Instituto Superior Técnico

Master's Degree in Information Systems and Computer Engineering

Software for Embedded Systems

4th Lab work – Project: Traffic lights control

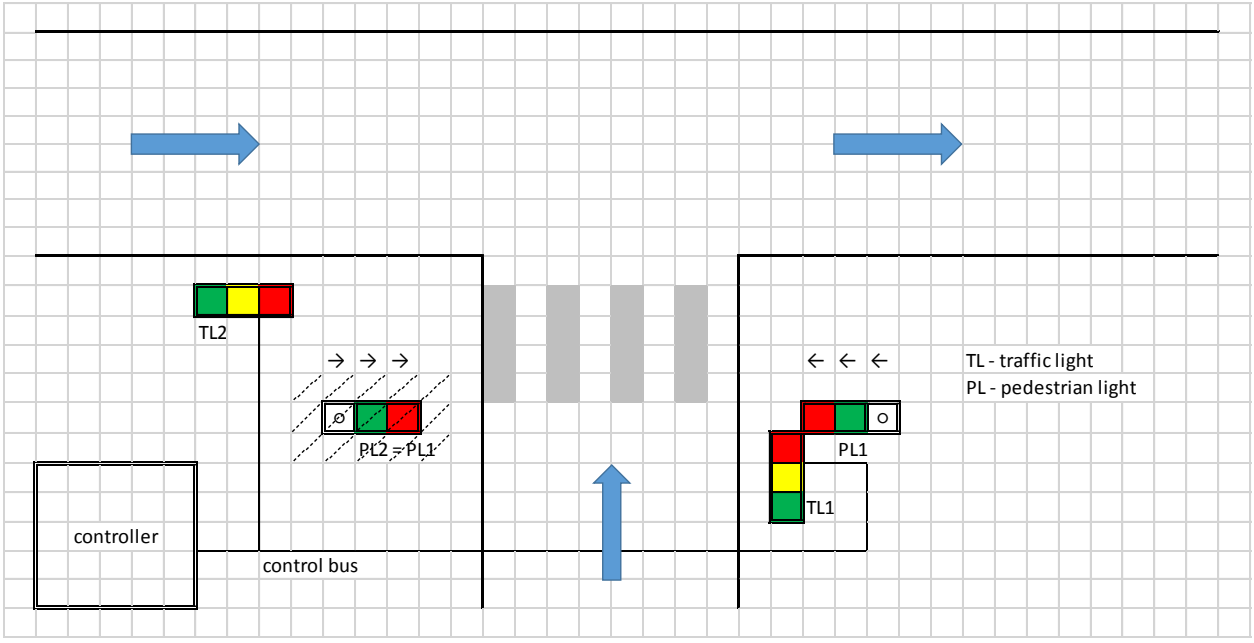
Goal:

The goal of this project is the implementation of a modular and automatic traffic lights system for a simple road junction. All traffic lights and its controller are connected via I2C bus. The overall system has fail safe behavior and some degree of fault tolerance.

In the laboratory each light will be replaced by a LED, otherwise the system to be implemented exhibits realistic behavior.

Description:

The purpose of the work is the design and development of a traffic lights system to control traffic at a simple road junction.



The system has two traffic lights (TL1, TL2) and a controller connected through an I2C bus. One of the traffic lights also has pedestrian lights and buttons for pedestrians (PL1, PL2) to signal their intent to cross the road. (To simplify the assembly of the circuit only PL1 will be implemented.)

The functionalities of the controller and of the traffic lights must be implemented according to a specified interface protocol so that any traffic light respecting the interface can work with any controller. (The traffic light implemented by a group of students must be interoperable with a controller implemented by another group.)

Traffic Light:

The traffic light consists of an Arduino UNO with multiple I/O devices. Every traffic light must have 5 LEDs: 3 of them represent the traffic control color signals – green, yellow, and red – and 2 represent the pedestrians control signals – green, and red. Every traffic light must also have a pedestrian button to reduce the waiting time for pedestrians to cross the road. Each traffic control lights must have fault detection capability to detect that a light is always turned OFF and does not react to its control.

Controller:

The controller is also an Arduino UNO with I/O devices attached.

- 3 LEDs: green and red LEDs to show the controller status (ON or OFF), and a blue LED indicates activity on the communications bus;
- an ON/OFF button;
- a potentiometer to select the period of traffic control (red → green → red).

Requirements:

1. Initial state of the system must – controller turned OFF (red light ON, and green light OFF), both traffic lights blinking yellow.
2. It must be possible to turn the controller ON and OFF, pressing the button:
    a. When turned ON, the controller must signal one of its traffic lights to go red and the other to go green.
    b. When turned OFF, the controller must signal both traffic lights to start blinking yellow, going back to the initial state.
3. While receiving or sending data the controller's blue LED must blink.
4. The traffic light color transitions must be:
    a. Red → Yellow (immediately before: pedestrian Green → Red).
    b. Yellow → Green.
    c. Green → Yellow.
    d. Yellow → Red (immediately after: pedestrian Red → Green).

5. It must be possible to control the red-yellow-green-yellow-red cycle time of the traffic lights using a potentiometer (min: 4 seconds between light transition; max: 10 seconds between light transition).
6. It must be possible to shorten the cycle time by half by pressing the pedestrian button. The reduction affects a single cycle after which the system reverts to its normal operation cycle.
7. While one traffic light is performing a red-yellow-green-yellow-red cycle, the other must have its red light always on (and, in part, pedestrian green). In other words, two traffic lights must not be simultaneously green.
8. All communications between the traffic lights must be performed via controller (see section Modular Programming).
9. It must be possible for the controller and both traffic lights, to detect faults of the communications link:
    a. After 2 periods (of 4 to 10 seconds) with persistent faults:
        i. The traffic light that detected the missing or faulty communication must start blinking yellow.
        ii. The controller must communicate to the other traffic light to start blinking yellow, too. Then it must turn itself off, going back to the initial state of the system.
10. All communications in the system must be performed using I2C protocol.

Modular Programming:

Communications between the modules of the system (traffic lights and controller) must respect a specific API in order to create a modular and interoperable system. This approach:

- will enable two teams of programmers, working in parallel, to build both controller and traffic light separately, and
- it allows controllers and traffic lights from different teams to work together properly.

Controller - Traffic light communication API:

Controller - controller should implement functions to be called when the following messages are received via I2C :

- RED {X}: where X is the identifier of the traffic light that sends the message. It alerts the controller that the specific traffic light has turned red.
- PING {X}: where X is the identifier of the traffic light that sends the message. It requests the controller for a signal that it is still alive.
- ACK {X}: where X is the identifier of the traffic light that sends the message. It is the “I’m alive” signal from the traffic light that sends the message.

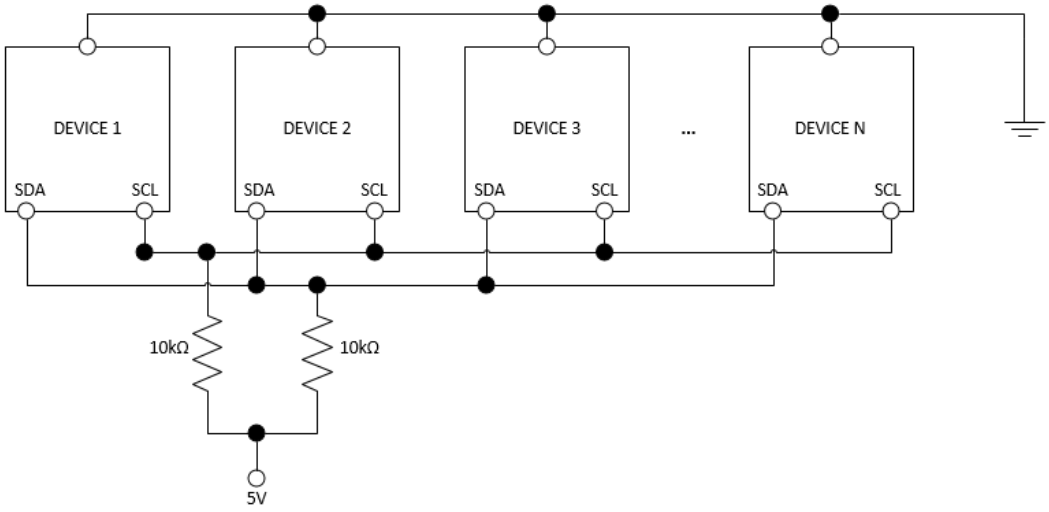
Traffic lights - traffic lights should implement functions to be called when the following messages are received via I2C :

- ON {CLR}: where CLR is the traffic light color identifier which the traffic light should be started with (RED: 1; YELLOW: 2; RED: 3).
- OFF: upon receiving this message the traffic light should start blinking yellow.

- GRN: this message signals the traffic light to start its color cycle.
- TIME {X}: where X is the time in milliseconds between 2000 and 10000. This message signals the traffic light to reset the time between light transitions.
- PING: . It requests the traffic light for a signal that it is still alive.
- ACK: It is the “I’m alive” signal from the controller.

I2C with Multi Master-Slave diagram:

A diagram is given so students can understand how to build a multi Master-Slave circuit for Arduino UNO.



Recommendations:

In order to fulfill your work with security and not damaging the hardware involved, remember to carry out the recommendations below. As you are working fill the boxes to be certain that you fulfill all security measures.

Always work with the circuit disconnect from the source.	
Call the professor or responsible for the laboratory, before you connect the circuit to the source.	
Make sure the circuit is well connected (resistors, capacitors, etc.) to prevent a short circuit, or damage the hardware.	

Plan and Deliverables:

1. Design and development of the project (first and second lab classes).
2. Demonstration (third lab class)
3. Implementation Report (printed report, and e-mail) including

    a. design of the circuits of the controller and traffic lights,
    b. overall architecture of the software in the controller and traffic lights,
    c. safety and fault-tolerance measures adopted,
    d. programs (properly commented) implementing the controller and traffic lights.
4. Demonstration of the integration of modules from different groups. (Example: controller from group A + traffic light from group A + traffic light from group B.) (fourth lab class)
5. Integration Report (printed report, and e-mail) including the results of the integration tests.