The Thunder SPARC Processor

HOT Chips VI

August 15-16, 1994

Bruce D. Lightner
Vice President, Development

Metaflow Technologies, Inc.
4250 Executive Square, Suite 300
La Jolla, California 92037
(619) 452-6608
Internet: lightner@metaflow.com
Metaflow Background

- Company founded in 1987
- Pioneers in out-of-order/speculative execution microprocessor design
- First major design contract was LSI Logic's “Lightning” SPARC chipset (completed 1991)
- “Thunder” SPARC 3-chip set is improved version of “Lightning” 4-chip set
- Project 100% funded by Hyundai Electronics
- Thunder chipset first silicon July 1993
- Patented out-of-order architecture
Thunder SPARC Mbus Module

- **Integer Unit (IU)**
  - 2.7M

- **Floating Point Unit (FPU)**
  - 0.8M

- **Cache Control/MMU/Bus Intf (CMB)**
  - 2.5M

- **Xcache RAM (1 Mbyte)**

Connections:
- FPU Instrs
- Bypass
- Address
- Data
- Mbus (multiprocessor)
- 85
- 36
- 32
- 64
- 17

The diagram illustrates the integration of various components in a SPARC Mbus Module, highlighting the flow of information and resources through the system.
Thunder SPARC Chipset

- Superscalar fetch, issue, and execution
- Dynamic scheduling (dataflow)
- Out-of-order execution
- Speculative execution
- Above “hidden” from the programmer
- Factor of 2-3 performance advantage from architecture
Microprocessor Hardware Trends

- Superscalar instruction issue
- Super-pipelined execution units
- Super-fast processor clocks (>100 MHz)
- Out-of-order execution
- Speculative execution
Out-of-Order Execution

**PROGRAM ORDER**

```c
float A, X, XX, X1
int n
.
.
.
X1 = 1 / A;
XX = A * A;
n = n + 1;
X = X + A;
.
.
.
.
```

**COMPLETION ORDER**

```c
.
.
.
.
n = n + 1;
X = X + A;
XX = A * A;
X1 = 1 / A;
.
.
.
.
```

<table>
<thead>
<tr>
<th>Operation</th>
<th>Clocks</th>
</tr>
</thead>
<tbody>
<tr>
<td>FP divide</td>
<td>10</td>
</tr>
<tr>
<td>FP multiply</td>
<td>5</td>
</tr>
<tr>
<td>FP add</td>
<td>3</td>
</tr>
<tr>
<td>Integer add</td>
<td>1</td>
</tr>
</tbody>
</table>
Speculative Execution

\[ \sum_{i=0}^{n} \sqrt{X_i} \]

**C EXAMPLE**

```c
for ( i = 0; i < MAX; ++i ) {
    if (X[i] < 0) error_exit();
    sum = sum + sqrt(X[i]);
}
```

**FORTRAN EXAMPLE**

```fortran
DO 100 i = 1, max
    IF (X(i) .LT. 0) CALL error_exit
100  SUM = SUM + SQRT(X(i))
```
Conventional Pipeline

FETCH

ISSUE

EXECUTE

RETIRE

Instruction Cache

Register File

 METAFLOW
Metaflow Pipeline (DRIS)

Instruction Cache

ISSUE

INSTRUCTION n
INSTRUCTION 2
INSTRUCTION 1

DRIS
Deferred-scheduling, Register-renaming
Instruction Shelf

RETIRE

Register File

UPDATE

SCHEDULE
# Metaflow DRIS Entry

<table>
<thead>
<tr>
<th>Source Operand 1</th>
<th>Source Operand 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Locked</td>
<td>Locked</td>
</tr>
<tr>
<td>RegNum</td>
<td>RegNum</td>
</tr>
<tr>
<td>ID</td>
<td>ID</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latest</td>
</tr>
<tr>
<td>RegNum</td>
</tr>
<tr>
<td>Instruction’s Results</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dispatched</th>
<th>Functional Unit</th>
<th>Executed</th>
<th>Program Counter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes/No</td>
<td>Class Number</td>
<td>Yes/No</td>
<td>Content</td>
</tr>
</tbody>
</table>
Thunder IU

**Predecoer**
- Out-of-order instruction results
- Speculative instruction results
- Automatic “register renaming”

**Instruction Cache**

**Branch Unit**

**Branch Shelf**

**INSTRUCTION n**
- ...
- INSTRUCTION 2
- INSTRUCTION 1

**DRIS**
- Out-of-order instruction results
- Speculative instruction results
- Automatic “register renaming”

**ISSUE**

**SCHEDULE**
- Addr
- +
- + * /

**Branch Unit**

**Register File**

**Predecoer**

**Condition Codes**

**Data Bus**

**Bypass Bus**

**Memory Shelf**

**UPDATE**

**Store Data**

**Address Bus**
Thunder Processor
Distinguishing Features

● 4 instructions issued per clock (3 integer, 2 floating point instructions and one branch)

● 8 parallel functional units (3 integer ALUs, 2 FP ALUs, 1 branch unit, 2 memory/bypass)

● Dataflow-based out-of-order instruction issue/execution (memory/ALU operations), in-order completion, precise traps and interrupts

● Speculative execution beyond unresolved conditional branches (multiple basic blocks with instant state repair on error)

● Above mechanisms transparent to executing program (strict SPARC V8 compatibility)
Thunder Unusual Features

- Eager evaluation ("folds" conditional branches)
- Multiple instances of memory locations ("memory renaming")
- Out-of-order memory references
  - Multiple simultaneous cache miss processing ("non-blocking cache")
  - Split address and data memory transactions (memory response re-ordering allowed)
- Speculative memory reads with respect to shared memory coherence restrictions (emulates "strong consistency" model)
- Uninterrupted transitions between user and supervisor modes (traps/interrupts)
Thunder Unusual Features (continued)

Floating Point Unit (FPU)

- Variable latency FPU (early termination for divide/square root)
- Non-blocking divide and square root (concurrent add/multiply)

Branch Prediction

- Dynamic branch prediction
- Generalized loop count prediction
- Return-from-subroutine (RET) “folding”
- Jump-through-register prediction (JMP L cache)
<table>
<thead>
<tr>
<th>Thunder Status</th>
<th>Thunder 1.0</th>
<th>Thunder 1.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>First Silicon</td>
<td>June/Nov 1993</td>
<td>Q4 1994</td>
</tr>
<tr>
<td>Foundry</td>
<td>VLSI Technologies</td>
<td>0.5 µm</td>
</tr>
<tr>
<td>CMOS technology</td>
<td>0.6/0.8 µm</td>
<td>4</td>
</tr>
<tr>
<td>Metal layers</td>
<td>3</td>
<td>3.6V</td>
</tr>
<tr>
<td>Supply voltage</td>
<td>5V</td>
<td>80 MHz</td>
</tr>
<tr>
<td>Clock rate (processor)</td>
<td>50 MHz</td>
<td>≤ 60 MHz</td>
</tr>
<tr>
<td>(Mbus)</td>
<td>40 MHz</td>
<td>9.1/11.8</td>
</tr>
<tr>
<td>Die Size (mm square)</td>
<td>14.5/17.4</td>
<td>TBGA (600+ pins)</td>
</tr>
<tr>
<td>Package</td>
<td>IPGA (391 pins)</td>
<td>Full custom</td>
</tr>
<tr>
<td>Methodology (data path/RAM)</td>
<td>Full custom</td>
<td>Standard cell</td>
</tr>
<tr>
<td>(control logic)</td>
<td>Standard cell</td>
<td>JTAG/full scan</td>
</tr>
<tr>
<td>(test)</td>
<td>“Viking”</td>
<td>“Pentium”</td>
</tr>
<tr>
<td>External cache RAM (type)</td>
<td>Eight 128K x 9</td>
<td>Eight 64K x 18</td>
</tr>
<tr>
<td>(size)</td>
<td>~6 million</td>
<td>~6 million</td>
</tr>
<tr>
<td>Total transistors (3 chips)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Thunder Status

<table>
<thead>
<tr>
<th>Feature</th>
<th>Thunder 1.0</th>
<th>Thunder 1.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Foundry</td>
<td>First Silicon</td>
<td>Q4 1994</td>
</tr>
<tr>
<td>Technology</td>
<td>VLSI Technologies</td>
<td>IBM</td>
</tr>
<tr>
<td>Metal layers</td>
<td>0.6/0.8 µm</td>
<td>0.5 µm</td>
</tr>
<tr>
<td>Supply voltage</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Clock rate (processor) (Mbus)</td>
<td>5V</td>
<td>3.6V</td>
</tr>
<tr>
<td>Die Size (mm square)</td>
<td>50 MHz</td>
<td>80 MHz</td>
</tr>
<tr>
<td>Package</td>
<td>40 MHz</td>
<td>≤ 60 MHz</td>
</tr>
<tr>
<td>Methodology (data path/RAM)</td>
<td>14.5/17.4</td>
<td>9.1/11.8</td>
</tr>
<tr>
<td>(control logic)</td>
<td>IPGA (391 pins)</td>
<td>TBGA (600+ pins)</td>
</tr>
<tr>
<td>(test)</td>
<td>Full custom</td>
<td>Full custom</td>
</tr>
<tr>
<td>External cache RAM (type) (size)</td>
<td>Standard cell</td>
<td>Standard cell</td>
</tr>
<tr>
<td>Total transistors (3 chips)</td>
<td>Eight 128K x 9</td>
<td>Eight 64K x 18</td>
</tr>
<tr>
<td></td>
<td>~6 million</td>
<td>~6 million</td>
</tr>
</tbody>
</table>
Thunder Performance Goals

- SPECint92
- SPECfp92

- Q1’94
  - 50 MHz 0.8/0.6 µm
  
- Q4’94
  - 80 MHz 0.5 µm
  
- Q2’95
  - 120 MHz 0.5 µm
Thunder 1.0 IU Floor Plan
Thunder 1.0 IU Floor Plan

Instruction Cache (20 Kbytes)

Load Buffer
Data TLB
Retirement
Result Shelf

Register File
ALUs

Load Shelf
Load/Store Logic

Instruction Shelf
Register Renaming

Store Shelf
Instr TLB
Physical Tags

Instr Pre-Dec

Load/Store Depend Check

Instr Issue Logic
Instr Decode

Virtual Tags
Branch Predict
Branch Shelf

PC Generator
Ret Addr Cache
Backup State