

Správa vstupných a výstupných zariadení

Portál: edu.ukf.sk - Vzdelávací portál - Univerzita
Konštantína Filozofa, Nitra

Kurz: Operačné systémy (KI/OS/15)

Kniha: Správa vstupných a výstupných zariadení

Vytlačil(a): Zuzana Pavlendová

Dátum: Streda, 1 december 2021, 17:59

Opis

4 a) SPRÁVA VSTUPNÝCH A VÝSTUPNÝCH ZARIADENÍ

4.1 Komunikácia V/V zariadení

4.1.1 Programová obsluha

4.1.2 Prerušenie (interrupt)

4.1.3 Vstup a výstup dát s DMA

4.1.4 Vstupné a výstupné radiče

4.1.5 Privilegované inštrukcie

4.1.6 Správa zariadenia

4.1.7 Časovač

Obsah

Úvod

Požiadavky

4.1 Komunikácia V/V zariadení

4.1.1 Programová obsluha

4.1.2 Prerušenie (interrupt)

4.1.3 Vstup a výstup dát s DMA

4.1.4 Vstupné a výstupné radiče

4.1.5 Privilegované inštrukcie

4.1.6 Správa zariadenia

4.1.7 Časovač

Úvod

Z hľadiska množstva prenášaných dát rozdeľujeme vstupné a výstupné (ďalej označujeme V/V alebo I/O) zariadenia na:

- znakové, kam patria klávesnice, znakové displeje a terminály, tlačiarne, myši, plottery, tablety apod.,
- blokové, kde sa radia disky, CD ROM, magnetické pásky apod.

Niektoré zariadenia do tohto delenia nezapadajú. Sú to zariadenia tzv. pamäťovo mapované, ako sú grafické displeje, špeciálne časovače apod. Rozhranie vstupných a výstupných zariadení poskytované operačným systémom, by malo byť jednotné pre všetky zariadenia do takej miery, ako je to len možné.

Požiadavky

V tejto kapitole sa dozviete:

- Akými spôsobmi komunikujú vonkajšie zariadenia s procesorom.
- Ako pracujú diskové pamäte a ako sa vyhľadáva informácia na disku.

Po jej preštudovaní by ste mali byť schopní:

- Charakterizovať spôsoby komunikácie V/V zariadení s procesorom.
- Po znatí vytváranie ovládačov periférnych zariadení.
- Porozumieť rozdielom v komunikácii prostredníctvom prerušenia a DMA.
- Popísať spôsoby vyhľadávania informácií na disku.
- Po znatí spôsob výpočtu rôznych parametrov diskov.

Kľúčové slová tejto kapitoly:

Prerušenie, DMA, kanál, ovládač, prístup na disk.

Doba potrebná k štúdiu: 4 hodiny

Sprievodca štúdiom

Štúdium tejto kapitoly je dosť náročné na pochopenie predovšetkým princípov komunikácie vstupných a výstupných zariadení s procesorom. V prípade, že máte preštudované základy architektúry počítačov, potom štúdium tejto kapitoly je doplnením základných znalostí princípov práce počítača. Na štúdium tejto časti si vyhradte aspoň 4 hodiny. Po celkovom preštudovaní

a vyriešení všetkých príkladov odporúčame vypracovať korešpondenčnú úlohu.

4.1 Komunikácia V/V zariadení

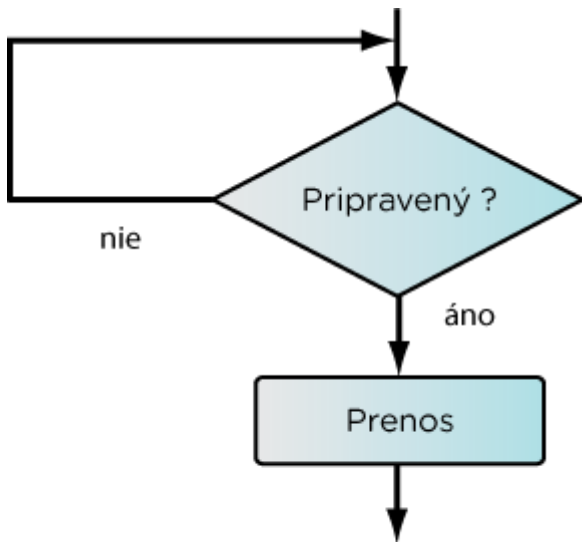
Vstupné a výstupné zariadenia, bez ohľadu na spôsob pripojenia, používajú štyri základné techniky riadenia prenosu:

- programové riadenie vstupu a výstupu,
- riadenie na základe prerušenia,
- priamy prístup k operačnej pamäti (DMA – Direct Memory Access),
- vstup a výstup pomocou špecializovaného procesora.

Teraz si uvedieme podrobnejšiu analýzu jednotlivých druhov komunikácií.

4.1.1 Programová obsluha

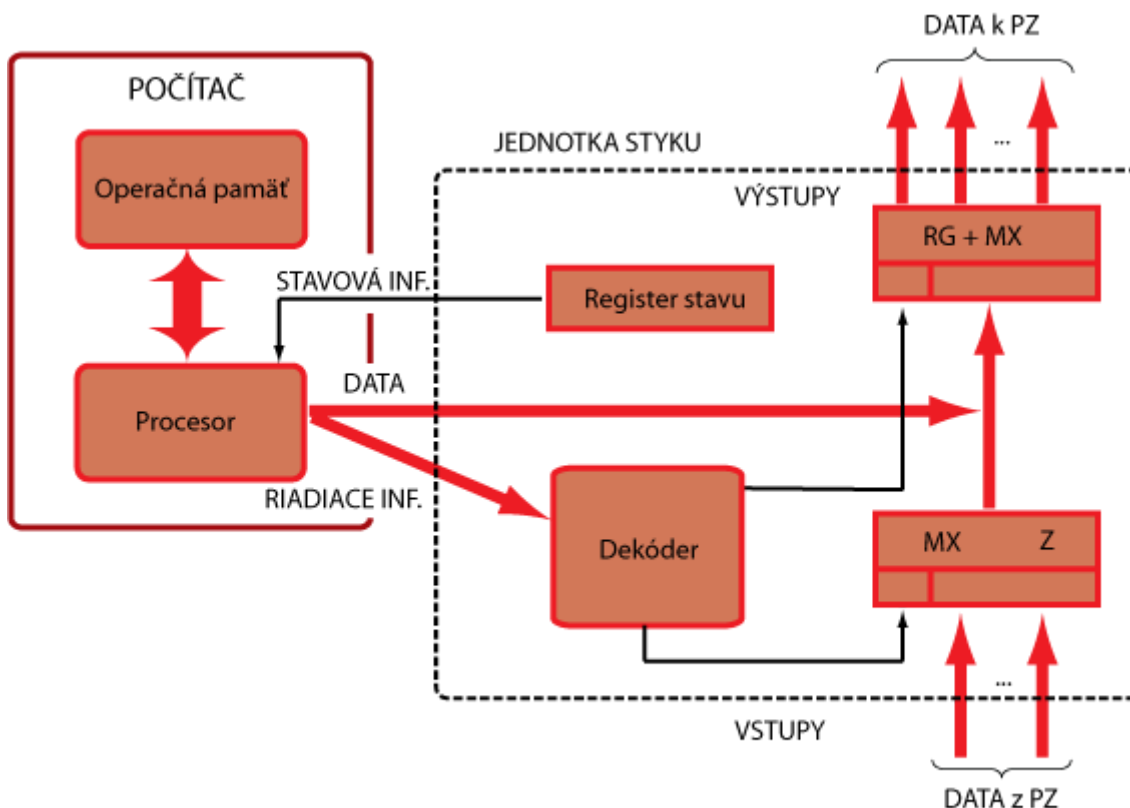
Pri tomto spôsobe riadenia V/V operácií určuje vždy procesor na základe programu okamihy prenosu údajov do alebo z periférneho zariadenia. Synchronizácia prenosu sa potom deje jednoduchým testovaním pripravenosti zariadenia na prenos dát, ako je uvedené na nasledujúcich obrázkoch.



4.2_Operacia_s_cinnym_cakanim

Princíp podmieneného prenosu

Táto technika je najmenej náročná na technickom vybavení. Hlavnou nevýhodou je to, že procesor je značne zaťažovaný neproduktívnou činnosťou. Technické vybavenie jednotky, pre pripojenie periférneho zariadenia zahŕňa register stavu, obsahujúci informáciu o pripravenosti na vstup/výstup dát a multiplexor, smerujúci dáta k adresovanému periférnemu zariadeniu. Bloková schéma technických prostriedkov programového V/V je uvedená na nasledujúcom obrázku.



Programovo riadené periférie

Vstupné a výstupné radiče sú realizované ako:

- pamäťovo mapované: registre sú adresované ako pamäť, prístupné pomocou bežných operácií čítania a zápisu do pamäti,
- izolované: registre sú prístupné pomocou špeciálnych inštrukcií (spravidla nazývaných IN a OUT); vďaka tomu sú adresné priestory pamäte a vstupov/výstupov oddelené.

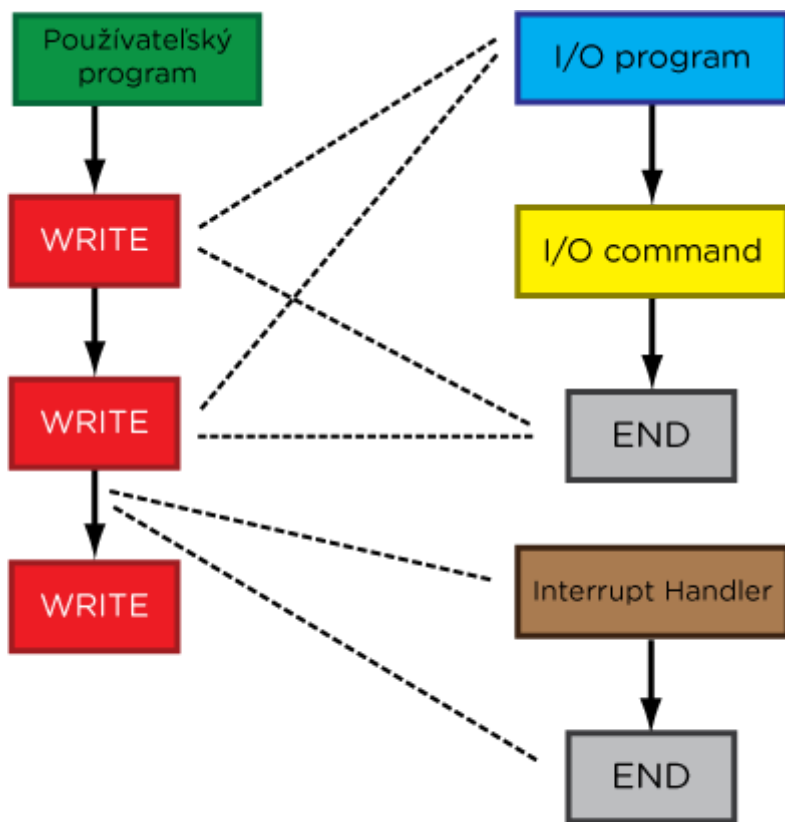
4.1.2 Prerušenie (interrupt)

Cieľom zavedenia prerušenia pri riadení vstupných/výstupných prenosov dát, je zlepšenie účinnosti týchto operácií. Prerušujúca udalosť spôsobí, že sa potlačenie vykonávania bežiaceho procesu v CPU takým spôsobom, aby ho bolo možné neskôr obnoviť. V dobe riešenia I/O operácie sa umožní, aby CPU vykonávala iné inštrukcie než periférne. Pôvodne sa tento mechanizmus používal len pre vyžiadanie pozornosti procesoru. Pri vyvolaní prerušený procesor začne vykonávať podprogram obsluhy prerušený (interrupt service routine - ISR) podobným spôsobom, ako by bol vyvolaný normálny podprogram. Podprogram musí uchovať stav procesoru, potom vykoná vlastnú obsluhu prerušenia (napríklad pošle znak alebo blok znakov na výstupné zariadenie) a nakoniec obnoví stav procesoru, aby prerušený program nič nepoznal (až na oneskorenie). Podobá sa vyvolaniu podprogramu, ale vykonáva sa špeciálnou inštrukciou. Vývojový diagram obsluhy prerušenia je uvedený na nasledujúcom obrázku.

Postupnosť obsluhy prerušenia je vyjadrená nasledujúcimi úkonmi:

- uchovanie stavu procesoru,
- vlastná obsluha prerušenia,
- obnovenie stavu procesoru.

Základný rozdiel medzi programovo riadenou komunikáciou a komunikáciou prostredníctvom prerušovacieho systému, je v synchronizácii prenosu údajov. Pri programovom riadení sa počas vykonávaní operácie čítanie/zápis údajov musí čakať na potvrdenie pripravenosti periférneho zariadenia a potom sa vykoná prenos, pričom procesor v dobe čakania nevykonáva žiadne následné inštrukcie. Pri obsluhu zariadení pomocou prerušenia, procesor pokračuje vo vykonávaní operácií a je oboznámený o pripravenom periférnom zariadení pomocou prerušenia. Tento princíp komunikácie je zobrazený na nasledujúcom obrázku.

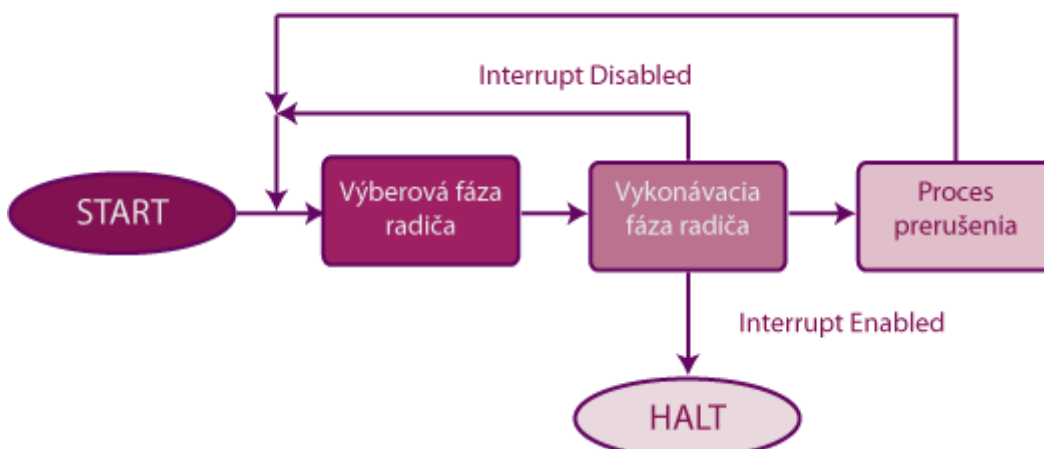


Operácia s prerušením

Je pravdepodobné, že princípy prerušenia sa tak nemusia používať len pre operácie vstupu/výstupu údajov, ale i pre iné typy synchronizácie procesov. Preto prerušenie delíme na:

- **Vonkajšie** - zdrojom sú radiče (hlavne I/O zariadenie) umiestnené "vo vnútri procesora". K prerušeniu dochádza bez ohľadu na práve prevádzané miesto v programe a ISR je vyvolaný po dokončení inštrukcie. Reakciu na prerušenie sa dá dočasne zakázať (maskovať). Potom dôjde k obsluhu po povolení prerušenia. Po návrate z ISR prerušený program pokračuje ďalšou inštrukciou.
- **Vnútorne** - prerušenie je vyvolané chybou pri vykonávaní strojovej inštrukcie (delenie nulou, pretečenie, porušenie ochrany pamäte, výpadok stránky). ISR môže vypísať chybové hlásenie a ukončiť program, dosadiť náhradný výsledok v prípade aritmetickej chyby, zaviesť stránku do vnútornej pamäte z disku apod. Pri niektorých chybách je možné zopakovať inštrukciu, ktorá chybu spôsobila.

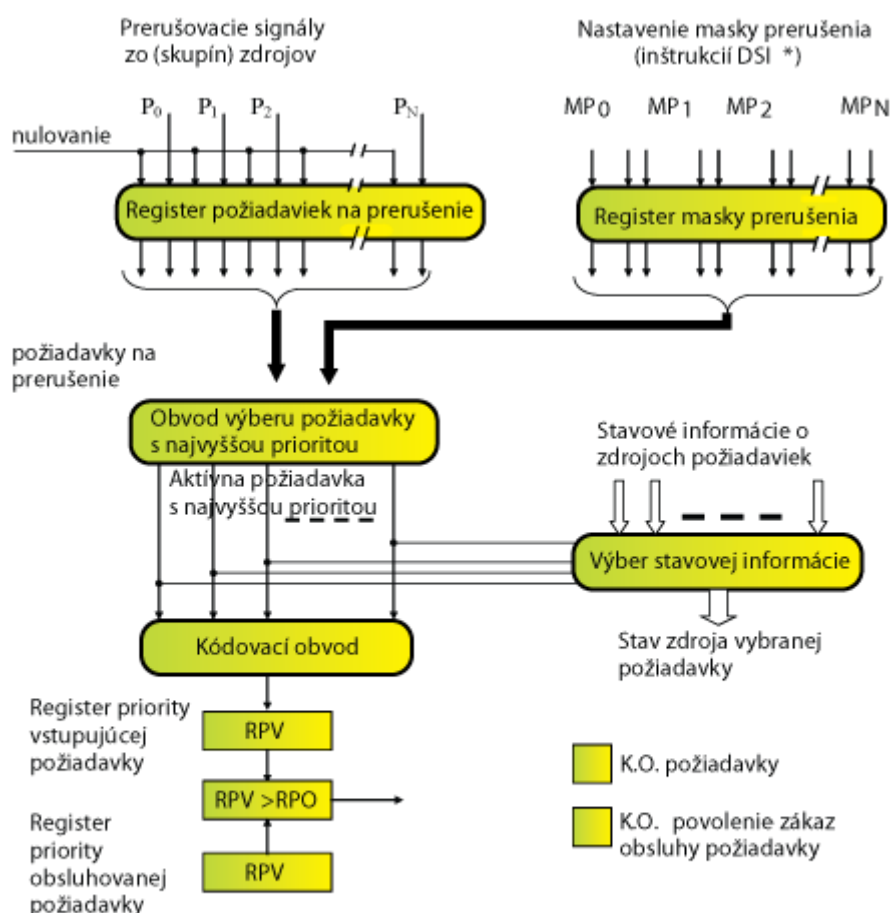
Programové - prerušenie je vyvolané inštrukciou volania prerušenia, umiestnenou priamo v programe. Používa sa pre volanie služieb operačného systému. Výhoda oproti volaniu podprogramov: nie je možné vyvolávať podprogramy na ľubovoľných adresách, iba adresy uvedené v tabuľke prerušenia. Vzhľadom k tomu, že synchronizácia činnosti procesora je daná návaznosťou výberovej a vykonávacej fázy cyklu radiča, je nutné asynchrónne požiadavky práve prerušené, začleniť do tohto synchronného procesu a to podľa nasledujúceho obrázka. Zákaz alebo povolenie prerušenia vyvoláva procesor.



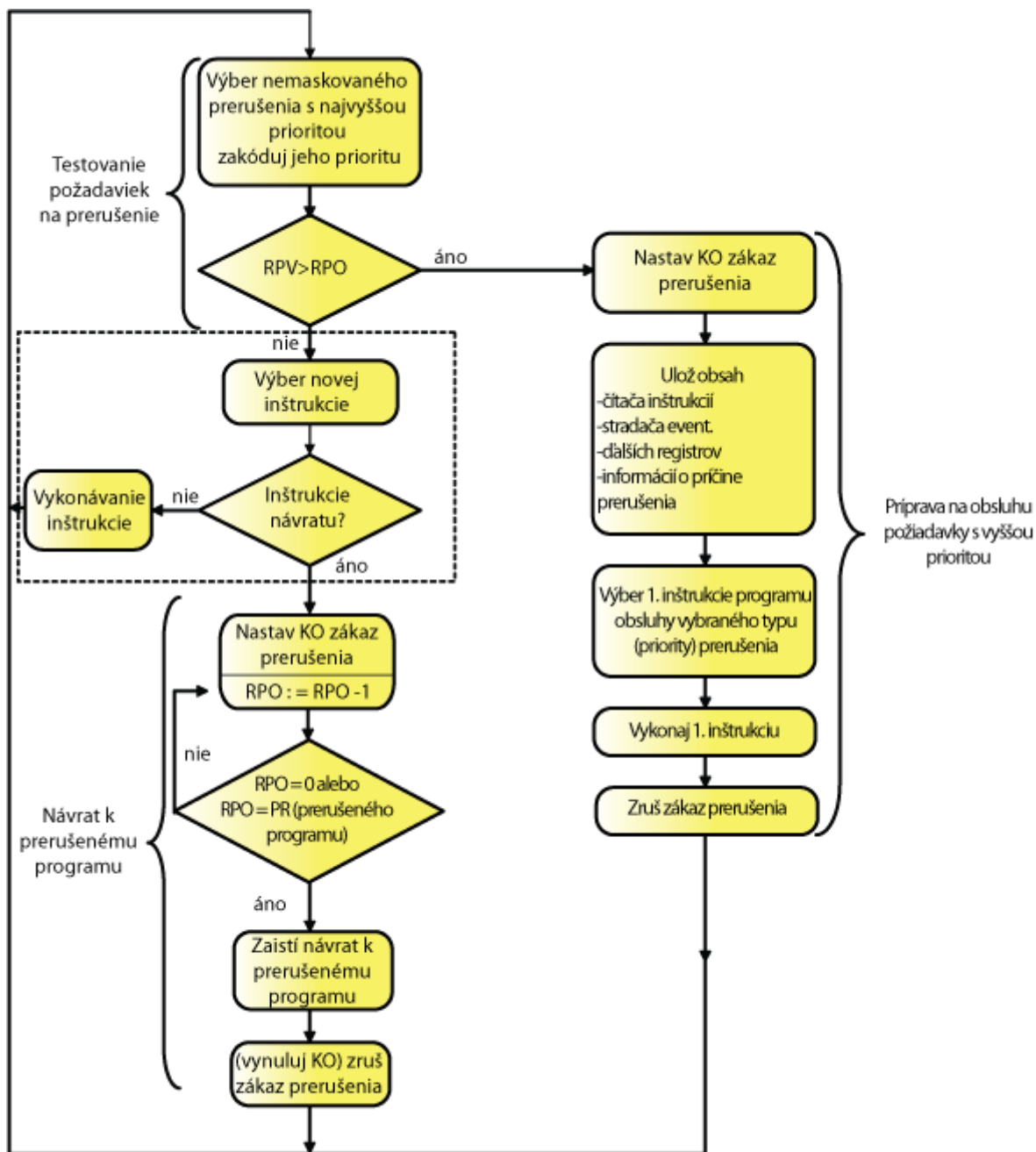
Z hľadiska riadenia činnosti je prerušenie (interrupt) udalosť, ktorá zmení sekvenciu, v ktorej CPU vykonáva inštrukcie. Ako sme už uviedli, je generované HW alebo SW. Keď sa vyskytne prerušenie, tak:

- operačný systém sa ujme riadenia (tj. HW odovzdá riadenie do operačného systému),
- operačný systém uloží stav prerušeného procesu,
- operačný systém analyzuje prerušenie a odovzdá riadenie do obslužnej rutiny prerušenia (dnes väčšinou zaistuje už HW),
- obslužná rutina prerušenia potom obsluži prerušenie,
- je obnovený stav prerušeného procesu,
- prerušený proces pokračuje.

Prerušenie môže byť generované **synchronne** bežiacim procesom (trap inštrukcie) alebo **asynchronne**, ktoré vznikne nejakou vonkajšou udalosťou. Ďalším možným typom sú výnimky (**exceptions**) vzniknuté nesprávnym chovaním programu (delenie 0, neznáma inštrukcia, ...). Hlavnou výhodou konceptu prerušenia je odstránenie potreby **ošahávania (polling)** zariadenia. Zdroj vydá požiadavky prerušenia, radič prerušenia určí konkrétnu príčinu prerušenia a vykoná potrebné akcie. Pri riadení prerušenia musí byť zachovaná možnosť vrátiť riadenie prerušenému programu do miesta, kde bol tento program prerušený prerušený. Na miesto prerušiteľnosti sa nekladú obmedzenia, musí sa teda uchovať stav behu procesu (obsah čítača inštrukcií, registra inštrukcií, registrov operačnej jednotky). Na nasledujúcich obrázkoch sú uvedené nevyhnutné technické prostriedky prerušovacieho systému a detailný algoritmus obsluhy prerušenia, realizovaný týmito technickými prostriedkami.



Technické prostriedky obsluhy prerušenia



Algoritmus obsluhy prerušenia

4.1.3 Vstup a výstup dát s DMA

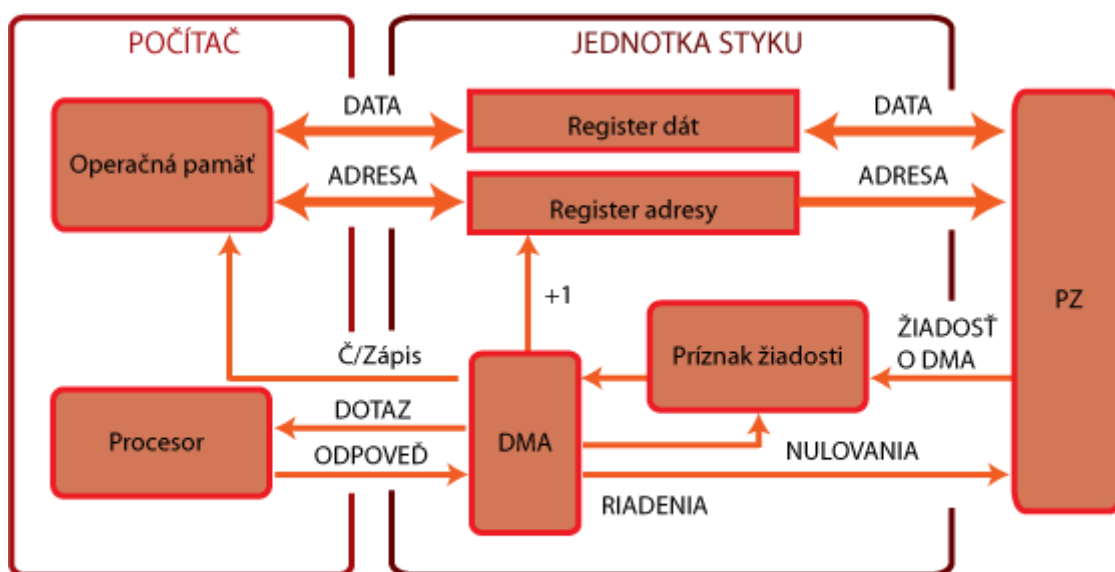
Pre odľahčenie procesora bývajú súčasťou počítača obvody schopné realizovať väčšie množstvo I/O operácií (jedná sa o odchýlku od Von Neumannovej schémy počítača):

- DMA kanály - pre kopírovanie blokov dát medzi pamäťou a I/O zariadením. Je potrebné ich naprogramovať zápisom do hardvérových registrov.
- Špecializované I/O procesory (niekedy nazývané kanály) - sú riadené postupnosťou vlastných inštrukcií (tzv. kanálovým programom):
 - selektorové - obsluhuje 1 rýchle zariadenie (mg. disk, páska),
 - multiplexné - môžu obsluhovať niekoľko pomalých zariadení (tlačiarne, niektoré terminály, apod.).

DMA kanály slúžia pre kopírovanie blokov dát medzi pamäťou a I/O zariadením. Je potrebné ich naprogramovať zápisom do hardvérových registrov tak, ako je uvedené v nasledujúcom vývojovom diagrame.

Neprogramovo riadený prenos využíva zvláštne obvody tzv. kanál priameho prístupu k pamäti (DMA). Pri jednoduchom riešení kanálu DMA býva počas prenosu týmto kanálom CPU vypojený z činnosti. Pri inom zložitejšom riešení sa prenos uskutočňuje tzv. kradnutím cyklov prebiehajúceho programu, tj. CPU pokračuje spomalenou rýchlosťou v bežnej činnosti.

Princíp prenosu riadeného technickými prostriedkami pri priamom prístupe k pamäti je uvedený na nasledujúcom obrázku. Keď periférne zariadenie (PZ) vyšle žiadosť o priamy prístup do pamäte, procesor prejde do stavu čakania (WAIT) alebo odpojenia od zbernice (HOLD) a informuje o tom obvody DMA. Keď skončí prenos dát medzi PZ a pamäťou, príznak žiadosti o DMA sa vynuluje, zruší sa dotazovací signál z obvodov DMA a procesor sa pripojí na zbernicu a tým k pamäti.



DMA riadená periféria

4.1.4 Vstupné a výstupné radiče

Vstupné a výstupné radiče slúžia k pripojovaniu I/O zariadení. Z hľadiska programátora radič vyzerá ako sada hardvérových registrov, pričom registre môžu byť:

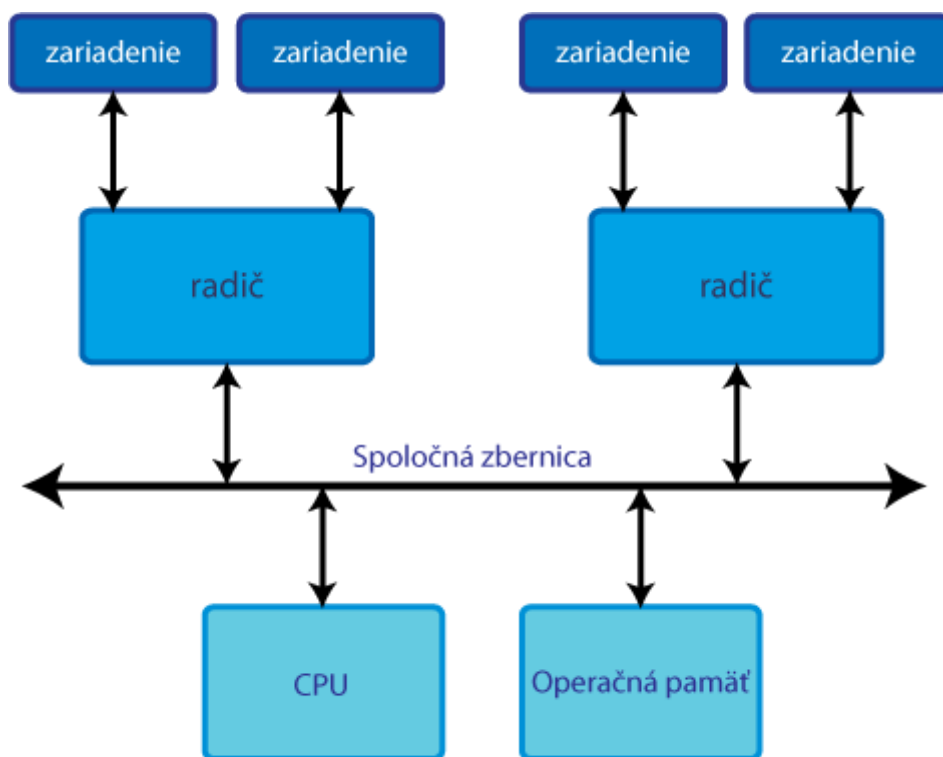
- len na čítanie,
- len na zápis,
- na čítanie i zápis.

Pri inicializácii počítača je potrebné zistiť, ktoré radiče sú v počítači zapojené a inicializovať ich. Pri vstupe alebo výstupe dát je spravidla nutné čítaním z radiča zistiť, či je zariadenie pripravené, zapísať do radiča príkaz a zapísať alebo prečítať dáta. Pokiaľ nie je možné ihneď pokračovať, zariadenie spravidla signalizuje svoju pripravenosť vyvolaním prerušenia. Podľa architektúry počítača sa vstupy/výstupy delia na:

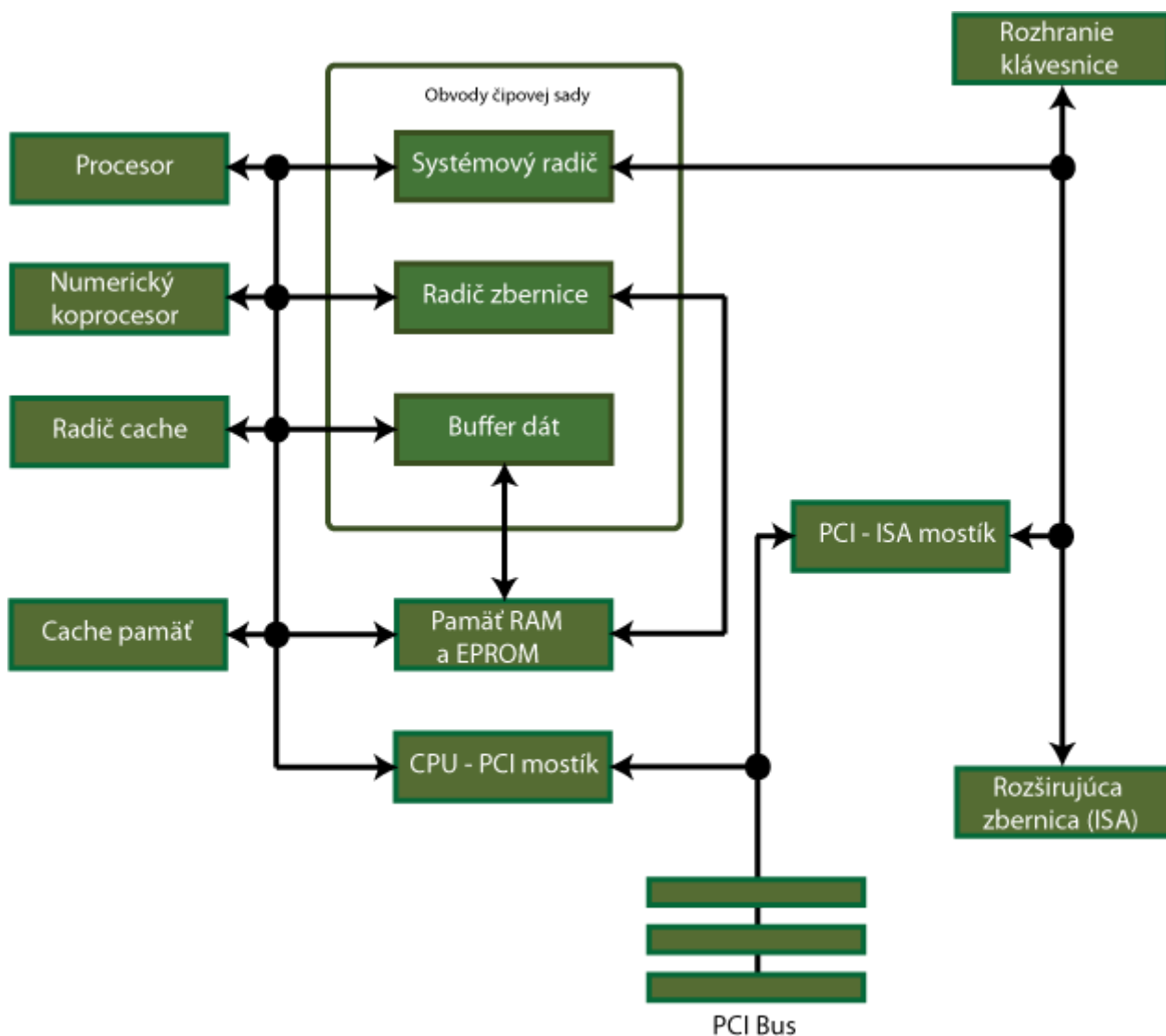
- pamäťovo mapované: registre sú adresované ako pamäť, prístupné pomocou bežných operácií čítanie a zápisu do pamäti,
- izolované: registre sú prístupné pomocou špeciálnych inštrukcií (spravidla nazývaných IN a OUT); vďaka tomu sú adresné priestory pamäti a vstupov/výstupov oddelené.

S rozvojom počítačov postupovala na mnohých úrovniach i štandardizácia, kde práve štandardizácia pripojovania periférnych zariadení je záväzná ako pre výrobcu periférnych zariadení tak pre výrobcu počítačov. Jeden spôsob štandardizácie je vo vzniku prepojenia časťou počítačov prostredníctvom zberníc, čo viedlo k normalizácii vnútorných a vonkajších zberníc a ich komunikačných protokolov. V moderných počítačových systémoch sa potom stretávame s rôznymi úrovňami zberníc:

- **vnútorná** (obvodová) zbernica prepájajúca funkčné jednotky vo vnútri integrovaného obvodu alebo tlačeneho spoja,
- **vonkajšia** (systémová, spoločná) zbernica prepájajúca zásuvné jednotky alebo funkčné celky,
- **V/V** (paralelné, sériové) zbernice pre jednotné pripojovanie rôznych periférnych zariadení. Na nasledujúcich obrázkoch sú naznačené zbernice PC.



Vstupné/ výstupné radiče



Pripojenie V/V zariadenia v PC pomocou zbernice

4.1.5 Privilegované inštrukcie

Niektoré inštrukcie by pri nevhodnom použití mohli viesť k zrušeniu operačného systému, k chybám alebo k narušeniu bezpečnosti. Jedná sa o I/O inštrukcie a niektoré inštrukcie riadenia procesora. Aby sa znemožnilo ich použitie používateľskými programami, má väčšina procesorov dva alebo viac stavov, s rôznym stupňom privilégií. Príkladom sú dva stavy: privilegovaný

a používateľský. Po štarte je procesor v privilegovanom stave a je spustený OS. V privilegovanom stave je dovolené prevádzkať i privilegované inštrukcie, beží v ňom operačný systém. V používateľskom stave pokus o vykonanie privilegovanej inštrukcie skončí chybou, používajú ho používateľské programy. Pri vyvolaní služieb operačného systému prechádza procesor do privilegovaného stavu, pri návrate zo systémovej služby prechádza do používateľského stavu.

4.1.6 Správa zariadenia

Účelom správy je zabezpečiť prístup k zariadeniu (pre operačný systém) štandardným spôsobom. Spravidla sa požaduje transparentnosť prístupu k zariadeniam (rovnaký prístup ako k súborom, kedy až pri behu programu sa dá určiť, kam výstup pôjde). Ďalšou úlohou je zaistiť pridelovanie a zdieľanie zariadení, ochrana zariadení (prístupové práva rôzne pre rôznych používateľov). Pridávanie nových druhov zariadení je realizované niekoľkými spôsobmi:

- zásahom do jadra operačného systému,
- nainštalovaním príslušného ovládača zariadení,
- kombináciou obidvoch možností.

Ovládače zariadenia majú tri časti:

- obslužný program prerušenia,
- časť závislú na zariadení,
- časť nezávislú na zariadení (správca vyrovnávacej pamäti, pomenovávanie zariadení apod.).

4.1.7 Časovač

Časovač umožňuje plánovať udalosti pri ktorých proces môže vyvolať službu operačného systému, ktorá ho zablokuje ("uspí") do uplynutia požadovaného času. Časovač sa taktiež používa pre niektoré systémové akcie (prideľovanie časových kvánt, zastavovanie motorov disketových mechaník apod.). Nie je možné predpokladať, že počítač má dostatok hardvérových časovačov pre všetky účely. Spravidla je k dispozícii len jeden časovač a programový časový front. Hardvérový časovač sa potom používa pre odmeranie času do najbližšej udalosti vo fronte. Do frontu sú udalosti riadené tým spôsobom, že sa zistí po ktorej udalosti má k novo pridávanej udalosti dôjsť a spočíta sa a zaznamená čas, ktorý má uplynúť medzi týmito dvoma udalosťami. Pri vynulovaní časovača sa vyberie prvá udalosť z frontu, vykoná sa príslušná akcia a hardvérový časovač sa preprogramuje na čas nasledujúcej udalosti.