Adder/Multiplier Design and Metric Evaluation

Michael Rathbun
Department of Electrical and Computer Engineering
University of Central Florida
Orlando, FL 32816-2362

Abstract—Many Arithmic Logic Unit (ALU) designs have been constructed and tested in order to optimized different functions in which the Computer Processing Unit (CPU) uses to carry out various operations. As it is used so often, it is important that the ALU function correctly, with high performance, and with low energy consumption. This paper reviews and analyses ten different ALU designs, focusing on metrics such as ALU performance, power dissipation, energy consumption, ITRS node technology and space complexity.

One interesting ALU design that is mentioned is the 16-bit ALU that is designed to lower power dissipation and increase performance. Disabling the adder function while processing other functions was one feature that was enabled to achieve an energy consumption of 54mW [9]. This design also used pipelining to help decrease adder execution time to 5ns [9].

Another interesting ALU design is one that used FIR filters and pipelining to improve energy efficiency. With a 64-bit data bus width and 0.24µm² cell area, energy efficiency was improved by 62.5% in comparison to the original design [4].

Keywords—Energy consumption, clock cycles, power dissipation, space complexity, ITRS node, optimization

I. INTRODUCTION

The ALU is critical and used often in computer processing. Because of this, it is important for it to be fast and reliable. Functions in the ALU that include add, multiply, shift, and logic compose most of the “bit crunching” that occurs in Computer Processing Units (CPU). The task of optimizing this unit can become complicated with the large amount of computing power. Many studies have been conducted to find the best way of achieving better performance or reliability, or both. Such characteristics as ITRS Node, pipelining and chip design can be improved and analyzed to improve performance, power consumption and power efficiency.

As research has shown, pipelining is an effective design to improve both performance and power consumption. In one study, power consumption was reduced by about 62.5% and 65-66% in another [4, 5]. By computing many stages of addition at once instead of one after the other, results can be obtained quicker and within less clock cycles.

Chip selection also shows great promise in improving ALU performance characteristics. A more condensed chip size means there are less components being used on that chip. With less components, there is less energy being consumed and dissipated. One study showed that by reducing chip size by 50%, ALU performance had increased 70% [10].

The Arithmic Logic Unit (ALU) is a critical piece of hardware in computer processing. Most computing that is needed by the processor is requested for the ALU. As a function is called by the CPU, the ALU takes inputs (operands) and directs them through wires (data path) and gates (control path). The control path can consist of such components as logical gates (XOR, OR, AND, etc.) or multiplexers, including others. Depending on the arrangement of bits that are contained in the instruction code, the control path will determine which data path will be taken. Once computation is done in the ALU, it is sent back to the CPU to be sent to its final destination.

This paragraph will define the metrics that will be analyzed in this report. Data bus width describes the number of bits in which data can be transferred from one functional unit to another. ITRS technology node describes the International Technology Roadmap for Semiconductors regulation on sizing for distance between transistors on a silicon wafer. Execution time is the period in which a functional processing unit completes its task in full. This period excludes time taken to wait for data that is needed to move forward. Power dissipation is the act of releasing energy in non-useful forms of energy. Heat is one of the most apparent examples in computer technology. Energy consumption of processors is the total amount of energy that a system uses to function. This includes both useful and non-useful forms of energy.

To come in this paper is a review and evaluation of various designs and metrics of adders and multipliers. I will go over such metrics as ITRS Node, data bus width, energy consumption and power dissipation. A data analysis over the span of the last fifteen years will also be covered to show various trends based on the afore mentioned metrics.

II. LITERATURE REVIEW

Starting in 1999, a 16-bit ALU was designed at the transistor level [9]. This structure uses a two-stage pipelined architecture and uses features such as disabling the adder while processing other functions to decrease energy consumption [9]. With 16 bits used for each operand and 0.6µm² cell area, this ALU processes its adder function in 5ns with an energy consumption of 54mW [9]. A few years later in 2002, an ALU is created using a method called back-gate forward substrate bias (BGFSB) to showcase its usefulness in lowering power dissipation and increasing performance [7]. Using a 1.2µm chip, 4 bits for operands and a ripple carry adder, this ALU is able to decrease execution time by up to 3.8 fold and decrease energy consumption and power dissipation by using a 1V voltage.
supply [7]. A year later, pipelining and FIR filter design were used to create low power, high throughput multipliers and adders [4]. This design consisted of a 64-bit data bus width and cell area of 0.24µm² [4]. The most impressive results were derived from the use of 2-dimensional pipelining. Energy consumption was reduced by 62.5% in comparison to the original design [4]. Furthermore, this design showed a 51.4% improvement over the 1-dimensional pipeline technique [4]. Finally, in 2004, a 32-bit NIOS 2.0 ALU design was demonstrated using Altera’s Apex 20KE FPGA architecture [10]. With this updated architecture, only 1200 LEs were needed to complete its functions [10]. This architecture also permitted a running rate of 85MHz [10]. Results show that in comparison to the NIOS 1.1, size of the ALU has been decreased by 50% and performance has increased 70% [10]. The most popular adder/multiplier design of this time-frame is the two-staged pipeline design.

An ALU with dual supply voltages, 64-bit operands, carry merge adder design and 90 nm ITRS node technology was tested in 2005 [8]. This design team was able to conquer such achievements as low power leakage and power consumption in addition to increased ALU performance as high as 7GHz at 2.1V [8]. Total power consumption in this design was 300 mW [8]. In 2006, a 2-dimensional pipeline gating technique was used to decrease energy consumption in a 16-bit multiplier [5]. It was seen that a 65-66% reduction in power and 44-47% reduction in latency time had occurred [5]. By using this technique, power could be scaled down considerably at low input precision modes [5]. With only two designs for this era, it is difficult to say with adder/multiplier design was most favorable.

Jumping ahead, in 2014, an energy efficient multiplier using a Virtex 6 FPGA chip with 18nm² cell area and 32-bit operands was designed [3]. In order to increase efficiency, probabilistic domain transformation is used with the basis of Fourier transforms and digital signal processing [3]. Results have shown that this design has decreased the clock period two fold; however, the system is only about 110% faster than the base model due to large clock cycles being performed [3]. Also, this configuration achieved a 300 to 400% reduction in power consumption [3]. Overall, the total power dissipation and energy consumption was decreased 10 fold compared to the base model. A year later, in 2015, 15 and 45 nm ITRS nodes were compared for energy efficiency [1]. Due to lower threshold voltage, static power leakage, dynamic power consumption and wire area, an increase of 300 to 400% in energy efficiency and a reduction of cell area by 30% was seen using the 15 nm ITRS node in comparison to the 45 nm ITRS node [1]. That same year, a 16-bit RISC processor with a 16-bit datapath width and 28 nm ITRS node technology was used to test pipelining’s ability to decrease power [6]. Results displayed a power dissipation of 0.220 Watts and latency of 1.5 cycles [6]. Finally, a month later, a fault-tolerant QCA full adder was designed with 1-bit operands and 18 nm ITRS node technology [2]. Results showed that execution occurred after four clock cycles and “gate probability of failure” had decreased [2]. For this period of time, it was found that the full adder was the most popular adder design.

III. DATA ANALYSIS

Fig. 1. Overall, chip area had decreased from 1999 to 2006. The higher chip area seen had been used to increase ALU performance.

Fig. 2. Various sized operands have been used for different testing reasons. A general goal is to have larger sized operands produce high performance results. In some designs, such as the QCA adder, use smaller bit operands so as to only prove functionality.

Fig. 3. It can be seen here that ITRS Technology nodes have decreased considerable due to new photolithography technology.
IV. CONCLUSION

Metrics such as ITRS node technology traced a typical downward trend over the space of 2004 to 2016. Smaller ITRS node results in lower energy consumption, so it is a popular goal among researchers in computer science. Operands displayed a different, non-existent trend over the past fifteen years. This may be because different sized operands are used for different experimental goals. One example is a high performance adder with larger operands. It was found that by using intuitive designs and factors of performance and energy, metrics can be increased or decreased to benefit the design of an ALU. Full adders were a popular design to use for test and carry select and ripple carry adders were also seen in the use of ALU studies.

REFERENCES

<table>
<thead>
<tr>
<th>ALU or Floating Point Architecture Name</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 Model of Chip used</th>
<th>Energy/Power Consumption (W or J) else indicate “low” or “high”</th>
</tr>
</thead>
<tbody>
<tr>
<td>Energy and Area Analysis of a Floating-Point Unit [1]</td>
<td>32 bits (Operands)</td>
<td>N/A</td>
<td>N/A</td>
<td>IEEE-754 Single Precision</td>
</tr>
<tr>
<td>Ultra-area-efficient fault-tolerant QCA full adder [2]</td>
<td>1 bit (Operands)</td>
<td>Ultra-area-efficient fault-tolerant QCA full adder</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>High Throughput Power-Aware FIR Filter Based On Fine-Grain Pipelining Multipliers and Adders [4]</td>
<td>64 bits (Datapath Width)</td>
<td>High Throughput FIR Filter Adder</td>
<td>High Throughput FIR Filter Multiplier</td>
<td>N/A</td>
</tr>
<tr>
<td>Improving Power awareness Multipliers using 2-DimensionalPipeline Gateing and its Application on FIR Design [5]</td>
<td>16 bits (Operands)</td>
<td>N/A</td>
<td>2-D Pipelined Unsigned Array Multiplier</td>
<td>N/A</td>
</tr>
<tr>
<td>16 bit RISC Processor Using low Power Pipelining [6]</td>
<td>16 bits (Datapath width)</td>
<td>Carry Select Adder</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>ALU Design in CMOS for Low Voltage Operation [7]</td>
<td>4 bits (Operands)</td>
<td>Ripple Carry Adder</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>ALU with Dual Supply Voltages [8]</td>
<td>64 bits (Operands)</td>
<td>Carry Merge Adder</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Low Power 16 Bits ALU [9]</td>
<td>16 bits (Operands)</td>
<td>5ns Two-Stage Pipelined Adder</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>32-bit ALU for Programmable Logic [10]</td>
<td>32 bits (Datapath Width)</td>
<td>8.3ns Three-Staged Pipelined Adder</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>