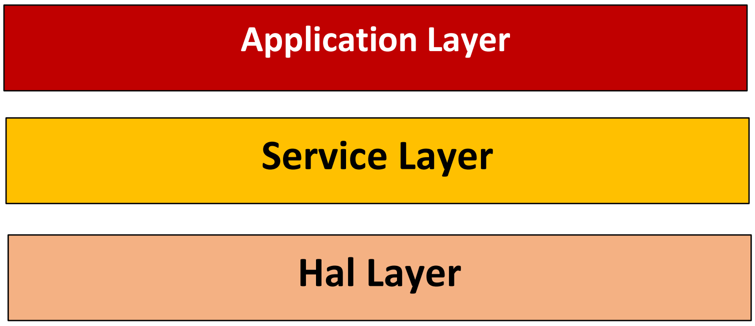
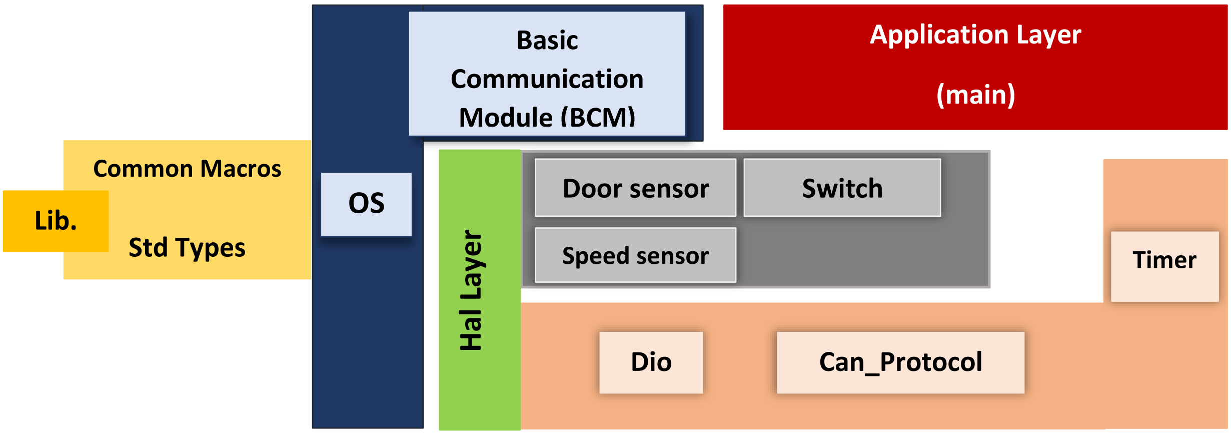
For ECU 1:

- **the layered architecture**



- **ECU components and modules**



- **full detailed APIs for each module**

Hal Layer

Dio_APIs:

- Void Dio_init(Dio_Pin y, Dio_Port z) : initialize for GPIO
- Void Dio_write(Dio_state x, Dio_Pin y, Dio_Port z) : write high or low for Gpio
- Dio_state Dio_read(Dio_Pin y, Dio_Port z) : return high or low from Gpio

Dio_ typdefs:

- Dio_state : typdef for high or low
- Dio_Pin : define number of pin
- Dio_Port: define port

Timer_APIs:

- Void Timer_init() : initialize for Timer
- Void Timer (uint_32 x) : write time value for Timer

Timer_ typdefs:

- uint_32 : typdef for long int

Can_APIs:

- Void Can_init() : initialize for Can Protocol

Door_APIs:

- Void Door_init() : initialize for Door Sensor
- Door_State Door_Read ()   : return high or low from Door Sensor

Door_ typdef:

- Door_state : typdef for high or low

Switch_APIs:

- Void Switch _init() : initialize for Switch
- Switch _State Switch _Read ()   : return high or low from Switch

Switch _ typdef:

- Switch _state : typdef for high or low

Speed_APIs:

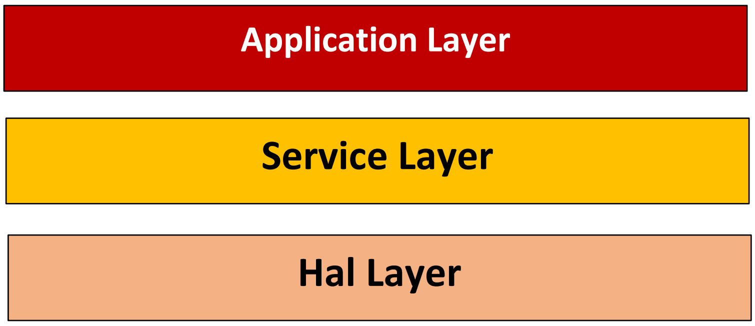
- Void Speed _init() : initialize for Speed Sensor
- Speed _State Speed _Read ()   : return high or low from Speed Sensor

Speed _ typdef:

- Speed _state : typdef for high or low

For ECU 2:

- the layered architecture

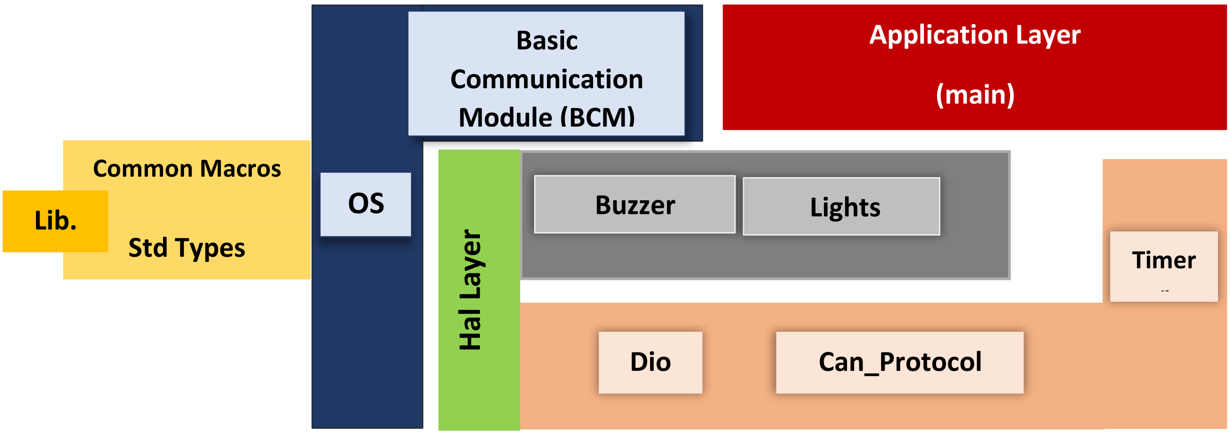


Application Layer

Service Layer

Hal Layer

- ECU components and modules



Basic Communication Module (BCM)

Application Layer

(main)

Common Macros

Lib.

Std Types

OS

Hal Layer

Buzzer

Lights

Timer

Dio

Can_Protocol

- full detailed APIs for each module

Hal Layer

Dio_APIs:

- Void Dio_init(Dio_Pin y, Dio_Port z) : initialize for GPIO
- Void Dio_write(Dio_state x, Dio_Pin y, Dio_Port z) : write high or low for Gpio
- Dio_state Dio_read(Dio_Pin y, Dio_Port z) : return high or low from Gpio

Dio_ typdefs:

- Dio_state : typdef for high or low
- Dio_Pin : define number of pin
- Dio_Port: define port

Timer_APIs:

- Void Timer_init() : initialize for Timer
- Void Timer (uint_32 x) : write time value for Timer

Timer_ typdefs:

- uint_32 : typdef for long int

Can_APIs:

- Void Can_init() : initialize for Can Protocol

Buzzer_APIs:

- Void Buzzer _init() : initialize for Buzzer Sensor
- void Buzzer_on()   : write high to Buzzer Sensor
- void Buzzer_off()   : write low to Buzzer Sensor

Rlight_APIs:

- Void Rlight _init() : initialize for Rlight
- void Rlight _on()   : write high to Rlight
- void Rlight _off()   : write low to Rlight

Llight_APIs:

- Void Llight _init() : initialize for Llight
- void Llight _on()   : write high to Llight
- void Llight _off()   : write low to Llight