COPROCESSORI X86 E 68K

COME FUNZIONANO I COPROCESSORI MATEMATICI X87?

In un sistema x86 con un coprocessore x87, le istruzioni della CPU e del coprocessore matematico sono eseguite contemporaneamente. Questo significa che la CPU può eseguire le sue istruzioni mentre il coprocessore esegue le sue nello stesso tempo. La contemporaneità però alle volte è limitata dal fatto che la CPU deve aiutare il coprocessore in alcune operazioni. Dato che la CPU e il coprocessore sono alimentati dallo stesso flusso di istruzioni e entrambi i flussi di istruzioni possono operare sugli stessi data, ci deve essere una sincronizzazione tra la CPU e il coprocessore.

COME FUNZIONANO I COPROCESSORI MATEMATICI 68K?

Le schede acceleratrici Amiga usano anche una unità Floating Point per i calcoli intensivi in virgola mobile. Le principali FPU sono esterne come il 68881 e 68882 e interne come quella per il 68040 e 68060.

1) 8087

In un sistema 8086/8088 con coprocessori 8087, entrambi i chip guardano ogni opcode proveniente dal bus. Per fare questo, entrambi i chip hanno lo stesso BIU (bus interface unit) e il BIU 8086 invia il segnale di status della sua coda di prefetch al BIU 8087. Questo assicura che entrambi i processori decodificano sempre le stesse istruzioni in parallelo. Dato che tutte le istruzioni del coprocessore partono con la sequenza di bit 11011, è facile per la 8087 ignorare tutte le altre istruzioni.

Allo stesso modo la CPU ignora tutte le istruzioni del coprocessore a meno che queste accedono alla memoria. In questo caso, la CPU calcola gli indirizzi dell'LSB (least significant byte) degli operandi della memoria e fa una lettura fittizia. L'8087 allora prende i dati dai dati del

bus. Se c'è più di un accesso alla memoria è necessario caricare un operando di memoria e l'8087 richiede il bus dalla CPU che genera gli indirizzi consecutivi dei byte di operandi e li recupera dai dati del bus.

Dopo aver completato questa operazione, l'8087 ridà il controllo del bus alla CPU.

Dato che l'8087 e la CPU sono collegate allo stesso bus sincrono, devono girare alla stessa velocità. Questo significa che con l'8087 sono possibili solo operazioni sincronizzate con la CPU.

ESISTE UN COPROCESSORE ANALOGO 68K?

Non ci sono coprocessori analoghi

2) 80287

L'interfaccia di questo coprocessore è completamente diverso dall'8087. Dato che l'80286 implementa la protezione della memoria grazie all'MMU basata sulla segmentazione, sarebbe stato troppo costoso duplicare l'intera logica della protezione della memoria sul coprocessore, che un interfaccia simile alla 8087 avrebbe richiesto. Invece, in un sistema 80286/80287, la CPU recupera e memorizza tutti gli opcode e gli operandi per il coprocessore. L'informazione è poi passata attraverso le porte CPU F8H-FFh.

La combinazione 8086/8087 può essere considerata come una cooperazione tra le parti con uguali diritti, mentre l'80286/80287 è più una relazione master-slave.

Questo tipo di relazione rende più facile la sincronizzazione, dato che il flusso completo di dati e istruzioni al coprocessore viaggia attraverso la CPU. Prima di eseguire la maggior parte delle istruzioni coprocessore, l'80286 testa il suo /BUSY pin, il quale è legato al coprocessore 287 e segnala se l'80287 sta ancora eseguendo una istruzione coprocessore precedente o ha incontrato un eccezione.

L'80286 allora aspetta fino a che il segnale /BUSY si sposti su "low" prima di caricare la prossima istruzione coprocessore nella 80287.

Pertanto non è richiesta una istruzione WAIT prima di ogni operazione del coprocessore. Questi WAIT sono ammissibili, ma non necessari nei programmi per 80287.

L'unità di esecuzione dell'80287 è identica a quella dell'8087. Comunque, a causa di un addizionale overhead dell'interfaccia CPU/coprocessore 80287, una combinazione 80286/80287 a 8 Mhz, può avere prestazione floating point inferiori rispetto ad una combinazione 8086/8087 che gira alla stessa velocità. Le vecchie schede 286 venivano spesso configurate per far girare il coprocessore a 2/3 della velocità della CPU facendo uso dell'abilità del coprocessore di girare in modo asincrono.

Intel 80287XL, Cyrix 82S87 e IIT 2C87 contengono internamente un coprocessore 387, ma sono pin compatibili con l'originale 287. Questi chip dividono il clock di sistema interno per 2, all'opposto dei 3 dell'originale 80287. Dato che l'80286 può dividere il clock di sistema per 2, questi coprocessori giravano sempre in modo sincrono rispetto alla CPU, ma potevano anche girare in modo asincrono.

E LA CONTROPARTE AMIGA?

Probabilmente possiamo comprare questa FPU alla 68881/2.

Vediamo in dettaglio come funzionano.

MOTOROLA 68881

Si tratta di una unità esterna di calcolo in virgola mobile che accelera moltissimo il funzionamento dei software che prevedono il suo utilizzo. Questa unità permette un certo livello di lavoro in parallelo e da la possibilità di eseguire certe operazioni mentre la CPU ne esegue delle altre. La gestione di questo coprocessore è possibile sia per l'uso di una istruzione di interfacciamento integrata nell'MC68020 e MC68030 o attraverso una emulazione software che è stata utilizzata dalle prime schede acceleratrici per Amiga dato che l'interfaccia raccomandata, quella del 68020 e 68030, è supportata solo da questo processori

L'MC68881 può funzionare in modo asincrono rispetto al clock della CPU e significa che non è obbligatorio far girare l'FPU alla stessa frequenza del processore.

Con questa funziona asincrona è possibile far girare una FPU più veloce per dare un effetto turbo alle operazioni in virgola mobile.

L'MC68881 supporta le caratteristiche IEEE in virgola mobile così come la maggior parte delle funzioni aritmetiche e trascendentali.Questa unità supporta la precisione semplice a 32 bit, la doppia precisione a 64 bit e la precisione estesa a 80 bit.

Gli MC68881 si trovano negli Amiga a frequenze tra i 12 e 33 Mhz. Nel 1985 questo coprocessore era presente sulle schede CBM A2620 a 14,28 Mhz, in tutte le prime schede A2630 a 25 Mhz e su A3000 a 16 Mhz.

Questo coprocessore è stato rimpiazzato dall'MC68882 che è compatibile a livello di piedinatura e bisogna solo fare degli aggiornamenti software.

3) 80387

L'interfaccia del coprocessore in un sistema 80386/80387 è molto simile a quello che si trova nei sistemi 80286/80287. Per prevenire la corruzione del contenuto del coprocessore da errori di programmazione, vengono usate le porte 800000F8h-800000FFh, le quali non sono accessibili ai programmi. L'interfaccia CPU/Coprocessore è stata ottimizzata e usa un trasferimento completo a 32-bit; l'overhead dell'interfaccia è stata ridotta a 14-20 cicli di clock. Per alcune operazioni sui 'cloni' 387 che prendono meno di 16 cicli di clock da completare, questo overhead limita effettivamente il tasso di esecuzione delle istruzioni coprocessore. L'unica possibile soluzione per avere buone prestazioni floating-point era di integrare la CPU e il coprocessore nello stesso chip che fu poi fatta dalla Intel con la CPU 80486.

La FPU nel 486 beneficia anche delle istruzioni pipeline e della cache integrata nel chip.

E LA CONTROPARTE AMIGA?

Motorola MC68882

Questa unità supporta le stesse interfacce e le operazione del precedente modello, ma con dei miglioramenti interni. Le istruzioni di numerose operazioni sono state ottimizzate per accelerare i tempi di risposta e la possibilità di effettuare più operazioni in virgola mobile simultaneamente. In più è composto da due parti: l'unità di conversione (che trasforma i dati in un formato interno a 80 bit) e l'unità aritmetica. In generale questa FPU lavora ad una velocità 1,5 volte meglio dell'MC68881 a parità di frequenza.

L'MC68882 si trova ad una velocità tra i 16 e 50 Mhz e questo dipende dal tipo di scheda che l'utilizza. Questa FPU si trovava nella maggior parte delle schede acceleratrici come ad esempio nell'A3000 a 25 Mhz o nell'A2630.

L'MC68882 si presenta come PGA a 68 pin

MOTOROLA MC68040/MC68060

I processori MC68040 e MC68060 integrano una FPU nello stesso processore. Questa FPU è una versione migliorata dell'MC68882 eliminando le funzioni trascendentali (sin, cos, ecc…) e complesse presenti nelle istruzioni dei modelli precedenti. La natura ottimizzata delle istruzioni di queste FPU permettono, in emulazione delle funzioni assenti, una esecuzione molto più veloce rispetto all'MC68882 in tutte le operazioni.

4) COPROCESSORI EMULATI

In assenza del coprocessore si possono emulare i calcoli floating-point via software con dei programmi che simulano le operazioni del coprocessore.

Due sono gli approcci più comuni per l'emulatore 80x87.

Il primo metodo fa uso del fatto che tutte le istruzioni coprocessore partono con le stesse cinque sequenze di bit 11011. Così il primo byte di una istruzione coprocessore sarà sempre nell'intervallo esadecimale D8-DF. In più, le istruzione del coprocessore sono di solito precedute da una istruzione WAIT (opcode 9Bh) che è un byte lungo.

Un approccio comune è quello di rimpiazzare l'istruzione WAIT e il primo byte dell'istruzione del coprocessore con una tra le 8 istruzioni di interrupt; i rimanenti byte dell'istruzione del coprocessore rimangono invariate. In questo metodo di emulazione vengono usati gli interrupt esadecimali da 34 a 3B.

Il secondo metodo di emulazione è disponibile solo macchine con 286/386/486.

Se viene attivato il bit di emulazione in queste macchine, il processore genererà un interrupt 7 anche se trova un'istruzione coprocessore. Il vettore per questo interrupt sarà stato impostato al punto in cui il pacchetto di emulazione decodifica l'istruzione e esegue l'operazione desiderata. Questo approccio ha il vantaggio che l'emulatore non deve essere incluso nel codice del programma, ma può essere caricato una volta (come un TSR o una driver per una periferica) e poi usato da ogni programma che richiede un coprocessore. L'emulazione attraverso l'interrupt 7 è trasparente il che significa che i programmi che contengono istruzione del coprocessore vengono eseguiti come se c'è un coprocessore, anche se più lentamente.

Questo approccio è stato usato per esempio dall'emulatore gratuito EM87, dallo shareware Q387 e dall'emulatore commerciale Franke387.

Ogni programma che richiede un coprocessore per girare come AutoCAD viene ingannato per fargli credere che il coprocessore è presente con l'emulatore che usa INT 7.

I sistemi operativi OS/2 2.0 e Windows 3.1 hanno un emulatore integrato che usa INT 7 e che viene automaticamente attivato se non trova un coprocessore anche se l'emulatore in Windows non sembra molto veloce. Alcune istruzioni non sono supportate da Windows 3.1 come la FBSTP.

5) DETTAGLI CPU+FPU X86

INTEL 8087

Questo è stato il primo coprocessore che Intel rese disponibile per la famiglia 80x86 e che fu introdotto nel 1980 e perciò non aveva la piena compatibilità con lo standard Floating-Point IEEE-754.

Questo coprocessore è implementato usando NMOS. Si presenta come un CERDIP da 40-pin (ceramic dual inline package). E' disponibile a 5 Mhz, 8 Mhz (8087-2) e nella versione a 10 Mhz (8087-1). Il consumo energetico è di 2400 mw al massimo sforzo.

INTEL 80187

L'80187 era un nuovo coprocessore progettato per supportare il controller embedded 80C186 (una versione CMOS della CPU 80186). Fu introdotto nel 1989 e implementa il set completo delle istruzioni 80387. Era disponibile come CERDIP a 40 pin e come PLCC (plastic leaded chip carrier) a 44 pin a 12,5 e 16 Mhz.

INTEL 80287

Questo era il coprocessore originale per l'80286 e fu introdotto nel 1983. Usa la stessa unità di esecuzione dell'8087 e perciò ha la sua stessa velocità e in alcune operazione è anche più lento. Come nell'8087 non ha la piena compatibilità con lo standard Floating-Point IEEE-754 che fu rilasciato nel 1985.

L'80287 fu progettato con tecnologia NMOS e si presentava come CERDIP a 40 pin a 6, 8 e 10 Mhz. Il consumo energetico può essere stimato sui 2400 mw sotto massimo sforzo.

L'80287 fu rimpiazzato dal suo successore, molto più veloce, Intel 287XL che fu introdotto nel 1990.

INTEL 80287XL

Questo chip rappresenta la seconda generazione di coprocessori 287 di Intel.

Dato che è basato sul core del coprocessore 80387 ha la piena compatibilità con lo standard IEEE-754. Intel dichiarava che le prestazioni erano del 50% superiori al vecchio 80287 nei tipici benchmark come il Whetstone.

Il 287XL fu progettato in CMOS e perciò consumava meno corrente rispetto alla vecchia tecnologia NMOS. A 12,5 Mhz, il consumo era al massimo di 675 mw, circa ¼ rispetto al vecchio 80287.

Era disponibile come CERDIP a 40 pin o come PLCC a 44 pin.

La velocità di questo coprocessore era fino a 12,5 Mhz.

AMD 80C287

Questo chip, progettato da Advanced Micro Devices (AMD), è l'esatta copia del vecchio Intel 80287 e il primo coprocessore AMD nel 1989.

Questo coprocessore contiene l'originale micro codice dell'80287 e perciò compatibili al 100% con esso. Come viene indicato nel nome, questo processore è stato progettato con tecnologia CMOS e quindi a basso consumo rispetto all'Intel 80287. A 12,5 Mhz consumava meno dell'Intel 287XL. Di questo processore esiste una variante, 80EC287 che usa un sistema intelligente per risparmiare corrente arrivando a consumi inferiori all'80C287.

L'80EC287 fu usato poi come CPU per i Laptop.

L'AMD 80C287 era disponibile a 10, 12 e 16 Mhz ed era disponibile come CERDIP a 40 pin o PLCC a 44 pin.

CYRIX 82S87

Questo chip fu sviluppato da Cyrix e compatibile con l'80287 di Intel ed è stato reso disponibile nel 1991. E' completamente compatibile con lo standard IEEE-754 e ha lo stesso set istruzione dell'Intel 80387. Implementa le funzioni trascendentali con lo stesso grado di accuratezza e la velocità è superiore al Cyrix 83D87. Il Cyrix 82S87 è il più veloce e più accurato coprocessore compatibile 287 fino ad allora disponibile.

L'82S87 fu progettato con tecnologia CMOS e poteva girare da 6 a 20 Mhz.

Il processore era disponibile come DIP 40 pin o PLCC 44 pin.

IIT 2C87

Questo chip era il primo clone 287 disponibile. Introdotto nel mercato nel 1989 ha la stessa velocità dell'Intel 287XL. Il 2C87 implementa lo stesso set istruzioni dell'80387, ma non compatibile con lo standard IEEE-754.

Il 2C87 aveva delle funzioni extra che non avevano gli altri chip 287. Aveva 24 registri floating-point organizzati in tre banchi di registri. Istruzioni aggiuntive (FSBP0, FSBP1, FSBP2) permettono lo scambio da un banco all'altro.

Lo scopo principale di questi banchi di registri era di aiutare la quarta istruzione addizionale che aveva il 2C87 (F4x4) che faceva una completa moltiplicazione di una matrice 4x4 grazie ad un vettore 4x1, un operazione comune nelle applicazioni grafiche 3D.

Il chip era disponibile come velocità fino a 20 Mhz e implementava un'avanzata tecnologia CMOS per un consumo tipico sui 500 mw.

80387

Questo chip rappresentava la prima generazione di coprocessori progettati specificatamente per le CPU Intel 80386. Fu introdotto nel 1986, circa un anno e mezzo dopo che fu lanciato sul mercato la CPU 80386. I primi sistemi 386 erano equipaggiati con entrambi i socket 80287 e 80387. L'80386 funziona con un 80287, ma le prestazioni erano pesantemente inadeguate per tale sistema.

L'80387 fu rimpiazzata dalla più potente e veloce 80387DX che era il 20% più veloce ed era disponibile in ceramica PGA a 68 pin e progettata con tecnologia CHMOS III.

INTEL 387DX

L'80387DX è il coprocessore di seconda generazione di Intel; fu velocemente introdotto per rimpiazzare l'80387 nel 1989. Questa versione è stata progettata con una tecnologia CMOS avanzata per arrivare ad una velocità di 33 Mhz (il vecchio 80387 non poteva superare i 25 Mhz).

L’80387DX era il 20% più veloce dell’80387 a parità di clock.

INTEL 387SX

Questo è il coprocessore accoppiato con la CPU Intel 386SX. Questa CPU è un Intel 80386 con un percorso dati a 16 bit anziché 32 bit. Questo riduceva il costo per costruire un sistema basato su 386SX.

Lo scopo principale di questo processore era quello di rimpiazzare la CPU 80286.

A causa del percorso dati a 16 bit, il 386SX è più lento di un 386DX e le prestazioni sono come quelle di un 80286 a parità di clock per le applicazioni a 16 bit.

Il 386SX internamente però è un completo 386 e ha la possibilità di far girare applicazioni 32 bit e supporta il modo 8086 virtuale (usato da Windows ad esempio).

Il 386SX ha tutte le caratteristiche dell’80387, inclusa l’abilità di far girare in modo asincrono le operazioni di CPU e coprocessore. A causa del percorso dati a 16 bit tra la CPU e il coprocessore, la 387SX è un po’ più lenta che una 80387 alla stessa frequenza di clock. In più, la 387SX è basata sul core dell’originale 80387 e quindi ancora più lenta del 387DX che rappresenta la seconda generazione.

L’80387SX si presenta come PLCC 68 pin a 16 e 20 Mhz.

INTEL RAPIDCAD

Il RapidCAD non è un coprocessore, nel vero senso del termine, anche se veniva pubblicizzato come tale. Questo chip è un completo sostituto di una CPU 80386: in pratica si tratta di una CPU Intel 486DX senza la cache interna e con una piedinatura standard 386. RapidCAD viene rilasciato come un set di due chip.

RapidCAD-1 si inserisce in un socket 386 e contiene la CPU e la FPU.

RapidCAD-2 si inserisce nel socket coprocessore 387 e contiene un semplice PAL che ha il solo scopo di generare un segnale FERR normalmente generato da un coprocessore.

Il set istruzione di RapidCAD è compatibile con il 386, quindi non ha nessuna nuova istruzione specifica per 486 come BSWAP. Tuttavia, dato che il core della CPU del RapidCAD è molto simile al core della CPU 486, l'esecuzione della maggior parte delle istruzioni register-to-register vengono eseguite nello stesso numero di cicli di clock del 486.

RapidCAD usa l'interfacciamento al bus standard 386 che fanno si che le istruzioni che accedono alla memoria sono eseguite alla stessa velocità del 386.

Le operazioni intere sul RapidCAD sono limitate dalla lenta banda della memoria che fornisce questa interfaccia (2 cicli di clock per cicli bus) e la mancanza di una cache interna. Le istruzioni CPU sono eseguite più velocemente prima che possano essere recuperate dalla memoria anche con una grande e veloce cache esterna.

Nelle prestazioni sugli interi il RapidCAD è il 35% più veloce di un 386, ma questa valutazione deriva da applicazioni che usano operazioni register-to-register e pochi accessi alla memoria.

Mentre le istruzioni CPU e Intere vengono eseguite in un ciclo di clock sul RapidCAD, le operazioni Floating-Point vengono eseguite a più di sette cicli di clock. Queste sono perciò abbastanza lente dovute alla bassa banda dell'interfaccia del bus del 386.

RapidCAD è disponibile a 25 e 33 Mhz.

RapidCAD-1 si presenta come PGA 132 pin, come l'80386 e il RapidCAD-2 si presenta come PGA 68 pin come l'80387.

Application	Time w/ 387DX	Time w/ RapidCAD	Speedup
AutoCAD 11	52 sec	32 sec	63%
AutoShade/Renderman	180 sec	108 sec	67%
Mathematica(Windows)	139 sec	103 sec	35%
SPSS/PC+ 4.01	17 sec	14 sec	21%

INTEL 486DX

L'intel 486DX non è un unico coprocessore. Questo chip fu introdotto la prima volta da Intel nel 1989 e combina la CPU (una fimplementazione pesantemente pipeline dell'architettura 386) con una FPU 387 migliorata e 8 kb di cache codice/data unificata sul chip. (questa è una descrizione molto semplificata, ne riparleremo forse in seguito). Il 486DX è più veloce negli interi dalle 2 alle 3 volte rispetto ad un 386 alla stessa frequenza di clock, mentre la FPU è dalle 3 alle 4 volte più veloce rispetto alla 387DX alla stessa frequenza di clock. Dato che la FPU è nello stesso chip con la CPU, non c'è il considerevole hoverhead di comunicazione tra CPU e FPU come tra il 386 e la 387, lasciando che le istruzioni FPU girino alla massima velocità permessa dall'implementazione. La FPU ha anche il vantaggio di sfruttare la cache sul chip e l'unita di esecuzione altamente pipeline. E' ancora esistente sul 486 l'esecuzione in contemporanea di istruzioni CPU e FPU come in tutti gli x86 e x87, ma alcune istruzione FPU come la FSIN non sono più simultanee con le istruzioni della CPU.

Oltre le sue alte prestazioni, la FPU del 486 ha migliorate le funzioni trascendentali rispetto al coprocessore 387DX. Per raggiungere una migliore latenza interrupt, le istruzioni FPU con lunghi tempi di esecuzione vengono interrotte se interviene un interrupt durante la loro esecuzione.

Il 486DX si presenta come PGA a 186 pin. E' disponibile a 25 e 33 Mhz. Dalla fine del 1991 era disponibile una versione a 50 Mhz prodotta con tecnologia CHMOS V.

INTEL 486DX2

Il 486DX2 rappresenta l'ultima generazione delle cpu Intel. Il suffisso "Dx2" indica che queste cpu sono versioni doppio clock. Un nomale 486DX opera alla frequenza di clock indicato. Il 486DX2 invece genera

un nuovo segnale di clock da quello fornito da un PLL (Phase locked loop). Nel DX2, questo segnale di clock, ha la frequenza doppia. Tutte le parti internet del 486DX2 (cache, CPU e FPU) girano alla frequenza più alta; solo il bus gira alla velocità normale (non raddoppiata). Usando questa tecnica, un Intel 486DX2-50 può girare su una scheda madre normale costruita per girare a 25 Mhz. Dato che le schede madri che girano a 50 Mhz sono più difficili da progettare e costruire rispetto a quelle a 25 Mhz, questo rende un sistema 486DX2-50 più economico che un equivalente sistema 486DX-50.

Per tutte le operazioni che non accedono alle risorse esterne al chip (operazioni sui registri), un 486DX2-50 ha le stesse prestazioni di una 486DX-50 e due volte più veloce che un 486DX-25. Tuttavia, dato che la memoria principale di un 486DX2-50 opera ancora a 25 Mhz, tutte le istruzioni che riguardano la memoria hanno degli accessi più lenti rispetto ad un 486DX-50 dove la memoria gira (presumibilmente) a 50 Mhz. La cache interna viene in aiuto in questo caso, ma in linea generale le prestazioni di un 486DX2-50 sono leggermente più lente rispetto ad un 486DX-50.

La documentazione Intel mostra una lieve perdita di prestazioni, anche se dipende molto dal tipo di applicazioni.

La cosa più interessante sul 486DX2 è il fatto che rende molto facile l'aggiornamento di un sistema con un 486 a 25 e 33 Mhz dato che il 486DX2 come piedinatura è completamente compatibile con il 486DX: è sufficiente staccare il 486DX e sostituirlo con un 486DX2.

INTEL 487SX

Il 487SX è un coprocessore matematico da usare sui sistemi con 486SX. Questo processore è praticamente un 486DX senza FPU. In origine Intel vendeva dei 486DXS con l'FPU difettosa come 486SXS, ma in seguito ha rimosso completamente l'FPU.

Intel introduce questo 486SX nel 1991 per cercare di recuperare mercato dai sistemi basati su 386 e per il grande successo di AMD con i suoi Am386 (AMD prese più del 40% del mercato 386 grazie ad alcune caratteristiche superiori come l'alta frequenza di clock, meno

consumo elettrico, buona progettazione e disponibilità di versione a 3V). Un 486SX a 20 Mhz era leggermente meno veloce di un AMD Am386 a 40 Mhz negli interi.

Per aggiungere le funzione Floating-Point ad un sistema 486SX, il modo più semplice era quello di sostituirlo con un 486DX che include l'FPU integrata. Intel però ha reso più difficile questa operazione, rendendo il 486SX leggermente diverso come piedinatura. Dato che solo tre pin sono assegnati in modo diverso, i produttori di schede madri dovevano quindi far uscire schede madri compatibili da un 486SX-20 fino ad un 486DX2-50. Un set di tre jumper assicurano il corretto segnale ai pin cambiati per ogni tipo di CPU. Per aggiornare un sistema 486SX senza questa funzione è necessario acquistare una 487SX e installarla nel "Performance Upgrade Socket" presente nei principali sistemi.

Una volta installato il 487SX, questo viene subito visto dal sistema come se fosse un normale 486DX anche con la lieve differenza di piedinatura. Questa scelta di Intel serviva per aggiungere l'FPU dei 486DX ai sistemi 486SX.

La CPU e l'FPU devono essere sullo stesso chip perché altrimenti la FPU non può usare la cache interna della CPU e non ci sarebbe il considerevole overhead di comunicazione tra CPU e FPU annullando l'accelerazione delle principali operazioni matematiche rispetto alla 387.

I 486SX, 487SX e 486DX non sono compatibili a livelli di piedinatura solo per una ragione di marketing.

Un altro modo per aggiornare un 486SX era di usare il chip Overdrive che è un 487SX, ma con il clock interno doppio e raddoppia le prestazioni di un sistema basato su 486SX/487SX.

Aggiungendo il 487SX, il 486SX viene letteralmente disattivato, quindi il 486SX andava rimosso dopo aver inserito il 487SX. Dato che lo spegnimento del 486SX era logico e non elettrico, il 486SX usava ancora l'alimentazione se usato con il 487SX anche se non era operativo.

Come per il 486SX anche il 487SX era disponibile a 20 e 25 Mhz e si presentava come PGA (pin grid array) a 169 pin.

WEITEK 1167

Questo coprocessore matematico era il predecessore dell'Weitek Abacus 3167. Era un piccolo circuito su scheda stampata con tre chip montati sopra. Al contrario della Weitek 3167, la 1167 non aveva le istruzione per la radice quadrata e questa funzione era calcolata da una sotto istruzione della libreria delle funzioni trascendentali di Weitek. Come prestazioni era leggermente più lenta della Weitek 3167.

WEITEK 3167

La 3167 fu introdotta da Weitek nel 1989 e come floating point aveva le prestazioni più veloci mai viste su un sistema 386 in quel periodo.

La 3167 non era un vero coprocessore, strettamente parlando, ma piuttosto una periferica di memoria mappata. L'architettura della 3167 era ottimizzata per la velocità se era possibile. Anche se usa l'interfaccia di memoria più veloce per la CPU (l'80x87 usava le porte IO), non supporta le principali caratteristiche dei coprocessori 80x87, permette che tutte le risorse del chip siano concentrate sulla velocità di esecuzione delle operazioni aritmetiche.

Nei benchmark comparativi, la Weitek 3167 è circa 2,5 volte più veloce di un Intel 387DX.

La Weitek Abacus 3167 si presenta come PGA a 121 pin e si inserisce in un socket EMC (un socket presente su molti sistemi 386) e non si può inserire nel normale socket per coprocessore 387.

Weitek 4167

La 4617 è un coprocessore di memoria mappata che ha la stessa architettura della 3167; è progettata per essere usata sui sistemi 486 con eccellenti prestazioni floating point. Esegue le istruzione coprocessore tre volte più velocemente della 3167. Anche se è l'80%

più veloce di un Intel 486 in alcuni benchmark, le prestazioni medie per le vere applicazioni sono circa del 10% in più. L'introduzione del 486DX2 ha più o meno reso obsoleta la 4167 perché le cpu DX2 anche se hanno le stesse prestazioni, hanno però tutte le istruzione 80x87 che le waitek non hanno.

BENCHMARK WEITEK

	Single Prec.		Double Prec.		Double Prec.	
	3167	4167	3167	4167	387	486
Linpack MFLOPS	1.8	5.0	0.8	3.2	0.4	1.6
Whetstone kWhet/sec	7470	22700	4900	14000	3290	12300

Comparison of floating-point performance [30,32]

single-precision

	Weitek 4167-33	Intel 486-33	Intel 486DX2-66
Linpack MFLOPS	5.0	1.8	3.5
Whetstones kWhet/sec	22700	12700	25500

double-precision

	Weitek 4167-33	Intel 486-33	Intel 486DX2-66
LINPACK MFLOPS	3.5	1.6	3.1
kWhetstones/sec	14000	12300	24700

BENCHMARK E RISULTATI

Questi sono i principali benchmark per testare le varie FPU:

- PEAKFLOP è un calcolo di un frattale
- TRANSFORM è un array di 8191 vettori con una trasformazione 3D della matrice (una matrice 4x4)

- LAWRENCE LIVERMORE LOOPS (LLL) che è un set di kernel presi da vere applicazioni che fanno uso pesante del floating point
- LINPACK è un famoso benchmark floating point es è un ottimo esercizio per la memoria di sistema
- WHETSTONE, un benchmark sintetico basato su statistiche raccolte dall'uso di certi controlli e strutture di dati in programmi scritti in un alto livello di programmazione
- SAVAGE che fa un test delle funzioni trascendentali

33.3 MHz	PEAKFLOP MFLOPS	TRNSFORM MFLOPS	LLL MFLOPS	Linpack MFLOPS	Whetstone kWhet/sec	Savage Func/sec	
Intel 386DX WITH:							
EM87 emulator	0.0070	0.0040	0.0050	0.0050	26	418	##
Franke387 emu.	0.0307	0.0246	0.0194	0.0179	137	3335	@@
TP/MS-FORT emu	0.0263	0.0227	0.0167	0.0158	133	3160	%%
Q387 emulator	0.0768	0.0583	0.0285	0.0288	251	7538	((
Intel 387DX	0.7647	0.6004	0.3283	0.2676	2046	43860	
ULSI 83C87	1.0097	0.6609	0.3239	0.2598	2089	47431	
IIT 3C87	0.8455	0.5957	0.3198	0.2646	2203	49020	
IIT 3C87,4X4	0.8455	1.4334	0.3198	0.2646	2203	49020	$$
ULSI DX/DLC	1.0097	0.6628	0.3228	0.2496	2144	51107	
C&T 38700	0.9455	0.6907	0.3338	0.2700	2376	62565	
Cyrix 387+	0.9286	0.6806	0.3293	0.2669	2435	66890	
Cyrix EMC87	1.0400	0.6628	0.3352	0.2808	2540	71685	//
Intel RapidCAD	1.8572	1.5798	0.6072	0.4533	3953	72464	
Intel 486DX	2.0800	1.7779	0.9387	0.6682	5143	82192	

40 MHz	PEAKFLOP MFLOPS	TRNSFORM MFLOPS	LLL MFLOPS	Linpack MFLOPS	Whetstone kWhet/sec	Savage Func/sec	
Intel 386DX WITH:							
EM87 emulator	0.0084	0.0080	0.0060	0.0060	31	502	##
Franke387 emu.	0.0369	0.0295	0.0233	0.0215	164	4002	@@
TP/MS-FORT emu	0.0316	0.0273	0.0200	0.0190	160	3794	%%
Q387 emulator	0.0922	0.0700	0.0342	0.0345	302	9053	((
Intel 387DX	0.9204	0.7212	0.3932	0.3211	2428	52677	
ULSI 83C87	1.2093	0.7936	0.3890	0.3120	2528	56926	
IIT 3C87	1.0196	0.7145	0.3834	0.3179	2663	58766	
IIT 3C87,4x4	1.0196	1.7244	0.3834	0.3179	2663	58766	$$
ULSI DX/DLC	1.2093	0.7935	0.3880	0.3000	2586	61287	
C&T 38700	1.0722	0.7908	0.4007	0.3222	2837	74906	
Cyrix 387+	1.1305	0.8162	0.3945	0.3208	2946	80322	
Cyrix EMC87	1.2381	0.7963	0.4025	0.3324	3061	86083	//
Intel RapidCAD	2.2128	1.8931	0.7377	0.5432	4810	86957	
Intel 486DX	2.4762	2.1335	1.1110	0.8204	6195	98522	

	PEAKFLOP MFLOPS	TRNSFORM MFLOPS	LLL MFLOPS	Linpack MFLOPS	Whetstone kWhet/sec	Savage Func/sec	
Pentium-66	11.5557	8.1900	2.5971	2.2150	44000	337079	))
i486DX2-66	4.1601	3.4227	1.6531	1.3010	10655	163934	
i486DX2-50	3.0589	2.6665	1.2537	0.9744	7962	123203	
Cyrix EMC87-25	0.7800	0.4971	0.2514	0.2106	1905	53763	[[
IIT 3C87-25	0.6341	1.0751	0.2399	0.1985	1652	36765	$$,[[
IIT 3C87-25	0.6341	0.4467	0.2399	0.1985	1652	36765	[[
i387, 20 MHz	0.2253	0.3271	0.1434	0.1171	952	21739	++
i387DX, 20 MHz	0.3567	0.4444	0.1484	0.1161	1034	24155	&&
i80287, 5 MHz	0.0281	0.0310	0.0242	0.0222	150	3261	!!
i8087,9.54 MHz	0.0636	0.0705	0.0321	0.0219	234	5782	**

Leggenda dei simboli sulla destra della tabella

- EM87 v1.2 è un emulatore PD di un coprocessore che si carica come un TSR. Usa INT 7 emesso da un 80286, 80386 or 486SX.

@@ - Franke87 è un emulatore commerciale di coprocessore 387 che è disponibile anche in versione shareware e in questo test è stata usata la versione shareware 2.4

((- Q387 è un emulatore shareware. Come indica il nome, questo emulatore usa codice specifico per 386 e supporta il set completo delle istruzioni 387. Per questo test è stato usato Q387 v7.04

%% - Questi benchmark girvano su emulatori integrati del TP 6.0 e MS-Fortrain 5.0

$$ - Nel vettore di trasformazione sono state usate delle istruzioni specifiche per il coprocessore 3C87

// - L'EM87 è stato usato nella modalità 387 compatibile e non è stata usata la modalità memoria mappata e dovrebbe essere identico ad un Cyrix 83D87

++ - vecchie schede madri senza chip set, senza cache CPU e 16 MB di memoria

&& - Sistema A, cache della CPU disabilitata attraverso un settaggio avanzato e il tasto turbo settato a metà velocità (20 Mhz)

[[- Sistema A, CPU Intel 386DX e oscillatore cambiato per girare a 25 Mhz

)) – Sistema basato su una scheda madre Intel con 256 kbyte di cache CPU e 16 MB di memoria con Bios AMI.

!! – 80386@20 Mhz / Intel 80387 @5 Mh, no CPU cache, 4 MB ram a causa della veloce cpu usata.

** - 8086/8087 con 640 Kb di RAM

ESEMPI DI BENCHMARK DA COMPILARE

```
{$N+,E+}
PROGRAM PCtrl;

VAR B,c: EXTENDED;
    Precision, L: WORD;

PROCEDURE SetPrecisionControl (Precision: WORD);
(* This procedure sets the internal precision of the NDP. Available *)
(* precision values:  0  -  24 bits (SINGLE)                        *)
(*                    1  -  n.a. (mapped to single)                 *)
(*                    2  -  53 bits (DOUBLE)                        *)
(*                    3  -  64 bits (EXTENDED)                      *)

VAR CtrlWord: WORD;

BEGIN {SetPrecisionCtrl}
   IF Precision = 1 THEN
      Precision := 0;
   Precision := Precision SHL 8; { make mask for PC field in ctrl word}
   ASM
      FSTCW    [CtrlWord]        { store NDP control word }
      MOV      AX, [CtrlWord]    { load control word into CPU }
      AND      AX, 0FCFFh        { mask out precision control field }
      OR       AX, [Precision]   { set desired precision in PC field }
      MOV      [CtrlWord], AX    { store new control word }
      FLDCW    [CtrlWord]        { set new precision control in NDP }
   END;
END; {SetPrecisionCtrl}

BEGIN {main}
   FOR Precision := 1 TO 3 DO BEGIN
      B := 1.2345678901234567890;
      SetPrecisionControl (Precision);
      FOR L := 1 TO 20 DO BEGIN
         B := Sqrt (B);
      END;
      FOR L := 1 TO 20 DO BEGIN
         B := B*B;
      END;
      SetPrecisionControl (3);   { full precision for printout }
      WriteLn (Precision, B:28);
   END;
END.
```

```
{$A+,B-,R-,I-,V-,N+,E+}
PROGRAM PeakFlop;

USES Time;

TYPE ParamRec = RECORD
                   MaxIter, MaxRad, YMax, XMax: WORD;
                   Qmax, Qmin, Pmax, Pmin: DOUBLE;
                   FastMod: WORD;
                   PlotFkt: POINTER;
                   FLOPS:LONGINT;
                END;

VAR Param: ParamRec;
    Start: LONGINT;


{$L APFELM4.OBJ}

PROCEDURE Apple87 (VAR Param: ParamRec);     EXTERNAL;


BEGIN
   WITH Param DO BEGIN
      MaxIter:= 50;
      MaxRad := 30;
      YMax   := 30;
      XMax   := 30;
      Pmin   :=-2.1;
      Pmax   := 1.1;
      Qmin   :=-1.2;
      Qmax   := 1.2;
      FastMod:= Word (FALSE);
      PlotFkt:= NIL;
      Flops  := 0;
   END;
   Start := Clock;
   Apple87 (Param);         { executes 104002 FLOP }
   Start := Clock - Start;  { elapsed time in milliseconds }
   WriteLn ('Peak-MFLOPS: ', 104.002 / Start);
END.
```