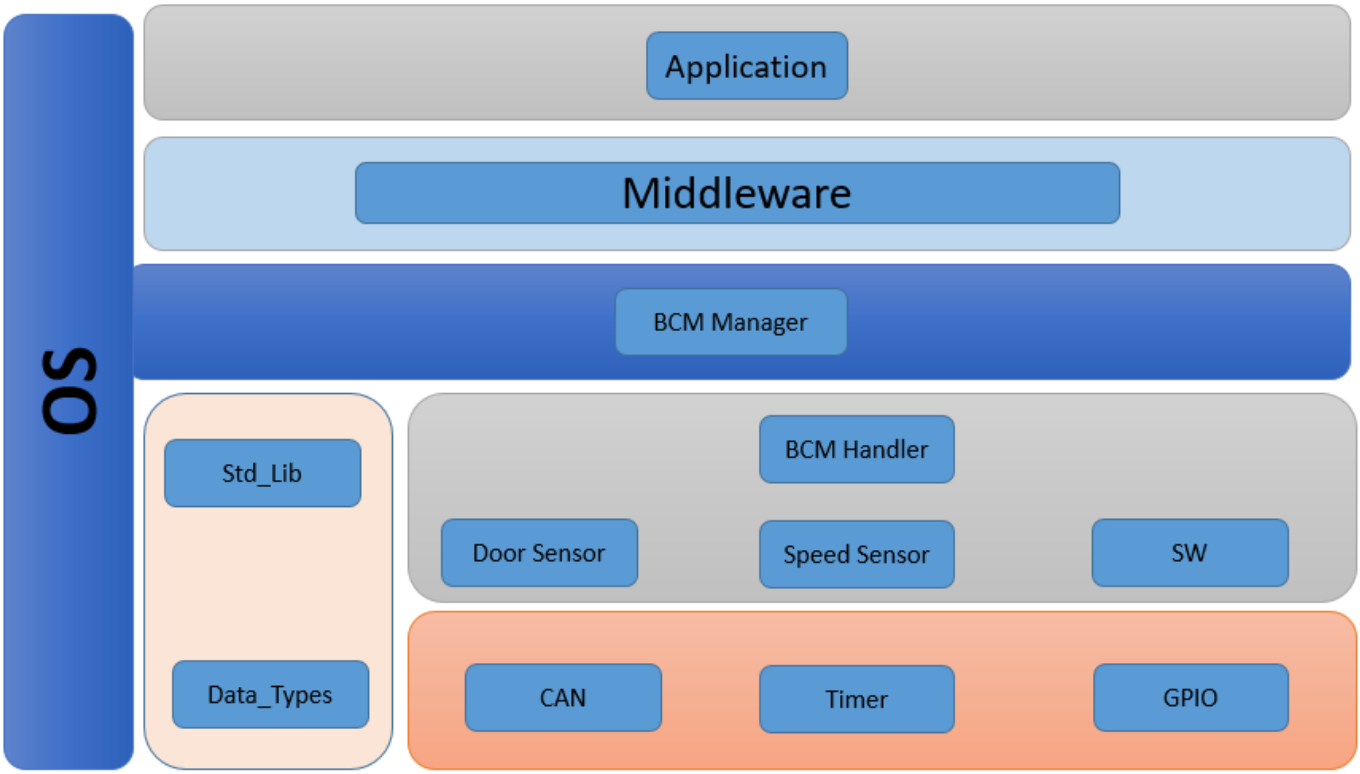
Static Design

1) For ECU 1

a. Layered Architecture for ECU 1



b. Main components and Modules
   i. Components
      1. Door Sensor
      2. Speed Sensor
      3. Light Switch
      4. CAN Unit
   ii. Modules
      1. GPIO Module
      2. Timer Module
      3. CAN Module
      4. Switch Module
      5. Door Sensor Module
      6. Speed Sensor Module

c. APIs Details
   i. Application Layer
      1. DoorSensorTask , Take Void and return Void
      2. LightSWTask , Take Void and Return Void
      3. SpeedSensorTask , Take Void and Return Void

ii. Service Layer
    1. BCM_Manger , Take Two Parameters (Data and Device ID) and return Void
iii. On Board Layer
    1. DoorSensor
        a. DSensorInit() , to init adc for using sensor
        b. DsensorRead() , used for reading value of Sensor Pin Analog Value(Door is Opened or not)
    2. LightSwitch
        a. SWInit() , to Init DIO Pins as input Pin
        b. SWRead() , to Read State of this SW (Pressed Or not)
    3. SpeedSensor
        a. SsensorInit() , to init adc for using sensor
        b. SsensorRead() , used for reading value of Sensor Pin Analog Value(Car Moving or Stooped)
iv. MCAL Layer
    1. GPIO
        a. GPIO_Init() , Init Registers values
        b. GPIO_Set_Direction() , To set Direction of pins (output or input)
        c. GPIO_Set_Value() , To Set or Clear status of any pin
    2. Timer
        a. Timer_Init() , To init Configurations of Timer needed
        b. Timer_Start() , to Start Timer
        c. Timer_Stop(), to Stop Timer at any Time
        d. Timer_Read() , To Read Value of Timer Register
    3. CAN
        a. CAN_int() , To init Configuration of Can Bus
        b. CAN_Send() To Send data to wanted node
    4. ADC
        a. ADC_Init(), To inti Registers values of ADC Peripheral
        b. ADC_ReadChannel() , Read Value of ADC Channel

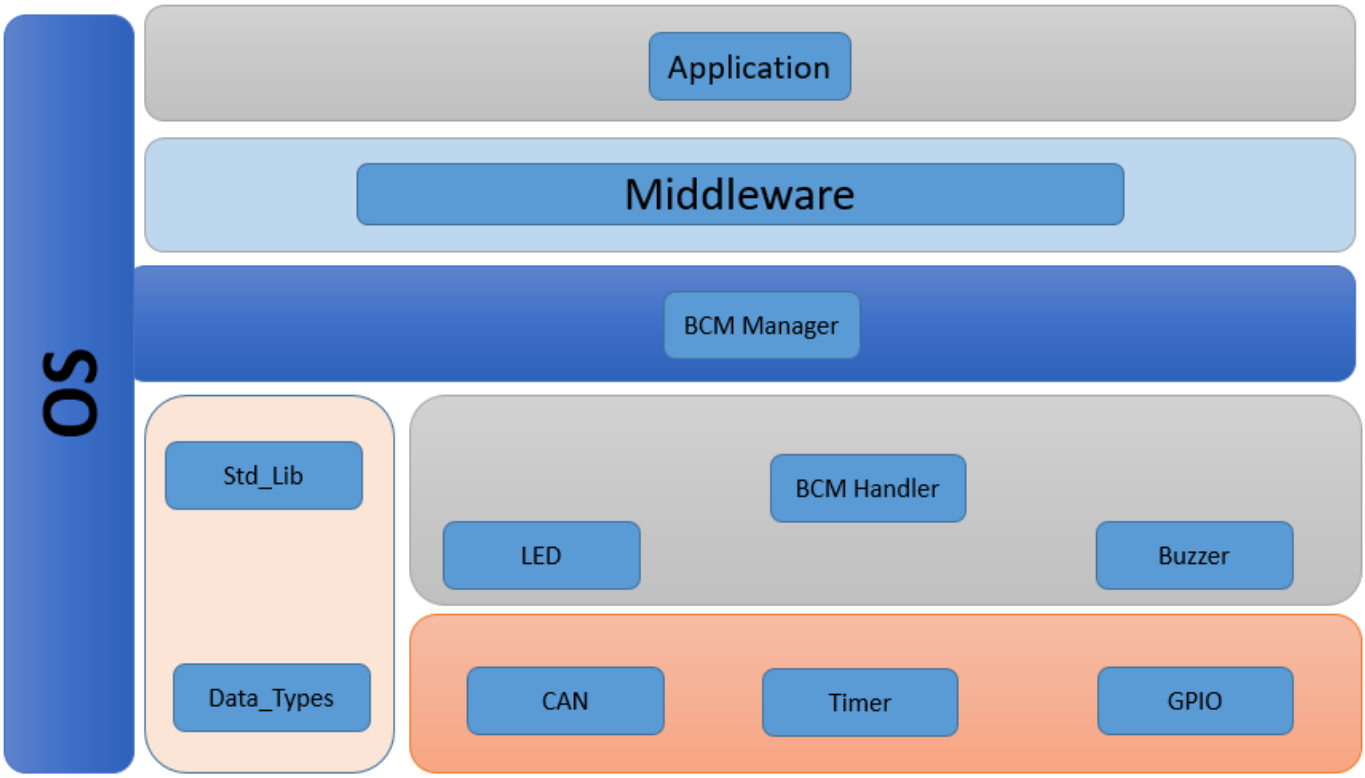
d. Folder Structure

i. Project Folder

1. Src Folder
    a. Main.c
    b. APP Folder
        i. App.c
    c. Service Folder
        i. OS.c
        ii. BCM.c
        iii. Communication_Manager.c
    d. On Board Folder
        i. Communication_Handler.c
        ii. SW.c
        iii. D_Sensor.c
        iv. S_Sensor.c
    e. MCAL Folder
        i. GPIO.c
        ii. Timer.c
        iii. CAN.c
2. Config Folder
    a. GPIO_Config.c
    b. CAN_Config.c
    c. Timer_Config.c
    d. D_Sensor_Config.c
    e. S_Sensor_Config.c
    f. SW_Config.c
    g. ADC_Config.c
3. Common Folder
    a. App.h
    b. OS.h
    c. BCM.h
    d. Communicaiton_Manager.h
    e. Communication_Handler.h
    f. SW.h
    g. D_Sensor.h
    h. S_Sensor.h
    i. GPIO.h
    j. Timer.h
    k. CAN.h

2) For ECU 1

a. Layered Architecture for ECU 1



b. Main components and Modules
   i. Components
      1. LED
      2. Buzzer
   ii. Modules
      1. GPIO Module
      2. Timer Module
      3. CAN Module
      4. LED Module
      5. Buzzer Module

c. APIs Details
   i. Application Layer
      1. DoorSensorTaskReciveState , Receive State of D Sensor
      2. SpeedSensorTaskReceiveState, Receive State of S Sensor
      3. LightTaskReceiveState , Receive State of Light Switch

   ii. Service Layer
      1. BCM_Manger , Take Two Parameters (Data and Device ID) and return Void

iii. On Board Layer
    1. LED
        a. LED_Init() , Init Led Pin as output
        b. LED_ON(), Turn On Led
        c. LED_Off(), Turn Off Led
    2. Buzzer
        a. Buzzer.Init() , Init Buzzer pin as output
        b. Buzzer_ON(), Turn Buzzer On
        c. Buzzer_Off , Turn Buzzer Off
iv. MCAL Layer
    1. GPIO
        a. GPIO_Init() , Init Registers values
        b. GPIO_Set_Direction() , To set Direction of pins (output or input)
        c. GPIO_Set_Value() , To Set or Clear status of any pin
    2. Timer
        a. Timer_Init() , To init Configurations of Timer needed
        b. Timer_Start() , to Start Timer
        c. Timer_Stop(), to Stop Timer at any Time
        d. Timer_Read() , To Read Value of Timer Register
    3. CAN
        a. CAN_int() , To init Configuration of Can Bus
        b. CAN_Send() To Send data to wanted node

d. Folder Structure
    i. Project Folder
        1. Src Folder
            a. Main.c
            b. APP Folder
                i. App.c
            c. Service Folder
                i. OS.c
                ii. BCM.c
                iii. Communication_Manager.c
            d. On Board Folder
                i. Communication_Handler.c
                ii. LED.c
                iii. Buzzer.c

    e. MCAL Folder
        i. GPIO.c
        ii. Timer.c
        iii. CAN.c
2. Config Folder
    a. GPIO_Config.c
    b. CAN_Config.c
    c. Timer_Config.c
    d. Buzzer_Config.c
3. Common Folder
    a. App.h
    b. OS.h
    c. BCM.h
    d. Communicaiton_Manager.h
    e. Communication_Handler.h
    f. LED.h
    g. Buzzer.h
    h. GPIO.h
    i. Timer.h
    j. CAN.h