

14. Procesor

- central processing unit
- jednotka, která je schopna provádět strojové instrukce, dané prostřednictvím programu, pomocí nichž pracuje s periferiemi
- první procesory, tranzistorová pole, taktovací frekvence v řádech megahertz, společnost IBM
- procesory s malou integrací, využití křemíkového polotovaru - dle na vytvoření prvních procesorů, jak je známe dnes, implementovali se zejména NOR hradla, oproti tranzistorovým byly menší, účinnější
- procesory s velkou integrací, LSI, CMOS logika, nevýhoda, že byly pomalé, vysoká úroveň integrace, velká instrukční sada
- mikroprocesory - s vynálezem tranzistorů typu MOSFET, což jsou tranzistory, jejichž kolektor se otevírá v závislosti na náboji - elektrické pole, jsou velice účinné a dají se vyrobit ultramale - řády nanometrů, parazitní kapacita tím byla eliminována, to umožnilo přejít do vyšších taktovacích frekvencí, GHz
- *Mooreův zákon* říká, že výpočetní výkon procesorů sleduje exponenciální růst

1. Obsah

- **řadič:** řídící jednotka, zajišťuje součinnost jednotlivých komponent procesoru, načítání operandů instrukcí z operační paměti
- **sada registrů:** uchování mezivýsledků či operandů, bitová šířka určuje velikost registru, pracovní, univerzální či indexregistry
- **ALU:** aritmeticko-logická jednotka provádí operace s daty
- **FPU:** matematický kompresor, operace s desetinnými čísly
- **koprocesor:** počítání s vektory a desetinnými čísly
- **AI koprocesor:** procesorová jednotka optimalizovaná pro výpočet násobných derivací funkce
- těchto komponent se mixuje a vytváří se jednotka zvaná jádro, jádra mohou být na určité věci optimalizovaná, potom má výsledný procesor větší integritu
- thread, čili vlákno, jedná se o podjednotku jádra, které se tváří jako nové jádro, ale není plnohodnotné, třeba sdílí cache jádra, ale používá se na paralelní činnosti
- *multicore:* CPU s více jádry
- *multithreading:* jedná se pouze o podporu, musí se konkrétně implementovat v programu, výhody spočívají v rychlejší odezvě, nižší spotřeba energie, paralelní zpracování instrukcí, mezi nevýhody patří zvýšená složitost kódu, odebrání cache jádra → snížení výkonu jádra
- *hyperthreading* je konkrétní implementace Intelem multithreadingu, ale už je spravovaná samotným procesorem, procesor se jeví s dvojnásobným počtem jader
- *scheduler:* implementováno jak operačním systémem, tak potom samotným procesorem, jedna úloha je rozsekaná na více a paralelně zpracovaná, záležitost multicore procesorů, existují různé strategie (priority, EDF, CFS - linux)

2. Architektury

- jedná se jednak o určitou vnitřní strukturu procesoru, jednak o instrukční sadu, kterou podporuje
- **RISC** - redukovaná instrukční sada, více optimalizované, délka jedné instrukce je právě jeden cyklus, využití řetězení instrukcí (tzv. *pipelining*), registry jsou víceúčelové, procesor komunikuje s pamětí přes sběrnici
- **CISC** - komplexní instrukční sada, pokrývají větší okruh instrukcí, jsou univerzálnější, nízký počet registrů, rychlejší programování, vyšší frekvence, cache
- **mikročipové:**
 - *PIC*
 - *ARM Cortex M*
- **PC:**
 - *x86_64*: moderní 64-bitová architektura
 - *x86_32*: 32-bitová architektura

3. Instrukční sada

- seznam dostupných mechanismů, které daný procesor podporuje
- obsahuje seznam instrukcí, datových typů, dostupných režimů, seznam registrů
- závisí na dané architektuře CPU