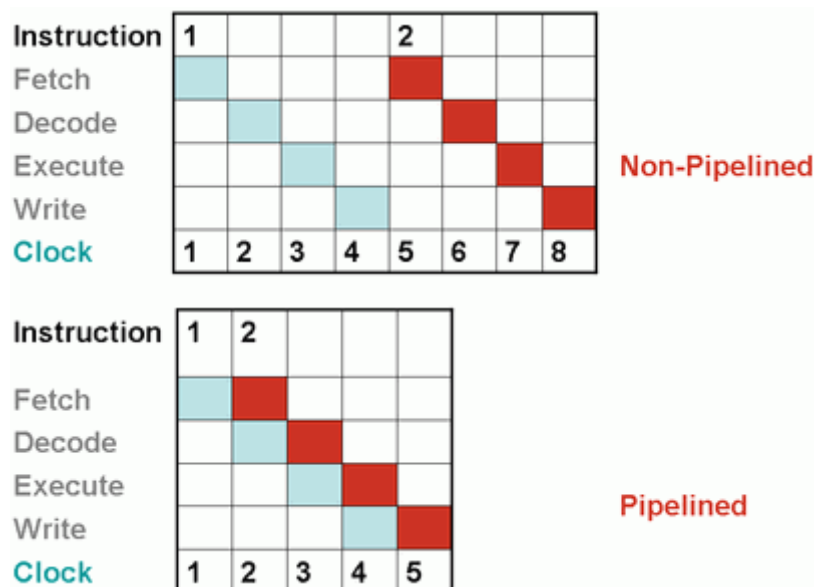


15. Vývoj procesorů od Pentia do současnosti

Processor skalární architektury

Procesor skalární architektury je složen z více funkčních bloků, které pracují samostatně. Je umožněn tzv. pipelining = zřetěžené zpracování či překrývání strojových instrukcí, což je způsob zvýšení výkonu procesoru současným prováděním různých částí několika strojových instrukcí. Základní myšlenkou je rozdělení zpracování jedné instrukce mezi různé části procesoru a tím i dosažení možnosti zpracovávat více instrukcí najednou. To je řešeno, tak že když jedna instrukce opustí třeba EU, tak se nahraje do EU další, které je za ní.



1. Výpočet adresy místa v operační paměti
2. Načtení instrukce z dané adresy v operační paměti
3. Dekódování instrukce(nalezení mikrokódu naznačující instrukci v ROM)
4. Provedení výpočtů v ALU / FPU
5. Zápis výsledku do registrů.

Čím více má pipeline stupňů, tím je efektivnější zpracování instrukcí v případě, že dojde v programu ke skoku, musí se celá pipeline (fronta instrukcí) vyprázdnit a načíst s dlouhým zdržením instrukce nové. Tyto procesory obsahovaly cache paměť, ale pouze L1.

Superskalární

Umožňují zpracování několika instrukcí v jednom instrukčním řetězci (pipeline) zároveň.

Paralelní ALU

Obsahují více prováděcích jednotek (především ALU) v jednom jádru procesoru, které jsou schopné paralelního zřetězení. V jednom taktu může být dokončeno více instrukcí, tato funkce se nazývá superpipelining. ALU jsou na sobě nezávislé, jedna nečeká na výsledek ostatních.

První superskalární architektura byla použita u Pentia, který měl 2 pipeline (U-pipe a V-pipe). U superskalární architektury rozhoduje o paralelním provádění operací řadič procesoru.

Oproti skalární architektuře je tedy konstrukce řadiče procesoru složitější, neboť kromě své běžné činnosti (tj. řízení pipeline a jednotlivých funkčních jednotek procesoru) musí ještě rozhodovat o tom, které instrukce je možné párovat a zpracovávat současně.

Aby se zabránilo vyprazdňování pipeline z důvodu skoku v programu, začala se používat technologie predikce (předvídání, odhadování) skoků v programu. Do novějších procesorů této architektury byla přidána cache L2 obsahovala několik nových funkcí:

Predikce skoků

Tato technologie umožňuje dopředu odhadnout, zda se skok v programu provede či nikoli a na základě toho se začne do instrukční pipeline vkládat (a postupně zpracovávat) buď instrukce, které se nachází ihned za skokem či naopak v cíli skoku.

Algoritmus je založen na následujícím faktu: vždy po provedení těla cyklu se podmíněnou skokovou instrukcí rozhoduje o jeho opětovém provedení. S významně větší pravděpodobností nastává situace, že při zopakování téže podmíněné skokové instrukce bude její výsledek stejný.

Příklad činnosti prediktoru skoku: Jednoduchý 1 bitový prediktor skoků pracuje tak, že si procesor u každého skoku uloží do 1 bitového registru informaci, zda byl skok proveden či nikoli. Když se v instrukční pipeline znovu skok objeví (souhlasí jeho adresa), tak se procesor na základě předešlého běhu programu rozhodne, které další instrukce se pravděpodobně budou zpracovávat. Při prvním volání skoku má prediktor pouze padesátiprocentní úspěšnost, ta se potom zvyšuje.

Kolize: pokud dojde k tomu, že bude ve dvou pipelinech instrukce, které vyžadují stejnou adresu pro svoji operaci ve stejný okamžik, tak dojde k dočasnému přerušování provádění.

Také pokud se použije operace jump na instrukci, které je v tu dobu v druhé pipeline, tak obě budou vymazány a instrukce se nahrají znova. K tomu je důležitá predikce skoků.

Technologie MMX (MultiMedia eXtensions)

Jelikož vylepšení této architektury přineslo nové instrukce, které pracovali pouze s celými čísly, tak bylo pro zrychlení procesoru přidáno 8 nových 64 bitových registrů, jsou označovány mm0-mm7.

Pro tyto registry byly vytvořeny nové instrukce, které využívali techniku SIMD (Single Instruction Multiple Data), která dovoluje zpracovat více operandů (dat, čísel) během jedné instrukce.

Tyto registry reprezentují

- 8 bajtů (označení B)
- 4 slova (words) (označení W) - 1 word je unsigned int (16bit, 0 a více hodnoty) - ushort
- 2 dvoj slova (dwords) (označení W) - 1 dword = double word je unsigned int (32bit, 0 a více hodnoty) - uint
- 1 čtyř slova (qword) (označení Q) - 1 qword je unsigned int (64bit, 0 a více hodnoty) - ulong

Instrukční sada se skládá z:

- Package všech MMX instrukcí
- Přidané instrukce
- Signed nebo unsigned (signed +, unsigned +)
- Obsah registru (bajt nebo slova)
- Cíl - číslo označení MMX registru
- Odkud - 64bit adresa v paměti nebo číslo označení MMX registru

Instrukce MMX

MOV - přesun dat mezi registry nebo paměti

Písmeno za MOV: B, W, D, Q

Příklad zapsání MOVW odkud, kam

Aritmetické operace

Logické

Min, Max, Average

Dynamické spouštění instrukcí (Byl až od Pentia PRO a poté až v hyperskalárních)

Jedná se o vylepšení bez pořádného mechanismu pro vykonávání instrukcí. Je to mechanismus, při němž se instrukce vykonávají v jiném pořadí, než jak uvádí program uložený v operační paměti.

Procesor (resp. řadič procesoru) provede tzv. analýzu toku dat (analýzu dekodovaných instrukcí) a zjišťuje, zda jsou závislé na dokončení jiných instrukcí. Následně seřadí instrukce do optimálního pořadí a provádí je tak, aby byly zpracovány v co nejkratším čase (na sobě nezávislé instrukce souběžně, tedy paralelně). Příklad využití: Pokud nemůže být některá instrukce provedena, protože nemá k dispozici data, provede se jiná instrukce. Tímto způsobem lze odstranit možná zpoždění, která vzniknou v důsledku nedostupnosti dat.

Je založen na 3 principech

- Branch prediction
 - umožňuje dekódovat instrukce mezi větvením - plné využití sběrnice
 - Pentium Pro procesor využívá vysoce optimalizovaný algoritmus pro predikci větvení, pro směřování toku instrukcí přes několik úrovní
- Dynamic data flow analysis
 - analýza toku dat procesorem v reálném čase pro určení závislostí mezi daty a registry
 - detekce příležitostí pro zpracování instrukcí (zpracování instrukcí mimo pořadí)
- Speculative execution
 - dává zprávu procesoru o přednostně vykonávaných instrukcích, ale jejich výsledky předává do původního programu, zajišťuje maximální využití procesoru

Mikrokód

Mikrokód je jako Firmware pro procesor. Mikrokód přenáší instrukce, které CPU přijme, do fyzických operací na úrovni obvodu, které se dějí uvnitř procesoru. U starších byl mikrokód dodáván BIOSem při startu počítače, u novějších si ho procesor při startu přečte z flash paměti. Mikrokód může být upravován, proto je pamět typu flash. (Aktualizace mikrokódů mohou také opravit chyby a další chyby bez nutnosti úplné výměny hardwaru CPU).

Hyperskalární

Je rozdělen na různé architektury. Umožňuje rozpracování až 20 instrukcí v jedné pipeline. Zdokonalená predikce skoku v programu: zpracovávají se obě varianty kódu pro případ, kdy je podmínka skoku splněna i nesplněna. Teprve až je výsledek podmíněného skoku znám, nevhodnou větev vyprázdní. Procesorová sběrnice FSB s technologií QPB- až 4 přenosy na sběrnici během 1 hodinového taktu.

RISC (Reduced Instruction Set Computer)

Označuje procesory s redukovanou instrukční sadou, jejichž návrh je zaměřen na jednoduchou, vysoce optimalizovanou sadu instrukcí oproti protikladu s množstvím specializovaných instrukcí ostatních architektur. Skládá se z: Jádro - hlavní funkční jednotka, je určena pro aritmeticko-logické operace a posuvy na 32b datech, 32b adresové výpočty pro přístup k instrukcím a datům a pro řízení činnosti procesoru.

VCP - vektorový koprocesor vykonávající aritmetické a logické operace na 64b pakovaných slovech

LMI a GMI - 2 identická programovatelná rozhraní pro externí paměť.

CP0 a CP1 - 2 identické komunikační porty, každý zajišťuje přenos dat pomocí obousměrné 8b sběrnice mezi procesorem a externím zařízením.

Mikroinstrukce jsou hardwarově implementovány na procesoru, čímž je velmi výrazně zvýšena rychlost jejich provádění

Cache

Skalární pouze L1

Superskalární L1 a L2

Hyperskalární L1, L2, s technologií Smart od core, od pozdějších přibyla L3
Je vysvětlena v otázce vnitřních pamětí.

Hyperthreading

Technologie vytváří z jednoho fyzického procesoru dva virtuální procesory tím, že jsou v něm aktivovány dvě řídící jednotky. Vůči softwaru systém představuje místo jednoho fyzického dva (logické) procesory.

Toho se dosáhlo zdvojením jednotek, které uchovávají aktuální stav výpočetních procesů – registrů. Všechny ostatní prostředky procesoru, jako jsou výpočetní jednotky ALU / FPU, prediktory a paměti CACHE, jsou sdílené. Hyper-threading umožňuje lépe využít hardware procesoru, snížit odezvu systému (latenci), zrychlit výpočty.

Jelikož se softwaru prezentuje, že má více jader, tak software zasílá do procesoru více instrukcí než do procesoru, který nemá aktivní hyper-threading.

Hyper-Threading umožňuje zvýšit výkon aplikačnímu softwaru, který umí pracovat s více vlákny současně (multithreading) nebo je schopný práce v režimu zpracování více úloh naráz (multitasking).

Další technologie

XD (Execute Disable) bit technologie

snaží se zabránit útokům červů nebo virů, kteří využívají přetečení paměťového zásobníku.

Intel 64 - rozšíření instrukční sady pro práci s 64bitovými čísly

SpeedStep

technologie, která dokáže upravit taktovací frekvenci jádra procesoru dle potřeby aplikace. Dokáže tedy šetřit elektrickou energii (snížovat spotřebu) procesoru a zároveň snižovat jeho zahřívání.

Tepelná ochrana

za použití přesného, továrně nastaveného teplotního čidla přímo na čipu je velmi rychle reagující obvod tepelné ochrany (TCC - Thermal Control Circuit) schopen udržet teplotu procesorového čipu v rozsahu továrních specifikací za prakticky všech podmínek (např. selhání chlazení). Tepelný monitor řídí teplotu procesoru pomocí změny taktovací frekvence jádra procesoru. Hodinový signál procesoru je upravován tehdy, je-li aktivován obvod TCC. V případě nárůstu teploty nad kritickou mez, obvody tepelné ochrany odstaví, CPU od napájení. Tato technologie umožňuje omezování výkonů procesorů v noteboocích, které nemají dobré chlazení.

Core a Core 2

Core, Core2, 1. Nehalem, 2. Sandy Bridge,....

Obsahuje více jader, nejprve 2 poté na 4 a dnes je na 8 a 10 a více.

Neměl Hyper-Threading

Pokud je vícejádrový procesor podporován operačním systémem a konkrétní aplikací, lze instrukce zpracovávat paralelně.

To znamená, že procesor může řešit více úloh současně, popřípadě jednu úlohu rozložit na více výpočetních jader.

Podporuje 64bitové aplikace (obsahuje plně 64 bitové výpočetní jednotky ALU)

Technologie

Wide Dynamic Execution

technika, která umožňuje zpracovat během jednoho hodinového taktu více instrukcí. Architektura Core má k dispozici čtyři instrukční dekodéry. Tři jsou určeny pro jednoduché instrukce (např. sčítání, násobení, odčítání), jeden dekodér je určen pro složitější instrukce, které jsou reprezentovány až čtyřmi mikroinstrukcemi, přičemž zvládá i ty jednoduché:

Intelligent Power Capability

technologie, která odpojuje od napájení nevyužitou část procesoru + řídí otáčky ventilátorů.

Advanced Smart Cache

jedná se o L2 cache paměť, která je sdílená mezi oběma jádry procesoru. Kapacitu je možné dynamicky přidělovat (alokovat) pro každé jádro (např. je-li jedno jádro nečinné, může druhé jádro využívat celou kapacitu L2 cache). Zároveň umožňuje výměnu dat přímo mezi L1 cache paměťmi obou jader procesoru (každé jádro 32 kB pro data a 32 kB pro instrukce), čímž se zrychluje výměna dat mezi jádry procesoru a nezatěžuje systémová sběrnice FSB.

Core Nehalem

Sběrnice QPI nahradila paralelní FSB. Přibyla Cache L3.

Měl Hyper-threading. Představil technologii Turbo Boost:

technologie, hlídající TDP procesoru a zároveň vytížení jader procesoru.

Pokud program nevyužívá všechna jádra procesoru, přepínají se nepoužívaná jádra do úsporného nebo spánkového režimu nebo jsou kompletně odpojena od napájení.

Novější

Zvyšování frekvencí jader a jejich počet. Zvyšování paměti cache.

Upgrade TurboBoost, zvýšení výkonu nad dané TDP, možnosti přetaktování se stále navyšují.

