**==> SET PIN DIRECTION**

Description: set pin Direction

Inputs : u8 u8PinNB : pin Number

: u8 u8Direction : if it DIO\_INPUT\_LOW mean that this pin is input , if it DIO\_INPUT\_HOGH mean that this pin is output

Outputs: Error state

u8 **DIO\_u8SetPinDirection**(u8 u8PinNB,u8 u8Direction);

No Error

Error

Check Input Error

ASSIGN\_BIT(BIT\_NUMBER,DIRECTION)

**DIO\_u8SetPinDirection(u8,u8)**

**==> SET PIN VALUE**

Description: set pin value

Inputs : u8 u8PinNB : pin Number

: u8 u8Value : if it DIO\_LOW mean that this pin is low , if it DIO \_HOGH mean that this pin is high

Outputs: Error state

u8 **DIO\_u8SetPinValue**(u8 u8PinNB,u8 u8Value);

No Error

Error

Check Input Error

ASSIGN\_BIT(BIT\_NUMBER,Value)

**DIO\_u8SetPinValue(u8,u8)**

**==> GET PIN VALUE**

Description: GET pin value

Inputs : u8 u8PinNB : pin Number

: u8\* u8Value : pointer to value which it may be high (1) or low (0)

Outputs: Error state

Outputs: Error state

u8 **DIO\_u8GetPinValue**(u8 u8PinNB,u8\* u8Value);

No Error

Error

Check Input Error

GET\_BIT(BIT\_NUMBER,Value)

**DIO\_u8GetPinValue(u8,u8\*)**

**==> SET MOTOR SPEED AND DIRECTION**

Description: SET MOTOR SPEED AND DIRECTION

Inputs : u8 u8Speed : value to control the speed

: u8 u8Dirctio: to set the direction of the motor (anticlockwise) or (clockwise)

Outputs: Error state

u8 **MOTOR\_u8ControlSpeedDirection**(u8 u8Dirctio,u8 u8Speed);

No Error

Check Input Error

**DIO\_u8SetPinValue** (BIT\_NUMBER,DIRECTION)

**MOTOR\_u8ControlSpeedDirection (u8,u8)**

Error

No Error

Error

PWM\_u8SetSpeed(Speed)

**==> MOVE FORWARD**

Description: set two motor to move in same speed and the same direction (clockwise)

Inputs : u8 u8Speed : value to control the speed

Outputs: Error state

u8 **CAR\_u8MoveForward**(u8 u8Speed);

No Error

Check Input Error

**CAR\_u8MoveForward (u8)**

**MOTOR\_u8ControlSpeedDirection\_M1**(clockwise,Speed)

Error

No Error

Error

**MOTOR\_u8ControlSpeedDirection\_M2**(clockwise,Speed)

No Error

Error

**==> MOVE BACK**

Description: set two motor to move in same speed and the same direction (Anticlockwise)

Inputs : u8 u8Speed : value to control the speed

Outputs: Error state

u8 **CAR\_u8MoveBcak**(u8 u8Speed);

No Error

Check Input Error

**CAR\_u8MoveBack (u8)**

**MOTOR\_u8ControlSpeedDirection\_M1**(Anticlockwise,Speed)

Error

No Error

Error

**MOTOR\_u8ControlSpeedDirection\_M2**(Anticlockwise,Speed)

No Error

Error

**==> MOVE RIGHT**

Description: set each motor with different speed in same direction M1(Right) M2(left)

Inputs : u8 u8Speed : value to control the speed

Outputs: Error state

u8 **CAR\_u8MoveRight**(u8 u8Speed);

No Error

Check Input Error

**CAR\_u8MoveRight (u8)**

**MOTOR\_u8ControlSpeedDirection\_M1**(clockwise,halfSpeed)

Error

No Error

Error

**MOTOR\_u8ControlSpeedDirection\_M2**(clockwise,Speed)

No Error

Error

**==> MOVE LEFT**

Description: set each motor with different speed in same direction M1(Right) M2(left)

Inputs : u8 u8Speed : value to control the speed

Outputs: Error state

u8 **CAR\_u8MoveLeft**(u8 u8Speed);

No Error

Check Input Error

**CAR\_u8MoveLeft (u8)**

**MOTOR\_u8ControlSpeedDirection\_M1**(clockwise,Speed)

Error

No Error

Error

**MOTOR\_u8ControlSpeedDirection\_M2**(clockwise,halfSpeed)

No Error

Error

**==> STOP**

Description: set each motor with speed zero to stop the car

Inputs : u8 u8Speed : value to control the speed

Outputs: Error state

u8 **CAR\_u8Stop**(u8 u8Speed);

No Error

Check Input Error

**CAR\_u8Stop (u8)**

**MOTOR\_u8ControlSpeedDirection\_M1**(clockwise,zero)

Error

No Error

Error

**MOTOR\_u8ControlSpeedDirection\_M2**(clockwise,zero)

No Error

Error