

PA039: Architektura superpočítačů a náročné výpočty

Luděk Matyska

Fakulta informatiky MU

Jaro 2016

- Účast na přednáškách není povinná
- Zkouška
 - Pouze písemná, 90 minut
 - Termíny budou k dispozici během dubna
- Kolokvium
 - Projekt, nutno se přihlásit před koncem března

High Performance Computing

- Formule 1 v oblasti počítačů
 - Velmi drahé stroje, ovšem špičkových parametrů (výkonu)
- Specifické uživatelské skupiny
 - Rozsáhlé simulace
 - Modelování (automobily, letadla, ...)
- S jídlém roste chuť
 - Požadavky rostou rychleji než výkon procesorů
 - Roste ale i složitost procesorů

Kvalita programování určuje použitelnost

High Performance Computing II

- Procesory
 - CISC
 - RISC
 - Vektorové procesory
 - Streaming procesory (např. GPU)
 - Speciální systémy FPGA, ...).
- Paměti – výkon se zpožďuje za procesory

- Klesá poměr teoretický_výkon/dosažený_výkon
- Reakce: je třeba lépe pochopit
 - architekturu použitého počítače;
 - příčiny, proč určitý kód je podstatně rychlejší než zdánlivě ekvivalentní varianta;
 - způsoby měření reálného výkonu (programu a/nebo procesoru)

High Throughput Computing

- Nejvyšší aktuální výkon versus Nejvyšší využití
 - dlouhodobé efektivní využití počítačových systémů
 - velké množství menších úloh
 - Není kritická rychlost zpracování jedné úlohy
 - Podstatný celkový čas zpracování
 - Efektivita
 - maximalizace „investice“
 - celková propustnost systému

PA039: Architektura superpočítačů a náročné výpočty

Procesory a paměti

Luděk Matyska

Fakulta informatiky MU

Jaro 2016

Základní aspekty – co určuje výkon

- Latence (zpoždění)
 - zpracování/přenos signálů uvnitř procesorů či paměti
 - přenos dat mezi procesorem a pamětí
 - zpoždění přímo v paměti
- Rychlost obnovení (cycle times)
 - rychlost přepínání obvodů
 - frekvence obvodů (vnitřní „hodiny“)
 - obnovení paměti (dynamická paměť)
- Propustnost (rychlost přenosu jednotky dat)
 - rychlost přenosu dat na chipu
 - počet instrukcí per cyklus
 - rychlost přenosu mezi komponentami
- Granularita
 - hustota na chipu
 - hustota paměti
 - velikost úlohy

Complex Instruction Set Computer

- Příklady:
 - PDP 11, VAX, IBM 370, Intel 80x86, Motorola 680x0, . . .
- Princip:
 - Nedělej programem to, co může udělat hardware
- Pojem CISC fakticky vytvořen až jako protiklad proti RISC procesorům

Důvody existence

- Velikost a rychlost paměti
 - Srovnání s rychlostí samotných procesorů
- Přímá podpora překladačů
- Adresování (přístup k paměti)

Mikroprogramování

CISC – složité instrukce

- Řídící část procesoru příliš rozsáhlá
 - Mikroinstrukce: Dekompozice na jednodušší instrukce
- Složitá instrukce == mikroprogram

Jednodušší návrh hardware

- Instrukce jsou *emulovány*

Je možno „snadno“ změnit instrukční sadu konkrétního počítače

⇒ *rodina počítačů* (IBM 360, 370, VAX, ...)

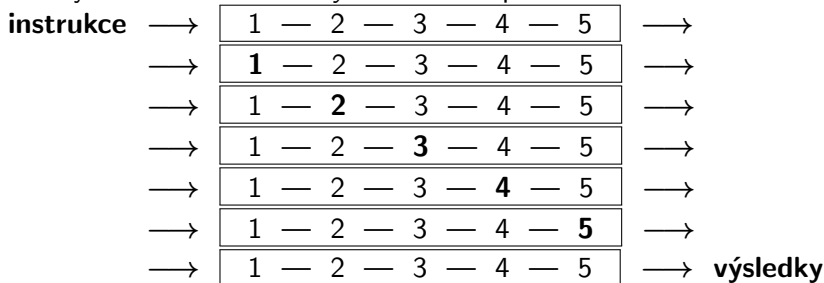
Nevýhody: příliš složité instrukce, stále složitější analýza instrukcí, zátěž zpětné kompatibility (v rámci rodiny)

Zvyšování výkonu

- Rychlost hodin udává výkon procesoru
 - Omezeno aktuálními technologickými možnostmi
 - Nelze neomezeně zvyšovat
 - Závislosti mezi komponentami
 - Rychlost šíření signálu
- Řešení: **paralelizace procesů**

Pipelining

Překrývání instrukcí v *různých fázích* rozpracovanosti



Tři základní oblasti:

- 1 Zpracování instrukcí
- 2 Přístupy k paměti
- 3 Výpočty v pohyblivé řádové čárce

Pipelining II

Běžný rozklad instrukcí (pětiúrovňový pipelining):

Instruction Fetch instrukce je načtena z paměti

Instruction Decode instrukce je rozeznána (dekódována)

Operand Fetch jsou připraveny operandy (načteny z registrů a/nebo paměti)

Execute instrukce je provedena

Writeback výsledky jsou zapsány zpět (do registrů a/nebo paměti)

Jednotlivé instrukce jsou zpracovávány paralelně, s posunem o jednu fázi pipeline.

Pipelining a paměť

- „Neviditelný“ pipelining
 - Předsunutí čtení (zápisu) z (do) paměti před vlastní instrukci pracující s daty
- „Viditelné“ pipelines
 - Explicitní instrukce, s přesně definovaným počtem cyklů do dokončení.
 - Např. Intel 80860

Reduced Instruction Set Computer

- První RISC: CDC 6600 (Seymour Cray)
 - První polovina 60. let (1964)

Explicitní RISC koncept představují osmdesátá léta

- Podmínky vzniku RISC systémů
 - Zavedení vyrovnávacích pamětí (cache)
 - Dramatický pokles ceny a vzrůst velikosti hlavních pamětí
 - Lepší pipelining
 - Kvalitně optimalizující překladače

RISC podmínky II

- Rychlost přístupu k paměti přestala být (hlavním) úzkým místem
 - využití vyrovnávacích pamětí (cache)
 - využití interních registrů (méně přímých přístupů do paměti)
- Velikost programu přestala být podstatná (i rozsáhlé programy se snadno vejdou do paměti)
- Problém: *zadržení* (stall) při čekání na výsledek předchozí instrukce (v CISC příliš složité vazby)
- Není třeba složitých instrukcí (naopak); čitelnost assembleru přestává být podstatná

Charakteristiky RISC

- Jednotná délka instrukcí
- Pečlivý výběr skutečně používaných instrukcí
- Jednoduché adresní módy
- Architektura Load/Store
- Dostatek registrů
- „Odložené“ skoky (delayed branches)
- Příklady:
 - Na začátku předchůdci MIPS (Stanford) a SUN SPARC (UoC, Berkeley) architektury
 - IBM s její Power Architecture (dnes PowerPC a POWER7)
 - HP s PA-RISC
 - DEC Alpha
 - Intel i860 a i960 či Motorola 88000
 - ARC, ARM, ...

RISC – pokročilý návrh

- Ideál RISC první generace:
 - Jedna instrukce každý tik hodin
- Dnešní realita:
 - Více jak jedna instrukce na tik

Nové vlastnosti

- Superskalární
- Superpipeline
- (Velmi) dlouhé instrukce ((Very) Long Instruction Word, (V)LIW)

Superskalární procesory

- Vícenásobné procesní jednotky
 - Aritmetické (ALU), Floating point (FPU) a další
- Příklady:
 - RS/6000, SuperSPARC a vyšší, Motorola 88110, HP PA 7100 a vyšší, DEC Alpha, MIPS R8000 a vyšší, Intel Pentium, IBM P4, P5

Superskalární procesory – vlastnosti

- Paralelismus v hardware
 - Sekvenční programy
 - „Automatická“ paralelizace technickými prostředky
 - Současné načtení více instrukcí
 - Instrukce MADD (Multiply Add)
 - Operace $X*Y+Z$

Superpipeline

- Další zjednodušení obvodů
 - Rozsáhlejší dekompozice pipeline
 - Rychlejší provádění jednotlivých částí
- Výsledkem rychlejší výpočet
 - Jiná forma paralelismu
- Nazývány též *hluboké* (deep) pipelines

- Obdoba superskalárních (mnoho jednotek)
- Paralelizace pod kontrolou překladače
 - nárůst složitosti překladačů
 - zjednodušený hardware dovoluje vyšší výkon
 - rozhodnutí které instrukce smí běžet paralelně je na překladači
- Výhody:
 - Jednodušší instrukce
 - Není třeba složitý řídicí hardware
 - Potenciál pro nižší spotřebu energie
- Příklady:
 - Intel i860
 - triMedia media processors
 - C6000 DSP family (Texas Instruments)
 - Itanium IA-64 EPIC (částečně)
 - Crusoe procesory firmy Transmeta
 - Ruské superpočítače Elbrus

RISC – další rysy

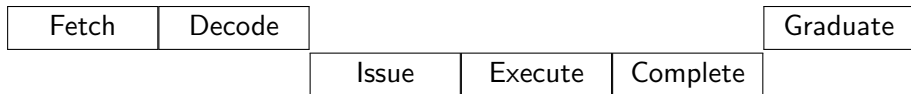
- Obcházení registrů
- Přejmenování registrů
- Skoky
 - nulování operace
 - podmíněné přiřazení ($a = b < c ? d : e;$)
 - vícenásobné „předčtení“ z paměti
 - buffer potenciálních cílů skoku
 - předpověď cíle skoku za běhu
 - statistická (předem dána)
 - dynamická

Architecture with **N**on-sequential **D**ynamic **E**xecution **S**cheduling

- Východiska
 - Zpomalení způsobeno čekáním na data
 - Dynamický paralelismus
- Příklady
 - HP PA 8000, MIPS R10000, ...

- Vícenásobné fronty instrukcí
 - celočíselná fronta pro celočíselné instrukce
 - adresní fronta pro operace Load/Store
 - fronta pohyblivé řádové čárky
- Nezávislá pipeline pro každou frontu
- Vlastnosti
 - instrukce vybírány podle *připravenosti*
 - není dodrženo pořadí instrukcí v programu
 - *dokončení* instrukcí zajišťuje správné uspořádání

ANDES – Spekulativní výpočet



ANDES – Další vlastnosti

- Spekulativní skoky:
 - výpočet pokračuje *předpovězenou* větví
 - nečeká na výsledek instrukce
- Neblokující Load/Store
- Přejmenování registrů

- Organizace paměti:
 - řádky a sloupce (matice)
 - adresa má dvě části
 - *page mode* – naráz čtena skupina souvisejících bytů

Vlastnosti pamětí

- Přístupová doba (memory access time)
 - vystav řádek **plus** vystav sloupec **plus** vystav data
- Cyklus paměti (memory cycle time)
 - určuje, jak často lze data číst
- Obé závisí na typu paměti (dynamická vs. statická)

Virtuální paměť

- Fyzická vs. logická adresa
 - Více adresních prostorů
- *Translation Lookaside Buffer (TLB)*
 - překlad logických adres na fyzické
 - součást hardware
 - TLB výpadky (misses)
- (Ne)použití v superpočítačích

Vyrovnávací paměť

- Hit poměr
- Velikosti 4 KB–16 MB
- Organizace: řádky pevné délky, 16–128 bytů
- Typy:
 - přímo adresovatelná (direct mapped)
 - množinově (částečně) asociativní (set-associative)
 - plně asociativní (fully-associative)

- *Harvard Memory Architecture*
 - oddělení paměti pro data a pro instrukce
- Programově ovládaná vyrovnávací paměť
 - řízení u (některých) superskalárních procesorů (DEC Alpha)

Přímo adresovatelná vyrovnávací paměť

- Statické mapování
 - každý řádek vyrovnávací paměti odpovídá předem určeným oblastem hlavní paměti
- Rychlé
- Jednoduché obvody
- Potenciálně neefektivní

Plně asociativní vyrovnávací paměť

- Dynamické mapování
 - asociativní paměť
 - každý řádek vyrovnávací paměti zná adresy „svého“ bloku
 - současný dotaz na všechny řádky
 - výběr řádku pro zneplatnění
- Velmi efektivní
- Velmi složité obvody – drahé

Částečně asociativní vyrovnávací paměť

- Množina přímo adresovatelných vyrovnávacích pamětí
- Kombinace lepších vlastností obou extrémních přístupů
 - zpravidla 2 a 4 cestné

- **Bandwidth** = maximální propustnost paměťového systému

- Měřena v bytech za sekundu

Propustnost není stejná mezi všemi komponentami

- Procesor – vyrovnávací paměť – hlavní paměť – externí paměť

- Zpoždění (Latence)

- Doba mezi časem požadavku a časem přísunu dat
- Zvláště významná pro přesun malých objemů dat

Prokládaná (Interleaved) paměť

- Rozdělení na menší bloky
 - Následující adresy mapovány do různých bloků
 - Umožňuje okamžitý přístup
- Běžné dvou až osminásobně prokládané paměťové subsystémy
 - superpočítače mají vícenásobné prokládání
 - Příklad: Convex C3 s 256 násobným prokládáním
 - Hodiny 16 ns
 - Opakovaný přístup k témuž banku: 300 ns (téměř 20 násobné zrychlení)
- Vyšší latence
 - Odstíněna použitím pipeline

Přeskládání přístupů k paměti

- Předchůdce ANDES
- Minimalizace následných přístupů do týchž banků paměti
- Kontrola závislostí Load a Store při běhu programu
- Příklad: Motorola 88110

Processor MIPS R8000

- Zaveden 1993
- Čtyřnásobná superskalární architektura, max 6 operací/cyklus
 - Zdvojená ALU, zdvojená FPU a dvě Load/Store jednotky
 - FPU s IEEE-754 standardní aritmetikou s nepřesným přerušením
 - 32 registrů (64 bit) pro celočíselné a 32 registrů (64 bit) pro float operandy
 - Podmíněné move instrukce (pro IF příkazy)
- Plně 64bitová architektura
 - 128-bit datová sběrnice
 - 40 bitová adresní sběrnice (max 1 TB fyzické paměti)
 - TLB dvoucestný, s 384 položkami

MIPS R8000 (II)

- Vyrovnávací paměti
 - 16 KB I-cache (instrukce)
 - 16 KB D-cache (dvoucestná, pouze pro celočíselná data)
 - 2 KB branch prediction cache
 - 4 MB streaming cache (výpočty v pohyblivé čárce)

- Vyrovnávací paměť instrukcí
 - Přímá adresovatelná
 - 1024 položek po 128 bitech
 - Adresována i označena (tagged) virtuální adresou
 - Obchází TLB
 - tag RAM – 512 položek (pro každý řádek)
 - příznak
 - ASID (Address space identifier)
ASID rozlišuje shodné virtuální ale různé fyzické adresy
 - bit platnosti
 - dva bity oblasti

- Vyrovnávací paměť pro data
 - Přímá adresovaná
 - Dva paralelní přístupy
 - 2 load nebo jedna load a jedna store instrukce současně
 - Adresována virtuální, označena fyzickou adresou
 - Write-through protokol

MIPS R8000 (IV)

Srovnání vyrovnávacích pamětí

Parametr	I-cache	Branch	D-Cache	TLB
Umístění	IU	IU	IU	IU
Velikost	16 KB	2 KB	16 KB	
Položka	128 bit	16 bit	64 bit	
Počet položek	1024	1024	2048	384
Počet portů	jeden	jeden	dva	dva
Mapování	přímé	přímé	přímé	3-cestné
Index	Virtuální	Virtuální	Virtuální	Virtuální
Tag	Virtuální	N/A	Fyzická	N/A
Přístup	jeden cyklus	jeden	jeden	jeden
Šířka	128 bit	16 bit	64 bit	
Propustnost	1,2 GB/s	159 MB/s	1,2 GB/s	
Řádek	32 bytů	N/A	32 bytů	
Miss penalty	11 cyklů	3 cykly		

MIPS R8000 (V) – Rychlost provádění operací

Celočíselné	Latence	
Add, shift, logical	1	
Load, store	1	
Multiply	4 (6)	
Divide	21	(jmenovatel ≤ 15 bitů)
	39	(jmenovatel 16–31 bitů)
	73	(jmenovatel 32–64 bitů)

Reálné	Latence	Zdržení
Move, negate, abs value	1	1
Add, Multiply, MADD	4	1
Load, Store	1	1
Compare, cond. move	1	1
Divide	14 (20)	11 (17)
Square root	14 (23)	11 (20)
Reciprocal	8 (14)	5 (11)
Reciprocal sq. root	8 (17)	5 (14)

Processor MIPS R10000

- Zaveden 1996
- ANDES architektura, tři fronty
- Superskalární, 4 instrukce současně
 - 2 ALU a 2 FPU (neekvivalentní)
 - FPU s IEEE-754 standardní aritmetikou a přesným přerušením
 - 32 (64 fyzických) registrů (64 bit) pro celočíselné operandy,
 - 32 (64 fyzických) registrů pro float operandy
 - přejmenování registrů
- Plně 64 bitová architektura
 - 128 bit datová sběrnice, 40 bitová adresní sběrnice
 - TLB plně asociativní, 64 položek (zdvojených) velikost stránky 4 KB–16 MB

MIPS R10000 (II)

- Vyrovnávací paměti
 - 32 KB I-cache (2-set associative)
 - 32 KB D-cache (dvoucestná, 2-set associative)
 - předpověď skoků (4 úrovně)
 - 1 MB L2 cache
- Neblokující instrukce `load` a `store`

MIPS R10000 (III)

- Výpočetní jednotky
- 2 ALU
 - Společně
 - Součet, Rozdíl a Logické operace
 - Rozdílné
 - ALU1: skoky a operace posunu
 - ALU2: násobení a dělení (iteračně)
- 2 FPU (Další dvě jednotky (bez pipeline) pro dělení a odmocninu (iteračně))
 - FPU1: sčítačka
 - FPU2: násobička

- **Celočíselná**

- 16 položek
- až 4 instrukce současně zapsány

- **Float**

- 16 položek
- až 4 instrukce současně zapsány
- nelze současně zahájit Divide a Square root instrukce
- MADD instrukce projde oběma FPU

- **Adresní**

- 16 položek (FIFO)
- instrukce spustitelné v libovolném pořadí
- zápis a vyjmutí musí být sekvenční (zajištěno FIFO bufferem)
- znovuspuštění instrukce při neúspěchu (cache miss, konflikt, závislost)

MIPS R10000 (V) – Rychlost provádění operací

Celočíselné	Latence	Zdržení
Add, shift, logical, branch	1	1
Load, store	2	1
Multiply (32 bit)	5–6	6
Multiply (64 bit)	9–10	10
Divide (32 bit)	34–35	35
Divide (64 bit)	66–67	67
Int to Float (32 bit)	4	1

Reálné	Latence	Zdržení
Move, negate, abs value	1	1
Add, Conversion, Mult	2	1
Load, Store	3	1
MADD	4	1
Divide	12 (19)	14 (21)
Square root	18 (33)	20 (35)
Reciprocal sq. root	30 (52)	20 (35)

Processor UltraSPARC-I

- Zaveden 1987 (Sparc V9)
- Čtyřnásobná superskalární architektura
 - 2 ALU, FPU (2 instrukce), GRU (Grafika)
 - 32 FPU (64 bit) registrů
- 64bitová architektura; možnost volby little a big endianu
 - 128 bitová datová sběrnice, 41 bitů fyzická adresa, 44 virtuální adresa
 - 64 položek v TLB, stránky s 8 K, 64 K, 512 K nebo 4 MB
- Visual Instruction Set

UltraSPARC-I (II)

- Vyrovnávací paměti
 - 16 KB neblokující D-cache
 - 16 KB I-cache (s predikcí skoku)
 - 0,5–4 MB L2 cache (propustnost 3,2 GB/s)
- Blokující load/store instrukce

UltraSPARC-I – výpočetní jednotky

- FPU

- Dělení a odmocnina samostatné (mimo FPU pipeline)
- 12 (22) cyklů pro jednoduchou (dvojnásobnou) přesnost
- neblokuje pipelinované FPU instrukce
- přesná přerušení

- GRU

- 16 a 32 bitové shlukované sčítání a boolovské instrukce
- 8 a 16 bitové násobení
- skládání a rozbor dat
- přímý přístup k (grafické) paměti obcházející D-cache
- přímá podpora „motion compensation“.

- 32bitová architektura (IA32) CISC
 - Vychází z 16bitového 8086 + 8087 a 80286
 - 80386 (i386), i486, Pentium (i586), ...
- 2001: Itanium (IA64)
 - nově navržená, zpětně nekompatibilní 64bitová architektura
 - spolupráce s HP, převzata řada znaků RISC
- 2003–2004: AMD Opteron a Intel Xeon Nocona
 - konzervativní rozšíření IA32
 - AMD64, EM64T/Intel64, neutrálně x86-64

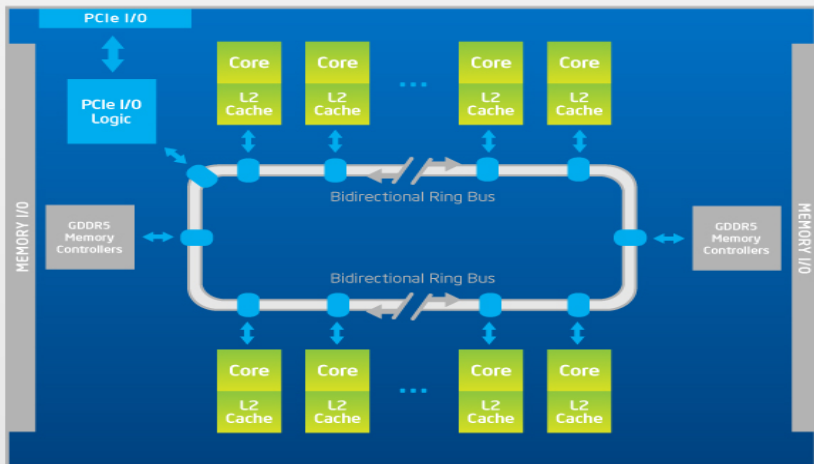
Intel Itanium

- Vlastnosti 1. generace (do 2001)
 - spekulativní vyhodnocení, predikce skoků, přejmenování registrů
 - hrubozrnný multithreading
 - 128 64 bit int a 128 82 bit float registrů
 - až 6 instrukcí v taktu
 - 6 ALU jednotek, 4 MADD jednotky
 - speciální instrukce pro multimédia apod.
 - hardwarová podpora virtualizace
 - pomalá emulace IA32, chybějící kompilátory, průměrný výkon
- Druhá generace (2002–2010)
 - společný vývoj s HP
 - určen spíše pro podnikové systém ynež HPC
 - poslední verze (Tukwila) na 65nm
 - Intel QuickPath propojení (místo sběrnice)
 - výrazné posílení paměťového subsystému, 4 jádra
- Itanium 9500 (2012)
 - 32nm, 8 jader, až 54 MB vyrovnávací paměť
 - naznačena postupná konvergence s Intel Xeon procesory

Současné procesory x86-64

- Označení Sandy Bridge (32nm) a Ivy Bridge (22nm)
- paměť
 - 3–4 paměťové kanály
 - 32+32 kB L1 cache, 4/8 cestná asociativní, privátní
 - 256 kB L2 cache, 8 cestná, privátní
 - až 24 MB L3 cache, 16 cestná, sdílená mezi jádry
- 4–8 jader, hyperthreading
 - cca. 10 paralelních výkonných jednotek
 - buffer cílů skoku
 - fúze instrukcí (např. porovnání + skok)
 - dekódování na mikroinstrukce (podobné MIPS), mikrofúze
 - out-of-order spekulativní vyhodnocení
 - AES instruction set, SHA-1
 - Advanced Vector Extensions, 256bitové instrukce

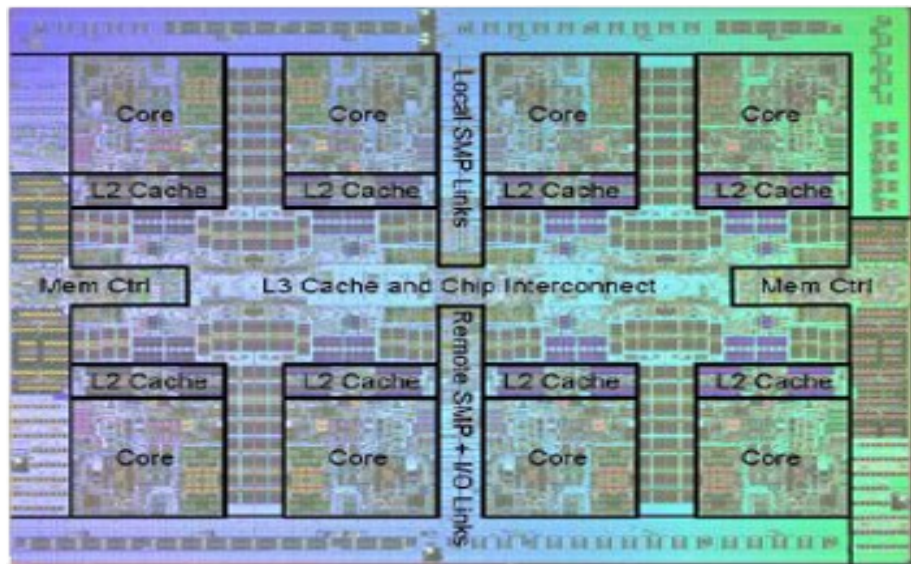
Intel® Xeon Phi™ Coprocessor Block Diagram



IBM Power7 processor

- vyvíjen pro HPC, až 8 jader
 - 12 procesních jednotek, 4 vlákna na jádro
- Parametry (45nm)
 - 256 KB L2 na jádro
 - 32 MB eDRAM sdílená L3 přes chip
 - Duální DDR3 paměťové kontroléry
 - 100 GB/s udržitelná propustnost na chip
 - 360 GB/s SMP propustnost per chip
 - frekvence až 4,25 GHz (kapalinou chlazené)

Power7

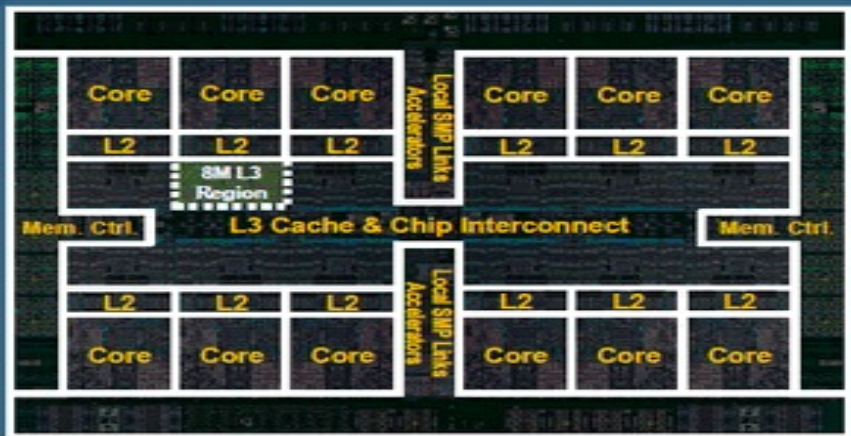


POWER8 Innovation

	POWER5 2004	POWER6 2007	POWER7 2010	POWER7+ 2012	POWER8
					
Technology	130nm SOI	65nm SOI	45nm SOI eDRAM	32nm SOI eDRAM	22nm SOI eDRAM
Compute					
Cores	2	2	8	8	12
Threads	SMT2	SMT2	SMT4	SMT4	SMT8
Caching					
On-chip	1.9MB	8MB	2 + 32MB	2 + 80MB	6 + 96MB
Off-chip	36MB	32MB	None	None	128MB
Bandwidth					
Sust. Mem.	15GB/s	30GB/s	100GB/s	100GB/s	230GB/s
Peak I/O	3GB/s	10GB/s	20GB/s	20GB/s	48GB/s

Technology

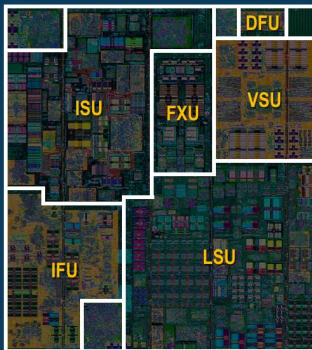
- 22nm SOI, eDRAM, 15 ML 650mm²



POWER8 Core

Execution Improvement vs. POWER7

- SMT4 → SMT8
- 8 dispatch
- 10 issue
- 16 execution pipes:
 - 2 FXU, 2 LSU, 2 LU, 4 FPU, 2 VMX, 1 Crypto, 1 DFU, 1 CR, 1 BR
- Larger Issue queues (4 x 16-entry)
- Larger global completion, Load/Store reorder
- Improved branch prediction
- Improved unaligned storage access



Larger Caching Structures vs. POWER7

- 2x L1 data cache (64 KB)
- 2x outstanding data cache misses
- 4x translation Cache

Wider Load/Store

- 32B → 64B L2 to L1 data bus
- 2x data cache to execution dataflow

Enhanced Prefetch

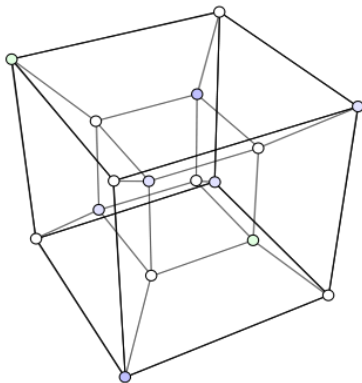
- Instruction speculation awareness
- Data prefetch depth awareness
- Adaptive bandwidth awareness
- Topology awareness

Core Performance vs. POWER7

~1.6x Single Thread
~2x Max SMT

Víceprocesorové systémy

- Frekvenci už nelze příliš zvyšovat
 - Zvyšování výkonu zvýšením počtu jader
 - Propojení více procesorů (socketů)



- Míra škálování (počet socketů)
 - AMD: 4, Intel 8, IBM 32
 - vlastní řešení HP (Intel) 8, Bull 16, SGI \sim 100
- Distribuovaná paměť
 - centralizovaná by byla úzkým místem
 - NUMA (Non-Uniform Memory Architecture)

- Koherence cache
 - přečtu, co jsem sám zapsal
 - přečtu, co zapsal dříve někdo jiný
 - pořadí zápisů vidí všichni stejné
- Stavby řádků cache
 - uncached, shared, modified, ...
- Protokoly udržování koherence
 - adresářové
 - snooping