

顶配版四轮两驱差速机器人开发手册

1. 简单入门
2. CAN 控制
3. CAN 采集数据
4. 串口控制
5. 串口采集数据与 OLED
6. 注意事项
7. 如何给小车下载程序

1. 简单入门与 APP 调参

收到小车后，在确认外观没有受损的情况下，就可以测试一下了。可以选择拆掉或者不拆掉上层的铝合金板。首先插上电池红色接头，此时，红色的电源指示灯亮起，系统供电正常。与此同时，蓝灯也开始闪烁，代表着单片机已经正常启动了。左上方的使能开关，拨到 ON 的时候才能控制电机。资料里面有测试视频教程。**注：为保证测试效果，请测试之前务必清除小车轮胎和地面的灰尘。**接下来请把最新版的 MiniBalance APP 安装到安卓手机上，然后根据相应的视频教程操作即可遥控小车或者进行参数在线调节等。具体每个通道的名称可以根据下图设置，点击右上方的菜单可以自定义。



另外，在调节 PID 参数之前，我们需要点击【获取设备参数】(菜单按钮调出)，把小车的 PID 参数更新到 APP 上面，然后拖动滑块，当我们松手的时候，APP 即可把参数发送到小车上。

2.CAN 控制

CAN (Controller Area Network)即控制器局域网，因为具有高性能、高可靠性以及独特的设计而越来越受到关注, 现已形成国际标准，被公认为几种最有前途的现场总线之一。小车提供 CAN 接口，可以通过 CAN 总线输入控制指令，下面是通信协议说明。

2.1 CAN 使能和失能

默认小车接收蓝牙指令，如果需要对小车接收 CAN 指令，需要先向 CAN 口发送使能指令。波特率是 1M。

使能指令格式如下：

标识符 ID:0x121

帧类型：标准帧

帧格式：DATA

DLC:8 个字节

数据域	tx[0]	tx[1]	tx[2]	tx[3]	tx4]	tx[5]	tx[6]	tx[7]
内容	10	12	15	19	24	30	37	Flag

Flag=1 时，CAN 控制使能，之后主板不再接收蓝牙模块的指令。此时，我们可以发送 CAN 指令对小车进行控制。如果在使能 CAN 控制之后要失能，也可发送上面的指令，但是 Flag=0；

2.2 CAN 控制指令说明

标识符 ID:0x121

帧类型：标准帧

帧格式：DATA

DLC:8 个字节

数据域	tx[0]	tx[1]	tx[2]	tx[3]	tx[4]	tx[5]	tx[6]	tx[7]
内容	A 电机速度控制	B 电机速度控制	A 电机方向控制	B 电机方向控制	预留	预留	预留	预留

可对每个电机单独进行闭环控制，tx[0]控制 A 电机的速度大小，tx[1]控制 B 电机的速度大小，tx[2]控制 A 电机的速度方向，非零是正，零是负。tx[3]控制 B 电机的速度方向，非零是正，零是负。

3.CAN 数据采集

小车在接受 CAN 控制指令的同时，也可发送小车自身的传感器数据

数据传输如下：

标识符 ID:0x101

帧类型：标准帧

帧格式：DATA

DLC:8 个字节

数据域	rx[0]	rx[1]	rx[2]	rx[3]	rx[4]	rx[5]	rx[6]	rx[7]
内容	电机 A 速度大小	电机 A 速度方向	电机 B 速度大小	电机 B 速度方向	Z 轴角度高 8 位	Z 轴角度低 8 位	Z 轴角速度高 8 位	Z 轴角速度低 8 位

接收之后需要对数据进行解包：

Yaw = (float)((rx[4]*256+rx[5])-30000)/100;//单位是° //Z 轴角度

GZ= rx[6]*256+rx[7]-32768;//MPU6050 原始 Z 轴角速度数据

Encoder_A=(1-rx[1])*rx[0];//每 10ms 采集得到的脉冲计数

Encoder_B=(1-rx[3])*rx[2];//每 10ms 采集得到的脉冲计数

发送的数据不包括电机位置信息，如果需要使用到位置信息，可以在接收数据之后，单位时间内对速度进行积分得到位置。

4. 串口控制

串行接口（Serial Interface）是指数据一位一位地顺序传送，其特点是通信线路简单，只要一对传输线就可以实现双向通信。小车可以通过串口控制，这大大降低了大家的使用难度，下面是通信协议说明。

4.1 串口控制使能

首先需要说明的是，小车上具体了 3 个串口，串口 2 用于连接蓝牙了。串口 1 和串口 3 都可以发送里程计或者接收控制指令，这 2 个串口具备同样的功能，我们可以选择任意一个串口使用，波特率都是 115200。当然，2 个串口不能同时发送指令，要不然会有异常，但是可以同时接收里程计。其他控制器向小车发送数据控制小车的频率原则上没有限制，建议不大于 200hz。但是串口 1 和串口 3 还是有区别的，解释如下：

串口 1 通过 USB 转 TTL 芯片接到了控制器的 MicroUSB 接口，我们可以使用 MicroUSB 数据线连接到笔记本电脑上面，通过串口调试助手发送或者接收数据。

串口 3（控制器上面引出的 4pin 排针，有丝印“串口”）是 TTL 电平的，我们可以通过杜邦线连接到另外的控制器，比如 STM32，Arduino 等进行通信。

默认小车接收蓝牙遥控指令，如果要其他的设备通过串口对小车进行控制，需要先使能串口控制。为了降低使用难度，串口使能机制比较简单，小车只要超过 10 次进入串口接收中断即可使能串口接收标志位，换一种理解就是：向小车发送 10 个任意字节即可使能串口控制。此时，我们就可以使用其他的单片机，比如 Arduino，通过串口发送指令对小车进行控制了，波特率是 115200。如果判断串口已经使能了呢？使用 APP 遥控一下，如果小车不再接收 APP 的遥控指令，即可说明串口已经使能。2 个串口对任意一个使能，另外一个也会跟着使能，需要说明的是串口（包括 CAN）接收数据都是不需要使能的，上电就可以接收小车的数

4.2 串口控制指令说明

数据域	帧头	帧头	tx[0]	tx[1]	tx[2]	tx[3]	tx4]	tx[5]	tx[6]	tx[7]
内容	0xff	0xfe	A 电机速度控制	B 电机速度控制	A 电机方向控制	B 电机方向控制	预留	预留	预留	预留

可对每个电机单独进行闭环控制，tx[0]控制 A 电机的速度大小，tx[1]控制 B 电机的速度大小，tx[2]控制 A 电机的速度方向，非零是正，零是负。tx[3]



控制 B 电机的速度方向，非零是正，零是负。通过电脑串口调试助手即可控制。如左图。这样相当于控制 A 电机以 18 的速度运动。下面解释一下这个 18 的单位，10ms 转 18(0x12) 个脉冲，1s 转 1800 个

脉冲，车轮转一圈，输出(编码器线数 500*减速比 28*4 倍频)=56000 个脉冲，实际上因为光电编码器精度非常高，为了方便信号处理与传输，在代码里面已经除以了 8 的，所以相当于车轮转一圈输出 7000 个脉冲。也就是每秒转 1800/7000 圈，再结合轮胎的直径信息，就可以得到小车的运行速度。

另外也可通过 TTL 电平的串口 3 控制小车。比如我们使用 STM32 控制器，发送 10 个字节就可以构成一个完整的控制指令：

```

Usart_Send(0xff);
Usart_Send(0xfe);
Usart_Send(0x12);
Usart_Send(0x14);
Usart_Send(0x01);
Usart_Send(0x00);
Usart_Send(0x00);
Usart_Send(0x00);
Usart_Send(0x00);
Usart_Send(0x00);

```

```
Usart_Send(0x00);
```

这个指令实现的控制效果和刚才使用电脑发送的指令一致。需要注意的是，发送的数据结构是 10 个字节，不是字符串。

5. 串口采集数据与 OLED

小车默认是通过 Android 手机的 APP 查看数据和进行遥控的，使用我们的最新的 MiniBalance APP 即可，OLED 显示屏也可以显示小车的信息，如下图所示：



第一行：X Y Z轴角度

第二行：Z轴陀螺仪数据和速度设定值

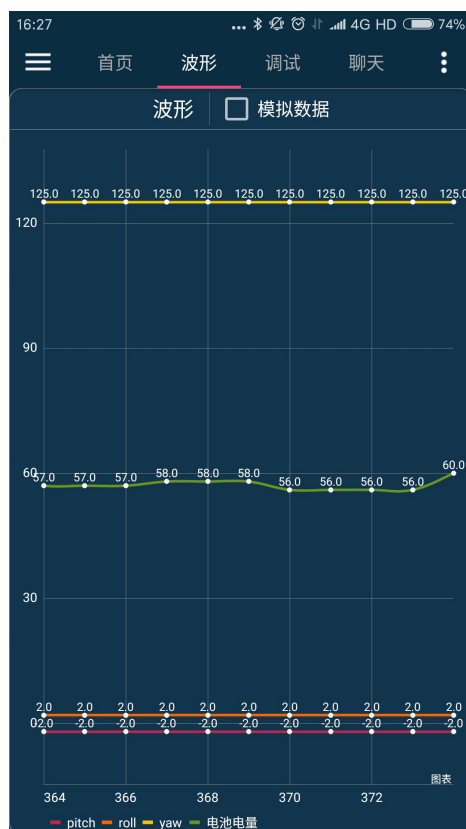
第三行：A电机目标速度（位置）和A电机实际速度（位置）

第四行：B电机目标速度（位置）和B电机实际速度（位置）

第五行：C电机目标速度（位置）和C电机实际速度（位置）

第六行：速度（位置）模式指示和电池电压

使用 APP 也可监控小车的参数，在波形界面，会通过波形显示小车的 3 轴姿态和电池电量。



我们根据上图可以对 APP 每个通道的名称重命名，以便更好的区分每个通道的数据。点击右上方的菜单按钮重命名。

另外，通过小车上面的串口 1 和串口 3 都可以采集 IMU 和里程计。可以两个串口同时接收，或者选择任意的串口接收。一帧数据有 8 个字节，其中 0xff、0xfe 是帧头，详细说明如下：

数据域	帧头	帧头	tx[0]	tx[1]	tx[2]	tx[3]	tx[4]	tx[5]
内容	0xff	0xfe	A 电机速度大小	A 电机速度方向	B 电机速度大小	B 电机速度方向	Z 轴陀螺仪高 8 位	Z 轴陀螺仪低 8 位

速度信息很容易理解，单位的解释参考控制部分。方向信息的解释：0 是正，2 是负，1 就代表数据为 0。其中 Z 轴陀螺仪的数据是+32768 再发过来的，所以，处理之后的真实 Z 轴陀螺仪数据应该是：

$$\text{GyroZ} = \text{tx}[4] * 256 + \text{tx}[5] - 32768$$

6. 注意事项

1. 轮胎或者地面上面的灰尘都会减小摩擦力，测试或者演示之前，建议清除轮胎和地面的灰尘，轮胎上面使用半湿的毛巾擦拭即可

2. 务必时刻注意电池电压，建议低于 22V 就开始充电。显示屏有显示电池电压。

3. 通过 CAN 或者串口控制的时候，在加减速的时候建议使用平滑变化的曲线，以减小小车急停或者高速启动带来的滑动摩擦，需要知道的是，编码器对滑动摩擦是无法感知的。

4. 主板上面的 5V 和 3.3V 引脚可以对外供电，但是不可带太大电流的负载，其中 5V 输出不建议带超过 1.5A 的负载，3.3V 输出不建议带超过 200mA 的负载。

5. 使用的过程中要避免电机堵转，否则很容易烧毁主板。

7. 如何给小车下载程序

小车可以通过串口或者 SWD 接口下载程序。

7.1 串口下载

主板采用了一键下载电路，下载程序非常方便。只需一根 MicroUSB 手机数据线就行了。

① 硬件准备

硬件：

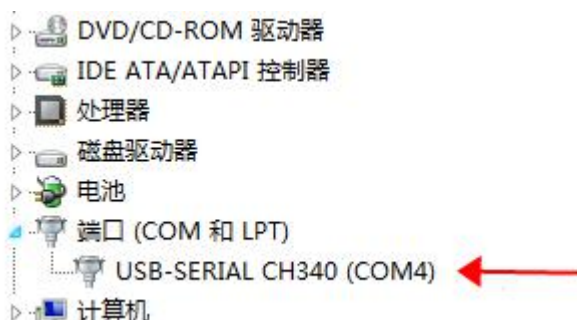
1. STM32 最小系统板

2. MicroUSB 手机数据线（尽量选择原装手机数据线）

② 软件准备

软件：FLYMCU 烧录软件（附送的资料有哈），相应的 USB 转 TTL 模块 CH340G 的驱动。附送的资料里面也有驱动哈，如果驱动安装实在困难，就下载个驱动精灵吧~

安装成功后可以打开设备管理器看看

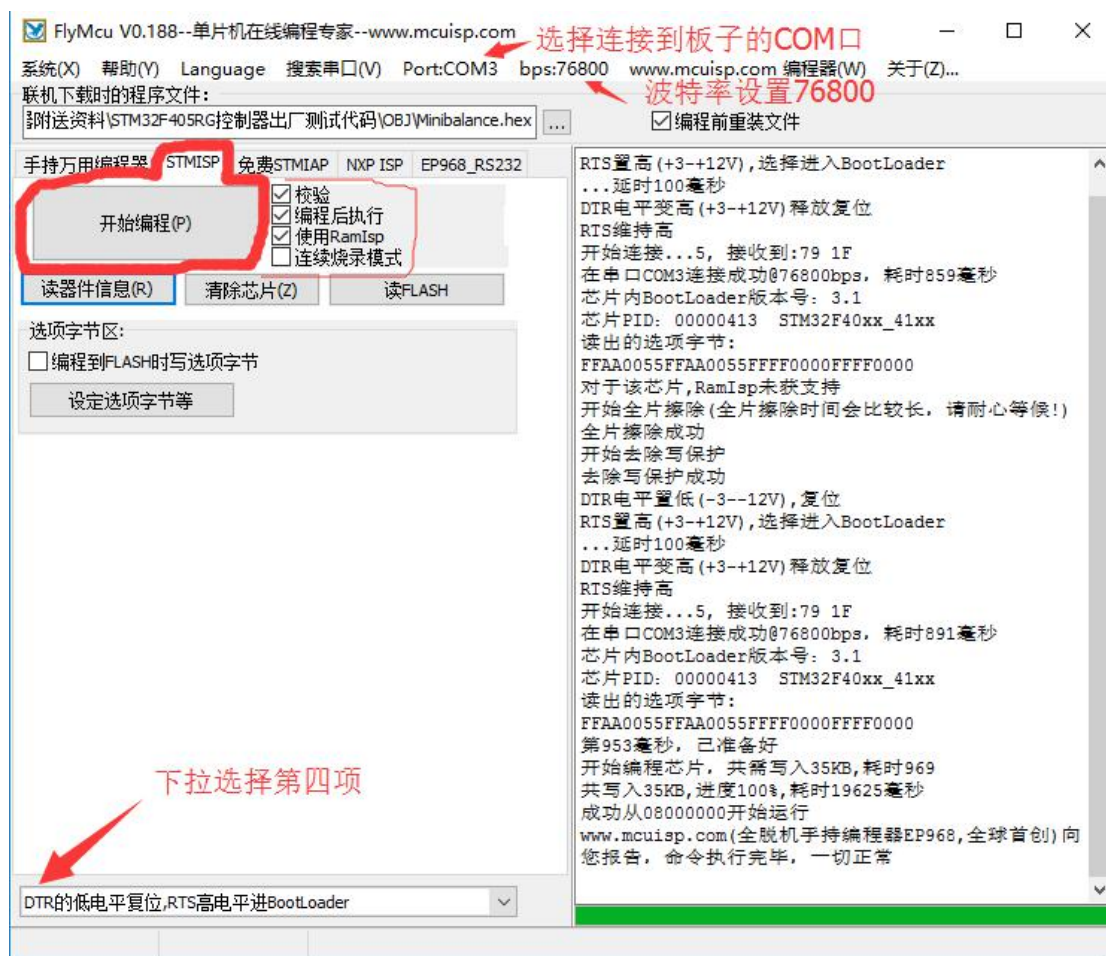


可以看到驱动已经安装成功，否则会有红色的感叹号哦！！

③ 接线

非常简单，数据线连接电脑和板子即可。

④ ISP 软件设置，打开附送资料里面的 MCUISP 软件，并做如下设置：



OK, 波特率如果是 STM32F1 下载程序, 可以设置 468000, 如果是 F4, 需要设置成 76800, 一切准备就绪, 然后点击开始编程, 程序就可以下载了! 因为勾选了编程后执行, 所以程序下载完后, 会自动运行。STM32F4 芯片擦除时间比较长, 建议使用 STlink 下载, 速度更快。

7.2 SWD 下载

小车可以通过 SWD 接口下载程序, 在主板上面有标识, 分别是 PA13 和 PA14。

1. 硬件准备

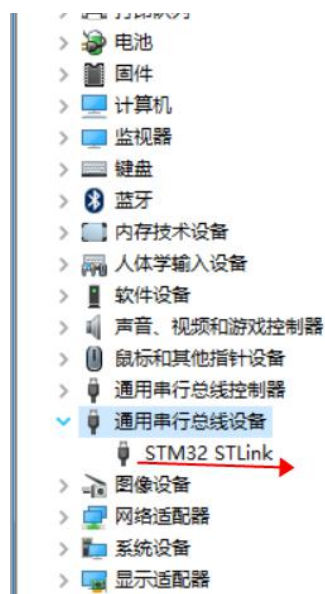
1. 小车

2. STlink

2. 软件准备

对应的 STlink 或者 Jlink 驱动的安装。

安装成功后可以打开设备管理器看看



可以看到驱动已经安装成功!

3. 接线

STlink ----- 小车

SWDIO-----PA13

SWCLK-----PA14

GND-----GND

4. OK, 一切准备就绪，确保小车已经上电了。

点击下图箭头所指的按钮，程序就可以下载了！因为勾选了编程后执行，所以程序下载完后，会自动运行。默认程序的配置是针对 STlink 的，如需配置 Jlink 下载，需要修改 MDK 设置。

