



```
typedef enum
{
    E_VALID,
    E_INVALID
}E_STAT;
```

- MCAL APIs:

- DIO APIs:

```
E_STAT DIO_init(ST_DIO_config_t* configurations);
E_STAT DIO_write(uint8_t port,EN_pins pin,uint8_t data);
E_STAT DIO_read(uint8_t port,EN_pins pin,uint8_t* data);
E_STAT DIO_toggle(uint8_t port,EN_pins pin);|
```

- PWM APIs:

```
E_STAT PWM_init(ST_PWM_config_t* configurations);
E_STAT PWM_start(EN_frequency_t frequency, EN_duty_t dutycycle);
E_STAT PWM_stop(void);|
```

- TIMER APIs:

```
E_STAT TIMER_init(ST_TIMER_config_t* configurations);  
E_STAT TIMER_start(uint64_t ticks);  
E_STAT TIMER_read(uint8_t* value);  
E_STAT TIMER_set(uint8_t value);  
E_STAT TIMER_checkstatus(uint8_t* status);|
```

- ECUAL APIs:

- Control Buttons APIs:

```
E_STAT BUTTON_init(EN_BUTTON_config_t* configutaions);  
E_STAT BUTTON_checkstatus(uint8_t* status);|
```

- Motor APIs:

```
E_STAT MOTOR_init(EN_MOTOR_direction_t direction);  
E_STAT MOTOR_stop(void);
```

- Application APIs:●

```
E_STAT CAR_init(void);  
E_STAT CAR_update(void);
```