

# Principy počítačů a operačních systémů

Systemová architektura,  
připojení a komunikace s externími zařízeními

Zimní semestr 2011/2012

# Uspořádání na úrovni systému

## Funkční bloky

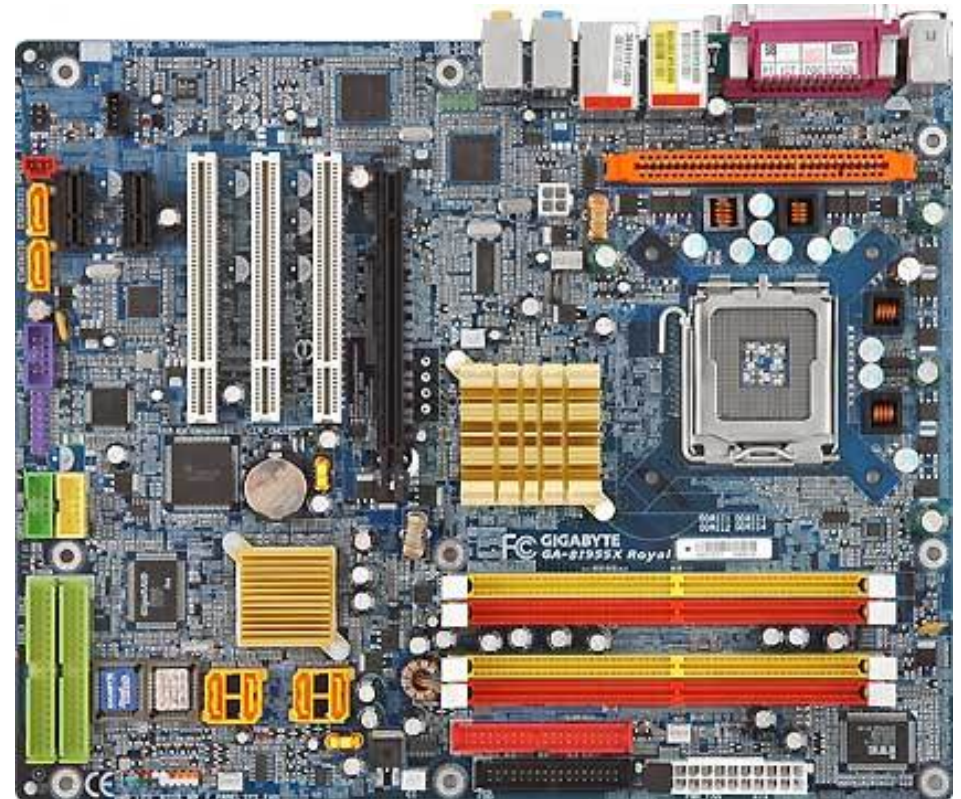
- procesory, paměti
- *periferní zařízení*

## Propojovací systém

- přenos dat v rámci systému a mimo něj

## Podpůrné obvody

- přidělování/řízení přístupu k prostředkům
- podpora přerušení a přímého přístupu do paměti



Zdroj: [www.tomshardware.com](http://www.tomshardware.com)



# Periferní zařízení

K čemu je, kdo je na druhé straně, kolik dat přenáší

Zařízení	Použití	Partner	Datový tok (Mbit/s)
klávesnice	vstup	člověk	0.0001
myš	vstup	člověk	0.0038
hlasový vstup	vstup	člověk	0.2640
zvukový vstup	vstup	stroj	0.0882-18.4320
scanner	vstup	člověk	3.2000
hlasový výstup	výstup	člověk	0.2640
zvukový výstup	výstup	člověk	1.4112-18.4320
laserová tiskárna	výstup	člověk	3.2000
obrazový výstup	výstup	člověk	800.0000-8000.0000
modem	vstup/výstup	stroj	0.0160-0.0640
lokální síť	vstup/výstup	stroj	10.0000-10000.0000
bezdrátová síť	vstup/výstup	stroj	1.0000-54.0000
optický disk	úložiště	stroj	80.0000
magnetická páska	úložiště	stroj	32.0000
magnetický disk	úložiště	stroj	240.0000-2560.0000



# Připojení/propojení periferií

---

## Propojovací systém

- návrh ovlivněn mnoha faktory
  - ♦ rozšiřitelnost, odolnost, výkonnost
- určuje architekturu systému

## Výkonnost propojovacího systému

- propustnost/šířka pásma (bandwidth)
  - ♦ množství dat přenesených za jednotku času
  - ♦ primárně mnoho “velkých” V/V operací
- zpoždění/latence
  - ♦ množství V/V operací za jednotku času
  - ♦ primárně mnoho “menších” V/V operací



# Propojovací systémy v počítači

---

## Dvoubodové spoje

- přímé spojení (port)
  - ♦ není nutné adresovat
- křížový přepínač (crossbar switch)
- propojovací síť (switch fabric)
  - ♦ blokující, neblokující, ...

## Vícebodové spoje

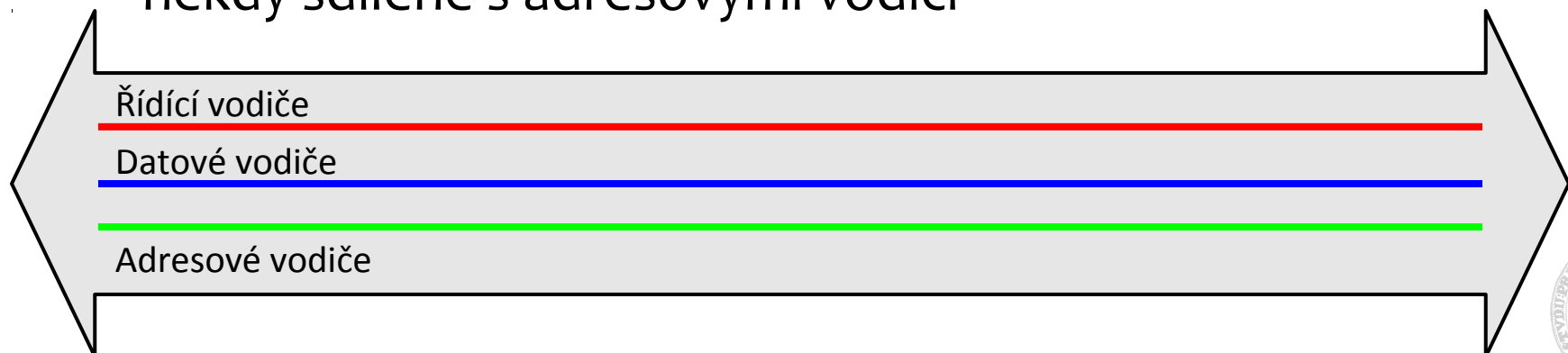
- účastníci sdílejí přenosové médium
  - ♦ broadcast: 1 vysílač, více příjemců
- **sběrnice (bus)**



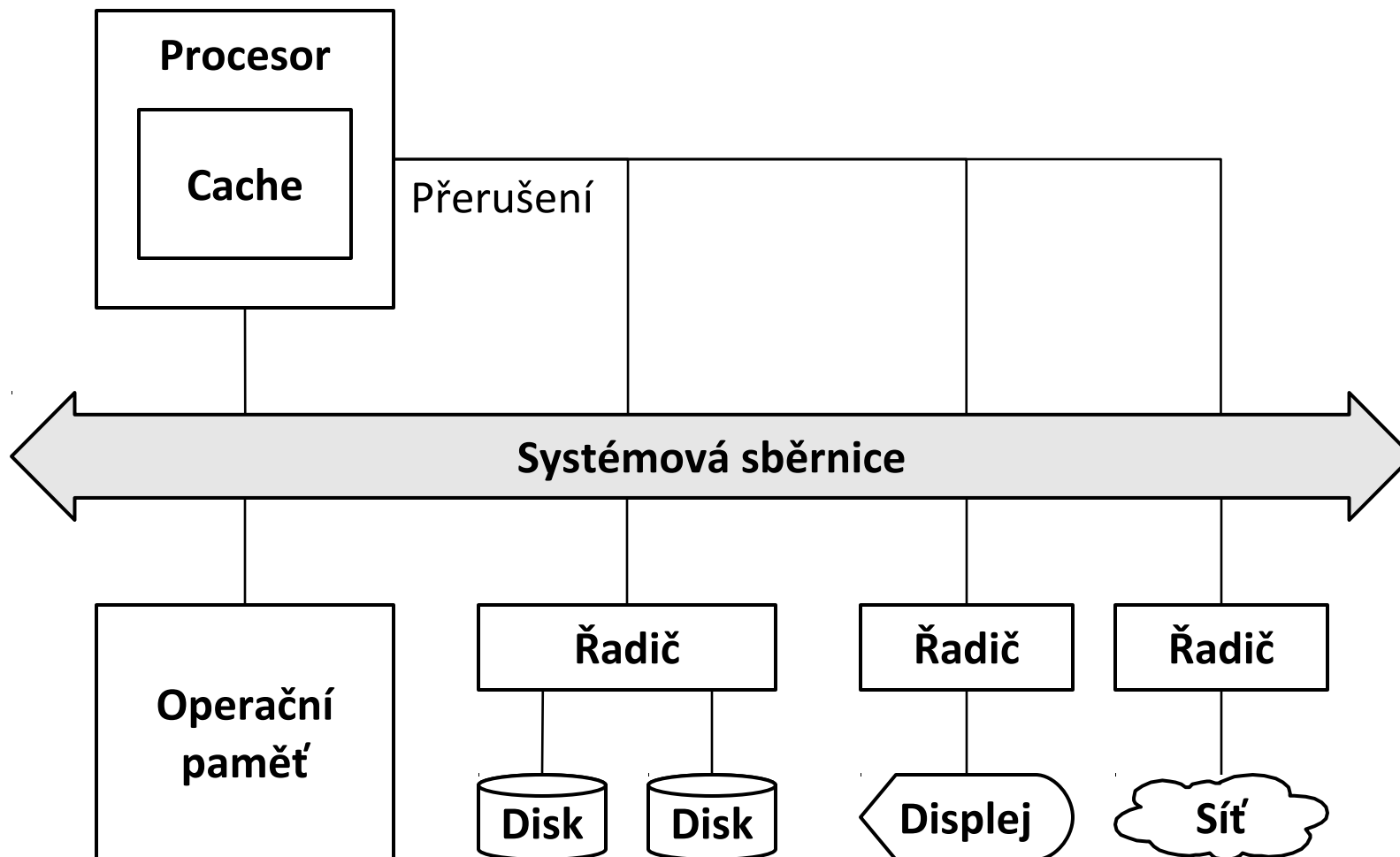
# Co je to sběrnice?

## Sada vodičů propojující více zařízení

- vodiče pro řídicí signály
  - ♦ řízení komunikace – požadavky/potvrzení
  - ♦ indikace typu dat na datových vodičích
- adresové vodiče
  - ♦ výběr cílového zařízení pro komunikaci
- vodiče pro datové signály
  - ♦ přenos dat mezi zdrojovým a cílovým zařízením
  - ♦ někdy sdílené s adresovými vodiči



# Typická (sběrnicevá) architektura systému



# Proč použít/nepoužít sběrnici?

---

## Hlavní výhody sběrnice

- flexibilita (a kompatibilita)
  - ♦ jednoduché připojení nejrůznějších periferií
  - ♦ přenos periferií mezi (kompatibilními) systémy
- nízká cena
  - ♦ jedna sdílená sada vodičů pro mnoho zařízení

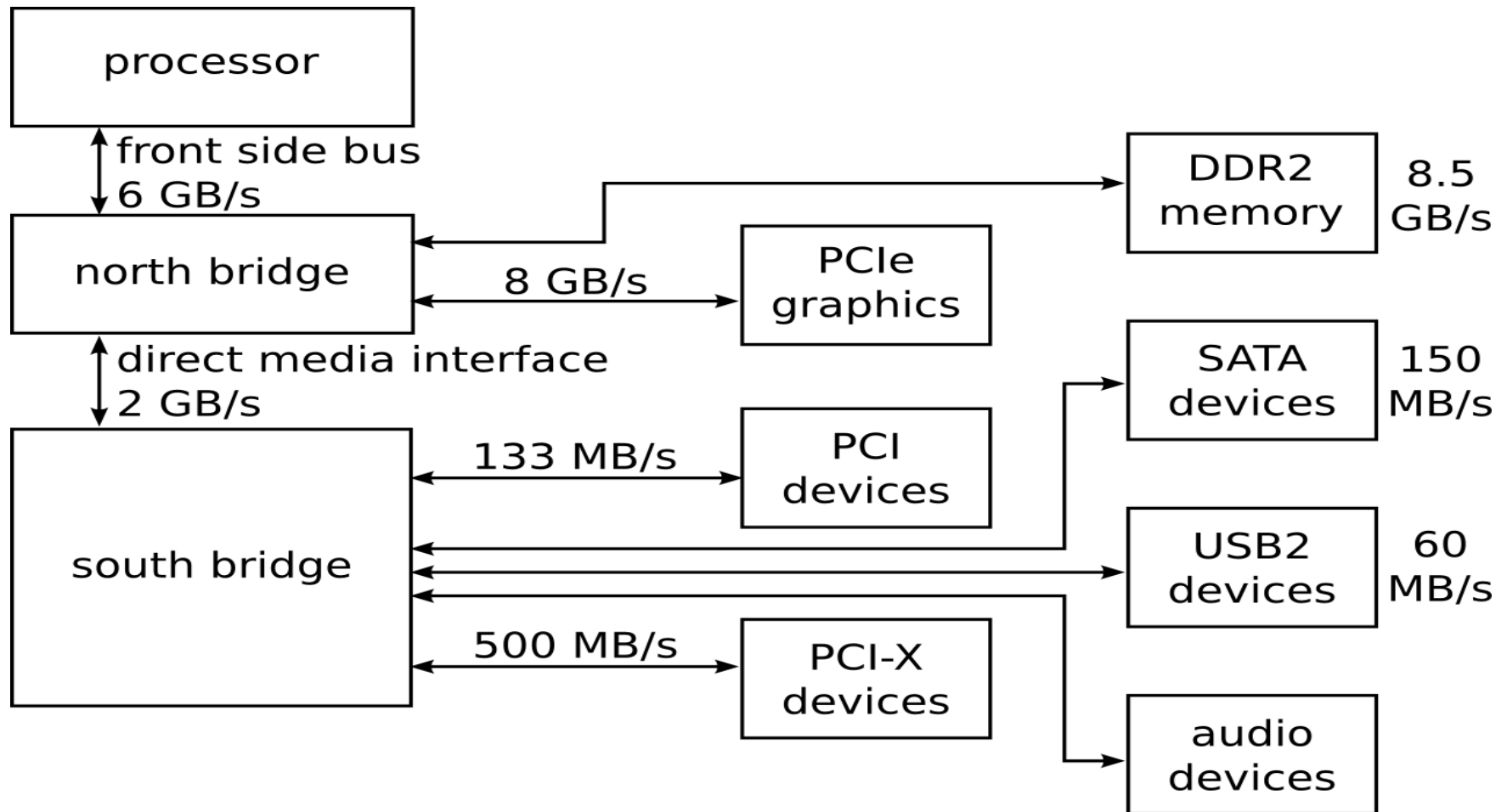
## Hlavní nevýhody sběrnice

- potenciální úzké hrdlo v systému
- omezení délkou sběrnice a počtem zařízení
- omezení v důsledku rozmanitosti zařízení
  - ♦ odlišné rychlosti, zpoždění a objemy přenášených dat





# System sběrnic v moderním PC



# Komunikace na sběrnici

---

## Transakce na sběrnici

- požadavek
  - ♦ vyslání příkazu a adresy cílového zařízení
  - ♦ odezva cílového zařízení potvrzující požadavek
- přenos dat
  - ♦ směr přenosu určen příkazem

## Účastníci (komunikace)

- **master/initiator** zahajuje sběrniceovou transakci
  - ♦ tj. určuje “co a s kým”
- **slave/target** posílá/přijímá data



# Logická charakteristika sběrnice

---

## Sběrníkový protokol

- definuje povolený průběh transakcí na sběrnici
  - ♦ logický význam signálů na vodičích
  - ♦ popis pomocí stavových a časových diagramů

## Způsob řízení a přidělování sběrnice

- synchronní vs. asynchronní řízení
  - ♦ přechody mezi fázemi transakce určeny hodinovým signálem resp. změnou řídicích signálů
- centralizované vs. distribuované přidělování
  - ♦ o přidělení sběrnice rozhoduje centrální arbitr resp. zařízení v rámci distribuovaného protokolu



# Synchronní sběrnice

---

## Obecná charakteristika

- řídicí vodiče zahrnují hodinový signál
- pevně stanovený komunikační protokol
  - ♦ přechody mezi fázemi transakce určeny hodinovým signálem resp. změnou řídicích signálů

## Hlavní výhody

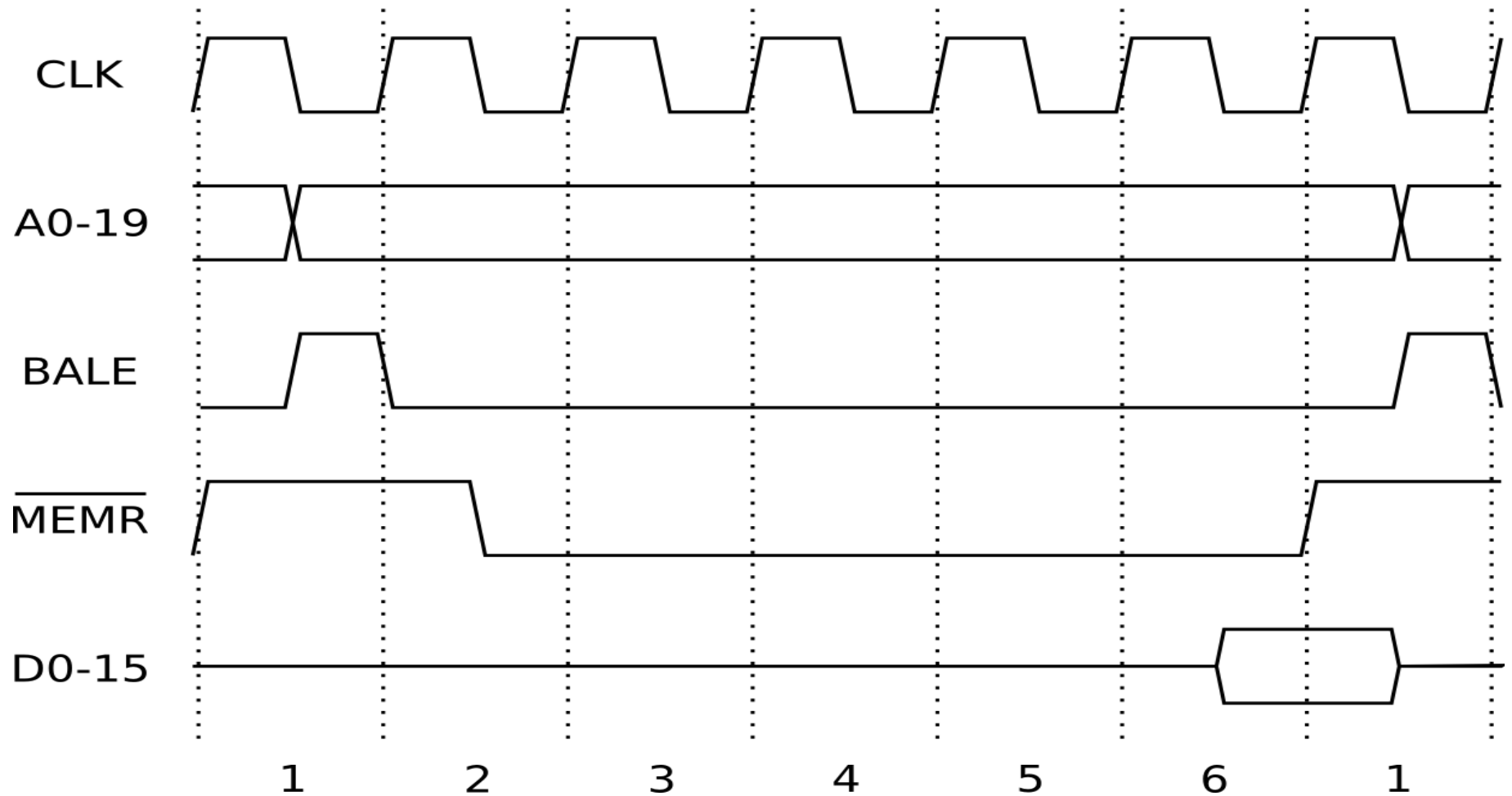
- jednodušší řízení  $\Rightarrow$  vysoká rychlost

## Hlavní nevýhody

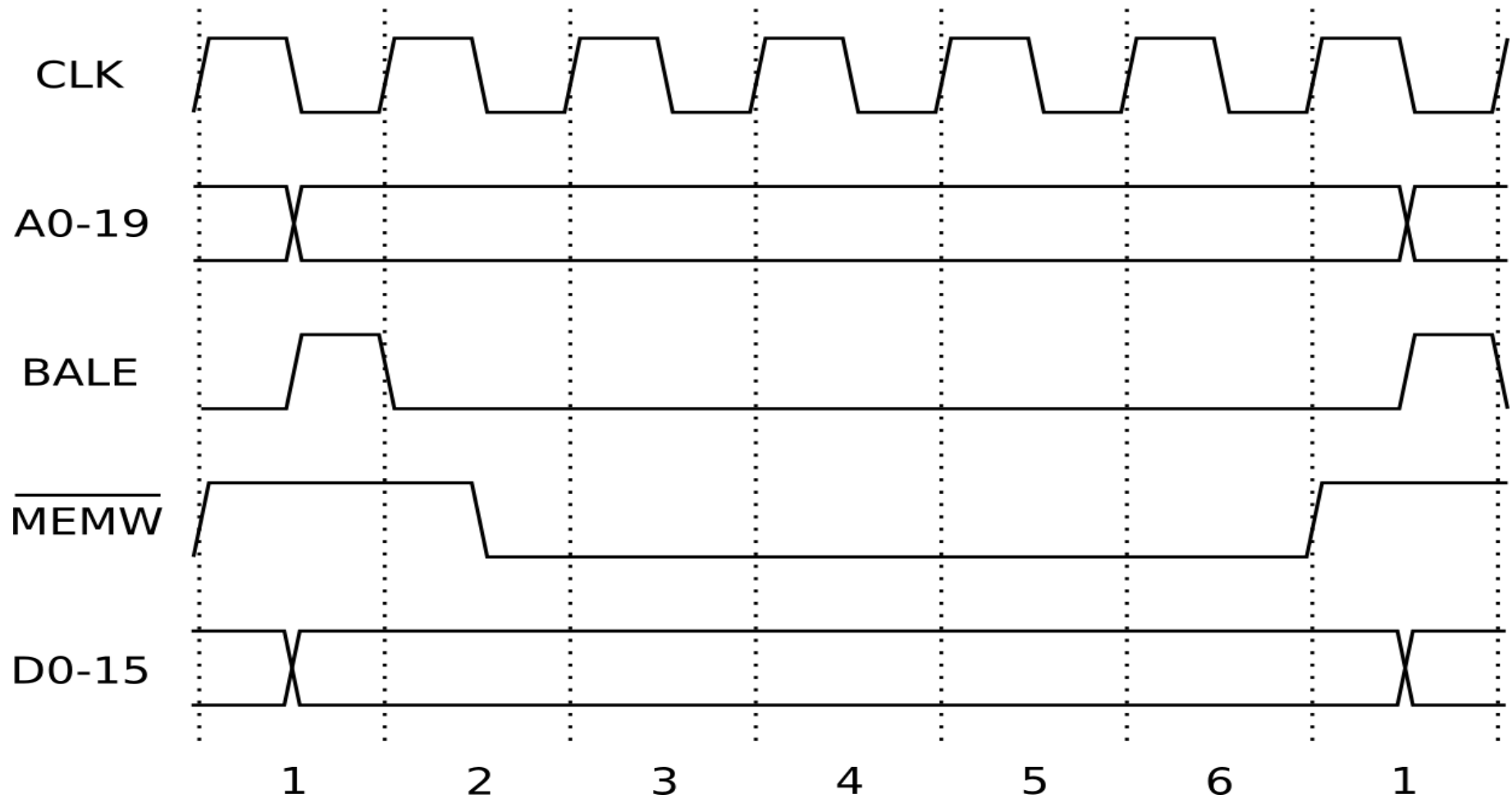
- zpoždění hodin (clock skew)  $\Rightarrow$  omezená délka
- hodinový signál stejně rychlý pro všechna zařízení



# Příklad: čtení z paměti na synchronní sběrnici ISA



# Příklad: zápis do paměti na synchronní sběrnici ISA



# Asynchronní sběrnice

---

## Obecná charakteristika

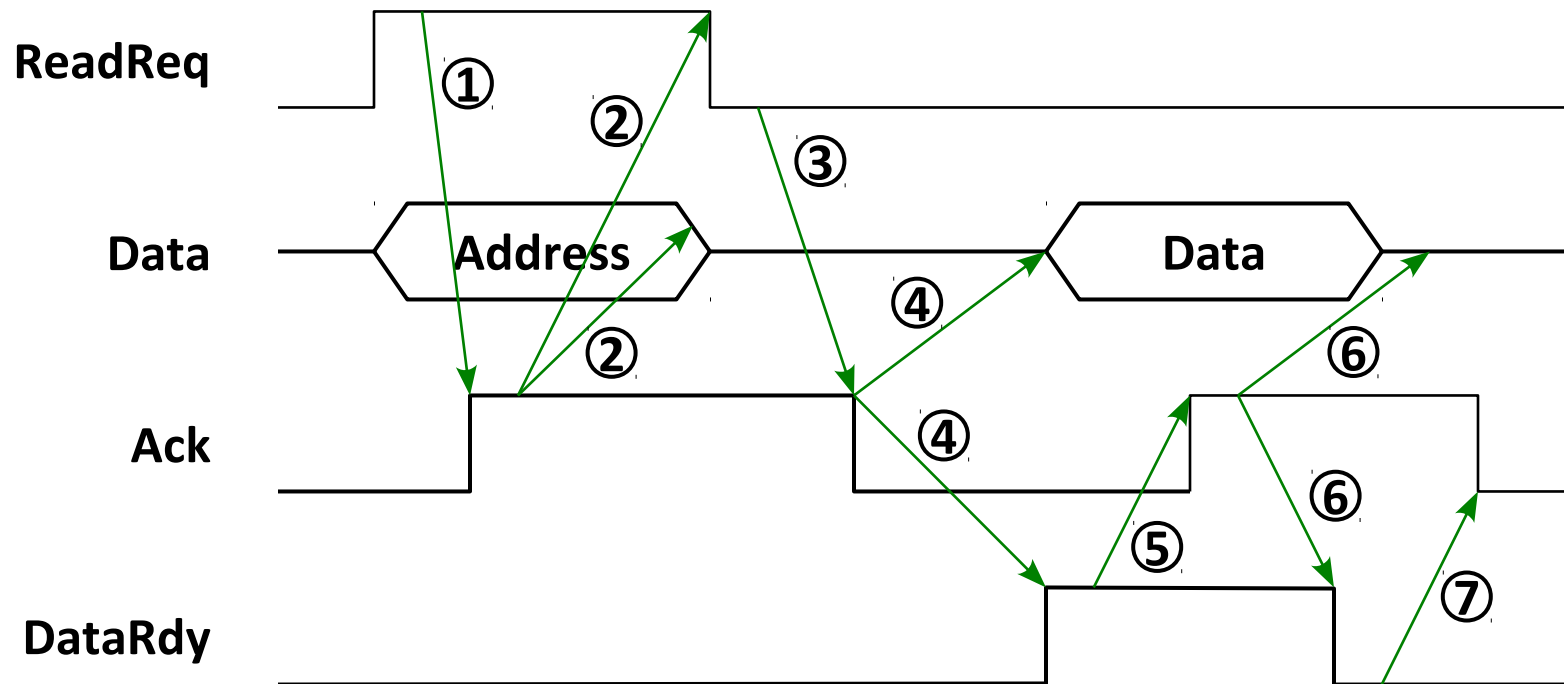
- není řízena hodinovým signálem
- umožňuje připojit zařízení o různé rychlosti
- není tolik omezena délkou (nemá hodiny)
- vyžaduje handshaking protokol
- složitější řízení, větší délka  $\Rightarrow$  nižší rychlosti

## Typické použití

- externí V/V sběrnice (USB 2.0, FireWire)



# Příklad: čtení dat na asynchronní sběrnici





# Přidělování sběrnice

---

## Základní princip

- master žádá a čeká na přidělení sběrnice
- arbitr přiděluje sběrnici žadatelům
  - ♦ při souběžných žádostech zohledňuje prioritu a fairness
- master informuje arbitra o uvolnění sběrnice

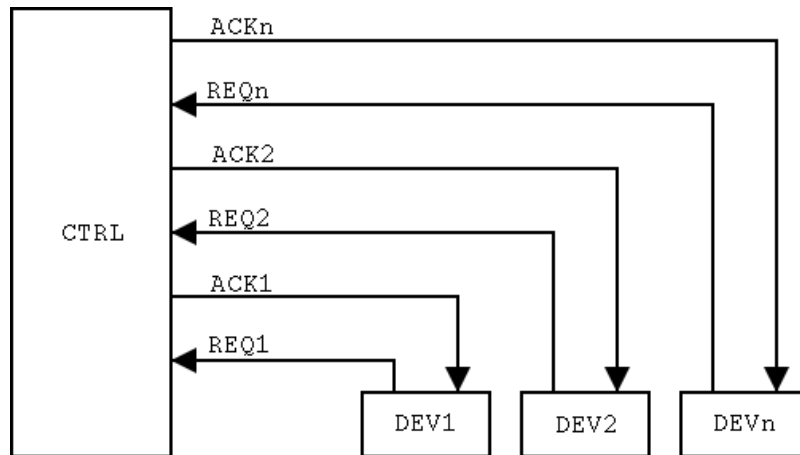
## Přidělovací mechanismy

- centralizované přidělování
- distribuované kolizní přidělování
- distribuované založené na “samovýběru”

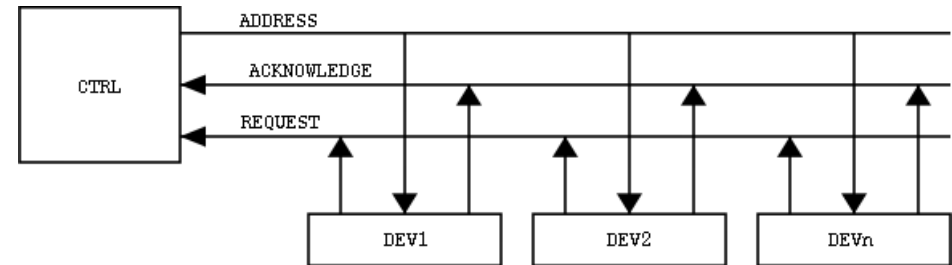


# Centralizované přidělování

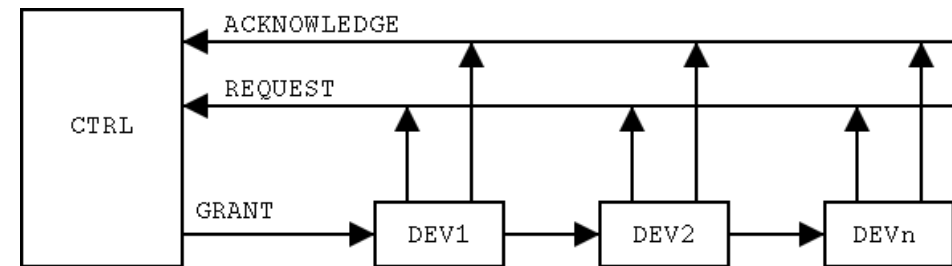
## Nezávislé žádosti



## Cyklické výzvy (polling)



## Prioritní zřetězení (daisy chain)



# Ovládání periferních zařízení

---

## Periferie ovládány programem

- kroky nutné k vykonání požadované operace
  - ♦ výběr zařízení
  - ♦ zápis příkazů k vykonání operace (specifických pro zařízení)
  - ♦ čtení stavové informace specifické pro zařízení

## Rozhraní procesor/periférie

- sdílený adresový prostor zařízení a paměti
  - ♦ zařízení mapována do paměti (memory-mapped I/O)
  - ♦ přístup pomocí běžných instrukcí typu load/store
- oddělený adresový prostor zařízení a paměti
  - ♦ přístup pomocí speciálních instrukcí typu in/out



# Komunikace procesoru se zařízením

---

## Informace o stavu zařízení

- polling
  - ♦ program periodicky kontroluje stav zařízení
  - ♦ značná režeie v případě pomalých zařízení
- interrupt-driven I/O
  - ♦ zařízení signalizuje procesoru změnu stavu
  - ♦ obslužná rutina přerušení reaguje na stav zařízení

## Přenos dat z/do zařízení (z/do paměti)

- s účastí procesoru (programmed I/O)
- bez účasti procesoru (DMA, bus mastering)



# Podpora interrupt-driven I/O

---

## Procesor musí podporovat

- signalizaci požadavku na přerušení
  - ♦ jeden nebo více vstupních signálů
- identifikaci zdroje přerušení
  - ♦ pevná identifikace odvozená od signálových vodičů
  - ♦ dynamická identifikace
- výběr obslužné rutiny přerušení
  - ♦ statická adresa vs. vektor přerušení

## System musí podporovat

- doručení přerušovacích signálů od periferie procesoru
- řadič přerušení – zdrojů více než má CPU vstupů
  - ♦ více přerušovacích signálů, vyhodnocení priority



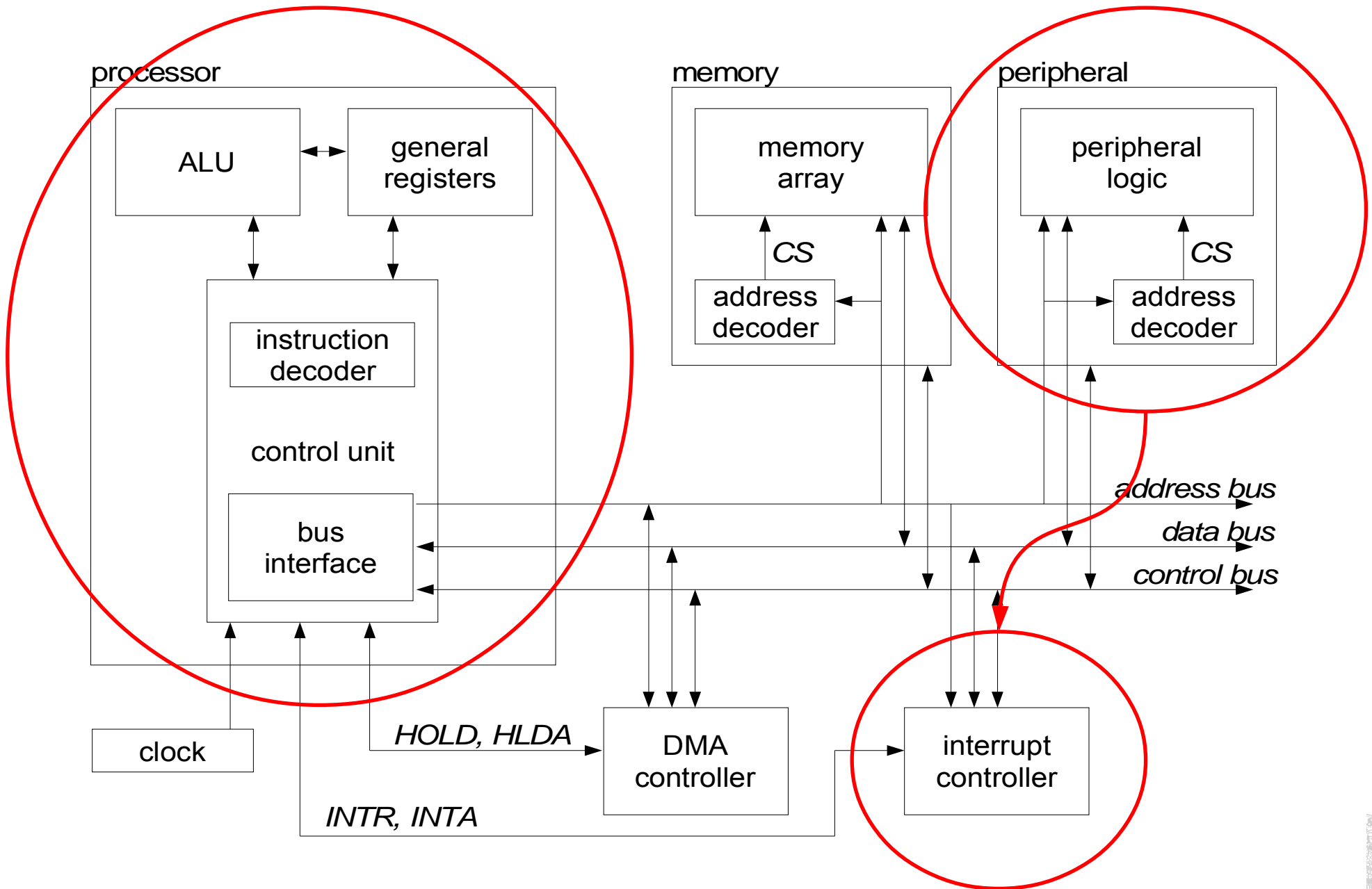
# Zpracování přerušení na procesoru MIPS

## Vyvolání a obsluha přerušení

1. Periferie aktivuje signál INTx
2. Procesor nastaví registry Cause a EPC a skočí na obsluhu General Exception na adrese 0x80000180
3. Obslužná rutina OS pro GE uschová stav procesoru, zjistí příčinu a vyvolá specifickou obslužnou rutinu
  - ♦ pokud bylo přerušení požadováno více zdroji, jsou tyto obslouženy v pořadí určeném obslužnou rutinou
4. Obslužná rutina OS obnoví stav procesoru a skočí na adresu v EPC – procesor pokračuje ve vykonávání přerušeného programu

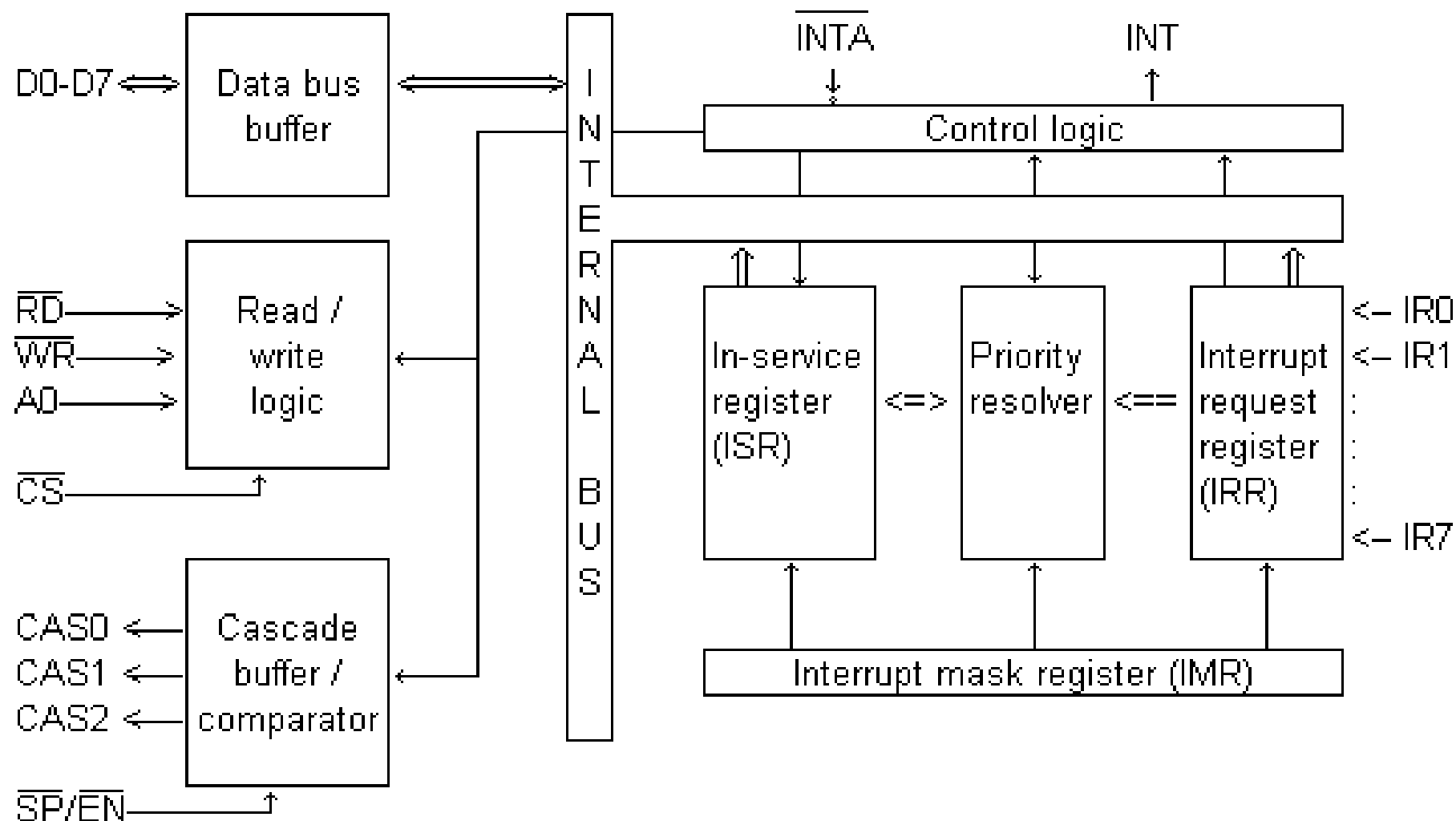


# System s řadičem přerušení



# Příklad řadiče přerušení (i8259)

8259 internal block diagram





# Zpracování přerušení s využitím i8259

## Vyvolání a obsluha přerušení

1. Aktivován signál IRx, nastaví bit v IRR
2. Řadič signalizuje přerušeni procesoru pomocí INT
3. CPU potvrdí příjem INT pomocí #INTA
  - ♦ Řadič nastaví v ISR bit s nejvyšší prioritou, vynuluje odpovídající bit v IRR
4. CPU podruhé signalizuje #INTA
  - ♦ Řadič vyšle na sběrnici číslo vektoru přerušeni
5. CPU zavolá obslužnou rutinu odpovídající vektoru
6. Ostatní přerušeni blokována, dokud řadič neobdrží příkaz “Non-Specific EOI”



# Přenos dat řízený programem (programmed IO)

---

Přenos iniciován při změně stavu zařízení

- polling, interrupt-driven I/O

Přenos realizován cyklem v programu

- přečíst slovo z periferie/paměti do registru
- zapsat slovo do paměti/periferie z registru
- posun na následující slovo v paměti
- opakovat pro požadovaný počet slov

Rychlý přenos, ale zatěžuje procesor

- procesor by měl dělat něco užitečnějšího



# Přímý přístup do paměti (*direct memory access*)

---

Přenos iniciován při změně stavu zařízení

- polling, interrupt-driven I/O

Přenos realizován řadičem nebo zařízením

- program musí pouze nastavit parametry přenosu

Přenos zařízení/paměť

- bus mastering, řadič DMA

Přenos zařízení/zařízení

- bus mastering

Nezatěžuje procesor, vyšší latence

- nevhodné pro malé objemy dat



# Bus mastering

---

## Sběrnici může řídit libovolný účastník

- nutno zažádat o přidělení sběrnice (arbitrace)
  - ♦ procesor v roli normálního účastníka

## Přenos dat mezi libovolnými účastníky

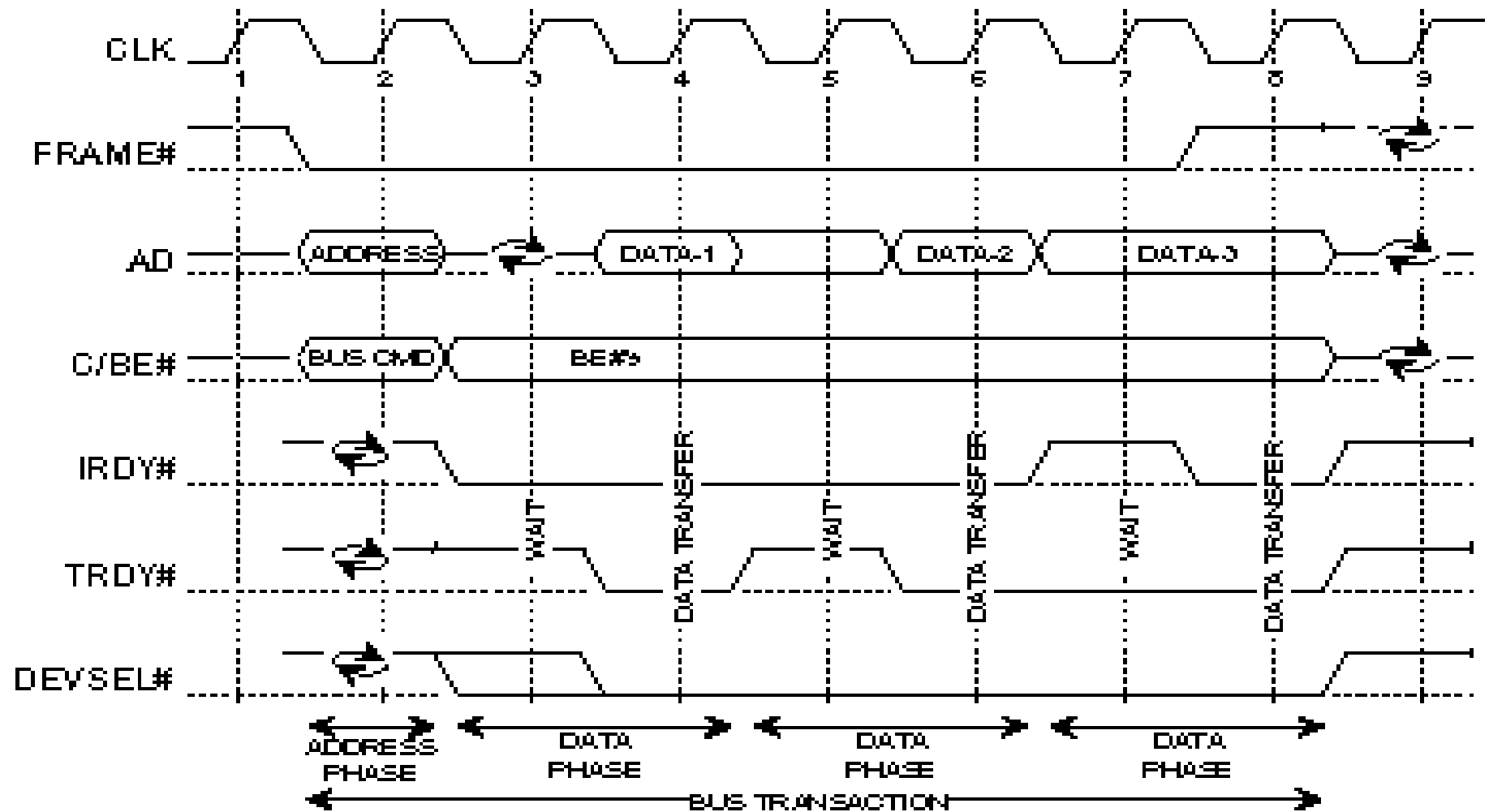
- paměť/periferie, periferie/periferie
- stále je nutné přenos nastavit z programu

## Efektivní přenos dat

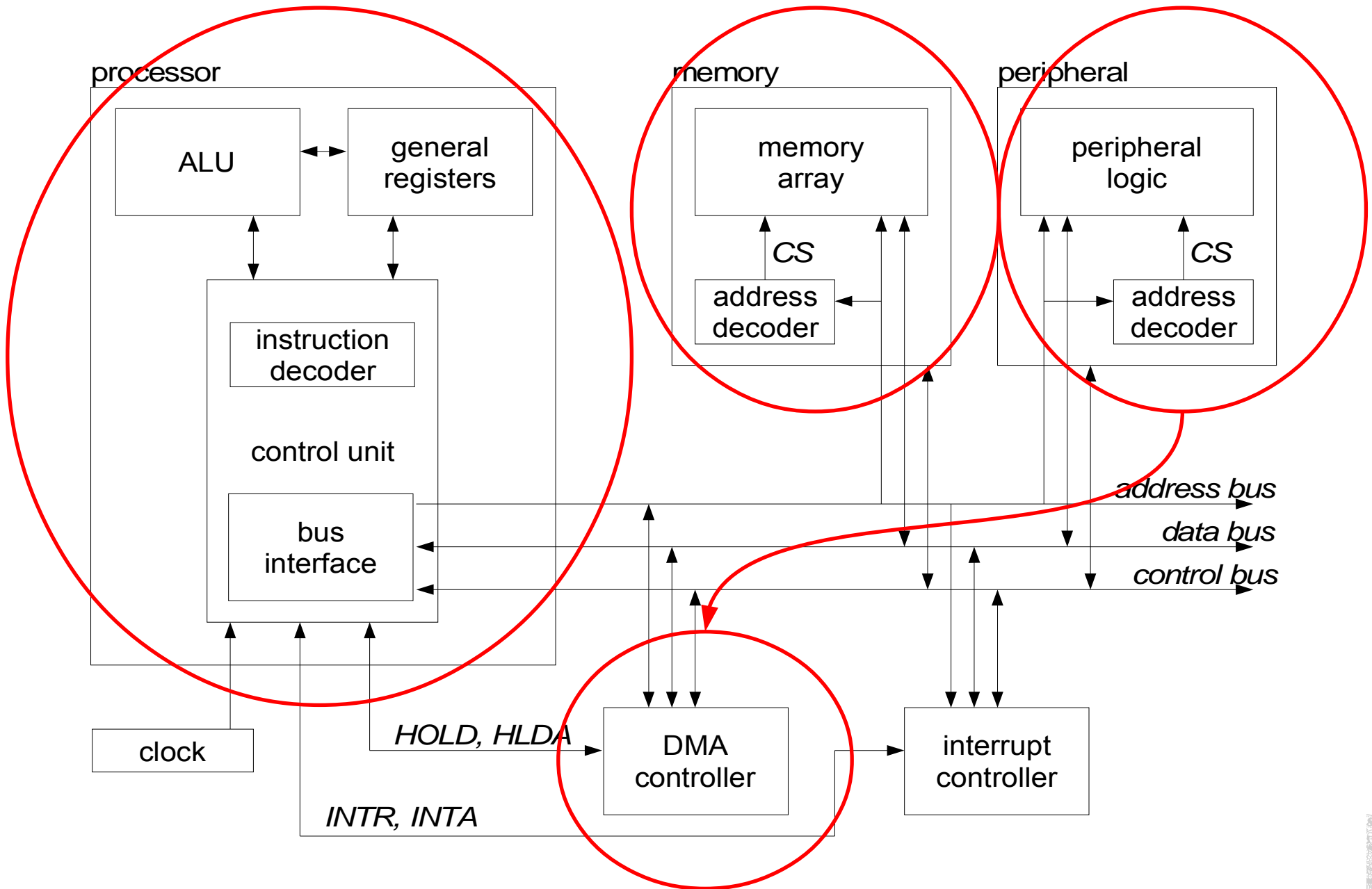
- dávkový režim (burst mode)
  - ♦ 1 adresový cyklus na blok dat
- přenos nesouvislých bloků (scatter/gather)



# Příklad: burst transakce na sběrnici PCI



# System s řadičem DMA

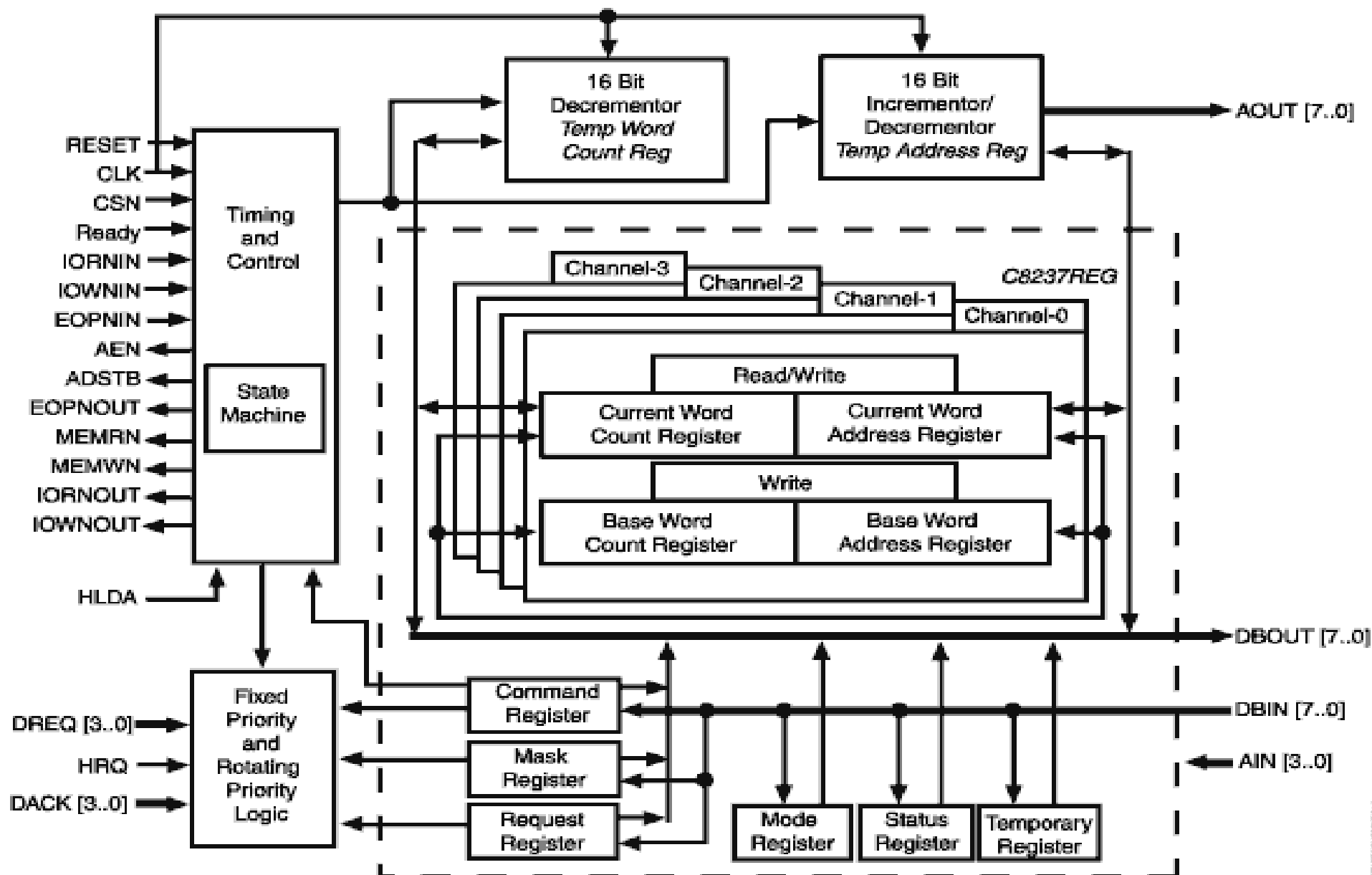


## Obvod pro řízení přenosů na sběrnici

- generuje adresy paměti a periferie, generuje řídicí signály pro čtení/zápis
  - ♦ při programmed I/O toto dělá procesor
- generuje signály pro procesor, aby zajistil, že procesor nepřistupuje (nezapisuje) na sběrnici
- řadič sám se chová jako periferie
  - ♦ program nastavuje parametry přenosu, tj. odkud se bude přenášet, kam, a kolik (2 čítače, kanál DMA)
  - ♦ zařízení připojena na kanál DMA, při přenosu je cílové zařízení aktivováno řadičem, nikoliv vystavením adresy



# Příklad řadiče DMA (i8237)





# Přenos bloku dat pomocí DMA (i8237) (1)

---

## Posloupnost událostí

1. program nastaví řadič a periférii a povolí přenos
2. aktivací signálu DREQx periférie požádá řadič DMA o přenos slova z/do paměti
3. řadič DMA zkontroluje nastavení kanálu vyhodnotí prioritu žádosti
4. aktivací signálu HOLD řadič DMA požádá CPU o přidělení sběrnice
5. pokud CPU nepotřebuje sběrnici, odpojí se od sběrnice a signalizuje HLDA
  - ♦ CPU testuje HOLD na začátku strojového cyklu



# Přenos bloku dat pomocí DMA (i8237) (2)

## Posloupnost událostí

- po přijetí HLDA řadič připraví sběrnici pro přenos
  - ♦ vystaví adresu v paměti a řídicí signály pro čtení/zápis z/do paměti/periferie
- řadič DMA aktivuje signál DACKx, kterým vyzve periferii k vystavení/přečtení dat na/ze sběrnice
- v závislosti na režimu buď přenos končí, nebo pokračuje dalším slovem dokud je DREQx aktivní
- při posledním slově řadič aktivuje signál EOP
- při ukončení přenosu řadič uvolní signál HOLD
- procesor uvolní HLDA a připojí se ke sběrnici



# Na co dát pozor u DMA/bus masteringu?

---

Přenos pomocí DMA není nutně rychlejší než PIO

- umožňuje procesoru dělat něco užitečnějšího

System musí zajistit platnost dat v cache

- write-back cache může mít novější data než paměť
- v paměti mohou být novější data než v cache
- procesor sleduje provoz z/do paměti



# Sběrnice pro připojování periférií

---

## Historické

- PC Bus
  - ♦ 8bit datová, 20bit adresová sběrnice, 8MHz
- ISA – Industry Standard Architecture (AT Bus)
  - ♦ 16bit datová, 24bit adresová sběrnice, 8MHz
- MCA – MicroChannel Architecture
  - ♦ 16/32/64bit data, 24/32bit adresy, 10MHz
- EISA – Extended Industry Standard Architecture
  - ♦ 32bit data, 32bit adresy, **8MHz**, programová konfigurace
- VL Bus – VESA Local Bus
  - ♦ 32bit data, 32bit adresy, až 50MHz, na syst. sběrnici



# Sběrnice pro připojování periferií

---

## PCI – Peripheral Component Interconnect

- připojení k systémové sběrnici přes můstek
- synchronní, multiplex dat a adres
- 32bit data, 32/64bit adresy
- 1992 v1.0: 33MHz, 1995 v2.1: 66MHz
- podpora dávkových přenosů (burst mode)

## PCI-X

- rozšíření na 64bit data, podpora ECC
- v1.0: 66 a 133MHz, v2.0: 266 a 533 MHz
- zpětná kompatibilita HW i SW



# Sběrnice pro připojování periferií

---

## AGP – Advanced Graphics Port

- dvoubodový spoj
- 4x frekvence PCI, tj. 133MHz
- pro přenos dat využita náběžná i sestupná hrana
- pipelining



# Sběrnice pro připojování periferií

---

## PCI Express (PCIe)

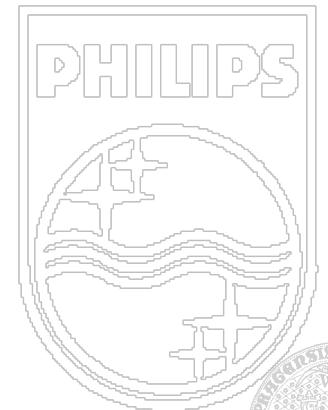
- sériová, paketový protokol
- 256 MB/s (PCIe x1) – 8 GB/s (PCIe x16 duplex)
  - ♦ pro srovnání PCI/32bit/33MHz - 133MB/s
- QoS, power management, hot-plug...



# Sběrnice pro komunikaci mezi IO

## I<sup>2</sup>C – Inter IC

- dvou vodičová (SDA, SCL) sériová sběrnice
  - ♦ v klidovém stavu udržováno kladné napětí (pull-up)
- přenos dat po bajtech, potvrzení příjmu
- rychlosti 0- 100/400/1000/3400 kbit/s
- master/slave řízení
- možnost multi-master konfigurace
  - ♦ synchronizace hodin + arbitrace
- [www.philipslogic.com/i2c](http://www.philipslogic.com/i2c)

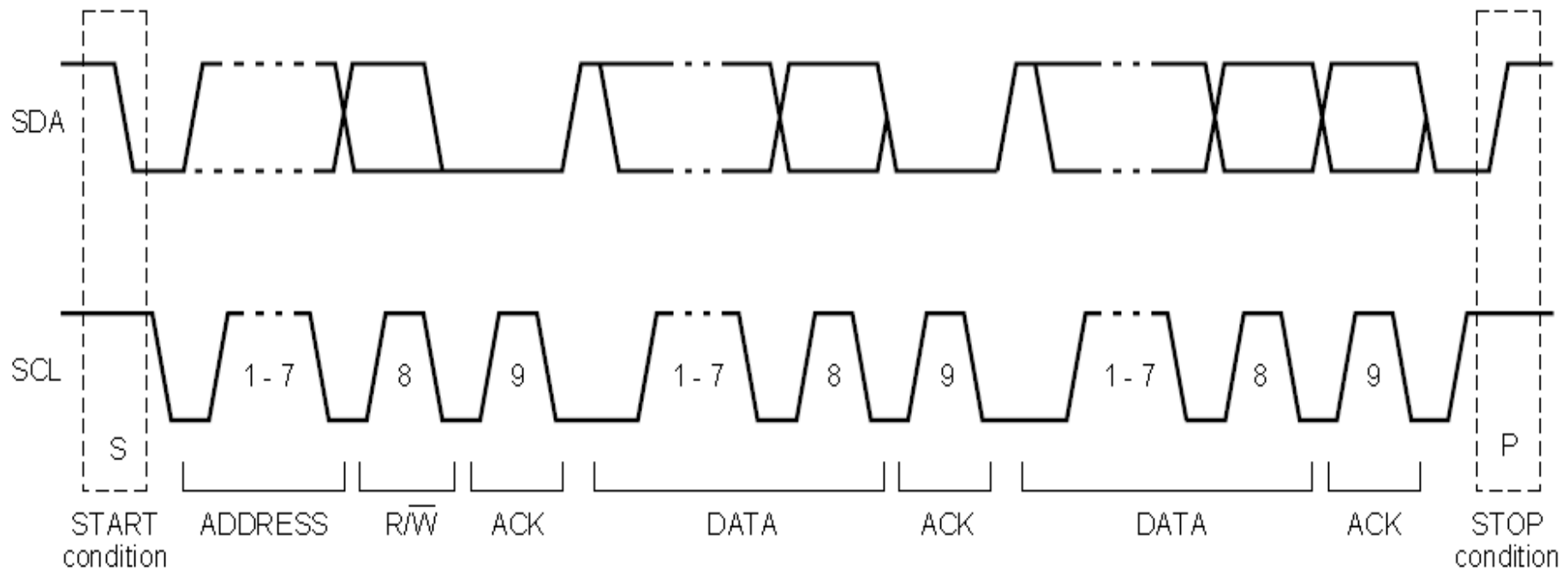




# Sběrnice pro komunikaci mezi IO

## Formát transakce na sběrnici I<sup>2</sup>C

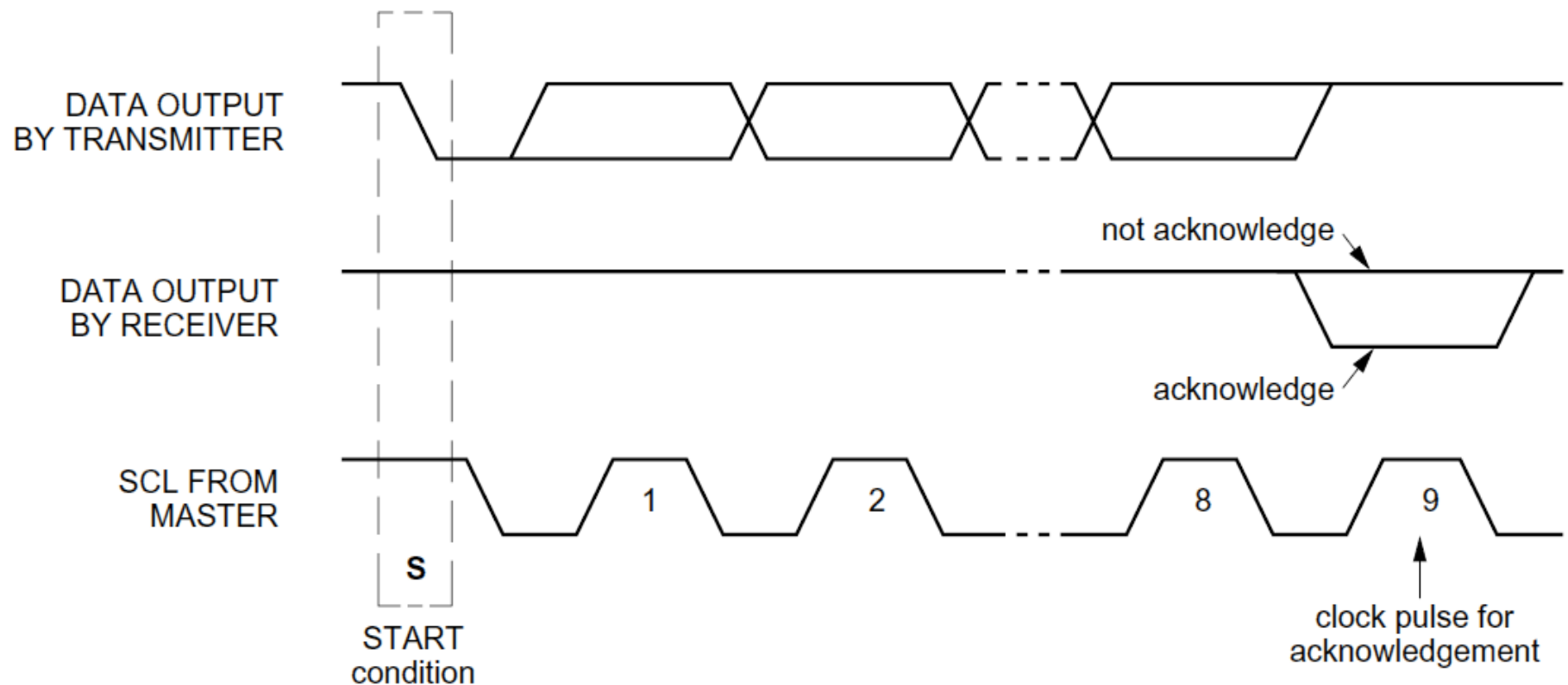
- adresa zařízení, směr přenosu, potvrzení výběru
- data, potvrzení, data, potvrzení, ...



# Sběrnice pro komunikaci mezi IO

## Přenos dat po sběrnici I<sup>2</sup>C

- master generuje SCL a provádí adresaci dat
- slave potvrzuje přijetí, může pozdržet hodiny

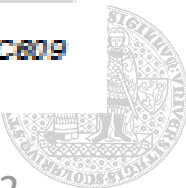
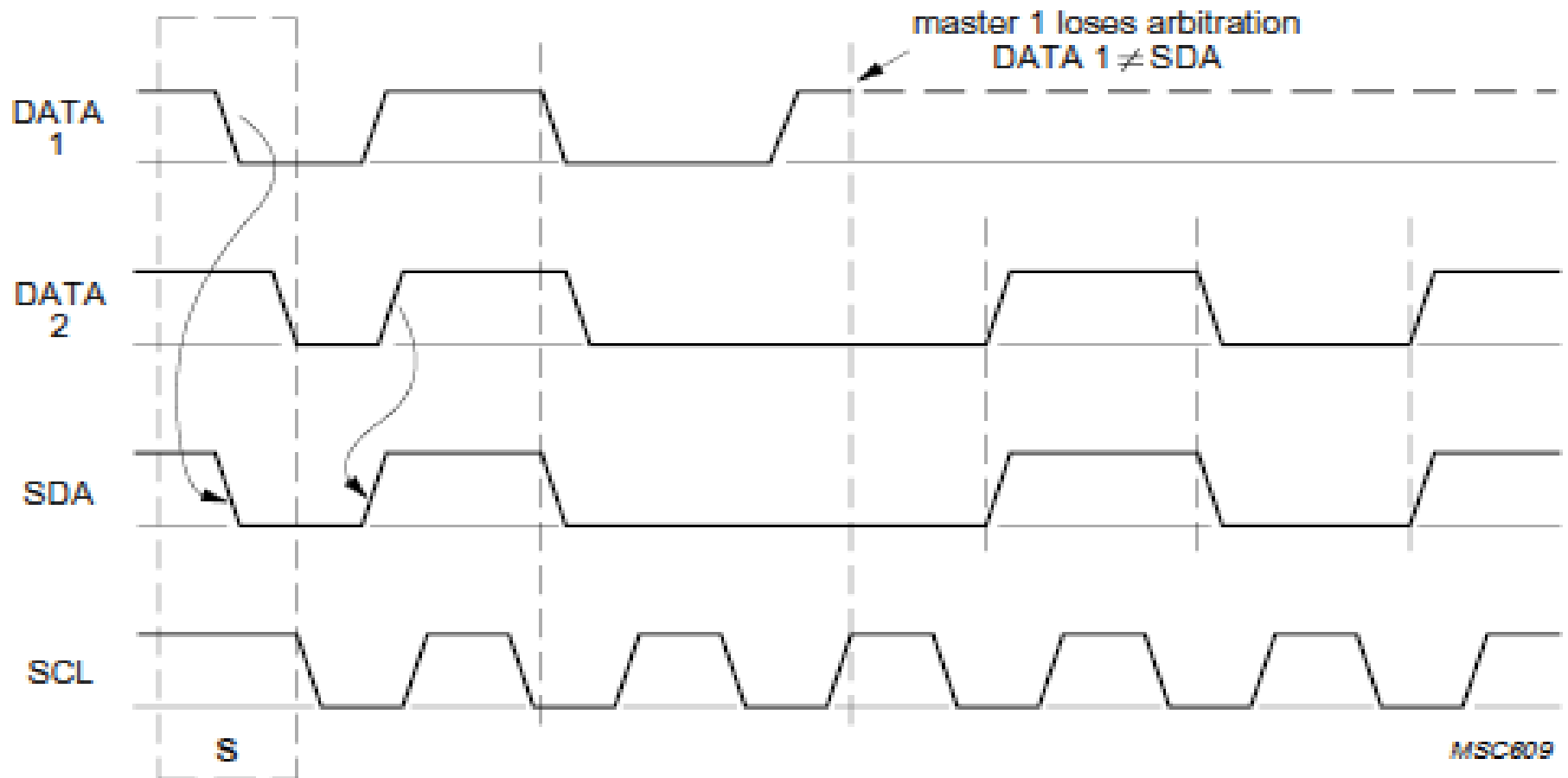


MBC602

# Sběrnice pro komunikaci mezi IO

## Arbitrace na sběrnici I<sup>2</sup>C

- preference dominantního vysílače



# Sběrnice pro připojování (nejenom) disků

---

## ATA/ATAPI – AT Attachment/w Packet Interface

- řadič připojený k systémové sběrnici
  - ♦ přenos 16-bit dat: PIO, Multiword DMA, UltraDMA
  - ♦ 2.1 – 16.7/4.2 – 16.7/16.7 – 100MB/s
  - ♦ UltraDMA: přenos doplněn o CRC
- 2 zařízení na kanál (kabel)
  - ♦ omezená délka kabelů (nevhodné vlastnosti)

## SATA – Serial ATA

- sériový přenos, 150MB/s, 300MB/s
- dvoubodový spoj, podpora hot-plug, ...



# Sběrnice pro připojování (nejenom) disků

---

## SCSI – Small Computer System Interface

- paralelní rozhraní, ANSI standard od r. 1986
- 8/16 bit data, až 640MB/s (160MHz, DDR)
- 15 zařízení na 1 sběrnici, podpora hot-plug
  - ♦ více logických jednotek na 1 zařízení
- inteligentní zařízení
- podmnožinu příkazů používá i ATAPI

## SAS – Serial Attached SCSI

- sériový přenos, dvoubodový spoj
  - ♦ 1bit, 8b/10b kódování
- 150 MB/s (1.0), 300 MB/s (1.1), 600 MB/s (2.0)



# Sběrnice pro připojování externích periférií

---

## Motivace pro nový typ sběrnice

- uživatel nemusí zařízení nijak konfigurovat
- uživatel nemusí otevřít počítač
- jeden typ kabelu pro všechna zařízení
  - ♦ napájení z připojovacího kabelu
- možnost připojení velkého počtu zařízení
- podpora real-time zařízení (zvuk, video)
- instalace zařízení za chodu, bez nutnosti restartovat operační systém
- levná výroba, “fool-proof” design



# Sběrnice pro připojování externích periférií

---

## USB – Universal Serial Bus

- 1,5/12/480 Mb/s
- Control, Bulk, Interrupt, Isochronous přenosy
- hvězdicová struktura – koncová zařízení / hub
- plug-and-play

## IEEE 1394 – FireWire (i.Link)

- stromová/zřetěžená topologie
- 100/200/400 Mb/s (IEEE 1394b: ~800Mb/s)
- paketový přenos

