

OPERATIVNI SISTEMI

Slajdovi su kreirani na osnovu knjige "Operativni sistemi, principi unutrašnje organizacije i dizajna, 7. izdanje", William Stallings, CET, Beograd, 2013.

U/I upravljanje i raspoređivanje diska



Izvor:

www.youtube.com

Kategorije U/I uređaja

- Za komunikaciju sa korisnikom računara
 - Štampač, tastatura, miš, ...



Kategorije U/I uređaja

- Za komunikaciju sa hardverom računara
 - Diskovi, senzori, kontroleri, ...



Kategorije U/I uređaja

- Za komunikaciju sa udaljenim uređajem
 - Modemi, mrežne kartice

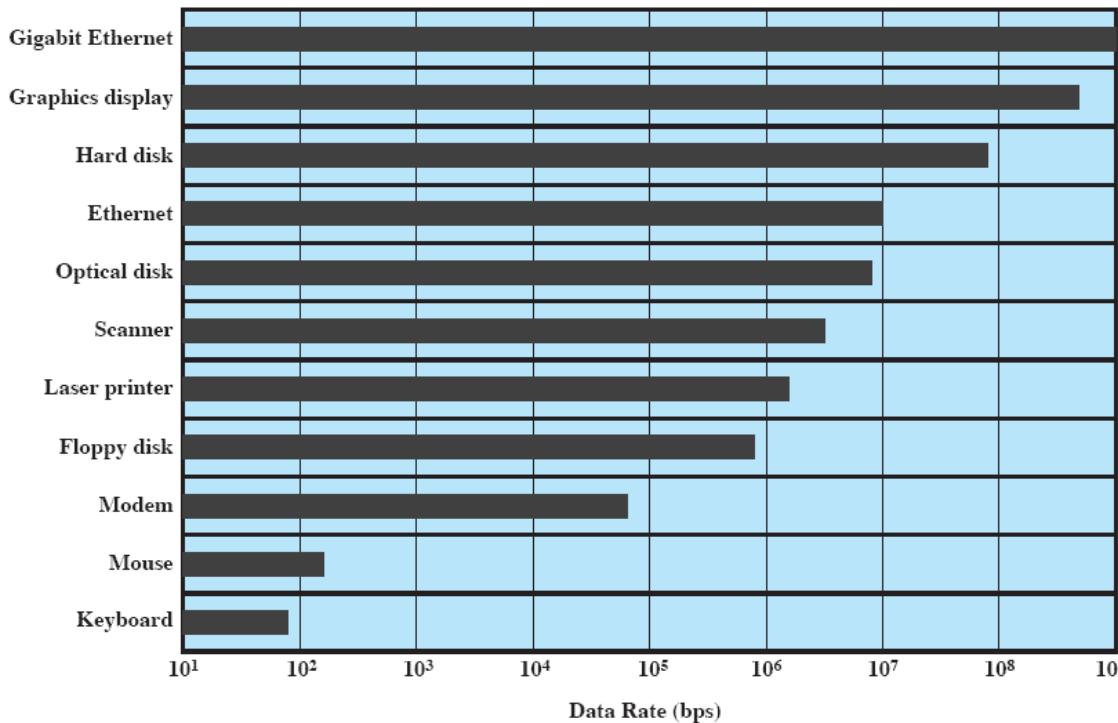


Razlike između U/I uređaja

- Brzina prenosa podataka
- Uloga u sistemu
- Složenost upravljanja
- Jedinica prenosa podataka
- Predstavljanje podataka
- Upravljanje greškama
- Način pristupa
- Podrška za upis i čitanje

Razlike između U/I uređaja

□ Brzina prenosa podataka



Razlike između U/I uređaja

- Uloga u sistemu
 - Uloga utiče na podršku OS za uređaj
- Složenost upravljanja
 - Neki uređaji zahtevaju jednostavno upravljanje
 - Miš, štampač
 - Neki uređaji zahtevaju složeno upravljanje
 - Disk
 - Od složenosti upravljanja zavisi složenost U/I modula OS za taj uređaj

Razlike između U/I uređaja

- Jedinica prenosa podataka
 - Orijentisani na blokove
 - Jedinica prenosa je blok
 - Orijentisani na tokove
 - Podaci se prenose kao tok bajtova (jedan po jedan)
- Predstavljanje podataka
 - Način kodovanja podataka
 - Format, parnost, ...
- Upravljanje greškama
 - Različit način reakcije na grešku
 - Različit skup kodova grešaka

Razlike između U/I uređaja

- Način pristupa
 - Sekvencijalni
 - Uredaj šalje/prima podatke u fiksnom redosledu
 - Npr. mrežni ruter, štampač
 - Proizvoljan
 - Moguće je uputiti zahtev uređaju da se pozicionira na bilo koju lokaciju na kojoj se nalaze podaci
 - Npr. Disk
- Podrška za upis i čitanje
 - Upis i čitanje (npr. disk)
 - Samo upis (npr. štampač)
 - Samo čitanje (npr. tastatura)

Razlike između U/I uređaja

- Potreban je jednoobrazan mehanizam za upravljanje raznolikim skupom uređaja

U/I hardver

- Uredaj razmenjuje podatke sa računarom putem nekog prenosnog medijuma
 - Kabel
 - Vazduh
- Uredaj je povezan sa računarom kroz
 - Port – Kada jedan uređaj koristi komunikacione kanale
 - Magistralu – Kada više uređaja deli isti skup kanala

U/I hardver

□ Kontroler

- Hardver koji upravlja portom, magistralom ili uređajem

□ *Host adapter*

- Kontroler u računaru zadužen za komunikaciju sa uređajem

□ *Device controller*

- Kontroler u uređaju namenjen za upravljanje uređajem

Komunikacija procesora sa U/I hardverom

- Prva varijanta
 - Procesor izvršava specijalnu U/I instrukciju koja direktno upisuje bitove u registre kontrolera (ili čita iz registara)
- Druga varijanta
 - (Memory mapped I/O)
 - Proces upisuje podatke u radnu memoriju
 - Lokacije u radnoj memoriji su namapirane na registre kontrolera
- Mogu i da se kombinuju ove dve varijante
 - Za jednostavnije upravljanje direktni pristup
 - Za veću količinu podataka druga varijanta

Registri U/I kontrolera

- *Data-in* registar
 - Iz njega procesor preuzima podatke
- *Data-out* registar
 - U njega procesor upisuje podatke za U/I uređaj
- Statusni registar
 - Informacija o statusu operacije za procesor
 - Da li je završena, kod greške, ...
- Upravljački registar
 - U ovaj registar procesor upisuje komandu za U/I uređaj

Organizacija U/I funkcije

□ Programirani U/I

- Procesor na zahtev procesa izdaje U/I komandu
- Dok U/I modul obavlja funkciju, proces je u zaposlenom čekanju (*busy waiting*) da se U/I operacija završi
- Proces stalno proverava da li je operacija završena



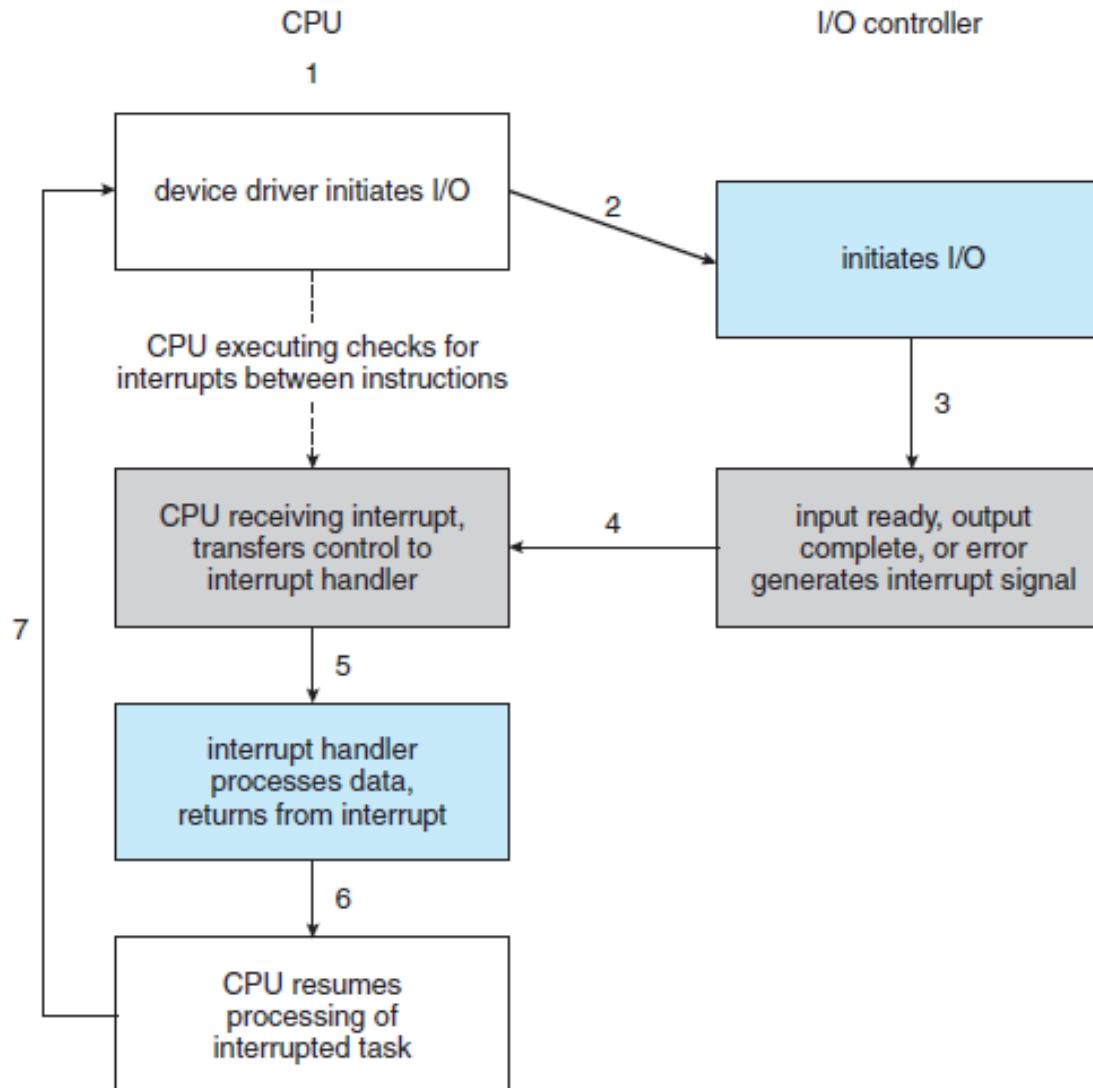
Izvor:

www.youtube.com

Organizacija U/I funkcije

- U/I pokretan prekidom
 - Procesor na zahtev procesa izdaje U/I komandu
 - Ako U/I instrukcija nije blokirajuća
 - procesor nastavlja da izvršava instrukcije procesa
 - Ako je U/I funkcija blokirajuća
 - Proses se stavlja u stanje blokiran
 - Raspoređuje se drugi proces
 - U/I uređaj postavlja prekid kada završi operaciju
 - Rutina za obradu prekida (ISR) se izvršava kao reakcija na završenu U/I operaciju

U/I pokretan prekidom



Isključivanje prekida

- U nekim situacijama potrebno je isključiti obradu prekida



Izvor:

www.youtube.com

Isključivanje prekida

- Prekidi se isključuju dok traje neka kritična operacija koja ne sme biti prekinuta
 - Najčešće dok se obrađuje prethodni prekid
- Procesor ima bit koji označava da li su prekidi uključeni
- Može se privilegovanom instrukcijom postaviti vrednost ovog bita
- Bit se odnosi samo na standardne prekide (*maskable interrupts*)

Isključivanje prekida

- Specijalni prekidi (*nonmaskable*) se ne mogu isključiti
 - Ovi prekidi se aktiviraju u slučaju fatalnih grešaka sistema



Izvor:

www.youtube.com

U/I pokretan prekidom

- Kada uređaj postavi prekid, potrebno je izvršiti rutinu za obradu prekida
- U drajveru je kod za obradu prekida
- Svaki drajver registruje svoju rutinu

Vektor prekida

- Kako procesor zna koji je uređaj postavio prekid?
- Pri signalizaciji prekida uređaj postavlja broj (adresu) koji ukazuje na tip prekida
- Vektor prekida
 - Niz koji sadrži adrese rutina za obradu prekida
 - Na osnovu adrese koju prekid postavi indeksira se element vektora prekida

Vektor prekida

- Obično vektor prekida ima manje elemenata nego što ima uređaja
 - Tada element vektora sadrži adresu liste obrađivača prekida
 - Skeniraju se svi elementi liste da se nađe odgovarajući obrađivač
- OS pri inicijalizaciji skenira uređaje i postavlja adrese obrađivača prekida u vektor prekida

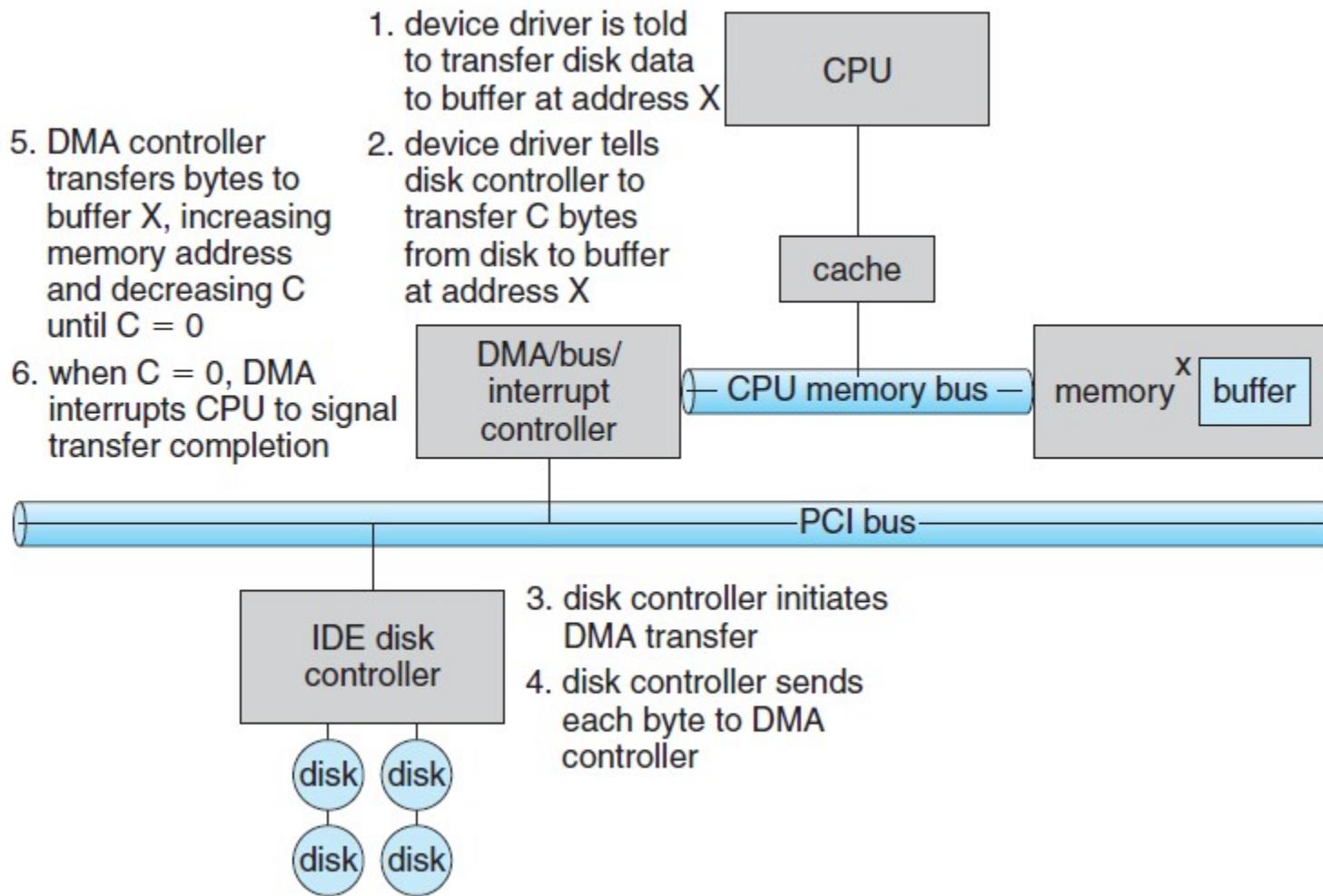
Nivoi prioriteta prekida

- Može da postoji više nivoa prioriteta prekida
- Obrada prekida nižeg prioriteta može biti prekinuta zbog obrade prekida višeg prioriteta

Organizacija U/I funkcije

- Direktan pristup memoriji (DMA)
 - Procesor može da delegira izvršavanje I/O operacija specijalizovanom procesoru – DMA kontroleru
 - DMA kontroler upravlja razmenom podataka između glavne memorije i U/I modula
 - Procesor šalje DMA modulu zahtev za prenos bloka podataka
 - Adresa sa koje se podaci prenose
 - Adresa na koju se podaci prenose
 - Broj bajtova za prenos
 - DMA modul postavlja prekid kada završi prenos

Koraci u DMA prenosu



Razvojne faze U/I funkcije

1. Procesor direktno upravlja U/I uređajem
2. Dodaje se kontroler (U/I modul)
 - Koristi se programirani U/I bez prekida
3. Ista konfiguracija kao pod 2, ali sa prekidima
 - Efikasniji sistem jer proces ne mora stalno da proverava završetak U/I operacije
4. U/I modul može direktno da upravlja memorijom preko DMA
 - Procesor može da obrađuje druge zadatke dok DMA prenosi ceo blok podataka

Razvojne faze U/I funkcije

5. U/I modul postaje poseban procesor
 - Čitav niz U/I aktivnosti obavlja U/I procesor bez uplitanja glavnog procesora
6. U/I modul dobija i sopstvenu memoriju
 - U/I modul je kao odvojen računar koji samostalno može da upravlja U/I uređajem uz minimalno angažovanje procesora

Ciljevi projektovanja U/I podistema

□ Efikasnost

- Većina U/I uređaja je izuzetno spora u odnosu na glavnu memoriju i procesor
- Multiprogramiranje dozvoljava da se drugi proces izvršava dok jedan proces čeka na U/I
- Razmenjivanje
 - Dobavlju se novi procesi sa diska da procesor ne bi bio besposlen
 - Najznačajnije je kako je realizovan U/I diska

Ciljevi projektovanja U/I

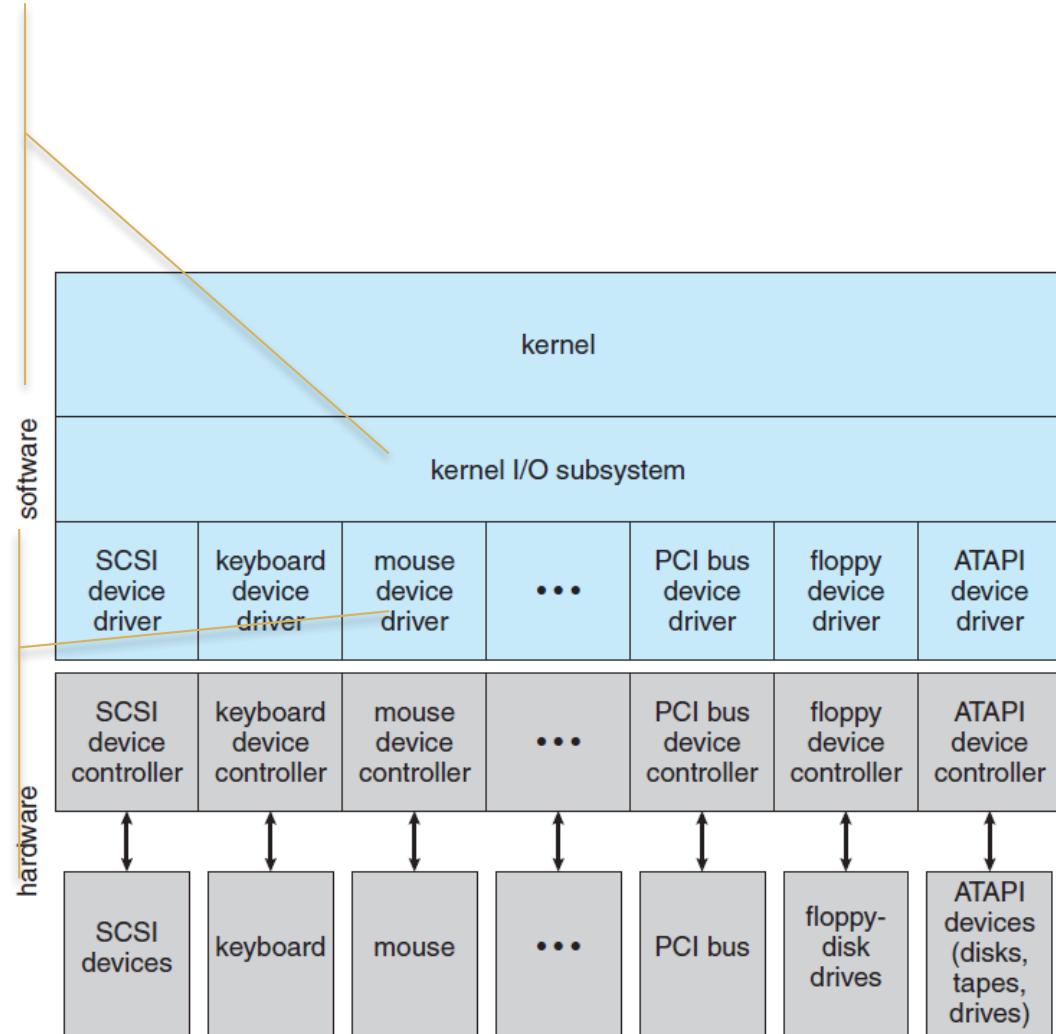
- Opštost
 - Radi jednostavnosti i obrade grešaka poželjno je upravljati svim U/I uređajima na jedinstven način
 - Sistem apstrahuje razlike između uređaja tako što identificuje zajedničke osobine
 - Zato OS koristi generičke funkcije za rad sa U/I
 - Operacije „piši“, „čitaj“, „otvorи“, „zaključaj“
 - Rutine nižeg nivoa obavljaju operacije nad konkretnim uređajem
 - Drajveri realizovani specifično za svaki uređaj implementiraju generičke funkcije za konkretan uređaj

Hijerarhijska struktura

- Zato se U/I funkcije organizuju hijerarhijski
 - Rutine nižeg nivoa sakrivaju detalje uređaja
 - Sistem se organizuje po slojevima
 - Svaki sloj
 - se oslanja na sledeći sloj da bi izvodio jednostavnije funkcije
 - Obezbeđuje servise sloju iznad sebe
 - Promene u jednom sloju ne bi trebalo da zahtevaju izmene drugih slojeva
 - Zato ne treba menjati kod OS da bi se podržao novi uređaj
 - Dovoljno je dodati drajver za uređaj
 - Drajver se implementira specifično za svaki OS

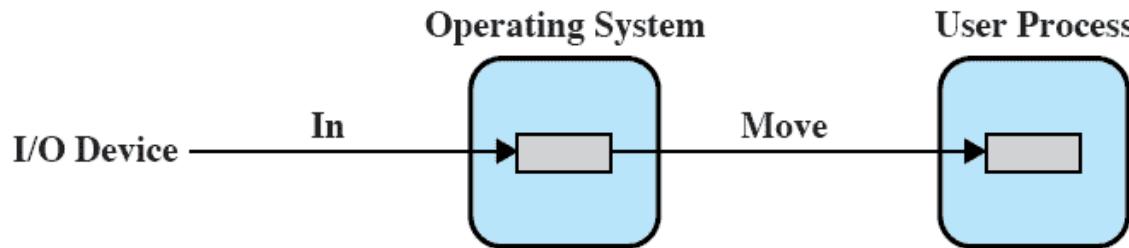
Organizacija slojeva u U/I podsistemu

- Radi sa uređajem kao logičkim resursom
 - Ne brine se o detaljima stvarnog uređaja
 - Izvršava generičke komande „upiši“, „pročitaj“, ...
-
- Stvarno čekanje u redovima i raspoređivanje U/I operacija
 - Nivo softvera koji stvarno komunicira sa hardverom uređaja



U/I baferovanje

- Bafer je prostor u memoriji u koji se skladište podaci koji se prenose od/ka U/I uređaja



- Proces ne mora da čeka na spore U/I operacije
 - Pri U/I upisu, proces postavi sve podatke odjednom u bafer, a kontroler ih šalje ka uređaju
 - Pri U/I čitanju, najpre se svi podaci prihvate u bafer, nakon čega im proces pristupa

U/I baferovanje



Izvor:

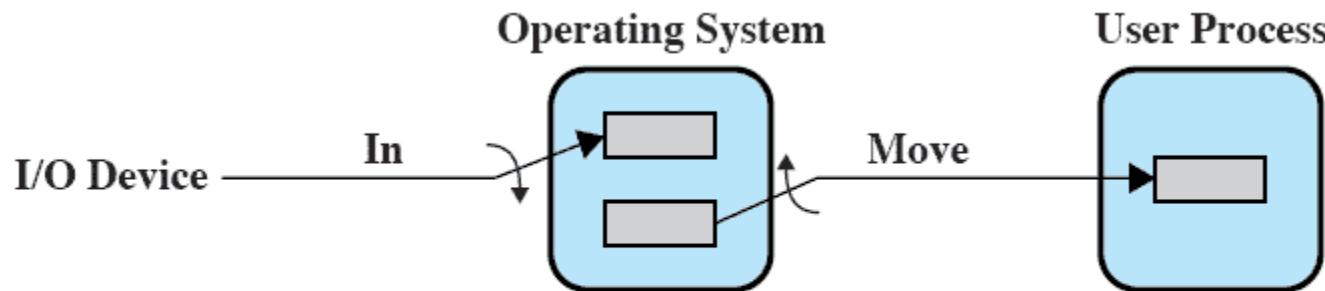
www.youtube.com

U/I baferovanje

- Pošto U/I operacija koristi podatke iz bafera, pri upisu podataka nije problem ako proces promeni podatke nakon izdavanja U/I operacije
- Stranica procesa ne mora u toku U/I operacije ni da bude u radnoj memoriji
 - Može biti zamjenjena na disk

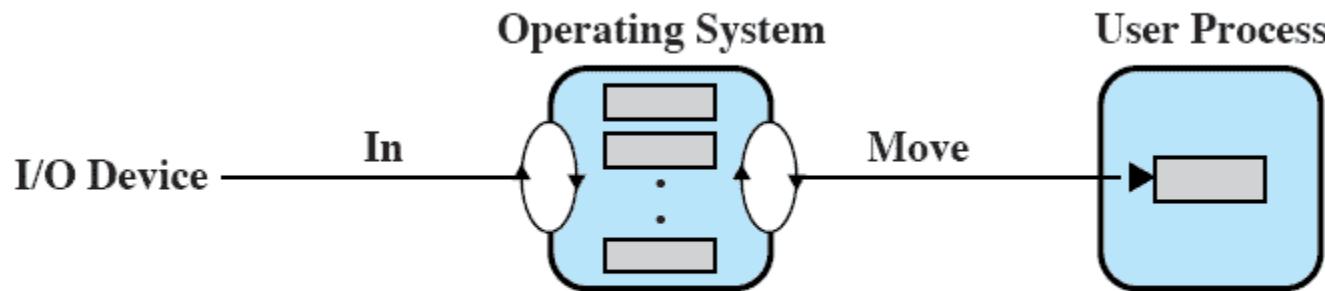
Dvostruki bafer

- Uvode se dva bafera
- Proces učitava podatke iz jednog bafera
- Za to vreme OS učitava sledeći blok u drugi bafer
- Onda proces pređe na drugi bafer, a OS se vrati na prvi i tako redom
- Slično (samo obrnuto) kod upisa podataka



Kružni bafer

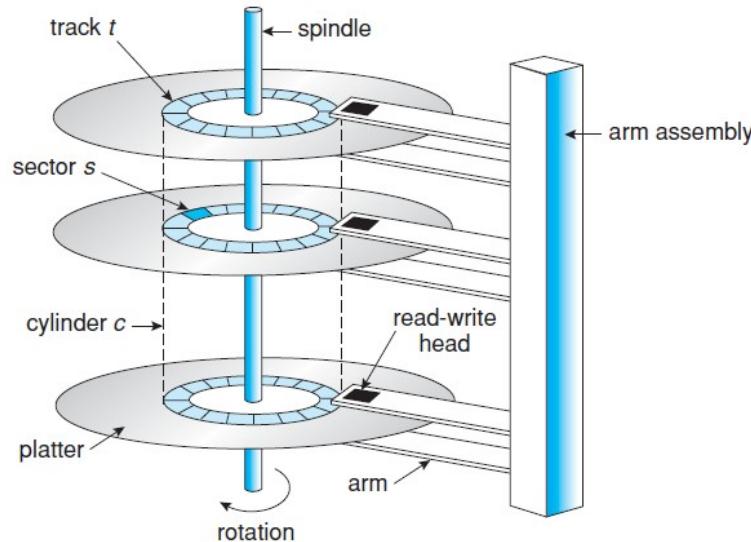
- Dalje proširenje ideje dvostrukog bafera
- Uvodi se N bafera
- Redom U/I uređaj puni/prazni bafera, dok OS čita/upisuje iz njih
- Nakon rada sa poslednjim baferom vraća se na prvi



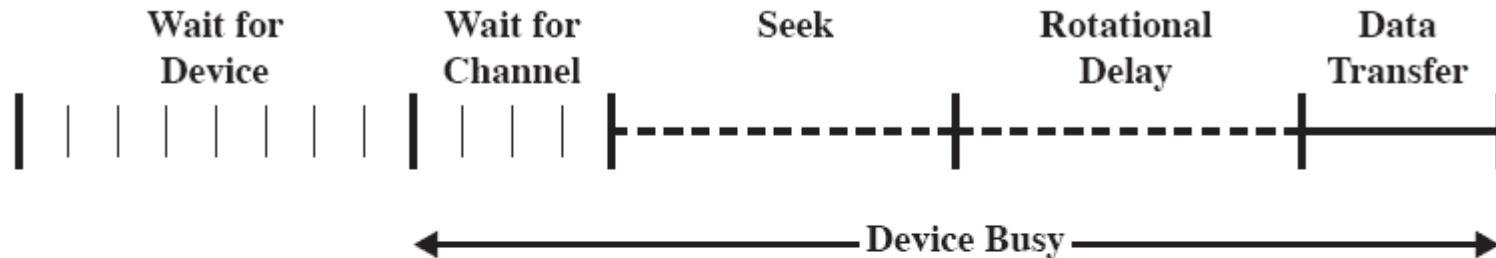
Raspoređivanje diska

- Performansa diska je izuzetno značajna za performansu celog sistema
- Disk je puno sporiji od radne memorije
- Često se pristupa disku zbog
 - Pristupa fajlovima
 - Stranica procesa u virtuelnoj memoriji

Parametri performanse diska



- Više operacija mora da se izvede za svaki zahtev prenosa podataka od/ka disku



Parametri performanse diska

- Čekanje na uređaj
 - Čekanje da uređaj postane raspoloživ procesu
- Čekanje na U/I kanal
 - Čekanje da kanal postane raspoloživ ako se kanal deli sa drugim U/I uređajima
- Vreme pozicioniranja
 - Vreme potrebno da se glava diska pomeri na potrebnu stazu

Parametri performanse diska

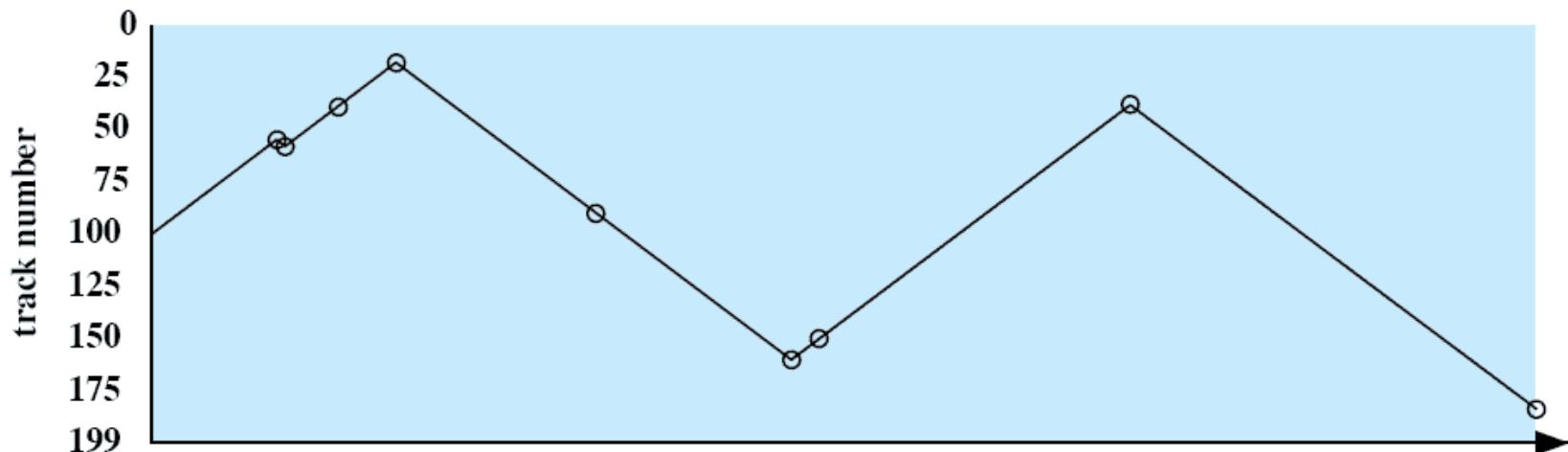
- Rotaciono kašnjenje
 - Vreme potrebno da se disk rotira tako da traženi sektor bude ispod glave
- Vreme prenosa
 - Vreme potrebno da se podaci pročitaju/upišu tako što se sektor kreće ispod glave rotiranjem diska

Politike raspoređivanja diska

- OS može da utiče na vreme pozicioniranja organizacijom zahteva
- Primer
 - Disk ima 200 staza
 - Glava je inicijalno pozicionirana na stazi 100
 - Zahteva se pristup stazama u redosledu
 - 55, 58, 39, 18, 90, 160, 150, 38, 184

Politike raspoređivanja diska

- Traženi redosled
 - 55, 58, 39, 18, 90, 160, 150, 38, 184
- Prvi unutra, prvi napolje
 - eng. *First in – first out* (FIFO)
 - Stavke iz reda čekanja se obrađuju redom
 - Fer prema svim procesima
 - Loše performanse jer se zahtevi ne organizuju



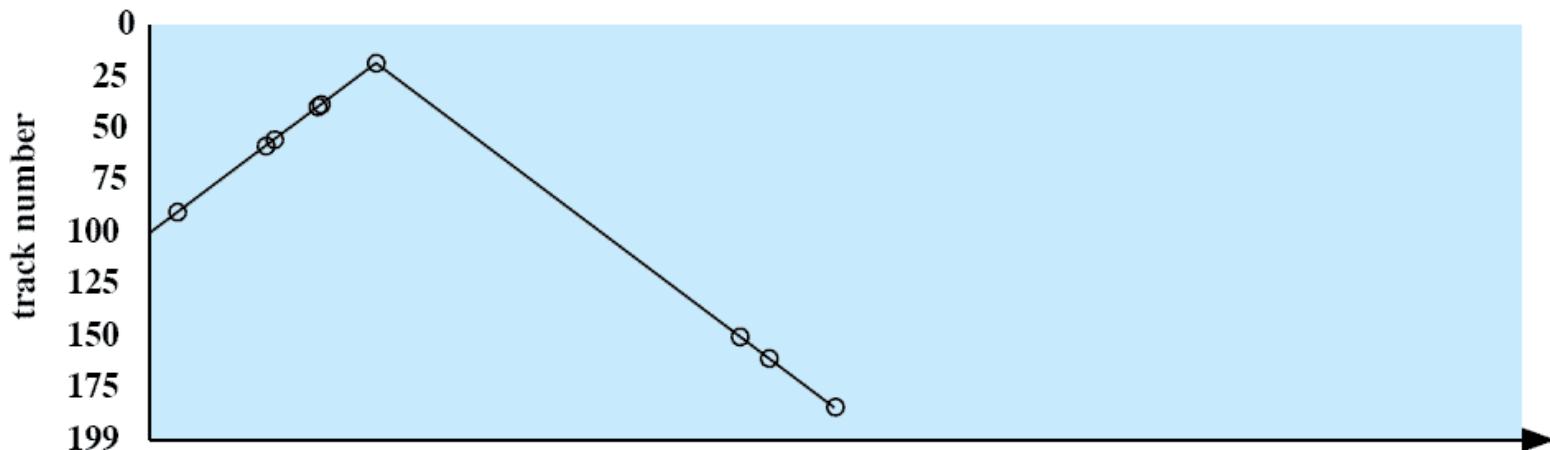
Politike raspoređivanja diska

□ Prioritet

- Određenoj vrsti poslova se daje veći prioritet
- Ti poslovi imaju prednost pri pristupu disku

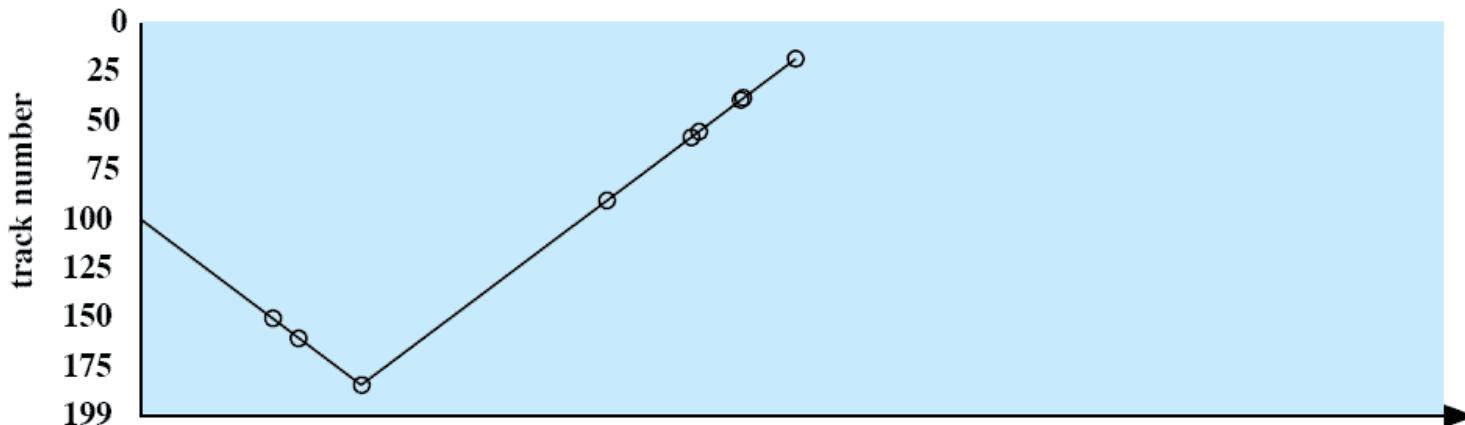
Politike raspoređivanja diska

- Traženi redosled
 - 55, 58, 39, 18, 90, 160, 150, 38, 184
- Prvi sa najkraćim vremenom usluge
 - eng. *Shortest service time first* (SSTF)
 - Bira se zahtev za koji je potrebno najmanje pomeranje ručice diska
 - Minimizira se vreme pozicioniranja



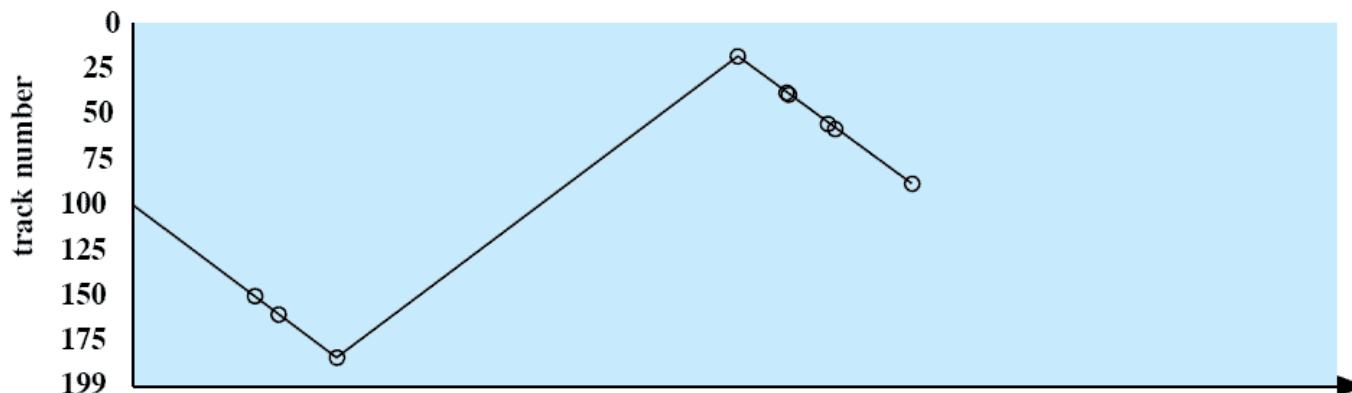
Politike raspoređivanja diska

- Traženi redosled
 - 55, 58, 39, 18, 90, 160, 150, 38, 184
- SCAN
 - Prioritet i SSTF mogu da dovedu do gladovanja zahteva
 - SCAN politika pomera ručicu samo u jednom smeru
 - Zadovoljava redom sve zahteve dok ne dođe do poslednje staze
 - Tada se menja smer i postupak ponavlja



Politike raspoređivanja diska

- Traženi redosled
 - 55, 58, 39, 18, 90, 160, 150, 38, 184
- C-Scan
 - Scan dovodi do dugog čekanja za staze na periferiji
 - C-SCAN ograničava skeniranje samo na jedan smer
 - Kada se dođe do kraja, ručica se vrati na početak i onda opet skenira staze



Formatiranje diska



Formatiranje diska

□ Fizičko formatiranje

- Deljenje diska u sektore
 - Zaglavljje
 - Podaci (obično 512 bajta)
 - Kod za ispravku greške (proračunava se po podacima)

□ Particioniranje

- Obavlja OS deljenjem diska na delove (particije)
- Svaka particija je logički disk

Formatiranje diska

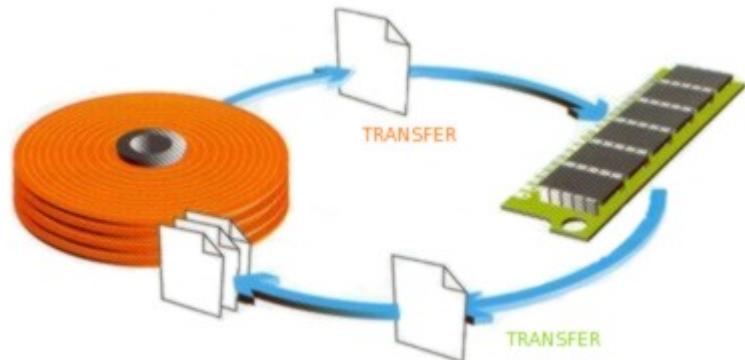
- Logičko formatiranje
 - OS inicijalizuje fajl sistem na disku
- Raw disk
 - OS može da napravi particiju bez fajl sistema
 - Na ovoj particiji podacima se pristupa kao nizu blokova
 - Omogućuje kontrolu niskog nivoa pri upravljanju podacima
 - Nema fajl sistema kao sloja između
 - Može da bude efikasnije u specifičnim situacijama

Boot particija

- Jedna particija mora da sadrži kod OS da bi se sistem mogao startovati
- Računar iz ROM memorije pokreće jednostavan program koji identificuje boot particiju
- Startuje kod za učitavanje OS sa predefinisane lokacije
 - Prvi sektor na disku (*Master boot record*)

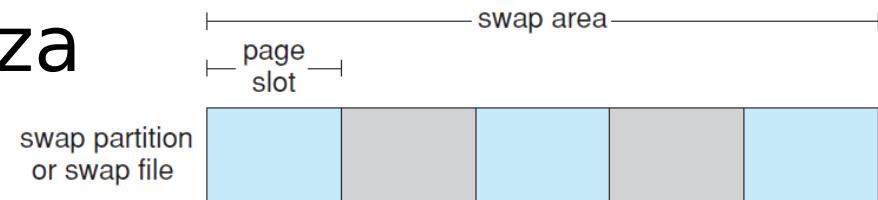
Prostor za razmenu

- Virtuelna memorija koristi disk kao dodatak radnoj memoriji
- Upravljanje ovim prostorom na disku veoma utiče na performanse
- Prostor za razmenu (*swap space*)
 - Prostor na disku namenjen za virtuelnu memoriju



Sadržaj prostora za razmenu

- Izdeljen na slotove za smeštanje stranica



- Šta se nalazi u prostoru?

- Više varijanti implementacije

1. Kompletna slika procesa (kod i podaci)
2. Samo podaci koji se menjaju u toku izvršavanja
 - Vrednosti iz *stack* i *heap* memorije
 - Programski kod i statički podaci su svakako na disku
 - Ubacuju se u radnu memoriju
 - Pri izbacivanju nema potrebe upisivati ih i u *swap*

Lokacija prostora za razmenu

1. U fajlu

- Koriste se ugrađene funkcije za pristup fajlu
- Jednostavno za upravljanje
- Manje efikasno

2. U posebnoj *raw* particiji

- Ne koriste se ugrađene funkcije za rad sa fajlovima
- Specijalizovane funkcije za upravljanjem prostorom za razmenu
- Efikasnije
- Nefleksibilno za promenu veličine prostora za razmenu

SSD diskovi

- *Solid-state disks*
- Dugoročna memorija koja se može koristiti kao sekundarno skladište umesto magnetnog diska
- Pouzdaniji od HDD jer nema pokretnih delova
- Brži jer nema kašnjenja zbog rotacije i pozicioniranja
- Manjeg kapaciteta i skuplji od HDD
- Isto vreme pristupa za sve delove pa se najčešće koristi prosto FCFS raspoređivanje



Korišćenje više diskova

- Performansa diska može se povećati raspoređivanjem operacija na više diskova paralelno
- Podaci se mogu oporaviti pri otkazu diska, ako se skladište dodatne informacije
 - Redundansa zbog poboljšanja pouzdanosti

RAID

- *Redundant Array of Independent Disks*
- Skup odvojenih fizičkih diskova koje OS vidi kao jedan logički uređaj
- Podaci su raspodeljeni po svim diskovima
- Redundantni disk(ovi) se koristi za skladištenje dodatnih informacija
 - Za obnovljivost podataka u slučaju otkaza diska

Vrste RAID

□ Hardverski RAID

- Hardverski kontroler raspoređuje podatke na više diskova i implementira RAID mehanizam

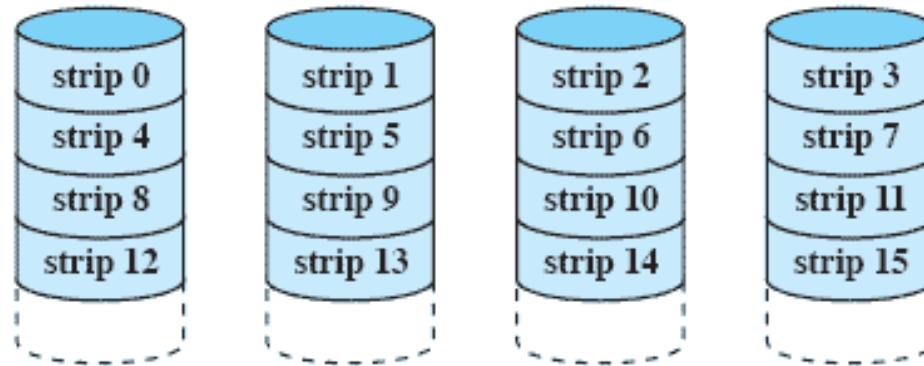
□ Softverski RAID

- OS vodi računa o raspoređivanju podataka na više diskova u skladu sa RAID šemom

RAID nivoi

- Različite varijante
 - Organizacije podataka na više diskova
 - Redundantnosti
- Svaka varijanta se zove RAID nivo
- Ima 7 nivoa
 - RAID 0 do RAID 6
 - U praksi se koriste 0,1, 5 i 6

RAID 0

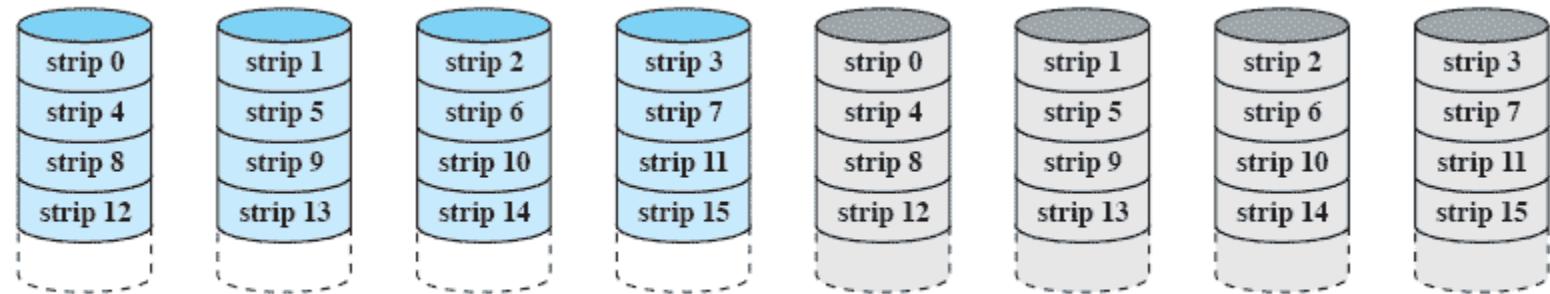


- Nije pravi RAID jer ne uključuje redundansu
- Raspoređuje podatke na više diskova
- Ako su podaci na različitim diskovima
 - Mogu se paralelno obrađivati
 - Unapređenje performanse

RAID 0

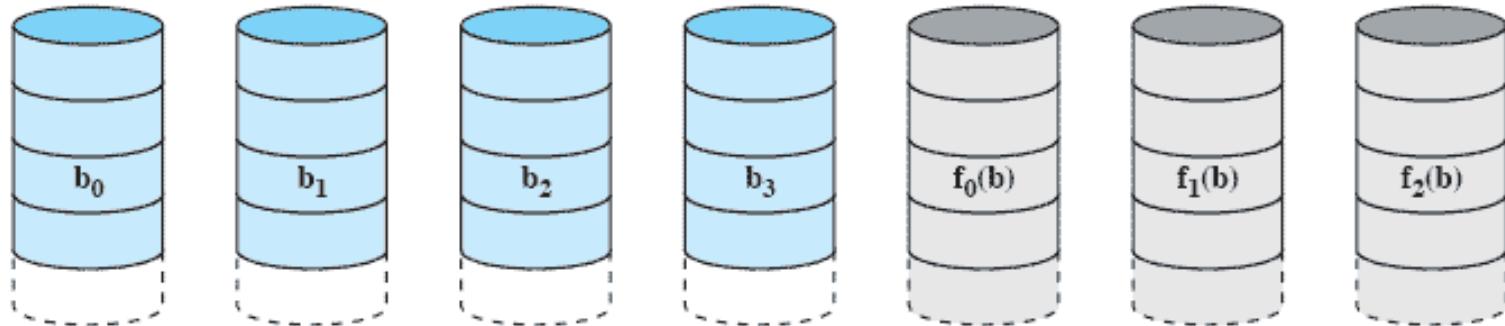
- Podaci se dele u trake (*strips*)
- Podaci se vide kao da su uskladišteni na jednom logičkom disku
 - Logički disk je podeljen u trake
- Trake se preslikavaju kružnim dodeljivanjem na susedne diskove u RAID nizu

RAID 1



- Redundansa se ostvaruje prostim dupliranjem podataka
- Ne gubi se na brzini jer se diskovima može pristupati paralelno
- Oporavak od otkaza diska je jednostavan
 - Pristupa se podacima na ispravnom disku
- Skupo
 - Zahteva duplo više prostora

RAID 2

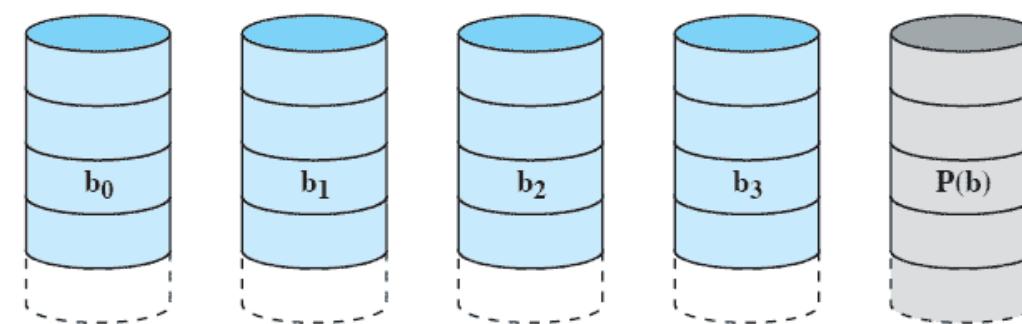


- Trake sadrže samo jedan bit
- Svi diskovi su sinhronizovani i isto pozicionirani
- Svi diskovi učestvuju u izvršavanju svakog zahteva

RAID 2

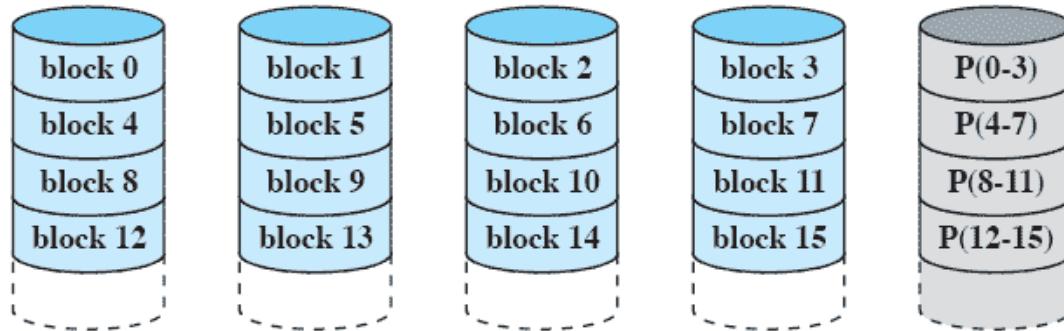
- Redundansa na bazi Hamingovog koda
 - Skladišti određeni broj dodatnih bitova
 - Omogućuju identifikaciju i oporavak od greške
- Nikada se nije koristio u praksi
 - Zahteva suviše dodatnih diskova

RAID 3



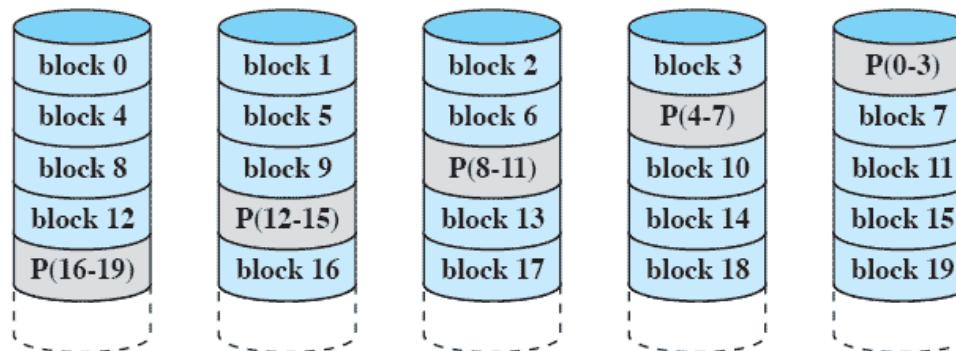
- Trake sadrže jedan bajt
- Svi diskovi su sinhronizovani i isto pozicionirani
- Samo jedan redundantni disk za bitove parnosti
- Moguć je oporavak od greške u slučaju otkaza jednog diska
- Ne koristi se u praksi
 - Nisu mogući istovremeni U/I zahtevi

RAID 4



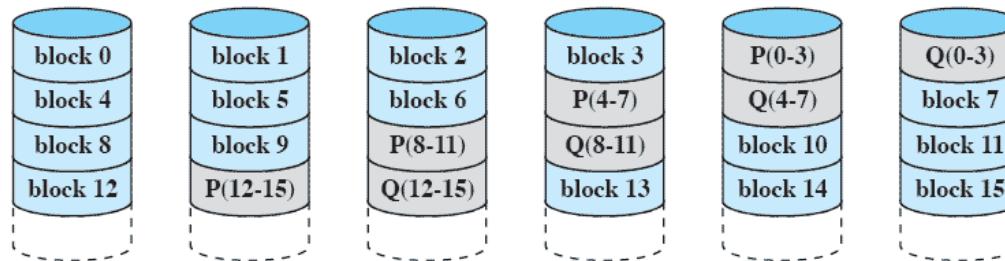
- Svaki disk se nezavisno pozicionira
 - Moguć paralelni rad
- Veće trake, sadrže blok
- Poseban disk za parnost
 - Potreban pristup ovom disku pri svakoj izmeni
 - Ovaj disk je usko grlo
- Retko se koristi u praksi

RAID 5



- Slično kao RAID 4, ali bitovi parnosti se smeštaju na svim diskovima
- Nema diska koji je usko grlo
- Sistem može da rekonstruiše podatke ako jedan disk otkaže

RAID 6



- Proširenje RAID 5 uvođenjem još jednog bloka za parnost
 - Blokovi za parnost distribuirani po svim diskovima
- Dva različita načina utvrđivanja bita parnosti
- Najpouzdaniji sistem
 - Sistem može da rekonstruiše podatke ako dva diska otkažu
- Spor
 - Dvostruko ažuriranje bitova parnosti

Keš diska

- Bafer u glavnoj memoriji za sektore diska
- Sadrži kopiju nekih od sektora diska
- Kada se pojavi I/O zahtev za pristup sektoru
 - najpre se proverava da li je sektor u kešu
- Postoje različite politike popunjavanja keša

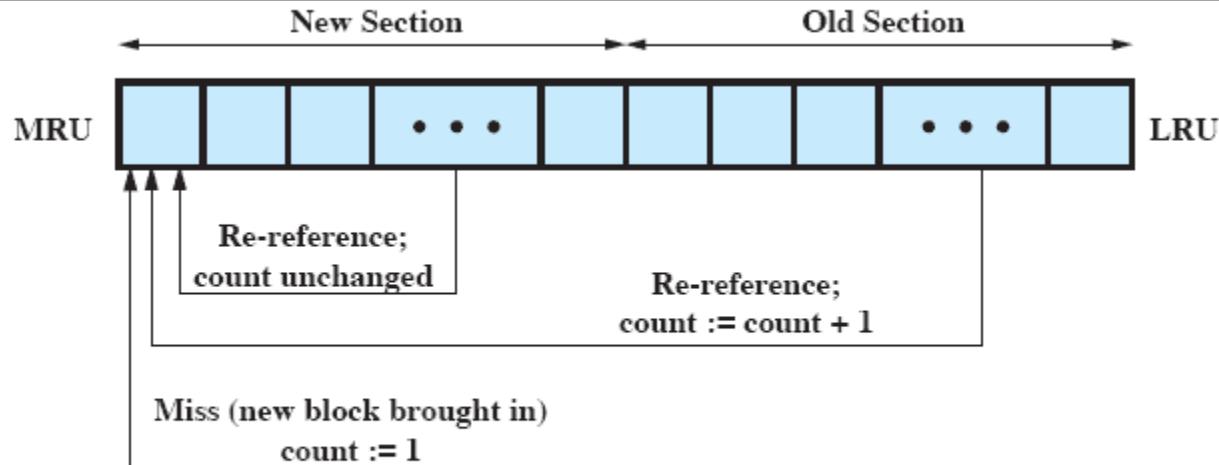
Zamena najmanje skoro korišćenog

- Novi blok se u keš ubacuje na mesto bloka od čijeg je referenciranja proteklo najviše vremena
- Za keš postoji stek pokazivača na blokove
 - Najskorije referencirani blok je na vrhu steka
 - Kada se blok referencira ili dodaje u keš, postavlja se na vrh steka
 - Iz keša se izbacuje blok sa dna steka

Zamena najređe korišćenog

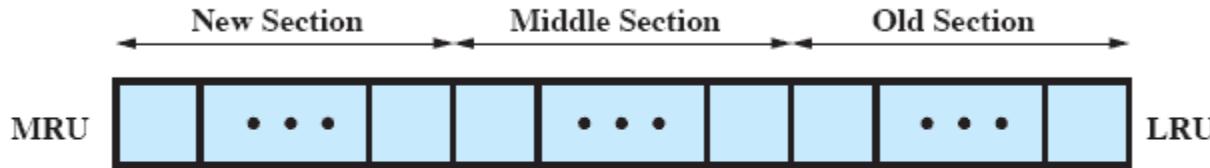
- Zamenjuje se blok koji je referenciran najmanji broj puta
- Brojač se dodeljuje svakom bloku
 - Inkrementira se svaki put kada se pristupi bloku
- Blok sa najmanjom vrednošću brojača se izbacuje iz keša kada je potrebno osloboditi mesto
- Vrednost brojača može da zavara, ako se blok referencira u retkim trenucima, ali se tada zbog lokalnosti često pristupa bloku
 - Tada je vrednost brojača velika, iako se blok ne koristi često i verovatno mu se neće uskoro pristupati

Zamena na bazi učestalosti korišćenja



- Inicijalna (lošija) varijanta sa dve sekcije
- Kada se referencira blok iz stare sekcije, povećava se brojač i prebacuje u novu sekciju
- Kada se referencira blok iz nove sekcije, brojač se ne uvećava
- Brojač ostaje isti ako se blok u kratkom periodu često referencira

Zamena na bazi učestalosti korišćenja



- Poboljšana varijanta sa tri sekcije
- Blokovima u novoj sekciji se brojač ne uvećava
- Kandidat za zamenu su blokovi u staroj sekciji
- Najbolje performanse od svih politika
 - Ne izbacuju se blokovi koji se relativno često koriste
 - Ne inkrementira se brojač pri čestim referenciranjima zbog lokalnosti