Static Design

Layered Architecture including components and modules.

ECU 1:

APP			ECU 1 APP		
Services			Basic Communication Manager		
HAL	Door Sensor	Light Switch	Speed Sensor	Communication Handler	
MCAL	Systick	GPIO	ADC	CAN	

ECU 2:

APP		ECU 2 APP	
Services		Basic Communication Manager	
HAL	Buzzer	Light	Communication Handler
MCAL	Systick	GPIO	CAN
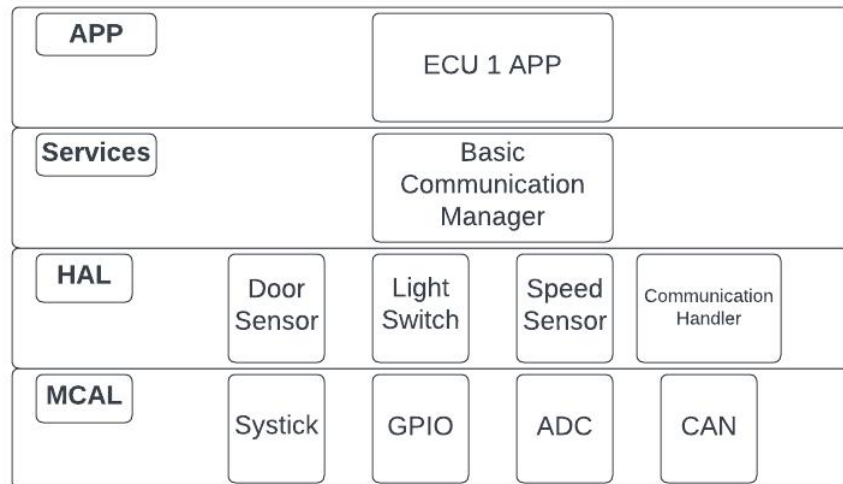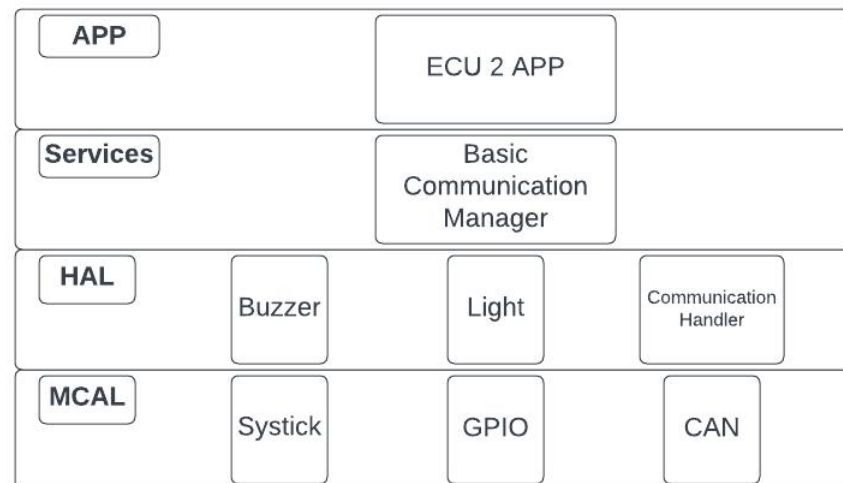
APIs

API	void GPIO_init()		
Description	Initialize the GPIO module		
Reentrancy	Non-Reentrant	Parameter	None
Sync	Synchronous	Return	None

API	Gpio_LevelType GPIO_read(Gpio_ChannelType ChannelId)		
Description	Read the required pin's level		
Reentrancy	Non-Reentrant	Parameter	Gpio_ChannelType ChannelId
Sync	Synchronous	Return	Gpio_LevelType Level
Gpio_ChannelType ChannelId : Enum with values from 1 to max number of GPIO pins, gives information about the required pin and port at the same time Gpio_LevelType Level : 8 bits, takes value of 0 or 1, defines the level to be written on the pin			

API	void GPIO_write(Gpio_ChannelType ChannelId, Gpio_LevelType Level)		
Description	Write a value to the required pin		
Reentrancy	Non-Reentrant	Parameter	Gpio_ChannelType ChannelId, Gpio_LevelType Level
Sync	Synchronous	Return	None

API	void STK_init(CLK_SOURCE)		
Description	Initialize the Systick module		
Reentrancy	Non-Reentrant	Parameter	CLK_SOURCE
Sync	Synchronous	Return	None
CLK_SOURCE : a Macro that defines predetermined clock sources of the used MCU Takes values of 1 to max num of sources.			

API	void ADC_init()	
Description	Initialize the ADC module	
Reentrancy	Non-Reentrant	**Parameter**	None
Sync	Synchronous	**Return**	None

API	Adc_ValueType Value ADC_read(Adc_ChannelType ChannelID)	
Description	Reads the required channel value	
Reentrancy	Non-Reentrant	**Parameter**	Adc_ChannelType ChannelID
Sync	Synchronous	**Return**	Adc_ValueType Value
Adc_ChannelType ChannelID : Enum with values from 1 to max number of ADC channels Adc_ValueType Value : 16 bit, takes values from 0 to 65535, defines the value read by the ADC		

API	ADC_startConversion	
Description	Starts the conversion of a channel	
Reentrancy	Non-Reentrant	**Parameter**	Adc_ChannelType ChannelID
Sync	Synchronous	**Return**	None

API	void CAN_init()	
Description	Initialize the CAN module	
Reentrancy	Non-Reentrant	**Parameter**	None
Sync	Synchronous	**Return**	None

API	void CAN_send(Can_ValueType Message, Can_IdType ID)	
Description	Sends the required message	
Reentrancy	Non-Reentrant	**Parameter**	Can_ValueType Message - Can_IdType ID
Sync	Synchronous	**Return**	None
Can_ValueType Message : 64 bit, defines the message to be sent through CAN bus Can_IdType ID : can be either 16 bit or 32 bit, actual ID size is 11 bit or 29 bit depending on the mode used, defines the ID of the message, lower ID = higher priority		

API	CAN_receive		
Description	Receives the required message		
Reentrancy	Non-Reentrant	**Parameter**	None
Sync	Synchronous	**Return**	Can_ValueType Message

API	DoorSensor_init		
Description	Initialize the Door Sensor		
Reentrancy	Non-Reentrant	**Parameter**	None
Sync	Synchronous	**Return**	None

API	LightSwitch_init		
Description	Initialize the Light Switch		
Reentrancy	Non-Reentrant	**Parameter**	None
Sync	Synchronous	**Return**	None

API	SpeedSensor_init		
Description	Initialize the Speed Sensor		
Reentrancy	Non-Reentrant	**Parameter**	None
Sync	Synchronous	**Return**	None

API	DoorSensor_status		
Description	Read status of the door sensor		
Reentrancy	Non-Reentrant	**Parameter**	None
Sync	Synchronous	**Return**	u8 status
u8 status : 8 bits, takes values of 0 or 1, defines the status of the Door (0 = closed, 1 = open)			

API	LightSwitch_status		
Description	Read status of the light switch		
Reentrancy	Non-Reentrant	**Parameter**	None

Sync	Synchronous	Return	u8 status
u8 status : 8 bits, takes values of 0 or 1, defines the status of the light switch(0 = closed, 1 = open)			

API		SpeedSensor_status	
Description		Read status of the speed sensor	
Reentrancy	Non-Reentrant	Parameter	None
Sync	Synchronous	Return	u16 speed
u16 speed : 16 bits, takes values from 0 to 65535, defines the speed read by the ADC			

API		Light_init	
Description		Initialize lights module	
Reentrancy	Non-Reentrant	Parameter	None
Sync	Synchronous	Return	None

API		Light_ON	
Description		Turn lights ON	
Reentrancy	Non-Reentrant	Parameter	None
Sync	Synchronous	Return	None

API		Light_OFF	
Description		Turn lights ON	
Reentrancy	Non-Reentrant	Parameter	None
Sync	Synchronous	Return	None

API		Buzzer_init	
Description		Initialize Buzzer module	
Reentrancy	Non-Reentrant	Parameter	None
Sync	Synchronous	Return	None

API	Buzzer_ON	
Description	Turn buzzer ON	
Reentrancy	Non-Reentrant	**Parameter**	None
Sync	Synchronous	**Return**	None

API	Buzzer_OFF	
Description	Turn Buzzer OFF	
Reentrancy	Non-Reentrant	**Parameter**	None
Sync	Synchronous	**Return**	None

API	ECU1_sendDoorStatus	
Description	Send door status to ECU2	
Reentrancy	Non-Reentrant	**Parameter**	None
Sync	Synchronous	**Return**	None

API	ECU1_sendSwitchStatus	
Description	Send door status to ECU2	
Reentrancy	Non-Reentrant	**Parameter**	None
Sync	Synchronous	**Return**	None

API	ECU1_sendSpeedStatus	
Description	Send door status to ECU2	
Reentrancy	Non-Reentrant	**Parameter**	None
Sync	Synchronous	**Return**	None

API	ECU2_receiveStatus	
Description	Send door status to ECU2	
Reentrancy	Non-Reentrant	**Parameter**	None
Sync	Synchronous	**Return**	Status