

În multe cazuri, este mai eficientă realizarea transferurilor direct între periferic și memorie. În aceste cazuri, UCP acordă unui modul de I/E permisiunea de citire din sau scriere în memorie, astfel că transferurile pot avea loc fără intervenției UCP. Această operație se numește *acces direct la memorie* (DMA).

4.5. Procesoarele Intel

Firma *Intel* a fost cea care a proiectat primul microprocesor, în anul 1969. Acesta a fost microprocesorul 4004, de 4 biți. Au urmat apoi microprocesoarele de 8 biți 8080 și 8085. Primul procesor al familiei 80x86 (numită și familia cu arhitectura *Intel – Intel Architecture*) a fost însă procesorul 8086, care a apărut în 1978, fiind urmat în scurt timp de o versiune mai ieftină pentru sisteme mai simple, procesorul 8088. Programele scrise pentru aceste procesoare pot fi executate fără modificări și pe ultimele procesoare ale familiei 80x86.

4.5.1. Prezentare generală a procesoarelor Intel

8086 și 8088

Procesorul 8086 are registre de 16 biți, o magistrală externă de date de 16 biți, și o magistrală de adrese de 20 de biți, astfel că spațiul de adresare este de maxim 1 MB. Procesorul 8088 este similar, cu excepția faptului că magistrala externă de date este de 8 biți. Aceste procesoare au introdus conceptul de segmentare a memoriei: memoria este împărțită în zone numite segmente de maxim 64 KB, iar cele patru registre de segment pot păstra adresele de bază ale segmentelor active. Prin utilizarea registrelor de segment poate fi adresat un spațiu de memorie de până la 256 KB fără modificarea conținutului acestor registre, fiind disponibil un spațiu total de memorie de 1 MB. Aceste procesoare pot funcționa numai în *modul real*, care este un mod uni-proces, în care se execută un singur proces (program sau *task*) la un moment dat. Din punct de vedere al programatorului, cele două procesoare sunt identice, cu excepția faptului că 8086 va prelucra datele într-un mod mai eficient dacă acestea sunt aliniate la adrese de cuvânt (adrese pare).

80186 și 80188

Procesoarele 80186 și 80188 sunt versiuni îmbunătățite ale procesoarelor 8086, respectiv 8088. Au fost adăugate noi instrucțiuni și anumite instrucțiuni vechi au fost optimizate pentru creșterea vitezei. Aceste procesoare conțin în aceeași capsulă și circuite suplimentare: un generator de tact, un controler DMA cu două canale, trei numărătoare/temporizatoare programabile, și un controler de întrerupere, care poate gestiona patru surse externe de întrerupere.

80286

Procesorul 80286 poate funcționa în *modul real* al procesoarelor precedente, dar dispune și de un mod de adresare virtuală sau *mod protejat*. Acest mod utilizează conținutul registrelor de segment ca selectori sau pointeri în tabele ale descriptorilor de segment. Descriptorii furnizează adrese de bază de 24 de biți, permițând adresarea unei memorii fizice de până la 16 MB. Procesorul dispune de o unitate de gestiune a memoriei virtuale, cu care poate translata adrese pentru o memorie virtuală de 1 GB în spațiul adreselor fizice de 16 MB. În modul protejat, procesorul poate funcționa în regim multi-proces (*multitasking*), în care pot fi executate mai multe procese în mod concurrent. În acest mod se realizează o comutare prin hardware între procesele care se execută concurrent. Memoria utilizată de fiecare proces este protejată față de acțiunile altor procese. Pentru aceasta procesorul dispune de diferite mecanisme de protecție. Acestea cuprind testarea limitelor segmentelor, existența atributelor de segment care permit doar citirea conținutului unui segment sau doar execuția codului din acel segment, și până la patru nivele de privilegiu pentru protecția codului sistemului de operare de programele de aplicații sau de programele utilizatorului. Existența unor tabele ale descriptorilor locali de segment permite sistemului de operare protejarea între ele a programelor de aplicații sau a programelor utilizatorului. Sunt disponibile instrucțiuni privilegiate suplimentare pentru setarea modului protejat și controlul proceselor multiple.

80386

Procesorul 80386 a introdus în cadrul arhitecturii *Intel* registre de 32 de biți, utilizate atât pentru păstrarea datelor, cât și pentru adresare. Pentru compatibilitate cu procesoarele anterioare, aceste registre s-au obținut prin extinderea registrelor de 16 biți, fiind posibilă utilizarea în continuare a vechilor registre, acestea constituind jumătatea de ordin inferior a registrelor de 32 de biți. A fost introdus un nou mod de funcționare, numit *mod virtual 8086*, care permite execuția mai eficientă a programelor scrise pentru procesoarele 8086 și 8088 pe noul procesor de 32 de biți. Magistrala externă de adrese a fost extinsă la 32 de biți, spațiul adreselor fizice fiind astfel de 4 GB. Spațiul adreselor virtuale este de 64 TB. Dimensiunea fiecărui segment nu mai este limitată la 64 KB, dimensiunea maximă a unui segment fiind de 4 GB. Instrucțiunile originale au fost extinse cu noi forme care utilizează operanzi și adrese de 32 de biți, și au fost introduse instrucțiuni complet noi, ca de exemplu instrucțiuni pentru operații la nivel de bit.

Procesorul 80386 a introdus de asemenea mecanismul de paginare ca metodă de gestiune a memoriei virtuale, dimensiunea fixă de 4 KB a paginilor de memorie asigurând o eficiență mai ridicată comparativ cu utilizarea segmentelor, metoda fiind complet transparentă pentru programele de aplicații, fără o reducere semnificativă a vitezei de execuție. În plus, posibilitatea definirii segmentelor având dimensiunea maximă a spațiului adreselor fizice de 4 GB, împreună cu paginarea, au permis introducerea modelului protejat de adresare liniară în cadrul arhitecturii, în care este necesară

o singură componentă a adresei pentru accesul la întregul spațiu de adresare al memoriei. Acest model este utilizat și de sistemul de operare UNIX.

Procesorul 80386 a fost primul din cadrul familiei 80x86 care a utilizat o formă de prelucrare paralelă. Arhitectura acestui procesor cuprinde un număr de șase unități care funcționează în paralel. Acestea sunt următoarele:

- Unitatea de interfață cu magistrala, care realizează accesul la memorie și la dispozitivele de I/E;
- Unitatea de încărcare a instrucțiunilor, care primește codul instrucțiunilor de la unitatea de interfață cu magistrala și le depune într-o coadă de 16 octeți;
- Unitatea de decodificare a instrucțiunilor, care decodifică codul obiect al instrucțiunilor și generează un microcod pentru execuția acestora;
- Unitatea de execuție, care execută microcodul corespunzător fiecărei instrucțiuni;
- Unitatea de segmentare, care translatează adresele logice în adrese liniare și efectuează testele necesare protecției;
- Unitatea de paginare, care translatează adresele liniare în adrese fizice, efectuează testele necesare protecției paginilor de memorie și conține o memorie *cache* cu informații despre până la 32 de pagini cel mai recent accesate.

80486

Procesorul 80486 a extins posibilitățile de execuție paralelă a instrucțiunilor, în principal prin extinderea unităților de decodificare a instrucțiunilor și de execuție ale procesorului 80386 în cinci etaje de tip *pipeline*, fiecare etaj operând în paralel cu celelalte asupra a cinci instrucțiuni aflate în diferite faze de execuție. Fiecare etaj poate termina prelucrările pe care le efectuează asupra unei instrucțiuni într-o perioadă de tact, astfel încât procesorul 80486 poate executa câte o instrucțiune în fiecare perioadă de tact. Procesorul i s-a adăugat o memorie *cache* de nivel 1 (L1 – *Level 1*) de 8 KB pentru a crește în mod semnificativ procentul instrucțiunilor care se pot executa la rata de o instrucțiune într-o perioadă de tact: astfel, și instrucțiunile cu acces la memorie se pot executa la această rată, dacă operanzii acestora se află în memoria *cache* L1. Procesorul 80486 a fost primul din familia 80x86 la care unitatea de calcul în virgulă mobilă a fost integrată în același circuit cu unitatea centrală. Au fost adăugați de asemenea noi pini și noi instrucțiuni care permit realizarea unor sisteme mai complexe: sisteme multiprocesor și sisteme care conțin o memorie *cache* de nivel 2 (L2 – *Level 2*).

Au fost dezvoltate versiuni ale procesorului 80486 în care au fost incluse facilități pentru reducerea consumului de putere, ca și alte facilități de gestiune a sistemului. Procesorul 80386 SL a fost de fapt primul care conținea asemenea facilități, acestea fiind extinse apoi la procesoarele 80486 SL și 80486 SL *Enhanced*, care au fost utilizate pentru realizarea calculatoarelor “*notebook*”. Una din aceste facilități

este noul mod de gestiune a sistemului (*System Management Mode – SMM*), pentru care s-a prevăzut un pin dedicat de întrerupere. Acest mod permite operații complexe de gestiune a sistemului (ca de exemplu gestiunea puterii consumate de diferitele subsisteme ale calculatorului), într-un mod transparent pentru sistemul de operare și pentru programele de aplicații. Facilitățile numite “*Stop Clock*” și “*Auto Halt Powerdown*” permit funcționarea unității centrale la o frecvență redusă a tactului pentru reducerea puterii consumate, sau chiar oprirea funcționării (cu memorarea stării).

Pentium

Procesorul *Pentium* a adăugat o nouă linie de execuție de tip *pipeline* a instrucțiunilor, pentru a se obține performanțe superscalare. Cele două linii de execuție a instrucțiunilor, numite *u* și *v*, permit execuția a două instrucțiuni pe durata unei perioade de tact. Capacitatea memoriei *cache* L1 a fost de asemenea dublată, fiind alocată 8 KB pentru instrucțiuni și 8 KB pentru date. Memoria *cache* pentru date utilizează protocolul MESI, care permite gestiunea memoriei *cache* atât prin metoda mai eficientă “*write-back*”, cât și prin metoda “*write-through*” utilizată de procesorul 80486. Procesorul *Pentium* utilizează predicția salturilor pentru a crește performanțele construcțiilor care utilizează bucle de program. S-au adăugat extensii pentru a crește eficiența modului virtual 8086, și pentru a permite atât pagini cu dimensiunea de 4 MB, cât și cu dimensiunea de 4 KB. Registrele generale sunt tot de 32 de biți, dar s-au adăugat căi interne de date de 128 și 256 de biți pentru a crește viteza transferurilor interne, iar magistrala externă de date a fost extinsă la 64 de biți. Procesorului i s-a adăugat un controler avansat de întreruperi (*Advanced Programmable Interrupt Controller – APIC*) pentru a permite realizarea sistemelor cu mai multe procesoare *Pentium*, fiind adăugate de asemenea noi pini și un mod special de procesare duală pentru sistemele cu două procesoare.

Pentium Pro

Procesorul *Pentium Pro* este primul din cadrul familiei de procesoare P6. Acest procesor are o arhitectură superscalară îmbunătățită, care permite execuția a trei instrucțiuni într-o perioadă de tact. Procesorul *Pentium Pro*, ca și următoarele procesoare din familia P6, se caracterizează prin execuția dinamică a instrucțiunilor, care constă din analiza fluxului de date, execuția instrucțiunilor într-o altă ordine decât cea secvențială, o predicție îmbunătățită a salturilor și execuția speculativă. Există trei unități de decodificare a instrucțiunilor, care lucrează în paralel pentru decodificarea codului obiect în operații elementare numite micro-operații. Acestea sunt depuse într-un rezervor de instrucțiuni, și pot fi executate într-o ordine diferită de cea secvențială dacă nu există dependențe de date. Execuția se realizează de către cele cinci unități paralele de execuție: două unități pentru numere întregi, două unități pentru numere în virgulă mobilă, și o unitate de interfață cu memoria. Unitatea de retragere (*Retirement Unit*) elimină din rezervorul de instrucțiuni micro-operațiile care au fost

executate, ținând cont de salturile posibile. Pe lângă cele două memorii *cache* L1 de câte 8 KB, prezente și la procesorul *Pentium*, procesorul *Pentium Pro* dispune și de o memorie *cache* L2 de 256 KB, aflată în același circuit cu unitatea centrală, conectată cu aceasta printr-o magistrală dedicată de 64 de biți. Memoria *cache* L1 are porturi duale (două porturi de acces), iar memoria *cache* L2 permite până la 4 accesuri concurente. Magistrala externă de date este orientată pe tranzații, ceea ce înseamnă că fiecare acces este tratat ca o cerere și un răspuns separat, fiind permise mai multe cereri în timp ce se așteaptă un răspuns. Aceste caracteristici de acces paralel la date, împreună cu posibilitățile de execuție paralelă a instrucțiunilor asigură creșterea gradului de utilizare a procesorului și îmbunătățirea performanțelor. Procesorul *Pentium Pro* are o magistrală de adrese extinsă la 36 de biți, astfel încât spațiul adreselor fizice este de până la 64 GB.

Pentium II

Procesorul *Pentium II* se bazează pe arhitectura *Pentium Pro*, la care s-au adăugat extensiile MMX (*Multimedia Extensions*). La acest procesor s-au utilizat noile tehnici de încapsulare numite “*Slot 1*” și “*Slot 2*”. În cazul acestor tehnici, în locul unui soclu se utilizează un conector. Memoria *cache* L2 a fost mutată în afara capsulei procesorului. Atât memoria *cache* L1 pentru date, cât și memoria *cache* L1 pentru instrucțiuni au fost extinse la 16 KB fiecare. Dimensiunea memoriei *cache* L2 poate fi de 256 KB, 512 KB, 1 MB sau 2 MB (numai cu tehnica “*Slot 2*”). Procesorul cu “*Slot 1*” utilizează pentru comunicația cu memoria *cache* L2 o magistrală care funcționează la o frecvență egală cu jumătatea frecvenței de tact a procesorului, în timp ce în cazul procesorului cu “*Slot 2*” această frecvență este egală cu frecvența de tact a procesorului. Procesorul *Pentium II* utilizează diferite stări cu consum redus de putere, ca de exemplu “*AutoHALT*”, “*Stop-Grant*”, “*Sleep*” și “*Deep Sleep*”, pentru reducerea puterii consumate în perioadele de inactivitate.

Pentium III

Acest procesor este ultimul din cadrul familiei P6, și se bazează pe arhitecturile procesoarelor *Pentium Pro* și *Pentium II*. Au fost adăugate 70 de noi instrucțiuni la setul de instrucțiuni existent. Acestea sunt destinate atât unităților funcționale existente la procesoarele precedente, cât și noii unități de calcul în virgulă mobilă de tip SIMD (*Single Instruction, Multiple Data*).

4.5.2. Microarhitectura procesoarelor din familia P6

4.5.2.1. Prezentare generală a microarhitecturii

Microarhitectura procesoarelor din familia P6 este una superscalară cu trei căi. Aceasta înseamnă că, prin utilizarea tehnicilor de prelucrare paralelă, procesorul

poate decodifica și executa în medie trei instrucțiuni în fiecare ciclu de ceas. Pentru a obține această rată de execuție a instrucțiunilor, procesoarele din familia P6 utilizează o cale de date de tip *pipeline* cu 12 etaje, care permite execuția instrucțiunilor într-o ordine diferită de cea specificată în program. Figura 4.11 prezintă structura acestei căi de date, care este împărțită în patru unități de prelucrare: unitatea de extragere și decodificare, unitatea de expediere și execuție, unitatea de retragere și rezervorul de instrucțiuni. Instrucțiunile și datele sunt transmise acestor unități prin unitatea de interfață cu magistrala.

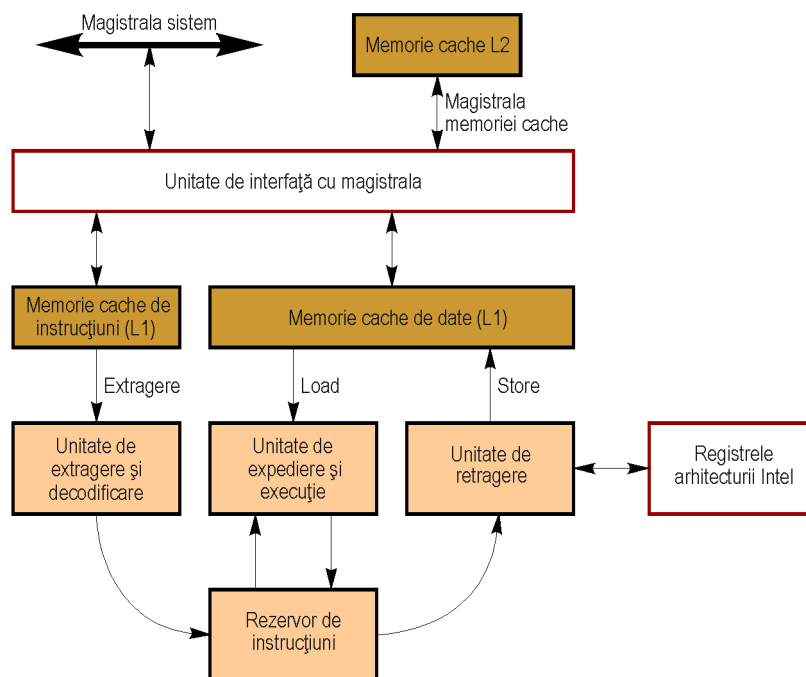


Figura 4.11. Unitățile de prelucrare din microarhitectura procesoarelor din familia P6 și interfața acestora cu subsistemul de memorie.

Pentru a asigura furnizarea constantă a instrucțiunilor și datelor la unitățile de execuție din calea de date, microarhitectura utilizează două nivele de memorie *cache*. Memoria *cache* L1 este împărțită într-o memorie *cache* de 8 KB pentru instrucțiuni și o memorie *cache* de 8 KB pentru date, ambele integrate în calea de date. Memoria *cache* L2 este o memorie RAM statică de 256 KB, 512 KB sau 1 MB, care este conectată la calea de date printr-o magistrală de 64 de biți, funcționând la frecvența de ceas a procesorului.

Microarhitectura procesoarelor din familia P6 utilizează *execuția dinamică* a instrucțiunilor. Mecanismul de execuție dinamică cuprinde trei concepte:

- Predicția salturilor;
- Analiza dinamică a fluxului de date;
- Execuția speculativă.

Predicția salturilor este un concept întâlnit la majoritatea arhitecturilor performante de calculatoare și a microprocesoarelor de viteză ridicată. Acest concept permite procesorului decodificarea instrucțiunilor de după cele de salt pentru a nu goli calea de date *pipeline* la execuția instrucțiunilor de salt. Unitatea de extragere și decodificare a instrucțiunilor utilizează un algoritm de predicție optimizat pentru anticiparea direcției fluxului de instrucțiuni prin nivele multiple de salturi, apeluri de proceduri și reveniri din proceduri.

Analiza dinamică a fluxului de date implică analiza în timp real a fluxului datelor prin procesor pentru a determina dependența datelor și a registrelor, și pentru a detecta posibilitatea execuției instrucțiunilor într-o ordine diferită de cea specificată în program. Unitatea de expediere și execuție a instrucțiunilor poate monitoriza simultan mai multe instrucțiuni și poate executa aceste instrucțiuni într-o ordine în care se optimizează utilizarea unităților multiple de execuție ale microarhitecturii, menținând în același timp integritatea datelor asupra cărora se operează. Această ordine de execuție asigură ocuparea unităților de execuție chiar și atunci când apar dependențe între datele instrucțiunilor.

Execuția speculativă se referă la posibilitatea procesorului de a executa instrucțiuni aflate înaintea instrucțiunii adresate de contorul de program, și de a furniza rezultatele în ordinea șirului inițial de instrucțiuni. Pentru ca execuția speculativă să fie posibilă, microarhitectura familiei P6 separă expedierea și execuția instrucțiunilor de producerea rezultatelor. Unitatea de expediere și execuție a instrucțiunilor utilizează analiza fluxului de date pentru a executa toate instrucțiunile din rezervorul de instrucțiuni și a memora rezultatele în registre temporare. Unitatea de retragere a instrucțiunilor caută apoi instrucțiunile care au fost executate și pentru care nu mai există dependențe de date cu alte instrucțiuni sau predicții nerezolvate ale salturilor. Atunci când se găsesc instrucțiuni care au fost executate, unitatea de retragere a instrucțiunilor depune rezultatele acestor instrucțiuni în memorie sau în registrele microarhitecturii (cele opt registre ale procesorului și opt registre ale unității de calcul în virgulă mobilă), în ordinea specificată în program, și retrage aceste instrucțiuni din rezervorul de instrucțiuni.

4.5.2.2. Prezentare detaliată a microarhitecturii

Figura 4.12 prezintă o schemă bloc funcțională a microarhitecturii procesoarelor din familia P6.

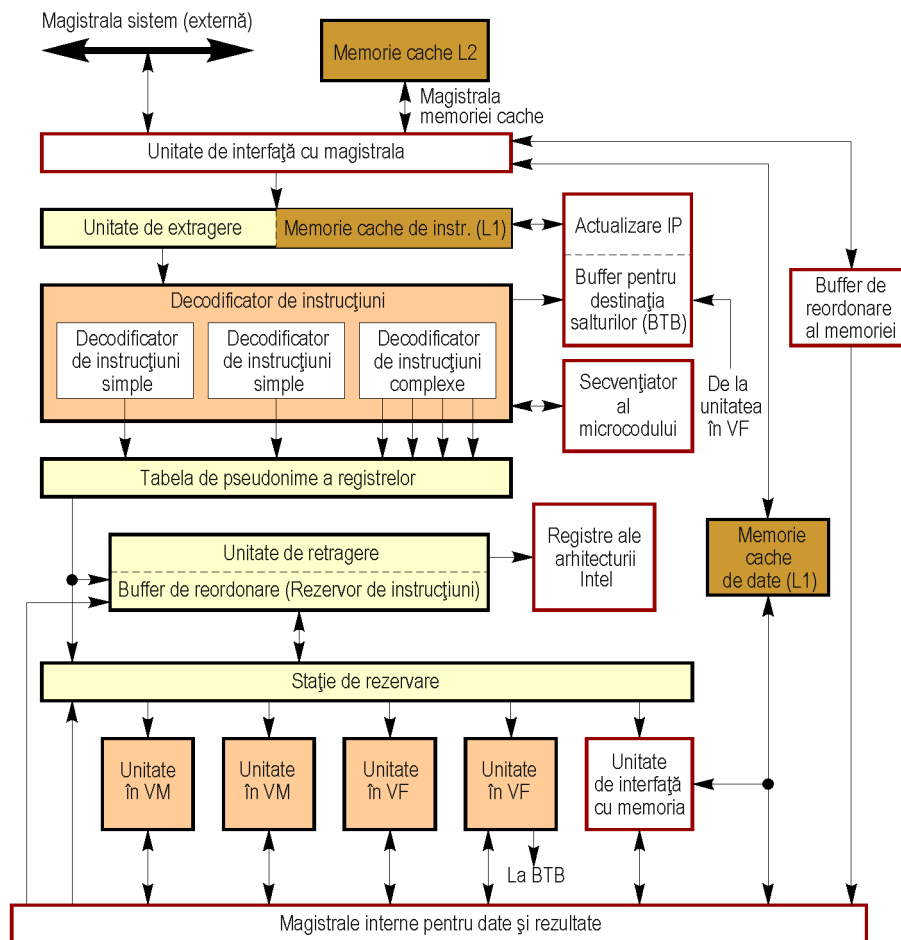


Figura 4.12. Schema bloc funcțională a microarhitecturii procesoarelor din familia P6.

În această schemă, există următoarele blocuri componente ale celor patru unități de execuție și a subsistemului de memorie din Figura 4.11:

- *Subsistemul de memorie:* magistrala sistem, memoria *cache* L2, unitatea de interfață cu magistrala, memoria *cache* L1 de instrucțiuni, memoria *cache* L1 de date, unitatea de interfață cu memoria și bufferul de reordonare al memoriei.
- *Unitatea de extragere și decodificare:* unitatea de extragere a instrucțiunilor, bufferul pentru destinația salturilor (BTB – *Branch Target Buffer*), decodificatorul de instrucțiuni, secvențiatorul microcodului și tabela de pseudonime a registrelor (*Register Alias Table*).

- Rezervorul de instrucțiuni: bufferul de reordonare.
- *Unitatea de expediere și execuție*: stația de rezervare, două unități de calcul pentru numere întregi, două unități de calcul în virgulă mobilă și două unități de generare a adreselor.
- *Unitatea de retragere a instrucțiunilor*: unitatea de retragere și setul de registre al arhitecturii Intel.

Subsistemul de memorie

Subsistemul de memorie al microarhitecturii P6 este format din memoria principală a sistemului, memoria *cache* primară (L1) și memoria *cache* secundară (L2). Unitatea de interfață cu magistrala accesează memoria sistemului prin magistrala sistemului (externă). Această magistrală de 64 de biți este o magistrală orientată pe tranzacții, ceea ce înseamnă că fiecare acces la magistrală este tratat prin operații separate de cerere și răspuns. În timp ce unitatea de interfață cu magistrala așteaptă răspunsul la o cerere de acces la magistrală, se pot emite alte cereri.

Unitatea de interfață cu magistrala accesează memoria *cache* L2 printr-o magistrală de 64 de biți a memoriei *cache*. Această magistrală este de asemenea orientată pe tranzacții, permițând până la patru accese concurente la memoria *cache*, și funcționează la aceeași frecvență de ceas ca și procesorul.

Accesul la memoria *cache* L1 se realizează prin magistrale interne, de asemenea la frecvența de ceas a procesorului. Memoria *cache* L1 de 8 KB pentru instrucțiuni este un set asociativ cu patru căi. Memoria *cache* L1 de 8 KB pentru date este un set asociativ cu două căi, având două porturi, și permite o operație de încărcare și una de memorare într-un ciclu.

Coerența între memoriile *cache* și memoria sistemului este asigurată prin utilizarea protocolului MESI (*Modified, Exclusive, Shared, Invalid*). Acest protocol este utilizat pentru asigurarea coerenței între memoriile *cache* atât la sistemele cu un singur procesor, cât și la cele multiprocesor.

Cererile de acces la memorie de la unitățile de execuție ale procesorului trec prin unitatea de interfață cu memoria și prin bufferul de reordonare al memoriei. În cazul unei lipse în memoria *cache* L1 pentru date, această memorie redirectează cererea către memoria *cache* L2, și apoi, în cazul unei lipse în această memorie, unitatea de interfață cu magistrala redirectează cererea către memoria sistemului.

Cererile de acces la memoria *cache* L2 sau la memoria sistemului trec prin bufferul de reordonare al memoriei, care funcționează ca o stație de planificare și expediție. Această unitate păstrează evidența tuturor cererilor de acces la memorie și poate reordona anumite cereri pentru a preveni blocajele și a îmbunătăți rata de transfer. De exemplu, bufferul de reordonare al memoriei permite operații speculative de încărcare. (Operațiile de memorare sunt executate întotdeauna în ordinea inițială, iar operațiile speculative de memorare nu sunt permise.)

Unitatea de extragere și decodificare

Unitatea de extragere și decodificare citește un șir de instrucțiuni din memoria *cache* L1 de instrucțiuni și le decodifică într-o serie de micro-operații. Acest șir de micro-operații este transmis apoi la rezervorul de instrucțiuni.

Unitatea de extragere a instrucțiunilor încarcă în fiecare ciclu de ceas o linie de 32 de octeți din memoria *cache* de instrucțiuni. Unitatea marchează începutul și sfârșitul instrucțiunilor în cadrul liniilor memoriei *cache* și transmite 16 octeți aliniați la decodificatorul de instrucțiuni. Unitatea de extragere a instrucțiunilor calculează valoarea pointerului de instrucțiuni, pe baza intrărilor de la bufferul pentru destinația salturilor, a stării excepțiilor și a întreruperilor. Pentru a executa predicția salturilor, bufferul pentru destinația salturilor, cu 512 intrări, urmărește un număr mare de instrucțiuni aflate înaintea contorului de program. În cadrul acestor instrucțiuni pot fi numeroase salturi, apeluri de proceduri și reveniri din proceduri, care trebuie anticipate în mod corect.

Decodificatorul de instrucțiuni conține trei decodificatoare: două decodificatoare pentru instrucțiuni simple și un decodificator pentru instrucțiuni complexe. Fiecare decodificator convertește o instrucțiune într-una sau mai multe micro-operații, fiecare având două surse logice și o destinație logică. Micro-operațiile sunt instrucțiuni primitive care sunt executate în unitățile paralele de execuție ale procesorului.

Multe instrucțiuni ale procesorului sunt convertite direct în câte o singură micro-operație de către decodificatoarele pentru instrucțiuni simple, iar anumite instrucțiuni sunt decodificate într-un număr de până la patru micro-operații. Instrucțiunile mai complexe sunt decodificate în secvențe programate de micro-operații obținute de la secvențiatorul microcodului. Decodificatoarele de instrucțiuni realizează de asemenea decodificarea prefixelor de instrucțiuni și a operațiilor de ciclare. Aceste decodificatoare pot genera până la șase micro-operații pe ciclu de ceas (câte una de către decodificatoarele pentru instrucțiuni simple și patru de către decodificatoarele pentru instrucțiuni complexe).

Setul de registre al procesorului poate cauza blocaje din cauza dependențelor între registre. Pentru a rezolva această problemă, procesorul dispune de 40 de registre interne, care se utilizează pentru calcule. Aceste registre pot păstra atât valori întregi, cât și valori în virgulă mobilă. Pentru alocarea registrelor interne, micro-operațiile de la decodificatorul de instrucțiuni sunt transmise la tabela de pseudonime ale registrelor, unde referințele logice la registrele procesorului sunt convertite în referințe fizice la registrele interne.

În etapa finală a procesului de decodificare, alocatorul din cadrul tabelii de pseudonime ale registrelor adaugă biți de stare la micro-operații pentru a le pregăti în vederea execuției într-o ordine diferită de cea secvențială, și transmite micro-operațiile rezultate la rezervorul de instrucțiuni.

Rezervorul de instrucțiuni

Înainte de depunerii micro-operațiilor în rezervorul de instrucțiuni (cunoscut și sub numele de buffer de reordonare), șirul de instrucțiuni decodificat în micro-operații

se află în aceeași ordine cu cea în care acest șir a fost transmis la decodificatorul de instrucțiuni.

Bufferul de reordonare este o memorie asociativă organizată în 40 de registre. Bufferul conține micro-operații care așteaptă pentru a fi executate, ca și micro-operații care au fost deja executate, dar ale căror rezultate nu au fost încă memorate. Unitatea de expediere și execuție poate executa micro-operațiile din bufferul de reordonare în orice ordine.

Unitatea de expediere și execuție

Unitatea de expediere și execuție planifică și execută micro-operațiile depuse în bufferul de reordonare ținând cont de dependențele de date și de disponibilitatea resurselor, și memorează temporar rezultatele acestor execuții speculative.

Planificarea și expedierea micro-operațiilor din bufferul de reordonare este realizată de stația de rezervare. Aceasta urmărește în continuu bufferul de reordonare pentru a determina micro-operațiile care sunt gata de execuție (cele pentru care toți operanzii sursă sunt disponibili) și le expediază pe acestea la unitățile de execuție disponibile. Rezultatele execuției unei micro-operații sunt returnate în bufferul de reordonare și sunt memorate împreună cu micro-operația până când aceasta este retrasă din buffer. Acest proces de planificare și expediere permite execuția într-o ordine oarecare, în care micro-operațiile sunt expediate la unitățile de execuție strict pe baza constrângerilor fluxului de date și a disponibilității resurselor utilizate pentru execuție, fără a ține cont de ordinea inițială a instrucțiunilor.

Execuția micro-operațiilor este realizată de două unități în virgulă fixă (VF), două unități pentru numere în virgulă mobilă (VM) și o unitate de interfață cu memoria, permițând planificarea a până la cinci micro-operații într-un ciclu de ceas.

Cele două unități în virgulă fixă pot executa în paralel două micro-operații pentru numere întregi. Una din unitățile în virgulă fixă poate gestiona micro-operațiile de salt. Această unitate poate detecta predicțiile eronate ale salturilor și poate semnaliza bufferului pentru destinația salturilor să inițializeze calea de date *pipeline*. Această operație este realizată după cum urmează. Decodificatorul de instrucțiuni marchează fiecare micro-operație de salt cu ambele adrese posibile de destinație ale saltului (adresa de salt și adresa instrucțiunii următoare). Atunci când unitatea în virgulă fixă execută micro-operația de salt, aceasta este în măsură să determine destinația care a fost selectată. Dacă destinația anticipată a fost selectată, atunci micro-operațiile executate speculativ sunt marcate ca fiind utilizabile și execuția continuă pe calea anticipată. Dacă destinația anticipată nu a fost selectată, o unitate de execuție din cadrul unității în virgulă fixă modifică starea tuturor micro-operațiilor de pe calea care nu a fost selectată pentru a le elimina din rezervorul de instrucțiuni. Unitatea respectivă transmite apoi destinația corectă a saltului la bufferul pentru destinația salturilor, care inițializează calea de date *pipeline* de la noua adresă.

Unitatea de interfață cu memoria gestionează micro-operațiile de încărcare și de memorare. Un acces de încărcare necesită specificarea doar a adresei de memorie, astfel încât poate fi codificat într-o micro-operație. Un acces de memorare necesită specificarea atât a adresei, cât și a datei, astfel încât este codificat în două micro-

operații. Partea unității de interfață cu memoria care gestionează micro-operațiile de memorare dispune de două porturi, ceea ce îi permite prelucrarea în paralel a celor două micro-operații care codifică accesul de memorare. Unitatea de interfață cu memoria poate executa deci în paralel atât o încărcare, cât și o memorare într-un ciclu de ceas.

Unitatea de retragere a instrucțiunilor

Unitatea de retragere a instrucțiunilor memorează în mod permanent rezultatele micro-operațiilor executate speculativ și elimină micro-operațiile din bufferul de reordonare. Ca și stația de rezervare, unitatea de retragere a instrucțiunilor testează în mod continuu starea micro-operațiilor din bufferul de reordonare, căutând acele micro-operații care au fost executate și nu au dependențe cu alte micro-operații din rezervorul de instrucțiuni. Această unitate retrage apoi micro-operațiile executate în ordinea inițială specificată în program, ținând cont de întreruperi, excepții și anticipări eronate ale salturilor.

Unitatea de retragere a instrucțiunilor poate retrage trei micro-operații pe ciclu de ceas. Pentru retragerea unei micro-operații, unitatea scrie rezultatele în registrele procesorului sau în memorie. După memorarea rezultatelor, micro-operația este eliminată din bufferul de reordonare.