

## 8 Prekidi (Interrupts)

Za vrijeme izvršavanja programa mogu nastati različite situacije koje zahtijevaju brzo reagovanje od strane CPU, bilo da se na taj način poboljšava ukupna efikasnost računara, bilo da se otklanjaju greške u funkcionisanju. U takvim situacijama potrebno je da se prekine sa izvršavanjem programa i da CPU posveti pažnju uzroku koji je izazvao prekid. Za vrijeme prekida, umjesto programa koji se izvršavao, biće izvršavan poseban program čiji je zadatak da utvrdi razlog prekida i da na odgovarajući način razriješi novonastalu situaciju. Nakon izvršavanja takvog posebnog programa, računar nastavlja rad sa prekinutim programom od mjesta prekida kao da se u međuvremenu ništa nije dogodilo. Izuzetno, u slučajevima kada je prekid izvršen zbog greške u funkcionisanju računara mogu nastati situacije kada se prekinuti program ne može nastaviti, pa se tada kontrola najčešće vraća operativnom sistemu.

U ovom poglavlju biće razmatran sistem za obezbeđenje prekida i tehnike razrješavanja pojave više istovremenih zahtjeva za prekidom.

### 8.1 Razlozi za pojavu interapta

U ovom poglavlju biće razmatrano nekoliko situacija u kojima se zahtijeva prekid rada procesora.

#### 8.1.1 Hardverske greške

Hardverske komponente računara pored elektronike koja obezbjeđuje funkcionalnost obično sadrže i samotestirajuće komponente koje mogu da ustanove greške u svojem radu. Ovdje će biti razmotrena dva primjera: problemi sa napajanjem i problemi sa memorijom.

Računari se najčešće napajaju iz standardne distributivne električne mreže. U slučaju nestanka napona iz mreže, rad procesora mora biti zaustavljen. Bilo bi poželjno da se nakon ponovnog vraćanja napona u mreži rad procesora nastavi kao da se ništa nije dogodilo. U tu svrhu, računari, obično posjeduju posebne elektronske komponente koje vrše nadzor nad ulaznim napajanjem. Kada napon padne ispod određene granice, recimo na 85 % od nominalne vrijednosti, elektronika koja vrši nadzor generiše zahtjev za prekid rada procesora (generiše interapt). Procesor prekida izvršavanja programa da bi posebnim programom sačuvao stanje registara. Nakon ponovnog uspostavljanja normalnog napajanja računar vraća u registre sačuvani sadržaj i normalno nastavlja rad.

Pri transferu podatak u i iz memorije jedan ili više bitova mogu biti pogrešni. Njačešći mehanizam za provjeru tačnosti podataka pri prenosu je tzv. provjera pariteta (parity check).

Kod ovog mehanizma se pored niza bitova koji definišu sam podatak prenosi i dodatni bit (parity bit). Ako je, na primjer, podatak definisan sa parnim brojem bitova postavljenih na 1 poseban bit pariteta ima vrijednost 1, inace ima vrijednost 0. Na taj način se može provjeriti ispravnost transfera podataka u i iz memorije. Ukoliko se ustanovi nepravilnost, generiše se interapt i aktivira poseban program koji može pokušati da ponovnim čitanjem/pisanjem u memoriju dobije tačan podatak. Ako se ni poslije više pokušaja ne dobije ispravan transfer može se javiti poruka o pogrešnom radu memorije što može, dalje zahtijevati intervenciju servisa (za zamjenu memorije, na primjer).

### **8.1.2 Programske greške**

Postoji niz uzroka koji mogu dovesti do neispravnog funkcionisanja programa, koje su rezultat grešaka pri programiranju.

Jedna klasa takvih grešaka nastaje pri aritmetičkim operacijama kada rezultat aritmetičke operacije izlazi van kapaciteta memorijske lokacije (tzv greške prekoračenja - overflow). U takvim situacijama, koje se detektuju testnim bitovima ALU-a (flegovima), generiše se interapt koji najčešće prekida rad programa i kontrolu vraća operativnom sistemu.

Druga klasa programskih grešaka je kada program pogrešno adresira podatke, pa se umjesto stvarne adrese na kojoj se nalazi podatak, dobije adresa instrukcije. Takve situacije, naravno, ne smiju biti dozvoljene pa se posebnim mehanizmom zaštite, utvrđuju takvi pokušaji i generišu interapti.

### **8.1.3 Ulaz/izlaz**

Iako može izgledati da periferni uređaji rade velikom brzinom, oni su zapravo veoma spori u odnosu na rad CPU-a. Sporiji periferni uređaji kao što su štampači i tastature prenose slovne znake jedan po jedan za redom. U cilju omogućavanja da CPU za vrijeme obavljanja ovog sporog transfera, ne čeka već izvršava neke druge operacije, potrebno je da takvi uređaji generisu interapte koji će označavati kraj transfera. U slučaju brzih uređaja kao što su magnetni diskovi, čitav blok podataka može biti transferovan velikom brzinom, pa se interapti generišu samo da označe kraj prenosa bloka podataka. Kada CPU detektuje interapt od strane perifernog uređaja koji je završio jednu operaciju transfera, tada CPU može dati instrukciju za početak transfera novog bloka podataka.

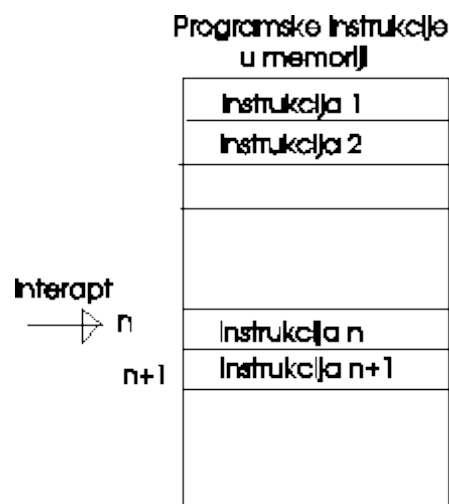
## **8.2 Upravljanje interaptima**

Pretpostavimo da CPU izvršava neki program i da se pri tome pojavi zahtjev od nekog

perifernog uređaja. Tada CPU mora prekinuti izvršavanje instrukcija programa i aktivirati poseban program koji će se baviti zahtjevom perifernog uređaja. Takav poseban program naziva se servisnim interapt programom (interrupt service routine). Nakon završetka rada servisnog programa, CPU treba da nastavi rad na prekinutom programu. U tu svrhu mora postojati mehanizam koji će upravljati procedurom opsluživanja interapta.

S obzirom da je čitava ideja interapt sistema da se poboljša efikasnost rada CPU-a, servisne interapt rutine moraju biti programirane da troše što manje CPU vremena.

Na Slici 8.1 prikazana je memorija sa programom koji će interapt sistemom biti prekinut za vrijeme izvršavanja instrukcije na adresi n.



Slika 8.1 Prekidanje izvršavanja programa (interapt zahtjev)

Proces opsluživanja interapta se odvija u sljedećim koracima:

- (1) CPU najprije završava instrukciju n tokom koje se pojavio zahtjev za prekid.
- (2) CPU zatim vrši:
  - (a) transfer sadržaja PC registra u posebnu memorijsku lokaciju (da sačuva sadržaj PC registra),
  - (b) transferuje adresu servisne interapt rutine u PC registar,
- (3) CPU izvršava instrukcije servisne interapt rutine

Prvim instrukcijama servisne interapt rutine moraju se sačuvati u posebnim memorijskim

lokacijama sadržaji svih registrara.

Prije završetka rada servisne rutine, vraća se sadržaj registara. Na kraju se u PC registar vrati adresa sačuvana u koraku (1) (a). Tako se obezbjeđuje normalan nastavak rada prekinutog programa.

Moguće je da nastupe slučajevi kada se u istom trenutku (ili za vrijeme izvršavanja servisne interupt rutine) pojavi novi interapt. Tada se predhodni mehanizam proširuje uz uvođenje prioriteta koji pokazuju da li novi interapt može prekinuti već izvršavanu servisnu interapt rutinu ili ne. Prioriteti se određuju na osnovu važnosti različitih vrsta interapta.