****

**VILNIUS UNIVERSITY  
SIAULIAI ACADEMY**

PROGRAMŲ SISTEMOS BACHELOR STUDY PROGRAMME

Software engineering

**ANNA KUTOVA**

**Programming of Embedded Systems**

**Laboratory work No.1**

**I/O Ports**

Šiauliai, 2025

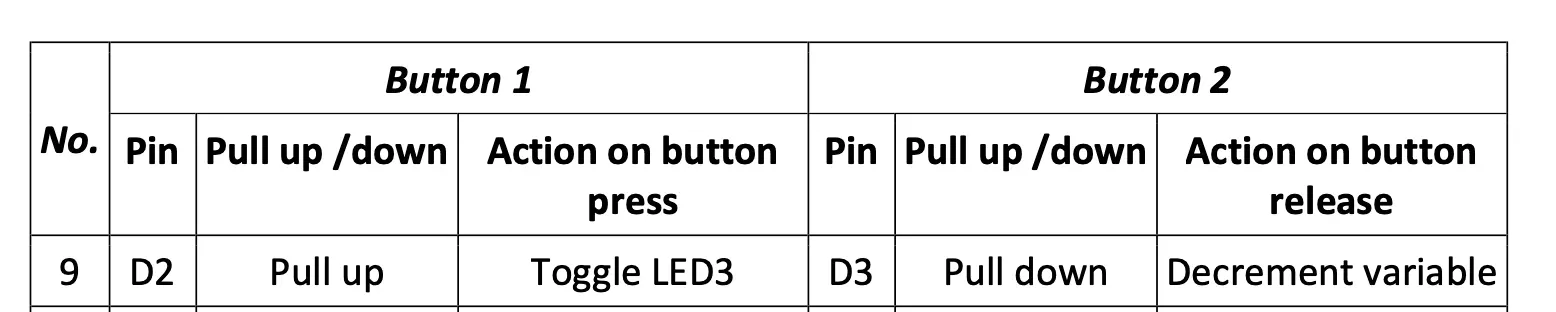
### **Laboratory Work Report**

#### **1. The Aim of the Laboratory Work**

The aim of this laboratory work is to develop a MicroPython program for the NUCLEO\_F756ZG microcontroller that processes input signals from buttons and performs actions based on state changes. The program should detect transitions between logical levels (1 to 0 and 0 to 1) and execute specific operations accordingly.

#### **2. Variant No and Data**

**Variant No:** 9



**Date:** 13/02/2025

#### 

#### **3. Program Algorithm**

#### 

#### **4. Program Body with Comments**

**from** **machine** **import** Pin

**import** **time**

**import** **pyb**

*# Initialise input pins*

pin\_D2 = Pin("D2", Pin.IN, Pin.PULL\_UP) *# Button with pull-up resistor*

pin\_D3 = Pin("D3", Pin.IN, Pin.PULL\_DOWN) *# Button with pull-down resistor*

x = 100 *# Initial value*

state = "INITIAL"

*# Store previous states of the buttons*

prev\_D2 = pin\_D2.value()

prev\_D3 = pin\_D3.value()

**while** True:

*# Read current states*

current\_D2 = pin\_D2.value()

current\_D3 = pin\_D3.value()

*# Detect changes in D2*

**if** current\_D2 != prev\_D2:

**if** current\_D2 == 0:

state = "BUTTON PRESSED"

pyb.LED(1).toggle() *# Toggle LED when button is pressed*

**else**:

state = "BUTTON RELEASED"

**print**(f"State: {state}, x: {x}")

*# Detect changes in D3*

**if** current\_D3 != prev\_D3:

**if** current\_D3 == 0:

x -= 2 *# Decrease x only on state change*

**print**(f"State: {state}, x: {x}")

*# Update previous states*

prev\_D2 = current\_D2

prev\_D3 = current\_D3

*# Small delay to reduce signal bouncing*

time.sleep\_ms(50)

**5. Conclusions**

In this laboratory work, a MicroPython program was successfully developed for the NUCLEO\_F756ZG microcontroller to process button inputs based on state changes. The program correctly detects transitions between logical levels and performs actions accordingly. This approach prevents continuous execution of the same action while a button is held down. The use of a small delay helps minimize signal bouncing. The experiment demonstrated how to work with GPIO, implement state tracking, and toggle LED outputs.