

分类号：_____

密级：_____

UDC：_____

编号：_____

专业硕士学位论文

（工程硕士）

基于瑞典轮的全向移动机器人
定位及路径跟踪研究

硕士研究生：李 辉

指导教师：陈东良 副教授

企业导师：季宝锋 工程师

工程领域：机械工程

论文主审人：王 茁 教授

哈尔滨工程大学

2014年4月

分类号：_____

密级：_____

UDC：_____

编号：_____

专业硕士学位论文

（工程硕士）

基于瑞典轮的全向移动机器人
定位及路径跟踪研究

硕 士 研 究 生：李 辉

指 导 教 师：陈东良 副教授

学 位 级 别：工程硕士

工 程 领 域：机械工程

所 在 单 位：机电工程学院

论文提交日期：2014 年 4 月

论文答辩日期：2014 年 6 月

学位授予单位：哈尔滨工程大学

Classified Index:

U.D.C:

A Dissertation for the Professional Degree of Master
(Master of Engineering)

Research of
Positioning and Trajectory Tracking for
Swedish Wheeled
Omni-Directional Mobile Robot

Candidate:	Li Hui
Supervisor:	Assoc. Prof. Chen Dongliang
Academic Degree Applied for:	Master of Engineering
Specialty:	Mechanical Engineering
Date of Submission:	April, 2014
Date of Oral Examination:	June, 2014
University:	Harbin Engineering University

哈尔滨工程大学 学位论文原创性声明

本人郑重声明：本论文的所有工作，是在导师的指导下，由作者本人独立完成的。有关观点、方法、数据和文献的引用已在文中指出，并与参考文献相对应。除文中已注明引用的内容外，本论文不包含任何其他个人或集体已经公开发表的作品成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

作者（签字）：

日期： 年 月 日

哈尔滨工程大学 学位论文授权使用声明

本人完全了解学校保护知识产权的有关规定，即研究生在校攻读学位期间论文工作的知识产权属于哈尔滨工程大学。哈尔滨工程大学有权保留并向国家有关部门或机构送交论文的复印件。本人允许哈尔滨工程大学将论文的部分或全部内容编入有关数据库进行检索，可采用影印、缩印或扫描等复制手段保存和汇编本学位论文，可以公布论文的全部内容。同时本人保证毕业后结合学位论文研究课题再撰写的论文一律注明作者第一署名为哈尔滨工程大学。涉密学位论文待解密后适用本声明。

本论文（☐在授予学位后即可 ☐在授予学位12个月后 ☐解密后）由哈尔滨工程大学送交有关部门进行保存、汇编等。

作者（签字）：

导师（签字）：

日期： 年 月 日

 年 月 日

摘 要

使用瑞典轮组成的全向移动机器人因其特殊的车轮结构及易于实现控制等优点而被广泛研究。这种全向移动机器人适合在相对平整但空间狭窄的环境中运行，在仓库货物转运、航母飞机调度等领域有较好的发展潜力。

本文依据相关设计指标研究三轮全向移动机器人的定位及路径跟踪算法。研究内容主要包括基于运动学的机器人定位数学模型及定位误差分析、机器人几何路径跟踪算法与位姿跟踪误差校正控制器的设计、基于微处理器的机器人嵌入式控制系统实现以及机器人样机实验研究四个方面。

首先，介绍基于三个瑞典轮的机器人底盘和随动轮系的相关结构。根据机器人运行时所需的技术指标，确定相关的尺寸结构。针对机器人的结构特点，建立机器人的局部和全局参考框架。按照先局部后整体的路线，依次建立单个瑞典轮和机器人整体的通用运动学模型。确定具体的驱动轮及其在局部参考框架下的位姿参数，代入机器人整体模型得到机器人的运动学控制方程。

其次，推导基于里程计和陀螺仪的定位数学模型，得到机器人的实时位姿更新方程。分析机器人定位误差的来源及误差校正的模型与方法。以定位为基础，设计机器人基于直线和圆弧的几何路径跟踪算法，分析了机器人在路径跟踪时的位姿误差，设计了基于PID的误差校正控制器。

再次，完成机器人控制系统设计。其中，包括基于串行口和并行口的通信、定位子系统设计、直流伺服电机驱动系统设计。以实时操作系统为基础设计机器人定位及几何路径跟踪的软件框架及控制流程。

最后，对机器人几何路径跟踪算法进行仿真，分析不同的实验条件对仿真结果的影响。介绍样机实验环境及实验数据采集、测量的方法。通过样机实验校正传感器本身和安装误差。在不同的样机实验条件下，分析机器人的位姿跟踪误差。

关键词：全方位；瑞典轮；定位；嵌入式控制系统；几何路径跟踪

Abstract

The omni-directional mobile robot with Swedish Wheels is extensively studied because of its many possible advantages, such as better flexibility, simpler structure and controlling more easily. This robot is suitable for the flat and narrow work environment and it can be used to carry goods in warehouses and participate in aircraft carrier scheduling.

The specific goal of this study was to develop an effective control method, positioning and path tracking, for the robot. The main content consists of four parts: deriving positioning model with the robot's kinematics and studying its error, investigating path tracking method and its error regulator, implanting the positioning and path tracking methods in a embedded control system (ECS) which is based on micro control units (MCU) and conducting experiments

Firstly, structure of the omni-directional mobile robot was introduced. The structure dimensions was determined by necessary technique data. Local and global reference frame of the robot were established based on the structural characteristics of the robot. According to the route that overall after partial, a common kinematic model of single Swedish wheel was built and then the overall kinematic model was obtained by combining these single models. The parameters of the position and orientation of each Swedish wheel in its local frame were determined and kinematic governing equations of the robot were derived by taking these parameters into the whole model.

Secondly, By developing the mathematic positioning model based on odometry and gyroscope, update equation for robot positioning was obtained. Source of the positioning error was analyzed, and a error compensation model was developed followed by relative methods. Straight line and circle trajectory tracking algorithm was designed based on positioning. Then, path tracking error was analyzed and a Proportional Integral Derivative (PID) controller was developed for compensating the tracking error.

Thirdly, the embedded robot control system was designed, including communication subsystem based on serial and parallel, positioning and servo motor control subsystem. Robot software framework based on Real Time Operating System (RTOS) was developed for positioning and geometric trajectory tracking, in which the robot overall control process

worked.

Finally, simulations were conducted aiming at verifying the algorithm for positioning and geometric trajectory tracking. The result of the simulations under different conditions were analyzed. Robot prototype testing environment and the methods of experimental data acquisition and measurement were introduced. The error of odometry and gyroscope was compensated through robot prototype testing. Position and orientation tracking error was analyzed under different test conditions.

Key words: Omni-Directional; Swedish Wheel; Positioning; Embedded Control System; Geometric Trajectory Tracking.

目 录

第 1 章 绪论	1
1.1 课题来源及研究的目的和意义	1
1.1.1 课题来源和背景	1
1.1.2 研究目的及意义	1
1.2 基于瑞典轮的全向移动机器人研究现状	2
1.3 全向移动机器人定位技术发展现状	3
1.4 全向移动机器人路径跟踪发展现状	5
1.5 本论文的主要研究内容	6
第 2 章 全向移动机器人整体设计及运动学分析	9
2.1 引言	9
2.2 机器人运行指标及总体结构介绍	9
2.2.1 机器人的设计指标	9
2.2.2 机器人系统组成	9
2.3 三轮全向移动机器人结构分析	10
2.3.1 瑞典轮运动特性分析	10
2.3.2 基于瑞典轮的机器人底盘	11
2.3.3 基于瑞典轮的机器人定位结构	11
2.4 机器人的运动学分析	12
2.4.1 机器人的参考框架	12
2.4.2 瑞典轮的运动学约束	13
2.4.3 机器人整体运动学分析	15
2.5 本章小结	17
第 3 章 全向移动机器人定位及路径跟踪算法研究	19
3.1 引言	19
3.2 机器人定位及路径跟踪整体方案分析	19
3.3 基于编码器和陀螺仪的机器人定位实现	20
3.4 机器人定位误差分析	22
3.4.1 机器人定位误差来源分析	22

3.4.2 机器人定位误差的校正	24
3.5 机器人几何路径跟踪及姿态控制	29
3.5.1 机器人几何路径跟踪	30
3.5.2 机器人的姿态控制	32
3.6 基于 PID 的机器人位姿跟踪误差校正	33
3.6.1 机器人几何路径跟踪的误差计算	33
3.6.2 机器人路径校正的 PID 控制器	34
3.7 本章小结	35
第 4 章 全向移动机器人嵌入式控制系统设计与实现	37
4.1 引言	37
4.2 控制系统总体结构	37
4.3 机器人主控芯片	38
4.4 控制系统通信设计	38
4.4.1 基于双端口 RAM 的并行通信	38
4.4.2 基于串行口的通信	40
4.5 直流伺服电机控制	42
4.5.1 电机控制方案	42
4.5.2 电机控制器 LM628 设计接口	42
4.5.3 功率放大模块	44
4.5.4 直流电机控制的软件设计	44
4.6 机器人定位子系统设计	46
4.6.1 基于编码器的里程测量系统	46
4.6.2 基于陀螺仪的姿态测量系统	49
4.7 机器人控制系统软件设计	52
4.7.1 基于 UCOS-III 的机器人控制软件的实现	52
4.7.2 机器人软件优化分析	56
4.8 本章小结	58
第 5 章 全向移动机器人定位及路径跟踪实验	59
5.1 引言	59
5.2 全向移动机器人路径跟踪算法仿真实验	59

5.2.1 运动学仿真模型建立	59
5.2.2 仿真实验结果及分析	60
5.3 全向移动机器人样机定位实验	61
5.3.1 样机实验环境介绍	61
5.3.2 编码器误差校正	64
5.3.3 陀螺仪及几何中心偏差测量实验	65
5.3.4 样机重复定位实验	66
5.3.5 机器人样机路径跟踪实验	67
5.4 本章小结	72
结 论	73
参考文献	75
攻读硕士学位期间发表的论文和取得的科研成果	81
致 谢	83

第1章 绪论

1.1 课题来源及研究的目的和意义

1.1.1 课题来源和背景

本课题来源于亚太机器人电视大赛哈尔滨工程大学代表队，基于瑞典轮的全向移动机器人开发。

1.1.2 研究目的及意义

机器人是当代自动化技术和人工智能技术发展的典型体现，也代表了一种新的制造技术水平^[1]。机器人技术的发展是衡量一个国家高端科学技术水平的重要标志和体现。随着计算机技术、自动控制理论、传感器技术、仿生机构学、人工智能等学科的蓬勃发展，机器人技术作为一类综合性的高新技术已广泛应用于人们的生活和生产中，逐渐朝着替代人类工作的方向发展^[3,4]。

机器人自诞生到现在已有了长足进步和发展。在种类众多的机器人类型中，能够在工作空间中运动的移动机器人更是发展迅速，已成为机器人学中繁荣发展的一个分支。以机器人整体构成和运动方式为分类依据，移动机器人可以分为与运动环境以轮式接触、履带式接触、腿式接触等类型的机器人（包括仿人机器人）。本文研究的对象是一种可以全方向移动的轮式移动机器人^[2]。

所谓全方位移动机器人，就是指如果不考虑驱动电机转速限制，机器人能够在任何时刻沿着平面上的任何方向运动^[25-27]。通常，大部分轮式机器人都使用差分驱动轮系作为机器人的驱动系统。这样的机器人能够在不同曲率半径的圆弧或者直线上运动，也能够原地旋转，但它不能沿轮子平面的法线方向运动。这个限制对全方位移动机器人来说并不存在。全向移动机器人在底盘的机械结构上使用瑞典轮（又称作 Mecanum 轮）作为驱动轮。瑞典轮的轮毂边缘装有很多可以自由滚动的小轮，可以满足移动机器人全方位移动的要求^[25]。全方位移动机器人适合工作在空间较小的环境中，能够以较高的机动性完成相对复杂的任务。例如，货物仓库的动态整理、工厂车间零件及材料的自动运输甚至是航空母舰上飞机在飞行甲板上的运输调度等等。因此，全方位移动机器人具有广阔的应用前景。许多国内外的学者针对全方位移动机器人一直从事相关的研究工作，在全向轮的结构设计、车轮在机器人上的布局方案和优选^[14,27]，以及针对不同结构的机器人的运动学、动力学分析等方面，开展了比较深入、全面的研究^[21-24]，并且也生产了许

多各具特色的机器人产品^[15-18]。

在机器人相关技术的发展进程中，控制问题始终是机器人的一个特别重要的研究方向。对于全向移动机器人，导航又是其中最关键也是难度最大的问题。机器人的导航是指机器人根据它的路径规划信息和传感器的反馈值使之尽量有效和可靠地完成它期望位姿变化的过程^[2-4]。

综合前文的论述，导航问题是移动机器人研究的核心问题，决定了机器人是否可以成功地应用于人们的日常生产生活中。移动机器人的定位与路径跟踪是机器人导航的重要组成部分。所以，本课题针对移动机器人在机器人实时定位以及路径跟踪算法上的探索具有重要的价值。

1.2 基于瑞典轮的全向移动机器人研究现状

现阶段，驱动轮的不同是全向移动机器人在机械结构上的主要区分点。常用的典型驱动轮有：对轮毂边缘做特殊处理的瑞典轮^[25]、使用完全不同构造的球轮、针对一种特定瑞典轮优化后的连续切换轮和日常生活中常见的万向轮等。

瑞士联邦理工大学设计的 Tribolo 机器人（图 1.1）使用三个球形轮，每个轮都有三个支撑点。其中两个点由球形轴承支撑，另外一个点由电机驱动。该机器人具有很好的机动性能，但是对运行环境要求严格，同时机器人制造、装配的难度较大^[36]。

如图 1.2 所示，使用普通车轮构成的全向移动机器人，在每个驱动轮上添加了旋转自由度。这种机器人的车轮能够根据控制需要改变姿态，从而影响整个机器人的运动方向。



图 1.1 Tribolo 机器人



图 1.2 室外四轮全向移动机器人

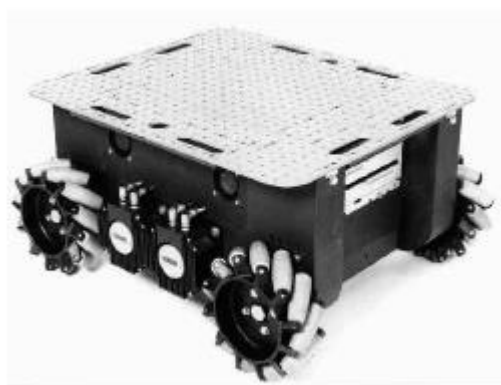


图 1.3 Uranus 机器人



图 1.4 三轮全向移动机器人

美国学者 Overholt J 等人研究了基于主动分离补偿车轮模块的室外四轮全向移动机器人（图 1.2）。该机器人带有车轮主动悬挂结构，能够保证机器人在地面不规整的室外环境里平稳运行^[36]。赵冬斌等人研究了基于主动控制万向轮的通用运动学及动力学模型，并分别分析了该类型机器人的运动控制方法，以及挤压力同时控制等相关的机器人控制^[25,29]。

卡内-基梅隆大学分别使用四个偏置角为 45° 的瑞典轮以及三个偏置角为 90° 的瑞典轮，设计制作了机器人 Uranus（图 1.3）和三轮全向移动机器人（图 1.4）。该设计结构简单，控制模型容易建立，得到广泛的关注。有学者分析了基于瑞典轮的全向移动机器人运动学及动力学模型^[65-67]。我国学者在机器人运动学与动力学模型的基础之上，针对基于瑞典轮的全向移动机器人的运动速度进行了详细的分析^[68]。由于在运动特点上不同于普通差分驱动机器人，所以基于瑞典轮的全向移动机器人的定位技术以及路径跟踪控制也倍受关注。

1.3 全向移动机器人定位技术发展现状

导航是机器人通过自身配备的传感器认识其周围的环境，并根据任务需求实时改变运行状态以适应环境，最终到达目标任务点的动态交互过程。美国麻省理工学院研究移动机器人的学者针对导航提出了以下三大方面的研究问：“我在哪里？”，“我将要去什么地方？”，“我该如何到达目的地？”，即导航要求机器人确定自己的位姿，具有充足的知识来进行路径规划和通过合适的驱动方案实现路径跟踪。其中感知和定位技术的研究在以往的几十年里受到了很大的关注，在这方面也取得了重大的进展。

作为轮式移动机器人的一大类，全向移动机器人具有和其他类型的机器人相似的运行环境参数、驱动方式。因此，针对移动机器人的相关定位技术同样适用于本课题的研

究对象。

1、地图的表达方式

机器人的定位首先要确定其定位的参考框架，即依据机器人运行的环境建立一个模型（地图）。环境地图通常有如图 1.5 所示的几种类型。

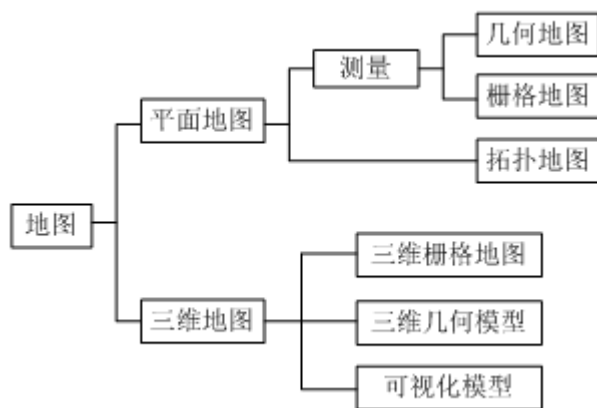


图 1.5 环境地图的分类

上述的三种平面地图都是将机器人的运行环境按照一定的方法进行分解，然后运用不同的理论实现二维环境建模。Mataric 首次实现了拓扑地图在机器人领域的应用^[6]。拓扑地图具有运行空间小、容易使用高效算法的特点，主要应用在基于图像处理的大型导航中。几何地图则以运行环境中的特征点为依据进行环境建模，特征观测与匹配是该方法的应用难点。栅格地图是以传统的二维几何位置表达框架为基础，对整个环境进行离散化建模的方式^[7]。三维地图通常是在二维模型的基础上增添一个空间维度，其地图的建模方法通常也基于二维地图的相关理论，但是更为复杂。

2、定位的方法

随着科技的不断发展，计算机技术和传感器技术有了长足的发展，同时也大大丰富了机器人领域的定位方法。移动机器人的定位技术具有多种分类方式，根据使用的传感器类别的不同，移动机器人领域主要的定位方法有基于里程计和惯性器件的定位、运动图像处理和模式识别技术的视觉定位、基于多角度测距的定位、使用 GPS 导航仪进行精度要求较低的定位，以及结合上述的一种或者几种定位方法的联合定位方案等。

以车轮为运动载体的机器人，通常用伺服电动机来驱动。而光电编码器是伺服电机的必备组件，它不仅可以用于伺服电机的闭环控制，而且可以通过机器人的运动学模型对机器人的位姿进行估算。美国的机器人专家 Borenstein J 早在九十年代就针对差分驱动的机器人进行里程计定位及误差的校正研究，并最终总结出一套科学的校正方法——UMBMark^[5]。王卫华等人针对 UMBMark 存在的缺陷，进一步地改进了该方法，并通过

试验验证了算法的正确性^[8,9]。刘玉鹏等人通过每个车轮的编码器信息，建立四轮全向移动机器人的车轮打滑判断模型，提高了机器人的快速自定位精度^[26]。随着传感器技术的不断发展与机器人应用指标的提高，使用单一的传感器（如里程计）已经远远不能满足定位要求。Martinelli A 等学者使用激光雷达、光纤陀螺仪等先进的传感器，研究了机器人里程计辅助校正方式^[29-34]。

基于路标的定位是根据人为添加或者环境固有的标示物进行实时监测与位置估计的过程。该方式要求对环境中的标示物进行预先定位，或者根据机器人的应用要求组建具备标示物特征的环境。在机器人的运动过程中，通过图像或者距离传感器等方式实时检测路标，并估计机器人位置最终达到机器人定位的目的^[8]。

基于 GPS 的定位方式通过覆盖整个地球表面的多个人造地球卫星，使用一定的姿态解算方法，实现了地球表面物体的绝对定位。使用该方法的机器人通常是在室外的环境下运行，例如无人驾驶汽车等。同时，由于定位覆盖的范围较大，通常定位的精度较低，一般在 5-10 米左右。因此，机器人经常使用其他的传感器并通过数据融合算法，进行辅助定位^[65]。

基于多传感器信息融合的概率定位是当前机器人定位领域的研究热点和难点。其中，使用 Markov 方法、Monte Carlo 方法和 Kalman 方法的定位应用取得了比较好的效果。多传感器融合的定位常常应用于机器人系统结构和运行环境都比较复杂的机器人上。为了实现复杂的算法，机器人一般使用通用型的计算机作为其控制中心，因此，整体结构相对比较大。

1.4 全向移动机器人路径跟踪发展现状

路径跟踪是机器人运动控制的重要内容之一，它的实现需要自动控制理论的支撑。机器人路径跟踪问题的研究离不开机器人的运动学及动力学模型。

移动机器人的控制问题来源于对实际生产的需要。在生产的过程中，移动机器人是否能够快速、准确地完成移动任务对于生产效率的影响是巨大的，为了实现机器人能够快速、准确地运动或者准确地跟踪某一特点曲线的目标，众多学者将精力投入到这一领域中，并且迅速的将控制系统的研究发展成对点镇定（point stabilization）、路径跟随（path following）和轨迹跟踪（trajectory tracking）三个基本问题的研究。

由于控制系统不具有一定的实体，相对较为抽象，所以对以上问题进行研究时，首先需要对控制系统进行建模。常用的建模方式从机理上可分为两种：

- 1、机理建模：结合物理知识对所研究的控制系统进行推导，获得所需要的各项参

数后，再结合相关数学知识建立控制系统的数学模型。

2、统计建模：该模型建立在统计学理论基础之上，根据统计学原理，对数学模型所需要的各项理论参数进行估计，或者采用系统识别方式获得所需参数。

从移动机器人的认知与运动控制方式角度来说，进行机器人路径跟踪的前提主要有以下两部分构成：1、在机器人的数学模型基础上建立满足运动学或者动力学方程的数学模型。2、根据现有定位信息对运动路径进行规划。第一部分内容是从机械系统角度对移动机器人的控制系统进行阐述，机器人的路径跟踪算法通过该模型计算相关执行体的输出，可充分反应出机器人的运动学和动力学性能。而第二部分的研究内容从本质上来说是一种优化算法，但是这种算法具有很多约束条件。例如：在一个具有若干障碍物的空间内，移动机器人的任务是从其中的 A 点到达 B 点，同时又不碰到其中的障碍物，这就是第二部分所研究的内容。机器人的路径跟踪以经过优化的路径为控制目标，同时借助机器人的定位信息计算路径跟踪的误差，进而确定机器人下一步的宏观运动。

很多国内外学者针对路径跟踪相关的机器运动控制问题展开了研究。Huang and Tsai 研究了自主全向移动机器人的嵌入式鲁棒自适应控制器，但是控制器的相关参数优化主要是通过试错法来实现的^[71]。Hsu-Chih Huang 等人使用模糊 PID 研究了基于瑞典轮的全向移动机器人路径跟踪及镇定控制，仿真实验取得了比较理想的效果^[72]。Liu Y 等人设计实现了基于路径线性化的机器人路径跟踪控制器^[73]。

1.5 本论文的主要研究内容

本文的主要工作是设计基于瑞典轮的全向移动机器人底盘结构。实现基于里程计和陀螺仪的机器人全场定位。在定位的基础上设计机器人几何路径快速准确的跟踪算法，完成基于微处理器的机器人嵌入式控制系统的实现，并在机器人的实验样机上进行相关的实验验证。

在实现基本的定位功能的基础上，对定位误差进行有效地校正。由于里程计的累计误差会随着时间的推移而不断增加，陀螺仪的输出也会逐渐产生漂移。如何针对传感器本身特点、定位系统存在的系统模型误差、机器人运行环境以及其他辅助手段对机器人的定位误差进行校正是本文的难点。

在机器人运动控制方面，本文针对所研究的机器人设计工程应用型机器人路径跟踪算法，这就要求机器人具有较好的实时性、控制鲁棒性。机器人采用传统的 PID 算法实现跟踪误差的校正。

第一章：介绍本课题背景，全向移动机器人的国内外发展现状，关键技术的简述，主要的工作和论文的内容。

第二章：对基于瑞典轮的全向移动机器人进行结构设计，同时根据机器人定位方案设计基于瑞典轮和光电编码器的机器人定位随动轮系结构；根据机器人的运动环境及机器人的结构特点，建立驱动轮的运动学约束，进而推导得到机器人整体的运动学模型，为后面章节的机器人定位及运动控制打下基础。

第三章：本章包含了机器人导航的两个主要方面——定位和运动控制。首先介绍基于里程计和陀螺仪的机器人位姿更新方程，根据误差传播的相关理论建立机器人定位系统的误差传播模型，为机器人定位误差的校正打下基础。运动控制方面主要是论述机器人对几何路径跟踪的控制方式。

第四章：设计机器人的控制系统。其中，包括常用通信方法的接口电路及软件流程、定位子系统中编码器接口和陀螺仪数据采集接口设计及其软件流程、基于速度伺服的电机控制系统设计。以实时操作系统为基础设计机器人定位及几何路径跟踪的软件框架及控制流程。

第五章：通过机器人的运动学仿真模型，验证机器人定位及几何路径跟踪的正确性，介绍机器人样机及相关实验环境，介绍不同实验条件下的实验方法及结果，并对结果进行分析。

zhi ku quan 20150807

第2章 全向移动机器人整体设计及运动学分析

2.1 引言

本课题研究的全向移动机器人使用瑞典轮作为驱动轮，主要实现了机器人在平面运动环境下的全场定位以及简单的几何路径跟踪两个功能。本章介绍机器人的整体结构设计，根据机器人运行时所需的技术指标，确定相关的尺寸结构。对设计的机器人进行运动学分析，为后面章节的机器人运动控制分析打下基础。

2.2 机器人运行指标及总体结构介绍

2.2.1 机器人的设计指标

根据赛事的规则制定了如下的设计指标，同时对一些特殊场合的应用（见 1.1 节）具有参考价值。

- (1) 运动形式：平面三自由度
- (2) 驱动方式：直流伺服电机
- (3) 最大平移运动速度：3m/s
- (4) 最大自转运动速度：120°/s
- (5) 重量：自身+负载重量小于等于 20kg
- (6) 路径跟踪误差：小于 0.5rad
- (7) 位置误差：1cm/m
- (8) 姿态误差： $\pm 1.5^\circ$
- (9) 工作温度：-10℃-70℃
- (10) 控制频率：200Hz

2.2.2 机器人系统组成

课题研究的全向移动机器人是典型的机电一体化案例。机器人整个系统主要由嵌入式控制系统、定位传感器及其结构和机器人底盘驱动结构三大部分组成。机器人的系统结构框图如图 2.1 所示。

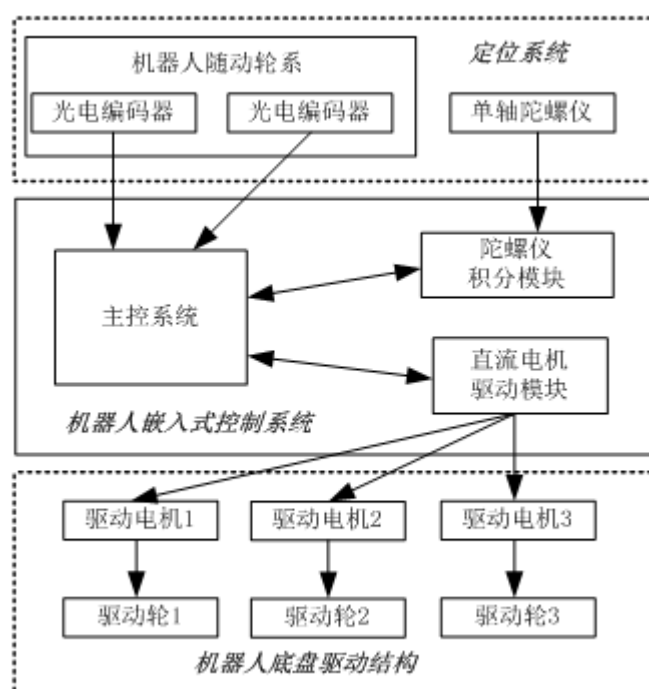


图 2.1 机器人的系统结构框图

本章将论述机器人的底盘和定位相关的结构，主要包括底盘驱动电机与驱动轮的安装、驱动轮的布局；机器人定位随动轮系的上下浮动结构及压紧机构。机器人的嵌入式控制系统的主要组成及实现将在第四章中详细介绍。

2.3 三轮全向移动机器人结构分析

2.3.1 瑞典轮运动特性分析

瑞典轮又叫做麦克纳姆轮，如图 2.2 及图 2.3 所示。瑞典轮没有垂直旋转轴，但通过合适的组合而成的机器人能够全方位的移动，即在二维的笛卡尔平面坐标系内具有三个自由度。

普通标准车轮只能沿着与车轮中心轴垂直方向运动，因此，在使用这种普通车轮设计全向移动机器人时，需要添加可以使车轮垂直旋转的驱动机构，这将大大增加结构设计及控制的难度。瑞典轮最大的结构特点是在普通标准轮毂边缘上添加了自由度，固定的轮毂边缘由可以自由滚动的橡胶辊子代替。机器人在运动过程中，车轮的实际运动方向是由车轮主轴和当前时刻与地面接触的边缘滚主轴的两个运动向量合成。这样，通过设定合适的两个速度向量，就可以实现车轮的全方位移动。

全方位瑞典轮种类有很多种，主要由自由辊子轴线和车轮轴线的夹角 γ 来区分。常用的两种瑞典轮是 $\gamma=90^\circ$ 和 $\gamma=45^\circ$ 的车轮。

图 2.2 $\gamma = 90^\circ$ 瑞典轮图 2.3 $\gamma = 45^\circ$ 瑞典轮

2.3.2 基于瑞典轮的机器人底盘

由于全方位轮的结构特殊性，全方位移动机器人可以由不同数量的瑞典轮组成，理论上说车轮数量可以是大于 2 的任意值，但从可控性以及经济性方面考虑，常见的有 3 轮、4 轮组成。本文研究的机器人是瑞典轮呈正三角布置的结构，它无需使用悬挂机构而依靠自身的重力即可保证三个瑞典轮全部与地面接触。而对于四轮的结构，通常需要增加相应的悬挂机构来保证瑞典轮在运动场地不平整时仍与地面接触，从而保证其运动学模型的有效性。机器人的底盘结构模型及样机实物如图 2.4 和图 2.5 所示。

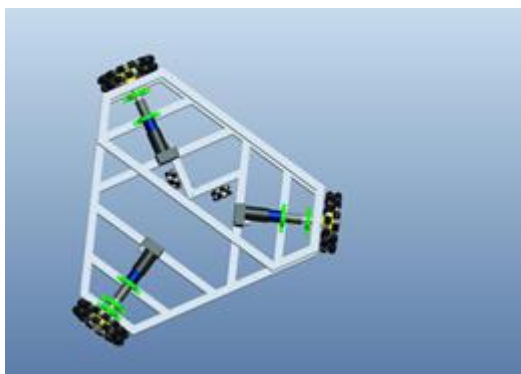


图 2.4 机器人底盘结构模型



图 2.5 机器人样机实物

2.3.3 基于瑞典轮的机器人定位结构

本文论述的机器人使用里程计对机器人的位置进行估计。里程计的测量通过小全向轮带动光电编码器的旋转实现。为了尽可能保证里程计信息的准确性，随动轮与机器人底盘的连接添加了直线滑轨和压紧装置，装置的三维模型和结构实物分别如图 2.6 及图 2.7 所示。因为机器人运动场地不平整，上述结构有效地减少了随动轮系的小全向轮不

能完全与地面接触的现象，从而也尽可能保证了测量信息的可信性。



图 2.6 机器人定位结构模型



图 2.7 机器人定位结构实物

2.4 机器人的运动学分析

本节首先引出移动机器人导航所涉及的两个参考框架，进而根据 2.2 节中的机器人结构，分别对单个瑞典轮及机器人运动底盘进行运动学分析。最终得到机器人运动控制所需的运动学方程。

2.4.1 机器人的参考框架

为了推导机器人的运动学模型，需要分析单个驱动轮对机器人整体运动的影响。根据机器人底盘的结构特点，多个驱动轮的约束联合起来就形成了机器人整体的运动约束。机器人实际上是一个在平面上运动的刚体。为了确定机器人的位置和姿态，建立如图 2.8 所示的机器人参考框架示意图。

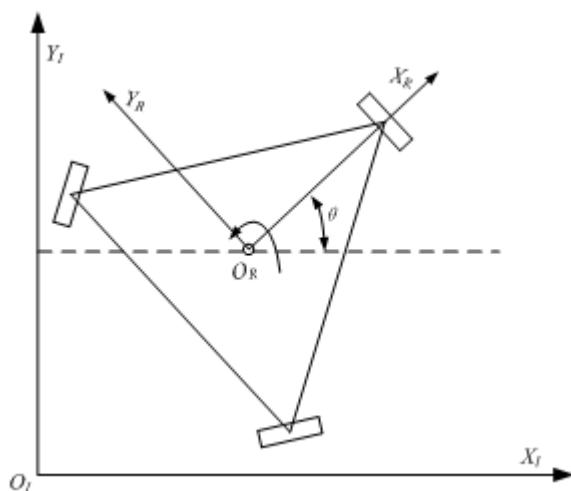


图 2.8 机器人的参考框架

上图描述了机器人全局参考框架和机器人局部参考框架之间的关系。图中，选定平面上的任意一点 O_I 为原点，建立坐标系 $X_I O_I Y_I$ 为机器人全局参考框架。选定机器人刚性底盘上的一点 O_R 作为机器人整体的位置参考点，以该点为原点建立如图所示的机器人局部参考框架 $X_R O_R Y_R$ 。下标（或者上标）“ I ”和“ R ”分别表示机器人全局参考框架和局部参考框架的标示，本文在其他章节的论述中将遵循该规则，不再说明。

机器人定位的最终目的是要确定机器人在全局参考框架下的位置 X_I 、 Y_I 以及姿态角 θ 。本文论述的机器人使用惯性测量元件陀螺仪测量机器人的姿态角，该信息本身就是机器人参考框架下的表达。而前面章节已经说明，用于机器人定位的随动轮系是固连在其本体结构上的，所以里程计本身的信息是机器人局部参考框架下的表达。

机器人在定位及路径跟踪运动时，控制器最直接的输出是机器人在全局参考框架下的运动速度，然而最终的执行体是驱动轮的电机，驱动轮及电机本身又是建立在机器人局部参考框架下的。所以，不论是机器人的定位还是运动控制，都需要相关信息在两个参考框架之间进行转换。该转换可以用如下正交旋转矩阵实现：

$$\mathbf{R}(\theta) = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2-1)$$

利用该矩阵及其逆阵即可实现定位和运动控制信息在两个参考框架之间两个方向的转换。

2.4.2 瑞典轮的运动学约束

假设机器人上第 i 个瑞典轮在机器人局部参考框架下的位姿表达如图 2.9 所示。在进行单个车轮的运动学约束分析之前，先作如下假设：

- 1、假设轮子与地面总是保持垂直。
- 2、瑞典轮在运动过程中，边缘橡胶辊子与地面没有相对滑动。
- 3、假设三个瑞典轮的结构参数相同。

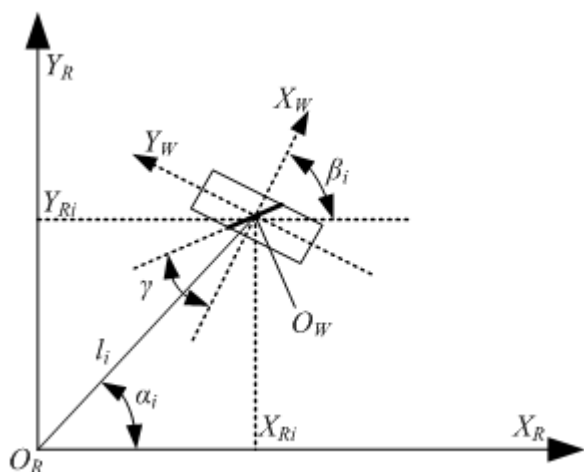


图 2.9 瑞典轮在局部框架的位姿示意图

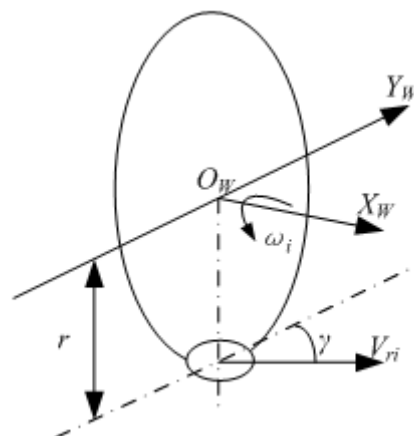


图 2.10 瑞典轮运动示意图

$X_R O_R Y_R$ —表示机器人局部参考框架；

$X_W O_W Y_W$ —表示与机器人第*i*个驱动轮中心 O_W 固联的坐标系；

γ —表示瑞典轮橡胶辊子的偏置角；

l_i —表示瑞典轮中心 O_W 到机器人局部参考框架中心 O_R 的距离；

α_i —表示第*i*个驱动轮中心 O_W 及机器人局部参考框架中心 O_R 连线 $\overline{O_R O_W}$ 与机器人局部参考框架坐标轴 X_R 的夹角；

r —表示瑞典轮的半径；

V_{ri} —表示与地面接触的橡胶辊子中心速度；

ω_i —表示瑞典轮的转动角速度；

$(X_{Ri}, Y_{Ri}, \beta_i)$ —表示第*i*个驱动轮在机器人局部参考框架下的位姿表示；

$\begin{bmatrix} V_X^R & V_Y^R & \omega \end{bmatrix}^T$ —表示 O_R 的广义速度；

$\begin{bmatrix} V_{Xi}^W & V_{Yi}^W & \omega_i^W \end{bmatrix}^T$ —表示第*i*个驱动轮中心 O_W 在坐标系 $X_W O_W Y_W$ 下的广义速度；

$\begin{bmatrix} V_{Xi}^R & V_{Yi}^R & \omega_i^R \end{bmatrix}^T$ —表示第*i*个驱动轮中心 O_W 在坐标系 $X_R O_R Y_R$ 下的广义速度。

由图 2.10 以及理论力学基本知识，得到如下关系：

$$\begin{bmatrix} V_{Xi}^W \\ V_{Yi}^W \end{bmatrix} = \begin{bmatrix} 0 & \sin \gamma \\ r & \cos \gamma \end{bmatrix} \begin{bmatrix} \omega_i \\ V_{ri} \end{bmatrix} = \mathbf{R}_{1i} \begin{bmatrix} \omega_i \\ V_{ri} \end{bmatrix} \quad (2-2)$$

根据图 2.9 中坐标系 $X_W O_W Y_W$ 与坐标系 $X_R O_R Y_R$ 的夹角关系 β_i ，可以将轮毂中心的速度换算到机器人局部参考框架下。

$$\begin{bmatrix} V_{Xi}^R \\ V_{Yi}^R \end{bmatrix} = \begin{bmatrix} \cos \beta_i & -\sin \beta_i \\ \sin \beta_i & \cos \beta_i \end{bmatrix} \begin{bmatrix} V_{Xi}^W \\ V_{Yi}^W \end{bmatrix} = \mathbf{R}_{2i} \mathbf{R}_{1i} \begin{bmatrix} \omega_i \\ V_{ri} \end{bmatrix} \quad (2-3)$$

同时，机器人在平面上运动时，有如下关系：

$$\begin{bmatrix} V_{Xi}^R \\ V_{Yi}^R \end{bmatrix} = \begin{bmatrix} 1 & 0 & -l_{Yi} \\ 0 & 1 & l_{Xi} \end{bmatrix} \begin{bmatrix} V_X^R \\ V_Y^R \\ \omega \end{bmatrix} = \mathbf{R}_{3i} \begin{bmatrix} V_X^R \\ V_Y^R \\ \omega \end{bmatrix} \quad (2-4)$$

式中， l_{Xi} 、 l_{Yi} 分别表示车轮中心 O_w 到机器人局部参考框架 $X_R O_R Y_R$ 坐标轴 Y_R 、 X_R 的距离。

由式 (2-3) 和式 (2-4) 可得

$$\mathbf{R}_{2i} \mathbf{R}_{1i} \begin{bmatrix} \omega_i \\ V_{ri} \end{bmatrix} = \mathbf{R}_{3i} \begin{bmatrix} V_X^R \\ V_Y^R \\ \omega \end{bmatrix} \quad (2-5)$$

又因为： $\det(\mathbf{R}_{1i}) \neq 0, \det(\mathbf{R}_{2i}) \neq 0$

可将式 (2-5) 进一步整理成：

$$\begin{bmatrix} \omega_i \\ V_{ri} \end{bmatrix} = \mathbf{R}_{2i}^{-1} \mathbf{R}_{1i}^{-1} \mathbf{R}_{3i} \begin{bmatrix} V_X^R \\ V_Y^R \\ \omega \end{bmatrix} \quad (2-6)$$

这样即可得到瑞典轮相对于机器人局部参考框架下的逆运动学方程。

2.4.3 机器人整体运动学分析

2.3.2 节已经建立了单个瑞典轮在机器人局部运动框架下的运动学关系方程，本节将根据 2.2 节中机器人具体的驱动轮布局，建立整体的运动学方程，并根据实际的设计尺寸得到确定的关系式。

在机器人实际运动控制时，并不需要关心瑞典轮轮毂边缘的自由辊子的运动情况，因此式 (2-6) 中与 V_{ri} 相关的项可以忽略。

最终得到通用的三轮全向移动机器人运动学表达式：

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \frac{-1}{r} \begin{bmatrix} \frac{\cos(\beta_1 - \gamma)}{\sin \gamma} & \frac{\sin(\beta_1 - \gamma)}{\sin \gamma} & \frac{l_1 \sin(\beta_1 - \gamma - \alpha_1)}{\sin \gamma} \\ \frac{\cos(\beta_2 - \gamma)}{\sin \gamma} & \frac{\sin(\beta_2 - \gamma)}{\sin \gamma} & \frac{l_2 \sin(\beta_2 - \gamma - \alpha_2)}{\sin \gamma} \\ \frac{\cos(\beta_3 - \gamma)}{\sin \gamma} & \frac{\sin(\beta_3 - \gamma)}{\sin \gamma} & \frac{l_3 \sin(\beta_3 - \gamma - \alpha_3)}{\sin \gamma} \end{bmatrix} \begin{bmatrix} V_X^R \\ V_Y^R \\ \omega \end{bmatrix} \quad (2-7)$$

而由 2.4.1 节可知：

$$\begin{bmatrix} V_X^R \\ V_Y^R \\ \omega \end{bmatrix} = \mathbf{R}(\theta) \begin{bmatrix} V_X^I \\ V_Y^I \\ \omega \end{bmatrix} \quad (2-8)$$

式中， V_X^I 、 V_Y^I 分别表示，机器人参考点在全局参考框架下沿 X_I 、 Y_I 两个坐标轴的速度分量。

至此，已经求得机器人的运动学模型。式（2-7）是通用的计算公式，在不考虑其他限制因素的情况下，可以应用于任何由 90° 偏执角的瑞典轮构成的机器人机构。

机器人各个驱动轮的相关参数如表 2.1 所示。

表 2.1 机器人驱动轮参数列表

	α_i	β_i	γ	l_i	r
	rad			mm	
1	$\frac{2\pi}{3}$	$\frac{2\pi}{3}$	$\frac{\pi}{2}$	350	76
2	$-\frac{2\pi}{3}$	$-\frac{2\pi}{3}$			
3	0	0			

将以上参数代入式（2-7）中，并结合式（2-8）得到如下的运动学方程。

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \frac{-1}{76} \begin{bmatrix} \frac{\sqrt{3}}{2} & \frac{1}{2} & -350 \\ -\frac{\sqrt{3}}{2} & -\frac{1}{2} & -350 \\ 0 & -1 & -350 \end{bmatrix} \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} V_X^I \\ V_Y^I \\ \omega \end{bmatrix} \quad (2-9)$$

在机器人实际的运动控制中，路径跟踪控制器在一个采样周期的输出是机器人在全局参考框架下的速度，通过式（2-9）计算每个驱动电机的转速，并向电机控制模块发送相应的指令实施控制。

2.5 本章小结

本章介绍了机器人系统设计指标，确定全向移动机器人呈正三角形的底盘驱动轮布局结构。设计用于机器人定位的随动轮系的机构，从机器人运行环境的角度说明了直线滑轨及压紧弹簧对减小定位误差的作用。根据机器人运行时所需的技术指标，确定相关的尺寸结构。本章第二部分针对机器人的结构设计，建立机器人的局部和全局参考框架。按照先局部后整体的路线，建立单个瑞典轮的通用运动学模型，然后将模型组合成机器人的整体运动学模型。确定具体的驱动轮及其在局部参考框架下的位姿参数，代入机器人整体模型，得到机器人的运动学控制方程。

第3章 全向移动机器人定位及路径跟踪算法研究

3.1 引言

导航是移动机器人研究的核心问题，而定位和路径跟踪又是其中的两个重要组成部分。定位的核心问题是传感器信息的采集与定位算法的设计；路径跟踪则强调机器人根据其相关的运动学或者动力学模型，使用一定的算法控制其执行体的输出，从而实现期望的运动轨迹。

本章主要论述三轮全向移动机器人定位和几何路径跟踪控制算法。首先，对机器人在二维平面运动场地上的定位数学模型进行详细推导，得到机器人实时位姿推算方程，并根据该方程进一步分析定位误差。第二部分主要分析机器人几何路径跟踪及姿态控制的设计原理以及跟踪误差校正控制器的设计。

3.2 机器人定位及路径跟踪整体方案分析

本文论述的机器人使用传统的里程计测量方式对机器人进行定位，同时结合机器人的运动学模型设计了基于几何路径跟踪的运动控制算法。机器人的导航整体方案如图 3.1 所示。

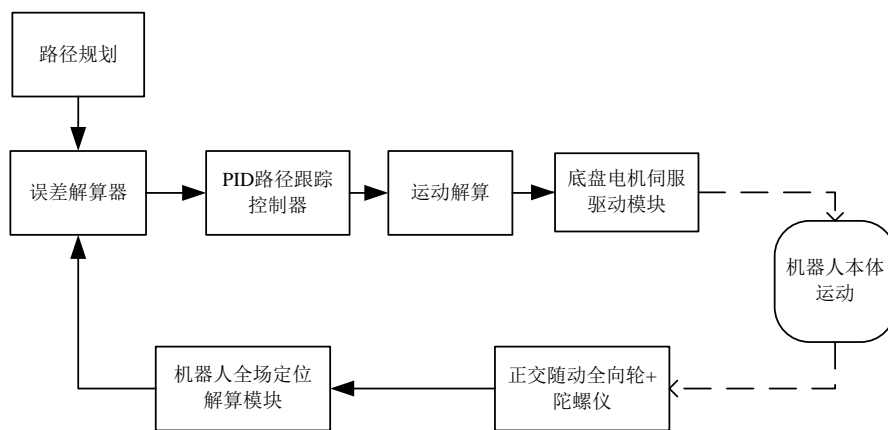


图 3.1 机器人整体控制流程

首先，机器人的路径规划是离线完成的，主要根据具体的任务以及机器人的运行环境设计基于直线和圆弧的机器人运动路线。机器人在接收到直线或者圆弧运动的命令后，开始以一定的基准速度向目标点运动，随着机器人本体的运动，机器人的定位模块读取用于定位的正交随动轮系和陀螺仪的数据，并处理得到机器人当前的位姿信息。误差解算模块通过比较机器人位姿与当前时刻理想路径下的位姿，得到基于 PID 的路径跟

踪控制器的误差输入。该控制器最终输出使机器人趋向理想路径的控制量，从而达到机器人实时路径跟踪的目的。

3.3 基于编码器和陀螺仪的机器人定位实现

机器人的定位目标：

在对机器人进行定位分析时，假设机器人的定位中心点是机器人局部参考框架下的原点。并且不考虑实际的各种误差，即在理想模型下进行分析。

由于本文介绍的全向移动机器人的运行环境是二维平面，所以机器人的位姿可以用以下的向量表示：

$$P_I = [X_I \quad Y_I \quad \theta]^T \quad (3-1)$$

其中 X_I 代表机器人在全局参考框架下的横坐标， Y_I 代表机器人在该框架下的纵坐标， θ 则表示机器人局部参考框架与全局参考框架的夹角。

同时，在机器人局部参考框架下，定义其局部位姿向量 P_R ：

$$P_R = [X_R \quad Y_R \quad \theta]^T \quad (3-2)$$

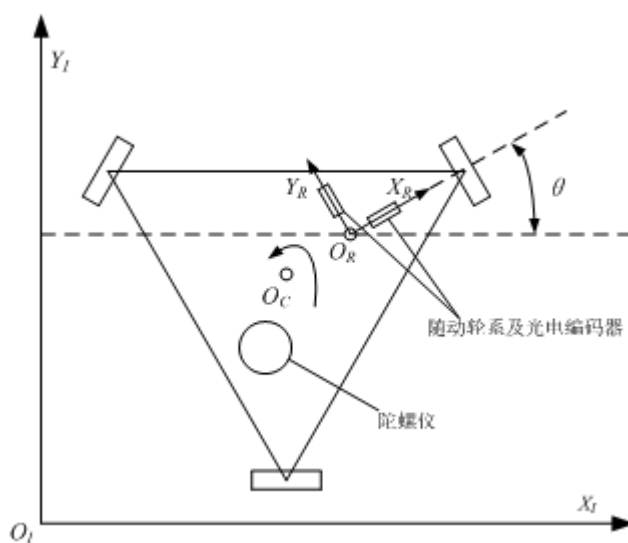


图 3.2 机器人的定位系统布局示意图

机器人使用小型的全向轮与增量式正交编码器组合的方式来构成机器人的里程计，由于全向轮的结构不同于普通的车轮，它在被动运动时只能够测量与轮轴垂直方向上的里程。机器人使用两个参数相同的里程计，并将它们正交布置，这样就构成了机器人坐标系的测量系统。同时，高精度陀螺仪反馈给机器人控制器的角度信息 θ 作为机器人局部参考框架与全局参考框架之间转换的桥梁。

定量地推导定位公式：

不考虑机器人上两个随动轮及正交编码器的差异,对于具有固定采样间隔 Δt 的离散系统,机器人在局部参考框架下的位姿增量 ΔP_R 为 $(\Delta X_R, \Delta Y_R, \Delta \theta)$ 。

机器人在全局参考框架下的位姿增量 $\Delta P_I(\Delta X_I, \Delta Y_I, \Delta \theta)$ 为：

$$\Delta P_I = \mathbf{R}(\theta)^{-1} \Delta P_R \quad (3-3)$$

式中, $\mathbf{R}(\theta)^{-1}$ 为机器人的定位信息由局部参考框架向全局参考框架下转换的正交旋转矩阵,它与 2.4 节中的 $\mathbf{R}(\theta)$ 互为逆阵。假设 t 时刻机器人更新过的位姿为 ΔP_{I_t} , 我们可以得到机器人在全局参考框架下的位姿更新公式：

$$P_{I(t+\Delta t)} = \Delta P_{I(t)} + \mathbf{R}(\theta)^{-1} \Delta P_R \quad (3-4)$$

根据式 (3-4) 中的关系, 我们进一步得到基于里程计和陀螺仪的全向移动机器人位姿更新基本方程：

$$P_{I(t+\Delta t)} = f(X_{I(t)}, Y_{I(t)}, \theta_t, \Delta X_I, \Delta Y_I, \Delta \theta) = \begin{bmatrix} X_{I(t)} \\ Y_{I(t)} \\ \theta_t \end{bmatrix} + \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta X_R \\ \Delta Y_R \\ \Delta \theta \end{bmatrix} \quad (3-5)$$

在机器人的控制系统实现中, 周期性的对里程计和陀螺仪积分模块采样, 并根据上式计算即可得到机器人采样时刻的位姿。

上述情况默认机器人局部参考框架与机器人的几何中心完全重合。根据前面运动学相关章节的分析, 机器人在定位时定位中心是机器人的几何中心。同样是加工及装配等原因, 导致机器人局部参考框架原点与机器人几何中心必然存在一定的距离偏差或者由于机器人装配的需要, 在设计时就存在一定的距离, 见示意图, 但是根据运动学模型的需要, 定位及运动控制时都是以机器人的几何中心为参考点的, 所以该距离需要实际测。

假设距离为 l_{RC} , 使用如下的方式进行校正：

机器人的运动可以认为是刚体的平面运动。由刚体的平面运动学知识可知, 任意时刻刚体上一点的速度可以看做是刚体平动和定轴转动的矢量合成。

由 2.3 节可知, 机器人在运动时, 机器人底盘结构上任意一点的瞬时转动中心是机器人的几何中心 O_C , 假设该点在机器人全局参考框架下的速度为 $\vec{V}_C^I (V_{CX}^I, V_{CY}^I)$ 。 \vec{V}_C^I 与机器人局部参考框架原点 O_R 在全局参考框架下的速度 $\vec{V}_R^I (V_{RX}^I, V_{RY}^I)$ 的关系可以表达为：

$$\vec{V}_R^I = \vec{V}_C^I + \vec{V}_{RC}^I \quad (3-6)$$

式中 \vec{V}_{RC}^I 为点 O_R 绕基点 O_C 的速度。

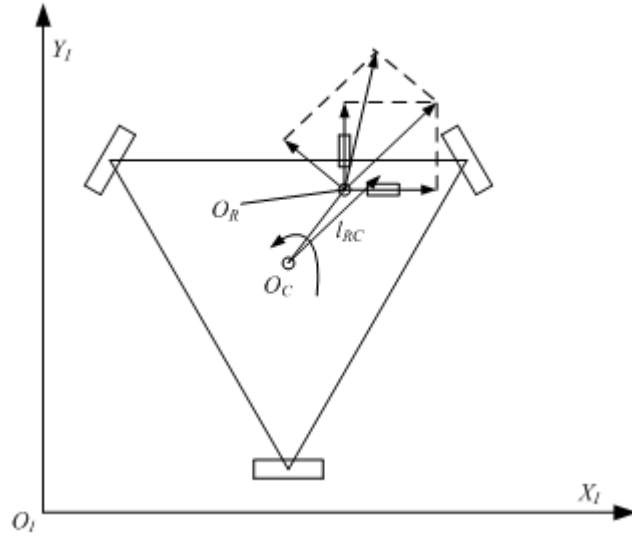


图 3.3 机器人定位中心与几何中心速度关系

对式 (3-6) 进行矢量分解运算可得

$$\begin{cases} V_{RX}^I = V_{CX}^I - \omega l_{RC} \sin \theta \\ V_{RY}^I = V_{CY}^I + \omega l_{RC} \cos \theta \end{cases} \quad (3-7)$$

式中, ω 为机器人自转角速度, 可由陀螺仪的数据计算得到。 l_{RC} 需要一定样机实验来确定, 实验的方法及结果将在第五章中介绍。

对式 (3-7) 中 V_{CX}^I, V_{CY}^I 反解可得

$$\begin{cases} V_{CX}^I = V_{RX}^I + \omega l_{RC} \sin \theta \\ V_{CY}^I = V_{RY}^I - \omega l_{RC} \cos \theta \end{cases} \quad (3-8)$$

机器人控制系统中, 局部参考框架原点 O_R 在全局参考框架下的速度 $(V_{RX}^I, V_{RY}^I, \omega)$ 可由采样周期内的位姿增量 ΔP_I 代替。

通过以上的分析及算式推导, 最终得到了机器人几何中心的实时位姿解算公式。

3.4 机器人定位误差分析

3.4.1 机器人定位误差来源分析

1、传感器误差

由 3.2 节可知, 全向移动机器人的定位实现依赖于传感器的数据。所以, 机器人的定位误差很大程度上来源于里程计和陀螺仪的误差。

1) 里程计由光电编码器和随动瑞典轮构成, 由于编码器固有精度有限、瑞典结构的特殊性以及机器人运动表面的摩擦力有限等因素, 在运动期间, 编码器常常会出现“丢

数”现象，因此里程计测得的里程信息会比真实的距离小。施加在滑道（小轮）上的正压力、机器人的运行速度（就是小轮的旋转速度）以及小轮与地面的摩擦系数是引起里程计误差的主要原因。

2) 陀螺仪是机器人能够实现全方位移动并实时定位的关键传感器，如式（3-5）所示，它输出的数据（机器人的姿态）决定着机器人定位模型的准确与否。由于陀螺仪的数据采集需要模数转换电路实现，因此，除了自身的技术指标，陀螺仪的误差主要与相关 AD 转换芯片、基准电压和系统电源的精度有关。

2、随动轮系结构误差

图 3.2 所示的机器人定位模型中，假设两个随动轮运动方向的夹角是 90° ，但是由于加工及装配精度有限，正交随动轮系的两个小全向轮运动方向不是严格相互垂直，如图 3.5 所示。而在机器人控制系统的实现中使用的是理想模型，所以实际的模型与程序的理想模型不符，最终会造成机器人定位上的误差。

3、机器人初始位姿误差

机器人定位最终得到的是定位中心在机器人全局参考框架下的位姿，而 3.2 节中的定位方法是在机器人初始位姿的基础上累加实现的，所以机器人定位的初始位姿误差将一直存在于整个定位过程中，如图 3.4 所示。

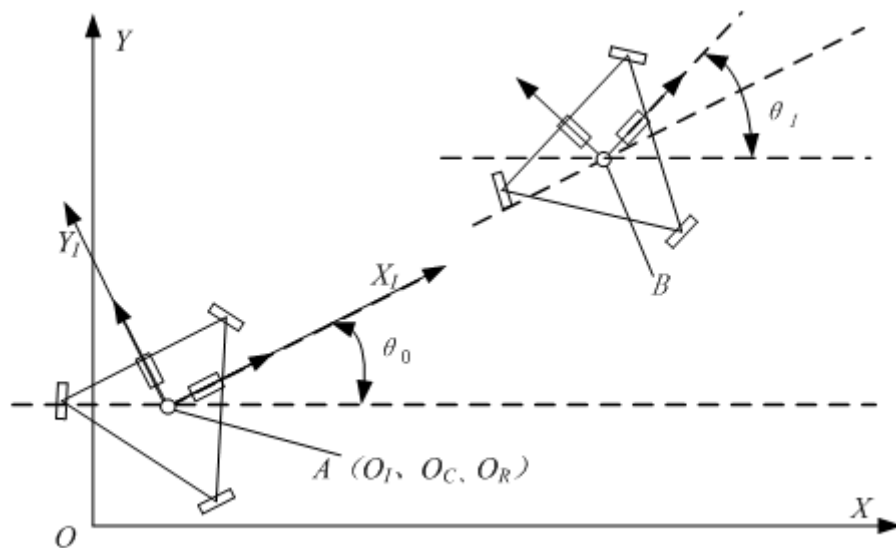


图 3.4 机器人初始位姿对定位的影响示意图

为了凸显初始位姿误差对机器人实际定位误差的影响，上图忽略前文所述的传感器误差和随动轮系结构误差。

上图在定位参考框架的基础上，添加了机器人实际运行的真实参考框架 XOY 。图

中表现了由初始点 A 到终点 B 时机器人在各个参考框架下的位姿关系。

(X_0, Y_0, θ_0) —机器人在真实参考框架 XOY 下的初始位姿

(X_1, Y_1, θ_1) —机器人在真实参考框架 XOY 下的终点位姿

(X_0', Y_0', θ_0') —机器人在理想的全局定位参考框架 $X_I O_I Y_I$ 下的初始位姿

(X_1', Y_1', θ_1') —机器人在理想的全局定位参考框架 $X_I O_I Y_I$ 下的终点位姿

上述位姿表达, 满足以下关系

$$\begin{bmatrix} X_1 \\ Y_1 \\ \theta_1 \end{bmatrix} = \begin{bmatrix} X_0 \\ Y_0 \\ \theta_0 \end{bmatrix} + \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta \theta \end{bmatrix} \quad (3-9)$$

式中, $[\Delta X \quad \Delta Y \quad \Delta \theta]^T$ 表示机器人在真实参考框架下的真实位姿变化, 可以由下式表达:

$$\begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta \theta \end{bmatrix} = \begin{bmatrix} \cos \theta_0 & -\sin \theta_0 & 0 \\ \sin \theta_0 & \cos \theta_0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_1' - X_0' \\ Y_1' - Y_0' \\ \theta_1 - \theta_0 \end{bmatrix} \quad (3-10)$$

将上式代入式 (3-9) 可得

$$\begin{bmatrix} X_1 \\ Y_1 \\ \theta_1 \end{bmatrix} = \begin{bmatrix} X_0 \\ Y_0 \\ \theta_0 \end{bmatrix} + \begin{bmatrix} \cos \theta_0 & -\sin \theta_0 & 0 \\ \sin \theta_0 & \cos \theta_0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_1' - X_0' \\ Y_1' - Y_0' \\ \theta_1 - \theta_0 \end{bmatrix} \quad (3-11)$$

由上式可以看出, 当机器人运行的路程较长时, 初始的姿态偏差将对最终的定位结果产生较明显的影响。

3.4.2 机器人定位误差的校正

基于前一节的分析, 有必要建立机器人定位的误差模型。

3.2 节中, 对机器人里程计的实际模型进行了简化, 由于基于里程表和陀螺仪的位置更新的一个比较粗略的估计, 所以有必要建立整体的机器人位姿估计误差模型, 得到位姿估计的协方差矩阵 $\Sigma_{p_{t+\Delta t}}$, 进而对机器人的误差分布情况进行分析。

机器人位姿估计方程 f 的微分方程可由一阶泰勒线性化公式近似得到, 根据传感器的误差传播理论, 可以得到以下结论:

$$\Sigma_{P_{I(t+\Delta t)}} = \nabla(P_{I(t)})f \Sigma_{P_{I(t)}} \nabla(P_{I(t)})f^T + \nabla(\Delta_{X_I Y_I \theta})f \Sigma_{\Delta_{X_I Y_I \theta}} \nabla(\Delta_{X_I Y_I \theta})f^T \quad (3-12)$$

我们假定在初始时刻, 协方差矩阵为 0, 而对于运动增量 $(\Delta X, \Delta Y, \Delta \theta)$, 假设其协

方差矩阵为：

$$\Sigma_{\Delta_{XY\theta}} = \text{covar}(\Delta X, \Delta Y, \Delta \theta) = \begin{bmatrix} k_x |\Delta X| & 0 & 0 \\ 0 & k_y |\Delta Y| & 0 \\ 0 & 0 & k_\theta |\Delta \theta| \end{bmatrix} \quad (3-13)$$

上式中， k_x 、 k_y 、 k_θ 是传感器的误差常数，其中 k_x 、 k_y 代表了与被动编码器连接的小全向轮和地面交互的非确定性参数，它们用于补偿由于轮子变形、轮子滑动、机器人运动表面不平、全向轮本身结构等原因造成的不精确运动； k_θ 则代表陀螺仪积分模块在机器人运动期间的非确定性运动参数，它主要用于补偿机器人振动以及陀螺仪随机性积分误差。在上式中我们做了如下的假设：

- 1) 三个传感器的误差是相互独立的；
- 2) 各个传感器误差的方差与其变化量的绝对值成正比。

根据方程 (3-8) 可以计算式中的两个雅可比矩阵：

$$F(p) = \nabla(p)f = \begin{bmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial y} & \frac{\partial f}{\partial \theta} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3-14)$$

$$F(\Delta_{XY\theta}) = \nabla(\Delta_{XY\theta})f = \begin{bmatrix} \frac{\partial f}{\partial(\Delta X)} & \frac{\partial f}{\partial(\Delta Y)} & \frac{\partial f}{\partial(\Delta \theta)} \end{bmatrix} = \begin{bmatrix} \cos(\theta + \alpha) & -\sin(\theta + \alpha) & 0 \\ \sin(\theta + \alpha) & \cos(\theta + \alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3-15)$$

将以上两式代入式 (3-12) 即可得到机器人位姿估计的协方差矩阵，进而针对定位误差的模型提出以下几个方面的误差校正方法。

1、编码器本身的误差校正

在完成基本的导航功能后，给定机器人由初始位姿 (X_0^I, Y_0^I, θ_0) 到运动目标位姿 (X_1^I, Y_1^I, θ_1) 的路径，即沿 X_I 轴方向上运动距离 L 满足：

$$L = |X_1^I - X_0^I| \quad (3-16)$$

将机器人任意摆放到实验场地上，只需保证机器人在运动方向上有足够的空间。通过对机器人的定位模型分析可知，机器人默认的情况下，机器人全局参考框架和机器人局部参考框架是完全重合的，在机器人自己的定位系统下，就是机器人局部参考框架下 Y_I 轴方向上编码器记录的里程信息，即 5000mm。测量机器人在执行完上述路径后实际走过的距离即可对机器人的里程计误差进行校正。

需要说明的是，由于随动轮系的结构存在偏差，机器人控制系统的实现中使用

的是理想模型，所以机器人在运动控制过程中会沿垂直于实际随动轮系 Y 轴的方向运动，实际的运行路线如图所示。

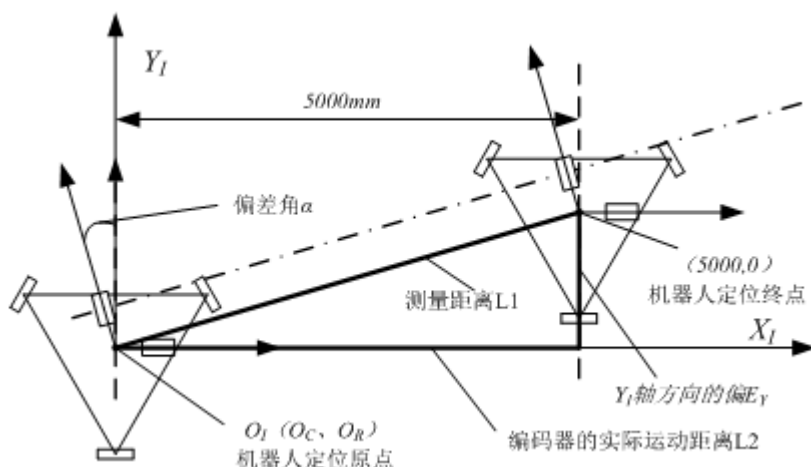


图 3.5 机器人里程计误差校正示意图

在机器人的样机制作中，随动轮系的固定框架使用高精度的数控铣床加工，因此轮系的偏差角主要来源于各个装配环节。因此，可以保证偏差角 α 在 $\pm 1^\circ$ 以内。

假设实验的测量参考点为机器人的几何中心，一次实验中测量距离为 $L1$ ，编码器的实际运动距离为 $L2$ 。则由图中的几何关系可知

$$\cos \alpha = \frac{L2}{L1} \geq 0.9998$$

因此在测试距离足够长的测试中可以将 $L1$ 的值用于编码器的校正。经过多次试验后取 $L1$ 平均值作为实际测量值， k_x 可按下式计算：

$$k_x = \frac{\sum_{i=1}^N L1_i}{N(X'_1 - X'_0)} \quad (3-17)$$

式中：

N —表示试验次数

$L1_i$ —表示第 i 次试验结果

使用同样的方法校正机器人局部参考框架下的 Y_R 轴方向的里程计误差，得到非确定性参数 k_y

2、机器人随动轮系角度偏差校正

前文已经论述，随动轮系两个轴存在角度偏差。尽管在进行编码器累计误差校正时忽略了该误差，由图 3.5 所示，当机器人的运行距离足够长时，机器人在图示误差模型的 X_R 轴方向上的最大偏差 $E_y \approx L1 * \alpha = 5000\text{mm} * 1^\circ = 87.27\text{mm}$ 。显然，这个误差值在实

际的机器人实验中是不可忽略的。

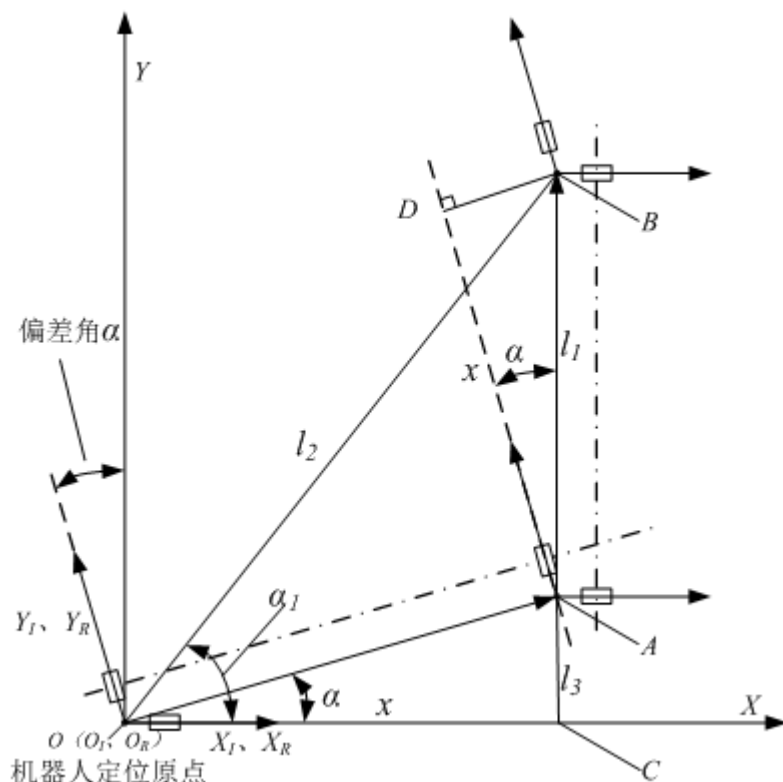


图 3.6 机器人正交随动轮系角度偏差校正示意图

机器人随动轮系偏差角 α 的校正可以按照如下的方法进行测量和校正。图 3.6 描述了机器人在真实的全局参考框架下沿直线 OB 运动的情况。偏差角的测量过程如下：

a) 给定机器人一条直线路径 OB ，初始点和 $B(x, x, 0)$ 终点分别为 $O(0, 0, 0)$ 和 $B(x, x, 0)$ 。由于机器人的定位系统使用的是理想模型，所以机器人局部参考框架 $X_R O_R Y_R$ 和全局参考框架 $X_I O_I Y_I$ 是完全重合的。但是由于随动轮系的角度偏差，导致前述的两个参考框架和真实的机器人全局参考框架 XOY 不重合。图 3.6 中，假定 $O_I X_I$ 轴与 OX 轴重合。

b) 为了到达图 3.6 所示的 B 点, 机器人先由 O 点运动到 A 点, 即认为机器人首先按照自身定位系统的指引完成 OX 轴方向上的运动。由于偏差角的存在, 机器人实际的轨迹如上图所示, 所以 A 点在 OX 轴上的投影 C 到 O 点的距离为期望的 OX 轴运动距离 x 。机器人到达 A 点以后, 定位系统的 Y_i 值仍为 0。

c) 接下来, 机器人由 A 点运动到 B 点。由于上一步已经完成了 OX 轴方向上的运动, 所以终点 B 与 A 点在 OX 轴上的投影点均为 C 点。该过程中, 机器人自身的定位系统中 Y_l 的增量为 x , 即线段 AB 在 $O_l Y_l$ 轴上的投影距离 AD 。

图 3.6 中:

α_1 —为直线 OB 与 OX 轴的夹角;

l_1 —为由 A 点运动至 B 点时，机器人中心在 OY 方向上的增量；

l_2 —为机器人由初始点到终点的测量距离；

l_3 —为由 O 点运动至 A 点时，机器人中心在 OY 方向上的增量；

由图 3.6 中的几何关系可知，

$$l_3 = x \cdot \tan \alpha \quad (3-18)$$

$$l_1 = x / \cos \alpha \quad (3-19)$$

$$l_2 = \sqrt{x^2 + (l_1 + l_3)^2} = x \sqrt{1 + \left(\frac{1 + \sin \alpha}{\cos \alpha}\right)^2} \quad (3-20)$$

$$\cos \alpha_1 = \frac{x}{l_2} = 1 / \sqrt{1 + \left(\frac{1 + \sin \alpha}{\cos \alpha}\right)^2} \quad (3-21)$$

进一步简化式 (3-21) 可得：

$$\cos \alpha_1 = \sqrt{\frac{1 - \sin \alpha}{2}} \quad (3-22)$$

由上式得到 α_1 与 α 的关系：

$$\cos 2\alpha_1 = \cos\left(\frac{\pi}{2} + \alpha\right) \quad (3-23)$$

图 3.6 中假设机器人随动轮系的夹角比 90° 要大，即实际的夹角为 $(90^\circ + \alpha)$ 。所以 α_1 为机器人随动轮系实际夹角的一半。所以有：

$$\alpha_1 = \left(\frac{\pi}{2} + \alpha\right) / 2 \quad (3-24)$$

机器人随动轮系的夹角小于 90° 的情况，可以得到与式 (3-24) 相似的结果：

$$\alpha_1 = \left(\frac{\pi}{2} - \alpha\right) / 2 \quad (3-25)$$

上述结论推导的过程与前文论述类似，这里不再详细说明。

在完成前文中里程计相关的校正后，图 3.6 中 x 即为实验中给定的已知量，则通过实际测量线段 OB 的距离 l_2 ，就可以结合式 (3-23) 求得角度 α_1 的值，通过式 (3-24) 求得随动轮系的偏角 α 。

按照图 3.6 所示的分析模型，认为在任何情况下 OX 轴方向上的里程计不存在偏差角，而仅仅是 OY 方向上存在。在考虑偏差角 α 后，3.3 节中机器人在局部参考框架下的位姿增量 ΔP_R 由原来的 $(\Delta X_R, \Delta Y_R, \Delta \theta)$ 修正为 $(\Delta X_R, \Delta Y_R / \cos \alpha, \Delta \theta)$ ，

3、机器人初始位姿误差的校正

机器人初始位姿误差的测量与校正需要依据具体的机器人实验场地实现。由 3.4.1 小节中的式 (3-11) 可知, 测量机器人在真实参考框架 XOY 下的初始位姿 (X_0, Y_0, θ_0) 是校正的根本。

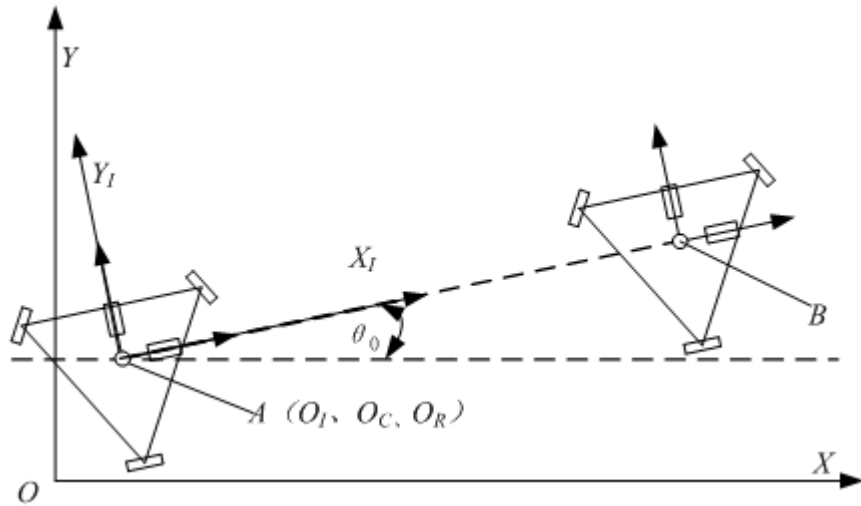


图 3.7 机器人初始位姿误差校正示意图

机器人在真实的参考框架下初始位姿的测量示意图, 如图 3.7 所示。图中, 假定已经完成了其他机器人定位相关的误差校正。给定机器人沿 $O_I X_I$ 轴方向的一段路径 AB , 其中 A 、 B 两点在 $X_I O_I Y_I$ 的位姿分别为 $(0,0,0)$ 、 $(x,0,0)$ 。 A 、 B 两点在 XOY 下的位置 (X_A, Y_A) 以及 (X_B, Y_B) 可以通过实际测量的方式得到, 根据图 3.7 中几何关系, 可以得到机器人初始姿态角的计算公式:

$$\theta_0 = \arctan \frac{Y_B - Y_A}{X_B - X_A} \quad (3-26)$$

通过上述的方式测得机器人的初始位姿后, 使用式 (3-11) 就可以实现机器人的位姿误差校正。

3.5 机器人几何路径跟踪及姿态控制

前面章节论述了机器人在二维运动平面的定位, 在不考虑机器人自身定位结果与实际位置误差的前提下, 机器人控制系统内部已经有一张二维的地图。本文论述的机器人将介绍笛卡尔坐标系下几何路径跟踪控制器。

不考虑机器人实际运动过程中的速度限制, 三轮全方位移动机器人是一个完整系统, 它的位置 (X_I, Y_I) 和姿态之间不存在不可以解耦的关系。由运动学方程 (2-8) 可以

看出，机器人的平移速度(V_x^I 、 V_y^I)和自转速度 ω 是可以独立控制的。基于运动学的机器人控制器输出实际上是这个速度控制器输出的矢量和。因此，分别对机器人的位置和姿态控制设计相应的控制器。

本节论述的路径跟踪是建立在理想的机器人定位与运动学模型之上的。同时，机器人的路径跟踪表示控制机器人位置（机器人中心在全局参考框架下的坐标： X_I 、 Y_I ）沿规则的几何路线变化的过程。姿态控制表示机器人在路径跟踪的同时，机器人保持某一姿态或者按照一定的规律变化到期望姿态的过程。

3.5.1 机器人几何路径跟踪

1、直线路径跟踪

在二维平面坐标系下，两点可以确定一条直线。假设机器人的一条直线运动路径起始点 A 和终止点 B 在机器人全局参考框架下的坐标分别为 (x_0, y_0) 、 (x_1, y_1) ，则机器人的运行路线可由以下方程来表达：

$$\begin{cases} (y - y_0) = \frac{y_1 - y_0}{x_1 - x_0} (x - x_0); (x_1 \neq x_0) \\ x = x_0; (x_1 = x_0) \end{cases} \quad (3-27)$$

理想情况下，机器人的运动速度方向与矢量 \overrightarrow{AB} 的方向相同，所以只需要给定机器人的初始运动矢量，根据 2.4 节中的式（2-9）即可驱动机器人向目标点运动。

但是，机器人在运动过程中一旦受到外力的作用，偏离运动轨迹，最终不能准确到达终点。通过一种简单的动态规划方法，就可解决上述的问题。因为，机器人能够在每个控制周期内更新位置，所以可以用每次更新后的位置 (x_t, y_t) 代替式（3-27）中的初始位置 (x_0, y_0) ，得到新的目标运动路线：

$$\begin{cases} (y - y_t) = \frac{y_1 - y_t}{x_1 - x_t} (x - x_t); (x_1 \neq x_t) \\ x = x_t; (x_1 = x_t) \end{cases} \quad (3-28)$$

这种控制方式下，即使机器人在运动期间受到一定程度的干扰，只要定位模块正常工作，也能在接下来的控制周期里通过式（3-28）将机器人的运动方向校正过来。需要说明的是，虽然在控制的时候使用了不同的直线表达式，但是机器人运动的期望路径仍然是直线 AB，而式（3-28）只是机器人运动控制的一种方式。

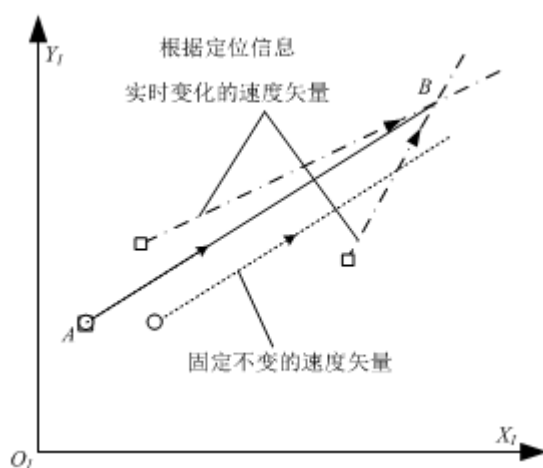


图 3.8 机器人直线路径跟踪示意图

2、圆弧路径跟踪

机器人圆弧路径主要用于不同运动方向的直线路径之间的过渡。同时，在一些特定的运行环境下，机器人的运动路径常常要求是参数已知的弧线。

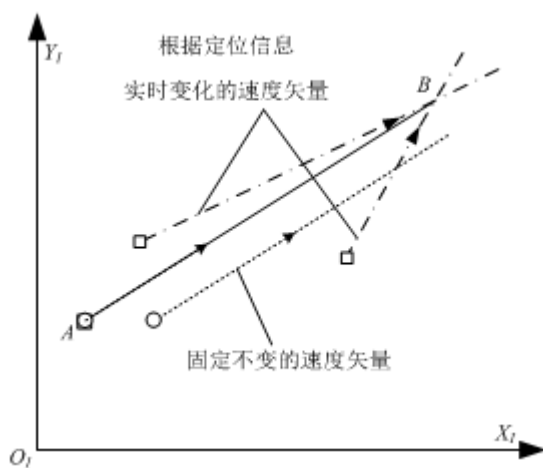


图 3.9 机器人圆弧路径跟踪示意图

假设机器人离线规划的圆弧路径的主要参数如下：

圆心： $O(x_c, y_c)$ ；

半径： R ；

起点： (x_0, y_0) ；

终点： (x_1, y_1) 。

由以上参数可得到圆弧的表达式：

$$\begin{cases} x = x_c + R \cos(\omega t) \\ y = y_c + R \sin(\omega t) \end{cases} \quad (3-29)$$

式中， ω 为预设的机器人沿圆弧路线运动的角速度。与直线路径不同的是，机器人在运行圆弧路径时，控制系统需要不断地计算当前控制时刻机器人的运动速度矢量。假设在机器人控制时刻 t ，则机器人的控制输出为方程（3-29）的导数：

$$\begin{cases} \dot{x} = -\omega R \sin(\omega t) \\ \dot{y} = \omega R \cos(\omega t) \end{cases} \quad (3-30)$$

3、直线到圆弧的过渡分析

基于运动学的路径跟踪控制器，没有考虑机器人在实际运动时驱动轮与地面的摩擦力的限制。实际上，当机器人的速度足够大时，地面不能够为机器人提供足够大的摩擦力而导致打滑。其中，机器人由直线路径向圆弧路径切换时，由于机器人做圆周运动需要提供向心力，机器人的受力情况会发生突变，从而导致机器人偏离预定轨迹。

为了提高机器人运动轨迹的平滑性，将机器人由直线到圆弧的直过渡改为曲率均匀变化的过渡。

3.5.2 机器人的姿态控制

机器人的姿态控制主要包括姿态要求有变化和保持不变两种情况。对于姿态不变的情况，控制器的输出只需要满足 $w=0$ 这一条件，即机器人只做平移运动。而对于有姿态角变化的情况，采取使机器人姿态均匀变化的驱动方式。

假设在某一段机器人的运行路径中，机器人起点和终点的姿态角分别为 θ_0 和 θ_1 ，机器人运行的速度为 V_t ，由3.4.1节可知，机器人的路径都能够在二维的笛卡尔坐标系下进行准确的解析式表达。驱动机器人姿态均匀变化有以下两种方式。

1、根据机器人路径的相关信息计算该段路径的距离 S ，以 $w=(\theta_1-\theta_0)\times V_t \div S$ 驱动机器人。

上述的方式理论上能够驱动机器人达到预定的姿态，但是在机器人运行时不可避免地要受到外界的干扰，此时机器人的姿态可能发生改变。因为机器人转动的角速度恒定，这最终有可能造成机器人运动到终点时与预期的值不同。

2、在控制时刻 t ，结合机器人的定位信息和机器人当前路径的解析表达式，计算出机器人剩余路径的长度 ΔS_t 和姿态角剩余变化量 $\Delta \theta_t$ 。此时，以 $w_t = \Delta \theta_t \times V_t \div \Delta S_t$ 驱动机器人。

第二种方式能够实时地计算当前机器人合适的角速度，而不是保持初始的计算恒定。这种方式通过机器人的样机实验，相关的实验结果将在第五章中详细介绍。

3.6 基于 PID 的机器人位姿跟踪误差校正

3.6.1 机器人几何路径跟踪的误差计算

1、直线路径的跟踪误差

机器人直线跟踪位置误差定义为机器人当前的定位位置 (x_t, y_t) 到理想运动直线 AB 的距离。根据理想直线路径的表达式 (3-28) 以及机器人的定位信息，可以计算误差。

$$e_t = \begin{cases} \frac{\left| \frac{(y_0 - y_1)x_t}{x_1 - x_0} + y_t + \frac{y_1x_0 - y_0x_1}{x_1 - x_0} \right|}{\sqrt{\left(\frac{y_0 - y_1}{x_1 - x_0}\right)^2 + 1}}; (x_1 \neq x_0) \\ x_t - x_0; (x_1 = x_0) \end{cases} \quad (3-31)$$

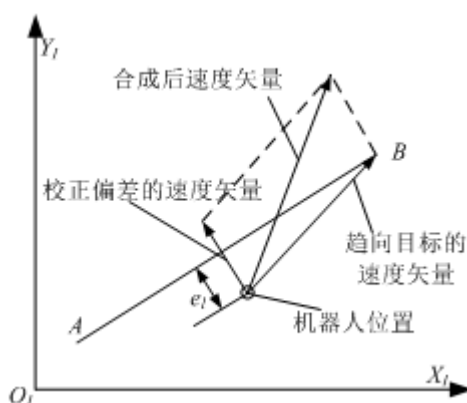


图 3.10 直线误差校正示意图

2、圆弧路径的跟踪误差

而对于圆弧路径，机器人跟踪的位置误差定义为机器人当前的定位位置 (x_t, y_t) 到理想运动圆弧圆心 $O(x_o, y_o)$ 的距离与半径 R 的差值。

$$e_c = \sqrt{(x_t - x_o)^2 + (y_t - y_o)^2} - R \quad (3-32)$$

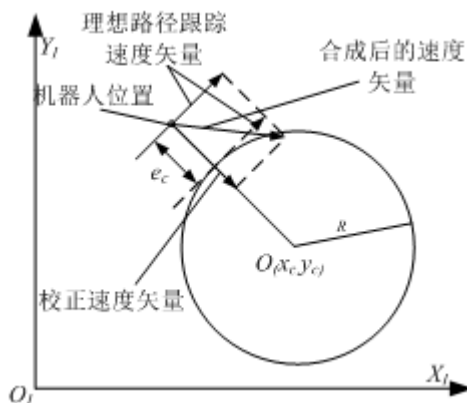


图 3.11 圆弧误差校正示意图

3、机器人的姿态偏差

根据前文的说明，姿态的误差控制主要用于其运动过程中姿态角要求恒定的情况。此时，偏差就是当前机器人的姿态角与期望值的差值，如式（3-33）所示。

$$e_a = \theta_t - \theta_0 \quad (3-33)$$

机器人理想的控制目标是实际的位姿误差为零。但是，由于特殊的驱动轮结构等原因，机器人在实际运过程中无法做到。实际上，如果将预期的姿态误差限设为零，控制器的输出将无法稳定，从而造成机器人的自旋震荡。为此，在满足机器人控制指标的前提下，设定机器人的姿态误差容限。当姿态误差在容限范围内时，将误差设置为零，从而保证了姿态控制器的快速稳定。

3.6.2 机器人路径校正的 PID 控制器

PID（Proportion Integration Differentiation）是广泛应用的控制算法^[36-39]。PID 控制器的基本控制方程为：

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + \frac{T_d de(t)}{dt} \right] \quad (3-34)$$

式中：

K_p, T_i, T_d ——表示比例、积分、微分作用参数；

$e(t), u(t)$ ——分别表示控制器输入偏差、控制器输出；

上式也称为位置式 PID。本课题主要使用增量式 PID 对机器人的位姿校正，其表达如式（3-35）所示：

$$u(k) = u(k-1) + K_p \left\{ e(k) - e(k+\frac{T}{T_i}) + e(k-\frac{T}{T_i}) - e(k-1) + \frac{T}{T_i} e(k-1) \right\} \quad (3-35)$$

上式中：

T ——表示离散系统的采样周期；

$e(k), u(k)$ ——分别表示控制器第 k 次采样时刻输入的误差和控制器的输出。

前面章节介绍了理想情况下机器人基于运动学的直线、圆弧以及过渡区的跟踪方法。但是在实际的运行中，机器人不可能按照预定的轨迹运动，主要原因有以下几点：

- 1) 机器人的运动学模型与实际的驱动轮安装配置参数存在误差
- 2) 机器人受速度和加速度的限制

其中,前者造成机器人实际的运动学模型与理想模型存在差异,从而机器人的实际运动速度与控制器的输出不同。后者主要影响机器人在不同曲率路径的接合区产生的轨迹不平滑现象。为了使机器人能够正确的跟踪预定几何路线,我们在3.4节跟踪控制的基础上添加了基于位置和姿态的跟踪误差实时校正环节,通过使用结构简单、鲁棒性强的经典PID控制算法,能够有效地控制机器人的跟踪误差。PID位姿跟踪误差校正控制器的结构如图3.12所示。

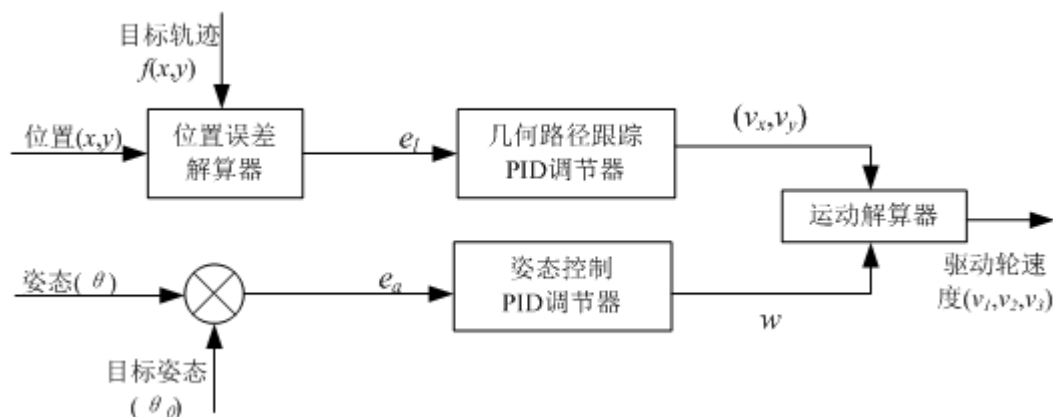


图 3.12 PID 位姿控制器

图中,可以看出控制系统使用两个独立的PID控制器分别对机器人的位置(X_t, Y_t)和姿态 θ 控制,主要基于以下几点考虑:

- 1) 机器人的位置和姿态在一定条件下可以完全解耦(见2.4节);
- 2) 机器人的位置和姿态的相关控制方法不同,若使用同一套PID参数,不易同时满足两方面的控制要求。

3.7 本章小结

本章设计实现了基于瑞典轮和里程计的正交随动轮系,并结合陀螺仪通过航位推算的方法得出了机器人的位姿更新方程。根据误差传播的相关理论建立机器人定位系统的误差传播模型,为机器人定位误差的校正打下基础。运动控制方面主要是论述机器人对几何路径跟踪的控制方式。在机器人全场定位的基础上,设计实现了基于解析法的机器人几何路径跟踪算法。分析了直线和圆弧路径的路径跟踪误差实时解算方法,基于PID设计了误差校正控制器。

第4章 全向移动机器人嵌入式控制系统设计与实现

4.1 引言

本章主要介绍全向移动机器人的嵌入式控制系统设计。由于控制系统涉及多种传感器的数据采集、直流电机的伺服控制、定位及运动控制算法的实现；同时第三章中论述的机器人定位及路径跟踪算法需要在嵌入式控制系统中离散化实现。这就要求机器人的控制系统在核心算法的实现上有较高的实时性，即能够在确定的时间内完成相应的控制任务；同时控制周期尽可能短，才能保证定位及路径跟踪算法具有很好的实现效果。

本章首先介绍控制系统的总体设计，然后分别介绍各个子控制系统的设计原理及相关的软硬件实现。由于机器人控制系统的完整实现需要处理很多复杂的任务，而系统又要求较高的实时性，因此有必要使用嵌入式实时操作系统。

4.2 控制系统总体结构

控制系统的硬件整体结构如图4.1所示。需要说明的是该控制系统仅仅包含了课题研究的定位及路径跟踪算法所需的基本电路。

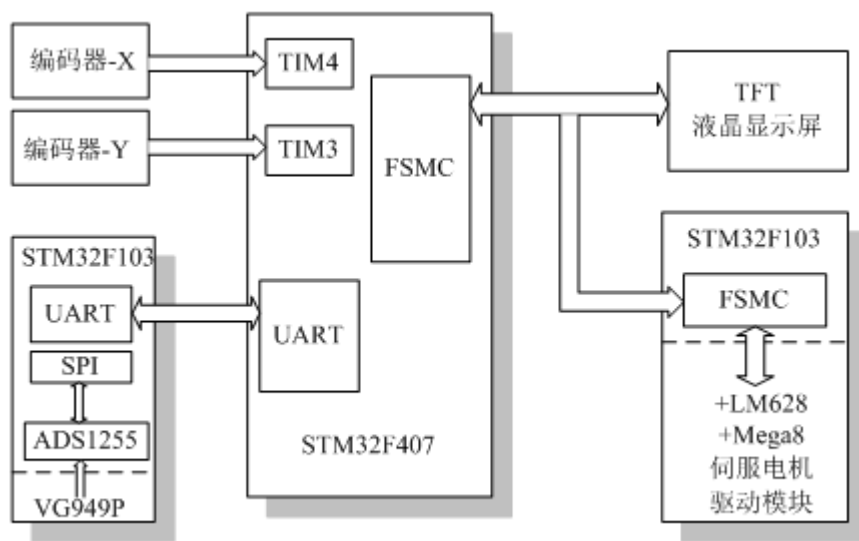


图4.1 控制系统总体结构

根据第三章机器人定位以及路径跟踪算法的论述，该控制系统实现了机器人定位所需的增量式正交编码器和陀螺仪的数据采集功能，同时机器人驱动电机的速度伺服控制子系统。机器人使用STM32作为其主控制器。机器人串行方式的通信使用了常用的UART、CAN总线、SPI等，并行方式主要使用双端口RAM来实现两个微处理器之间的高速通信。

4.3 机器人主控芯片

机器人主控芯片选用 STM32F407ZE。它的最高工作频率高达 168MHz，内置 512K 字节程序及数据存储和 192K 系统内存。处理器具有丰富的外设接口，包括高达上百个输入输出，8 个通用和高级定时器，两个 I²C、SPI、CAN 等常用通信接口，具有 12 位精度的数模、模数转换器，同时还具有丰富的 DMA 外设通道，能够有效地提高系统外设之间的通信的效率^[53]。

除了上述的特性之外，选择该处理器还有如下的原因：

1) 芯片使用 ARM 在工控领域领先的 CM4 内核处理器，具有更高的指令执行效率。同时中断的响应能力更高，适合实时、任务复杂的系统。

2) STM32F407ZE 内部具有 FPU 协处理器，能够实现 DSP 运算功能，能够实现机器人导航相关的复杂运算。

3) 处理器的相关存储配置能够很好地配合使用嵌入式操作系统，便于机器人控制算法、任务执行等模块的软件实现。

4) 另外，ST 系列的处理器具有强大的开发环境支持，包括完善的外设库文件及使用例程，活跃的社区讨论等。因此，系统的软件开发相对容易。

4.4 控制系统通信设计

4.4.1 基于双端口 RAM 的并行通信

1、STM32 扩展存储器接口

STM32 具有高效的外扩存储设备控制器—FSMC，它提供多种存储设备的总线接口，其中，能够同时支持四块最大 64M 的 SRAM 存储设备的接入。FSMC 支持 8 或者 16 位的数据总线，同时具有时序可编程等灵活多样的设置功能，满足大部分嵌入式应用对扩展存储器的要求^[53]。图 4.2 描述了 STM32 系列控制器使用 FSMC 与双端口 RAM 的接口。图中 0x6000000 为第一个扩展 SRAM 存储器在内核总线上的映射基址，地址增长 64M 后是第二块 SRAM 的基址，即 0x64000000。此后的情况以此类推。

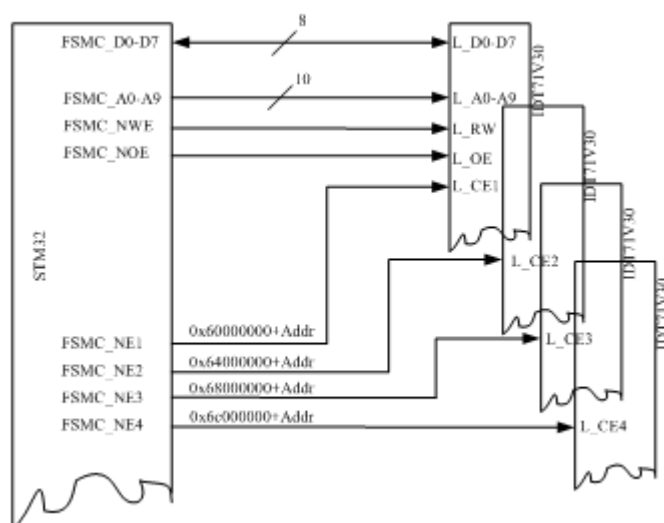


图 4.2 FSMC 存储器总线接口

2、双端口 RAM—IDT71V30 接口

双端口 RAM 具有两套独立的总线，两端的控制器可以同时异步访问同一个存储区。考虑到在该机器人的控制系统中，两个处理器的通信量有限，我们采用 IDT71V30，它具有 1K 字节的存储空间，8 位的数据总线，它与 STM32 系列微处理器的硬件接口如图 4.3 所示。

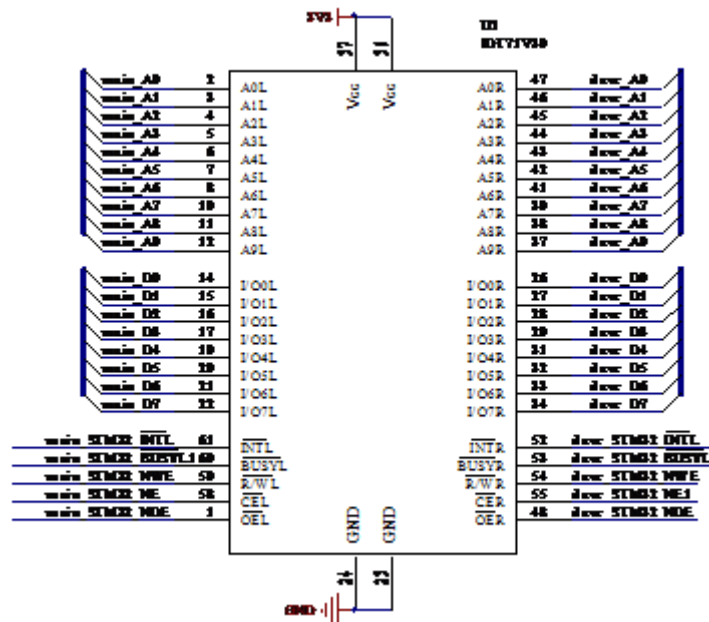


图 4.3 IDT71V30 电路原理图

需要说明的是，根据当前所选微控制器的性能参数，理论上机器人的主控制器可以同时连接四个协处理器。如果机器人的控制系统变得更加复杂，系统可能需要多于两个处理器同时工作。同时其他类型的专用处理设备可能会被加入到系统中，例如擅长信号

处理的 DSP 器件。

由于系统采用了两款同一系列的微处理器，因此它们对存储器访问的软件实现基本相同，这大大降低了系统开发的难度和工作量。

两个处理器通信的软件实现主要包括初始化、通信协议的制定以及访问冲突控制三部分。其中初始化主要包括对 FSMC 的使用配置，通信协议则根据机器人的控制要求，为每一个驱动电机分配固定的控制参数访问地址，方便两个处理器的快速数据交换。由于双端口 RAM 仅支持两个处理器在同一内存单元的异步访问，所以在实际的应用中，还是会遇到访问冲突的情况。该系统使用 IDT71V30 的访问仲裁单元提供的 BUSY 信号和邮箱中断机制来解决这一问题。

4.4.2 基于串行口的通信

串行口应用广泛的通信接口，基于串行口的通信可以分为 UART、SPI、I²C、CAN 等多种类型。在微处理器的应用中，UART 因为简单的接口和配置成为使用最频繁的通信方式之一。现在几乎所有的常用微处理器都具有片上 UART 控制器，用户只需简单的配置和连接就可以实现微处理器之间的相互通信^[49-51]。

如图 4.1 所示，在本文论述控制系统中，使用了基于 UART 的串行通信。由于单片机系统使用电压较低的 TTL 或者更低的电平标准，而机器人系统中又存在用于电机控制等的相对较高的功率系统。因此，单纯 TTL 电平的 UART 通信常常受到不同程度的干扰，进而影响整个机器人控制系统的正常运行。为此，我们使用 RS485 电平替代 TTL 电平。RS485 是一种串行通信的电平标准，与普通 UART 的单线传输不同，RS485 使用两线差分的方式进行传输，大大增强了通信的抗干扰能力。RS485 通信的接口原理如图 4.4 所示。

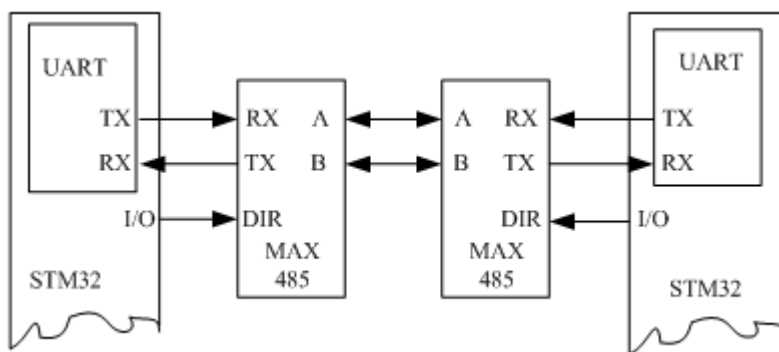


图 4.4 串行通信的 RS485 接口

由于 MAX485 是仅支持半双工的 UART 通信，即同一时刻只能发送或者接受，电

路设计时需要使用通用 IO 口来控制 MAX485 的方向信号。

尽管使用了上述的方式进行通信，并且 UART 控制器通常具有内部奇偶校验功能，但是在机器人的实际运行过程中干扰仍难以避免。因此，通信协议的制定以及正确的数据处理流程显得至关重要。表 4.1 是控制系统进行 UART 口通信的数据协议示例。

表 4.1 UART 数据通信协议

序号	1	2	3	4	5	6	7
含义	帧头 1	帧头 2	校验和	有效值	有效值	有效值	有效值
值	0-FF						

上述的通信协议的数据结构通常使用字节数组 buffer 实现。其中，序号 1-3 的字节是数据保护字段。帧头 1 用于接收软件识别数据传输的开始，因为帧头是固定的值，而校验和字节以及其他有效数据是不断变化的。因此，存在程序将其他数据识别为帧头的可能。为此，协议使用两个帧头来识别数据的开始，大大降低了传输错误的概率。校验和用于检测有效数据是否存在误码，它的值是所有有效数据累加溢出后的值。表 4.1 只是一个协议实例，为了使通信具有更高的可靠性，也可以增加帧头，当然也要付出效率的代价。

使用上述通信协议的软件流程如图 4.5 所示。

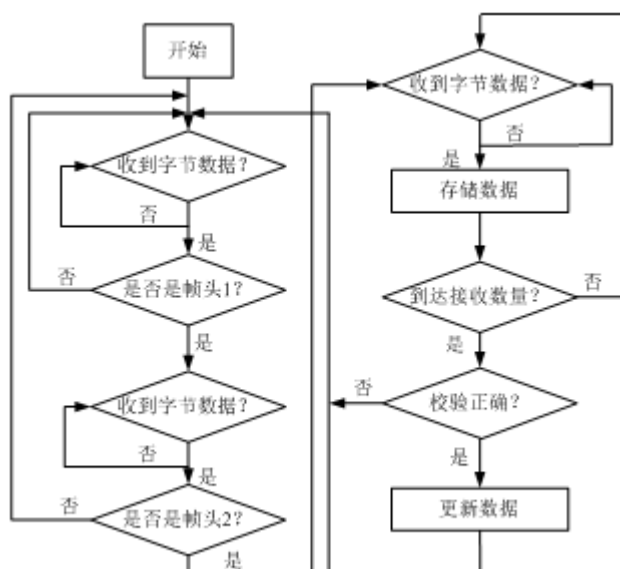


图 4.5 串行通信接收软件流程图

上述流程经过机器人的实际试验，结果证明数据协议及软件可靠。

4.5 直流伺服电机控制

4.5.1 电机控制方案

基于运动学的控制方案中，机器人底盘的驱动电机是以速度闭环的方式运行的，所以电机的实时同步控制不但要求高速通信，而且要求性能优越的控制算法，能够高效动态响应控制指令。

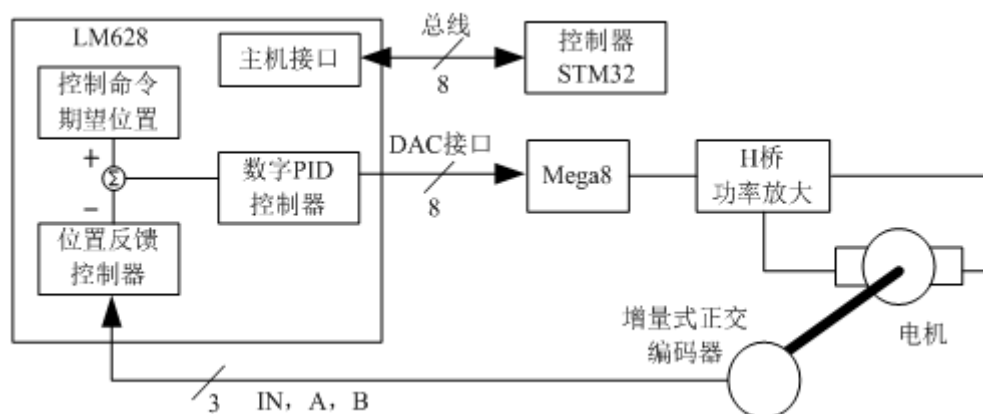


图 4.6 基于 LM628 的直流电机控制原理

电机的控制方案如图 4.6 所示。伺服电机的转动轴上安装有正交光电编码器，专用的伺服运动控制器 LM628 能够通过编码器结构获取电机的速度和位置信息，然后根据电机当前的控制要求改变输出信号，该信号经过功率放大后加载到电机上。电机驱动信号的改变最终会以转速变化的方式表现。

4.5.2 电机控制器 LM628 设计接口

1、LM628 与微控制器的接口

LM628 电机专用控制芯片，能够高效实现电机的速度伺服控制。LM628 提供标准的总线接口，微控制器通过该接口可以实现电机控制相关的命令和数据传输^[49]。

由于在实际应用中，它可以通过通用总线与计算机相连，实现基于 PC 机的多种运动控制应用。在我们的应用中，对比 SRAM 和 LM628 的总线操作时序可以发现，通过配置 FSMC 总线控制器的相关时序参数，可以使两个时序匹配。所以，使用 STM32 的 FSMC 来代替 PC 机上的通用总线，实现了 LM628 在嵌入式控制系统中的应用，与传统的 IO 模拟时序的方式相比，所有的控制信号均由硬件产生，因此控制效率更高，软件更加简洁。

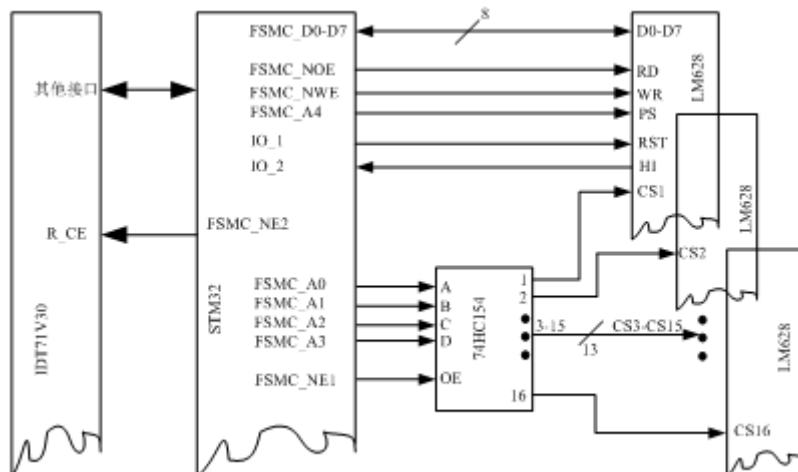


图 4.7 STM32 驱动多个 LM628 电路原理

LM628 的八位数据总线、控制信号 WR、RD 分别与 STM32 的 FSMC_D0~FSMC_D7、FSMC_NOE、FSMC_NWE 相连。与双端口 RAM 不同的是，LM628 有用于区分指令和数据的 PS 控制信号，我们将它与 FSMC_A4 相接，使用不同的操作地址来区分数据和命令。由于控制系统需要使用多个 LM628 来同时驱动多个电机，FSMC 的块使能信号 FSMC_NE1 和地址总线的 FSMC_A0-FSMC_A3 通过地址译码器 74HC154 产生不同的片选信号，该方案最多可同时控制 16 个 LM628，满足机器人底盘运动以及其他执行机构的控制需求。

2、LM628 与功率模块的接口

LM628 的输出为八位的 DAC 数字信号，使用 ATmega8 单片机的一组八位通用 IO 口接收该信号，并通过处理器的定时器将该信号转换成两路脉宽调制信号，下文将要介绍的功率放大模块将使用这两路 PWM 信号生成双极性的功率放大信号，进而驱动直流电机。电路原理如图 4.8 所示。

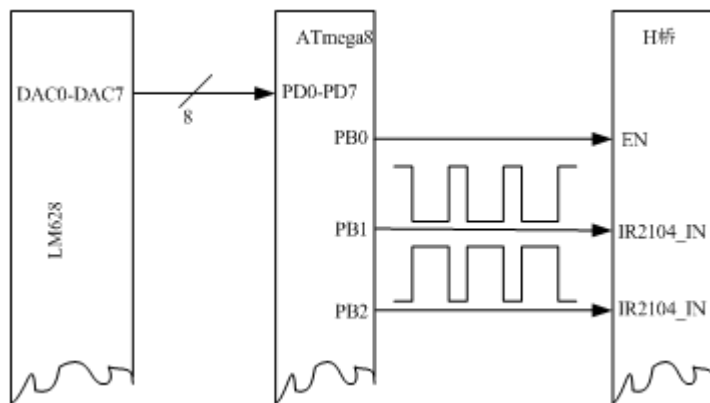


图 4.8 双极性 PWM 电机驱动原理

4.5.3 功率放大模块

直流电机的功率放大模块主要完成对直流电机专用控制器的输出信号的放大，进而直接驱动直流电机。模块的基本原理是驱动控制领域常用的 H 桥驱动方案，其基本原理这里不再赘述。本文论述的全向移动机器人使用如图 4.9 所示的 H 桥驱动方案。

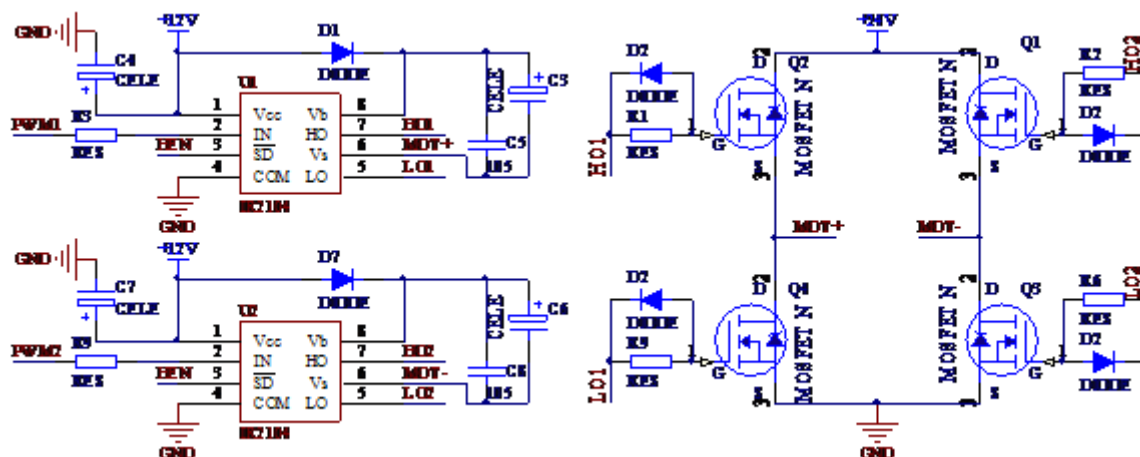


图 4.9 H 桥功率放大电路原理

模块主要由 MOS 管以及 MOS 管驱动器 IR2104 构成。H 桥由四个 n 沟道的 MOS 管搭成，IR2104 接收前面章节所述的两路 PWM 信号，并将其转换成可以使 MOS 管开断的驱动信号。

4.5.4 直流电机控制的软件设计

1、不同 LM628 接口实现方式的效率对比

以下是两种版本程序的对比：

1) IO 模拟版本的 C 程序如下：

```
u8 readStatus(u8 addr){                                     //读取一个字节的状态值
    u8 i,state;
    setDataPortIn;                                          //设置总线为输入状态
    for(i = 0; i < CPLD_NUMBER; i++){ //使能地址的片选信号
        if(i != addr) CS_1(i);
        else CS_0(i);}
    PS0;RD0;WR1;                                           //配置读状态命令的时序
    CPLD_GetData(state);                                   //获取数据
    RD1;PS1;                                                //恢复相应的控制引脚
```

```
        return (state);                //返回状态值
    }

2) 使用 FSMC 版本的 C 程序:

u8 readStatus(u8 addr){                //读取一个字节的状态值
    u32 RAM_BASE = 0x60000000 + addr;    //计算命令的映射地址
    return (*(vu8 *)RAM_BASE);          //读取相应的状态数据并返回
}
```

对比可知,使用并行总线的方式驱动 LM628,一条控制命令的有效代码量(C语言)由原来的 12 条语句(I/O 模拟的方式)减少到 2 条。这有效地减少了代码量,增加了系统软件的易读性。为了比较两种方案的实际效率,我们设计了如下实验:

对于 IO 模拟和并行总线驱动的方式,控制分别发送相同次数的同一指令,通过控制器的定时器设定精度为 1us 的定时。分别测定各自的执行时间,最终得到的实验结果如表所示,表中第二列代表取 10 次指令执行时间的平均值(us),其他列依次类推,最后一列给出了所有数据的总平均值。

表 4.1 不同接口执行一条指令用时对比

	10 次	100 次	1000 次	10000 次	100000 次	总平均值
IO 模拟	7.300	7.020	7.490	6.999	7.999	7.362
FSMC	1.990	1.990	2.335	2.500	2.499	2.263

实验结果显示,使用 IO 模拟的方式驱动 LM628 时一条指令执行的总平均时间为 7.362us,使用 FSMC 总线驱动时为 2.263us。因此 LM628 的方式使得程序执行的效率提升两倍多,这对具有频繁电机操作命令的控制系统来说具有重要意义。

2、电机控制的软件流程

电机的控制主要是微处理器与 LM628 通过并行接口进行电机控制相关命令及电机信息采集的流程。软件基于前后台控制系统结构实现,主程序中等待主控制器的相关命令。当主控制器有电机的控制命令发出时,电机控制模块会产生一个 RAM 邮箱地址的中断,在中断中判断命令的类型。如果命令是电机的速度或者位置控制命令,则控制器直接读取相应 RAM 地址的数据,然后向 LM628 传送控制命令。如果是数据获取命令,例如读取电机当前的转速或者位置信息,则控制器首次向 LM628 发送数据获取命令,得到数据以后,写入相应的 RAM 地址中。最后以邮箱中断的方式通知主控制器命令已

经完成，可以发送下一条指令。详细的软件控制流程如图 4.10 所示。

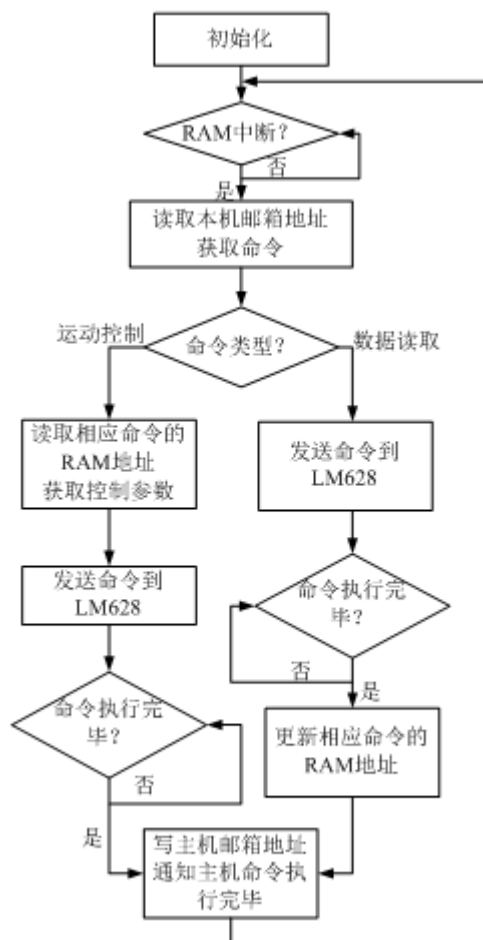


图 4.10 电机控制的软件流程

4.6 机器人定位子系统设计

4.6.1 基于编码器的里程测量系统

1、编码器测距原理及选择

光电编码器是机器人里程计功能实现的核心器件。如图 4.11 所示，它通过与小尺寸的瑞典轮配合，能够将旋转运动转变为计算机系统可识别的脉冲信号。通常，光电编码器具有两路相位差为 90° 的输出，控制器通过检测与用于检测两路输出的相位关系和电平状态组合，实现方向鉴别和四倍计数的功能。编码器的最主要参数是线数，即旋转盘上光栅个数 N 。 N 越大，其精度越高。

式 (3-3) 是里程计位置增量 ΔS 的计算公式。

$$\Delta S = \frac{\pi D}{4N} (n_{t+\Delta t} - n_t) \quad (4-1)$$

式中:

n_t — t 时刻编码器的计数值;

$n_{t+\Delta t}$ — $t+\Delta t$ 时刻编码器的计数值;

N —编码器线数;

D —瑞典轮直径。

课题研究的机器人控制系统实现中, 选择线数 $N=512$ 的光电编码器, 小瑞典轮直径为 50mm , 则单个编码器脉冲值代表的距离 $S=\pi*50/(4*512)\approx 0.0767\text{mm}$ 。

2、测量方案的选择

本文研究的机器人针对所使用的编码器实现了以下两种测量方案。

1) 方案一:

STM32f103 的定时器, 只需通过简单的软件配置能够很方便的读取编码器的数值。定时器的编码器功能模块能够自动完成编码器正反转的鉴相以及计数的四倍频, 这大大简化了机器人定位模块的程序设计以及微处理器的工作负载。

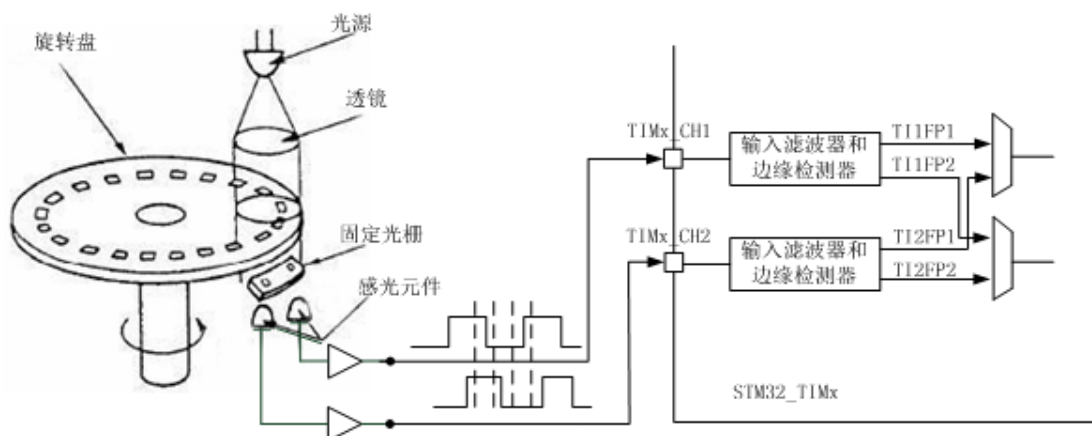


图 4.11 STM32 编码器接口原理

由于系统使用的编码器是集电极开漏输出, 所以加入上拉电阻来保证它的正确输出。

采样时刻, 编码器测量结果的计算示例如下:

$$\text{EncCounterX} = (\text{overFlowTIMx} \ll 16) + \text{TIMx_CNT};$$

式中:

overFlowTIMx—为定时器 x 的溢出次数

TIMx_CNT—定时器 x 计数寄存器的值

由于 STM32 的通用定时器具有 16 位的计数精度, 当计数值超过 65535 时, 定时器

的相关寄存器会产生溢出，此时需要设置触发溢出中断，在中断处理程序中判断编码器的计数方向并更新溢出次数 `overFlowTIMx`，如果计数方向向上，则 `overFlowTIMx` 加一，反之则减一。由于变量为有符号整型变量，因此当前计数值 `EncCounter` 也带有符号，规定正值为机器人局部参考框架 X_R 轴正方向，负值为 Y_R 轴负方向。具体代码如下：

```
void BSP_IntHandlerTIM4(void){
    if(TIM_GetITStatus(TIM4, TIM_FLAG_Update) != RESET ){
        //清除中断标志
        TIM_ClearFlag(TIM4, TIM_FLAG_Update);
        //如果计数方向向下，则溢出次数减一
        if ((TIM4->CR1 & TIM_CounterMode_Down) == TIM_CounterMode_Down) {
            overFlowTIM4--;}
        else if((TIM4->CR1 & TIM_CounterMode_Up) == TIM_CounterMode_Up){
            overFlowTIM4++;}
        }
    }
```

2) 方案二：

使用直流电机专用运动控制器 LM628 的编码器接口，前面的章节中已经说明 LM628 具有正交编码器的接口，它可以实时获取编码器 A、B、Z 三相的脉冲输入，进而计算编码器旋转轴的转速等。使用该方案，只需使用相应的查询命令即可高速读取编码器的累积数值。

上述的两种方式均经过实验验证，实验结果证明两种方法都能有效的采集编码的数据。第一种方式的接口电路非常简单，但是软件部分需要处理相应的中断，并且随着机器人运行速度的加快，编码器信号输出频率的增加，系统相关的中断频率也会相应增加。第二种方式需要增加相应的接口，外围电路相对较多。但是在软件方面，系统不需要处理类似第一种方式的中断，所以也就没有因为机器人运行速度增加带来的潜在软件隐患。

4.6.2 基于陀螺仪的姿态测量系统

1、陀螺仪的选择

陀螺仪是机器人定位系统中常见的惯性传感器。它能够反馈机器人的瞬时自旋角速度，该信息经过计算机积分处理后即是机器人的姿态信息。陀螺仪已经有超过百年的发展历史，现阶段应用比较广泛的有激光陀螺、光纤陀螺以及微机械陀螺等。其中，光纤陀螺的数据精度、偏置稳定性等核心性能在整体上要优于其他类型的陀螺。

由于机器人平面运动，只需考虑垂直于运动平面的轴，所以使用单轴陀螺仪测量采样时刻机器人局部参考框架与全局参考框架的夹角，即可完成机器人的导航任务。

基于以上分析，课题研究的机器人定位系统使用单轴光纤陀螺仪—VG949P。其主要性能参数如表 4.2 所示。

表 4.2 VG949P 型陀螺仪性能参数表

名称	值	名称	值
测量范围	$\pm 230^{\circ}/s$	振动	2g(20-500Hz)
比例因子 (SF)	$6\text{mv}/^{\circ}/s$	供电电源	$+5\text{v} \pm 0.25\text{v}$
频带	0-1KHz	加速度	5g
偏置稳定性	$20^{\circ}/h$	工作寿命	20000h
稳定时间	0.02s	安装误差	$90^{\circ} \pm 1^{\circ}$
角度随机游走	$0.05^{\circ}/\sqrt{h}$	功耗	1w
线性误差	4%	SF 变动误差	0.1%
工作温度	$-30^{\circ}\text{C} - 70^{\circ}\text{C}$	撞击	40g(1ms)

2、陀螺仪子系统的构成

由于机器人对传感器的实时性能要求较高，而陀螺仪的数据采集需要软硬件滤波、数值积分等复杂操作，所以采用了单独子系统来完成陀螺仪的数据采集工作。陀螺仪子系统主要由微处理器 STM32F103、具有高精度的模数转换器 ADS1255 以及其他相关外围电路构成，如图 4.14 所示。微控制器通过 SPI 接口完成对 ADS1255 命令发送与数据传输工作，然后将数据经过软件均值滤波、数值积分等操作，最终得到机器人当前的姿态信息。在机器人的定位采样时刻，控制器通过 UART (RS232 或者 RS485) 将该信息传送到主控制器上。

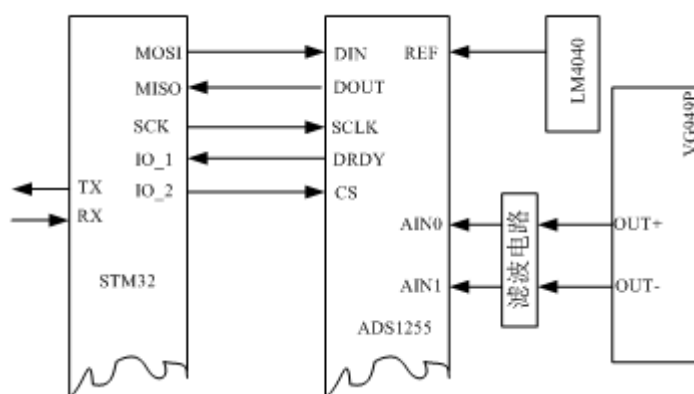


图 4.12 陀螺仪子系统接口原理

3、陀螺仪 AD 采集分析

VG949P 型光纤陀螺仪的信号需要经过模数转换才能被机器人使用。选用高精度的 AD 转换器 ADS1255，它具有 24 位的精度，最高 30kSPS 的采样频率；同时，采样非线性度最高只有 $\pm 0.0010\%$ ，长时间使用时将有效提升陀螺仪的积分效果。ADS1255 电路原理如图 4.12 所示。

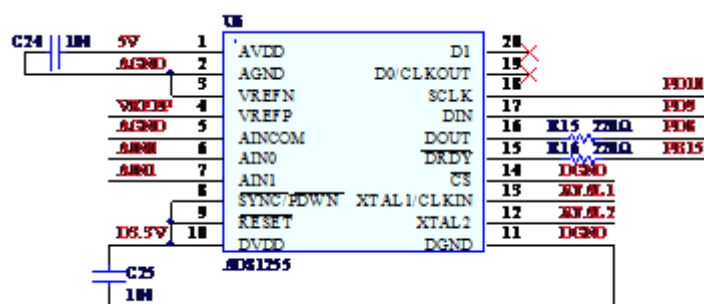


图 4.13 ADS1255 接口原理图

ADS1255 使用电源基准芯片 LM4040 为其提供 $V_{REFP}=2.5\text{v}$ 的正参考电压，负参考电压 V_{REFN} 接地，因此 AD 的参考电压 $REF=V_{REFP}-V_{REFN}=2.5\text{v}$ 。由表 4.2 可知，VG949P 的模拟量比例因子 $SF=0.6\text{mv}/^\circ/\text{s}$ ，最大转速限制为 $\pm 230^\circ/\text{s}$ 。ADS1255 内置最大为 64 可编程输入增益系数 PGA，通过 STM32 的 SPI 接口即可实现相关配置。因此，理论上陀螺仪的模拟量输出最大值 $V_{\text{max}}=PGA \times 0.6\text{mv} \times 230 = (1.38 \times PGA)\text{v}$ 。假设角速度为 ω ，则 ω 可按式（4-2）求解。

$$\omega = \frac{D_{out} * 2.5}{PGA * 2^{24} * 0.6} \quad (4-2)$$

式中， D_{out} 表示 AD 度数。式（4-2）可得：

$$D_{out} = \frac{\omega * PGA * 2^{24} * 0.6}{2.5} \quad (4-3)$$

在相同的角速度下，AD 输出的值越大，外界干扰对数据的影响就越小，精度也就越高。在应用中设置 PGA=2，此时理论上陀螺仪的输出极限值 $V_{\max}=2.76\text{V}$ 。由于在实际的应用中机器人实际的自转角速度的绝对值不会达到 $230^\circ/\text{s}$ ，而对数据采集的精度要求较高。在此我们可以合理地估计 $V_{\max}<2.5\text{V}$ 。因此，将 PGA 设置为 2 不但能够满足机器人的使用要求，并且能提高数据的精度。

模拟电路对系统电源的要求很高。选用德州仪器的 TPS7A47 线性稳压电源为陀螺仪模块供电。TPS7A47 具有 +3V 至 +36V 的输入电压范围，具有多种灵活的可配置输出电压，最大输出电流为 1A，能够满足陀螺仪机器人相关组件的供电要求。TPS7A47 的电路原理如下图所示。

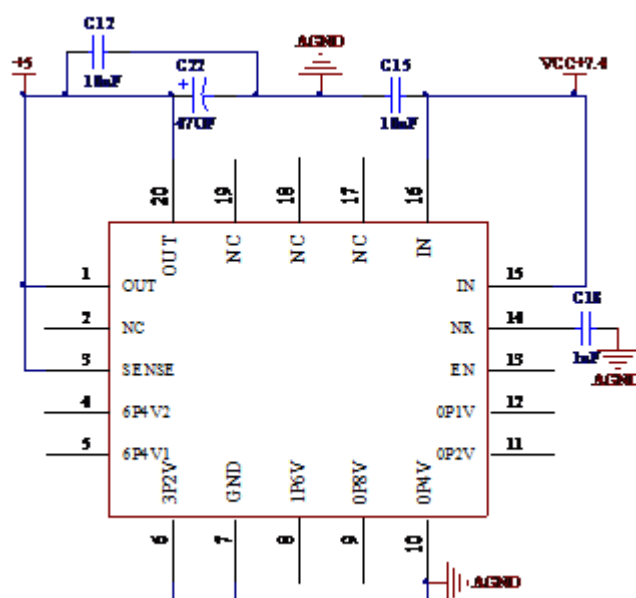


图 4.14 TPS7A47 电路原理

4、陀螺仪积分算法实现

前文介绍了陀螺仪接口原理。由于它的信息表征的是角速度，需要进行数值积分才能得到姿态角。对于数值积分，常用的方式是直接进行累加，但是因为采样率有限，随着时间的增长，这种方式会产生较大的累计误差。本文采用了 Simpson 公式来改进测量效果。

二次多项式插值的数学表达为

$$\int_{x_0}^{x_1} f(x)dx \approx \int_{x_0}^{x_1} L_2(x)dx \quad (4-4)$$

式中 $L_2(x)$ 为拉格朗日二次插值多项式，即

$$L_2(x) = \frac{(x-x')(x-x_1)}{(x_0-x')(x_0-x_1)} f(x_0) + \frac{(x-x_0)(x-x_1)}{(x'-x_0)(x'-x_1)} f(x') + \frac{(x-x_0)(x-x')}{(x_1-x_0)(x_1-x')} f(x_1) \quad (4-5)$$

特别地，当 $x' = 1/2(x_0 + x_1)$ 时，得到 Simpson 公式：

$$\int_{x_0}^{x_1} f(x)dx \approx \frac{(x_1-x_0)}{6} \left[f(x_0) + 4f\left(\frac{x_0+x_1}{2}\right) + f(x_1) \right] \quad (4-6)$$

陀螺仪的输出与机器人的实际运行情况相关，不同的运行路线都有不同的输出 $f(t)$ ，在基于微处理器的陀螺仪数据采集积分应用中，假设采样周期为 ΔT ，则根据上式可得到离散化的 Simpson 积分公式：

$$\int_t^{t+2\Delta T} f(t)dt \approx \frac{\Delta T}{3} [f(t) + 4f(t+\Delta T) + f(t+2\Delta T)] \quad (4-7)$$

使用式（4-7）即可完成陀螺仪的积分，积分流程图如下

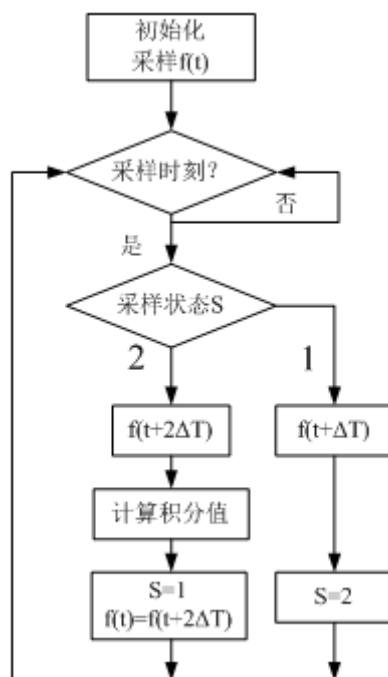


图 4.15 陀螺仪积分流程图

4.7 机器人控制系统软件设计

4.7.1 基于 uCOS-III 的机器人控制软件的实现

uC/OS 是嵌入式控制领域被广泛应用的微型操作系统，1992 至今已经发展至第三代 -uC/OS-III。uC/OS-III 对任务的个数无限制，并且支持多种嵌入式系统的任务管理功能。除此之外，该系统也因为有更多方便实用的特殊功能而备受青睐。

由于本文论述的全向移动机器人的控制器是基于运动学模型的，所以要求控制系统

必须满足一定的实时性，才能达到理想的定位及运动控制效果，因此机器人控制系统是典型的实时系统。

嵌入式系统中常用的软件实现结构主要包括以主程序为入口的后台程序，以及响应特性要求较高的各种中断处理的前台程序。在日常生活中的常用小电器等产品实现大都基于以上结构。这种结构的运行流程如图 4.16 所示，基于该结构的控制系统要求中断的处理程序具有较高的响应速度，以满足主程序中各个任务的准确响应。

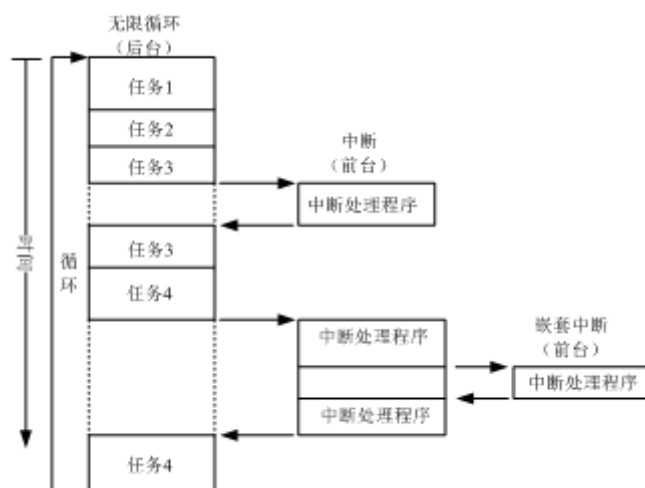


图 4.16 前后台操作系统

从实时性的角度考虑，结构简单的前后台系统在简单的应用中具有较高的实时性，例如微波炉、电话和玩具等。它具有的主要特点是：

- 1) 系统任务计算量及任务数量少；
- 2) 任务之间共享资源较少，无需处理复杂的资源共享冲突问题；
- 3) 功能相对固定，软件比较稳定。

但是，在复杂的机器人应用中，需要实时地处理多种运算任务。仅仅使用前后台系统常常要求开发人员管理多种资源，会使系统的逻辑变得复杂，难以维护甚至造成系统崩溃。同时前后台系统在处理器使用效率方面存在不足，当任务进一步变得复杂时，由于处理器资源限制会使系统的实时性降低。

uC/OS-III被设计用于 32 位处理器，而全向移动机器人的主控制器 STM32 的 CM3 内核增添了专为系统应用的系统滴答定时模块 SysTick 和可挂起中断 PendSV。其中，SysTick 是内核级 24 位定时器，可以为嵌入式操作系统提供周期性的溢出时基，在溢出中断中调用任务切换。PendSV 用于上下文切换（在不同任务之间切换）。综上，uC/OS-III能够结合主控制器组成一个高效的机器人导航算法实现平台。

机器人的控制系统要求控制软件实时高效地完成定位以及路径跟踪算法的实现。其

中，根据各个算法对时间敏感度的分析，合理的将算法的实现分解成多个任务。例如，将定位分为传感器的采集和定位算法模型实现。无论是前后台系统还是实时操作系统，它们的处理器核心在每个时刻只能做一件事。可以将实时操作系统认为是逻辑高度复杂的多前后台系统的组合，这对于全向移动机器人导航的模块化软件实现具有重要意义。

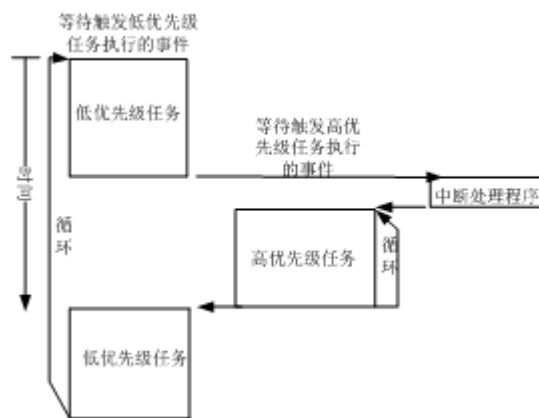
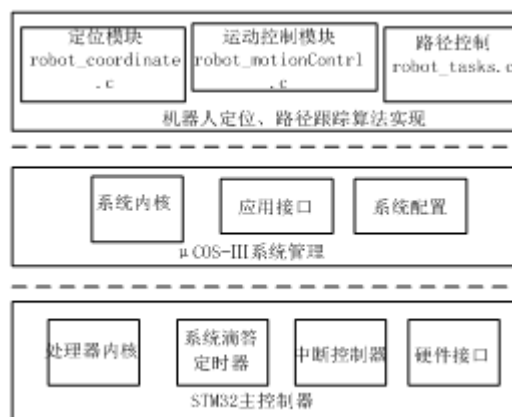


图 4.17 嵌入式系统内核原理



4.18 机器人控制软件框架结构示意图

如图 4.17 所示，在底层的原理上，两种系统其实没有本质的差别。但是操作系统级的软件框架能够有效的管理处理器的各种资源，课题在 $\mu C/OS-III$ 下实现了机器人定位及运动控制算法。基于图 4.18，设计实现了如图 4.19 所示的机器人定位及路径跟踪控制流程，程序的运行步骤如下：

- 1) 进行系统的初始化工作，其中主要包括定位任务、路径跟踪任务以及离线路径规划任务的创建和启动。
- 2) 执行离线路径规划任务，判断任务中是否存在已经规划好的一系列路径。如果没有，则程序直接结束，反之则读取一条路径信息。
- 3) 每条路径信息中都包含路径解析表达信息、起点和终点位姿，路径跟踪任务根据这些信息选择相应的距离和姿态角控制路线，最终解算出当前的控制量。
- 4) 根据式 (2-9) 解算驱动电机的速度并发送指令。至此，该周期控制任务已经完成，机器人将向趋向理想路径的方向运动。
- 5) 定位任务等待相关的信号量，当采样时刻到达后，采集编码器及陀螺仪的相关数据，依据式 (3-8) 进行实时位姿解算。
- 6) 判断当前位姿是否足够接近目标的终点，如果是则执行步骤 2) 进行下一路径的跟踪，反之则执行步骤 3) 继续执行当前路径的跟踪任务。

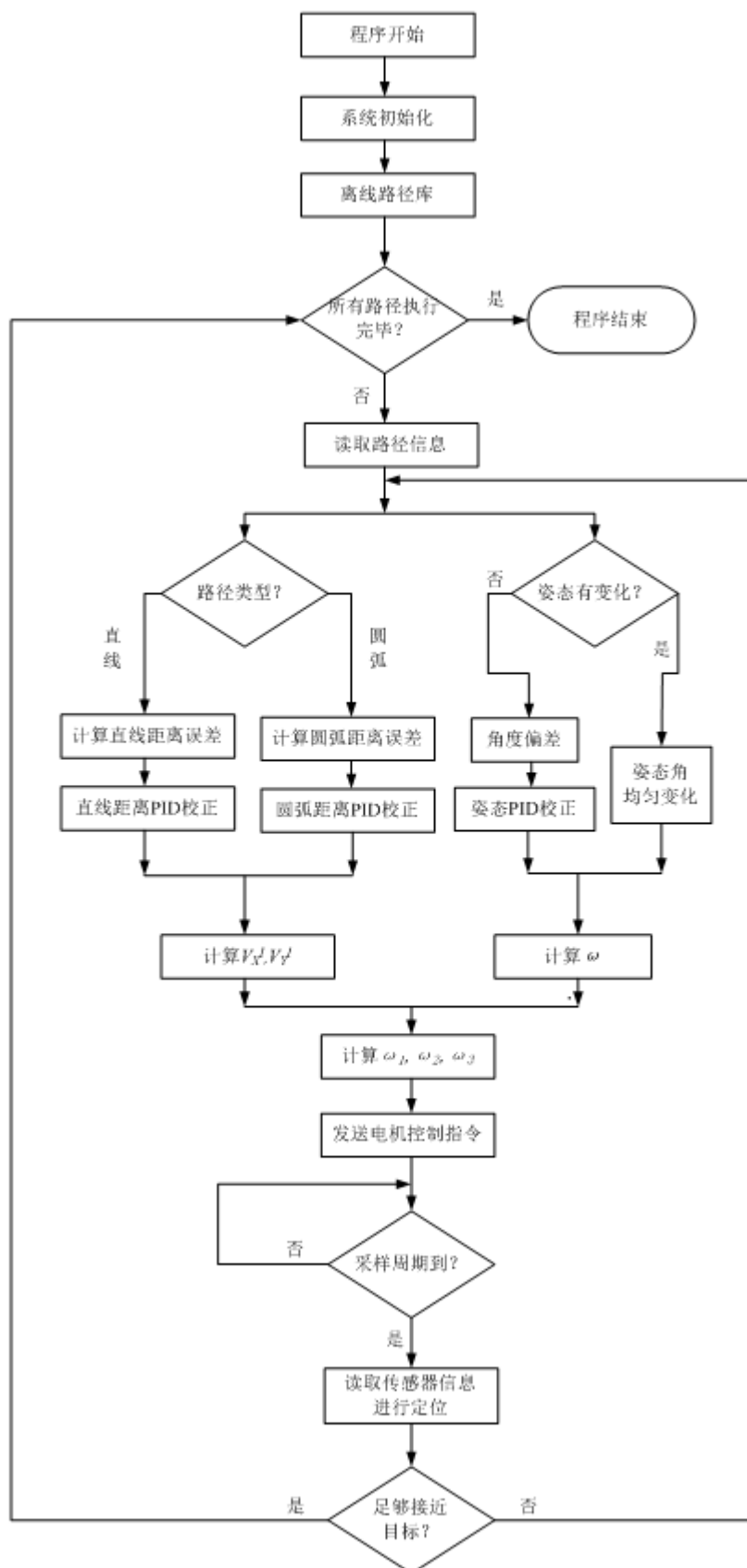


图 4.19 机器人定位及几何路径跟踪流程图

4.7.2 机器人软件优化分析

1、DMA 在通信中的应用

现阶段，无论是在微处理器还是在通用计算机处理器领域，存储器访问速度与中央处理器运算速度存在严重的不匹配现象，这是限制其效率的主要因素。因此，减少处理器在与外界通信过程中 CPU 对外设的访问能够提高机器人控制软件的运行效率，同时能够有效避免由于通信数据量增加带来的潜在软件执行错误。

中断的即时响应是嵌入式控制系统必然会面临的问题，尤其是在具有多任务处理能力的嵌入式操作系统的框架下进行机器人软件的开发，中断即时响应更为重要。本文介绍的全向移动机器人对机器人的控制的实时性具有较高的要求，机器人在与其他模块通信过程中经常使用中断机制来保证传感器或者其他控制信息的实时传输。但是频率过高的中断请求势必会增大微处理器的负载，在 uCOS 中频繁的中断会造成任务的严重延时，进而影响整个控制系统的性能。因此，中断的有效减少意义重大。

STM32F103 内部集成的 DMA 控制器提供了多种数据通道。以下是在机器人控制系统中，串口通信使用 8 位的数据传输格式，分别针对串口发送与接收进行执行效率的分析。数据传输格式如表 4.3 所示，实验结果如表 4.4 所示。

表 4.3 串口通信数据协议示例

序号	1	2	3	4	5	6	7	8
含义	数据头	数据头	数据类型	校验和	有效值	有效值	有效值	有效值
值	AA	08	0-FF	0-FF	0-FF	0-FF	0-FF	0-FF

表 4.4 两种中断方式的对比

	使用串口中断	使用 DMA
发送中断处理次数	8	1
发送 CPU 内存访问次数	8	8
接收中断处理次数	8	4
接收 CPU 内存访问次数	8	0

由上表可知，CPU 在通信过程中，使用 DMA 的串口通信时 CPU 的内存访问次数比普通的串口中断接收模式时要少。由于 DMA 传输过程中使用了连续传输的模式，这种模式下相同类型的数据可以连续传输，例如表 4.3 中，字节 5 至字节 8 只需要一次连续传输及一次中断。因此中断的处理次数在发送和接收均能得到的有效的减少。这将大

大降低中断及内存太过频繁而造成任务延时的概率，改善机器人的控制效果。现阶段微处理器的存储器容量足够大，可以忽略空间复杂度提高产生的影响。但是随着机器人导航算法时间复杂度的提高这种通信方式的作用将会越来越重要。

2、指针在数据传输中的应用

本文论述的机器人控制系统使用了多处理器结构，陀螺仪模块中，由处理器的积分后的机器人姿态角是 `double` 类型的数据，对于该数据的传输，有以下两种方案：

方案 1：将 `double` 类型的数据转换成 ASCII 码格式的字符串，然后将字符串按字节发送出去。这种方式的实现需要将数据按位处理，通信负载会随着数据精度的提高而增加。

方案 2：根据 IEEE754 浮点数据定义的标准可知，在 32 位的处理器上，`double` 类型数据的内存模型使用了连续的 8 字节的内存地址。使用字符指针指向数据的首地址，读取字节发送出去。同样的，在数据接收端，使用 `double` 类型的指针指向字符型接收 `buffer` 的首地址，对指针直接进行取操作即可完成对数据的接收。这种方式无需对数据位进行字节处理，无需关心处理器的大小端模式，而且能够以固定的数据传输长度，保留双精度浮点型数据的完整精度。

根据以上的对比，我们使用方案 2 完成了该模块的程序设计：

1) 发送端的程序执行过程：

- a) 根据公式计算当前的角度值 `angle`
- b) 定义字符指针，并使指针指向数据

```
unsigned char *gyroSendBuffer;  
gyroSendBuffer = (unsigned char *)&angle
```

- c) 发送 `gyroSendBuffer` 指向的数据块（8 字节）

2) 接收端的程序执行过程：

- a) 接收端将接收到的数据存储在缓冲内存 `usart3DmaReceiveBuffer[8]` 中
- b) 定义双精度数据指针，并将指针指向串口接收缓冲区的首字节

```
robotCoor.Angle = *((volatile double *)&usart3DmaReceiveBuffer[0])
```

- c) 使用 `robotCoor.Angle` 对机器人运动控制进行相关计算

实验证明，上述程序能够有效地完成陀螺仪的数据传送工作。这种方式不仅在通信的整体效率上体现出优势，而且大大减少的通信相关的代码量，使软件更加简洁明了。

4.8 本章小结

本章主要完成全机器人控制系统设计。设计实现了以 STM32 主控接口，介绍了主控器的主要特性。使用双端口 RAM 和 FSMC 总线控制器实现了机器人控制系统多机并行通信。介绍了基于 Simpson 中矩公式的陀螺仪数值积算法，基于嵌入式操作系统 uC/OS-III 构建了机器人控制系统软件框架，基于 DMA 在嵌入式控制器之间通信的应用以及高效的指针应用，实现对机器人控制系统的优化。

第5章 全向移动机器人定位及路径跟踪实验

5.1 引言

前面的章节对课题研究的三轮全向移动机器人的定位及路径跟踪算法进行了论述，同时完成了机器人嵌入式控制系统的设计实现。本章将分别对机器人的定位及运动控制算法进行实验分析。实验包括运动控制算法模型的仿真实验和样机实验。

5.2 全向移动机器人路径跟踪算法仿真实验

5.2.1 运动学仿真模型建立

在 Matlab/Simulink 环境下对机器人的路径跟踪算法进行建模。由于本文设计的机器人路径跟踪算法是基于运动学的仿真，并且算法的实现依赖定位的实现。如图 5.1 所示的仿真模型在实现机器人的实时定位时采用了积分模，即直接对机器人的控制速度直接积分作为定位的实现。如图 5.1 所示。这种假设与实际的机器人运动模型不符，但是在验证机器人运动控制算法的实现时该假设是合理的。因为与实际的运动控制模型存在差异，所以模型中 PID 误差校正与实际的情况存在较大差异。

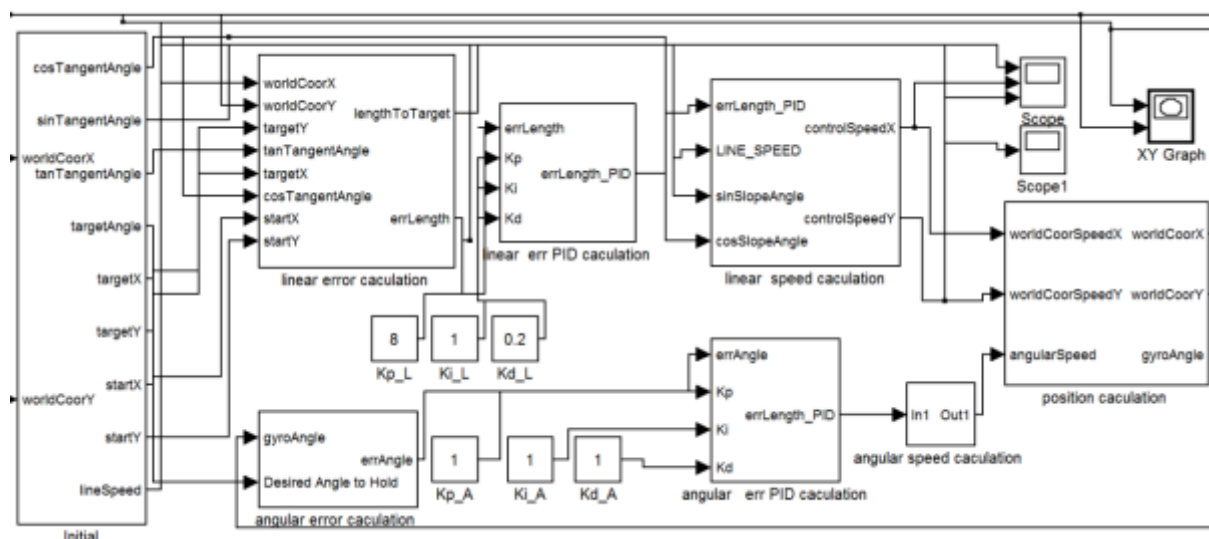


图 5.1 机器人运动学仿真模型仿真

5.2.2 仿真实验结果及分析

采用如图 5.1 所示的直线路径仿真模型进行仿真，初始姿态为 (1m, 0m, 0rad)，终点姿态为 (7m, 6m, 0rad)，仿真过程中分别设置机器人的运动速度 $V=0.2\text{m/s}$ 、 $V=0.6\text{m/s}$ 以及 $V=1.0\text{m/s}$ 。

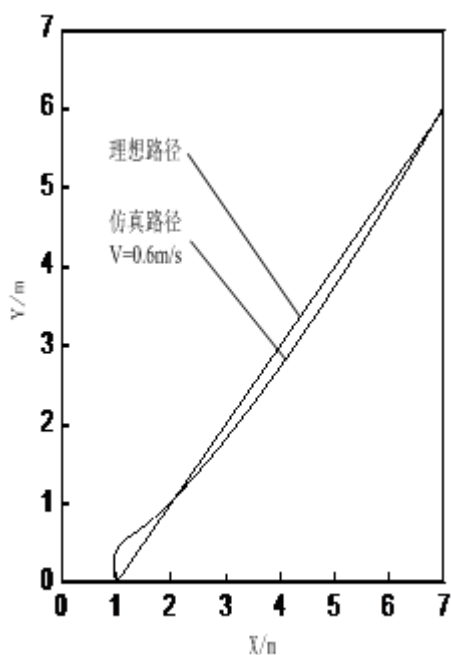


图 5.2 直线路径仿真结果

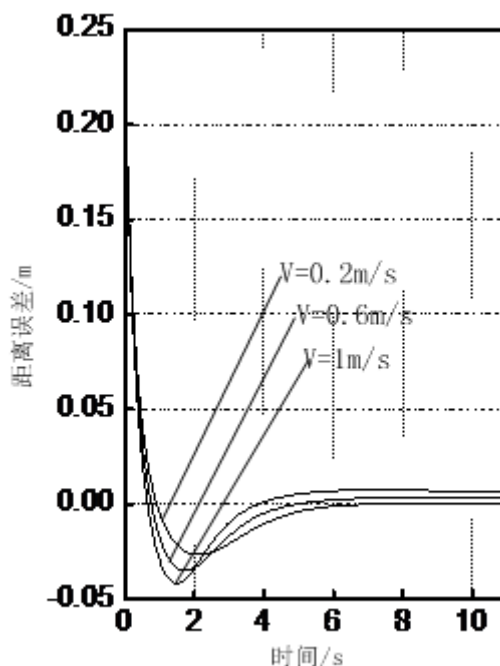


图 5.3 不同仿真速度的误差

如前文所述，机器人定位系统的实现是机器人控制速度的直接积分，如果将仿真的初始误差设置为 0，那么机器人 PID 的控制输出也将为零，最终将无法看到控制器的调节效果，所以在仿真的初始时刻设定机器人的初始位置误差为 0.25m。直线路径仿真如图 5.2 所示，仿真结果表明，机器人基于运动学的几何路径跟踪算法能够实现机器人在平面环境下正确运动。机器人在不同仿真速度下的位置误差输出如图 5.3 所示，结果表明，机器人的运动速度越大时，在给定的前述仿真条件下，机器人偏差最大值也越大。仿真的数据结果如下表所示。

表 5.1 不同仿真速度的最大距离偏差

速度 V (m/s)	0.2	0.6m	1.0
最大偏差 (m)	-0.026600795	-0.035081117	-0.041708298

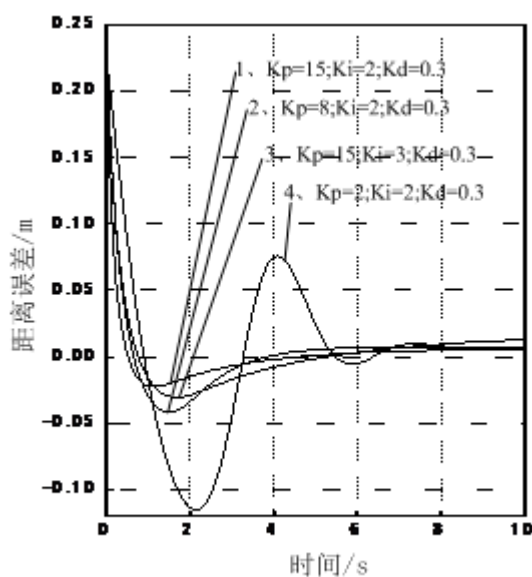


图 5.4 不同 PID 参数的仿真结果

图 5.4 为在相同的速度 $V=1\text{m/s}$ 的情况下，设置不同的仿真 PID 参数，最终得到的误差输出。比较曲线 1 和 4 发现，比例参数对仿真误差输出的影响较大。因为仿真的模型并不能完全代表实际的机器人运动过程，上述的 PID 参数调节与实际的运行时存在不同。

5.3 全向移动机器人样机定位实验

5.3.1 样机实验环境介绍

机器人的样机实验环境主要包括机器人运动场地和机器人实验数据采集系统介绍。由于机器人定位及路径跟踪算法所涉及的传感器及驱动轮对运动场地的要求较高，因此在不同的实验环境下，可能得到差别很大的实验结果。同时，根据前面章节有关机器人定位算法的论述，样机实验环境要求具备准确的机器人参考位置，以及用于初始化机器人状态的机械定位结构。机器人样机实验需要有相关的测量装置及实验数据采集。试验样机具备了机器人定位及路径跟踪的基本功能模块。底盘结构由 $2\text{cm} \times 2\text{cm}$ 的铝管搭建，控制系统使用 24V 锂电池供电。

1、实验场地

机器人的实验场地为亚太机器人电视大赛机器人训练场地，场地的主要参数如下：

面积： $12\text{m} \times 12\text{m}$ ；

材质：木质（胶合板）；

平整度： $\pm 5\text{mm}$ （注：地面的不平整主要表现在两块胶合板接合处）。

2、基于 labview 的实验数据采集系统

在样机实验过程中需要实时的采集机器人的运动状态信息，例如，机器人实时位姿和机器人运动速度等。使用 labview 结合控制系统的异步串口（UART）搭建的实验数据采集系统。系统的结构如图 5.5 所示，样机实物及数据采集示意如图 5.8 和图 5.9 所示。由于机器人本体和数据处理是两个独立的系统，而试验中机器人位置是在不断变动的，所以两者之间的通信使用无线传输模块（WAP200），这样省去了两个系统之间的串行通信线。

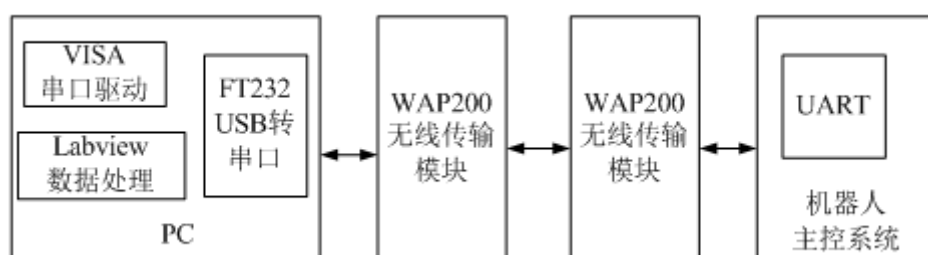


图 5.5 数据采集系统原理

在基于 Labview 的数据采集系统中，设计实现了机器人路径跟踪距离偏差、运动轨迹、姿态角以及姿态角跟踪误差四个主要的显示界面。图 5.6 和图 5.7 是在机器人样机实验中界面的显示示例。

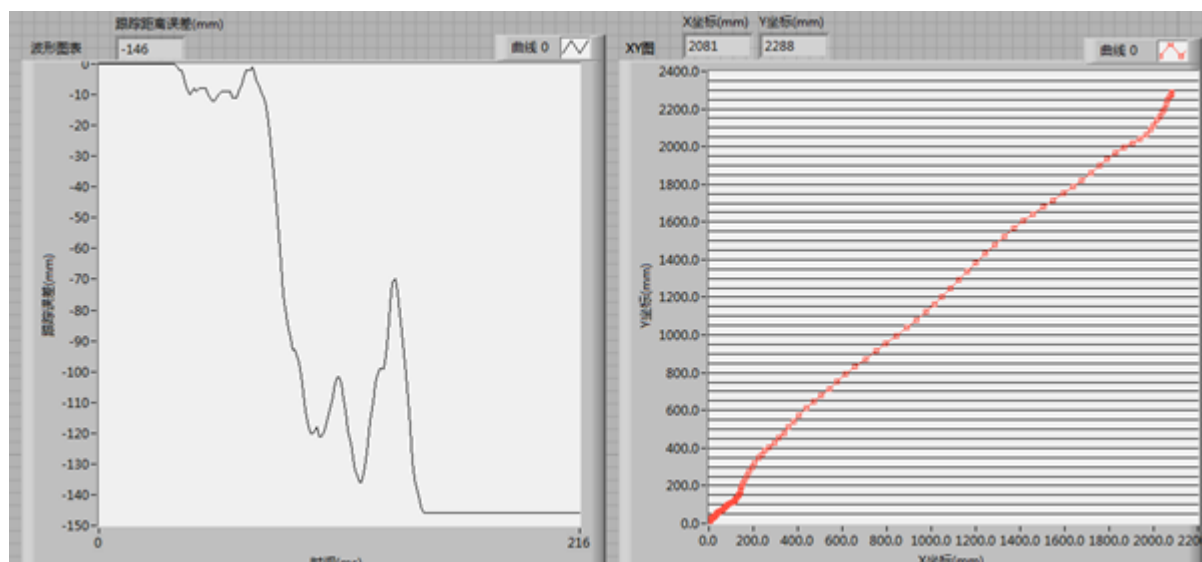


图 5.6 机器人路径跟踪距离偏差（左图）和运动轨迹（右图）界面

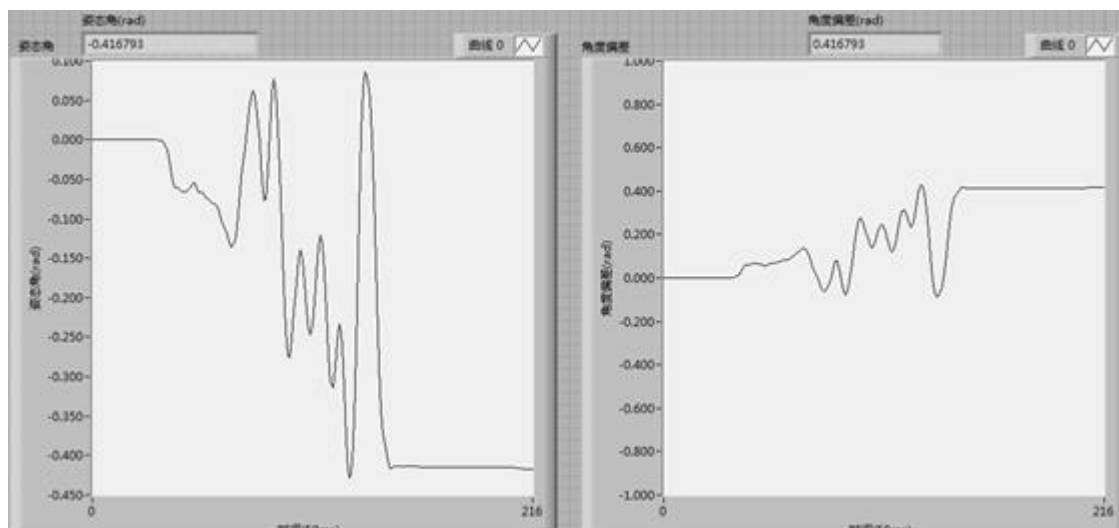


图 5.7 机器人姿态角显示（左图）和姿态角误差显示（右图）界面



图 5.8 样机数据采集

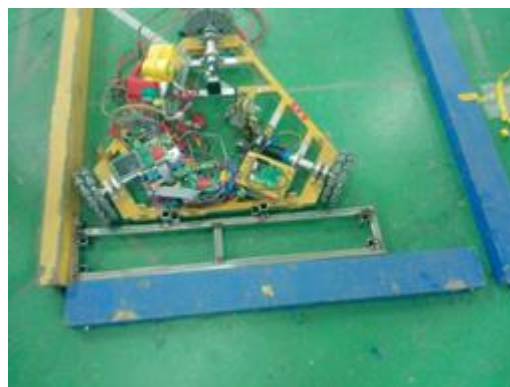


图 5.9 真实参考框架

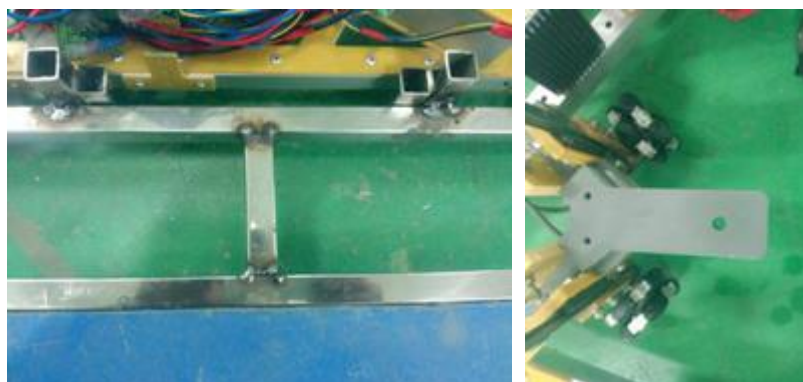


图 5.10 机器人初始定位结构（左图）和实际测量参考点（右图）

3、数据测量基准：

在样机实验的过程中需要测量机器人移动的距离，对于全向移动机器人来说，选择合适的参考点及测量方式是保证实验结果可信的关键。由 3.5.1 节可知，机器人在运动

时误差会有一定容限 $\pm\theta_l$ ，所以机器人在一条路径的终点姿态会有最大的偏差，这个偏差是两倍 θ_l ，这对于实际的机器人应用来说或许无关紧要。但是，在机器人的定位误差实验阶段，可能会产生很大影响。

如图 5.11 所示，假设机器人选用测量参考点 A，即距离机器人几何中心最远的点。当机器人在终点出现姿态偏差时，将会产生较大的测量误差。由表 2.1 可知机器人试验样机测量参考点 A 到机器人几何中心的距离近似为 $l_i=350\text{mm}$ ，则图 5.11 所示的测量偏差 Error 满足如下算式：

$$\text{Error} \approx l_i * (2\theta_l) \quad (5-1)$$

θ_l 按照机器人的相关设计指标以 1.5° 计算，最终得出机器人以 A 点为测量参考点时的最大测量误差 Error 约为 9.163mm，这对于机器人的误差校正来说是致命的。

如果以参考点 B，即机器人的几何中心为参考点，则不会产生上述的测量误差。因此，本文在机器人样机实验中，凡是涉及到机器人距离测量的时候，都以机器人的几何中心为测量参考点。

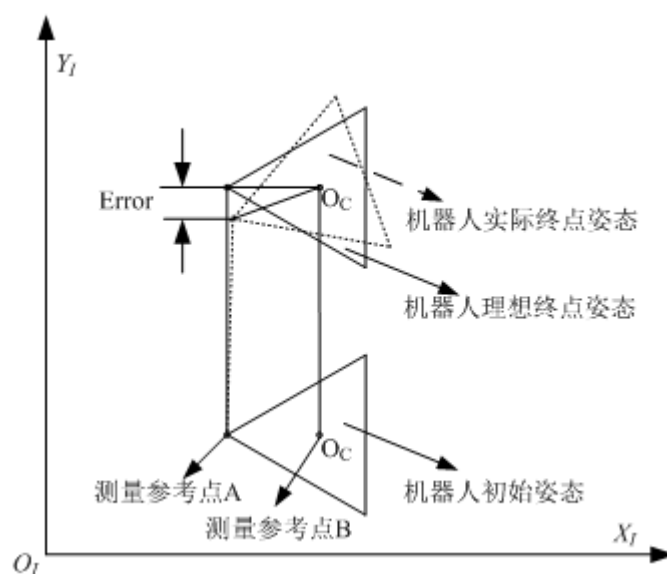


图 5.11 测量误差示意图

5.3.2 编码器误差校正

依据 3.4.2 节中的相关论述，设计实现了以下编码器误差校正实验，旨在确定机器人定位误差模型中与里程计不确定性相关的参数。

在相同的实验环境下，以不同的速度控制机器人运行直线路径 AB，A 点姿态为 $(0,0,0)$ ，B 点姿态为 $(5000,0,0)$ 。根据前文所述的在样机实验中，数据测量基准对实

验结果的影响分析,选取如图 5.10 所示的数据测量基准。机器人分别以 0.4m/s、0.8m/s、1.2m/s 的速度运行,上述实验所得数据如下表所示:

表 5.2 编码器实验测量结果

V/(m/s)	1	2	3	4	5	6	7	8	9	10	平均值
0.4	5036	5032	5038	5033	5031	5033	3030	5031	5030	5033	5032.7
0.8	5035	5038	5038	5039	5034	5036	5038	5033	5035	5036	5036.2
1.2	5045	5036	5045	5047	5040	5044	5040	5040	5042	5046	5042.5

因为在机器人的运动过程中编码器的误差主要来源于脉冲数的减少,所以机器人定位系统“认为”的距离要比实际的距离大。由表中的数据分析可知,机器人实际走过的距离总是大于预期的距离,实验结果符合上述分析。

由实验结果分析可知,机器人在不同的速度下,编码器的累计性误差有所不同。机器人运动的速度越快,编码器的累计误差越大。

5.3.3 陀螺仪及几何中心偏差测量实验

1、陀螺仪积分误差实验

本实验主要用于测量机器人在运动期间陀螺仪积分的不确定性参数。实验的方法如下:陀螺仪由初始的 0° 开始,先顺时针转动到 -90° ,然后逆时针转动到 90° ,最后顺时针转回 0° 。期间每隔 10° 采集一次积分值,最终的积分误差分布如图 5.12 所示。

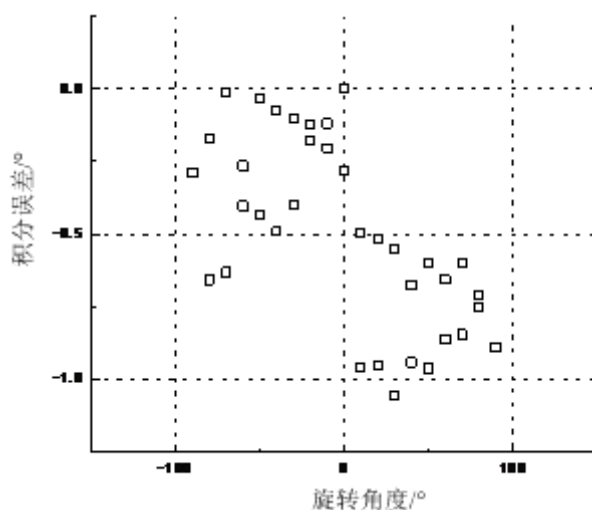


图 5.12 陀螺仪积分误差

由上图可以看出,在 $\pm 90^\circ$ 的测试范围内陀螺仪积分误差主要分布在 0° 至 -1° 之间,满足机器人的使用要求。

2、机器人局部参考框架与机器人几何中心的距离偏差测量

在给定三个驱动轮相同的速度的情况下，机器人底盘在绕着自己的几何中心运动。如果机器人的局部参考框架的原点与几何中心存在一定的距离，那么机器人的底盘做自旋运动时，机器人定位的结果一定是按照下述的规律变化的。

$$\begin{cases} x = l_{RC} \cos(\omega t) \\ y = l_{RC} \sin(\omega t) \end{cases} \quad (5-2)$$

上式中， l_{RC} 即为两点之间的距离， ω 为机器人自旋运动的角速度。

实际的实验中，多次实验取坐标变化的最大值，再对这些最大值取平均处理即可确定 l_{RC} 的值。给定机器人的自旋角速度 $\omega = 0.2 \text{ rad/s}$ ，测得的结果如表 5.3 所示。

表 5.3 定位中心偏移实验测量结果表（单位：mm）

	1	2	3	4	5	平均值
X 最大误差	22	20	18	23	21	20.8
Y 最大误差	24	18	19	22	20	20.6

表中数据显示，该样机实验使用的机器人局部参考框架与机器人几何中心的距离偏差在 20mm 左右。

5.3.4 样机重复定位实验

上节中的所有校正都是在机器人姿态角固定的情况下进行校正的，根据 3.3 节中对机器人的定位误差模型的分析，机器人姿态角偏差对最终的定位结果有重要的影响。样机实验是在实验环境中选取如图 5.10 所示的初始点。此时，3.4 节所述的机器人真实参考框架为 XOY 。给定机器人由初始点（0,0,0）至终点（2000,2000,0）的一条直线路径进行多次实验。实验结果如表 5.4 所示。

表 5.4 编码器校正前重复定位实验结果（单位：mm）

	1	2	3	4	5	平均值
X	2030	2028	2035	2033	2030	2031.2
Y	2010	2015	2008	2010	2010	2010.6

使用上述的测量结果对相应编码器的里程不确定性系数进行校正，并在此按照前文所述的实验条件重新进行实验，得到如表 5.5 所示的测试结果。比较重复定位的结果显示，在校正里程计的误差后，定位精度有了明显的提升。

表 5.5 编码器校正后重复定位实验结果 (单位: mm)

	1	2	3	4	5	平均值
X	2005	2007	2004	2006	2007	2006
Y	2000	2006	2008	2010	2002	2005

5.3.5 机器人样机路径跟踪实验

本实验所采集的数据以机器人自身的定位信息为参考, 并不能代表真实的实时位姿。由于机器人对定位精度对初始的姿态以及真实的参考框架比较敏感, 难以测得真实的实时定位数据。而该实验环节主要研究机器人运动控制算法的性能, 主要考虑机器人在运动过程中的实时位姿误差的变化情况, 而不关心其在实际的全局参考框架下的位姿, 所以这种方案是合理的。

1、机器人姿态调整实验

机器人的姿态调整的实验主要是结合第三章中的机器人定位与路径跟踪算法, 在机器人定点控制时人为施加姿态角的干扰, 通过实时反馈的姿态角误差来分析 PID 控制器的响应情况。机器人在不同的 PID 参数时, 姿态角误差的输出如图 5.13 所示。

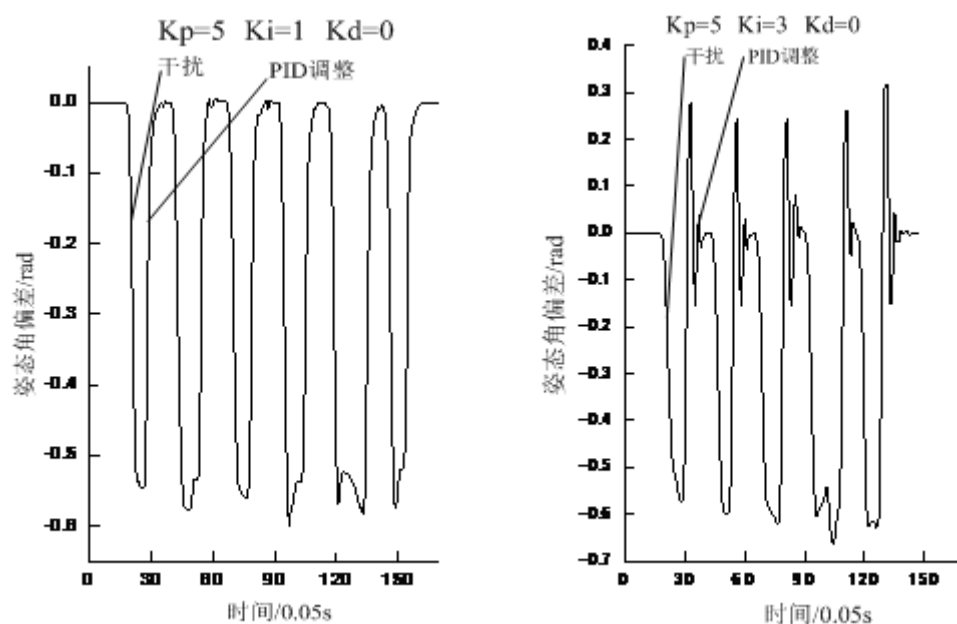


图 5.13 不同 PID 参数的姿态角调整

由上图可知, 每次人为施加约 0.6rad 的干扰。对比左右两图可以发现, PID 控制器的积分参数对机器人的姿态角调整的影响较大, 积分参数加大以后, 姿态角的调整出现了明显的震荡。

2、机器人位置调整实验

机器人的位置调整实验同样是在机器人定点控制时人为施加距离干扰(约 200mm)，通过采集控制器输出的距离误差的绝对值来分析不同 PID 控制器参数对位置调整的影响。实验结果如图 5.14 所示。

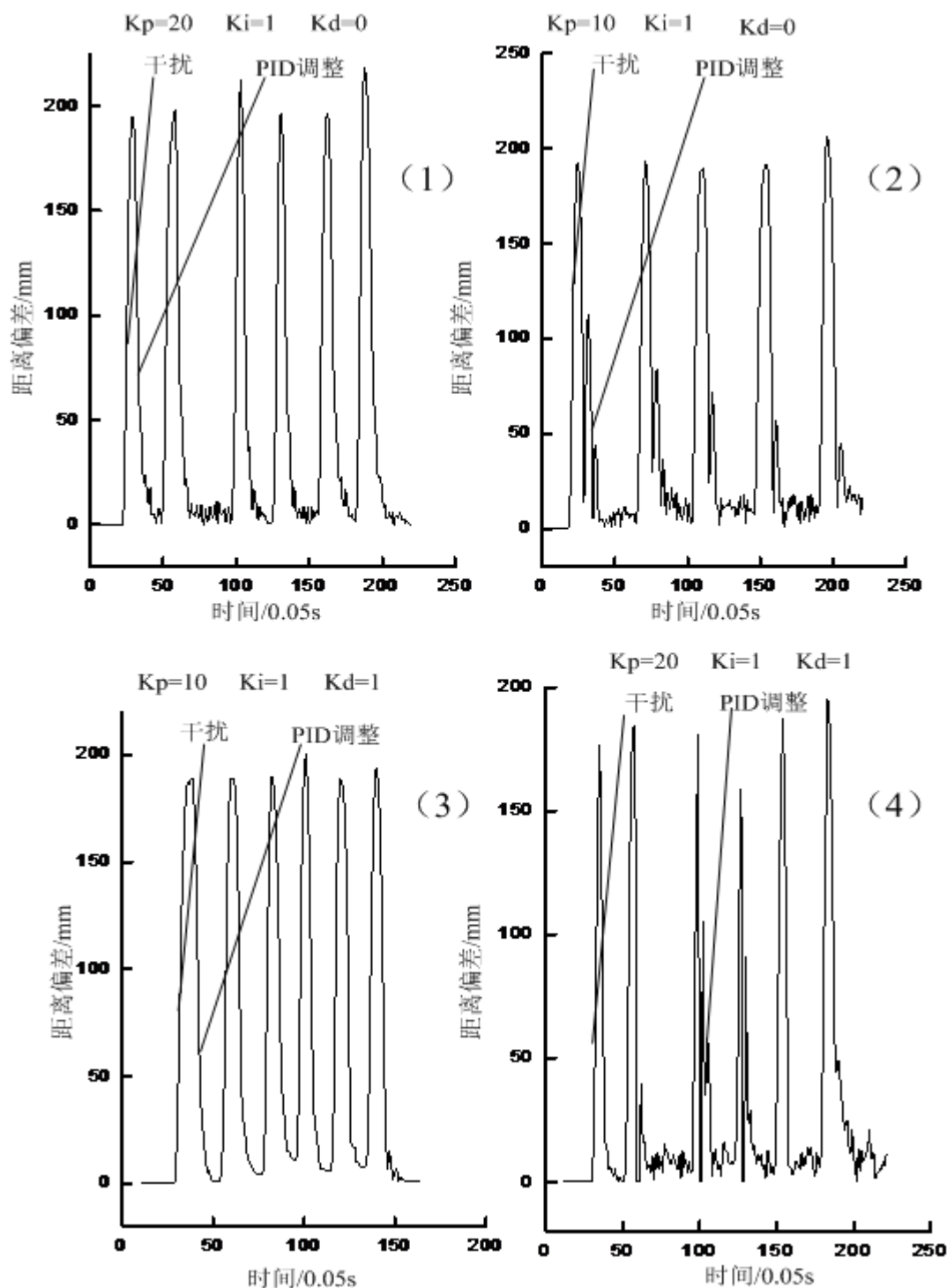


图 5.14 不同 PID 参数的位置调整

对比图 5.14 中的 (1) 和 (2) 发现当 K_d 的值较大时, 距离调整较快, 但是仍存在明显的波动。对比 (1) 和 (3) 发现, 当加入适当的微分调节时, 可以有效的改善控制器的调节性能。对比 (1) 和 (4) 可知, 在比例系数较大的情况下, 与 (2) 相同的微分系数并不能有效改变调节效果。

3、机器路径跟踪实验

图 5.15 为机器人分别以 0.8m/s 和 1.5m/s 的速度由初始点 $(0,0,0)$ 沿直线运动到目标点 $(2000,2000,0)$ 的轨迹图。由图可知, 当机器人的运行速度较小时比速度较大时的轨迹波动明显。因为机器人路径跟踪算法的实现基于机器人运动学模型, 所以在具有相同 PID 参数的情况下, 机器人误差校正的时间是固定的, 当机器人速度提高时机器人实际的轨迹会变得平缓。因此, 图 5.15 所示的轨迹跟踪结果符合实际的运动情况。

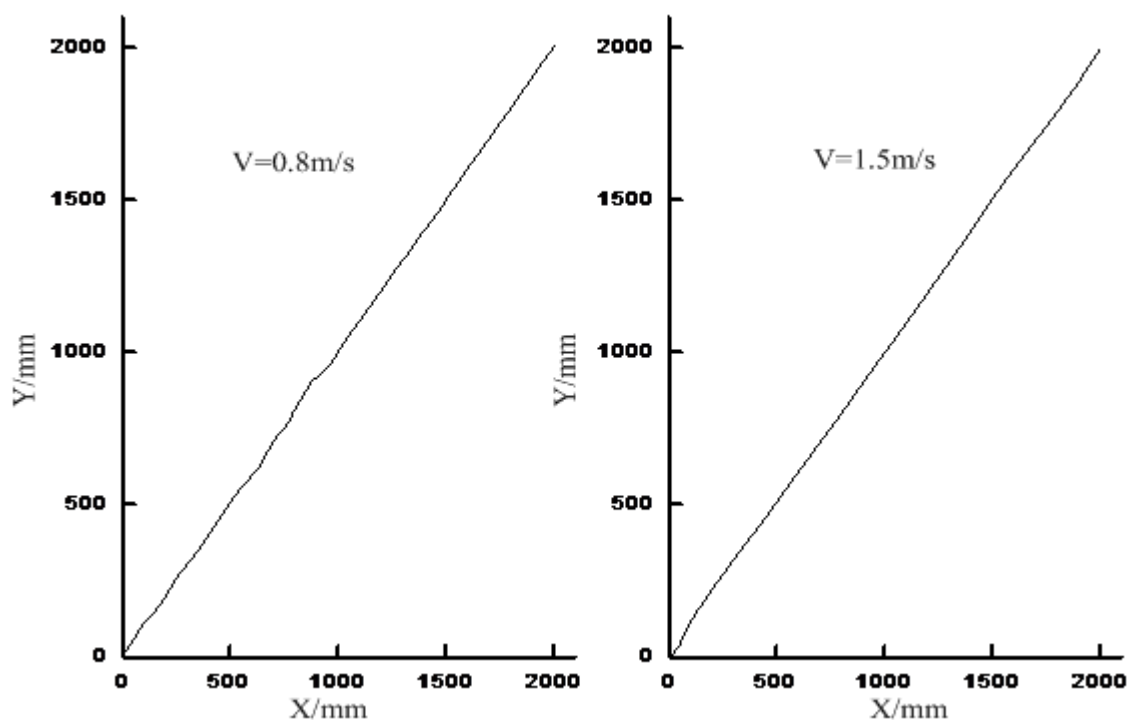


图 5.15 机器人样机直线路径跟踪轨迹

图 5.16 为机器人在两种速度下距离跟踪误差随时间的变化情况, 结果显示机器人在两种速度下的距离误差在 40mm 以内, 是可以接受的范围。同时, 对比两图不难发现, 机器人在较高的速度下, 距离偏差的变化频率较小, 这也证实了对图 5.15 相关分析。

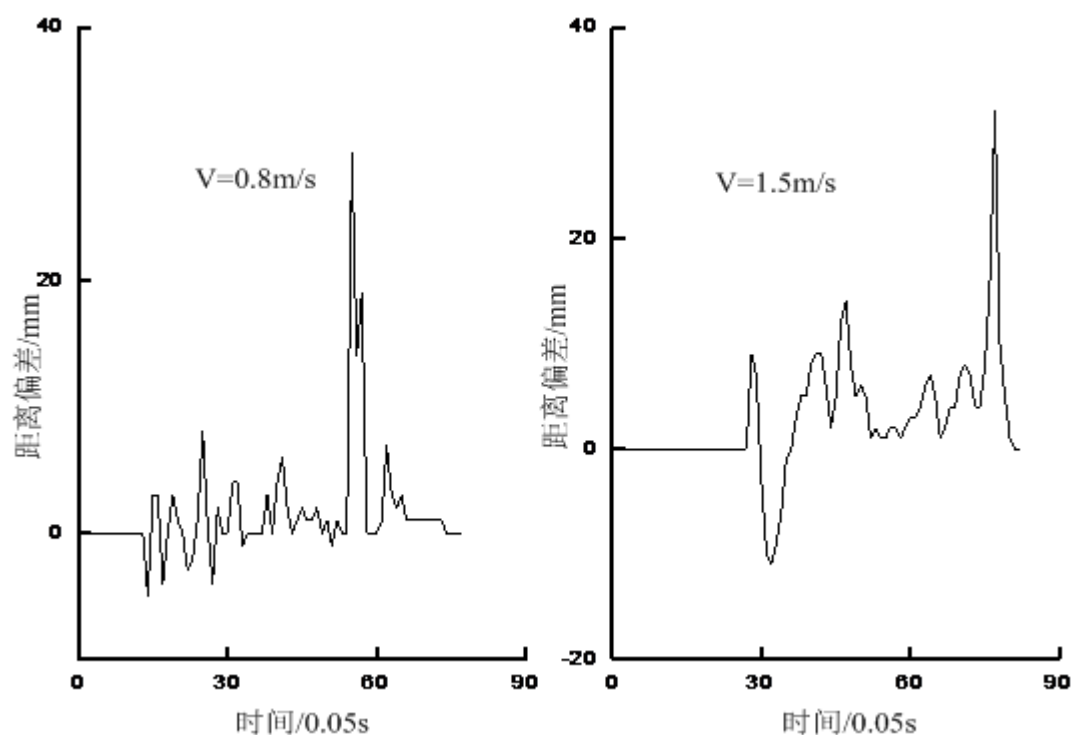


图 5.16 机器人样机直线路径跟踪距离偏差

图 5.17 为机器人在不同速度下跟踪直线轨迹时姿态角随时间的变化情况,对比可以发现,机器人在速度较高时,如果 PID 的参数调节不当,姿态角会产生明显的变化,这种变化主要是机器人各个驱动轮的加速较快,并且与接触地面摩擦力不足所致。

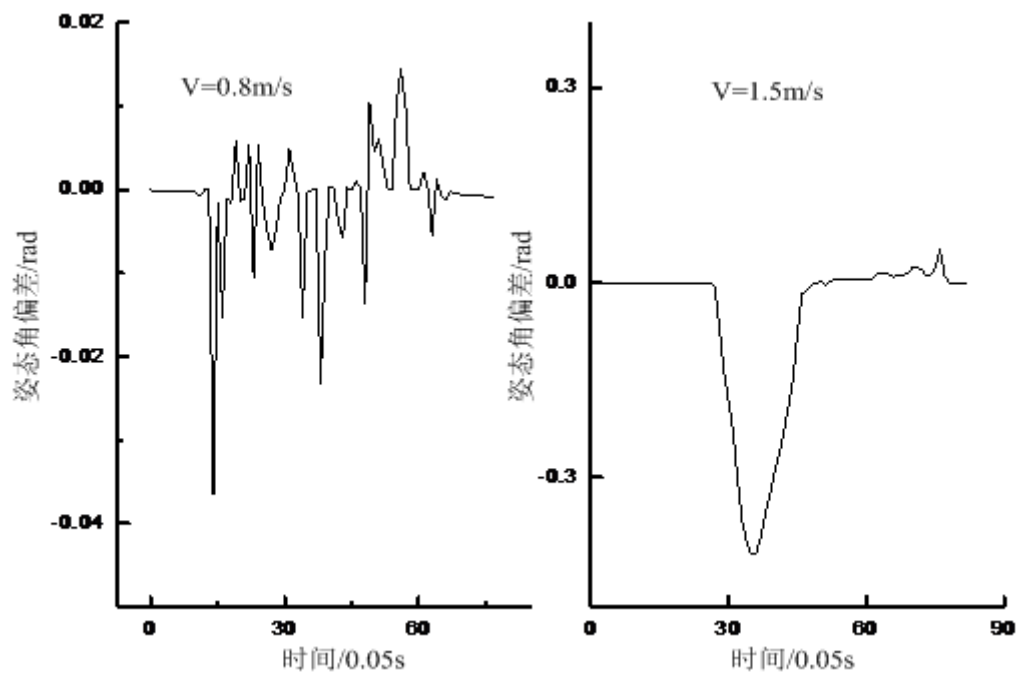


图 5.17 机器人样机直线路径跟踪姿态角偏差

图 5.18 为机器人跟踪圆弧路径的轨迹,实验结果表明机器人在基于运动学的几何路径跟踪算法的驱动下可以完成圆弧轨迹的跟踪。

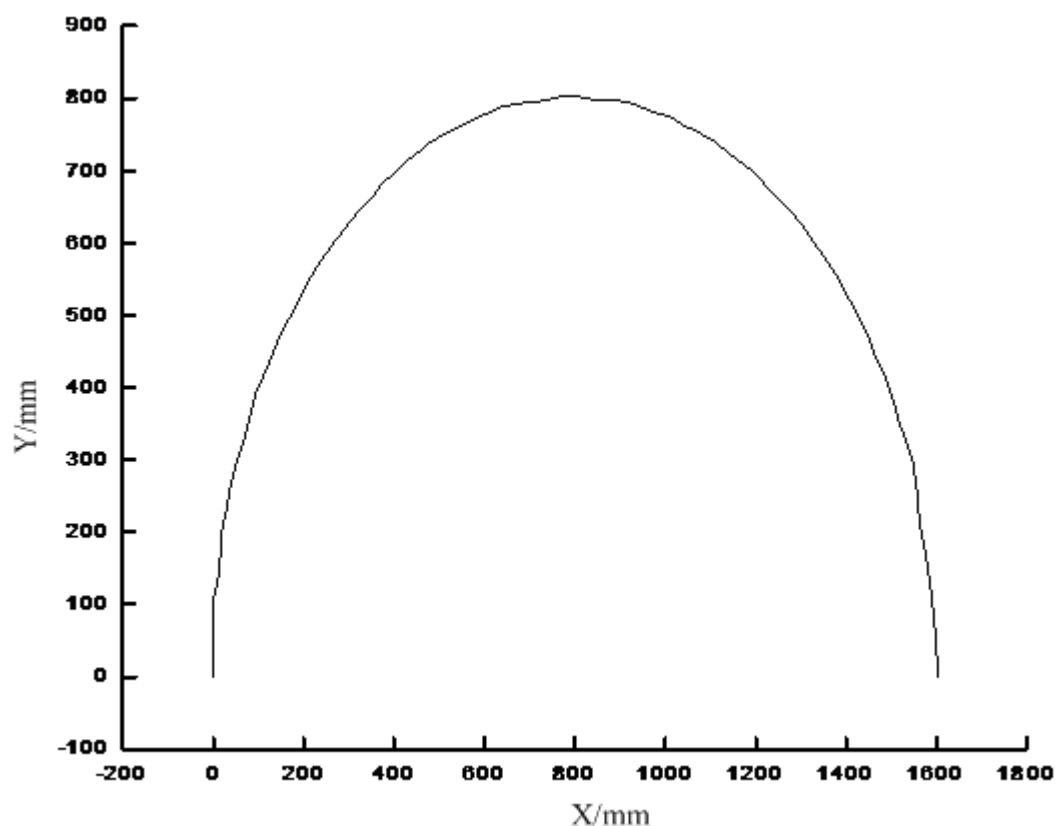


图 5.18 机器人样机圆弧路径跟踪轨迹

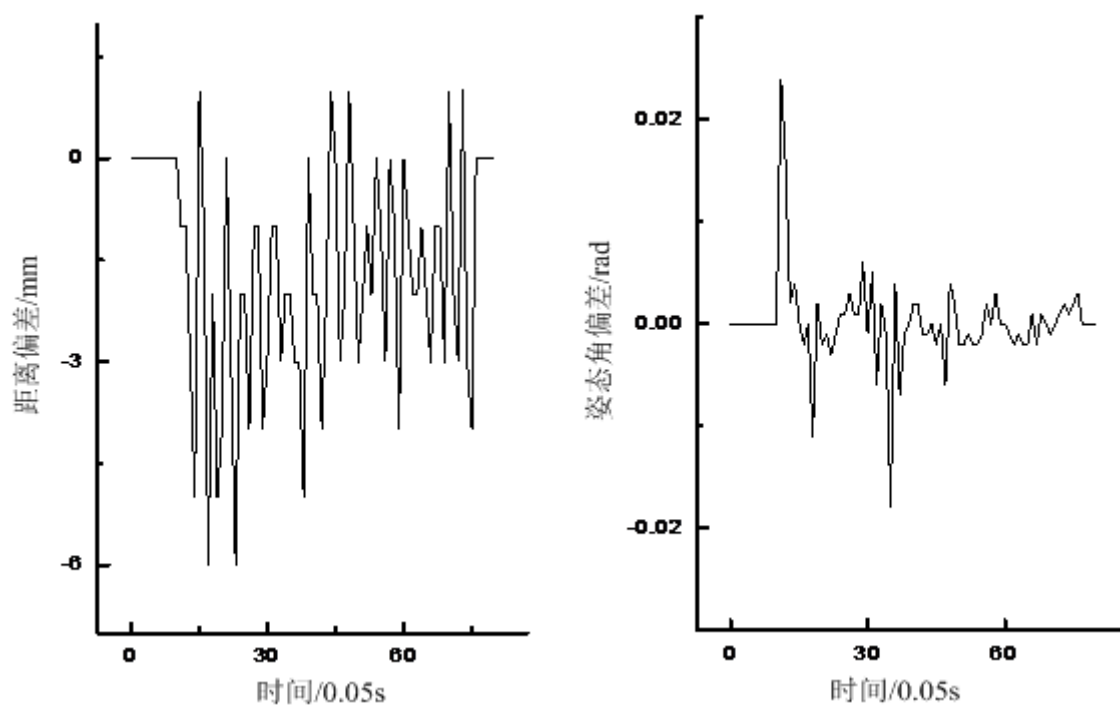


图 5.19 机器人样机圆弧路径跟踪距离偏差(左图)和姿态角偏差(右图)

图 5.19 为机器人在跟踪圆弧路径时，距离偏差和姿态角偏差随时间的变化情况。距离偏差最大跟踪误差为 6mm，姿态角跟踪偏差在 1° 左右，两者均是可以接受的范围。比较图 5.16 和图 5.19 发现，机器人在跟踪圆弧路径时，距离偏差的平均值为负值，而跟踪直线路径时，距离跟踪误差分布在零点附近。以上的结果与机器人的 PID 位姿误差校正控制的具体实现相关。

5.4 本章小结

本章的主要完成了机器人定位及几何路径算法的仿真和样机实验验证。首先，通过 Matlab/Simulink 仿真，验证了机器人几何路径跟踪算法的正确性。在样机实验方面首先分别介绍了实验环境相关参数和实验数据测量与采集方案。重点描述了定位随动轮系中里程计的误差校正过程。结合机器人自身的直线路径跟踪算法，设计了机器人在不同运行速度和随动轮系正压力等条件下的编码器累计误差。通过对定位模型相关参数的校正，实现了机器人短时间内较高的定位精度。机器人的路径跟踪实验，验证了路径跟踪算法的正确性。

结 论

全方位轮式移动机器人相对普通轮式移动机器人具有更高的运动灵活性和机动性。基于里程计和陀螺仪的定位方案能够在短时间内保持较高的定位精度，以瑞典轮作为底盘驱动轮的结构方案能够有效地减小系统结构和控制的复杂程度。因此，全向移动机器人在日常生产生活中的一些特定场合具有很好的应用前景。本文以基于瑞典轮的全向移动机器人为研究对象，对其定位及路径跟踪算法及嵌入式控制系统的实现进行研究：

(1) 设计实现了基于瑞典轮和里程计的正交随动轮系，并结合陀螺仪通过航位推算的方法得出了机器人的位姿更新方程。

(2) 在机器人全场定位的基础上，设计实现了基于解析法的机器人几何路径跟踪算法。分析了直线和圆弧路径的路径跟踪误差实时解算方法，设计了 PID 误差校正控制器。

(3) 设计实现了以 STM32 系列控制器为核心的机器人嵌入式控制系统。其中主控制器使用具有 DSP 功能的 STM32F407 微控制器，用来完成机器人定位及运动控制算法中的复杂计算。使用 STM32F103 作为陀螺仪数据采集模块和直流伺服电机模块的子控制器，设计实现了相关的软硬件并做了优化。使用双端口 RAM 和 FSMC 总线控制器实现了机器人控制系统多机并行通信。基于嵌入式操作系统 $\mu\text{C}/\text{OS-III}$ 构建了机器人控制系统软件框架。样机的正确运行验证了上述控制系统设计的有效性。

(4) 通过 Matlab/Simulink 仿真，验证了机器人几何路径跟踪算法的正确性。通过样机实验依次测量了传感器自身误差、机器人定位模型误差（随动轮系的偏差角）以及机器人在不同条件下的重复定位误差。通过对定位模型相关参数的校正，实现了机器人短时间内较高的定位精度。机器人的路径跟踪实验，验证了路径跟踪算法的正确性。

机器人的导航系统是极复杂的系统工程，它涉及的范围涵盖了机械动力学、自动控制理论、软件工程、人工智能和图像处理等多学科内容。目前国内的技术相比国外还有一定差距，需要不断的努力和探索。从理论到试验样机，再到实际产品，是一个长期不断努力试验的过程，需要对各方面进行研究和改进。基于瑞典轮的全向移动机器人已做了初步的探索，但是由于论文所涉及的内容较多，系统仍有很多不足之处，在此提出对进一步的研究工作及展望：

(1) 机器人的运动控制算法是基于运动学模型设计实现的，而对于一个完善的机器人控制系统而言，有必要研究机器人的动力学特性，并将其运用到机器人的路径跟踪算法中。

(2) 机器人的定位系统结构单一，里程计的累计误差及陀螺仪的漂移无法消除，因此机器人无法保持长时间的定位精度要求。结合图像处理及模式识别等相关知识，开发机器人的辅助定位系统是课题进一步的研究重点。

参考文献

- [1] 蔡自兴.机器人学.清华大学出版社,2000:1-20.
- [2] 刘华军,杨静宇,陆建峰,等.移动机器人运动规划研究综述.中国工程科学,2006,8(1):85-94.
- [3] 杨晶东.移动机器人自主导航关键技术研究.哈尔滨工业大学,2008:1-12.
- [4] 王志文,郭戈.移动机器人导航技术现状与展望.机器人,2003,25(5):470-474.
- [5] Borenstein J, Feng L. Gyrodometry: A new method for combining data from gyros and odometry in mobile robots//Robotics and Automation, 1996. Proceedings. 1996 IEEE International Conference on. IEEE, 1996, 1: 423-428.
- [6] Mataric M J. Integration of representation into goal-driven behavior-based robots. Robotics and Automation, IEEE Transactions on, 1992, 8(3): 304-312.
- [7] Moravec H P, Elfes A. High resolution maps from wide angle sonar//Robotics and Automation. Proceedings. 1985 IEEE International Conference on. IEEE, 1985, 2: 116-121.
- [8] Henrik I Christensen, David Austin. Nomadic robotics . <http://drobot.sourceforge.net/faq.html>.
- [9] 王卫华,熊有伦,孙容磊.测程法系统误差的测量与校核.机器人,2004,05:454-460.
- [10] 王卫华,熊有伦,孙容磊.一种移动机器人轮子打滑的实验校核方法.机器人,2005,27(3):197-202.
- [11] 刘玉鹏,海丹,郑志强.基于测程法的四轮全向移动机器人自定位方法研究.2005 中国机器人大赛论文集,2005:17-22.
- [12] 王一治,常德功.Mecanum 三轮多向行走系统结构与运动原理.世界科技研究与发展,2008,30(6):803-806.
- [13] 刘洲,吴洪涛.Mecanum 四轮全方位移动机构运动分析与仿真.中国制造业信息化,2011,40(3):43-46.
- [14] 王一治,常德功. Mecanum 四轮全方位系统的运动性能分析及结构形式优选.机械工程学报,2009,05:307-310.
- [15] Sciavicco L, Villani L. Robotics: modelling, planning and control[M]. Springer, 2009.
- [16] Tlale N, de Villiers M. Kinematics and dynamics modelling of a mecanum wheeled mobile platform//Mechatronics and Machine Vision in Practice, 2008. M2VIP 2008. 15th

- International Conference on. IEEE, 2008: 657-662.
- [17]Leng C, Cao Q. Velocity analysis of omnidirectional mobile robot and system implementation//Automation Science and Engineering, 2006. CASE'06. IEEE International Conference on. IEEE, 2006: 81-86.
- [18]Han K L, Choi O K, Kim J, et al. Design and control of mobile robot with Mecanum wheel//ICCAS-SICE, 2009. IEEE, 2009: 2932-2937.
- [19]黄超.模块化移动机器人控制系统及运动算法的研究.哈尔滨工业大学,2010:1-5.
- [20]邓正隆编著.惯性技术.哈尔滨工业大学出版社,2006:1-15.
- [21]Siegwart R, Nourbakhsh I R, Scaramuzza D. Introduction to autonomous mobile robots. MIT press, 2011:182-191.
- [22]Indiveri G. Swedish wheeled omnidirectional mobile robots: kinematics analysis and control. IEEE transactions on robotics, 2009, 25(1): 164-171.
- [23]Ren X, Feng Z, Chang H, et al. Kinematics modeling and analysis for three-wheel omnidirectional mobile robot//Intelligent Control and Automation, 2008. WCICA 2008. 7th World Congress on. IEEE, 2008: 2608-2613.
- [24]Y. Byungju, K. Wheeluk, "The Kinematics for Redundantly Actuated Omni-directional Mobile Robots," in Proc. IEEE Int. Conf. Robotics and Auto., San Francisco, April 2000, pp.2485-2492
- [25]全方位移动机器人导论.科学出版社,2010:150-186.
- [26]曹其新,张蕾.轮式自主移动机器人.2012:1-50.
- [27]冷春涛,曹其新.四轮全方位移动机器人各向相异性研究.智能系统学报,2007,2(3):45-51.
- [28]熊蓉,张翮,褚健,等.四轮全方位移动机器人的建模和最优控制.控制理论与应用,2006,23(1):93-98.
- [29]赵冬斌,易建强,邓旭.全方位移动机器人结构和运动分析.机器人,2003,25(5).
- [30]Martinelli A, Siegwart R. Estimating the odometry error of a mobile robot during navigation. other words, 2003, 1(3):394-398.
- [31]Kelly A. General solution for linearized systematic error propagation in vehicle odometry//Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on. IEEE, 2001, 4: 1938-1945.

- [32] Kelly A. Linearized error propagation in odometry. The International Journal of Robotics Research, 2004, 23(2): 179-218.
- [33] Borenstein J. Experimental results from internal odometry error correction with the omnimate mobile robot. Robotics and Automation, IEEE Transactions on, 1998, 14(6): 963-969.
- [34] Borenstein J, Feng L. Correction of systematic odometry errors in mobile robots//Intelligent Robots and Systems 95.'Human Robot Interaction and Cooperative Robots', Proceedings. 1995 IEEE/RSJ International Conference on. IEEE, 1995, 3: 569-574.
- [35] Komoriya K, Oyama E. Position estimation of a mobile robot using optical fiber gyroscope (OFG)//Intelligent Robots and Systems' 94.'Advanced Robotic Systems and the Real World', IROS'94. Proceedings of the IEEE/RSJ/GI International Conference on. IEEE, 1994, 1: 143-149.
- [36] Overholt J, Hudak G, Iagnemma K, et al. Performance Analysis and Odometry Improvement of an Omnidirectional Mobile Robot for Outdoor Terrain. 2011.
- [37] 刘金琨.先进 PID 控制 MATLAB 仿真.电子工业出版社,2004:25-65.
- [38] 刘金琨.智能控制.电子工业出版社,2005:1-20.
- [39] 李啸,张洪钺,李骥.基于模糊 PID 的轮式移动机器人轨迹控制.机器人技术应用,2002,5:30-33.
- [40] 陈猛.基于 DSP 和模糊控制 PID 的稳定平台的设计.哈尔滨工程大学,2009:2-8.
- [41] Huang H C, Wu T F, Yu C H, et al. Intelligent fuzzy motion control of three-wheeled omnidirectional mobile robots for trajectory tracking and stabilization//Fuzzy Theory and it's Applications (iFUZZY), 2012 International Conference on. IEEE, 2012: 107-112.
- [42] Reznik L, Ghanayem O, Bourmistrov A. PID plus fuzzy controller structures as a design base for industrial applications. Engineering applications of artificial intelligence, 2000, 13(4): 419-430.
- [43] Qi T, Li C, Kai W, et al. Research on motion control of mobile robot with fuzzy PID arithmetic//Electronic Measurement & Instruments, 2009. ICEMI'09. 9th International Conference on. IEEE, 2009: 3-363-3-366.
- [44] L.D. Stephen, D.L. Brett, "Control of an Omni-directional Robotic Vehicle with Mecanum Wheels," in Proc. Nati. Telesystems Conf., vol.1,1991, pp.323-328

- [45]舒志兵,杜永红,黄益群等.最新运动控制技术进展.电气时代,2005,2(08):48-50.
- [46]廖晓钟刘向东编著.控制系统分析与设计.北京:清华大学出版社2008:1-20..
- [47]张瑞东.轮式移动机器人控制系统研究与实现.河北工程大学,2012:1-16.
- [48]秦磊,尤波.基于 LM628 芯片的三轴伺服控制器.哈尔滨理工大学学报,2003,8(4):44-47.
- [49]肖会芹,赖旭芝,曹桂秀.基于并行口的手臂机器人控制系统数据通信设计.计算机测量与控制,2006,14(2):247-249.
- [50]张凤登编著.现场总线技术与应用.北京:社会科学出版社2008:34-56.
- [51]饶怡欣,胥布工,匡付华.基于 CANopen 的电动执行机构远程监控主站的实现.计算机测量与控制,2010(2):373-375.
- [52]STM32 Reference Manual, STMicrocontroller,2009.6:1-733
- [53]Sloss A N, Symes D, Wright C, et al. ARM 嵌入式系统开发——软件设计与优化[J].2005:94-146.
- [54]Labrosse J J. MicroC/OS-II: The Real Time Kernel. CRC Press, 2002:2-316.
- [55]LM628/LM629 Precision Motion Controller, National Semiconductor,1999.11:1-24
- [56]谭浩强.C 程序设计.清华大学出版社有限公司,2005:1-364.
- [57]Reek K A.C 和指针.人民邮电出版社,2003:1-445.
- [58]诸静.机器人与控制技术.杭州:浙江大学出版社,1991.10:35-64.
- [59]周新伦,关绮玲.机器人.上海:复旦大学出版社,1994.8:1-23.
- [60]孙迪生,王炎.机器人控制技术.北京:机械工业出版社,1997.4:1-12.
- [61]陈哲,吉熙章.机器人技术基础.北京:机械工业出版社,1997.10:1-25.
- [62]孟繁华.机器人应用技术.哈尔滨:哈尔滨工业大学出版社,1989.6
- [63]戴博,肖晓明,蔡自兴.移动机器人路径规划技术的研究现状与展望.控制工程,2005,12(3),198-201
- [64]孟祥荔,曹作良.基于 GPS 的移动机器人多传感器定位信息融合.天津理工大学学报,2007,22(6):47-49.
- [65]Leow, Y.P.; Low, K.H.; Kinematic modelling and analysis of mobile robots with omni-directional wheels. Loh, W.K.; Control, Automation, Robotics and Vision, 2002. ICARCV 2002. 7th International Conference on Volume 2, 2-5 Dec.2002 Page(s):820 - 825 vol.2

- [66]Muir, P.; Neuman, C.; Kinematic modeling for feedback control of an omnidirectional wheeled mobile robot. Robotics and Automation. Proceedings. 1987 IEEE International Conference on Volume 4, Mar 1987 Page(s):1772 - 1778
- [67]Betourne, A.; Dynamic modelling and control design of a class of omnidirectional mobile robots. Campion, G.; Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on Volume 3, 22-28 April 1996 Page(s):2810 - 2815 vol.3
- [68]Leng C, Cao Q. Velocity analysis of omnidirectional mobile robot and system implementation//Automation Science and Engineering, 2006. CASE'06. IEEE International Conference on. IEEE, 2006: 81-86.
- [69]Li T S, Chen C Y, Hung H L, et al. A fully fuzzy trajectory tracking control design for surveillance and security Robots//Systems, Man and Cybernetics, 2008. SMC 2008. IEEE International Conference on. IEEE, 2008: 1995-2000.
- [70]Huang H C, Wu T F, Yu C H, et al. Intelligent fuzzy motion control of three-wheeled omnidirectional mobile robots for trajectory tracking and stabilization//Fuzzy Theory and it's Applications (iFUZZY), 2012 International Conference on. IEEE, 2012: 107-112.
- [71]H. C. Huang and C. C. Tsai, “FPGA implementation of an embedded robust adaptive controller for autonomous omnidirectional mobile platform,” IEEE Transactions on Industrial Electronics, vol. 56, no. 5, pp. 1604-1616, 2009.
- [72]Huang H C, Wu T F, Yu C H, et al. Intelligent fuzzy motion control of three-wheeled omnidirectional mobile robots for trajectory tracking and stabilization//Fuzzy Theory and it's Applications (iFUZZY), 2012 International Conference on. IEEE, 2012: 107-112.
- [73]Liu Y, Wu X, Zhu J J, et al. Omni-directional mobile robot controller design by trajectory linearization. dynamics, 2004, 2: 2-2.

攻读硕士学位期间发表的论文和取得的科研成果

- [1] 第三发明人.一种多位置承力伸缩机构,发明专利,2014年3月.公开号:
CN103640031A
- [2] 第三发明人. 一种硬币抛射装置,发明专利,2014年1月.公开号: CN103505861A
- [3] 第九发明人. 液压驱动二自由度仿生脊柱关节模块,发明专利,2013年9月.公开号:
CN103302677A
- [4] 第九发明人. 液压驱动二自由度仿生脊柱关节模块,实用新型,2013年9月.公开号:
CN203344063U

致 谢

岁月如梭，六年的生活的欢笑和泪水即将在 2014 年 7 月结束！在此首先感谢导师陈东良副教授这几年来的关心教导，在从大四保研和研究生的共三年的时间里，不仅从他的身上学到了很多电子电路、机械结构方面的专业知识。陈老师渊博深锐的学识、泰然大方的处事风格，一丝不苟的做事态度，让学生以此为榜样受益终生！

感谢王茁教授对我的论文细致入微的评阅。感谢潘文林老师一直以来无私的关心和帮助！

感谢我的同学张志远、刘祥、王成华、刘琦等，感谢我的师弟杨建、李爽、王宏伟、王洪等，感谢师妹黄新禹、王静，感谢我的室友申建平、王晓宇、郭丙岐、孟凡森，大家在一起欢声笑语、携手共进的美好时光令我此生难忘！

感谢陪我一起走过机器人电视大赛国内选拔活动的所有队员，我们为了与机器人相关的同一个梦想，拼搏奋斗了整整一年。在这期间有欢笑、泪水，更有争执和误解，但是我们最终一起走过困境。大学的生活因此而变得更加丰富精彩，同时也让我在大学生科技创新方面小有收获！

感谢我的母校哈尔滨工程大学，为我的成人和成才提供了优良的环境！学校日趋完善的软硬件条件是我生活学习的最坚实保障！

最后特别要感谢我父母情重如山的养育之恩！你们永远是我温馨的港湾！

“古之立大事者，不惟有超世之才，亦必有坚忍不拔之志”，我欲乘风破浪、披荆斩棘、开拓进取创就一番事业，以回报我所有的老师、同学、亲人、朋友，祝你们身体健康，生活幸福！

祝愿母校的未来更美好！祝愿祖国的明天更强盛！