./

GENESIS - Learning Outcome & Mini-project Summary Report



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ver. Rel. No.** | **Release Date** | **Prepared. By** | **Reviewed By** | **To be Approved** | **Remarks/Revision Details** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Details**

Contents

[Contents 3](#_Toc69569162)

[Mini Project -1 6](#_Toc69569163)

[**Applied System Development Life Cycle and Software Testing** 6](#_Toc69569164)

[Introduction: 6](#_Toc69569165)

[**Research and Cost and features Diagram:** 6](#_Toc69569166)

[**System Design (UML Diagram)** 6](#_Toc69569167)

[**Test Plan:** 8](#_Toc69569168)

[**Implementation Summary** 11](#_Toc69569169)

[**Git Link** 11](#_Toc69569170)

[**Git Dashboard** 11](#_Toc69569171)

[**Summary** 12](#_Toc69569172)

[**Git inspector summary** 12](#_Toc69569173)

[Build 12](#_Toc69569174)

[Code quality and Issues or Bug Tracking 13](#_Toc69569175)

[Unit Testing 14](#_Toc69569176)

[Individual Contribution & Highlights 14](#_Toc69569177)

[Summary 15](#_Toc69569178)

[Challenges faced and how were they overcome 15](#_Toc69569179)

[Future Scope 15](#_Toc69569180)

[Mini project -2 16](#_Toc69569181)

[**Advance Python Programming – Testing** 16](#_Toc69569182)

[**Introduction** 16](#_Toc69569183)

[**Design** 16](#_Toc69569184)

[**Details Requirement** 17](#_Toc69569185)

[**Code Snippet** 18](#_Toc69569186)

[**Implementation Summary** 19](#_Toc69569187)

[**Git Link** 19](#_Toc69569188)

[**Git Dashboard** 19](#_Toc69569189)

[Summary 19](#_Toc69569190)

[Mini project -3 20](#_Toc69569191)

[Embedded -C 20](#_Toc69569192)

[Introduction 20](#_Toc69569193)

[Objectives & Requirements 18](#_Toc69569194)

[Design 19](#_Toc69569195)

[Code Snippet: 19](#_Toc69569196)

[Implementation Summary 19](#_Toc69569197)

[Video Summary 20](#_Toc69569198)

[SharePoint Link 20](#_Toc69569199)

[Summary 20](#_Toc69569200)

[Mini Project 4: 21](#_Toc69569201)

[Implementation of Cruise control Door lock unlock features and Interior light using CANoE tools 21](#_Toc69569202)

[Introduction: 21](#_Toc69569203)

[Procedure: 21](#_Toc69569204)

[Requirements: 21](#_Toc69569205)

[Design: 21](#_Toc69569206)

[SharePoint Link: 25](#_Toc69569207)

[Summary: 25](#_Toc69569208)

[Mini Project 5: 26](#_Toc69569209)

[MBSE: 26](#_Toc69569210)

[Introduction: 26](#_Toc69569211)

[Requirements: 26](#_Toc69569212)

[Design: 26](#_Toc69569213)

[Test Plan: 30](#_Toc69569214)

[SharePoint Link: 31](#_Toc69569215)

[Summary: 31](#_Toc69569216)

[Mini Project 6 32](#_Toc69569217)

[Evolved MBSE 32](#_Toc69569218)

[Introduction: 32](#_Toc69569219)

[Requirements: 32](#_Toc69569220)

[Design: 32](#_Toc69569221)

[Test Plan: 33](#_Toc69569222)

[Share Point Link: 34](#_Toc69569223)

[Summary: 34](#_Toc69569224)

[Future Scope: 34](#_Toc69569225)

[Mini Project 7 35](#_Toc69569226)

[Intermediate C++ 35](#_Toc69569227)

# Mini Project -1

## **Applied System Development Life Cycle and Software Testing**

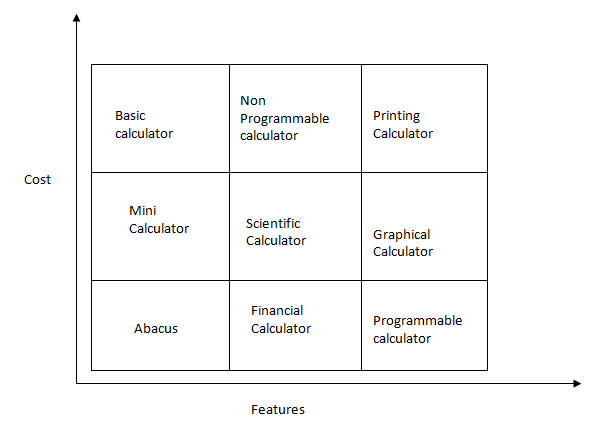
Under this section we have experienced the industrial flow of the project cycle by the example of the calculator.

### Introduction:

A calculator is a machine which allows people to do math operations more easily. For example, most calculators will add, subtract, multiply, and divide. Some also do square roots, and more complex calculators can help with calculus and draw function graphs. Calculators are found everywhere.

## **Research and Cost and features Diagram:**

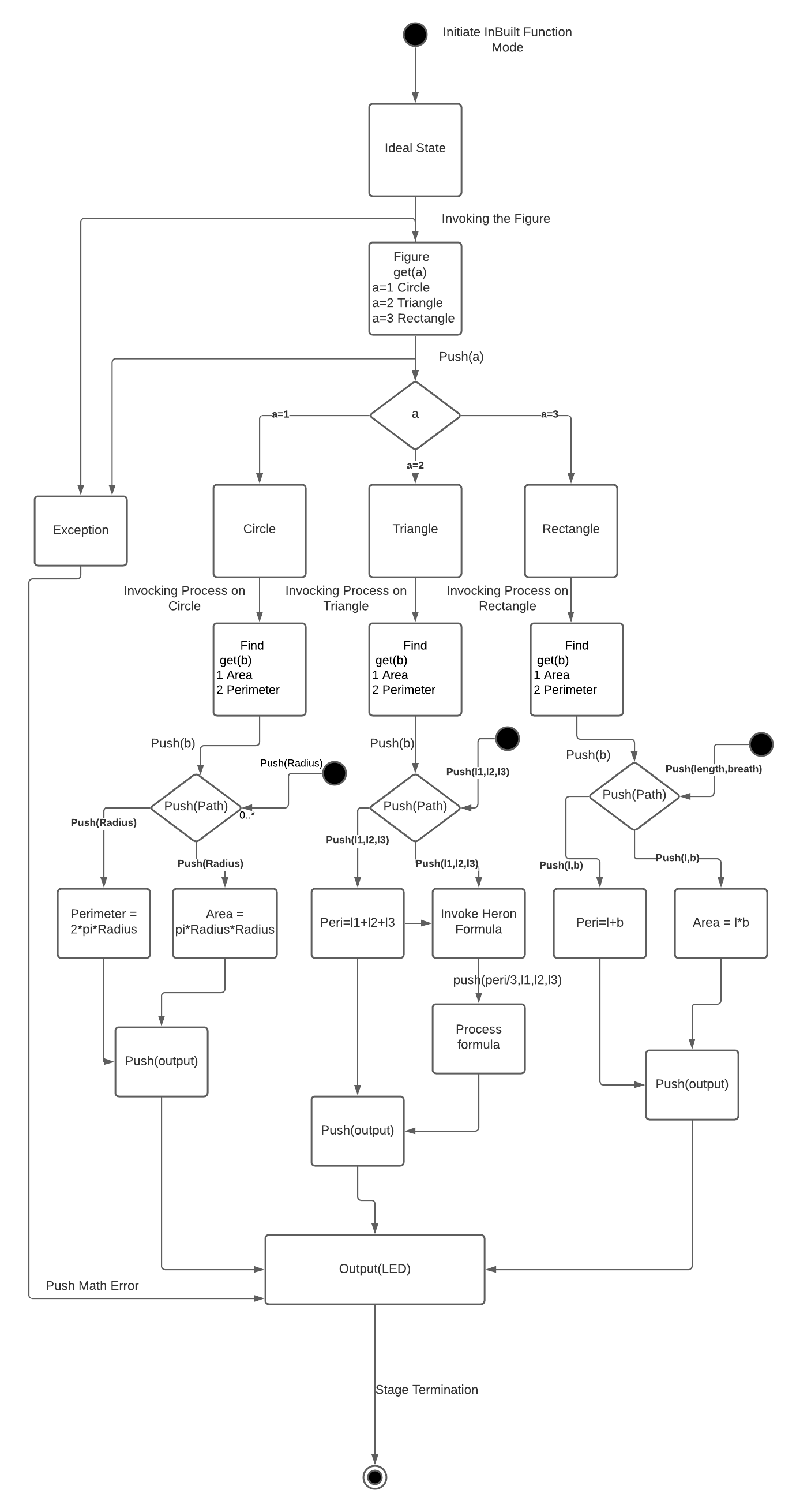
Research on various calculators are done which are present in the market like scientific calculator, Graphing calculator, Abacus, Basic calculator, and their functions are also studied.



## **System Design (UML Diagram)**

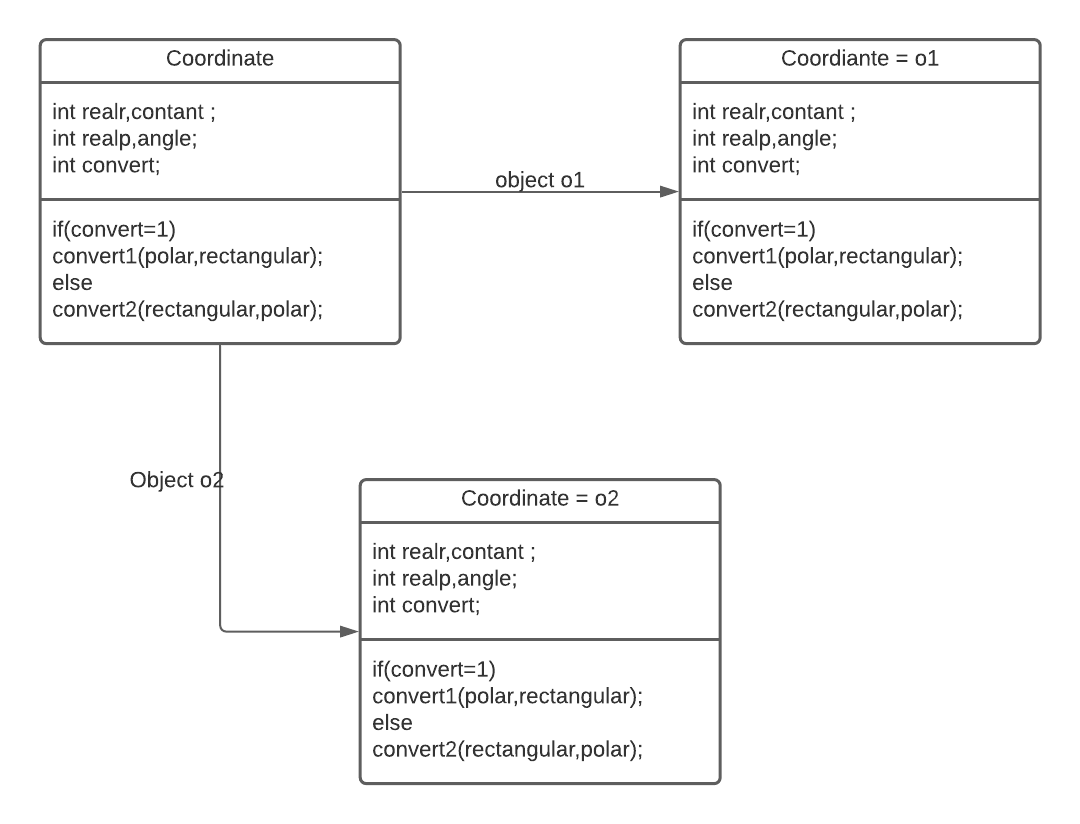
High Level Diagram

1)State Diagram (Inbuild Function):

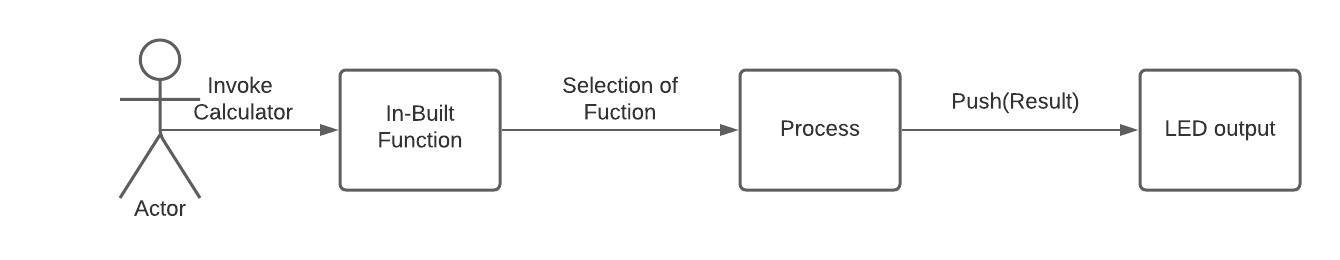


Low Level diagram

1) Object Diagram (Polar to rectangular conversion)



2) Deployment Diagram (Polar to rectangular conversion)



## **Test Plan:**

1)High Level Test Plan:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test ID** | **Description** | **Exp IN** | **Exp OUT** | **Actual Out** | **Type of Test** |
| H\_03 | Integer is taken for the function selection | 1,7 | 154 | 154 | Requirement Based |
| H\_03 | Integer is taken for the function section (other than number) | Ar | Invalid | Invalid | Scenario based |
| H\_03 | Integer is taken for function selection other than given function | 3 | Invalid | Invalid | Boundary Based |

2)Low Level Test Plan:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test ID** | **Description** | **Exp IN** | **Exp OUT** | **Actual Out** | **Type Of Test** |
| L\_11 | Input is taken as a to select the function then the radius data is taken | 7 | 154 | 153.9 | Requirement Based |
| L\_11 | Input taken for the area of the circle | -7 | 153.9 | Invalid Input | Scenario Based |
| L\_11 | Input taken for the area of the circle | 100000000  000000 | 3140000000000  0000000000 | Infy | Boundary Based |
| L\_12 | Input is taken for the polar coordinate and converted into the Cartesian | 7,10 | 6.849,1.215 | 6.849,1.215 | Requirement Based |
| L\_12 | Input is taken for the polar coordinate and converted into the Cartesian (negative number) | -10, -180 | 10,0.006 | 10,0.006 | Scenario Based |
| L\_12 | Input is taken for the polar coordinate and converted into the Cartesian | 100000000000000,  10000000000000 | 220054504836431872.000, -975487540796588032.000 | 22124212231153.000, -554334434664434343334434 | Boundary Based |

## **Implementation Summary**

* Conversion of Polar to Rectangular coordinate and rectangular to polar coordinate,
* Calculation related to figures such as area and perimeter.

## **Git Link**

https://github.com/99003724/LTTS722-727.git

## **Git Dashboard**

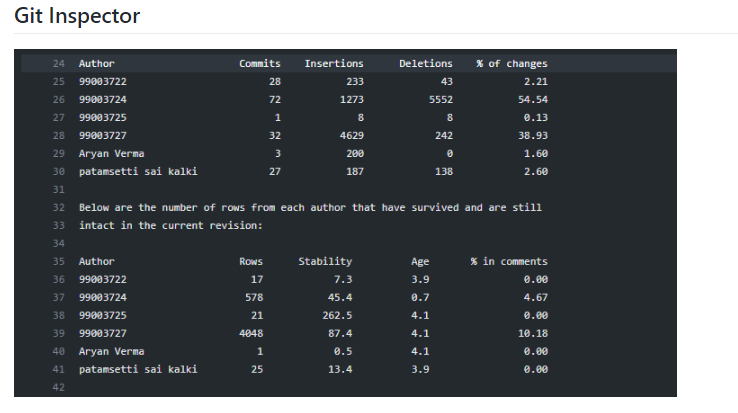


## **Summary**

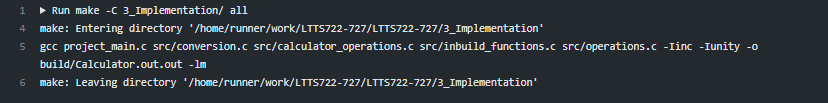
In the overall implementation we agglomerate different feature of the calculator and combine our code to implement overall features.

Features our calculator have basic operation, bit wise operation area calculation of the figure conversion of polar to rectangular coordinate and vice versa.

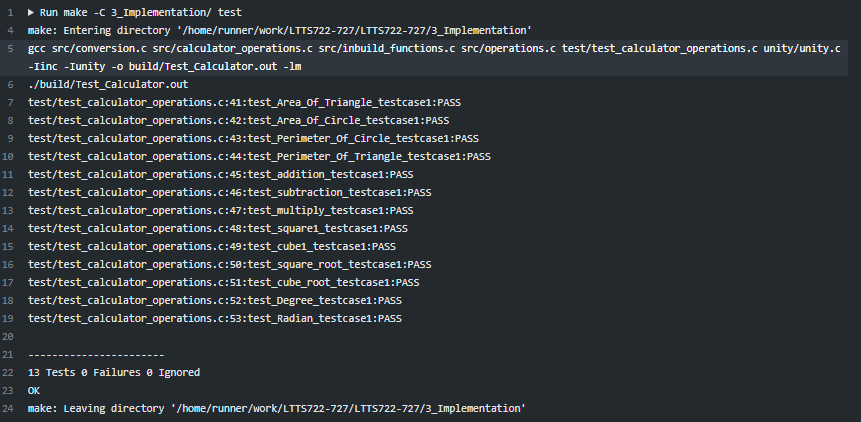
## **Git inspector summary**



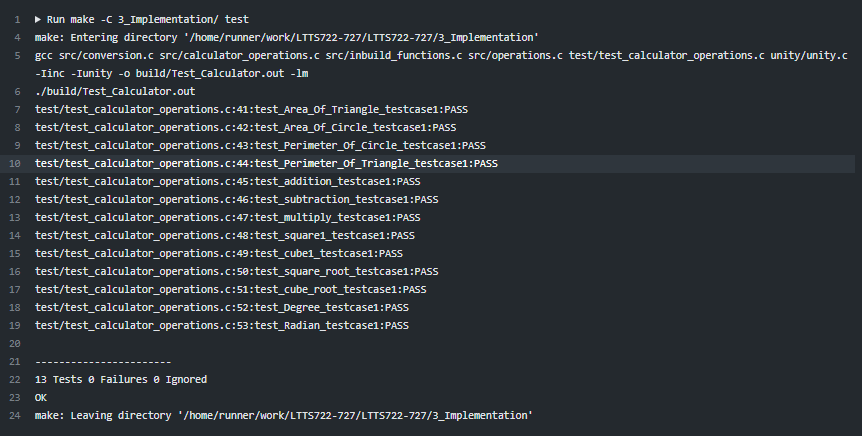
# Build



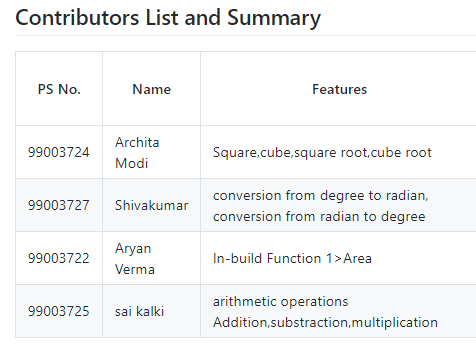
# Code quality and Issues or Bug Tracking



# Unit Testing



# Individual Contribution & Highlights



# Summary

The key learning from this is the construction of the header file .c file and github introduction by implementing it on the practical application with an example of calculator in it.

# Challenges faced and how were they overcome

1. Merging code overcame by creating separate branch and merging to master.
2. Make all command overcame by making separate header file, separate .c file and source file.
3. Conversion of angular coordinate to Cartesian coordinate in which value of pi goes out of bound by defining the constant value of pi.
4. including the headers and to merge the code from local branch to master.

# Future Scope

In this we can have the future scope for the large number calculation.

# Mini project -2

## **Advance Python Programming – Testing**

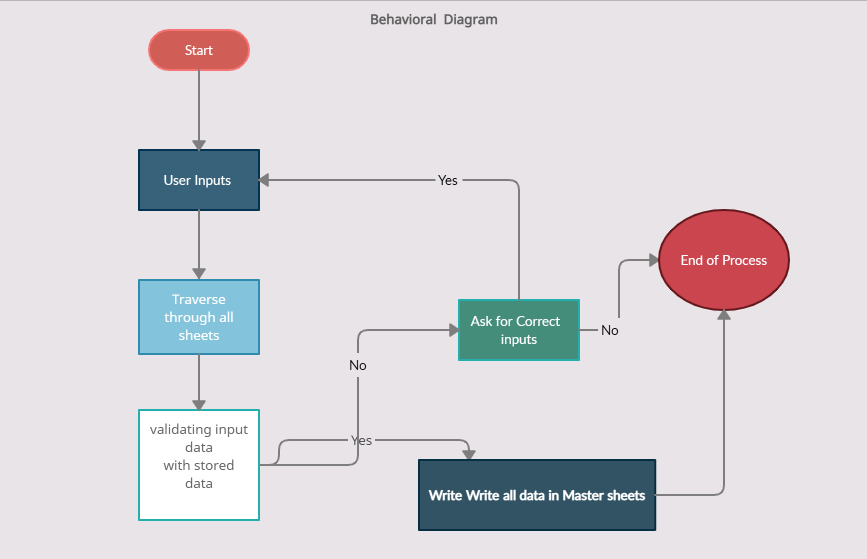
In this Module we have gone through the python syntax, oops concept, regular expression and different tool like PyCharm, jupyter notebook etc.

## **Introduction**

In this we have generated the python script for finding the word given by the user throughout the text file and return the previous and after word and number of times the word got repeated in the output file.

## **Design**

Behavioral Diagram



## **Details Requirement**

High Level Requirement:

|  |  |  |
| --- | --- | --- |
| ID | Requirement | Explanation |
| HLR\_01 | Text File | To get the data for input |
| HLR\_02 | Validating Input | Read given data |
| HLR\_03 | Creating the output text file | To transverse through the data and generating txt file. |

Low Level Requirement

|  |  |  |
| --- | --- | --- |
| ID | Requirement | Explanation |
| LLR\_01 | Search of word | To transverse through the list |
| LLR\_02 | Generating of regular expression | To catch the word from text file |
| LLR\_03 | To save the user input | To get the five word |

## **Code Snippet**



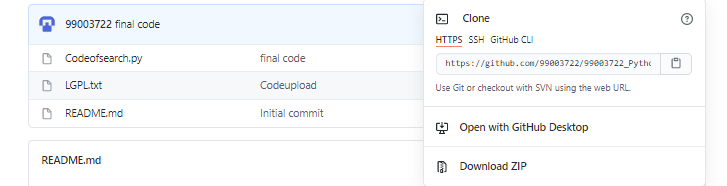
## **Implementation Summary**

For the implementation of the transverse search we have to get the word from the user and transverse through whole text file and use the regular expression to search the given word and generate the output in text file.

## **Git Link**

https://github.com/99003722/99003722\_Python.git

## **Git Dashboard**



# Summary

The data is searched, and the respective text file is generated for the word and the word respective data is stored.

# Mini project -3

## Embedded -C

In this we have used various basic sensor to integrate it with STM 32 module and implement various features to be integrated with the MBSE model.

### Introduction

In this we the following are the snapshot of the driver and pin diagram and the basic knowledge of STM software and assign pin.

MAKE FILE

Below is the make file for the program:



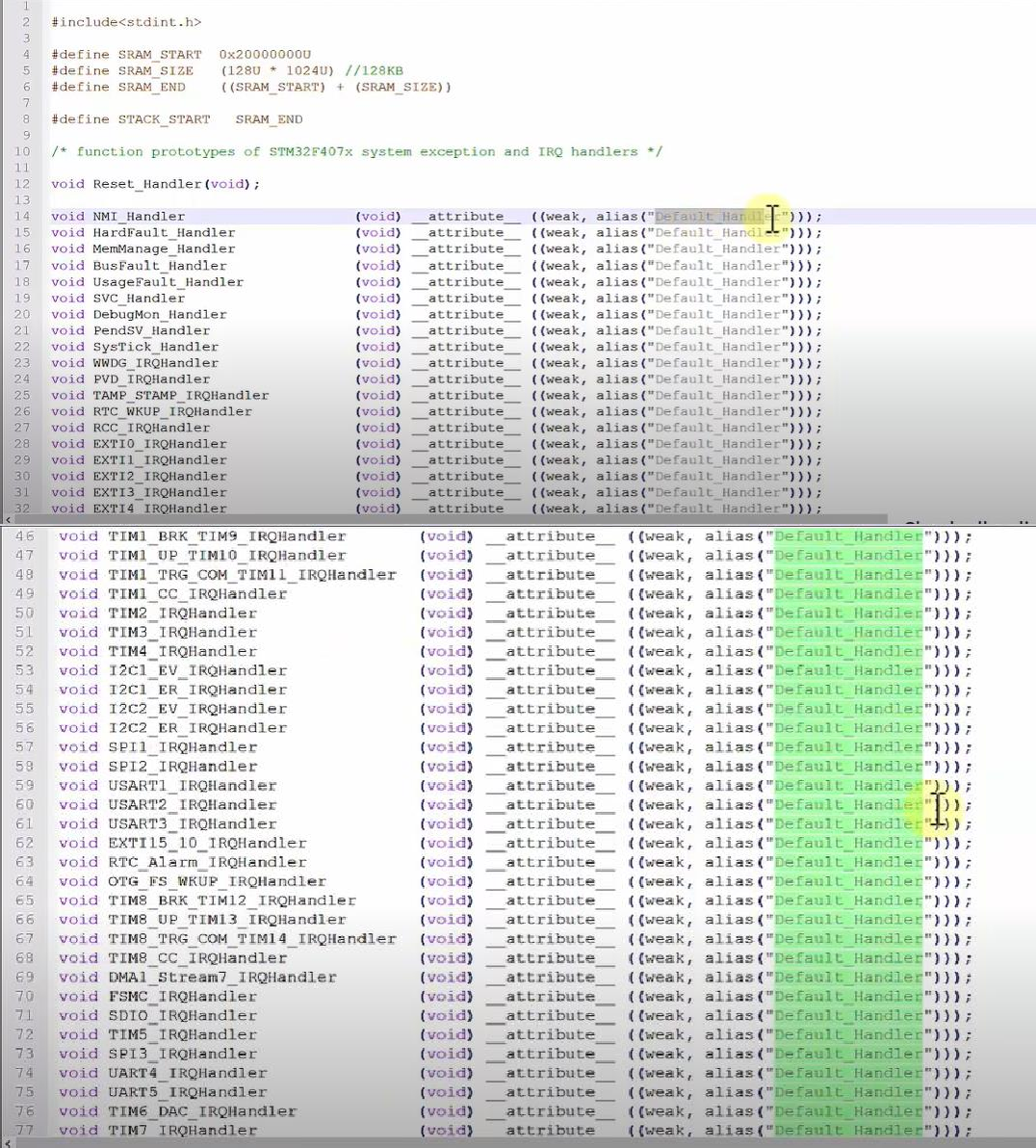
make file

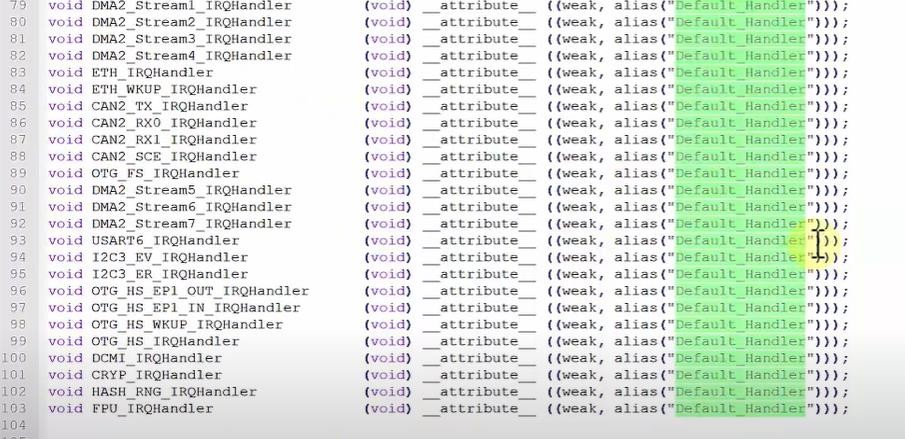
The command to run this make file in the command prompt is:



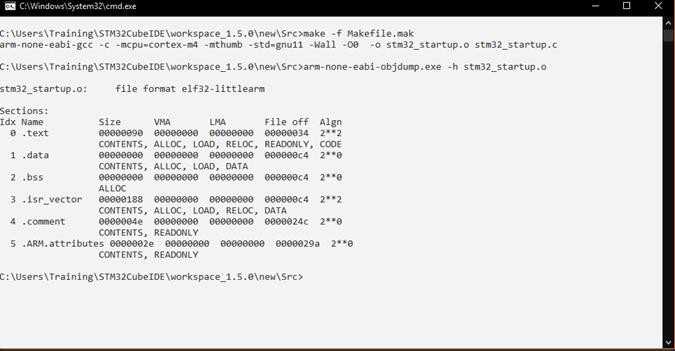
Make command

STARTUP CODE

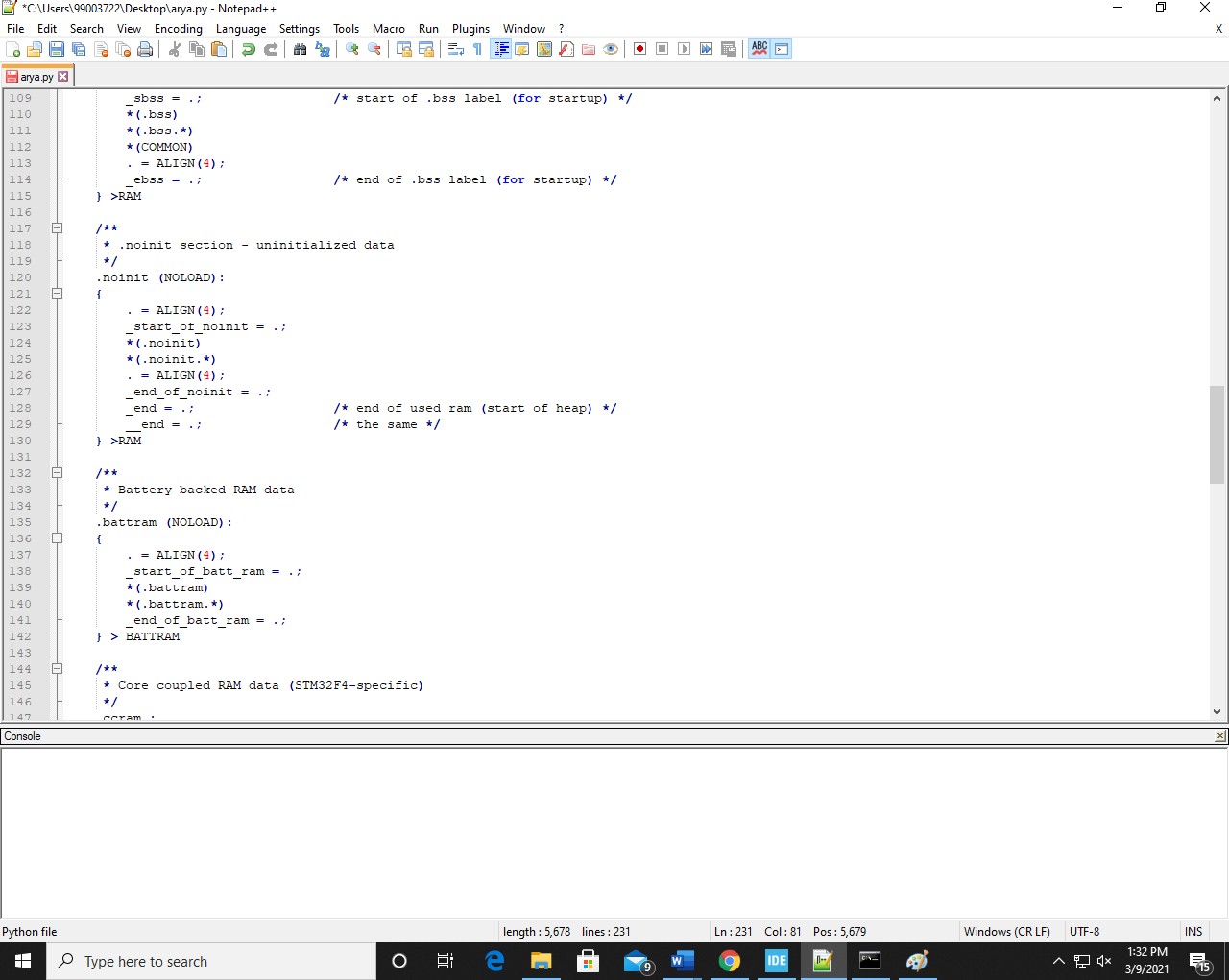




Startup code



Startup command

 LINKER SCRIPT



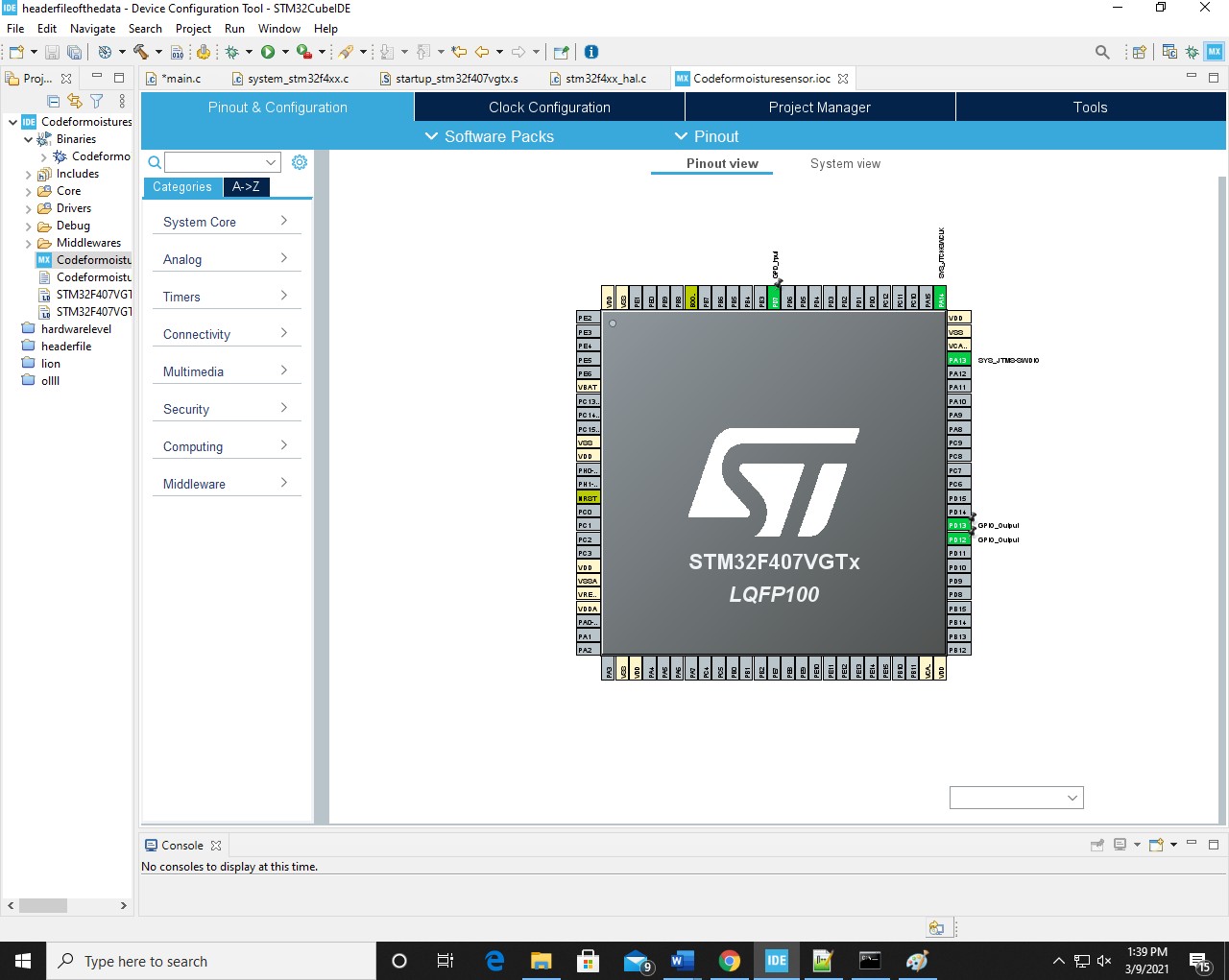
command to generate final. Elf file

IMPLEMENTATION OF PROTOCOLS USING STM IDE

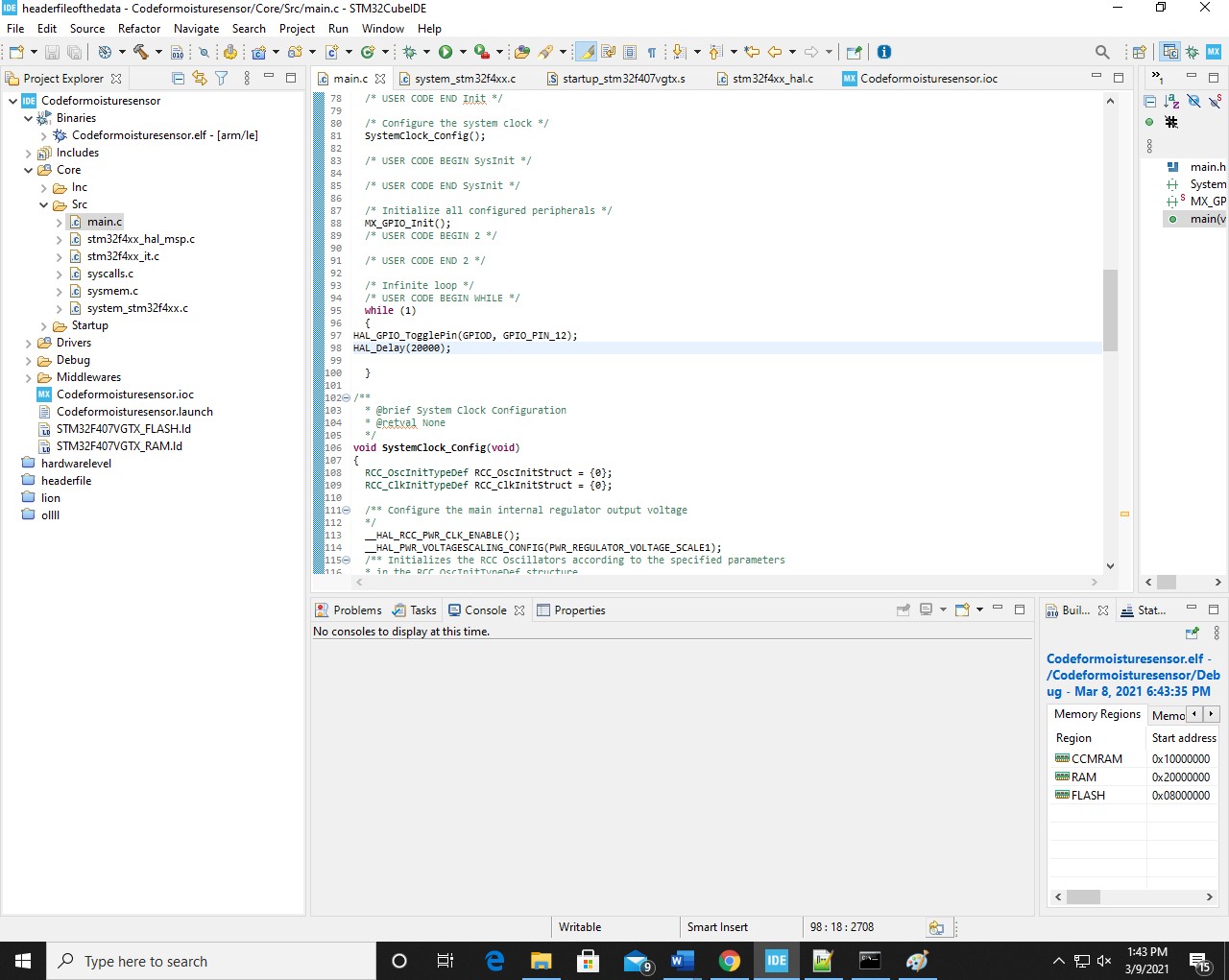
Implementation of protocols for STM32F407VG microcontroller featuring ARM32 bit ARM-cortex - M4 with FPU core using HAL library.

GPIO:

Toggling LED at pin PD12 at GREEN\_LED\_GPIO\_PORT. Serial wire is enabled at pin PA13.



GPIO pin configuration

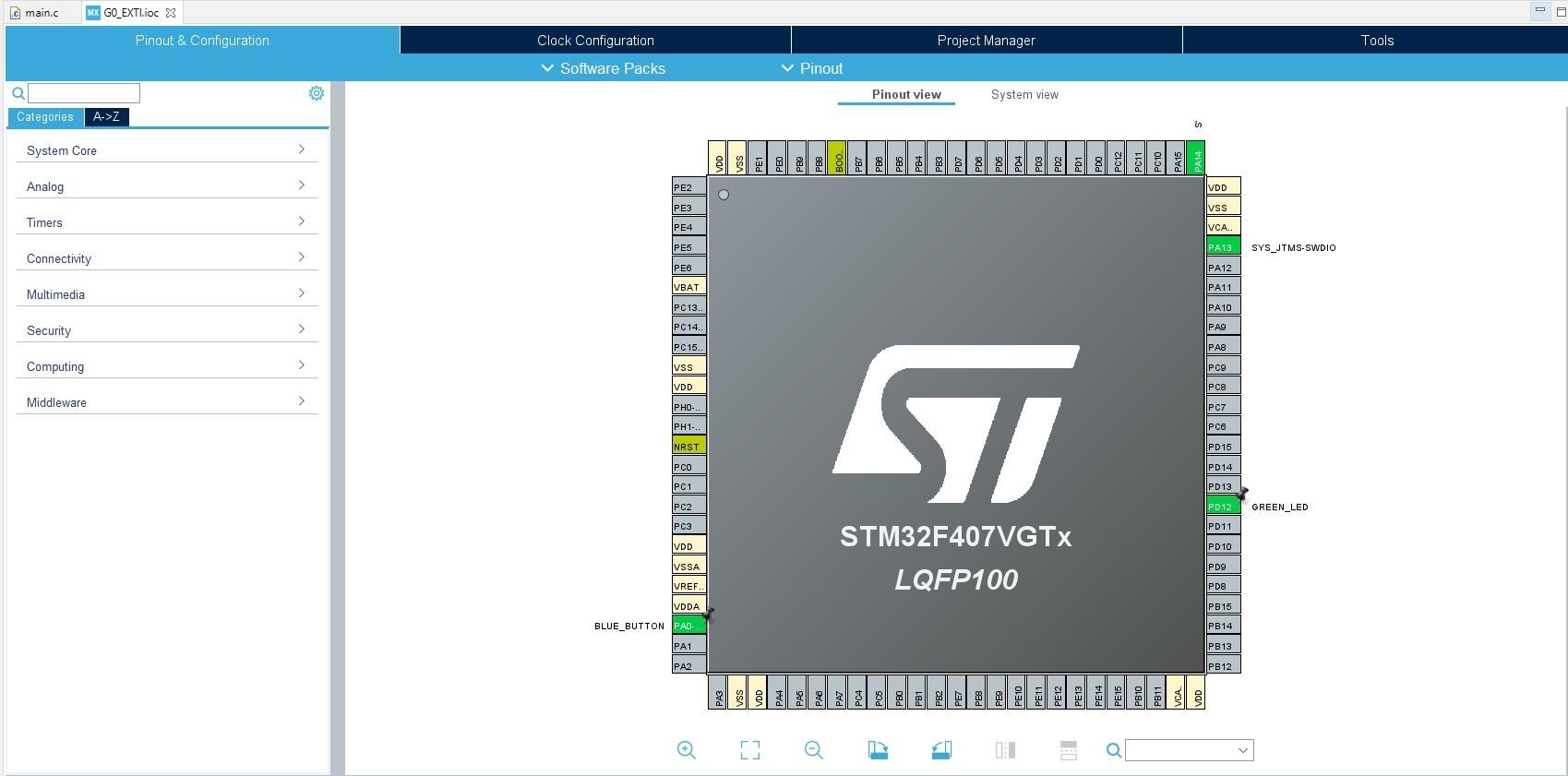


GPIO configuration code

EXTI:

PA0 works as an external interrupt.

When the blue button is pressed the Green LED toggles.



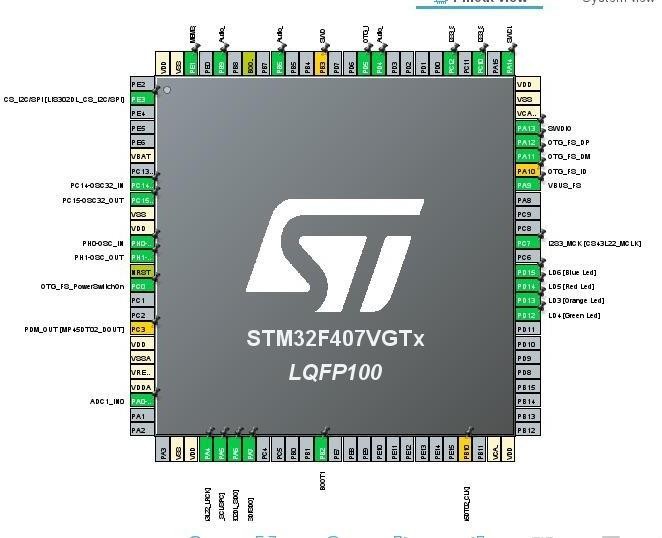
EXTI pin configuration

In the main.c file a flag is initialized and if the flag == 1, the condition under the if loop executed to toggle the LED at PD12.

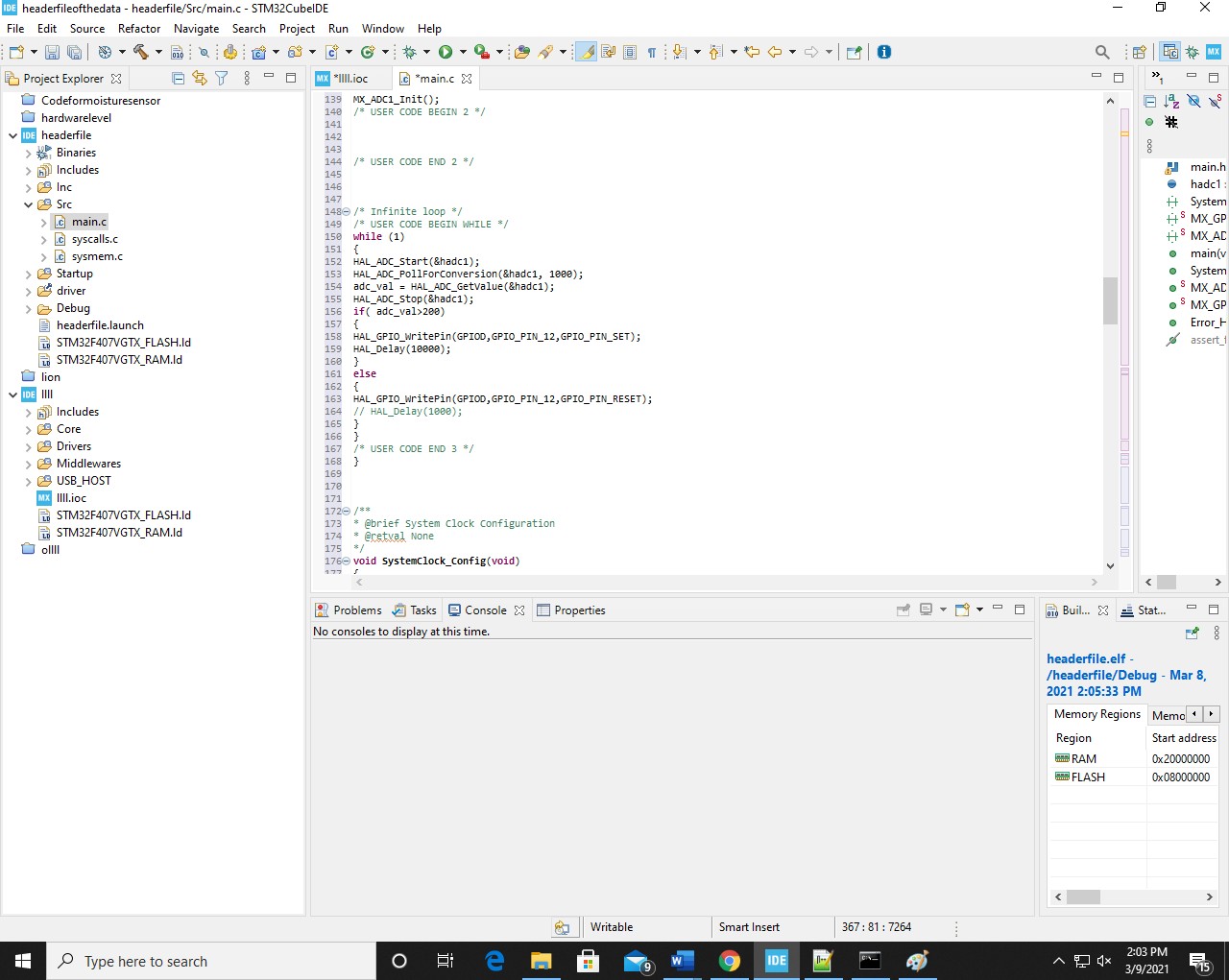


EXTI configuration code

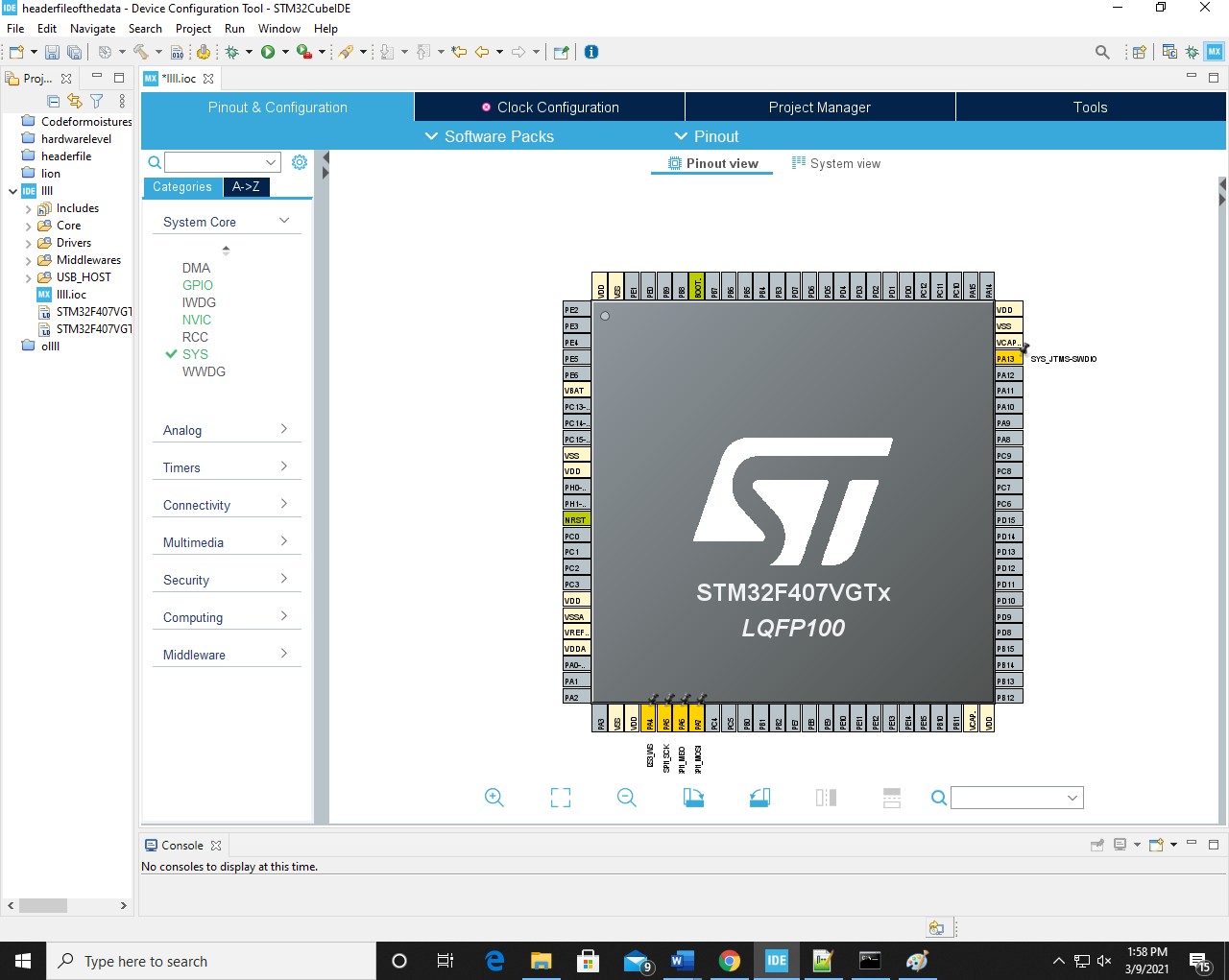
ADC



ADC pin configuration

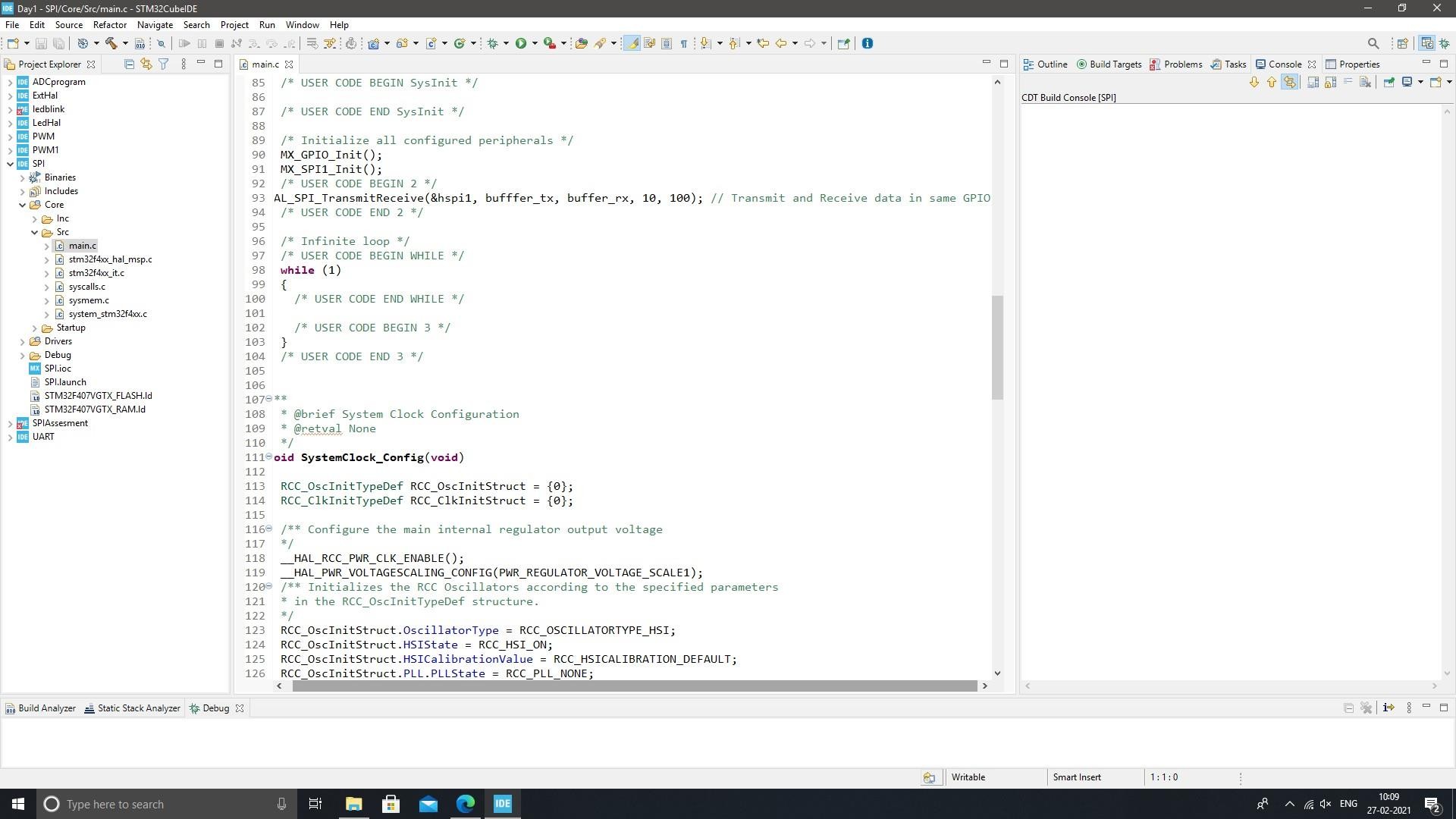


ADC configuration code

SPI

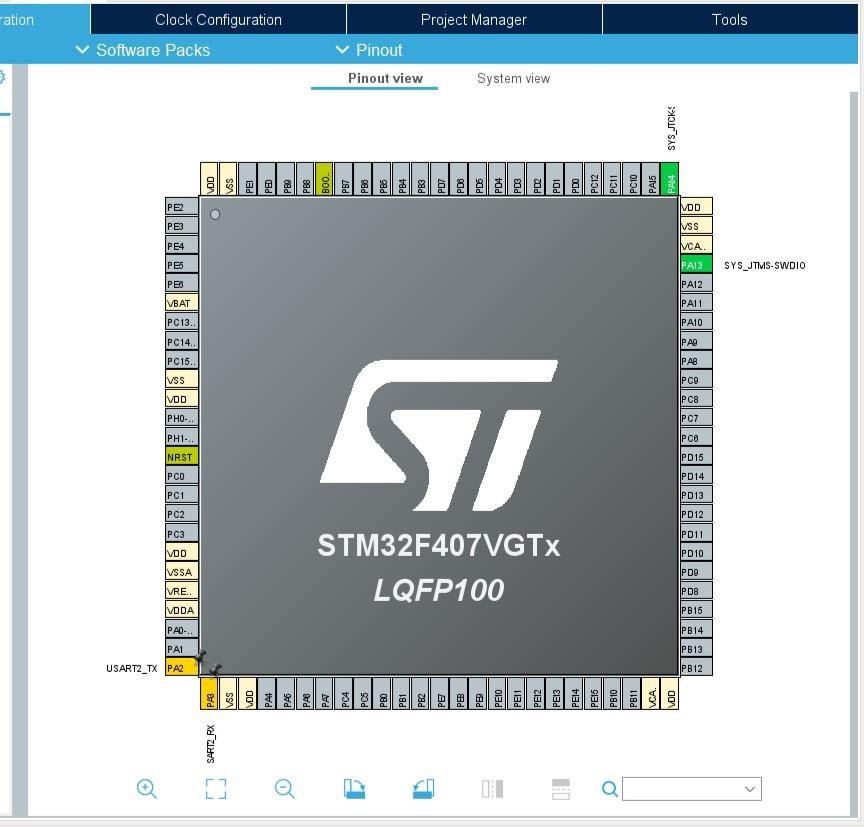
**.**

SPI Pin configuration

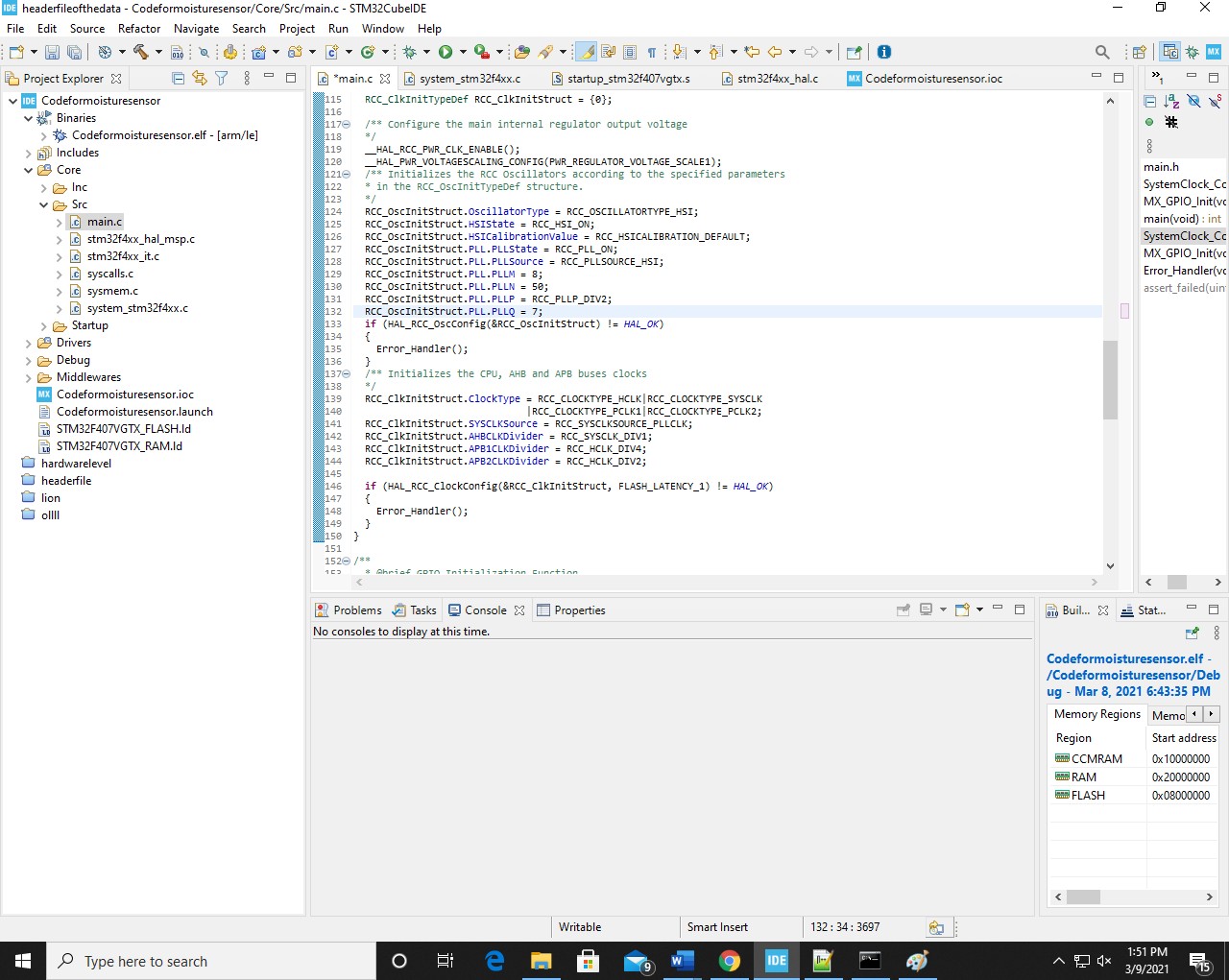


SPI configuration code

UART



UART Pin configuration



UART configuration code

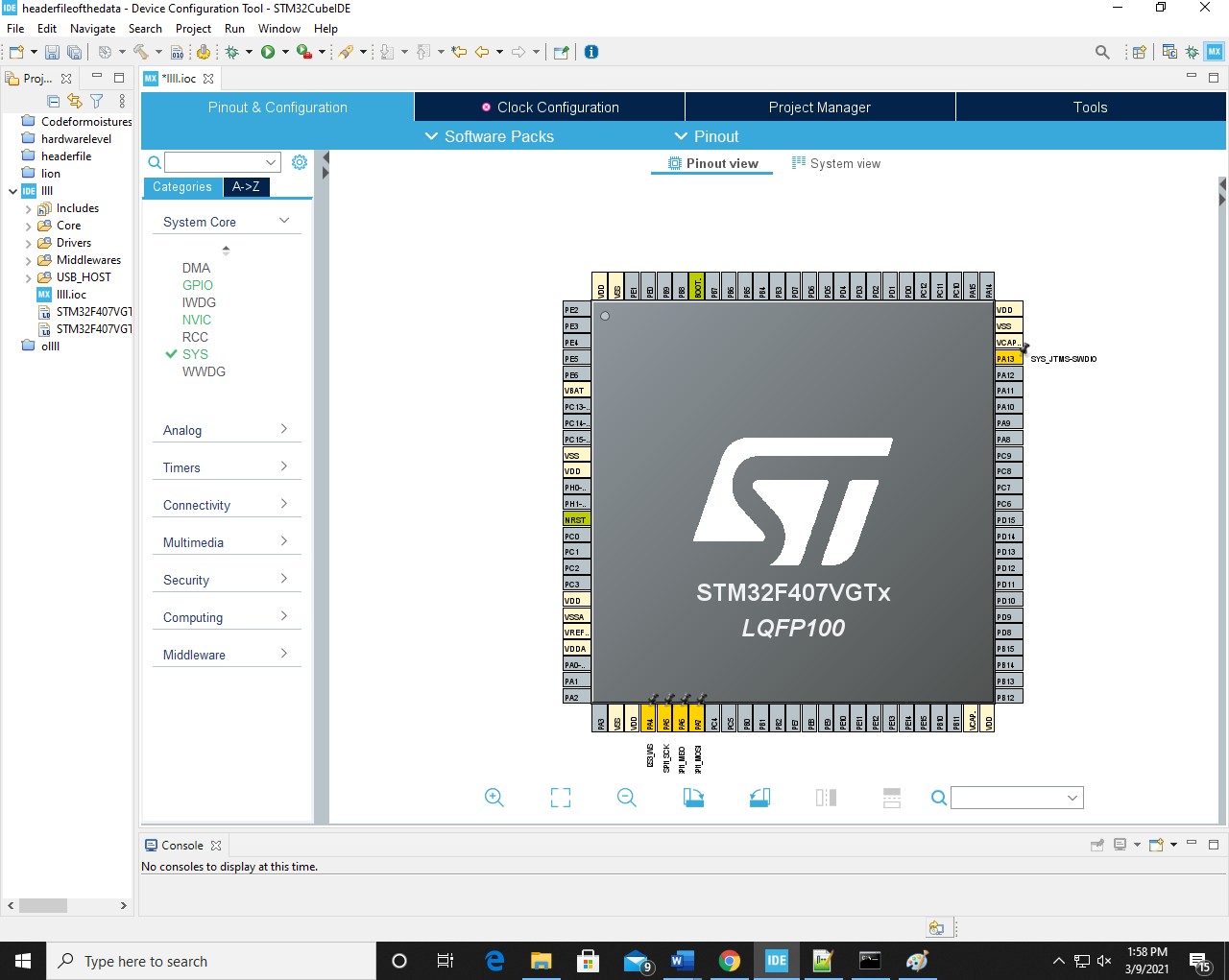
## Objectives & Requirements

The objective of this is to get to know about the embedded system creating the driver and know about the transmission protocol.

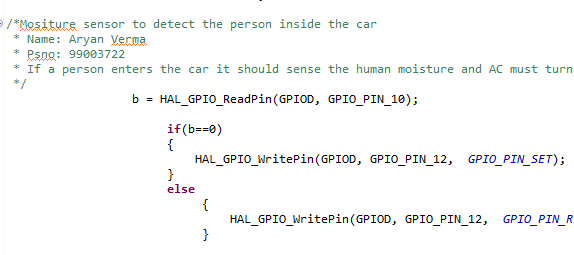
Requirement are the STM-32 IDE and STM board and basic sensor for creating real life project.

## Design

Pin Diagram:



## Code Snippet:



## Implementation Summary

In the implementation of the feature we integrate all the sensor and develop the basic features which can be used in automobile industries.

Integration of LDR, Moisture sensor, Motion sensor, Alcohol sensor on the STM board and creating the prototype of using it on car model.

### Video Summary

In video we have shown the agglomeration of the different sensor on the STM-32 Board and project the output by blinking the LED.

### SharePoint Link

<https://lnttsgroup.sharepoint.com/sites/GEA/Global%20Engineering%20Academy/GENESIS%20%20Jan%202020%20Submission/Forms/AllItems.aspx?ct=1618560455375&or=OWA%2DNT&cid=9292fe1d%2Dc175%2Dfa05%2D2acd%2D6c3f3fc60e1a&originalPath=aHR0cHM6Ly9sbnR0c2dyb3VwLnNoYXJlcG9pbnQuY29tLzpmOi9zL0dFQS9HbG9iYWwlMjBFbmdpbmVlcmluZyUyMEFjYWRlbXkvRWtMWUlHVW8xT0pCdUlpR0ZWUW5WMmtCV0ZLNkZCY1FKOTA0X0VldC15R202UT9ydGltZT1wQ2tac3E0QTJVZw&id=%2Fsites%2FGEA%2FGlobal%20Engineering%20Academy%2FGENESIS%20%20Jan%202020%20Submission%2FSubmission%2FBANGALORE%2F2002BLRGT04%2F99003722%2FEmbeddedC%2FPRJ%5FT%2DTeam3%2899003722%2C99003765%2C99003767%29%2Ezip&parent=%2Fsites%2FGEA%2FGlobal%20Engineering%20Academy%2FGENESIS%20%20Jan%202020%20Submission%2FSubmission%2FBANGALORE%2F2002BLRGT04%2F99003722%2FEmbeddedC>

### Summary

In the implementation of the automobile features we develop the driver for the input and the outport port of the STM-32 board and connecting it with sensor and develop the prototype.

# Mini Project 4:

# Implementation of Cruise control Door lock unlock features and Interior light using CANoE tools

## Introduction:

In this we have used he Vector CANoE tool to implement the Automobile Features.

## Procedure:

In this we agglomerate all the button and put everything on the infotainment clustere and write the CApl Script for all the button which represent the prototype of the features we have implemented.

## Requirements:

High Leve Requirement:

|  |  |  |
| --- | --- | --- |
| ID | Requirement | Explanation |
| HL\_01 | Door locking features | Door lock and unlock features |
| HL\_02 | Turning Light | Car turning and changing lane indicator |
| HL\_03 | Cruise Control | To Control the speed |

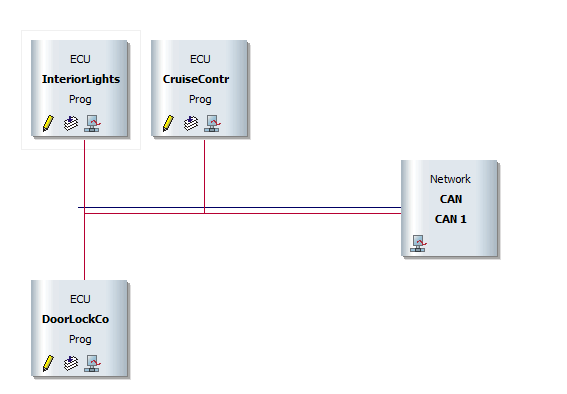
Low Level Requirement:

|  |  |  |
| --- | --- | --- |
| ID | Requirement | Explanation |
| LL\_01 | Speed Door locking | If the speed of car is greater than 15Km/Hr. |
| LL\_02 | Automatic off turning signal | Turning off the signal after 5sec |

## Design:

CAN Model:

In this we have connected three ECU with the CAN network.



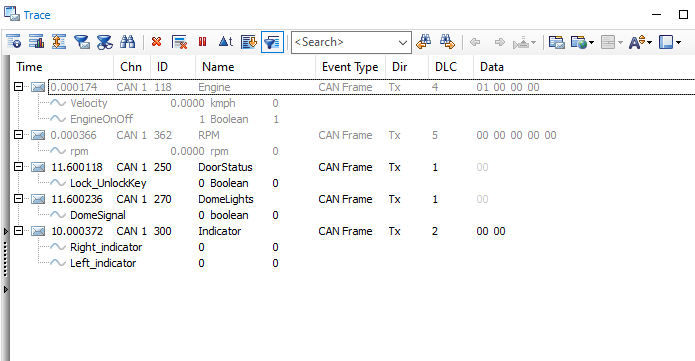
Dashboard:

In this we agglomerate speedometer cruise control and lighting control by the door and cruise control.



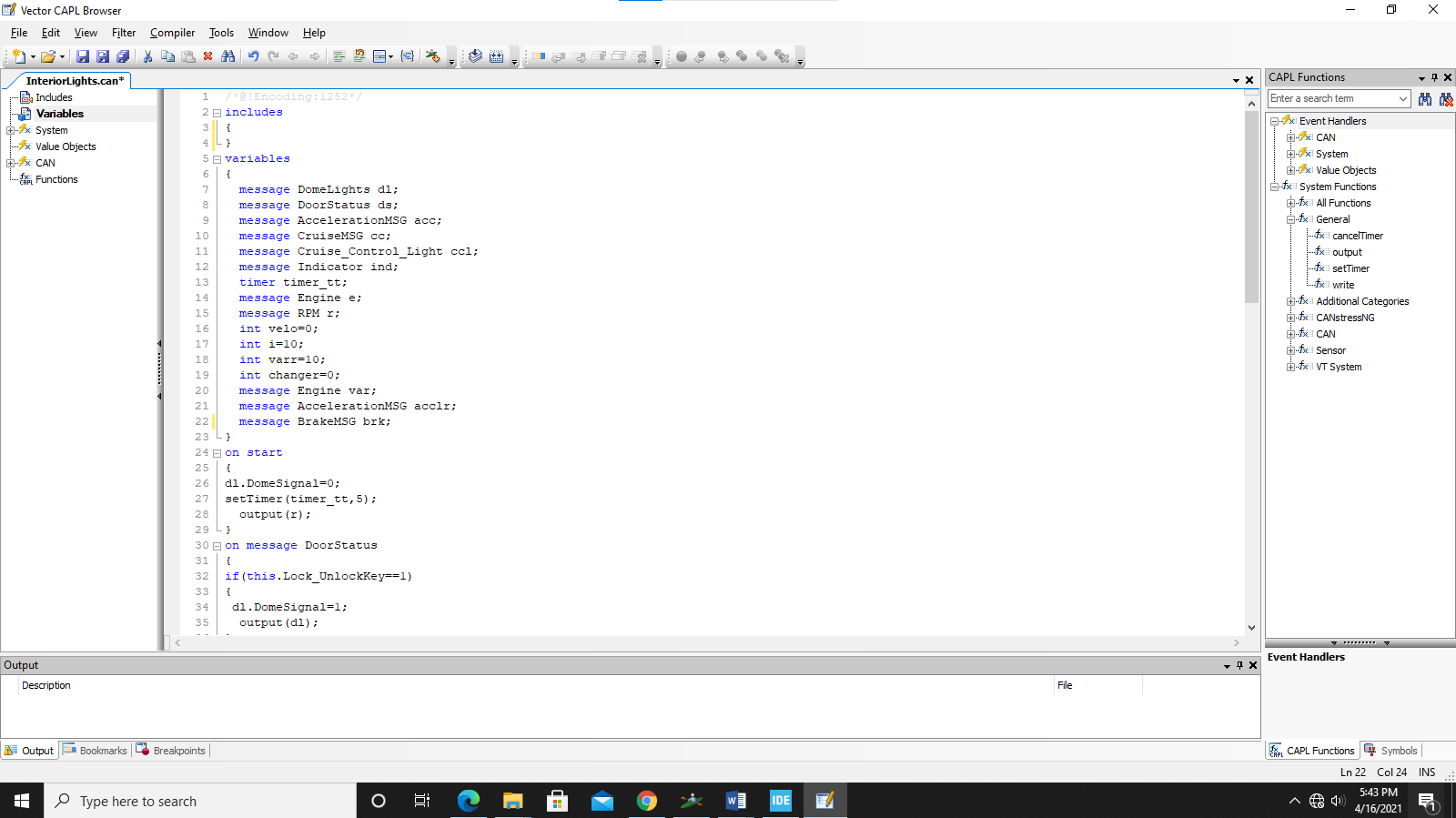
Trace Window:

Trace window to analyses the numerical values of the applied input.



CAPL Scripting:

Scripting of the interior light ECU.



## SharePoint Link:

<https://lnttsgroup.sharepoint.com/sites/GEA/Global%20Engineering%20Academy/GENESIS%20%20Jan%202020%20Submission/Forms/AllItems.aspx?csf=1&web=1&e=HLDjAs&cid=b65a94f3%2D5a90%2D4cef%2D9f85%2D3eb40b9c0625&RootFolder=%2Fsites%2FGEA%2FGlobal%20Engineering%20Academy%2FGENESIS%20%20Jan%202020%20Submission%2FSubmission%2FBANGALORE%2F2002BLRGT04%2F99003729%2FCANoe%20MiniProject&FolderCTID=0x012000CE8923C53B55214CA65BC5D23F0E8415>

## Summary:

Implementation of cruise control, Interior Lighting, Door locking and unlocking features. Writing of the CAPL script of all the features and output is being displayed on Dashboard.

# Mini Project 5:

# MBSE:

## Introduction:

Safety blind Zone:

Safety blind Zone is the feature in which we have either used Radar sensor or image processing for tracking of the vehicle so that if the car came in the certain range it will activate the signal to not change the lane and it will blink the hazard lamp.

Fuel Indicator:

Fuel indicator will display the fuel and display the calibrated value on the display and indicate the tank condition.

## Requirements:

In Matlab modelling we must develop the constraints for the required sensor value and put the constraint in the same model to develop real modeling.

High level Requirement:

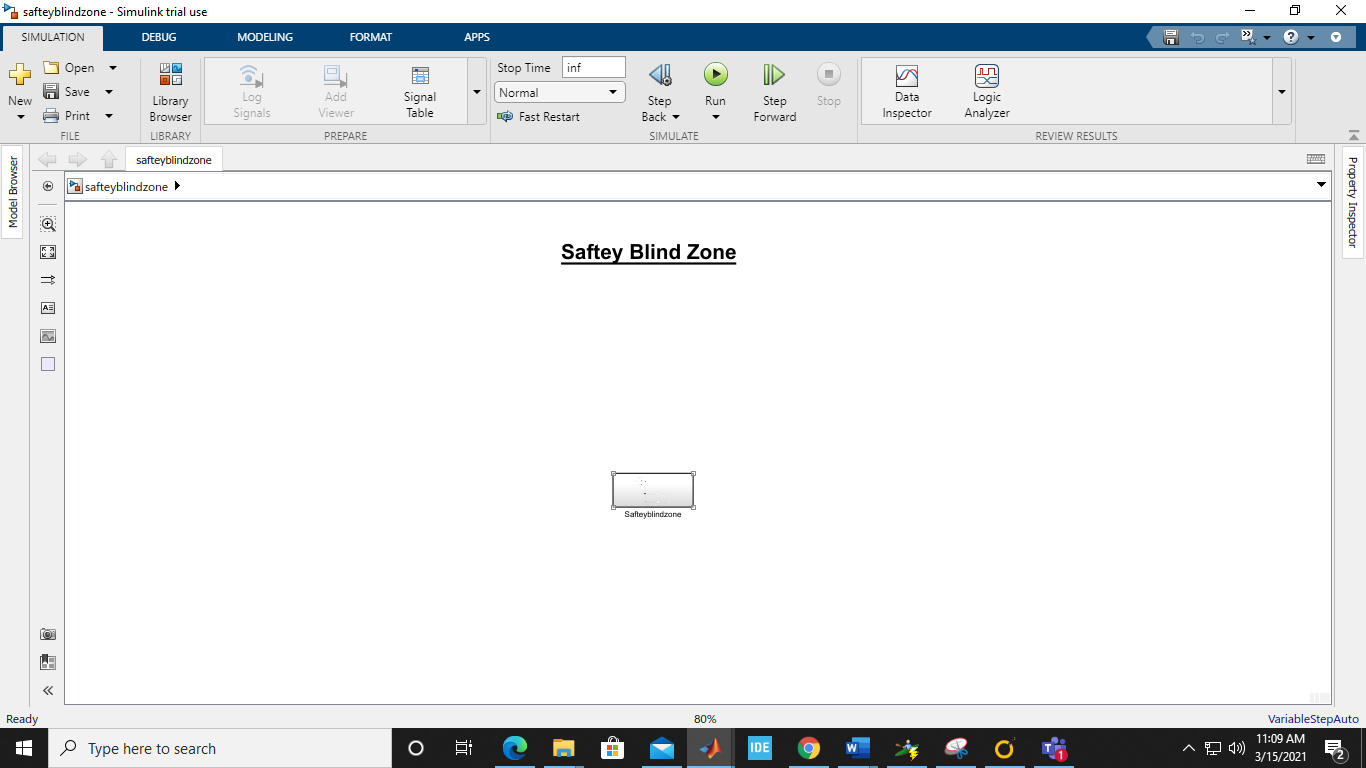
|  |  |
| --- | --- |
| Requirement | Explanation |
| HLR\_1(Safety Blind zone) | Detection of the vehicle |
| HLR\_1(Safety Blind zone) | Accurate activation of the hazard lamp |
| HLR\_1(Fuel Indicator) | Accurate measure of the fuel value |
| HLR\_1(Fuel Indicator) | Blinking of the reserve fuel value |

Low Level Requirement:

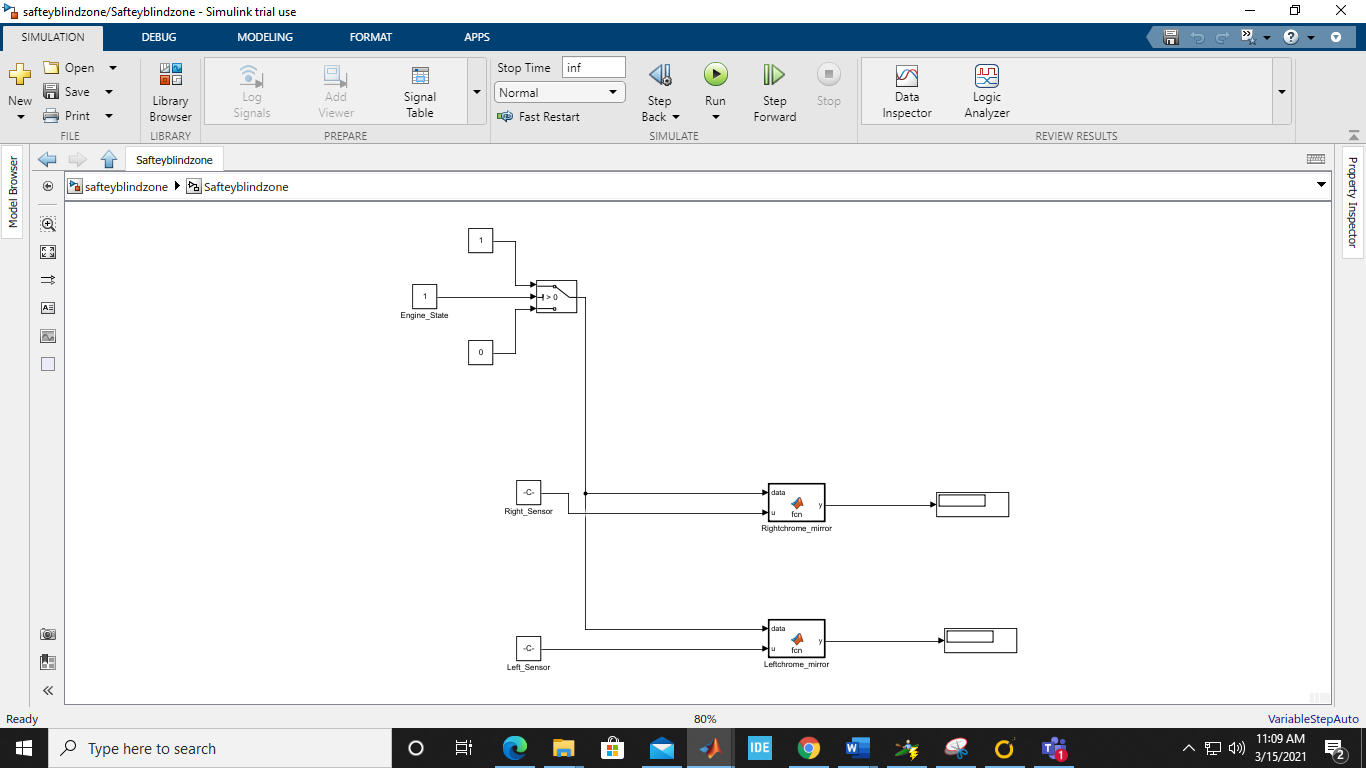
|  |  |
| --- | --- |
| Requirement | Explanation |
| LLR\_1(Safety blind Zone) | Proximity of the sensor must be accurate |
| LLR\_2(Safety blind Zone) | Too nearby object must not be detected |
| LLR\_1(Fuel indicator) | Fuel density must be calibrated |
| LLR\_2(Fuel indicator) | Fuel data conversion to part per million (ppm) |

## Design:

Safety Blind Zone:



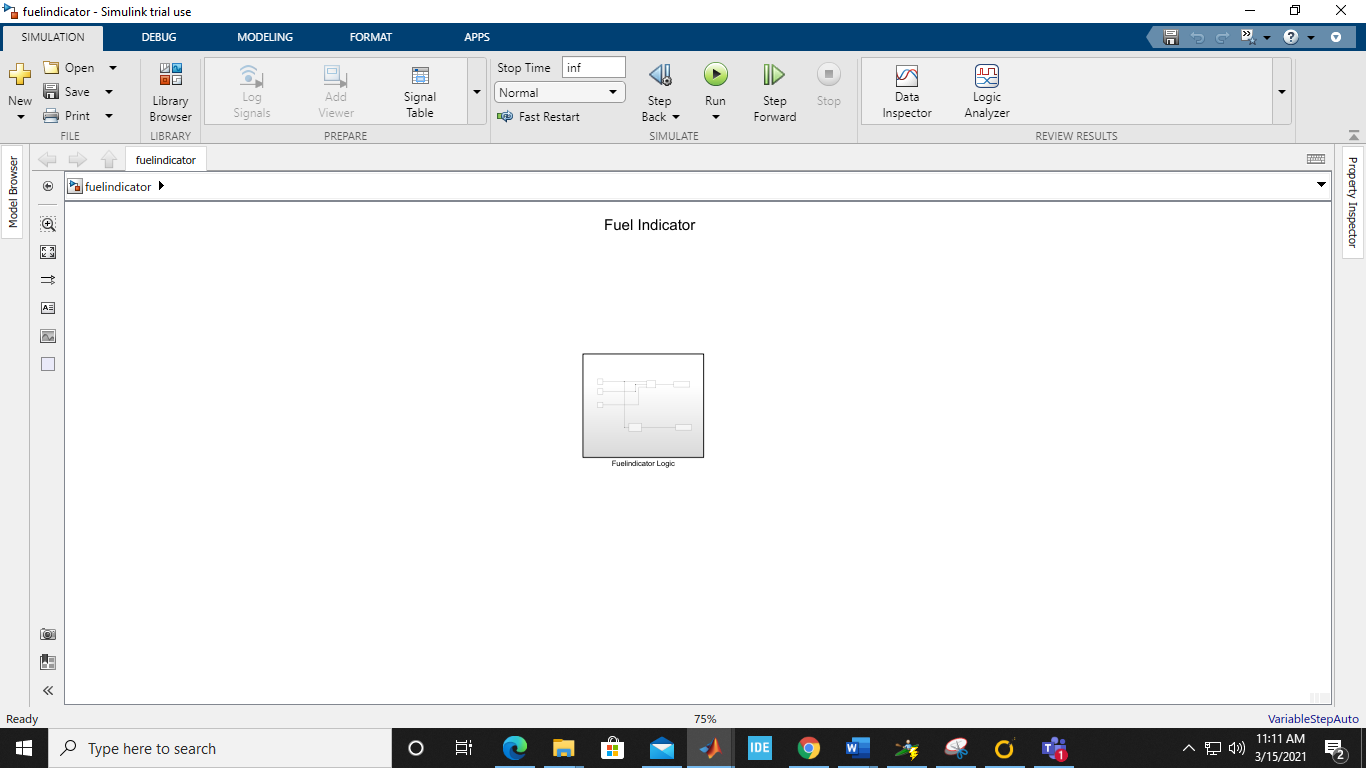
Subsystem:



Code for function block:



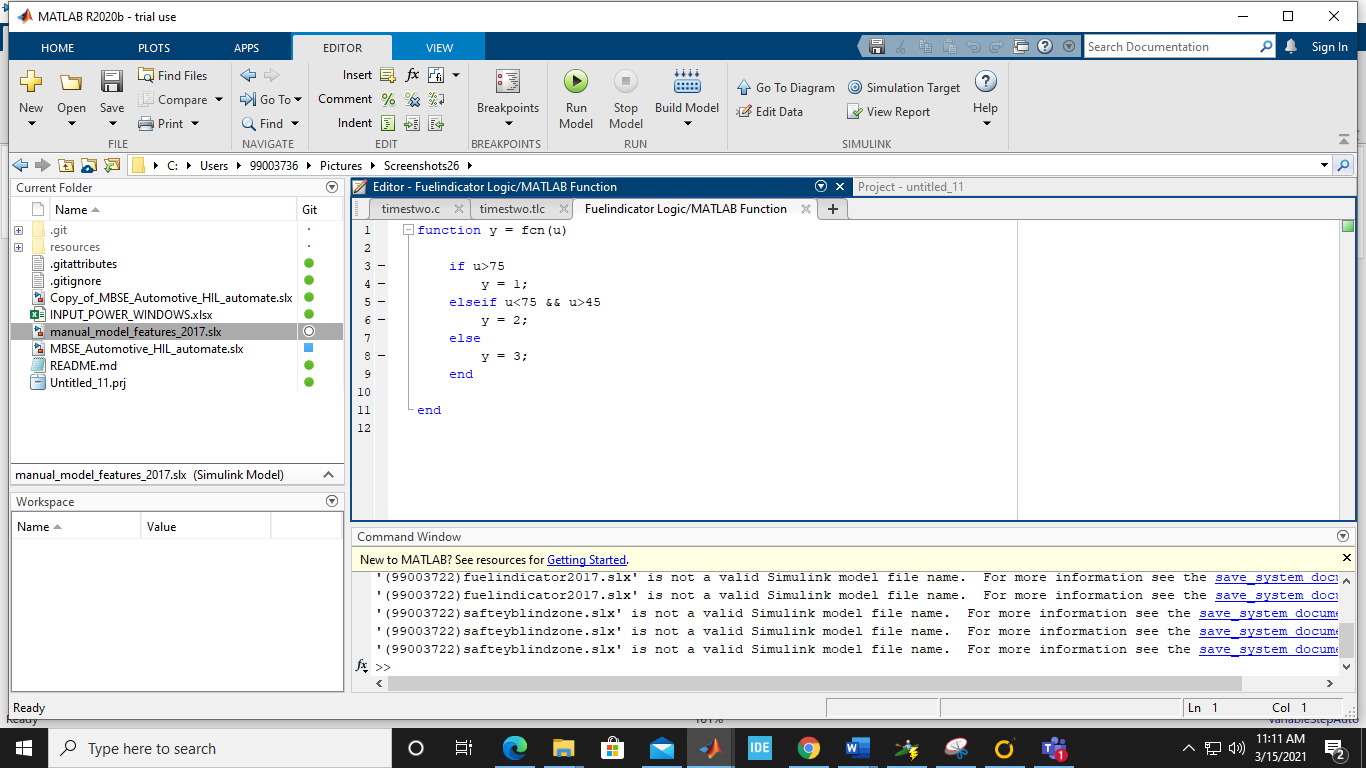
Fuel Indicator:



Sub System:



Code for function Block”



## Test Plan:

High Level Test plan:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TEST ID | DESCRIPTION | EXPECTED INPUT | ACTUAL  INPUT | EXPECTED  OUTPUT | ACTUAL  OUTPUT | TYPE OF TEST |
| HL\_SBZ\_1 | Engine is off | Ignition=off | Ignition=off | Lamp = off | Lamp = off | Boundary condition |
| HL\_SBZ\_2 | Engine is on, left sensor and right sensor both detect no input | Ignition=on | Ignition=on | Lamp=off | Lamp = off | In Range |
| HL\_SBZ\_3 | Engine is on, both sensors detect input | Ignition = on, And the object is behind the car | Ignition = on, And the object is behind the car | Lamp = on | Lamp= on | In Range |
| HL\_SBZ\_4 | Engine is on, left sensor and right sensor both detect no input | Ignition = Off, And the object is behind the car | Ignition = Off, And the object is behind the car | Lamp = off | Lamp = off | Out of bound |
| HL\_FI\_1 | Engine is off | Ignition =off | Ignition=off | Indictor=0 | Lamp = Off | Boundary condition |
| HL\_FI\_2 | Fuel value input | Fuel Value = 50 | Fuel Value = 50 | Reserve Fuel Indicator = 0 | Reserve Fuel Indicator = 0 | In Range |
| HL\_FI\_3 | Fuel Value Input | Fuel Value = 35 | Fuel Value = 35 | Reserve Fuel Indicator = 1 | Reserve Fuel Indicator = 1 | In range |
| HL\_FI\_4 | Fuel Value Input | Fuel Value=-10 | Fuel Value =-10 | Fuel Indicator =0 | Fuel Indicator =0 | Out of bound |

Low Level Test Plan:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TEST ID | DESCRIPTION | EXPECTED INPUT | ACTUAL  INPUT | EXPECTED  OUTPUT | ACTUAL  OUTPUT | TYPE OF TEST |
| LL\_SBZ\_1 | The object is 0.5m behind the car. | Sensor = 0.5m | Sensor = 0.5m | Lamp = on | Lamp = off | Out of bound |
| LL\_SBZ\_2 | The object is 1m behind the sensor | Sensor = 1m | Sensor = 1m | Lamp = on | Lamp = on | In Range |
| LL\_SBZ\_3 | The Object is 149m behind | Sensor = 149m | Sensor = 149m | Lamp = on | Lamp = on | In Range |
| LL\_SBZ\_4 | The object is 160m behind | Sensor = 160m | Sensor = 160m | Lamp = off | Lamp = off | Out of bound |
| LL\_FI\_1 | Fuel value=75(Low Density Fuel) | Fuel value=75 | Fuel Value =75 | Lamp=green | Lamp=green | In range |
| LL\_FF\_2 | Fuel Value=75  (High Density Fuel) | Fuel value=75 | Fuel Value=65 | Lamp=yellow | Lamp=green | Out of boundary |

## SharePoint Link:

<https://lnttsgroup.sharepoint.com/sites/GEA/Global%20Engineering%20Academy/GENESIS%20%20Jan%202020%20Submission/Forms/AllItems.aspx?ct=1618560455375&or=OWA%2DNT&cid=9292fe1d%2Dc175%2Dfa05%2D2acd%2D6c3f3fc60e1a&originalPath=aHR0cHM6Ly9sbnR0c2dyb3VwLnNoYXJlcG9pbnQuY29tLzpmOi9zL0dFQS9HbG9iYWwlMjBFbmdpbmVlcmluZyUyMEFjYWRlbXkvRWtMWUlHVW8xT0pCdUlpR0ZWUW5WMmtCV0ZLNkZCY1FKOTA0X0VldC15R202UT9ydGltZT1wQ2tac3E0QTJVZw&viewid=c6456521%2D0829%2D4568%2Db8d1%2D27a784675a0f&id=%2Fsites%2FGEA%2FGlobal%20Engineering%20Academy%2FGENESIS%20%20Jan%202020%20Submission%2FSubmission%2FBANGALORE%2F2002BLRGT04%2F99003722%2FMBSE>

## Summary:

Safety bind zone is the area developed by the radar zone which alert driver and activate the hazard lamp so that

The driver does not change the lane and the proximity of the sensor was calibrated properly.

Fuel Indicator is the proper conversion of fuel calibrated data to the appropriate indicated data.

# Mini Project 6

# Evolved MBSE

## Introduction:

In evolved MBSE the MBSE project is upgraded and automated script is generated for the automatic generation of the block using excel sheet and generate the graph for the respective in Simulink.

## Requirements:

High level Requirement:

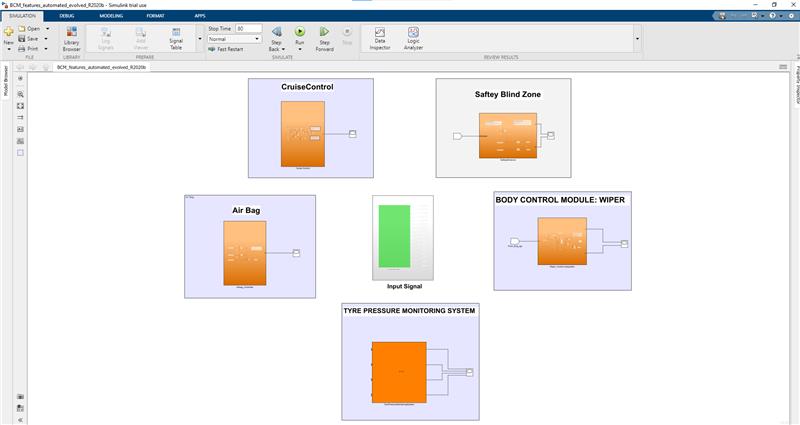
|  |  |
| --- | --- |
| High Level Requirement ID | Description |
| H1\_SBZ | Processing of all the sensor data |
| H2\_SBZ | Response should be accurate and fast |

Low level Requirement:

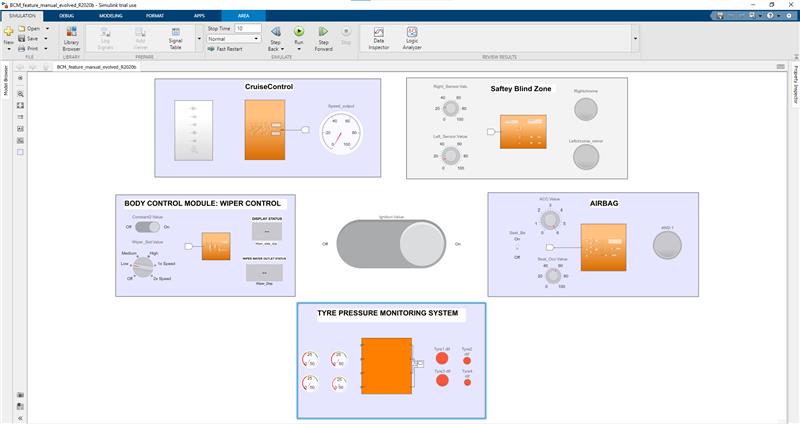
|  |  |
| --- | --- |
| Low Level Requirement ID | Description |
| L1\_SBZ | Sensor should be accurate. |
| L2\_SBZ | EPU process all the Data at a single time |

## Design:

Automated Diagram of the evolved MBSE features:



Manual Diagram of the evolved MBSE features:



## Test Plan:

High Level Test plan:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TEST ID | DESCRIPTION | EXPECTED INPUT | ACTUAL  INPUT | EXPECTED  OUTPUT | ACTUAL  OUTPUT | TYPE OF TEST |
| HL\_SBZ\_1 | Engine is off | Ignition=off | Ignition=off | Lamp = off | Lamp = off | Boundary condition |
| HL\_SBZ\_2 | Engine is on, left sensor and right sensor both detect no input | Ignition=on | Ignition=on | Lamp=off | Lamp = off | In Range |
| HL\_SBZ\_3 | Engine is on, both sensors detect input | Ignition = on, And the object is behind the car | Ignition = on, And the object is behind the car | Lamp = on | Lamp= on | In Range |
| HL\_SBZ\_4 | Engine is on, left sensor and right sensor both detect no input | Ignition = Off, And the object is behind the car | Ignition = Off, And the object is behind the car | Lamp = off | Lamp = off | Out of bound |

Low Level Test Plan:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TEST ID | DESCRIPTION | EXPECTED INPUT | ACTUAL  INPUT | EXPECTED  OUTPUT | ACTUAL  OUTPUT | TYPE OF TEST |
| LL\_SBZ\_1 | The object is 0.5m behind the car. | Sensor = 0.5m | Sensor = 0.5m | Lamp = on | Lamp = off | Out of bound |
| LL\_SBZ\_2 | The object is 1m behind the sensor | Sensor = 1m | Sensor = 1m | Lamp = on | Lamp = on | In Range |
| LL\_SBZ\_3 | The Object is 149m behind | Sensor = 149m | Sensor = 149m | Lamp = on | Lamp = on | In Range |
| LL\_SBZ\_4 | The object is 160m behind | Sensor = 160m | Sensor = 160m | Lamp = off | Lamp = off | Out of bound |

## Share Point Link:

<https://lnttsgroup.sharepoint.com/sites/GEA/Global%20Engineering%20Academy/GENESIS%20%20Jan%202020%20Submission/Forms/AllItems.aspx?ct=1618560455375&or=OWA%2DNT&cid=9292fe1d%2Dc175%2Dfa05%2D2acd%2D6c3f3fc60e1a&originalPath=aHR0cHM6Ly9sbnR0c2dyb3VwLnNoYXJlcG9pbnQuY29tLzpmOi9zL0dFQS9HbG9iYWwlMjBFbmdpbmVlcmluZyUyMEFjYWRlbXkvRWtMWUlHVW8xT0pCdUlpR0ZWUW5WMmtCV0ZLNkZCY1FKOTA0X0VldC15R202UT9ydGltZT1wQ2tac3E0QTJVZw&viewid=c6456521%2D0829%2D4568%2Db8d1%2D27a784675a0f&id=%2Fsites%2FGEA%2FGlobal%20Engineering%20Academy%2FGENESIS%20%20Jan%202020%20Submission%2FSubmission%2FBANGALORE%2F2002BLRGT04%2FProject%2FTeam%2D4%2FMBSE%20Evolved>

## Summary:

In this evolved the project using ultrasonic sensor we have shifted the scope of study to the radar sensor.

In scripting the

## Future Scope:

In future scope we can use image processing for controlling the feature of safety blind zone.

# Mini Project 7

## Intermediate C++

In intermediate c++