Mastering Embedded System Online Diploma

www.learn-in-depth.com

First Term (Final Project1)

Report for “High_Pressure_Detection” project

Name: Osama Khallaf Samir Moktar

Mail: osamakhallaf0285@gmail.com

Githup repo: https://github.com/Osama485/Embedded_System_Online_Diploma.git

My profile: https://www.learn-in-depth-store.com/certificate/osamakhallaf0285%40gmail.com

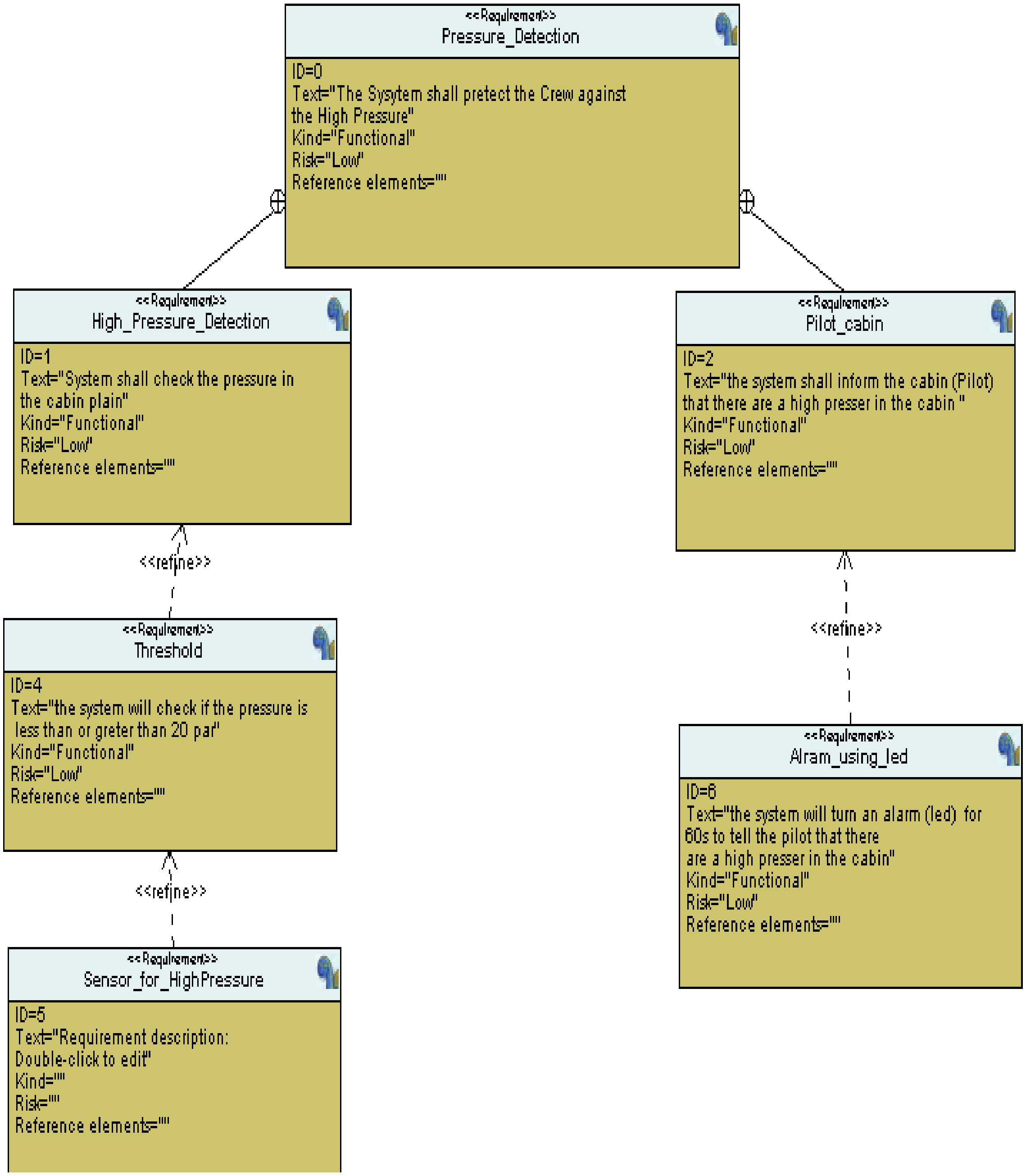
Case Study

The system should be in a plain cabin, if the pressure in the cabin is higher we will put a sensor -to sense the high pressure- and connected with an Alarm to tell the crew in the cabin that there is a high pressure to be safe

The Sensor should make Alarm if the pressure is greater than 20 bar.

Then the Alarm -The led- will turn on 60s and the Alarm still turn on if the pressure is greater than 20bar to tell the crew or the Pilot that there are a danger on the plain.

Requirements Diagram



ID=0
Text="The Sysytem shall pretect the Crew against the High Pressure"
Kind="Functional"
Risk="Low"
Reference elements=""

ID=1
Text="System shall check the pressure in the cabin plain"
Kind="Functional"
Risk="Low"
Reference elements=""

ID=2
Text="the system shall inform the cabin (Pilot) that there are a high presser in the cabin "
Kind="Functional"
Risk="Low"
Reference elements=""

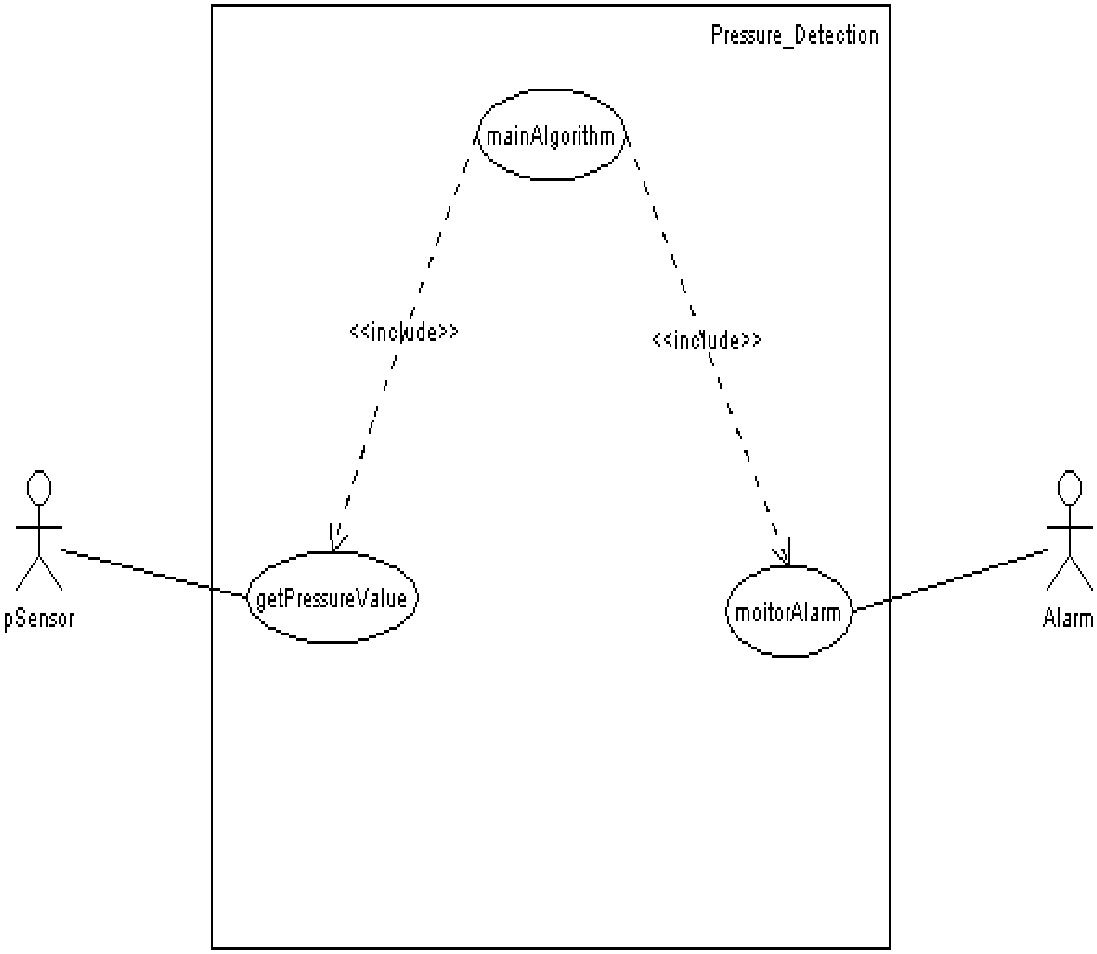
ID=4
Text="the system will check if the pressure is less than or greter than 20 par"
Kind="Functional"
Risk="Low"
Reference elements=""

ID=6
Text="the system will turn an alarm (led) for 60s to tell the pilot that there are a high presser in the cabin"
Kind="Functional"
Risk="Low"
Reference elements=""

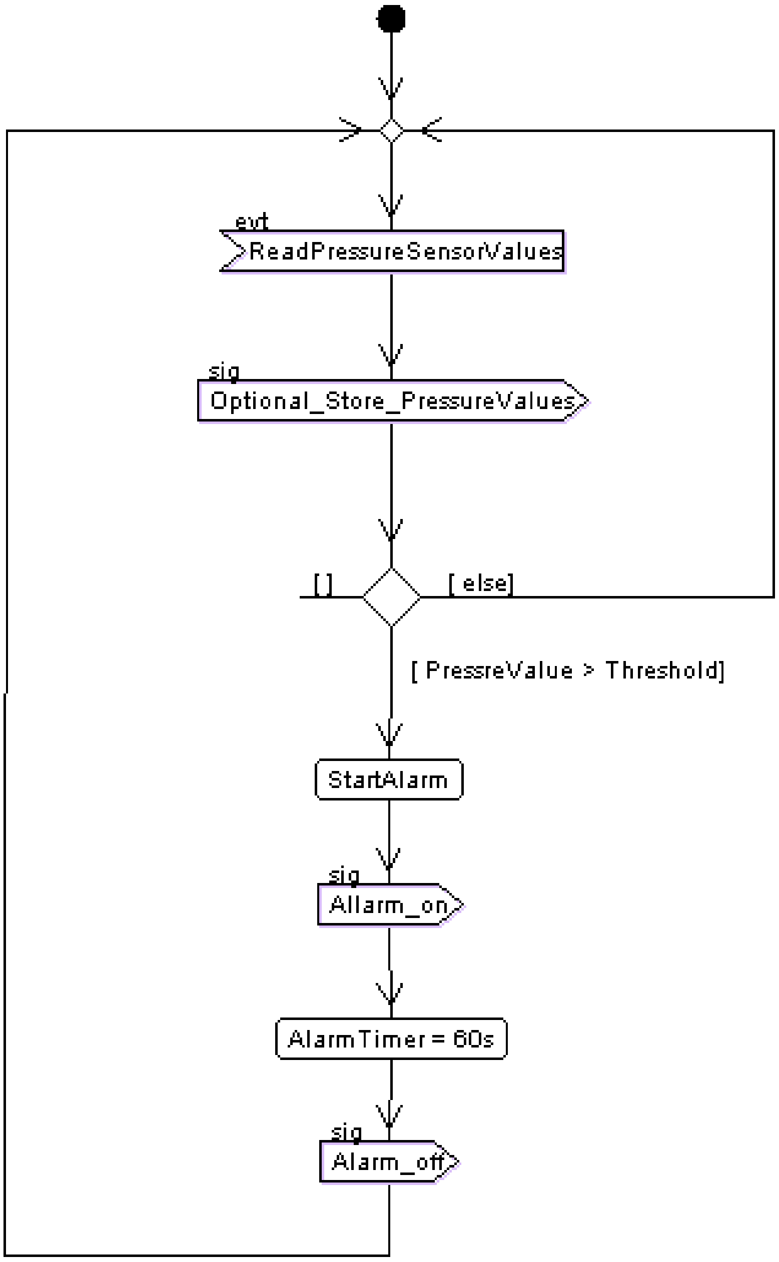
ID=5
Text="Requirement description: Double-click to edit"
Kind=""
Risk=""
Reference elements=""

System Analysis

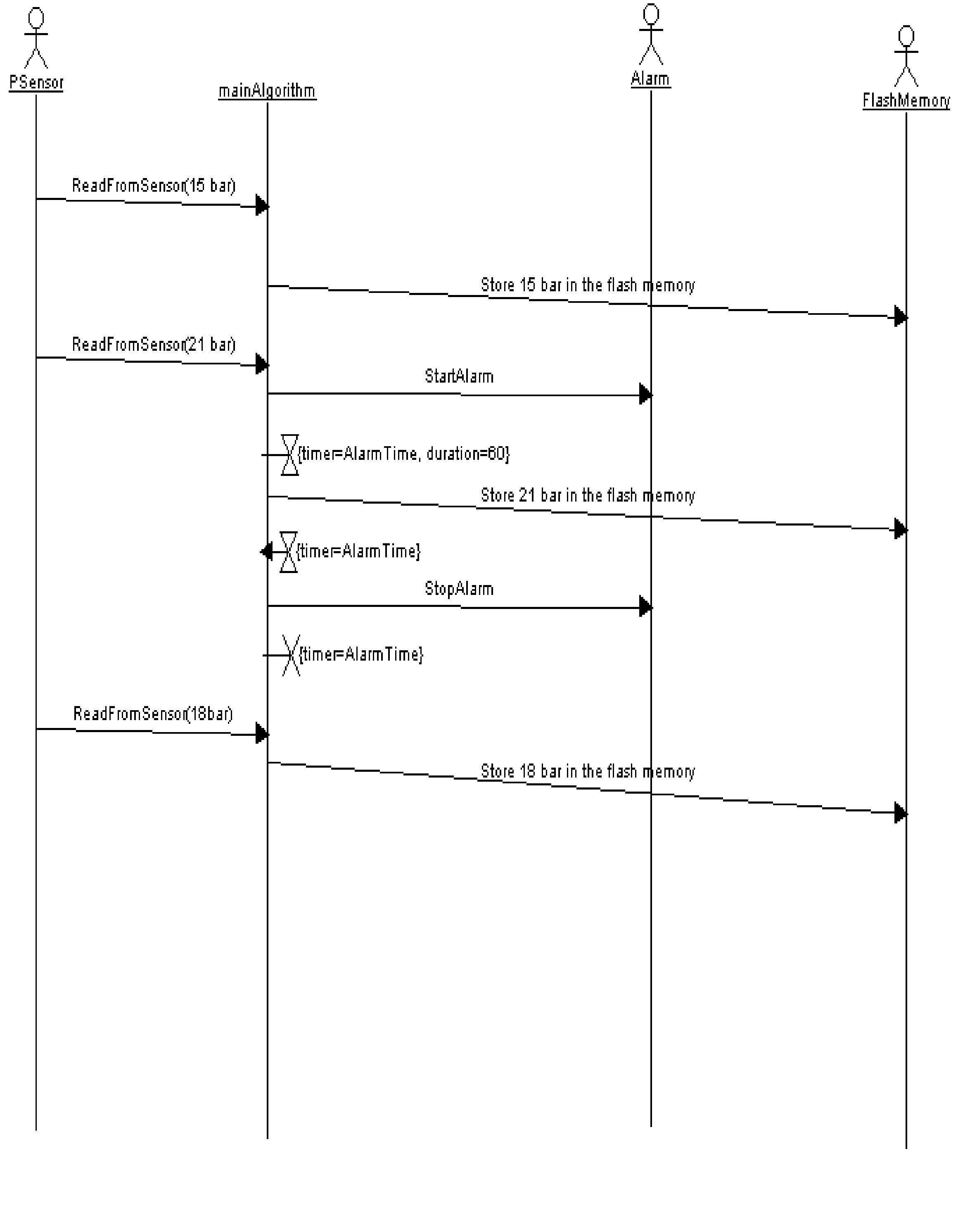
1. Usecase Diagram



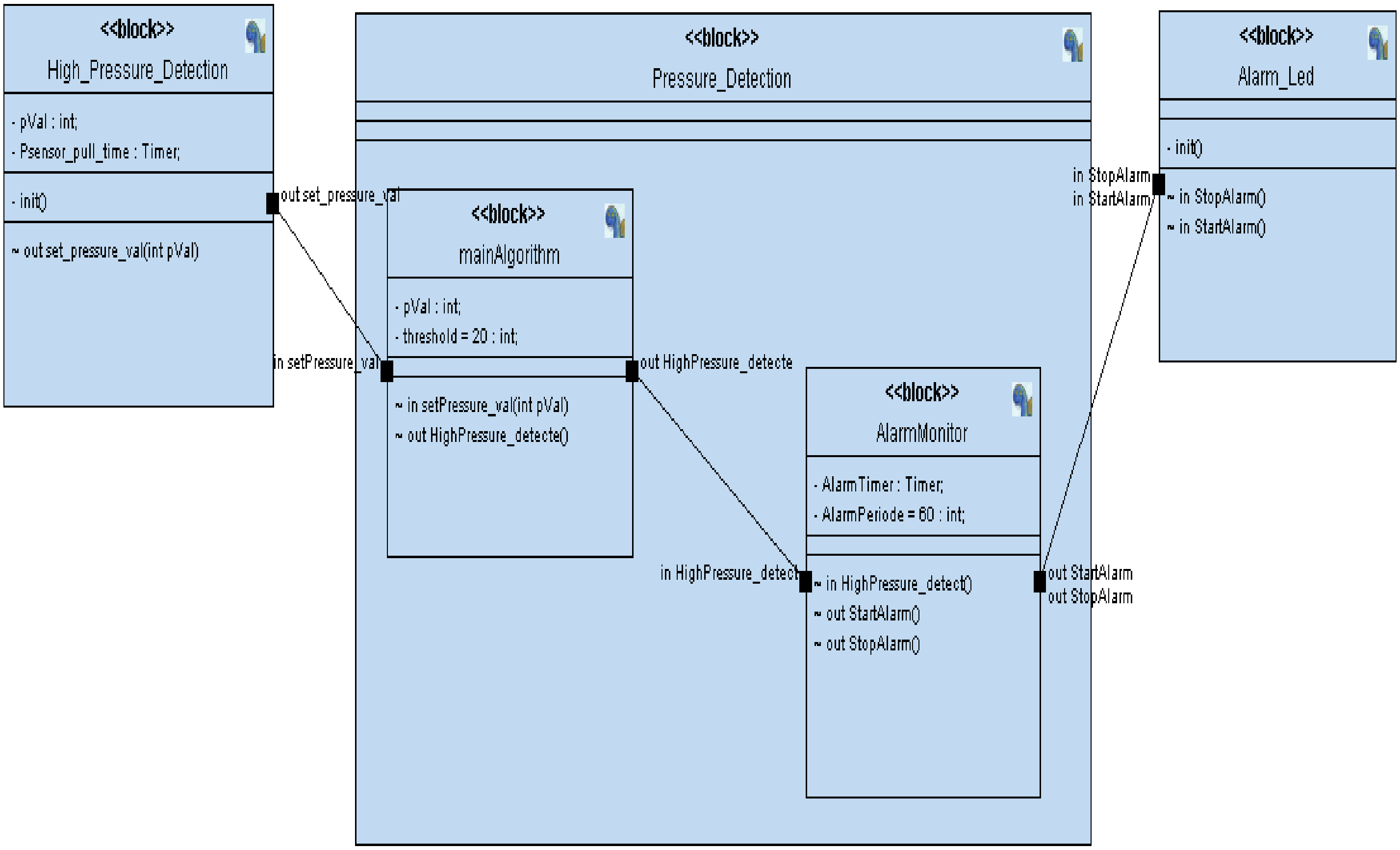
2. Activity Diagram



3. Sequence Diagram

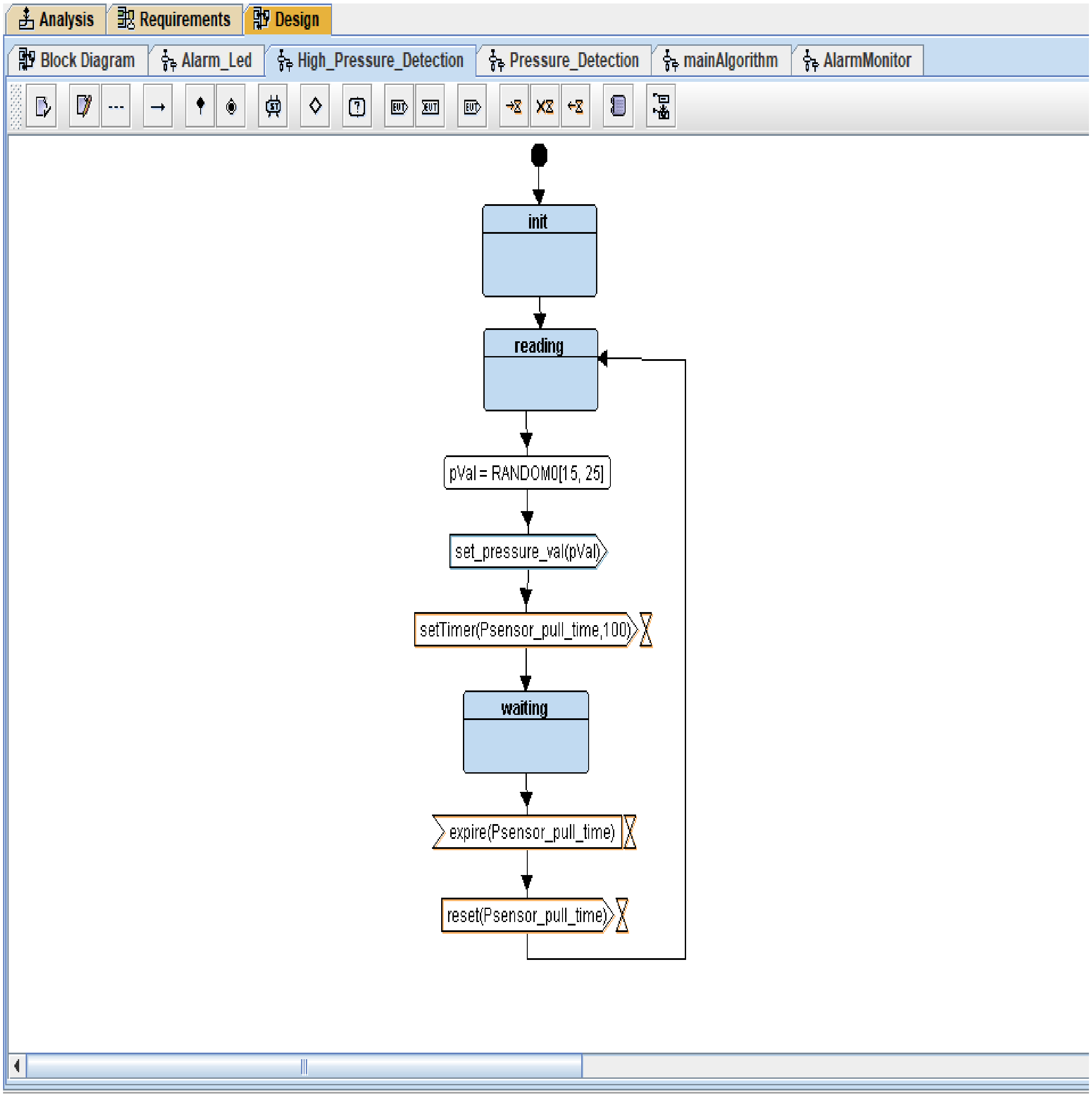


System Design Diagram

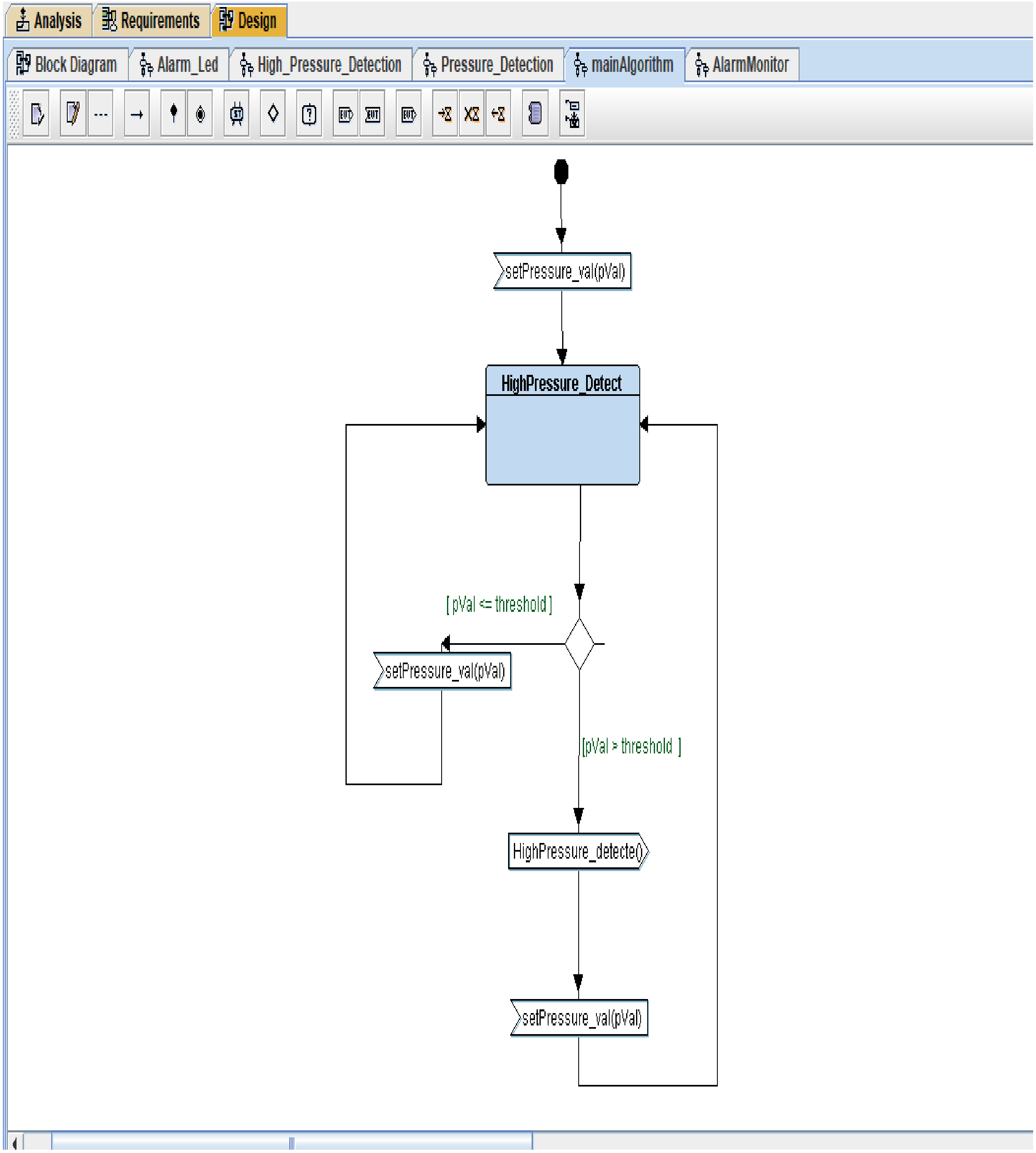


State Machine of Design Diagram

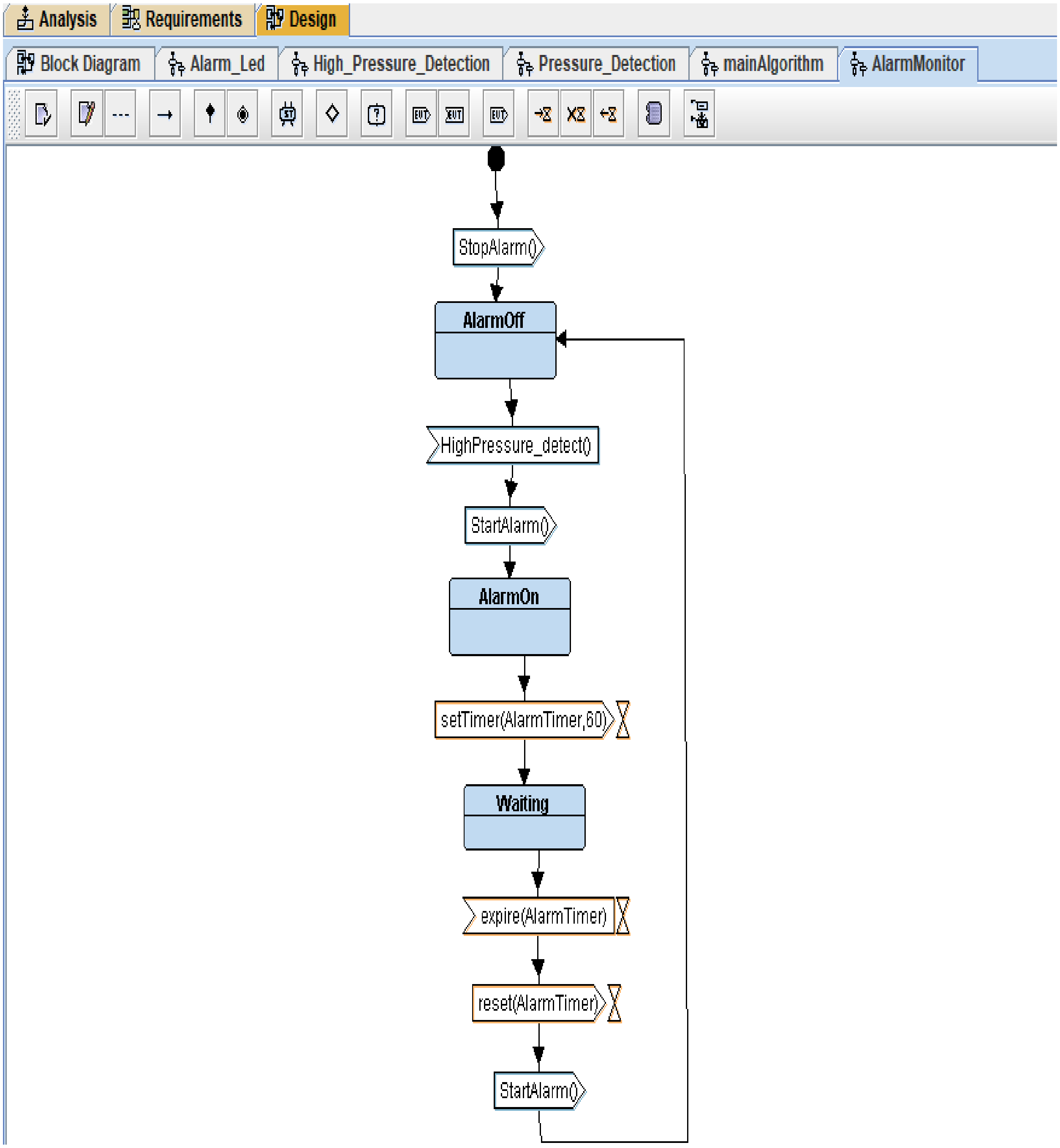
1. High_Pressure_Detection



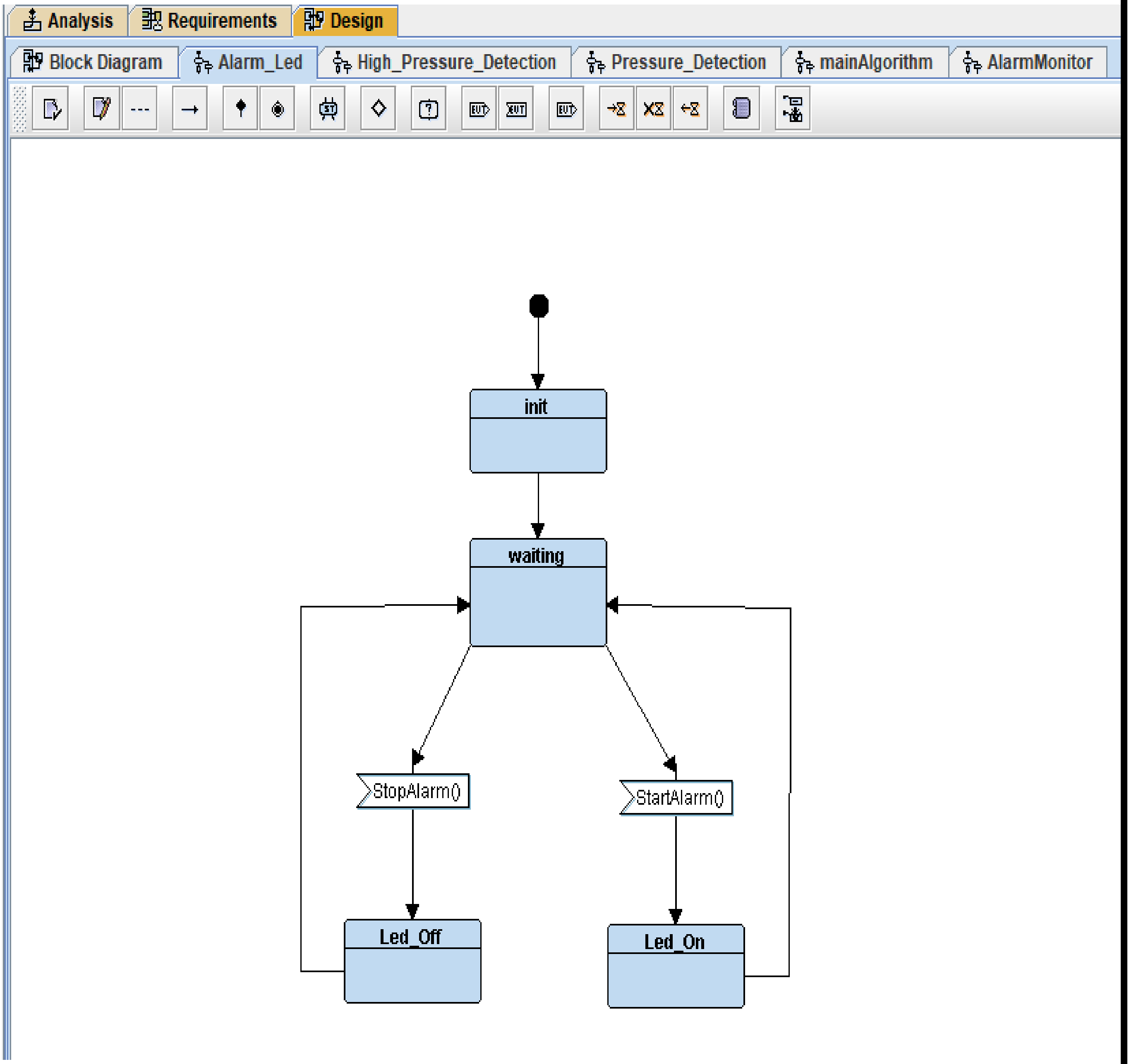
2. *Main_Algorithm*



3. Alarm_Monitor

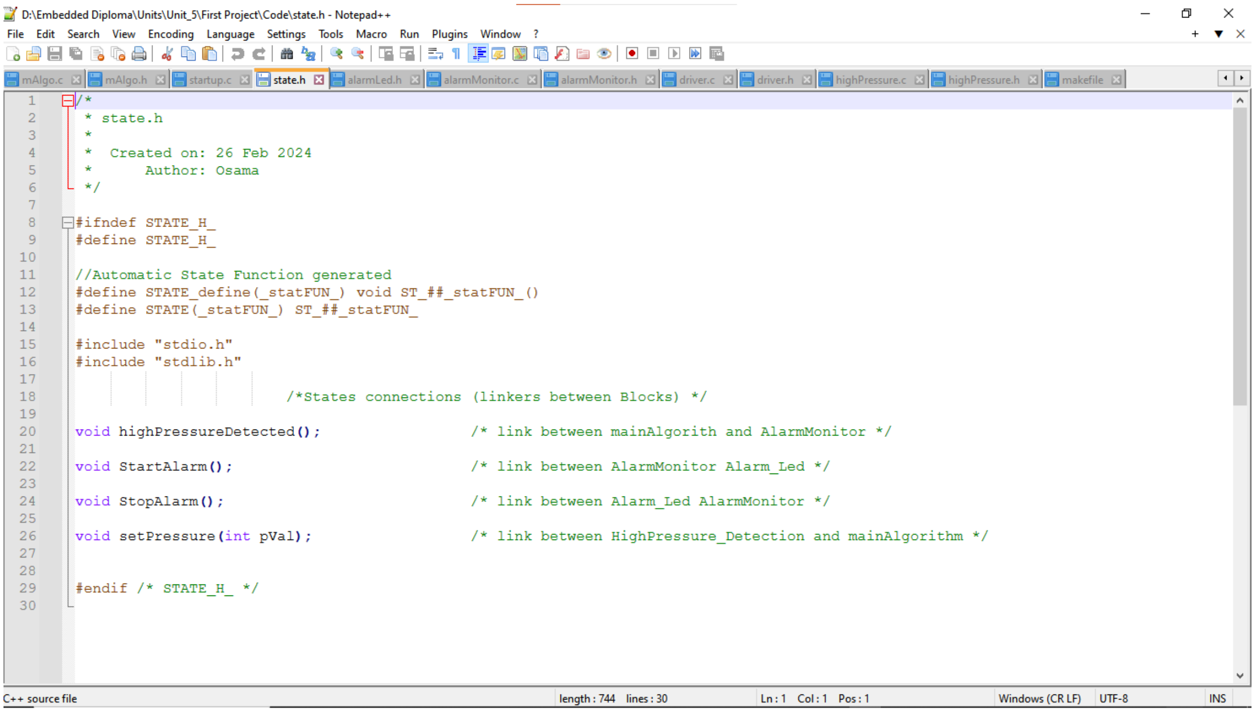


4. Alarm_Acuator(led)



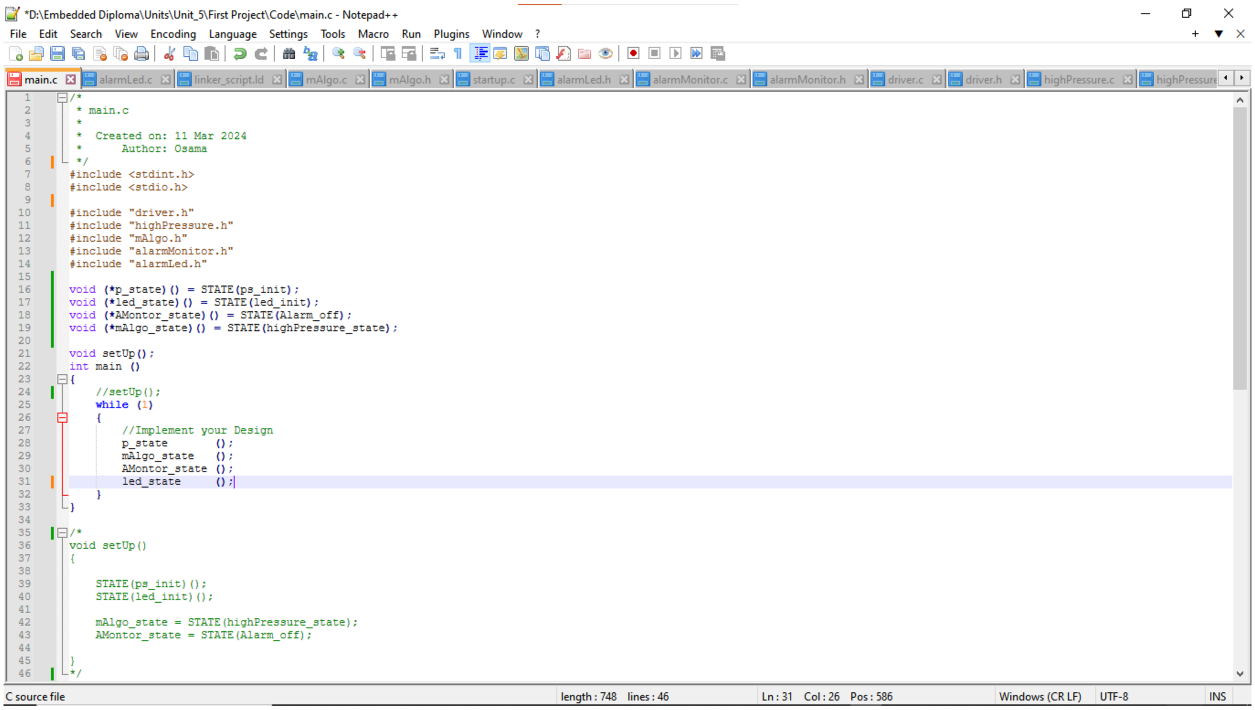
Files

1. State.h


D:\Embedded Diploma\Units\Unit_5\First Project\Code\state.h - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
mAlgo.c mAlgo.h startup.c state.h alarmLed.h alarmMonitor.c alarmMonitor.h driver.c driver.h highPressure.c highPressure.h makefile
C++ source file length : 744 lines : 30 Ln : 1 Col : 1 Pos : 1 Windows (CR LF) UTF-8 INS

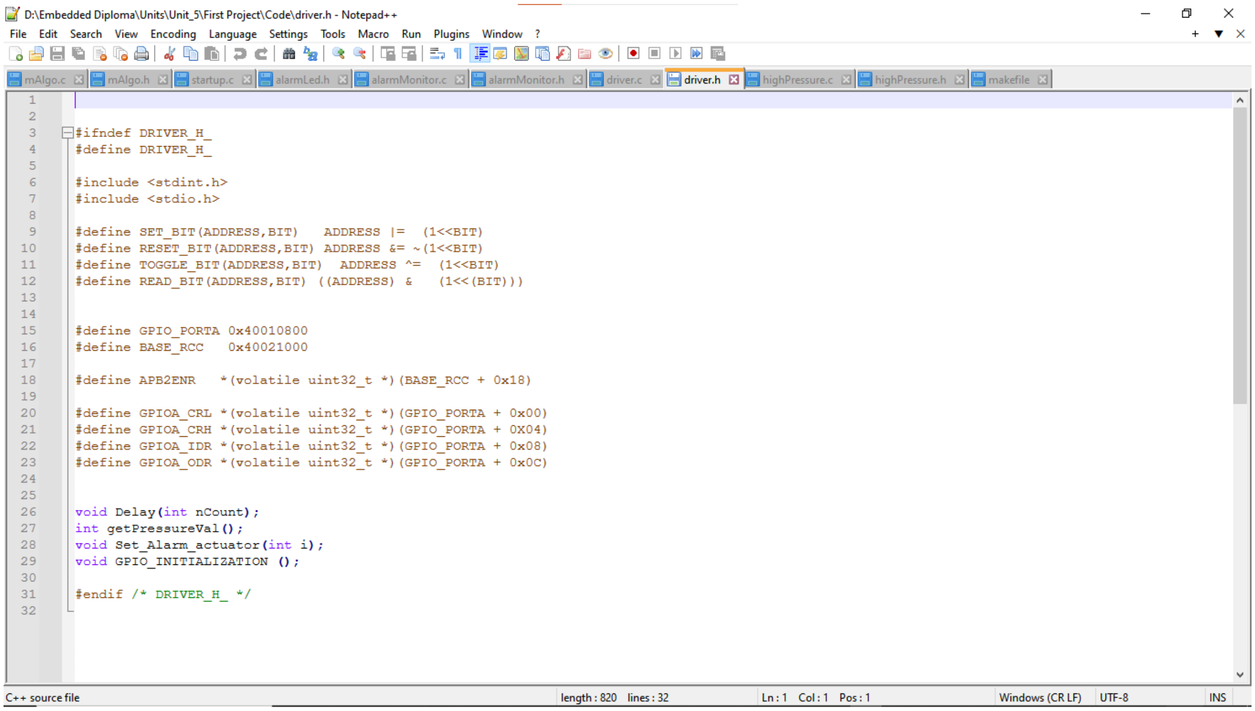
```
/*
 * state.h
 *
 *  Created on: 26 Feb 2024
 *      Author: Osama
 */

#ifndef STATE_H_
#define STATE_H_

//Automatic State Function generated
#define STATE_define(_statFUN_) void ST_##_statFUN_()
#define STATE(_statFUN_) ST_##_statFUN_

#include "stdio.h"
#include "stdlib.h"

                        /*States connections (linkers between Blocks) */

void highPressureDetected();                /* link between mainAlgorith and AlarmMonitor */

void StartAlarm();                          /* link between AlarmMonitor Alarm_Led */

void StopAlarm();                           /* link between Alarm_Led AlarmMonitor */

void setPressure(int pVal);                 /* link between HighPressure_Detection and mainAlgorithm */


#endif /* STATE_H_ */

```

2. Main.c


*D:\Embedded Diploma\Units\Unit_5\First Project\Code\main.c - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
main.c alarmLed.c linker_script.ld mAlgo.c mAlgo.h startup.c alarmLed.h alarmMonitor.c alarmMonitor.h driver.c driver.h highPressure.c highPressure
C source file length : 748 lines : 46 Ln : 31 Col : 26 Pos : 586 Windows (CR LF) UTF-8 INS

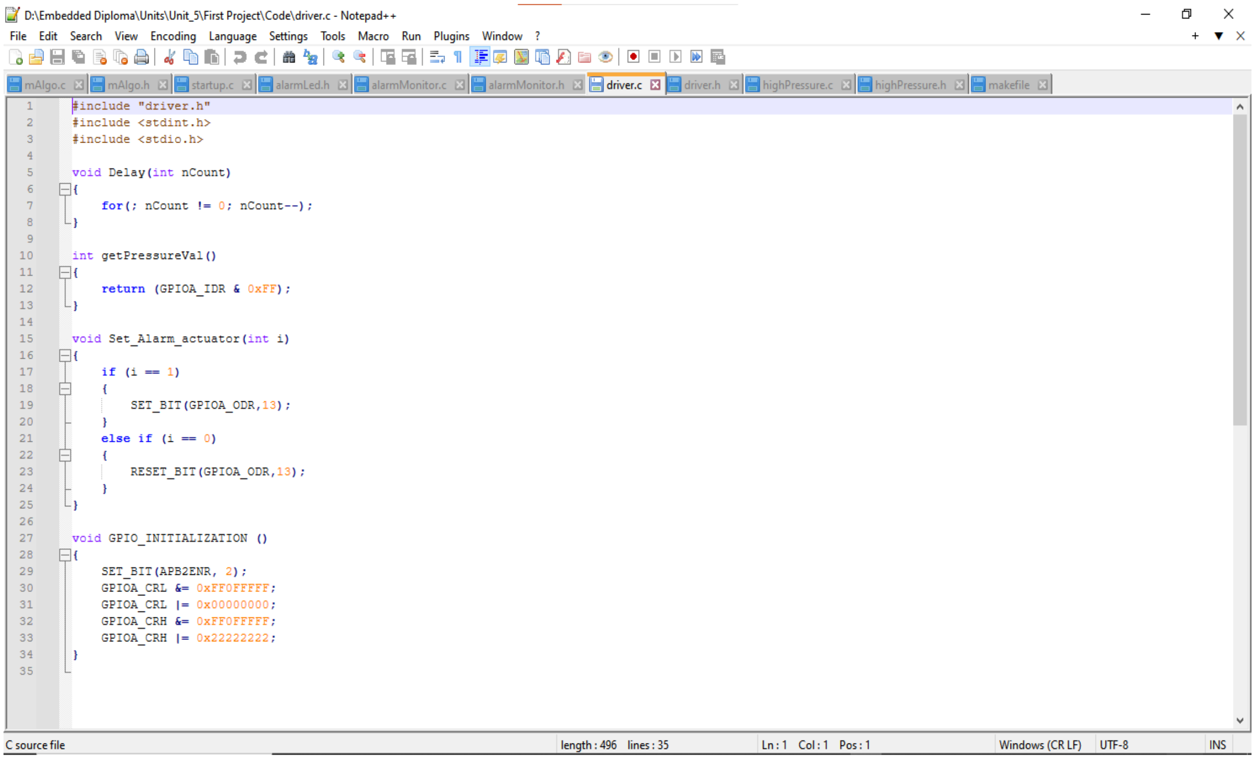
```
/*
 * main.c
 *
 *  Created on: 11 Mar 2024
 *      Author: Osama
 */
#include <stdint.h>
#include <stdio.h>

#include "driver.h"
#include "highPressure.h"
#include "mAlgo.h"
#include "alarmMonitor.h"
#include "alarmLed.h"

void (*p_state)() = STATE(ps_init);
void (*led_state)() = STATE(led_init);
void (*AMontor_state)() = STATE(Alarm_off);
void (*mAlgo_state)() = STATE(highPressure_state);

void setUp();
int main ()
{
    //setUp();
    while (1)
    {
        //Implement your Design
        p_state       ();
        mAlgo_state   ();
        AMontor_state ();
        led_state     ();
    }
}

/*
void setUp()
{

    STATE(ps_init)();
    STATE(led_init)();

    mAlgo_state = STATE(highPressure_state);
    AMontor_state = STATE(Alarm_off);

}
*/
```

3. Driver.h



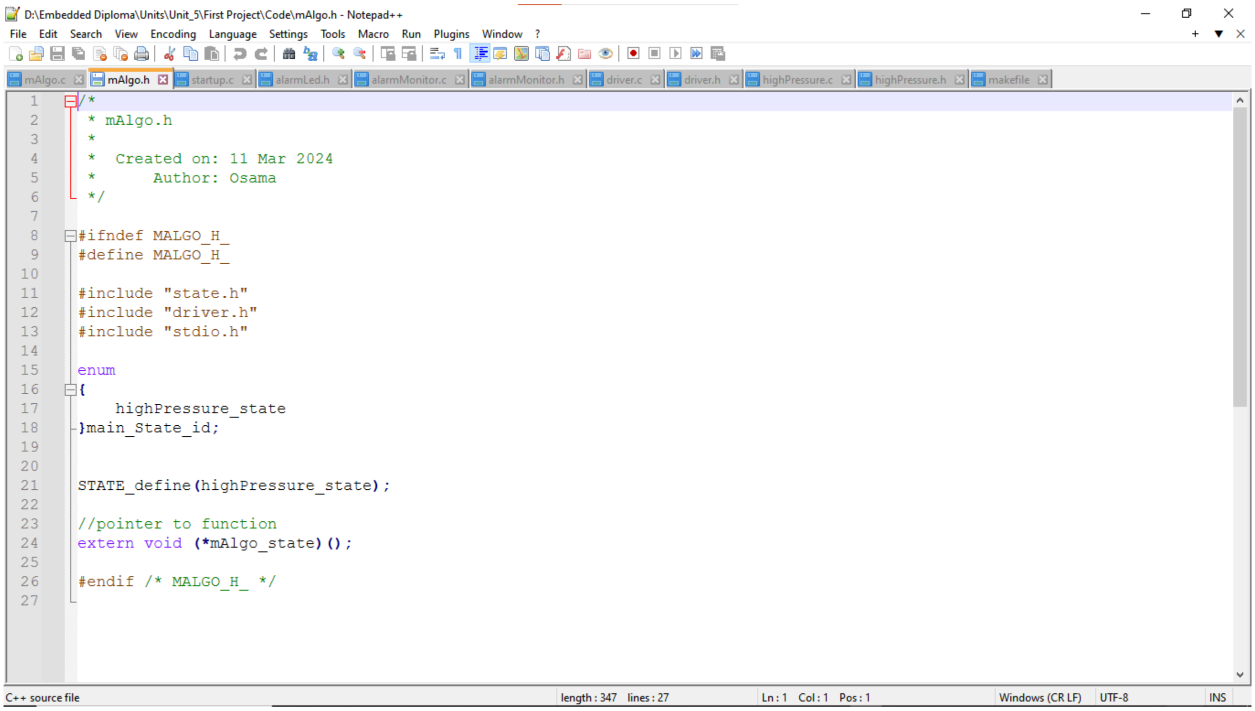
```c


#ifndef DRIVER_H_
#define DRIVER_H_

#include <stdint.h>
#include <stdio.h>

#define SET_BIT(ADDRESS,BIT)   ADDRESS |=  (1<<BIT)
#define RESET_BIT(ADDRESS,BIT) ADDRESS &= ~(1<<BIT)
#define TOGGLE_BIT(ADDRESS,BIT)  ADDRESS ^=  (1<<BIT)
#define READ_BIT(ADDRESS,BIT) ((ADDRESS) &   (1<<(BIT)))


#define GPIO_PORTA 0x40010800
#define BASE_RCC   0x40021000

#define APB2ENR   *(volatile uint32_t *)(BASE_RCC + 0x18)

#define GPIOA_CRL *(volatile uint32_t *)(GPIO_PORTA + 0x00)
#define GPIOA_CRH *(volatile uint32_t *)(GPIO_PORTA + 0X04)
#define GPIOA_IDR *(volatile uint32_t *)(GPIO_PORTA + 0x08)
#define GPIOA_ODR *(volatile uint32_t *)(GPIO_PORTA + 0x0C)


void Delay(int nCount);
int getPressureVal();
void Set_Alarm_actuator(int i);
void GPIO_INITIALIZATION ();

#endif /* DRIVER_H_ */

```

4. Driver.c



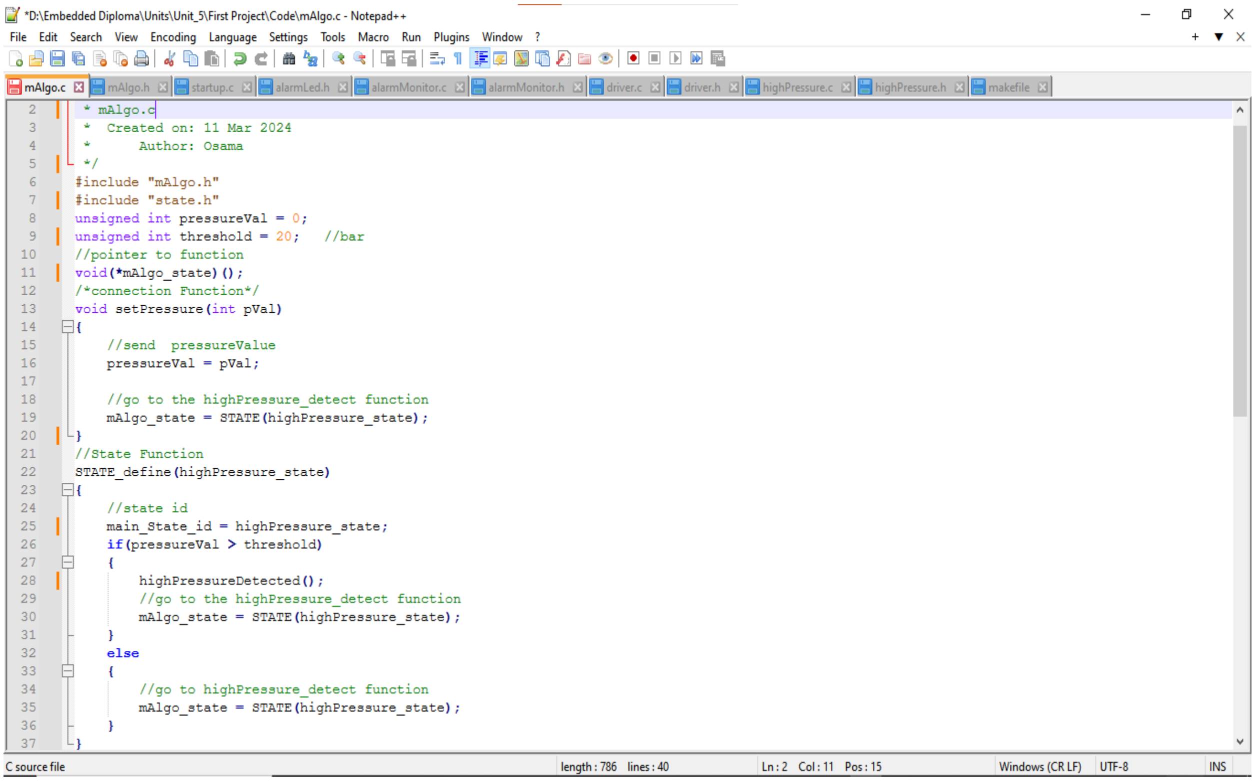
```c
#include "driver.h"
#include <stdint.h>
#include <stdio.h>

void Delay(int nCount)
{
    for(; nCount != 0; nCount--);
}

int getPressureVal()
{
    return (GPIOA_IDR & 0xFF);
}

void Set_Alarm_actuator(int i)
{
    if (i == 1)
    {
        SET_BIT(GPIOA_ODR,13);
    }
    else if (i == 0)
    {
        RESET_BIT(GPIOA_ODR,13);
    }
}

void GPIO_INITIALIZATION ()
{
    SET_BIT(APB2ENR, 2);
    GPIOA_CRL &= 0xFF0FFFFF;
    GPIOA_CRL |= 0x00000000;
    GPIOA_CRH &= 0xFF0FFFFF;
    GPIOA_CRH |= 0x22222222;
}

```

5. mainAlgorith.h



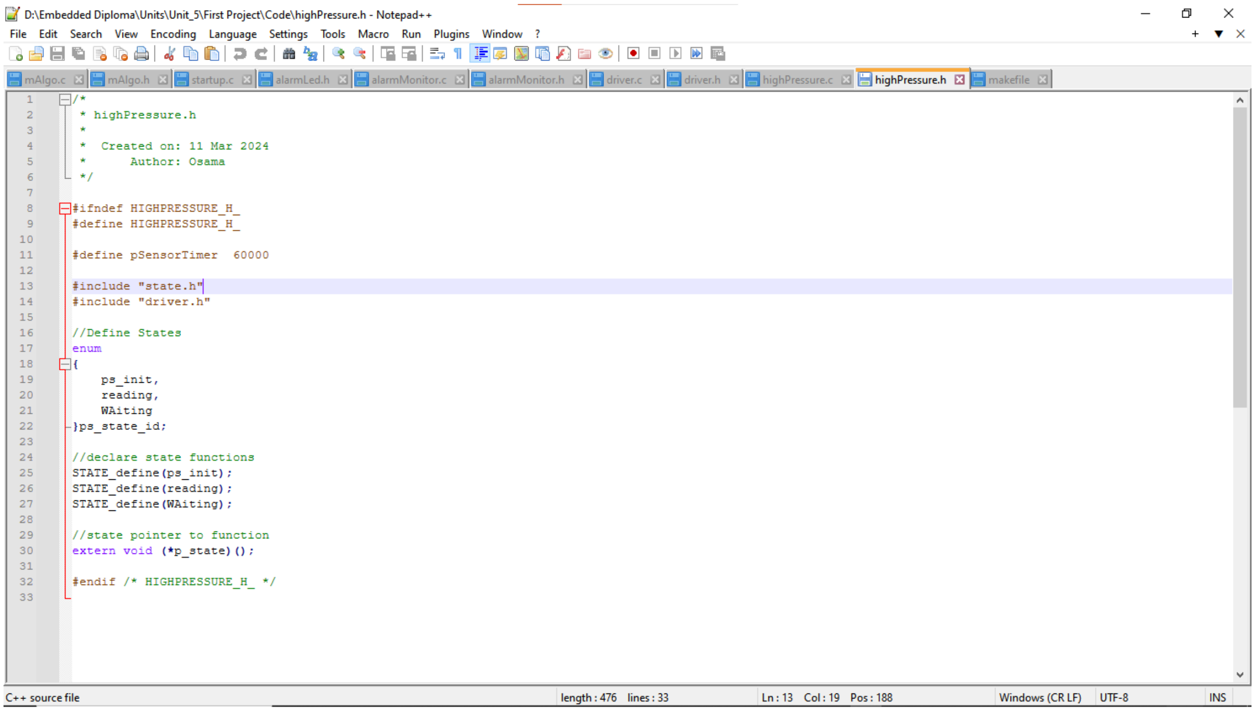
```c
/*
 * mAlgo.h
 *
 *  Created on: 11 Mar 2024
 *      Author: Osama
 */

#ifndef MALGO_H_
#define MALGO_H_

#include "state.h"
#include "driver.h"
#include "stdio.h"

enum
{
    highPressure_state
}main_State_id;


STATE_define(highPressure_state);

//pointer to function
extern void (*mAlgo_state)();

#endif /* MALGO_H_ */

```

6. mainAlgoritm.c



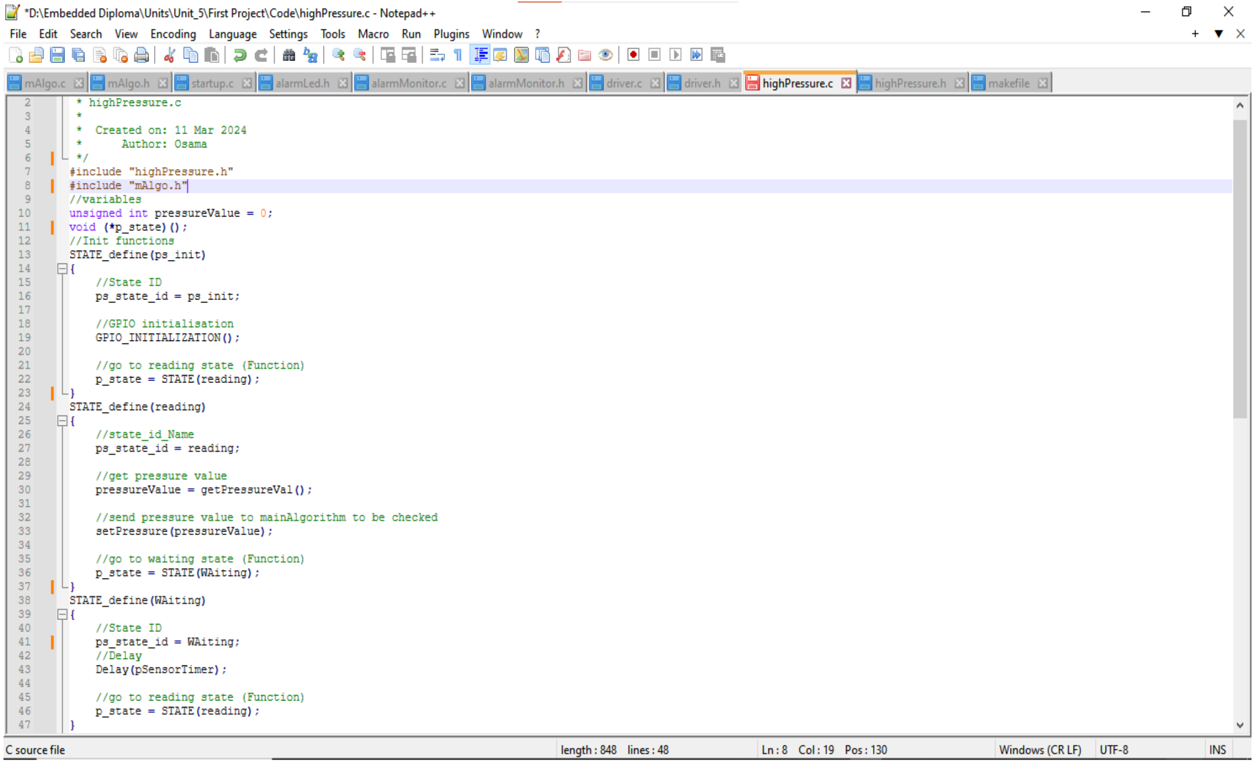
```c
 * mAlgo.c
 *  Created on: 11 Mar 2024
 *      Author: Osama
 */
#include "mAlgo.h"
#include "state.h"
unsigned int pressureVal = 0;
unsigned int threshold = 20;   //bar
//pointer to function
void(*mAlgo_state)();
/*connection Function*/
void setPressure(int pVal)
{
    //send  pressureValue
    pressureVal = pVal;

    //go to the highPressure_detect function
    mAlgo_state = STATE(highPressure_state);
}
//State Function
STATE_define(highPressure_state)
{
    //state id
    main_State_id = highPressure_state;
    if(pressureVal > threshold)
    {
        highPressureDetected();
        //go to the highPressure_detect function
        mAlgo_state = STATE(highPressure_state);
    }
    else
    {
        //go to highPressure_detect function
        mAlgo_state = STATE(highPressure_state);
    }
}
```

7. highpressre_detction.h


D:\Embedded Diploma\Units\Unit_5\First Project\Code\highPressure.h - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
mAlgo.c mAlgo.h startup.c alarmLed.h alarmMonitor.c alarmMonitor.h driver.c driver.h highPressure.c highPressure.h makefile
C++ source file
length : 476 lines : 33
Ln : 13 Col : 19 Pos : 188
Windows (CR LF) UTF-8 INS

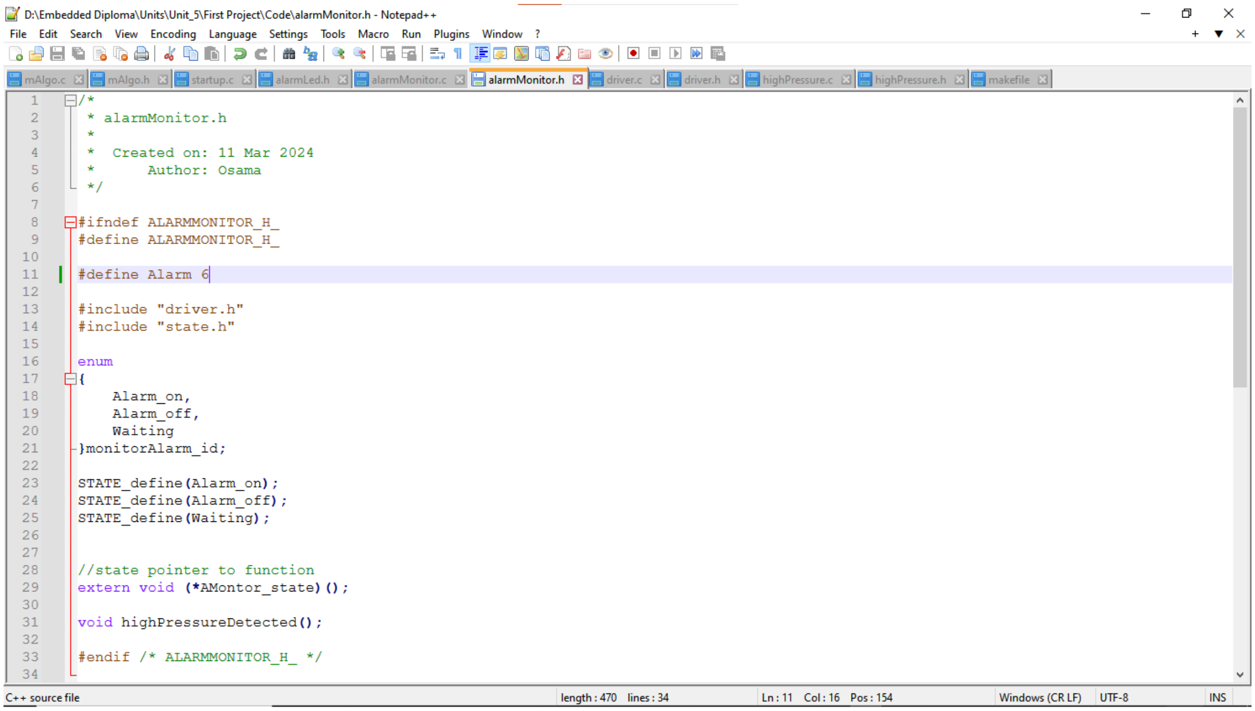
```c
/*
 * highPressure.h
 *
 *  Created on: 11 Mar 2024
 *      Author: Osama
 */

#ifndef HIGHPRESSURE_H_
#define HIGHPRESSURE_H_

#define pSensorTimer  60000

#include "state.h"
#include "driver.h"

//Define States
enum
{
    ps_init,
    reading,
    WAiting
}ps_state_id;

//declare state functions
STATE_define(ps_init);
STATE_define(reading);
STATE_define(WAiting);

//state pointer to function
extern void (*p_state)();

#endif /* HIGHPRESSURE_H_ */
```

8. highpressre_detction.c


*D:\Embedded Diploma\Units\Unit_5\First Project\Code\highPressure.c - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
mAlgo.c mAlgo.h startup.c alarmLed.h alarmMonitor.c alarmMonitor.h driver.c driver.h highPressure.c highPressure.h makefile
C source file
length : 848 lines : 48
Ln : 8 Col : 19 Pos : 130
Windows (CR LF) UTF-8 INS

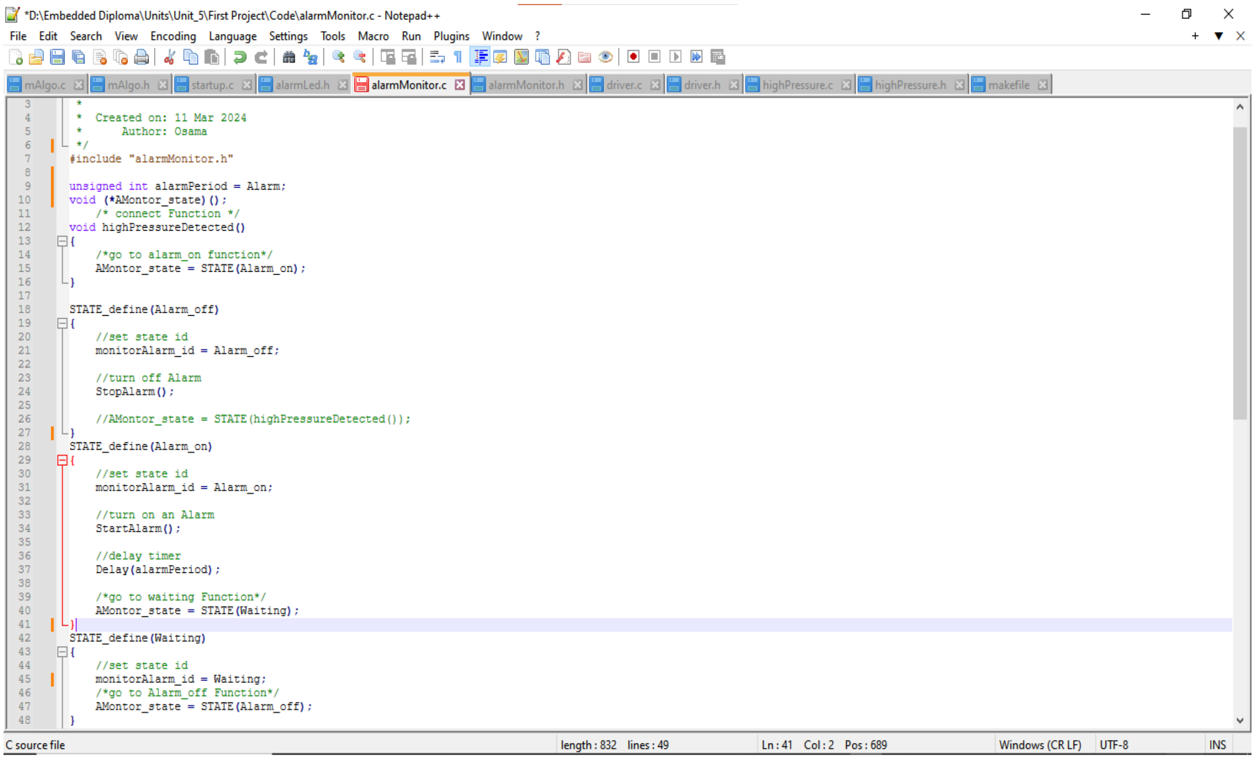
```c
 * highPressure.c
 *
 *  Created on: 11 Mar 2024
 *      Author: Osama
 */
#include "highPressure.h"
#include "mAlgo.h"
//variables
unsigned int pressureValue = 0;
void (*p_state)();
//Init functions
STATE_define(ps_init)
{
    //State ID
    ps_state_id = ps_init;

    //GPIO initialisation
    GPIO_INITIALIZATION();

    //go to reading state (Function)
    p_state = STATE(reading);
}
STATE_define(reading)
{
    //state_id_Name
    ps_state_id = reading;

    //get pressure value
    pressureValue = getPressureVal();

    //send pressure value to mainAlgorithm to be checked
    setPressure(pressureValue);

    //go to waiting state (Function)
    p_state = STATE(WAiting);
}
STATE_define(WAiting)
{
    //State ID
    ps_state_id = WAiting;
    //Delay
    Delay(pSensorTimer);

    //go to reading state (Function)
    p_state = STATE(reading);
}
```

9. MonitorAlarm.h


D:\Embedded Diploma\Units\Unit_5\First Project\Code\alarmMonitor.h - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
mAlgo.c mAlgo.h startup.c alarmLed.h alarmMonitor.c alarmMonitor.h driver.c driver.h highPressure.c highPressure.h makefile
C++ source file
length : 470 lines : 34
Ln : 11 Col : 16 Pos : 154
Windows (CR LF) UTF-8 INS

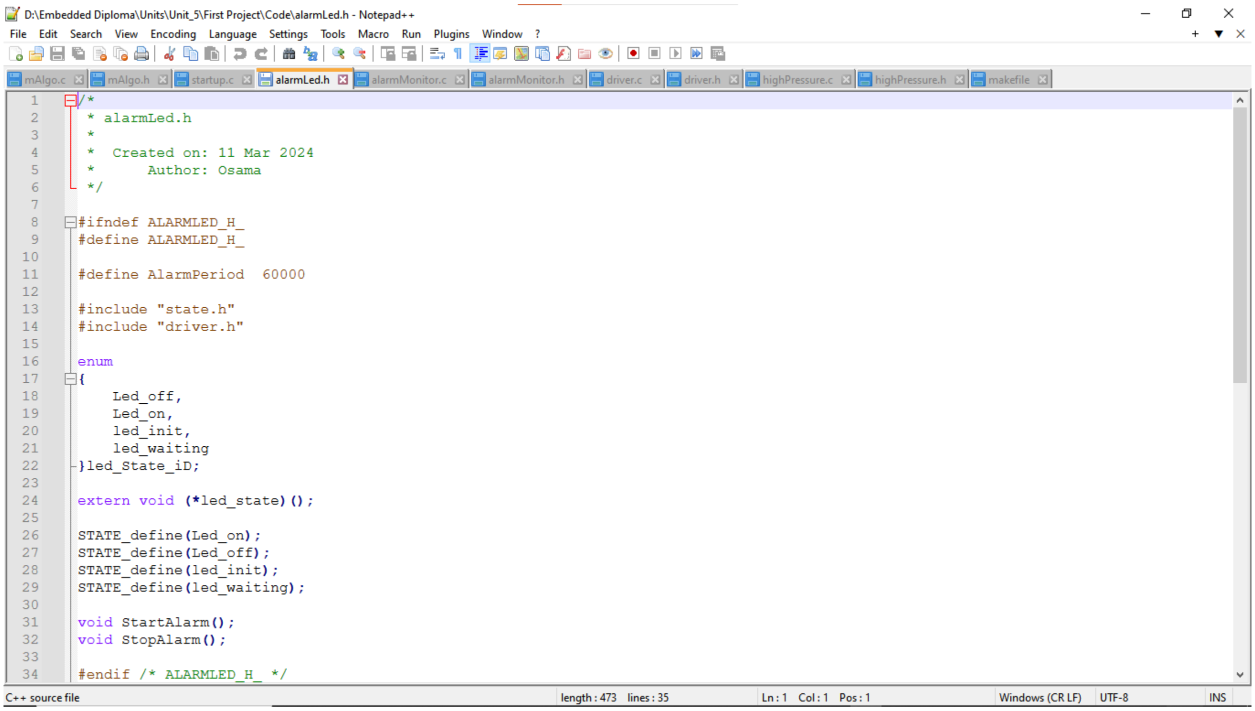
```c
/*
 * alarmMonitor.h
 *
 *  Created on: 11 Mar 2024
 *      Author: Osama
 */

#ifndef ALARMMONITOR_H_
#define ALARMMONITOR_H_

#define Alarm 6

#include "driver.h"
#include "state.h"

enum
{
    Alarm_on,
    Alarm_off,
    Waiting
}monitorAlarm_id;

STATE_define(Alarm_on);
STATE_define(Alarm_off);
STATE_define(Waiting);


//state pointer to function
extern void (*AMontor_state)();

void highPressureDetected();

#endif /* ALARMMONITOR_H_ */

```

10. MonitorAlarm.c


*D:\Embedded Diploma\Units\Unit_5\First Project\Code\alarmMonitor.c - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
mAlgo.c mAlgo.h startup.c alarmLed.h alarmMonitor.c alarmMonitor.h driver.c driver.h highPressure.c highPressure.h makefile
C source file
length : 832 lines : 49
Ln : 41 Col : 2 Pos : 689
Windows (CR LF) UTF-8 INS

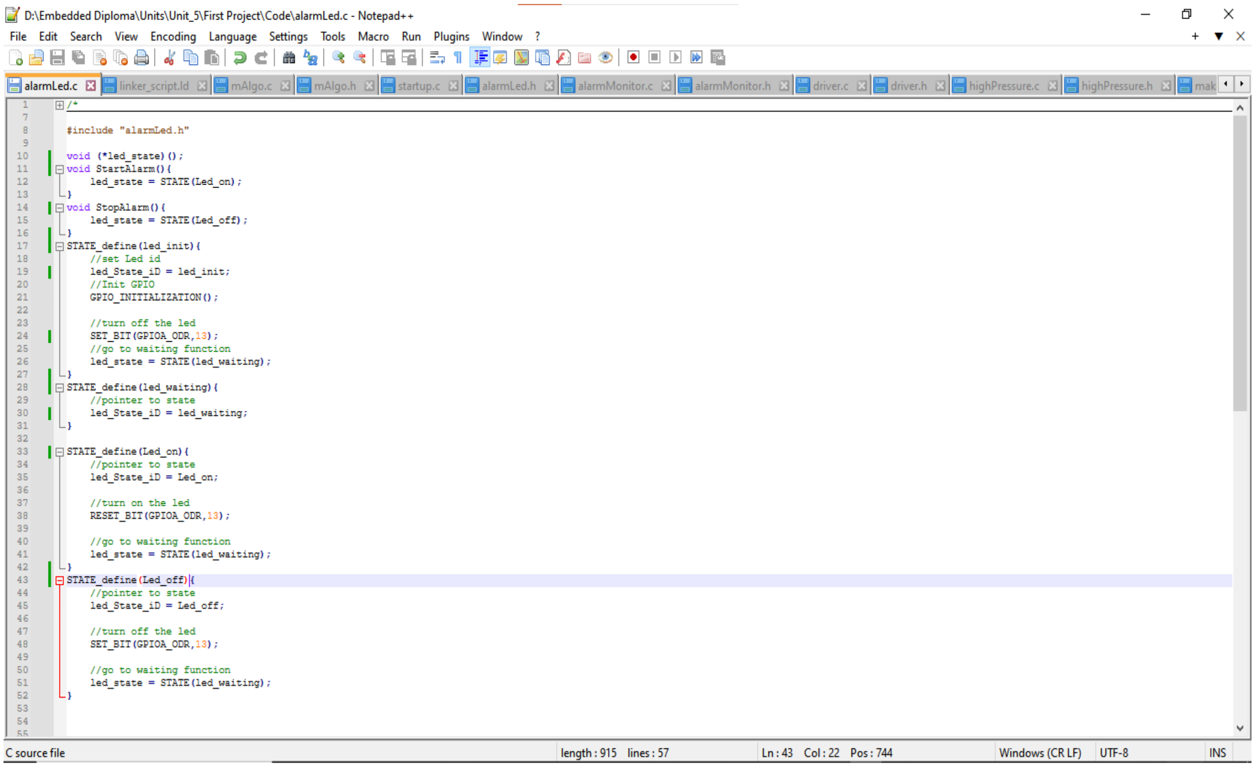
```c
 *
 *  Created on: 11 Mar 2024
 *      Author: Osama
 */
#include "alarmMonitor.h"

unsigned int alarmPeriod = Alarm;
void (*AMontor_state)();
    /* connect Function */
void highPressureDetected()
{
    /*go to alarm_on function*/
    AMontor_state = STATE(Alarm_on);
}

STATE_define(Alarm_off)
{
    //set state id
    monitorAlarm_id = Alarm_off;

    //turn off Alarm
    StopAlarm();

    //AMontor_state = STATE(highPressureDetected());
}
STATE_define(Alarm_on)
{
    //set state id
    monitorAlarm_id = Alarm_on;

    //turn on an Alarm
    StartAlarm();

    //delay timer
    Delay(alarmPeriod);

    /*go to waiting Function*/
    AMontor_state = STATE(Waiting);
}
STATE_define(Waiting)
{
    //set state id
    monitorAlarm_id = Waiting;
    /*go to Alarm_off Function*/
    AMontor_state = STATE(Alarm_off);
}
```

11. AlarmLed.h


D:\Embedded Diploma\Units\Unit_5\First Project\Code\alarmLed.h - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
mAlgo.c mAlgo.h startup.c alarmLed.h alarmMonitor.c alarmMonitor.h driver.c driver.h highPressure.c highPressure.h makefile
C++ source file
length : 473 lines : 35
Ln : 1 Col : 1 Pos : 1
Windows (CR LF) UTF-8 INS

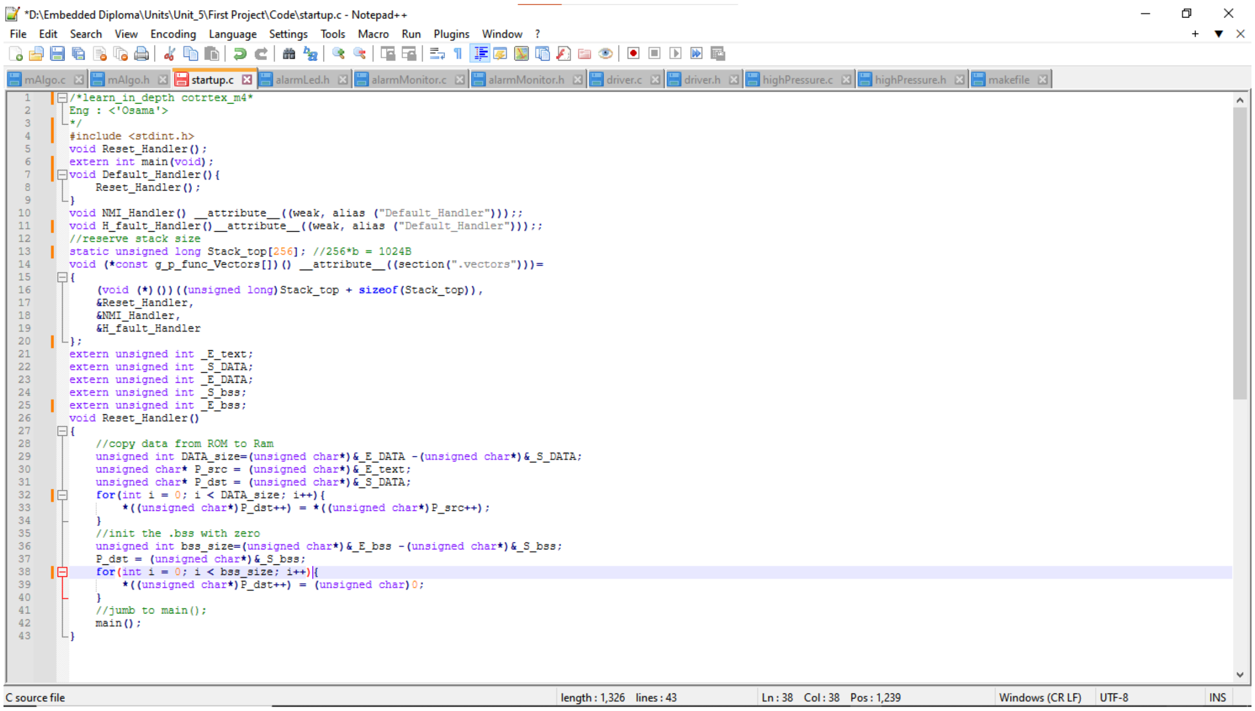
```c
/*
 * alarmLed.h
 *
 *  Created on: 11 Mar 2024
 *      Author: Osama
 */

#ifndef ALARMLED_H_
#define ALARMLED_H_

#define AlarmPeriod  60000

#include "state.h"
#include "driver.h"

enum
{
    Led_off,
    Led_on,
    led_init,
    led_waiting
}led_State_iD;

extern void (*led_state) ();

STATE_define(Led_on);
STATE_define(Led_off);
STATE_define(led_init);
STATE_define(led_waiting);

void StartAlarm();
void StopAlarm();

#endif /* ALARMLED_H_ */
```

12. AlarmLed.c


D:\Embedded Diploma\Units\Unit_5\First Project\Code\alarmLed.c - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
alarmLed.c linker_script.ld mAlgo.c mAlgo.h startup.c alarmLed.h alarmMonitor.c alarmMonitor.h driver.c driver.h highPressure.c highPressure.h mak
C source file
length : 915 lines : 57
Ln : 43 Col : 22 Pos : 744
Windows (CR LF) UTF-8 INS

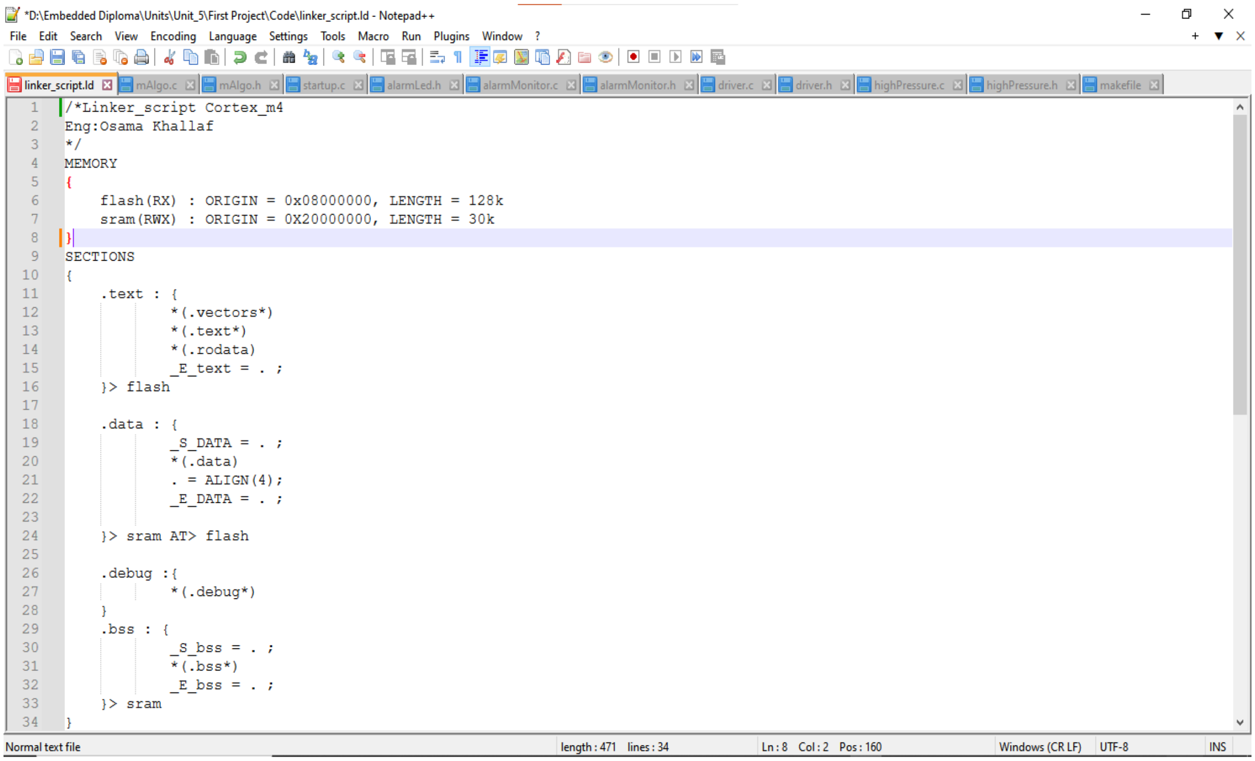
```c
/*

#include "alarmLed.h"

void (*led_state)();
void StartAlarm(){
    led_state = STATE(Led_on);
}
void StopAlarm(){
    led_state = STATE(Led_off);
}
STATE_define(led_init){
    //set Led id
    led_State_iD = led_init;
    //Init GPIO
    GPIO_INITIALIZATION();

    //turn off the led
    SET_BIT(GPIOA_ODR,13);
    //go to waiting function
    led_state = STATE(led_waiting);
}
STATE_define(led_waiting){
    //pointer to state
    led_State_iD = led_waiting;
}

STATE_define(Led_on){
    //pointer to state
    led_State_iD = Led_on;

    //turn on the led
    RESET_BIT(GPIOA_ODR,13);

    //go to waiting function
    led_state = STATE(led_waiting);
}
STATE_define(Led_off){
    //pointer to state
    led_State_iD = Led_off;

    //turn off the led
    SET_BIT(GPIOA_ODR,13);

    //go to waiting function
    led_state = STATE(led_waiting);
}
```

13. Startup.c


*D:\Embedded Diploma\Units\Unit_5\First Project\Code\startup.c - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
mAlgo.c mAlgo.h startup.c alarmLed.h alarmMonitor.c alarmMonitor.h driver.c driver.h highPressure.c highPressure.h makefile
C source file
length : 1,326 lines : 43
Ln : 38 Col : 38 Pos : 1,239
Windows (CR LF) UTF-8 INS

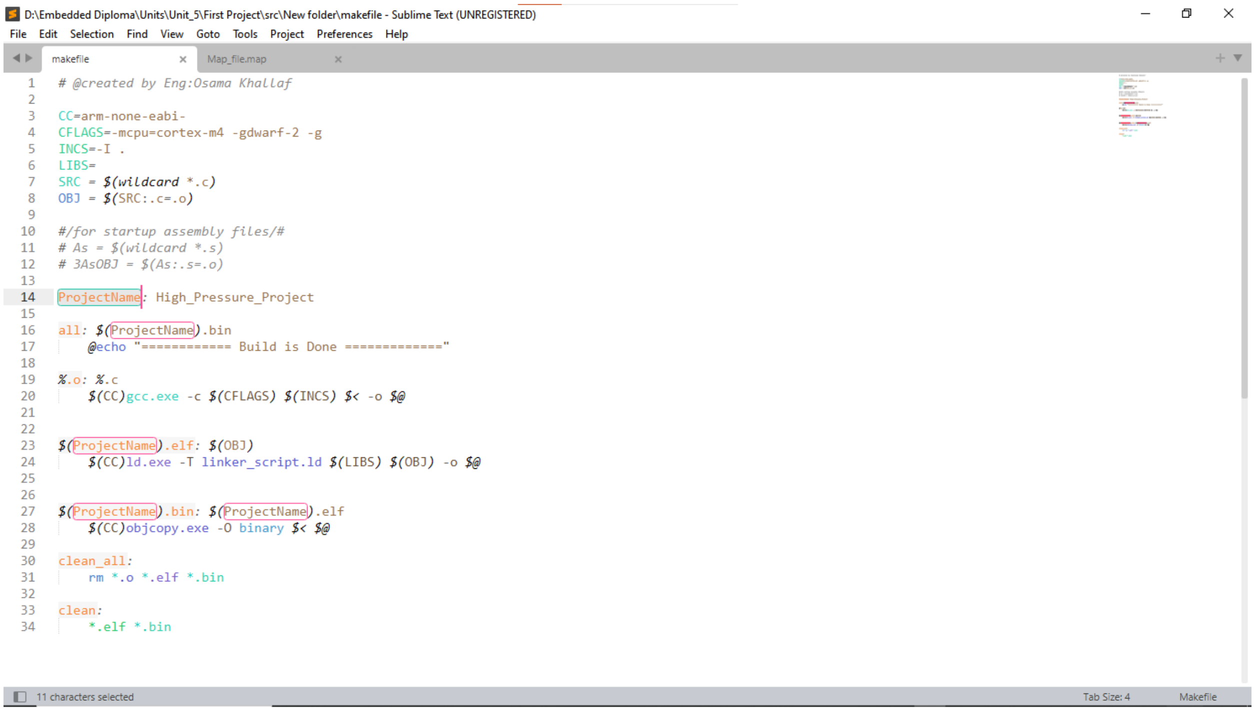
```c
/*learn_in_depth cotrtex_m4*
Eng : <'Osama'>
*/
#include <stdint.h>
void Reset_Handler();
extern int main(void);
void Default_Handler(){
    Reset_Handler();
}
void NMI_Handler() __attribute__((weak, alias ("Default_Handler")));;
void H_fault_Handler()__attribute__((weak, alias ("Default_Handler")));;
//reserve stack size
static unsigned long Stack_top[256]; //256*b = 1024B
void (*const g_p_func_Vectors[])() __attribute__((section(".vectors")))=
{
    (void (*)())((unsigned long)Stack_top + sizeof(Stack_top)),
    &Reset_Handler,
    &NMI_Handler,
    &H_fault_Handler
};
extern unsigned int _E_text;
extern unsigned int _S_DATA;
extern unsigned int _E_DATA;
extern unsigned int _S_bss;
extern unsigned int _E_bss;
void Reset_Handler()
{
    //copy data from ROM to Ram
    unsigned int DATA_size=(unsigned char*)&_E_DATA -(unsigned char*)&_S_DATA;
    unsigned char* P_src = (unsigned char*)&_E_text;
    unsigned char* P_dst = (unsigned char*)&_S_DATA;
    for(int i = 0; i < DATA_size; i++){
        *((unsigned char*)P_dst++) = *((unsigned char*)P_src++);
    }
    //init the .bss with zero
    unsigned int bss_size=(unsigned char*)&_E_bss -(unsigned char*)&_S_bss;
    P_dst = (unsigned char*)&_S_bss;
    for(int i = 0; i < bss_size; i++){
        *((unsigned char*)P_dst++) = (unsigned char)0;
    }
    //jumb to main();
    main();
}
```

14. Linker_script.ld


*D:\Embedded Diploma\Units\Unit_5\First Project\Code\linker_script.ld - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
linker_script.ld mAlgo.c mAlgo.h startup.c alarmLed.h alarmMonitor.c alarmMonitor.h driver.c driver.h highPressure.c highPressure.h makefile
Normal text file
length : 471 lines : 34
Ln : 8 Col : 2 Pos : 160
Windows (CR LF) UTF-8 INS

```
/*Linker_script Cortex_m4
Eng:Osama Khallaf
*/
MEMORY
{
    flash(RX)  : ORIGIN = 0x08000000, LENGTH = 128k
    sram(RWX)  : ORIGIN = 0X20000000, LENGTH = 30k
}
SECTIONS
{
    .text : {
            *(.vectors*)
            *(.text*)
            *(.rodata)
            _E_text = . ;
    }> flash

    .data : {
            _S_DATA = . ;
            *(.data)
            . = ALIGN(4);
            _E_DATA = . ;

    }> sram AT> flash

    .debug :{
            *(.debug*)
    }
    .bss : {
            _S_bss = . ;
            *(.bss*)
            _E_bss = . ;
    }> sram
}
```

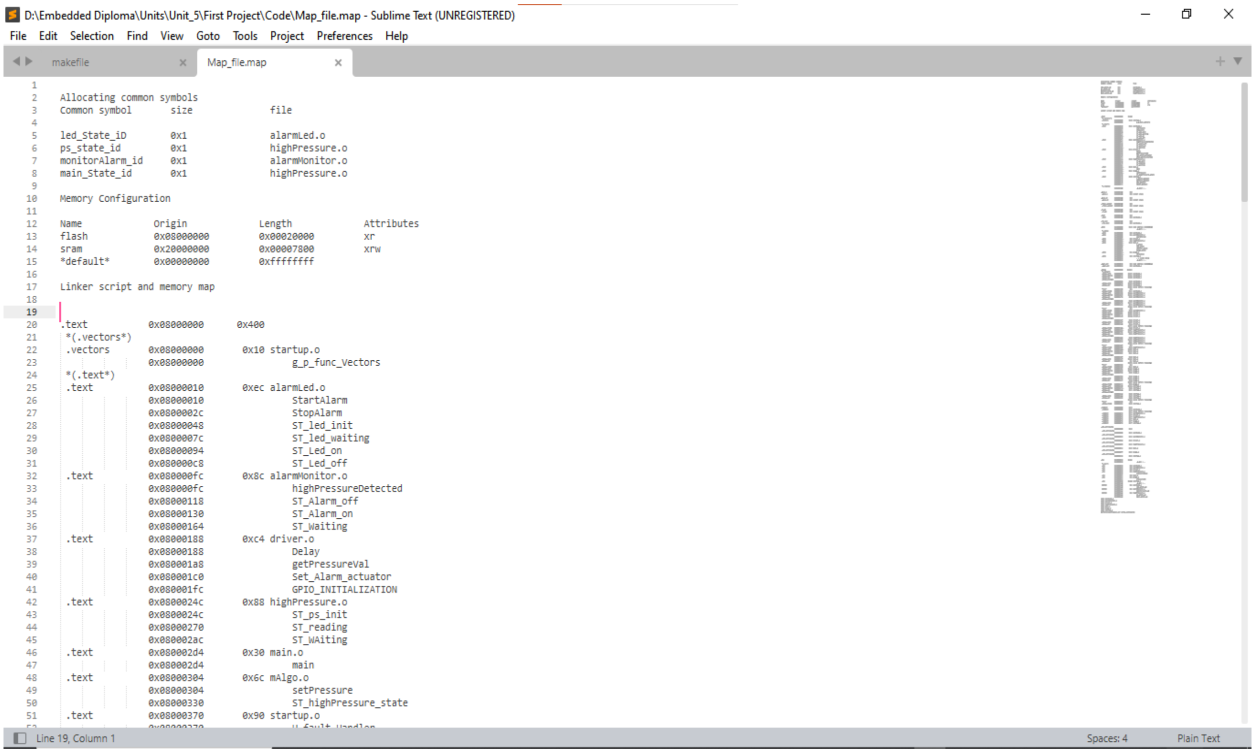
15. Makefile



D:\Embedded Diploma\Units\Unit_5\First Project\src\New folder\makefile - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
makefile
Map_file.map
11 characters selected
Tab Size: 4
Makefile

```makefile
# @created by Eng:Osama Khallaf

CC=arm-none-eabi-
CFLAGS=-mcpu=cortex-m4 -gdwarf-2 -g
INCS=-I .
LIBS=
SRC = $(wildcard *.c)
OBJ = $(SRC:.c=.o)

#/for startup assembly files/#
# As = $(wildcard *.s)
# 3AsOBJ = $(As:.s=.o)

ProjectName: High_Pressure_Project

all: $(ProjectName).bin
	@echo "============ Build is Done ============"

%.o: %.c
	$(CC)gcc.exe -c $(CFLAGS) $(INCS) $< -o $@


$(ProjectName).elf: $(OBJ)
	$(CC)ld.exe -T linker_script.ld $(LIBS) $(OBJ) -o $@


$(ProjectName).bin: $(ProjectName).elf
	$(CC)objcopy.exe -O binary $< $@

clean_all:
	rm *.o *.elf *.bin

clean:
	*.elf *.bin
```

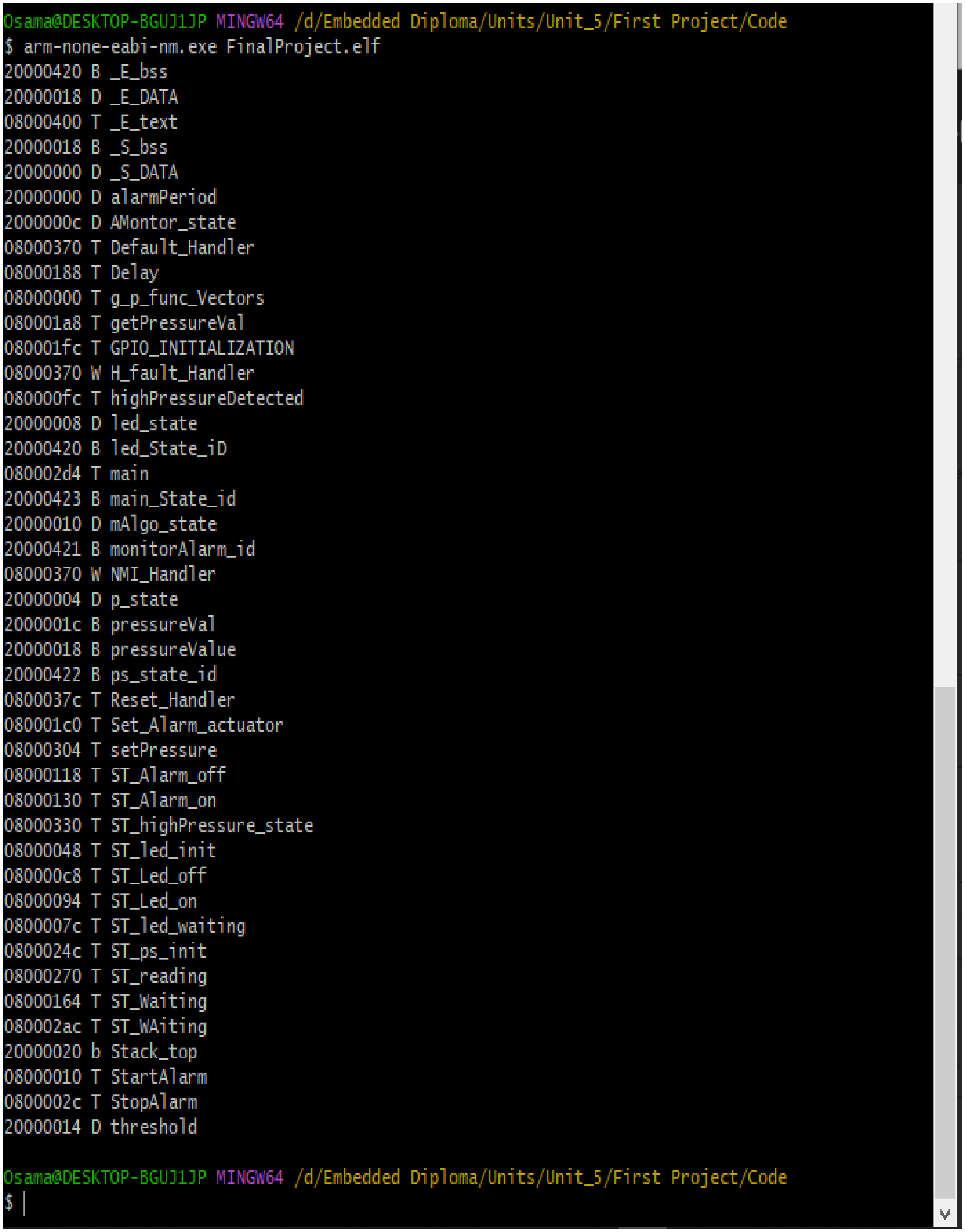
16. mapFile



D:\Embedded Diploma\Units\Unit_5\First Project\Code\Map_file.map - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
makefile
Map_file.map
Line 19, Column 1
Spaces: 4
Plain Text

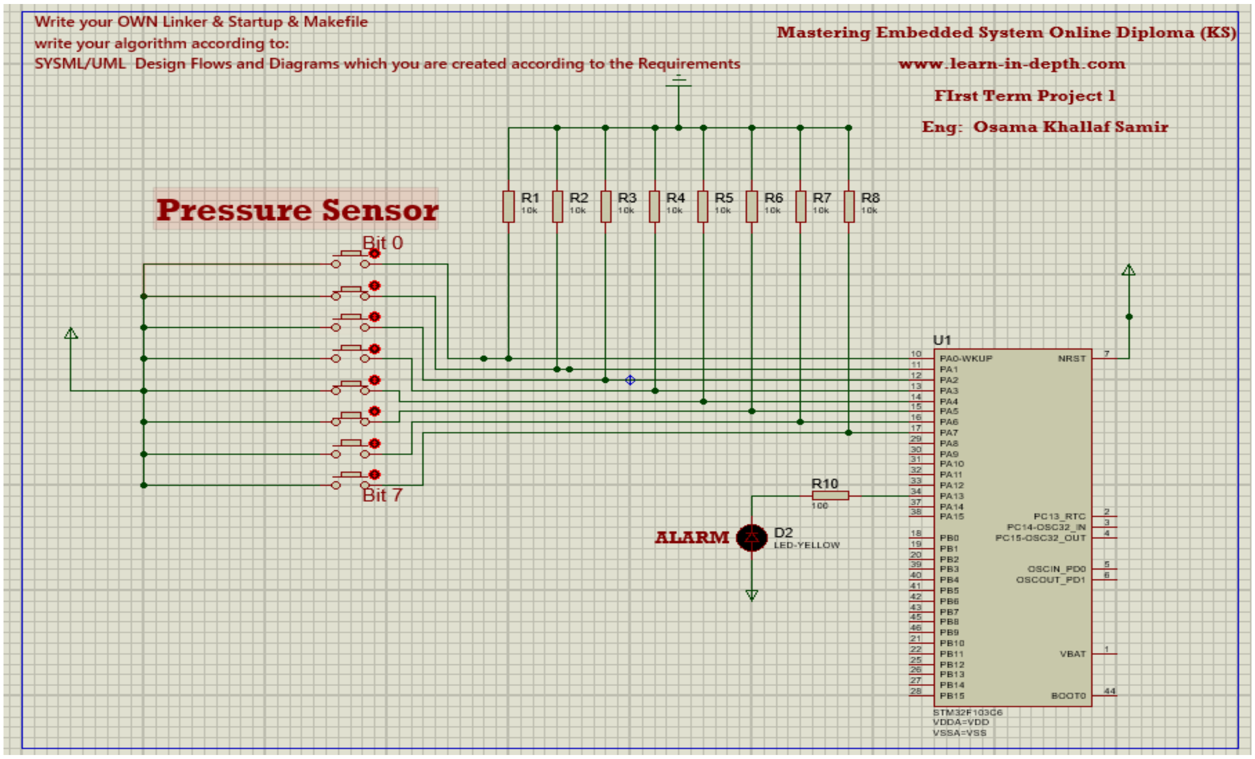
```

Allocating common symbols
Common symbol       size              file

led_State_iD        0x1               alarmLed.o
ps_state_id         0x1               highPressure.o
monitorAlarm_id     0x1               alarmMonitor.o
main_State_id       0x1               highPressure.o

Memory Configuration

Name             Origin             Length             Attributes
flash            0x08000000         0x00020000         xr
sram             0x20000000         0x00007800         xrw
*default*        0x00000000         0xffffffff

Linker script and memory map


.text           0x08000000      0x400
 *(.vectors*)
 .vectors       0x08000000       0x10 startup.o
                0x08000000                g_p_func_Vectors
 *(.text*)
 .text          0x08000010       0xec alarmLed.o
                0x08000010                StartAlarm
                0x0800002c                StopAlarm
                0x08000048                ST_led_init
                0x0800007c                ST_led_waiting
                0x08000094                ST_Led_on
                0x080000c8                ST_Led_off
 .text          0x080000fc       0x8c alarmMonitor.o
                0x080000fc                highPressureDetected
                0x08000118                ST_Alarm_off
                0x08000130                ST_Alarm_on
                0x08000164                ST_Waiting
 .text          0x08000188       0xc4 driver.o
                0x08000188                Delay
                0x080001a8                getPressureVal
                0x080001c0                Set_Alarm_actuator
                0x080001fc                GPIO_INITIALIZATION
 .text          0x0800024c       0x88 highPressure.o
                0x0800024c                ST_ps_init
                0x08000270                ST_reading
                0x080002ac                ST_WAiting
 .text          0x080002d4       0x30 main.o
                0x080002d4                main
 .text          0x08000304       0x6c mAlgo.o
                0x08000304                setPressure
                0x08000330                ST_highPressure_state
 .text          0x08000370       0x90 startup.o
                0x08000370                H_fault_Handler
```

17. symbolTable

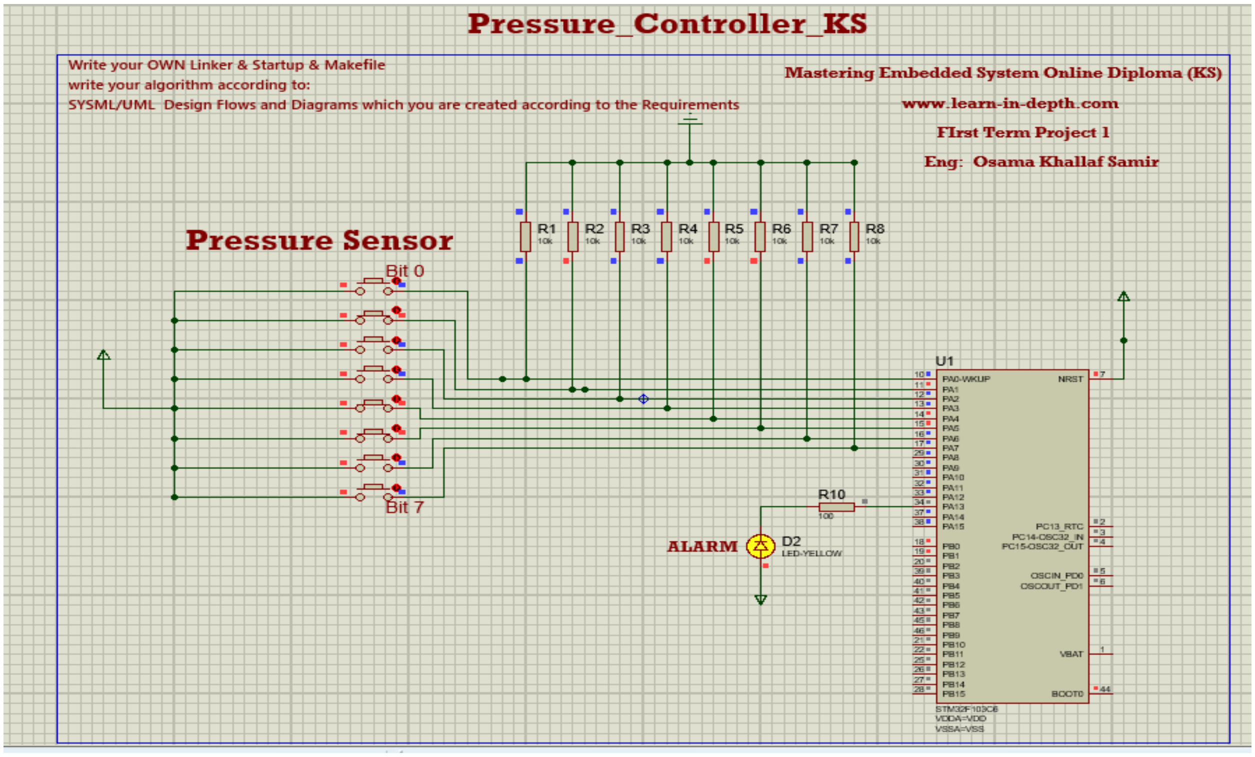


```
Osama@DESKTOP-BGUJ1JP MINGW64 /d/Embedded Diploma/Units/Unit_5/First Project/Code
$ arm-none-eabi-nm.exe FinalProject.elf
20000420 B _E_bss
20000018 D _E_DATA
08000400 T _E_text
20000018 B _S_bss
20000000 D _S_DATA
20000000 D alarmPeriod
2000000c D AMontor_state
08000370 T Default_Handler
08000188 T Delay
08000000 T g_p_func_Vectors
080001a8 T getPressureVal
080001fc T GPIO_INITIALIZATION
08000370 W H_fault_Handler
080000fc T highPressureDetected
20000008 D led_state
20000420 B led_State_iD
080002d4 T main
20000423 B main_State_id
20000010 D mAlgo_state
20000421 B monitorAlarm_id
08000370 W NMI_Handler
20000004 D p_state
2000001c B pressureVal
20000018 B pressureValue
20000422 B ps_state_id
0800037c T Reset_Handler
080001c0 T Set_Alarm_actuator
08000304 T setPressure
08000118 T ST_Alarm_off
08000130 T ST_Alarm_on
08000330 T ST_highPressure_state
08000048 T ST_led_init
080000c8 T ST_Led_off
08000094 T ST_Led_on
0800007c T ST_led_waiting
0800024c T ST_ps_init
08000270 T ST_reading
08000164 T ST_Waiting
080002ac T ST_WAiting
20000020 b Stack_top
08000010 T StartAlarm
0800002c T StopAlarm
20000014 D threshold

Osama@DESKTOP-BGUJ1JP MINGW64 /d/Embedded Diploma/Units/Unit_5/First Project/Code
$ 
```

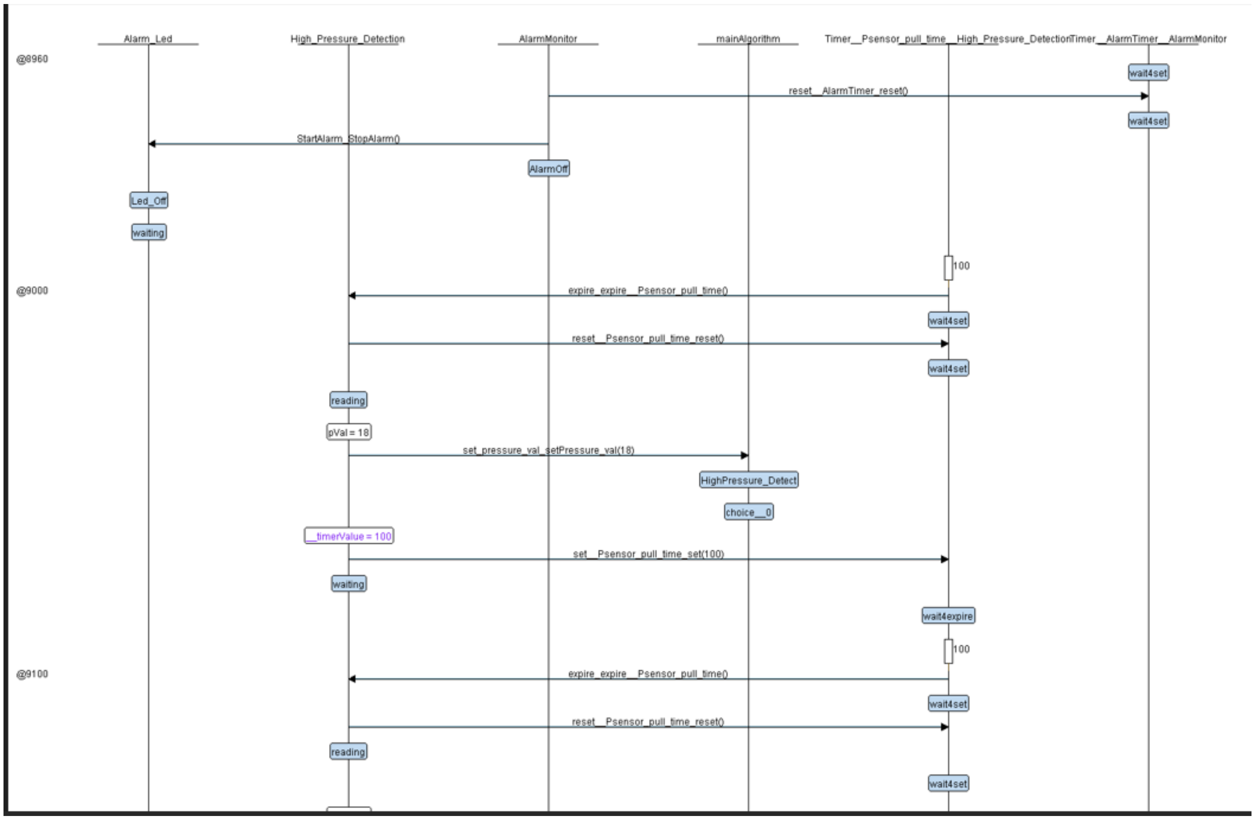
18. simulation before burn the bin file



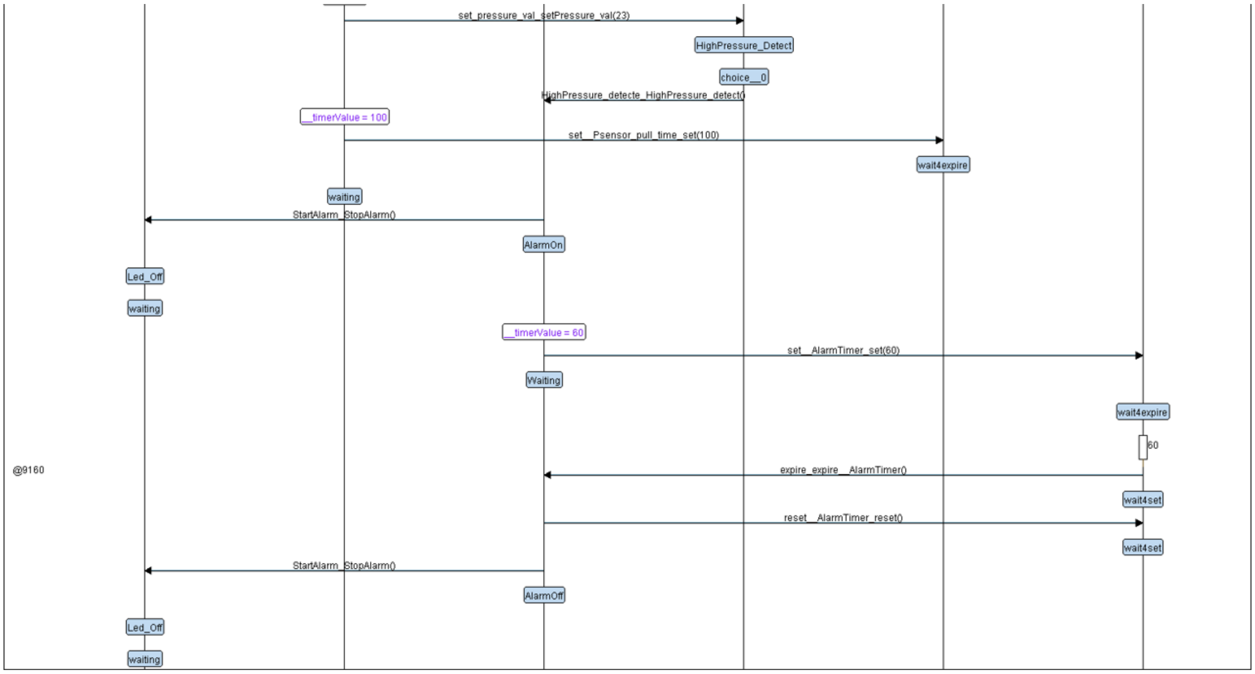
19. simulation after burn the bin file



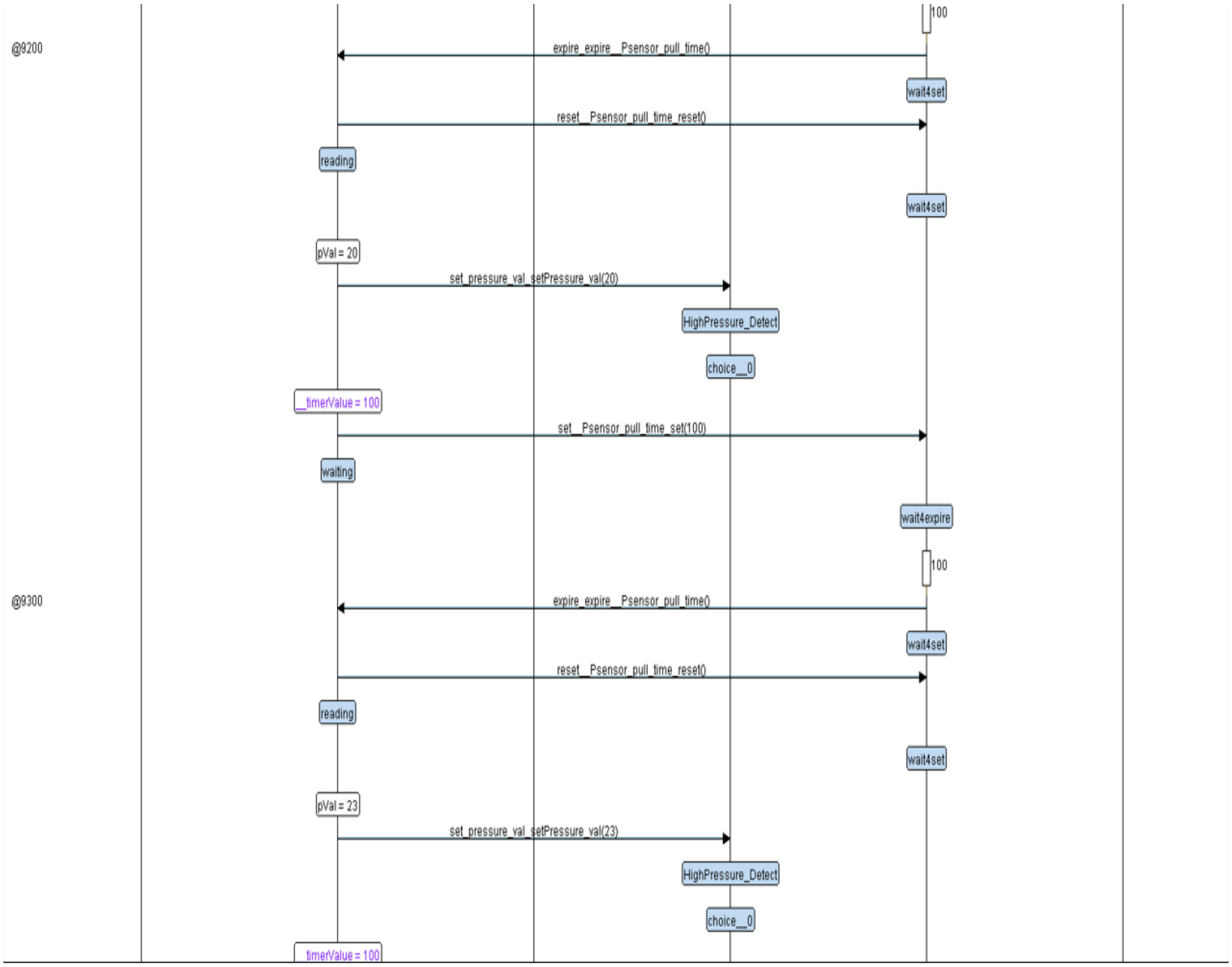
20. interactive simulation1


Alarm_Led
High_Pressure_Detection
AlarmMonitor
mainAlgorithm
Timer__Psensor_pull_time__High_Pressure_DetectionTimer__AlarmTimer__AlarmMonitor
@8960
wait4set
reset__AlarmTimer_reset()
wait4set
StartAlarm_StopAlarm()
AlarmOff
Led_Off
waiting
100
@9000
expire_expire__Psensor_pull_time()
wait4set
reset__Psensor_pull_time_reset()
wait4set
reading
pVal = 18
set_pressure_val_setPressure_val(18)
HighPressure_Detect
choice__0
__timerValue = 100
set__Psensor_pull_time_set(100)
waiting
wait4expire
100
@9100
expire_expire__Psensor_pull_time()
wait4set
reset__Psensor_pull_time_reset()
reading
wait4set

21. interactive simulation2



22. interactive simulation3


100
@9200
expire_expire__Psensor_pull_time()
wait4set
reset__Psensor_pull_time_reset()
reading
wait4set
pVal = 20
set_pressure_val_setPressure_val(20)
HighPressure_Detect
choice__0
__timerValue = 100
set__Psensor_pull_time_set(100)
waiting
wait4expire
100
@9300
expire_expire__Psensor_pull_time()
wait4set
reset__Psensor_pull_time_reset()
reading
wait4set
pVal = 23
set_pressure_val_setPressure_val(23)
HighPressure_Detect
choice__0
__timerValue = 100