

Област вежби: *Linux* руковаоци

## РУКОВАОЦИ У/И УРЕЂАЈА НА LINUX ОПЕРАТИВНОМ СИСТЕМУ – ЗАДАТАК 1

### Предуслови:

- Rpi2 рачунар са додатном плочицом са LE диодама, прекидачима и тастерима,
- Преводиоц *GCC* освежен на верзију 4.7 или новију,
- Преузето и подешено језгро Linux оперативног система на Raspberry Pi уређају, према опису из документа „УВОД - Raspberry Pi рачунар“,
- Подешен мрежни приступ на један од начина представљених у документу „УВОД - Raspberry Pi рачунар“ уколико се ради преко мреже. Ако се Rpi2 рачунар користи као самосталан рачунар овај захтев се може занемарити,
- Познавање језика Це и материјала из вежби „УВОД У КОНКУРЕНТНО ПРОГРАМИРАЊЕ, „СИНХРОНИЗАЦИЈА И СИГНАЛИЗАЦИЈА ПРОГРАМСКИХ НИТИ“ и „ТУТОРИЈАЛ О РУКОВАОЦИМА У/И УРЕЂАЈА НА LINUX ОПЕРАТИВНОМ СИСТЕМУ“.

### Увод

У овој вежби је приложен код за Linux модул и пример тестне апликације у препознатљивој структури директоријума са одговарајућим Makefile-овима. Да би се реализовао задатак, неопходно је проширити дати Linux модул и направити одговарајућу тестну апликацију, а приложену користити као полазну тачку. Приложени Linux модул на сваких 250 ms укључује/искључује LE диоду LED0, и уједно читава стање прекидача SW0 и исписује га у *kernel log*.

### Задатак

Задатак је потребно реализовати по угледу на примере из туторијала о прављењу руковаоца („ТУТОРИЈАЛ О РУКОВАОЦИМА У/И УРЕЂАЈА НА LINUX ОПЕРАТИВНОМ СИСТЕМУ“) и конкурентног програмирања („УВОД У КОНКУРЕНТНО ПРОГРАМИРАЊЕ“ и „СИНХРОНИЗАЦИЈА И СИГНАЛИЗАЦИЈА ПРОГРАМСКИХ НИТИ“). Руковалац *memory* представља репрезентативни пример на основу којег треба да се реализује руковалац и његова тестна апликација која треба да провери његову функционалност. И овај руковалац треба реализовати као карактерни уређај (*chardev*).

На основу физичке архитектуре LE диода (LED0~LED3) и прекидача (SW0~SW3), односно њиховог мапирања на одговарајуће GPIO пролазе дате на слици 1, потребно је:

1. Имплементирати подршку за руковање улазно/излазним уређајима (GPIO).  
Руковалац GPIO треба да омогући:

- У поступку иницијализације модула поставити GPIO пролазе везане на диоде/прекидаче у одговарајући смер (GPIO за LE диоде као излазне, GPIO за прекидаче као улазне),
- Промену стања LE диода кроз операцију писања:
  - LED x 1 (укључивање LE диоде x)
  - LED x 0 (искључивање LE диоде x)

Приликом имплементације водити се примером са слике, односно користити дате идентификаторе пролаза и користити постојеће функције за дефинисање смера пролаза, промену стања излаза (диоде), односно читање вредности са улаза (прекидачи):

- void SetGpioPinDirection(char pin, char direction)
- void SetGpioPin(char pin)
- void ClearGpioPin(char pin)
- char GetGpioPinValue(char pin)

2. Параметризовати руковалац тако да се омогући иницијализација LE диода на прослеђену вредност.
3. Проширити руковалац тако да се омогући и читање стања са прекидача кроз операцију читања.
4. Проширити тест апликацију употребом *pthread* библиотеке тако да се омогући конкурентно читање стања прекидача и промена стања LE диода. Направити 4 нити које читавају стање са прекидача, и на сваку промену стања прекидача мењају стање LE диоде у складу са тим, тако да стање диода LED0~LED3 одговара стању прекидача SW0~SW3. Водити рачуна о заштити дељених ресурса с обзиром да је сада дескриптор датотеке дељени ресурс (уколико ова заштита није реализована у руковаоцу).

LED0	LED1	LED2	LED3	SW0	SW1	SW2	SW3
GPIO6	GPIO13	GPIO19	GPIO26	GPIO12	GPIO16	GPIO20	GPIO21

Слика 1 Мапирање LE диода и прекидача на GPIO пролазе