

## Projektni zadatak iz kolegija Sustavi za vođenje proizvodnih procesa

Projektni zadatak potrebno je odraditi u timu prema uputama danim ovdje. Uz projektni zadatak potrebno je napisati i dokumentaciju projekta te napraviti prezentaciju. U zadnjim tjednima nastave bit će održane prezentacije i obrane projektnih zadataka. Projektne zadatke potrebno je dovršiti i predati, zajedno s dokumentacijom (pdf) i prezentacijom (ppt(x) ili pdf), do 5.1.2015. u 23:59.

Napraviti dvije aplikacije koje međusobno komuniciraju. Jedna aplikacija simulira proizvodni sustav. U njoj se samo tekstualno prikazuju promjene stanja u procesu. Druga aplikacija je nadzorni sustav koji prima informacije iz proizvodnog sustava (aplikacije koja ga simulira).

Iz nadzorne aplikacije se mora omogućiti pokretanje simulacije procesa što zapravo počinje simuliranje rada određenog pogona.

Aplikacije se mogu izraditi u bilo kojem programskom jeziku (C#, Java, Python, PHP, C/C++, Javascript, ...). Nadzorna i simulatorska aplikacija ne moraju biti izrađene u istom programskom jeziku. Komunikaciju između aplikacija realizirati na bilo koji način (TCP/IP, UDP, cijevi, XML, REST, dijeljena memorija...).

Komunikacijski protokol .....	2
Projekt SVPP_2014_1 - Punjenje boca na proizvodnoj traci. ....	3
Projekt SVPP_2014_2 - Montaža dijela na kućište.....	5
Projekt SVPP_2014_3 - Inspekcija boca na traci .....	7
Projekt SVPP_2014_4 - Vodosprema .....	9
Projekt SVPP_2014_5 – CNC stroj .....	11

## Komunikacijski protokol

Za komunikaciju koristiti sljedeći oblik protokola (odnosi se na projekte 1-4).

### *Slanje informacija o senzoru:*

**[Sxx-v]**

gdje je:

**xx** - broj senzora

**v** - vrijednost na senzoru

npr.

simulatorska aplikacija šalje podatak da je na senzoru 3 očitana logička jedinica

**[S03-1]**

simulatorska aplikacija šalje podatak da je na senzoru 1 očitana logička nula

**[S01-0]**

### *Slanje informacija o aktuatoru (motoru ili ventilu):*

**[Axx-v]**

gdje je,

**xx** - broj aktuatora

**v** - vrijednost aktuatora (ako je vrijednost veća od 1 onda je to podešavanje brzine)

npr.

simulatorska aplikacija šalje podatak da je motor 1 isključen

**[A01-0]**

Nadzorna aplikacija šalje podatak da se uključi motor 2

**[A02-1]**

### *Pokretanje ili zaustavljanje simulacije:*

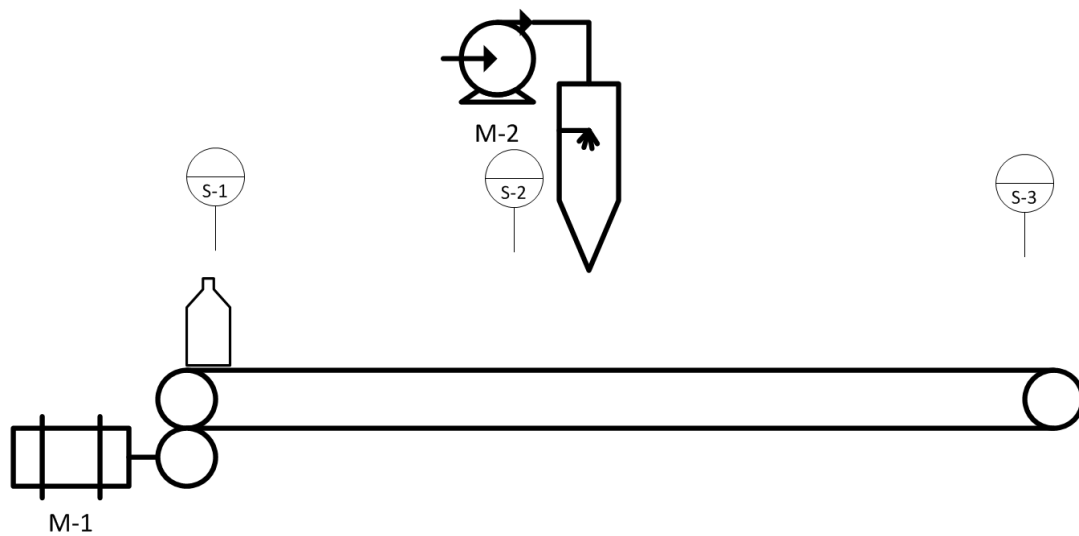
**[SIM-0]** - zaustavljanje

**[SIM-1]** - pokretanje

### *Napomena – Projekt SVPP\_2014\_5*

Za komunikaciju u projektu 5 koristi se slanje podataka definiranih u zadatku uz korištenje protokola za pokretanje ili zaustavljanje simulacije.

## Projekt SVPP\_2014\_1 - Punjenje boca na proizvodnoj traci.



Jedan motor koji pokreće traku, a drugi motor pokreće pumpu za punjenje. Jedan senzor se nalazi na početku pokretne trake i kad se na njoj nađe boca, pokreće se motor koji pogoni pokretnu traku. Drugi senzor se nalazi kod prostora za punjenje. Kad se boca nađe kod tog senzora potrebno je zaustaviti traku i pokrenuti pumpu. Nakon određenog vremena obrade, pumpu treba zaustaviti, ponovno pokrenuti traku i odvesti bocu do trećeg senzora na kraju trake kad ona nestaje. Nakon što je jedna boca došla do trećeg senzora, može doći druga boca na traku.

#### Način izrade simulatorske aplikacije:

Definirati varijable za motore i senzore.

Sve postaviti na početne vrijednosti: M1-0, M2-0, S1-0, S2-0, S3-0.

(Motori 1 i 2 isključeni, senzori ne očitavaju prisutnost boce)

Svaku promjenu stanja motora i senzora slati prema nadzornoj aplikaciji.

Po dobivanju naredbe za pokretanje simulacije započeti slijed.

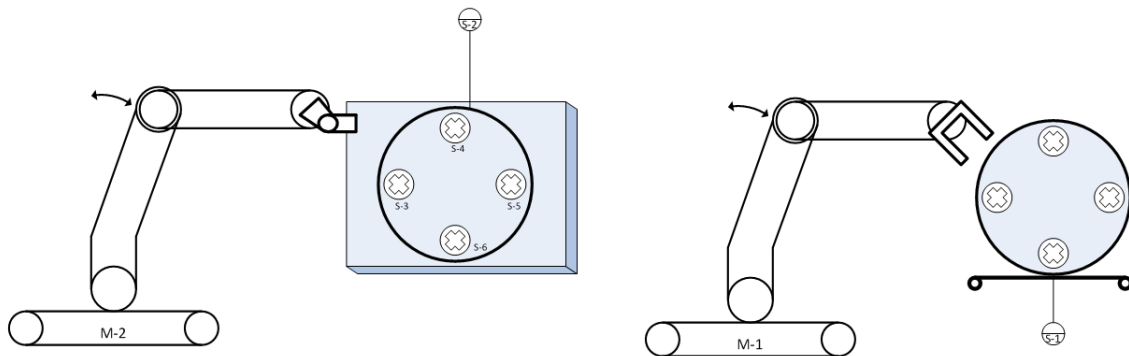
Ponavljati

```
{  
    Postaviti S1 na 1. (Boca se nalazi na početku trake)  
    Čekanje 300ms.  
    Postaviti M1 na 1. (Pali se motor pokretne trake)  
    Postaviti S1 na 0. (Boca odlazi s početka trake)  
    Čekanje 2000ms. (Simulacija voženja boce do punjenja)  
    Postaviti S2 na 1. (Boca došla do prostora za punjenje)  
    Postaviti M1 na 0. (Ugasiti motor pokretne trake)  
    Čekanje 300ms.  
    Postaviti M2 na 1. (Upaliti pumpu za punjenje)  
    Čekanje 2000ms (Simulacija punjenja)  
    Postaviti M2 na 0. (Zaustavljanje pumpe)  
    Postaviti M1 na 1. (Pokretanje pokretne trake)  
    Postaviti S2 na 0. (Boca otišla iz prostora za punjenje)  
    Čekanje 2000ms. (Simulacija voženja do kraja trake)  
    Postaviti S3 na 1. (Boca došla do kraja trake)  
    Postaviti M1 na 0. (Zaustaviti traku)  
    Čekanje 500ms.  
    Postaviti S3 na 0. (Boca nestaje - odlazi dalje u proces)  
    Čekanje 300ms.  
}
```

#### Nadzorna aplikacija:

Prikazuje stanja na grafičkoj pozadini paljenjem lampica ili animacijom. Omogućuje ručno aktiviranje motora kad je simulacija zaustavljena.

## Projekt SVPP\_2014\_2 - Montaža dijela na kućište



Jedan senzor govori kad je dio stigao na poziciju za obradu. Nakon toga se aktivira robot koji donosi dio do kućišta i tamo ga pridržava. Drugi senzor govori sustavu da je dio stigao do kućišta. Potom se aktivira drugi robot koji donosi stroj za pričvršćivanje i odvodi ga do spojne točke 1 (preprogramirana lokacija u robotu, nema senzora) te tamo započinje pričvršćivanje dok ne stegne na određenu razinu (senzor). Zatim se prebacuje na točku 2 i ponavlja. Sve se obrađuje u 4 točke i potom se odmiče prvi i drugi robot.

### Način izrade simulatorske aplikacije:

Definirati varijable za motore i senzore.

Sve postaviti na početne vrijednosti: M1-0, M2-0, S1-0, S2-0, S3-0, S4-0, S5-0, S6-0. (Motori 1 i 2 isključeni, senzori očitavaju 0)

Svaku promjenu stanja motora i senzora slati prema nadzornoj aplikaciji.

Po dobivanju naredbe za pokretanje simulacije započeti slijed.

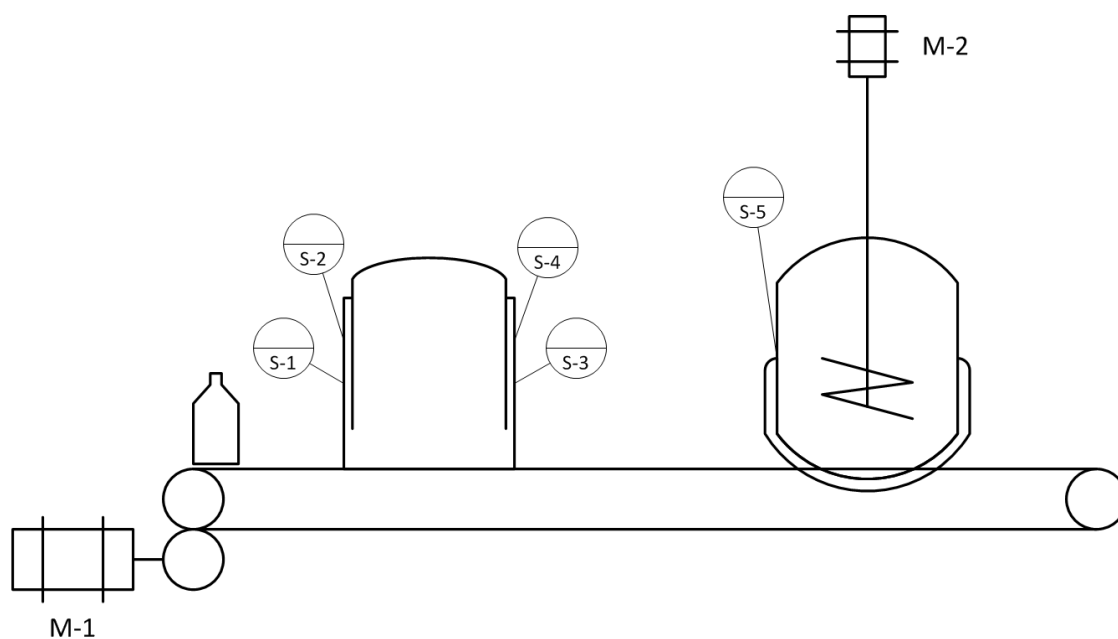
Ponavljati

```
{
    Postaviti S1 na 1. (Dio je stigao na poziciju za obradu)
    Čekanje 300ms.
    Postaviti M1 na 1. (Robot 1 se aktivira)
    Postaviti S1 na 0. (Dio više nije na poziciji za obradu)
    Čekanje 2000ms (Simulacija prijenosa dijela do kućišta)
    Postaviti S2 na 1. (Dio je stigao do kućišta)
    Postaviti M1 na 0. (Robot 1 se deaktivira)
    Čekanje 500ms.
    Postaviti M2 na 1. (Robot 2 se aktivira)
    Čekanje 500ms.
    Postaviti S3 na 1. (Stegnuta je prva vida)
    Čekanje 500ms.
    Postaviti S4 na 1. (Stegnuta je druga vida)
    Čekanje 500ms.
    Postaviti S5 na 1. (Stegnuta je treća vida)
    Čekanje 500ms.
    Postaviti S6 na 1. (Stegnuta je četvrta vida)
    Čekanje 500ms.
    Postaviti M2 na 0. (Robot 2 se deaktivira)
    Postaviti S2 na 0. (Dio i kućište odlaze dalje)
    Postaviti S3-S6 na 0.
}
```

### Nadzorna aplikacija:

Prikazuje stanja na grafičkoj pozadini paljenjem lampica ili animacijom. Omogućuje ručno aktiviranje motora kad je simulacija zaustavljena.

## Projekt SVPP\_2014\_3 - Inspekcija boca na traci



Boce putuju po traci. Motor trake je konstantno aktivan, omogućiti promjenu brzine. Boce prolaze kroz inspekciju cjelovitosti i valjanosti boce što je određeni uređaj s 4 senzora. Ukoliko bilo koji od senzora na nekoj boci uoči nedostatak boca će pri prolasku kroz stroj za uklanjanje (motor), koji se nalazi dalje na traci, biti uklonjena. Identificirati kojom se brzinom pomiču boce i ukoliko se detektira nedostatak kad stroj za uklanjanje treba ukloniti bocu.

#### Način izrade simulatorske aplikacije:

Definirati varijable za motore i senzore.

Sve postaviti na početne vrijednosti: M1-0, M2-0, S1-0, S2-0, S3-0, S4-0. (Motori 1 i 2 isključeni, senzori očitavaju 0)

Svaku promjenu stanja motora i senzora slati prema nadzornoj aplikaciji.

Po dobivanju naredbe za pokretanje simulacije započeti slijed.

Ponavljati

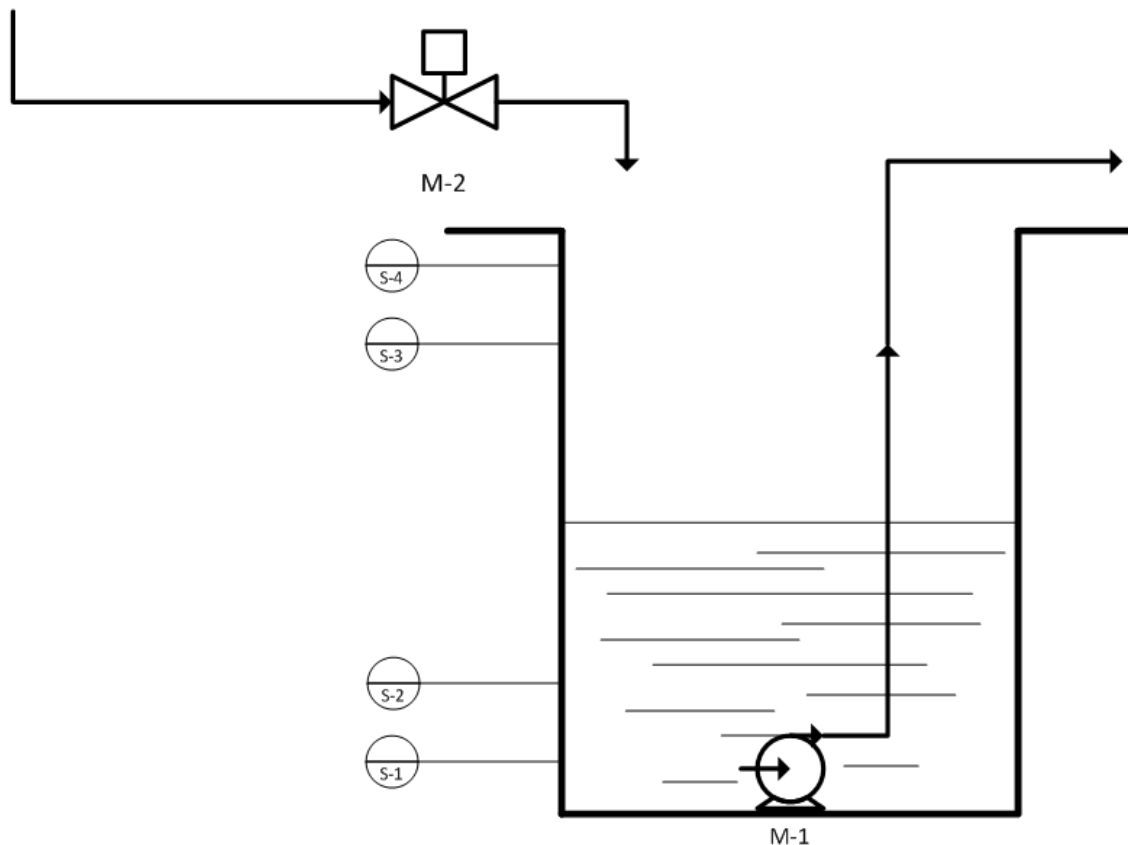
```
{  
    Postaviti M1 na vrijednost x. (Aktivirati motor od trake na  
    brzinu definiranu iz nadzorne aplikacije, x=2 do 30)  
    Čekanje 3000/x ms. (Simuliranje dolaska boce do uređaja za  
    inspekciju boce)  
    Postaviti vrijednost S1, S2, S3 i S4 na nasumične vrijednosti  
    između 0 i 1. Nasumične vrijednosti odabrati na način da samo u 10%  
    slučajeva vrijednost postane 1, a u 90% postane 0.  
    Čekanje 3000/x ms. (Simuliranje dolaska boce do stroja za  
    uklanjanje boce)  
    Ukoliko je S1, S2, S3 ili S4 postavljen na vrijednost 1,  
    postaviti M2 na 1. (Aktivirati motor stroja za uklanjanje boce)  
    Čekanje 200ms.  
    Postaviti M2 na 0.  
    Postaviti S1, S2, S3 i S4 na 0.  
    Čekanje 3000/x ms.  
}
```

#### Nadzorna aplikacija:

Prikazuje stanja na grafičkoj pozadini paljenjem lampica ili animacijom. Omogućuje ručno aktiviranje motora kad je simulacija zaustavljena.



## Projekt SVPP\_2014\_4 - Vodosprema



Spremnik tekućine sadrži određenu količinu tekućine. Na spremniku se nalaze 4 senzora (LL, L, H, HH). U spremnik je uronjena pumpa za crpljenje vode. Voda se crpi po aktivaciji iz nadzorne aplikacije. Kad se dostigne razina L potrebno je aktivirati obavijest. Ako se dostigne razina LL potrebno je isključiti pumpu i aktivirati alarm. Iz simulatora se definira i dolazak vode u preko ventila na cijevi za dovod vode. Ako pumpa ne radi spremnik se puni. Kad je dostignuta razina H potrebno je aktivirati obavijest u nadzornoj aplikaciji. Ako se dostigne HH potrebno je zatvoriti ventil koji služi za dovod vode i aktivirati alarm.

### Način izrade simulatorske aplikacije:

Definirati varijable za motore i senzore.

Definirati varijablu razina.

Sve postaviti na početne vrijednosti: M1-0 (pumpa), M2-0 (ventil), S1-0, S2-0, S3-0, S4-0. (Pumpa isključena, ventil zatvoren, senzori očitavaju 0) Varijablu razina postaviti na 500.

Svaku promjenu stanja motora i senzora slati prema nadzornoj aplikaciji.

Po dobivanju naredbe za pokretanje simulacije započeti slijed.

Ponavljati

```
{  
    Po primanju naredbe iz nadzorne aplikacije aktivirati pumpu  
    (postaviti M1 na 1) ili otvoriti ventil (postaviti M2 na 1).  
    Ukoliko je aktivna samo pumpa promijeniti vrijednost razine kao  
    razina=razina-10  
    Ukoliko je samo otvoren ventil promijeniti vrijednost razine kao  
    razina=razina+10  
    Ukoliko je aktivna pumpa i otvoren ventil vrijednost razine se  
    mijenja kao razina=razina+6.  
    Ukoliko je razina manja ili jednaka 60 postaviti S1 na 1.  
    (Dosegnuta vrlo niska razina - LL). S2, S3 i S4 postaviti na 0. Tad  
    se postavi M1 na 0. (Sigurnosno isključenje pumpe)  
    Ukoliko je razina manja ili jednaka 200, a veća od 60 postaviti  
    S2 na 1. (Dosegnuta niska razina - L) S1, S3 i S4 postaviti na 0.  
    Ukoliko je razina veća ili jednaka 800, a manja od 940 postaviti  
    S3 na 1. S1, S2 i S4 postaviti na 0.  
    Ukoliko je razina veća ili jednaka 940 postaviti S4 na 1. S1, S2  
    i S3 postaviti na 0. Tad se postavi M2 na 0. (Sigurnosno zatvaranje  
    dovodnog ventila.)  
    Ukoliko je razina veća od 200 i manja od 800 postaviti S1, S2,  
    S3 i S4 na 0.  
    Čekanje 100ms.  
}
```

### Nadzorna aplikacija:

Prikazuje stanja na grafičkoj pozadini paljenjem lampica ili animacijom. Omogućuje ručno aktiviranje motora i ventila kad je simulacija zaustavljena.

## Projekt SVPP\_2014\_5 – CNC stroj

Aplikacije za vođenje stroja po koordinatama. Stroju (simulatorska aplikacija) se iz upravljačke aplikacije upisuje slijed koordinata u obliku (x,y,OBRADA(T/F)). Može se upisati proizvoljni broj koordinata. U svaki red se upisuje nova koordinata i da li se obrađuje na putu do te koordinate. Za početak simulacije se šalje naredba za početak simulacije. Zatim se naredbe proslijeđuju simulatorskoj aplikaciji te se tamo marker pomiče od točke do točke i ukoliko je vrijednost OBRADA T(true) crta se liniju do te koordinate. Ako je vrijednost OBRADA F(false) marker samo putuje do te koordinate. Unaprijed se definiraju maksimalne vrijednosti koje koordinate mogu imati te se ne dozvoljava unos većih vrijednosti.