

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

**Izvještaj za laboratorijsku vježbu 5**  
Analogni izlazi i širinsko-impulsna modulacija (PWM)

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 .....	3
<b>2</b>	<b>Analiza programskog rješenja</b> .....	<b>4</b>
2.1	Zadatak 1 .....	4
2.2	Zadatak 2 .....	4
2.3	Zadatak 3 .....	4
<b>3</b>	<b>Korišteni hardverski resursi</b> .....	<b>5</b>
3.1	LPC1114ETF .....	5
3.2	picoETF .....	5
3.3	FRDM-KL25Z .....	6
<b>4</b>	<b>Povezivanje komponenti sa razvojnim sistemima</b> .....	<b>7</b>
4.1	Zadatak 1 .....	7
4.2	Zadatak 2 .....	8
4.3	Zadatak 3 .....	9
<b>5</b>	<b>Zaključak</b> .....	<b>10</b>
<b>6</b>	<b>Prilog</b> .....	<b>11</b>
6.1	Zadatak 1: Izvorni kôd .....	11
6.2	Zadatak 2: Izvorni kôd .....	12
6.3	Zadatak 3: Izvorni kôd .....	13

# 1 Pseudokod

## 1.1 Zadatak 1

```
act ← output(LED_ACT)  
button1, button2 ← input(Taster_1), input(Taster_2)  
pm ← analogInput(AD1)  
led ← pwm(LED1)  
  
value(act) ← 0  
  
while true  
  if value(button1) = 1  
    period(led) ←  $50 \cdot 10^{-6}$   
  else if value(button2) = 1  
    period(led) ←  $500 \cdot 10^{-3}$   
  
  value(led) ← value(pm)  
  pause(0.2)
```

## 1.2 Zadatak 2

```
leds ← {pwm(4..11)}  
ldr ← analogInput(28)  
  
for led in leds  
  frequency(led) ← 10000  
  
function lights(x)  
  p ← 65535 ÷ 8  
  
  for i in 0..8  
    if  $x \geq i \cdot p$  and  $x < (i + 1) \cdot p$   
      for j in 0..(i - 1)  
        duty(leds[j]) ←  $\text{int}(x \div 2^{i-j-1})$   
      for j in i..7  
        duty(leds[j]) ← 0  
      break  
  
while true  
  lights(value(ldr))  
  pause(0.02)
```

### 1.3 Zadatak 3

```
signal ← analogOutput(PTE30)
```

```
function signal1()  
  for i in 0..49  
    value(signal) ← 0.5 · sin(i ÷ 8) + 0.5
```

```
function signal2()  
  for i in 0..1  
    for j in 0..24  
      value(signal) ← j ÷ 25
```

```
function signal3()  
  for i in 0..1  
    for j in 25..1 step −1  
      value(signal) ← j ÷ 25
```

```
function signal4()  
  for i in 0..1  
    for j in 13..1 step −1  
      value(signal) ← j ÷ 13  
    for j in 1..12  
      value(signal) ← j ÷ 12
```

```
function signal5()  
  for i in 0..49  
    value(signal) ← |sin(i ÷ 8)|
```

```
while true  
  signal2()
```

## **2 Analiza programskog rješenja**

### **2.1 Zadatak 1**

- [1-7] uvoz odgovarajućih biblioteka, te deklaracija i inicijalizacija potrebnih varijabli
- [13] ukoliko je pritisnut prvi taster, postavljanje vrijednosti PWM perioda na  $50\mu s$
- [14] ukoliko je pritisnut drugi taster, postavljanje vrijednosti PWM perioda na  $500ms$
- [16] očitavanje vrijednosti potencijometra i postavljanje vrijednosti PWM-a
- [17] pauza u izvršavanju programa

### **2.2 Zadatak 2**

- [1-5] uvoz odgovarajućih biblioteka, te deklaracija i inicijalizacija potrebnih varijabli
- [7-8] postavljanje frekvencije PWM varijabli
- [10-19] funkcija koja, na osnovu datog ulaza (očitanje vrijednosti sa analognog ulaza), realizira *VU metar*, pritom gradijentno raspoređujući intenzitete na 8 LED dioda
- [22] očitavanje vrijednosti fotootpornika i poziv navedene funkcije
- [23] pauza u izvršavanju programa

### **2.3 Zadatak 3**

- [1-4] uvoz odgovarajućih biblioteka, te deklaracija i inicijalizacija potrebnih varijabli
- [6-47] realizacija različitih signala
- [50] poziv funkcije za prikaz željenog signala

### 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

#### 3.2 picoETF

	<i>Komponenta</i>	<i>Opis</i>	<i>Količina</i>
1	RP2040	mikrokontroler	1
2	LED dioda	digitalni izlaz	8
3	fotootpornik	ulazna komponenta	1
4	otpornik	ulazna komponenta	1
5	konektor	uvezivanje sistema	
6	USB A – USB Micro	napajanje i komunikacija	1

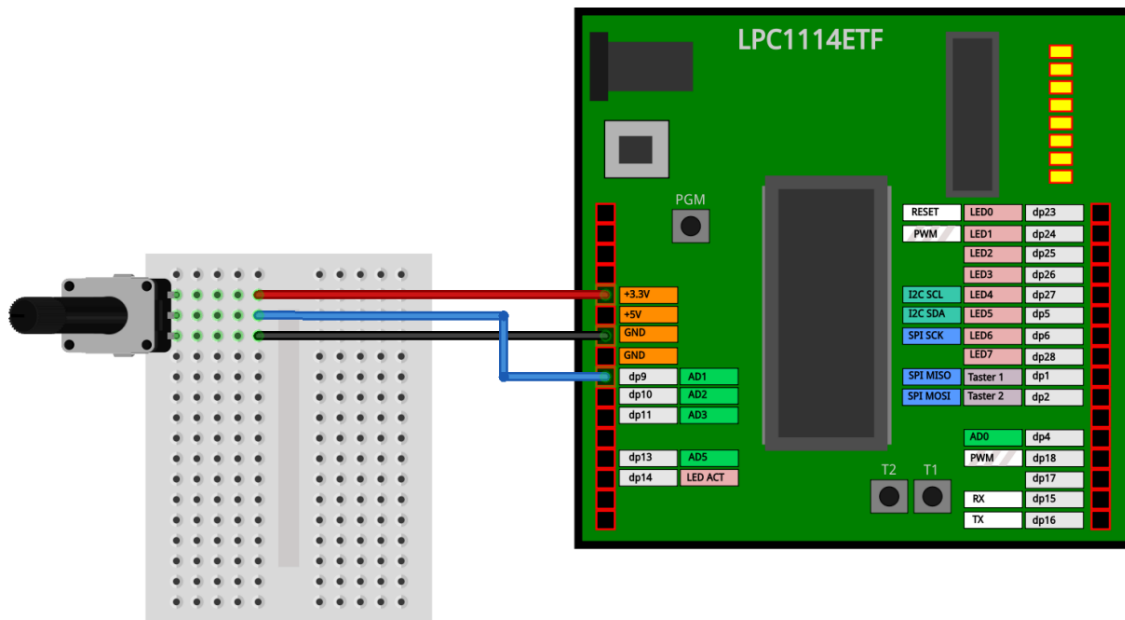
### 3.3 FRDM-KL25Z

	<i>Komponenta</i>	<i>Opis</i>	<i>Količina</i>
1	KL25Z128VLK	mikrokontroler	1
2	osciloskop	izlazna komponenta	1
3	sonda	uvezivanje sistema	1
4	konektor	uvezivanje sistema	
5	USB A – USB Mini B	napajanje i komunikacija	1



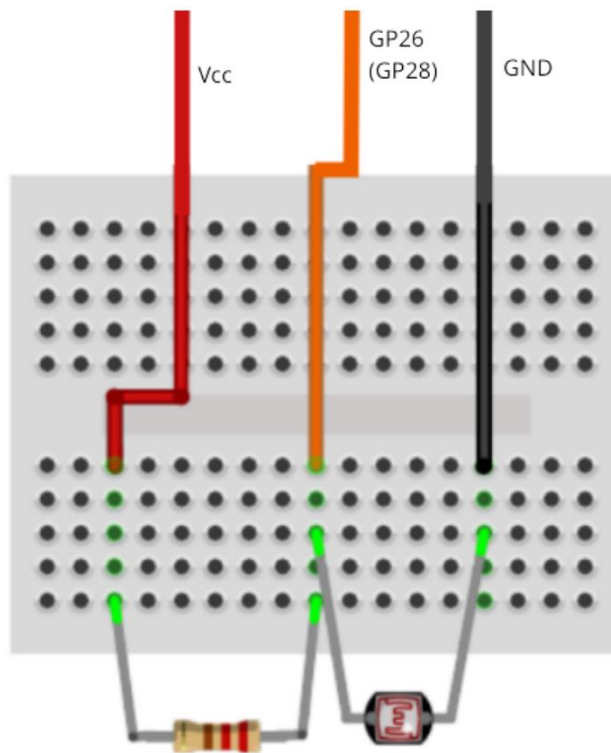
## 4 Povezivanje komponenti sa razvojnim sistemima

### 4.1 Zadatak 1



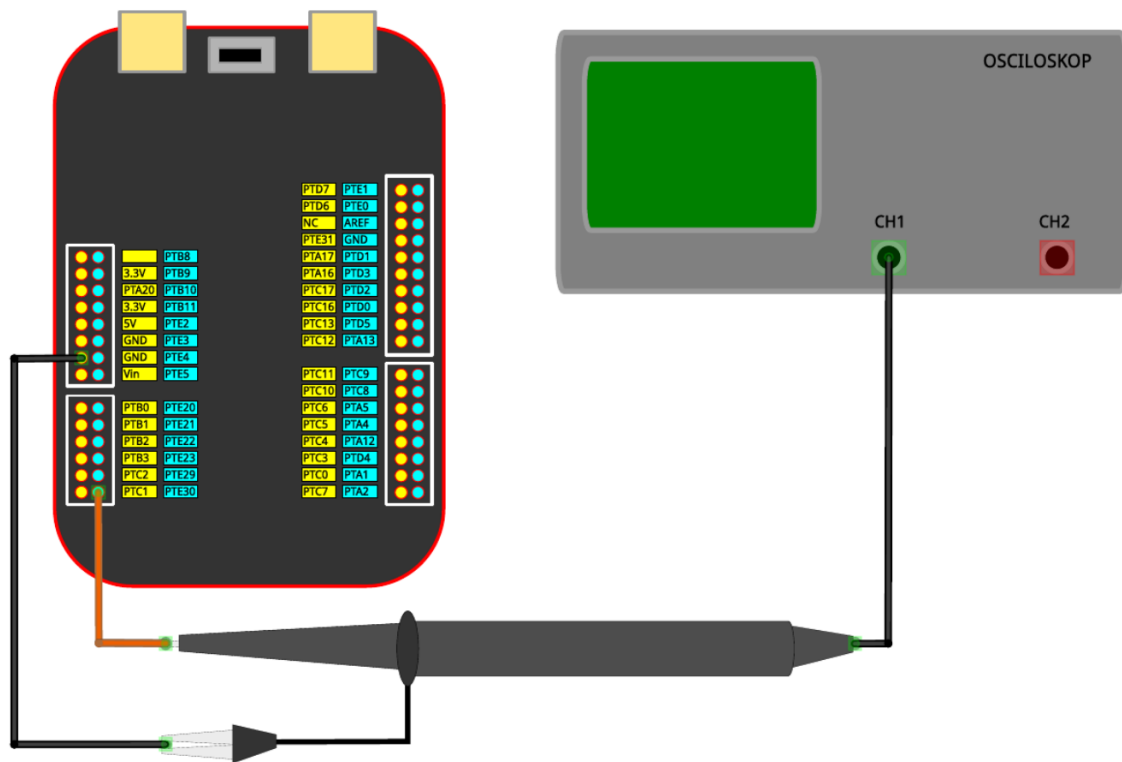
Slika 1: Povezivanje potencijometra sa razvojnim sistemom LPC1114ETF

## 4.2 Zadatak 2



**Slika 2:** Povezivanje fotootpornika sa razvojnim sistemom picoETF

### 4.3 Zadatak 3



Slika 3: Povezivanje osciloskopa sa razvojnim sistemom FRDM-KL25Z

## 5 Zaključak

Zadaci u okviru Laboratorijske vježbe 5 su bili zanimljivi, a hardverska realizacija tokom laboratorijske vježbe je protekla uredno i bez problema.

U okviru Zadatka 1 je bilo potrebno uočiti i analizirati promjenu osvjetljenja LED diode za PWM period od  $T = 50\mu s$  i  $T = 500ms$ . Manifestirana razlika je bila lako uočljiva. U slučaju većeg perioda, konkretno  $T = 500ms$ , dolazi do treperenja, odnosno, može se primijetiti kako se LED dioda uključuje i isključuje. Takvo nešto za slučaj manjeg perioda, konkretno  $T = 50\mu s$ , nije bilo uočeno, odnosno promjena u intenzitetu osvjetljenja se dešavala glatko.

Zadatak 3 je bio zanimljiv s aspekta analize vremena izvršavanja programa. Naime, u zadatku je definisana širina jedne „stepenice“ u trajanju od  $40\mu s$ . Očekivano je, ali ipak iznenađujuće u kojoj mjeri je potrebno smanjiti vrijeme zaustavljanja programa, kako bi se ostvarila definisana frekvencija od  $500Hz$ . Takvo ponašanje je bilo najlakše uočiti prilikom realizacije 1. i 5. signala, s obzirom da se koristi poziv funkcije *sin*, kao i da se vrši necjelobrojno dijeljenje.

Ipak, treba napomenuti da, iako je svaki signal napisan u svega par linija kôda, je moguće postići bolju vremensku efikasnost. Recimo, ukoliko se funkcija unaprijed uzorkuje u 25 tačaka, a njene vrijednosti pohrane u niz, sve što preostaje je pristupati tim vrijednostima, najbolje pokazivačkom aritmetikom.

Za kraj, nova znanja stečena u okviru Laboratorijske vježbe 5 podrazumijevaju rad s analognim izlazima i PWM-om, te s novim hardverskim komponentama, konkretno, fotootpornikom i osciloskopom.

## 6 Prilog

### 6.1 Zadatak 1: Izvorni kôd

```
01: #include "mbed.h"
02: #include "lpc1114etf.h"
03:
04: DigitalOut act(LED_ACT);
05: DigitalIn button1(Taster_1), button2(Taster_2);
06: AnalogIn pm(AD1);
07: PwmOut led(LED1);
08:
09: int main() {
10:     act.write(0);
11:
12:     for (;;) {
13:         if (button1.read()) led.period_us(50);
14:         else if (button2.read()) led.period_ms(500);
15:
16:         led.write(pm.read());
17:         wait_us(2e5);
18:     }
19: }
```

## 6.2 Zadatak 2: Izvorni kôd

```
01: from machine import Pin, PWM, ADC
02: from time import sleep
03:
04: leds = [ PWM(Pin(i)) for i in range(4, 12) ]
05: ldr = ADC(Pin(28))
06:
07: for led in leds:
08:     led.freq(10000)
09:
10: def lights(x):
11:     p = (65007 - 512) / 8
12:
13:     for i in range(0, 9):
14:         if x >= i * p and x < (i + 1) * p:
15:             for j in range(0, i):
16:                 leds[j].duty_u16(x // (2 ** (i - j - 1)))
17:             for j in range(i, 8):
18:                 leds[j].duty_u16(0)
19:             break
20:
21: while True:
22:     lights(ldr.read_u16())
23:     sleep(0.02)
```

## 6.3 Zadatak 3: Izvorni kôd

```
01: #include "mbed.h"
02: #include "cmath"
03:
04: AnalogOut signal(PTE30);
05:
06: void signal1() {
07:     for (int i = 0; i < 50; i++) {
08:         signal.write(0.5 * std::sin(i / 8.) + 0.5);
09:         wait_ns(10);
10:     }
11: }
12:
13: void signal2() {
14:     for (int i = 0; i < 2; i++)
15:         for (int j = 0; j < 25; j++) {
16:             signal.write(j / 25.f);
17:             wait_us(20);
18:         }
19: }
20:
21: void signal3() {
22:     for (int i = 0; i < 2; i++)
23:         for (int j = 25; j > 0; j--) {
24:             signal.write(j / 25.f);
25:             wait_us(20);
26:         }
27: }
28:
29: void signal4() {
30:     for (int i = 0; i < 2; i++) {
31:         for (int j = 13; j > 0; j--) {
32:             signal.write(j / 13.f);
33:             wait_us(20);
34:         }
35:         for (int j = 1; j < 13; j++) {
36:             signal.write(j / 12.f);
37:             wait_us(20);
38:         }
39:     }
40: }
41:
42: void signal5() {
43:     for (int i = 0; i < 50; i++) {
44:         signal.write(std::fabs(std::sin(i / 8.)));
45:         wait_ns(10);
46:     }
47: }
48:
49: int main() {
50:     for (;;) signal2();
51: }
```