Pressure Controlling System

Embedded C Project

Final project

Radwa Mahmoud Bahey Eldeen Elhateem

Check code : github , drive

CV
mail

Pressure Controlling System

1st: CASE STUDY:

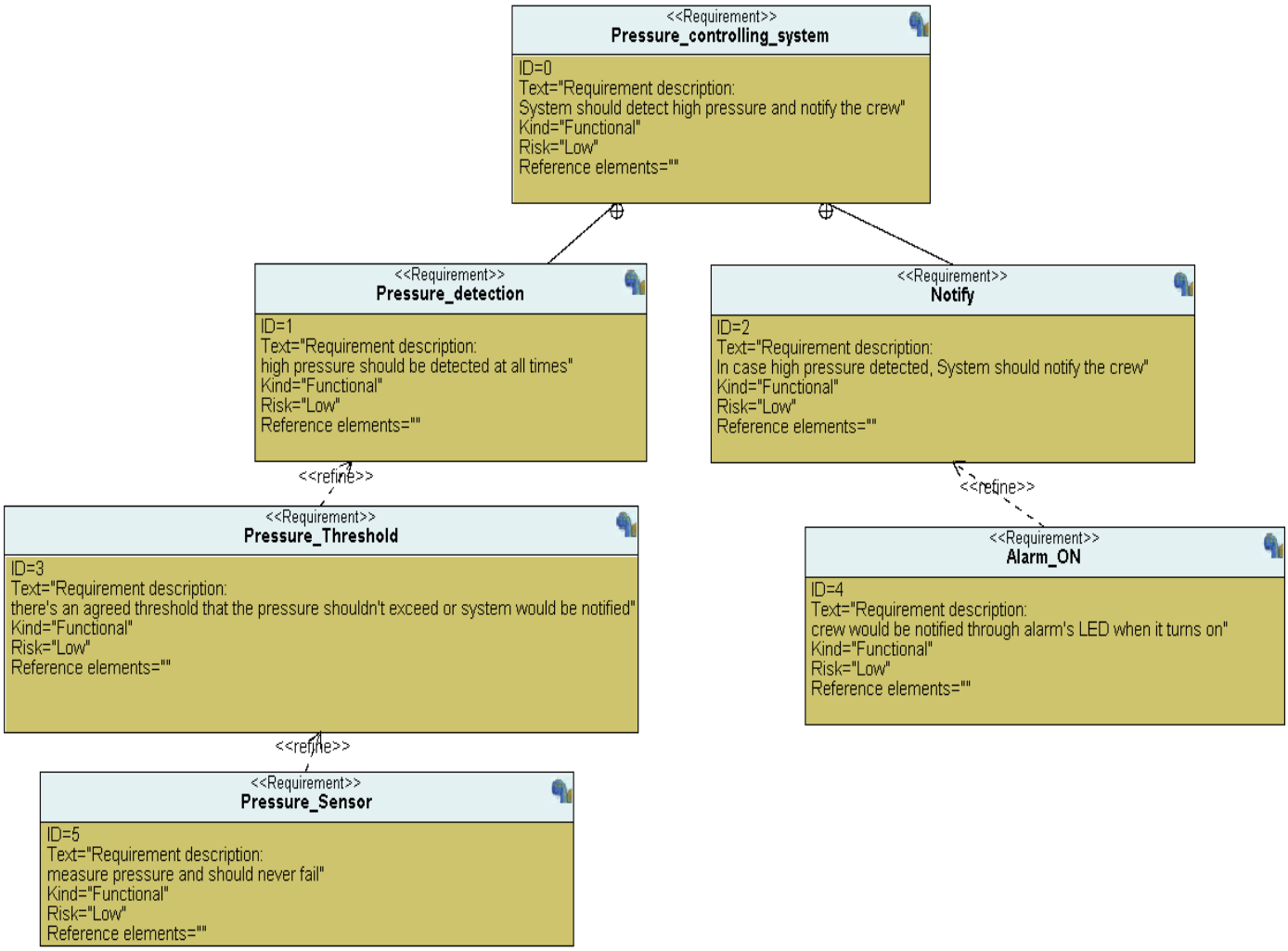
- No power outage at any cost.
- The reading of the pressure sensor is always accurate.
- The pressure sensor never fails.
- The duration for the alarm turning on is always accurate.
- The alarm never fails.
- Alarm duration 60 seconds
- The controller set up and shut down aren't modeled.
- The controller never faces power cut.
- The controller maintenance isn't modeled.
- The threshold for the pressure is set and agreed on.

Pressure Controlling System

2nd: METHOD:

- V-Model

3rd: REQUIRMENTS:

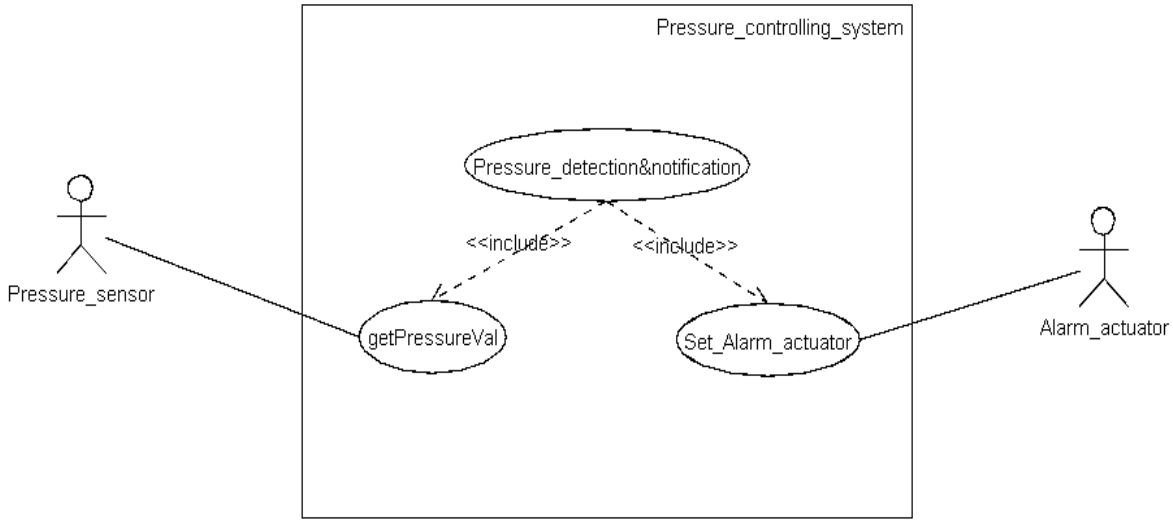


4th: Space Exploration:

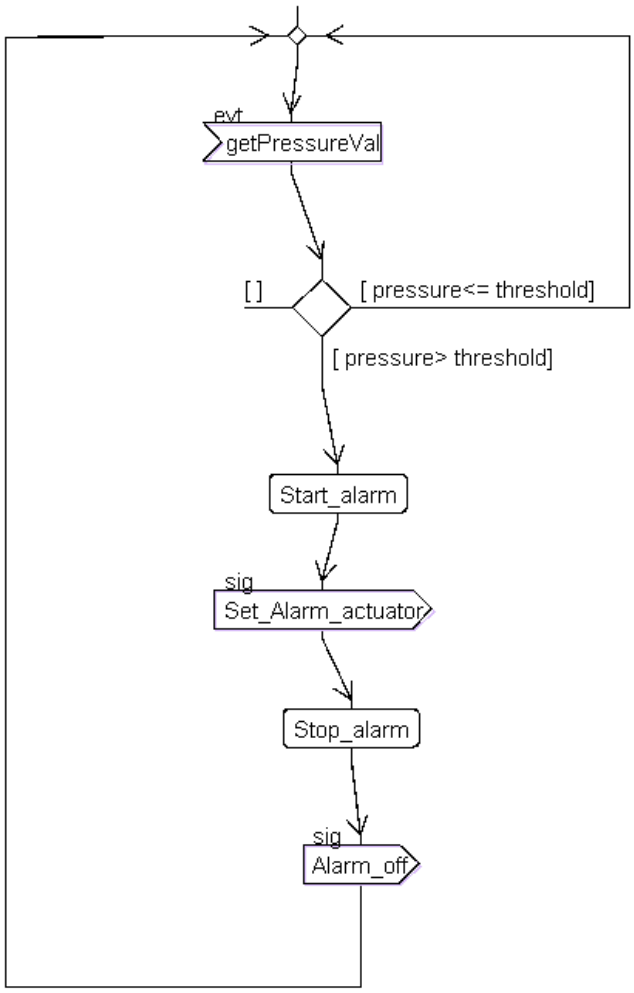
- A single SOC Stm32 microcontroller with a cortex m3 processor will be used to implement this project.

5th: System Analysis:

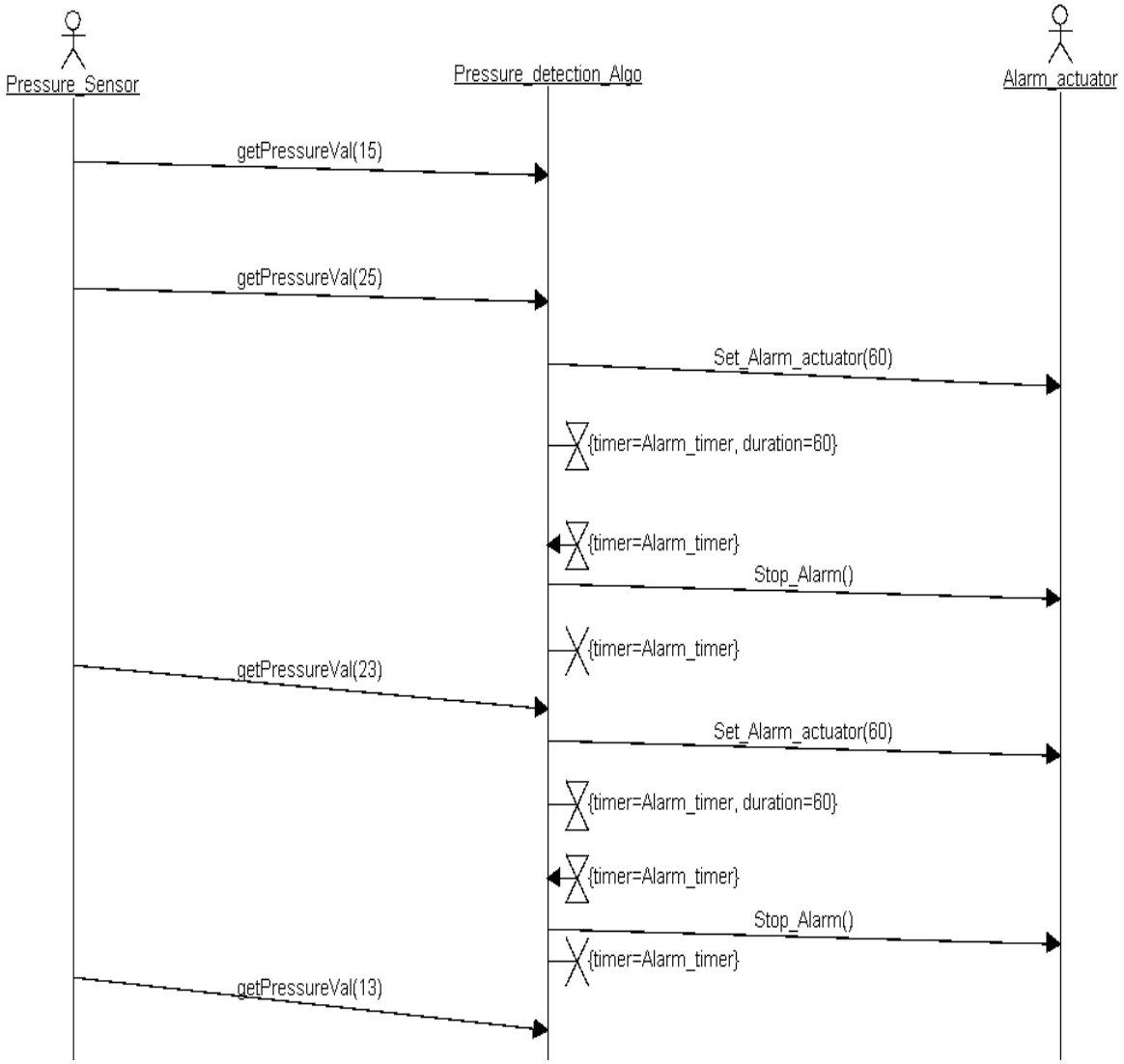
1) Use case diagram:



2) Activity diagram:

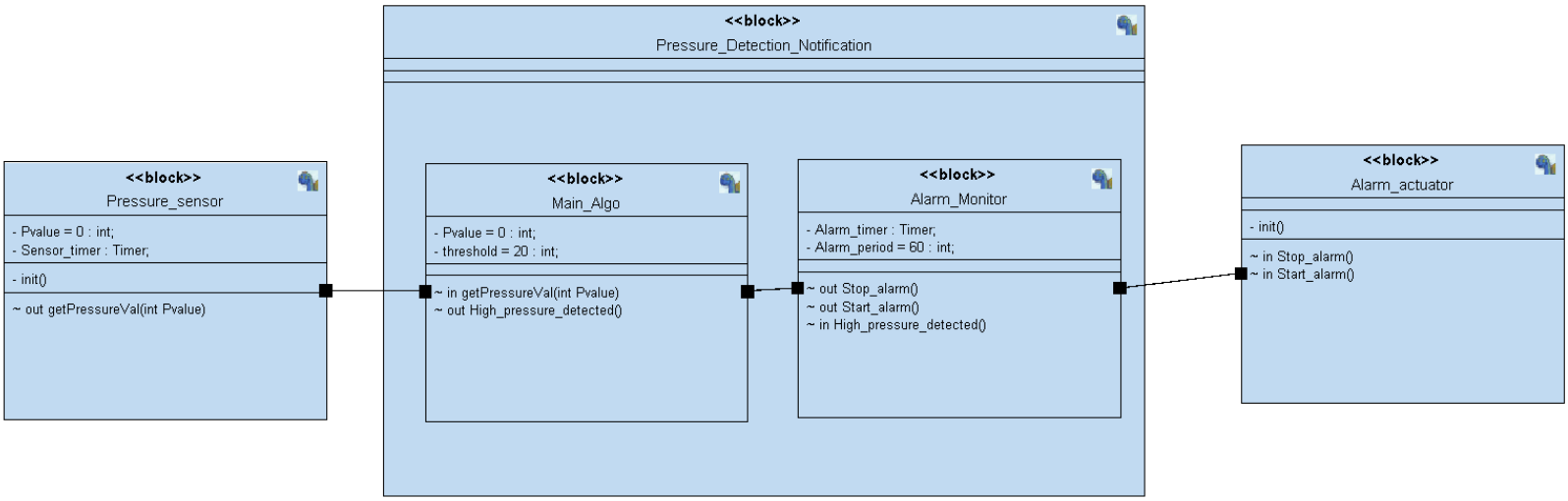


3) Sequence diagram:

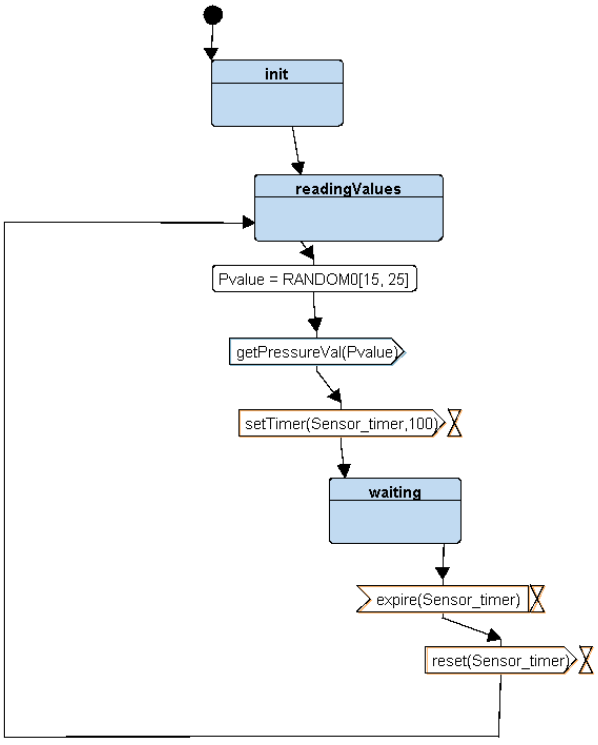


6th : System Design:

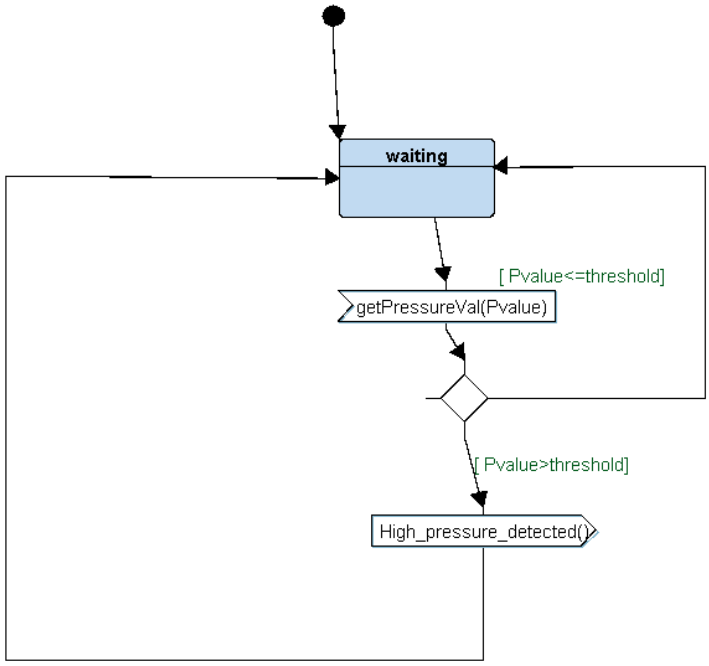
The whole design:



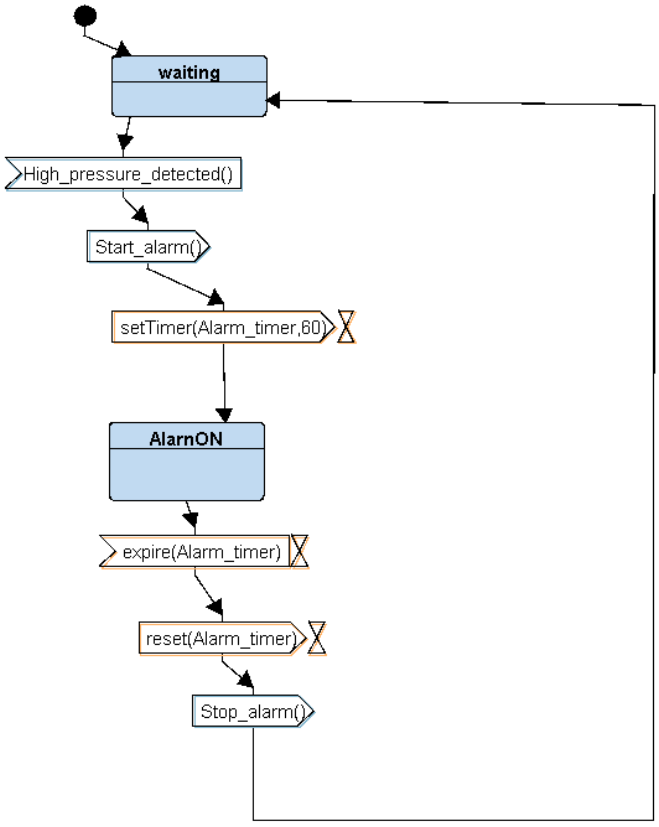
1st) THE PRESSURE SENSOR:



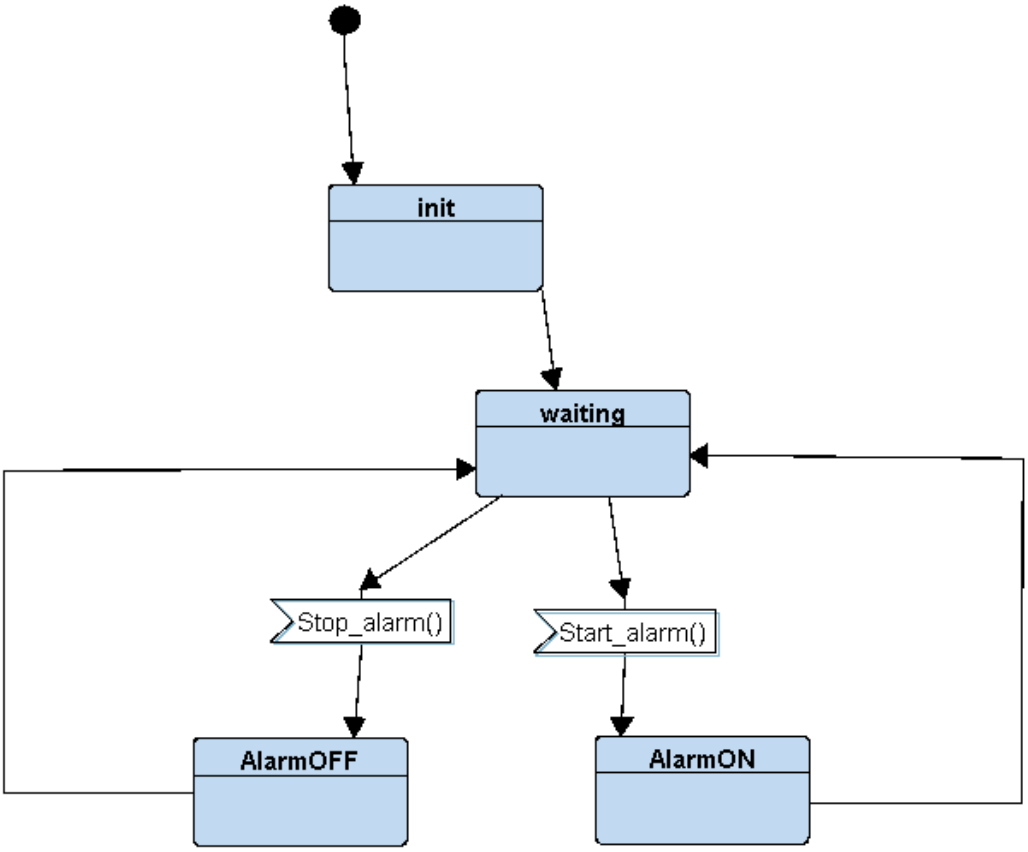
2nd) THE MAIN ALGO:
where comparison between threshold and pressure values is done.



3rd) THE ALARM MONITOR:



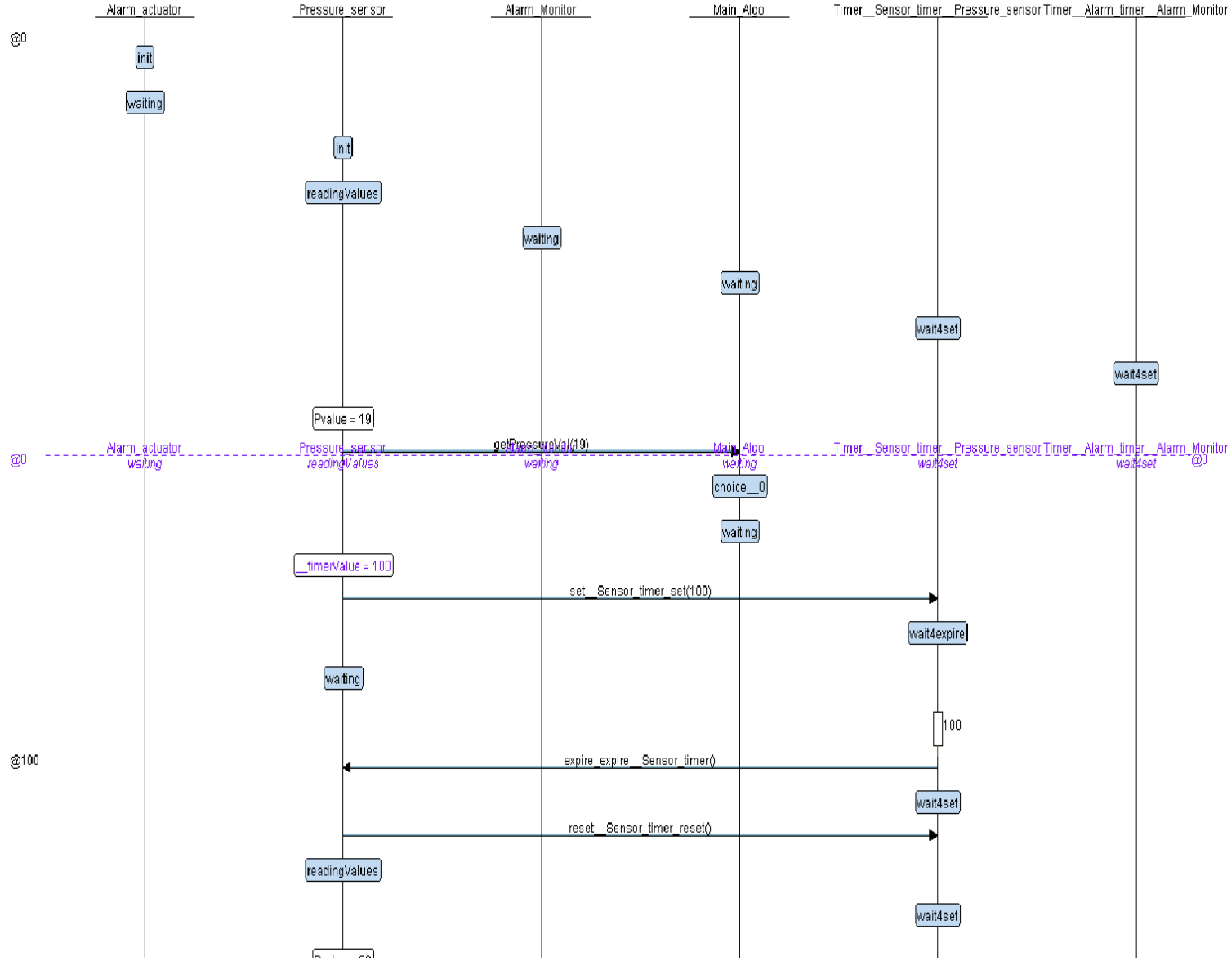
4th) THE ALARM ACTUATOR:



7th)System UML:

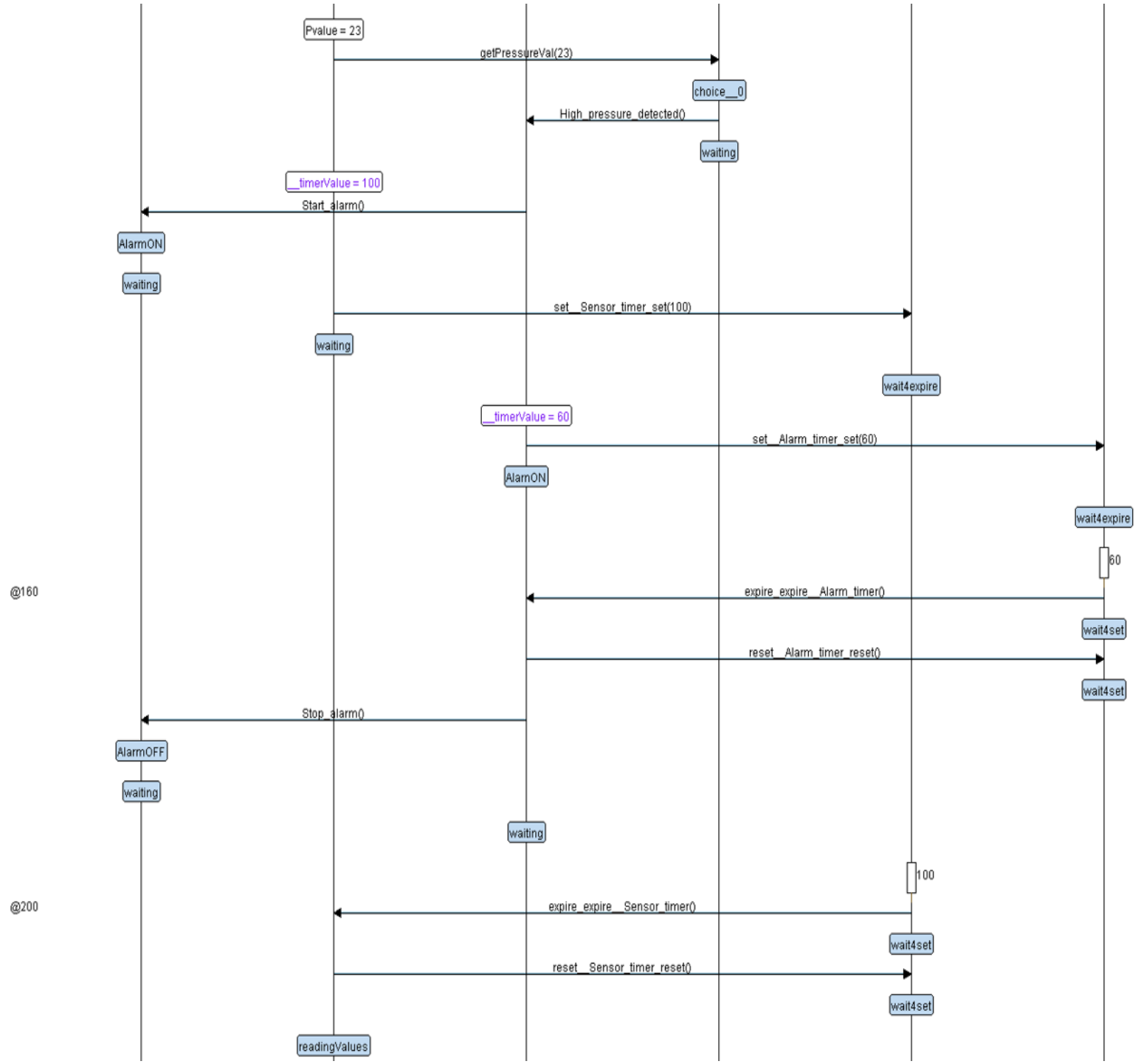
We have 2 main scenarios:

1)The pressure value <= threshold:



Pressure Controlling System

1)The pressure value > threshold:

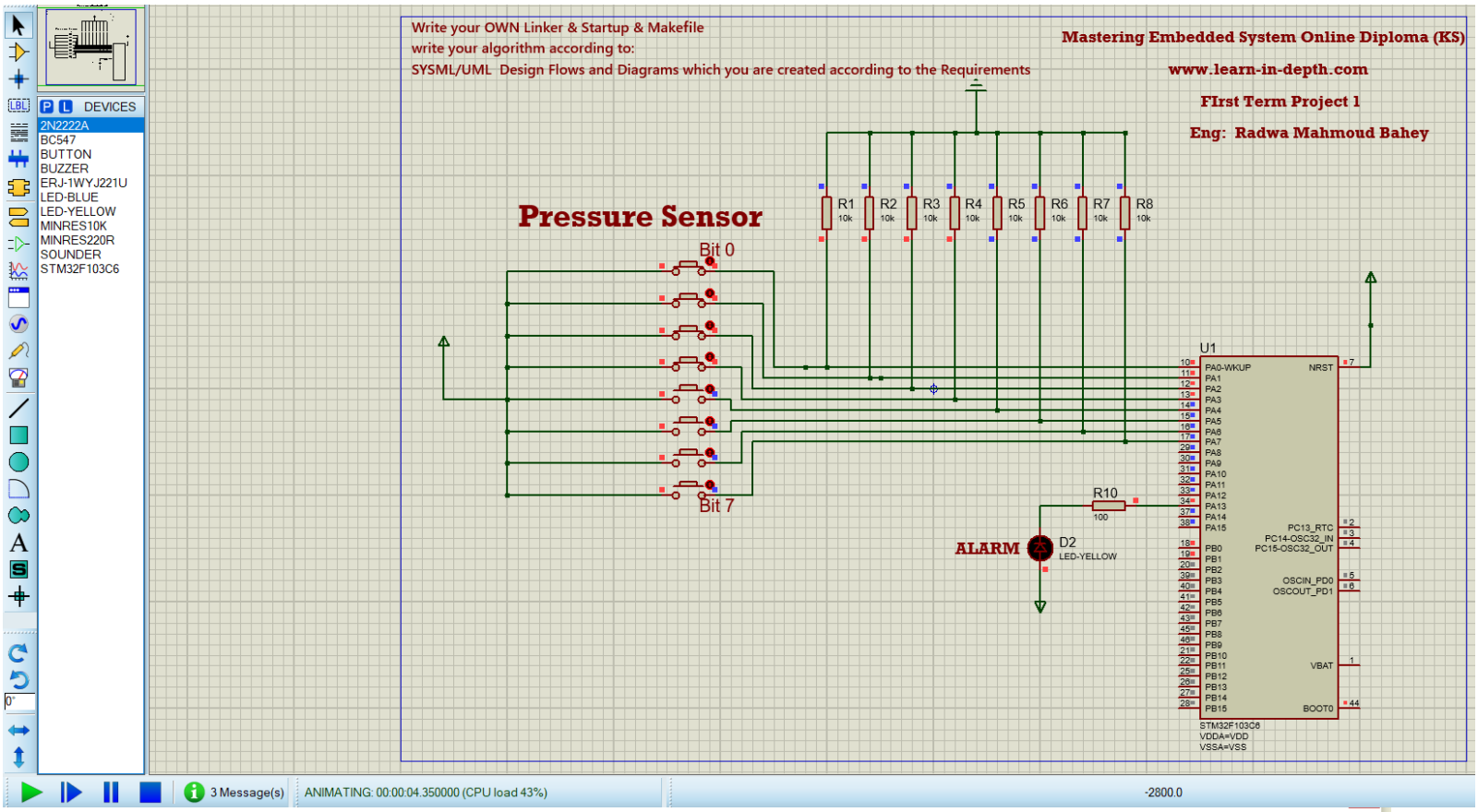


Pressure Controlling System

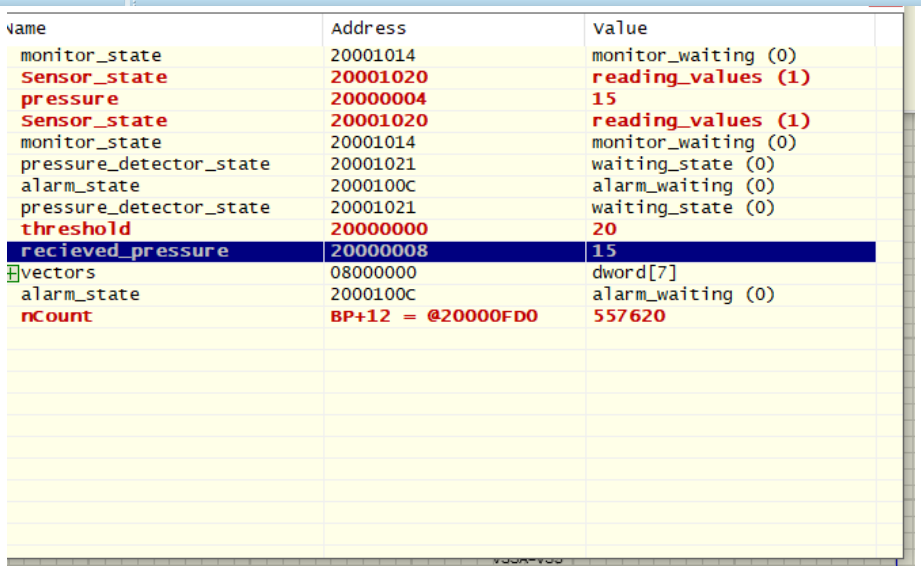
8th)Proteus: proteus video

When received pressure is under the threshold:

alarm's led is off



received pressure=15 {lower than threshold}

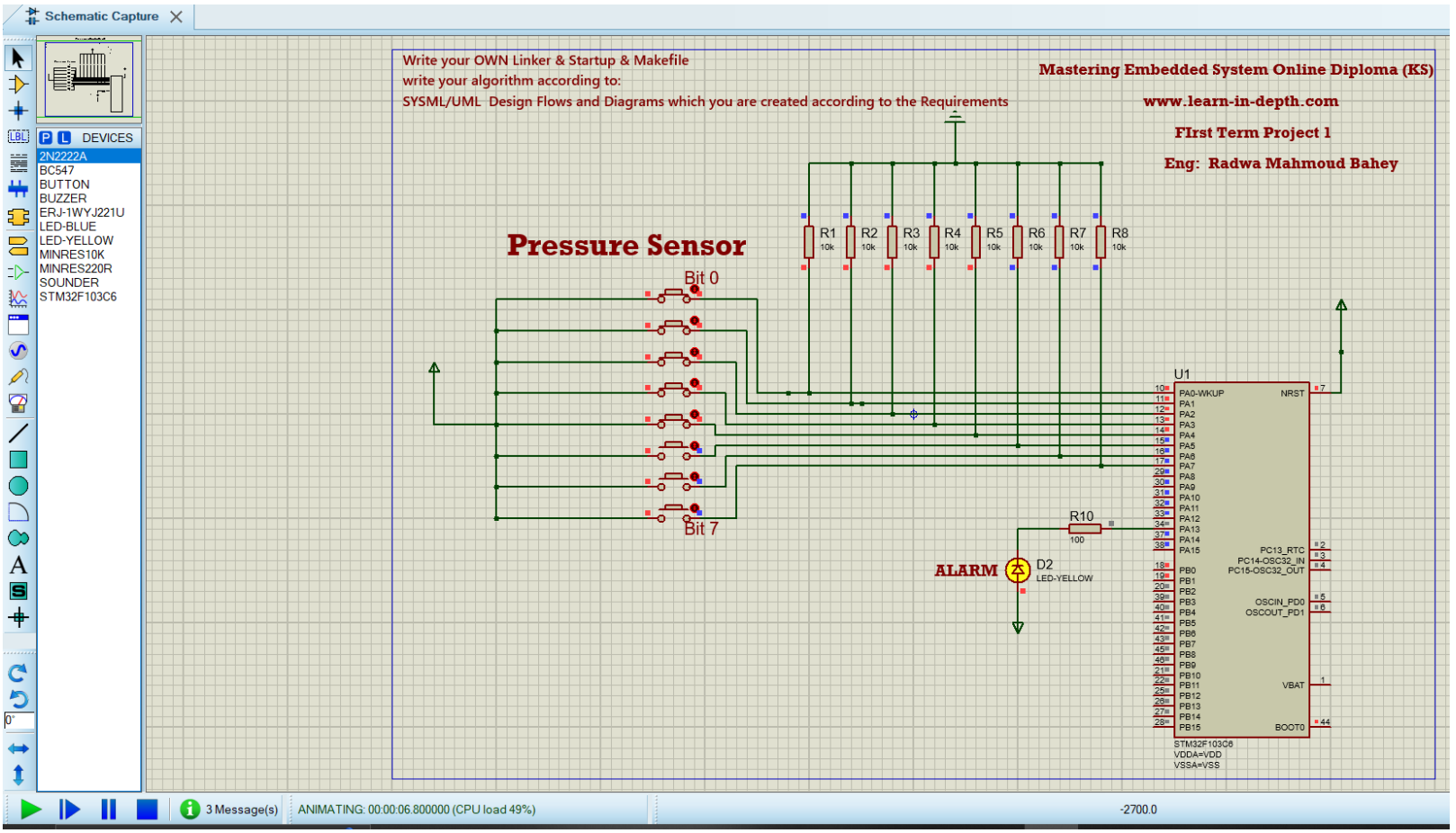


Name	Address	Value
monitor_state	20001014	monitor_waiting (0)
Sensor_state	20001020	reading_values (1)
pressure	20000004	15
Sensor_state	20001020	reading_values (1)
monitor_state	20001014	monitor_waiting (0)
pressure_detector_state	20001021	waiting_state (0)
alarm_state	2000100C	alarm_waiting (0)
pressure_detector_state	20001021	waiting_state (0)
threshold	20000000	20
recieved_pressure	20000008	15
vectors	08000000	dword[7]
alarm_state	2000100C	alarm_waiting (0)
nCount	BP+12 = @20000FD0	557620

Pressure Controlling System

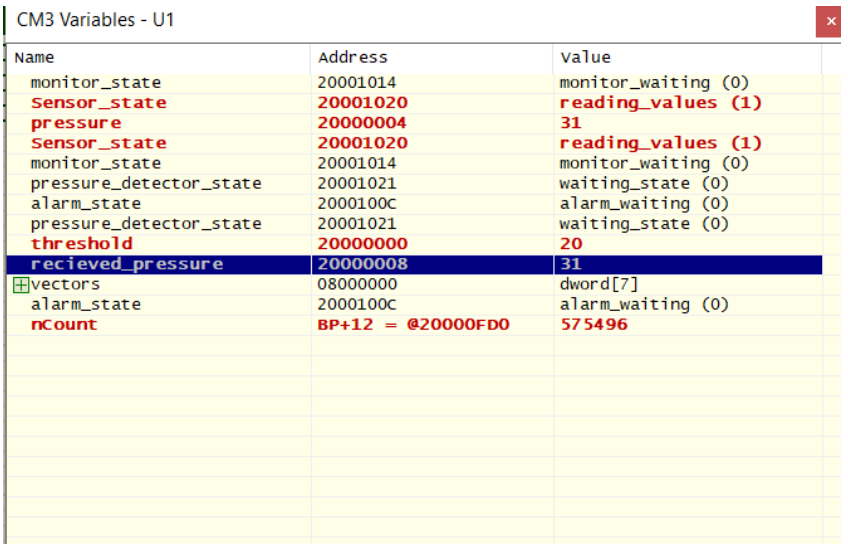
When received pressure is above the threshold:

alarm's led is on



received pressure=31

{higher than threshold}



9th)Symbols:

```
MINGW32:/e/course1/embedded_System_Online_Diploma/Embedded C Project
C Project (master)
$ arm-none-eabi-nm.exe Pressure_detector.elf
20001010 B alarm
0800001c T Alarm_init
2000100c B alarm_state
080003cc W Bus_fault_Handler
080003cc T default_Handler
08000148 T Delay
2000000c B E_BSS
20000004 D E_DATA
08000488 T E_TEXT
0800016c T getPressureVal
080001d4 T GPIO_INITIALIZATION
080003cc W H_fault_Handler
080000dc T high_pressure_detected
0800030c T main
080003cc W MM_fault_Handler
20001018 B monitor
20001014 B monitor_state
080003cc W NMI_Handler
20000004 B pressure
20001024 B pressure_detector
20001021 B pressure_detector_state
20000008 B recieved_pressure
080003d8 T reset_Handler
20000004 B S_BSS
20000000 D S_DATA
2000101c B Sensor
08000254 T Sensor_init
20001020 B Sensor_state
08000184 T Set_Alarm_actuator
08000354 T set_pressure_val
080002c4 T setup
08000110 T st_Alarm_Activated
080000b0 T st_Alarm_OFF
08000084 T st_Alarm_ON
0800006c T st_alarm_waiting
080000f8 T st_monitor_waiting
08000270 T st_reading_values
080003b4 T st_waiting_state
2000100c B stack_top
0800002c T start_alarm
0800004c T stop_alarm
20000000 D threshold
080003cc W Usage_fault_Handler
```

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

Pressure_detector.elf:     file format elf32-littlearm

Sections:
Idx Name          Size      VMA       LMA       File off  Algn
  0 .text         00000488  08000000  08000000  00008000  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
  1 .data         00000004  20000000  08000488  00010000  2**2
                  CONTENTS, ALLOC, LOAD, DATA
  2 .bss          00001024  20000004  0800048c  00010004  2**2
                  ALLOC
  3 .debug_info   00000810  00000000  00000000  00010004  2**0
                  CONTENTS, READONLY, DEBUGGING
  4 .debug_abbrev 00000454  00000000  00000000  00010814  2**0
                  CONTENTS, READONLY, DEBUGGING
  5 .debug_loc    000003cc  00000000  00000000  00010c68  2**0
                  CONTENTS, READONLY, DEBUGGING
  6 .debug_aranges 000000e0  00000000  00000000  00011034  2**0
                  CONTENTS, READONLY, DEBUGGING
  7 .debug_line   00000335  00000000  00000000  00011114  2**0
                  CONTENTS, READONLY, DEBUGGING
  8 .debug_str    0000032a  00000000  00000000  00011449  2**0
                  CONTENTS, READONLY, DEBUGGING
  9 .comment      00000011  00000000  00000000  00011773  2**0
                  CONTENTS, READONLY
 10 .ARM.attributes 00000033  00000000  00000000  00011784  2**0
                  CONTENTS, READONLY
 11 .debug_frame  000002a4  00000000  00000000  000117b8  2**2
                  CONTENTS, READONLY, DEBUGGING
```