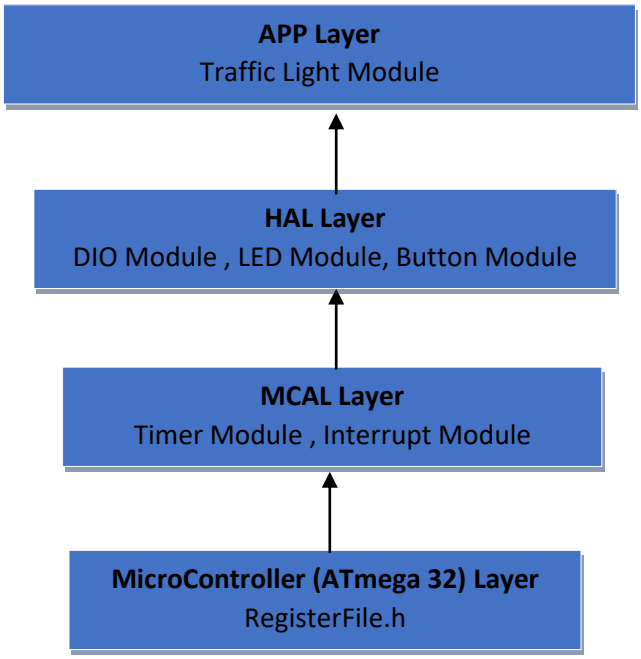
Traffic Lights Documentation

1:- The Static Design :



2:- Modules Functions :

1- MicroController Layer :

- **RegisterFile.h** :- This file contains the micro controller registers names and addresses

2- MCAL Layer :

- **Timer Module :-**

```
1-:     * Function Timer0_Stop
        * Desc     Function To Stop the Timer
        * Input    nothing
        * Output   nothing

2-      * Function    Timer0_InitNormal_Polling
        * Desc        Function To Initialize Timer0 in normal Mode and Polling
        * Input    nothing
        * Output   nothing

3-      * Function Timer0_InitNormal_Interrupt
        * Desc     Function To Initialize Timer0 in normal Mode With Interrupts
        * Input    interrupt Type
        * Output   nothing
```

```
4-      * Function Timer0_SetValue_ms
        * Desc     Function To set Timer 0 for a Certain Time in milliseconds
        * Input    Time in milliseconds
        * Output   nothing

5-      * Function Timer0_SetValue_micro
        * Desc     Function To set Timer 0 for a Certain Time in microSeconds
        * Input    Time in  microSeconds
        * Output   nothing

6-      * Function Timer0_Delay_ms
        * Desc     Function To set a delay using Timer 0 in milliseconds
        * Input    Time in  milliseconds
        * Output   nothing

7-      * Function Timer0_Delay_micro
        * Desc     Function To set a delay using Timer 0 in microSeconds
        * Input    Time in  microSeconds
        * Output   nothing

8-      * Function Timer0_setCallBackFunc
        * Desc     Function used to set the callback function for the interrupt ISR
        * Input    nothing
        * Output   nothing

9-      * Function Timer01_PrescalerReset
        * Desc     Function To reset the timer 0 prescaler
        * Input    nothing
        * Output   nothing

10-     * Function Timer0_NormalMode_Handler
        * Desc     The Timer 0 Interrupt ISR Handler
        * Input    nothing
        * Output   nothing

11-     * Function Timer0_CTC_init
        * Desc     Function To set the Timer 0 in CTC mode to create a square wave
        with a specific frequency
        * Input    the Action at compare and the frequency of the generated square
        wave
        * Output   nothing

12-     * Function Timer0_FastPWM_init
        * Desc     Function To set the Timer 0 in Fast PWM mode to create a square
        wave with a specific duty cycle
        * Input     the Action at compare and the duty cycle of the generated square
        wave
        * Output   nothing

13-     * Function Timer0_PhaseCorrect_init
        * Desc     Function To set the Timer 0 in Phase Correct PWM mode to create
        a square wave with a specific duty cycle
        * Input    the Action at compare and the duty cycle of the generated square
        wave
        * Output   nothing

14-     * Function Timer0_EventCount_init
        * Desc     Function To set the Timer 0 event counting  mode ,
        * Input    nothing
        * Output   nothing
```

15- * Function Timer0_Counter_Handler
* Desc Function To handle the ISR of counter mode interrupt
* Input nothing
* Output nothing

- **Interrupt Module:**

1- * Function EX_INT0_init
* Desc Function To Initialize External interrupt 0 INT0
* Input The trigger signal type
* Output nothing

2- * Function EX_INT1_init
* Desc Function To Initialize External interrupt 1 INT1
* Input The trigger signal type
* Output nothing

3- * Function EX_INT2_init
* Desc Function To Initialize External interrupt 2 INT2
* Input The trigger signal type
* Output nothing

4- * Function setCallBackFunc_INT0
* Desc Initialize the callback function for interrupt 0 INT0
* Input pointer to the interrupt handler
* Output nothing

5- * Function setCallBackFunc_INT1
* Desc Initialize the callback function for interrupt 1 INT1
* Input pointer to the interrupt handler
* Output nothing

6- * Function setCallBackFunc_INT2
* Desc Initialize the callback function for interrupt 2 INT2
* Input pointer to the interrupt handler
* Output nothing

7- * Function INT0_Handler
* Desc Function of the ISR handler of INT0
* Input nothing
* Output nothing

8- * Function INT1_Handler
* Desc Function of the ISR handler of INT1
* Input nothing
* Output nothing

9- * Function INT2_Handler
* Desc Function of the ISR handler of INT2
* Input nothing
* Output nothing

3:- HAL Layer：

- **DIO Module:**

1- * Function DIO_init
* Desc Function To Initialize a the direction of specific pin in a specific port
* Input the Port and the Pin and the direction
* Output nothing

```
2-      * Function DIO_write
        * Desc     Function To Write to a pin either high or low
        * Input    the Port and the Pin and the pin status
        * Output   nothing

3-      * Function DIO_read
        * Desc     Function To Read  a pin status in a specific port
        * Input    the Port and the Pin
        * Output   the Pin status

4-      * Function DIO_toggle
        * Desc     Function To toggle the status of a pin in a specific port
        * Input    the Port and the Pin
        * Output   nothing
```

- **LED Module:**

```
1-      * Function LED_init
        * Desc     Function To Initialize a led
        * Input    the Port and the Pin
        * Output   nothing

2-      * Function LED_on
        * Desc     Function To set a led on
        * Input    the Port and the Pin
        * Output   nothing

3-      * Function LED_off
        * Desc     Function To set a led off
        * Input    the Port and the Pin
        * Output   nothing

4-      * Function LED_toggle
        * Desc     Function To toggle a led
        * Input    the Port and the Pin
        * Output   nothing

5-      * Function LED_blink
        * Desc     Function To blink a led
        * Input    Port and the Pin and the between blinks
        * Output   nothing
```

- **Button Module:**

```
1-      * Function BUTTON_init
        * Desc     initialize a pin as a button
        * Input    pin number and port
        * Output   none

2-      * Function BUTTON_read
        * Desc     Function To read the status of a button
        * Input    pin number and the port
        * Output   the button status
```