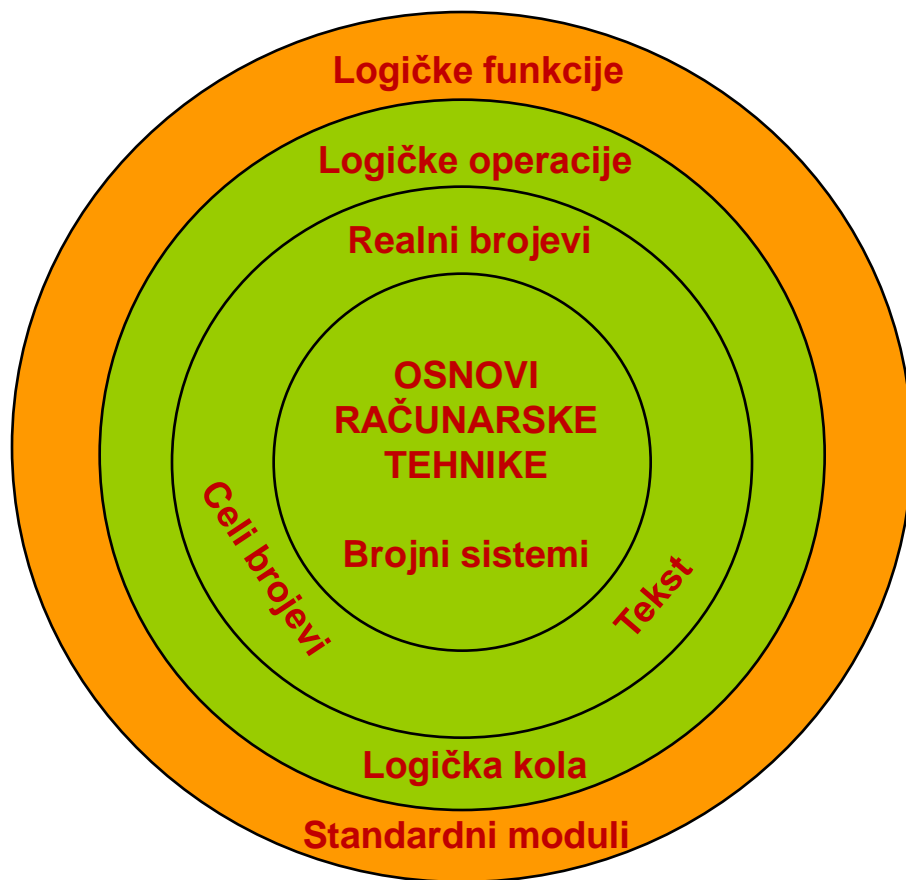


Standardni moduli



TEME

- ✓ Kombinatorni moduli
 - ✓ Koderi i dekoderi
 - ✓ Multiplekseri i demultiplekseri
 - ✓ Sabirači
 - ✓ Aritmetičko-logičke jedinice
- ✓ Sekvencijalni moduli
 - ✓ Registri
 - ✓ Brojači
 - ✓ Memorije

Uvod

- ❑ Prekidačkim mrežama se mogu realizovati razni funkcionalni zahtevi.
- ❑ **Standardni moduli** su često korišćene prekidačke mreže u digitalnim sistemima.
- ❑ Poznavanje njihove realizacije nije neophodno za njihovu primenu i uključivanje u složenije logičke strukture; dovoljno je znati **veze između ulaza i izlaza**.

Vrste standardnih modula:

- **kombinacioni** – sadrže samo logičke elemente (koderi, dekoderi, multiplekseri, demultiplekseri, sabirači, ALU itd.)
- **sekvencijalni** – osim logičkih, sadrže i memorijske elemente (registri, brojači, memorije itd.)

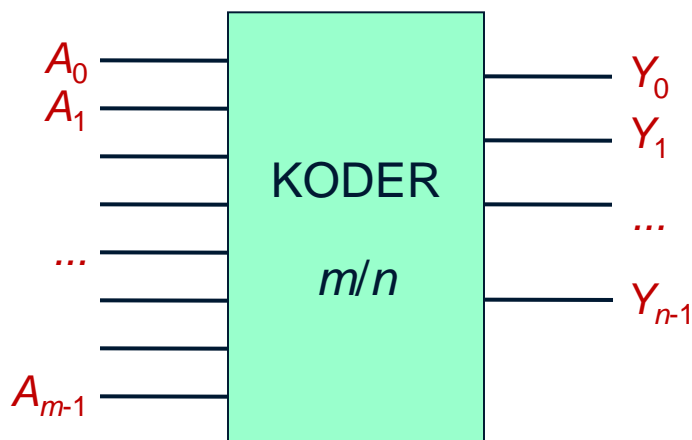
Kombinacioni moduli

- ☐ Koder
- ☐ Dekoder
- ☐ Multiplekser
- ☐ Demultiplekser
- ☐ Sabirač
- ☐ Aritmetičko-logička jedinica



Koder

Koder je kombinaciona mreža koja obavlja funkciju kodovanja informacija. **Aktivan signal** (na pr. 1) koji predstavlja informaciju dovodi se **samo na jedan od ulaza mreže**, dok je na ostalim ulazima neaktivan signal, tj. 0. Na izlazu se dobija kodovana informacija u vidu **binarnog broja koji odgovara rednom broju aktivnog ulaza**.



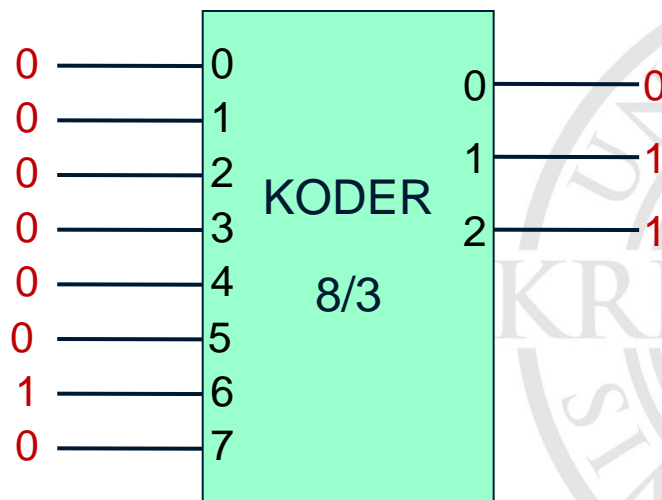
U zavisnosti od broja ulaza i izlaza, koderi mogu biti:

- ❖ **potpuni**, kod kojih važi $m = 2^n$
 - ima n izlaza i 2^n ulaza
- ❖ **nepotpuni**, kod kojih je $m < 2^n$
 - ima n izlaza i manje od 2^n ulaza

Napomena: Ukoliko se istovremeno na dva ili više ulaza koderu dovede aktivan signal, koder će na izlazu generisati pogrešan kod.

Potpuni koder 8/3

- ❑ Potpuni koder 8/3 ima 8 ulaza i 3 izlaza.
- ❑ U datom trenutku samo jedan od ulaza koderu može biti aktivan.
- ❑ U tom trenutku, u zavisnosti od toga koji je ulaz aktivan, na izlazu se generiše binarna kombinacija bitova koja odgovara rednom broju aktivnog ulaza.



Potpuni koder 8/3

Realizacija

Kombinaciona tablica

A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	Y_2	Y_1	Y_0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	1
0	1	0	0	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	1	1

Funkcije izlaza

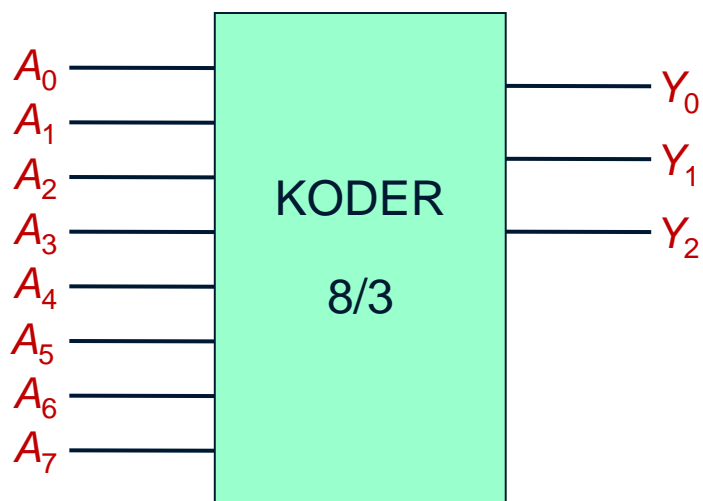
$$Y_0 = A_1 + A_3 + A_5 + A_7$$

$$Y_1 = A_2 + A_3 + A_6 + A_7$$

$$Y_2 = A_4 + A_5 + A_6 + A_7$$

Potpuni koder 8/3

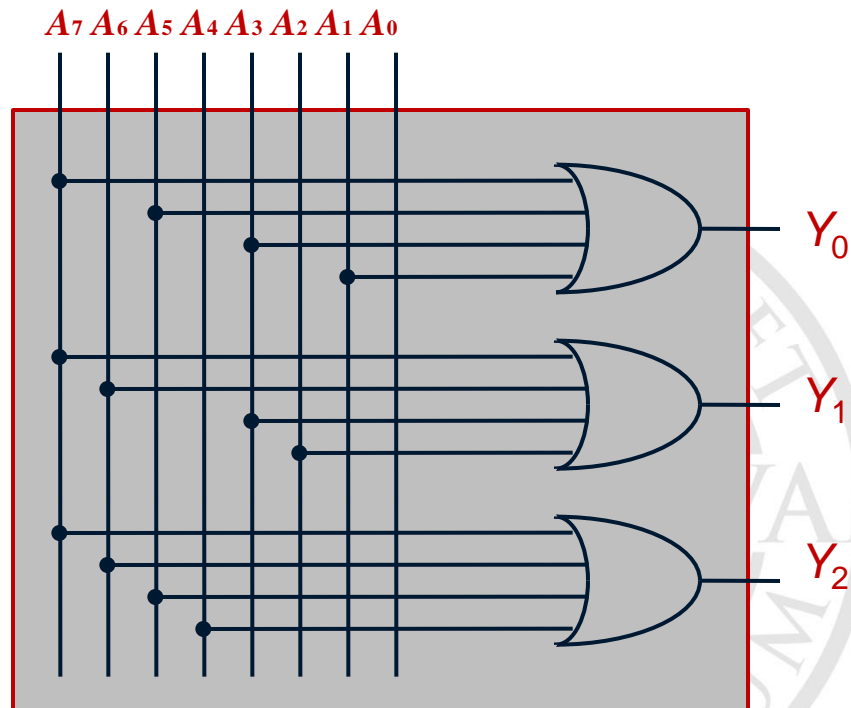
Realizacija



$$Y_0 = A_1 + A_3 + A_5 + A_7$$

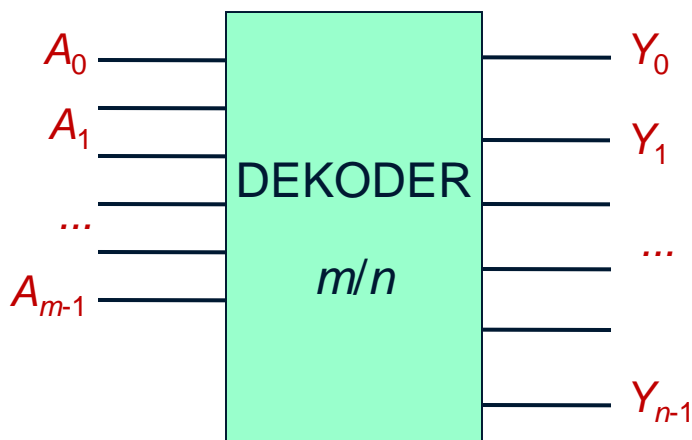
$$Y_1 = A_2 + A_3 + A_6 + A_7$$

$$Y_2 = A_4 + A_5 + A_6 + A_7$$



Dekoder

Dekoder je kombinaciona mreža koja obavlja funkciju dekodovanja binarno kodirane informacije dovedene na njegove ulaze. Aktivira se samo jedan izlaz čiji redni broj odgovara ulaznoj binarnoj kombinaciji.

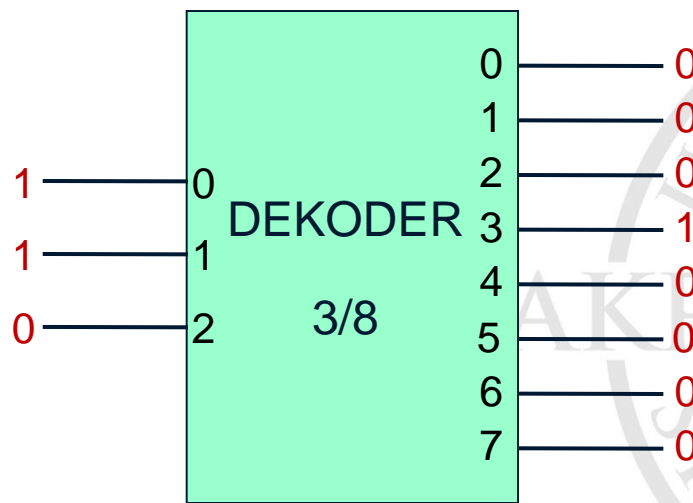


U zavisnosti od broja ulaza i izlaza, dekoderi mogu biti:

- ❖ **potpuni**, kod kojih važi $n = 2^m$
 - ima m ulaza i 2^m izlaza
- ❖ **nepotpuni**, kod kojih je $n < 2^m$
 - ima m ulaza i manje od 2^m izlaza (neke ulazne kombinacije se ne mogu pojaviti)

Potpuni dekodler 3/8

- ❑ Potpuni dekodler 3/8 ima 3 ulaza i 8 izlaza.
- ❑ U datom trenutku na ulaz se dovodi 3-bitna binarna kombinacija.
- ❑ U tom trenutku, u zavisnosti od ulazne kombinacije, aktivira se samo jedan izlaz i to onaj čiji redni broj odgovara toj ulaznoj kombinaciji.



Potpuni dekodler 3/8

Realizacija

Kombinaciona tablica

A_2	A_1	A_0	Y_7	Y_6	Y_5	Y_4	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	1	0	0
0	1	1	0	0	0	0	1	0	0	0
1	0	0	0	0	0	1	0	0	0	0
1	0	1	0	0	1	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

Funkcije izlaza

$$Y_0 = \overline{A}\overline{B}\overline{C}$$

$$Y_1 = \overline{A}\overline{B}C$$

$$Y_2 = \overline{A}B\overline{C}$$

$$Y_3 = \overline{A}BC$$

$$Y_4 = A\overline{B}\overline{C}$$

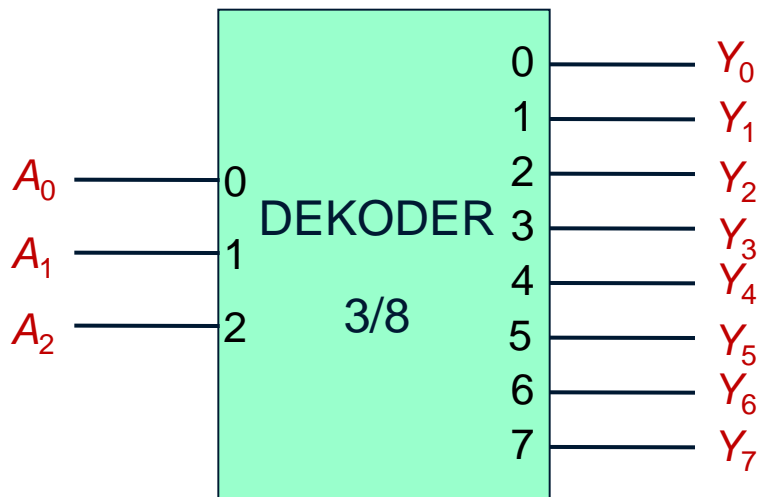
$$Y_5 = A\overline{B}C$$

$$Y_6 = AB\overline{C}$$

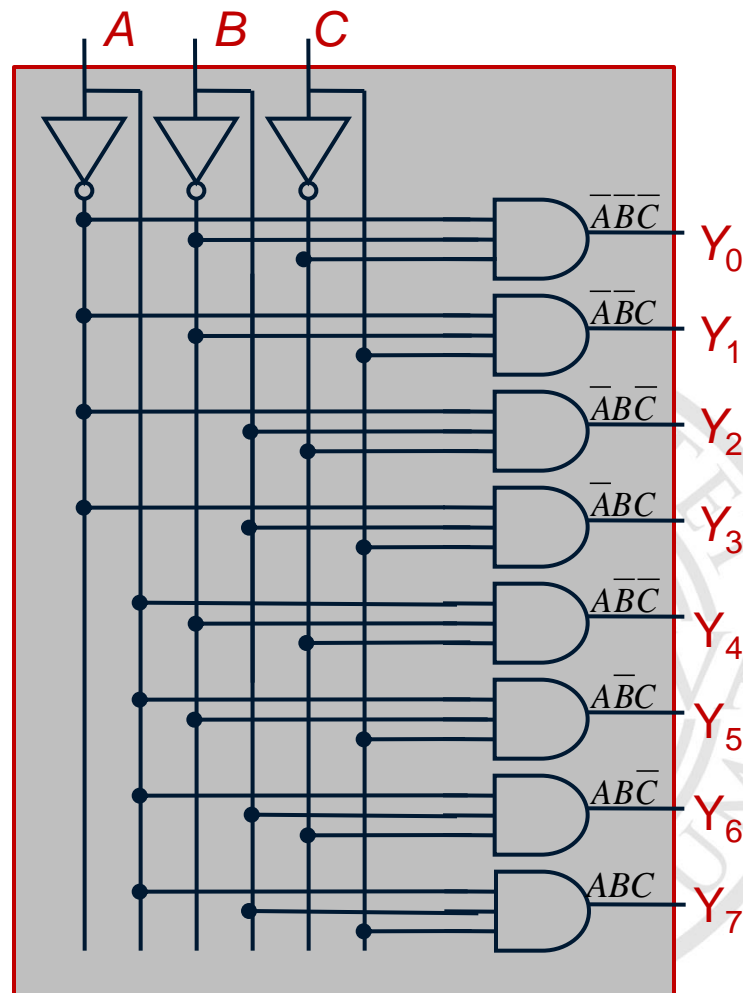
$$Y_7 = ABC$$

Potpuni dekoder 3/8

Realizacija

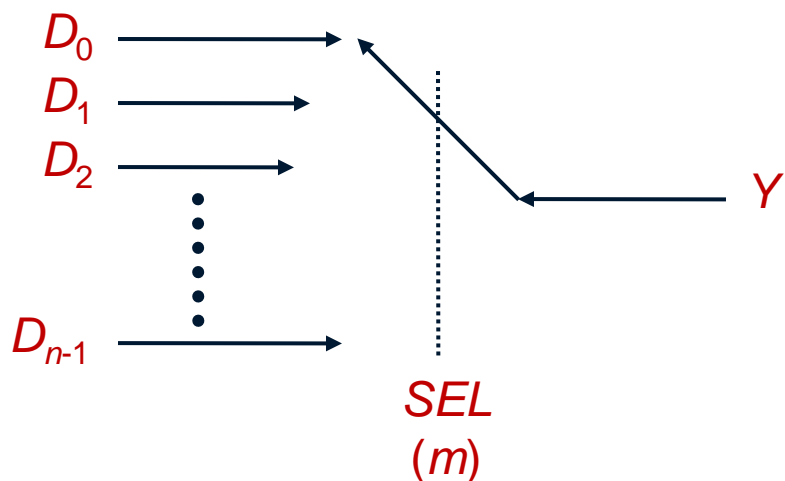


$Y_0 = \overline{A}\overline{B}\overline{C}$	$Y_4 = A\overline{B}\overline{C}$
$Y_1 = \overline{A}\overline{B}C$	$Y_5 = A\overline{B}C$
$Y_2 = \overline{A}B\overline{C}$	$Y_6 = AB\overline{C}$
$Y_3 = \overline{A}BC$	$Y_7 = ABC$



Multiplexer

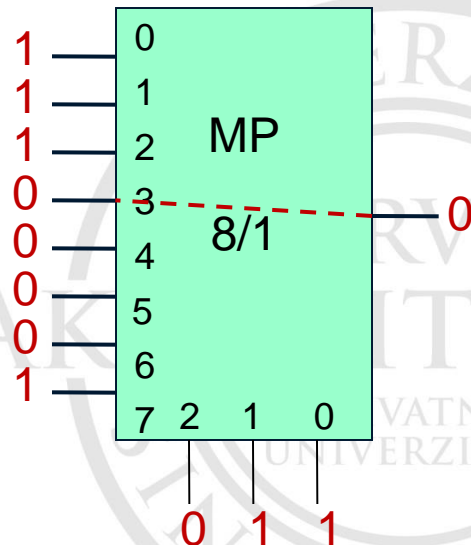
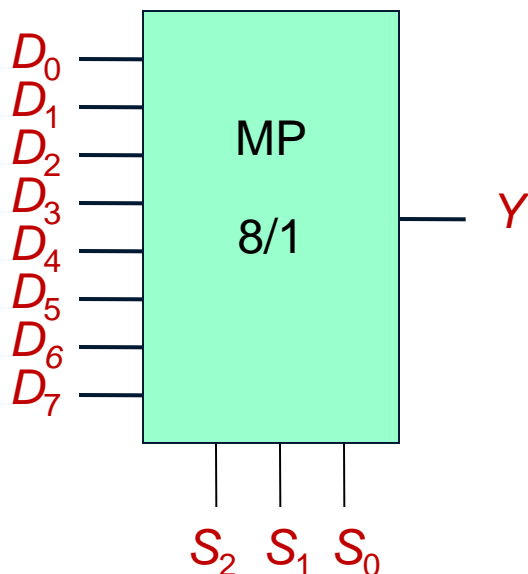
Multiplexer je kombinaciona mreža sa više informacionih ulaza (n), više selekcionih ulaza (m) i jednim izlazom (Y) koja obavlja funkciju **digitalnog višepoložajnog prekidača**. Zadavanjem selekcionih ulaza, prekidač se postavlja u položaj koji **direktno povezuje** informacioni ulaz čiji redni broj odgovara selekcionim signalima sa izlazom.



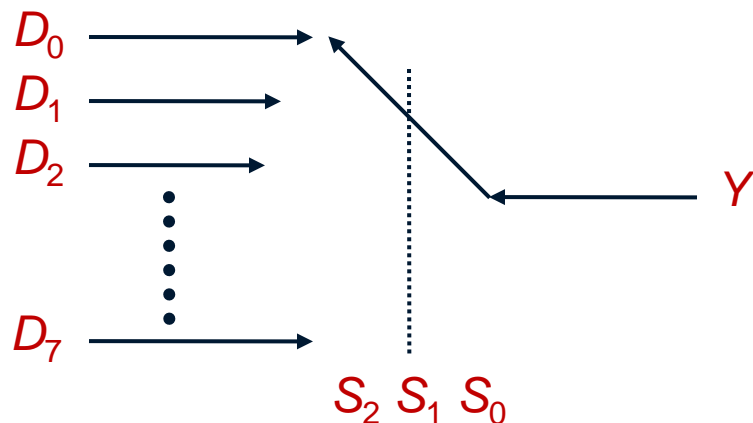
Za selekciju n informacionih ulaznih signala potrebno je $m = \log_2 n$ selekcionih signala, odnosno mora da važi $n = 2^m$.

Multiplexer 8/1

- ❑ Multiplexer 8/1 ima 8 informacionih ulaza, 1 izlaz i 3 selekciona signala.
- ❑ Dovođenjem selekcionih signala, formira se binarna kombinacija koja predstavlja redni broj ulaza na koji se postavlja prekidač, čime se vrednost sa tog ulaza direktno prosleđuje na izlaz.



Multiplexer 8/1



Kombinaciona tablica

S_2	S_1	S_0	Y
0	0	0	D_0
0	0	1	D_1
0	1	0	D_2
0	1	1	D_3
1	0	0	D_4
1	0	1	D_5
1	1	0	D_6
1	1	1	D_7

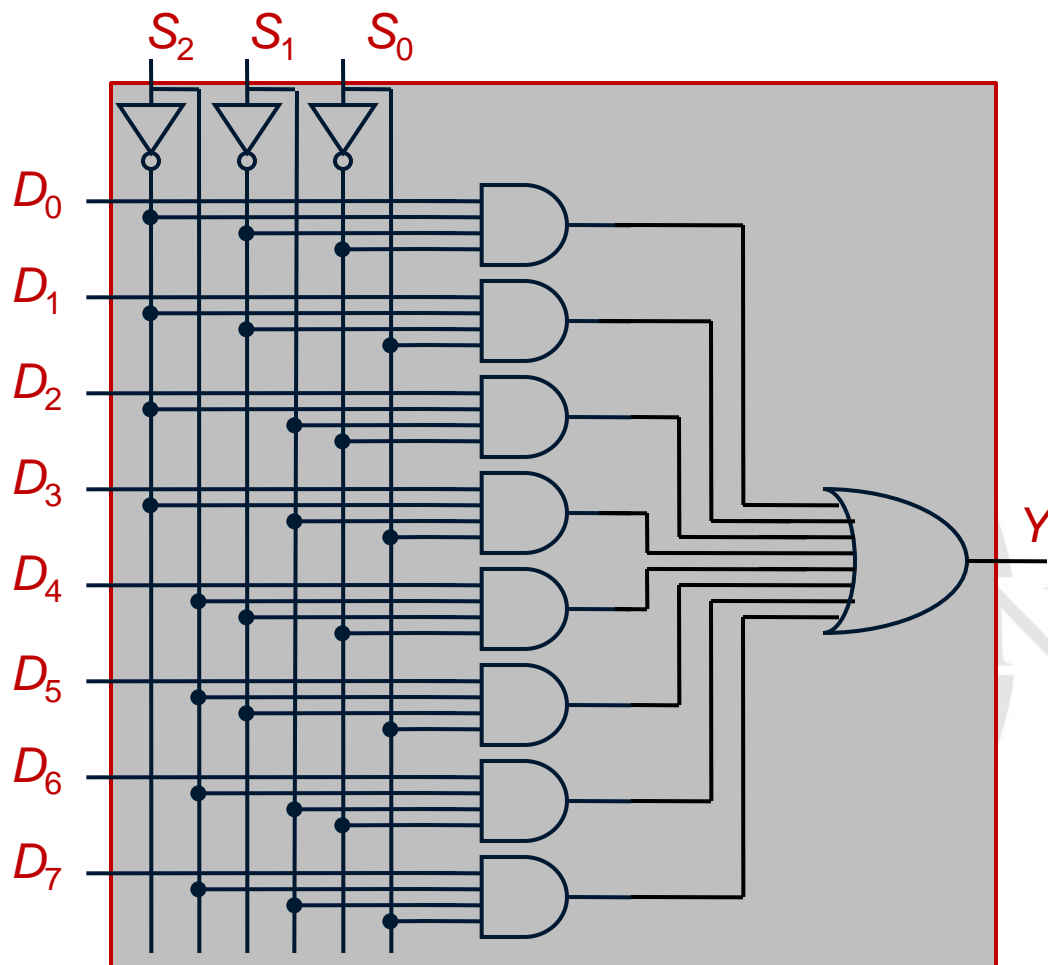
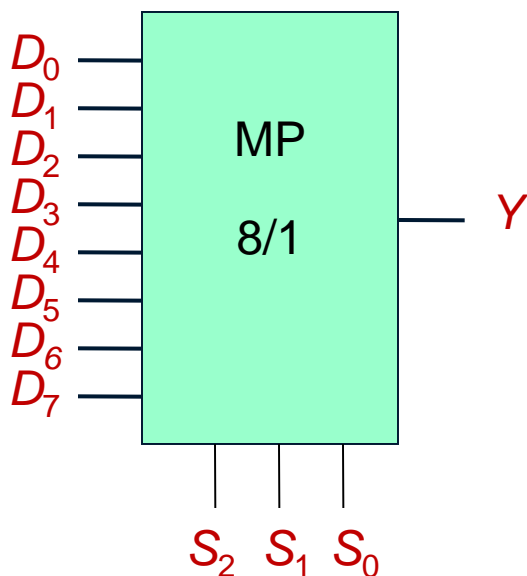
Funkcija izlaza

$$Y = D_0 \overline{S_2} \overline{S_1} \overline{S_0} + D_1 \overline{S_2} \overline{S_1} S_0 + D_2 \overline{S_2} S_1 \overline{S_0} + \dots + D_7 S_2 S_1 S_0$$

Multiplexer 8/1

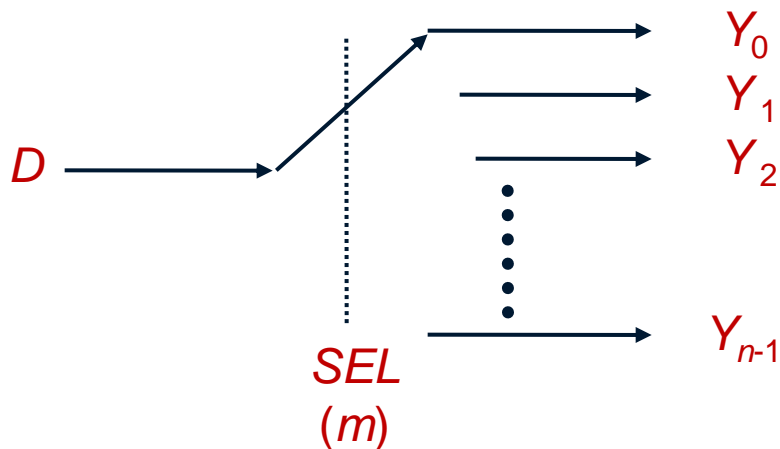
$$Y = D_0 \overline{S_2} \overline{S_1} \overline{S_0} + D_1 \overline{S_2} \overline{S_1} S_0 + \dots + D_7 S_2 S_1 S_0$$

Realizacija



Demultiplekser

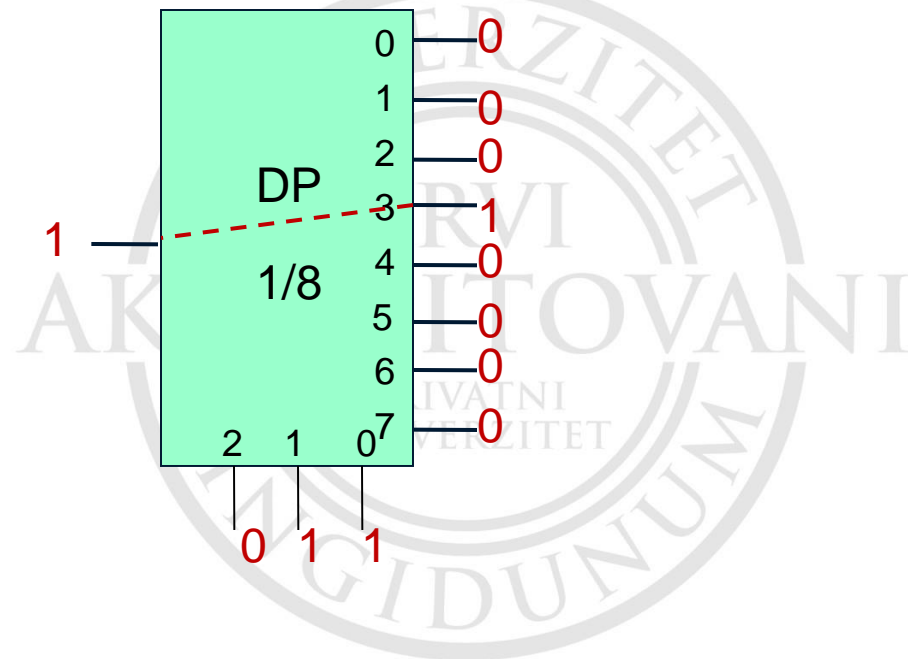
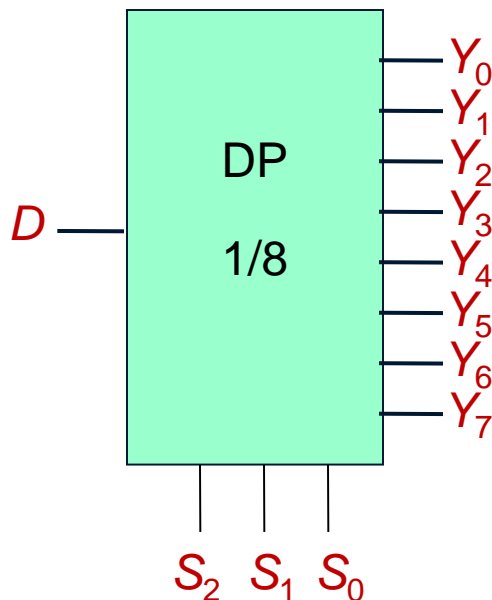
Demultiplekser je kombinaciona mreža sa jednim informacionim ulazom (D), više izlaza (n) i više selekcionih ulaza (m). Njegova funkcija je inverzna funkciji multipleksera. Zadavanjem selekcionih ulaza, prekidač se postavlja u položaj koji **direktno povezuje informacioni ulaz sa izlazom čiji redni broj odgovara selekcionim signalima**.



Za selekciju n izlaznih signala potrebno je $m = \log_2 n$ selekcionih signala, odnosno mora da važi **$n = 2^m$** .

Demultiplekser 1/8

- Demultiplekser 1/8 ima jedan informacioni ulaz, 8 izlaza i 3 selekciona ulaza.
- Dovođenjem selekcionih signala S_2 , S_1 i S_0 formira se binarna kombinacija koja predstavlja redni broj izlaza na koji se postavlja prekidač, čime se vrednost sa informacionog ulaza direktno prosleđuje na taj izlaz.



Demultiplekser 1/8

Kombinaciona tablica

S_2	S_1	S_0	Y_7	Y_6	Y_5	Y_4	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	0	0	0	0	0	D
0	0	1	0	0	0	0	0	0	D	0
0	1	0	0	0	0	0	0	D	0	0
0	1	1	0	0	0	0	D	0	0	0
1	0	0	0	0	0	D	0	0	0	0
1	0	1	0	0	D	0	0	0	0	0
1	1	0	0	D	0	0	0	0	0	0
1	1	1	D	0	0	0	0	0	0	0

Funkcije izlaza

$$Y_0 = D\bar{S}_2\bar{S}_1\bar{S}_0$$

$$Y_1 = D\bar{S}_2\bar{S}_1S_0$$

$$Y_2 = D\bar{S}_2S_1\bar{S}_0$$

$$Y_3 = D\bar{S}_2S_1S_0$$

$$Y_4 = DS_2\bar{S}_1\bar{S}_0$$

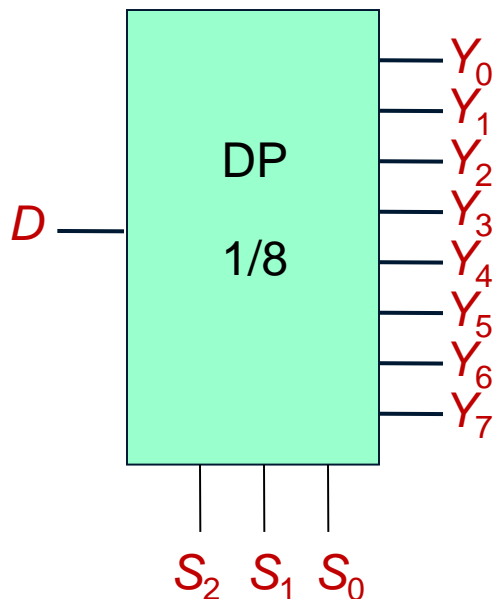
$$Y_5 = DS_2\bar{S}_1S_0$$

$$Y_6 = DS_2S_1\bar{S}_0$$

$$Y_7 = DS_2S_1S_0$$

Demultiplekser 1/8

Realizacija



$$Y_0 = D \bar{S}_2 \bar{S}_1 \bar{S}_0$$

$$Y_1 = D \bar{S}_2 \bar{S}_1 S_0$$

$$Y_2 = D \bar{S}_2 S_1 \bar{S}_0$$

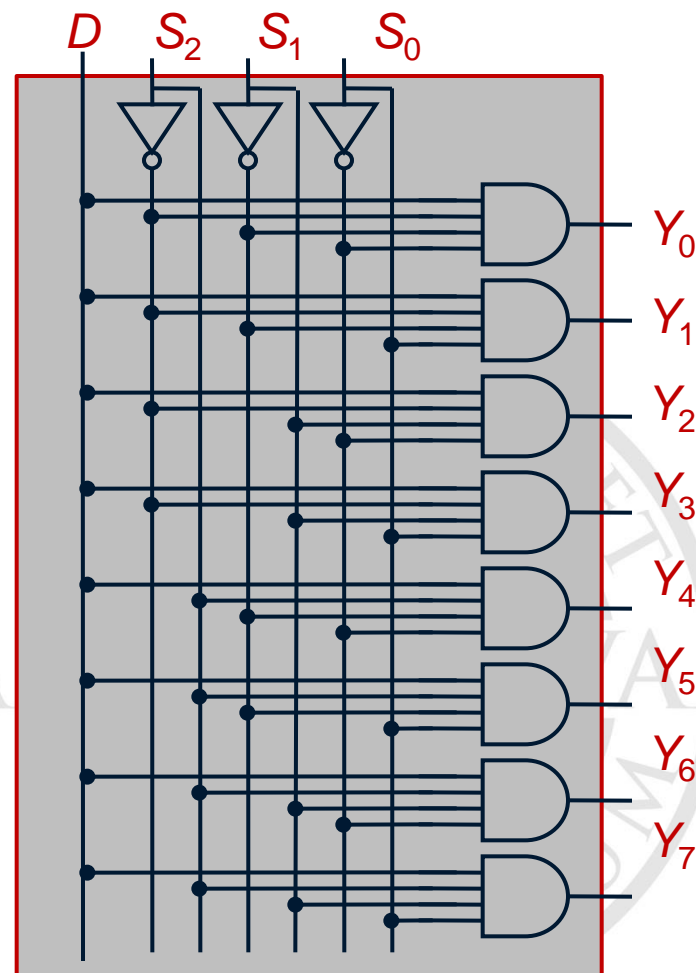
$$Y_3 = D \bar{S}_2 S_1 S_0$$

$$Y_4 = D S_2 \bar{S}_1 \bar{S}_0$$

$$Y_5 = D S_2 \bar{S}_1 S_0$$

$$Y_6 = D S_2 S_1 \bar{S}_0$$

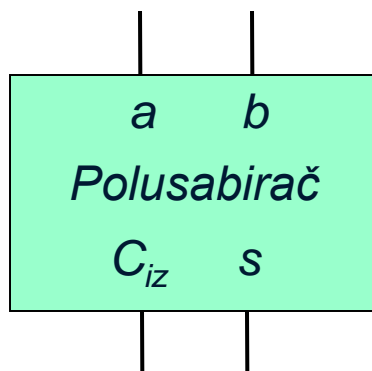
$$Y_7 = D S_2 S_1 S_0$$



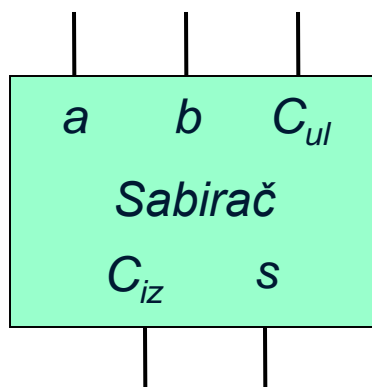
Sabirač

Sabirač je kombinaciona mreža koja omogućava **aritmetičko** sabiranje dva jednocifrena binarna broja.

Polusabirač

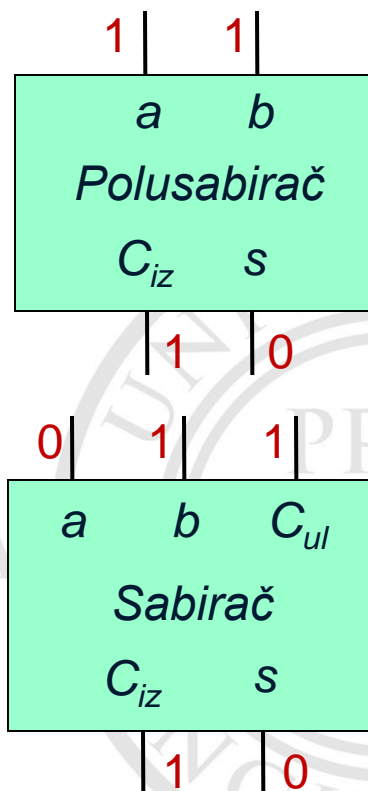


Potpuni sabirač



Sabiranje
višecifrenih brojeva

Ne može



Može

Sabiranje višecifrenih brojeva

- ❑ Ostvaruje se kaskadnom vezom više potpunih sabirača
 - broj potpunih sabirača u kaskadnoj vezi jednak je broju cifara u sabirku sa više cifara
 - veza između sabirača se ostvaruje vezivanjem izlaznog prenosa nižeg razreda na ulazni prenos višeg razreda
- ❑ Primenom potpunih sabirača mogu se sabirati kako neoznačeni brojevi, tako i označeni brojevi predstavljeni u komplementu dvojke.

Primer 1

Za sabiranje binarnih brojeva 10110 i 1100101 potrebno je 7 potpunih sabirača.

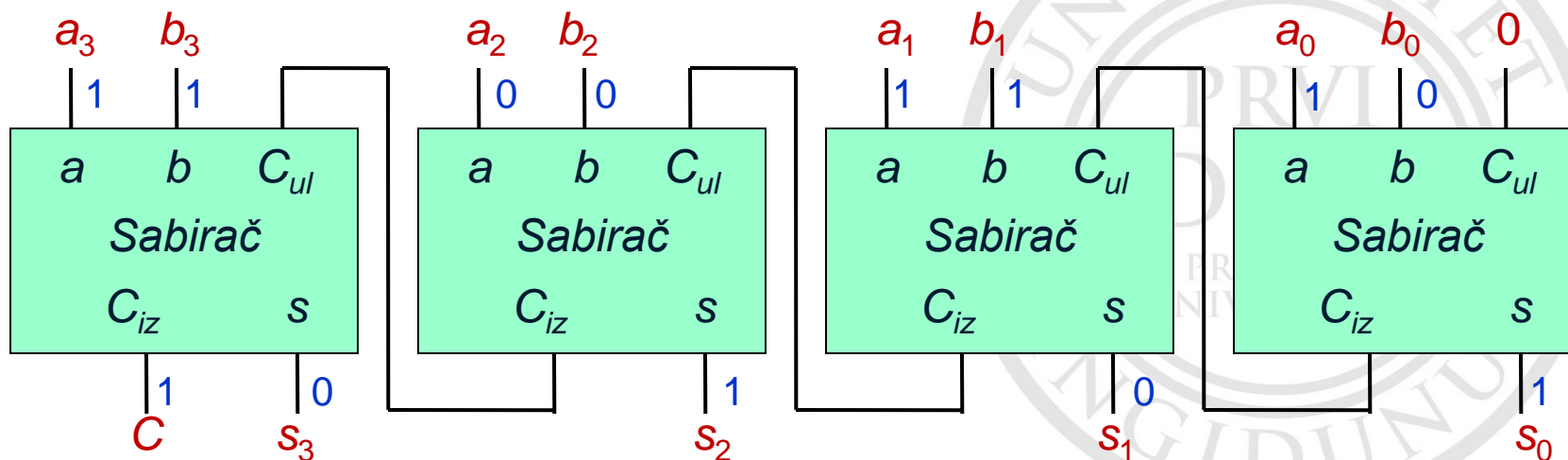
Kaskadna veza

Primer 2

Četvorobitni sabirač realizovati kaskadnom vezom potpunih sabirača.
Iskoristiti vezu za sabiranje brojeva 1011 i 1010.

$$\begin{array}{r} 1011 \\ + 1010 \\ \hline 10101 \end{array}$$

$$\begin{array}{r} a_3 a_2 a_1 a_0 \\ + b_3 b_2 b_1 b_0 \\ \hline C s_3 s_2 s_1 s_0 \end{array}$$



Aritmetičko-logička jedinica

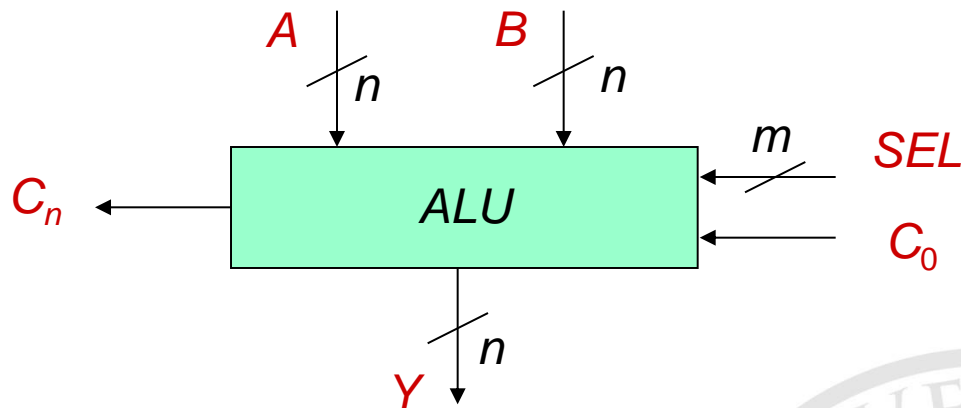
Aritmetičko-logička jedinica (*ALU – Arithmetic Logic Unit*) je centralni deo svakog procesora i mogućnosti procesora direktno zavise od karakteristika ove jedinice.

ALU predstavlja višefunkcionalnu kombinacionu mrežu koja može da obavlja različite **aritmetičke i logičke operacije nad dva n -bitska broja**.

Izbor operacije koju će u datom trenutku *ALU* izvršiti nad binarnim signalima koji se dovode na njen ulaz, zadaje se pomoću **upravljačkih selekcionih signala**.

Broj selekcionih signala definiše broj mogućih operacija koje *ALU* može da izvrši (**m selekcionih signala** → najviše **2^m operacija**).

Aritmetičko-logička jedinica



Ulazi ALU:

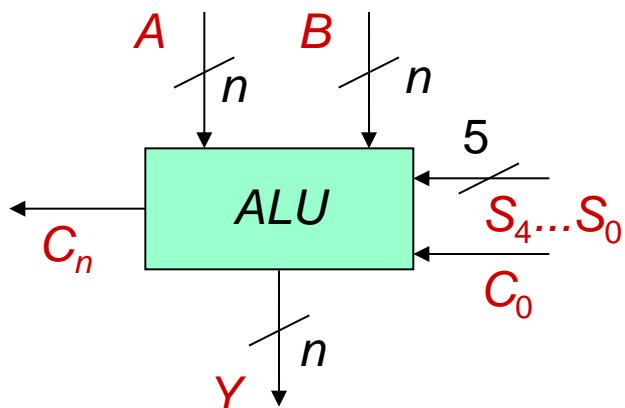
- **A** i **B** – binarni n -bitski brojevi nad kojima se obavlja operacija
- **C₀** – ulaz bitan za pojedine operacije
- **SEL** – m upravljačkih signala za adresiranje 2^m različitih operacija

Izlazi ALU:

- **C_n** – izlaz bitan za pojedine operacije
- **Y** – n -bitski binarni broj koji predstavlja rezultat operacije

ALU sa 5 selekcionih ulaza

Može da obavi 32 različite operacije (obično je 16 logičkih i 16 aritmetičkih).

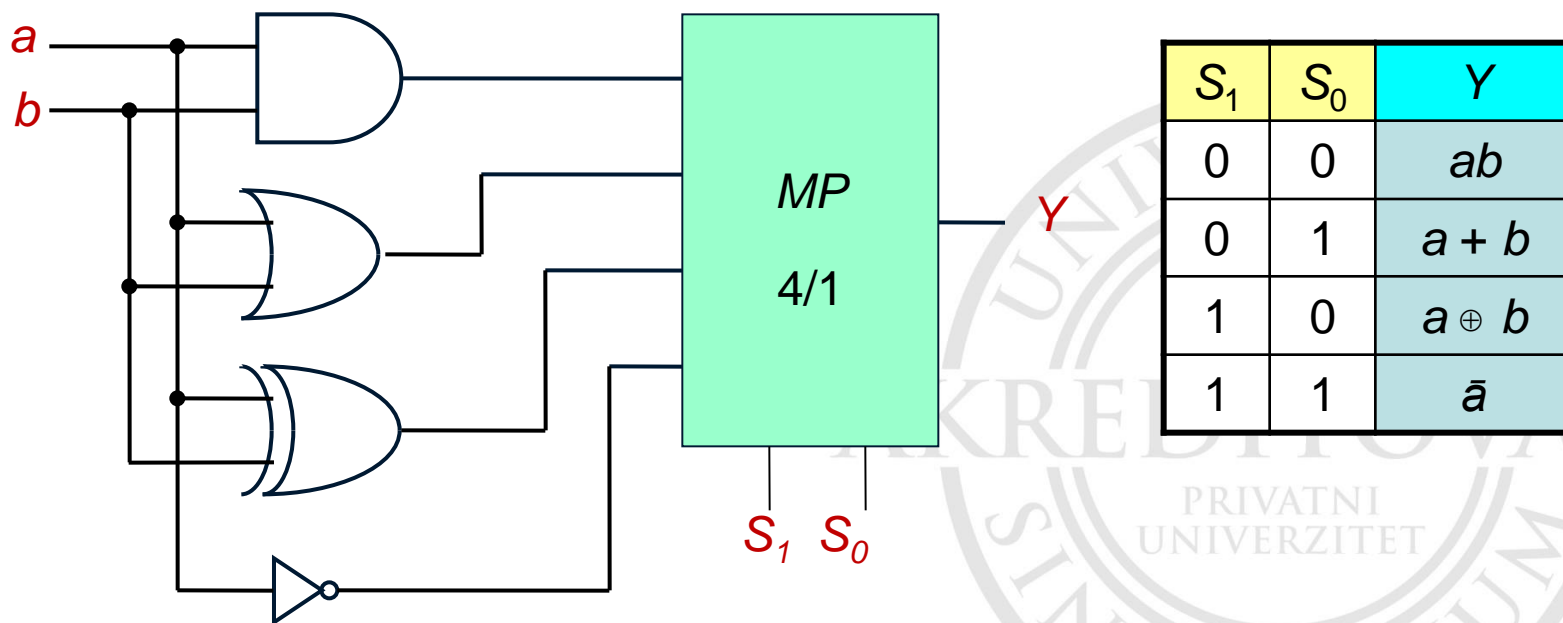


S ₃	S ₂	S ₁	S ₀	S ₄ = 1	S ₄ = 0
0	0	0	0	$Y = \bar{A}$	$Y = A \text{ plus } C_0$
0	0	0	1	$Y = \bar{A} + \bar{B}$	$Y = (A + B) \text{ plus } C_0$
0	0	1	0	$Y = \bar{A}B$	$Y = (A + \bar{B}) \text{ plus } C_0$
0	0	1	1	$Y = 0000$	$Y = 0 \text{ minus } \bar{C}_0$
0	1	0	0	$Y = \bar{A}\bar{B}$	$Y = A \text{ plus } (\bar{A}\bar{B}) \text{ plus } C_0$
0	1	0	1	$Y = \bar{B}$	$Y = (A + B) \text{ plus } (\bar{A}\bar{B}) \text{ plus } C_0$
0	1	1	0	$Y = A \oplus B$	$Y = A \text{ minus } B \text{ minus } \bar{C}_0$
0	1	1	1	$Y = \bar{A}\bar{B}$	$Y = (\bar{A}\bar{B}) \text{ minus } \bar{C}_0$
1	0	0	0	$Y = \bar{A} + B$	$Y = A \text{ plus } (\bar{A}B) \text{ plus } C_0$
1	0	0	1	$Y = \bar{A} \oplus B$	$Y = A \text{ plus } B \text{ plus } C_0$
1	0	1	0	$Y = B$	$Y = (A + \bar{B}) \text{ plus } (\bar{A}B) \text{ plus } C_0$
1	0	1	1	$Y = \bar{A}B$	$Y = (\bar{A}B) \text{ minus } \bar{C}_0$
1	1	0	0	$Y = 1111$	$Y = A \text{ plus } A \text{ plus } C_0$
1	1	0	1	$Y = A + \bar{B}$	$Y = (A + B) \text{ plus } A \text{ plus } C_0$
1	1	1	0	$Y = A + B$	$Y = (A + \bar{B}) \text{ plus } A \text{ plus } C_0$
1	1	1	1	$Y = A$	$Y = A \text{ minus } C_0$

Realizacija *ALU*

Primer 3

Realizovati *ALU* koja obavlja osnovne logičke operacije *I*, *ILI*, *NE* i *ekskluzivno ILI* nad dva jednobitna binarna broja.



Sekvencijalni moduli

☐ Registar

☐ Brojač

☐ Memorija



Registar

Registri su sekvencijalne mreže koje služe za memorisanje malih količina binarnih podataka.

Vrste registara

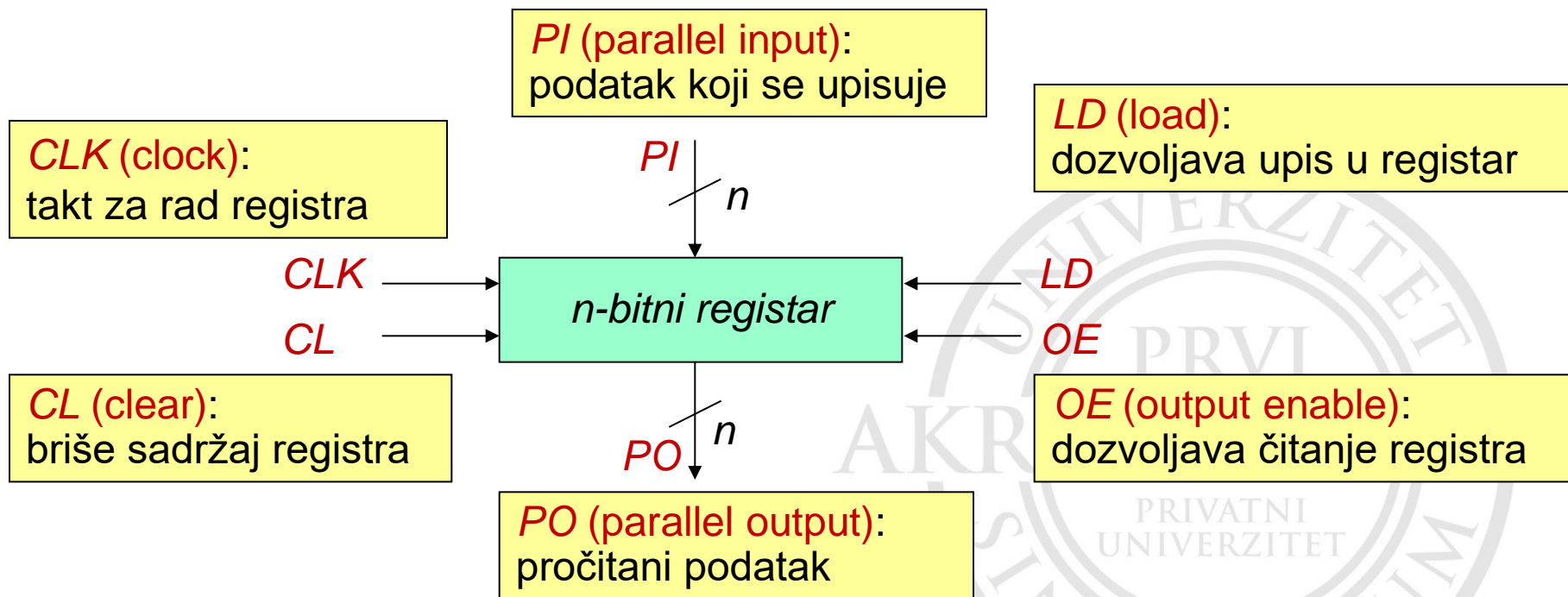
- ❖ paralelni registri
- ❖ serijski registri

- ❖ 8-bitni registri
- ❖ 16-bitni registri,...



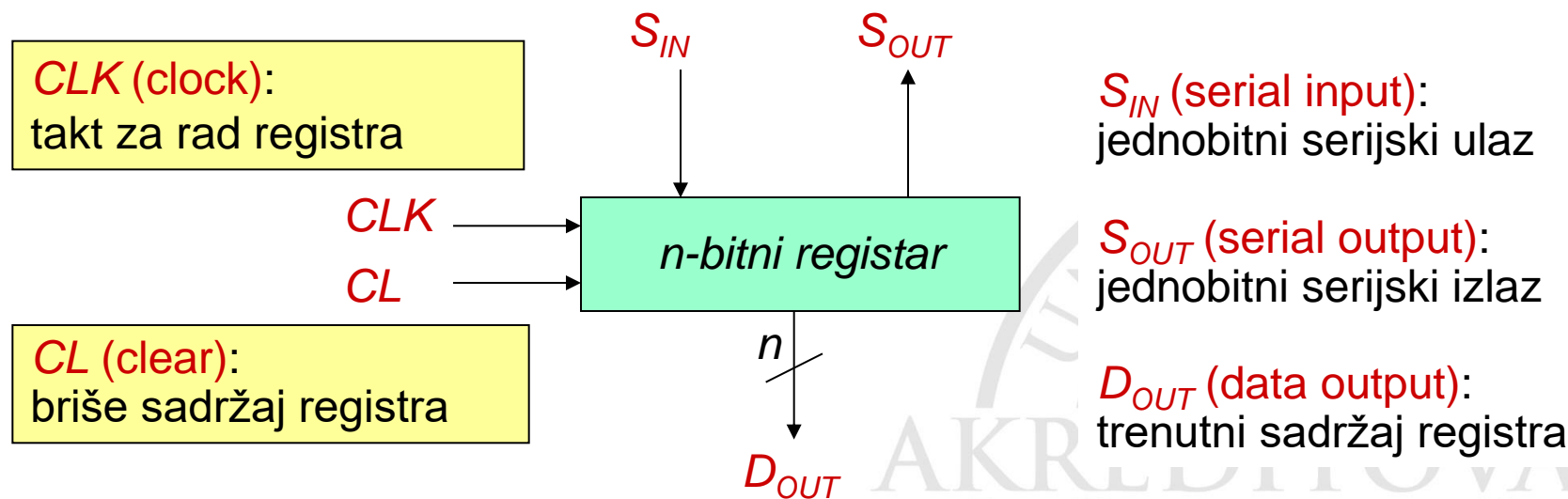
Paralelni registar

Paralelni registar omogućava upis informacije od n bitova istovremeno, tj. za vreme jednog takta.



Serijski registar

Serijski ili **pomerački registar** omogućava pomeranje sadržaja za jedno mesto od ulaza ka izlazu u skladu sa taktom. Koristi se u slučajevima kada je potrebno omogućiti serijski prijem ili slanje podataka (bit po bit).



Serijski prenos je sporiji, jeftiniji i manje podložan greškama od paralelnog.

Brojač

Brojači su sekvencijalne mreže koje po nailasku signala takta generišu na svom izlazu binarne brojeve u rastućem ili opadajućem redosledu.

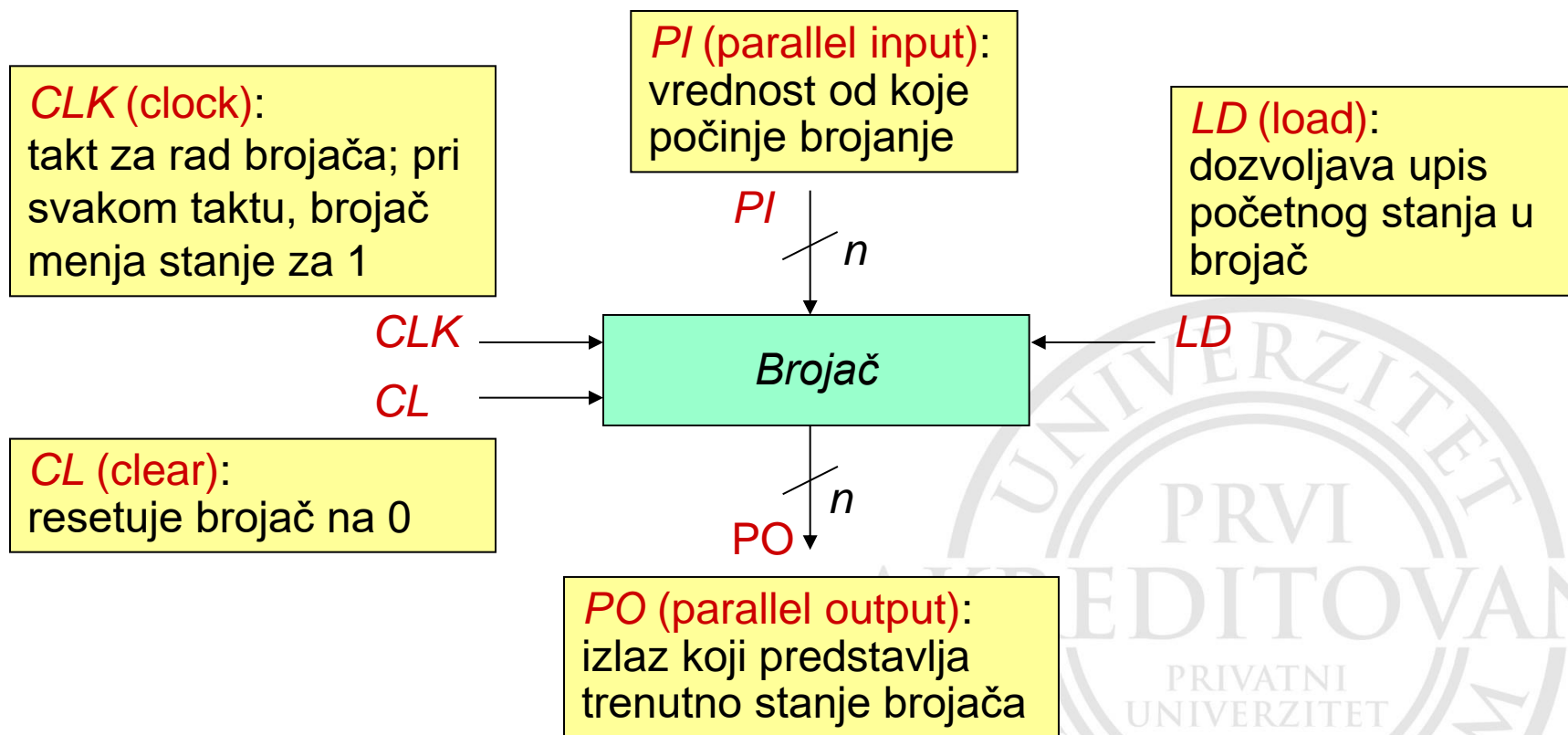
Moduo brojača: broj različitih brojeva koje brojač može da generiše

Brojač po modulu m broji od 0 do $m-1$ (ako broji unapred), a zatim se resetuje i ponovo počinje da broji od 0.

Vrste brojača

- ❖ inkrementirajući (broje unapred)
- ❖ dekrementirajući (broje unazad)

Brojač

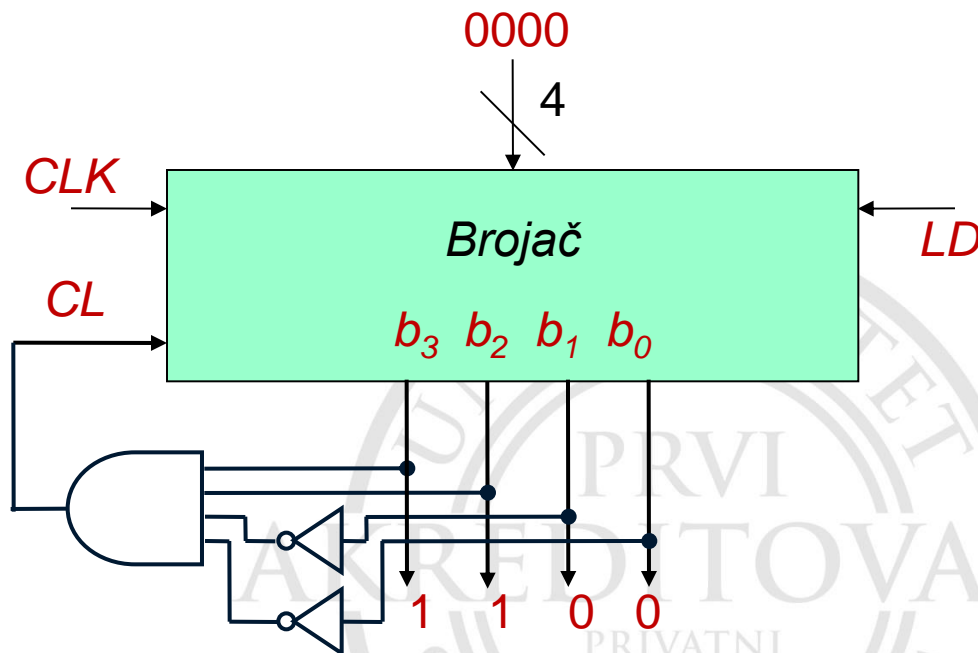


Brojač

Primer 4 Realizovati brojač po modulu 13.

Osobine:

- broji od 0 do 12
- pošto je $12_{(10)} = 1100_{(2)}$, brojač je 4-bitni
- kada sadržaj brojača postane 12, brojač treba resetovati na 0 u narednom taktu



Memorija

Memorije služe za **pamćenje** većih količina binarnih podataka.

Karakteristike idealne memorije

- visoka gustina pakovanja
- trajnost čuvanja podataka
- kratko vreme upisa i čitanja podataka
- veliki broj upisa pre otkaza memorije
- niska potrošnja
- niska cena

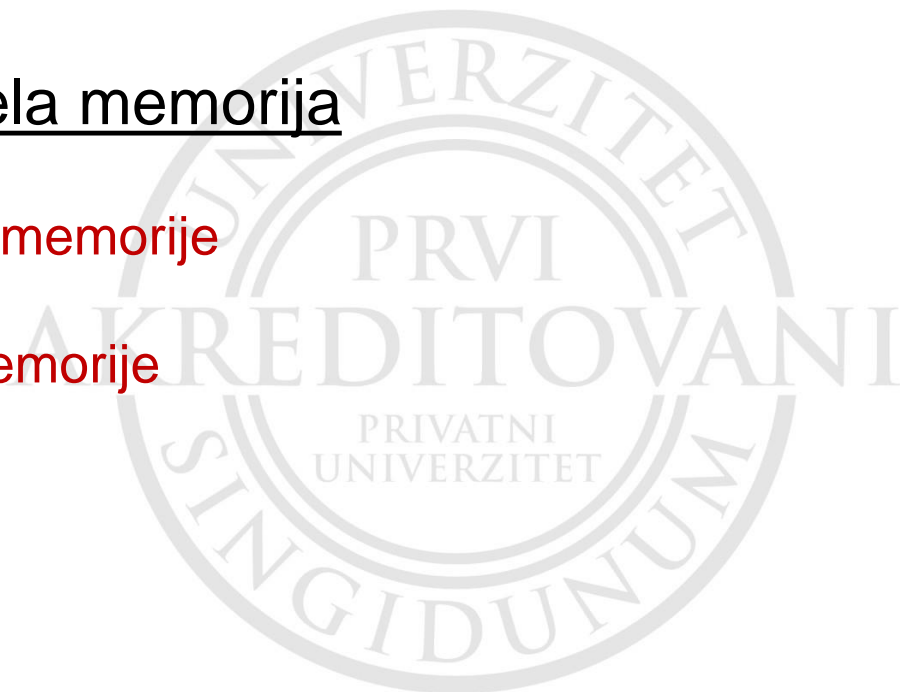


Memorija

- ❑ S obzirom da nijedna vrsta memorije ne zadovoljava sve karakteristike idealne memorije, razvijeno je više vrsta memorija koje, prema potrebi, zadovoljavaju samo neke od navedenih karakteristika.

Osnovna podela memorija

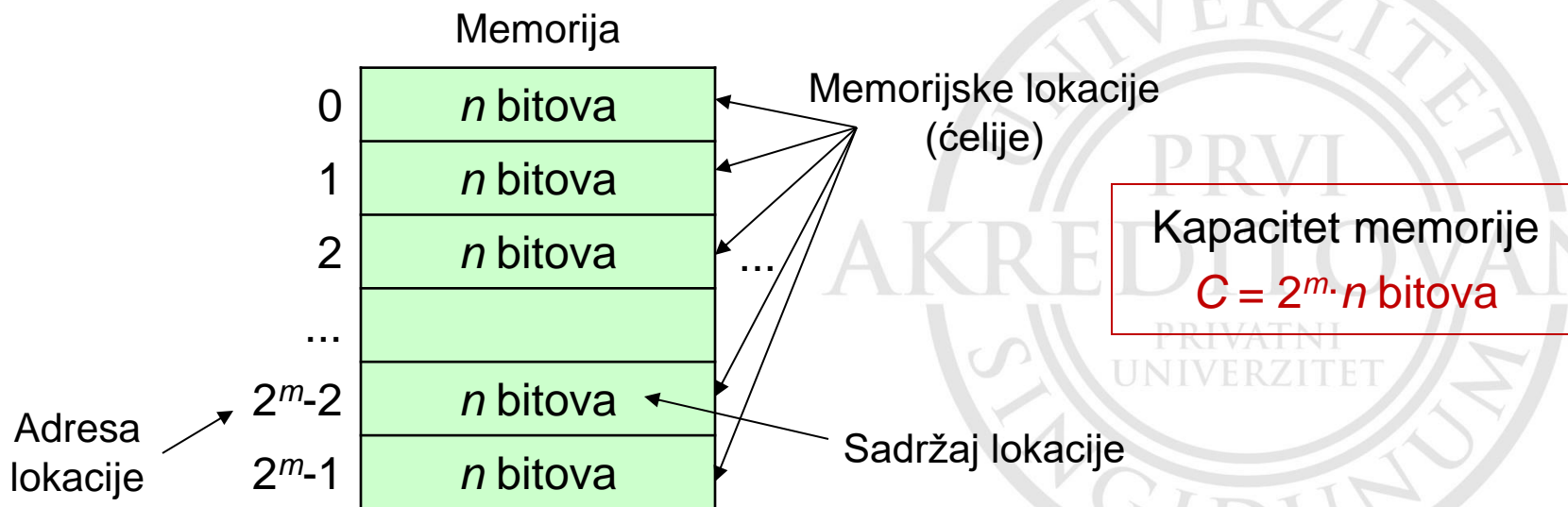
- nepermanentne memorije
- permanentne memorije



Nepermanentna memorija

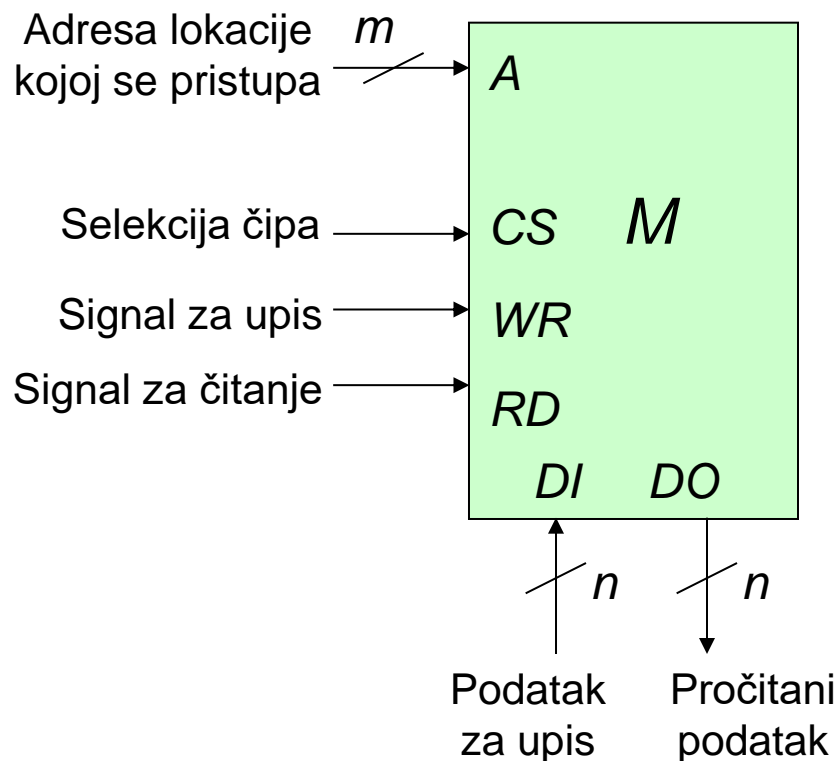
Nepermanentne memorije su memorije koje **gube sadržaj** koji je u njima zapisan ukoliko im se isključi napajanje.

- ❑ Najčešće korišćena nepermanentna memorija je **RAM** (*Random Access Memory*) – **memorija sa ravnopravnim pristupom**.



RAM

Princip rada



Upis podatka

- na A se dovede adresa lokacije za upis
- na DI se dovede podatak za upis
- na CS i WR se dovedu aktivni signali
- upisivanje podatka

Čitanje podatka

- na A se dovede adresa sa koje se čita
- na CS i RD se dovedu aktivni signali
- na DO se pojavi pročitani podatak

Napomena:

Pristup lokaciji ne menja njen sadržaj.

RAM

RAM se može realizovati na različite načine.

SRAM (Static RAM)

Napravljen od memorijskih elementa.

Karakterisitke:

- mala gustina pakovanja
- velika brzina pristupa (reda *ns*)
- mala verovatnoća greške
- mala potrošnja električne energije
- loš odnos kapacitet/cena

DRAM (Dynamic RAM)

Napravljen od elektronskih komponenata (neophodno osvežavanje na par *ms* koje traje reda *ns* - zauzeće oko 2% vremena).

Karakterisitke:

- velika gustina pakovanja
- mala brzina rada
- manja pouzdanost (veća verovatnoća greške)
- mala potrošnja električne energije
- niska cena
- proizvodnja je jednostavna

Permanentna memorija

Permanentne memorije su memorije koje pamte svoj sadržaj i u slučaju da ostanu bez napajanja.

✓ *ROM – Read Only Memory*

- najjednostavniji postupak
- sadržaj se upisuje prilikom proizvodnje čipa
- upisan sadržaj se ne može menjati

✓ *PROM – Programmable ROM*

- fleksibilniji
- korisnik može sam da upisuje sadržaj
- upisani sadržaj se ne može menjati

✓ *EPROM – Erasable PROM*

- ima mogućnost brisanja sadržaja (UV svetlošću kroz stakleni prozor na kućištu)
- brisanje je moguće samo nekoliko desetina puta

✓ *Flash*

- poluprovodnička memorija
- upisno/čitajuća memorija
- velika brzina pristupa
- veliki kapacitet
- velika pouzdanost

Šta smo naučili?

