IA-64 Microarchitecture --- Itanium Processor

By
Subin kizhakkeveettil
CS B-S5....no:83
INDEX

• INTRODUCTION
• ARCHITECTURE
• MEMORY ARCH:
• INSTRUCTION FORMAT
• INSTRUCTION EXECUTION
• PIPELINE STAGES:
• FLOATING POINT PERFORMANCE
• INTEGER PERFORMANCE
• CONCLUSION
• REFERENCES
Itanium Processor

• First implementation of IA-64
• Compiler based exploitation of ILP
• Also has many features of superscalar
INTRODUCTION

• **Itanium** is the brand name for 64-bit Intel microprocessors that implement the **Intel Itanium architecture** (formerly called **IA-64**).

• *Itanium*'s architecture differs dramatically from the **x86** architectures (and the **x86-64** extensions) used in other Intel processors.

• The architecture is based on explicit **instruction-level parallelism**, in which the **compiler** makes the decisions about which instructions to execute in parallel.
Memory architecture

- From 2002 to 2006, Itanium 2 processors shared a common cache hierarchy. They had 16 KB of Level 1 instruction cache and 16 KB of Level 1 data cache. The L2 cache was unified (both instruction and data) and is 256 KB. The Level 3 cache was also unified and varied in size from 1.5 MB to 24 MB. The 256 KB L2 cache contains sufficient logic to handle semaphore operations without disturbing the main arithmetic logic unit (ALU).

- Main memory is accessed through a bus to an off-chip chipset. The Itanium 2 bus was initially called the McKinley bus, but is now usually referred to as the Itanium bus. The speed of the bus has increased steadily with new processor releases. The bus transfers 2x128 bits per clock cycle, so the 200 MHz McKinley bus transferred 6.4 GB/s and the 533 MHz Montecito bus transfers 17.056 GB/s.
IA-64 Instruction Format

Note: Exact arrangement of fields within bundles and instructions is unknown. Intel and HP might divulge additional fields later.

128-bit LIW (long instruction word) bundle

T  Instruction 1  Instruction 2  Instruction 3

Template field identifies which instructions in this bundle and in following bundles the CPU can execute in parallel.

IA-64 instruction

Opcode field  PR field  GPR field  GPR field  GPR field

Instructions can be out of order, and they can originate from different paths of a branch.

IA-64 instructions are fixed length, about 40 bits long.

6-bit field defines 64 predicate registers (see text).

7-bit fields define 128 general-purpose registers (128 integer, 128 floating-point).

IA-64 packs three fixed-length instructions into each 128-bit bundle.
Instruction execution

• Each 128-bit instruction word contains three instructions,
• When the compiler can take maximum advantage of this, the processor can execute six instructions per clock cycle.
• The processor has thirty functional execution units in eleven groups.
The execution unit groups include:

- Six general-purpose ALUs, two integer units, one shift unit
- Four data cache units
- Six multimedia units, two parallel shift units, one parallel multiply, one population count
- Two floating-point multiply-accumulate units, two "miscellaneous" floating-point units
• Three branch units

• The compiler can often group instructions into sets of six that can execute at the same time.

• The floating-point units implement a multiply-accumulate operation, a single floating point instruction can perform the work of two instructions when the application requires a multiply followed by an add: this is very common in scientific processing.
Pipeline Stages

• It has a 10-stage pipeline..
• The most important stages are given below..

• Front-end
• Instruction delivery
• Operand delivery
• Execution
Front-end

• Prefetches up to 32 bytes per cycle (2 bundles) into a prefetch buffer (up to hold 8 bundles)

• Branch prediction is done using a multilevel adaptive predictor
Instruction delivery

• Distributes up to 6 instructions to the 9 functional units
• Implements registers renaming for both rotation and register stacking.
Operand delivery

- Accesses the register file
- Performs register bypassing
- Accesses and updates a register scoreboard
- Checks predicate dependences.
Execution

• Executes instructions through ALU and load or store units
• Detects exceptions.
• Retires instructions and performs write-back
10 Stage In-Order Core Pipeline

Front End
- Pre-fetch/Fetch of up to 6 instructions/cycle
- Hierarchy of branch predictors
- Decoupling buffer

Execution
- 4 single cycle ALUs, 2 Id/str
- Advanced load control
- Predicate delivery & branch
- Nat/Exception/Retirement

Instruction Delivery
- Dispersal of up to 6 instructions on 9 ports
- Reg. remapping
- Reg. stack engine

Operand Delivery
- Reg read + Bypasses
- Register scoreboard
- Predicated dependencies
FLOATING POINT PERFORMANCE

- Itanium is **quicker** than Alpha 21264 and Pentium 4.
- **108%** of P4, **120%** of Alpha

**Itanium**: HP rx4610, 800MHz, 4MB off-chip, L3 cache

**Alpha 21264**: Compaq GS320, 1GHz, on-chip L2 cache

**Pentium 4**: Compaq Precision 330, 2GHz, on-chip L2 cache
Integer Performance

- Itanium is **considerably slower** than Alpha 21264 and Pentium 4.
- Only: **60%** of of P4, **68%** of Alpha

**Itanium**: HP rx4610, 800MHz, 4MB off-chip L3 cache

**Alpha 21264**: Compaq GS320, 1GHz, on-chip L2 cache

**Pentium 4**: Compaq Precision 330, 2GHz, 256KB on-chip L2 cache
Conclusion

- Large code size
- Only static instruction-level parallelism
- Cannot manage cache misses/hits flexibly
- Lack of applications
- Good floating point performance
- Poor integer performance
- Overall: not so good as Intel has advertised
REFERENCES:
www.intel.com
THANK YOU