StrongARM SA110

A 160Mhz 32b 0.5W CMOS ARM Processor

Sribalan Santhanam
Digital Equipment Corporation
Hot Chips 1996
Overview

- Highlights
- Design choices
- μ Architecture details
- Powerdown Modes
- Measured Results
- Performance Comparisons
- Summary
Processor Highlights

◆ Target Market Segments
  – Embedded consumer applications
  – PDA’s, set-top boxes, Internet browsers

◆ Function
  – Implements ARM V4 Instruction set
  – Bus compatible with ARM 610, 710 and 810

◆ Performance
  – Record breaking performance/watt and price/performance
  – 160Mhz @ 1.65V delivers 185 Dhrystone MIPS at < 0.45W
  – 215Mhz @ 2.0V delivers 245 Dhrystone MIPS at < 0.9W
Processor Highlights

◆ Process
  – 2.5 Million transistors (2.2 Million in caches)
  – 3 Metal CMOS
  – $t_{ox}$ of 60 Å, $L_{EFF}$ of 0.25 micron, and $V_T$ of 0.35v

◆ Packaging
  – 7.8mm X 6.4mm -> 50mm$^2$
  – 144 pin plastic TQFP
StrongARM Design Choices

- Chose a simple design with low latency functional units to fit portable power budgets
  - Simple single issue 5 stage pipeline
  - Long tick model, low latency
  - Could have pipelined deeper for faster cycle time but would have exceeded the power budget
  - Could have gone superscalar but that would have increased control logic cost and power and increased per cycle memory interface needs
  - Would have increased design time
StrongARM Design Choices

◆ **Power reduction**
  – Run core at low voltage and I/O at standard voltage
  – Scale technology
  – Only run logic section needed in a cycle
    • Conditional clocking: Only drive clocks to sections running
    • Edge triggered flip flops allowed reduction in number of latches

◆ **Result**
  – Best Mips/Watt in the industry
  – A core voltage of 1.65V yields 411 Mips/Watt @160MHz
Power Reduction Factors

Start with Alpha 21064: 200Mhz @ 3.45V : Power 26W

- VDD reduction: Power Reduction: 4.4x to 5.9W
- Reduce functions: Power Reduction: 3x to 2.0W
- Scale Process: Power Reduction: 2x to 1.0W
- Clock Load: Power Reduction: 1.5x to 0.6W
- Clock Rate: Power Reduction: 1.25x to 0.5W
**StrongARM μArchitecture Highlights**

- Simple 5 stage pipeline
- Early branch execution
- Integer datapath with single cycle shift and add
- 5 register file ports
- High performance integer multiplier
- Split I/D caches
- Asynchronous and Synchronous bus interface
SA110 Block Diagram
**Basic Pipeline**

<table>
<thead>
<tr>
<th>Instruction</th>
<th>F</th>
<th>I</th>
<th>E</th>
<th>B</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>100: ADDS R1,</td>
<td>pc&lt;100</td>
<td>Read rm,rn</td>
<td>w&lt;- rn+rm</td>
<td>w' &lt;- w</td>
<td>R1&lt;-w'</td>
</tr>
<tr>
<td></td>
<td>ib&lt;- ADDS</td>
<td></td>
<td>cc&lt;-alu.cc</td>
<td></td>
<td></td>
</tr>
<tr>
<td>104: LDR R0, [R1,d)!</td>
<td>pc&lt;104</td>
<td>Read Rm, Rn</td>
<td>w,la&lt;- d+R1</td>
<td>L&lt;- mem(la)</td>
<td>R0&lt;- L</td>
</tr>
<tr>
<td></td>
<td>ib &lt;- LDR</td>
<td></td>
<td>w' &lt;- w</td>
<td>R1&lt;-w’</td>
<td></td>
</tr>
<tr>
<td>108: SUB x,R0</td>
<td>pc&lt;108</td>
<td>Read RM, RN</td>
<td>w&lt;-R0-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ib &lt;- SUB</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10C: xxx</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Basic 5 stage pipeline**
- Instruction Fetch (F)
- Register read and Issue checks (I)
- Execute/Effective Address (E)
- Buffer and cache access (B)
- Register file Write (W)

Arrows show data forwarding paths
### Branch Example

<p>| | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>F</td>
<td>I</td>
<td>E</td>
<td>B</td>
<td>W</td>
<td></td>
</tr>
<tr>
<td>104: BLEQ 200</td>
<td>pc&lt;-104</td>
<td>pc+d &lt;200</td>
<td>w&lt;- pc-4</td>
<td>w&lt;- w</td>
<td>R14 &lt;- w'</td>
<td></td>
</tr>
<tr>
<td>108: xxx</td>
<td>pc&lt;-108</td>
<td>ib&lt;- xxx</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>200: yyy</td>
<td>pc&lt;-200</td>
<td>ib&lt;- yyy</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Branches performed in Issue stage
- Condition codes can be bypassed from an instruction in Execute stage
- Only one cycle lost for branch taken
**Multiplier**

- Perform signed and unsigned multiply and multiply accumulate producing 32 or 64 bit results.
  - $32b \times 32b \rightarrow 64b$
  - $32b \times 32b + 64b \rightarrow 64b$
  - $32b \times 32b \rightarrow 32b$
  - $32b \times 32b + 32b \rightarrow 32b$

- Multiply accumulate folds accumulate into mul array
- Multiplier array retires 12 bits of the multiplier per cycle
- Early out for short multipliers
- Multiplier adder produces 32 bits of result per cycle
- Latency of 2-4 cycles for 32 bits and 3-5 cycles for 64 bits
StrongARM Caches

- **Separate 16KB I and D caches**
  - 32 byte block 32-way set associative
  - Dcache is writeback with no write allocation
  - Cache tags are virtual addresses
  - Dcache stores physical address with cache line.
  - Caches occupy half of total chip area
  - Self-timed to save power
Why a 32-way associative cache?

– Wanted Associativity of at least 4-way
– Wanted to minimize power so cache was divided into 8 banks so that only one eighth of the cache was enabled per access.
– Required Single cycle access for read and writes
– Our implementation provided a 32-way associative for free meeting the above criteria.

**Writes are done in single cycle same as reads**

– removes need for buffer and tag for read after write hazard
– read done before write
– Memory management exceptions writeback original data
MMUs and Write Buffer

Memory Management Units

- Separate I and D MMU's, each with 32 fully associative entries which can map 4KB, 64KB or 1MB page
- ARM architecture includes extensions to memory management protection for efficient support of object oriented systems.
  - Additional checks must be performed in series with TLB lookup
  - Self timing required to perform lookup and protection checks in one cycle

Write Buffer

- Eight 16 byte entries with single entry merge buffer
Other StrongARM SA110 Features

- **Shift + ALU operation in a single cycle**
  - Provides low latency with simple control logic
- **Shifter bypassed for shifts by zero**
  - Provides power savings when shifter not needed
- **MOV PC, Rx executed in Issue stage**
  - Provides low latency for subroutine returns
StrongARM SA110 System Interface

◆ Clocking
  – 3.68 MHz input clock multiplied by on-chip low power PLL
    • Generates 16 frequencies from 88MHz to 287MHz
    • Power dissipation = 1.5mW
  – System interface clock can
    • Either be driven by the core at 1/2 to 1/9 of the core frequency
    • Or be driven by the system at any frequency up to 66MHz

◆ System interface compatible with existing ARM parts
  – Separate 32-bit address and data bus
  – Enhancements to current ARM bus provides for wrapped reads to return critical word first and general write merging
Powerdown Modes

**Idle Mode**
- For short periods of inactivity with quick restart
- Clock trees and all local clocks stop
- PLL continues to run
- Power consumption limited to 20mW

**Sleep Mode**
- For extended periods of inactivity
- Core power supply turned off
- I/O remains powered and maintain bus state
- Standby current limited to 50µA
Measured Results

- Running Dhrystone on StrongARM in evaluation board. MCLK frequency = 1/3 PLL frequency.
- Dhrystone fits in cache so internal clocks are running at full speed.
- Measurements taken for I/O Vddx = 3.3v and core Vdd = 1.65v and 2.0v
  - For Vdd = 1.65v, Total power = 2.54mW/MHz -> 254mW @ 100MHz & 406mW @ 160MHz
  - For Vdd = 2.0v, Total power = 3.3mW/MHz -> 528mW @ 160MHz & 710mW @ 200MHz
- Typical power much less on more realistic applications
Simulated Power Breakdown

<table>
<thead>
<tr>
<th>Component</th>
<th>Power (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICACHE</td>
<td>27%</td>
</tr>
<tr>
<td>IBOX</td>
<td>18%</td>
</tr>
<tr>
<td>DCACHE</td>
<td>16%</td>
</tr>
<tr>
<td>CLOCK</td>
<td>10%</td>
</tr>
<tr>
<td>IMMU</td>
<td>9%</td>
</tr>
<tr>
<td>EBOX</td>
<td>8%</td>
</tr>
<tr>
<td>DMMU</td>
<td>8%</td>
</tr>
<tr>
<td>WRITE BUFFER</td>
<td>2%</td>
</tr>
<tr>
<td>BIU</td>
<td>2%</td>
</tr>
<tr>
<td>PLL</td>
<td>&lt; 1%</td>
</tr>
</tbody>
</table>
StrongARM SA110 Die photo
Performance Comparison

![Performance Comparison Diagram](image-url)
StrongARM SA110 Summary

◆ Function
  – Implements ARM Version 4 instruction set
  – Bus Compatible with ARM 610, 710, and 810

◆ Performance
  – Best performance/watt and price/performace
  – 160MHz @ 1.65v -> 185 Dhrystone MIPS at < 0.45W
  – 215MHz @ 2.0v -> 245 Dhrystone MIPS at < 0.9W

◆ Process and Package Technology
  – 2.5 million transistors fabricated in 0.35 μm 3 metal CMOS with 0.35v $V_T$ and 0.25μm $L_{EFF}$
  – Die size: 50mm$^2$ in a 144 pin plastic TQFP