

**江苏科技大学**

**本 科 毕 业 设 计（论文）**

学 院 自动化学院

专 业 自动化

学生姓名 方贻培

班级学号 192210301210

指导教师 林焘

2023年 6月

江苏科技大学本科毕业论文

图像识别的导航智能车

（Image Recognition Navigation Intelligent Car）

学生姓名：方贻培

指导教师：林焘

江苏科技大学

二O二三年六月

目 录

[摘要 I](#_Toc135158102)

[ABSTRACT II](#_Toc135158103)

[第1章 绪论 4](#_Toc135158104)

[1.1 研究背景与意义 4](#_Toc135158105)

[1.2 国内外研究现状 5](#_Toc135158106)

[1.3 主要研究内容与章节安排 6](#_Toc135158107)

[1.3.1 研究内容 6](#_Toc135158108)

[1.3.2 章节安排 6](#_Toc135158109)

[第2章 系统总体方案设计 7](#_Toc135158110)

[2.1 功能需求 7](#_Toc135158111)

[2.2 系统总体方案设计思路 7](#_Toc135158112)

[2.3 硬件选型 9](#_Toc135158113)

[2.4 方案实施步骤 10](#_Toc135158114)

[2.5 小结 10](#_Toc135158115)

[第3章 室内定位研究与方案设计 10](#_Toc135158116)

[3.1 室内定位方案概述和选择 10](#_Toc135158117)

[3.2 视觉定位系统理论基础 12](#_Toc135158118)

[3.2.1 成像原理 12](#_Toc135158119)

[3.2.2 可视标签和图像采集设备选型 13](#_Toc135158120)

[3.2.3 AprilTag标签检测 13](#_Toc135158121)

[3.3 室内定位方案设计 14](#_Toc135158122)

[3.3.1 引导地图模型 14](#_Toc135158123)

[3.2.3 机器人循迹控制 14](#_Toc135158124)

[3.4 小结 14](#_Toc135158125)

[第4章 系统硬件电路设计 15](#_Toc135158126)

[4.1 系统总体硬件结构 15](#_Toc135158127)

[4.2 模块电路设计 15](#_Toc135158128)

[4.2.1 主控电路设计 15](#_Toc135158129)

[4.2.2 电源电路设计 16](#_Toc135158130)

[4.2.3 电压检测电路设计 17](#_Toc135158131)

[4.2.4 电机驱动电路设计 18](#_Toc135158132)

[4.2.5 MPU6050电路设计 19](#_Toc135158133)

[4.2.6 拓展高电流输出电路设计 20](#_Toc135158134)

[4.2.7 拓展IO电路设计 20](#_Toc135158135)

[4.2.8 串口电路设计 21](#_Toc135158136)

[4.2.9 OpenMV Cam M7和Esp32的接口 21](#_Toc135158137)

[4.2.10 超声波HC-SR04 22](#_Toc135158138)

[4.3 小结 23](#_Toc135158139)

[第5章 系统软件设计 23](#_Toc135158140)

[5.1 系统总体软件结构 23](#_Toc135158141)

[5.2 软件开发环境 23](#_Toc135158142)

[5.3 基于FreeRTOS嵌入式操作系统的底盘运动实现 24](#_Toc135158143)

[5.3.1 底盘运动任务设计 24](#_Toc135158144)

[5.3.2 调试串口任务设计 27](#_Toc135158145)

[5.3.3 通信串口任务设计 28](#_Toc135158146)

[5.3.4 陀螺仪任务设计 28](#_Toc135158147)

[5.3.5 电池电压任务设计 29](#_Toc135158148)

[5.3.6 状态灯任务设计 29](#_Toc135158149)

[5.3.7 其它调试任务设计 30](#_Toc135158150)

[5.4 基于OpenMV Cam的图像处理实现 30](#_Toc135158151)

[5.4.1 路线识别 30](#_Toc135158152)

[5.4.2 终点识别 31](#_Toc135158153)

[5.4.3 AprilTag标签识别 31](#_Toc135158154)

[5.5 基于OpenMV Cam的十字路口处理 32](#_Toc135158155)

[5.5.1 直行路口 32](#_Toc135158156)

[5.5.2 转弯路口 32](#_Toc135158157)

[5.6 基于OpenMV Cam的返程处理 33](#_Toc135158158)

[5.6.1 返程终点转弯 33](#_Toc135158159)

[5.6.2 返程路口 33](#_Toc135158160)

[5.7 基于Esp32的WebServer上位机 33](#_Toc135158161)

[5.8 基于串口的OpenMV Cam通信实现 34](#_Toc135158162)

[5.8.1 底盘和OpenMV Cam通信 34](#_Toc135158163)

[5.8.2 Esp32和OpenMV Cam通信 34](#_Toc135158164)

[5.9 小结 34](#_Toc135158165)

[第6章 系统测试和分析 34](#_Toc135158166)

[6.1 测试内容及环境 34](#_Toc135158167)

[6.2 识别路线测试 35](#_Toc135158168)

[6.2.1 直线识别 35](#_Toc135158169)

[6.2.2 十字路口识别 35](#_Toc135158170)

[6.3 识别AprilTag标签测试 36](#_Toc135158171)

[6.4 转弯和直行运动测试 36](#_Toc135158172)

[6.5 终点停车和掉头测试 37](#_Toc135158173)

[6.5.1 终点停车 37](#_Toc135158174)

[6.5.2 掉头 37](#_Toc135158175)

[6.6 返回遇转弯测试 38](#_Toc135158176)

[6.7 运行遇障碍物测试 38](#_Toc135158177)

[6.8 拓展WebServer传送目标点2测试 38](#_Toc135158178)

[6.9 运行总体性能测试 39](#_Toc135158179)

[6.10 小结 39](#_Toc135158180)

[第7章 总结与展望 39](#_Toc135158181)

[7.1 总结 39](#_Toc135158182)

[7.2 展望 40](#_Toc135158183)

[参考文献 40](#_Toc135158184)

[致谢 41](#_Toc135158185)

[附录 41](#_Toc135158186)

摘要

自从20世纪30年代，材料的提纯技术得到足够的改进，半导体产业开始被学术界认可。这里的半导体也称为集成电路。集成电路的组成是由成千上万，甚至上亿个晶体管组成，自从晶体管在贝尔实验室被研制出来，许多半导体公司的技术得到了空前的发展，特别是在处理器（CPU）领域。比如，英特尔公司研发的处理器从最初的4004处理器、i8086处理器到现在的酷睿系列处理器。从这些方面可以看出晶体管对整个集成电路的巨大推动作用。所以，在有了强大的硬件平台的基础之上，电子工作者可以借此来完成对算力要求更高的应用，比如导航、识别、SLAM等等。而本文也将借此设计出一款智能导航机器人。

在特定的场景下，比如医院里，较大传染性的病房需要无接触的收到药物，以方便病人的治疗，同时还要保障医务人员和家属的身心健康，此时，需要一个能按指定路线，从指定地点自动导航到病房的送药物的小车；又比如酒店里，客人需要进食或者需要更换服务物品，且需要尽量减少人员流动的情况下，只需在手机或个人电脑上点击需求，小车便能将物品送到门口，在充分保障酒店工作人员和客户双方身心健康的同时，也提供了极大的便利。生活中许多需要减少人员间的流动来保障人们安全和便利人们生活方式的场景。于是，本文设计开发出一款机器人，该机器人首要要求当然是安全、稳定、可靠，其次该机器人可以代替人类去做一些基本的服务工作，以减少人员间的流动，保障人们的安全。

结合目前图像识别技术的发展，有著名的AprilTag标签检测的识别算法。一方面，该算法在嵌入式设备上的识别准确率能达到90%以上，算法运算花费的时间也完全能搭配各种机器人的运动，实现准确定位和导航。另一方面，硬件OpenMV Cam也为图像识别提供了很方便和高性价比的嵌入式开发平台。同时，传统的STM32单片机结合FreeRTOS也能很方便的开发出稳定性较强的机器人底盘驱动程序，所以本文用OpenMV Cam作为系统图像识别导航的主控，且采用基于FreeRTOS的实时操作系统去搭建机器人底盘，同时开发和OpenMV Cam等上位机的通讯设计。同时，为了能在手机或电脑上指定导航目标点，采用了Esp32设计了款网页端的上位机。也利用了超声波测距模块进行距离的测量，能使智能车和物体保持安全的距离，使小车更加智能。

通过对样机进行功能测试后得到得实验数据表明，本文设计的图像识别导航机器人在硬件和软件各项性能指标上，完全能达到需求。在图像处理上也实现了较稳定的算法，同时系统因采用一个OpenMV Cam识别图像配合颜色路线识别算法，便实现了识别导航，具备显著的低成本的优势。还设计了网页端的智能交互，且进行了用户登录认证，保障了机器人运行的安全性和可靠性。

关键词：OpenMV Cam；智能导航机器人；AprilTAg；Esp32；FreeRTOS

ABSTRACT

Since the 1930s, the purification of materials has improved enough that the semiconductor industry has been recognized by academics. The semiconductors here are also known as integrated circuits. The integrated circuit is composed of thousands, even hundreds of millions of transistors, and since the transistor was developed in Bell LABS, many semiconductor companies have achieved unprecedented development of technology, especially in the processor (CPU) field. For example, Intel's development of processors from the original 4004 processor, i8086 processor to the current core series of processors, from these aspects can see the transistor to the entire integrated circuit of the great promotion. Therefore, on the basis of a powerful hardware platform, electronic workers can use this to complete higher computing power navigation, recognition, SLAM, etc. And this paper will also design an intelligent navigation robot.

In certain situations, such as hospitals, where a large infectious ward needs to receive medicines without contact to facilitate the treatment of patients and to protect the physical and mental health of medical staff and family members, a medicine delivery car that can automatically navigate to the ward from a designated location on a designated route is needed. For example, in the hotel, guests need to eat or need to change the service items, or very need to reduce the flow of personnel, just click on the demand on the mobile phone or personal computer, the car will be able to send items to the door, in the full protection of the hotel staff and customers both physical and mental health at the same time, but also greatly convenient both sides. There are also many scenarios that need to reduce communication between people while keeping people safe and making people's lives easier. Therefore, this paper designed and developed a robot, the first requirement of the robot is, of course, safety, stability and reliability, followed by the robot can replace people to do some basic service work, to reduce the flow of personnel, at the same time to protect the safety of people.

Combined with the current development of image recognition technology, AprilTag

tag detection recognition algorithm, on the one hand, the budget method on embedded equipment recognition accuracy can reach more than 90%, the time spent algorithm operation can fully match the movement of various robots, to achieve accurate positioning and navigation. On the other hand, the hardware OpenMV Cam also provides a very convenient and cost-effective embedded development platform for image recognition. And the traditional STM32 microcontroller combined with FreeRTOS can also be very convenient to develop a stable robot chassis driver, so this paper uses OpenMV Cam as the main control of image recognition and navigation system. The real-time operating system based on FreeRTOS is used to build the robot chassis and the communication design of the upper computer such as OpenMV Cam. At the same time, in order to be able to specify navigation target points on the phone or computer, Esp32 is used to design a web-based host computer. The ultrasonic ranging module is also used to measure the distance, which can keep a safe distance and make the car more intelligent.

The experimental data obtained after the functional test of the prototype show that the image recognition navigation robot designed in this paper can fully meet the requirements in terms of hardware and software performance indexes. At the same time, because the system uses an OpenMV Cam with color route, it can realize the recognition and navigation, which has the advantage of low cost. The intelligent interaction of web page is also designed, and the web page encryption is carried out to ensure the safety and reliability of the robot operation.

Keywords: OpenMV Cam; Intelligent navigation robot; AprilTAg; Esp32; FreeRTOS

第1章 绪论

* 1. 研究背景与意义

得益于半导体产业的蓬勃发展，集成电路芯片实现了跳跃性的进步和成就。在这背景下，各种高运算力的处理器接踵而至，给人们提供了更高性能的硬件平台，也提供了更多的解决问题的方向。

同时，研究机器人的自主导航能力是当今国内外学者的主要研究方向[1]。ROS为开发者提供了强大的通信方式和通用的开发环境，开发者可以在此框架下利用激光雷达或者相机进行自主导航的开发[2]。其中室内导航是自主导航的研究热点，是指移动机器人通过传感器感知周围环境并获取自身位姿，在有障碍物的室内环境中朝着目标进行自主运动[3]。为了应对复杂的导航情况，诞生了SLAM (Simultaneous Localization and Mapping)即同时定位与地图构建技术[4]。SLAM是为了在未知环境下进行进行运动来确定机器人的位置，同时构建出环境的地图模型，从而达到准确的自主导航[5]。SLAM主要含激光SLAM[6]和视觉SLAM[7]。

针对于智能车搬运方案，为了节约成本和缩短开发周期，有的方案以OpenMV作为核心，使用颜色识别以及特征点检测算法进行物块识别以及机械臂的配合，最终达到目标[8]。还有的方案在数字识别上采用模板匹配的算法[9]。路径寻找一般都是在场地下铺颜色线，利用多路循迹模块，如TCRT5000模块或其它灰度传感器进行循迹[10]。目前的传统方案，虽然已经尽可能的降低了实际的硬件成本，但是巡线用的灰度传感器完全可以使用OpenMV来实现，即图像识别和巡线可以用一个OpenMV传感器来实现。对于采用一个OpenMV实现物体识别导航的方案，有学者利用Arduino做机器人的底盘，设计出了使用OpenMv4来实现数字识别以及寻迹功能[11]。其中OpenMV数字识别采用NCC匹配算法，循迹采用基于霍夫变换（Huff）直线检测原理的get\_regression()方法[11,12]。此方案本人测试过，发现图像识别采用NCC匹配算法，识别正确率的高低受环境的影响较大，且会随着拍照角度变化。其次，因为采用霍夫变换去检测直线，进而进行巡线，需要二值化及图像腐蚀、图像膨胀、图像开闭运算等去对原始图像进行预处理，较为复杂，且很难处理巡线算法和NCC匹配算法的关系。但是一个OpenMV传感器同时实现图像识别和巡线的方案，给我提供了一个设计的方向。于是在此方向上，本文针对于一个OpenMV传感器同时实现图像识别和巡线设计出一款更加经济、更稳定和更可行的方案，同时自行设计开发出网页操控端，让机器人更智能。

这个方案能够解决在特定场景下，需要尽量减少人员接触，并能将物品送到指定目的，比如医院送药、酒店送餐等。这样避免了各种因人员接触带来的安全隐患。本文设计的方案相对其它方案更经济、更安全和更灵活，所以应用前景广阔。

* 1. 国内外研究现状

由于国外对移动机器人的研究起步较早、投资多、应用广泛，因此在移动机器人各种技术上都取得了比较成熟的科研成果[13]。国外一开始在机器人视觉、自主移动机器人导航和其他相关技术展开了系统研究，上世纪末室外自主移动机器人是在斯坦福研究所（SRI）的NilsNilssen和CharlesRosen等人开发出来，命名为Shakey[14]。之后美国设立了室外移动机器人的项目，进行快速推动研究，比如80年代和90年代美国能源部指定的RIPS-机器人和智能系统计划。这个阶段虽然是初级阶段，但是为后来的机器人发展积累了宝贵的理论和实验分析。在处理器的主要部件晶体管被研发出来后，移动机器人的发展也上了高速发展的道路。尤其是在自主导航方面，基于AGV自主导航的搬运机器人，最早出现在美国Ford Motor 公司推出的一款有轨引导的运输车[15]。2016年Clavien等人设计了一款偏心轮的搬运机器人并设计出一款时间缩放原理的控制器[16]。同年的Takahashi等人设计出一种全向轮搬运机器人[17]。Nuro公司推出可在城市上绝大多数地 面道路行驶的无人AGV被Nuro公司在2019年推出，并于2020年新冠疫情期间在加利福尼亚体育场用于运送食品和药品[18]。在SLAM技术方面，2016年 Hess 等人提出了Cartographer SLAM方法[19]，该方法采用图优化原理，将SLAM过程分割为“前端+后端”的方式，前端主要负责扫描匹配，后端主要负责回环检测和位姿优化[20]，完善了SLAM的流程基础框架。在搬运机器人路径规划方面，1968年Hart对Dijkstra算法进行改进，提出了 A\*算法[21]。后续相继推出了蚁群算法[22]、粒子群算法[23]等。

在国内，在对机器人的的研究起步较晚，基础设施也较差，总体研究水平也相对于国外较低[24]。尽管如此，国内还是有许多科研人员带着热爱去专研移动机器人的技术，而清华大学是最早展开机器人研究的科研机构，并于2000年引入激光雷达测距信息对周围环境进行建模，设计出一款自然导航的THMR-V型号的室外环境的AGV系统[25]。浙江大学则在基于拓扑地图的AGV智能路径规划技术研究上，用改进的Dijkstra路径搜索算法作为全局路径搜索算法,采用启发式搜索算法A\*算法作为局部路径搜索算法，有效的防止了冲突[26]。在SLAM技术方面，很多学者进行了SLAM的应用和改进。比如将图优化SLAM用于建筑场景地图构建，实现了机器人用于建筑环境测绘[24]；将激光SLAM用于服务机器人，在航站楼进行导航建模[27]；通过在激光约束中融入视觉语义信息与结构语义信息提高了激光SLAM的实时性与精度[28]；利用多传感器融合，将传感器导航方式集成为单独的节点，各种导航方式可以任意切换[29]。在路径规划方面改进和应用遗传算法[30]、蚁群算法[31]、粒子群算法[32]等。

同时近些年来，国内教育部为了促进机器人方面的研究，大力促进了各类机器人的比赛，比如全国大学生电子竞赛、RobotCup等。Frontier-ITM机器人就参加了RobotCup比赛，该机器人在比赛中展现了可靠的稳定性、开放性和极大的拓展性。中华牌月球车机器人，由湖南大学等多个高校研发，于2013年随着嫦娥三号登上月球[13]。还有近些年来许多机器人公司也大力进行机器人技术的研究，比如大疆无人机（DJI）的无人机的技术，已经达到了全球领先，拥有较强的稳定性和智能性。

* 1. 主要研究内容与章节安排

1.3.1 研究内容

本文主要是基于OpenMV Cam实现十字路口、Apriltag标签、颜色线的识别。论文的主要研究内容包括：

（1）机器人的移动底盘，采用FreeRTOS开发，并做好正逆解运动解析和写好与上位机通讯的协议，并写好串口调试的协议。

（2）OpenMV识别颜色线，实现循迹功能。

（3）利用对十字路口的感兴趣区域（ROI）进行划分，实现对十字路口各种路况的解析。

（4）利用OpenMV识别记录目标Apriltag标签，识别十字路口的Apriltag标签，进而判断十字路口是左转还是右转，亦或是直行。

（5）利用OpenMV识别矩形并统计数量，进行终点的判断。

（6）加入超声波HC-SR04模块进行距离的检测，小于一定距离会停车。

（7）拓展Esp32单片机，进行网页端控制。

1.3.2 章节安排

第一章，绪论。首先介绍课题研究背景与意义，在其中对比各个方案的优缺点，然后分析国内外研究的现状，最后介绍论文的研究内容和章节安排。

第二章，系统总体方案设计。首先写出课题的功能需求，然后设计好系统总体方案和各个功能模块，并进行硬件选型概述，最后介绍方案实施步骤。

第三章，室内定位研究与方案设计。首先介绍室内定位方案的选择，然后介绍视觉定位系统的理论基础，这其中包括最基本的成像原理、可视标签和嵌入式设备平台和AprilTag标签，最后再进行本章小结。

第四章，系统硬件电路设计。本章首先概述系统总体硬件结构，然后进行主控电路、电源电路、电压检测电路、电机驱动电路、MPU6050电路、拓展高电流输出电路和拓展IO电路等模块电路的设计。最后再进行本章的小结。

第五章，系统软件设计。本章首先概述系统总体软件的结构，然后介绍软件的开发环境，再介绍基于FreeRTOS嵌入式操作系统的底盘运动实现，这包括调试串口任务、通信串口任务、正逆运动解析的底盘移动任务、LED状态灯任务、电池电压检测任务等，然后介绍OpenMV Cam的图像处理实现，设计了路线识别、十字路口识别、终点识别、AprilTag标签识别。然后重点分析了十字路口时直行路口还是转弯路口。还分析了基于OpenMV Cam返程时对终点转弯掉头的处理，对转弯路口的处理和对直行路口的处理。其次开发基于Esp32的WebServer的网页端操作，即上位机。再然后详细介绍基于串口的OpenMV与底盘和Esp32的通信。最后进行本章的小结。

第六章，系统测试和分析。本章首先介绍测试的内容及环境，然后进行十字路口转向及直行测试，然后进行停车识别AprilTag标签测试其识别，再进行返程掉头测试和返程遇转弯测试，还要进行行进过程中突然停车的可靠性测试，综合上述功能，进行总体性能测试。最后进行本章的小结。

第7章，总结与展望。先总结本课题的设计及测试情况，再展望后期的改进方向。

1. 系统总体方案设计

2.1 功能需求

本文设计的图像识别的导航智能车要实现的功能有：

（1）机器人底盘的开发，底盘的前后左右和旋转。

（2）准确定位，给定的目标点，并按指定路径到达，且到达后能停车，还能返回原点。

（3）指定物品，识别给定的智能车的目标点。

（4）自动规避，距离小于一定值，会让车停下，当障碍物走后，再继续执行到目标点。

综合上述功能需求可得，要实现一辆能在一开始指定目的地，然后智能小车能识别目标点，最后自动导航到目标点的智能小车。

2.2 系统总体方案设计思路

本文的基于图像识别的导航智能车设计是属于嵌入式产品的开发设计，所以系统的总体方案设计的主要步骤被分为硬件模块设计和软件模块设计。通过对功能需求的分析，在硬件模块设计方面，机器人底盘的开发选择运动控制较稳定的STM32作为主控制器，在图像识别定位和识别物品上选择性价比较高的OpenMV Cam，自动规避则选择廉价的超声波HC-SR04模块。同时为了拓展网页端控制功能，且需要快速开发HC-SR04测距，选择了Esp32进行网页端上位机的开发和HC-SR04数据的读取。STM32底盘驱动开发板还拓展了陀螺仪MPU6050模块和引出了许多拓展IO接口，方便后续开发。其总体的硬件模块设计框架如图2.1所示。

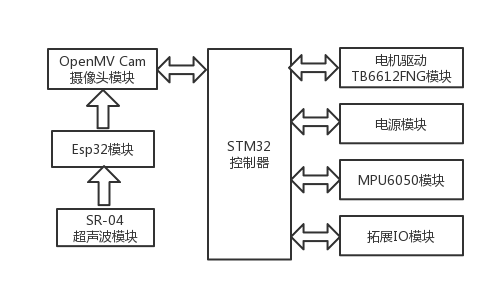


图2.1 总体的硬件设计思路

在软件模块设计方面，机器人底盘的开发是基于FreeRTOS嵌入式操作系统的设计，设计底盘运动任务、调试串口任务、通信串口任务、陀螺仪任务、电池电压任务、LED任务、KEY任务等。图像识别的定位一方面是识别路径，采用了OpenMV识别颜色线进行划分方块进行巡线，另一方面是识别AprilTag标签，采用硬件平台固件内置的AprilTag算法，从而最大限度的加快了识别的速度。识别指定的物体，也采用了识别AprilTag进行模拟。自动规避则采用Esp32按协议读取HC-SR04模块的数据并通过串口传给OpenMV Cam进行检测障碍物处理。拓展处Esp32的网页端上位机开发，指令也是通过串口和HC-SR04数据一起传送给OpenMV，并在嵌入式网页开发基础上加入了加密保障，让整体方案更加安全、更加可靠和更加智能。其总体的软件模块设计框架如下图所示。

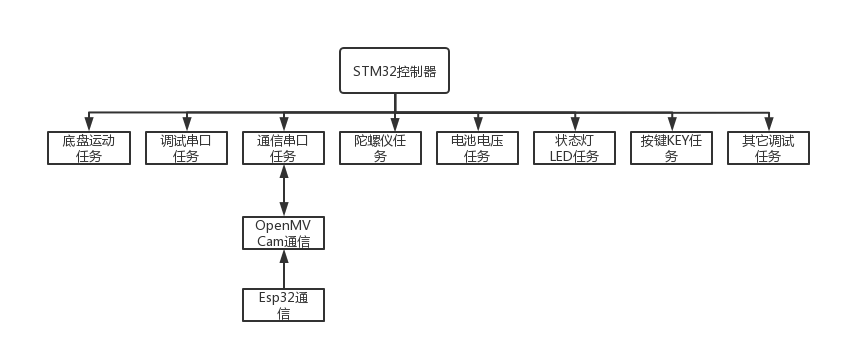


图2.2 STM32控制器的软件设计思路



图2.3 OpenMV Cam的软件设计思路

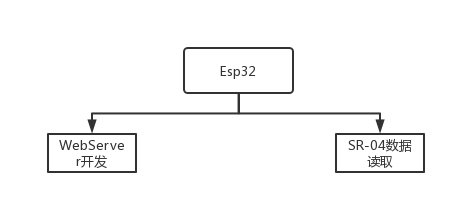


图2.4 Esp32网页端服务器开发思路

图2.2所示，是基于FreeRTOS开发的各个功能模块，同时显示了和OpenMV Cam和Esp32间的关系。图2.3所示，是基于MicroPython语言开发的OpenMV，主要是对颜色线和AprilTag标签的识别。图2.4所示，是基于Esp32下的FreeRTOS开发的网页端服务器，并且读取了超声波HC-SR04的测量距离，充分利用了其硬件资源。

2.3 硬件选型

对于硬件的选型，包括如下：

（1）机器人底盘选择STM32F103RCT6型号的单片机，该型号单片机在硬件配置上含有4个16位定时器、2个基本定时器、2个PWM定时器、3个USART、2个UART、3个ADC，硬件配置完全能满足开发需求。同时闪存为256K，属于大容量的ST产品。以STM32F103RCT6为主控，设计周围其它模块电路，调试串口选择CP2102芯片，陀螺仪选择MPU6050芯片，电机驱动选择TB6612FNG芯片，5V稳压选择MP2236芯片，3V稳压选择AMS1芯片。最后设计PCB并焊接测试，集成出一块底盘核心控制板。

（2）图像识别选择OpenMV Cam M7 ，配合OV7725摄像头，M7系列含有2个UART，可以利用这两个串口分别进行和底盘及ESP32的通信，同时图像处理含有内置算法，运行及开发均较有较好的表现，所以完全满足开发需求。OpenMV是一个便捷、低功耗、识别效率高的视觉模块[2]。

（3）网页端服务器开发选择Esp32，含有2个CPU，时钟调节频率范围80MHz到240MHz，支持RTOS，是一款通用型Wi-Fi+BT+BLE MCU模组，同时支持UART、ADC、GPIO等外设接口，可以满足需求。

2.4 方案实施步骤

按照系统总体方案和各个功能模块的设计，方案具体实步骤为：

（1）分析功能需求，主要是室内定位研究方案设计、系统硬件电路设计及硬件模块选型、系统软件设计、系统测试和分析这四个部分。

（2）室内定位研究方案设计，分析室内定位方案和理论基础，并进行可行性研究计设计可用的实际模型。

（3）系统硬件电路设计及硬件模块选型，分析硬件系统设计的总体方案和模块的选型及对应原理图电路的设计。

（4）系统软件设计，设计出可行的软件实施方案，并设计出对应的控制算法。

（5）系统测试和分析，包括系统总体的实际测试，并记录测试数据，然后由测试情况，分析本文设计系统的总体运行情况。

2.5 小结

本章依据实际功能需求，提出了后续设计开发的系统的总体设计方案，主要包括硬件和软件的设计，并做出了硬件的选型，最后做出了方案实施步骤，为接下来的工作明确了具体设计的总体路线。

第3章 室内定位研究与方案设计

3.1 室内定位方案概述和选择

室外定位依靠GPS、卫星等的定位方式，可以设计出定位系统，对环境的抗干扰性也较强。相对于室外定位的定位方式，室内定位如果也是采用卫星信号的话，会得到很差的效果，因为室内环境下，卫星的信号较差，且室内的要求精度更高。本节将简述集中室内定位的方案，并选择合适的定位方案进行本方案的设计。

目前室内的定位方案主要有：UWB（超宽带）脉冲信号、RFID定位、ZigBee室内定位、超声波定位、LED定位、WiFi定位、蓝牙定位、地磁定位和计算机视觉定位[33]等。

（1）UWB（超宽带）脉冲信号

UWB脉冲信号定位，在由多个传感器选用TDOA和AOA定位算法对标签方位进行解析，多经分辨才能更强，精度才能更高，定位精度可达厘米级，同时抗干扰能力强。但是UWB定位难以完成大规模的室内掩盖，且系统计算量比较大。

（2）RFID定位

RFID即超广波技术，利用射频信号的电磁传输特性在射频标签及射频阅读器之间实现数据通信。其优点有数据传输率高、安全性好，但是精度较低，抗干扰能力较弱，要求的计算量也较大，同时需要定位的区域部署大量的RFID设备，且要求定位人员手持RFID标签，很不方便，因此在室内定位的领域并没有得到广泛的应用。

（3）ZigBee室内定位

Zigbee技术基于IEEE 802.15.4协议，主要原理是经过若干个待定位的盲节点和一个已知方位的参考节点与网关之间形成组网，每个细小的盲节点之间相互和谐通讯以完成全部定位。主要用于低数据传输、低功耗和低成本的无线传感器领域，但是一般的用户的智能设备不支持IEEE 802.15.4协议，因此在室内定位的领域也没有得到广泛的应用。

（4）超声波定位

利用超声波进行定位，但是超声波在空气中的衰减较大，不适合大型场合，另外反射测距时受多径效应和非视距传达影响很大，要形成精度较高的超声波定位，成本较高。因此室内定位也没广泛应用，但是在较小的避障小车里，要求的精度不高，可以采用此方案。

（5）LED定位

LED定位，即可见光通信技术。此技术采用光源发出的高频闪烁信号来编码LED发射源的位置坐标信息，感光传感器将接收到的可见光信号解码并计算出移动设备的位置信息。其优点是对硬件部署要求不高，因为可以利用室内已经部署的LED灯光，且不受无线电波干扰。但是此技术要求LED灯光具备闪烁编码功能，并且会由于非视线通信问题造成定位精度急剧下降。

（6）WiFi定位

室内的Wi-Fi网络可以作为一般网络基础设施，也可以利用其空间传播的路径损耗效应（Pathloss）建立路径损耗模型，从而实现位置数据的解算。此方案可以利用室内已经部署好的WiFi设备，当然如果室内无部署，就需要额外去部署。但是因为无线电波在室内空间的传播会存在阴影效应（Shadowing）和多路传播效应（Multipath），所以其受干扰较大，另外定位的精度也较低。因此此方案在室内定位未被广泛应用。

（7）蓝牙定位

定位原理和WiFi定位原理类似，此方案的优点是蓝牙设备较低功耗，但是需要大量部署beacon设备，且定位的精度也较低。因此此方案在室内定位也未被广泛应用。

（8）地磁定位

此方案的原理是，把地球视为一个磁偶极，且地磁场在不同点的特征是变化的，所以可以利用这个特征来表征某个确定点的位置信息。利用地磁场进行定位是对目标区域的特征点进行测量，关联地磁特这部分与位置坐标，后期采用匹配算法对当前测量的地磁信号进行位置的推算，估计出当前位置信息。优点在于对硬件成本要求低，无需部署设备，一般情况下抗干扰能力较强。缺点在于因为是依靠地磁场，所以其会受地磁场的变化而干扰，且定位精度不够高和开发难度较大。所以此方案暂时没有得到广泛应用。

（9）计算机视觉定位

计算机视觉技术是靠感光元件获取当前环境的图像，然后进行图像的处理，再提取出图像的特征点进行位置的计算。根据感光元件的数量，可以分为单目视觉和双目视觉，甚至还有专业获取图像深度信息的RGBD深度相机。单目相机在硬件结构上较简单，视觉信息不需要进行复杂的数据融合，所以处理速度很快，能满足实时性要求高的场合；双目则因未直接获取图像的深度信息，要进行数据融合，进而可以得到深度信息。计算机视觉定位的优点是较容易获取外部位置信息，在可视范围内有较好的识别效果，且系统的鲁棒性较强。此方案的缺点是运算量大，算法复杂，硬件成本高，但是目前嵌入式硬件平台的发展，已经能提供很强大的运算平台，同时算法的开发也已经有很多成熟的方案，缺点已经不明显。所以近些年来的室内定位，广泛的采用了基于计算机视觉定位的方案。

综合上述的所有方案，本文设计的图像识别的导航机器人，要求能够进行图像识别，且进行精准的定位导航，所以选择了基于计算机视觉的定位方案。

3.2 视觉定位系统理论基础

3.2.1 成像原理

针孔相机模型，如图3.1所示。

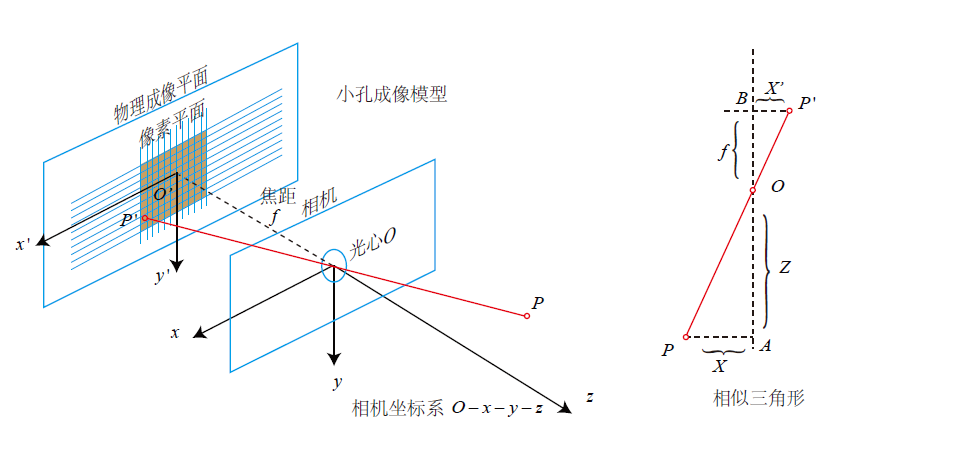


图3.1 针孔相机模型

设O-x-y-z为相机坐标系，O为摄像机的光心，现实世界的P点，经过小孔O的投影，落在物理成像平面上，成像点为。设P的坐标为，坐标为，且物理成像平面到小孔的距离为焦距f。于是，根据相似三角形可知，

 (3.1)

其中的负号表示成像是倒立的。于是考虑实际情况不可能是倒立的，为了更符合实际情况，等价的把符号去掉，即把成像平面对称的放到相机前方，如图3.2所示。

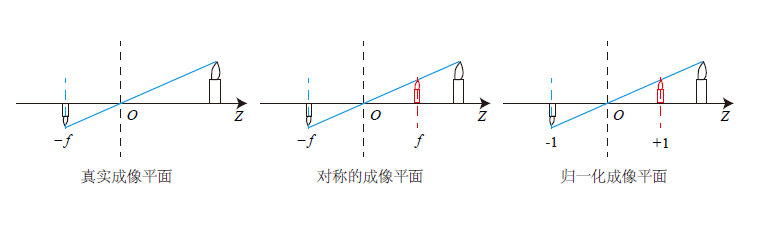


图3.2 真实成像平面，对称的成像平面，归一化的成像平面

于是可以把式3.1改写为

 （3.2）

再把，放到等式左边，整理可得

， （3.3）

式3.3即为空间点P再物理成像平面对应得点的坐标点。得益于嵌入式硬件平台的发展，大部分硬件已经帮我们把和成像原理相关的集成到硬件上。

3.2.2 可视标签和图像采集设备选型

目前常用的可视标签有二维码、条形码和AprilTag等，二维码可以存储百个字节的数据，但是要正确识别出数据，需要很高分辨率的摄像头，所以此限制对硬件提出了较高的要求，且识别二维码对环境要求较高。而AprilTag标签，具有保持存储有效数据的基础上，大幅度的降低了对摄像头的需求，也提高了实时性和识别正确率，降低了对环境的要求。即使在很低的分辨率的情况下，还能检测到AprilTag标签。综合考虑，本文在可视标签上，选择了AprilTag标签。

AprilTag标签根据组成维度和校验信息的不同分为许多个家族，即有TAG25M7、TAG25H9、TAG36H10和TAG36H11等家族。考虑TAG36H11家族，36表示改造家族维数为6，即属于6x6矩阵，可以存储36bit数据；11表示最小汉明距离，即任意两个存在变换关系的AprilTag标签之间信息编码差别的位数最少11位。汉明距离越大，表示两个AprilTag标签之间的差别越大，识别率越高。所以在考虑识别率的情况，本文设计采用了TAG36H11家族作为识别的AprilTag标签。

在图像采集设备上，选择OpenMV Cam M7系列的开源机器视觉模块。基于MicroPython开发OpenMV，有利于处理复杂的机器视觉算法。配合OV7725摄像头即可对TAG36H11家族标签的识别达到较高的帧率和较高的识别率。性能完全满足本文方案的需求。

3.2.3 AprilTag标签检测

AprilTag标签的检测过程，第一步式根据梯度检测出图像中的边缘，边缘检测算法可以是Canny算法等；第二步是找出图像中的四边形，即利用多边形凸包寻找等多边形检测算法，找出四边形；第三步是对图像进行二维码解码，利用第二步检测到的四边形内生成点阵列用来计算每块色块的值，再根据局部二值模式构造出简单分类器对色块进行分类，多色块组成1或0的编码。最后在得到编码后再与已知库内的编码进行匹配，即可得出解码的AprilTag标签。

3.3 室内定位方案设计

3.3.1 引导地图模型

本文设计的地图模型是基于大部分的酒店或者医院房间的布局，用颜色线代表寻的路径规划，交叉线代表十字路口，多个黑色矩形代表终点，在十字路口贴有TAG36H11家族的标签，用来模拟不同的房间号码。同时，采用初始识别的目标TAG36H11标签来模拟指定机器人应该到达的目的地，同时在终点处再给掉头TAG36H11 标签来模拟机器人已经到达指定地点，并且已完成本次导航的任务，可以掉头回原点。设计的引导地图模型如图3.3所示。

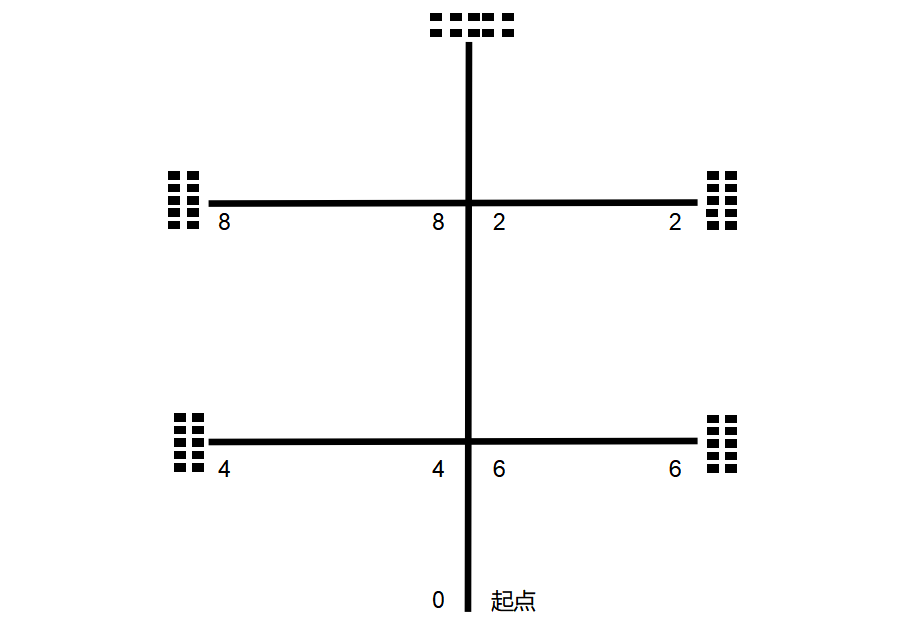


图3.3 引导地图模型

3.2.3 机器人循迹控制

在循迹控制方面，传统的方案一般采用灰度模块传感器，通过采集AD信号，判断车体的偏航误差，进而进行校正控制，达到巡线的效果。而本文的设计采用比较新颖的方法，即利用OpenMV对颜色线的识别，划分上中下左右的5个ROI区域，按5个ROI的各种状况进行巡线处理，同时还兼顾了十字路口的处理状况，是属于比较适合本课题设计的要求的巡线方法。

3.4 小结

本章节对室内定位的各种方案进行了概述和选择了适合本课题的方案，同时介绍了视觉定位系统的一些理论基础，包括小孔成像和可视标签。最后针对本课题，提出了合适的引导地图模型和机器人循迹控制方案。

第4章 系统硬件电路设计

4.1 系统总体硬件结构

分析总体的系统功能需求，设计出系统硬件的总体结构如图4.1所示。

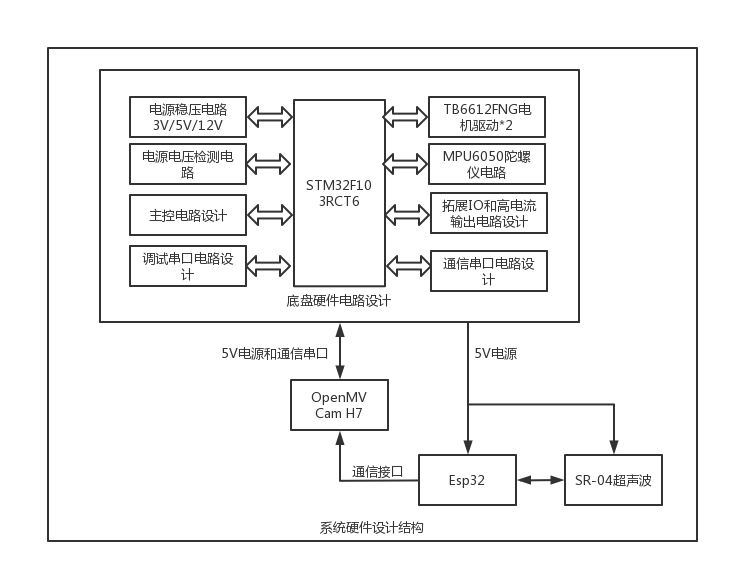


图4.1 系统总体硬件结构

对于机器人的底盘硬件电路设计含主控电路、电源稳压电路、电源电压检测电路、TB6612FNG电机驱动电路、MPU6050陀螺仪电路、拓展IO和高电流输出电路、串口通信电路设计。而OpenMV采用成熟的设计方案，直接选用官方的OpenMV Cam M7，后面会给出相对应得接口图。同样的Esp32和HC-SR04也采用模块开发，后面也是直接给出相对应的接口图。

4.2 模块电路设计

4.2.1 主控电路设计

主控电路是一个处理器芯片稳定运行的保障，本设计采用STM32F103RCT6芯片，并进行了主控电路的设计，如图4.2所示。

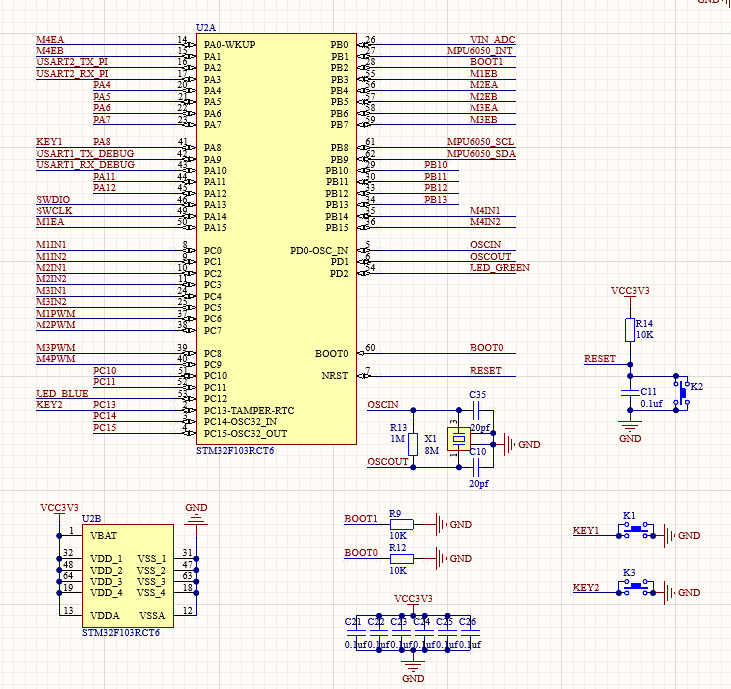


图4.2 主控电路

主控电路的核心是复位电路和时钟电路，其中复位电路是将处理器从运行状态复位到最开始上电时的状态，且复位引脚RESET为低电平有效。从上图的复位电路看出，RESET引脚一上电由于电容C11的充电，最初是低电平状态进行复位，当C11充满电，电阻R14将RESET上拉成高电平状态，使芯片进入运行状态。当按键K2按下RESET被拉低及进行复位。时钟电路时为处理器提供外部时钟信号，处理器在外部时钟的节拍下进行运作。选用8MHZ的时钟晶振与2个电容及1个1M电阻组成RC震荡电路作为系统的外部时钟源。同时为了让芯片工作更加稳定，还设计了6个电容对3.3V电压进行滤波和稳压。同时拓展了2个按键输入IO，方便进行主控电路调试。

4.2.2 电源电路设计

电机驱动需要12V的电压，同时需要为其他拓展电路提供5V电源和为主控等提供3.3V电源，所以需要12V降压到5V的电源电路，如图4.3，以及需要5V降压到3.3V的电源电路，如图4.4。电源设计电路如下图所示。

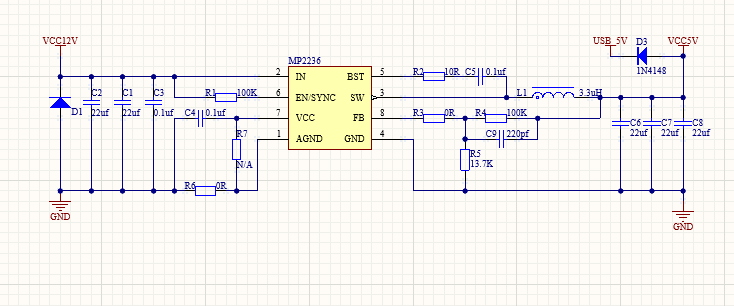


图4.3 12V转5V电路

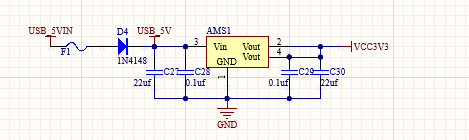


图4.4 5V转3.3V电路

图4.3采用了MP2236作为降压芯片，依据芯片的手册选择了降到5V的电路设计，且选用了二极管1N4148作为反向电压保护，使其作为保护USB口的电源电路组成的一部分。其中R6的设计，实现了模拟地和数字地分开，使电压更稳定，电路干扰较弱。电感L1的设计使电源的输出可以使输出的电流纹波更少，实现更稳定的电流输出。电容C6、C7、C8对输出的5V进行了稳压，使5V电压不至于突变。

图4.4选用AMS1的高电流和高电压型号，电路设计也是依据手册的推荐，同时二级管1N4148也是作为反向电压保护，组成了保护USB口电源电路的另一部分。电压从5V输入，实现了3.3V输出。

4.2.3 电压检测电路设计

电压检测电路是为了能够读取出电池电压值，在电压较低时可以停止智能车的运作，同时通过上位机显示电压值，有利于对电池电压剩余量的掌握。电路如图4.5所示。

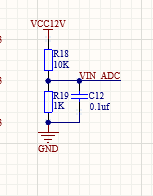


图4.5 电压检测电路

R11和R19的分压作用和下，忽略C12的充放电时间，设X为电池电压值，我们可以得到

 (4.1)

进而得出

 (4.2)

所以X的值即为电池电压值。

4.2.4 电机驱动电路设计

电机驱动是为了带动底盘的4个电机运动，而且要求是能够进行PWM调速和转向控制的，还能进行编码器脉冲的读取。电路如图4.6所示。

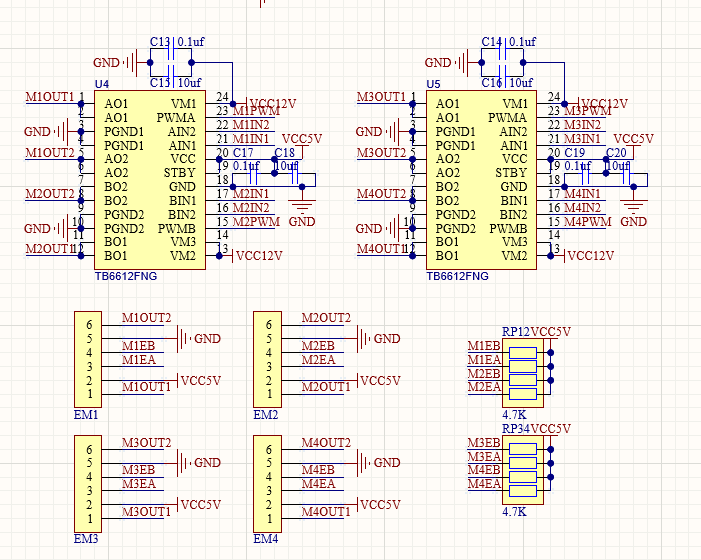


图4.6 电机驱动电路

电机驱动采用集成的TB6612FNG的电机驱动芯片，依据芯片的手册，对输入电源采用了电容进行滤波，同时为了方便，引出了4个集成的电机端接口。还对电机编码器的AB相脉冲输出进行了排阻上拉。对电机的控制的方法，比如M1PWM接口接入PWM信号进行调速，M1IN1和M1IN2接入GPIO引脚进行方向控制。

4.2.5 MPU6050电路设计

MPU6050陀螺仪能够获取机器人的姿态数据，依据姿态数据能够实现对车体的姿态的控制。电路如下图4.7所示。

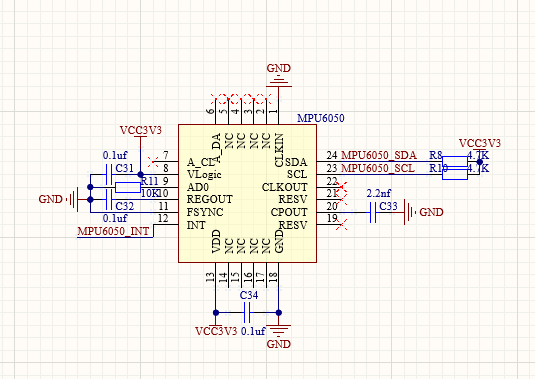


图4.7 MPU6050陀螺仪

电路的设计是依据了手册的典型电路，采用了IIC接口操作MPU6050的寄存器，同时还引出了中断接口，方便以一定的频率去读取出原始陀螺仪数据，然后进行数据融合等算法得出姿态角，再利用姿态角进行控制。

4.2.6 拓展高电流输出电路设计

为了能够提供给OpenMV Cam等需要较高电流的输入的设备，设计了高电流输出的电路，如上图4.3的电源电路的5V输出。该电路可以为工作时需要高电流的设备提供稳定的电压和电流。

4.2.7 拓展IO电路设计

把芯片没有用到的IO口引到外部排针，方便拓展其它功能，如下图4.8所示。

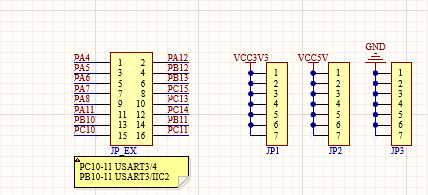


图4.8 拓展IO接口

有了拓展的IO接口，当OpenMV的巡线稳定性不高时，可以很方便的配合灰度循迹传感器实现智能车的循迹等开发。

4.2.8 串口电路设计

为了方便的调试底盘运动，设计了Type C接口的调试串口，如图4.9。同时，为了实现底盘系统和OpenMV系统的通信，引出了另一个通信串口，如图4.10所示。

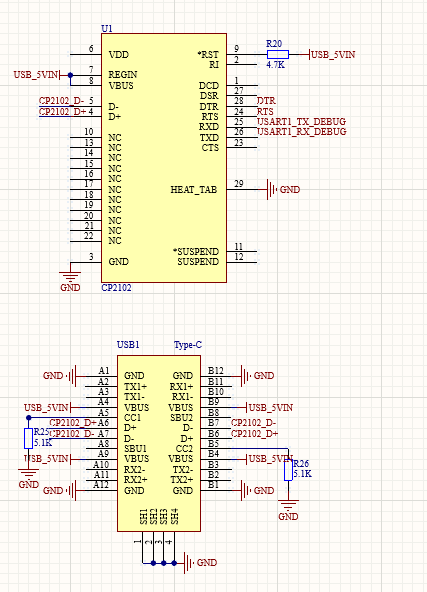


图4.9 Type C调试串口

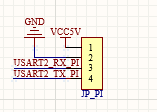


图4.10 通信串口

图4.9的调试串口电路选择了CP2102芯片，同时选择了16脚的Type C接口，按芯片典型电路设计了调试串口，极大的方便了后期的调试等需求。同时图4.10的通信串口，使其很方便的和OpenMV等上位机进行通信。

4.2.9 OpenMV Cam M7和Esp32的接口

图4.11为OpenMV Cam M7的接口图，图4.12为Esp32的接口图，如下所示。

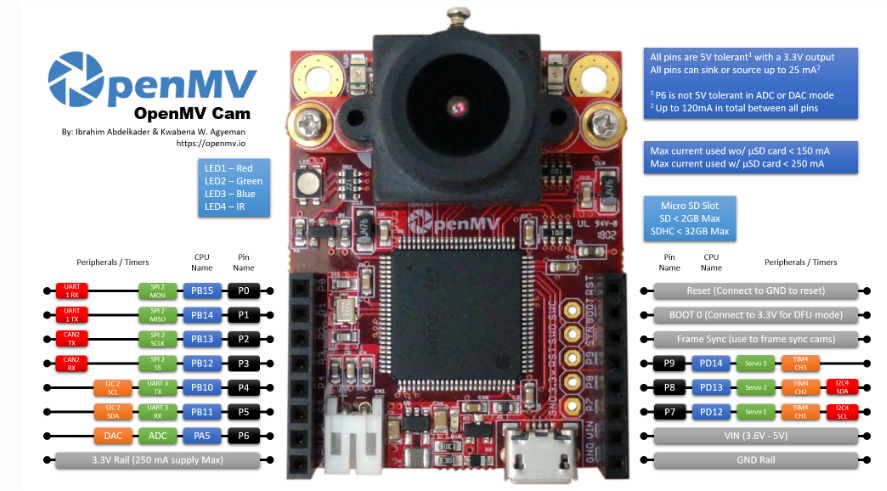


图4.11 OpenMV Cam M7接口

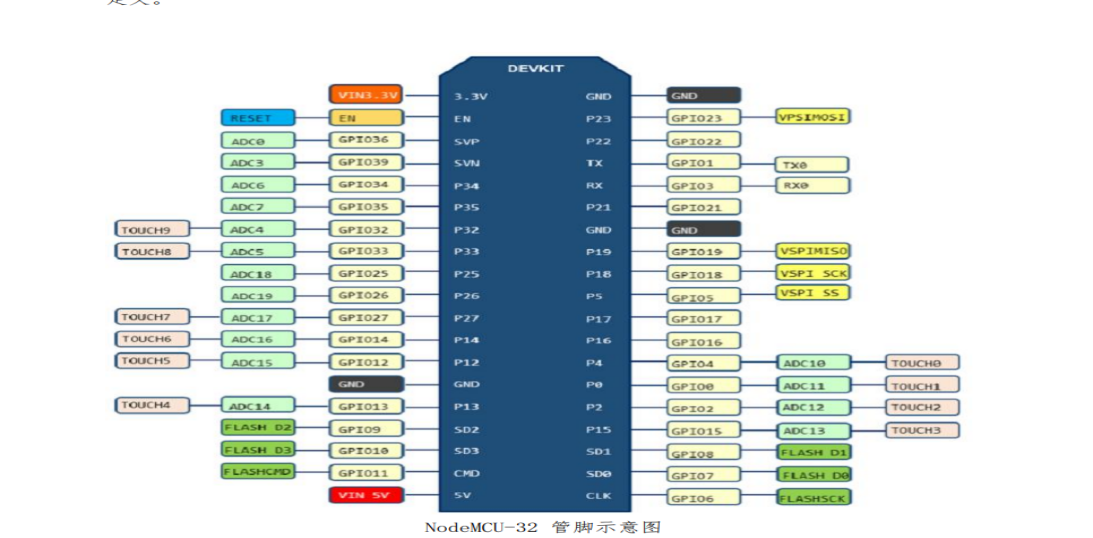


图4.12 Esp32接口

主要用到 OpenMV Cam M7的电源输入口和P0、P1、P4、P5的2个串口，Esp32的TX串口引脚和P5、P18的GPIO读取HC-SR04数据。

4.2.10 超声波HC-SR04

实物图如图4.13所示。



图4.13 HC-SR04实物

通过给Trig脚一个大于10us的高电平，循环发出脉冲信号，脉冲信号碰到障碍物会返回，返回的脉冲信号会使Echo脚变为高电平，读取Echo高电平时间即可测量距离L。计算公式为

 (4.3)

T为脉宽周期，C为声速。

4.3 小结

本章详细的阐述了硬件的设计，介绍了各个模块的电路设计，其中包括主控电路、电源电路、电压检测电路、电机驱动电路、MPU6050陀螺仪电路、拓展了高电流输出电源接口、拓展IO接口、串口电路、OpenMV Cam M7和Esp32的接口，介绍了超声波HC-SR04的硬件原理。

第5章 系统软件设计

5.1 系统总体软件结构

系统总体软件结构如下图5.1所示。



图5.1 系统总体软件结构

对于软件的设计，先介绍基于FreeRTOS的底盘结构，再对OpenMV Cam的图像处理进行功能模块的介绍，特别要注意十字路口的处理和返程处理手段，最后设计WebServer上位机，并利用通信模块设计，将整个系统联系起来。

5.2 软件开发环境

底盘运动系统是在FreeRTOS框架上运行，软件的开发环境有基于自己移植FreeRTOS的标准库，有基于STM32CubeMX的FreeRTOS框架，因为后一种基于HAL库，开发效率比较高，所以本课题选择后一种软件开发环境。同时，底盘是需要下载烧录到单片机里的，这里选择ST官方的Keil5进行烧录和调试。OpenMV开发是基于MicroPython的，选择官方的OpenMV IDE进行开发。Esp32开发WebServer是在Arduino IDE用C语言进行开发。

5.3 基于FreeRTOS嵌入式操作系统的底盘运动实现

5.3.1 底盘运动任务设计

电机的运动方面，520的编码器电机为11线，1：30减速比，四倍频，所以电机编码器的分辨率P为11\*30\*4=1320，即转一圈编码器产生的脉冲数为1320；TB6612FNG的电机驱动器，能够进行PWM调速和方向控制，配合上一章的电机驱动图4.6可得表5.1。

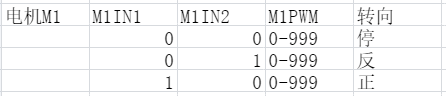


表5.1 电机M1的换相表

表5.1是以电机M1为例，其它电机也按照这个表进行控制。其中M1PWM的范围0-999，这样便可实现电机的调速。对于拥有4个编码器电机的底盘驱动，还需要对底盘进行运动解析才可使用。

对底盘4个电机进行正向运动学解析，即由轮子编码值得出底盘三轴里程计值。

首先机器人底盘属于四轮差速运动模型（SSMR）。先对左侧两轮进行速度分析，建系如图5.2所示。

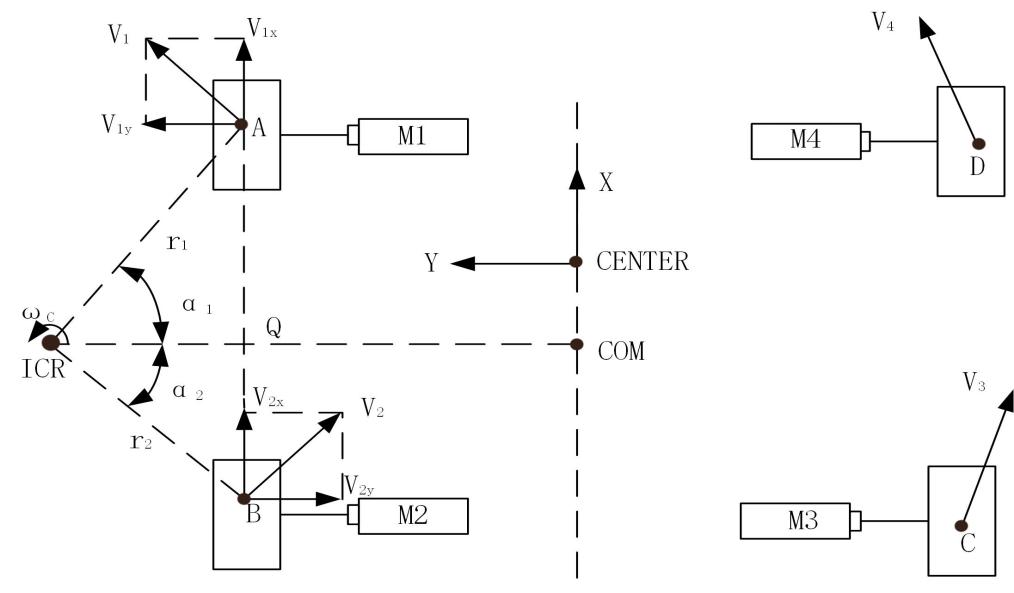


图5.2 SSMR运动模型1

以几何中心为原点建系X-CENTER-Y，以质心COM为原点建系X-COM-Y，COM和CENTER距离为dcc。机器人前向运动方向为X正方向，Y为向左为正方向，Z为垂直直面向外为正。A、B、C、D为轮子和地面的接触点，Q为AB和ICR-COM的交点。对图5.2的两个左侧轮分析其速度可得

 （5.1）

和表示两个左侧A和B的线速度，分别对应图5.2中的V1和V2，和分别对应其转弯半径。利用△A-Q-ICR和△B-Q-ICR的余弦关系可得

 （5.2）

α1和α2分别表示线段A-ICR、B-ICR与Q-ICR的夹角。化简上式可得

 （5.3）

其中、分别为和在X方向上的速度，即纵向速度。所以A和B的纵向速度是相等的，同理可以分析出C、D的纵向速度也相等。接下来分析A、D存在的速度关系。首先P点为直线ICR-Q和DC的交点，可以得到图5.3。

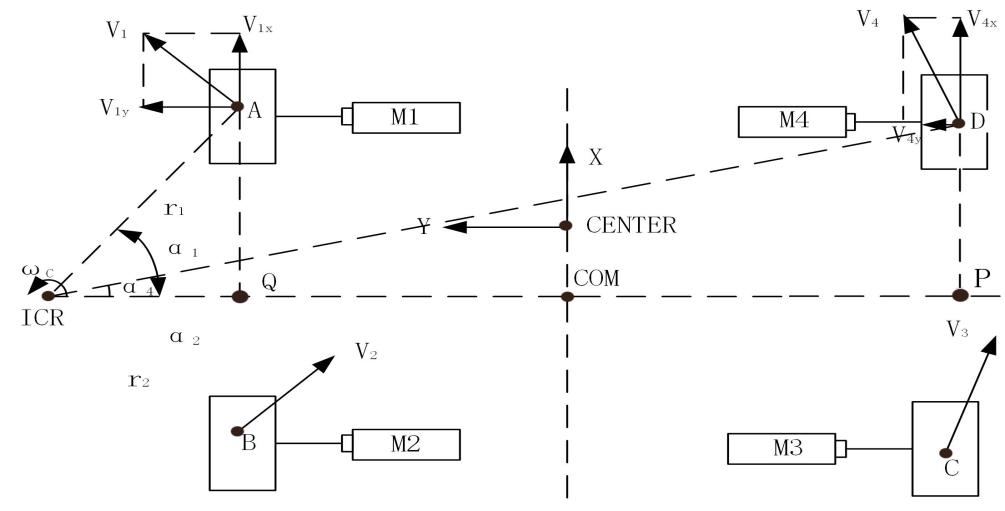


图5.3 SSMR运动模型2

采用和两侧轮一样的分析思路，对图5.3的两个前轮分析可得

 （5.4）

其中表示D的线速度，对应图5.3的V4，则对应其转弯半径。由△A-Q-ICR和△D-P-ICR的正弦定理可得

 （5.5）

式中，α1和α4分别表示A-ICR、D-ICR与P-ICR的夹角。进一步化简可得

 （5.6）

其中、表示和的横向速度。所以A和D的横向速度相等。同理B和C的横向速度也相等。综合式5.3和式5.6可得

，，， （5.7）

和分别表示左侧和右侧轮子的纵向速度，和分别表示前端和后端轮子的横向速度。最后化简SSMR的模型为两轮差速驱动机器人的运动模型，可得图5.4。

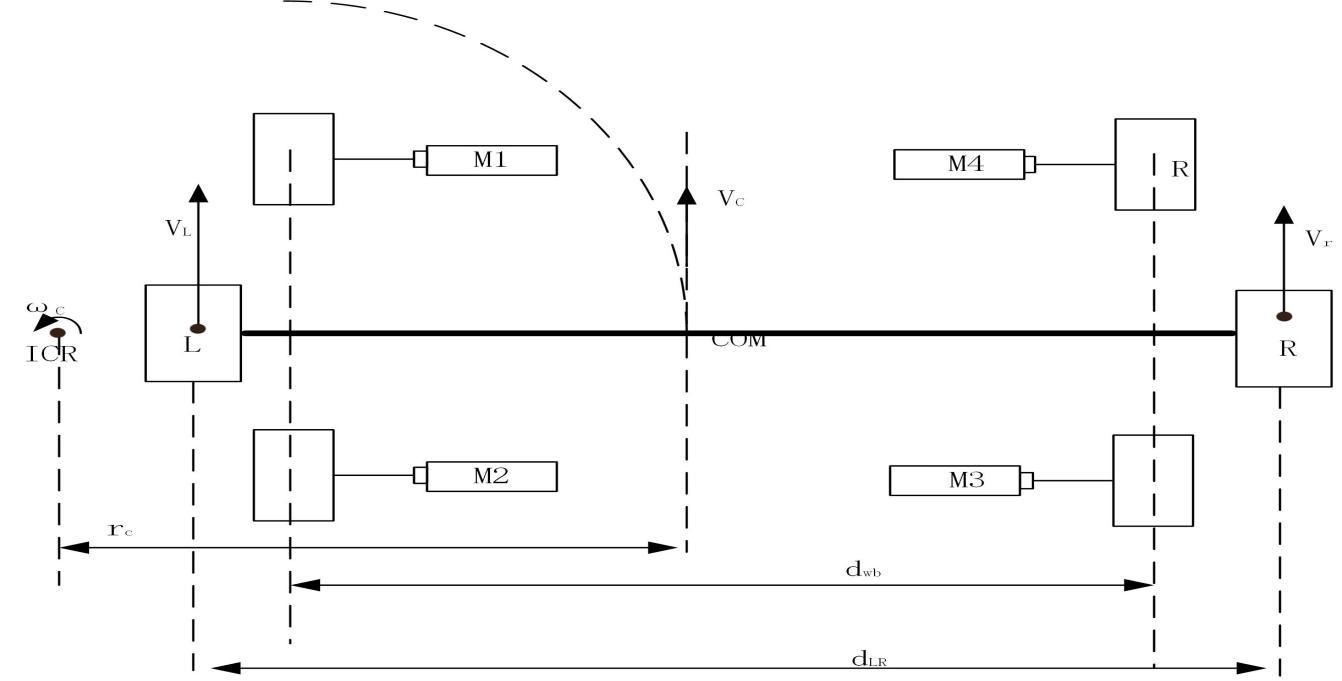


图5.4 SSMR简化模型

表示虚拟轮间距，表示底盘左右轮间距。所以简化的正运动模型式基于两轮驱动模型的，利用两轮模型绕点ICR的圆周运动，很容易得

 （5.8）

和可由式5.7得到，可以按两轮差速模型公式

 （5.9）

注意的是，按式5.9计算得的只是粗略的计算方式，准确的测量是要引入无量纲参数γ，但是此处采用式5.9计算是为了简化分析。同时，逆运动学模型为从底盘三轴里程计值得出轮子编码值，简化的模型为

 （5.10）

综合式5.8和式5.10即为四轮驱动移动机器人的运动模型（SSRM）分析的结果。同时注意横向速度Vy是由和的速度差产生的。

最后可以设计出底盘运动的程序流程图如图5.5所示。



图5.5 底盘运动任务流程图

正运动学运算依据底盘电机的编码分辨率P和轮子的直径D（m），得

（5.11）

式5.11的M为每米脉冲数。最后通过计算每个轮子的实际脉冲数乘上M，即可得到每个轮子的里程，且因为是在单位时间内的运算，里程数值完全可以代替车轮的速度。于是带入上式5.8，可以计算底盘三轴里程计数据X、Y、YAW的偏移量。反之，当直接给定三轴里程计X，Y，YAW数值，可以按式5.10和5.11计算出每个车轮编码器变化的目标值，最后利用读取的实际编码器编码值进行PID控制。PID控制器的设计选择了位置式PID，公式如下

（5.12）

其中等于编码器变化的目标值和实际值的误差，为上一次误差,、、分别为比例系数、积分系数、微分系数。为每个电机的最终控制量，于是可以实现对四个底盘电机的稳定控制。

5.3.2 调试串口任务设计

调试串口任务的帧格式如下表5.2所示。

JS6V8_Q2`0@B`F%QAOJ)11Y

表5.2 调试串口帧格式

data1和data2等属于传输的数据内容。因为调试的指令是字符串，可以直接利用帧尾的两字节及进行判断该帧是否接收完毕，从而完成调试的指令。

5.3.3 通信串口任务设计

通信串口任务的帧格式如下表5.3所示。

T3Q5Q80AUJTX_P(OIHOA0K4

表5.3 通信串口帧格式

其中Lenth为该帧的长度，Code为传输的功能性消息码，比如10为底盘的三轴数据。data1\_h为传输的第一个数据的高8位，data1\_l则为低8位。CRC-8为循环冗余校验码校验，本课题采用的多项式是x8+x4+x3+x2+1，序列为100011101，同时初始值选择0xFF，且运算进行的是循环左移，所以序列变为了0x1d。CRC8检验本质是模2除法，即异或运算。最后再加上帧头0x55和0xAA，帧尾0xOD和0x0A,进行帧头和帧尾校验。在此帧格式下，数据的传输很稳定，为和上位机OpenMV Cam的数据传输提供了稳定的通信条件。CRC8的算法步骤如图5.6所示。

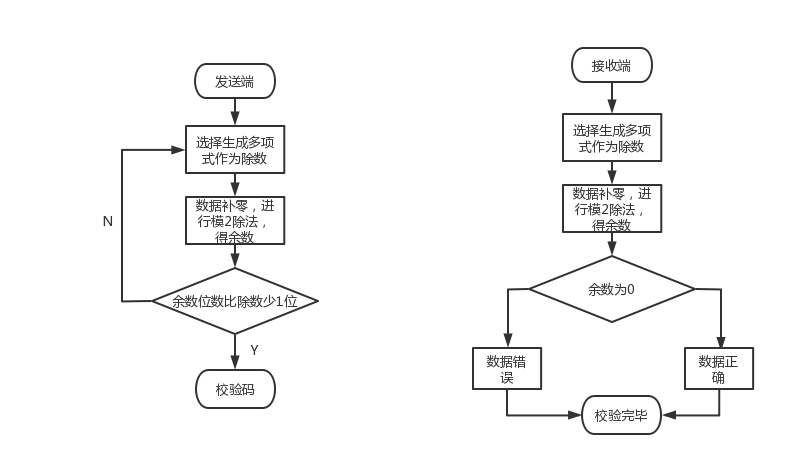


图5.6 CRC8校验步骤

图5.6的左图是发送端对一个字节的数据进行CRC的结果，对每个字节进行同样的处理将得到CRC码，我们将获得的CRC码补到数据末尾即获得最终的数据。右图是接收端的处理，同样进行循环的模2除法，最后判断余数，如果为0则是正确的数据，否则数据错误。

5.3.4 陀螺仪任务设计

陀螺仪MPU6050数据的读取，选择了DMP库进行处理，算法的步骤如图5.7所示。

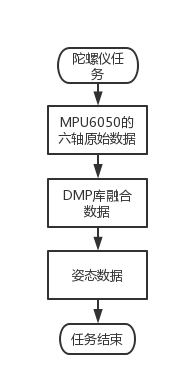


图5.7 陀螺仪处理流程

首先读取出原始的三轴加速度和三轴角速度数据，然后利用DMP库对读取的原始数据及进行融合，得出姿态角，即偏航角、俯仰角和横滚角。但是因为Z轴方向的偏航角没有磁力计进行滤波处理，所以该姿态角是不断进行漂移的，可以加入磁力计进行地磁场的数据融合，以此来校正偏航角。

5.3.5 电池电压任务设计

电池电压是采用ADC进行采样，步骤如图5.8所示。



图5.8 电压读取

图5.8的K值，是模拟量和数字量转换的比例系数。为了求出K，先利用芯片内部参照电压1.2V进行校准，得出内部校准K1，再者因为硬件的设计里的式4.2，VIN\_ADC=数字量\*K1，为了方便表达，选择了K=K1\*(11K/1K)，所以式4.2可以转化为

 （5.13）

5.3.6 状态灯任务设计

状态灯主要是为了显示电池的状况，板子上设计了2个灯显示。当电池需要充电时，1号灯亮，2号灯闪；当电池低于最低阈值时，1号灯灭，2号灯常量。具体设计如图5.9所示。

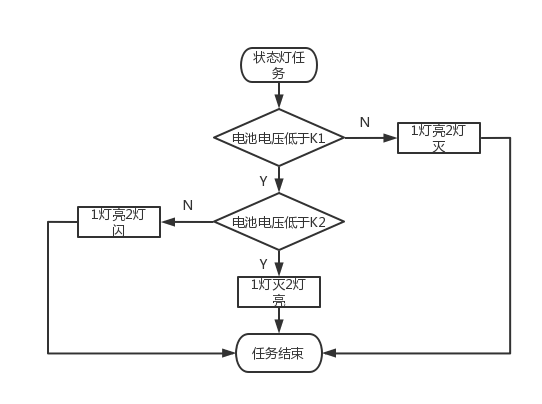


图5.9 状态灯任务

5.3.7 其它调试任务设计

本任务的设计是为了调试程序时打印用的，利用了重定向打印函数来实现。

5.4 基于OpenMV Cam的图像处理实现

5.4.1 路线识别

对于直线的识别，程序流程图如图5.10所示。

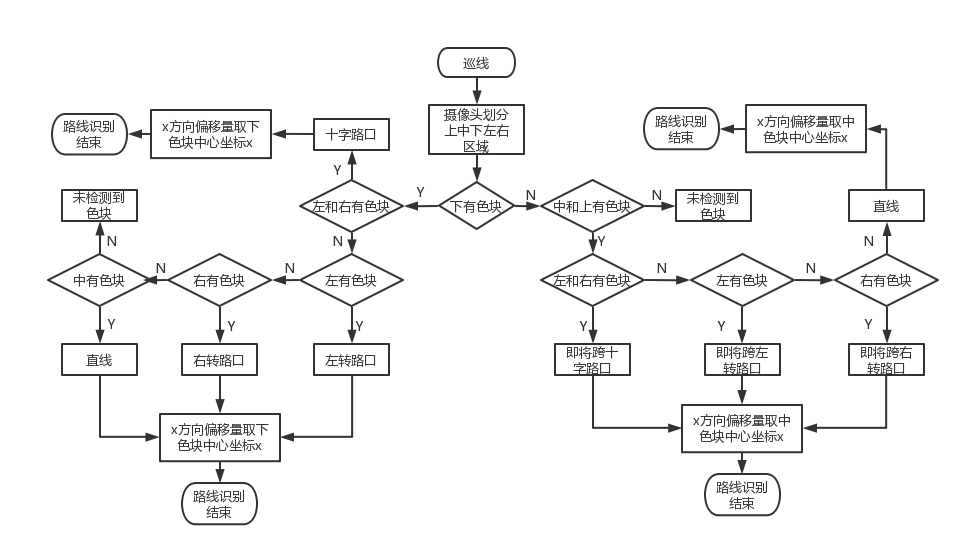


图5.10 路线识别流程

图5.10是路线的识别整体流程，要注意的是，为了防止太靠近左侧或者右侧出现误判，还需要在其中加入特例的判断，如图5.11所示。

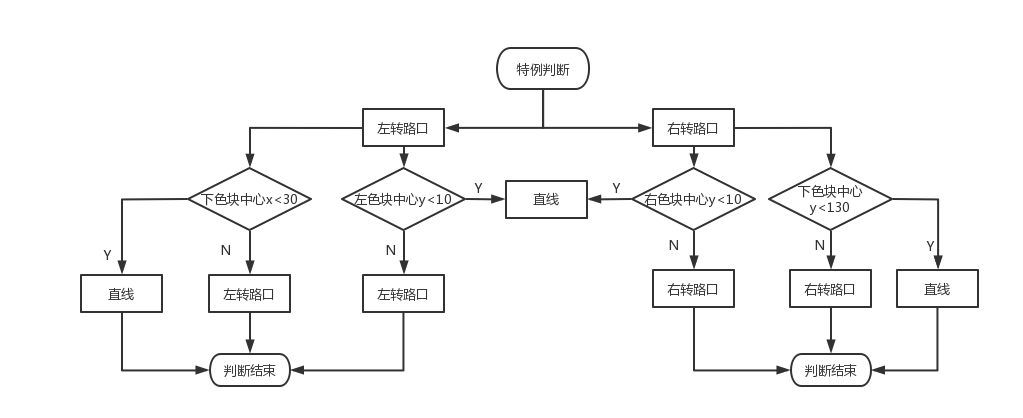


图5.11 路线识别的特例判断

5.4.2 终点识别

终点的识别是通过识别黑色矩形的大小，同时进行矩形数量的计数，当达到一定数量的时候，便识别到了终点线。设计流程图如图5.12所示。



图5.11 终点识别

图5.11的k是可调参数，按照实际的矩形的大小，调整k的值，使识别矩形更准确。

5.4.3 AprilTag标签识别

AprilTag标签的识别分为目标AprilTag识别、十字路口AprilTag识别和掉头AprilTag识别。具体的流程如图5.12所示。

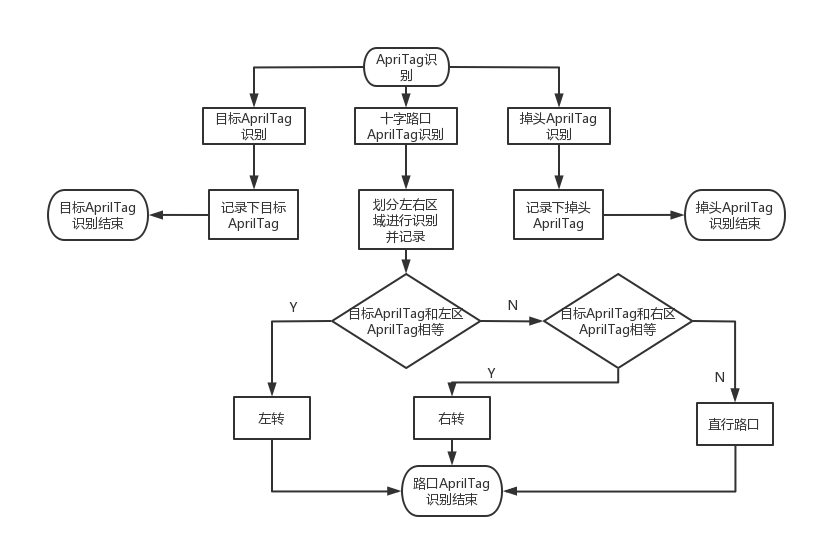


图5.12 AprilTag标签识别

由图5.12可知，识别出的AprilTag的各种情况，可以得到转弯的判断，这给直行路口和转弯路口的处理提供了依据。

5.5 基于OpenMV Cam的十字路口处理

5.5.1 直行路口

由图5.12可以看出，在十字路口的AprilTag识别，我们可以看到当路口没有匹配目标AprilTag标签时，此处的十字路口即为直行路口。图5.13为直行路口处理。

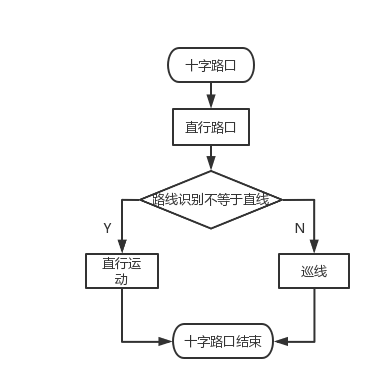


图5.13 直行路口处理

5.5.2 转弯路口

由图5.12可以看出，在十字路口的AprilTag识别，当目标AprilTag和左区域识别到的AprilTag相等时，此十字路口要左转；反之如果和右区域识别的AprilTag相等，则进行右转。图5.14为左转路口处理。

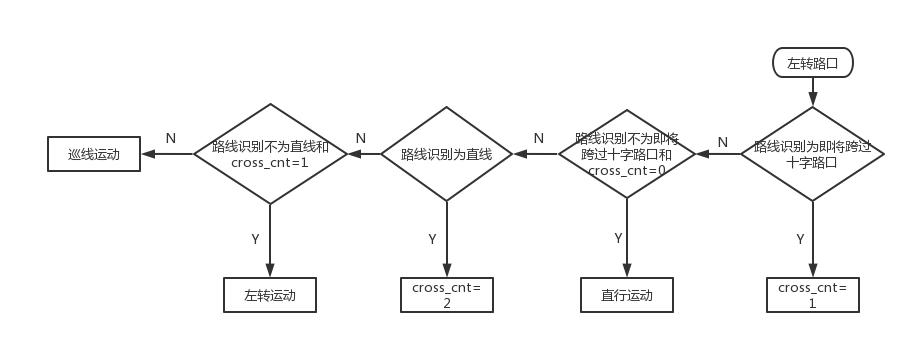


图5.14 左转路口处理

图5.14中的cross\_cnt属于建立的每个环节的互锁标志变量。对于右转路口同理。

5.6 基于OpenMV Cam的返程处理

5.6.1 返程终点转弯

由图5.12可知，在识别到达掉头AprilTag后进行掉头，掉头时不断判断摄像头是否重新识别到直线，识别直线的依据是图5.10所示。在图5.10中，当下有色块时，再判断中间有色块则为识别到直线，或者下无色块，中和上有色块，但是左右无色块，此时识别到的也是直线。

5.6.2 返程路口

返程的第一个转弯路口，按照掉头AprilTag的记录值进行转弯路口处理，而后的每一个转弯路口都当成直行路口处理。具体处理方式参考图5.13和图5.14。

5.7 基于Esp32的WebServer上位机

WebServer上位机是属于拓展的功能，编写流程如图5.15所示。

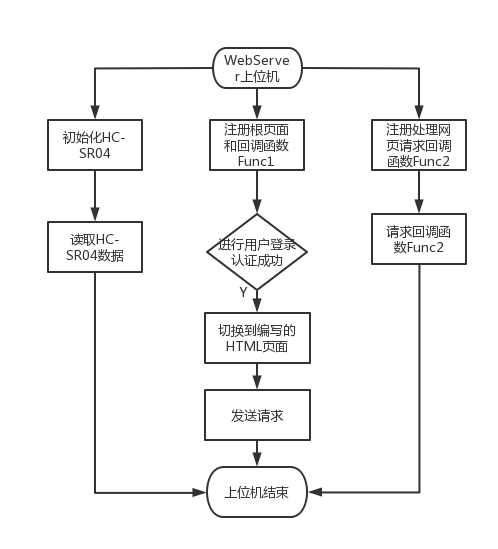


图5.15 WebServer上位机

图5.15的回调函数Func1是根页面最初的用户登录认证页面，只有认证成功才会切换到编写的第二个控制页面，且在控制页面可以输入控制值并发送请求，在程序中循环进行请求的处理，即由请求处理函数Func2来完成请求的处理。同时循环读取超声波HC-SR04的数据。

5.8 基于串口的OpenMV Cam通信实现

5.8.1 底盘和OpenMV Cam通信

因为要把转弯、巡线、直行运动和停车时对底盘运动的速度要求和运动解析时的运动解析，向底盘发送三轴里程计值的值，进行底盘运动的控制，所以要进行底盘的STM32和OpenMV Cam的通信。OpenMV按照表5.3的帧格式组合数据，并进行发送，校验码也是属于CRC8的校验。

5.8.2 Esp32和OpenMV Cam通信

Esp32读取的超声波要控制机器人遇到障碍物停止，同时发送目标的位置点，所以需要把控制指令传送给OpenMV进行处理。发送的数据格式如表5.4所示。

$7)H_5I%]RP@_`%_TP0T[E5

表5.4 Eep32帧格式

tag\_num是要发送的目标位置点，dist是超声波测量的距离。OpenMV依据帧头的0X55和0X53进行接收数据。

5.9 小结

本章是关于系统软件的具体设计，介绍了软件的开发环境，基于FreeRTOS的底盘运动分析及实现流程。OpenMV图像识别的处理，主要分析了路线、十字路口、终点和AprilTag标签的识别处理，还详细介绍了返程十字路口的左转、右转、直行处理，以及左转运动、右转运动、直行运动的实现，还介绍了返程时十字路口路口的处理。详细介绍了拓展的基于Esp32的WebServer的上位机功能。最后介绍了底盘和OpenMV Cam的通信，Esp32和OpenMV Cam的通信。本章是课题设计中最重要的一章。

第6章 系统测试和分析

6.1 测试内容及环境

测试的环境如附录图1所示。机器人实物含底盘、OpenMV Cam、Esp32和HC-SR04如附录图2所示。

测试内容有：

（1）智能车能实现识别直线；

（2）识别出十字路口、AprilTag识别；

（3）十字路口处能进行左转运动、右转运动、直行运动；

（4）在终点处能精准停下和进行掉头；

（5） 返程遇到第一个十字路口能判断转向，之后的路口能直接当直行路口；

（6）图6.1所示，能从原点0依次到达4、6、2、8，并返回；

（7）在运行途中，遇到障碍物会停下，障碍物去掉会继续进行任务；

（8）拓展WebServer上位机发送目标点2；

6.2 识别路线测试

6.2.1 直线识别

直线的识别如图6.1所示。



图6.1 直线识别

通过下和中均识别到色块，可以识别出直线。这和路径识别的图5.10分析符合。

6.2.2 十字路口识别

对于十字路口识别的测试图如图6.2所示。



图6.2 十字路口识别

通过图6.5看出，下、左、右均有色块，则可以识别出十字路口，这符合路径识别的图5.10的分析。

6.3 识别AprilTag标签测试

识别AprilTag标签如图6.3所示。

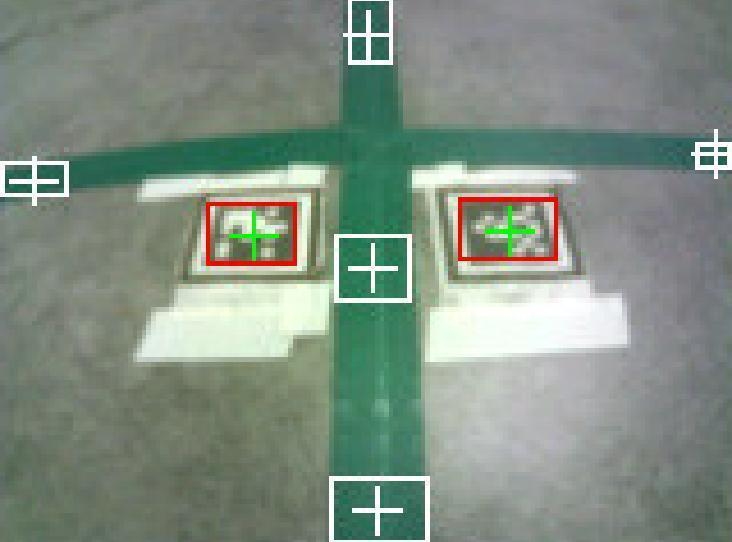


图6.3 识别十字路口AprilTag标签

图6.6是识别十字路口的AprilTag标签，左右标签均能识别得到，所以能够利用软件设计中的图5.12判断此路口为左转路口或是右转路口，亦或是直行路口。同理，可以识别出目标AprilTag标签和掉头AprilTag标签。

6.4 转弯和直行运动测试

对于转弯运动和直行运动，关键点是图6.4所示。



图6.4 即将跨过十字路口

因为图6.7属于下无色块，中、左、右均有色块，所以是即将跨过路口标志。可以按图5.14左转路口处理的流程来处理左转运动，同理处理右转运动。按图5.13 直行路口处理直行路口的运动。

6.5 终点停车和掉头测试

6.5.1 终点停车

终点时测试的到的如下图6.5所示。



图6.5 终点识别

可以看到，识别出的矩形有1个丢失，但是参考软件设计中的图5.11终点识别，可以假设识别到5到6个矩形即为终点。这样便可达到停车的效果。

6.5.2 掉头

返程时的掉头，首先需要手动给掉头AprilTag，这个动作模拟了到达目的后，人们拿掉车上的物品，然后机器人通过原地旋转找直线进行返程。参考图5.10路线识别流程实现。

6.6 返回遇转弯测试

第1个转弯处理，终点识别的掉头AprilTag标签会记录遇第一个转弯的处理，按照转弯运动进行转弯。其它转弯当直行路口的运动处理。

6.7 运行遇障碍物测试

测试读取超声波的数据图6.6所示。

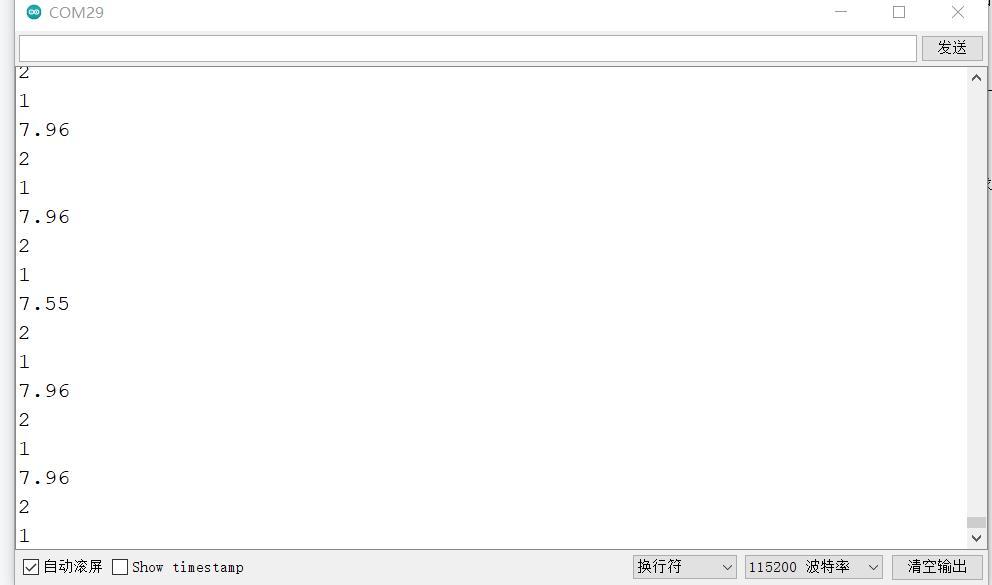


图6.6 Esp32读取的数据

比如图中的7.96为超声波测量里障碍物的距离，2为接收网页客户端发送的目标点，再转发给OpenMV Cam。同时因为设置的距离小于15会进行停车，7.96小于15所以停车发送1。

6.8 拓展WebServer传送目标点2测试

测试网页端登录的用户认证界面如图6.7所示。

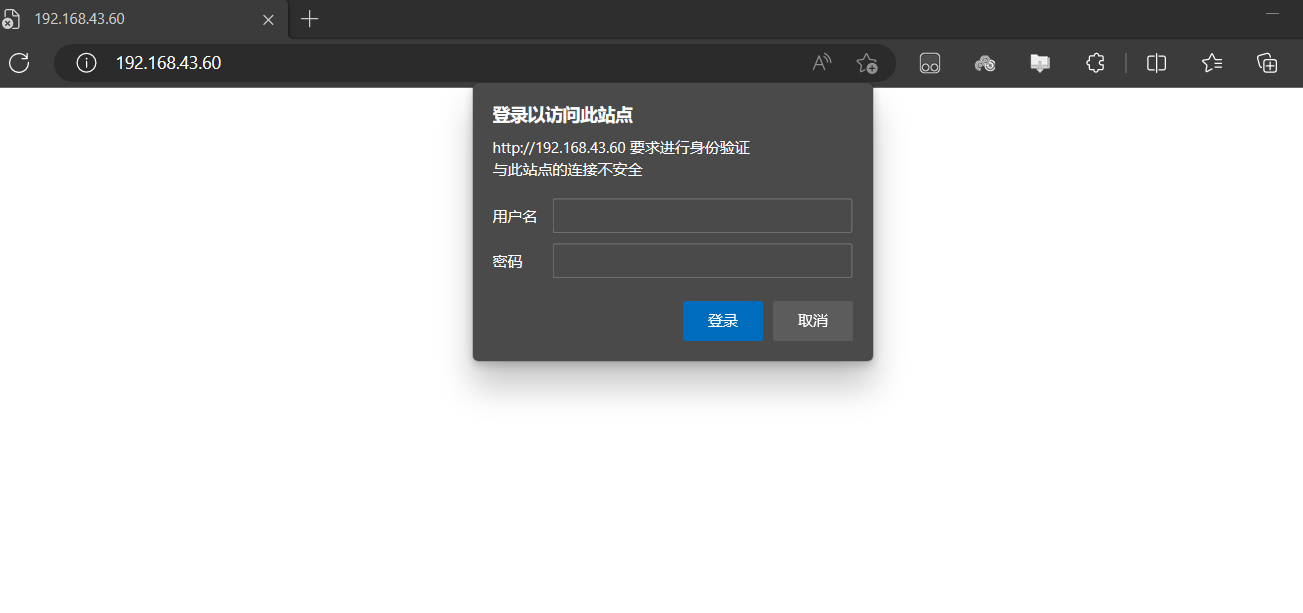


图6.7 用户认证登录

测试网页的发送目标点页面如图6.8所示。

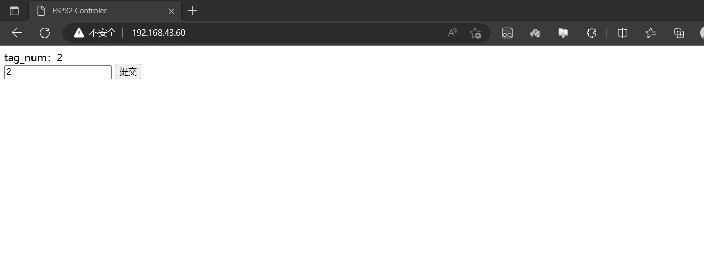


图6.8 网页端发送目标点

图6.8把目标点发送给了Esp32，然后Esp32转发给OpenMV Cam，如图6.6中的目标点2。

6.9 运行总体性能测试

综合上面的测试和具体的流程设计，设计的智能车完全能够实现需求的功能。

6.10 小结

本章是对本课题的所有设计的实物的实地测试，包括识别直线、十字路口、AprilTag，实现了十字路口左右转运动和直行运动、对终点的停车和掉头、返程的十字路口、整个系统无障碍物进行运行和有障碍物运行，完成了课题的所有要求。同时还拓展了WebServer上位机的发送目标位点，实现用户登录认证操作。至此，依据整个系统的实地测试可知，本课题设计的图像识别的智能导航小车满足功能需求。

第7章 总结与展望

7.1 总结

本课题的设计，是为了能够在一些需要减少人员流动的场景，用机器人代替传统的物品运送，比如医院的送药、酒店的生活用品、火车的订餐送餐等。比较有创新性的设计了网页端的用户登录操作，使得整个系统更加合理化和便捷化。因为系统巡线和识别均采用一个OpenMV模块，极大的降低了物品运送机器人的开发成本。

本论文所做的主要工作与成果有：

（1）设计出硬件平台，包括硬件选型、原理图设计、PCB设计和焊接调试，并基于FreeRTOS设计了按四轮差速运动模型（SSMR）进行运动学解析的底盘运动系统。

（2）设计出软件结构，包括OpenMV Cam的图像识别和Esp32的WebServer开发。

（3）进行了系统各方面测试，并完成了系统综合性测试，不仅达到了预期的效果，还进行了网页端上位机控制的拓展，具有明显的安全性和方便性。

7.2 展望

本课题在进行时，遇到了许多方面的问题：

（1）OpenMV Cam的识别受光亮影响较大，会对图像识别产生影响。

（2）超声波模块的稳定性较差，测量的物体距离也比较近。

（3）为了保证Esp32网页端更流畅，Esp32设计的网页端控制界面功能较少。

后续对于问题（1），可以在OpenMV Cam加入合适的灯光进行辅助，对于出现的问题（2）则可以更换更加稳定且测距更远的模块，对于问题（3）也可以尝试更换模块。

参考文献

李锦. 机器人室内导航与避障系统应用研究[D].中国石油大学(北京),2020.DOI:10.27643/d.cnki.gsybu.2020.000966.

李杨. 基于ROS的室内全向自主导航机器人研究[D].中国矿业大学,2020.DOI:10.27623/d.cnki.gzkyu.2020.001344.

李想. 基于SLAM的室内移动机器人导航技术研究[D].哈尔滨工业大学,2018.

秦圣然. 基于激光传感器的移动机器人导航系统研究[D].沈阳工业大学,2020.DOI:10.27322/d.cnki.gsgyu.2020.000527.

李红光. 基于2D激光雷达的物流AGV高精度地图构建方法研究[D].广西大学,2020.DOI:10.27034/d.cnki.ggxiu.2020.000338.

李思亮. 基于ROS与激光SLAM的自主导航移动机器人研究[D].东南大学,2019.DOI:10.27014/d.cnki.gdnau.2019.002079.

任凡. 基于Kinect的移动机器人视觉轨迹跟踪控制研究[D].西安理工大学,2017.

李虹,张宇晨,苗旭焘.OpenMV智能物料搬运方向的开发[J].电子技术与软件工程,2022(03):107-112.

吴必瑞,魏榛灺,姜良麒.OpenMV智能送药机器小车的设计与实现[J].宁德师范学院学报(自然科学版),2023,35(01):52-56.DOI:10.15911/j.cnki.35-1311/n.2023.01.007.

齐曼,胡乃瑞,安天洋,高唯峰.基于STM32和OpenMV的全自动无接触送药小车系统[J].电子制作,2022,30(21):21-23.DOI:10.16589/j.cnki.cn11-3571/tn.2022.21.022.

肖光亚.基于Arduino的智能送药小车[J].南方农机,2022,53(02):184-186.

沈中坤,徐劲节.基于OpenMV视觉模块和MPU6050角度传感器的智慧寻路小车[J].电子制作,2022,30(03):28-30+19.DOI:10.16589/j.cnki.cn11-3571/tn.2022.03.016.

李婕. 基于STM32的无线视频监控智能小车设计[D].兰州理工大学,2014.

蔡自兴,贺汉根,陈虹.未知环境中移动机器人导航控制研究的若干问题[J].控制与决策,2002(04):385-390+464.DOI:10.13195/j.cd.2002.04.1.caizx.001.

丁声雷. 激光导航AGV车载控制系统设计与路径跟踪算法研究[D].西华大学,2018.

Clavien, Lionel,Michaud, Francois,Lauria, Michel. Instantaneous Centre Of Rotation Based Motion Control For Omnidirectional Mobile Robots With Sidewards Off-centred Wheels[J]. ROBOTICS AND AUTONOMOUS SYSTEMS, 2018, 106: 58-68. 10.1016/j.robot.2018.03.014.

Takahashi, Masaki,Yorozu, Ayanori. Obstacle Avoidance With Translational And Efficient Rotational Motion Control Considering Movable Gaps And Footprint For Autonomous Mobile Robot[J]. INTERNATIONAL JOURNAL OF CONTROL AUTOMATION AND SYSTEMS, 2016, 14(5): 1352-1364. 10.1007/s12555-014-0452-2.

陆淼嘉,尹钦仪. “最后一公里”无人车配送发展现状及应用前景[J]. 综合运输, 2021, 43(01): 117-121.

Xu J.,Wang D.,Liao M.,Shen W.. Research Of Cartographer Graph Optimization Algorithm Based On Indoor Mobile Robot[J]. Journal of Physics: Conference Series, 2020, 1651(1):. 10.1088/1742-6596/1651/1/012120.

贾浩. 基于Cartographer算法的SLAM与导航机器人设计[D]. 山东大学, 2019.

王帅军. 基于D\*算法的移动机器人路径规划[D]. 广西大学, 2019.

Gao, Ying,Xiao, Chu-Feng,Ma, Yi-Ning,Gong, Yue-Jiao,Zhang, Jun. Path Planning For Autonomous Underwater Vehicles: An Ant Colony Algorithm Incorporating Alarm Pheromone[J]. IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, 2019, 68(1): 141-154. 10.1109/TVT.2018.2882130.

Tharwat, Alaa,Elhoseny, Mohamed,Hassanien, Aboul Ella,Kumar, Arun,Gabel, Thomas. Intelligent Bezier Curve-based Path Planning Model Using Chaotic Particle Swarm Optimization Algorithm[J]. CLUSTER COMPUTING-THE JOURNAL OF NETWORKS SOFTWARE TOOLS AND APPLICATIONS, 2019, 22(2): S4745-S4766. 10.1007/s10586-018-2360-3.

李婕. 基于STM32的无线视频监控智能小车设计[D].兰州理工大学,2014.

杨明,王宏,何克忠,张钹. 基于激光雷达的移动机器人环境建模与避障[J]. 清华大学学报：自然科学版, 2000, 40(07): 112-116.

郭丽晓. 基于拓扑地图的AGV智能路径规划技术研究[D]. 浙江大学, 2013.

王闻博. 航站楼服务机器人适应性导航方法研究[D]. 中国民航大学, 2020.

李昱辰. 一种面向自动驾驶的多传感器融合SLAM框架[D]. 浙江大学, 2020.

曹勇. 基于多传感器融合的仓储AGV导航定位系统设计与实现[D]. 山东大学, 2019.

程亮. 多约束条件下基于改进遗传算法的路径规划与导航[D]. 重庆邮电大学, 2020.

王健. 基于蚁群算法的车辆导航自适应路径规划算法研究[D]. 青岛科技大学, 2011.

陈跃跃. 基于粒子群优化算法的移动机器人运动规划[D]. 杭州电子科技大学, 2018.

卢宁. 变电站中的室内巡检机器人研究与设计[D]. 山东大学, 2019.

致谢

我终究还是要告别这四年时光的。

感谢指导老师提供的实验室场地和设备，让我能够有足够的条件去完成我的毕业设计。特别感谢魏教授和刘学长在我的学习之路提供的指导，或许他们不会看到这，可这是我最衷心的感谢。

还有，感谢伴我一路走来的叶先生、唐先生、志先生、HLL战队，望未来顶峰相见。

最后感谢家人的陪伴和支持，让我不断前行。

至此，我的大学生活正式落幕。

不温不火，保持热枕，奔赴山海。

附录

测试的实际环境如图1所示，智能车实物如图2所示。

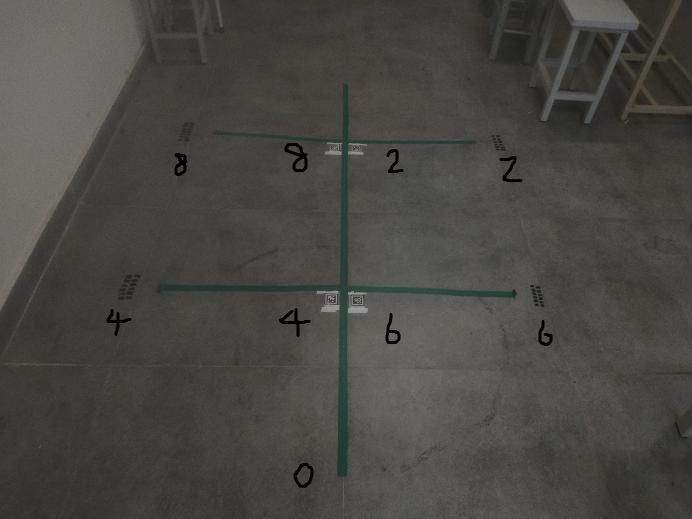


图1 测试环境

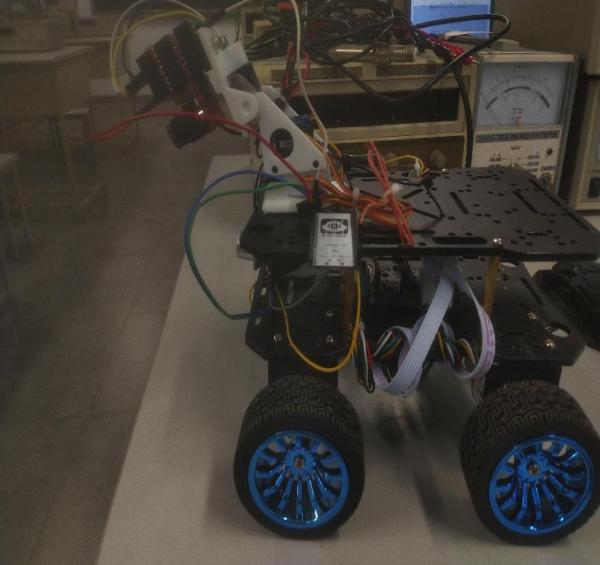
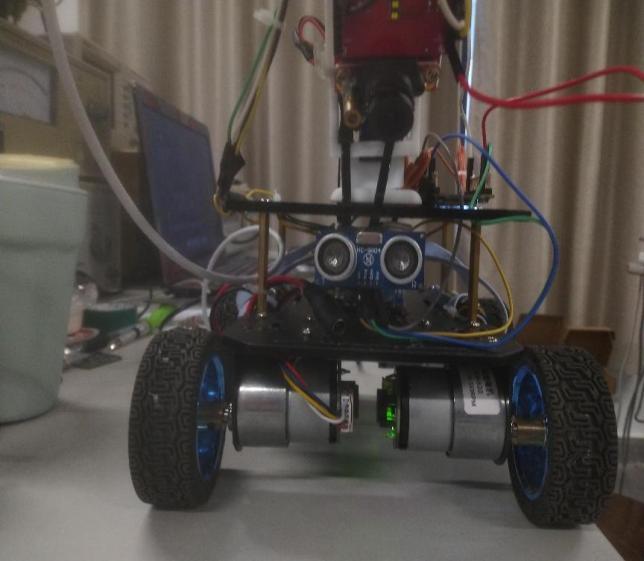
 

图2 智能车实物图