\*\*\* Motor driver MC & MA are swapped on backward left motor!!

电机驱动淘宝链接&资料

<https://item.taobao.com/item.htm?spm=a1z09.2.0.0.2e852e8d61dPX7&id=534387708096&_u=42505sik8762>

电机驱动IC淘宝链接&资料

<https://item.taobao.com/item.htm?spm=a1z09.2.0.0.1d8e2e8dnXELkK&id=570026841533&_u=42505sikb8f6>

轮毂电机淘宝链接&资料

<https://item.taobao.com/item.htm?spm=a1z09.2.0.0.1d8e2e8dnXELkK&id=549972092980&_u=42505sikdd96>

**UART**

Data: 02 00 0X 00 0Y

X-> A or C; A for right wheel, C for left wheel

Y-> 0 or 1; 0 for forward, 1 for backward

Data: 02 00 0X 00 YY

X -> B or D; B for right wheel, D for left wheel

YY-> 00 to FF; (0~255) speed value

Data read from STM32: 0X 00 XX 0Y 00 YY

X -> 0 or 1; for right wheel direction XX->00 to FF (0~255) speed value for right wheel speed

Y -> 0 or 1; for left wheel direction YY->00 to FF (0~255) speed value for left wheel speed

04 F0 -> Enable odometry update

04 0F -> Disable odometry update

**接线方式**

**1.编码器**

Pull-up resistors: 4k ohm

pin-strap: green -> PA7 Right back wheel

white -> PA6

green -> PB6 Left back wheel

white -> PB7

**2.motor div**

pin-strap:

|  |  |  |  |
| --- | --- | --- | --- |
| **VR** | **ZF** | **EL** | **WHEEL** |
| PA0 | PA8 | PB14 | Forward right wheel |
| PA1 | PA9 | PB15 | Backward right wheel |
| PA2 | PA10 | PB12 | Forward left wh-eel |
| PA3 | PA11 | PB13 | Backward left wheel |

**3.遥控器**

CH1 -> PB8

CH2 -> PB9

USART3 PB10 – WHITE, PB11 - GREEN

需要遥控器控制时候，Remote\_flag置1；不需要时，Remote\_flag置0 ；

程序烧录：1.已经配置好了ST\_link，可用ST\_link直接接线烧录程序2.采用窗口烧录程序

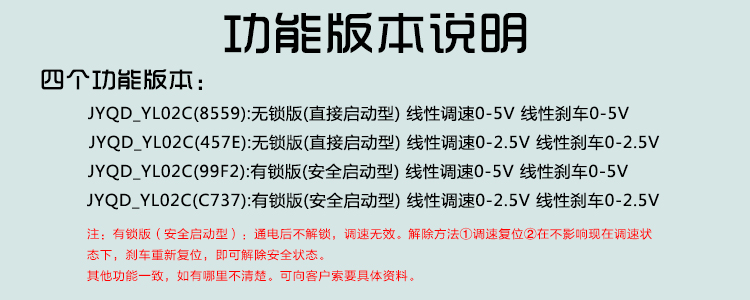
1、驱动芯片性能问题 。

——描述：电机无法快速启动，快速刹车以及快速反转。大大降低了底盘的转向性能以及灵活性。

——改善方案：更换IC（8559）

——期望效果：电机快速启动，快速制动，快速反转，底盘控制灵敏

——实际效果：实现地盘控制，pid算法可行，控制适应性待测试



2、STM32引脚输出电压问题

——描述：(negative)STM32IO口输出电平最大为3.3V，而电机驱动需要0-5V的控制，导致控制性能不高。(positive)具有芯片保护作用，控制电压强行被限幅为0%~66%。

——改善方案：增加变压模块

——期望效果：实现多路0~5V的控制信号，提高控制性能

——实际效果：待实施

3、底盘机械连接问题

——描述：测试时容易损坏

——改善方案：改用金属打印件

——期望效果：稳固

——实际效果：待实施

4、编码器问题

——描述：四轮驱动，只用两个编码器采集两个后轮的速度。每一组轮子的速度默认相同，即前轮速度默认等于后轮速度。因此在前后轮存在机械差异或是环境差异的时候，同组轮子的前后轮速度将不同，因此在控制上面存在误差。

——改善方案：

（1）采用四个编码器，分别采集每一个轮子的速度脉冲值，每一个轮子单独进行闭环控制。 优点：通过程序给定的数据，实现同组轮速度一只，误差相对较小。缺点：占用资源（硬件是需多加两个编码器，多加一个MCU，或是更换资源更丰富的STM32;占用芯片；占用芯片资源多）多，计算算法复杂。

（2）前轮改为从动轮，通过机械连接方式，保证前后论转速一致。优点：控制算法简单，保证了前后论速度一致，误差相对较小。缺点：机械结构搭建问题。

——期望效果：精准控制，稳定；

——实际效果：待实施

Issues:

**1)Printf 无法使用**

-通过上面的了解，我们知道像printf()这类函数是使用了半主机模式，我们现在只需不在半主机模式下使用printf就可以解决问题了。

\_\_use\_no\_semihosting\_swi，即不使用半主机模式，   
因程序中并没有对这些函数的底层实现，使得设备运行时会进入软件中断BAEB处，这时就需要\_\_use\_no\_semihosting\_swi这 个声明，使程序遇到这些文件操作函数时不停在此中断处.

**2) stm32 不断的重新启动的原因**

由于 stm32 系统不断的重新启动，最后的原因就是 stm32 开发板子上面的 reset 引脚是悬空的状态，受到其他信号的干扰的，不接的情况下 ，有可能当做是天线的作用 导致mcu 不断的重新启动的原因。

**3)电机驱动启动堵转保护，会导致大约5s之后再给电机PWM信号控制电机会不转**

**(not happening in relay mode?)**

JY01(8559)在调速信号特别低的时候，芯片的阈值很低，处于“弱驱动”状态，就是给与电机加一点点电值电机不转，当驱动电压低于某个值的时候，需要帮刹车加一点，堵转保护就会解除，但每次遇到这样的状况都要处理，刹车那边要给信号或者在将VR端口电压调到0V或直接断电再通电解除保护。

控制电机驱动EL端口给予5V高电平解除堵转保护，使能电机驱动刹车功能；轮子需要转动时再DISABLE EL低电平