

Kodovi za otkrivanje i popravku grešaka

Greške u radu sa memorijom

Istraživanje iz 2010, sistem Jaguar, 360TB ECC RAM (engl. Error-Correcting Code)

- greške u radu sa memorijom: 350 u minuti
- skalirano na računar sa 8 GB memorije:
otprilike 1 greška na svaka 2 sata

Google istraživanje iz 2009: oko 5 single-bit grešaka na 8 GB memorije na sat

Manji tranzistori – manje energije za pobudu

Pozadinsko zračenje (mahom od kosmičkih zraka)

Izvori: How To Kill A Supercomputer: Dirty Power, Cosmic Rays, and Bad Solder, *IEEE Spectrum*, 2016; Flipping Bits in Memory Without Accessing Them: An Experimental Study of DRAM Disturbance Errors, Proc. ISCA 2014.

Kodovi za otkrivanje i popravku grešaka

Zasnivaju se na **dodavanju (redundantnih) informacija** u podatak

Dodavanjem **bita provere** (engl. *check bit*) podatku se dobija **kodna reč** (engl. *codeword*)

Svi bitovi kodne reči moraju zadovoljiti neko **pravilo**

Skup kodnih reči je podskup svih kombinacija bitova

Kodovi za otkrivanje i popravku grešaka

Jednostavan način – **bit parnosti** (engl. *parity bit*)

- parna parnost (ukupan broj jedinica je paran)

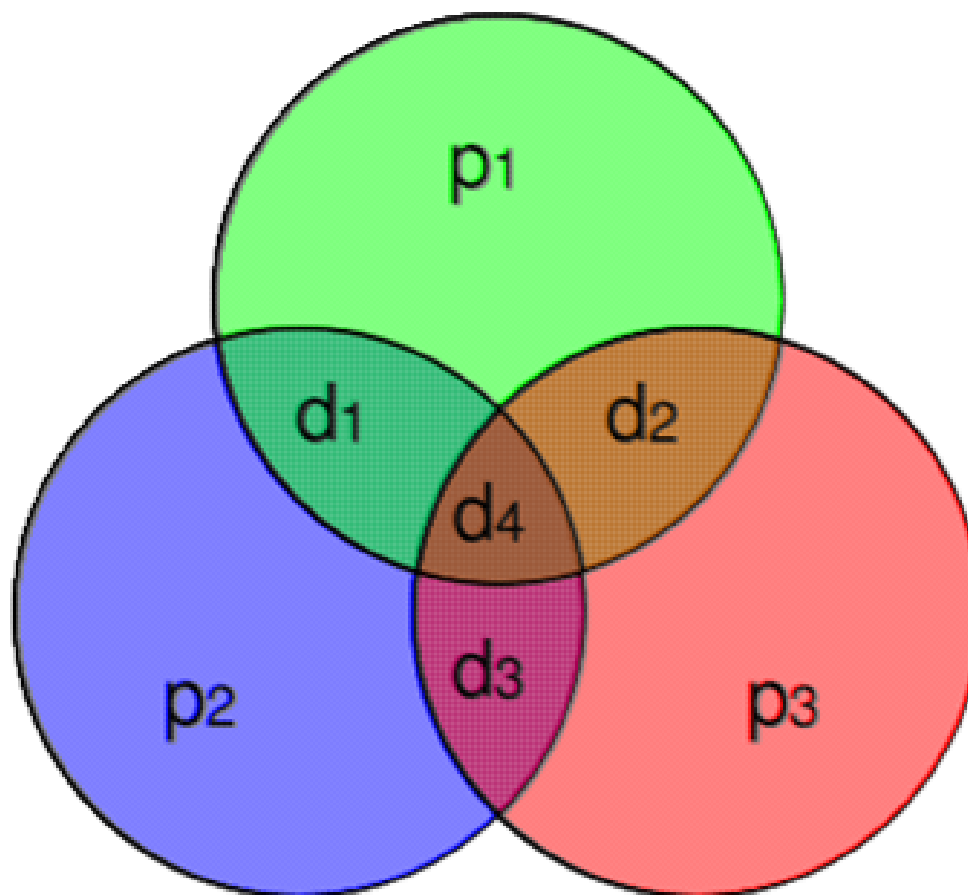
kodne reči	
biti podataka	bit parnosti
00	0
01	1
10	1
11	0

- sve kodne reči se razlikuju u dva bita =>
može detektovati promenu jednog bita

Hamingova udaljenost – broj različitih bitova

Hamingov (7, 4) kod (1950)

Više bita parnosti može omogućiti i korekciju



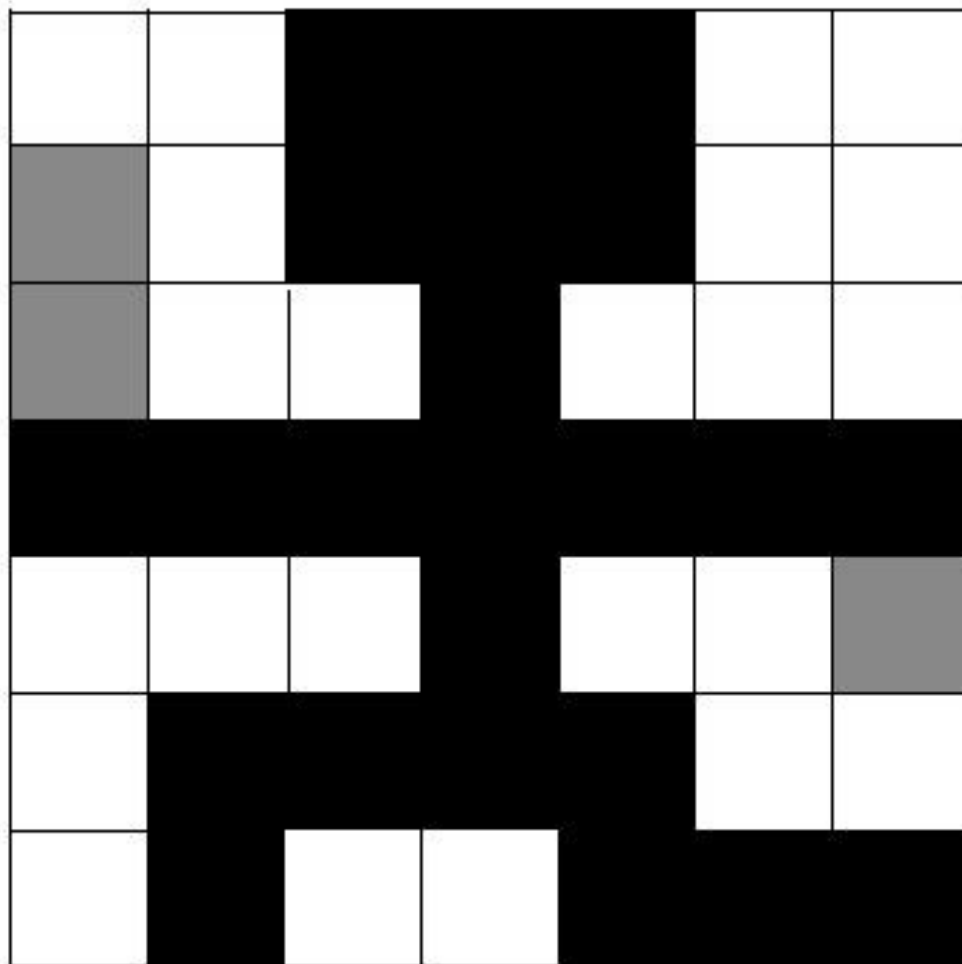
Hamingov (7, 4) kod (1950)

Više bita parnosti može omogućiti i korekciju

	redni brojevi bita kodne reči						
biti parnosti	1	2		4			
biti podataka			3		5	6	7
podskup 1. bita parnosti	→		+		+		+
podskup 2. bita parnosti		→	+			+	+
podskup 3. bita parnosti				→	+	+	+
primer kodne reči	0	1	1	0	0	1	1

1. ako je netačan samo jedan od bita parnosti, tada je on pogrešan
2. ako su netačni bita parnosti 1 i 2, tada je pogrešan bit podataka 3
3. ako su netačni bita parnosti 1 i 4, tada je pogrešan bit podataka 5
4. ako su netačni bita parnosti 2 i 4, tada je pogrešan bit podataka 6, a
5. ako su netačni bita parnosti 1, 2 i 4, tada je pogrešan bit podataka 7

Reed-Muller-ov kod i Mariner 9



000000 – bela

000111 – siva

111111 – crna

Kodne reči:

- dužine 32 bita
- ukupno 64 različitih
- Hamingova distanca 16
(ispravlja do 7 grešaka)

Slika sa 64 nivoa sive – 6 bitova + 26 bitova za ECC

Operativni sistem

Operativni sistem

Rad sa BIOS-om i komandnim režimom oslonjenim na BIOS podrazumeva

- poznavanje šta je zauzeto, a šta slobodno u radnoj memoriji
- poznavanje šta je zauzeto, a šta slobodno u masovnoj memoriji
- poznavanje mašinskog formata naredbi

Operativni sistem

Mnogo je lakše

- čuvati podatke i programe u obliku datoteka
 - datoteke sa podacima
 - datoteke sa programima
- pokretati programe bez ulaženja u detalje kao što su:
 - gde će se program smestiti u memoriji
 - koje naredbe ga čine

Koncept **procesa** (engl. *process*)

- **angažovanje procesora koje daje neki rezultat**

Koncept **datoteke** (engl. *file*)

- **razdvaja upotrebu sadržaja datoteke od načina organizacije**

Struktura operativnog sistema

Modul za rukovanje datotekama

- podržava operacije za rad sa datotekama: stvaranje, brisanje, čitanje, pisanje...
- omogućava razlikovanje datoteka putem naziva
- deskriptor datoteke: sadrži attribute datoteke
 - naziv
 - veličina
 - redni brojevi blokova
 - vreme nastanka, izmene, prava pristupa, ...
 - ...

Struktura operativnog sistema

Modul za rukovanje procesima

- podržava operacije za rad sa procesima: stvaranje, pokretanje, uništavanje
- slika procesa
 - naredbe koje čine program
 - vrednosti promenljivih
 - sadržaj steka
- deskriptor procesa: sadrži attribute procesa
 - broj lokacija za smeštanje
 - evidencija zauzetih lokacija
- inicijalna slika procesa se nalazi u izvršnoj datoteci

Struktura operativnog sistema

Modul za rukovanje radnom memorijom

- neophodan za rad prethodna dva modula
- rukovanje slobodnim i zauzetim lokacijama

Modul za rukovanje kontrolerima

- skup drajvera

modul za rukovanje procesima
modul za rukovanje datotekama
modul za rukovanje radnom memorijom
modul za rukovanje kontrolerima

Iznad OS-a su korisnički programi

Interpreter komandi operativnog sistema

Kada se uvedu datoteke i procesi, više nije poželjno da se

- direktno pristupa lokacijama memorije
- direktno pristupa blokovima na disku

Interpreter ostaje sa samo jednom funkcijom

- **pokretanje zadatog programa** (putem naziva)
 - preuzimanje imena izvršne datoteke
 - modul za rad sa kontrolerima
 - pokretanje programa
 - modul za rukovanje datotekama
 - modul za rukovanje procesima i radnom memorijom

Interpreter komandi operativnog sistema

Spada u korisničke programe

- izvršavanje se oslanja na OS
- OS se prema korisničkim programima odnosi kao prema svojim potprogramima

Dva nivoa korišćenja OS-a

- **interaktivni**
- **programski**
 - pozivanje operacija modula operativnog sistema – **sistemski pozivi**

Sistemske programi

- **editor**
- **makro pretprocesor**
- **prevodilac** (engl. *assembler/compiler*)
- **povezivač** (engl. *linker*)
- **punilac** (engl. *loader*)
- **dibager** (eng. *debugger*)
- pomoćni programi za rad sa datotekama

BIOS i OS

Računar započinje rad izvršavanjem BIOS-a

Inicijalni punilac (engl. *bootstrap loader*)

- obično se nalazi u nultom bloku diska (engl. *boot block*)
 - MBR – *Master Boot Record*
 - GPT – *GUID Partition Table*
- BIOS (nakon početnih inicijalizacija računara) učitava nulti blok, smešta ga u memoriju i pokrene
- puni u radnu memoriju preostale delove OS-a

Više operativnih sistema – *multiboot*

Promena konteksta – preključivanje (engl. *context switch*)

Promena konteksta

Izbegavanje radnog čekanja

Višeprocetni režim rada

- više slika procesa istovremeno u memoriji
- prebacivanje (preključivanje) procesora sa jedne na drugu sliku

Stanja procesa

- aktivan
- čeka
- spreman

Sistemske procese

- aktivan kada svi ostali čekaju

Promena konteksta

Do promene konteksta dolazi

- kada se završi aktivnost procesa
- kada aktivnost procesa zavisi od spoljašnjeg događaja
 - komunikacija sa diskom
 - komunikacija sa terminalom
- UI vođeno preključivanje – obavljaju ga drajveri

Modul za rukovanje procesorom

- bira proces kome će se dodeliti procesor

modul za rukovanje procesima
modul za rukovanje datotekama
modul za rukovanje radnom memorijom
modul za rukovanje kontrolerima
modul za rukovanje procesorom

Promena konteksta

- Ako su svi procesi nezavisni (zasebna memorija i datoteke), zajednički resurs su samo registri procesora, svakom procesu se dodeljuje registarski bafer
- %0 – bafer tekućeg aktivnog procesa, %1 – bafer novog aktivnog procesa

IZBACI	POČETAK	preključivanje	IZBACI	%12
	MAKRO	R	IZBACI	%13
	PREBACI_RP	R, (%0)	IZBACI	%14
	DODAJ_1	%0	IZBACI	%15
	KRAJ		UBACI	%2
UBACI	MAKRO	R	UBACI	%3
	PREBACI_PR	(%1), R	UBACI	%4
	DODAJ_1	%1	UBACI	%5
	KRAJ		UBACI	%6
			UBACI	%7
preključivanje:	IZBACI	%2	UBACI	%8
	IZBACI	%3	UBACI	%9
	IZBACI	%4	UBACI	%10
	IZBACI	%5	UBACI	%11
	IZBACI	%6	UBACI	%12
	IZBACI	%7	UBACI	%13
	IZBACI	%8	UBACI	%14
	IZBACI	%9	UBACI	%15
	IZBACI	%10	NATRAG	
	IZBACI	%11	KRAJ	

Prekid

(engl. *interrupt*)

Prekid

Provera spoljašnjih događaja samo prilikom preključivanja nije efikasna

- do reakcije na spoljašnji događaj dolazi tek kada dođe trenutak preključivanja, iako se događaj mogao desiti i ranije

Dešavanje spoljašnjeg događaja treba da odmah pokrene izvršavanje odgovarajućeg drajvera

- prekid izvršavanja tekućeg procesa
- obrada događaja
- nastavak aktivnosti prekinutog procesa

Prekid

Prekid (engl. *interrupt*)

- **obrađivač prekida** (engl. *interrupt handler*)
 - preko vektora prekida (ulazna adresa obrađivača)
- svaka vrsta prekida (tastatura, disk, ...) ima svoj vektor i obrađivač
- prekide izazivaju kontroleri
 - javi procesoru da se desio događaj
 - dostavi vektor obrađivača prekida

Mehanizam prekida

Svi vektori prekida čine **tabelu vektora prekida** kojoj se pristupa preko broja prekida

Linija najave prekida (engl. IRQ - *interrupt request*)

- kontroler javlja da se desio događaj

Linija potvrde prekida (engl. *interrupt acknowledge*)

- procesor traži broj vektora

Čuvanje programskog brojača (%I3) i status registra (%I4) pre obrade prekida

Obradivač prekida čuva preostale registre

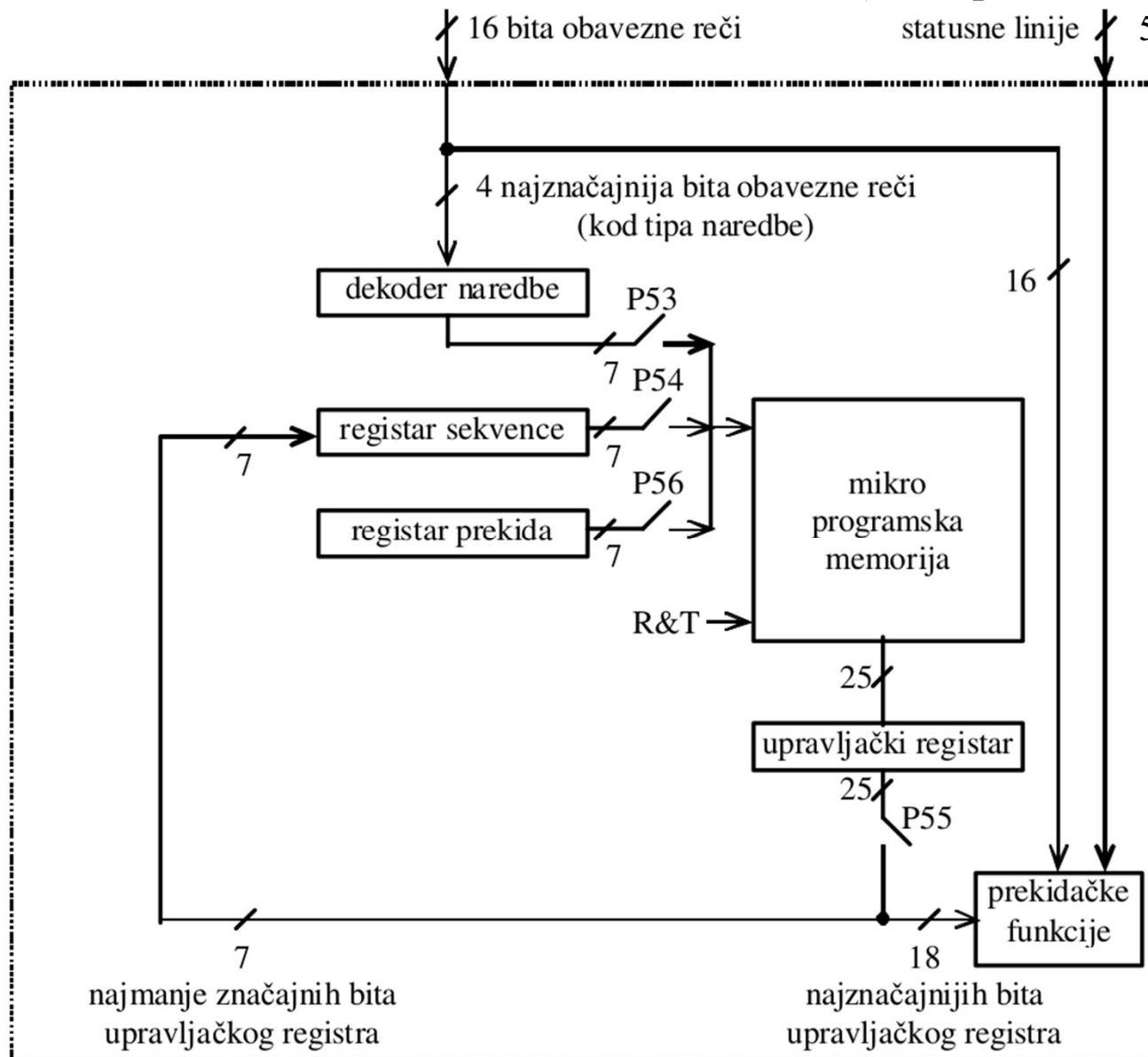
Mehanizam prekida

Obradivači prekida po završetku treba da restauriraju i programski brojač i status registar

- naredba NASTAVI
- kod KONCEPT-a nema prekida unutar prekida
- SR_4 – bit prekida (I-omogućeni)
- postavlja se na 0 čim se uđe u obradu prekida

Stek omogućava prekide u više nivoa

Realizacija prekida



Upravljačka jedinica

registar prekida:
sadrži adresu mikro-
programa prekida

Realizacija prekida

Logička promenljiva NAJAVA

Izvršavanje mikro programa prekida – pre faze dobavljanja sledeće naredbe

Mikro-program dobavljanja na $000000I_2$

$PRE_DOBAVLJANJA = \sim RS_6 \& \sim RS_5 \& \sim RS_4 \& \sim RS_3 \& \sim RS_2 \& \sim RS_1 \& RS_0$

Mogućnost obavljanja prekida

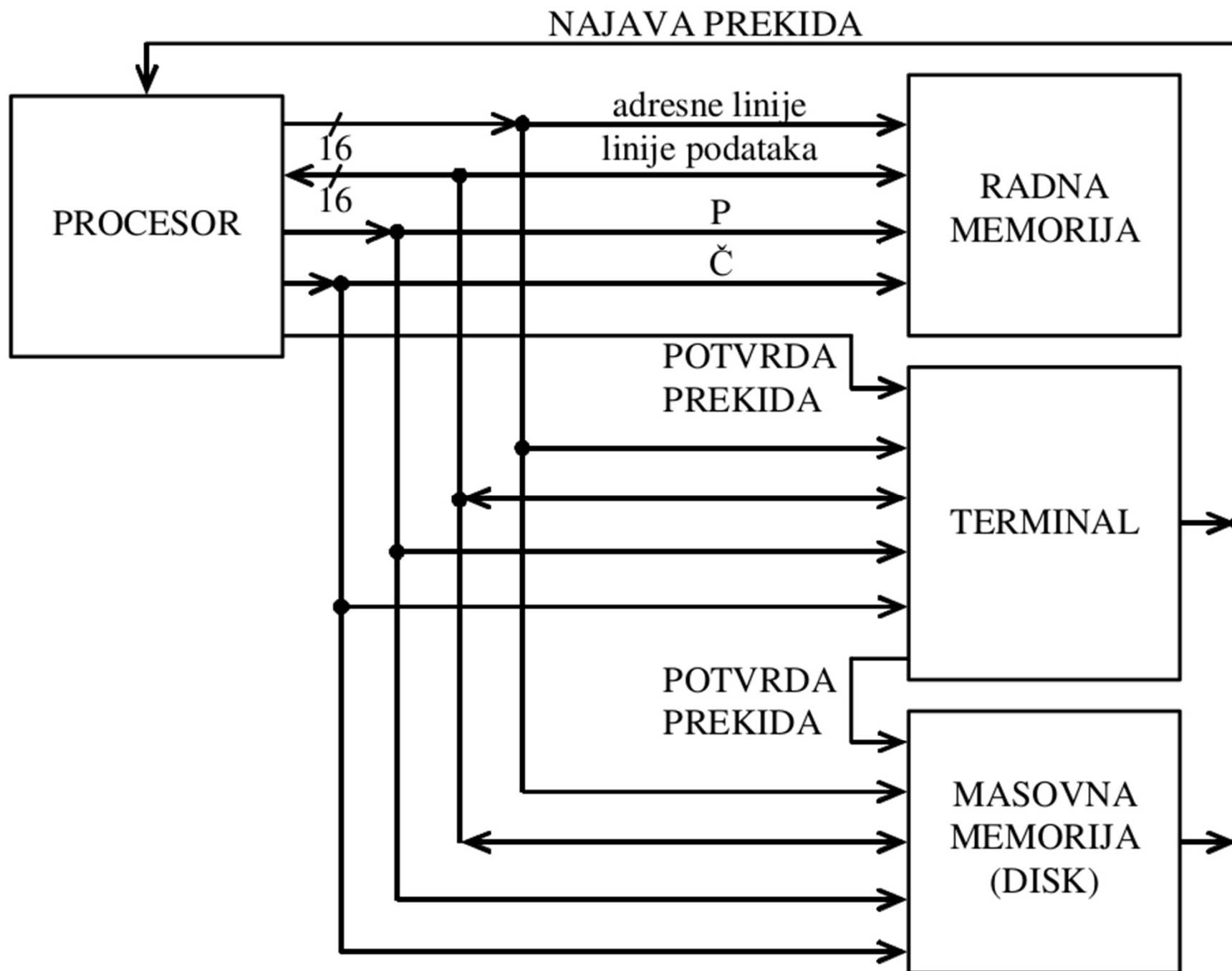
$PREKID = NAJAVA \& SR_4 \& PRE_DOBAVLJANJA$

Rukovanje P56 i P54

$P56 = R \& T \& PREKID$

$P54 = R \& T \& \sim IZA_DOBAVLJANJA \& \sim PREKID$

Serijsko povezivanje kontrolera na liniju potvrde



Realizacija prekida

Svaki kontroler ima u sebi registar broja prekida

- stavlja ga na linije podataka po dobijanju potvrde

Mikro-program prekida

1. ciklus: programski brojač \rightarrow %13
2. ciklus: status registar \rightarrow %14
3. ciklus: 0 \rightarrow SR₄
4. ciklus: 1 \rightarrow POTVRDA PREKIDA
linije podataka \rightarrow pomoćni registar
5. ciklus: pomoćni registar \rightarrow adresne linije
1 \rightarrow č
linije podataka \rightarrow programski brojač

Mikro-program naredbe NASTAVI

1. ciklus: %13 \rightarrow programski brojač
2. ciklus: %14 \rightarrow status registar

Odnos obrade prekida i preključivanja

Obrada prekida ne zahteva preključivanje, ali ga može izazvati

Prioritet procesa

Na početku rada

- inicijalizacija tabele prekida (funkcija modula za rukovanje kontrolerima)
- omogućavanje prekida

Podela drajvera

- donji deo – **obrađivač prekida**
- gornji deo – **komunikacija sa višim slojevima**

Organizacija drajvera terminala

Donji deo

- obrađivač prekida tastature
- obrađivač prekida ekrana

Gornji deo

- potprogrami terminala



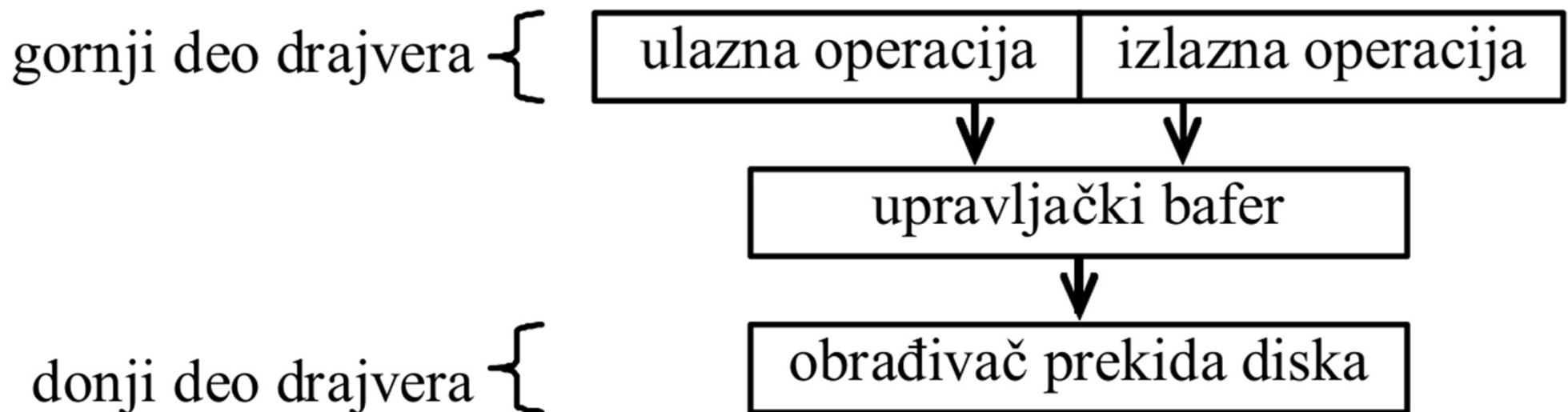
Organizacija drajvera diska

Donji deo

- obrađivač prekida diska

Gornji deo

- ulazna i izlazna operacija (prijavljuju kraj rada tek kada ceo blok bude prenet)



Usklađivanje rada kontrolera i uređaja

Asinhroni rad:

- kontroleri
- procesor

Sinhroni rad:

- kontroler
- uređaj

Rukovanje (engl. *handshaking*)

- logička promenljiva KONTROLER
- logička promenljiva UREĐAJ
- samo kada su obe na 1 moguća je komunikacija