

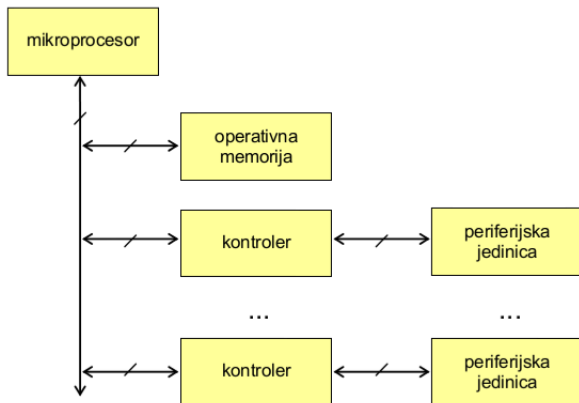
Ulazno izlazni podsistem

dr Predrag Teodorovic

Fakultet Tehničkih Nauka, Novi Sad

May 17, 2018

Organizacija U/I podsistema



Kontroleri

- ▶ Elektronske komponente koje služe za prilagođenje ili interfejs između perifernih jedinica i mikroprocesora
- ▶ Zašto? Zbog toga što su periferni jedinice specifične i nisu prilagođene načinu i brzini rada mikroprocesora
- ▶ Kontroleri omogućavaju efikasno upravljanje perifernim jedinicama i prilagođenje između perifernih jedinica i mikroprocesora u pogledu brzine rada, naponskih nivoa i sinhronizacije (primer: kontroler za tastaturu)
- ▶ Mikroprocesor nema neposredan pristup perifernim jedinicama, već “vidi” samo kontrolere i komunicira sa kontrolerima

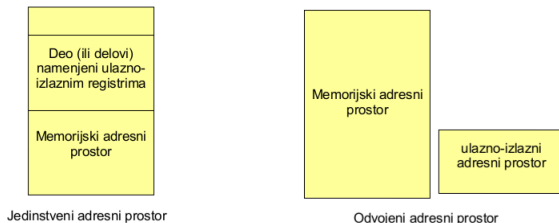
Programabilni registri

- ▶ Programabilne jedinice koje poseduju svoje upravljačke jedinice sposobne da obavljaju zadatke vezane za upravljanje i komunikaciju sa perferijskim jedinicama
- ▶ Poseduju programabilne registre preko kojih se razmenjuju upravljačke informacije i podaci sa mikroprocesorom
- ▶ Naravno, programabilni registri imaju svoje adrese i mikroprocesor mora da generiše tačnu adresu programabilnog registra kome pristupa
- ▶ U jednostavnim primerima ulazno/izlaznih podsistema, programabilni registri mogu da se neposredno koriste za ulaz i izlaz podataka (taster pritisnut na tastaturi može biti pročitano iz registra, podatak koji treba prikazati na monitoru takođe)

Ulazno/izlazni registri

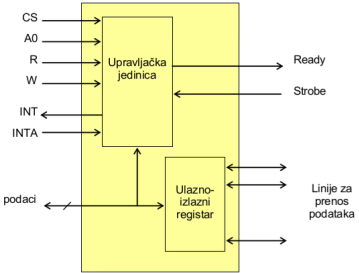
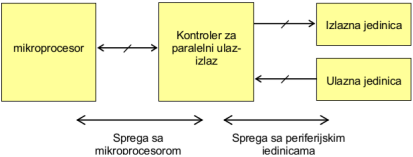
- ▶ Svi registri koji se nalaze u ulazno/izlaznom podsistemu
- ▶ Mogu da se nalaze u jedinstvenom adresnom prostoru zajedno sa memorijom ili da budu u odvojenom, ulazno/izlaznom adresnom prostoru
- ▶ Prednost jedinstvenog adresnog prostora je u tome što sve instrukcije koje se koriste za pristup memoriji mogu da se koriste i za pristup ulazno/izlaznim registrima i što se memorijski i ulazno/izlazni registri tretiraju na isti način
- ▶ Prednost odvojenog adresnog prostora je što omogućava jednostavniju organizaciju sistema u kome se jedan adresni prostor koristi isključivo za memorijske, a drugi za ulazno/izlazne komponente

Adresni prostor i operacije nad njim



M	IO	R	W	Operacija
1	0	1	0	Čitanje iz memorije
1	0	0	1	Upis u memoriju
0	1	1	0	Čitanje iz ulazno-izlaznog registra
0	1	0	1	Upis u ulazno-izlazni registar
1	1	x	x	Nije dozvoljeno
x	x	1	1	Nije dozvoljeno

Kontroler za paralelni ulaz/izlaz



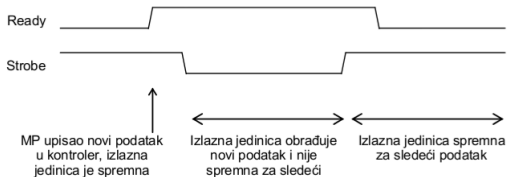
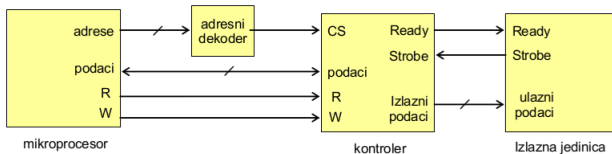
Sprega sa mikroprocesorom

- ▶ U mikroprocesorskom sistemu mora da postoji dekođer ulazno/izlaznih adresa koji generiše signal CS za aktiviranje kontrolera
- ▶ U slučaju da poseduje više ulazno/izlaznih registara kontroler mora da ima dodatne adresne signale (u ovom primeru samo A0 što omogućava postojanje dva ulazno/izlazna registra)
- ▶ Kontroler ima linije za pristup magistrali podataka mikroprocesorskog sistema preko kojih mikroprocesor komunicira sa kontrolerom
- ▶ Konačno, kontroler obično ima signal INT kojim zahteva prekid od mikroprocesora i signal INTA kojim mikroprocesor obaveštava kontroler da je prekid prihvaćen

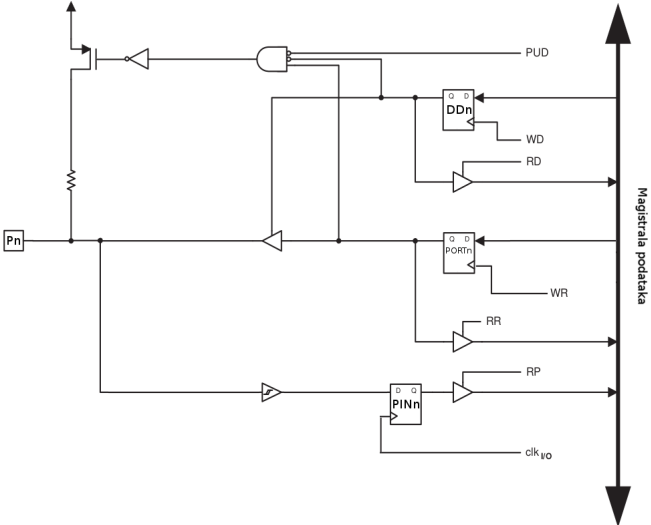
Sprega sa periferijskom jedinicom

- ▶ Linije za prenos podataka preko kojih se podaci šalju ka ili primaju od periferijske jedinice
- ▶ Signali za sinhronizaciju *Ready* i *Strobe*
- ▶ Uloge ova dva signala različite su kod ulaznih i izlaznih operacija (kontroler za paralelni ulaz/izlaz)
- ▶ Na primer, ako je u pitanju izlaz podataka prema periferijskoj jedinici:
 - ▶ $Ready = 0$ \Rightarrow na izlaznim linijama prema periferijskoj jedinici je 'stari' podatak
 - ▶ $Ready = 1$ \Rightarrow na izlaznim linijama je novi podatak
 - ▶ $Strobe = 0$ \Rightarrow periferijska jedinica nije spremna da prihvati novi podatak
 - ▶ $Strobe = 1$ \Rightarrow periferijska jedinica je spremna da prihvati novi podatak

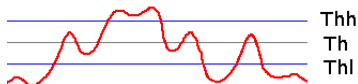
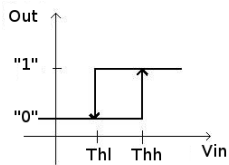
Sistem sa kontrolerom i izlaznom jedinicom



Primer: Digitalni ulazno/izlazni port kod AVR



Detalj: komparator sa histerezisom i signali



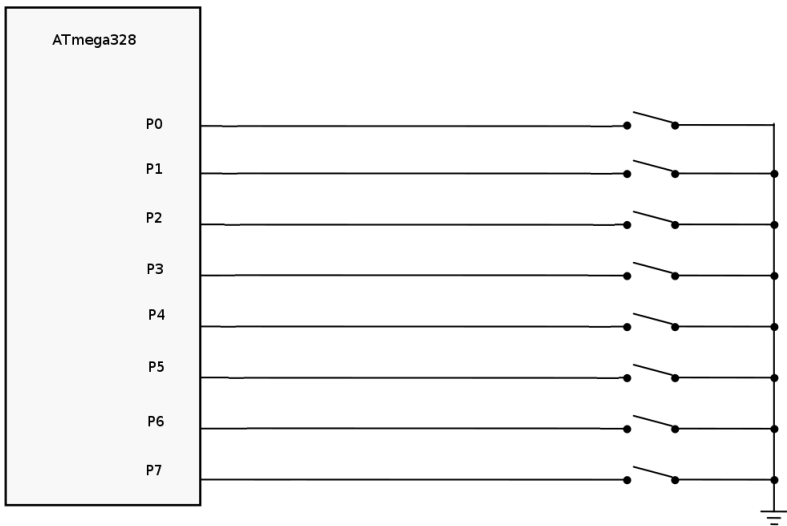
Izlaz komparatora sa jednom pragom Th



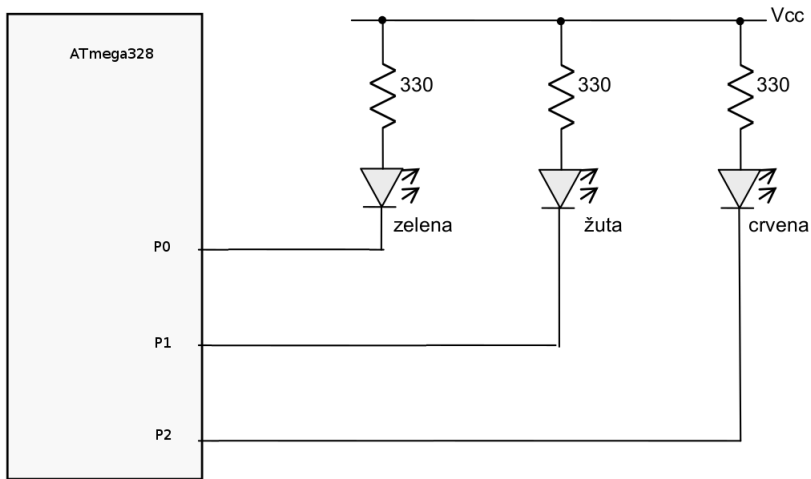
Izlaz komparatora sa histerezisom (Thh, Thl)



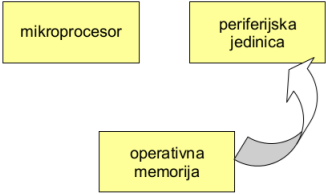
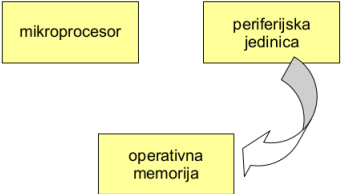
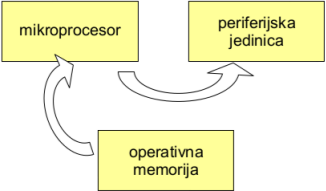
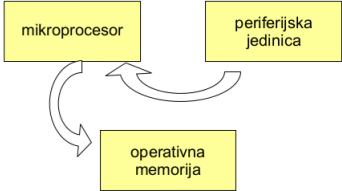
Kako koristimo U/I port kao ulaz?



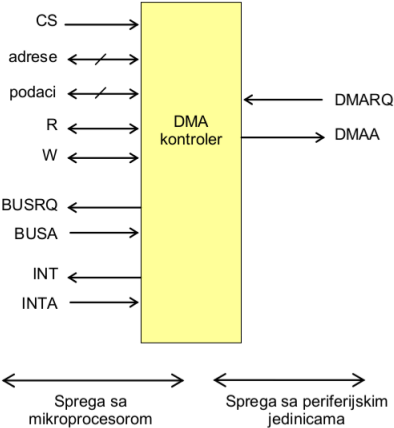
Kako koristimo U/I port kao izlaz?



Direktan pristup memoriji (DMA)



DMA kontroler koji ovo omogućava



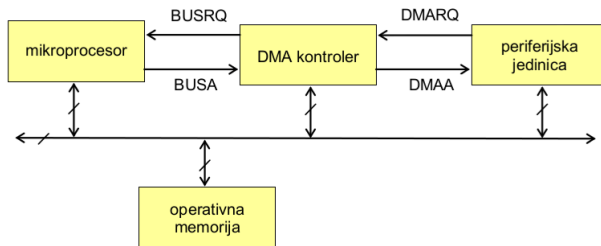
DMA kontroler, sinhronizacija sa mikroprocesorom

- ▶ Da bi organizovao direktan pristup memoriji DMA kontroler mora prvo da preuzme upravljanje magistralama i generiše adresne signale i signale R i W
- ▶ Pošto u svakom trenutku samo jedna jedinica može upravljati magistralama, DMA kontroler mora od mikroprocesora da traži dozvolu
- ▶ Signali *BUSRQ* (Bus Request) i *BUSA* (Bus Acknowledge) služe za sinhronizaciju pristupa magistralama između DMA kontrolera i mikroprocesora:
 - ▶ Logičkom 1 na liniji *BUSRQ* DMA kontroler traži od mikroprocesora upravljanje magistralama
 - ▶ logičkom 1 na liniji *BUSA* mikroprocesor obaveštava DMA kontroler da dozvoljava upravljanje magistralama

DMA kontroler, sinhronizacija sa periferijskom jedinicom

- ▶ Signalom DMARQ (DMA Request) periferijska jedinica obaveštava DMA kontroler da je spremna za DMA prenos
- ▶ Signalom DMAA (DMA Acknowledge) DMA kontroler obaveštava periferijsku jedinicu da je započeo ciklus DMA prenosa

Struktura sistema sa DMA kontrolerom



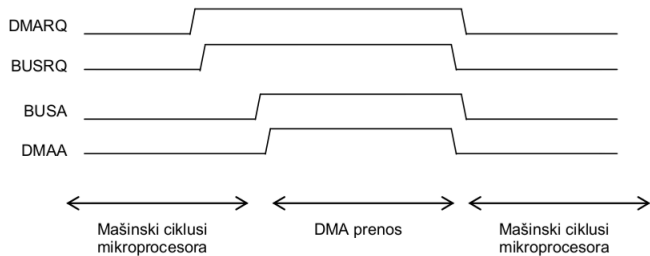
Pre DMA prenosa

- ▶ Mikroprocesor programira perifernu jedinicu i DMA kontroler
- ▶ Na primer, u slučaju diska, mikroprocesor u kontroler diska upisuje sektore kojima treba pristupiti, smer prenosa podataka i količinu podataka koje treba preneti, a u DMA kontroler mikroprocesor upisuje smer prenosa podataka, količinu podataka koje treba preneti i početnu memorijsku adresu zone u koju treba upisati podatke sa perifernu jedinicu ili iz koje treba čitati podatke koji se prenose perifernu jedinicu
- ▶ Nakon toga, mikroprocesor ne učestvuje u DMA prenosu i može da obavlja druge operacije
- ▶ DMA kontroler organizuje DMA prenos tako što za svaki podatak koji treba preneti između perifernu jedinicu i memorije, traži od mikroprocesora pristup magistralama i kada dobije dozvolu pristupa, generiše sve signale koji su neophodni za pristup memoriji

DMA prenos

- ▶ DMA prenos počinje tako što periferna jedinica signalom DMARQ javlja DMA kontroleru da je spremna za prenos
- ▶ DMA kontroler zatim aktivira signal BUSRQ kojim od mikroprocesora traži upravljanje magistralama
- ▶ Kada mikroprocesor završi tekući mašinski ciklus, prevodi svoje magistrale u stanje visoke impedanse i signalom BUSA obaveštava DMA kontroler da ima dozvolu pristupa magistralama
- ▶ Sada DMA kontroler signalom DMAA obaveštava perifernu jedinicu da započinje ciklus DMA prenosa
- ▶ U slučaju da se upisuje u memoriju, periferna jedinica prenosi podatak na magistralu podataka, dok DMA kontroler generiše adresne signale i signal W. Kao rezultat, podatak sa magistrale podataka memorija smešta u adresiranu memorijsku lokaciju i time se završava DMA prenos jednog podatka

Vremenski dijagrami



Nakon DMA prenosa

- ▶ Kod svakog DMA prenosa, DMA kontroler smanjuje za jedan sadržaj registra u kome se nalazi broj podataka koje treba preneti i inkrementira (ili dekrementira) adresu memorijske lokacije kojoj se pristupa u toku DMA prenosa
- ▶ Kada se prenesu svi podaci DMA kontroler obično generiše prekid kojim obaveštava mikroprocesor da je DMA prenos je završen. Mikroprocesor može da preduzme dalje akcije, na primer da obradi prenesene podatke ili da pripremi novi DMA prenos
- ▶ Efekat DMA prenosa može se posmatrati preko traga mikroprocesorskog sistema. Na primeru dve instrukcije A i B sa po tri mašinska ciklusa to izgleda kao na narednom slajdu

Vremenski dijagrami

