Mastering Embedded System Online Diploma

https://www.learn-in-depth.com

First Term Project 1: Pressure Detection System

Name: Ahmed Atef Ahmed Aseel

GitHub:

https://github.com/Ahmed-Aseel

Table Of Content: -

➢ Case study:

A client expects to deliver the software of the following system:

❖ Specification (from the client)

- A pressure controller informs the crew of a cabin with an alarm by turn on led when the pressure exceeds 20 bars in the cabin.
- The alarm duration equals 60 seconds.
- Store the measured values.

➢ Assumptions:

- The system setup and shutdown procedures are not modeled.
- The system maintenance is not modeled.
- The pressure sensor never fails.
- The alarm never fails.
- The system never faces power cut.

➤ Requirement Diagram:

We will divide our case study to three main Requirements:

- High Pressure Detection.
- Inform The Crew at High Pressure Levels.
- Optional Pressure Values Storage.

Each main requirement has refinement requirements as mentioned below.
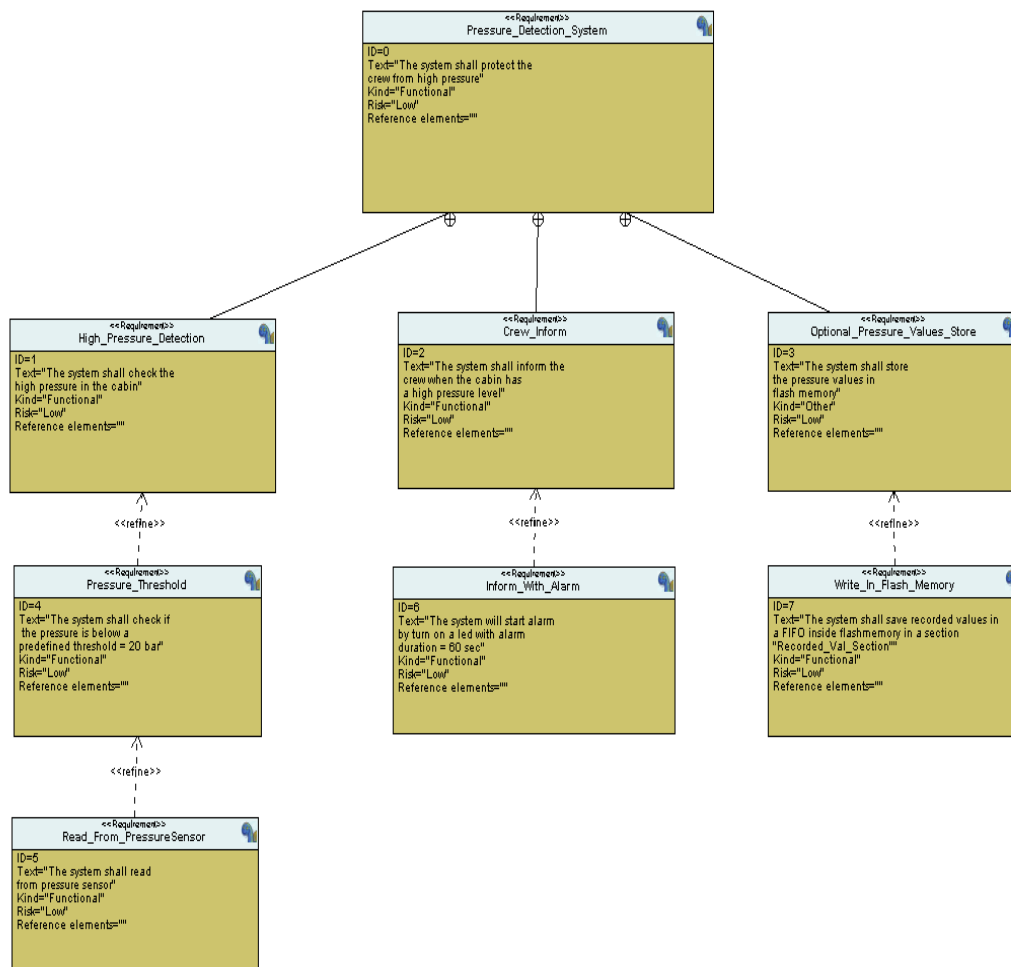
➢ System Analysis Diagram:

• Use Case Diagram:

In use case diagram, we discuss system boundary and main functions, so in our use case diagram we will observe that our system boundary includes the main algorithm, get pressure value and alarm monitor and exclude the pressure sensor, alarm actuator and flash memory for storing values.
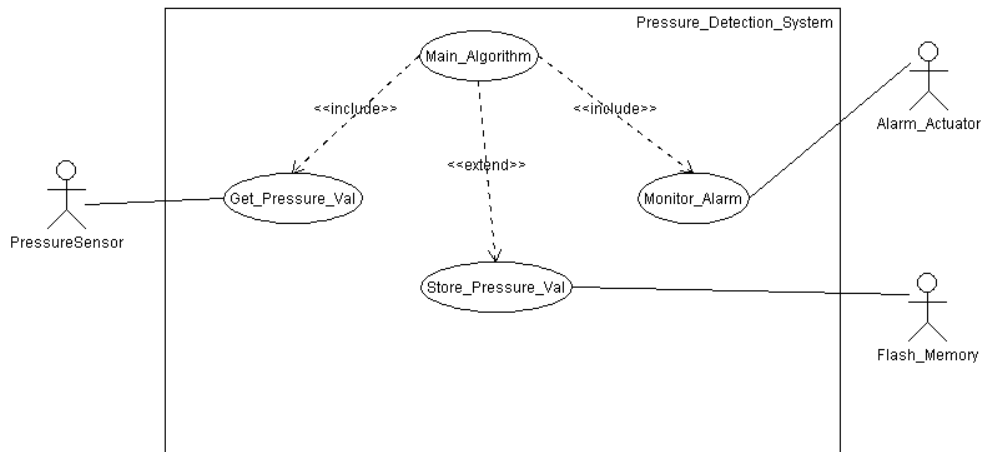
• Activity Diagram:

In Activity Diagram, we discuss relation between main functions, as shown below, firstly we will read the pressure value and then store it in flash memory, after that we compare the value of pressure with threshold in case study, if it exceeds

this value, we will turn on the alarm for 60 seconds to inform the crew that the pressure exceeded the threshold.

```
                    ●
                    │
                    ▼
       ┌──────────▷◇◁──────────┐
       │            │           │
       │        evt │           │
       │      ▷Read_PressureSensor_Val│
       │            │           │
       │            ▼           │
       │        sig             │
       │      Optional_Store_Pressure_Val▷
       │            │           │
       │            ▼           │
       │  []      ◇    [else]   │
       │          │             │
       │          │ [Pressure_Val > Threshold]
       │          ▼
       │     ┌──────────┐
       │     │Start_Alarm│
       │     └──────────┘
       │          │
       │       sig▼
       │     Alarm_ON▷
       │          │
       │          ▼
       │  ┌──────────────────────┐
       │  │Wait_Alarm_Time_Periode│
       │  └──────────────────────┘
       │          │
       │       sig▼
       └─────Alarm_OFF▷
```
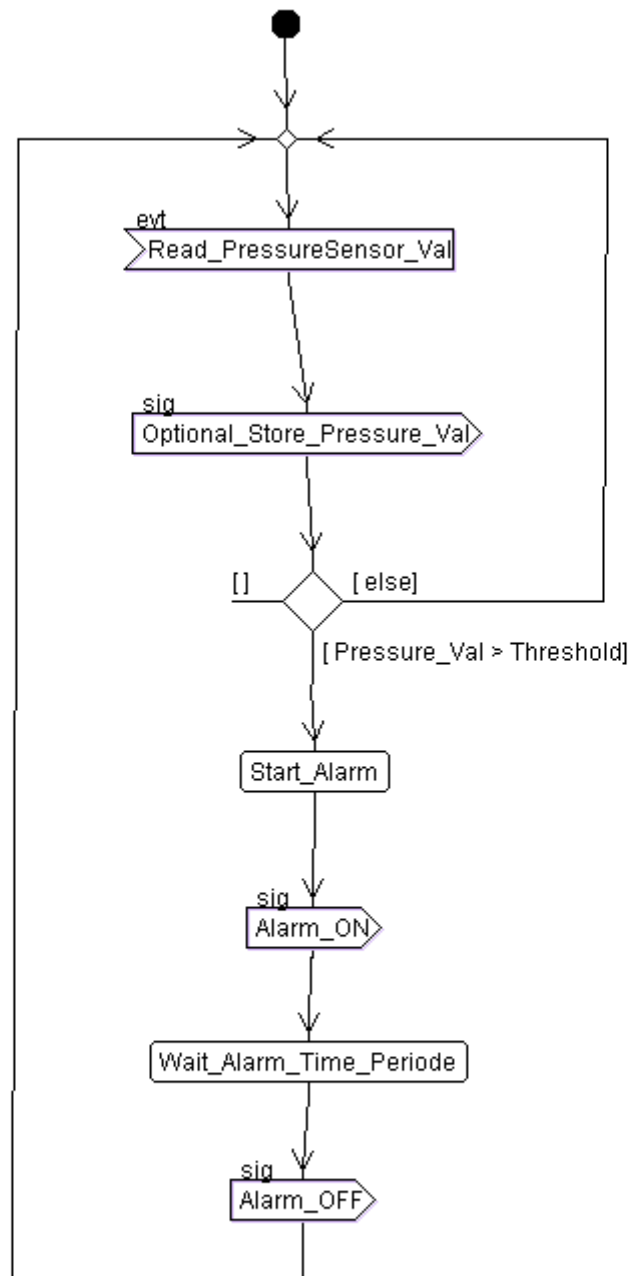
- **Sequence Diagram:**

In use Sequence Diagram, we discuss Communication between main system entities and actors, as we mentioned above in activity diagram, we will see interaction between Pressure sensor actor, main algorithm, alarm actuator and flash memory.
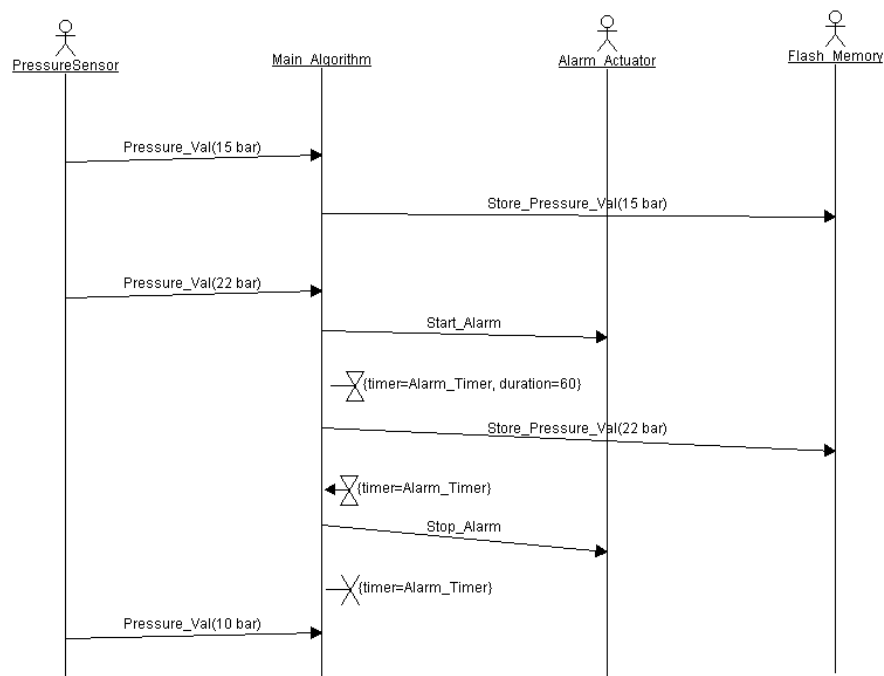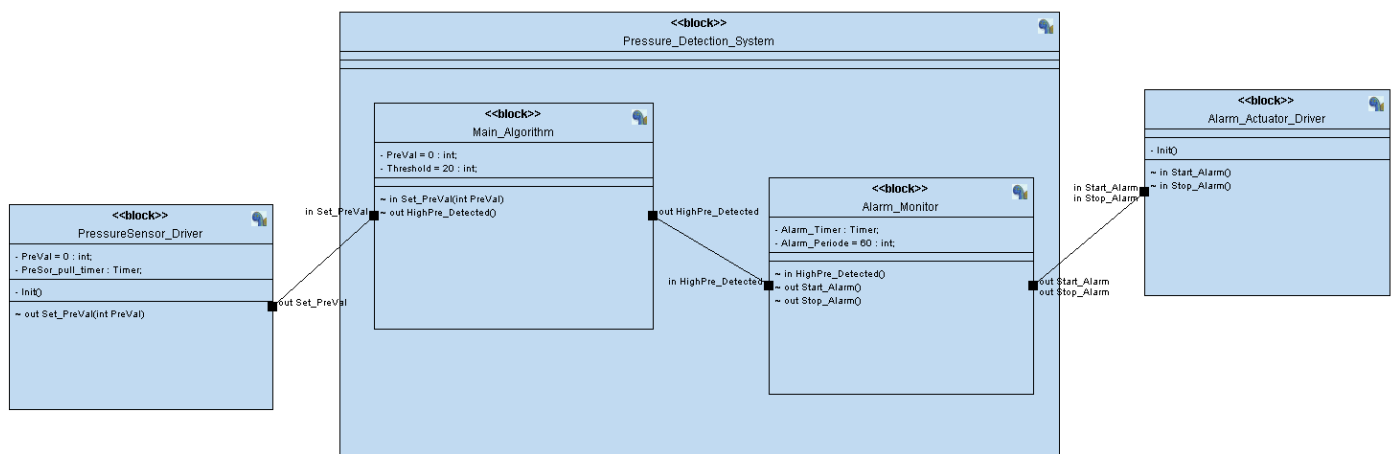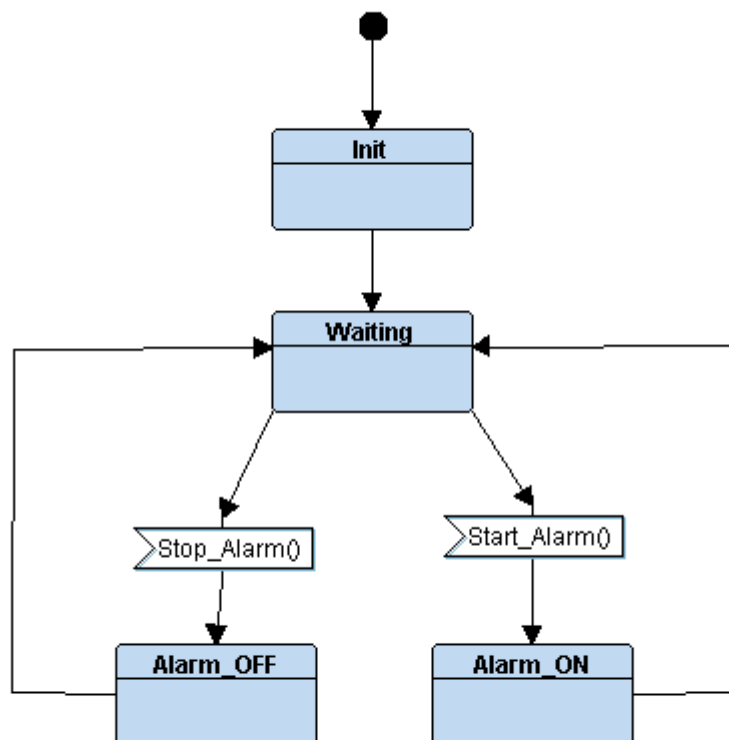

➢ **Block Diagrams & State Machines**
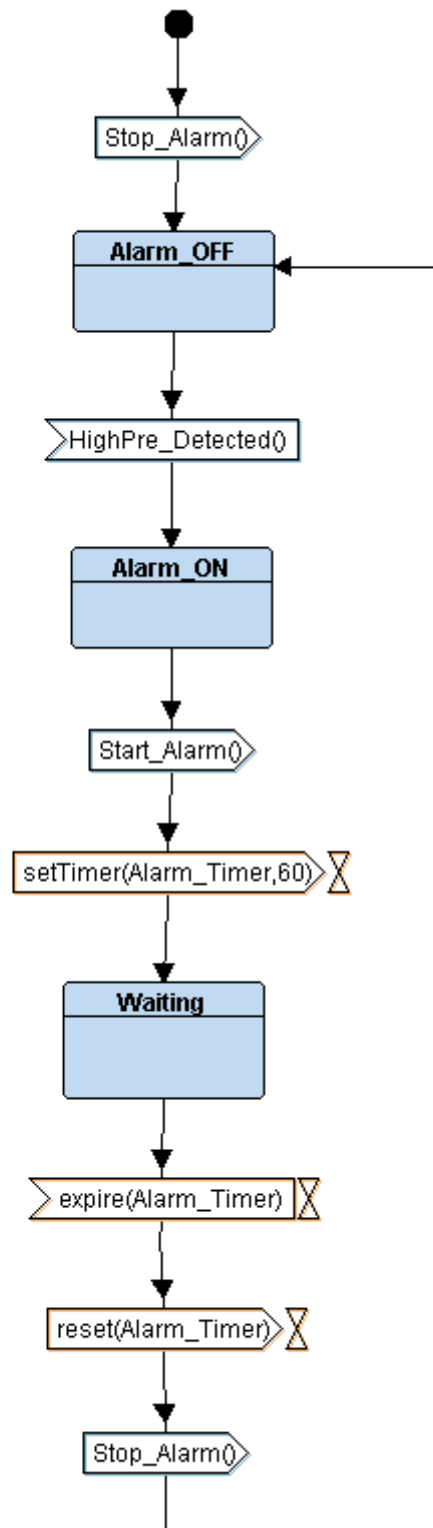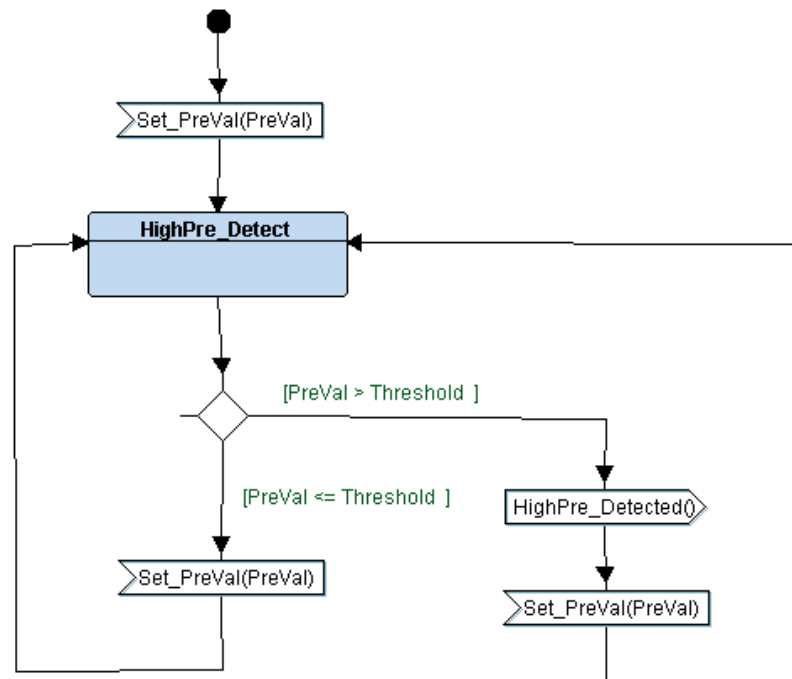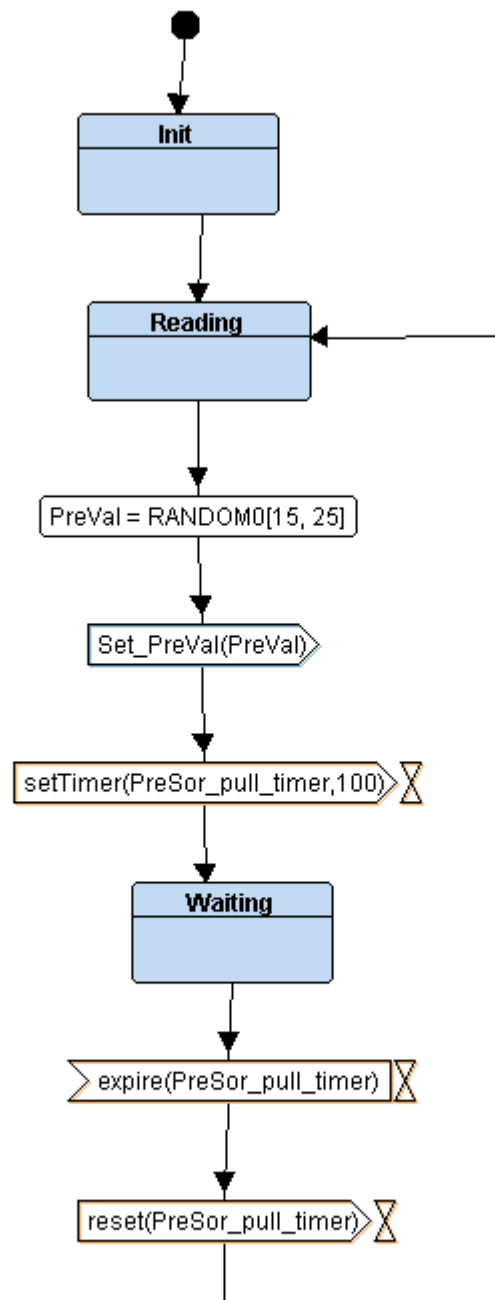
- ## Alarm_Actuator_State Machine

- **Alarm_Monitor_State Machine**

```
                    ●
                    │
                    ▼
            ┌──────────────┐
            │  Stop_Alarm()│
            └──────────────┘
                    │
                    ▼
          ┌──────────────────┐
          │    Alarm_OFF     │◄──────────┐
          ├──────────────────┤           │
          │                  │           │
          └──────────────────┘           │
                    │                     │
                    ▼                     │
          ┌──────────────────┐           │
          │ HighPre_Detected()│          │
          └──────────────────┘           │
                    │                     │
                    ▼                     │
          ┌──────────────────┐           │
          │     Alarm_ON     │           │
          ├──────────────────┤           │
          │                  │           │
          └──────────────────┘           │
                    │                     │
                    ▼                     │
            ┌──────────────┐             │
            │  Start_Alarm()│            │
            └──────────────┘             │
                    │                     │
                    ▼                     │
      ┌──────────────────────────┐       │
      │ setTimer(Alarm_Timer,60) │X      │
      └──────────────────────────┘       │
                    │                     │
                    ▼                     │
          ┌──────────────────┐           │
          │     Waiting      │           │
          ├──────────────────┤           │
          │                  │           │
          └──────────────────┘           │
                    │                     │
                    ▼                     │
      ┌──────────────────────┐           │
      │ expire(Alarm_Timer)  │X          │
      └──────────────────────┘           │
                    │                     │
                    ▼                     │
      ┌──────────────────────┐           │
      │ reset(Alarm_Timer)   │X          │
      └──────────────────────┘           │
                    │                     │
                    ▼                     │
            ┌──────────────┐             │
            │  Stop_Alarm()│             │
            └──────────────┘             │
                    │                     │
                    └─────────────────────┘
```

- ## **Main_Algo_State Machine**

Set_PreVal(PreVal)

HighPre_Detect

[PreVal > Threshold]

[PreVal <= Threshold]

Set_PreVal(PreVal)

HighPre_Detected()

Set_PreVal(PreVal)

Pre_Sensor_State Machine

```
         ●
         │
         ▼
    ┌─────────┐
    │  Init   │
    ├─────────┤
    │         │
    └─────────┘
         │
         ▼
    ┌─────────┐
    │ Reading │◄──────────────┐
    ├─────────┤               │
    │         │               │
    └─────────┘               │
         │                    │
         ▼                    │
  ( PreVal = RANDOM0[15, 25] )│
         │                    │
         ▼                    │
   ⟩Set_PreVal(PreVal)⟩       │
         │                    │
         ▼                    │
  ⟩setTimer(PreSor_pull_timer,100)⟩⋈
         │                    │
         ▼                    │
    ┌─────────┐               │
    │ Waiting │               │
    ├─────────┤               │
    │         │               │
    └─────────┘               │
         │                    │
         ▼                    │
   ⟨expire(PreSor_pull_timer)⟨⋈
         │                    │
         ▼                    │
   ⟩reset(PreSor_pull_timer)⟩⋈
         │                    │
         └────────────────────┘
```

➢ Sections & symbols for object files

- **Main**

```
metro@Aseel MINGW32 /g/Embedded/Keroles shinouda/Unit5 First_Final/FIRST_TERM_project1/COTS
$ 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         00000030  00000000  00000000  00000034  2**2
                  CONTENTS, ALLOC, LOAD, RELOC, READONLY, CODE
  1 .data         00000000  00000000  00000000  00000064  2**0
                  CONTENTS, ALLOC, LOAD, DATA
  2 .bss          00000000  00000000  00000000  00000064  2**0
                  ALLOC
  3 .debug_info   0000009b  00000000  00000000  00000064  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING
  4 .debug_abbrev 00000051  00000000  00000000  000000ff  2**0
                  CONTENTS, READONLY, DEBUGGING
  5 .debug_loc    00000038  00000000  00000000  00000150  2**0
                  CONTENTS, READONLY, DEBUGGING
  6 .debug_aranges 00000020  00000000  00000000  00000188  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING
  7 .debug_line   0000003d  00000000  00000000  000001a8  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING
  8 .debug_str    000000e2  00000000  00000000  000001e5  2**0
                  CONTENTS, READONLY, DEBUGGING
  9 .comment      00000012  00000000  00000000  000002c7  2**0
                  CONTENTS, READONLY
 10 .ARM.attributes 00000033  00000000  00000000  000002d9  2**0
                  CONTENTS, READONLY
 11 .debug_frame  00000030  00000000  00000000  0000030c  2**2
                  CONTENTS, RELOC, READONLY, DEBUGGING
```

```
metro@Aseel MINGW32 /g/Embedded/Keroles shinouda/Unit5 First_Final/FIRST_TERM_project1/COTS
$ arm-none-eabi-nm.exe main.o
         U ALARM_INITIALIZATION
         U Delay
         U Get_Pressure_Val
         U GPIO_INITIALIZATION
         U High_Pressure_Detected
00000000 T main
```

- **GPIO**

```
metro@Aseel MINGW32 /g/Embedded/Keroles shinouda/Unit5 First_Final/FIRST_TERM_project1/COTS
$ arm-none-eabi-objdump.exe -h GPIO_Program.o

GPIO_Program.o:     file format elf32-littlearm

Sections:
Idx Name          Size      VMA       LMA       File off  Algn
  0 .text         00000080  00000000  00000000  00000034  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
  1 .data         00000000  00000000  00000000  000000b4  2**0
                  CONTENTS, ALLOC, LOAD, DATA
  2 .bss          00000000  00000000  00000000  000000b4  2**0
                  ALLOC
  3 .debug_info   00000093  00000000  00000000  000000b4  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING
  4 .debug_abbrev 0000004d  00000000  00000000  00000147  2**0
                  CONTENTS, READONLY, DEBUGGING
  5 .debug_loc    0000002c  00000000  00000000  00000194  2**0
                  CONTENTS, READONLY, DEBUGGING
  6 .debug_aranges 00000020  00000000  00000000  000001c0  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING
  7 .debug_line   00000058  00000000  00000000  000001e0  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING
  8 .debug_str    000000fb  00000000  00000000  00000238  2**0
                  CONTENTS, READONLY, DEBUGGING
  9 .comment      00000012  00000000  00000000  00000333  2**0
                  CONTENTS, READONLY
 10 .ARM.attributes 00000033  00000000  00000000  00000345  2**0
                  CONTENTS, READONLY
 11 .debug_frame  00000028  00000000  00000000  00000378  2**2
                  CONTENTS, RELOC, READONLY, DEBUGGING
```

```
metro@Aseel MINGW32 /g/Embedded/Keroles shinouda/Unit5 First_Final/FIRST_TERM_project1/COTS
$ arm-none-eabi-nm.exe GPIO_Program.o
00000000 T GPIO_INITIALIZATION
```

- **PreSen_Program**

```
metro@Aseel MINGW32 /g/Embedded/Keroles shinouda/Unit5 First_Final/FIRST_TERM_project1/COTS
$ arm-none-eabi-objdump.exe -h PreSen_Program.o

PreSen_Program.o:     file format elf32-littlearm

Sections:
Idx Name          Size      VMA       LMA       File off  Algn
  0 .text         00000018  00000000  00000000  00000034  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
  1 .data         00000000  00000000  00000000  0000004c  2**0
                  CONTENTS, ALLOC, LOAD, DATA
  2 .bss          00000000  00000000  00000000  0000004c  2**0
                  ALLOC
  3 .debug_info   00000097  00000000  00000000  0000004c  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING
  4 .debug_abbrev 0000004f  00000000  00000000  000000e3  2**0
                  CONTENTS, READONLY, DEBUGGING
  5 .debug_loc    0000002c  00000000  00000000  00000132  2**0
                  CONTENTS, READONLY, DEBUGGING
  6 .debug_aranges 00000020  00000000  00000000  0000015e  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING
  7 .debug_line   00000056  00000000  00000000  0000017e  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING
  8 .debug_str    000000fa  00000000  00000000  000001d4  2**0
                  CONTENTS, READONLY, DEBUGGING
  9 .comment      00000012  00000000  00000000  000002ce  2**0
                  CONTENTS, READONLY
 10 .ARM.attributes 00000033  00000000  00000000  000002e0  2**0
                  CONTENTS, READONLY
 11 .debug_frame  00000028  00000000  00000000  00000314  2**2
                  CONTENTS, RELOC, READONLY, DEBUGGING
```

```
metro@Aseel MINGW32 /g/Embedded/Keroles shinouda/Unit5 First_Final/FIRST_TERM_project1/COTS
$ arm-none-eabi-nm.exe PreSen_Program.o
00000000 T Get_Pressure_Val
```

• AlmMon_Program

```
metro@Aseel MINGW32 /g/Embedded/Keroles shinouda/Unit5 First_Final/FIRST_TERM_project1/COTS
$ arm-none-eabi-objdump.exe -h AlmMon_Program.o

AlmMon_Program.o:     file format elf32-littlearm

Sections:
Idx Name          Size      VMA       LMA       File off  Algn
  0 .text         00000058  00000000  00000000  00000034  2**2
                  CONTENTS, ALLOC, LOAD, RELOC, READONLY, CODE
  1 .data         00000000  00000000  00000000  0000008c  2**0
                  CONTENTS, ALLOC, LOAD, DATA
  2 .bss          00000000  00000000  00000000  0000008c  2**0
                  ALLOC
  3 .debug_info   000000cd  00000000  00000000  0000008c  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING
  4 .debug_abbrev 00000074  00000000  00000000  00000159  2**0
                  CONTENTS, READONLY, DEBUGGING
  5 .debug_loc    00000090  00000000  00000000  000001cd  2**0
                  CONTENTS, READONLY, DEBUGGING
  6 .debug_aranges 00000020  00000000  00000000  0000025d  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING
  7 .debug_line   0000006a  00000000  00000000  0000027d  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING
  8 .debug_str    00000122  00000000  00000000  000002e7  2**0
                  CONTENTS, READONLY, DEBUGGING
  9 .comment      00000012  00000000  00000000  00000409  2**0
                  CONTENTS, READONLY
 10 .ARM.attributes 00000033  00000000  00000000  0000041b  2**0
                  CONTENTS, READONLY
 11 .debug_frame  00000064  00000000  00000000  00000450  2**2
                  CONTENTS, RELOC, READONLY, DEBUGGING
```

```
metro@Aseel MINGW32 /g/Embedded/Keroles shinouda/Unit5 First_Final/FIRST_TERM_project1/COTS
$ arm-none-eabi-nm.exe AlmMon_Program.o
00000000 T ALARM_INITIALIZATION
00000034 T Delay
00000010 T High_Pressure_Detected
         U Set_Alarm_Actuator
```

- **AlmAct_Program**

```
metro@Aseel MINGW32 /g/Embedded/Keroles shinouda/Unit5 First_Final/FIRST_TERM_project1/COTS
$ arm-none-eabi-objdump.exe -h AlmAct_Program.o

AlmAct_Program.o:     file format elf32-littlearm

Sections:
Idx Name          Size      VMA       LMA       File off  Algn
  0 .text         00000070  00000000  00000000  00000034  2**2
                  CONTENTS, ALLOC, LOAD, RELOC, READONLY, CODE
  1 .data         00000000  00000000  00000000  000000a4  2**0
                  CONTENTS, ALLOC, LOAD, DATA
  2 .bss          00000000  00000000  00000000  000000a4  2**0
                  ALLOC
  3 .debug_info   000000da  00000000  00000000  000000a4  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING
  4 .debug_abbrev 00000076  00000000  00000000  0000017e  2**0
                  CONTENTS, READONLY, DEBUGGING
  5 .debug_loc    00000090  00000000  00000000  000001f4  2**0
                  CONTENTS, READONLY, DEBUGGING
  6 .debug_aranges 00000020  00000000  00000000  00000284  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING
  7 .debug_line   0000005f  00000000  00000000  000002a4  2**0
                  CONTENTS, RELOC, READONLY, DEBUGGING
  8 .debug_str    0000011b  00000000  00000000  00000303  2**0
                  CONTENTS, READONLY, DEBUGGING
  9 .comment      00000012  00000000  00000000  0000041e  2**0
                  CONTENTS, READONLY
 10 .ARM.attributes 00000033  00000000  00000000  00000430  2**0
                  CONTENTS, READONLY
 11 .debug_frame  00000060  00000000  00000000  00000464  2**2
                  CONTENTS, RELOC, READONLY, DEBUGGING
```

```
metro@Aseel MINGW32 /g/Embedded/Keroles shinouda/Unit5 First_Final/FIRST_TERM_project1/COTS
$ arm-none-eabi-nm.exe AlmAct_Program.o
00000000 T Set_Alarm_Actuator
00000028 T Start_Alarm
0000004c T Stop_Alarm
```

➢ Sections & Symbols for final elf file


```
metro@Aseel MINGW32 /g/Embedded/Keroles shinouda/Unit5 First_Final/FIRST_TERM_project1/COTS
$ arm-none-eabi-objdump.exe -h  Pressure_Detection_System.elf

Pressure_Detection_System.elf:     file format elf32-littlearm

Sections:
Idx Name          Size      VMA       LMA       File off  Algn
  0 .text         00000264  08000000  08000000  00008000  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
  1 .bss          00001000  20000000  08000264  00010000  2**0
                  ALLOC
  2 .debug_info   000004e0  00000000  00000000  00008264  2**0
                  CONTENTS, READONLY, DEBUGGING
  3 .debug_abbrev 0000029b  00000000  00000000  00008744  2**0
                  CONTENTS, READONLY, DEBUGGING
  4 .debug_loc    00000214  00000000  00000000  000089df  2**0
                  CONTENTS, READONLY, DEBUGGING
  5 .debug_aranges 000000c0 00000000  00000000  00008bf3  2**0
                  CONTENTS, READONLY, DEBUGGING
  6 .debug_line   0000026f  00000000  00000000  00008cb3  2**0
                  CONTENTS, READONLY, DEBUGGING
  7 .debug_str    00000228  00000000  00000000  00008f22  2**0
                  CONTENTS, READONLY, DEBUGGING
  8 .comment      00000011  00000000  00000000  0000914a  2**0
                  CONTENTS, READONLY
  9 .ARM.attributes 00000033 00000000  00000000  0000915b  2**0
                  CONTENTS, READONLY
 10 .debug_frame  00000190  00000000  00000000  00009190  2**2
                  CONTENTS, READONLY, DEBUGGING
```



```
MINGW32:/g/Embedded/Keroles shinouda/Unit5 First_Final/FIRST_TERM_project1/COTS

metro@Aseel MINGW32 /g/Embedded/Keroles shinouda/Unit5 First_Final/FIRST_TERM_pr
oject1/COTS
$ arm-none-eabi-nm.exe Pressure_Detection_System.elf
20000000 B _E_bss
20000000 T _E_data
08000264 T _E_text
20000000 B _S_bss
20000000 T _S_data
20001000 B _stack_top
080001f4 T ALARM_INITIALIZATION
08000138 W Bus_fault_handler
08000138 T default_handler
08000228 T Delay
0800024c T Get_Pressure_Val
08000174 T GPIO_INITIALIZATION
08000138 W H_fault_handler
08000204 T High_Pressure_Detected
08000144 T main
08000138 W MM_fault_handler
08000138 W NMI_handler
08000088 T reset_handler
08000018 T Set_Alarm_Actuator
08000040 T Start_Alarm
08000064 T Stop_Alarm
08000138 W Usage_fault_handler
08000000 T vectors
```

➢ Simulation using Proteus