CS152 – Computer Architecture and Engineering
Lecture 19 – Advanced Pipelining: Pentium III & 4, AMD Athlon & Opteron, VLIW and Itanium I & II

2003-10-30

Dave Patterson
(www.cs.berkeley.edu/~patterson)

www-inst.eecs.berkeley.edu/~cs152/

Review 1/2

- Reservations stations: renaming to larger set of registers + buffering source operands
  - Prevents registers as bottleneck
  - Avoids WAR, WAW hazards of Scoreboard
  - Allows loop unrolling in HW
- Not limited to basic blocks (integer units gets ahead, beyond branches)
  - Dynamic hardware schemes can unroll loops dynamically in hardware
  - Dependent on renaming mechanism to remove WAR and WAW hazards
- Helps cache misses as well

Review 2/2

- Reorder Buffer:
  - Provides generic mechanism for “undoing” computation
  - Instructions placed into Reorder buffer in *issue order*
  - Instructions exit in same order – providing *in-order-commit*
  - Trick: Don’t want to be canceling computation too often!
- Branch prediction important to good performance
  - Depends on ability to cancel computation (Reorder Buffer)
- Explicit Renaming: more physical registers than ISA.
  - Separates *renaming* from *scheduling*
    - Opens up lots of options for resolving RAW hazards
  - Rename table: tracks current association between architectural registers and physical registers
  - Potentially complicated rename table management
- Parallelism hard to get from real hardware beyond today

Review: Road to Faster Processors

- Time = Instr. Count x CPI x Clock cycle time
- How get a shorter Clock Cycle Time?
- Can we get CPI < 1?
- Can we reduce pipeline stalls for cache misses, hazards, … ?
- IA-32 P6 microarchitecture (μarchitecture): Pentium Pro, Pentium II, Pentium III
- IA-32 “Netburst” μarchitecture (Pentium 4, …
- IA-32 AMD Athlon, Opteron μarchitectures
- IA-64 Itanium I and II microarchitectures

Dynamic Scheduling in Pentium Pro, II, III

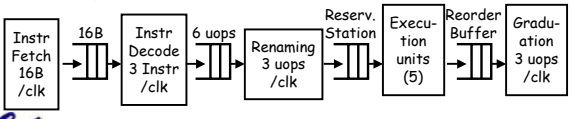
- P6 doesn’t pipeline 80x86 instructions
- P6 decode unit translates the Intel instructions into 72-bit "micro-operations" (~ MIPS instructions)
- Takes 1 clock cycle to determine length of 80x86 instructions + 2 more to create the micro-operations
- Most instructions translate to 1 to 4 micro-operations
- Sends micro-operations to reorder buffer & reservation stations

Dynamic Scheduling in P6 (Pentium Pro, II, III)

- Complex 80x86 instructions are executed by a conventional microprogram (8K x 72 bits) that issues long sequences of micro-operations
- 10 stage pipeline for micro-operations
- 14 clocks in total pipeline

P6 Pipeline

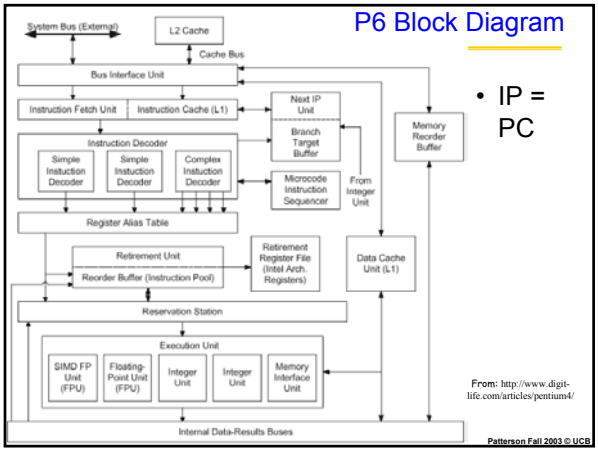
- 14 clocks in total (~3 state machines)
- 8 stages are used for in-order instruction fetch, decode, and issue
  - Takes 1 clock cycle to determine length of 80x86 instructions + 2 more to create the micro-operations (uops)
- 3 stages are used for out-of-order execution in one of 5 separate functional units
- 3 stages are used for instruction commit



Dynamic Scheduling in P6

Parameter	80x86	microops
Max. instructions issued/clock	3	6
Max. instr. complete exec./clock		5
Max. instr. commited/clock		3
Window (Instrs in reorder buffer)		40
Number of reservations stations		20
Number of rename registers		40
No. integer functional units (FUs)		2
No. floating point FUs		1
No. SIMD Fl. Pt. FUs		1
No. memory Fus		1 load + 1 store

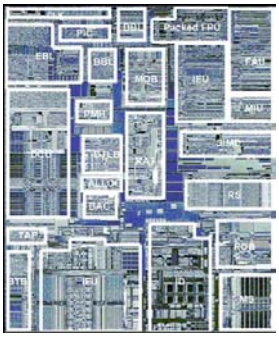
P6 Block Diagram



- IP = PC

From: http://www.digit-life.com/articles/pentium4/

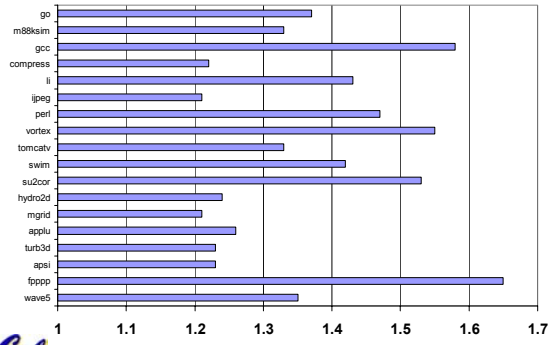
Pentium III Die Photo



- EBL/BBL - Bus logic, Front, Back
- MOB - Memory Order Buffer
- Packed FPU - MMX Fl. Pt. (SSE)
- IEU - Integer Execution Unit
- FAU - Fl. Pt. Arithmetic Unit
- MIU - Memory Interface Unit
- DCU - Data Cache Unit
- PMH - Page Miss Handler
- DTLB - Data TLB
- BAC - Branch Address Calculator
- RAT - Register Alias Table
- SIMD - Packed Fl. Pt.
- RS - Reservation Station
- BTB - Branch Target Buffer
- IFU - Instruction Fetch Unit (+I$)
- ID - Instruction Decode
- ROB - Reorder Buffer
- MS - Micro-instruction Sequencer

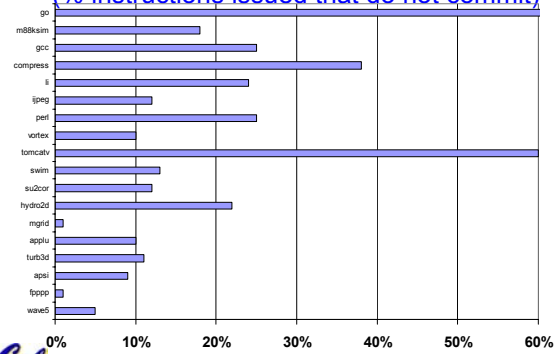
1st Pentium III, Katmai: 9.5 M transistors, 128 mm**2 in 0.25-micron

P6 Performance: uops/x86 instr
200 MHz, 8KI$/8KD$/256KL2$, 66 MHz bus



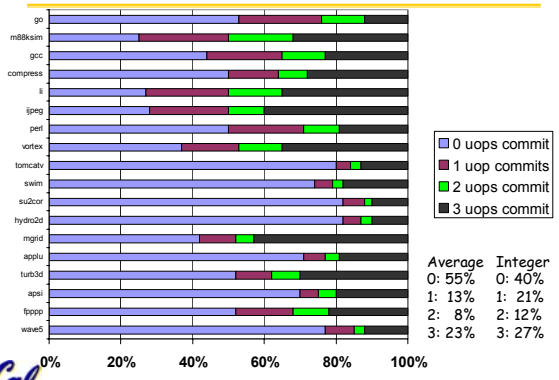
1.2 to 1.6 uops per IA-32 instruction: 1.36 avg. (1.37 integer)

P6 Performance: Speculation rate
(% instructions issued that do not commit)

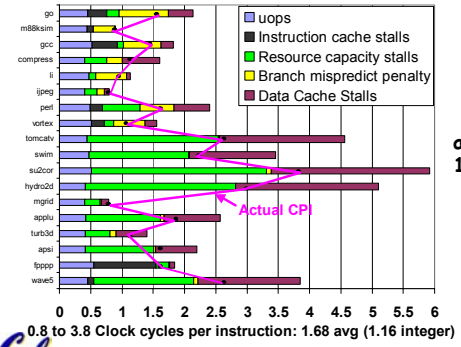


1% to 60% instructions do not commit: 20% avg (30% integer)

P6 Performance: µops commit/clock



P6 Dynamic Benefit?
Sum of parts CPI vs. Actual CPI



Ratio of sum of parts vs. actual CPI: 1.38X avg. (1.29X integer)

0.8 to 3.8 Clock cycles per instruction: 1.68 avg (1.16 integer)

Administrivia

- Full cache demo board Friday 10/31
  - 8 more PCs in 125 Cory this week; more boards?
- Thur 11/6: Design Doc for Final Project due
  - Deep pipeline? Superscalar? Out-of-order?
- Tue 11/11: Veteran's Day (no lecture)
- Fri 11/14: Demo Project modules
- Wed 11/19: 5:30 PM Midterm 2 in 1 LeConte
- Tues 11/22: Field trip to Xilinx
- CS 152 Project Week: 12/1 to 12/5
  - Mon: TA Project demo, Tue: 30 min Presentation, Wed: Processor racing, Fri: Written report

Pentium 4 Architecture Features

- Called "NetBurst" Microarchitecture (for Pentium 4, Pentium 5, …)
- Instruction Cache (Execution Trace Cache)
- Out-of-Order (OOO) execution engine
- Double-pumped Arithmetic Logic Unit
- Memory Subsystem (L1 access in 2 CP)
- Floating Point/Multi-Media performance

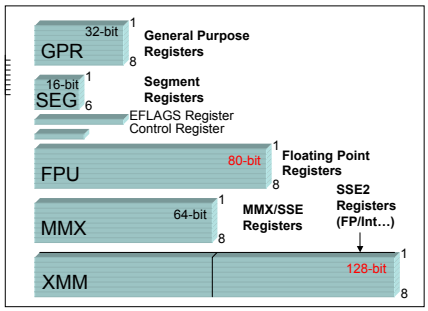
Pentium 4

- Still translate from 80x86 to micro-ops
- P4 has better branch predictor, more FUs
- Instruction Cache holds micro-operations vs. 80x86 instructions
  - no decode stages of 80x86 on cache hit
  - called "trace cache" (TC)
- Faster memory bus: initially 400 MHz v. 133 MHz
- Caches
  - Pentium III: L1I 16KB, L1D 16KB, L2 256 KB
  - Pentium 4: L1I 12K µops, L1D 8 KB, L2 256 KB
  - Block size: PIII 32B v. P4 128B; 128 v. 256 bits/clock
- Initial P4 Clock rates:
  - Pentium III 1 GHz v. Pentium IV 1.5 GHz
  - 14 stage pipeline vs. 24 stage pipeline

Pentium 4 features

- Multimedia instructions 128 bits wide vs. 64 bits wide => 144 new instructions
  - When used by programs??
  - Faster Floating Point: execute 2 64-bit Fl. Pt. Per clock
  - Memory FU: 1 128-bit load, 1 128-store /clock to MMX regs
- Using RAMBUS DRAM
  - Bandwidth faster, latency same as SDRAM
  - Later changed to support DDR SDRAM
- ALUs operate at 2X clock rate for many ops
- Pipeline doesn't stall at this clock rate: µops replay
- Rename registers: 40 vs. 128; Window: 40 v. 126
- BTB: 512 vs. 4096 entries (Intel: 1/3 misprediction improvement)

Registers



SIMD: Single Instruction Multiple Data

- Beginning with Pentium II, "SIMD" instructions added
- "Partitions" ALU to do multiple narrow data operations in 1 clock cycle by breaking carry chain:
  - 64 bits => 2 32-bit int ops OR 4 16-bit ops OR 8 8-bit ops
- SSE2 added in Pentium 4
  - 128 bits => 2 64-bit Fl. Pt. OR 4 32-bit Fl. Pt. OR ...

Instructions	Packed Data	Registers MXM 64-bit	Registers XMM 128bit	APPS
MMX (57) Pentium II	INT B,W,Q	Yes	---	Imaging, MM, comm.
SSE (70) Pentium III	SP Float	Yes	---	3-D geo/rendering video en/decode
SSE2 (144) Pentium 4	INT, SP/DP Float	Yes	Yes	4-D graphics Scientific Comp

Pentium 4 Cache

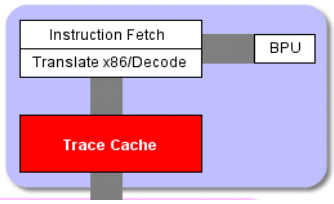
Level	Capacity	Assoc-iativity	Line Size (bytes)	Latency int/float (clocks)	Write Update Policy
First Data	8KB	4	64	2/9	write through
Trace Cache	12K µops	8	N/A	N/A	N/A
Second	256KB, 512KB	8	128 read 64 write	7/7	write back
Third	0, 512KB or 1MB or 2 MB	8	128 read 64 write	14/14	write back

Pentium 4 Trace Cache 1/4

•P4 places its L1 instruction cache after the Instruction Fetch.

• Arranges decoded instructions (µops) into some mini-programs that are ready to be used whenever there is a L1 Cache Hit.

•The trace cache can send up to 3 µops directly to execution engine.



Pentium 4 Trace Cache 2/4

• **What happens when there is a Trace Miss?**

- Trace Miss happens when L1 Cache misses, therefore, it needs to go to L2 cache, and fetch it from there. This results in 8 pipeline stages in order to translate and decode the instructions.

- Trace cache operates in two modes :

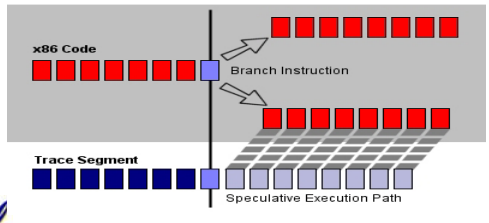
1) Execute mode : trace cache -> execution logic->executed. This is the mode Trace cache normally runs on when there is no Cache miss
2) Trace segment build mode: Happens when L1 cache miss. Fetch code from L2 cache, translate to µops, build trace segment, load segment to trace cache.

Pentium 4 Trace Cache 3/4

Trace cache applies Branch Prediction when building a trace.

It gets the code from the branch that it thinks the program will run on behind the code that it knows the program will take.

x86 code with branch: Trace cache build a trace from instructions up to including branch instruction, then pick a branch.



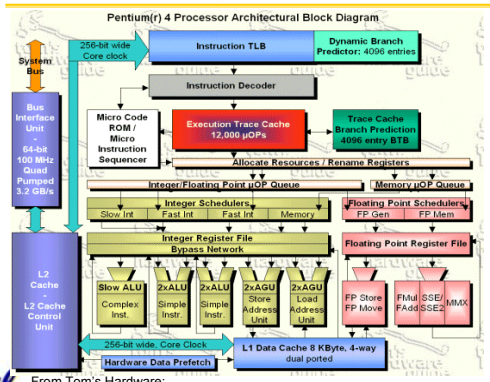
Pentium 4 Trace Cache 4/4

Conventional way:
Branch predictor figure outs branch to speculatively execute, then loada branch. takes up to 1 cycle of delay after every conditional branch instruction

With Trace cache:
the branch code is within the trace segment so there is no delay associated with bringing in the branch code.

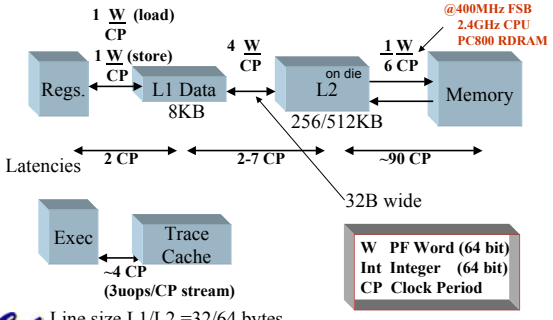
- Most x86 instructions decode into 2 or 3 µops
- Rare long instructions, which could decode into 100s of µops. PIII and P4 use microcode ROM which process these instructions so the regular decoder can do decoding on normal smaller instructions.
- Trace cache put a tag in trace segment when sees long instruction, Tag points to section of microcode ROM contains the µop sequence.
- When trace cache encounters the flag in execute mode, it lets microcode ROM stream proper sequence of µops into instruction stream for execution engine

Block Diagram



From Tom's Hardware:

Pentium 4 Speeds & Feeds



Line size L1/L2 =32/64 bytes

Out-of-Order Execution -- Pipeline

Pentium III processor misprediction pipeline

1	2	3	4	5	6	7	8	9	10
Fetch	Fetch	Decode	Decode	Decode	Rename	ROB Rd	Rdy/Sch	Dispatch	Exec

1 2	3 4	5 6	7 8	9 10	11 12	13 14	15 16	17 18	19 20
TC Fetch	TC Fetch	DriveAlloc	Rename	Que Sch	Sch Sch	Disp Disp	FR FR	Ex Flgs	BrCkDrive

Pentium 4 processor misprediction pipeline

Pentium 4 Basic pipeline stages:

Pentium 3 Basic Pipeline stages :

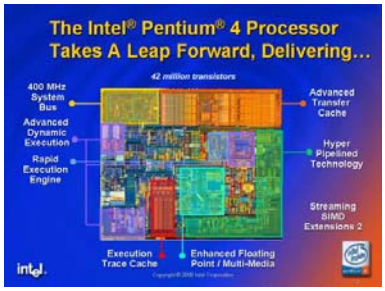
Stage	Work
1	Trace Cache next instruction pointer
2	Trace Cache next instruction pointer
3	Trace Cache fetch
4	Trace Cache fetch
5	Drive
6	Allocation
7	Rename
8	Rename
9	Queue
10	Schedule
11	Schedule
12	Schedule
13	Dispatch
14	Dispatch
15	Register Files
16	Register Files
17	Execute
18	Flags
19	Branch Check
20	Drive

1	Fetch
2	Fetch
3	Decode
4	Decode
5	Decode
6	Rename
7	ROB Rd
8	Rdy/Sch
9	Dispatch
10	Exec

Pentium 4 Basic Features

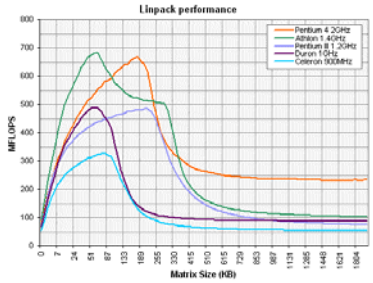
- 42 million transistors (256 KB L2 cache)
  55 watts @1.5GHz, 217 mm**2 (0.18u)
- 55 million transistors (512 KB L2 cache)
  82 watts @ 3.0 GHz, 131 mm**2 (0.13u)
- Xeon (server): 160 million transistors
  (512 KB L2 cache + 2048 KB L3)
  65 watts @ 2.0 GHz, 211 mm**2 (0.13u)
- 400/533/800 MHz Front Side Bus
  - Bus to Memory Hub, which connects to DRAM, AGP graphics bus, and I/O Hub

Pentium-4 die floor plan



L1 Dcache
L2 cache

Performance Comparison



100 x 100 x 8 = 80 KB

Scott Wasson "Intel's Pentium 4 Processor, Radical Chic"
www.tech-report.com/reviews/2001q3/pentium4-2ghz/

AMD Althon

- Similar to P6 microarchitecture (Pentium III), but more resources
- Transistors: PIII 24M v. Athlon 37M
- Die Size: 106 mm^2 v. 117 mm^2
- Power: 30W v. 76W
- Cache: 16K/16K/256K v. 64K/64K/256K
- Window size: 40 vs. 72 uops
- Rename registers: 40 v. 36 int +36 Fl. Pt.
- BTB: 512 x 2 v. 4096 x 2
- Pipeline: 10-12 stages v. 9-11 stages
- Clock rate: 1.0 GHz v. 1.2 GHz
- Memory bandwidth: 1.06 GB/s v. 2.12 GB/s

Benchmarks: Pentium 4 v. PIII v. Althon

- SPECbase2000
  - Int, P4@1.5 GHz: 524, PIII□@1GHz: 454, AMD Athlon@1.2Ghz:?
  - FP, P4@1.5 GHz: 549, PIII□@1GHz: 329, AMD Athlon@1.2Ghz:304
- WorldBench 2000 benchmark (business) PC World magazine, Nov. 20, 2000 (bigger is better)
  - P4 : 164, PIII : 167, AMD Athlon: 180
- Quake 3 Arena: P4 172, Athlon 151
- SYSmark 2000 composite: P4 209, Athlon 221
- Office productivity: P4 197, Athlon 209
- S.F. Chronicle 11/20/00: "… the challenge for AMD now will be to argue that frequency is not the most important thing--precisely the position Intel has argued while its Pentium III lagged behind the Athlon in clock speed."

Why Athlon, PIII were initially faster than P4?

Which explain performance advantage?

1) Athlon Instruction count less than P4
2) Athlon, PIII Average CPI better than P4
3) Athlon, PIII Clock rates better than P4

1. ABC: FFF
2. ABC: FFT
3. ABC: FTF
4. ABC: FTT
5. ABC: TFF
6. ABC: TFT
7. ABC: TTF
8. ABC: TTT

VLIW: Very Long Instruction Word

- Tradeoff instruction space for simple decoding
- The long instruction word has room for many operations
- By definition, all the operations the compiler puts in the long instruction word can execute in parallel
- E.g., 2 integer operations, 2 FP ops, 2 Memory refs, 1 branch
  - 16 to 24 bits per field => 7*16 or 112 bits to 7*24 or 168 bits wide
- Need compiling technique that schedules across several branches to have enough instructions

Loop Unrolling in VLIW

Memory reference 1	Memory reference 2	FP operation 1	FP op. 2	Int. op/ branch	Clock
LD F0,0(R1)	LD F6,-8(R1)				1
LD F10,-16(R1)	LD F14,-24(R1)				2
LD F18,-32(R1)	LD F22,-40(R1)	ADDD F4,F0,F2	ADDD F8,F6,F2		3
LD F26,-48(R1)		ADDD F12, F10, F2	ADDD F16,F14,F2		4
		ADDD F20, F18, F2	ADDD F24,F22,F2		5
SD 0(R1),F4	SD -8(R1),F8	ADDD F28, F26, F2			6
SD -16(R1),F12	SD -24(R1),F16				7
SD -32(R1),F20	SD -40(R1),F24			SUBI R1,R1,#48	8
SD -0(R1),F28				BNEZ R1,LOOP	9

Unrolled 7 times to avoid delays
7 results in 9 clocks, or 1.3 clocks per iteration
Need more registers in VLIW(EPIC => 128int + 128FP)

Superscalar v. VLIW

- Smaller code size
- Binary compatibility across generations of hardware
- Simplified Hardware for decoding, issuing instructions
- No Interlock Hardware (compiler checks?)
- More registers, but simplified Hardware for Register Ports (multiple independent register files?)

Problems with First Generation VLIW

- Increase in code size
  - generating enough operations in a straight-line code fragment requires ambitiously unrolling loops
  - whenever VLIW instructions are not full, unused functional units translate to wasted bits in instruction encoding
- Operated in lock-step; no hazard detection HW
  - a stall in any functional unit pipeline caused entire processor to stall, since all functional units must be kept synchronized
  - Compiler might predict function units, but caches hard to predict
- Binary code compatibility
  - Pure VLIW => different numbers of functional units and unit latencies require different versions of the code

Intel/HP IA-64 "Explicitly Parallel Instruction Computer (EPIC)"

- IA-64: instruction set architecture; EPIC is type
  - EPIC = 2nd generation VLIW
- Itanium™ is name of first implementation (2001)
  - Highly parallel and deeply pipelined hardware at 800Mhz
  - 6-wide, 10-stage pipeline at 800Mhz on 0.18 µ process
- 128 64-bit integer registers + 128 82-bit floating point registers
  - Not separate register files per functional unit as in old VLIW
- Hardware checks dependencies (interlocks => binary compatibility over time)
- Predicated execution (select 1 out of 64 1-bit flags) => 40% fewer mispredictions?

Intel/HP IA-64 "Explicitly Parallel Instruction Computer (EPIC)"

- Instruction group: a sequence of consecutive instructions with no register data dependences
  - All the instructions in a group could be executed in parallel, if sufficient hardware resources existed and if any dependences through memory were preserved
  - An instruction group can be arbitrarily long, but the compiler must explicitly indicate the boundary between one instruction group and another by placing a stop between 2 instructions that belong to different groups
- IA-64 instructions are encoded in bundles, which are 128 bits wide.
  - Each bundle consists of a 5-bit template field and 3 instructions, each 41 bits in length
- 3 Instructions in 128 bit "groups"; field determines if instructions dependent or independent
  - Smaller code size than old VLIW, larger than x86/RISC
  - Groups can be linked to show independence > 3 instr

5 Types of Execution in Bundle

Execution Unit Slot	Instruction type	Instruction Description	Example Instructions
I-unit	A	Integer ALU	add, subtract, and, or, cmp
	I	Non-ALU Int	shifts, bit tests, moves
M-unit	A	Integer ALU	add, subtract, and, or, cmp
	M	Memory access	Loads, stores for int/FP regs
F-unit	F	Floating point	Floating point instructions
B-unit	B	Branches	Conditional branches, calls
L+X	L+X	Extended	Extended immediates, stops

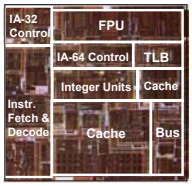
- 5-bit template field within each bundle describes both the presence of any stops associated with the bundle *and* the execution unit type required by each instruction within the bundle

IA-64 Registers

- The integer registers are configured to help accelerate procedure calls using a register stack
  - mechanism similar to that developed in the Berkeley RISC-I processor and used in the SPARC architecture.
  - Registers 0-31 are always accessible and addressed as 0-31
  - Registers 32-128 are used as a register stack and each procedure is allocated a set of registers (from 0 to 96)
  - The new register stack frame is created for a called procedure by renaming the registers in hardware;
  - a special register called the current frame pointer (CFM) points to the set of registers to be used by a given procedure
- 8 64-bit Branch registers used to hold branch destination addresses for indirect branches
- 64 1-bit predict registers

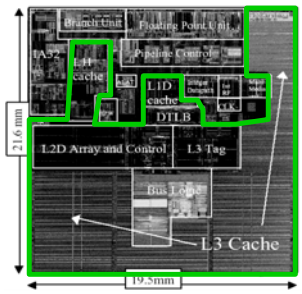
Itanium™ Processor Silicon

(Copyright: Intel at Hotchips '00)



Core Processor Die

Itanium II CPU/cache area comparison



Caches

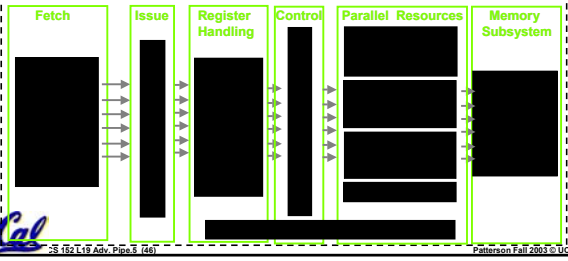
Itanium™ EPIC Design Maximizes SW-HW Synergy

(Copyright: Intel at Hotchips '00)

Architecture Features programmed by compiler:

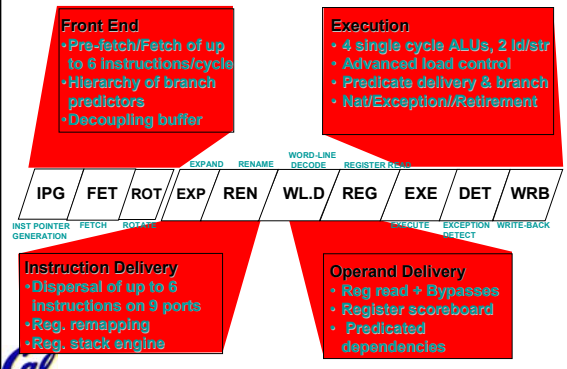
Branch Hints | ***Explicit Parallelism*** | ***Register Stack & Rotation*** | ***Predication*** | ***Data & Control Speculation*** | ***Memory Hints***

Micro-architecture Features in hardware:



10 Stage In-Order Core Pipeline

(Copyright: Intel at Hotchips '00)



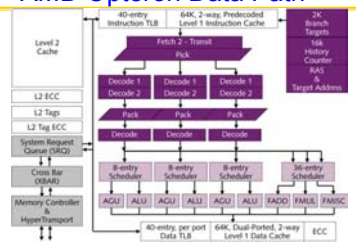
Comments on Itanium

- Remarkably, the Itanium has many of the features more commonly associated with the dynamically-scheduled pipelines
  - strong emphasis on branch prediction, register renaming, scoreboarding, a deep pipeline with many stages before execution (to handle instruction alignment, renaming, etc.), and several stages following execution to handle exception detection
- Surprising that an approach whose goal is to rely on compiler technology and simpler HW seems to be at least as complex as dynamically scheduled processors!

AMD Opteron

- 9 execution units, 3 integer units (ALUs), 3 address-generation units(AGUs), 3 floating point units.
- Opteron can decode up to 3 x86 instructions and dispatch up to 9 μops per cycle --- assume each of them is mapped to one of the nine execution units.
- 12 pipeline stages

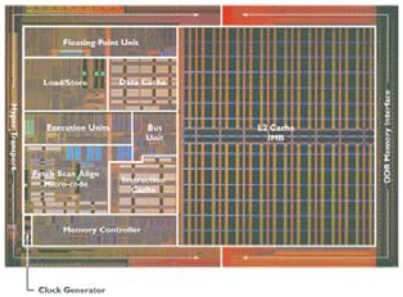
AMD Opteron Data Path



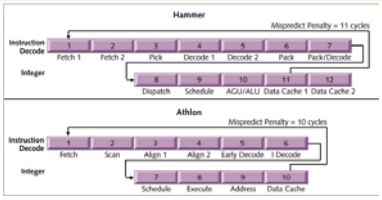
From Microprocessor Report November 26, 2001 "AMD Takes Hammer to Itanium"

- Basically an enhanced Athlon
- Predecode bits in L1 instruction cache include branch prediction.
- L1 data cache now dual ported and can support two 64-bit stores in one cycle.

AMD Opteron Die Photo



Opteron ("Hammer") Pipeline vs. Athlon

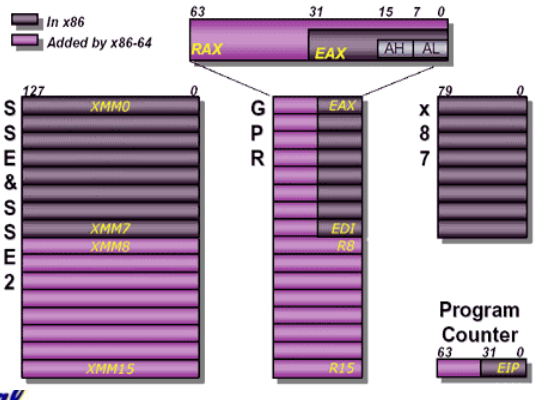


From Microprocessor Report November 26, 2001 "AMD Takes Hammer to Itanium"

- 2 stages for Instr Fetch, Data cache
  - fetch 2 ~ P 4 drive stage, spent moving data across die
- Pick stage = scan deciding whether the instruction is for the MROM or hardware decoders
- Decode 1 & 2 stages ~ align 1 & 2 stages
- Commit stage (unseen) updates architectural regs

AMD64: 64 bit

- Opteron : 64 bit register file.
- Old x86 registers are extended to 64 bits with new registers added.
- The existing x86 binaries wont see the upper half of the eight new registers, its only visible to new 64 bit code.
- AMD Opteron with 16 64-bit registers. (Itanium has 128 general purpose + 128 FP)
- In 64-bit mode, AMD has 1/16 the quantity of registers that Itanium has.



AMD Opteron vs. Itanium

- Opteron and Itanium both have 9 execution units.
- Opteron can dispatch 9 μops to Itanium's six.
- Opteron has 3 execution units for FP, 6 for Interger code(3 AGU and 3 ALU)
- Itanium has 2 units for FP, 7 for integer units (2 integer units, two combo integer and load/store units, two floating point units, 3 branch units).

AMD Opteron vs. Itanium

- 2 double data-rate (DDR) controller that directly manage external SDRAM memory.
- 3 HyperTransport links. More advantage in multiprocessing. Up to 8 Opteron processor can communicate amongst themselves using buildt-in hypertransport links.
- First Opteron has no L3 cache (Itanium II has L3 Cache)
- Single instruction multiple data (SIMD)
- Opteron includes SSE2, for compatibility for both families.
- Opteron : SSE, SSE2, 3D now

Opteron v. Itanium Registers

	AMD Opteron	INTEL ITANIUM
General Purpose Register	16	128
Floating Point Registers	8	128
SIMD Registers	16	128

Workstation Microprocessors 3/2001

Processor	Alpha 21264B	AMD Athlon	HP PA-8600	IBM Power3-II	Intel Pentium III	Intel Pentium 4	MIPS R12000	Sun Ultra-II	Sun Ultra-III
Clock Rate	833MHz	1.2GHz	552MHz	450MHz	1.0GHz	1.8GHz	400MHz	480MHz	900MHz
Cache (I/D/L2)	64K/64K	64K/64K/256K	512K/1M	32K/64K	16K/16K/256K	12K/8K/256K	32K/32K	16K/16K	32K/64K
Issue Rate	4 issue	3 x86 instr	4 issue	4 issue	3 x86 instr	3 x ROPs	4 issue	4 issue	4 issue
Pipeline Stages	7/9 stages	9/11 stages	7/9 stages	7/8 stages	12/14 stages	22/24 stages	6 stages	6/9 stages	14/15 stages
Out of Order	80 instr	72ROPs	56 instr	32 instr	40 ROPs	126 ROPs	48 instr	None	None
Rename regs	48/41	36/36	56 total	16 int/24 fp	40 total	128 total	32/32	None	None
BHT Entries	4K x 9-bit	4K x 2-bit	2K x 2-bit	2K x 2-bit	2K x 2-bit	4K x 2-bit	2K x 2-bit	512 x 2-bit	16K x 2-bit
TLB Entries	128/128	280/288	120 unified	128/128	32I / 64D	128I/65D	64 unified	64I/64D	128I/512D
Memory B/W	2.66GB/s	2.1GB/s	1.54GB/s	1.6GB/s	1.06GB/s	3.2GB/s	539 MB/s	1.9GB/s	4.8GB/s
Package	CPGA-588	PGA-462	LGA-544	SCC-1088	PGA-370	PGA-423	CPGA-527	CLGA-787	1368 FC-LGA4
IC Process	0.18μ 6M	0.18μ 6M	0.25μ 2M	0.22μ 6m	0.18μ 6M	0.18μ 6M	0.25μ 4M	0.29μ 6M	0.18μ 7M
Die Size	115mm²	117mm²	477mm²	163mm²	106mm²	217mm²	204mm²	126 mm²	210mm²
Transistors	15.4 million	37 million	130 million	23 million	24 million	42 million	7.2 million	3.8 million	29 million
Est mfg cost*	$160	$62	$330	$110	$39	$110	$125	$70	$145
Power(Max)	75W*	76W	60W*	36W*	30W	55W(1.5G)	25W*	20W*	65W
Availability	1Q01	4Q00	3Q00	4Q00	2Q00	4Q00	2Q00	3Q00	4Q00

- Max issue: 4 instructions (many CPUs)
  Max rename registers: 128 (Pentium 4)
  Max Window Size (OOO): 126 instructions (Pentium 4)
  Max Pipeline: 22/24 stages (Pentium 4)

Source: Microprocessor Report, www.MPRonline.com

Cost (Microprocessor Report, 8/25/03)

Processor	Alpha 21364	AMD Athlon XP	HP PA-8700	IBM Power4+	Intel Itanium 2	Intel XeonMP	Intel Xeon	MIPS R14000	Sun UltraSPARC-III
Clock Rate	1.15GHz	2.17GHz	870MHz	1.45GHz	1 GHz	2.0GHz	3.06GHz	600MHz	1.05GHz
Cache (I/D/L2/L3)	64K/64K/ 1.75M	64K/64K/ 512K	750K/ 1.5M	64K/32K/ 8.5MB	16K/16K/ 256K/3M	12K/8K/ 512K/2M	12K/8K/ 512K	32K/32K	32K/64K
Issue Rate	4 issue	3 x86 instr	4 issue	8 issue	6 issue	3 ROPs	3 ROPs	4 issue	4 issue
Pipeline Stages	7/9 stages	10/11 stages	7/9 stages	12/13 stages	8 stages	22/24 stages	22/24 stages	6 stages	14/15 stages
Out of Order	80 instr	72ROPs	56 instr	200 instr	None	126 ROPs	126 ROPs	48 instr	None
Rename Regs	48/41	36/36	56 total	48/40	328 total	128 total	128 total	32/32	None
BHT Entries	4K x 9-bit	4K x 2-bit	2K x 2-bit	3 x 16K x 1-bit	512 x 2-bit	4K x 2-bit	4K x 2-bit	2K x 2-bit	16K x 2-bit
TLB Entries	128/128	280/288	240 unified	1,024 unified	32L1I/32L1D/ 256L2D	128I/64D	128I/64D	64 unified	128I/512D
Memory B/W	12GB/s	2.7GB/s	1.54GB/s	12.8GB/s	6.4GB/s	3.2GB/s	4.3GB/s	1.6GB/s	4.8GB/s
Package	FC-LGA-1443	PGA-462	LGA-544	MCM	mPGA-700	mPGA-603	PGA-423	FCBGA-1153	FC-LGA 1368
IC Process	0.18-nm 7M	0.13-nm 6M	0.18-nm 7M	0.13-nm 7m	0.18-nm 6M	0.13-nm 6M	0.13-nm 6M	0.15-nm 7M	0.15-nm 7M
Die Size	397mm²	101mm²	304mm²	267mm²*	418mm²*	211mm²	131mm²	142mm²	210mm²
Transistors	1.35 million	54.3 million	130 million	184 million**	221 million	160 million*	55 million	7.7 million	29 million
Est Die Cost	$180*	$46*	$96*	$144**	$166*	$64*	$55*	$68*	$72*
Power (Max)	110W*	76W(MTP)	75W*	85W**	130W	65W(Max)	82W(TDP)	16W*	75W*
Availability	1Q03	1Q03	3Q02	4Q02	3Q02	1Q03	4Q02	2Q02	1Q02

- 3X die size Pentium 4, 1/3 clock rate Pentium 4
- Cache size (KB): 16+16+256+3076 v. 12+8+512

SPEC 2000 Performance 3/2001 *Source: Microprocessor Report, www.MPRonline.com*

essor	Alpha 21264B	AMD Athlon	HP PA-8600	IBM Power 3-II	Dell PIII	Intel P4	MIPS R12000	Sun Ultra-II	Sun Ultra-III
em or herboard	Alpha es40 Model 6	AMD GA-7ZM	HP9000 j6000	RS/6000 44P-170	Dell Prec. 420	Intel D850GB	SGI 2200	Sun Enterpa 450	Sun Blade 10
k Rate	833MHz	1.2GHz	552MHz	450MHz	1GHz	1.5GHz	400MHz	480MHz	900MH
rnal Cache	8MB	None	None	8MB	None	None	8MB	8MB	8MB
gzip	392	n/a	376	230	545	553	226	165	349
vpr	452	n/a	421	285	354	208	384	212	383
gcc	617	n/a	577	350	401	588	318	232	500
mcf	441	n/a	384	498	276	473	563	356	474
crafty	694	n/a	472	304	523	497	334	175	439
parser	360	n/a	361	171	362	472	283	211	412
eon	648	n/a	395	280	615	650	360	209	465
perlbmk	526	n/a	406	215	614	703	246	247	457
gap	366	n/a	229	256	443	708	204	171	300
vortex	673	n/a	764	312	717	735	294	304	581
bzip2	560	n/a	349	258	396	420	334	237	500
twolf	658	n/a	479	414	394	403	451	243	473
Cint_base2000	518	n/a	417	288	454	524	320	225	438
wupside	529	360	340	360	416	759	280	284	497
swim	1,156	506	761	279	493	1,244	300	285	752
mgrid	580	272	462	319	274	558	231	226	377
applu	424	298	563	327	280	641	237	150	221
mesa	713	302	300	330	541	553	289	273	469
galgel	558	468	569	429	335	537	989	735	1,266
art	1,540	213	419	969	410	514	995	920	990
equake	231	236	347	500	249	739	222	149	211
facerec	822	411	258	257	307	451	411	459	718
ammp	488	221	376	326	294	366	373	313	421
lucas	731	237	370	284	349	764	259	205	204
fma3d	528	365	302	340	297	427	192	207	302
sixtrack	340	256	286	234	170	257	199	159	273
aspi	593	278	523	349	371	427	257	189	340
Cfp_base2000	590	304	400	356	329	549	319	274	427

Performance (Microprocessor Report, 8/25/03)

Processor	Alpha 21364	AMD Athlon XP	HP PA-8700	IBM Power4+	Intel Itanium 2	Intel XeonMP	Intel Xeon	MIPS R14000	Sun UltraSPARC III
System or Motherboard	Alpha GS1280/7	ASUS A7N8X	HP9000 C3750	IBM eServer p690 6M2	HP RX2600	Dell PowerEdge 6650	Dell Prec. 350	SGI 3200	Sun Blade 2050
Clock Rate	1.15GHz	2.17GHz	875MHz	1.9GHz	1.0GHz	2.0GHz	3.06GHz	600MHz	1.05GHz
External Cache	None	None	None	16MB	None	None	None	8MB	8MB
164.gzip	583	1,026	588	673	583	758	1,138	322	433
175.vpr	822	693	688	902	704	625	606	572	460
176.gcc	859	755	906	914	1,014	1,100	1,236	445	577
181.mcf	712	420	494	1,391	834	599	773	783	659
186.crafty	987	1,297	758	884	881	717	1,179	507	598
197.parser	514	906	495	881	600	778	1,075	409	488
252.eon	958	1,483	592	1,150	1,004	920	1,387	507	527
253.perlbmk	768	1,306	619	912	815	952	1,381	467	540
254.gap	636	1,059	339	936	880	722	1,417	308	372
255.vortex	1,094	1,608	1,106	1,428	1,193	1,118	1,658	679	738
256.bzip2	824	840	534	965	750	745	856	493	629
300.twolf	1,018	887	911	1,198	880	1,009	900	615	570
SPECint_base2000	795	960	642	900	810	816	1,085	483	537
168.wupwise	883	1,131	446	1,532	1,003	816	1,406	434	659
171.swim	2,590	1,006	991	1,417	3,205	848	1,837	529	980
172.mgrid	708	799	621	850	1,720	449	1,047	379	487
173.applu	1,518	654	702	979	2,033	496	1,168	381	310
177.mesa	928	1,103	694	937	642	814	1,165	425	548
178.galgel	2,109	738	1,615	4,196	2,569	1,200	1,596	1,398	1,713
179.art	2,014	495	670	1,864	4,226	1,147	716	1,436	9,389
183.equake	519	730	413	2,098	1,871	469	1,291	347	646
187.facerec	1,105	1,008	430	1,595	1,152	762	1,215	647	968
188.ammp	735	587	553	923	788	729	644	573	509
189.lucas	1,522	853	448	1,306	1,206	682	1,522	442	371
191.fma3d	1,019	850	494	898	797	551	1,089	306	490
200.sixtrack	469	538	471	621	891	1,378	564	298	366
301.apsi	1,242	705	696	966	678	695	833	406	471
SPECfp_base2000	1,124	776	600	1,128	1,356	872	1,092	499	701

1.2X 1.5X 1.7X 1.5X

Performance of IA-64 Itanium?

- Whether this approach will result in significantly higher performance than other recent processors is unclear
- The clock rate of Itanium (733 MHz) and Itanium II (1.0 GHz) is slower than the clock rates of several dynamically-scheduled machines, including the Intel Pentium 4 and AMD Opteron

Summary

- OOO processors
  - HW translation to RISC operations
  - Superpipelined P4 with 22-24 stages vs. 12 stage Opteron
  - Trace cache in P4
  - SSE2 increasing floating point performance
- Very Long Instruction Word machines (VLIW)
  ⇒ Multiple operations coded in single, long instruction
  - EPIC as a hybrid between VLIW and traditional pipelined computers
  - Uses more registers
- 64-bit: New ISA (IA-64) or Evolution (AMD64)?
  - 64-bit Address space needed larger DRAM memory