

# 麦克纳姆轮运动学分析及编程指南

塔克出品



## 版权申明

本手册版权归属塔克创新所有，并保留一切权力。非经(书面形式)同意，任何单位及个人不得擅自摘录或修改本手册部分或全部内容，违者我们将追究其法律责任。

感谢您购买塔克创新产品，在使用产品之前，请仔细地阅读该手册并且确保知道如何正确使用该产品，不合理的操作可能会损坏产品，使用过程中随时参考该手册以确保正确使用。此手册不断更新中，建议您登录论坛、博客、交流群或公众号下载最新版本。

## 塔克媒体

塔克淘宝店铺	<a href="https://shop246676508.taobao.com/">https://shop246676508.taobao.com/</a>
塔克官网社区	<a href="http://www.xtark.cn">www.xtark.cn</a>
塔克技术博客	<a href="http://www.cnblogs.com/xtark/">http://www.cnblogs.com/xtark/</a>
技术商务联系	18660035816 (微信 QQ 同号)
塔克服务邮箱	<a href="mailto:service@xtark.cn">service@xtark.cn</a>

塔克淘宝，微信公众号二维码：



更多精品 · 欢迎关注塔克淘宝店铺



更多资讯 · 欢迎关注塔克创新微信号

## 1. 目录

1. 产品介绍 .....	4
1.1. 产品概述 .....	4
1.2. 参数指标 .....	错误!未定义书签。
1.3. 产品清单 .....	错误!未定义书签。

## 1. 麦克纳姆轮介绍

### 1.1. 什么是麦克纳姆轮

麦克纳姆轮，是瑞典麦克纳姆公司设计。在中心轮圆周方向又布置了一圈独立的、倾斜角度（ $45^\circ$ ）的行星轮，这些成角度的行星轮把中心轮的前进速度分解成 X 和 Y 两个方向，实现前进及横行。麦克纳姆轮结构紧凑，运动灵活，是很成功的一种全方位轮。有 4 个这种新型轮子进行组合，可以更灵活方便的实现全方位移动功能。最近比较火的 DJI ROBOMASTER S1 机器人也采用了麦克纳姆轮。

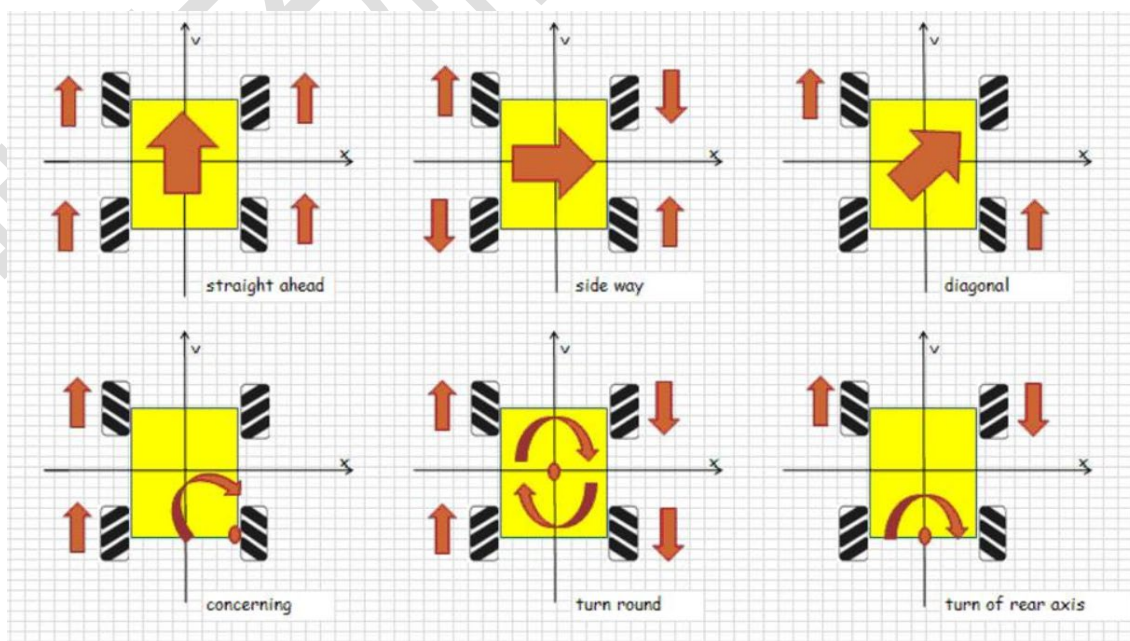


依靠各自机轮的方向和速度，这些力的最终合成在任何要求的方向上产生一个合力矢量从而保证了这个平台在最终的合力矢量的方向上能自由地移动，而不改变机轮自身的方向。在它的轮缘上斜向分布着许多小滚子，故轮子可以横向滑移。小滚子的母线很特殊，当轮子绕着固定的轮心轴转动时，各个小滚子的包络线为圆柱面，所以该轮能够连续地向前滚动。麦克纳姆轮结构紧凑，运动灵活，是很成功的一种全方位轮。有 4 个这种新型轮子进行组合，可以更灵活方便的实现全方位移动功能。



## 1.2. 麦克纳姆轮行走原理

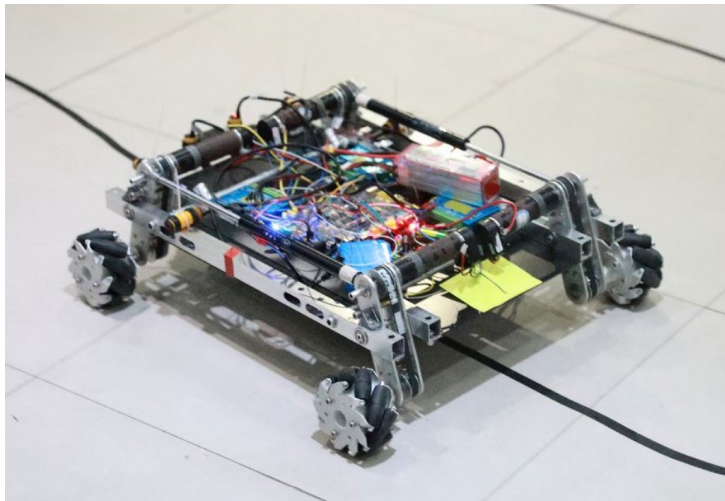
四个麦克纳姆轮分别连接到电机上进行独立控制。机器人可以像四个普通车轮一样前进，后退和旋转。滚子的配置在  $45^\circ$  还允许机器人在任何方向上横向移动并通过它们的组合（即使在旋转！）。我们将力量分成两个向量，一个向前/向后和一个右/左。当一侧的车轮以相反的方向旋转时，向前和向前旋转向后向量抵消，而两个横向向量加起来。与之相反另外两个轮子导致四个增加的侧向矢量。





### 1.3. 应用场景

目前麦轮主要有两个应用场景，竞赛机器人和特殊工种机器人。竞赛机器人，主要以 ROBOMASTER 机甲大师赛和 ROBOCON 机器人竞赛，如下图。



工业机器人如下图。



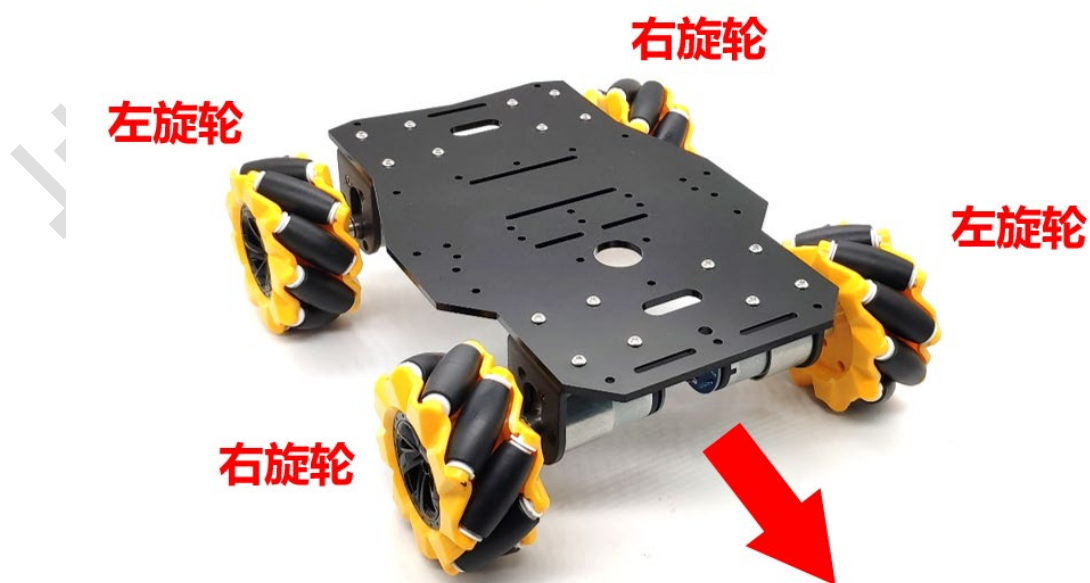
## 2. 运动建模

### 2.1. 轮子安装方式

麦轮一般是四个一组使用，两个左旋轮，两个右旋轮。左旋轮和右旋轮呈手性对称，区别如下图所示。



安装方式可参考如下图。



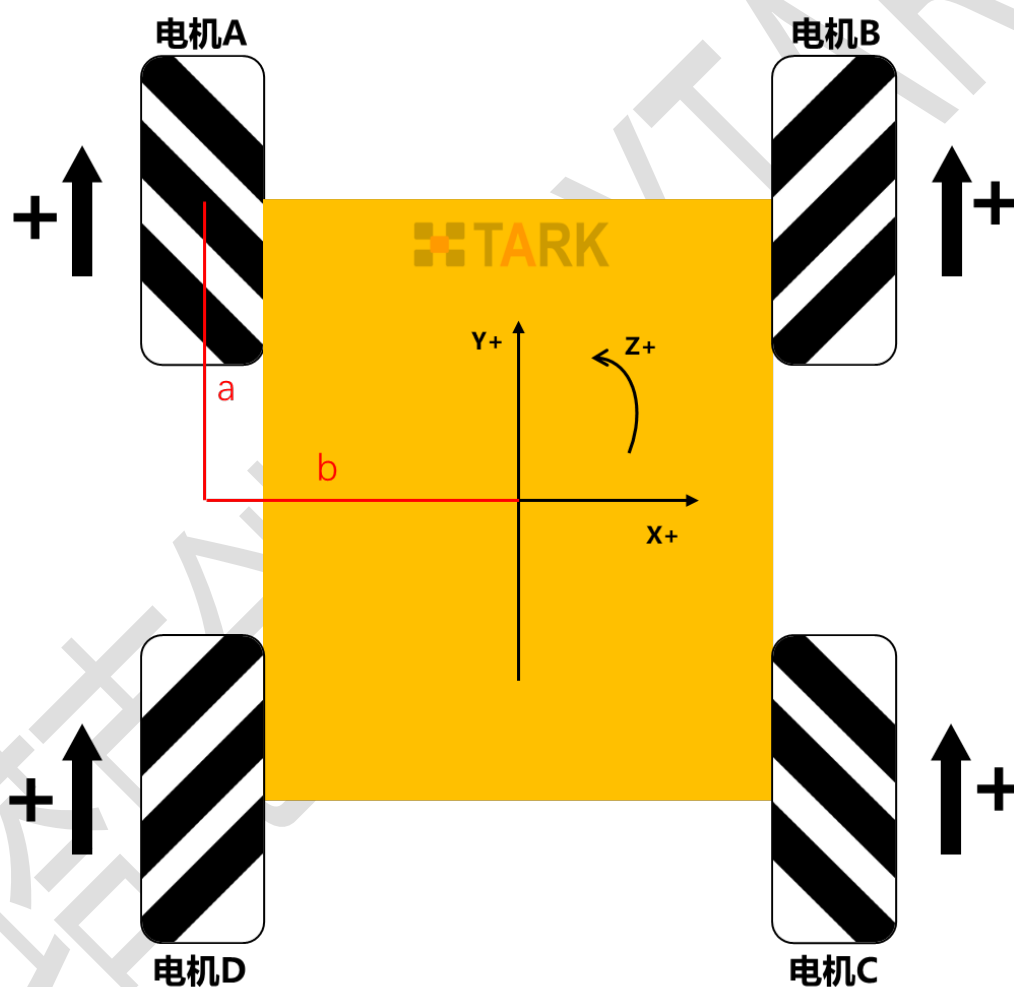
## 2.2. 逆运动模型解析

正运动学模型 (forward kinematic model) 将得到一系列公式, 让我们可以通过四个轮子的速度, 计算出底盘的运动状态; 而逆运动学模型 (inverse kinematic model) 得到的公式则是可以根据底盘的运动状态解算出四个轮子的速度。本小节进行逆运动学模型解析。

### 运动模型假设

1. 全向轮与地面摩擦力足够, 不打滑;
2. 四个轮子分布在长方形四个角上, 轮子之间平行。

机器人坐标系与地理坐标系重合, 电机及机器人运动方向如下图所示。



### 变量定义

#### 底盘三维运动分量

$v_x$  表示  $X$  轴运动的速度, 即左右方向, 定义向右为正;

$v_y$  表示  $Y$  轴运动的速度, 即前后方向, 定义向前为正;



$v_z$  表示 Z 轴运动的速度，即自转角速度，定义逆时针为正；

$V_A, V_B, V_C, V_D$  分别表示四个轮子转速。

$a, b$  如上图所示，表示轮子与机器人中心的垂直和水平距离。

### 模型解析

机器人沿 X 轴运动时，即横向运动。

$$V_A = v_x$$

$$V_B = -v_x$$

$$V_C = v_x$$

$$V_D = -v_x$$

机器人沿 Y 轴运动时，即前后运动。

$$V_A = v_y$$

$$V_B = v_y$$

$$V_C = v_y$$

$$V_D = v_y$$

机器人沿 Z 轴运动时，即绕几何中心旋转。

$$V_A = -v_z (a+b)$$

$$V_B = v_z (a+b)$$

$$V_C = v_z (a+b)$$

$$V_D = -v_z (a+b)$$

综合以上三个解析，可计算获得机器人四个轮子的转速。

$$V_A = v_x + v_y - v_z (a+b)$$

$$V_B = -v_x + v_y + v_z (a+b)$$

$$V_C = v_x + v_y + v_z (a+b)$$

$$V_D = -v_x + v_y - v_z (a+b)$$

## 3. 编程指南

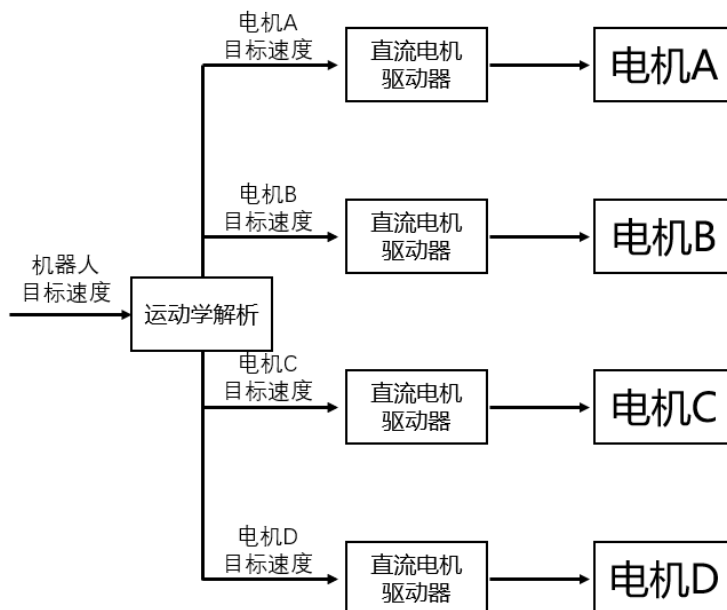
提供的示例代码，符合塔克 X-SOFT 编程规范，易读易用。

### 3.1. 电机开环控制指南

电机开环控制相对简单，可直接将解算的速度赋值为电机的 PWM 变量，实现电机转速控制。注意比例系数和 a,b 变量可根据自己实际情况修改。

```
/**
 * @简述 由坐标XYZ速度解析为电机转速
 * @参数 vx, vy, vz 三轴坐标速度
 * @返回值 全局变量，每个轮子PWM速度
 */
void AX_MOVE_Kinematics(int16_t vx, int16_t vy, int16_t vz)
{
    //运动解析
    ax_motor_pwm[0] = 10*(vx + vy - ROBOT_AB*vz);
    ax_motor_pwm[1] = 10*(-vx + vy + ROBOT_AB*vz);
    ax_motor_pwm[2] = 10*(vx + vy + ROBOT_AB*vz);
    ax_motor_pwm[3] = 10*(-vx + vy - ROBOT_AB*vz);
}
```

控制框图如下。



### 3.2. PID 闭环控制指南

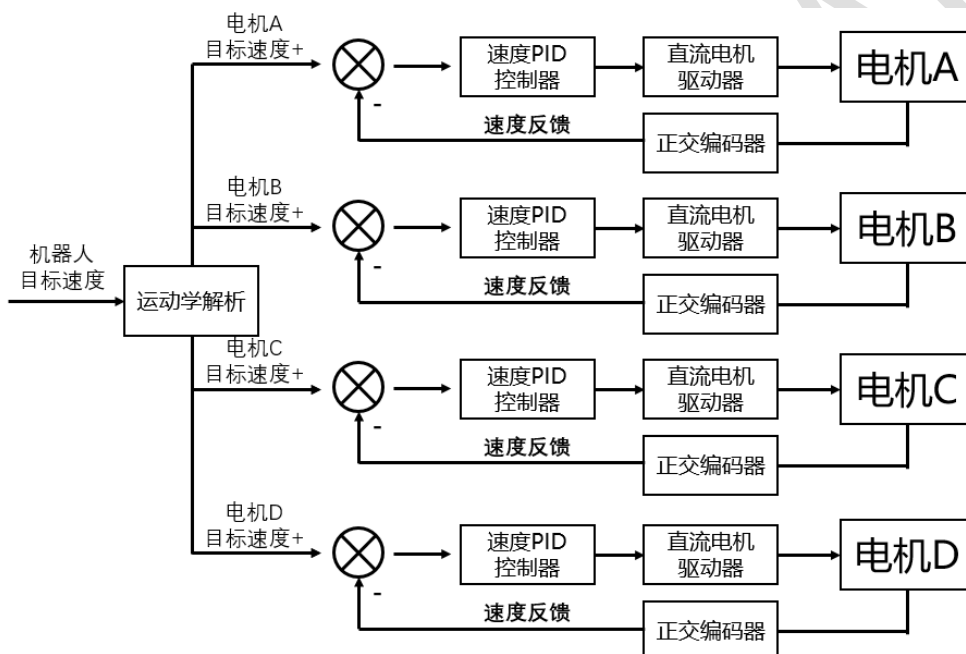
开环控制由于电机差异及环境影响，智能实现粗略的运动模型控制。闭环控制与开环控制相似，将解算的电机驱动赋值为电机的目标速度，电机通过 PID 实现目标速度跟随控制。

```

/**
 * @简述 由坐标XYZ速度解析为电机目标转速
 * @参数 vx, vy, vz 三轴坐标速度
 * @返回值 全局变量，每个轮子目标速度
 */
void AX_MOVE_Kinematics(int16_t vx, int16_t vy, int16_t vz)
{
    //运动解析
    ax_encoder_delta_target[0] = (vx + vy - ROBOT_AB*vz);
    ax_encoder_delta_target[1] = (-vx + vy + ROBOT_AB*vz);
    ax_encoder_delta_target[2] = (vx + vy + ROBOT_AB*vz);
    ax_encoder_delta_target[3] = (-vx + vy - ROBOT_AB*vz);
}

```

关于电机 PID 控制代码，此处不详述，请参考代码示例。控制框图下图。



更多精彩原创开源硬件，请关注塔克创新淘宝店铺。

<https://shop246676508.taobao.com/>