Mastering Embedded System Online Diploma

First Term (Final project 1): Pressure Controller

Eng. Mahmoud Magdy Ayoub

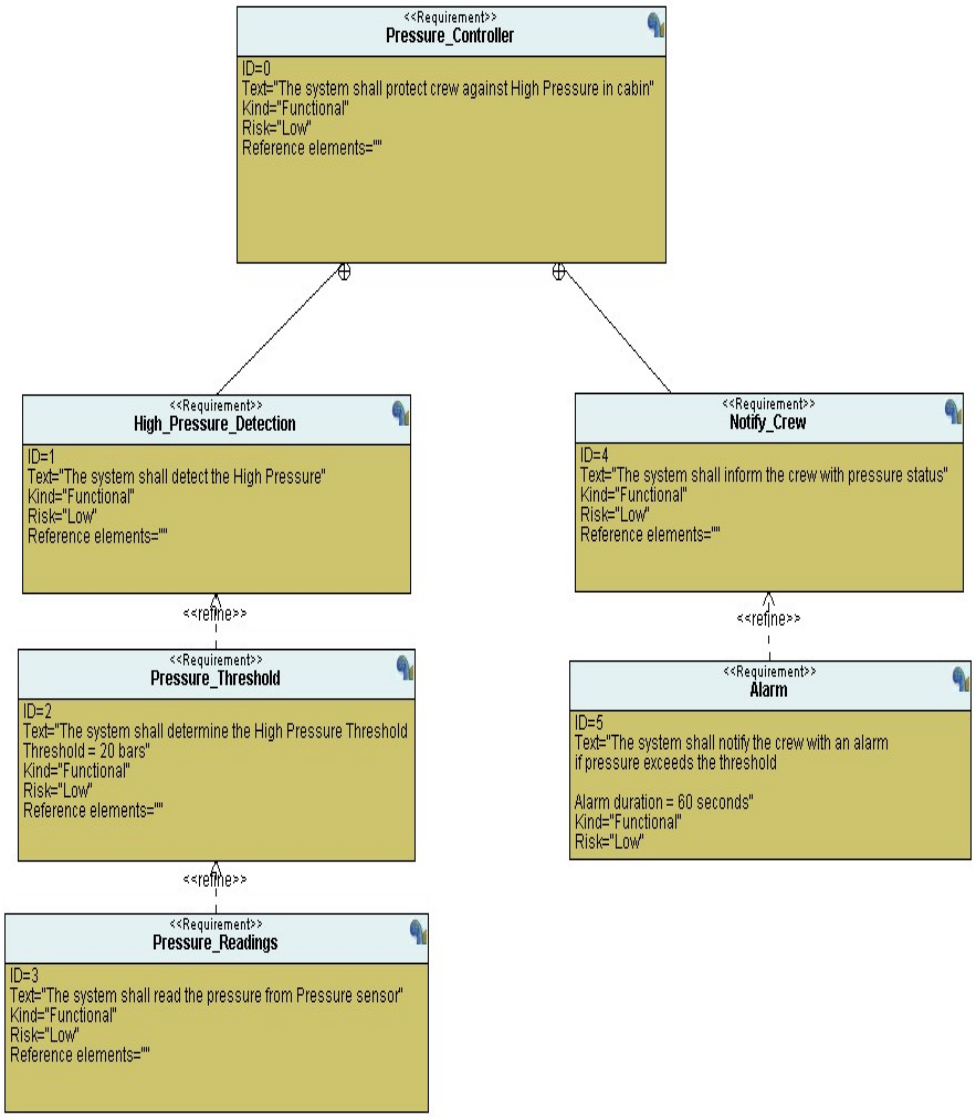
My profile:

https://www.learn-in-depth.com/online-diploma/mahmoudmagdyayoub99@gmail.com

Case Study:

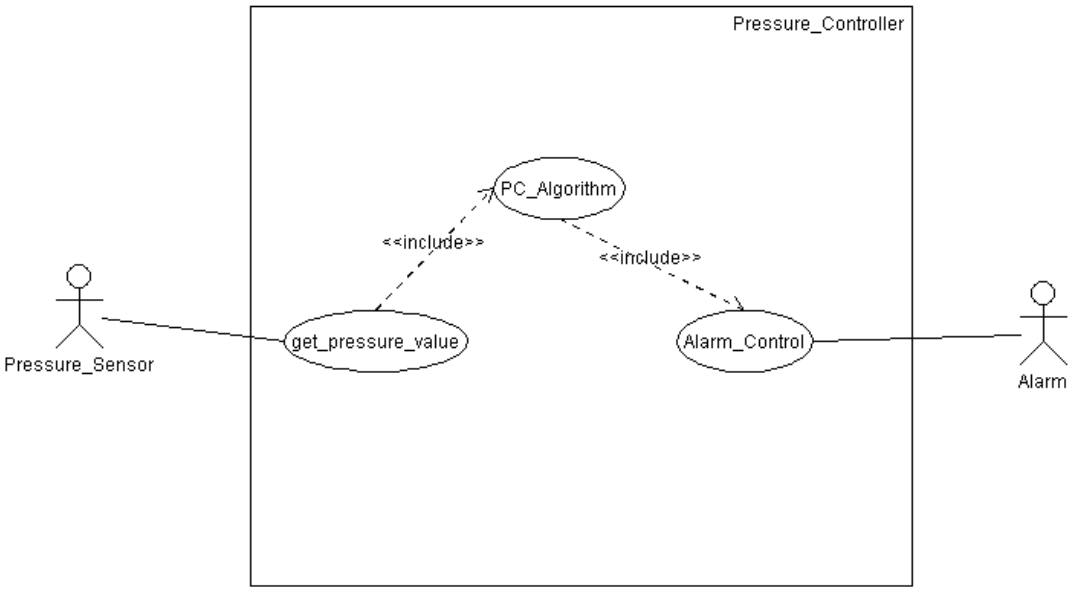
Design a pressure controller system to inform the crew of a cabin with an alarm when the pressure exceeds 20 bars in the cabin. Alarm should last for 60 seconds.

Requirements:

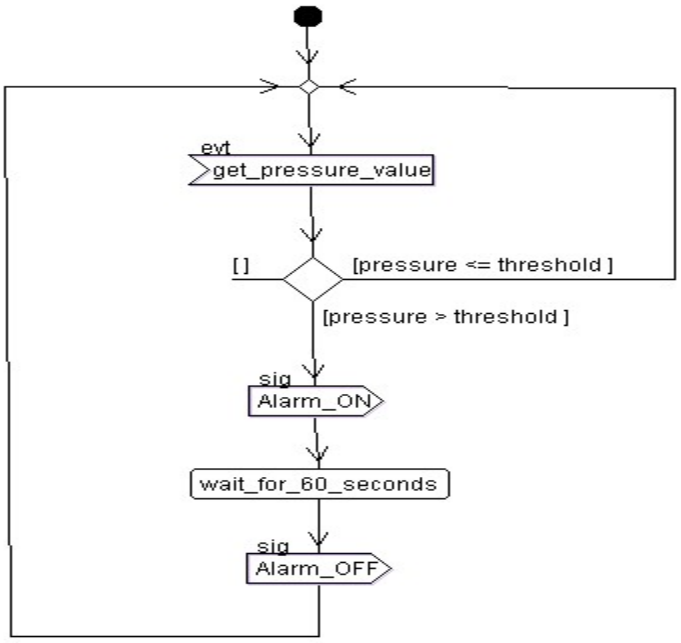


System Analysis:

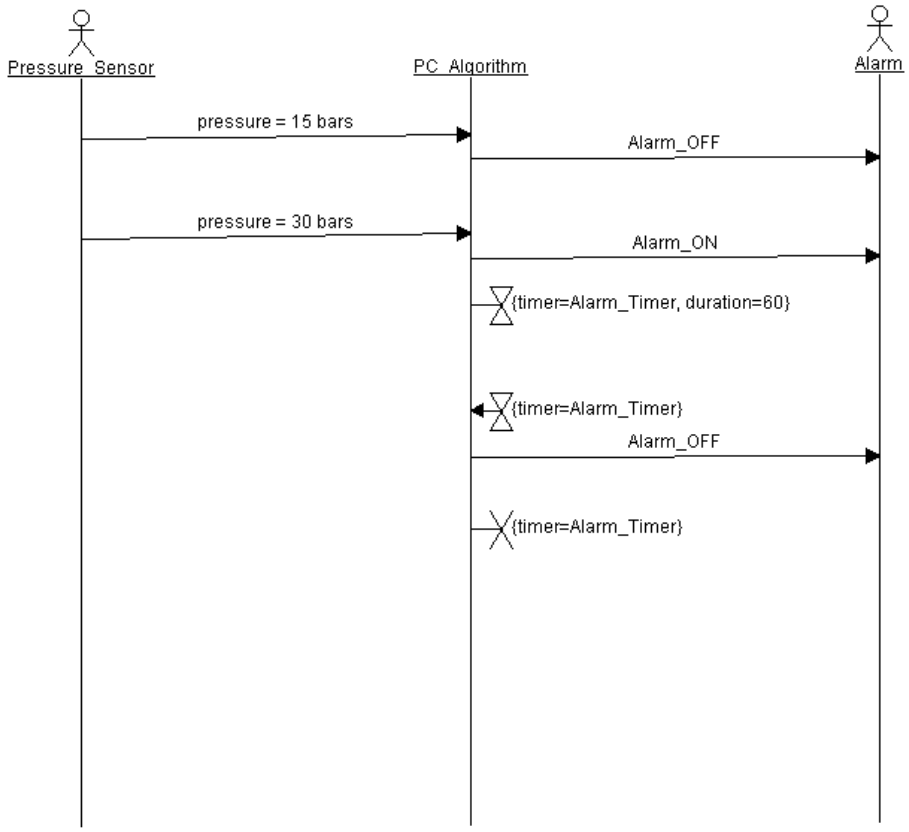
- Use case diagram



- Activity Diagram

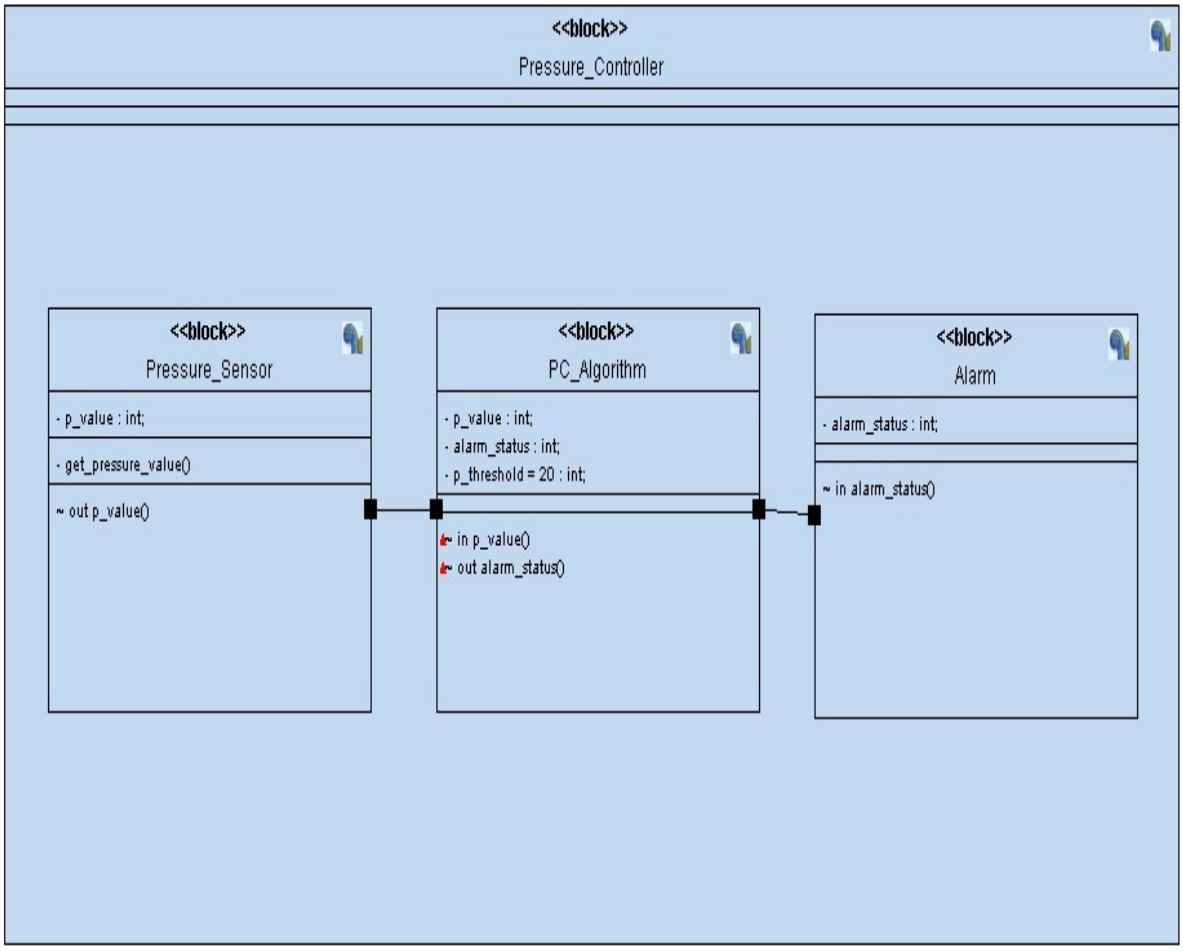


- Sequence Diagram

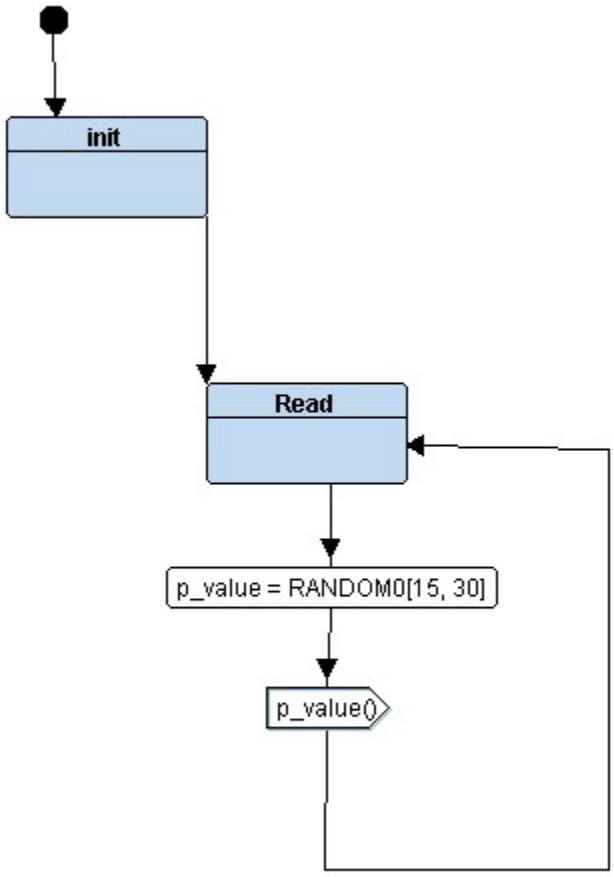


System Design:

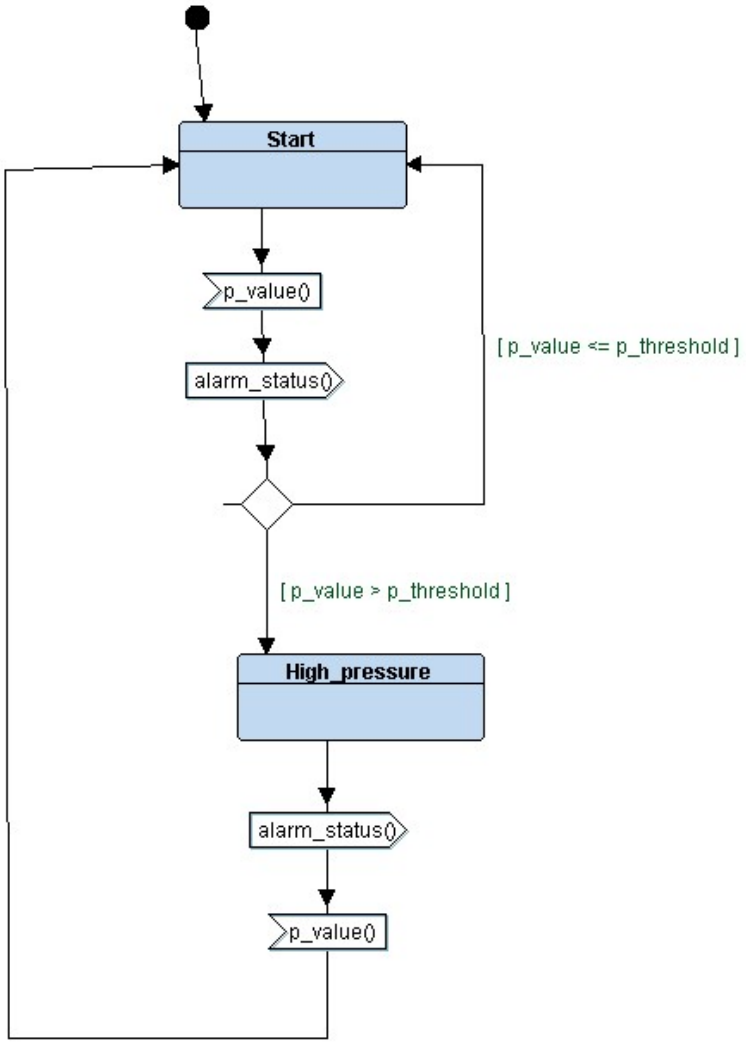
- Block Diagram



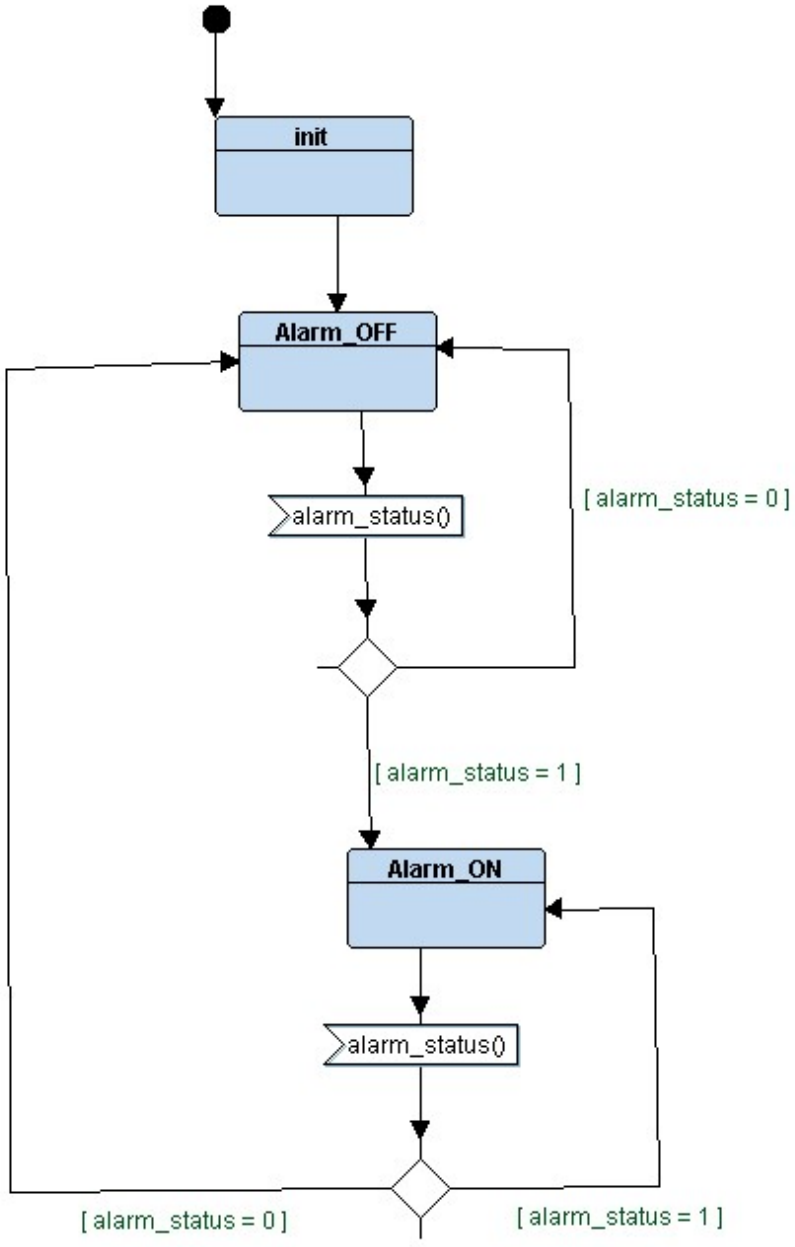
- Pressure sensor state diagram



- PC_Algorithm state diagram

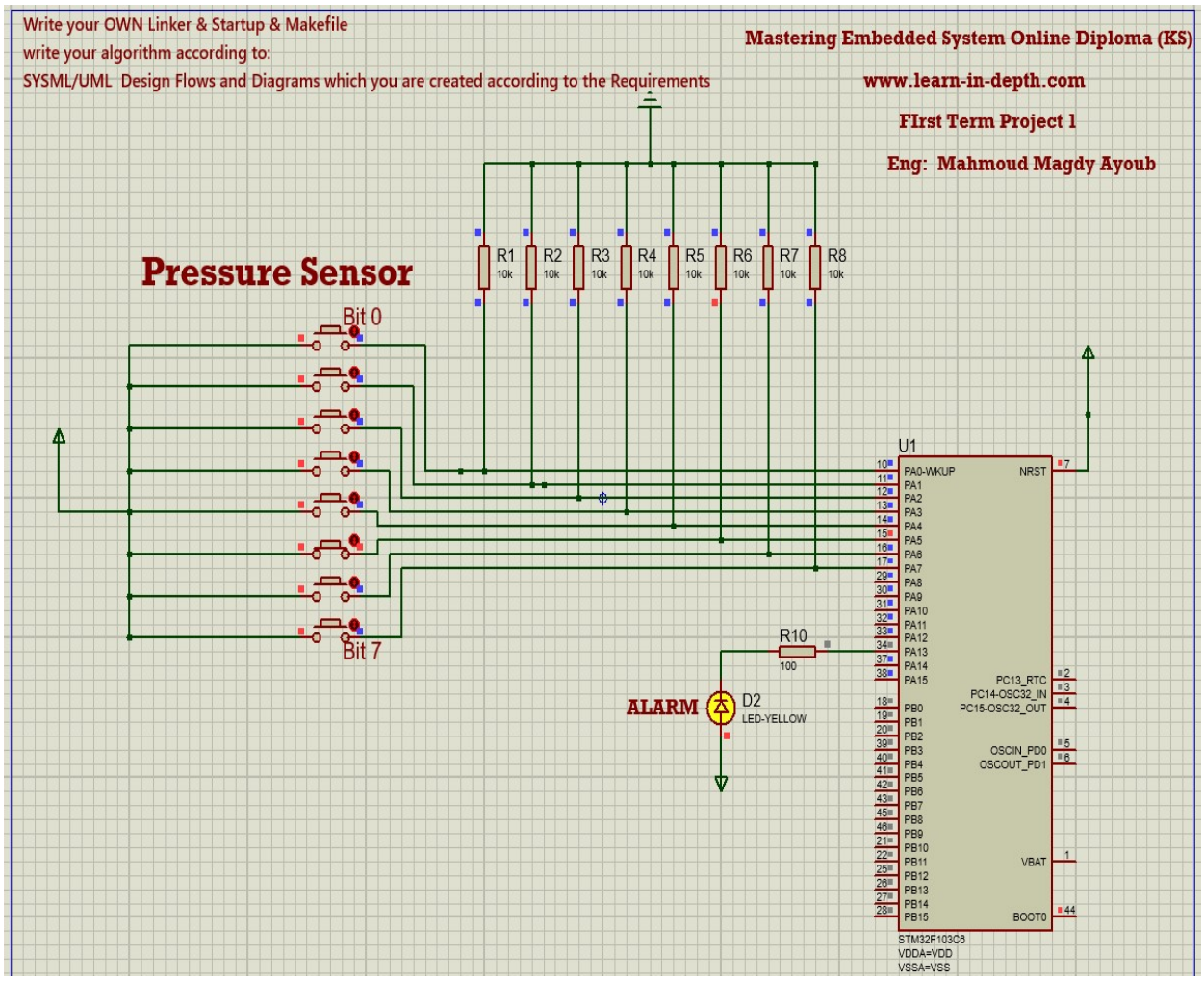


- Alarm state diagram

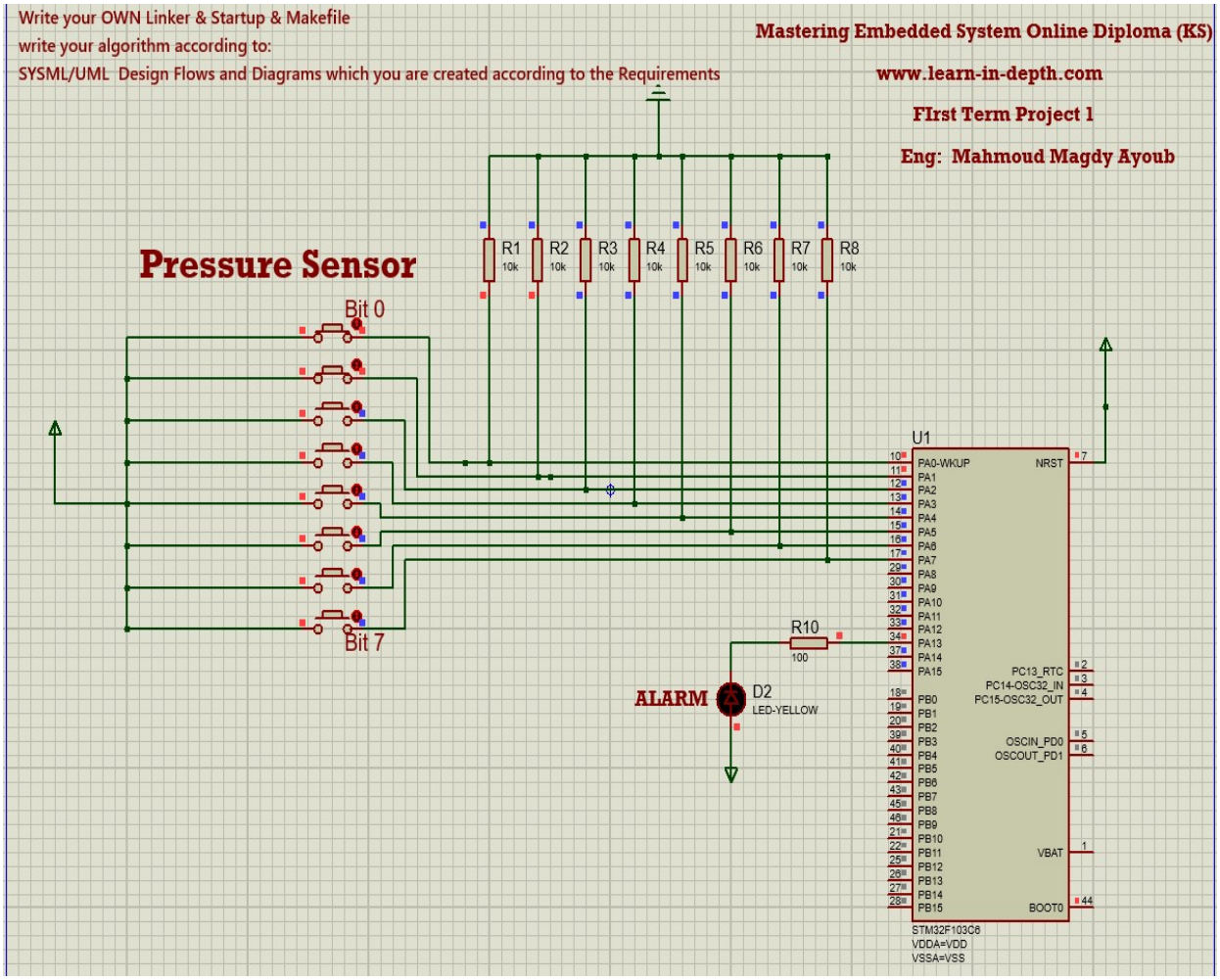


Results:

Alarm_ON



Alarm_OFF


Write your OWN Linker & Startup & Makefile
write your algorithm according to:
SYSML/UML Design Flows and Diagrams which you are created according to the Requirements
Mastering Embedded System Online Diploma (KS)
www.learn-in-depth.com
FIrst Term Project 1
Eng: Mahmoud Magdy Ayoub
Pressure Sensor
Bit 0
Bit 7
R1 10k
R2 10k
R3 10k
R4 10k
R5 10k
R6 10k
R7 10k
R8 10k
R10 100
ALARM
D2 LED-YELLOW
U1
PA0-WKUP
PA1
PA2
PA3
PA4
PA5
PA6
PA7
PA8
PA9
PA10
PA11
PA12
PA13
PA14
PA15
PB0
PB1
PB2
PB3
PB4
PB5
PB6
PB7
PB8
PB9
PB10
PB11
PB12
PB13
PB14
PB15
NRST
PC13_RTC
PC14-OSC32_IN
PC15-OSC32_OUT
OSCIN_PD0
OSCOUT_PD1
VBAT
BOOT0
STM32F103C6
VDDA=VDD
VSSA=VSS

Memory sections:

we only have 8 bytes in .rodata section as we have 2 global constant variables

(pressure_threshold + delay_time)

```
Blu-Ray@DESKTOP-BN637DV MINGW32 /d/Embedded Systems/Learn In Depth Online Diploma/First_Term proje
cts/project1_pressure controller/code
$ 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         00000148  08000000  08000000  00010000  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
  1 .rodata       00000008  08000148  08000148  00010148  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  2 .debug_info   00000403  00000000  00000000  00010150  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  3 .debug_abbrev 00000241  00000000  00000000  00010553  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  4 .debug_loc    000001d0  00000000  00000000  00010794  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  5 .debug_aranges 000000c0  00000000  00000000  00010968  2**3
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  6 .debug_line   000003a0  00000000  00000000  00010a28  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  7 .debug_str    00000205  00000000  00000000  00010dc8  2**0
                  CONTENTS, READONLY, DEBUGGING, OCTETS
  8 .comment      0000004d  00000000  00000000  00010fcd  2**0
                  CONTENTS, READONLY
  9 .ARM.attributes 0000002b  00000000  00000000  0001101a  2**0
                  CONTENTS, READONLY
 10 .debug_frame  00000138  00000000  00000000  00011048  2**2
                  CONTENTS, READONLY, DEBUGGING, OCTETS
```

Symbol Table:

```
Blu-Ray@DESKTOP-BN637DV MINGW32 /d/Embedded Systems/Learn In Depth Online Diploma/First_Term proje
cts/project1_pressure controller/code
$ arm-none-eabi-nm.exe Pressure_Controller.elf
08000140 t _reset
0800011a T Alarm_OFF
080000fc T Alarm_ON
08000054 T Delay
0800014c R delay_time
08000128 T getPressureVal
08000076 T GPIO_INITIALIZATION
080000c8 T main
08000148 R pressure_threshold
08000018 T Set_Alarm_actuator
08000146 t Vector_handler
```