Charakteristika dalších verzí procesorů Pentium

Cíl přednášky

- Poukázat na principy architektur nových verzí typů Pentií.
- Prezentovat aktuální pojmy.

Úvod

- Paralelní systémy lze třídit z hlediska počtu toků instrukcí a počtu toků dat:

  SI – systém s jedním tokem instrukcí (Single Instruction stream)

  MI – systém s několika toky instrukcí (Multiple Instruction stream)

  SD – systém s jedním tokem dat (Single Data stream)

  MD – systém s několikanásobným tokem dat (Multiple Data stream)
- Použití těchto dvou hledisek vede ke vzniku **čtyř základních typů počítačů označovaných zkratkami SISD, SIMD, MISD a MIMD**.

Kombinace základních architektur

- **SISD** (Single Instruction stream, Single Data stream) – klasický jednoprocesorový počítač von Neumannova typu zpracovávající data sériově.
- **SIMD** (Single Instruction stream, Multiple Data stream) – pole procesorů zpracovávající paralelně pole hodnot podle společného programu.
- **MIMD** (Multiple Instruction stream, Multiple Data stream) – multiprocesorový systém, v němž každý procesor je řízen samostatným programem a pracuje s jinými daty než ostatní procesory.
- **MISD** – soustava procesorů pracujících podle různých programů na společných datech (označováno jako enfant terrible této klasifikace) – v praxi neobsazená alternativa.

Uplatnění SIMD

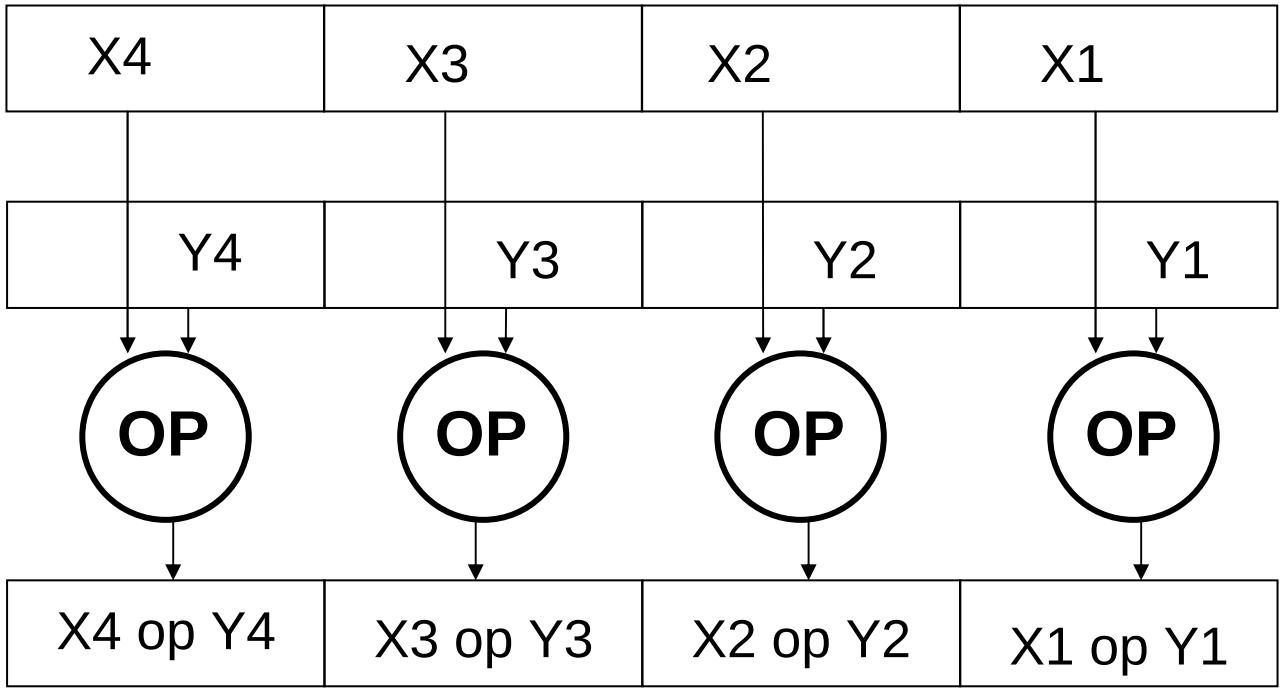
- Typický rys multimediálních aplikací – paralelně prováděné operace na stejných datech.
- Strategie SIMD – jeden tok instrukcí na jednom procesoru, procesorů je více, instrukce operuje na násobných ale shodných datech (typech).
- Příklad: technologie MMX, která operuje paralelně s násobnými daty ve zkomprimované formě uloženými v 8/16 bitovém registru.
- Jiný případ: stejná hodnota se přičítá k různým datům, častá situace v multimediálních aplikacích.
- Změna jasu celého obrazu: každý pixel sestává z informace o třech složkách jasu R, G, B. Pokud potřebujeme změnit jas celého obrazu, přečteme R, G, B z paměti, přičteme (odečteme) patřičnou hodnotu a výsledek zapíšeme zpět do paměti.

Vývoj technologie SIMD na úrovni instrukcí

- SIMD – Single Instruction, Multiple Data, jeden typ instrukce se provádí na více datech.

  Typická operace SIMD – obr. 1.

  Dvě skupiny operandů, každá obsahuje 4 zkomprimované datové prvky (X1, X2, X3, X4 a Y1, Y2, Y3, Y4).



Obr. 1 – operace v architektuře SIMD

Pentium II a Pentium III

- Takto to bylo u technologie MMX.
- Technologie MMX umožňovala realizovat operace SIMD na zkomprimovaných slabikách, slovech, dvouslovech (vše celá čísla) uložených v osmi 64 bitových registrech.
- Registry MMX – 64 bitů a XMM – 128 bitů.
- Specielní instrukce byly šity na míru pro multimediální a komunikační aplikace.

- **Pentium III** – model SIMD byl změněn na Streaming SIMD Extensions (SSE).
- Rozdíl oproti klasickému SIMD:
  - Operace se provádějí na operandech obsahujících čtyři komprimovaná čísla v pohyblivé řádové čárce.
  - Operandy jsou uloženy buď v paměti nebo v osmi 128 bitových registrech (registry XMM).
  - Specielní množina instrukcí vytvořená pro procesory MMX byla rozšířena o dalších 64 instrukcí.

Pentium 4

Pentium 4 – byl použit model **Streaming SIMD Extensions 2 (SSE2)** s těmito vlastnostmi:

- Operace se provádějí na těchto operandech:

  dvě čísla v pohyblivé řádové čárce, dvojnásobná přesnost,

16 zkomprimovaných slabik,

8 zkomprimovaných slov,

4 zkomprimovaná dvouslova,

2 zkomprimovaná čtyřslova

(čtyři poslední alternativy – čísla typu integer).

- Operandy mohou být v paměti nebo registrech.
- Podpora aritmetiky SIMD pro práci se 64 bitovými operandy typu integer.
- Instrukce pro konverzi mezi datovými typy (původními a novými).

• **Trend: SIMD na úrovni instrukcí, rozšiřování množiny datových typů (změna v množině instrukcí).**

Hyper-threading v Pentiu 4

- Hyper-threading – vláknové technologie.
- Předcházející obrázek: Pentium 4 CPUs umožňuje akcelerovat různé typy operací.
- Sestává pouze z jednoho fyzického procesoru – ten se ale jeví tak, že sestává ze dvou nebo více logických (virtuálních) procesorů (tzv. vlákna).
- Typy procesorů spadajících do kategorie Pentium 4:

  CELERON 4

  XEON – serverové aplikace
- Rok 2005 – Pentium D a Pentium Extreme Edition dual-core CPUs.
- Pojem Hyper-threading se týká architektur s jedním fyzickým procesorem.
- Používá se zkratka **HTT - Hyper-Threading Technology**.

Další pojmy

- **Simultaneous multithreading (SMT)** – technika pro zvýšení efektivity superskalárních architektur.
- Existence několika nezávislých vláken – lepší využití hardware, který je v každé frontě k dispozici.
- Pozor – zkratka SMT má i jiný význam: Sufface Mounted Technology.

Další pojmy – pojem core

- Pojem core je odlišný od pojmu logický procesor.
- Core má svou vlastní sadu EU (execution units).

  Důsledek: dvě vlákna, každé existující na samostatném core, „nesoutěží“ o zdroje (tzn. EU).
- Core může sestávat z více logických procesorů.
- Příklad multi-core HT Technology: jeden fyzický procesor sestávající ze dvou core (dual-core) (čtyř logických procesorů).
- Použití: multi-threading aplikace.

V čem spočívá implementace vlastností Netburst micro-architecture v Pentiu 4

- **Rychlé vyrovnávací paměti integrované do čipu (on-chip cache)**
- Na rozdíl od dřívějších typů to není pouze klasická rychlá vyrovnávací paměť L1, ale:
  - rychlá vyrovnávací paměť L1 pro data kapacity 8 kB,
  - rychlá vyrovnávací paměť L2, 8 way, kapacity 256 kB **(**v terminologii Intel **Advanced Transfer Cache**), instrukce a data
  - rychlá vyrovnávací paměť typu **Execution Trace Cache** kapacity 12K μop,

  Pojem **Advanced Transfer Cache** – vyrovnávací paměť L2 (integrována do čipu procesoru), pracuje na stejném kmitočtu jako procesor).

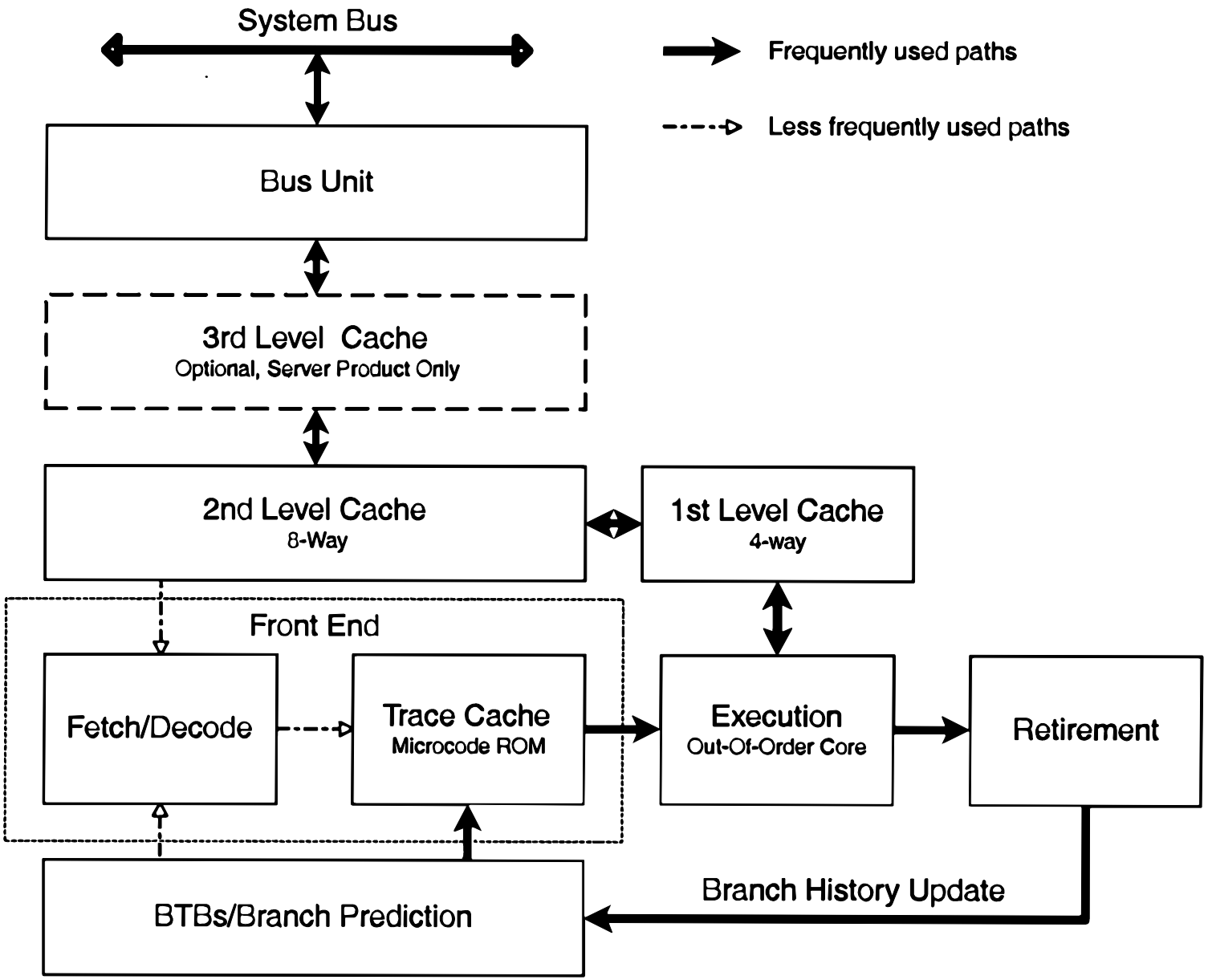
Rychlost komunikace procesoru s okolím

- Pentium III komunikovalo s okolím na frekvenci 100/133 MHz, Pentium 4 - komunikace s okolím synchronizována kmitočtem 266 MHz (přenosy realizovány 4x v rámci jednoho cyklu) – v začátcích.
- Dnes běžně 400/533/1066 MHz.
- **Trend: zvyšování rychlosti komunikace procesoru s okolím.**

- Pojem FSB (Front Side Bus) – rozhraní procesoru

Další vlastnosti Intel NetBurst Micro-Architecture

- Hyperskalární (zřetězení instrukcí – více jak 10 jednotek) a superskalární architektura (dvě fronty).
- Dokonalé (?) uplatnění principů zřetězení, přičemž různé komponenty jsou synchronizovány různými frekvencemi, některé vyššími, některé nižšími frekvencemi než je frekvence procesoru.
- Synchronizace procesoru - frekvence 3 - 4 Ghz.
- Zvyšování kmitočtu procesoru – výrazná snaha o zvýšení počtu jednotek (kvůli zvyšování kmitočtu je to nutnost).
- Provádění mikroinstrukcí mimo pořadí.
- Konstrukce obvodů tak, aby se častěji prováděné instrukce realizovaly rychleji.
- **Trend: snaha o uplatnění principů architektur RISC na úrovni provádění mikroinstrukcí v procesorech Intel.**


System Bus
Frequently used paths
Less frequently used paths
Bus Unit
3rd Level Cache
Optional, Server Product Only
2nd Level Cache
8-Way
1st Level Cache
4-way
Front End
Fetch/Decode
Trace Cache
Microcode ROM
Execution
Out-Of-Order Core
Retirement
BTBs/Branch Prediction
Branch History Update

Funkce Front End

- Sestává ze dvou bloků:
  - Fetch/Decode
  - Execution Trace Cache
- Front End má tuto funkci:
  - čtení instrukcí, jejich dekódování a náhrada posloupnostmi mikrooperací (Intel rozlišuje tyto typy instrukcí: instruction, complex instruction, special purpose instruction).
  - přenos dříve dekódovaných instrukcí z Execution Trace Cache,
  - predikci výsledků instrukcí skoku.
- **Trend: architektura moderních mikroprocesorů fy Intel usiluje o originální řešení problémů různých zpoždění, k nimž dochází při provádění instrukcí. Mezi tato zpoždění patří, např.:**
  - **doba potřebná na dekódování instrukcí**
  - **doba potřebná na řešení problémů větvení programu (skoky).**

Execution Trace Cache

- **Princip:**
  - Instrukce jsou rozdekódovány (Translation Engine) do posloupnosti mikrooperací (µop).
  - Instrukce jsou reflektovány posloupností mikrooperací – tyto posloupnosti se nazývají traces (kopie, obrazy) a jsou uloženy do Execution Trace Cache.
  - Posloupnosti mikrooperací jsou uloženy tak, jak odpovídá toku programu.
- Instrukce skoku - v Execution Trace Cache jsou k těmto mikrooperacím uloženy do stejných řádků výsledky těchto skoků => zvyšuje se pravděpodobnost správné predikce, zvyšuje se podíl kódu prováděného z Execution Trace Cache.

Execution Trace Cache

- Pentium 4 – Execution Trace Cache může uchovat až 12K μoperací.
- Procesor Pentium 4 - častěji prováděné instrukce jsou realizovány z Execution Trace Cache, menší množství instrukcí se provádí z paměti mikroprogramů (microcode ROM).
- Výsledek: jistá část instrukcí (počet závisí na velikosti Execution Trace Cache) má svou reprezentaci uloženou v Execution Trace Cache, pouze pro malou část je využívána paměť mikroprogramů.
- **Trend: zkrátit čas potřebný k dekódování instrukcí a zajištění přístupu k mikroprogramům.**

Jednotka Out-of-Order Core

- Jednotka Out-of-Order Core umí přeuspořádat provádění instrukcí tak, aby neutrpěla logika programu a byla přitom zohledněna konkrétní kriteria pro přeuspořádání.
- Princip: pokud nemůže být některá µop provedena, protože nemá k dispozici data, provede se jiná µop => tímto způsobem je možné odstranit možná zpoždění, která vzniknou v důsledku nedostupnosti dat.
- V mechanismech rozhodujících o zahájení realizace konkrétní µoperace se bere v úvahu také dostupnost potřebných hardwarových prostředků.
- Pořadí provádění mikrooperací se může modifikovat podle toho, zda jsou k dispozici hodnoty operandů pro patřičné mikrooperace a jsou volné hardwarové prostředky pro její realizaci.
- **Trend: optimalizovat principy realizace instrukcí zpracovávaných ve frontě s cílem zkrátit jejich provádění.**

Zpětné uspořádání výsledků posloupnosti instrukcí

- Tuto činnost provádí jednotka, kterou Intel označuje jako Retirement Section.
- Za ukončení mikrooperace se považuje stav poté, co je výsledek uložen do cílového registru.
- Jednotka Reorder Buffer (ROB) provádí zpětné uspořádání výsledků μinstrukcí tak, aby odpovídaly původní posloupnosti.
- Jednotka Retirement Section uchovává informaci o tom, jak dopadly skoky a předává ji do BTB (Branch Target Buffer).
- Podle této informace se obnovuje obsah BTB.

Techniky předvídání výsledků skoků

- Předvídání výsledků skoků je velmi důležité pro procesory zpracovávající fronty instrukcí.
- Techniky předvídání skoků (**Branch Prediction**) umožňují pokračovat ve zpracování programu na správném místě předtím, než je instrukce skoku zpracována a je znám výsledek skoku.
- Pojem **Branch Delay** reprezentuje zpoždění, které nastane, pokud dojde při předvídání výsledku skoku k omylu.
- Správná předpověď - Branch Delay je nulové.
- Mylná předpověď - Branch Delay bude odpovídat délce fronty (do fronty se musí načíst instrukce z jiné adresy než bylo předpovězeno a ta se do prováděcí jednotky dostane za takový počet cyklů, který odpovídá počtu komponent podílejících se na zpracování).
- **Trend: v oblasti předpovídání výsledků skoků se usiluje o inovaci těchto technik.**