# Traffic Light System

## System Description

This project is traffic light with pedestrian button it works in two modes Normal mode and pedestrian mode.

By default, the normal mode is running, if the button is pressed, we enter pedestrian mode for a while, then go back to normal mode.

In normal mode cars' LEDs will be changed every five seconds starting from green then yellow then red then yellow then green. The yellow LED will blink for five seconds before moving to green or red LEDs, while Pedestrian LEDs will also change inversely with cars LEDs, but Pedestrian green and red led don’t turn off while yellow LED is blinking.

If the button is pressed that means Pedestrian want to cross the street, This will make Traffic Light jumps from its state to the ready mode of pedestrians where both yellow LEDs are blinking for five seconds then pedestrian green LED is on for five seconds for pedestrian to cross the street, but if the button is pressed while pedestrian green is on or while yellow LEDs are blinking nothing will happen.

After any of these cases, we return to the normal mode.

## System Design

This system consists of 3 layers:

1- Application Layer
2- ECU Abstraction Layer (ECUAL)
3- Microcontroller Abstraction Layer (MCAL)e

Application Layer

This is layer which contains logic of the system

It consists of 8 APIs:

1- Initialize, this function initializes Inputs and outputs and enable interrupt
2- Normal Mode, this function will execute what should be done in the normal mode
3- Pedestrian Mode, this function defines what will happen if the button is pressed
4- Cars Period, this function Turn on cars green LED and pedestrian red led for 5 seconds
5- From cars to pedestrian, this function turns off cars green led and blink both yellow LEDs for 5 seconds
6- Pedestrian period, this function turns on cars red led and turn off both pedestrian red and green LEDs then turn pedestrian green led for 5 seconds
7- From pedestrian to cars, this function blinks both yellow LEDs for 5 seconds then turn on cars green led and turn off pedestrian both red and yellow LEDs
8- ISR, interrupt service routine of INT0 pin.

ECUAL

This is layer which contains drivers of the components we will use, it consists of 2 drivers

1- **LED Driver**
   This driver consists of 5 APIs (Initialize, Turn ON, Turn OFF, Toggle, Read) each function takes struct which contains port and pin number then do its function and return -1 if the input is wrong else it will return 0
2- **Button Driver**
   This driver consists of 2 APIs (Initialize, Read)
   a) Initialize, this function takes each function takes struct which contains port and pin number then do its function and return -1 if the input is wrong else it will return 0
   b) Read, this function takes button struct and reference to value, then it reads the value of the pin and put it in that reference

MCAL

This is layer which contains drivers of the microcontroller we will use, it consists of 3 drivers

1- **DIO Driver**
   This driver is responsible for I/O ports.
   It consists of 8 APIs:
   a) Initialize pin, this function defines the direction of certain pin whether it is output, or input or input

pullup then return -1 if the input is wrong else it will return 0

b) Initialize port, the same as the previous but with ports instead of pins
c) Set pin, this function put output value High or Low on certain pin then return -1 if the input is wrong else it will return 0
d) Set port, the same as the previous but with ports instead of pins
e) Read pin, this function read the input value on certain pin whether it is High or Low then return -1 if the input is wrong else it will return 0
f) Read port, the same as the previous but with ports instead of pins
g) Toggle pin, this function toggles the output of certain pin then return -1 if the input is wrong else it will return 0
h) Toggle port, the same as the previous but with ports instead of pins

**2- Timer Driver**

This driver is responsible for timing

It consists of 5 APIs:

a) Init, this function set the mode of the timer
b) Set initial value, this function set initial value for the timer so when timer start will start counting from this value
c) Set prescaler, this function set prescaler of the timer and start it

d) Read flag, this function returns the value of timer overflow interrupt flag
e) Delay_ms, this function takes number of milliseconds to be delayed

**3- External Interrupt Driver**

This driver is responsible for external interrupt pins
It consists of 3 APIs:

a) Enable, this function enables certain interrupt pin and set its mode
b) Disable, this function disables certain interrupt pin
c) Read flag, this function returns the value of external pin interrupt flag

**4- Global Interrupt Driver**

This driver is responsible for enable and disable global interrupts
It consists of 2 APIs:

a) Enable, which enables global interrupts
b) Disable, which disables global interrupts

## State Machine

We have 4 states

1- Cars period: Cars Green LED is on and Pedestrian Red LED is off
2- From Cars to Pedestrian: Cars Yellow Led is blinking, Pedestrian Red LED is on, and Yellow LED is blinking
3- Pedestrians period: Cars Red LED is on and Pedestrian Green LED is on

4- From Pedestrian to Cars: Cars Yellow LED is blinking, Pedestrian Green Led is on, and Yellow LED is blinking.

Black arrows represent normal mode

Blue arrows represent button pressing