PROJECT 2

ON-DEMAND TRAFFIC LIGHT CONTROL

Project 2

On-demand Traffic Light Control

Author:	**Maged Magdy Asaad**
Nanodegree:	**Embedded Systems**
Track:	**Professional**

Powered by egFWD Scholarship Program

Table of Contents

Table of Figures

System Description

This is a system for on-demand traffic light control. The system simulates a traffic light for cars and pedestrians using LEDs and a button. The system uses different LEDs for cars and pedestrians, and uses a timer to control the duration of the different light states (green, yellow, red). The system also has a button that allows the user to switch between normal mode, where the traffic light follows a normal sequence, and pedestrian mode, where the traffic light sequence is adjusted to allow pedestrians to cross.

In normal mode, the traffic light sequence starts with the green light for cars and lasts for 5 seconds, then the yellow light for cars blinks for 5 seconds, then the red light for cars turns on and lasts for 5 seconds, and finally, the yellow light for cars blinks for 5 seconds. In pedestrian mode, the traffic light sequence is adjusted to allow pedestrians to cross. The car's red light turns on and the pedestrian's green light turns on and lasts for 5 seconds, then the pedestrian's yellow light blinks for 5 seconds, and finally, the pedestrian's red light turns on and lasts for 5 seconds.

This system provides several benefits for traffic control. Firstly, it allows for on-demand control of the traffic light sequence, allowing for adjustments to be made in real-time based on traffic conditions. This can lead to improved traffic flow and reduced congestion. Additionally, the pedestrian mode allows for safe crossing for pedestrians, promoting safety and accessibility. Furthermore, the program is written in C which makes it more flexible, and can be easily ported to other microcontroller platforms. Lastly, the system can be controlled remotely via a button, providing convenience for the user.

System FlowChart

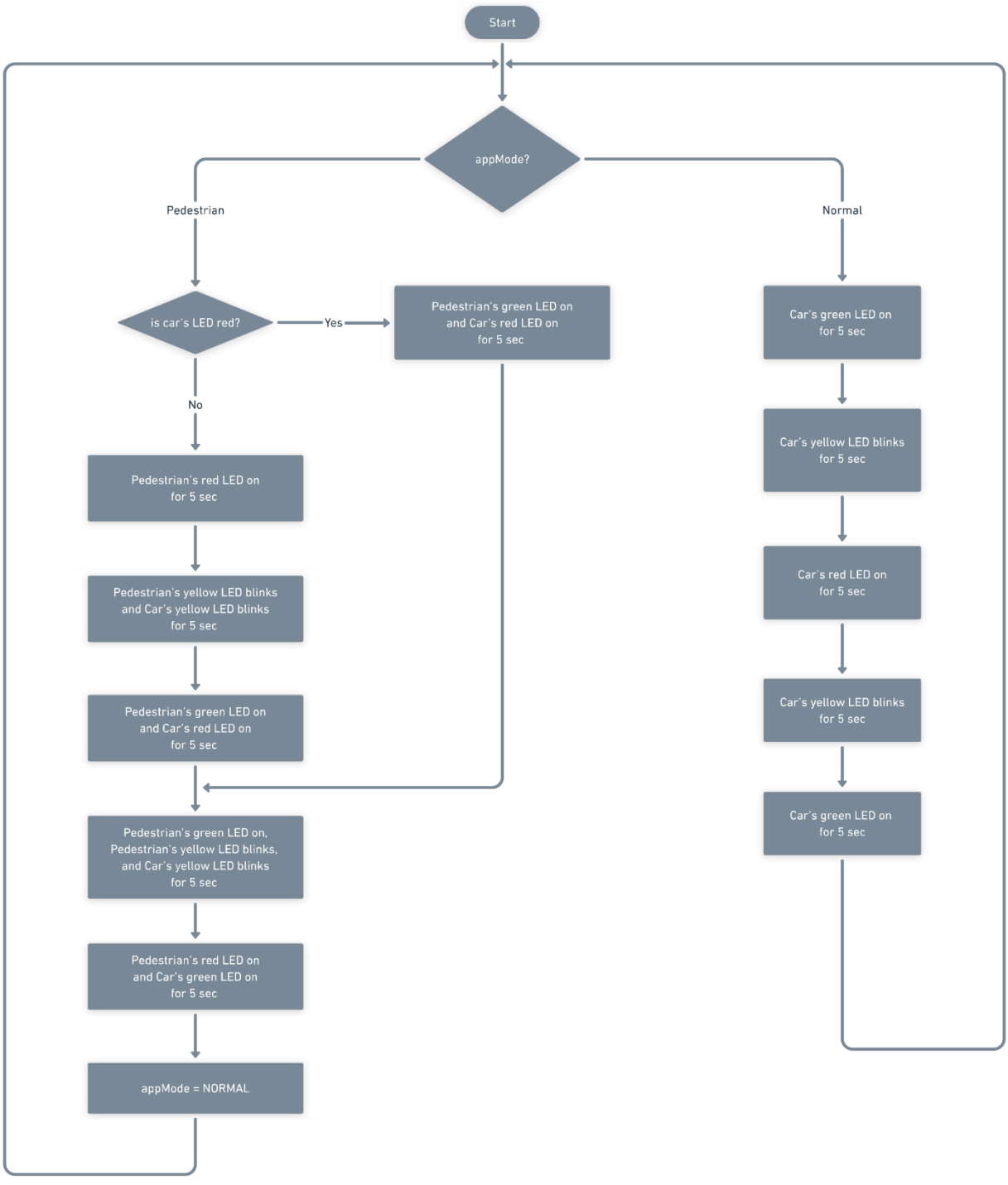


Figure 1: System Flowchart

System Design

The system design of the on-demand traffic light control system is based on an ATMEGA32a microcontroller and utilizes LEDs and a button to simulate a traffic light for cars and pedestrians. The system uses different LEDs for cars and pedestrians, and uses a timer to control the duration of the different light states (green, yellow, red).

The system includes several components:

- 6 LEDs: The system uses different LEDs to indicate the different traffic light states for cars and pedestrians. Green LEDs indicate a green light, yellow LEDs indicate a yellow light, and red LEDs indicate a red light.
- 1 Button: The system uses a button to switch between normal mode and pedestrian mode connected to PIN 2 in PORTD.
- 1 Timer: The system uses a timer 0 to control the duration of the different light states (generate 5 seconds delay).
- 1 External Interrupt: The system uses INT0 to sense a rising edge and switch between normal mode and pedestrian mode.

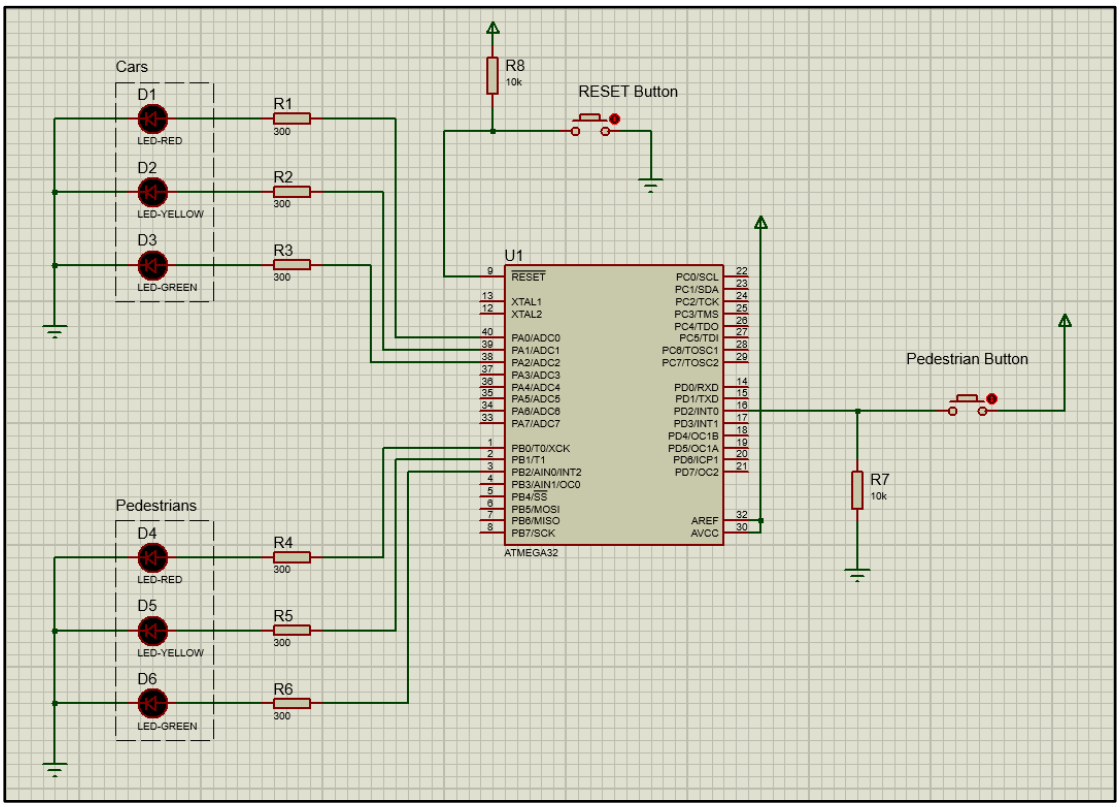
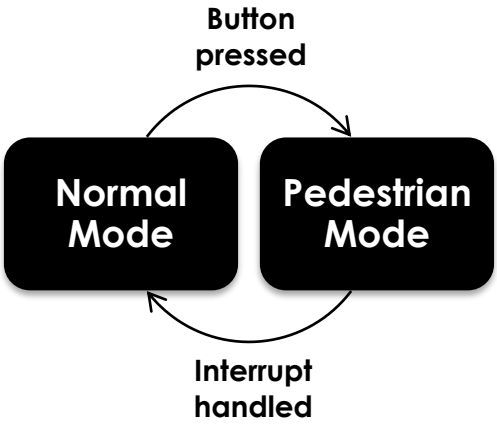


Figure 2: System Circuit

The system design is based on a state machine approach where the program is in a specific state and based on the state it performs a certain action and then it moves to the next state. The system is designed to handle different traffic scenarios by switching between normal mode and pedestrian mode, and it uses a timer to control the duration of the different light states. The system also uses INT0 to sense a rising edge and switch between normal mode and pedestrian mode based on user input.



The system is designed to be simple and easy to use, and it can be easily ported to other microcontroller platforms. The system also provides flexibility in terms of handling different traffic scenarios, promoting safety and accessibility, and it can be controlled remotely via a button.

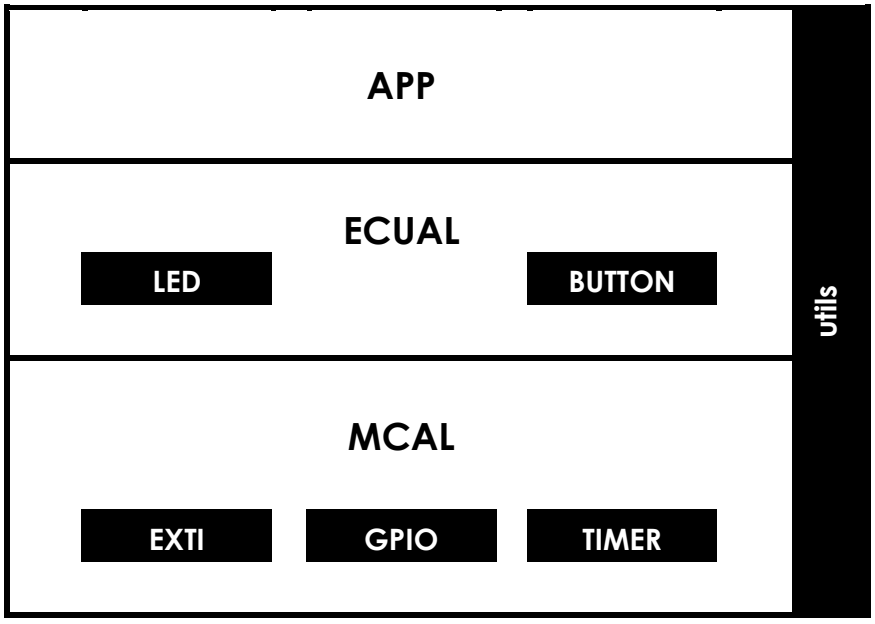


Figure 3: Layered Architecture

The on-demand traffic light control program uses a layered architecture to separate the different tasks and functions of the system. The architecture is divided into three main layers: the application layer (APP), the electronic control unit abstraction layer (ECUAL), and the microcontroller abstraction layer (MCAL).

The application layer is the highest layer in the architecture and it contains the main application code. This layer handles the control of the traffic light sequence, the switch between normal mode and pedestrian mode, and the integration of the different functions and data structures.

the electronic control unit abstraction layer is the middle layer and it contains the code for the different drivers such as the LED driver, button driver. This layer handles the communication between the application layer and the microcontroller abstraction layer.

The microcontroller abstraction layer is the lowest layer and it contains the code for the different drivers such as general purpose intput/output driver (GPIO), external interrupt driver (EXTI), and timer driver. This layer handles the communication between the ECU layer and the physical hardware.

The layered architecture allows for a clear separation of concerns and makes it easier to develop, test, and maintain the code. It also improves the flexibility of the system, as it can be easily ported to other microcontroller platforms by only modifying the hardware layer. Furthermore, the layered architecture allows for the easy integration of new features or functions, as they can be added to the appropriate layer without affecting the other layers.

System Constraints

There are several constraints of the system:

1. The system is designed to control traffic lights using LEDs and a button, so it may not be suitable for larger-scale traffic control systems.
2. The system uses a timer to control the duration of the different light states, so it may not be able to handle sudden changes in traffic conditions.
3. The system is dependent on the microcontroller platform and the availability of specific ports and pins, so it may not be easily portable to other platforms.
4. The system uses a button to switch between normal mode and pedestrian mode, so it may not be suitable for situations where remote control is required.
5. The system uses a fixed timer delay of 5 seconds, so it may not be able to handle different traffic scenarios that require different delay times.

In order to change the 5 seconds delay, change the initial value and number of overflows in TMR0_Config.h file.

The calculations were as following to generate 5 seconds delay:

$$Tick\ Time = \frac{1024}{1 \times 10^{6}} = 1.024\ msec$$

$$Max\ delay = 2^{8} \times 1.024 = 262.144\ msec$$

$$N_{Overflow} = \frac{5}{262.144 \times 10^{-3}} = 19$$

$$Timer_{Initial\ Value} = 2^{8} - \frac{\left(\frac{5}{1.024 \times 10^{-3}}\right)}{19} = 0$$

Figure 4: Timer Calculations

Solution Explorer

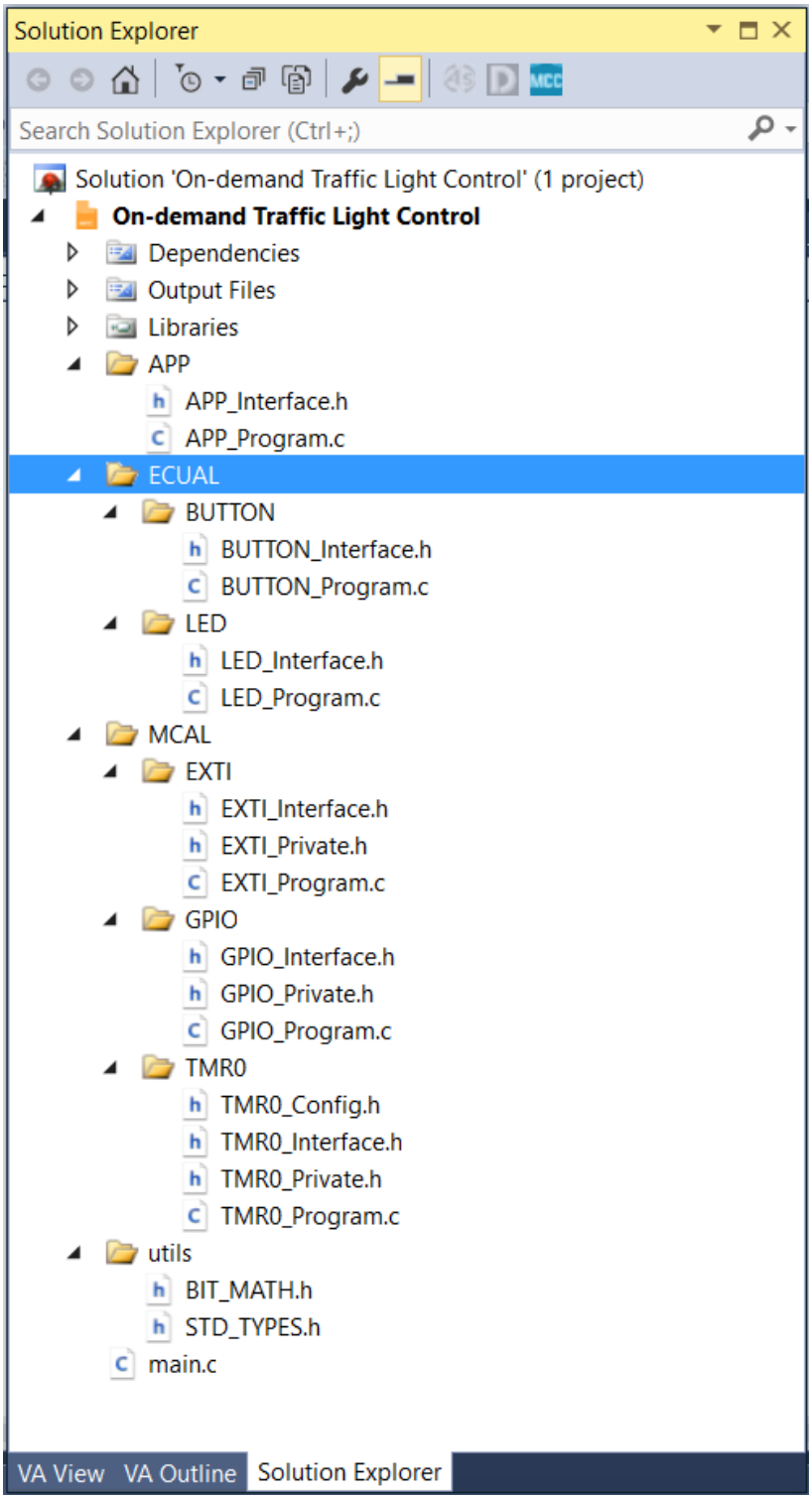


Figure 5: Solution Explorer

Future Development

There are several potential areas for future development for the on-demand traffic light control program:

1. Adding support for remote control: The system currently uses a button for switching between normal mode and pedestrian mode, adding support for remote control via a wireless connection such as WiFi or Bluetooth would allow for more flexibility in controlling the traffic lights.
2. Implementing an adaptive traffic light control algorithm: The system currently uses a fixed timer delay of 5 seconds, implementing an adaptive traffic light control algorithm would allow the system to adjust the delay times based on real-time traffic conditions, leading to improved traffic flow and reduced congestion.
3. Adding support for multiple intersections: The system currently only controls one traffic light, adding support for multiple intersections would allow for coordinated control of multiple traffic lights and improved traffic flow.
4. Adding support for multiple modes: adding support for different modes such as emergency mode, rush hour mode, and off-peak mode would allow the system to adapt to different traffic scenarios.
5. Improving the fault-tolerance: The system can be enhanced to handle errors and unexpected situations, by adding error handling mechanisms and fail-safe modes.

Overall, the on-demand traffic light control program has a lot of potential for future development, and with the implementation of these features, it can be used in more complex traffic control systems, and can improve the traffic flow and safety in the city.