

# Микропроцесорски системи (13х114мипс)

## Пример друге предиспитне обавезе

### Напомене:

(а) Решење се оцењује по принципу “ради или не ради” посматрано на нивоу сваке појединачне ставке.

(б) Поени за ставку се добијају искључиво уколико се помоћу адекватног примера прикаже да је урађено све што се датом ставком тражи. Формирање адекватног примера је задатак студента. Непостојање примера или неадекватан пример тумачи се као да студент није разумео суштину која се датом ставком испитује услед чега за њу добија нула поена. Формирање засебног адекватног примера није обавезно за сваку појединачну ставку уколико је на основу једног примера могуће приказати да је више ставки исправно урађено.

(в) Све ставке, за које ће бити додељен ненулта број поена, морају истовремено бити испуњене приликом доделе укупног броја поена осим уколико поставком задатка није другачије експлицитно назначено.

### Задатак:

Користећи алате *arm-none-eabi-\**, *build-tools* и *Eclipse CDT*, који се налазе распаковани у оквиру [C:\Temp](#) директоријума, *Proteus* симулатор и *CubeMX* алат потребно је испунити ставке које се налазе у наставку. Приликом израде решења на располагању је следећа документација:

- *GNU make* ([https://www.gnu.org/software/make/manual/html\\_node/index.html](https://www.gnu.org/software/make/manual/html_node/index.html))
- *GNU as* (<https://sourceware.org/binutils/docs/as/>)
- *GNU ld* (<https://sourceware.org/binutils/docs/ld/>)
- *GNU gcc* (<https://gcc.gnu.org/onlinedocs/gcc-10.2.0/gcc/>)
- микроконтролер *STM32F103R6* (мрежни диск са материјалима)

(1) [3 поена] Направити *CubeMX* пројекат за микроконтролер *STM32F103R6*. Конфигурисати пине *PB12-PB15* као општенаменске у излазном моду. Потом направити *Proteus* пројекат и на пине *PB12-PB15* повезати по један *LED (light emitting diode)*. Написати функцију са следећим потписом:

```
void setGPIOB(uint16_t GPIO_PIN, GPIO_PinState GPIO_PIN_STATE)
```

која треба на пин *GPIO\_PIN* порта *B* да постави вредност *GPIO\_PIN\_STATE*. Уколико се проследи пин који није из опсега *PB12-PB15*, функција треба бити без дејства. Потребно је користити *HAL* библиотеку приликом имплементације функције.

(2) [3 поена] Конфигурисати пин *PB11* микроконтролера *STM32F103R6* тако се генерише прекид уколико се на улазном сигналу овог пина препозна силазна ивица. У симулатору *Proteus* повезати на овај пин дугме користећи *pull down* отпорник. Унутар прекидне рутине која одговара овом прекиду инвертовати стање на пину *PB12*.

(3) [3 поена] У симулатору *Proteus* повезати седмосегментни екран ширине четири цифара, а кроз алат *CubeMX* омогућити генерисање прекида од стране бројача *TIM1* услед *update event*. Написати програмски код који ће приказивати цифре 1, 2, 3 и 4 на екрану.

- (4) [3 поена] Написати програмски код који ће коришћењем истог бројача из претходне тачке реализовати штоперицу која на две више цифре приказује број минута, а на две ниже цифре приказује број секунди на седмосегментном екрану.
- (5) [2 поена] Допунити прекидну рутину из тачке 2) тако да врши ресетовање штоперице.
- (6) [4 поена] Коришћењем бројача *TIM1* и то његовог канала 1 израчунати фреквенцију сигнала генератора такта који се доводи на пин *PA8*. Фреквенцију изразити у *mHz*. У симулатору *Proteus* на пин *PA8* повезати генератор такта и подесити његову периоду на 2 секунде.
- (7) [2 поена] На пин *PB10* у симулатору *Proteus* повезати дугме из тачке 2). Реализовати преко овог пина да се приликом држања дугмета у притиснутом положају на седмосегментном екрану уместо времена штоперице испише израчуната фреквенција у тачки 6).
- (8) [4 поена] Повезати *LED (light emitting diode)* на пин *PA6* у симулатору *Proteus*. Користећи бројач *TIM3* и његов канал 1 реализовати да ова диода не светли 0,5 секунди, па након тога светли 1,5 секунду и тако наизменично. Није дозвољено директно односно експлицитно постављање вредности пина *PA6* ни на који начин кроз програмски код.