Ministry of Communications and Information Technology

egyptfwd

مستقبلنا رقمي

System Design Report

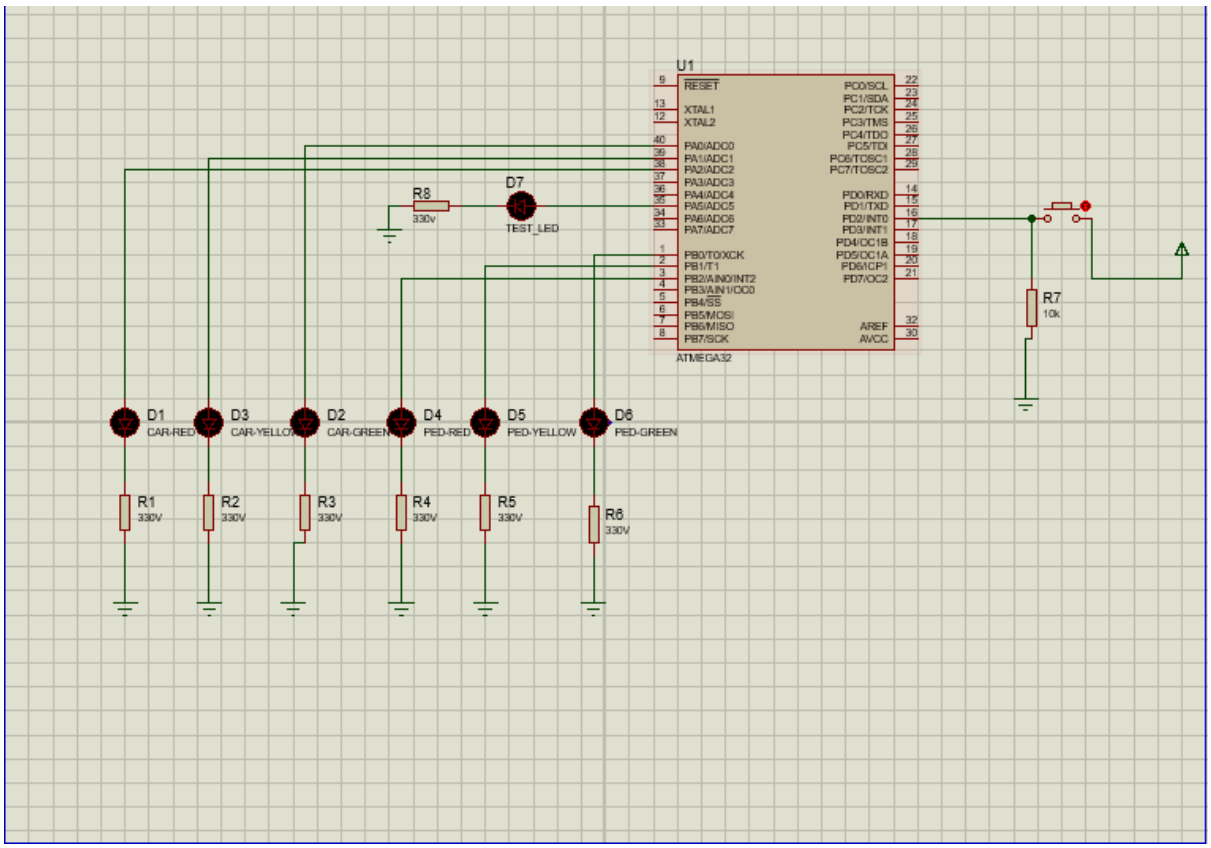
A DESCRIPTION FOR THE IMPLEMENTED SYSTEM:

ON-DEMAND TRAFFIC LIGHT CONTROL PROJECT

OLA MOHAMED AHMED MOHAMED

1. System Overview

A traffic light system with an on-demand crosswalk button which lets the traffic system know that there is someone who wants to cross the street, so the system adjusts the traffic for them and gives them enough time to cross.



2. System Description

Hardware Components:

1. ATmega32 microcontroller

2. 1 push button connected to INT0 pin (PD2) for pedestrian mode routine

3. 3 LEDs for cars (Green - Yellow – Red), connected on PORTA to PINs 0, 1, and 2

4. 3 LEDs for pedestrians (Green - Yellow – Red), connected on PORTB, PINs 0, 1, and 2

5. 6 resistors (330-ohm) 6. 1 resistor (10k-ohm)

System Requirements:

normal mode:

1. Cars' LEDs will be changed every five seconds starting from Green then yellow then red then yellow then Green.

2. The Yellow LED will blink for five seconds before moving to Green or Red LEDs.

pedestrian mode:

1. Change from normal mode to pedestrian mode when the pedestrian button is pressed.

2. If pressed when the cars' Red LED is on, the pedestrian's Green LED and the cars' Red LEDs will be on for five seconds, this means that pedestrians can cross the street while the pedestrian's Green LED is on.

3. If pressed when the cars' Green LED is on or the cars' Yellow LED is blinking, the pedestrian Red LED will be on then both Yellow LEDs start to blink for five seconds, then the cars' Red LED and pedestrian Green LEDs are on for five seconds, this means that pedestrian must wait until the Green LED is on.

4. At the end of the two states, the cars' Red LED will be off and both Yellow LEDs start blinking for 5 seconds and the pedestrian's Green LED is still on.

5. After the five seconds the pedestrian Green LED will be off, and both the pedestrian Red LED and the cars' Green LED will be on.

6. Traffic lights signals are going to the normal mode again.

3. System Design

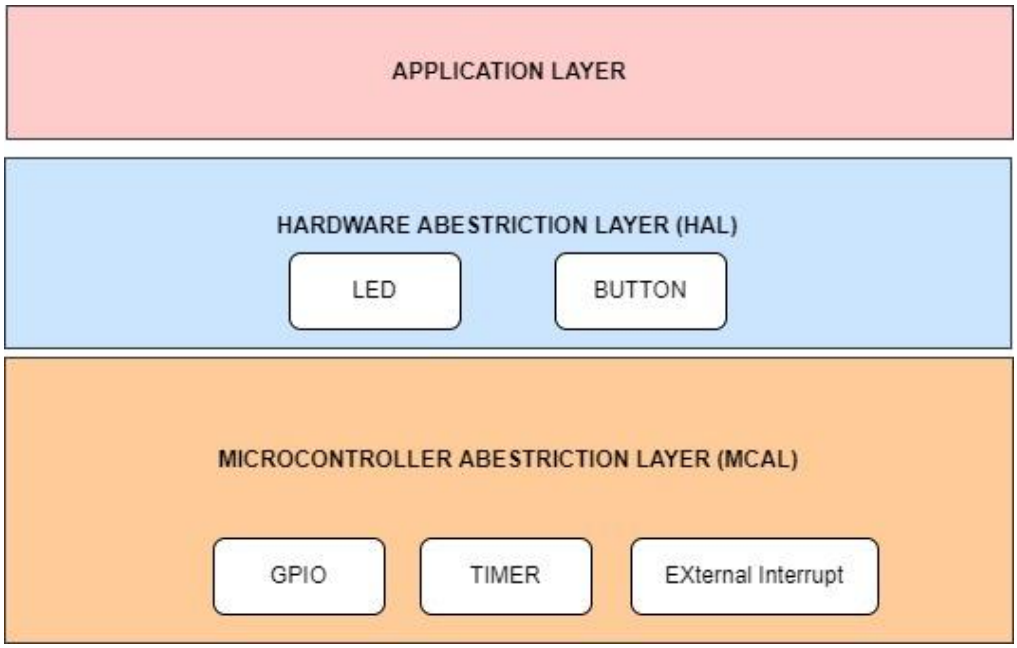
This system consists of 2 modes (states), normal mode and pedestrian mode. The normal mode is the default routine running all the time in the application background and the pedestrian mode is the one that interrupts the normal mode routine by pushing the push button connected to the external interrupt pin in the micro-controller. After that interrupt finishes its routine, the normal mode back again to continue its routine till the next interrupt.

Software Layered Architecture:

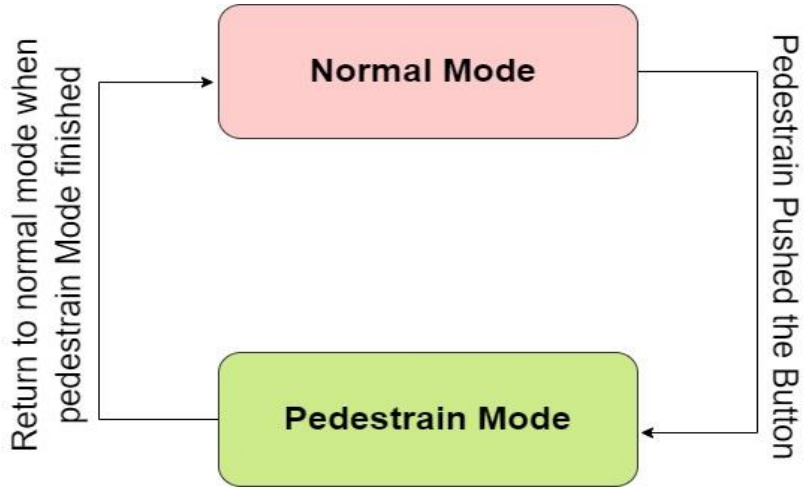
The system is statically designed as a layered architecture, and it's divided into 3 layers:

1. Microcontroller Abstraction Layer **(MCAL)**:
   It includes 3 modules: GPIO, Timer, and External Interrupt.
   - The GPIO module is responsible for initializing, setting, and reading pins.
   - The Timer module is for making delays.
   - The External Interrupt module is for running the Interrupt service routine (ISR) when its activation event occurs
2. Hardware Abstraction Layer **(HAL)**:
   It includes 1 module: LED
   - The LED module is responsible for Initializing all required pins for LEDs as output pins, turning LEDs ON or OFF.
3. Application Layer:

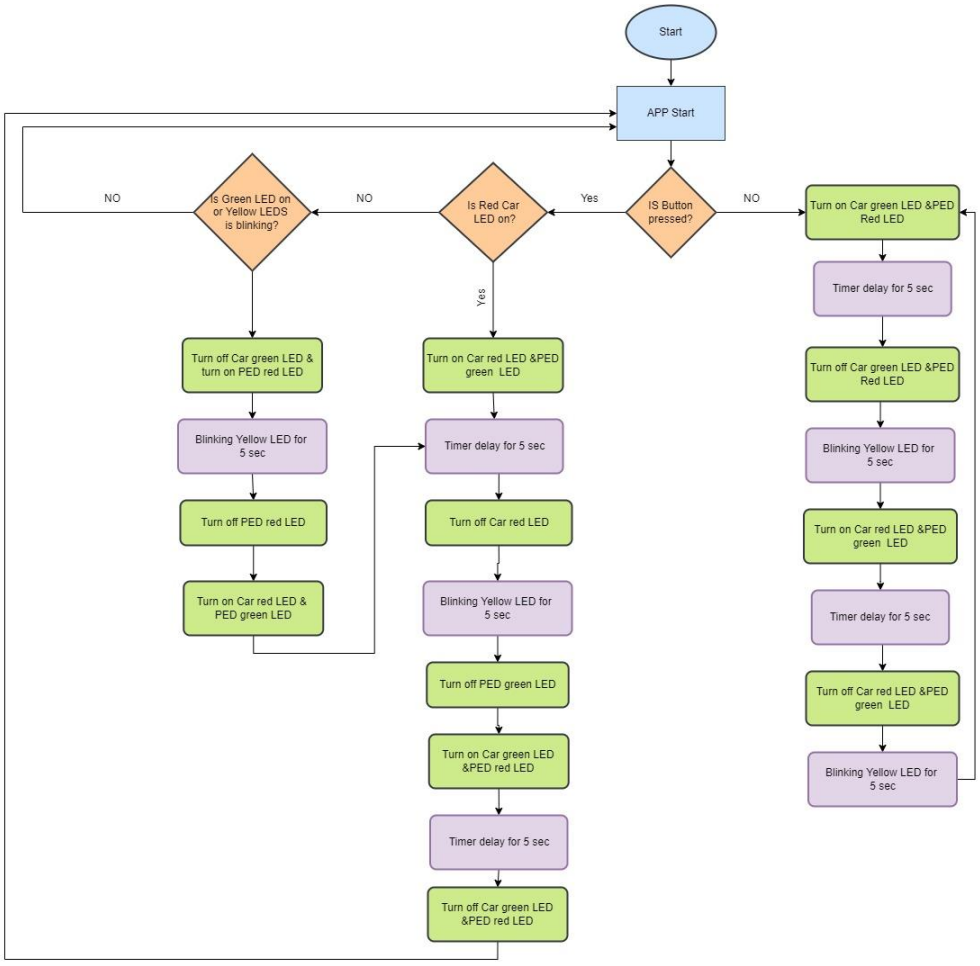
   At which we have all the application routines.



4. State Machine Diagram (SMD)



5. System Flow Chart



6. System Constraints

Here we have 2 constraints: 1. Dealing with the long press: We solve it in a way that whether the press was short or long it has the same effect on the system and the interruption will occur only once till it is totally executed. 2. Dealing with the double press: We solve it in a way that we neglect any press after the first one till the interruption is totally executed.