SISTEME CU MICROPROCESOARE (µP 2)

OBIECTIVE

- Studiul hardware-ului PC-ului
  - Pentium overview…
  - Memoria (Baza, Cache, Virtuala)
  - Interfete Programabile (Timer, PIC, DMAC, PIO)
  - (arhitectura, programare, aplicatii)
  - COM, USB, I2C, SPI, I2S, I3C, SPP
  - Bus-uri (ISA, PCI, PCIe)
  - Aplicatii
  - CNA/CAN
  - ESP32/8266+

https://edc.intel.com/content/www/us/en/design/ipla/software-development-platforms/client/platforms/alder-lake-mobile-p/intel-600-series-chipset-family-on-package-platform-controller-hub-pch-datash/introduction/
https://ocw.mit.edu/courses/6-004-computation-structures-spring-2017/pages/c21/c21s1/

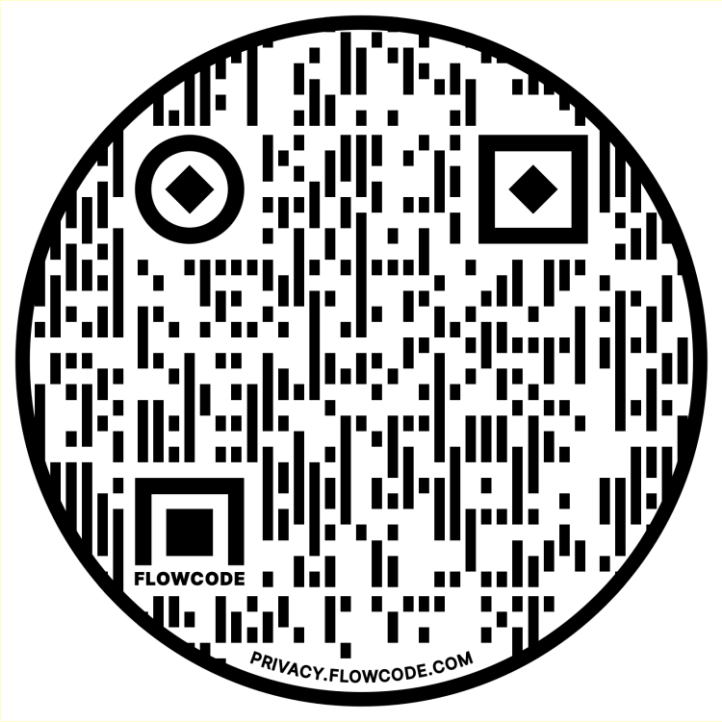
BIBLIOGRAFIE

- J.L. Hennessy, D.A. Patterson “Computer Architecture. A Quantitative Approach” Morgan-Kaufmann 2017
- Lupu, E. , Mesaroş, A. , Suciu, A.F. “MICROPROCESSORS Architectures and Applications” Ed. RISOPRINT Cluj-Napoca 2002, ISBN 973-656-392-8
- Lupu, E. SISTEME CU MICROPROCESOARE. Resurse hardware. Prezentare, programare şi aplicaţii. Ed. Albastră Cluj Napoca 2004, ISBN 973-650-109-4
- Tischer M., Jennerich B. *“LA BIBLE PC” PROGRAMMATION SYSTEME.* MICRO Application 1997
- Buchanan, W. *PC interfacing, communications and Windows Programing* Addison Wesley 1999
- N. Mathivanan Microprocessors, PC Hardware and Interfacing PHI Learning Pvt. Ltd., 2003
- *www.pcguide.com, www.intel.com* ,…..

C:\6-823-fall-2005\contents\lecture-notes\index.htm !!!!

- Evaluare: Nota Finala

  - 65% examen (teorie + probleme)
  - 35 % laborator (teste) note, media ≥4.5



Prezentari curs : http://elupu.utcluj.ro/ Psw: mp2

„Om liber este acela care nu are nevoie să spună nici o minciună.“

N. Iorga

C1. PENTIUM overview

I. Introducere

II. Pentium arhitectura

- https://studylib.net/doc/8963518/intel-core-processors
- http://developer.intel.com/software/products/itc/architec/ia32/pentdown.htm
- **http://bwrc.eecs.berkeley.edu/CIC/archive/cpu_history.html**

Processor	Year	Freq.	Nr. Transistors	Register Dimension	Data Bus	Memory Space	Cache	MIPS~
8086	1978	8 MHz	29 K	16-bit (GPRegisters)	16	1 MBytes	-	0.8
80286	1982	12,5 MHz	134 K	16-bit GP	16	16 MBytes	-	2.7
80386DX	1985	20 MHz	275 K	32-bit GP	32	4 GBytes	-	6
80486DX	1989	25 MHz	1,2 M	32-bit GP, 80-bit FPU	32	4 GBytes	8KB L1	20
Pentium	1993	60 MHz	3,1 M	32-bit GP, 80-bit FPU	64	4 GBytes	16KB L1	100
Pentium Pro	1995	150 MHz	5,5 M	32-bit GP, 80-bit FPU	64	64 GBytes	16 KBy L1; 256, 512KB L2	440
Pentium II	1997	266 MHz	7 M	32-bit GP, 80-bit FPU, 64-bit MMX	64	64 GBytes	32 KB L1; 256, 512 KB L2	440
Pentium III	1999	500 MHz	8,2 M	32-bit GP, 80-bit FPU, 64-bit MMX, 128-bit XMM	64	64 GBytes	32 KB L1; 256, 512 KB L2	700
Pentium 4	2001	1.4 GHz	42 M	32-bit GP, 80-bit FPU, 64-bit MMX, 128-bit XMM	64	64 GBytes	20 KBL1; 512 kB L2	3K (1.5GHz)
Pentium 4 Extreme Edition	2004	3.2-3.4GHz	55 M				+ 2MB L3	13k

.....

http://bwrc.eecs.berkeley.edu/CIC/

https://www.slideshare.net/Slideshare/what-to-upload?next_slideshow=48674690

PENTIUM: Caracteristici de baza

1. Superscalar architecture
2. Pipelined floating-point unit
3. Separate code and data caches
4. Bus cycle pipelining
5. Internal parity checking
6. Execution tracing
7. IEEE 1149.1 boundary scan
8. Virtual mode extensions
9. Advanced power management feature
10. On-chip local APIC (Advanced Programmable Interrupt Controller) device
11. Dynamic branch prediction
12. Improved instruction execution time

B.Kannan, RIT

13. 64-bit data bus
14. Address parity
15. Functional redundancy checking and lock-step operation
16. Performance monitoring
17. System management mode
18. Dual processing support
19. Fractional bus operation

Table 4.19 History of Intel microprocessors over three decades

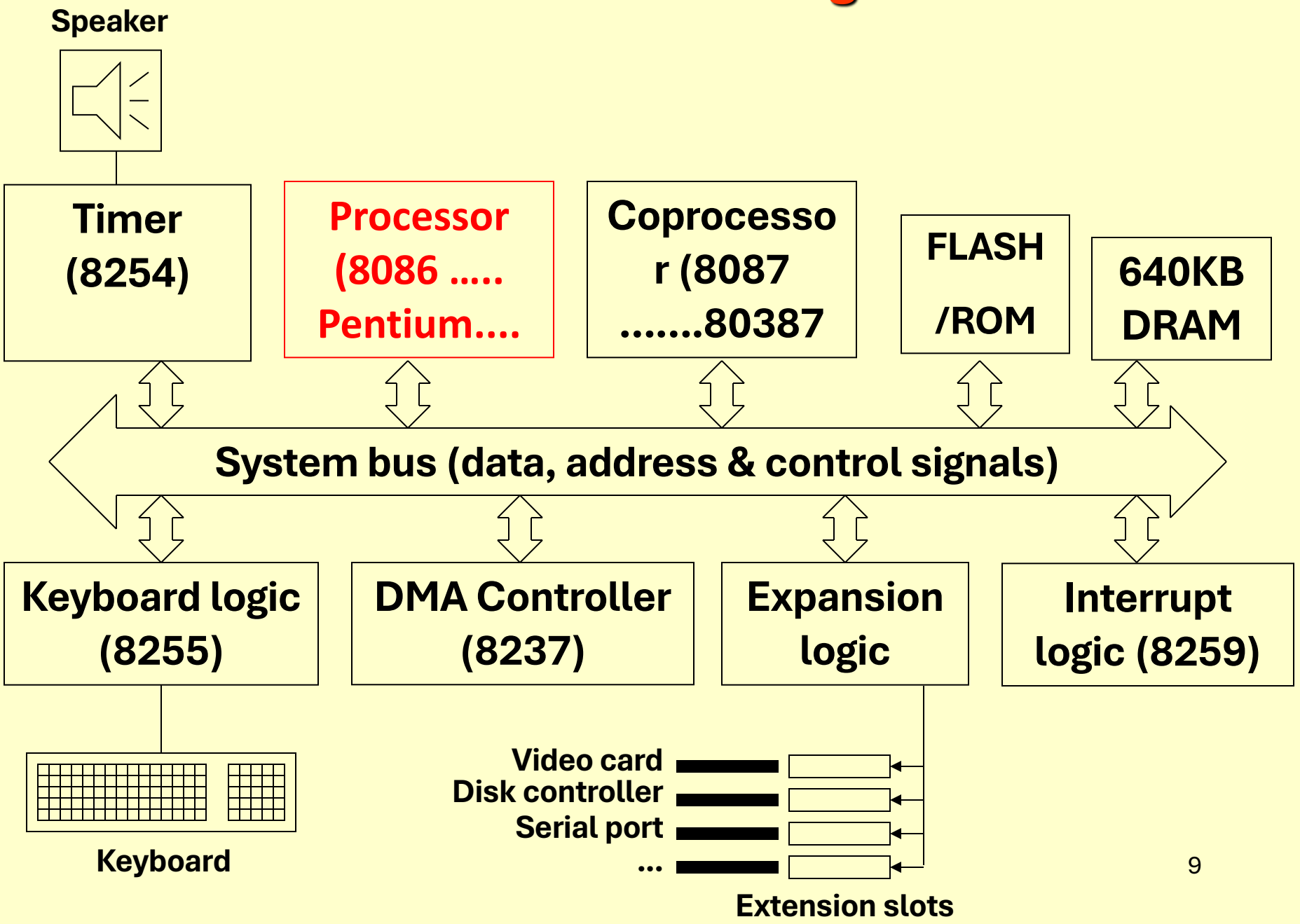
Processor	Year	Feature Size (µm)	Transistors	Frequency (MHz)	Word size	Package
4004	1971	10	2.3k	0.75	4	16-pin DIP
8008	1972	10	3.5k	0.5–0.8	8	18-pin DIP
8080	1974	6	6k	2	8	40-pin DIP
8086	1978	3	29k	5–10	16	40-pin DIP
80286	1982	1.5	134k	6–12	16	68-pin PGA
Intel386	1985	1.5–1.0	275k	16–25	32	100-pin PGA
Intel486	1989	1–0.6	1.2M	25–100	32	168-pin PGA
Pentium	1993	0.8–0.35	3.2–4.5M	60–300	32	296-pin PGA
Pentium Pro	1995	0.6–0.35	5.5M	166–200	32	387-pin MCM PGA
Pentium II	1997	0.35–0.25	7.5M	233–450	32	242-pin SECC
Pentium III	1999	0.25–0.18	9.5–28M	450–1000	32	330-pin SECC2
Pentium 4	2001	0.18–0.13	42–55M	1400–3200	32	478-pin PGA

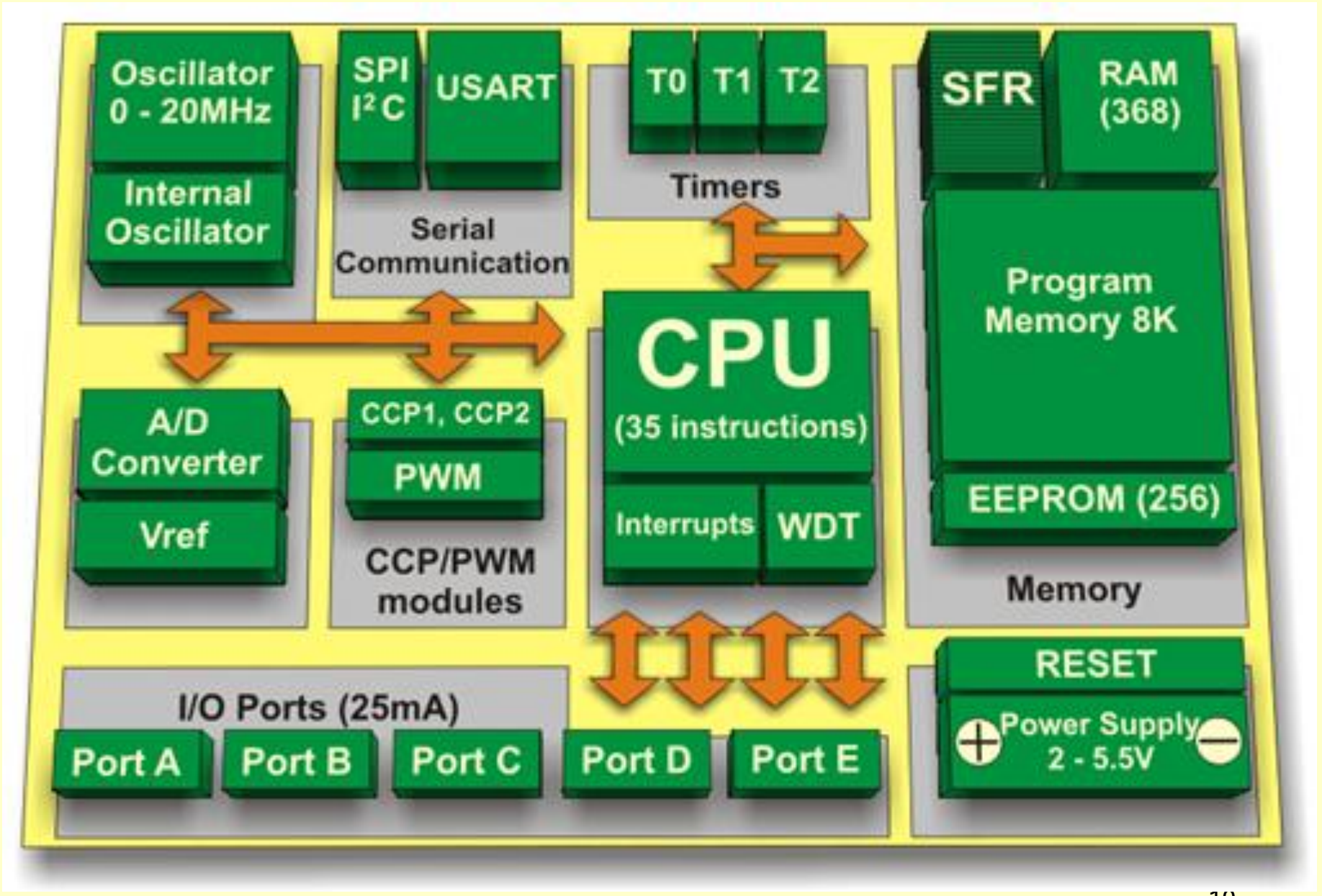
- $>10^4$ increase in transistor count & clock frequency over 30 years!

x86 PROCESSORS (from Intel)

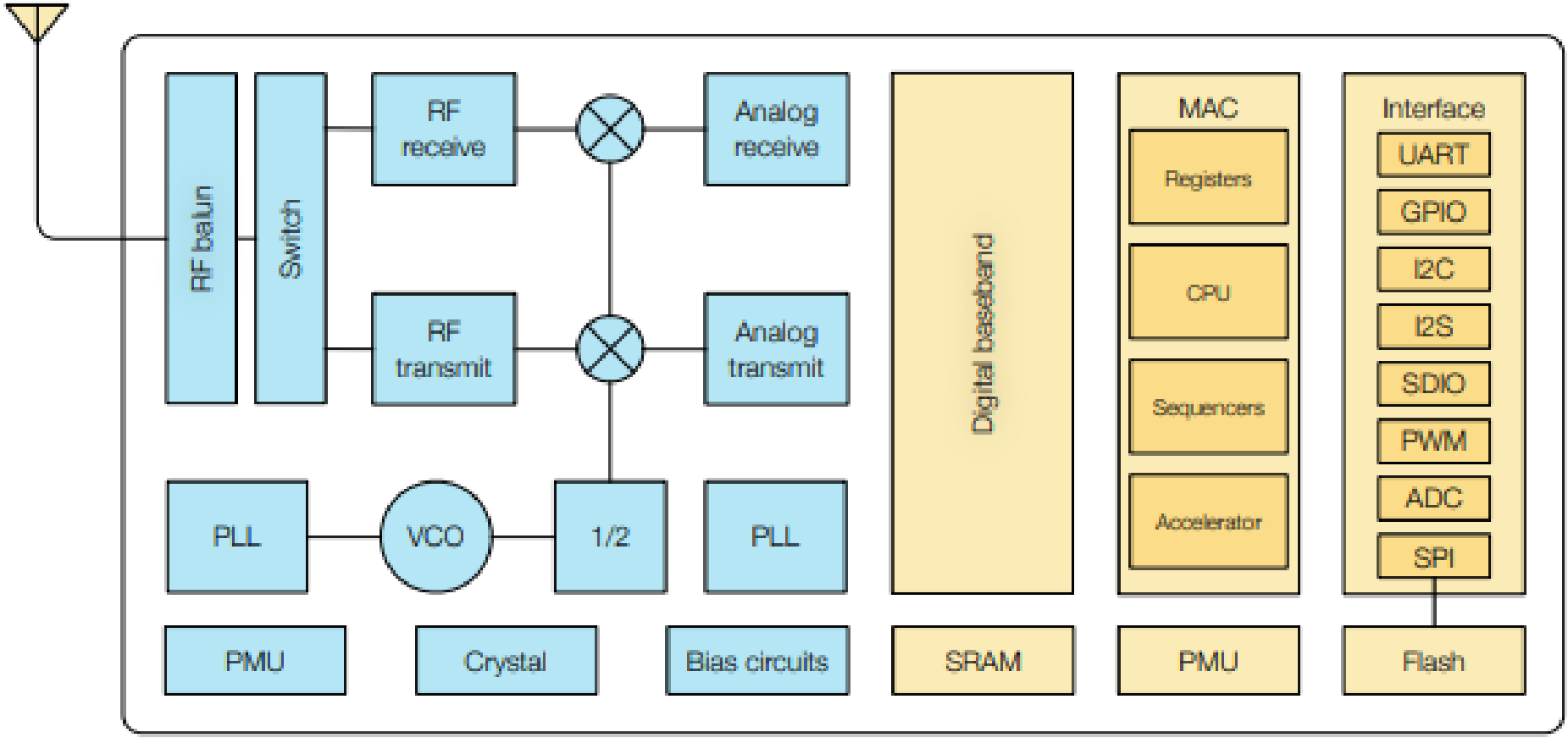
Bits	Family	Clock Speed (approx.)	Bus Size (bits)	Max RAM	Storage Range	OS
64	Xeon	4.3GHz	64	3TB	500GB-10TB	Windows: 10, 8, 7 XP, 2000 NT, 98 95, 3.x Linux Mac OS X SCO Unix Solaris DOS DR DOS OS/2 Misc DOS Multiuser
	Core i9	3.3GHz		128GB		
	Core i3, i5, i7	3.3GHz		64GB		
	Core 2 Duo	2.6GHz				
	Pentium 4	3.8GHz				
	Pentium D	3.4GHz				
32	Core Duo	2.2GHz				
	Pentium 4	2.8GHz				
	Xeon	3.2GHz				
	Celeron	2.4GHz		4GB	500MB-60GB	
	Pentium III	1.2GHz				
	Pentium II	450MHz				
	Pentium Pro	233MHz		64GB		
	Pentium	200MHz		4GB		
	486DX	100MHz	32		200 - 500MB	
	486SX	40MHz				
	386DX	40MHz			60 - 200MB	
	386SX	33MHz				
	386SL	25MHz				
16	286	12MHz	16	16MB	20-80MB	DOS DR DOS Win 3.x OS/2 1.x
	8086	10MHz		1MB	10-20MB	DOS DR DOS
	8088	5MHz	8			

Arhitectura PC-AT originala





Schema bloc a unui microcontroller PIC



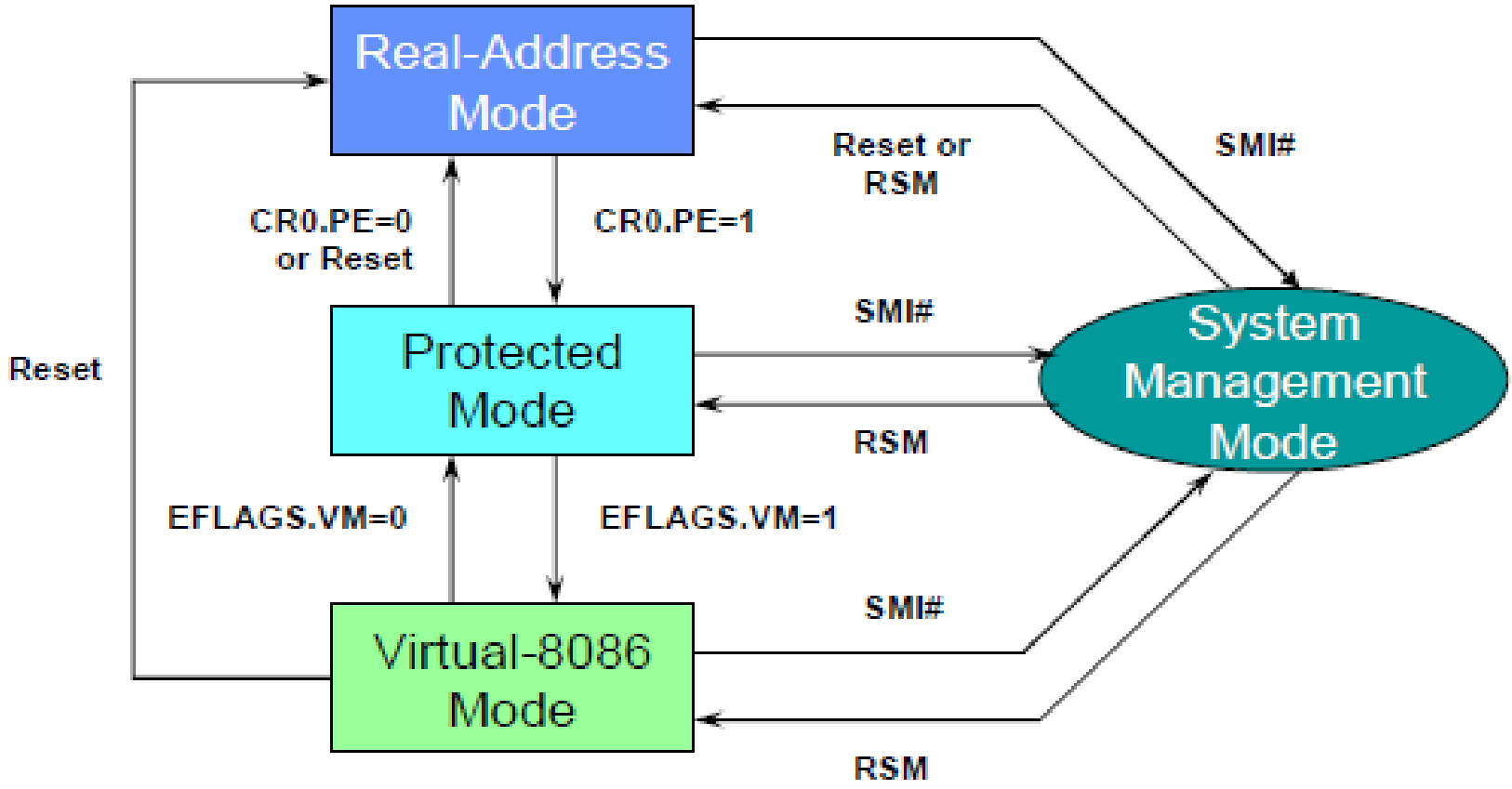
ESP8266 Functional Block Diagram (IOT chip)

https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf

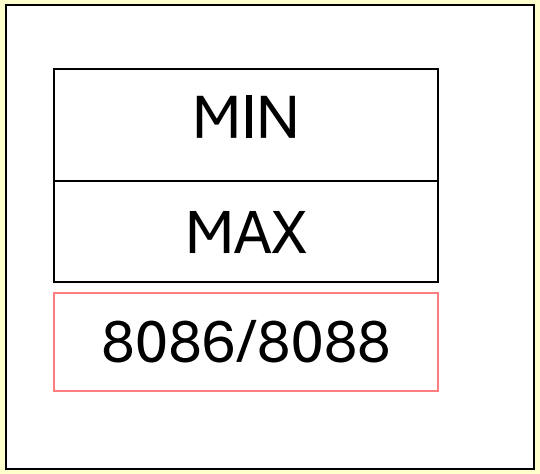
	Microcontroller	Microprocessor
Design complexity	Low	High
Clock speed	Slow	Fast
Operating system	No	Yes
Processing speed	Low	High
Power consumption	Low	High
Memory	Small / Internal	Large / External
I/O pins	Yes	No
Number of bits	8-32 bits	32-64 bits
Cost	Low	High

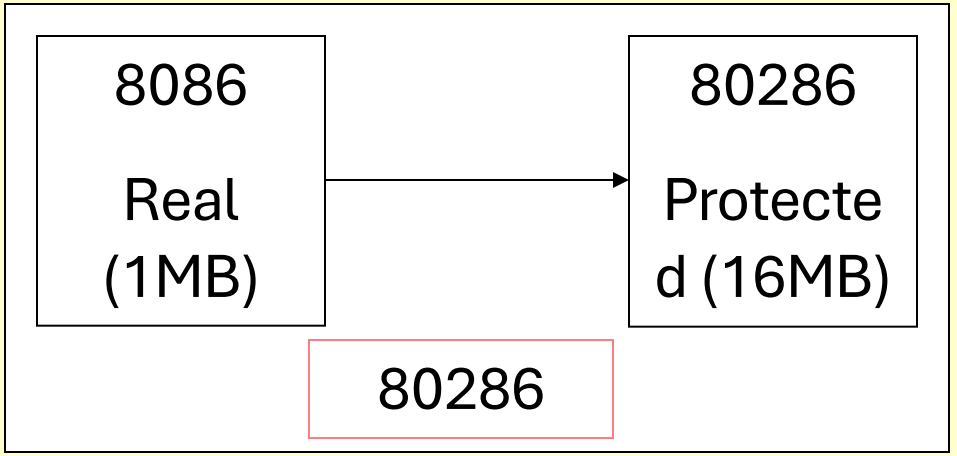
Table 1 – Comparison of a microcontroller versus a microprocessor

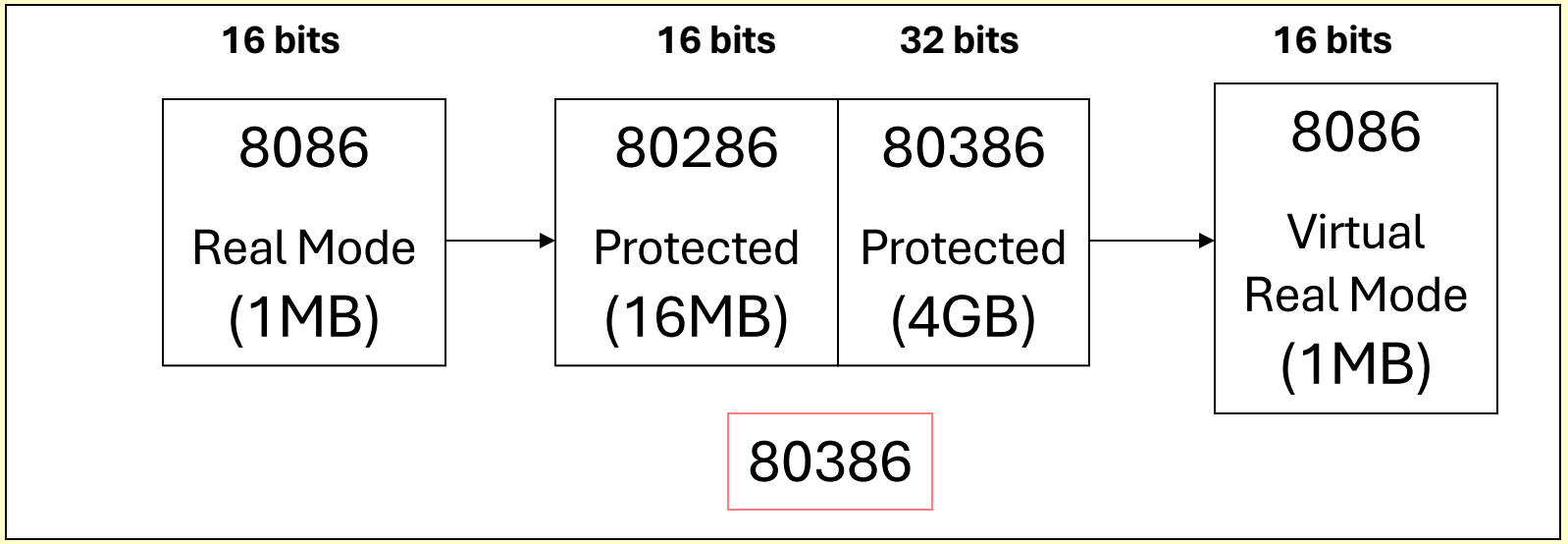
Modurile de operare ale µP X86



- **16 bit**
  - Real Mode (8086 and up)
  - Protected mode (286 and up)
  - Virtual 8086 mode (386 and up)
- **32 bit**
  - Protected mode (386 and up)
  - Paging enabled/disabled
- **64 bit modes are not shown here**


MIN
MAX
8086/8088

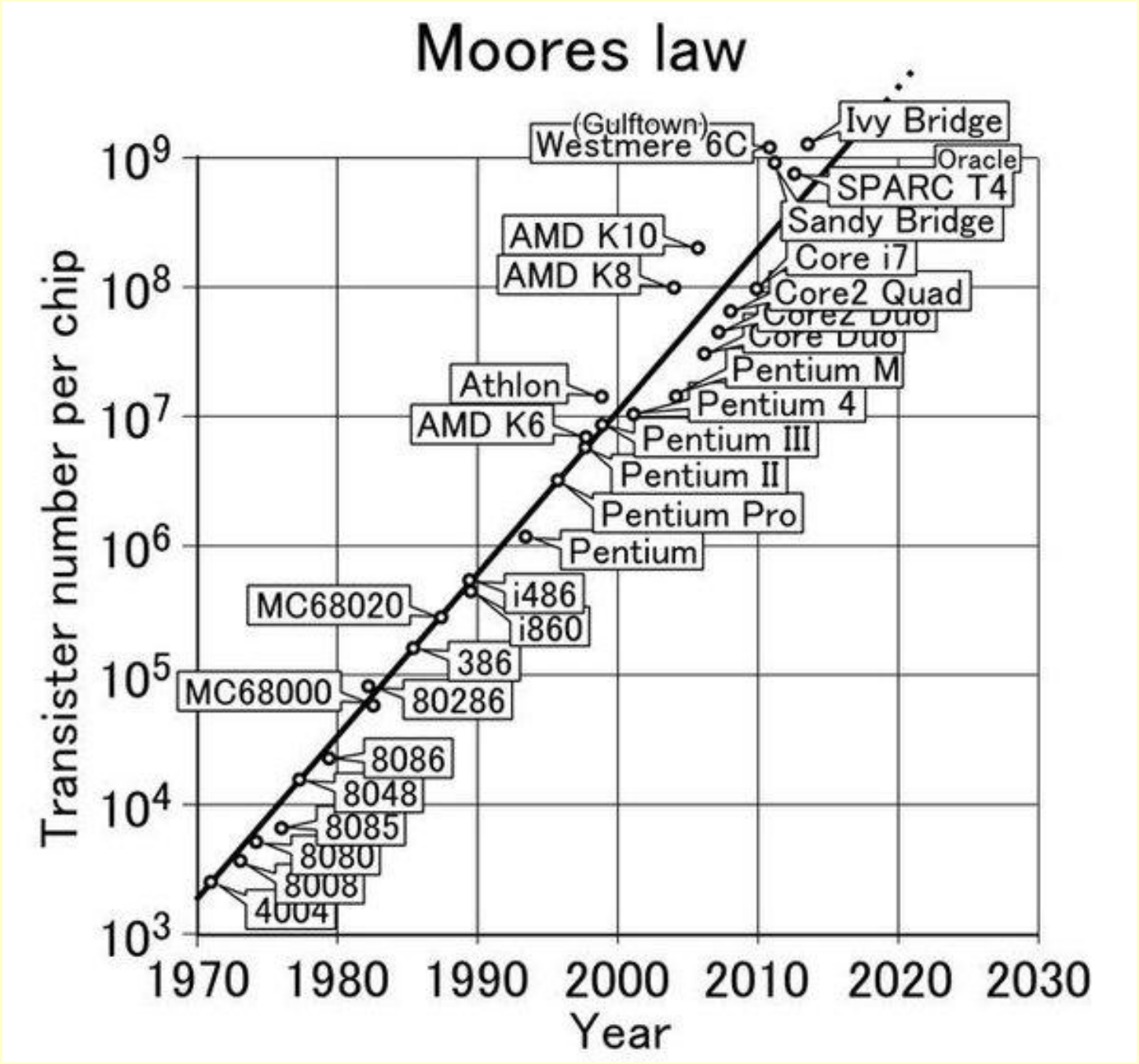

8086
Real
(1MB)
80286
Protecte
d (16MB)
80286


16 bits
16 bits
32 bits
16 bits
8086
Real Mode
(1MB)
80286
Protected
(16MB)
80386
Protected
(4GB)
8086
Virtual
Real Mode
(1MB)
80386

CUM se îmbunătăţesc PROCESOARELE ?

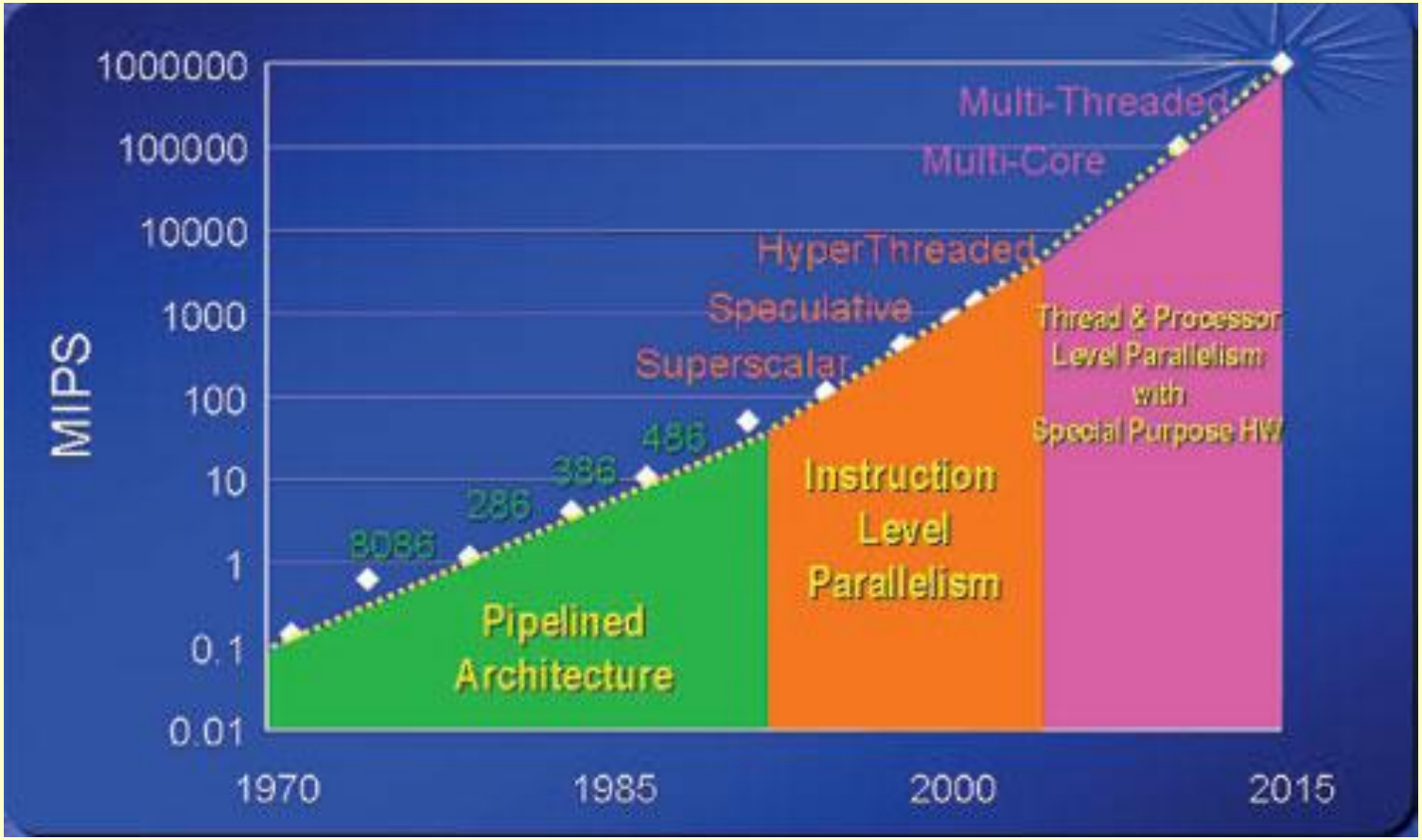
Mijloace de a creste performanta µP, prin:

- cresterea frecventei CLK (Overclocking)
- imbunatatirea arhitecturii (creste nr. tranzistoare)
- setul de instructiuni
- dimensiunea datelor/registrelor 16/32/64
- cresterea capacitatii memoriei cache


Moores law
Transister number per chip
Year
10^3
10^4
10^5
10^6
10^7
10^8
10^9
1970
1980
1990
2000
2010
2020
2030
4004
8008
8080
8085
8048
8086
80286
386
i860
i486
Pentium
Pentium Pro
Pentium II
Pentium III
Pentium 4
Pentium M
Core Duo
Core2 Duo
Core2 Quad
Core i7
Sandy Bridge
SPARC T4
Oracle
Ivy Bridge
(Gulftown)
Westmere 6C
AMD K10
AMD K8
Athlon
AMD K6
MC68020
MC68000

Optimizarea Executiei

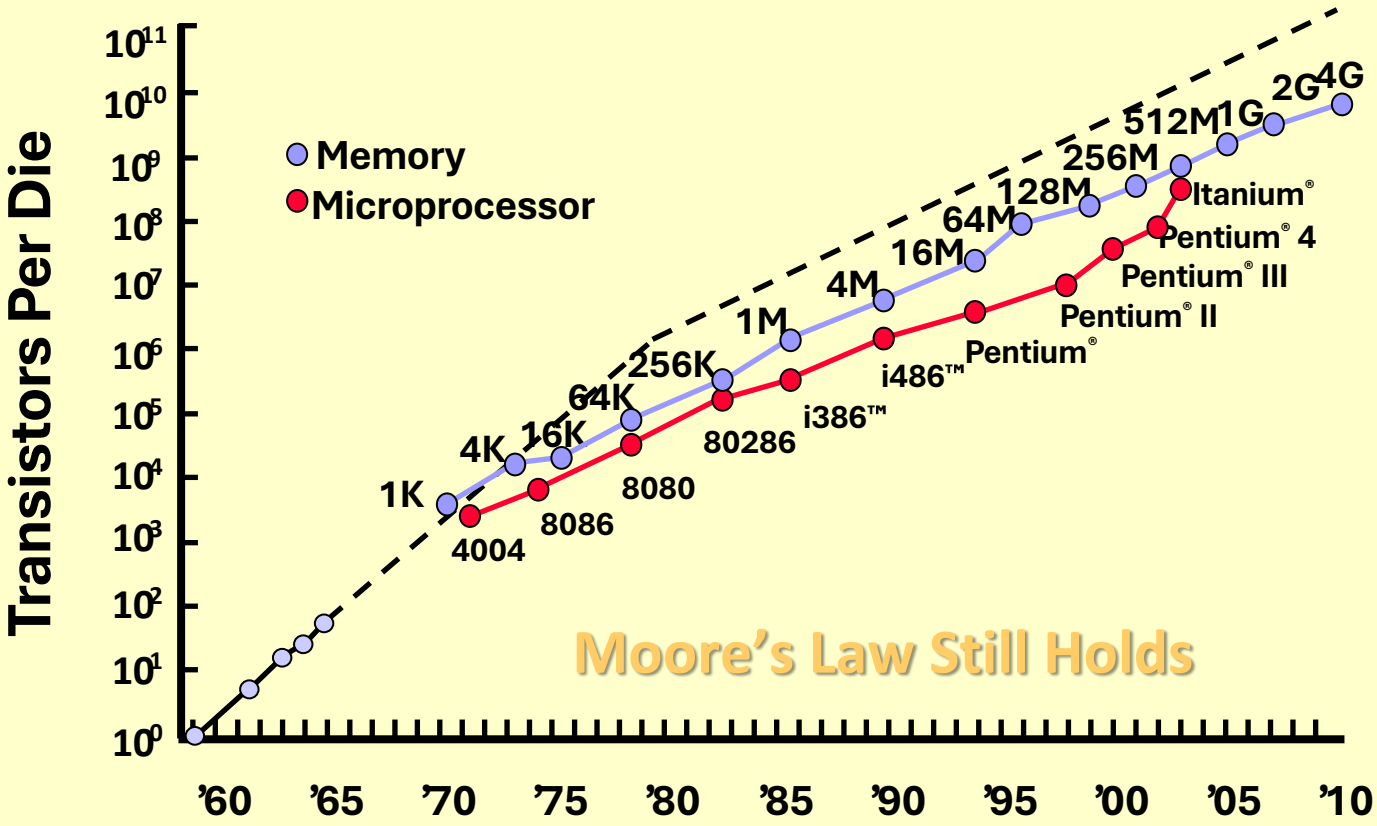
- Instructiuni mai puternice
- Tehnici de optimizarea executiei (pipelining, branch prediction, execution of multiple instructions, reordering instruction stream, OOO, hyper-threading, multicore *etc*.)

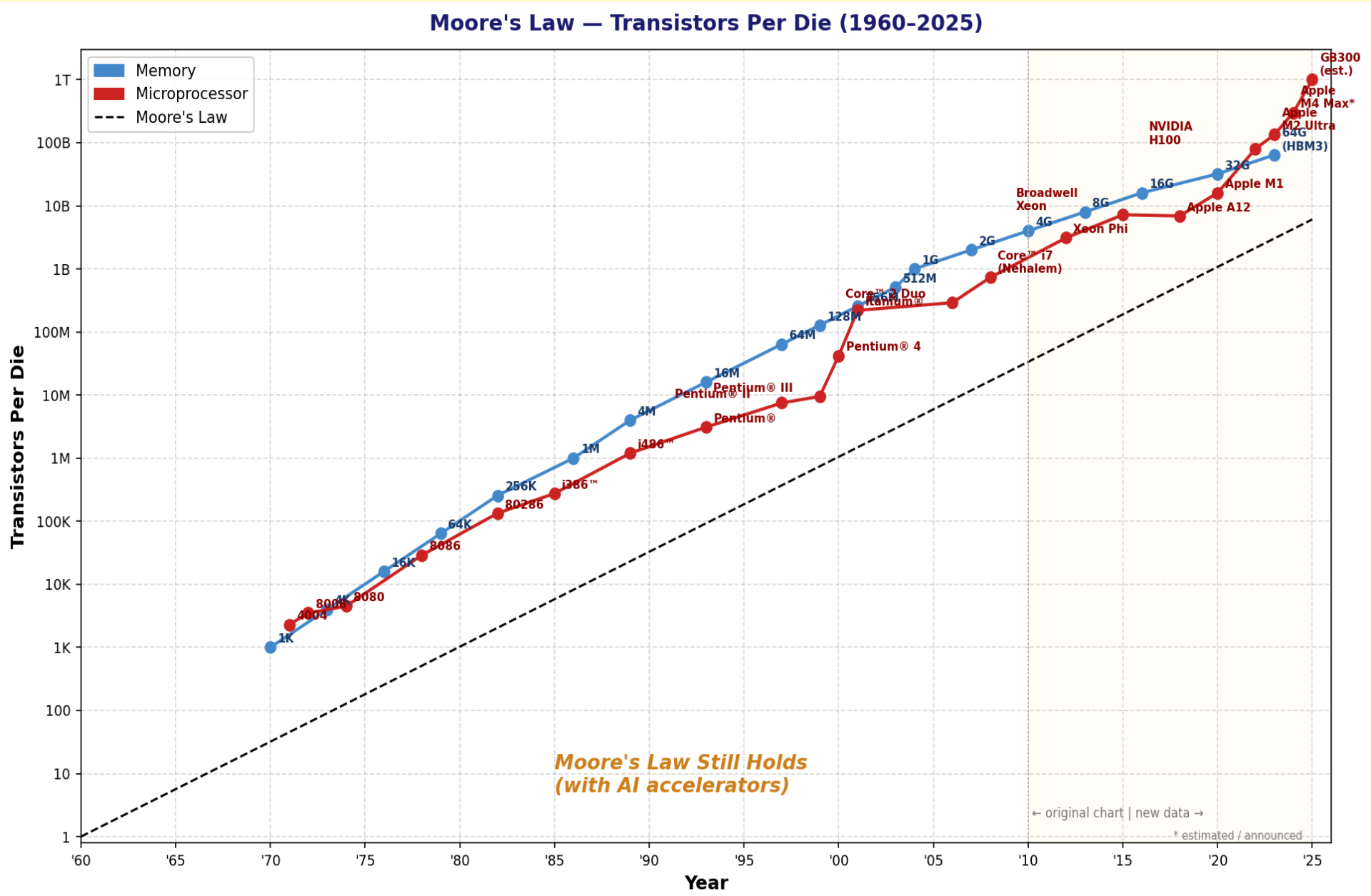


Adapted from Johan De Gelas, *Quest for More Processing Power*, AnandTech,.

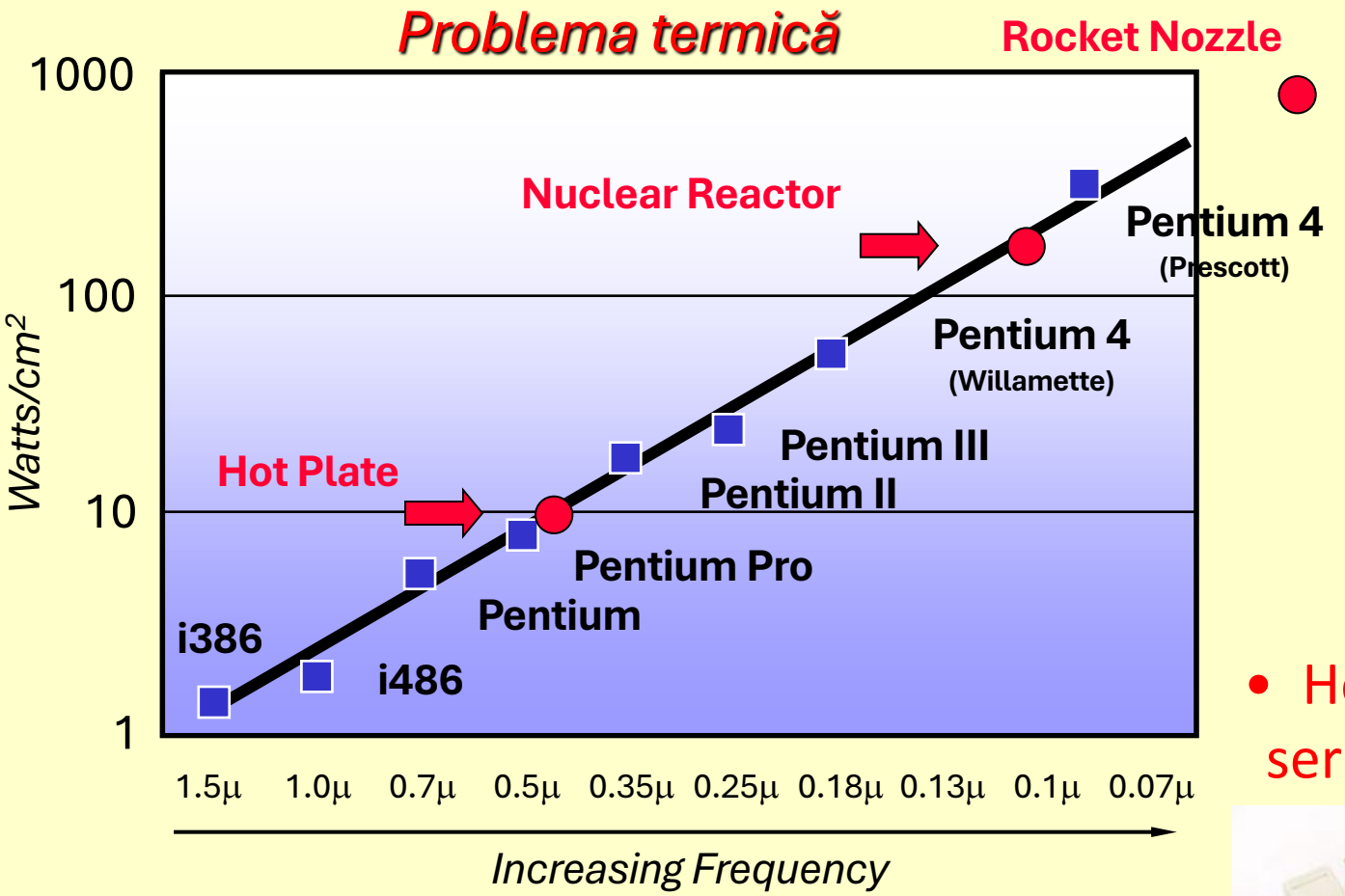
Cache marit

- On-chip caches amelioreaza disparitatea crescuta intre viteza procesorului si latenta/banda memoriei
- On-chip caches continua sa creasca in marime si sa reduca disparitatile in performanta subsistemelor computerului




Moore's Law — Transistors Per Die (1960–2025)
Memory
Microprocessor
Moore's Law
Transistors Per Die
Year
1T
100B
10B
1B
100M
10M
1M
100K
10K
1K
100
10
1
'60
'65
'70
'75
'80
'85
'90
'95
'00
'05
'10
'15
'20
'25
1K
4K
16K
64K
256K
1M
4M
16M
64M
128M
512M
1G
2G
4G
8G
16G
32G
64G
(HBM3)
4004
8008
8080
8086
80286
i386™
i486™
Pentium®
Pentium® II
Pentium® III
Pentium® 4
Core™2 Duo
Itanium®
Core™ i7
(Nehalem)
Broadwell
Xeon
Xeon Phi
NVIDIA
H100
Apple A12
Apple M1
Apple
M2 Ultra
Apple
M4 Max*
GB300
(est.)
Moore's Law Still Holds
(with AI accelerators)
← original chart | new data →
* estimated / announced

Limitari ale cresterii performantelor :



- Heat sinks in 6XX series Pentium 4s

- Pentium: 60 MHz.....3,800 MHz in 12 years
  - Resulted in ~80% performance increase

The Resulting Shift to Multicore

Perf = 1
Power = 1

Perf = 2
Power = 4

Perf = 4
Power = 4

•Calculatorul standard este un dispozitiv care are o UCP care execută instrucțiuni unice, instrucțiuni foarte simple, foarte repede.

•Prin executia atâtor de multe instrucțiuni, într-un timp atât de scurt se creează iluzia de a face lucruri complicate.

•Deci, până în anii 1990, majoritatea calculatoarelor au fost bazate foarte puternic ***pe modelul secvențial simplu*** și în anii 1990 computerele au devenit mai puternice făcând ca ***frecventele de ceas*** să crească mai mult și a existat efectiv o cursă de crestere a frecventei de ceas între principalii producători.

•Dar apoi, la începutul anilor 2000, a devenit clar că, în principal din motive termice, ***nu puteau să păstreze, făcând mașinile secvențiale*** să meargă mai repede, pur și simplu s-ar fi încălzit prea tare. Și așa toți au încetat să încerce să avanseze în acest fel și, în schimb, au ales să pună ***mai multe nuclee, mai lente pe un acelasi chip, atunci avem modelul paralel.***

•Ceea ce vedem acum este doar acel model paralel care continuă să se extindă la niveluri crescânde de paralelism. Acest lucru este valabil pentru *desktop și pentru smartphone.* Aproape nu mai există mașini secvențiale cu un singur fir de executie. Chiar și smartphone-urile au procesoare duale, quad sau octo și grafică cu procesoare multi-core.

GCSE Bitesize Computer Science- Steve Furber on CPU cores

Arhitectura x86 IA-32 – Caracteristici

- Microarhitectura si *ISA (instruction set architecture*) constituie impreuna arhitectura calculatorului

Microarhitectura **+**

ISA (Instruction Set Architecture)

ARHITECTURA COMPUTERULUI

✓ Dimensiuni de Date Multiple si Metode de Adresare

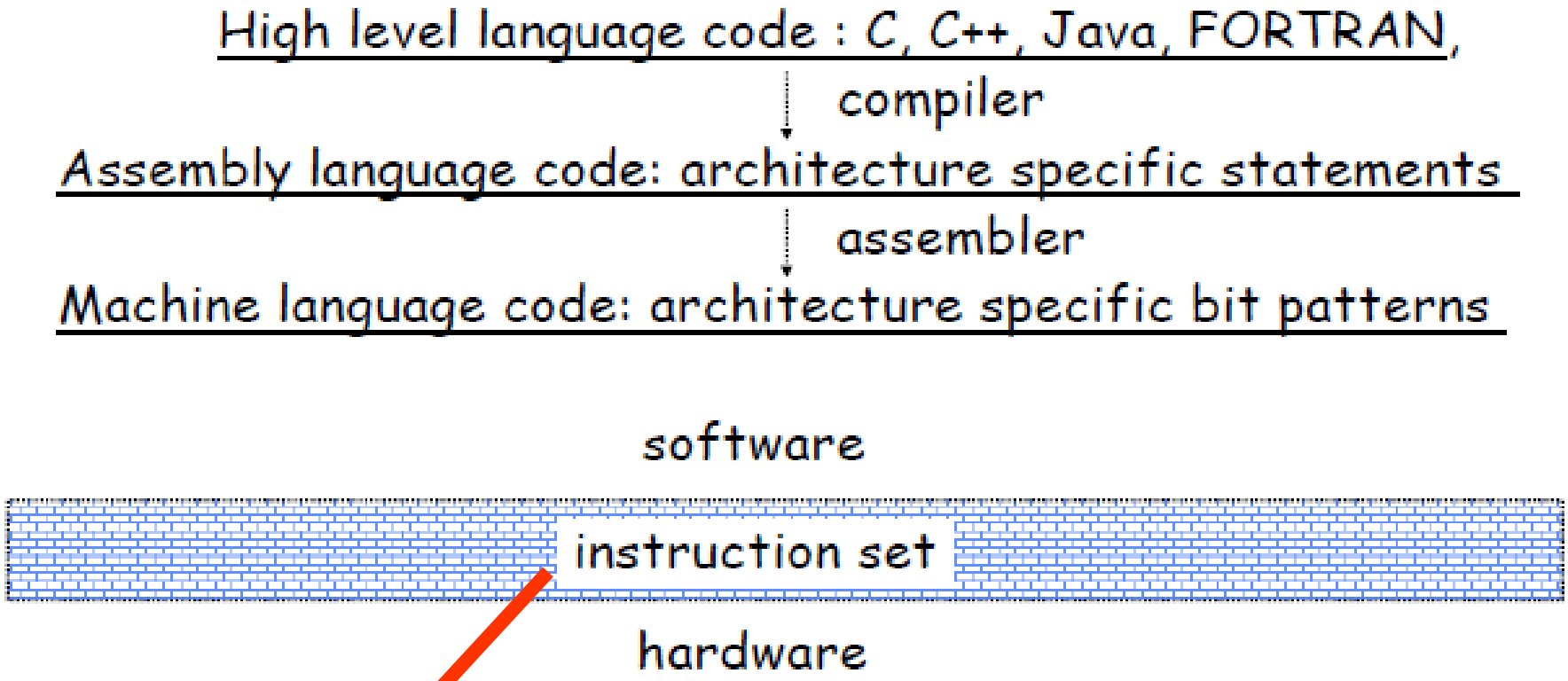
- Generatiile recente sunt optimizate pentru moduri pe 32-biti

✓ Numar limitat de Registre

- Procedura de apel (call) orientata pe stiva si set de instructiuni in virgula flotanta
- Programele acceseaza intensiv memoria (41%)

✓ Lungimi variabile ale Instructiunilor (CISC)

- Primii octeti descriu operatia si operanzii
- Ceilalti octeti dau datele imediate sau deplasamentul (offset) de adresa
- Media 2.5 bytes/instr.



- Data transfer
- Common ALU operations
- Control flow
- String
- FP, MMX and SSE instructions
- OS support
- I/O instructions
- “Exotic” instructions (DAA, XLAT etc.)

x86 - Complex Instruction Set

CISC dezavantaje:

- Multe instructiuni sunt complicate, ele se descompun in secvente de micro-pasi (microprograme)
- Acesti pasi reprezinta Micro-cod, stocat in ROM pe chip-ul procesorului
- Micro-codul din ROM implica: timp de acces, dimensiune chip, putere
- Necesita ROM suplimentar si logica de decodare (tranzistoare)

RISC: “Less is More”

- RISC = Reduced Instruction Set Computer
- 20/80 Rule: 20% dintre instructiuni iau 80% din timpul de executie
- Uneori executia unei secvente de instructiuni simple se face mai rapid decat o singura instructiune complexa cu acelasi efect

RISC Background

- Reducerea setului de instructiuni simplifica decodarea

  Smaller Instruction Set -> Simpler Logic -> Smaller Logic -> Faster Execution

- *Eliminare microcod* –> executia instructiunilor este cablata
- *Pipeline ptr. instructiuni* - decodarea si executia – realizeaza mai multe operatii in paralel (ILP)
- *Load/Store a*rchitecture – numai instr. load si store pot accesa memoria

– Celelalte instructiuni lucreaza cu registrii interni ai procesorului

– Este necesar ptr. executia single-cycle – unitatea de executie (ALU) nu poate astepta citirea/scrierea datelor

- *Creste numarul registri interni datorita* arhitecturii Load/Store
- Registrii sunt de uz general (GPR) si au mai putin functii specifice
- Design-ul compilatorului se face odata cu proiectarea procesorului RISC
- Compilatorul trebuie să tina cont de arhitectura procesorului pentru a produce cod care poate fi executat în mod eficient

II. Caracteristici Pentium

- 0.8 µm technology
- 60MHz CLK
- 100 MIPS
- 100% compatibility with earlier generations
- 32 b registers GP
- 32 b Address BUS
- 64 b Data BUS
- super-scalar architecture
- 2 pipeline (u,v) , ALU
- 2 simple instr./clk
- 2 cache mem. (8k+8k)
- fast FPU look-up tables
-Dynamic Branch Prediction

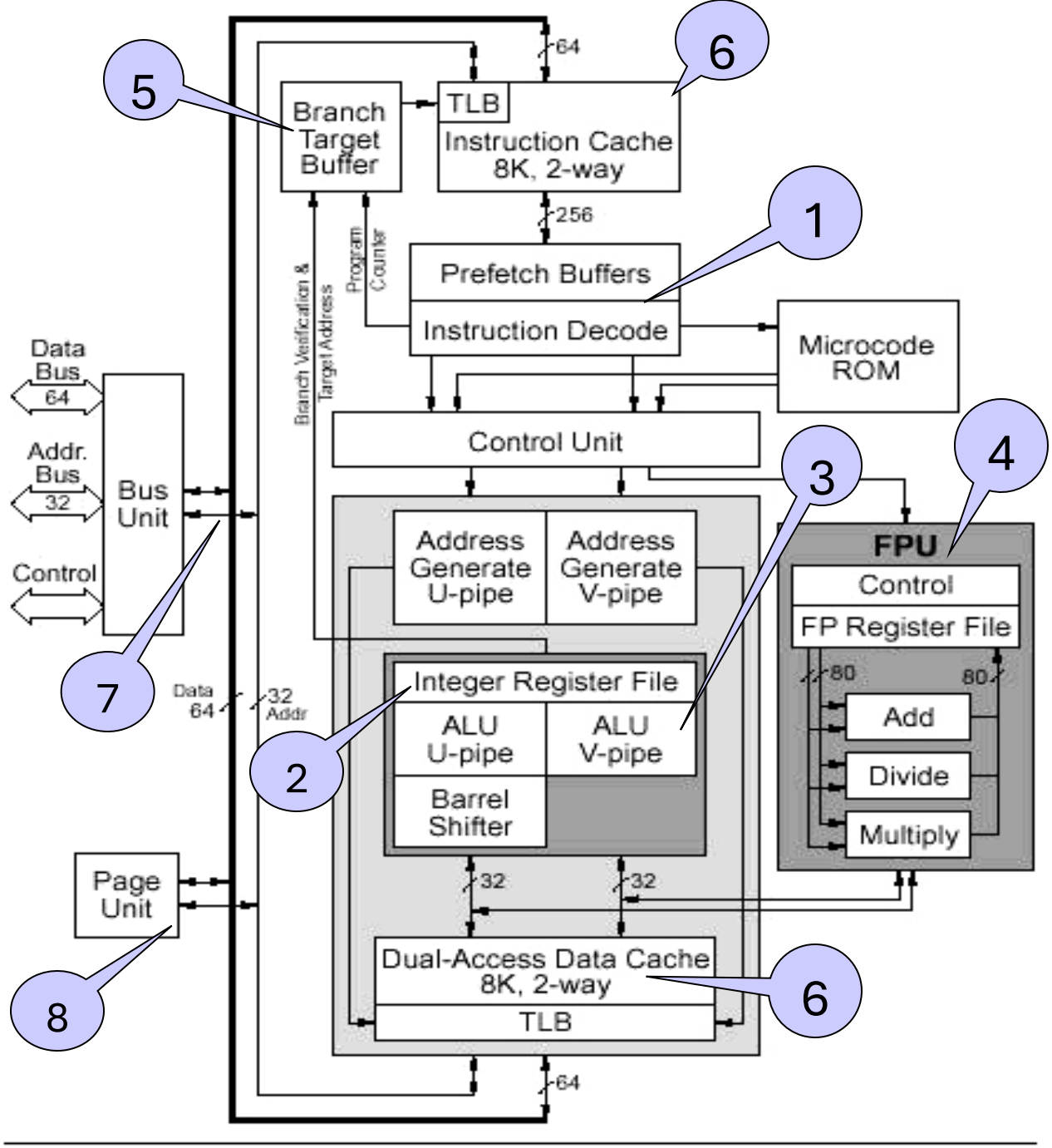
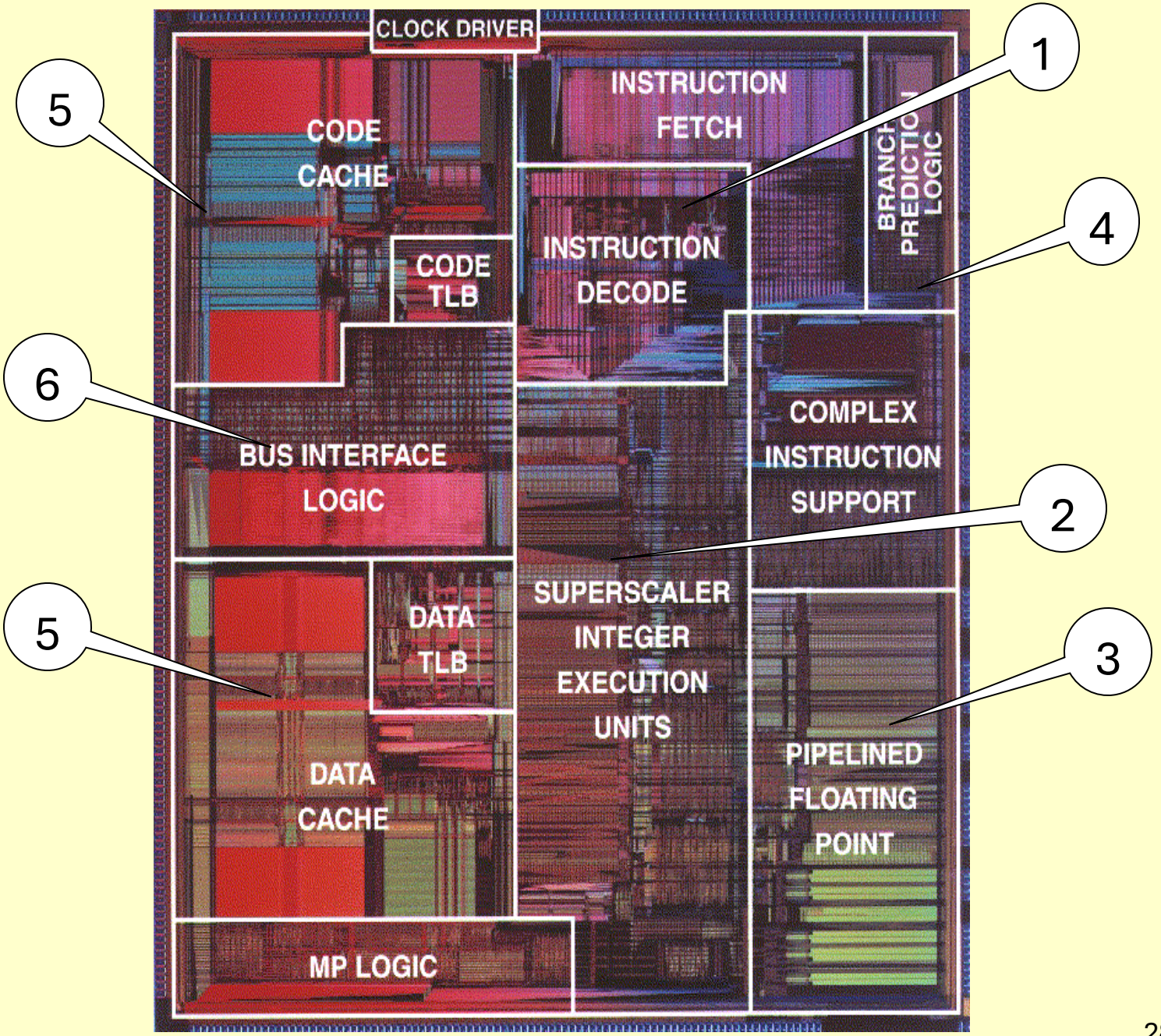


Figure 1. Pentium block diagram. (Pentium microarchitecture)



http://bwrc.eecs.berkeley.edu/CIC/die_photos/pentium.gif

PENTIUM – RISC Like Architecture

•Microprocesorul Pentium (lansat în 1994) a adoptat un design superscalar de tip RISC-like, păstrând compatibilitatea cu setul de instrucțiuni x86 CISC, având două conducte de execuție integer numite „U” (superioară) și „V” (inferioară) – practic două „motoare” paralele.

Execuție cu 2 conducte/pipeline

•**Instrucțiuni simple (conducte U/V)**: Gestionate prin logică hard-wire în ambele conducte simultan pentru execuție rapidă a operațiilor comune – operații integer ALU (adunare, deplasare, înmulțire), salt și transfer de date se execută într-un ciclu de ceas, cu debit maxim de până la 2 instrucțiuni/ciclu. Pipeline U gestionează toate instrucțiunile simple, iar pipeline V suportă un subset (fără deplasări sau registre parțiale).

•**Instrucțiuni complexe (motor microcod)**: Operațiile rare sau complicate (operatii pe șiruri, manipulare de biți, operatii flotante dincolo de FMUL/FADD de bază) sunt decodificate în *secvențe de micro-operații* RISC-like simple printr-un ROM cu microcod pe chip (aprox. 8K micro-op), apoi trimise către conducte – evitând complexitatea hardware a decodorului.

https://www.eeeguide.com/internal-architecture-of-pentium-processor/

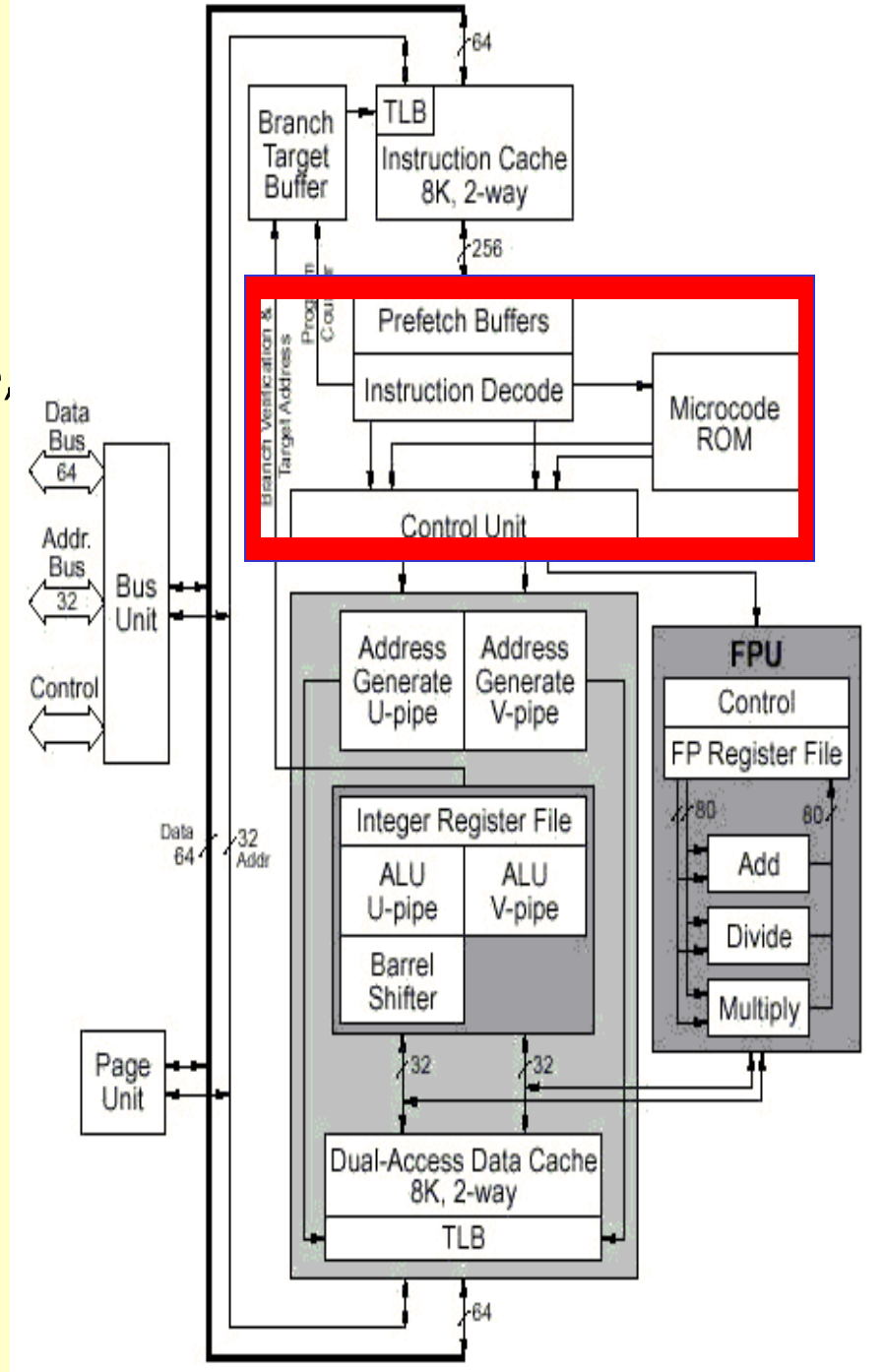
Criterii de Performanță

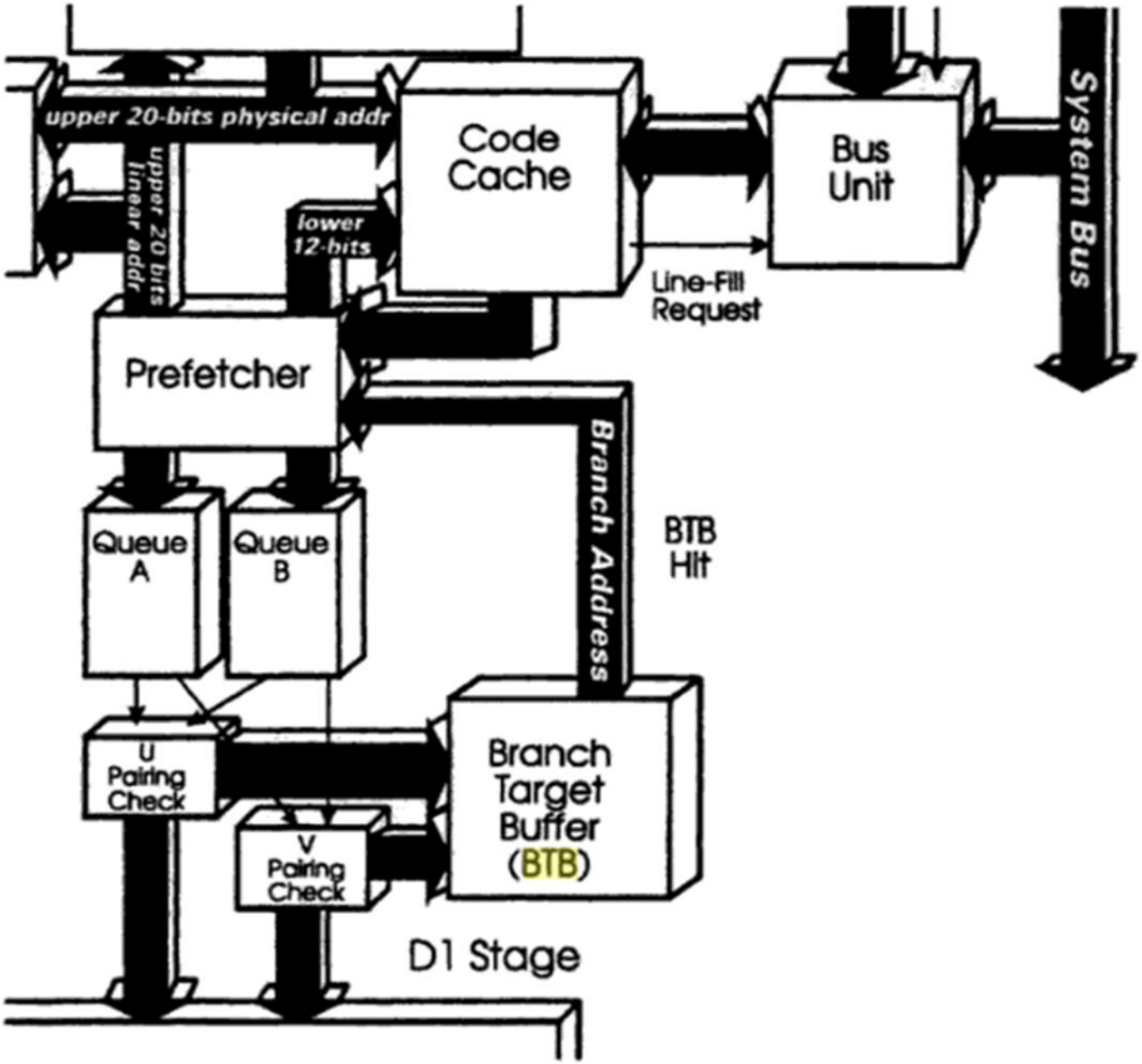
Criteriu	Motor Simplu (Hard-wired)	Motor Microcod
Debit	2 IPC maxim (dual-issue)	1-1.5 IPC med. (multi-ciclu)
Latență	1 ciclu (ex. ADD)	5-20+ cicluri (ex. REP MOVS)
Instrucțiuni	~60% din set x86 (60/143)	Restul ~40%
Etape Conductă	5 (fetch-decode-execute-writeback)	Decodare micro-op adaugă 2-3 etape
Viteză Ceas (CLK)	60-200 MHz (P5/P54C)	Aceeași, dar serializată

- Această abordare hibridă a oferit ~2x performanță față de 486 la aceeași frecvență, combinând eficiența RISC (cale simplă) cu universalitatea CISC.

Prefetch Buffers

- Are 4 prefetch buffers si funcționează ca *două perechi independente.*
- Când instrucțiunile sunt preluate din cache, acestea sunt plasate într-un singur set de prefetch- buffere
- Celălalt set este utilizat când se prezice o instrucțiune de salt
- Buffer-ul Prefetch trimite o pereche de instrucțiuni către decodorul de instrucțiuni




upper 20-bits physical addr
upper 20 bits linear addr
lower 12-bits
Code Cache
Bus Unit
System Bus
Line-Fill Request
Prefetcher
Branch Address
BTB Hit
Queue A
Queue B
U Pairing Check
V Pairing Check
Branch Target Buffer (BTB)
D1 Stage

- Instruction Decode Unit

- Apare la 2 nivele – Decode1 (D1) si Decode2(D2)
- **D1** verifica daca instructiunile pot fi asociate/ imperechiate
- **D2** calculeaza adresa operanzilor rezidenti in memorie

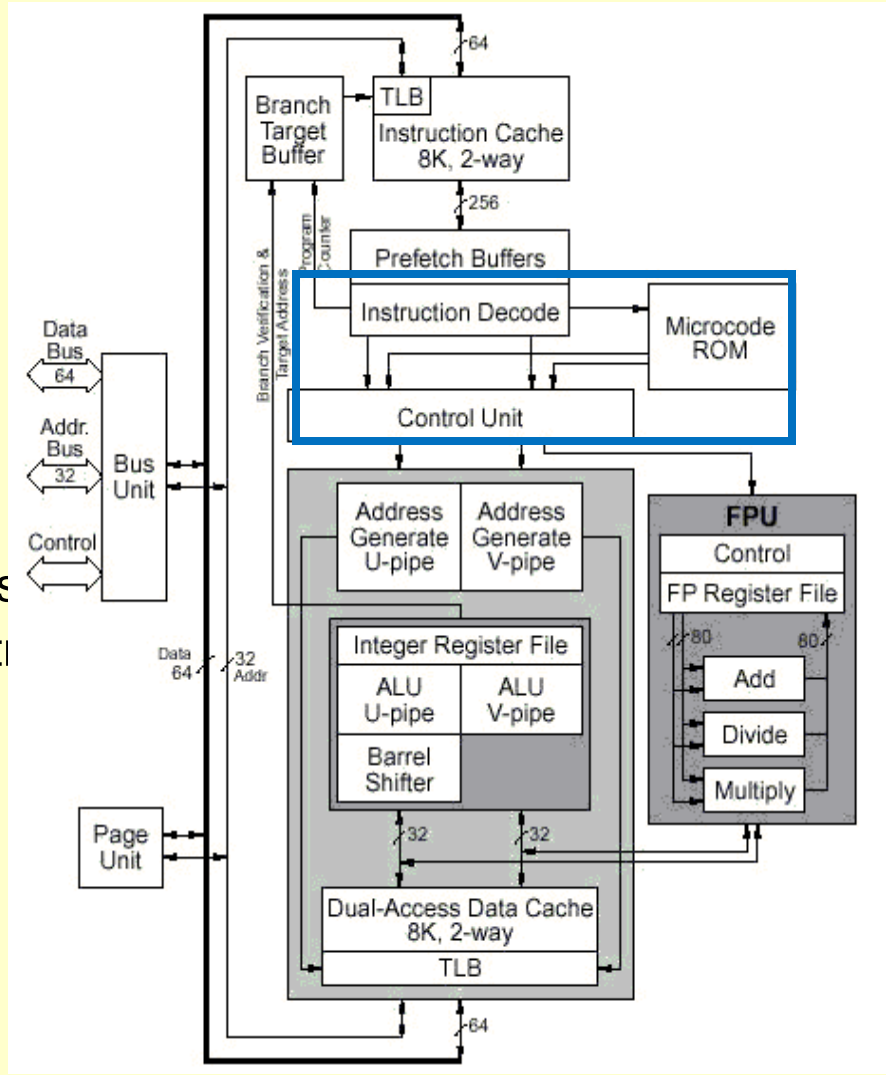
- Control Unit

- Aceasta unitate interpreteaza instructiunea si microcodului furnizat de catre Instruction Decode Unit
- Aceasta manipuleaza exceptiile, breakpoints intreruperile si controleaza pipeline-urile int si secventele FP floating point

- Microcode ROM :

- Stocheaza secventele de microcod

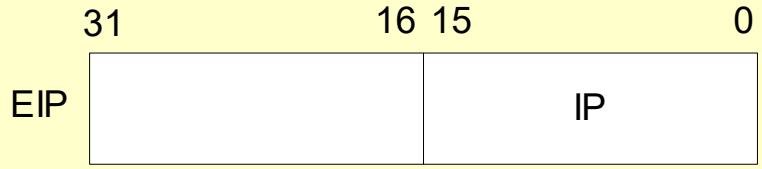
Care corespund instructiunilor complexe



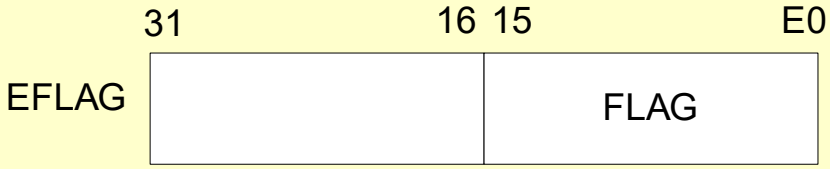
2. Set Registre

- Registrii Generali+de adresare (EAX, EBX, ...ESI, EDI)
- Registrii Segment (ES, DS, CS, SS, FS, GS)
- Registrii de Control (CR0-CR4)
- Registrii Managment Memorie
- Registrii Debbug
- EFLAGS Register

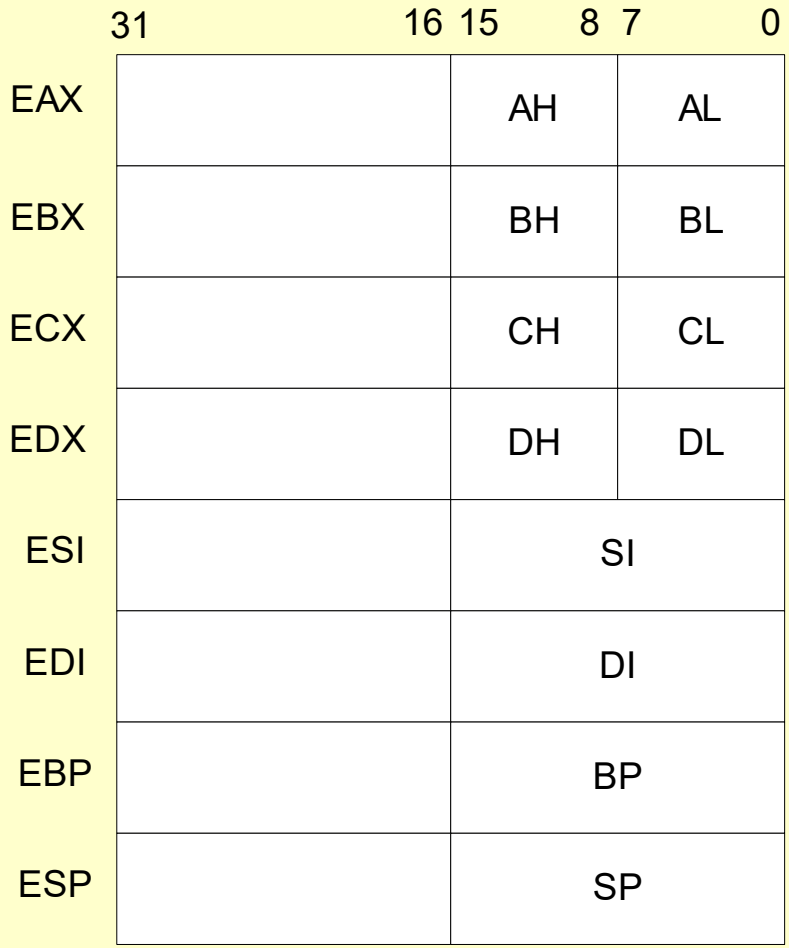
Instruction Pointer



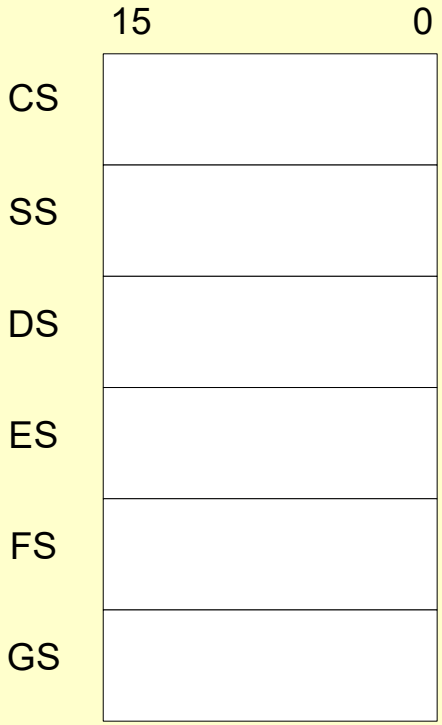
EFLAG Register

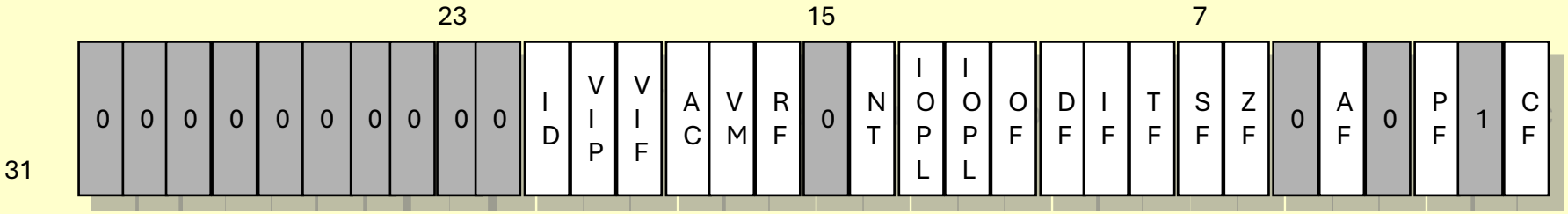


General-Purpose Registers



Segment Registers





ID	X	ID Flag (CPUID support)
VIP	X	Virtual Interrupt Pending
VIF	X	Virtual Interrupt Flag
AC	X	Alignment Check
VM	X	Virtual 8086 Mode
RF	X	Resume Flag
NT	X	Nested Task
IOPL	X	I/O Privilege Level
OF	S	Overflow Flag
DF	C	Direction Flag
IF	X	Interrupt Enable Flag
TF	X	Trap Flag
SF	S	Sign Flag
ZF	S	Zero Flag
AF	S	Auxiliary Carry Flag
PF	S	Parity Flag
CF	S	Carry Flag

S = Status Flag

C = Control Flag

X = System Flag

Bit Positions shown as “0” or “1” are Intel reserved.

EFLAGS

80286 and up:

- **IOPL (I/O privilege level)** *: Acesta conține nivelul de privilegii la care trebuie să ruleze codul dvs. pentru a executa orice instrucțiuni legate de I/O. 00 este cel mai înalt.*
- **NT (Nested Task)** *: Se setează atunci când o sarcină de sistem a invocat o altă sarcină prin intermediul unei instrucțiuni CALL în modul protejat.*

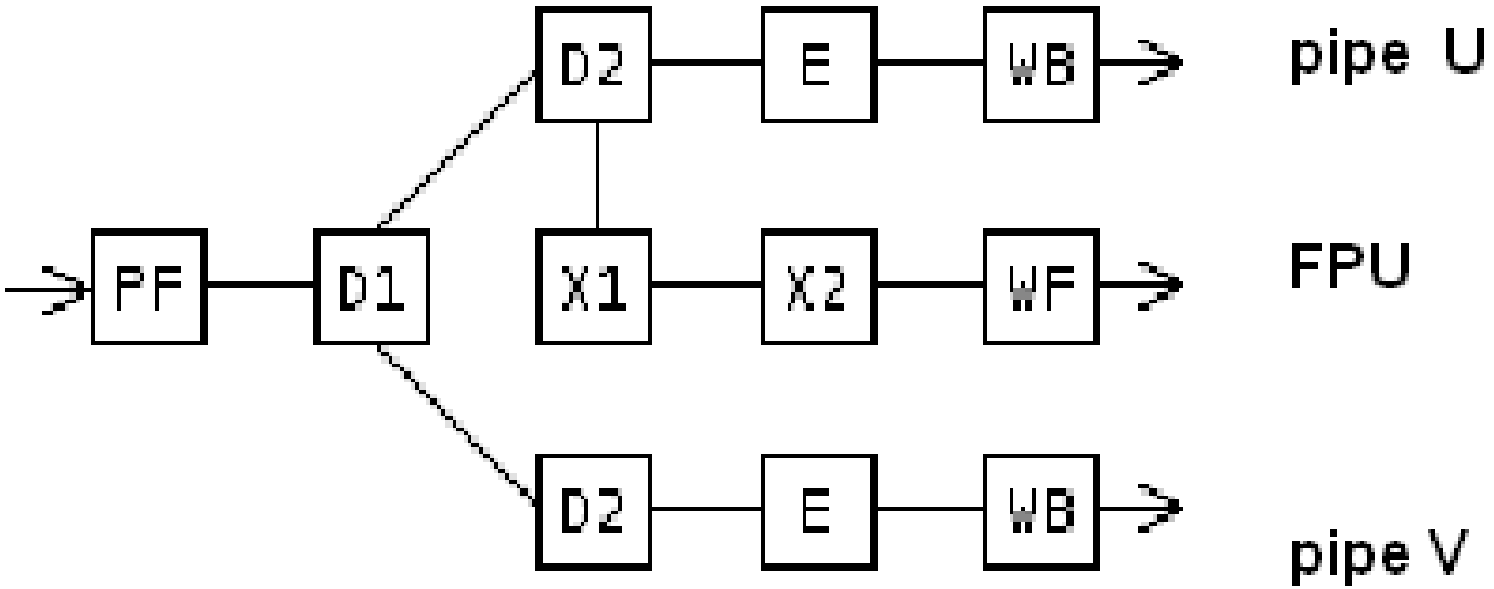
80386 and up:

- **RF (Resume)** *: Utilizat la depanare pentru a masca selectiv unele excepții.*
- **VM (Virtual Mode)** *: Atunci când VM=0, procesorul poate funcționa în modul Protejat, modul Emulare 286 sau în modul Real. Când VM=1, CPU este convertit la un 8086 de mare viteză. Acest bit are un impact enorm.*

80486SX and up:

- **AC (Alignment Check)** *: Instrucțiuni specializate pentru 80486SX.*
- *Pentium and up:*
- **VIF (Virtual Interrupt Flag)** *: Copie a interrupt flag bit.*
- **VIP (Virtual Interrupt Pending)** *: Oferă informații despre o întrerupere a modului virtual.*
- **ID (Identification)** *: Suportă instrucțiunea CPUID, care furnizează numărul de versiune și informații despre producător despre microprocesor și altele*

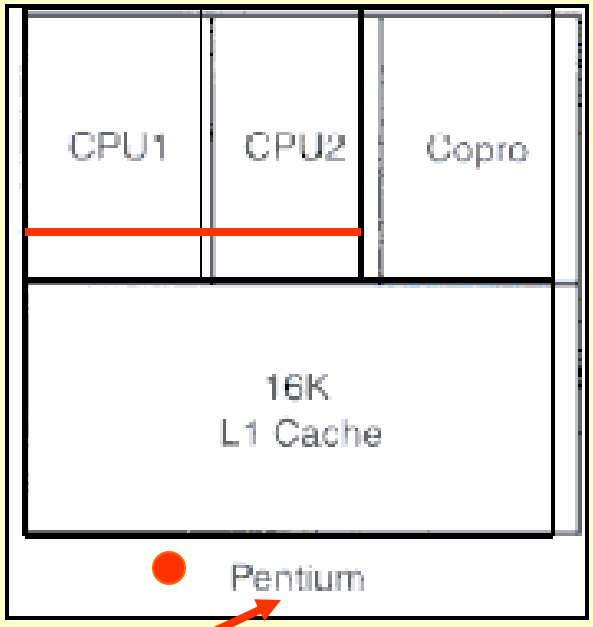
3. Unitatile de executie (ALU)

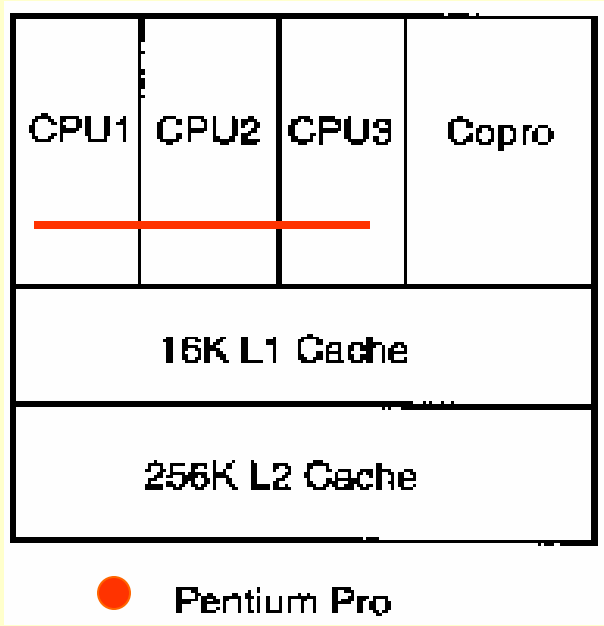


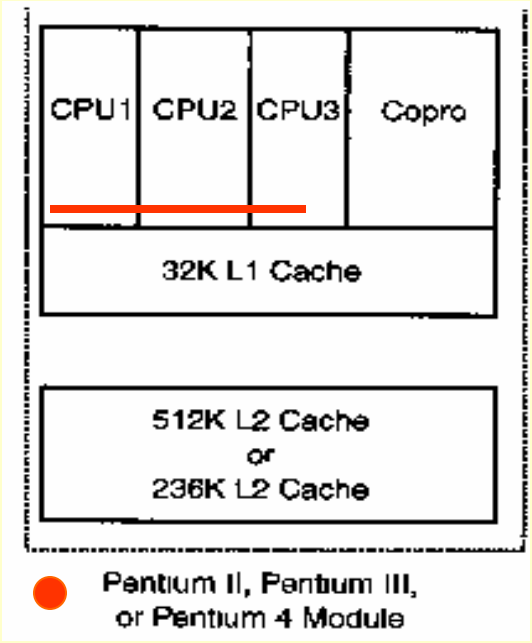
PF (Prefetch)
D1 (Decode 1)
D2 (Decode 2)
E (Execute)
WB (Write Buffer)

X1 - Ex. 1
X2 - Ex. 2
WF - Write float

- ***Pipeline-ul*** *este o tehnica de a descompune un proces secvential in sub-operatii, fiecare executandu-se intr-o unitate special dedicata, care opereaza concurent cu celelalte unitati*

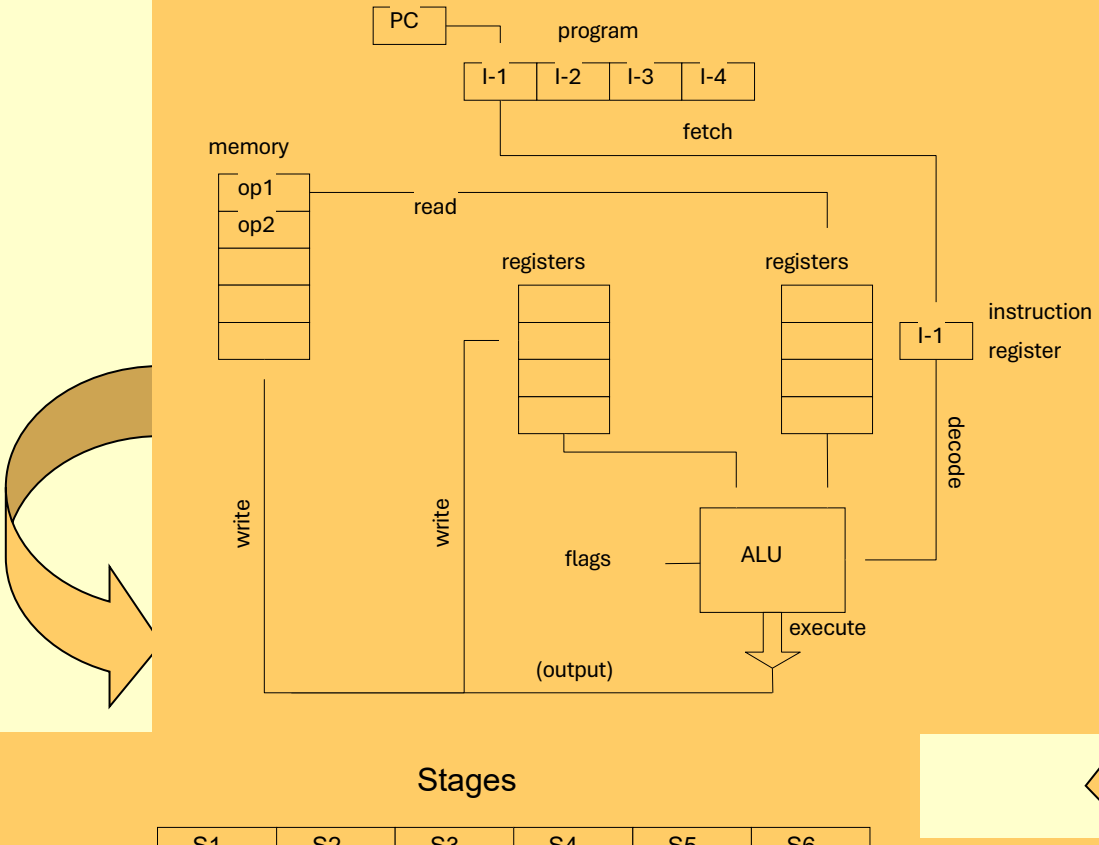






- Doua pipeline-uri intregi independente (U,V) si un pipeline flotant
- Executia unor comenzi scurte, prin instructiuni cablate
- Compatibilitate binara pentru instrucţiuni complexe i386, printr-o unitate CISC microprogramata

- Arhitecturile secventiala si pipeline



Stages

Cycles

	S1	S2	S3	S4	S5	S6
1	I-1					
2		I-1				
3			I-1			
4				I-1		
5					I-1	
6						I-1
7	I-2					
8		I-2				
9			I-2			
10				I-2		
11					I-2	
12						I-2

Stages

Cycles

	S1	S2	S3	S4	S5	S6
1	I-1					
2	I-2	I-1				
3		I-2	I-1			
4			I-2	I-1		
5				I-2	I-1	
6					I-2	I-1
7						I-2

Architectura Superscalara

Problema

Stages

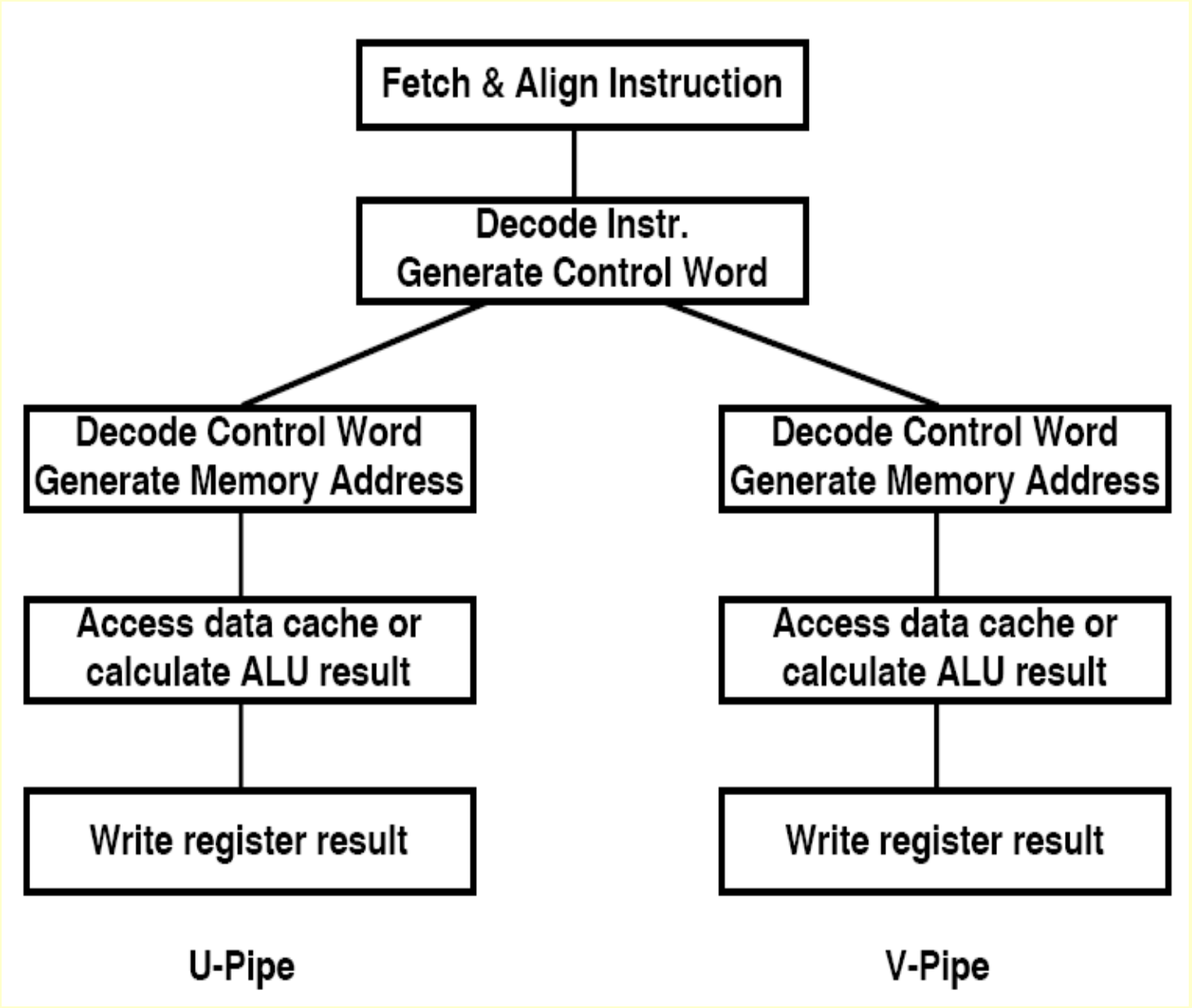
Cycles	S1	S2	S3	S4 (exe)	S5	S6
1	I-1					
2	I-2	I-1				
3	I-3	I-2	I-1			
4		I-3	I-2	I-1		
5			I-3	I-1		
6				I-2	I-1	
7				I-2		I-1
8				I-3	I-2	
9				I-3		I-2
10					I-3	
11						I-3

Solutie

Stages

Cycles	S1	S2	S3	S4: u	S4: v	S5	S6
1	I-1						
2	I-2	I-1					
3	I-3	I-2	I-1				
4	I-4	I-3	I-2	I-1			
5		I-4	I-3	I-1	I-2		
6			I-4	I-3	I-2	I-1	
7				I-3	I-4	I-2	I-1
8					I-4	I-3	I-2
9						I-4	I-3
10							I-4

Pentium Pipeline



PFetch

• Muta 32 bytes de instructiuni in coada de cod (program)
• Nu e necesar tot timpul
  – ~ 5 instructiuni extrase deodata
  – Utila numai daca nu se face salt

D1

• Determina lungimea totala a instructiunii
– Semnaleaza aliniatorului cozii de program (cod) unde incepe o noua instructiune
• Poate necesita 2 cicli:
– Cand sunt operanzi multipli care trebuie decodati
– ~ 6% in aplicatiile DOS tipice

D2

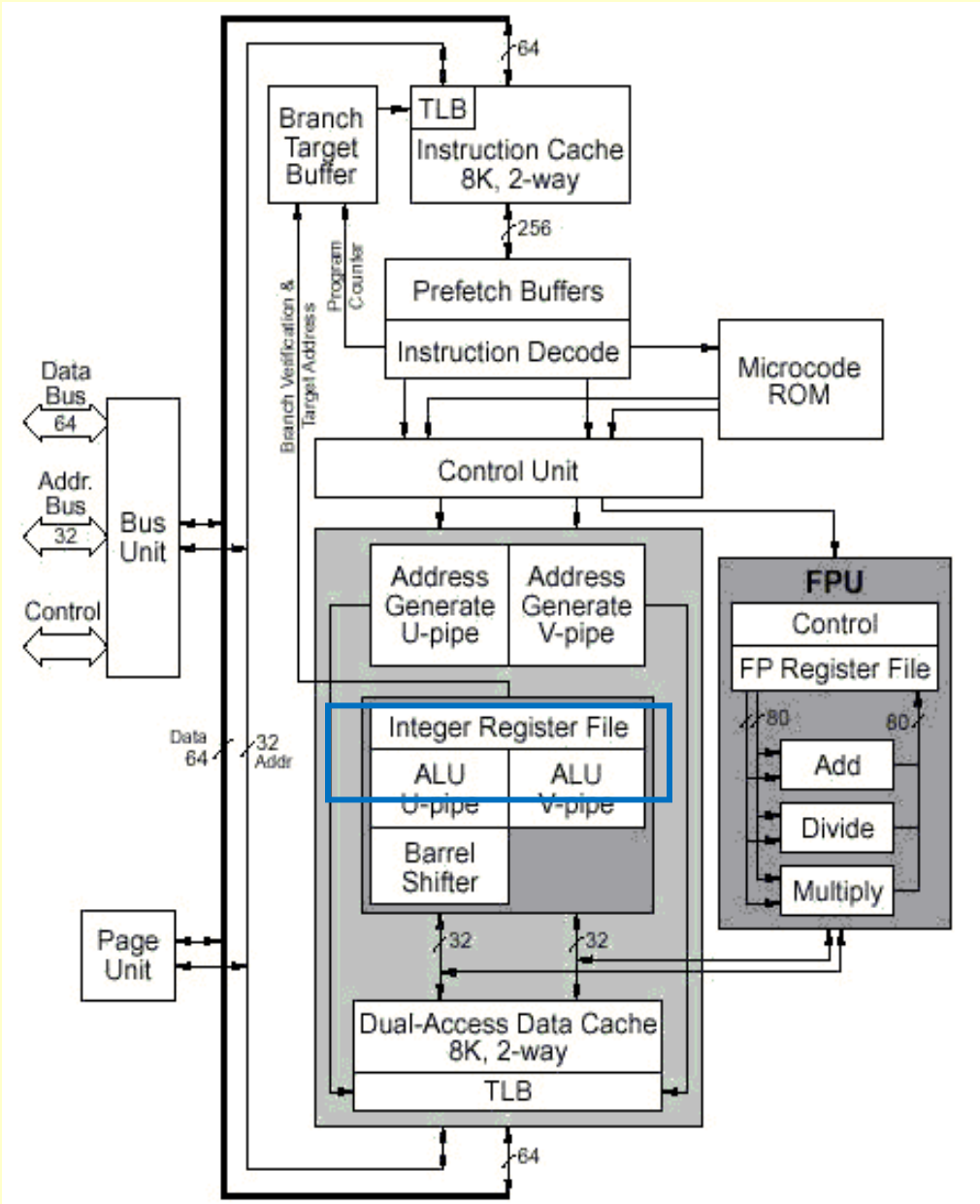
• Extrage offset-ul de memorie si operanzii imediati
• Calculeaza adresele de memorie – reg. de baza + reg. index..
• Poate necesita 2 cicli
– Daca reg. index este implicat sau ambele: adresa & operand imediat
– Aprox. 5% din instructiunile executate

EX

• Citeste operanzii din registru
• ALU efectueaza operatiile
• R/W memoria (data cache

WB

• Actualizeaza registrul rezultat/stare



Rank	80x86 instruction	Integer average (% total executed)
1	load	22%
2	conditional branch	20%
3	compare	16%
4	store	12%
5	add	8%
6	and	6%
7	sub	5%
8	move register-register	4%
9	call	1%
10	return	1%
Total		96%

The top 10 instructions for the 80x86.

- Executia simultana/ secventiala a 2 instr. este decisa in faza D1

Conditiile de executie simultana a 2 instr.:

- instr. simple (cablate) – nu necesita microcod (90%)
- nu exista dependente de date
  - I1 nu este JMP
  - Destinatia I1 nu e sursa la I2
  - Destinatia lui I1 nu este si destinatia lui I2
- instr. nu au deplasamente si operanzi imediati simultan
- instructiunile cu prefixe sunt executate in “pipeline u”

Dependente de date

- Daca 2 instructiuni acceseaza acelasi registru sau locatie de memorie, pot fi dependente
  - Dependenta: write/read
    I1: mov **ax,** [bx+20] ; I2: add cx,**ax** ;ax este destinatie apoi sursa
  - Antidependenta : read/write
    I1: Add cx,**ax** ; I2: mov **ax**,[bx+20] ; ax este sursa apoi destinatie
  - Dependenta de iesire (output)
    I1: mul bx ; I2 : mov **ax**,cx ; ambele modifica ax (destinatie)

Instructiuni Simple-generic

1. mov reg, reg/mem/imm
2. mov mem, reg/imm
3. alu reg, reg/mem/imm
4. alu mem, reg/imm
5. inc reg/mem
6. dec reg/mem
7. push reg/mem
8. pop reg
9. lea reg,mem
10. jmp/call/jcc near
11. nop
12. test reg, reg/mem
13. test acc, imm

Instruction Issue Algorithm

- Decode the two consecutive instructions I1 and I2
- If the following are all true
  - I1 and I2 are simple instructions
  - I1 is not a jump instruction
  - Destination of I1 is not a source of I2
  - Destination of I1 is not a destination of I2
- Then issue I1 to u pipeline and I2 to v pipeline
- Else issue I1 to u pipeline

COMPARISON OF ENERGY USAGE

Processor	Year	Clock Rate	Pipeline Stages	Cores/ Chip	Power (Watts)
Intel 486	1989	25 MHz	5	1	5
Intel Pentium	1993	66 MHz	5	1	10
Pentium Pro	1997	200 MHz	10	1	29
Pentium 4 Willamette	2001	2000 MHz	22	1	75
Pentium 4 Prescott	2004	3600 MHz	31	1	103
Intel Core	2006	2930 MHz	14	2	75
Intel Core i5 Nehalem	2010	3300 MHz	14	1	87
Intel Core i5 Ivy Bridge	2012	3400 MHz	14	8	77

4. FPU – unitatea in virgula flotanta

- pipeline cu 8 nivele (primele 4 in pipeline-U)
- formatele acceptate 32/64/80 biti => IEEE 754/85
- operatii FPU implementate “look-up tables” (like RISC)
- rezultatele sunt in tabele, iar operanzii sunt indecsi
- Pipelined FPU de 2..10 X mai rapida decat 486 FPU.

REMEMBER!! FDIV bug! Free replacement…

962,306,957,033 / 11,010,046 = 87,402.6282027341 (correct answer)
962,306,957,033 / 11,010,046 = 87,399.5805831329 (flawed Pentium)

Pentium FPU (Floating Point Pipeline)

- FP Pipeline are 8 nivele, partajeaza primele 4 nivele cu integer pipeline U
- WB al pipeline U este primul nivel de executie al pipline-ului FP (X1)
- NU se pot imperechea instr. flotante (exceptie FXCH) cu instr. in pipeline V

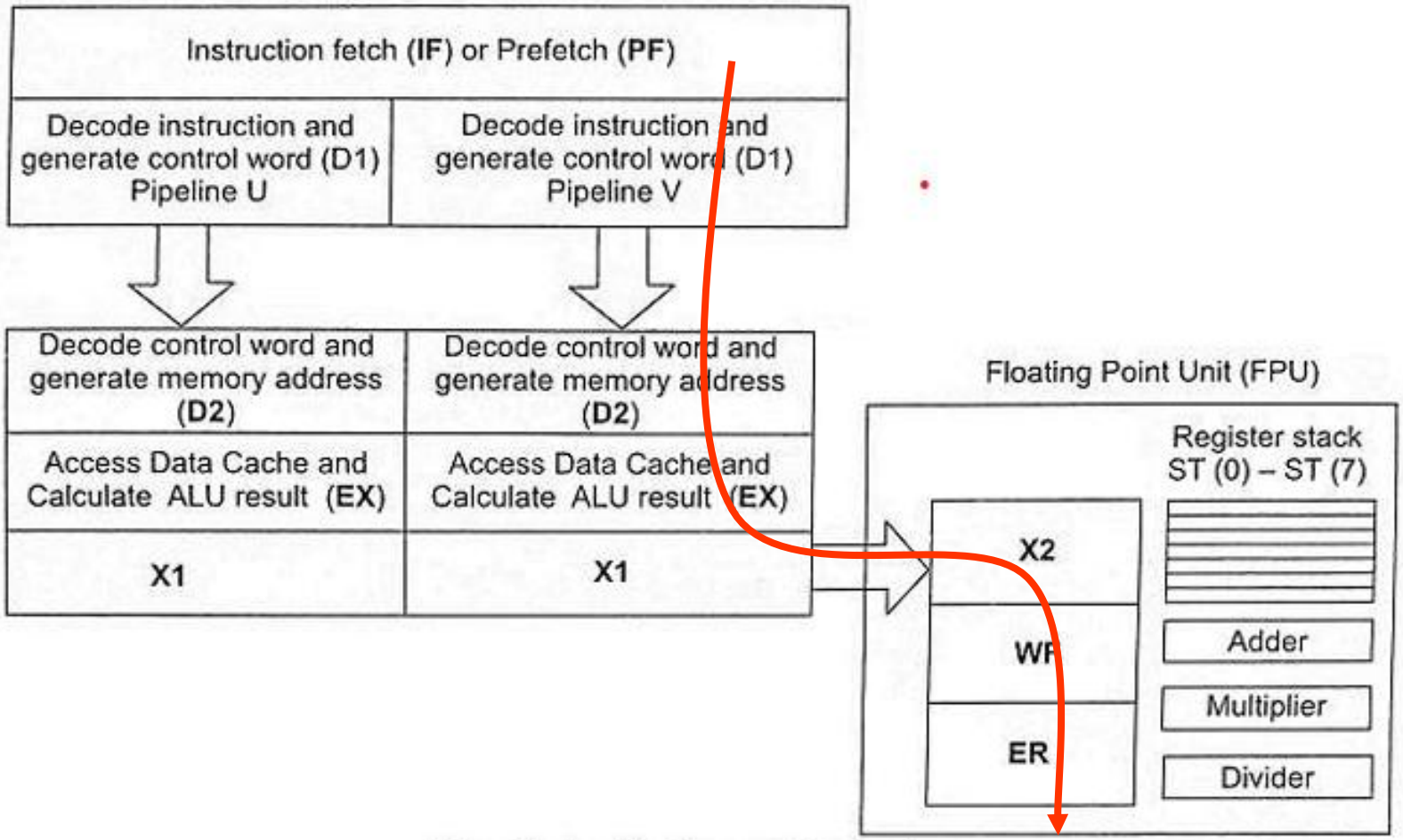


Fig. 12.4 *Floating-point pipeline*

5. BTB - Branch Target Buffer

- Stocheaza informatii despre salturile anterior executate, indexate de adresele instructiunilor, specifica adresa saltului si daca saltul se executa sau nu

- apare in faza D1 la instructiunile de salt conditionat /apel near

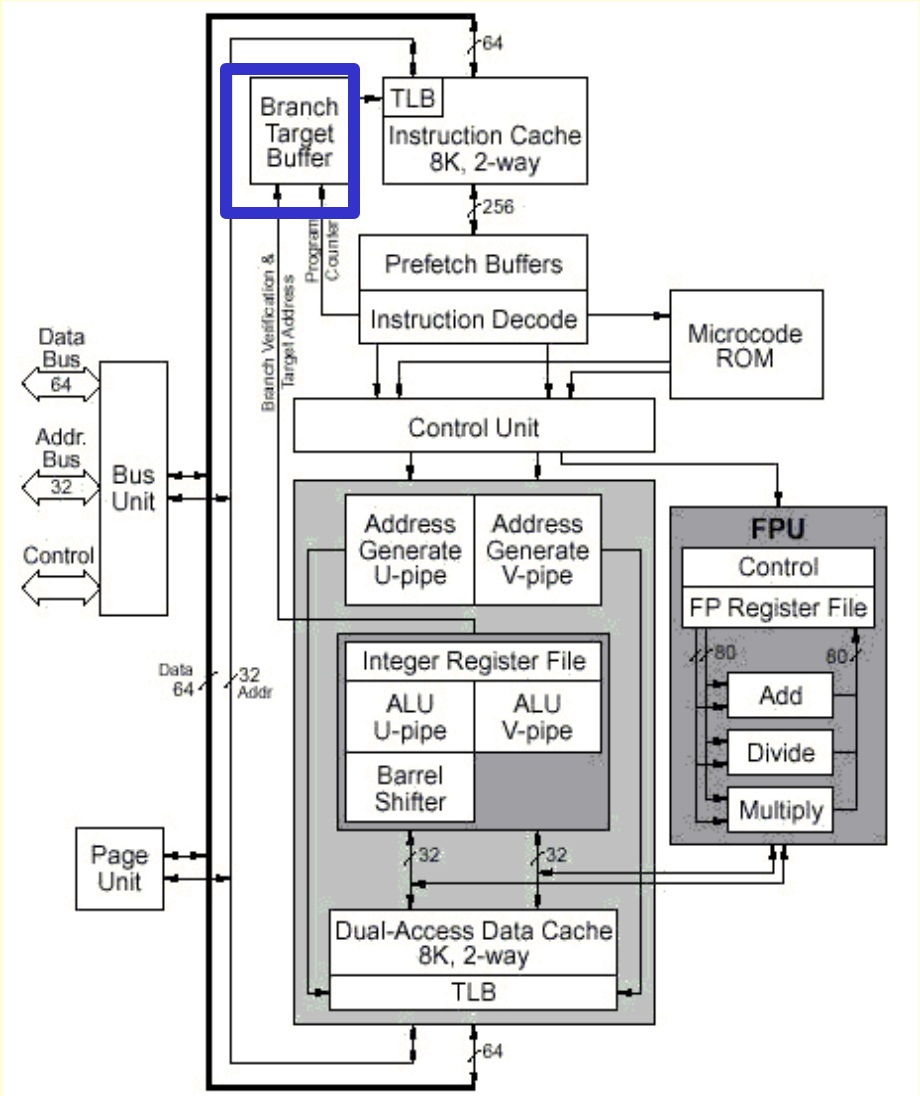
- se exploreaza tabela BTB (256 intrari)

- la fiecare salt nou µP stocheaza adr. ***Instr. de salt*** si ***adr. destinatiei saltului***

- daca adr. este in BTB se presup. ca saltul se executa la acea adr. fct. de bitul de istoric

- doar la executie se stie precis daca saltul se face sau nu

- daca se face, a fost anticipat corect => nu sunt intarzieri



https://en.wikipedia.org/wiki/Branch_predictor

```
.Program
.
$       Add ax,cx
$+2     Cmp ax, 0
$+4     Jc add1
$+6     Sub cx,2
.
.

add1:   xor ax,76h
.
.
```

if C=1

BTB

Extra prediction state bits

0	$+4	addr.1	
1	Jump address 2	Dest. Addr.2	
2	Jump address 3	Dest. Addr.3	
FF	Jump address FF	Dest. Addr. FF	

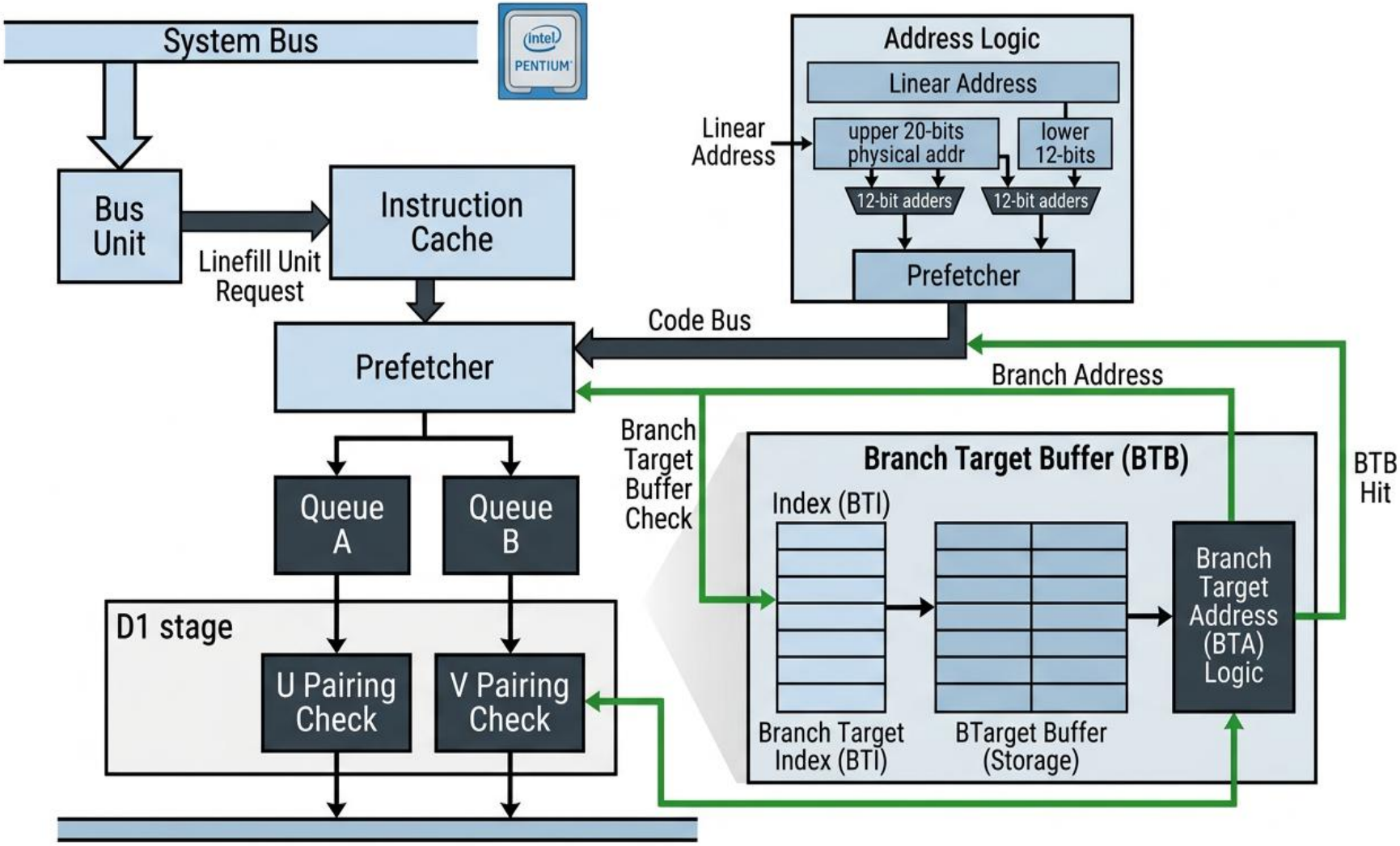


Diagrama de Flux a Branch Target Buffer

Diagrama ilustrează mecanismul de predicție dinamică a ramurilor în arhitectura Pentium (P5), folosind Branch Target Buffer (BTB) – un cache de 256 intrări (4-way set-associative) care accelerează execuția prin anticiparea țintelor de salturi, reducând penalitățile de pipeline flush.

Fluxul Principal de Execuție

Prefetcher: Generează adrese de instrucțiuni cu 2 parti -12-bit pentru adrese H/L (20-bit fiecare), pre-fetched din Instruction Cache (8KB) via Code Bus.

Queue A & B: Două cozi de instrucțiuni (12 intrări fiecare) stochează instr. pentru decodare dual (U/V pipes).

Pairing Check & Dual Decode: Verifică perechi compatibile pentru execuție simultană în cele două pipeline-uri; instrucțiunile de ramură sunt detectate devreme (D1 stage).

Rolul BTB în Predicție

Branch Address (BTB): Indexează BTB cu adresa saltului; stochează istoricul (taken/not taken) și ținta.

BTB & Branch Target Index (BTI): Furnizează adresa țintă prezisă dacă hit; predicția "taken" schimbă fetch-ul imediat, evitând stall-uri de 20+ cicluri.

BTB Check: Compară adresa curentă; la miss /mispredict, pipeline-ul se golește și se reia fetch-ul secvențial/correct.

Performanță și Avantaje

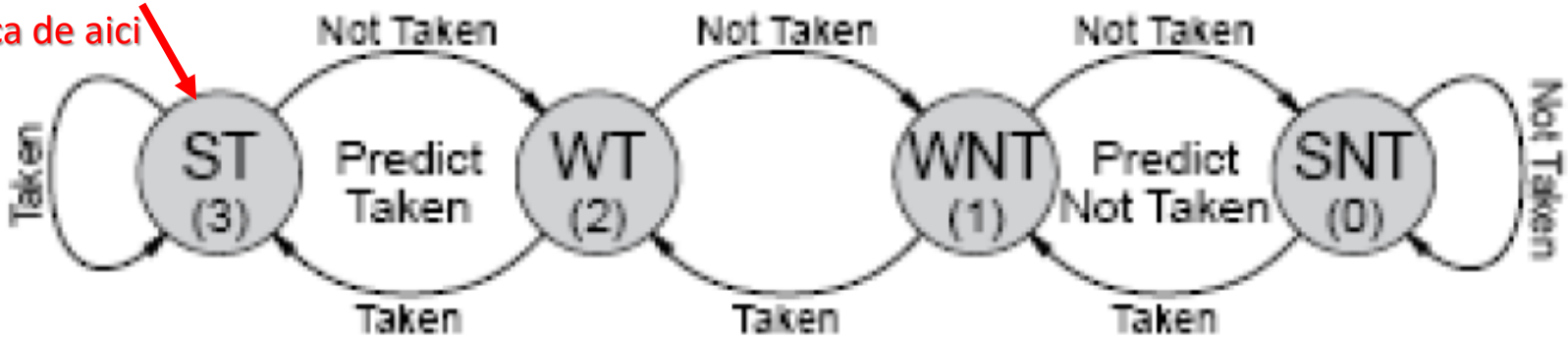
Precizie: > 90% în programe tipice, cu BTB actualizat la fiecare ramură executată (feedback de la EU).

Penalități: Mispredict flush ~ 2-5 cicluri pierdute; BTB reduce frecvența cu 50-70% față de predicție statică.

Capacitate: 256 intrări limitează bucle scurte/interuperi; țintele sunt păstrate cu adrese fizice via TLB

ALGORITMUL SMITH DE PREDICTIE DINAMICA A SALTURILOR

O noua instr. de salt pleaca de aici



La algoritmul Smith (cu 2 biti), istoricul bitilor implementeaza o masina de stare cu 4 stari posibile: (ST) strongly taken, (WT) weakly taken, (WNT) weakly not teaken si (SNT) strongly not taken. In starile ST si WT, viitoarele salturi sunt prezise ca se vor lua, iar la WNT si SNT salturile sunt prezise ca nu se vor lua.

MECANISMUL DE PREDICTIE DINAMICA AL SALTURILOR

History Bits	Resulting Description	Prediction Made	If branch is taken	If branch is not taken
11	Strongly Taken	Branch Taken	Remains Strongly Taken	Downgrades to Weakly Taken
10	Weakly Taken	Branch Taken	Upgrades to Strongly Taken	Downgrades to Weakly Not Taken
01	Weakly Not Taken	Branch Not Taken	Upgrades to Weakly Taken	Downgrades to Strongly Not Taken
00	Strongly Not Taken	Branch Not Taken	Upgrades to Weakly Not Taken	Remains Strongly Not Taken

Rata de predicție a BTB în Pentium

Branch Target Buffer (BTB) din Pentium (P5) atinge rate de predicție de **85-95%** în aplicații reale tipice (SPECint, programe sistem), datorită designului său cu 256 intrări și counter 2-bit saturating (strongly/weakly taken/not taken).

Factori care Influențează Precizia

- **Hit rate**: ~90% pentru salturi frecvente (bucle mici, funcții recurente); miss-uri duc la penalități de 10+ cicluri.
- **Pattern recognition**: Counter-ul de 2-bit se adaptează rapid
  – ex. bucle "taken" se stabilizează la "strongly taken" după 2 iterații.
- **Mispredict rate**: 5-15% în benchmark-uri, reducând IPC cu ~10% în cazuri extreme (salturi nepredictibile).

Comparație Performanță

Context	Rată Predicție	Penalizare/Mispredict
Bucle simple	98-100%	2 cicluri
Aplicații mixte	90-95%	5 cicluri medie
Workload-uri dificile	75-85%	10-20 cicluri

OPTIMIZAREA PREDICTIEI SALTURILOR

- se face prin eliminarea si reducerea numarului de salturi

De ce?

- Se elimina posibilitatile de salturi gresit prezise
- Se reduce numarul de intrari in BTB

Cum?

- Utilizarea de instructiuni care sa inlocuiasca salturile:
  - ✓ SETcc
  - ✓ CMOVcc or FCMOVcc

!! Combin JNE (JGE , etc.) cu MOV intr-o singura instructiune

OPTIMIZAREA PREDICTIEI SALTURILOR

Ex.1 If A>=B EBX = C2
else EBX=C1

Original instruction	Optimized instruction

cmp A,B ;A-B =>>flags
jge E0;If A≥B,S=O? jump
mov ebx,C1; EBX=C1
jmp E1
E0:
mov ebx,C2
E1:
	xor ebx, ebx ;EBX=0
	cmp A,B ;A-B =>>flags
	SETGE ebx ;If S=O,A>=B,EBX=1)
	dec ebx ; EBX=EBX-1
	and ebx,(C1-C2)
	add ebx, C2

Ex.2.If C=0,ECX=ECX+val
else C=1 ECX=EBX=0;C=carry

Original instruction	Optimized instruction

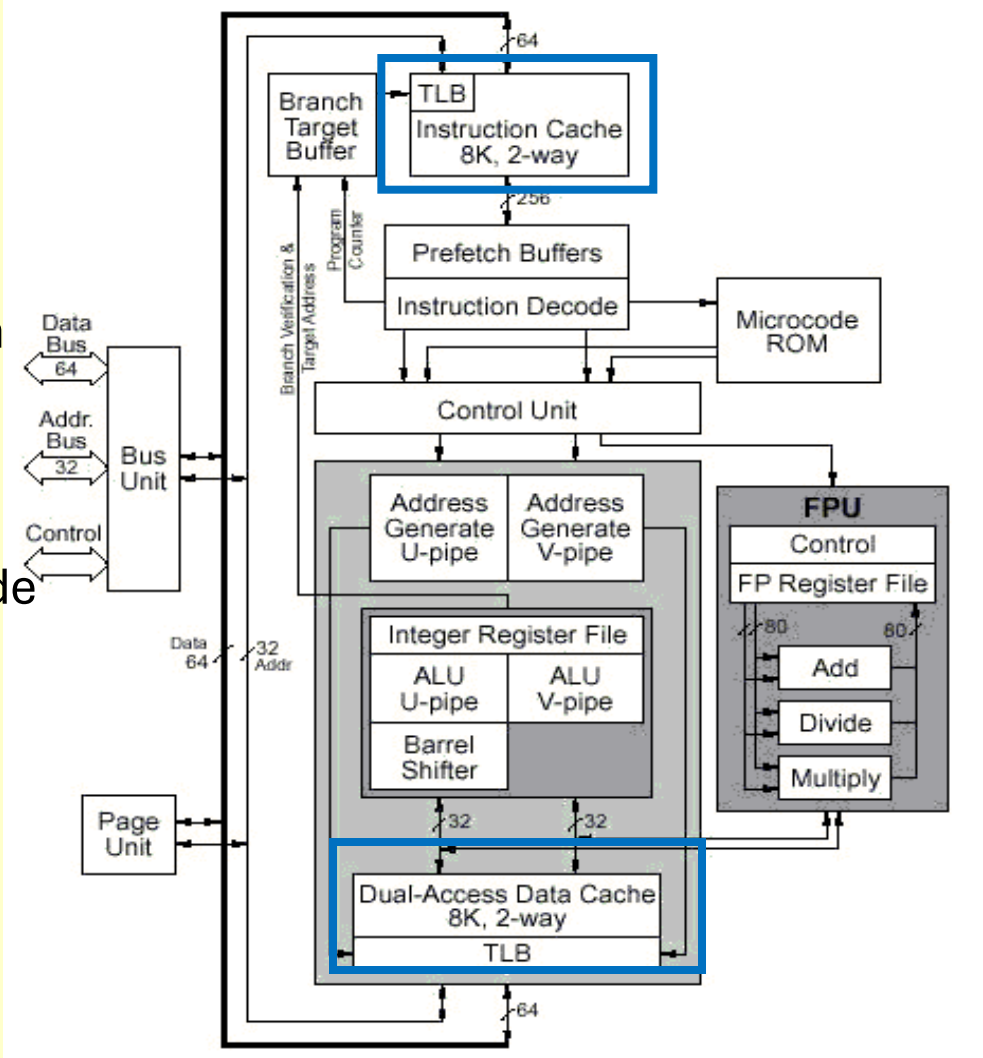
XOR EBX,EBX ; EBX=0
ADD ECX,[Val];ECX=ECX+val
JNC Continue; If C=0 jump
MOV ECX,EBX ;If C=1,ECX=EBX=0
Continue: ;ECX=ECX+val
	xor ebx,ebx ; EBX=0
	add ecx,[val] ; ECX=ECX+val
	CMOVC ecx,ebx ; If C=1, ECX=0

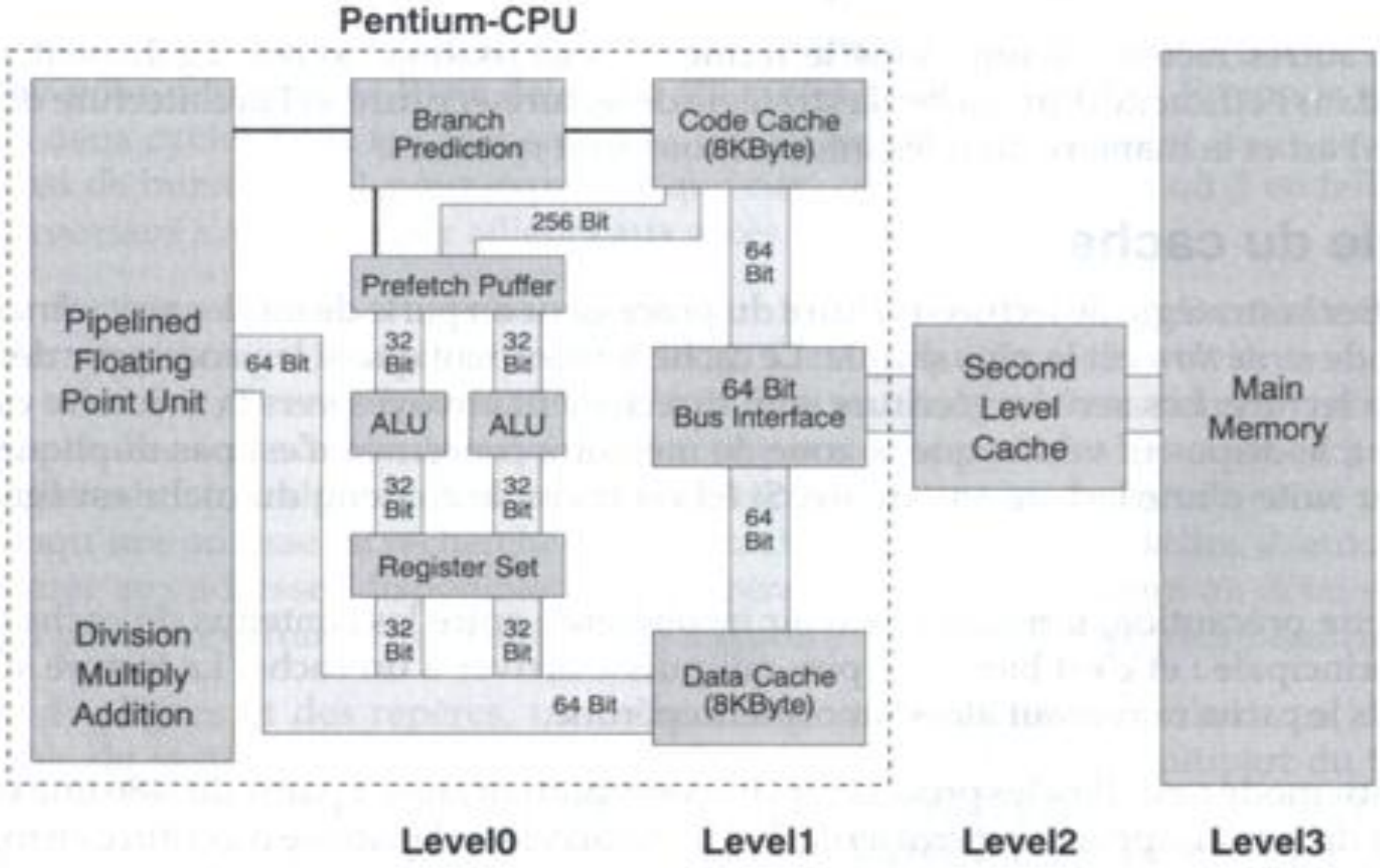
6. Memoria Cache 16Ko (8k code + 8k data)

• Sunt 2 **memorii cache separate :**
8kB – ptr. cod si **8kB** ptr. Date

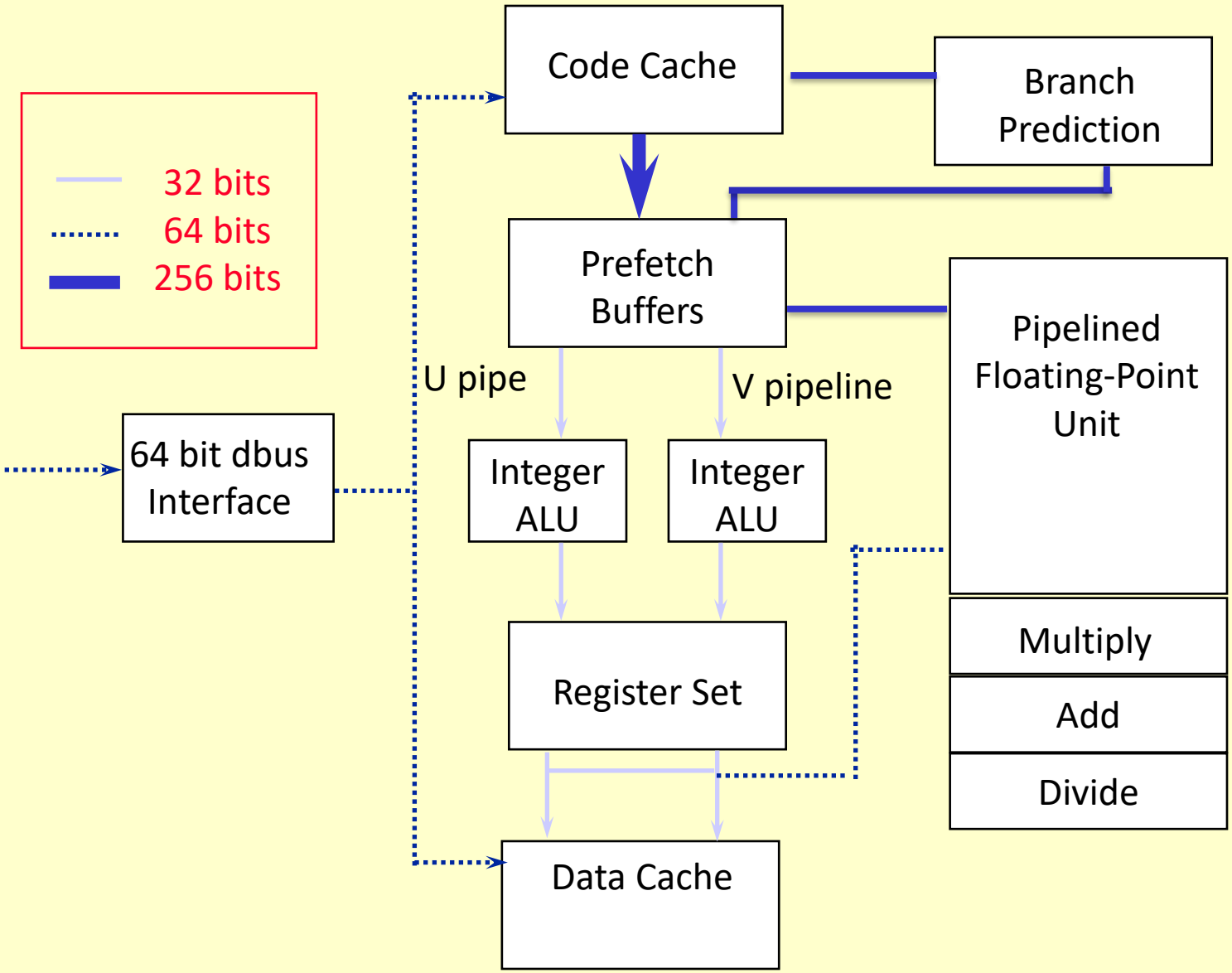
• Fiecare cache are translator de adresa, **TLB (Translation Lookaside Buffer),** separat, care translateaza adresa lineara la adresa fizica

• **Cache Code:**
– 2 cai, set-associative cache, 256 linii code cache si **prefetch buffer-ul** permite preextragerea a 32 bytes (256/8) de instructiuni




Pentium-CPU
Pipelined Floating Point Unit
Division Multiply Addition
Branch Prediction
Code Cache (8KByte)
256 Bit
Prefetch Puffer
32 Bit
32 Bit
ALU
ALU
32 Bit
32 Bit
Register Set
32 Bit
32 Bit
64 Bit
64 Bit
64 Bit
64 Bit Bus Interface
64 Bit
Data Cache (8KByte)
Second Level Cache
Main Memory
Level0
Level1
Level2
Level3

7. Magistrale/bus-uri



- Procesoarele Pentium au Data Bus de 64 biți si Adr. Bus de 32 biti (4GB)

- Pentium este un procesor pe 32 de biți, având registre de 32 de biti.

- Un ciclu standard de transfer poate citi sau scrie până la 64 de biți o data (8bytes)

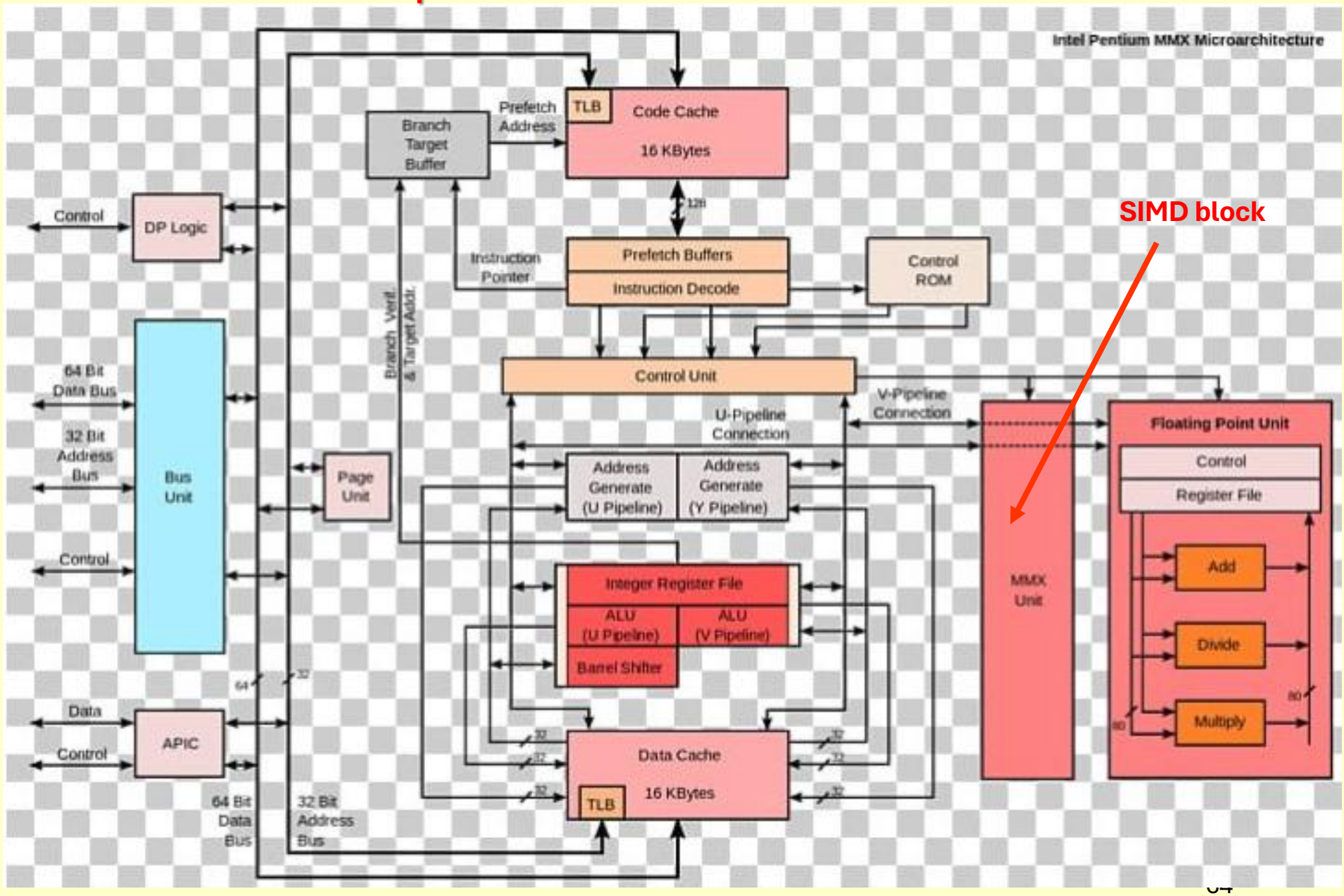
- Procesoarele Pentium suporta **cicluri Burst** de read/write

- Ciclurile in modul Burst sunt utilizate pentru operațiile cu memoria cache și se transferă cate 32 bytes în 4 tacti : (4 * 8 Bytes = 4 * 64 biți =256 biti/ 8Bytes)

- Dimensiunea liniei Cache la Pentium este de 32 Bytes

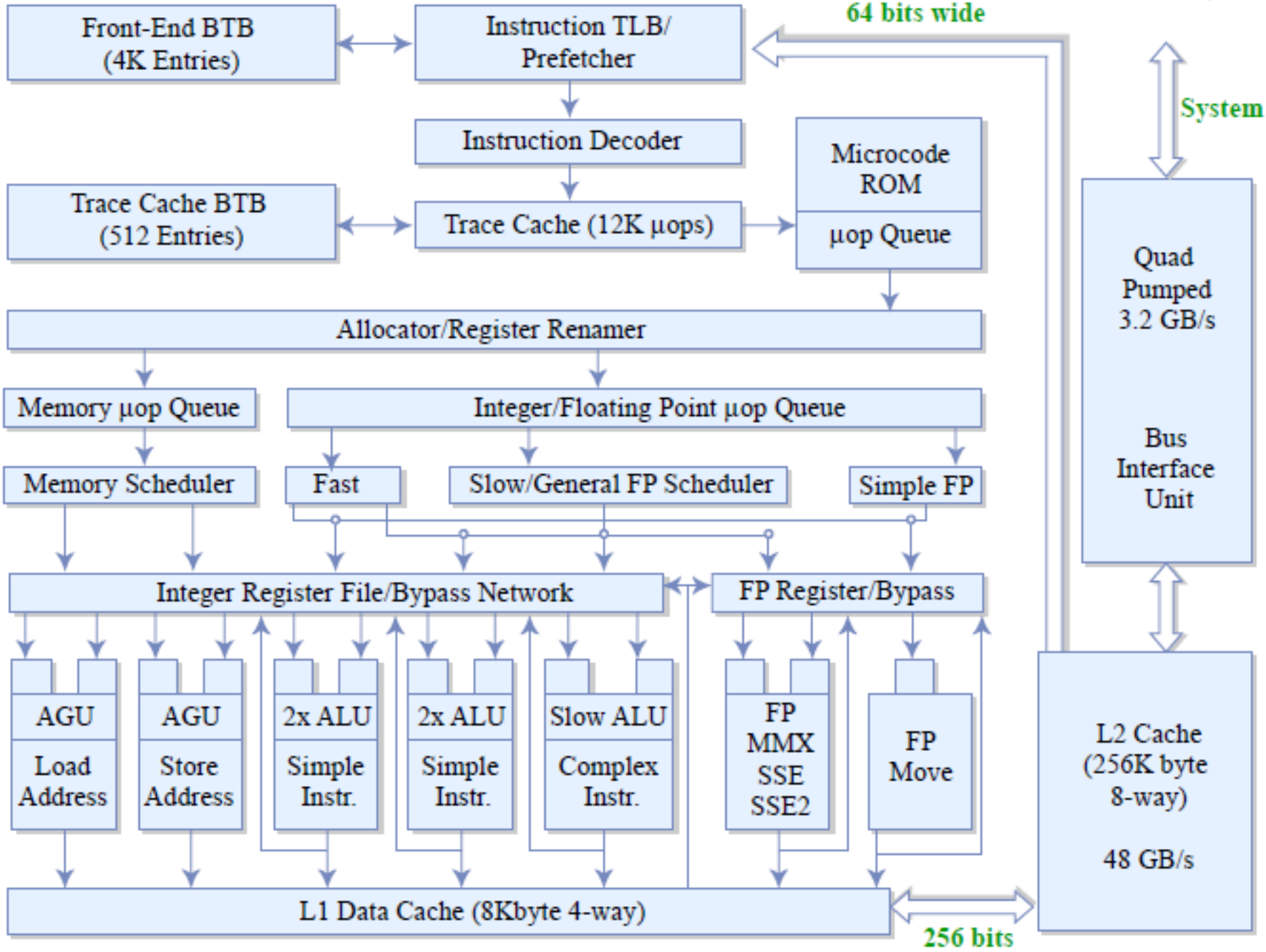
- La Pentium, toate operațiile cu memoria cache sunt cicluri burst (in avalansa).

Exemple de arhitecturi imbunatatite



Microarhitectura INTEL Pentium MMX

Pentium 4


Front-End BTB
(4K Entries)
Instruction TLB/
Prefetcher
64 bits wide
System Bus
Instruction Decoder
Microcode
ROM
Trace Cache BTB
(512 Entries)
Trace Cache (12K µops)
µop Queue
Quad
Pumped
3.2 GB/s
Allocator/Register Renamer
Memory µop Queue
Integer/Floating Point µop Queue
Bus
Interface
Unit
Memory Scheduler
Fast
Slow/General FP Scheduler
Simple FP
Integer Register File/Bypass Network
FP Register/Bypass
AGU
Load
Address
AGU
Store
Address
2x ALU
Simple
Instr.
2x ALU
Simple
Instr.
Slow ALU
Complex
Instr.
FP
MMX
SSE
SSE2
FP
Move
L2 Cache
(256K byte
8-way)
48 GB/s
L1 Data Cache (8Kbyte 4-way)
256 bits

- În timpul reumplerii cache-ului de instrucțiuni L1, µP traduce instrucțiunile complexe x86 în micro-operații de tip RISC (µops)

- de ex:

  „R: = R op Mem” se traduce în

  Load T, Mem ; Încărca din Mem în reg T

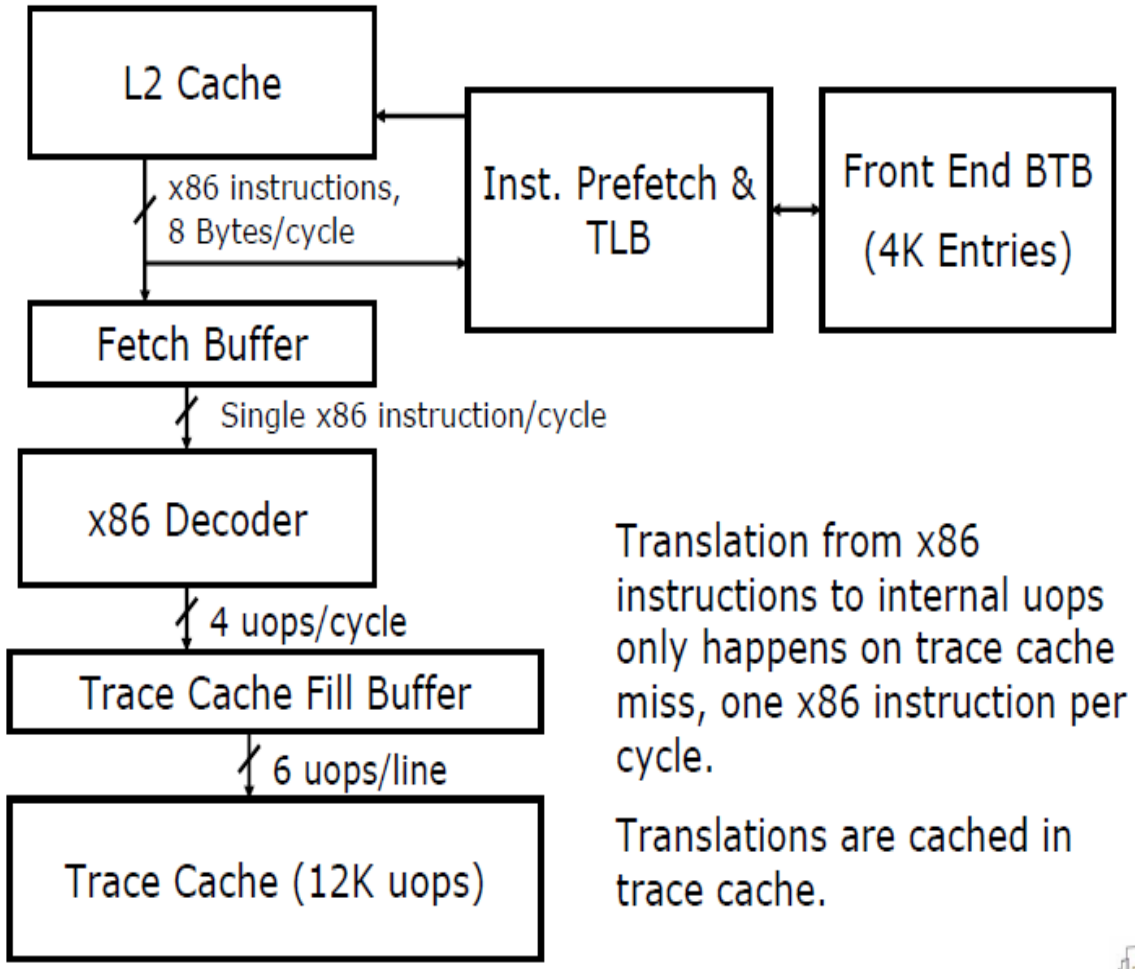
  R: = R op T ; Operație folosind valoarea din T

- Executa µops utilizând un motor superscalar, speculativ în afara ordinii OOO, cu redenumirea registrului

- Traducerea in µops a fost introdusă în arhitectura familiei Pentium Pro-P6

- Traducerea instrucțiunilor x86 in uop interne se întâmplă numai la cache trace miss, o instrucțiune x86/ciclu.

- Traducerile sunt memorate în memoria trace-cache.



Tema: Studiu

- **“Netburst”** - (Pentium 4, …) success not clear. Much longer pipeline, higher clock rate in same technology as P6, Trace Cache to capture micro-operations, avoid hardware translation…

- Multithreading - to increase performance for servers, parallel programs written to use threads…..

- Exemple de arhitecturi IA32 dezvoltate -anexe

- 10 quizz-uri cu 4 raspunsuri

http://arstechnica.com
https://books.google.ro/books/about/MICROPROCESSORS_PC_HARDWARE_AND_INTERFAC.html?id=t9ka7wmt_PQC&redir_esc=y

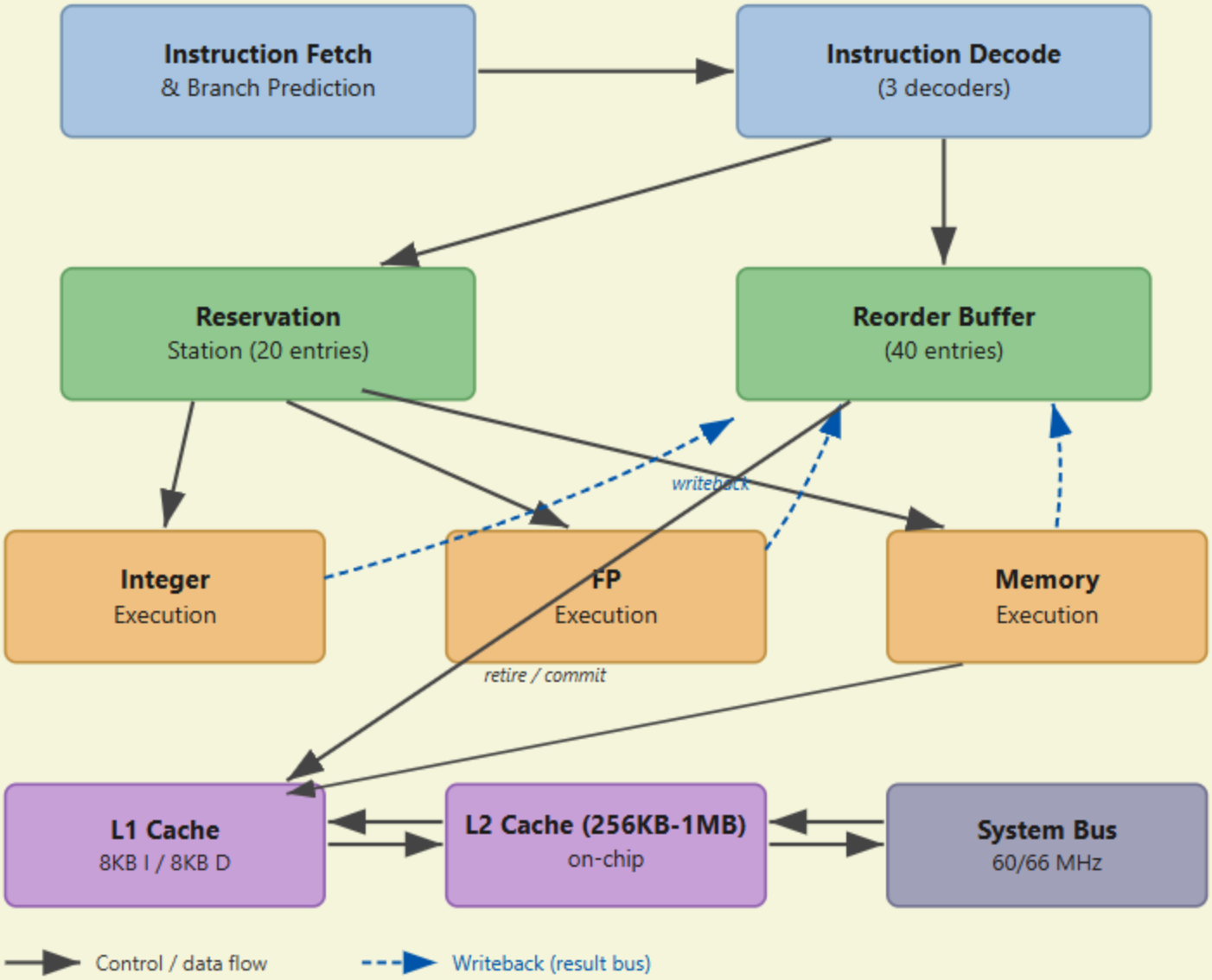
Pentium Pro - Microarhitectura Out-of-Order (1995)

- Primul procesor Intel cu motor de execuție complet out-of-order (OOO)
- Three-stage instruction procesing: Fetch și Decodificare, Issue to RS, Execute & Retire
- Buffer de Reordonare (ROB) reține 40 de micro-operații pentru execuție speculativă
- 3,3M-5,5M tranzistori, proces de 0,5 µm BiCMOS proces
- Până la 1MB cache L2 integrat în același pachet (off-chip, tensiune dual)
- Micro-arhitectura P6: baza tuturor designurilor Pentium / Core ulterioare
- Execuție dinamică: Predicție ramuri + Out-of-order + Execuție speculativă
- Suport configurații SMP (dual-procesor) prin magistrala APIC

• Decodificare instrucțiuni → Stație de Rezervare (Decodificatorul distribuie µ-ops la RS)
•Decodificare instrucțiuni → Buffer Reordonare (Decodificatorul alocă o intrare ROB per µ-op)
• RS → toate cele trei unități de execuție (RS emite µ-ops gata out-of-order)
• Unitățile de execuție transferă rezultatele la write-back-ul ROB (linii albastre punctate — magistrala comună de rezultate returnează rezultatele la ROB).
• ROB → Cache L1 (retire/commit) (ROB confirmă în ordine; stocările se retrag în cache)
• Săgeți bidirecționale între Cache L1↔L2 și L2↔Magistrala Sistem

Fișă tehnică Intel Pentium Pro, Intel Corporation, 1995

Pentium Pro Microarchitecture



Pentium II - Klamath și Deschutes (1997-1999)

- Combină motorul OOO Pentium Pro cu extensiile multimedia MMX
- Ambalaj tip cartuș SEC (Single Edge Contact) - interfață Slot 1
- Cache-ul L2 rulează la 1/2 din frecvența nucleului pe același substrat, off-die
- 57 instrucțiuni SIMD întregi MMX noi ce operează pe registre MM de 64 biți
- A introdus magistrala Front Side Bus de 100 MHz cu Deschutes (proces de 250nm)
- Frecvență maximă: 450 MHz; 7,5M tranzistori per die nucleu
- Variant mobil: Pentium II Mobile cu SKU-uri Low Voltage și Ultra-Low Voltage
- Super-scalar dual-issue pe 5 porturi de execuție (2 întreg + 1 FP + 2 load/store)

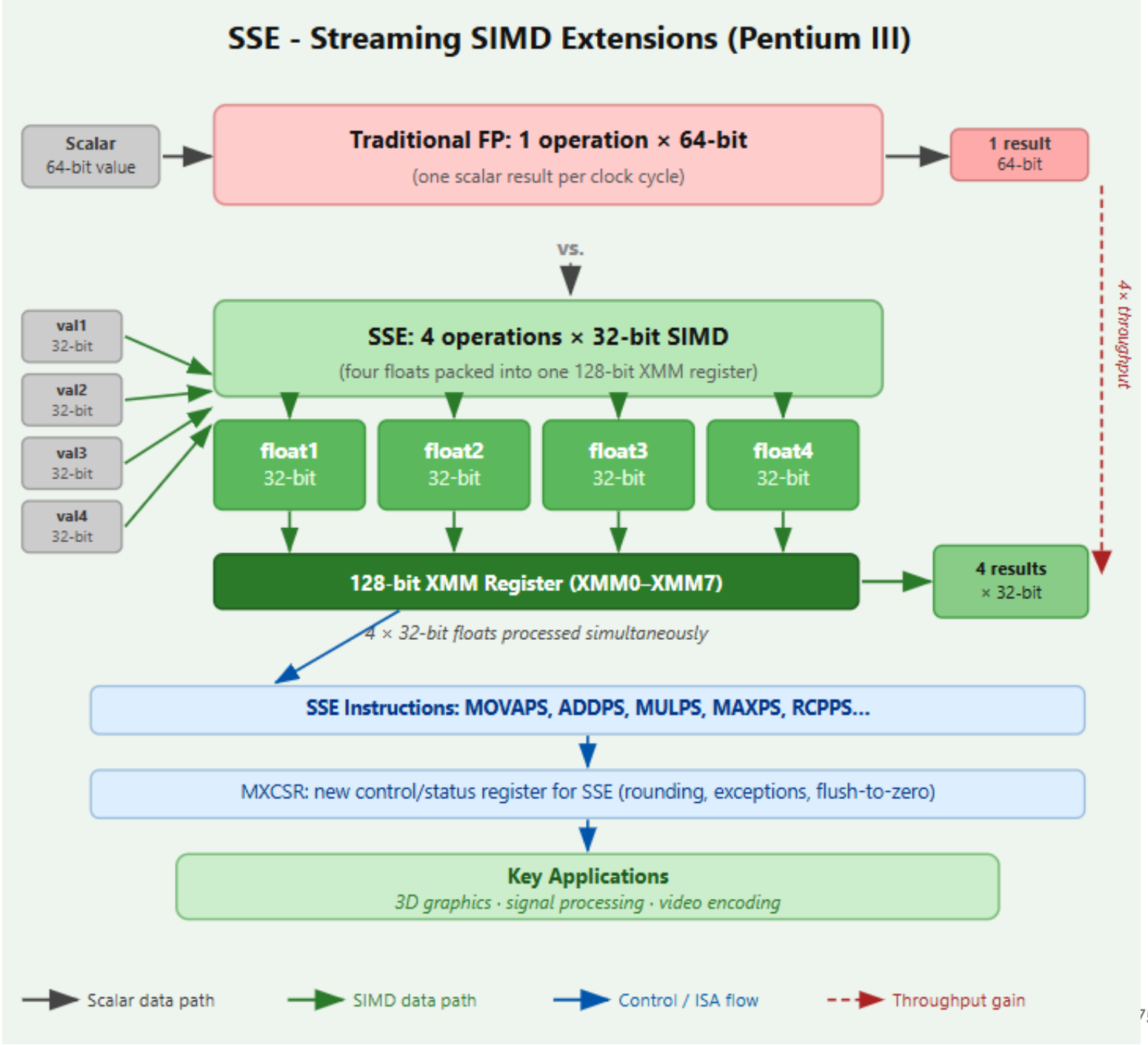
Fișă tehnică Procesor Intel Pentium II, Intel Corporation, 1998

Pentium III - Extensii SIMD Streaming (1999)

- SSE adaugă 70 de instrucțiuni noi pentru SIMD în virgulă mobilă cu precizie simplă
- 8 registre XMM noi de 128 biți, independente de starea x87 FPU și MMX
- Procesează 4 x 32 biți valori în virgulă mobilă cu precizie simplă per instrucțiune SIMD
- Debit maxim: 4 FLOP-uri per ciclu față de 1 FLOP în mod FP scalar
- FXSAVE/FXRSTOR: salvare rapidă la comutare de context OS a stării combinate FP/SSE

- Coppermine (180nm): 256KB cache L2 on-die la frecvența completă a nucleului
- Controversă: funcția Număr de Serie al Procesorului (PSN), dezactivată ulterior prin hardware
- Frecvență maximă: 1,4 GHz; primul CPU Intel cu 256KB L2 on-die la viteză complete

Fișă tehnică Procesor Intel Pentium III, Intel Corporation, 1999

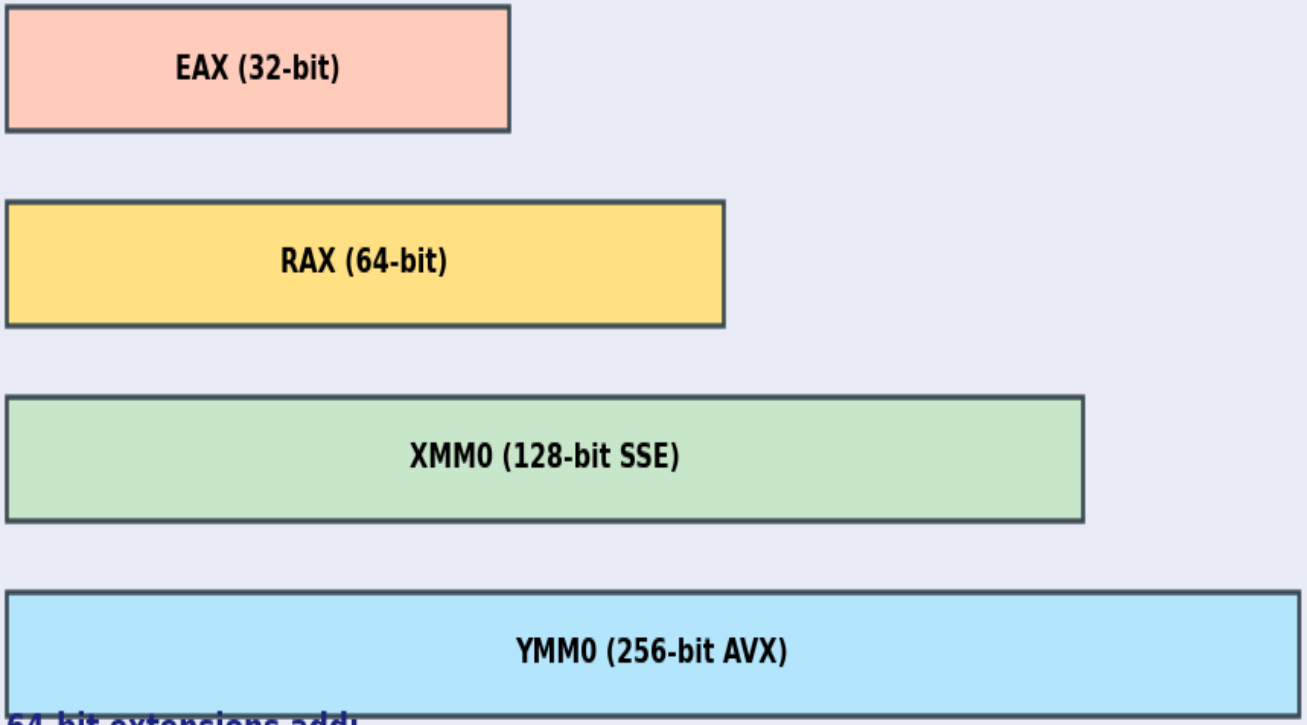

SSE - Streaming SIMD Extensions (Pentium III)
Scalar
64-bit value
Traditional FP: 1 operation × 64-bit
(one scalar result per clock cycle)
1 result
64-bit
vs.
4 × throughput
val1
32-bit
val2
32-bit
val3
32-bit
val4
32-bit
SSE: 4 operations × 32-bit SIMD
(four floats packed into one 128-bit XMM register)
float1
32-bit
float2
32-bit
float3
32-bit
float4
32-bit
128-bit XMM Register (XMM0–XMM7)
4 results
× 32-bit
4 × 32-bit floats processed simultaneously
SSE Instructions: MOVAPS, ADDPS, MULPS, MAXPS, RCPPS...
MXCSR: new control/status register for SSE (rounding, exceptions, flush-to-zero)
Key Applications
3D graphics · signal processing · video encoding
Scalar data path
SIMD data path
Control / ISA flow
Throughput gain

Intel 64 (EM64T) - Extensia x86 pe 64 biți

- Extinde IA-32 la adresare virtuală pe 64 biți cu compatibilitate retroactivă completă
- 16 registre de uz general pe 64 biți: RAX, RBX, RCX, RDX, RSI, RDI, RSP, RBP + R8-R15
- Spațiu de adrese virtuale pe 48 biți suportă până la 256 TB memorie virtuală
- Spațiu de adrese fizice pe 40 biți cu PAE suportă până la 1 TB RAM
- Adresare relativă RIP: datele sunt accesate relativ la indicatorul de instrucțiune
- 16 registre XMM disponibile în modul 64 biți (față de doar 8 în modul 32 biți)
- Bit de pagină NX (No-Execute) obligatoriu: împiedică paginile de date să execute cod
- Prima implementare Intel: Pentium 4 Prescott (2004) / Xeon Nocona

Manualul Dezvoltatorului de Software pentru Arhitecturi Intel 64 și IA-32, Volumul 1, Intel 2023

Intel 64 (EM64T) – x86-64 Extension



64-bit extensions add:

- 16 general-purpose 64-bit registers (RAX–R15)
- 48-bit virtual / 40-bit physical address space
- RIP-relative addressing mode
- 16 XMM registers (up from 8 in 32-bit mode)
- Mandatory NX (No-Execute) bit support

REFERINTE

- **Pentium™ Processor User's Manual**
  - Order Number 241428
- **Pentium™ Architecture & Programming Manual**
  - Order Number 241430
- **Pentium™ Processor System Architecture**
  - Mindshare (ISBN 1-881609-07-3)
- **The Indispensable Pentium™ Book**
  - Addison-Wesley (ISBN 0-201-87727-9)
- **"RISC vs. CISC Still Matters", by DeMone**
  http://www.realworldtech.com/page.cfm?ArticleID=RWT021300000000