

Architektura procesorů a registry

Procesor (Central Processing Unit - CPU) je hlavní řídicí jednotka počítače, která vykonává instrukce programu uložené v operační paměti. Aby mohl procesor pracovat extrémně rychle, obsahuje vlastní malá, ale blesková úložiště dat nazývaná **registry**.

1. Základní komponenty procesoru

Moderní procesory vycházejí z **Von Neumannovy architektury** a skládají se z několika klíčových částí:

- ALU (Arithmetic Logic Unit):** Provádí veškeré matematické operace (sčítání, násobení) a logické operace (AND, OR, NOT).
- CU (Control Unit):** Řídicí jednotka, která dekoduje instrukce z paměti a řídí tok dat mezi ostatními komponentami.
- Registry:** Nejrychlejší paměťové buňky přímo uvnitř čipu, které drží data, se kterými se právě pracuje.
- Cache:** Rychlá vyrovnávací paměť (L1, L2, L3), která tvoří most mezi pomalou RAM a superrychlým procesorem.

2. Co jsou to Registry?

Registry jsou v podstatě „pracovní stůl“ procesoru. Kapacita registrů určuje tzv. **bitovou šířku procesoru** (dnes standardně 64-bit).

Typy registrů (Architektura x86-64):

Typ registru	Označení	Funkce
Akumulátor	RAX	Používá se pro aritmetické operace a návratové hodnoty funkcí.
Indexové registry	RSI, RDI	Používají se pro operace s řetězcí a poli dat.
Stack Pointer	RSP	Ukazuje na vrchol zásobníku (stack) v paměti.
Instruction Pointer	RIP	Klíčový registr. Obsahuje adresu příští instrukce, která se má vykonat.
Flag Registry	EFLAGS	Drží informaci o stavu systému (např. zda byl výsledek operace nula nebo došlo k přetečení).

3. Instrukční cyklus (Fetch-Decode-Execute)

Procesor pracuje v nekonečné smyčce, kterou řídí systémový takt (frekvence v GHz). Jeden cyklus má tři fáze:

1. **Fetch (Načtení):** CU načte instrukci z paměti RAM na adrese uložené v registru RIP.
2. **Decode (Dekódování):** Řídící jednotka zjistí, co instrukce znamená (např. "sečti dvě čísla").
3. **Execute (Vykonání):** ALU provede výpočet a výsledek uloží zpět do registru nebo paměti.

[Image of the Fetch-Decode-Execute cycle diagram]

4. Architektury: CISC vs. RISC

Ve světě procesorů existují dva hlavní filozofické přístupy k návrhu instrukční sady:

CISC (Complex Instruction Set Computer)

- Obsahuje velké množství komplexních instrukcí (jedna instrukce může vykonat několik operací najednou).
- **Příklad:** Architektura **x86** (Intel, AMD) v PC a serverech.

RISC (Reduced Instruction Set Computer)

- Obsahuje malou sadu jednoduchých, ale velmi rychlých instrukcí. Komplexní úkoly skládá z těchto malých kroků.
- **Příklad:** Architektura **ARM** (Apple Silicon M1/M2, smartphony, [IoT](#)). Výhodou je nižší spotřeba a menší zahřívání.

5. Pipelining a Superskalární architektura

Aby procesory neztrácely čas, používají **Pipelining** (zřetězení). Zatímco se jedna instrukce vykonává, druhá se už dekoduje a třetí se načítá. Moderní procesory jsou navíc **vícejádrové**, což znamená, že mají více těchto jednotek vedle sebe a mohou pracovat na zcela odlišných úlohách současně.

Související články:

- [Druhá generace jazyků \(Assembler\)](#)
- [Datové struktury a paměť](#)
- [Virtualizace a hardwarová podpora](#)

Tagy: hw cpu architecture registers x86 arm risc cisc

From:

<https://www.serviceit.cz/> - **IT ENCYKLOPEDIE**

Permanent link:

<https://www.serviceit.cz/doku.php?id=it:dev:vyvody>

Last update: **2026/01/02 14:03**

