

# Projekti SRV za školsku 2021/2022.

## 1 Zadatak

Realizovati sistem kojim se:

- AB==00

Startuje akvizicija sa kanala A0 i A1 na svakih 1000 ms pomoću:

- C==0 - softverskog tajmera
- C==1 - taska `xTaskTimer` koji koristi `vTaskDelayUntil`

Potrebno je implementirati odloženu obradu prekida (*deffered interrupt processing*) AD konvertora, tako što task koji će vršiti obradu podataka (`xTask1`) čeka na redu sa porukama (`Queue`), a iz prekidne rutine se upisuje u red sa porukama. Poruka treba da sadrži informaciju o kanalu koji je očitana i 12-bitnu vrednost koja je očitana.

`xTask1` računa srednju vrednost zadnjih 16 primljenih odbiraka za svaki pojedinačni kanal. Izračunatu srednju vrednost šalje taskovima `xTask2` i `xTask3` preko

- D==0 - *mailbox*-a (`Queue` dužine 1)
- D==1 - direktne notifikacije taska (*Direct-to-task notification*) koji se koristi u vidu *mailbox*-a

Vrednosti se upisuju sa atributom *overwrite*. Taskovi `xTask2` i `xTask3` se ne blokiraju kada treba da pročitaju vrednosti iz *mailbox*-a.

Na:

- E==0 - pritisak tastera S3 i S4
- E==1 - prijem karaktera '2' i karaktera '3' preko UART-a

taskovi `xTask2` i `xTask3`, respektivno, očitavaju odgovarajuće *mailbox*-ove i prikazuju očitane srednje vrednosti na multipleksiranom LED displeju (*Skalirati vrednost tako da se dobije dovoljno tavan prikaz*). Pristup multipleksiranom LED displeju je zaštićen:

- F==0 - *Gatekeeper* taskom `xTaskLED` sa kojim se komunicira preko reda sa porukama
- F==1 - muteksom

- AB==01

Startuje akvizicija sa kanala A0, A1 na svakih 1000 ms pomoću:

- C==0 - softverskog tajmera
- C==1 - taska `xTaskTimer` koji koristi `vTaskDelayUntil`

Potrebno je implementirati odloženu obradu prekida (*deffered interrupt processing*) AD konvertora, tako što se rezultat konverzije u prekidnoj rutini upisuje u red sa porukama (`Queue`) i obaveštava se task `xTask1` o prispeću nove poruke putem:

- D==0 - direktne notifikacije taskova (*Direct-to-task notification*) u vidu grupe događaja
- D==1 - grupe događaja (`EventGroup`)

Poruka treba da sadrži informaciju o kanalu koji je očitana i gornjih 9 bita rezultata AD konverzije.

Task `xTask1` čuva poslednju očitano vrednost za svaki kanal.

Task `xTask2`:

- E==0 - ispituje tastere S1-S4 i na pritisak odgovarajućeg tastera
- E==1 - implementira odloženu obradu prekida za UART *callback* rutinu i na prijem karaktera '1'-'4'

obaveštava task `xTask1` putem:

- D==0 - direktne notifikacije taskova (*Direct-to-task notification*) u vidu grupe događaja
- D==1 - grupe događaja (`EventGroup`)

o kanalu čije očitane vrednosti rezultata konverzije treba da šalje tasku `xTask3`. Svaki put kada stigne nova vrednost sa AD konvertora task `xTask1` smešta odgovarajući podatak u red sa porukama na kojem čeka task `xTask3`.

Task `xTask3` računa razliku između uzastopnih vrednosti očitano kanala i prikazuje na:

- F==0 - multipleksiranom LED displeju
- F==1 - UART-u

- AB==10

Startuje akvizicija sa kanala A0, A1 na svakih 500 ms pomoću:

- C==0 - softverskog tajmera
- C==1 - taska xTaskTimer koji koristi vTaskDelayUntil

Potrebno je implementirati odloženu obradu prekida (*deffered interrupt processing*) AD konvertora, tako što se rezultat konverzije u prekidnoj rutini upisuje u red sa porukama (Queue). Poruka treba da sadrži informaciju o kanalu koji je očitao i gornjih 8 bita rezultata AD konverzije.

Task xTask1 čita podatke iz reda sa porukama i računa srednju vrednost zadnji 8 podataka za svaki kanal pojedinačno.

Task xTask2:

- D==0 - ispituje tastere S3 i S4 i na pritisak odgovarajućeg tastera
- D==1 - implementira odloženu obradu prekida za UART *callback* rutinu i na prijem karaktera '3' ili '4'

obaveštava task xTask1 putem posebnog reda sa porukama o tasku kojem treba da pošalje srednje vrednosti rezultata konverzije (tasku xTask3 ili xTask4). Task xTask1 pakuje sva 4 podatka u jedan 32-bitni podatak i šalje ga odgovarajućem tasku putem

- E==0 - *mailbox*-a (Queue dužine 1)
- E==1 - direktne notifikacije taska (*Direct-to-task notification*) koji se koristi u vidu *mailbox*-a

Vrednosti se upisuju sa atributom *overwrite*. Taskovi xTask3 i xTask4 se ne blokiraju kada treba da pročitaju vrednosti iz *mailbox*-a.

Taskovi xTask3 i xTask4 po prijemu podatka prikazuju sva četiri podatka na

- F==0 - multipleksiranom LED displeju (najviša 4 bita srednje vrednosti po kanalu, jedna cifra po kanalu)
- F==1 - UART-u (jedna poruka sa sve 4 vrednosti)

Obezbediti kontrolu pristupa izlaznoj periferiji (u zavisnosti od F).

*Napomena:* Task xTask1 treba da čeka na oba reda sa porukama istovremeno.

- AB==11 Startuje akvizicija sa kanala A0, A1 na svakih 1500 ms pomoću:
  - C==0 - softverskog tajmera
  - C==1 - taska `xTaskTimer` koji koristi `vTaskDelayUntil`

Potrebno je implementirati odloženu obradu prekida (*deffered interrupt processing*) AD konvertora, tako što se rezultat konverzije u prekidnoj rutini se upisuje u red sa porukama (`Queue`) i obaveštava se task `xTask1` o prispeću nove poruke putem:

- D==0 - direktne notifikacije taskova (*Direct-to-task notification*) u vidu binarnog semafora
- D==1 - binarnog semafora (`Binary semaphore`)

Poruka treba da sadrži informaciju o kanalu koji je očitana i gornjih 12 bita rezultata AD konverzije.

Task `xTask1` računa srednju vrednost zadnja 4 primljena odbirka za svaki kanal i upisuje izračunatu srednju vrednost u:

- E==0 - *mailbox*-ove (`Queue` dužine 1) sa *overwrite*-om
- E==1 - promenljive zaštićene muteksom

Preko UART-a je moguće zadati graničnu vrednost za signalizaciju. Iz *callback* rutine UART-a se podatak o željenoj graničnoj vrednosti smešta u red sa porukama `xQueueT2` i task `xTask2` treba da očita tu vrednost.

Ukoliko je očitana srednja vrednost određenog kanala:

- F==0 - veća
- F==1 - manja

od granične vrednosti, odgovarajuća dioda treba da se uključi.

Task `xTask3` ispituje stanje tastera S1 i S2 i šalje obaveštenje tasku `xTask2` preko reda `xQueueT2` ukoliko je neki od tastera pritisnut. Task `xTask2` na pritisak odgovarajućeg tastera inkrementira/dekrementira trenutnu graničnu vrednost i informaciju o trenutnom stanju dioda (uključene/isključene) šalje preko UART-a.

## 2 Uputstvo za rad i predaju

Student radi jedan od zadataka definisan parametrima A, B, C, D, E i F. Traženi parametri određuju se na osnovu binarnog zapisa rednog broja studenta na listi koja je u prilogu, gde je A bit najveće a F bit najmanje težine. Studenti čiji je redni broj sa spiska (RB) veći od 63, parametre A,B,C,D i F određuju po formuli  $RB \bmod 64$

Svi projekti se rade na razvojnoj ploči baziranoj na RS\_MSP430F5529 mikrokontroleru koja je korišćena na računskim vežbama.

Ocenjivanje projekata se realizuje kroz dve faze:

- 1. faza - ocenjuje asistent
- 2. faza - ocenjuje profesor

Uspešno ocenjena prva faza je uslov da bi se realizovala druga faza ocenjivanja rešenja projektnog zadatka. U okviru prve faze, na ocenjivanje se predaje kod projektnog zadatka, koji implementira tražene funkcionalnosti, i kratak izveštaj (**v1**) koji sadrži

- Blok dijagram koji opisuje arhitekturu realizovanog softvera sa koje se vidi način na koji su sinhronizovani taskovi
- Dijagram aktivnosti za svaki od implementiranih taskova i za svaku prekidnu rutinu
- Opciono vremenski dijagram (*Ukoliko je potrebno dodatno pojasniti neke funkcionalnosti*)

U okviru druge faze predaje se isključivo detaljan izveštaj (**v2**) koji predstavlja nadogradnju (**v1**) gde su detaljnije objašnjeni svi aspekti realizacije.

**v1** izveštaj treba napisati tako da je jednoznačno moguće pronaći elemente iz dijagrama u *source* kodu projektnog rešenja. Na primer, ako se u dijagramu neki task označava sa *xTaskDsp* taj task treba da postoji i u kodu. Za kreiranje traženih dijagrama moguće je koristiti *DrawIO* softver <https://app.diagrams.net>.

Izveštaj **v1** se šalje isključivo u PDF formatu čija struktura naziva treba da bude u sledećem formatu `V1_RB_GGGG_BBBB_Ime_Prezime.pdf` (RB je redni broj sa spiska datog u nastavku. Na primer RB je 00, 02, 03 ... 34 ...

Početnu tačku realizacije *source* koda projektnog zadatka predstavlja template projekat (kao što je rađeno i na labovima). Pored realizacije traženih funkcionalnosti, kod realizovanog rešenja treba da bude uredno napisan sa dovoljnim brojem komentara. Kod koji nije smisleno iskomentarisano neće biti razmatran.

Obično se dešava da fakultetski email servis isfiltrira emailove koji sadrže izvršne fajlove (kao što je .exe, .elf, ...). Nakon kompajliranja koda vašeg projektnog zadatka uvek se iznova kreira *Debug* folder u okviru projektnog direktorijuma. U ovom direktorijumu se nalaze izvršni fajlovi koji mogu dovesti do toga da pošaljete rešenje a da ono ne bude primljeno. Zbog toga, neposredno pre zipovanja rešenja, izbrišite *Debug* folder

Projekat koji sadrži kod rešenja, zajedno sa izveštajem **v1** u PDF formatu, zapakovati u arhivu čija struktura naziva treba da bude `RB_GGGG_BBBB_Ime_Preime.zip`. Zapakovanu arhivu poslati na email [haris@etf.rs](mailto:haris@etf.rs) sa naslovom `SRV-Projekat-2022-RB`.

### 3 Spisak studenata

0. 2002/0363 Татјана Бабић
1. 2011/0166 Ђировић Стефан
2. 2012/0304 Баришић Милица
3. 2013/0468 Крушчић Рашко
4. 2013/0514 Боровић Јелена
5. 2013/0557 Вујић Милан
6. 2015/0038 Бралић Андреј
7. 2015/0065 Стипсић Павле
8. 2015/0300 Ђосовић Алекса
9. 2015/0425 Ђурђевић Светислав
10. 2015/0651 Анђела Генчић
11. 2016/0006 Ђурчин Александар
12. 2016/0282 Гарашанин Јован
13. 2016/0335 Ивковић Филип
14. 2016/0370 Стошић Младен
15. 2016/0389 Лазовић Лука
16. 2016/0472 Зиндовић Стефан
17. 2016/0494 Крстић Лука
18. 2017/0017 Николина Бунијевац
19. 2017/0041 Милашиновић Андрија
20. 2017/0225 Ристић Филип
21. 2017/0284 Ђурчић Предраг
22. 2017/0307 Шћекић Марко
23. 2017/0321 Новаковић Стеван
24. 2017/0473 Јовановић Бојан
25. 2017/0526 Кангрга-Микулић Стефан
26. 2017/0611 Илић Вељко
27. 2017/0649 Александра Марјановић
28. 2017/0658 Милошевић Александар
29. 2017/0712 Валентина Цветковић
30. 2018/0037 Јахић Алма
31. 2018/0038 Дамњановић Филип
32. 2018/0044 Антић Ђорђе
33. 2018/0065 Боровић Никола
34. 2018/0068 Красић Лука
35. 2018/0075 Арсеновић Марко
36. 2018/0090 Мрђа Никола
37. 2018/0101 Миловић Милица
38. 2018/0109 Ристивојевић Ленка
39. 2018/0120 Маринковић Реља
40. 2018/0142 Марковић Љубица
41. 2018/0163 Фирез Андрија
42. 2018/0187 Ђосић Ивана
43. 2018/0200 Пртењак Николина

44. 2018/0201 Куљанин Сара
45. 2018/0212 Војиновић Александар
46. 2018/0214 Сарић Жарко
47. 2018/0226 Лежаја Стеван
48. 2018/0246 Сретовић Алекса
49. 2018/0284 Савић Катарина
50. 2018/0319 Калуђеровић Љубомир
51. 2018/0384 Јанковић Јана
52. 2018/0395 Миловановић Јелена
53. 2018/0460 Ивановић Немања
54. 2018/0503 Вујиновић Милан
55. 2018/0698 Перић Лазар
56. 2018/0717 Марија Турањанин
57. 2019/0735 Ђерић Васо
58. 2020/3427 Димитријевић Јелена
59. 2021/3472 Станковић Марко

*Напомена:* Студенти који су претходне године добили пројекат могу да раде исти пројекат. Потребно је само да обавесте асистента о томе.