

Upravljanje ulazno-izlaznim prenosom

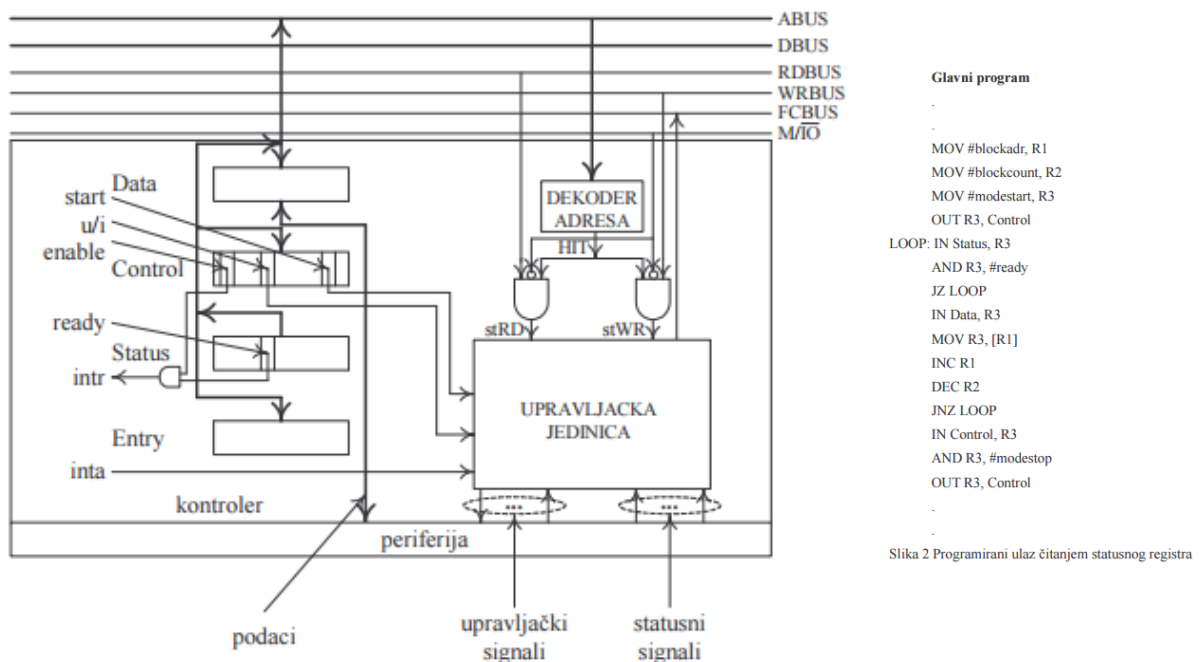
Kontroler dobija instrukcije od CPU i izvršava ih obavljajući zahtevani prenos. Svaki periferni uređaj ima svoj modul interfejsa koji dekodira instrukcije dovedene na magistralu i generiše signale za kontroler.

Sinhronizacija ulaza se sastoji u tome što interfejs dobija podatke brzinom koja odgovara mogućnostima periferije a šalje ih u CPU preko ui magastrale sa mnogo većom brzinom. Za sinhronizaciju izlaza interfejs dobija podatke sa magistrale i pamti ih privremeno u svom prihvatnom registru. Podaci se prenose periferiji brzinom kojom ih ona može prihvatiti.

Za razmenu podataka sa ui uređajem CPU postavlja adresu uređaja na adresne linije. U svakom modulu interfejsa priključenom na magistralu postoji dekodera adresa koji nadgleda adresne linije. Pri otkrivanju svoje adrese interfejs aktivira puteve od magistrale do kontolera periferije. CPU postavlja na drugu grupu linija magistrale i funkcionalni kod. Izabrani interfejs dekodira ovaj kod i prelazi u njegovo izvršavanje Funkcionalni kod se naziva i komandna reč.

Za ui prenos podataka postoje tri načina:

1. Programirani ulaz-izlaz



Slika 2 Programirani ulaz čitanjem statusnog registra

Slika 1 Kontroler bez direktnog pristupa memoriji

Razmena podataka se obavlja preko nekog registra u procesoru, najčešće akumulatora. Upravljanje procesom se vrši preko posebnog programa koji izvršava CPU.

Posle postavljanja adrese i upravljačke informacije, UI instrukcijama se ispituje stanje svakog perifernog uređaja a zatim se upravlja prenosom. Ovim načinom CPU potpuno upravlja prenosom.

Programirani ui može biti uslovni i bezuslovni. Bezuslovni je kada je periferni uređaj uvek spreman za prenos. Uslovni se obavlja pomoću ispitne petlje, programa kojim CPU proverava spremnost perifernog uređaja za prenos. Koristi se kada vreme za prenos nije od ključne važnosti za računar.

Programirani ui ima malu produktivnost, spor je, zahteva izvršavanje velikog broja dodatnih funkcija i procesa (konverzije podataka, formatiranje podataka, kontrola ispravnosti procesa). Programirani ui je osnova za ATA-2 (EIDE standard) interfejs tvrdih diskova.

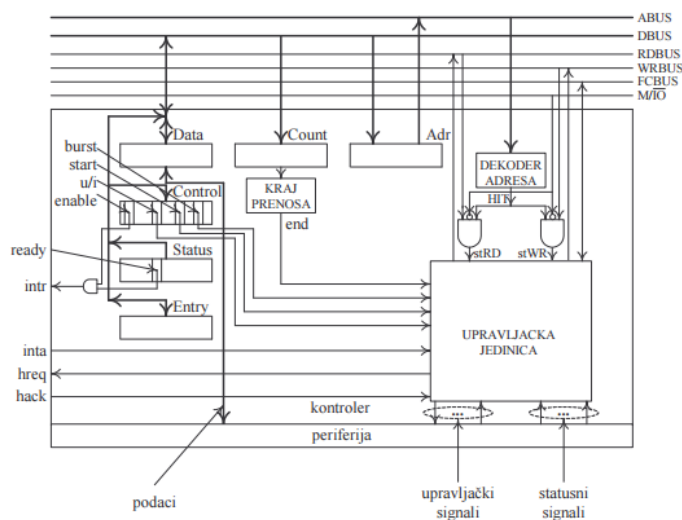
2. UI upravljani prekidom

Ovaj način otkljanja nedostatak programiranog ui načina korišćenja ispitne petlje. CPU prenosi upravljačku informaciju kontroleru perifernog uređaja a zatim se diskonektuje od njega i nastavlja paralelno da izvršava neki drugi program. Kada perifireni uređaj završi operaciju, šalje procesoru zahtev za prekid i registruje stanje perifernog uređaja.

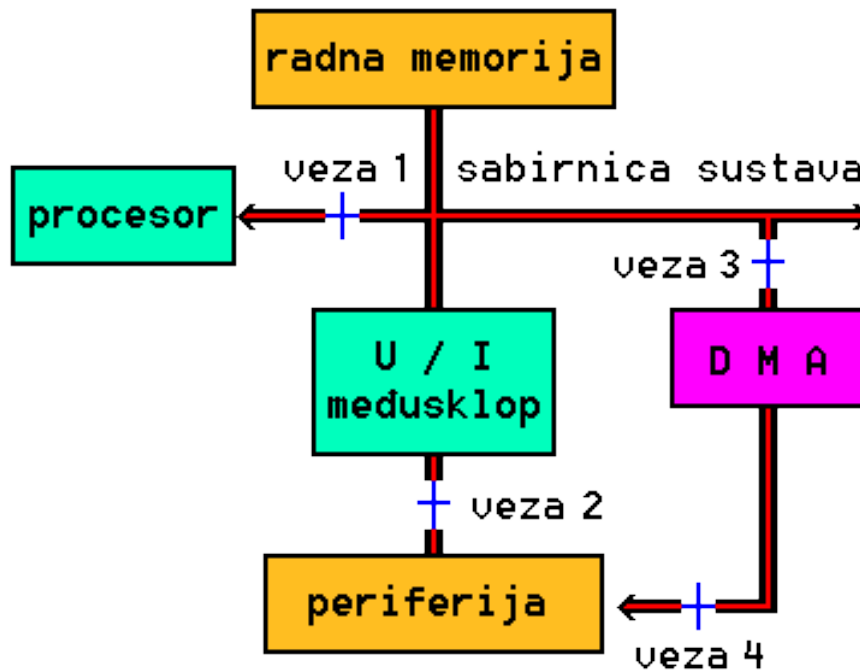
Glavni program	Prekidna rutina
MOV #blockadr, mem1	PUSH R3
MOV #blockcount, mem2	IN Data, R3
MOV #modestart, R3	MOV R1, [mem1]
OUT R3, Control	INC mem1
MOV #1, sem	DEC mem2
! Program u kome se ne koriste	JNZ BACK
! podaci iz bloka memorije u koji	MOV #0, sem
! se unose podaci sa periferije	IN Control, R3
LOOP: CMP sem, #0	AND R3, #modestop
JNZ LOOP	OUT R3, Control
! Program u kome se koriste	BACK: POP R3
! podaci iz bloka memorije u koji	RTI
! su uneti podaci sa periferije	Slika 3 Programirani ulaz generisanjem prekida

3. Direktan pristup memoriji (DMA)

Problemi sa prethodnim načinima prenosa je u smanjenoj efikasnosti korišćenja procesora kada treba izvršiti prenos blokova podataka između većeg broja lokacija glavne memorije i perifernih uređaja. Ovo se rešava uvođenjem paralelnog prenosa u radu CPU i kontrolera periferije. To se postiže tako što više perifernih uređaja ima mogućnost pristupa glavnoj memoriji pomoću DMA sistema.



Slika 4 Kontroler sa direktnim pristupom memoriji



DMA omogućava da brzi ui direktno komuniciraju sa glavnom memorijom ne ometajući rad CPU. Zato mora postojati poseban kontroler DMA kontroler. On ima prenost u korišćenju magistrale čak i u odnosu na CPU.

Komunikacija glavne memorije i nekog DMA kontrolera se obalja po principu pozamljivanja ciklusa od CPU. Zbog sporosti nekih periferiala njima je bitnije dati prvenstvo u pristupu magistrali.

DMA formira poseban ulazni signal za CPU (signal zahteva za korišćenje magistrale) kojim se zahteva direktan pristup memoriji. CPU potvrđuje taj signal, DMA može da koristi magistralu i upravlja prenosom. Na kraju DMA obaveštava CPU da je završio prenos i bloka podataka i CPU može da nastavi prekinut proces i da koristi magistralu.

DMA ima dodatni registar adresa, registar dužine bloka...Ovim načinom se prenose podaci prema ekranima, brzim pretvaračima, između dva CPU...

Ranije za povezivanje tvrdih diskova koristio se PIO uesto DMA jer je tada DMA bio vezan za spore bus što dan nije slučaj.