

Mastering Embedded System Online Diploma

www.learn-in-depth.com

First Term (Final Project 1)

Eng. Osama Mahmoud Hanafi

My Profile:

<https://www.learn-in-depth.com/online-diploma/usamamahmoud.2022%40gmail.com>

➤ Case Study: A Pressure Detection System

➤ Specification from the Client

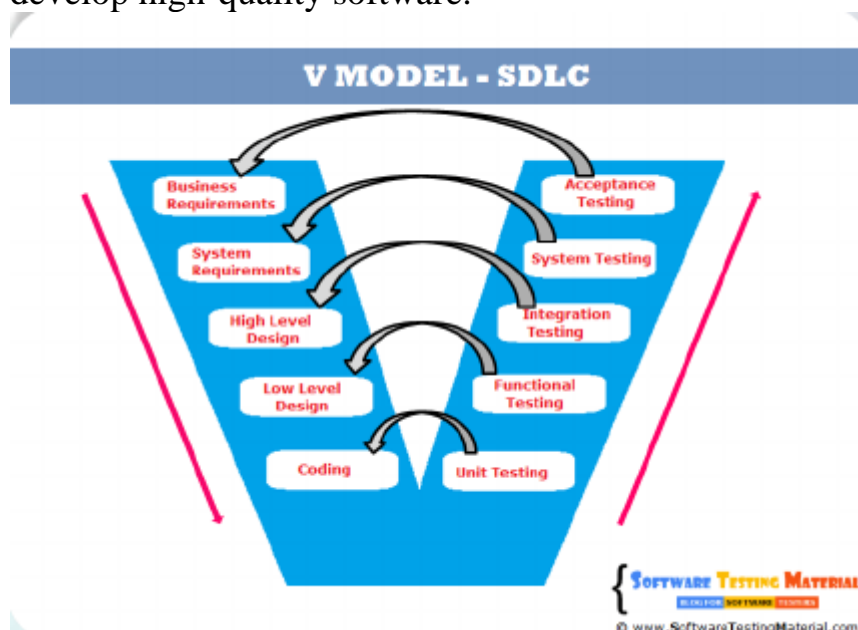
- A Pressure Detection informs the crew of a cabin with an alarm when the pressure exceeds 20 bars in the cabin.
- The alarm duration equals 60 Seconds.

➤ Method:

There are a lot of method used in a software development life cycle, and I decided to choose The **V Model – SDLC**.

➤ SDLC is **Software Development Life Cycle**

- It is the sequence of activities carried out by Developers to design and develop high-quality software.

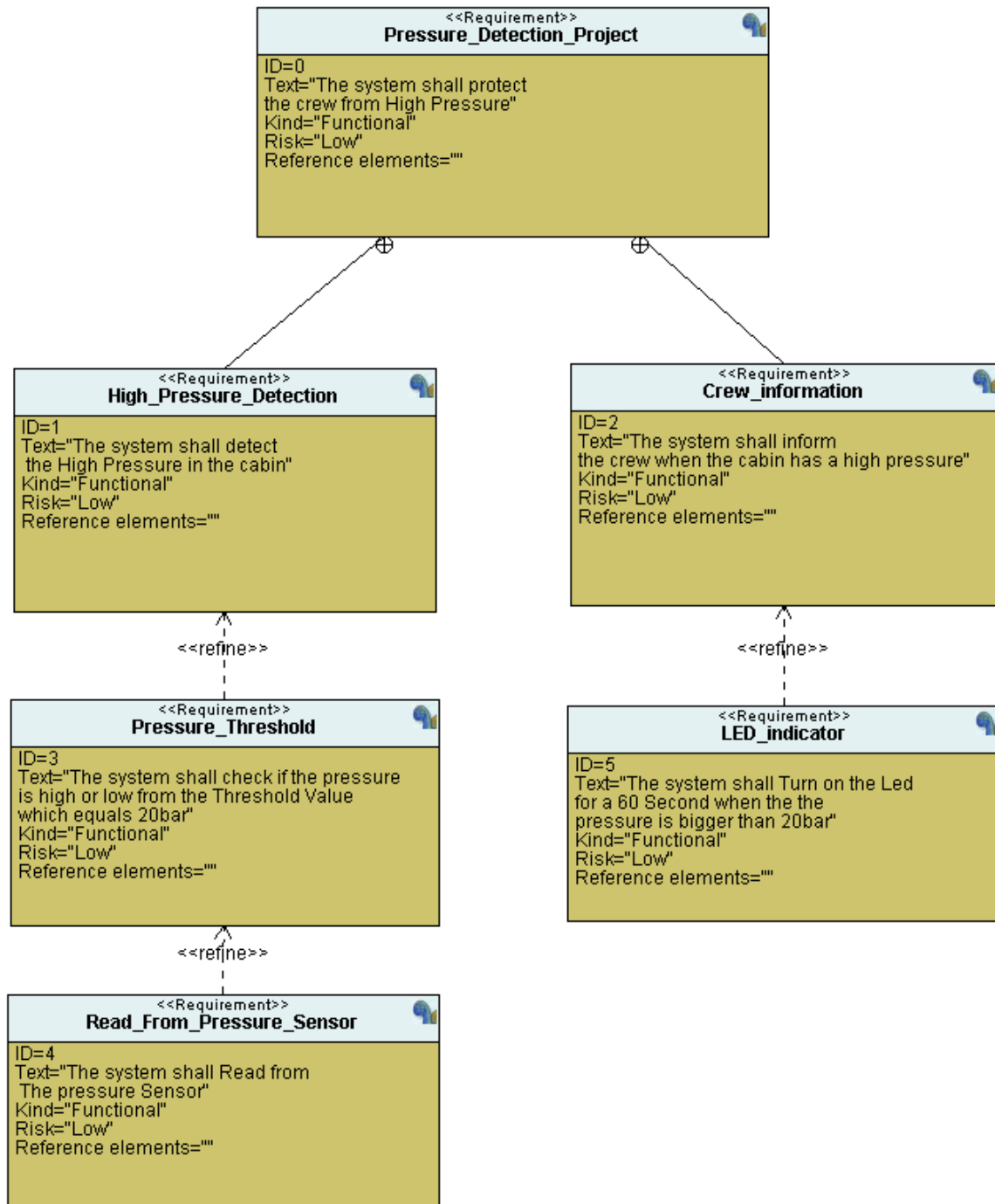


➤ Space Exploration:

The Multinational Companies use tools to know the number of ECU for the system. In our case for small application, I will use the **STM32F103** to Run the software of the Pressure detector System.

➤ Requirements:

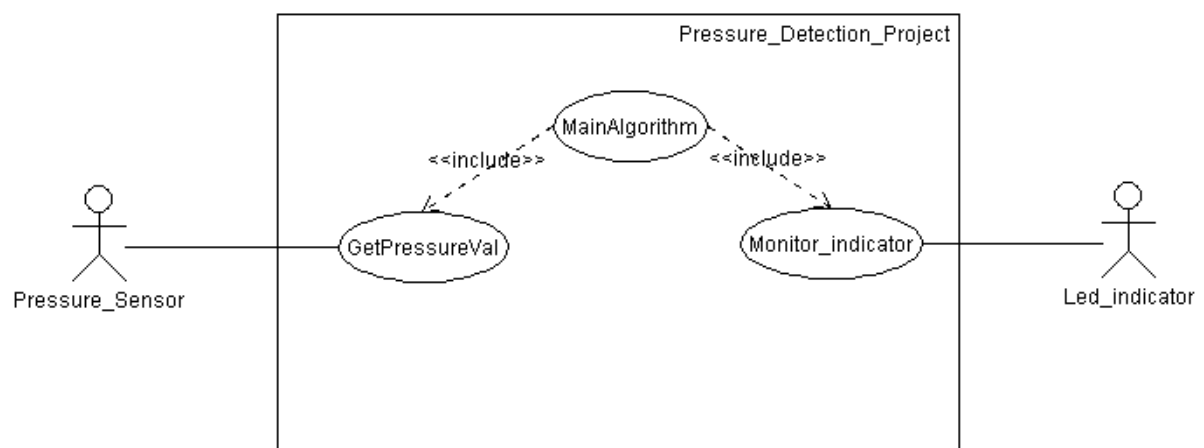
After talking with the customer, I got the requirements of this project and then I show the requirements diagram to the customer to ensure its right or not.



➤ System Analysis:

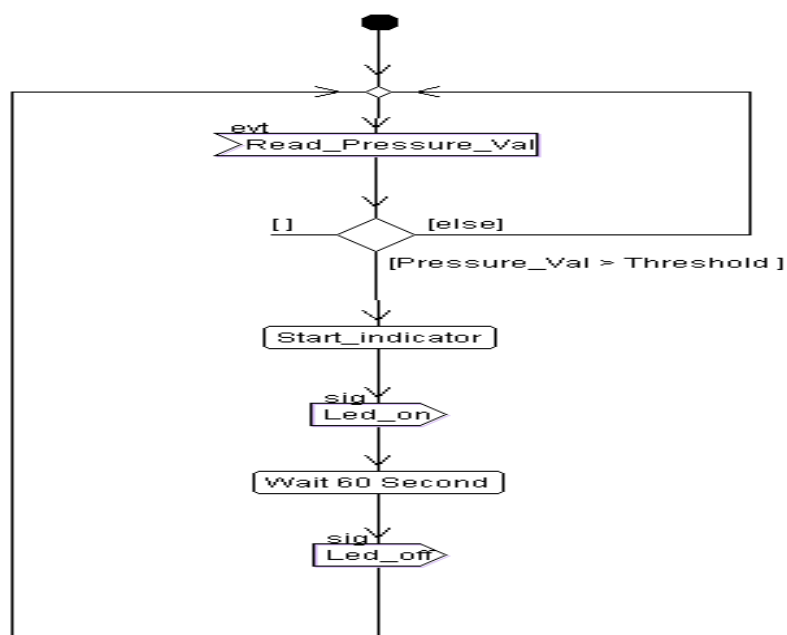
➤ Use Case Diagram

- System boundary and main functions.
- Not describe step by step algorithm.



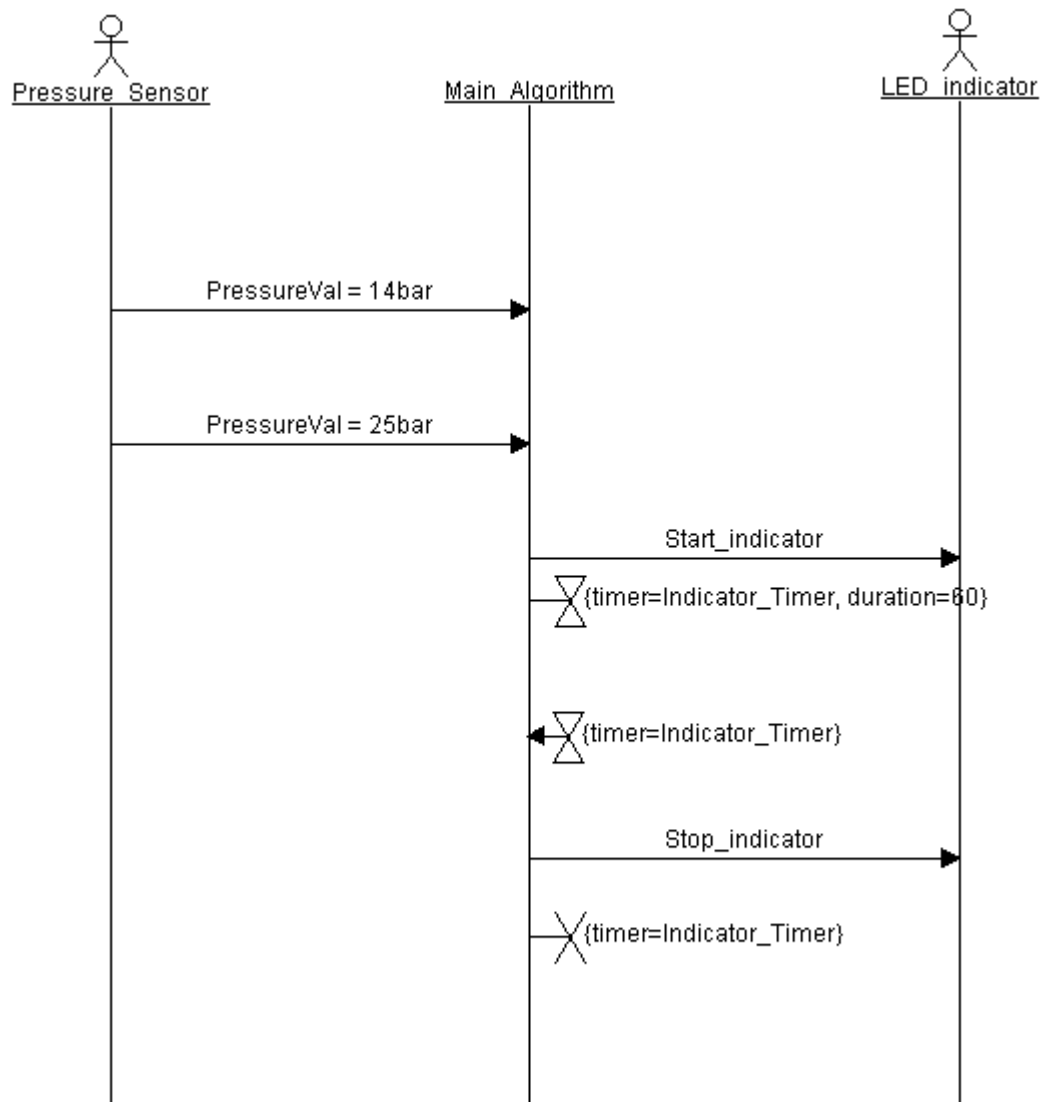
➤ Activity Diagram

- Describe the workflow behavior of a system.

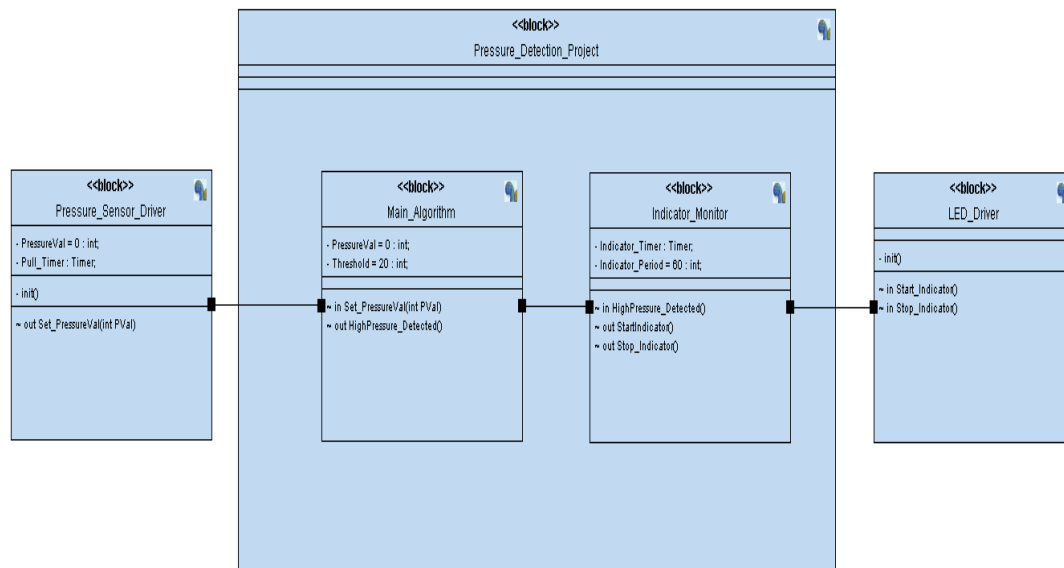


➤ Sequence Diagram

If Pressure Value is bigger than 20bar the indicator will start and wait for 60 seconds.

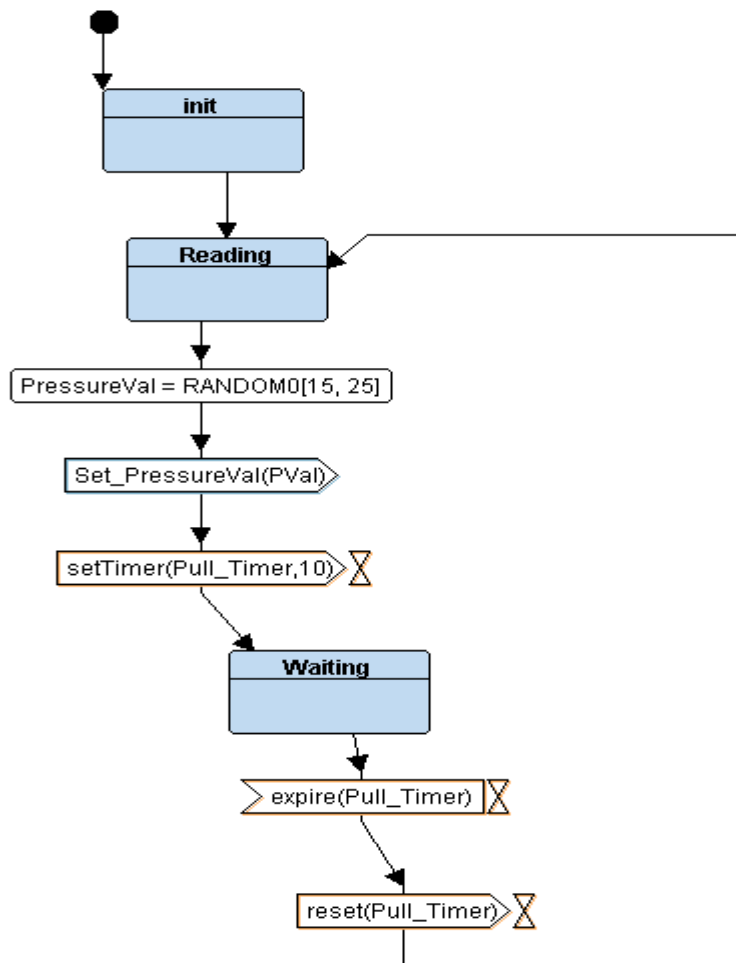


➤ System Design

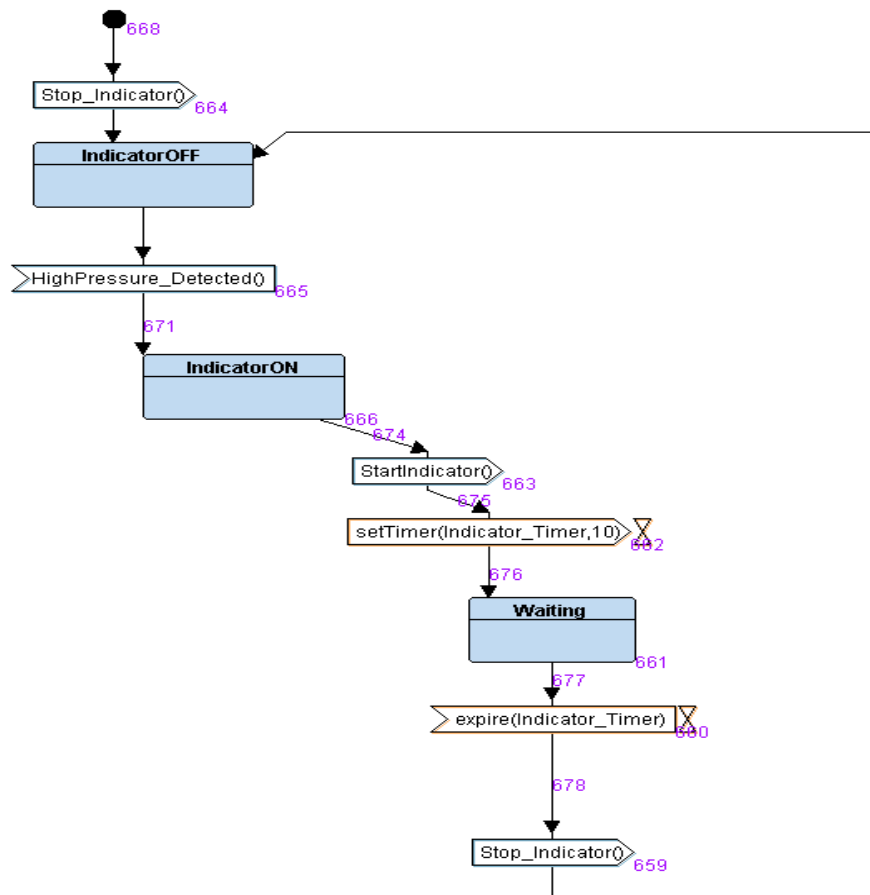


➤ Pressure Sensor Block

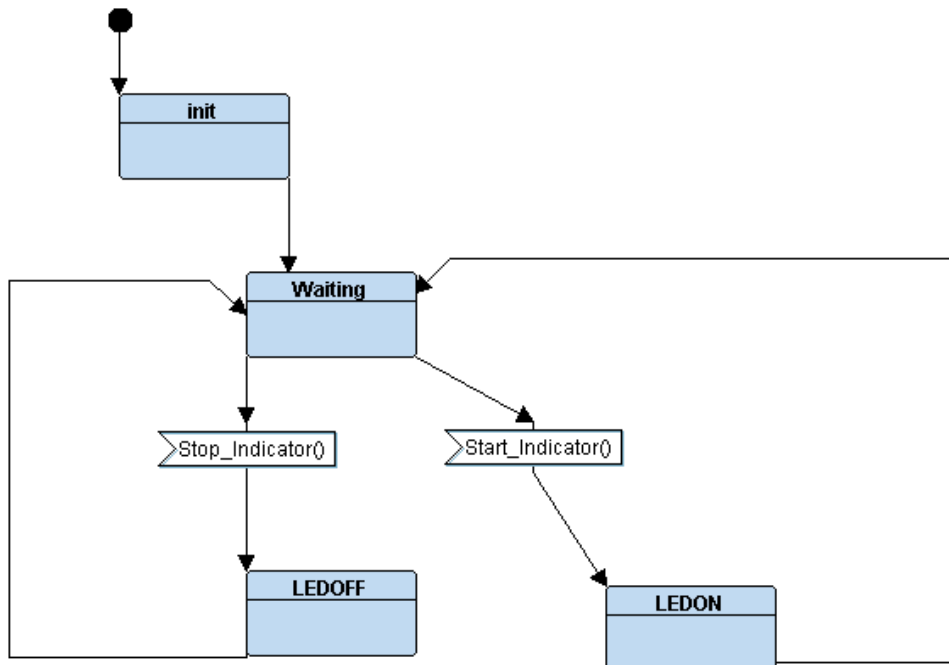
The time in Set Timer is 60 instead of 10.



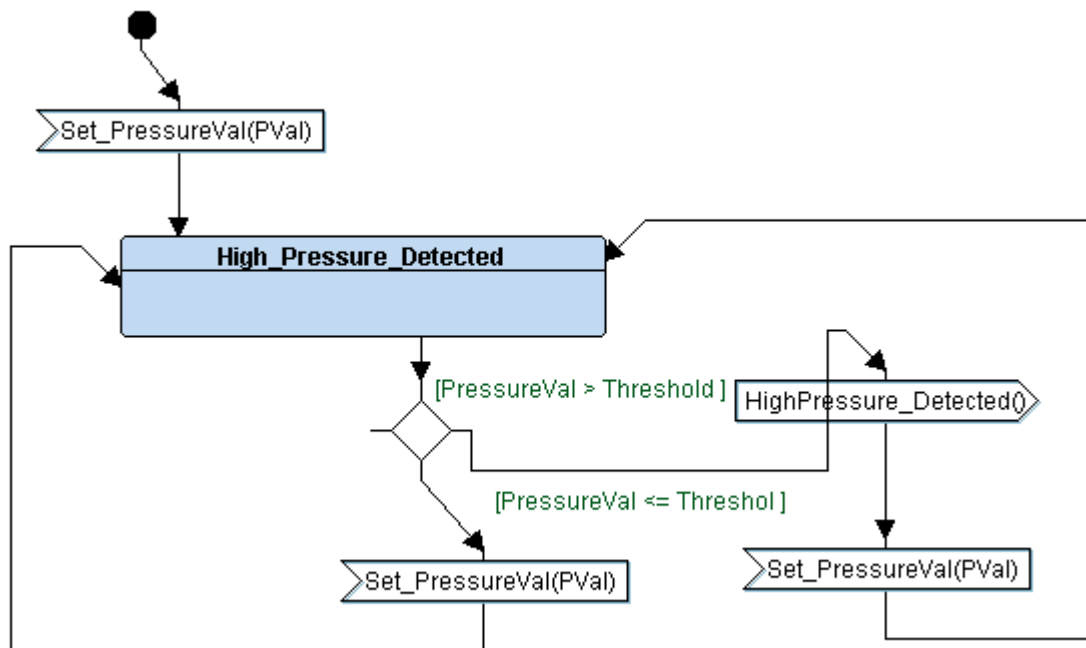
➤ Indicator Monitor



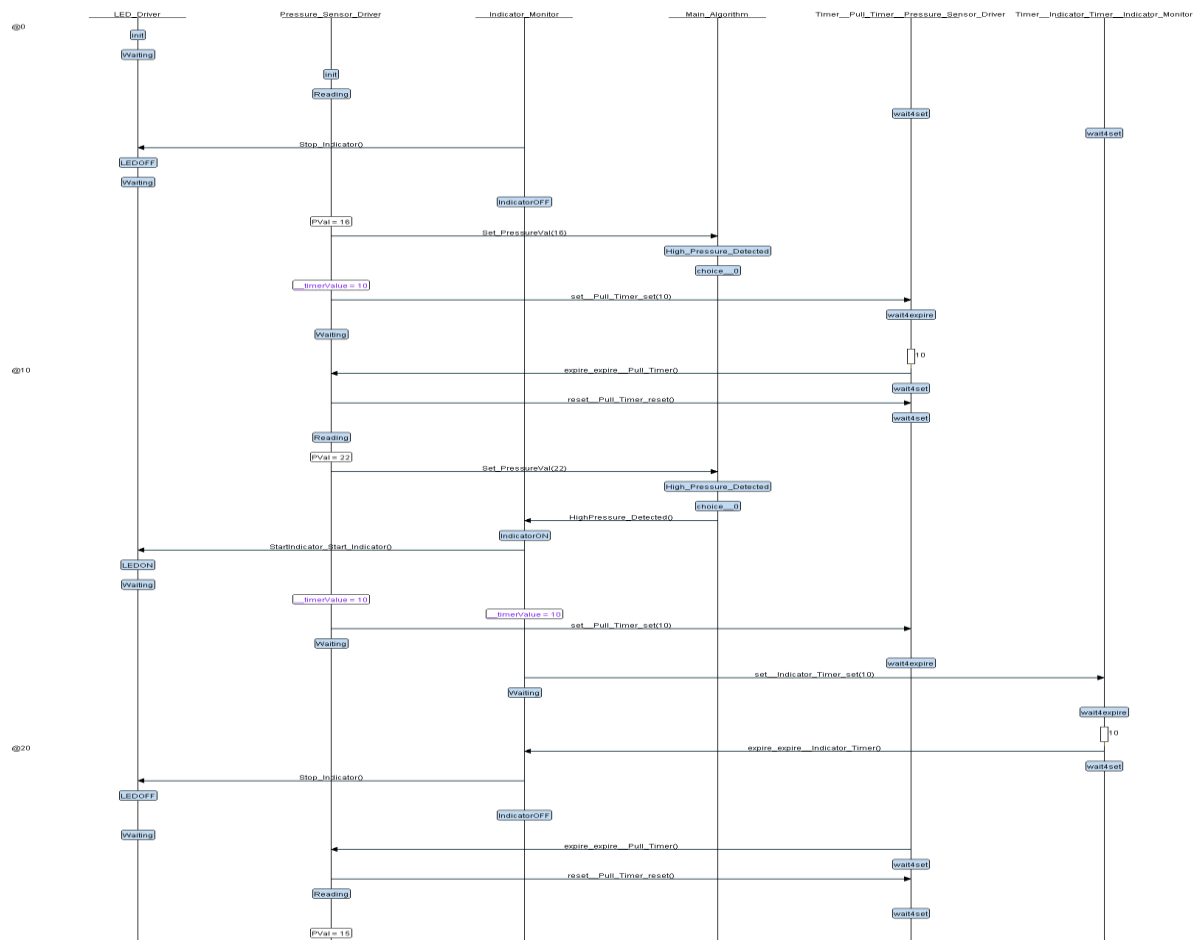
➤ LED Driver



➤ Main Algorithm



➤ Analysis and Timing Diagram



➤ Symbols for each block

➤ Symbols for PressureSensor.o

```
MINGW64:/f/OSAMAA/Embedded System/Learn In Depth/UNIT 5 FIRST TERM/...  
asss5@DESKTOP-J8I47FB MINGW64 /f/OSAMAA/Embedded System/Learn In Depth/UNIT 5 FI  
RST TERM/Project 1/Project1  
$ arm-none-eabi-nm.exe PressureSensor.o  
                 U getPressureVal  
                 U GPIO_INITIALIZATION  
00000000 b Pressure_Val  
00000000 T PS_init  
00000004 C PS_State  
00000001 C PS_State_Id  
                 U Set_PressureVal  
0000000c T ST_Reading
```

➤ Symbols for MainAlgorithm.o

```
asss5@DESKTOP-J8I47FB MINGW64 /f/OSAMAA/Embedded System/Learn In Depth/UNIT 5 FI  
RST TERM/Project 1/Project1  
$ arm-none-eabi-nm.exe MainAlgorithm.o  
                 U High_Pressure_Detected  
00000004 C Main_State  
00000000 b Pressure_Val  
00000001 C PS_State_Id  
00000000 T Set_PressureVal  
00000030 T ST_HIGH_Pressure_State  
00000000 d Threshold
```

➤ Symbols for MonitorIndicator.o

```
MINGW64:/f/OSAMAA/Embedded System/Learn In Depth/UNIT 5 FIRST TERM/...  
asss5@DESKTOP-J8I47FB MINGW64 /f/OSAMAA/Embedded System/Learn In Depth/UNIT 5 FI  
RST TERM/Project 1/Project1  
$ arm-none-eabi-nm.exe MonitorIndicator.o  
                 U Delay  
00000000 T High_Pressure_Detected  
00000000 D Indicator_period  
00000004 C Indicator_State  
00000001 C Indicator_State_Id  
00000058 T ST_Start_Indicator  
0000001c T ST_Stop_Indicator  
                 U Turn_Off  
                 U Turn_On
```

➤ Symbols for Led.o

```
MINGW64:/f/OSAMAA/Embedded System/Learn In Depth/UNIT 5 FIRST TERM/...  
asss5@DESKTOP-J8I47FB MINGW64 /f/OSAMAA/Embedded System/Learn In Depth/UNIT 5 FIRST TERM/Project 1/Project1  
$ arm-none-eabi-nm.exe Led.o  
U GPIO_INITIALIZATION  
00000001 C Indicator_State_Id  
00000000 T LED_init  
00000001 C LED_State_Id  
U Set_LED  
00000028 T Turn_Off  
0000000c T Turn_On
```

➤ Symbols for main.o

```
MINGW64:/f/OSAMAA/Embedded System/Learn In Depth/UNIT 5 FIRST TERM/...  
asss5@DESKTOP-J8I47FB MINGW64 /f/OSAMAA/Embedded System/Learn In Depth/UNIT 5 FIRST TERM/Project 1/Project1  
$ arm-none-eabi-nm.exe main.o  
U Indicator_State  
00000001 C Indicator_State_Id  
00000001 C LED_State_Id  
00000000 T main  
U Main_State  
U PS_init  
U PS_State  
00000001 C PS_State_Id  
U ST_HIGH_Pressure_State  
U ST_Reading  
U ST_Stop_Indicator
```

➤ Symbols for PressureDetector.elf

```
MINGW64:/f/OSAMAA/Embedded System/Learn In Depth/UNIT 5 FIRST TERM/Project 1/Project1  
asss5@DESKTOP-J8I47FB MINGW64 /f/OSAMAA/Embedded System/Learn In Depth/UNIT 5 FIRST TERM/Project 1/Project1  
$ arm-none-eabi-nm.exe First_term_project1_Pressure_Detector.elf  
20000010 B _E_bss  
20000008 D _E_data  
080003e0 T _E_text  
20000008 B _S_bss  
20000000 D _S_data  
20001010 B _stack_top  
080003d4 W Bus_Fault  
080003d4 T Default_handler  
0800001c T Delay  
08000040 T getPressureVal  
080000ac T GPIO_INITIALIZATION  
080003d4 W H_fault_Handler  
0800024c T High_Pressure_Detected  
20000004 D Indicator_period  
20001018 B Indicator_State  
20001010 B Indicator_State_Id  
0800012c T LED_init  
20001011 B LED_State_Id  
08000170 T main  
20001014 B Main_State  
080003d4 W MM_Fault_Handler  
080003d4 W NMI_Handler  
20000008 b Pressure_Val  
2000000c b Pressure_Val  
080002e4 T PS_init  
2000101c B PS_State  
20001012 B PS_State_Id  
08000324 T Reset_Handler  
08000058 T Set_LED  
080001d4 T Set_PressureVal  
08000204 T ST_HIGH_Pressure_State  
080002f0 T ST_Reading  
080002a4 T ST_Start_Indicator  
08000268 T ST_Stop_Indicator  
20000000 d Threshold  
08000154 T Turn_Off  
08000138 T Turn_On  
080003d4 W Usage_Fault_Handler  
08000000 T vectors
```

➤ Sections for each Block

➤ Section for PressureSensor.o

```
RST TERM/Project 1/Project1
$ arm-none-eabi-objdump.exe -h PressureSensor.o

PressureSensor.o:      file format elf32-littlearm

Sections:
Idx Name              Size      VMA          LMA          File off  Algn
  0 .text              00000040  00000000  00000000  00000034  2**2
                   CONTENTS, ALLOC, LOAD, RELOC, READONLY, CODE
  1 .data              00000000  00000000  00000000  00000074  2**0
                   CONTENTS, ALLOC, LOAD, DATA
  2 .bss               00000004  00000000  00000000  00000074  2**2
                   ALLOC
```

➤ Section for MainAlgorithm.o

```
asss5@DESKTOP-J8I47FB MINGW64 /f/OSAMAA/Embedded System/Learn In Depth/UNIT 5 F
RST TERM/Project 1/Project1
$ arm-none-eabi-objdump.exe -h MainAlgorithm.o

MainAlgorithm.o:      file format elf32-littlearm

Sections:
Idx Name              Size      VMA          LMA          File off  Algn
  0 .text              00000078  00000000  00000000  00000034  2**2
                   CONTENTS, ALLOC, LOAD, RELOC, READONLY, CODE
  1 .data              00000004  00000000  00000000  000000ac  2**2
                   CONTENTS, ALLOC, LOAD, DATA
  2 .bss               00000004  00000000  00000000  000000b0  2**2
                   ALLOC
```

➤ Section for MonitorIndicator.o

```
MINGW64:/f/OSAMAA/Embedded System/Learn In Depth/UNIT 5 FIRST TERM/...  -  □  ×
asss5@DESKTOP-J8I47FB MINGW64 /f/OSAMAA/Embedded System/Learn In Depth/UNIT 5 FI
RST TERM/Project 1/Project1
$ arm-none-eabi-objdump.exe -h MonitorIndicator.o

MonitorIndicator.o:   file format elf32-littlearm

Sections:
Idx Name              Size      VMA          LMA          File off  Algn
  0 .text              00000098  00000000  00000000  00000034  2**2
                   CONTENTS, ALLOC, LOAD, RELOC, READONLY, CODE
  1 .data              00000004  00000000  00000000  000000cc  2**2
                   CONTENTS, ALLOC, LOAD, DATA
  2 .bss               00000000  00000000  00000000  000000d0  2**0
                   ALLOC
```

➤ Section for Led.o

```

MINGW64:/f/OSAMAA/Embedded System/Learn In Depth/UNIT 5 FIRST TERM/...
asss5@DESKTOP-J8I47FB MINGW64 /f/OSAMAA/Embedded System/Learn In Depth/UNIT 5 FIRST TERM/Project 1/Project1
$ arm-none-eabi-objdump.exe -h Led.o

Led.o:      file format elf32-littlearm

Sections:
Idx Name          Size      VMA           LMA           File off  Algn
  0 .text          00000044  00000000  00000000  00000034  2**2
               CONTENTS, ALLOC, LOAD, RELOC, READONLY, CODE
  1 .data           00000000  00000000  00000000  00000078  2**0
               CONTENTS, ALLOC, LOAD, DATA
  2 .bss            00000000  00000000  00000000  00000078  2**0
               ALLOC

```

➤ Section PressureDetector.o

```

MINGW64:/f/OSAMAA/Embedded System/Learn In Depth/UNIT 5 FIRST TERM/...
First_term_project1_Pressure_Detector.elf:  file format elf32-littlearm

Sections:
Idx Name          Size      VMA           LMA           File off  Algn
  0 .text          000003e0  08000000  08000000  00008000  2**2
               CONTENTS, ALLOC, LOAD, READONLY, CODE
  1 .data           00000008  20000000  080003e0  00010000  2**2
               CONTENTS, ALLOC, LOAD, DATA
  2 .bss            00001018  20000008  080003e8  00010008  2**2
               ALLOC

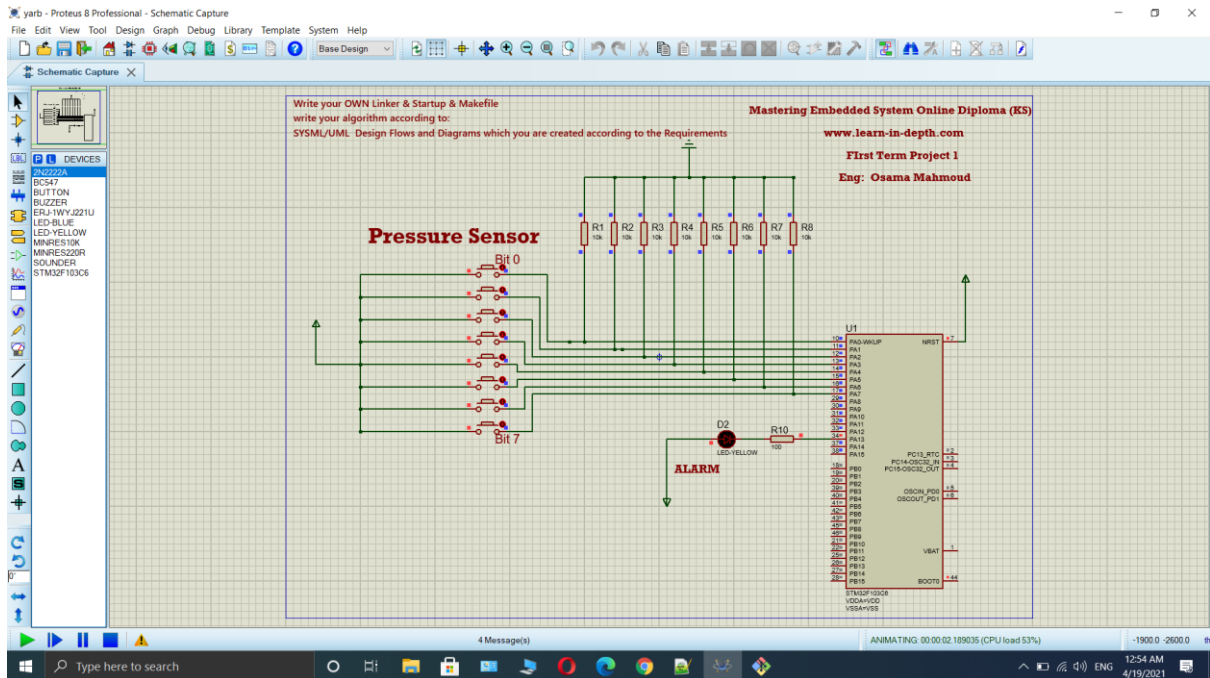
```

➤ MapFile

1	Allocating common symbols			
2	Common symbol	size	file	
3				
4				
5	PS_State_Id	0x1	main.o	
6	Indicator_State_Id	0x1	Led.o	
7	LED_State_Id	0x1	Led.o	
8	Indicator_State	0x4	MonitorIndicator.o	
9	PS_State	0x4	PressureSensor.o	
10	Main_State	0x4	MainAlgorithm.o	
11				
12	Memory Configuration			
13				
14	Name	Origin	Length	Attributes
15	flash	0x08000000	0x00020000	rx
16	sram	0x20000000	0x00005000	rxw
17	*default*	0x00000000	0xffffffff	
18				
19	Linker script and memory map			
20				
21				
22	.text	0x08000000	0x3e0	
23	*(.vectors*)			
24	.vectors	0x08000000	0x1c startup.o	
25		0x08000000	vectors	
26	*(.text*)			
27	.text	0x0800001c	0x110 driver.o	
28		0x0800001c	Delay	
29		0x08000040	getPressureVal	
30		0x08000058	Set_LED	
31		0x080000ac	GPIO_INITIALIZATION	
32	.text	0x0800012c	0x44 Led.o	
33		0x0800012c	LED_init	
34		0x08000138	Turn_On	
35		0x08000154	Turn_Off	

➤ Simulation

- If the pressure Value is less than 20 bar, the Led is off.



- If the Pressure is bigger than 20 bar, the is on for 60 Second.

