

Memorijski sistem



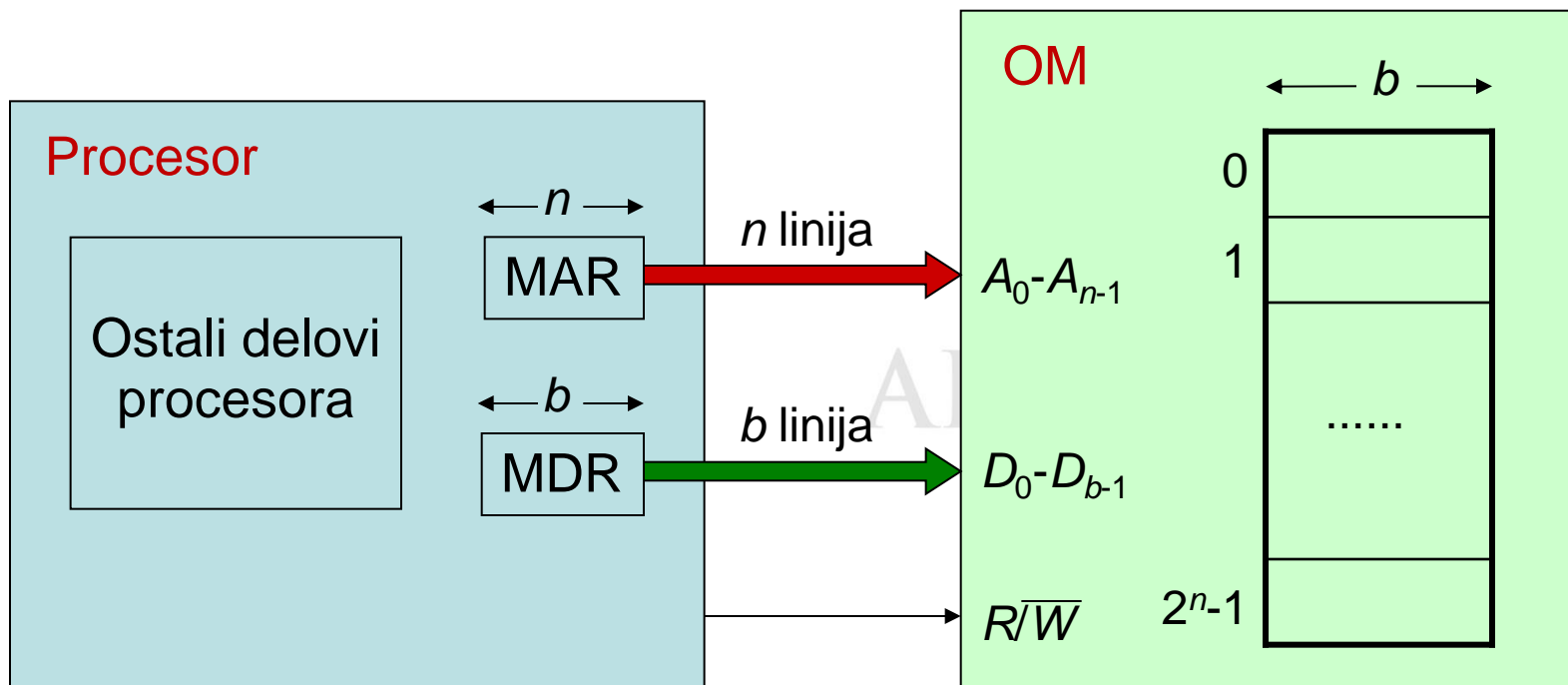
TEME

- ✓ Operativna memorija
- ✓ Virtuelna memorija
 - ✓ Stranična organizacija

Operativna memorija

OM je glavna, radna, sistemska memorija (eng. *Main Memory*) za smeštaj trenutno aktivnih programa (instrukcija i operanada) u računaru.

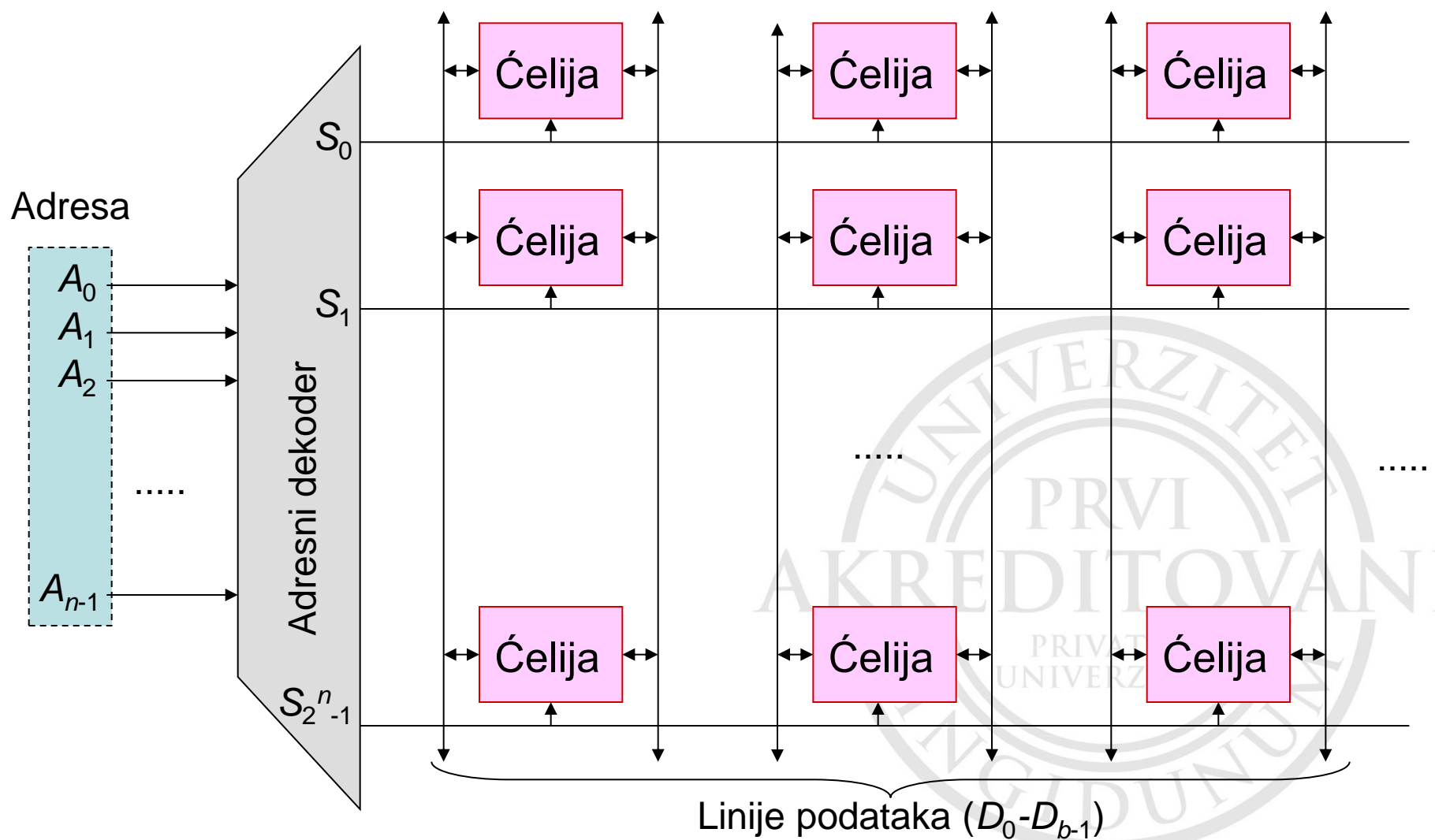
Veza između OM i procesora



Struktura OM

- ❑ Tipičnu strukturu OM čine **osnovne ćelije** (*basic cells*) raspoređene po vrstama i kolonama.
 - ❑ U osnovnu ćeliju se smešta informacija od 1 bita.
 - ❑ Sadržaji svih ćelija u jednoj vrsti formiraju jednu **memorijsku reč**.
 - ❑ Adresne linije se koriste kao ulazi za adresni dekoder koji generiše signale na **selekcionim linijama vrsta**.
 - ❑ U datom trenutku, adresni dekode
- aktivira samo jednu selekcionu liniju vrsta, dok su signali na svim ostalim linijama neaktivni.
- ❑ Selekciona linija vrste se koristi da omogući upis/čitanje svih ćelija u vrsti.

Struktura OM



Struktura OM

- ❑ Čelija može biti samo u jednom od dva stabilna stanja (0 ili 1).
- ❑ Svaka ćelija ima dve linije za podatke (zbog realizacije pomoću nekoliko tranzistora).
- ❑ Po jednoj liniji se bit unosi u ćeliju (upis), a po drugoj se bit iz ćelije prosleđuje dalje (čitanje).
- ❑ Jedna linija podataka je zajednička za sve ćelije u datoj koloni, ali u datom trenutku je aktivna samo jedna od ćelija.

Virtuelna memorija

- ❑ Kada procesor zahteva neku programsku instrukciju ili podatak, oni bi trebalo da budu u OM.
- ❑ U slučaju velikih programa, to bi zahtevalo OM velikog kapaciteta, što bi značajno povećalo njenu cenu.
- ❑ U praksi, nije neophodno smestiti kompletan program koji se izvršava ili sve podatke u OM.
- ❑ Delovi programa koji se trenutno ne izvršavaju mogu biti na hard disku, dok su delovi koji se trenutno izvršavaju u OM.
- ❑ Ovakva organizacija otežava rad OM jer je potrebno uvesti mehanizme za dovođenje/vraćanje delova programa i podataka prema potrebama.
- ❑ Ovi mehanizmi čine koncept **virtuelne memorije (VM)**.

Virtuelna memorija

- ❑ Koncept virtuelne memorije služi za “proširivanje” fizičke veličine OM na potrebnu veličinu.
- ❑ Podrazumeva se da za programiranje korisnici imaju na raspolaganju svu memoriju (OM i hard disk).
- ❑ Prenos delova programa i podataka sa hard diska u OM i obrnuto je transparentan za korisnika i obavlja ga operativni sistem.
- ❑ Principi rada virtuelne memorije su slični principima rada keš memorije, razlike su uglavnom u implementaciji.
- ❑ Glavna razlika je u tome što je cena “promašaja” u virtuelnom pristupu 100-200 puta veća od cene “promašaja” u keš pristupu (hiljade instrukcija mogu da se izvrše tokom transfera sa hard diska u OM).

Virtuelna i realna adresa

- ❑ Adresa koju generiše procesor ne odgovara fizičkoj, **realnoj adresi** u OM (na pr. 32-bitni procesor adresira 4GB, a OM može da bude 1GB); ova adresa se naziva **virtuelnom (logičkom) adresom**.
- ❑ Opseg adresa koje generiše procesor zove se **virtuelni adresni prostor - VAP**, dok opseg adresa u OM predstavlja **realni adresni prostor – RAP**.
- ❑ MMU je odgovorna za prevođenje virtuelne adrese u odgovarajuću realnu adresu u OM.
- ❑ Potrebno je voditi evidenciju o tome koji delovi programa se nalaze u OM i u kom njenom delu.
- ❑ Evidencija se vodi pomoću posebnih tabela koje se nalaze u OM i nazivaju se **tabele preslikavanja**.

Tipovi VM

1) VM sa straničnom organizacijom

2) VM sa segmentnom organizacijom

VAP se deli na delove promenljive veličine – segmente, koji se smeštaju u OM u prostor odgovarajuće veličine; kada se OM napuni, segment se vraća na disk i dovodi novi.

3) VM sa segmentno-straničnom organizacijom

VAP se deli na delove promenljive veličine – segmente, a zatim se segmenti dele na delove fiksne veličine – stranice; RAP se deli u blokove veličine jednake veličini stranice; stranice segmenata se smeštaju u blokove OM; kada se OM napuni, neka od stranica se vraća na disk i unosi nova u OM.

VM sa straničnom organizacijom

- ❑ VAP se deli na delove fiksne veličine koji se nazivaju **stranicama**, a RAP se deli na delove fiksne veličine koji se nazivaju **blokovima**.
- ❑ Veličina stranice je jednaka veličini bloka (obično 2-16KB dužine).
- ❑ Stranice ne treba da budu previše male, jer pristup disku dugo traje.
- ❑ Stranice ne treba da budu previše velike, jer se mnogi njihovi delovi ne bi koristili.
- ❑ Stranice se smeštaju u blokove OM.
- ❑ Kada su svi blokovi OM popunjeni, neka od stranica se vraća na hard disk kako bi nova stranica mogla da se unese u OM.

VM sa straničnom organizacijom

Virtuelna adresa ima dva polja:

- *page* – broj stranice, polje ima dužinu *p* bitova
- *word* – adresa reči u stranici, polje ima dužinu *w* bitova

Veličina VAP je 2^{p+w} .

Realna adresa ima dva polja:

- *block* – broj bloka, polje ima dužinu *b* bitova
- *word* – adresa reči u bloku, polje ima dužinu *w* bitova

Veličina RAP je 2^{b+w} .

VAP je znatno veći od RAP.

VM sa straničnom organizacijom

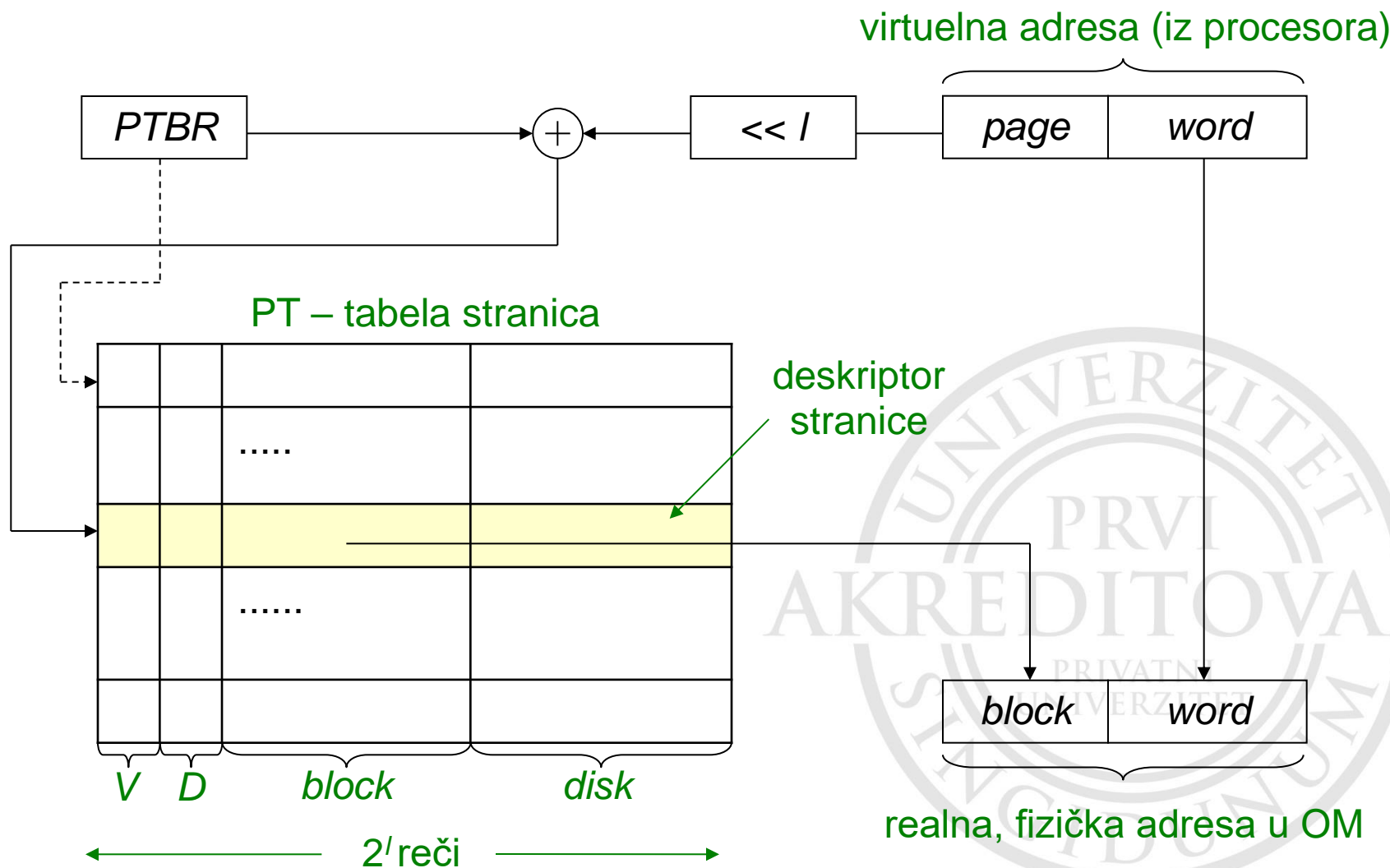
- ❑ Informacije potrebne za preslikavanje stranica VAP u blokove OM nalaze se u **tabeli stranica** (**PT – Page Table**).
- ❑ Informacije o stranici se nazivaju **deskriptor stranice**.
- ❑ Pošto u VAR ima 2^p stranica, i PT ima 2^p ulaza.
- ❑ PT se nalazi u OM (bilo bi bolje da je u MMU, ali to nije izvodljivo zbog veličine tabele), a početna adresa joj je u posebnom procesorskom registru **PTBR – Page Table Base Register**.
- ❑ Na osnovu sadržaja **PTBR** i broja stranice, MMU dolazi do deskriptora te stranice.

VM sa straničnom organizacijom

Deskriptor stranice se sastoji od 4 polja:

- V (1 bit)** – statusni bit koji pokazuje da li je stranica u OM; postavlja ga OS prilikom dovlačenja stranice sa diska u OM; omogućava da OS stranicu proglasi ne-validnom bez njenog uklanjanja iz OM
- D (1 bit)** – statusni bit koji pokazuje da li je stranica modifikovana; postavlja ga MMU ako je bilo operacije upisa; koristi ga OS da odredi da li stranicu treba upisati na disk pre izbacivanja iz OM
- block (b bitova)** – broj bloka OM u kome se nalazi data stranica (ako $V = 1$); polje postavlja OS pri dovlačenju stranice sa diska, a koristi MMU za formiranje realne adrese
- disk (d bitova)** – adresa stranice na disku

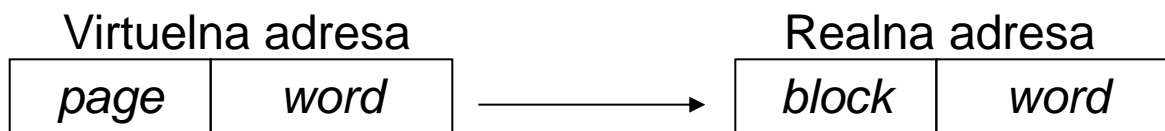
VM sa straničnom organizacijom



VM sa straničnom organizacijom

Preslikavanje virtuelne u realnu adresu

(realizuje se hardverski)



1. Polje *page* predstavlja broj ulaza u PT u kome je deskriptor stranice (DS); ako DS zauzima 2^l reči, broj ulaza se pretvara u pomeraj u odnosu na početak PT tako što se sadržaj polja *page* pomera za l mesta ulevo (pomeranje odgovara množenju sa 2^l).
2. U registru *PTBR* nalazi se početna adresa PT; pomeraj iz prethodnog koraka se sabira sa sadržajem *PTBR* i formira adresa traženog DS

VM sa straničnom organizacijom

Preslikavanje virtuelne u realnu adresu - nastavak (realizuje se hardverski)

3. Počev od formirane adrese, čita se reč koja odgovara polju *V* i na osnovu pročitane vrednosti proverava da li se stranica nalazi u OM.
4. Ako je stranica u OM, iz DS se čita sadržaj polja *block* koji predstavlja broj bloka u kome se nalazi data stranica; konkatencijom (spajanjem) sadržaja polja *block* i polja *word* iz virtuelne adrese formira se realna adresa.
5. Ako stranica nije u OM, generiše se prekid u cilju dovlačenja stranice sa hard diska; ovaj deo posla radi OS i on se realizuje softverski.