

Számítógép architektúrák

Korszerű architektúrák

Mai program

- Pentium P6 processzor (esettanulmány)
- Párhuzamosítások a CPU-n kívül

Az Intel P6 család

- IA instrukciókat feldolgozó (x86 és Katmai Iset), háromutas szuperskalár, sok fokozatos (12-14 stage) futószalagelvű processzorok
- Spekulatív végrehajtás, elágazás előrejelzés
 - P1.
 - Pentium Pro
 - Pentium II., Pentium II Xeon
 - Pentium Celeron
 - Pentium III., Pentium III Xeon
- Nézzük ezt ...

Jellemzők

- Szuperskalár CISC processzor, RISC maggal
- Ciklusonként három RISC μ -operáció kibocsátására, öt RISC μ -operáció kiküldetésére képes
- 20 bejegyzéses várakoztató állomása van
- A szigorúan soros konzisztenciát átrendező puffer (ROB) biztosítja
- A regiszterátnevezést is a ROB biztosítja

CISC processzor — RISC maggal

- Az ilyen processzorokban a CISC instrukciókat RISC instrukciókká konvertálják
 - A RISC mikro-instrukciókat (μ ops) egy RISC mag hajtja végre
 - A μ ops-ok három operandusúak (triadikus jelleg): egy „kimenő” és két „bemenő” operandus a szokásos
 - Egy CISC instrukció egy vagy több μ ops-sá is konvertálódhat
- Pl. a köv. CISC instrukció kettővé:
- ```
SUB EAX, [EDI] // CISC
MOV r1,[EDI] // r1 ← [EDI]
SUB EAX,r1 // EAX ← EAX - r1
```
- Bonyolult CISC instrukciók sok  $\mu$ ops-sá

---

---

---

---

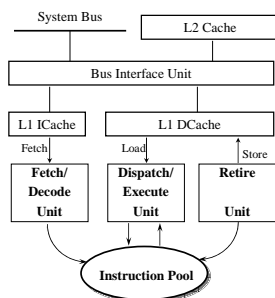
---

---

---

---

## 3 egység kommunikál közös instrukció mezőn keresztül



---

---

---

---

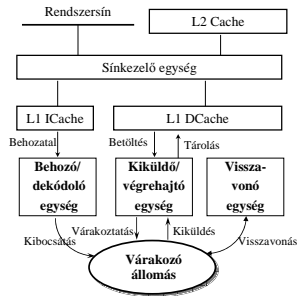
---

---

---

---

### 3 egység kommunikál közös instrukció mezőn keresztül




---

---

---

---

---

---

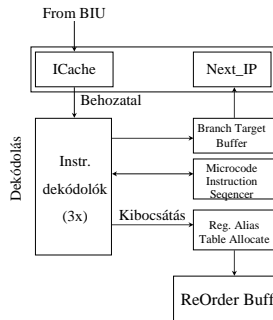
---

---

---

---

### A behozó és dekódoló egység



Next\_IP: L1 ICache index.

Az ID-be 16 byte-os egységekben IA instrukciókat hoznak be (behozatal, fetch).

A 3 dekóder (2 egyszerű, 1 komplex) az IA instrukciókat  $\mu$ ops-okká alakítja (1 v. 4  $\mu$ ops-sá). Amit nem tudnak: a MIS alakít át (sok  $\mu$ ops-sá).

A  $\mu$ ops-ok sorban átkerülnek a RAT-ba: a hamis függőségek feloldására (az IA regiszter hivatkozások fizikai regiszter hivatkozásokra képződnek). Ez az ún. instrukció kibocsátás.

---

---

---

---

---

---

---

---

---

---

### Kibocsátáshoz kötött operandus behívás

- A kibocsátás és regiszterátnevezés során megtörténik a (bemenő) operandusok „behívása”
  - Az átnevezett (bemenő operandust tartalmazó) regiszterek értéket kapnak (a megfelelő regiszterektől),
  - A ROB-beli  $\mu$ ops-okba beolvasódnak a regiszterértékek (regiszterfájl olvasás futószalag fázis)
- Létezik kiküldéshez kötött operandus-behívás politika is
- A kibocsátott  $\mu$ ops-ok a várakozó állomásban (Reservation Station) várakoznak

---

---

---

---

---

---

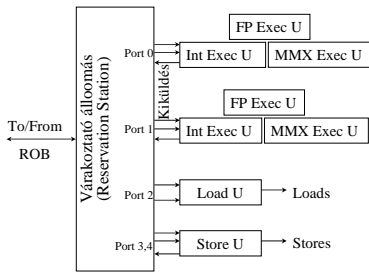
---

---

---

---

## A kiküldés (dispatching)



A várakoztató állomás "kiegyenlítő tartály". Bekerülnek a  $\mu$ ops-ok, és ha minden operandusuk előállt (és megfelelő ALU rendelkezésre áll), a megfelelő porton megfelelő ALU-ba kerülnek végrehajtásra. Onnan az eredményükkel együtt visszajutnak az I Poolba.

Ciklusonként max 5  $\mu$ ops kiküldése lehetséges, átlag 3 a tipikus.

---

---

---

---

---

---

---

---

---

---

---

---

## A várakoztató állomás elv

- A várakoztató állomás elv azt jelenti, hogy a VÁ várakoztatásra, utasításátrendezésre és regiszterátnevezésre szolgál egyidejűleg.
- Minden ALU számára közös a VÁ, ezért a Pentiumok ún. központi várakoztató állomás típusúak.

---

---

---

---

---

---

---

---

---

---

---

---

## A kiküldési politika

- A „kiküldés” feladatai:
  - A VÁ-ban pufferelt  $\mu$ ops-ok közül a végrehajthatók kiválasztása (kiválasztási szabály: aminek minden forrásoperandusa már rendelkezésre áll: spekulatív politika, adatfolyam elvű a kiválasztás),
  - A megfelelő ALU kiválasztás (melyik port)
  - Döntés a kiküldési sorrendről (ha a küldhetőnél több  $\mu$ ops létezik). (A Pentiumnál az Intel nem specifikálta pontosan, FIFO.)
- Létezik sorrendben kiküldés, vagy részben sorrendben kiküldési politika is, de nem a Pentiumoknál.

---

---

---

---

---

---

---

---

---

---

---

---

## A végrehajtó egységek (ALU-k)

- A kiküldött  $\mu$ ops-ok valamely kapun (port) jutnak a VE-ek valamelyikébe
- Szokásos ALU-k:
  - Fixpontos ALU (IU, legalább 2, egy-egy porton)
  - Fixpontos osztó (IDIV), fixpontos eltoló (ISHF)
  - Lebegőpontos összeadó (FADD), osztó (FDIV), szorzó (FMUL)
  - MMX ALU
  - Elágazás kezelő (JEU, elágazási cím számító)
  - Címkszámító adatbetöltéshez (load AU)
  - Címkszámító adat tároláshoz (store AU)
    - Utóbbiakból a kiküldött  $\mu$ ops-ok a BIU-ba kerülnek

---

---

---

---

---

---

---

---

## A Bus Interface Unit

- Az adat betöltő (load) és kiíró (store)  $\mu$ ops-ok ide kerülnek
  - A betöltéshez kell: memória cím, a szélesség és a céloperandus. Egyetlen  $\mu$ ops-sá dekódolható. Spekulatíven végrehajtható.
  - Kiíráshoz kell cím, szélesség és az adat. Két  $\mu$ ops kell, egyik generálja az adatot, másik a címet és szélességet. Nem hajthatók végre spekulatíven. Ezeket nem „zavarhatják” load-ok, régi store-ok.

---

---

---

---

---

---

---

---

## Memory Reorder Buffer

- A BIU-ban van Memory Reorder Buffer: „kiegyenlítő tartály” (ROB-hoz hasonló):
  - Engedi, hogy load-ok előzzenek load-okat;
  - Blokkol függő load-okat és store-okat;
  - Újraütemezi a blokkoltakat, mikor a blokkoló feltételek (függőség, erőforrás hiány) megszűnnek.

---

---

---

---

---

---

---

---

## A VÁ és a ROB „frissítése”

- A végrehajtott egységekből (ill. a BIU MRB-ből) a végrehajtott  $\mu$ ops-ok visszakerülnek a várakozó állomásba (Reservation Station)
  - Asszociatív kereséssel keres olyan  $\mu$ ops-okat, melyek valamely forrásoperandusa az éppen végrehajtott céloperandussal egyezik. Ezeket frissíti: az eredményt beírja a ROB-ba, érvényességi biteket beállítja.
  - A frissítés kell a függőségek megszüntetésére, illetve a visszavonások előkészítésére.

---

---

---

---

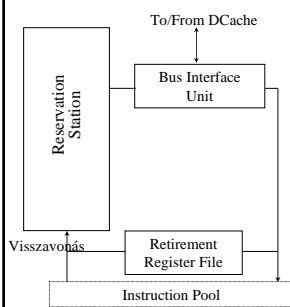
---

---

---

---

## A Retire Unit (Visszavonó)



Ez is nézi a  $\mu$ ops-okat az VÁ-ban: olyanokat keres, amik már végrehajtottak és kivehetők (can be removed). Nézi az eredeti sorrendet (ROB), és a RAT leképzéseket, hogy helyes eredményt kapjunk, figyelemmel az interruptokra, trap-ekre, töréspontokra és a helytelen előrejelzésekre is. A visszavont  $\mu$ ops-ok eredményeit beírja az IA regiszterekbe, vagy az L1 Dcache-be (M/BIU segítségével).

Ciklusonként 3  $\mu$ ops-ot tud visszavonni.

A load/store-ok a MIU-n át jutnak vissza az VÁ-ba.

---

---

---

---

---

---

---

---

## A konzisztencia

- A pentiumoknál erős processzor konzisztenciát biztosítanak
  - A ROB-ba beírt sorrend szerint, azaz az IA instrukció sorrendje szerint történik a visszavonás
- A memória konzisztencia nem erős.
- A kivételkezelés konzisztenciája erős

---

---

---

---

---

---

---

---

## Intel Pentium 4

- **NetBurst Micro-Architecture**
  - Hyper-Pipelined Technology (20-stage)
  - 400 MHz System Bus (3,2 Gbytes/sec)
  - **Advanced Dynamic Execution**
    - 126 instrukció a spekulatív végrehajtás időablakában (P6-nál csak 42)
    - 4K a Branch Target Buffer: részletesebb história az elágazásokról, ezekből jobb előrejelzési algoritmus, kb. 33%-kal kevesebb a téves előrejelzés a P6-hoz képest
  - **Rapid Execution Engine**
    - Az ALU-k a processzoron belül kétszeres frekvenciával dolgoznak

<ftp://download.intel.com/design/pentium4/papers/24943801.pdf>

---

---

---

---

---

---

---

---

---

---

## Intel Pentium 4 (folyt)

- **NetBurst Micro-Architecture (folyt)**
  - **Advanced Transfer Cache**
    - 256 KB L2 Cache, 256 bites interfésszel
    - 48 GB/s érhető el (P6-nál ez csak 16GB/s volt)
  - **Execution Trace Cache**
    - Új L1 Icache: képes dekódolt mops-okat pufferelni (nem kell újra dekódolni)
    - Az elágazások instrukciószekvenciái ugyanazon a cache vonalon: gyorsabb elérés.
  - **Streaming SIMD Extension 2 (SSE2)**
    - SIMD párhuzamosságok mind a fix, a lebegőpontos és MMX aritmetikában
    - 144 új instrukció
  - **Hardware Prefetcher**
    - Konkurencia a memória elérés és a számítások között

---

---

---

---

---

---

---

---

---

---

## Intel® Desktop Processor Roadmap

| 2005. 2. félév                                                                                                                                         | 2006. 1. félév                                                                                                                                                    |
|--------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Intel® Pentium® Processor Extreme Edition 840</b><br>• 2x1MB L2 cache, 3.20 GHz, 800 MHz FSB<br>• Intel® 955X Express Chipset                       | <b>Intel® Pentium® Processor Extreme Edition 840 or greater</b><br>• 2x1MB L2 cache, 3.20 GHz, 800 MHz FSB<br>• Intel® 955X Express Chipset                       |
| <b>Intel® Pentium® D Processor 840</b><br>• 2x1MB L2 cache, 3.20 GHz, 800 MHz FSB<br>• Intel® 955X Express Chipset<br>• Intel® 945G/P Express Chipsets | <b>Intel® Pentium® D Processor 840 or greater</b><br>• 2x1MB L2 cache, 3.20 GHz, 800 MHz FSB<br>• Intel® 955X Express Chipset<br>• Intel® 945G/P Express Chipsets |

<http://www.intel.com/products/roadmap/>

---

---

---

---

---

---

---

---

---

---





## Flynn osztályozása

- **Lehetséges osztályok:**
  - **Single Instruction Single Data stream (SISD)**
    - A Neuman gép ilyen
  - **Single Instruction Multiple Data stream (SIMD)**
    - Adatpárhuzamos processzorok
  - **Multiple Instruction Single Data stream (MISD)**
    - Nem valósult meg, bár egyesek a futószalag elvű gépeket és a szisztolikus tömböket ebbe az osztályba sorolják
  - **Multiple Instruction Multiple Data stream (MIMD)**
    - Osztott memóriás rendszerek
    - Saját erőforrásos, üzenetváltásos rendszerek

---

---

---

---

---

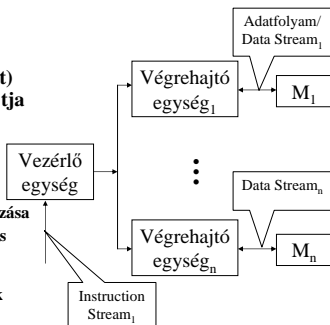
---

---

---

## Az adatpárhuzamosság

- **Az összes processzor (végrehajtó egység) ugyanazt az algoritmust különböző adatokon hajtja végre**
- **Gyakori feladatok**
  - Mátrixok és vektorok elemeinek feldolgozása
  - Kép pixelpontok feldolgozása
  - AB rekordok párhuzamos feldolgozása
  - Hierarchikus fa- és piramisszerkezetű adatok feldolgozása



---

---

---

---

---

---

---

---

## Adatpárhuzamos architektúrák

- **Asszociatív és neurális számítógépek**
  - Processzoron belül említettük az asszociatív memória kezelését. Az elven külső párhuzamosság is elképzelhető
    - Az adatok minden elemét párhuzamosan összevetjük egy AB megfelelő elemeivel, és az egyezés mértékét (egyezés jóságát), vagy az ettől függő eredményt kapjuk meg
  - A neurális hálókról később tanulnak.
- **Szisztolikus és irányított tömbök**
  - A bevitt adatokat „átpumpálják” egy műveletvégző rendszeren (a szív szisztolikus pumpálása analógiájára)
- **Vektorelrendezésű architektúrák**
- **A klasszikus SIMD**
  - Finom, ill durva szemcsézettség

---

---

---

---

---

---

---

---

## A MIMD lehet

- **Közös erőforrású struktúrák** (pl. korábbi zeus)
  - osztoznak a memórián, eszközökön,
- **részben v. teljesen saját erőforrású struktúrák** (pl. transzputerek)
  - processzorokénti memória,
  - esetleg eszközök,
  - intenzív kommunikáció.
- **A párhuzamosság granulátuma lehet**
  - durva szemcsés (pl. processzek egy-egy CPU-hoz kötődnek, ilyen volt a zeus),
  - finom szemcsés (egy processz fonalai más CPU-kon).

---

---

---

---

---

---

---

---

## Két fogalom

- **Grid computing** (Computational grid ; peer-to-peer-computing) (grid = rács)
  - Számítógépek összekapcsolása hálózaton keresztül, mely gépek ugyanazon probléma ugyanabban az időben történő megoldásán egységként dolgoznak
- **Cluster computing** (gépek klasztere)
  - Személyi gépek, munkaállomások együttes használata úgy, hogy egy felhasználó számára egy egységként tűnjenek.

---

---

---

---

---

---

---

---

## Számítógép architektúrák

### Korszerű architektúrák

---

---

---

---

---

---

---

---