**ECU 1**

Speed Sensor

Light Switch

Basic Communication

Module

OS

Common

Macros

Std Types

Door Sensor

DoorStateTask

LightSwitchStateTask

SpeedStateTask

ADC

TIMER

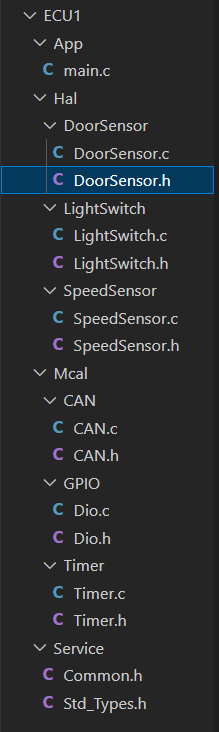
GPIO

CAN

**ECU Components and Modules:**

* **Components**:
* Door Sensor
* Speed Sensor
* Light Switch
* **Modules**
* GPIO
* ADC
* CAN
* TIMER

ECU1 APIs:

* Application Layer
  + **readDoorState**(): This API reads the current state of the door sensor.
  + **readLightSwitchState**(): This API reads the current state of the light switch.
  + **readSpeedState**(): This API reads the current speed of the vehicle.
  + **sendData**(uint32\_t data): This API sends data to ECU 2 via the CAN bus.
* Mcal
  + GPIO
    - **Dio\_SetChannelDirection** (***Dio\_PortType*** PortID, ***Dio\_ChannelType*** ChannelID): This API Set the Direction of a specific Channel in specific port.
    - **Dio\_ReadChannel**(***Dio\_PortType*** PortID, ***Dio\_ChannelType*** ChannelID):: This API return the state of specific channel in specific port.
    - **Dio\_WriteChannel**(***Dio\_PortType*** PortID, ***Dio\_ChannelType*** ChannelID):: This API Write logical high on specific channel in specific port.
    - **Port\_Init**(**const PortConfig\_Type**\* PortConfig): This API takes an array of structs that contain the Configurations of every Channel.
* ADC
* **ADC\_Init**(): This API Initializes the ADC Peripheral
* **ADC\_Read(**Dio\_ChannelType ChannelId**):** This API read the digital signal of the analog sensor that connected to a specific channel.
  + CAN which connect to the Basic Communication Module
    - **initCAN():** This API initializes the CAN bus.
    - **sendCANData(uint32\_t data):** This API sends data over the CAN bus.
    - **receiveCANData():** This API receives data from the CAN bus.
  + TIMER Connected to the OS to Handle SysTicks.

**Folder Structure**

**ECU 2**

Basic Communication

Module

TIMER

GPIO

CAN

OS

Common

Macros

Std Types

TurnOffBuzzer

TurnOnBuzzer

TurnOffLights

TurnOnLights

Lights

Buzzer

**ECU Components and Modules:**

* **Components**:
* Buzzer
* Left Light
* Right Light
* **Modules**
* GPIO
* CAN
* TIMER

ECU1 APIs:

* Application Layer
  + **turnOnLights** (): This API Turn on the Lights depending on the message coming from ECU1.
  + **turnOffLights** (): This API Turn Off the Lights depending on the message coming from ECU1.
  + **turnOnBuzzer** (): This API Turn on the Buzzer depending on the message coming from ECU1.
  + **turnOffBuzzer** (): This API Turn off the Buzzer depending on the message coming from ECU1.
  + **receiveData**(): This API receives data from ECU 1 via the CAN bus.
* Mcal
  + GPIO
    - **Dio\_SetChannelDirection** (***Dio\_PortType*** PortID, ***Dio\_ChannelType*** ChannelID): This API Set the Direction of a specific Channel in specific port.
    - **Dio\_ReadChannel**(***Dio\_PortType*** PortID, ***Dio\_ChannelType*** ChannelID):: This API return the state of specific channel in specific port.
    - **Dio\_WriteChannel**(***Dio\_PortType*** PortID, ***Dio\_ChannelType*** ChannelID):: This API Write logical high on specific channel in specific port.
    - **Port\_Init**(**const PortConfig\_Type**\* PortConfig): This API takes an array of structs that contain the Configurations of every Channel.
  + CAN which connect to the Basic Communication Module
    - **initCAN():** This API initializes the CAN bus.
    - **sendCANData(uint32\_t data):** This API sends data over the CAN bus.
    - **receiveCANData():** This API receives data from the CAN bus.
  + TIMER Connected to the OS to Handle SysTicks.
* **File Structure**

