A Comprehensive Comparison of ALU or Floating Point Unit Architecture and Device Technologies

Alexis Drayton
Department of Electrical and Computer Engineering
University of Central Florida
Orlando, FL 32816-2362

Abstract— As time passes technology can only progress. This progress can be seen in the comparison of the review technologies through the years. In this paper, a comparative study of Arithmetic Logic Unit and Floating Point Unit designs are presented. ALUs and FPUs are the fundamental building blocks of the important computing circuits in a computer preforming basic operations like addition, subtraction, multiplication, division and bitwise operations on binary and integer numbers, floating point unit programming these on floating point numbers. For the purpose of the this paper a comparison of the different designs of ALUs and FPUs will be made, comparing their fundamental metric such as their device technologies, datapath width, the run time of their arithmetic functions, technology node length and their power consumption. Some of the designs looked over will include, Low-Power Fault Testable 32 bit ALU design, Low power high speed 32 bit ALU design and a floating point unit 15 nm CMOS design to name a few.

Keywords— Arithmetic Logic Unit, Floating Point Unit, Device Technology, CMOS technology, ITRS technology node, data bus width, power dissipation, Energy consumption.

I. INTRODUCTION

ALUs or Floating Point Units are the hearts of any central processing unit and their designs are important to how the computer will compute its basic operations. The basic operations which the ALU is accountable for include addition, subtraction, multiplication, division and bitwise operations done to integer and binary numbers. For decimal values, the Floating Point Unit takes control and preforms the same operations on these values. The calculations made in these units are decisive in everyday computing as most programs as are compiled down to machine code which these units manipulate using their basic operations to execute the desired instructions.

The Arithmetic Logic Unit (ALU) is central to every processor. The ALU has two input operands and one output operand all with the same datapath width. ALUs are designed primarily with Boolean gates and each instruction has its own set of bit crunching gates. The adder is key in designing ALUs where the adder is used to build the multiplier etc. The Floating Point unit is only slightly different, capable of doing such operations for decimal values. The FPU splits the decimal into its sign, exponent and mantissa fraction bits to calculated the basic operations for decimal values. The sign say whether or not the number is positive, the exponent says where the decimal place is and the mantissa is the decimal value.

Before continuing some key terms must be defined; data bus width, ITRS technology node, execution time, power dissipation, and energy consumption of processors. These terms are user frequently in this paper. Data bus width is the number of bits that can be in the input or output of the ALU determining the amount of data that can be transferred at once, determining the processing power of a computer. The ITRS technology node is in relation to nanometer and is half the distance between transistors or the “half pitch between two adjacent DRAM metal lines. ITRS stands for the International Technology Roadmap for Semiconductors and they provide guidance for the various technology nodes. Execution time in computing is the amount of time it takes a process or program to run and terminate successfully. Power dissipation is the amount of power the system consumes in a neutral state. The energy consumption is the amount of energy consumed to successfully complete the desired task.

In the following section, eight ALU and Floating point unit designs spanning from the year 2002 to today are reviewed. Specifically seeing how the technology has changed thorough the years by looking at specific key metrics such as node size, datapath width, operation time and their power consumption.

II. LITERATURE REVIEW

2000-2005

Beginning in 2002, the first ALU design reviewed is the design of a 4 bit ALU using reconfigurable logic of multi input floating gate (MIFG) CMOS [5]. This unit was designed in 1.5 um technology for 3V operations, preforming four arithmetic and four logical operations. The full adder in this unit, the main function of any ALU, was made with MIFG transistors. His design is a relatively simple structure with a significant decrease in the number of transistors and interconnections in comparison to earlier ALU designs. The results shown that this full adder design yields a 32 ns full adder sum execution time with a 0.5 mW power consumption. The design is rather successful only using 8 transistors opposed to the 50 transistors used in conventional full adder designs in that time [5].

In 2004, the ALU design used on Altera’s Apex 20KE FPGA architecture was reviewed. This ALU design is used in Altera’s NIOS 2.0 soft processor and implemented on Altera’s Apex 20KE FPGA architecture [7]. The datapath for this ALU analyzed is 32 bit and 16 bit. The adder’s execution time for the
16-bit was 3 ns and 5.0 ns for the 32-bit. The power consumption is assumed to be low powered with the chip model NIOS 2.0.

In 2005, Low Power delay fault testable 32 bit ALU in 180 nm CMOS technology design was analyzed. This design allows low power consumption operation while supporting a design-for-test (DFT) scheme for delay-fault testability [2]. The full adder in ALU for this design had an execution time of 257 ps.

Another ALU design reviewed in 2005 was a Low-Voltage Full Adder Cells in different CMOS logic styles for the predominating tree structured arithmetic circuits [3]. This design is just 1-bit full adder cell with 0.18 um CMOS technology. The sum and carry generation circuits of a new hybrid style full adder are designed with hybrid logic styles. The full adder operation time was 1.42 ns with 0.8V C-CMOS.

2009 – 2015

In 2009, a low power, high speed ALU design using feedback switch logic (FSL) is reviewed. FSL is suitable for high speed and low power because it offers fast switching, reduced capacitance and input switching dependent activity without the need of a clock [6]. This ALU design is a 32-bit design based on static CMOS and FSL logic at 90 nm CMOS technology. Tests were run both in CMOS and using FSL. In CMOS the adder average time was 488.7 ps with 571.1 uW average power consumption. In FSL, the adder average time was 422 ps with 617 uW power consumption.

The ALU design reviewed in 2011 is high speed, low power full adder Cells testing with a datapath width of 1 bit. This is designed with an alternative internal logic structure and pass-transistor logic styles that lead to a reduced power-delay product [8]. This full adder design uses 0.18 um CMOS technology. In this review two full adders of this design were tested and compared with other hybrid adders. The speeds for the two created adder were 289 ps and 278 ps. The energy consumption of these adder are 55.1 uW and 60.6 uW respectively.

In 2015 two units were reviewed. Firstly, an IEEE 754 Single Precision Floating Point Unit design was analyzed in both single and double precision with datapath widths of 32 bit and 64 bit respectively on 45nm and 15nm technologies. The floating point operation time for the 45nm technology was 5 ns and for the 15nm technology 0.4 ns. The power consumption of the technologies was 2.04 mW and 0.6 nW respectively. The 15 nm technology using 4 time less energy and 3-fold smaller footprint [1].

Also in 2015 a low power high speed hybrid full adder circuit was analyzed with both 1 bit and 32 bit datapath widths. The circuit was also implemented using 180 nm and 90 nm technologies [4]. Using the 1 bit datapath width, the operation time for the adder using 180 nm technology was 224 ps with an average power consumption of 4.1562 uW for 1.8V supply. With 90 nm technology, the adder time is 91.3 ps with an average power consumption of 1.17662 uW at 1.2 V supply voltage. The design was extended also for a 32 bit datapath with result of, for the 180 nm technology, 5.578 ns adder operation time and 112.79 uW power consumption for 1.8 V supply voltage. For 90 nm datapath, the adder operation time is 2.45 nm and 53.36 uW power consumption for 1.2V.

III. Data Analysis

The first comparison made in this report is the trends of the yearly data versus the width of the data bus. Shown in the below graph and in Table 1, the years for each device technology being reviewed doesn’t vary much. The trend between year and data bus width might not be clearly represented as much of the Device technologies being reviewed for a single year are discussed using varying bus widths. For example, in 2015 for different data bus widths were reviewed.

![Figure 1 Year versus Data Bus Width](image1)

The second data analyzed was the year and the execution time of the various device technologies discussed in Table 1. Noticeably the execution time dips drastically after 2002. This could be due to the data bus width difference in the other years. Some Devices are only 1-bit adders having a significantly shorter runtime, others range between 16-64-bit ALUs or Floating Point Units with faster runtime, while in 2002 a 4 bit ALU was studied. Other than that the execution times stay in the same range each year.

![Figure 2 Year versus Execution Time](image2)

To further analysis Table 1, we look at the yearly difference in ITRS technology node lengths. Similar to execution time we see an initial decrease and then a stagnant curve for the rest of
the years. The non-variation of the node data can be attributed to the data mostly focusing on Adder designs being relatively low powered and have high speed.

The final comparison made was the power consumption for each year. The power for each technology relatively stays the same except for in 2015, there is a spike in the power consumption for double precision 64 bit 45 nm node device technology, floating point unit. The main goal for most of the device technologies studied was for them to be low powered and most of them are staying within 1000 uW except for the floating point unit.

IV. CONCLUSION

In conclusion, after reviewing the literature and making the data analysis, the deduction can be made of the progress of ALU and FPU design throughout the years. It can be seen that the unit and adder designs are all relatively progressing in terms of adder speed and power consumption. The comparative speed and power consumed in 2004 of the 4 bit ALU design was much more that the 32 bit ALU in 2009, where the speed for the 2009 ALU would be considerably faster that the ALU in 2004 but also without much increase in power consumption. Also the low power high speed full Adder circuit in 2011 for its 1-bit datapath and 90 nm node length uses considerable more energy that the low power high speed full adder in 2015 with 1 bit datapath width and 90 nm technology node length with only difference being that the latter is a hybrid full adder. In general, the progression of ALU design through the years is present in the findings of this study.

REFERENCES


<table>
<thead>
<tr>
<th>ALU or Floating Point Architecture Name</th>
<th>Device Technology</th>
<th>Datapath width (bits) or #bits in operands</th>
<th>Time for Operation or Design Type</th>
<th>ITRS Technology Node (nm) or Area (µm²)</th>
<th>Model of Chip used</th>
<th>Energy/Power Consumption (W or J) else indicate “low” or “high”</th>
</tr>
</thead>
<tbody>
<tr>
<td>4-bit ALU design using reconfigurable logic of multi-input floating gate [5] 2002</td>
<td>MIFG - CMOS</td>
<td>4 bit</td>
<td>32 ns (full adder sum in ALU)</td>
<td>N/A</td>
<td>N/A</td>
<td>1.5 µm 0.5 mW</td>
</tr>
<tr>
<td>ALU design used in on Altera’s Apex 20KE FPGA architecture [7] 2004</td>
<td>NIOS</td>
<td>16 bit 32 bit</td>
<td>3.0 ns (16 bit) 5.0 ns (32 bit)</td>
<td>N/A</td>
<td>N/A</td>
<td>NIOS 2.0 Low power</td>
</tr>
<tr>
<td>1.7-GHz Low-Power Delay-Fault-Testable 32-b ALU [2] 2005</td>
<td>CMOS</td>
<td>32 bit</td>
<td>267 ps (full adder in ALU)</td>
<td>N/A</td>
<td>N/A</td>
<td>180 nm 0.84 uW (0.8V C-CMOS)</td>
</tr>
<tr>
<td>Low-Voltage Full Adder Cells [3] 2005</td>
<td>CMOS</td>
<td>1 bit</td>
<td>1.42 ns (0.8 V C-CMOS)</td>
<td>N/A</td>
<td>N/A</td>
<td>0.18 µm 571.1 uW (Average in CMOS) 617 uW (FSL)</td>
</tr>
<tr>
<td>Low Power High Speed 32-bit ALU [6] 2009</td>
<td>CMOS FSL</td>
<td>32 bit</td>
<td>488.7 ps (Adder average in CMOS) 422 ps (Average in FSL)</td>
<td>N/A</td>
<td>N/A</td>
<td>90 nm 55.1 uW (Ours1) 60.6 uW (Ours2)</td>
</tr>
<tr>
<td>High-Speed and Low-Power Full-Adder Cells [8] 2011</td>
<td>CMOS</td>
<td>1 bit</td>
<td>289 ps (Ours1) 278 ps (Ours2)</td>
<td>N/A</td>
<td>N/A</td>
<td>0.18 µm 45 nm 2.04 mW 15 nm 0.6 mW</td>
</tr>
<tr>
<td>Floating-Point Unit Design [1] 2015</td>
<td>CMOS</td>
<td>32-bit single precision (operalnd) 64-bit double precision</td>
<td>N/A</td>
<td>5 ns (45 nm) 0.4 ns (15 nm)</td>
<td>45 nm 15mn 45 nm 2.04 mW 15 nm 0.6 mW</td>
<td></td>
</tr>
<tr>
<td>Low-Power High-Speed Hybrid Full Adder Circuit [4] 2015</td>
<td>CMOS</td>
<td>1 bit 32 bit</td>
<td>224 ps (180 nm 1 bit) 91.3 ps (90 nm 1 bit) 5.578 ns (180 nm 32 bit) 2.45 ns (90 nm 1 bit)</td>
<td>N/A</td>
<td>N/A</td>
<td>180 nm 180 nm 180 nm 1.17664 uW (1 bit) 53.36 uW (32 bit)</td>
</tr>
</tbody>
</table>