

RoboMasters 2015

全国大学生机器人大赛

技术报告

东北林业大学

技术报告

学校名称: 东北林业大学

队伍名称: 东林—天空领域—Ares 战队

指导教师: 宋文龙 王琢 白雪冰

作者名称: 张永超

2015年7月

目录

摘要

Abstract

1 绪论	1
1.1 比赛简介	1
1.2 整车设计框架结构预览	3
1.3 本文主要研究的内容	4
2 全方位移动平台的设计分析	5
2.1 全方位轮的种类	5
2.2 基于麦克纳姆轮的四轮系全方位移动平台机械设计	7
2.3 全方位移动平台运动分析	9
2.4 小结	10
3 运动控制系统的硬件设计	11
3.1 运动控制系统简介	11
3.2 全方位移动平台控制系统结构	11
3.3 核心器材选型	12
3.3.1 处理器芯片的选择	12
3.3.2 电源选择	13
3.3.3 电机选择	13
3.4 控制系统硬件电路设计	14
3.4.1 稳压电源设计	14
3.4.2 单片机最小系统设计	14
3.4.3 全桥驱动器的设计	15
3.4.4 过流保护检测电路	16
3.5 小结	17
4 全方位移动平台软件设计	18
4.1 软件设计思想与整体架构	18
4.2 遥控器解析协议	19
4.3 CAN 通讯	20
4.4 底盘速度分解程序	21
4.5 速度闭环控制	22
4.5.1 PID 控制原理	22
4.5.2 数字 PID	23
4.5.3 PID 的参数整定	23
4.6 小结	24
5 云台部分	25
5.1 整体机械结构设计	25
5.2 云台控制	26
5.3 云台调试	29
5.3.1 通信协议	30
5.3.2 PID 调试	30
6 爬杆机器人设计	32
6.1 机械	32

6.2 硬件	33
6.3 程序	34
6.4 总结	35
结论.....	37
参考文献.....	39

1 绪论

1.1 比赛简介

RoboMasters 是一项全新的机器人超级对抗赛。参赛队伍通过自行设计制造多种机器人进行射击对抗，完成指定任务，由比赛裁判系统判定比赛胜负。

参赛机器人包括可以发射“弹丸”的手动机器人以及能够完成一定任务的自动机器人。参赛队员需要遥控手动机器人在复杂的场地中移动并发射“弹丸”，攻击对方机器人和基地以取得胜利。自动机器人将在比赛中自动完成指定任务。

比赛采用红、蓝双方对抗的形式，每场限时 5 分钟。

其中参赛机器人的种类包含：机器射手，如图 1-1 所示；机器步兵如图 1-2 所示；机器炮手如图 1-3 所示；还包括基地、哨兵、等等元素。



图 1-1 机器射手



图 1-2 机器步兵

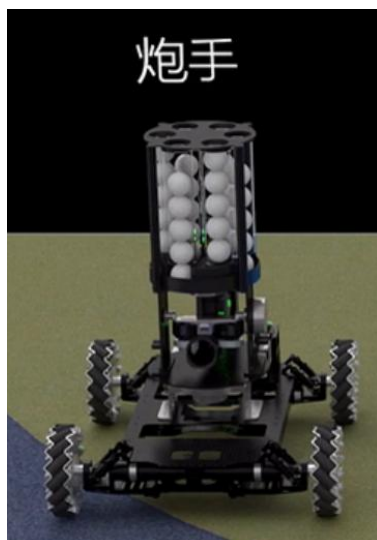


图 1-3 机器炮手

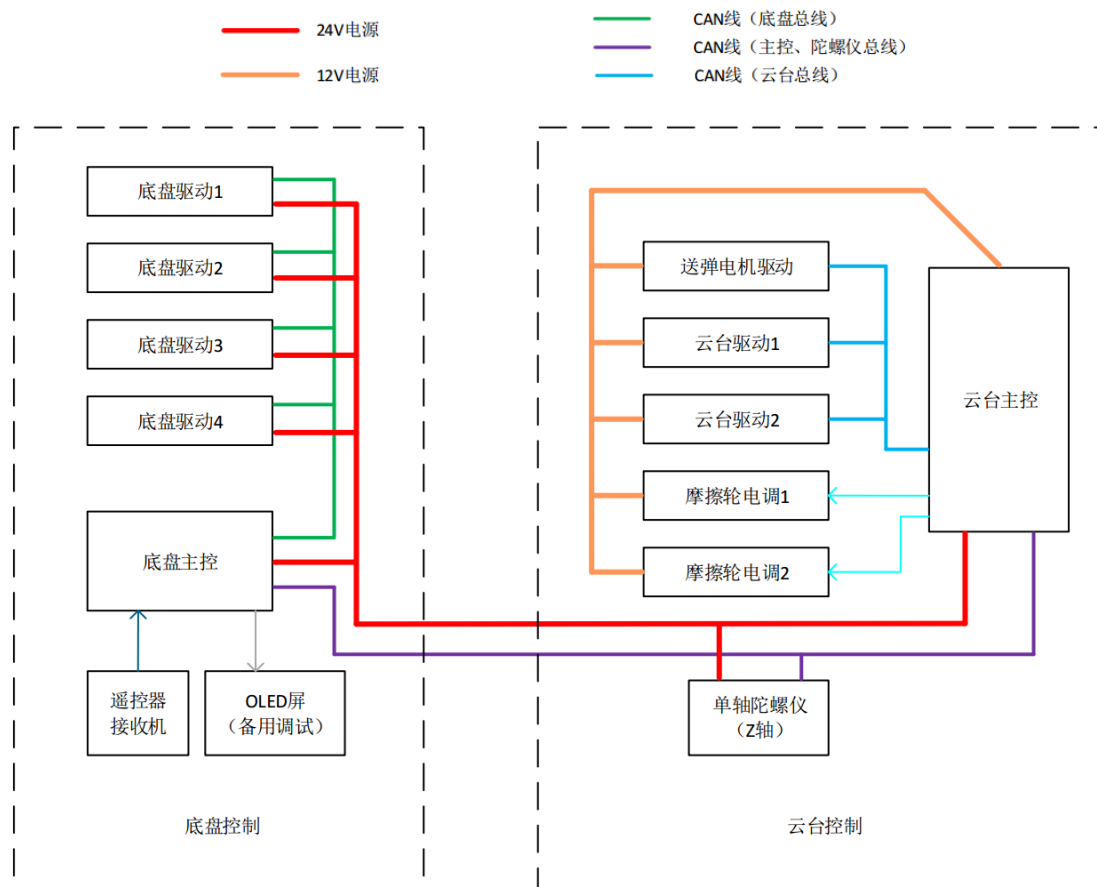


图 1-4 场地概括



图 1-5 基地

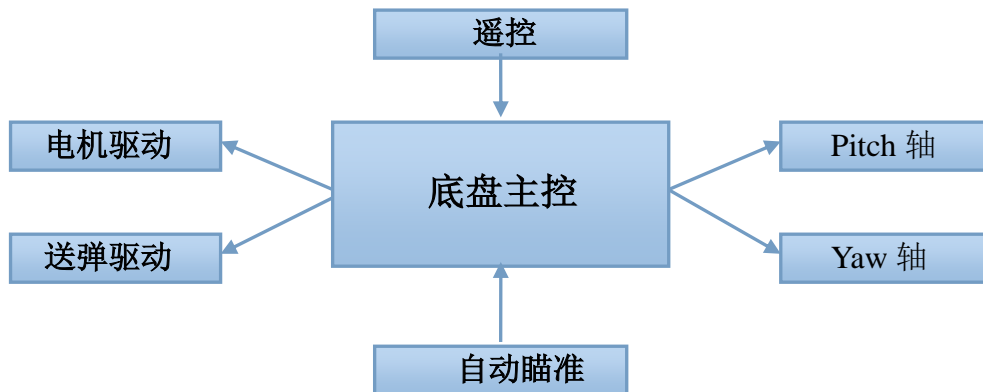
1.2 整车设计框架结构预览



具体说明如下：

- 1、供电电源 24V 由裁判系统提供，裁判系统根据战车当前的状态决定电源的通断；
- 2、底盘驱动 1~4 以及送弹电机驱动为直流有刷电机驱动器，内部实现了速度闭环和位置闭环的控制；
- 3、云台驱动 1~2 为永磁同步电机（PMSM）驱动器，内部实现了电流闭环的控制；
- 4、摩擦轮电调 1~2 为 BLDC 电子调速器（ESC），由主控发送 PWM 波实现调速；
- 5、单轴陀螺仪用来获取云台 Yaw 轴偏转角度，实现云台偏转时底盘跟随的效果；
- 6、遥控器数据包括云台数据和底盘两部分，底盘主控接收到数据后，将云台控制数据通过 CAN 透传给云台。

机器射手充当整个战队里面最有效、最灵活又最具有战斗力输出的一个角色，所以射手的性能极其关键，性能的好坏直接影响到一个团队成绩的好坏，所以我们在射手上也下了很多的功夫。系统结构总图设计如下：



1.3 本文主要研究的内容

本课题主要研究内容是针对 RoboMasters 全国大学生机器人大赛赛事做的整个项目研究，包含：全方位移动底盘、射手云台、炮手云台、发射机构、爬杆机器人等等。

前四章非常详细介绍了全方位移动底盘设计全部流程；第五章介绍云台控制；第六章介绍爬杆机器人制作过程。

主要的任务工作如下：

- (1) 设计一个微小式四轮全方位移动机器人底盘，完成机械设计，搭建机械平台，并简单分析机械结构的稳定性。
- (2) 设计一套底盘控制电路与驱动电路。底盘主控制电路基于 STM32F405RGT6 芯片设计的，下文称作底盘主控。驱动电路基于 STM32F103C8T6 芯片设计的，电机驱动部分采用 MOSFET 设计的一款全桥驱动器，导通电流能达到 130A，下文称作底盘驱动。
- (3) 整车采用 24V 动力电池，本方案设计一款具有高负载能力的 24V 稳 5V 的开关电源，为整个系统正常稳定工作带来更高的可靠性。
- (4) 研究基于麦克纳姆轮的四轮系全方位底盘的运动模型，分析底盘运动状态与各个麦克纳姆轮的运动状态的关系，同时将解析后的速度关系匹配成程序代码。
- (5) 构建 CAN 总线通信系统，搭建底盘主控与四个底盘驱动的通信关系，并预留 CAN 总线协议端口。
- (6) 控制算法采用增量式 PID 控制，完成各个驱动电机的速度闭环。
- (7) 云台控制研究。
- (8) 爬杆机器人控制研究。

2 全方位移动平台的设计分析

2.1 全方位轮的种类

全方位运动方式不同于传统底盘运动方式，是一种可以在二维平面内向任意方向、任何轨迹进行连续运动的运动方式，可以完全控制机器人在平面运动的三个自由度，包括两个水平运动分量和一个车身旋转姿态分量。

决定全方位底盘的运动性能，除了采用的电机驱动方式的不同，还取决于轮全方位轮的性能。为了方便控制和设计，大都采用相同型号的全向轮和 4 电机做为底盘运动机构的元素。然而轮子数量的不同，也会直接影响全方位移动平台的性能。

全方位移动机器人比较常用的大致可以分为以下几种：

(1) **麦克纳姆轮系** 麦克纳姆轮如图 2-1 所示：由轮辐和固定在外周的许多小滚子构成，轮子和滚子之间的夹角为 Y ，通常夹角 Y 为 45° ，每个轮子具有三个自由度，第一个是绕轮子轴心转动，第二个是绕滚子轴心转动，第三个是绕轮子和地面的接触点转动^[5]。轮子由电机驱动，其余两个自由度自由运动。由三个或以上的麦克纳姆轮子可以构成全方位移动机器人。麦克纳姆轮结构紧凑、运动灵活，是广泛应用的一种全方位轮。最初的装载轮子上的小滚子与大轮子的夹角为 45° 或 135° ，基本上为四轮的，各大轮平行。如图 2-2 所示。

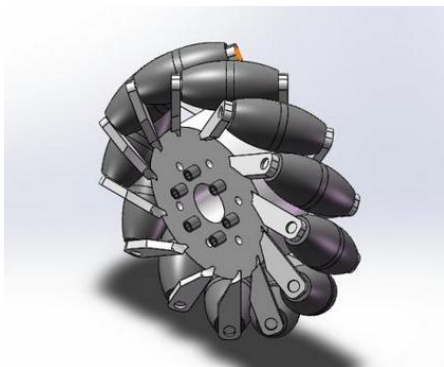


图 2-1 麦克纳姆轮



图 2-2 简单的移动底盘

(2) **万向行走轮系** 万向行走轮系是由麦克纳姆轮演化改进过来的，将麦克纳姆轮小滚子与大轮子的轴向方向的夹角由 45° 或 135° 变成了 90° ，这样就组成了万向行走轮系结构。万向行走轮系有单排的，如科内尔大学的 D. A. Raffaello 研制的由 3 个万向轮组成的足球机器人，并且在多次国际大赛上取得好成绩，如图 2-3 所示。随着时间的慢慢考证，这种单排轮的从动棍子在运动的过程中，当每个从动棍子切换的过程中会产生很大的冲击力，使用寿命低，后来的改进为双排从动棍子，这样在从动棍子来回切换的过程中会有重叠作用，使轮子使用寿命更加长久耐用，其结构如图 2-4 所示。

万向行走轮系结构，相对于麦克纳姆轮系结构来说，其运动起来稳定性以及效率都要更高一些，承载能力更强一些，所以在实际应用中比较常见。

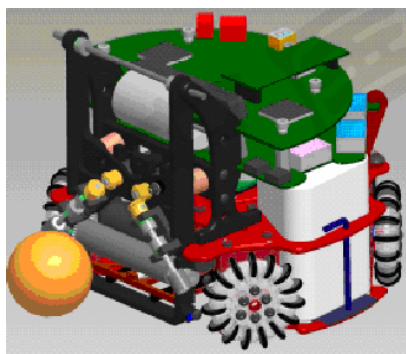


图 2-3 单排万向行走轮

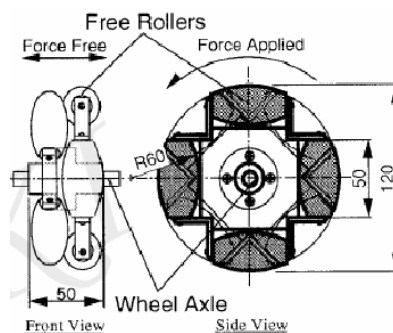


图 2-4 双排万向行走轮

(3) **正交轮系** 正交轮是由两个形状一样的球形轮子架，固定在同一个壳体上，如图 2-5 所示，每个轮架都有 2 个自由度：围绕轮子架的驱动转动，和围绕轮子轴心自由转动。两个轮子架的转动轴方向相同，由一个电机驱动，两个轮子的轴线相互垂直，因而称正交轮^[6]。

从图可以看出正交轮也是麦克纳姆轮的演化产品，它采用两个运动方向成 90° 的可自由运动的棍子代替大轮上的很多的小棍子，它们两个固定在一起，通过一个电机带动旋转，来实现万向轮的功能。2001 年的 RoboRoos 机器人比赛中中澳大利亚昆士兰大学用的就是该结构的正交轮，如图 2-6 所示。

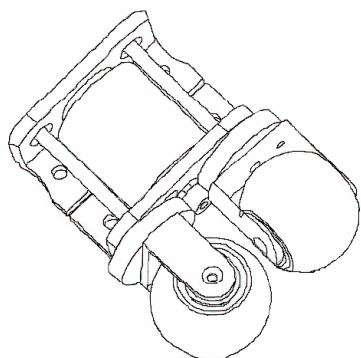


图 2-5 正交轮模型



图 2-6 正交轮

(4) **球轮** 球轮由一个滚动球体、一组支撑滚子和一组驱动滚子组成，其中支撑滚子固定在车底盘上，驱动滚子固定在一个可以绕球体中心转动的支架上，如图 2-7 所示。每个球轮上的驱动滚子由一个电机驱动，使球轮绕驱动滚子所构成平面的法线转动，同时可以绕垂直的轴线自由转动^[7]。

球轮的典型代表有麻省理工学院的 Mark West 和 Haruhiko Asada 研制了由球轮组成的全方位机器人，以及他们后来做的减少运动机器人内部冗余应力的机构，该机构能随冗余应力的方向与大小作出调整而减少内部应力，保护机构，球轮全方位移动机器人如图 2-8 所示。

球轮的总体特点是运动平稳，承载能力强，方向性好，可在狭小的空间中实现任意转向，但结构复杂，当球体上沾的杂物过多时该机构功能失效。

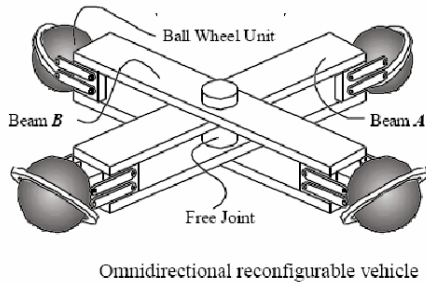


图 2-7 球轮模型

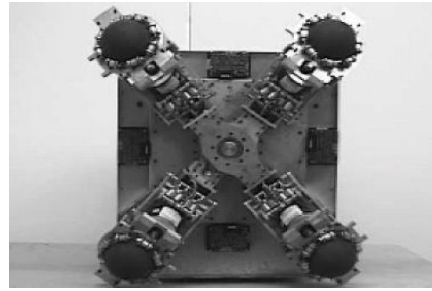


图 2-8 球轮底盘移动机器人

2.2 基于麦克纳姆轮的四轮系全方位移动平台机械设计

麦克纳姆轮的设计最初由瑞典人 Rengtllon 设计的。而全方位移动平台的设计则是在底盘上安装 4 个麦克纳姆轮，保证其中 2 个是右旋轮，2 个是左旋轮，如图 2-9 所示，同旋向的按照对角线放置。4 个轮子独立转动，每个轮子上面沿着轮廓，按照一定角度斜放着鼓形辊子。鼓形辊子具有旋转的自由度。车体的全方位移动就是通过这 4 个轮子的不同转向和转速的组合而实现的。轮子在电机的驱动下和地面之间发生滚动摩擦，而鼓形辊子直接和地面接触，地面给辊子的摩擦力就是驱动着车体移动的牵引力，方向为沿着辊子的轴线方向。

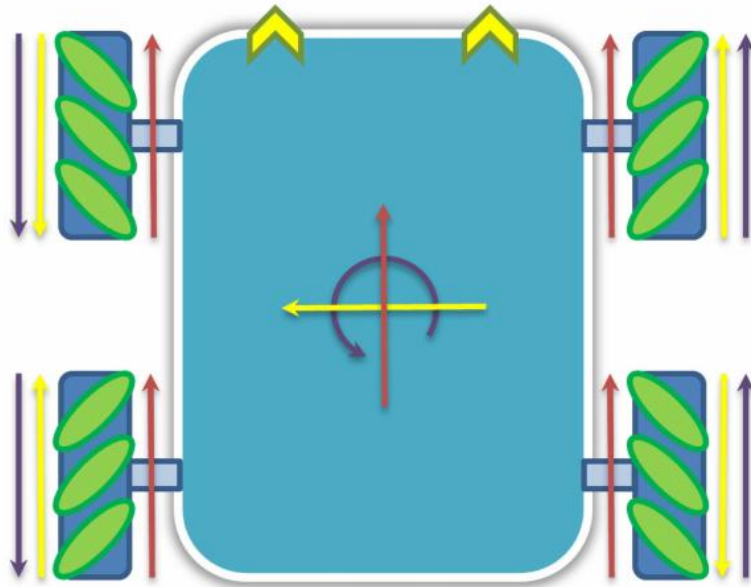


图 2-9 麦克纳姆轮布局情况

本次机械设计基于 SolidWorks 2013 平台，利用该软件建立本次设计的全方位移动底盘的三维模型，设计的车体如图 2-10 所示，车长 600mm，车宽 450mm，轮直径 150mm，底盘上平面距离地面 90mm。

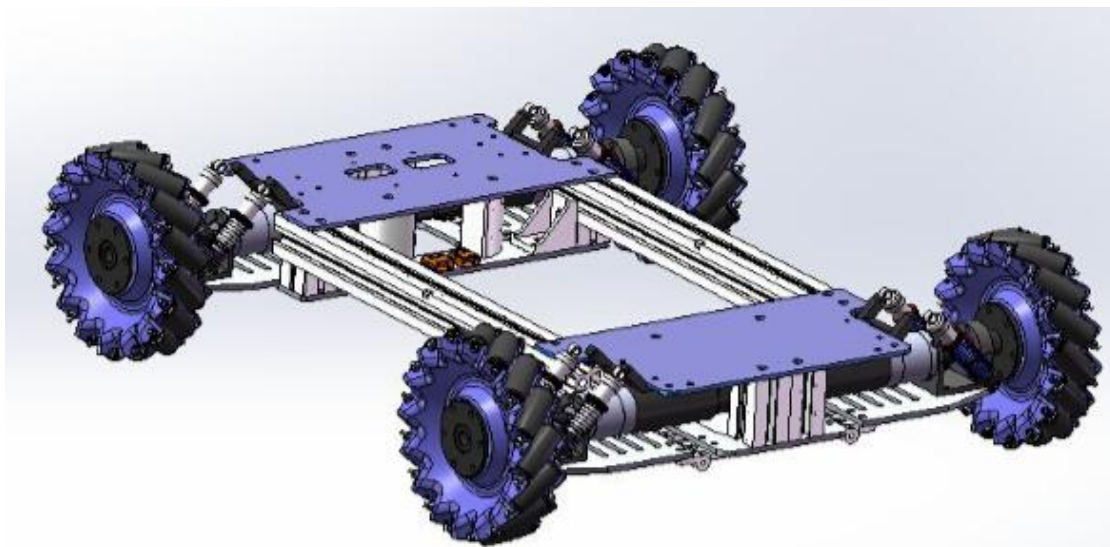


图 2-10 平台机械三维设计图

根据三维图纸绘制加工图纸，送至机械加工厂加工。在选材上由于车体骨架由于受力比较大采用铝合金，联轴器在电机频繁的换向时候极易损坏所以采用钢材，车前后驱动的上下车身采用碳纤维板，主要目的为了减少车身重量，减少电机的负载。实际加工组装后的效果图如图 2-11 所示。

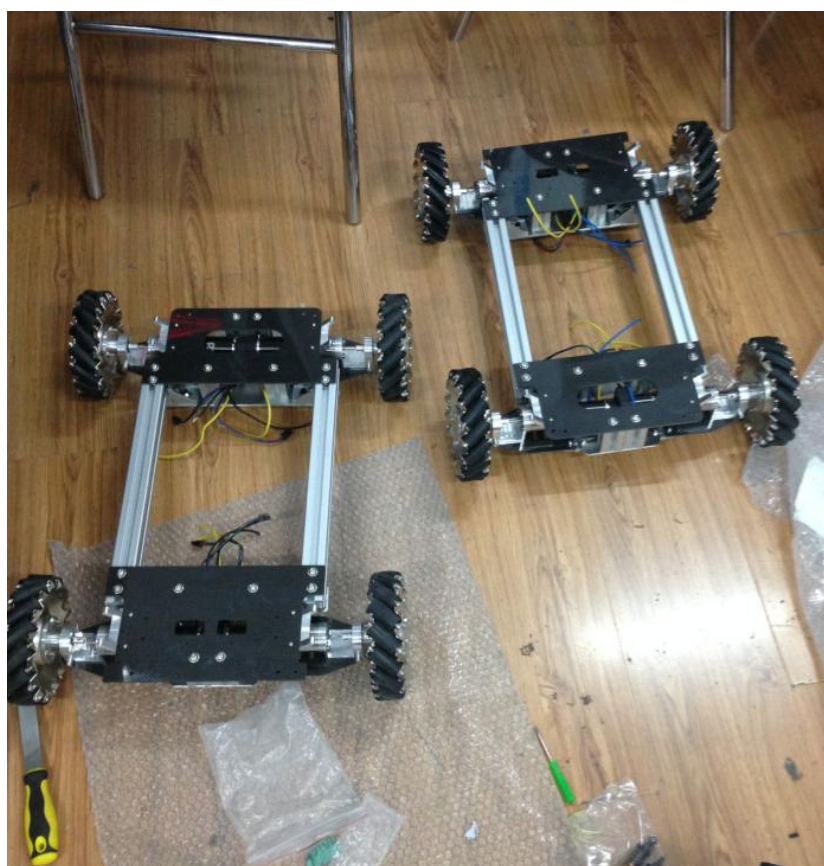


图 2-11 移动平台实际组装后效果图

2.3 全方位移动平台运动分析

如图 2-12 所示, 左侧为整车的一个坐标系, 右侧为轮 1 坐标系, 4 个麦克纳姆轮由 4 个电机驱动, 实现独立旋转, 由鼓形辊子的安装方向和转动方向决定了摩擦力的方向, 从而决定了车体的移动方向。设 4 个麦克纳姆轮的转动速度为 $\omega_1, \omega_2, \omega_3, \omega_4$; 整个车体的速度为 v_x, v_y, ω_0 ; 辊子的速度为 $v_{g1}, v_{g2}, v_{g3}, v_{g4}$ 。以轮 1 为例, O 为系统运动中心, O_1 为轮 1 轮毂中心。 XOY 为与车体运动中心固联的直角坐标系 $X_1O_1Y_1$ 为与轮毂中心固联的直角坐标系。

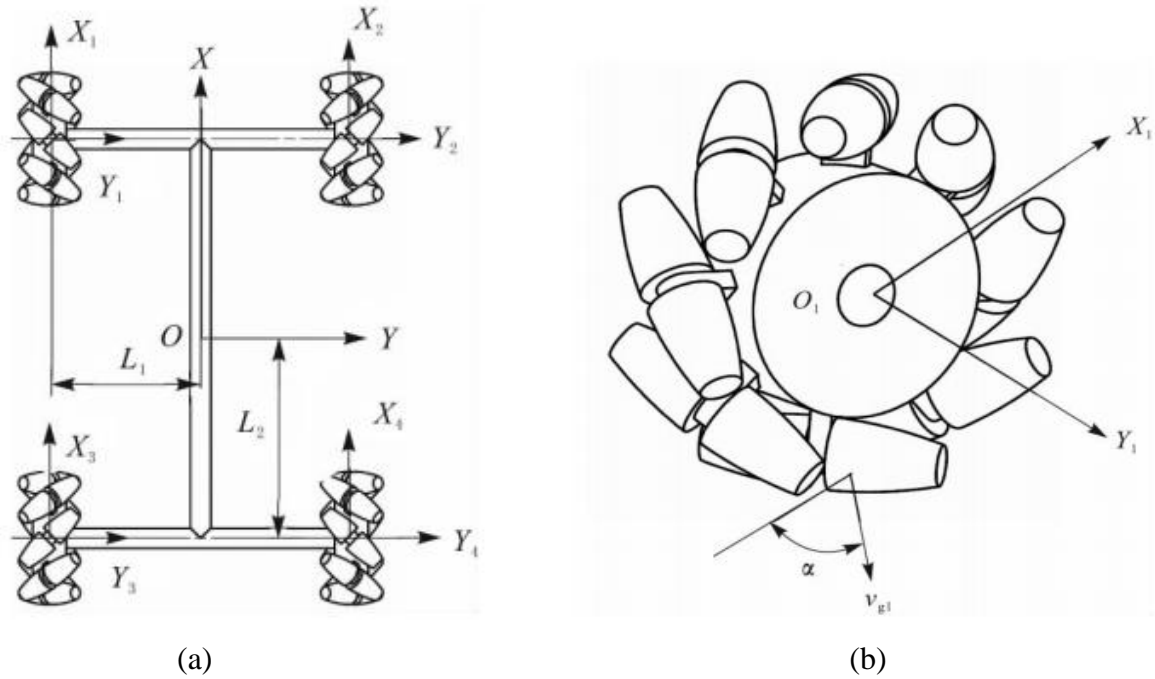


图 2-12 车体坐标系以及轮 1 坐标系

在与车体固联的 XOY 坐标系下, 轮 1 中心 O_1 点的移动速度为:

$$v_{O_1X} = v_x - \omega_0 L_1 \quad (2-1)$$

$$v_{O_1Y} = v_y - \omega_0 L_2 \quad (2-2)$$

在与轮 1 固联的 $X_1O_1Y_1$ 坐标系下, 轮 1 中心 O_1 点的移动速度为:

$$v_{O_1Y} = v_{g1} \sin \alpha \quad (2-3)$$

$$v_{O_1X} = -v_{g1} \cos \alpha + \omega_1 R \quad (2-4)$$

其中 v_{O_1X} 为轮 1 沿 x 方向移动速度, v_{O_1Y} 为轮 1 沿 y 方向移动速度, v_{g1} 为辊子的转速。由以上 (2-1) 至 (2-4) 公式联立得:

$$\begin{cases} v_x - \omega_0 L_1 = -v_{g1} \cos \alpha + \omega_1 R \\ v_y - \omega_0 L_2 = v_{g1} \sin \alpha \end{cases} \quad (2-5)$$

得:

$$\omega_1 = \frac{1}{R} \left[1 \quad \frac{1}{\tan \alpha} \quad - \left(L_1 + \frac{L_2}{\tan \alpha} \right) \right] \begin{bmatrix} v_x \\ v_y \\ \omega_0 \end{bmatrix}$$

化简得：

$$\omega_1 = \frac{1}{R} \begin{bmatrix} 1 & 1 & -(L_1 + L_2) \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ \omega_0 \end{bmatrix} \quad (2-6)$$

代入得到：

$$v_{g1} = \frac{v_x - v_y + \omega_0(L_2 - L_1) - \omega_1 R}{\cos \alpha + \sin \alpha} \quad (2-7)$$

同理分析其它轮，可得：

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix} = \frac{1}{R} \begin{bmatrix} 1 & 1 & -(L_1 + L_2) \\ 1 & -1 & (L_1 + L_2) \\ 1 & -1 & -(L_1 + L_2) \\ 1 & 1 & (L_1 + L_2) \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ \omega_0 \end{bmatrix} \quad (2-8)$$

$$\begin{bmatrix} v_{g1} \\ v_{g2} \\ v_{g3} \\ v_{g4} \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -1 & -L_1 + L_2 \\ 1 & 1 & L_1 - L_2 \\ 1 & 1 & -L_1 + L_2 \\ 1 & -1 & L_1 - L_2 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ \omega_0 \end{bmatrix} \begin{bmatrix} \omega_1 R \\ \omega_2 R \\ \omega_3 R \\ \omega_4 R \end{bmatrix} \quad (2-9)$$

由式 (2-9) 所示， v_1 ， v_2 ， ω_0 取不同值时候，车体可以实现全方位移动。当 $v_x=0$ ， $v_y=0$ ， $\omega_0 \neq 0$ 时车体可实现原地转动

$$\text{即：} \begin{cases} \omega_1 = -\omega_4 \\ \omega_2 = -\omega_3 \\ \omega_1 \neq \omega_2 \\ \omega_3 \neq \omega_4 \end{cases} \quad (2-10)$$

当 $v_x = v \cos \theta$ ， $v_y = v \sin \theta$ ， $\omega_0 = 0$ 时，车体就可以以 v 的速度沿与 X 轴成任意 θ 角运动，即达到全方位运动特点。

2.4 小结

本章首先对全向轮的种类加以简单的介绍，同时利用 SolidWorks 2013 软件设计基于麦克纳姆轮的四轮系全方位移动平台的三维图纸。同时深入的分析四轮系全方位移动平台的运动状态，对其建模构建状态表达式，得出每个轮子的速度与车体速度之间的函数表达，方便接下来的控制系统的设计。

3 运动控制系统的硬件设计

3.1 运动控制系统简介

运动控制的核心内容主要就是对于电机的控制。在一个控制系统中最有效的控制方式便是对运动源的控制，因此，在搭建控制系统结构时候，一般都是选用直接控制电机的方法进而达到控制被控对象的目的。

通常一个闭环的控制系统必不可少的便是信息反馈，常见被测对象有温度，湿度，压力，液体流速，物体转速，电压值等等，实现方法通常采用模拟电路将这些信号转化成电压信号，电压信号的大小进而反映这些被测对象的大小，所以便需要控制系统的控制芯片具备 AD 采集功能；运动控制系统中常用的执行机构为主要为电机，实现电机控制的简单方法为 PWM 信号控制方式，通过改变占空比的大小改变电机驱动器导通状态，从而改变电机两端电压，达到控制电机转速快慢的效果，所以便需要控制系统的控制芯片具备输出占空比可调的 PWM 功能。以上两点是构建运动控制系统的关键所在，所以实现运动控制系统的主流方法一般采用可编程控制器，主要有单片机微处理器，FPDA/CPLD 可编程逻辑器件，可编程 DSP 控制器，或者是专用的运动算法控制器，这些方法的特点在于：电路简单稳定，能实现复杂算法的运算，信号采集频率快精度高，通常支持多路 A/D 转换，多路脉冲捕捉功能和多路 PWM 输出功能。用可编程控制器搭建的运动控制系统在控制算法代码升级时简单容易，可维护性好扩展性高。

除了以上采用可编程控制器搭建运动控制系统之外，还有通过模拟电路的搭建进而实现控制系统的搭建，比如硬件 PID 控制器。这种方法由于使用元器件数目比较多，受温度影响比较大，控制器参数不稳定，温漂严重，同时难以升级系统，由于系统规模限制很难完成复杂控制算法的实现。

本次课题采用的方案便是基于单片机微处理器构建的全方位移动控制系统。

3.2 全方位移动平台控制系统结构

全方位移动平台的驱动与控制是机器人关键部分，能够直接的影响到平台上搭建的其他结构的灵活度以及机动性能。国内外常见的移动平台都是采用直流电机作为驱动源，配上合适的减速箱，能为整个移动平台带来可靠的带负载能力。本课题完成的便是对多电机控制系统的搭建，同时要求较高的实时性与稳定性。硬件系统设计总体框架如图 3-1 所示。

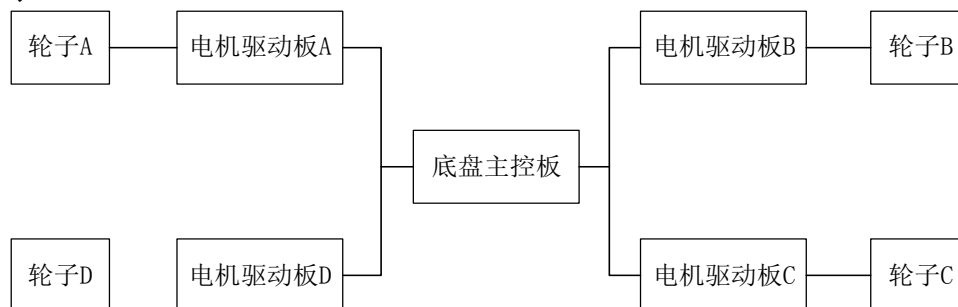


图 3-1 硬件结构系统设计

本次设计的移动平台硬件系统结构主要是底盘主控与底盘驱动之间的协调工作，如图 3-2 所示。包含了单片机最小系统模块、电源稳压模块、电机驱动模块，串口通讯模块、遥控接收模块以及传感器模块。通过各个模块的配合使用与驱动，完成全方位移动平台的硬件系统搭建。

单片机最小系统模块是构成控制系统的核心所在，整个算法控制代码都烧写在单片机芯片中，所以单片机最小系统稳定的工作是保证整个控制系统安全可靠运行

电源稳压模块采用开关电源以及线性电源结合使用方式。开关电源因为其具备效率高，驱动电流大，带负载能力强功耗大的特点，用来保证整个系统的正常稳定的运行，防止因为功耗过大而发热。线性电源因为具备电压纹波小，安全性比较高的特点，用来给单片机芯片供电，保证了芯片的稳定运行，防止出现复位，跑飞等现象。

移动平台的电机驱动方式采用常见的 PWM 信号控制方式，电机驱动采用 MOSFET 搭建的全桥驱动器，能够实现正反转以及速度控制。

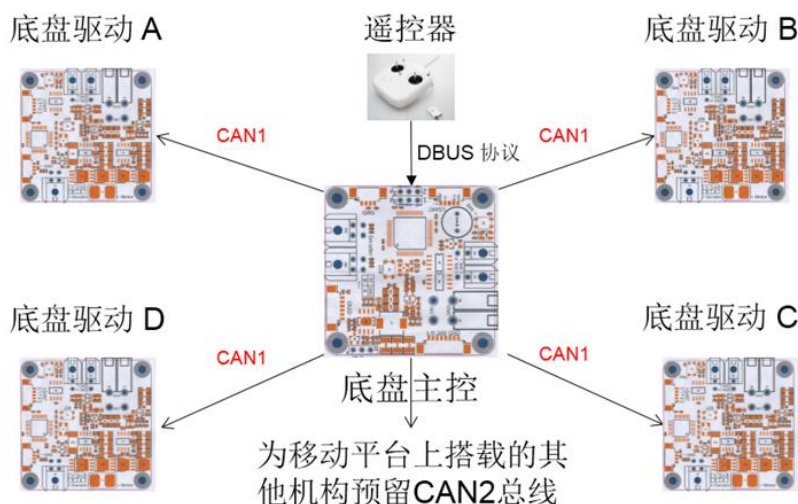


图 3-2 全方位移动平台硬件控制系统结构图

3.3 核心器材选型

在设计之初要有一个宏观的设计方案，考虑选择何种处理器、电源以及电机，这三个要素决定之后硬件电路的设计方案以及设计内容。

3.3.1 处理器芯片的选择

本课题芯片选择如下：底盘主控采用 STM32F405RGT6 芯片作为核心处理器，底盘驱动采用 STM32F103C8T6，能够优异的完成全方位移动平台算法控制。

意法半导体公司生产的芯片，从 STM8 到 STM32 产品的升级改进，从 72MHz 处理频率到 168MHz 处理频率，仍然能够保持其不变的稳定性。内存从 64K 到 256K 闪存，嵌入式 SRAM 从 20K 到 64K，我们看到了 STM 系列的提高。最近几年生产的系列大多采用 64 脚或者 100 脚的封装形式，管脚数量的一致性成为 STM32 控制器的设计理念，最小的硬件变化完全可以满足开发人员进行对程序性进行能提高与优化。

STM32 系列芯片外设极为丰富，像常用的定时器、串行口、CAN 总线、SPI、IIC 接口，并且接口大多配备 DMA 功能，可以大大释放总线工作负荷，并采用相同功能的集成化接口，接口的共用性大大的节省了开发时间并且提高了 STM32 系列芯片的兼容性，程序员可以在多个程序中重复使用同一个软件。

F405 系列控制器大多数使用 3.3V 电压供电，方便大多数外设的使用。主频可以达到 168MHz 的高性能工作状态，在低功耗模式下维持 2mA 工作电流依然能够保证正常的运行。睡眠唤醒仅需要 6 微妙的时间，足以和 TI 的低功耗产品相匹敌。STM32F405RGT6 芯片，具备了输入输出的绝大部分功能，并且高性能的主频速度，能够良好的完成底盘主控控制算法的运行。

F103 系列控制器在性能上稍微逊色于 F405 系列，但是涵盖了脉冲捕捉功能，CAN 总线通讯以及多路 PWM 脉冲输出功能。这些功能是底盘驱动必须具备的，用以完成编

码器脉冲采集，底盘驱动与底盘主控之间的 CAN 通讯以及控制电机转速。所以选用 STM32F103C8T6 芯片足以完成底盘驱动的任务。

3.3.2 电源选择

目前在航模和遥控赛车中常用的能源供给方式主要有电池供电以及柴油供给，电池供电方案相对烧油来说能更好维护更干净安全一些，同时在选择电源时候按照以下原则：

- (1) 大容量，可持续工作 30 分钟以上；
- (2) 可充电，重复使用；
- (3) 重量轻，体积小，减少移动平台负重，方便布局。

基于以上三点原则，本次设计选用航模中常用的动力电池，使用两块 3 芯动力电池串联使用，构成 24V 电源。单块电池为 5200mA，为整个全方位移动底盘提供了可靠安全的动力。单个电池如图 3-3 所示。

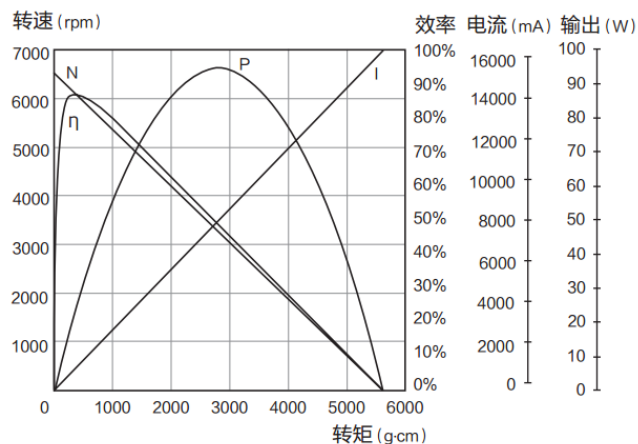


图 3-3 选用的电池

3.3.3 电机选择

电机作为执行机构，其性能的优劣直接影响到全方位移动平台的灵活度，速度，以及负重能力。所以电机的选型即为重要。

本次选用的电机为 DJI 公司定做的 RM35 电机，该电机是一款为 5kg 到 20kg 轮式机器人量身定做的底盘驱动电机，具有高效率、高可靠性、低噪点的特点，可为全方位平台提供充足的动力。如图 3-4 所示，图中给出不同转矩下对应的电机转速、电流、输出以及效率的变化曲线。



注：N—转速，I—电流，P—输出， η —效率

电机减速箱减速比 1:16

图 3-4 电机参数性能图

3.4 控制系统硬件电路设计

3.4.1 稳压电源设计

电源是保证硬件系统正常工作的基础，设计中要详细的分析：系统能够提供的电源输入；单板需要产生的电源输出；各个电源需要提供的电流大小；电源电路效率；各个电源能够允许的波动范围；整个电源系统需要的上电顺序等等。

在弱电控制系统中，传感器、各类电子元器件通用的电源电压为 5V 或者 3.3V，所以如何从 24V 稳压到 5V 和 3.3V 是本次电源设计中的关键所在。电源设计解决方案分为两种，第一种为开关电源，第二种为线性电源。

开关电源具有效率高，开关频率快，带负载能力强的特点，比较适合用来做控制板的第一级稳压，原理图如图 3-5 所示，稳压芯片采用 MP2482DN，芯片前端的 C18、C19、C20、C21 起到去耦滤波的作用，同样的芯片后端 C23 至 C26 也起到相同作用，调节电阻 R18 与 R20 的比值进而能够改变输出的电压值，按照图中的参数（电阻 R18=40.2K，电阻 R20=7.68K）输出端调至 5V 输出，该开关电源的设计能输出近 3A 的电流，为整个系统带来可靠的带负载能力。

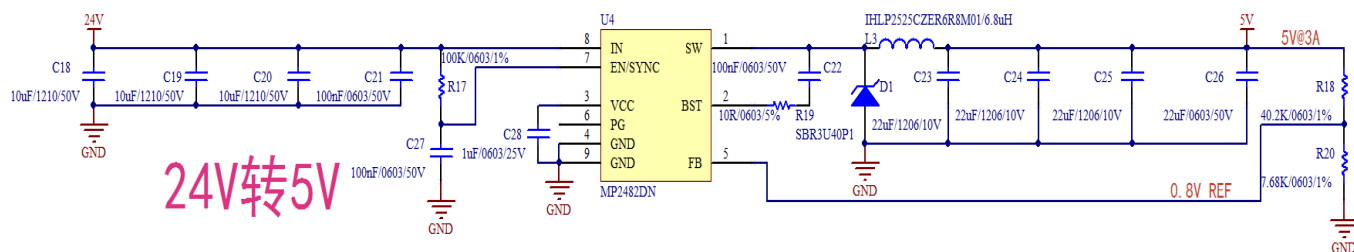


图 3-5 24V 开关电源

线性电源具有电路简单容易搭建，元器件少，输出纹波小安全可靠较高的特点，而且像 STM32F405RGT6 芯片和 STM32F103C8T6 芯片的电源需要一个输出纹波相对平滑、稳定可靠的电源，因此采用线性电源做 5V 稳 3.3V 的方案。线性稳压电源比较常见的有 LM1117、TPS7333、SPX5205M5 等等稳压芯片，本次采用的是 SPX5205M5 稳压芯片，主要因为其封装小，电路组成元器件少，印制的 PCB 结构紧凑体积小，原理图如图 3-6 所示，由于线性稳压芯片片内就能完成稳压功能，故一般只需要在芯片的输入端和输出端加上旁路电容和滤波电容即刻，所以图中的 C29、C30、C31、C32、C33 都起到滤波去耦等作用，目的是为了输出纹波更小，抗干扰能力更强。

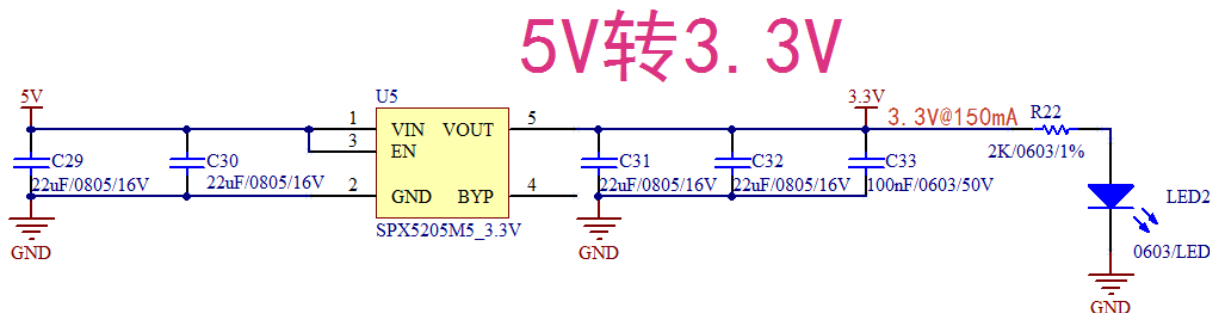


图 3-6 3.3V 线性电源

3.4.2 单片机最小系统设计

单片机最小系统是保证单片机芯片工作的最小单位电路，其涵盖了芯片供电、复位

电路、外部时钟电路和下载调试电路^[10]。如图 3-7 所示，芯片 5、6 引脚外接 25M 无源晶振构成时钟电路，第 7 引脚 RST 为低电平复位方式，故采用按键 S1 按下引脚接地复位，按键 S1 松开电容 C4 上端电压为高电平停止复位。

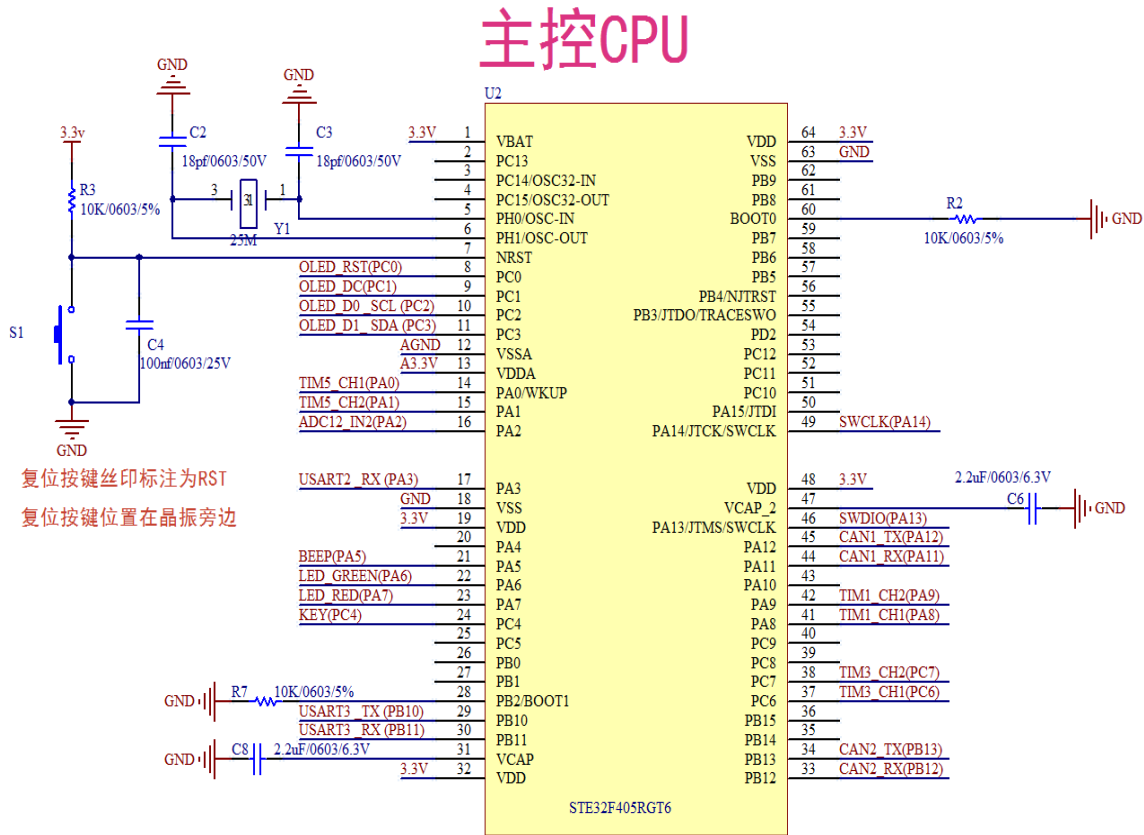


图 3-7 单片机最小系统原理图

3.4.3 全桥驱动器的设计

电机驱动是整个硬件设计的关键所在，决定整个全方位移动平台的性能的好坏。其设计的过程中主要考虑以下几点：1、在功能上是单向转动还是双向转动，是否需要调速控制；2、在性能上需要考虑输出电流和电压范围，它决定能驱动多大功率的电机，本次设计方案采用 PSMN1R8-40YLC 芯片，该芯片为恩智浦半导体生产的 N 沟道 MOSFET 管，其漏源两极导通电流最大能达到 100A，工作电压最高达到 48V，能够提供一个高效率的输出；3、在安全上需要考虑 H 桥相同侧桥臂导通引起的电源正负两端短接，这是全桥驱动设计中的问题，解决此问题一般需要一个驱动芯片，驱动芯片有全桥式也有半桥式，在功能上两个半桥驱动芯片等同于一个全桥驱动芯片。本次设计采用 IRS21867S 半桥驱动芯片作为 MOS 管导通控制器。

全桥驱动器的优点在于能实现正反转的速度控制，一般控制信号为 PWM 信号。改变 PWM 信号的占空比，便可控制 MOS 管导通时间，进而控制电机两端的平均电压，从而改变了电机的转速^[11]。一般 PWM 信号的频率达到 10K 到 50K 左右，所以人的感官上分辨不出在电机在高频的使能与失能。理论上来说频率越高，输出效率越高。但是由于驱动芯片的开关频率以及 MOS 管导通的开关频率限制，总会有一个饱和值。

整体设计如图 3-8 所示，4 片 MOS 管构成的全桥驱动电路，控制信号由控制器给出，送至 U1，U2 的 HIN 和 LIN。此驱动电路的使用方法：使电机正转时候 U1 的 LIN 控制信号为 1 同时保证 U2 的 LIN 为 0，改变 U1 的 HIN 端的 PWM 信号占空比，从而改变正转速度；使电机反转则 U1 的 LIN 控制信号为 0 同时保证 U2 的 LIN 为 1，改变

U2 的 HIN 端的 PWM 的占空比，从而改变反转速度。注意一点就是两片半桥驱动芯片的 LIN 端的电平一定为反极性，不能同时为 1 或者同时为 0，否则导致 H 桥上同侧的上桥与下桥同时导通，整个 24V 电源与地短接，烧毁驱动器。图中 D1 与 C2、D2 与 C6 分别构成自举电路，本设计自举电压接入 18.5V，经过实际实验自举电压越大，控制端可给的最大占空比越高，此自举电压最高可给 24V。

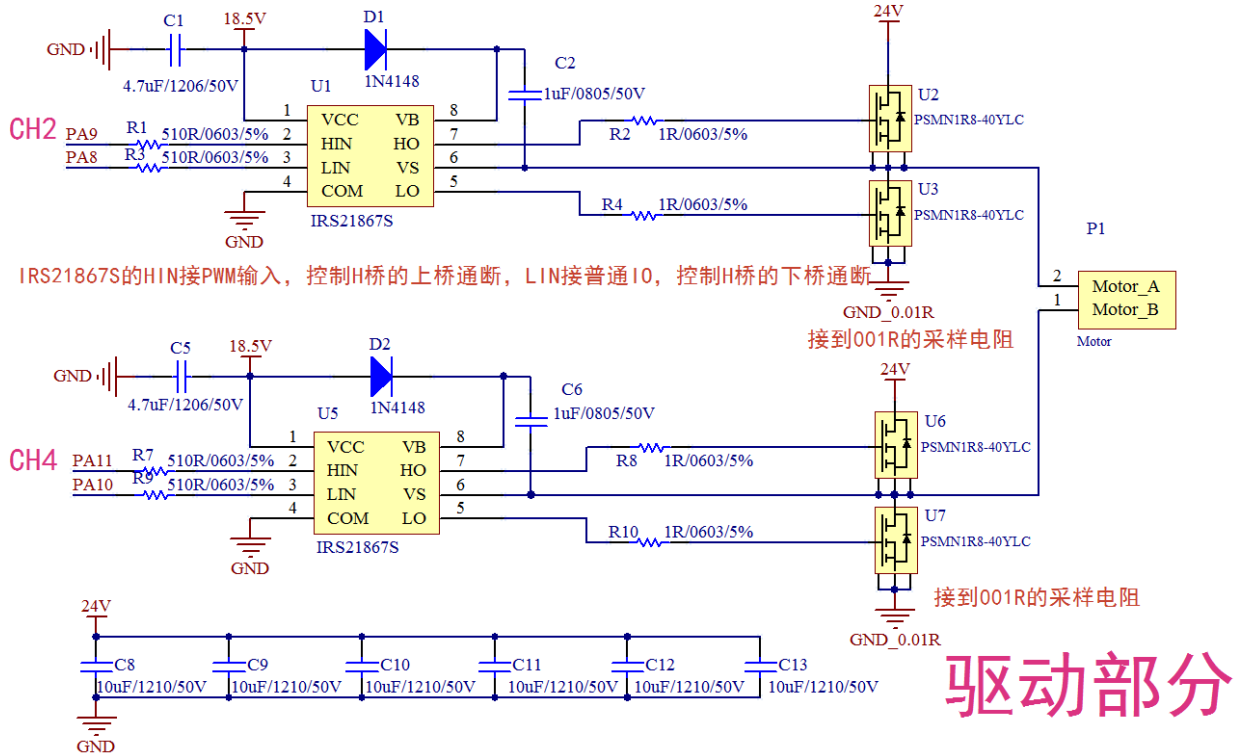


图 3-8 全桥驱动电路

3.4.4 过流保护检测电路

平台在运行过程中难免会遇到突发事故，就比如装到障碍使轮子堵转，堵转会使得输出电流增大时间久了会使电机发热、MOS 管发烫，甚至烧毁电机与电路，所以过电流检测电路显得必不可少。如图 3-9 所示，所有的 MOS 管上的电流都要经过采样电阻 R13 留至 GND，所以检测 R13 上的电压大小就可以反映出 H 桥上流过的电流。采用的是轨对轨运放 LMV611 的运算放大器电路。不过此电路存在缺陷，输入偏置电压典型值 1mV，最大为 4mV，导致采样电阻上即便是没有电流流过，12 位 ADC 采集可能会读到 0-0.8V 的电压。但是此方案并不影响正常的使用，可以在软件上设置好阈值，保证大电流流过时候启动保护电机程序即可。

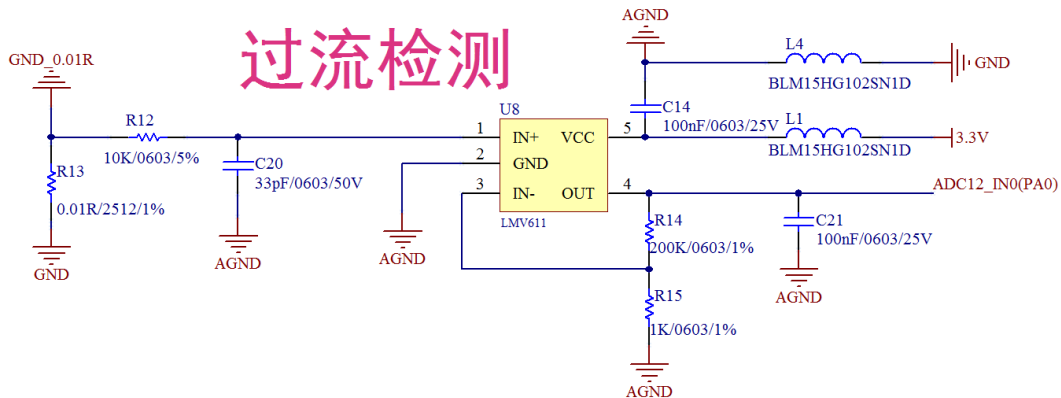


图 3-9 过流检测电路

3.5 小结

本章的主要内容是对全方位移动平台的硬件设计做详细的介绍以及深入的分析。在原理图设计过程中要有“拿来主义”，现在的芯片厂家一般都可以提供参考设计的原理图，所以要尽量借助这些资源，在充分理解参考设计的基础上，做一些自己的发挥。当主要的芯片选定以后，最关键的外围设计包括了电源，时钟和芯片间的互连。

本次课题硬件设计部分主要完成以下几点：

- (1)、电源设计包含 5V 开关电源和 3.3V 线性电源，其中开关电源；
- (2)、底盘主控和底盘驱动的单片机最小系统设计；
- (3)、MOS 全桥电机驱动器设计；
- (4)、过留检测电路设计。

硬件设计中不仅仅给出了电路原理图，同时详细的分析各个部分电路参数的变化对整个电路性能的影响。

4 全方位移动平台软件设计

机械与硬件是全方位平台的可视基础，软件以及控制算法是全方位平台的内在灵魂，三者相辅相成，缺一不可，三者相互配合构成一个完整的控制系统。相对于硬件设计来说，软件设计比较灵活一些，也为控制系统的设计带来方便。

4.1 软件设计思想与整体架构

常见的移动机器人控制主要有两种方式：

- (1) 全自动方式 机器人上电后，自主完成相应动作，但是一般都是设定好的运动轨迹，比较单一，想适应新的环境需要重新设定运动轨迹，不够灵活。
- (2) 半自动方式 机器人上电后，人通过肉眼视觉上的反馈人为的主导车的运动方向运动速以及运动轨迹。这种方法比较灵活，在各种新的复杂场地都能得到良好的运行。

为了适应各个复杂环境，本次设计方案采用半自动方式，通过遥控器下达全方位移动平台的运动方向运动速度以及运动轨迹，这种方法能让全方位平台灵活的穿梭在各个狭小空间，控制起来比较灵活方便。

根据对全方位移动底盘运动控制系统的要求，设计软件流程图如图 4-1 所示。

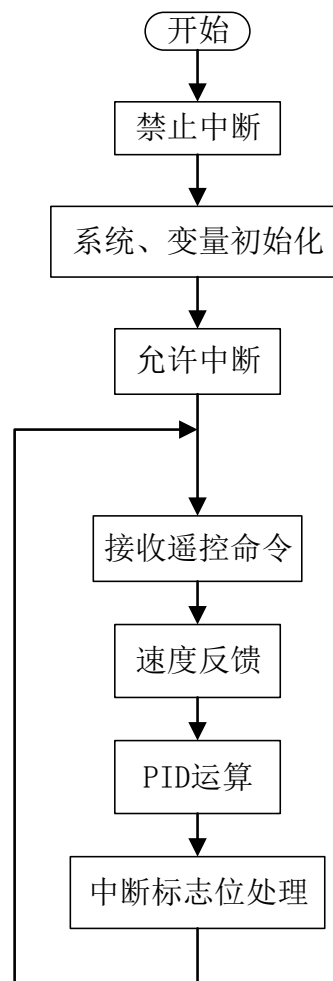


图 4-1 软件控制流程图

4.2 遥控器解析协议

遥控器的主要作用有以下两点：

- (1)、移动平台控制的使能、失能；
- (2)、平台的运动姿态，包含运动方向、运动轨迹的规划以及运动速度的给定。

遥控器选用 DJI 公司提供的 2.4GHz 遥控器，具体型号为 DT7，接收端为 DR16。遥控器接收机输出信号为标准的 DBUS 协议数据，当接收机和发射机建立连接后，接收机会每隔 7ms 通过 DBUS 发送一帧数据（18 字节），DBUS 的通信参数如表 4-1 所示，需要注意一点 DBUS 信号控制电平符合 TTL，但是和普通 UART 信号是相反的，所以需要在 MCU 端需要增加三极管取反电路，MCU 才能正常识别出 UART 信号

表 4-1 DR16 接收机通信参数

DBUS 参数	数值
波特率	100Kbps
单元数据长度	8
奇偶校验位	偶校验
结束位	1
流控	无

遥控器的遥感通道信息如图表 4-2 所示，只有了解了信道数据形式，才能编写正确的通讯协议。

表 4-2 遥控器信息

域	通道 0	通道 1	通道 2	通道 3	S1	S2
偏移	0	11	22	33	44	46
长度 (bit)	11	11	11	11	2	2
符号位	无	无	无	无	无	无
范围	最大值 1684	最大值 1684	最大值 1684	最大值 1684	最大值 3	最大值 3
	中间值 1024	中间值 1024	中间值 1024	中间值 1024	最小值 1	最小值 1
	最小值 364	最小值 364	最小值 364	最小值 364		
功能	无符号类型 遥控器通道 0 控制信息	无符号类型 遥控器通道 1 控制信息	无符号类型 遥控器通道 2 控制信息	无符号类型 遥控器通道 3 控制信息	遥控器发射机 S1 开关位置 1: 上 2: 下 3: 中	遥控器发射机 S2 开关位置 1: 上 2: 下 3: 中

根据上表的遥控信息，编写的数据结构体如下：

```
#pragma pack(1)
typedef union
{
    struct
    {
        struct
        {
            uint8_t  ch0_h:8;           //!< Byte 0
            uint8_t  ch0_l:3;           //!< Byte 1
            uint8_t  ch1_h:5;
            uint8_t  ch1_l:6;           //!< Byte 2
            uint8_t  ch2_h:2;
            uint8_t  ch2_m:8;           //!< Byte 3
        }
    }
};
```

```

uint8_t  ch2_l:1;           //!< Byte 4
uint8_t  ch3_h:7;
uint8_t  ch3_l:4;         //!< Byte 5
uint8_t  s1:2;
uint8_t  s2:2;
}rc;
struct
{
    int16_t  x;             //!< Byte 6-7
    int16_t  y;             //!< Byte 8-9
    int16_t  z;             //!< Byte 10-11
    uint8_t  press_l;       //!< Byte 12
    uint8_t  press_r;       //!< Byte 13
}mouse;
struct
{
    uint16_t  v;            //!< Byte 14-15
}key;

uint16_t  resv;            //!< Byte 16-17
};
uint8_t  buf[18];         //!< Union --> Byte<0-17>
}RC_Ctl_Define_t;

```

4.3 CAN 通讯

本次方案中模块与模块之间的通信方式采用 CAN 总线通信方式，主要因为其通信速率快、可扩展性强、程序开发简单效率高。

CAN 全称为 Controller Area Network，又称控制器局域网，属于现场总线的范畴，最初的设计是专门为汽车电子行业开发的一种串行通信总线。^[12]目前由于它的通信速度快、调试简单方便、工作可靠稳定、使用灵活和性价比高的优点，已经在汽车、船舶、航天、工业控制、国防安全等等领域中得到了广泛的应用。

在总线中传送的报文，每帧由 7 部分组成。CAN 协议支持两种报文格式，这两种报文格式唯一的不同的仅仅是标识符(ID)长度不同，标准格式的标识符为 11 位长度，扩展格式的标识符为 29 位长度^[14]。

CAN 的 ID 称为它的识别符，CAN 总线中，就是用标示符来确定发送端的优先级。^[15]当同时有多个节点发送数据时，标示符低的将获得总线的优先发送权，其他节点进入接收状态，等待总线再次空闲时，再次比较优先级，进入下一次发送。需要指出的时，CAN 的发送信息是以广播的形式，一个节点发送数据，总线上所有的节点都能够接收到，只是根据标示符，是不是在自己的接收队列中，决定是否接收该数据^[16]。另外，CAN 的 ID 实际上是对于消息的标示。当然，目前我们使用的 CAN-ID 把它对应到具体的一个节点上，也能实现我们的功能。

在使用 CAN 通讯时，每个节点都对应一个 CAN-ID，分为主从机，主机上的不设置过滤规则，接收总线上所有的数据。从机设置只接受主机发送的数据，通过查找响应的 ID 号，从而实现 CAN 通讯。

在 STM32f405 单片机中定义 CAN 通讯发送端和接收端的结构体如下，其中 StdId 为 0x00-0xFF 之间的地址段，在全方位平台控制系统中，底盘驱动与底盘主控之间的通讯就是通过查询 StdId 地址来获得响应的数据，数据存储于 Data[8]中。

发送包数据结构体:

```
typedef struct
{
    uint32_t StdId;
    uint32_t ExtId;
    uint8_t IDE;
    uint8_t RTR;
    uint8_t DLC;
    uint8_t Data[8];
} CanTxMsg;
```

接收包数据结构体:

```
typedef struct
{
    uint32_t StdId;
    uint32_t ExtId;
    uint8_t IDE;
    uint8_t RTR;
    uint8_t DLC;
    uint8_t Data[8];
    uint8_t FMI;
} CanRxMsg;
```

定义的四个电机驱动 CAN 通讯地址为:

```
#define BASE_MOTOR_A_ID 0x01
#define BASE_MOTOR_B_ID 0x02
#define BASE_MOTOR_C_ID 0x03
#define BASE_MOTOR_D_ID 0x04
```

4.4 底盘速度分解程序

正如 2.3 小节中的运动分析得到的速度分解表达式 (2-9) 得知, 通过各个轮子之间的速度方向以及速度大小的匹配进而完成整个车体的运动姿态。所以本方案采用底盘主控得到遥控器给定目标速度以及目标轨迹, 通过速度分解得到每个轮子的转速以及转向, 再通过 CAN 通信发送给各个 ID 地址的底盘驱动, 每个底盘驱动获得自身的给定, 根据给定进行速度闭环控制, 进而完成指定任务。

速度分解程序是依据 2.3 小节中的公式 (2-9) 设计的, 先给出一个变量的使用说明

(1) 函数入口传入遥控给定的信息, 结构体内包含三个方向上的给定量

Direction_X (X 方向上的给定值)

Direction_Y (Y 方向上的给定值)

Direction_W (旋转方向上的给定值)

(2) 函数输出得到分解后的各个轮子的速度期望值

Speed_Set.Motor_A_Speed_Want (轮 A 速度期望)

Speed_Set.Motor_B_Speed_Want (轮 B 速度期望)

Speed_Set.Motor_C_Speed_Want (轮 C 速度期望)

Speed_Set.Motor_D_Speed_Want (轮 D 速度期望)

程序代码如下:

```
void SpeedDistribute(MOTOR_BASE_CONTROL *Motor_Base_Control)
{
    Speed_Set.Motor_A_Speed_Want=-(-Direction_X - Direction_Y - Direction_W);
    Speed_Set.Motor_B_Speed_Want=+(+Direction_X - Direction_Y + Direction_W);
    Speed_Set.Motor_C_Speed_Want=+(-Direction_X - Direction_Y + Direction_W);
    Speed_Set.Motor_D_Speed_Want=-(+Direction_X - Direction_Y - Direction_W);
}
```

4.5 速度闭环控制

控制系统的核心思想便是消除误差，使被控量达到期望值，并稳定在期望值上下波动。而一个好的控制系统需要满足响应快、抗干扰能力强、稳定性高、震荡小等要求。

本次方案设计的闭环调速系统如图 4-2 所示，每个电机都根据自身的给定值独自闭环。工业上常用的控制算法为 PID 控制，PID 又分位置式和增量式。而电机的控制需要采用增量式控制，主要因为增量式 PID 每次计算的是相对于上一状态的增量值，该增量的变化相对与位置式 PID 每次的输出量更加连续一些，对电机输出的扰动比较小，控制起来比较连续稳定。

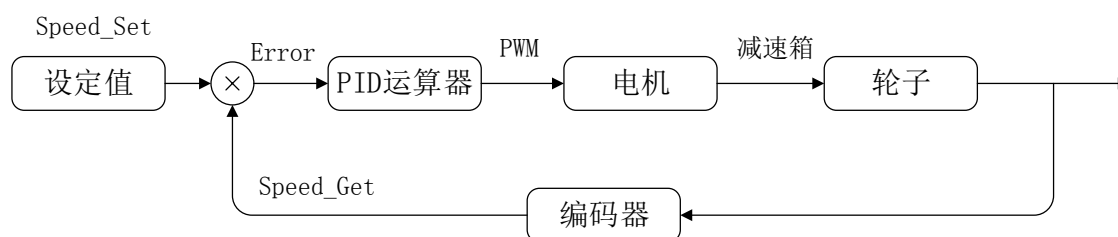


图 4-2 单轮速度闭环控制系统

4.5.1 PID 控制原理

最常用的控制规律就是 PID 控制，常规的 PID 控制系统原理框图如图 4-3 所示，系统由模拟 PID 控制器和被控对象组成。

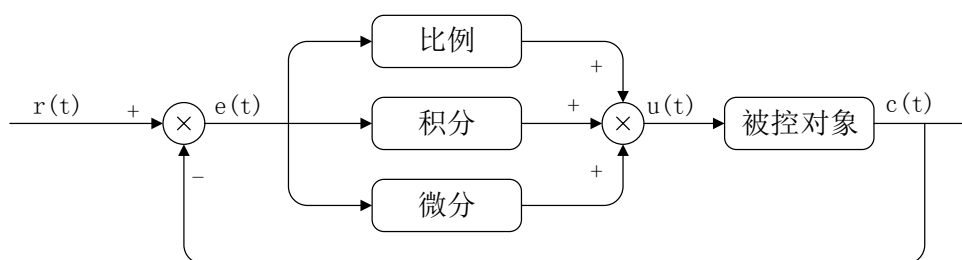


图 4-3 模拟 PID 控制原理

PID 控制器的偏差方程式为：

$$e(t) = r(t) - c(t)$$

PID 控制规律为：

$$u(t) = k_p(e(t) + \frac{1}{T_i} \int_0^t e(t) dt + \frac{T_d de(t)}{dt})$$

其中 k_p 为比例系数， T_i 为积分时间常数， T_d 为微分时间常数。

比例调节 k_p 的作用：该项系数乘在偏差项上，所以控制系统一旦产生偏差，比例调节就立即起到作用抑制偏差。比例参数越大，系统调节速度越快，控制器越硬，但是过大之后，系统发生震荡失去稳定

积分调节 T_i 的作用：积分项用以消除系统的稳态误差的，通常参数不宜过大。积分项的强弱通常取决于控制周期的长短，周期越小积分作用越强，反之越小^[18]。加入积分项会使控制系统滞后。

微分调节 T_d 的作用：微分项作用于系统偏差的变化率，有一定的预判性，起到一个超前控制的作用，改善了控制系统的动态性能^[19]。

4.5.2 数字 PID

单片机等处理器处理的都是数字信号，并且都是采样控制方式，只能根据采样时刻的反馈计算 PID 输出量，所以需要模拟 PID 进行离散化，需要离散的变量为积分项和微分项，式 4-1 为增量式 PID 型控制算法^[20]。

$$\Delta u(k) = u(k) - u(k-1) = k_p(e(k) - e(k-1)) + k_i e(k) + k_d(e(k) - 2e(k-1) + e(k-2)) \quad (4-1)$$

其中 k_p ：比例系数， k_i ：积分系数， k_d ：微分系数， $\Delta u(k)$ ，当前状态相对上次状态的增量， $e(k)$ ：当前时刻偏差， $e(k-1)$ ：上一状态时刻的偏差， $e(k-2)$ ：上上一状态的偏差。

4.5.3 PID 的参数整定

设计一个控制器的途径有很多种，就比如通过极点配置和状态观测器来设计反馈控制系统，但是这两种方法通常需要对控制系统建模，同时要详细的知道各个环节的理论参数，一旦有一个环节改变，就需要重新设计控制器，不够灵活效率低。

当被控对象的结构和参数不能完全把握，或得不到精确的数学模型时，控制理论的其它技术难以采用时，系统控制器的结构和参数必须依靠经验和现场调试来确定，这时应用 PID 控制技术最为方便。

在 PID 参数进行整定时假如能够有理论的方法确定 PID 参数当然是最理想的方法，但是在实际的应用中，更多的是通过凑试法来确定 PID 的参数。而试凑又不是随心所欲的，往往通过理解 k_p 、 k_i 、 k_d 三个参数的变化对控制系统性能的影响来做的。

增大比例系数 k_p 一般将加快系统的响应，在有静差的情况下有利于减小静差，但是过大的比例系数会使系统有比较大的超调，并产生振荡，使稳定性变坏。

增大积分时间 k_i 有利于减小超调，减小振荡，使系统的稳定性增加，但是系统静差消除时间变长。

增大微分时间 k_d 有利于加快系统的响应速度，使系统超调量减小，稳定性增加，但系统对扰动的抑制能力减弱。

根据以上调试方法进行 PID 参数整定，红色为期望速度，黄色为实际转速，完成轮子的速度闭环控制。



图 4-4 轮子转速控制曲线

4.6 小结

本次方案中的软件设计主要完成以下几个任务：

(1)、对遥控器信道的解析，获得底盘使能、失能、速度给定量以及底盘运动轨迹等信息。

(2)、CAN 通讯协议包设计，定义各个轮子的电机驱动器的 CAN 通讯 ID 地址。

(3)、底盘速度分解函数，通过遥控器给定的整车运动状态，分解出各个轮子的转速以及转向。

(4)、PID 闭环控速系统设计，分析 k_p 、 k_i 、 k_d 的变化对整个控制系统的影响，同时进行参数整定，最终得到效果比较理想的速度控制器。

程序整体的框架总结为：底盘主控获得遥控器给定的速度期望，进行速度分解计算得到四个轮子各自的速度期望，再通过 CAN 通信方式将四个轮子的转速发送给不同 CAN-ID 地址的底盘驱动，底盘驱动接收到自己的运动信息，进行 PID 运算，完成速度闭环控制，从而实现全方位移动底盘按照遥控的给定量进行不同速度不同路径的运动。

5 云台部分

5.1 整体机械结构设计

常见的云台通常为两轴云台，分别为 Pitch 轴、Yaw 轴，结合本次参赛任务要求，同时根据个人经验，我们决定舍弃 Yaw 轴，设计成仅有 Pitch 轴控制枪管角度的机械结构，如图 5-1 至 5-3 所示，实际图如图 5-4 所示。其中摩擦轮采用外直径 60mm 的橡胶轮，提供很好的线速度，增大子弹威力。

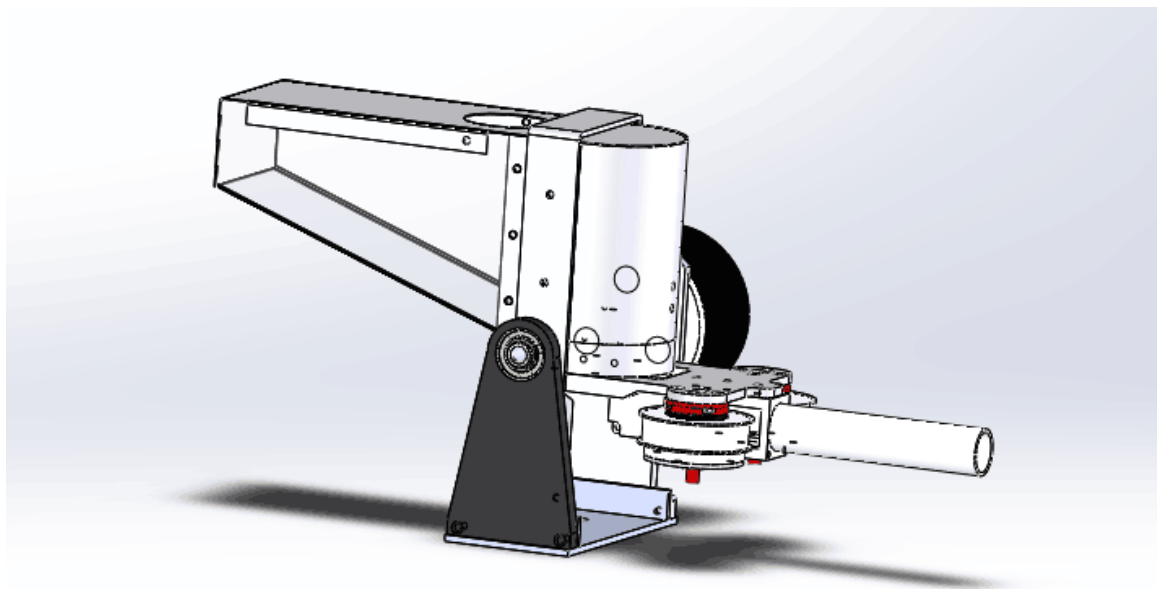


图 5-1

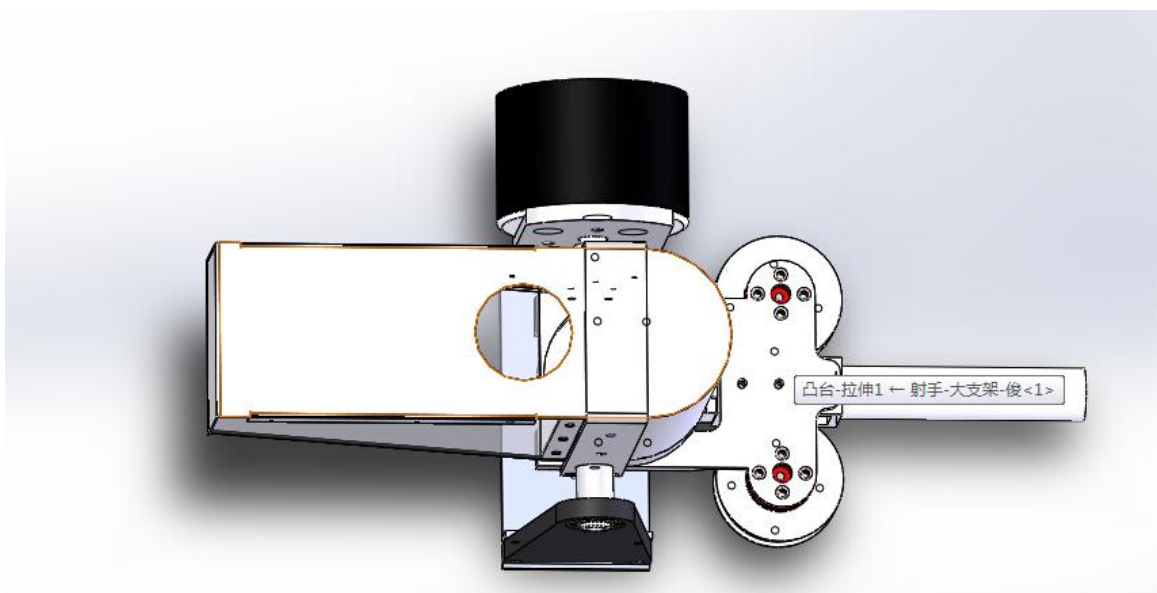


图 5-2

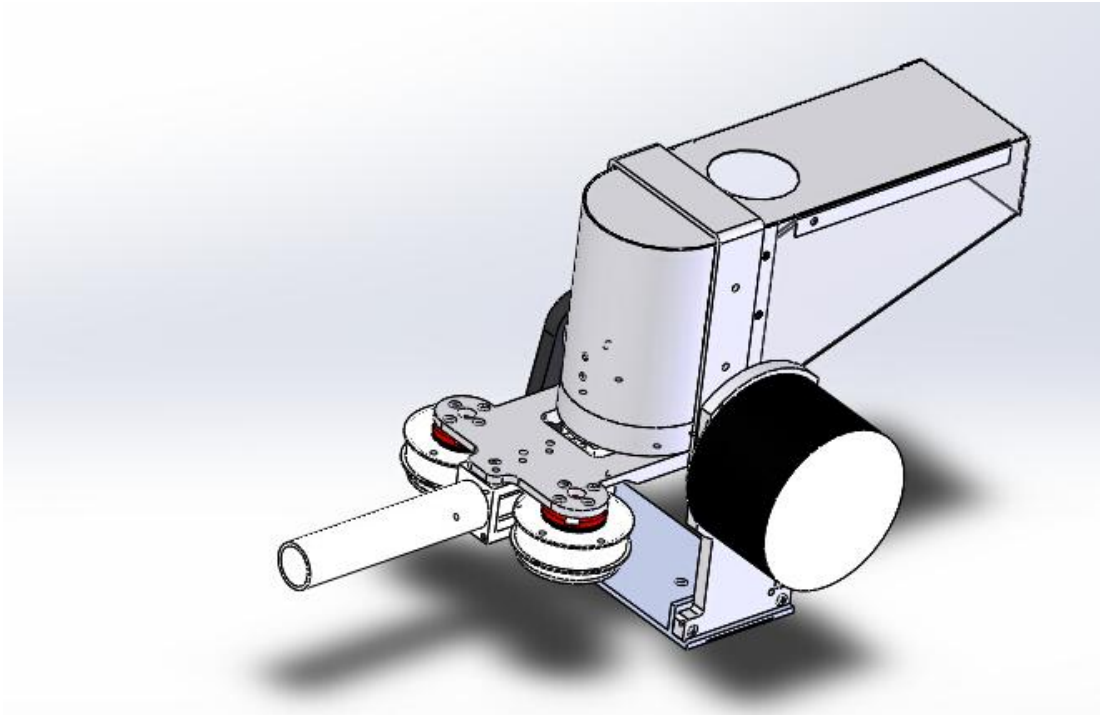


图 5-3

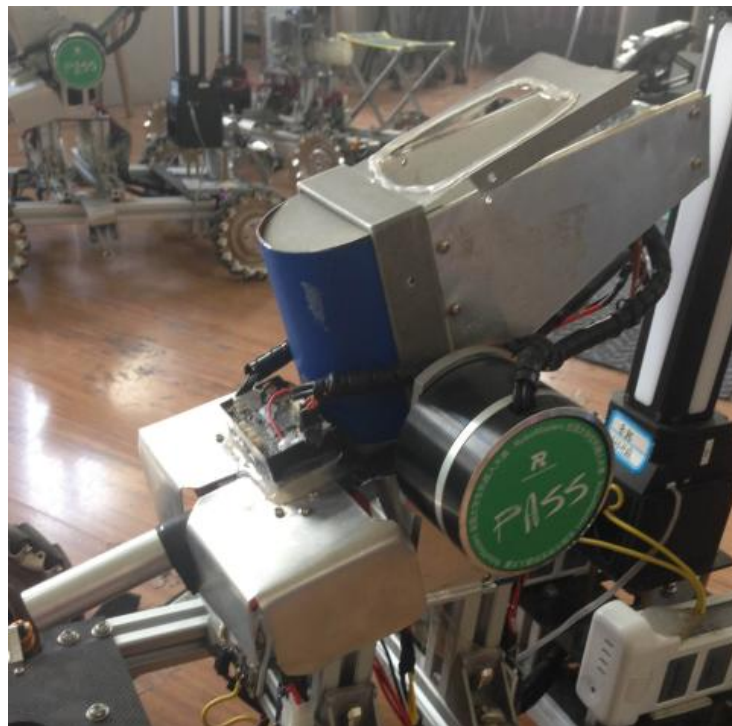
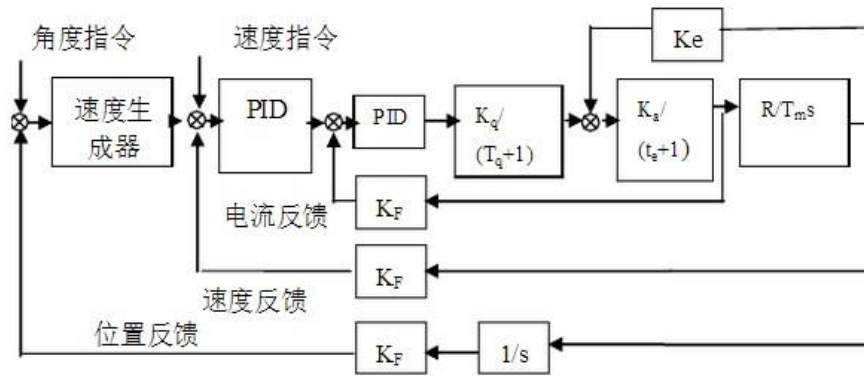


图 5-4 云台整体机械结构

5.2 云台控制

云台调的性能好坏，直接影响战车敌我对战时候的响应性能，决定机器战车打子弹时候的灵活性。所以本次云台控制主要采用双环控制系统，系统框图如下：



内环为速度环，外环为位置环，速度反馈是采用 MPU6050 X 轴的陀螺仪得到当前变化速度，位置反馈采用编码器反馈得到角度位置。

先给出一个变量的使用说明

- (1) 函数入口传入遥控给定的信息，结构体内包含三个方向上的给定量
 - this_201_angle（编码器得到的反馈值）
 - target_pitch_angle（遥控器给定的目标位置）
 - position_201_output（位置环算出的输出值）
 - MPU6050_Real_Data.Gyro_X（X轴方向上的陀螺仪反馈）
 - velocity_201_output（云台电调最后电流输出）

程序代码如下：

```
position_201_output = Position_Control_201(this_201_angle, target_pitch_angle);
```

```
velocity_201_output
```

```
= Velocity_Control_201(MPU6050_Real_Data.Gyro_X, position_201_output);
```

- (2) Velocity_Control_201（）函数为速度环PID，函数代码如下：

```
float Velocity_Control_201(float current_velocity_201, float target_velocity_201)
{
  const float v_p = 25; // 60.0;
  const float v_d = 12;
  static float error_v[2] = {0.0, 0.0};
  static float output = 0;
  if (abs(current_velocity_201) < GAP)
  {
    current_velocity_201 = 0.0;
  }
}
```

```
error_v[0] = error_v[1];
error_v[1] = target_velocity_201 - current_velocity_201;
output = error_v[1] * v_p + (error_v[1] - error_v[0]) * v_d;
if(output > ESC_MAX)
{
    output = ESC_MAX;
}
if(output < -ESC_MAX)
{
    output = -ESC_MAX;
}
return output;//cyq:for6015 反向
}
```

Position_Control_201 () 函数为位置环 PID。函数代码如下：

```
float Position_Control_201(float current_position_201,float target_position_201)
{
    const float l_p = 30;//20.5;
    const float l_i = 0.01;
    const float l_d = 30;
    static float error_l[2] = {0.0,0.0};
    static float output = 0;
    static float inte = 0;
    error_l[0] = error_l[1];
    error_l[1] = target_position_201 - current_position_201;
    inte += error_l[1];
    output = error_l[1] * l_p+ inte * l_i+ (error_l[1] - error_l[0]) * l_d;
    if(output > ESC_MAX)
    {
        output = ESC_MAX;
    }
}
```



```
}  
  
if(output < -ESC_MAX)  
{  
    output = -ESC_MAX;  
}  
  
return output;  
  
}
```

5.3 云台调试

由于本设计采用串级 PID 控制器，必然设计到 PID 参数的整定使得云台控制性能的提高。如果采取 Keil 软件在线调试参数，不能对 PID 的阶跃响应曲线进行直观的观察，从而无法精确快速的整定出合适的参数。开发一款在线观察波形和在线调试 PID 参数的上位机软件，是很有必要的步骤。

上位机功能主要包括以下几个方面：

1. 串行口的自动识别与用户选择；
2. 四通道数据波形显示与设定；
3. 四通道 PID 窗口制作；
4. 云台姿态 3D 模型仿真。

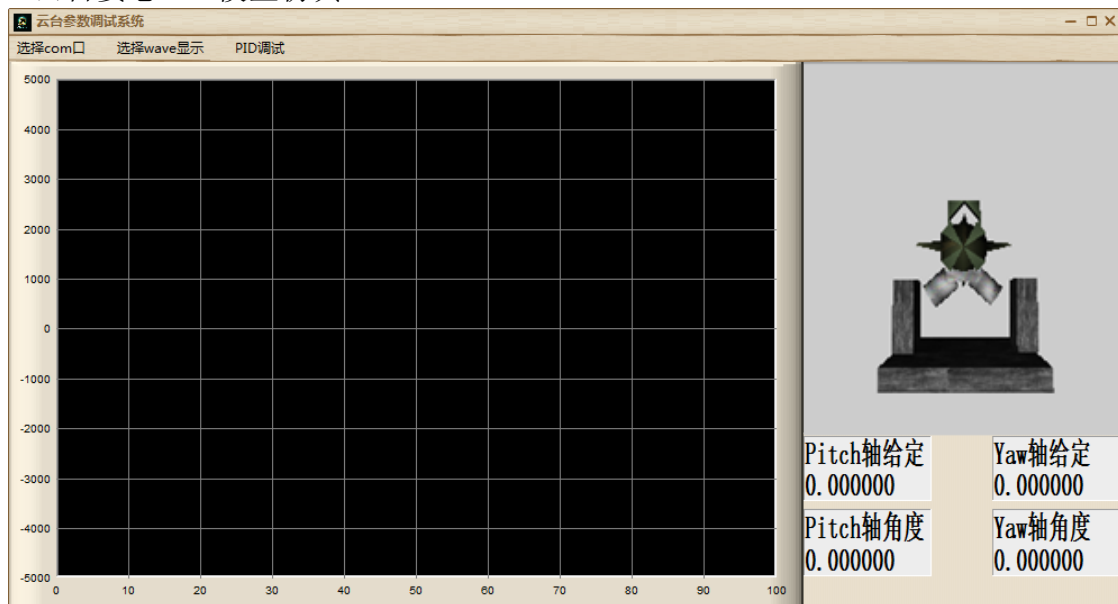


图 5-5 上位机主界面

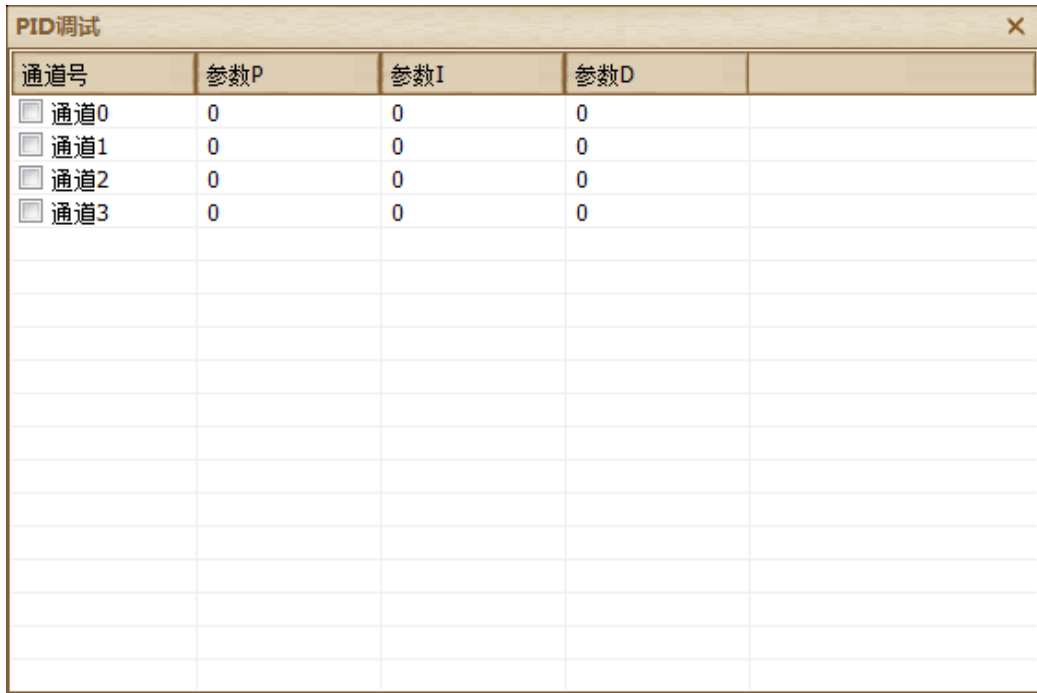


图 5-6 上位机 PID 调试窗口

5.3.1 通信协议

由于通道发送数据类型均为浮点型，数据量相对较大，而且 UART 为半双工通信方式，本身又存在不稳定性。我们必须对收发数据制作一套稳定的通信接口，如下图。

帧头	数据包长度	帧头包校验	数据	总校验
0X5A	LEN	CRC8	DATA	CRC16

数据帧头: 0x5A。转化为二进制数据 0101 1010 我们可以观察到，对于二进制传输的 UART，这样定义帧头会轻松的检测出错误数据报文。

数据包长度: 可以告之接收端，发送数据包的大小容量，方便进行数据提取。

帧头包校验: 即使帧头定义为 0x5A 使侦错概率大大提高，但是成千上万个数据包中，总可能出现偶然的情况，对帧头包校验，增加侦错率。

数据: 用户自定义传输内容，本设计为 28 个字节数据。

总校验: 对总数据长度进行 CRC 校验，是一个完整数据包必不可少的步骤，CRC 校验基本可以百分之百的实现数据包侦错。

5.3.2 PID 调试

如图 5-7，如果通过上位机只给定 PID 参数 P，我们发现普通 PID（图右）作用下的控制系统的输出震荡十分的严重，云台基本无法达到期望姿态。然而对于串级 PID（图左）单单给定内外环参数 P，云台经过短暂震荡后，可以较快的达到相对稳定状态。可见内环可以起到消除震荡的效果，和第三节中介绍的串级 PID 原理相对应。

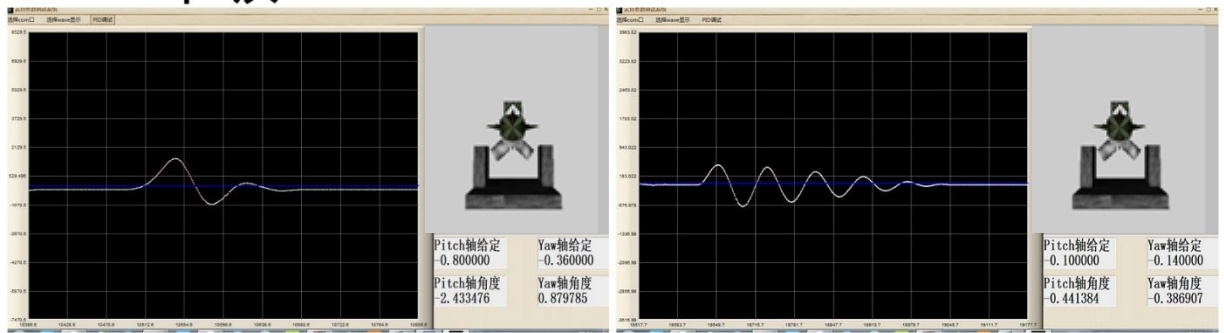


图 4-5 串级 PID 与普通 PID 对比 P 控制

如图 5-8, 当控制系统加上 D 参数后, 系统明显稳定了许多, 响应速度得到了很大的提升。无论是普通 PID (图右) 还是串级 PID (图左) 都可以达到良好的控制, 这与 D 参数作用对象选用陀螺仪输出值有很大关系。陀螺仪输出可以反映物体实际运动姿态, 弥补了本身控制器的滞后性。当时我们发现普通 PID 的输出波形有一定的积分特性, 可以看出系统的响应速度还是不如串级 PID 迅速, 也再次验证了串级 PID 具有提高系统响应的能力。

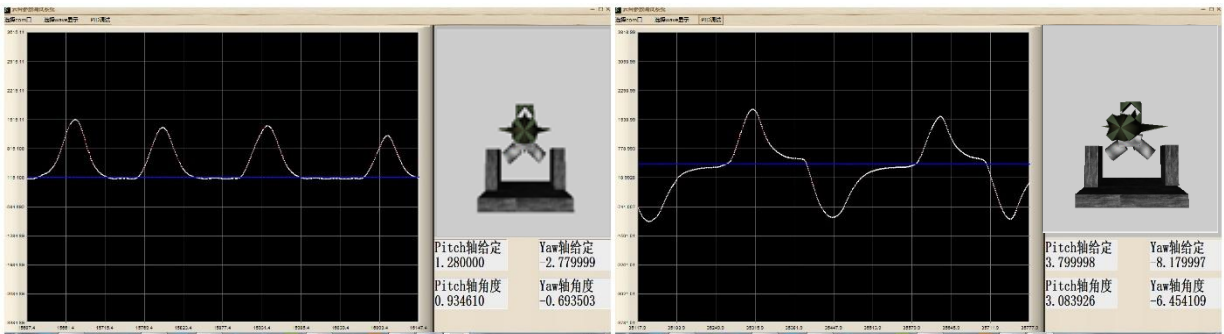


图 5-8 串级 PID 与普通 PID 对比 PD 控制

如图 5-9, 引入 I 系数后, 无论是普通 PID (图右) 还是串级 PID (图左), 给定姿态和实际姿态基本数值相同, 可见积分项的引入, 的确消除了稳态误差。通过和图 4-7,4-8 的对比, 发现稳态误差明显减少了很多, 给定姿态和实际姿态基本一致, 在允许的误差范围之内, 从而实现稳定的云台控制系统。但是从输出曲线中我们可以看出串级 PID 的波形响应速度明显高于普通 PID 的, 串级 PID 的输出达到稳定的时间基本为普通 PID 的一半, 再次表现出其优越性。

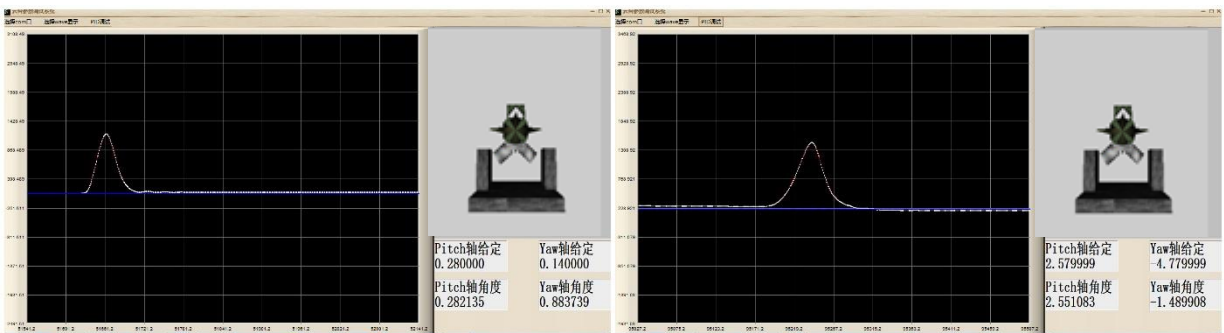


图 5-9 串级 PID 与普通 PID 对比 PID 控制

通过 PID 参数调试, 我们体会到了 P,I,D 三个参数在提升控制系统性能中起到的作用, 更加体会到串级 PID 的优势在哪里。

6 爬杆机器人设计

6.1 机械

机械设计本着最简单电机数量最小的原则设计的，本次采用两个电机完成前进已及抱合以及攀爬功能。最初由两个电机沿着前进方向提供动力，使整个爬杆机器人向前前进，内衬机构顶住柱子，然后动力轮仍然提供动力，使机器人两侧向内靠近合拢，最后靠永磁体锁住，然后动力轮提供向上动力，机器人开始攀爬。设计图如图 6-1 至 6-3 所示。

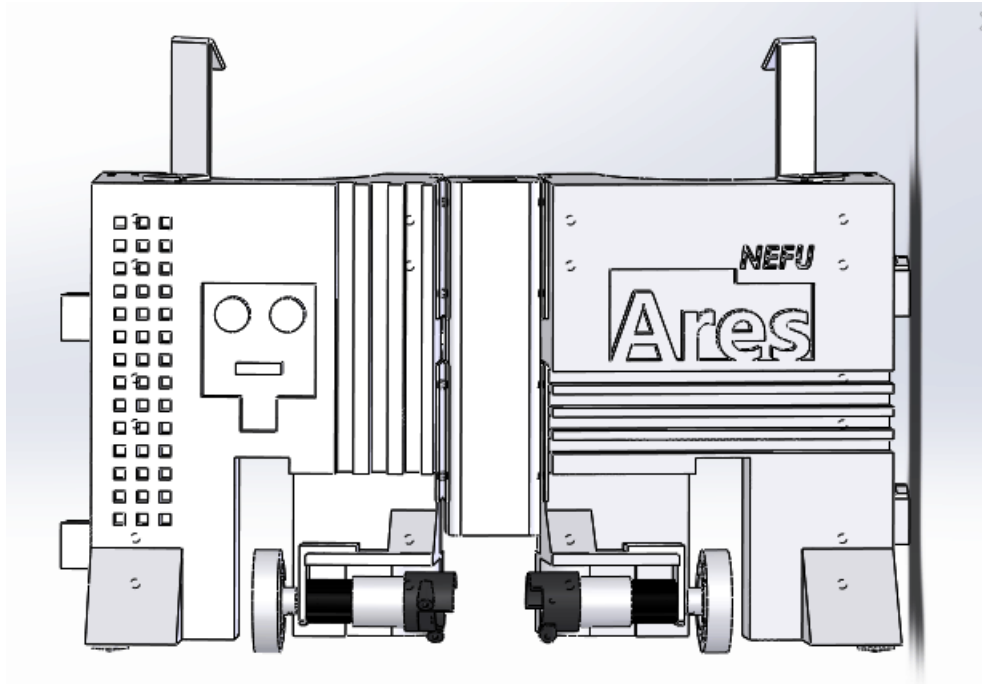


图 6-1 展开图

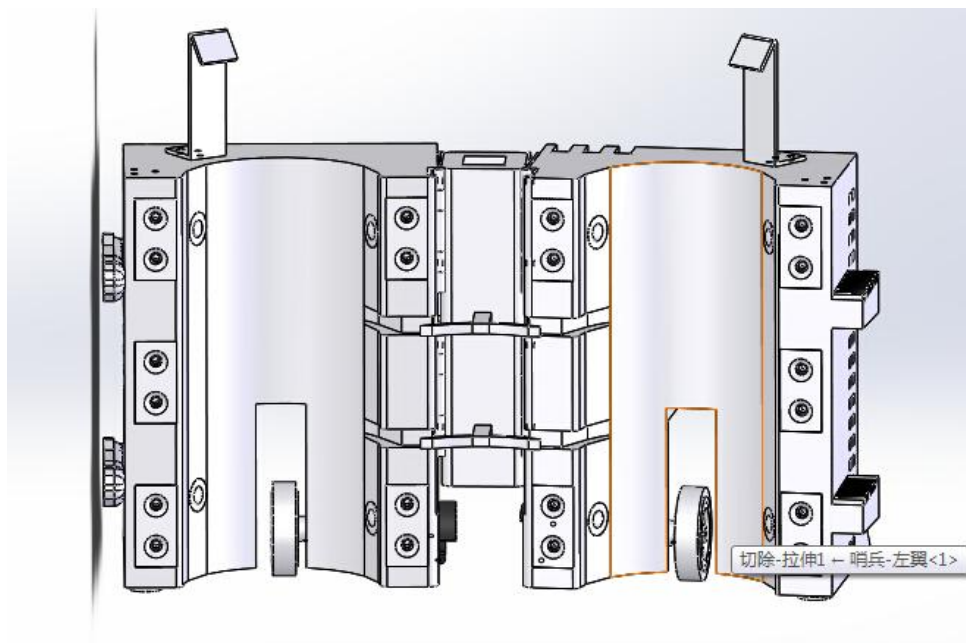


图 6-2 内视图

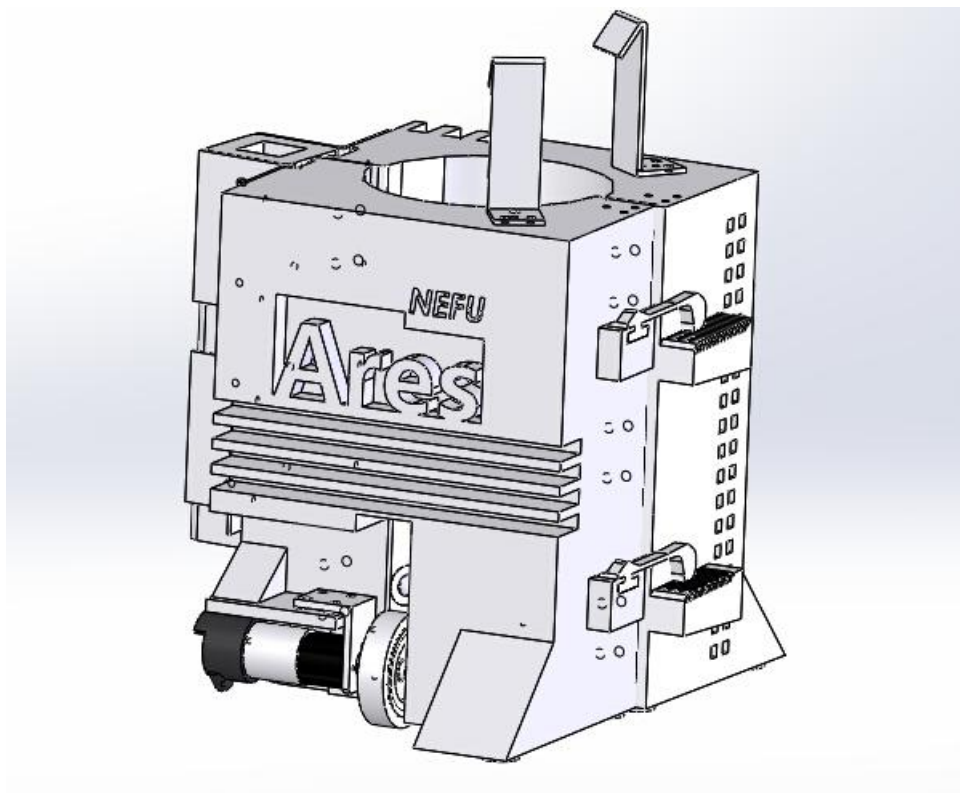
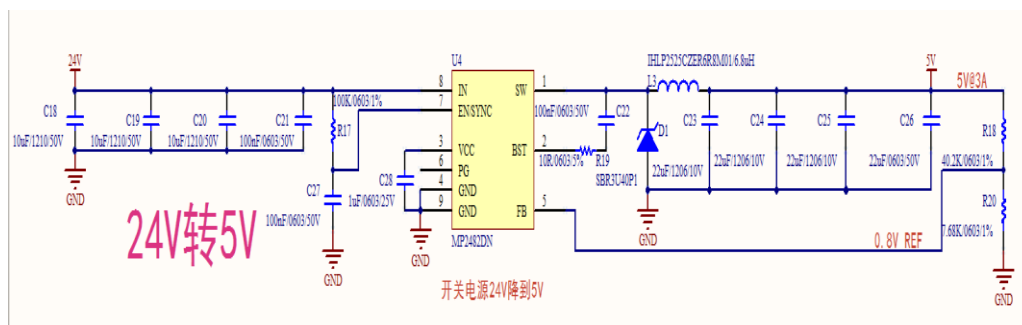


图 6-3 抱紧图

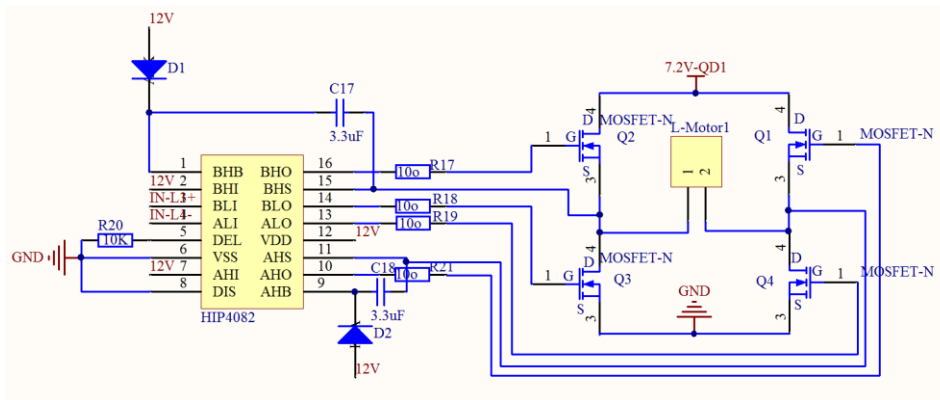
6.2 硬件

硬件上采用飞思卡尔 MK60 芯片为主控制器。供电电池采用动力电池 12V，故需要做一些电源模块，本次方案中最初采用的线性电源 LM2940 芯片做的 12V 稳 5V 电源，但是缺点在于，功耗太大，发热特别严重。经过查阅相关资料，得知开关电源适合做相关的功能，故采用 MP2482DN 芯片搭建的开关电源，原理图如下：



该开关电源特点为，驱动电流达到 3A，而且芯片发热量很小，5V 稳压纹波很小，使用起来比较乐观。

由于使用的动力系统为直流电机，故驱动器采用 MOS 管搭建的全桥驱动。驱动桥芯片采用 HIP4082，MOS 管为 LR7843。该驱动器能达到 120A 的大电流工作条件，为整个爬杆机器人的动力系统带来更可靠的动力方式。



6.3 程序

程序的核心控制为 PID 控制直立电机。采用的是增量式 PID 控制算法，相对于位置式 PID 更适合做电机的控制。核心控制代码如下：

```
#include "include.h"
#include "app.h"
Speed_Ctl_s ls, rs;
extern uint8 Stop_Flag;
extern uint8 Start_Flag;
/* 脉冲采集 */
void GetMotorPulse(void)
{
    ls.Speed.Real = FTM_QUAD_get(FTM1);
    ls.Speed.Real = abs(ls.Speed.Real);
    FTM_QUAD_clean(FTM1);
    rs.Speed.Real = FTM_QUAD_get(FTM2);
    rs.Speed.Real = abs(rs.Speed.Real);
    FTM_QUAD_clean(FTM2);
}
//速度控制
void Speed_Ctl(void)
{
    ls.Speed.Want = 100;
    rs.Speed.Want = 100;
    if( Start_Flag == 0)
    {
        ls.Speed.Want = 110;
        rs.Speed.Want = 90;
    }
}
```

```
}
if( Stop_Flag == 1)
{
    ls.Speed.Want = 0;
    rs.Speed.Want = 0;
}
ls.PID.P = 3;
ls.PID.I = 0.3;
ls.PID.D = 1;
rs.PID.P = 3;
rs.PID.I = 0.3;
rs.PID.D = 1;
ls.PID.Err = ls.Speed.Want - ls.Speed.Real;
rs.PID.Err = rs.Speed.Want - rs.Speed.Real;
ls.Speed.Out += ls.PID.P * (ls.PID.Err - ls.PID.PErr) +
ls.PID.I * ls.PID.Err + ls.PID.D * (ls.PID.Err - 2 * ls.PID.PErr
+ ls.PID.PPErr);
rs.Speed.Out += rs.PID.P * (rs.PID.Err - rs.PID.PErr) +
rs.PID.I * rs.PID.Err + rs.PID.D * (rs.PID.Err - 2 *
rs.PID.PErr + rs.PID.PPErr);
ls.PID.PPErr = ls.PID.PErr;
ls.PID.PErr = ls.PID.Err;
rs.PID.PPErr = rs.PID.PErr;
rs.PID.PErr = rs.PID.Err;
if(ls.Speed.Out > 950) ls.Speed.Out = 950;
if(ls.Speed.Out < 0) ls.Speed.Out = 0;
if(rs.Speed.Out > 950) rs.Speed.Out = 950;
if(rs.Speed.Out < 0) rs.Speed.Out = 0;
FTM_PWM_Duty(FTM0, FTM_CH2, ls.Speed.Out);
FTM_PWM_Duty(FTM0, FTM_CH6, rs.Speed.Out);
}
```

6.4 总结

整体实验效果还是非常不错的，但是还是有一些问题需要克服的，存在问题在于在攀爬过程中会有旋转，导致爬到杆顶端时候角度旋转了。解决此问题主要是因为两个轮子在抱紧柱子时候，不能保证完全呈 180 度直线，需要不断的纠正修改。其次一个问题是攀爬速度不够快，但是正是慢的原因，在攀爬过程中，视野也比较清晰，不会因为过快导致视野缺失。整体的一个实物效果图如下：



结论

本文主要做了以下工作：

1、首先介绍了全方位移动平台的发展，在各类机器人比赛中的出色表现，以及生活中、工业上、军事上的应用。

2、简单的介绍了全方位轮的种类以及各自的特点。同时设计了基于麦克纳姆轮的四轮系全方位移动机器人，给出车长车高等尺寸。深入的分析全方位移动平台的运动学、动力学的数学模型，加以推导分析，得出每个轮子与整体车身姿态速度关系，为程序上的速度分解带来有效的理论依据。

3、详细的介绍了全方位移动平台的硬件设计，包含了电池、电机、处理器的选择，并给出该选择方案的原因。对硬件电路设计做一个非常详细的讲解，包括开关电源设计、线性电源设计、最小系统设计、全桥驱动器设计以及过流保护电路设计，并深入的分析到各个部分的重要元器件所起的作用，为他人参考带来方便。

4、在软件设计方便，交代了整个平台采用遥控形式的控制策略，详细的介绍了遥控的通讯协议；底盘主控单片机与底盘驱动单片机的通信方式采用 CAN 总线方式，增加了可扩展性以及稳定性；又详细的介绍了增量式 PID 采用经验法调试参数的方法，以及做出了上位机进行速度闭环的监控以及参数的整定。

5、云台双环 PID 控制，摩擦轮大小不断的改进等等

6、爬杆机器人设计包含软件、硬件、机械。

经历了本次课题后，学到了很多的东西，就比如机械上理论的设计图和工厂加工的实际加工件是存在差距的，这种差距虽然给组装底盘时候带来不便，但是经过后期打磨处理，加工件都能满足设计需求；再比如硬件电源设计上，最初选用线性电源做的，结果上电不久，芯片就发热保护，导致整个控制系统的崩溃，经过不断的学习改进，最后采用开关电源解决电路发热问题，效果非常理想。

虽然设计的机器人功能已经实现，但是目前设计的机器人还存在很多问题以及待改进的地方：

1、机械结构上需要更加牢固的设计，增加防撞保险杠，改进车骨架构设计使车身更加紧固稳定，联轴器以及其他已损坏的部件应该设计成钢材。

2、全方位移动平台的控制方法还需进一步研究，目前仅仅完成了最基本的一系列功能，并没有涉及到复杂运动曲线的路径规划研究。

3、增加一些人工智能元素，使控制系统更智能化，就比如自动避障，视觉提取路况等等。同时需要做一个实时操作系统，对整个车体以及其他传感器进行统一有序的管理，最大限度的利用了处理器的资源，提高了控制效率。

收获的不仅仅是技术不仅仅是知识，收获更多的真性情。

无论干什么自己一个人的力量往往是薄弱的，所以到哪里都要提到团队合作这一古老又不可或缺的话题。作为一个好的团队，讨论、意见上的分歧肯定是有的，而且必须是有的，但是这些意见分歧、队员彼此间的讨论只应该出现在会议室，大家走出会议室应该要有一个统一的意见。几个人的一支队伍不管在整个项目的哪个阶段都应该有统一的领导、步调一致，这样团队才会有更大的凝聚力和战斗力。

作为团队的一员我们应该相信队友，切记不要自己单干，如果每个人都单干，这样人员就不能得到充分的利用，会导致部分人忙的要命，而有些人却不知道干嘛。一个主力队员太过于自信不仅会给队友带来压力，在自己的问题不能解决的时候更会给自己带来压力，所以我们应该形成一个相互学习、交流，大家畅所欲言的环境。同样从对手的身上可以学到自己不具备的素养、品质。

最后我想说这绝对是一段一辈子都无法忘记的回忆，是一段丰富多彩的人生阅历，一笔无比宝贵的人生财富。我们每一位队员都付出了很多，牺牲了很多，同时我们也学到了很多，成长了很多。最后预祝 robomasters 这个机器人比赛能够越办越好，能带给大学生们带来更多乐趣，成为大学校园里一个热门的比赛，更希望能给大学生们带来更多成长的机会。

感谢我的学校，应有尽有的资助，让整个战队后枕无忧。

感谢我的老师，无微不至的关心，让每个队员动力十足；

感谢我的队友，日日夜夜的努力，让整个团队走到今天；

感谢组委会的帮助 给了每个队伍无尽的帮助！

感谢组委会的美女们 给了单身狗们奋斗的动力！

感谢我的小王老师 让我写了上一条！

祝全国总决赛顺利举办，祝东林—天空领域—Ares 战队的每个队员在决赛上广泛交友、开阔视野、玩的开心，！

参考文献

- [1] 张超. 机器人全方位移动平台的设计与控制研究[D]. 河海大学硕士论文,2007.
- [2] 罗均,谢少荣,霍宇毅,王琦等. 特种机器人[M]. 化学工业出版社,2006.
- [3] 王坤兴. 机器人发展趋势[J]. 机器人技术与应用,1999.12(6):26-31
- [4] 蔡自兴. 21世纪机器人技术的发展趋势[J]. 南京化工大学学报.2007(7):73-78.
- [5] 海丹. 全向移动平台的设计与控制[D]. 国防科技大学硕士学位论文,2006.
- [6] 李磊,叶涛,谭民,陈细军. 移动机器人技术研究现状与未来[J]. 机器人.2002.5(9):475-480.
- [7] 聂晓璐. 全方位小型足球机器人运动性能研究[D]. 国防科技大学硕士学位论文, 2006.
- [8] 王东梅. 未来战争将推动军用机器人研究[J]. 国防技术基础.2002.6(5):21-25.
- [9] 周永生. 美军机器人要上伊战场[J]. 环球军事.2005.2(6):35-40.
- [10] 欧青立,何克忠. 室外智能移动机器人的发展及其关键技术研究[J]. 机器人,2000.6:519-526.
- [11] 马瑞卿. 自举式 IR2110 集成驱动电路的特殊应用[J]. 电力电子技术,2000,1:31-33.
- [12] 韩成浩,高晓红. CAN 总线技术及其应用[J]. 制造业自动化,2010.2:146-149.
- [13] 江浩,樊炳辉等. 新型移动机器人的结构设计[J]. 应用科技.2000.7:3-5.
- [14] 赵冬斌,易建强,邓旭. 全方位移动机器人结构和运动分析[J]. 机器人.2003.5:394-399.
- [15] 刘中新,张春亮,王占富. 全方位移动平台运动分析与仿真[J].机械与电子.2013.8:310-315.
- [16] 肖永清. 现代汽车的多路总线传输与网络通讯系统[J]. 客车技术与研究.2009.4:45-49.
- [17] 王晓明. 电动机的单片机控制[M]. 北京航空航天大学出版社,2002.5.
- [18] 田春鹏. 增加制丝叶片处理线筒温自动控制功能[J]. 科学创新导报.2012.4:2-4.
- [19] 刘玲玲. PID 参数整定技术的眼睛及应用.[D]. 郑州大学硕士学位论文,2010.
- [20] 徐洪华. 模糊 PID 参数自整定技术的研究[D]. 长春理工大学硕士学位论文,2004.
- [21] Tom  Lozano-P rez,Michael A. Wesley. An algorithm for planning collision-free paths among polyhedral obstacles[J]. Communications of the ACM . 2010 (10)
- [22] TADAKUMA K,TADAKUMA R,NAGATANI K,et al.Crawler vehicle with circular cross-section unit to realize sideways motion. 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems . 2008
- [23]OlafDiegel,A.B.G.B.Improved Mecanum Wheel Design for Omni-directional Robots. 2002 Australasian Conference on Robotics and Automation . 2012