

Dodatni resursi: kontrola vremena. Zadatak se sastoji iz dva dela.

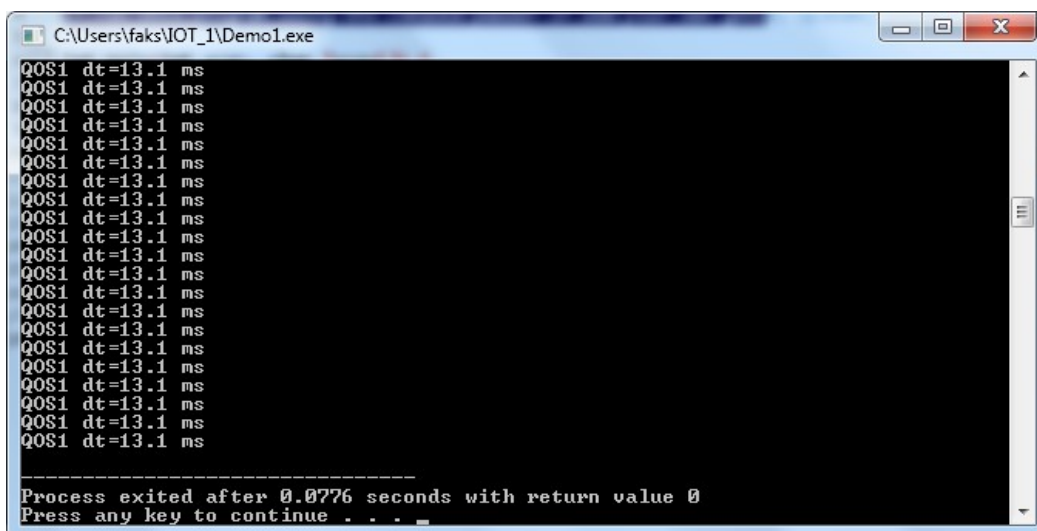
Za kontrolu tajminga unutar C programa koristi se jedna od standardnih bibliotekasadržana u time.h. Biblioteku treba (pored ostalih) uključiti sa #include <time.h>u program.

1. deo.

Napisati program koji testira prosečno vreme odziva brokera:

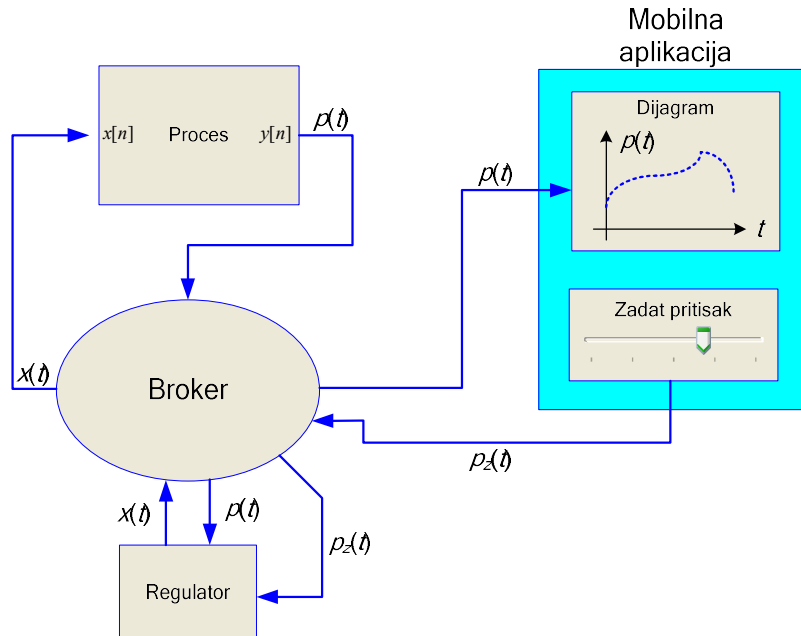
- a) Realizovati klijent program koji će biti prijavljen na topik, i koji će u isti taj topik u beskonačnoj petlji da upisuje test poruke proizvoljne ali promenljive sadržine, na primer brojeve od 0 do 255.
- b) Neposredno pre slanja prve poruke na topik uključi se tajmer.
- c) Čim se 100-ta poruka dobije nazad od brokera, isključi se tajmer i memoriše vremenski interval.
- d) Deljenjem vremenskog intervala sa 100 dobija se prosečno vreme odziva.
- e) Ponavljati prethodne 3 tacke u beskonačnoj petlji.
- f) Štampati na ekranu, jedno ispod dugog, prosečno vreme nakon svakog takvog ciklusa merenja vremena.
- g) Realizovati bilo kakvu komandu za izlazak iz programa
- h) Oceniti za sve tipove QOS prosečno vreme odziva, kao i odstupanje od prosečnog vremena.

Primer:



```
C:\Users\faks\IOT_1\Demo1.exe
QOS1 dt=13.1 ms
QOS1 dt=13.1 ms
QOS1 dt=13.1 ms
QOS1 dt=13.1 ms
QOS1 dt=13.1 ms
QOS1 dt=13.1 ms
QOS1 dt=13.1 ms
QOS1 dt=13.1 ms
QOS1 dt=13.1 ms
QOS1 dt=13.1 ms
QOS1 dt=13.1 ms
QOS1 dt=13.1 ms
QOS1 dt=13.1 ms
QOS1 dt=13.1 ms
QOS1 dt=13.1 ms
QOS1 dt=13.1 ms
QOS1 dt=13.1 ms
QOS1 dt=13.1 ms
QOS1 dt=13.1 ms
QOS1 dt=13.1 ms
-----
Process exited after 0.0776 seconds with return value 0
Press any key to continue . . . _
```

2. deo. Izabrati QOS za slučaj gde prosečno vreme odziva najmanje fluktuiira. To prosečno vreme se označava sa T_p . Sistem koji simulira kontrolu pritiska u industrijskom procesu prikazan je na sledećoj slici:



a) Industrijski proces je opisan diferencnom jednačinom koja modeluje sistem prvog reda sa vremenskom konstantom od 10s i pojačanjem 1:

$$y[n] = \frac{1}{21}(x[n] + x[n-1]) + \frac{19}{21}y[n-1]$$

$y[n]$ predstavlja pritisak $p(t)$ u relativnim jedinicama [0-100%]. Generisanje odbiraka $y[n]$ na osnovu pobudnog signala se obavlja periodom $T_s=1s$ (koristiti tajmer). Sve početne vrednosti za diferencnu jednačinu su 0. Kada god se generiše nova vrednost $p(t)$ ona se upisuje na odgovarajući topik. Mobilna aplikacija je prijavljena na taj topik i iscrtava na grafiku vrednosti pritiska svake sekunde - kad god se upiše novo $y[n]$ na taj topik. Pored toga, na mobilnoj aplikaciji se slajderom generiše zadati pritisak $p_z(t)$ u opsegu 10-90 koji se upisuje na drugi odgovarajući topik.

b) Regulator je prijavljen na topike trenutnog pritiska $p(t)$ (dobija svake sekunde) i zadatog pritiska $p_z(t)$ (dobija kad se promeni slajder). Regulatorom je realizovano histerezisno ON-OFF upravljanje tako da ako je trenutni pritisak $p(t)$ za Δp veći od $p_z(t)$ tada se na odgovarajući topik upisuje 0. Ukoliko je $p(t)$ za Δp manji od $p_z(t)$ tada se na taj topik upisuje 100. Ako je u granicama ne upisuje se ništa. Na taj topik je prijavljen industrijski proces i sa njega od brokera dobija ili 0 ili 100, i tu vrednost prepisuje u x_{temp} . U svakom trenutku odabiranja T_s , dobija se $x[n] = x_{temp}$.

c) Pokrenuti simulaciju za $\Delta p=5$. Evidentirati kvalitet regulacije za nekoliko položaje slajdera 10,40,60,80. Ponoviti za $\Delta p=1$.

Projekat se radi u parovima.

Sve C fajlove, slike ekrana, objašnjenja i ostalu dokumentaciju koja demonstrira urađene zahteve oblikovati u formu izveštaja, zipovati i poslati na adresu Milan@el.etf.rs. najkasnije do 1.9.2020.