

Principy počítačů a operačních systémů

Architektura počítačového systému

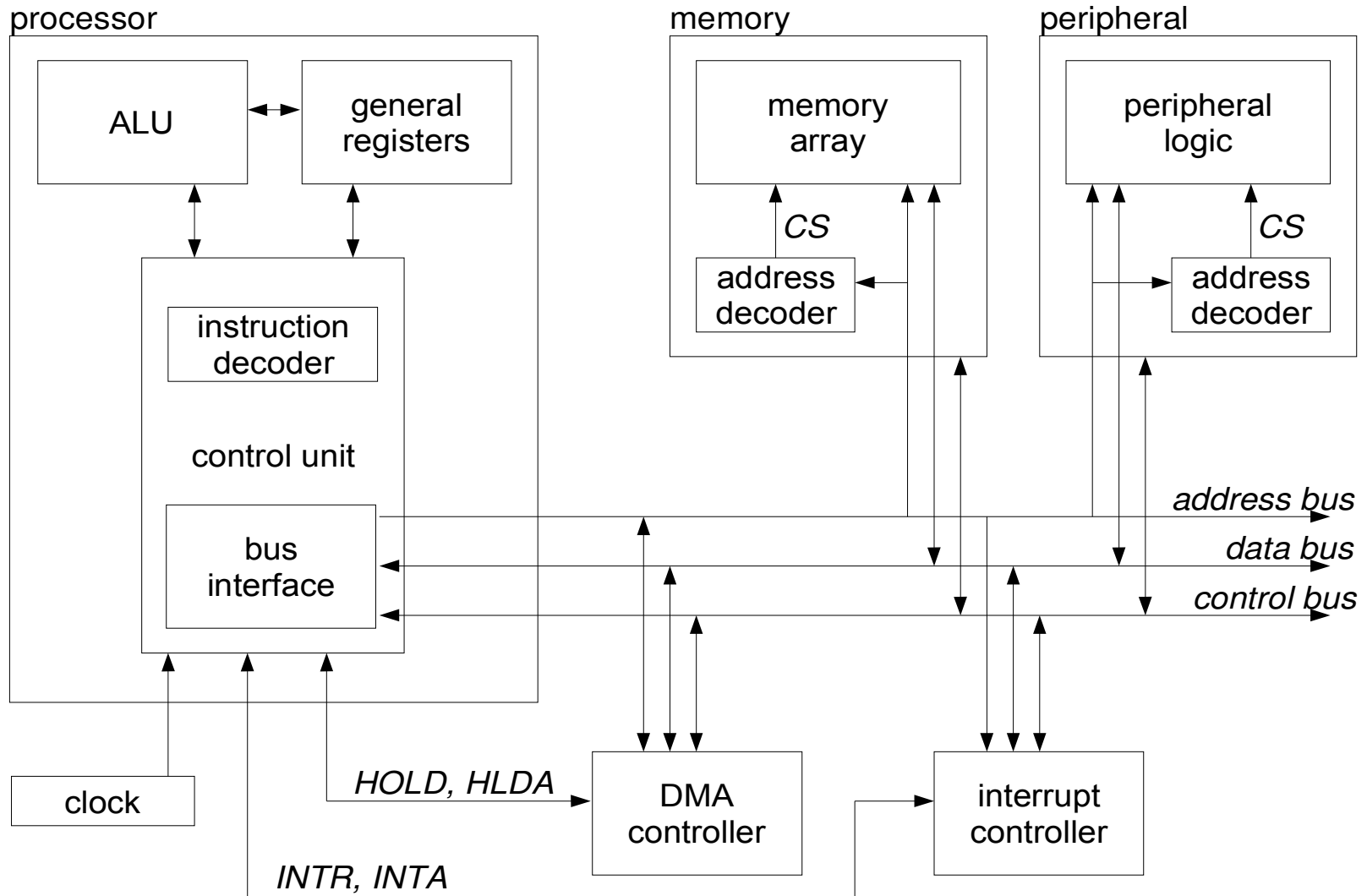
Zimní semestr 2007/2008

Architektura počítačového systému

Uspořádání na úrovni systému

- funkční bloky
 - ♦ procesory, paměti
 - ♦ periferní zařízení
- propojovací systém
 - ♦ přenos dat v rámci systému a mimo něj
- podpůrné obvody
 - ♦ přidělování prostředků
 - ♦ přerušení a přímý přístup do paměti

Architektura PC (přelom 80. a 90. let)



Propojovací systém

Dvoubodové spoje

- přímé spojení (port)
 - ♦ není nutné adresovat
- křížový přepínač (crossbar switch)
- propojovací síť (switch fabric)
 - ♦ topologie

Vícebodové spoje

- účastníci sdílejí přenosové médium
 - ♦ broadcast: 1 vysílač, více příjemců
- sběrnice: sada sdílených vodičů
 - ♦ adresové/datové vodiče, řídicí vodiče

Sběrníkové systémy

Výhody

- nízká cena a flexibilita
- lehké zvládnutí složitosti systému

Nevýhody

- potenciální úzké hrdlo v systému
- rychlost omezena délkou a počtem zařízení
- potřeba spojit různorodá zařízení
 - ♦ různé rychlosti, různé objemy přenášených dat

Sběrníkové systémy

Princip fungování

- vystavení adresy cílového zařízení (address)
- potvrzení cíle a výběr typu přenosu (control)
- přenos dat (data)

Fyzická charakteristika sběrnice

- vodiče, napět'ové úrovně, frekvence hodin, ...

Logická charakteristika sběrnice

- sběrníkový protokol

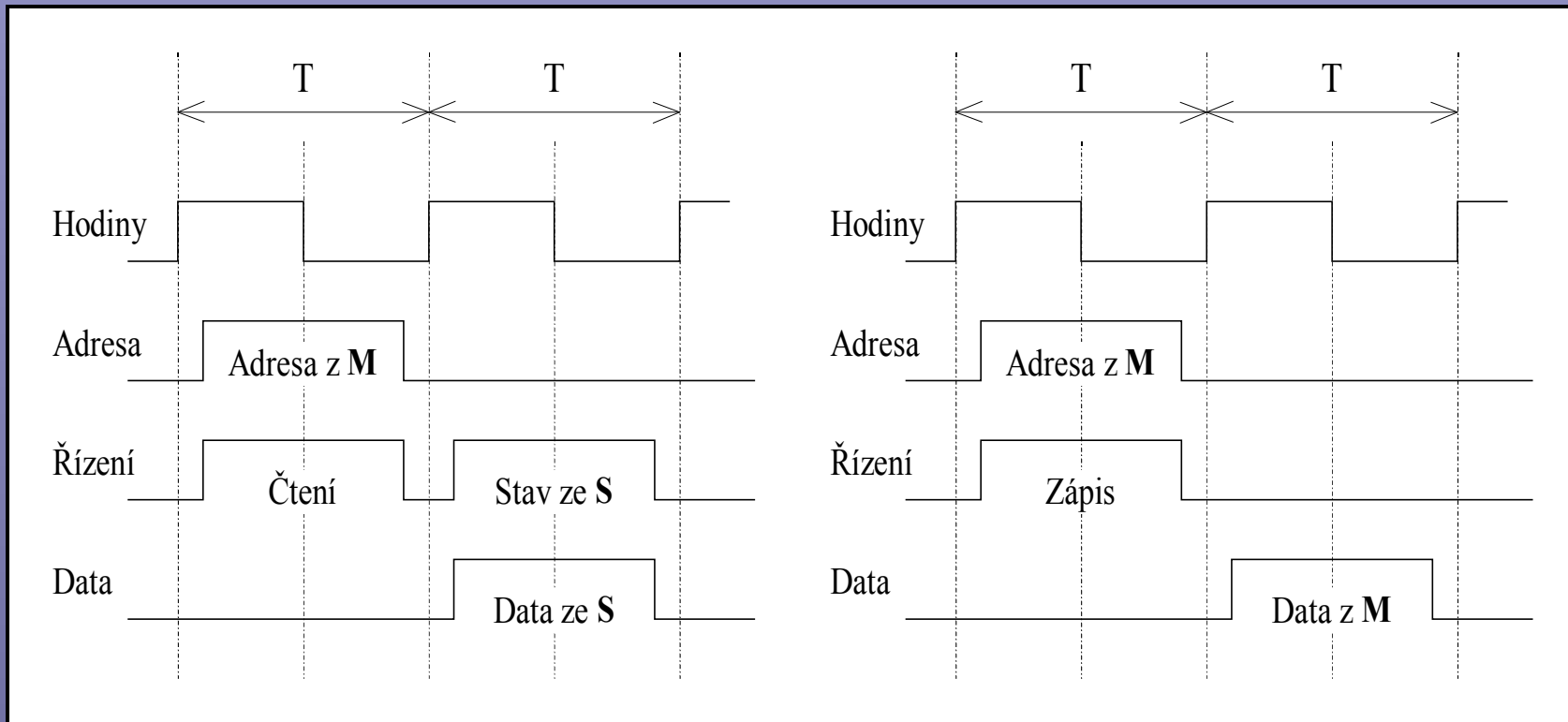
Sběrníkové systémy

Sběrníkový protokol

- definuje povolený průběh transakcí na sběrnici
 - ♦ logický význam signálů na vodičích
 - ♦ popis pomocí stavových a časových diagramů
- synchronní vs. asynchronní protokol
 - ♦ přechody mezi fázemi transakce určeny hodinovým signálem resp. změnou řídících signálů
- centralizované vs. distribuované přidělování
 - ♦ o přidělení sběrnice rozhoduje centrální arbitr resp. zařízení v rámci distribuovaného protokolu

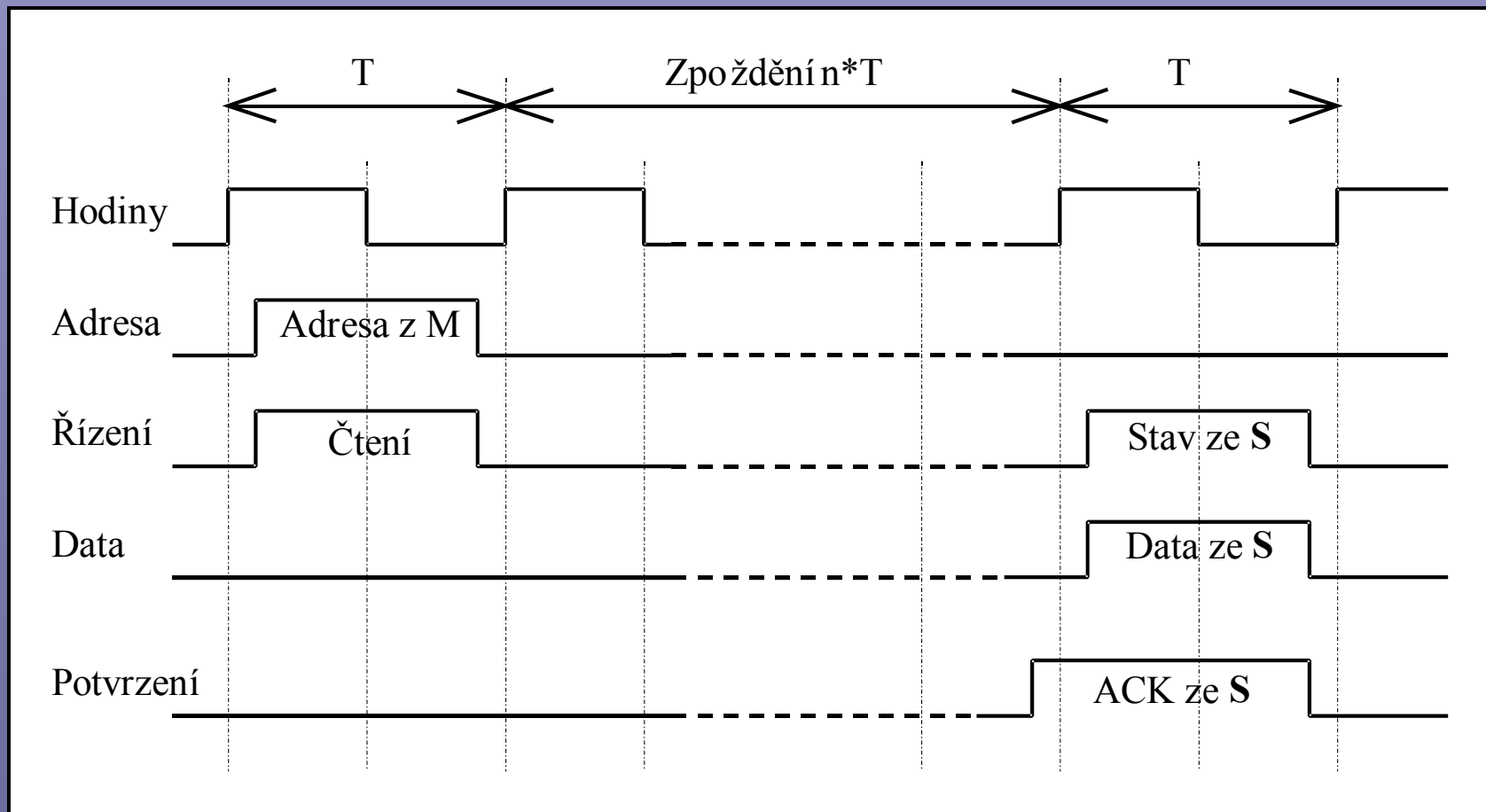
Sběrníkové systémy

Synchronní čtení/zápis



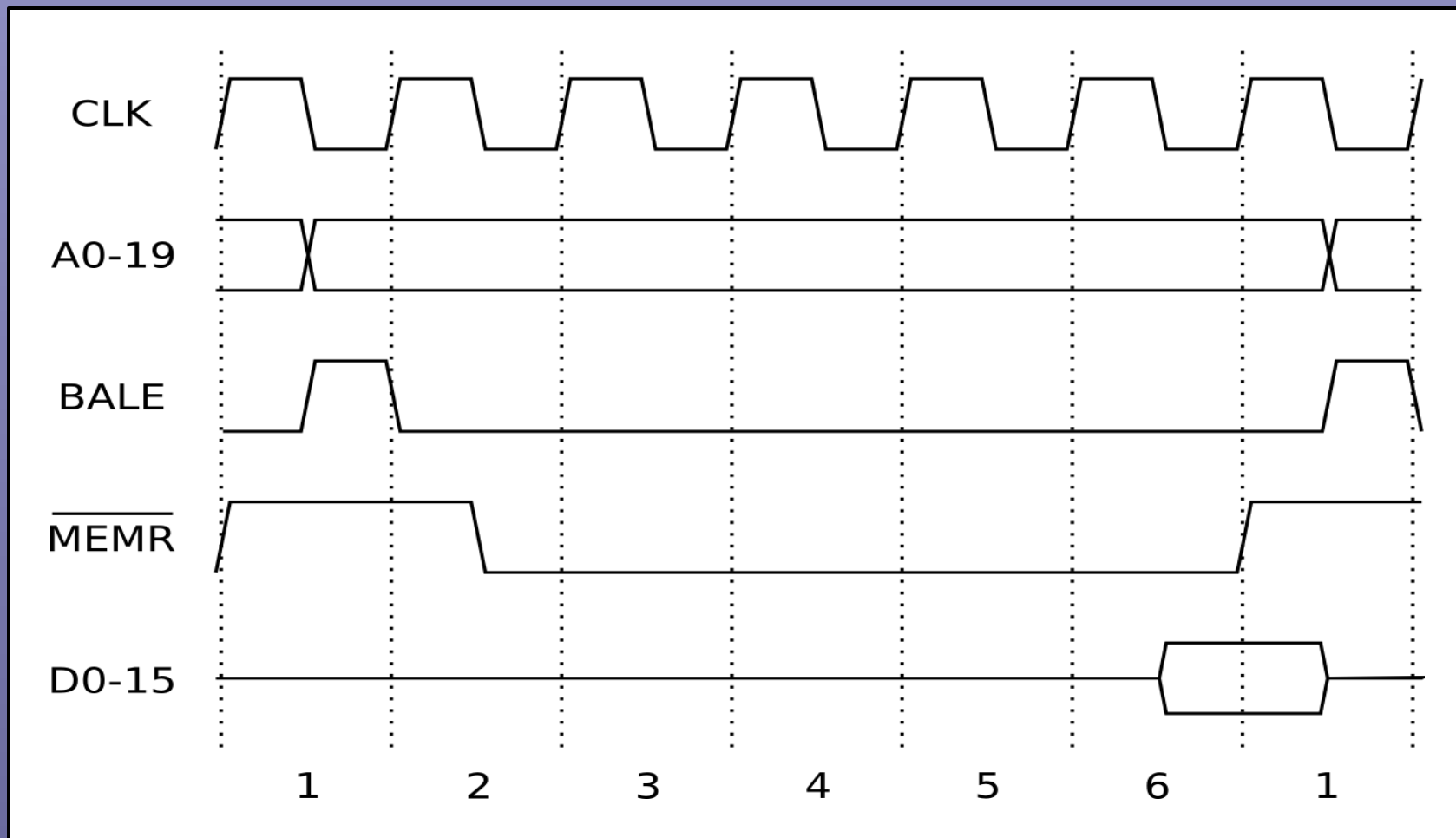
Sběrníkové systémy

Synchronní čtení s čekáním



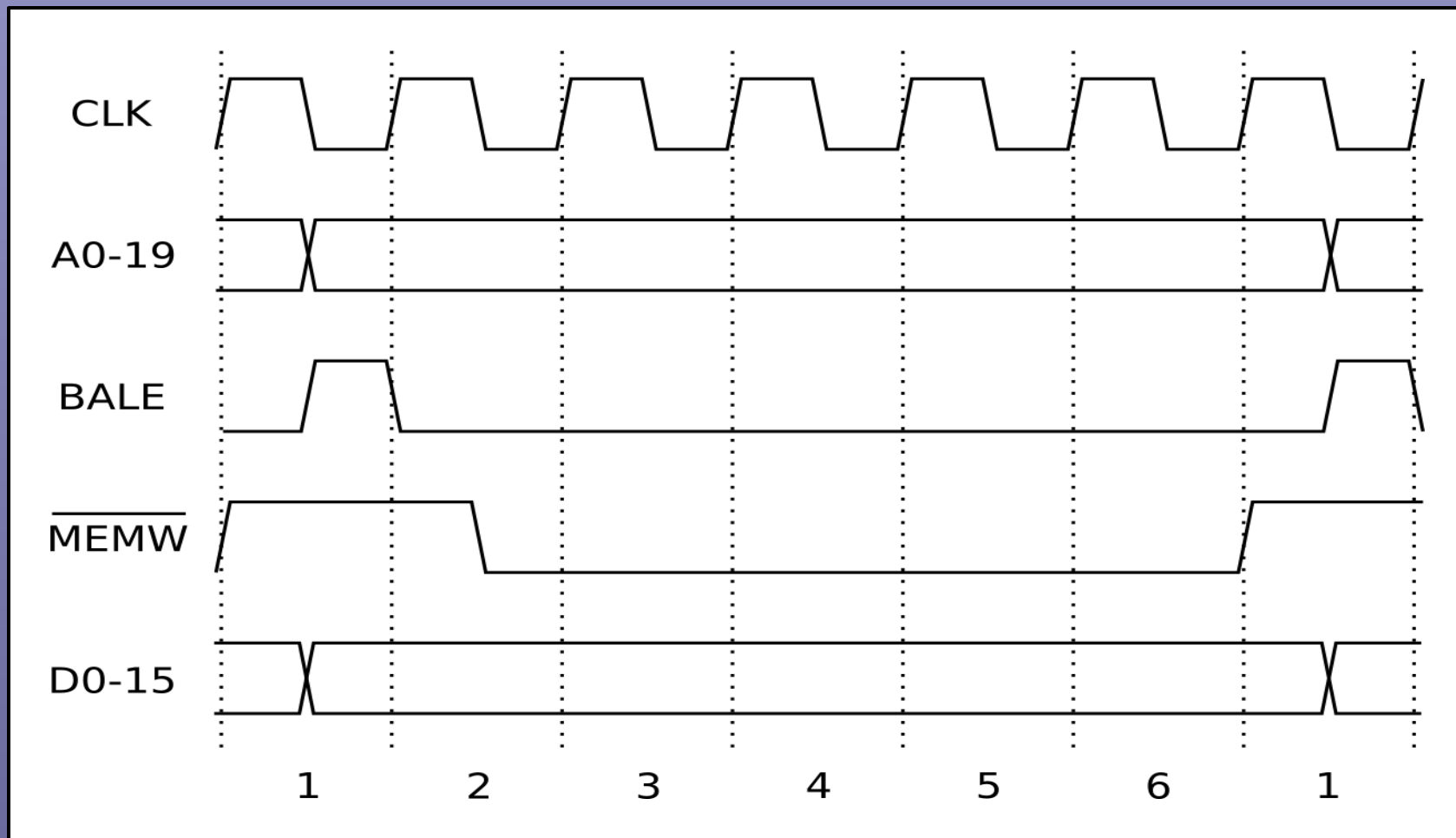
Sběrnice systémy

Čtení z paměti na sběrnici ISA



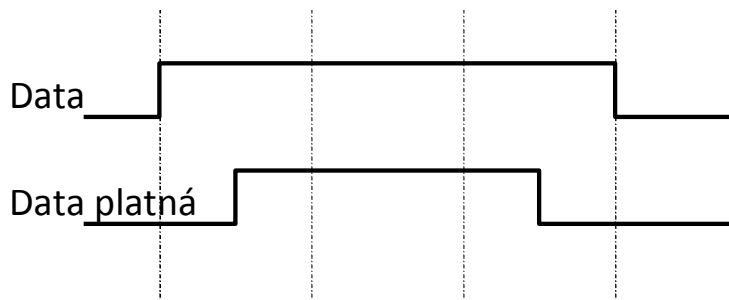
Sběrníkové systémy

Zápis do paměti na sběrnici ISA

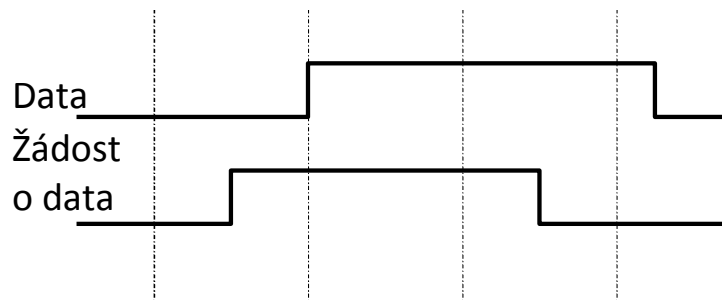


Sběrníkové systémy

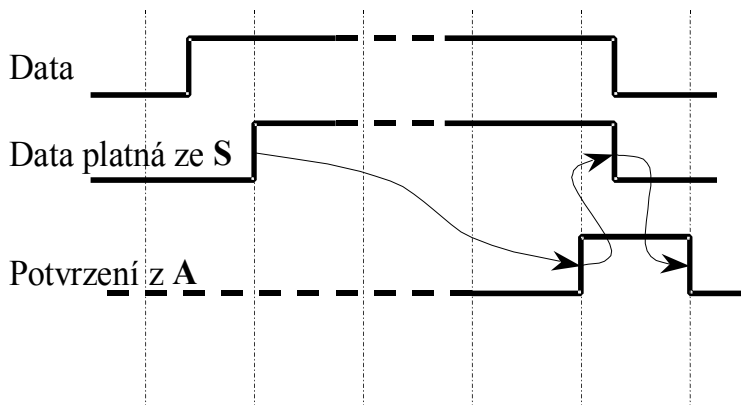
Asynchronní čtení/zápis



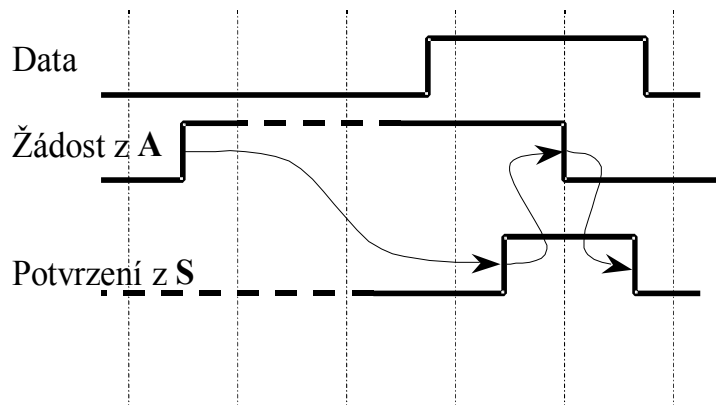
a) Řízení zdrojem



b) Řízení příjemcem



a) přenos vyvolaný zdrojem



b) přenos vyvolaný příjemcem

Sběrníkové systémy

Centrální přidělování sběrnice

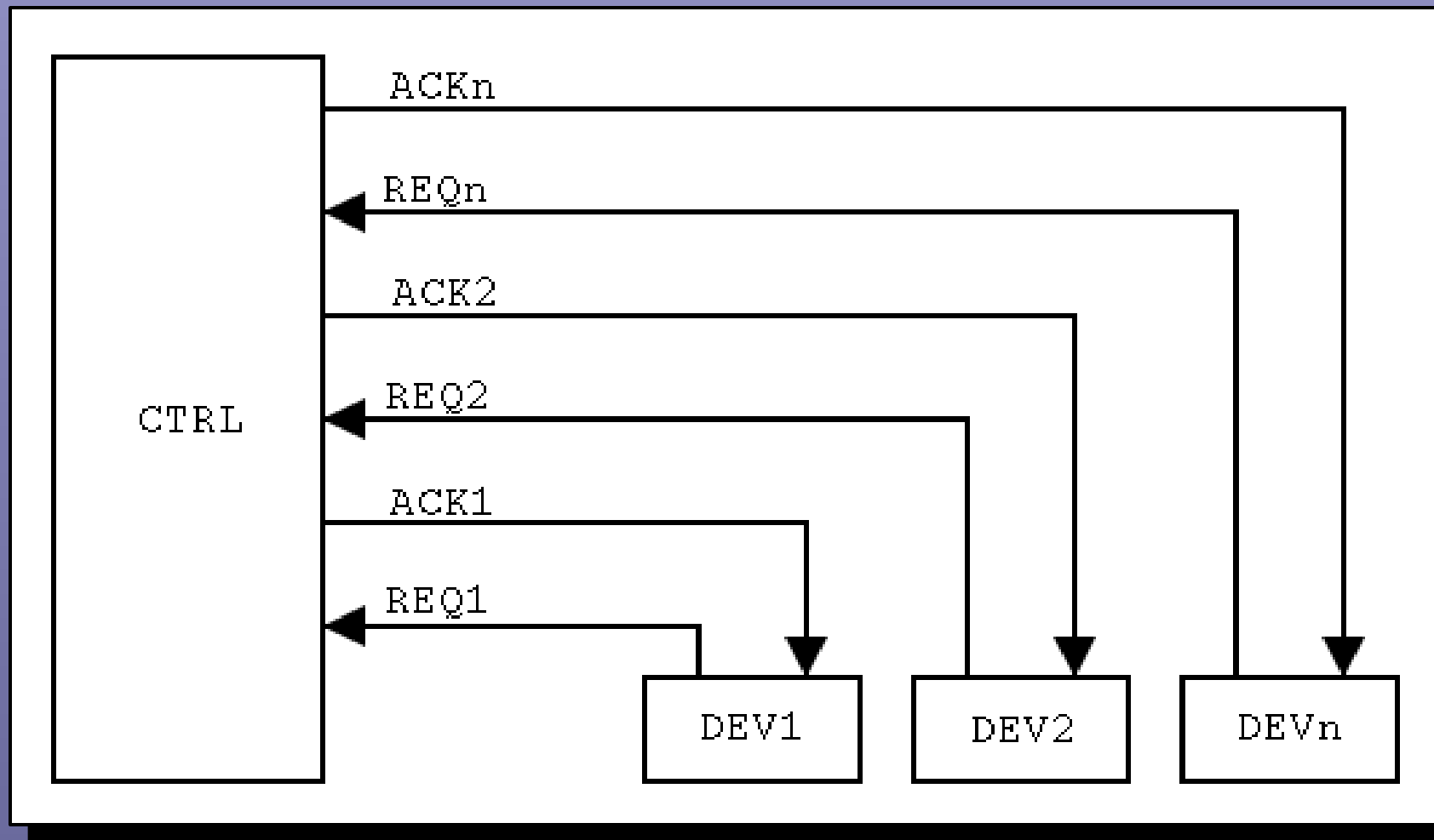
- fyzické uspořádání
 - ♦ samostatné žádosti, cyklické výzvy (polling)
 - ♦ prioritní zřetězení (daisy chain)
- řešení současných požadavků
 - ♦ náhodné, dle pořadí vzniku, prioritní

Distribuované přidělování sběrnice

- kolizní (CSMA/CD)
- token bus
- adresová priorita

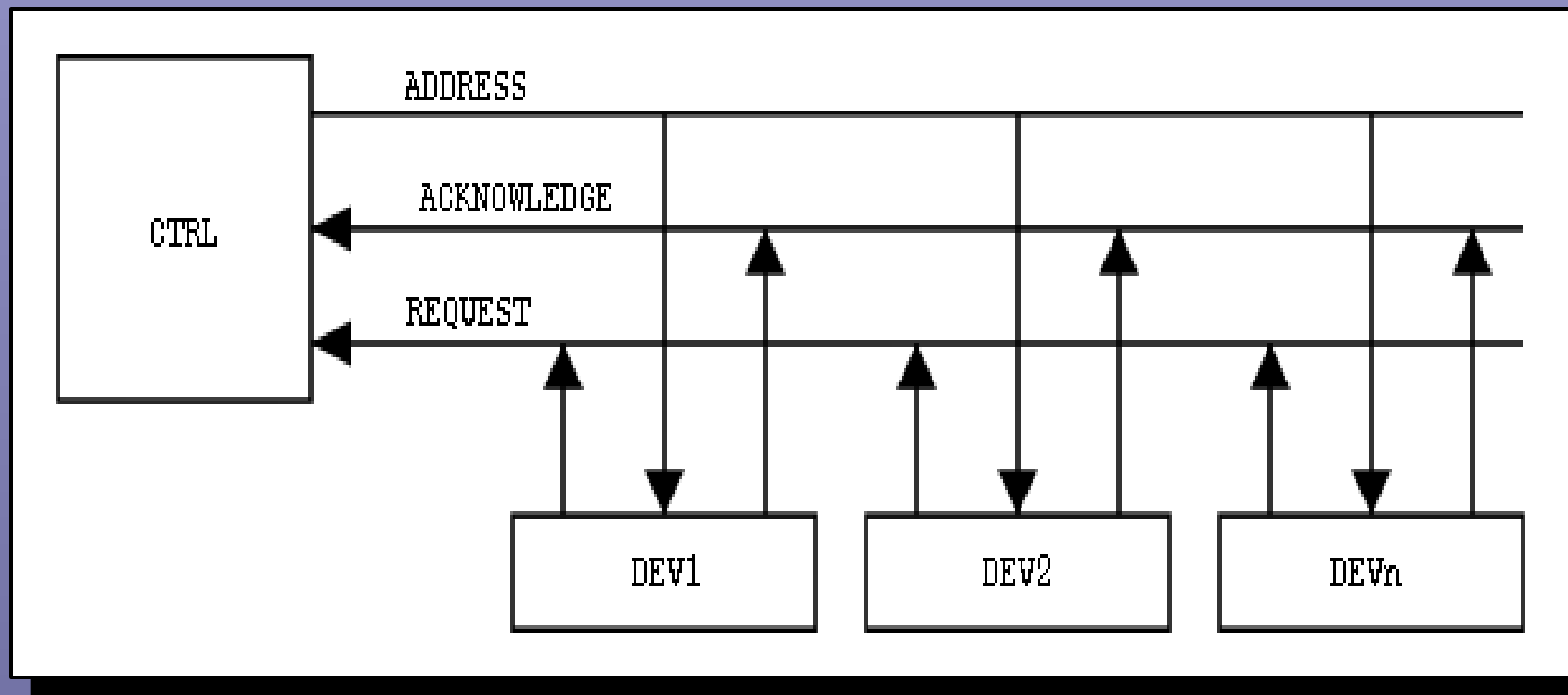
Centrální přidělování sběrnice

Nezávislé žádosti



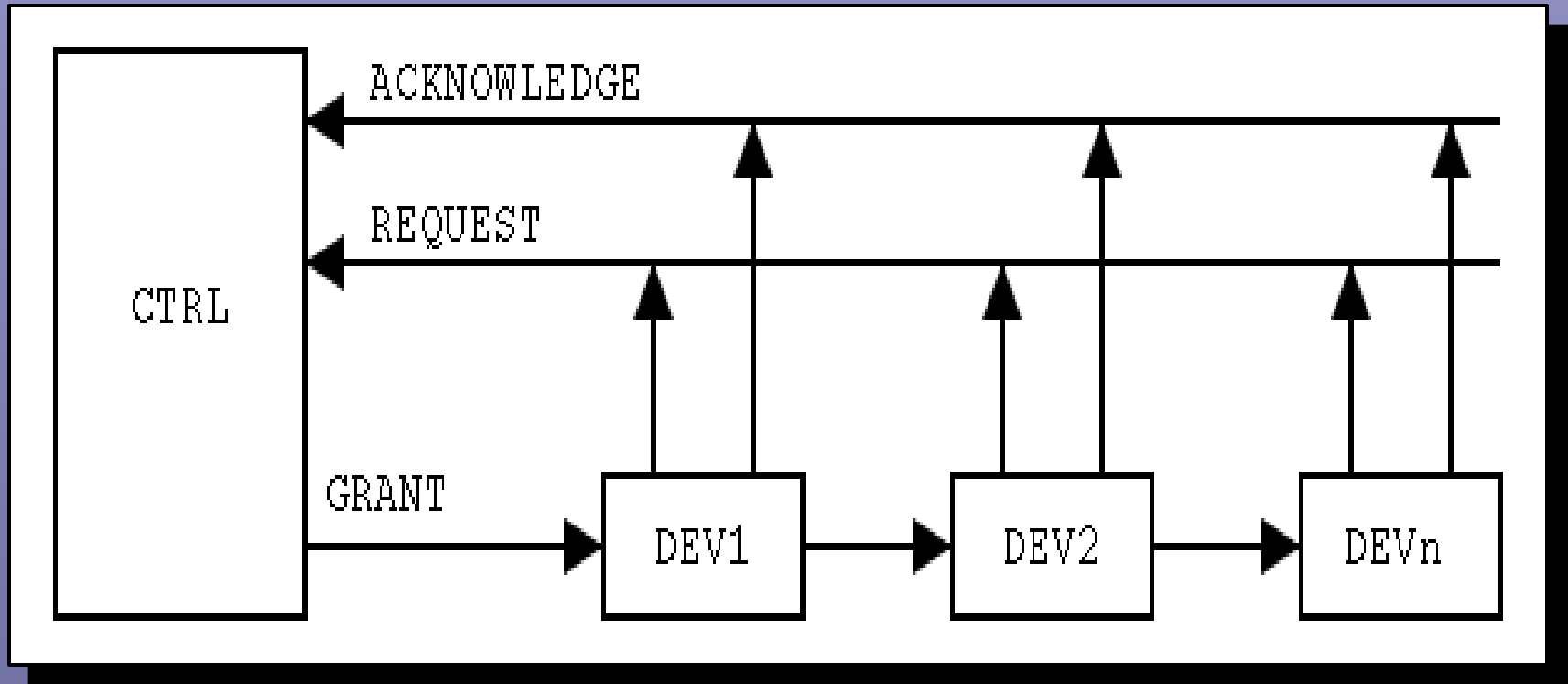
Centrální přidělování sběrnice

Cyklické výzvy (polling)



Centrální přidělování sběrnice

Prioritní zřetězení (daisy chain)



Ovládání periferních zařízení

Periferie ovládány programem

- vykonání specifické operace
 - ♦ výběr zařízení
 - ♦ zápis příkazů specifických pro zařízení
 - ♦ čtení stavové informace specifické pro zařízení

Rozhraní paměť/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 režie 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

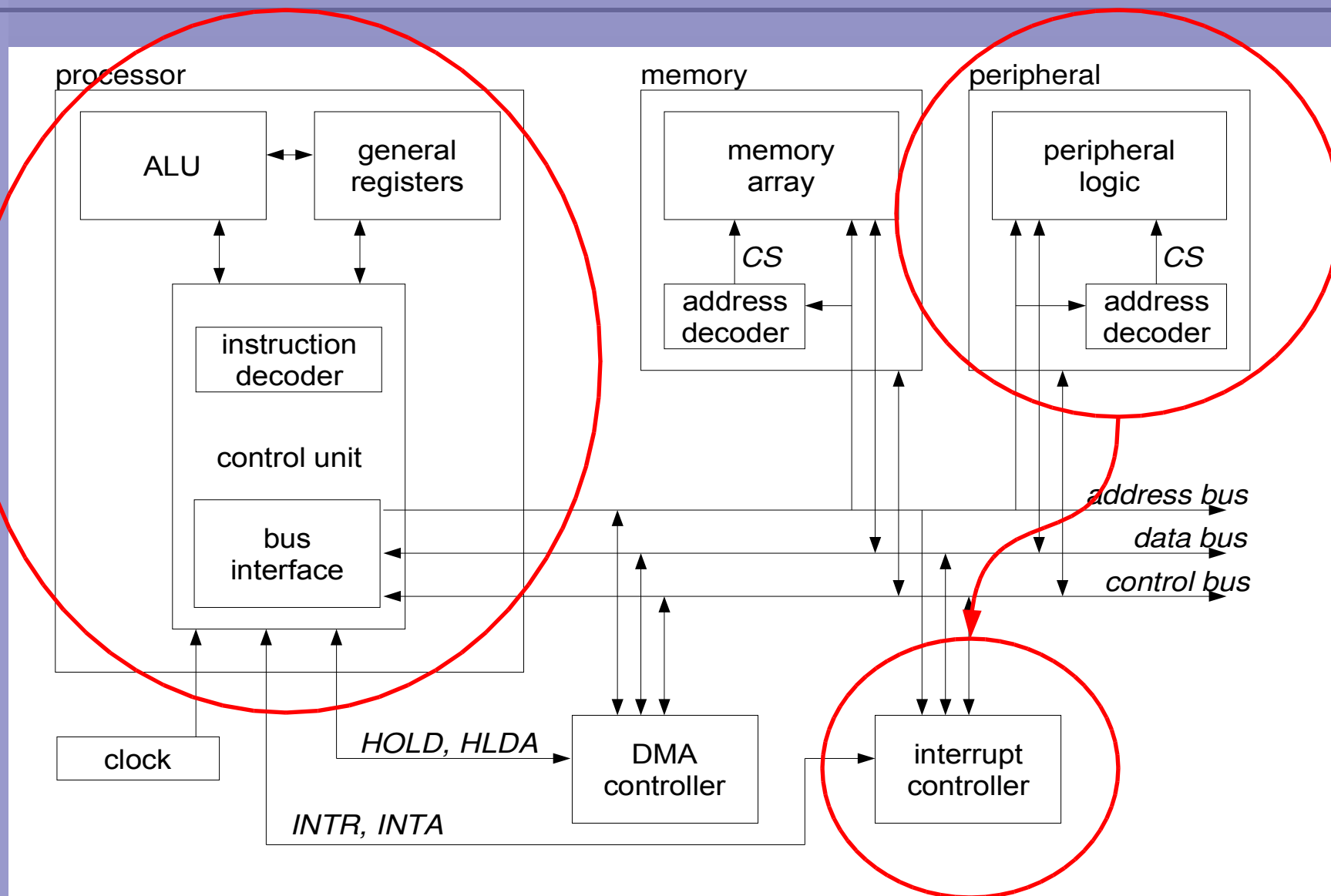
Procesor

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

Systém

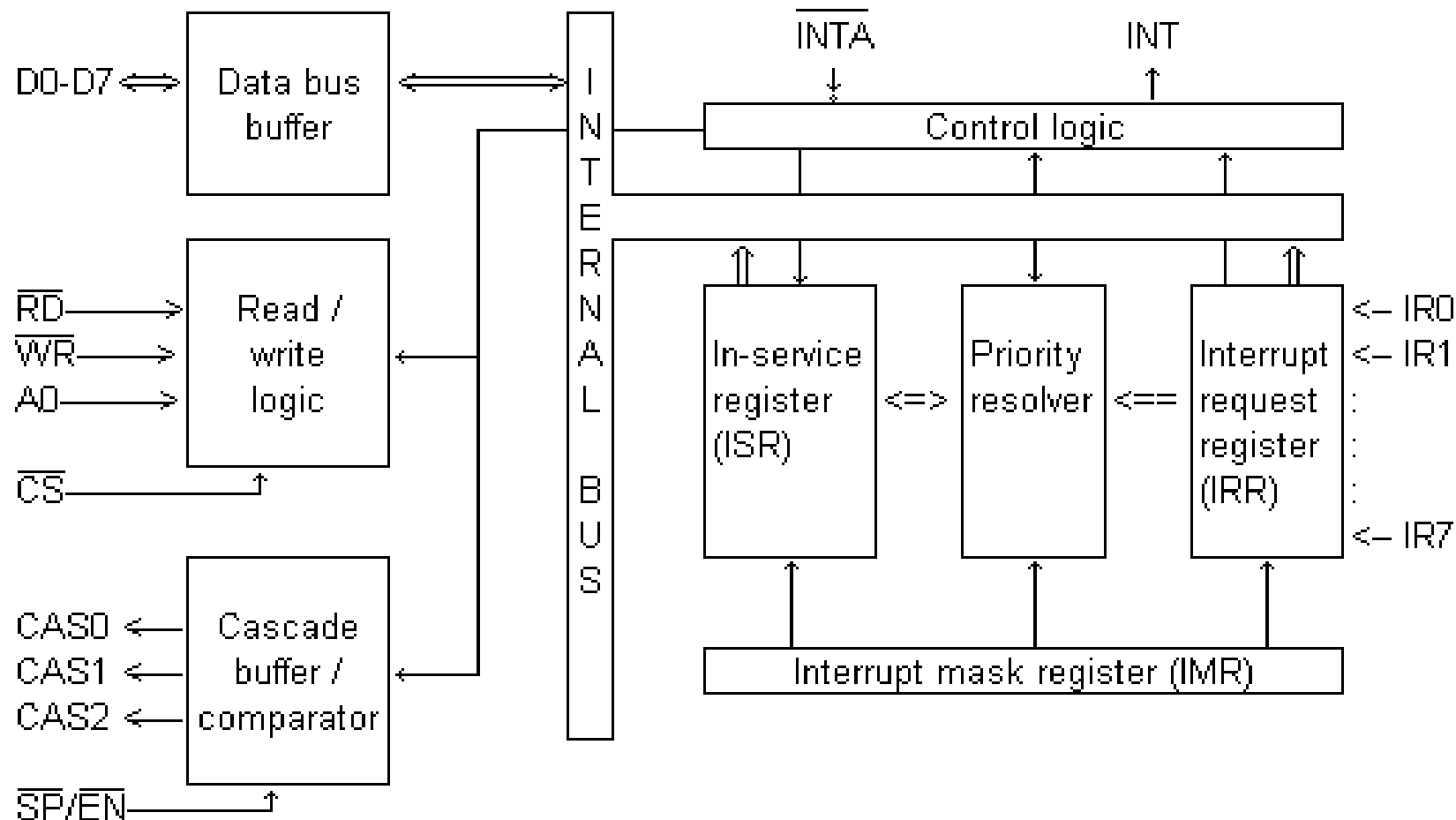
- řadič přerušení
 - ♦ více přerušovacích signálů, vyhodnocení priority

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 přerušení 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 přerušení
5. CPU zavolá obslužnou rutinu odpovídající vektoru
6. Ostatní přerušení blokována, dokud řadič neobdrží příkaz “Non-Specific EOI”

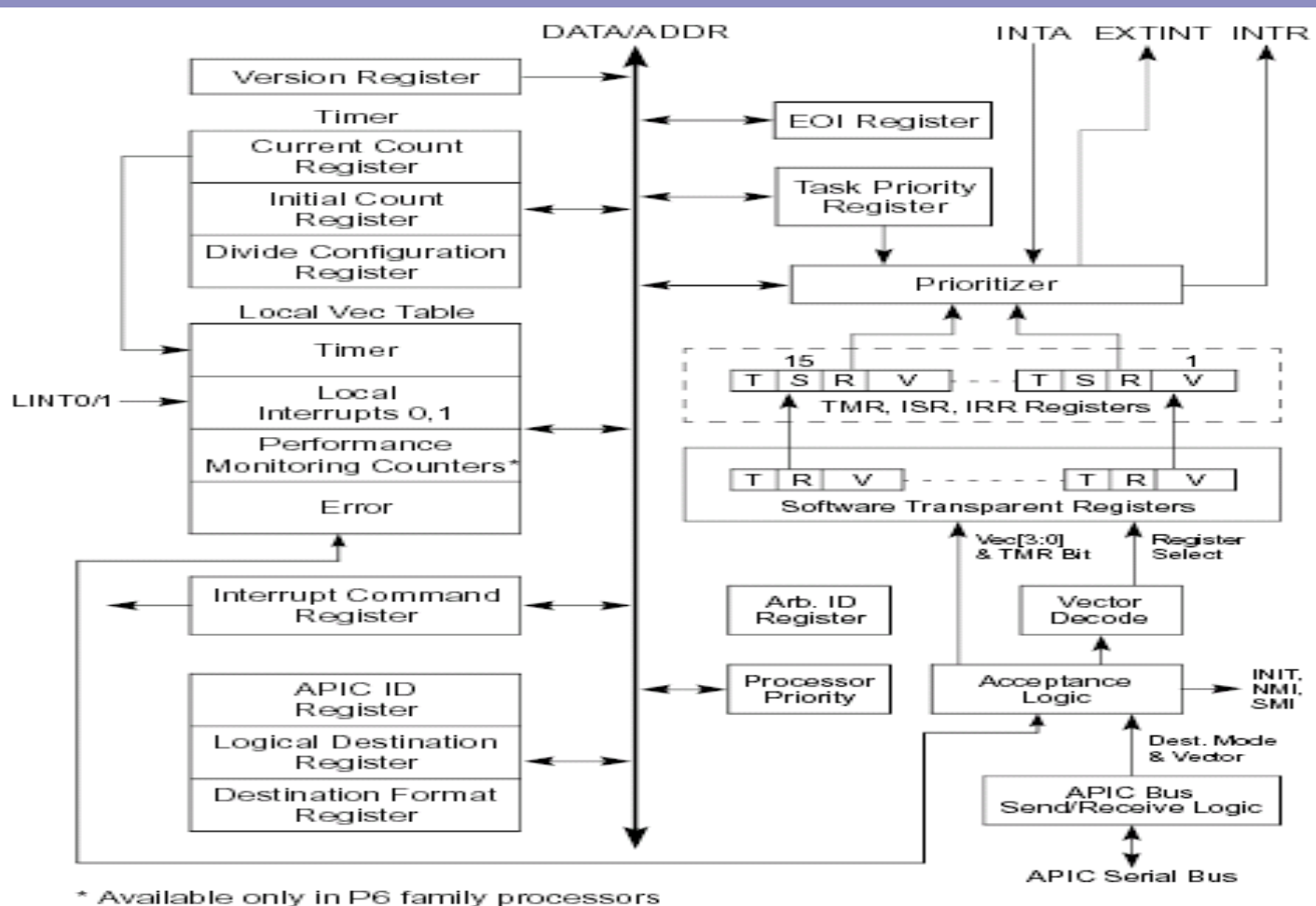
Zpracování přerušení na procesoru MIPS

Vyvolání a obsluha přerušení

1. Aktivován signál INTx
2. Procesor nastaví registry Cause a EPC a skočí na obsluhu General Exception na adrese 0x80000180
 - ♦ dojde k uschování stavu procesoru
3. Obsluha GE zjistí příčinou a vyvolá specifickou obslužnou rutinu
 - ♦ pokud bylo přerušení požadováno více zdroji, jsou tyto obslouženy v pořadí určeném obslužnou rutinou
4. Při návratu je obnoven stav procesoru a skočí se na adresu v EPC

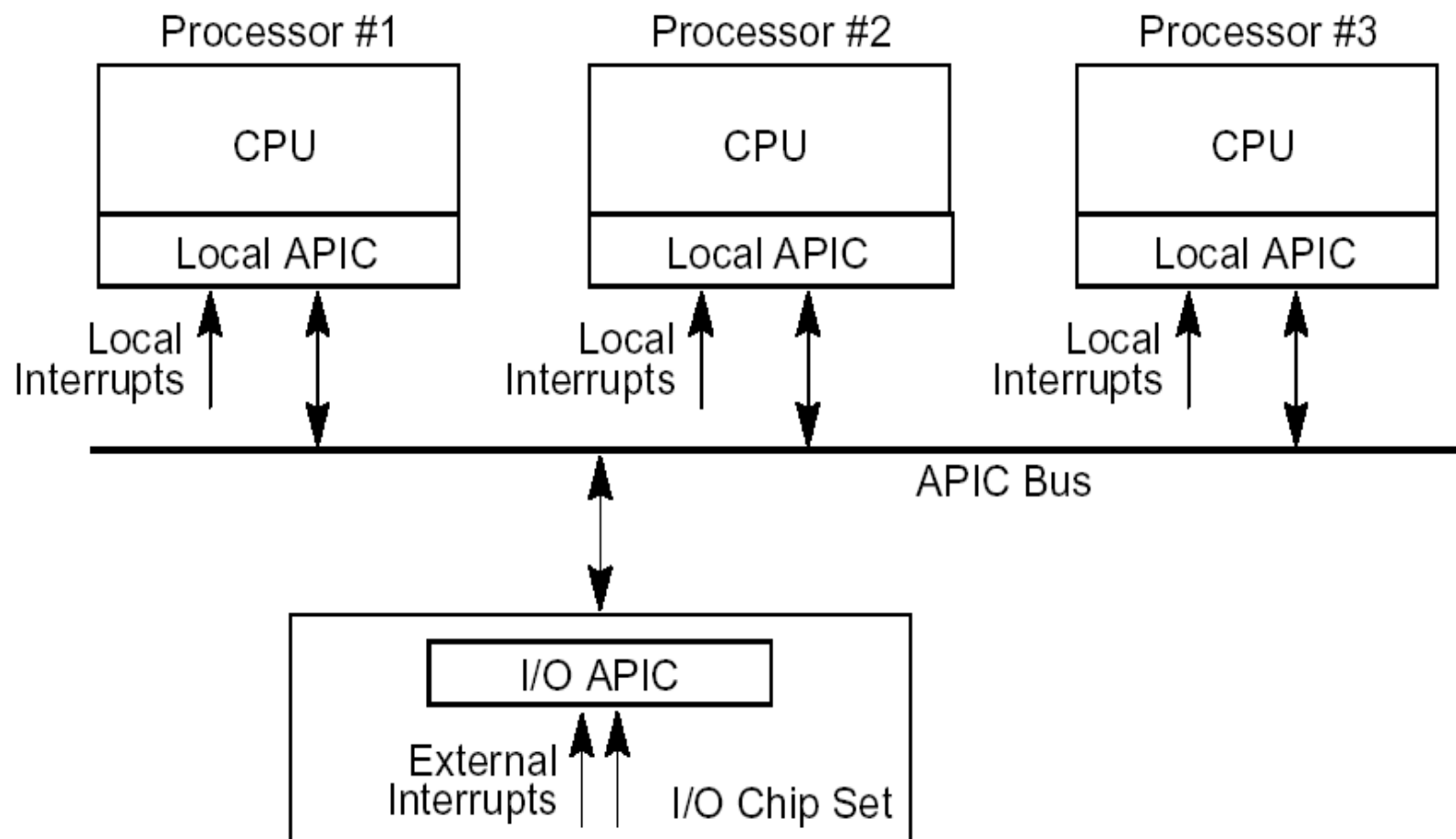
Moderní přerušovací subsystém

Řadič integrovaný na procesoru



Moderní přerušovací subsystém

Přerušování ve víceprocesorových systémech



Přenos dat po sběrnici

Za účasti procesoru

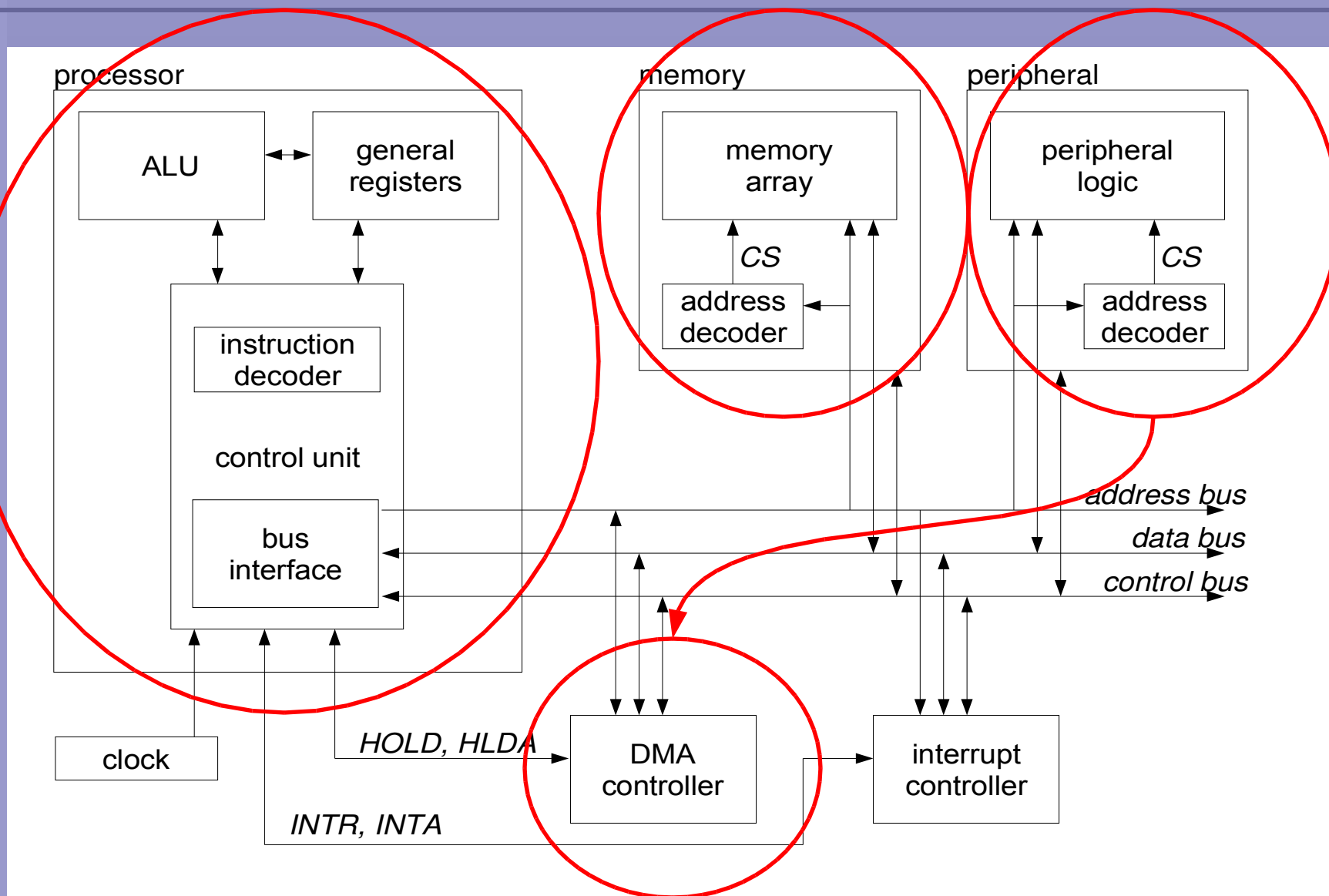
- přenos iniciován při změně stavu zařízení
 - ♦ polling, interrupt-driven I/O
- přenos realizován cyklem v programu
 - ♦ přečíst slovo z periferie/paměti
 - ♦ zapsat slovo do paměti/periferie
 - ♦ „programmed I/O“
- rychlý přenos, vytěžuje hlavní procesor
 - ♦ procesor by měl dělat něco užitečnějšího

Přenos dat po sběrnici

Bez účasti procesoru

- přenos iniciován při změně stavu zařízení
- přenos realizován řadičem nebo zařízením
 - ♦ program musí pouze nastavit parametry přenosu
- přenos zařízení/paměť
 - ♦ direct memory access
 - ♦ řadič DMA, bus mastering
- přenos zařízení/zařízení
 - ♦ bus mastering

System s řadičem DMA

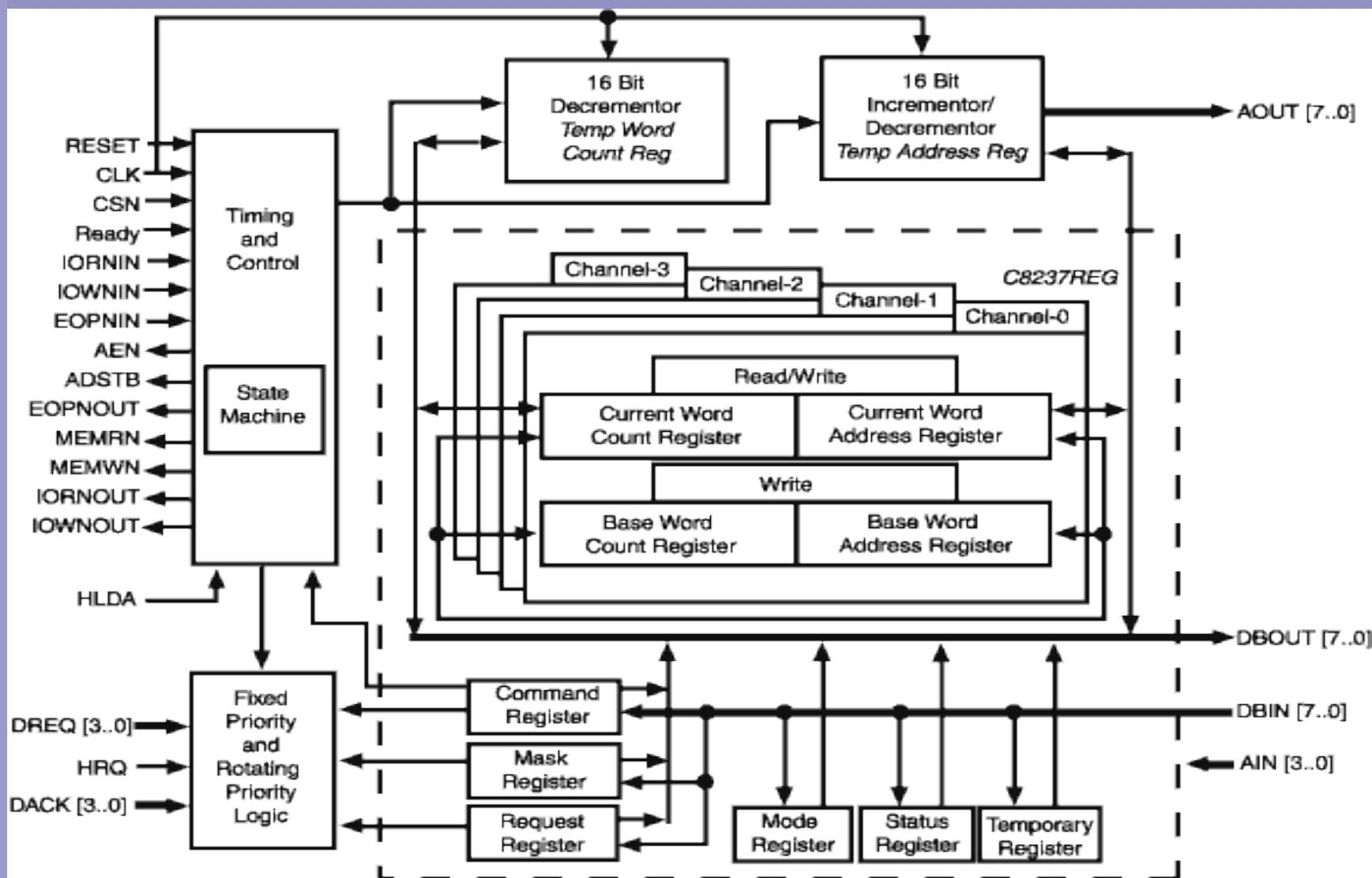


Řadič DMA

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 dat po sběrnici

Přenos bloku dat pomocí DMA (i8237)

1. program nastaví řadič a periférii a spustí přenos
2. na začátku cyklu sběrnice řadič signalizuje HOLD
3. CPU testuje HOLD na začátku strojového cyklu
 - ♦ pokud nepotřebuje sběrnici, signalizuje HLDA a uvolní ji
4. po přijetí HLDA řadič přenese slovo z/do paměti
 - ♦ vystaví adresu v paměti a řídicí signály, a aktivuje periférii aby přečetla/zapsala slovo z/na datovou sběrnici
5. na konci cyklu sběrnice řadič uvolní HOLD
6. procesor uvolní HLDA a připojí se ke sběrnici

Přenos dat po sběrnici

Transparentní režim přenosu s pomocí DMA

- přenos probíhá nezávisle na procesoru
- řadič používá sběrnici v době, kdy ji nepoužívá procesor
 - ♦ sběrnice má 2 zdroje hodinového signálu, vzájemně fázově posunuté
 - ♦ použije se náběžná i sestupná hrana hodinového signálu
 - ♦ efektivně se zdvojnásobí propustnost sběrnice

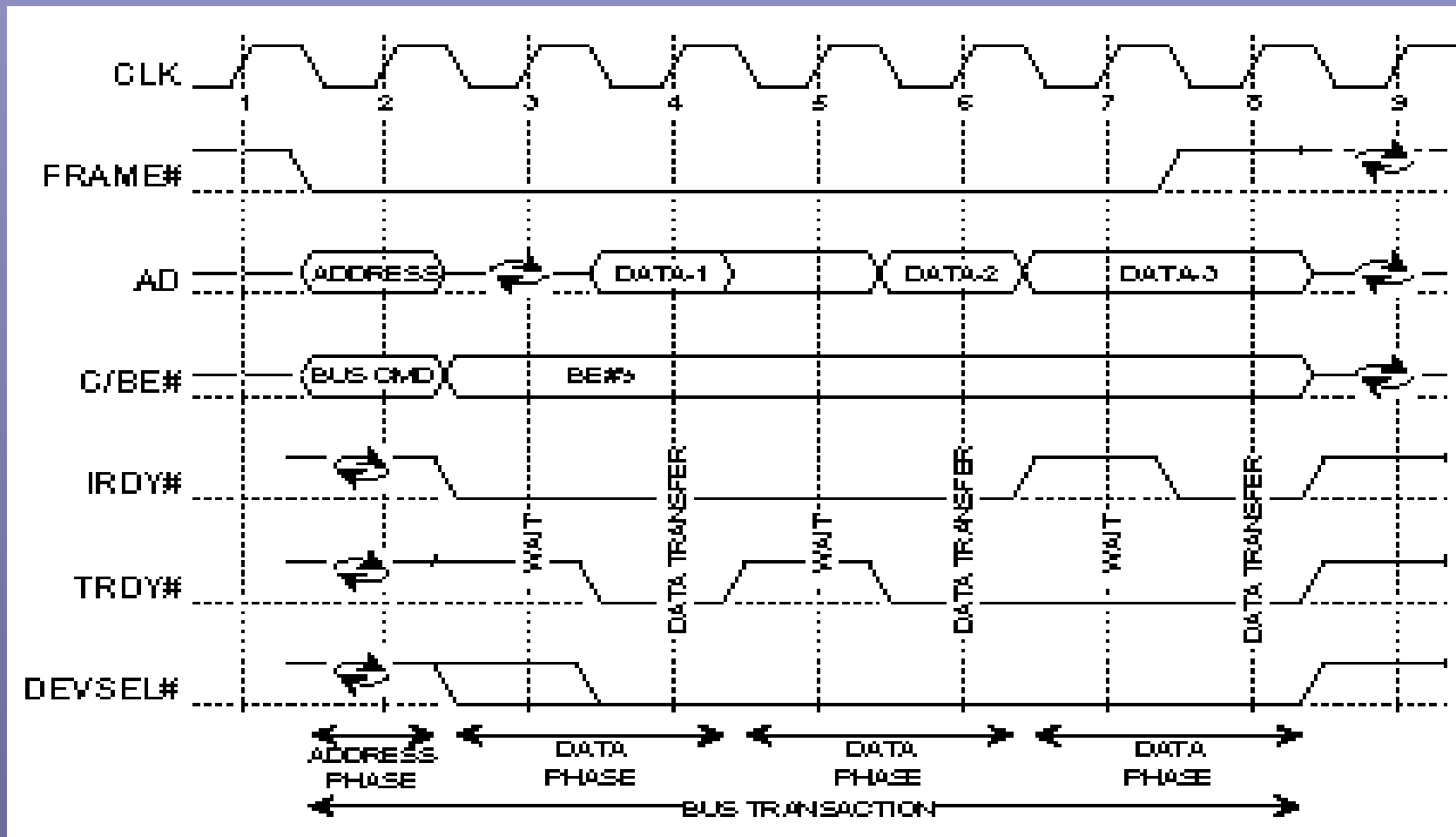
Přenos dat po sběrnici

Bus mastering

- zobecnění konceptu DMA
- sběrnici může řídit libovolný účastník
 - ♦ nutno požádat 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
- dávkový režim přenosu (burst mode)
 - ♦ 1 adresový cyklus na blok dat
- přenos nesouvislých bloků (scatter/gather)

Přenos dat po sběrnici

Burst transakce na sběrnici PCI



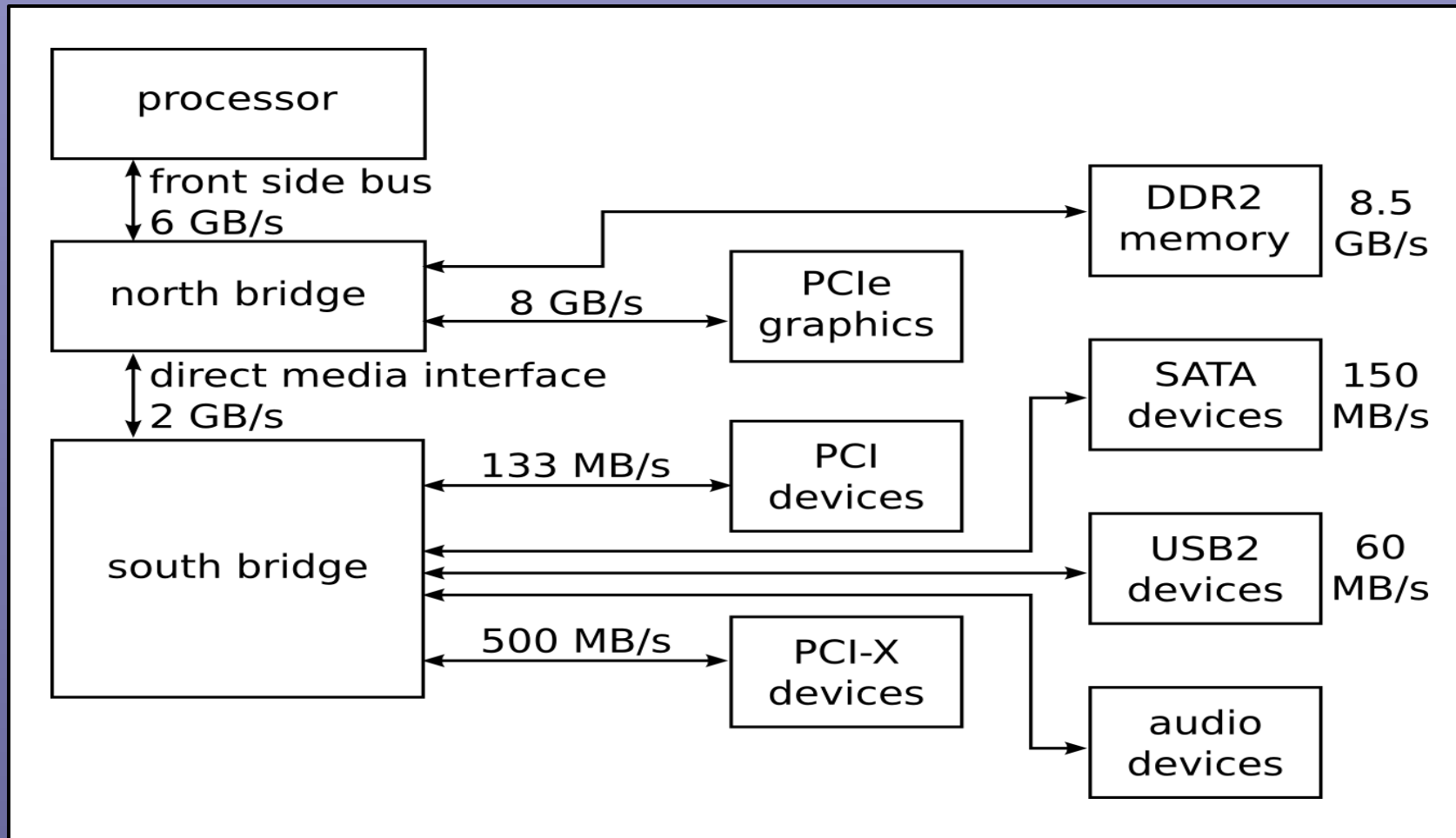
Přenos dat po sběrnici

Obecné poznámky k DMA

- přenos pomocí DMA není nutně rychlejší než PIO
 - ♦ umožňuje procesoru dělat něco užitečnějšího
- systém 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

System sběrnic v moderním PC

Propojení různě rychlých sběrnic



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 periférií

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 periférií

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

I²C – Inter IC

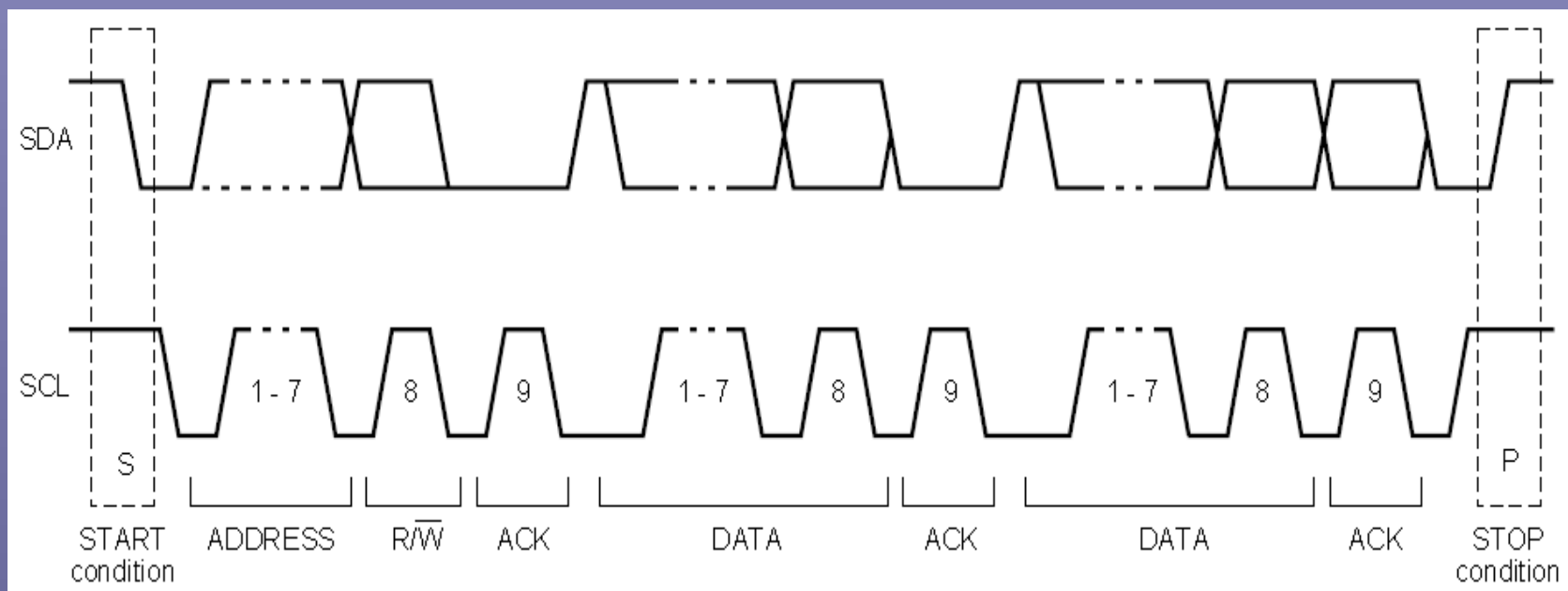
- dvouvodičová (SDA, SCL) sériová sběrnice
 - ♦ v klidovém stavu udržováno kladné napětí (pull-up)
- přenos dat po bajtech, potvrzení příjmu
- rychlosti 0- 100/400/1000/3400 kbit/s
- master/slave řízení
- možnost multi-master konfigurace
 - ♦ synchronizace hodin + arbitrace
- www.philipslogic.com/i2c



Sběrnice pro komunikaci mezi IO

Formát transakce na sběrnici I2C

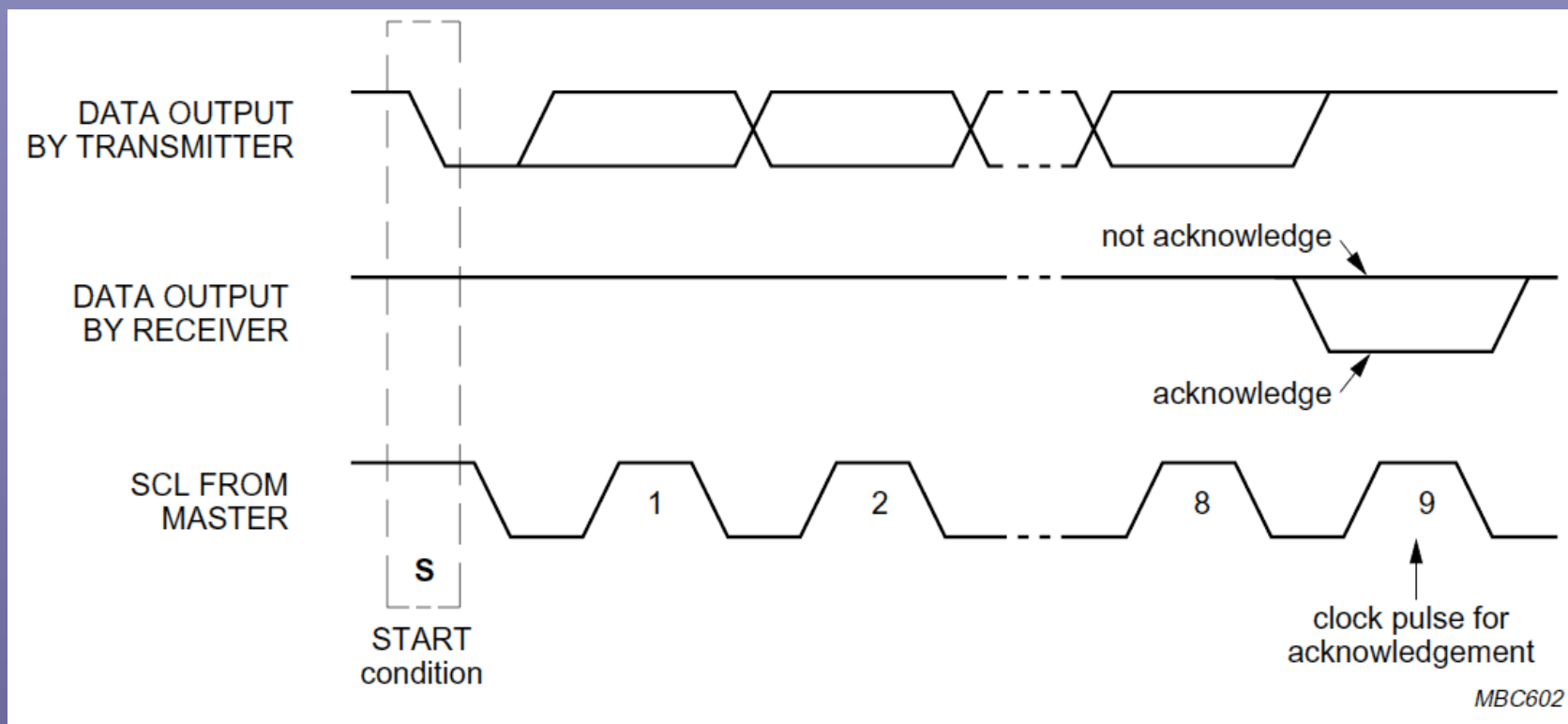
- adresa zařízení, směr přenosu, potvrzení výběru
- data, potvrzení, data, potvrzení, ...



Sběrnice pro komunikaci mezi IO

Přenos dat po sběrnici I2C

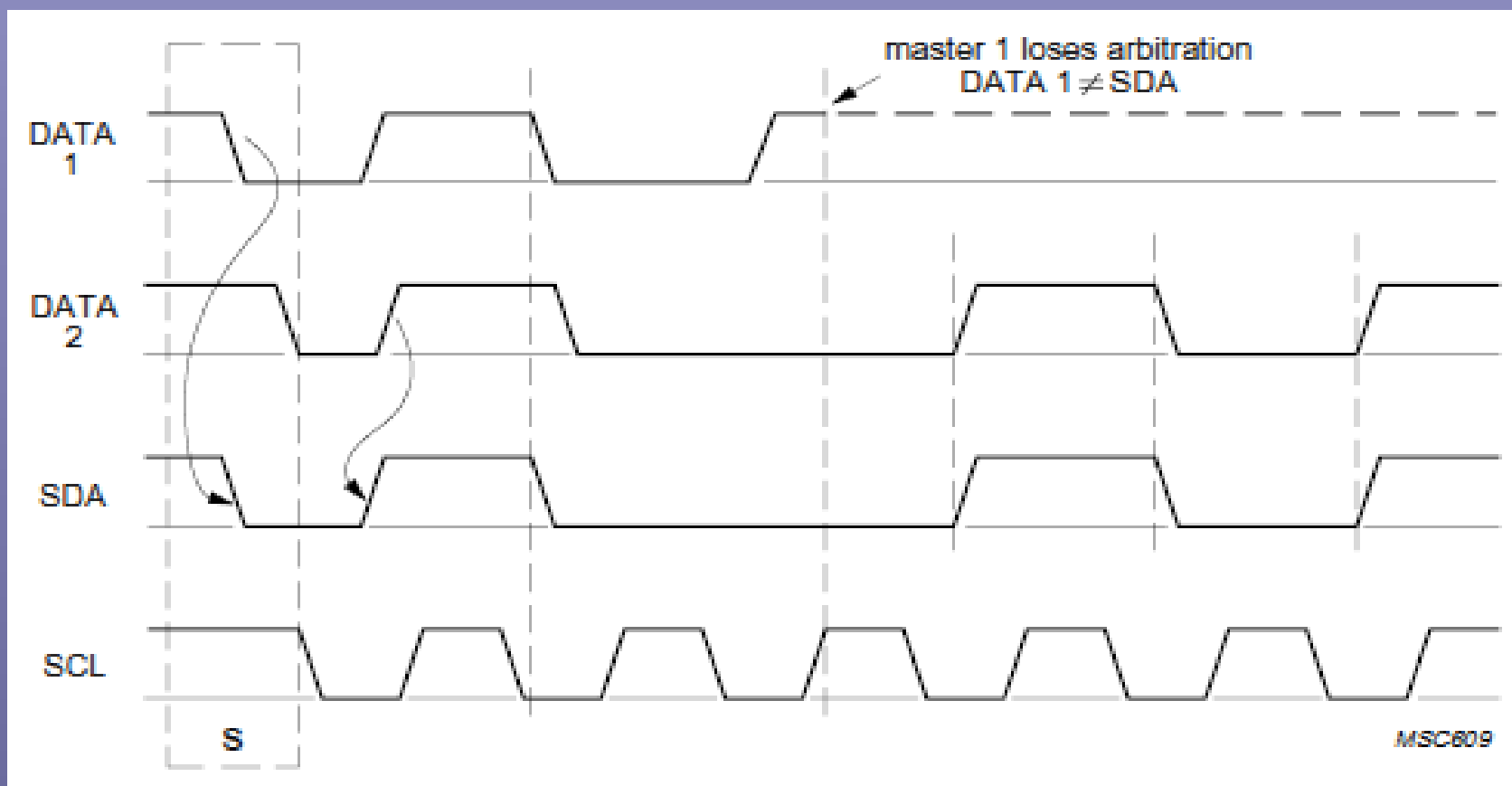
- master generuje SCL a provádí adresaci dat
- slave potvrzuje přijetí, může pozdržet hodiny



Sběrnice pro komunikaci mezi IO

Arbitrace na sběrnici I²C

- 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
- podmožinu příkazů používá i ATAPI

SAS – Serial Attached SCSI

- sériový přenos, dvoubodový spoj
- 300 MB/s (1bit, 3GHz)

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
- stromová struktura – koncová zařízení / hub
- plug-and-play

IEEE 1394 – FireWire (i.Link)

- synchronní i asynchronní
- 100/200/400 Mb/s (IEEE 1394b: ~800Mb/s)
- paketový přenos

Sběrnice pro připojování externích periférií

Standardizace do posledního detailu...

