

Univerzitet u Sarajevu

Elektrotehnički fakultet

Ugradbeni sistemi 2023 / 24

Izvještaj za laboratorijsku vježbu br. 8

Upoznavanje sa Raspberry Pi

Ime i prezime: **Ivona Jozić**

Broj indeks-a: **19357**

07. maj 2024.

Sadržaj

1 Pseudokod i / ili dijagram toka.....	3
1.1 Zadatak 5	3
2 Analiza programskog rješenja	4
2.1 Zadatak 1	4
2.2 Zadatak 2	4
2.3 Zadatak 3	4
2.4 Zadatak 4	4
2.5 Zadatak 5	4
3 Korišteni hardverski resursi.....	5
4 Zaključak	5
5 Prilog	6
5.1 Zadatak 5 / izvorni kod.....	6

1 Pseudokod i / ili dijagram toka

U laboratorijskoj vježbi 8 je akcenat bio na upoznavanju sa Raspberry Pi 2B računarom, te su zadaci bili koncipirani tako da se steknu osnovna znanja i vještine u radu. S obzirom na to, samo je u 5. zadatku ove laboratorijske vježbe bilo zahtijevano kreiranje vlastitog programskog rješenja, pa će u nastavku biti prezentiran samo taj pseudokod. Kompletно rješenje je moguće pronaći u odjeljku 5 ovog izvještaja.

1.1 Zadatak 5

Prilikom razmatranja pseudokoda, treba imati na umu da je na Raspberry Pi-u obrnuta logika, odnosno da je pri maksimalnoj vrijednosti duty cycle (100) najmanji intenzitet svjetla na LED diodi, dok kada je duty cycle 0 onda LED dioda svijetli najjačim intenzitetom.

```
led_duty -> 100
while(1)
    if(taster pritisnut)
        led_duty -> led_duty - 10

        if(led_duty < 0)
            led_duty -> 100
        end_if

        promijeni_duty (led_duty)
    end_if

    sleep(0.01)
end_while
```

2 Analiza programskog rješenja

2.1 Zadatak 1

Cilj prvog zadatka je bio upoznavanje sa digitalnim ulazima i izlazima na Raspberry Pi računaru. U terminalu su pokretane osnovne naredbe za paljenje i gašenje LED dioda povezanih na GPIO2 (plava), GPIO3 (zelena), GPIO4 (crvena). Ono što je interesantno za primijetiti je činjenica da se LED diode pale logičkom 0, a gase logičkom 1. Nakon toga je bilo potrebno očitavati stanje tastera poveznog na GPIO17 koji predstavlja digitalni ulaz. Slično kao i sa LED diodama, u trenutku kada je taster otpušten kao rezultat naredbe se dobiva logička 1, a kada je pritisnut rezultat je 0. Za pokretanje naredbi je bilo neophodno koristiti administratorske (sudo) privilegije.

2.2 Zadatak 2

U drugom zadatku je na Raspberry Pi bilo potrebno kreirati bash skriptu koja se nalazi u prilogu vježbe, te istu pokrenuti. Sve je bilo potrebno uraditi koristeći naredbe u terminalu sa administratorskim privilegijama. Rezultat izvršenja skripte je naizmjenično paljenje i gašenje plave LED diode (GPIO2) na svakih 1s.

2.3 Zadatak 3

Treći zadatak je za cilj imao da demonstrira da je GPIO Raspberry Pi računara moguće koristiti i pomoću neke od raspoloživih datoteka, kao što je WiringPi. Izvorni kod datoteke je preuzet, a zatim kompajliran, nakon čega je kreiran C program koji naizmjenično pali i gasi LED diodu (GPIO2) na svakih pola sekunde. Kod koji je pokrenut je preuzet iz priloga za ovu laboratorijsku vježbu.

2.4 Zadatak 4

Kod iskorišten za 4. zadatak za zadatak ima promjenu duty cycle na LED diodama, tako što se isti podešava na osnovu mjerenja tri ose akcelerometra na mobilnom telefonu. Komunikacija između računara i mobilnog uređaja zasniva se na http protokolu. Prvo je bilo potrebno instalirati aplikaciju koja je namjenski kreirana za potrebe ove vježbe i podesiti ispravnu IP adresu kao i http server, zatim podesiti IP adresu računara, te nakon toga kreirati skriptu i izvršiti njen sadržaj na računaru. Kada je sve uspješno instalirano i pokrenuto, svakom promjenom položaja mobilnog telefona dolazi do promjene intenziteta boja na LED diodama, čime se kreiraju različite boje.

2.5 Zadatak 5

Cilj petog zadatka je bio upoznavanje sa GPIO bibliotekom, koja omogućava korištenje digitalnih ulaza i izlaza, te PWM. Bilo je potrebno napisati kod koji omogućava da se na plavoj LED diodi realizira PWM, te da se pri svakom pritisku na taster duty cycle uvećava za 10%, te se vraća na 0 kada dostigne 100%. Za implementaciju ovog zadatka, bilo je neophodno inicijalizirati GPIO, a zatim kroz while petlju kontrolisati stanje tastera. Ukoliko je on pritisnut, duty cycle se uvećava unutar dozvoljenih granica. Ono što je važno napomenuti je da je na Raspberry Pi-u obrnuta logika, odnosno da je pri maksimalnoj vrijednosti duty cycle (100) najmanji intenzitet svjetla na LED diodi, dok kada je duty cycle 0 onda LED dioda svijetli

najjačim intenzitetom. Shodno tome, unutar while petlje je umjesto povećavanja duty cycle, implementirano njegovo umanjenje, a inicijalna vrijednost je podešena na 100.

3 Korišteni hardverski resursi

Za potrebe laboratorijske vježbe 8 korišten je računar Raspberry Pi 2B, baziran na procesoru sa četiri ARM Cortex-A7 jezgra, koja rade na 1GHz, te sa 1GB RAM memorije. U sklopu računara su korištene tri LED diode, te jedan taster za realizaciju postavljenih zadataka u sklopu vježbe.

Za Raspberry Pi računar:

- 3 LED diode
- taster

ULAZI	IZLAZI
Taster (digitalni; GPIO17)	LED plava (digitalni; GPIO2)
	LED zelena (digitalni; GPIO3)
	LED crvena (digitalni; GPIO4)



Slika 1. – Manji duty cycle u 5. zadatku



Slika 2. – Veći duty cycle u 5. zadatku

4 Zaključak

Prilikom izvođenja Laboratorijske vježbe 8 nije bilo poteškoća bez obzira što je korišten potpuno novi računar sa kojim se do sada nismo imali priliku susresti. Upute u samoj postavci su bile dovoljno detaljne, tako da su svi zadaci radili u skladu sa očekivanjima. „Najkomplikovaniji“ dio vježbe je bio na samom početku kada je bilo potrebno instalirati OS i povezati računar na internet, dok su svi ostali dijelovi završeni u hodu. Cilj vježbe je bio upoznavanje sa Raspberry Pi računarom, što je i postignuto.

5 Prilog

U prilogu je dat kod za 5. zadatak iz ove laboratorijske vježbe koji se može pokrenuti na Raspberry Pi računar.

5.1 Zadatak 5 / izvorni kod

```
try:
    import RPi.GPIO as GPIO
except RuntimeError:
    print("Unable to import RPi.GPIO library. Run your script using 'sudo'")

from time import sleep

GPIO.setmode(GPIO.BCM)

GPIO.setup(17, GPIO.IN)
GPIO.setup(2, GPIO.OUT)
GPIO.setup(3, GPIO.OUT)
GPIO.setup(4, GPIO.OUT)
GPIO.output(3, True)
GPIO.output(4, True)

channels = [17, 2]

led = GPIO.PWM(channels[1], 1000)
led.start(100)
currentDuty = 100

while True:
    if GPIO.input(channels[0]):
        currentDuty = currentDuty - 10
        if currentDuty == 0:
            currentDuty = 100
        led.ChangeDutyCycle(currentDuty)
        while GPIO.input(channels[0]):
            pass

    sleep(0.01)
```