# Přerušení

* Schopnost procesoru přerušit právě vykonávaný program a začít vykonávat jiný(obsluhu přerušení)
* Častá věc při přepínání procesů

Typy přerušení:

* Vnější (asynchronní)
  + nečekané/neplánované
  + zmáčknutí klávesy
* Vnitřní (asynchronní)
  + Vykonává je sám procesor
  + Problém s vykonávanou instrukcí
  + Dělení 0
  + Nečekané
* SW (synchronní)
  + Speciální strojová instrukce
  + Očekávané
  + Vyplívá z podstaty programu
* Maskované
  + Slouží ovladačům
* Nemaskované
  + Výpadek zdrojů
  + Musí se provést
* Signál přerušení přijímá Interrupt Controler (řadič přerušení)
* vysílá ho například periferie C8 klávesy, IC přijme požadavek na
* přerušení, vyhodnotí prioritu s probíhajícím procesem, řekne procesoru, že to bude
* Procesor dokončí instrukci a uloží adresu následující operace do Stacku (uloží se tam i registry a tak dále)
* procesor jsi z řadiče vezme číslo (každé přerušení má určené číslo, které popisuje
* určitou službu) přerušení.
* procesor hledá v tabulce vektory přerušení (co je úsek operační paměti)
* vektory přerušení – virtuální adres první instrukce obsluhy přerušení. Ke každému
* číslu přerušení náleží 1 vektor přerušení. Musí jsi vypůjčit adresu.

# DMA (Direct memory acess)

* souvisí s přerušením
* přímí přístup do paměti blokového zařízení bez účasti procesoru (to jde pak rychleji, umožňuje zrychlení)

**zařízení se dělí na blokové a bytové**

**blokové – všechny paměti, harddisk, přenosy po síti**

**bytové – například klávesnice**

* **Proceso**r naprogramuje DMA odkud, kolik a kam se bude přenášet + číslo přerušení nejpomalejší periferie, která se bude přenášet
* DMA je oznámeno, že může začít přenos
* DMA vyšle příkaz HOLD procesoru. Procesor ukončí činnost na sběrnici, odpoví příkazem HA (Hold accepted), oznámí opuštění sběrnice to odpojí procesor od sběrnice
* DMA dělá přenos hardwarově – pomocí čítače nastaví (procesor by musel mít instrukce) adresy na sběrnici, snižuje se stále dolů až na 0(pak je to hotovo)
* používají se pulzy (nikoli instrukce)

# Rozdělení procesorů

**Na Harvardské struktuře:**

* sběrnice pro data
* sběrnice pro instrukce
* paměť pro instrukce
* paměť pro data
* paměti mohou pracovat paralelně

**Na Von Neumanově struktuře:**

* 1 paměť pro data a instrukce dohromady
* buď se zapisují data nebo instrukce

**RISC (Reduced instruction set computer):**

* má jednotný HW
* má složitější program assembleru
* v každém taktu je ukončena 1 instrukce
* je nutný pipelining
* obsahuje velký počet registrů
* obsah registů není nutné přehrávat jinam, registry se pouze přejmenují(přeadresují)
* jsou zde banky registů, když se změní PREFIX tak se změní registr
  + PREFIX – část adresy
    - nahrává se do Stacku
    - nemají pevný akumulátor (není pevně připojen na ALU)
    - jako akumulátor může sloužit libovolný registr
* malý počet instrukcí, delší program assembleru
* pouze 2 instrukce pro práci s pamětí (LOAD, STORE)
* optimalizovaný kompilátor umí pracovat s blokovou algebrou
* dokáže přehazovat instrukce

**CISC (Complex instruction set computer):**

* mají akumulátor pevně vázaný na ALU
* instrukce se nejdřív přeloží na mikroinstrukce a potom do strojového kódu dvojí překlad

**Parametry procesoru:**

* rychlost(frekvence) - počet operací za sekundu, udává se v Hz
* efektivita mikrokódu – počet kroků na provedení jedné instrukce
* numerický Koprocesor FPU – buďto není/je přítomen, operace v plovoucí desetinné čárce
* počet instrukčních kmitů – maximální počet instrukcí v jednom taktu(pipelingu)
* šířka slova – maximální počet bitů, které je možné zpracovat během jedné operace (8, 16, 32, 64 bitový)