accurate logic and also this architecture is follows the previous existing architecture of the ALU. Pure Accurate ALU follows total 16 instruction set which is the combination of arithmetic and logical operation.

Semi-Accurate ALU: In this architecture we are using some approximate and some accurate instruction set. In this block all logic is accurate and some arithmetic is accurate. Here every operation is truncated 4 LSB bit of the ALU.

Semi Approximate ALU: In this architecture we are using all arithmetic instruction as an approximate and some logical as a approximate which is two's complement. Here for every approximate arithmetic operation, We will truncate 2 LSB bit of 8 bit ALU. Here it follows total 16 instruction set which is a combination of arithmetical and logical.

Pure Approximate ALU: In this architecture we are using all arithmetic instruction as an approximate and some logical as an approximate which is two's complement. Here for every approximate arithmetic operation, we will truncate 4 LSB bit of 8 bit ALU. Here it follows total 16 instruction set which is a combination of arithmetical and logical.

Implementation Details:

We are using hardware descriptive language, Verilog. Using this language we were design existing and proposed approach on FPGA based designing tool and design verification is done on Model SIM. Here we are designing our complete design in FPGA which is known as ARTIX-7 which is based on 28nm Technology.

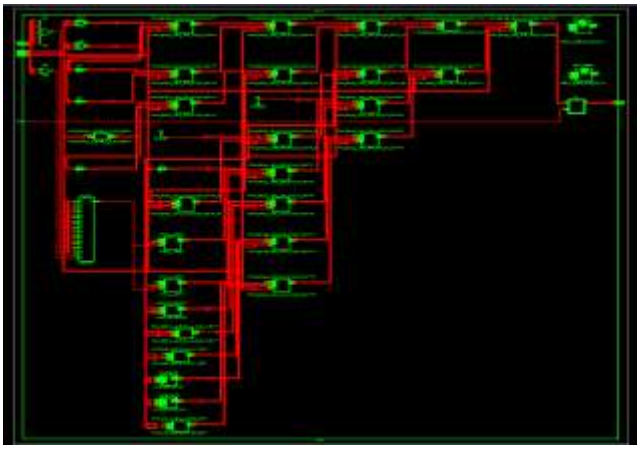

Figure. 2 RTL Level Schematic 32 bit Accurate ALU

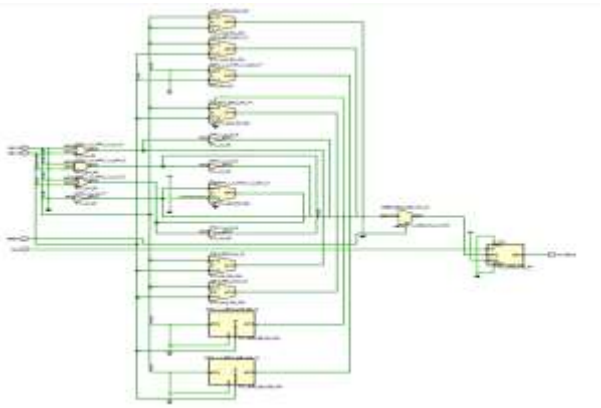

Figure 3 Gate level Schematic 32 bit Accurate ALU

ISSN: 2277-9655
Impact Factor: 4.116

- **Implementation of Vedic based ALU Architecture [11]:**

Here we are designing existing Vedic based 32 bit ALU logic with 16 different instructions set. Here basically ALU is having one multiplier module which is based on the Vedic concept of URDHAVA multiplier. Here we show Gate level schematic and logic based schematic of accurate ALU:

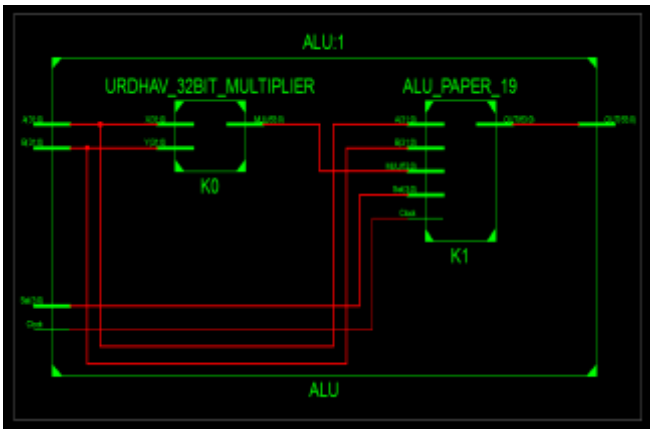

Fig 4 RTL Schematic Vedic ALU

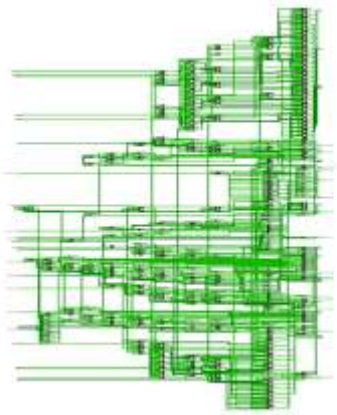

Fig 5 Gate level Schematic Vedic ALU

- **Implementation of Pipe based ALU Architecture [5]:**

Here we are designing an existing pipeline based 32 bit ALU logic with 16 different instructions set. Here basically ALU is divided in two different part first one is LSB based ALU, Second one MSB based ALU. Here we show Gate level schematic and logic based schematic of accurate ALU:

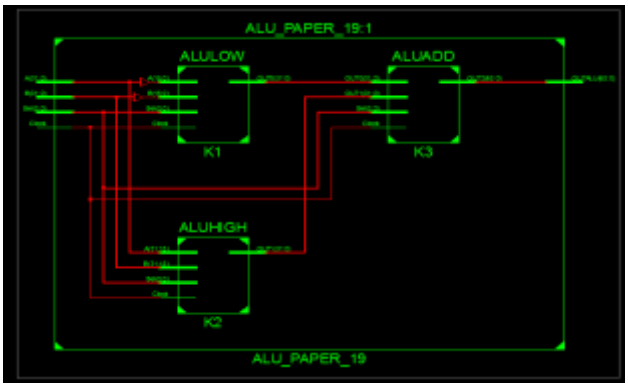

Fig 6 RTL Schematic Pipe based ALU

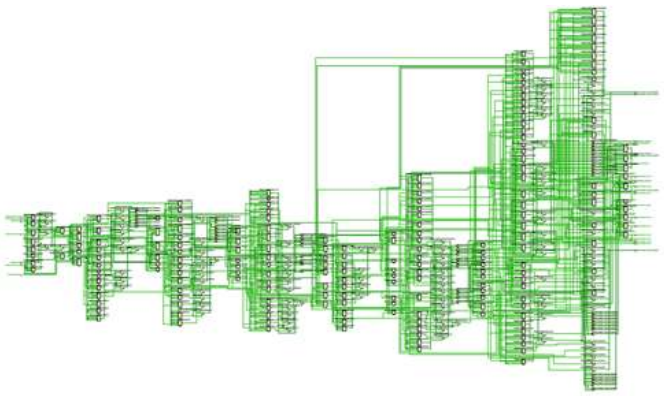

Fig 7 Gate Level Schematic Pipe Based ALU

- **Implementation of ALU Architecture [6]**

Here author Josip proposed an ALU architecture where he divides instruction set in the section LOGIC, Arithmetic, Bit Here we show Gate level schematic and logic based schematic of accurate ALU:

ISSN: 2277-9655
Impact Factor: 4.116

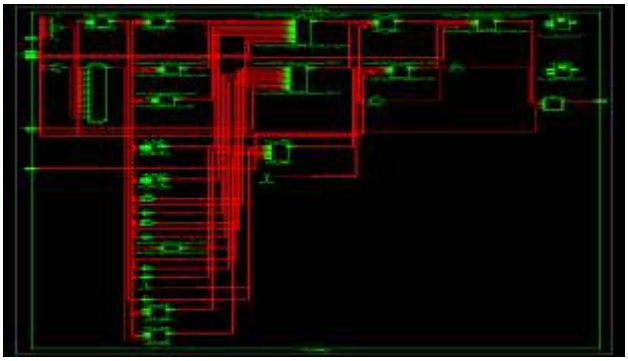

Fig 8 RTL Schematic Josip implemented ALU ALU

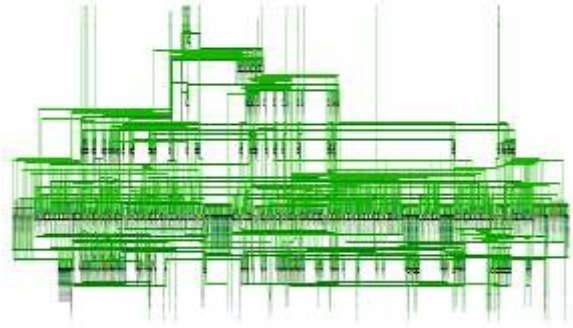

Fig 9 Gate Level Schematic of Josip implemented

- **Implementation of Proposed ALU Architecture:**

Here we are present a proposed architecture of our ALU. Which is basically having four different eight bit block. Which is known as Accurate, Semi accurate and approximate. Here all implementation is done on Xilinx 14.2 tool using Verilog. Basically, our design is work with 16 different instruction set of combination of arithmetic and logical. Here we show Gate level schematic and logic based schematic of accurate ALU:

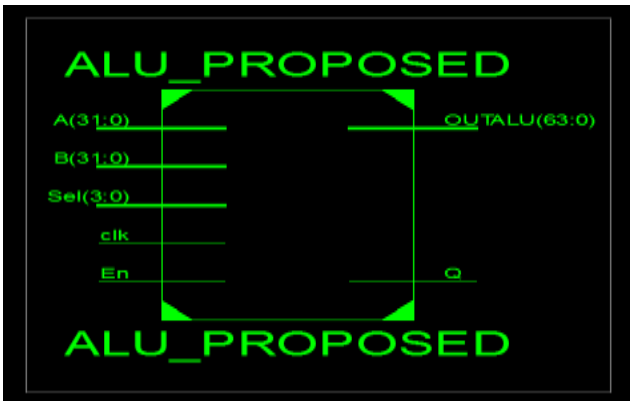

Fig 10 Proposed ALU Schematic

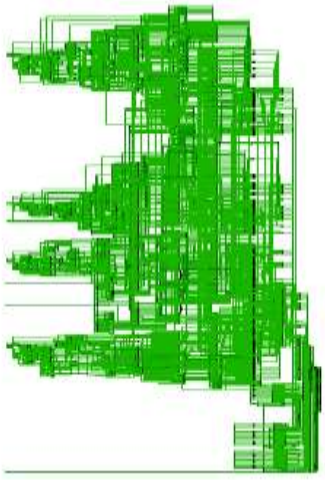

Fig 11 Gate level Schematic Proposed ALU

RESULTS

In this section we are representing comparative analysis of proposed ALU architecture with existing ALU architecture in terms of power, area (LUT), delay and frequency. The FPGA comparison analysis of proposed and accurate are shown below, here hardware analysis is done on Vertix 6 FPA which is 45nm based technology.

ISSN: 2277-9655
Impact Factor: 4.116

A. **Simulation Output**

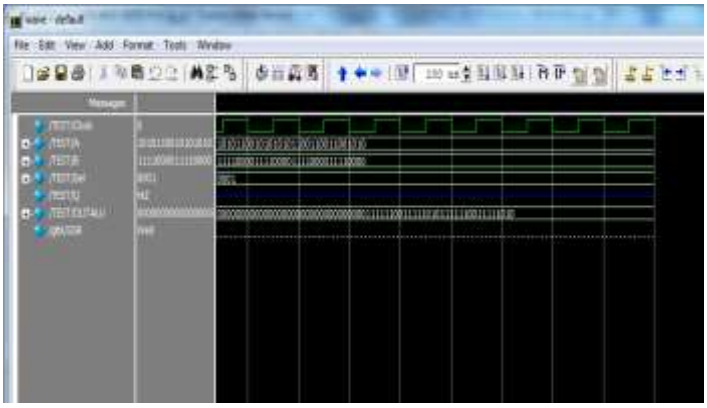

Fig 12 Simulation Output

Xpower Output:

ALU Accurate:

Fig. 13 Xpower Output ALU Accurate

ALU [4]:

Fig. 14 Xpower Output Vedic ALU

ALU [12]:

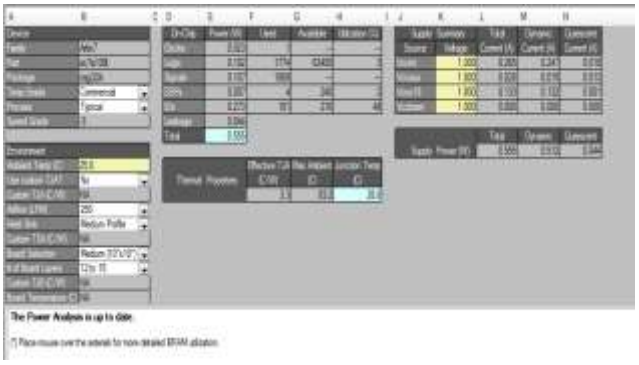

Fig 15 Xpower Output Pipe Based ALU

ALU [6]:

Fig. 16 Xpower output Josip based ALU

ISSN: 2277-9655
Impact Factor: 4.116

ALU PROPOSED:

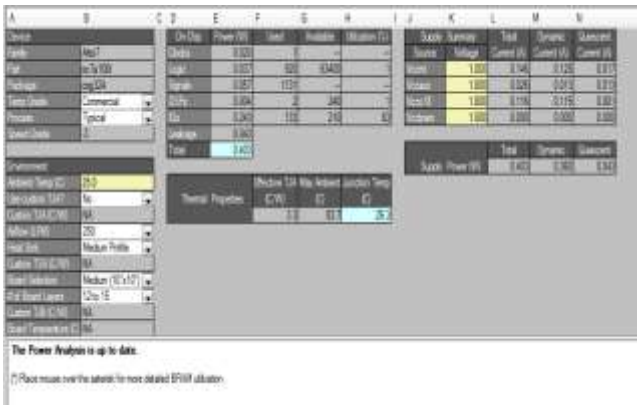

Fig 17 Xpower Output Proposed ALU

LUT (Logic Blocks):

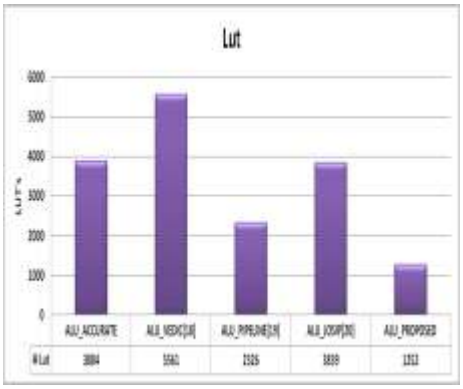

Fig 18 Comparison analysis of Logic Block

Power (W):

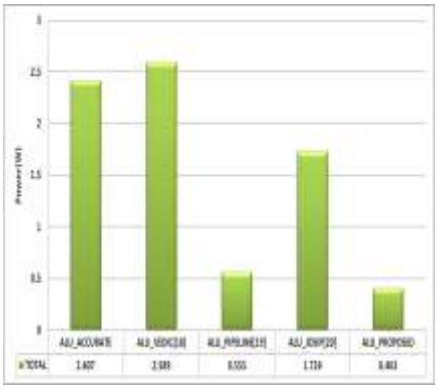

Fig 19 Power Blocks

Static Power (W):

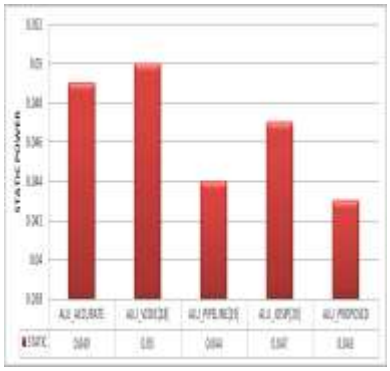

Fig 20 Static Power Blocks

Dynamic Power (W):

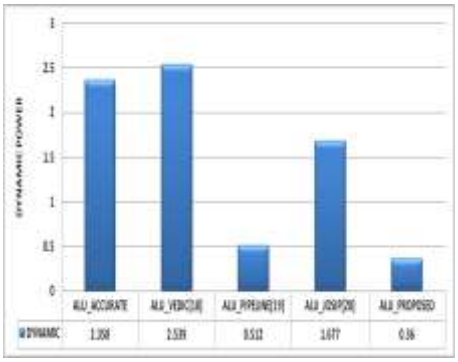

Fig. 21 Dynamic Power Blocks

Delay (ns):

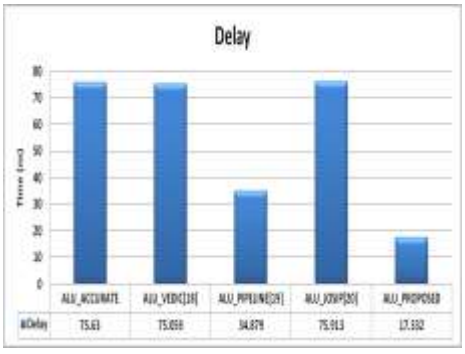

Fig. 22 Delay Blocks

Frequency (MHz)

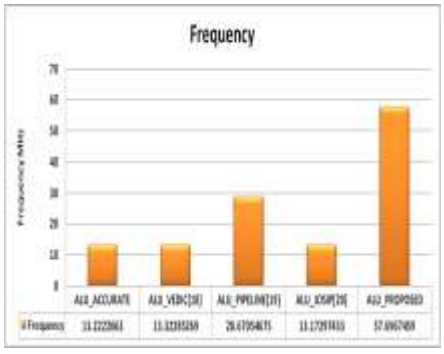

Fig. 23 Frequency Blocks

ISSN: 2277-9655
Impact Factor: 4.116

CONCLUSION

According to our previous discussion as we already saw there is lots of issues in previous existing ALU design so in this thesis we try to resolve some of the issues, according to this thesis. We are presenting a novel approach for 32 bit ALU architecture which is a combination of four sub ALU block of eight bits where, one block is 100% accurate, rest three are 98%, 95% and 90%. The second block is semi accurate, third block is semi approximate and fourth block is pure approximate. All hardware implementations are done on Xilinx 14.2 and design simulation is done on model SIM. Here we are using 28nm technology based FPGA which is known as Artix-7. Here simulation results show that there is a 75% improvement over delay and frequency. For power there is a 50% improvement is done at last as for area there is a 72% improvement is done as compared to previous existing approach.

REFERENCES

[1] A. Booth, “A signed binary multiplication technique,” The Quarterly Journal of Mechanics and Applied, vol. 4, no. 2, pp.236-240, 1951.
[2] C. S. Wallace, “A suggestion for a fast multiplier,” Electronic Computers, IEEE Transactions on, vol. EC–13, no. 1, pp. 14–17, 1964.
[3] L. Huang, S. Ma, L. Shen, Z. Wang, and N. Xiao, “Low CostBinary128 Floating-Point FMA Unit Design with SIMD Support,” IEEE Transactions on Computers, vol. PP, no. 99,pp. 1-8, 2011.
[4] Gupta, A., U. Malviya, and Vinod Kapse. "Design of speed, energy and power efficient reversible logic based vedic ALU for digital processors." Engineering (NUiCONE), 2012 Nirma University International Conference on. IEEE, 2012.
[5] Gupte, Ruchir, et al. "Pipelined ALU for signal processing to implement interval arithmetic." Signal Processing Systems Design and Implementation, 2006. SIPS'06. IEEE Workshop on. IEEE, 2006.
[6] Divic, Josip, and Marino Debeljuh. "Model of 8-bit microprocessor intended for lecturing." MIPRO, 2012 Proceedings of the 35th International Convention. IEEE, 2012.
[7] Zhou, Yu, and Hui Guo. "Application specific low power ALU design." Embedded and Ubiquitous Computing, 2008. EUC'08. IEEE/IFIP International Conference on. Vol. 1. IEEE, 2008.
[8] Kyaw, Khaing Yin, Wang Ling Goh, and Kiat Seng Yeo. "Low-power, high-speed multiplier for error-tolerant application." 2010 IEEE International Conference of Electron Devices and Solid-State Circuits (EDSSC). 2010.
[9] Lee, Kyoungwoo, et al. "Error-exploiting video encoder to extend energy/qos tradeoffs for mobile embedded systems." Distributed Embedded Systems: Design, Middleware and Resources. Springer US, 2008. 23-34.
[10] Kathuria, Jagrit, M. Ayoubkhan, and Arti Noor. "A review of clock gating techniques." MIT International Journal of Electronics and Communication Engineering 1.2 (2011): 106-114.
[11] Gupta, A., U. Malviya, and Vinod Kapse. "Design of speed, energy and power efficient reversible logic based Vedic ALU for digital processors." Engineering (NUiCONE), 2012 Nirma University International Conference on. IEEE, 2012
[12] S. Anderson and J. Earle, “The IBM system/360 model 91:floating-point execution unit,” IBM Journal of, no. January, 1967