

Pressure Controller

[Learn in depth]

Mastering Embedded System 1st term project

Supervisor : ENG / Keroles Khalil

Represented by : Ibrahim Reda Elmsery

System Architecting /Design sequence

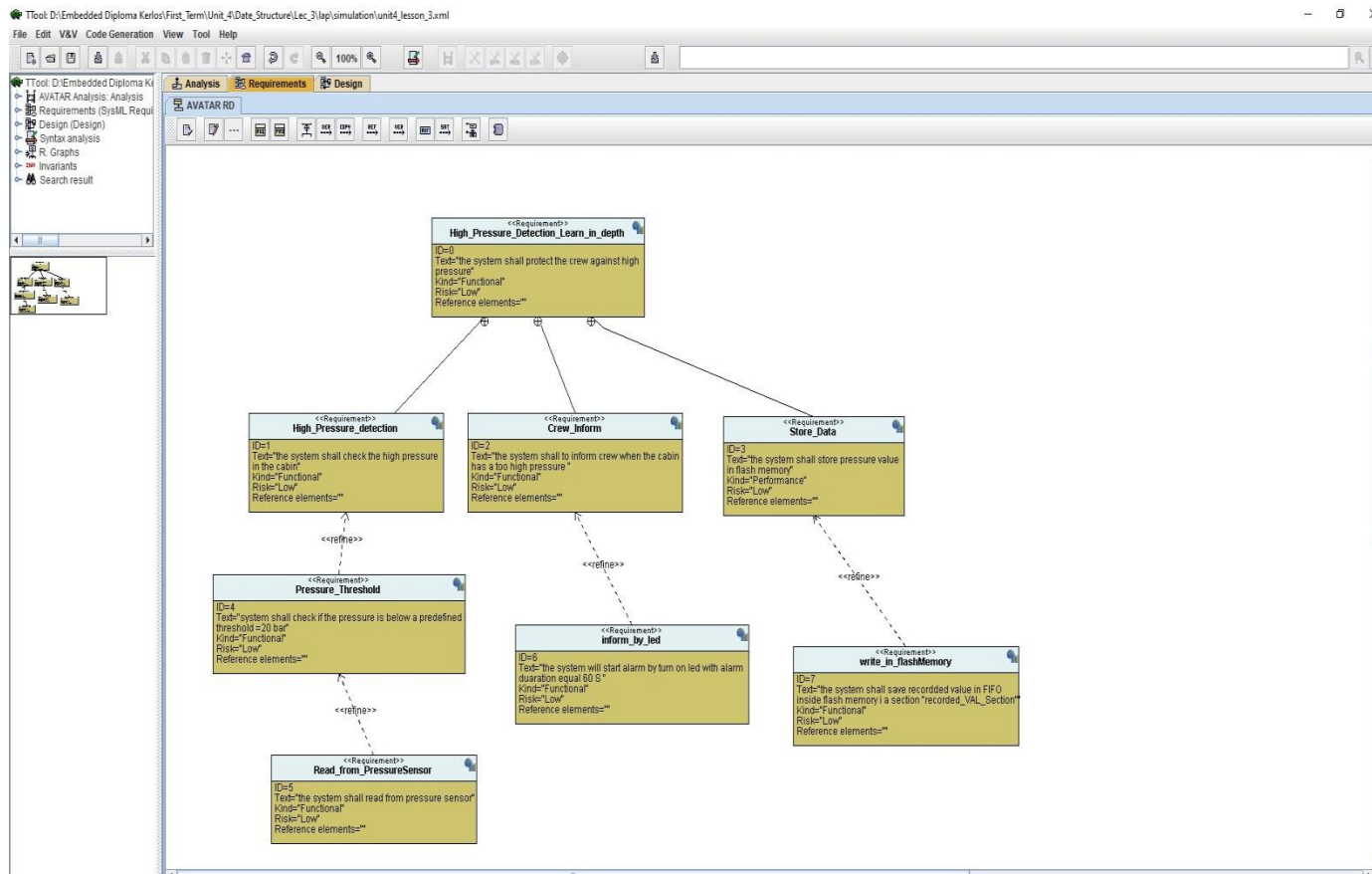
1-Case Study

Pressure controller detect pressure and if pressure higher than 20 bar it send alarm (LED) for 60 seconds and store reading in flash memory(optional) .

2-Method

V –Model

3-Requirement diagram

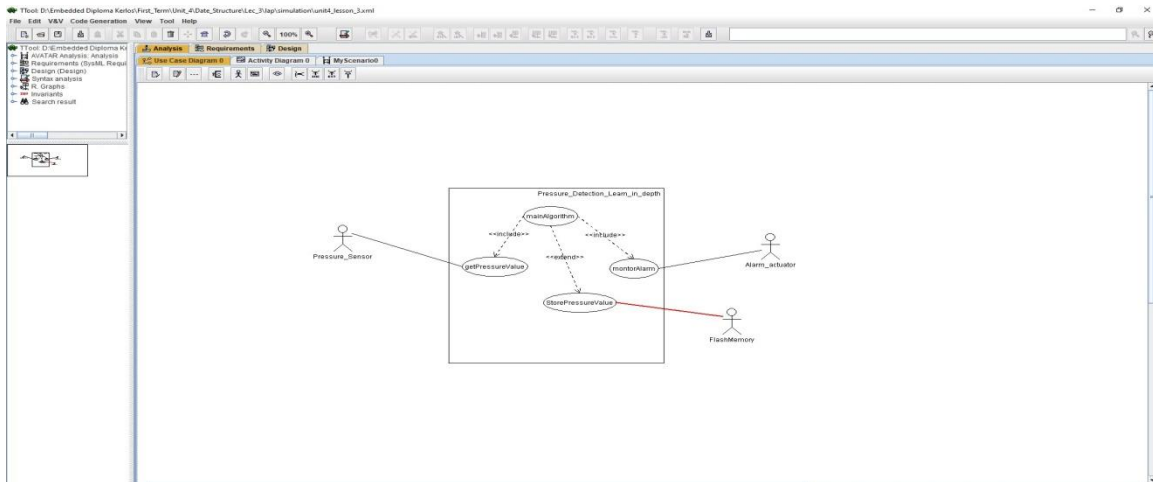


4-Microconroller

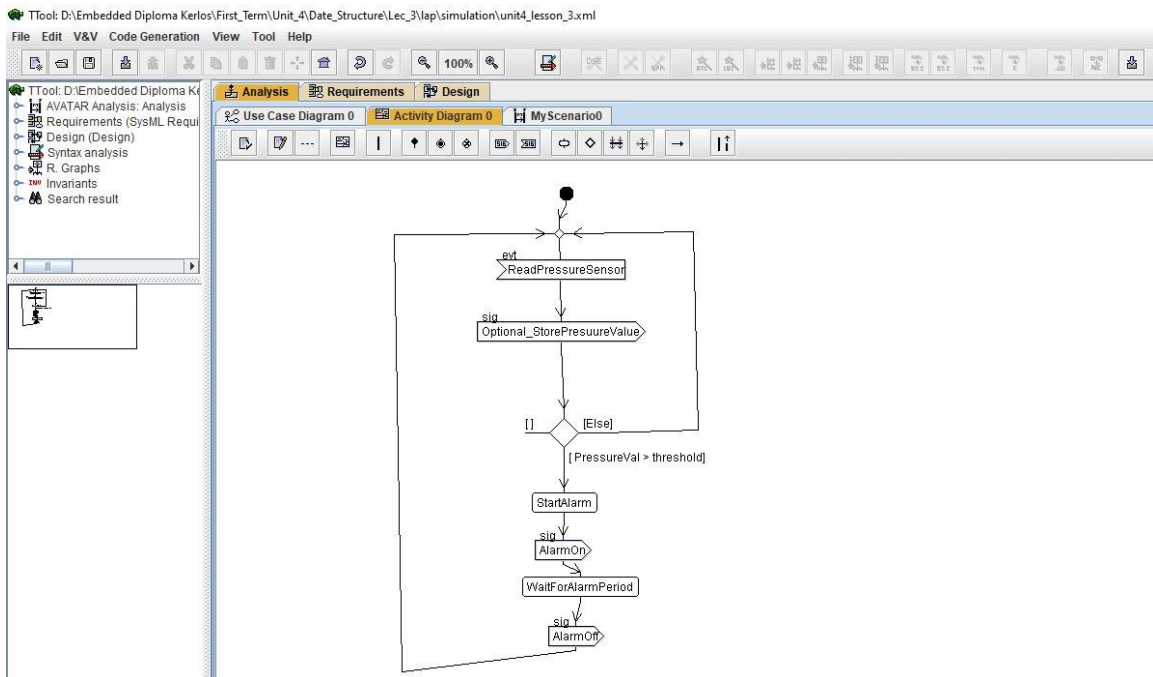
Stm32f103c6t6 (cortex m-3)

5-System analysis

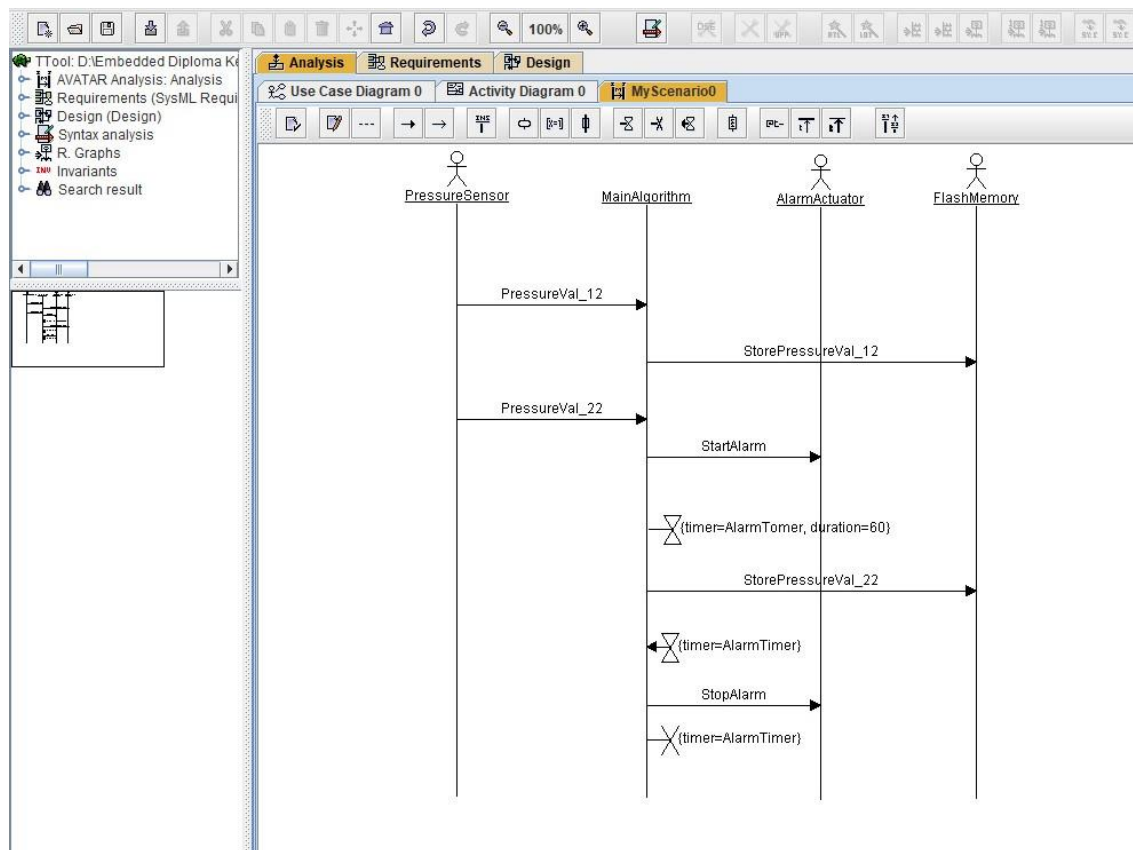
1-case diagram



1-Activity diagram

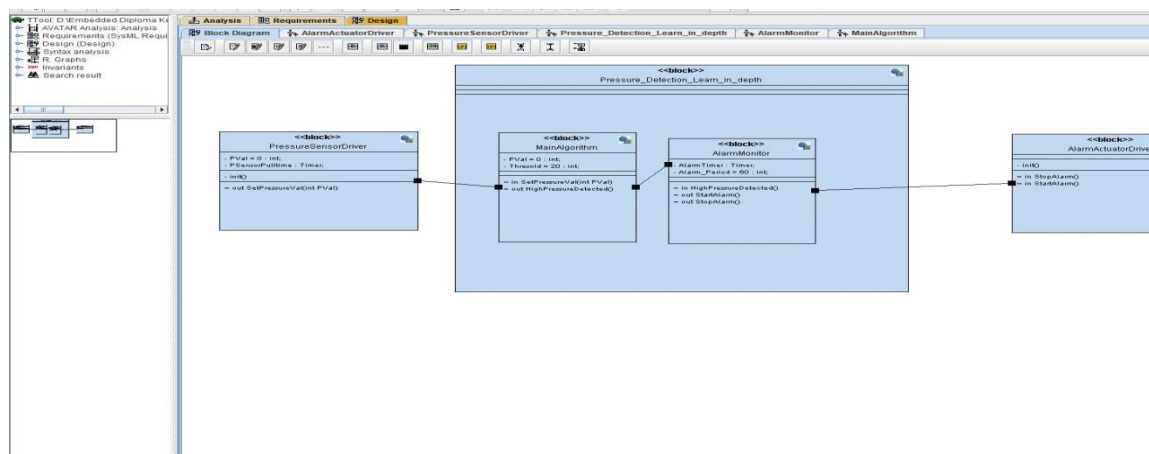


3-Scenario



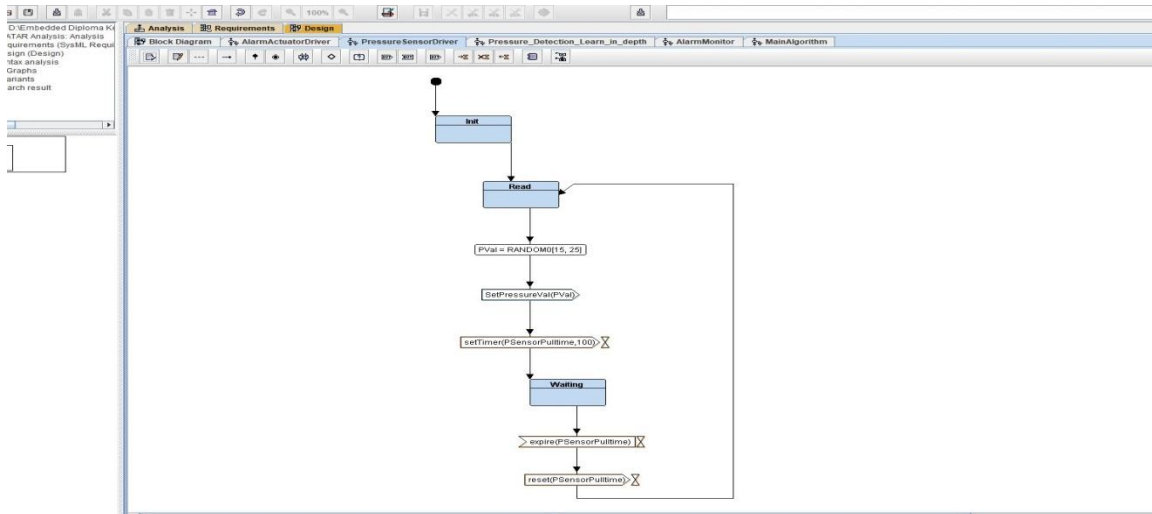
5-System design

1-Project block diagram

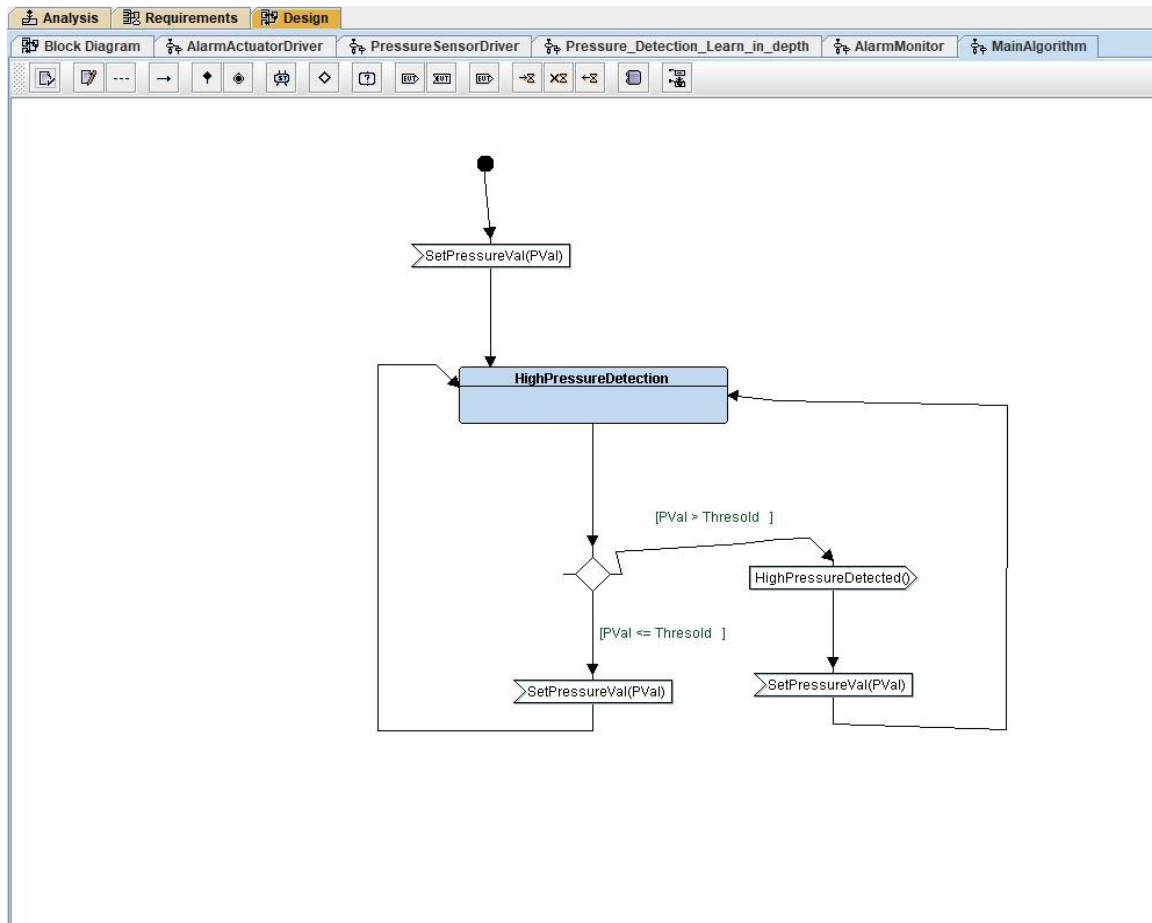


2-State machine for each module in project

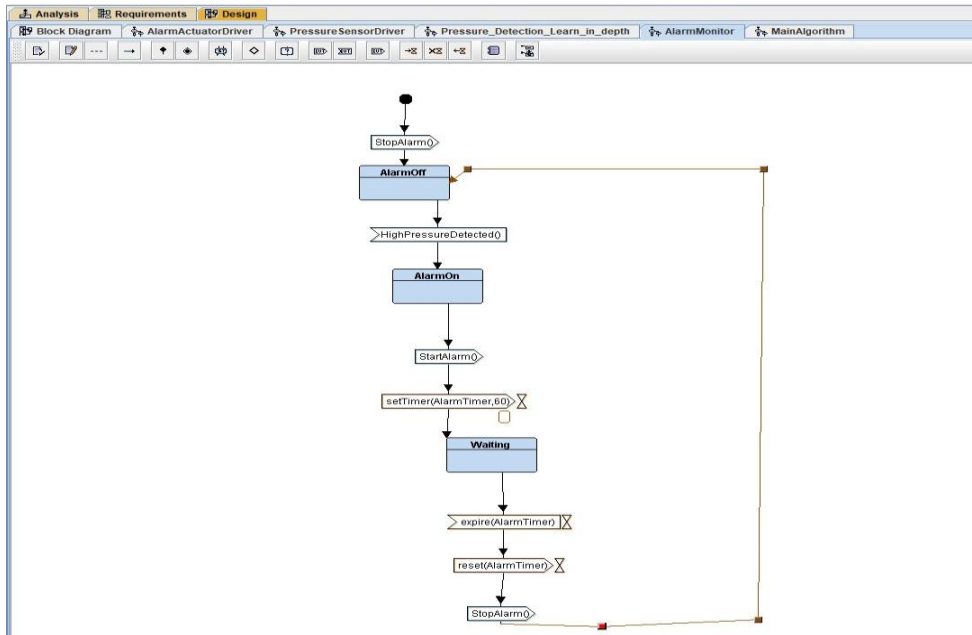
1-Pressure Sensor Driver



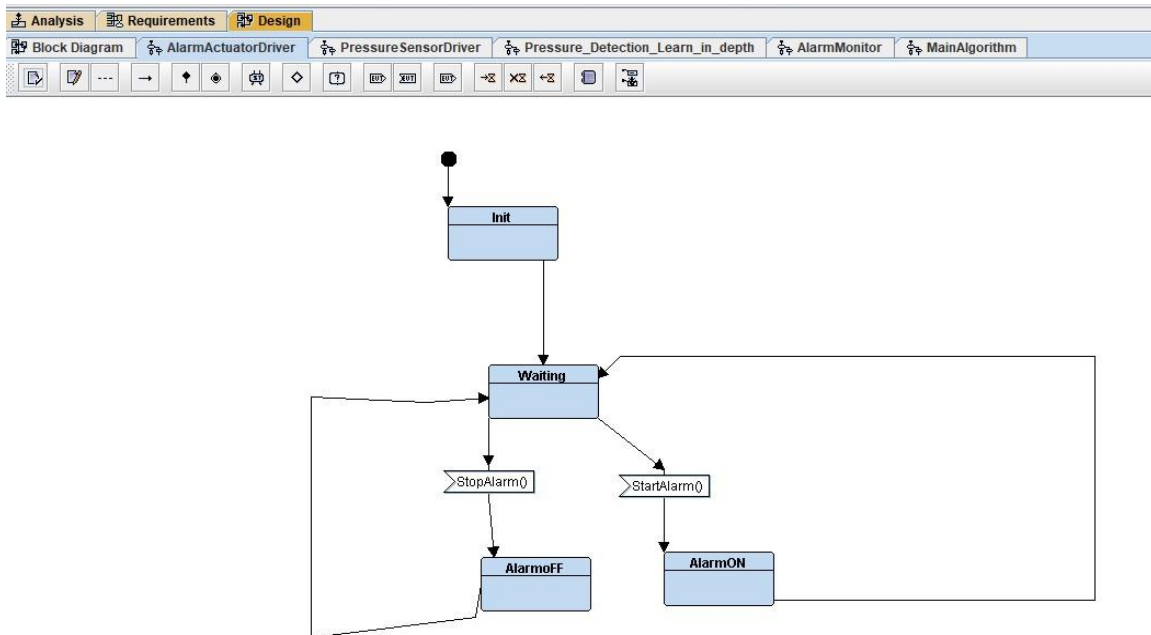
2-MainAlgorithm



3-AlarmMonitor



4-AlarmActuatorDriver



Trace

Tool D:\Embedded\Dioma\KerkoFirst_Term\Unit_4\Date_Structure\Lee_3\lap\simulation\unit4_lesson_3.xml

File Edit View Code Generation View Tool Help

Analysis Requirements Design

Block Diagram AlarmActuatorDriver PressureSensorDriver Pressure_Detection_Learn_in_depth AlarmMonitor MainAlgorithm

```
stateDiagram-v2
    [*] --> StopAlarm
    StopAlarm --> AlarmOff
    AlarmOff --> HighPressureDetected
    HighPressureDetected --> AlarmOn
    AlarmOn --> StartAlarm
    StartAlarm --> setTimerAlarmTimer60
    setTimerAlarmTimer60 --> Waiting
    Waiting --> expireAlarmTimer
    expireAlarmTimer --> resetAlarmTimer
    resetAlarmTimer --> StopAlarm
    StopAlarm --> AlarmOff
```

Interactive simulation

Terminate simulation and quit

Commands

Control Save trace

Simulation information

Status: Stopped Time: 400 Transactions: 188 Coverage: 81.4%

Options

- ☒ Displayed blocks
- ☒ Latencies
- ☒ Randomness
- ☒ Async msg
- ☒ Blocks
- ☒ Variables
- ☒ Transactions
- ☒ Net states

☒ Animate UML diagrams

☐ Show AVATAR IDs on UML diagrams

☐ Show hidden state in sequence diagram

☒ Automatically open active state machine diagram

☒ Trace in sequence diagram # of transactions: 1 Index of last transaction: 0

☒ Automatically execute empty transitions

☒ Automatically enter states

Pending transactions

In Block AlarmMonitor: Transition L_timeValue = 60, ...

```
sequenceDiagram
    participant P as P[Val = 25]
    participant H as HighPressureDetection
    participant C as choice_0
    participant T as _timeValue = 100
    participant S as set
    participant W as Waiting
    participant A as AlarmOn
    participant V as Waiting

    P->>H: GetPressureVal(25)
    H->>C: 
    C->>T: 
    T->>S: 
    S->>W: 
    W->>A: Start(Alarm)
    A->>V: 
    V->>P: 
```

Autosave done in D:\Embedded\Dioma\KerkoFirst_Term\Unit_4\Date_Structure\Lee_3\lap\simulation\unit4_lesson_3.xml

Run simulation for x commands. Works only if the simulator is "ready"



TTTool D:\Embedded Diploma Kerlos\First_Term\Unit_4\Date_Structure\Lec_3\lap\simulation\unit4_lesson_3.xml

File Edit V&V Code Generation View Tool Help

Analysis Requirements Design

Block Diagram AlarmActuatorDriver PressureSensorDriver Pressure_Detection_Learn_in_depth AlarmMonitor MainAlgorithm

```
stateDiagram-v2
    [*] --> StopAlarm0
    StopAlarm0 --> AlarmOff
    AlarmOff --> HighPressureDetected0
    HighPressureDetected0 --> AlarmOn
    AlarmOn --> StartAlarm0
    StartAlarm0 --> setTimerAlarmTimer0
    setTimerAlarmTimer0 --> Waiting
    Waiting --> expireAlarmTimer0
    expireAlarmTimer0 --> resetAlarmTimer0
    resetAlarmTimer0 --> StopAlarm0
    StopAlarm0 --> AlarmOff
```

Interactive simulation

Terminate simulation and quit

Commands

Control Save trace

Nb of steps: 1

Pending transactions

In Block AlarmMonitor: [SYNCHRO] Sending signal StartAlarm

Simulation information

Status: Stopped Time: 400 Transactions: 181 Coverage: 71.4%

Options

- ☐ Displayed blocks
- ☐ Latencies
- ☐ Randomness
- ☐ Asynch. msg
- ☐ Blocks
- ☐ Variables
- ☐ Transactions
- ☐ Met states

- ☒ Animate UML diagrams
- ☐ Show AVATAR IDs on UML diagrams
- ☐ Show hidden state in sequence diagram
- ☒ Automatically open active state machine diagram
- ☒ Trace in sequence diagram # of transactions: 1 Index of last transaction: 0
- ☒ Automatically execute empty transitions
- ☒ Automatically enter states

```
sequenceDiagram
    participant Read
    participant PVal as PVal = 25
    participant SetPressureVal as SetPressureVal(25)
    participant HighPressureDetection
    participant choice_0 as choice_0
    participant HighPressureDetected0 as HighPressureDetected()
    participant AlarmOn
    participant timerValue as timerValue = 100
    participant Waiting
    participant set_PSensorPullTime as set_PSensorPullTime set(100)
    participant reset_PSensorPullTime as reset_PSensorPullTime reset()

    Read->>PVal: Read
    PVal->>SetPressureVal: SetPressureVal(25)
    SetPressureVal->>HighPressureDetection: HighPressureDetection
    HighPressureDetection->>choice_0: choice_0
    choice_0->>HighPressureDetected0: HighPressureDetected()
    HighPressureDetected0->>AlarmOn: AlarmOn
    AlarmOn->>timerValue: timerValue = 100
    timerValue->>Waiting: Waiting
    Waiting->>set_PSensorPullTime: set_PSensorPullTime set(100)
    set_PSensorPullTime->>reset_PSensorPullTime: reset_PSensorPullTime reset()
    reset_PSensorPullTime->>Read: Read
```

Autosave done in D:\Embedded Diploma Kerlos\First_Term\Unit_4\Date_Structure\Lec_3\lap\simulation\unit4_lesson_3.xml-

Run simulation for x commands. Works only if the simulator is "ready"

26°C صافي غالية ENG 12:18

TTool D:\Embedded Diploma Kertos\First_Term\Unit_4\Date_Structure\Lec_3\lap\simulation\unit4_lesson_3.xml

File Edit V&V Code Generation View Tool Help

Analysis Requirements Design

Block Diagram AlarmActuatorDriver PressureSensorDriver Pressure_Detection_Learn_in_depth AlarmMonitor MainAlgorithm

```
stateDiagram-v2
    [*] --> Init
    Init --> Read
    Read --> Read
    Read --> PVal : PVal = RANDOMQ[15, 25]
    PVal --> SetPressureVal : SetPressureVal(PVal)
    SetPressureVal --> SetTimer : setTimer(PSensorPulltime, 100)
    SetTimer --> Waiting
    Waiting --> Waiting
    Waiting --> Expire : expire(PSensorPulltime)
    Expire --> Reset : reset(PSensorPulltime)
    Reset --> Read
```

Interactive simulation

Terminate simulation and quit

Commands

Control Save trace

Nb of steps: 1

Pending transactions

- in Block PressureSensorDriver: Transition timerValue =
- in Block AlarmMonitor: [SYNCHRO] Sending signal StartAlarm

Simulation information

Status: Stopped Time: 400 Transactions: 173 Coverage: 71.4%

Displayed blocks Latencies Randomness Asynch. msg

Options Blocks Variables Transactions Met states

- ☒ Animate UML diagrams
- ☐ Show AVATAR IDs on UML diagrams
- ☐ Show hidden state in sequence diagram
- ☒ Automatically open active state machine diagram
- ☒ Trace in sequence diagram # of transactions: 1 Index of last transaction: 0
- ☒ Automatically execute empty transitions
- ☒ Automatically enter states

```
sequenceDiagram
    participant Read
    participant PVal as PVal = 25
    participant SetPressureVal as SetPressureVal(25)
    participant HighPressureDetection as HighPressureDetection
    participant choice_0 as choice_0
    participant AlarmOn as AlarmOn

    Read->>PVal: PVal = 25
    PVal->>SetPressureVal: SetPressureVal(25)
    SetPressureVal->>HighPressureDetection: HighPressureDetection
    HighPressureDetection->>choice_0: choice_0
    choice_0->>AlarmOn: HighPressureDetected()
    AlarmOn-->>Read: 
```

Autosave done in D:\Embedded Diploma Kertos\First_Term\Unit_4\Date_Structure\Lec_3\lap\simulation\unit4_lesson_3.xml-

Run simulation for x commands. Works only if the simulator is "ready"

26°C صافي غاليه ENG 12:10 ١٤٣٧/٩/١٣

Tool: D:\Embedded Diploma\Kerlos\First_Term\Unit_4\Date_Structure\Lec_3\lap\simulation\unit4_lesson_3.xml

File Edit V&V Code Generation View Tool Help

Analysis Requirements Design

Block Diagram AlarmActuatorDriver PressureSensorDriver Pressure_Detection_Learn_in_depth AlarmMonitor MainAlgorithm

Init

Read

PVal = RANDOM() [15, 25]

SetPressure(PVal)

setTimer(PSensorPullTime, 100)

Waiting

expire(PSensorPullTime)

reset(PSensorPullTime)

Interactive simulation

Terminate simulation and quit

Commands

Control Save trace

No of steps: 1

Pending transactions

in Block PressureSensorDriver: [SYNCHRO] Sending signal

Simulation information

Status: Stopped Time: 400 Transactions: 159 Coverage: 61.4%

Displayed blocks Latencies Randomness Asynch. msg

Options

Blocks Variables Transactions Met states

☒ Animate UML diagrams

☐ Show AVATAR IDs on UML diagrams

☐ Show hidden state in sequence diagram

☒ Automatically open active state machine diagram

☒ Trace in sequence diagram # of transactions: 1 Index of last transaction: 0

☒ Automatically execute empty transitions

☒ Automatically enter states

choice_0

timer value = 100

Waiting

set PSensorPullTime_set(100)

expire expire PSensorPullTime()

reset PSensorPullTime_reset()

Read

PVal = 25

Autosave done in D:\Embedded Diploma\Kerlos\First_Term\Unit_4\Date_Structure\Lec_3\lap\simulation\unit4_lesson_3.xml-

Run simulation for x commands. Works only if the simulator is "ready"

12:18 ص 26°C صافي عالية ٨٠ ٩١ ENG ٢٠٢٢/٩/١٣

UML Modeler Interface showing a state machine diagram and its simulation.

State Machine Diagram:

- Starts at an initial state.
- Transitions to **Init** state.
- From **Init**, transitions to **Read** state.
- From **Read**, transitions to **PVal = RANDOM(15, 25)** activity.
- From **PVal = RANDOM(15, 25)**, transitions to **SetPressure(PVal)** activity.
- From **SetPressure(PVal)**, transitions to **setTimer(PSensorPullTime, 100)** activity.
- From **setTimer(PSensorPullTime, 100)**, transitions to **Waiting** state.
- From **Waiting**, transitions to **expire(PSensorPullTime)** activity.
- From **expire(PSensorPullTime)**, transitions to **reset(PSensorPullTime)** activity.
- From **reset(PSensorPullTime)**, transitions back to **Read** state.

Simulation Information:

- Status: **Stopped**
- Time: 0
- Transactions: 39
- Coverage: 45.7%

Simulation Log:

- Read
- StopAlarm()
- PVal = 18
- PressureSensorDriver: PVal = 18
- MainAlgorithm: PVal = 18
- AlarmMonitor: AlarmOn
- choice_0
- timerValue = 100

Simulation Controls:

- Terminate simulation and quit
- Commands: Control, Save trace
- Simulation information: Options, Blocks, Variables, Transactions, Met states
- Simulation options: ☒ Animate UML diagrams, ☐ Show AVATAR IDs on UML diagrams, ☐ Show hidden state in sequence diagram, ☒ Automatically open active state machine diagram, ☒ Trace in sequence diagram # of transactions: 1 Index of last transaction: 0, ☒ Automatically execute empty transitions, ☒ Automatically enter states

Autosave done in D:\Embedded Diploma Keriol\First_Term\Unit_4\Date_Structure\Lec_3\lap\simulation\unit4_lesson_3.xml

UML Modeler Interface

File Edit View Code Generation View Tool Help

Analysis Requirements Design

Block Diagram AlarmActuatorDriver PressureSensorDriver Pressure_Detection_Learn_in_depth AlarmMonitor MainAlgorithm

Init

Read

PVal = RANDOMQ(15, 25)

SetPressureVal(PVal)

setTimer(PSensorFulltime, 100)

Waiting

expire(PSensorFulltime)

reset(PSensorFulltime)

Interactive simulation

Terminate simulation and quit

Commands

Control Save trace

No of steps: 1

Pending transactions

in Block PressureSensorDriver: Transition _timerValue =

Simulation information

Status: Stopped Time: 0 Transactions: 38 Coverage: 44.2%

Displayed blocks Latencies Randomness Asynch. msg

Options Blocks Variables Transactions Met states

☒ Animate UML diagrams

☐ Show AVATAR IDs on UML diagrams

☐ Show hidden state in sequence diagram

☒ Automatically open active state machine diagram

☒ Trace in sequence diagram # of transactions: 1 Index of last transaction: 0

☒ Automatically execute empty transitions

☒ Automatically enter states

StopAlarm()

PVal = 18

SetPressureVal(18)

HighPressureDetection

choice_0

AlarmOn

Autosave done in D:\Embedded Diploma Keriol\First_Term\Unit_4\Date_Structure\Lec_3\lap\simulation\unit4_lesson_3.xml-

Run simulation for x commands. Works only if the simulator is "ready"

12:17 ص 26°C صافي غالة ENG ٢٠٢٣/٠٩/١٣

Tool: D:\Embedded\Diagrams\First_Term\Unit_4\Date_Structure\Lec_3\lap\simulation\unit4_lesson_3.xml

File Edit View Code Generation View Tool Help

Analysis Requirements Design

Block Diagram AlarmActuatorDriver PressureSensorDriver Pressure_Detection_Learn_in_depth AlarmMonitor MainAlgorithm

```
stateDiagram-v2
    [*] --> Init
    Init --> Read
    Read --> Read : PVal = RANDOM(15, 25)
    Read --> Read : SetPressure(PVal)
    Read --> Read : setTimer(PSensorPullTime, 100)
    Read --> Waiting
    Waiting --> Read : expire(PSensorPullTime)
    Waiting --> Read : reset(PSensorPullTime)
```

Switch to the first diagram

Interactive simulation

Terminate simulation and quit

Commands

Control Save trace

Simulation information

Status: Stopped Time: 0 Transactions: 32 Coverage: 35.7%

Options

- ☒ Displayed blocks
- ☐ Latencies
- ☐ Randomness
- ☐ Asynch. msg
- ☐ Blocks
- ☐ Variables
- ☐ Transactions
- ☐ Met states

☒ Animate UML diagrams

☐ Show AVATAR IDs on UML diagrams

☐ Show hidden state in sequence diagram

☒ Automatically open active state machine diagram

☒ Trace in sequence diagram # of transactions: 1 Index of last transaction: 0

☒ Automatically execute empty transitions

☒ Automatically enter states

Pending transactions

in Block PressureSensorDriver: (SYNCHRO) Sending signal

```
sequenceDiagram
    participant PS as PressureSensorDriver
    participant AM as AlarmMonitor
    PS->>PS: Init
    PS->>PS: Waiting
    AM->>AM: Init
    AM->>AM: Read
    AM->>PS: StopAlarm()
    Note over PS: PVal = 10
```

Run simulation for x commands. Works only if the simulator is "ready"

12:17 ص 28°C صافي غاليه ENG 1471/4/18

Interactive simulation

File Edit View V&V Code Generation View Tool Help

Block Diagram AlarmActionDriver AlarmActionDriver PressureGenerator PressureGenerator AlarmMonitor AlarmMonitor

Control Save Trace

Simulation information: Status: Stopped Time: 200ps Transactions: 20 Coverage: 34.4%

Options: ☐ Displayed blocks ☐ Transactions ☐ Randomness ☐ Asynch. msg ☐ Blocks ☐ VAVs ☐ Transactions ☐ Net states ☐ Animate UML diagrams ☐ Show VAVs on UML diagrams ☐ Show hidden states in sequence diagrams ☐ Automatically open active state machine diagram ☐ Traces in sequence diagrams: # of transactions: 1 Index of last transaction: 0 ☐ Automatically execute empty transitions ☐ Automatically enter states

UML Sequence Diagram:

Participants: AlarmActionDriver, PressureGenerator, AlarmMonitor

Messages:

- AlarmActionDriver->>AlarmMonitor: init
- AlarmActionDriver->>PressureGenerator: init
- PressureGenerator->>AlarmMonitor: Read
- AlarmMonitor->>PressureGenerator: Pval = RANDOM(1:2)
- PressureGenerator->>AlarmMonitor: SetPressure(Pval)
- AlarmMonitor->>PressureGenerator: TimeToProcessPressure
- PressureGenerator->>AlarmMonitor: Wait
- AlarmMonitor->>PressureGenerator: GetPressure(Pval)
- PressureGenerator->>AlarmMonitor: GetPressure(Pval)

UML State Machine Diagram:

States: init, Read, Pval = RANDOM(1:2), SetPressure(Pval), TimeToProcessPressure, Wait, GetPressure(Pval), GetPressure(Pval)

Transitions:

- init --> Read
- Read --> Pval = RANDOM(1:2)
- Pval = RANDOM(1:2) --> SetPressure(Pval)
- SetPressure(Pval) --> TimeToProcessPressure
- TimeToProcessPressure --> Wait
- Wait --> GetPressure(Pval)
- GetPressure(Pval) --> Read

Simulation log: "Ready" for simulation

Tool D:\Embedded Diploma\Kerlos\First_Term\Unit_4\Date_Structure\Lec_3\api\simulation\unit4_lesson_3.xml

File Edit V&V Code Generation View Tool Help

Analysis Requirements Design

Block Diagram AlarmActuatorDriver PressureSensorDriver Pressure_Detection_Learn_in_depth AlarmMonitor MainAlgorithm

```
stateDiagram-v2
    [*] --> StopAlarm1: 
    StopAlarm1 --> AlarmOff: 
    AlarmOff --> HighPressureDetected: >HighPressureDetected()
    HighPressureDetected --> AlarmOn: 
    AlarmOn --> StartAlarm: StartAlarm()
    StartAlarm --> setTimer: setTimer(AlarmTimer,60)
    setTimer --> Waiting: 
    Waiting --> expire: >expire(AlarmTimer)
    expire --> reset: reset(AlarmTimer)
    reset --> StopAlarm2: StopAlarm()
    StopAlarm2 --> [*]: 
```

Switch to the first diagram

Interactive simulation

Terminate simulation and quit

Commands

Control Save trace

Simulation information

Status: **Stopped** Time: 0 Transactions: 20 Coverage: 20.0%

Displayed blocks Latencies Randomness Async. msg

Options Blocks Variables Transactions Met states

☒ Animate UML diagrams

☐ Show AVATAR IDs on UML diagrams

☐ Show hidden state in sequence diagram

☒ Automatically open active state machine diagram

☒ Trace in sequence diagram # of transactions: 1 Index of last transaction: 0

☒ Automatically execute empty transitions

☒ Automatically enter states

Nb of steps: 1

Pending transactions

in Block PressureSensorDriver: Random between 15 and 2

in Block AlarmMonitor: [SYNCHRO] Sending signal StopAlarm

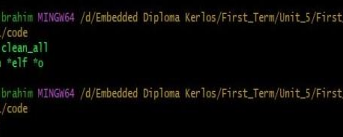
```
sequenceDiagram
    participant AlarmActuatorDriver
    participant PressureSensorDriver
    participant MainAlgorithm
    AlarmActuatorDriver->>AlarmActuatorDriver: Init
    AlarmActuatorDriver->>AlarmActuatorDriver: Waiting
    PressureSensorDriver->>PressureSensorDriver: Init
    PressureSensorDriver->>PressureSensorDriver: Read
    AlarmActuatorDriver-->>PressureSensorDriver: Read
    PressureSensorDriver-->>MainAlgorithm: startstate
```

Ready...

12:17 ص 26°C صافي غاليّة ١٣/٧/٢٠٢٠

Code

File Name	File Type	Size
Alarm_Actuator.c	C File	1 KB
Alarm_Actuator.h	H File	1 KB
Alarm_monitor.c	C File	1 KB
Alarm_monitor.h	H File	1 KB
driver.c	C File	1 KB
driver.h	H File	1 KB
linker_script.ld	LD File	1 KB
main.c	C File	1 KB
Main_ALG.c	C File	1 KB
Main_ALG.h	H File	1 KB
MAKEFILE	File	1 KB
Map_file.map	MAP File	11 KB
Pressure_Sensor.c	C File	1 KB
Pressure_Sensor.h	H File	1 KB
Pressure_Sensor_cortex_m3.elf.asm	ASM File	1 KB
startup.c	C File	2 KB
state.h	H File	1 KB



A screenshot of a Windows command prompt window. The title bar at the top reads "MINGW64/d/Embedded Diploma Kerlos/First_Term/Unit_5/First_Term_proj...". The window contains the following text:

```
MAGIC@ibrahim MINGW64 /d/Embedded Diploma Kerlos/First_Term/Unit_5/First_Term_pr
ject_1/code
$ make clean_all
rm *bin *elf *o

MAGIC@ibrahim MINGW64 /d/Embedded Diploma Kerlos/First_Term/Unit_5/First_Term_pr
ject_1/code
$ make
```

Name	Date modified	Type	Size
Alarm_Actuator.c	٢٠٢٣/٩/١٧ م ١٤:٥٢	C File	1 KB
Alarm_Actuator.h	٢٠٢٣/٩/١٧ م ١٤:٥٩	H File	1 KB
Alarm_Actuator.o	٢٠٢٣/٩/١٧ م ١٥:٢٧	O File	9 KB
Alarm_monitor.c	٢٠٢٣/٩/١٧ م ١٥:٢٥	C File	1 KB
Alarm_monitor.h	٢٠٢٣/٩/١٧ م ١٥:٢٥	H File	1 KB
Alarm_monitor.o	٢٠٢٣/٩/١٧ م ١٥:٢٧	O File	9 KB
driver.c	٢٠٢٣/٩/١٧ م ١٥:٢٣	C File	1 KB
driver.h	٢٠٢٣/٩/١٧ م ١٥:٢٣	H File	1 KB
driver.o	٢٠٢٣/٩/١٧ م ١٥:٢٥	O File	9 KB
linker_script.ld	٢٠٢٣/٩/١٧ م ١٥:٢٣	LD File	1 KB
main.c	٢٠٢٣/٩/١٧ م ١٥:٢٥	C File	1 KB
main.o	٢٠٢٣/٩/١٧ م ١٥:٢٧	O File	9 KB
Main_ALG.c	٢٠٢٣/٩/١٧ م ١٥:٢٧	C File	1 KB
Main_ALG.h	٢٠٢٣/٩/١٧ م ١٥:٢٧	H File	1 KB
Main_ALG.o	٢٠٢٣/٩/١٧ م ١٥:٢٧	O File	4 KB
MAKEFILE	٢٠٢٣/٩/١٧ م ١٥:٢١	File	1 KB
Map_file.map	٢٠٢٣/٩/١٧ م ١٥:٢٧	MAP File	11 KB
Pressure_Sensor.c	٢٠٢٣/٩/١٧ م ١٥:٢٩	C File	1 KB
Pressure_Sensor.h	٢٠٢٣/٩/١٧ م ١٥:٢٩	H File	1 KB
Pressure_Sensor.o	٢٠٢٣/٩/١٧ م ١٥:٢٧	O File	9 KB
Pressure_Sensor_cortex_m3.bin	٢٠٢٣/٩/١٧ م ١٥:٢٧	BIN File	1 KB
Pressure_Sensor_cortex_m3.elf	٢٠٢٣/٩/١٧ م ١٥:٢٧	ELF File	155 KB
Pressure_Sensor_cortex_m3.elf.asm	٢٠٢٣/٩/١٧ م ١٥:٢٣	ASM File	1 KB
startup.c	٢٠٢٣/٩/١٧ م ١٥:٢٤	C File	2 KB
startup.o	٢٠٢٣/٩/١٧ م ١٥:٢٧	O File	5 KB
state.h	٢٠٢٣/٩/١٧ م ١٥:٢٩	H File	1 KB

```
MINGW64~/Embedded Diploma Kerlos/First_Term/Unit5/First_Term_proj...
arm-none-eabi-gcc.exe -c -I. -Ithumb -Icpu-cortex-m3 -Igdwdfw-2 Alarm_Actuator.o
arm-none-eabi-gcc.exe -c -I. -Ithumb -Icpu-cortex-m3 -Igdwdfw-2 Alarm_monitor.o
arm-none-eabi-gcc.exe -c -I. -Ithumb -Icpu-cortex-m3 -Igdwdfw-2 driver.o
arm-none-eabi-gcc.exe -c -I. -Ithumb -Icpu-cortex-m3 -Igdwdfw-2 Main.ALG.c -o Main.ALG.o
arm-none-eabi-gcc.exe -c -I. -Ithumb -Icpu-cortex-m3 -Igdwdfw-2 Pressure_Sensor.o
arm-none-eabi-gcc.exe -c -I. -Ithumb -Icpu-cortex-m3 -Igdwdfw-2 startup.o
arm-none-eabi-ld.exe -T linker_script.ld Alarm_Actuator.o Alarm_monitor.o driver.o Main.ALG.o Pressure_Sensor.o startup.o -o Pressure_Sensor_cortex_m3.o
arm-none-eabi-objcopy.exe -O binary Pressure_Sensor_cortex_m3.elf Pressure_Sensor_cortex_m3.bin
-----build is done-----
MAG2IC@bhrath MINGW64 ~/Embedded Diploma Kerlos/First_Term/Unit5/First_Term_proj...
project3/code
$ make clean_all
```

Simulation

