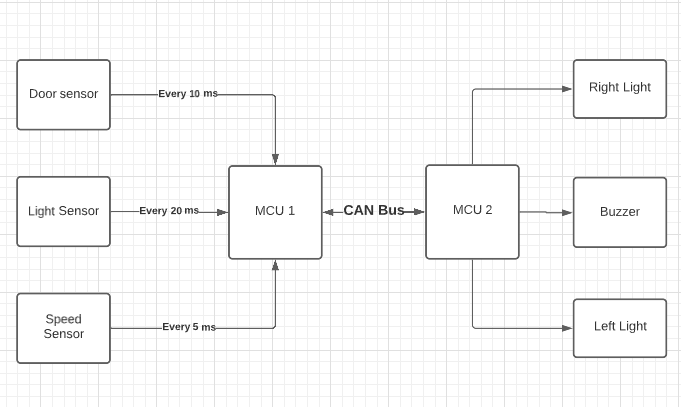
# Static Design

**Block** Diagram

****

## ECU 1

### Layered Architecture

### Modules

1. MCAL Layer

* GPIO

|  |  |
| --- | --- |
| **Name** | **void** GPIO\_Init (**struct\*** ConfigPtr) |
| **Description** | Initializes GPIO based on the given struct. |
| **Return Value** | void |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

|  |  |
| --- | --- |
| **Name** | **Uint** GPIO\_Read(**uint** Port\_no , **uint** Pin\_no) |
| **Description** | Reads the value of the given Pin |
| **Return Value** | uint |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

|  |  |
| --- | --- |
| **Name** | **void** GPIO\_Write (**uint** Port\_no , **uint** Pin\_no, **uint** Value) |
| **Description** | Writes the given value to the required pin |
| **Return Value** | void |
| **Synch** | Synchronous |
| **Reentrancy** | Reentrant |
| **Recursion** | No |

Used Typedefs

|  |  |
| --- | --- |
| **Name** | PinConfig |
| **Type** | Structure |
| **Contents** | Pin / Mode / Pin Value / Direction |
| **Description** | Struct used for the initialization of the module using given configurations |

|  |  |
| --- | --- |
| **Name** | **Void** ADC\_Init (**struct\*** ConfigPtr) |
| **Description** | Initializes ADC based on the given struct. |
| **Return Value** | void |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

Used Args

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | Port\_no | Pin\_no | Value |
| **Type** | **Uint8** | **Uint8** | **Uint8** |
| **Range** | 0 – 2^8 | 0-2^8 | 0-1 |
| **Description** | Ranges from zero to number of ports in mcu | Ranges from zero to number of pins in mcu | Value of specified pin, high or low |

* ADC

|  |  |
| --- | --- |
| **Name** | **uint** ADC\_read (**uint8** Channel) |
| **Description** | Takes input from specified channel |
| **Return Value** | uint |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

U

Used Typedefs

|  |  |
| --- | --- |
| **Name** | PinConfig |
| **Type** | Structure |
| **Description** | Struct used for the initialization of the module using given configurations |

Used Args

|  |  |
| --- | --- |
| **Name** | Channel |
| **Type** | **Uint8** |
| **Range** | 0 – 2^8 |
| **Description** | Ranges from zero to number of available channels |

* CAN

|  |  |
| --- | --- |
| **Name** | **void** CAN\_Init (**struct\*** ConfigPtr) |
| **Description** | Initializes CAN bus based on the given struct. |
| **Return Value** | void |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

|  |  |
| --- | --- |
| **Name** | **void** CAN\_Send(**uint** pin **, uint** data) |
| **Description** | Sends the given data using the specified pin |
| **Return Value** | void |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

Used Typedefs

|  |  |
| --- | --- |
| **Name** | PinConfig |
| **Type** | Structure |
| **Description** | Struct used for the initialization of the module using given configurations |

Used Args

|  |  |  |
| --- | --- | --- |
| **Name** | Pin | Data |
| **Type** | **Uint8** | **Uint32** |
| **Range** | 0-2^8 | 0-2^32 |
| **Description** | Ranges from zero to number of pins in mcu | Data we want to send on the CAN bus |

1. HAL Layer

* Door Sensor

|  |  |
| --- | --- |
| **Name** | **void** Sensor\_Init (**struct\*** ConfigPtr) |
| **Description** | Initializes Door Sensor pin using GPIO |
| **Return Value** | void |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

|  |  |
| --- | --- |
| **Name** | **Uint** Door\_Read(**uint** Port\_no , **uint** Pin\_no) |
| **Description** | Reads the value of the given Pin |
| **Return Value** | uint |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

Used Typedefs

|  |  |
| --- | --- |
| **Name** | PinConfig |
| **Type** | Structure |
| **Contents** | Pin / Mode / Pin Value / Direction |
| **Description** | Struct used for the initialization of the module using given configurations |

Used Args

|  |  |  |
| --- | --- | --- |
| **Name** | Port\_no | Pin\_no |
| **Type** | **Uint8** | **Uint8** |
| **Range** | 0 – 2^8 | 0-2^8 |
| **Description** | Ranges from zero to number of ports in mcu | Ranges from zero to number of pins in mcu |

* Light Sensor

|  |  |
| --- | --- |
| **Name** | **void** Sensor\_Init (**struct\*** ConfigPtr) |
| **Description** | Initializes Light Switch pin using GPIO |
| **Return Value** | void |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

|  |  |
| --- | --- |
| **Name** | **Uint** Light\_Read(**uint** Port\_no , **uint** Pin\_no) |
| **Description** | Reads the value of the given Pin |
| **Return Value** | uint |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

Used Typedefs

|  |  |
| --- | --- |
| **Name** | PinConfig |
| **Type** | Structure |
| **Contents** | **Uint** Pin / **uint** Mode / **bool** Pin Value / **bool** Direction |
| **Description** | Struct used for the initialization of the module using given configurations |

Used Args

|  |  |  |
| --- | --- | --- |
| **Name** | Port\_no | Pin\_no |
| **Type** | **Uint8** | **Uint8** |
| **Range** | 0 – 2^8 | 0-2^8 |
| **Description** | Ranges from zero to number of ports in mcu | Ranges from zero to number of pins in mcu |

* Speed Sensor

|  |  |
| --- | --- |
| **Name** | **void** Sensor\_Init (**struct\*** ConfigPtr) |
| **Description** | Initializes Light Switch pin using ADC |
| **Return Value** | void |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

|  |  |
| --- | --- |
| **Name** | **Uint** Speed\_Read(**uint** Port\_no , **uint** Pin\_no) |
| **Description** | Reads the value of the given Pin |
| **Return Value** | uint |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

Used Typedefs

|  |  |
| --- | --- |
| **Name** | PinConfig |
| **Type** | Structure |
| **Contents** | Pin / Mode / Pin Value / Direction |
| **Description** | Struct used for the initialization of the module using given configurations |

Used Args

|  |  |  |
| --- | --- | --- |
| **Name** | Port\_no | Pin\_no |
| **Type** | **Uint8** | **Uint8** |
| **Range** | 0 – 2^8 | 0-2^8 |
| **Description** | Ranges from zero to number of ports in mcu | Ranges from zero to number of pins in mcu |

1. Service Layer

* Comm. Manager

|  |  |
| --- | --- |
| **Name** | **void** Communication\_Handler (**struct\*** ConfigPtr) |
| **Description** | Sends the specified message through the bus specified in the struct |
| **Return Value** | void |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

Used Typedefs

|  |  |
| --- | --- |
| **Name** | Comm\_Config |
| **Type** | Structure |
| **Contents** | **Uint** message / **uint** bus |
| **Description** | Struct to hold info for Communication handler |

1. Application Layer

* App

|  |  |
| --- | --- |
| **Name** | **Uint** Light\_State(**void**) |
| **Description** | Sends value of light switch |
| **Return Value** | uint |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

|  |  |
| --- | --- |
| **Name** | **uint** Speed\_State(void) |
| **Description** | Sends value of speed sensor |
| **Return Value** | Uint |
| **Synch** | Synchronous |
| **Reentrancy** | Reentrant |
| **Recursion** | No |

|  |  |
| --- | --- |
| **Name** | **Uint** Door\_State(**void**) |
| **Description** | Sends value of door sensor |
| **Return Value** | uint |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

* Data logger

|  |  |
| --- | --- |
| **Name** | **void** Receive\_Data (**uint32** data)  Args -> data: **Range**(0-2^32), **desc :** data being received |
| **Description** | Saves the data sent to it |
| **Return Value** | Void |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |

|  |  |  |  |
| --- | --- | --- | --- |
| **MCAL** | **HAL** | **Service** | **Application** |
| GPIO.c | Door\_Sensor.c | Comm\_manager.c | Data\_logger.c |
| ADC.c | Light\_switch.c | Sensor\_manager.c | App.c |
| CAN.c | Speed\_Sensor.c |  |  |
| Systick.c |  |  |  |

### Folder Structure

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **MCAL (inc)** | **HAL(inc)** | **Service(inc)** | **Application(inc)** | **Config** |
| GPIO.h | Door\_Sensor.h | Comm\_manager.h | Data\_logger.h | GPIO\_cfg.h |
| ADC.h | Light\_switch.h | Sensor\_manager.h | App.h | ADC\_cfg.h |
| CAN.h | Speed\_Sensor.h |  |  | CAN\_cfg.h |
| Systick.h |  |  |  | Systick\_cfg.h |
|  |  |  |  | Door\_cfg.h |
|  |  |  |  | Light\_cfg.h |
|  |  |  |  | Speed\_cfg.h |

## ECU 2

### Layered Architecture

### Modules

1. MCAL Layer

* GPIO

|  |  |
| --- | --- |
| **Name** | **void** GPIO\_Init (**struct\*** ConfigPtr) |
| **Description** | Initializes GPIO based on the given struct. |
| **Return Value** | Void |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

|  |  |
| --- | --- |
| **Name** | **Uint** GPIO\_Read(**uint** Port\_no , **uint** Pin\_no) |
| **Description** | Reads the value of the given Pin |
| **Return Value** | uint |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |

|  |  |
| --- | --- |
| **Name** | **void** GPIO\_Write (**uint** Port\_no , **uint** Pin\_no, **uint** Value) |
| **Description** | Writes the given value to the required pin |
| **Return Value** | Void |
| **Synch** | Synchronous |
| **Reentrancy** | Reentrant |
| **Recursion** | No |

Used Typedefs

|  |  |
| --- | --- |
| **Name** | PinConfig |
| **Type** | Structure |
| **Contents** | Pin / Mode / Pin Value / Direction |
| **Description** | Struct used for the initialization of the module using given configurations |

Used Args

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | Port\_no | Pin\_no | Value |
| **Type** | **Uint8** | **Uint8** | **Uint8** |
| **Range** | 0 – 2^8 | 0-2^8 | 0-1 |
| **Description** | Ranges from zero to number of ports in mcu | Ranges from zero to number of pins in mcu | Value of specified pin, high or low |

* CAN

|  |  |
| --- | --- |
| **Name** | **void** CAN\_Init (**struct\*** ConfigPtr) |
| **Description** | Initializes CAN bus based on the given struct. |
| **Return Value** | void |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

|  |  |
| --- | --- |
| **Name** | **Uint** CAN\_Receive(**uint** pin **, uint** data) |
| **Description** | Receives the given data using the specified pin |
| **Return Value** | uint |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

Used Typedefs

|  |  |
| --- | --- |
| **Name** | PinConfig |
| **Type** | Structure |
| **Description** | Struct used for the initialization of the module using given configurations |

Used Args

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | Port\_no | Pin\_no | Value |
| **Type** | **Uint8** | **Uint8** | **Uint8** |
| **Range** | 0 – 2^8 | 0-2^8 | 0-1 |
| **Description** | Ranges from zero to number of ports in mcu | Ranges from zero to number of pins in mcu | Value of specified pin, high or low |

1. HAL Layer

* Light

|  |  |
| --- | --- |
| **Name** | **void** Light\_Init (**struct\*** ConfigPtr) |
| **Description** | Initializes both light pins using GPIO |
| **Return Value** | void |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

|  |  |
| --- | --- |
| **Name** | **void** Light\_Write(**uint** Port\_no , **uint** Pin\_no, **uint** value) |
| **Description** | Writes the value to the given Pin |
| **Return Value** | void |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

Used Typedefs

|  |  |
| --- | --- |
| **Name** | PinConfig |
| **Type** | Structure |
| **Contents** | Pin / Mode / Pin Value / Direction |
| **Description** | Struct used for the initialization of the module using given configurations |

Used Args

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | Port\_no | Pin\_no | Value |
| **Type** | **Uint8** | **Uint8** | **Uint8** |
| **Range** | 0 – 2^8 | 0-2^8 | 0-1 |
| **Description** | Ranges from zero to number of ports in mcu | Ranges from zero to number of pins in mcu | Value of specified pin, high or low |

* Buzzer

|  |  |
| --- | --- |
| **Name** | **void** Buzzer\_Init (**struct\*** ConfigPtr) |
| **Description** | Initializes Buzzer pin using GPIO |
| **Return Value** | void |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

|  |  |
| --- | --- |
| **Name** | **void** Buzzer\_Write(**uint** Port\_no , **uint** Pin\_no, **uint** value) |
| **Description** | Writes the value to the given Pin |
| **Return Value** | void |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

Used Typedefs

|  |  |
| --- | --- |
| **Name** | PinConfig |
| **Type** | Structure |
| **Contents** | **Uint** Pin / **uint** Mode / **bool** Pin Value / **bool** Direction |
| **Description** | Struct used for the initialization of the module using given configurations |

Used Args

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | Port\_no | Pin\_no | Value |
| **Type** | **Uint8** | **Uint8** | **Uint8** |
| **Range** | 0 – 2^8 | 0-2^8 | 0-1 |
| **Description** | Ranges from zero to number of ports in mcu | Ranges from zero to number of pins in mcu | Value of specified pin, high or low |

1. Service Layer

* Comm. Manager

|  |  |
| --- | --- |
| **Name** | **void** Communication\_Handler (**struct\*** ConfigPtr) |
| **Description** | Receives the specified message through the bus specified in the struct |
| **Return Value** | void |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

Used Typedefs

|  |  |
| --- | --- |
| **Name** | Comm\_Config |
| **Type** | Structure |
| **Contents** | **Uint** message / **uint** bus |
| **Description** | Struct to hold info for Communication handler |

* Sensor Manager

|  |  |
| --- | --- |
| **Name** | **uint** Sensor\_Handler (**uint8** pin)  Args-> pin: **Range:** (0-2^8),  **Desc:** ranges from zero to number of pins |
| **Description** | Choose which sensor will operate |
| **Return Value** | Uint |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

1. Application Layer

|  |  |
| --- | --- |
| **Name** | **Uint** Receive(**void**) |
| **Description** | Receives values from ECU1 |
| **Return Value** | uint |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |
| **Recursion** | No |

* Data logger

|  |  |
| --- | --- |
| **Name** | **void** Receive\_Data (**uint32** data)  Args-> data: **Range:** (0-2^32),  **Desc:** data received by the data logger |
| **Description** | Saves the data sent to it |
| **Return Value** | Void |
| **Synch** | Synchronous |
| **Reentrancy** | None-Reentrant |

|  |  |  |  |
| --- | --- | --- | --- |
| **MCAL** | **HAL** | **Service** | **Application** |
| GPIO.c | Light.c | Comm\_manager.c | Data\_logger.c |
| CAN.c | Buzzer.c |  | App.c |
| Systick.c |  |  |  |

### Folder Structure

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **MCAL (inc)** | **HAL(inc)** | **Service(inc)** | **Application(inc)** | **Config** |
| GPIO.h | Light.h | Comm\_manager.h | Data\_logger.h | GPIO\_cfg.h |
| CAN.h | Buzzer.h |  |  | CAN\_cfg.h |
| Systick.h |  |  |  | Systick\_cfg.h |
|  |  |  |  | Buzzer\_cfg.h |
|  |  |  |  | Light\_cfg.h |