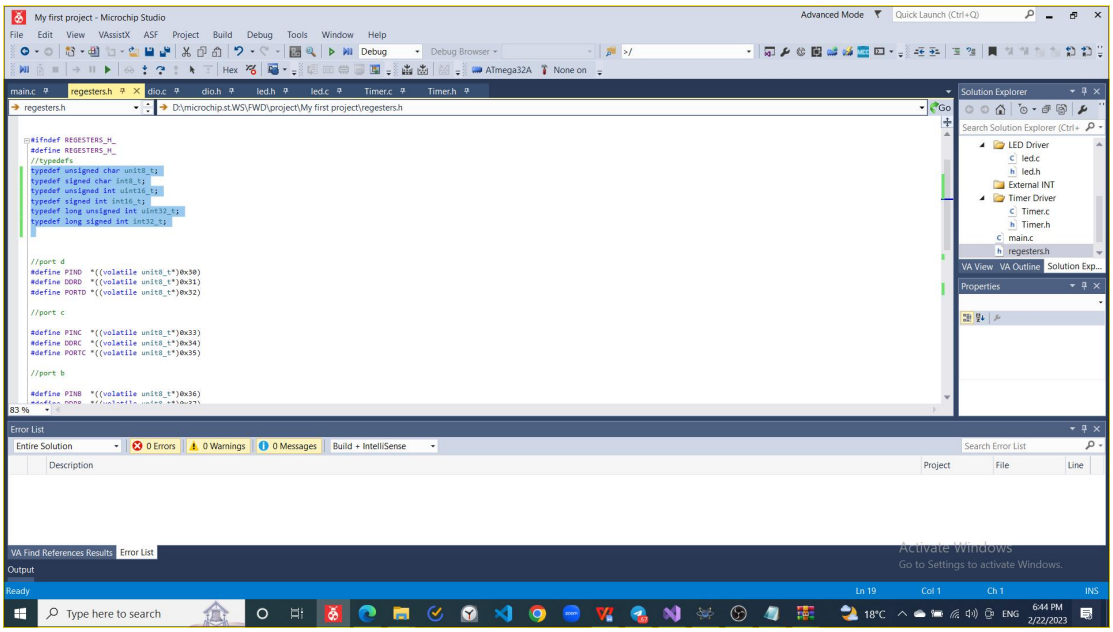
On damned traffic
Software desgine

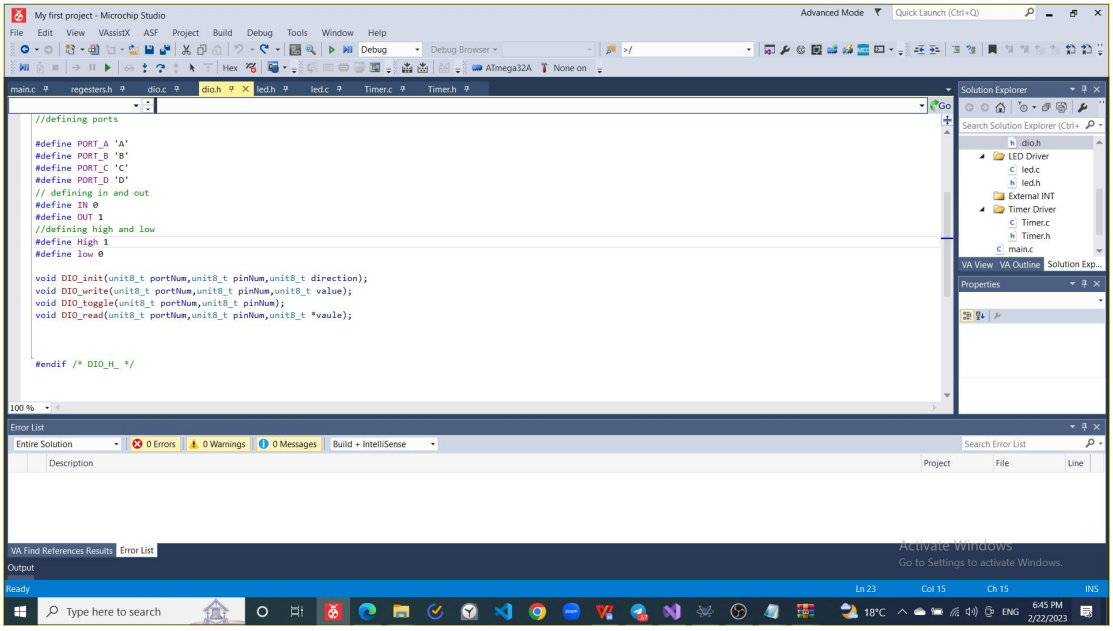
Layer 1

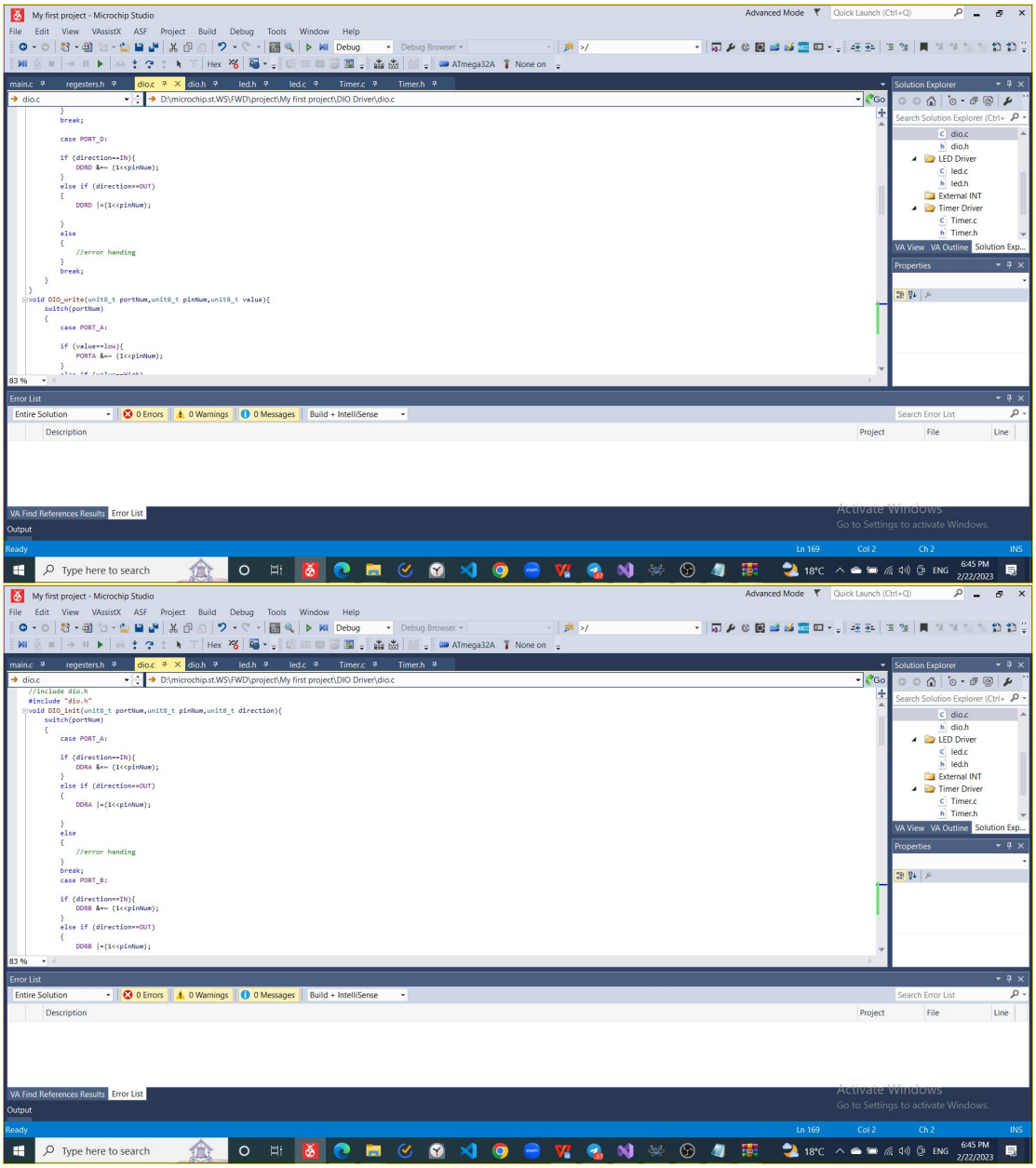
Registers
Typedefs



Layer 2

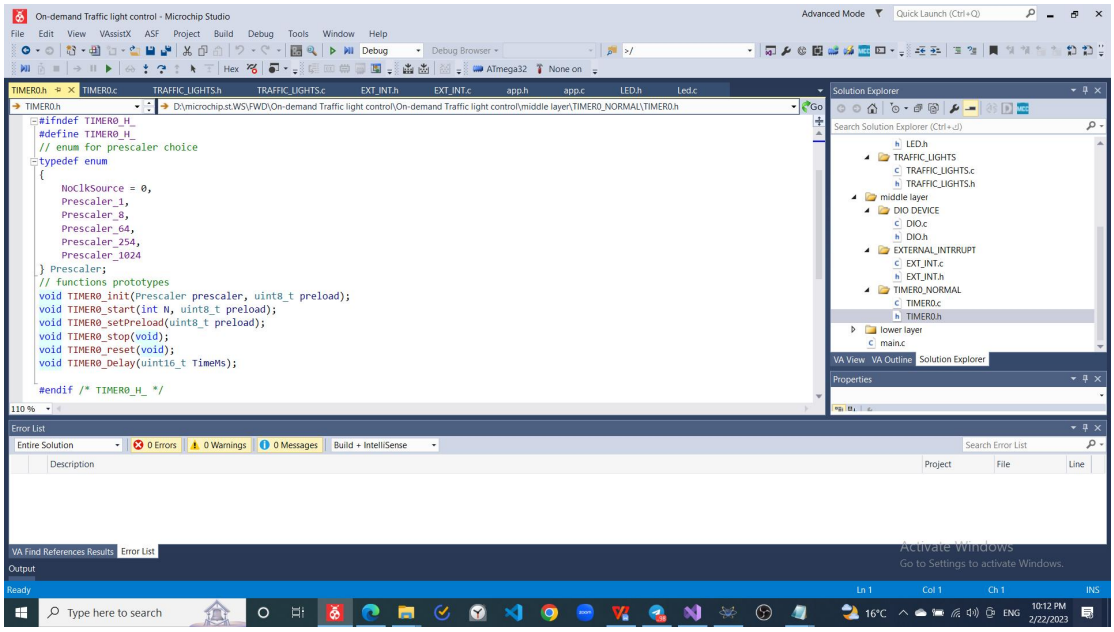
dio device

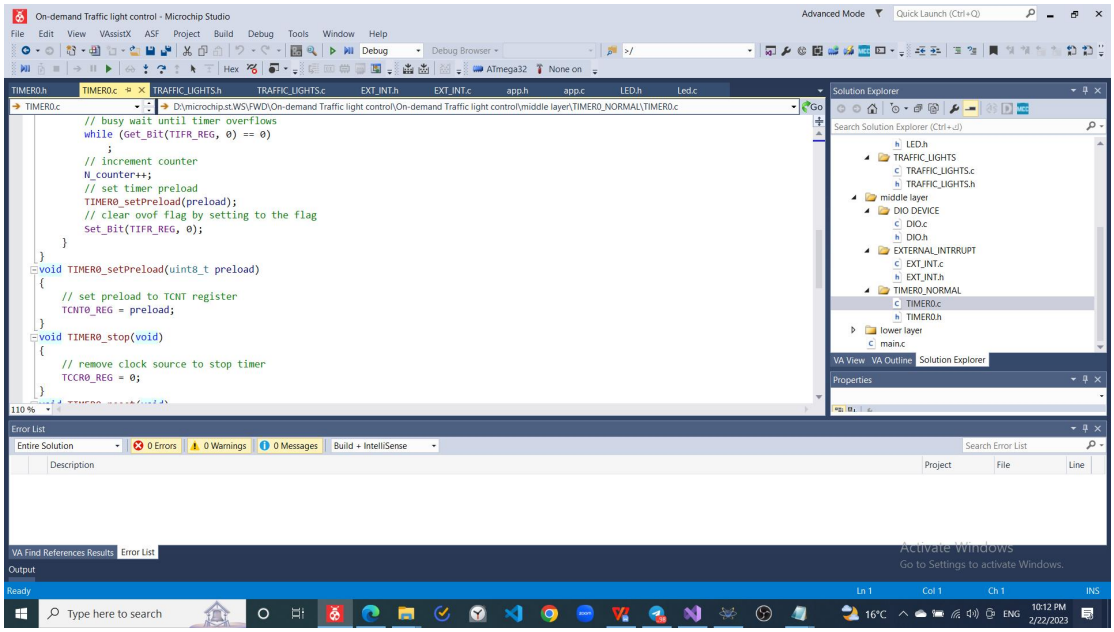




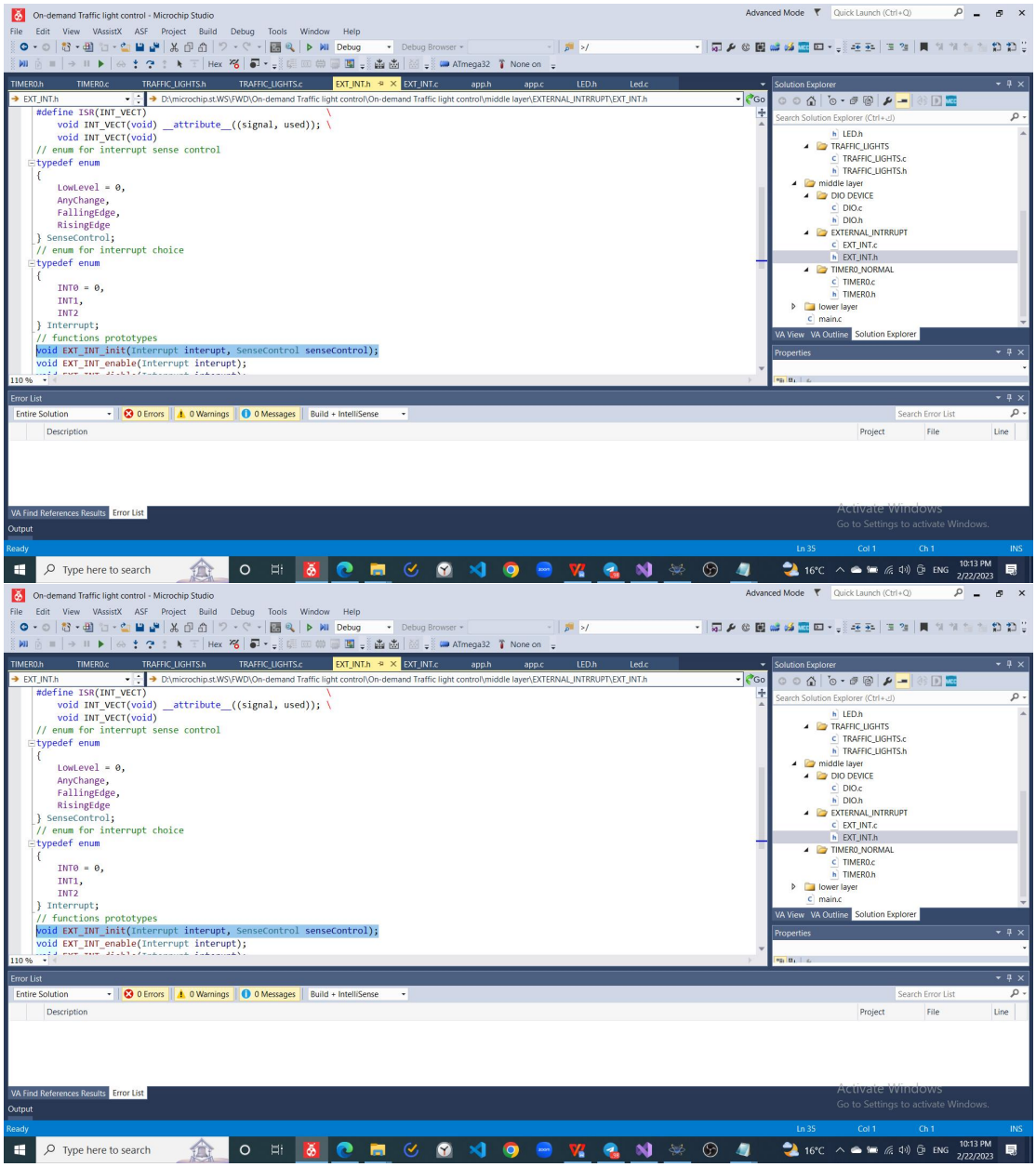
Layer 3

timer



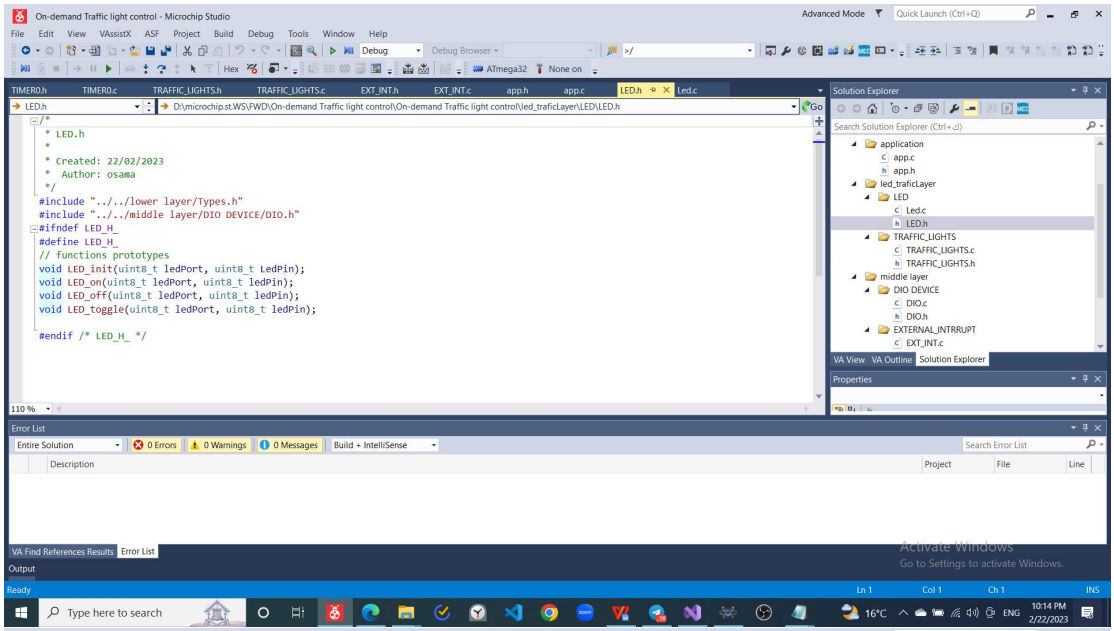


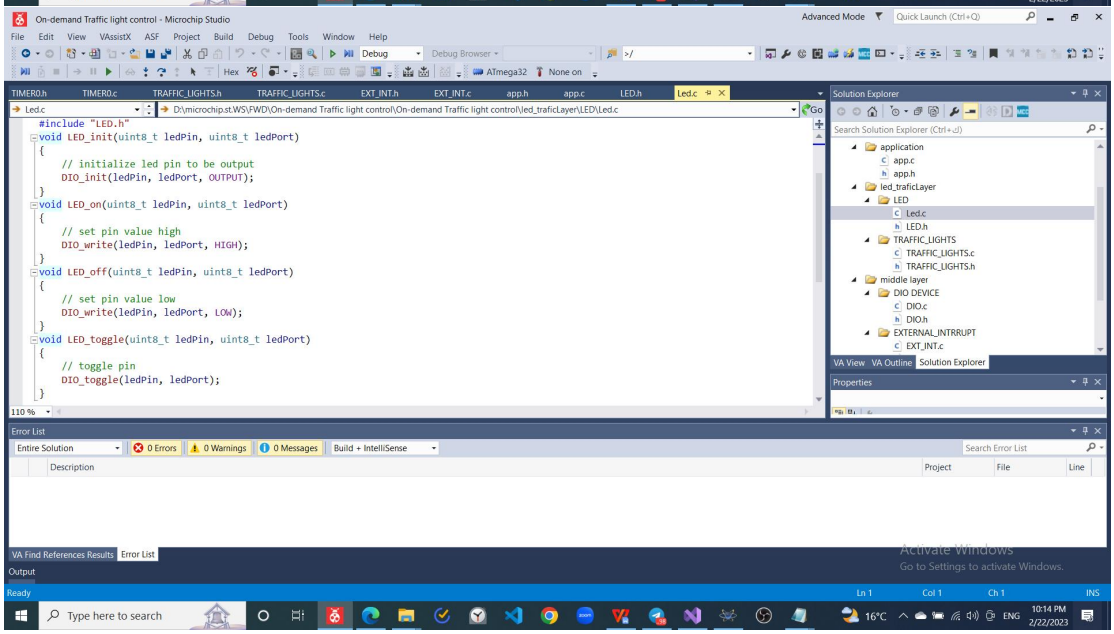
Layer 4 : external interrupt


On-demand Traffic light control - Microchip Studio
#define ISR(INT_VECT) \
void INT_VECT(void) __attribute__((signal, used)); \
void INT_VECT(void)
// enum for interrupt sense control
typedef enum
{
LowLevel = 0,
AnyChange,
FallingEdge,
RisingEdge
} SenseControl;
// enum for interrupt choice
typedef enum
{
INT0 = 0,
INT1,
INT2
} Interrupt;
// functions prototypes
void EXT_INT_init(Interrupt interupt, SenseControl senseControl);
void EXT_INT_enable(Interrupt interupt);
Solution Explorer
LED.h
TRAFFIC_LIGHTS
TRAFFIC_LIGHTS.c
TRAFFIC_LIGHTS.h
middle layer
DIO DEVICE
DIO.c
DIO.h
EXTERNAL_INTRRUPT
EXT_INT.c
EXT_INT.h
TIMER0_NORMAL
TIMER0.c
TIMER0.h
lower layer
main.c
Error List
Activate Windows
Go to Settings to activate Windows.

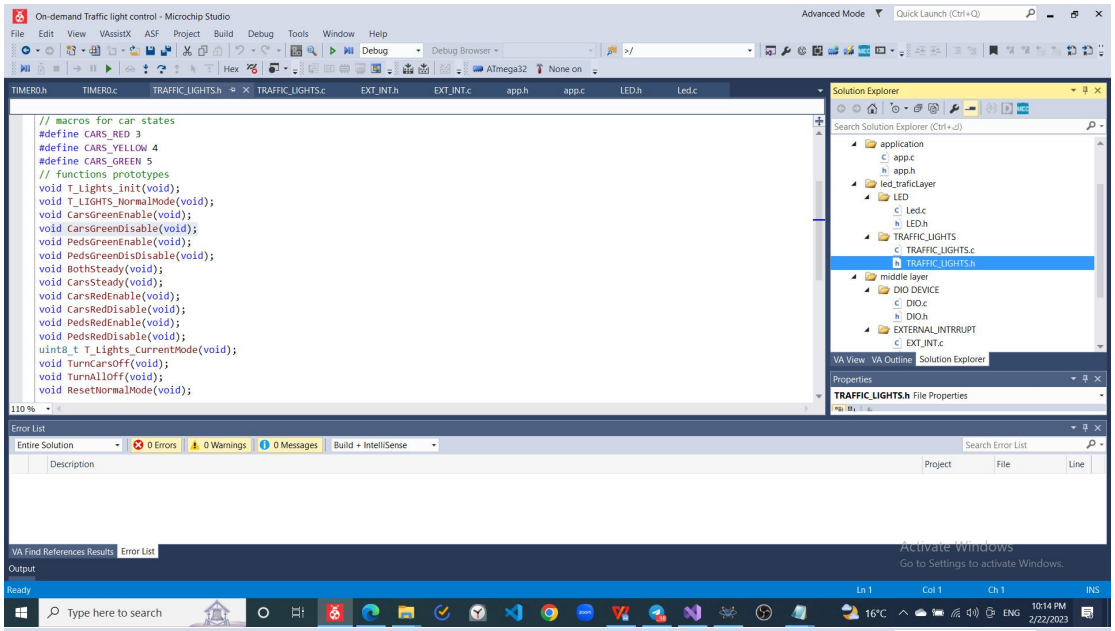
On-demand Traffic light control - Microchip Studio
#define ISR(INT_VECT) \
void INT_VECT(void) __attribute__((signal, used)); \
void INT_VECT(void)
// enum for interrupt sense control
typedef enum
{
LowLevel = 0,
AnyChange,
FallingEdge,
RisingEdge
} SenseControl;
// enum for interrupt choice
typedef enum
{
INT0 = 0,
INT1,
INT2
} Interrupt;
// functions prototypes
void EXT_INT_init(Interrupt interupt, SenseControl senseControl);
void EXT_INT_enable(Interrupt interupt);
Solution Explorer
LED.h
TRAFFIC_LIGHTS
TRAFFIC_LIGHTS.c
TRAFFIC_LIGHTS.h
middle layer
DIO DEVICE
DIO.c
DIO.h
EXTERNAL_INTRRUPT
EXT_INT.c
EXT_INT.h
TIMER0_NORMAL
TIMER0.c
TIMER0.h
lower layer
main.c
Error List
Activate Windows
Go to Settings to activate Windows.

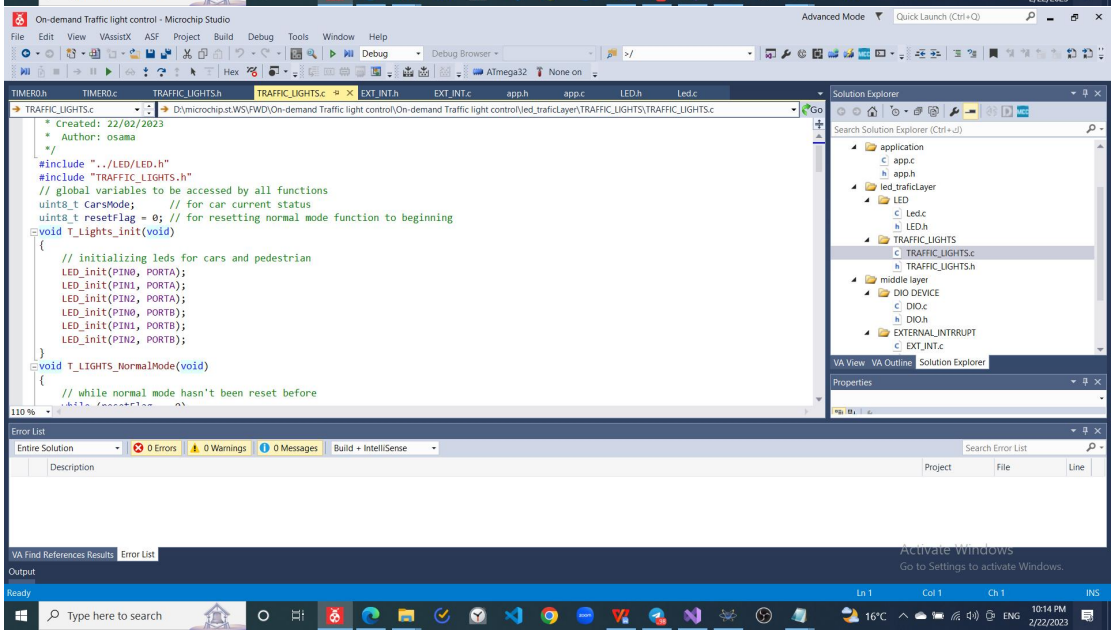
Layer 5: led

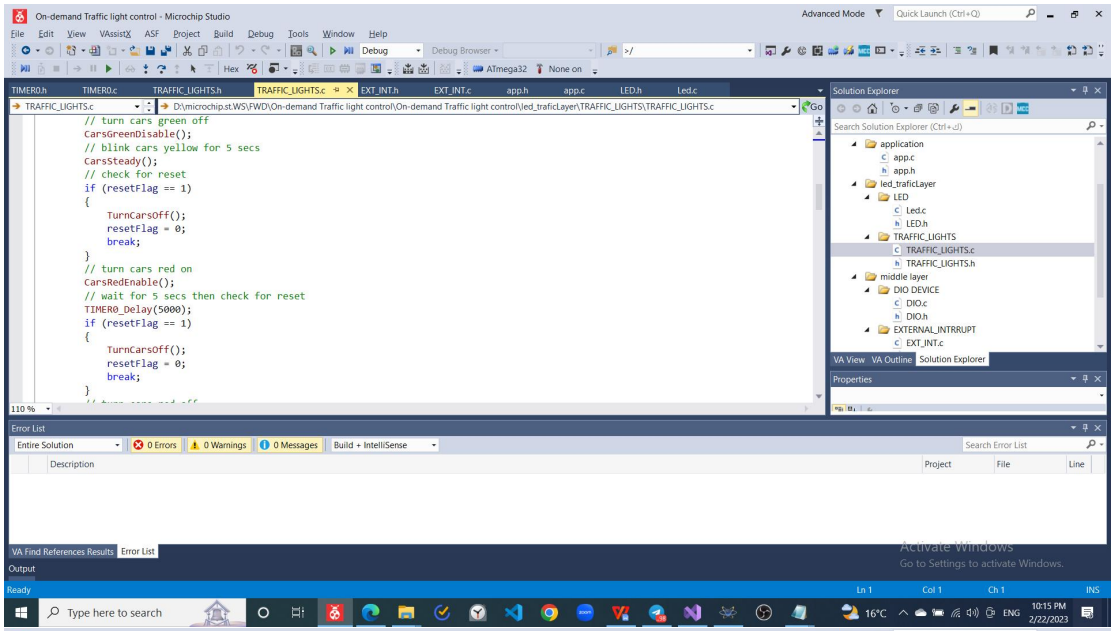


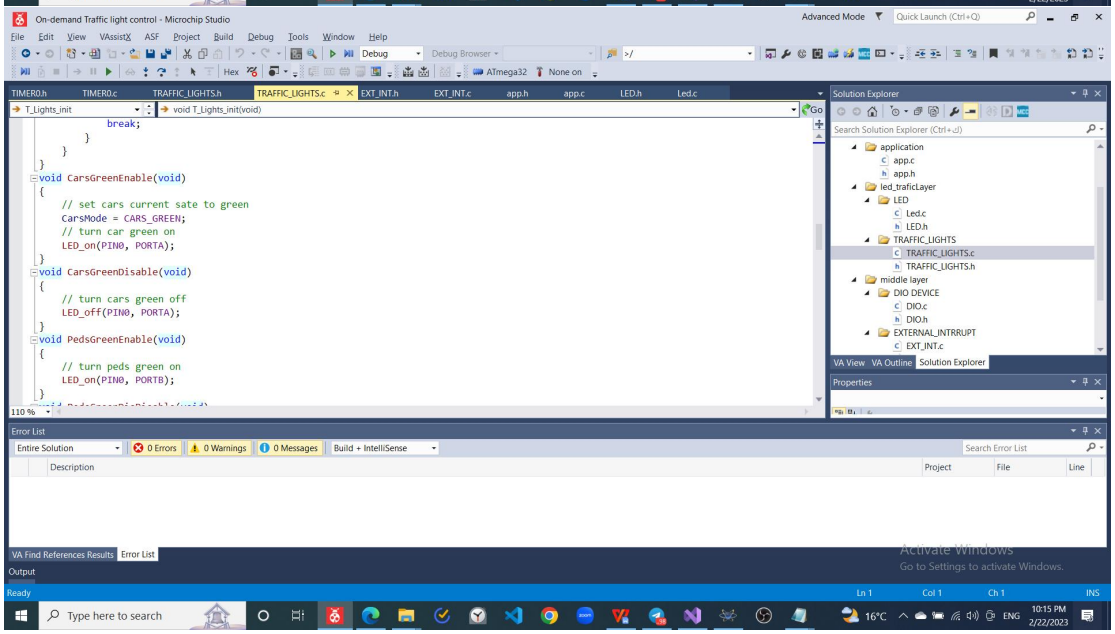


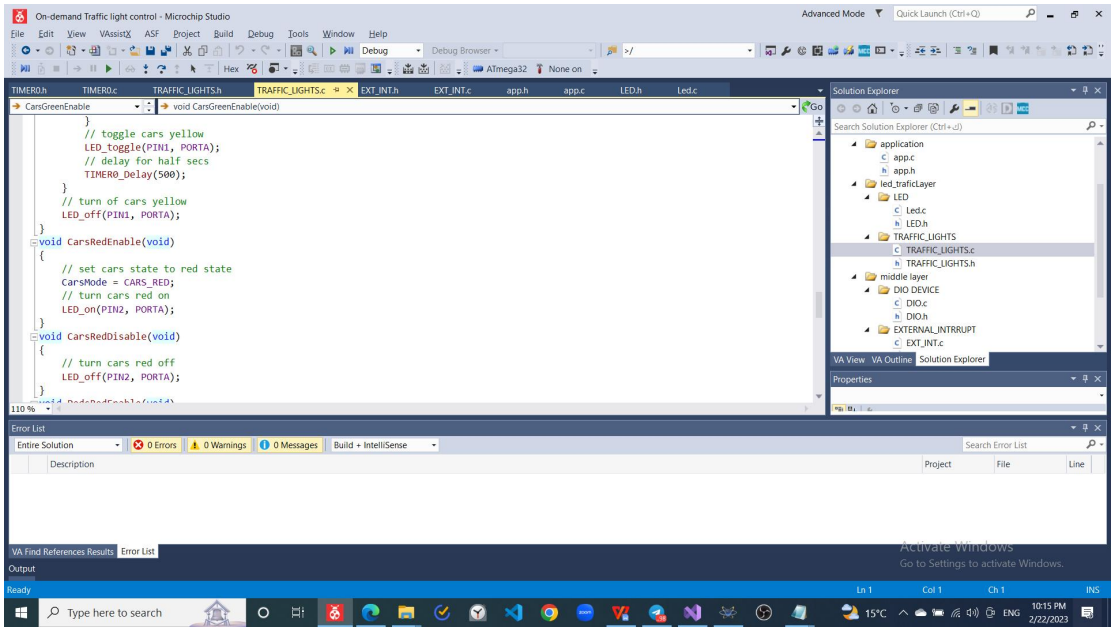
Layer 6: full traffic light control


On-demand Traffic light control - Microchip Studio
TRAFFIC_LIGHTS.h
TRAFFIC_LIGHTS.c
// macros for car states
#define CARS_RED 3
#define CARS_YELLOW 4
#define CARS_GREEN 5
// functions prototypes
void T_Lights_init(void);
void T_LIGHTS_NormalMode(void);
void CarsGreenEnable(void);
void CarsGreenDisable(void);
void PedsGreenEnable(void);
void PedsGreenDisDisable(void);
void BothSteady(void);
void CarsSteady(void);
void CarsRedEnable(void);
void CarsRedDisable(void);
void PedsRedEnable(void);
void PedsRedDisable(void);
uint8_t T_Lights_CurrentMode(void);
void TurnCarsOff(void);
void TurnAlloff(void);
void ResetNormalMode(void);
Solution Explorer
application
app.c
app.h
led_traficLayer
LED
Led.c
LED.h
TRAFFIC_LIGHTS
TRAFFIC_LIGHTS.c
TRAFFIC_LIGHTS.h
middle layer
DIO DEVICE
DIO.c
DIO.h
EXTERNAL_INTRRUPT
EXT_INT.c
Properties
Error List
Description
Project
File
Line
Activate Windows
Go to Settings to activate Windows.
Ready

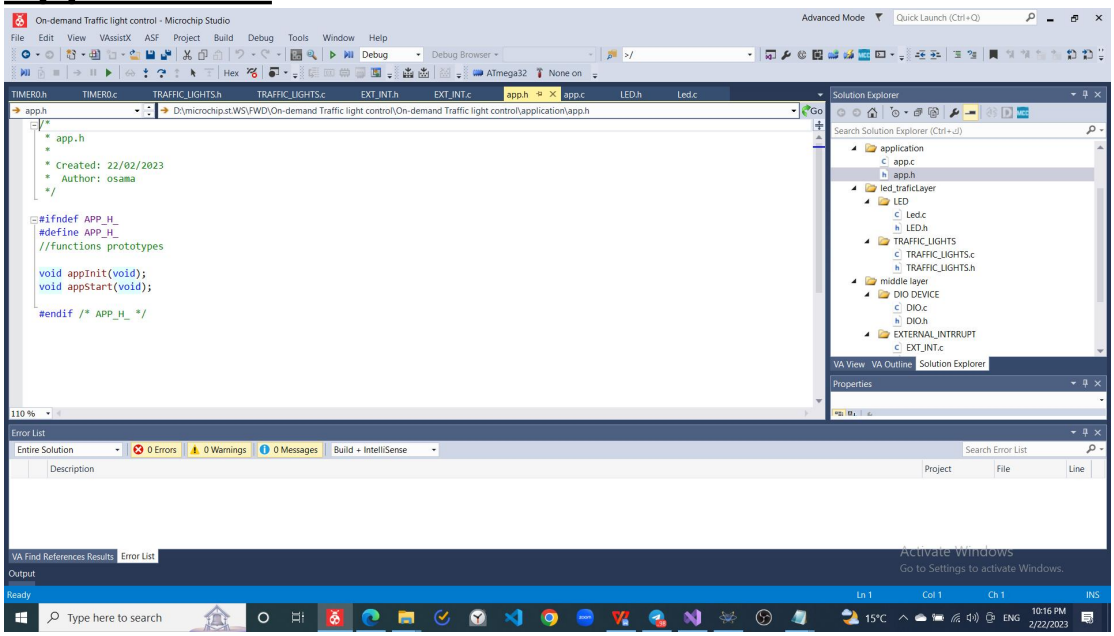

On-demand Traffic light control - Microchip Studio
TRAFFIC_LIGHTS.c
D:\microchip.st.WS\FWD\On-demand Traffic light control\On-demand Traffic light control\led_traficLayer\TRAFFIC_LIGHTS\TRAFFIC_LIGHTS.c
* Created: 22/02/2023
* Author: osama
#include "../LED/LED.h"
#include "TRAFFIC_LIGHTS.h"
// global variables to be accessed by all functions
uint8_t CarsMode;   // for car current status
uint8_t resetFlag = 0; // for resetting normal mode function to beginning
void T_Lights_init(void)
{
// initializing leds for cars and pedestrian
LED_init(PIN0, PORTA);
LED_init(PIN1, PORTA);
LED_init(PIN2, PORTA);
LED_init(PIN0, PORTB);
LED_init(PIN1, PORTB);
LED_init(PIN2, PORTB);
}
void T_LIGHTS_NormalMode(void)
{
// while normal mode hasn't been reset before
Solution Explorer
application
app.c
app.h
led_traficLayer
LED
Led.c
LED.h
TRAFFIC_LIGHTS
TRAFFIC_LIGHTS.c
TRAFFIC_LIGHTS.h
middle layer
DIO DEVICE
DIO.c
DIO.h
EXTERNAL_INTRRUPT
EXT_INT.c
Properties
Error List
Description
Project
File
Line
Activate Windows
Go to Settings to activate Windows.
Ready

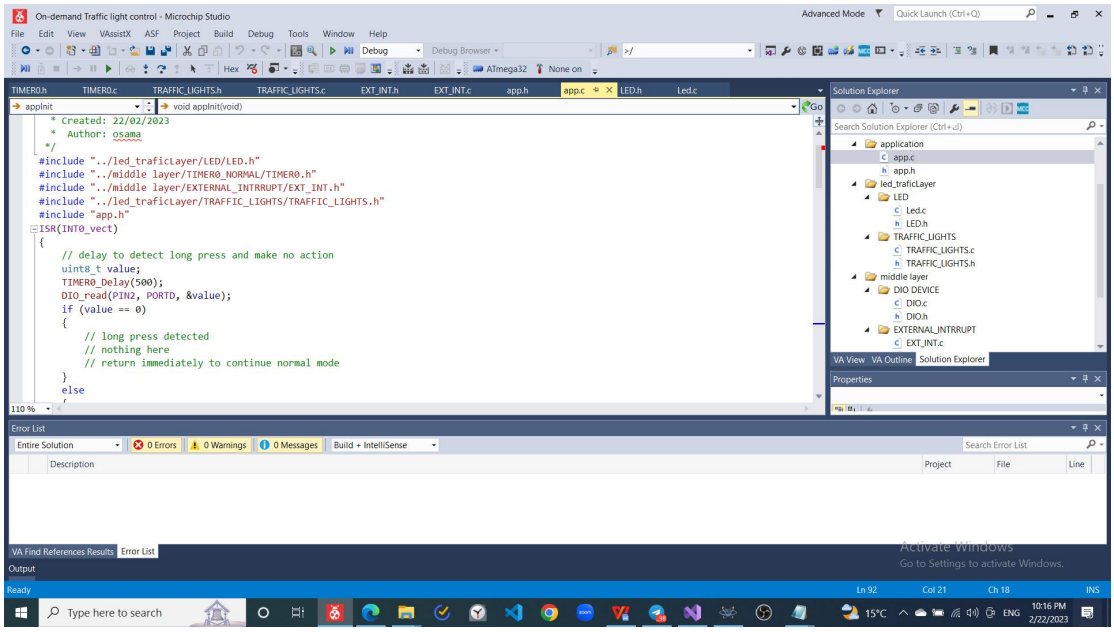

On-demand Traffic light control - Microchip Studio
// turn cars green off
CarsGreenDisable();
// blink cars yellow for 5 secs
CarsSteady();
// check for reset
if (resetFlag == 1)
{
TurnCarsOff();
resetFlag = 0;
break;
}
// turn cars red on
CarsRedEnable();
// wait for 5 secs then check for reset
TIMER0_Delay(5000);
if (resetFlag == 1)
{
TurnCarsOff();
resetFlag = 0;
break;
}
Solution Explorer
application
app.c
app.h
led_traficLayer
LED
Led.c
LED.h
TRAFFIC_LIGHTS
TRAFFIC_LIGHTS.c
TRAFFIC_LIGHTS.h
middle layer
DIO DEVICE
DIO.c
DIO.h
EXTERNAL_INTRRUPT
EXT_INT.c
Error List
Activate Windows
Go to Settings to activate Windows.

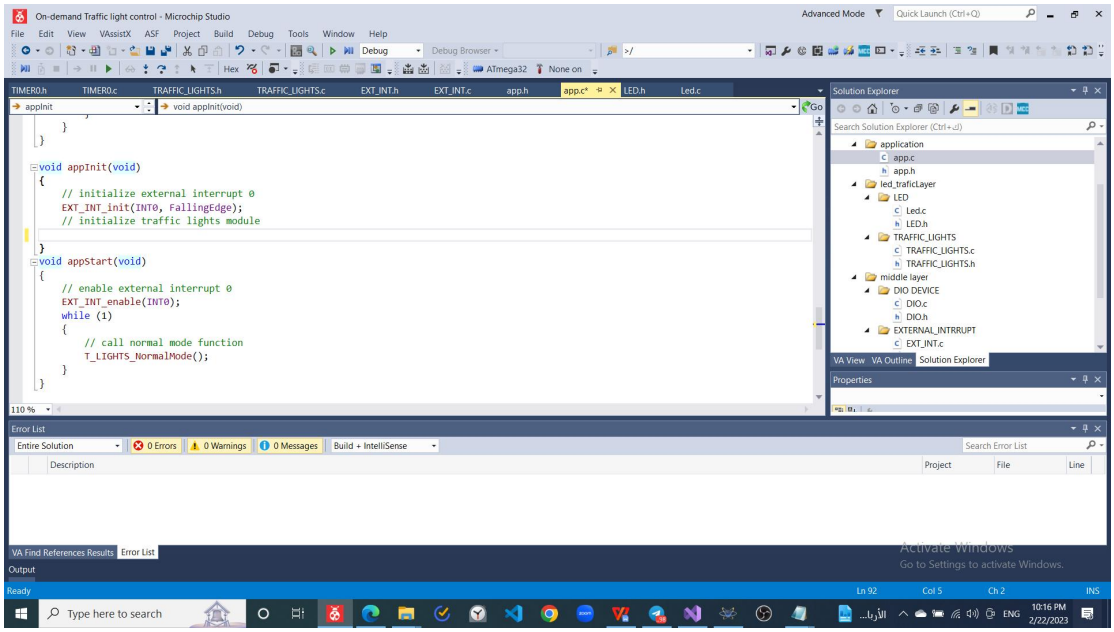

On-demand Traffic light control - Microchip Studio
break;
}
}
}
void CarsGreenEnable(void)
{
// set cars current sate to green
CarsMode = CARS_GREEN;
// turn car green on
LED_on(PIN0, PORTA);
}
void CarsGreenDisable(void)
{
// turn cars green off
LED_off(PIN0, PORTA);
}
void PedsGreenEnable(void)
{
// turn peds green on
LED_on(PIN0, PORTB);
}
Solution Explorer
application
app.c
app.h
led_traficLayer
LED
Led.c
LED.h
TRAFFIC_LIGHTS
TRAFFIC_LIGHTS.c
TRAFFIC_LIGHTS.h
middle layer
DIO DEVICE
DIO.c
DIO.h
EXTERNAL_INTRRUPT
EXT_INT.c
Error List
Activate Windows
Go to Settings to activate Windows.



Application:




On-demand Traffic light control - Microchip Studio
Advanced Mode
Quick Launch (Ctrl+Q)
File Edit View VAssistX ASF Project Build Debug Tools Window Help
Debug
Debug Browser
ATmega32
None on
TIMER0.h TIMER0.c TRAFFIC_LIGHTS.h TRAFFIC_LIGHTS.c EXT_INT.h EXT_INT.c app.h app.c LED.h Led.c
appInit
void appInit(void)
* Created: 22/02/2023
* Author: osama
*/
#include "../led_traficLayer/LED/LED.h"
#include "../middle layer/TIMER0_NORMAL/TIMER0.h"
#include "../middle layer/EXTERNAL_INTRRUPT/EXT_INT.h"
#include "../led_traficLayer/TRAFFIC_LIGHTS/TRAFFIC_LIGHTS.h"
#include "app.h"
ISR(INT0_vect)
{
// delay to detect long press and make no action
uint8_t value;
TIMER0_Delay(500);
DIO_read(PIN2, PORTD, &value);
if (value == 0)
{
// long press detected
// nothing here
// return immediately to continue normal mode
}
else
Solution Explorer
Search Solution Explorer (Ctrl+;)
application
app.c
app.h
led_traficLayer
LED
Led.c
LED.h
TRAFFIC_LIGHTS
TRAFFIC_LIGHTS.c
TRAFFIC_LIGHTS.h
middle layer
DIO DEVICE
DIO.c
DIO.h
EXTERNAL_INTRRUPT
EXT_INT.c
VA View VA Outline Solution Explorer
Properties
110 %
Error List
Entire Solution
0 Errors
0 Warnings
0 Messages
Build + IntelliSense
Search Error List
Description
Project
File
Line
VA Find References Results Error List
Output
Activate Windows
Go to Settings to activate Windows.
Ready
Ln 92
Col 21
Ch 18
INS
Type here to search
15°C
ENG
10:16 PM
2/22/2023


On-demand Traffic light control - Microchip Studio
Advanced Mode
Quick Launch (Ctrl+Q)
File Edit View VAssistX ASF Project Build Debug Tools Window Help
Debug
Debug Browser
ATmega32
None on
TIMER0.h TIMER0.c TRAFFIC_LIGHTS.h TRAFFIC_LIGHTS.c EXT_INT.h EXT_INT.c app.h app.c* LED.h Led.c
appInit
void appInit(void)
}
}
void appInit(void)
{
// initialize external interrupt 0
EXT_INT_init(INT0, FallingEdge);
// initialize traffic lights module
}
void appStart(void)
{
// enable external interrupt 0
EXT_INT_enable(INT0);
while (1)
{
// call normal mode function
T_LIGHTS_NormalMode();
}
}
Solution Explorer
Search Solution Explorer (Ctrl+;)
application
app.c
app.h
led_traficLayer
LED
Led.c
LED.h
TRAFFIC_LIGHTS
TRAFFIC_LIGHTS.c
TRAFFIC_LIGHTS.h
middle layer
DIO DEVICE
DIO.c
DIO.h
EXTERNAL_INTRRUPT
EXT_INT.c
VA View VA Outline Solution Explorer
Properties
110 %
Error List
Entire Solution
0 Errors
0 Warnings
0 Messages
Build + IntelliSense
Search Error List
Description
Project
File
Line
VA Find References Results Error List
Output
Activate Windows
Go to Settings to activate Windows.
Ready
Ln 92
Col 5
Ch 2
INS
Type here to search
ENG
10:16 PM
2/22/2023