

Organizacija i arhitektura računala

Poglavlje 3

Sistemske sabirnice

Koncept “programa”

- ⌘ Ožičeni programski sustavi *nisu fleksibilni*
- ⌘ Hardware za općenitu namjenu može obavljati raznorazne poslove, uz ispravno programiranje upravljačkih signala upravljačke jedinice
- ⌘ Umjesto ponovnog reožičenja dodan je novi skup kontrolnih signala / novi set (sekvenci) instrukcija

Što je to program?

- ⌘ To je niz/sekvenca koraka obrade nad podacima iz glavne memorije
- ⌘ Za svaki korak obavlja se ili aritmetička ili logička operacija
- ⌘ Za svaku operaciju potreban je i djeluje različiti skup upravljačkih signala

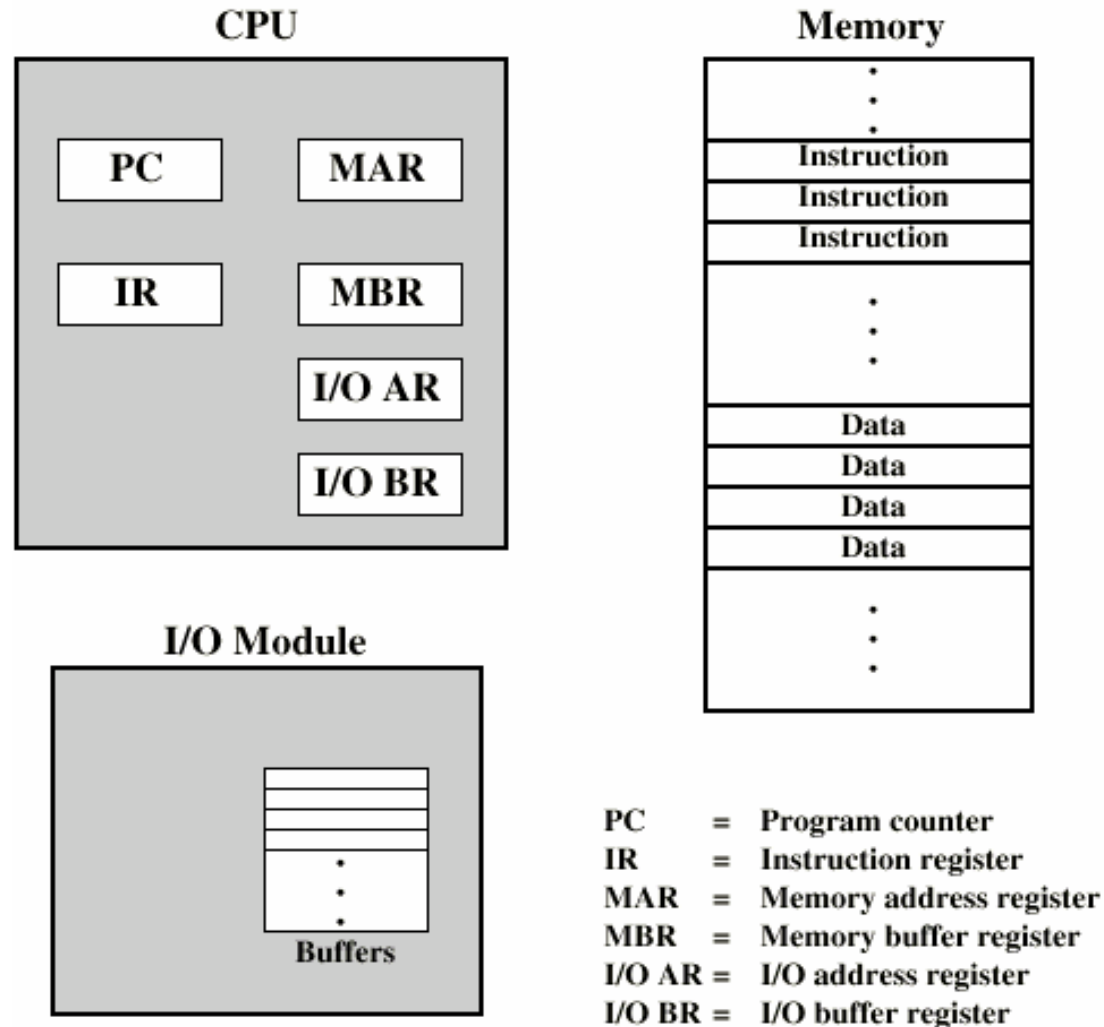
Funkcija upravljačke jedinice

- ⌘ Svakoj operaciji je dodijeljen jedinstveni kôd
 - ☐ npr. ADD, MOVE, PUSH, JMP
- ⌘ Hardware i dekoderska jedinica prima i dekodira kôd te upravlja upravljačkim signalima upravljačke jedinice
- ⌘ i... imamo **RAČUNALO!**

Komponente / sastavni dijelovi

- ⌘ Upravljačka i ALU jedinica čine CPU
- ⌘ Podaci i instrukcije trebaju se unijeti u sustav a rezultati obrade se prosljeđuju van tog sustava
 - ☒ Ulazi/izlazi
- ⌘ Potrebna je “privremena” memorija za pohranu programskog koda i rezultata obrade
 - ☒ Glavna memorija

Računalne komponente: “Top Level” prikaz

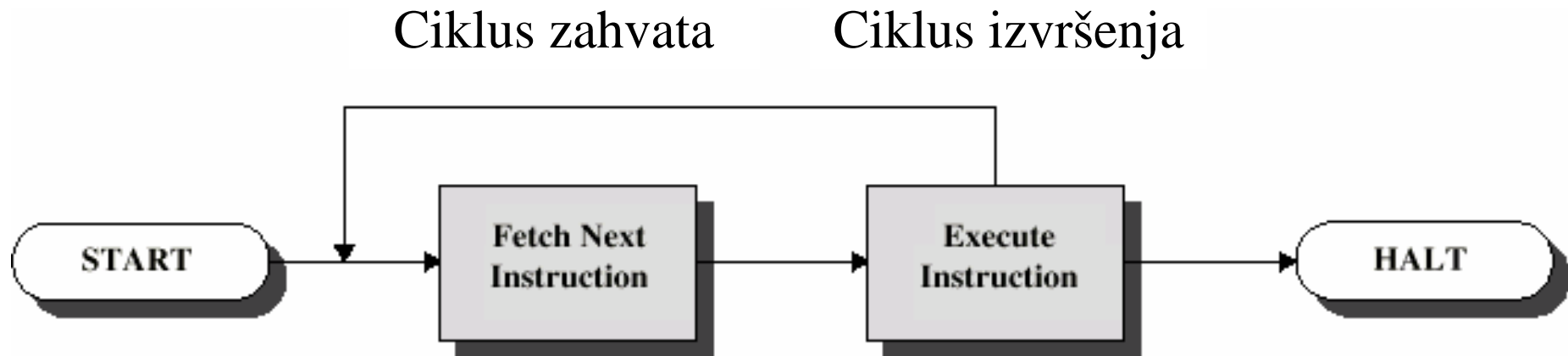


Instrukcijski ciklus...

⌘ Dvo-koračni ciklus:

☑ Zahvati/Fetch

☑ Izvrši/Execute



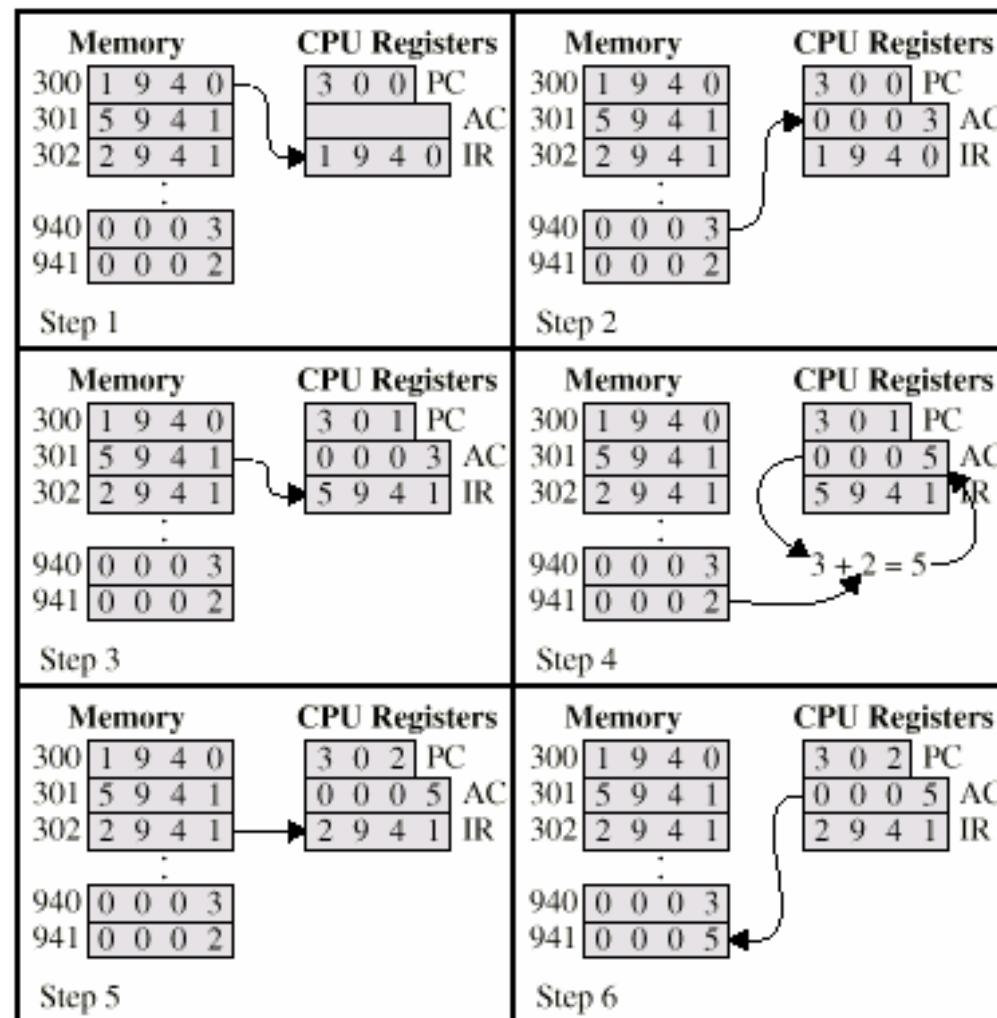
Ciklus zahvata - Fetch Cycle

- ⌘ Programski brojač (PC) sadrži adresu zahvata slijedeće instrukcije
- ⌘ Procesor zahvaća instrukcije iz dijela memorije adresiranog od strane PC-a
Increment PC
 - ☒ Ukoliko to drugačije (programski) nije rečeneo
- ⌘ Instrukcija se učitava u "Instrukcijski registar" / IR
- ⌘ Procesor (dekoderska jedinica) interpretira značenje instrukcije i izvršava zadanu akciju

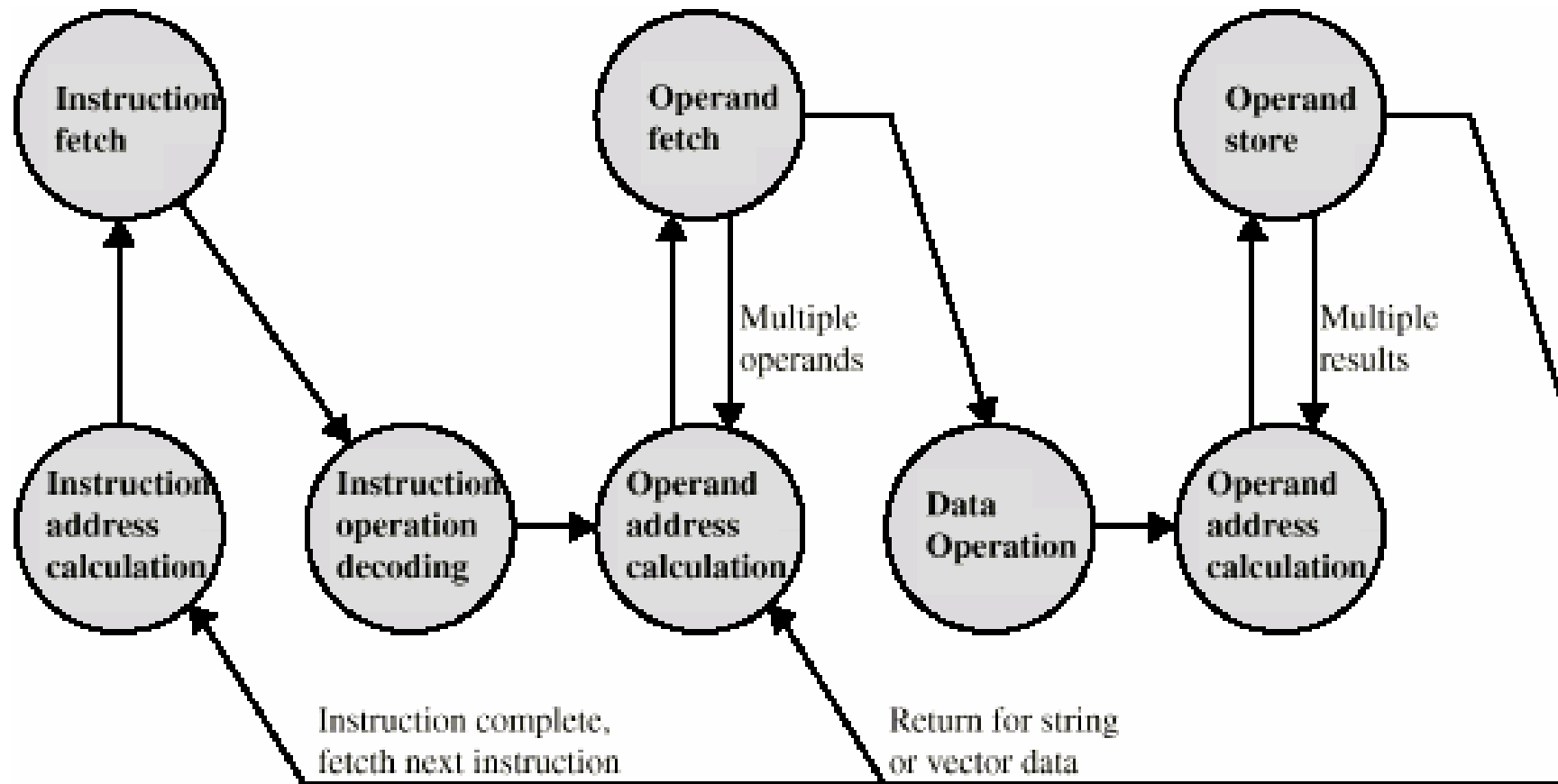
Ciklus izvršenja - Execute Cycle

- ⌘ Procesor↔memorija
 - ☒ prijenos podataka između procesora i memorije
- ⌘ Procesor↔I/O
 - ☒ prijenos podataka između procesora i I/O jedinice
- ⌘ Procesiranje/obrada podataka
 - ☒ Primjena aritmetičkih i logičkih operacija nad podacima
- ⌘ Upravljanje - Control
 - ☒ Izmjena sekvencijskog niza zadanih operacija/prog. instrukcija
 - ☒ npr. skokovi (jump)
- ⌘ Kombinacija gore navedenih operacija

Primjer izvršenja programa



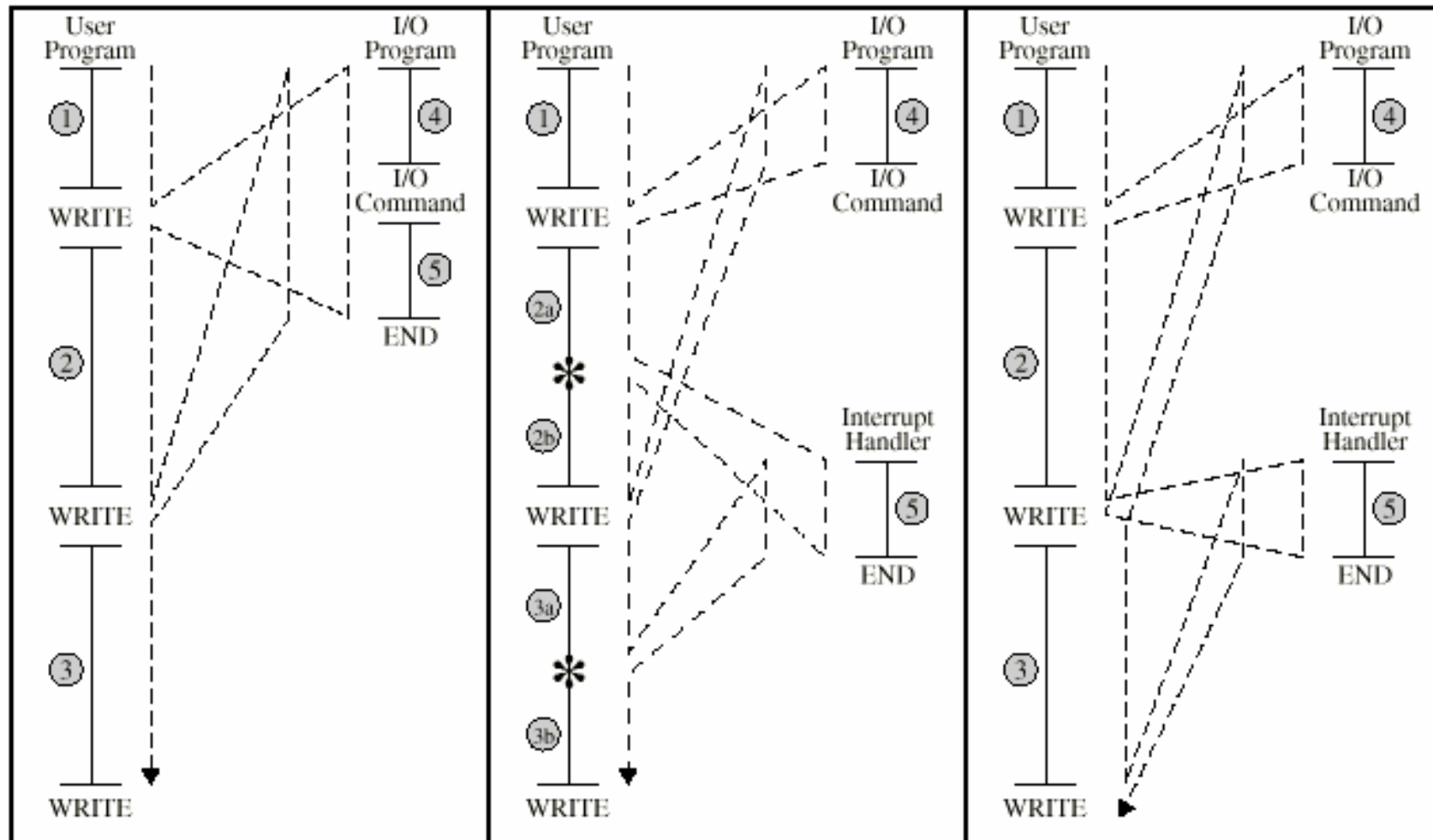
Instrukcijski ciklus - Dijagram stanja



Programski prekid - Interrupts

- ⌘ To je mehanizam pomoću kojega ostale jedinice računala (npr. I/O jedinica) može prekinuti normalni slijed sekvencijalnog izvršavanja programa
 - ⊞ npr. programski prelje/overflow, dijeljenje sa nulom
- ⌘ Vremenski sklopovi - Timer
 - ⊞ Generirani od strane internog procesorskog timera
 - ⊞ Korišteni u pre-emptive multi-tasking sustavima
- ⌘ I/O
 - ⊞ od strane I/O upravljača/kontrolera
- ⌘ Hardware/sklopovska pogreška
 - ⊞ npr. memorijska greška pariteta

Upravljanje programskim tijekom - Program Flow Control



(a) No interrupts

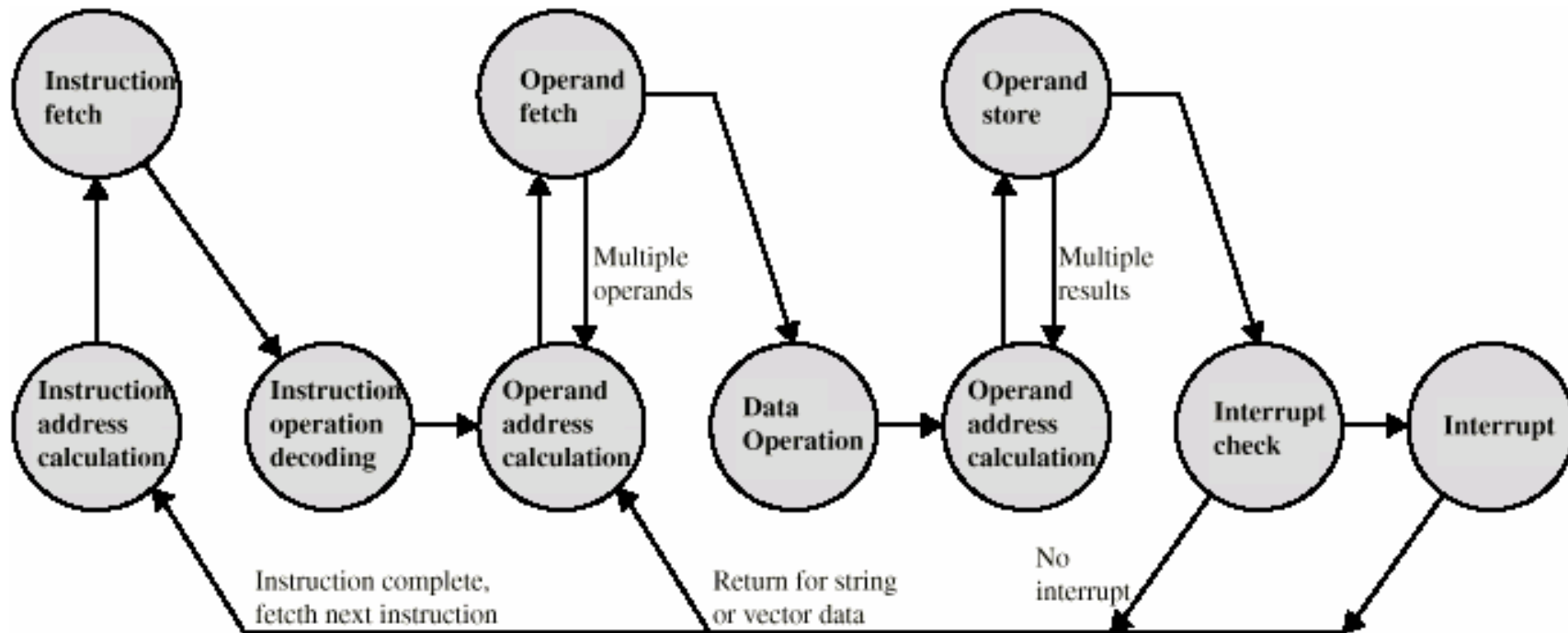
(b) Interrupts; short I/O wait

(c) Interrupts; long I/O wait

Ciklus programskog prekida - Interrupt Cycle

- ⌘ Dodano instrukcijskom ciklusu – dio instrukcijskog ciklusa
- ⌘ Procesor provjerava pojavnost interrupta
 - ☒ pojava interrupta daje se na znanje procesoru putem upravljačkih interrupt signala
- ⌘ Ako nema prog. prekida dalje se nastavlja sa fazom zahvata slijedeće instrukcije
- ⌘ Ako se dogodi prekid:
 - ☒ Obustavlja se izvršenje trenutnog programa
 - ☒ Pohranjuje se stanje trenutnih operacija obrade
 - ☒ PC se postavlja na pokazivanje adrese programa u memoriji zaduženog za obradu prog. prekida
 - ☒ Procesiranje/obrada prekidne rutine
 - ☒ Ponovno uspostavljanje stanja operacija obrade programa prije nastanka prog. prekida

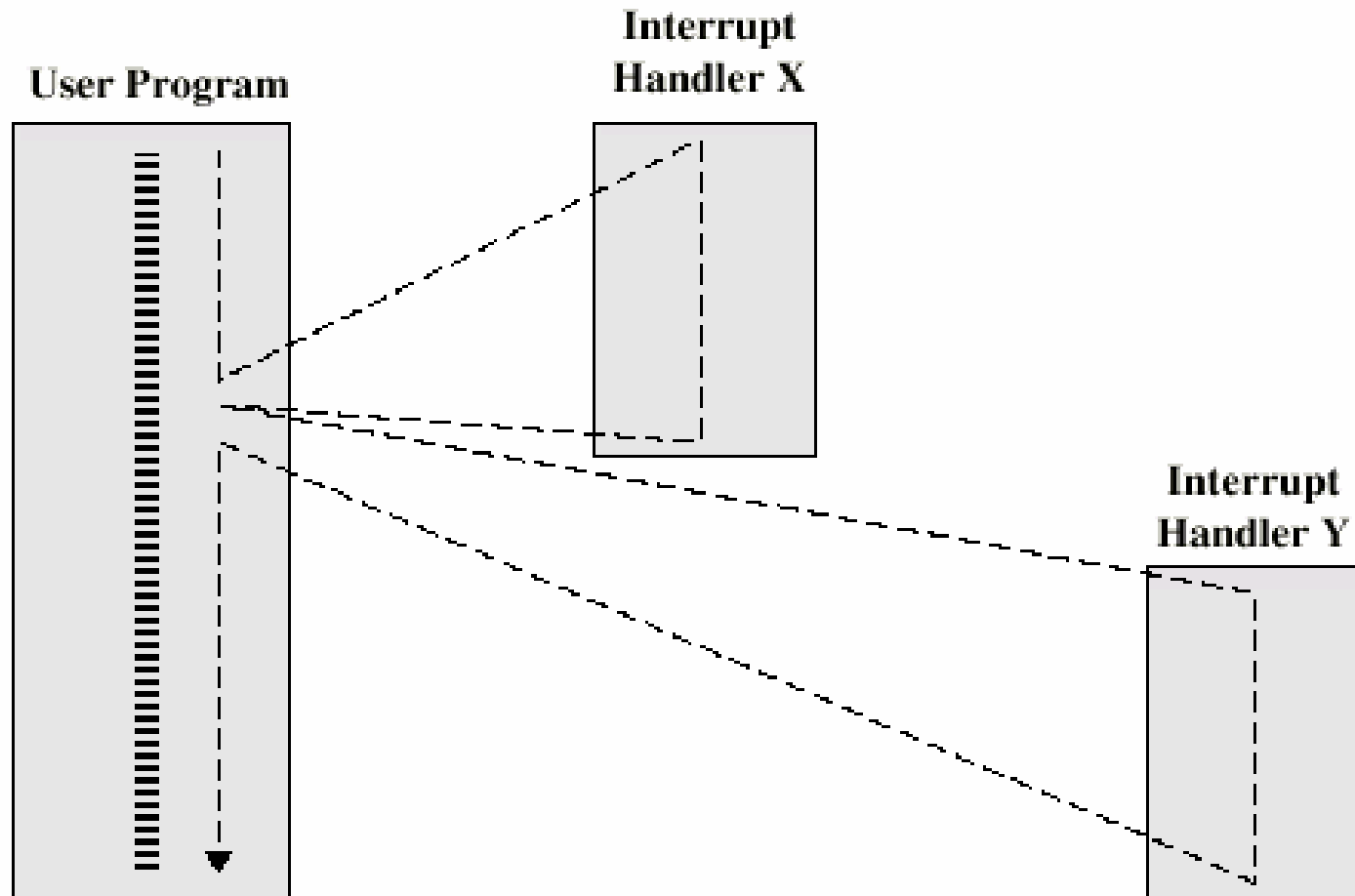
Instrucijski ciklus (sa prog. prekidom) - Dijagram stanja



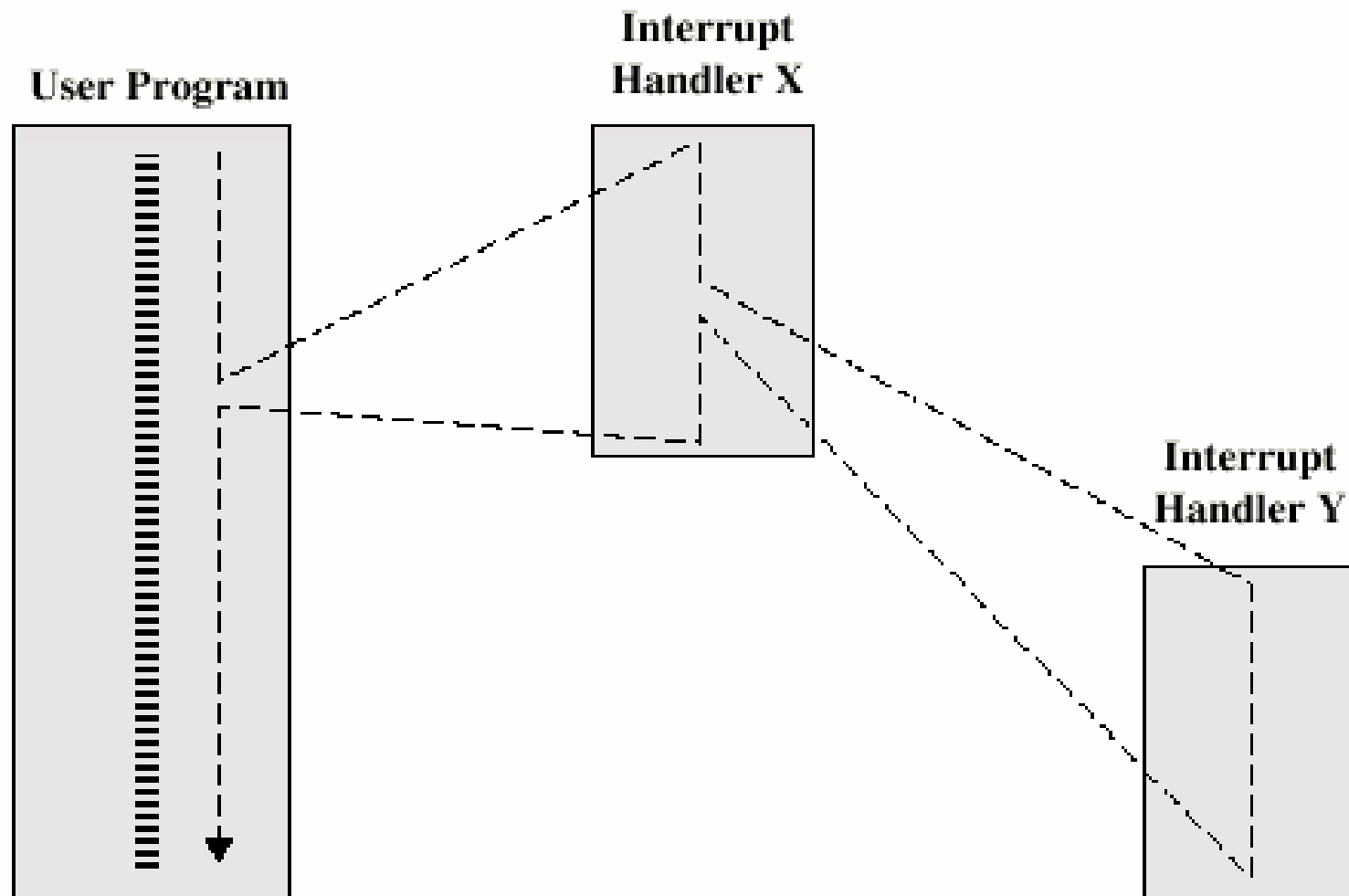
Višestruki programski prekidi...

- ⌘ Onemogućavanje djelovanja prog. prekida
 - ☒ Procesor će ignorirati pojavu ostalih prekida dok obrađuje trenutni prog. prekid
 - ☒ Pojava prekida se bilježi po redu i u nizu te se niz pojavnosti prekida provjerava tek kada se izvrši obrada prvog prekida u nastalom nizu
 - ☒ Obavlja se obrada prekida onim redosljedom kako su se pojavili u prekidnom nizu
- ⌘ Definiranje prioriteta
 - ☒ Prekidi sa niskom razinom značajnosti mogu se prekinuti prekidima većeg značaja
 - ☒ Kada se obradi prekid većeg značaja, procesor se vraća na obradu prekida nižeg značaja/manje važnosti.

Višestruki prekidi – Sekvencijalni niz izvršenja prekida



Višestruki prekidi – Nested/ugniježđeno izvršavanje prekida



Povezivanje/poveznice...

- ⌘ Sve funkcionalne jedinice računala moraju biti povezane
- ⌘ Koriste se razne vrste povezivanja za razne vrste jedinica
 - ☑ Povezivanje memorijskog podsustava
 - ☑ Ulazno/izlazni, I/O, podsustav
 - ☑ CPU

Memorijsko sučelje/sabirnica

- ⌘ Prima i šalje podatke
- ⌘ Prima adrese mem. lokacija
- ⌘ Primanje upravljačkih signala
 - ☑ Čitanje/Read
 - ☑ Pisanje/Write
 - ☑ Vremensko reosvježavanje/Timing

Sučelja I/O jedinica(1)

⌘ Gledano od strane računala ovo sučelje je slično memorijskom sučelju

⌘ Izlaz(i)

☑ Prima podatke od računala

☑ Šalje podatke periferiji računala

⌘ Ulazi

☑ Prima podatke od periferije

☑ Šalje podatke računalu

Sučelja I/O jedinica(2)

- ⌘ Primanje upravljačkih signala od računala
- ⌘ Slanje upravljačkih signala periferiji
 - ☑ npr. zavrti ploče diska, otvori ladicu CD/DVD-a
- ⌘ Prima podatke od računala
 - ☑ npr. koji je to broj porta za identifikaciju periferije
- ⌘ Slanje signala prog. prekida (upravljanje prog. prekidima)

CPU sučelje

- ⌘ Čitanje instrukcija i podataka
- ⌘ Pisanje podataka nakon procesiranja istih
- ⌘ Slanje upravljačkih signala ostalim jedinicama računala
- ⌘ Primanje, registriranje i reagiranje na pojavu prog. prekida

Sabirnice - Busevi

- ⌘ Postoji bezbroj mogućih načina i sistema povezivanja
- ⌘ Jednostruke i višestruke sabirničke BUS strukture su najčešće u uporabi
 - ☒ npr. Control/Address/Data sabirnica (PC)
 - ☒ npr. Unibus (DEC-PDP)

Sto je to sabirnica/bus?

- ⌘ Komunikacijski put kao poveznica dva ili više uređaja
- ⌘ Uobičajena je tehnika “prozivanja” uređaja
- ⌘ Često su funkcionalno grupirane
 - ☒ Broj komunikacijskih kanala u jednoj sabirnici
 - ☒ npr. 32 bit data bus čini 32 odvojena 1-bit kanala
- ⌘ Linije napajanja u grupiranoj sabirnici nisu stvarnim funkcionalnim dijelom sabirnice u funkciji razmjene podataka

Podatkovna sabirnica...

⌘ Prenosi podatke

☑ Treba zapamtiti da na ovom nivou prijenosa informacija ne postoji razlika između "podataka" i "instrukcija" – sve su to podaci koji se prenose

⌘ Ključni faktor performansi predstavlja "širina" sabirnice

☑ 8, 16, 32, 64 bit

Adresna sabirnica

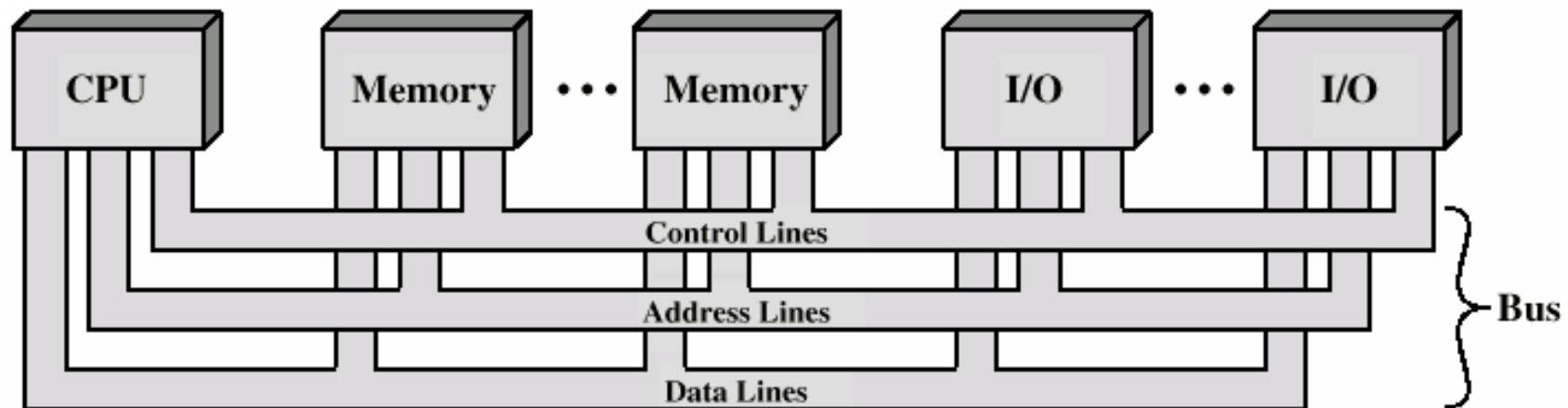
- ⌘ Identificira izvorište odakle se žele prenijeti podaci – identifikacija drugog kom. uređaja
- ⌘ npr. Procesor treba pročitati instrukciju/podatak iz memorijske lokacije zadane adresnom sabirnicom
- ⌘ Širina sabirnice određuje najveći memorijski kapacitet računalnog sustava
 - ☒ npr. 8080 ima 16 bit adresnu sabirnicu i omogućava 64kB adresnog prostora

Upravljačka/kontrolna sabirnica

⌘ Prenosi upravljačke i vremenski periodičke/clock signale i informacije

- ☑ signal za čitanje/pisanje memorije
- ☑ zahtjevi za programskim prekidima
- ☑ signali takta / clock signali

Strukturna shema sabirnice...



Kako prepoznati sabirnicu?

⌘ Kako ustvari sabirnica izgleda/fizički?

- ☑ Paralelni vodovi na matičnoj ploči
- ☑ Ribbon kabele / plosnati višezični kabele
- ☑ Višepolne priključnice na matičnoj ploči / slotovi
 - ☒ npr. PCI, PCI-Express, ISA, DDR, SDRAM,...
- ☑ To je ustvari skupina od mnogo žica...

Problemi korištenja samo jedne sabirnice...

⌘ Puno uređaja spojenih na jednoj sabirnici izravno vodi do:

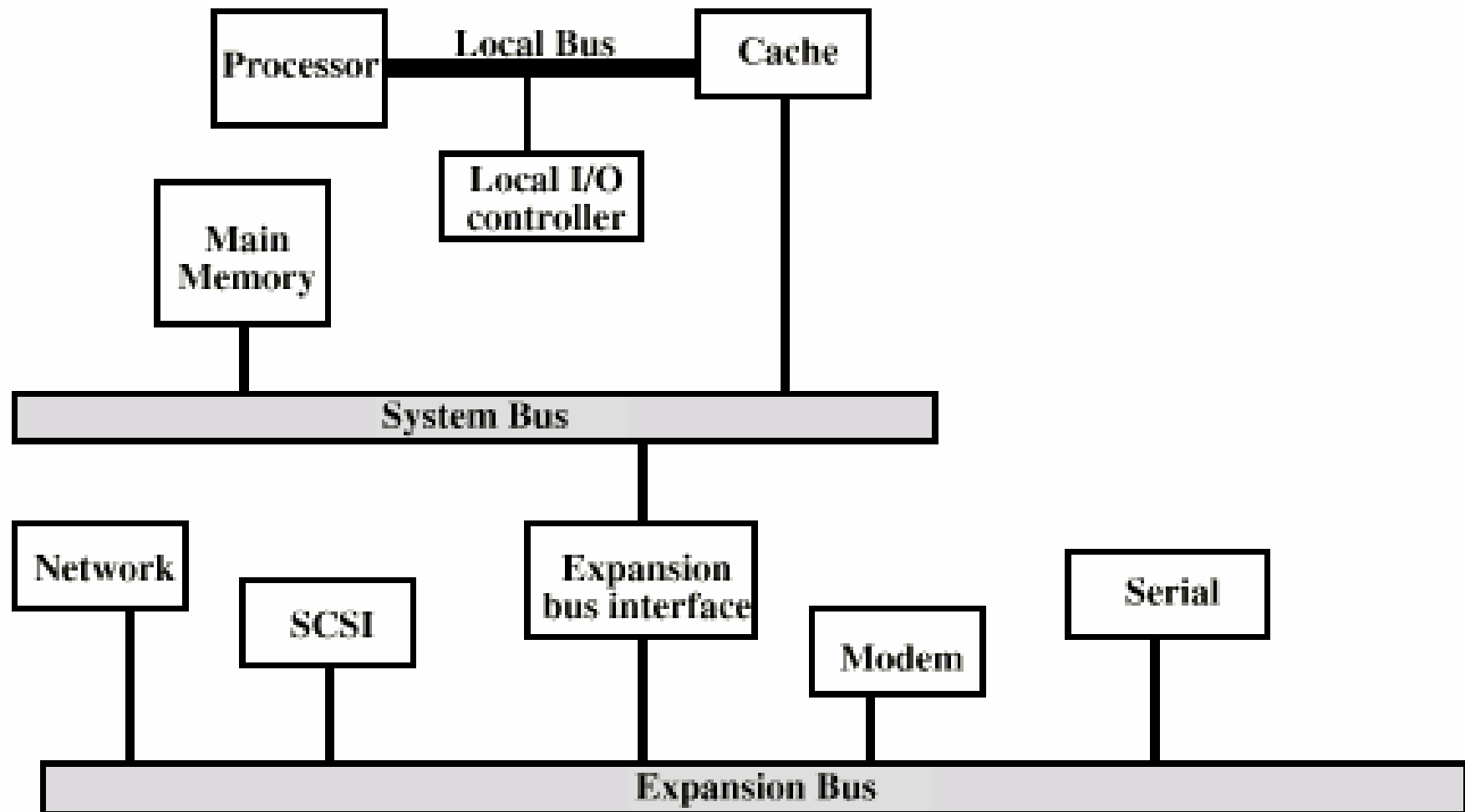
⊠ Kašnjenja u propagaciji signala

⊠ Dugački vodovi značajno i negativno utječu na performanse sabirnice,

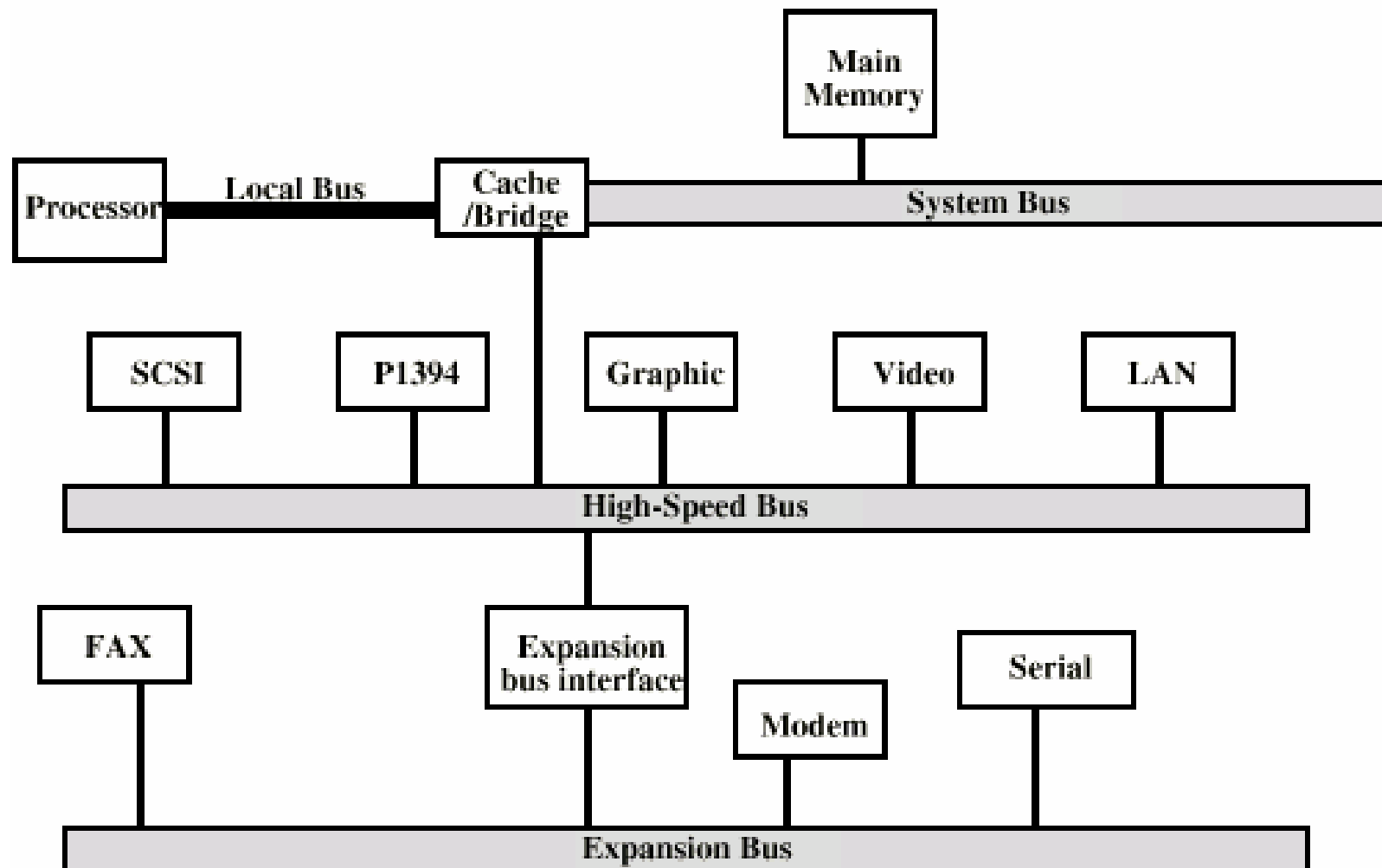
⊠ Približavanje max. mogućem kapacitetu prijenosa podataka putem ove sabirnice

⌘ Većina sustava koristi višestruke sabirnice da bi zaobišla upravo navedene probleme

Tradicionalni topologija sabirnice (ISA) (sa priručnom memorijom)



Sabirnica visokih performansi



Vrste sabirnica...

⌘ Posve određene / Dedicated...

- ☑ Odvojene data i adresne linije

⌘ Multipleksirane...

- ☑ Linije dijele funkcionalne uloge
- ☑ Address valid ili data valid upravljača linija
- ☑ Prednost – manje komunikacijskih linija
- ☑ Nedostaci
 - ☒ Složeno upravljanje takvom sabirnicom
 - ☒ Ultimate performance

Arbitracija pristupa sabirnici...

- ⌘ Više od jednog uređaja može kontrolirati sabirnici i njen rad
- ⌘ npr. CPU i DMA kontroler
- ⌘ Samo jedan uređaj istovremeno može kontrolirati rad i pristupati sabirnici
- ⌘ Arbitracija/pregovor u kontroli pristupa sabirnici može biti:
 - ☑ Centralizirana ili
 - ☑ distribuirana

Centralizirana arbitracija

- ⌘ Samo jedan sklop ili uređaj kontrolira pristup sabirnici
 - ☑ Sabirnički kontroler
 - ☑ Arbiter
- ⌘ Može biti dio CPU-a ili zaseban sklop/uređaj

Distribuirana arbitracija

- ⌘ Svaki uređaj na sabirnici može preuzeti kontrolu nad njom
- ⌘ Upravljača logika kontrole pristupa sabirnice nalazi se na svakom uređaju spojenog na nju.

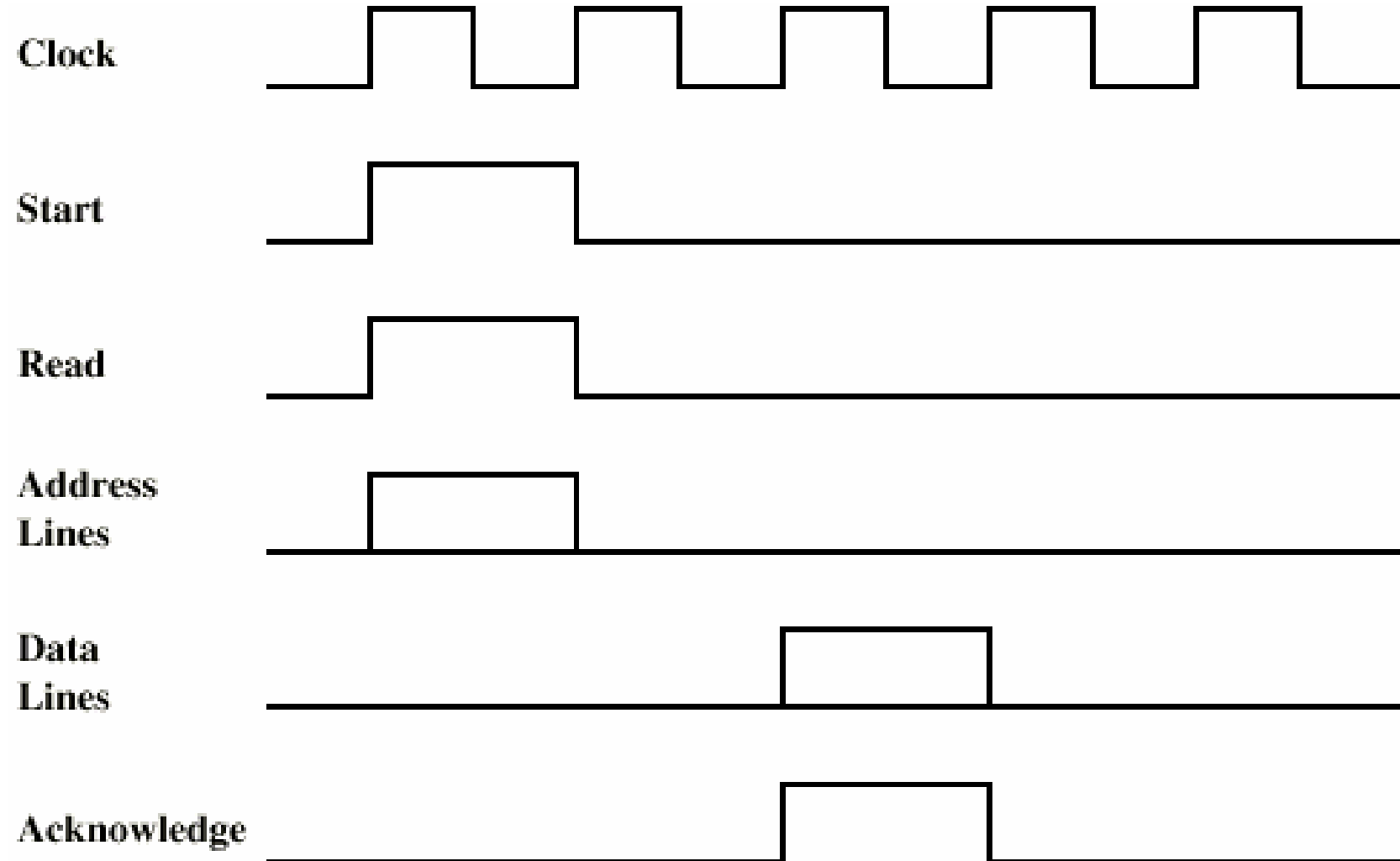
Vremensko upravljanje - Timing

⌘ Koordinacija događaja na sabirnici

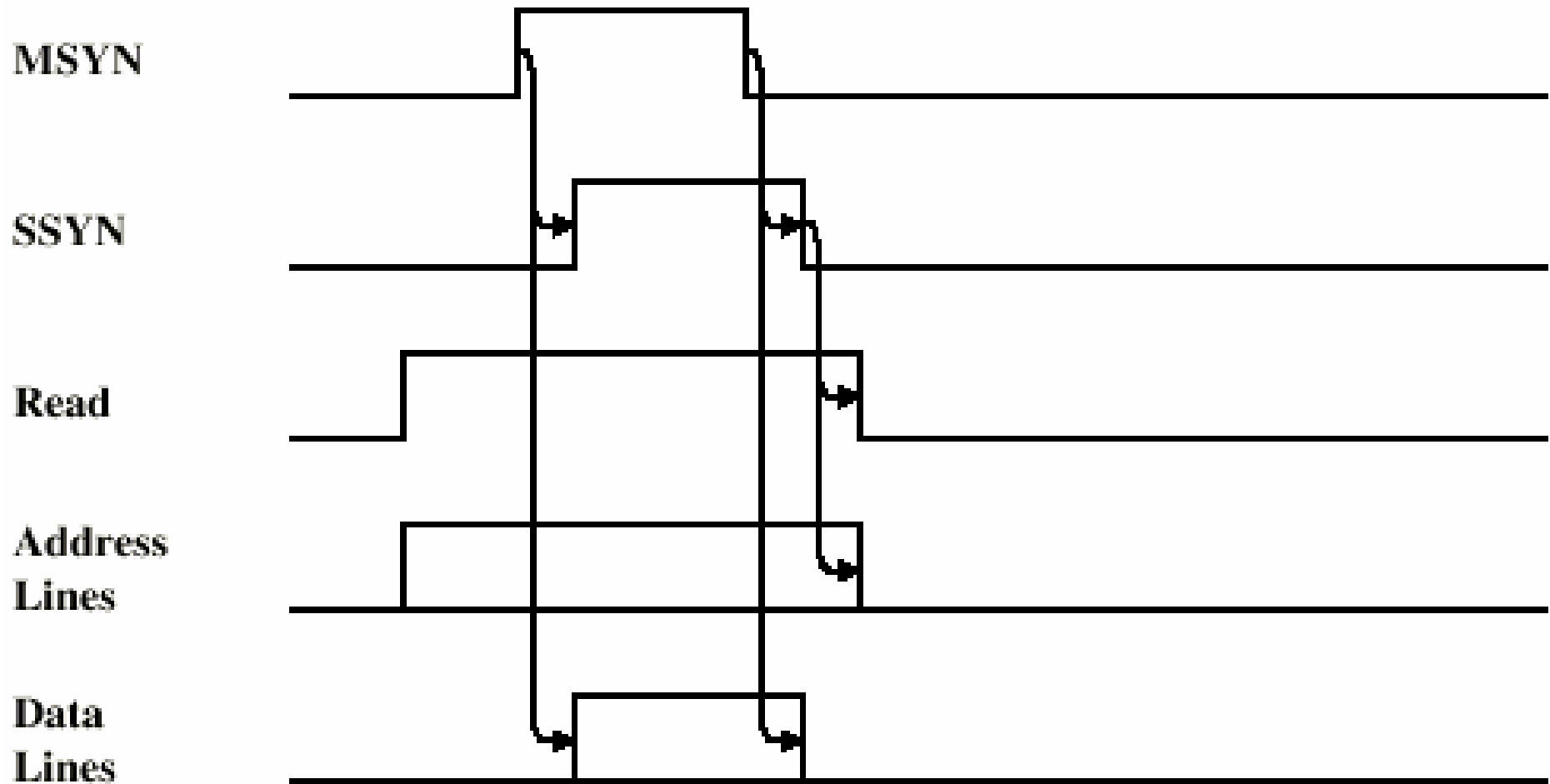
⌘ Sinkrone sabirnice

- ☒ Pojavnost događaja određena je clock/taktnim signalom
- ☒ Kontrolni dio sabirnice sadržava signal takta
- ☒ Prijelaz lokičkih 1-0 predstavlja takt sabirnice
- ☒ Svi uređaji mogu primiti takt signal
- ☒ Uobičajena je sinkronizacija uređaja na rastući brid takt signala
- ☒ Uobičajeno jedan događaj traje jedan ciklus takt signala

Sinkroni timing diagram



Asinkroni timing diagram



PCI Sabirnica

- ⌘ Peripheral Component Interconnection
- ⌘ razvio je Intel
- ⌘ 32 ili 64 bit
- ⌘ 50 linija

PCI sabirničke linije (nužne)

⌘ Sistemske linije

- ☒ uključujući takt i reset signal

⌘ Adresne i podatkovne linije

- ☒ 32 multipleksirane linije za adrese i podatke
- ☒ Linije za prog. prekide i linije za validaciju podataka/adresa1

⌘ Linije za kontrolu sučelja

⌘ Arbitracija

- ☒ Nije distribuirana – centralizirana je
- ☒ Izravno povezivanje na PCI arbitarski sklop

⌘ linije za oznaku pojavnosti pogreški – error lines

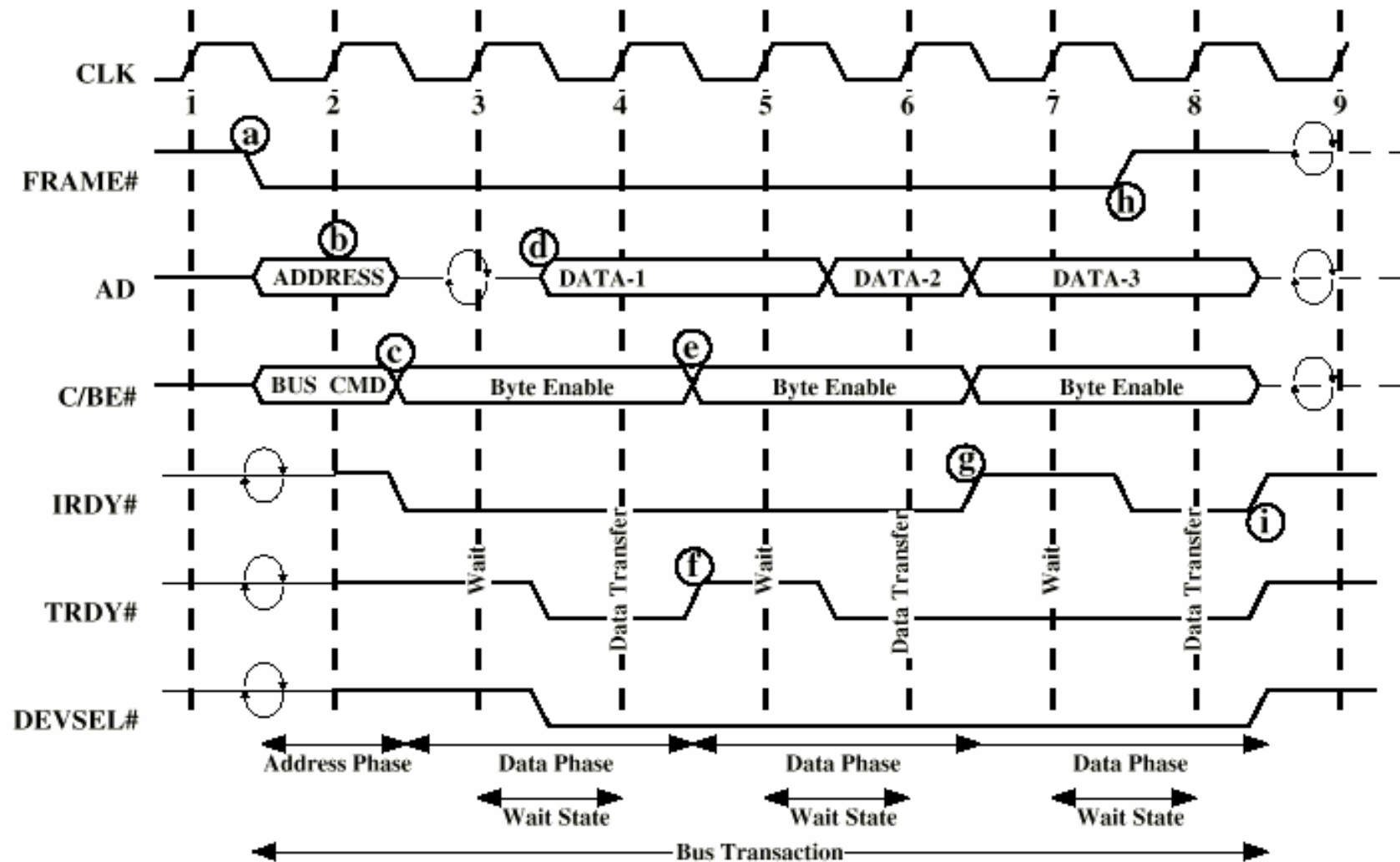
PCI sabirničke linije (opcionalne)

- ⌘ Linije prog. prekida
 - ☒ Ne dijele se između uređaja
- ⌘ Podrška za priručnu memoriju
- ⌘ 64-bit proširenje sabirnice
 - ☒ dodatne 32 linije
 - ☒ vremenski multipleksirane
 - ☒ 2 dodatne linije za odobrenje uređaju korištenja 64bit načina prijenosa podataka
- ⌘ JTAG/Boundary Scan
 - ☒ Za potrebe testiranja uređaja

PCI Naredbe

- ⌘ Transakcije između inicijatora (master) i odredišta (target)
- ⌘ Master uređaj preuzima kontrolu na sabirnicom
- ⌘ Određivanje tipa transakcije/razmjene
 - ☒ npr. čitanje I/O jedinice, ili pisanje,...
- ⌘ Adresna faza – samo jedna
- ⌘ Jedna ili više podatkovnih faza

PCI timing diagram čitanja..



PCI sabirničko pregovaranje/arbitracija...

