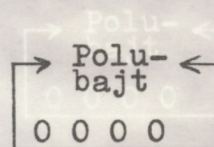


4. Kodiranje numeričkih znakova
4. Kodiranje numeričkih znakova

Pojedinim jezgrima (BITOVIMA) u okviru jedne pozicije Pojedinim jezgrima (BITOVIMA) u okviru jedne pozicije i memorije mogu se dodeliti određene "Vrednosti". Da bi se na pr. memorisala jedna cifra 0-9, potrebna bi bila 4 jezgra sa vrednostima 1, 2, 4, 8. Zbir vrednosti svih jezgara u položaju "1" daje vrednost memorisane cifre.

Cifra	4 jezgra sa vrednostima			
	8	4	2	1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1



Za memorisanje jednog broja potrebno je onoliko polubajtova, koliko ima cifara:

na pr: 520

8	4	2	1	8	4	2	1	8	4	2	1
0	1	0	1	0	0	1	0	0	0	0	0
5				2				0			

Oblik memorisanja: Pakovan — Po polubajtu memoriše se jedna cifra

Kodiranje alfabetskih znakova5. Kodiranje alfabetskih znakova

Memorisanje alfabetskih znakova nije dovoljan jedan bajt.

Za memorisanje alfabetskih znakova nije dovoljan jedan polubajt. Jednog alfabetskog znaka zahteva prema tome 8 bitova = 1 bajt.

Memorisanje jednog alfabetskog znaka zahteva prema tome 8 bitova = 1 bajt. Kodiranje alfabetskih znakova obavlja se analogno kodiranju na bušenoj kartici u 4 grupe:

Kodiranje alfanumeričkih znakova obavlja se analogno kodiranju na bušenoj kartici u 4 grupe:

A-I, J-R, S-Z, 0-9. u tzv. "Proširenom 8-bitnom BCD kodu".

Kodiranje 4 grupe u tzv. "Proširenom 8-bitnom BCD kodu".

Grupa	Kombinacije bušenja		BIT-kombinacija	
	Zona	Cifra	Zona	Cifra
A-I	12	1-9	1100	0001 do 1001
J-R	11	1-9	1101	0001 do 1001
S-Z	0	2-9	1110	0010 do 1001
0-9	nema	0-9	1111	0000 do 1001

Memorisanja: Alfanumerički — memoriše se 1 znak po bajtu

Oblik memorisanja: Alfanumerički — memoriše se 1 znak po bajtu

Primer: 4 DM

Zona	Cifra	Zona	Cifra	Zona	Cifra
1 1 1 1	0 1 0 0	1 1 0 0	0 1 0 0	1 1 0 1	0 1 0 0
4		D		M	
← 1 bajt →					

6. Kodiranje specijalnih znakova

Od 256 mogućnosti za predstavljanje znakova iskorišćene su

26 kombinacija alfabetski znaci

10 kombinacija numerički znaci

Ostale kombinacije omogućavaju predstavljanje specijalnih znakova (videti tabelu o 256 kombinacionih mogućnosti).

PREDSTAVLJANJE 256 BIT-KOMBINACIJA

Gornje polje: Standardni znak

Donje polje: Kombinacija bušenja (Z = 12 vrstica, E = 11 vrstica)

Zonski deo (B-poz. 0-3)		Cifarski deo (Bit-pozicije 4 - 7)															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0	0000	Z09 81	Z91	Z92	Z93	Z94	Z95	Z96	Z97	Z98	Z98 1	Z98 2	Z98 3	Z98 4	Z98 5	Z98 6	Z98 7
1	0001	ZE9 81	E91	E92	E93	E94	E95	E96	E97	E98	E98 1	E98 2	E98 3	E98 4	E98 5	E98 6	E98 7
2	0010	E09 81	091	092	093	094	095	096	097	098	098 1	098 2	098 3	098 4	098 5	098 6	098 7
3	0011	ZE0 981	91	92	93	94	95	96	97	98	981	982	983	984	985	986	987
4	0100	Blank PRAZ NO	Z09 1	Z09 2	Z09 3	Z09 4	Z09 5	Z09 6	Z09 7	Z09 8	Z81	Z82	Z83	Z84	Z85	Z86	Z87
5	0101	Z	ZE9 1	ZE9 2	ZE9 3	ZE9 4	ZE9 5	ZE9 6	ZE9 7	ZE9 8	E81	E82	E83	E84	E85	E86	E87
6	0110	E	01	E09 2	E09 3	E09 4	E09 5	E09 6	E09 7	E09 8	081	ZE	083	084	085	086	087
7	0111	ZE0	ZE0 91	ZE0 92	ZE0 93	ZE0 94	ZE0 95	ZE0 96	ZE0 97	ZE0 98	81	82	83	84	85	86	87
8	1000	Z08 1	Z01	Z02	Z03	Z04	Z05	Z06	Z07	Z08	Z09	Z08 2	Z08 3	Z08 4	Z08 5	Z08 6	Z08 7
9	1001	ZE8 1	ZE1	ZE2	ZE3	ZE4	ZE5	ZE6	ZE7	ZE8	ZE9	ZE8 2	ZE8 3	ZE8 4	ZE8 5	ZE8 6	ZE8 7
A	1010	E08 1	E01	E02	E03	E04	E05	E06	E07	E08	E09	E08 2	E08 3	E08 4	E08 5	E08 6	E08 7
B	1011	ZE0 81	ZE0 1	ZE0 2	ZE0 3	ZE0 4	ZE0 5	ZE0 6	ZE0 7	ZE0 8	ZE0 9	ZE0 82	ZE0 83	ZE0 84	ZE0 85	ZE0 86	ZE0 87
C	1100	Z0	Z1	Z2	Z3	Z4	Z5	Z6	Z7	Z8	Z9	Z09 82	Z09 83	Z09 84	Z09 85	Z09 86	Z09 87
D	1101	E0	E1	E2	E3	E4	E5	E6	E7	E8	E9	ZE9 82	ZE9 83	ZE9 84	ZE9 85	ZE9 86	ZE9 87
E	1110	Blank BUS 082	E091	02	03	04	05	06	07	08	09	E09 82	E09 83	E09 84	E09 85	E09 86	E09 87
F	1111	0	1	2	3	4	5	6	7	8	9	ZE0 982	ZE0 983	ZE0 984	ZE0 985	ZE0 986	ZE0 987

PREDSTAVLJANJE PREDZNAKA

7. Primeri za oblike memorisanja

alfanumerički

1100	1001	1100	0010	1101	0100	0110	0001	1111	0011	1111	0110	1111	0000
I	B	M	/	3	6	0							
BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT

0000	0000	0000	0000	0000	0000	1111	1000	1111	0111	1111	0110	1101	0100
0	0	0	0	0	0	8	7	6	4				
BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT

decimalni pakovani

0000	0000	0000	0000	0000	0000	0000	0000	0000	1000	0111	0110	0100	1101
00	00	00	00	00	00	00	00	00	00	00	00	00	00
BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT

1101 0100 → M ili nepakovano -4

PREDSTAVLJANJE PREDZNAKA

Bit-kombinacija	Predznak
1100 (1010, 1110,)	+ CAFE
1101 (1011)	Dopuna uz nepakovan broj
1111	Dopuna uz nepakovan broj

UVOD U BROJNE SISTEME
UVOD U BROJNE SISTEME

1. Decimalni brojni sistem

1. Decimalni brojni sistem

Predstavljanje: Moguće cifre 0 do 9 (= 10 cifara)

Predstavljanje: Moguće cifre 0 do 9 (= 10 cifara)

Baza 10 cifre u broju pokazuje

Položaj cifre u broju pokazuje

njenu mesnu vrednost

Mesna vrednost	10^5	10^4	10^3	10^2	10^1	10^0
Mesna vrednost	10^5	10^4	10^3	10^2	10^1	10^0
Mesna vrednost	100 000	10 000	1 000	100	10	1
467	0x100 000	0x10 000	0x1 000	4x100	6x10	7x1
467	0x100 000	0x10 000	0x1 000	4x100	6x10	7x1
	0	0	0	4	6	7

Deci-
Deci-
malni broj

2. Dualni brojni sistem

2. Dualni brojni sistem

a) Predstavljanje: Moguće cifre 0 i 1 (= 2 cifre)

a) Predstavljanje: Moguće cifre 0 i 1 (= 2 cifre)

Baza 2

Mesna vrednost	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Mesna vrednost	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Mesna vrednost	256	128	64	32	16	8	4	2	1
467	1x256	1x128	1x64	0x32	1x16	0x8	0x4	1x2	1x1
467	1x256	1x128	1x64	0x32	1x16	0x8	0x4	1x2	1x1
	1	1	1	0	1	0	0	1	1

Dual-
Dual-
ni broj

Provera: $256 + 128 + 64 + 16 + 2 + 1 = 467$

Provera: $256 + 128 + 64 + 16 + 2 + 1 = 467$

b) Pravila računanja u dualnom sistemub) Sabiranje

	$0 + 0 = 0$	
	$0 + 1 = 1$	
Sabiranje	$10 + 00 = 10$	
	$10 + 11 = 01$	sa prenosom na sledeće više dualno mesto
	$1 + 0 = 1$	
	$1 + 1 = 0$	sa prenosom na sledeće više dualno mesto
Oduzimanje	$0 - 0 = 0$	
	$1 - 0 = 1$	
Oduzimanje	$00 - 10 = 10$	
	$11 - 10 = 01$	sa pozajmljivanjem iz predhodnog višeg dualnog mesta
	$0 - 1 = 1$	
	$1 - 1 = 0$	sa pozajmljivanjem iz predhodnog višeg dualnog mesta
Množenje	$0 \times 0 = 0$	
	$1 \times 0 = 0$	
Množenje	$0 \times 10 = 00$	
	$1 \times 10 = 10$	
	$0 \times 1 = 0$	

c) Pretvaranje decimalnog broja u dualni brojc) Pretvaranje 467 decimalnog broja u dualni broj

233	: 2 = 116	ostatak 1
116	: 2 = 58	ostatak 0
58	: 2 = 29	ostatak 0
29	: 2 = 14	ostatak 1
14	: 2 = 7	ostatak 0
7	: 2 = 3	ostatak 1
3	: 2 = 1	ostatak 1
1	: 2 = 0	ostatak 1

1 1 1 0 1 0 0 1 1

1 1 1 0 1 0 0 1 1

MEMORISANJE I OBRADA DUALNIH BROJEVA

1. Aritmetika fiksnog zareza

Predstavljanje dualnih brojeva u feritnoj memoriji je svakako moguće.

Dualne brojeve moguće je obradivati u sledećim fiksnim veličinama:

polureč	= 2 bajta = 16 bitova	(= 15 dualnih mesta + predznak)
reč	= 4 bajta = 32 bitova	(= 31 " " + ")
dupla reč	= 8 bajta = 64 bitova	(= 63 " " + ")

Primer: "1295"

				00000000	00000000	00000101	00001111
BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT	BAJT
polureč		polureč		polureč		polureč	
reč				reč			
dupla reč							

Obrada dualno memorisanih brojeva izvodi se u posebnim računskim delovima = registrima.

Prednosti dualne obrade:

Kod višemesnih brojeva traži dualno kodiran broj manje bitova kao pakovano kodiran.

(2 bajtova = 1 polureč → predstavljanje brojeva od
0 do ± 32 767 ($=2^{15}-1$))

(4 bajtova = 1 reč → predstavljanje brojeva od
0 do ± 2.147.483.647 ($=2^{31}-1$))

Obrada u registrima posebno se odlikuje po velikim brzinama.

2. Aritmetika kliznog zareza

a) Principijelno predstavljanje

$$+ 512 = + 10^3 \times 0,512$$

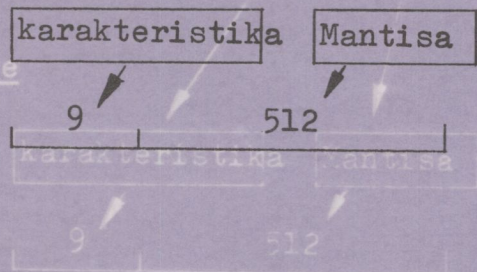
a) Principijelno predstavljanje

$$+ 0,000512 = + 10^6 \times 0,512$$

$$+ 512\ 000\ 000 = + 10^9 \times 0,512$$

$$+ 0,000512 = + 10^6 \times 0,512$$

b) Pojednostavljeno predstavljanje



c) Predstavljanje predznaka

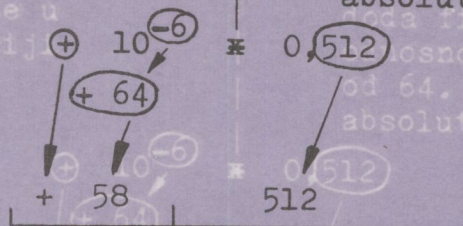
c) Predstavljanje predznaka

$$+ 10^6 \times 0,512$$

Predznak vrednosti

Memorisanje kao kod fiksne zapete u 1. bit-poziciji

Memorisanje kao kod fiksne zapete u 1. bit-poziciji



Predznak eksponenta

Kod pozitivnog eksponenta se doda fiksni vrednosti 64 odnosno kod negativnog oduzme od 64. Ne postoji memorisanje absolute vrednosti eksponenta.

Kod pozitivnog eksponenta se doda fiksni vrednosti 64 odnosno kod negativnog oduzme od 64. Ne postoji memorisanje absolute vrednosti eksponenta.

d) Formati sa kliznim zarezom u /360

Instrukcije kliznog zareza obrađuju samo brojeve u dualnom obliku. Broj sa kliznim zarezom u dualni predstavi izgleda:

d) Formati sa kliznim zarezom u /360

	PR	Karakt.	Mantisa	običajna tačnost
broj bitova	1	7	24	
	PR	Karakt.	Mantisa	dupla tačnost
broj bitova	1	7	56	
	PR	Karakt.	Mantisa	dupla tačnost
broj bitova	1	7	56	

Predstavljanje broja u heksadecimalnom obliku

1 polovina bajta = 4 dualna mesta, omobučavaju 16 različitih kombinacija. Svih 16 mogućnosti zamenimo sa jednim JEDNOMESTNIM BROJEM:

Za prvih 10 kombinacija uzmemo brojke od 0 do 9, za preostalih 6 kombinacija nam služe slova od A do F.

Dualno predstavljanje po polovini byta	Heksadecimalno predstavljanje
0000	0 ₁₆
0001	1 ₁₆
0010	2 ₁₆
0011	3 ₁₆
0100	4 ₁₆
0101	5 ₁₆
0110	6 ₁₆
0111	7 ₁₆
1000	8 ₁₆
1001	9 ₁₆
1010	A ₁₆
1011	B ₁₆
1100	C ₁₆
1101	D ₁₆
1110	E ₁₆
1111	F ₁₆

Heksadecimalno predstavljanje (16 mogućih kombinacija = heksadecimalni sistem)

Primer: 467_{10} kao broj u dualnom obliku

$0000\ 0001\ 1101\ 0011_2$

u heksadecimalnom obliku

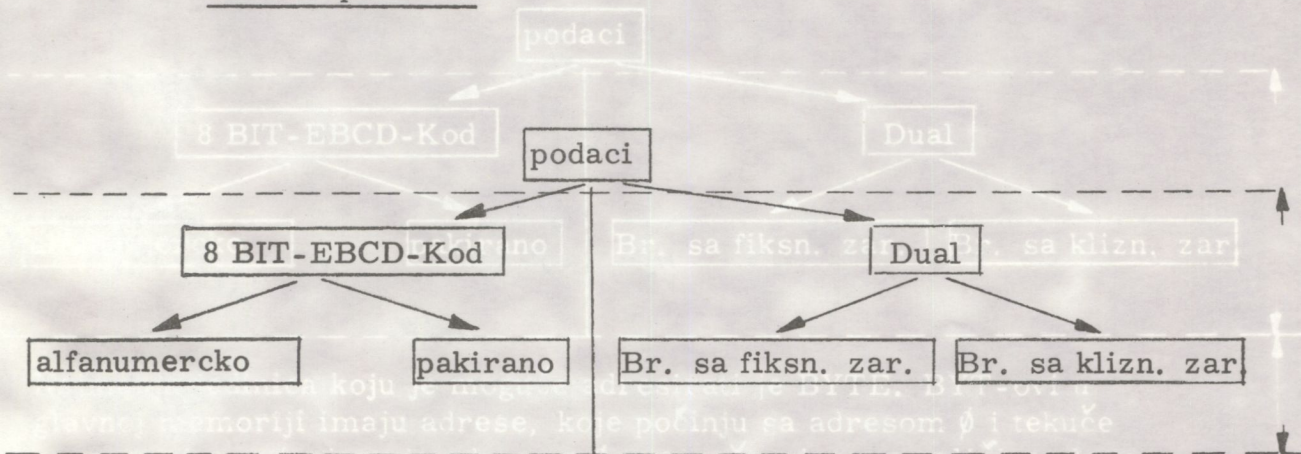
$0\ 1\ D\ 3_{16}$

Svrha: Svaku moguću bit-kombinaciju feritne memorije moguće je ispisati u heksadecimalnom obliku koji zauzima pola bajta.

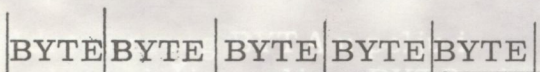
obrada podataka

Pregled oblika memoriziranja - adresiranje

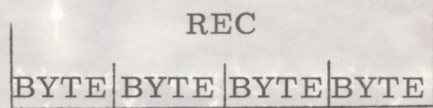
obrada podataka



Najmanja jedinica koju je moguće adresirati je BYTE. BYT-ovi u glavnoj memoriji imaju adrese, koje počinju sa adresom 0 i tekuće su numerisane (najmanja adresa = 0, najveća adresa = veličina glavne memorije).



Adresa prvog BYTA u polju i podatak dužine polja u BYTO-vima su sastavni delovi instrukcije.

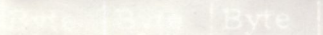


Adresa prvog byta polja obrada pola reči, reč, dupla reč sadržana je u instrukciji.

Kako se vrši memorisanje odnosno obrada?

promenlj.

Promenljiv broj bytova.



u glavnoj memoriji u željenim pozicijama glavne memorije.

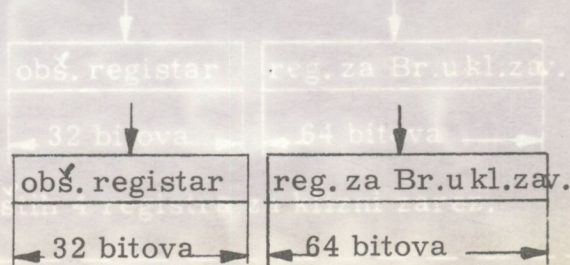
u samoj glavnoj mamoriji u željenim pozicijama glavne memorije.

fiksna

Samo 1 fiksna ili 1 ili 1 dupla reč.

u pos Samo 1 polureč ili 1 ili 1 dupla reč, skim registrima.

u posebnim za to predviđenim računskim registrima.



16 opštih 4 registra za klizni zarez.

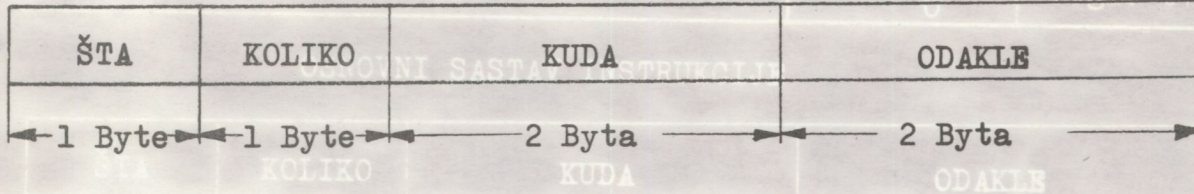
OSNOVNI SASTAV INSTRUKCIJE

I B M ŠKOLA

P O D S E T N I K

/360-30

Pogl. 070

Instrukcija označava:

- ŠTA** treba uraditi (kod operacije)
- KOLIKO** bytova treba obraditi (dužina polja)
- KUDA** odlaze podaci (adresa 1. polja)
- ODAKLE** dolaze podaci (adresa 2. polja)

ŠTA Operacioni kod daje:

- koja operacija se izvodi,
- da li su podaci za obradu fiksne ili promenljive dužine,
- da li su podaci u decimalnom ili dualnom obliku,
- da li su podaci u registru ili u glavnoj memoriji, koliko je duga instrukcija.
- da li su podaci u decimalnom ili dualnom obliku,

KOLIKO Dužina polja daje:

- koliko bytova se obrađuje tom instrukcijom.
- Kod instrukcija, koje imaju podatak fiksne dužine polja, otpada taj podatak.

KUDA Adresa prvog polja označava mesto memorije, gde se posle instrukcije nalazi rezultat.

ODAKLE Adresa drugog polja označava mesta memorije, gde stoje podaci za obradu.

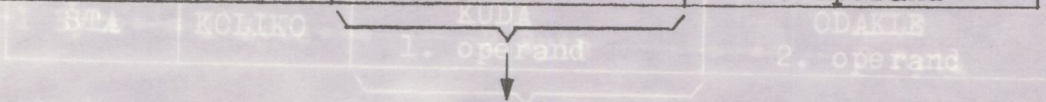
Oba polja mogu se nalaziti ili u glavnoj memoriji ili u registrima. Ako imamo neki podatak memorisan u registru, onda je njegova adresa broj registra (Ø do 15).

Oba polja mogu se nalaziti ili u glavnoj memoriji ili u registrima. Ako imamo neki podatak memorisan u registru, onda je njegova adresa broj registra (Ø do 15).

ADRESIRANJE GLAVNE MEMORIJE

ADRESIRANJE GLAVNE MEMORIJE

ŠTA	KOLIKO	KUDA 1. operand	ODAKLE 2. operand
-----	--------	--------------------	----------------------



2 Byta = 4 polubyta = 16 bitova

1. Mogućnost: Adresa navedenog polja se direktno navodi u instrukciji.

Adresa KUDA = 2 byta → max.adresa = $2^{16} - 1 = 65.535$
 Adresa ODAKLE = 2 byta → max.adresa = $2^{16} - 1 = 65.535$

Nedostatak: Ograničena mogućnost adresiranja

2. Mogućnost: Adresa se izračunava iz podataka u polju KUDA/ODAKLE (način rada /360).

ŠTA	KOLIKO	KUDA		ODAKLE	
KOD OPERACIJE	DUŽINA POLJA	ADRESA PRVOG POLJA		ADRESA DRUGOG POLJA	
OP	L	BROJ BASIS REGISTRA	RELATIVNA ADRESA	BROJ BASIS REGISTRA	RELATIVNA ADRESA
		B ₁	D ₁	B ₂	D ₂

8 bitova

8 bitova

4 bitova

12 bitova

$2^4 = 16$ mogućnosti

$2^{12} = 4\ 096$ mogućnosti

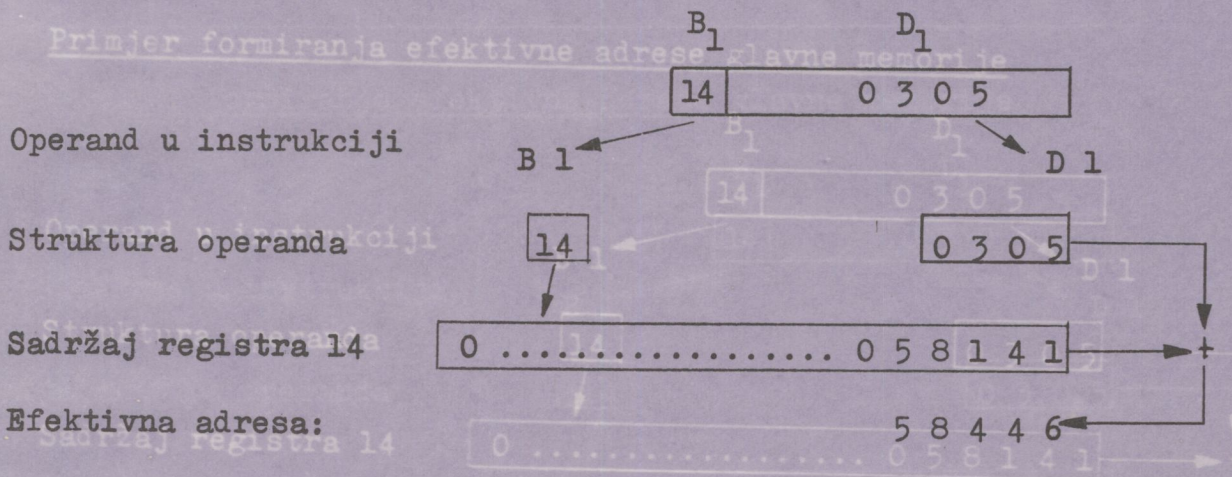
Adresa jednog registra (Ø do 15) upotrebljenog kao BASIS registar. Sadrži BASIS-ADRESU.

Relativna adresa = DISPLACEMENT od Ø do 4095

PRAVILA ZA RAČUNANJE ADRESA:

Sadržaj navedenog Basis registra (BASIS-ADRESA)
 + Relativna adresa (DISPLACEMENT)
 = Efektivna adresa glavne memorije
 + Relativna adresa (DISPLACEMENT)
 = Efektivna adresa glavne memorije

Primjer formiranja efektivne adrese glavne memorije



Maksimalna mogućnost adresiranja: 1 registar = 32 bita
 maksimal.adresa $2^{32} - 1 =$

Maksimalna mogućnost adresiranja: = 4.294.967.296 bita

Od toga za sada upotrebljenih, kod adresiranja sa basis-registrom:
 24 bita \rightarrow najveća adresa $2^{24} - 1 = 16.777.215$.

Od toga za sada upotrebljenih, kod adresiranja sa basis-registrom:

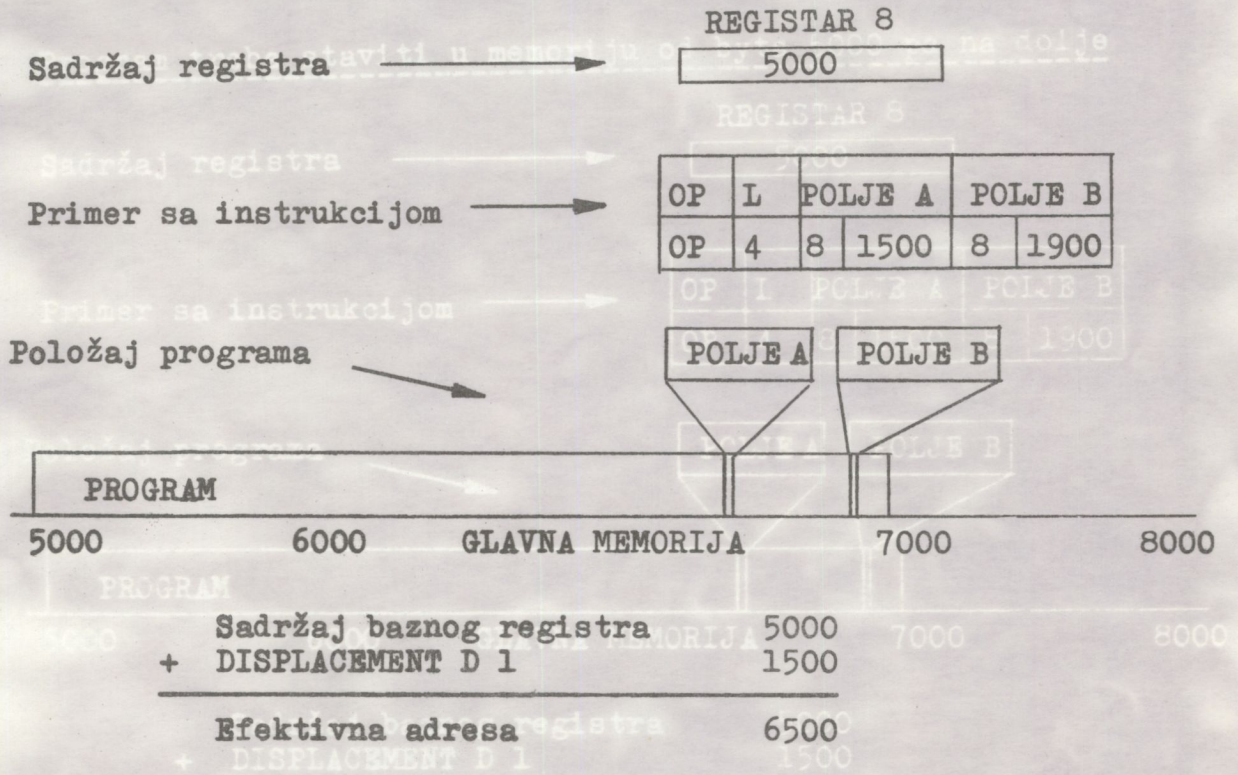
PREDNOSTI adresiranja sa BASIS-REGISTROM:

- Značajno proširena mogućnost adresiranja
- Pomerljivost programa
- Značajno proširena mogućnost adresiranja
- Kod promjene BASIS ADRESE promjeni se mesto programa u glavnoj memoriji.

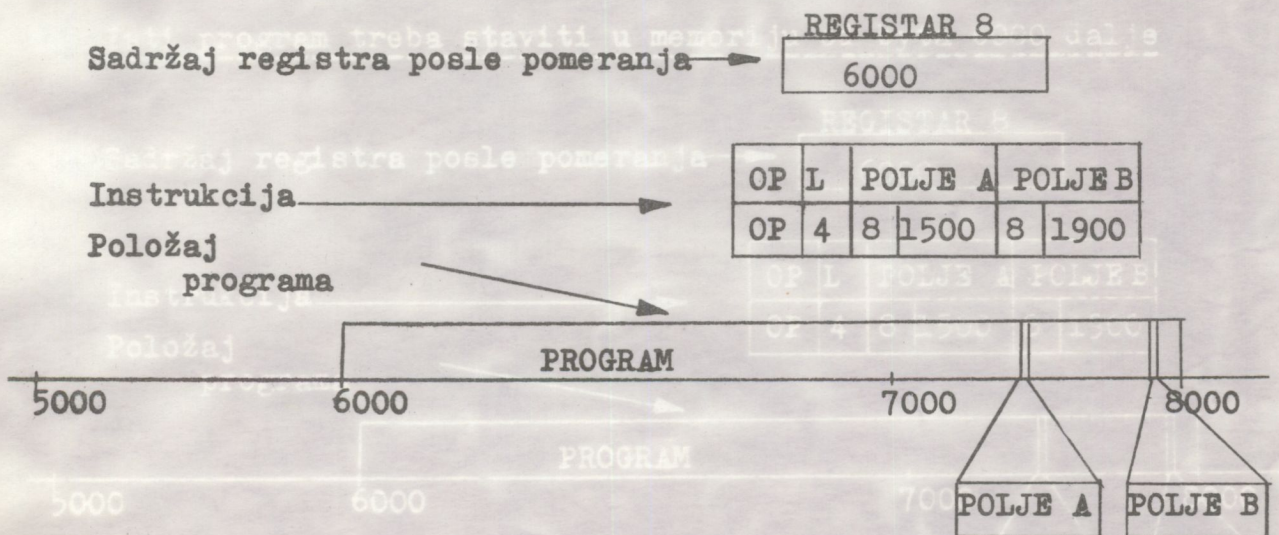
Kod promjene BASIS ADRESE promjeni se mesto programa u glavnoj memoriji.

Primer za pomeranje programa

A. Program treba staviti u memoriju od byta 5000 pa na dalje

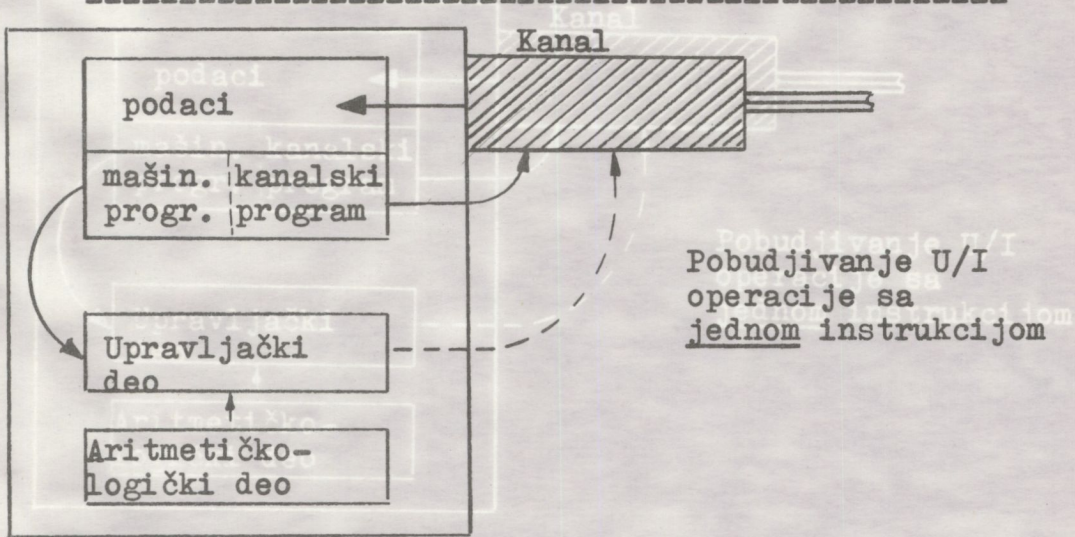


B. Isti program treba staviti u memoriju od byta 6000 dalje



Pomeranje se izvodi samo promenom BASIS-ADRESE u BASIS-REGISTRU

Pomeranje se izvodi samo promenom BASIS-ADRESE u BASIS-REGISTRU

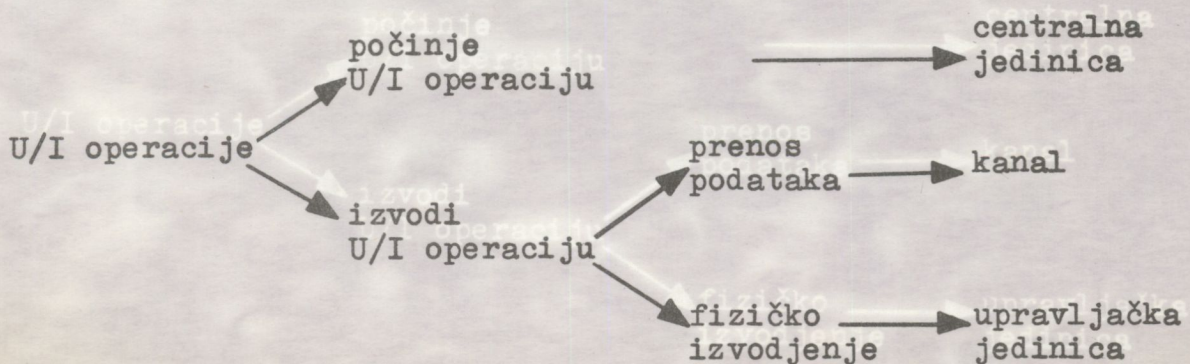
b) Kanali sa potpuno samostalnim radom = kanali /360Centralna jedinica:

Daje sa instrukcijom (START U/I) kanalu sledeće informacije:

- da se izvodi bilo koji U/I zadatak
- koju U/I jedinicu treba da nazove
- gde u glavnoj memoriji se nalazi kanalski program (CCW)

Kanal:

Unosi pojedine U/I instrukcije (kanalske instrukcije) simultano u kanalski upravljački deo i izvodi instrukcije samostalno, paralelno sa mašinskim programom. Transport podataka u/iz glavne memorije vrši se istovremeno.

Prednosti: Princip redosleda u sistemu /360

PRINCIP UPRAVLJANJA U/I SISTEMOM

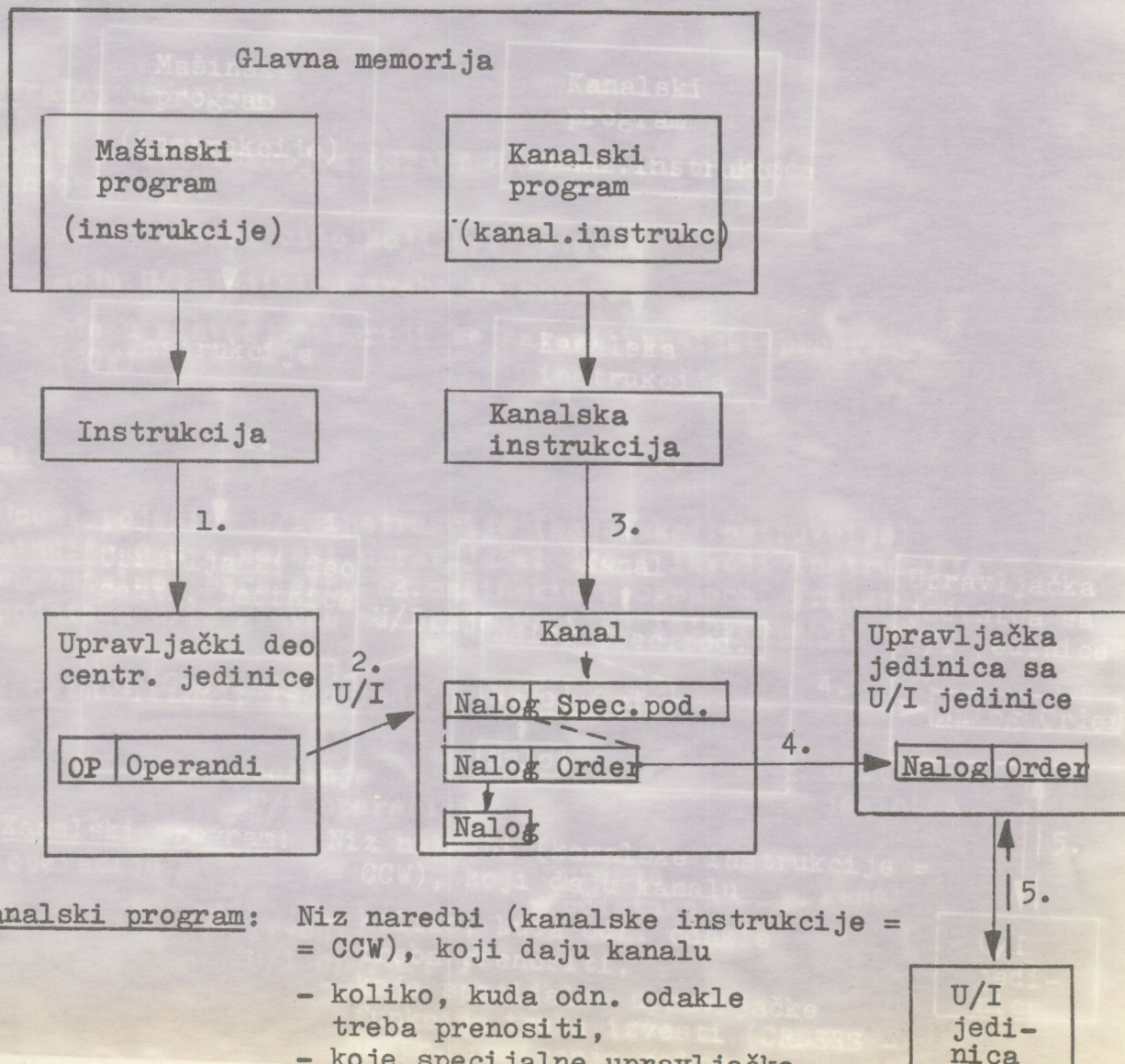
PRINCIP UPRAVLJANJA U/I SISTEMOM

1. Počinjavanje U/I operacije

Pripremanje veze "centralna jedinica-kanal-upravljački deo-U/I jedinica" sa instrukcijom "START U/I" (SIO).

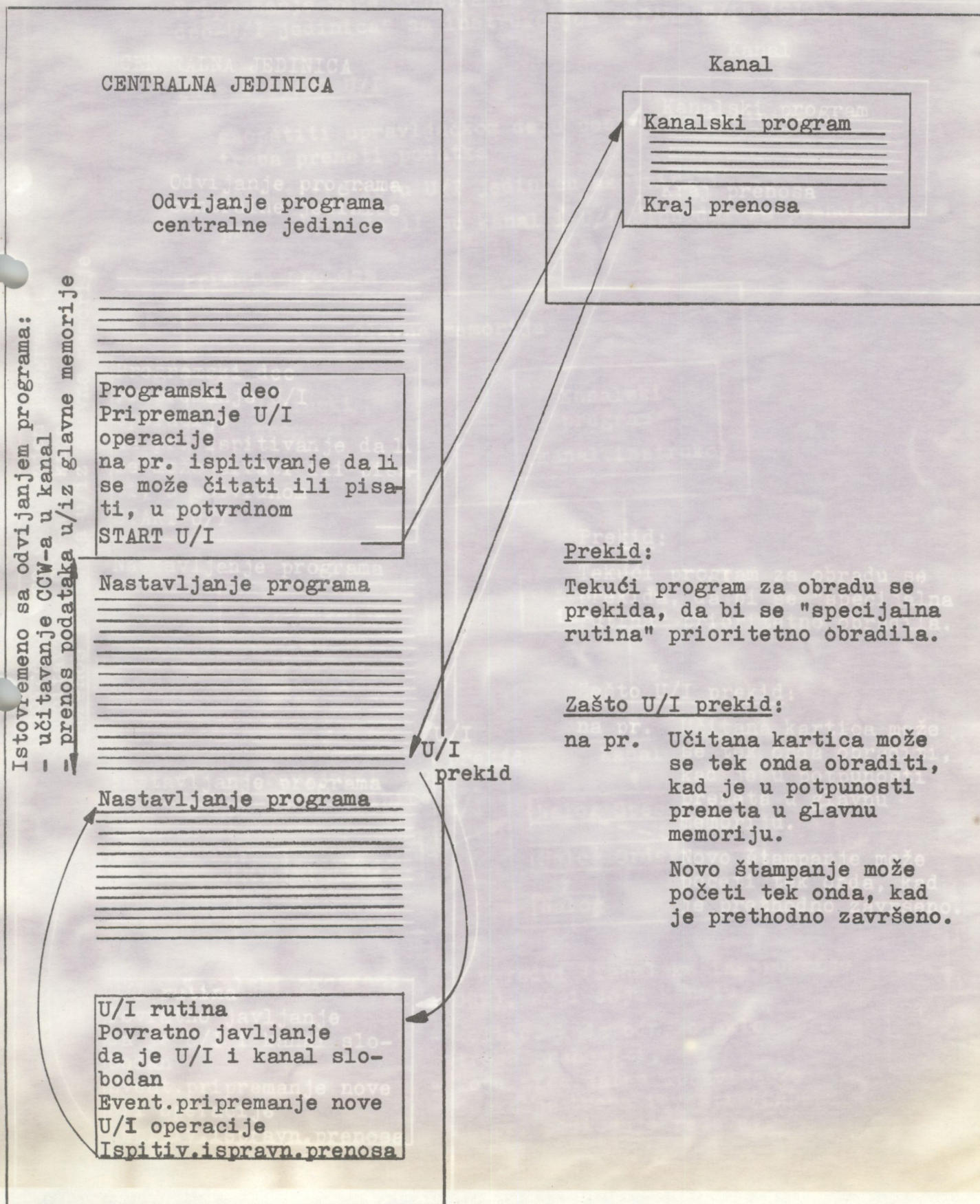
Zadaci START U/I

- saopštiti upravljačkom delu centralne jedinice, da treba preneti podatke
- saopštiti, koju U/I jedinicu aktivirati
- proveriti, da li su kanal i U/I slobodni za prenošenje.

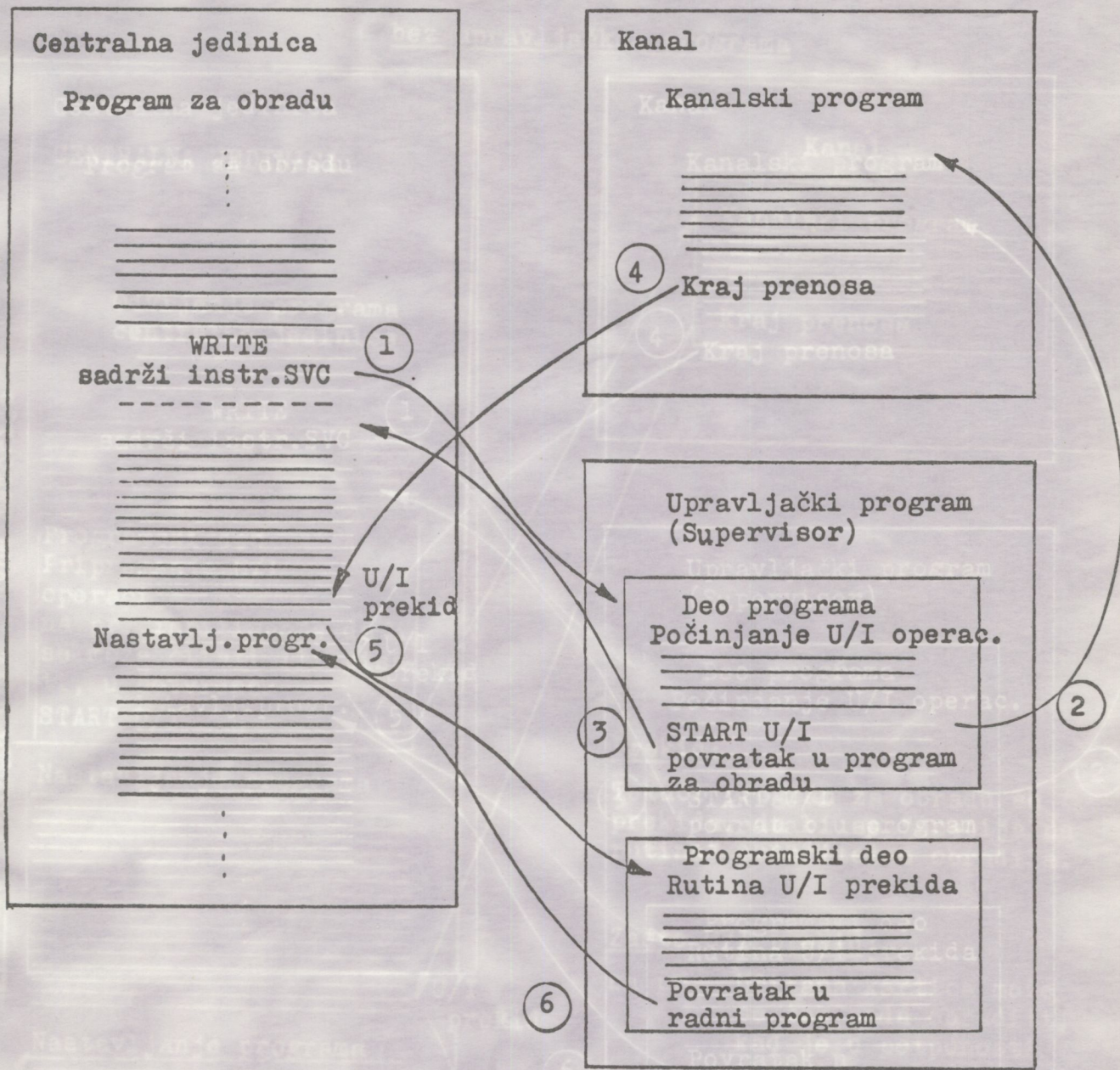
Princip procesa

Kanalski program: Niz naredbi (kanalske instrukcije = CCW), koji daju kanalu

- koliko, kuda odn. odakle treba prenositi,
- koje specijalne upravljačke funkcije treba izvesti (ORDERS - na pr. pomicanje redova).

2. Izvodjenje i završavanje U/I operacijebez upravljačkog programa

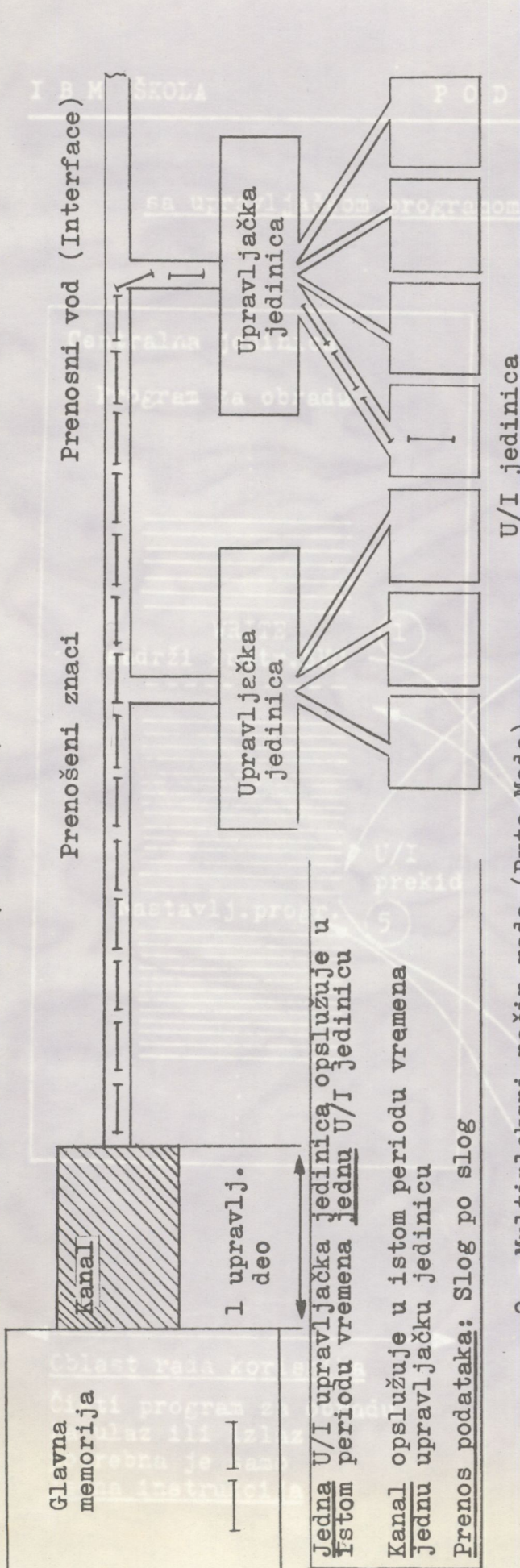
sa upravljačkom programom = metoda kod /360



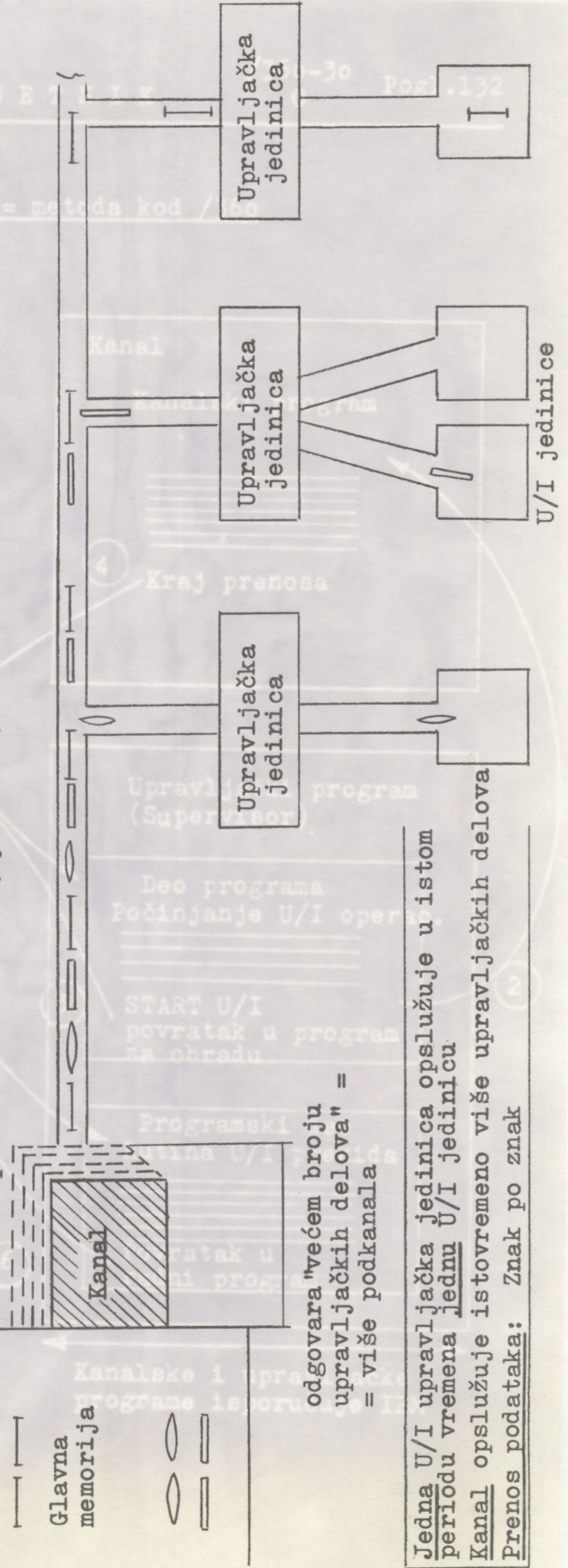
Oblast rada korisnika
Čisti program za obradu.
Za ulaz ili izlaz
potrebna je samo
jedna instrukcija

Kanalske i upravljačke
programe isporučuje IBM

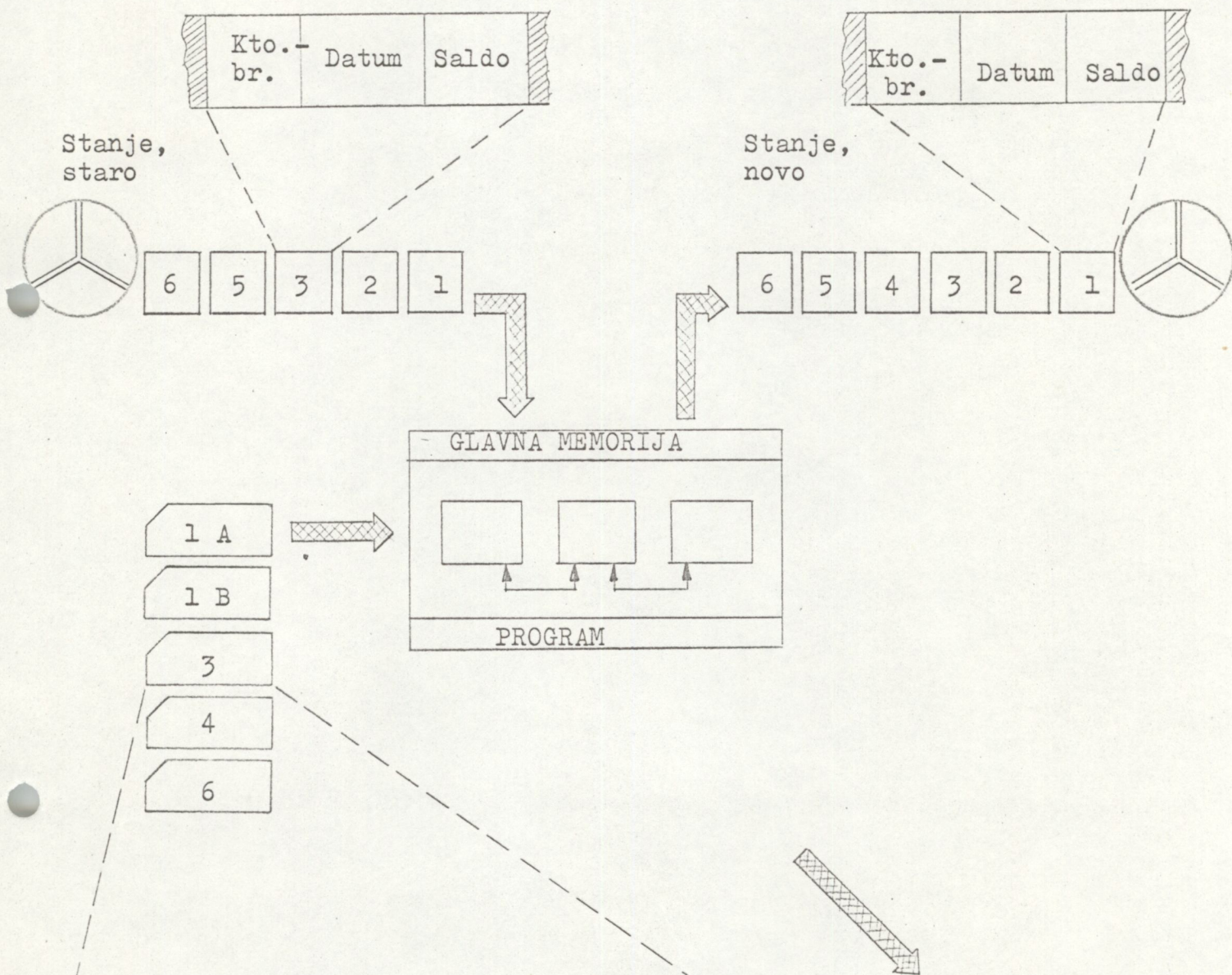
1. Standardni način rada (Burst-Mode)



2. Multipleksni način rada (Byte-Mode)



Primer za obradu na magnetnoj traci



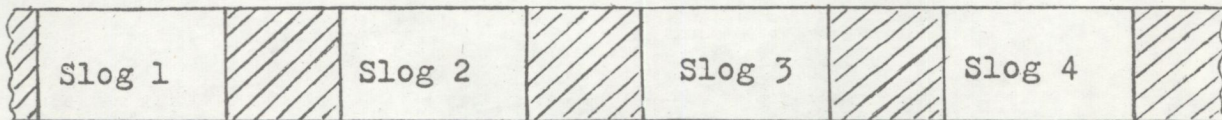
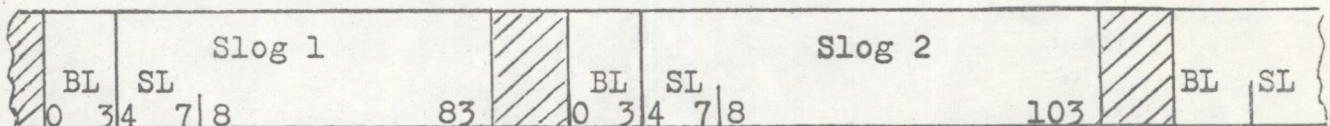
Kto. br.	Datum	Iznos	Br. naloga
000000	000000	00000000	00000000
111111	11111111	11111111	11111111
222222	22222222	22222222	22222222
333333	33333333	33333333	33333333
444444	44444444	44444444	44444444
555555	55555555	55555555	55555555

Dolaz materijala

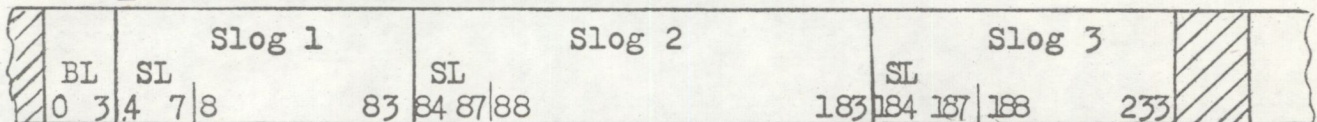
Knjiženje dolaza
(Kto. 4 je novi konto)

Kto. br	Datum	Saldo stari	Dolaz	Saldo novi
1	x-x-x	x-x-x	x-x-x	
2	x-x-x	x-x-x	x-x-x	x-x-x
3	x-x-x	x-x-x	x-x-x	x-x-x
4	x-x-x		x-x-x	x-x-x
5	x-x-x	x-x-x		x-x-x
6	x-x-x	x-x-x	x-x-x	x-x-x

OBLICI SLOGOVA NA MAGNETNOJ TRACI

1. Fiksna dužina sloga, neblokirano2. Fiksna dužina sloga, blokirano3. Promenljiva dužina sloga, neblokirano

BL = 84 SL = 80 BL = 104 SL = 100
 BL = duž. SL = dužina
 bloka sloga;

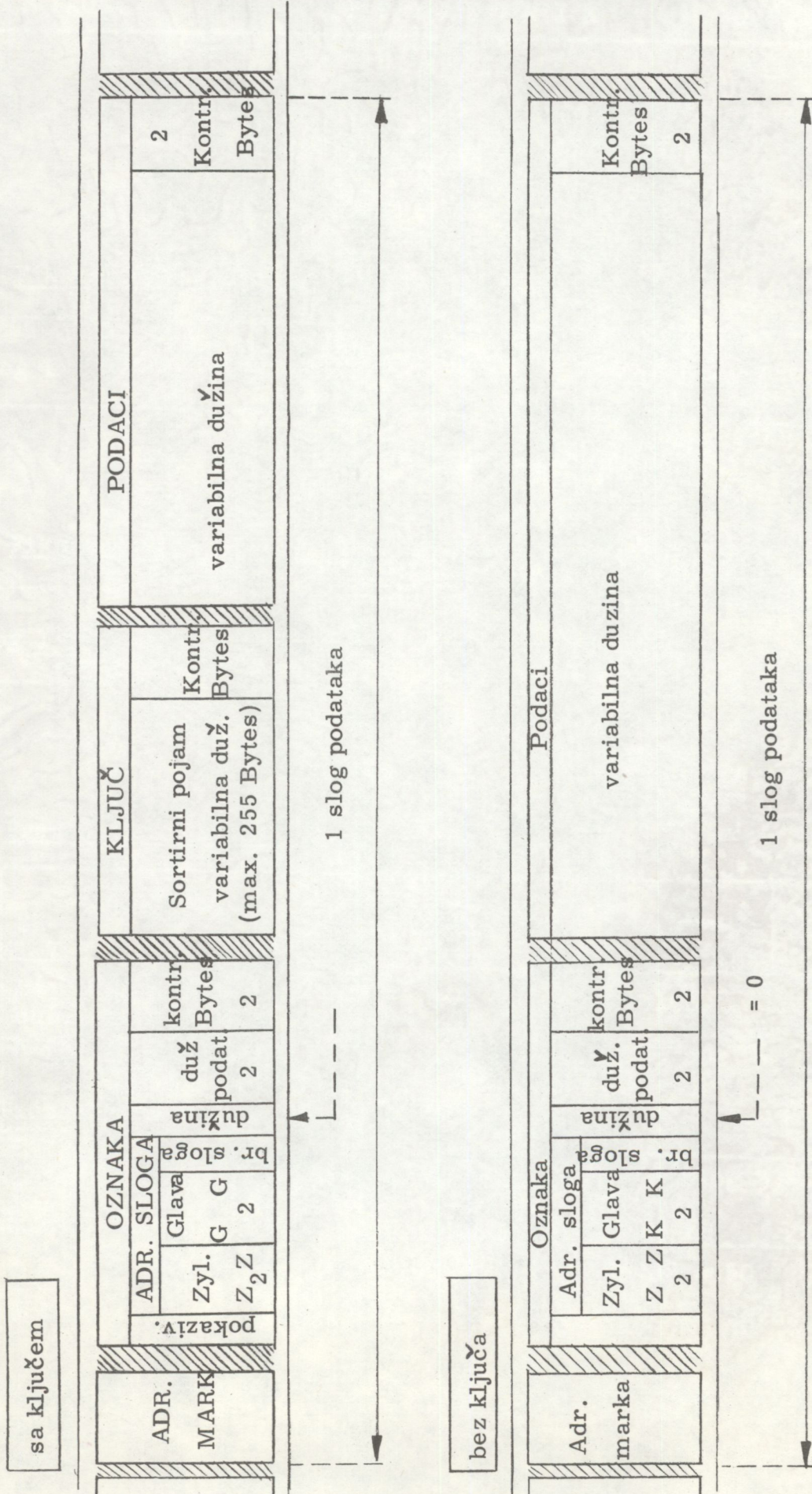
4. Promenljiva dužina sloga, blokirano

BL = 234 SL = 80 SL = 100 SL = 50
 BL = dužina SL = dužina
 bloka sloga

5. Nedefinisano (Promenljiva dužina sloga bez podatka o dužini)

Blokiranje i deblokiranje treba da izvede sam korisnik.

Sastav slogova podataka (važi za sve Random-memorije)



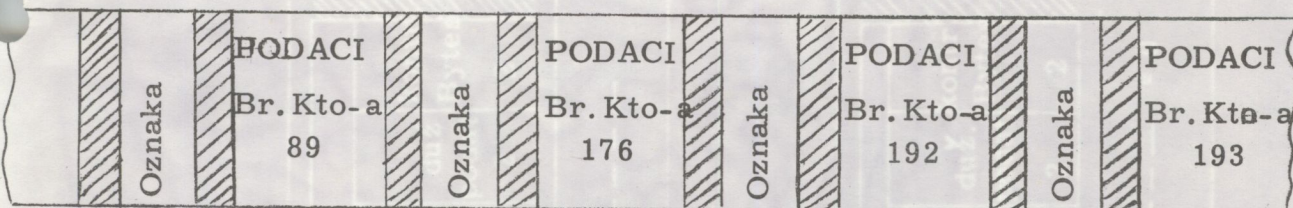
RANDOM MEMORIJE

A. SEKVENCIJALNI OBLIK MEMORISANJA

1. Karakteristika memorisanja

- Svi slogovi su jedan za drugim ^{bez} praznina
- Moguće razdvajanje u više delimičnih področja
- U normalnom slučaju slogovi sortirani prema sortirnom pojmu
- Ne postoji relacija između sortirnog pojma i adrese sloga
- Blokiranje slogova daje bolje iskorišćenje kapaciteta
- Ne zahteva se posebno polje za kod sloga
- Blokiranje i deblokiranje izvodi upravljački program

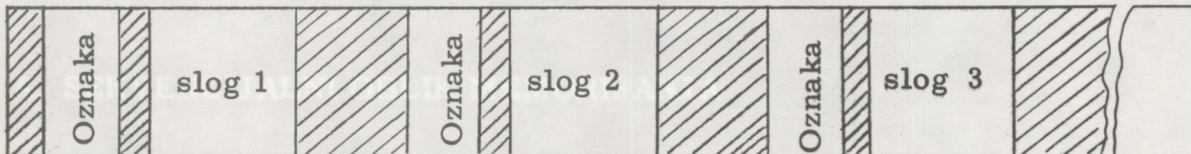
Primer



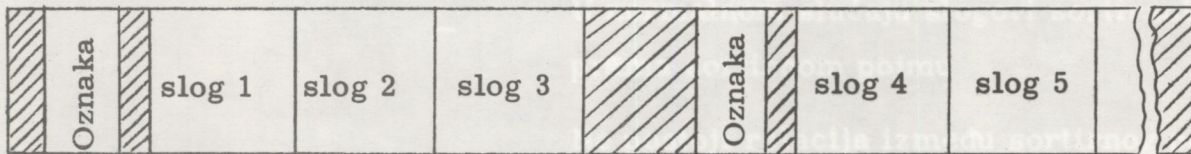
Oznaka sadrži adresu sloga odn. bloka.

2. Mogući oblici slogova kod sekvencijalnog memorisanja

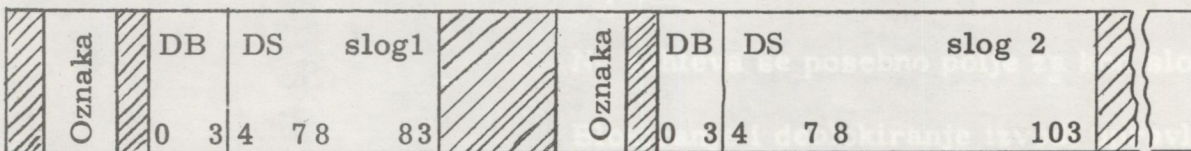
1) Fiksna dužina sloga, neblokirano



2) Fiksna dužina sloga, blokirano



3) Variabilna dužina sloga, neblokirano



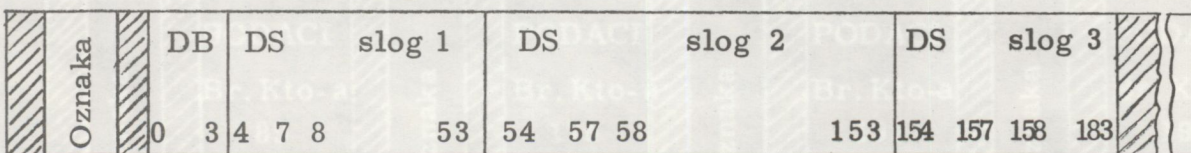
DB = 84 DS = 80

DB = 104 DS = 100

DB = Dužina bloka, DS = Dužina sloga

Primer

4) Variabilna dužina sloga, blokirano



DB = 184 DS = 50

DS = 100

DS = 30

DB = Dužina bloka, DS = Dužina sloga

5) Variabilna dužina sloga bez podatka o dužini

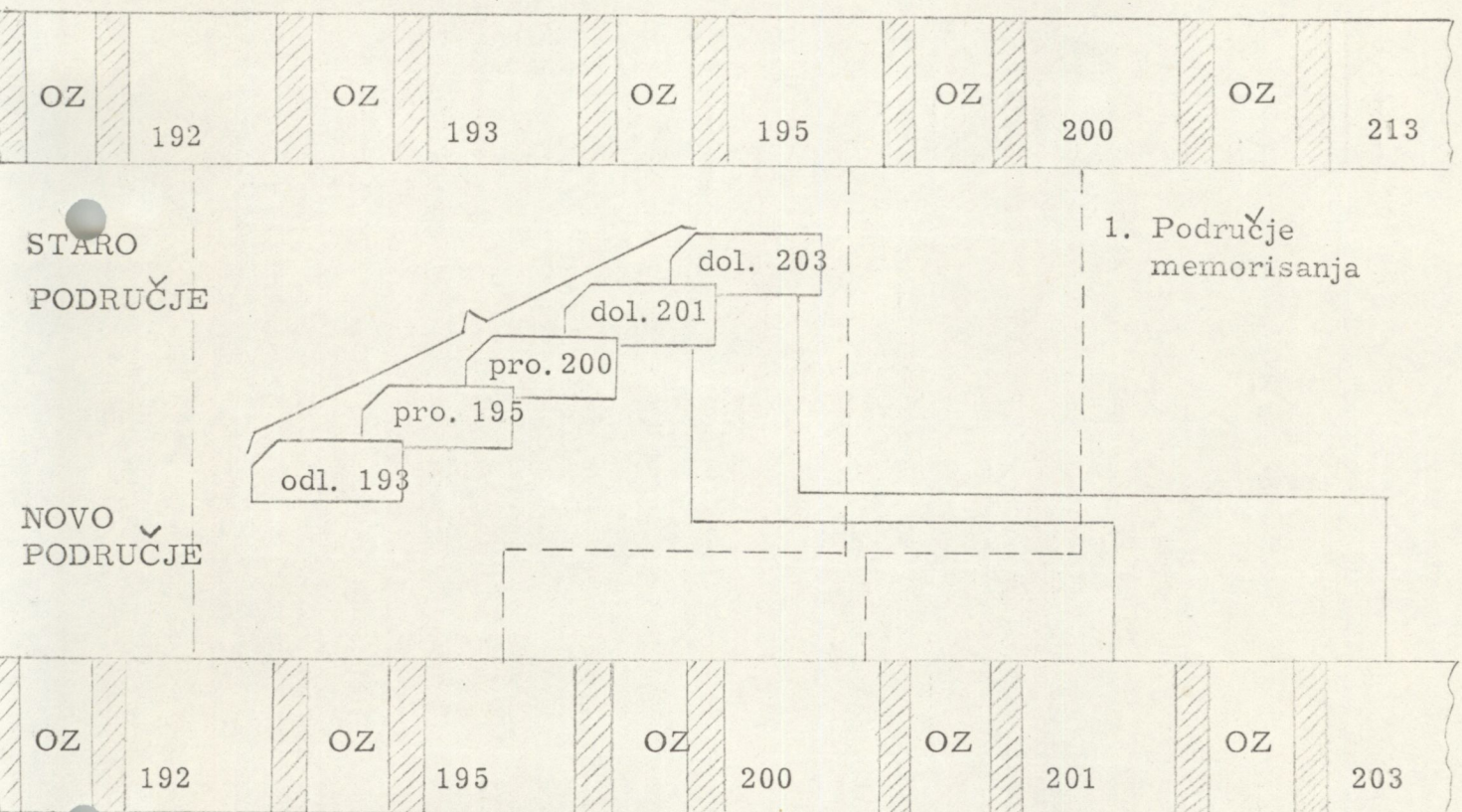
Blokiranje i deblokiranje mora izvršiti korisnik sam.

3. Pristup do memorisanih podataka

Kao kod tehnike sa trakom prelazi se čitava datoteka od početka

4. Obrada: dolaza (nova konta)

odlaza (konta, koja se neće dalje voditi)



2. Područje memorisanja

5. Oznake obrade

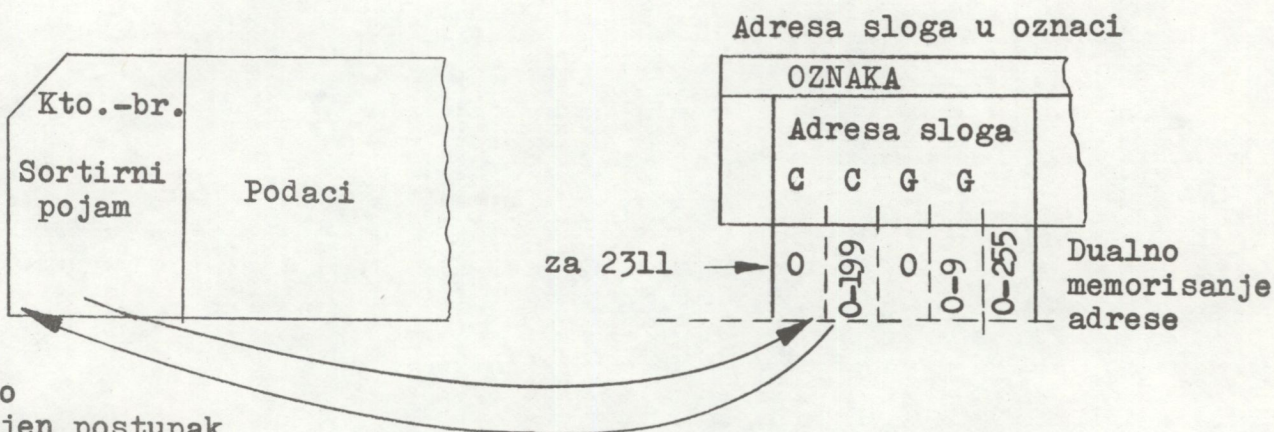
- samo u redosledu memorisanja
- bez razmaka, slog za slogom, (i nepokretne)
- sortiran ulaz eventualnih promena potreban, pošto nema Random pristupa do izvesnog bloka (sloga)
- ponovno pisanje promenjenih blokova (slogova) na staro mesto je moguće (update)
- umetanje ili/i fizičko odstranjivanje slogova traži prememorisanje čitavog stranja u drugo(a) područje(a), (tehnika magnetne trake)

B. RASUTI OBLIK MEMORISANJA

Direktno adresiranje

1. Karakteristike memorisanja

- Sortirni pojam upotrebljava se za određivanje adrese sloga



Fiksno
odredjen postupak
preračunavanja.
Svaki br. konta
daje jednu adresu

- Obratno jednoobrazan odnos izmedju sortirnog pojma i adrese memorisanja
- Sortirni redosled datoteke ostaje kod memorisanja nepromenjen
- Praznine v sortirnom pojmu imaju kao posledicu praznine u zauzimanju memorije
- Svaki sortirni pojam ima svoje tačno definisano i samo njemu predodređeno mesto memorije (nikakvih višekratnih zauzimanja istog mesta u memoriji).
- Posebno polje za ključ nije potrebno
- Dolazi upisuju se u slobodne praznine

Primer:

Potrebno je memorisati slogove podataka 100, 101, 119, 122, 139.

20 slogova moguće je na pr. memorisati u jednoj stazi

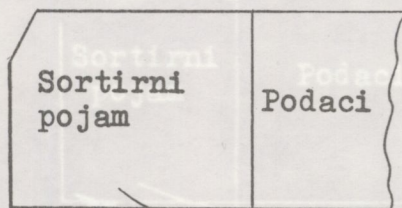
Staza 1	Ad- resa staze	Opis. slog staze	Slog podataka 100	Slog podataka 101	(slobodno)	Slog podataka 119
Staza 2	Ad- resa staze	Opis. sloga staze	(slobodno)	(slobodno)	Slog podataka 122	Slog podataka 139

B. RASUTI OBLIK MEMORISANJA

C. RASUTI OBLIK MEMORISANJA

Direktno adresiranje

Indirektno adresiranje

1. Karakteristike memorisanja

Sortirni pojam upotrebljava se za određivanje adrese sloga

- Sortirni pojmovi preračunavaju se u adrese memorije

Nema mogućnosti računanja unatrag od adrese na sortirni pojam

Postupak preračunavanja, da bi se reducirao broj mesta i izravnale velike praznine.

Pikano odredjen postupak preračunavanja. Svak daje U redosledu sortirnog pojma na mnogim mestima nastaju velike praznine

Obratno jednostrazan odnos izmedju sortirnog pojma i adrese memorisanja

Sortirni redosled datoteke ostaje kod memorisanja nepromenjen

- Preračunavanje više sortirnih pojmova može da dovede na istu adresu (više-kratno zauzimanje adresa memorije). Na drugoj strani više adresa se ne izračuna ("praznine" u području memorije). Samo njesu predodređeno mesto memorije (nikakvih višekratnih

- Kroz preračunavanje sortirnih pojmova nema sortirano memorisanja.

- Memorisanje sa ili bez ključa (ovisi o načinu adresiranja).

- Dolazi (Zugänge) upisuju se u slobodne praznine ili u vlastito prenosno područje.

Primer: Potrebno je memorisati
119, 122, 139.

2. Mogući oblici slogova

- 1) Fiksni neblokirani (Operating system BOS, DOS, OS);
- 2) Variabilna dužina slogova neblokirani (Operating system OS);
- 3) Variabilna dužina slogova bez oznake dužine (BOS, DOS, OS).