

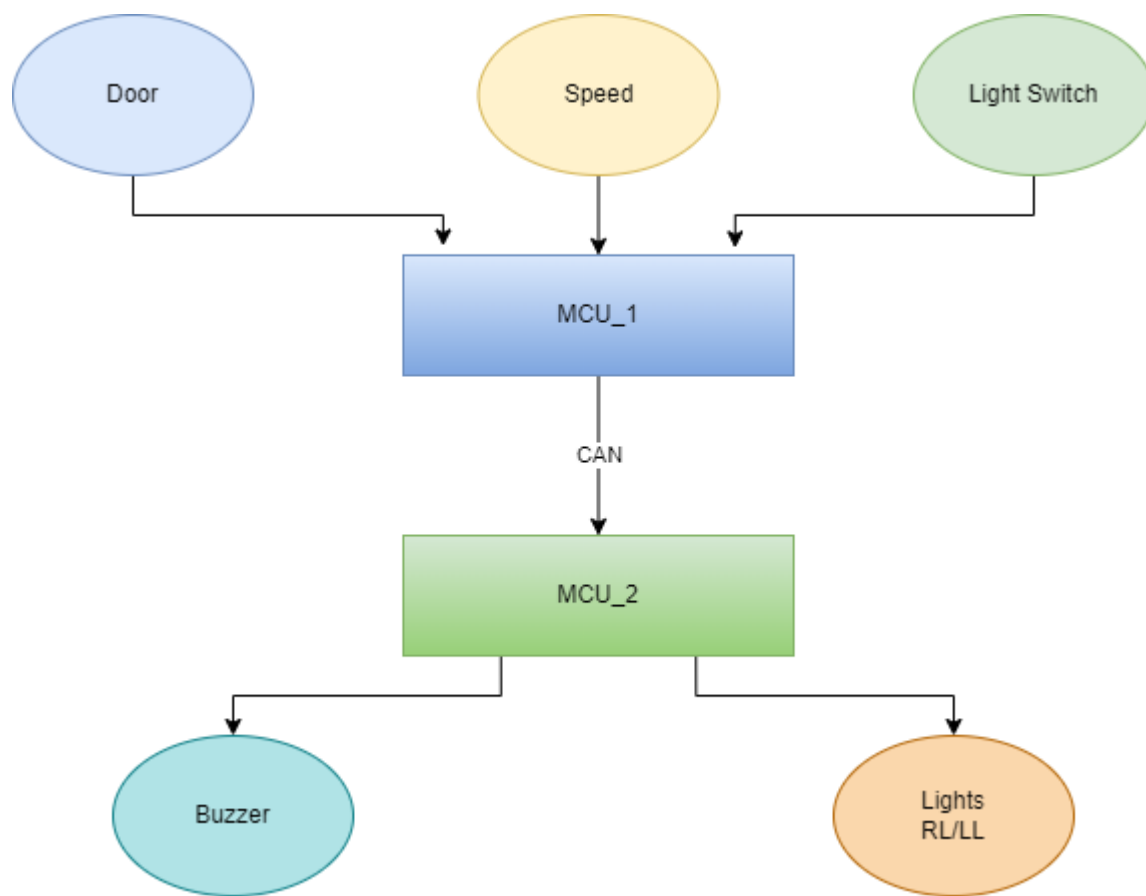
Static Design

Automotive Door Control System Design

Contents

System Overview	3
MCU 1	4
PORT API's	4
DIO API's	5
CAN API's	6
ADC API's	6
Door API's	7
Light Switch	7
Speed Sensor	7
Buzzer	7
Lights	7

System Overview



MCU 1 receives input from Door Sensor (D), Speed Sensor (S) and Light Sw (L)

Communicates with MCU 2 through CAN Protocol to send the status of its inputs periodically

Sends Door, Speed and Light switch status each 10 ms, 5 ms and 20 ms, respectively

MCU 2 output connected to Buzzer (B) and Right Light (RL) and Left light (LL)

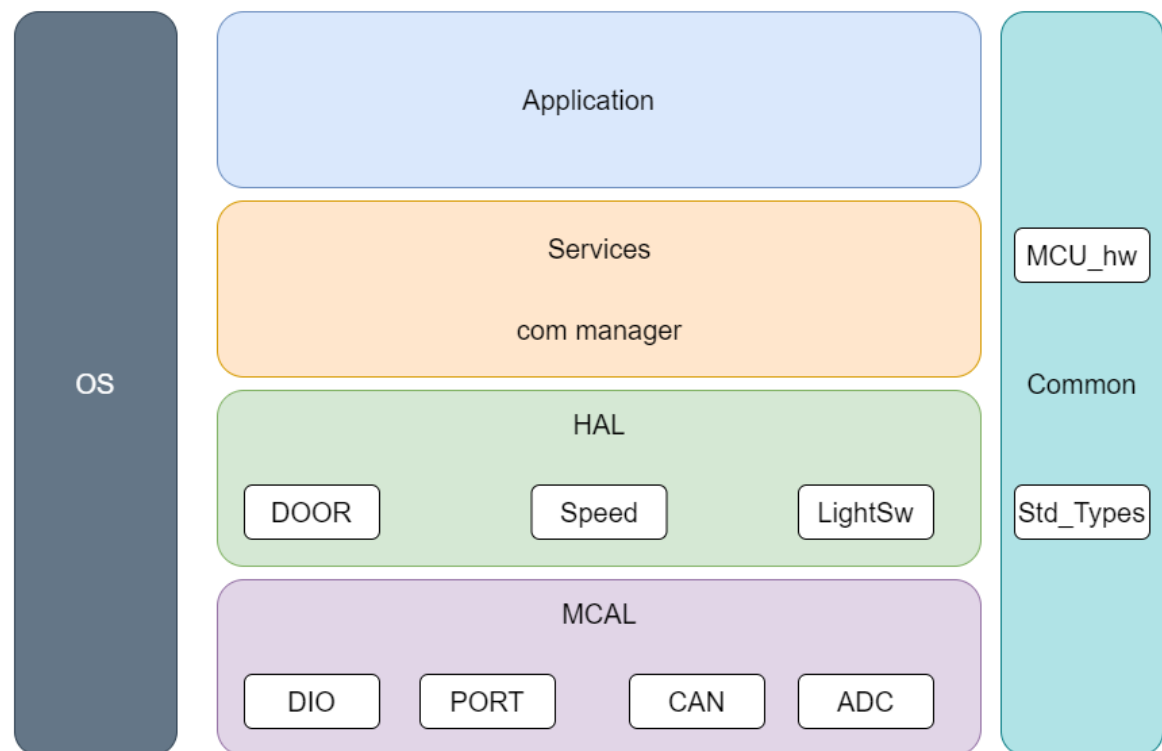
MCU 2 receives the CAN msg and responds accordingly:

If:

- 1- D open and S is not zero = B ON , Lights OFF
- 2- D open and S is zero = B OFF, Lights ON
- 3- D closed and Lights are ON = Lights switched off after 3 seconds
- 4- S is not zero and L is pressed= B OFF, Lights ON
- 5- S is zero and L is pressed = B ON, Lights ON

MCU 1

MCU_1 Layered Architecture



PORT API's

Name	Void PORT_Init(Port_ConfigType * ConfigPtr)
Arguments	Port_ConfigType * ConfigPtr
Return	Void
Description	Initialises the PORT driver with the specified configurations

Types

Name	Port_ConfigType
Type	Struct
Description	Configurations for the specified pin

Name	Port_PinModeType;
Type	Typedef enum
Range	Digital Analog
Description	Set pin mode

Name	Port_PinDirectionType;
Type	Typedef enum
Range	Input Output
Description	Set pin direction

DIO API's

Name	Void DIO_Write(Dio_ChannelType ChannelId, Dio_LevelType Level)
Arguments	Dio_ChannelType ChannelId Dio_LevelType Level
Return	Void
Description	Writes the specified level in the correct channel

Name	Dio_LevelType Dio_ReadChannel(Dio_ChannelType ChannelId);
Arguments	Dio_ChannelType ChannelId
Return	DIO_LevelType
Description	Returns the level of the specified channel

Types

Name	DIO_ChannelType
Type	Typedef enum
Range	PA0 to PF7
Description	The address of the pin

Name	Dio_LevelType
Type	Typedef enum
Range	0 is LOW 1 is HIGH
Description	The level of the pin

CAN API's

Name	<code>void CAN_Init(void);</code>
Arguments	Void
Return	Void
Description	Initialise Can module and declare a message object

Function Name	<code>Void CAN_Send (CAN_Msg Status_msg);</code>
Argument	CAN_Msg Status_msg
Return	Void
Description	Set message object and send message

Function Name	<code>UInt8_t CAN_Recieve (CAN_Msg *Status_msg);</code>
Argument	CAN_Msg *Status_msg
Return	UInt8_t can message
Description	receives a CAN message from the bus and stores it in the provided message buffer

Types

Name	<code>CAN_Msg</code>
Type	struct
Description	CAN msg structure

ADC API's

Name	<code>Void ADC_Init(ADC_ConfigType * ConfigPtr)</code>
Arguments	ADC_ConfigType * ConfigPtr
Return	Void
Description	Initialises the ADC driver with the specified configurations

Name	<code>UInt8_t ADC_Read(ADC_ChannelType Channel)</code>
Arguments	ADC_ChannelType ADC_Channel
Return	UInt8_t adc value
Description	reads the converted value from the specified channel

Types

Name	<code>ADC_ConfigType</code>
Type	Struct
Description	ADC configuration structure

Name	ADC_ChannelType
Type	Typedef enum
Description	ADC channel

Door API's

Name	Door_State DoorSensor_Read(DIO_ChannelType ChannelID)
Arguments	DIO_ChannelType ChannelID
Return	Door_State
Description	Gets the state of the specified door sensor

Light Switch

Name	LightSw_State LightSw_Read(DIO_ChannelType ChannelID)
Arguments	DIO_ChannelType ChannelID
Return	LightSW_State
Description	Gets the state of the specified light switch

Speed Sensor

Name	SpeedSensor_State SpeedSensor_Read(DIO_ChannelType ChannelID)
Arguments	DIO_ChannelType ChannelID
Return	SpeedSensor_State
Description	Gets the speed of the specified speed sensor

Buzzer

Name	void Buzzer_ON(Dio_ChannelType ChannelId)
Arguments	Dio_ChannelType ChannelId
Return	Void
Description	Sets the buzzer on

Name	void Buzzer_OFF(Dio_ChannelType ChannelId)
Arguments	Dio_ChannelType ChannelId
Return	Void
Description	Sets the buzzer off

Lights

Name	void L_ON(L_Direction L_ID)
Arguments	L_Direction L_ID
Return	Void
Description	Switch the specified light ON

Name	void L_OFF(L_Direction L_ID)
Arguments	L_Direction L_ID
Return	Void
Description	Switch the specified light OFF

MCU_2 Layered Architecture

