

36PZ – Periferní zařízení

Laboratorní úloha A:

Měření na PCI sběrnici

Michal Augustýn

Josef Bouška

Martin Prchlík

Michal Trs

Datum měření: 11.4.2006

1 Zadání

Cílem tohoto měření je sledování základních cyklů sběrnice PCI a porozumění konfiguraci zařízení PCI.

1. Změřte průběhy signálů základních cyklů na PCI sběrnici.
2. Pomocí programu `dumppci` zjistěte informace o PCI sběrnici a typy připojených adaptérů.

2 Teorie

2.1 Měření průběhů signálů

Měření průběhů proveďte pomocí dvou počítačů PC. Jeden z dvojice počítačů je vybaven měřicí kartou PCI sběrnice. Měřicí karta je osazena obvodem XILINX, který snímá průběhy jednotlivých signálů a ukládá je do dynamické paměti (DIMM moduly umístěné přímo na měřicí kartě). Měřicí karta je propojena s druhým počítačem přes paralelní port. Tento počítač je vybaven sběrným softwarem, který zobrazí snímané průběhy.

2.1.1 Postup měření:

Počítač, který je vybaven měřicí kartou, restartujte v módu MS-DOS. Přejděte do adresáře `c:\testpci`. V tomto adresáři je umístěn program `testpci`, který generuje sběrníkové cykly.

Program `testpci` je napsán v assembleru. Zdrojový kód máte k dispozici. Prostudujte jej. Program se spouští vždy s parametrem; podle něj se generují sběrníkové cykly. Parametry programu jsou následující:

```
mread.....čtení jednoho bytu z paměti
mwrite.....zápis jednoho bytu do paměti
ioread.....čtení 8-bitového portu
iowrite....zápis 8-bitového portu
bytermerge..spojování několika 8-bitových zápisů
cfgread....čtení konfiguračního registru
longwrite..zápis několika po sobě následujících dwordů
iack.....potvrzování přerušování, vyslání vektoru
special....vyvolání speciálního cyklu
```

Sejměte průběhy všech cyklů, které program umí generovat (mimo `iack`). Uvedte je do protokolu a patřičně okomentujte.

2.1.2 Poznámky k měření:

2.1.2.1 Čtení/zápis do paměti

Smyčka pro tento parametr zapisuje jednu slabiku do videopaměti (na adresu `B800h:0000h`). Vyzkoušejte různé varianty zápisu, zejména zápis slabiky na lichou adresu, zápis na adresu, která není dělitelná čtyřmi, zápis slova (word), dvojslova (double word). Pozorujte, jak se změny v programu projeví na přenášené adrese a signálech `C/BE#`.

2.1.2.2 Varianty zápisu s parametrem `bytemerge`, `longwrite`

Zjistěte, pro jaké typy přenosu dat (jaké části kódu) optimalizuje můstek Host-to-PCI zápisy (spojení do blokového přenosu apod.). Zjistěte, co se stane, změníte-li adresy zápisu v případě varianty `longwrite` (např. posun offsetu všech adres o 1).

2.1.2.3 Čtení konfiguračního registru

Analyzujte kód programu a zjistěte, z kterého zařízení (jaké ID) a jaké konfigurační registry jsou čteny. Z přenášených dat na sběrnici dekódujte obsah těchto registrů.

2.1.2.4 Speciální cyklus

Zasílaná zpráva během speciálního cyklu je z množiny rezervovaných kódů, tj. bez významu. Popište sled signálů. Upravte kód programu tak, aby byla zasílána zpráva SHUTDOWN. Pozorujte chování počítače.

POZN.: Cykly v jednotlivých podprogramech jsou realizovány instrukcí `loop`. Jde tedy o cykly s pevným počtem opakování. Program se neukončuje stiskem libovolné klávesy, jak tomu většinou bývá.

2.2 Zjištění připojených PCI adaptérů

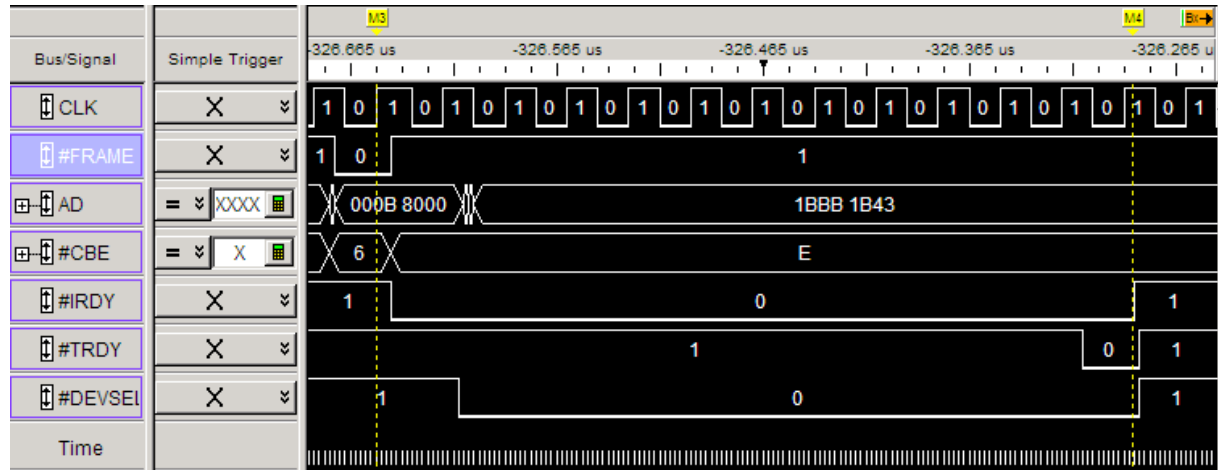
Prostudujte zdrojový kód programu `dumppci.c`, zejména využití jednotlivých služeb PCI BIOSu. Program spusťte a zjistěte informace o PCI sběrnici a připojených zařízeních. Na základě výpisu uveďte do protokolu tabulku s dekódovanými informacemi: výrobce zařízení a typ výrobku (číselné označení i jméno výrobce – vyhledejte na internetu), třídu zařízení podle přílohy D normy PCI, přidělenou adresu počátku paměťového a V/V prostoru, přerušovací vektor, časování signálu DEVSEL.

Pro zájemce: Konfigurační registr `COMMAND` obsahuje bit *Memory Space*. Nastavením tohoto bitu na hodnotu `log. 0` zakážeme příslušnému zařízení akceptovat příkazy přistupující k paměťovému prostoru (memory read/write). Upravte smyčku `cfread` na zápis tak, aby byl nulován tento bit na VGA kartě (nejprve musíte zjistit její ID z výpisu všech zařízení). Pozorujte její chování.

3 Měření

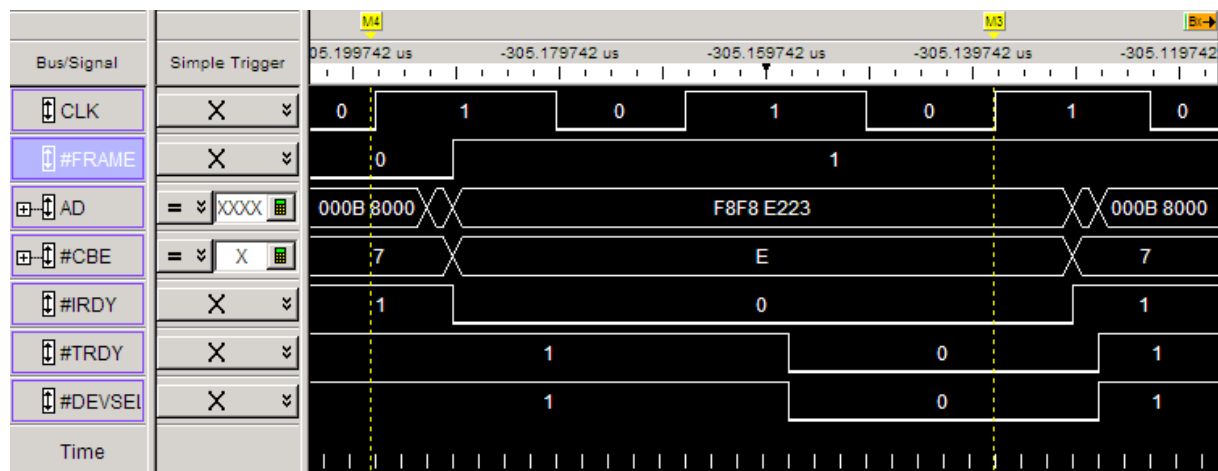
3.1 Měření průběhů signálů

3.1.1 Memory read - čtení jednoho bytu z paměti



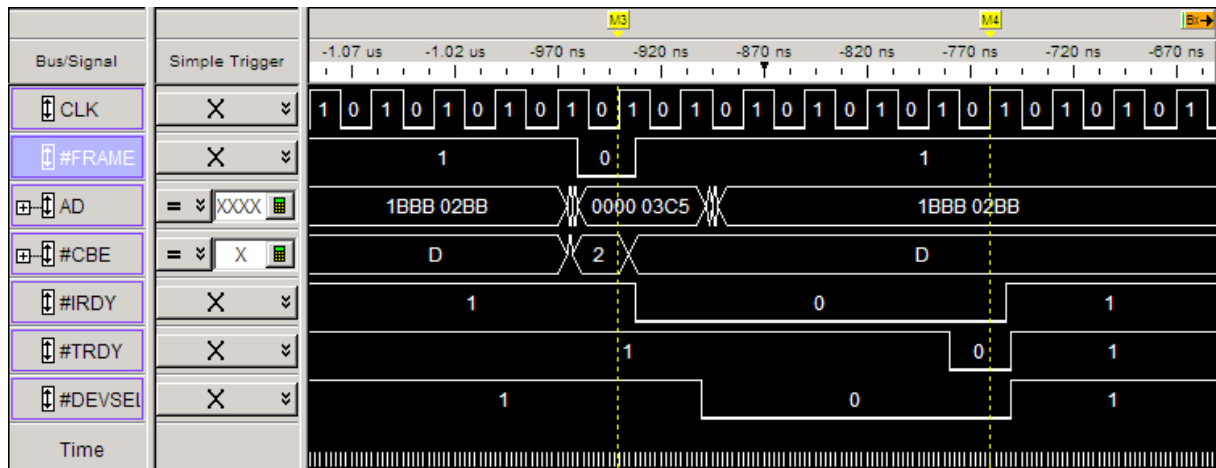
Popis: Načteme (#CBE == 6) z adresy B8000h dolních 8 bitů (#CBE == E), tj hodnotu 43h. Operace trvá 13 taktů, z toho 11 taktů čeká na připravenost zařízení.

3.1.2 Memory write - zápis jednoho bytu do paměti



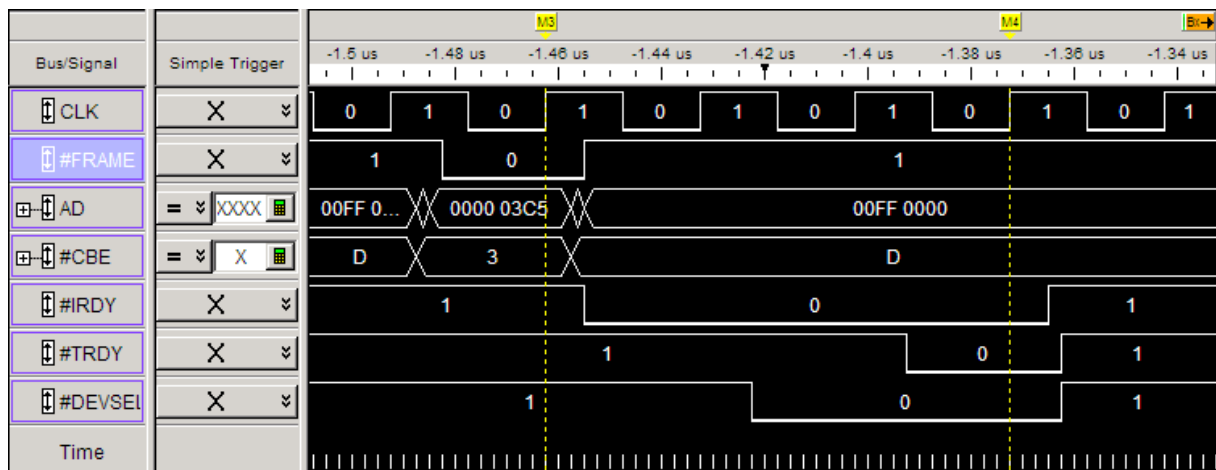
Popis: Zapišeme (#CBE == 7) na adresu B8000h dolních 8 bitů (#CBE == E), tj hodnotu 23h. Operace trvá 3 takty, zařízení je připraveno ihned po rozpoznání adresy (#DEVSEL).

3.1.3 IO read - čtení 8-bitového portu



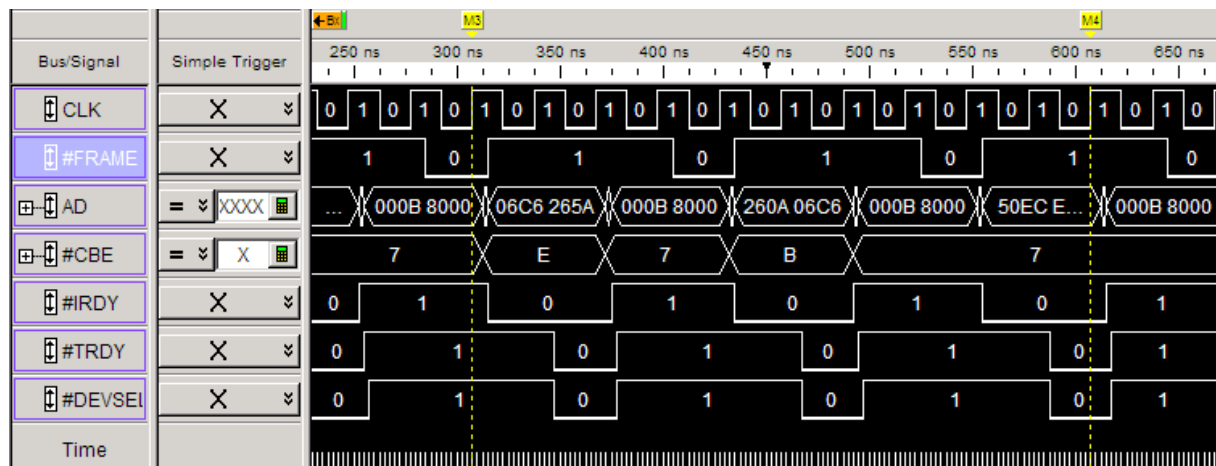
Popis: Načítáme ($\#CBE == 2$) z IO portu na adrese 03C6h 8 bitů ($\#CBE == D$), tj hodnotu 02h. Operace trvá 7 taktů, 5 taktů se čeká než bude zařízení připraveno.

3.1.4 IO write - zápis 8-bitového portu



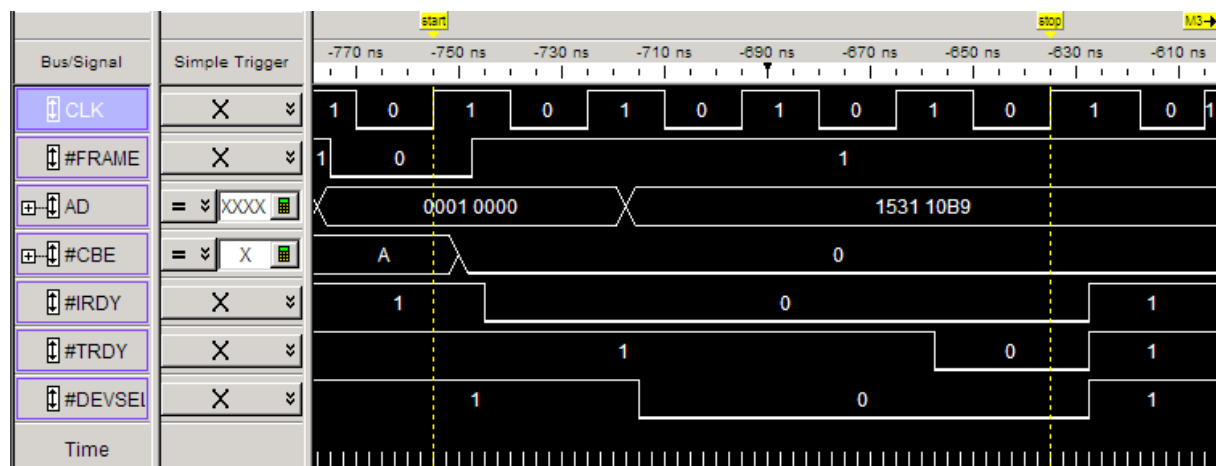
Popis: Zapisujeme ($\#CBE == 3$) na IO port na adrese 03C6h 8 bitů ($\#CBE == D$), tj hodnotu 00h. Operace trvá 4 taktů, zařízení je připraveno za 1 takt od rozpoznání adresy ($\#DEVSEL$).

3.1.5 Bytemerge - spojování několika 8-bitových zápisů



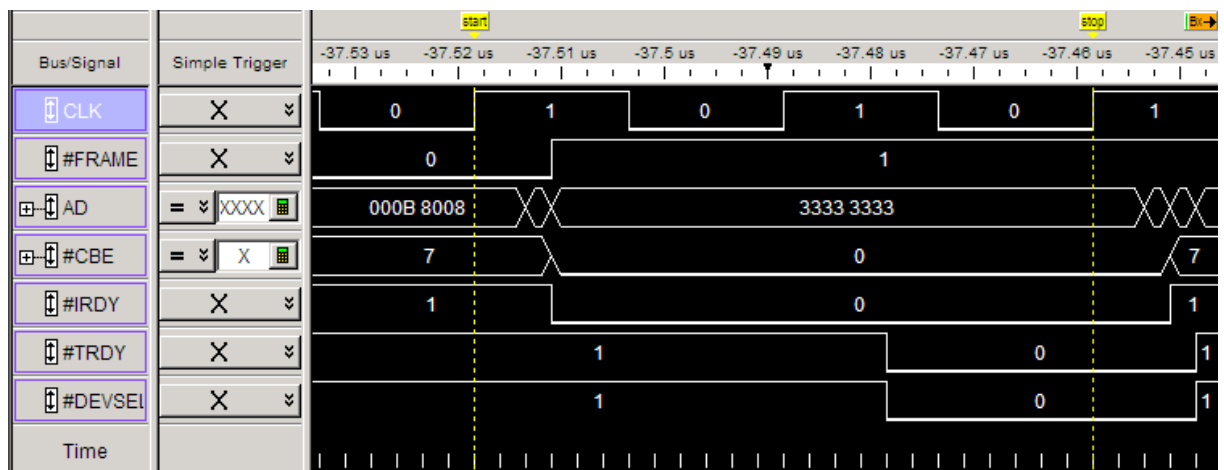
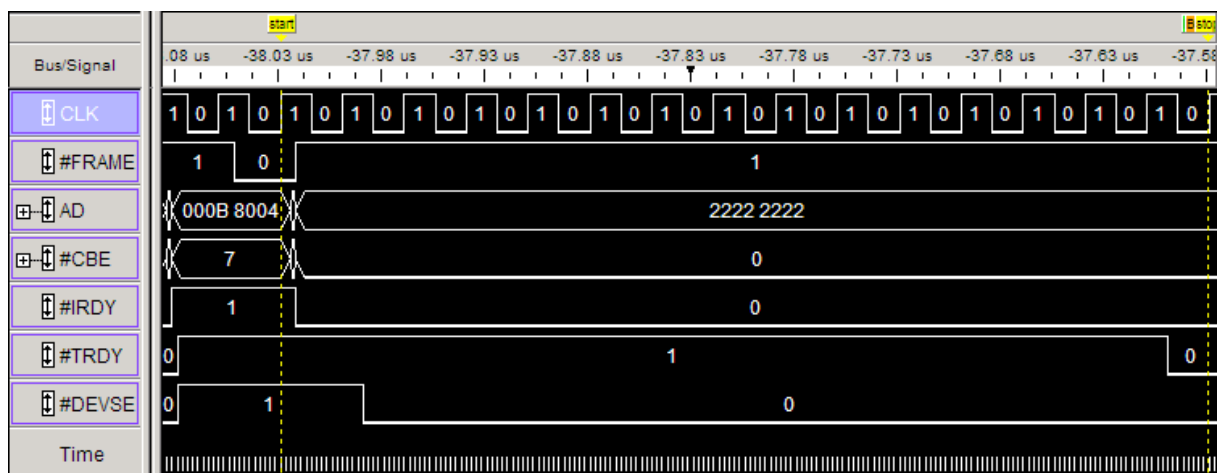
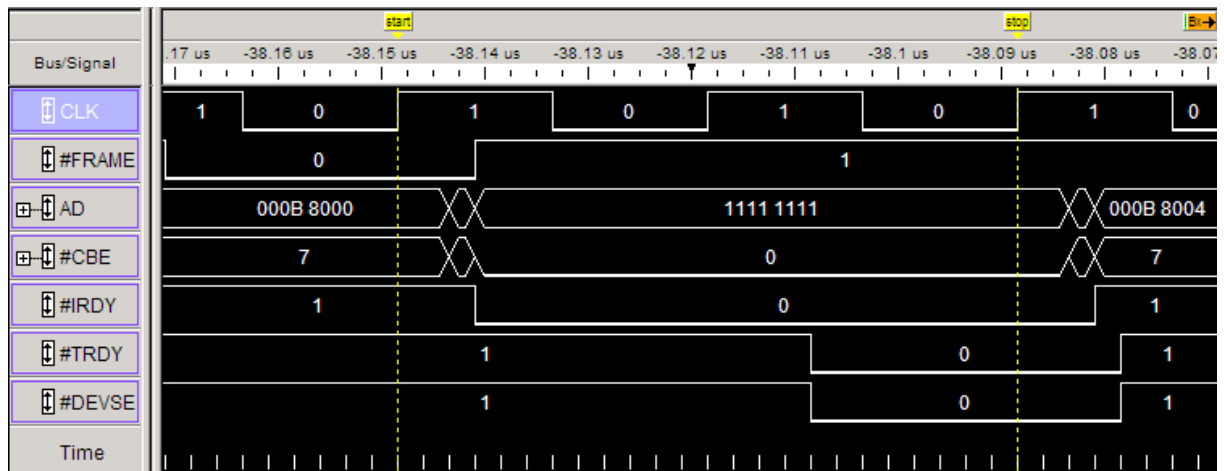
Popis: Zapisujeme ($\#CBE == 7$) do paměti od adresy B8000h 24 bitů, tj hodnotu 5Ah ($\#CBE == E$) na adresu B8000h, 0Ah na adresu B8002h ($\#CBE == D$) a 50h na adresu B8003h ($\#CBE == 7$). Operace trvá celkem 11 taktů.

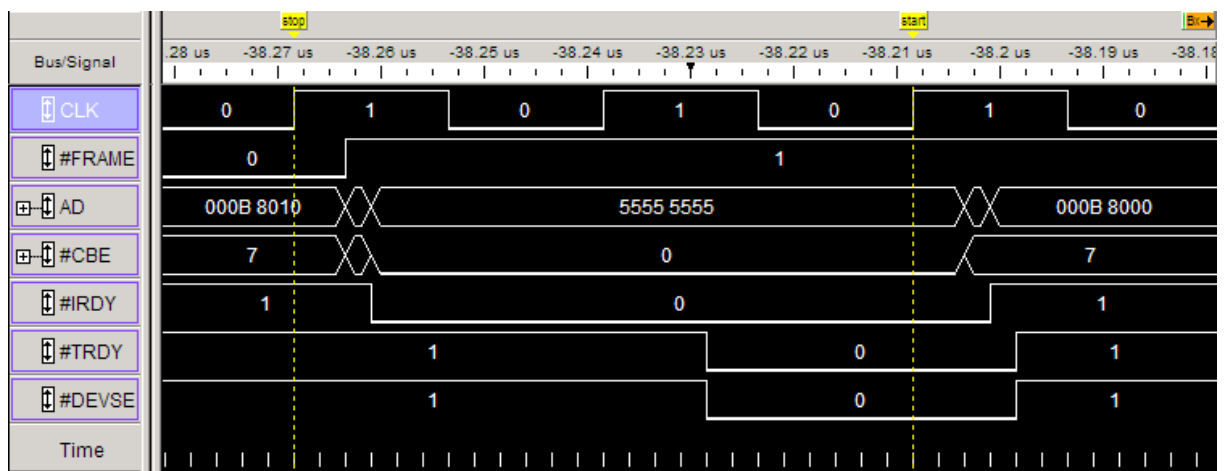
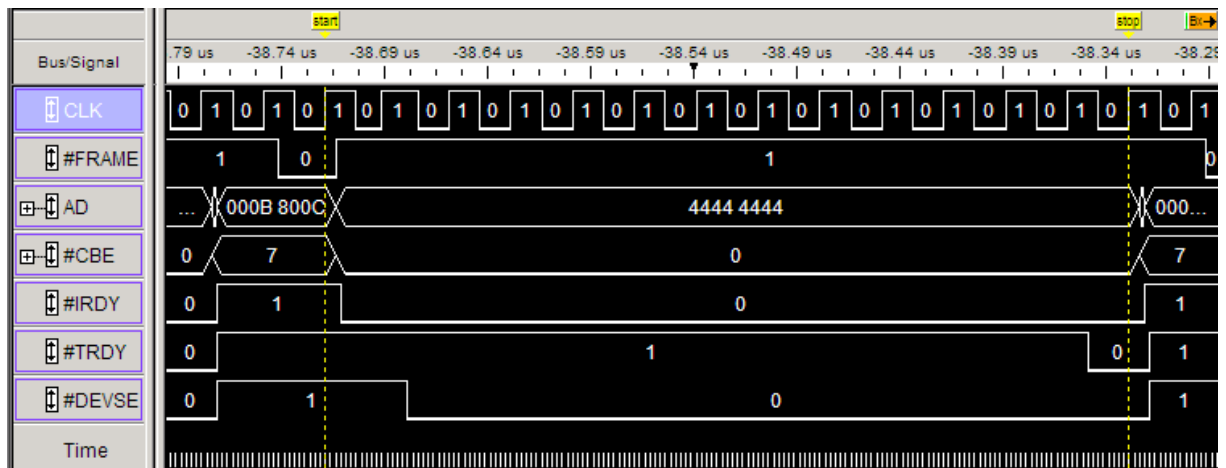
3.1.6 CFG read - čtení konfiguračního registru



Popis: Konfigurační čtení ($\#CBE == A$), Typu 00, načteme 32b ($\#CBE == 0$), tj. hodnotu 1531 10B9h. Tento údaj odpovídá (**Vendor Id:** 0x10B9) **Ali Corporation**, (**Device Id:** 0x1531) **North Bridge**. Operace trvá celkem 5 taktů.

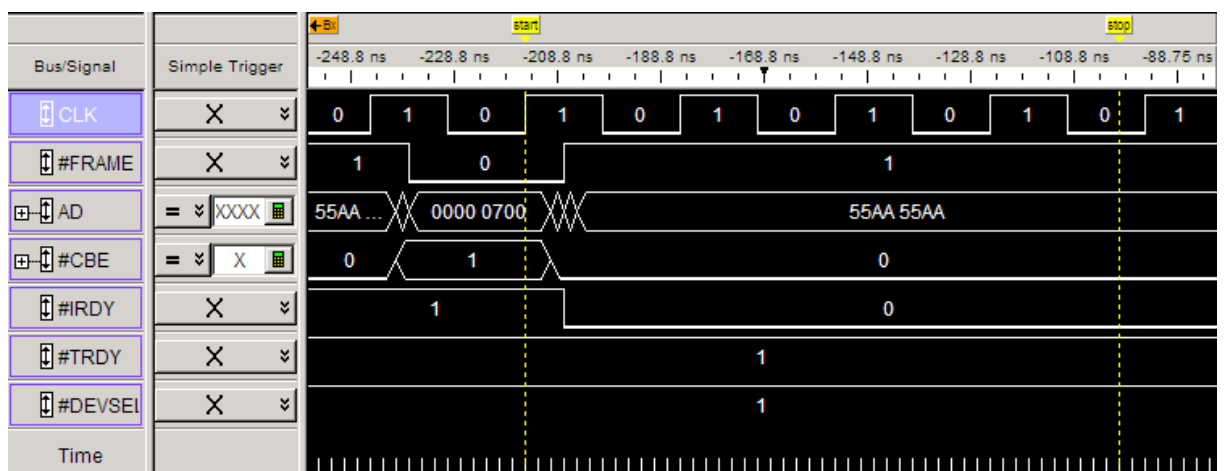
3.1.7 Longwrite - zápis několika po sobě následujících dwordů





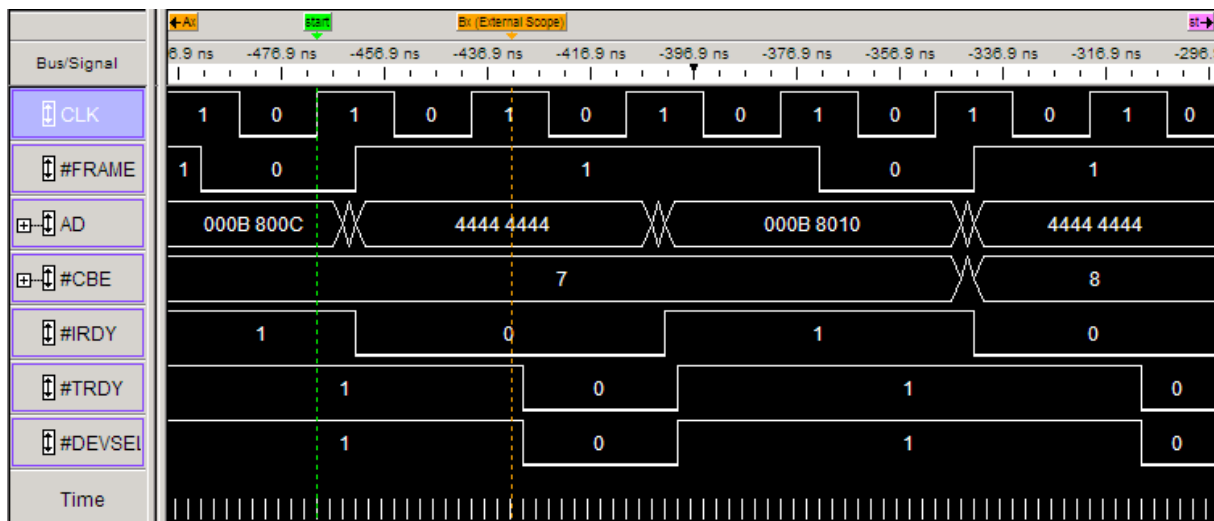
Popis: Cílem bylo zapsat ($\#CBE == 7$) do paměti 5x32b, v ideálním případě by operace trvala 6 taktů. V našem případě se pro každých 32b generovala nový FRAME a adresa, čekalo se na rozpoznání adresy a připravenost zařízení. V tomto případě trval přenos 39 taktů.

3.1.8 Special - vyvolání speciálního cyklu



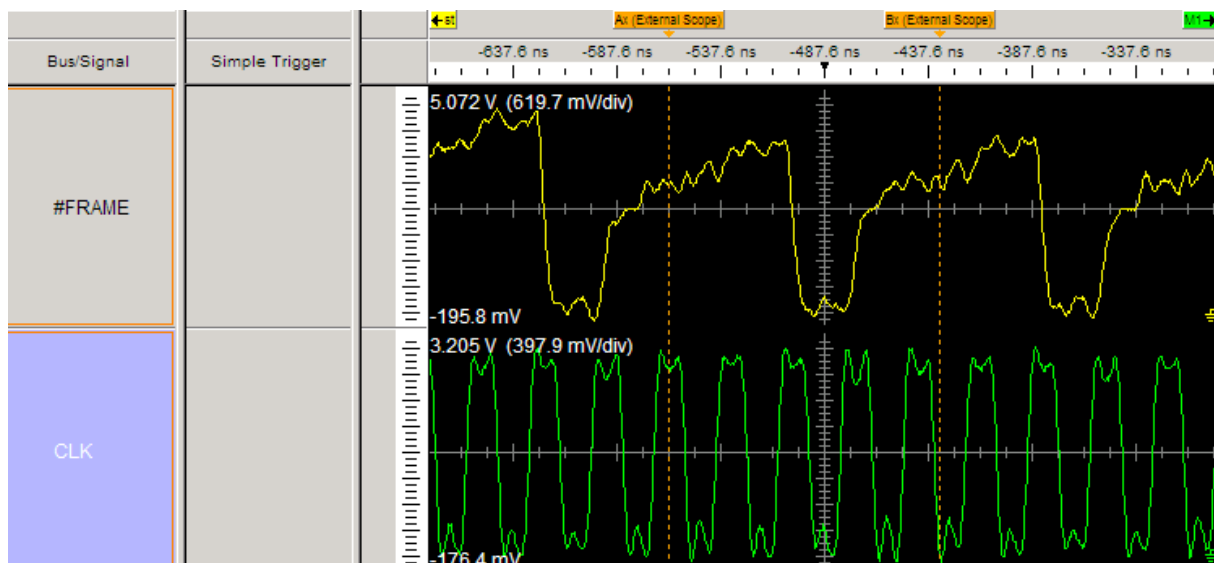
Popis: Speciální cyklus ($\#CBE == 1$), zpráva má hodnotu 55AAh a doplňková informace 55AAh. Všechna zařízení na PCI sběrnici ji přijmou, ale neprovedou žádnou akci (zpráva 55AAh není definována).

3.1.9 Memory write – zápis na liché adresy

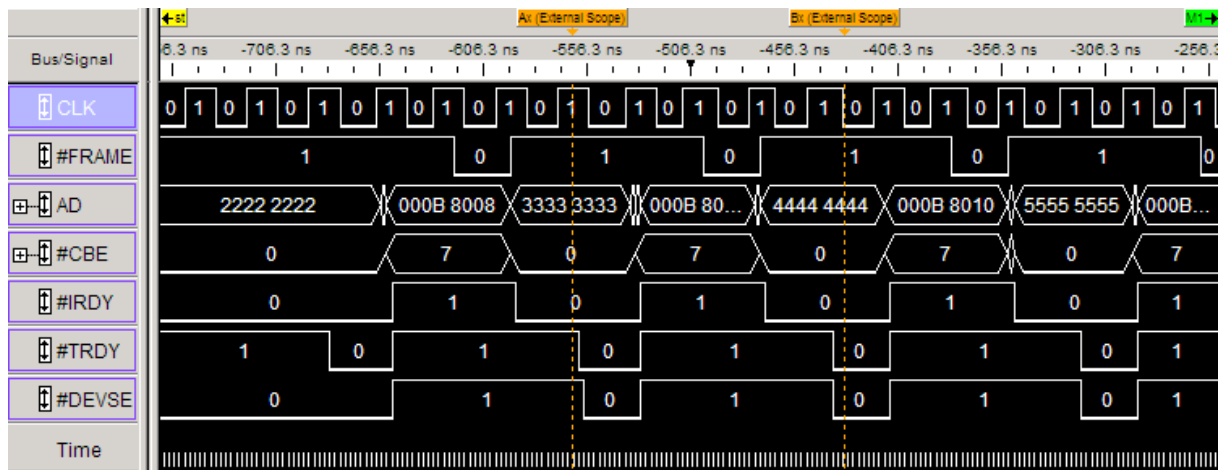


Popis: Zapišeme ($\#CBE == 7$) 32b hodnotu (4444 4444h) na adresu B800Fh. Protože se jedná o adresu 15, tj nesoudělnou 4, provede se operace ve 2 FRAMEch. Nejprve se data uloží na adresu B800Fh ($\#CBE == 7$) a poté na adresy B8010h – B8012h. Zápis trvá 7 taktů.

3.1.10 Spouštění



Skutečný tvar signálů CLK a FRAME.



Popis: Z těchto průběhů je patrné, ve kterých okamžicích spouští logický analyzátor (osciloskop) běh časové základny.

3.2 Zjištění připojených PCI adaptérů

Pro tento úkol jsme využili program dumppci.c a následně dohledali zařízení v databázi www.pcidatabase.com.

3.2.1 Výpis programu dumppci

Device 0... Vendor ID = 0x10b9. Device ID = 0x1531. Revision ID = 0xb2.
Command = 0x6. Status = 0x2400.

Cmd I/O Space : 0. Cmd Memory Space : 1.
Cmd Bus Master : 1. Cmd Special Cycles : 0.
Cmd Mem Write&Inv. : 0. Cmd VGA Palette Snoop : 0.
Cmd Parity Error Detect: 0. Cmd Wait Cycle Control : 0.
Cmd SERR# Enable : 0. Cmd Fast Bk2Bk Ena. : 0.

Status Fast Bk2Bk Capable : 0. Status Data Parity Detected : 0.
Status DEVSEL Timing : 2. Status Signalled Tgt Abrt : 0.
Status Received Tgt Abrt : 0. Status Received Mstr Abrt : 1.
Status Signalled Sys Err : 0. Status Detected Par Error : 0.

Class Code = 06,00,00.

Cache Line Size = 0. Latency Timer = 32. Header Type = 0. BIST = 0.

Base Address 0 = 0. Base Address 1 = 0.

Base Address 2 = 0. Base Address 3 = 0.

Base Address 4 = 0. Base Address 5 = 0.

EPROM Base Address = 0.

Int Line = 0. Int Pin = 0. Min Gnt = 0. Max Lat = 0.

Device 2... Vendor ID = 0x10b9. Device ID = 0x1533. Revision ID = 0xb4.
Command = 0xf. Status = 0x3200.

Cmd I/O Space : 1. Cmd Memory Space : 1.
Cmd Bus Master : 1. Cmd Special Cycles : 1.
Cmd Mem Write&Inv. : 0. Cmd VGA Palette Snoop : 0.
Cmd Parity Error Detect: 0. Cmd Wait Cycle Control : 0.
Cmd SERR# Enable : 0. Cmd Fast Bk2Bk Ena. : 0.

Status Fast Bk2Bk Capable : 0. Status Data Parity Detected : 0.

Status DEVSEL Timing : 1. Status Signalled Tgt Abrt : 0.
Status Received Tgt Abrt : 1. Status Received Mstr Abrt : 1.
Status Signalled Sys Err : 0. Status Detected Par Error : 0.
Class Code = 06,01,00.

Cache Line Size = 0. Latency Timer = 0. Header Type = 0. BIST = 0.
Base Address 0 = 0. Base Address 1 = 0.
Base Address 2 = 0. Base Address 3 = 0.
Base Address 4 = 0. Base Address 5 = 0.
EPROM Base Address = 0.

Int Line = 0. Int Pin = 0. Min Gnt = 0. Max Lat = 0.

Device 4... Vendor ID = 0x5333. Device ID = 0x8a01. Revision ID = 0x1.
Command = 0x7. Status = 0x200.

Cmd I/O Space : 1. Cmd Memory Space : 1.
Cmd Bus Master : 1. Cmd Special Cycles : 0.
Cmd Mem Write&Inv. : 0. Cmd VGA Palette Snoop : 0.
Cmd Parity Error Detect: 0. Cmd Wait Cycle Control : 0.
Cmd SERR# Enable : 0. Cmd Fast Bk2Bk Ena. : 0.
Status Fast Bk2Bk Capable : 0. Status Data Parity Detected : 0.
Status DEVSEL Timing : 1. Status Signalled Tgt Abrt : 0.
Status Received Tgt Abrt : 0. Status Received Mstr Abrt : 0.
Status Signalled Sys Err : 0. Status Detected Par Error : 0.
Class Code = 03,00,00.

Cache Line Size = 0. Latency Timer = 64. Header Type = 0. BIST = 0.
Base Address 0 = 0xe0000000. Base Address 1 = 0.
Base Address 2 = 0. Base Address 3 = 0.
Base Address 4 = 0. Base Address 5 = 0.
EPROM Base Address = 0.

Int Line = 11. Int Pin = 1. Min Gnt = 4. Max Lat = 255.

Device 11... Vendor ID = 0x10b9. Device ID = 0x5229. Revision ID = 0x20.
Command = 0x5. Status = 0x280.

Cmd I/O Space : 1. Cmd Memory Space : 0.
Cmd Bus Master : 1. Cmd Special Cycles : 0.
Cmd Mem Write&Inv. : 0. Cmd VGA Palette Snoop : 0.
Cmd Parity Error Detect: 0. Cmd Wait Cycle Control : 0.
Cmd SERR# Enable : 0. Cmd Fast Bk2Bk Ena. : 0.
Status Fast Bk2Bk Capable : 1. Status Data Parity Detected : 0.
Status DEVSEL Timing : 1. Status Signalled Tgt Abrt : 0.
Status Received Tgt Abrt : 0. Status Received Mstr Abrt : 0.
Status Signalled Sys Err : 0. Status Detected Par Error : 0.
Class Code = 01,01,fa.

Cache Line Size = 0. Latency Timer = 64. Header Type = 0. BIST = 0.
Base Address 0 = 0. Base Address 1 = 0.
Base Address 2 = 0. Base Address 3 = 0.
Base Address 4 = 0xf001. Base Address 5 = 0.
EPROM Base Address = 0.

Int Line = 0. Int Pin = 1. Min Gnt = 2. Max Lat = 4.

Nalezli jsme 4 zařízení připojená k PCI sběrnici, výrobce a typ je uveden v následující tabulce:

Číslo zařízení	Vendor ID	Device ID	Výrobce	Název
Device 0	0x10b9	0x1531	Ali Corporation	North Bridge
Device 2	0x10b9	0x1533	Ali Corporation	PCI South Bridge
Device 4	0x5333	0x8a01	S3 Graphics Co., Ltd	ViRGE /DX & /GX
Device 11	0x10b9	0x5229	Ali Corporation	EIDE Controller

4 Závěr

Měření a následné zpracování bylo díky modernímu logickému analyzátoru velice komfortní a přesné. Pro další ročníky by bylo jistě zajímavé „vyhrabat“ o něco menší trosku než bylo měřené PC se sběrnici PCI. Tato sběrnice neumožňovala přenos několika po sobě jdoucích dworků a pro každý generovala nový FRAME. Ostatní operace se nám podařilo nasimulovat bez problémů. Pro zápis do paměti na liché adresy jsme upravili zdrojový soubor. U speciálního cyklu jsme přepsáním zprávy z hodnoty 55AA 55AAh na 0h docílili vypnutí počítače. Bohužel se nám při této operaci nepodařilo zachytit průběhy na sběrnici. Dále je velice zajímavé sledovat skutečné průběhy signálů. Oproti teoretickému (obdélníkovému) průběhu jsou značně deformovány.