



IZVRŠAVANJE INSTRUKCIJA



■ **Ponovimo!**

1. Što je instrukcija? Navedi primjer!
2. Preko koje se sabirnice instrukcije prenose u mikroprocesor?
3. Što sadrži instrukcijski registar?
4. Koji dio procesora izvršava instrukciju?
Kako?

Oblik instrukcijske riječi

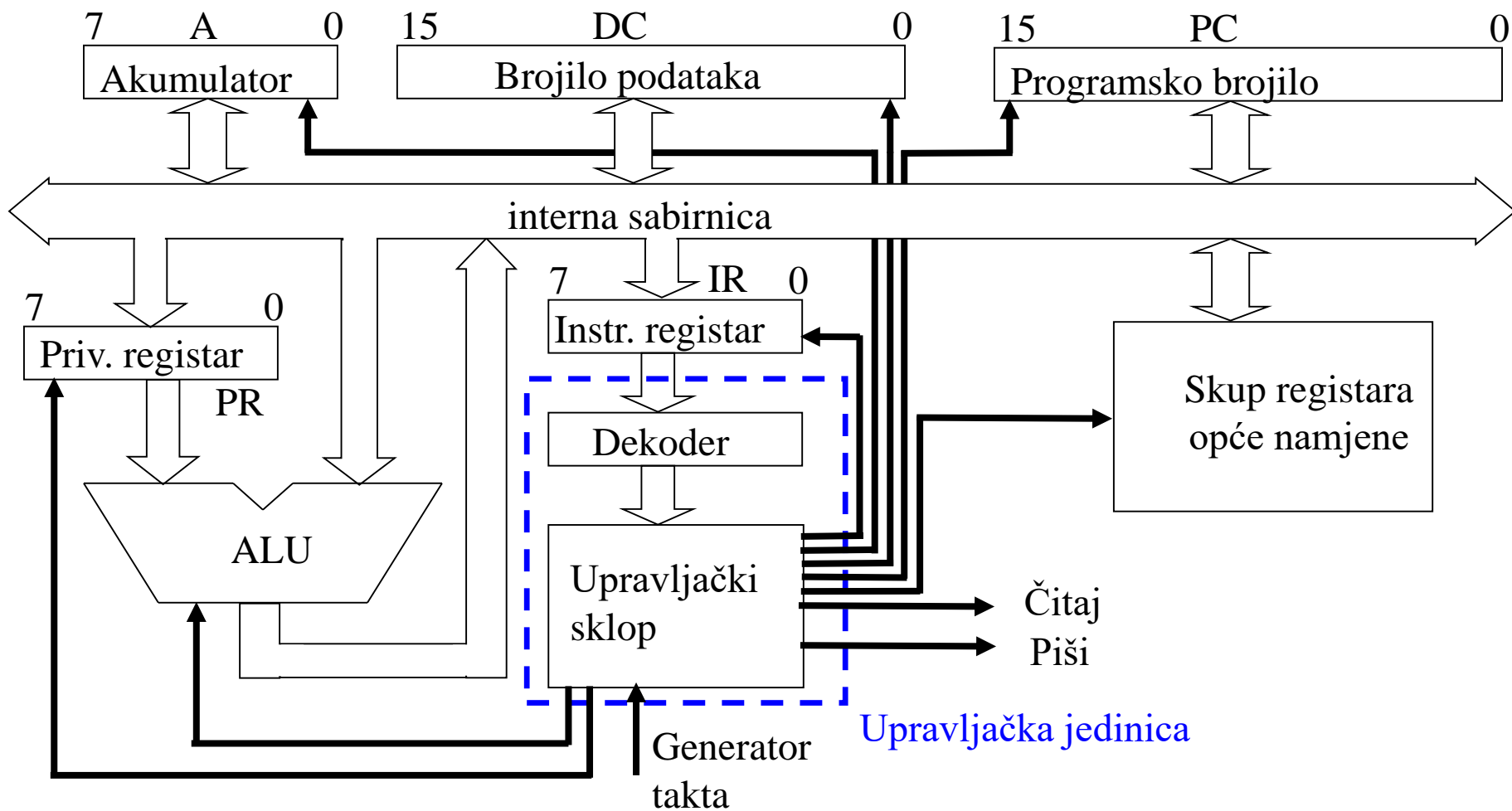


- **operacijski kod** – određuje vrstu operacije
- **način adresiranja** – način na koji se pribavljaju operandi / pohranjuju rezultati operacije
- **adresno polje** - određuje adresu operanda i rezultata

Izvršavanje instrukcija

- instrukcije se izvršavaju u dvije faze:
 - **PRIBAVI** (fetch)
 - **IZVRŠI** (execute)
- za vrijeme faze **PRIBAVI** procesor pribavlja instrukciju (operacijski kod i adresu operanda ili operand)
- za vrijeme faze **IZVRŠI** procesor dekodira kod i izvršava operaciju koja je zadana operacijskim kodom

Pojednostavljeni model procesora



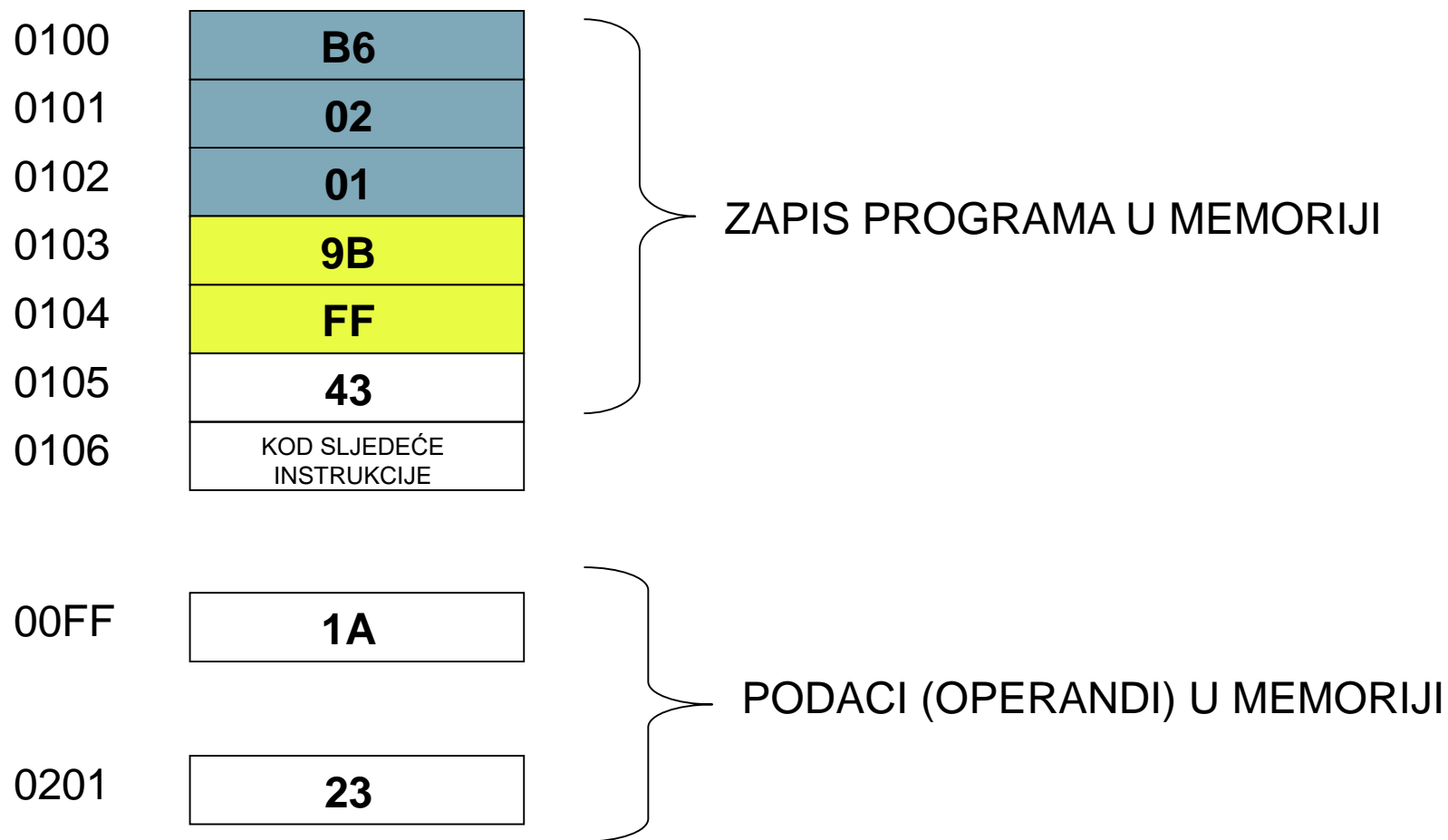
Primjer izvođenja instrukcije

- zadatak: potrebno je zbrojiti dva operanda i rezultat komplementirati; prvi operand (23h) se nalazi na adresi 0201, a drugi operand (1Ah) se nalazi na adresi 00FFh
- postupak:
 1. potrebno je najprije operand 23h pohraniti u akumulator A,
 2. akumulatoru dodati operand 1Ah, a rezultat se pohranjuje ponovo u akumulator
 3. sadržaj akumulatora se komplementira

Oblik programa u mnemoničkom kodu

- LDA A \$ 0201; u akumulator A pohrani operand s adrese 0201
- ADD A \$ FF; sadržaju akumulatora dodaj operand s adrese 00FF
- COM A; komplementiraj sadržaj akumulatora A

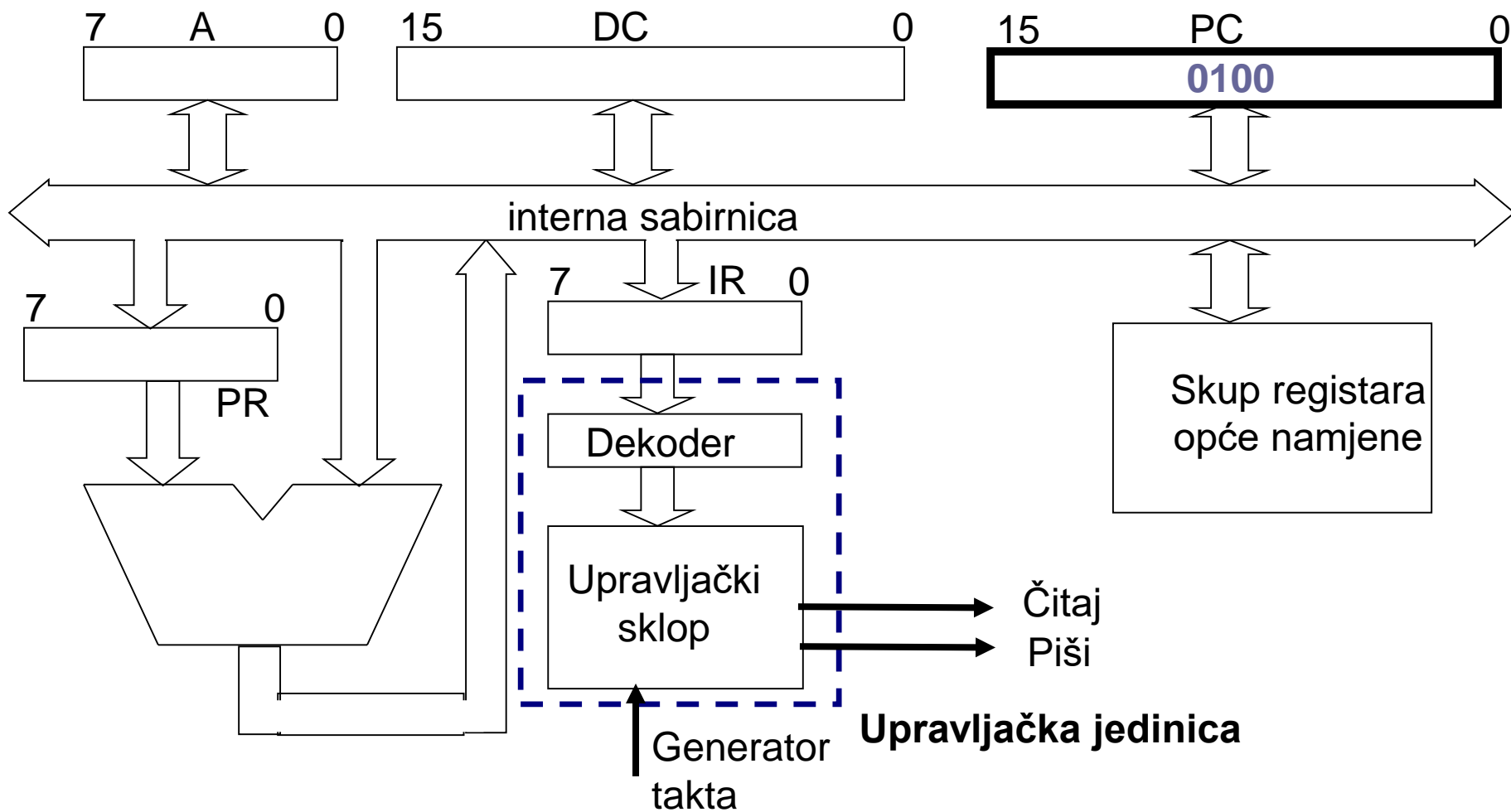
Zapis programa i podataka u memoriji



Početni uvjeti

- sve adrese i podaci – heksadecimalni zapis
- program je pohranjen u memoriji mikroračunala na adresama: 0100, 0101, 0102, ...
- na početku programsko brojilo (PC) ima vrijednost 0100 i pokazuje na operacijski kod 1. instrukcije

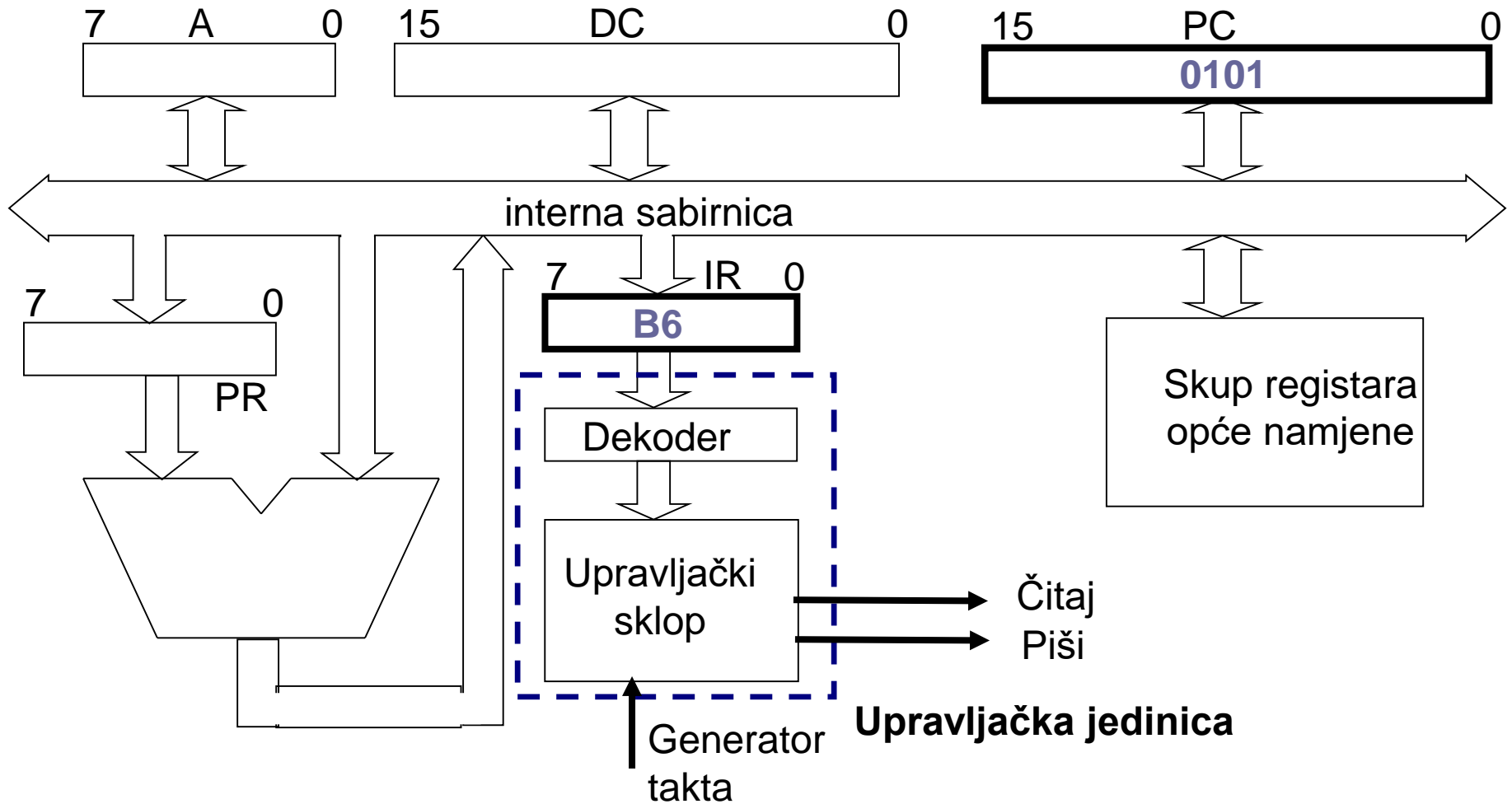
Početni uvjeti




1. instrukcija - faza PRIBAVI

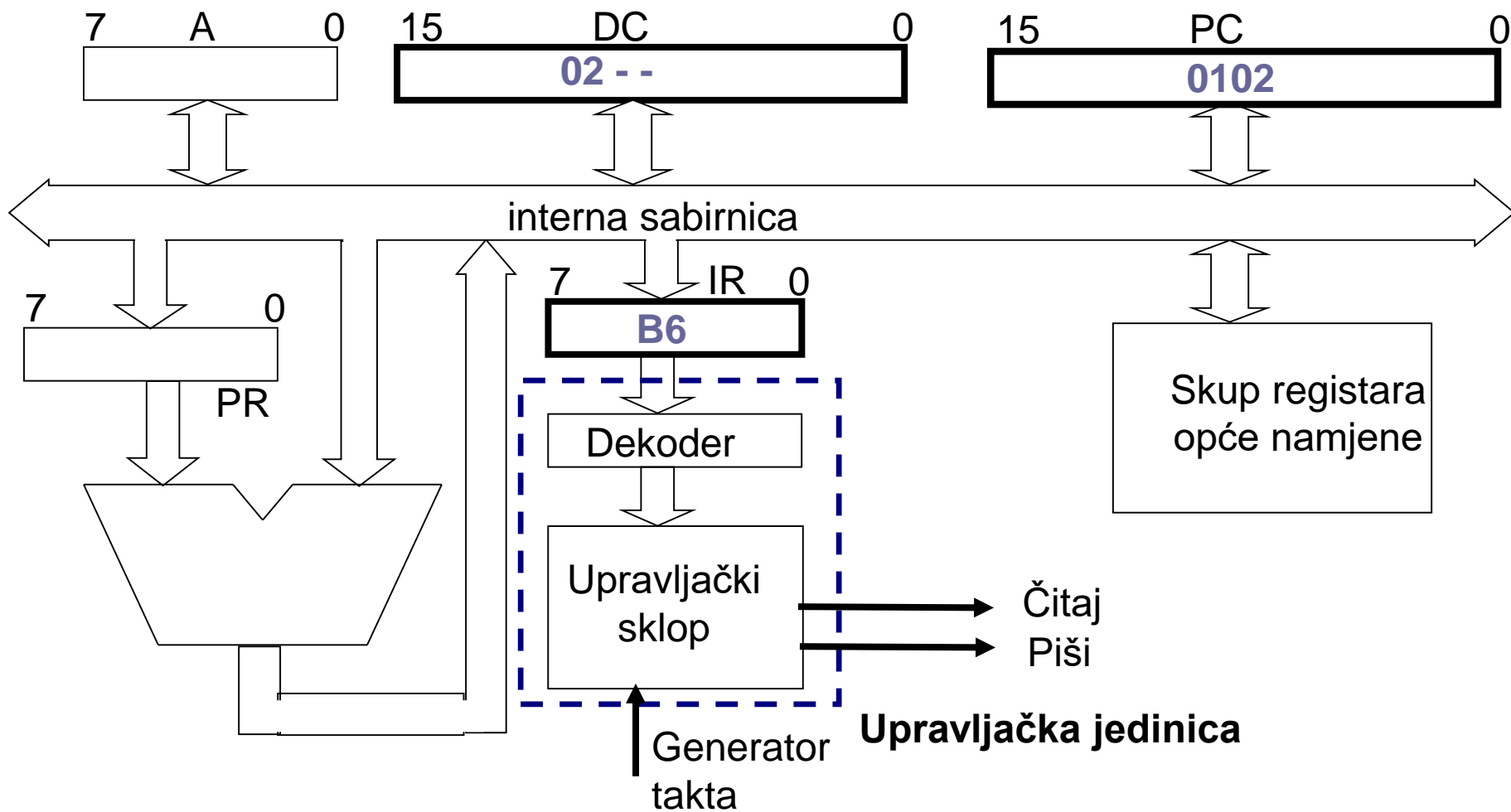
- sa lokacije 0100 se čita operacijski kod (B6) i pohranjuje u IR, a sadržaj PC se povećava za 1 i iznosi 0101

1. instrukcija - faza PRIBAVI

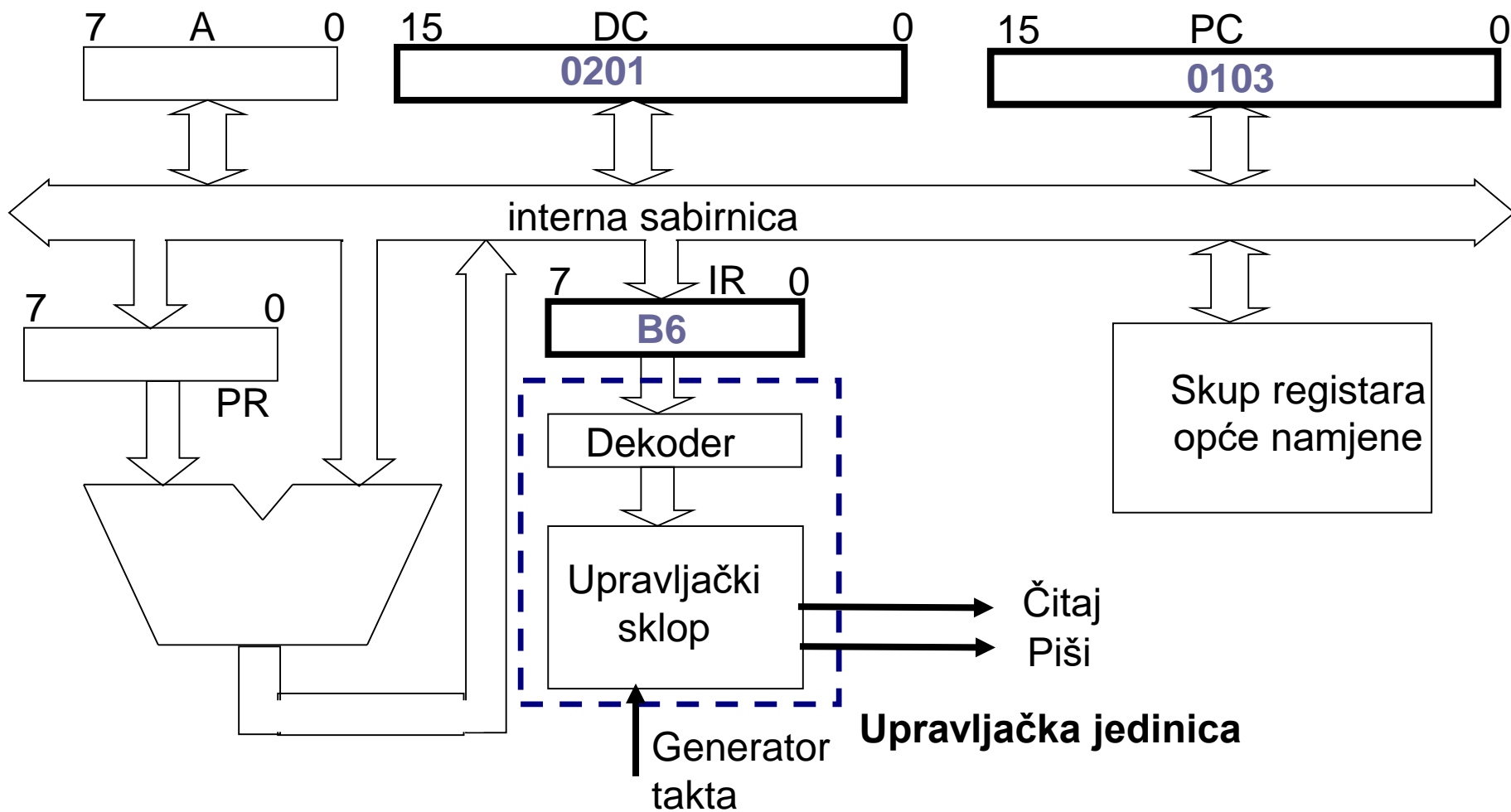


- 
- mikroprocesor nastavlja sa fazom PRIBAVI
 - pribavlja se novi byte koji se smješta u DC, a PC se povećava za 1
 - na isti način se pribavlja još jedan byte te tako imamo potpunu adresu operanda

1. instrukcija - faza PRIBAVI



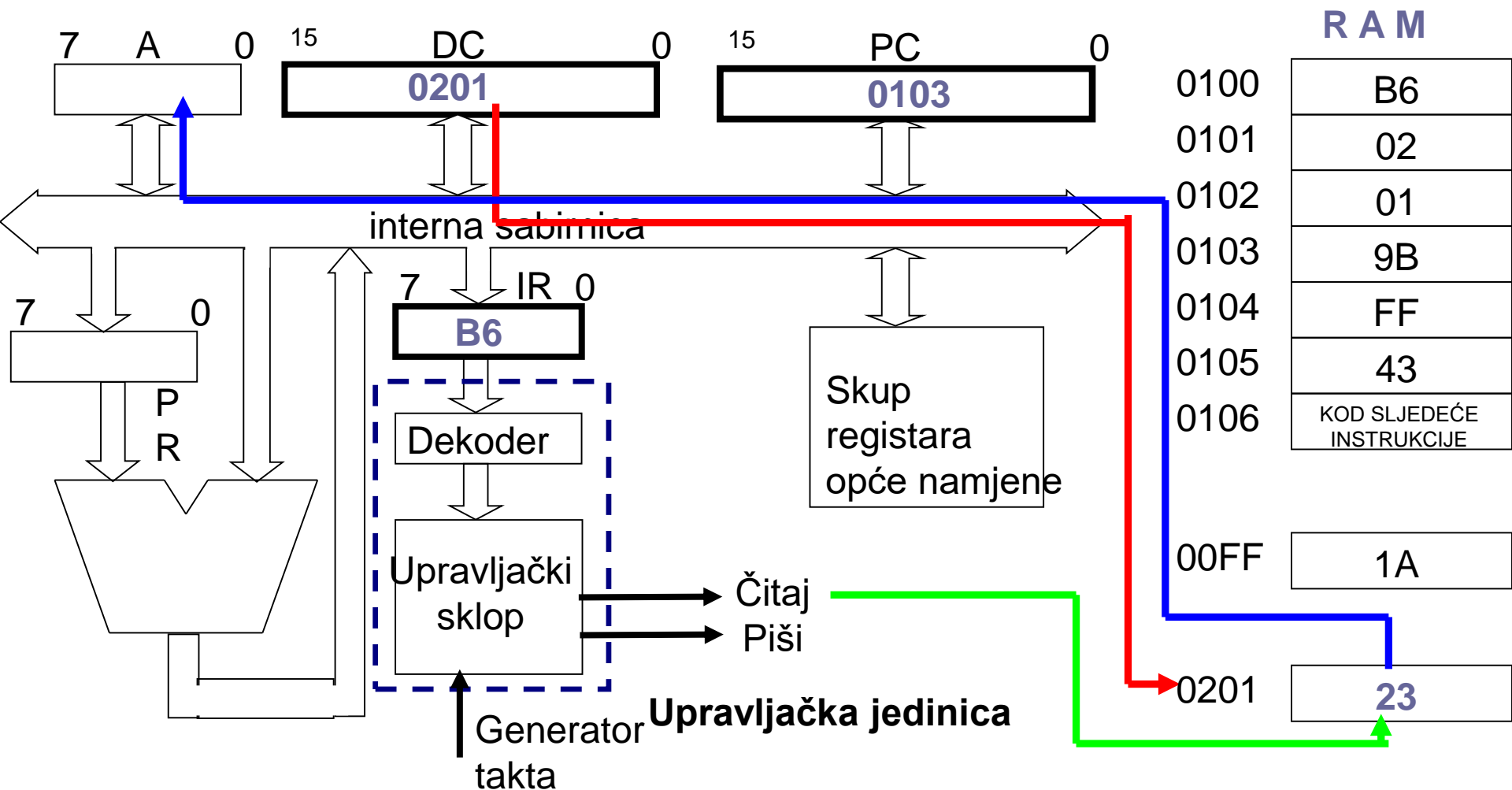
1. instrukcija - faza PRIBAVI



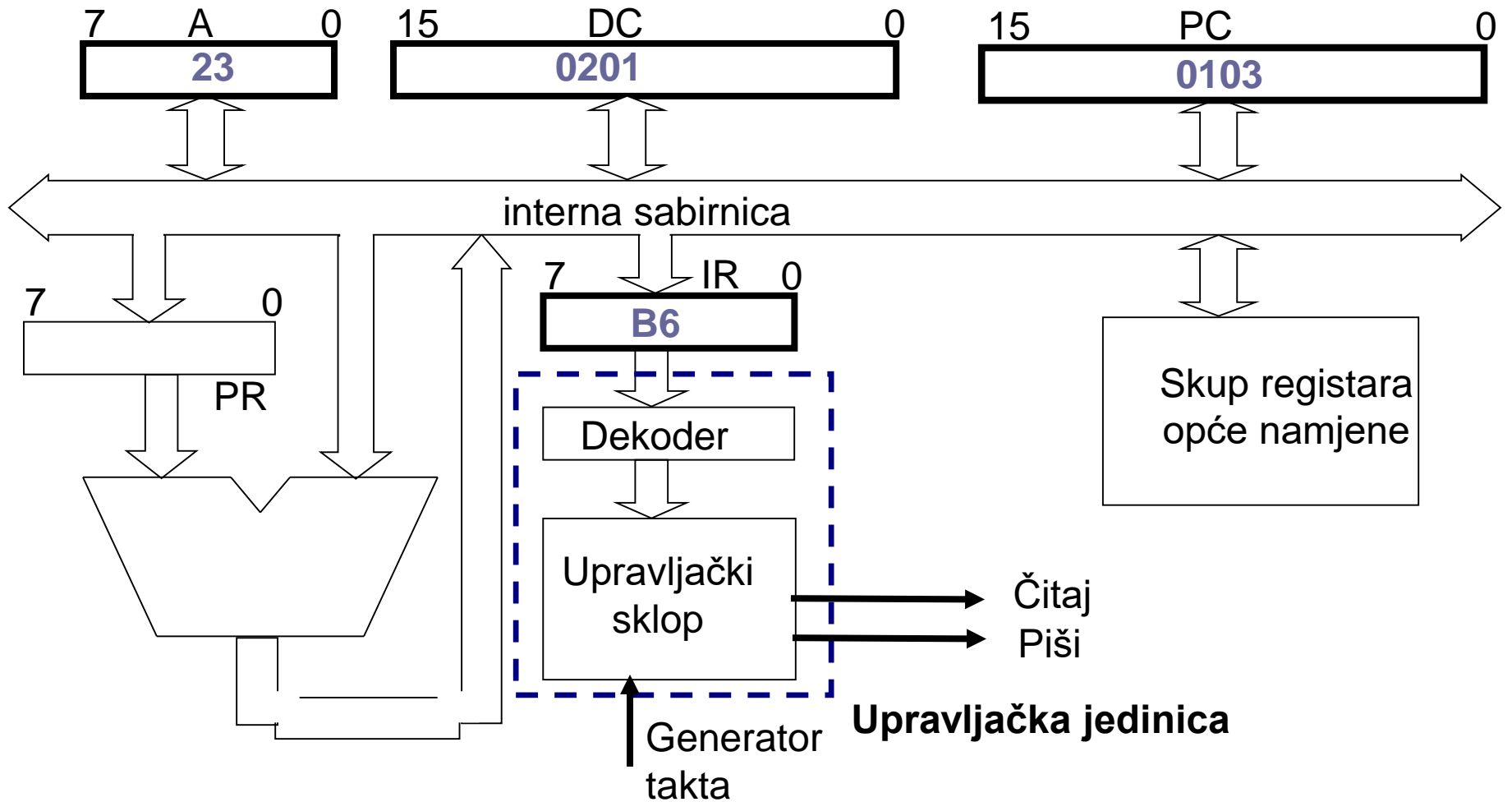
1. instrukcija - faza IZVRŠI

- nakon završetka faze PRIBAVI, mikroprocesor prelazi u fazu IZVRŠI
- sadržaj IR se dekodira i tumači se kao:
napuni akumulator A sadržajem memorijske lokacije koja je zapisana u brojilu podataka (DC, DPTR)
- faza IZVRŠI sastoji se od pribavljanja operanda s memorijske lokacije 0201 u akumulator A
- sadržaj brojila podataka DC postavlja se na adresnu sabirnicu i generira se upravljački signal ČITAJ

1. instrukcija - faza IZVRŠI



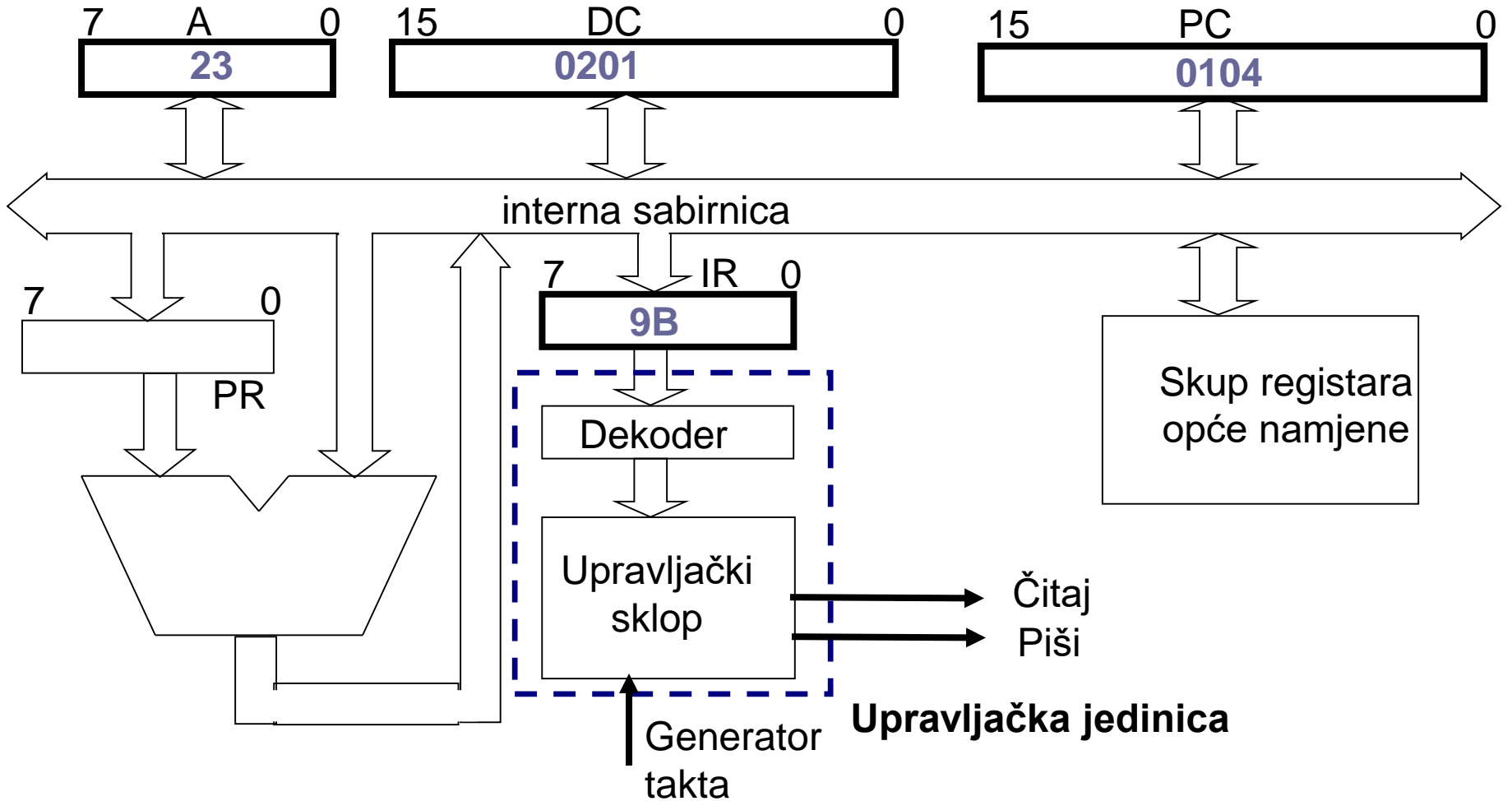
Stanje registara nakon 1. instrukcije



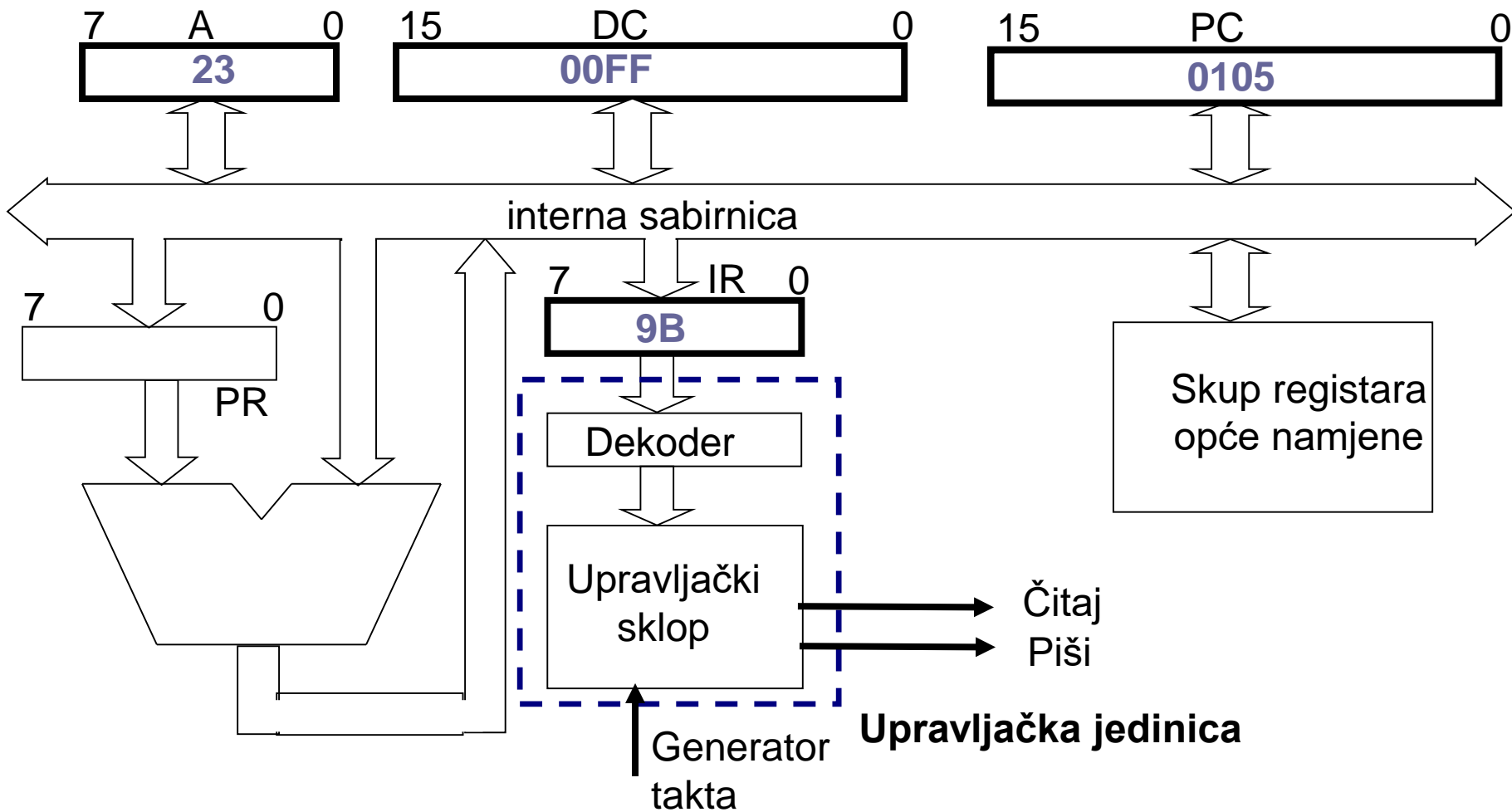
2. instrukcija - faza PRIBAVI

- nakon izvršenja prve instrukcije procesor kreće u fazu PRIBAVI druge instrukcije
- PC ima sadržaj sljedeće instrukcije, a ostali registri imaju sadržaje pribavljene tijekom prve instrukcije

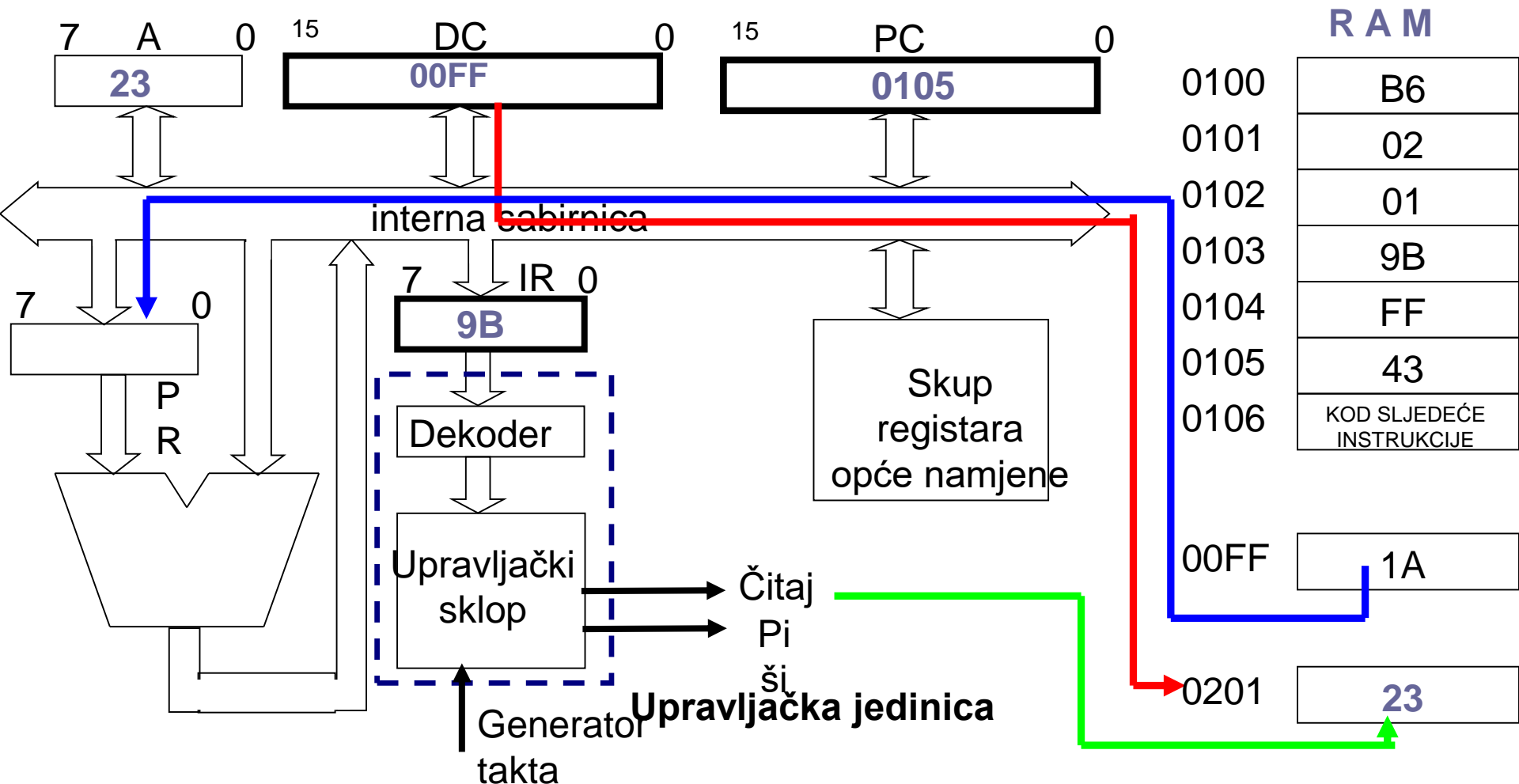
2. instrukcija - faza PRIBAVI



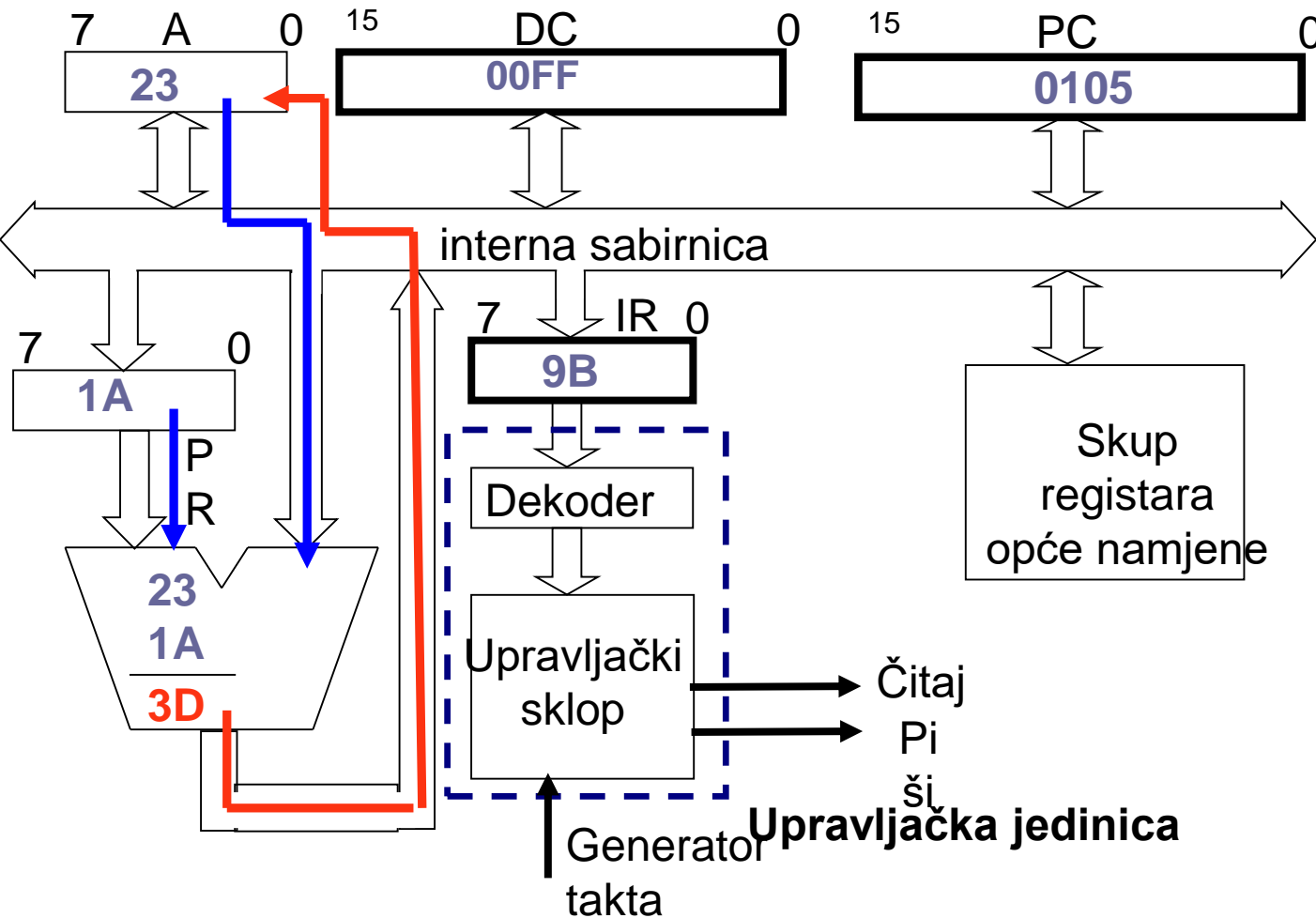
2. instrukcija - faza PRIBAVI



2. instrukcija - faza IZVRŠI



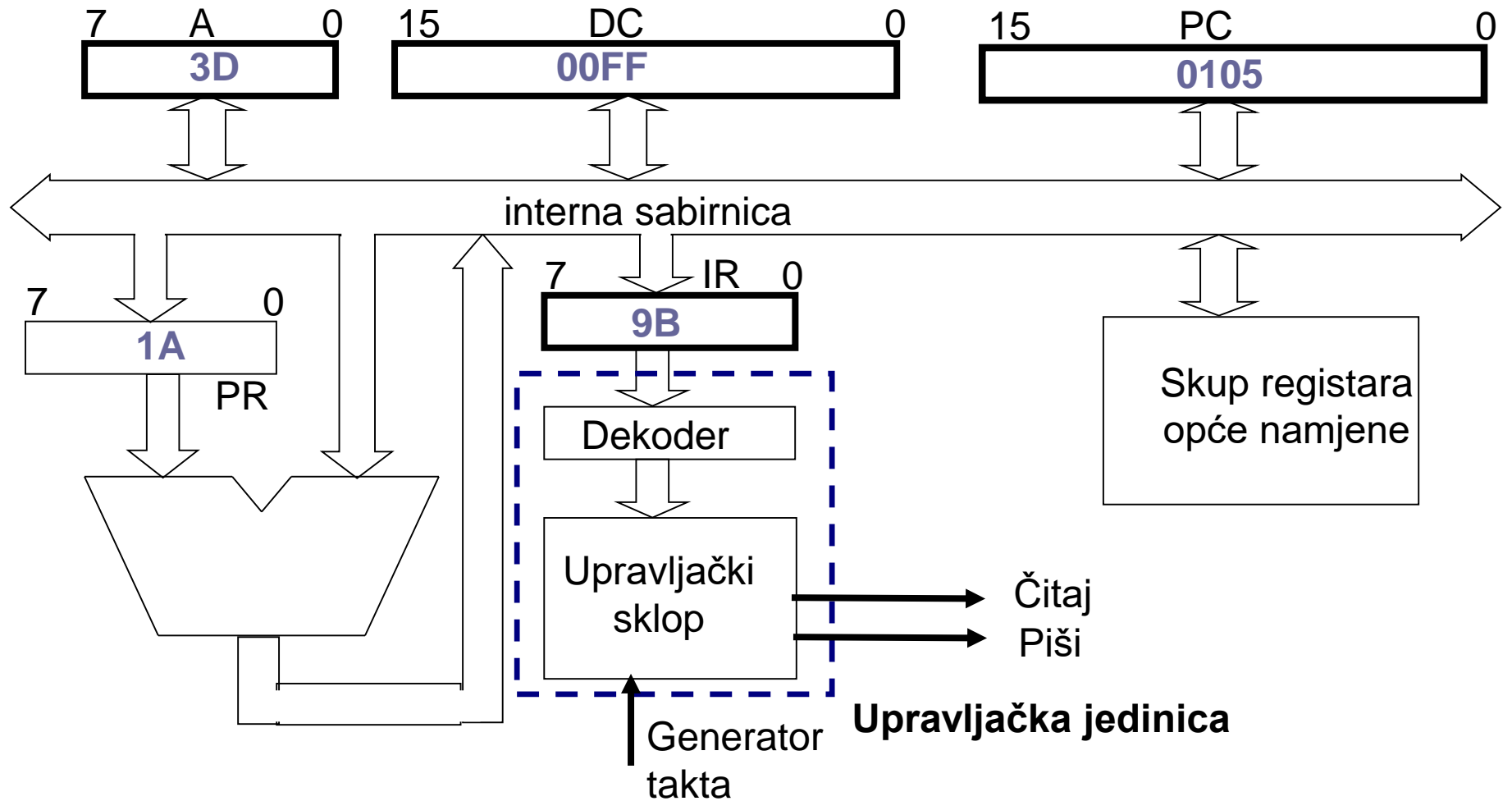
2. instrukcija - faza IZVRŠI



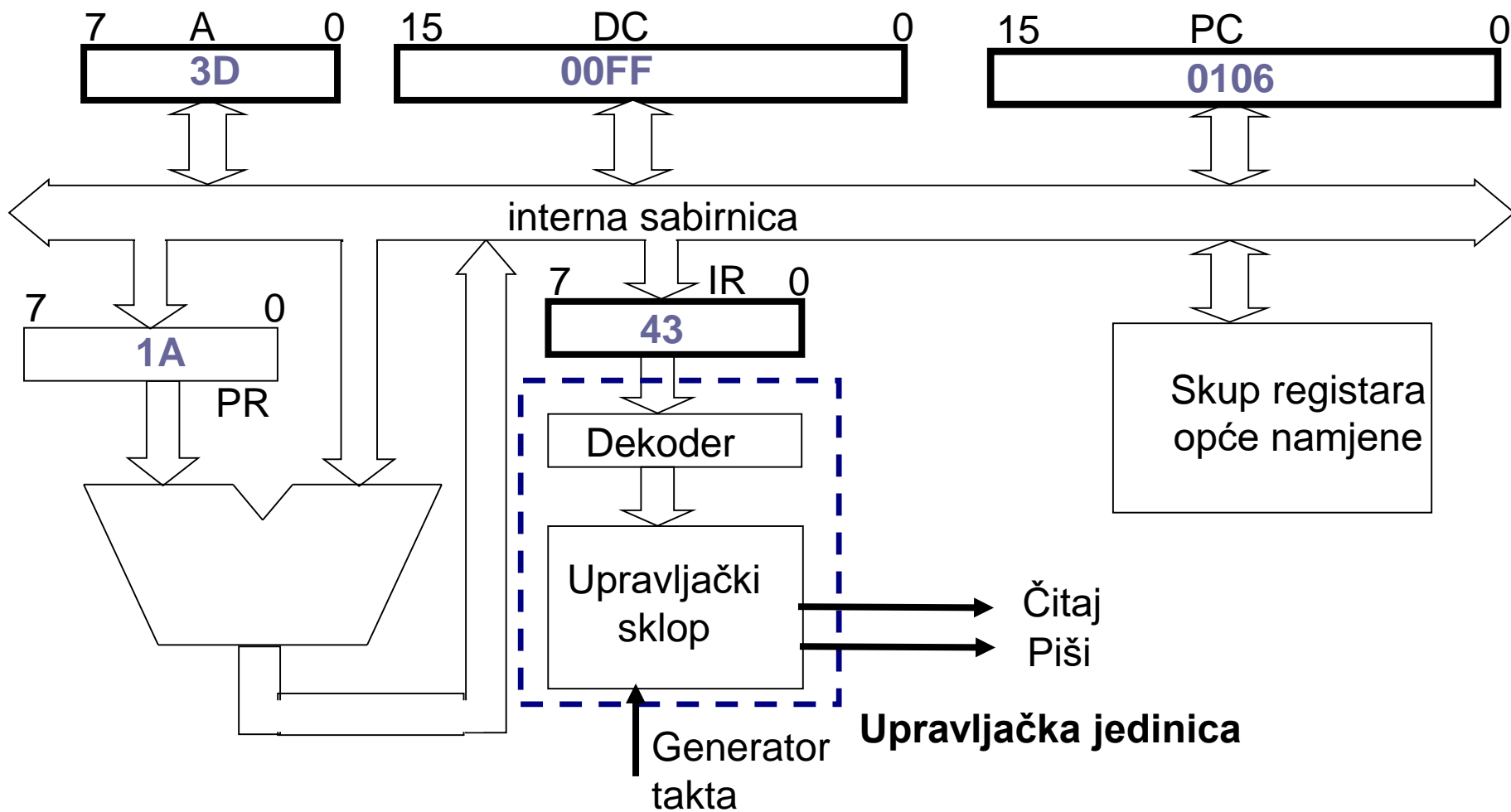
RAM

0100	B6
0101	02
0102	01
0103	9B
0104	FF
0105	43
0106	KOD SLJEDEĆE INSTRUKCIJE
00FF	1A
0201	23

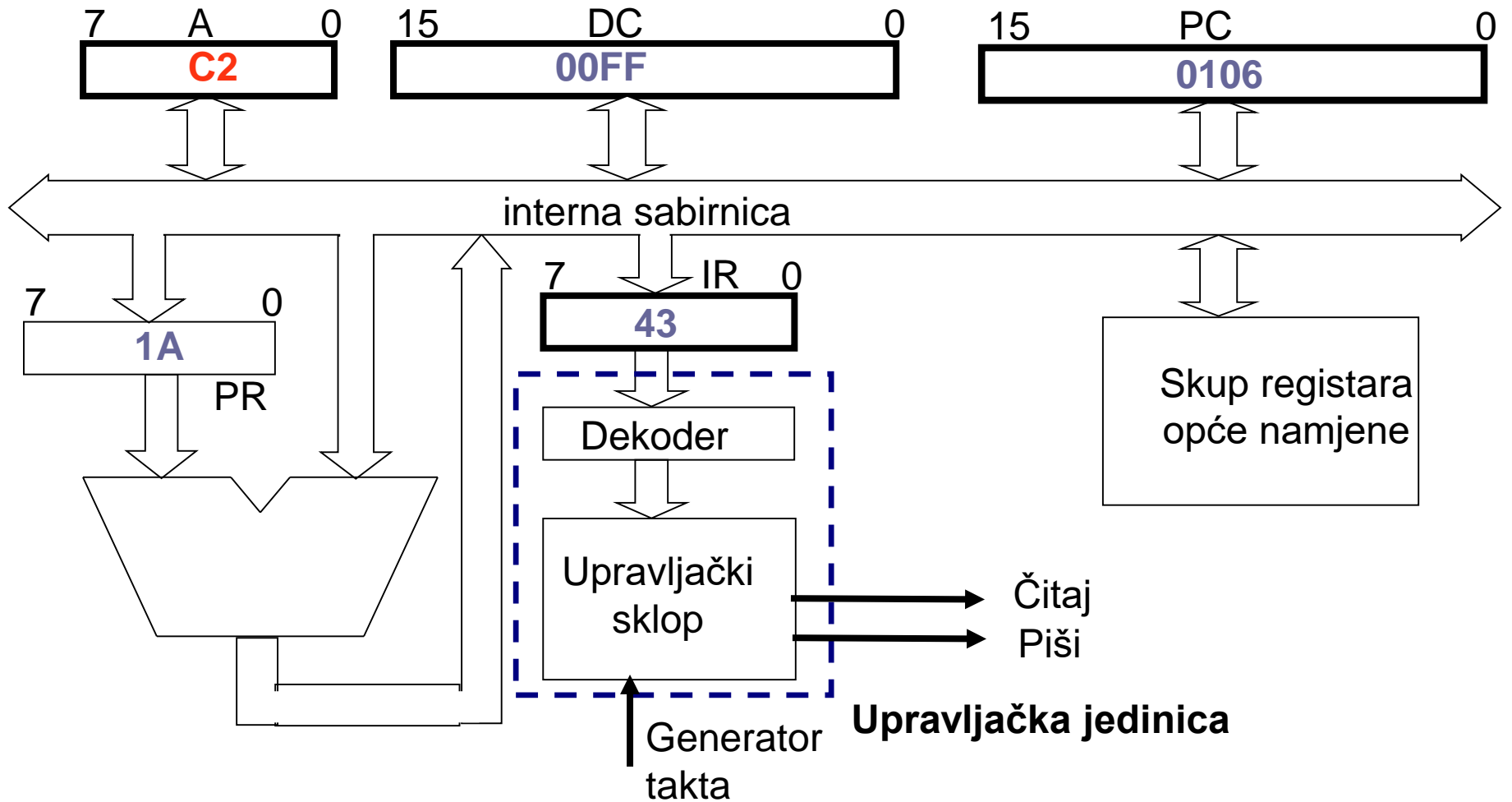
Stanje registara nakon 2. instrukcije



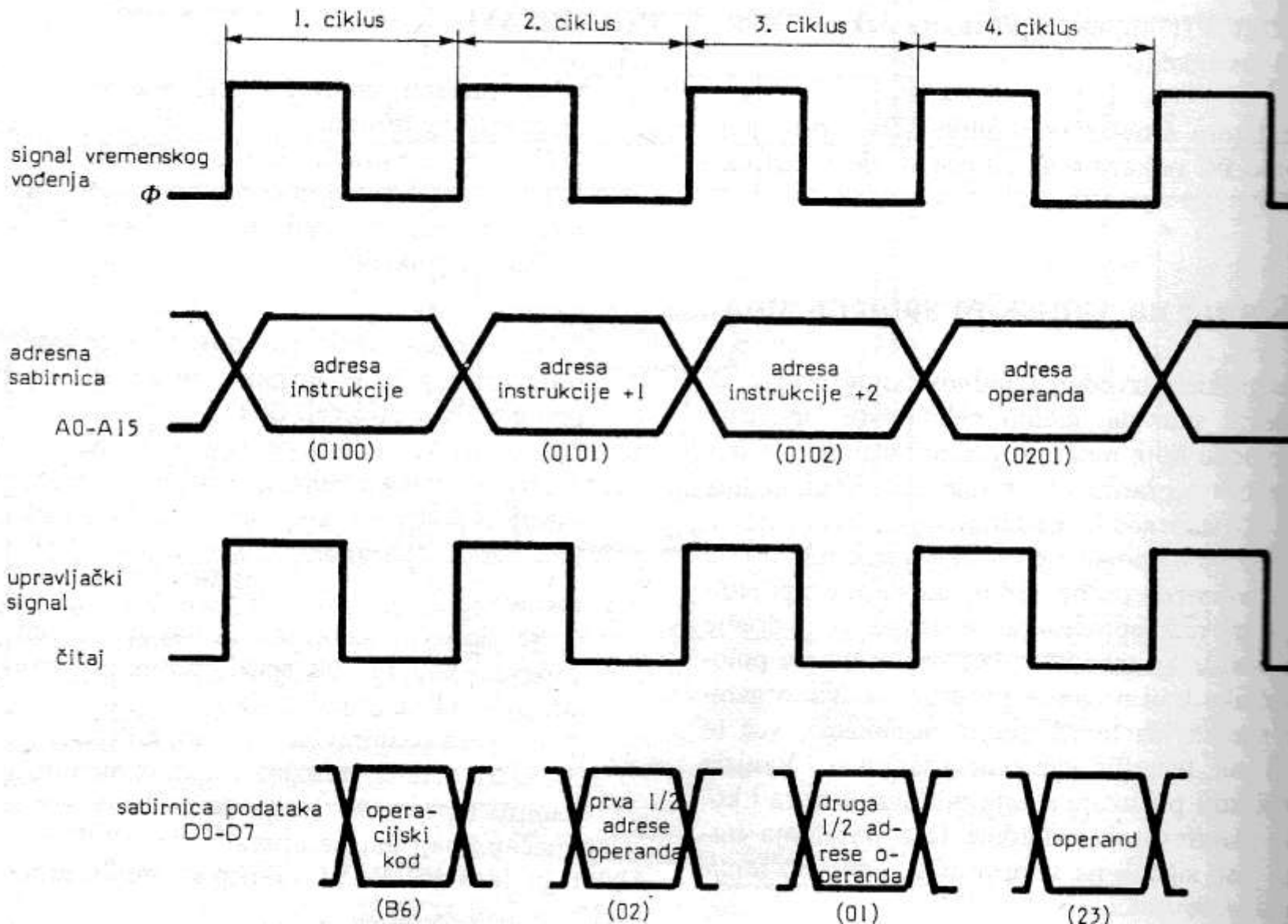
3. instrukcija - faza PRIBAVI



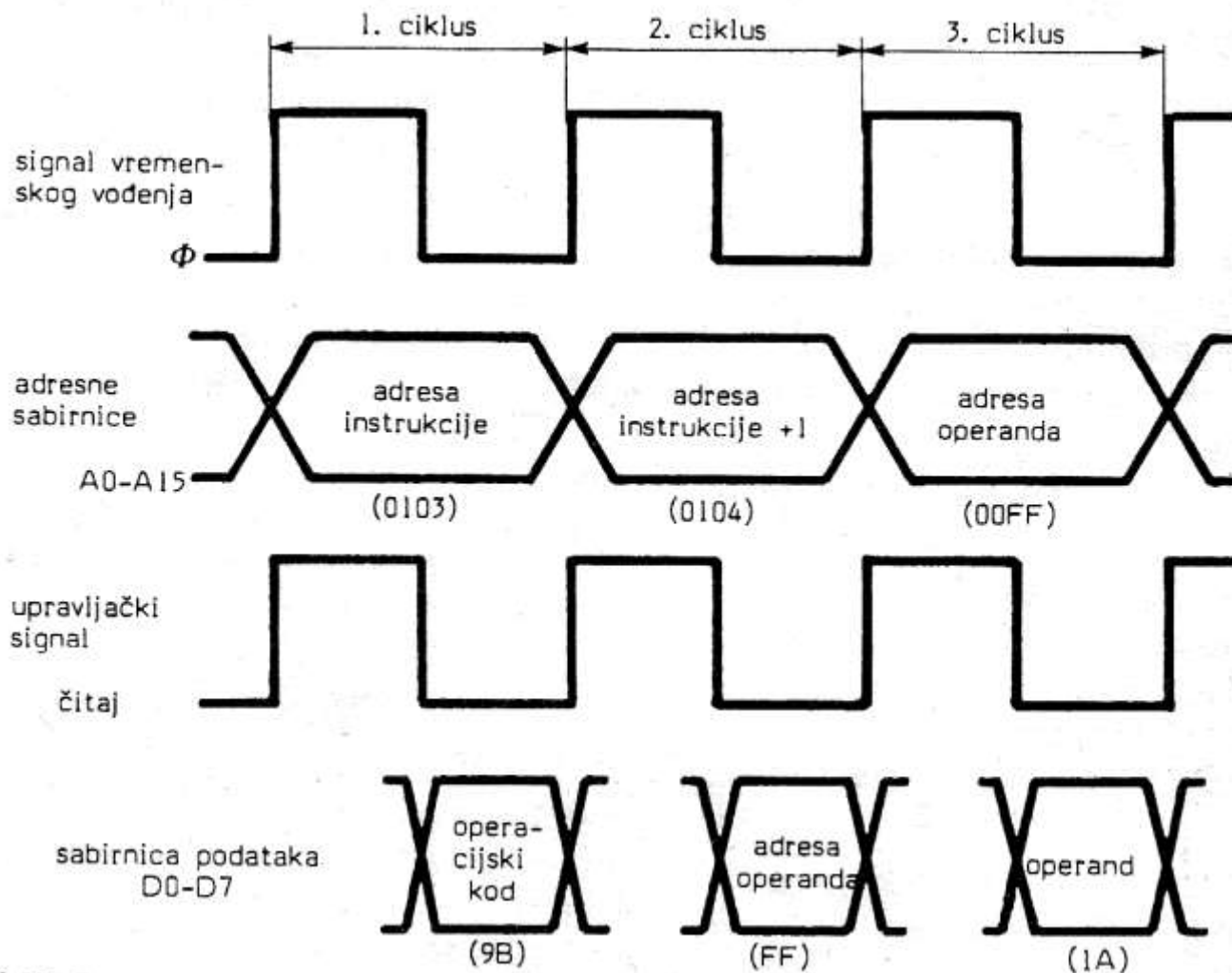
3. instrukcija - faza IZVRŠI



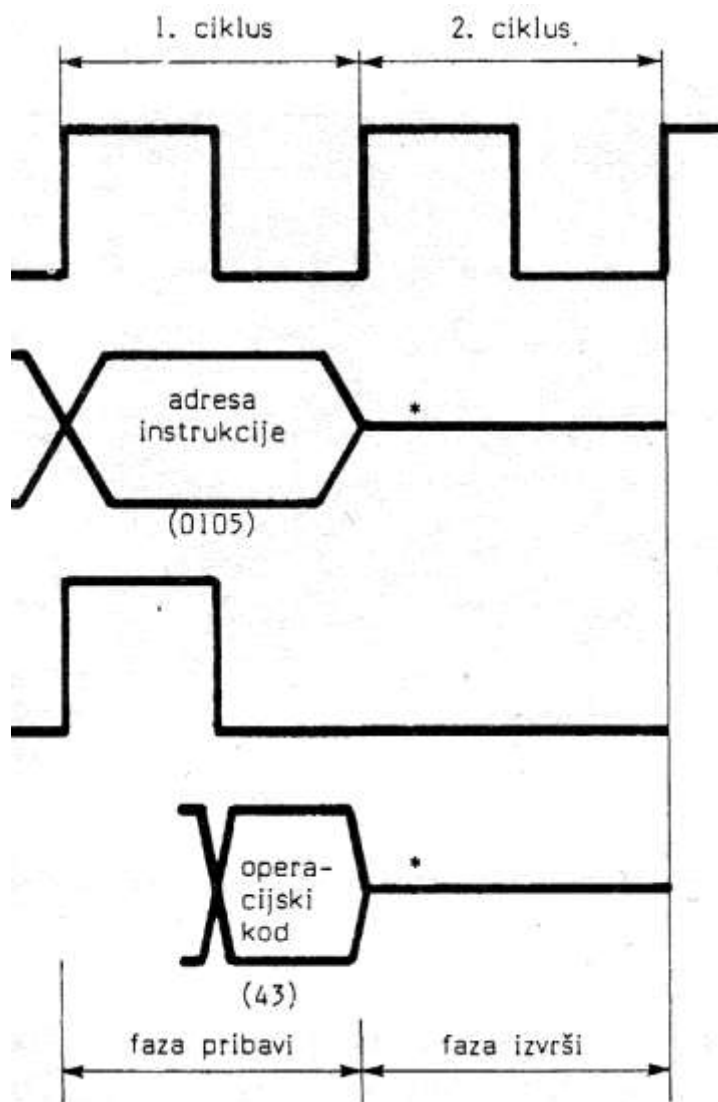
Stanje na sabirnicama – 1. instrukcija



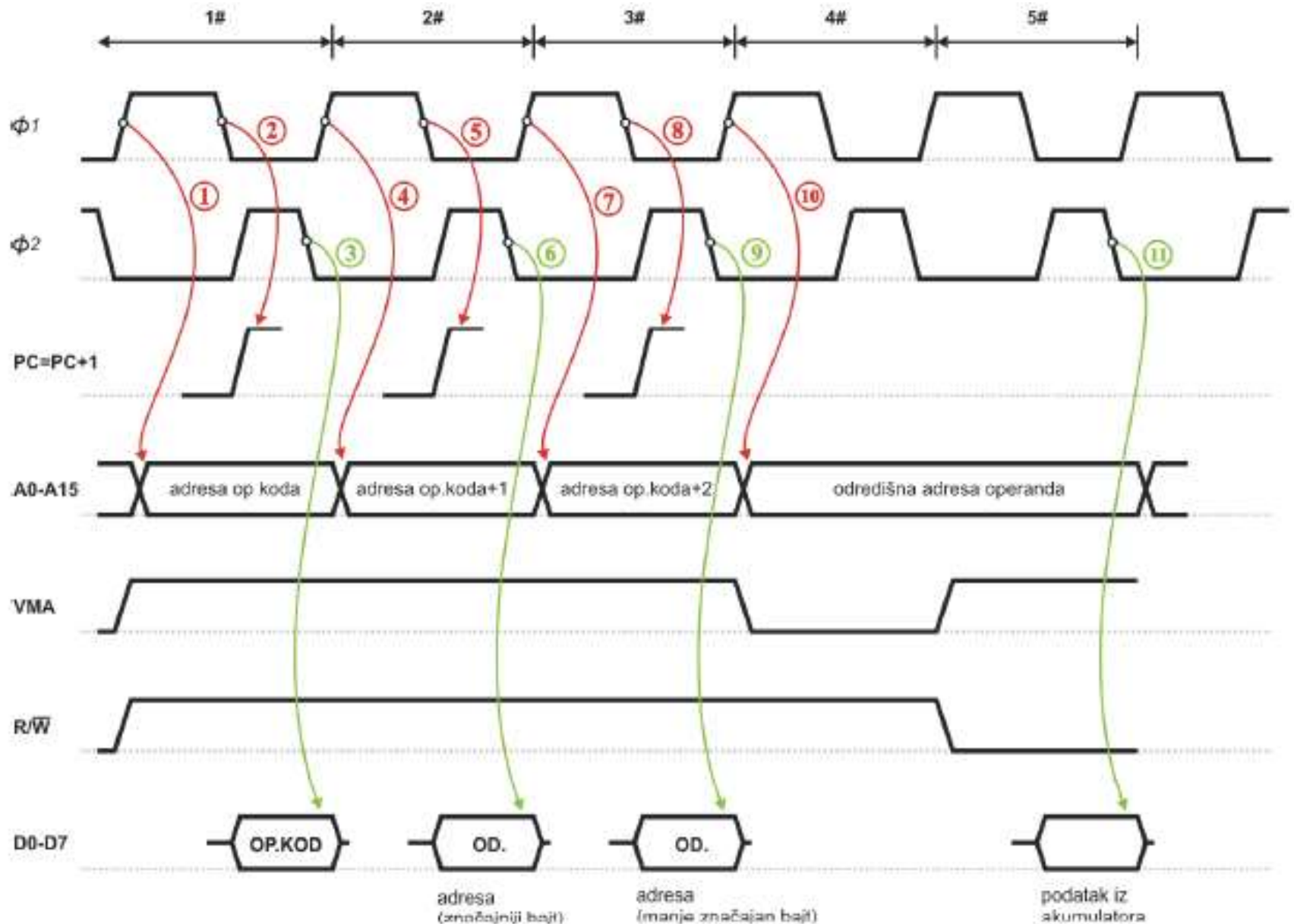
Stanje na sabirnicama – 2. instrukcija



Stanje na sabirnicama – 3. instrukcija



Stanje na sabirnicama –instrukcija STA



Ponovimo!

1. Navedi strukturu instrukcije (assembler)!
2. Objasni faze izvođenja instrukcija!
3. Objasni prikaz stanja na sabirnicama!

Domaća zadaća!

1. Koristeći se knjigom, prikaži tablicom promjene stanja registara procesora te vremenski dijagram tijekom izvođenja instrukcije:

LDA A \$ 073A