

Sistemi u Realnom Vremenu

Projektni zadatak

FreeRTOS raspoređivanje sporadičnim serverom

Funkcionalnosti operativnog sistema

Implementirati sporadični server u FreeRTOS sistemu, sa podrškom za periodične i aperiodične taskove. Periodični taskovi se raspoređuju po RM algoritmu. Napraviti posebne funkcije za kreiranje i zaustavljanje periodičnih, kao i za kreiranje aperiodičnih taskova unutar tasks.c. Dozvoljeno je modifikovati FreeRTOS u proizvoljnoj meri. Nema potrebe da default FreeRTOS raspoređivač bude funkcionalan.

Sistem je inicijelno u fazi konfiguracije, gde se očekuje da se preko konzole dobiju parametri za sporadični server. Nakon što se dobiju ovi parametri, počinje se sa normalnim radom gde se izvršavaju taskovi po pristizanju zahteva, takođe preko konzole.

Periodični taskovi se kontrolišu preko konzole. Dozvoljeno je da ima konačno mnogo funkcija periodičnih taskova, i da one imaju pridružena konstantna stringovna imena koja se koriste pri startovanju i zaustavljanju posla preko serijske veze. Sistem automatski stavlja periodične taskove na ponovno izvršavanje. Sami taskovi treba da budu napisani tako da se funkcija prirodno završi nakon jednog izvršavanja, uz zahtev da funkcija taska mora da pozove `vTaskDelete()` ili slično pred kraj izvršavanja.

Korisnik može da započinje aperiodične taskove kada poželi na jedan od dva načina:

- Preko konzole, uz zadavanje svih parametara preko konzole.
- Pritiskom na taster, povezan na eksterni prekid. (važi samo za rad na mikrokontroleru; u kodu će biti fiksirano koji aperiodični task, i koje su vrednosti parametara za ovaj slučaj)

Neophodno je podržati i "batch" rad, tj. da korisnik zada skup periodičnih i aperiodičnih taskova, sa sve vremenima započinjanja aperiodičnih taskova, relativnim u odnosu na započinjanje čitavog batcha. Sistem treba da poseduje logiku za proveru rasporedivosti batcha.

Taskovi koji se izvršavaju mogu da budu proizvoljno jednostavni. Dovoljno je da postoji neki ispis na konzoli, ili po jedna LED dioda za svaki task, i kada task dođe na red, on ispiše svoju poruku, ili uključi svoju diodu i isključi sve ostale. Taskovi treba da budu napisani tako da je njihovo vreme izvršavanja zavisno od argumenta koji se zadaje pri startu posla. Skup parametara posla će minimalno imati: stringovni id, naziv funkcije koja se izvršava, periodu (za periodične taskove), proizvoljan argument. Vremenski parametri se zadaju u tickovima operativnog sistema. Taskovi treba da budu implementirani tako da neki od njih koriste `vTaskDelay()`, a neki ne.

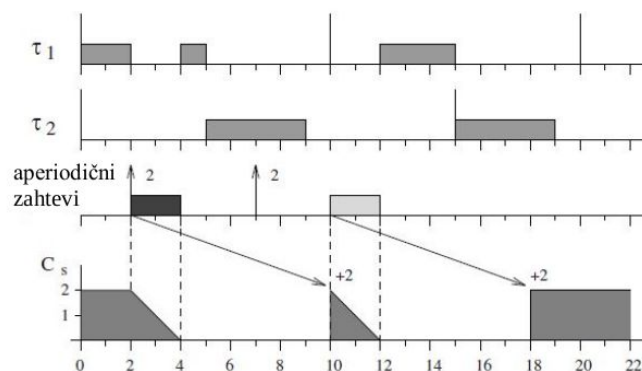
Interakcija sa sistemom

Za interakciju sa sistemom implementirati [Java aplikaciju koja komunicira sa mikrokontrolerom preko serijske veze](#). Ako se FreeRTOS implementira pod Eclipse, komunikacija između Java i OS se vrši preko datoteka, kao u drugom domaćem zadatku. Ova Java aplikacija treba da podrži:

- Inicijelno podešavanje sporadičnog servera.
- Startovanje periodičnog ili aperiodičnog taska sa zadavanjem relevantnih parametara.
- Zaustavljanje periodičnog taska.
- Startovanje više periodičnih i aperiodičnih taskova kao “batch” posao.
- Pri zadavanju “batch” posla, mikrokontroler treba da izvrši proveru rasporedivosti, i da javi nazad ako sistem nije rasporediv sa zadatim parametrima.
- Računanje maksimalnog kapaciteta servera koji zadržava rasporedivost, uz zadati skup periodičnih taskova. Račun se vrši na mikrokontroleru i vraća na Java aplikaciju.
- Za sve ove stavke treba da postoji posebna GUI komponenta za unos - nije prihvatljivo da Java aplikacija bude konzolna i da samo šalje String koji je korisnik zadao na mikrokontroler / upisuje String u datoteku.

Pored kontrole poslova, sistem takođe treba da prikazuje trenutno stanje sistema uz vremensku osu obeleženu tickovima sistema. Prikaz treba da bude aktivan samo ako ima startovanih periodičnih ili aperiodičnih taskova. Prikaz treba da uključuje:

- Markere za periode periodičnih taskova.
- Trenutke startovanja aperiodičnih taskova, uz informaciju o vremenu izvršavanja tog aperiodičnog taska.
- Jasan prikaz koji task je aktivan u kom trenutku (pravougaonici na donjem dijagramu).
- Kapacitet servera.



Napomene

Domaći zadatak se radi isključivo individualno.

Domaći zadatak može da se radi unutar Eclipse okruženja - simulirano, ili na mikrokontroleru. Rad na mikrokontroleru donosi dodatne poene. Ako se zadatak radi na mikrokontroleru, umesto konzole koristiti serijski port.

Umesto projekta mogu da se rade treći i četvrti domaći zadatak. Nije dozvoljeno kombinovati projekat i domaće zadatke, s time da nema potrebe prijavljivati se, tj. odlučiti se za jednu opciju u startu. Nakon što se odbrani treći domaći zadatak, projekat više nije dostupan.

Kontaktirati asistenta putem mejla za rezervisanje termina za rad sa Arduino Starter Kit.

Projekat se predaje putem mail-a (bmilojkovic@raf.rs), u ponoć, subota-nedelja, 11-12 januar. Za odbranu će biti iskorišćen termin u kolokvijumskoj nedelji rezervisan za SRV. Projekat mora da se odbrani do ovog termina.

Pri odbrani projekta je neophodno demonstrirati rad sistema u potpunosti.

Bodovanje:

- FreeRTOS - 30 poena, podeljeno na:
 - Provera rasporedivosti - 5 poena
 - Određivanje kapaciteta servera - 5 poena
 - Dodavanje i raspoređivanje taskova - 15 poena
 - Uklanjanje taskova - 5 poena
- Java aplikacija (važi samo uz implementiran raspoređivač) - 10 poena
- Projekat urađen na mikrokontroleru - 15 poena (važi samo uz implementiran raspoređivač)