

Procesory

Úvod

Procesory jsou poměrně zákeřná věc. Doporučuji si ujasnit co všechno v PC a procesor a co je CPU, protože všechny CPU jsou procesory, ale ne všechny procesory jsou CPU. Tato otázka taky hodně souvisí se maturitní otázkou o základní desce (4.) takže doporučuji se je učit společně.

Definice

Je to čip. Procesor je elektronický obvod složený z tranzistorů, který provádí operace na nějakém externím zdroji dat, obvykle v paměti nebo zpracovává jiný datový tok či prostě počítá matematické operace. Vše dělá díky dostupným instrukcím uložených v jeho architektuře při návrhu.

Architektura

Pomáhá nám navrhnout co od počítače budeme chtít. Je potřeba si definovat základní otázky jako jak chceme reprezentovat data, adresaci, velikost datových struktur a jaké instrukce procesoru dáme.

Dělení podle instrukčních sad:

- RISC (Reduced Instruction Set Computer) – není úplná, procesor je založen na jednoduchých a rychlých pokynech, tam kde je potřeba ušetřit energii a zachovat účinnost
- CISC (Complex Instruction Set Computer) – mají malé „programky“ které jsou založeny na hromadě instrukcí (a trvají dlouho), jsou dobré na víceřadkové operace

Dělení podle zpracovávání instrukcí

- Sekvenční – instrukce se zpracovávají jedna po druhé, neefektivní
- Skalární – zpracovává se několik instrukcí najednou, a to buď promyšleným návrhem (1 takt) nebo zdvojením funkčních celků
- Pipelining – viz Pipelining

Typy

Všechny typy vycházejí z Von Neumannova konceptu. Kdy procesor má řídicí postavení a k němu je připojena řada jednotek se kterými spolupracuje (Odlehčení zátěže procesoru). Jednotlivé architektury se liší podle toho, kam je toto odlehčení převedeno

1. Přímé řízení procesorem
 - a. Periferní zařízení jsou vybavena řadičem, se kterým komunikuje procesor
2. Přímý přístup do paměti (DMA)
 - a. Hodí se pro I/O operace, kde je přímý přesun dat z periferního zařízení do operační paměti
3. Kanálová architektura
 - a. Mezi periferní zařízení a procesor je vložen „kanál“ (Procesor řízený programem), který je schopný problémy řídit samostatně
 - b. Funkční DMA

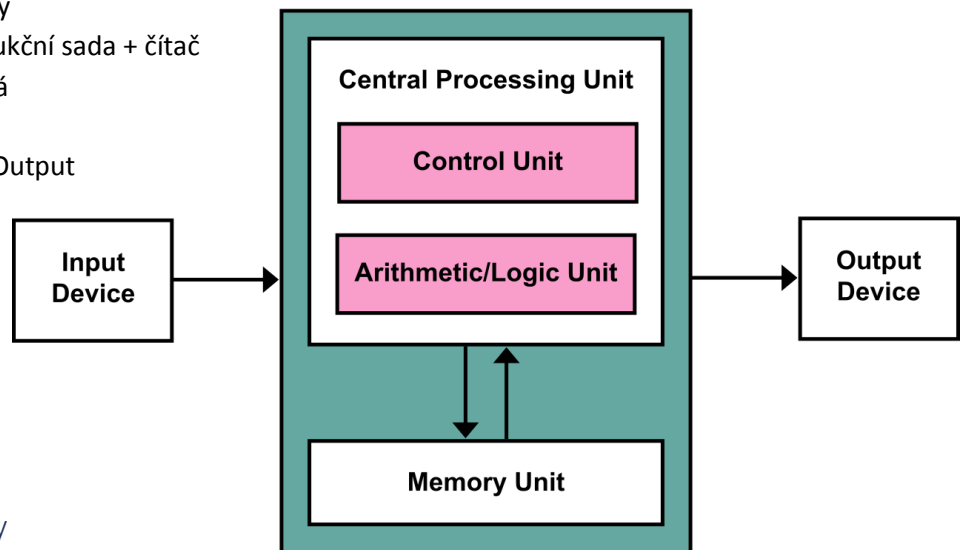
Von Neumannova koncepce

Je jednoduché schéma, které používá sběrnici, na kterou jsou připojeny všechny aktivní prvky. Instrukce a operandy jsou v téže paměti. Struktura počítače je nezávislá na typu úlohy. Paměťové buňky jsou očíslované (adresy).

Tato architektura popisuje počítač s:

- Procesor = ALU + registry
- CU (control unit) = instrukční sada + čítač
- Paměť, která má uložená data a instrukce
- Externí paměť, Input & Output

Je jednodušší a rychlejší než Hardwarová architektura.



Non Von Neuman architektury

Harvardská architektura

Fyzicky odděluje paměť programu a data a instrukce

Neuronové počítače - založené na učení neuronových

Druhy

1. CPU – Central processing unit
2. GPU – Graphics processing unit
3. VPU – Video processing unit
4. NPU – Neural processing unit
5. APU – Accelerated Processing Unit - Procesor do kterého je integrována GPU
6. Zvukový čip
a další...

Pipelining a flag registr

Pipelining

Zřetěžené zpracování instrukcí, které jsou vykonávány v rozdílném clocku. Každá instrukce je rozdělena do několika stupňů.

Například: Máme-li pračku a sušičku a chceme vyprat minimálně dvě várky oblečení, tak nejprve hodíme do pračky prádlo, necháme ho vyprat a následně ho dáme do sušičky. V tento moment si připravíme další várku oblečení a dáme jí do pračky. V dalším kroku vyndáme usušené prádlo první várky a vyprané prádlo druhé várky dáme do sušičky. Nakonec vyndáme druhou várku ze sušičky.

Nejčastější je pětistupňová pipeline:

1. Výběr instrukce (čtení operačního znaku)
2. Dekódování
3. Výběr operandů (vstupní hodnota)
4. Vykonávání (provedení operace)
5. Zápis výsledku

Pozdější procesory měli až 20ti stupňovou pipeline, což není vždy výhodou. Tento problém se však řeší pomocí Hyper-Threading.

Druhy

- Sekvenční - Řadič načítá jednu instrukci po druhé, dekóduje je a postupně je provádí. Každou instrukci lze rozdělit na jednodušší operace. Je málo efektivní.
- Skalární – „Všechny“ operace probíhají současně. Jak je vysvětleno v 5 stupňové pipeline. Nejčastější.
- Superskalární – Paralelní skalární architektura – dvě a více funkčních jednotek provádějí zároveň

Flag registr Stavový registr

Je součástí procesoru. Každý jeho bit má jiný význam a představuje jiný znak. Obsahuje informace o stavech procesoru. Obsah bitů je závislý na výsledcích operace v ALU.

Nejčastější flagy:

- Z (Zero flag) – Výsledek operace byl nula
- C (Carry flag) – Přenos bitu do vyššího řádu
- S / N (Sign flag / Negative flag) – Ukazuje, že výsledek byl záporný. V některých procesorech se tyto 2 flagy liší. Jeden ukazuje, zda poslední výsledek byl negativní, zatímco druhý, zda došlo k odečtení/přidání.
- O / V / W (Overflow) – Došlo k přetečení registru
- P (Parity flag) – Zjišťuje, jestli je číslo liché nebo sudé

Výrobci a Využití

- Motorola, Qualcomm - Smartphones
- Texas Instruments, Casio – Kalkulačky
- AMD, Intel, Via, IBM, Cyrix, Nvidia – Počítače jak už základní desky, grafické karty či přímo CPU
- ASIC - Bitcoin

Parametry

- Sada instrukcí (CISC x RISC)
- Architektura
- Výrobce
- Šířka datové sběrnice [b] - 32-bitový nebo 64-bitový
- Šířka adresové sběrnice [b]
- Vyrovnávací paměti (počet, typy, velikosti [kB])
- Registry (velikosti, počet)
- Frekvence [MHz, GHz]
- Napájení [V]
- Typ patice (Socket – viz Základní deska 4.)
- Počet jader (možnost Hyper Threading)
- Velikost vyrovnávacích pamětí Cache
- TDP [W] - tepelný výkon procesoru
- Úrovně cache (L1,L2,L3) a velikost (768, 4096, 16 384) [kB]
- Otevřený/Uzavřená násobič – otevřený je vhodný pro taktování

CPU

Je hlavní součást počítače, která obsahuje instrukce pomocí, kterých dělá operace jako jednoduché počty, logické operace, kontrolu a výstupní a vstupní operace.

Struktura

- Řadiče – zařizuje součinnost procesoru podle svých instrukcí, ukazuje, kam mají data jít
- Sada registrů – pro rychlejší zpracování dat
 - Pracovní – přesně dané jaké jsou zde operace
 - Univerzální - lze do nich zapisovat, číst z nich a používat je v různých strojových instrukcích
- ALU (Arithmetic-Logic Unit) – může jich být více, provádí se zde základní aritmetické operace
- FPU (Floating point unit) – provádí aritmetické operace s desetinnými čísly
- Cache – ukládá zde kopie dat přečtených z operační paměti, vyrovnávací paměť
 - L1 – pro instrukce a data uložená v registrech
 - L2 - pro data operační paměti RAM
 - L3 – vyrovnává L1 a L2
- GPU – grafické jádro procesoru

Chlazení

- Pasivní – nepotřebuje žádné napájení, většinou měděné či hliníkové žebrování
 - Heat-pipes
 - Měděný válec s póry a kapalinou uvnitř, které sublimuje při zahřátí a kapalně při ochlazení
- Aktivní
 - Ventilátor s žebrováním (nejčastěji)
- Tekutým dusíkem – velmi nákladné a nebezpečné, nutno průběžně doplňovat, určeno k experimentálním účelům, nedá se využívat dlouho

Taktování

Zvyšování frekvence procesoru za účelem získání lepšího výkonu procesoru. Toto efektu dosáhneme dvěma způsoby, a to zvýšením frekvence FrontSideBus/QPI/HyperTransport nebo navýšením násobiče. Protože frekvence FSB ovlivňuje také další komponenty, kterým pracovní frekvenci zvyšovat nechceme, bylo by logickým krokem zvýšit násobič. Drtivá většina procesorů je však dodávána se zamčeným násobičem, aby si uživatelé zakoupily novější procesor a netaktovali, a i přesto že několik modelů s otevřeným násobičem existuje, jsou poměrně drahé. Při taktování pustíme do procesoru více energie při vyšších taktech a bude se i více zahřívat

$$\text{Frekvence procesoru} = \text{frekvence FrontSideBus/QPI/HyperTransport} \times \text{násobič}$$

Zdroje

1. https://cs.wikipedia.org/wiki/Centr%C3%A1ln%C3%AD_procesorov%C3%A1_jednotka
2. [https://en.wikipedia.org/wiki/Processor_\(computing\)](https://en.wikipedia.org/wiki/Processor_(computing))
3. https://cs.wikipedia.org/wiki/Harvardsk%C3%A1_architektura
4. https://cs.wikipedia.org/wiki/%C4%8C%C3%ADta%C4%8D_instruk%C3%AD
5. <https://cs.wikipedia.org/wiki/DMA>
6. https://cs.wikipedia.org/wiki/Architektura_procesoru
7. https://en.wikipedia.org/wiki/Program_counter
8. https://en.wikipedia.org/wiki/Von_Neumann_architecture
9. https://en.wikipedia.org/wiki/Von_Neumann_architecture#/media/File:Von_Neumann_Architecture.svg
10. https://cs.wikipedia.org/wiki/Von_Neumannova_architektura
11. <https://cs.wikipedia.org/wiki/Cache>
12. https://cs.wikipedia.org/wiki/Registr_procesoru
13. <https://www.slideshare.net/KailasKharse/difference-between-cisc-risc-harward-vonneuman>
14. https://en.wikipedia.org/wiki/Instruction_set_architecture
15. https://cs.wikipedia.org/wiki/Instruk%C4%8Dn%C3%AD_sada
16. <https://www.elprocus.com/difference-between-risc-and-cisc-architecture/>
17. [https://en.wikipedia.org/wiki/Pipeline_\(computing\)](https://en.wikipedia.org/wiki/Pipeline_(computing))
18. <https://www.czc.cz/amd-ryzen-7-1700x/208288/produkt>
19. https://cs.wikipedia.org/wiki/Skal%C3%A1rn%C3%AD_procesor
20. https://cs.wikipedia.org/wiki/Superskal%C3%A1rn%C3%AD_architektura