



Univerzite u Novom Sadu

Fakultet Tehničkih Nauka

Katedra za računarsku tehniku i međuračunarske komunikacije



Algoritmi i arhitekture DSP I

- ❖ OPERATIVNI SISTEMI DIGITALNIH SIGNAL PROCESORA



OPERATIVNI SISTEMU DSP-a

- ❖ Operativni sistem je program koji kontrolira pristup resursima sistema i rukuje redosledom izvršenja posebnih programa (procesa) na procesoru
- ❖ Proces je program pripremljen za izvršavanje
- ❖ Glavni resursi operativnog sistema:
 - ❖ Procesor
 - ❖ Memorija
 - ❖ Ulazno-izlazni podsistem
 - ❖ Informacije (podaci i programi)



FUNKCIJE OPERATIVNOG SISTEMA U REALNOM VREMENU

- ❖ Formiranje i odstranjivanje procesa
- ❖ Upravljanje tokom realizacije procesa
- ❖ Delovanje u uslovima izuzetnih događaja
- ❖ Raspodela fizičkih resursa među procesima
- ❖ Omogućavanje pristupa programskim resursima
- ❖ Obezbeđenje zaštitnog mehanizma, pristup informacijama
- ❖ Obezbeđenje komunikacije i sinhronizacije među procesima



OPERATIVNI SISTEMI-PODELA

- ❖ Opšte namenski operativni sistemi
 - ❖ Obrada programskih paketa i rad sa podelom vremena
- ❖ Operativni sistemi za rad u realnom vremenu
 - ❖ Reagovanje na spoljne događaje u unapred definisanom vremenskom intervalu
 - ❖ Vreme odziva se meri ms
- ❖ Operativni sistemi orjentisani na transakcionu obradu
 - ❖ Održavanje deljivih baza podataka



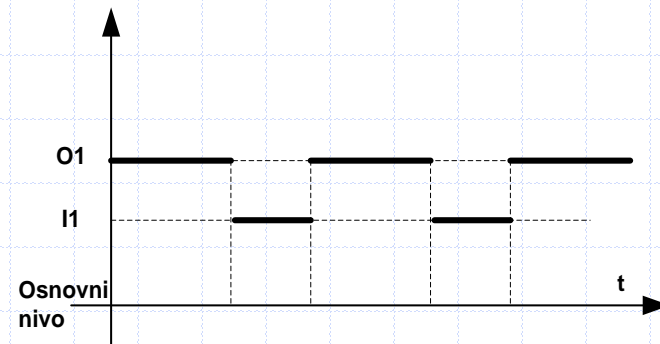
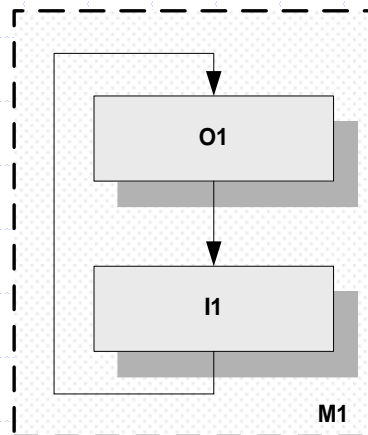
Multiprogramski režimi rada

- ❖ Multiprogramiranje bez vremenskih prekida
- ❖ Multiprogramiranje sa vremenskim prekidima
- ❖ Multiprogramiranje sa vremenskim prekidima i programskim prioritetima na bazi podele vremena



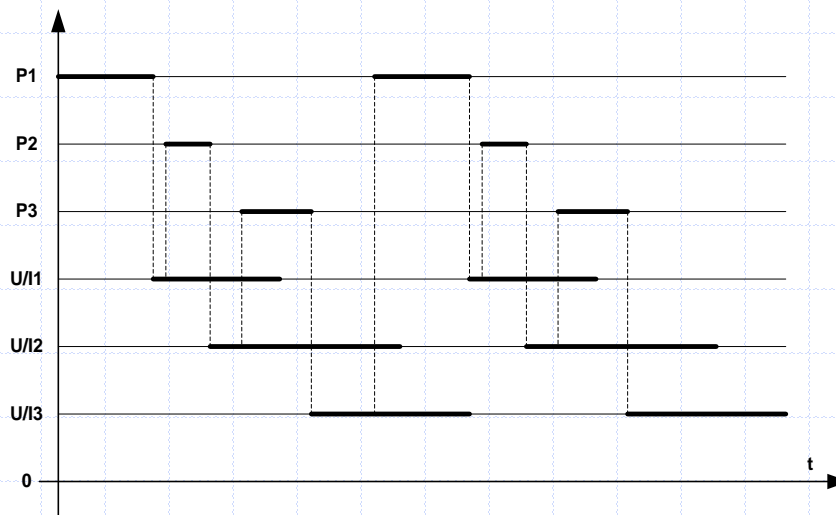
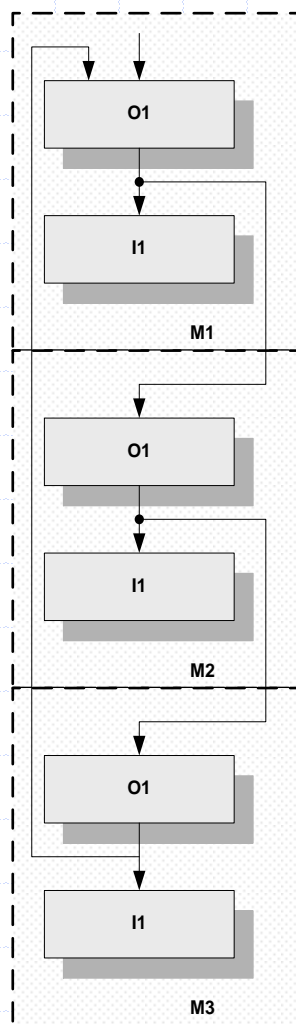
Multiprogramiranje bez vremenskih prekida

- ❖ Dobro za programiranje koje nema "tvrde" vremenske zahteve
- ❖ Osnovna ideja smanjenje praznog hoda CPU (za vreme U/I je neaktivan)
- ❖ Velika iskoristivost pojedinih funkcija
- ❖ Po prekidu rada jednog programa (U/I aktivnost), predaje se upravljanje upravljačkom programu (rukovaocu U/I) a potom drugom programu koji čeka





Multiprogramiranje bez vremenskih prekida





Multiprogramiranje sa vremenskim prekidom

- ❖ Osnovni cilj poboljšanje kvaliteta usluživanja korisnika a ne maksimalna iskorišćenost resursa!
- ❖ Ukupno vreme procesora se deli među programima
- ❖ Vremenski prekid po isteku dodeljenog intervala
- ❖ Svi programi imaju isti prioritet
- ❖ Smanjeno vreme odziva u odnosu na sekvencijalni pristup
- ❖ Ne uzima se u obzir čekanje na neki program UI



Multiprogramiranje sa vremenskim prekidom i dodelom prioriteta programima

- ❖ Programi se izvršavaju redom, u toku vremenskih intervala po čijem završetku dolazi do prekida
- ❖ Dužina intervala nije konstanta, zavisi od programa i uslova u momentu prenosa upravljanja
- ❖ Upravljački program na osnovu sistema prioriteta dodeljuje vremenski interval i upravlja prenosom
- ❖ Računarski sistem trenutno reaguje na zahteve za realno vreme usluživanja
- ❖ Minimizuje se vreme odziva



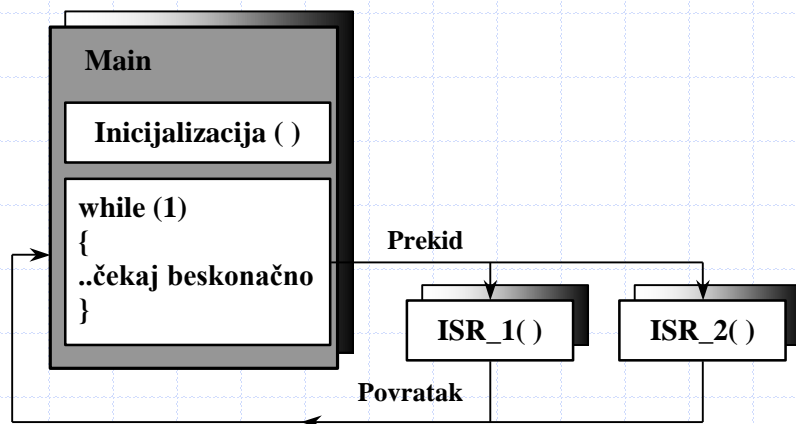
Operativni sistemi za rad u realnom vremenu

- ❖ *Garantuje odziv u skladu sa zahtevima realnog vremena*
- ❖ *Realno vreme: maksimalna količina vremena između pojave signala prekida i pokretanja nekog procesa*
- ❖ *Ulazne događaje generišu ulazne jedinice, a ne čovek. Uniformno raspoređeni, moraju se uneti i opslužiti sistemom unutar unapred definisanog intervala vremena*
- ❖ *Procesi mali, organizovani po prioritetima*
- ❖ *Komunikacija preko deljivih segmenata operativne mem.*
- ❖ *Operativni sistem u realnom vremenu i aplikacija nisu nezavisni!*



RADNA OKRUŽENJA ZA REALNO VREME

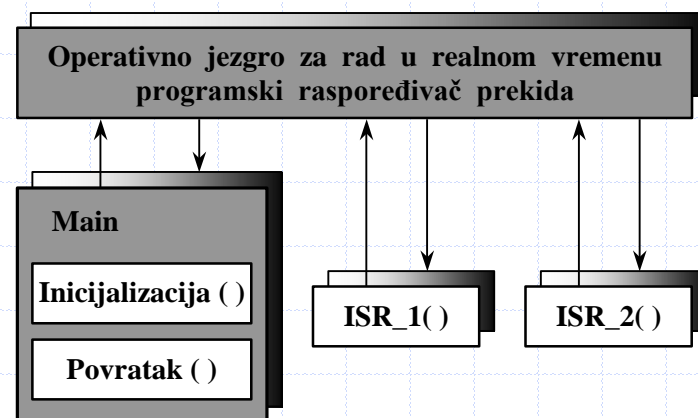
- ❖ Smeštanje operativnog koda unutar rutina za posluživanje prekida (ISR)
- ❖ Problemi:
 - ❖ Teško praćenje događaja redosledom kojim su se dogodili.
 - ❖ Kritične sekcije
 - ❖ Teško profilisanje
 - ❖ Teško "debugovanje"
 - ❖ Teško raspoređivanje (scheduling) zadataka





OPERATIVNI SISTEMI ZA RAD U REALNOM VREMENU

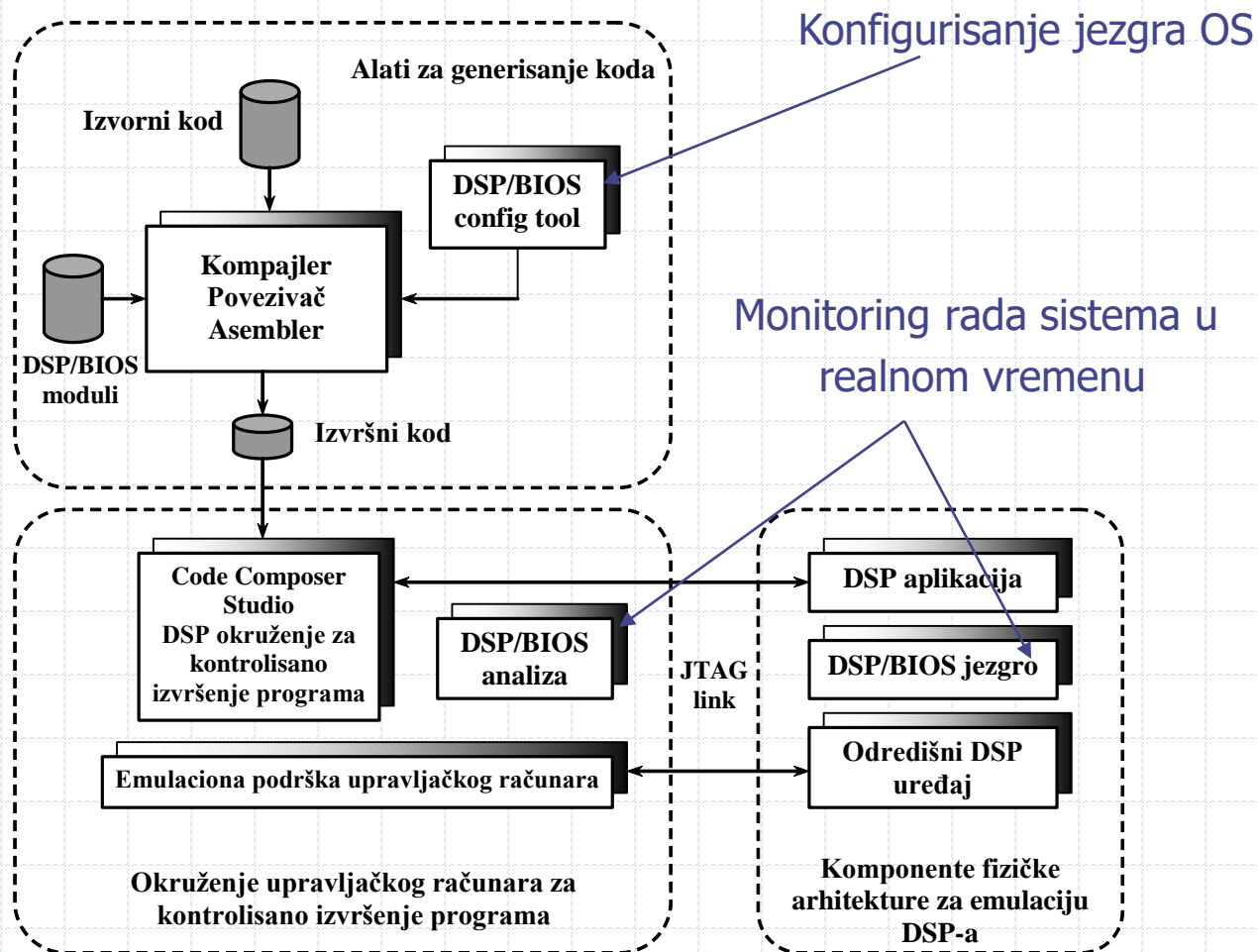
- ❖ Engleski nazivi:
 - ❖ Real-time operating systems
 - ❖ Real-time kernels
 - ❖ Interrupt schedulers



- ❖ Osnovna ideja:
 - ❖ Raspoređivač obezbeđuje sloj između stvarnih fizičkih prekida i korisničke app. pp



PRIMER RADNOG OKRUŽENJA ZA RAD U REALNOM VREMENU DSP/BIOSII





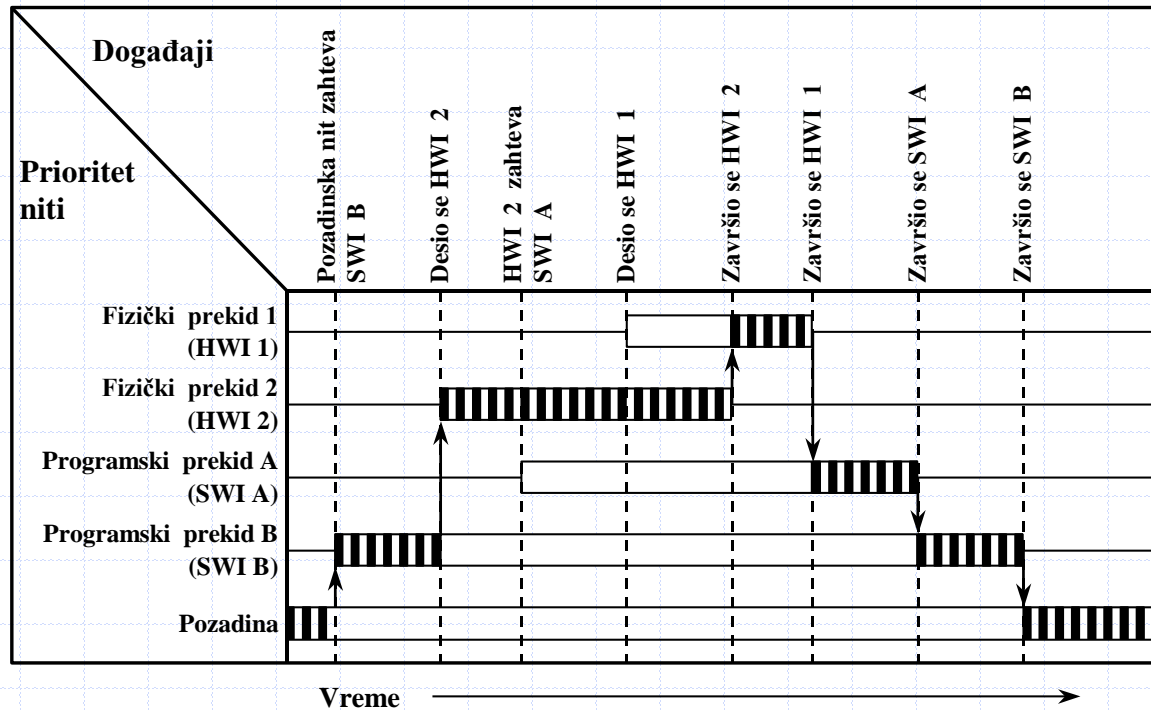
KORACI U STARTOVANJU DSP/BIOSII

- ❖ Korak 1
 - ❖ Inicijalizuj DSP. Aplikacioni programi započinju izvršenje u ulaznoj tački u C okruženje. C_int00, vektor za reset se postavlja na c_int00 nakon reseta
- ❖ Korak 2
 - ❖ Poziv BIOS_init, radi inicijalizacije modula radnog okruženja(vektori prekida, init modula,komunikacioni kanali ka hostu)
- ❖ Korak 3
 - ❖ Poziv main()
- ❖ Korak 4
 - ❖ Poziv BIOS_start(pokretanje radnog okruženja)
- ❖ Korak 5
 - ❖ Ulaz u besposlenu petlju (IDL_loop)



NITI (THREAD) U DSP/BIOSII

- ❖ Pozadinska ili besposlena nit, IDL
- ❖ Programski prekid, SWI
- ❖ Hardverski prekid, HWI
- ❖ Istiskivanje i predavanje niti (preemption and yielding)





DSP/BIOSII JEZGRO

- ❖ 200-2K reči
- ❖ Više niti izvršavanja programa
- ❖ Rukovanje U/I aktivnostima
- ❖ Operativna statistika
- ❖ Širok niz alata za analizu zasnovan na upravljačkom računaru



VIZUALIZACIJA I RAZVOJ ALGORITAMA

- ❖ Do sada obrađeno projektovanje DSP aplikacije sa stanovišta ARHITEKTURE
- ❖ Važna polovina projektovanja DSP aplikacije je ALGORITMIKA
- ❖ Postoje alati koji obezbeđuju simulaciju algoritma na nivou blok-dijagrama
 - ❖ MATLAB
 - ❖ MATHEMATICA
- ❖ Simulacija nepokretnog zarez, projektovanje filtera, generisanje koda



PREDNOST KORIŠĆENJA ALATA ZA VIZUELIZACIJU I RAZVOJ ALGORITAMA

- ❖ Lakši i brži razvoj algoritama na nivou blok dijagrama
- ❖ Mogućnost simulacije efekata kvantizacije, propagacije greške
- ❖ Automatsko generisanje C koda za DSP aplikacije
- ❖ Interakcija u realnom vremenu sa DSPom



MATLAB

- ❖ MATLAB je računarski jezik koji integriše proračune, vizualizaciju i programiranje u okruženje jednostavno za upotrebu, u kom se problemi i rešenja izražavaju matematičkim zapisima
 - ❖ MATLAB jezik
 - ❖ Radno okruženje MATLAB-a
 - ❖ Rukovalac grafikom
 - ❖ MATLAB API (C/C++, FORTRAN)



MATLAB-SIMULINK

- ❖ Interaktivni sistem za simuliranje nelinearnih dinamičkih sistema
- ❖ Postojeća infrastruktura za razvoj DSP aplikacija
 - ❖ DSP blockset
 - ❖ "Fixed-Point" Blockset
 - ❖ "Real-Time Workshop"