Université d'Ottawa
Faculté de génie

École de science informatique
et de génie électrique

L'Université canadienne
Canada's university

University of Ottawa
Faculty of Engineering

School of Electrical Engineering
and Computer Science

CSI4141/SEG4155 Real Time System Design
Winter 2025

To be submitted via https://uottawa.brightspace.com/ by January 24 at 11:59 p.m.

Lab1: Interactive traffic lights

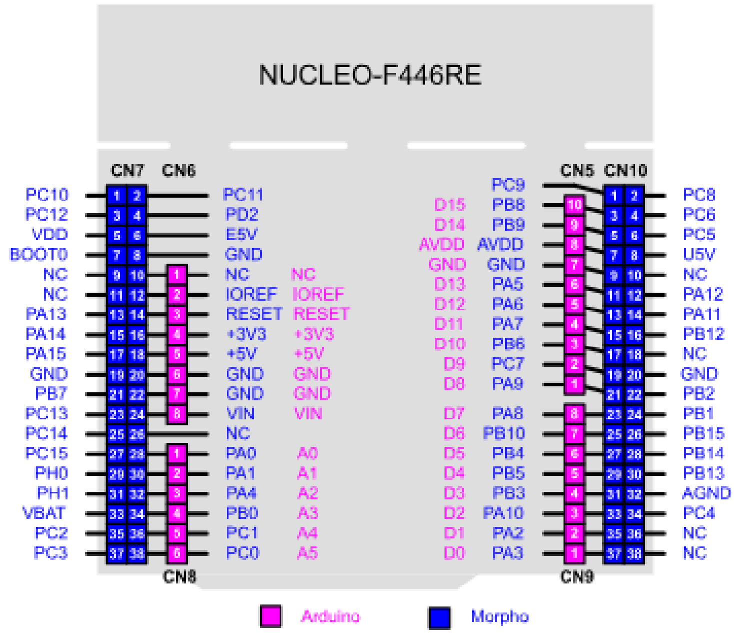
1. Objective

In this lab, students will create different tasks with the freeRTOS real-time kernel, using the STM32CubeIDE integrated development environment to implement an interactive traffic light system.

2. Development

In this laboratory, students will use the nucleo-F446RE development board shown below. Among other features, this board is equipped with a user push-button and a reset push-button. The nucleo-F446RE board also features GPIOs for connecting LEDs to simulate interactive traffic lights. To protect the LEDs, we recommend connecting a resistor of 230 to 470 ohms between the nucleo-f446RE board's GPIOS and the LEDs.

Note that this application can be realized with a single task, but it is mandatory to use one task for each LED and one for the push-button (for a total of 6 tasks), in order to become familiar with the task concept. To carry out Lab 1, students can use the file FirstProjectFreeTROS.7z, which contains a starter project. In this application, three tasks have been created to flash and send a message on the serial port. USAR2 is used to send the messages and GPIOs PB3, PB4 and PB5 to flash the three LEDs on the nucleo-F446RE board. The following figure shows the GPIOs on the board.



Part 1

In the first part of the lab, you'll light up three light-emitting diodes. They won't all light up at the same time, of course! In fact, you'll be building traffic lights. Control is in phases. Then, in the second part, you'll complete your traffic lights with pedestrian lights. First, let's take a look at the list of components.

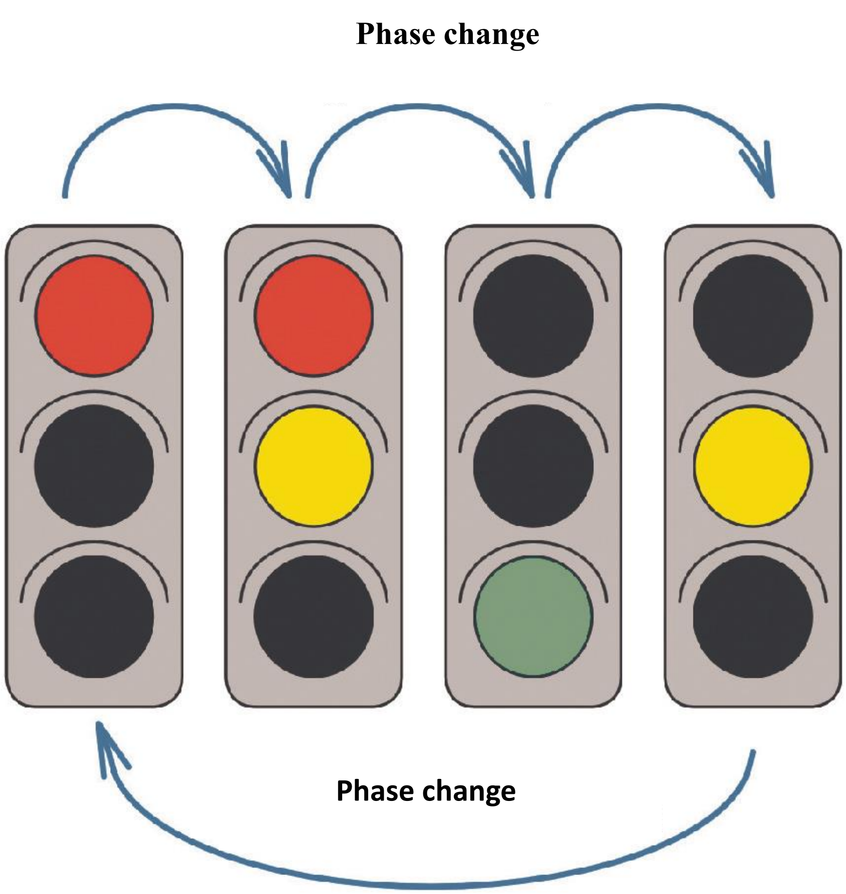
Required components.

This assembly requires the following components.

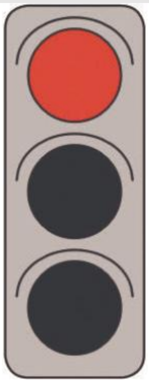
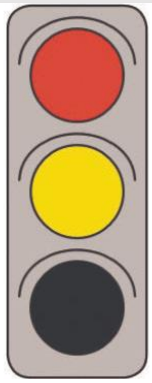
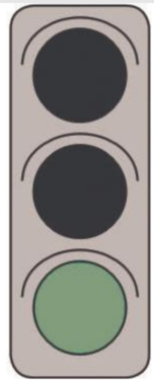
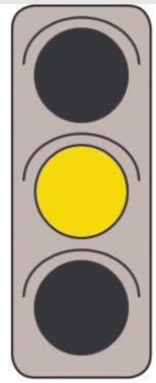
Component	
2 red LEDs	
1 orange LED	
2 green LEDs	
5 resistors of the order of hundreds of ohms ≈ 330 Ω	
1 push-button	Blue card push-button

Signaling phases

First, let's look at the different possible signaling phases:

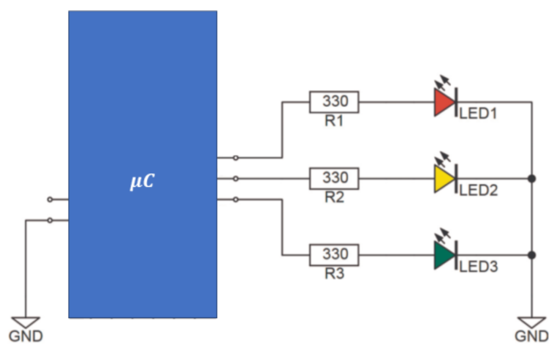


The individual phases are scrolled through from left to right, with the cycle then starting again from the beginning. Each phase has a defined display duration that can be adjusted individually. Here's an example of switch-on times.

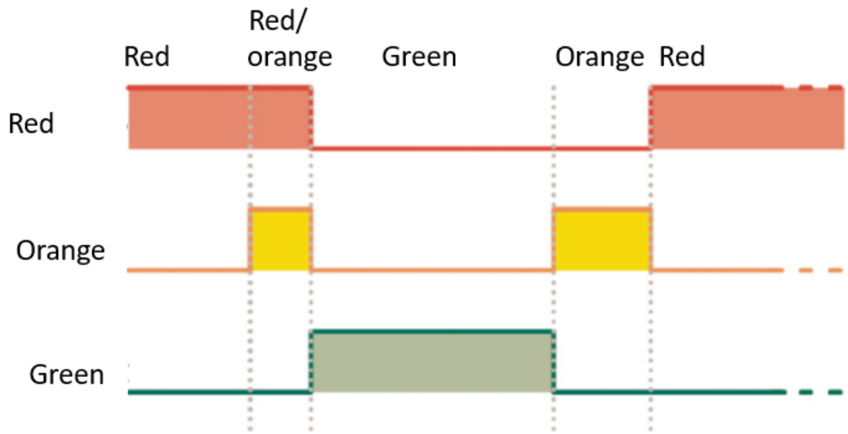
Phase 1	**Phase 2**	**Phase 3**	**Phase 4**
			
Duration: 10 seconds	Duration: 2 seconds	Duration: 10 seconds	Duration: 3 seconds

Diagram

The LEDs should be mounted as shown in the following figure. Each LED is controlled by a 330 Ω series resistor.



For the traffic lights to work properly, you have to remember not only to switch the various LEDs on, but also to switch them off. Processing takes place continuously, from top to bottom, within each task. When moving from phase 1 to phase 2, only one orange LED is added to the red one. The red one continues to shine. But when you move from phase 2 to phase 3, make sure that the red and orange LEDs go out before the green LED lights up. Then, at the transition from phase 4 to phase 1, when the phases start again from the beginning, the orange LED should go out. Take a look at the timeline to see how the LEDs are lit in turn during the different phases.



Part 2: Interactive traffic lights

Up to now, the system has been relatively simple. Now you're going to change it slightly. Let's imagine pedestrian traffic lights on a straight section of a main road. There's no point in the phases for motorists changing all the time if no pedestrians want to cross the traffic lane. How should the traffic lights work with their phases? What equipment is needed and how can the logic be extended? Here's what you need to consider.

- If no pedestrians come forward to cross the road, the light remains green for motorists and red for pedestrians.
- If a pedestrian presses the light button to cross safely, the light turns amber and then red for motorists. The light then turns green for pedestrians. After a predefined time, the light changes back to red for pedestrians, and the red-light changes to amber and then green for motorists.

The starting situation looks like this:

Phase 1

Motorist	Pedestrian	Explanations
		These two light signals remain on until a pedestrian approaches and presses the button controlling the lights. Only then are the phase changes triggered, turning the light red for motorists and green for pedestrians.

Let's take a closer look:

Phase 2

Motorist	Pedestrian	Explanations
		The phase change is triggered by pressing the button controlling the lights. The light turns amber for motorists, meaning that it will soon turn red. Duration: 3 seconds

Phase 3

Motorist	Pedestrian	Explanations
		For safety reasons, the light is red first for motorists and then for pedestrians. This allows motorists to clear the crosswalk if necessary. Duration: 1 second

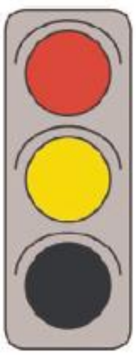
Phase 4

Motorist	Pedestrian	Explanations
		The light turns green for the pedestrian after a short time. Duration: 10 seconds

Phase 5

Motorist Pedestrian	Explanations
 	The light changes back to red for pedestrians. Duration: 1 second

Phase 6

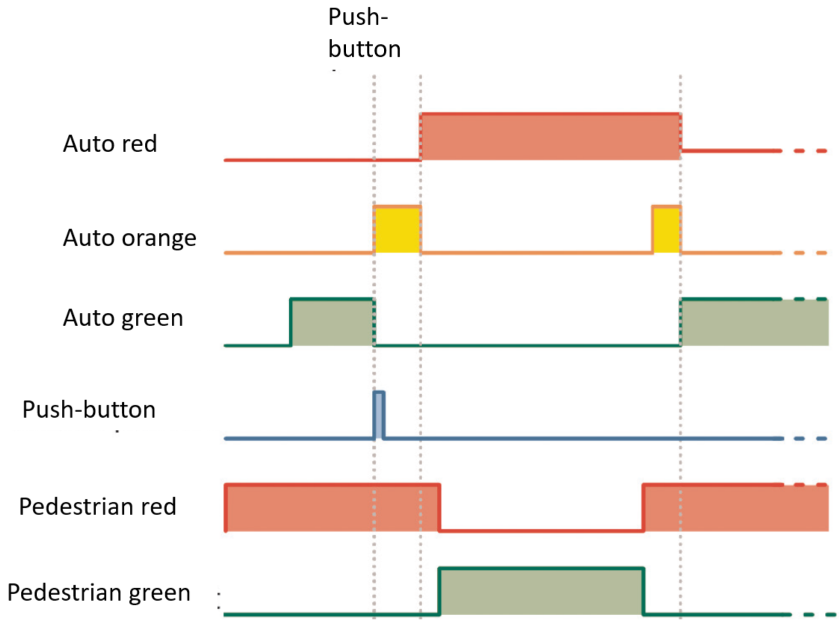
Motorist Pedestrian	Explanations
 	The light changes from red to amber for motorists, warning them that the light will soon turn green. Duration: 2 seconds

Phase 7

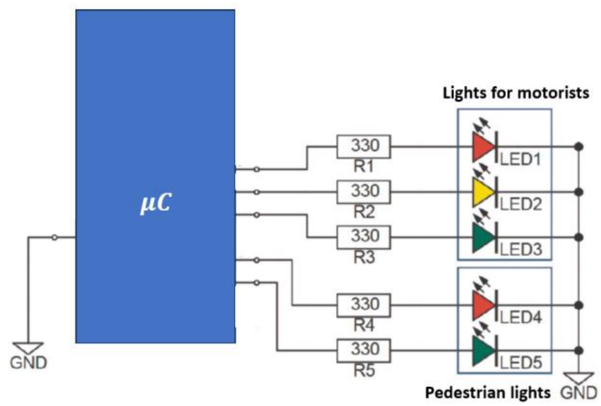
Motorist Pedestrian	Explanations
 	The light turns green for motorists and red for pedestrians. This last phase is similar to the first. Duration: until next button press

Diagram

Before we move on to the diagram, here's another chronogram showing the different light-up times in relation to each other. The initial situation shows that the traffic light is green for motorists and red for pedestrians. A pedestrian intending to cross the road at a supposedly safer spot presses the button controlling the lights, initiating the phase changes.



The interactive traffic light circuit diagram is shown in the following figure. Note that the board's push button is not shown in this circuit.



3. Deliverables

- Brief description of the purpose and theory of the problem.
- Brief explanation of your solution algorithm.
- Design document.
- Screenshots of the application demonstration.
- Discussion.
- Conclusion.

4. Evaluation

- Points for correct operation of the application: /50.
- Points for the laboratory report: /50.