2022

On Demand Traffic Light Control Design Document

MOHAMED YASSIN

system description:

On-demand Traffic light control is a system aimed to handle both Car and pedestrian traffic. So, our system will work in two modes.

1- Normal Mode:

In this Mode, the light traffic will handle cars and pedestrians automatically without any user trigger. And it will be power-on LEDs in the Car traffic side on the following Sequence.

- 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.

2- Pedestrian mode:

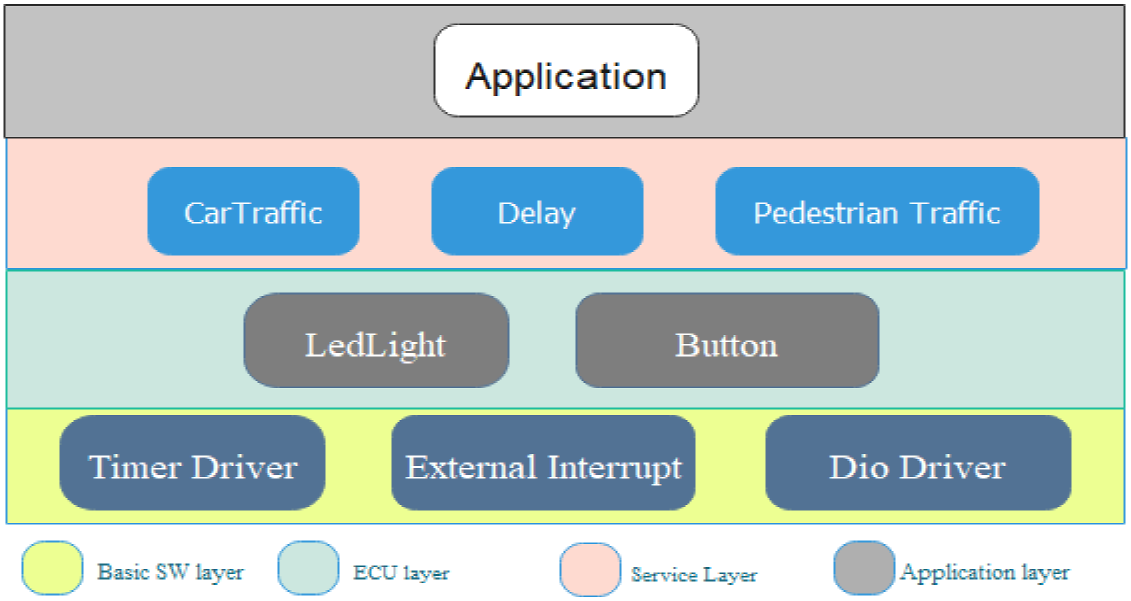
Users can change the system operating mode by pressing a button so, the system will change its operating system to Pedestrian mode. The transaction scenario will be handled as shown below.

- Change from normal mode to pedestrian mode when the pedestrian button is pressed.
- If pressed when the car's Red LED is on, the pedestrian's Green LED and the car's Red LEDs will be on for five seconds, this means that pedestrians can cross the street while the pedestrian's Green LED is on.
- If pressed when the car's Green LED is on or the car's 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.
- 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.
- After five seconds the pedestrian Green LED will be off and both the pedestrian Red LED and the cars' Green LED will be on.
- Traffic lights signals are going to the normal mode again.

system design:

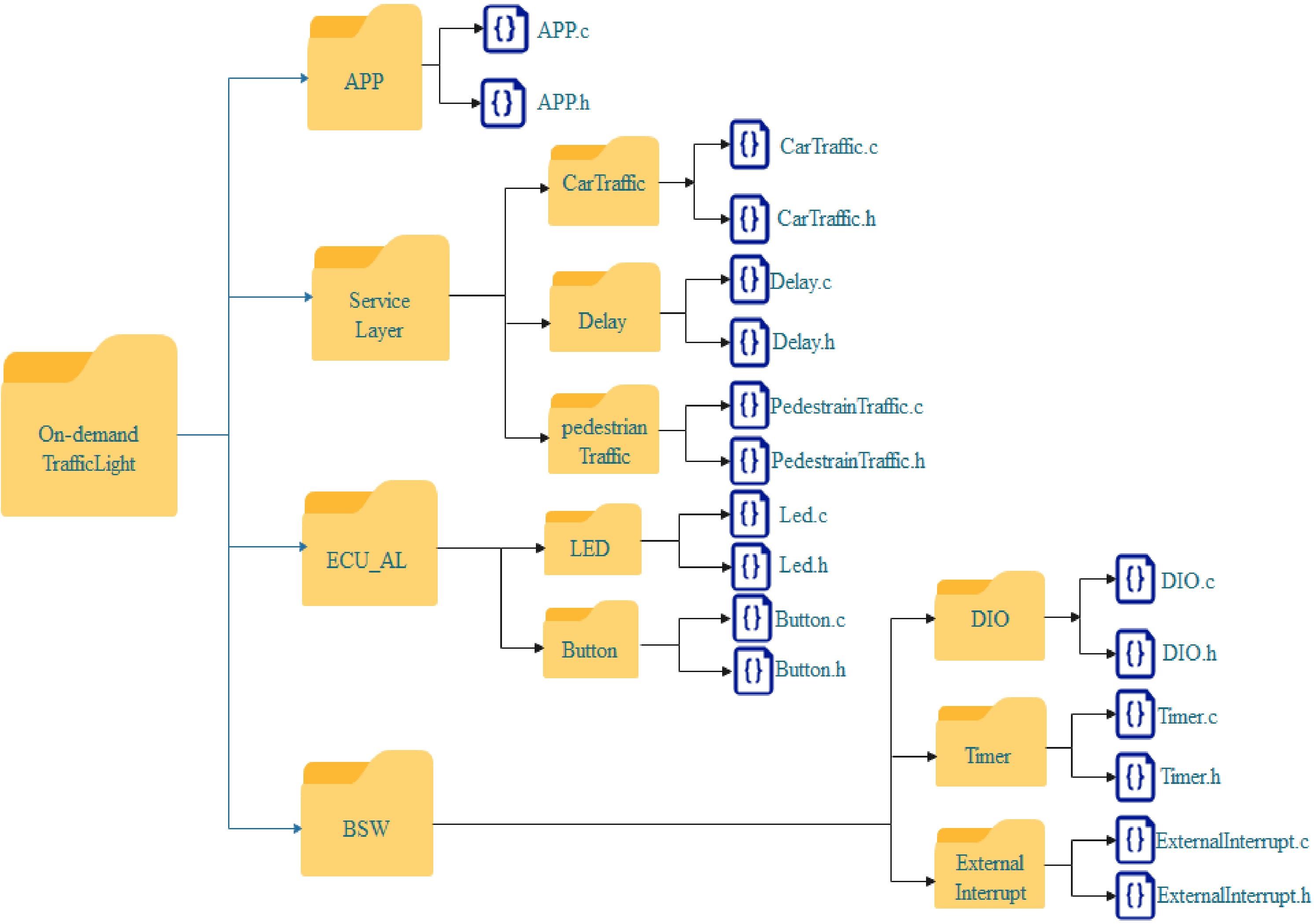
Static Design:

we will follow the layered architecture concept in our system as shown below.



System Folder structure:

The developer should use the below folder structure during the creation of the system skeleton.



Timer Module:

Timer Module APIs and Data Types.

Timer Driver

```
void Timer0_Init(TimerMode Mode,ClockSource ClkSource,TimerPinFun T_PinMode,
uint8_t InitValue);
```

Timer Operating Mode		
Name	TimerMode	
Type	Enum	
Elements	NormalMode	0
	OutputCompareMode	1
	FastPWMMode	2
	PhaseCorrectMode	3
Description	This Enum will map Timer operating mode	

Timer Clock Source

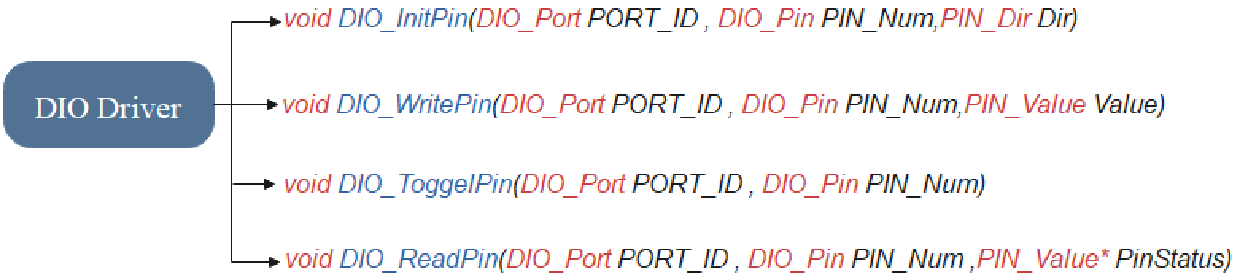
Name	ClockSource	
Type	Enum	
Elements	NoClockSource	0
	clk_NoPrescaling	1
	clk_FromPrescaler	2
	clk_64Prescaler	3
	clk_256Prescaler	4
	clk1024_Prescaler	5
	ExtClkSource_T0FallingEdge	6
	ExtClkSource_T0RisingEdge	7

Timer Pin-Function

Name	TimerPinMode	
Type	Enum	
Elements	OCx_Disconect	0
	CTC_ToggleOCx	1
	CTC_ClearOCx	2
	CTC_SetOCx	3
	FPWM_OCx_NonInvertMode	4
	FPWM_Ocx_InvertMode	5
	PPWM_SetOCxDownCount	6
	PPWM_ClrOCxDownCount	7

DIO Module:

DIO Module APIs and Data Types:



void DIO_InitPin(DIO_Port PORT_ID , DIO_Pin PIN_Num,PIN_Dir Dir)

void DIO_WritePin(DIO_Port PORT_ID , DIO_Pin PIN_Num,PIN_Value Value)

void DIO_ToggelPin(DIO_Port PORT_ID , DIO_Pin PIN_Num)

void DIO_ReadPin(DIO_Port PORT_ID , DIO_Pin PIN_Num ,PIN_Value* PinStatus)

DIO Port ID

Name	DIO_Port	
Type	Enum	
Elements	PORT_A	0
	PORT_B	1
	PORT_C	2
	PORT_D	3

DIO Pin ID

Name	DIO_Pin	
Type	Enum	
Elements	PIN_0	0
	PIN_1	1
	PIN_2	2
	PIN_3	3
	PIN_4	4
	PIN_5	5
	PIN_6	6
	PIN_7	7

PIN Direction

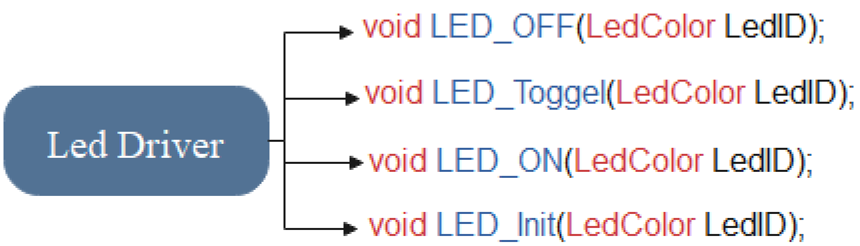
Name	PIN_Dir	
Type	Enum	
Elements	IN	0
	OUT	1

PIN Output Value

Name	PIN_Value	
Type	Enum	
Elements	Low	0
	High	1

Led Module:

Led APIs and Data Types:

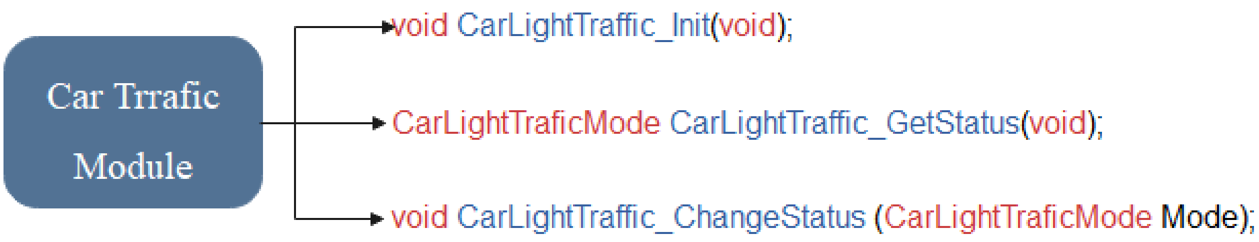


```
void LED_OFF(LedColor LedID);
void LED_Toggel(LedColor LedID);
void LED_ON(LedColor LedID);
void LED_Init(LedColor LedID);
```

Led Color Enum		
Name	LED_Color	
Type	Enum	
Elements	Green	0
	Yellow	1
	Red	2

Car Traffic Module:

Car Traffic APIs and Data Types:



```
void CarLightTraffic_Init(void);
CarLightTraficMode CarLightTraffic_GetStatus(void);
void CarLightTraffic_ChangeStatus (CarLightTraficMode Mode);
```

Car Traffic operation Mode		
Name	CarLightTraficMode	
Type	Enum	
Elements	CarStop	0
	CarWait	1
	CarWalk	2

Car Traffic APIs:

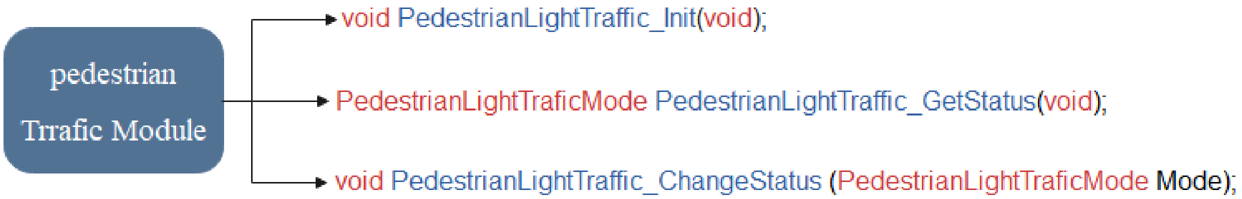
Init Led Lights_API	
Function Name	CarLightTraffic_Init
Arguments	Void
Return	Void
Description	This API will init all Pins connected to LED as Output and in LOW level

CarLight Traffic Status API			
Function Name	CarLightTraffic_GetStatus		
Arguments	Void		
Return	CarLightTraficMode	Value	Type
		Car_Stop	Enum
		Car_Wait	
		Car_Walk	
Description	This API is used as a getter to the current state of the car light traffic		

Car Light Traffic Set Status API			
Function Name	CarLightTraffic_SetStatus		
Arguments	Name	Value	Type
	Mode	Car_Stop	CarLightTraficMode Enum
		Car_Wait	
		Car_Walk	
Return	void		
Description	This API is used for setting Light Traffic in the required Mode: 1- Stop Mode: Red Led On and Both Green & Yellow LEDs Off. 2- Wait Mode: Yellow Led to toggle mode and Red & Green LEDs to Off. 3- Walk Mode: Green led to On and Red & Yellow LEDs to Off.		

Pedestrian Traffic Module:

Pedestrian Traffic APIs and Data Types:



```
void PedestrianLightTraffic_Init(void);
PedestrianLightTraficMode PedestrianLightTraffic_GetStatus(void);
void PedestrianLightTraffic_ChangeStatus (PedestrianLightTraficMode Mode);
```

Pedestrian Status Enum		
Name	PedLightTraficMode	
Type	Enum	
Elements	Ped_Stop	0
	Ped_Wait	1
	Ped_Walk	2

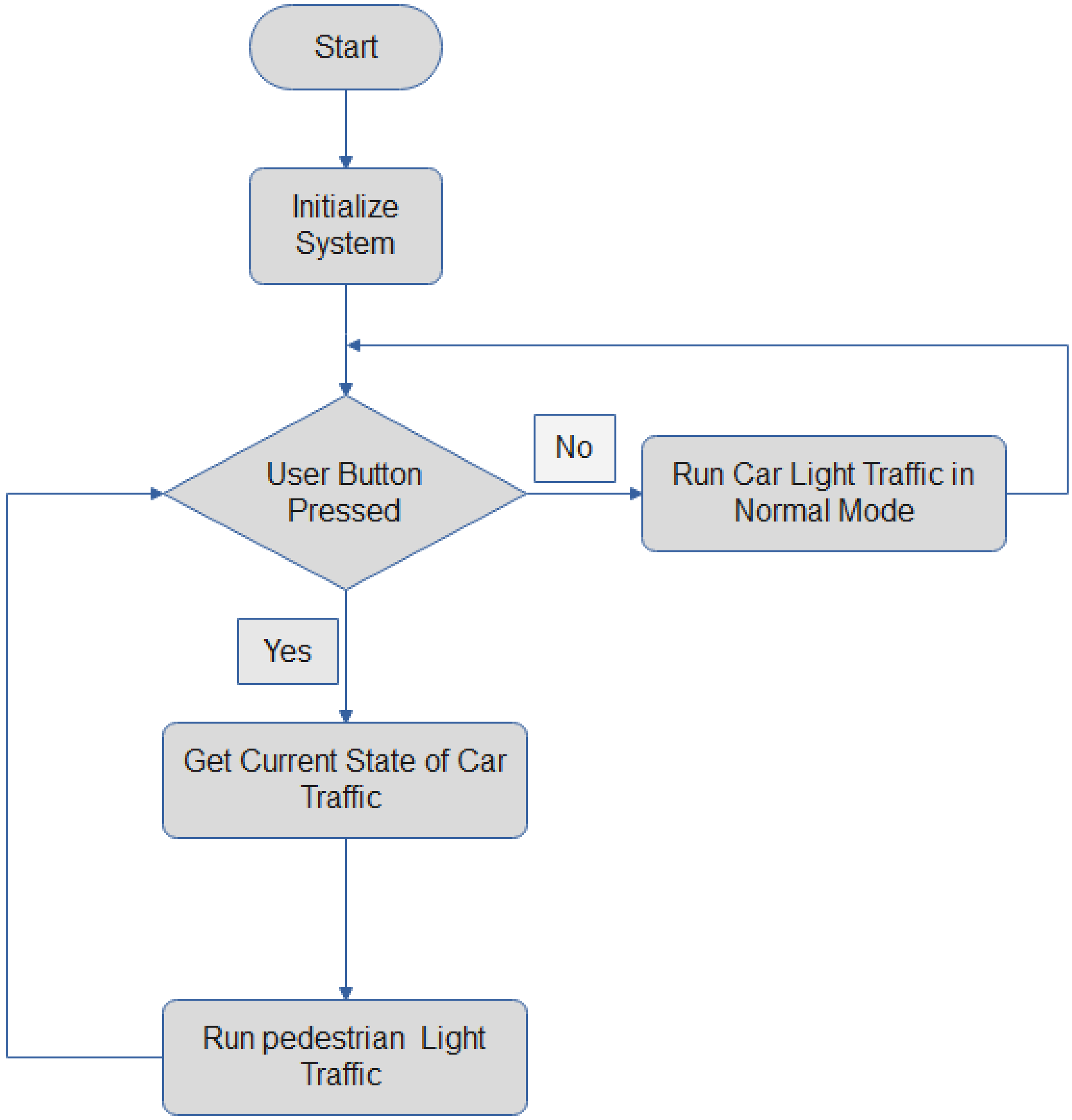
Pedestrian Traffic APIs:

Pedestrian Init Led Lights	
Function Name	Pedestrian LightTraffic_Init
Arguments	Void
Return	Void
Description	This API will init all Pins connected to Pedestrian LED as Output and in LOW level

Pedestrian Light Traffic Status API			
Function Name	PedestrianLightTraffic_GetStatus		
Arguments	Void		
Return	PedLightTraficMode	Value	Type
		Ped_Stop	Enum
		Ped_Wait	
		Ped_Walk	
Description	This API is used as a getter to the current state of the Pedestrian light traffic		

Pedestrian Light Traffic Set Status API			
Function Name	PedLightTraffic_SetStatus		
Arguments	Name	Value	Type
	Mode	Ped_Stop	PedLightTraficMode Enum
		Ped_Wait	
		Ped_Walk	
Return	void		
Description	This API is used for setting Pedestrian Light Traffic in the required Mode: 1- Stop Mode: Red Led on and Both Green & Yellow LEDs Off. 2- Wait Mode: Yellow Led to toggle mode and Red & Green LEDs to Off. 3- Walk Mode: Green led to On and Red & Yellow LEDs to Off.		

System Float Chart:


Start
Initialize System
User Button Pressed
No
Run Car Light Traffic in Normal Mode
Yes
Get Current State of Car Traffic
Run pedestrian Light Traffic