Univerzitet u Sarajevu

Elektrotehnički fakultet

**Ugradbeni sistemi 2023/24**

**Izvještaj za laboratorijsku vježbu br. 1**

Razvojni sistem picoETF

Ime i prezime: **Mirza Mahmutović**

Broj index-a: **19320**

12. mart 2024. godine

Sadržaj

[1 Pseudokod i/ili dijagram toka 1](#_Toc161731068)

[1.1 Zadatak 4 1](#_Toc161731069)

[1.2 Zadatak 5 1](#_Toc161731070)

[1.3 Zadatak za dodatne bodove 2 2](#_Toc161731071)

[2 Analiza programskog rješenja 3](#_Toc161731072)

[2.1 Zadatak 4 3](#_Toc161731073)

[2.2 Zadatak 5 3](#_Toc161731074)

[2.3 Dodatni zadatak 1 3](#_Toc161731075)

[2.4 Dodatni zadatak 2 4](#_Toc161731076)

[3 Korišteni hardverski resursi 4](#_Toc161731077)

[4 Zaključak 4](#_Toc161731078)

[5 Prilog 4](#_Toc161731079)

[5.1 Zadatak 4/izvorni kod 4](#_Toc161731080)

[5.2 Zadatak 5/izvorni kod 5](#_Toc161731081)

[5.3 Zadatak za dodatne bodove 2/izvorni kod 6](#_Toc161731082)

# Pseudokod i/ili dijagram toka

## Zadatak 4

**while** (**true**) **do**

LED7.value() = Taster4.value()

sleep(0.1)

**end\_while**

## Zadatak 5

c = 0

**while** (**true**) **do**

**if**(T1.pressed()) **then**

c = c + 1

**if**(c > 255) **then**

c = 0

**end\_if**

**while**(T1.pressed()) **do**

sleep(0.1)

**end\_while**

**end\_if**

**if**(T2.pressed()) **then**

c = c - 1

**if**(c < 0) **then**

c = 255

**end\_if**

**while**(T2.pressed()) **do**

sleep(0.1)

**end\_while**

**end\_if**

**if**(T3.pressed()) **then**

c = 0

**while**(T3.pressed()) **do**

sleep(0.1)

**end\_while**

**end\_if**

**if**(T4.pressed()) **then**

c = 255

**while**(T4.pressed()) **do**

sleep(0.1)

**end\_while**

**end\_if**

cBinary = c.toBinary()

**for** i = 0 **to** 7 **do**

**if**(cBinary[i] == 1) **then**

LED[7-i].on()

**else**

LED[7-i]\_OUT.off()

**end\_if**

**end\_for**

**end\_while**

## Zadatak za dodatne bodove 2

pins = [red, green, blue]

trajanje = 0.1

**while**(**true**) **do**

**while**(trajanje < 1.1) **do**

**for** lRGB = binary(1) to binary(7) **do**

pins.setValue(lRGB)

sleep(trajanje)

**end\_for**

trajanje += 0.1

**end\_while**

**while**(trajanje > 0.1) **do**

**for** lRGB = binary(1) to binary(7) **do**

pins.setValue(lRGB)

sleep(trajanje)

**end\_for**

trajanje -= 0.1

**end\_while**

**end\_while**

# Analiza programskog rješenja

## Zadatak 4

U zadatku je bilo potrebno kreirati prvu skriptu koja na osnovu stanja tastera T4 mijenja stanje diode LED7. While petlja u rješenju ovog zadatka omogućava neprekidno učitavanje stanja tastera T4, te postavljanja stanja diode LED7 na osnovu učitanog stanja tastera. Kod svakog mijenjanja tastera. Pored toga, u ovom zadatku smo se upoznali sa pokretanjem i prekidanjem programskog koda, koji je prethodno sačuvan u fajl.

## Zadatak 5

U zadatku je bilo potrebno implementirati 8-bitni binarni brojač uz pomoc LED dioda LED0 – LED7. Ovaj brojač je kontroliran od strane 4 tastera T1 – T4. Na početku samog programa je definisana funkciju *display()*, koja služi da se u svakom prolazu kroz *while* petlju na osnovu trenutne vrijednosti brojača *c* upale, odnosno ugase potrebne diode. Brojač *c* se inicijalizira na vrijednost 0 i ulazi se u beskonačnu *while* petlju. Zatim se provjerava koji od 4 tastera je pritisnut, te na osnovu njega mijenja vrijednost brojača *c*. Ukoliko se pritiskom na taster T1 pređe najveća moguća reprezentativna vrijednost sa 8 LED dioda (a to je 255), vrijednost brojača *c* se postavlja na 0. Analogno, ukoliko pritiskom na tasterom T2 vrijednost brojača *c* postane negativna, vrijednost se postavlja na 255. Zatim, uz pomoć *sleep()* metode vrši se blokiranje svih ostalih radnji sve dok se jedan od tastera drži. U ovom dijelu koda, uz pomoć unaprijed definisane funkcije *display()* provjerava se koje su cifre binarnog broja jednake jedinici, a koje nuli. Na osnovu tih vrijednosti se pale, odnosno gase, LED diode.

## Dodatni zadatak 1

U zadatku je bilo potrebno kroz REPL prompt testirati rad RGB diode. Ova dioda ima 3 osnovne boje, čijim kombiniranjem se može dobiti ukupno 7 različitih boja. 3 osnove boje (crvena, zelena i plava) su predstavljene varijablama *red*, *green* i *blue*, a njihovim kombinacijama se dobijaju boje na sljedeći način:

* crvena + zelena = tirkizna
* crvena + plava = ljubičasta
* zelena + plava = žuta
* crvena + zelena + plava = bijela

Sljedeći isječak koda predstavlja način na koji se može dobiti tirkizna boja:

red = Pin(14, Pin.OUT)

green = Pin(12, Pin.OUT)

blue = Pin(13, Pin.OUT)

red.value(1)

green.value(1)

## Dodatni zadatak 2

U zadatku je bilo potrebno omogućiti prolazak kroz sve moguće boje RGB LED, pri čemu je postepeno potrebno mijenjati trajanje svake od boja. U rješenju se u početku definiše niz, sačinjen od pinova, koji predstavljaju redom boje: crvena, zelena i plava. Zatim se u beskonačnoj *while* petlji izmjenjuju dvije *while* petlje, jedna za povećavanje trajanja, druga za smanjenje istog. U obe *while* petlje se korištenjem matrica i *zip()* funkcije postiže željeni efekat mijenjanja vrijednosti *red*, *green*, *blue* pinova, što dovodi do mijenjanja boja na samoj diodi.

# Korišteni hardverski resursi

Na ovoj vježbi je korišten razvojni sistem *picoETF,* baziran na mikrokontroleru RP2040. Sistem *picoETF* posjeduje razne elemente, a za ovu vježbu su korišteni:

* 8x LED
* 4x TASTERA
* 1x RGB LED

# Zaključak

Cilj ove vježbe je bio upoznavanje sa radom u laboratoriju i upoznavanje sa *picoETF* razvojnim sistemom. Pored toga, upoznali smo se sa *MicroPython* implementacijom programskog jezika Python 3 za mikrokontrolere, te sa razvojnim okruženjem *Thonny*. Kroz ovu vježbu smo naučili na koje načine je moguće implementirati, pokretati i prekidati programski kod, te smo kroz taj isti kod naučili neke osnovne načine za manipulaciju sa stanjima LED dioda i tastera. Unatoč prvom susretanju sa navedenim alatima, svi potrebni zadaci su uspješno urađeni.

# Prilog

## Zadatak 4/izvorni kod

from time import sleep

from machine import Pin

T4 = Pin(3, Pin.IN)

LED7 = Pin(11, Pin.OUT)

while True:

LED7.value(T4.value())

sleep(0.001)

## Zadatak 5/izvorni kod

from time import sleep

from machine import Pin

def display(x)

for i in range(0,8) :

if x[i] == „1“:

Pin(11-i,Pin.OUT).on()

else:

Pin(11-i,Pin.OUT).off()

c=0

while True:

if Pin(0,Pin.IN).value()== True :

c+=1

if(c > 255):

c = 0

while Pin(0,Pin.IN).value()== True :

sleep(0.1)

if Pin(1,Pin.IN).value()== True :

c-=1

if(c < 0):

c = 255

while Pin(1,Pin.IN).value()== True :

sleep(0.1)

if Pin(2,Pin.IN).value()== True :

c=0

while Pin(2,Pin.IN).value()== True :

sleep(0.1)

if Pin(3,Pin.IN).value()== True :

c = 255

while Pin(3,Pin.IN).value()== True :

sleep(0.1)

display('{0:08b}'.format(c))

## Zadatak za dodatne bodove 2/izvorni kod

from time import sleep

from machine import Pin

red = Pin(14, Pin.OUT)

green = Pin(12, Pin.OUT)

blue = Pin(13, Pin.OUT)

pins = [red, green, blue]

trajanje = 0.1

while True:

while trajanje < 1.1:

for lRGB in [[1, 0, 0],[0, 1, 0],[0, 0, 1],[1, 0, 1],[0, 1, 1],[1, 1, 0], [1, 1, 1]]:

for pin, value in zip(pins, lRGB):

pin.value(value)

sleep(trajanje)

trajanje += 0.1

while trajanje > 0.1:

for lRGB in [[1, 0, 0],[0, 1, 0],[0, 0, 1],[1, 0, 1],[0, 1, 1],[1, 1, 0], [1, 1, 1]]:

for pin, value in zip(pins, lRGB):

pin.value(value)

sleep(trajanje)

trajanje -= 0.1