

---

Elektrotehnički fakultet u Beogradu  
Katedra za računarsku tehniku i informatiku

*Predmet:* Programiranje u realnom vremenu (MS1PRV)  
*Nastavnik:* Prof. dr Dragan Milićev  
*Asistent:* Bojan Furlan  
*Školska godina:* 2011/2012.

# Projekat za domaći rad

## - Projektni zadatak –

**Važne napomene:** Pre čitanja ovog teksta, **obavezno** pročitati opšta pravila predmeta i pravila vezana za izradu domaćih zadataka! Pročitati potom ovaj tekst **u celini i pažljivo**, pre započinjanja realizacije ili traženja pomoći. Ukoliko u zadatku nešto nije dovoljno precizno definisano ili su postavljeni kontradiktorni zahtevi, student treba da uvede razumne pretpostavke, da ih temeljno obrazloži i da nastavi da izgrađuje preostali deo svog rešenja na temeljima uvedenih pretpostavki. Zahtevi su namerno nedovoljno detaljni, jer se od studenata očekuje kreativnost i profesionalni pristup u rešavanju praktičnih problema!

# Zadaci

Projekat koji izrađuje svaki student sastoji se iz zadataka opisanih u nastavku. Student može da bira koliko i koje od ovih zadataka će izraditi. Da bi položio ispit, student mora da sakupi najmanje 10 poena izradom i odbranom svog projekta.

Štampane materijale pripremiti prema uputstvima datim u zadacima, a sve zajedno na sledeći način:

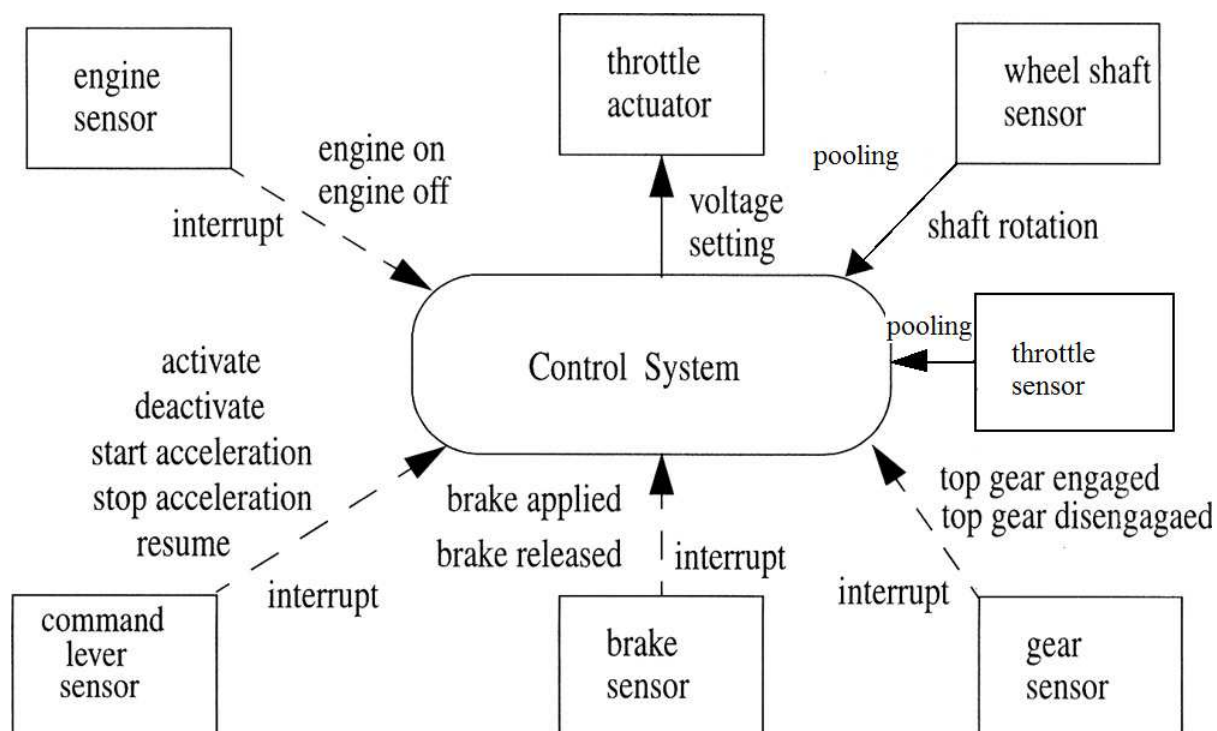
1. Na naslovnoj strani jasno napisati naziv predmeta, prezime i ime studenta, broj indeksa i adresu e-pošte.
2. Svaki zadatak početi jasno istaknutim naslovom sa oznakom tog zadatka.
3. Sve zajedno čvrsto povezati u jednu celinu, tako da se listovi ne mogu rasipati (najbolje spiralom).

## Sistem za automatsku regulaciju brzine automobila

### Opis Sistema

Sistem za automatsku regulaciju brzine automobila (*Automobile Cruise Control System - ACCS*) treba da automatski održava brzinu vozila zadatu od strane vozača.

Relacije između upravljačkog sistema i spoljnih uređaja prikazane su na sledećoj slici.



## Funkcionalni zahtevi

Sistem može primati komande eksplicitno ili implicitno. Eksplicitne komande se mogu zadati preko kontrolne table, tj. ručice koja se nalazi na vratu volana. Ručica ima nekoliko položaja koje predstavljaju sledeće instrukcije sistemu (komande su jedino validne kada je motor upaljen):

- **activate** – uključen ACCS ukoliko je automobil u najvišoj brzini; održava (i pamti) trenutnu brzinu,
- **deactivate** – isključen ACCS,
- **start accelerating** – ubrzava do određene brzine,
- **stop accelerating** – zaustavlja ubrzavanje i održava (i pamti) trenutnu brzinu,
- **resume** – vraća brzinu vozila na poslednju zapamćenu brzinu i održava tu brzinu.

Implicitne komande se zadaju kada vozač:

- *promeni brzinu* – ACCS se deaktivira kada vozač prebaci iz najviše brzine u neku nižu brzinu,
- *pritisne kočnicu* – ACCS se deaktivira kada vozač koči.

Brzina automobila se meri pomoću brzine rotacije zadnje osovine. Osovina pri svakoj rotaciji generiše određeni napon srazmeran brzini rotacije koji se može očitati pomoću A/D konvertora uređaja.

Pogon motora vozila zavisi od pozicije poklopca (leptir ventila) na grlu karburatora. Pozicija se određuje na osnovu dva faktora: pritiska na pedal za gas i vrednosti dostavljene od strane ACCS-a. Komponenta kojom utiče ACCS se zadaje pomoću promenljivog napona na aktuatoru gasa. Vrednosti su u opsegu od 0V (poklopac skroz zatvoren) do 8V (poklopac skroz otvoren). Napon se može zadavati u jedinicama od po 0,1V, pri čemu se zadata vrednost zadržava 1 sekundu. U koliko nije zadata, podrazumevana vrednost je 0V. Pretpostavlja se da kombinovanje ove dve komponente koje utiču na položaj poklopca na grlu karburatora se obavlja izvan ACCS sistema.

Pored zadatih funkcionalnih zahteva sistem mora ispunjavati i dva uslova bezbednog kretanja: „održavanje trenutne brzine“ i „komforno ubrzavanje“.

### *Održavanje trenutne brzine*

Na brzinu kojom se vozilo kreće, pored pogona motora, tj. pozicije leptir ventila, utiču i različiti spoljni faktori kao što su nagib puta (nizbrdica, uzbrdica) i trenutni otpor vetra. Iz tog razloga za istu vrednost zadatog napona mogu se dobiti različite očitane vrednosti brzine kretanja vozila.

Zadavanje napona na leptir ventilu se vrši na sledeći način:

- Ako je razlika željene brzine i trenutne brzina veća od 2km/h, postaviti poklopac karburatora sasvim otvoren.
- Ako je razlika trenutne brzine i željene brzine veća od 2km/h, postaviti poklopac karburatora sasvim zatvoren.
- U suprotnom, vrednost napona se izračunava pomoću formule:

$$V = \begin{cases} 8 & (S_D - S_A) > 2 \\ 2(S_D - S_A + 2) & -2 \leq (S_D - S_A) \leq 2 \\ 0 & -2 > (S_D - S_A) \end{cases}$$

Gde  $V$  predstavlja zadati napon, a  $S_D$  i  $S_A$  željenu i trenutnu brzinu respektivno.

*Komforno ubrzanje.* Radi izbegavanja naglih promena brzine, napon se ne sme menjati za više od 0.8V u toku perioda od jedne sekunde.

Napomena: Iz ovog zahteva sledi da je potrebno postepeno pomerati poziciju poklopca (ne uzimati u obzir samo formulu u datom trenutku).

### Vremenski zahtevi

- Svi uređaji koji očitavaju neku fizičku veličinu iz okruženja pomoću senzora (npr. A/D konvertor) funkcionišu po principu da procesor prvo mora da zada, tj. upiše određenu vrednosti u upravljački (kontrolni) registar uređaja, a nakon toga se vrši potrebno očitavanje, tj. uzimanje odbirka sa analognog senzora i početak njegove konverzije u digitalni oblik, što zahteva određeno vreme. Kada se završi konverzija i pripremi podatak za očitavanje u svom registru za podatke, uređaj postavlja odgovarajući indikator u svom statusnom registru. Taj podatak za očitavanje može biti spreman tek posle određenog vremena, a o spremnosti tog podatka procesor (tj. aplikacija) može biti obaveštena puštanjem da protekne odgovarajuće vreme za koje će ova vrednost sigurno biti raspoloživa, odnosno za koje će konverzija biti završena. U slučaju da konverzija nije uspela ili je došlo do bilo kakve greške, podaci o grešci se mogu očitati iz statusnog registra uređaja.
- Implementirati sve procese koji vrše preriđična očitavanja senzora tako da ne pate od kumulativnog plivanja (engl. *cumulative drift*), a samo očitavanje realizovati tehnikom pomeranja periode (engl. *period displacement*).

### Pretpostavke

- Svi uređaji su memorijski mapirani i imaju pridružen kontrolni i registar podataka. Senzori u vozilu detektuju promene stanja i generišu odgovarajuće prekide kontrolnom sistemu. Za svaki prekid u registru podataka određenog uređaja može se očitati događaj koji je prouzrokovao ovaj prekid (npr. *command lever senzor* u svom *data* registru može sadržati vrednosti koda *activate*, *deactivate*, *start accelerating*, *stop accelerating*, *resume*).
- Vrednost napona na aktuatoru gasa može imati vrednosti u opsegu od 0 do 80. Ovo predstavlja napon u jedinicama od 0,1V.
- Vrednost napona generisanog pri rotaciji zadnje osovine srazmeran je brzini vozila.
- Proces koji kontroliše položaj poklopca karburatora ima ulogu da periodično proverava da li zaista poklopac menja položaj u zavisnosti od zadate vrednosti. Zbog inertnosti fizičkog sistema (otvaranja i zatvaranja poklopca karburatora), procesu koji kontroliše daje se perioda od 200 ms, pri čemu će on koristiti dva uzastopna očitavanja da bi utvrdio stvarni položaj. Da bi se obezbedilo da su dva uzastopna očitavanja zaista na zadatom razmaku ovom procesu je potrebno zadati tesan rok od 30 ms. Takođe, po istom principu i sa istim vremenskim parametrima proces koji je zadužen za očitavanje

brzine vozila treba da periodično očitava vrednost napona srazmernog brzini kretanja vozila.

Napomena: U slučaju očitavanja nekorektne vrednosti treba izvršiti oporavak od kvara. Kvar može biti prouzrokovan iz više razloga, npr. zaglavljivanjem mehanizma koji pomera poklopac.

- Pretpostavlja se da su svi ostali procesi (osim merenja brzine i određivanja položaja leptir ventila) pokretani događajem (engl. *event-driven*), tj. sporadični i da sistem treba da na njih reaguje u roku od 200 ms. Fizika sistema ukazuje da će proći bar 500 ms između dva ovakva događaja.
- Vreme koje se sme dozvoliti od trenutka aktiviranja nekog događaja koji treba da deaktivira sistem do njegovog gašenja je 200 ms.
- Senzori će sigurno završiti svoje očitavanje i konverziju za 50 ms od trenutka zadavanja konverzije.

## Zadatak 1 – RT UML model (10 poena)

Pomoću alata Rational Rose Real Time (Rational Rose Technical Developer) modelovati dati sistem.

### *Proizvodi*

Za usmenu odbranu potrebno je uraditi i pripremiti u elektronskoj formi sledeće:

1. Detaljan RT UML model sa proizvoljnim programskim jezikom na nivou detalja. Može se koristiti proizvoljna RT platforma (operativni sistem ili izvršno okruženje). Potrebno je doneti kompletan projekat.
2. Detaljno dokumentovati dati model. Naglasiti sve specifičnosti.
3. Demonstrirati realizovani sistem prikazom i simulacijom odgovarajućeg test-modela. Test-model treba da demonstrira sve realizovane funkcionalnosti i njihove osnovne i specijalne slučajeve pomoću odgovarajućih pobuda (*probes*).

U štampanoj formi na prvi deo ispita treba doneti kompletan model i dokumentaciju.

## Zadatak 2 – Implementacija (10 poena)

Potrebno je u potpunosti implementirati gore navedeni softverski sistem.

Sistem treba da bude u potpunosti realizovan prema zahtevima i preporukama datim na predavanjima i vežbama. Sve parametre sistema (npr. kritične vrednosti veličina koje se mere) treba definisati kao simboličke konstante i lokalizovati ih u programu. Slično, sve interakcije sa ulazno/izlaznim uređajima treba lokalizovati u posebne procedure koje će simulirati rad sa stvarnim hardverskim uređajima realnog sistema.

Izgled korisničkog interfejsa i nivo njegove informativnosti i intuitivnosti treba da odredi student. U svakom slučaju, korisnički interfejs mora da bude interaktivan, tako da obezbedi:

- ručno upravljanje aktuatorima.
- interaktivnu simulaciju sa **moгуćnošću zadavanja vremenskih karakteristika i vanrednih situacija** (npr. zadavanje brzine vozila, otkaza nekog aktuatora, itd. )

### ***Proizvodi***

Za usmenu odbranu potrebno je uraditi i pripremiti u elektronskoj formi sledeće:

1. Implementirati softverski sistem na proizvoljnom programskom jeziku. Može se koristiti proizvoljna RT platforma (operativni sistem ili izvršno okruženje). Takođe, može se koristiti bilo koje programsko razvojno okruženje. Potrebno je doneti kompletan izvorni kod.
2. Detaljno dokumentovati dati softverski sistem. Naglasiti upotrebljene projektne uzorke.
3. Demonstrirati realizovani sistem prikazom i simulacijom odgovarajućeg test-okruženja. Test-okruženje treba da bude interaktivno, što znači da se u toku rada programa ručno mogu zadavati ulazni parametri i pobude sa njihovim vremenskim karakteristikama. Način zadavanja ulaza i prikazivanja izlaza izabrati po želji.

U štampanoj formi na prvi deo ispita treba doneti kompletnu dokumentaciju, a u elektronskoj formi(CD/DVD) izvorni kod.

## **Zadatak 3 – Proračun rasporedivosti (10 poena)**

**Napomena:** Ovaj zadatak može se raditi samo ako je prethodno urađen Zadatak 2.

Potrebno je izvršiti procenu vremena izvršavanja u najgorem slučaju za procese u sistemu realizovanom u Zadatku 2 na osnovu testa rasporedivosti za FPS zasnovanog na vremenu odziva (engl. *response time*). Takođe, na osnovu uvedenih pretpostavki potrebno je odrediti period  $T$  i vremenski rok  $D$  za svaki proces.

Vreme izvršavanja u najgorem slučaju svakog od identifikovanih procesa treba što preciznije izmeriti na realizovanoj platformi.

### ***Proizvodi***

Za usmenu odbranu treba pripremiti u štampanoj formi i na prvi deo ispita doneti kratak izveštaj o načinu izvođenja analize i njenim rezultatima (najviše na tri strane).