
Abhijith Santhoshkumar
Department of Electrical and Computer Engineering
University of Central Florida
Orlando, FL 32816-2362

Keywords— ITRS, Data Bus Width, Energy Use, Clock Rate, Adder, Multiplier, Node Technology, Execution Time

I. INTRODUCTION

ALU stands for arithmetic logic unit. As insinuated by its name, the ALU handles any arithmetic or logic requirements needed by the processor. This includes basic arithmetic operations such as addition, multiplication, and Boolean instructions. The CPU gives the inputs to the ALU through the data buses. The size of the buses can vary greatly, based on the bit of the CPU. Along with the data, an instruction command is also passed. The ALU then computes this instruction on the given data and stores it in the register file.

The data bus width is what limits how much information can travel through the bus. An 8-bit wide bus allows values 8-bits wide while a 64-bit wide bus allows data of much larger complexities. The ITRS is the International Technology Roadmap for Semiconductors. It deals with the size of the nodes. No CPU has achieved complete efficiency. As it uses electricity, some of this energy is lost as heat. This is the power dissipation. The rest is used to switch parts of the CPU as needed. All this energy together is the energy consumption of the processor.

The upcoming sections of this paper contains three more sections. First is the literature review. This section will contain a review of ten computer systems with specific metrics. Second is the data analysis section. Here, there will be various charts based on data from Table 1 including an analysis of the trend shown in each of the charts. Finally, there is the conclusion section which contains a recap of this paper.

II. LITERATURE REVIEW

This section will begin by reviewing designs from the 2000-2004 range. During this range, the ripple carry adder seems to be the most prominent. There was also a large focus on the pipelining method. In 2001, a sub 500-ps 64-bit CMOS system was used to present an “SOI-optimal redesign of the ALU using a novel deep-stack quaternary-tree architecture” [17]. This study confirmed that the redesign offered a 14% performance increase [17]. Then in 2002, a study was done on “dual-rail pipelined Modified Baugh–Wooley MAC, which yielded a speedup of 2.5 over its initial non-pipelined version” [4]. And again, another study in 2003 was published based on fine-grain pipelining multipliers and adders. This study found that “by pipelining multipliers and adders, very high throughput can be achieved” [5].

Now in the date range of 2004-2009, there was once again a focus on pipelined designs. In 2006, a study on improving power awareness based on a pipelined design was published stating that “Simulation results show that an average power saving of 65-66% and latency reduction of 44-47% can be achieved for multipliers under equal input precision probabilities” [8]. In 2009, a study was published based on scalable architecture using Dynamic Partial Reconfiguration. They found that “zonal coding can be achieved by changing the number of the PEs, and full or reduced precision DCT computations can be achieved by changing the internal logics of the PEs.” [7]. This allowed the FPGA to continue working while the configuration was changed [7]. Going back to the pipelining method, a study in 2009 used a 4 level pipelining structure which implemented “each submission parallelism operation and enhance system performance of computer” [21]. Another study also in 2009 designed a system using feedback switch logic which achieved “14% reduction in delay but at the cost of 8% increased power consumption compared to static CMOS logic” [20].

Finally, in the range of 2009-2015, the carry skip was the most popular adder type. A study done in 2015 “presented an efficient layered structure for 1-bit full adder with high fault-tolerance attribute in quantum-dot cellular automata (QCA)” [2]. Another study in 2013 found that “On 1GHz, there is 66.93% less reduction in overall dynamic power of 64-bit ALU, when clock gate is added to the device” [19]. In order to improve energy efficiency, another study in 2013 found that “There is 67.04% dynamic power reduction with LVCMOS12 when we migrate from 90-nm Spartan-3 FPGA to 40-nm Virtex-6 FPGA” [15].

III. DATA ANALYSIS

The data gleamed from the table shows that there are direct correlations between the years and the specs of technology. The clearest example of this can be seen in Figure 3. Node Technology vs. Year. This figure shows a great example of
Moore’s Law. The size of the node technology has been decreasing at a relatively fast rate and can be seen clearly in the trend line.

Looking at Figure 1, it can be seen that the size of Data Bus Widths have been steadily increasing over time. This makes sense since systems have been dealing with more and more information as time passes.

From Figure 2, it would seem that systems are becoming less and less efficient as technology develops. However, this assumption would be incorrect. While it may in fact be using more energy, the efficiency is actually increasing greatly as the systems become more and more powerful and are able to do exponentially more tasks with about the same amount of energy.

Lastly, there is the very worrying trend of execution time vs. year in Figure 4. It seems like execution time is increasing as the years go by, but this information is not really accurate since the two systems being compared were built for two completely different purposes and should not really be compared.

IV. CONCLUSION

Based on all the preceding data, various trends of exponential scaling in digital technology can be confirmed. The size of nodes are heavily decreasing as theorized by Moore’s Law. Energy use compared to the amount of instructions processed is also greatly decreasing, even despite an increase in energy usage. The overall data bus width is also steadily increasing.

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>NULL Convention Multiply and Accumulate Unit with Conditional Rounding, Scaling, and Saturation 2002 [4]</td>
<td>32-bits (Operands)</td>
<td>Ripple-carry adder</td>
<td>12.7 ns</td>
<td>N/A</td>
</tr>
<tr>
<td>High Throughput Power-aware FIR Filter Design based on Fine-grain Pipeline Multipliers and Adders 2003 [5]</td>
<td>16-bits</td>
<td>revised FIR filter structure (pipelined scheme)</td>
<td>fine grained pipelined</td>
<td>N/A</td>
</tr>
<tr>
<td>A CPL-based Dual Supply 32-bit ALU for Sub 180nm CMOS Technologies 2004 [20]</td>
<td>32-bits (Operands)</td>
<td>Compound Domino Logic</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Design of a 1.7-GHz low-power delay-fault-testable 32-b ALU in 180-nm CMOS technology 2005 [16]</td>
<td>32-bits (Datapath)</td>
<td>radix-2, Han–Carlson architecture with a sparse carry-merge tree (CMT)</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>A 4-GHz 300-mW 64-bit integer execution ALU with dual supply voltages in 90-nm CMOS 2005 [14]</td>
<td>64-bits (Datapath)</td>
<td>compact energy-efficient 64-b adder core with single-cycle latency</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Improving Power-awareness of Pipelined Array Multipliers using 2-Dimensional Pipeline Gating and its Application to FIR Design 2006 [8]</td>
<td>4-bits (Operands)</td>
<td>Pipelined Adders</td>
<td>Pipelined Multipliers</td>
<td>N/A</td>
</tr>
<tr>
<td>Ultra-area-efficient fault-tolerant QCA full adder 2014 [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>Design &amp; analysis of 16 bit RISC processor using low power pipelining 2015 [9]</td>
<td>16-bits (Operands)</td>
<td>Carry Select Adder</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Energy and Area Analysis of a Floating-Point Unit 2015 [11]</td>
<td>32 bits (Operands)</td>
<td>N/A</td>
<td>N/A</td>
<td>IEEE-754 Single Precision</td>
</tr>
</tbody>
</table>