

## 8. Přerušeni

K čemu je přerušeni dobré (vynucená bezodkladná obsluha asynchronní události). [[wikipedia](#)]

Obsluha přerušeni. [[11.ppt](#)]

Možno ukázat program a okomentovat strojový kód. [[zivak.zip](#)], [[lis.zip](#)]

---

**Přerušeni** ([anglicky](#) interrupt) je v [informatice](#) nástroj pro asynchronní obsluhu událostí, kdy [procesor](#) přeruší vykonávání instrukcí a vykoná obsluhu přerušeni. Původně přerušeni sloužilo k obsluze [hardwarových](#) zařízení, později se přidala synchronní softwarová přerušeni a vnitřní přerušeni pro oznamování chyb procesorem vzniklých při provádění strojových instrukcí.

### 1. Obsah

- [1 Hardwarové \(vnější\) přerušeni](#)
- [2 Softwarové přerušeni](#)
- [3 Obsluha přerušeni](#)
  - o [3.1 ISR – Interrupt Service Routine](#)
  - o [3.2 Průběh hardwarového přerušeni](#)
  - o [3.3 Průběh softwarového přerušeni](#)
  - o [3.4 Softwarové přerušeni v architektuře x86](#)
- [4 Standardní využití linek u PC AT](#)

### 2. Hardwarové (vnější) přerušeni

Původně se tak označovalo [IRQ](#), tedy hardwarové přerušeni, které nastává jako reakce na signál od zařízení, které jím upozorňuje procesor (obvykle [operační systém](#), přesněji svůj [ovladač](#) v tomto systému), že potřebuje obsloužit. Procesor při příchodu přerušeni přestane provádět současný výpočet, uloží část svého stavu a začne vykonávat [obsluhu přerušeni](#). Takovýto typ přerušeni (vyvolaný HW) je tzv. asynchronní, tedy může přijít v libovolném bodu výpočtu (nikoliv ovšem doslova libovolném: například nemůže přijít uprostřed instrukce, a to ani u moderních [superskalárních](#) procesorů, ve kterých se instrukce rozkládá na více interních instrukcí). Přerušeni přichází bez ohledu na to, kolik je momentálně vykonávaných [programů](#). Přerušeni lze také v některých případech zakázat [maskováním](#) – k obsluze přerušeni dojde až po opětovném povolení přerušeni.

### 3. Softwarové přerušeni

Obecnějším termínem se to stalo při zavedení softwarových přerušeni, tedy přerušeni vyvolávaných přímo uvnitř [procesoru](#) a využívající stejnou nebo podobnou metodu předání řízení z probíhajícího výpočtu do obsluhy přerušeni. Softwarové přerušeni může být vyvoláno buď speciální [instrukcí](#) (přerušeni se používá jako způsob volání [operačního systému](#) z [aplikace](#)), nebo výjimečným stavem výpočtu, neboli výjimkou (dělení nulou, přístup k neexistující adrese). Softwarová přerušeni jsou vždy tzv. synchronní. Pokud přijde více přerušeni, musí systém vyhodnotit, v jakém pořadí se budou provádět.

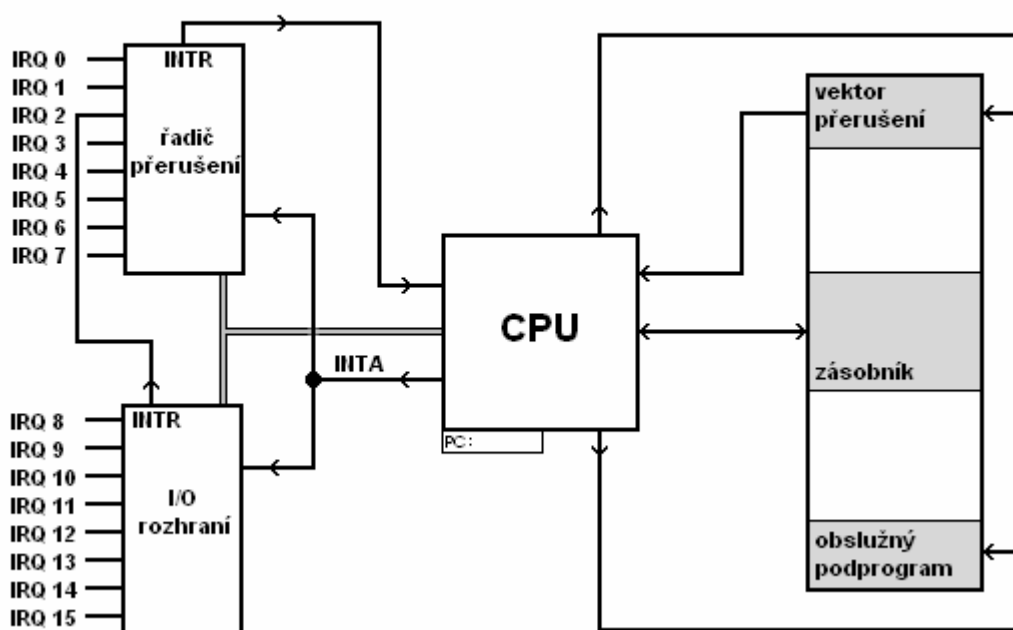
## 4. Obsluha přerušení

### ISR – Interrupt Service Routine

ISR je obslužný [podprogram](#) pro vykonání obsluhy daného přerušení a je obvykle součástí [ovladače](#) zařízení, které přerušení vyvolalo. ISR musí nejprve uložit stavové informace [CPU](#) na [zásobník](#) (FR a PC). Poté provede obsluhu přerušení a nakonec obnoví ze zásobníku stav [CPU](#) a přerušená činnost pokračuje v místě kde byla při příchodu přerušení vykonávaná [instrukce](#) pozastavena. Přerušený [program](#) v podstatě nic nepozná – až na malé zpoždění.

### Průběh hardwarového přerušení

1. Vnější zařízení vyvolá požadavek o přerušení
2. I/O rozhraní vyšle signál IRQ na řadič přerušení (na port IRQ 2)
3. Řadič přerušení vygeneruje signál INTR – „někdo“ žádá o přerušení a vyšle ho k procesoru.
4. Procesor se na základě [maskování](#) rozhodne obsloužit přerušení a signálem INTA se zeptá, jaké zařízení žádá o přerušení.
5. Řadič přerušení identifikuje zařízení, které žádá o přerušení a odešle číslo typu přerušení k [Procesoru](#)
6. Procesor uloží stavové informace o právě zpracovávaném programu do [zásobníku](#).
7. Podle čísla typu příchozího přerušení nalezne ve vektoru přerušení adresu příslušného obslužného [podprogramu](#).
8. Vyhledá obslužný [podprogram](#) obsluhy přerušení v [paměti](#) a vykoná ho.
9. Po provedení obslužného programu opět obnoví uložené stavové informace ze [zásobníku](#) a přerušený [program](#) pokračuje dál.



### Průběh softwarového přerušení

1. Při příchodu přerušení se uloží stavové informace o právě zpracovávaném [programu](#) do [zásobníku](#).
2. Zakáže se další přerušení.
3. Procesor zjistí [vektor přerušení](#) (podle operandu)
4. Nalezne obslužný [podprogram](#) a vykoná ho.

5. Po návratu z podprogramu obnoví uložené stavové informace o přerušeném [programu](#).

## Softwarové přerušení v architektuře x86

Architektura [x86](#) má dvě instrukce pro softwarové přerušení: `int` a `int03`, speciální krátkou formu určenou jako breakpoint pro [debugger](#).

Má šest výjimek:

1. dělení nulou
2. krokovací přerušení - na [i386](#) rozšířeno na víceúčelové debugovací přerušení
3. [NMI](#)
4. breakpoint - vyvolaný již zmíněnou instrukcí `int03`
5. přetečení - je vyvolané instrukcí `into` pokud je ve vlajkách zaznamenáno přetečení
6. překročení mezí - volané instrukcí `bound` pokud byly překročeny meze pole, přidané v [80186](#)
7. chybná instrukce

U procesoru [i386](#) přibylo 9 dalších:

1. nedostupnost koprocessoru - nepřítomnost nebo nepřepnutá úloha - dřív bylo hlášeno externím hardwarovým přerušením
2. dvojitý výpadek - volané pokud dojde k výjimce při vyvolávání výjimky
3. překročení limitu [segmentu](#) koprocessorem
4. chybný TSS - chyba při pokusu o změnu úlohy
5. nepřístupný segment
6. překročení limitu zásobníku
7. obecná chyba ochrany - většina chyb související se [segmentací](#) kromě těch obslužených ostatními výjimkami
8. výpadek stránky - obecně libovolná chyba při [stránkování](#)
9. (nepoužito)
10. chyba koprocessoru

## 5. Standardní využití linek u PC AT

- IRQ 0 Řadič systémového času
- IRQ 1 Řadič klávesnice
- IRQ 2 Připojení řadičů přerušení z I/O rozhnání (kaskáda)- zde jsou sjednoceny IRQ 8 až IRQ 15
- IRQ 3 Sériový port 2
- IRQ 4 Sériový port 1
- IRQ 5 Paralelní port 2
- IRQ 6 Řadič disket
- IRQ 8 Hodiny reálného času
- IRQ 9 Volný, podle připojeného zařízení
- IRQ 10 Volný, podle připojeného zařízení
- IRQ 11 Volný, podle připojeného zařízení
- IRQ 12 Volný, podle připojeného zařízení
- IRQ 13 Numerický koprocessor
- IRQ 14 Řadič pevného disku
- IRQ 15 Volný, podle připojeného zařízení

## Obrázky z Prezentace

