

Univerzitet u Sarajevu
Elektrotehnički fakultet
Ugradbeni sistemi 2023 / 24

Izveštaj za laboratorijsku vježbu br. 8
Upoznavanje sa Raspberry Pi

Ime i prezime: **Ismar Muslić**
Broj index-a: **19304**

07. maj 2024.

Sadržaj

1 Pseudokod	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
4 Zaključak	6
5.1 Zadatak 5/Izvorni kod	7

1 Pseudokod

U ovoj laboratorijskoj vježbi veći fokus je bio na upoznavanju sa sistemom Raspberry Pi kroz konzolno okruženje, dok je samo za 5. zadatak bilo potrebno programsko rješenje, pa je samo za njega i pružen pseudokod.

1.1 Zadatak 5

```
led.dutyCycle(100)
while(true)
    if(taster pritisnut)
        dutyCycle -= 10
        if(dutyCycle < 0)
            led.dutyCycle(100)

    led.dutyCycle(dutyCycle)

    sleep(0.01)
```

2 Analiza programskog rješenja

2.1 Zadatak 1

U prvom zadatku je bilo potrebno upoznati se sa korištenjem GPIO portova na sistemu Raspberry Pi, kroz konzolno okruženje. Ovi portovi se kontroliraju na način da se modificira odgovarajući pseudofajl za dati port u direktoriji `/sys/class/gpio`. Na taj način su komandama u terminalu paljene i gašene LED diode (GPIO2 – plava, GPIO3 – zelena, GPIO4 – crvena), te je očitavana vrijednost tastera (GPIO17). Interesantno za primijetiti jeste da je za upaljenu LED diodu potrebno podesiti stanje GPIO porta na 0. Također, kod očitavanja stanja tastera je za pritisnut taster vrijednost bila 0, a za otpušten 1.

2.2 Zadatak 2

Zadatak 2 je zahtijevao kreiranje i pokretanje bash skripte koja upravlja sa GPIO2 portom (plava LED dioda). Skripta je spremljena u direktorij `~`, a ista pali i gasi plavu LED diodu na svakih 1s. Skriptu je bilo potrebno pokrenuti administratorskim privilegijama, te se isto radilo standardno za terminal operativnog sistema Linux – korištenjem sudo komande.

2.3 Zadatak 3

Treći zadatak je trebao da pokaže da se sa GPIO portovima sistema Raspberry Pi može upravljati i koristeći neku drugu biblioteku. Tako je korištena biblioteka WiringPi, čiji izvorni kod je preuzet za git repozitorije, te zatim i buildan. Nadalje, korištenjem gcc kompajlera je kompajliran program u C-u koji koristi ovu biblioteku, koji je zatim i pokrenut. Demonstrirano je da program funkcioniše, uspješno upravljajući sa GPIO pinovima koristeći WiringPi biblioteku.

2.4 Zadatak 4

Kod koji je dat kao dodatni materijal za korištenje u 4. zadatku funkcioniše tako da se veže na http server te čita vrijednosti akcelerometra sa mobilnog telefona, a na kojem je instalirana namjenska aplikacija za ovu LV. Naime, Android aplikacija čita vrijednosti akcelerometra te ih preko http protokola šalje Raspberri Pi-u na mreži (postavljanjem tačne IP adrese je to osigurano), na osnovu čega sistem mijenja vrijednost duty cycle-a LED dioda koje su kontrolisane PWM-om. Kada se program pokrene, boje na LED diodama se mijenjaju zbog konstantne promjene intenziteta usljed pomjeranja telefona.

2.5 Zadatak 5

U posljednjem zadatku bilo je potrebno koristeći ugrađenu podršku za GPIO Raspberry Pi-ja u pythonu isprogramirati skriptu koja kontroliše PWM-om LED diodu. Naime, svakim pritiskom na taster koji je ugrađen u sistem, intenzitet LED diode treba da se uveća za 10%, sve dok se ne postigne da LED dioda svijetli maksimalnim intenzitetom. Tada, na sljedeći pritisak tastera LED dioda treba da se ugasi. Za implementaciju je bilo neophodno inicijalizirati GPIO, te podesiti način adresiranja pinova, a zatim se kroz `while(true)` petlju tražena funkcionalnost izvršava kontinuirano. Specifičnost ovog sistema jeste da je logika obrnuta, odnosno da za maksimalan duty cycle stavljamo vrijednost 0, a za minimalan (ugašenu diodu) stavljamo 100. Shodno tome je duty cycle i mijenjan na isti način da bi se postigao željeni efekt. Python skripta je kompajlirana i pokrenuta direktno na sistemu.

3 Korišteni hardverski resursi

Za potrebe ove laboratorijske vježbe korišten je Raspberri Pi model 2B, baziran na quad-core ARM Cortex-A7 procesoru koji radi na 1GHz, te sa 1GB RAM memorije. U sklopu istog je ugrađen taster, kao i RGB LED dioda.

- Rotacijski potencijometar
- DHT11 Digitalni senzor temperature i vlažnosti zraka

ULAZI	IZLAZI
Taster (digitalni, GPIO17)	RGB LED (digitalni; GPIO2 – GPIO4)



Slika 1 – Slabije podešenje duty cycle-a u 5. zadatku



Slika 2 – Jače podešenje duty cycle-a u 5. zadatku

4 Zaključak

Prilikom izvođenja ove laboratorijske vježbe nije bilo poteškoća. Korišten je potpuno nov sistem, međutim, rad sa konzolnim okruženjem Linux-baziranog operativnog sistema je relativno poznat od ranije, stoga nije bilo poteškoća u navigaciji direktorijama, pokretanju skripti, modificiranjem fajlova i sl. Cilj vježbe je bio upoznavanje sa novim sistemom Raspberry Pi 2B, što je uspješno i postignuto.

5 Prilog

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)
```