The HP PA-8000 RISC CPU

A High Performance
Out-of-Order Processor

Hot Chips VIII
IEEE Computer Society
Stanford University
August 19, 1996

Ashok Kumar
Hewlett-Packard Company
Engineering Systems Lab - Fort Collins, CO
Systems Performance Lab - Cupertino, CA
ashok@cup.hp.com
Presentation Overview

* Design Objectives
* Hardware Highlights
* Chip Statistics
* Performance
* IRB Design
Design Objectives

* Leadership Performance
* Full Support for 64-bit Applications
  - New PA 2.0 Architecture
  - Binary compatibility with existing code
* Glueless Support for up to 4-way MP
PA-RISC 2.0 Enhancements

* New 64 Bit Architecture
  - Wider Registers
  - New computational units
  - Virtual addressing
  - Physical addressing
* Fast TLB insert instructions
* Load/Store instructions with 16-bit displacement
* Memory prefetch instructions
* Variable sized pages
* Multimedia half-word instructions
* Branch with 22-bit displacement, short pointer
* Branch prediction hinting
* Floating point multiply-accumulate
* FP multiple compare result bits
* Carefully selected others
Application Performance

In order to achieve sustained performance on large applications one needs:

* Large Primary Caches

* Methods to hide Memory Latency
  Dynamic Instruction Reordering

* High Bandwidth System Bus
  RUNWAY: 768 MB/sec
  Split Transaction
  Incorporates support for multiple outstanding memory requests
Hardware Highlights

* Completely redesigned core/new microarchitecture
* 56 Entry Instruction Reorder Buffer (IRB)
* Peak execution rate of 4 instructions/cycle
* 8 Computational Units
  - FPMAC: 3 cycle latency, fully pipelined
  - DIV/SQRT: 17 cycle latency, not pipelined
  - all others: single cycle latency
* 2 Load/Store Units
* 32 Entry Branch Target Address Cache (fully associative)
  - Zero state taken branch penalty for branches that hit in BTAC
* Branch Prediction Hardware
  - 256 Entry Branch History Table
  - Static or Dynamic Prediction
Cache Design

* No on-chip cache
* Single Level off-chip
* Split Instruction/Data up to 4M/4M
* Direct Mapped
* Uses industry standard synchronous SRAMs
* Two state pipelined access
Chip Statistics

- Fabricated in HP's 0.5 micron, 3.3V CMOS Process
  - 0.28 \text{um } L_{eff}
  - 5 metal layers
- Die size:
  - 17.68 mm x 19.1 mm
- Transistor Count:
  - 3.8 million
- Flip-Chip Packaging Technology
  - 704 signals, 1,200 Power/Ground bumps
  - 1,085 pin package
  - Ceramic Land Grid Array
Die Photo
Performance

At 180 MHz:

11.8 Spec Int 95
20.2 Spec FP 95

Currently in production

Systems are shipping!
Performance Enablers

* Large number of functional units
* Aggressive Out-of-Order Execution
  - Robust dependency tracking
  - Large window of available instructions
* Explicit Hinting from Compiler
  - Data Prefetch
  - Branch Prediction
* High Performance Bus Interface

Sustained superscalar operation
Effect of Instruction Reordering

Efficiency (SPECint95 / MHz x 1000)

- All examples are 4-way Superscalar designs except Pentium Pro which is 3-way

- Sun UltraSPARC (167 MHz)
- Digital 21164 (400 MHz)
- IBM PPC 620 (166 MHz)
- IBM PPC 604e (166 MHz)
- MIPS R10000 (200 MHz)
- Intel Pentium Pro (200 MHz)
- H-P PA-8000 (180 MHz)

Source: Microprocessor Report 4/15/96
Instruction Reorder Buffer

* 56 entries, split into ALU/FP IRB and MEM IRB

* Reorders instructions on the fly

* Tracks all dependencies between instructions

* Tracks branch prediction status
  Capable of flash invalidating all instructions that were incorrectly fetched.

* Consists of 850K transistors and consumes 20% of Die Area
Block Diagram of IRB

- ALUs
- SMUs
- MACs
- DIVs

ALU

MEM

SORT

LOAD STORE UNITS

2 inst

4 inst

4 inst

2 inst
Instruction Insertion

* In Order
* Fetch any mix of four instructions/cycle
* Routed to appropriate portion of IRB
* Branches stored in both ALU and MEM IRB
* Instructions with two targets (such as LDWM) split into two parts
Instruction Launch

* Out of Order
* Oldest even and oldest odd instruction from each segment of IRB with all dependencies cleared is allowed to execute
* 4 instructions maximum
* Results stored in associated rename register for each entry
Instruction Retire

* In Order
* Up to two ALU/FP instructions and two MEM instructions each cycle
* Results moved from RRs to GRs/PSW
  Allows for precise exceptions
Dependency Tracking

All possible instruction dependencies are identified at INSERT time.

- Operand
- Carry Borrow (CB)
- Shift Amount Register (SAR)
- Control (CTL)
- Nullify
- Address
  - Handled by separate ARB unit that maintains state information about pending loads and stores.
- Many others . . .
Operand Dependencies

★ Occur when source data of one instruction is the result of an earlier instruction.

★ Most Recent Writer of Source data determined at insert time utilizing a two-pass mechanism.

★ High Performance Broadcast mechanism.

★ Upon launch, an IRB entry broadcasts its slot number to all other entries in the IRB. If a later instruction's source tag matches that driven on the launch bus, the dependency has cleared.

★ Dependent instructions can launch very next cycle after a producer instruction executes.

★ The IRB also sends information to the functional units about where its source data should come from (RRs, bypass, etc.) and where the results should be stored.
Carry Borrow Dependencies

★ Occur when an instruction uses CB bits of the Processor Status Word.
★ Most recent IRB entry passes information to incoming instructions regarding whether there is an instruction prior to it that sets CB bits.
★ An instruction is aware it has a dependency, but does not know which instruction it is dependent on until its dependency has been cleared.
★ Complex control
Propagation System
Tags travel up to two IRB entries/cycle

Each IRB entry can:
Block tag bus - if instruction writes CB bits and has not executed yet
Drive Tag Bus - when an instruction that writes CB bits launches
Pass tags from previous entry - if instruction does not write CB bits

Trade-off Increased Latency for Area Savings
In common case where an instruction that uses CB information immediately follows the setting instruction, there is no performance impact.
Conclusion

The HP PA-8000 RISC CPU delivers high performance by:

- Aggressive Out-of-Order Execution
- Intelligent design choices
- Effective balancing of hardware to prevent bottlenecks
Acknowledgement

The author would like to recognize the contributions of the entire processor design team from HP's Engineering Systems Lab in Fort Collins, Colorado.