

# Principy počítačů a operačních systémů

Zvyšování výkonnosti a vývoj RISC

Zimní semestr 2007/2008

# Další způsoby zvyšování výkonnosti

## Simultánní multithreading

- do pipeline se zavádějí instrukce různých vláken, které na sobě skoro jistě nebudou závislé

## Návrh ISA pro pipelining, multiple-issue, atd.

- důraz kladen na návrh “dobře zpracovatelné” instrukční sady, která umožní realizovat uvedené techniky zvyšování výkonu

# Návrh ISA pro výkonné procesory (1)

## Statistika využití instrukcí (IBM/360)

skupina	zastoupení
přesuny dat	45,28%
řízení	28,73%
aritmetika	10,75%
porovnávání	5,92%
logické operace	3,91%
posuny, rotace	2,93%
bitové operace	2,05%
I/O a ostatní	0,43%

# Návrh ISA pro výkonné procesory (2)

## Další pozorování

- 56% konstant je v rozsahu  $\pm 15$  (5 bitů)
- 98% konstant je v rozsahu  $\pm 511$  (10 bitů)
- 95% podprogramů potřebuje pro předání parametrů méně než 24 bytů

## Výzkum DEC: v typickém programu bylo

- použito 58% instrukční sady
- pro 98% instrukcí stačilo pouhých 15% firmware

# Návrh ISA pro výkonné procesory (3)

## Změna priorit při návrhu ISA

- původní cíl
  - ♦ rozsáhlé instrukční sady, složité instrukce
  - ♦ překlenutí sémantické mezery mezi assemblerem a vyšším programovacím jazykem
- nový cíl
  - ♦ snaha o jednoduché instrukce → rychlé provádění
  - ♦ rychlejší paměti → složité instrukce nejsou klíčové
  - ♦ optimalizující kompilátory vyšších jazyků

# Vznik procesorů typu RISC (1)

---

## Reduced Instruction Set Computer

- omezená a jednoduchá instrukční sada
- velké množství obecných registrů
- důraz na optimalizaci pipeline

## Výzkumné systémy

- RISC-1 (1981, Berkeley University)
- MIPS (1982, Stanford University)
- IBM-801

# Vznik procesorů typu RISC (2)

## Charakteristické rysy RISC procesorů

- instrukční sada
  - ♦ pevný formát instrukce, velké množství registrů
  - ♦ malý počet a jednoduché instrukce a adresovací režimy
- provádění instrukcí
  - ♦ load/store architektura, operace registr/registr
  - ♦ proudové zpracování instrukcí, 1 instrukce na cyklus
  - ♦ zvláštní zpracování skoků
  - ♦ hardwired návrh obvodového řešení
- silná závislost na překladači

# Typické RISC procesory

## Desktop/server

- Alpha (DEC)
- PA-RISC (HP)
- PowerPC (IBM + Motorola)
- MIPS (Silicon Graphics)
- SPARC (Sun Microsystems)

## Embedded systems

- ARM (Advanced RISC Machines)
- Thumb (Advanced RISC Machines)
- SuperH (Hitachi)
- M32R (Mitsubishi)
- MIPS16 (Silicon Graphics)



# Procesory typu CISC

---

## Complex Instruction Set Computer

- zachycuje původní trend vývoje
  - ♦ přesun složitosti do HW pro snažší programování
  - ♦ méně instrukcí pro daný úkol znamená méně přístupů do (pomalé a drahé) paměti
  - ♦ implementace pomocí mikrokódu se dá snadno změnit
- zpětné označení jako alternativa k RISC

# Konvergence CISC a RISC

## Důsledek vývoje technologie i znalostí

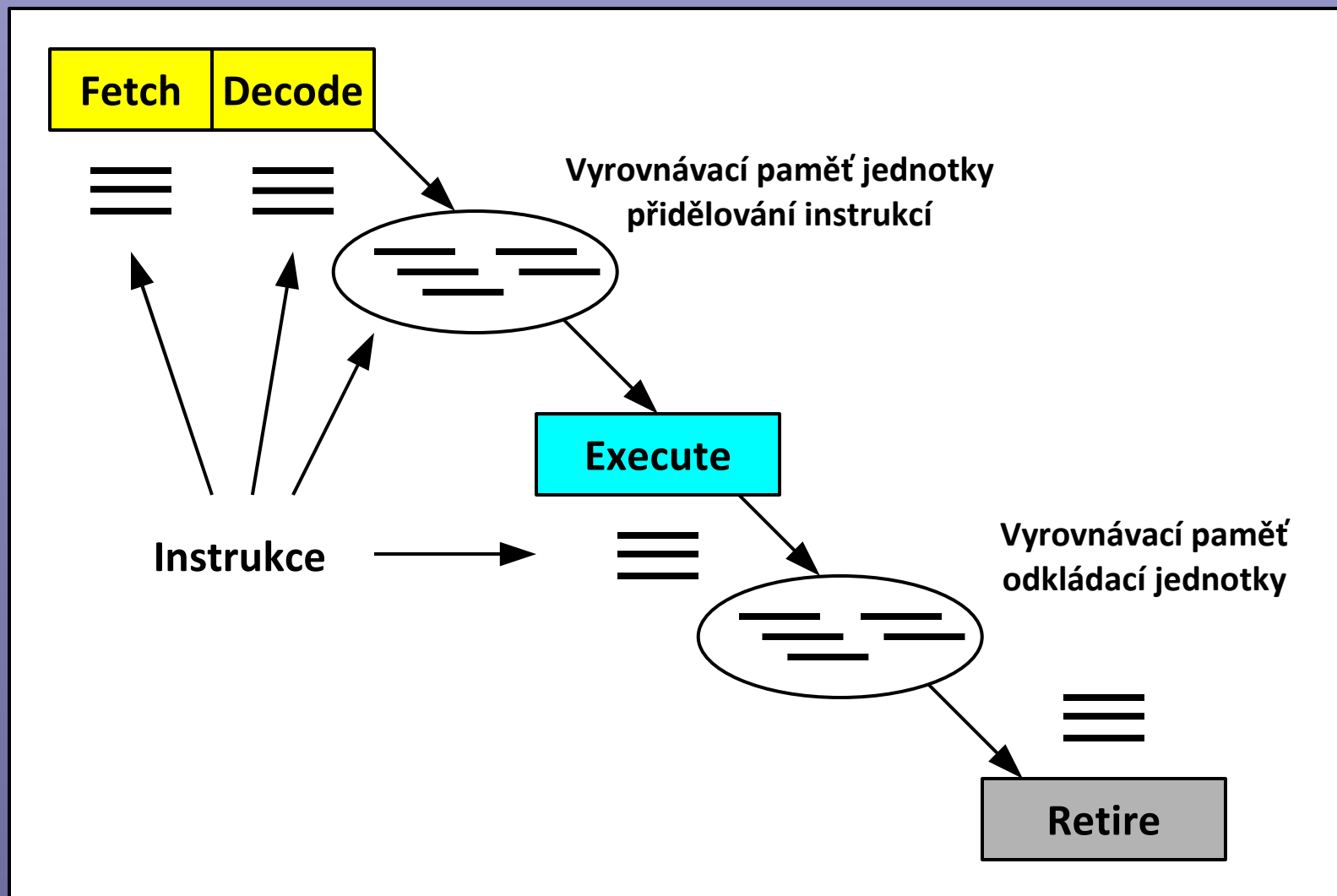
- mnohé techniky jsou používány v obou typech
  - ♦ CISC – schopné vykonávat v 1 taktu více instrukcí,
  - ♦ RISC – vyplnění komplikovanějšími instrukcemi
- vznik „post-RISC“ procesorů
  - ♦ kombinace obou přístupů s novými metodami
  - ♦ zachovává charakteristické vlastnosti RISC

# Post-RISC procesory

## Rozdíl od superskalárních RISCů

- přidání ne-RISCových instrukcí (pro zvýšení výkonu)
- agresivní přerovnávání instrukcí při zpracování
  - ♦ out-of-order/speculative execution
  - ♦ odklon od závislosti na kompilátoru
- nové uspořádání
  - ♦ nové jednotky, větší míra paralelismu
- výkon RISC vs. post-RISC
  - ♦ RISC: dán stupněm paralelismu
  - ♦ post-RISC: dán počtem najednou odkládaných instrukcí

# Proudové zpracování v post-RISC



# Přechod RISC – Post-RISC

Company	Old Processor	SpecInt92	New Processor	SpecInt92	% Increase Clock Rate	% Increase SpecInt92
DEC	21064a 300 MHz	220	21164 333 MHz	400	11%	82%
HP	PA-7150 125MHz	136	PA-8000 133MHz	360	6%	164%
IBM	PPC-601 80MHz	91	PPC-604 133MHz	176	66%	93%
Intel	Pentium 166MHz	198	P6 200MHz	320	20%	62%
Sun	HyperSparc 125MHz	131	UltraSPARC 140MHz	200	12%	52%
MIPS	R-4400 200MHz	141	R-10000 200MHz	300	0%	113%

# Dataflow CPU vs Post-RISC

