The Intel P6 and NetBurst microarchitectures

M. Sonza Reorda

Politecnico di Torino
Dipartimento di Automatica e Informatica

P6

The Intel P6 microarchitecture is the basis for

- **PentiumPro**
- **Pentium II**
- **Pentium III.**

The 3 processors differ because of

- **Some instruction extensions (MMX, SSE)**
- **Clock rate**
- **Cache architecture**
- **Memory interface.**

Evolution in P6 processors

Processor	First ship date	Clock rate range	L1 cache	L2 cache
Pentium Pro	1995	100–200 MHz	8 KB instr. + 8 KB data	256 KB–1024 KB
Pentium II	1998	233–450 MHz	16 KB instr. + 16 KB data	256 KB–512 KB
Pentium II Xeon	1999	400–450 MHz	16 KB instr. + 16 KB data	512 KB–2 MB
Celeron	1999	500–900 MHz	16 KB instr. + 16 KB data	128 KB
Pentium III	1999	450–1100 MHz	16 KB instr. + 16 KB data	256 KB–512 KB
Pentium III Xeon	2000	700–900 MHz	16 KB instr. + 16 KB data	1 MB–2 MB

Ev

The CPU (5.5. million transistors) and the L2 cache (2 million transistors) are integrated into a single package (*cartridge*) composed of two dies.

The two parts are connected by a dedicated bus running at the full CPU speed (200 MHz).

Processor	First ship date	Clock rate range	L1 cache	L2 cache
Pentium Pro	1995	100–200 MHz	8 KB instr. + 8 KB data	256 KB–1024 KB
Pentium II	1998	233–450 MHz	16 KB instr. + 16 KB data	256 KB–512 KB
Pentium II Xeon	1999	400–450 MHz	16 KB instr. + 16 KB data	512 KB–2 MB
Celeron	1999	500–900 MHz	16 KB instr. + 16 KB data	128 KB
Pentium III	1999	450–1100 MHz	16 KB instr. + 16 KB data	256 KB–512 KB
Pentium III Xeon	2000	700–900 MHz	16 KB instr. + 16 KB data	1 MB–2 MB

Ev 6 processors

MMX extension

Processor	st ship date	Clock rate range	L1 cache	L2 cache
Pentium Pro	1995	100–200 MHz	8 KB instr. + 8 KB data	256 KB–1024 KB
Pentium II	1998	233–450 MHz	16 KB instr. + 16 KB data	256 KB–512 KB
Pentium II Xeon	1999	400–450 MHz	16 KB instr. + 16 KB data	512 KB–2 MB
Celeron	1999	500–900 MHz	16 KB instr. + 16 KB data	128 KB
Pentium III	1999	450–1100 MHz	16 KB instr. + 16 KB data	256 KB–512 KB
Pentium III Xeon	2000	700–900 MHz	16 KB instr. + 16 KB data	1 MB–2 MB

5

Ev 6 processors

SSE extension

Processor	p date	Clock rate range	L1 cache	L2 cache
Pentium Pro	1995	100–200 MHz	8 KB instr. + 8 KB data	256 KB–1024 KB
Pentium II	1998	233–450 MHz	16 KB instr. + 16 KB data	256 KB–512 KB
Pentium II Xeon	1999	400–450 MHz	16 KB instr. + 16 KB data	512 KB–2 MB
Celeron	1999	500–900 MHz	16 KB instr. + 16 KB data	128 KB
Pentium III	1999	450–1100 MHz	16 KB instr. + 16 KB data	256 KB–512 KB
Pentium III Xeon	2000	700–900 MHz	16 KB instr. + 16 KB data	1 MB–2 MB

6

mops

The P6 translates each IA-32 instruction into a series of micro-operations (mops), which are similar to typical RISC instructions.

Up to 3 IA-32 instructions are fetched, decoded, and translated into mops at each clock cycle.

mops execution

mops are executed by an out-of-order speculative pipeline using register renaming and ROB.

Up to 3 mops per clock can be renamed and dispatched to the reservation stations.

Up to 3 mops per clock can be committed.

Pipeline

It is composed of 14 stages:

- **8 stages are used for in-order instruction fetch, decode, and dispatch**
- **3 stages are used for out-of-order execution in one out of 5 functional units**
- **3 stages are used for instruction commit.**

A stage stalls if the input buffer does not provide it with new data, or if the output buffer is full.

Fetch unit

It fetches one cache line (32 bytes) per clock cycle from the L1 code cache.

It identifies instruction boundaries, and performs branch prediction.

Decoding unit

It is composed of 3 units working in parallel: two for simple instructions and one for complex instructions.

Simple instructions are mapped to single m-ops. Complex instructions are mapped to several m-ops (up to 4): this procedure requires more than one clock cycle.

The decoding stage can thus produce up to 6 m-ops per clock cycle, which are then written to the *instruction pool*.

Instruction pool

It stores up to 20/30 m-ops.

It continuously checks which m-ops can be sent to the functional units.

m-ops can be sent to functional units in any order.

Branch prediction

It is based on a Branch Target Buffer (BTB) with 512 entries.

If the BTB fails, a static prediction is exploited, i.e.,

- **Backward branches are predicted taken**
- **Forward branches are predicted untaken.**

The penalty for mispredicted branches is between 10 and 15 cycles, plus the overhead for incorrectly speculated instructions.

Register alias table

It converts references to x86 registers to references to the 40 internal registers, thus performing *register renaming*.

Dispatch unit

It assigns instructions

- **to one of the 20 reservation stations *and***
- **to one of 40 entries in the ROB.**

Execution units

They are

- **Two units for FP instructions (FPUI, FPUII)**
- **Two units for integer instructions (INTI, INTII)**
- **One memory interface unit (MIU).**

The 5 units can operate in parallel and independently.

Latency and repeat rate

Instruction name	Pipeline stages	Repeat rate
Integer ALU	1	1
Integer load	3	1
Integer multiply	4	1
FP add	3	1
FP multiply	5	2
FP divide (64-bit)	32	32

Retire unit

It continuously looks in the instruction pool for completed instructions.

Retirement happens according to a strict in-order strategy.

As soon as it becomes possible, it performs

- **Data dependencies resolution**
- **Branch prediction verification**
- **Real register updating.**

Performance measurements

The following data come from a paper by Bhandarkar and Ding (1997).

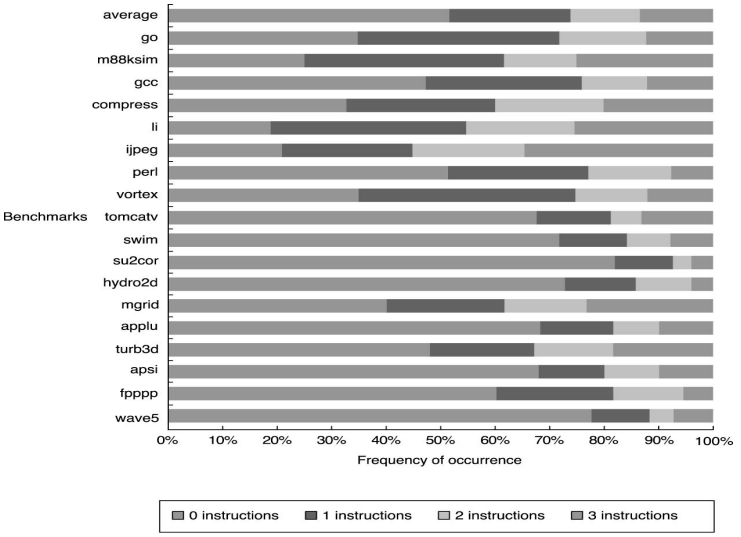
In the case of P6, the ideal CPI is 0.33.

The P6 can fall behind because at some clock cycles it failed in completing 3 instructions per clock cycle. This can be due to a number of problems occurred in any of the previous stages.

Possible causes for stalls

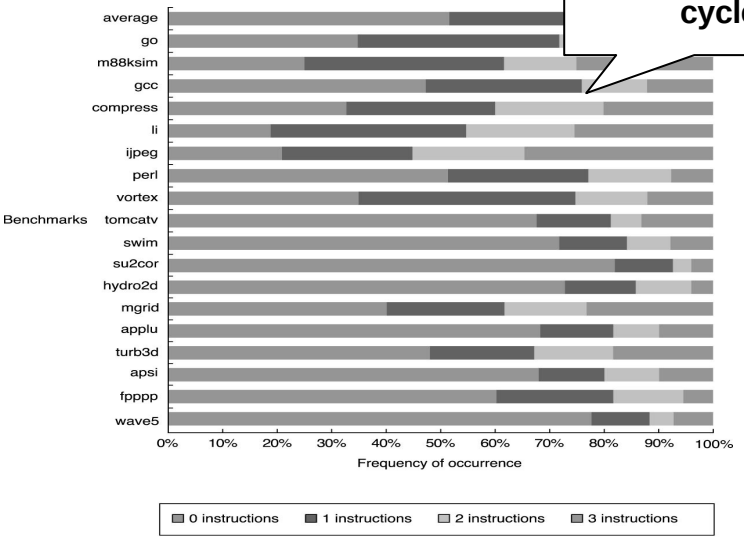
- **Instruction cache miss**
- **The 3 decoded IA-32 instructions generated more than the maximum number of mops**
- **Lack of reservation stations or ROB entries**
- **Data dependency**
- **Data cache miss**
- **Branch mispredictions.**

Stalls in the decode cycle

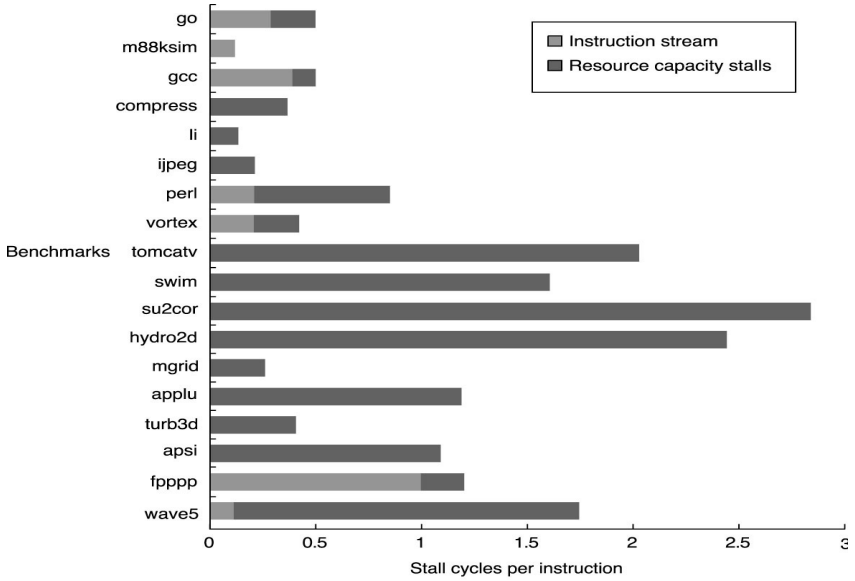


Stalls in the decode

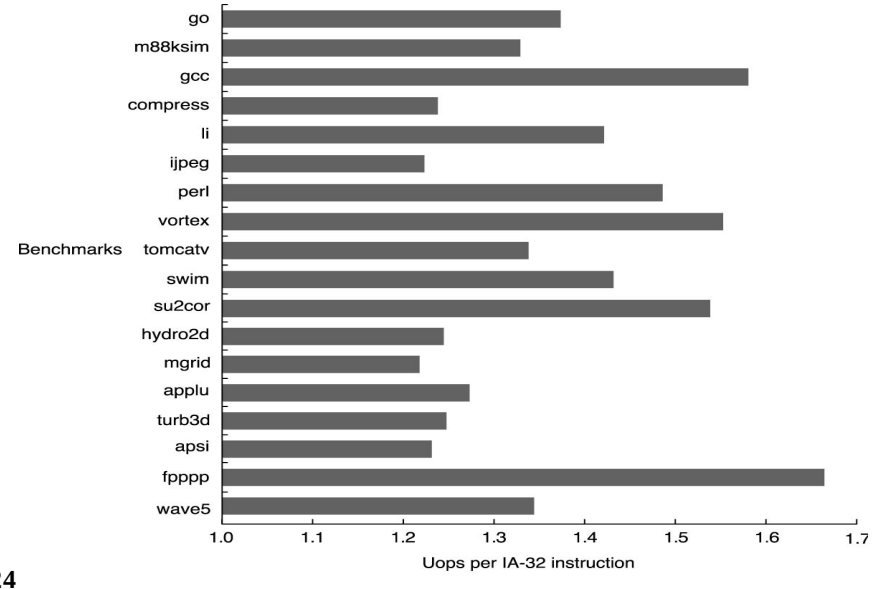
In the average, 0.87 instructions are decoded per cycle



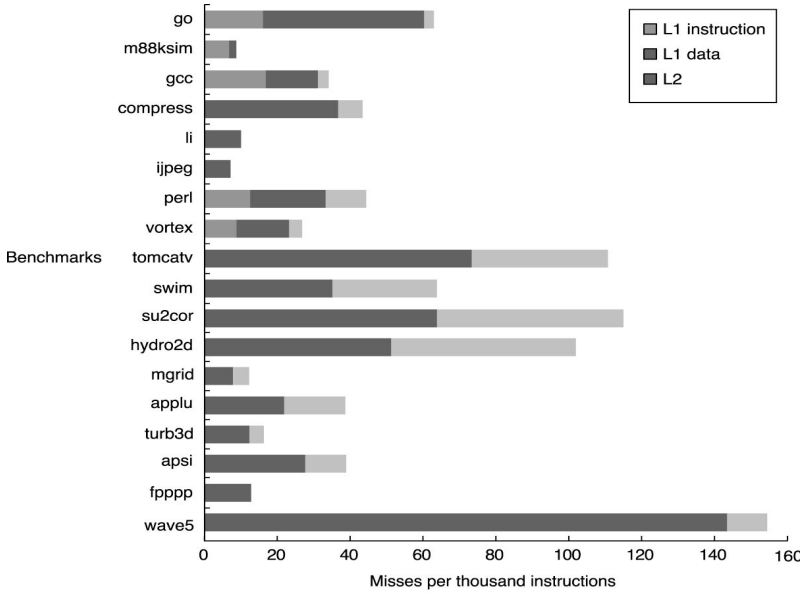
Causes of stalls in the decode cycle



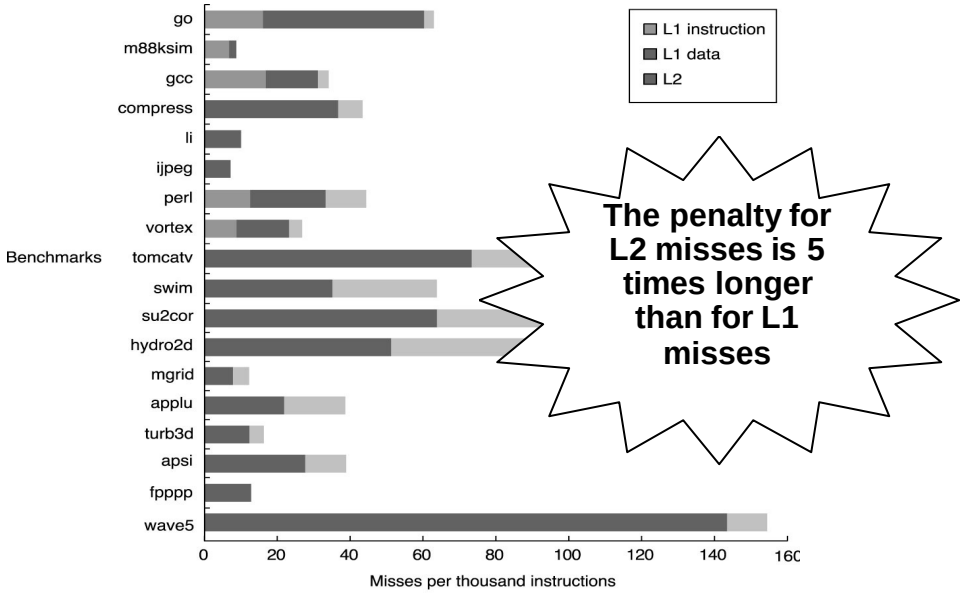
Number of mops per instruction



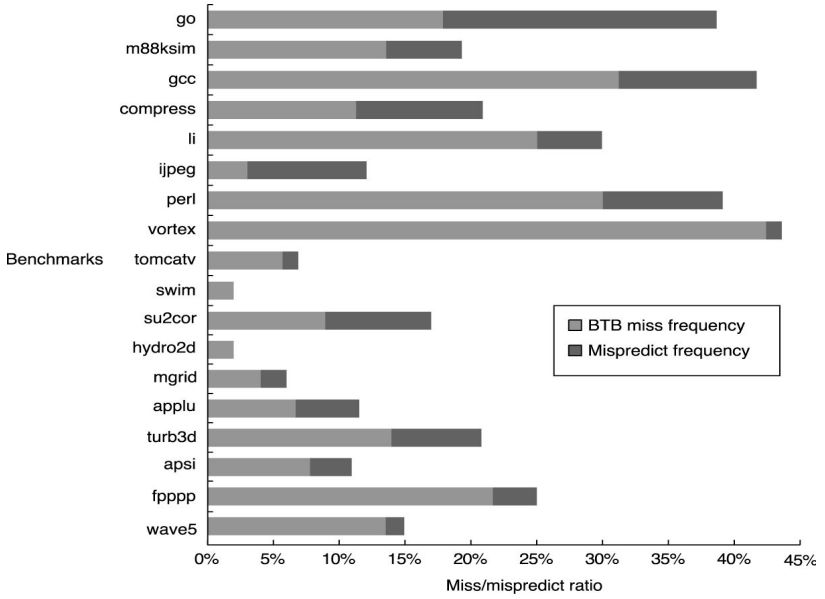
Cache miss ratio



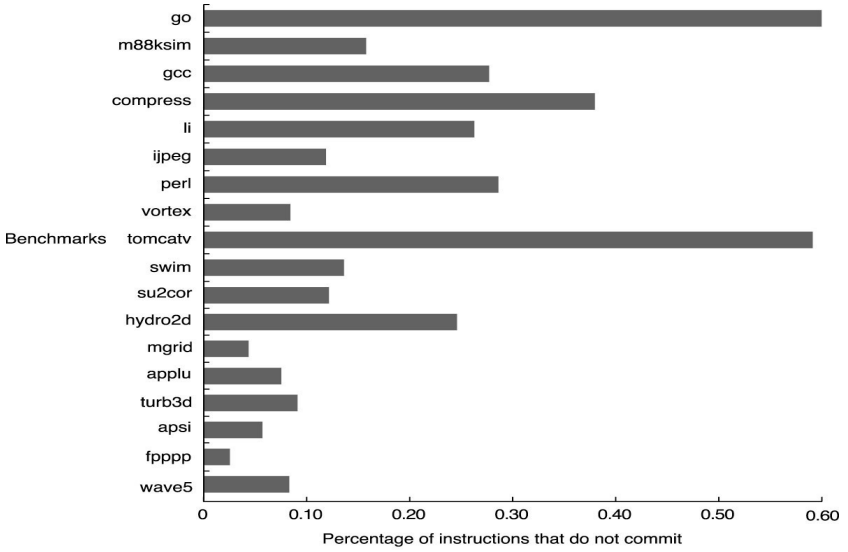
Cache miss ratio



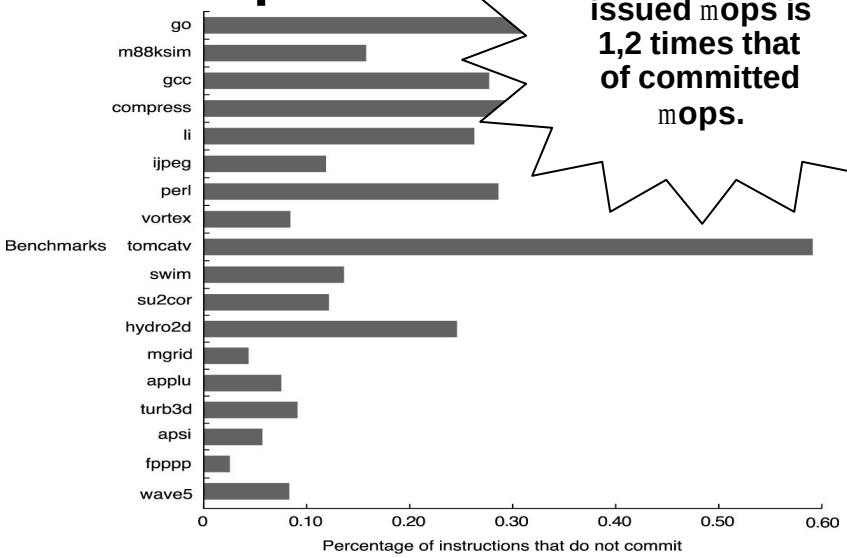
Prediction accuracy



Speculation factor

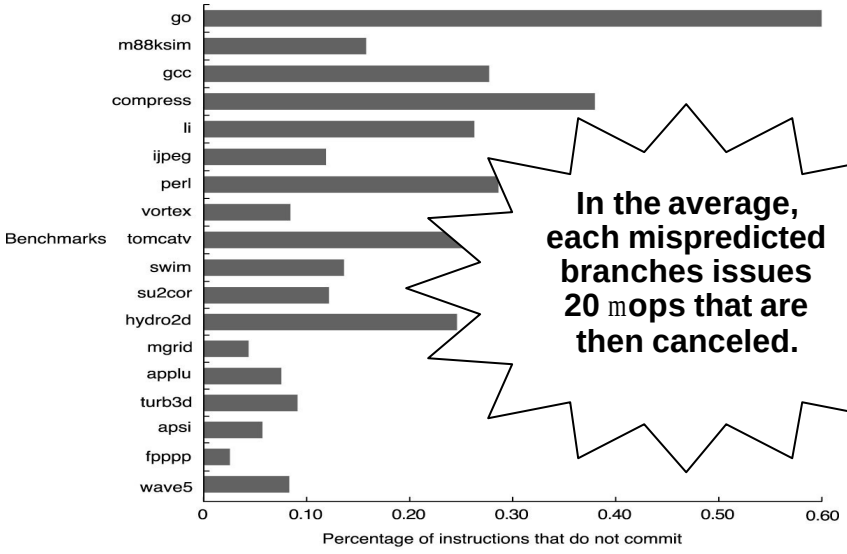


Speculation



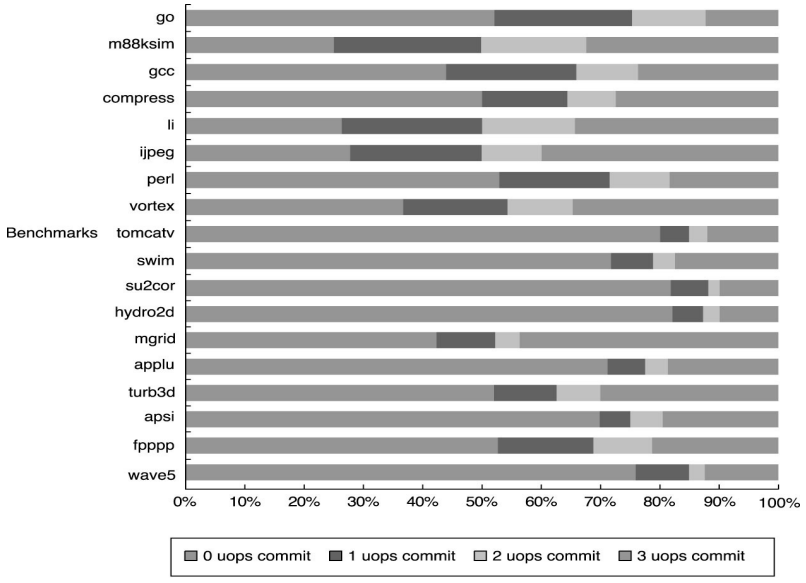
The number of issued mops is 1,2 times that of committed mops.

Speculation factor

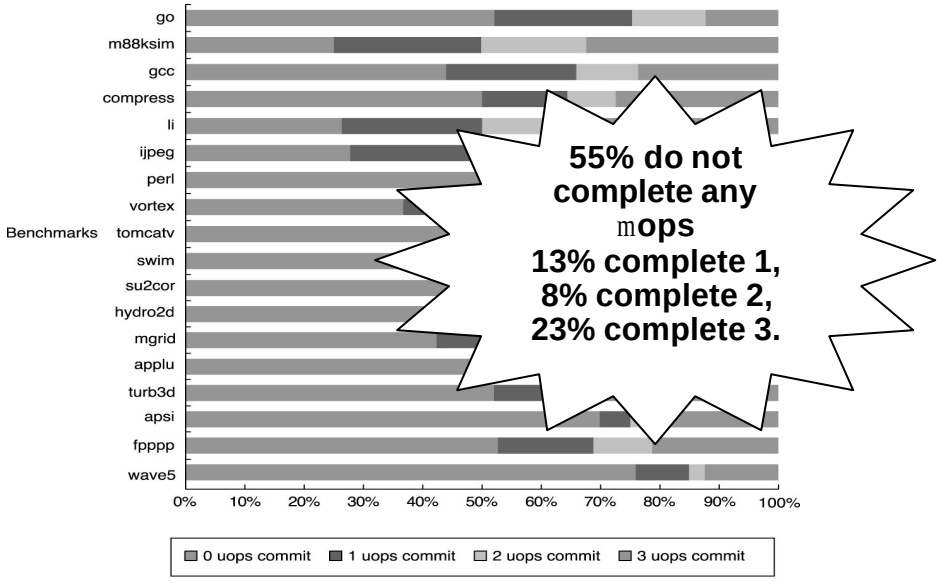


In the average, each mispredicted branches issues 20 mops that are then canceled.

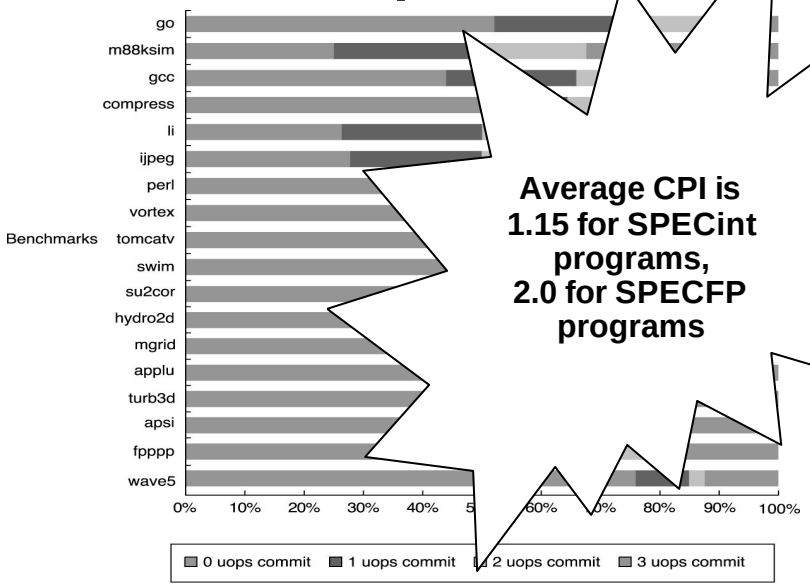
Overall performance



Overall performance



Overall performance



Average CPI is 1.15 for SPECint programs, 2.0 for SPECFP programs

The NetBurst architecture

NetBurst is the name of the Pentium 4 microarchitecture.

The two main goals of this architecture are

- **To support a higher clock frequency**
- **To maintain the sustained execution throughput as close as possible to the ideal maximum.**

To achieve these goals, several differences were introduced in the NetBurst architecture with respect to the P6 one.

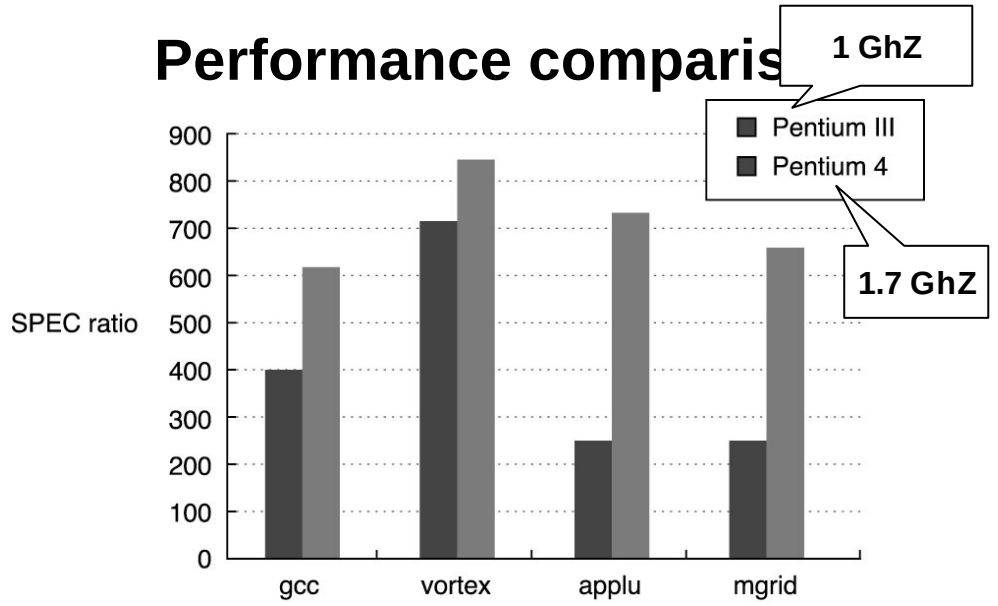
P6/NetBurst differences (I)

- **Deeper pipeline: NetBurst requires 20 cycles for an add instruction, while P6 only required 10 cycles**
- **NetBurst uses a 128 entry register renaming rather than the reorder buffer with 40 entries used in P6**
- **NetBurst has 7 integer execution units instead of the 5 of P6 (one more integer ALU and one more address computation unit)**
- **Faster ALU (0.5 cycles vs. 1) and faster data cache (2 cycles vs. 3)**

P6/NetBurst differences (II)

- **Trace cache to improve instruction fetch performance**
- **More effective BTB: eight time larger and with an improved prediction algorithm**
- **SSE2 floating point instructions, making the Pentium 4 more effective when executing floating point programs.**

Performance comparis



The Pentium 4 is 1.2 faster than the Pentium III for integer programs, and 2,9 for floating point programs.

comparis