ALU and Floating Point Optimization using Multi-stage Pipelining and Clock Gating

Jack Gray
Department of Electrical and Computer Engineering
University of Central Florida
Orlando, FL 32816-2362

Abstract—The objective of this paper is to analyze and compare the limiting factors in ALU design. The fundamental elements of an ALU are the input operands, Arithmetic and Logic Design, CPU schematics, and power dissipation. The time space trade off relates the rate instructions are executed to the ALU’s hardware component operational metrics. Time, in terms of throughput metric varies from 16-64 bit input operands and clock rates from 100MHz – 1THz. The processor units range in size to 1 μm – 1 nm. The relationship between ALU size and clock rate of operation causes variation of the energy used and power dissipated by the device. Some designs increased throughput using pipelining techniques in the arithmetic unit of the ALU, variations of multi-stage pipelining synchronized with clock gating allows for greater throughput and lower power dissipation.

Keywords—Arithmetic/Logic Unit, Power Dissipation, Clock rate, Pipelining, Clock gating

I. INTRODUCTION

The design of the ALU investigated manipulates operands using multi-stage pipelining techniques and clock gating. Pipelining makes all the addition and multiplications delay times the same, allowing for the instruction execution to run uninterrupted and in quick succession. Clock gating technique reduces the dynamic power dissipation by adding logic in the ALU to minimize clocks needed to carry out the arithmetic. The optimization of these arithmetic/logic operations minimizes the number of Boolean gates, thus minimizing the total energy used to execute each instruction.

An Arithmetic/Logic Unit is the hardware design in the processor that evaluates fundamental mathematic and shifting operations. These operations are known as arithmetic instructions, add, sub, logic instructions include, and or, nor, andi, sll, and srl. The internal components of the ALU are made of Boolean gates, multiplexers, decoders, and shift registers. To perform an instruction the ALU receives bits from the register file transmitted through the data path to be used as input operands. Additional bits are received from the function field of the instruction to specify the operation needed, this is the control path. The output is carried out in bits to the output data path.

Data bus width is the number of wires connect from the register to the ALU, fundamentally it is the number of bits being transferred across the data path. International Technology Roadmap of Semiconductors has standardized the measurement of the half-pitch length between the transistors. Execution time is time spent by the processor to carry out the instruction. Power dissipation is the energy consumed by the device measured through capacitance, voltage and frequency, and limited by the number of instructions and energy per instruction, energy consumption is limited by the efficiency of the design.

ALU Designs modifying gate assembly and clock rates from 1999 – 2015 are reviewed in the following section. Evaluation of this modification represents the tradeoff between power dissipation and throughput.

II. LITERATURE REVIEW

In 2003, 0.24 μm Static CMOS Logic was used to compare 1-D and 2-D pipelined techniques to reduce the dynamic power of operation [6]. Results have shown the parallelism for this Finite Impulse Response filter benefits from the pipelining, reducing power dissipation by 62.5%. A similar design manipulation of 2-stage pipelining architecture was used on a 0.6 μm Single-Poly Triple-Metal CMOS, in 1999 [19]. Figure 1 shows that optimizing the adder design architecture to minimize the area of logic gates, the power is reduced to 54 nW, given the operating frequency of 200 MHz.

Manipulation of individual arithmetic and logic sections by means of clock gating allows for improved overall energy efficiency without altering the clock rate. Energy efficiency of Arithmetic block, Multiplier DSP block, and MAC run on a coprocessor based on the Karatsuba algorithm. This 2009 study ran the Vedic MAC unit in 44.87 ns, generating comparably less power dissipation [11]. Similar to this approach, clock gating reduces the logic to clock ratio, reducing dynamic power. Executing the Complex Programmable Logic Device on a 90 nm Spartan-3 chip at varying frequencies of 100MHz to 1 THz [12]. The 2013 study’s results show the most efficient dynamic power dissipation was 1.725 W. In a study from the same year, clock gating was used in a Virtex-6 FPGA with varying frequencies to result 0.736 W of dynamic power [20]. The 64-bit ALU used 1-10 GHz frequency range to optimize this Ultra Scale FPGA. Virtex FPGA technology was used in a Fault Demolition study, using 32-bit operands at a clock rate of 108 MHz [7]. The digital signal processing block uses synchronous computation to increase throughput and dynamic power to 0.950 W. These studies represent how the space reduction of a processor by optimized logic and circuitry can decrease the power used by the device as seen in Figure 2.
Varying the design type, allowing from a majority of the processing to be executed through adder, multipliers, or floating point requires different size of processing chips. For a 16-bit reduced instruction set computation, the Xilinx Kintex chip size is 28 nm [10]. The power dissipation in minimal, 0.220 W for a processor running at 100 MHz. Reduction of the size of the CMOS transistors associates to lower power dissipation. The ITRS Technology node compares 15nm to 45 nm nodes to find the relative energy consumption. The results of this 2015 study show the 15 nm technology uses 4-fold less energy, 0.6340 mW [1]. Optimizing the design of a QCA full adder makes for an area efficient 18nm² cell area [2]. This 2015 study shows, improved latency, complexity and area metrics in comparison to previous analyzed full adder designs. Use of minimized CMOS technology can improve dynamic power used by the device. Through probabilistic domain transformation, a 40 nm Virtex 6 chip can be used at 250 MHz to exploit the energy efficiency of real time applications [3]. The study decreases dynamic power to a reasonable 166.63 nJ operating power consumption. Figure 3 follows the trend that through the different design optimization based on arithmetic/logic configurations, this causes minimal size of the processing unit, and thus diminishing the power dissipation.

III. DATA ANALYSIS

Figure 1. Improved pipelining techniques reduced the size need for the technology node.

Figure 2. Modifications to clock gating improved the efficiency of the processor. With the optimization of clock rate the dynamic power was reduced over time.

Figure 3. The simplification of the processing logic reduced the space needed for each node. The dynamic power shows a similar trend to the size relationship.

IV. CONCLUSION

A direct correlation between the size of the processing unit and power dissipation can be seen from the evaluation of these ALU and Floating Point designs. This relationship exemplifies the time space trade off when making the device energy efficient. To increase throughput without increasing energy consumptions the processing logic or clock rate must be optimized. Pipelining allows for the instruction execution to carry out in quick succession. This method synchronized with clock gating decreases the clock cycles need per instruction, thus decreasing CPI and increasing throughput. With an optimized logic schematic and minimal gates, the chip area decreases as well as power dissipation. Realization of alternative logic designs lead to improvements in all areas of the ALU and Floating Point metrics.

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 or 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>Adder</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>Multiplier</td>
<td>Ultra-area-efficient fault-tolerant QCA full adder</td>
<td>N/A</td>
</tr>
<tr>
<td>High Throughput Power-aware FIR Filter Design Based on Fine-Grain Pipelining Multipliers and Adders [6]</td>
<td>16-bit (Operands)</td>
<td>Time for Operation or Design Type</td>
<td>0.8 μs 2-D Finite Impulse Response Filter</td>
<td>0.8 μs 2-D Finite Impulse Response Filter</td>
</tr>
<tr>
<td>Fault Demotion Using Reconfigurable Slack [7]</td>
<td>32-bit (Operands)</td>
<td>Time for Operation or Design Type</td>
<td>N/A</td>
<td>9.26 μs DSP48 Block</td>
</tr>
<tr>
<td>Design and Analysis of 16 bit RISC Processor Using low Power Pipelining [10]</td>
<td>16-bit (Operands)</td>
<td>Design Type</td>
<td>10 μs Reduced Instruction Set Computing</td>
<td>N/A</td>
</tr>
<tr>
<td>High Speed Energy Efficient ALU Design [11]</td>
<td>64-bit (Operands)</td>
<td>Time for Operation or Design Type</td>
<td>44.87 ns MAC (Multiply-Accumulate)</td>
<td>44.87 ns Vedic Multiplier</td>
</tr>
<tr>
<td>Clock Gating Based Energy Efficient ALU Design and Implementation on FPGA [12]</td>
<td>8-bit ALU</td>
<td>Time for Operation or Design Type</td>
<td>10 μs – 1ps Clock Gated ALU</td>
<td>10 μs – 1ps Clock Gated ALU</td>
</tr>
<tr>
<td>Multi-Level Approaches to Low Power 16-bits ALU Design [19]</td>
<td>16-bit ALU</td>
<td>Time for Operation or Design Type</td>
<td>5 ns Two-Stage Pipelined Architecture</td>
<td>N/A</td>
</tr>
<tr>
<td>64-bit Green ALU Design Using Clock Gating Technique on Ultra Scale FPGA [20]</td>
<td>64-bit ALU</td>
<td>Time for Operation or Design Type</td>
<td>1 ns - .0001ns Clock Gating FPGA</td>
<td>N/A</td>
</tr>
</tbody>
</table>