

# PCI sběrnice

## studijní materiál k laboratornímu cvičení

### Úvod

Sběrnice PCI Local Bus (Peripheral Component Interconnect) je pseudosynchronní 32 nebo 64-bitová sběrnice. Ač se v názvu vyskytuje označení Local Bus, její postavení v architektuře PC neodpovídá lokální sběrnici. Mezi její přednosti patří zejména blokový přenos dat (burst mód) a podpora PnP. Umožňuje (podobně jako u sběrnice ISA) prodloužit sběrniceový cyklus. Na rozdíl od sběrnice ISA je přívlástek označující vlastnost synchronnosti skutečně na místě, neboť všechny události jsou vztaženy k vzestupným hranám synchronizačních (hodinových) pulsů. Přidělování sběrnice je centralizované. Zařízení, kterému je sběrnice přidělena, se nazývá master (iniciátor), „partner“ komunikace s iniciátorem je target (cíl). Základní frekvence hodinového signálu je 33 Mhz, norma definuje i verzi PCI 66 MHz. Špičkově je možné přenést data na každou hranu hodinového signálu, což implikuje špičkové přenosové rychlosti všech kombinací šířky dat a frekvence hodin takto (tabulka č. 1):

Šířka datové sběrnice	Frekvence hod. signálu	Špičková přenos. rychlost
32 bitů	33 MHz	132 MB/s
32 bitů	66 MHz	264 MB/s
64 bitů	33 MHz	264 MB/s
64 bitů	66 MHz	528 MB/s

**Tabulka 1: Přenosové rychlosti variant sběrnice PCI**

Na běžných deskách je implementována verze 32bitů/33 MHz, verze 66MHz je základem rozhraní AGP, 64 bitovou sběrnici nalezneme díky vyšší ceně implementace pouze na deskách určených pro serverové aplikace.

### Signály sběrnice

Norma PCI definuje 47 (resp. 49 pro zařízení typu Master) povinných signálů, které musejí být implementovány na každém zařízení PCI. Další signály jsou volitelné. Vzhledem k vysoké frekvenci hodinového signálu je na konektoru téměř každý čtvrtý pin uzemněn, norma definuje striktně např. i vzdálenosti konektorů, maximální délky vodičů na základní desce a zásuvných adaptérech, zatěžující kapacity a indukčnosti pinů obvodů rozhraní apod.

V dalším textu jsou podrobněji popsány pouze základní signály, se kterými se pracuje v rámci laboratorního cvičení. Na obrázku 1 je uveden přehled těchto signálů (znak # označuje inverzní signál, tj. aktivní v log. 0).

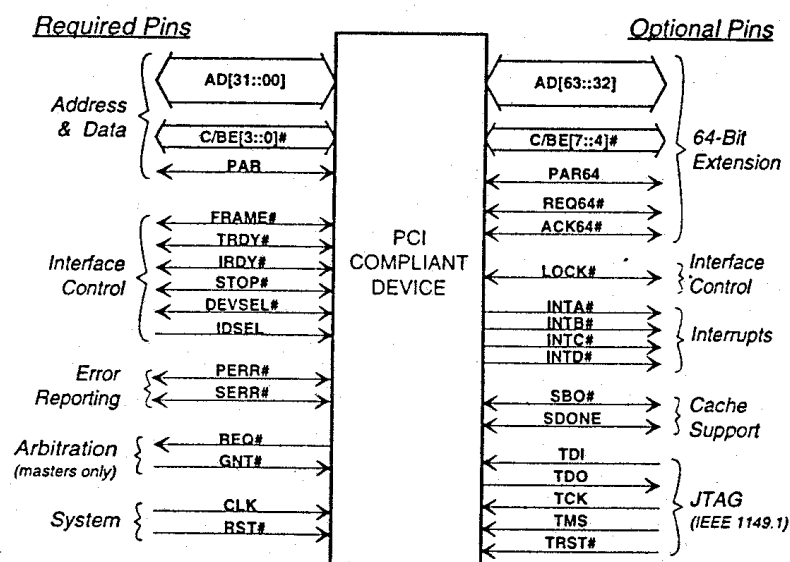
**CLK**            *Clock* je hodinový (synchronizační) signál  
**RST#**           *Reset* je asynchronní reset

**Datové signály:**

**AD[31::00]** *Address and Data* jsou multiplexované adresové a datové vodiče. Stav sběrnice určuje, zda je na těchto vodičích právě přenášena adresa či data.

**C/BE[3:0]#** *Bus Command and Byte Enables* jsou opět multiplexované signály. Po těchto vodičích se přenáší příkaz nebo příznak platnosti dat. Význam je opět určen stavem sběrnice.

**PAR** *Parity* je sudá parita. Zabezpečuje společně signály AD a C/BE.



Obrázek 1: Signály sběrnice PCI

### Signály k řízení sběrnice:

- FRAME#** *Cycle Frame* řídí transakci (sběrnice cyklus). Určuje její začátek a konec.
- IRDY#** *Initiator ready* oznamuje, že iniciátor je připraven k datovému přenosu. V případě zápisu to znamená, že jsou na datové sběrnici platná data, v případě čtení indikuje, že přečte přenášená data.
- TRDY#** *Target ready* oznamuje, že cílové zařízení je připraveno k datovému přenosu.
- STOP#** Signál *Stop* je řízen cílem (targetem). Žádá iniciátor o zastavení probíhající transakce.
- IDSEL#** *Initialization Device Select* je dvoubodový spoj, vedoucí od řadiče sběrnice ke každému zařízení. Slouží k výběru zařízení (jako chip-select) při konfiguraci zařízení.
- DEVSEL#** *Device Select* je výstup adresového dekodéru zařízení. Indikuje aktuálnímu masteru, zda bylo nějaké zařízení vybráno (adresou na sběrnici).
- LOCK#** Umožňuje zamknout sběrnici pro provedení atomické transakce.

### Signály k řízení přidělování sběrnice:

- REQ#** Signál *Request* je dvoubodový signál. Pomocí něj žádá zařízení (potenciální master) přidělovač sběrnice o její přidělení.
- GNT#** Signál *Grant* je opět dvoubodový. Přidělovač sběrnice oznamuje tímto žádajícímu zařízení, že mu sběrnice byla přidělena (odpověď na signál REQ#).

### **Hlášení chyb:**

- PERR#** *Parity Error* se používá k hlášení chyby parity. Řídí jej příjemce dat.  
**SERR#** *System Error* je generován při chybě parity během speciálního cyklu nebo při jiné chybě, např. přetečení adresy.

### **Přerušení:**

- INTA#** Signály žádosti o přerušení (nepovinné). Každé konstruované zařízení by mělo primárně využít signál INTA#, pouze u tzv. multifunkčních zařízeních (více zařízení v jednom čipu s jedním rozhraním nebo více zařízení na jednom adaptéru) mohou být zapojeny také signály další.

## **Příkazy (operace):**

Řízení sběrnice PCI je navrženo inteligentněji než např. u sběrnice ISA. Typ cyklu (čtení/zápis do paměti, na port ...) není určen speciálními signály (MEMW, IOW), ale příkazem přenášeným k zařízení pomocí vodičů C/BE#. Seznam příkazů je v následující tabulce č. 2:

<b>C/BE[3::0]#</b>	<b>Typ příkazu</b>
0000	Potvrzení přerušení (Interrupt Acknowledge)
0001	Speciální cyklus (Special Cycle)
0010	Čtení z portu (I/O Read)
0011	Zápis na port (I/O Write)
0100	Rezervováno (Reserved)
0101	Rezervováno (Reserved)
0110	Čtení z paměti (Memory Read)
0111	Zápis do paměti (Memory Write)
1000	Rezervováno (Reserved)
1001	Rezervováno (Reserved)
1010	Konfigurační čtení (Configuration Read)
1011	Konfigurační zápis (Configuration Write)
1100	Memory Read Multiple
1101	Dual Address Cycle
1110	Memory Read Line
1111	Memory Write and Invalidate

**Tabulka 2: Přehled příkazů**

### **Poznámky:**

*Interrupt Acknowledge* je žádost o potvrzení přerušení. Na základě tohoto příkazu vysílá zařízení na datovou sběrnici číslo přerušení (IRQx), které mu bylo přiděleno v rámci PnP.

*Speciální cyklus* představuje mechanismus „broadcastu“ na sběrnici. V datech je posílána zpráva pro všechna zařízení. Význam zpráv definuje příloha A normy.

Konfigurační čtení a zápis slouží ke čtení/zápisu konfiguračních registrů zařízení. Využívá zejména v rámci PnP.

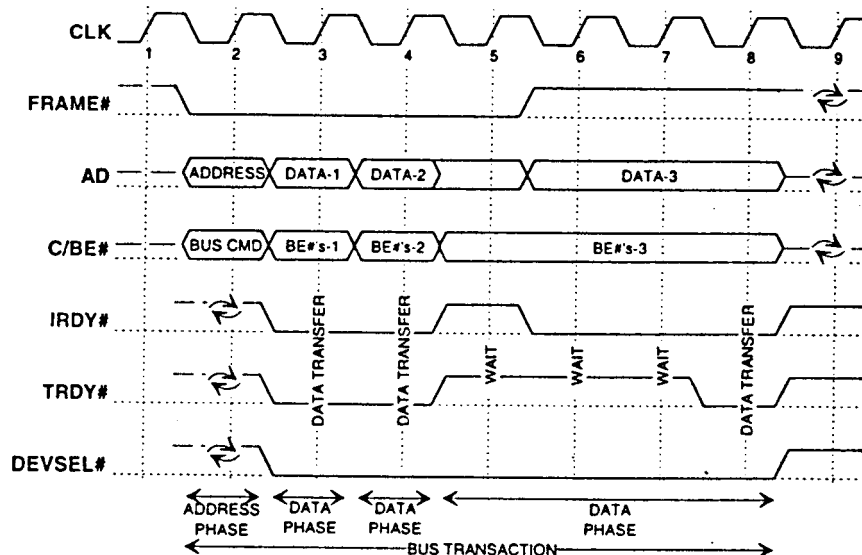
Dual Address Cycle umožňuje adresovat zařízení se šířkou sběrnice 32 bitů pomocí 64 bitové adresy, která se zapisuje pomocí tohoto cyklu ve dvou fázích.

Memory Read Multiple, Memory Read Line, Memory Write and Invalidate se liší od normálního čtení/zápisu podporou vyrovnávací paměti.

## Popis nejdůležitějších sběrniceových cyklů (transakcí)

### Zápis do paměti/čtení z paměti

Sběrniceový cyklus je v normě označován pojmem *transakce*. Na obrázku č. 2 je příklad zápisu do paměti. Připomeňme ještě jednou, že všechny události jsou vztaženy k vzestupné hraně hodinových pulsů. Cyklus začíná aktivováním signálu FRAME# (sestupná hrana v cyklu 1). První fáze transakce se nazývá adresová a trvá jeden takt. Na adresových/datových vodičích je platná adresa a na vodičích C/BE# je přenášén příkaz (hodinový cyklus 2).<sup>1</sup> Další fáze slouží k přenosu dat. Na adresových/datových vodičích jsou pak přenášena data, vodiče C/BE# určují platnost dat. Na obrázku je znázorněn blokový přenos, kdy na jednu adresovou fázi připadá několik datových. V tomto případě dochází k autoinkrementaci adresy (o hodnotu 4). Špičkově je možné přenést blok dat na každou vzestupnou hranu hodin (viz cykly 3 a 4). Pokud není iniciátor, resp. cíl připraven k přenosu, oznamuje skutečnost neaktivitou signálu IRDY#, resp. TRDY#. Např. v taktu 5 se nepřenášejí žádná data, protože ani iniciátor ani cíl nejsou připraveny k přenosu; v taktu 6 a 7 je již iniciátor (data jsou platná), ale cíl stále není schopen data převzít (čekací -wait- stavy). Přenos bloku dat číslo 3 se uskuteční až v taktu 8.

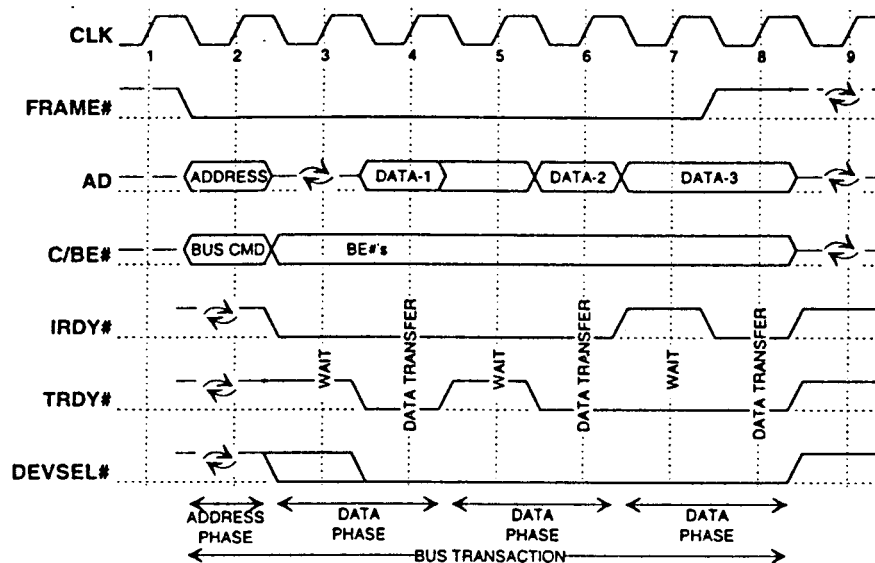


Obrázek 2: Signálový sled při zápisu do paměti

Zajímavě je u PCI řešeno ukončení cyklu. Přejít signálu FRAME# do neaktivního stavu neznamená překvapivě ukončení transakce, ale signalizuje, že transakce je v poslední

<sup>1</sup> Z obrázku vyplývá, že cílové zařízení musí zachytit adresu při první vzestupné hraně hodinového signálu poté, co se signál FRAME# stane aktivním.

fázi, tj. proběhne ještě jeden přenos. Pokud by byla obě zařízení připravena, proběhl by poslední přenos v cyklu 6, v této ukázce se se tak děje díky čekacím stavům až v cyklu 8. Jinými slovy, cyklus končí tehdy, jsou-li oba signály FRAME# a IRDY# neaktivní. Čtení

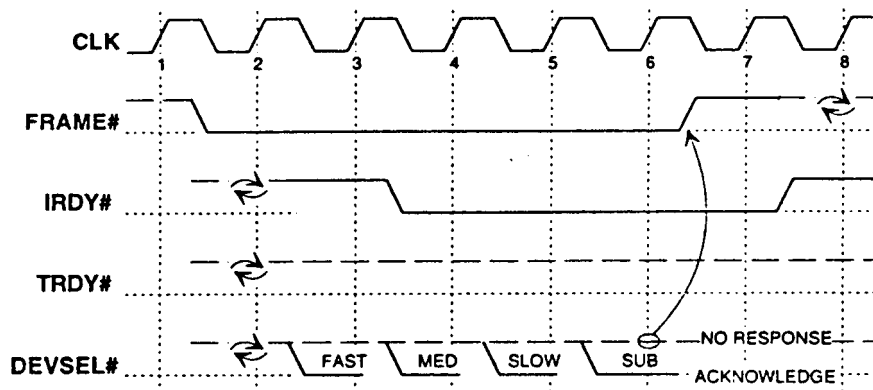


Obrázek 3: Čtení z paměti

z paměti probíhá obdobně a je zachyceno na obrázku č. 3 (s tím rozdílem, že příkaz BUS CMD v adresové fázi je Memory Read). Za zmínku stojí ještě adresování při operaci s pamětí. Adresa je v tomto případě vždy dělitelná 4, dva nejnižší bity adresy v adresní fázi mají jiný význam (informují o způsobu autoinkrementace adresy při blokovém přenosu, většinou je lineární, jiný se může použít při příkazech podporující cacheline - viz. norma). Zápis na jiné adresy se realizuje posuvem dat po datové sběrnici v součinnosti se signály C/BE#.

Nejlépe vše osvětlí příklad. Předpokládejme, že registr ES obsahuje adresu ES=B800h. Procesor vykoná instrukci MOV ES:[0], AL (v reálném módu). Adresa na sběrnici bude 000B8000h. Platná data se budou přenášet na vodičích AD[7::0], bity vyššího řádu dat budou obsahovat libovolnou hodnotu, která musí být stabilní kvůli generování parity (nejčastěji samé log. 1). Platnost dat je určena vodiči C/BE#[3::0]=1110b. Bude-li mít vykonávaná instrukce offset 1, tj. MOV ES:[1], AL, platná adresa v adresové fázi bude stejná 000B8000h, ale posunutí je realizováno až v datové fázi. Tentokrát budou data přenášena na vodičích AD[15::8] a příznak platnosti dat bude C/BE#[3::0]=1101b. Nahradíme-li v první instrukci registr AL za AX, změní se v datové fázi pouze šířka přenášených dat - jsou využity vodiče AD[15::0] a příznak je C/BE#[3::0]=1100b. Další varianty (různé posuvy adres, přenos z registru EAX) jsou již zřejmé.

Všimneme si ještě signálu DEVSEL#. Ten je výstupem adresového dekodéru zařízení. Je-li DEVSEL# aktivován po adresové fázi, dává tím iniciátoru nějaké zařízení najevo, že rozpoznalo svoji adresu a je „ochotné“ komunikovat. Norma stanoví tři možné časy, do kdy musí zařízení aktivovat signál DEVSEL# po adresní fázi. Rozlišují se zařízení FAST, MEDIUM a SLOW (obrázek č. 4). Pak následuje tzv. subtraktivní adresování, kdy na sběrnici může existovat jen jedno zařízení bez přidělené adresy a to se stává cílem v případě, že se do této doby žádné jiné zařízení „nepřihlásí“ pomocí DEVSEL#. V opačném případě iniciátor násilně ukončí transakci bez přenosu dat.



Obrázek 4: Časování signálu DEVSEL#

### Zápis na port/čtení z portu

Operace zápisu na port a čtení z portu se v podstatě neliší od operací pro přenos dat do/z paměti. Rozdíl je pouze v tom, že adresu portu může mít libovolnou hodnotu (není zde podmínka dělitelnosti 4).

### Speciální cyklus

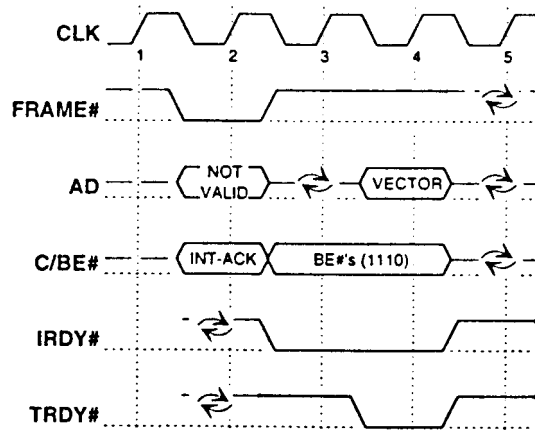
Speciální cyklus představuje jakýsi mechanismus broadcastu na sběrnici PCI. Takto se zasílají jednorázově zprávy všem zařízením (např. provedení sw resetu zařízení). Cyklus je opět podobný zápisu do paměti. Jako příkaz je zasílán v adresové fázi příkaz *Special Cycle* (číselně 0001b). Adresa je v tomto případě nesmyslné číslo (musí být opět stabilní pro generování parity) - jakmile se na sběrnici objeví příkaz speciálního cyklu, data musejí převzít všechna zařízení, adresa je tedy zbytečná. Signál DEVSEL# se nikdy při speciálním cyklu neaktivuje. Většinou se přenáší pouze jedno 32-bitové číslo, které je označováno jako zpráva (hlášení). Na datových vodičích AD[15::0] je kód zprávy, vodiče AD[31::16] nesou doplňkovou informaci. Kódy zpráv definuje příloha A normy. Ve verzi PCI 2.1 jsou pouze 3 zprávy, ostatní kódy jsou prozatím rezervovány (tabulka č. 2).

AD[15::0]	Zpráva
0000h	SHUTDOWN
0001h	HALT
0002h	Specifické pro architekturu x86

Tabulka č. 2: Zprávy zasílané při speciálním cyklu

### Potvrzení přerušení

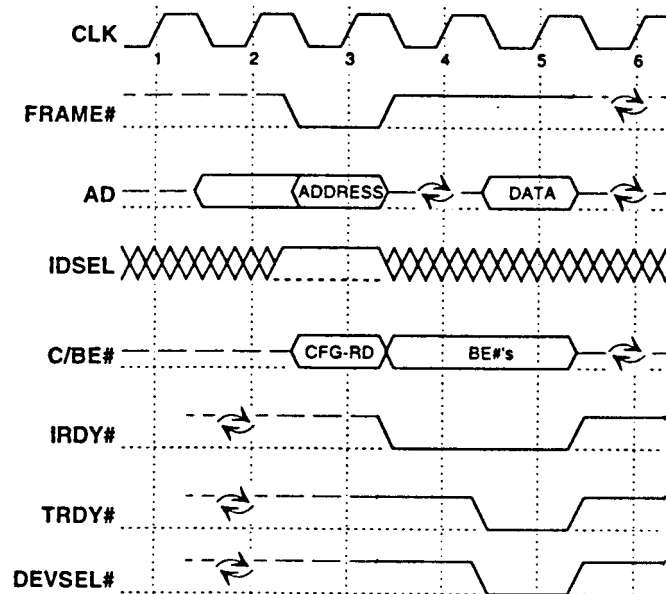
Při tomto cyklu zasílá zařízení vektor přerušení (IRQx) přidělený při konfiguraci (obrázek č. 5).



Obrázek 5: Potvrzení přerušení

### Konfigurační čtení/zápis

Při těchto transakcích se přistupuje ke konfiguračním registrům zařízení. Oproti klasickému čtení/zápisu je zde zásadní rozdíl v adresaci. Zařízení není vybíráno adresou v adresové fázi, ale dvoubodovým signálem IDSEL, který je zapojen od řadiče sběrnice (můstku) ke každému zařízení. Adresové vodiče během adresní fáze nesou speciální informaci (viz dále). Přenos dat se neliší od předchozích cyklů (obrázek č. 6).

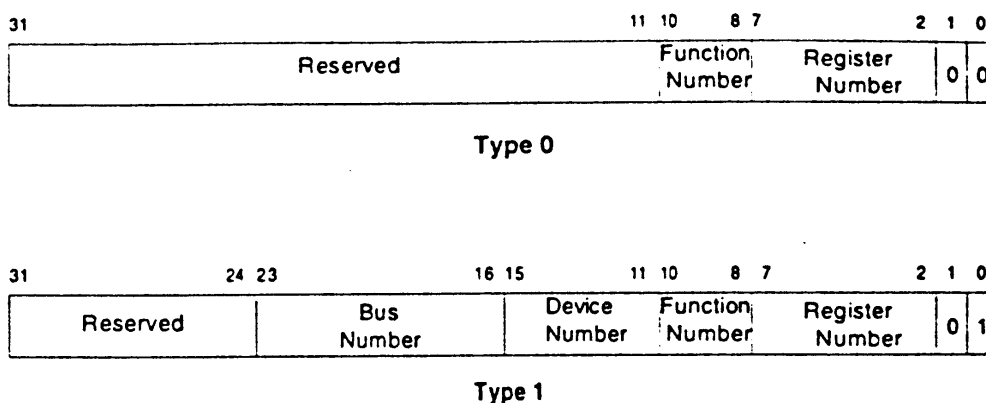


Obrázek 6: Konfigurační čtení

Norma definuje 2 typy konfiguračních přístupů, které jsou rozlišeny vodiči AD[1::0] v adresové fázi:

*Type 00* - určen pro zařízení na sběrnici, na které je spuštěn

*Type 01* - určen pro zařízení na jiné sběrnici (běžné zařízení musí tento přístup ignorovat, reagují pouze PCI-to-PCI můstky).



**Obrázek 7: Rozlišení konfiguračních přístupů**

V systému totiž může existovat více nezávislých PCI sběrnic propojených PCI-to-PCI můstky a je potřeba zajistit šíření konfiguračních cyklů.

Obrázek č. 7 ukazuje formát přenášené informace na vodičích AD. Dva bity nejnižšího řádu určují typ konfiguračního přístupu. *Register Number* je adresa konfiguračního registru, který má být zapisován/čten. *Function Number* se využívá u tzv. multifunkčních zařízení (adaptér nebo čip, který má jedno rozhraní na sběrnici, ale sdružuje více zařízení). Tato položka rozlišuje, ke kterému konfiguračnímu prostoru se bude přistupovat.

U konfiguračního přístupu typu 1 jsou navíc přenášeny informace *Device Number* a *Bus Number*. Objeví-li se při konfiguračním cyklu na sběrnici typ 1, reagují pouze mezsběrnicové můstky, které převezmou data. Podle čísla sběrnice přenesou typ nebo konvertují konfigurační cyklus na typ 0 a podle *Device Number* aktivují příslušný signál IDSEL#.

## Konfigurační mechanismy

Konfigurační mechanismus je postup, jak spustit softwarově v počítači PC konfigurační cyklus na sběrnici PCI. Existují dva způsoby, označené konfigurační mechanismus 1 a 2. Druhý mechanismus je starší a není doporučeno jej v PC ( uvnitř Host-to-PCI můstku) již implementovat. Dále popíšeme jen mechanismus číslo 1.

V každém PC jsou dva porty na adresách CF8h a CFCh označené takto:

CF8h	DWORD registr CONFIG_ADDRESS
CFCh	DWORD registr CONFIG_DATA

Postup pro vyvolání konfiguračního cyklu je následující:

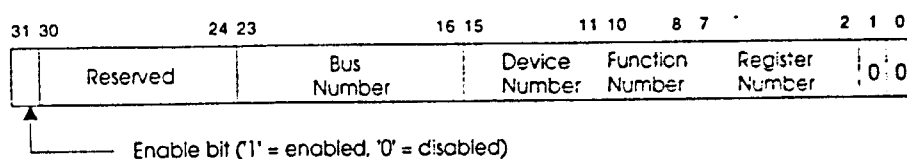
1. Zapišeme požadavek na konfiguraci (adresu zařízení, adresu registru) na port CF8h pomocí instrukce OUT. Zápis musí být 32 bitový.
2. Instrukci IN nebo OUT podle toho, zda chce číst nebo zapisovat z/do konfiguračního prostoru, vyvoláme konfigurační cyklus. Instrukce může být pracovat s daty o šířce 8, 16 nebo 32 bitů.



Formát dat, který se zapisuje na port CF8h, je téměř shodný s tím, co je vysláno v adresové fázi konfiguračního cyklu - tedy zapisujeme číslo sběrnice, číslo zařízení, adresu konfiguračního registru, identifikaci v rámci multifunkčního zařízení (obrázek č. 8).

Nejvyšší bit musí být nastaven na hodnotu log. 1, jinak můstek vyvolá na sběrnici operaci čtení/zápis z/na port. Bity [1::0] se nastavují v adresovém registru na 00. Můstky rozhodnou, zda je konfigurační požadavek určen jejich sběrnici a rozhodnou o typu mechanismu mezi 00 a 01.

Pomocí konfiguračního mechanismu 1 je možné generovat na sběrnici speciální cyklus. Nastavíme-li pole *Device Number*, *Function Number* binárně na samé 1 (srovnej s broadcastem v IP protokolu) a *Register Number* na hodnotu 0, můstek vybudí místo konfiguračního zápisu speciální cyklus.



**Obrázek 8: Formát dat zapisovaných do CONFIG\_ADDRESS**

## PCI BIOS

Konfigurační cykly je možné spouštět i pomocí PCI BIOSu. PCI BIOS je mapován na softwarové přerušení INT 1Ah, funkce AH=B1h.

*Některé služby:*

- AL=01h Installation check**
  - test instalace PCI sběrnice
- AL=02h Find PCI Device**
  - prohledávání PCI sběrnice
- AL=88h Read Configuration Byte**
  - prohledávání PCI sběrnice
- AL=8Fh Set PCI IRQ**
  - nastavení přerušení

**Příklad:** čtení slabiky z konfiguračního prostoru

AX = B188h

BH = číslo sběrnice

BL = číslo zařízení/číslo podfunkce v rámci multifunkčního zařízení  
(bity 7-3 zařízení, bity 2-0 podfunkce)

DI = adresa registru v konfiguračním prostoru (0000h-00FFh)

Více najdete např. na <http://www.delorie.com/djgpp/doc/rbinter/ix/1A/B1.html>

## Konfigurační prostor

Každé zařízení připojené ke sběrnici PCI musí implementovat konfigurační prostor o velikosti 256B. Přístup k těmto registrům se uskutečňuje pomocí operace *Konfigurační čtení/zápis*. Uložená data jsou ve formátu little-endian. Je možné číst/zapisovat data o šířce 8, 16 nebo 32 bitů.

Na obrázku č. 9 je zobrazen konfigurační prostor. Povinná podpora je pro pole *DeviceID*, *VendorID*, *Status*, *Command*, *Class Code*, *RevisionID* a *Header Type*. Implementace ostatních registrů je volitelná a závisí na typu zařízení. Rozlišují se dva typy konfiguračních prostorů podle obsahu bitů 0-6 pole *Header Type*. Typ 00 je určen pro běžná zařízení (standardní tvar - zobrazen na obrázku), typ 01 pro PCI-to-PCI můstky. Bit 7 tohoto pole určuje, zda jde o multifunkční zařízení.

Device ID		Vendor ID		00h
Status		Command		04h
Class Code			Revision ID	08h
BIST	Header Type	Latency Timer	Cache Line Size	0Ch
Base Address Registers				10h
				14h
				18h
				1Ch
				20h
CardBus CIS Pointer				24h
CardBus CIS Pointer		CardBus CIS Pointer		28h
Subsystem ID		Subsystem Vendor ID		2Ch
Expansion ROM Base Address				30h
Reserved				34h
Reserved				38h
Max_Lat	Min_Gnt	Interrupt Pin	Interrupt Line	3Ch

**Obrázek 9: Konfigurační prostor**

### *Význam některých polí*

**VendorID** je číselná identifikace výrobce, např:

0x0018	Creative Electronic Systems
0xFFFF	neplatný kód výrobce

Neplatný kód výrobce se využívá při prohledávání PCI sběrnice. Systémový software se postupně dotazuje všech zařízení na sběrnici na hodnotu VendorID. Pokud není zařízení fyzicky přítomné pod dotazovaným Device Number, můstek musí vrátit hodnotu 0xFFFF. Software tak pozná, pod kterým Device Number jsou zařízení připojena na sběrnici.

**DeviceID** je číselná identifikace výrobku (určuje ji výrobce). Seznam čísel můžete nalézt např. na <http://www.pcidatabase.com/>

**RevisionID** je verze výrobku (určuje výrobce).

**ClassCode** představuje specifikaci třídy zařízení. Je rozdělen na 3 části po slabikách (*class*, *subclass*, *interface*). *Class* určuje třídu zařízení (např. síťový adaptér, radič,...), *subclass* konkretizuje zařízení (např. Ethernet adaptér); hodnotu *interface* určuje výrobce a může sloužit např. pro rozlišení verzí při instalaci ovladače. Číselné kódy definuje příloha D normy.

Třída	Význam
00h	Zařízení vyrobené před specifikací
01h	Mass storage controler
02h	Network controller
03h	Display controller
04h	Multimedia device
05h	Memory controller
06h	Bridge device
07h	Simple communication controller
08h	Base system peripheral
09h	Input device
0Ah	Docking station
0Bh	Processor
0Ch	Serial bus controller
0D-FEh	Reservováno

Podtřída *Multimediální zařízení*

Třída	Podtřída	Interface	Význam
04h	00h	00h	Zařízení typu video
	01h	00h	Zařízení typu audio
	80h	00h	Jiné multimed. zařízení

**BIST (Built-in self test)** nabízí možnost spuštění interního testu zařízení (pokud je implementováno) a zjištění výsledku.

Pro řízení zařízení a zjišťování stavu slouží dva registry *Command* a *Status*:

**Command** je příkazový register, pomocí něho je možné ovlivnit chování zařízení, např:

- řízení odezvy na požadavek provedení I/O (memory) operace (umožňuje zakázat/povolit reakci na)
- povolit možnost být masterem
- povolit generování **SERR#** nebo ignorovat chyby

**Status** je stavový register, umožňuje přečíst informace o zařízení, např:

- schopnost pracovat na 66MHz
- časování **DEVSEL#** (SLOW, MED, FAST)
- detekce chyb parity a systémových chyb

- o detekce ukončení transakce (Abort)

### Přerušeni

**Interrupt Pin** - zde je od výrobce zapsáno, který signál INTx# je zapojen (0=nepoužívá přerušeni, 1=INTA#, ...)

**Interrupt Line** - číslo přerušeni (linky), přiděluje se v rámci PnP

### Bázové registry adres

Každému zařízení je možné přidělit adresový prostor. Počáteční (bázové) adresy jsou zapsány v části Base Address Registers (jak pro V/V, tak pro paměťový prostor). Bázových registrů je více, protože zařízení může být implementováno více oddělených prostorů s různými bázovými adresami. Zda je příslušný registr přidružen k V/V nebo paměťovému prostoru, určuje nejnižší bit (obrázek č. 10). Nejprve tedy systém přečte registry a zjistí, k jakému prostoru jsou vázány. Dále je potřeba před zapsáním bázové adresy zjistit nároky na velikost prostorů. Podle normy je nejprve třeba zapsat binárně samé log. 1 do registru a vzápětí přečíst zpět obsah. Zařízení vrátí bitovou masku určující velikost nárokovaného prostoru.

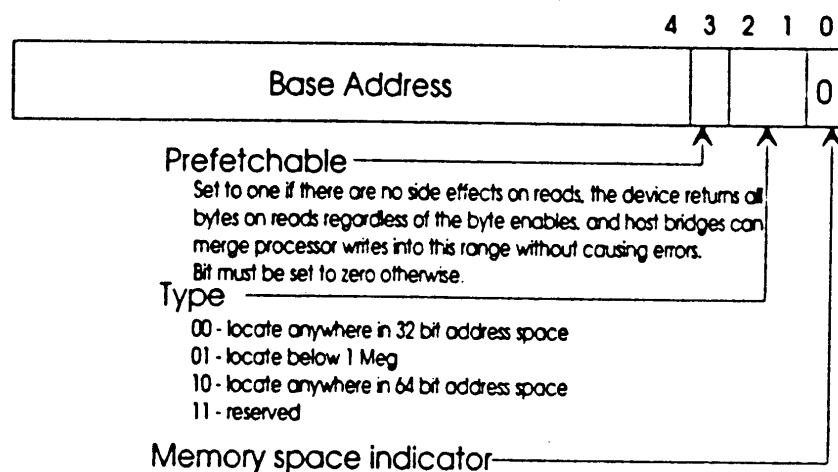
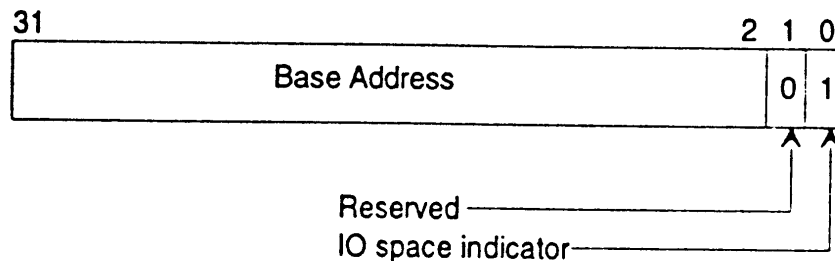


Figure 6-5: Base Address Register for Memory



Obrázek 10: Formát bázových registrů adres