

Procesory

Struktura současných procesorů
(x86/64). Techniky optimalizace
provádění instrukcí, snižování spotřeby,
Rozšířené instrukční sady.

- Procesor jakožto elektronický obvod (složený z velkého množství součástek - převážně tranzistorů) je umístěn na křemíkové destičce
- CPU je složitý integrovaný obvod
- Populární díky výpočetní technice (PC, laptop, mobilní zařízení, ...), dnes se s nimi můžeme setkat i třeba v pračce nebo ledničce
- Procesor zpracovává informace (instrukce) v podobě strojového kódu a procesory s obdobnou strukturou jádra, které zpracovávají shodný strojový kód, mají stejnou architekturu

Vnitřní architektura

- Podle vnitřního uspořádání dělíme procesory na RISC a CISC lišící se počtem instrukcí, které procesor umí vykonat
- Současné procesory Intel, nebo ty, které jsou s nimi kompatibilní, jsou vnitřně RISC, ač uvnitř vlastně probíhají složité CISC instrukce
- Dále dovedou měnit, resp. Aktualizovat, svůj řídicí mikro kód
- Architektura jde ruku v ruce s instrukční sadou a její délkou, jež díky překotnému vývoji roste
- Jednodušší procesory dodnes používané jsou 8bitové, a naopak na vrcholu jsou dnes 64bitové, používané hlavně ve výkonnějších výpočetních strojích a serverech

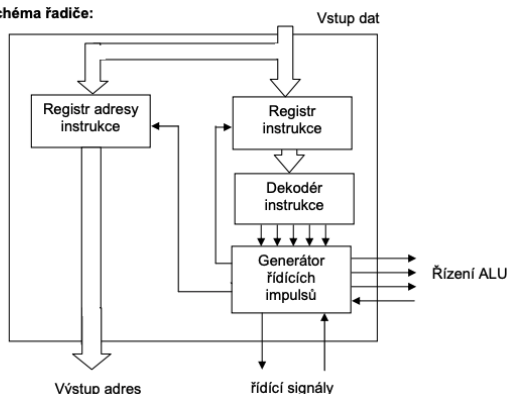
Struktura moderních CPU

Vícejádrový procesor

- Vícejádrový procesor je mikroprocesor, který v jednom pouzdře nebo na jednom čipu integruje více CPU => jádra
- Obvykle jde o jádra, která jsou vzájemně programově kompatibilní a mohou ta snáze spolupracovat
- Jejich společná činnost se nazývá multitasking, případně multithreading
- U vícejádrových procesorů je dobrá součinnost důležitá, protože umožňuje využít celý systém v maximální možné míře
- Spolupráce jader se obecně děje prostřednictvím operační paměti (RAM)
- Přirozené zefektivnění spolupráce jader nastává v případě společné sdílené paměti cache, která umožňuje další zlepšení spolupráce příslušných jader
- Pro využití vícejádrových procesorů je obvykle třeba podpora ze strany OS

Řadič

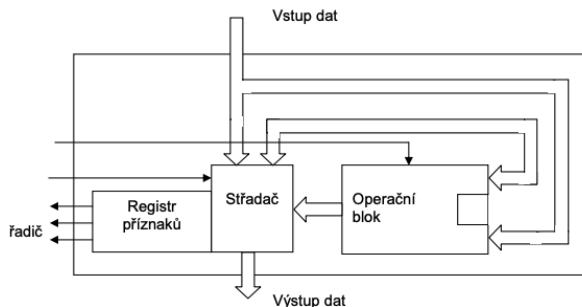
Blokové schéma řadiče:



- Registr adres instrukcí
 - 0 až do hodnoty adresy poslední paměťové buňky v OP
 - Touto hodnotou je omezena OP (operační paměť), kterou lze k CPU připojit
 - Např. u 32bitového registru lze adresovat max. $2^{32} = 4 \text{ GB}$
 - Při zapnutí počítače RAI nastaví na výchozí hodnotu, zpravidla nulu
 - Tímto nulovým obsahem se na výstupu RAI objeví číslo, které odpovídá adrese první instrukce uložené v OP
 - Výstup RAI je spojen s adresovou sběrnicí
- Registr instrukce:
 - Do tohoto registru se po datové sběrnici přivádí z OP instrukce
 - Zde se uloží až do okamžiku, než je přepsaná instrukcí následující
- Dekodér instrukce:
 - Instrukce je přivedena z registru na dekodér, kde se dekoduje a postupuje do generátoru řídicích impulsů
 - Procesem dekódování se rozumí nalezení mikrokódu k vykonání instrukce ve vnitřní paměti ROM řadiče
- Generátor Řídicích impulsů
 - Spuštění mikrokódu se v určité časové posloupnosti generují řídicí impulsy do ostatních jednotek počítače

ALU

- Provádí s daty příslušné aritmetické a logické operace



-
- Operační blok
 - Zpracovává operandy přiváděné na dva vstupy a výsledek se předává jedním výstupem k dalšímu zpracování
- Střadač
 - Je registr, v němž se uchovávají data
 - 1. operand, mezivýsledky, výsledky
- Registr příznaků
 - Stavový registr
 - Skládá se z řady jednobitových pamětí, ve kterých je uložena 0 nebo 1, podle výsledků
 - Obsah stavových registrů kontroluje řadič, který na ně příslušně reaguje
 - Nejčastější příznaky:
 - CARRY
 - ZERO
 - SIGN

Cache v procesoru

- Ukládá kopie dat přečtených z adresy v operační paměti
- Pokud při čtení obsahu slova z adresy v paměti je tato položka nalezena v cache paměti, je její obsah přečten z cache paměti, a ne z operační paměti
- Mezi procesorem a cache pamětí se přenášejí jednotlivá slova, mezi cache pamětí a operační pamětí se přenášejí rámce slov o velikosti několikanásobku velikosti slova procesoru
- Protože asi 90 % operací procesoru je čtení paměti, většinou sekvenční, je tímto způsobem dosaženo větší propustnosti dat z operační paměti do procesoru => vyšší výpočetní výkon
- Vyrovnávací paměť procesoru bývá dvojstupňová
- Část paměti o malé kapacitě je přímo součástí procesoru a je stejně rychlá, jako vlastní procesor (L1)
- Další paměť, pomalejší, ale s větší kapacitou, je mezi procesorem a operační pamětí, dnes se již umísťuje do pouzdra s procesorem (značí se L2)
- Protože cena paměti stoupá s její rychlostí, je možné tímto uspořádáním najít kompromis mezi cenou a rychlostí
- Na přelomu 2008 a 2009 se začíná používat L3 cache i v běžných procesorech, která je pro všechny jádra společná a většinou má velikost několik megabajtů

iGPU

- Integrovaná grafická karta
- Dvě skupiny
- Úzce vázané, které sdílí paměť s procesorem, se kterým také mohou bezprostředně spolupracovat a ty, které třeba i sdílí část operační paměti, ale nejsou schopny úzce spolupracovat
- Systém s úzce vázanými procesory je zvláště výhodný v případě výpočetně náročných operací, kdy CPU a GPU spolupracují na stejných datech
- V tomto případě odpadá nadbytečné kopírování dat mezi pamětí CPU a GPU
- Výhodou tohoto řešení je kromě lepší spolupráce i nižší spotřeba než v případě dedikovaných grafických karet

Severní můstek (Northbridge)

- Systémový řadič
- Jedním ze dvou základních čipů na základní desce
- Zajišťuje komunikaci mezi CPU, pamětí RAM (řadič paměti), AGP portem nebo PCI Express sběrnici a také zajišťuje spojení s jižním mostem
- Některé severní můstky obsahují integrované grafické karty
- Protože různé procesory a paměti vyžadují rozdílnou signalizaci, pracuje severní můstek pouze s jedním nebo se dvěma typy procesorů a zpravidla pouze s jedním typem paměti RAM
- Severní můstek je na základních deskách základním prvkem, který určuje rychlost, druh procesorů, jejich množství a druh paměti RAM, který bude použit
- Ostatní faktory, jako jsou regulace napětí a počet konektorů, také hrají roli

Serverový a počítačový procesor

Více jader v CPU serveru

- Protože musí spouštět více procesů než PC, musí mít větší počet jader
- Důvod více jader a větší mezi paměti souvisí s ekonomikou
- Společnost, co staví servery má dostatek kapitálu na to, aby zaplatila za nákup velkého procesoru s velkým počtem jader a mezi paměti

Více procesorů na serverech

- U serverů je normální mít několik CPU na stejné desce, každý s rozdílnou patičkou
- U PC jsou CPU s chiplety, ale není to stejné, protože i nadále používají jednu zásuvku a není to stejné velikosti jako zásuvky pro servery

Patice a základní deska CPU serveru se liší

- Procesor serveru je mnohem větší než procesor počítače = speciální zásuvka a speciální základní deska
- Základní desky serverů mají věci, které konvenční počítače nemají
 - víc ethernet portů
 - vysílače a přijímače SFP, které umožňují použití optického propojení
 - porty SAS pro připojení disků na serverech
- Pokud jde o RAM, mnoho procesorů serveru používá paměť ECC, kterou nelze použít u konvenčních procesorů a tradiční RAM nelze použít u základních desek, a tedy u serverových procesorů

CPU serveru je vždy spuštěno

- Je zapnuto 24 hodin denně, 7 dní v týdnu a 365 dní v roce
- Pokud jde o provádění testů kvality na CPU pro servery, jsou mnohem náročnější a počet procesorů, které projdou testem, je mnohem nižší
- Proto mají CPU serveru tendenci běžet při nižší průměrné rychlosti hodin na jádro než jejich desktopové protějšky
- Nižší taktovací frekvence je vyrovnávána vyšším počtem jader
 - Paralelní spuštění větší počet podprocesů

Žádné GPU ani integrované koprocесory médií

- CPU serveru se nepoužívají k reprodukci plné grafiky, navíc konstruktéři neumísťují žádný grafický hardware na stejný čip jako samotný procesor, takže nemají integrovaný GPU
- Jedním z důvodů je tepelné udušení, které může vysoce výkonný GPU umístit na CPU sdílením prostoru s ním

Mobilní procesory

- Nejdůležitější parametr je spotřeba energie
- Režim se sníženou frekvencí => snížení energie
- Například notebooky s mobilním procesorem bývají tišší a lehčí
- Velmi dobrý výpočetní výkon i v porovnání s desktopovými počítači

Technika optimalizace provádění instrukcí

Vývoj technologie výroby čipů

- Zmenšování šířky spojů mezi jednotlivými tranzistory
- Zmenšování tranzistorů
- Snižování napájecího napětí
- Zvyšování taktovací frekvence
- Vyzařený výkon = frekvence * napětí
- Zvětšení plochy čipů na waferu

Rozšíření bitové šířky zpracování

- Nejsnazším metoda zvýšení výpočetního výkonu
- Rozšíření počtu bitů, které dokážou mikroprocesory v každé instrukci zpracovávat
- 8-bit mikrořadiče dodnes používány
- Desktopové 32-bit procesory nahrazeny 64-bit

Zvýšení počtu pracovních registrů

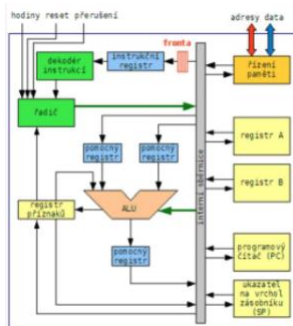
- Přidané tranzistory dávají možnost zvýšení počtu registrů
- Dříve se na registrech šetřilo kvůli vyšší ceně
- Dnes je počet rozsáhlý:
 - Zabírají minimální místo
 - Tvoří nejrychlejší úroveň paměti

Hierarchické uspořádání paměti

- Kvůli ceně je nemožné, aby celá operační paměť byla nejrychlejší:
 - L1 cache
 - L2 cache
 - Operační paměť
 - Virtuální paměť

Fronta instrukcí

- Fronta do které se vkládají operační kódy instrukcí načítané z operační paměti



Pipelining

- Instrukce se zřetězí a současně se tak zpracovává větší množství instrukcí, ale každá instrukce se nachází v jiné části zpracování

- Např. je možné provádět aritmetickou operaci na ALU a současně ukládat výsledek předchozí operace do pracovního registru a současně již načítat operační kód následující instrukce

Explicitní paralelní zpracování instrukcí

- Všechny výkonné jednotky umístěny paralelně vedle sebe – možnost vykonávat operace nezávisle na sobě
- Speciální formát operačních kódů - VLIW

Skalární architektura

- Mikroprocesory, které v jednom taktu načtou maximálně jednu instrukci a provedou na základě jejího operačního kódu pouze jednu instrukci a provedou na základě jejího operačního kódu pouze jednu aritmetickou či logickou operaci s jednoduchou (skalární) hodnotou

Prediktory skoků

- Dopředu odhadnou, zda se skok provede či nikoli a na základě toho začnou do instrukční pipeline vkládat instrukce, které se nachází ihned ze skokem či naopak v cíli skoku

SISD

- Single Instruction, Single Data
- Každá instrukce pracovala maximálně se dvěma operandy

SIMD

- Architektura, ve které se pomocí jedné instrukce může zpracovat větší množství dat

MISD

- Umožňuje na jedny data aplikovat více operací zapsaných ve více instrukcích za sebou – zobecněná pipeline

MIMD

- Největší úroveň paralelismu
- V procesoru nebo procesorovém poli je paralelně zpracováváno větší množství dat a to nezávislé na sobě

Masivně paralelní architektury

- Místo sběrnice je použita nějaká obecnější struktura, například v nejjednodušším případě mřížka, dále pak hyperkostka, „tlustý binární strom“ apod.

Provádění instrukcí mimo pořadí

- Je možné začít provádět nějakou jinou instrukci, u níž je zaručeno, že používá jiné pracovní registry a příznaky
- Interně se řazení instrukcí provádí na základě jejich ukládání do instrukční fronty, ze které jsou instrukce vybírány ve chvíli, kdy jsou známy hodnoty jejich operandů

Snižování spotřeby

Snížením napětí

- Snížením napětí se zmenší maximální frekvence
- Redukuje se tím i spotřeba zdroje a základní desky

Snížením frekvence

- Nižší frekvence umožňuje snížit napětí ještě více

Vypnutí jader procesoru

- Vypnutím jader procesoru se sníží výkon, tím pádem i spotřeba

Odpojení nepotřebných periférií

- Odpojení nedůležitých periférií se sníží celková spotřeba systému

Změna otáček ventilátoru

- Snížením otáček se sníží spotřeba i výkon

Výměna HDD za SSD

- Výměna nám umožní vyšší rychlost a nižší spotřebu

Snížení jasů monitoru

- Menší jas sníží spotřebu několikanásobně

Snížení spotřeby v OS

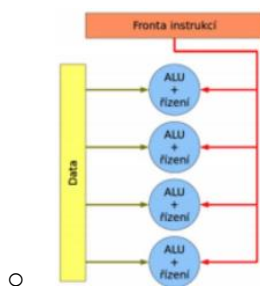
- Režim spánku
- Režim hibernace
- Zapnutí úsporného režimu

Rozšířené instrukční sady

- Skupina nových instrukcí rozšiřující instrukční sadu dané mikro architektury procesoru
- Mohou přidávat nové registry anebo jednotky integrovaného obvodu
- Mají zvýšit reálný výpočetní výkon procesoru bez zvyšování hodinové frekvence či kapacity vyrovnávací paměti

SIMD

- Provádění instrukce s více daty
- Urychlení běhu programů jejichž algoritmy provádí velké množství stejných operací s rozsáhlým objemem dat
- SIMD = Single Instruction Stream, Multiple Data Stream



Vektorové operace

- Zatímco u skalárních procesorů se každá instrukce provádí s jedním či dvěma operandy (příkladem může být instrukce ADD R1, R2)
- Vektorová operace je prováděna s obsahem vektorové proměnné
- Vektor obsahuje několik hodnot stejného typu
- Vektorové instrukce se kromě již zmíněné x86 architektury používají i v ostatních architekturách např. ARM, PowerPC, MIPS ...

MMX

- První rozšiřující instrukční sadou obsahující SIMD operace pro architekturu x86
- Navržena v roce 1996 ve firmě Intel
- 57 nových instrukcí a čtyři datové typy, které byl těmito instrukcemi podporovány
- 4 nové datové typy: 3 z nich vektorové
- Většina nových instrukcí přidáných v rámci sady MMX byla určena pro provádění aritmetických a bitových operací s celočíselným operandy o šířce 8, 16, 32 či 64 bitů

Datové typy přidáné MMX

Datový typ	Bitová šířka operandu	Počet prvků vektoru
packed byte	8 bitů	8
packed word	16 bitů	4
packed doubleword	32 bitů	2
quadword	64 bitů	1

Instrukce přidáné MMX

#	Skupina instrukcí	Příklady instrukcí
1	Základní aritmetické operace	PADD, PADDS, PADDUS, PSUBS, PSUBUS, PMULHW, PMULLW
2	Logické (bitové) operace	PAND, PANDN, POR, PXOR
3	Bitové posuny	PSLL, PSRL, PSRA
4	Porovnávání	PCMPEQ, PCMG
5	Konverze dat	PACKUSWB, PACKSS, PUNPCKH, PUNPCKL
6	Přenosy dat + práce s pamětí	MOV
7	Řízení jednotky MMX	EMMS

Problém MMX s FPU

- FPU = floating-point unit, anebo MCP = math coprocessor
- Aby se ušetřilo místo na čipu, MMX instrukce pracovali s osmicí 64bitových registrů využívaných matematickým koprocесorem (FPU)
- Kvůli tomu bylo komplikované používat MMX a FPU najednou

3DNow!

- Odpověď firmy AMD na MMX
- Kromě stejných instrukcí jako sada MMX také obsahuje nové instrukce pro práci s čísly reprezentovanými 32bitovými hodnotami s plovoucí řadovou čárkou
- Přidám vector obsahující dvojici 32bitových čísel s plovoucí řadovou čárkou
- Nabízí instrukce se zaokrouhlením směrem k nule a instrukce se zaokrouhlením k nejbližší reprezentované hodnotě
- Díky vectoru a instrukcím pracují s 32bitovými čísly s plovoucí řadovou čárkou, se už k tomu nemusela používat matematický koprocessor (FPU)
- FPU bylo stále potřeba pro práci s hodnotami s dvojitou (double) či rozšířenou přesností (extended)

SSE

- Streaming SIMD Extension
- Přidány registry XMM0 až XMM7 a XMM8 až XMM15 u 64bitových platformách mají šířku 128 bitů
- Přidáno 70 instrukcí
- Zaokrouhlení směrem k nule, ke kladnému nekonečnu, k zápornému nekonečnu i k nejbližšímu reprezentovatelnému číslu
- Volba režimu se provádí přes řídicí registr MXCSR

Instrukce přidané SSE

- Instrukční sada typu SIMD
- Některé instrukce měly skalární a vektorovou podobu
- SS = single scalar
- PS = parallel scalar
- Operace ADDSS sčítá pouze dvojici skalárních hodnot typu float/single (pokud se jedná o XMM registry bere z nejnižších 32 bitů v registru)
- Operace ADDPS sečte dvojici vektorů, z nichž každý obsahuje čtyři 32bitové hodnoty typu float/single

AVX (Advanced Vector Extensions)

- Vylepšení SSE
- Namísto 128bitových registrů XMM0-XMM15 přidává 256bitové registry YMM-YMM15
- Přidává SSE instrukce upravené pro 256 bitů

AVX2

- Přidává další instrukce pro celá čísla

AVX-512

- Mění velikost registrů na 512 bitů a zvyšuje jejich počet na 32 a přidává některé instrukce upravené pro 512 bitů

Další instrukční sady

AMD-V/Intel-VT_x

- Rozšíření Hardwarově pomáhající s virtualizací

AES

- Rozšíření, které zrychluje šifrování podle standartu AES

SHA

- Zrychluje algoritmus SHA