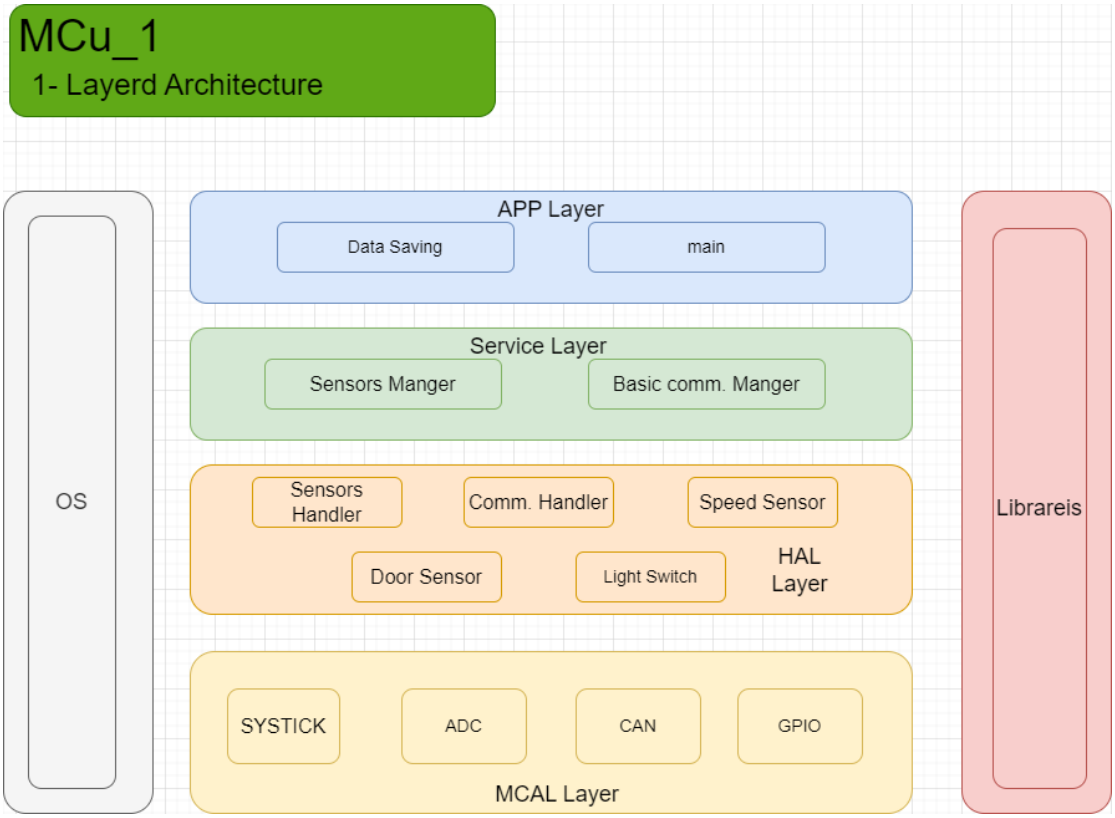


2-APIs

GPIO Module:

API	Void GPIO_init(void)		
Description	Initialize the GPIO with fixed configurations		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	void	Return	void

API	Void GPIO_write_Pin(u8 Port_Num,u8 pin_Num, U8 value);
Description	Write the required value on the specific Pin

Sync/Async	sync	Reentrancy	Non reentrant
Parameters	Port num,pin num,pin value	Return	void

API	U8 GPIO_write_Pin(u8 Port_Num,u8 pin_Num,);		
Description	Read the value from the gpio pin		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	Port num,pin num,pin value	Return	U8

CAN Module:

API	Void CAN_init(void);		
Description	Initialize the CAN with fixed configurations		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	void	Return	void

API	Void CAN_transmit(u8 CanPin_ID,u64 message);

Description	Send a required message via specific pin id		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	Can pin number,message	Return	void

Speed Sensor Module;

API	Void SpeedSensor_init(void)		
Description	Initialize the SpeedSensor with fixed configurations		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	void	Return	void

API	U16 SpeedSensor_getSpeed(void)		
Description	Get the speed from the sensor via adc		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	void	Return	U16

Door Sensor Module:

API	Void DOORSensor_init(void)		
Description	Initialize the door sensor with fixed configurations		
Sync/Async	sync	Reentrancy	Non reentrant

Parameters	void	Return	void

API	U8 DoorSensor_getStatus(void)		
Description	Read the door sensor staus		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	void	Return	U8

Light Switch Module:

API	Void LightSwitch_init(void)		
Description	Initialize the light switch with fixed configurations		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	void	Return	void

API	U8 LightSwitch_getStatus(void)		
Description	Read the light switch status		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	void	Return	U8

Sensor handler Module:

API	U32 Sensor_handler(u8 sensor_ID)		
Description	Choose which sensor to read from		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	U8 Sensor id	Return	U32

Communication handler Module:

API	Void BCM_handler(u64 handler_Message,u8 bus)		
Description	Choose which bus to read from		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	U64 message ,u8 bus	Return	void

Sensor manger Module:

API	U32 sensor_manger(u8 sensor_id)		
Description	Make the app layer to choose the sensor		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	Sensor id	Return	void

Basci Communication manger Module:

API	Void BCM_Manger(u64 Manger_Message,u8 bus)
Description	Make the app layer to choose the bus

Sync/Async	sync	Reentrancy	Non reentrant
Parameters	Message,bus	Return	void

Data Save Module:

API	U32 Data_Save(u64 data)		
Description	Save the required data		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	data	Return	void

Application Module:

API	Void SensorDoorState(void)		
Description	Send the door state via can		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	void	Return	void

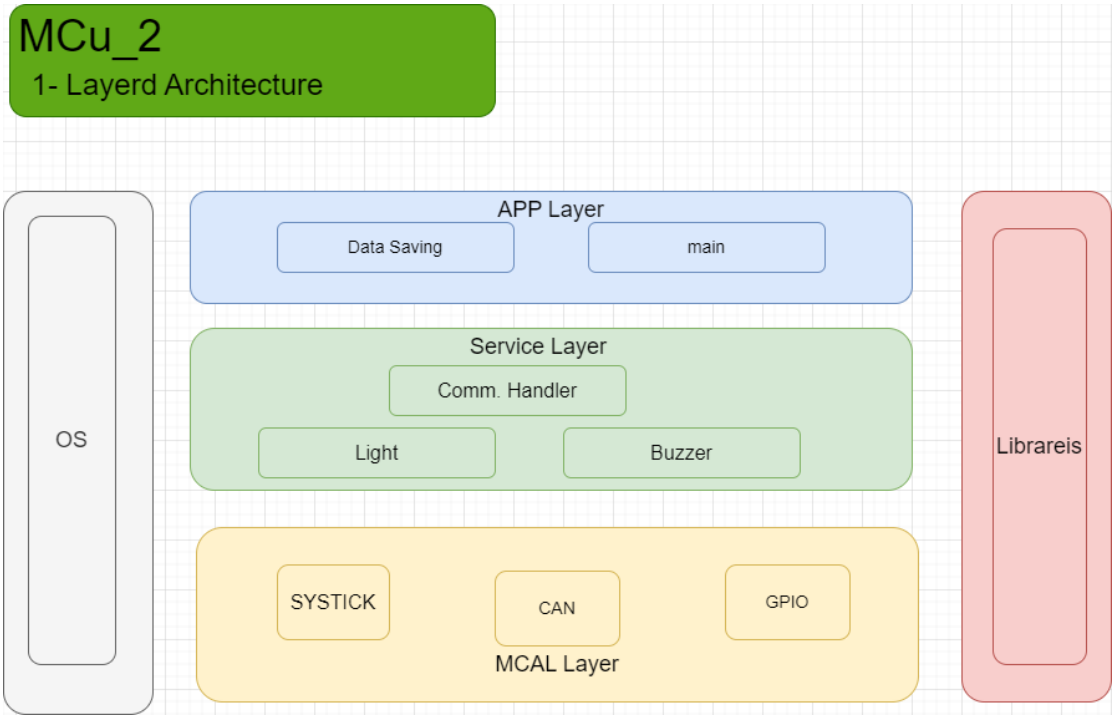
API	Void SendSpeed(void)		
Description	Send the speed sensor via can		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	void	Return	void

API	Void SendLightSwitchState(void)		
Description	Send the light switch state via can		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	void	Return	void

MCAL	HAL	Service	App
Systick.c	Sensor_handler.c	Os.c	Data_save.c
ADC.c	Comm_handler.c	Basic_comm_mngr.c	Main.c
CAN.c	Light_switch.c	Sensor_mngr.c	
GPIO.c	Door_sensor.c		
	Speed_sensor.c		

Header files folder:

systick	os	door	
Sensor_handler	Std_lib	Data_save	
Speed	CAN	Sensor_mngr	
Common_macros	switch	Comm_mngr	
ADC	memoryMap		
Comm_handler	gpio		



2-APIs

GPIO Module:

API	Void GPIO_init(void)		
Description	Initialize the GPIO with fixed configurations		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	void	Return	void

API	Void GPIO_write_Pin(u8 Port_Num,u8 pin_Num, U8 value);		
Description	Write the required value on the specific Pin		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	Port num,pin	Return	void

	num,pin value		

API	U8 GPIO_write_Pin(u8 Port_Num,u8 pin_Num,);		
Description	Read the value from the gpio pin		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	Port num,pin num,pin value	Return	U8

CAN Module:

API	Void CAN_init(void);		
Description	Initialize the CAN with fixed configurations		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	void	Return	void

API	U64 CAN_receiveu8 CanPin_ID);		
Description	Receive a required message via specific pin id		
Sync/Async	sync	Reentrancy	Non reentrant

Parameters	Can pin number	Return	U64

Buzzer Module

API	Void Buzzer_init(void)		
Description	Initialize the Buzzer switch with fixed configurations		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	void	Return	void

API	Void Buzzer_ON(void)		
Description	Set the Buzzer on		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	void	Return	void

API	Void Buzzer_OFF(void)		
Description	Set the Buzzer off		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	void	Return	void

Communication handler Module:

API	U64 BCM_handler(u8 bus)		
Description	Choose which bus to read from		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	u8 bus	Return	U64

Buzzer Module

API	Void Light_init(void)		
Description	Initialize the Light switch with fixed configurations		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	void	Return	void

API	Void Light_ON(void)		
Description	Set the Light on		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	void	Return	void

API	Void Light_OFF(void)		
Description	Set the Light off		
Sync/Async	sync	Reentrancy	Non reentrant

Parameters	void	Return	void

Data Save Module:

API	U32 Data_Save(u64 data)		
Description	Save the required data		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	data	Return	void

Application Module:

API	Void Receive_Message(void)		
Description	Receive the message from MCU_1 periodically		
Sync/Async	sync	Reentrancy	Non reentrant
Parameters	void	Return	void

3- Folder Structure

MCAL	HAL	Service	App
Systick.c	Light.c	Os.c	Data_save.c
CAN.c	Comm_handler.c	Basic_comm_mngr.c	Main.c
GPIO.c	Buzzer.c		

Header files folder:

systick	os	Light	
Buzzer	Std_lib	Comm_mngr	
Data_save	CAN	Sensor_mngr	
Common_macros	switch		
Light	memoryMap		
Comm_handler	gpio		