Mastering Embedded System Online Diploma

www.learn-in-depth.com

First Term Final Project 1

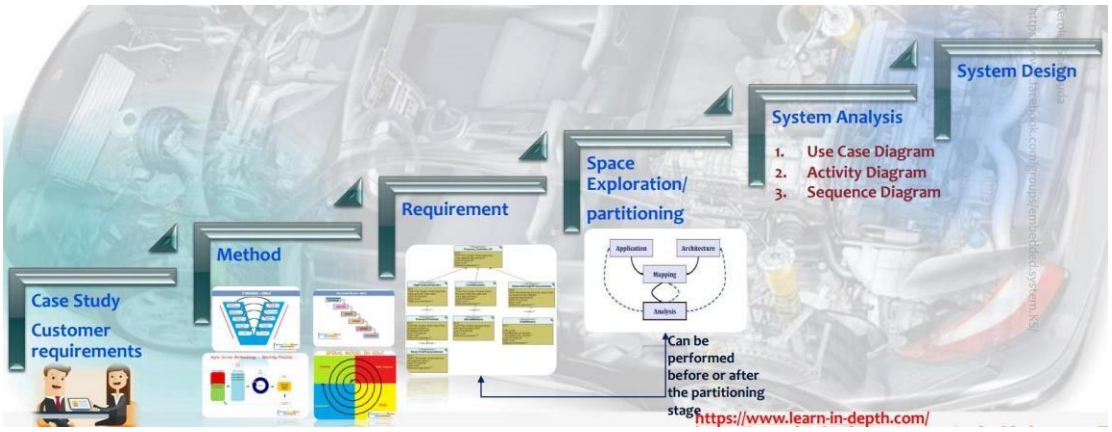
Pressure Control System

Eng. Ahmed Abdelmonem Mohamed

My profile:

https://www.learn-in-depth-store.com/certificate/ahmed15517563%40gmail.com

System Architecting/Design Sequence



1.Case study:

The pressure control system is designed to measure the pressure in the cabin, if it exceeds a certain threshold it triggers an alarm.

● Pressure control system aims to achieve:

- monitoring of pressure values

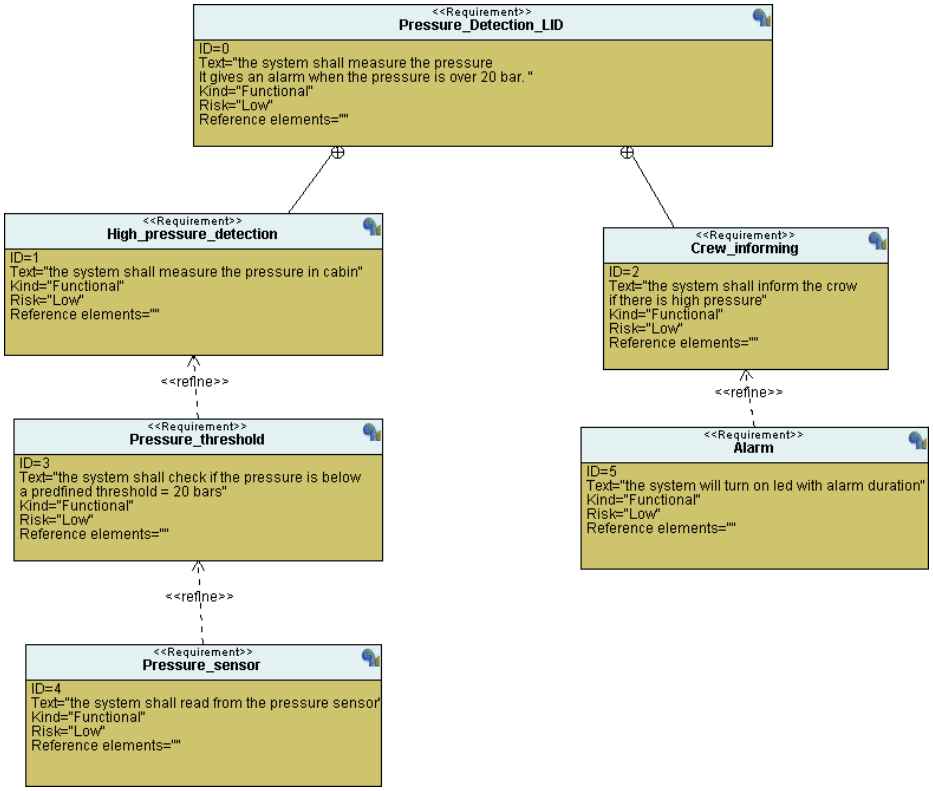
- Alarm activation when pressure exceeds 20 bars and Automatic alarm deactivation after 60 seconds if the pressure becomes below 20 bars

2.Method:

I used the waterfall model to this project.

3.Requirement:

The following diagram is the UML requirement diagram for this system.



4. Space exploration and hardware / software partitioning:

Hardware:

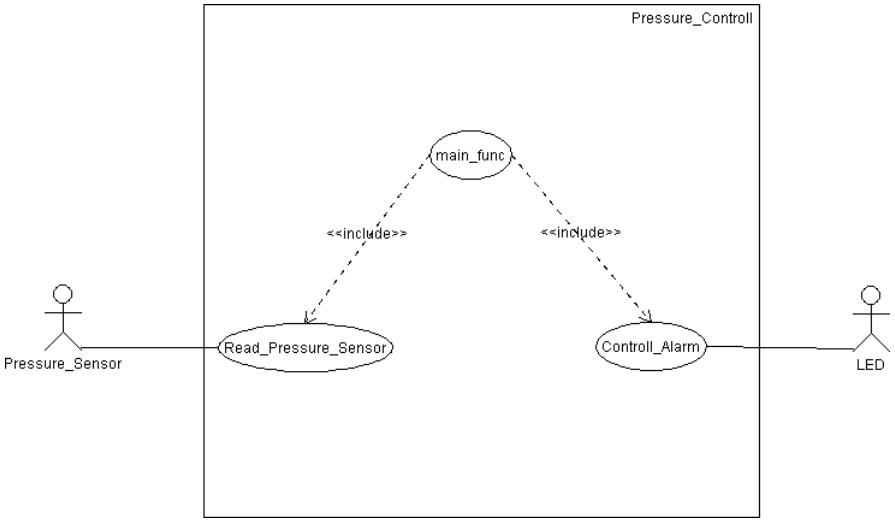
● Controller: STM32F103C6. ● Alarm: LED. ● Sensor: Pressure sensor

Software:

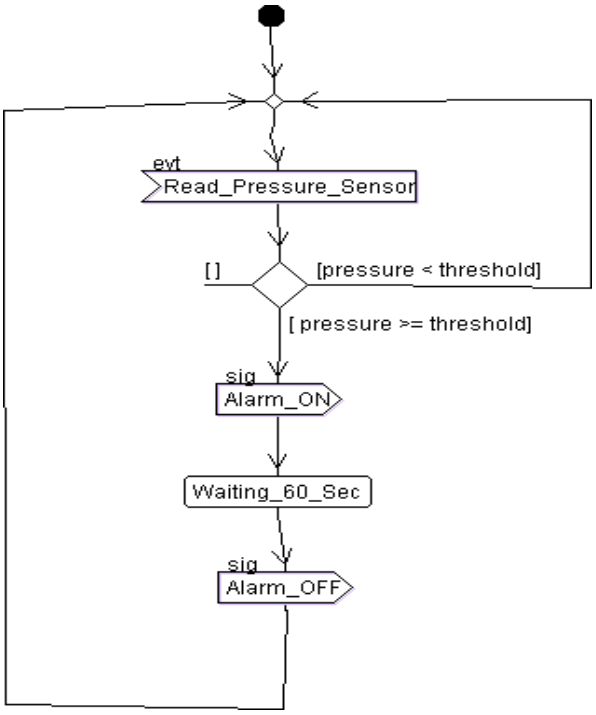
● Pressure sensor driver. ● Main function. ● Alarm actuator driver.

5.System analysis:

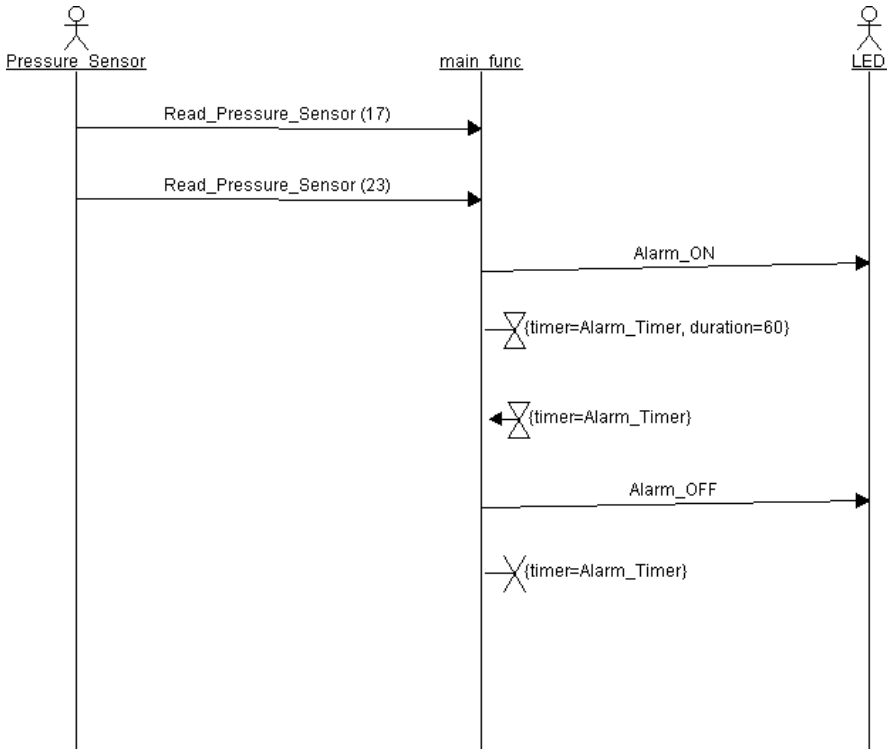
Use case diagram:



Activity Diagram:

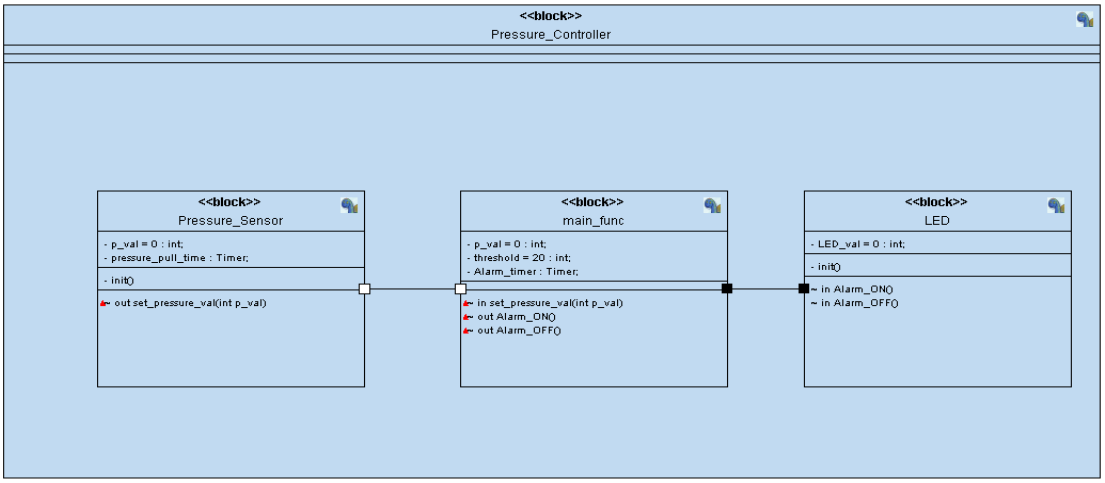


Sequence Diagram:

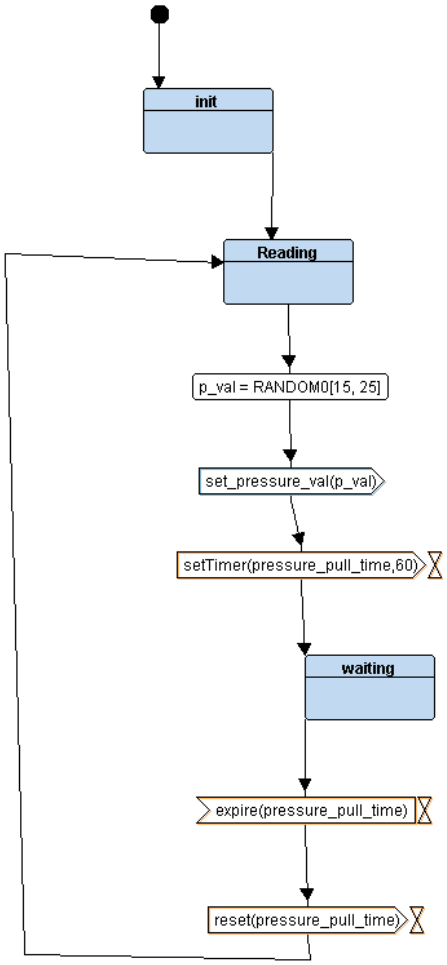


6.System design:

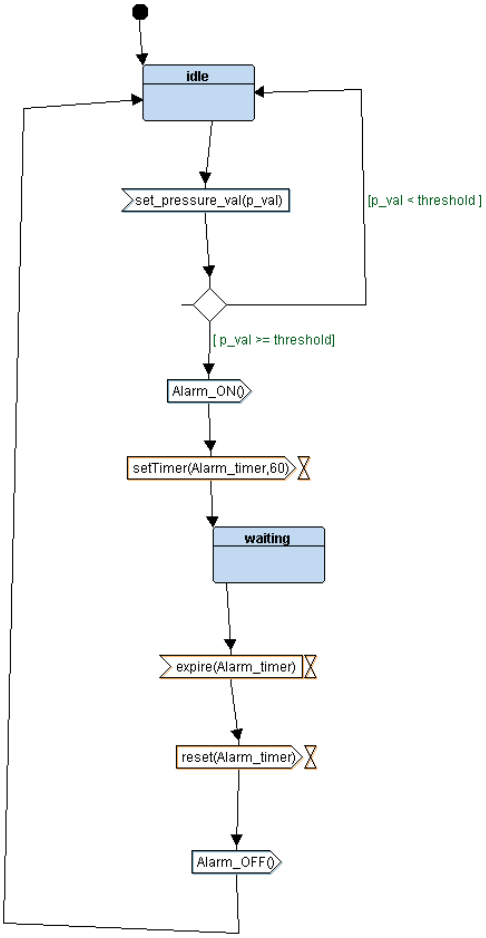
System block diagram:



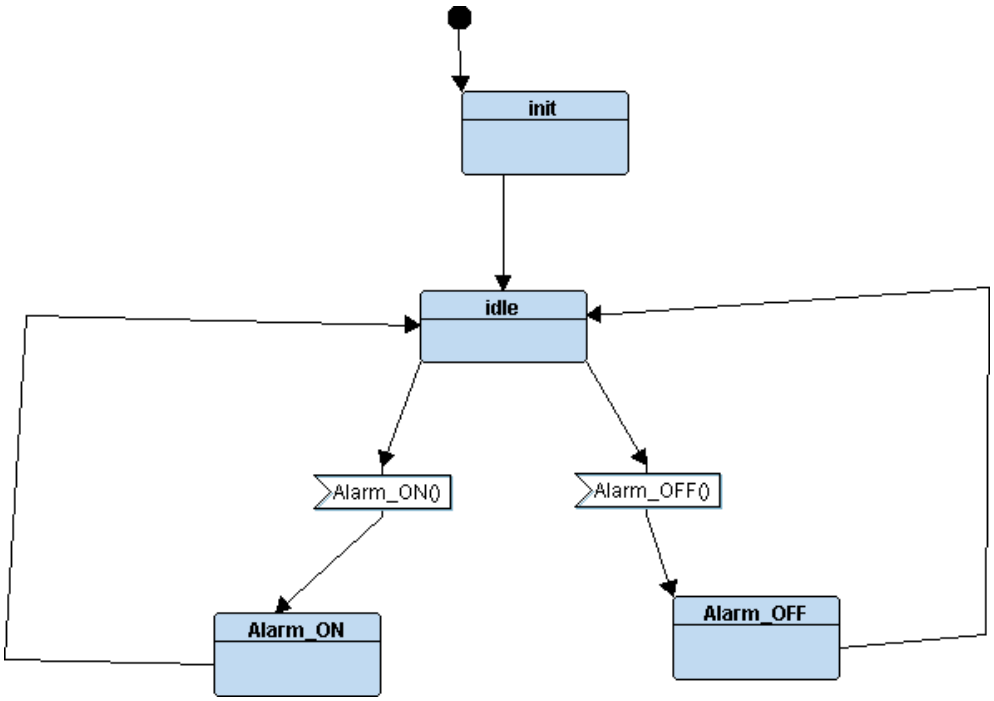
State Machine : pressure sensor



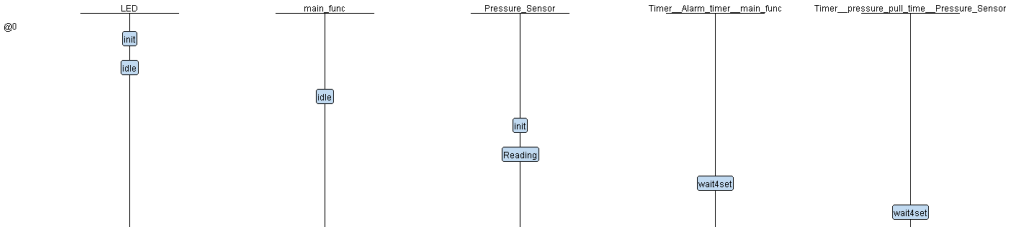
State Machine : main function



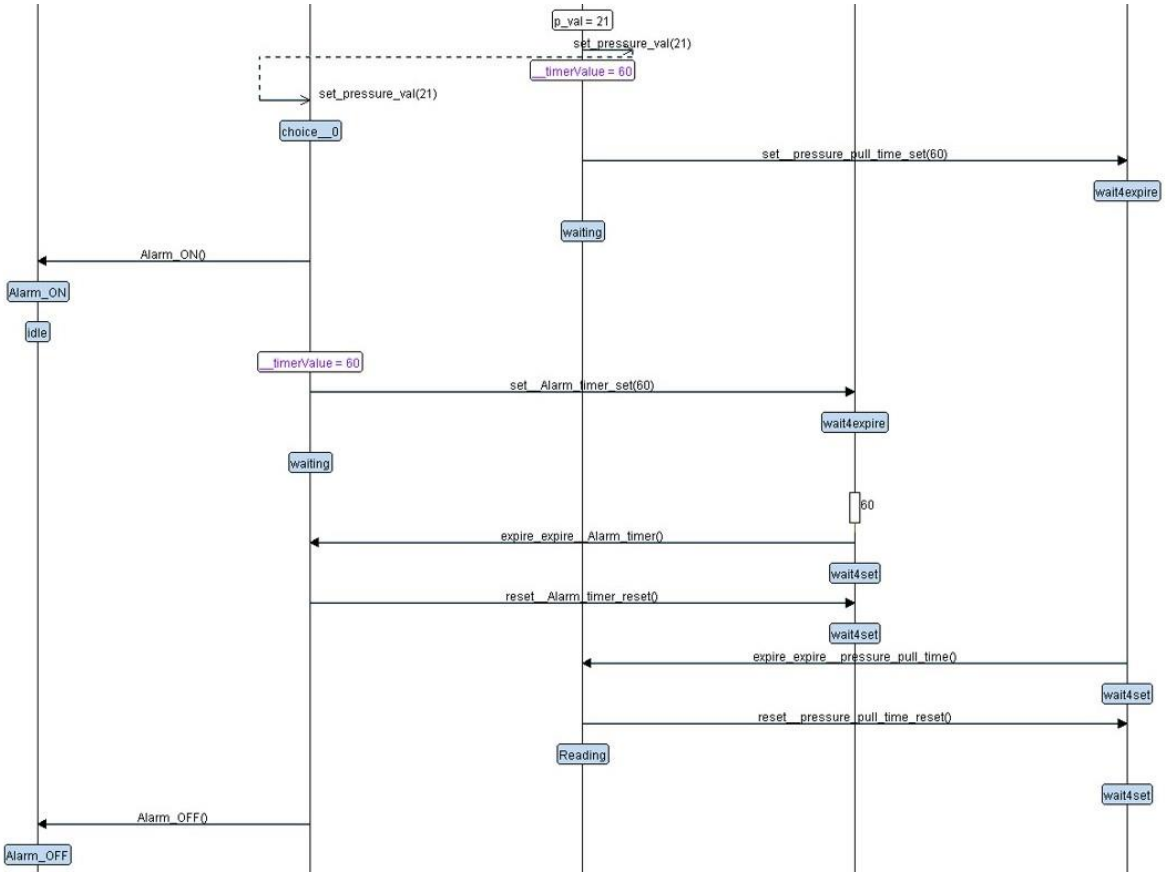
State Machine : alarm (LED)



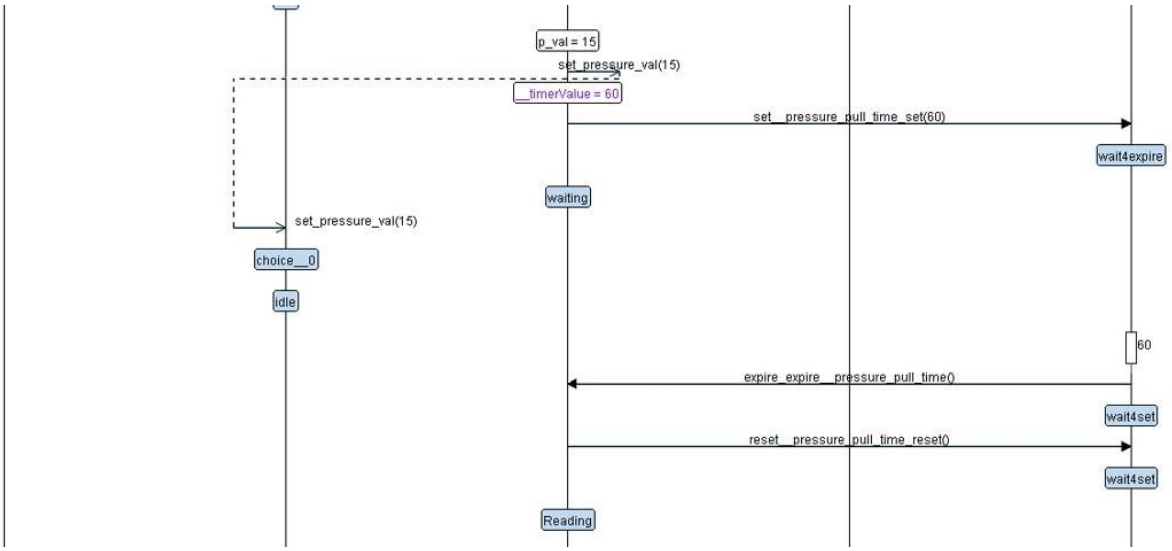
At program start



Case 1: Pressure (21)



Case 2: Pressure (15)



When we finished the implementation

1.Symbol of driver

```
$ arm-none-eabi-nm.exe driver.o
00000000 T Delay
00000022 T getPressureVal
00000074 T GPIO_INITIALIZATION
00000038 T Set_Alarm_actuator
```

2.Symbol of Pressure sensor

```
$ arm-none-eabi-nm.exe P_Sensor.o
         U getPressureVal
00000000 T read_p_sensor
```

3.Symbol of main function

```
$ arm-none-eabi-nm.exe main.o
         U Alarm_turn_OFF
         U Alarm_turn_ON
         U Delay
         U GPIO_INITIALIZATION
00000000 T main
         U read_p_sensor
```

4.symbol of Alarm(LED)

```
$ arm-none-eabi-nm.exe Alarm.o
0000000e T Alarm_turn_OFF
00000000 T Alarm_turn_ON
         U Set_Alarm_actuator
```

5.symbol of pressure controller

```
$ arm-none-eabi-nm.exe Pressure_Controller.elf
08000194 t _reset
080000ac T Alarm_turn_OFF
0800009e T Alarm_turn_ON
080000d0 T Delay
080000f2 T getPressureVal
08000144 T GPIO_INITIALIZATION
08000050 T main
080000ba T read_p_sensor
08000108 T Set_Alarm_actuator
0800019a t Vector_handler
```

6.header of driver

```
$ arm-none-eabi-objdump.exe -h driver.o

driver.o:     file format elf32-littlearm

Sections:
Idx Name          Size      VMA       LMA       File off  Algn
  0 .text         000000c4  00000000  00000000  00000034  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
  1 .data         00000000  00000000  00000000  000000f8  2**0
                  CONTENTS, ALLOC, LOAD, DATA
  2 .bss          00000000  00000000  00000000  000000f8  2**0
                  ALLOC
  3 .debug_info   00000112  00000000  00000000  000000f8  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING, OCTETS
  4 .debug_abbrev 000000c3  00000000  00000000  0000020a  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  5 .debug_loc    00000140  00000000  00000000  000002cd  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  6 .debug_aranges 00000020  00000000  00000000  0000040d  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING, OCTETS
  7 .debug_line   0000014f  00000000  00000000  0000042d  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING, OCTETS
  8 .debug_str    000001b1  00000000  00000000  0000057c  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  9 .comment      0000004a  00000000  00000000  0000072d  2**0
                  CONTENTS, READONLY
 10 .debug_frame  000000a0  00000000  00000000  00000778  2**2
                  CONTENTS, RELOC, READONLY, DEBUGGING, OCTETS
 11 .ARM.attributes 0000002d  00000000  00000000  00000818  2**0
                  CONTENTS, READONLY
```

7.header of Pressure sensor

```
$ arm-none-eabi-objdump.exe -h  P_Sensor.o

P_Sensor.o:     file format elf32-littlearm

Sections:
Idx Name          Size      VMA       LMA       File off  Algn
  0 .text         00000016  00000000  00000000  00000034  2**1
                  CONTENTS, ALLOC, LOAD, RELOC, READONLY, CODE
  1 .data         00000000  00000000  00000000  0000004a  2**0
                  CONTENTS, ALLOC, LOAD, DATA
  2 .bss          00000000  00000000  00000000  0000004a  2**0
                  ALLOC
  3 .debug_info   000000a4  00000000  00000000  0000004a  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING, OCTETS
  4 .debug_abbrev 00000055  00000000  00000000  000000ee  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  5 .debug_loc    00000050  00000000  00000000  00000143  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  6 .debug_aranges 00000020  00000000  00000000  00000193  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING, OCTETS
  7 .debug_line   00000043  00000000  00000000  000001b3  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING, OCTETS
  8 .debug_str    00000170  00000000  00000000  000001f6  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  9 .comment      0000004a  00000000  00000000  00000366  2**0
                  CONTENTS, READONLY
 10 .debug_frame  00000034  00000000  00000000  000003b0  2**2
                  CONTENTS, RELOC, READONLY, DEBUGGING, OCTETS
 11 .ARM.attributes 0000002d  00000000  00000000  000003e4  2**0
                  CONTENTS, READONLY
```

8.header of main function

```
$ arm-none-eabi-objdump.exe -h  main.o

main.o:     file format elf32-littlearm

Sections:
Idx Name          Size      VMA       LMA       File off  Algn
  0 .text         0000004e  00000000  00000000  00000034  2**1
                  CONTENTS, ALLOC, LOAD, RELOC, READONLY, CODE
  1 .data         00000000  00000000  00000000  00000082  2**0
                  CONTENTS, ALLOC, LOAD, DATA
  2 .bss          00000000  00000000  00000000  00000082  2**0
                  ALLOC
  3 .debug_info   000000b4  00000000  00000000  00000082  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING, OCTETS
  4 .debug_abbrev 0000005a  00000000  00000000  00000136  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  5 .debug_loc    00000038  00000000  00000000  00000190  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  6 .debug_aranges 00000020  00000000  00000000  000001c8  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING, OCTETS
  7 .debug_line   0000005f  00000000  00000000  000001e8  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING, OCTETS
  8 .debug_str    0000016d  00000000  00000000  00000247  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  9 .comment      0000004a  00000000  00000000  000003b4  2**0
                  CONTENTS, READONLY
 10 .debug_frame  00000030  00000000  00000000  00000400  2**2
                  CONTENTS, RELOC, READONLY, DEBUGGING, OCTETS
 11 .ARM.attributes 0000002d  00000000  00000000  00000430  2**0
                  CONTENTS, READONLY
```

9.header of Alarm(LED)

```
$ arm-none-eabi-objdump.exe -h  Alarm.o

Alarm.o:     file format elf32-littlearm

Sections:
Idx Name          Size      VMA       LMA       File off  Algn
  0 .text         0000001c  00000000  00000000  00000034  2**1
                  CONTENTS, ALLOC, LOAD, RELOC, READONLY, CODE
  1 .data         00000000  00000000  00000000  00000050  2**0
                  CONTENTS, ALLOC, LOAD, DATA
  2 .bss          00000000  00000000  00000000  00000050  2**0
                  ALLOC
  3 .debug_info   000000a6  00000000  00000000  00000050  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING, OCTETS
  4 .debug_abbrev 00000042  00000000  00000000  000000f6  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  5 .debug_loc    00000058  00000000  00000000  00000138  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  6 .debug_aranges 00000020  00000000  00000000  00000190  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING, OCTETS
  7 .debug_line   00000046  00000000  00000000  000001b0  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING, OCTETS
  8 .debug_str    00000176  00000000  00000000  000001f6  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  9 .comment      0000004a  00000000  00000000  0000036c  2**0
                  CONTENTS, READONLY
 10 .debug_frame  00000048  00000000  00000000  000003b8  2**2
                  CONTENTS, RELOC, READONLY, DEBUGGING, OCTETS
 11 .ARM.attributes 0000002d  00000000  00000000  00000400  2**0
                  CONTENTS, READONLY
```

10. header of pressure controller

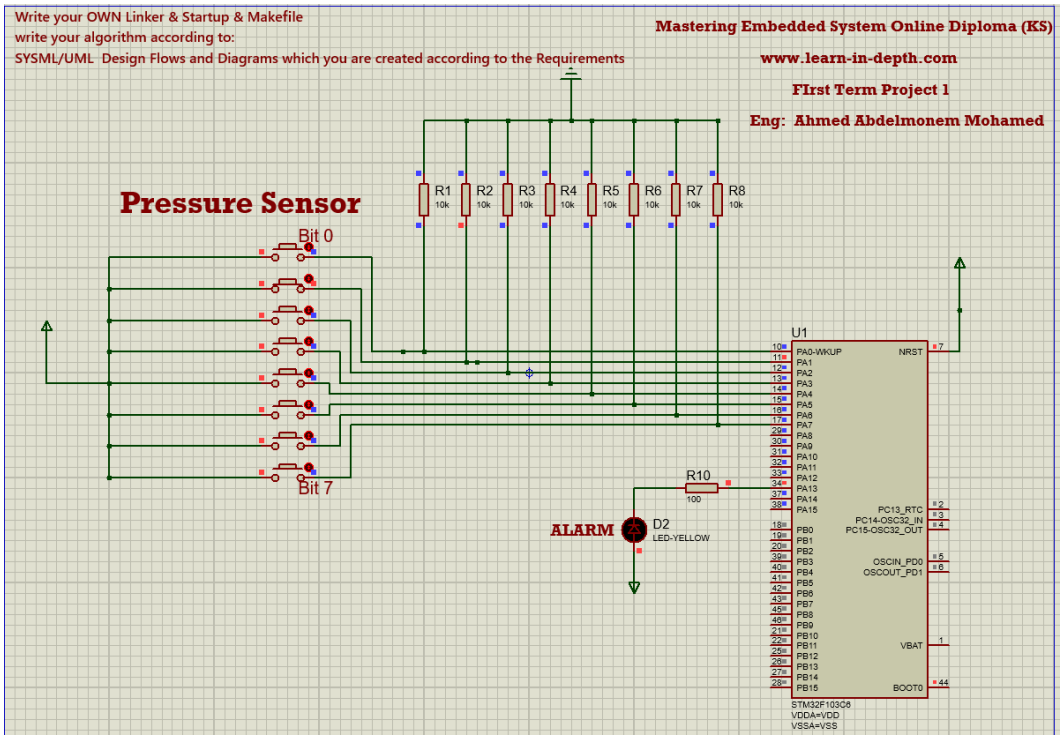
```
$ arm-none-eabi-objdump.exe -h  Pressure_Controller.elf

Pressure_Controller.elf:     file format elf32-littlearm

Sections:
Idx Name          Size      VMA       LMA       File off  Algn
  0 .text         0000019c  08000000  08000000  00010000  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
  1 .debug_info   00000336  00000000  00000000  0001019c  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  2 .debug_abbrev 000001c8  00000000  00000000  000104d2  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  3 .debug_loc    00000220  00000000  00000000  0001069a  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  4 .debug_aranges 000000a0  00000000  00000000  000108c0  2**3
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  5 .debug_line   00000272  00000000  00000000  00010960  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  6 .debug_str    000001e1  00000000  00000000  00010bd2  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  7 .comment      00000049  00000000  00000000  00010db3  2**0
                  CONTENTS, READONLY
  8 .ARM.attributes 0000002b  00000000  00000000  00010dfc  2**0
                  CONTENTS, READONLY
  9 .debug_frame  0000014c  00000000  00000000  00010e28  2**2
                  CONTENTS, READONLY, DEBUGGING, OCTETS
```

Simulation with proteus:

Case 1 : Pressure (4)



Case 1 : Pressure (34)