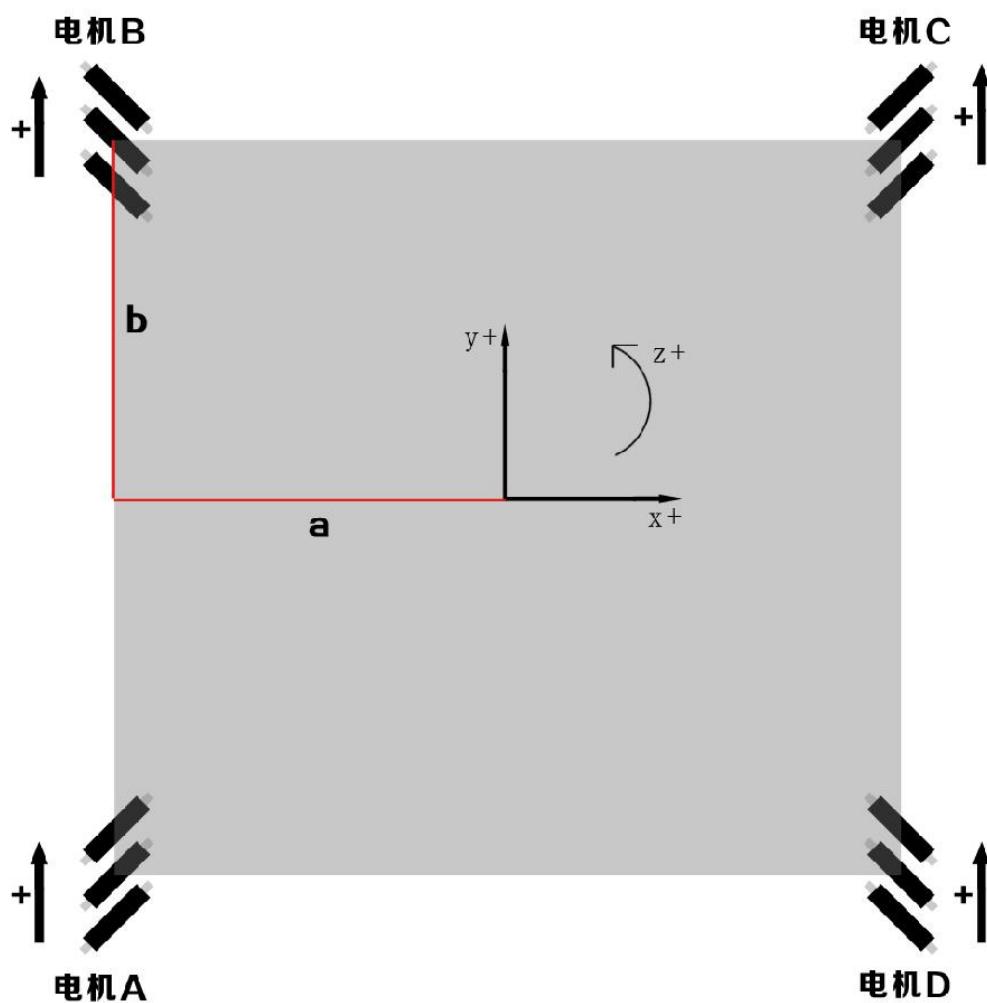


步进电机麦轮小车套件开发手册

1. 简单入门与 APP 在线调参
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. 运动学分析
7. 注意事项
8. 如何给小车下载程序



以上是电机和小车的正方向标定，下面的所有方向说明，以上图为准。

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。因电流较大，小车不设电源开关，但是有一个电机的使能开关在控制器的左上方。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。小车上上面有模式选择开关，启动的过程中，小车检查开关的状态，判断进入速度模式还是位置模式。小车启动之后，修改模式选择开关的状态，需要重新启动才能生效，资料里面有测试视频教程。左上方的使能开关，拨到 ON 的时候才能控制电机。

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。点击【掉电保存参数】可以把参数保存到 STM32 的 Flash，下次启动小车的时候，可以选择读取 Flash 的参数，具体的方法是：

如果小车开机过程的前 2s，用户按键都处于按下状态，那么小车就会从 Flash 里面读取参数，也就是我们上次掉电保存的参数。否则就使用默认的参数。

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] | tx4] | tx[5] | tx[6] | tx[7] |
|-----|-------|-------|-------|-------|------|-------|-------|-------|
| 内容 | Mode | 控制量 | 控制量 | 控制量 | 控制量 | 控制量 | 控制量 | 方向控制位 |

2.3 速度控制模式

Mode=1 时，可对小车进行坐标控制，我们只需输入每个坐标轴的速度量，小车自行进行运动学分析。tx[1]和 tx[2]合成的 16 位无符号数控制 X 轴速度的大小;tx[3]和 tx[4]合成的 16 位无符号数控制 Y 轴速度的大小;tx[5]和 tx[6]合成的 16 位无符号数控制 Z 轴速度的大小。

| 数据域 | tx[0] | tx[1] | tx[2] | tx[3] | tx4] | tx[5] | tx[6] | tx[7] |
|-----|-------|---------------|---------------|---------------|---------------|---------------|---------------|-------|
| 内容 | Mode | X 轴速度控制量高 8 位 | X 轴速度控制量低 8 位 | Y 轴速度控制量高 8 位 | Y 轴速度控制量低 8 位 | Z 轴速度控制量高 8 位 | Z 轴速度控制量低 8 位 | 方向控制位 |

tx[7]是方向控制位，由低 3 位分别控制 3 个轴的方向，说明如下：

| tx[7] | 0 | 0 | 0 | 0 | 0 | 0/1 | 0/1 | 0/1 |
|-------|----|----|----|----|----|------------------------|------------------------|------------------------|
| 内容 | 预留 | 预留 | 预留 | 预留 | 预留 | 0：X 轴正向移动 1：X 轴负向移动 | 0：Y 轴正向移动 1：Y 轴负向移动 | 0：Z 轴正向转动 1：Z 轴负向转动 |

2.4 位置控制模式

Mode=1 时，可对小车进行坐标控制，我们只需输入位移量，小车自行进行运动学分析。tx[1]和 tx[2]合成的 16 位无符号数控制 X 轴位移量;tx[3]和 tx[4]合成的 16 位无符号数控制 Y 轴位移量;tx[5]和 tx[6]合成的 16 位无符号数控制 Z 轴位移量。

| 数据域 | tx[0] | tx[1] | tx[2] | tx[3] | tx4] | tx[5] | tx[6] | tx[7] |
|-----|-------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|-----------|
| 内容 | Mode | X 轴 位 移量高 8 位 | X 轴 位 移量低 8 位 | Y 轴 位 移量高 8 位 | Y 轴 位 移量低 8 位 | Z 轴 位 移量高 8 位 | Z 轴 位 移量低 8 位 | 方向控 制位 |

tx[7]是方向控制位，由低 3 位分别控制 3 个轴的方向。

| tx[7] | 0 | 0 | 0 | 0 | 0 | 0/1 | 0/1 | 0/1 |
|-------|----|----|----|----|----|------------------------------------|------------------------------------|------------------------------------|
| 内容 | 预留 | 预留 | 预留 | 预留 | 预留 | 0：X 轴正 向移动 1：X 轴负 向移动 | 0：Y 轴正 向移动 1：Y 轴负 向移动 | 0：Z 轴正 向转动 1：Z 轴负 向转动 |

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:0x100

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 | X 轴 角 度 高 8 位 | X 轴 角 度 低 8 位 | Y 轴 角 度 高 8 位 | Y 轴 角 度 低 8 位 | 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：

Pitch= (float)((rx[0]*256+rx[1])-30000)/100;//单位是°

```
Roll = (float)((rx[2]*256+rx[3])-30000)/100;//单位是°
```

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

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

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

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。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。

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] | tx[4] | tx[5] | tx[6] | tx[7] |
|-----|------|------|-------|-------|-------|-------|-------|-------|-------|-------|
| 内容 | 0xff | 0xfe | Mode | 控制量 | 控制量 | 控制量 | 控制量 | 控制量 | 控制量 | 方向控制位 |

4.3 速度控制模式

速度控制模式的指令和 CAN 控制完全一致，具体请参考 2.3 节。

4.4 位置控制模式

位置控制模式的指令和 CAN 控制完全一致，具体请参考 2.4 节。

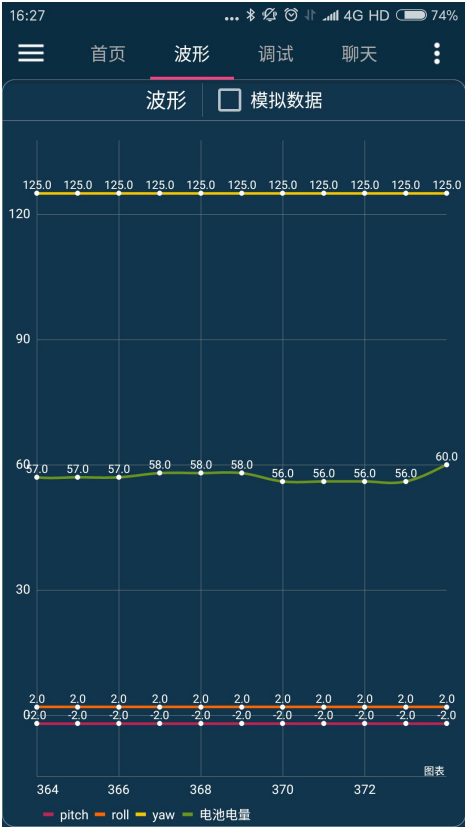
5.串口采集数据与 OLED

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：

| OLED | | | | |
|----------|-------|-------|---------------|----------------|
| X:1 | Y:359 | Z:123 | 第一行：X Y Z三轴角度 | |
| + | 0 | + | 0 | 第二行：A电机目标值和测量值 |
| + | 0 | + | 0 | 第三行：B电机目标值和测量值 |
| + | 0 | + | 0 | 第四行：C电机目标值和测量值 |
| + | 0 | + | 0 | 第五行：D电机目标值和测量值 |
| VELOCITY | | 0.00V | 第六行：模式和电池电压 | |

注：步进电机的速度和位置是通过频率间接计算得到的

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。

如果需要使用 PC 上位机，那么双击一下小车上面的用户按键即可，此时，小车开始以波特率 128000 向 PC 机发送数据包，因为上位机需要严格的时序，使用上位机的时候，小车会停止刷新 OLED 显示屏并不再向 MiniBalance APP 发送数据。

然后我们使用赠送的 USB 数据线连接到电脑。如果没安装 CH340G 驱动，还需要安装一下资料包里面的 CH340G 驱动，否则电脑无法识别。

接下来我们打开 MiniBalance 上位机，注意，不能直接双击打开，必须右键以管理员身份运行，开启之后选择合适的端口并设置好波特率即可通过上位机查看数据，非常方便。另外，计算和图形性能好的 PC 机还可以通过菜单中的扩展功能图形化显示数据！

在速度模式下，上位机的曲线对应的数据做如下说明。

第一行：A 电机实际速度

第二行：B 电机实际速度

第三行：C 电机实际速度

第四行：D 电机实际速度

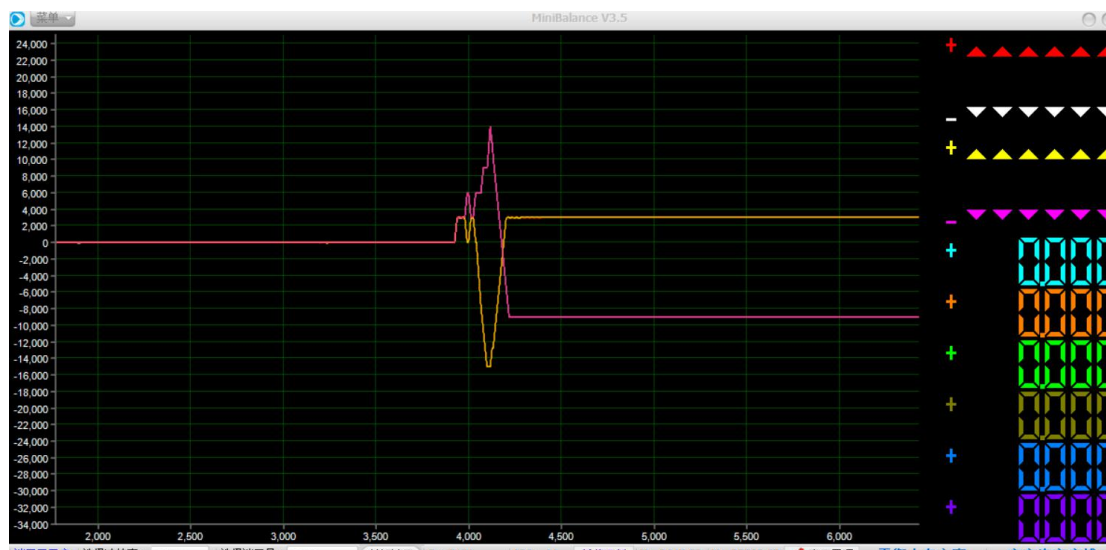
在位置模式下，上位机的曲线对应的数据做如下说明。

第一行：A 电机实际位置

第二行：B 电机实际位置

第三行：C 电机实际位置

第四行：D 电机实际位置



6.运动学分析

全向移动机器人是一种富有趣味性的，可以通过自身的运动机构并借助控制系统的帮助到达指定位置或者沿指定轨迹运动的机器人。因其全方位移动的特性，在工业、医药等领域有普遍的应用，在一些机器人竞赛如 robocon 中，也有一些高校应用了。

全向移动机器人一般使用 2 种轮子，全向轮（Omni Wheel）和麦克纳姆轮（Mecanum Wheel），这里我们研究的是麦克纳姆轮，也简称麦轮。

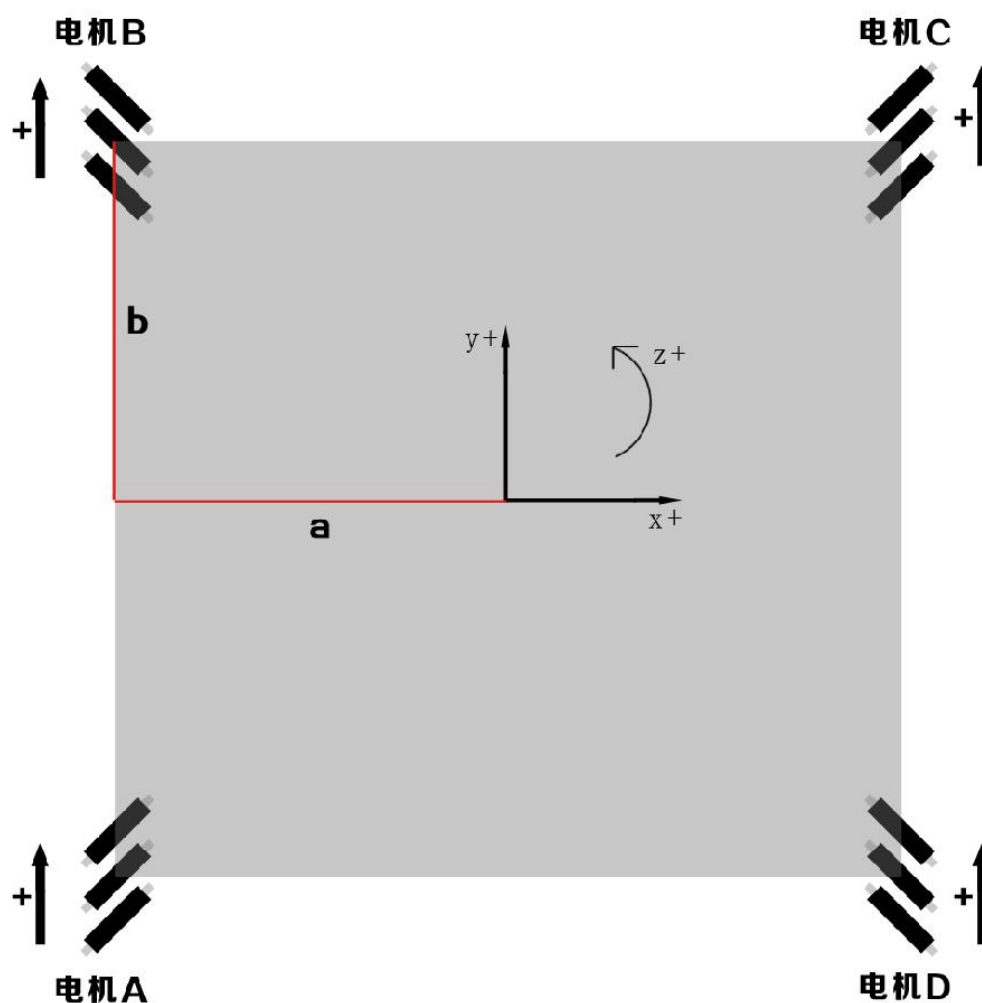
Mecanum 轮是瑞典 Mecanum AB 公司的工程师 Bengt Ilon 于 1973 年提出，随后，世界众多科研机构进行了深入的研究，主要集中在有何轮体结构、改进建模方法及其精确运动控制上。Mecanum 轮由轮毂和一组均匀分布在轮毂周围的鼓状辊子组成，轮毂轴线与辊子轴线成一定角度（通常为 45° ）。

6.1 运动建模

机器人控制的基础是运动学分析，利用运动学分析可以得出机器人运动过程中各类参数的变化规律和相互之间的关系，然后通过 C 语言在控制器里面进行验证。在运动建模之前，为简化运动学数学模型，做下列几种理想化假设：

- （1）全向轮不与地面打滑，同时地面有足够摩擦力；
- （2）4 个轮子分布在长方形或者正方形的 4 个角上，轮子之间相互平行。

假定机体坐标系和地理坐标系重合，机器人运动方向做如下图所述的规定。



刚体运动可以线性分解为三个分量。那么只需计算输出麦轮底盘在沿 $X+$, $Y+$ 方向平移， $Z+$ 方向转动时四个轮子的速度，就可以通过公式的合并，计算出这三种简单运动所合成的“平动+旋转”运动时所需要的四个轮子的转速。

其中 V_A 、 V_B 、 V_C 、 V_D 分别为 A、B、C、D 四个轮子的转速，也就是电机的转速。 V_x 为小车沿着 X 轴平移速度， V_y 为小车沿着 Y 轴平移速度， ω 为小车沿 Z 轴的旋转速度。

当小车沿着 X 轴平移时：

$$V_A = -V_x$$

$$V_B = +V_x$$

$$V_C = -V_x$$

$$V_D = +V_x$$

当小车沿着 Y 轴平移时：

$$V_A = +V_y$$

$$V_B = +V_y$$

$$V_C = +V_y$$

$$V_D = +V_y$$

当小车绕几何中心自转时：

$$V_A = -\omega (a+b)$$

$$V_B = -\omega (a+b)$$

$$V_C = +\omega (a+b)$$

$$V_D = +\omega (a+b)$$

将以上三个方程组合并，即可根据小车的运动状态解算出四个轮子的转速：

$$V_A = -V_x + V_y - \omega (a+b)$$

$$V_B = +V_x + V_y - \omega (a+b)$$

$$V_C = -V_x + V_y + \omega (a+b)$$

$$V_D = +V_x + V_y + \omega (a+b)$$

6.2 C 语言实现与控制原理图

根据以上的分析，我们可以很容易使用 C 语言编程。

```
#define a_PARAMETER      (0.1025f)
#define b_PARAMETER      (0.086)

void Kinematic_Analysis(float Vx, float Vy, float Vz)
{
    Target_A    = -Vx+Vy-Vz*(a_PARAMETER+b_PARAMETER);
```

```

Target_B    = +Vx+Vy-Vz*(a_PARAMETER+b_PARAMETER);

Target_C    = -Vx+Vy+Vz*(a_PARAMETER+b_PARAMETER);

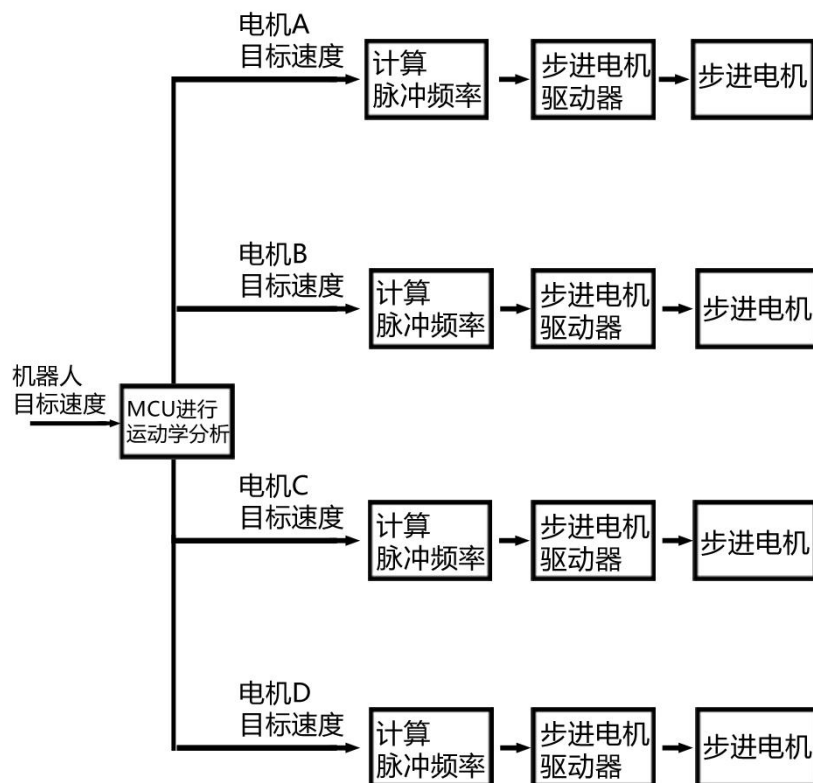
Target_D    = +Vx+Vy+Vz*(a_PARAMETER+b_PARAMETER);

}

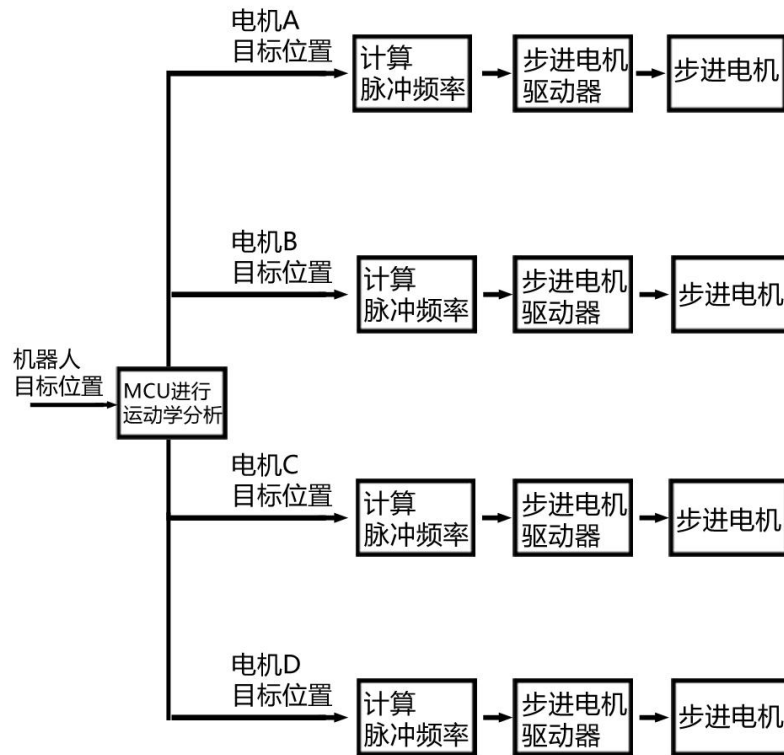
```

入口参数为 X Y Z 轴速度，解算出四个电机的目标转速，送入 PID 控制器进行控制。

以下分别是速度控制和位置控制的原理图。



全向移动机器人速度控制原理图



全向移动机器人位置控制原理图

7.注意事项

1. 麦轮相比普通的轮胎更容易打滑，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 11.5V 就开始充电，锂电池过放必然会导致电池永久损坏。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，APP 遥控的时候自带了加减速控制。

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 1A 的负载，3.3V 输出不建议带超过 200mA 的负载。

5. 电池充电的时候请不要使用电池，接线如下图所示：



8.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。

8.1 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

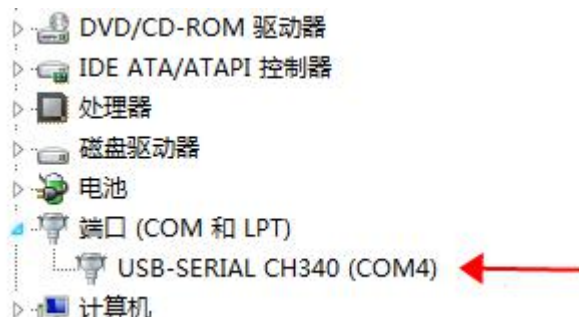
硬件：

1. 小车
2. MicroUSB 手机数据线

② 软件准备

软件：MCUIISP 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

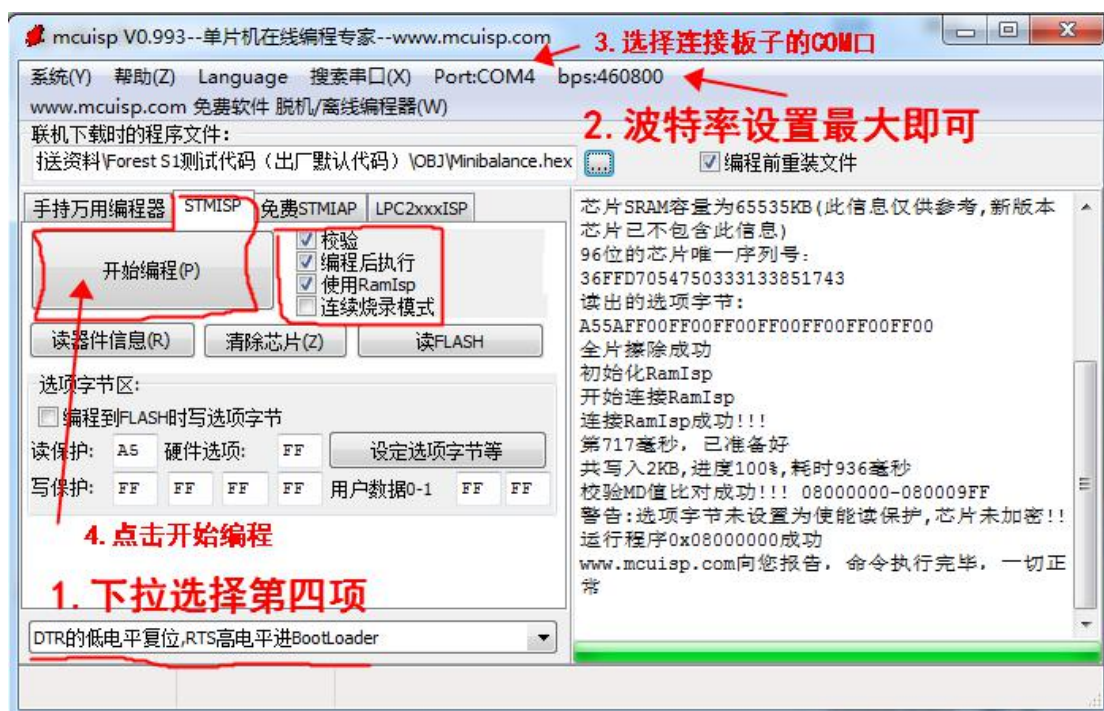


可以看到驱动已经安装成功，否则会有红色的感叹号哦！！

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，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。

（如果下载错误，请去掉上面软件中 RamIsp 前面的勾选）

8.2 SWD 下载

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

1. 硬件准备

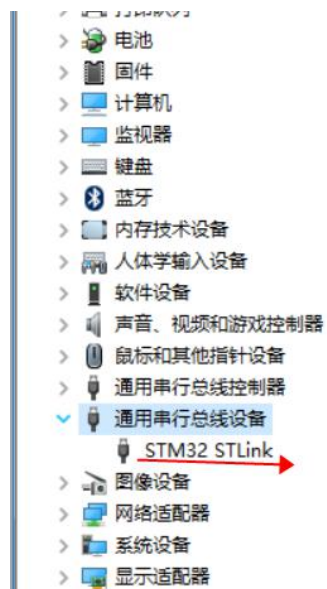
1. 小车

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。

