# 割草机底盘函数接口说明

## MOTOR\_driver.c函数接口

**#define MOTOR\_SWITCH\_DEFAULT (-1)//--电机选择错误**

**#define OUTPUT\_ENABLE\_DEFAULT (-1)//--输出使能错误**

**#define MOTOR\_FWD\_REV\_SELECT\_DEFAULT (-1)//--电机转向错误**

**#define MOTOR\_BRAKE\_DEFAULT (-1)//--电机刹车错误**

**#define LEFT\_MOTOR (0)//--左电机**

**#define RIGHT\_MOTOR (1)//--右电机**

**#define MOWING\_MOTOR (2) //--割草电机**

**#define WORK\_MODE\_NORMAL (0) //--正常模式**

**#define WORK\_MODE\_OUTPUT\_DISABLE (1)//--无输出模式**

**#define WORK\_MODE\_BRAKE\_HARD (2)//--刹车模式**

**#define BRAKE\_ENABLE (0)//--刹车使能**

**#define BRAKE\_DISABLE (1)//--刹车禁能**

**#define MOTOR\_FWD (0)//--正转**

**#define MOTOR\_REV (1)//--反转**

**#define OUTPUT\_ENABLE (0)//--输出使能**

**#define OUTPUT\_DISABLE (1)//--输出禁能**

### 1、函数接口

**typedef struct \_motor\_function\_interface{**

**voidt(\*motor\_driver\_init)(uint8\_t motor\_switch,uint8\_t fwd\_rev\_flag);**

**int (\*motor\_speed\_pwm\_duty\_cycle\_set)(uint8\_t motor\_switch,uint16\_t value);**

**int (\*motor\_fwd\_rev\_select)(uint8\_t motor\_switch, uint8\_t flag);**

**int (\*get\_motor\_station)(uint8\_t motor\_switch,uint16\_t \*motor\_station);**

**int (\*motor\_driver\_work\_mode)(uint8\_t motor\_switch,uint8\_t workMode);**

**}ST\_motor\_handle,\*ST\_motor\_handle\_p;**

**extern ST\_motor\_handle motor\_function\_interface\_handle;**

**extern ST\_motor\_handle\_p motor\_function\_interface\_handle\_p;**

### 2、函数说明

**void(\*motor\_driver\_init)(uint8\_t motor\_switch,uint8\_t fwd\_rev\_flag);**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Function name: motorDriverOutputEnable

\*\* Descriptions: 电机驱动器输出使能控制

\*\* input parameters: motor\_switc：电机选择，可以取一下值：

\*\* LEFT\_MOTOR ： 左侧电机

\*\* RIGHT\_MOTOR： 右侧电机

\*\* MOWING\_MOTOR：割草电机

\*\* fwd\_rev\_flag：电机正反转标志，可取以下值：

\*\* MOTOR\_FWD：正转————顺时针

\*\* MOTOR\_REV：倒转————逆时针

\*\* Returned value: 无

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**int (\*motor\_speed\_pwm\_duty\_cycle\_set)(uint8\_t motor\_switch,uint16\_t value);**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Function name: Motor\_speed\_pwm\_duty\_cycle\_set

\*\* Descriptions: 转速占空比调节

\*\* input parameters: motor\_switc：电机选择，可以取一下值：

\*\* LEFT\_MOTOR ： 左侧电机

\*\* RIGHT\_MOTOR： 右侧电机

\*\* MOWING\_MOTOR：割草电机

\*\* value: 脉冲宽度 0-2500

\*\* Returned value: MOTOR\_SWITCH\_DEFAULT

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**int (\*motor\_fwd\_rev\_select)(uint8\_t motor\_switch, uint8\_t flag);**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Function name: Motor\_Fwd\_Rev\_Select

\*\* Descriptions: 电机正反转选择

\*\* input parameters: motor\_switc：电机选择，可以取一下值：

\*\* LEFT\_MOTOR ： 左侧电机

\*\* RIGHT\_MOTOR： 右侧电机

\*\* MOWING\_MOTOR：割草电机

\*\* flag：电机正反转选择标志，可取以下值：

\*\* MOTOR\_FWD：正转————顺时针

\*\* MOTOR\_REV：倒转————逆时针

\*\* Returned value: MOTOR\_SWITCH\_DEFAULT

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**int (\*get\_motor\_station)(uint8\_t motor\_switch,uint16\_t \*motor\_station);**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Function name: Get\_Motor\_Station

\*\* Descriptions: 获取电机状态

\*\* input parameters: motor\_switc：电机选择，可以取一下值：

\*\* LEFT\_MOTOR ： 左侧电机

\*\* RIGHT\_MOTOR： 右侧电机

\*\* MOWING\_MOTOR：割草电机

\*\* motor\_station[0]

\*\* xxxx xxxx

\*\* bit 0 : 0-输出禁能 1-输出使能

\*\* bit 1 : 0-后退 1-前进

\*\* bit 2 : 0-刹车禁能 1-刹车使能

\*\* bit 3-4: 00-正常模式 01-无输出模式 10-紧急刹车模式

\*\* bit 5-15 : 保留

\*\* motor\_station[1]

\*\* 电机转速占空比

\*\* Returned value: MOTOR\_SWITCH\_DEFAULT

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**int (\*motor\_driver\_work\_mode)(uint8\_t motor\_switch,uint8\_t workMode);**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Function name: Motor\_Driver\_Work\_Mode

\*\* Descriptions: 设置电机驱动芯片的工作模式

\*\* input parameters: motor\_switc：电机选择，可以取一下值：

\*\* LEFT\_MOTOR ： 左侧电机

\*\* RIGHT\_MOTOR： 右侧电机

\*\* MOWING\_MOTOR：割草电机

\*\* workMode ：电机驱动器工作模式选择，可取以下值：

\*\* WORK\_MODE\_NORMAL ：正常模式

\*\* WORK\_MODE\_OUTPUT\_DISABLE ：无输出模式

\*\* WORK\_MODE\_BRAKE\_HARD ：紧急刹车模式

\*\* Returned value: MOTOR\_SWITCH\_DEFAULT

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

## speed.c函数接口

**/\*===========宏定义==============\*/**

**#define SPEED\_SIZE 15 //--缓存大小**

**#define REDUCTION\_RATIO (125) //--减速比**

### 1、函数接口

**typedef struct \_speed\_function\_interface{**

**void (\*speed\_init)(void);**

**int32\_t (\*refresh\_speed)(uint8\_t motor\_switch);**

**int32\_t (\*read\_speed)(uint8\_t motor\_switch);**

**}ST\_speed\_function\_interface\_handle,\*ST\_speed\_function\_interface\_handle\_p;**

**extern ST\_speed\_function\_interface\_handle speed\_function\_interface\_handle;**

**extern ST\_speed\_function\_interface\_handle\_p speed\_function\_interface\_handle\_p**;

### 函数说明

**void (\*speed\_init)(void);**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Function name: Speed\_Init

\*\* Descriptions: 关闭外部中断，开启计数器

\*\*input parameters: 无

\*\* Returned value: 无

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**int32\_t (\*refresh\_speed)(uint8\_t motor\_switch);**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Function name: Refresh\_Speed

\*\* Descriptions: 更新速度

\*\* input parameters: motor\_switc：电机选择，可以取一下值：

\*\* LEFT\_MOTOR ： 左侧电机

\*\* RIGHT\_MOTOR： 右侧电机

\*\* MOWING\_MOTOR：割草电机

\*\* Returned value: 0

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**int32\_t (\*read\_speed)(uint8\_t motor\_switch);**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Function name: Read\_Speed

\*\* Descriptions: 速度读取

\*\* input parameters: motor\_switc：电机选择，可以取一下值：

\*\* LEFT\_MOTOR ： 左侧电机

\*\* RIGHT\_MOTOR： 右侧电机

\*\* MOWING\_MOTOR：割草电机

\*\* Returned value: 速度

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

## pid.c函数接口

**/\*===========宏定义==============\*/**

**//--限幅输出**

**#define max(a, b) (a>b? a:b)**

**#define min(a, b) (a<b? a:b)**

**#define range(x, a, b) (min(max(x, a), b))**

### 函数接口

**typedef struct \_pid\_function\_interface{;**

**uint32\_t (\*wheel\_integral\_separation\_pid)(uint8\_t motor\_switch);**

**uint16\_t (\*car\_integral\_separation\_pid)(int32\_t yaw\_err);**

**void (\*set\_pid\_speed)(uint8\_t motor\_switch,uint16\_t to\_speed);**

**}ST\_pid\_function\_interface\_handle,\*ST\_pid\_function\_interface\_handle\_p;**

**extern ST\_pid\_function\_interface\_handle pid\_function\_interface\_handle;**

**extern ST\_pid\_function\_interface\_handle\_p pid\_function\_interface\_handle\_p;**

### 函数说明

**uint32\_t (\*wheel\_integral\_separation\_pid)(uint8\_t motor\_switch);**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Function name: wheel\_Integral\_Separation\_Pid

\*\* Descriptions: 轮子调速积分分离PID算法实现

\*\* input parameters: motor\_switc：电机选择，可以取一下值：

\*\* LEFT\_MOTOR ： 左侧电机

\*\* RIGHT\_MOTOR： 右侧电机

\*\* MOWING\_MOTOR：割草电机

\*\* to\_speed : 目标速度

\*\* Returned value: pwm:PWM占空比

\*\*pid.Kp\*pid.err+pid.Ki\*pid.integral+pid.Kd\*(pid.err-pid.err\_last)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**uint16\_t (\*car\_integral\_separation\_pid)(int32\_t yaw\_err);**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Function name: Car\_Integral\_Separation\_Pid

\*\* Descriptions: 小车走直线积分分离PID算法实现

\*\* input parameters: motor\_switc：电机选择，可以取一下值：

\*\* LEFT\_MOTOR ： 左侧电机

\*\* RIGHT\_MOTOR： 右侧电机

\*\* MOWING\_MOTOR：割草电机

\*\* to\_speed : 目标速度

\*\* Returned value: pwm:PWM占空比

\*\*pid.Kp\*pid.err+pid.Ki\*pid.integral+pid.Kd\*(pid.err-pid.err\_last)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

## wheel.c函数接口

### 函数接口

**typedef struct \_wheel\_function\_interface{**

**uint32\_t (\*wheel\_go)(uint8\_t motor\_switch,uint16\_t speed);**

**uint32\_t (\*wheel\_back)(uint8\_t motor\_switch,uint16\_t speed);**

**uint32\_t (\*wheel\_stop)(uint8\_t motor\_switch,uint8\_t workMode);**

**}ST\_wheel\_function\_interface\_handle,\*ST\_wheel\_function\_interface\_handle\_p;**

**extern ST\_wheel\_function\_interface\_handle wheel\_function\_interface\_handle;**

**extern ST\_wheel\_function\_interface\_handle\_p wheel\_function\_interface\_handle\_p;**

### 函数说明

**uint32\_t (\*wheel\_go)(uint8\_t motor\_switch,uint16\_t speed);**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Function name: Wheel\_Go

\*\* Descriptions: 轮子前进

\*\* input parameters: motor\_switc：电机选择，可以取一下值：

\*\* LEFT\_MOTOR ： 左侧电机

\*\* RIGHT\_MOTOR： 右侧电机

\*\* MOWING\_MOTOR：割草电机

\*\* Returned value: 0

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**uint32\_t (\*wheel\_back)(uint8\_t motor\_switch,uint16\_t speed);**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Function name: Wheel\_Back

\*\* Descriptions: 轮子后退

\*\* input parameters: motor\_switc：电机选择，可以取一下值：

\*\* LEFT\_MOTOR ： 左侧电机

\*\* RIGHT\_MOTOR： 右侧电机

\*\* MOWING\_MOTOR：割草电机

\*\* Returned value: 0

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**uint32\_t (\*wheel\_stop)(uint8\_t motor\_switch,uint8\_t workMode);**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Function name: Wheel\_Stop

\*\* Descriptions: 轮子停止

\*\* input parameters: motor\_switc：电机选择，可以取一下值：

\*\* LEFT\_MOTOR ： 左侧电机

\*\* RIGHT\_MOTOR： 右侧电机

\*\* MOWING\_MOTOR：割草电机

\*\* workMode ：电机驱动器工作模式选择，可取以下值：

\*\* WORK\_MODE\_OUTPUT\_DISABLE ：自由停车

\*\* WORK\_MODE\_BRAKE\_HARD ：紧急刹车模式

\*\* Returned value: 0

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

## car.c函数接口

**/\*===========宏定义==============\*/**

**#define CAR\_LEFT (1) //--左转**

**#define CAR\_RIGHT (2) //--右转**

**#define CAR\_FWD (1) //--前进**

**#define CAR\_REV (2) //--后退**

**#define WHEEL\_SPACING 200 //--车轮半径，mm单位**

### 函数接口

**typedef struct \_car\_function\_interface{**

**void (\*car\_straight\_line)(uint8\_t fwd\_rev,uint16\_t speed);**

**void (\*car\_turn)(uint8\_t fwd\_rev,uint8\_t turn\_left\_right,int32\_t car\_r,uint16\_t speed);**

**void (\*car\_stop)(uint8\_t workMode);**

**void (\*car\_init)();**

**void (\*car\_speed\_updata)();**

**}ST\_car\_function\_interface\_handle,\*ST\_car\_function\_interface\_handle\_p;**

**extern ST\_car\_function\_interface\_handle car\_function\_interface\_handle;**

**extern ST\_car\_function\_interface\_handle\_p car\_function\_interface\_handle\_p;**

### 函数说明

**void (\*car\_straight\_line)(uint8\_t fwd\_rev,uint16\_t speed);**

**void (\*car\_turn)(uint8\_t fwd\_rev,uint8\_t turn\_left\_right,int32\_t car\_r,uint16\_t speed);**

**void (\*car\_stop)(uint8\_t workMode);**

**void (\*car\_init)();**

**void (\*car\_speed\_updata)();**