

长沙理工大学

硕士学位论文

三轮全向移动机器人运动控制研究

姓名：谢志诚

申请学位级别：硕士

专业：控制理论与控制工程

指导教师：李益华

20100401

# 摘 要

足球机器人涉及到多学科、多领域，为研究多智能体系统、人工智能等理论提供了一个很好的研究平台，引起了越来越多的科研机构和大专院校的关注，也吸引了越来越多的科研人员和学者的参与。

论文主要对 RoboCup 中型组足球机器人——三轮全向驱动机器人的运动控制进行了研究。首先建立了三轮全向移动机器人非线性系统模型，通过分析它的运动特性，采用基于运动学模型的分层控制方案实现机器人的运动控制。为了验证底层运动控制系统和控制策略的合理性，在分析无刷直流电机数学模型的基础上，利用 Matlab/Simulink 建立了无刷直流电机转速、电流双闭环控制系统的仿真模型，通过仿真得到了与理论分析相一致的仿真试验结果，证明了该控制方案的可行性。其次，本文设计了三轮全向移动机器人的运动控制算法，包括基于 Lyapunov 直接法的姿态稳定控制算法和基于输入变换解耦的轨迹跟踪控制算法，使系统具有良好的全局稳定性，实现了移动机器人的位置和方位的跟踪控制。仿真实验中考虑了驱动电机的动态特性，检验了控制算法的鲁棒性和有效性。

最后，根据 RoboCup 中型组足球机器人底层控制系统的基本任务，采用双 DSP 构架，选择高性能 DSP 处理器 TMS320LF2407A 作为核心进行底层运动控制系统的设计。硬件部分，按照模块化的思想，主要设计了以 TMS320LF2407A 为核心的控制电路和无刷直流电机驱动模块；软件部分，在硬件平台基础上编制了各个模块的软件程序，给出了各模块的程序框图。

**关键词：**三轮全向机器人；无刷直流电机；运动控制；姿态稳定；轨迹跟踪

# ABSTRACT

Robot soccer, referring to multi-domain and multi-disciplinary, is a good test bed for multi-agent system and artificial intelligence theory. So far, it has been focused by many research institutes and colleges, more and more researchers and scholars pay attention to it.

This thesis presents the research on motion control system of omni-directional mobile robots with three wheels for the RoboCup's Middle Size Robot League. The thesis, firstly, builds the dynamic models of omni-directional robots, and decides to use the control based on dynamics model to achieve motion control of soccer robots by analyzing the characteristics of its movement. In order to verify the whole control system and the control strategy, a simulation model of motor control system had been founded on base of BLDCM mathematical model with Matlab/Simulink. The results of simulation were consistent with the theoretical analysis, which proved the control system was feasible. Secondly, some control algorithms for omni-directional mobile robots with three wheels are designed, including posture stabilization controller base on the input decoupling and the trajectory tracking controller base on Lyapunov stability theory, so the tracking performance of system has the global stability and the independent control of both position and orientation for a mobile robot is realized. Simulation experiments validate the stabilities of all control algorithms, and the robustness and practicability of all algorithms are also verified.

Finally, the motion control system used a double DSP structure, and DSP microprocessor TMS320LF2407A with high advanced capacity is used as the chief chip to design the bottom movement control system from the basic tasks of RoboCup's middle-league soccer robots. For hardware part, the control system of the microprocessor and BLDCM controller modular are designed according to the modular thought. For software part, the whole modules are programmed in foundation of the hardware platform, and the program flow charts of the various modules are given.

**Key Words:** three-wheel omni-directional mobile robot; brushless DC motor; motion control; posture stabilization; trajectory tacking

# 长沙理工大学

## 学位论文原创性声明

本人郑重声明：所呈交的论文是本人在导师的指导下独立进行研究所取得的研究成果。除了文中特别加以标注引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写的成果作品。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后果由本人承担。

作者签名：谢志诚 日期：2010年 5 月 20 日

## 学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权长沙理工大学可以将本学位论文的全部内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

本学位论文属于

1、保密 ☐，在 \_\_\_\_\_ 年解密后适用本授权书。

2、不保密 ☒。

(请在以上相应方框内打“√”)

作者签名：谢志诚 日期：2010年 5 月 20 日

导师签名：李金华 日期：2010年 5 月 20 日

# 第一章 绪论

RoboCup 联盟（起初称作 Robot World Cup Initiative）是一个国际性研究和教育组织，它通过提供一个标准问题来促进人工智能和智能机器人的研究。这个领域应该可以集成并检验很大范围内的技术，同时也可被用作综合的面向工程应用的教育。为了这个目的，RoboCup 联盟选择了足球比赛作为一个基本领域，并组织了国际上级别最高、规模最大、影响最广泛的机器人足球赛事和学术会议——机器人足球世界杯及学术会议（The Robot World Cup Soccer Games and Conferences, 简称 RoboCup）。为了能让一个机器人球队真正能够进行足球比赛，必须集成各种各样的技术，包括自治智能体的设计准则、多主体合作、策略获取、实时推理、机器人学以及感知信息融合等<sup>[1]</sup>。对一个由许多快速运动的机器人组成的球队来说，RoboCup 是一项在动态环境下的任务。在软件方面，RoboCup 还提供了软件平台以便于研究。

在人工智能与机器人学的历史上，1997 年将作为一个转折点被记住。在 1997 年 5 月，IBM 的深蓝击败了人类国际象棋世界冠军，人工智能界四十年的挑战终于取得了成功。在 1997 年 7 月 4 日，NASA 的“探路者”在火星成功登陆，第一个自治机器人系统 Sojourner 释放在火星的表面上。与此同时，RoboCup 也朝开发能够战胜人类世界杯冠军队的机器人足球队走出了第一步。

## 1.1 课题来源及研究意义

本课题源于 2009 年 RoboCup 中型组(Middle Size Robot League)足球机器人竞赛，也称为 F-2000。针对中型组足球机器人的性能要求，结合长沙理工大学的足球机器人和当前机器人技术的发展，对三轮全向轮式移动足球机器人的运动控制系统进行了分析和研究。

移动机器人运动控制系统是机器人实现其控制任务的基础，它决定了轮式移动机器人性能的优劣。机器人运动控制系统是根据指令以及传感信息控制机器人作相应的运动或完成一定作业任务的装置<sup>[2]</sup>。随着信息化的进程、人工智能、计算机科学与技术、信号处理理论与方法等的迅速发展，移机器人需要处理的数据量也就越来越大，并且对控制精度、数据处理速度和实时性的要求也越来越高，这就对机器人运动控制系统的性能要求也就越来越高。

因此，对机器人运动控制系统进行研究开发，提高机器人运动控制的实时性和准确性，对机器人越来越复杂的运动要求，设计出更有效的控制算法，这将对推动移动机器人的发展是非常重要的<sup>[3]</sup>。

## 1.2 RoboCup 中型组足球机器人系统简介

RoboCup 中型组机器人足球比赛中，每个参赛机器人拥有各自独立的视觉系统、决策系统、运动控制系统，并且相互之间可进行无线通信。各参赛机器人在长度为 12 米至 18 米，宽度 8 米以上的场地上进行足球比赛，并且要求所有参赛的机器人必须是全自主的，即决策信息、传感器及运动控制由机器人自身处理，各机器人之间的相互协作由无线通信来实现。RoboCup 中型组机器人足球比赛规模大，场面壮观，每个机器人是一个独立的智能体，没有人为的干预，在比赛中各机器人相互协作或对抗，体现了机器人的智能。RoboCup 中型组机器人足球比赛系统如图 1.1 所示。

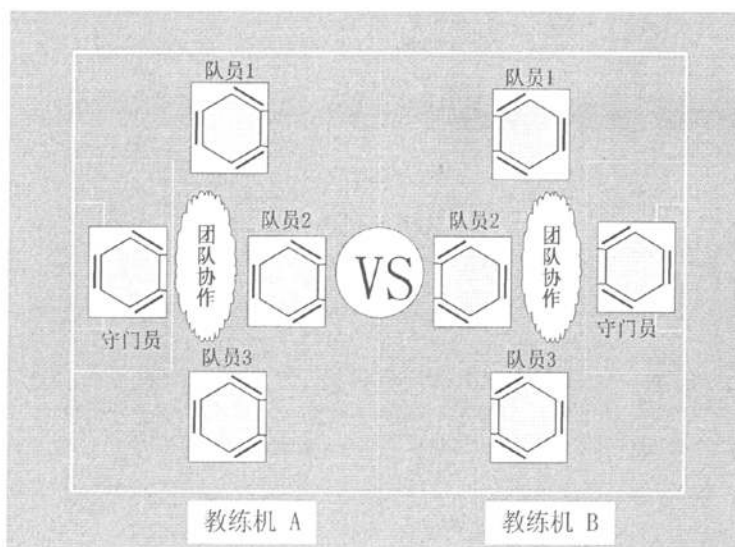


图 1.1 中型组机器人（4：4）足球比赛

比赛时足球机器人不但依靠自身的环境感知能力和系统决策进行比赛，利用无线通信系统与队友共享彼此的环境感知数据和行为状态。场外教练机融合各参赛机器人的环境感知结果，建立一个全局环境模型，分析场上的形势，再通过无线网络把融合后的环境模型发送给每个机器人，供其决策系统使用，避免单个机器人由于信息的局部性而进行盲目行动。

完整的 RoboCup 中型组足球机器人系统从功能上分一般由以下几个部分组成：视觉系统、决策系统、通信系统、运动控制系统。其内部结构如图 1.2 所示。

比赛时，机器人视觉系统主要由全景照相机组成，负责采集比赛场景图像，并对获得的数字图像进行图像处理与识别，将全局环境建模识别结果传送给决策系统。决策系统对比赛场上的形势和视觉系统的辨识结果进行分析，规划出机器人的运动轨迹，将动作指令传送到运动控制系统。运动控制系统按机器人接机器人接收到的动作指令控制机器人运行，或执行相应的动作。通信系统是机器人之

间相互协作以及和对方机器人进行对抗的基础,通过无线局域网与其他机器人进行信息交互。下面对中型组机器人系统的几个组成部分作简要的介绍。

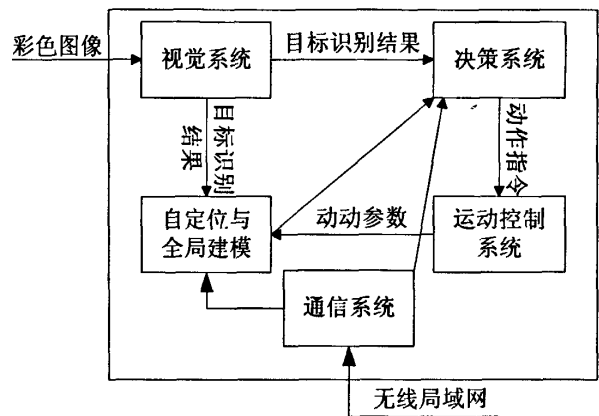


图 1.2 中型组足球机器人内部结构图

(1) 运动控制系统：运动控制是机器人控制的核心，它是机器人实现各种智能的基础，控制机器人的运动轨迹、姿态和相应的执行机构。机器人运动控制系统通过串行接口接收上位机发送的轮速、射门及控球等指令，控制机器人各驱动轮轮速、射门及控制装置的启停，并且将自身相应动作信息实时传给上位机，供上位机决策之用。

(2) 视觉子系统：使机器人具有视觉感知功能的系统。机器人视觉可以通过视觉传感器获取环境的二维图像，并通过视觉处理器进行分析和解释，进而转换为符号，让机器人能够辨识物体，并确定其位置。机器视觉子系统提供对执行某一特定任务相关的景物描述，它是机器人系统的“眼睛”。

(3) 决策子系统：决策系统根据现场的情况，如当前得分、谁控制球、对手的水平等因素安排各自的策略，决定是进攻还是防守，再由策略库进行战术部署。策略应根据比赛规则和经验进行提取，并存在知识库中。知识库还应有一个学习智能体，用于不断丰富策略。各种智能算法如神经网络、模糊算法、遗传算法等也可以应用到构造策略库以及策略选择过程中。

决策过程是从机器人感知所建立的系统特征状态空间到机器人动作空间的一个映射过程<sup>[4]</sup>。决策子系统的基本设计思想是把比赛场上的态势抽象化、模糊化，通过选择少量有特点的模糊集合构成状态空间(主要是把场地划分为若干区域，根据双方机器人和球所在的具体区域)，通过决策控制和推理，使机器人选择合适的动作，实现对机器人系统实时控制，并且要尽可能多的考虑所面对的情况使决策子系统具有鲁棒性<sup>[5]</sup>。

(4) 通信子系统：全自主移动机器人系统的通信包主要包括以下几个部分。

1、机器人与场外教练机之间的通信：对比赛中的机器人进行现场指挥，实

现各方机器人之间的团队协作进行踢球。可通过为各足球机器人装备基于 IEEE 802.11b 标准协议的无线通信系统，并在两端分别用 Socket 进行编程加以实现。

2、机器人视觉系统与决策系统之间的通信：机器人通过视觉系统获取比赛场上机器人姿态信息和场上态势，通过 Socket 实现通信，供决策系统使用。

3、上位机与下位机之间串口通信：上位机决策系统发出相应的运动或动作指令通过串行通信接口传给运动控制子系统，实现机器相应的运动。同时下位机运动控制系统把机器人驱动轮转速和一些状态信息通过串口传给决策子系统作为机器人自定位的辅助信息。

### 1.3 全向移动足球机器人运动控制系统的研究现状

全向移动足球机器人根据轮子的移动特性可分为非全向移动机器人和全向移动机器人两种<sup>[6]</sup>。平面上移动的机器人存在前后、左右、自转三个自由度的运动，即运动中的轮式移动机器人的姿态信息由三个参数来确定。若轮式移动机器在平面沿任意方向移动必须改变其姿态，即机器人所具有的自由度少于三个，则为非全向移动机器人，它只能在与轮子轴线垂直的方向作前后移动，在不打滑的情况下不具有侧向移动的能力。全向移动机器人作为一种特殊之轮式机器人，具有可任意在平面上移动而不需改变其姿态的优点，它能够在保持机器人姿态不变情况，在沿平面上任意方向移动，并可以在原地上任意角度转动。可见，全向移动机器人运动灵活，非常适合工作空间有限、对机器人的机动性能要求高的场合。全向移动机器若要实现全向移动还必须采用独特设计的全向轮。全向轮有效地避免了普通轮子不能侧向运动带来的非完整性约束<sup>[7]</sup>，理论上可以在任意角度在机器人所处平面上运动。

RoboCup 中型组机器人足球赛中所用到机器人，主要采用以下两种驱动方案：（1）双轮差动驱动方式；（2）全向轮驱动方式。

双轮差动驱动：采用两个共轴线的驱动轮共同驱动机器人的运动，机构较为简单，应用也较为普遍，但这种驱动方式也存在一些缺点：除了两个驱动轮外，另外还必须存在一个支撑点，使得机器人的底盘很低，并且这个支撑点的设计比较困难。双轮差动驱动的机器人从运动学角度讲是一非完整系统，当机器人需要向与轮面垂直的方向运动时，必须先转体再前后移动，降低了运动效率，使得机器人灵活性不够。

全向轮驱动：采用独特设计的三个或四个全向轮作为机器人的驱动轮，实现任意方向上的瞬时运动，这种方案避免了双轮差动机构中的上述缺点。其中四轮驱动机构具有较好的稳定性，它的驱动方式多样，目前已被广泛采用。但是四轮机器人在不平的地面上行进时有可能出现一个轮子不着地的情况，这样在计算轮速的时候将产生很大的误差。另外，如果在四个轮子上安装四个电机来驱动的话，



由于机器人在平面上运动时最多有三个自由度，这样便会出现冗余的情况。

为了提高机器人系统的运动控制性能，近年来其驱动电机也呈现多样化。中型组足球机器人驱动电机目前使用较多的主要有两种：直流电动机和无刷直流电动机（Brushless DC Motor, BLDCM）。

（1）直流电动机调速性能好，可以在重负载条件下，实现均匀、平滑的无级调速，而且调速范围较宽；起动力矩大，可以均匀而经济地实现转速调节。但直流电动机有电流换向的问题，成本高，运行中的维护检修也比较麻烦，严重影响了它在机器人驱动中的实际应用。

（2）无刷直流电动机又称无换向器电动机、无整流子直流电动机。它是用半导体逆变器取代一般直流电动机中的机械换向器，构成没有换向器的直流电动机。这种电机结构简单，运行可靠，没有火花，电磁噪声低，它拥有一般直流电动机优良的静、动态特性，非常适合作为轮式移动机器人的驱动电机，不过无刷直流电动机的控制比较复杂。

轮式移动机器人运动控制的主要任务有：姿态稳定控制、轨迹跟踪控制和路径跟踪控制。全向移动机器人是典型的完整系统，相对双轮差动机构轮式移动机器人（非完整系统），它的控制精度更高，因此全向移动机器人，吸引了越来越多的学者参与研究。RoboCup 中型组足球机器人的全向移动机构及其控制水平已经比较成熟，采用全向驱动的机器人在比赛中屡屡取得佳绩，以优异的运动性能展现出全向移动平台的优势，吸引了众多科研机构关注。纵观近年来国内或是国际上的 RoboCup 足球比赛，各参赛机器在运动性能、反应灵活性方面都在逐年提高，越来越多球队采用了全向轮式移动机器人。为了充分发挥全向轮式移动机器人的优势，对机器人底层运动控制系统的设计提出了更高的要求，如对处理器的资源和处理速度要求也越来越高，主控芯片经历了 8 位单片、16 位单片机、32 位单片机、16 位或 32 位 DSP 的变化和发展。全向移动机器人运动控制系统已经成为移动机器人设计和研究的重点。

## 1.4 本论文的主要内容及组织结构

本文的主要内容是：本文基于 RoboCup 中型机器人足球竞赛平台，研究全方位移动机器人的运动控制和实现。研究内容主要包括全方位移动机器人的运动控制系统模型的建立、控制算法设计、仿真分析，以及机器人运动系统的硬件设计和控制软件实现。

本文的主要内容包括：

第一章主要介绍了课题来源和背景及 RoboCup 中型组足球机器人系统，分析了机器人运动控制系统国内外的研究现状，阐述了课题研究的意义和目的以及论文的主要内容。

第二章建立了三轮全向移动机器人的运动学和动力学模型,分析机器人基本的运动特性,设计三轮全向移动机器人的运动控制方案。

第三章在 Matlab/Simulink 环境下对底层 BLDCM 驱动系统进行建模和仿真研究,验证底层运动控制系统控制策略的可行性。

第四章介绍了机器人的基本控制任务,然后分析了三轮全向机器人的控制特性,设计了三轮全向足球机器人姿态稳定和轨迹跟踪控制算法。

第五章重点介绍了机器人运动控制系统软、硬件设计与开发。硬件系统开发主要包括控制板和驱动板的硬件电路设计;软件部分主要包括主程序设计及各种中断程序的设计。

最后对本文做的工作进行了简要总结,对以后研究方向做出展望。

## 1.5 本章小结

本章分析和探讨了足球机器人的研究背景和研究意义,介绍了 RoboCup 中型组足球机器人系统和足球机器人运动控制系统国内外的研究现状,提出了本文的研究内容和解决的问题,最后给出了文章的结构安排。

## 第二章 机器人系统模型研究与控制方案设计

### 2.1 三轮全向移动机器人系统建模

双轮差动驱动方式机器人是一非完整系统，其轨迹规划较为复杂，运动轨迹归结为直线和圆弧。而三轮全向移动机器人为完整系统，有三个自由度，对于机器人的三个参数( $x$   $y$   $\theta$ )是无约束，两点之间可以直线运动，并可在行进中转向以调整姿态角<sup>[8]</sup>，其运动轨迹归结为折线。下面将分别从运动学和动力学两个方面分析三轮全向移动机器人的控制方法。

#### 2.1.1 三轮全向移动机器人运动学建模

三轮全向移动机器人其驱动轮由三个全向轮组成，径向对称安装，各轮互成 $120^\circ$ 角，滚柱垂直于各主轮。三个全向轮的大小和质量完全相同，而且由性能相同的电机驱动。建立图 2.1 所示的世界坐标系  $xoy$  和机器人坐标系  $XOY$ 。图中， $\theta$  为机器人坐标系与世界坐标系之间的夹角； $\phi$  为驱动轮间的夹角， $\phi = 120^\circ$ ； $L$  为机器人中心到轮子中心的水平距离。

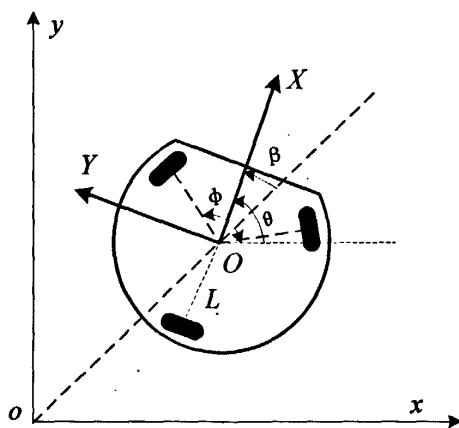


图 2.1 三轮全向轮式机器人示意图

设  $v_1, v_2, v_3$  为全向轮线速度， $v_x, v_y$  分别为机器人在  $XOY$  坐标系  $X$  轴和  $Y$  轴的速度分量； $\omega$  为机器人自转的角速度。则  $(v_1 \ v_2 \ v_3)^T$  与  $(v_x \ v_y \ \omega)^T$  之间的关系可表示如下：

$$\begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \begin{pmatrix} \sin(\phi/2) & \cos(\phi/2) & L \\ -\sin(\phi/2) & \cos(\phi/2) & L \\ 0 & -1 & L \end{pmatrix} \begin{pmatrix} v_x \\ v_y \\ \omega \end{pmatrix} \quad (2.1)$$

以三轮全向轮式机器人的中心  $O$  为参考点，取广义坐标为  $q = (x \ y \ \theta)^T$ ，其中  $(x \ y)$  为机器人中心  $O$  在世界坐标系  $xoy$  中的坐标， $\theta$  为机器人坐标系与世界坐标系之间的夹角。根据坐标系统的建立情况，可知世界坐标系与机器人坐标系之间的变换关系如下：

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} v_x \\ v_y \\ \omega \end{pmatrix} \quad (2.2)$$

由式(2.1)，(2.2)可以得出：

$$\begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \begin{pmatrix} \sin(\phi/2 - \theta) & \cos(\phi/2 - \theta) & L \\ -\sin(\phi/2 + \theta) & \cos(\phi/2 + \theta) & L \\ \sin\theta & -\cos\theta & L \end{pmatrix} \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} \quad (2.3)$$

式(2.3)表示了机器人在世界坐标系中的速度与驱动轮速度之间的关系。

### 2.1.2 三轮全向移动机器人动力学模型

设  $f_i$  为第  $i$  个驱动电机提供给机器人的驱动力，在世界坐标系  $xoy$  下，绕机器人中心的转动惯量为  $I$ ，由牛顿第二运动定律有：

$$F_{xa} = m\ddot{x}_a, \quad F_{ya} = m\ddot{y}_a, \quad M_i = I\ddot{\theta} \quad (2.4)$$

根据各驱动力方向及机器人的姿态，可写出具体的动力学方程如下：

$$m\ddot{x}_a = f_1 \sin(\varphi + \theta) - f_2 \sin(\varphi - \theta) + f_3 \cos\theta \quad (2.5)$$

$$m\ddot{y}_a = f_1 \sin(\varphi + \theta) - f_2 \cos(\varphi - \theta) + f_3 \sin\theta \quad (2.6)$$

$$I\ddot{\theta} = L \sum_{i=1}^3 f_i \quad (2.7)$$

式中  $\varphi$  为  $f_i$  与驱动轮线速度的夹角，由机器人全向轮结构可知  $\varphi = 30^\circ$ 。

第  $i$  个轮子的动力学模型可描述为：

$$I\ddot{\theta} = L \sum_{i=1}^3 f_i k u_i - l \ddot{\psi}_i = I_w \ddot{\psi}_i + f_i r \quad (2.8)$$

式中：

$k, l$ ——常数；

$I_w$ ——轮子绕其轴线的转动惯量；

$r$ ——轮子半径；

$\dot{\psi}_i, \ddot{\psi}_i$ ——第  $i$  个轮子的角速度和角加速度；

$u_i$ ——第  $i$  个驱动电机的驱动电压；

式(2.8)中轮子转动惯量  $I_w$  很小， $I_w \ddot{\psi}$  相对于  $f_i r$  所消耗的扭矩很小，在实际问题中采用简化计算形式，式(2.8)可近似写为：

$$k u_i - l \dot{\psi}_i = f_i r \quad (2.9)$$

三轮全向移动机器人系统的动力学模型方程为：

$$\begin{bmatrix} m\ddot{y}_a \\ m\ddot{y}_a \\ I\ddot{\theta} \end{bmatrix} = \frac{k}{r} \begin{bmatrix} -\sin(\theta+30^\circ) & -\sin(30^\circ-\theta) & \cos\theta \\ \cos(\theta+30^\circ) & -\cos(30^\circ-\theta) & \sin\theta \\ L & L & L \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} - \frac{3l}{2r} \begin{bmatrix} \dot{x}_a \\ \dot{y}_a \\ 2L^2\dot{\theta} \end{bmatrix} \quad (2.10)$$

## 2.2 三轮全向移动机器人结构设计

### 2.2.1 全向轮结构设计

全向轮驱动方式的机器人具有全向运动能力的关键在于其全向轮系结构。全向轮包括轮毂和从动轮，从动轮为套在大轮边缘可侧向移动小轮，这样一来机器不论是前后移动还是横向移动，与地面间的摩擦始终为滚动摩擦，减小机器人在行进中的阻力。

目前，全向轮结构主要有 2 种<sup>[9~10]</sup>：一种为互补结构，从动轮沿大轮圆周切向方向交错互补安装，这种结构始终有一个小轮的边缘着地，运行稳定，轮子较宽，承载能力强，但这种互补安装方式的全向轮着地点内外交错，对机器人的旋转会造成非线性影响，使机器人在运动方向上有所偏移；另一种是非互补结构，轮毂的外圆周处均匀开设有 3 个或 3 个以上的轮毂齿，每两个轮毂齿之间装设有一从动轮，该从动轮的径向方向与轮毂外圆周的切线方向垂直。这种结构轮子的宽度比较小，着地点始终在一个圆上，不会对机器人带来非线性影响<sup>[11]</sup>，但相邻的两个小轮之间存在间隙，使轮子在运动中的直径会发生变化，机器人的上下振动比较厉害，其承载能力也不如前一种结构。非互补结构全向轮在运动中直径的变化幅度：

$$\Delta R = \left[ 1 - \cos\left(\frac{180^\circ}{n}\right) \right] \cdot R \quad (2.11)$$

式 (2.11) 中  $n$  为从动轮个数。

第 2 种，非互补结构的全向轮控制精度和结构紧凑程度要比互补结构的全向轮好。但考虑到中型组机器人体积大、质量重的特点，其结构不要求像小型组机器人那么紧凑，而要求有一定的承载能力，虽然互补结构全向轮机器人旋转时会带来非线性影响，但可以通过机器人视觉反馈加以修正。综合考虑，本课题选用互补结构的全向轮作为机器人的驱动轮。

### 2.2.2 车体布局及驱动电机的选取

机器人车体布局和运动机构对机器人的运动性能有着重要的影响。在设计时应尽量减少车体重量、尽可能降低车体重心，降低转动惯量，增加机器人稳定性以及增强机器人抗碰撞能力。减少机器人车体重量主要从材料上考虑，使用轻型高强度材料等；增加车体稳定性主要体现在机器人各部件的连接方式；增强机器

人抗碰撞能力应尽量使用防护材料,采取减震措施等<sup>[12]</sup>;降低车体转动惯量就是尽量使车体重心位于机器人中心。中型组机器人由于其运动控制较复杂,结构不要求像小型组机器人那般紧凑,应尽量采用对称的布局结构。中型组机器人布局可分为以下三层:最上面一层为上位机(笔记本电脑)和全景摄相机存放及安装空间,中间一层运动控制卡的存放空间,最下层为机器人的驱动机构,一般电池也置于这一层,这样可降低机器人重心。由于中型组机器人体积较大,所以整个结构采用板柱结构,各层用支撑杆以及螺丝连接。

驱动电机的性能直接影响着机器人的运动性能,并且对机械结构也会有一定的影响。无刷直流电动机采用 PWM 控制,只需要通过软件改变 PWM 波的占空比即可实现无级调速,有利于提高移动机器人运动中的灵活性。

考虑到机器人自身体积较小,因此首选带减速器电机,这样将无刷直流电机输出轴与机器人驱动轮连接起来即可。

综合考虑,本课题选择了瑞士 MAXON 公司的 EC45 系列无刷直流电动机(自带霍尔传感器)作为机器人的驱动电机,所配减速器为 MAXON 公司的 GP42C 系列,减速比 156/1,运行效率为 75%。电机参数如下:

额定功率	250W	额定电压	24V	最大工作电流	7.47A
额定转矩	310mNm	最大效率	84%	起动电流	55.8A
堵转转矩	2420mNm	空载电流	435mA	空载转速	5250rpm
额定转速	4520rpm	转矩常数	43.3mNm/A	速度常数	221rpm/V
机械时间常数	4.80ms	相间电阻	0.430 Ω		
相间电感	0.170mH	转子惯量	209gcm <sup>2</sup>		

### 2.3 轮式移动机器人运动系统控制方案选择

全向移动机器人运动控制系统一般可采用以下两种控制方案:基于运动学模型的分层控制和基于动力学模型的分层控制<sup>[13~14]</sup>。在选择机器人的运动控制方案时必须根据实际情况权衡利弊加以选择。

#### (1) 基于运动学模型的分层控制

基于机器人运动学模型的分层控制系统框图如图 2.2 所示。它可分为基于机器人运动学模型的运动控制和驱动电机转速控制上下两层。上层的基于机器人运动学模型的控制器输入机器人的姿态信息、输出为机器人驱动轮的期望速度,它不需要考虑机器人的动力学特性。下层的驱动电机转速控制相对上层控制器是透明的,即假设机器人驱动电机期望转速为其实际转速。

三轮全向移动机器人的运动学模型较其动力学模型相对简单,反映了机器人的主要运动特性。基于机器人运动学模型的分层控制,不需要考虑机器人的动力学特性,因此针对运动学特性相同而动力学特性不同的机器人,可以采用相同的

运动控制算法，控制器具有通用性。另外，上层的基于运动学模型的控制器和下层的电机转速控制器可分开进行设计，有利于采用模块化的方法进行开发，也便于控制算法的改进与更新，降低了控制器的设计难度。

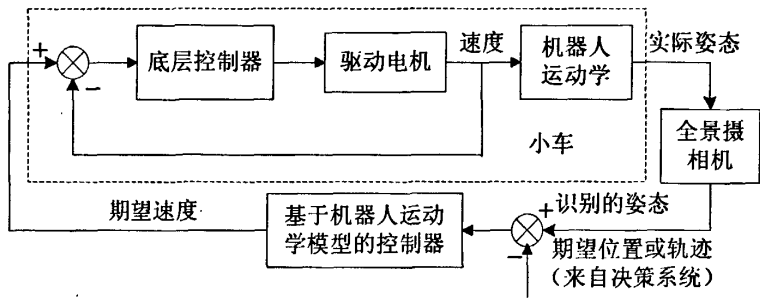


图 2.2 基于运动学模型的分层控制框图

然而基于机器人运动学模型的分层控制也有其缺点：底层驱动电机转速不能立刻调节到期望值，给系统带来了控制上的延时；另外，对驱动电机进行转速控制就需要进行转速的检测，这样一来就需要更多的硬件设备，增加了底层运动控制电路的设计难度，使车体的机械构造也变得复杂，增加了系统的设计制作成本。

### （2）基于动力学模型的控制

基于机器人动力学模型的控制框图如图 2.3 所示：通过分析机器人动力学模型可知，这种控制方案不需要对每个驱动电机进行转带闭环控制，基于机器人动力学模型的控制器输入为机器人姿态信息，输出为机器人驱动电机的驱动电压或者电流。

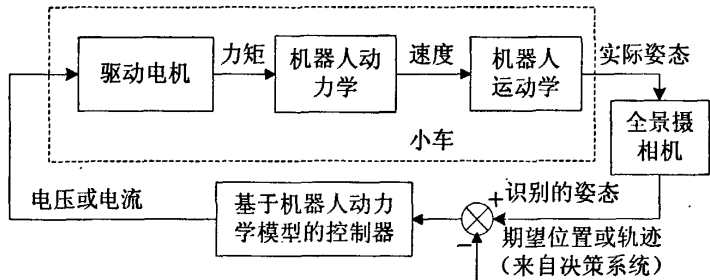


图 2.3 基于动力学模型的控制框图

由于这种控制方案不需对驱动电机进行转速控制，消除了底层转速控制带来的延时，因此若能建立机器人精确动力学模型，将提高控制精度。基于动力学模型的控制不需要直接控制驱动电机的转速，也就不需要进行驱电机转速检测，因此其底层运动控制电路设计较为简单些，机械结构也不如第一种控制方案复杂，降低了系统的制作成本。但是，由前面对三轮全向移动机器人进行运动学和动力学建模可知，三轮全向移动机器人的动力学模型要复杂得多，其动力学参数难以确定，很难建立它的精确动力学模型。

针对长沙理工大学机器人实验室现有的足球机器人，通过分析对比以上两种

控制方案的优缺点,本课题选择基于运动学模型的分层控制来实现三轮全向移动机器人的运动控制。主要原因有以下几点:

1. 机器人运动学模型相对简单,反映了轮式移动机器人的主要运动特性,控制器的设计相对容易。
2. 现有的三轮全向移动机器人的机构较为复杂,加工精度不够高,质量分布不均匀,存在明显的个体差异,很难建立精确的动力学模型。
3. 对于无刷直流电机的转速控制已经有了很成熟的控制算法,能达到很好的控制效果,可以尽量消除底层运动控制带来的延时。
4. 所采用无刷直流电机自带霍尔传感器,可以对电机实现较为精确的速度控制。
5. 分层控制有利于系统模块化设计与调试,也有利于控制算法改进和更新。

## 2.4 本章小结

本章建立了三轮全向机器人的运动学和动力学模型,并分析了它的运动特性。设计了三轮全向移动机器人的结构,通过分析比较,选择了基于运动学模型的分层控制作为三轮全向机器人运动控制的基本控制方案。对研究对象进行数学建模,确定控制方案,为机器人的运动控制系统软、硬件设计和运动控制算法研究奠定了基础。



# 第三章 基于 Matlab 的控制系统建模与仿真

第二章分析了三轮全向移动机器人的运动学模型,在已知机器人整体速度时就可以求出三个驱动轮的速度。三轮全向移动机器人运动控制系统框图如图 3.1 所示。全景摄相机捕捉到机器人小车的位置姿态信息,通过上位机的控制算法,规划出机器人的运动路径,由无线通讯模块将小车平向移动速度及转动角速度发给底层控制器,得到小车三个轮子的给定速度,经电机控制算法,达到姿态控制和轨迹跟踪的目的。本课题所研究的三轮全向机器人采用 BLDCM 驱动, BLDCM 是非线性、强耦合系统,下面对它的原理、数学模型、及 BLDCM 双闭环控制系统仿真模型分别给予分析。

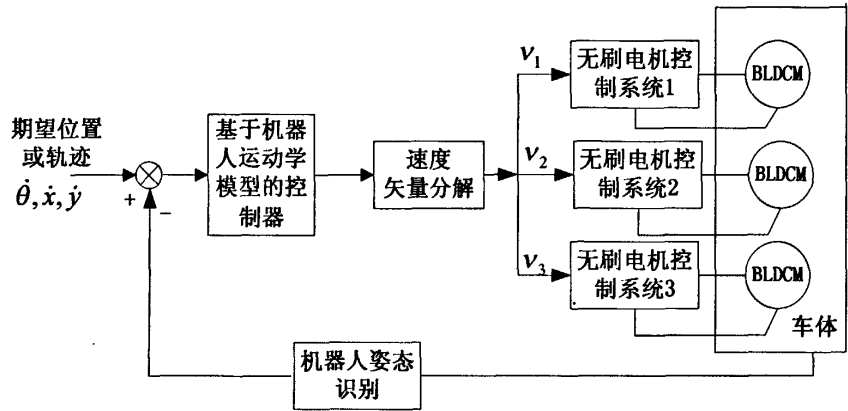


图 3.1 三轮全向移动机器人运动系统框图

## 3.1 BLDCM 基本原理

无刷直流电机是在有刷直流电动机的基础上发展起来的,主要由电动机本体,霍位置传感器和电子开关线路三部分组成<sup>[15~16]</sup>。它的电枢绕组是经电子“换向器”接到直流电源上,可把它视为直流电动机的一种。从供电逆变器的角度看,它又可属于永磁同步电动机的一种,因为无刷直流电动机转速变换以及电枢绕组的电流变换是和逆变器的输出频率一致的。但无刷直流电动机电枢绕组中流过的电流以方波形式变化,故又称方波电流永磁交流电动机。

以三相星形定子绕组为例,如图 3.2 所示,三相桥式电子换相电路工作在二相导通 120°三相六状态模式,即每一时刻总有两个功率开关导通,每隔 1/6 通电周期 (60°电角度) 要根据转子位置传感器的信号进行一次换相,每次换相一个功率管开通,另一个则关断,每个功率管均导通 120°电角度,三相桥式逆变器六只功率管的导通顺序是: V1V6→V1V2→V3V2→V3V4→V5V4→V5V6……, 依次

循环导通。当功率管 V1、V2 导通时，电流从 V1 管流入 A 相绕组，再从 C 相绕组流出，经 V2 管回到电源。

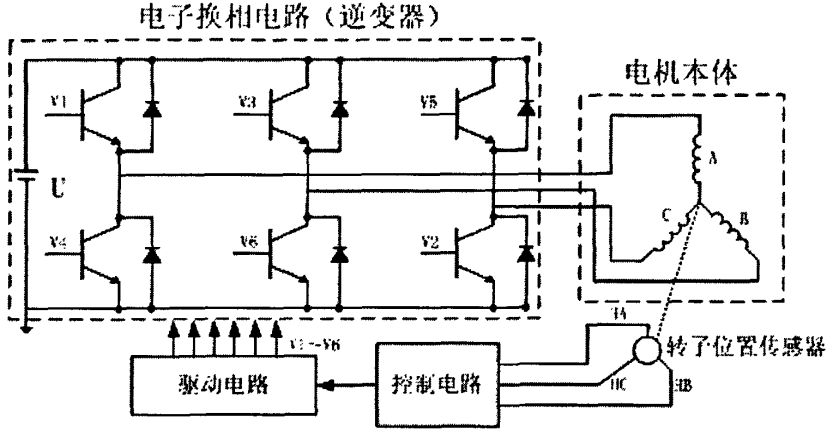


图 3.2 无刷直流电机主回路示意图

## 3.2 BLDCM 的数学模型

本课题所采用的无刷直流电动机定子绕组为三相星形连接，采用两相导通三相六拍运行方式。假定三相绕组完全对称均匀分布于光滑定子的内表面，忽略齿槽效应；且气隙磁场为方波，定子电流、转子磁场分布皆对称；忽略磁路饱和，不计涡流和磁滞损耗。则可根据无刷直流电机特性，建立其动态数学模型。

### 3.2.1 电压方程

无刷直流电机三相定子电压的平衡方程<sup>[17]</sup>可用如下状态方程表示：

$$\begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix} = \begin{bmatrix} R_a & 0 & 0 \\ 0 & R_b & 0 \\ 0 & 0 & R_c \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \frac{d}{dt} \begin{bmatrix} L_a & L_{ab} & L_{ac} \\ L_{ba} & L_b & L_{bc} \\ L_{ca} & L_{cb} & L_c \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} e_a \\ e_b \\ e_c \end{bmatrix} \quad (3.1)$$

式中  $u_a, u_b, u_c$  为定子绕组相电压 (V)； $i_a, i_b, i_c$  为定子绕组相电流 (A)； $L_a, L_b, L_c$  为绕组自感 (H)； $L_{ab}, L_{ac}, L_{ba}, L_{bc}, L_{ca}, L_{cb}$  为绕组间互感 (H)； $R_a, R_b, R_c$  为绕组电阻 ( $\Omega$ )； $e_a, e_b, e_c$  为反电势 (V)。由于转子是永磁的，忽略其磁阻影响，并假设三相对称，则可认为： $L_a = L_b = L_c = L$ ， $R_a = R_b = R_c = R$ ， $L_{ab} = L_{ac} = L_{ba} = L_{bc} = L_{ca} = L_{cb} = M$ ，又因为  $i_a + i_b + i_c = 0$ ，于是，式 (3.1) 可以改写成：

$$\begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix} = \begin{bmatrix} R & 0 & 0 \\ 0 & R & 0 \\ 0 & 0 & R \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \frac{d}{dt} \begin{bmatrix} L-M & 0 & 0 \\ 0 & L-M & 0 \\ 0 & 0 & L-M \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} e_a \\ e_b \\ e_c \end{bmatrix} \quad (3.2)$$

根据上面的讨论，可以得到无刷直流电动机三相等效电路图，如图 3.3 所示：

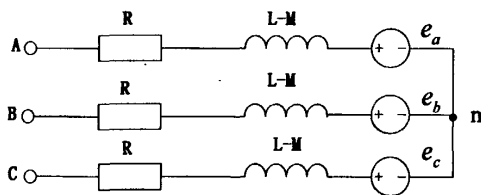


图 3.3 无刷直流电动机三相等效电路

### 3.2.2 电磁转矩方程和运动方程

无刷直流电动机的电磁转矩方程和运动方程为：

$$T_e = (e_a i_a + e_b i_b + e_c i_c) / \omega \quad (3.3)$$

$$T_e - T_L = B\omega + J \frac{d\omega}{dt} \quad (3.4)$$

式中：  $T_e$  为电磁转矩 ( $N \cdot m$ )；  $\omega$  为角速度 ( $rad/s$ )；  $T_L$  为负载转矩 ( $N \cdot m$ )；  $B$  为阻尼系数 ( $N \cdot m \cdot s / rad$ )；  $J$  为转子的转动惯量 ( $kg \cdot m^2$ )；  $\frac{d\omega}{dt}$  为转子机械角加速度 ( $rad/m^2$ )。

### 3.3 BLDCM 双闭环调速系统仿真模型的建立

为了研究电机控制算法和控制策略的合理性，本章在 Matlab/Simulink 的环境下，建立了针对本课题所研究的电机控制系统的仿真模型。

前面分析了无刷直流电动机的数学模型，在此基础上，本文利用 Matlab/Simulink 中 SimPowerSystem Toolbox 提供丰富的模块库，提出了一种建立无刷直流电动机系统仿真模型的新方法，系统的设计框图如图 3.4 所示。系统采用转速、电流双闭环控制方案：转速环采用单神经元自适应 PID 调节器，电流环采用电流滞环控制。采用模块化建模方法，将图 3.4 所示的控制系统分割成各部分功能独立的子模块。

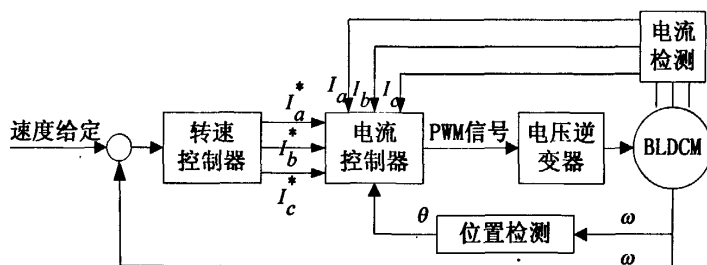


图 3.4 BLDCM 控制系统框图

图 3.5 为 BLDCM 双闭环调速系统的仿真模型框图，其中主要包括 BLDCM 本体模块、速度控制模块、电流控制模块、电源和逆变器模块。把这些功能模块与

其中实现特定功能的 S-Function 相结合,在 Matlab7.1/Simulink 中搭建出 BLDCM 控制系统的仿真模型<sup>[18~22]</sup>, 图中各功能模块的结构和作用简述如下。

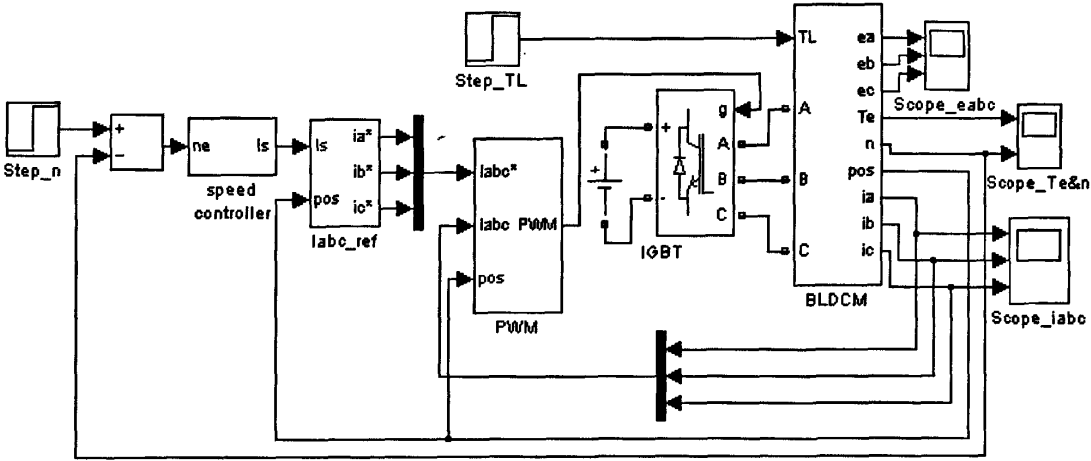


图 3.5 BLDCM 控制系统仿真模型

### 3.3.1 BLDCM 本体模块

#### (1) 电机电压方程模块

根据式 (3.2) 定子端电压方程可以得出 BLDCM 等效模型如图 3.6 所示。

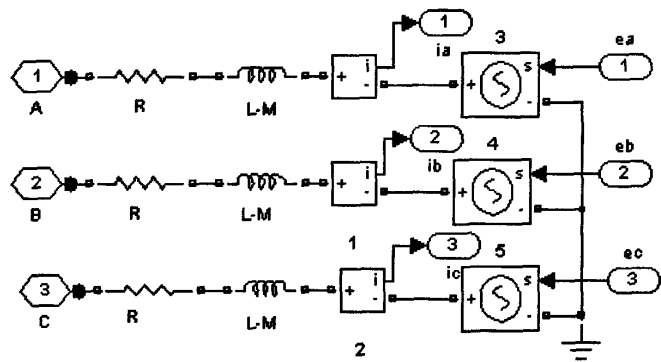


图 3.6 电机电压方程模块

#### (2) 电机的反电动势模块

BLDCM 梯形波反电动势的求取是较难解决的问题,反电动势波形不理想会造成转矩脉动增大、相电流波形不理想等问题,严重时会导致换向失败使电机失控。因此,获得理想的反电动势波形是 BLDCM 仿真建模的关键问题之一。目前求取反电动势较常用的三种方法为:有限元法<sup>[23]</sup>、傅立叶变换法<sup>[24]</sup>及分段线性法<sup>[25]</sup>。

相比较以上所提出的三种求取 BLDCM 梯形反电势的方法,分段线性法最为简单,精度较高,满足建模仿真的要求且易于实现,综合比较,本文采用分段线性法求取 BLDCM 梯形波反电动势。BLDCM 梯形反电势波形如图 3.7 所示,将

BLDCM 的一个运行周期  $0 \sim 360^\circ$  分为 6 个阶段，每  $60^\circ$  为一个换向阶段，每一相的每一个运行阶段都可用一段直线表示，根据某一时刻的转子位置和转速信号，确定该时刻各相所处的运行状态，通过直线方程即可求得反电动势波形。理想情况下，两相导通三相六状态运行无刷直流电机，其梯形波反电势的波形图如图 3.7 所示。

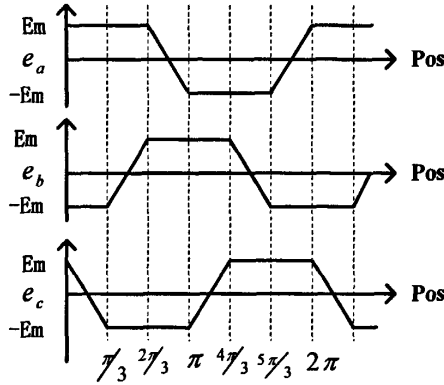


图 3.7 三相反电动势波形

设  $K$  为无刷直流电机反电势系数； $pos$  为无刷直流电机转子位置信息； $\omega$  为无刷直流电机的转速，则无刷直流电机转过的圈数  $per = \text{fix}(pos / (2\pi)) \times 2\pi$ 。可以得出无刷直流电机三相梯形波反电动势在  $0 \sim 2\pi$  区间内的函数表达式，以 A 相反电动势  $e_a$  为例。

$$e_a = \begin{cases} K \times \omega & 0 \leq pos \leq 2\pi/3 \\ K \times \omega \times \left( \frac{per + 2\pi/3 - pos}{\pi/6} + 1 \right) & 2\pi/3 \leq pos \leq \pi \\ -K \times \omega & \pi \leq pos \leq 5\pi/3 \\ K \times \omega \times \left( \frac{pos - 5\pi/3 - per}{\pi/6} - 1 \right) & 5\pi/3 \leq pos \leq 2\pi \end{cases} \quad (3.5)$$

由于电机的反电动势是一分段函数，所以 A, B, C 相绕组反电动势可以方便地用 S-Function 来实现（程序参见附录 B）程序中  $\text{fix}$  为实现取整的函数。B 相，C 相绕组反电动势相位与 A 相分别相差  $120^\circ$  和  $240^\circ$ 。

### （3）电磁转矩模块

根据 BLDCM 电磁转矩方程式 (3.3) 可建立如图 3.8 所示的电磁转矩仿真模型，输入为 BLDCM 三相相电流  $I_a$ ， $I_b$ ， $I_c$  与三相反电动势  $e_a$ ， $e_b$ ， $e_c$ ，通过 Matlab/Simulink 中的加、乘模块进行运算，即可求得电磁转矩信号  $T_e$ 。

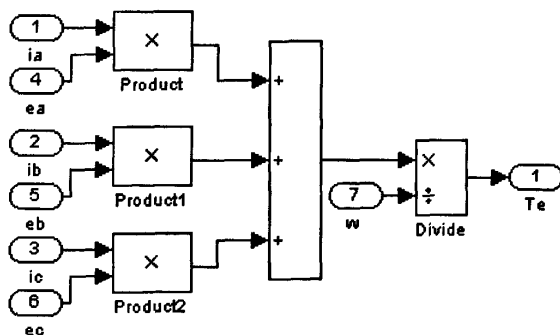


图 3.8 电磁转矩计算模块

#### (4) 机械运动方程模块

由式 (3.4) 可建立起如图 3.9 所示的机械运动模块

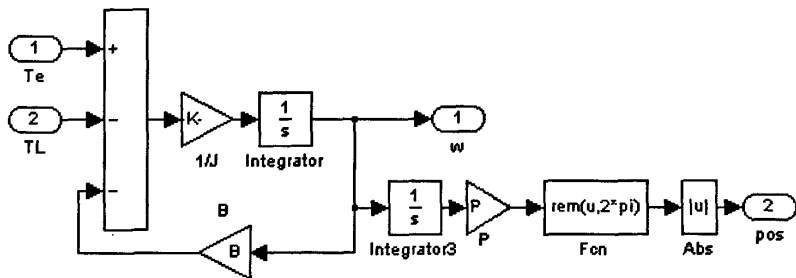


图 3.9 机械运动模块

### 3.3.2 速度控制模块

传统的 PID 控制器问世至今已有近 70 年历史，它以其结构简单、稳定性好、工作可靠而成为工业控制的主要技术之一，但 PID 参数难以实时在线调整，当被控对象的结构和参数不能完全掌握，或参数时变，或是得不到精确的数学模型时，传统的 PID 控制就难以实现良好的控制效果。采用无刷直流电机作为机器人的驱动电机，对其调速性能要求较高，然而无刷直流电动机系统是一个多变量、强耦合的非线性系统，传统的 PID 控制难以实现理想的效果。

自适应控制可以依据对象的输入输出数据，不断地辨识模型参数，这个过程称为系统的在线辨识。随着生产过程的不断进行，通过在线辨识，模型会越来越准确，越来越接近于实际。显然，基于这种模型综合出来的控制作用也将随之不断的改进。在这个意义下，控制系统具有一定的适应能力。神经元是神经网络的基本组成单元，具有自学习和自适应能力，由它构成的控制系统算法简单，能够适应环境的变化，有较强的鲁棒性，可实时控制<sup>[26]</sup>。

#### 3.3.2.1 单神经元自适应 PID 控制器设计

单神经元自适应 PID 控制器，不需要对被控制对象进行精确的辨识，且能适应环境参数的变化，具有自适应和自学习能力，结构简单，鲁棒性强。由于它的这些优良特性，单神经元自适应 PID 控制已被广泛应用工业控制。

## 一、几种典型的学习规则

### (1) 无监督 Hebb 学习规则

Hebb 学习规则是一种联想学习算法，其根本思想为：若两个神经元同时处于激活状态，它们激励的乘积越大则它们连接强度相应增强也越大，反之则越小，他们之间存在一个正比关系。 $o_i$  和  $o_j$  分别表示神经元  $o$  和神经元  $j$  的激活值， $w_{ij}$  表示连接从神经元  $o$  到神经元  $j$  的权值，则无监督 Hebb 学习规则可用公式表示：

$$\Delta w_{ij} = \eta o_i(k) o_j(k) \quad (3.6)$$

式 (3.6) 中， $\eta$  为无监督 Hebb 学习规则的学习速率。

### (2) 有监督的 Delta 学习规则

有监督学习的 Delta 学习规则其实质是在无监督的 Hebb 学习规则中引入一个教师信号，即引入期望输出  $d_j$ ，将期望输出  $d_j$  与实际输出  $o_j$  的差作为新的激活值，其学习规则可表示为：

$$\Delta w_{ij} = \eta (d_j(k) - o_j(k)) o_i(k) \quad (3.7)$$

### (3) 有监督的 Hebb 学习规则

有监督的 Hebb 学习规则是以上两种学习规则的结合，其学习规则可表示为：

$$\Delta w_{ij} = \eta (d_j(k) - o_j(k)) o_j(k) o_i(k) \quad (3.8)$$

## 二、单神经元自适应 PID 控制

单神经元自适应 PID 控制结构如图 3.10 所示。

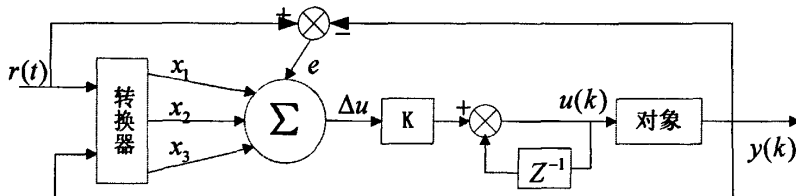


图 3.10 单神经元自适应 PID 控制结构图

由上图可知，单神经元自适应控制器通过引入教师信号即有监督的 Hebb 学习规则调整神经元的连接权值，以此来实现控制器的自适应、自组织功能。该控制器的控制算法和学习算法用公式表示如下：

$$u(k) = u(k-1) + K \sum_{i=1}^3 w'_i(k) x_i(k) \quad (3.9)$$

$$w'_i(k) = w_i(k) / \sum_{i=1}^3 |w_i(k)| \quad (3.10)$$

$$w_1(k) = w_1(k-1) + \eta_I z(k) u(k) x_1(k) \quad (3.11)$$

$$w_2(k) = w_2(k-1) + \eta_P z(k) u(k) x_2(k) \quad (3.12)$$

$$w_3(k) = w_3(k-1) + \eta_D z(k) u(k) x_3(k) \quad (3.13)$$

式 (3.10~3.13) 中：

$$x_1(k) = e(k) \quad (3.14)$$

$$x_2(k) = e(k) - e(k-1) \quad (3.15)$$

$$x_3 = \Delta^2 e(k) = e(k) - 2e(k-1) + e(k-2) \quad (3.16)$$

$\eta_i$ ,  $\eta_p$ ,  $\eta_d$  分别为 PID 控制器积分因子  $I$ 、比例因子  $P$  和微分因子  $D$  的学习速率,  $K$  是神经元的比例系数是一个大于 0 的数。为了权系数进行调整, 学习速率  $\eta_i$ ,  $\eta_p$ ,  $\eta_d$  采用不同的值。比例系数  $K$  越大, 则快速性得到加强, 但相应地超调量会较大, 严重时影响系统的稳定; 对于有增大时延的被控对象, 为保证系统的稳定, 须选择较小的  $K$  值; 但是选择的  $K$  值过小, 系统的快速性则受到影响。因此, 比例系数  $K$  的值须慎重选择。

### 3.3.2.2 单神经元自适应 PID 控制器实现及仿真

Simulink 是基于 MATLAB 的系统建模与仿真软件包, 提供了图形用户接口, 可方便快捷的实现动态系统建模与仿真并迅速得到系统的仿真结果。图 3.11 即为用 Simulink 搭建的单神经元自适应 PID 控制器模型, 其中转换器部分是用单位时间延迟模块 Unit Delay 来实现。由于单神经元学习算法不能直接用传递函数加以描述, 无法用 Simulink 中的模块进行建模仿真, 需引入用 Matlab 语言编写的 S 函数嵌入系统完成仿真。在控制器的输出部分接入了饱和限幅非线性模块, 使得系统输出更接近实际。

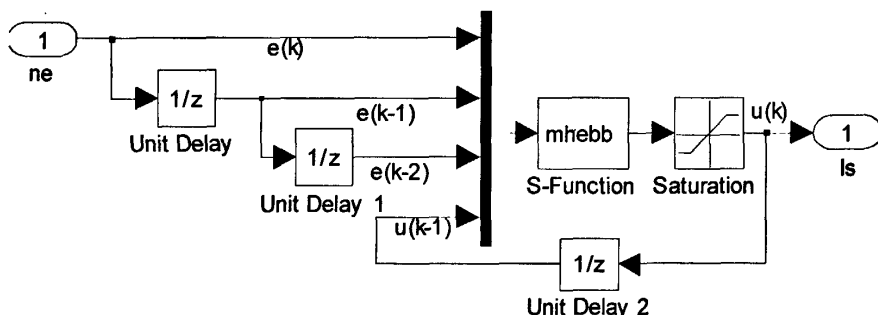


图 3.11 单神经元自适应 PID 仿真模型

单神经元自适应PID控制算法S-Function实现程序参见附录B。程序中输入变量 $u$ 为速度误差信号 $ne$ 。 $x(1)$ ,  $x(2)$ ,  $x(3)$ 为神经元权值, 其初值在程序初始化部分设为0.4, 0.4, 0.4。 $K$ ,  $ni$ ,  $np$ ,  $nd$ 分别神经元比例系数、积分学习速率、比例学习速率、微分学习速率。本系统中设置为 $K=5$ ,  $ni=1.6$ ,  $np=1.0$ ,  $nd=1.5$ 。

### 3.3.3 电流控制模块

#### (1) 参考电流模块

无刷直流电机的三相三流为方波, 三相参考电流可以根据无刷直流电机转子的置和电流的幅值来求取, 用于与实际三相电流相比较, 实现电流闭环控制。无



刷直流电机转子位置与三相参考电流之间的对应关系如表3.1所示，可方便的运用S-Function来实现三相参考电流（程序参见附录B）。

表 3.1 转子位置与三相参考电流对应关系表

参考电流 转子位置	$I_a^*$	$I_b^*$	$I_c^*$
$0 \sim \pi/3$	$I_s$	$-I_s$	0
$\pi/3 \sim 2\pi/3$	$I_s$	0	$-I_s$
$2\pi/3 \sim \pi$	0	$I_s$	$-I_s$
$\pi \sim 4\pi/3$	$-I_s$	$I_s$	0
$4\pi/3 \sim 5\pi/3$	$-I_s$	0	$I_s$
$5\pi/3 \sim 2\pi$	0	$-I_s$	$I_s$

(2) 电流滞环控制模块

机器人底层电机调速系统采用转速、电流双闭环控制，速度环输出  $I_s$  的绝对值是参考电流的给定值，再根据转子的位置信号，就可以求得三相参考电。设置适合的滞环宽度，使三相实际电流跟随给定参考电流变化，实现对电流的滞环跟踪控制，模块结构框图如图 3.12 所示。  $i_{abc}$ 为三相参考电流，  $i_{abc}^*$ 为三相实际电流，输出为 PWM 逆变器控制信号。

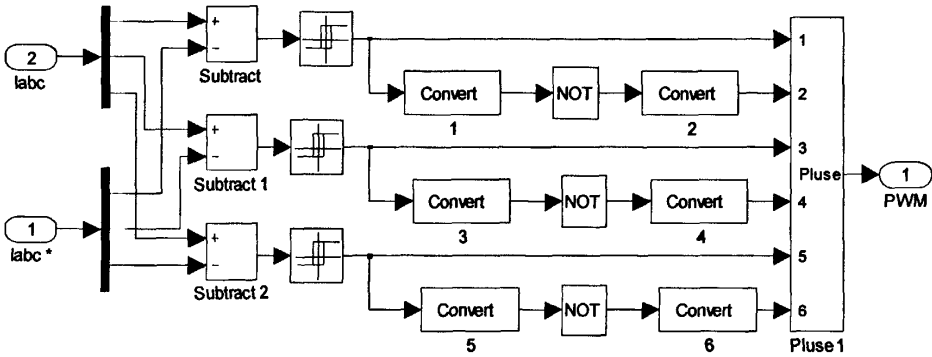


图 3.12 电流滞环控制模型

3.3.4 逆变器换向逻辑模块

BLDCM 控制系统中逆变器的换向是通过检测转子的位置来实现的，并与各相反电动势波形是相对应的。由图 3.7 所示反电动势的波形可知，当反电动势到达波顶或波谷时对应的开关管导通，导通120°电角度后关断，相邻开关管互差60°电角度开通。当转子转过60°电角度时，逆变器的开关管之间就进行一次换流。根据两相通导星形三相六状态下的 BLDCM 基本工作原理，可以得到开关管的导通顺序为：V1V6→V1V2→V3V2→V3V4→V5V4→V5V6……，依次循环导通。依据此导通顺序可以方便地用 S-Function 生成换向逻辑信号（程序参见附录 B）。

### 3.3.5 电源和逆变器模块

电源选用Matlab中的DC电源模块；逆变器选用Universal Bridge模块，设置为IGBT功率开关器件。将电流滞环控制的PWM输出信号和换向逻辑模块输出信号逻辑与，就可产生6路脉冲信号，控制IGBT开关器件的导通和关断；得到可调的三相端电压输出，给BLDCM供电。

### 3.4 实例仿真及结果

根据上述所建立的无刷直流电机控制系统的仿真模型，进行仿真实验。仿真时空载启动，于 $t=0.2s$ 时突加负载 $TL=0.3Nm$ ，得到相反电动势、相电流、转速及转矩的仿真曲线如图 3.13~3.18 所示。

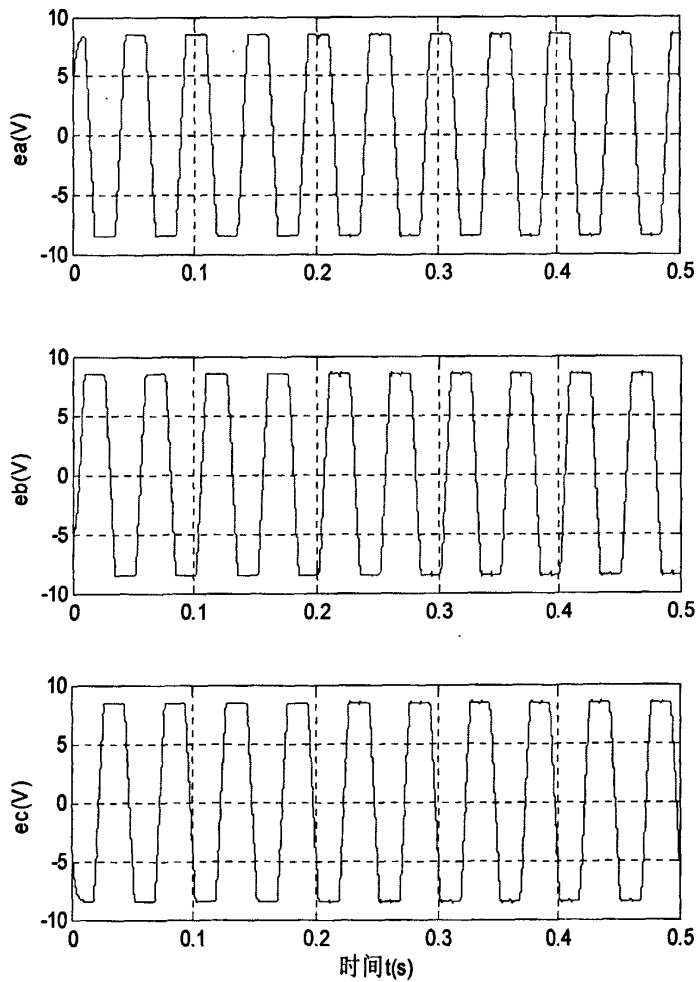


图 3.13 三相反电势波形

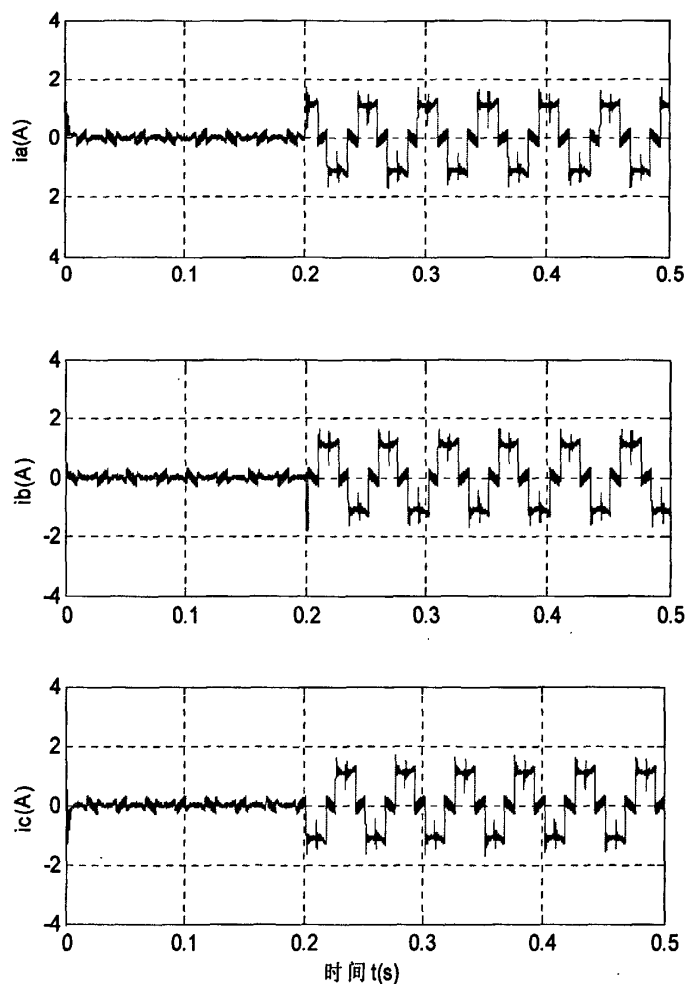


图 3.14 三相电流波形

三相反电势的波形如图 3.14 所示，其平顶部分约为 120 度；由三相电流的波形图 3.14 可知，BLDCM 空载运行时电流近似为 0，加上负载后电流增加并迅速达到稳定，由于换向的影响，方波电流波形的平顶部分有脉动。波形与理论分析基本相同，仿真结果表明了本文所建立的 BLDCM 仿真模型的有效性及控制系统的合理性。

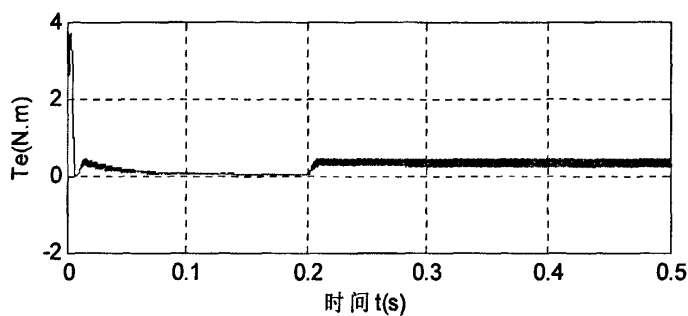


图 3.15 PID 控制下的转矩响应曲线

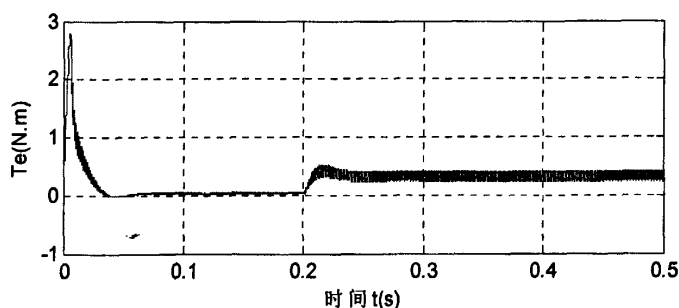


图 3.16 单神经元 PID 控制下转矩响应曲线

图 3.15 和图 3.16 为机器人底层电机调速系统在普通 PID 和单神经元自适应 PID 控制下的转矩响应曲线。BLDCM 空载稳定运行时，电磁转矩近似为零，加上负载后，转矩有轻微的脉动，但能迅速调整到新的稳定状态，这主要是由于电流换向和电流滞环控制器的频繁切换造成的。此外，相电流的波动也会导致转矩的脉动。与普通 PID 控制相比，单神经元自适应 PID 控制的转矩波动较小。

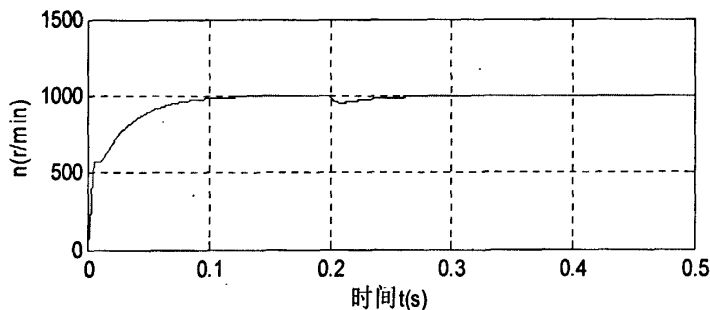


图 3.17 PID 控制下的转速响应曲线

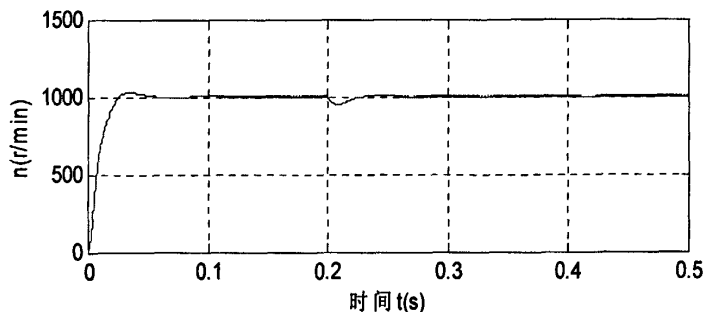


图 3.18 单神经元 PID 控制下转速响应曲线

图 3.17 和图 3.18 系统在普通 PID 和单神经元自适应 PID 控制下的转速响应曲线，在 PID 控制下系统的响应时间为 0.12s，而在单神经元 PID 控制下，系统在较短的时间内（0.05s）进入稳定状态，在  $t=0.2s$  时突加负载，转速发生突降，但又能迅速恢复到平衡状态。

仿真结果表明，采用单神经元自适应 PID 控制器，无论是跟随性还是抗干扰性都比单纯 PID 控制要好，充分体现了单神经元 PID 控制动态响应快、控制

稳定性好的特点。

### 3.5 本章小结

本章介绍了三轮全向移动机器人驱动电机——无刷直流电机的原理及数学模型，建立了无刷直流电机双闭环调速系统的仿真模型，给出了仿真结果，仿真结果表明该无刷直流电机调速系统响应速度快、调速性能好。采用该仿真模型可以灵活地实现各种控制算法和控制策略，能有效地节省系统设计周期，为三轮全向机器人运动控制系统软、硬件设计作了理论上的指导，具有一定的实际意义。

## 第四章 机器人的姿态控制与轨迹跟踪

三轮全向移动机器人运动灵活,并且容易扩展机构,所以近几年来大多参加 RoboCup 中型组比赛的球队纷纷改用三轮全向移动机器人。通过分析三轮全向足球机器人的运动学模型可知:在不打滑的情况下,三轮全向机器人不存在非完整的运动约束,是一个完整约束系统。相比较于其它驱动形式的机器人,三轮全向机器人运动控制算法的设计相对简单些。

### 4.1 轮式移动机器人的主要控制任务

在水平面上运动的轮式移动机器人其控制任务主要有以下三种:姿态稳定控制、轨迹跟踪控制以及路径跟踪控制<sup>[28]</sup>。

1、机器人姿态稳定控制:如图 4.1 所示,对机器人的姿态稳定控制要求机器人能由任意的初始姿态  $\xi_i = (x_i \ y_i \ \theta_i)^T$  移动到末姿态  $\xi_f = (x_f \ y_f \ \theta_f)^T$ 。机器人在移动过程中没有预定的运动轨迹,也不考虑可能遇到障碍的情况。机器人的姿态稳定控制是其最基本的控制任务。对任意给定的机械系统可控,则有:对任意给定的初始姿态  $q_i$  与末姿态  $q_f$  存在一条满足约束的轨迹  $q(t)$  连接这两个状态。RoboCup 中型组机器人足球比赛中,机器人站位、射门等动作的实现需要对机器人进行姿态稳定控制。

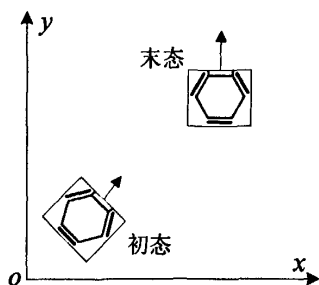


图 4.1 轮式机器人姿态控制示意图

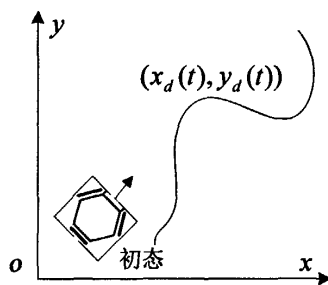


图 4.2 轮式机器人轨迹跟踪控制示意图

2、机器人轨迹跟踪控制:如图 4.2 所示,机器人轨迹跟踪控制要求机器人能够在一定时间内跟踪一条连续的几何曲线。轨迹一般以时间为变量的参数方程描述为  $(x_d(t), y_d(t)), t \in [0, T]$ 。

RoboCup 中型组机器人足球赛中,比赛场地上还有其不断运动的机器人,机器人运动时需要躲避可能的障碍物,这里的障碍物不仅包括对方机器人,也包括队友和边墙。因此要求事先规划出机器人由当前位置出发的一条轨迹,让机器人跟踪该轨迹。因此,轨迹跟踪控制是 RoboCup 中型组足球机器人运动控制的

又一项重要任务。

3、机器人路径跟踪控制：机器人路径跟踪控制与轨迹跟踪控制类似，它要求机器人跟踪一条预定的几何轨迹，但机器人是以一恒定速度跟踪该路径，没有时间上的限制。由于轨迹跟踪控制有时间上的约束，因此对移动机器人控制更为灵活一些，也容易预测到机器人在某一时刻的位置。相对于轨迹跟踪控制，机器人路径跟踪控制在 RoboCup 足球赛中使用较少。

## 4.2 三轮全向移动机器人的控制特性

取三轮全向轮式移动机器人的中心  $O$  为参考点，全局坐标： $q = (x \ y \ \theta)^T$  其中， $(x \ y)$  为机器人中心  $O$  在世界坐标系  $xoy$  中的坐标， $\theta$  为机器人坐标系与世界坐标系之间的夹角。由第二章分析可将三轮全向移动机器人的运动学模型表示为：

$$\dot{q} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = [g_1(q) \ g_2(q) \ g_3(q)]U = \begin{bmatrix} \cos\theta \\ \sin\theta \\ 0 \end{bmatrix} v_x + \begin{bmatrix} -\sin\theta \\ \cos\theta \\ 0 \end{bmatrix} v_y + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega \quad (4.1)$$

其中， $U = (u_1 \ u_2 \ u_3)^T = (v_x \ v_y \ \omega)^T$  为系统的输入， $(v_x \ v_y \ \omega)^T$  表示机器人在本体坐标系中的速度，与轮子转速间的关系见公式 (2.1)。

式 (4.1) 表示的系统可用如下的控制系统表示：

$$\Sigma: \dot{q} = g_1(q)u_1 + g_2(q)u_2 + \cdots + g_m(q)u_m \quad q \in R^n, \ u \in U \subset R^m \quad (4.2)$$

此类系统称为无漂移系统 (Driftless System)，即当输入  $u$  为 0 时，系统的状态不发生漂移。

设  $\bar{\Delta} = L(g_1, \dots, g_m)$  是由  $g_1, \dots, g_m$  组成的李代数，称  $\bar{\Delta}$  为可控制李代数。

根据 W.L.Chow 提出的非线性系统可控性的判别<sup>[29~30]</sup>：对于无漂移系统  $\Sigma$ ，若系统可控制性李代数的秩为  $n$ ，则系统可控的。

式 (4.1) 所表示的系统可控制性李代数为：

$$\bar{\Delta} = [g_1 \ g_2 \ g_3] = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (4.3)$$

$\text{rank}\bar{\Delta} = 3$ ，所以非线性系统 (4.1) 可控。因此可用非线性控制器实现系统的姿态稳定控制和轨迹跟踪控制。

## 4.3 三轮全向移动机器人的姿态控制

### 4.3.1 基于 Lyapunov 直接法的机器人姿态控制

基于极坐标系下设计系统的控制器首先是由 Aicardi<sup>[31]</sup>等人提出。该方法将系统状态方程变换到极坐标下，应用 Lyapunov 直接法设计系统反馈控制律。

如图 4.3 所示, 取三轮全向移动机器人的期望末位置为世界坐标系的原点, 期望方向与世界坐标系的  $x$  轴重合。令  $\rho$  为机器人中心  $P$  到坐标原点的距离,  $\alpha$  为  $x$  轴正方向到极轴的角度, 所以  $\beta = \theta - \alpha$ 。可得:

$$\rho = \sqrt{x_p^2 + y_p^2}, \quad \alpha = \arctan(y_p/x_p), \quad x_p = \rho \cos \theta, \quad y_p = \rho \sin \theta \quad (4.4)$$

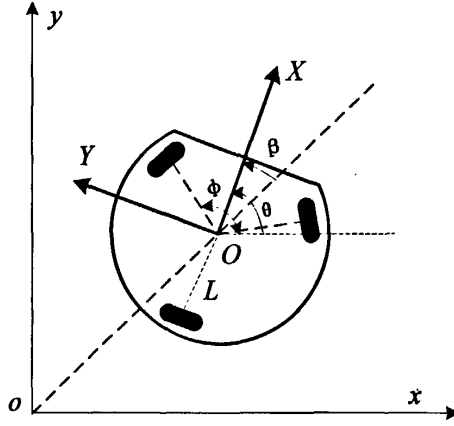


图 4.3 三轮全向机器人极坐标系下示意图

对式 (4.4) 两边进行求导, 其结果代入式 (4.1) 得极坐标下系统的方程为:

$$\begin{pmatrix} \dot{\rho} \\ \dot{\alpha} \\ \dot{\beta} \end{pmatrix} = \begin{pmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha / \rho & \cos \alpha / \rho & 0 \\ \sin \alpha / \rho & -\cos \alpha / \rho & 1 \end{pmatrix} \begin{pmatrix} \dot{x}_p \\ \dot{y}_p \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos \beta & \sin \alpha & 0 \\ \sin \beta / \rho & \cos \beta / \rho & 0 \\ -\sin \beta / \rho & -\cos \beta / \rho & 1 \end{pmatrix} \begin{pmatrix} v_x \\ v_y \\ \omega \end{pmatrix} \quad (4.5)$$

取如下的 Lyapunov 方程:

$$V = \frac{1}{2}(\lambda \rho^2 + \alpha^2 + \beta^2) \quad \lambda > 0 \quad (4.6)$$

显然  $V$  正定, 由式(4.5)可得:

$$\dot{V} = \dot{V}_1 + \dot{V}_2 \quad (4.7)$$

其中,

$$\begin{cases} \dot{V}_1 = \lambda \rho \cos \beta v_x + \alpha \frac{\cos \beta}{\rho} v_y \\ \dot{V}_2 = \beta(\omega - \frac{\sin \beta}{\rho} v_x - \frac{\cos \beta}{\rho} v_y) + \alpha \frac{\sin \beta}{\rho} v_x - \rho \sin \beta v_y \end{cases} \quad (4.8)$$

假设:

$$v_x = -k_1 \rho \cos \beta, \quad v_y = -k_2 \alpha \cos \beta \quad (4.9)$$

其中,  $k_1, k_2$  均为大于 0 的常数。把式 (4.9) 代入式 (4.8) 得:

$$V_2 = \beta \omega + (k_1 \beta - k_1 \alpha + k_2 \alpha \rho^2) \sin \beta \cos \beta + k_2 \alpha \beta \cos^2 \beta \quad (4.10)$$



$$\text{取 } \omega = k_3\beta - [k_1 - \alpha(k_1 + k_2\lambda\rho^2)]/\beta \sin\beta \cos\beta - k_2\alpha \cos^2\beta, \quad (k_3 > 0) \quad (4.11)$$

把式 (4.11) 代入式 (4.10) 得:

$$\dot{V}_2 = k_3\beta^2 \quad (4.12)$$

综上所述可得:

$$\dot{V} = -k_1\lambda\rho^2 \cos^2\beta - k_2\alpha^2 \cos^2\beta - k_3\beta^2 \leq 0 \quad (4.13)$$

因为  $V$  正定,  $\dot{V}$  为负定, 所以存在反馈控制律:

$$\begin{cases} v_x = -k_1\rho \cos\beta \\ v_y = -k_2\alpha \cos\beta \\ \omega = k_3\beta - [k_1 - \alpha(k_1 + k_2\lambda\rho^2)]/\beta \sin\beta \cos\beta - k_2\alpha \cos^2\beta \end{cases} \quad \lambda, k_1, k_2, k_3 > 0 \quad (4.14)$$

使系统 (4.5) 渐近稳定, 平衡点为  $(0 \ 0 \ 0)^T$ 。在  $\rho=0$  时,  $\alpha, \beta$  为任意值, 此时只需要对  $\theta$  进行控制即可, 可取如下的反馈控制律:

$$\begin{cases} v_x = 0 \\ v_y = 0 \\ \omega = -k\theta \end{cases} \quad (k > 0) \quad (4.15)$$

使系统稳定在平衡点。

### 4.3.2 姿态稳定控制仿真实验

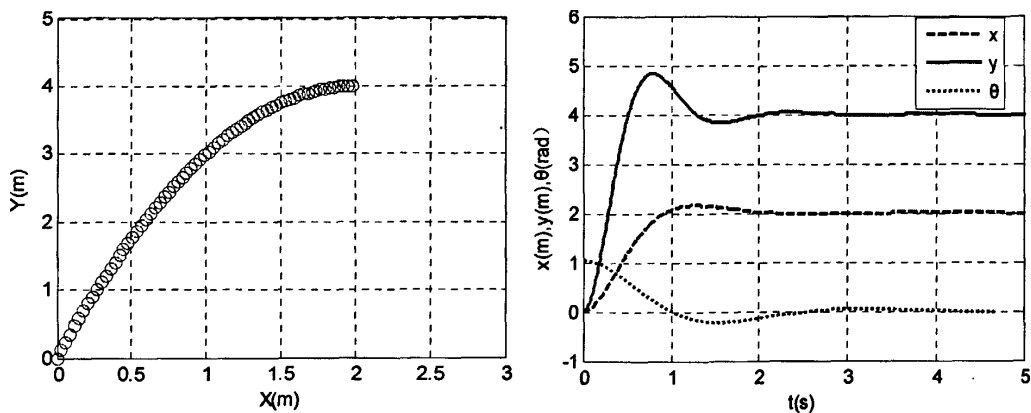


图 4.4 (a) 机器人运动轨迹

(b) 机器人中心坐标随时间变化的曲线

针对基于 Lyapunov 直接法的机器人姿态稳定控制算法, 本文进行了仿真研究 (仿真实验中底层驱动电机的控制算法采用第三章所设计的转速、电流双闭环控制算法)。设机器人的初始姿态为:  $q(0) = (0 \ 0 \ \pi/3)^T$ , 末姿态为:  $q(T) = (2 \ 4 \ 0)^T$ 。图 4.4(a) 为机器人的运动轨迹, 图 4.4(b) 为机器人中心坐标随时间变化的曲线。

仿真结果表明基于 Lyapunov 直接法的机器人姿态稳定控制算法能快速地实现三轮全向移动机器人的姿态稳定控制。

## 4.4 三轮全向机器人轨迹跟踪控制器设计

### 4.4.1 基于输入变换解耦的跟踪控制器设计

取三轮全向移动机器人的中心点  $O$  为参考点，机器人需要跟踪的轨迹参数方程为： $q_d(t) = (x_d(t), y_d(t))^T$ ， $t \in [0, T]$ ，且  $x_d(t)$ 、 $y_d(t)$  连续。

根据三轮全向移动机器人运动方式的特点，可以单独规划机器人正方向的轨迹。设机器人在正方向上的期望轨迹为  $\theta_d(t)$ ， $\theta_d(t)$  连续。则机器人中心参考点  $O$  沿  $x$  轴、 $y$  轴的期望速度和机器人期望角速度为：

$$v_{xd} = \dot{x}_d(t), v_{yd} = \dot{y}_d(t), \omega_d = \dot{\theta}_d(t) \quad (4.16)$$

将式(4.1)做如下的变换：

$$\begin{cases} u_1 = \cos\theta v_x - \sin\theta v_y \\ u_2 = \sin\theta v_x + \cos\theta v_y \\ u_3 = \omega \end{cases} \quad (4.17)$$

将式（4.17）代入到系统(4.1)可得：

$$\begin{cases} \dot{x} = u_1 \\ \dot{y} = u_2 \\ \dot{\theta} = u_3 \end{cases} \quad (4.18)$$

将  $(u_1 \ u_2 \ u_3)^T$  作为输入，系统方程变成了三个独立的积分器，从而实现了系统的解耦<sup>[32]</sup>。三轮全向移动机器人的跟踪误差为：

$$e(t) = \begin{pmatrix} x(t) - x_d(t) \\ y(t) - y_d(t) \\ \theta(t) - \theta_d(t) \end{pmatrix} \quad (4.19)$$

根据文献[33]，机器人中心点  $O$  在机器人坐标系下的跟踪误差为：

$$\begin{bmatrix} e_x \\ e_y \\ e_\theta \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_d - x \\ y_d - y \\ \theta_d - \theta \end{bmatrix} \quad (4.20)$$

$(e_x \ e_y \ e_\theta)^T$  为跟踪误差在机器人坐标系中的坐标。

将式(4.20) 对时间  $t$  求导，可得到如式（4.21）的误差模型：

$$\begin{bmatrix} \dot{e}_x \\ \dot{e}_y \\ \dot{e}_\theta \end{bmatrix} = \begin{bmatrix} \omega e_y + v_{xd} - v_x \\ -\omega e_x + v_{yd} - v_y \\ \omega_d - \omega \end{bmatrix} \quad (4.21)$$

式（4.21）中： $v_{xd}$ ， $v_{yd}$  为机器人坐标系下  $X$ 、 $Y$  方向的速度分量，它与全局坐标系下的速度  $\dot{x}_d$ 、 $\dot{y}_d$  有如下关系：

$$\begin{bmatrix} v_{xd} \\ v_{yd} \end{bmatrix} = \begin{bmatrix} \cos \theta_d & \sin \theta_d \\ -\sin \theta_d & \cos \theta_d \end{bmatrix} \begin{bmatrix} \dot{x}_d \\ \dot{y}_d \end{bmatrix} \quad (4.22)$$

轨迹跟踪控制的目的是要设计一个针对式(4.21)所表示模型的控制器,使三轮全向移动机器人在相应时刻,按全局坐标系下的实际位置和方位去跟踪理想轨迹。由式(4.21)所设计出的控制器为:

$$\begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} = \begin{bmatrix} v_{xd} + k_x e_x + \omega e_y \\ v_{yd} + k_y e_y - \omega e_x \\ \omega_d + k_\theta e_\theta \end{bmatrix} \quad k_x, k_y, k_\theta > 0 \quad (4.23)$$

式(4.23)中:  $k_x, k_y, k_\theta$  为正实数。将式(4.23)代入式(4.21)得到系统闭环误差模型为:

$$\begin{bmatrix} \dot{e}_x \\ \dot{e}_y \\ \dot{e}_\theta \end{bmatrix} = \begin{bmatrix} -k_x e_x \\ -k_y e_y \\ -k_\theta e_\theta \end{bmatrix} \quad (4.24)$$

由式(4.24)可知系统平衡点为坐标原点,并且在平衡点处是全局渐近稳定的。为了加速系统的收敛,可取:

$$k_x = \frac{1}{a_x + b_x |e_x|} + c_x, \quad k_y = \frac{1}{a_y + b_y |e_y|} + c_y, \quad k_\theta = \frac{1}{a_\theta + b_\theta |e_\theta|} + c_\theta \quad (4.25)$$

式(4.25)中,  $a_x, b_x, c_x, a_y, b_y, c_y, a_\theta, b_\theta, c_\theta$  均为大于零的正数。

距平衡点较远处的反馈控制律增益比较小,可防止控制器输出饱和;距平衡点比较较的地方反馈控制律增益相对比较大,可以加快系统的收敛速度。

#### 4.4.2 轨迹跟踪控制仿真实验

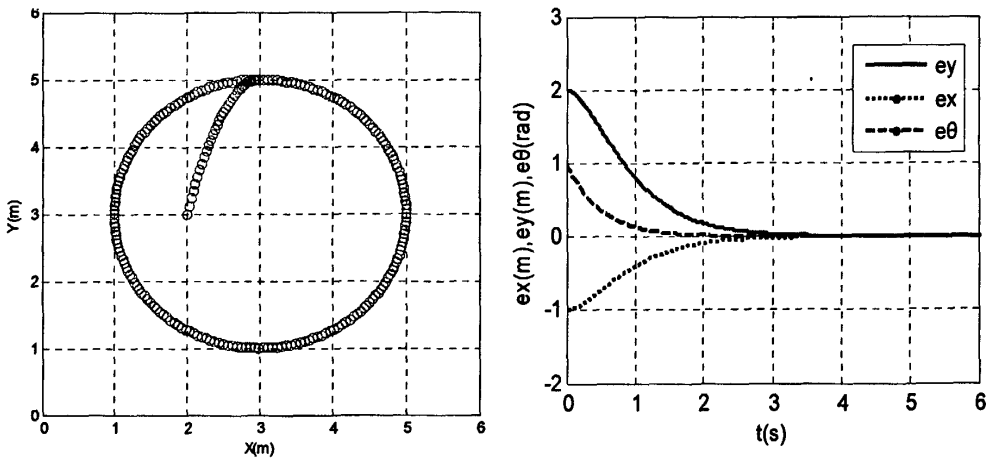


图 4.5 (a) 机器人跟踪圆形曲线运动轨迹 (b) 机器人跟踪圆形曲线误差

本文对基于输入解耦的全局轨迹跟踪控制算法进行了仿真研究(仿真实验中

底层驱动电机 BLDCM 控制算法采用第三章所设计的转速、电流双闭环控制算法），根据反复实验，本文取控制器的参数为： $a_x = a_y = 0.25$ ； $b_x = b_y = 0.15$ ； $c_x = c_y = 0.48$ ； $a_\theta = b_\theta = 0$ ； $c_\theta = 2$ 。

仿真实验验证了机器人跟踪圆形轨迹的控制效果。假定三轮全向移动机器人的初始姿态为： $q(0) = (2 \ 3 \ 3\pi/4)^T$ ，机器人期望正方向恒为 $\theta_d = 0$ 。圆形轨迹的参数方程为：

$$\begin{cases} x_d(t) = 2 * \cos \omega t + 3 \\ y_d(t) = 2 * \sin \omega t + 3 \end{cases}, \omega t \in [0 \ 2\pi] \quad (4.26)$$

图 4.5(a)、(b)分别给出了机器人跟踪圆形曲线的运动轨迹和跟踪误差随时间变化的曲线。仿真结果表明所设计反馈控制具有良好的轨迹跟踪效果和全局渐近稳定特性。

## 4.5 本章小结

本章首先介绍了轮式移动机器人的基本控制任务，然后分析了三轮全向机器人的控制特性，确定系统可控，应用 Lyapunov 直接法设计了机器人的姿态稳定控制算法；研究了移动机器人的轨迹跟踪，应用输入变换解耦算法，设计具有全局稳定的跟踪控制律，避免了以前采用线性方法处理非线性问题带来的局部稳定特性，使得设计过程简单。仿真结果表明了控制算法的可行性和系统的稳定性。

## 第五章 机器人底层运动控制系统设计

RoboCup 中型组足球机器人底层控制系统的基本任务是：接收上位机（笔记本电脑）传来的运动控制指令，然后根据接收到的运动指令控制机器人运行以实现规划的运动。良好的底层运动控制系统除了要满足快速性和准确性要求外，还要考虑以后机器人硬件平台的改进与更新的问题，化软件的模块、可移植性、可再升级性，以及整个底层运动控制系统的可移植性。

### 5.1 处理器单元选型

三轮全向移动机器人的运动控制指令由底层运动控制器发出，因此底层运动控制器芯片的选择尤为重要。

控制器芯片的处理能力和运算速度决定了机器人运动控制系统的整体性能，一款高性能机器人运动控制器芯片可将机器人整体水平提上一个新台阶<sup>[34]</sup>。在选择机器人运动控制器芯片时，需要考虑多个方面的因素，如控制器本身的性能及参数，开发环境等。通过对 Microchip, Motorola, Intel, TI 和 AD 等公司生产的一些型号的处理器的对比，本系统最终选择 TI 公司的 TMS320LF2407A 处理器作为三轮全向移动机器人运动控制的控制器，该芯片广泛应用于各类电机的控制，它具有灵活的指令集、内部操作灵活、高速的运算能力等优点，并且成本较低。TMS320LF2407A 是控制器是所有 240x 控制器中的一种功能最全的控制器，TMS320 系列 DSP 的体系结构专为实时信号处理而设计，它将实时处理能力和控制器外设功能集于一身，为控制系统应用提供了一个理想的解决方案。TMS320LF2407A 具有以下显著特点<sup>[35~36]</sup>：

- 采用高性能静态 CMOS 技术，使得供电电压降为 3.3V，减小了控制器的功耗；30MIPS 的执行速度使得指令周期缩短到 33ns(30MHZ)，从而提高了控制器的实时控制能力。
- 片内有高达 32K 字的 FLASH 程序存储器，高达 1.5K 字的数据/程序 RAM，544 字双口 RAM(DRAM)和 2K 字的单口 RAM(SRAM)。
- 两个事件管理模块 EVA 和 EVB，每个包括：两个 16 位通用定时器；8 个 16 位的脉宽调制 (PWM) 通道。它们能够实现：三相反相器控制；PWM 的对称和非对称波形；可编程的 PWM 死区控制以防止上下桥臂同时输出触发脉冲；3 个捕获单元；片内光电编码器接口电路；16 通道 A/D 转换器。事件管理器模块适用于交流电机、无刷直流电机、逆变器等等。

- 可扩展的外部存储器总共 192K 字空间：64K 字程序存储器空间；64K 字数据存储器空间；64K 字 I/O 寻址空间。
- 控制器局域网（CAN）2.0B 模块；串行通信接口（SCI）模块；基于锁相环的时钟发生器；高达 40 个可单独编程的或复用的通用输入/输出引脚（GPIO）。

## 5.2 运动控制系统的体系结构设计

足球机器人底层运动控制系统应具备与上位计算机、指南针模块的串口异步通信功能；能够对三轮全向移动机器人的三个驱动电机进行各自独立的速度闭环控制；并且具备自检功能。三轮全向移动足球机器人底层运动控制系统内部结构框图如图 5.1 所示。

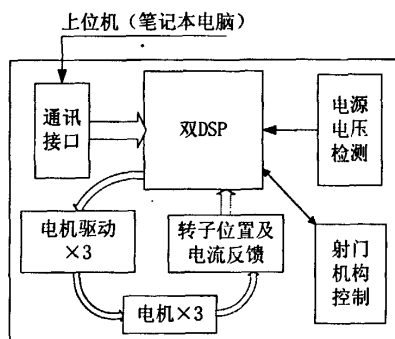


图 5.1 系统内部结构框图

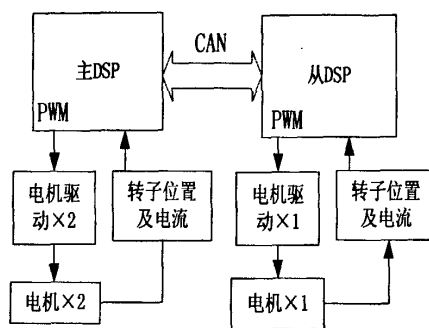


图 5.2 双 DSP 构架

每个 TMS320LF2407A 芯片具有两个事件管理器 EVA 和 EVB，EVA 和 EVB 各自有 8 个 16 位的 PWM 通道，而单个无刷直流电动机就需要 6 路 PWM 信号来驱动，单个 DSP 只能驱动两个无刷直流电动机，而中型组机器人的三轮全向移动机构有三个电机，并且还有其他外围事件，使得单个 DSP 芯片负担很重，实时性难以保证。为了解决此问题，本课题采用双 DSP 架构，如图 5.2 所示。系统中，3 个无刷直流电动机采取 3 个独立的控制信号，这样的优点是机器人各轮速可独立闭环控制，机器人转弯半径为 0，控制灵活，可实现全向移动。

## 5.3 机器人底层运动控制系统硬件设计

### 5.3.1 DSP 外围电路设计

本文只对机器人底层运动控制部分 DSP 外围电路进行了介绍，电路图如图 5.3 所示，包括 DSP 外部晶振电路、产生系统时钟锁相环电路、手动复位电路和 JTAG 接口电路。

DSP 2407A 的时钟既可以由外接晶振电路产生也可以由内部电部构成晶体

振荡器产生,由于机器人系统对运算处理速度要求较高,需要采用高频率的晶振,因此本系统 DSP 控制器的时钟由外接晶振电路提供。根据系统需要,设计了手动复位电路。锁相环电路和 JTAG 接口电路为 TI 公司的提供的 DSP 2407A 标准典型电路。

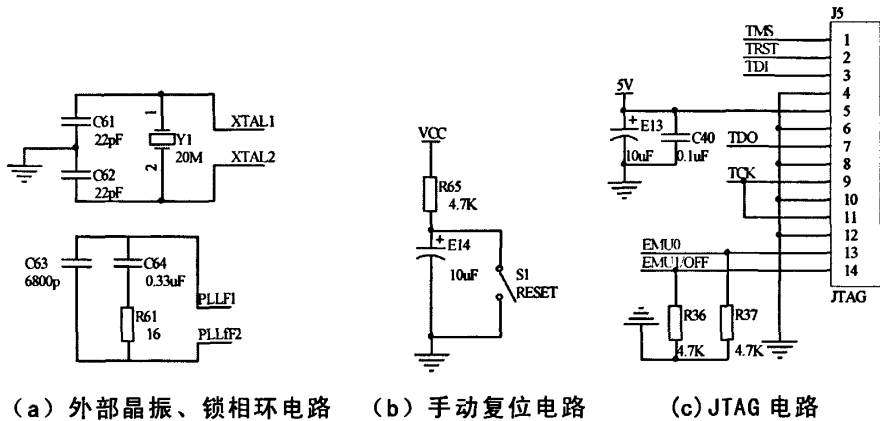


图 5.3 DSP 控制芯片接口电路设计

### 5.3.2 功率驱动及其逆变电路设计

机器人驱动电机采用 BLDCM, 本文设计了其功率驱动电路, 如图 5.3 所示。本系统采用 IR2132S 芯片驱动 BLDCM。IR2132S 正常工作的电源电压 VCC 为 +15V,要使它正常工作还需要接入一些外部器件,在电源 VCC 和 VB1, VB2, VB3 引脚之间分别接入三个快速恢复二极管 D1,D2, D3, VB1, VB2, VB3 引脚与 VS1, VS2, VS3 引脚之间分别接入三个自举电容 C1, C2, C3。

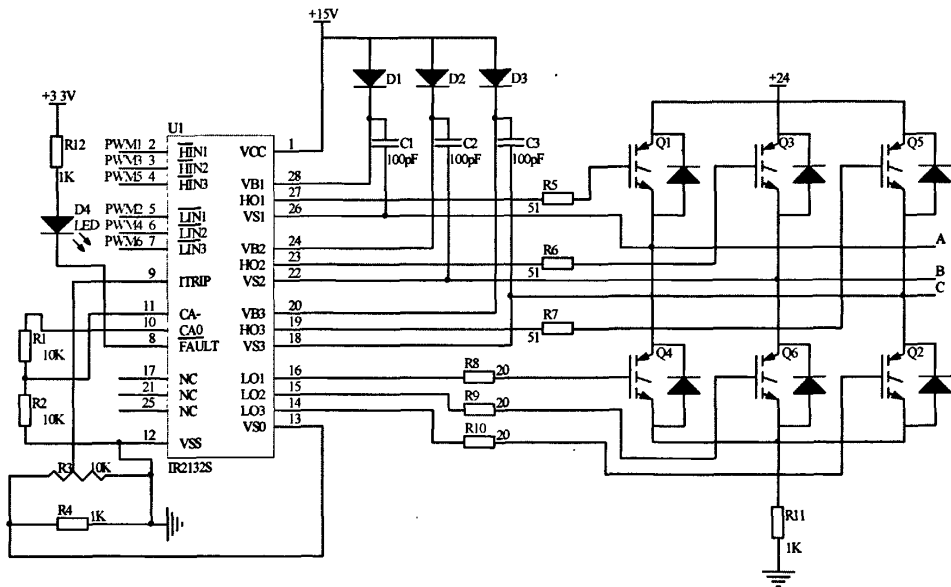


图 5.3 驱动电路

为了保证系统稳定地运行,在 BLDCM 驱动电路的基础之上实现了 BLDCM

的自保护、过电流及欠电压保护。电流检测电阻 R12 上的电压值与流入 CA-引脚的电流在反馈电阻 R2 上产生的电压相比较,若 R12 上的电压值超出设定参与电压值,则 IR2132S 内部电流保护生效,关闭 IR2132S 输出通道,驱动电路上的发光二极管 D4 点亮。如果负载或驱动电路出现过电流或欠电压的情况,IR2132S 内部可实现硬件保护,引脚 FAULT 输出到 DSP 的 PDPINT 引脚,把 PDPINT 引脚变为低电平,关闭 DSP 所有输出通道,置为高阻态,实现控制电路的保护。

下面对 IR2132S 的驱动原理<sup>[37~38]</sup>作简要介绍,当 Q4 导通、Q1 关断时,电源 VCC 经过快速恢复二极管 D1 和 Q4 给电容 C1 充电,以保证当 Q4 关断,需要开通 Q1 时,Q1 管的栅极能够依靠自举电容的储能来驱动其开通,实现自举式驱动。自举电容的大小决定了其充电后的储能是否能使功率管导通。电容过小则充电后能量不够,电容过大则会导致回路的充电时间过长,高频情况下电容充电不足,也不能使功率管导通。

由前面的分析可知:要使保证电路正常工作还必须选择合适大小的自举电容 C 和快速恢复二极管 D。在上桥臂上的 IGBT 导通时,快速恢复二极管上承受了很高的反向电压,本采用 MUR160 作为快速恢复二极管,其耐压值为 600V,在上桥臂 IGBT 导通时,自举电容上有能量损失,为了减小能量损失,在选择快速恢复二极管时,还得考虑其恢复时间。自举电容的大小由 IGBT 的栅极驱动要求及最大开通时间决定,自举电容 C 悬浮电压值与电源 VCC 的电压值是相同的,要保证 IGBT 完全导通,其栅极电压必须足够高。由于 IR2132S 上桥臂有驱动信号的欠压自锁功能,因此,自举电容在为 IGBT 提供导通所需要的电压后,还必须维持电压在 8.3V 以上才不至于使 IGBT 关闭。电源 VCC 与 VSS 之间电容(电源电容)和输出端的自举电容的匹配也很重要,一般情况下电源电容应为自举电容的 10 倍以上。在绘制系统的 PCB 电路时,应尽量使电源电容的布线应靠近芯片,自举电容布线尽可能靠近 IGBT。

系统中功率开关器件 IGBT 采用英飞凌公司生产的 IGBT 模块 FS15R06VXE3,其内部集成 6 个 IGBT,连接成三相全控制桥式电路,并且包含了检测温度要用到的热敏电阻。每一个 IGBT 管的集电极与发射极之间所能承受的最大电压为 600V,最大集电极电流为 15A,峰值电流为 30A,导通延时只有 0.014 $\mu$ s,关断延时只有 0.1 $\mu$ s,完全能满足 BLDCM 的驱动电路设计要求。

### 5.3.3 电流检测

为使机器人驱动电机 BLDCM 有较好的动态性能以及对 BLDCM 三相相电流进行监控,需要对 BLDCM 三相相电流进行检测。电流检测的目的主要是用于和参考电流相比较,改变输出 PWM 占空比,实现 BLDCM 双闭环调速;另外对



BLDCM 进行过流保护。一般情况下可采用以下两种方法检测电流：

(1)采用霍尔电流传感器实现电流检测；

(2)在 IGBT 逆变桥下桥臂与地之间串联电阻，将电流信号转化成电阻上的电压降。霍尔电流传感器是利用霍尔效应制成的隔离式电流传感器，它能有效地实现主回路与控制回路的电气隔离，然而霍尔电流传感器成本较高。第二种方法是直接将 BLDCM 三相相电流转化为电阻上的电压降送给控制电路，简便易行，电阻上的电压降正比于电机相电流。因而本课题采用串联电阻来检测 BLDCM 三相绕组相电流。机器人驱动电机 BLDCM 最大工作电流为 7.47A，串联电阻采用 0.1  $\Omega$ ，3W 的精密电阻，当额定相电流流过电阻时，最高电压为 0.747V。因为 DSP 2407A 的 ADC 输入信号范围为 0~3.3V 模拟输入电压，故串联电阻上电压信号需要放大，电路中采用运算放大器 LM258 将输入电压放大 4 倍。其电路如图 5.4 所示，其中 IN 是电机相电流经过采样电阻的信号，ADCIN0 为 DSP 模数转换器 ADC 模块的输入信号。

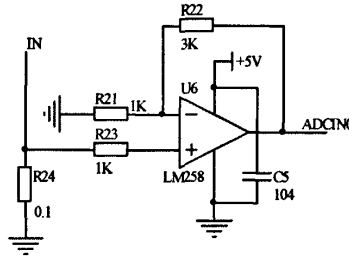


图 5.4 电流检测电路

令  $u_i$  为采样电阻 R24 上的电压降， $u_o$  为 LM258 的输出电压，由于 LM258 工作在同相输入状态，因此：

$$u_o = A_u u_i = \left(1 + \frac{R_{22}}{R_{21}}\right) u_i = \left(1 + \frac{3k}{1K}\right) u_i = 4u_i \quad (5.1)$$

输入模拟电压经 DSP 2407A 的 10 位 A/D 得到电压数字值公式为：

$$\text{数字值} = 1023 * \text{输入模拟电压值} / 3.3 \quad (5.2)$$

A/D 转换得到的数字值保存在 ADC 结果寄存器中，通过读取寄存器的值，通过计算就可以得 BLDCM 相电流的值。

当 BLDCM 相电流达到最大工作电流 7.47A（所选 BLDCM 最大工作电流）时，采样电阻 R24 上的压降达最大，为 0.747V，经 LM256 放大之后，输入 DSP 的最大电压为 2.988V，此时的电流可表示为：

$$\text{数字值} = 1023 * 2.988 / 3.3 = 926.28 \approx 926 \quad (5.3)$$

由式 (5.3) 可知，当 ADC 检测到数字值超过 926 时，即输入电压大于 2.988V 时，就认为 BLDCM 过载，输出中断信号，系统响应中断，将 DSP F2407 的 PWM 输出引脚全部拉为高阻态，使系统停止，达到过流保护的作用。

### 5.3.4 位置检测及转速计算

BLDCM 转子位置检测主要用于 BLDCM 换相控制和转速计算，因此位置检测是相当关键的。由三相六状态 BLDCM 的控制原理可知，要想得到恒定的、最大的转矩，就必须对 BLDCM 进行换相控制，在准确的时刻进行换相，减小转矩的波动。

BLDCM 转子位置的确定是通过安装在无刷直流电动机上霍尔传感器来检测的。霍尔传感器采用集电极开路输出，因此，其输出端必须经过上拉电阻才能有效地输出 BLDCM 转子位置方波信号，然后经过光耦隔离器件输出到 DSP 2407A EVA 模块的 CAP1~CAP3，对 BLDCM 转子位置信号进行捕捉。由于 DSP 的 I/O 口输入输出高电平都是 3.3 V，因此输入到 CAP1~CAP3 的 BLDCM 霍尔位置传感器信号也必须是 3.3 V。本课题中采用高速光电隔离器件 Q817 将 BLDCM 三路 5V 霍尔信号转化为 3.3 V 信号，同时实现机器人驱动电机与 DSP 控制器间的电气隔离。其应用电路如图 5.5 所示。

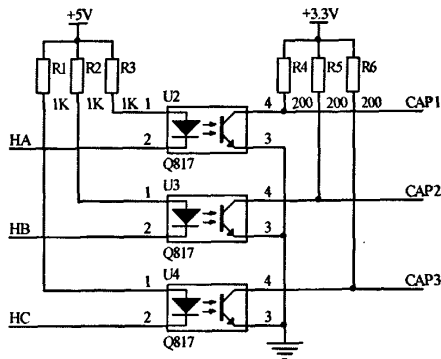


图 5.5 霍尔信号采集电路图

BLDCM 转子位置信号除了用于电机准确换相，还用于电机转速的计算，BLDCM 每转过一周需要进行六次换相，即转子每转过  $60^\circ$  进行一次换相。因此，只要测得 BLDCM 两次换相之间的时间间隔  $\Delta t$ ，就可以由式 (5.4) 计算出两次换相间隔期间的 BLDCM 转动的平均角速度：

$$\omega = 60^\circ / \Delta t \quad (5.4)$$

BLDCM 两次换相之间的时间间隔  $\Delta t$  可以通过捕捉转子位置信号捕获中断时定时器 2 的 T2CNT 寄存器的值来得到。设置定时器 2 为连续增计数模式，转子转速越低， $\Delta t$  越长，T2CNT 寄存器中的值就越大。假定定时器 2 周期值为 0xFFFFH，预分频值为 1/128，因此转子每转过  $60^\circ$  所需要的最长时间为(计数时钟周期为 50ns)：

$$50ns \times 128 \times 2^{16} = 0.4194s \quad (5.5)$$

BLDCM 转过一周所需的最长时间为：

$$6 \times 0.4194s = 2.5165s \quad (5.6)$$

BLDCM 最小平均转速为：

$$(60/2.5165)r/min = 23.84r/min \quad (5.7)$$

当  $T2CNT=0xFFFFH$  时，BLDCM 对应的转速为  $23.84r/min$ ； $T2CNT=x$  时，BLDCM 对应转速为  $23.84r/min$  的  $0xFFFFH/x$  倍，计算所得到的速度值作为速度反馈量和转速调节计算。

### 5.3.5 保护电路设计

当机器人驱动电机出现过电压或欠电压时，必须使 BLDCM 停转，因此机器人底层运动控制系统还需要设计驱动电机过电压及欠电压保护电路。驱动电机的最大工作电压为 36V，因此 BLDCM 输入电压超过 36V 时，需进行过电压保护，封锁 DSP 控制器 PWM 输出信号；同理，BLDCM 输入电压低于 12V 时，需进行机器人驱动电机欠电压保护。其硬件实现电路如图 5.6。

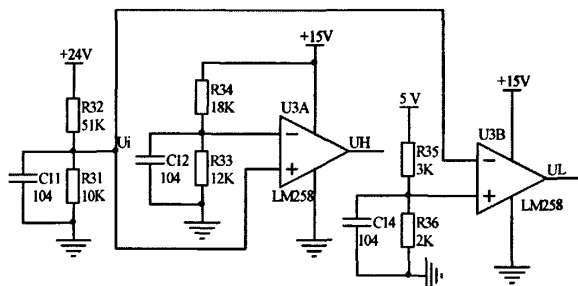


图 5.6 过压欠压保护电路

直流母线+24V 电压经过分压电阻 R31, R32 得到 4V 的输入电压  $U_i$ ，然后分别输入电压比较器 U3A 和 U3B 作电压比较。U3A 用用过电压比较，当  $U_i$  小于 U3A 负向输入电压+6V 时，U3A 输出端 UH 值为 0；当直流母线电压超过 36V 过电压时， $U_i$  大于负向输入电压，U3A 输出端 UH 值为+15V。当过压即 UH 为+15V 时，将其转换成低电平，而后输入到 DSP2407A 的 PDPINTA 引脚上，实现过电压保护<sup>[39]</sup>。U3B 实现欠压保护，其原理和过压保护类似，这里不再说明。

## 5.4 机器人底层运动控制系统软件设计

机器人底层运动控制系统要正常工作，不仅需要稳定可靠的硬件平台，而且还必须要有高质量的软件部分和合理的控制算法，这样才能组成一个完整的控制系统<sup>[40]</sup>。本课题采用采用模块化设计思想对机器人底层运动控制系统进行软件设计，软件部分包括主程序模块、各中断服务子程序模块和控制算法的实现。中断服务子程序模块主要包括 BLDCM 转子位置捕获中断模块、串行口中断模块、ADC 转换结束中断模块几部分。

系统控制程序的编写是在 TI 公司的 DSP 芯片集成开发环境 CCS(Code

Composer Studio)中进行的, 版本为 CCS2000(针对 2000 系列 DSP)<sup>[41]</sup>。

### 5.4.1 主程序设计

主程序主要完成变量声明、系统的初始化、设置中断向量表等。主程序流程如图 5.7 所示。

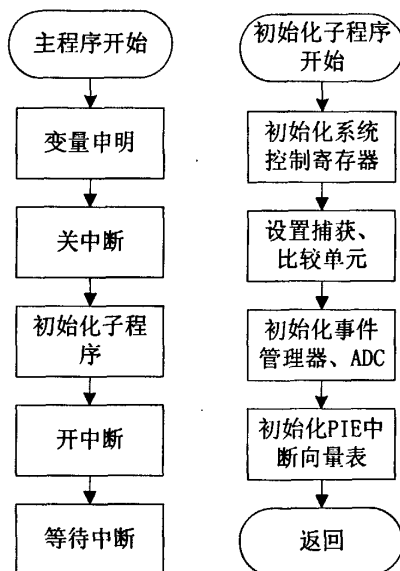


图 5.7 主程序流程图

系统初始化主要包括：

**PLL 初始化：**锁相环 (PLL) 模块给底层控制系统提供各种所需时钟信号，初始化主要的主要工作是对 PLL 控制寄存器 CKCRO 和 CKCR 进行正确配置。本课题中系统外部晶振为 10MHz，而 TMS320LF2407A 正常工作频率为 40MHz，因此需要将 PLL 设置为 4 倍频。

**WD 初始化：**TMS320LF2407A 包含了一个看门狗(WD)模块用来监视底层运动控系统软件的运行，当系统软件进入死循环，或当 TMS320LF2407A 发生暂时混乱，则 WD 定时器溢出，产生系统复位。

**I/O 初始化：**TMS320LF2407A 大多数 I/O 都时分时复用，所以在 I/O 口初始化程序里面需要对所用到的 I/O 功能进行指定，这部分工作通过对 I/O 复用控制寄存器以及数据和方向控制寄存器进行设置来完成。

**GP Timer 初始化：**TMS320LF2407A 有 4 个通用可编程定时器(GP)，本系统设置 GP1 为连续递增计数模式，控制 PWM 输出占空比，设置 GP2、GP4 为定向增/减计数器模式，记录无刷直流电机霍尔传感器的脉冲数，设置 GP3 为连续递增计数模式，用作定时器控制定时器中断子程序中中断周期。

**SCI 初始化：**TMS320LF2407 的 SCI (串行异步通信接口) 初始化包括：把相应的 I/O 配置成具有 SCI 特殊功能，所有设置都是通过相对应的 SCI 控制寄存

器实现。

系统中断向量表存放中断服务子程序的入口地址,每个中断都有自己的中断向量,在系统初始化时,对中断向量表进行定位,确保中断响应时能够准确的进入中断服务子程序。在初始化的过程中,为了防止中断的意外到来,在主程序的开始处关中断,完成初始化后,开中断。

5.4.2 中断服务程序设计

(1) 转子位置捕获中断服务子程序

本课题所研究的三轮全向移动机器人采用 BLDCM 驱动,机器人驱动电机转速控制是底层运动控制的中心环节,而 BLDCM 的转速是通过检测转子的位置计算得到的。BLDCM 稳定运行时需要及时、准确地检测转子位置信号,以此来确定 PWM 电路相应输出状态,从而 BLDCM 定子绕组正确换相。系统中 BLDCM 转子位置信号(电机上的霍尔传感器信号)采用捕获中断的方式来获得。TMS320F2407A 的每个事件管理器中有三个捕获单元 CAP1, CAP2, CAP3,每个捕获单元都能产生中断,且具有各自独立的中断向量。将 CAP1, CAP2, CAP3 设置成捕获外部信号上下沿跳变功能,分别去捕获 BLDCM 上霍尔传感器输出的三路位置信号上下跳沿。BLDCM 运行时,对于每路转子位置信号,电机的每旋转 120 度电角度,其上下跳沿就会触发一次事件管理器的该路位置信号的捕获中断。BLDCM 转子位置捕获中断服务子程序流程如图 5.8。

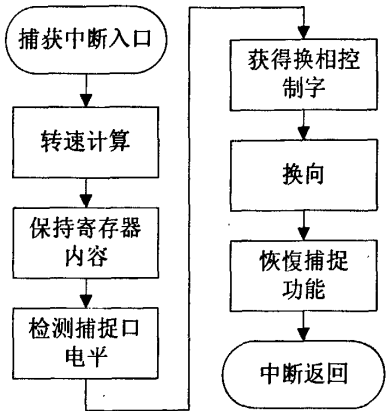


图 5.8 位置捕获中断流程图

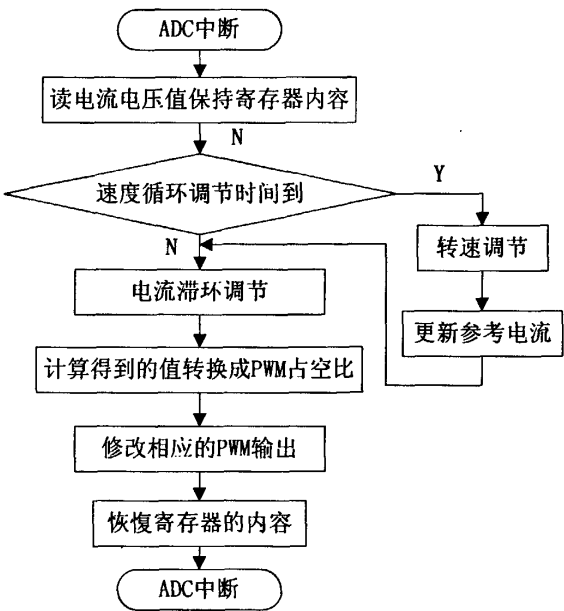


图 5.9 ADC 单元中断子程序

(2) ADC 转换结束中断

每次 PWM 周期发生后,可以通过计算 ADC 转换的数值得到当前相电流值

和相电压值，然后判断是否进行速度环的调节，再进行电流环的调节。本系统电流采样周期为 50us，速度采样周期为 2ms，所以每运行 40 次的电流环调节，就进行一次速度环调节。ADC 单元中断子程序框图如图 5.9 所示。在速度环中采用单神经元 PID 控制，电流环采用电流滞环控制，转速环的输出值为电流环的参考电流值。而电流环的调节过程也就是 PWM 的产生过程。

(3) 串口异步通信 SCI 中断子程序

上位机（笔记本电脑）与主 DSP 2407A 通信的指令帧格式如表 5.1 所示。

表 5.1 指令帧格式

0	1	2	3	4
帧头	射门力度	1 号电机给定转速值	2 号电机给定转速值	3 号电机给定转速值
0xFF	0~250	0~254	0~254	0~254

指令帧由 5 个字节数据组成，数据帧第一字节为应答信号；数据帧第二字节为射门力度值，该值由上位机的射门决策库给定；数据帧第三字节到第五字节分别为机器人三个驱动电机速度给定值。DSP 2407A 每接收完一个数据帧，通信接收程序即将该指令帧保存起来，并将通信接收标志 ReceValid 置 1，通知上位机指令接收完毕。程序流程图如图 5.10 所示。

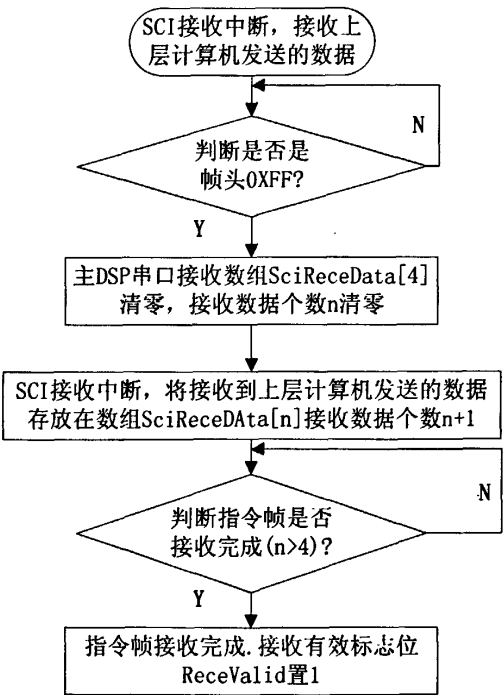


图 5.10 SCI 串行异步通信流程图

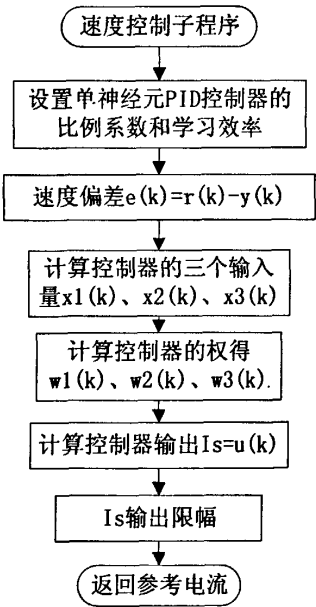


图 5.11 单神经元控制算法流程

### 5.4.3 电机控制算法的实现

#### (1) 速度环的控制

机器人驱动电机的转速采用单神经元自适应 PID 控制法，具体设计细节见第三章。速度环的输入参数一个是给定转速，另一个是实际测得转速信号，输出是调整好的速度信号供电流闭环输入。单神经元自适应 PID 控制算法的流程如图 5.11 所示。

#### (2) 电流环的控制

机器人驱动电机电流调节器采用电流滞环跟踪型 PWM 控制<sup>[42]</sup>，其突出特点是控制原理简单，且易于实现。当功率器件的开关频率很高时，滞环控制器响应速度非常快、带内部限流，并且对负载和电路参数的变化不敏感。传统的 PI 控制虽然具有较高的稳态精度高，但其参数整定比较困难。研究表明，滞环控制具有较好的系统稳定性和快速性。由于 BLDCM 相电流滞环跟踪控制的软件实现比较简单，这里就不再作进一步介绍。

## 5.5 本章小结

本章根据三轮全向移动机器人底层运动控制系统的基本任务，详细介绍了系统主要硬件与软件部分的实现。其中硬件部分详细介绍了 DSP 外围电路的设计(包括外部晶振及锁相环电路、手动复位电路、JTAG 电路设计)，BLDCM 功率驱动及其逆变电路的设计，检测电路(BLDCM 转子位置信号检测、BLDCM 相电流检测)和保护电路的设计。软件部分，采用模块化设计思想，介绍了主程序模块和中断服务子程序模块(转子位置捕获中断模块、SCI 串行口通信中断模块、ADC 转换结束中断模块)，另外，对 BLDCM 转速控环和电流控制环的控制算法进行了阐述，并设计了流程框图。

## 总结与展望

本论文以 RoboCup 中型组机器人足球赛为应用背景,对三轮全向移动机器人运动控制系统进行了研究,并在此基础上对三轮全向移动机器人运动系统的控制方案、控制算法以及底层运动控制系统软、硬件设计进行了详细的描述。

综合起来,本论文主要完成以下工作:

1. 建立了三轮全向移动机器人的普遍运动学和动力学模型,为分析中型组机器人的运动特性和控制特性以及设计运动控制器提供了系统模型。分析比较了基于运动学模型的分层控制和基于动力学模型控制的优缺点,并结合实验室的现有条件和机器人的特点,本文决定采用基于运动学的分层控制方案实现机器人的运动控制。

2. 以 BLDCM 作为机器人的驱动电机,分析了 BLDCM 数学模型,并建立基于 Matlab/Simulink 的 BLDCM 转速、电流双闭环控制系统仿真模型,速度环采用单神经元自适应 PID 控制算法,电流环采用滞环控制。仿真结果和分析表明:波形符合理论分析、控制性能好、满足系统控制的要求。为三轮全向移动机器人底层运动控制系统设计提供了理论指导。

3. 介绍了轮式移动机器人的基本控制任务。设计了三轮全向机器人的运动控制算法,包括姿态稳定控制算法和轨迹跟踪算法,并证明了算法的稳定性。仿真实验验证了控制算法的有效性。

4. 以 DSP 芯片 LF2407A 作为核心,设计了移动机器人运动控制系统的整体结构及主要硬件电路。根据系统功能采用模块化方法进行了软件设计,包括主程序;速度捕获、SCI 串行通信、ADC 转换中断服务子程序以及电机控制算法实现等内容,并给出了主要的程序流程图。

经过前面章节的详细叙述,论文从应用的角度,以 RoboCup 中型组机器人足球比赛为景,对三轮全向足球机器人运动控制系统进行了研究。机器人系统正朝着高速、高精度、开放化、智能化、网络化发展,以及工作环境复杂度的提高,对机器人运动控制系统的要求越来越高,其研究内容和系统结构也日益广泛和复杂,新技术和新方法也不断出现。由于工作基础、工作时间和实验条件的限制,本文的研究工作还需进一步的深入和完善。今后可在以下几个方面作进一步的研究:

1. 机器人动力学模型参数的整定和基于动力学模型的控制系统设计。
2. 在有障碍环境下,全向机器人运动控制算法的研究。



## 参考文献

- [1] 王文学, 徐心和. 机器人足球: 高科技的对抗[J], 百科知识, 2000.10:16-17.
- [2] 范永, 谭民. 机器人控制器的现状及展望[J]. 机器人, 1999, 21(1):75-77.
- [3] 徐国保, 尹怡欣, 周美娟. 智能移动机器人技术现状及展望[J]. 机器人技术与应用, 2007(2):29-34.
- [4] Farzad P, Mattias P K. Adaptive control of dynamic mobile robots with nonholonomic constraints[J]. Computers and Electrical Engineering, 2002, 28 (3) :241-253.
- [5] G.Oriolo A.De Luca,M.Venditelli. Control of Wheeled Mobile Robots: An Experimental Overview[J], RAMSETE: Articulated and Mobile Robotics for Services and Technologies.2001:181-226.
- [6] A.K.Mack worth. On seeing robots. Computer Vision: System, Theory and Applications[M], World Science Press Singapore, 1993: 1-13.
- [7] 邱秉权. 分析力学[M]. 北京: 中国铁道出版社, 1998年8月第一版.
- [8] Tomas Kumar-Nagy, Rafaela D Andrea, Puritan Gangly: Near-optimal dynamic trajectory generation and control of an omni-directional vehicle[J]. Robotics and Autonomous Systems, 2003, 46:47-64.
- [9] Raul Rojas. Omni-directional Control[M]. Freie University Berlin, 2005, 09.
- [10] 牟学刚, 朱劲, 蒋平. 三轮全向足球机器人结构设计与系统模型研究[J]. 机械与电子, 2006, 6(6):38-41.
- [11] Krishnan H and Mc.CTamnxh N H. Tracking in nonlinear differential algebraic control systems with applications to constrained robot systems [ J ]. Automatica. 1994, 30 (12):1885-1897.
- [12] 龚振邦. 等. 机器人机械设计[M]. 北京: 电子工业出版社. 1995.
- [13] Lages, W. F. and Hemerly, E. M. Smooth Time-invariant Control of Wheeled Mobile Robots[C]. In 7th International Conference on Systems Science, Wroclaw, Poland, September, 1998.
- [14] J. M. Yang and J. H. Kim. Sliding Mode Control for Trajectory Tracking of Non-homonymic Wheeled Mobile Robots[J]. In IEEE Transactions on Robotics and Automation, 1999, 1(15):578-587.
- [15] 张深. 无刷直流电动机原理及应用[M]. 北京: 机械工业出版社. 2004. 第2版
- [16] 韩安太, 刘峙飞, 黄海编著. DSP控制器原理及其在运动控制系统中的应用

- [M].北京:清华大学出版社, 2003, 10.
- [17] PillayP, Krishnan R. Modeling, simulation, and analysis of permanent-magnet motor drives, Part II: The brushless DC motor drive[J]. IEEE Trans. on Industry Applications,1989, 25(2): 274-279.
- [18] 纪志成.基于Matlab无刷直流电机系统仿真建模的新方法[J].系统仿真学报.2003. 12(15): 1745-1748.
- [19] Evans PD, Brown D. Simulation of brushless DC drive[J]. IEEE Pt.B. Elect. Power PROC. APPI(UK)Sep,1990,137:290-298.
- [20] Jeon Y S, Mok H.S, Choe G.H, et al. A New Simulation Model of BLDC Motor with Real Back EMF Waveform[J]. Proceeding from Computers in Power Electronics,2000, 18(2): 17-20.
- [21] Thomas Salem, Tim A Haskew. Simulation of the Brushless DC machine[J]. Proceeding from System Theory,1995,12 (14):18-22.
- [22] 薛定宇, 陈阳泉. 基于Matlab/Simulink的系统仿真技术与应用[M]. 北京: 清华大学出版社.2002.
- [23] Ionel D M, Eastham J F, Betzer T. Finite element analysis of a novel brushless DC motor with flux barriers[J]. IEEE Trans on Magnetics, 1995 , 31(6):3749-3751.
- [24] Jeon Y S, Mok H S, Choe G.H. A new simulation model of BLDC motor with real back EMF waveform[J]. Proceeding from Computers in Power Electronics, 2000, 16(18):217-220.
- [25] 王季铁, 曲家骐. 执行电动机[M]. 北京: 电子工业出版社, 1997: 74-91.
- [26] 刘金琨. 先进PID控制MATLAB仿真[M]. 北京: 电子工业出版社, 2004.
- [27] 张世韬, 杨风, 郝骞. 单神经元PID控制器研究及仿真[J]. 机械工程与自动化,2009,154(3):69-70.
- [28] 季秀才.RoboCup小型组机器人运动控制研究, 国防科技大学硕士学位论文, 2003.
- [29] Murray, R., Li, Z. and Sastry, S. A Mathematical Introduction to Robotic Manipulation[M]. CRC Press, 1994
- [30] 胡跃明.非线性控制系统理论与应用[M], 国防工业出版社, 2002
- [31] Aicardi M, Casalino G., Bicchi A, Balestrino A. Closed Loop Steering of Unicycle-like Vehicles via Lapunov techniques[J]. In IEEE Robotics & Automation Magazine, 1995, 2(1).
- [32] Warren E D ,Darren M D ,Erkan Z ,et al. Nonlinear control of wheeled mobile robots [ M ] . London: Springer , 2001.

- [33] 李艳, 林廷圻, 黄玉美, 高峰. 基于神经网络的3自由度移动机器人跟踪方法研究[J]. 西安交通大学学报, 2003.37(7):711-714.
- [34] 王学慧, 柳林. 足球机器人小车总体设计思想及其单片机选型[J], 机器人技术与应用, 2001 (5):10-14.
- [35] 刘和平, 严利平, 张学锋, 卓清锋. TMS320LF240XDSP结构原理及应用[M], 北京:北京航空航天大学出版社, 2002.
- [36] 王晓明, 王玲. 电动机的DSP控制-TI公司的DSP应用[M]. 北京:北京航空航天大学出版社, 2004.
- [37] 黄永志, 陈卫东. 两轮移动机器人控制系统的设计与实现[J]. 机器人, 2004.26(1):40-44.
- [38] 谢运祥. IR2130驱动器及其在逆变器中的应用[J]. 微电机, 2001, 34(2):50-52.
- [39] 张培仁, 张志坚. 16/32位DSP机器人控制系统设计与实现[M]. 北京:清华大学出版社, 2006.
- [40] 谈世哲, 梅志千, 杨汝清. 基于DSP的工业机器人控制器的设计与实现[J]. 机器人, 2002, 24(2):134-137.
- [41] 彭启琼, 管庆等编著. DSP集成开发环境[M]. 北京:电子工业出版社, 2004
- [42] Chapuis Y A, Roye D. Direct Torque Control and Current Limitation Method in Start up of an Induction Machine[J]. Power Electronics and Variable Speed Drives, Conference Publication, 1998, 456(3):21-23.

## 致 谢

在本论文即将完成之际，衷心的感谢我的导师李益华教授，本文正是在他的悉心指导下完成的。从论文前期材料的准备、选题、撰写到最后的定稿，李老师都花费了大量的时间和精力，给予了我很大的指导和帮助。李老师严谨的治学态度、勤恳的工作作风、平易近人的工作态度都值得我在今后的工作和学习中学习和借鉴，也将使我终生受益。李老师不仅在学习上督促我，在生活上也给予了极大的关怀，在此，特向恩师表示深深的敬意和由衷的感谢！同时感谢几年来电气与信息工程学院各位老师给予我的支持和帮助！值此学业完成之际，谨怀着崇敬、激动的心情向培养我多年仍在辛勤耕耘的导师表示由衷的感谢和诚挚的祝愿！

论文的顺利完成，还得力邓本再教授给予我接触机器人的机会和悉心的指点，在此向他们表示衷心的感谢。

感谢上海英集斯自动化技术有限公司的赵伟、张飞工程师。是他们的帮助和支持，课题相关项目才得以顺利进行，论文才得以完成！

课题的完成和论文撰写过程中，得到了同学黄苗、师弟王国伟无私的帮助，在此一并表示感谢！

感谢我的家人，正是父母在背后的默默支持和鼓励，激励我认真地过每一天，不断学习，不断进步。

最后，向百忙之中审阅论文的各位专家、教授致以崇高的敬意和深深的谢意！

## 附录 A （攻读硕士学位期间发表论文目录）

- [1] 谢志诚，李益华，林利红.无刷直流电机双闭环控制系统的建模与仿真，计算技术与自动化.
- [2] 谢志诚，李益华，林利红.基于 ARM 的无刷直流电动机无传感器控制系统，伺服控制.

## 附录 B （Matlab 仿真的 S 函数源程序）

参考电流 S 函数（current\_ref.m）

```
function [sys,x0,str,ts] = current_ref(t,x,u,flag)
```

```
switch flag,
```

```
    case 0,
```

```
        [sys,x0,str,ts]=mdlInitializeSizes;
```

```
    case 3,
```

```
        sys=mdlOutputs(t,x,u);
```

```
    otherwise
```

```
        error(['Unhandled flag = ',num2str(flag)]);
```

```
end
```

```
function [sys,x0,str,ts]=mdlInitializeSizes
```

```
sizes = simsizes;
```

```
sizes.NumContStates = 0;
```

```
sizes.NumDiscStates = 0;
```

```
sizes.NumOutputs = 3;
```

```
sizes.NumInputs = 2;
```

```
sizes.DirFeedthrough = 1;
```

```
sizes.NumSampleTimes = 1;
```

```
sys = simsizes(sizes);
```

```
x0 = [];
```

```
str = [];
```

```
ts = [0 0];
```

```
function sys=mdlOutputs(t,x,u)
```

```
Is=u(1);
```

```
pos=u(2);
```

```
if(position>=0&&position<pi/3)
```

```
    iar=Is;    ibr=-Is;    icr=0;
```

```
elseif(position>=pi/3&&position<2*pi/3)
```

```
    iar=Is;    ibr=0;    icr=-Is;
```

```
elseif(position>=2*pi/3&&position<pi)
```

```
    iar=0;    ibr=Is;    icr=-Is;
```

```

elseif(position>=pi&&position<4*pi/3)
    iar=-Is;    ibr=Is;    icr=0;
elseif(position>=4*pi/3&&position<5*pi/3)
    iar=-Is;    ibr=0;    icr=Is;
elseif(position>=5*pi/3&&position<=2*pi)
    iar=0;      ibr=-Is;   icr=Is;
end
sys = [iar,ibr,icr];

```

反电动势 S 函数 (efun.m)

```

function [sys,x0,str,ts] = efun(t,x,u,flag,P,K)
switch flag,
case 0,
    [sys,x0,str,ts]=mdlInitializeSizes;
case 1,
case 3,
    sys=mdlOutputs(t,x,u,P,K);
case 4,
otherwise
    error(['Unhandled flag = ',num2str(flag)]);
end
function [sys,x0,str,ts]=mdlInitializeSizes
sizes = simsizes;
sizes.NumContStates = 0;
sizes.NumDiscStates = 0;
sizes.NumOutputs = 3;
sizes.NumInputs = 2;
sizes.DirFeedthrough = 1;
sizes.NumSampleTimes = 1;
sys = simsizes(sizes);
x0 = [];
str = [];
ts = [0 0];
function sys=mdlOutputs(t,x,u,P,K)
position=u(1);

```

```

 $\omega = u(2)/P;$ 
per=fix(position/(2*pi))*2*pi;
if(position>=0&&position<pi/3)
    ea=K* $\omega$ ;    eb=-K* $\omega$ ;    ec=K* $\omega$ *((per-position)/(pi/6)+1);
elseif(position>=pi/3&&position<2*pi/3)
    ea=K* $\omega$ ;    eb=K* $\omega$ *((position-pi/3-per)/(pi/6)-1);    ec=-K* $\omega$ ;
elseif(position>=2*pi/3&&position<pi)
    ea=K* $\omega$ *((per+2*pi/3-position)/(pi/6)+1);    eb=K* $\omega$ ;    ec=-K* $\omega$ ;
elseif(position>=pi&&position<4*pi/3)
    ea=-K* $\omega$ ;    eb=K* $\omega$ ;    ec=K* $\omega$ *((position-pi-per)/(pi/6)-1);
elseif(position>=4*pi/3&&position<5*pi/3)
    ea=-K* $\omega$ ;    eb=K* $\omega$ *((per+4*pi/3-position)/(pi/6)+1);    ec=K* $\omega$ ;
elseif(position>=5*pi/3&&position<=2*pi)
    ea=K* $\omega$ *((position-5*pi/3-per)/(pi/6)-1);    eb=-K* $\omega$ ;    ec=K* $\omega$ ;
end
sys = [ea,eb,ec];

```

换相逻辑函数 (pluse.m)

```

function [sys,x0,str,ts] = pluse(t,x,u,flag)
switch flag,
    case 0,
        [sys,x0,str,ts]=mdlInitializeSizes;
    case 3,
        sys=mdlOutputs(t,x,u);
    otherwise
        error(['Unhandled flag = ',num2str(flag)]);
end

function [sys,x0,str,ts]=mdlInitializeSizes
sizes = simsizes;
sizes.NumContStates = 0;
sizes.NumDiscStates = 0;
sizes.NumOutputs = 6;
sizes.NumInputs = 1;
sizes.DirFeedthrough = 1;
sizes.NumSampleTimes = 1;

```



```

sys = simsizes(sizes);
x0 = [];
str = [];
ts = [0 0];
function sys=mdlOutputs(t,x,u)
position=u(1);
if(position>=0&&position<pi/3)
    g1=1; g2=0; g3=0; g4=1; g5=0; g6=0;
elseif(position>=pi/3&&position<2*pi/3)
    g1=1; g2=0; g3=0; g4=0; g5=0; g6=1;
elseif(position>=2*pi/3&&position<pi)
    g1=0; g2=0; g3=1; g4=0; g5=0; g6=1;
elseif(position>=pi&&position<4*pi/3)
    g1=0; g2=1; g3=1; g4=0; g5=0; g6=0;
elseif(position>=4*pi/3&&position<5*pi/3)
    g1=0; g2=1; g3=0; g4=0; g5=1; g6=0;
elseif(position>=5*pi/3&&position<=2*pi)
    g1=0; g2=0; g3=0; g4=1; g5=1; g6=0;
end
sys = [g1,g2,g3,g4,g5,g6];

```

单神经元自适应 PID 控制算法实现 S 函数 (mhebb.m)

```

function [sys,x0,str,ts] = mhebb (t,x,u,flag,K,ni,np,nd)
switch flag,
case 0,
    [sys,x0,str,ts]=mdlInitializeSizes;
case 2,
    sys=mdlUpdate(t,x,u,ni,np,nd);
case 3,
    sys=mdlOutputs(t,x,u,K);
case {1,4,9},
    sys=[];
otherwise
    error(['Unhandled flag = ',num2str(flag)]);
end

```

```

function [sys,x0,str,ts]=mdlInitializeSizes

sizes = simsizes;
sizes.NumContStates  = 0;
sizes.NumDiscStates  = 3;
sizes.NumOutputs     = 1;
sizes.NumInputs      = 4;
sizes.DirFeedthrough = 1;
sizes.NumSampleTimes = 1;
sys = simsizes(sizes);
x0=[0.4,0.4,0.4]';
str = [];
ts  = [-1 0];

function sys=mdlUpdate(t,x,u,ni,np,nd)
sys(1)=x(1)+ni*u(1)*u(4)*u(1);
sys(2)=x(2)+np*u(1)*u(4)*(u(1)-u(2));
sys(3)=x(3)+nd*u(1)*u(4)*(u(1)-2*u(2)+u(3));
function sys=mdlOutputs(t,x,u,K)
xx=[u(1),u(1)-u(2),u(1)+u(3)-2*u(2)];
sys =[u(4)+K*xx/sum(abs(x));

```