

Zadatak 3.3. (ispitni zadatak)

U nekom sustavu javljaju se zahtjevi za prekid: P1 u 3. ms, P2 u 1. ms te P3 u 4. ms. Prioritet prekida određen je brojem (P3 ima najveći prioritet). Obrada svakog prekida traje po 4 ms. Grafički prikazati aktivnosti procesora u glavnom programu (GP), procedurama za obradu prekida (Pi) te procedurama za prihvrat prekida (PP) i povratak iz prekida (PiP) i to:

- a) u idealnom slučaju (*prekidi se obrađuju prema prioritetu, trajanje kućanskih poslova se zanemaruje*)
- b) bez sklopa za prihvrat prekida (*u sustavu koji nema sklop za prihvrat prekida u kojem se po prihvatu nekog prekida on obrađuje do kraja – nema ni programske ni sklopovske potpore za obradu prekida prema prioritetima*), uz trajanje prihvata prekida (PP) od 1 ms (*uključuje potragu za izvorom prekida – zahtjevi većeg prioriteta se prvi prihvataju*) te trajanje povratka iz prekida (PiP) od 0,5 ms (*obnova konteksta prekinute dretve*)
- c) bez sklopa ali s programskom potporom (*u sustavu koji nema sklop za prihvrat prekida, ali se programski određuje prioritet prekida – obrada prekida se odvija s dozvoljenim prekidanjem*), uz trajanje procedure za prihvrat prekida i određivanje prioriteta prekida od 1,5 ms (PP), te 1 ms za povratak iz prekida (PiP) (*na dovršetku obrade prekida ponovno treba pogledati ima li još nešto za obraditi*)
- d) sa sklopom za prihvrat prekida uz vrijeme prihvata prekida (*pohrana konteksta prekinute dretve*) od 0,5 ms (PP) te vrijeme povratka iz prekida (*obnove konteksta*) od 0,5 ms (PiP)

U trenutku t_x prikazati stanje korištene podatkovne strukture (za c) i d)).

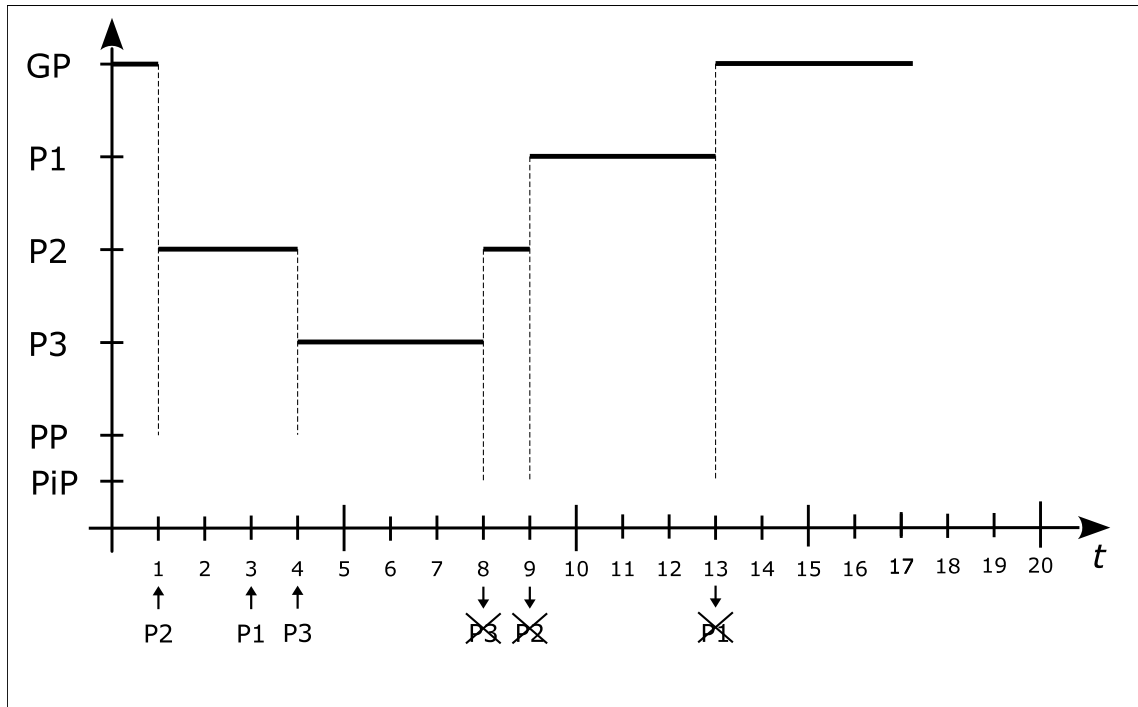
Dodatna pravila za rješavanje:

- ako u istom trenutku neka obrada završava i pojavljuje se novi zahtjev za prekid pretpostavljamo da je ipak najprije završila obrada, a potom se dogodio zahtjev za prekid
- ako u istom trenutku više naprava generira zahtjev za prekid najprije se prihvata onaj najvećeg prioriteta – ostale i dalje imaju postavljen zahtjev za prekid (na prekidnom ulazu procesora ili sklopa za prihvrat prekida)

Rješenje (par stranica)

a) [u idealnom slučaju]

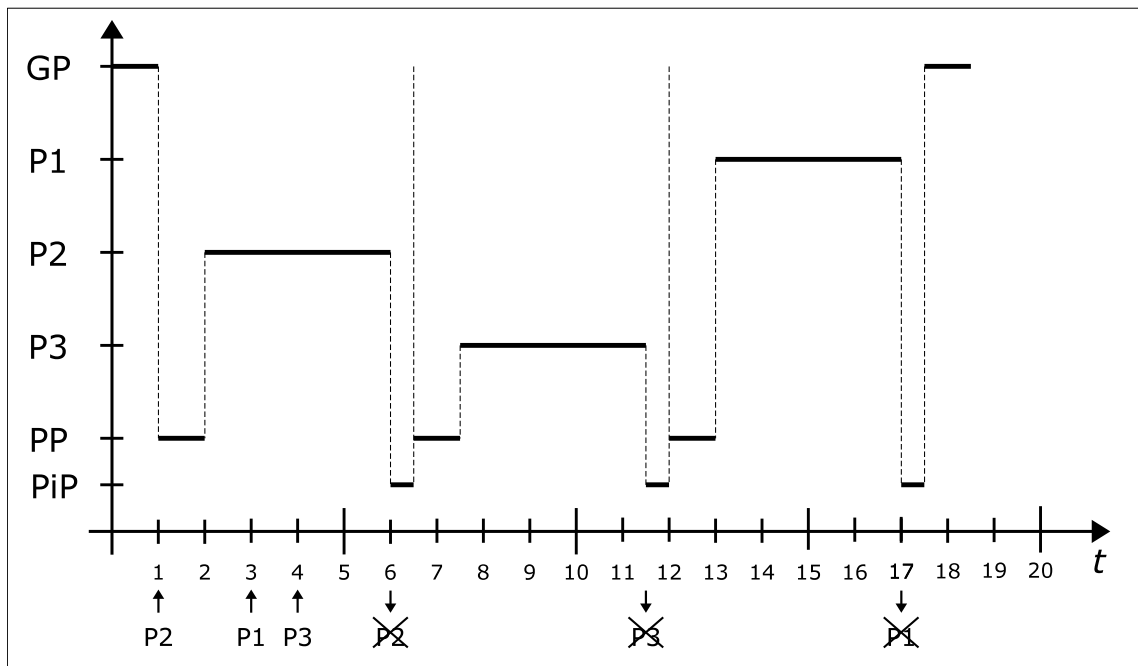
U idealnom slučaju zanemarujemo trajanje kućanskih poslova, a prekidi se obrađuju prema prioritetu: zahtjev veće prioriteta odmah prekida obradu zahtjeva manjeg prioriteta.



Prekriženi broj prekida označava da je taj prekid u tom trenutku obrađen do kraja (gotovo je s njegovom obradom).

PP i PiP i nije neophodno prikazivati na grafu, tj. crta ide izravno do GP do P2 (tako i kasnije).

b) [bez sklopa za prihvatanje prekida, obrada uz zabranjeno prekidanje]



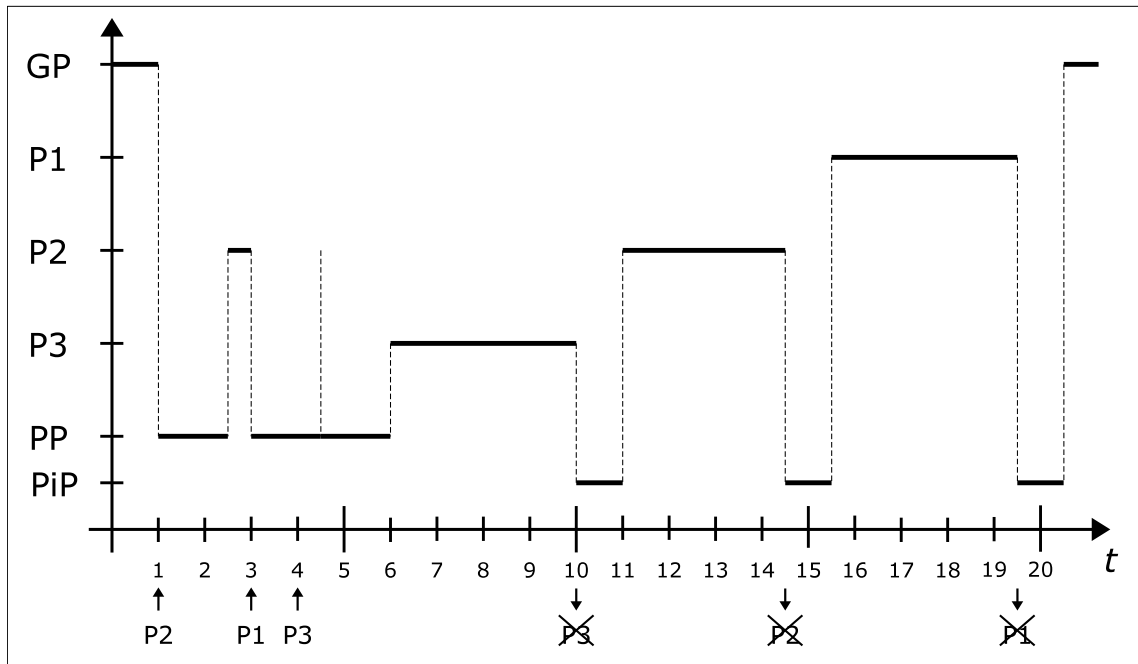
PP = Pohrana konteksta + prozivanje = 1ms

PiP = Obnova konteksta = 0,5 ms

Zanimljivi trenutci na grafu:

- 1. ms: pojava zahtjeva za prekid P2 te prihvatanje prema opisu u 3.3.1. (procesor + prekidni potprogram) do "obrade prekida naprave I"
- 2. ms: početak obrade prekida naprave ("obradi prekid naprave I")
- 3. i 4. ms: javlja se P1 ali se ne prihvaća (prekidi su tada zabranjeni i ostaju tako do završetka P2)
- 6. ms: završava P2 te se obnavlja kontekst prekinute dretve = glavne dretve. Zadnjom instrukcijom (vrati_se_u_prekinutu_dretvu) dozvoljava se prekidanje te se tada propušta i prihvaća P3. Iako je prije došao zahtjev od P1 s obzirom na to da imamo zajednički prekidni signal (koji ide od svih naprava do procesora) procesor "proziva" i pritom prvo vidi prioritetniji zahtjev => P3.

c) [bez sklopa ali s programskom potporom]



PP = Pohrana konteksta na stog + prozivanje + premještanje konteksta u KON[i] = 1,5 ms

PiP = Obnova konteksta na stog + pregled što dalje + (obnova ili premještanje konteksta) = 1 ms

Zanimljivi trenutci na grafu:

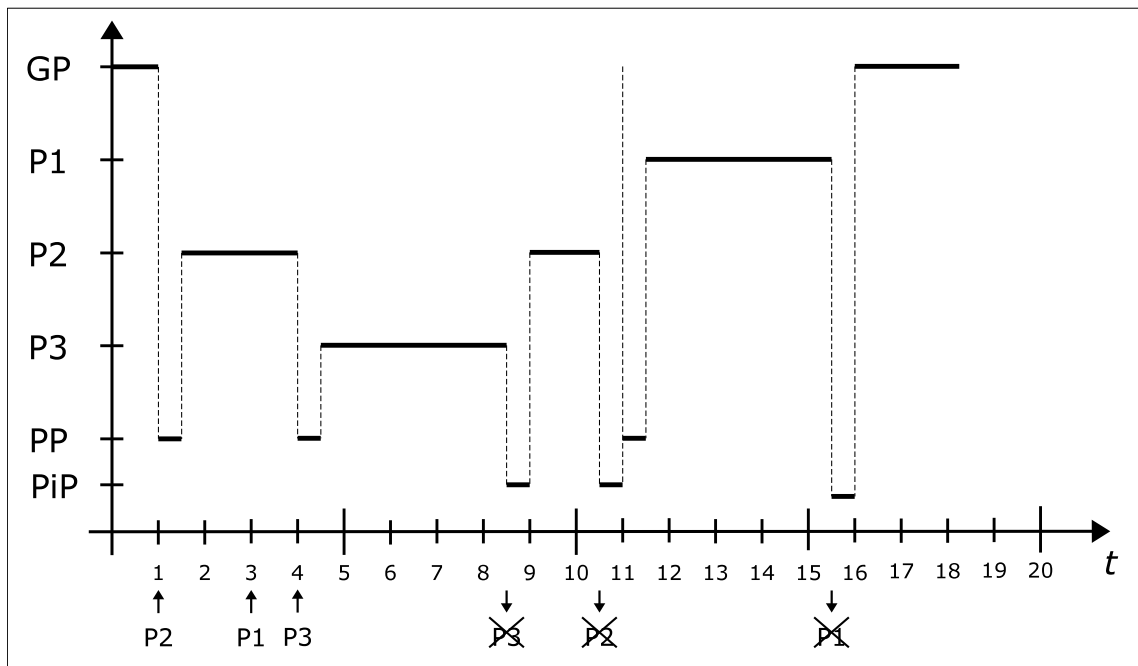
- 1. ms: pojava zahtjeva za prekid P2 te prihvata prema opisu u 3.3.2. (procesor + prekidni potprogram) do "obrada prekida naprave I"
- 3. ms: P1 izaziva prekid, poziva se PP (uz zabranu prihvata daljnjih prekida). U PP se zaključuje da P1 nema veći prioritet od P2 (trenutnog posla) te da se tamo treba vratiti.
- 4,5 ms: Po okončanju kućanskih poslova od prekida (tj. neposredno prije povratka u obradu P2) opet se dozvoljava prekidanje ("vratiti se u prekinutu dretvu") te se zahtjev za prekid od P3 prihvaća (koji čeka od 4. ms). Stoga još 1,5 ms na kućanske poslove prije početka obrade P3 (najprioritetnijeg zahtjeva).

Stanje strukture podataka u t=7. ms:

- TEKUCI_PRIORITET = 3
- OZNAKA_CEKANJA[] = 0 0 1
- KON[1] = -
- KON[2] = {TP=0, kontekst(GP)}
- KON[3] = {TP=2, kontekst(P2)}

Prioritetniji prekid (P3) prekida obradu prekida manjeg prioriteta. Prioritetniji prekidi prije dolaze na red, ali to "plaćamo" više (kućanski poslovi duže traju i ponekad se prekidaju značajnije obrade od novih zahtjeva). U ovom zadatku to nije ispalo puno prije nego u a) ali općenito je ovaj pristup daleko bolji.

d) [sa sklopom za prihvata prekida]



Prihvata prekida (PP) i povratak iz prekida (PP) jednako traju jer se te operacije svode na pohranu konteksta prekinute dretve (ili obrade manjeg prioriteta) te obnovu konteksta (na kraju obrade prekida)

Zanimljivi trenutci na grafu:

- 3. ms: P1 ne prekida obradu P2
- 4. ms: P3 prekida obradu P2 jer je prioritetniji
- 7,5 ms: P3 završava obnavlja se kontekst od P2 koji još nije gotov

Stanje strukture podataka u t=7. ms (prema kodu za prihvat prekida u skripti!):

- registar TP = 1 1 0
- registar KZ = 0 0 1
- stog (vrh prema dnu): kontekst(P2), kontekst(GP)

U ovakvim sustavima ima manje kućanskih poslova jer nam sklop pomaže. Zato i PP traje kraće nego u b) slučaju, tj. sustav je efikasniji od b) i stoga što ne propušta manje prioritetnije zahtjeve do procesora dok on radi prioritetnije obrade.

Općenito:

U ovom zadatku se koriste mnogo veća vremena za kućanske poslove pri prihvatu prekida nego li je to u stvarnim sustavima u odnosu na trajanja obrada kako bi trajanje obrade prekida mogli lakše prikazati na slikama.