



Arquitectures x86 i x87

Estudi teòric-històric de la família de processadors x86 i dels coprocessadors matemàtics x87 als anys 80 i 90.

Consultor: Francesc Guim Bernat

Autor: Esteban Bonachera López

15/06/2011

A María José,

Gracias, amada esposa, por tu paciencia sin límites y por el apoyo prestado durante todo el tiempo que he pasado elaborando este estudio.

También quiero agradecerlos a vosotros, mis niños, queridos Paula y Arnau, vuestra comprensión por el “abandono” al que os he sometido mientras buscaba información, redactaba y corregía este documento.

ÍNDIX

1. Introducció	1
2. Motivació	1
3. Descripció del Treball	2
3.1. Objectius i resultats	3
3.2. Anàlisi de riscos	3
4. Abast de la proposta	4
5. Organització del Treball	4
6. Pla de treball	5
6.1. Relació d'activitats	5
6.2. Fites principals	6
6.3. Calendari de treball.....	6
7. Valoració econòmica	8
8. La família x86	9
8.1. Introducció	9
8.2. L'Intel 8086/8088	27
8.2.1. L'Intel 8086: l'origen	27
8.2.2. El 8088 el naixement del PC	30
8.3. Processador 80286. L'aparició de l'IBM PC AT	31
8.4. L'Intel 80386: l'era dels 32 bits	34
8.4.1. L'Intel 80386 DX	34
8.4.2. La variant 80386 SX	37
8.5. L'Intel 80486	39
8.5.1. El 80486 DX: un autèntic gegant	39
8.5.2. La "variant" econòmica: 80486 SX	41
8.5.3. Els OverDrive	42

9. La família x87	43
9.1. Introducció	43
9.2. L'Intel 8087	47
9.3. El coprocessador Intel 80287	49
9.4. L'Intel 80387	51
9.4.1 El coprocessador 80387 DX	53
9.4.2. L'econòmic 80387 SX	54
9.4.3. Alternatives a Intel	55
9.4.3.1. L'IIT 3C87	55
9.4.3.2. Cyrix 83D87	57
9.4.3.3. Cyrix EMC87	59
9.4.3.4. Weitek Abacus 3167	60
9.5. L'Intel 80487 SX. La gran "estafa"?	62
9.6. Comparatives de rendiment	64
10. Conclusions	67
11. Desviacions en la planificació original i dificultats trobades	69
12. Annex A. La CMOS RAM	71
13. Annex B. Característiques tècniques	74
14. Annex C. El 80387: La visió del programador	78
15. Bibliografia	81
16. Referències	82
17. Glossari	84

1. Introducció

Aquest document conforma la proposta d'un Treball de recerca en l'àmbit de les Tecnologies de la Informació i la Comunicació. Bàsicament, es presentaran els diferents **microprocessadors** [1] i **coprocessadors matemàtics** [2] que varen participar en el naixement dels antecessors dels actuals ordinadors personals.

D'una banda, es començarà, tot seguint un ordre cronològic, amb la creació per part d'*Intel* dels microprocessadors 8086-88 i, com a conseqüència, l'aparició de l'*IBM PC XT*; fins a arribar al potent microprocessador *Intel* 80486. D'altra banda, també es descriuran els coprocessadors matemàtics corresponents als diferents processadors, és a dir, es començarà amb l' *Intel* 8087 i s'arribarà al 80487 *SX*, fent comparacions entre els coprocessadors compatibles oferts pels diferents fabricants, sense oblidar el "misteriós" *Weitek*.

Així mateix, s'inclouran diferents annexos amb material addicional i necessari que completarà aquest Treball de recerca.

Finalment s'afegirà la bibliografia i la cita de totes les fonts que es faran servir i, per acabar, un glossari amb tots els tecnicismes utilitzats.

2. Motivació

L'arribada de l'*IBM PC XT* a l'empresa i a l'entorn domèstic, va marcar un punt d'inflexió al món informàtic. És evident que es va produir un canvi social molt gran. Amb l'apropament de les computadores al gran públic, les vides personals, socials i laborals han evolucionat ràpidament.

Diverses tasques com la comptabilitat domèstica, la correspondència, gestions, estudis, hobbies o negocis poden realitzar-se en un temps record i sense sortir de casa.

Al llarg dels anys vuitanta i principis dels noranta, van sorgir diferents versions dels microprocessadors que formaren la **família x86** [3]. Anàlogament, el mateix va passar amb els coprocessadors matemàtics i la **família x87** [4].

Aquest treball realitza un estudi exhaustiu de tots els microprocessadors i coprocessadors matemàtics que formen el que s'anomena **Quarta generació de computadores** [5].

3. Descripció del Treball

Amb aquest treball el que es pretén és donar a conèixer, d'una forma amena i molt completa, els diferents processadors, i la seva evolució, que varen formar el que es coneix com *PC* (ordinador personal) i, per extensió, la Quarta generació de computadores.

Primerament, després d'una breu introducció, es parlarà de la família x86: s'introduirà el microprocessador 8086, fabricat per l'empresa nord-americana *Intel*. A continuació s'explicarà el micro 8088, base del que es coneix com *PC XT*, creació del gegant *IBM*.

Es continuarà amb la introducció del 80286, les variants del 80386 (tant la variant *DX* com la variant *SX*) i les variants del 80486 (tant la versió *SX* com la versió *DX*, *DX2*,...).

Per finalitzar aquesta primera part, s'introduiran els *OverDrive*, que són un híbrid entre els 80486 i els microprocessadors de la següent generació: el *Pentium*, destinats a l'aprofitament de tot el maquinari present amb un 80486 instal·lat, mitjançant un processador més ràpid i modern.

Seguidament s'introduirà la família x87, que correspon als diferents coprocessadors matemàtics que afegeixen extensions en *coma flotant* als diferents processadors de la família x86.

Així mateix, es farà una comparació entre els diferents coprocessadors compatibles, doncs cada fabricant oferia característiques i funcions diferents. També es veurà el coprocessador *Weitek*, que era un món apart i necessitava un connector especial a les plaques base compatibles.

Per finalitzar aquest apartat s'inclouran proves de rendiment dels diferents coprocessadors treballant amb el mateix processador.

Finalment, i ja per acabar, s'inclouran, a banda de les conclusions finals del Treball, diferents annexos amb documentació, articles i proves, la bibliografia consultada i un glossari.

3.1. Objectius i resultats del Treball

A continuació, s'enumeraran els principals objectius d'aquest treball:

- Conèixer l'**arquitectura x86** [6] i les seves característiques principals.
- Fer un estudi dels diferents microprocessadors i coprocessadors matemàtics que formen les famílies x86 i x87.
- Comprendre els avantatges que oferien uns microprocessadors envers uns altres; és a dir, la seva evolució.
- Entendre la diferència entre microprocessadors de 16 *bits* i 32 *bits*.
- Intentar aclarir si realment l'ús dels coprocessadors matemàtics augmentaven el rendiment dels computadors.
- Conèixer els diferents modes de treball dels microprocessadors que comparteixen l'arquitectura x86.
- Comprendre que no tan sols els *Megahertz* són sinònim de major prestacions en aquesta família de microprocessadors.

3.2. Anàlisi de riscos

El present Treball, tal i com succeeix a tots els Projectes, els recursos són limitats. S'ha de minimitzar el possible impacte que pugui produir una situació no desitjada de risc. A part de les garanties de qualitat de tot el procés, es prendran les següents mesures preventives:

- Realitzar un seguiment setmanal de l'estat del treball complint els terminis establerts a la planificació inicial.
- Assegurar la disponibilitat dels recursos necessaris: material bibliogràfic, disponibilitat de l'ordinador i aplicacions ofimàtiques necessàries per cercar informació i redactar el Treball, connexió a Internet...
- Prioritzar les tasques mantenint un ordre lògic i coherent. Per això s'haurà de respectar, amb molta cura, l'abast del treball (Veure l'apartat següent).

4. Abast de la proposta

El treball s'atendrà exclusivament a les dues arquitectures de processadors que s'han definit als apartats anteriors: la família x86 i la família x87. A més s'introduirà el coprocessador matemàtic *Weitek*, tot i que estava basat en una arquitectura diferent. Concretament els models que s'analitzaran són:

- **Família x86:** 8086, 8088, 80286, 80386 *DX*, 80386 *SX*, 80486 *DX*, 80486 *SX* i els *OverDrive*.
- **Família x87:** 8087, 80287, 80387 *DX*, 80387 *SX*, *Cyrix* 83D87, *Cyrix* EMC87, *IIT* 3C87, *Weitek* Abacus 3167 i 80487 *SX*.

A banda de l'anàlisi de les dues famílies de processadors anteriors, també s'inclouran uns annexos, necessaris, que tractaran diversos temes imprescindibles que complementaran la informació aportada.

5. Organització del Treball

Aquest apartat descriu amb detall els recursos que cal assignar per poder portar aquest Treball a bon termini:

Recursos Humans. Atès el caràcter divulgatiu d'aquest Treball, només ens caldrà assignar un rol a una mateixa persona: l'autor d'aquest Treball que, al mateix temps, és estudiant d'aquesta assignatura.

No podem parlar de Gerents, Arquitectes de Sistemes, Programadors,... El rol que necessitem és el de cercador d'informació, redactor i corrector del document final.

Recursos Temporals. Tenint en compte el caràcter docent d'aquest Treball, el nombre de crèdits universitaris assignats i, a més, la càrrega lectiva de l'autor; el còmput global d'hores invertides no hauria de superar en cap cas les 225 hores.

Recursos Materials. Els recursos materials necessaris són:

- **Documentals.** Accés a la bibliografia (llibres, revistes,...) i consulta de diversos recursos a Internet. Tot aquest material formarà part de la font bibliogràfica del Treball.
- **Tecnològics.** Les eines tecnològiques, avui dia imprescindibles en qualsevol ambient de treball tecnològic, seran les eines ofimàtiques necessàries per realitzar les consultes a Internet, el programari que es farà servir per compilar i redactar el Treball, un ordinador personal per emmagatzemar i tractar la informació, un iPad per consultar documents i textos i, finalment, una connexió de banda ampla. En concret, els recursos tecnològics que es faran servir són:
 - **Programari:** Microsoft Internet Explorer 9, Microsoft Office Enterprise 2007, Microsoft Project Professional 2003 i Adobe Reader 9.
 - **Maquinari:** Ordinador portàtil HP Pavilion dv7-2150es, Apple iPad i connexió ADSL de Telefònica 10Mb.

6. Pla de Treball

6.1. Relació d'activitats

A continuació s'enumeraran les diferents etapes i tasques que conformaran aquest Treball:

1. Cerca d'informació de cada una de les famílies de processadors i dels models que componen cada família.
2. Recopilació i preparació de la informació obtinguda, eliminació de dades repetides i traducció al català de tota la informació que estigui en un altre idioma.
3. Redacció de l'esborrany del Treball i cerca de tots els termes que s'hagin d'incloure al glossari.
4. Redacció de la Memòria, dels diferents annexos, el glossari i la bibliografia.
5. Creació de l'índex i la portada del Treball.

6. Correcció final de la Memòria: gramàtica, estil, composició del document, desviacions sobre els objectius plantejats en aquest document,...
7. Extracció de la informació més rellevant que s'inclourà a la presentació virtual del Treball.
8. Creació de la presentació virtual.

6.2. Fites principals

L'autor del Treball garantirà que les fites principals quedaran assolides en els termes i terminis d'aquesta proposta. A la següent taula es poden observar les fites més importants:

Fita	Descripció
18/04/2011	Data límit per lliurar la PAC2
19/05/2011	Data límit per lliurar la PAC3
06/06/2011	Data límit per lliurar la Memòria del Treball
06/06/2011	Data límit per lliurar la Presentació virtual del Treball

Taula 1. Fites principals

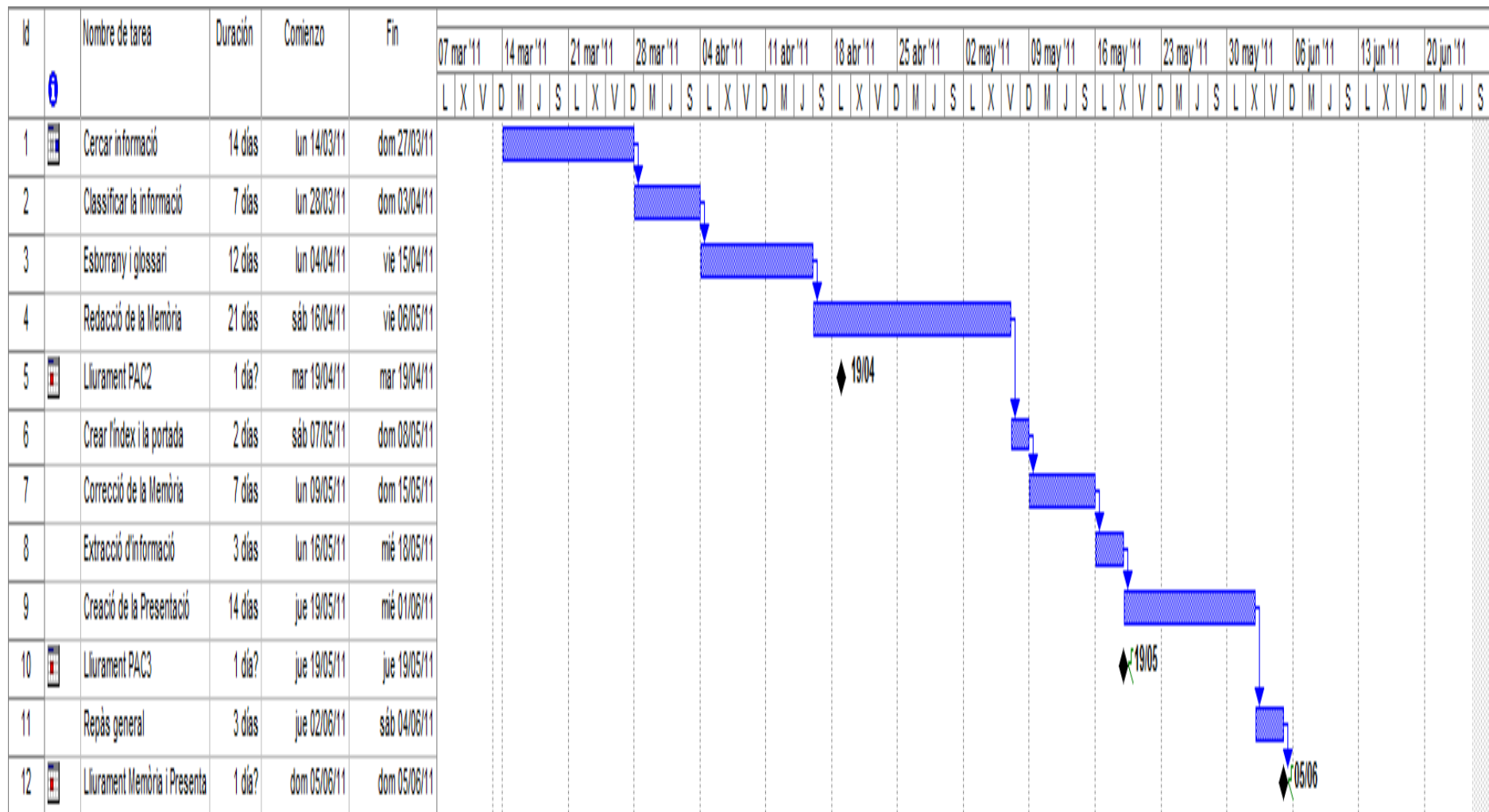
6.3 Calendari de treball

A continuació s'estableix el calendari laboral per a dur a terme el Treball. Aquest calendari estarà format per blocs setmanals, incloent els festius i caps de setmana. L'inici correspon al dilluns **14 de març de l'any 2011**, i l'acabament al dilluns **6 de juny del 2011**^{*}. Això comporta un total de **12 setmanes**. Tenim en compte que la dedicació al Treball depèn de la càrrega de la resta d'assignatures, es dedicaran un mínim de **12 hores** setmanals a treballar en el present Treball. Així doncs, com a mínim, es disposaran de **144 hores**. En qualsevol cas, malgrat s'hagi d'afrontar algun imprevist, s'haurà de respectar el límit màxim de **225 hores** (Veure apartat 5 d'aquest document).

A la pàgina següent s'adjunta un diagrama de Gantt on es mostren totes les activitats del Treball fins al seu tancament. Es contemplen també les fites principals.

* **Nota:** Al no tenir cap data especificada al calendari de l'aula, s'ha escollit el primer dilluns de juny com a data límit per al lliurament de la Memòria i la Presentació virtual del Treball.

Arquitectures x86 i x87



7. Valoració econòmica

Donat el caràcter teòric d'aquest Treball, és gairebé irrellevant l'assignació de costos als diferents recursos, tant humans com materials. Es podria considerar el consum energètic dels equips informàtics, però aquest maquinari estaria funcionant independentment de la realització d'aquest Treball.

S'ha de tenir present, no obstant, que l'autor assumeix els diferents rols que calen per dur a terme aquest estudi. Així doncs, l'autor s'encarrega de les següents tasques:

- Cerca, classificació i traducció de la informació.
- Redacció i correcció de l'estudi.
- Control i seguiment de la realització de les diferents tasques.

A continuació es poden veure els costos establerts per dur a bon termini aquest estudi. S'ha assignat un preu de 40 €/hora per a totes les tasques amb l'excepció de la *El·laboració* i la *Correcció* de l'estudi, doncs aquestes activitats són molt més laborioses. Així doncs el preu/hora és 10€ més alt per aquestes dues tasques.

Tasca	Preu/Hora	Hores dedicades	Total
Cerca d'informació	40€	42	1.680€
Classificació de la informació	40€	20	840€
Esborrany i glossari	40€	36	1.440€
El·laboració de l'estudi	50€	92	4.600€
Creació de l'índex i portada	40€	1	40€
Correcció de la Memòria	50€	9	450€
Selecció informació important	40€	2	80€
Creació de la presentació virtual	40€	15	600€
Repàs general	40€	8	320€
Total	--	225 hores	10.050€

Taula 2. Costos totals del Treball.

8. La família x86

En la societat actual, els sistemes electrònics ens envolten en la feina, el col·legi, la universitat i, fins i tot, a la nostra pròpia llar, facilitant-nos tasques de tot tipus que van des de el càlcul fins a l'entreteniment o la comunicació amb els nostres congèneres. Podem trobar-los adoptant formes molt diferents: un televisor, un telèfon mòbil, una vídeo consola, un microones i, per descomptat, en els ordinadors.

8.1. Introducció

Tots aquests dispositius tenen en comú una característica: el seu funcionament ho regeix un programa emmagatzemat en una memòria i executat per un microprocessador. Però, què és un microprocessador? És el circuit integrat central i més complex d'una computadora, i està constituït per milions de components electrònics integrats.

Constitueix la *CPU* (*Central Processing Unit*: unitat central de processament) d'un *PC* catalogat com **microcomputador** [7]. Des del punt de vista funcional és l'encarregat de realitzar tota operació aritmètica-lògica, de control i de comunicació amb la resta dels components integrats que conformen un *PC*. L'operació fonamental de la majoria de les *CPU*, és executar una seqüència d'instruccions emmagatzemades (programa).

El programa és representat per una sèrie de nombres que es mantenen en un tipus de memòria de computador. Hi ha quatre passos que gairebé totes les *CPU* d'**arquitectura de von Neumann** [8] utilitzen en la seva operació: **fetch**, **decode**, **execute** i **writeback** [9] (llegir, descodificar, executar i escriure).

El primer pas, *fetch*, implica recuperar una instrucció (que es representa per un nombre o una seqüència de nombres), de la memòria de programa. La localització en la memòria es determinada per un **comptador de programa** [10] (*CP*), que emmagatzema un nombre que identifica la posició actual al programa. En altres paraules, el *CP* indica a la *CPU* la posició de la instrucció al programa.

Després que es llegeix una instrucció, el *CP* és incrementat per la longitud de la paraula d'instrucció en termes d'unitats de memòria. La instrucció a ser llegida s'ha de recuperar d'una memòria relativament lenta, parant a la *CPU* mentre espera que la instrucció sigui retornada. Aquest problema és tractat en processadors moderns en gran part per les **memòries cau** [11] i les arquitectures **Pipelines** [12].

La instrucció que la *CPU* llegeix des de la memòria es usada per determinar què haurà de fer la *CPU*. En el pas de *descodificació* la instrucció es divideix en parts que tenen significat per a altres unitats de la *CPU*. La manera en què el valor de la instrucció numèrica és interpretat està definida per l'**ISA** [13] (*Instruction Set Architecture*, Arquitectura del conjunt d'instruccions) de la *CPU*.

Un grup de nombres en la instrucció, anomenats **opcode** [14], indiquen quina operació realitzar. Els altres nombres proporcionen informació requerida per aquesta instrucció, com per exemple, operands per una operació d'addició. Tals operands es poden donar com un valor constant (valor immediat), o com un lloc per localitzar un valor, que pot ser un *registre* o una adreça de memòria.

En dissenys més antics, les unitats de la *CPU* responsables de *descodificar* la instrucció eren dispositius de maquinari fixos. No obstant això, en *CPUs* i *ISAs* més abstractes i complicats, és freqüent l'ús d'un **microprograma** [15] per ajudar a traduir instruccions en diversos senyals de configuració per la *CPU*. Aquest *microprograma* pot ser modificat per canviar la forma de descodificar les instruccions.

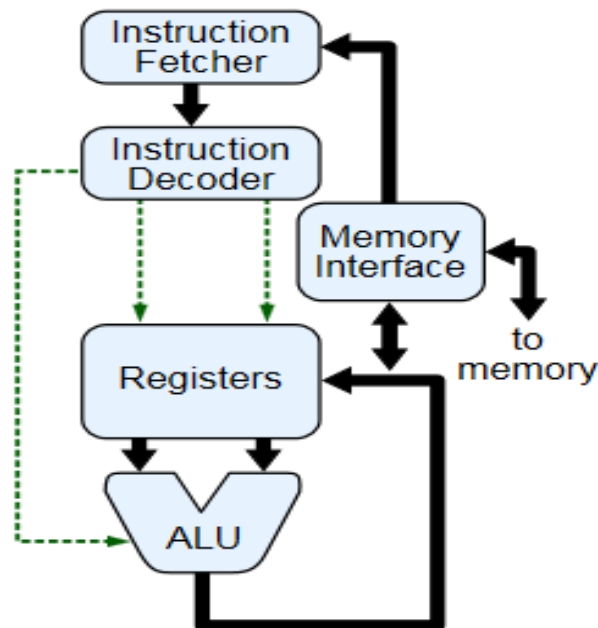


Figura 1. Diagrama de blocs d'una CPU.

El tercer pas, l'*execució* de la instrucció, consisteix en la connexió de diverses unitats de la CPU de manera que elles poden realitzar l'operació desitjada. Si, per exemple, una operació d'addició es sol·licitada, una unitat aritmètic lògica (ALU) serà connectada a un conjunt d'entrades i un conjunt de sortides. Les entrades proporcionen els nombres a ser sumats, i les sortides contindran la suma final.

El pas final, l'*escriptura*, simplement "escriu" els resultats del pas d'*execució* a la memòria o a algun *registre* intern de la CPU. Alguns tipus d'instruccions manipulen el *comptador de programa* en lloc de directament produir dades de resultat.

Després de l'*execució* de la instrucció i de l'*escriptura* de les dades resultants, el procés sencer es torna a repetir amb el següent *cicle d'instrucció*, normalment llegint la següent instrucció en la seqüència que està indicada pel valor del *comptador de programa*.

En el cas que la instrucció completada fos un salt, el *comptador de programa* serà modificat per contenir l'adreça de la instrucció a la qual es va saltar i l'execució del programa continua amb normalitat. En algunes *CPU* més complexes, es poden llegir varies instruccions, i poden ser *descodificades* i *executades* simultàniament.

Arquitectura d'un microprocessador

S'abordarà aquest estudi detallant l'arquitectura bàsica de qualsevol microprocessador "actual", entenent per arquitectura els detalls necessaris per poder programar el sistema utilitzant el llenguatge ensamblador específic del processador. A continuació es veurà en detall cada element de l'esquema simplificat d'un microprocessador de propòsit general.

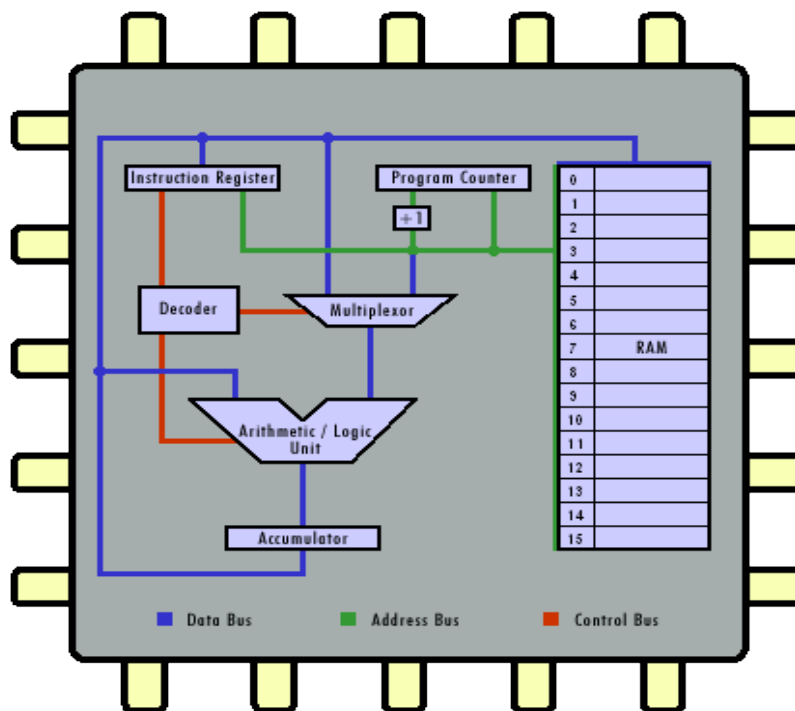


Figura 2. Esquema simplificat de l'arquitectura d'un microprocessador.

La unitat de control

Un microprocessador és un circuit integrat format per milions de components lògics que és necessari coordinar perquè cadascun realitzi el seu treball en el moment que s'espera, tasca que recau en la *CU* (*Control Unit*: unitat de control).

La *CU* s'encarrega d'emetre els senyals necessaris perquè la següent instrucció d'un programa, la localització del qual en memòria indica un registre específic al que sol anomenar-se *comptador de programa* o *punter d'instrucció*, sigui transferida fins a l'interior del microprocessador (fase de captació).

A continuació aquesta instrucció s'analitza i es preparen els operands que precisi (fase de *descodificació*) para a continuació executar-la (fase d'*execució*) i generar els resultats que corresponguin (fases d'*escriptura en memòria i registres*). La *CU* es comunica amb la resta d'elements del microprocessador mitjançant un *bus* intern, per exemple indicant quins *registres* són els operands en càlcul determinat.

La unitat aritmètic lògica

L'*ALU* (*Arithmetic Logic Unit*: Unitat aritmètic lògica) és la calculadora interna del microprocessador, realitza operacions aritmètiques i operacions lògiques. Els primers processadors no comptaven amb una *ALU* com a tal, sinó que disposaven d'una sèrie d'*acumuladors* cadascun dels quals tenia la capacitat de realitzar operacions, més o menys simples, entre el valor contingut de l'*acumulador* i un segon operant.

D'aquella arquitectura es va heretar el disseny que apareix en la [figura 2](#), en el qual l'*ALU* pres sempre com un dels seus operands el contingut de l'*acumulador* i deixa el resultat en aquest mateix *registre*. En l'operació també intervé el *registre d'estat* del microprocessador, però aquest no és pas un operand si no que és un agent extern que pot influir en l'operació que realitzi l'*ALU*.

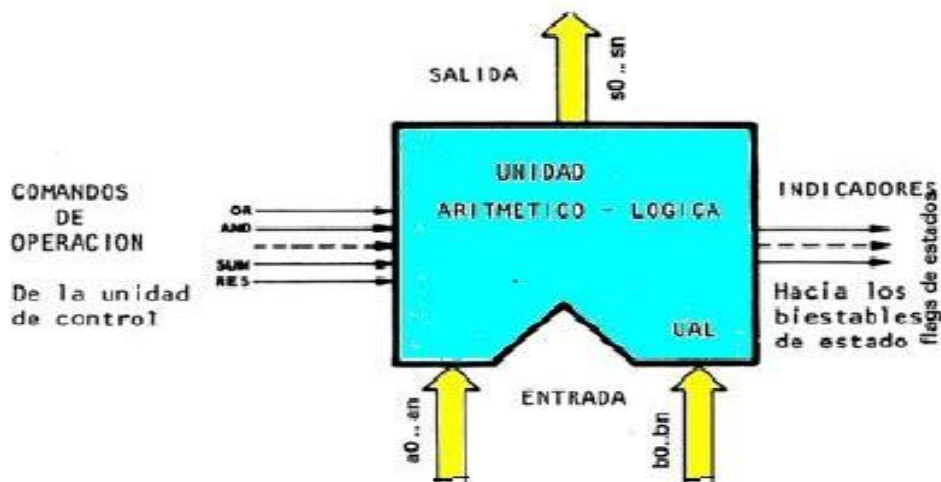


Figura 3. Esquema d'una ALU

Aquesta operació estarà dictada per la *CU*, que a través del *bus* intern també s'encarrega de controlar el funcionament de l'*ALU*. Com es pot observar a la [figura 3](#), l'*ALU* compta amb dues entrades i una sortida. Una de les entrades sempre és l'*acumulador* i, l'altra, procedeix d'un *registre* temporal. La sortida de l'*ALU*, torna a l'*acumulador*.

El banc de registres

Qualsevol informació sobre la qual vagi a actuar el microprocessador, per exemple per realitzar un càlcul, s'ha de transferir prèviament a l'interior de la *CPU*. Això és així perquè l'*ALU* no pot operar directament sobre el contingut de la memòria del sistema, físicament allotjada en circuits integrats independents i, per tant, externa al microprocessador.

En conseqüència aquest ha de comptar amb una zona d'emmagatzematge intern, accessible directament per l'*ALU* i la *Unitat de Control*. Aquesta petita àrea de memòria es divideix en porcions d'una certa grandària, anomenades **registres** [\[16\]](#), assignant-li a cada porció un nom concret. D'aquesta forma és possible referir-se a ells mitjançant un nom en lloc d'utilitzar una adreça.

Tradicionalment el banc de registres s'ha dividit en dos grups: els *registres de propòsit general* i els que tenen una *funció específica*. Al primer grup pertanyen aquells que el programador pot usar lliurement per emmagatzemar temporalment dades, mentre que els segons s'utilitzen de manera indirecta.

Els registres amb *funció específica* més usuals són:

- **Comptador de programa:** Conté l'adreça de la memòria on està allotjada la següent instrucció a executar. Actua, per tant, com un punter i, de fet, en alguns microprocessadors es denomina *punter d'instrucció*. És la *Unitat de Control* la que utilitza aquest *registre* per recuperar les instruccions del programa, incrementant el seu contingut a mesura que s'avança en l'execució o modificant-ho quan es troba una instrucció de salt.
- **Punter de pila** [\[17\]](#): En ocasions és necessari guardar temporalment el *comptador de programa*, per exemple al saltar a una subrutina o quan el microprocessador ha d'atendre una *interrupció* externa, amb la intenció de recuperar-ho posteriorment. Els primers microprocessadors comptaven amb una **pila** [\[18\]](#) interna, en el propi circuit integrat, que tenia una capacitat limitada i solia permetre cinc o vuit nivells com a màxim. Actualment la pila s'emmagatzema en memòria principal, externa a la *CPU*, de manera que aquesta l'única cosa que necessita és conèixer l'adreça on està el topall o part alta de la *pila*. Emmagatzemar aquesta adreça és l'objectiu del registre de *punter de pila*.

- **Acumulador** [19]: Pot ser usat com a *registre de propòsit general* en moltes situacions, però en unes altres adquireix el paper de *registre específic* en ser el destinatari de diferents operacions aritmètiques, lògiques o d'Entrada/Sortida.
- **Estat** (veure [Registre d'Estat](#)): La seva denominació canvia segons el tipus de disseny i fabricant, però la seva finalitat és sempre la mateixa: mantenir una sèrie de *bits* indicant l'estat en què es troba el microprocessador. Aquest estat prové normalment de l'execució de l'última instrucció, podent influir en com s'executaran les posteriors. També és possible que certs *bits* modifiquin la manera de funcionament del processador, de forma general o davant determinades instruccions.
- **Altres registres**: Si bé els quatre citats poden considerar-se els més importants, tots els microprocessadors disposen d'altres *registres* d'ús específic, ocults en la seva major part, que s'utilitzen per emmagatzemar el codi de la instrucció que s'està executant, contenir temporalment dades procedents de memòria que van a intervenir en un càlcul, etc.

Els registres de *propòsit general* es poden agrupar segons la seva grandària i el tipus d'informació que són capaços de mantenir. Al principi els microprocessadors comptaven amb *registres* de només 8 ó 16 *bits* pensats per operar amb aritmètica entera, però en l'actualitat la mida ha crescut fins els 32, 64 i inclús 80 *bits*, contemplant tant l'aritmètica entera com la de *punt flotant*.

Busos

Per construir un sistema informàtic, a més del microprocessador, es precisen altres elements de suport que facilitin l'emmagatzematge temporal de la informació i la comunicació amb l'exterior. En la [figura 4](#) s'ha representat esquemàticament el microprocessador i la forma en què s'estenen els seus busos a través del sistema, permetent la connexió amb una altra sèrie de circuits integrats.

El **bus de dades** [20] està format per una sèrie de línies, físicament surten del microprocessador en forma de pins que es connecten sobre un sòcol en una placa de circuit imprès, per les quals es transmeten en paral·lel un nombre determinat de *bits*, tants com a línies existeixin. Al nombre d'aquestes línies se li denomina *ample del bus* que, per regla general, coincideix amb la capacitat de l'*acumulador* i una part dels *registres de propòsit general*.

El *bus de dades* és bidireccional, de manera que permet tant enviar dades des del microprocessador cap a l'exterior com al revés.

Mitjançant el **bus d'adreces** [21] el microprocessador selecciona la posició de memòria en la qual va a escriure's o de la qual es vol llegir. També serveix per seleccionar dispositius d'E/S en cas que aquests tinguin associat un espai

d'entrada/sortida en la memòria. A diferència del *bus de dades*, el *bus d'adreces* és unidireccional.

El nombre de línies que ho formen determina l'ample del *bus* que, al seu torn, fixarà el nombre màxim d'adreces que és possible compondre i, en conseqüència, el límit de memòria al que es pot accedir de manera directa. Tant el *bus de dades* com el *bus d'adreces* tenen una estructura homogènia, en el sentit que les seves línies contribueixen per igual, amb un *bit*, a generar la dada o l'adreça que va a transferir-se ocupant el *bus* complet.

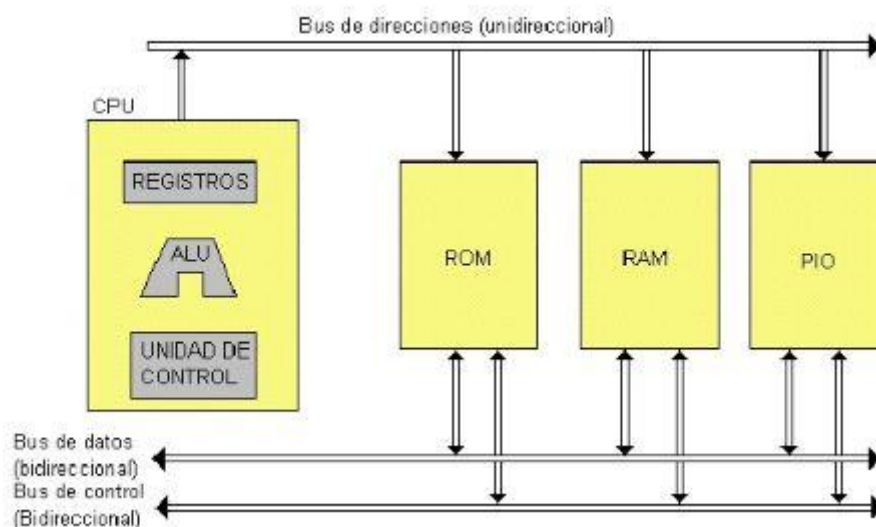


Figura 4. Representació d'un microprocessador i els seus busos.

El **bus de control** [22], per contra, és heterogeni i les línies que ho formen tenen cadascuna una fi diferent, per la qual cosa rarament s'utilitzen de manera simultània. Algunes d'aquestes línies són solament de Sortida, unes altres únicament d'Entrada i, en comptades ocasions, d'Entrada i Sortida.

A través de les línies del *bus de control* el microprocessador comunicarà al sistema si l'adreça que està col·locant en el *bus d'adreces* ha de ser enviada a la memòria o a un dispositiu d'E/S o si el que es vol és efectuar una lectura o una escriptura. Els dispositius externs empraran una línia d'aquest *bus* per comunicar al microprocessador que necessiten la seva atenció, provocant una **interrupció** [23].

Dissenys clàssics: CISC i RISC

Quan es parla d'arquitectura de microprocessadors sempre s'esmenten com a dissenys fonamentals dos casos concrets: **CISC** [24] (*Complex Instruction Set Computer*) i **RISC** [25] (*Reduced Instruction Set Computer*). A continuació, de manera superficial, s'expliquen les característiques i diferències entre aquests dos dissenys clàssics.

Els microprocessadors *CISC* van ser els primers a aparèixer, caracteritzant-se per tenir un conjunt d'instruccions que va guanyar en complexitat a mesura que l'escala d'integració ho permetia. Els programes escrits per a processadors *CISC* resulten compactes, ja que existeixen instruccions capaces d'efectuar operacions relativament complexes. Aquestes instruccions, no obstant això, triguen més a executar-se que altres més senzilles.

El disseny *RISC* va sorgir en la dècada dels 80 amb l'objectiu d'aconseguir microprocessadors capaços d'executar un major nombre d'instruccions per segon. Per això es va decidir que el microprocessador implementés un conjunt reduït d'instruccions, totes elles d'una longitud similar i fàcils d'executar.

Els programes per a microprocessadors *RISC* són més llargs, en haver de codificar com diverses instruccions les operacions complexes, però cada instrucció triga menys a executar-se. Durant uns anys els processadors *RISC* van oferir un rendiment molt superior als dissenys *CISC*, ja que l'augment en les freqüències de rellotge i les tècniques de "paral·lelització" afavorien en major mesura a l'arquitectura *RISC*.

A banda del conjunt d'instruccions, un altre aspecte que diferenciava clarament a un microprocessador *CISC* d'un *RISC* era el banc de registres. Els dissenys *CISC* comptaven amb un reduït nombre de *registres* que, en principi, tenien assignades funcions específiques.

Tant aquestes com els noms dels registres eren una herència de les computadores primigènies de les dècades dels 50 i 60. Pràcticament tots els microprocessadors *CISC* compten amb *registres* anomenats *A* (*Accumulator*), *B* (*Base address*), *C* (*Counter*) i *D* (*Data*), als quals posteriorment es van ser agregant uns altres addicionals.

Els microprocessadors *RISC*, per contra, es caracteritzen per tenir un nombre de registres relativament gran i sense cap funció específica assignada, als quals sol denominar-se com *R_{nn}*, sent *nn* un nombre que identifica al *registre*. El nombre de *registres* disponibles era un dels avantatges dels *RISC* sobre els *CISC*, ja que no es precisaven tants accessos a memòria per emmagatzemar resultats intermedis.

En l'actualitat no existeixen dissenys *RISC* o *CISC* purs, sinó que es tendeix a utilitzar el millor d'ambdós mons. Els microprocessadors *x86* dels últims anys, per exemple, compten amb un conjunt d'instruccions *CISC*, molt ampli i amb operacions complexes, però la *unitat d'execució* és de tipus *RISC*, existint una etapa intermèdia en la qual les instruccions són dividides en microoperacions més simples.

Segueixen existint els registres clàssics procedents del 8086 original, complementat amb un nombre més extens de *registres* similar als d'un processador *RISC*.

Arquitectura dels processadors x86

Sabent ja quina és, en termes generals, l'estructura d'un microprocessador qualsevol, s'endinsarà ara en els detalls de *l'arquitectura x86*, corresponent als microprocessadors que en l'actualitat es troben en milions de dispositius.

S'ha de tenir en compte que en aquests microprocessadors es conjuguen aspectes heretats, procedents del primer 8086 dissenyat fa més de 30 anys, amb els últims avanços de fabricació de circuits integrats que fan possibles, per exemple, l'existència de múltiples nuclis i conjunts d'instruccions molt avançats.

El conjunt d'instruccions d'aquests microprocessadors és de tipus *CISC*, si bé també incorpora algunes característiques de tipus *RISC*.

Les característiques més rellevants d'aquesta arquitectura són:

- L'arquitectura coneguda com a x86 comença en 1978 amb els processadors *Intel 8086/88* (encara que prèviament havien aparegut el 4004, el 8080 i el 8085.). Aquests processadors pertanyien a l'arquitectura *IA-16 (Intel Architecture 16 bits)*.
- Durant tota la seva evolució, des de 1978, *Intel* ha mantingut la compatibilitat binària amb els processadors precedents. Aquesta compatibilitat es trenca amb els processadors *Itanium* i *Itanium2*, amb arquitectura *IA-64*, que són totalment incompatibles amb els seus predecessors.
- El 80386 va ser el primer processador *d'Intel* amb un joc d'instruccions de 32 *bits (IA-32)*. Tant els operands com l'adreçament en memòria utilitzen 32 *bits*, per tant, l'i386 té un *espai d'adreçament* de 4 *GB*.
- El 80386 també és el primer a introduir una **MMU** [\[26\]](#) (*Memory Management Unit*) i la paginació, amb una grandària de pàgina fixa de 4 *KB*. Els processadors anteriors ja posseïen un esquema de segmentació, que en l'*Intel i386* es pot obviar utilitzant un esquema de memòria pla.
- A partir del 80486 *DX*, s'incorpora una *unitat de punt flotant* en el mateix encapsulat.

- L'espai d'adreces lineal és de 4 GB, encara que la memòria física pot arribar fins als 64 GB en alguns models.
- Un programa normal disposa de vuit *registres de propòsit general* de 32 bits, sis *registres de segment* de 16 bits, un *registre d'estat EFLAGS* i un *punter d'instrucció EIP*, ambdós de 32 bits. Aquests *registres* es poden accedir des de les operacions de *propòsit general*, compostes per les instruccions d'aritmètica sencera, les instruccions de control de flux, les d'operacions amb *bits* i amb cadenes de *Bytes*, i les instruccions d'accés a memòria.
- Es disposa, a més, d'un conjunt de vuit *registres de punt flotant* de 80 bits, un conjunt de vuit *registres MMX* i *XMM*, de 64 i 128 bits respectivament, per realitzar operacions *SIMD*. També es disposa de un conjunt de recursos per al maneig de la *pila* i la invocació de subrutines.
- El SO disposa a més de ports *d'E/S*, *registres de control*, de maneig de memòria, de depuració, de monitoratge, etc.

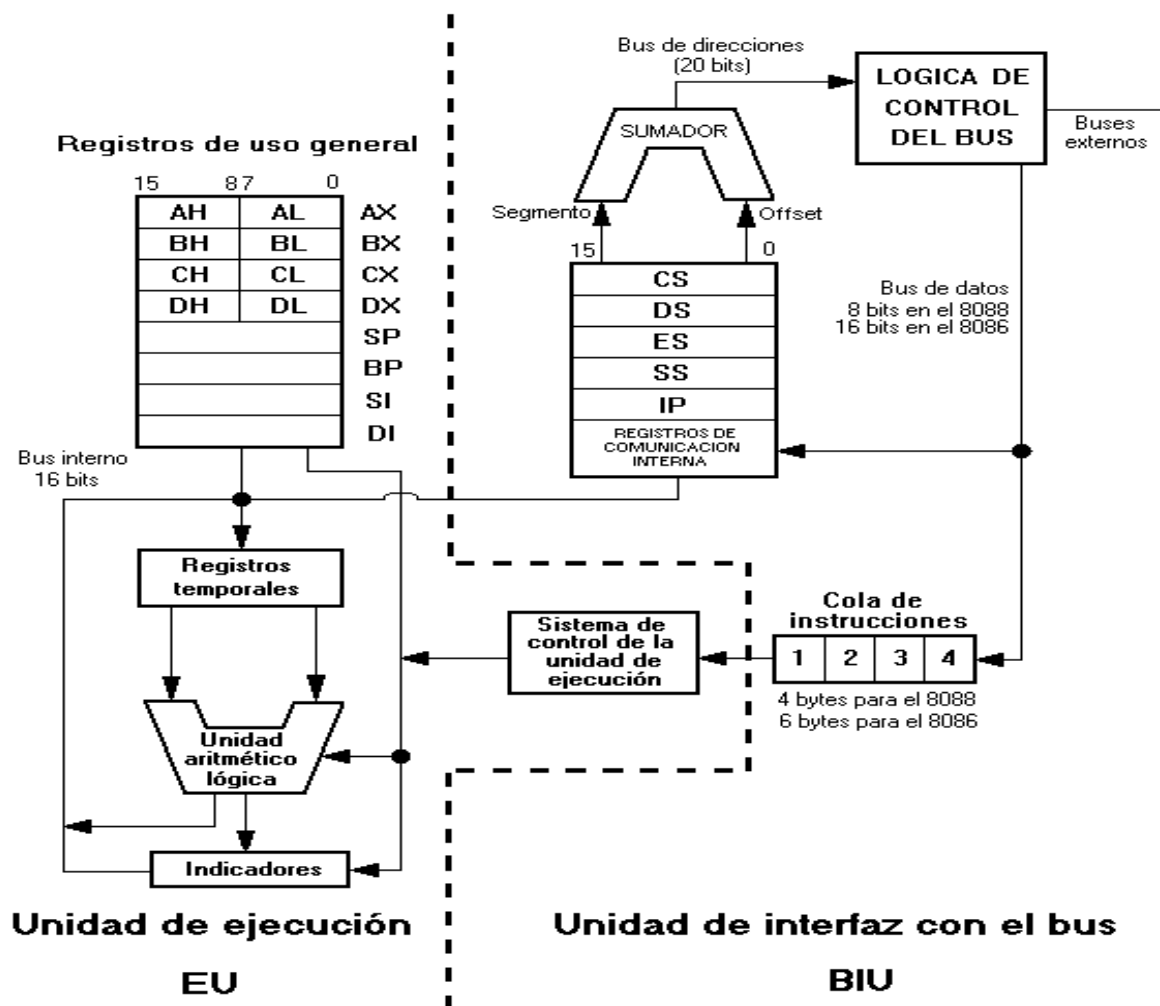


Figura 5. Estructura dels processadors 8086/8088

Modes de funcionament

Els microprocessadors x86 actuals, com l'*Intel Core i7* o l'*AMD Phenom*, compten amb diverses **modes de funcionament** [27] configurables per programari, segons els quals es comporten com un dels microprocessadors previs en l'arbre familiar. La manera de funcionament afecta a la capacitat i manera d'adreçament de la memòria, el banc de registres disponibles i, en menor mesura, les instruccions disponibles.

En els primigenis 8086/8088 no existien *modes de funcionament*, però la forma en què operaven, amb un accés segmentat a la memòria en tenir un *bus d'adreces* de 20 *bits* però *registres* amb només 16 *bits*, se li va denominar *mode real* a partir de l'aparició de l'*Intel 80286*.

Aquest *mode* és important perquè és el que s'activa en tots els microprocessadors x86 quan es posen en funcionament. Podria dir-se que tots ells, fins i tot els més moderns, es comporten com un 8086 molt ràpid, però amb les mateixes limitacions, *banc de registres* i *conjunt d'instruccions*.

Al dissenyar el 80286, un microprocessador amb un *bus d'adreces* més ample i noves característiques d'adreçament i protecció de memòria, *Intel* decideix que per defecte, per tal de mantenir la compatibilitat amb els milers d'aplicacions existents, es comporti com el 8086, apareixent llavors la denominació de *mode real*, per referir-se a l'antic *mode de funcionament*, i [*mode protegit*](#) per designar a aquell que permet utilitzar les millores del nou microprocessador.

El canvi d'un *mode* a un altre es duu a terme mitjançant programari, sent normalment el sistema operatiu l'encarregat de fer-ho.

Des del 80386 fins a una gran part de la família *Pentium/Athlon*, aquests microprocessadors van comptar amb els dos *modes* citats, el *real* i el *protegit*, aquest últim estès per operar amb adreçament de 32 *bits*, afegint un tercer denominat *mode virtual 8086*. Aquest *mode* permetia executar en *mode real* aplicacions no preparades per aprofitar el *mode protegit*, sent la base de les primeres màquines virtuals que feien possible utilitzar el *DOS* en una finestra *Windows*.

Amb la presentació de l'*Athlon 64* per part d'*AMD* apareix l'últim *mode* en què poden operar els actuals microprocessadors: el mode llarg, que també és conegut com mode *x86-64*. És un *mode* pensat per aprofitar l'ample estès de 64 *bits*, tant del *bus de dades* com d'*adreces*, així com un conjunt també ampliat de *registres* i instruccions.

El mode protegit

Des de l'aparició dels primers *PC*, a principis de la dècada dels 80, són milions d'aplicacions les que s'han desenvolupat per a *DOS*, el sistema operatiu que incorporaven en principi els ordinadors d'*IBM* i tots els compatibles apareguts amb posterioritat.

Aquest volum de programari, com és lògic, no podia canviar-se de la nit al dia per la presentació, per part d'*Intel*, d'un nou processador amb més possibilitats: el 80286. Era obligatori mantenir la compatibilitat cap a enrere, fent que aquest nou processador es comportés, en principi, com si fos un 8086.

Ja que no es podia saber per endavant si el sistema operatiu que s'anava a instal·lar en l'ordinador equipat amb un 80286 reconeixeria o no aquest nou processador, l'única solució era que, al connectar-se, el 80286 funcionés exactament igual que un 8086. A aquest *mode de funcionament* se li va conèixer a partir de llavors com a *mode real*, entenent per *real* el del primer microprocessador de la família x86.

Quin sentit tindria crear un nou microprocessador amb més capacitat, per exemple amb un *bus d'adreces* de 24 *bits* en lloc de 20, si no hi havia forma d'aprofitar-ho? Treballant en *mode real* no era possible accedir a aquesta capacitat, així que era necessari habilitar un nou *mode de funcionament*, *mode* al que es va denominar *protegit* pel fet que comptava amb mecanismes de protecció de la memòria.

Qualsevol programa, ja sigui una aplicació o un sistema operatiu, que volgués treballar en aquest *mode* havia d'activar-ho explícitament. L'*Intel* 80286 comptava amb un mecanisme documentat per passar del *mode real*, l'existent al posar-se en marxa l'ordinador, al *mode protegit*, però no per donar el pas invers. Un cop s'havia entrat en *mode protegit*, difícilment podia tornar-se al *mode real*.

No obstant això, empreses com *IBM* van enginyar diversos trucs per fer-ho possible dins de les limitacions existents.

La posterior aparició del processador 80386, l'any 1985, es va trobar amb el mateix problema ja existent: la pràctica totalitat de les aplicacions funcionaven en *DOS* en *mode real*. Per això, tots els microprocessadors de la família x86 funcionen en *mode real* quan es posen en marxa*.

El microprocessador Intel 80386 va millorar l'operació en *mode protegit*, fent possible el retorn al *mode real* si era necessari, i va afegir un nou mode de treball conegut com *V86*. En aquest mode, realment una extensió del mode protegit, el processador pot emular diversos 8086 funcionant de manera concurrent.

És el mode que fa possible, per exemple, obrir des de *Windows* una finestra de consola i tenir aplicacions *DOS* funcionant en ella. Realment aquesta consola no està operant en *mode real* sinó en *mode virtual* o *V86*.

(*) Fins el microprocessador d'*Intel* més recent, el *Core i7* amb múltiples nuclis i **HyperThreading** [28], es comporta com un 8086 quan es connecta, però, això sí, executant instruccions a velocitats properes als 3 GHz, en comptes dels 4,77 MHz com el 8086, i amb la capacitat de processar múltiples instruccions paral·lelament. Tindrem, per tant, un sistema molt més ràpid però amb el mateix límit dels 640 KB.

Banc de registres

El nombre i grandària dels registres disponibles en un microprocessador X86 dependrà, com s'ha indicat en el punt anterior, de la manera en què estigui operant. Existeix un conjunt mínim, el dels registres que incorporava el 8086, que s'ha anat ampliant amb cada nou membre de la família.

La figura 6 representa el conjunt de registres que es poden considerar de propòsit general, establint la denominació i relació que guarden entre si. En la part superior apareixen marcats, de dreta a esquerra, els bits que correspondrien a cada registre.

	63	31	15	8	7	0
RAX	EAX	AF	AH	AL		
RBX	EBX	BF	BH	BL		
RCX	ECX	CF	CH	CL		
RDX	EDX	DF	DH	DL		
RSI	ESI	SI				
RDI	EDI	DI				
Rn						
RBP	EBP	BP				
RSP	ESP	SP				

Figura 6. Registres de propòsit general en un processador x86.

Tots els x86, des del 8086 original, compten amb els *registres AX, BX, CX i DX* amb una grandària de 16 *bits* que, quan interessa, poden ser tractats com a *byte alt* i *byte baix* de manera independent, amb les denominacions *AH* i *AL* per *AX*, *BH* i *BL* per *BX* i així successivament. El sufix **H** prové de la paraula *High*, denotant que es tracta de la part alta, mentre el sufix **L** es deu a la paraula *Low* que indica que és la part baixa.

AH i *AL*, per tant, no són dos *registres* diferents a *AX*, sinó que junts formen *AX*. El mateix és aplicable per als altres tres registres. Els *registres SI* i *DI* també són de 16 *bits* però, a diferència dels quatre anteriors, no poden ser dividits en dues parts. Encara que són de *propòsit general*, en la majoria dels casos actuen com a índexs en operacions de transferència de dades.

BP i *SP* són *registres* de 16 *bits* i, encara que en certs casos poden emprar-se com a *registres de propòsit general*, estan pensats per operar específicament com a *punters de pila*.

Quan es treballa en *mode protegit* en un entorn de 32 *bits*, el conjunt inicial de *registres* s'amplia per donar cabuda a *registres* de 32 *bits*. Aquests tenen el mateix nom que els originals però amb el prefix **E**, de manera que *EAX* és un *registre* de 32 *bits*, dels quals els primers 16 *bits* es segueixen denominant *AX*.

Si el *mode* en què està treballant el microprocessador és el *mode llarg*, específic per a sistemes operatius de 64 *bits*, els *registres de propòsit general* s'expandeixen novament a aquesta grandària i adquireixen la denominació original precedida per la lletra **R**.

Per tant, el registre *RAX* és de 64 *bits*, i s'utilitza *EAX* per operar sobre els 32 *bits* de menor pes o *AX* pels 16 *bits* de menor pes, així com *AH* i *AL* per treballar amb els dos *Bytes* d'*AX* de manera independent. A més, s'agreguen vuit *registres* més de *propòsit general*, tots ells de 64 *bits* i inexistents en la resta de *modes*. Aquests nous *registres* s'anomenen: *R8*, *R9* ... *R14* i *R15*.

Als *registres de propòsit general* ja citats cal sumar un altre conjunt, de caràcter més específic, que es pot observar en la [figura 7](#). En la part superior apareixen el *registres IP* i *F*, denominació en *mode real*, que s'estenen com *EIP/EF* en *mode protegit* de 32 *bits* i com *RIP/RF* en *mode llarg* de 64 *bits*. *IP* (*Instruction Pointer*) és el *registre* que manté l'adreça de memòria on està allotjada la següent instrucció a executar.

El contingut del *registre IP* no està sota el control directe dels programadors d'aplicacions, sinó que s'actualitza automàticament a mesura que s'executen instruccions, avançant de manera seqüencial en la majoria dels casos o bé de manera directa davant instruccions de salt.

El *registre F* (*Flags*) es denomina habitualment [registre d'estat](#) i es veurà, amb major detall, en el següent apartat d'aquest estudi.

Els *registres de segment* són sempre de 16 *bits*, en qualsevol *mode* de treball, si bé la forma d'utilitzar-los difereix d'un *mode* a un altre. En *mode real* s'utilitzen en combinació amb altres *registres* per compondre adreces físiques de memòria, mentre en la resta de *modes* el contingut d'aquests *registres* no és una adreça física sinó un *selector*.

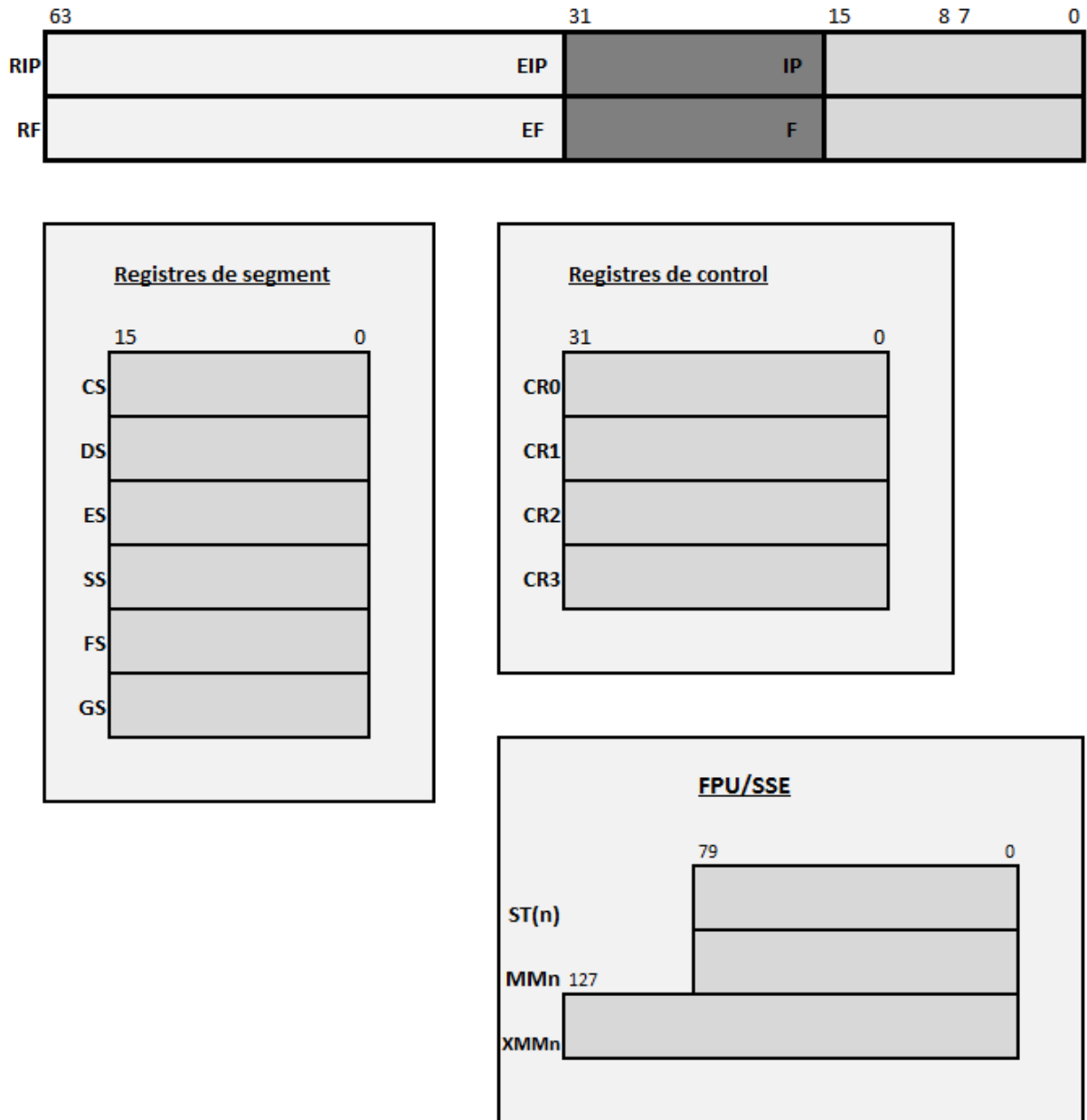


Figura 7. Registres d'ús específic en un microprocessador x86.

Els registres FS i GS es van agregar en l'Intel 80386, mentre que els altres quatre existeixen des del 8086 original. Per defecte certs registres de segment s'associen a blocs de memòria determinats dels programes, utilitzant-se en combinació amb registres de propòsit general o específics concrets. La norma a qualsevol programa és la següent:

- **CS (Code Segment):** S'utilitza amb combinació amb IP per determinar l'adreça on es troba la propera instrucció a executar.

- **SS** (*Stack Segment*): La majoria dels programes necessiten d'una *pila* per poder funcionar, emmagatzemant aquest *registre* la localització del segment de memòria utilitzat per tal fi. Els *registres* que s'associen a aquest, per comprendre l'adreça final, són *SP* (*Stack Pointer*) i *BP* (*Base Pointer*).
- **DS** (*Data Segment*): Determina on s'inicia el segment de dades, utilitzant-se en combinació amb *registres de propòsit general*, com *BX* o *SI*, per obtenir adreces que permetin accedir a les dades.
- **ES** (*Extra Segment*): Sol utilitzar-se com a segon recurs per a accés a dades i en certes ocasions, com les de transferència de dades en cadena, s'associa al *registre DI*.

Els *registres FS* i *GS* no tenen una finalitat específica i, igual que *ES*, poden utilitzar-se en qualsevol context on es precisi accedir a un segment de memòria addicional.

A la dreta dels *registres de segment* (veure la [figura 7](#)) apareix un bloc denominat *registres de control*, format pels *registres CR0*, *CR1*, *CR2* i *CR3*. Aquests són de 32 *bits* en *mode protegit* i de 64 *bits* en *mode llarg*, sent la seva finalitat modificar o informar sobre diferents aspectes del funcionament del microprocessador.

L'últim conjunt de *registres* mostrat en la [figura 7](#) és el dels quals incorporen la *FPU* (*Floating Point Unit*), part de l'*ALU* que s'encarrega dels càlculs en *punt flotant*, i la unitat encarregada de les instruccions del tipus *MMX/SSE/3DNow*. Els *registres ST(0)* a *ST(7)* s'estructuren en forma de *pila* i, amb una grandària de 80 *bits*, faciliten la realització d'operacions en simple i doble precisió.

Els *registres MM0* a *MM7*, lligats als anteriors, van ser els que es van introduir en els primers *Pentium MMX* per accelerar l'execució de certes operacions relacionades amb multimèdia. Finalment els *registres XMM0* a *XMM7* (fins a *XMM15* en els microprocessadors actuals) són de 128 *bits* i amplien les possibilitats del conjunt d'instruccions *MMX/SSE*.

El registre d'estat

El registre d'estat del processador és molt important, doncs s'encarrega de guardar, i d'informar, de l'estat del microprocessador en tot moment. Aquest registre, en l'arquitectura x86, és de 16 *bits*. Cadascun d'aquests bits té un significat particular. A la Figura 8 es pot observar com s'organitza aquest registre.

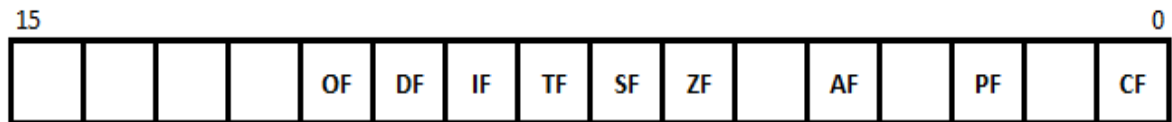


Figura 8. Estructura del registre d'estat d'un microprocessador x86.

A continuació es descriuen els nou *bits* que són comuns per a tota la família x86:

- **CF** (*Carry Flag*): Aquest indicador comunica que s'ha produït un acarregit durant una operació aritmètica, per exemple al sumar dos nombres de 16 *bits* i obtenir un resultat que necessita un *bit* addicional per poder ser emmagatzemat.
- **PF** (*Parity Flag*): Indica la paritat del resultat de l'última operació executada.
- **AF** (*Auxiliary Carry Flag*): Aquest *bit* és útil amb treballar amb aritmètica *BCD*, perquè indica l'acarregit del primer *nibble* (paquet de 4 *bits*) al segon dins d'un *Byte*.
- **ZF** (*Zero Flag*): Comunica que el resultat de l'última operació efectuada ha estat zero.
- **SF** (*Sign Flag*): Després de qualsevol operació aritmètica, aquest *bit* indicarà el signe del resultat.
- **TF** (*Trap Flag*): Aquest bit es utilitza pels *debuggers* per facilitar l'execució pas a pas dels programes.
- **IF** (*Interrupt Flag*): Determina si el microprocessador atindrà o no les interrupcions enmascarables.
- **DF** (*Direction Flag*): Controla el sentit en què es recorren zones de memòria amb determinades instruccions de repetició, incrementant o reduint el contingut dels registres *SI* i *DI*.

- **OF** (*Overflow Flag*): Indica que l'última operació ha generat un resultat que no es pot emmagatzemar en la destinació corresponent, és a dir, "la destinació s'ha desbordat".

En aquest apartat s'ha introduït, de manera superficial, el que és un microprocessador genèric i el funcionament de les seves parts més rellevants. També s'han tractat les característiques pròpies de l'arquitectura x86 i a més, han estat presentades les parts més fonamentals d'aquest tipus de microprocessadors i s'ha descrit el seu funcionament. A continuació es presenten els diferents processadors que conformen la família x86.

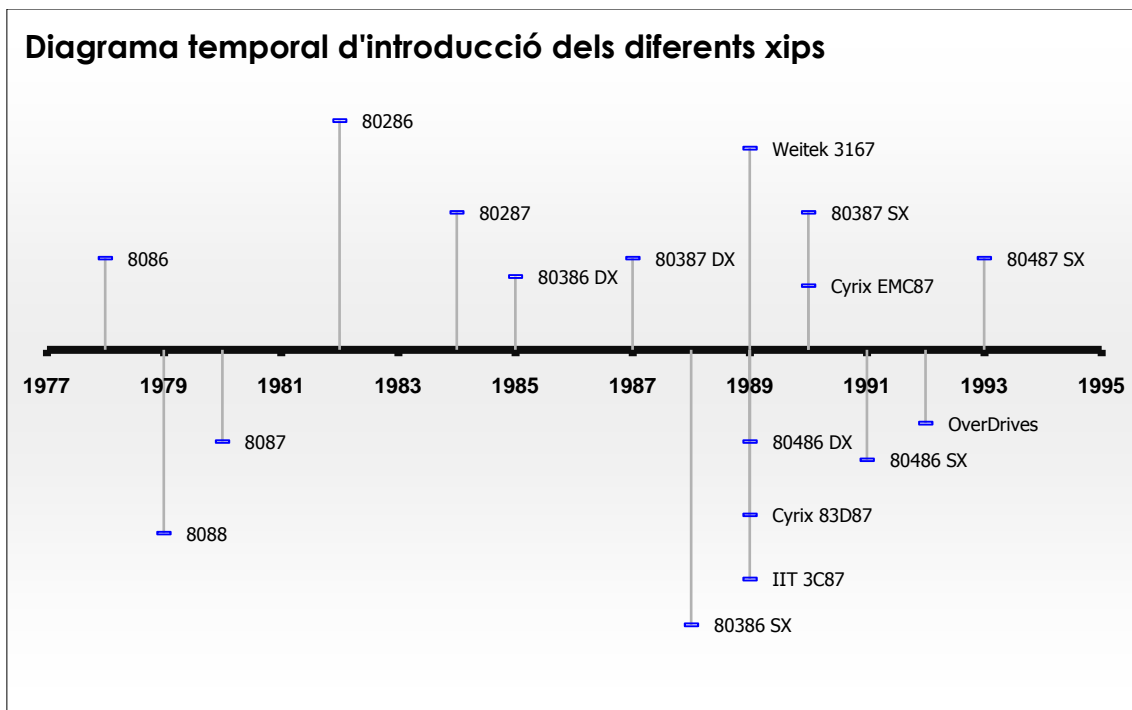


Figura 9. Diagrama temporal de l'aparició dels xips que es tracten en aquest estudi.

Als següents apartats s'introdueixen les quatre primeres generacions de microprocessadors que formen l'*arquitectura x86* i que són la base d'aquest estudi. Es començarà amb el processador *Intel 8086*, primer membre d'aquesta família, fins arribar, finalment, al 80486 i les seves variants més rellevants.

8.2. L'Intel 8086/8088

A mitjans de la dècada dels 70, *Intel* estava desenvolupant un nou microprocessador que inclogués el nou estàndard per a *punt flotant* del *IEEE*. Aquest processador es convertiria en el 8086, el predecessor del 8088, que va servir com a base per a l'*IBM PC*.

En el moment en què s'estava desenvolupant el 8086, abans de la seva comercialització en 1.978, crear un circuit integrat era un procés molt més exòtic del que ho és ara. La tecnologia era encara primitiva, i la grandària i nombre dels components que es podien afegir i gravar en una fina placa de silici limitava la complexitat dels possibles (o almenys permissibles) circuits integrats.

Llavors els microprocessadors eren els circuits integrats més complexos que es podien dissenyar. Com més gran fos el xip, més probable era que contingués algun defecte que ho deixés inservible. D'altra banda, la tecnologia reduïa al màxim possible els detalls del disseny del xip. Si els traçats eren petits, també existien possibilitats que sofrissin defectes de fabricació, deixant de nou al processador inservible [\[29\]](#).

Per exemple, els detalls més petits possibles del microprocessador 8086 mesuraven de cinc a deu *microns*. El 80486 *DX* es fabricava amb detalls tan diminuts com un *micró*. Junts, aquests dos límits conspiraven per posar fi a la complexitat dels circuits. El propi 8086 lluitava afanosament contra aquests límits. Fer el xip cada vegada més complex resultava impossible.

8.2.1. L'Intel 8086: l'origen

L'Intel **8086** [\[30\]](#), oficialment anomenat *iAPX 86*, és un microprocessador de 16 *bits* dissenyat per *Intel* l'any 1.978. És el primer processador de l'*arquitectura x86*. El seu disseny està basat en el processador *8080* (un xip d'*Intel* de propòsit general desenvolupat en 1.974, de vuit *bits*, que contenia 4.500 transistors i podia executar unes 200.000 instruccions per segon).

El 8086 va ser dissenyat per treballar amb llenguatges d'alt nivell, disposant d'un suport maquinari amb el qual els programes escrits en aquests llenguatges ocupen un petit espai de codi i es poden executar a gran velocitat. Aquesta concepció, orientada a l'ús de compiladors, es materialitza en un conjunt de facilitats i recursos, i en unes instruccions que faciliten moltes operacions.

En el seu moment, el 8086 juntament amb el 8088 van ser els microprocessadors més emprats dins de la seva categoria, especialment des que *IBM* els va adoptar per a la construcció de la seva [computadora personal](#). Molts fabricants de microordinadors van utilitzar aquesta família de microprocessadors per fabricar equips destinats al mercat professional.

Avui dia, la utilització de l'*Intel* 8086 és molt reduïda, quedant principalment orientat a l'ensenyament, com a base dels microprocessadors de l'última generació.



Figura 10. Imatge de l'*Intel* 8086 original.

A continuació s'enumeren les característiques més destacables del microprocessador *Intel* 8086:

- El 8086 disposa d'instruccions especials per al tractament de cadenes de caràcters.
- Els *registres* del 8086 tenen una missió específica, per la qual cosa es podria dir que cadascun d'ells té la seva pròpia personalitat, encara que varis comparteixen tasques comunes.
- L'*encapsulat* del 8086 està format per 40 pins, simplificant així el *maquinari*, encara que per contra, és necessari la multiplexació del *bus de dades* amb el *d'adreces*.
- El 8086 disposa d'un conjunt de *registres*, denominats "*cuà d'instruccions*", en el qual es van emmagatzemant de forma anticipada els codis de les instruccions, aconseguint que aquest augmenti la seva velocitat de treball.
- Les 20 línies del *bus d'adreces* només permeten adreçar una memòria d'un *Megabyte*.
- El 8086 requereix un senyal de rellotge exterior, sent 5 i 8 *MHz* les freqüències típiques de funcionament, encara que una versió del xip treballa a 10 *MHz*.
- El 8086 disposa d'una arquitectura [pipeline](#), és a dir, que la *CPU* pot seguir llegint instruccions en els temps en què el *bus* no s'utilitza.

Els 14 *registres* de 16 *bits* que té el 8086 es poden classificar en tres grups d'acord amb les seves funcions. El *grup de dades*, que és essencialment el conjunt de registres aritmètics; el grup *d'apuntadors*, que inclou els registres *base i índex* i, també, el *comptador de programa*; i el grup de *registres de segment*, que són *registres de propòsit especial* (veure [Banc de registres](#)).

El 8086 fa servir un esquema anomenat **segmentació** [31] per accedir correctament a un *Megabyte* complet de memòria, amb referències d'adreces de només 16 *bits*, i tot això gràcies a la utilització de *registres de segment* que divideixen essencialment l'espai de memòria en *segments* de 64 *KB* de longitud, que poden estar separats entre si, adjacents o superposats, i que comencen en una adreça divisible per 16.

La forma en què es completen els 20 *bits* del *bus d'adreces*, disposant en la *CPU*, solament, registres de 16 *bits*, s'aconsegueix de la següent manera:

Es parteix del contingut d'un dels *registres de segment*, que actuen com a base. Després, es multiplica per 16 el contingut del *registre de segment*, la qual cosa, en binari, significa afegir-li 4 zeros a la dreta i convertir-ho en una magnitud de 20 *bits*. Finalment, se suma un desplaçament al resultat de la multiplicació anterior.

Així doncs, la fórmula per calcular una adreça de memòria és:

$$\text{Adreça Física} = 16 \cdot (\text{registre de segment}) + \text{desplaçament}$$

D'aquesta manera, sobre l'adreça base que apunta el *registre de segment* multiplicat per 16, existeix un marge de 64 *KB*, controlat per un desplaçament de 16 *bits*.

Igual que els més recents microprocessadors, el 8086 requereix un únic senyal de rellotge. Aquest microprocessador no genera el seu propi senyal de rellotge sent necessari la utilització del *generador de rellotge 8284*, que usa un cristall oscil·lador per determinar la freqüència de senyal. Intercanviant aquest cristall, es pot seleccionar diferents velocitats d'operació.

Intel té una versió de 5 *MHz* i una altra de 8 *MHz* pel 8086. Aquestes versions representen les velocitats més altes, recomanables per a aquest xip. Per a un rendiment òptim, el 8086 requereix un senyal de rellotge que es mantingui a tensió alta una tercera part del temps total de cicle. Això significa que el rellotge està actiu una tercera part del temps i desactivat les dues terceres parts del temps.

NOTA: Per conèixer més detalls, com el connexionat, els modes de treball (Màxim i Mínim) o els xips de suport del 8086, es recomana visitar: <http://www.alpertron.com.ar/8088.HTM>.

8.2.2. El 8088: El naixement del “Personal Computer”

El microprocessador Intel **8088** [32] va ser presentat a l'any 1979, només un any més tard que el 8086. Ambdós processadors tenen la mateixa arquitectura i l'única diferència entre el 8088 i el 8086 és l'amplada del *bus de dades extern*, que es va reduir de 16 *bits* a 8 *bits*. El 8088 utilitza dos cicles de *bus* consecutius per llegir o escriure 16 bits de dades, en comptes d'un sol cicle com el 8086. Això fa que el 8088 sigui més lent.

Aquest canvi en el *maquinari* es va fer per garantir la compatibilitat amb els xips 8080/8085. Aquest va ser un dels factors determinants en escollir el processador 8088 per la línia d'ordinadors *PC* d'*IBM*, perquè en aquell temps els xips de suport de 8 *bits* eren molt més econòmics que els xips de 16 *bits*.



Figura 11. Imatge del microprocessador *Intel 8088-2*

El 8088 té registres de 16 *bits*, *bus de dades* intern de 16 *bits* i *bus d'adreçament* de 20 *bits*, que permet al processador adreçar fins a 1 *MB* de memòria. El 8088 utilitza el mateix esquema de *segmentació* de memòria que el 8086. Igual que aquest últim, l'Intel 8088 suporta el *coprocessador matemàtic* 8087.

Aquesta *CPU* reconeix totes les instruccions en *punt flotant* i, quan és necessari, calcula l'adreça de memòria per l'operand de la instrucció en *punt flotant* i realitza una falsa lectura de memòria. El coprocessador captura l'adreça calculada i, possiblement, les dades i procedeix a executar la instrucció en *punt flotant*.

Al mateix temps, el microprocessador 8088, comença l'execució de la següent instrucció. Així doncs, ambdues instruccions enteres i instruccions en *punt flotant* es poden executar concurrentment.

L'Intel 8088 original va ser fabricat utilitzant tecnologia *HMOS*, hi havia, també, versions *CHMOS* del xip. Aquests microprocessadors tenien un consum energètic molt més baix i presentaven un *mode standby*.

8.3. Processador 80286. L'aparició de l'*IBM PC AT*

Aquest microprocessador va aparèixer al febrer de 1.982. Els avanços d'integració que van permetre agregar una gran quantitat de components perifèrics a l'interior del 8086/8088, es van utilitzar en el 80286 [32] per fer un microprocessador que suporti noves capacitats, com la *multitasca* (execució simultània de diversos programes), la qual cosa requereix que els programes no "xoquin" entre si, alterant un les dades o les instruccions d'altres programes.

El 80286 té dos modos d'operació: mode real i mode protegit. En la mode real, es comporta igual que un 8086, mentre que en mode protegit, les coses canvien completament. Això va necessitar un nivell d'integració molt major. El 80286 conté 134.000 transistors dins de la seva estructura (un 360% més que el 8086).

Externament està encapsulat en format *PLCC* (*Plastic Leaded Chip Carrier*) amb pins en forma de **J** per a muntatge superficial, o en format *PGA* (*Pin Grid Array*), en ambdós casos amb 68 pins. El microprocessador 80286 ha afegit un nou nivell de satisfacció a l'arquitectura bàsica del 8086, incloent una gestió de memòria amb l'extensió natural de les capacitats d'adreçament del processador.

L'Intel 80286 té incorporades unes facilitats de protecció de dades molt elaborades. Altres característiques inclouen totes les característiques del joc d'instruccions del 80186, així com l'extensió de l'espai adreçable a 16 *MB*, utilitzant 24 *bits* per adreçar ($2^{24} = 16.777.216$).



Figura 12. Imatge d'una *CPU Intel 80286*.

El 80286 revisa cada accés a instruccions o dades per comprovar si pot haver-hi una violació dels drets d'accés. Aquest microprocessador està dissenyat per usar un sistema operatiu amb diversos nivells de privilegi. En aquest tipus de sistemes operatius hi ha un nucli que, com el seu nom indica, és la part més interna del sistema operatiu. El nucli té el màxim privilegi i els programes d'aplicacions el mínim.

Existeixen quatre nivells de privilegi. La protecció de dades en aquest tipus de sistemes es duu a terme tenint segments de codi (que inclou les instruccions), dades (que inclou la *pila* a part de les variables dels programes) i del sistema (que indiquen els drets d'accés dels altres *segments*). Per a un usuari normal, *els registres de segmentació* (*CS, DS, ES, SS*) semblen tenir els 16 *bits* usuals.

No obstant això, aquests *registres* no apunten directament a memòria, com ho feien en el 8086. En el seu lloc, apunten a taules especials, cridades *taules de descriptors*, algunes de les quals tenen a veure amb l'usuari i unes altres amb el sistema operatiu. Actualment als 16 *bits*, cada *registre de segment* del 80286 manté altres 57 *bits* invisibles per a l'usuari.

Vuit d'aquests *bits* serveixen per mantenir els drets d'accés (només lectura, només escriptura i uns altres), altres *bits* mantenen l'adreça real (24 *bits*) del principi del *segment* i uns altres mantenen la longitud permesa del segment (16 *bits*, per tenir la longitud màxima de 64 *KB*). Per això, l'usuari mai sap en quina posició real de memòria està executant o on se situen les dades i sempre es manté dins de certes fronteres.

Com a protecció addicional, mai es permet que l'usuari escrigui en el segment de codi (en mode real es pot escriure sobre aquest segment). Això prevé que l'usuari modifiqui el seu programa per realitzar actes il·legals i potencialment perillosos. Hi ha també provisions per preveure que l'usuari introdueixi en el sistema un "Troia" que pugui proporcionar-li un estat d'alt privilegi.

El 80286 té quatre nous *registres*. Tres d'ells apunten a les *taules de descriptors* actualment en ús. Aquestes taules contenen informació sobre els objectes protegits en el sistema. Qualsevol canvi de privilegi o de segment ha de realitzar-se a través d'aquestes taules. Addicionalment hi ha diversos indicadors nous.

Existeixen diverses instruccions noves, a banda de totes les del processador 8086 i les introduïdes amb el 80186. Totes aquestes instruccions es refereixen a la gestió de memòria i protecció del sistema, fent diverses tasques com carregar i emmagatzemar el contingut dels indicadors especials i els *punters* a les *taules de descriptors*.

Intel 80286 architecture

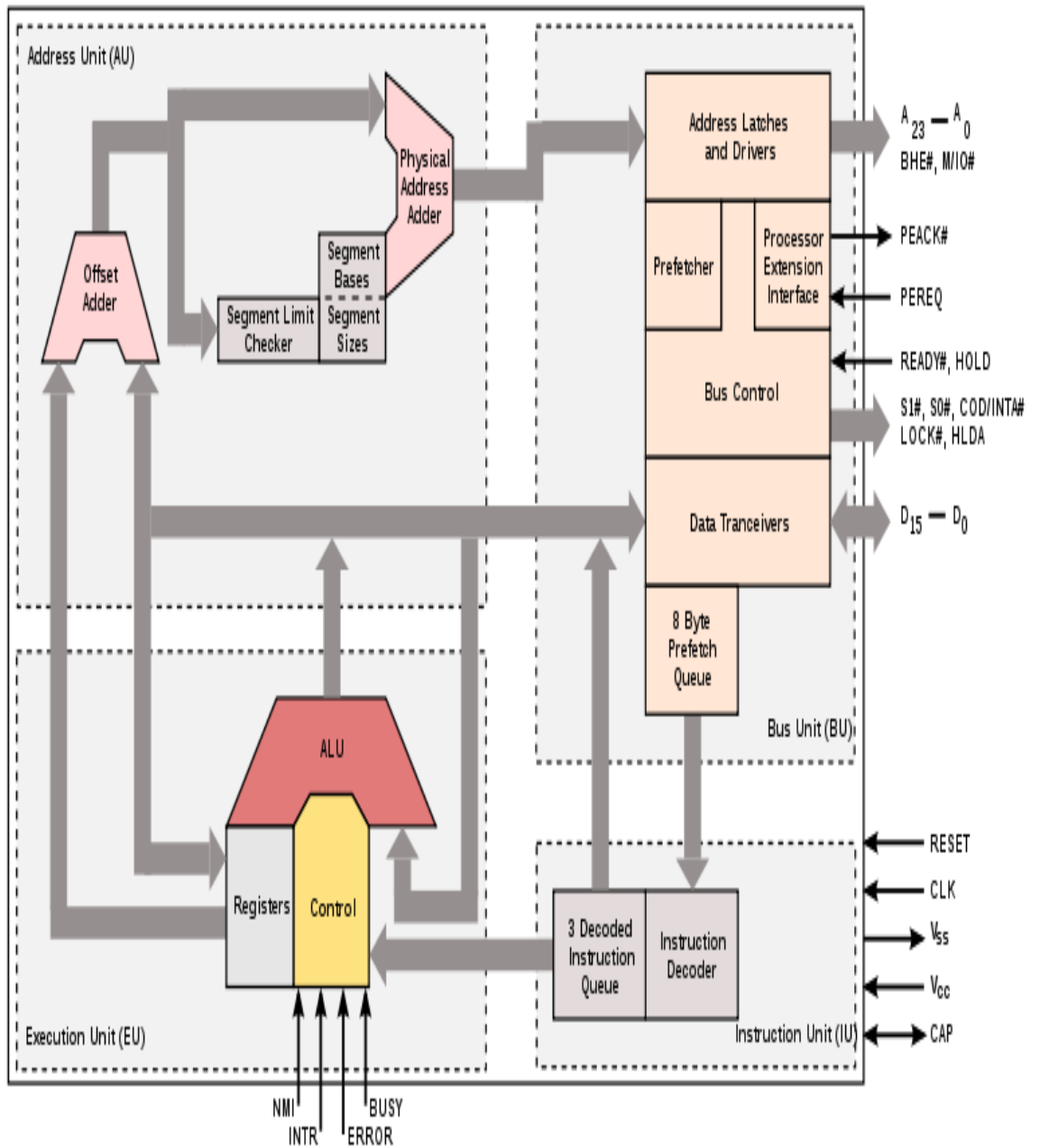


Figura 13. Arquitectura del microprocesador 80286.

8.4. L'Intel 80386. L'era dels 32 bits

El microprocessador 80386 es va presentar a l'any 1.985, va ser el primer processador *Intel* amb *bus de dades* i *bus d'adreces* de 32 bits. Pot adreçar quatre *Gigabytes* (2^{32} Bytes) de memòria, en comptes dels 16 *Megabytes*, com a màxim, de l'*IBM PC* típic. El 80386 permet l'execució de múltiples programes d'aplicació alhora (quan corre sota sistemes operatius específics de l'i386) utilitzant el mode protegit.

8.4.1. L'Intel 80386 DX

El 80386 *DX* consisteix en una *unitat central de procés (CPU)*, una *unitat de maneig de memòria (MMU)* i una *unitat d'interfície amb el bus (BIU)*. La *CPU* està composta per la *unitat d'execució* i la *unitat d'instruccions*. La *unitat d'execució* conté els vuit *registres* de 32 bits de *propòsit general* que s'utilitzen per al càlcul d'adreces i operacions amb dades i un **barrel shifter** [33] de 64 bits que s'utilitza per accelerar les operacions de desplaçament, rotació, multiplicació i divisió.

Al contrari dels microprocessadors previs, la lògica de divisió i multiplicació utilitza un algorisme d'1 bit per cicle de rellotge. L'algorisme de multiplicació acaba la iteració quan els bits més significatius del multiplicador són tots zeros, la qual cosa permet que les multiplicacions típiques de 32 bits es realitzin en menys d'un microsegon.

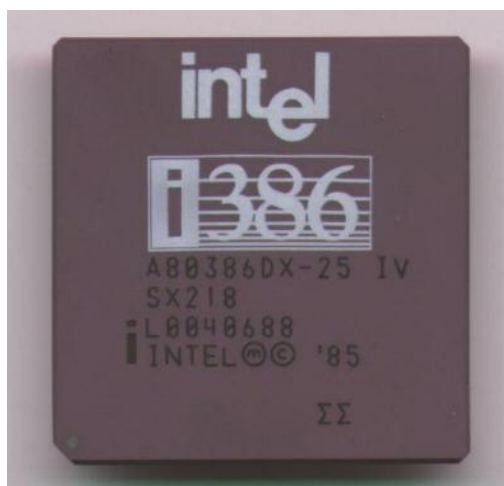


Figura 14. Detall d'un microprocessador *Intel 80386 DX*.

La *unitat d'instrucció* descodifica els codis d'operació (*opcodes*) de les instruccions que es troben en una *cua d'instruccions* (la longitud de les quals és de 16 Bytes) i els emmagatzema en la *cua d'instruccions* descodificats (hi ha espai per a tres instruccions).

El sistema de control de la *unitat d'execució* és l'encarregat de descodificar les instruccions que li envia la *cua* i enviar-li les ordres a l'*ALU* segons una taula que té emmagatzemada en *ROM* anomenada *CROM (Control Read Only Memory)*.

La *unitat de maneig de memòria (MMU)* consisteix en una *unitat de segmentació* (similar a la del 80286) i una *unitat de paginat* [34] (nova en aquest microprocessador). La *segmentació* permet el maneig de l'espai d'adreces lògiques agregant un component d'adreçament extra, que permet que el codi i les dades es puguin "resituar" fàcilment. El mecanisme de paginat opera per sota i és transparent al procés de segmentació, per permetre el maneig de l'espai d'adreces físiques.

Cada segment es divideix en un o més pàgines de 4 *KB*. Per implementar un sistema de memòria virtual (aquell on el programa té una grandària major que la memòria física i ha de carregar-se per parts (pàgines) des del disc dur), el 80386 *DX* permet seguir executant els programes després d'haver-se detectat fallades de segments o de pàgines.

Si una pàgina determinada no es troba en memòria, el 80386 *DX* ho indica al sistema operatiu mitjançant l'*excepció 14*, després aquest carrega aquesta pàgina des del disc i finalment pot seguir executant el programa, com si hagués estat aquesta pàgina tot el temps. Com es pot observar, aquest procés és transparent per a l'aplicació, per la qual cosa el programador no ha de preocupar-se per carregar parts del codi des del disc ja que això ho fa el sistema operatiu amb l'ajuda del microprocessador.

La memòria s'organitza en un o més segments de longitud variable, amb grandària màxima de 4 *Gigabytes*. Aquests segments, tenen atributs associats, que inclouen la seva ubicació, grandària, tipus (*pila*, codi o dades) i característiques de protecció. La *unitat de segmentació* proveeix quatre nivells de protecció per aïllar i protegir les aplicacions i el sistema operatiu.

Aquest tipus de protecció per maquinari permet el disseny de sistemes amb un alt grau d'integritat. El 80386 *DX* té dos *modos d'operació*: *mode d'adreçament real* (mode real), i *mode d'adreçament virtual protegit* (*mode protegit*). En *mode real* el 80386 opera com un 8086 molt ràpid, amb extensions de 32 *bits* si es desitja. El mode real es requereix primàriament per preparar el processador perquè operi en mode protegit.

El *mode protegit* proveeix l'accés al sofisticat *maneig de memòria i paginat*. Dins del mode protegit, el programari pot realitzar un canvi de tasca per entrar en tasques en *mode 8086 virtual (V86 mode)* (això és nou amb aquest microprocessador). Cadascuna d'aquestes tasques es comporta com si fos un 8086 el que ho està executant, la qual cosa permet executar programari de 8086 (un programa d'aplicació o un sistema operatiu).

Les tasques en *mode 8086 virtual* es poden aïllar entre si i del sistema operatiu (que ha d'utilitzar instruccions del 80386), mitjançant l'ús del *paginat* i el *mapa de bits* de permís d'E/S. Finalment, per facilitar dissenys de maquinari d'alt rendiment, la interfície amb el bus del 80386 *DX* ofereix *pipelining* d'adreces, grandària dinàmica de l'ample del *bus de dades* (pot tenir 16 o 32 *bits* segons es desitgi en un determinat cicle de *bus*) i senyals d'habilitació de *Bytes* per cada *Byte* del *bus de dades*.

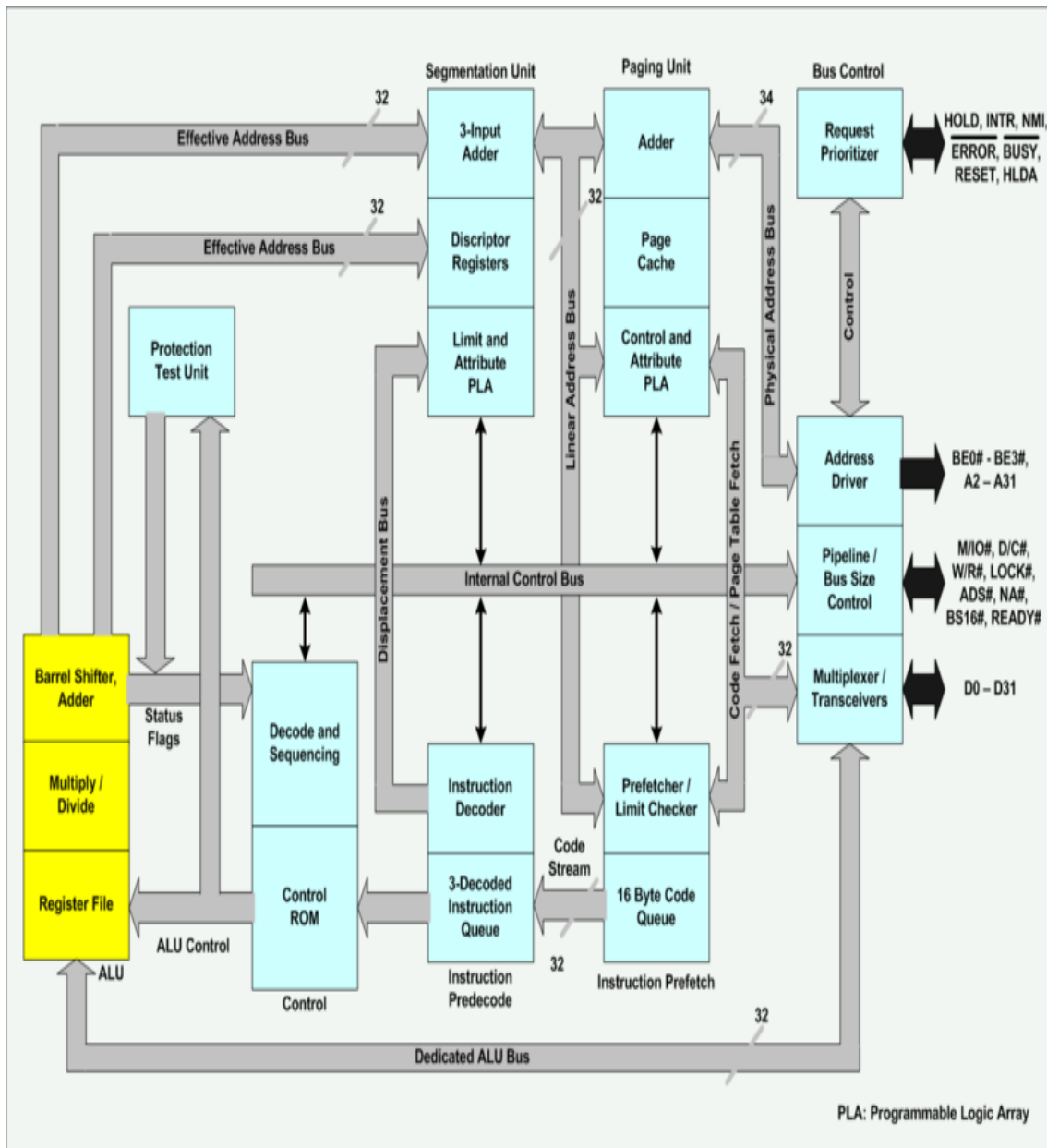


Figura 15. Arquitectura del 80386 DX.

NOTA: Per conèixer amb profunditat més detalls del microprocessador 80386, es recomana visitar: <http://developer.intel.com/Assets/PDF/manual/253665.pdf>

8.4.2. La variant 80386 SX

L'any 1.988, *Intel* va introduir el 80386 SX, una versió de baix cost del 80386 amb un *bus de dades* de 16 bits. Aquest microprocessador té una arquitectura de 32 bits, però es comunica amb l'exterior mitjançant un [bus extern](#) de 16 bits, i té un [bus de d'adreces](#) de 24 bits. Això fa que només pugui adreçar 16 MB. Com a conseqüència de tenir els busos més "estrets", la variant SX de l'i386 treballa "a la meitat" de velocitat de la variant DX amb la mateixa freqüència de rellotge.



Figura 16. Imatge d'una CPU Intel 80386 SX.

Una de les principals diferències, a banda de l'ample dels *busos de dades i adreces*, entre el microprocessador 80386 i els seus predecessors, el 8086 i el 80286, és la *segmentació de memòria*. A la figura 17, es poden comparar els tres models de maneig de memòria que s'han introduït en aquest estudi.

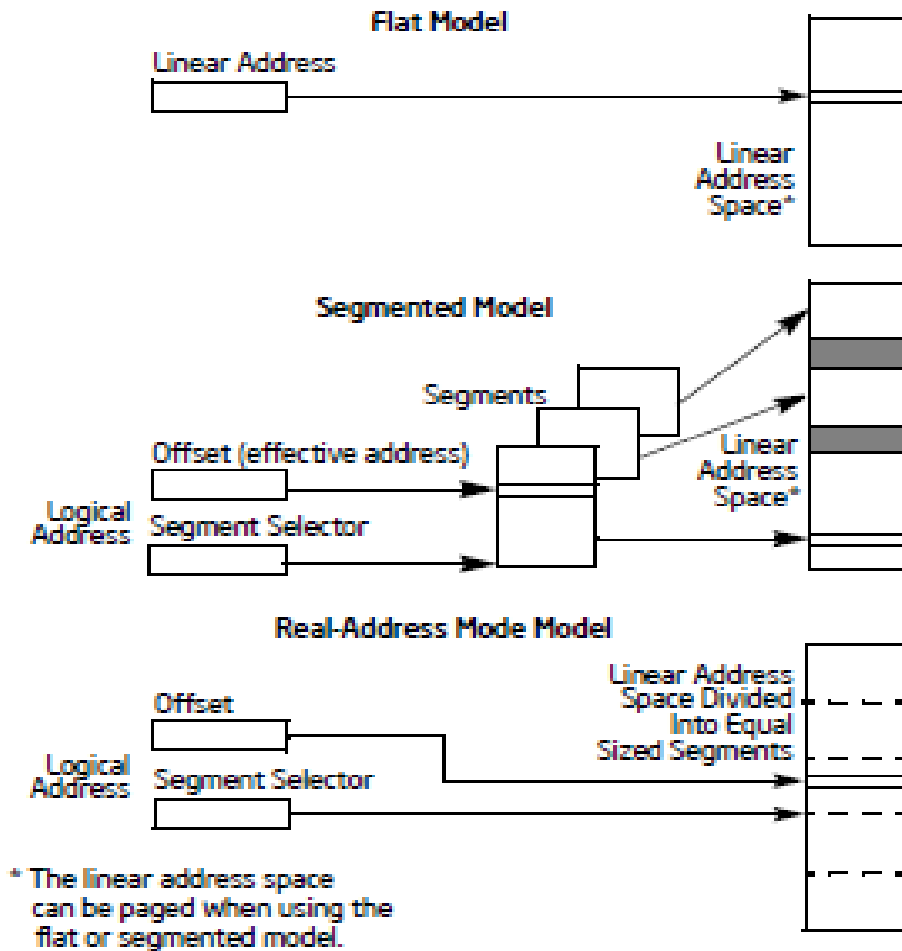


Figura 17. Els tres models de maneig de memòria utilitzats pels xips Intel.

8.5. L'Intel 80486

L'Intel 80486 [35] va ser presentat l'any 1989 i és molt similar al seu predecessor, l'80386. Les diferències principals entre ambdós xips són un *conjunt d'instruccions* optimitzat, una *memòria cau* unificada de dades i instruccions interna, una unitat de *punt flotant (FPU)* interna segons la versió, i una unitat d'*interfície de bus* millorada.

Aquestes millores li concedeixen el doble de potencia que un microprocessador 80386 amb la mateixa velocitat de rellotge.

A continuació es descriuran les característiques més importants d'aquest microprocessador i s'introduiran breument les diferents versions que es van comercialitzar.

8.5.1. Un autèntic gegant: el 80486 DX

Com ja s'ha comentat, aquest microprocessador és bàsicament un 80386 amb l'agregat d'una *unitat de punt flotant* compatible amb el coprocessador matemàtic Intel 80387 i una *memòria cau* de 8 KB. Per tant, els blocs que componen l'Intel 80486 són els següents:

- **Unitat d'execució:** Inclou els registres d'ús general de 32 bits, la unitat *lògica-matemàtica* i un *barrel shifter* de 64 bits. La *unitat d'execució* està millorada, així doncs, es necessita un només cicle de rellotge per executar les instruccions més freqüents.
- **Unitat de segmentació:** Inclou els *registres de segment*, els "cachés" d'informació de *descriptors* i la *lògica de protecció*. No té diferències pel que fa al microprocessador 80386.
- **Unitat de paginació:** És l'encarregada de traduir les *adreces lineals* (generades per la unitat anterior) en *adreces físiques*. Inclou el *buffer de conversió per recerca (TLB)* [36]. Els últims models (DX4 i alguns DX2) suporten pàgines de 4 MB a banda de les de 4 KB del 80386.
- **Unitat de "caché":** L'evolució de les memòries va fer que el temps d'accés de les mateixes decreixessin lentament, mentre que la velocitat dels microprocessadors augmentava exponencialment. Per tant, l'accés a memòria representava el coll d'ampolla. La idea del "caché" és tenir una memòria relativament petita amb la velocitat del microprocessador. La majoria del codi que s'executa ho fa dins de cicles, amb el que, si es té el cicle complet dins del "caché", no seria necessari accedir a la memòria externa. Amb les dades passa una cosa similar: també ocorre un efecte de localitat. El "caché" es carrega

ràpidament mitjançant un procés conegut com a "*ràfega*", amb el qual es poden transferir quatre *Bytes* per cicle de rellotge.

- **Interfície amb el bus:** Inclou els manejadors del *bus d'adreces*, *bus de dades* de 32 *bits* i *bus de control*.
- **Unitat d'instruccions:** Inclou la *unitat de "pre-recerca"* que li demana els *Bytes* d'instruccions al "*caché*" (ambdós es comuniquen mitjançant un *bus intern* de 128 *bits*), una *cua d'instruccions* de 32 *Bytes*, la *unitat de descodificació*, la *unitat de control*, i la *ROM* de control (que indica el que han de fer les instruccions).
- **Unitat de punt flotant:** Inclou vuit *registres de punt flotant* de 80 *bits* i la lògica necessària per realitzar operacions bàsiques, arrels quadrades i operacions matemàtiques complexes de *punt flotant*. És tres o quatre vegades més ràpida que un 386DX i 387DX a la mateixa freqüència de rellotge (veure [comparativa de rendiment](#)).

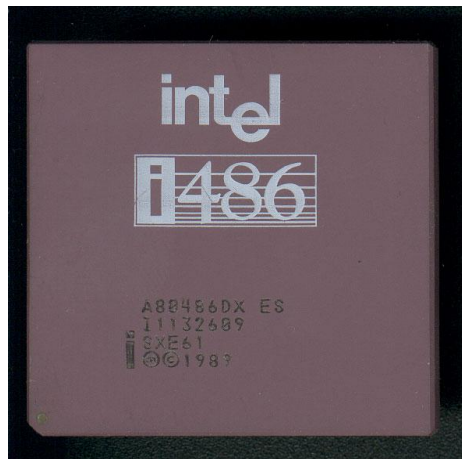


Figura 18. Imatge d'una *CPU Intel 80486 DX*.

Intel va comercialitzar diferents versions del microprocessador 80486 *DX*:

- **80486 DX2:** Al març de 1992 va aparèixer el 80486 *DX2*, que posseeix un duplicador de freqüència intern, amb el que les diferents funcions a l'interior del xip s'executen al doble de velocitat, mantenint constant el temps d'accés a memòria. Això permet gairebé duplicar el rendiment del microprocessador, ja que la majoria de les instruccions que han d'accedir a memòria en realitat accedeixen al *caché intern* de 8 *KB* del xip.
- **80486 SL:** Es va comercialitzar en l'any 1.992, amb característiques especials d'estalvi d'energia estava destinat a equips portàtils.

- **80486 DX4:** Seguint amb la filosofia del *DX2*, en 1994 va aparèixer el 80486 *DX4*, que triplica la freqüència de rellotge i augmenta la grandària del “caché” intern a 16 *KB*.

El xip s'empaqueta en el format *PGA* (*Pin Grid Array*) de 168 pins en totes les versions, encara que els *DXn* també utilitzen el format *PQFP* (*Plastic Quad Flat Package*) de 208 pins (veure la [taula de sòcols](#)).

Les freqüències més utilitzades en aquests microprocessadors són: *DX*: 33 i 50 *MHz*, *DX2*: 25/50 *MHz* i 33/66 *MHz* i *DX4*: 25/75 i 33/100 *MHz*. En els dos últims models, la primera xifra indica la freqüència del *bus extern* i la segona la del *bus intern*. Per tenir una idea de la velocitat, el 80486 *DX2* de 66 *MHz* executa 54 milions d'instruccions per segon.

8.5.2. La variant econòmica: 80486 SX

El microprocessador 80486 *SX*, va ser introduït per Intel a principis de l'any 1.991. Es tractava d'un i486 *DX* sense la *unitat de punt flotant*, per tant, el nombre de transistors de la versió *SX* era menor que el de la versió *DX*, van passar de 1.200.000 a 1.185.000. (Veure [L'Intel 80407 SX. La gran “estafa”?](#)).



Figura 19. Detall d'un microprocessador *Intel 80486 SX*.

8.5.3. Els OverDrive

El xip *OverDrive* va sortir al mercat a l'any 1.992. Bàsicament es tractava d'un microprocessador i486 *DX* de la diversificació que va elaborar *Intel*, amb una curiosa característica de construcció. Simplement estava dissenyat per ser inserit en el sòcol destinat al i487 *SX* i, a més, posseïa un duplicador de rellotge causant l'efecte d'estar treballant amb el i486 *DX2* a 25 *MHz*, i amb algunes aplicacions el doble de ràpid.

És interessant esmentar aquest cas perquè gràcies a com *Intel* va col·locar els seus diferents xips al mercat, va crear una necessitat pels usuaris del processador 80486 *DX* que es va reflectir en un augment de velocitat en els xips de la família i486. *Intel* va saber aprofitar aquest fet i no va canviar el seu model de negoci. No va desenvolupar cap nou xip amb tecnologia avançada i si una boníssima oportunitat de negoci. (Veure [L'Intel 80407 SX. La gran "estafa"?](#)).

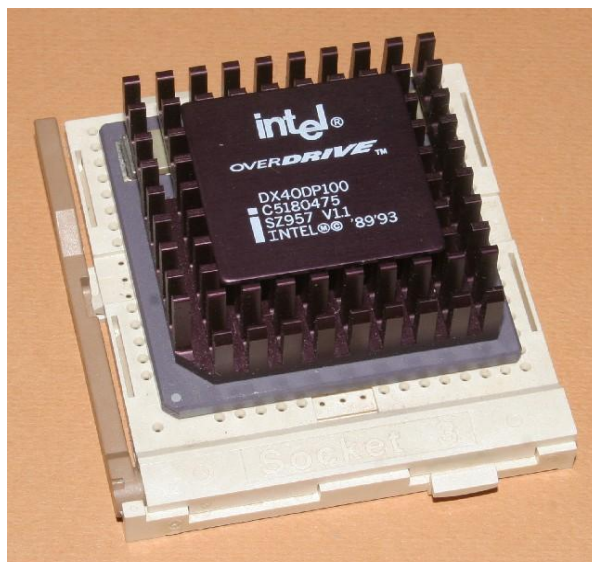


Figura 20. Detall d'un OverDrive i486 *DX4* a 100 *MHz*.

Al següent punt s'introdueix l'*arquitectura x87*, tractant tots els coprocessadors matemàtics que es van dissenyar per acompanyar als processadors que s'han estudiat en aquest darrer capítol. També es descriuen xips alternatius a l'Intel 80387

Després de la introducció de l'*arquitectura x87* i els tipus de coprocessadors que es van dissenyar, es descriuen tots els xips que van aparèixer des de el primer 8087 fins a la seva desaparició, com element independent, amb la incorporació d'una unitat de *punt flotant* dins de l'encapsulat de l'*Intel 80486 DX*.

9. La família x87

Es coneix com família o arquitectura x87 al conjunt d'instruccions matemàtiques dels processadors x86 d'*Intel*. Aquest conjunt d'instruccions, originalment, s'executaven en un xip dedicat diferent del processador principal: el coprocessador matemàtic. El coprocessador o *FPU (Floating Point Unit)* és un component *hardware* especialitzat en la realització d'operacions matemàtiques en punt flotant, pot executar instruccions molt complexes en pocs cicles de rellotge.

9.1. Introducció

A mitjans dels anys 70 no existien coprocessadors matemàtics. Tots els ordinadors eren mainframes o miniordinadors, i tots ells realitzaven operacions en *punt flotant* i obtenien resultats diferents. Això vol dir que quan es calculaven nombres irracionals i s'arrodonia el resultat, les últimes posicions decimals variaven depenent de la marca i model de computador que realitzés el càlcul.

El problema era que els nombre irracionals tenen un nombre infinit de decimals i les memòries de les computadores són finites. Així doncs les respostes irracionals només podien ser aproximacions, i els diferents computadores feien servir mètodes d'aproximació i d'arrodoniment distints. Així, el *IEEE (Institute of Electrical and Electronics Engineers)* va formar un comitè per desenvolupar estàndards per els càlculs en punt flotant.

A continuació s'introduirà el concepte de coprocessador matemàtic i s'explicarà el seu funcionament. Més endavant es veuran tots els coprocessadors matemàtics de la família *Intel* i la seva evolució històrica i tecnològica. Conclourem aquesta part de l'estudi comparant els xips d'*Intel* amb coprocessadors matemàtics alternatius d'altres fabricants.

Fonaments dels coprocessadors

El concepte de coprocessador matemàtic és molt senzill d'entendre. Un coprocessador és, simplement, quelcom que treballa en cooperació amb el microprocessador d'un ordinador personal. L'objectiu és proporcionar una millora del rendiment mitjançant la especialització i la divisió de les tasques i, al mateix temps, alliberar al xip principal d'una part de la seva càrrega.

Els coprocessadors estan dissenyats per a ocupar-se de totes les funcions complexes: divisió d'enters, funcions trigonomètriques, arrels i logaritmes. Aquestes operacions produeixen nombres en *punt flotant*, el tipus de dada que els coprocessadors matemàtics manegen a la perfecció.

Per realitzar operacions matemàtiques complexes amb nombres en *punt flotant*, el coprocessador funciona d'una forma molt semblant a un microprocessador de propòsit general. Fa servir la lògica digital, processa trames de *bits* que contenen informació, els nombres en *punt flotant*, sota el control d'altres trames de *bits* que constitueixen les instruccions. Aquestes operacions es realitzen en registres interns del coprocessador.

A l'hora de calcular una operació matemàtica, el coprocessador carrega primer un dels operands en un dels seus registres i, després, càrrega el segon operant en un altre registre. A continuació llegeix la instrucció de programa que l'indica al xip la operació en particular que es deu calcular. La instrucció inicia una seqüència de microinstruccions que s'executa a l'interior del coprocessador.

Aquest microprograma obliga a la circuiteria del xip a calcular la resposta desitjada. El grup de programes interns del coprocessador que responen a les instruccions que el xip pot executar, és el que s'anomena: [microcodi](#). Un cop que s'ha calculat un resultat, per treure la instrucció del coprocessador, és necessària l'execució d'una altra instrucció.

El coprocessador obté el seu avantatge en velocitat utilitzant nombres en *punt flotant* perquè el seu grup de comandes inclou operacions matemàtiques d'alt nivell. També té una circuiteria interna dedicada per portar a termini aquestes instruccions. Per exemple, un microprocessador pot calcular una arrel irracional, però ha d'executar un bucle de senzilles instruccions centenars de vegades per obtenir el resultat i realitzar centenars d'iteracions de matemàtiques enteres. El coprocessador resol el mateix problema amb una sola instrucció.

Arquitectures de coprocessador

Sigui el que sigui el nombre de component o la velocitat, tots els coprocessadors matemàtics d'*Intel* comparteixen els mateixos elements arquitectònics. El disseny *Intel* converteix a un coprocessador matemàtic en una peça de *hardware* individual que és lògicament una part integrant del microprocessador. L'únic canvi que suposa la incorporació d'un coprocessador és l'ampliació del repertori d'instruccions que entén el processador.

L'enfocament d'*Intel* representa un dels dos mètodes utilitzats per connectar un coprocessador a l'interior d'un *PC*. El primer mètode el constitueixen els xips que es connecten amb el processador mitjançant una connexió directa de port d'entrada i sortida a través dels quals s'envien i es reben dades i instruccions. És el que s'anomena [Estructura d'E/S](#). El segon mètode utilitza un rang d'adreces de memòria per fer el mateix. Aquest mètode s'anomena [Estructura de Memòria](#) i els xips de *Weitek* són d'aquest tipus.

Tots els coprocessadors compatibles amb Intel, incloent el *Cyrix 83D87* i l'*III 3C87* són d'estructura de E/S. El cas del *Cyrix EMC87* té característiques d'ambdós tipus de coprocessadors. Funciona com un xip d'estructura d'E/S per compatibilitat amb el programari escrit per l'*Intel*, malgrat pot accelerar-se a velocitat d'estructura de Memòria amb vistes a futures aplicacions escrites per aprofitar aquestes funcions especials.

Arquitectura d'estructura d'E/S d'*Intel*

En el disseny *Intel*, el microprocessador i el coprocessador estan connectats a les línies de dades que porten la informació a l'interior del *PC*. Normalment, el microprocessador s'encarrega de totes les instruccions en la gran majoria de programes d'ordinador. Malgrat això, el coprocessador matemàtic reconeix determinades instruccions per si mateix i pot executar-les directament.

En cert sentit, el coprocessador d'estructura d'E/S és un paràsit que no pot viure sense el microprocessador. Només el processador principal té circuiteria per controlar les línies d'adreçament del *PC* per trobar la informació. En conseqüència, l'adequat funcionament del coprocessador requereix d'una acurada coordinació de la seva feina amb la del xip principal.

Els esforços d'ambdós xips es dirigeixen a un enllaç *hardware* directe controlat elèctricament mitjançant ports d'entrada/sortida. Aquests ports són interns en els dos xips i, a diferència dels ports d'E/S utilitzats pels perifèrics del computador, l'usuari no pot accedir a ells.

Tant el microprocessador com el coprocessador disposen dels seus propis *registres*, en els quals tenen lloc els càlculs, i de circuiteria de control interna. Com a resultat de tot això, ambdós xips poden funcionar, en certa manera, independentment. És a dir, mentre el coprocessador s'esforça per resoldre un problema particularment difícil, el microprocessador pot fer quelcom més.

Els coprocessadors d'estil *Intel* funcionen amb registres de 80 *bits*. Aquesta mida és exactament la adequada per albergar 64 *bits* de dades significatives amb 15 *bits* més per albergar l'exponent, i un *bit* addicional pel signe del nombre. No obstant, els registres dels xips d'*Intel* no estan limitats a aquest únic format de dades. Poden calcular amb nombres en punt flotant de 32, 64 ó 80 *bits* i *BCD* de 18 dígitos.

Tots els xips d'*Intel* tenen vuit d'aquests *registres* per realitzar els seus càlculs. Les instruccions dels seus programes ja li indiquen el format dels números amb els que deu treballar i com fer-ho. L'única diferència real és la manera de traslladar al microprocessador els resultats dels càlculs. Els processadors d'*Intel* manipulen els registres en blocs d'un *Byte*, en comptes d'utilitzar 80 *bits* com fan els coprocessadors.

Els vuit registres de 80 bits també difereixen dels d'un microprocessador en la forma en la qual son adreçats. Les comandes pels registres individuals d'un processador s'encaminen directament al registre apropiat. En canvi, els registres d'un

coprocessador estan ordenats en una *pila*. A cada valor que s'introdueix a la *pila*, el nombre antic baixa un nivell.

Teòricament, aquest disseny d'estructura d'E/S pot afegir un grau de processament en paral·lel al computador. Però, molt sovint no ho fa. La majoria de programes envien dades al coprocessador matemàtic en busca d'uns resultats i, mentrestant, deixen al microprocessador principal en estat d'espera.

Arquitectura d'estructura de Memòria

Els coprocessadors d'estructura de Memòria, com la línia Abacus de *Weitek*, es comuniquen amb els seus programes i microprocessadors fent servir les adreces de memòria com bústies.

Un petit rang d'adreces, normalment una pàgina de 4 *KB*, en paràgrafs remots de la *RAM* del sistema, per sobre dels 16 *MB* que la majoria de computadores basats en el 80386 poden utilitzar per la *RAM* física, però dins del rang d'adreçament de 4 *GB* del microprocessador, està acordonat per aquestes comunicacions. El processador envia instruccions a un grup d'adreces i les dades a un altre grup.

El coprocessador respon amb els seus resultats de la mateixa forma. En aquestes posicions no hi ha instal·lats xips de memòria *RAM*. Al seu lloc, la memòria necessària per poder acomodar les comandes i les dades, formen part de la circuiteria interna del coprocessador matemàtic.

Un requisit obvi del disseny d'estructura de Memòria és que el coprocessador ha de tenir accés a les línies d'adreçament utilitzades pel microprocessador. Els coprocessadors d'estructura d'E/S no tenen necessitat d'aquesta informació d'adreçament, per això les línies d'adreçament no estan disponibles en els sòcols de coprocessador dissenyats per xips 80387.

Per això, els coprocessadors d'estructura de Memòria requereixen grans sòcols amb més pins per albergar totes les línies d'adreçament que necessiten. Així doncs, aquests tipus de coprocessadors disposen d'uns sòcols de la mida dels empleats pels microprocessadors 80386, mentre que els sòcols dels 80387, i compatibles, són molt més petits.

Aquests sòcols especials per coprocessadors d'estructura de Memòria s'anomenen *EMC* (*Extended Math Coprocessor interface*: interfície de coprocessador matemàtic ampliat). Teòricament, un xip d'aquesta estructura pot ser més ràpid que un xip d'estructura d'E/S perquè l'intercanvi de comandes i dades a través de la memòria *RAM* és molt més ràpid que a través de la ruta d'E/S.

Mentre que els xips d'estructura d'E/S mouen instruccions i dades en operacions individuals durant varis cicles de rellotge, els xips d'estructura de Memòria poden adquirir totes les dades i instruccions que necessiten en només una operació. Un cop

carregada la informació al seu rang de memòria, el coprocessador té tot el que necessita.

El gran inconvenient dels coprocessadors d'estructura de Memòria és que la seva interfície encara no s'ha estandarditzat. Cada coprocessador disposa de les seves pròpies comandes i utilitza el seu propi rang d'adreçament inequívoc. A fi d'aprofitar el coprocessador, els programes deuen conèixer aquests secrets. Per això, cada xip d'estructura de Memòria requereix la seva pròpia versió d'un determinat programa.

9.2. L'Intel 8087

A mitjans dels anys 70 el *IEEE* va formar un comitè per desenvolupar estàndards per a càlculs en *punt flotant*. Al mateix temps, *Intel* estava desenvolupant el seu microprocessador 8086 i va decidir que desenvoluparia una implementació de maquinari del estàndard *IEEE* com a part del programa del seu nou processador.

Per descomptat, en aquell moment ningú considerava els ordinadors personals com un producte massa interessant, però *Intel* va considerar les possibilitats de la *coma flotant* per a la robòtica i les aplicacions de control numèric. Però, en aquell moment, la majoria d'aplicacions no necessitaven un processador en *punt flotant* que seguís l'evolució estàndard *IEEE*.

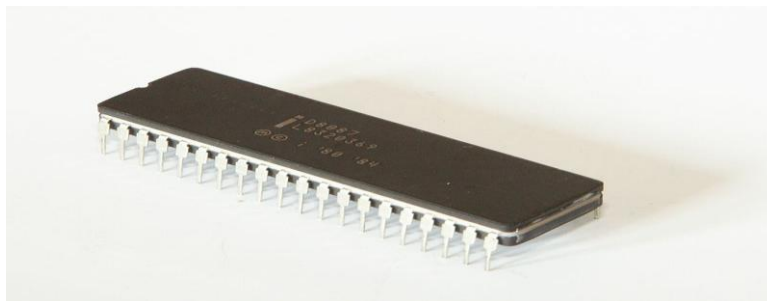


Figura 21. Imatge d'un coprocessador 8087.

Com a conseqüència, les funcions matemàtiques sofisticades no s'inclouien en el 8086, sinó que quedaven relegades a un element apart, que finalment es va produir a nivell comercial l'any 1.980. El **8087** [\[37\]](#) es va retardar respecte al microprocessador que anava a acompanyar per una bona raó: era molt difícil de dissenyar. Segons *Intel*, era el circuit integrat de gran escala més complex que havien produït.

Encara així, el 8087 no podia realitzar les seves operacions matemàtiques per si mateix. Havia de treballar en cooperació amb el 8086. Ambdós funcionen com un sol processador, un que entén 68 noves instruccions en codi màquina per a operacions en *punt flotant* que el 8086. Aquest coprocessador pot treballar també amb la resta de xips derivats del 8086: el 8088, empleat en *PC* i *XT*, a més dels 80186/80188.

El 8087 s'ajusta en un [sòcol DIP de 40 pins](#), proporcionant les mateixes habilitats d'adreçament i maneig de les dades dels microprocessadors que anava a igualar. Encara que pot acceptar dades d'un bus de 16 *bits*, també funcionarà sense modificacions amb el bus de dades de 8 *bits* del 8088.

El 8087 comparteix normalment la mateixa freqüència de rellotge amb el seu company microprocessador, i funciona a la mateixa velocitat. *Intel* segueix oferint chips 8087 en tres models de diferents velocitats: un a 5* *MHz*, un altre a 8 *MHz* i un tercer a 10 *MHz*. També va aparèixer un derivat, el 80C187, a 16 *MHz*. Aquest coprocessador matemàtic estava destinat a usos industrials.

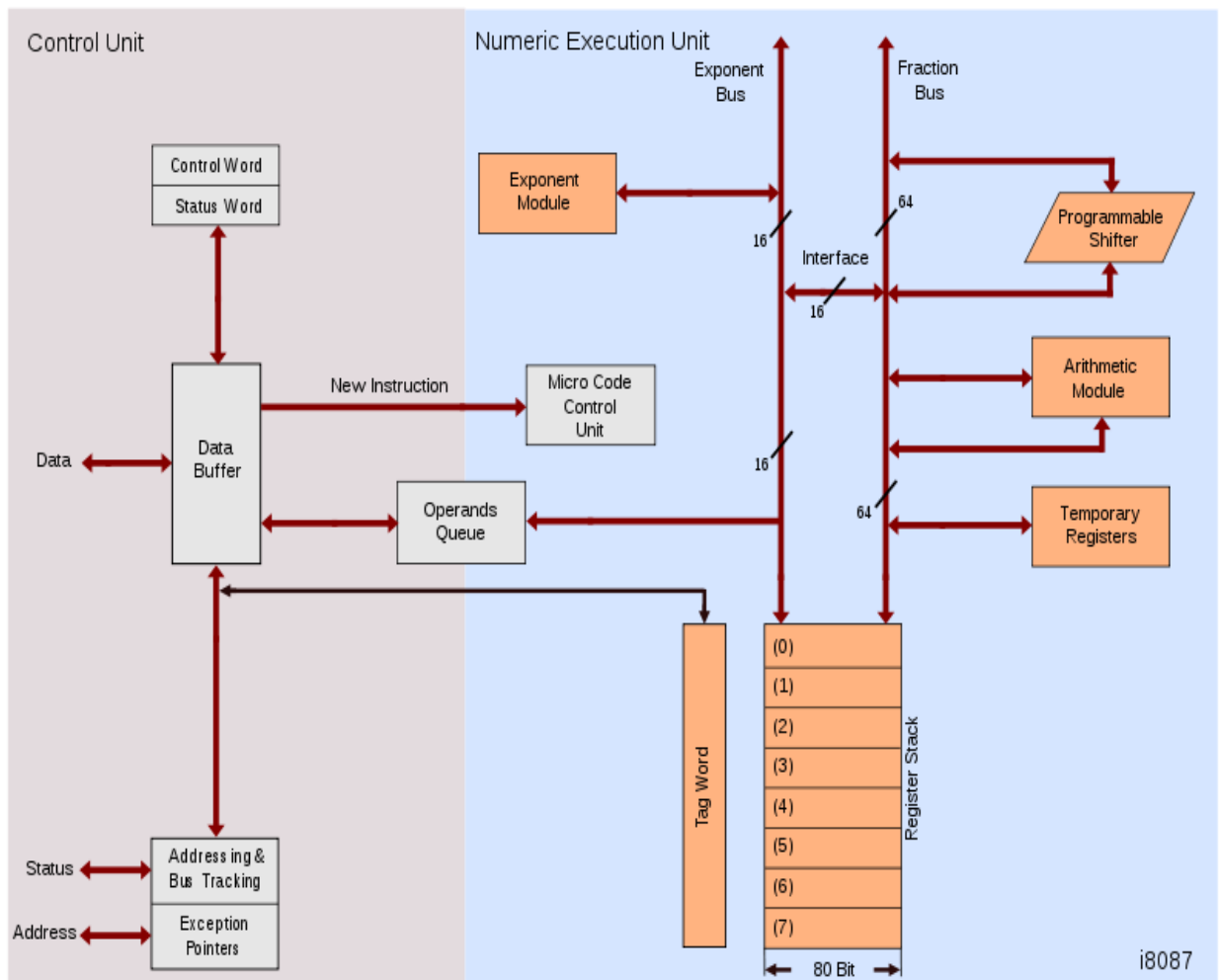


Figura 22. Arquitectura de l'Intel 8087.

Com anècdota, comentar que encara que el cost de fabricació del 8087 rondava els 40\$, el preu de venda va superar els 700\$ (!) en el moment de la seva presentació.

(*) Realment són 4,77 *MHz* però per qüestions de màrqueting, *Intel* i el seu gran aliat, *IBM*, van decidir adoptar la xifra 5 com a reclam publicitari.

9.3. El coprocessador *Intel 80287*

En el disseny inicial del 8087, Intel va tenir la previsió de dividir la circuiteria del xip en dos elements funcionals: una unitat d'interfície de bus i una unitat de punt flotant. El primer element enllaça el coprocessador amb la resta del sistema en el qual està instal·lat (bàsicament amb el microprocessador). El segon element s'encarrega de l'execució dels càlculs.

Com a resultat de tot això, actualitzar les capacitats matemàtiques del 8087 a un altre entorn només requereix la modificació de la interfície del bus, en comptes de tornar a dissenyar un xip nou. El 80287 [38], introduït l'any 1.984, s'aprofita d'aquesta estratègia. Essencialment, aquest coprocessador és la secció de *punt flotant* del 8087 lligada a la nova lògica d'interfície per formar parella amb el microprocessador 80286.

Aquest disseny, idèntic al del 8087, confia en el microprocessador principal per a suport del sistema. El 80287 ni tan sols va tenir accés a les *línies d'adreçament* del computador i, per tant, totes les operacions associades a la memòria queden fora del seu control. Aquest disseny permet al i287 tractar amb els *modes real* i *protegit* del 80286, i li permet adreçar tot el rang de 16 MB del microprocessador. El 8087, en canvi, només funciona en *mode real* (veure [Modes de funcionament](#)).

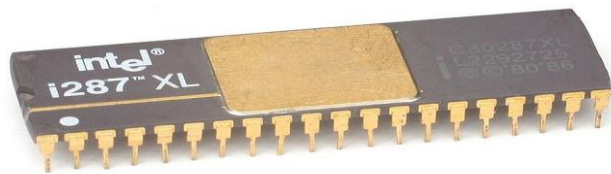


Figura 23. Detall del coprocessador *Intel 80287 XL*

El coprocessador 80287, igual que el 8087, es col·loca en un sòcol *DIP* de 40 pins, però no és compatible pin a pin amb el xip anterior. A diferència del 8087, l'*Intel 80287* està dissenyat per funcionar de forma asíncrona. És a dir, el coprocessador no funciona necessàriament a la mateixa velocitat que la *CPU* principal. Els dos xips saben com ajustar les seves operacions, esperant el temps necessari, per ajustar els seus cicles de transferència de dades.

Normalment, el 80287 es connecta al mateix oscil·lador que fa funcionar la resta d'un *PC*. No obstant, un divisor intern redueix la freqüència de rellotge del coprocessador a un terç de la seva velocitat inicial abans d'arribar a la circuiteria de *punt flotant*. Així doncs, un 80287 funciona realment a una tercera part de la velocitat de rellotge que se li presenta.

En la majoria dels sistemes basats en el 80286, el rellotge que fa funcionar el microprocessador es divideix a la meitat abans de ser connectat al 80286. La velocitat de rellotge doble inicial es connecta al 80287, de forma que el coprocessador funciona eficaçment a dos terços de la velocitat de la *CPU*. Per exemple, en un ordinador *IBM AT* a 8 MHz, el xip 80287 funciona a 5,33 MHz.

Alguns sistemes li proporcionen al 80287 un rellotge dedicat propi, permetent a l'enginyer dissenyar el sistema per poder funcionar a qualsevol velocitat que desitgi. Utilitzar un rellotge dedicat pot augmentar el rendiment de les dades del coprocessador 80287 enormement. Malgrat *Intel* va arribar a oferir quatre velocitats diferents, aviat va reduir el nombre a només dues.

Com el 80287 està basat principalment en la circuiteria 8087 de *punt flotant*, el xip és quasi complement compatible amb el 8087 i executarà la majoria del programari que aquest executa (malgrat no tot perquè no és una còpia "porta per porta", si no una actualització millorada). La principal diferència entre els dos xips és la forma de tractar els errors.

El disseny lògic de [bus de control](#) d'Intel en el 80287, fa que el coprocessador confii en la *CPU* amfitriona per adreçar informació, significa que el i287 no està limitat per la capacitat de maneig de la memòria del 80286. El seu disseny versàtil permet al 80287 funcionar amb microprocessadors 80386 i, fins i tot, al llarg de dos anys va ser el coprocessador oficial del món empresarial.

Però el 80287 presenta inconvenients en l'entorn 80386. El i287 es va dissenyar amb una interfície de 16 bits, mentre el i386 és capaç de realitzar operacions de 32 bits. A banda d'això, el 80287 havia quedat retardat degut a l'evolució de l'estàndard en *punt flotant* del *IEEE*. Després de començar la producció del 80287, es va donar la forma a l'estàndard definitiu: ***l'ANSI-IEEE 754-1985*** [\[39\]](#).

Com a conseqüència, el 80287 no és el coprocessador matemàtic ideal, particularment per un computador basat en 80386. No obstant, un lent i287 és encara molt més ràpid en operacions de *punt flotant* que un sistema i386 sense cap tipus de coprocessador.

9.4. L'Intel 80387

Quan Intel va començar a dissenyar un coprocessador per al 80386, va decidir desenvolupar una nova unitat *d'interfície de bus* i una nova unitat de *punt flotant*.

El nou disseny de la unitat de *punt flotant* va fer quelcom més que implementar el nou estàndard *IEEE*. És cinc vegades més ràpida que l'antiga 80387/80287. Des de la seva introducció l'any 1.987 com a part integrant del coprocessador 80387 [40], la nova unitat de *punt flotant* s'ha convertit en el fonament de tots els coprocessadors matemàtics posteriors.

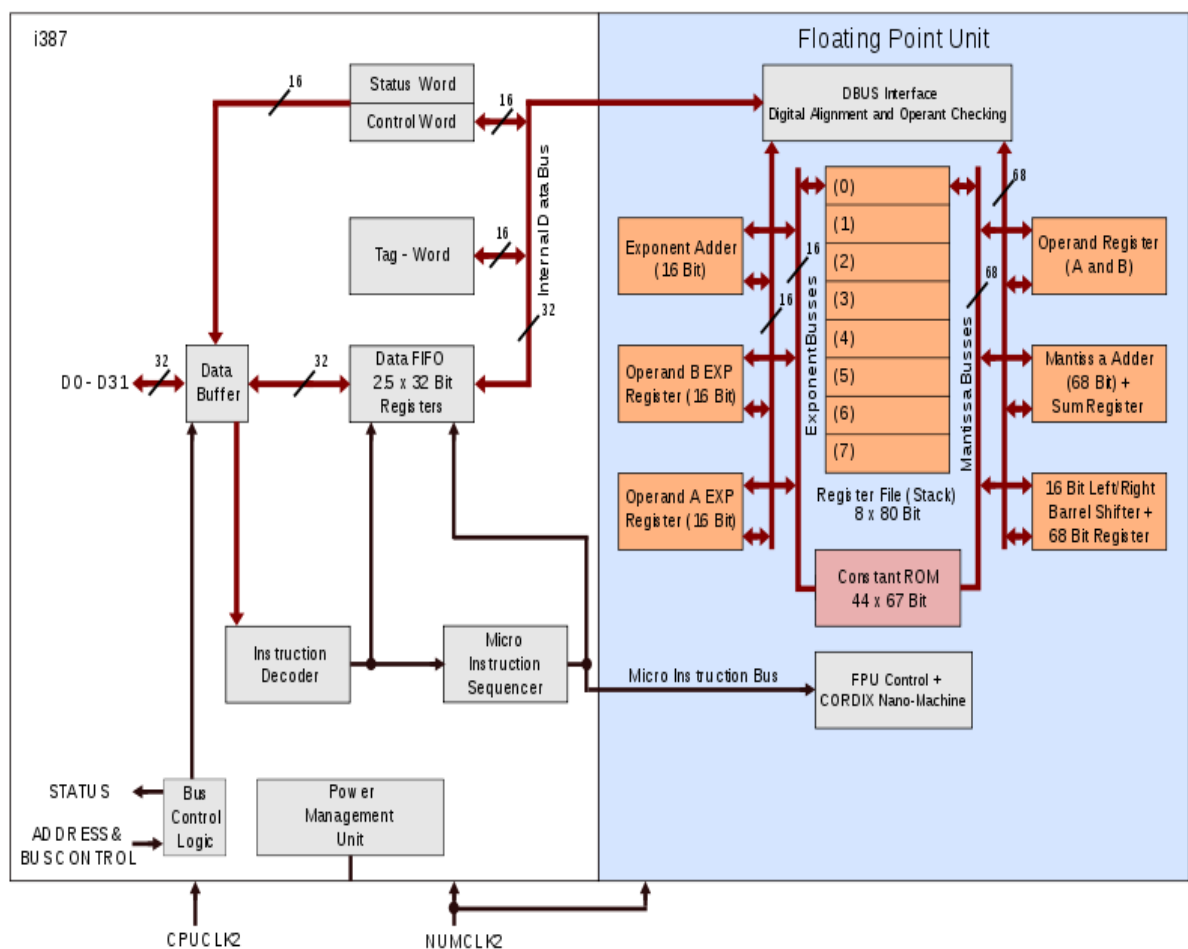


Figura 24. Arquitectura de l'Intel 80387 (DX).

També serveix com a base del coprocessador 80387 SX [41] i les noves revisions del i287: 80287 XL i 80287 XLT. El i387 SX és essencialment el mateix xip que el i387 però dissenyat per funcionar amb el bus de 16 bits del 80386 SX en comptes de fer-ho amb un bus de dades complet de 32 bits. Bona part del disseny del i387 es va introduir en la secció de *punt flotant* del microprocessador 80486.

El 80387 promet un cert grau de compatibilitat amb el i287, similar a la d'aquest últim amb el 8087. Les principals diferències apareixen de nou en el tractament d'errors, degut principalment a modificacions en l'estàndard *IEEE*. Aquestes diferències es gestionen adequadament si el programari ha estat codificat correctament.

Un altre canvi que va introduir *Intel* al modificar les unitats de punt flotant del 80387 va ser la incorporació de un rang més ampli de funcions transcendents, incloent funcions de sinus, cosinus, tangent, arc tangent i logarítmiques. Com a conseqüència d'això el i387 ha de ser capaç d'executar tots els programes escrits per al 80287, encara que l'invers no és necessàriament cert.

Malgrat l'*Intel* 80387 pot funcionar de manera asíncrona, un i387 funciona a la mateixa velocitat que el 80386 amb el qual està instal·lat. Es poden trobar coprocessadors matemàtics 80387 disponibles en totes les variants de velocitat del 80386 fins als 33 *MHz*.

9.4.1. El coprocessador 80387 (DX)

El coprocessador matemàtic i387 s'assembla molt al microprocessador 80386, només és una mica més petit. La seva carcassa [PGC \(Pin Grid Array\)](#) de 68 pins quadrada té el mateix aspecte de pissarra que el microprocessador. La mitjana de velocitat del xip ve donada en *Megahertz*s després del nombre de component. Així doncs, un 80387-20 funciona a 20 *MHz*.

El disseny de l'Intel 80387 no ha estat estàtic. Quan es va fer necessari impulsar el i387 a 33 *MHz*, també es van necessitar majors millores en el disseny. Intel va passar de la tecnologia *NMOS* a la *CMOS* i va utilitzar nous processos de fabricació que permetien gravar detalls tan fins com un *micró* en el silici del xip (els primers 80387 es limitaven a detalls de 1,5 *microns*).



Figura 25. Imatge de l'Intel 80387 DX.

Aquestes millores, a més d'alguns "remiendos" en la mateixa unitat de *punt flotant*, van produir una millora de rendiment d'un 20 per cent. L'Intel 80387 a 33 *MHz*, comercialitzat l'abril de 1.989, incorporava totes aquestes funcions. L'1 d'octubre de 1.990, les versions a 16, 20 i 25 *MHz* del i387 es van actualitzar a la nova tecnologia.

Es poden distingir els 80387 amb l'anterior tecnologia dels xips actualitzats pel codi numèric situat sota el número de component. Els antics i387 sempre inicien aquesta línia de 10 xifres amb la lletra **S**. Els xips de nova tecnologia no incorporen aquesta lletra.

9.4.2. L'Intel 80387 SX. Destinat a equips econòmics

El complement de coprocessador matemàtic 80387 SX pel microprocessador i836 SX es va presentar al gener de 1.990, produït també amb la nova tecnologia CMOS. Només estan disponibles dues versions del 80387 SX: un xip treballant a 16 MHz i un altre que ho fa a 20 MHz.



Figura 26. Detall del coprocessador Intel 80387 SX.

A banda de les freqüències de treball més baixes, l'única diferència existent entre la variant DX i la variant SX del coprocessador 80387 és que aquesta última està preparada per treballar amb un bus de dades de 16 bits.

Malgrat el tàndem i386 SX - i387 SX, estava destinat a petites empreses i al món acadèmic, Intel va continuar inferint nous coprocessadors 80287 amb tecnologia CMOS. Potser per por de perdre quota de mercat, va decidir actualitzar els seus xips 287. Va comercialitzar els nous 287 XL i 287 XLT, ambdós xips eren molt més ràpids i menors consumidors d'energia que els anteriors versions NMOS.

9.4.3. Alternatives a *Intel*

Encara que *Intel* va autoritzar a altres fabricants de xips produir els seus dissenys de microprocessador i coprocessador per a les famílies 80286 i 80287, l'empresa es va negar fermament a autoritzar el 80386 i 80387. Aquest xips, més recents en aparició i amb major potència, han produït uns immensos beneficis per *Intel*, i els empresaris han posat els seus ulls amb enveja en tals beneficis.

Dos grups, *Cyrix* i *IIT*, estan tractant de compartir aquesta abundància creant els seus propis coprocessadors compatibles.

9.4.3.1. *Cyrix* 83D87

Cyrix va ser fundada l'any 1.988 per dissenyar i comercialitzar components. L'empresa va decidir desenvolupar coprocessadors matemàtics compatibles amb *Intel* com a primer producte, a causa que era un mercat obert en una zona sense competidors significatius. La seva primera sèrie de coprocessadors *FasMath* es va comercialitzar al mes d'octubre de 1.989.

Anomenats 83D87, són substitucions compatibles pin a pin amb l'*Intel* 80387. Una altra versió de baix cost destinada a ordinadors 80386 *SX*, anomenada 83S87, es va introduir al mes de març de 1.990.

Els productes *Cyrix* estan dissenyats per ser totalment compatibles amb la família i387, encara que no es copien mitjançant mètodes tradicionals (passar un xip per rajos x i determinar la seva estructura interna). En el seu lloc, *Cyrix* ha inclòs en els seus coprocessadors un disseny lògic totalment diferent basat en les funcions documentades i no documentades dels productes d'*Intel*.

Potser la diferència més important és que els xips *Cyrix* confien més en la lògica de maquinari que en el *microcodi*. A partir d'aquest disseny alternatiu, poden aconseguir una velocitat essencialment superior que els xips d'*Intel* en operacions en *punt flotant*. Amb aquesta lògica amb maquinari, els models de *bits* que configuren les comandes provoquen directament canvis d'estat en la circuiteria.

Cada model (cada instrucció lògica) s'ha de dissenyar específicament en el *hardware* del coprocessador. En el disseny de *microcodi*, les instruccions enviades al microprocessador fan que el xip porti a terme els distints passos que conformen un programa intern en miniatura. Aquest mini programa li indica a la lògica de propòsit més general del xip que porti a terme la funció requerida.

El disseny de *microcodi* és l'enfocament més estructurat. Proporciona al dissenyador una major flexibilitat i li pot ajudar a comercialitzar productes al mercat amb major rapidesa. També permet que els *circuits de propòsit general* manegin complexos grups d'instruccions. Però el microcodi pot ralentir el procés de pensament del xip. A més, executar el [microcodi](#) imposa més càrrega a cada càlcul.



Figura 27. Imatge del Cyril 83D87.

Les proves de *PC Labs* [\[42\]](#) van mostrar que l'enfocament de disseny emprat per Cyril té un pronunciat efecte sobre el rendiment en *punt flotant*. En les [proves de laboratori](#) el Cyril 83D87 va demostrar ser gairebé dues vegades més ràpid que el seu equivalent *Intel*. No obstant això, el rendiment en programari d'aplicacions comercials va resultar menys sorprenent.

Fins i tot en les tasques en les quals eren necessaris més càlculs, el Cyril 83D87 només va ser capaç de superar a l'*Intel* 80387 en aproximadament un 10 per cent (si no gens en absolut). Amb aplicacions estàndard, el rendiment per si sol no és una raó per triar el 83D87. No obstant això, el xip de Cyril mai va empitjorar respecte al i387, per tant un preu baix de sortida podria convertir al Cyril en una opció a considerar.

No obstant això, els preus que van fixar ambdues empreses estan bastant igualats. El 83S87 està disponible en dues versions, una a 16 MHz i una altra a 20 MHz, i el 83D87 es comercialitza en quatre: xips a 16, 20, 25 i 33 MHz.

9.4.3.2. *Cyrix EMC87*

Cyrix espera consolidar-se com una mica més que un simple fabricant de xips clònics. L'empresa ha tractat d'evitar les limitacions inherents dels coprocessadors d'estructura d'E/S. El nou EMC87 es va desenvolupar com un substitut [d'estructura de Memòria](#) per l'*Intel* 80387.

El coprocessador EMC87 està basat en la mateixa arquitectura de processador que l'*Intel* i387, amb vuit registres de 80 bits, i disposa en essència del mateix grup de comandes. No obstant això, la lògica del seu bus de control ha estat totalment revisada, fins a l'extrem que *Cyrix* afirma haver obtingut una millora cinc vegades superior respecte a *Intel*.

Però apropar-se a tal velocitat requereix programari escrit especialment per l'EMC87. Les aplicacions que suporten l'*Intel* 80387 utilitzen únicament instruccions d'estructura d'E/S. A més, l'EMC87 és incompatible amb codi escrit per als xips de *Weitek* perquè les dues famílies de coprocessadors tenen arquitectures totalment diferents per a les seves unitats en *punt flotant*.



Figura 28. Vista del coprocessador *Cyrix* EMC87.

Com el *Cyrix* EMC87 és d'estructura de Memòria, necessita accedir a totes les línies d'adreçament emprades en els ordinadors 80386. Així doncs, disposa d'un complet complement de 121 pins que s'ajusta en el [sòcol EMC](#) normalment reservat per a un coprocessador *Weitek*.

Comprent la necessitat de programari disponible per recollir els beneficis deixats pel xip, *Cyrix* ofereix un convertidor que adapta codi font en llenguatge assemblador d'instruccions d'estructura d'E/S a instruccions de Memòria per l'EMC87. Encara que el convertidor de codi pot ser interessant per als programadors, no té valor per a un usuari final: No converteix aplicacions comercials per fer-les compatibles.

Sabent que ningú desitja un producte que no funcioni amb cap programari disponible, Cyrix ha creat una solució "intermèdia i infinita" amb l'EMC87. El xip també comprèn instruccions [d'estructura d'E/S](#) d'estil *Intel*, i proporcionarà un equivalent en rendiment al *Cyrix 83D87* en programes escrits per al *i387*.

De fet, les unitats de *punt flotant* a l'interior del *83D87* i del *EMC87* són idèntiques, només difereixen en la lògica del bus de control. El coprocessador *EMC87* ofereix així compatibilitat actual a més de la promesa d'una major acceleració. Fins que (i si) els editors de programari optin per treure partit de l'*EMC87*, aquest xip seguirà sent més una curiositat que una millora addicional.

Però com la versió a 33 MHz manté el mateix preu que el *Cyrix 83D87* a la mateixa velocitat i ofereix una promesa de futur rendiment sense cap cost addicional, per a qualsevol que disposi d'una computadora amb el sòcol *EMC* necessari, l'*EMC87* podria ser una interessant inversió.

	Instruction	FasMath™ EMC87		Intel 80387
		EMC Interface	D87 Interface	
IEEE Load/Store: - Identical Results Compared With 80387. - Faster Instruction Execution. - All Status, Exception, Control, Tag & Round Bits Are Identical. - Verified Against U.S. Berkeley IEEE Conformance Tests	FLD Reg	4	11	14
	FLD DP	12	20	25
	FLD SP	10	17	20
	FILD Int 16	7	21	61-65
	FILD Int 32	7	17	45-52
	FST Reg	4	11	11
	FST DP	11	30	45
	FST SP	9	26	44
	FIST Int 16	11	25	82-95
	FIST Int 32	11	26	79-93
IEEE Arithmetic: - Dedicated Parallel Adder and Hardware Array Multiplier. - Proprietary High Speed Divide & Square Root Algorithms. - Identical Results Compared With 80387. - Faster Instruction Execution. - All Status, Exception, Control, Tag & Round Bits Are Identical. - Verified Against U.S. Berkeley IEEE Conformance Tests	FABS	4	11	22
	Reg	6	15	26-34
	FADD DP	15	25	29-37
	SP	13	21	24-32
	FCHS	4	11	24-25
	Reg	4	11	24
	FCOM DP	13	20	31
	SP	11	17	26
	Reg	14-24	31	91
	FDIV DP	21-32	30	94
	SP	20-31	27	89
	Reg	10	19	52
	FMUL DP	18	25	57
SP	16	21	35	
FSQRT	26	31	122-129	
Reg	6	15	29-37	
FSUB DP	15	21	28-36	
SP	13	21	24-32	
Processor Control: - Identical Results In All Cases.	FCLEX	4	16	18
	FINIT	5	18	18
	FSTSW	4	10	10
Transcendental Functions: - Greater accuracy than 80387. - Maximum relative error less than $1.5 \cdot 2^{-66}$. - Faster Instruction Execution. - All Status IEEE Conformal.	F2XM1	63	63	211-476
	FCOS	87	87	123-772
	FPATAN	83	83	314-487
	FPTAN	75	75	91-497
	FSIN	63	63	122-771
	FYL2X	87	87	120-538
	FYL2XPI	79	79	257-547

Figura 29. Comparativa de rendiment, dades oficials, de l'EMC87 vs. l'Intel 80387.

9.4.3.3. IIT 3C87

Integrated Information Technology (IIT) és en el fons una empresa de coprocessadors matemàtics. Fundada l'any 1.988 per dos enginyers que van deixar *Intel* per treballar per *Weitek* (de fet un d'ells era un dels cofundadors), l'empresa ofereix xips compatibles amb els coprocessadors 80287 i 80387 d'*Intel*.

Com en el cas dels productes *Cyrix*, els de *IIT* van ser desenvolupats partint de zero en lloc d'utilitzar enginyeria inversa. Ambdós són dissenys *CMOS* basats en tecnologia de 1,2 *microns*. El disseny del coprocessador *IIT 3C87* té 32 registres de 80 *bits* en comptes de tenir només vuit.

Aquests registres estan dividits en quatre grups i estan dissenyats per facilitar els càlculs de matrius 4 x 4, que poden accelerar el rendiment en aplicacions gràfiques. Utilitzar simplement un d'aquests quatre grups simula un coprocessador d'estructura d'E/S, utilitzar tots quatre requereix programes especials codificats per aquest xip. Només l'aplicació **FastCAD 3-D** va aprofitar la instrucció de matrius 4 x 4.



Figura 30. Detall del coprocessador *IIT 3C87*.

IIT afirma que el 3C87 pot calcular un 50 per cent més ràpid que un *Intel 80387*. En [proves de laboratori](#) i programari comercial estàndard, no es van trobar mai cap avantatge significatiu en rendiment. A diferència dels xips de *Cyrix*, el coprocessador 3C87 no duplica exactament el funcionament del 80387. El tractament d'errors dels xips d'*Intel* i d'*IIT* difereix, igual que succeeix amb el i287 i el i387.

Alguns fabricants de xips han explotat aquesta diferència escrivint programes especials que mostraven resultats anòmals quan s'executaven en els coprocessadors d'*IIT*. No obstant això, segons *IIT*, tals estranyes respostes només sorgeixen quan es tracta a propòsit d'exacerbar les diferències en el maneig d'errors dividint per un nombre proper a zero.

Amb programari estàndard no van aparèixer resultats anòmals. De fet, *PC Labs* [42] no va detectar mai respostes estranyes o incompatibilitats en les [proves amb programari](#) d'aplicacions comercials. *IIT* reconeix que una primera aproximació del 3C87 tenia un error intern que donava com a resultat errors al utilitzar l'operació arc tangent quan executava **AutoCAD** sota *Unix*.

Aquest error es va corregir al juny de 1.990, i l'empresa informa que no ha detectat més des de llavors.

9.4.3.4. *Weitek* Abacus 3167

L'empresa *Weitek* es va formar en l'any 1.981 i al 1985 ja estava produint coprocessadors de punt flotant per a una sèrie d'estacions de treball, incloent les basades en els microprocessadors **Motorola 68020** i **SPARC** de *Sun*. En aquell moment, *Intel* va contractar a *Weitek* per desenvolupar un coprocessador que acompanyés al xip 80386.

Segons *Weitek*, el programa *Intel* 387 intern estava en preparació i *Weitek* va desenvolupar el seu producte en paral·lel amb l'equip 387. Aquests esforços van conduir al *Weitek* 1167, el primer coprocessador de la línia Abacus. Es tractava d'una petita placa de circuit que combinava dos dels elements del coprocessador de l'empresa emprats en computadores 68020 a més de la lògica d'interfície per al i386.

A l'abril de 1.988 *Weitek* va introduir el 3167, un equivalent en un sol xip del 1167, utilitzat per millorar els processadors *Intel* 80386. Més endavant, al novembre del 1.989, va ser presentat el 4167, un coprocessador matemàtic dissenyat per millorar l'*Intel* i486 que mantenia la compatibilitat amb l'Abacus 3167.



Figura 31. Aspecte del *Weitek* Abacus 3167.

Per al microprocessador principal, el *Weitek Abacus* apareix com un bloc de memòria de 64 KB que comença a l'adreça física 0C000000h. Cada adreça d'aquest bloc es correspon a una instrucció de coprocessador. L'accés a una ubicació de memòria especificada dins d'aquest bloc amb la instrucció *MOV*, provoca l'execució de la corresponent instrucció *Weitek*.

Les instruccions han estat hàbilment assignades a posicions de memòria de manera que les càrregues a registres de coprocessador consecutius poden fer ús d'una seqüència d'instruccions *MOV* del processador. Aquesta interfície de memòria mapada és molt més ràpida que el protocol orientat a E/S utilitzat per unir la *CPU* principal amb el 80387.

El *Weitek 3167* no pot ser utilitzat en el [mode real](#) del processadors 80386, ja que el seu rang d'adreces de memòria comença a l'adreça 0C000000h i queda fora d'un *MByte*, la quantitat màxima de memòria que pot manejar un processador x86 en *mode real*.

Weitek WTL 1167 architecture

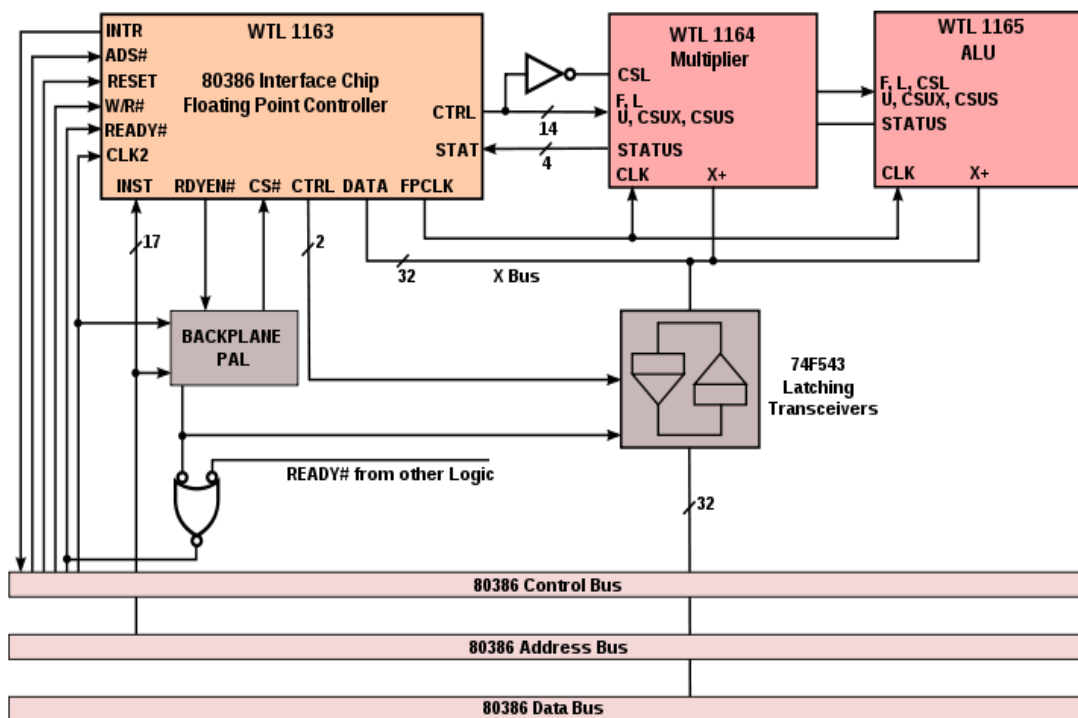


Figura 32. Arquitectura dels coprocessadors matemàtics *Weitek*.

Weitek va comercialitzar només tres models del coprocessador Abacus 3167: una versió a 20 MHz, una a 25 MHz i una tercera a 33 MHz. Del 4167 tan sols existeixen dues versions: el model a 25 MHz i el model a 33 MHz.

Els xips Abacus no funcionaran en programari que suporti l'*Intel* 80387 perquè els coprocessadors matemàtics *Weitek* estan tots fabricats entorn d'una arquitectura totalment diferent de la de la família i387. Tret que el programari amb el qual es treballa suporti explícitament un Abacus, no es detectarà cap millora en la velocitat emprant un (veure la [comparativa de rendiment](#)).

9.5. L'*Intel* 80487 SX. La gran “estafa”?

Quan *Intel* va comercialitzar el microprocessador 80486 SX, ja havia preparat el coprocessador matemàtic necessari per aquests nou xip. Aquest nou producte, anomenat 80487 SX va resultar ser, ni més ni menys, que un 80486 DX perfectament funcional (amb la seva unitat de *punt flotant* interna inclosa dins del mateix processador).

Però la cosa més sorprenent d'aquesta maniobra dels fabricants (doncs també els productors de plaques bases estaven implicats) és el fet de veure com a l'hora d'instal·lar el coprocessador amb aquest “programari sofisticat” era un mecanisme que una vegada col·locat el i487 SX en el seu *sòcol* indicava al processador principal que es desactivés.

Així doncs, el processador principal quedava inhabilitat i a partir d'aquell moment el 80487 SX es feia càrrec de totes les tasques del sistema: tant de les processos generals com de les instruccions en *punt flotant*. Realment l'ordinador actualitzat es convertia en un potent sistema i486 DX. Aquesta maniobra era veritablement més enginyosa que no pas enganyosa.



Figura 33. Detall de l'*Intel* 80487 (SX).

El 80487 *SX* va resultar ser tant bon negoci pels fabricants, que es va voler continuar amb aquesta estratègia amb l'amplia varietat de processadors que hauria de venir en el futur. Després d'aquesta experiència tant rentable, *Intel* va començar a experimentar amb la diversificació de productes prenent un producte realment bo, com era el i486 *DX*, i tornant-ho a empaquetar de diferents maneres.

A banda de l'absència de la unitat de *punt flotant*, el microprocessador 80486 *SX* és un i486 *DX* perfectament funcional, però empaquetat d'una manera diferent. Els pins de connexió d'ambdós xips s'organitzen d'una forma distinta. És físicament impossible instal·lar un i486 *DX* en el sòcol de l'*Intel* 80487 *SX*.

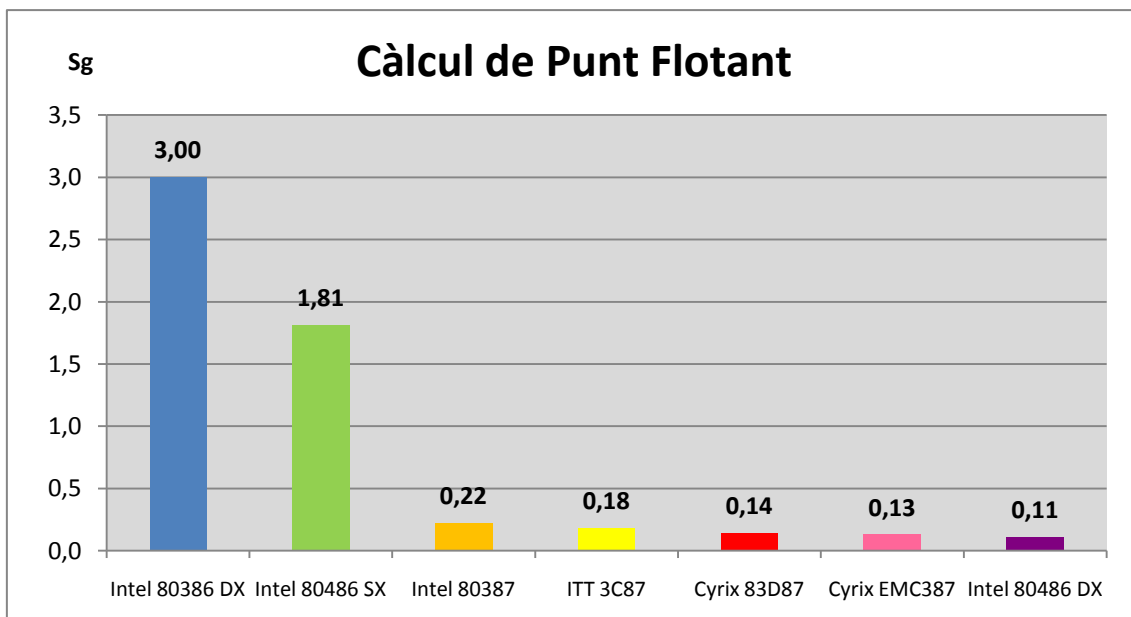
Així doncs, *Intel* s'assegura la comercialització dels seus xips, doncs molts dels usuaris que s'havien decantat per l'econòmic 80486 *SX*, acabaven comprant un segon processador i486 *DX* disfressat com a coprocessador matemàtic.

9.6. Comparatives de rendiment

Al mes d'octubre de 1991, l'empresa nord-americana *PC Labs* [7] va testar als seus laboratoris els coprocessadors matemàtics compatibles amb l'*Intel* 80387. Després de sotmetre a cada coprocessador a diverses proves de rendiment amb programari estàndard, *PC Labs* va arribar a la següent conclusió: “Els xips de *Cyrix* i *ITT* superen a l'oferta d'*Intel* amb innovacions de disseny i d'electrònica, però el 80486 és el campió total en velocitat”.

Les proves de coprocessadors matemàtics es van executar en un ordinador Compaq System Pro amb 12 Mb de memòria RAM, 840 Mb d'emmagatzematge en disc i una targeta de processador *Intel* 80386 a 33 MHz, 64 Kb de memòria cau externa i sockets per coprocessadors 80387 i *Weitek* Abacus 3167. Per fer les proves amb el i486 es va substituir la targeta de processador i386 amb una targeta 80486 a 33 MHz.

La prova Càlcul de *Punt Flotant* mesura la velocitat del coprocessador realitzant una sèrie de càlculs en *punt flotant*, incloent multiplicació, divisió, exponenciació i funcions logarítmiques i trigonomètriques. Sense coprocessador instal·lat, els mateixos càlculs es realitzen a la memòria RAM utilitzant un programa d'emulació de *punt flotant*.



Gràfica 1. Resultats de la prova de Càlcul de Punt Flotant.

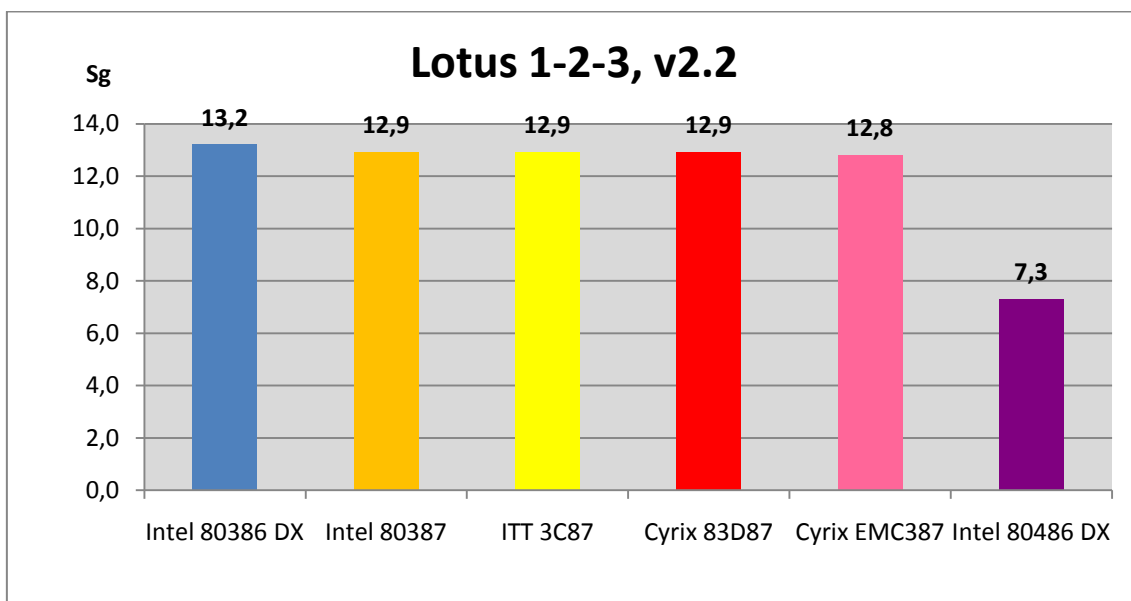
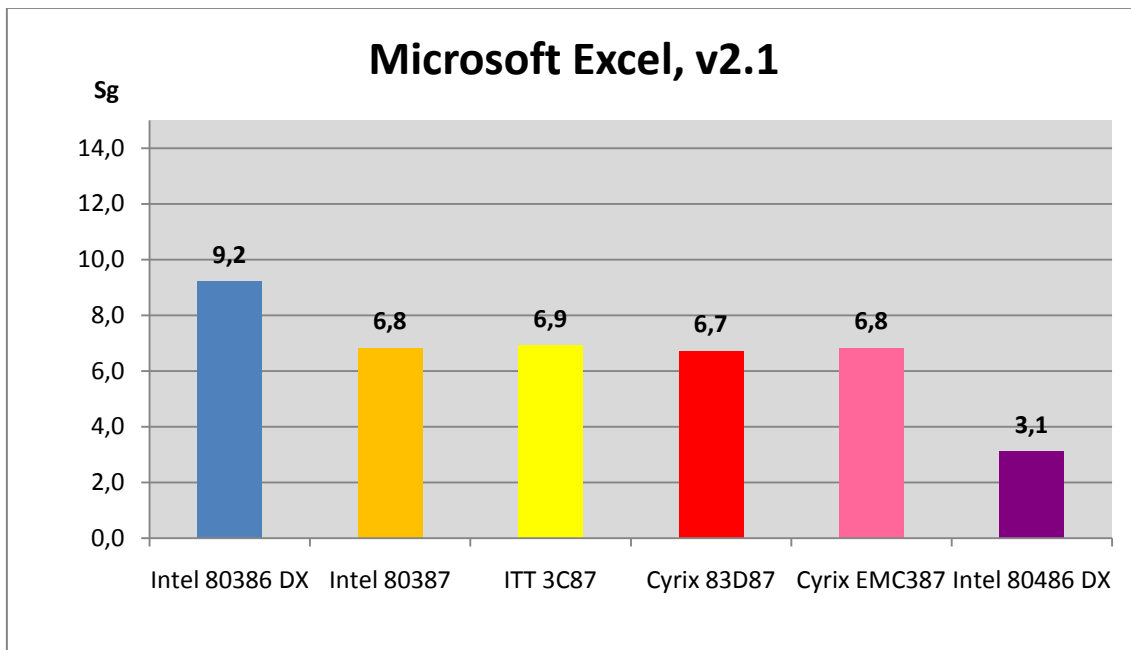
Els coprocessadors matemàtics van realitzar aquesta prova en menys d'una dècima part del temps que trigà un 80386 a 33 MHz sense coprocessador.

L'*ITT* 3C87 va obtenir uns resultats millors que l'*Intel* 80387 degut al seu disseny més eficaç. Els xips 83D87 i EMC87 de *Cyrix*, van executar la prova a una velocitat molt superior a la del coprocessador d'*Intel*. El motiu d'aquesta diferència de velocitats és

l'ús de codi hard en els *Cyrix*, en comptes del microcodi d'*Intel*. Sempre que s'executa microcodi s'afegeix temps de procés.

Malgrat aquestes dades van mostrar clares diferències de rendiment, probablement tots els coprocessadors treballaven igual amb aplicacions reals.

A continuació es mostren els resultats d'una prova amb programari comercial: Recàlcul de Fulls Matemàtics. Aquesta prova mesura el temps que triga un programa en calcular un full de càlcul que conté 1456 funcions matemàtiques (arrel quadrada, suma, resta, multiplicació, divisió i @SUM).



Gràfica 2. Resultats del test amb el programari de Microsoft i Lotus, els dos més utilitzats.

Com es pot observar a les gràfiques, la eficàcia amb la que les dues aplicacions utilitzen operacions en punt flotant al realitzar tasques idèntiques és diferent. **Lotus 1-2-3**, versió 2.2, fa servir menys el coprocessador matemàtic i mostra un increment general més petit en rendiment a totes les proves.

Cal destacar la impossibilitat de comparar el coprocessador de *Weitek* amb els dels altres fabricants. Les aplicacions utilitzades per la realització de les proves, no ofereixen suport pel *Weitek* 3167. Només una aplicació suporta tots els coprocessadors independentment de la seva estructura: **VersaCAD-386**.

No obstant, els resultats de les proves de rendiment amb aquest programari no van mostrar cap diferència significativa. El motiu del aparentment baix rendiment del *Weitek* Abacus 3167 és l'excés d'operacions de E/S, cosa molt comuna a les aplicacions comercials.

Per reflectir les impressionants característiques que utilitza *Weitek* en els seus materials de promoció, l'empresa va executar les seves proves emprant un gran disc *RAM* per reduir al mínim l'excés d'activitat en disc, i un coprocessador de gràfics d'alt cost per accelerar les operacions en pantalla. Si s'opta per tal estratègia incondicional de millora del rendiment, els xips Abacus de *Weitek* poden merèixer la pena.

Per concloure, si es treballa amb aplicacions que només tenen instruccions en *punt flotant*, l'addició d'un coprocessador matemàtic augmenta molt més el rendiment del computador que les dues o tres futures generacions de processadors.

En el cas del nostre estudi, pel que fa al rendiment, és preferible un i486 *DX* a qualsevol combinació de i386 i coprocessador. La raó de ser dels coprocessadors és el seu cost: és més econòmica l'adquisició conjunta d'un 80386 i un coprocessador matemàtic que l'adquisició d'un 80486 *DX*.

10. Conclusions

L'arquitectura x86 ha significat molt per la societat. Des de l'aparició dels primers microprocessadors d'aquesta família, hem estat pendents dels nous avanços i de les noves prestacions que cada nova generació ens proporcionava. Fins l'arribada de l'*Intel Core Duo*, quan es va posar fi a la carrera dels *GHz*, l'usuari només demanava potència (velocitat) i compatibilitat amb el programari que utilitzava.

A mesura que avançava la tecnologia, es dissenyaven processadors, i els seus corresponents coprocessadors, més ràpids, eficients i avançats. Les novetats que el 80386 va aportar a la família x86 van ser espectaculars, a banda del gran pas que va significar passar dels 16 als 32 *bits*. Un altre novetat, amb certa polèmica, va ser la inclusió d'una unitat de punt flotant a l'encapsulat a partir del i486.

Després de llegir amb molta cura l'article de comparació de rendiment dels coprocessadors matemàtics, arribo a la conclusió de que realment aquest dispositius, *quasi màgics*, no oferien l'espectacular augment de velocitat que prometien els fabricants. Calia un programari molt especialitzat o dominar molt bé les matemàtiques i la programació en assembler, per poder aprofitar els coprocessadors.

Només amb programari especialitzat, com per exemple els programes de disseny gràfic (*CAD*) o les aplicacions que feien moltíssims càlculs, principalment en punt flotant, es podia notar realment un augment a la velocitat de treball. De fet, molts de aquest programes no es podien executar sense la presència d'un coprocessador matemàtic al sistema.

S'ha quedat fora d'aquest estudi una "eina" molt coneguda a principis dels 90 que és: l'*EM87*. Aquest programari, que cabia a un disquet de 720 *KB*, permetia l'execució de grans aplicacions, com l'*AutoCAD*, sense necessitat d'adquirir un car coprocessador matemàtic. Si bé el rendiment de l'*EM87* no era espectacular i, a més, disminuïa la quantitat de memòria disponible, complia amb eficàcia la seva missió.

Es podria dir que aquest emulador per programari, era unes 3,5 vegades més lent que el coprocessador matemàtic que suplantava, però era ideal per estudiants, doncs qualsevol company podia "passar-te" el programa i no calia comprar una *FPU* o un flamant i486 *DX*, ambdós elements hardware suposaven una despesa impensable per un estudiant de l'època.

No podia acabar aquest apartat sense dedicar unes línies a la singular política comercial d'*Intel*. Ja s'ha comentat el cas del copro i487 *SX* i els *Overdrive*. Però tenim també el cas del *RapidCAD*, amb la col·laboració de tota la indústria informàtica. Realment és sorprenent que als Estats Units, amb les seves peculiars lleis antimonopoli i comercials, s'hagin permès aquestes pràctiques.

Però no va ser només Intel, doncs el seu gran aliat, *IBM*, va col·laborar tot el temps amb el fabricant de xips. Més tard es va trencar aquesta aliança, amb el famós cas de *MS Windows* i *IBM OS/2*, tothom traïa a tothom amb la intenció d'acaparar tota la quota de mercat. Finalment, i complint la llei, només van quedar dos fabricants principals de xips: *Intel* i *AMD*.

El que va ser, simplement, un gran fabricant de clònics dels xips d'*Intel*, s'ha convertit en el seu competidor, amb alguns productes molt més bons que els, suposadament, originals. L'*AMD K6-II* (i *K6-III*) va superar a totes les versions del Pentium 3 d'*Intel*. I també va ser *AMD* qui va apropar els 64 *bits* al mercat de consum, com ja havia fet abans al mercat professional.

No obstant això, Intel no només ens ha deixat tàctiques de mercat poc lícites. Gràcies als seus processadors, concretament al 8086/8088, hem conegut un dels majors invents de la societat moderna: el *PC*. Pràcticament, no podem viure sense un ordinador personal a cada llar, complint així les expectatives d'*Intel*, d'*IBM* i, com no, de Bill Gates.

El binomi *Windows/Intel* (*Wintel*) és, avui dia, qui domina el mercat. I a cada llançament d'una nova versió dels productes de qualsevol de les dues empreses, més o menys cada dos anys, implica una actualització de l'altra, complint així la llei de Moore.

Però el més curiós de tot és, que malgrat les crítiques i la gran expansió del programari lliure, la necessitat creada, tant per *Intel* com per *Microsoft*, fa que el consumidor desitgi tenir totes les novetats que aquestes empreses llancen al mercat. Estic convençut que si es tornés a llençar un 8086 amb un *DOS v3.3*, tornariem a treballar igual que fa 25 anys.

11. Desviacions en la planificació original i dificultats trobades

A l'hora de començar a redactar aquest document, s'han observat incoherències en els terminis planificats. Per tal de portar a bon termini aquest Treball, s'ha considerat oportú modificar la planificació de algunes de les tasques.










Un exemple molt clar es troba a la tasca **número 6**: "Creació de l'índex i la portada". Originalment, s'havia calculat un total de 2 dies per realitzar aquesta tasca, però, en la pràctica, només amb una hora dia n'hi ha prou. Així mateix, pel que fa a la tasca **número 4**: "Redacció de la Memòria", s'han augmentat el nombre de dies dedicats de 21 fins a 32, donat el gran volum d'informació que s'ha de processar.

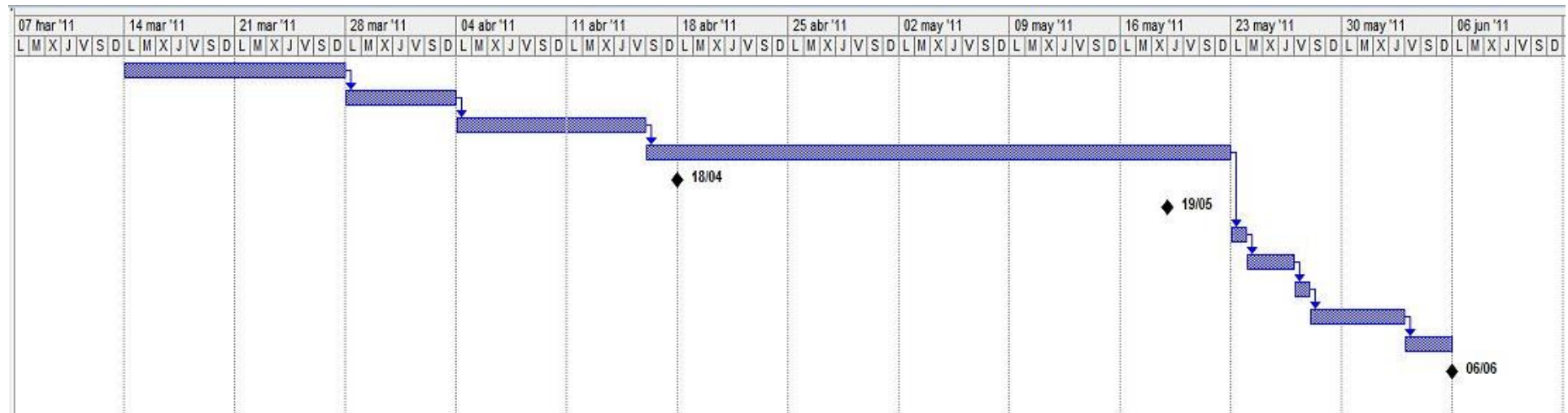
Pel que fa a les dificultats trobades, cal destacar la utilització dels temps verbals a l'hora de redactar els capítols clau d'aquest Treball. És molt difícil escriure coherentment en temps present sobre proves i articles de fa més de 30 anys. Ha resultat especialment difícil la redacció del capítol 9.6, "Comparatives de rendiment", al tractar els resultats d'uns tests que es van dur a terme l'octubre de 1991.

Una altra dificultat és la modificació, i realització, del diagrama de Gantt amb la nova planificació. Com es pot observar a la [pàgina 6](#) d'aquest mateix document, la qualitat del diagrama importat de l'aplicació **MS Project** és molt dolenta. Després de varies hores intentant instal·lar el programari gratuït **Open Project** sense èxit, es va optar per realitzar dues captures de pantalla del diagrama de Gantt per millorar la seva qualitat i aconseguir una major llegibilitat.

A la pàgina següent tenim el diagrama de Gantt definitiu d'aquest Treball amb els terminis definitius i adaptats a la realitat,

Arquitectures x86 i x87

		Nombre de tarea	Duración	Comienzo	Fin
1		Cercar informació	14 días	lun 14/03/11	dom 27/03/11
2		Classificar la informació	7 días	lun 28/03/11	dom 03/04/11
3		Esborrany i glossari	12 días	lun 04/04/11	vie 15/04/11
4		Elaboració de l'estudi	32 días	sáb 16/04/11	dom 22/05/11
5		Lliurament PAC2	1 día	lun 18/04/11	lun 18/04/11
6		Lliurament PAC3	1 día	jue 19/05/11	jue 19/05/11
7		Crear índex i portada	1 día	lun 23/05/11	lun 23/05/11
8		Correcció de la Memòria	3 días	mar 24/05/11	jue 26/05/11
9		Seleccionar informació	1 día	vie 27/05/11	vie 27/05/11
10		Creació de la Presentació	5 días	sáb 28/05/11	jue 02/06/11
11		Repàs general	3 días	vie 03/06/11	dom 05/06/11
12		Lliurament Memòria i Presentació	1 día	lun 06/06/11	lun 06/06/11



12. Annex A. La CMOS RAM

La memòria *CMOS RAM* és un àrea de memòria no volàtil utilitzada per els ordinadors *AT* i compatibles per emmagatzemar el rellotge en temps real i altres informacions de la configuració del sistema. Una de les principals diferències entre els ordinadors *PC/XT* i els *AT*, entenent per ordinador *AT* tot sistema amb microprocessador 80286 o superior, és la presència de la *CMOS RAM*.

Tots els ordinadors *AT* tenen un xip pel rellotge en temps real. Aquest xip és el **Motorola MC146818A** [\[43\]](#), o un clònic, i posseeix, al menys, 64 *Bytes* de memòria. Com indica el seu nom, es tracta d'una memòria de baixa potència alimentada mitjançant una bateria interna; és a dir, el seu contingut no es perd al apagar el sistema.

La *CMOS RAM* ha de ser memòria de lectura o escriptura perquè el seu contingut s'actualitza constantment. No és només que el rellotge en temps real es modifiqui cada segon, si no també que la configuració del computador pateix canvis al llarg del seu temps de vida útil.

La *CMOS* deu ser una memòria no volàtil per dos motius. D'una banda, el rellotge en temps real ha d'estar encès en tot moment. D'altra banda, les dades de configuració i diagnòstic emmagatzemats a la *CMOS RAM* s'utilitzen durant el *POST* del sistema. El *POST* designa el procés de chequeig i inicialització generat al encendre la computadora.

Les tres primeres accions del *POST* són, en aquest ordre, inicialitzar el microprocessador, realitzar un *checksum* de la *ROM BIOS* i llegir els valors de la *CMOS RAM*. A partir d'aquest punt, totes les operacions posteriors del *POST* assumeixen els valors de la *CMOS RAM* com paràmetres de configuració del sistema.

Estructura general

Els 64 *Bytes* de la *CMOS RAM* es divideixen en tres grans blocs. En primer lloc 10 *Bytes* que emmagatzemen la informació del rellotge en temps real, incloent la data i l'hora activa, així com l'hora de l'alarma.

A continuació un bloc de 4 *Bytes* correspon a varis registres d'estat. Aquests registres s'interpreten a nivell de bits i inclouen indicadors de l'estat de la *interrupció* de l'alarma o de la *interrupció* periòdica o el valor dels *flags* *IRQF*, *PF*, *AF* i *UF*.

Finalment, apareixen varis *Bytes* amb informació diversa referent a la configuració del sistema i el procés de *POST*. Per exemple, la mida de la memòria convencional i la memòria expandida, el tipus de disc dur instal·lat al sistema, ...

Un *Byte* molt important és l'indicador de la raó de la reinicialització del sistema (offset Fh). En els ordinadors *AT* el microprocessador arrenca en *mode real* i testeja els dispositius i la memòria convencional.

Seguidament, commuta a *mode protegit* i testeja la memòria expandida. Però, al tornar de nou al *mode real*, el microprocessador es reinicialitza i, per tant, es produeix un altre procés de *POST*.

Per evitar la reinicialització del sistema, el procés *POST* llegeix el *Byte* de reinici (*shutdown*) situat al offset Fh de la *CMOS RAM*. Si el seu valor està dins l'interval que va des de 1h fins a 8h, no es realitza un altre cop un nou *POST*.

L'accés a la *CMOS RAM*

Per accedir al contingut de la memòria *CMOS RAM*, s'utilitzen els ports E/S 70h i 71h. El primer és un port d'escriptura en el qual s'escriu l'offset del valor que es vol llegir o escriure. En canvi, el 71h és un port de lectura i escriptura en el qual es llegeix o s'escriu el valor del *Byte* de l'offset indicat al port 70h.

Un detall important és no utilitzar la lectura i l'escriptura mitjançant ports quan es treballi amb els valors del rellotge en temps reals (offset 00h-09h). És preferible fer servir la *Interrupció* 1Ah que funciona a una velocitat molt més ràpida.

A la pàgina següent es pot observar una taula amb l'estructura general de la *CMOS RAM*,

LA CMOS RAM		
Offset	Longitud	Descripció
00h	1 byte	Segons de l'hora activa
01h	1 byte	Segons de l'alarma
02h	1 byte	Minut de l'hora activa
03h	1 byte	Minuts de l'alarma
04h	1 byte	Hores de l'hora activa
05h	1 byte	Hores de l'alarma
06h	1 byte	Dia de la setmana actual
07h	1 byte	Dia del mes actual
08h	1 byte	Mes actual
09h	1 byte	Any actual
0Ah	1 byte	Registre d'estat A
0Bh	1 byte	Registre d'estat B
0Ch	1 byte	Registre d'estat C
0Dh	1 byte	Registre d'estat D
0Eh	1 byte	Byte de diagnòstic
0Fh	1 byte	Causa del reset
10h	1 byte	Tipus de les unitats de disquets Bits 7 – 4 = Unitat de disquets 1 (1) Bits 3 – 0 = Unitat de disquets 2 (1)
11h	1 byte	Tipus de les unitats de disquets Bits 7 – 4 = Unitat de disquets 3 (1) Bits 3 – 0 = Unitat de disquets 4 (1)
12h	1 byte	Tipus BIOS dels discs durs Bits 7 – 4: Tipus de la unitat de disc dur 1 (2) Bits 3 – 0: Tipus de la unitat de disc dur 2 (2)
13h	1 byte	(Reservat)
14h	1 byte	Byte d'equip Bits 7 – 6: Nombre d'unitats de disquets – 1 Bits 5 – 4: Sistema de vídeo (3) Bit 1 = 1: Coprocessador matemàtic instal·lat Bit 0 = 0: No n'hi ha unitats de disquets
15h	2 bytes	Mida en Kb de memòria convencional
17h	2 bytes	Mida en Kb del total de memòria expandida
19h	1 byte	Tipus BIOS unitat disc dur 1 (si és major que 14)
1Ah	1 byte	Tipus BIOS unitat disc dur 2 (si és major que 14)
1Bh	19 bytes	(Reservat)
2Eh	2 bytes	Valor Checksum per 10h – 2Dh
30h	2 bytes	Mida en Kb de memòria expandida lliure
32h	1 byte	Segle en format BCD
33h	1 byte	Flag d'informació

(1) 0 = No n'hi ha unitat; 1 = Unitat de 360 Kb; 2 = Unitat de 1.2 MB; 3 = Unitat de 720 KB; 4 = Unitat de 1.44 MB

(2) 0 = No n'hi ha unitat; 1h – Eh = Unitat del tipus 1-14; Fh = Veure offset 19h ó 1Ah

(3) 0 = Reservat; 1 = CGA 40 columnes; 2 = CGA 80 columnes; 3 = CGA monocrom

Taula 3. Estructura general de la CMOS RAM

13. Annex B. Característiques tècniques

A continuació es troben les taules per poder comparar les característiques tècniques de tots els xips tractats en el present estudi,

Xip	Fabricant	Freqüències (MHz)	Bus de dades	Bus d'adreces	Mida circuit	Transistors	Encapsulat	Any
8086	Intel	5, 8, 10	16 bits	20 bits	3,00 µ	29.000	DIP-40	1.978
8088	Intel	5, 8	8 bits	20 bits	3,00 µ	29.000	DIP-40	1.979
80286	Intel	6, 8, 10, 12	16 bits	24 bits	1,50 µ	134.000	LCC-68, PGA-68	1.982
80386 DX	Intel	16, 20, 25, 33	32 bits	32 bits	1,50 µ - 1,00 µ	275.000	PGA-132	1.985
80386 SX	Intel	16, 20, 25, 33	16 bits	24 bits	1,50 µ	275.000	PQFP-100	1.988
80486 DX	Intel	25, 33, 50	32 bits	32 bits	1,00 µ - 0,8 µ	1.185.000	PGA-168	1.989
80486 SX	Intel	20, 25, 33	32 bits	32 bits	1,00 µ - 0,8 µ	1.185.000	PGA-168	1.991
80486 DX2	Intel	40, 50, 66	32 bits	32 bits	0,80 µ	1.200.000	PQFP-208, PGA-168	1.992
80486 DX4	Intel	75, 100	32 bits	32 bits	0,60 µ	1.600.000	PQFP-208, PGA-168	1.994
Overdrive	Intel	50, 66, 75, 100	32 bits	32 bits	0,80 µ	1.600.000	PGA-168	1.992

Taula 4. Característiques dels microprocessadors estudiats.

Xip	Fabricant	Freqüències (MHz)	Bus de dades	Bus d'adreces	Mida circuit	Transistors	Encapsulat	Any
8087	Intel	5, 8, 10	16 bits	20 bits	3,00 µ	45.000	DIP-40	1.980
80287	Intel	6, 8, 10, 12	16 bits	24 bits	N/D	N/D	DIP-40	1.984
80387 DX	Intel	16, 20, 25, 33	32 bits	32 bits	1,50 µ - 1,00 µ	N/D	PGA-68	1.987
80387 SX	Intel	16, 20, 25, 33	32 bits	32 bits	1,00 µ	N/D	PLCC-48	1.990
83D87	Cyrix	16, 20, 25, 33	32 bits	32 bits	1,20 µ	N/D	PGA-68	1.989
EMC87	Cyrix	20, 25, 33, 40	32 bits	32 bits	1,20 µ	N/D	PGA-121	1.990
3C87	IIT	16, 20, 25, 33	32 bits	32 bits	1,20 µ	N/D	PGA-68	1.989
Abacus 3167	Weitek	20, 25, 33	32 bits	32 bits	N/D	N/D	PGA-121	1.989
80487 SX	Intel	25	32 bits	32 bits	1,00 µ	1.175.000	PGA-168	1.993

Taula 5. Característiques tècniques dels coprocessadors matemàtics.

En la següent taula es descriuen els sòcols utilitzats per cadascun dels xips d'aquest estudi,

Sòcol/encapsulat	Descripció	Xip que el fa servir
DIP-40	<i>(Ceramic) Dual In-Line Package.</i> De 40 pins. Consisteix en un bloc amb dues fileres paral·leles de pins. La separació estàndard entre dos pins és de 0,1" (2,54 mm).	8086, 8088, 8087, 80287
LCC-68	<i>Leadless Chip Carrier.</i> De 68 pins. Consisteix, generalment, en un quadrat amb el mateix nombre de pins a cada costat, encara que, a vegades, tenen forma rectangular. La distància entre pins sempre és 0,05" (1,27 mm).	80286
PGA-68	<i>Pin Grid Array.</i> De 68 pins. És un quadrat de connectors amb forma de forats on s'insereixen els pins del xip a pressió. La separació entre dos pins és de 0,1" (2,54 mm).	80286, 80387 DX, Cyrix 83D87, IIT 3C87
PGA-132	<i>Pin Grid Array.</i> De 132 pins. (Veure PGA-68)	80386 DX
PQFP-100	<i>(Plastic) Quad Flat Package.</i> De 100 pins. Consisteix en un quadrat amb els pins repartits a parts iguals per les quatre cares. La separació entre pins varia en funció del nombre d'aquest, entre 0,4 mm i 1 mm.	80386 SX
PGA-168	<i>Pin Grid Array.</i> De 168 pins. (Veure PGA-68)	Totes les variants de l'Intel 80486 DX
Socket 1 (PGA-169)	<i>Pin Grid Array.</i> De 169 pins. (Veure PGA-68)	80486 SX
Socket 2 (PQFP-208)	<i>(Plastic) Quad Flat Package.</i> De 208 pins. (Veure PQFP-100).	Tots els i486 DXn
PLCC-48	<i>(Plastic) Leadless Chip Carrier.</i> De 48 pins. (Veure LCC-68)	80387 SX
PGA-121	<i>Pin Grid Array.</i> De 121 pins. (Veure PGA-68)	Weitek 3117, Cyrix EMC87

Taula 6. Característiques dels diferents sòcols.

Alguns exemples d'encapsulat de microprocessadors,

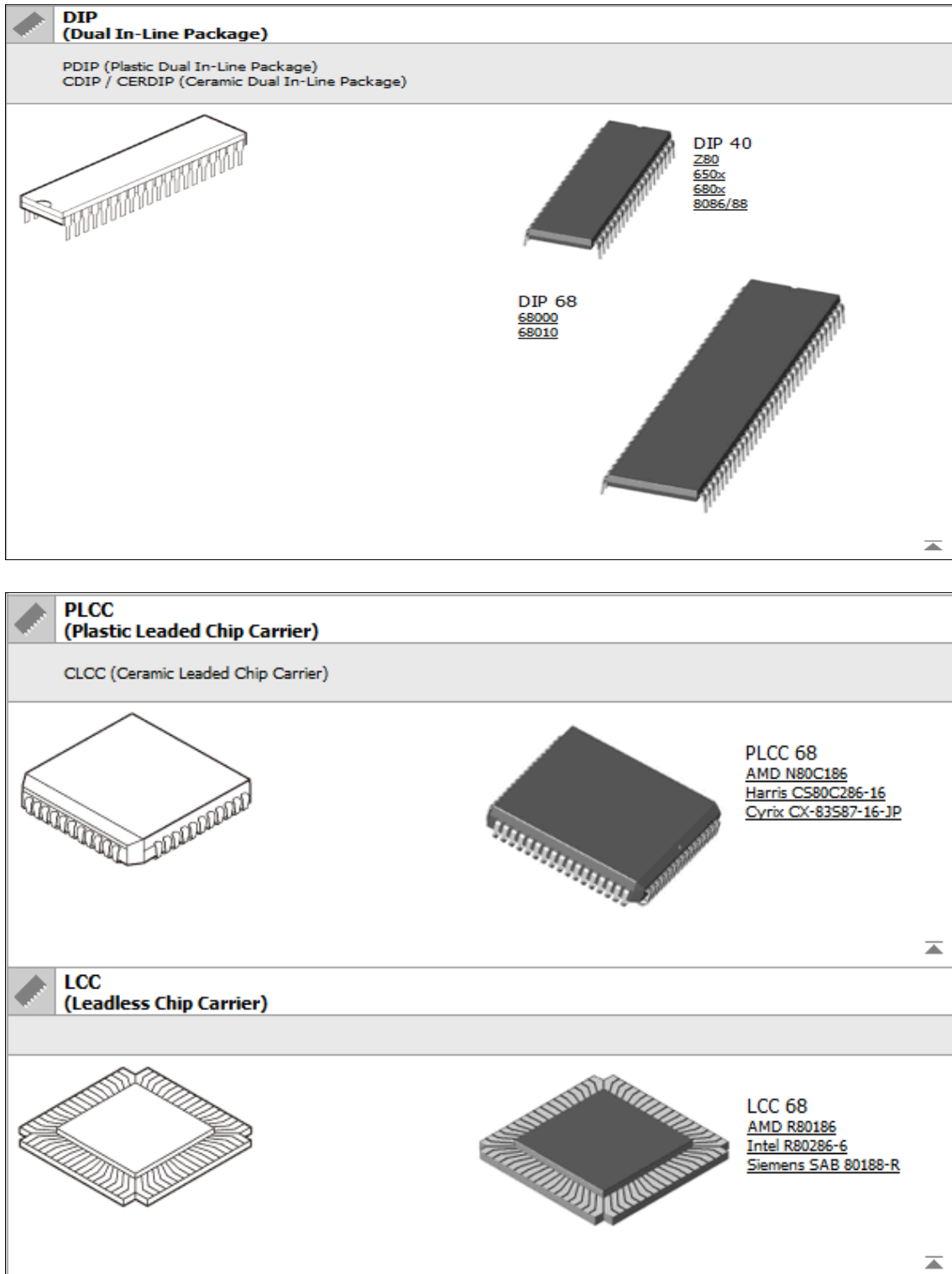


Figura 34. Detall de varis dels encapsulats dels xips tractats a l'estudi.

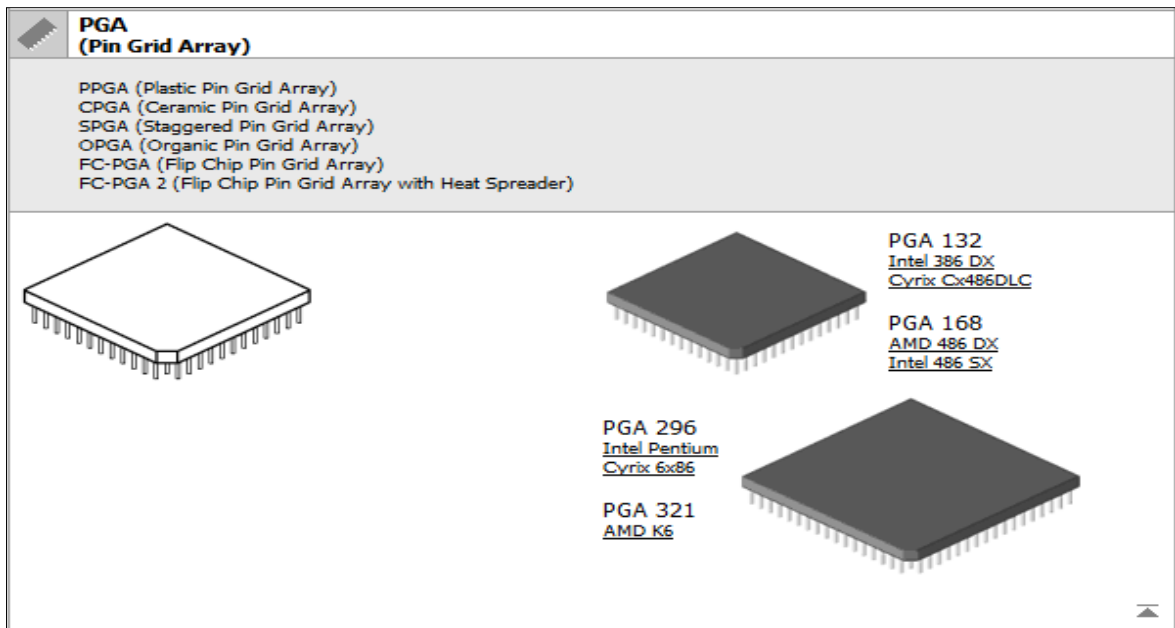


Figura 35. Detall de l'encapsulat *PGA*.

14. Annex C. 80387: La visió del programador

Per els programadors, el [80387](#) és més que un perifèric secundari: és part integrant d'un sistema [80386](#).

Dedicat a manejar aritmètica de nombres reals, el 80387 pot realitzar càlculs numèrics més ràpida i fàcilment que el 80386. Però només connectant un 80387 a un *PC* no s'aconsegueix automàticament que el programari s'executi més ràpid al realitzar matemàtica en *punt flotant*. Per aconseguir la potència del i387, un programa s'ha d'assegurar que els càlculs es realitzin en el coprocessador.

Execució concurrent

El microprocessador 80386 crea la il·lusió de que està realitzant les operacions en *punt flotant*.

Quan el i386 es troba una instrucció 80387 en un programa, espera fins que el coprocessador està preparat i després li envia la instrucció al coprocessador. Un cop que el i387 inicia l'execució, el 80386 passa a la següent instrucció: no espera que el coprocessador acabi. Això permet seguir funcionant al i386 mentre el coprocessador està executant una instrucció complexa.

Tipus de dades 387

Malgrat el 80387 només utilitza nombres en *punt flotant* internament, és capaç de reconèixer i accedir a set tipus de dades en un programa.

Tres d'aquests tipus de dades, de mides 16, 32 i 64 *bits*, són enters binaris. Són idèntics als utilitzats en la programació en llenguatge ensamblador 8086 convencional i pot emmagatzemar valors màxims de 9E18. Un quart tipus fa servir un format *BCD* que reuneix dos enters decimals en grups de 4 *bits*. Pot emmagatzemar un nombre de 18 dígitos als seus 72 *bits*.

Per últim, el 80387 suporta tres formats de nombre real en 32, 64 i 80 *bits*, i emmagatzema valors d'un màxim de 3.37E4932. Dins del coprocessador matemàtic 80387, tots els números es converteixen automàticament a l'últim d'aquests tipus, un format conegut com "temporal real". Els set tipus reserven el primer *bit* com *bit* de signe.

El fet de tenir tants tipus de dades proporciona una considerable flexibilitat al programador, malgrat s'hagin de conèixer les diferents versions d'una mateixa instrucció 80387. Per exemple, *FADD* s'utilitza per sumar nombres reals, però *FIADD* es fa servir per sumar nombres enters. Afortunadament, un assemblador pot detectar tots els possibles errors.

Arquitectura 387

El processador 80386 ofereix *registres*, *flags* i *bits* d'estat per carregar i manipular adreces i valors. El coprocessador 80387 també els ofereix.

El 80387 conté vuit registres de 80 *bits* que corresponen al format de dades de precisió ampliada de 80 *bits*. Malgrat aquestos es poden utilitzar com *registres* individuals, estan dissenyats per ser utilitzats com una *pila*. Moltes instruccions 387 tenen l'efecte lateral de modificar la part superior de l'indicador de *pila*, i implícitament col·loquen els resultats al registre superior de la *pila*.

Així mateix, el registre d'estat 80387 li indica si el coprocessador està preparat, quin registre de dades correspon a la part superior de la pila i si s'ha produït una excepció (error). Un registre de control conté camps per controlar la precisió, l'arrodoniment i les interrupcions.

Programant el 80387

Els programadors poden incloure instruccions 80387 als seus programes 80386 en llenguatge assemblador.

Però, deuen utilitzar la directiva *.387* per l'assemblador per acceptar instruccions 80387 i permetre la utilització de notació específica, com 1.23E4, per exemple, quan s'especifiquen números. Aquesta directiva també converteix constants a format de nombre real.

A nivell binari, totes les instruccions 80387 comencen amb els mateixos cinc *bits*: "11011", que instrueixen al 80386 per passar-los al coprocessador matemàtic. Així doncs, totes les comandes 80387 comencen per "F", com per exemple *FADD* en comptes d'*ADD*. La instrucció *FINIT* inicialitza el coprocessador i és necessari el seu ús la primera vegada que el i387 s'utilitza en un programa.

El 80387 ofereix un conjunt d'instruccions que suporta una multitud d'operacions matemàtiques, incloent comandes per carregar logaritmes i el nombre pi, a més de calcular arrels quadrades. Altres instruccions permeten gravar i emmagatzemar el contingut d'un 80387 en una posició de memòria. A més, algunes instruccions són exclusives i no es poden trobar en el grup d'instruccions d'un 80287.

Pels programadors detectar un coprocessador 80387 no és difícil. Un cop s'ha reinicialitzat, el 80386 ajusta automàticament un *bit* del seu *Registre de Control* a 0. Si està fixat en 1, aquest *bit* indica la presència d'un 80387, en qualsevol altre cas, el *bit* val 0. A diferència d'anteriors sistemes, els sistemes 80387 no requereixen canvis als paràmetres després d'instal·lar un coprocessador, es detecta automàticament.

15. Bibliografia

PHOENIX TECHNOLOGIES, Ltd (1989). *System BIOS for IBM PC/XT/AT Computers and Compatibles*. [s.l.]: Addison-Wesley.

CHARTE, F. (2009). *Ensamblador*. Madrid: Anaya Multimedia.

SAN JUAN, J. (1991). "Coprocesadores matemáticos". *PC Magazine, Edición Española* (núm. 42, pàg. 226-241).

BALCH, M. (2003). *Complete Digital Design*. New York [etc.]: McGraw-Hill.

SALAS, J. (1989). *Sistemas Operativos y compiladores*. Madrid [etc.]: McGraw-Hill.

MICROSOFT CORPORATION (ed.). (1993). *MS-DOS 6.2, Manual del usuario*. Neunkirchen: Microsoft Corporation.

BARZANALLANA, R. (2005). *Historia de la informática*. [en línia]. http://www.wikilearning.com/monografia/historia_de_la_informatica/3653-3 [data de consulta: 17/03/2011].

HERNANDEZ, E. (2001). *Microprocesadores*. [en línia]. <http://www.monografias.com/trabajos/microproce/microproce.shtml> [data de consulta: 21/03/2011].

Anònim (2011?). *CPU Collection*. [en línia]. <http://www.cpu-collection.de/?tn=0> [data de consulta: 23/03/2011].

Anònim (2007). *Coprocessor info*. [en línia]. <http://coprocessor.cpu-info.com/index.php> [data de consulta: 25/03/2011].

YRAOLAGOITIA, J. de (1990). "La memoria CMOS RAM". *PC WORLD España* (núm. 55, pàg. 289-291).

CALLES, I. (1991). "Arquitectura de Microprocesadores: CISC y RISC". *PC WORLD España* (núm. 74, pàg. 189-200).

Bibliografia complementaria

NICOLAU, F.; CUENCA, M. J. (2010). *Competència comunicativa per a professionals de les TIC*. Barcelona: FUOC.

16. Referències externes

- [1] microprocessadors. <http://es.wikipedia.org/wiki/Microprocesador>
- [2] coprocessadors matemàtics. <http://es.wikipedia.org/wiki/Coprocesador>
- [3] família x86. <http://es.wikipedia.org/wiki/X86>
- [4] família x87. <http://es.wikipedia.org/wiki/X87>
- [5] Quarta generació de computadores. http://es.wikipedia.org/wiki/Cuarta_generaci%C3%B3n_de_computadoras
- [6] Arquitectura x86. http://es.wikipedia.org/wiki/Categor%C3%ADa:Arquitectura_x86
- [7] Microcomputadora. <http://es.wikipedia.org/wiki/Microcomputadora>
- [8] Arquitectura de von Neumann. http://es.wikipedia.org/wiki/Arquitectura_de_von_Neumann
- [9] Cicle d'instrucció. http://es.wikipedia.org/wiki/Ciclo_de_instrucci%C3%B3n
- [10] Comptador de programa. http://es.wikipedia.org/wiki/Contador_de_programa
- [11] Memòria cau. <http://es.wikipedia.org/wiki/Cach%C3%A9>
- [12] Pipeline. <http://es.wikipedia.org/wiki/Pipeline>
- [13] ISA. http://es.wikipedia.org/wiki/Conjunto_de_instrucciones
- [14] Opcode. http://es.wikipedia.org/wiki/C%C3%B3digo_de_operaci%C3%B3n
- [15] Microcodi o microprograma. <http://es.wikipedia.org/wiki/Microc%C3%B3digo>
- [16] Registre. [http://es.wikipedia.org/wiki/Registro_\(hardware\)](http://es.wikipedia.org/wiki/Registro_(hardware))
- [17] Punter de Pila. http://es.wikipedia.org/wiki/Registro_de_pila
- [18] Pila. [http://es.wikipedia.org/wiki/Pila_\(inform%C3%A1tica\)](http://es.wikipedia.org/wiki/Pila_(inform%C3%A1tica))
- [19] Acumulador. [http://es.wikipedia.org/wiki/Acumulador_\(inform%C3%A1tica\)](http://es.wikipedia.org/wiki/Acumulador_(inform%C3%A1tica))
- [20] Bus de dades. http://es.wikipedia.org/wiki/Bus_de_datos#Clases_de_buses
- [21] Bus d'adreces. http://es.wikipedia.org/wiki/Bus_de_direcciones
- [22] Bus de control. http://es.wikipedia.org/wiki/Bus_de_control
- [23] Interrupció. <http://es.wikipedia.org/wiki/Interrupci%C3%B3n>
- [24] CISC. http://es.wikipedia.org/wiki/Complex_instruction_set_computing
- [25] RISC. <http://es.wikipedia.org/wiki/RISC>

- [26] MMU. http://es.wikipedia.org/wiki/Unidad_de_gesti%C3%B3n_de_memoria
- [27] Modes de funcionament. http://es.wikipedia.org/wiki/Categor%C3%ADa:Modos_de_operaci%C3%B3n_de_los_microprocesadores_x86
- [28] HyperThreading. <http://www.intel.com/cd/corporate/techtrends/emea/spa/platform-technology/hyper-threading/310512.htm>
- [29] Consultar bibliografia, article "[Coprocesadores matemáticos](#)".
- [30] 8086 Datasheet. Dades oficials de l'Intel 8086. http://www.datasheetcatalog.org/datasheets/2300/499305_DS.pdf
- [31] Segmentació. [http://es.wikipedia.org/wiki/Segmentaci%C3%B3n_\(inform%C3%A1tica\)](http://es.wikipedia.org/wiki/Segmentaci%C3%B3n_(inform%C3%A1tica))
- [32] 8088 Datasheet. Dades oficials de l'Intel 8088. <http://www.datasheetcatalog.org/datasheet/Intel/mXrysuv.pdf>
- [33] Barrel Shifter. http://en.wikipedia.org/wiki/Barrel_shifter
- [34] Paginació. http://es.wikipedia.org/wiki/Paginaci%C3%B3n_de_memoria
- [35] 80486 DX Datasheet. Dades oficials de l'Intel 80486 DX. http://www.datasheetcatalog.org/datasheets/restul/493187_DS.pdf
- [36] TLB. <http://es.wikipedia.org/wiki/TLB>
- [37] 8087 Datasheet. Dades oficials de l'Intel 8087. http://www.datasheetcatalog.org/datasheets/2300/45014_DS.pdf
- [38] 80287 Datasheet. Dades oficials de l'Intel 80287. <http://www.datasheetcatalog.org/datasheet/Intel/mXrvury.pdf>
- [39] L'estàndard ANSI IEEE 754-1985. <http://standards.ieee.org/findstds/standard/754-1985.html>
- [40] 80387 DX Datasheet. Dades oficials de l'Intel 80387 DX. <http://www.datasheetcatalog.org/datasheet/Intel/mXtyvwz.pdf>
- [41] 80387 SX Datasheet. Dades oficials de l'Intel 80387 SX. <http://www.datasheetcatalog.org/datasheet/Intel/mXuzzyw.pdf>
- [42] PC Labs. Test de rendiment dels diferents coprocessadors matemàtics. [PC Labs](#)
- [43] Datasheet del Motorola MC146818A. http://www.datasheetcatalog.com/datasheets_pdf/M/C/1/4/MC146818A.shtml

17. Glossari

AMD: Advanced Micro Devices, Inc. Empresa nord-americana dedicada a la producció de microprocessadors, compatibles amb la família x86 d'*Intel*, i d'altres dispositius semiconductors. Adreça web: <http://www.amd.com>

Bit: Acrònim, anglès. Binary digit (dígit binari). Unitat bàsica d'informació que pot representar dos valors assignant un dels valors a l'estat apagat (0) i l'altre a l'estat encès (1).

Byte: Anglès, que significa Octet. S'utilitza com unitat bàsica d'emmagatzament d'informació. Un Byte correspon a 8 bits.

CMOS RAM: Acrònim, anglès. Complementary Metal Oxide Semiconductor, que ens indica que es tracta d'una memòria de baixa potència alimentada mitjançant bateria interna. Veure [l'annex A](#).

Cyrix: Empresa fundada en 1988 per ex executius de Texas Instruments. Aquesta empresa es va especialitzar en la fabricació de coprocessadors matemàtics d'alt rendiment per sistemes 80286 i 80386. A l'any 1999 va ser comprada per VIA Technologies, Inc. Adreça web: <http://www.viatech.com>.

DX: Acrònim, anglès. Double-word eXternal. Utilitzat en els xips d'Intel per indicar que el bus de dades (extern) té una mida Double-Word (32 *bits*).

Flag: Anglès, que significa bandera o indicador. Es refereix a un o més *bits* que s'utilitzen per emmagatzemar un valor binari o codi que té assignat un significat.

FPU: Acrònim, anglès. Floating Point Unit (Unitat de punt flotant). Amb aquestes sigles es coneixen els coprocessadors matemàtics.

Hardware. Anglès que correspon a totes les parts tangibles d'un computador: components electrònics, mecànics, cables, etc.

Hertz: és la unitat de la freqüència i significa un cicle per segon. En informàtica s'utilitzen els Megahertz per mesurar la velocitat de rellotge dels processadors. Als anys 90 es pensava que a major nombre de MHz major era el rendiment dels processadors, però es va demostrar que això no era del tot cert.

IBM: International Business Machines Corporation, empresa fundada al juny de 1911 en Nova York (USA). La companyia fabrica i ven ordinadors, maquinari divers, programari i serveis arreu del món. És un dels gegants del món informàtic. Adreça web: <http://www.ibm.com>.

IEEE: Acrònim, anglès. Institute of Electrical and Electronics Engineers (Institut d'enginyers elèctrics i electrònics). És una associació tècnic-professional mundial dedicada a l'estandardització, entre d'altres coses. El seu propòsit principal és compartir i aplicar els avanços tecnològics per a benefici de la humanitat i dels professionals. Adreça web: <http://www.ieee.org>.

IIT: Integrated Information Technology, Inc. Aquesta empresa es va crear l'any 1987 in Califòrnia (USA) i es va especialitzar en la producció de coprocessadors matemàtics i controladores gràfiques destinades als usuaris domèstics. Al 1997 la companyia va canviar la línia de producte i es va dedicar a la producció de chips de videoconferència, canviant també el seu nom per 8x8. Adreça web: <http://www.8x8.com>.

Intel: Intel Corporation, empresa nord-americana de semiconductors caracteritzada per la creació de la família de microprocessadors x86, els més utilitzats arreu del món en la informàtica personal i empresarial. Adreça web: <http://www.intel.com>.

Megahertz: Equivalent a 10^6 hertz (1 milió). El seu símbol és MHz. (Veure [Hertz](#)).

Micró: El micró o micròmetre és una unitat de longitud equivalent a la milionèsima part d'un metre (o el que és el mateix, una mil·lèsima part d'un mil·límetre). El seu símbol és μm . En el passat es va anomenar micra (símbol μ), però aquest nom ara és obsolet.

NMOS: Acrònim, anglès. Negative-channel Metal-Oxide Semiconductor. És un tipus de semiconductor que es carrega negativament de mode que els transistors s'encenen o s'apaguen amb el moviment dels electrons. És el tipus de tecnologia utilitzat en la fabricació dels primers circuits integrats.

Open Project: Programari de codi obert alternatiu al totpoderós Microsoft Project. Adreça web: <http://openproject.sourceforge.net>.

PC: Acrònim, anglès. Personal Computer (ordinador personal), es tracta d'una micro computadora dissenyada per ser utilitzada només per una sola persona. Va ser l'estratègia d'*IBM* per ingressar en el mercat de les computadores domèstiques.

PC AT: Evolució natural del *PC XT*. Equipat amb microprocessador 80286 a 8 MHz, disc dur de 20 MB, bus de 16 bits i 1 MB de memòria RAM com a màxim.

PC Labs: Empresa nord-americana especialitzada en la realització de test i proves de rendiment i funcionament de maquinari i programari divers. És tot un referent a nivell mundial. Adreça web: <http://www.pclabs.com>.

PC XT: Evolució del *PC* original destinada al món empresarial. Les seves característiques més rellevants eren: processador *Intel* 8088 a 4.77MHz, disc dur de 10 MB, adaptador de pantalla MDA (monocrom) i 256 KB de memòria.

POST: Acrònim, anglès. Power-On Self Test, que designa el procés de chequeig i inicialització generat quan s'encén la computadora.

Punt flotant (o coma flotant): És una forma de notació científica utilitzada als microprocessadors i *FPU* amb la qual es poden representar nombres reals molt grans o molt petits d'una manera molt eficient i compacta, i amb la que es poden fer operacions aritmètiques.

RAM: Acrònim, anglès. Random Access Memory (Memòria d'accés aleatori). És la memòria des de on el processador rep les instruccions i guarda els resultats.

ROM BIOS (o BIOS): Acrònim, anglès. BIOS (Basic Input/Output Sytem - Sistema bàsic d'entrada/sortida) o ROM (Read Only Memory – Memòria de només lectura) BIOS. Es tracta d'un xip que no es pot esborrar i que té gravades les instruccions bàsiques necessàries per poder engegar una computadora i que comprova tots els discos, memòria, perifèrics, etc., que estan connectats a l'equip per veure si estan correctament configurats.

Shutdown: Anglisme, que significa apagat o tancament. Es refereix al procés d'apagament d'una computadora.

SX: Acrònim, anglès. Simple-word eXternal. Utilitzat en els xips d'Intel per indicar que el bus de dades (bus extern) té una mida de 16 bits (Word o Simple-Word).

Weitek: Weitek Corporation, va ser una empresa nord-americana dedicada a la producció de coprocessadors matemàtics per varies famílies de microprocessadors. A principis dels anys 90 va ser el referent en el disseny de supercomputadores de processament en paral·lel. Problemes amb llicències i patents van forçar la seva desaparició. L'any 1996 va ser comprada per Rockwell Semiconductor. Adreça web: <http://rockwellautomation.com>.