

Univerzitet u Sarajevu  
Elektrotehnički fakultet  
**Ugradbeni sistemi 2023/24.**

**Izvještaj za laboratorijsku vježbu 4**  
Analogni ulazi i displeji

Ime i prezime: **Kanita Kadušić**  
Broj index-a: **19327**

Sarajevo, april 2024.

# Sadržaj

<b>1</b>	<b>Pseudokod</b>	<b>1</b>
1.1	Zadatak 1	1
1.2	Zadatak 2	2
1.3	Zadatak 3	2
<b>2</b>	<b>Analiza programskog rješenja</b>	<b>3</b>
2.1	Zadatak 1	3
2.2	Zadatak 2	3
2.3	Zadatak 3	3
<b>3</b>	<b>Korišteni hardverski resursi</b>	<b>4</b>
3.1	LPC1114ETF	4
3.2	picoETF	4
<b>4</b>	<b>Povezivanje komponenti sa razvojnim sistemima</b>	<b>5</b>
4.1	Zadatak 1	5
4.2	Zadatak 2	6
4.3	Zadatak 3	7
<b>5</b>	<b>Zaključak</b>	<b>8</b>
<b>6</b>	<b>Prilog</b>	<b>9</b>
6.1	Zadatak 1: Izvorni kôd	9
6.2	Zadatak 2: Izvorni kôd	10
6.3	Zadatak 3: Izvorni kôd	12

# 1 Pseudokod

## 1.1 Zadatak 1

```
diodes ← outputCollection(4, 5, 6, 7, 8, 9, 10, 11)  
seconds ← analogInput(28)
```

```
while true
```

```
  while  $i \leftarrow 1 \neq 256$  factor 2  
    value(diodes) ← i  
    pause(value(seconds))
```

```
  value(diodes) ← 255  
  pause(value(seconds))
```

```
  while  $i \leftarrow 128 \neq 0$  factor 0.5  
    value(diodes) ←  $i - 1$   
    pause(value(seconds))
```

```
  value(diodes) ← 0  
  pause(value(seconds))
```

## 1.2 Zadatak 2

```
adc ← analogInput(28)
timeStart ← value(time)

while true
  wastingStart ← value(time)

  voltage ← (value(adc) ÷ 65535) · 3300
  temperature ← voltage ÷ 10
  duration ← value(time) – timeStart

  managePosition()
  displayPrint(voltage, temperature, duration)

  managePosition()
  displayPrint('.')

  wasted ← value(time) – wastingStart
  pause(1 – wasted)
```

## 1.3 Zadatak 3

```
act ← output(LED_ACT)
pm ← analogInput(AD1)
leds ← outputCollection(LED0, LED1, LED2, LED3, LED4, LED5, LED6, LED7)

function lights(x)
  p ← 1 ÷ 8
  values ← {255, 127, 63, 31, 15, 7, 3, 1, 0}

  for i, j in 0..8, 1..9
    if  $x \geq a \cdot p$  and  $x < b \cdot p$ 
      return values[a]

  return 0

value(act), value(leds) ← 0

while true
  value(leds) ← lights(value(pm))
  pause(0.5)
```

## **2 Analiza programskog rješenja**

### **2.1 Zadatak 1**

[1-2] uvoz odgovarajućih biblioteka

[4-15] klasa koja olakšava rad s proizvoljnim brojem digitalnih izlaza (njen interfejs sadrži metodu za čitanje i pisanje)

[17-24] klasa koja olakšava rad s potencijometrom (njen interfejs sadrži metodu za čitanje)

[26-27] deklaracija i inicijalizacija potrebnih varijabli

[30-34] realizacija sekvence brojeva 1, 2, 4, 8, 16, 32, 64, 128 (stepeni dvojke)

[39-43] realizacija sekvence brojeva 127, 63, 31, 15, 7, 3, 1, 0 (stepeni dvojke umanjeni za 1)

[33/37/42/46] pauza u izvršavanju programa u trajanju od očitano broj sekundi

### **2.2 Zadatak 2**

[1-31] uvoz odgovarajućih biblioteka, te deklaracija i inicijalizacija potrebnih varijabli

[34-40] iscrtavanje koordinatnih osa

[43-50] prikaz vrijednosti 20, 30 i 40 na y-osi

[58] konverzija očitavanja ADC-a u stvarni napon

[61] računanje temperature

[64] vrijeme proteklo od početka izvršavanja programa u sekundama

[68-70/74] odgovarajući ispis na displej

[66-67/72-73] pozicioniranje ispisa na displeju i odabir željenog fonta

[77] pauza u izvršavanju programa u trajanju od izračunatog broja sekundi

### **2.3 Zadatak 3**

[1-7] uvoz odgovarajućih biblioteka, te deklaracija i inicijalizacija potrebnih varijabli

[9-18] funkcija koja „traži“ pripadajući interval za proslijeđeni parametar, te vraća odgovarajuću vrijednost kako bi se upalile potrebne diode

[25] očitavanje napona, poziv navedene funkcije, te uključivanje LED dioda

[26] pauza u izvršavanju programa u trajanju od 0.2 sekunde

### 3 Korišteni hardverski resursi

#### 3.1 LPC1114ETF

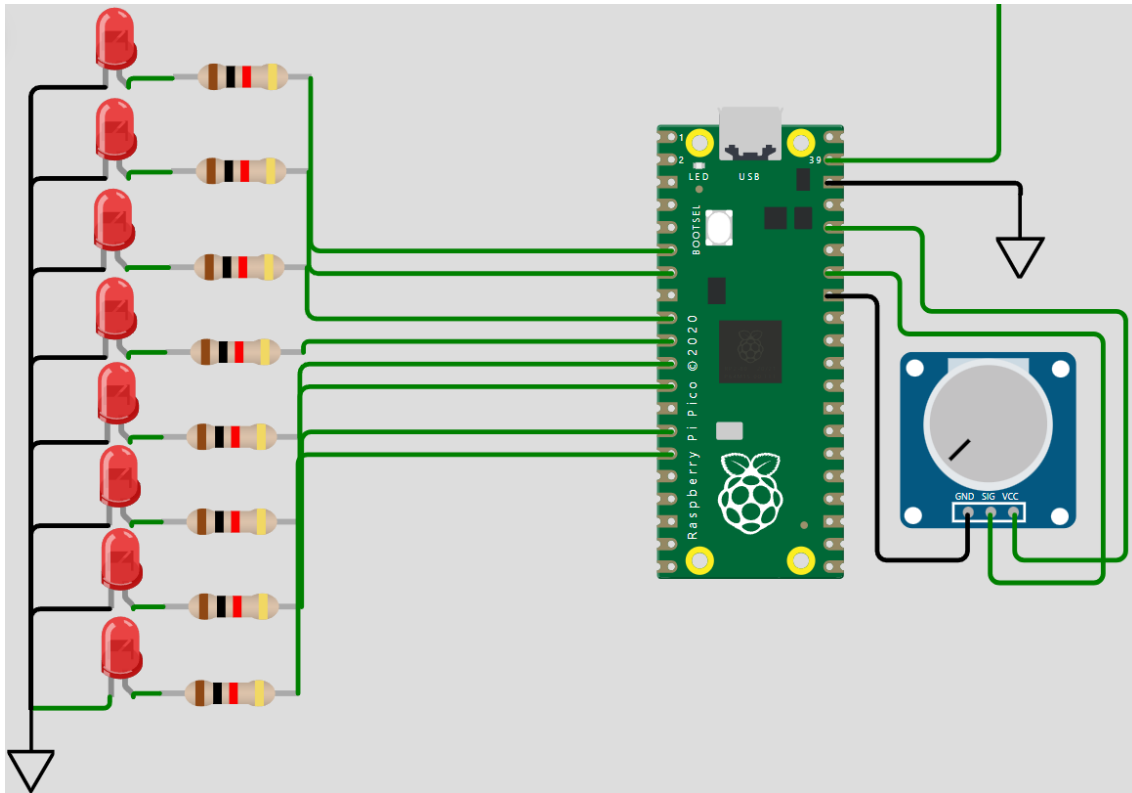
	<i>Komponenta</i>	<i>Opis</i>	<i>Količina</i>
1	LPC1114FN28	mikrokontroler	1
2	LED dioda	digitalni izlaz	8
3	potenciometar	ulazna komponenta	1
4	konektor	uvezivanje sistema	
5	USB A kabal	napajanje i komunikacija	1
6	LED_ACT pin	uključivanje LED dioda	1

#### 3.2 picoETF

	<i>Komponenta</i>	<i>Opis</i>	<i>Količina</i>
1	RP2040	mikrokontroler	1
2	LED dioda	digitalni izlaz	8
3	potenciometar	ulazna komponenta	1
4	Banggood displej	izlazna komponenta	1
5	LM35 temperaturni senzor	ulazna komponenta	1
6	konektor	uvezivanje sistema	
7	USB A – USB Micro	napajanje i komunikacija	1

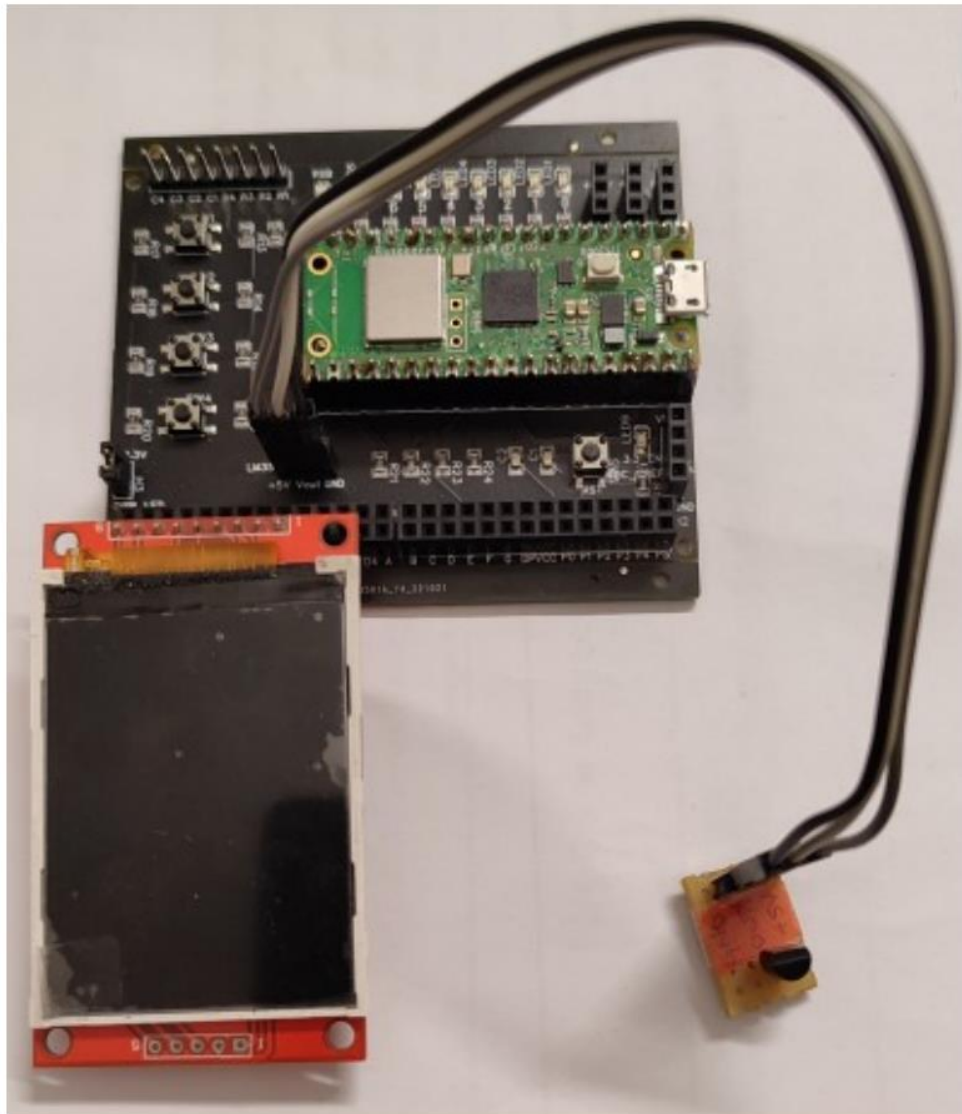
## 4 Povezivanje komponenti sa razvojnim sistemima

### 4.1 Zadatak 1



**Slika 1:** Povezivanje potenciometra na picoETF

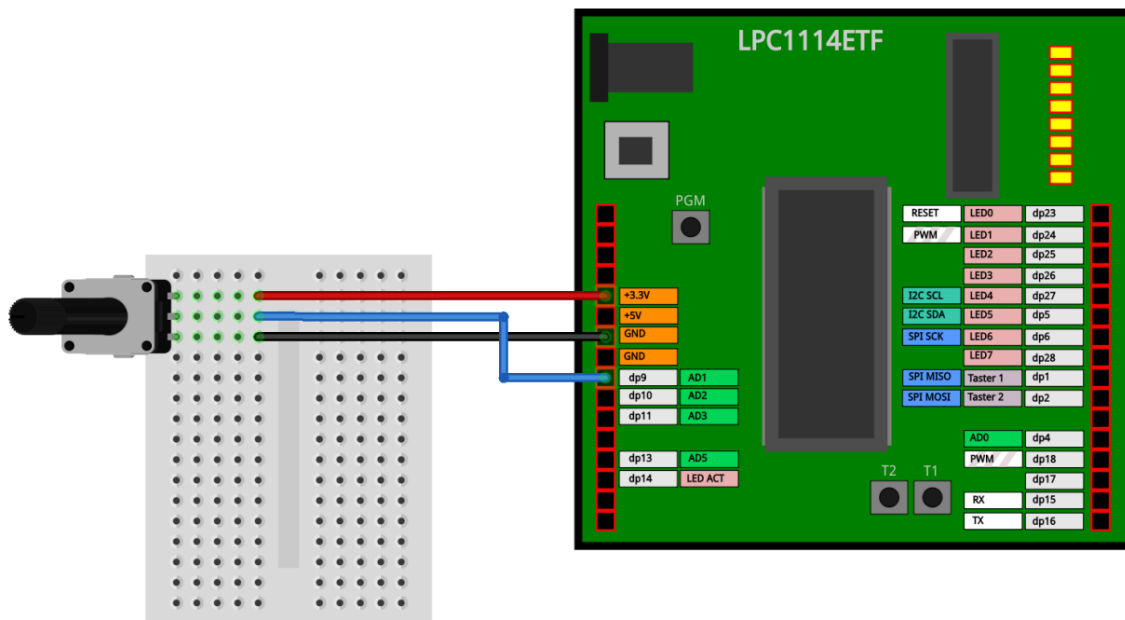
## 4.2 Zadatak 2



**Slika 2:** Povezivanje senzora LM35DZ sa sistemom picoETF



### 4.3 Zadatak 3



**Slika 3:** Povezivanje potenciometra na LPC1114ETF

## **5      Zaključak**

Zadaci u okviru Laboratorijske vježbe 4 su bili zanimljivi, ali je namještanje koordinata za ispis na displejima prilično dosadan posao.

Prvi i treći zadatak se rade rutinski, dok je drugi zadatak zahtijevao mnogo više vremena.

Nova znanja, stečena izradom Laboratorijske vježbe 4, obuhvataju rad s analognim ulazima i dodatnim hardverskim resursima, konkretno, potencijometrom, temperaturnim senzorom i displejima.

## 6 Prilog

### 6.1 Zadatak 1: Izvorni kôd

```
01: from machine import Pin, ADC
02: from time import sleep
03:
04: class BusOut:
05:     def __init__(self, *pins):
06:         self.pins = [ Pin(pin, Pin.OUT) for pin in pins ]
07:         self.value = 0
08:
09:     def write(self, value):
10:         self.value = value
11:         for i, pin in enumerate(self.pins):
12:             pin.value((value >> i) & 1)
13:
14:     def read(self):
15:         return self.value
16:
17: class Potentiometer:
18:     def __init__(self, pin, a = 0, b = 3.3):
19:         self.pin = ADC(Pin(pin, Pin.IN))
20:         self.a = a
21:         self.b = b
22:
23:     def read(self):
24:         return (self.b - self.a) * self.pin.read_u16() / 65535 + self.a
25:
26: leds = BusOut(4, 5, 6, 7, 8, 9, 10, 11)
27: seconds = Potentiometer(28, 0.1, 2)
28:
29: while True:
30:     i = 1
31:     while i != 256:
32:         leds.write(i)
33:         sleep(seconds.read())
34:         i = i * 2
35:
36:     leds.write(255)
37:     sleep(seconds.read())
38:
39:     i = 128
40:     while i != 0:
41:         leds.write(i - 1)
42:         sleep(seconds.read())
43:         i = i // 2
44:
45:     leds.write(0)
46:     sleep(seconds.read())
```

## 6.2 Zadatak 2: Izvorni kôd

```
01: from time import sleep, time
02: from machine import Pin, ADC, SPI
03: from math import log
04:
05: # Uvoz preporucenih biblioteka
06: from ili934xnew import ILI9341, color565
07: import tt14
08: import tt32
09:
10: power = Pin(22, Pin.OUT)
11: power.value(1)
12:
13: adc = ADC(Pin(28))
14:
15: spi = SPI(
16:     0,
17:     baudrate = 40000000,
18:     miso = Pin(16),
19:     mosi = Pin(19),
20:     sck = Pin(18)
21: )
22:
23: display = ILI9341(
24:     spi,
25:     cs = Pin(17),
26:     dc = Pin(15),
27:     rst = Pin(20),
28:     w = 320,
29:     h = 240,
30:     r = 3
31: )
32:
33: # Iscrtavanje koordinatnih osa
34: for i in range(0, 320, 10):
35:     display.set_pos(i, 230)
36:     display.print('-')
37:
38: for i in range(0, 240, 10):
39:     display.set_pos(0, i)
40:     display.print('|')
41:
42: # Prikaz vrijednosti na y-osi
43: display.set_pos(7, (235 - 40 + (20 - 40) * 7))
44: display.print('40')
45:
46: display.set_pos(7, (235 - 30 + (20 - 30) * 7))
47: display.print('30')
48:
49: display.set_pos(7, (235 - 20 + (20 - 20) * 7))
50: display.print('20')
51:
```

```

52: timeStart = time()
53:
54: while True:
55:     wastingStart = time()
56:
57:     # Konverzija ocitanja ADC-a u stvarni napon
58:     voltage = round((adc.read_u16() / 65535) * 3300)
59:
60:     # Racunanje temperature
61:     temp = voltage / 10
62:
63:     # Vrijeme proteklo od pocetka izvorsavanja programa u sekundama
64:     duration = time() - timeStart
65:
66:     display.set_pos(220, 0)
67:     display.set_font(tt14)
68:     display.print('Napon: ' + str(voltage) + 'mV')
69:     display.print('Temp: ' + str(temp) + 'C')
70:     display.print('Vrijeme: ' + str(duration) + 's')
71:
72:     display.set_pos(int(duration * 10) + 30, int((235 - temp + (20 - temp) * 7)))
73:     display.set_font(tt32)
74:     display.print('.')
75:
76:     wasted = time() - wastingStart
77:     sleep(1 - wasted)

```

## 6.3 Zadatak 3: Izvorni kôd

```
01: #include "mbed.h"
02: #include "lpc1114etf.h"
03: #include "BusOut.h"
04:
05: DigitalOut act(LED_ACT);
06: AnalogIn PM(AD1);
07: BusOut leds(LED0, LED1, LED2, LED3, LED4, LED5, LED6, LED7);
08:
09: int lights(float x) {
10:     const float p = 1. / 8;
11:     int values[] = {255, 127, 63, 31, 15, 7, 3, 1, 0};
12:
13:     for (int a = 0, b = 1; a <= 8; a++, b++)
14:         if (x >= a * p && x < b * p)
15:             return values[a];
16:
17:     return 0;
18: }
19:
20: int main() {
21:     act.write(0);
22:     leds.write(0);
23:
24:     for (;;) {
25:         leds.write(lights(PM.read()));
26:         wait_us(2e5);
27:     }
28: }
```