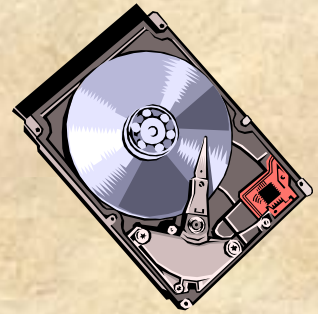
Operativni sistemi: interna struktura i principi dizajna

Poglavlje 11
Upravljanje U/I i raspoređivanje diska

Sedmo izdanje
William Stallings

Kategorije U/I uređaja

Spoljašnji uređaji koji obavljaju U/I operacije sa računarskim sistemom mogu se podeliti u tri kategorije:



Čitljivi za čoveka

- pogodni za komunikaciju sa korisnikom računara
- štampači, terminali, ekran, tastatura, miš

Čitljivi za mašinu

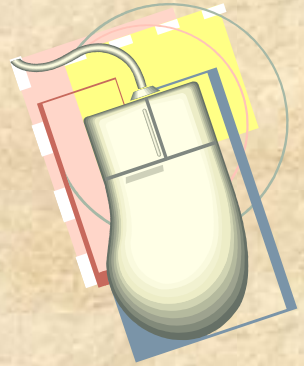
- pogodni za komunikaciju sa elektronskom opremom
- disk jedinice, USB fleš memorije, senzori, kontroleri

Komunikacioni

- pogodni za komunikaciju sa udaljenim uređajima
- modemi, digitalni linijski drajveri

Razlike među U/I uređajima

- Uređaji se razlikuju u više pogleda:



Brzina prenosa

- brzine prenosa podataka mogu se razlikovati za više redova veličine

Primena

- način korišćenja uređaja utiče na softver

Složenost upravljanja

- uticaj na operativni sistem zavisi od složenosti U/I modula koji upravlja uređajem

Jedinica prenosa

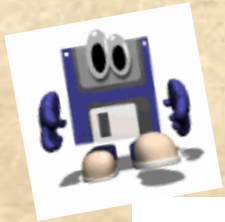
- podaci se mogu prenositi kao tok bajtova ili znakova, ili u većim blokovima

Prikaz podataka

- različiti uređaji koriste različite šeme kodiranja podataka

Greške

- priroda grešaka, način njihovog prijavljivanja, njihove posledice i raspoloživi načini reagovanja razlikuju se od uređaja do uređaja



Brzine prenosa

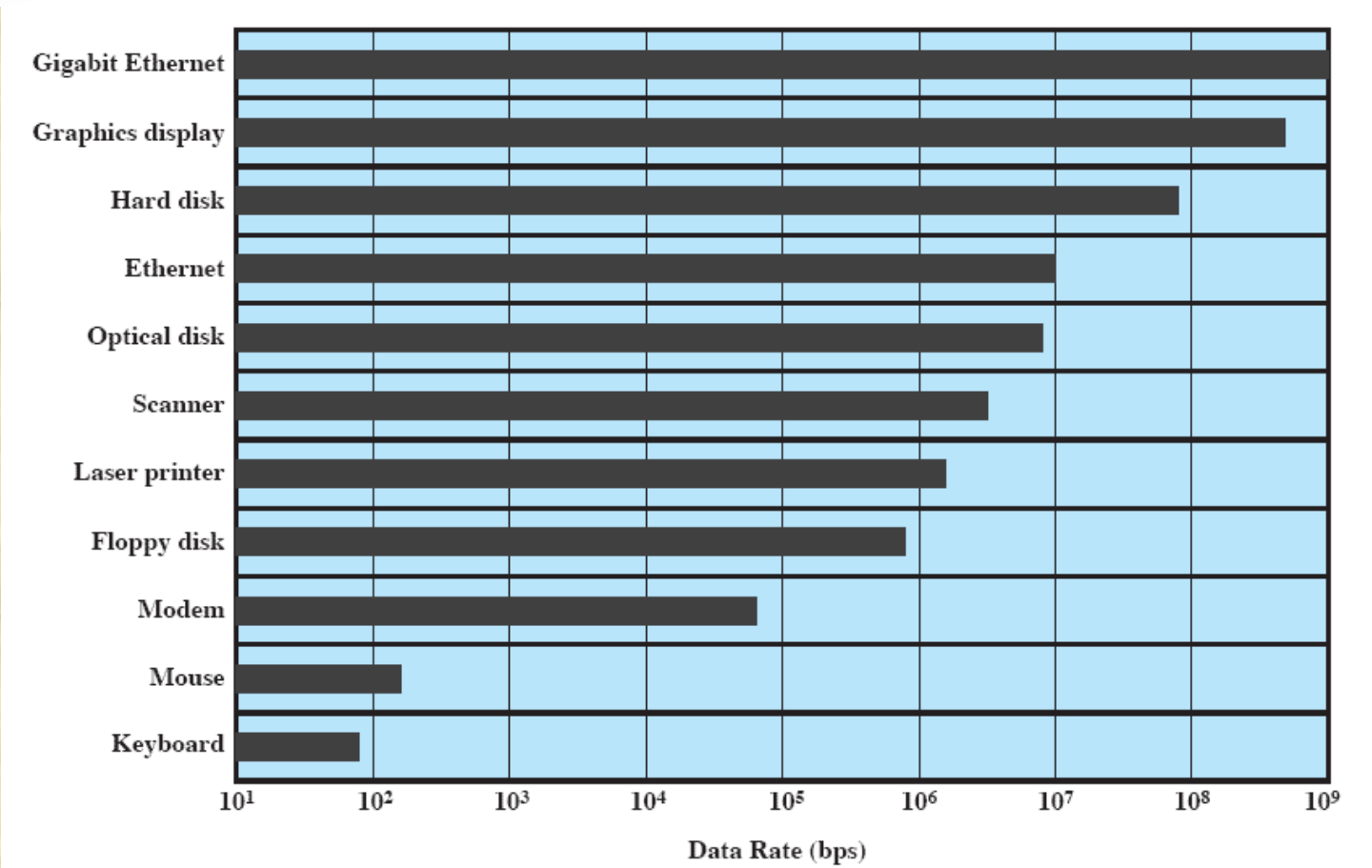


Figure 11.1 Typical I/O Device Data Rates

Organizacija U/I funkcije

- Postoje tri tehnike obavljanja U/I:
- **Programirani U/I**
  - procesor u ime procesa izdaje U/I komandu U/I modulu; proces zatim aktivno čeka da se operacija završi pre nego što nastavi
- **U/I upravljan prekidima**
  - procesor izdaje U/I komandu u ime procesa
    - ako je neblokirajuća – procesor nastavlja da izvršava instrukcije procesa koji je izdao U/I komandu
    - ako je blokirajuća – sledeća instrukcija koju procesor izvršava pripada OS-u, koji tekući proces prevodi u blokirano stanje i raspoređuje drugi proces
- **Direktan pristup memoriji (DMA)**
  - DMA modul upravlja razmenom podataka između glavne memorije i U/I modula

Tehnike obavljanja U/I

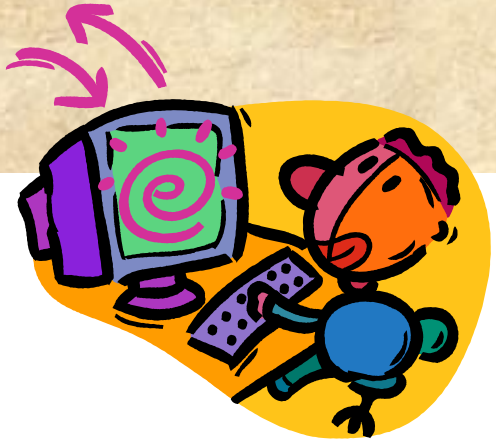


Table 11.1 I/O Techniques

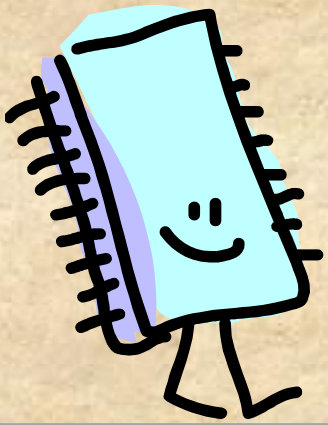
	No Interrupts	Use of Interrupts
I/O-to-memory transfer through processor	Programmed I/O	Interrupt-driven I/O
Direct I/O-to-memory transfer		Direct memory access (DMA)

Razvoj U/I funkcije

1. Procesor direktno upravlja perifernim uređajem
2. Dodaje se kontroler ili U/I modul
3. Ista konfiguracija kao u koraku 2, ali se sada koriste prekidi
4. U/I modul dobija direktnu kontrolu nad memorijom putem DMA
5. U/I modul postaje zaseban procesor, sa specijalizovanim skupom instrukcija prilagođenim za U/I
6. U/I modul ima sopstvenu lokalnu memoriju i zapravo je samostalan računar



Direktan pristup memoriji



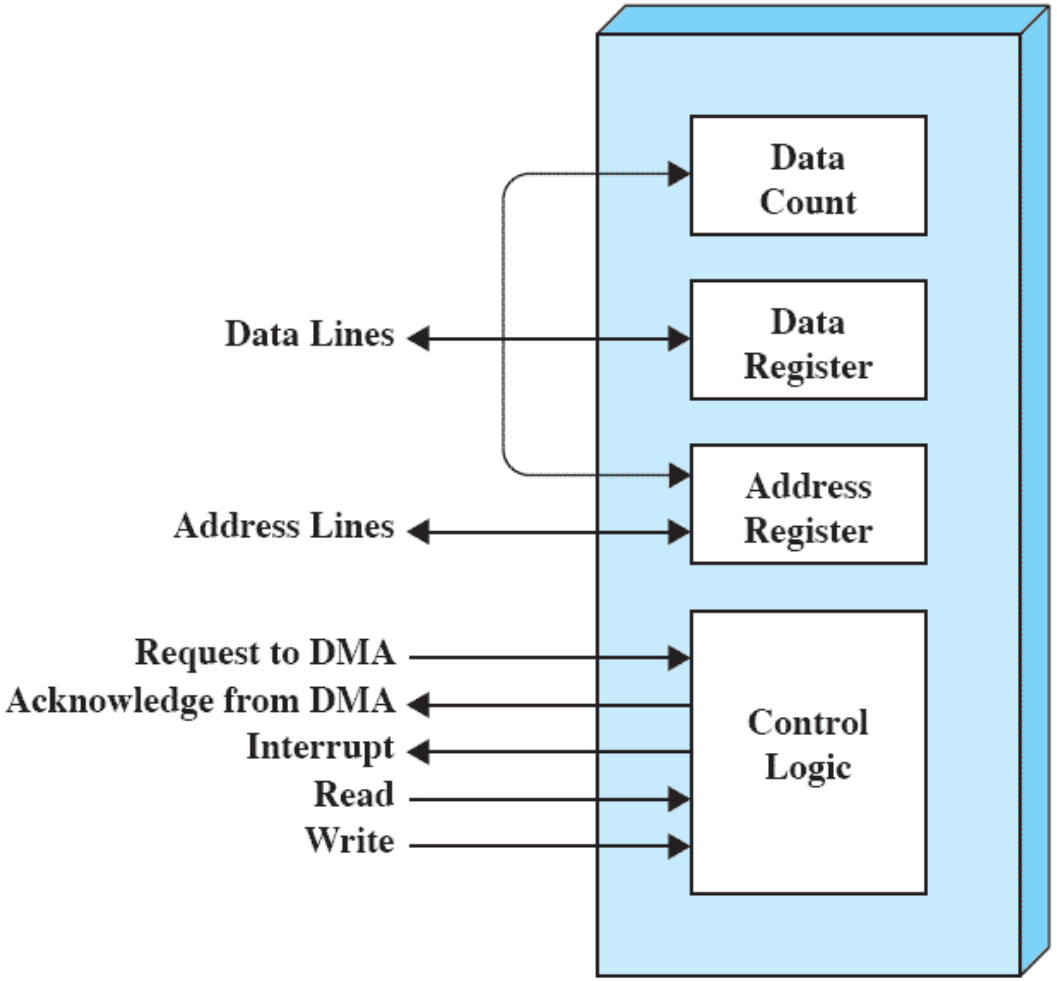


Figure 11.2 Typical DMA Block Diagram

Parametri performansi diska

- Konkretni detalji U/I operacija sa diskom zavise od:
  - računarskog sistema
  - operativnog sistema
  - prirode U/I kanala i hardvera kontrolera diska

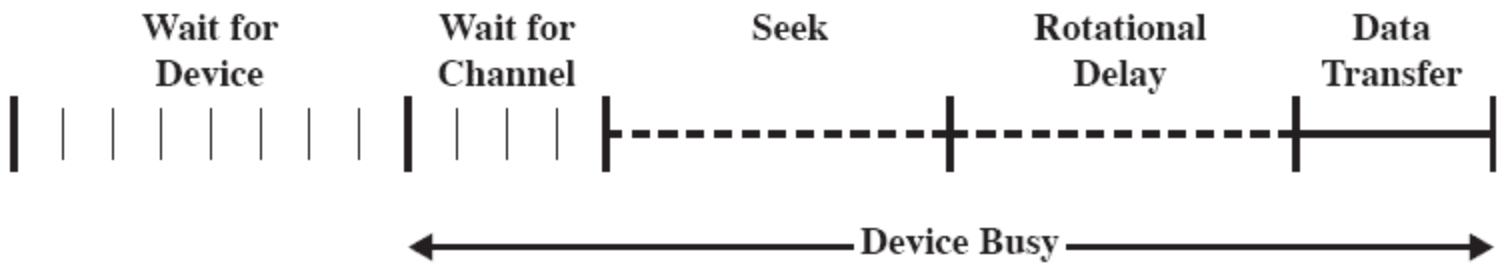


Figure 11.6 Timing of a Disk I/O Transfer

Pozicioniranje glava za čitanje/upis

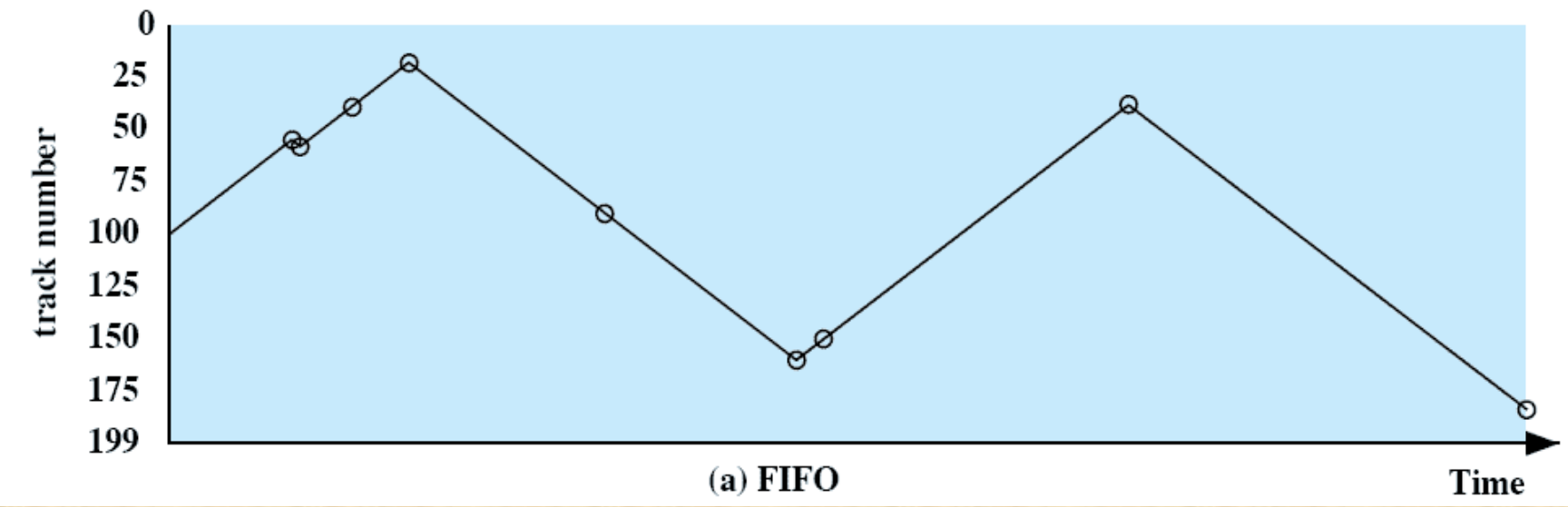
- Dok disk jedinica radi, disk se okreće konstantnom brzinom
- Za čitanje ili upis glava mora biti pozicionirana na željenu stazu i na početak željenog sektora na toj stazi
- Izbor staze podrazumeva pomeranje glave kod sistema sa pokretnom glavom ili elektronski izbor jedne glave kod sistema sa fiksnim glavama
- Kod sistema sa pokretnom glavom, vreme potrebno da se glava pozicionira na stazu naziva se **vreme pozicioniranja** (seek time)
- Vreme potrebno da početak sektora stigne do glave naziva se **rotaciono kašnjenje** (rotational delay)
- Zbir vremena pozicioniranja i rotacionog kašnjenja čini **vreme pristupa** (access time)

Naziv	Opis	Napomene
Izbor prema podnosiocu zahteva		
RSS	Slučajno raspoređivanje	Za analizu i simulaciju
FIFO	Prvi ušao, prvi izašao	Najpravičniji od svih
PRI	Prioritet prema procesu	Kontrola izvan upravljanja redom za disk
LIFO	Poslednji ušao, prvi izašao	Maksimalna lokalnost i iskorišćenost resursa
Izbor prema traženoj stavci		
SSTF	Najkraće vreme usluživanja prvo	Visoka iskorišćenost, kratki redovi
SCAN	Napred-nazad po disku	Bolja raspodela usluge
C-SCAN	Jednosmerno, brz povratak	Manja varijabilnost usluge
N-step-SCAN	SCAN po N zapisa odjednom	Garantovana usluga
FSCAN	N-step-SCAN sa N = veličina reda na početku SCAN ciklusa	Osetljiv na opterećenje

Tabela 11.3 Algoritmi raspoređivanja diska

Prvi ušao, prvi izašao (FIFO)

- Obrađuje zahteve redom kojim pristižu
- Pravičan prema svim procesima
- Ako se mnogo procesa nadmeće za disk, performanse su približne slučajnom raspoređivanju



Prioritet (PRI)

- Raspoređivanje nije pod kontrolom softvera za upravljanje diskom
- Cilj nije optimizacija iskorišćenosti diska, već ispunjavanje drugih ciljeva
- Kratki paketni poslovi i interaktivni poslovi dobijaju viši prioritet
- Obezbeđuje dobro vreme odziva u interaktivnom radu
- Duži poslovi ponekad moraju da čekaju predugo
- Loša politika za sisteme baza podataka

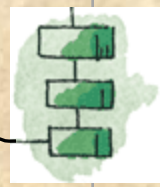




Kriterijumi raspoređivanja

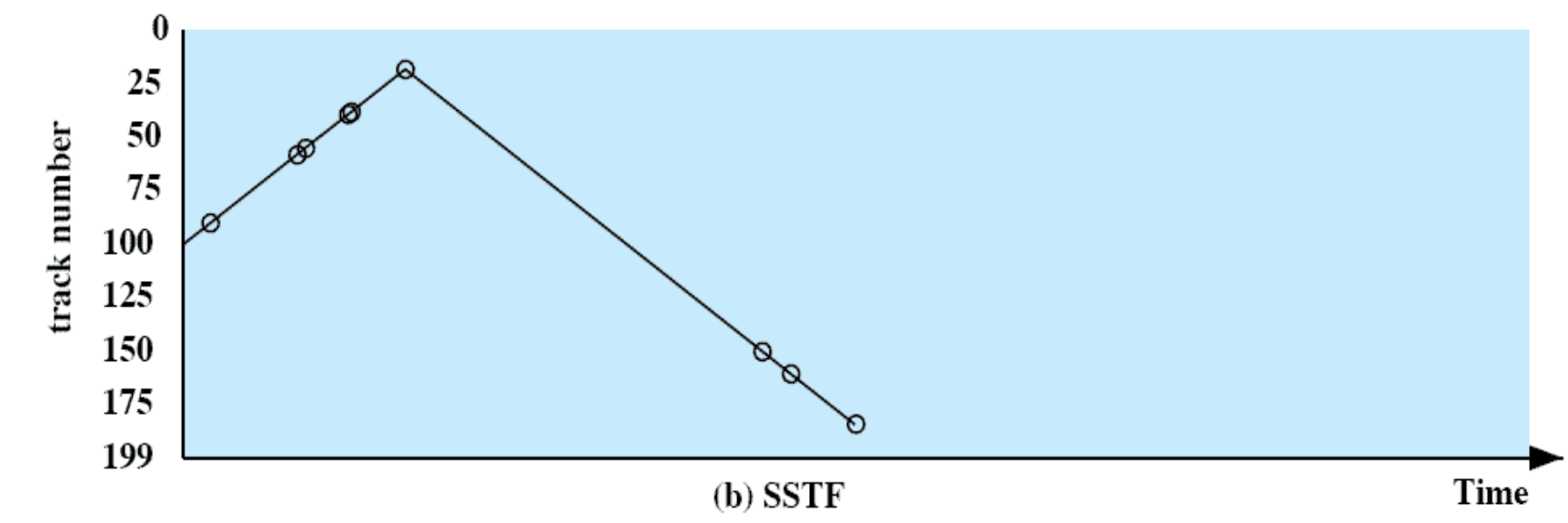
- Strategije raspoređivanja diska obično se projektuju tako da sve zahteve tretiraju kao da imaju jednak prioritet. Cilj strategije je optimizacija jedne ili više sledećih veličina:
- Propusnost: broj obrađenih zahteva u jedinici vremena
- Prosečno vreme odziva: čekanje na obradu zahteva
- Varijansa vremena odziva: bez izgladnjivanja (beskonačnog odlaganja). Svaki zahtev treba da bude obrađen u razumnom roku.
  - Meri pravičnost i predvidljivost.





Najkraće vreme usluživanja prvo (SSTF)

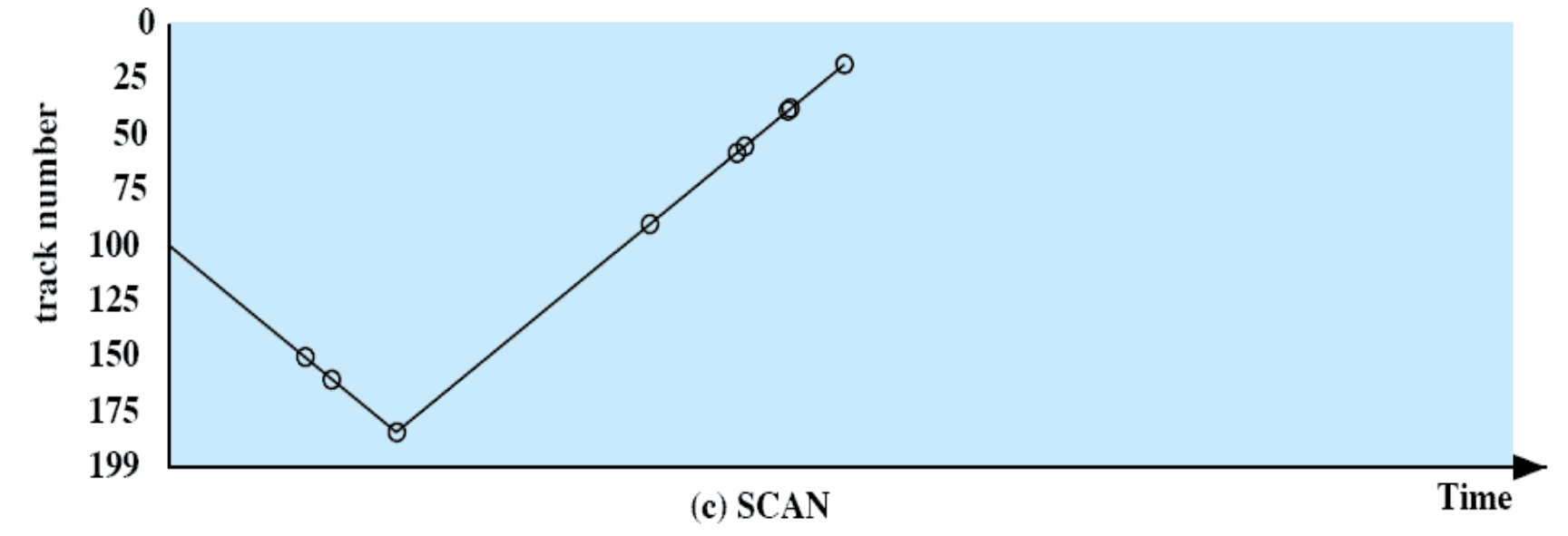
- Bira se zahtev za koji se ručica diska najmanje pomera od trenutnog položaja
- Uvek se bira najmanje vreme pozicioniranja
- Moguće izgladnjivanje: pretpostavimo da stalno stižu zahtevi za staze između 100 i 1



(b) SSTF

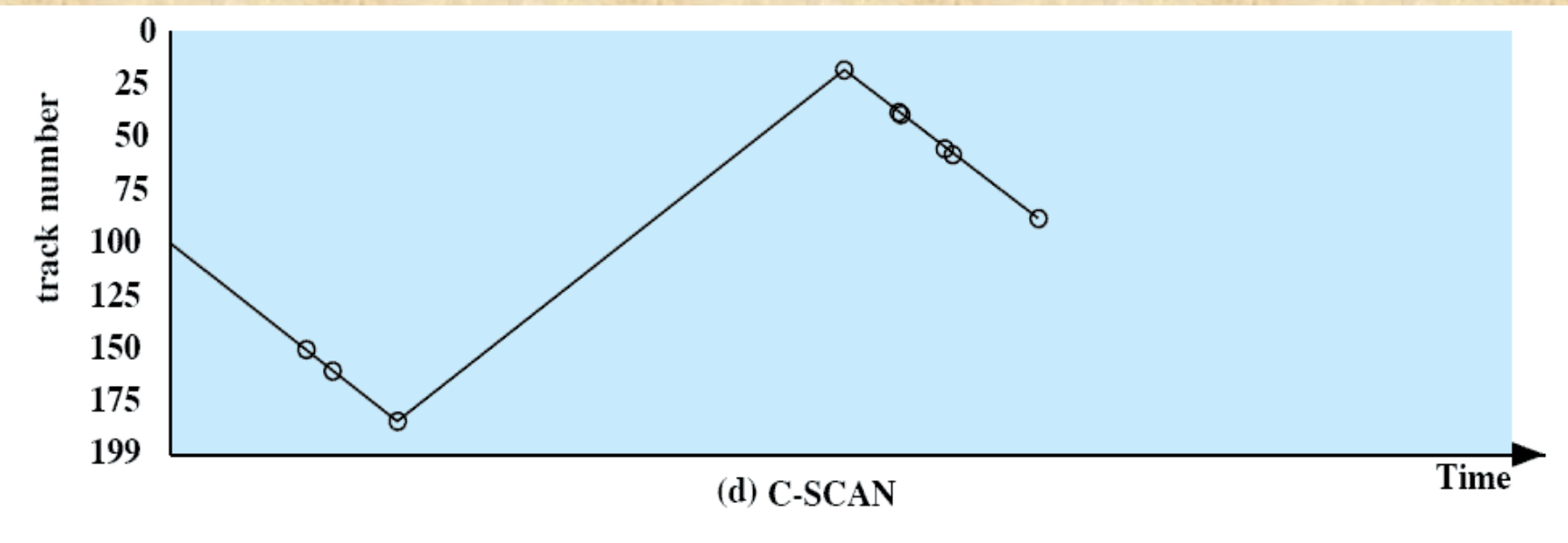
SCAN

- Poznat i kao algoritam lifta
- Ručica se kreće samo u jednom smeru
  - uslužuje sve zahteve na čekanju dok ne stigne do poslednje staze u tom smeru, a zatim menja smer
- Favorizuje poslove sa zahtevima za staze najbliže krajnjoj unutrašnjoj i krajnjoj spoljašnjoj stazi, ali ne izaziva izgladnjivanje.



C-SCAN
(kružni SCAN)

- Skeniranje se ograničava samo na jedan smer
- Kada se obiđe poslednja staza u jednom smeru, ručica se vraća na suprotni kraj diska i skeniranje počinje ponovo
- Smanjuje maksimalno kašnjenje za nove zahteve.



(a) FIFO (starting at track 100)		(b) SSTF (starting at track 100)		(c) SCAN (starting at track 100, in the direction of increasing track number)		(d) C-SCAN (starting at track 100, in the direction of increasing track number)	
Next track accessed	**Number of tracks traversed**	**Next track accessed**	**Number of tracks traversed**	**Next track accessed**	**Number of tracks traversed**	**Next track accessed**	**Number of tracks traversed**
55	45	90	10	150	50	150	50
58	3	58	32	160	10	160	10
39	19	55	3	184	24	184	24
18	21	39	16	90	94	18	166
90	72	38	1	58	32	38	20
160	70	18	20	55	3	39	1
150	10	150	132	39	16	55	16
38	112	160	10	38	1	58	3
184	146	184	24	18	20	90	32
Average seek length	55.3	**Average seek length**	27.5	**Average seek length**	27.8	**Average seek length**	35.8

Tabela 11.2 Poređenje algoritama raspoređivanja diska

RAID

- Redundantni niz nezavisnih diskova
- Cilj: poboljšati performanse i pouzdanost U/I sa diskom
- Ima sedam nivoa, od 0 do 6
  - Nivoi 0 i 1 ne uključuju provere parnosti
  - Detalji podele na trake i parnosti zavise od nivoa

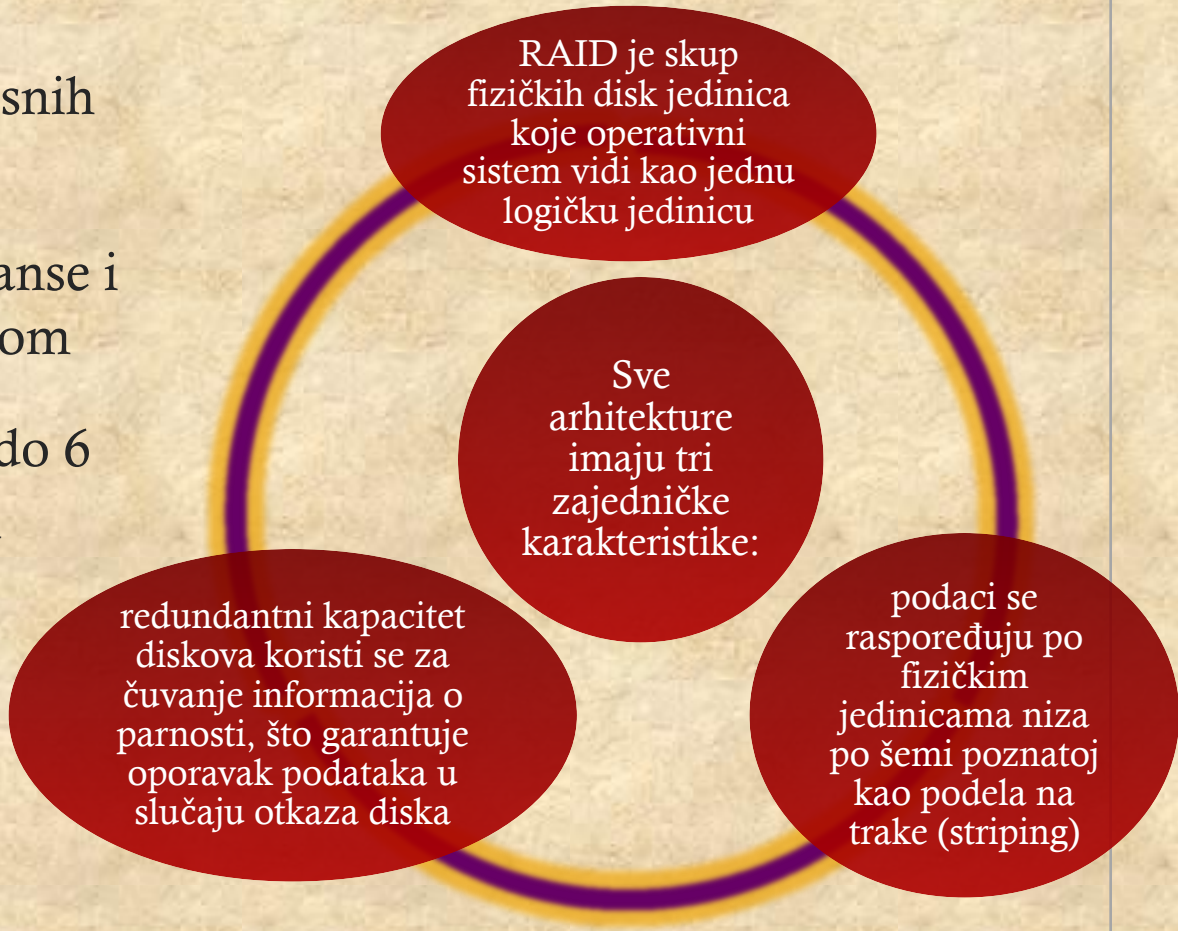


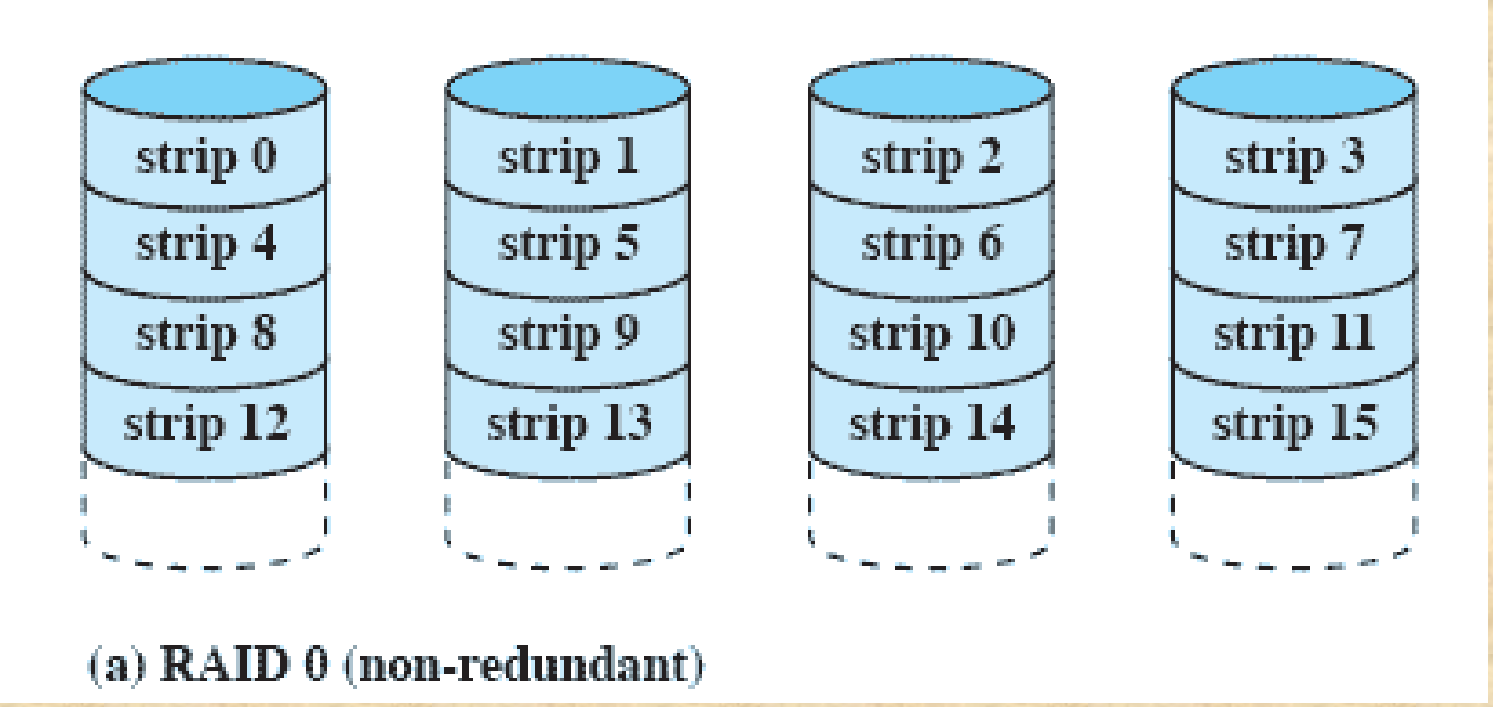
Tabela 11.4 RAID nivoi

Kategorija	Nivo	Opis	Potreban broj diskova	Dostupnost podataka	Brzina prenosa podataka za velike U/I zahteve	Brzina obrade malih U/I zahteva
Podela na trake	0	Bez redundantnosti	N	Manja nego kod jednog diska	Veoma visoka	Veoma visoka i za čitanje i za upis
Preslikavanje	1	Sa preslikavanjem	$2N$	Veća nego kod RAID 2, 3, 4 ili 5; manja nego kod RAID 6	Za čitanje veća nego kod jednog diska; za upis slična kao kod jednog diska	Za čitanje do dva puta veća nego kod jednog diska; za upis slična kao kod jednog diska
Paralelan pristup	2	Redundantnost pomoću Hammingovog koda	$N + m$	Mnogo veća nego kod jednog diska; uporediva sa RAID 3, 4 ili 5	Najveća od svih navedenih alternativa	Približno dvostruko veća nego kod jednog diska
	3	Parnost sa preplitanjem bitova	$N + 1$	Mnogo veća nego kod jednog diska; uporediva sa RAID 2, 4 ili 5	Najveća od svih navedenih alternativa	Približno dvostruko veća nego kod jednog diska
Nezavisan pristup	4	Parnost sa preplitanjem blokova	$N + 1$	Mnogo veća nego kod jednog diska; uporediva sa RAID 2, 3 ili 5	Za čitanje slična kao kod RAID 0; za upis znatno manja nego kod jednog diska	Za čitanje slična kao kod RAID 0; za upis znatno manja nego kod jednog diska
	5	Distribuirana parnost sa preplitanjem blokova	$N + 1$	Mnogo veća nego kod jednog diska; uporediva sa RAID 2, 3 ili 4	Za čitanje slična kao kod RAID 0; za upis manja nego kod jednog diska	Za čitanje slična kao kod RAID 0; za upis uglavnom manja nego kod jednog diska
	6	Dvostruka distribuirana parnost sa preplitanjem blokova	$N + 2$	Najveća od svih navedenih alternativa	Za čitanje slična kao kod RAID 0; za upis manja nego kod RAID 5	Za čitanje slična kao kod RAID 0; za upis znatno manja nego kod RAID 5

N = broj diskova sa podacima; m proporcionalno log N

RAID nivo 0

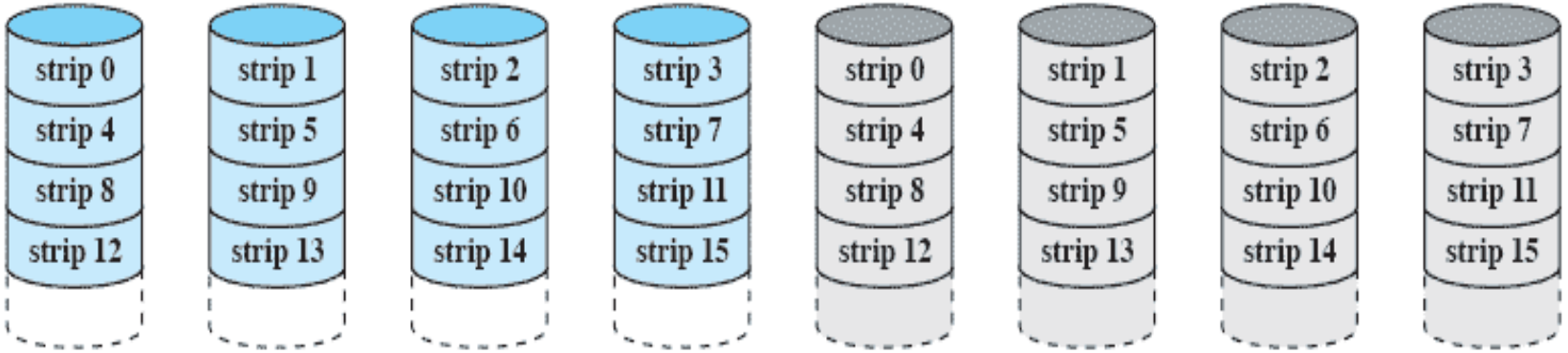
- Nije pravi RAID: nema redundantnosti
- Podaci se raspoređuju po svim diskovima
- Dva nepovezana U/I zahteva mogu se obaviti paralelno ako su na različitim diskovima, ili
- Ovde trake 0-3, 4-7 itd. čine po jednu *prugu* (stripe): uzastopne bajtove datoteke, koji se čitaju paralelno, jednom operacijom.



(a) RAID 0 (non-redundant)

RAID nivo 1

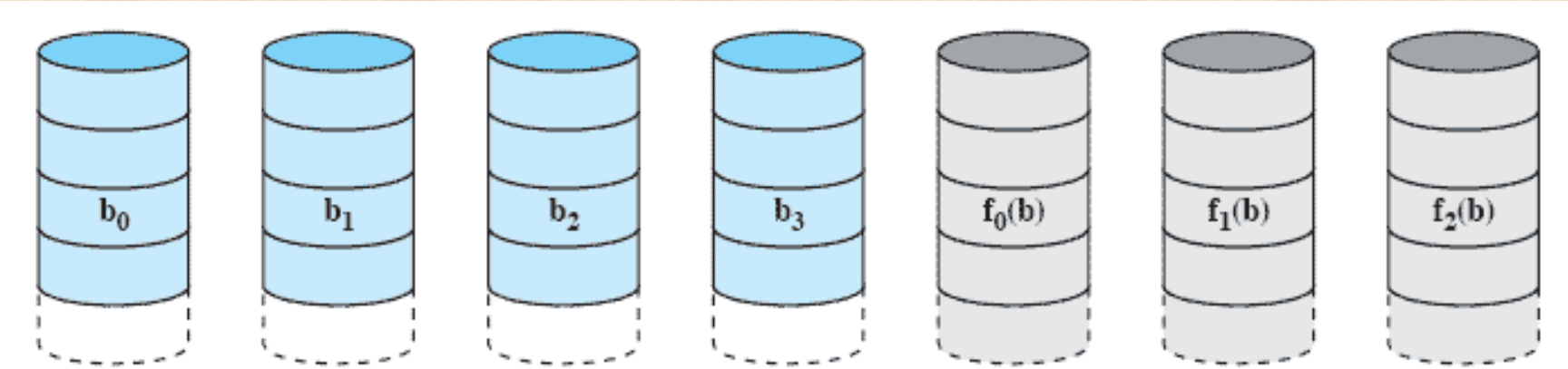
- Redundantnost se postiže jednostavnim dupliranjem svih podataka
- Nema „kazne pri upisu“ (vremena potrebnog za izračunavanje i ažuriranje bitova parnosti).
- Kada jedan disk otkaže, podacima se i dalje može pristupiti sa drugog diska
- Glavni nedostatak je cena



(b) RAID 1 (mirrored)

RAID nivo 2

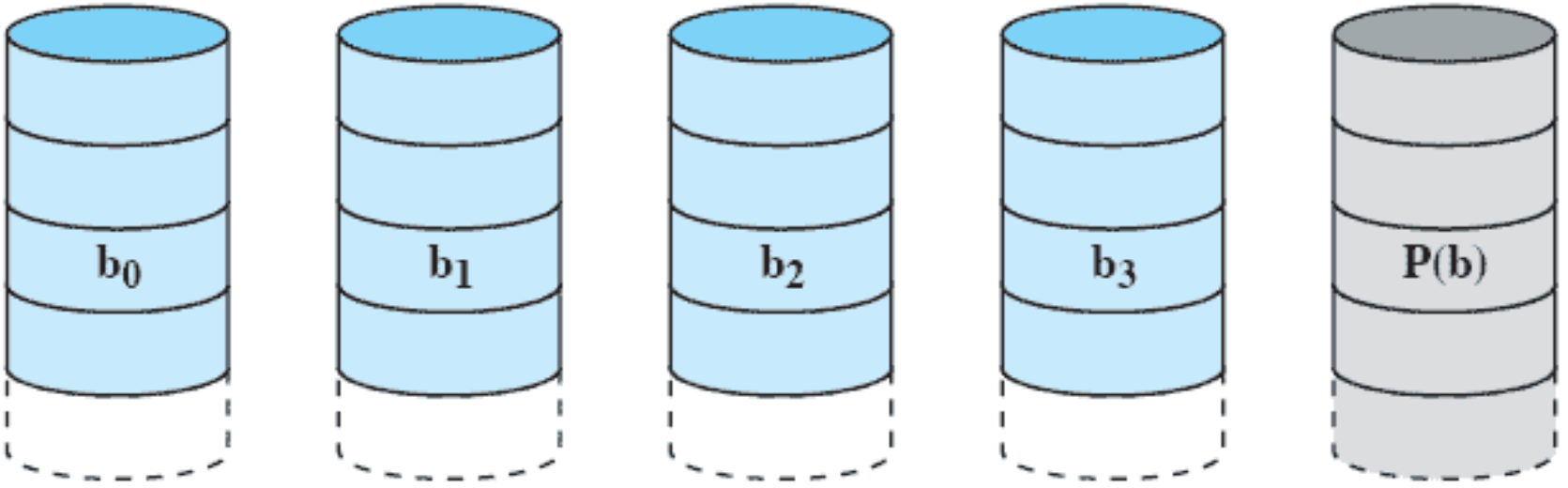
- Koristi tehniku paralelnog pristupa: svi diskovi učestvuju u svakoj operaciji
- Koristi se podela podataka na trake, obično male
- Obično se koristi Hammingov kod; može da otkrije/ispravi greške u jednom bitu i da otkrije greške u dva bita.
- Efikasan izbor u okruženju u kome se javlja mnogo grešaka diska, a inače preterano rešenje: potrebno je previše dodatnih diskova.



(c) RAID 2 (redundancy through Hamming code)

RAID nivo 3

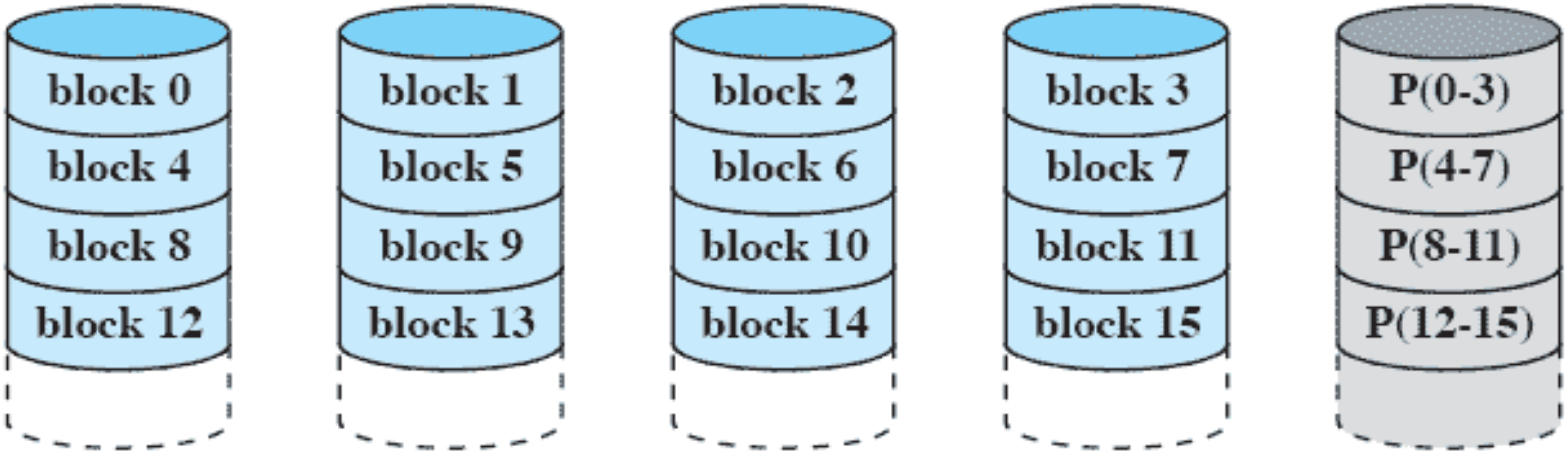
- Zahteva samo jedan redundantni disk, koji služi kao disk parnosti. Ako jedan disk otkaže, njegov sadržaj može se rekonstruisati pomoću diska parnosti.
- Koristi paralelan pristup, a podaci su raspoređeni u male trake
- Zbog male veličine traka može postići veoma velike brzine prenosa podataka.



(d) RAID 3 (bit-interleaved parity)

RAID nivo 4

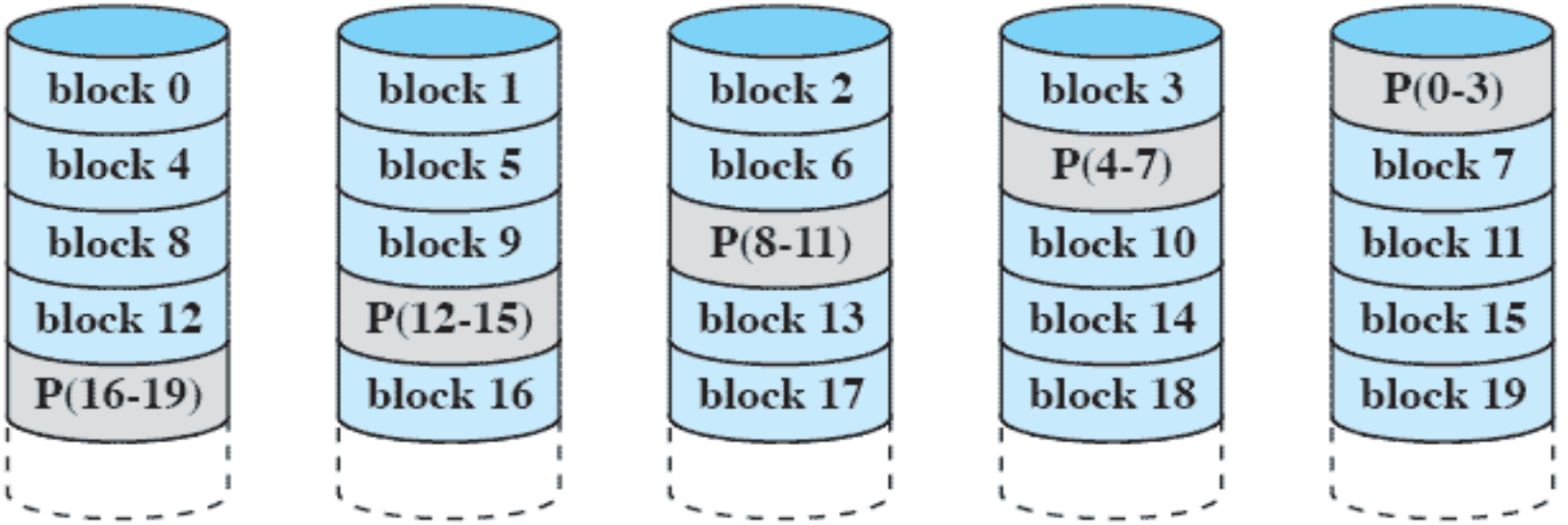
- Koristi tehniku nezavisnog pristupa
- Traka parnosti računa se bit po bit nad odgovarajućim trakama svakog diska sa podacima, a bitovi parnosti čuvaju se u odgovarajućoj traci diska parnosti
- Postoji kazna pri upisu kada se izvršava mali U/I zahtev za upis



(e) RAID 4 (block-level parity)

RAID nivo 5

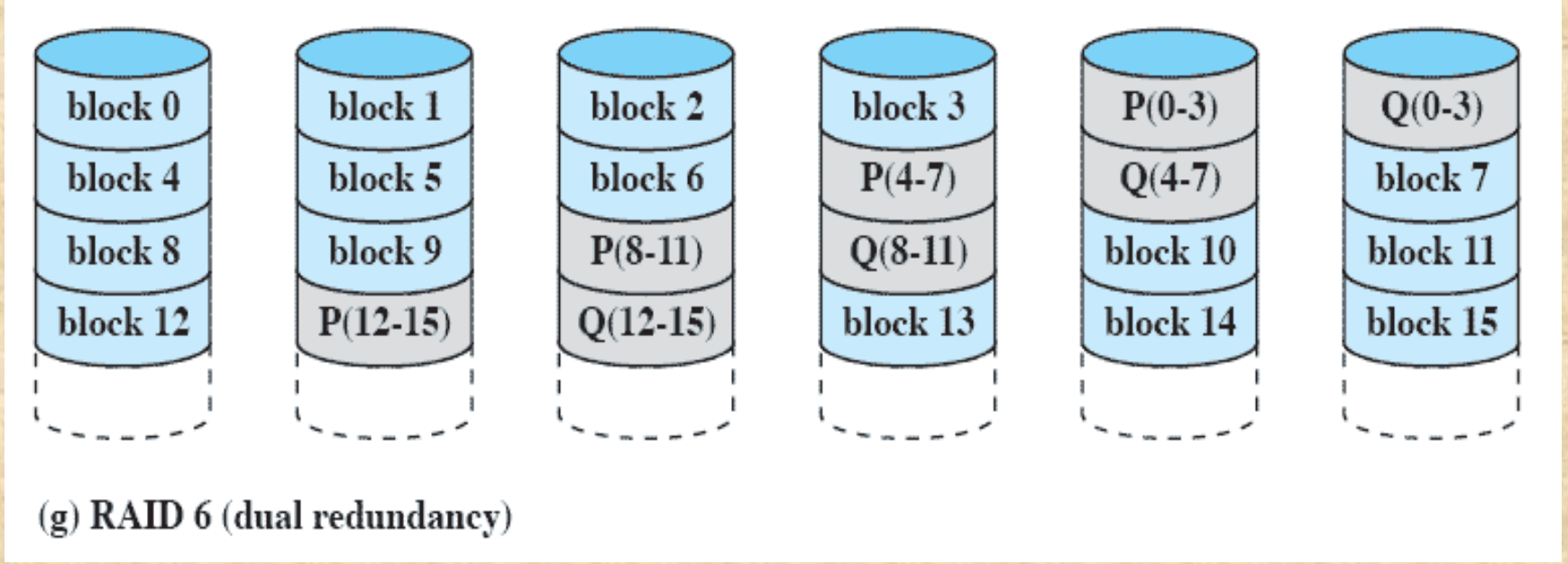
- Sličan nivou RAID-4, ali bitove parnosti raspoređuje po svim diskovima
- Raspodela se obično vrši po round-robin šemi
- Gubitak bilo kog pojedinačnog diska ne dovodi do gubitka podataka



(f) RAID 5 (block-level distributed parity)

RAID nivo 6

- Izvode se dva različita izračunavanja parnosti, a rezultati se čuvaju u zasebnim blokovima na različitim diskovima
- Obezbeđuje izuzetno visoku dostupnost podataka
- Izaziva znatnu kaznu pri upisu jer svaki upis utiče na dva bloka parnosti



(g) RAID 6 (dual redundancy)

Disk keš

- *Keš memorija* je naziv za memoriju koja je manja i brža od glavne memorije i nalazi se između glavne memorije i procesora
- Smanjuje prosečno vreme pristupa memoriji koristeći princip lokalnosti
- *Disk keš* je bafer u glavnoj memoriji za sektore diska
- Sadrži kopiju nekih sektora sa diska



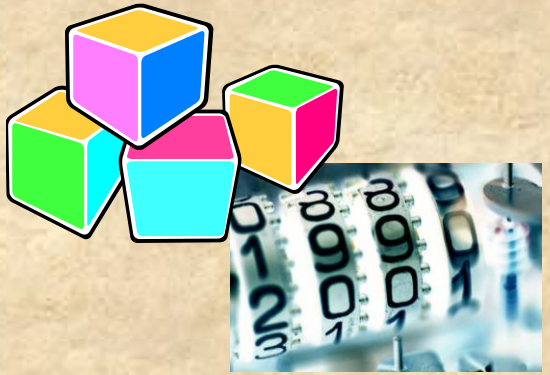
Najdavnije korišćen (LRU)

- Najčešće korišćen algoritam za rešavanje pitanja strategije zamene
- Zamenjuje se blok koji je najduže u kešu bez ijednog pristupa
- Stek pokazivača ukazuje na blokove u kešu
  - blok kome je najskorije pristupano nalazi se na vrhu steka
  - kada se bloku pristupi ili kada se on unese u keš, stavlja se na vrh steka



Najređe korišćen (LFU)

- Zamenjuje se blok kome je najmanje puta pristupano
- Svakom bloku pridružen je brojač
- Brojač se uvećava pri svakom pristupu bloku
- Kada je potrebna zamena, bira se blok sa najmanjom vrednošću brojača



Rezime

- U/I arhitektura je interfejs računarskog sistema prema spoljnom svetu
- U/I sa diskom ima najveći uticaj na ukupne performanse sistema
- Dva najšire korišćena pristupa su raspoređivanje diska i disk keš
- Disk keš je bafer, obično u glavnoj memoriji, koji služi kao keš blokova diska između disk memorije i ostatka glavne memorije