

中南大学

---

硕士学位论文

---

智能小车的运动控制系统研究

---

姓名：许寰

---

申请学位级别：硕士

---

专业：控制科学与工程

---

指导教师：鲁五一

---

20090530

## 摘 要

由于工业现场中大多数恶劣环境和危险环境中仍采用的人工的操作方式，这促使了我们对智能车的研究和开发。本文是基于Freescale的S12系列MCU芯片开发的智能小车系统，具有路径识别和智能驾驶功能，简单实用、能适应对硬件要求相对较低的特点，是一项具有现实应用的可行性和广阔的应用前景的智能车技术。智能车技术是提高路径识别和路径跟踪的准确性、可靠性和实时性的核心，本文以Freescale S12系列MCU芯片开发的智能小车系统为平台，开展了旨在提高小车路径识别和路径跟踪的准确性、稳定性和实时性的研究。

本文首先介绍了智能小车系统的构建。这包括系统控制器等各个硬件模块的设计、路径识别方法的研究和各个模块的软件实现方法。其中路径识别采用硬件二值法和黑线中心提取算法实现，并通过试验验证效果良好。然后在路径识别的基础上，针对速度控制和舵机控制先是采用常规PID控制。对速度的控制能够实现快速准确的响应，但是舵机的控制效果不佳，具体表现在智能小车的稳定性和快速性耦合效果不好。因此，提出了基于模糊PID的舵机控制：以智能小车的车体纵轴线相对于路径中线的方向偏差和侧向位置偏差为输入，为转向轮的转角输出设计了一个模糊控制器进行路径跟踪，有效地克服了智能小车路径跟踪系统的非线性和不确定性的不良的影响，提高了系统的鲁棒性。通过对两种算法的试验数据比较，可以发现，采用常规PID控制速度，结合模糊PID控制舵机不管是在可靠性还是在快速性上都更为优异。最后在前述控制方法的基础上简单描述了记忆算法的思想和实现步骤。但该算法的效果还不够理想，这也将是以后努力的方向。

本文所研制的智能控制小车在总体方案和设计思想上有新意，仿真结果表明该方案是能够满足使小车的稳定性、快速性、实时性的设计要求。

**关键词：** 路径识别、常规PID、模糊PID、记忆算法

## ABSTRACT

Most of the industrial scene as a result of poor environmental conditions and dangerous environment is still used in the operation manual, which prompted our intelligent vehicle research and development. This article is based on Freescale's S12 series MCU chip smart car system developed with the path of recognition and smart driving function, simple and practical, able to adapt to a relatively low hardware requirements for the characteristics of a practical application of the feasibility and broad application the prospects for intelligent vehicle technologies. Intelligent vehicle technology is to improve the path of identifying and tracking the path of the accuracy, reliability and real-time core. Based on Freescale S12 series MCU chip development platform for intelligent vehicle system, launch vehicle designed to improve the path of identifying and tracking the exact path and stability studies and real-time.

This article introduces the Intelligent Vehicle System. This includes the system controller module and other hardware design, the path identification of research and various methods of software modules. Path to identify which hardware and binary black center line extraction algorithm, and test results. And then to identify the basis of the path for the speed control and steering control using conventional PID control first. Speed control to achieve rapid and accurate response, but the effect of poor steering control, a concrete manifestation in the smart car's stability and rapid effect of poor coupling. Therefore, based on the steering gear Fuzzy-PID control: smart car body longitudinal centerline line relative to the direction of the path deviation and lateral deviation of the location of input, as turning round the corner of the output of a fuzzy controller design for path tracking effectively overcome the path of intelligent vehicle tracking system and uncertainties of the nonlinear effects of the negative and enhance the system robustness. Two algorithms by comparing the experimental data, we can see that the speed of conventional PID control, PID control combined with fuzzy steering gear, whether in terms of reliability or speed are more outstanding. Finally, the

aforementioned control method based on the simple algorithm described in the memory of the ideas and implementation steps. However, the effectiveness of the algorithm is not ideal, it will also be the direction of future efforts.

In this paper, the development of the intelligent control of the car in the overall plan and design innovative thinking, simulation results show that the program is able to meet so that the stability of cars, fast, real-time nature of the design requirements.

**KEY WORDS:** Path identification, Conventional PID, Fuzzy-PID, Memory algorithm

## 原创性声明

本人声明，所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了论文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得中南大学或其他单位的学位或证书而使用过的材料。与我共同工作的同志对本研究所作的贡献均已在论文中作了明确的说明。

作者签名： 许霞 日期： 2009年5月30日

## 学位论文授权使用授权书

本人了解中南大学有关保留、使用学位论文的规定，即：学校有权保留学位论文并根据国家或湖南省有关部门规定送交学位论文，允许学位论文被查阅和借阅；学校可以公布学位论文的全部或部分内容，可以采用复印、缩印或其它手段保存学位论文。同时授权中国科学技术信息研究所将本学位论文收录到《中国学位论文全文数据库》，并通过网络向社会公众提供信息服务。

作者签名： 许霞 导师签名： 肖 日期： 2009年5月30日

# 第一章 绪 论

## 1.1 选题背景

由于工业现场中大多数恶劣环境和危险环境中仍采用的人工的操作方式,这促使了我们对智能车的研究和开发。智能车是一个集环境感知、规划决策等功能于一体的综合系统。它在各个领域都具有广泛的应用前景。例如,在工业生产中,可以代替人类完成恶劣环境下的货物搬运、设备检测等任务;在军事上,可以在危险地带代替人类完成侦察、排雷等任务;在民用上,可以作为导盲车为盲人提供帮助;在科学研究方面,可以代替人类完成外星球勘探或者一矿藏勘探等。智能车按照工作环境下要分为室外智能车和室内智能车两种,这两种环境有着较大的区别。室外环境一般较开阔,受光照、天气、时一间等的影响,环境变化很大,既包括简单的结构化环境也包括复杂的非结构化环境。室外自主移动机器人导航一般要求更高的智能。室内环境一般较狭窄,光线较稳定日环境复杂程度有限,一般视作结构化环境。

本论文结合以飞思卡尔半导体公司的HCS12单片机为核心的全国智能车邀请赛项目展开研究。该项赛事最初是在韩国举办。韩国大学生智能模型车竞赛是韩国汉阳大学汽车控制实验室在飞思卡尔半导体公司资助下举办的以HCS12单片机为核心的大学生课外科技竞赛。组委会将提供一个标准的汽车模型、直流电机和可充电式电池,参赛队伍要制作一个能够自主识别路线的智能车,在专门设计的跑道上自动识别道路行驶,跑完整个赛程用时最短,而且技术报告评分较高的参赛队就是获胜者。制作智能车,需要参赛队伍学习和应用嵌入式软件开发工具软件CodeWarrior和在线开发手段,自行设计和制作可以自动识别路径的方案、电机的驱动电路、模型车的车速传感电路、模型车转向伺服电机的驱动以及微控制器MC9S12DG128控制软件的编程,等等。其专业知识涉及控制、模式识别、传感技术、汽车电子、电气、计算机、机械等多个学科,对学生的知识融合和实践动手能力的培养,对高等学校控制及汽车电子学科学术水平的提高,具有良好的长期的推动作用。

本文是在比赛之后总结结果的基础上,通过更深的理论研究来找到更加合适的控制方法,完善智能小车的运动控制系统。

## 1.2 智能控制理论的发展及小车轨迹跟踪的智能控制

### 1.2.1 智能控制理论的产生和发展

智能控制是自动控制学科发展里程中的一个崭新的阶段。与其他学科一样,智能控制是由于科学技术发展的需要,从解决重大工程和技术问题的实践中产生和发展起来的。

从 20 世纪 60 年代起,由于空间技术、计算机技术及人工智能技术的发展,控制界学者在研究自组织、自学习控制的基础上,为了提高控制系统的自学习能力,开始注意将人工智能用于控制系统。1965 年,美国著名控制论专家 L. A. Zadeh 创立了模糊集合论,为解决复杂系统的控制问题提供了强有力的数学工具;同年,Feigenbaum 着手研制世界上第一个专家系统;傅京逊教授提出把人工智能中的知觉推理方法用于学习控制系统;1966 年, Mendel 进一步在空间飞行器的控制中应用了人工智能技术,并提出了“人工智能控制”的概念。1967 年, Leondes 和 Mendel 开始使用“潜能控制”一词,并把记忆、目标分解等一些简单的人工智能技术用于学习控制系统,提高了系统处理不确定性问题的能力。

从 20 世纪 70 年代初开始,傅京逊、Gloriso 和 Saridis 等人从控制论角度进一步总结了人工智能技术与自适应、自组织、自学习控制的关系,正式提出了智能控制就是人工智能技术与控制理论的交叉,并创立了人一机交互式分级递阶智能控制的系统结构。

在 20 世纪 70 年代中后期,以模糊集合论为基础,从模仿人的控制决策思想出发,智能控制在规则控制方面取得了重大的进展。1974 年, Mamdani 将模糊集和模糊语言逻辑用于控制,创立了用模糊语言描述控制规则的模糊控制器,并成功地用于工业过程控制。1979 年,他又成功地研制出自组织模糊控制器,使得模糊控制具有了较高的智能。

进入 20 世纪 80 年代以来,由于计算机技术的迅速发展以及专家系统技术的逐渐成熟,智能控制和决策的研究及应用领域逐渐扩大,取得了丰硕的成果。智能控制由研制、开发阶段转向应用阶段。80 年代中后期,神经网络的研究获得了重大进展,为智能控制的研究起到了重大的推进作用。1985 年 5 月, IEEE 在美国纽约召开的智能控制专题讨论会,标志着智能控制作为一个新的学科分支正式被控制界公认。从 1987 年开始,每年都举行一次智能控制国际研讨会,形成了智能控制的研究热潮。20 世纪 90 年代以来,智能控制的研究势头异常猛烈,1992 年 4 月,美国国家自然科学基金委员会和美国电力研究院联合发出《智能控制》研究项目倡议书;1993 年 5 月美国正控制系统学会智能控制专业委员会成立专家小组,专门探讨“智能控制”的含义。1994 年 6 月在美国奥兰多召开

了 IEEE 全球计算智能大会, 综合讨论了模糊控制、神经网络、进化计算三方面的内容, 1995 年 10 月国际智能自动化学会筹委会第一次会议在加拿大温哥华召开。国际智能自动化学会的成立将在世界范围内对智能化的研究起到推动作用<sup>[1][2]</sup>。

### 1.2.2 模糊控制的产生和发展

1965 年, 美国加州大学伯克利分校的 L. A. Zadeh 教授发表了著名的论文“fuzzy set”(模糊集)开创了模糊理论。1973 年, 他又给出了模糊逻辑控制的定义和相关的定理。1974 年, 英国的 Mandani 首先把模糊控制应用于工业控制中, 并取得了良好的效果, 这一开拓性的工作标志着模糊控制论的诞生。

模糊控制既具有系统化的理论, 又有着大量实际应用背景。模糊控制的发展最初在西方遇到了较大的阻力然而在东方尤其是在日本, 却得到了迅速而广泛的推广应用。40 多年来, 模糊控制不论从理论上还是技术上都有了长足的进步, 成为自动控制领域中一个非常活跃而又硕果累累的分支。其典型应用的例子涉及生产和生活的许多方面, 例如在家用电器设备中有模糊洗衣机、空调、微波炉、吸尘器、照相机和摄录机等在工业控制领域中有水净化处理、发酵过程、化学反应、水泥窑炉等的模糊控制在专用系统和其它方面有地铁靠站停车、汽车驾驶、电梯、自动扶梯、蒸汽引擎以及机器人的模糊控制等。模糊控制技术的成功应用, 确认了这一前沿技术的广阔应用前景。

模糊控制是基于模糊推理和模仿人的思维方法, 对难以建立精确数学模型的对象实施的一种控制, 其成功应用的根源在于模糊逻辑本身提供了由专家构造语言信息并将其转化为控制策略的一种系统的推理方法, 它是处理许多复杂而无法建立精确数学模型的系统控制问题的有效方法。模糊控制与常规控制方法相比有如下优点:

- (1) 控制系统设计依据经验和操作数据, 而不需要精确的数学模型。
- (2) 具有较强的鲁棒性, 适用于解决传统控制难以解决的非线性、时变及时滞系统。
- (3) 应用语言变量而不是数学变量, 易于构造形成专家系统推理过程采用“不精确推理”, 推理过程模仿人的思维过程, 能够处理复杂甚至“病态”系统。
- (4) 由离线计算得到控制查询表, 提高了控制系统的实时性。
- (5) 控制的机理符合人们对过程控制作用的直观描述和思维逻辑, 为智能控制应用打下了基础。

一般的模糊控制的结构如图 1-1 所示:



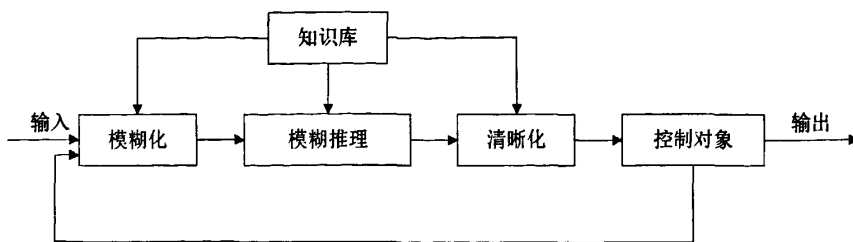


图 1-1 模糊结构控制图

然而，模糊控制由于其自身的因素，也有其不足之处，其主要缺陷有：

(1) 信息简单的模糊处理将导致系统的控制精度降低和动态品质变差，若要提高精度则必然增加量化级数，从而导致规则搜索范围增大，降低决策速度，甚至不能实现实时控制。

(2) 模糊控制的设计尚缺乏系统性，无法定义控制目标。控制规则的选择、论域的选择、模糊集的定义、量化因子的选取等多采用试凑法，这对于复杂系统的控制规则是难以奏效的，这些缺陷需要根据具体情况来与其它算法配合一起，使控制器的性能得到改进和完善。

### 1.2.3 小车轨迹跟踪智能控制

智能车辆又称为轮式移动机器人是一个集环境感知、规划决策、自动驾驶等多种功能于一体的综合系统。由于其实行实时控制，考虑到小车的舵机的转向和图像信息采集处理所需的时间，故在实际决策或发出指令时会出现时滞现象。要想系统很好的实现预期的控制效果，必需考虑到时滞的影响。传统时滞系统的经典控制方法包括常规 PID 控制、Smith 预估计控制和大林算法的控制算法。由于其具有结构简单、可靠性及实用性强等特点，在实际生产过程中得到了广泛的应用。但它们都是记忆参数模型的控制方法，因而自适应性和鲁棒性差、对模型的精确性要求高、抗干扰能力差。常规 PID 控制需要在对象模型精确已知的情况下，实现参数的在线整定，当被控对象特性发生变化时，就必须重新对系统进行模型辨识，自适应性很差。Smith 预估计控制是由 Smith 于 1957 年提出的，通过引入一个与被控制对象相并联的纯滞后环节，是补偿后的被控对象的等效传递函数不包括纯滞后项。但在实际操作中存在很大的问题：

(1) 系统对扰动的响应很差；

(2) 若被控对象中包含极点时，即使控制对象中含有积分器，系统对扰动稳态误差也不为零。

大林算法由美国 IBM 公司 Dalin 于 1968 年针对工业过程控制中的纯滞后特性而提出的一种控制算法，该算法的目标是设计一个合适的数字调节器，使整个系统的闭环传递函数相当于一个带有纯滞后的一个惯性环节，而且要求闭环系统

的纯滞后的时间等于被控制对象的纯滞后时间。但是由于其在设计中存在振铃现象,且与 Smith 算法一样,需要一个准确的过程数字模型,当模型的误差较大时,控制质量将大大的恶化,甚至系统会变得不稳定。

在传统的控制方法中,它们通常都依赖于被控对象的数学模型。但是如果需要机器实现高速运动,而且具有较好的控制品质,PID 反馈控制难以取得较好的控制效果。当机器工作环境及工作目标的性质和特征在工作过程中随时间发生变化时,控制系统的特性有未知和不确定的特性。这些未知因素和不确定性使控制系统的性能降低。因此,采用传统的控制方案已经不能满足控制要求。

随着人们对智能机器高速度、高精度要求的不断提高,并且要求智能机器在完成复杂的任务时具有自行规划和决策的能力,有自动躲避障碍运动到期望目标位置的能力等,使得整个智能机器系统对其控制部分的要求也越来越高,开发具有智能的机器已经成为人们研究的热点<sup>[3,4]</sup>。

智能机器技术能得到广泛地应用,得益于计算机技术和控制技术的发展和完善。常规的 PID 控制方法、模糊控制、神经网络控制、专家控制、自适应控制、鲁棒控制等现代控制理论方法都在智能机器系统中得到了应用。

### 1.3 本文所开展的研究工作和内容安排

本文结合 FREESCALE 芯片制作的智能小车,由于对外在环境进行实时检测的,需要小车能根据当前的环境状况完成既定的任务,这就需要小车具有良好的智能性。在众多智能控制方法中以常规 PID 控制、模糊控制应用最为广泛。

模糊控制是由 Zandeh 于 1965 年首先创立的,其核心是模拟人的思维方式对一些无法推导精确数学模型的被控对象设计模糊控制器,通过建立输入输出模糊集及模糊规则来实现有效的控制。该算法的优点是可以不需要被控对象的数学模型即可实现较好的控制,因而特别适合于那些具有纯滞后大惯性参数漂移大的非线性不确定复杂系统,但存在控制精度不高的缺点,为了克服这个缺点,提出了一种 Fuzzy-PID 控制方法,在大偏差误差内采用模糊控制取得较好的动态响应并抑制随机干扰,在小偏差范围内采用 PID 控制来抑制稳态误差。这样既可以充分发挥模糊控制器动态性能好、鲁棒性强的特点,取得对纯滞后系统较好的控制效果,又可以避免繁琐的参数整定过程,弥补了各自的不足,因而在实际中应用的最为广泛。

本文中智能小车的运动控制系统的研究采用两种方法:常规 PID 控制算法和模糊控制算法,并对两种算法的试验效果进行比较,得出更科学合理的小车运动控制算法——模糊 PD 控制算法。最后在此基础上还介绍了专门基于智能小车大赛的一种控制算法——记忆算法,但还未能成熟使用与本系统中,有待以后努力。

本论文的具体研究内容和具体章节安排如下：

第一章：绪论。本章介绍了选题背景，简要地回顾了智能控制理论的发展过程，并阐述了小车轨迹跟踪上的智能控制。

第二章：智能小车系统。本章首先介绍了小车系统的硬件电路设计方案，接着详细的介绍了主要模块的功能及其软件实现的方法等。

第三章：基于常规 PID 的智能小车系统控制。本章详细介绍了常规 PID 的基本原理及其设计步骤，尤其是数学模型的建立和模型参数的整定等。最后给出基于常规 PID 速度控制模型和舵机控制模型。

第四章：基于模糊 PID 的智能小车系统控制。本章详细介绍了模糊控制的基本原理及模糊控制器的设计步骤。然后提出了基于模糊理论的舵机控制，并通过试验验证效果更佳。最后研究了基于模糊 PID 控制的记忆算法。

第五章：总结和展望。本章总结了本文的主要研究内容和成果。并结合当前的智能自动导向车的技术，提出了小车需要改进的方向和方法。

## 第二章 智能小车系统设计

### 2.1 系统简介

智能车系统的功能模块主要包括控制核心模块（MCU）、电源管理模块、路径识别模块、后轮电机驱动模块、转向舵机控制模块、速度检测模块以及辅助调试模块。每个模块都包括硬件和软件两部分。硬件为系统工作提供硬件实体，软件为系统提供各种算法。系统功能模块结构图如图 2-1 所示：

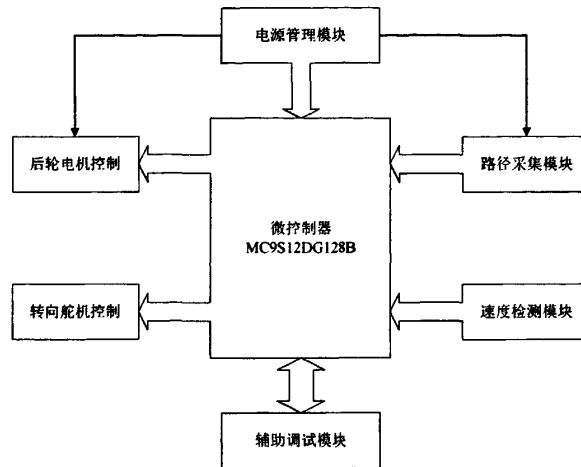


图 2-1 智能小车系统结构图

在这些模块中，有些模块是小车完成比赛所必须的，如控制核心模块、电源管理模块、路径识别模块、后轮电机驱动模块、转向舵机控制模块、速度检测模块。这些模块是小车系统的核心模块，也是小车取得好成绩的关键所在。而有些模块则对小车的行驶效果没有太大影响，它们只是为了增强系统功能而设计的模块，如小车故障诊断模块，调试辅助模块，这些模块统称为小车系统的辅助模块。小路上的每一个模块，无论是核心模块还是辅助模块，其制作过程中都包括以下几个步骤查阅资料，科学的论证，电路制作，软件编程，最后经过实践检验才完成的。

从结构图我们看到，智能小车系统其实是两路输入两路输出系统。路径采集模块是智能小车的导航系统，速度检测模块作为速度电机的反馈控制。输出两路 PWM 波驱动后轮电机和舵机实现小车的智能行驶。因此智能小车的关键就是结合导航系统获取前方道路信息，再结合运动控制算法实现对速度和舵机的控制。若想了解更多有关上述智能小车系统建立和分析的内容可翻阅文献[5-11]。

## 2.2 系统硬件平台设计

本节将详细介绍智能小车系统硬件平台的设计。参照前面的系统结构图，我们可将系统硬件平台分为 MCU 模块、电源管理模块、路进识别模块、速度模块、舵机模块。其中速度模块包括直流电机模块和速度检测模块。

### 2.2.1 MCU 模块

本次智能汽车邀请赛使用的微控制器是 freescale 公司推出的 S12 系列单片机中具有增强型的 16 位单片机 MC9S12DG128B，该系列单片机在汽车电子领域有着广泛的应用。S12 系列单片机的中央处理器 CPU12 由以下三部分组成：算术逻辑单元 ALU，控制单元和寄存器组。CPU 外部总线频率为 8MHZ 或者 16MHZ，通过内部锁相环(PLL)，可以使内部总线速度可以达到 25MHZ。寻址方式有 16 种。内部寄存器组中的寄存器，堆栈指针和变址寄存器均为 16 位。它具有很强的高级语言支持功能。CPU12 的累加器 A 和 B 是 8 位的，也可以组成 16 位累加器 D。CPI 的寄存器组包括 5 个部分<sup>[12-14]</sup>：

(1) 8 位累加器 A, B 或 16 位的累加器 D。

(2) 16 位寻址寄存器 X 和 Y 是用来处理操作数的地址。可分别用于源地址，目的地址的指针型变量运算。

(3) 堆栈指针 SP 是 16 位寄存器。

(4) 程序计数器 PC 是 16 位寄存器。它表示下一条指令或下一个操作数地址。

(5) 条件码寄存器 CCR。

#### 1、单片机 MC9S12DG128B 的内部资源

MC9S12DG128B 作为 S12 系列的一种，内部资源非常丰富，简要介绍如下：

(1) 时钟和复位模块：PLL（锁相环频率合成器），COP（看门狗），时钟监控。

(2) 存储器：128KB FLASH EEPROM, 2KB EEPROM, 8KB RAM。

(3) 两个 8 路 A/D 转换器：10 位精度，具有外部触发转换功能。

(4) 增强型捕捉定时器：16 位主计数器，7 位分频系数，8 个输入捕捉通道或输出比较通道，两个 8 位或一个 16 位脉冲计数器。

(5) 8 路 PWM 通道：可编程周期以及占空比，8 位 8 路或 16 位 4 路 PWM，独立控制各路 PWM 的周期和占空比，中间对齐和左对齐输出，频率范围宽的可编程时钟选择逻辑，紧急时刻快速关闭输出。

(6) 串行接口：两个异步串行通信接口模块 SCI，1 个 IIC 总线接口，2 个同步串行外设接口 SPI。

(7) 3 个 1M/S, CAN2. 0A, CAN2. 0B 兼容模块。

(8) SAE J1850 网络通讯口。

MC9S12DG128B 结构框图如图 2-2 所示:

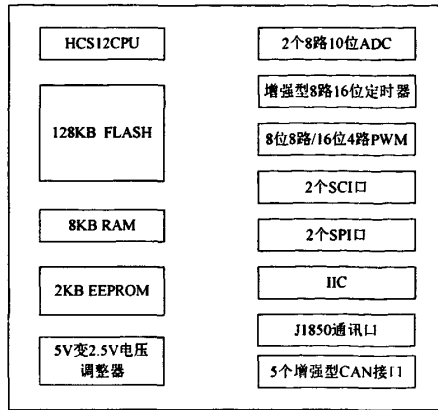


图 2-2 MC9S12DG128B 结构框图

MC9S12DG128B 支持在线编程，即在线写入，擦除，在线下载程序。在线编程的基本原理是，单片机片内的 CPU 有能力对片内 FLASH 进行写入，擦除操作，用户需以某种方式，一般为串行地将命令和数据传送给单片机就可以了，单片机的编程接口除完成 FLASH 写入，擦除功能外，还可用于应用程序的调试，甚至可以在应用程序中运行时，动态地获取 CPU 寄存器的值，存储器等的瞬态信息，这就是所谓的 BDM(Background Debug Mode) 调试方式。除此之外，可以先通过 BDM 把监控程序写入 Flash，然后可以脱离 BDM，通过 RS-232 下载程序，调试等<sup>[15]</sup>。

## 2、单片机 MC9S12DG128B 的最小硬件系统

虽然单片机将 CPU、ROM、RAM 以及 I/O 系统集成在一个集成电路芯片上，但仍需要一些外围电路的支持。这些外围电路主要为单片机系统提供电源、时钟、I/O 口驱动、通信口等，让单片机动起来。这个最小系统，即要能够实现人机通信，一般是通过前面介绍的 BDM 或者串口实现。需要能够接收 PC 机发送的命令，下载程序和调试程序以及在线监控等。有了这些功能就可以调试单片机的硬件系统了，在硬件基本上没有问题的情况下才能进行软件开发。最小系统板也称为开发板，主要包括以下几个方面：

### (1) 电源电路。

开发板的工作电压是靠外部提供+5V 直流电源。其实，单片机的 I/O 模块的供电多数采用+5V 供电，单片机 CPU 的速度越来越快，所需供电电压也越来越低，一般采用 3.3V、2.5V 甚至更低的工作电压。S12 单片机片内使用 2.5V 电压，片外 I/O 使用 5V 电压，较低的片内电压的 CPU 运算速度快，功耗低；较高的 I/O 电平有利于抗外界干扰，所有 S12 单片机特别适合工作在恶劣环境下的控制系

统。由于 S12 单片机内部集成了电压调整器模块，电压调整器产生片内所需的电压，因此只要向单片机提供+5V 的外部电源就可以了。为了稳定这些不同的电压需要外接一些电容。这些电容有两类：电容值比较大的如 1uf、10uf 等称为储能电容，储能电容消除三极管电路导通与截止时的电流变化；另一类较小值的电容如 0.1uf、0.01uf 称为去耦电容，去掉单片机运行时产生的高频噪声。

### (2) 时钟电路。

主要由一些石英晶体振荡器和一些电容电阻组成。虽然单片机可以由内部集成的 RC 振荡器产生工作所需的时钟，但这种简单的时钟电路频率的稳定性得不到保证，不如外部晶振可靠。速度很快的单片机往往使用内部集成的压控振荡器（VCO）产生的高频振荡作为系统时钟，但 VCO 也需要外部的晶振提供稳定的频率来锁定起振频率。其外部晶振有两种接线方式：一种是串联振荡电路，另一种是并联振荡电路。在智能小车控制的主板设计中，采用的是并联振荡电路。具体的电路图如图 2-3 所示：

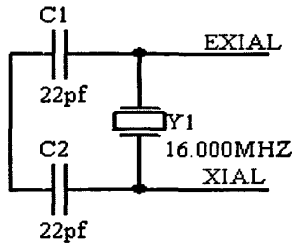


图 2-3 并联振荡电路图

(3) 复位电路。虽然单片机片内集成有上电复位电路，单片机上电时可自动产生复位信号，但加上一个手动复位按钮会给调试带来方便。外部复位电路可以使用简单的按钮加阻容电路，也可以加专门的复位芯片。在该控制系统主板设计中，采取按钮加阻容电路。具体的电路如图 2-4 所示：

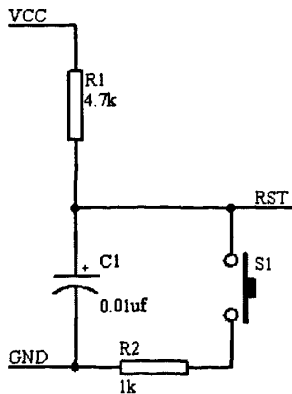


图 2-4 复位电路图

这里需要说明的是，在该电路中充电电容 C1 的容值不能太大，在本次设计中选用 0.01uf，当然你也可以直接用 S1 组成简单的复位电路。在这里画这个复

位电路图，主要是为了说明一点，在 51 单片机中你可以选用 10uf 的电容，但在 freescale S12 系列单片机的复位电路却不能选择这么大，主要是因为容值太大会导致单片机无法进入 BDM 工作模式。

(4) RS-232 驱动电路。

RS-232 驱动电路提供 TTL 到 RS-232 的电平转换。单片机可以利用异步串行通信协议与 PC 通信。RS-232 电平转换芯片可以实现 TTL 电平与 RS-232 电平之间的转换，然后再通过 9 芯串行接口与 PC 进行串行通信。在智能小车运动控制系统中，上位机可以通过 RS-232 向单片机控制板下载程序。RS-232 的接口电路如图 2-5 所示，图中的 MAX232 是电平转换芯片，完成 TTL 电平到 RS-232 电平的转换功能。

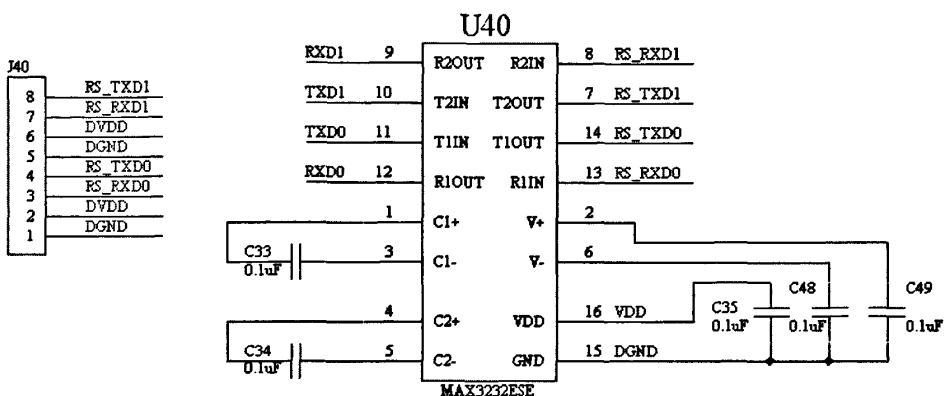


图 2-5 RS-232 接口电路图

(5) BDM 调试接口。

当单片机进入 BDM 模式后，上位机可以通过 BDM 接口向单片机下载程序，进行在线调试。BDM 方式也是单片机最根本的调试方式，因为上位机要想通过 RS-232 向单片机下载程序，必须首先要通过 BDM 方式向单片机的中下载监控程序。接口通信示意图如图 2-6 所示：

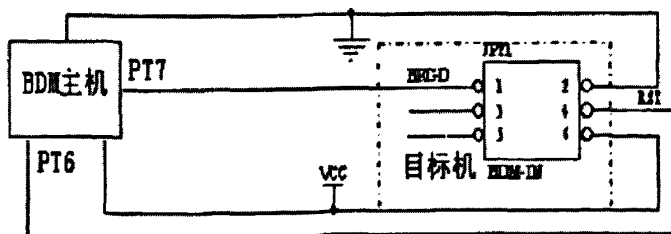


图 2-6 BDM 接口通信示意图

从上面分析可以看出，通过上位机向单片机写程序有两种方式，一种是 BDM 方式，另一种就是 RS-232 方式。如图 2-7 所示：



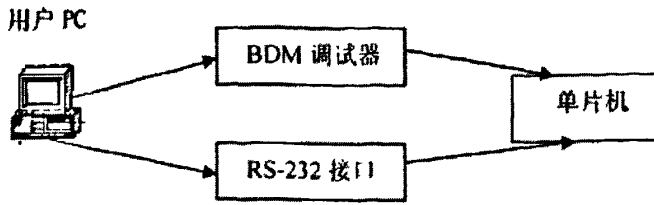


图 2-7 上位机和单片机的通讯方式示意图

(6) 调试显示。

在单片机的某个 I/O 接口上装上一些二极管对最小系统的调试很有用，将调用的发光二极管接在 PORTK 口上，虽然这部分并不是最小系统必需的。另外，把单片机的所有 I/O 口都引出来，这样方便单片机与外界电路的联接。

(7) 单片机模式选择模块<sup>[16]</sup>。

S12 系列单片机一共有两种工作模式，分别是单片运行模式和扩展运行模式。单片运行模式是单片机最常用的一种运行模式，单片运行模式又分为普通单片模式和特殊单片模式。普通单片模式是正常运行应用程序时应使用的模式，特殊单片模式是指 BDM 调试模式。扩展运行模式允许通过 CPU 外部总线扩展 RAM、FLASH、I/O 等。工作模式的确立由 MODA、MODB、MODC 的输入电平状态来确定的，在复位的时候，复位信号上升沿锁存 MODA、MODB、MODC 等引脚上的输入电平状态到运行模式寄存器 (Mode Register) 中的相应位，随后单片机便进入相应的工作模式，其对应的模式选择如表 2-1 所示，因为 MC9S12DG128B 有足够的存储器资源，所以在智能车控制系统中只需要工作在单片运行模式。从表 2-1 可知，要使单片机工作在单片运行模式，MODA 和都跳线到 MODB 低电平。然后当 MODC(BKGD)为低电平时，就进入特殊单片模式，当为高电平的时候就进入一般单片模式。通常，当单片机输入引脚浮空时，CPU 默认高电平，BKGD 引脚上的低电平是由 BDM 调试工具的相应引脚提供的。所以可理解为，当插上 BDM 头时，可进入特殊单片模式；当不插头时，自动进入普通单片模式。

表 2-1 工作模式选择

PE5=MODA	PE6=MODB	BKGD=MODC	工作模式描述
0	0	0	特殊单片模式，允许 BDM
0	0	1	普通单片模式
0	1	0	特殊测试模式，允许 BDM
0	1	1	特殊外围，BDM 不能使用
1	0	0	窄竞争扩展模式，允许 BDM
1	0	1	一般窄扩展模式，允许 BDM
1	1	0	宽竞争扩展模式，允许 BDM
1	1	1	一般宽扩展模式，允许 BDM

### 2.2.2 电源管理模块

电源作为小车动力来源，为小车上的控制器，执行器，传感器提供可靠的工作电压，现在市面上常见的充电电池包括镍镉电池、镍氢电池、锂离子电池、碱性电池和封闭式铅酸电池等几类。由于镍镉电池具有价格便宜、技术成熟、电路简单、瞬间大电流供应能力强等优势，它占据了大部分的消费性电子产品市场，因此在本次的大赛中，国产镍镉可充电电池被选为动力车的电源。

单个镍镉电池只能提供 1.2V 的供电电压，这次比赛所使用的电池上 6 节相同型号的电池串联起来从而得到 7.2V 的电池组，其标称容量为 2000mAh，也就是说，该电池可以在 2A 的供电电流下持续供电 1 小时。

电源管理模块的功能对电池进行电压调节，为各个模块正常工作提供可靠的工作电压。在智能车控制系统中，主控制模块、车速传感器模块以及电压检测模块需要 5V 电压，路径识别模块需要 12V 电压，舵机有 4.8V 和 6V 两种工作电压，为了提高舵机的灵敏度，舵机选用 6V 供电，直流电机可以使用 7.2V 蓄电池直接供电。

智能小车电源分配图如图 2-8 所示：

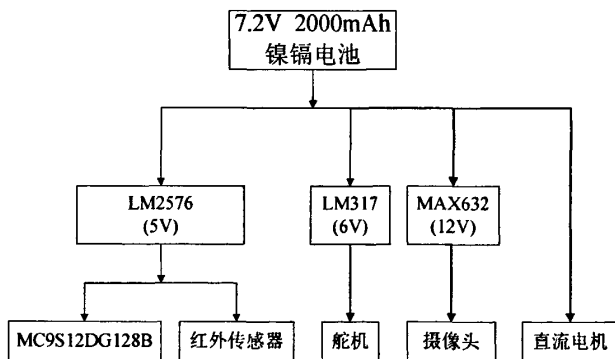


图 2-8 电源模块结构图

### 2.2.3 路径识别模块

寻线传感器的选择，是此次智能车竞赛需要首先考虑的问题。路径识别模块是智能车控制系统的关键模块之一，它将路况的信息传输给主控制模块，路径识别方案的好坏，直接影响着小车的控制效果。在智能车控制系统中，小车有多种寻迹方案。包括光电传感器寻迹方案，单独采用摄像头寻迹方案以及摄像头寻迹与光电传感器寻迹结合在一起的寻迹方案。采用不同的寻线传感器，意味着完全不同的比赛策略与车辆控制算法，在电路设计、机械结构安装以及软件程序上体现完全不同的取向。根据韩国之前的比赛经验，考虑到车模比较小，单片机处理速度不高，多数参赛队伍选用红外光电传感器阵列作为主要的寻线传感器。红外

光电传感器是利用红外光在黑色和白色的反射率相差大的特性，识别黑线。其电路设计简单，设备成本低，判断黑线快速准确，只需一路普通I/O，即可捕获黑线。采用单排或者双排的光电传感阵列，获取前方赛道信息，控制车体运动，完成比赛。光电传感器稳定可靠，但是检测距离近，限制了车辆控制算法的预判性。而摄像头寻迹，就是通过摄像头把智能车前面的路径信息传输到控制系统，来进行路径识别的一种寻迹方法。摄像头有面阵和线阵两种。它的优点是可以更远更早地感知赛道的变化，但是硬件电路比较复杂，信息处理量大，如何对摄像头一记录的图像进行分割和识别，加快处理速度是摄像头方案的难点之一<sup>[17]</sup>。

### 1、摄像头的工作原理

摄像头分黑白和彩色两种，为达到寻线目的，只需提取画面的灰度信息，而不必提取其色彩信息，所以本设计中采用的是黑白摄像头。较使用同等分辨率的彩色摄像头而言，这样可减少单片机采样的负担。

摄像头主要由镜头、图像传感芯片和外围电路构成。图像传感芯片是其最重要的部分，但该芯片要配以合适的外围电路才能工作。将芯片和外围电路制作在一块电路板上，称为“单板”。若给单板配上镜头、外壳、引线和接头，就构成了通常所见的摄像头，如聊天用的摄像头；若只给单板配上镜头，这就是“单板摄像头”。单板摄像头日常生活中不多见，生产单板的公司通常将它们卖给其它公司，其它公司再按自己的要求包装这些单板。单板通常有三个端子：电源端、地端和视频信号端（有的还多出一个端子，那是音频信号端）。电源接的电压要视具体的单板而定，目前一般有两种规格，6-9V或9-12V。视频信号的电压一般位于0.5V-2V之间。

摄像头的工作原理是：按一定的分辨率，以隔行扫描的方式采集图像上的点，当扫描到某点时，就通过图像传感芯片将该点处图像的灰度转换成与灰度一一对应的电压值，然后将此电压值通过视频信号端输出。如图2-9所示<sup>[18]</sup>：

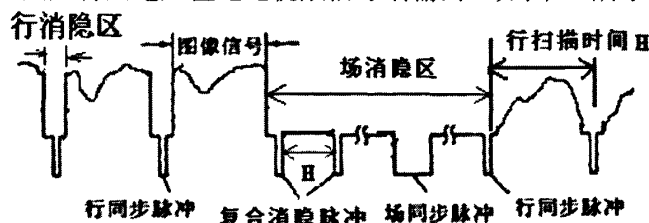


图2-9 摄像头视频信号

摄像头连续地扫描图像上的一行，则输出就是一段连续的电压信号，该电压信号的高低起伏反映了该行图像的灰度变化。当扫描完一行，视频信号端就输出一个低于最低视频信号电压的电平（如0.3V），并保持一段时间。这样相当于，紧接着每行图像信号之后会有一个电压“凹槽”，此“凹槽”叫做行同步脉冲，它是扫描换行的标志。然后，跳过一行后（因为摄像头是隔行扫描的），开始扫

描新的一行，如此下去，直到扫描完该场的视频信号，接着又会出现一段场消隐区。该区中有若干个复合消隐脉冲，其中有个远宽于（即持续时间长于）其它的消隐脉冲，称为场同步脉冲，它是扫描换场的标志。场同步脉冲标志着新的一场的到来，不过，场消隐区恰好跨在上一场的结尾和下一场的开始部分，得等场消隐区过去，下一场的视频信号才真正到来。摄像头每31秒扫描25幅图像，每幅又分奇、偶两场，先奇场后偶场，故每秒扫描50场图像。奇场时只扫描图像中的奇数行，偶场时则只扫描偶数行。

摄像头有两个重要的指标：有效像素和分辨率。分辨率实际上就是每场行同步脉冲数，这是因为行同步脉冲数越多，则对每场图像扫描的行数也越多。事实上，分辨率反映的是摄像头的纵向分辨能力。有效像素常写成两数相乘的形式，如“320x240”，其中前一个数值表示单行视频信号的精细程度，即行分辨能力；后一个数值为分辨率，因而有效像素=行分辨能力×分辨率。值得注意的是，通常产品说明上标注的分辨率不是等于实际分辨率（即每场行同步脉冲数），而是等于每场行同步脉冲数加上消隐脉冲数之和。因此，产品说明上标注的“分辨率”略大于实际分辨率。我们要知道实际的分辨率，就得实际测量一下。通过S12单片机的定时器模块对单个脉冲的下降沿和上升沿间隔、两相邻脉冲上升沿间隔进行计时，可得每行信号和每个脉冲持续的时间。实际测得所用摄像头摄像头扫描的每场中有320行信号，其中第23行到310行是视频信号，第311行到下一场的第22行是场消隐信号。在视频信号区，每行信号持续的时间相同，约为62us；每行的行同步脉冲持续时间也相同，约为4.7us。而在场消隐区，每行持续的时间会有所变化，每行对应的消隐脉冲持续的时间，尽管其中大多数为3.5us，但也有变化。在场消隐区中，第320行的消隐脉冲持续的时间远长于其他消隐脉冲的时间，此脉冲即为场同步脉冲。

## 2、摄像头的选择

因为S12单片机的AD转换时间在不超频的情况下最短为7us。如果选用一个分辨率为320线的摄像头，则单行视频信号持续时间约为 $20\text{ms}/320=62.5\text{us}$ ，AD对单行视频信号采样的点数将不超过 $[62.5/7]+1=9$ 个。若使用分辨率为640线的摄像头，则单行视频信号持续的时间约为 $20\text{ms}/640=31\text{us}$ ，AD对单行视频信号采样的点数将不超过 $[31/7]+1=5$ 个。这就是说，分辨率越高，单行视频信号持续的时间就越短，AD对单行视频信号所能采样的点数就越少。

如前所述，摄像头的分辨率越高，尽管可提高纵向分辨能力，却会减少单片机AD采样单行信号的点数，削弱了横向分辨能力。现在市场上摄像头的分辨率通常都在300线以上，由此推得单行视频信号的持续时间至多为 $20\text{ms}/300=66\text{us}$ 左右，AD采样每行视频信号的点数至多为 $[66/7]+1=10$ 个（不超频情况下），这对

我们的智能车定位系统来说是不够的。所以，在选择摄像头时，我们应当尽量选择分辨率低的摄像头，这样做会降低摄像头的纵向分辨能力（降低后对我们来说仍然远远够用），却可以增加单片机采样单行视频信号的点数，提高横向分辨能力。

在本次设计过程中，我们在市场上所能找到的分辨率最低的摄像头是采用1/3 Omni Vision CMOS 为传感芯片的摄像头，其分辨率为320线。另外，为了使得视野大小合适，选用了3.6mm的镜头。

### 3、电路设计

要能有效地对视频信号进行采样，首先要处理好的问题是如何提取出摄像头信号中的行同步脉冲、消隐脉冲和场同步脉冲。这里有两种可行的方法。第一，直接通过单片机AD进行提取。因为行同步脉冲、消隐脉冲或场同步脉冲信号的电平低于这些脉冲以外摄像头信号的电平，所以据此可设定一个信号电平阈值来判断AD 采样到的信号是否为上述三类脉冲。第二，就是给单片机配以合适的外围芯片，此芯片要能够提取出摄像头信号的行同步脉冲、消隐脉冲和场同步脉冲以供单片机作控制之用。考虑到单片机的速度有限，而一些脉冲的间隔时间又较短，同时为了减轻其处理负担，我们采用了第二种方法进行信号提取。LM1881视频同步信号分离芯片（简称LM1881）可从摄像头信号中提取信号的时序信息，如行同步脉冲、场同步脉冲和奇、偶场信息等，并将它们转换成TTL 电平直接输给单片机的I/O口作控制信号之用<sup>[19]</sup>。LM1881的端口接线方式如图2-10所示。

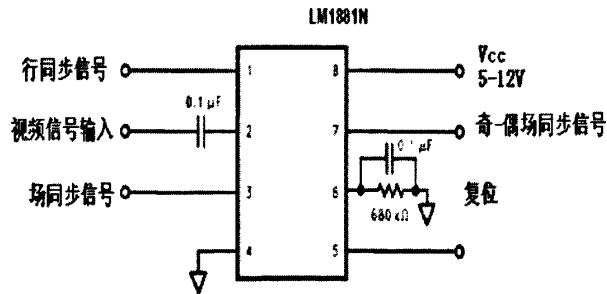


图2-10 LM1881端口连接图

其中，引脚2为视频信号输入端，引脚1为行同步信号输出端（如图2-10中的b）。引脚3为场同步信号输出端，当摄像头信号的场同步脉冲到来时，该端将变为低电平，一般维持230us，然后重新变回高电平（如图2-10中的c）。引脚7为奇-偶场同步信号输出端，当摄像头信号处于奇场时，该端为高电平，当处于偶场时，为低电平。事实上，我们不仅可以用场同步信号作为换场的标志，也可以用奇-偶场间的交替作为换场的标志<sup>[20]</sup>。

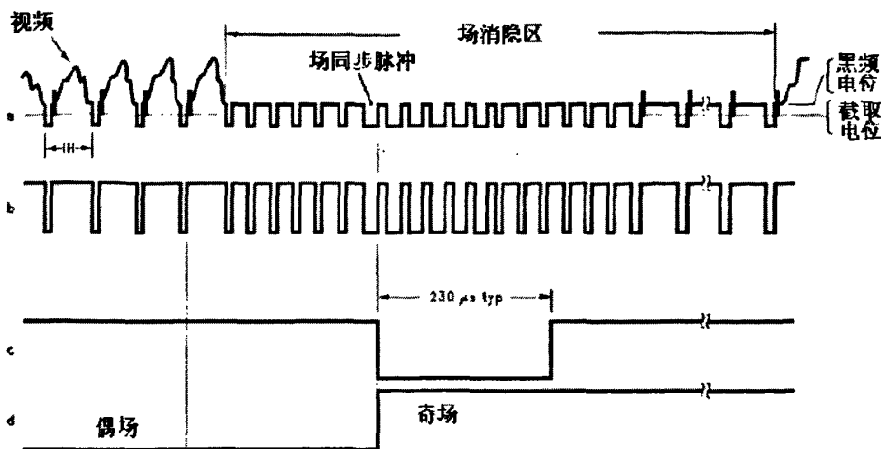


图2-11 LM1881信号时序图

其中，a为视频信号，b为视频行同步信号，c为场同步信号，d为奇偶场同步信号。通过LM1881可以将视频信号分离为场同步信号，行同步信号，奇偶场同步信号，从而方便单片机采集每场的有效视频信号。单片机采集信号的方法一般是直接通过S12集成的A/D模块直接采集。这种方法可以获得更为准确的赛道的数据，针对黑线，黑线旁边以及远离黑线都能非常清晰的分辨出来。但由于单片机A/D转换速率无法满足摄像头视频信号的转换，这样就会导致信息流失，从而无法正确获取赛道信息。因此，本设计种采用另外一种方法——二值法。二值法就是通过试验设定一个合适的电压阈值，视频信号然后通过电压比较器，高于阈值就是1，低于阈值电压就为0。然后单片机直接通过IO口读取，能够完全读取整个行有效信号。具体的摄像头采样电路如图2-12所示。

摄像头视频信号端接LM1881的视频信号输入端，LM1881的行同步信号端（引脚1）通过跳线选择，分别接入ECT模块中的PT0和外部中断引脚（IRQ），通过跳线选择，既可以采用脉冲捕捉方式获取行同步信号，也可以采用硬件中断获取。同时将LM1881的场同步信号和奇-偶场同步信号也输入到ECT模块中（选用PT1，PT2），这样，既可以采用查询方式获取奇偶场信号跳变，又可以采用脉冲捕捉方式获取电平变化。通过这样的接线，为软件开发提供了多种选择的机会，使程序更加灵活。同时，比较器的阈值电压也可以根据场地的变化而调节出合适的电压值。

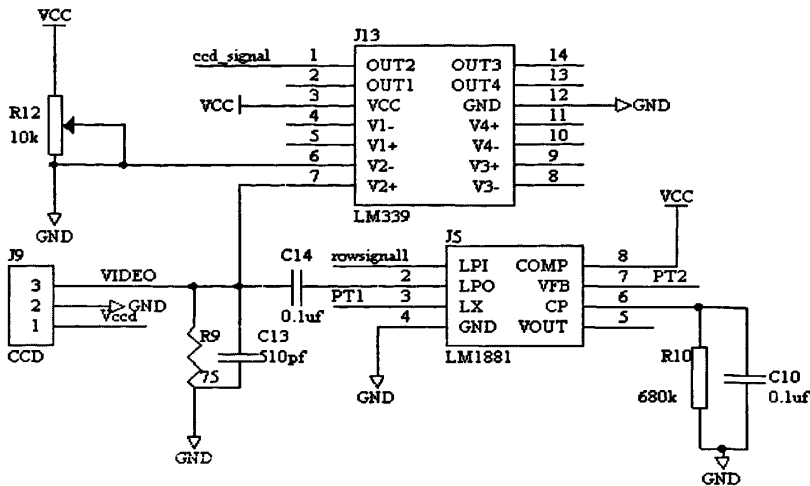


图2-12 摄像头采样电路图

### 2.2.4 速度模块

智能小车前进的动力是通过直流电机来驱动的，直流电动机是最早出现的电动机，也是最早能实现调速的电动机。采用专用集成电路芯片可以很方便地组成单片机控制的小功率直流伺服系统。本文选用的驱动芯片是飞思卡尔半导体公司的H桥式驱动器MC33886<sup>[21]</sup>。其管脚示意图如图2-13所示，它的应用电路也很简单，如图2-14所示。

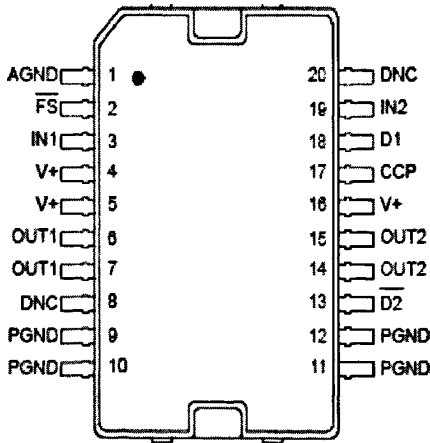


图2-13 MC33886管脚示意图

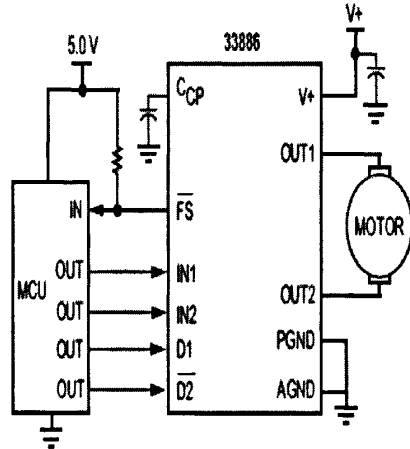


图2-14 MC33886应用电路图

在闭环控制系统中，速度指令值通过微控制器变换到驱动器，驱动器再为电机提供能量。速度传感器再把测量的的小车的速度量的实际值回馈给微控制器。以便微控制器进行控制。因此要对控制系统实行闭环控制，必须要有感应速度量的速度传感器。本文采用反射式红外传感器加上自制的码盘实现对小车速度的实时检测。如图2-15所示：

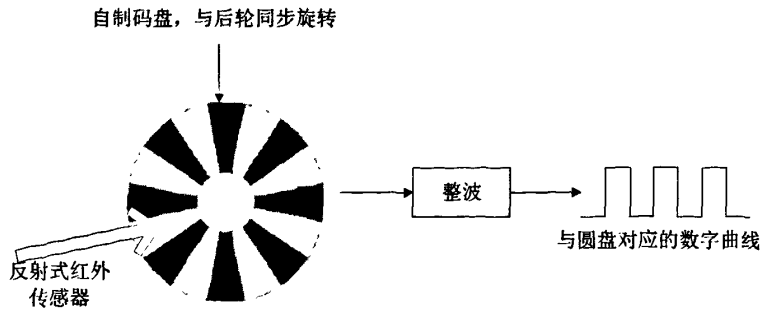


图 2-15 测速模块功能示意图

### 2.2.5 舵机模块

舵机采用的是 HS-925 型号舵机。该舵机的工作角度为 45 度/200 $\mu$ s，有两种工作电压，分别是 4.8V 和 6.0V。工作电压为 4.8V 时，速度为 0.11sec/60，堵转力矩为 6.1kg·cm；工作电压为 6.0V 时，速度为 0.08sec/60，堵转力矩为 7.7kg·cm 在设计中，为了提高舵机的响应速度和工作力矩，采用 6.0V 的工作电压。该舵机有三个接口：电源、地、驱动信号。驱动信号直接接 MCU 的 PWM 输出端，硬件电路非常简单，在此就不累述了。

## 2.3 系统的软件设计

智能小车系统的软件主要是输入信号的处理和输出驱动信号的实现。输入信号的处理主要就是对视频信号的处理，而输出驱动信号即舵机和速度的驱动信号。下面就将分这两部分来讲述。

### 2.3.1 视频信号处理

基于摄像头获得跑道图像，同时通过黑线中心检测出黑线，来预判前方的路况信息，从而控制小车的舵机转向和行驶速度。因此，视频信号处理就将围绕着路径黑线中心的提取和对弯道的预判来完成。

#### 1、黑线中心提取算法

在通过摄像头模块硬件处理后，我们能够获得跑道二值信号。在这基础上，接下来采用黑线中心提取算法，来获得跑到的黑线位置。该算法主要的思想是：针对每行获得的信号，找出每行的黑线起点和终点，取两个值得中心作为该行黑线的中心。在一场扫描完之后，再对整个场中的黑线中心进行软件滤波，获得更为准确的信息。具体过程如下：

①首先找到小车行扫描的中心点，也就是以小车为原点整个跑道的坐标。通



过实验，采集 220 点的时候，110 点正好处于以摄像头为垂直轴的原点。这样我们可以将每行的第 110 点作为横坐标上的原点，左边的作为横坐标的负轴，右边的作为横坐标的正轴；

②从第一个点开始依次向右判断，寻找黑线的两个边缘点。其中起始点要求本身值为 0，而且它上一个值为 1，此时将它作为黑线的起始点；结束点则同样要求本身值为 0，而且它的下一个值为 1。同时，为了防止有干扰，要求起始点要小于结束点，且差值大于 2，这就能保证至少有三个点表示黑线；

③在上面的基础上，可以顺利求出黑线中心的横坐标位置。同样，当扫描完一场时，也就获得该场中每行黑线中心的横坐标位置；

④最后要对每场的黑线中心坐标进行软件调整。首先，每行的黑线中心横坐标要满足位于它前后两个黑线中心坐标之间，若不满足，取它前后黑线中心坐标的平均值作为它的值。但这需要注意第一个和最后一个的值不需要调整；

通过上述四步之后，就获得了一场信息中扫描的各行黑线中心坐标值，而且在碰到一些小弯道时，如 S 弯道，可以通过第四步修改成直道，使得小车在过 S 道的时候可以快速通过，免做无用功。实验证明，黑线中心算法可以很好的获得所需的赛道信息<sup>[22]</sup>。

## 2、弯道识别

影响小车速度的一个非常重要的因素就是对弯道和直道的提前判断识别。对弯道的正确提前预判，可以让小车提前降速从而实现安全过弯道，快速跑直道，提高小车的平均速度。而摄像头在这方面有天然的优势——探测距离远，可以采集多行信息。根据实验，最后选择一个场中采集 10 行信息，作为该场的路径识别算法的依据<sup>[23,24]</sup>。下面简单介绍一下该算法。

在黑线中心提取算法的基础上，可以获得该场中 10 行黑线中心的坐标值。我们根据式 (2-1) 算出这 10 行黑线中心坐标的平均值。

$$\bar{X} = \frac{\sum_{row=0}^{row\_max} X[row]}{row\_max} \quad (2-1)$$

然后，根据式 (2-2) 求出每行相对平均值的位移，最后求和。

$$curve = \sum_{row=0}^{row\_max} |X[row] - \bar{X}| \quad (2-2)$$

由于，curve 越大说明跑道弯曲程度越大。因此，给定 curve 一个合适的阈值，就可以正确识别出跑道当前的弯道程度。

实验表明，只要阈值取得合适，该算法非常可靠。其中阈值还可以根据跑道的改变而作相应得变化。

### 2.3.2 舵机驱动信号的设计

舵机驱动信号的实现包括控制算法和驱动信号的设计。这里只介绍信号的软件设计，控制算法将在后面的详细介绍。

舵机控制是智能车系统中很重要的一个环节，舵机控制的好坏也直接影响了小车的控制效果，舵机的控制信号为周期是20ms的脉宽调制（PWM）信号，其中脉冲宽度从0.5ms—2.5ms，相对应舵盘的位置为0—180度，呈线性变化。也就是说，给它提供一定的脉宽，它的输出轴就会保持在一个相对应的角度上，无论外界转矩怎样改变，直到给它提供一个另外宽度的脉冲信号，它才会改变输出角度到新的对应的位置上。因为舵机的控制信号是一个脉宽调制信号，所以很方便和数字系统进行接口。只要能产生标准的控制信号的数字设备都可以用来控制舵机，例如软件计数方式，定时器中断方式，PWM硬件产生方式等<sup>[25,26]</sup>。

智能小车系统实现对舵机输出转角的控制，必须首先完成两个任务：首先是产生基本的PWM周期信号，产生的周期20ms信号；其次是脉宽的调整，即单片机模拟PWM信号的输出，并且调整占空比。为了提高控制精度，在设计中采用一个16位的PWM通道，即PWM0和PWM1联合使用组成。因为舵机转动的时候，都需要一定的时间，因此在写入控制寄存器后，要延迟一定的时间，延迟的时间与需要转动的角度成正比。舵机控制流程图如图2-16所示：

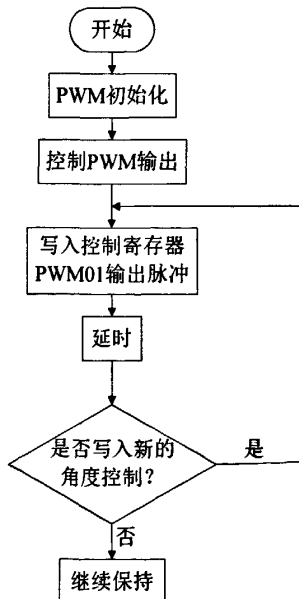


图2-16 舵机控制流程图

在小车的舵机控制算法中，舵机采取的是PID控制。在该控制算法下，小车跑直线时，表现非常好，但是在进曲率半径较小的弯道时，往往会冲出去。我们发现这和比例系数不够大有关，因此我们采用了模糊控制算法，对不同的弯道曲

率用不同的比例参数。我们将在后面的两章中对这两种方法进行全面的介绍和比较，这也将是本文的研究重点。

脉宽调制模块有8路独立的可设置周期和占空比的8位PWM通道，每个通道配有专门的计数器。该模块有4个时钟源，能分别控制8路信号。通过配置寄存器可设置PWM的使能与否、每个通道的工作脉冲极性、每个通道输出的对齐方式、时钟源以及使用方式（八个8位通道还是四个16位通道）。为了提高控制精度，我们将PWM0、PWM1两路8位通道合并为一个16位通道来控制舵机，这样可使舵机的控制精度从1/255提高到1/65536。PWM模块的初始化设置过程为<sup>[27]</sup>：

```
PWME=0x00;           // PWM output is disabled
PWMCTL_CON01=1;     // combine PWM0 and PWM1 to a 16-bit PWM
PWMPRCLK=0x33;      // select clock A and B equal to busclock/8 = 4MHz
PWMSCLA=100;        // clock SA = clock A/2/100 = 20kHz,
                    // control infrared LED array
PWMSCLB=1;          // clock SB = B clock/2/1 = 2MHz
PWMCLK=0x1C;        // select clock A for PWM0 and PWM1, clock SA for
// PWM4, clock SB for PWM2 and PWM3
PWMPOL=0xFF;        // all PWM channels are high at the beginning of the cycle
PWMCAL=0x00;        // select left aligned outputs for each PWM channel
PWMPER0=0x9C;
PWMPER1=0x40;        // 0x9C40 = 40000, set period of PWM1 equal to
                    // 1/4MHz*40000=0.01s or 10ms
PWMDTY0=0x19;
PWMDTY1=0x00; // 0x1900 = 6400, thus the duty cycle is 6400/40000 = 16%
PWME_PWME1=1;      // steer enable
PWMPER2=200;        // set the frequency of PWM2 equal to 10kHz
PWMDTY2=100;        // the duty cycle is 50%
PWME_PWME2=1;      // driven motor enable
```

### 2.3.3 速度信号的设计

驱动电机的软件设计也是通过单片机集成的PWM实现的。由于这里的速度不需要太高精度，故直接采取一路PWM就可以很好的完成对驱动电机的驱动。具体实现的方法与舵机是一样的，就不再累述。这里就介绍一下速度检测的软件实现。

从前面我们知道速度检测是通过红外传感器感应码盘的转数来实现的。光电传感器检测速度的原理和检测赛道黑白线时的原理是一样的。当圆盘随着齿轮转

动时,光电管接收到的反射光强弱交替变化,由此可以得到一系列和小车速度一致的脉冲信号。将脉冲信号输入到单片机ECT 模块中,利用其强大的脉冲捕获功能,同时捕获脉冲的上升沿和下降沿。根据原理,后轮转动一圈,采可以获得32个触发信号。通过累计一定时间内的脉冲数,或者记录相邻脉冲的间隔时间,可以得到和速度对应的参数值<sup>[28]</sup>。

如前所述,在软件上有两种对脉冲信号的处理方法。一种是测试一定时间内的脉冲数,另一种是记录脉冲间隔时间。

对于脉冲累加的方式,ECT (Enhanced Capture Timer) 模块中共有4路8位的脉冲累加器,同时可以组成2路16位脉冲累加器。4路8位从P0-P3引脚接入,如果组成2路16位,则需要从P0和P7引脚接入信号。由于16位脉冲累计无法同时捕获脉冲上升沿与下降沿,为了让单圈信号数增加,提高速度传感器精度,因此选用8位脉冲累加器。这里,将接收管信号接入单片机ECT模块中的P3引脚。首先,通过设置TIOS寄存器,设置P3引脚为输入。然后,设置TCTL4寄存器,选择获取上升沿还是下降沿,或者任何边沿。之后,将ICOVW\_NOVW置为1,保护脉冲累加器的数据。最后通过ICPAR,对脉冲累加器进行使能,设置此寄存器之后,脉冲累加器开始计数。接下去,只要定时读取PACN3这个寄存器,获取当前的脉冲累加值即可。前后两次累加值的差反映的就是当前的速度。整个设置过程如下:

```
TIOS=0x00;           // PT0-2 is set to be IC (input capture) mode
TCTL4=0b11000000;  // PT3 catch any edge;
ICOVW_NOVW3=1;    // protected
ICPAR_PA3EN=1;    // enable pulse accumulator 3
```

对于获取相邻脉冲间隔的方式,利用ECT模块中提供的IC queue mode (队列模式)可以方便地完成这一功能。首先,通过TSCR2寄存器设置预分频数值,即每个脉冲的时间间隔,它决定了测量精度。然后,将TSCR1寄存器中的TEN置位,打开TCNT 计数器。之后,将PT1设为输入捕捉模式,同时设置是用上升沿触发还是下降沿触发,或者二者都行。接下来,设置保护寄存器、打开缓存寄存器、应用队列模式。

```
TSCR2=0x07;           // TCNT prescaler setup 16us per pulse and BIT7 was
                        // set to disable INTERRUPT when TCNT overflow
TSCR1=0x80;           // enable timer TCNT, activate bit TEN
TIOS=0x00;           // PT0-PT2 are set to be IC (input captured) mode
TCTL4_EDG1A=1;       // capture on rising edges or falling edges
TCTL4_EDG1B=1;
ICOVW_NOVW1=0;
```

```
ICSYS_BUFEN=1;      // open hold register
ICSYS_TFMODE=1;     // after transfer the data to hold register, set the flag
C3F
ICSYS_LATQ=0;       // apply queue mode
```

初始化设置完后，只要分别读取TC1、TC1H 的值并作差，就可以得到和当前速度成反比的脉冲间隔值。

### 第三章 基于常规 PID 的智能小车系统控制

从第二章我们给智能小车设计了可靠的导航模块，足以满足小车能够正确识别前方路况。接下来就是给系统提供合适的运动控制算法，供舵机和速度的驱动信号使用。本章将介绍基于常规 PID 控制算法的运动控制系统控制，并给出仿真数据和试验效果。

#### 3.1 常规 PID 控制理论

PID 控制是最早发展起来的控制策略之一，由于其算法简单、可靠性高，被广泛应用于工业过程控制。常规的 PID 控制器作为一种线性控制器，是按照偏差的比例（P-Proportional）、积分（I-Integral）和微分（D-Derivative）的线性组合构成控制量，对被控制对象进行控制<sup>[29]</sup>。其控制规律为：

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

下面以本系统中所用的后轮驱动电机为模型，对常规 PID 控制理论各个方面进行介绍。

##### 3.1.1 比例控制作用对控制品质的影响<sup>[30]</sup>

比例控制器其输出与偏差成比例：

$$u(t) = k_c e(t) + u_0 \quad (3-2)$$

式中， $u(t)$  为控制器输出； $e(t)$  为设定值与测量变送信号之差； $K_c(t)$  为控制器增益； $u_0$  为当偏差  $e(t)$  为零时的输出信号值，它反应比例控制的工作点。

通常，工业用控制器不用  $K_c$  而用比例度（比例带）PB 来进行刻度：

$$PB = \frac{100}{k_c} \quad (3-3)$$

通常  $1 < PB < 500$ 。比例控制器的传递函数表达式为：

$$G_c(s) = k_c \quad (3-4)$$

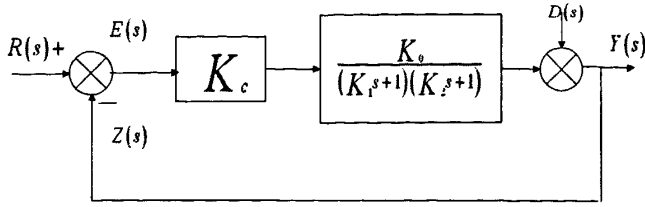


图 3-1 比例控制系统框图

如图 3-1 所示控制系统中，系统在干扰  $D(s)$  作用下的闭环传递函数为：

$$\frac{Y(s)}{D(s)} = \frac{1}{1 + \frac{k_c k_0}{(T_1 s + 1)(T_2 s + 1)}} = \frac{(T_1 s + 1)(T_2 s + 1)}{T_1 T_2 + (T_1 + T_2)s + k_c k_0 + 1} \quad (3-5)$$

以上二阶系统可以从控制系统的阻尼系数  $\varepsilon_p$  进行分析，以此来判定放大系数  $K_c$  与系统稳定性之间的关系：

$$s^2 + 2\varepsilon_p \omega_0 s + \omega^2 = 0$$

式中：

$$\omega^2 = \frac{a_0}{a_1} = \frac{1 + k_0 k}{T_1 T_2}, \quad 2\varepsilon_p \omega_0 = \frac{a_1}{a_2} = \frac{T_1 + T_2}{T_1 T_2}$$

即

$$\varepsilon_p = \frac{a_1}{2\sqrt{a_1 a_2}} = \frac{T_1 + T_2}{2\sqrt{T_1 T_2 (1 + k_c k)}} \quad (3-6)$$

由式 (3-6) 可见：当  $k_c$  减小时， $\varepsilon_p$  值较大，并有可能大于 1，这时过渡过程为不振荡过程。随着  $k_c$  值的增加， $\varepsilon_p$  值逐渐减小，直至小于 1，相应的过渡过程将由不振荡过程而变为不振荡的临界情况，并随  $k_c$  的继续增大， $\varepsilon_p$  继续减小，过渡过程的振荡加剧。但是，不论  $k_c$  值增大到多大， $\varepsilon_p$  不可能小于零，因而这个系统不可能出现发散振荡，即该系统总是稳定的。

对于稳定的系统，可用终值定理求得在幅值为  $A$  的阶跃干扰作用下，系统的稳态值：

$$y(\infty) = \lim_{t \rightarrow \infty} y(t) = \lim_{s \rightarrow 0} s \frac{A(T_1 s + 1)(T_2 s + 1)}{[T_1 T_2 s^2 + (T_1 + T_2)s + k_c k_0 + 1]} = \frac{A}{1 + k_c k} \quad (3-7)$$

式 (3-7) 表明：应用比例控制器构成的系统，其控制结果的稳定值不为零，即系统存在余差。随着控制器放大系数  $k_c$  的增大，余差将减小，但不能完全消除，因而比例控制为有差控制。 $k_c$  大小对受控变量过程的影响所产生的基本矛盾：稳定程度与控制精度（又称为最大偏差与余差）的矛盾。 $k_c$  增加能使控制精度提高，但是稳定程度变差。 $k_c$  参数的整定，就是对这两项指标在作权衡。

表 3-1 Kc 变化对控制过程各项指标的影响

放大系数 K 由小变大		放大系数 Kc 由小变大	
衰减系数	大到小	稳定程序	逐渐降低
衰减比	大到小	最大偏差	大到小
		余 差	大到小

### 3.1.2 积分控制作用对控制品质的影响

积分控制器常和比例控制器一起组成比例积分控制器来完成控制作用。函数关系式如 (3-8) 所示：

$$u = k_c \left( e + \frac{1}{T_i} \int_0^t e dt \right) + u_0 \quad (3-8)$$

如果偏差为零，则积分控制器的输出不变。视偏差是正或负，偏差积分后使控制器输出 u 向上或向下变化（其变化方向还与控制器正、反作用相关）。图 3-2 所示为一个比例积分控制系统：

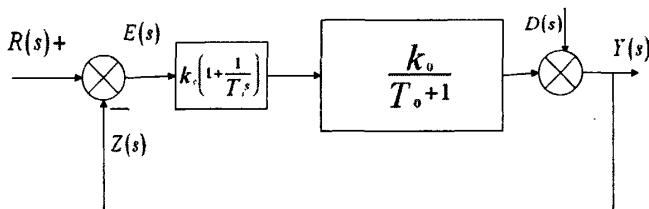


图 3-2 比例积分控制系统

系统在阶跃干扰 d 的作用下，闭环传递函数为：

$$\frac{Y(s)}{D(s)} = \frac{1}{1 + k_c \left( 1 + \frac{1}{T_i s} \right) \left( \frac{k_o}{T_o s + 1} \right)} = \frac{T_i s (T_o s + 1)}{T_i s (T_o s + 1) + k_c k_o (T_i s + 1)} \quad (3-9)$$

假定在阶跃干扰的幅值为 A，则应用终值定理，可求得在此阶跃干扰作用下的系统稳态值：

$$y(\infty) = \lim_{s \rightarrow 0} s Y(s) = 0 \quad (3-10)$$

添加积分作用的基本目的，是在系统经受扰动后，使系统输出返回设定值，即消除余差，这是它独有的特点。

从闭环传递函数特征方程根的性质分析积分作用对控制系统稳定性的影响，即由式 (3-9) 的特征方程根的判别式如下，可得以下三种结果：

$$\Delta = T_i^2 (k_c k + 1)^2 - 4 T_i T k_c k$$



(1) 当  $\Delta > 0$  时, 此式经移向化简可改为  $(k_c k + 1)^2 > \frac{4Tk_c k}{T_1}$ , 只有  $T_1$  较大,

不等式才能成立, 这时特征根  $s_1$ 、 $s_2$  均为负实根, 所以控制系统的过渡过程为非振荡的。

(2) 当  $\Delta = 0$  时, 此式可变换为  $(k_c k + 1)^2 = \frac{4Tk_c k}{T_1}$ , 要使其成立,  $T_1$  一定要

比第一种情况时的值小, 此时特征根  $s_1$ 、 $s_2$  均为两个相等的实根, 控制系统的过渡过程处于临界状态。

(3) 当  $\Delta < 0$  时, 此式可变换为  $(k_c k + 1)^2 < \frac{4Tk_c k}{T_1}$ , 同样, 要使这一关系成

立, 此时的  $T_1$  值要比第一种情况时的  $T_1$  小, 特征根  $s_1$ 、 $s_2$  均为一对共轭复根, 控制系统的过渡过程处于振荡状态, 并且, 随着  $T_1$  的进一步减小, 振荡加剧。

### 3.1.3 微分控制作用对控制品质的影响

比例作用根据偏差的大小进行自动调节, 积分作用可以减小被调参数的余差, 对于一般调节系统来说, 使用比例积分调节器已经满足生产过程自动化的要求了。但对于一些要求比较高的自动化系统常有以下要求: 希望根据被调参数变化的趋势, 而采取调节措施, 防止被调参数产生更大的偏差。这个要求使用具有微分作用的调节器可以得到满足。

所谓被调参数的变化趋势, 就是偏差变化的速度。调节器微分作用的输出余偏差变化的速度成正比。可用式 (3-11) 表示:

$$Y_D = T_D \frac{dX}{dt} \quad (3-11)$$

式中  $\frac{dX}{dt}$  —— 偏差变化的速度;

$T_D$  —— 微分时间。

式 (3-11) 表明微分时间越长, 或偏差变化的速度越大, 则微分作用的输出越大, 对于一个固定不变的偏差, 不管这个偏差有多大, 微分作用的输出总是零。图 3-3 所示为一个比例微分控制系统:

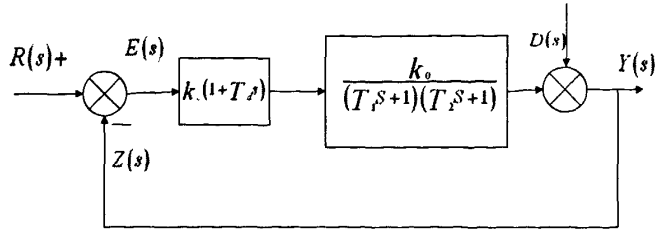


图 3-3 比例微分控制系统

系统在干扰作用下的闭环传递函数为：

$$\frac{Y(s)}{D(s)} = \frac{1}{1 + k_c(1 + T_D s) \frac{k_0}{(T_1 s + 1)(T_2 s + 1)}} \quad (3-12)$$

系统的特征方程式为：

$$T_1 T_2 s^2 + (T_1 + T_2 + k_c k_0 T_D) s + (1 + k_c k_0) = 0 \quad (3-13)$$

或

$$s^2 + 2\varepsilon_D w_0 s + w_0^2 = 0$$

式中

$$2\varepsilon_D w_0 = \frac{T_1 + T_2 + k_c k_0 T_D}{T_1 T_2}, \quad w_0^2 = \frac{1 + k_c k_0}{T_1 T_2}$$

因此，系统的衰减系数为：

$$\varepsilon_D = \frac{T_1 + T_2 + k_c k_0 T_D}{2\sqrt{T_1 T_2 (1 + k_c k_0)}} \quad (3-14)$$

由上式可以得到， $\varepsilon$  值的增加将使系统过渡过程的振荡程度降低，也就是递减比增大，因而在纯比例作用的基础上增加微分作用提高了系统的稳定性，也减小了最大偏差。此时，为了维持原有的递减比，即与纯比例作用具有相同的递减系数，须将放大系数  $K_c$  适当增加，因此引起的稳定性下降由微分作用使稳定性提供来补偿。

系统在幅值为  $A$  的阶跃干扰作用下，由式 (3-10) 应用终值定理可求得过渡过程的稳态值为：

$$y(\infty) = \lim_{s \rightarrow 0} \frac{A}{s} \frac{(T_1 s + 1)(T_2 s + 1)}{T_1 T_2 s^2 + (T_1 + T_2 + k_c k_0 T_D) s + (1 + k_c k_0)} = \frac{A}{1 + k_c k_0} \quad (3-15)$$

由此可见，微分作用不能消除余差。但如上所述，这时  $K_c$  值较纯比例作用时的  $K_c$  为大，所以，余差较纯比例作用时小。另外，微分作用是按偏差变化的

速度来工作的，因而对于克服对象容量滞后的影响有明显的作用，但对于纯滞后则无能为力。

综上所述，控制系统引入微分作用后，将全面提高控制品质。但如果控制器的微分时间  $T_d$  整定得太大，这时即使偏差变化的速度不是很大，也会因微分作用太强而使控制器的输出发生很大变化，从而引起控制阀时而全开，时而全关，如同双位控制，将严重影响控制品质和安全生产。因此，控制器参数整定时，不能把  $T_d$  取得太大，而应根据对象特性和控制要求做具体分析。

由上述分析可知：积分作用能消除余差，但降低了系统的稳定性，特别是当  $T_I$  比较小时，稳定性下降较为严重。因此，控制器在参数整定时，欲得到纯比例作用时相同的稳定性，当引入积分作用之后，应当把  $K_c$  适当减小，以补偿积分作用造成的稳定性下降。

### 3.1.4 数字 PID 控制器

在计算机控制系统中，使用的是数字 PID 控制器。由于计算机控制是一种采样控制，它只能根据采样时刻的偏差计算控制量，因此式 (3-1) 中的积分和微分项不能直接使用，需要进行离散化处理。

#### 1、位置式 PID 控制算法

处理后的表达式为：

$$u(k) = k_p e(k) + k_i \sum_{j=0}^k e(j) + k_d [e(k) - e(k-1)] \quad (3-16)$$

式 (3-16) 为位置式 PID 控制算法，式中： $u(k)$  为第  $k$  次采样时刻的计算机输出值； $e(k)$  为第  $k$  次采样时刻输入的偏差值； $e(k-1)$  为第  $(k-1)$  次采样时刻输入的偏差； $k_i$  为积分系数， $k_i = k_p \frac{T}{T_i}$ ； $k_d$  为微分系数， $k_d = k_p \frac{T}{T_d}$ ； $T$  为采样周期。

另外，为改善 PID 控制的动态特性，人们提出了多种改进的数字 PID 控制算法，如积分分离 PID 控制算法、遇限削弱积分 PID 控制算法、不完全微分 PID 控制算法、微分先行 PID 控制算法及带死区的 PID 控制算法等。在此不再详细介绍。

对于不同的被控对象只要适当地整定 PID 的三个参数，就可以获得满意的控制效果，实际上它是对比例、积分和微分三部分控制作用的折中。

### 3.1.5 PID 控制器参数整定

PID 控制器参数整定是指在控制器的形式已经确定 (PI, PID 调节规律) 的情况下, 通过调整控制器参数, 达到要求的控制目标。另外, 数字式 PID 控制器还需要决定控制系统的采样周期  $T$ 。多年来人们致力于研究 PID 控制器参数的整定方法, 提出了各种各样的方法<sup>[31]</sup>。归纳起来大致有以下几类:

#### 1、基于对象参数辨识的整定方法

辨识法适用于模型结构已知, 模型参数未知的对象, 采用系统辨识的方法得到过程模型参数, 并依据参数估计值进行参数调整的确定性等价控制规律结合起来, 综合出所需的控制器参数, 周期性地更新控制器参数。在这类方法中, 不同的辨识方法和整定算法的组合将形成不同的整定方案。常用的辨识方法有参数模型辨识方法和非参数模型辨识方法。

对象参数模型辨识方法 (亦称现代的辨识方法) 是在假定一种模型结构的基础上, 通过极小化模型与过程之间的误差准则函数来确定模型的参数, 比较常用的方法有最小二乘法、梯度校正法、极大似然法。这类方法对特性分明的被控对象的控制参数整定是十分有效的, 但这种方法比较复杂, 要得到精确的数学模型, 需要较复杂的试验手段和数学手段, 并且这种方法对被控过程模型有较强的限制, 因而对不能或难以用精确数学模型描述的复杂过程难以奏效, 这也是目前阻碍这类方法广泛应用的主要原因之一。

非参数模型辨识方法 (亦称经典辨识方法) 获得的模型是对象的非参数模型, 即对象的阶跃响应、脉冲响应、频率响应等, 其表现形式是以时间或频率为自变量的实验曲线。这种方法在假定过程在线性的前提下, 不必事先确定模型的具体结构, 因而可适用于任意复杂的过程。其所得的非参数模型经适当的数学处理, 可转变为参数模型——传递函数形式, 而后应用适当的整定方法或计算公式可得控制器参数。

目前工程上常用的方法是对过程对象施加阶跃输入信号, 测取过程对象的阶跃响应, 然后由阶跃响应曲线确定过程的近似传递函数。当阶跃响应曲线比较规则时, 近似法、半对数法、切线法和两点法都能比较有效地导出近似传递函数。辨识出对象的数学模型后, 就可应用前面介绍过的整定方法进行控制器参数整定。但这并不意味着就一定能够转变为用户的效益, 因为与此方法相关联的一些问题, 例如闭环辨识、时滞估计、测量噪声和干扰输入的抑制以及安全保护措施等, 虽然已被了解, 但并未得到有效解决。

#### 2、抽取过程对象输出响应特征值的控制器参数

整定方法基于对象模型辨识的参数整定方法算法较复杂, 且从控制器参数整定的角度看, 对象模型中含有的冗余信息量往往很大。由此可见, 可以压缩对象

模型的信息量,而抽取其主要特征进行参数整定。目前,基于对象输出响应特征值来进行 PID 参数整定的方法较多,比较常用的是基于开环对象 Nyquist 曲线上的一个特征点的知识来进行控制参数整定,比较著名的有闭环 Z-N 方法、继电整定法等。

闭环 Z-N 方法(也称临界比例度法、稳定边界法),是 Ziegler 和 Nichlos 在 1942 年提出的,方法是将对象与一纯比例控制器接成闭环,将比例作用由小到大变化,直至系统输出出现不衰减的等幅振荡,记录下临界振荡周期  $P_u$  和增益  $G_u$ ,则控制器参数可通过查表确定。

Astrom 等人于 1984 年提出了继电整定法。用继电特性的非线性环节代替 Z-N 法中的比例控制器,能使闭环系统自动地稳定在等幅振荡的状态,振荡的幅值也可通过改变继电特性的特征值控制。其优点是不会出现增幅振荡现象,更不会使系统毁坏;缺点是对于纯滞后很小的低阶系统,整定得到的参数往往偏大,这是由于开环 Nyquist 曲线与负实轴的交点离原点太近所致。另外,当存在噪声的场合,需用带滞环的继电环节,而不能使用理想的继电环节。由继电法得到特征参数后,可用幅值、相位预度法(GPM)整定 PID 参数,也可用其他方法和公式。

### 3、参数优化方法

控制器的参数整定问题从运筹学的角度看,就是调整控制器的参数,在满足一定约束条件下,使某个目标函数达到最优(最大或最小)。这样就需要解决两方面的问题:提出合适的目标函数和采用适当的寻优策略。用于控制器参数整定、寻优的目标函数必须与系统调节指标函数密切相关,反映系统的调节品质。

参数优化有两种途径,一是间接寻优,即写出目标函数的解析式,然后根据目标函数取极值的充分与必要条件,求出参数的最优解。另一种途径是直接寻优,即直接在参数空间中按照一定的规律进行探索寻优,寻得的目标函数即为最小参数点。这种方法适用于变量 PID 控制系统和火电机组协调控制等系统。

### 4、基于控制器自身行为的 PID 参数整定方法

近年来,随着专家系统、模糊控制及人工神经网络等人工智能技术的发展,智能控制得到迅速发展,而将智能技术应用于参数整定,就产生了“智能整定”这一新的 PID 参数整定方法。这种智能整定方法不依赖于对象的数学模型,而是根据自身的控制行为来调整控制参数。系统的控制行为表现为偏差  $e$  和偏差变化率  $e'$ 。

PID 控制是由比例作用、积分作用、微分作用三部分组合而成的,其中积分作用是对象过去受控效果的总和,比例作用是过去控制效果的现时表现,而微分作用是对未来控制作用的需求。因此 PID 控制规律是根据对象对控制作用的历史

效果、现时表现及未来需求的综合来确定的，而不是靠对象的具体数学模型来决定的。如果我们根据控制行为的反映：偏差 $e$ 、偏差变化率 $e'$ 、 $\sum e$ 自动地校正 $K_p$ 、 $T_i$ 、 $T_d$ ，从而获得良好的控制效果。智能整定多应用模糊理论，还有神经网络整定法及各种专家系统整定方法。

### 5、其他整定方法

除了上面介绍的几大类外，PID参数整定还有很多有特色的方法。目前，工程中广泛采用以下几种方法：动态特性参数法、稳定边界法、衰减曲线法和模糊控制法。接下来就将介绍采用该PID控制算法及合适的参数整定法实现对小车运动控制系统的控制。

## 3.2 小车运动系统的常规PID控制

根据前面一节介绍我们对常规PID控制理论有了一个清晰的理解。下面详细介绍运动系统基于常规PID控制算法的实际，即PID控制器模型的建立和仿真。

### 3.2.1 速度PID控制器的设计分析

在智能汽车控制系统中，整个小车通过一个直流电机来驱动。所以，控制小车的行驶速度就是控制电机的转速。这是一个典型的直流电机速度控制系统，在该系统中，控制对象就是直流电机，控制变量是PWM信号占空比的一个不断的变化值，对象变量是编码器测得的电机当前的实际转速值。在该系统中，仅使用比例——微分就能满足控制要求，所以没有加入积分项。

虽然赛道是未知的，但是我们可以根据赛道的固有特性对赛道进行分析。赛道主要包括5种情况：直道、十字路口、大弯道、U型弯道、急转弯和S型弯道。其中十字路口我们可以当作直道处理。因此通过第二章的路径识别模块介绍我们可以得出一个反应弯道曲率的变量CURVE。这样我们就可以将CURVE的变化量作为我们速度PID控制器的输入量，而电机控制算法的作用是接受指令速度值，通过运算向电机提供适当的驱动电压，尽快地、尽量平稳地使电机转速达到速度值，并维持这个速度值。换言之，一旦电机转速达到了指令速度值，即使遇到各种不利因素的干扰下也应该保持该速度值不变。

控制算法（有时也称作控制规则），是任何闭环控制方法的核心。现在已经有各种各样的控制算法，PID控制算法是控制系统中技术比较成熟，而且应用最广泛的一种控制器。它的结构简单，参数容易调整，不一定需要系统的确切数据模型，因此在工业的各个领域中都有应用。

PID控制器最先出现在模拟控制系统中，传统的模拟PID控制器是通过硬件（电子元件、气动和液压元件）来实现它的功能。随着计算机的出现，把它移植到

计算机控制系统中来,将原来的硬件实现的功能用软件来代替,因此称作数字工控制器,所形成的一整套算法则称为数字PID算法。数字PID控制器与模拟PID控制器相比,具有非常强的灵活性,可以根据试验和经验在线调整参数,因此可以得到更好的控制性能。

在设计算法之前,先说明一些控制算法中常用的术语:

(1) 控制对象。有时也称作执行装置,它是控制回路所要控制的对象。  
 (2) 控制变量。它既是控制回路的输出量,也是控制对象的输入量,通常将其定义为CV。

(3) 对象变量。通常简写为PV,是控制对象的反馈值。

(4) 期望值。通常简写为SP,是人们所希望的控制对象的取值。

不论具体的控制算法是否属于PID算法,许多基于偏差的单输出控制系统的操作都可以归结为以下几个步骤:

第一步:读取当前指定的控制对象的SP值。

第二步:读取PV值。

第三步:用SP减去PV得到偏差信号。偏差信号表明控制对象的输出与期望值之间相差多少。偏差信号的值较大,说明控制对象的输出与期望值(SP)相差得还很远;偏差信号的值较小,说明两者相差不多;偏差信号的值为零,说明控制对象的输出符合期望值。偏差信号的值可以为正,也可以为负。

第四步:如果偏差信号值不等于零,就应该将其转换为有效的CV值输出,目的在于使控制对象的输出更为接近SP。换句话说,即从偏差信号中获得新的CV值(该值用于调节电机的供给电压),从而使PV值(测得的电机的当前转速值)更加接近SP值(电机的期望值)。注意,某些控制器会在一定范围内忽略较小的偏差信号,这各范围被称为“死区”。

第五步:重复前面四步,往返循环反馈控制。

上述控制系统的方框图如图3-4所示:

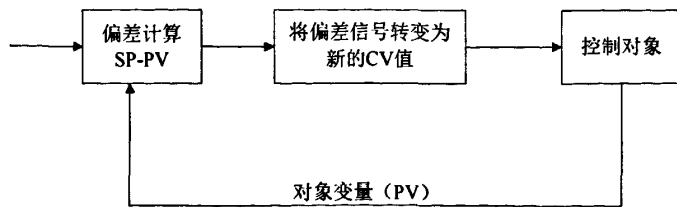


图3-4 闭环控制系统框图

在PID控制器中,偏差转换环节对应于前述的步骤四。它可以分解为三个环节:比例环节、积分环节和微分环节。这些环节同时起转换作用。三个环节的输出求和后共同作用于控制对象,产生新的输出。图3-5表示一个常用的PID系统控

制回路。

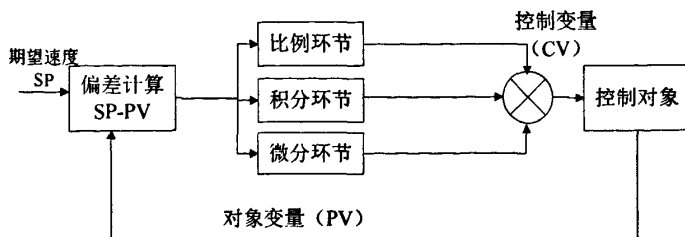


图3-5 PID系统控制框图

某些情况下，可以不必使用积分或是微分控制器。不使用积分控制器，仅使用比例—微分（简称PD）控制器。同样地，也可以既不使用微分，也不使用积分控制器，这样就只剩下简单的比例（简称P）控制器。

### 1、驱动电机PD控制结构

在智能汽车控制系统中，整个小车通过一个直流电机来驱动。所以，控制小车的行驶速度就是控制电机的转速，这是一个典型的直流电机速度控制系统。在该系统中，控制对象就是直流电机，控制变量是PWM信号占空比的一个不断的变化值，对象变量是编码器测得的电机当前的实际转速值。在该系统中，仅使用比例—微分（PD）就能满足控制要求，所以没有加入积分项。控制框图如图3-6所示：

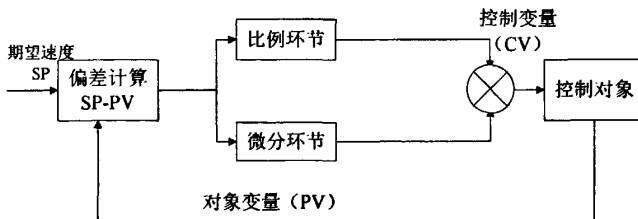


图3-6 PD控制框图

下面介绍PD控制算法的具体设计过程：

(1) 比例项的设计。控制回路中的第一个偏差转换环节就是比例项。这一环节简单地将偏差信号乘以比例常数 $K_p$ 得到新的CV值。基本的比例控制器的算法如

以下代码所示：

Loop:

```
PV=ReadMotorSpeed();    //read the PV
Err=SV-PV;
CV=Err*Kp;
SetPwm(CV);
```



### Go to Loop;

注意上一段程序中的SetPwm()函数并非将CV值作为绝对的PWM占空比来对待。否则,不断降低的偏差值会使输出值接近零,而且由于电机工作时需要持续的PWM信号,控制系统将使电机稳定在低速运转状态上,从而导致控制策略失败。

相反, CV值被取作当前占空比PWM的改变量,并被附加到当前的PWM占空比上。这也要求SetPwm()函数必须将相加后所得的PWM占空比限制在0%-100%之间。

正的CV值将使电机两端的电压增加,负的CV值将使电机两端的电压降低,如果CV值等于0,则无须改变当前占空比。

#### (2) 微分项的设计。

任何变量的微分被用来描述该变量是如何相对于另一个变量(多为时间)变化的,换句话说,任何变量的微分就是它随时间的变化率。

在PD控制器中,绝大多数把控制器微分项定义为:

$$\text{Rate}=(E-\text{Elast})/T$$

其中, E为当前偏差值, Elast为前次偏差值, T为两次测量时间间隔。在这里假设T等于1,从而得到偏差信号的变化率:

$$\text{Rate}=E-\text{Elast}$$

当微分想被具体应用到控制器中时,将一个常数Kd,乘以该微分项,并将它加到比例项上,就可以得到最终的值计算公式:

$$\text{CV}=(\text{Kd}*E)+(\text{Kp}*Rate)$$

当偏差信号接近零时, CV值将为负,所以当偏差信号开始改善时,微分项的作用将逐渐减弱校正输出量。在某些场合下,微分项还有利于超调量的消除,并可以允许使用较大的Kp值,从而可以改善响应的快速性。在某种意义上,微分环节还预示了偏差信号的变化趋势。当控制对象对控制器的输出响应迟缓时,微分环节的作用尤为明显。

含有微分项的控制算法的伪代码实现如下:

Loop:

PV=ReadMotorSpeed();

LastError=Error;

Error=SP-PV;

Rate=Error-LastError;

CV=Kp\*Error+Rate\*Kd;

SetPwm(CV);

### Go to Loop;

微分环节的引入改善了系统的动态特性，但微分对于干扰非常敏感，这使它可能会造成不利的影晌。由于采样周期一般都非常小，差分（特别是二阶差分）对数据误差和噪声特别敏感，一旦出项干扰，表现为差分突然变大，从而引起控制量的非正常增大；假如这时系统已进入稳态，干扰会通过微分项使系统产生振荡。为了避免微分项对控制系统的影响过大，在设计控制器时限定微分结果的范围。

## 2、PD控制系统的参数和采样周期的选择

### (1) PD参数选择

在对电动机控制中，首先要求系统是稳定的，在给定值变化时，被控量应能迅速、平稳地跟踪，超调量要小。在各种干扰下，被控量应能保持在给定值附近。另外，控制变量不宜过大，以避免系统过载。显然，上述要求都满足是很困难的，因此，必须根据具体的实际情况，抓主要方面，兼顾其他方面。

在选择控制器参数之前，应首先确定控制器结构。在小车控制系统里，在设计中，选用PD控制器，所以下面将重点讨论控制器PD参数的确定。

PD参数的选择有两种可用方法：理论设计法和实验设计法。理论设计法确定PD控制参数的前提，是要有被控对象准确的数学模型，这在电动机控制中很难做到，因此，用试验法来确定PD控制参数，就成为目前经常采用的，并且是行之有效的方法。

凑试法是通过模拟或闭环运行系统，来观察系统的响应曲线，然后根据各个控制参数对系统响应的大致影响，来改变参数，反复凑试，直到认为得到满意的响应为止。凑试前，要先了解控制器参数值对系统的响应有哪些影响。增大比例系数，可以加快系统的响应速度，有利于减少静态误差但是，过大的比例系数会是系统有较大的超调，因此产生振荡，破坏系统的稳定性。增大微分常数，可以加快系统的响应，使超调量减少，稳定性增加但系统的抗干扰能力降低。如图3-7所示为不稳定的控制系统。

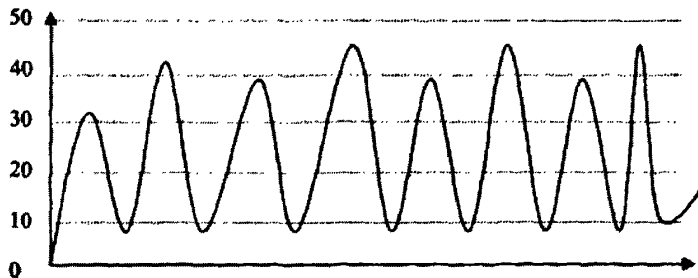


图3-7 不稳定控制系统

在考虑了以上参数对控制过程的影响后，凑试时，可以按照先比例，再微分

的顺序反复调节参数，具体的步骤如下：

- A、首先只调整比例部分，将比例系数由小到大，并观察系统所对应的响应，直到得到响应快、超调量小的响应曲线为止。
- B、在凑试前，先给一个很小的微分常数，以后逐渐增大，同时相应地改变比例系数，直到获得满意的效果为止。

这里所谓的“满意”的效果，是根据被控对象的不同和对控制要求的不同，而得到的相对满意程度。因为比例、微分二者的控制作用有相互重叠之处，某一环节作用的减小往往可以由其他环节作用的增加来补偿，因此，能达到“满意”效果的参数组合并不是唯一的。

另外，需要指出的是，为了尽可能的提高指令的执行速度，提高系统的实行性，在参数设计中尽可能不使用浮点数，例如比例系数、微分系数可以使用移位的方法来近似浮点数运算。

### (2) 采样周期的选择

数字PID控制算法是模仿连续系统的PID控制器，在近似离散化的基础上，通过计算机实现数字控制。这种控制方式要求采样周期要足够短，一般要远小于系统的时间常数，这是采用数字PID控制器的前提。采样周期越小，数字控制效果就越接近连续控制。采样周期的选择要受到多方面因素的影响（比如单片机的速度）。在电动机控制软件设计中，采样周期也是一个重要因素。在实际选择采样周期时，必须从需要和可能性两方面综合考虑，一般要考虑的因素有如下：

A、从调节品质和数字算法要求方面考虑，采样周期应取得短些。一般说来，控制精度要求越高，采样周期应该越短。采样周期应比被控对象的时间常数小得多，否则，采样信号无法反映系统的瞬变过程。

B、为了使连续信号采样后输入计算机而不失真，应根据香农采样定理，采样周期需要满足如下关系式：

$$T < \frac{1}{2f_{\max}}$$

其中， $f_{\max}$  为被采样信号的最高频率，T为采样周期。

由于 $f_{\max}$ 难准确的确定最大值，所以如果按照香农定理，采样周期其实还要缩小4至6倍。

C、从控制系统的动态性能和抗干扰性能来考虑，也要求采样周期短些。这样，给定值的改变可以迅速通过采样得到，而不至于在控制中产生较大的延迟，此外，对低频扰动，采用短的采样周期可以迅速加以校正。

D、从执行元件的响应速度和要求来看，有时需要输出信号保持一定的时间。

如果执行元件响应速度慢，那么过短的采样周期往往没有必要。

E、从单片机控制在一个采样周期内要完成的运算工作量来考虑，一般要求采样周期长些，以保证单片机有充分的实时响应时间和处理时间。

F、从单片机本身的精度考虑，过短的采样周期是不实际的。这是因为目前用于电动机控制的单片机的字长一般都较短，并且多采用定点数运算。如果采样周期过短，前、后两次采样信号的数值接近，反而因单片机的运算精度不高而无法区分，使控制作用减弱。

从以上分析可以看出，各种因素对采样周期的要求是不同的，甚至是矛盾的，因此必须根据具体情况和要求综合作出选择。

### 3.2.2 小车速度PID控制器建立和仿真

#### 1、PID控制器的建立<sup>[32]</sup>

根据上述可以得出速度增量PID控制器的数学表达式：

$$\Delta u(k) = q_0 e(k) + q_1 e(k-1) + q_2 e(k-2) \quad (3-17)$$

其中，T为比例采样周期， $K=N*T$ ， $u_0$ 为控制量的基值，即 $K=0$ 时的控制。q为合并后的各项的系数。式(3-17)表示了各次误差量的对速度控制量的影响，而且，PID算法只要存储最近的三次采样值就可以了。

关于 $u_0$ 的确定，我们可以通过路径识别模块可以预判前方的路径，一般会出现以下几种路况：直线、大弯道、十字路口、U型弯道、急转弯、S弯道。针对这几种路型，我们设定了三种速度值：最大速度 $V_{max}$ 、中速 $V_{mid}$ 、慢速 $V_{min}$ 。前面的三种路况，我们采用最大速度行驶，U型弯道采用中速行驶，急转弯采用慢速行驶。S弯道是最特殊的因为它如果判断出是S弯道就可以当成直线采用最大速度行驶，但是我们摄像头又不能直接完成这种功能，而是会检测成几个急转弯连在一起。因此我们在第一圈的时候采用慢速安全行驶，然后第二圈的时候通过记忆算法实现对S弯道的识别，从而以最大速度通过S弯道。故在软件设计中，将路况分为max、mid、min三种，分别对应快速、中速、慢速。这三种速度就作为各个路况的对应的初始输入值 $u_0$ ，而实际速度就通过测速模块获得，两者之间的偏差作为偏差值。

在本次整定参数的过程中，我们通过串口工具将调速实验中的速度(脉冲数)和控制量(PWM占空比)读出后分析，以推测参数调整的方向。特别注意的是，总结出关于PID参数整定的一些经验：

(1) 对控制对象的不同阶跃控制量下的动态响应和稳态响应要有认识。具体来说，应该认真记录不同PWM占空比下的动态响应曲线和稳态值。特别是在最

大控制量PWM占空比下，车速上升情况和达到稳态后撤去控制电压时，速度降落情况，代表了系统在最强的控制下的响应情况。

(2) 控制的动态性能要求要和系统稳态响应以及积分项系数 $K_i$ 紧密结合考虑，例如要求车速从0开始经 $N$ 个控制周期末达到某个值，则可以通过稳态值记录查出对应这个给定速度大概需要的PWM占空比，然后结合对动态时间的要求，估计出需要的 $K_i$ 值。

反复调节参数，却还对动态性能不满意时，查阅在最大最小控制量下的动态响应记录，看要求是否在系统的控制能力之内。如果不在单向控制能力之内，则考虑降低调速性能要求（转而从能否提前控制角度考虑）或增大控制能力（主要针对降速，可采用PWM双向控制）。若在单向调节能力范围之内，可考虑使用BANG-BANG控制，来提高动态性能并缩短调试周期。

至于BANG-BANG控制，即当速度偏差值在一定控制域值范围之外时，采用最大或最小控制量输出。实际上，域值的确定在很大程度上取决于调试者的主观因素，是一种经验性的控制。为了尽量减少主观偏见的影响，就必须需结合先前的记录数据，进行分析不同速度在最大或最小控制的下几个控制周期的值。若速度很快就达到并超过需要，则BANG-BANG控制在这样的参数下就很容易发生振荡或超调。

## 2、MATLAB仿真<sup>[33,34]</sup>

电机在开环控制系统下达到稳定速度要0.12s左右，直流电机的传递函数可简化为二阶系统，根据二阶系统调节时间可简单估算控制对象的传递函数：

$$G(s) = \frac{400}{s^2 + 50s}$$

通过MATLAB工具，编写PID数字控制算法，在输入信号为阶跃信号下，采样时间设置为20ms，选用参数 $q_0=8$ ， $q_1=0.1$ ， $q_2=10$ ，仿真效果如图3-8所示：

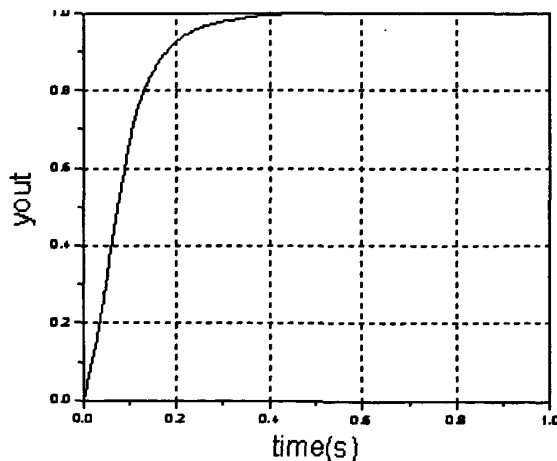


图3-8 阶跃响应仿真结果

在  $q_0=10$ , 其他参数不变的情况下, 仿真效果图如下:

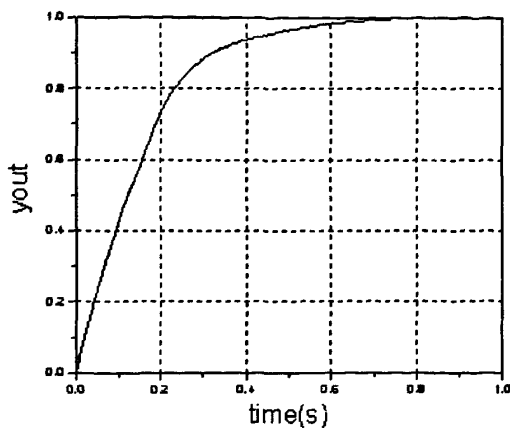


图3-9 增大比例环节Kp值的变化响应图

在  $q_1=0.3$ , 其他参数不变的情况下, 仿真效果图如下:

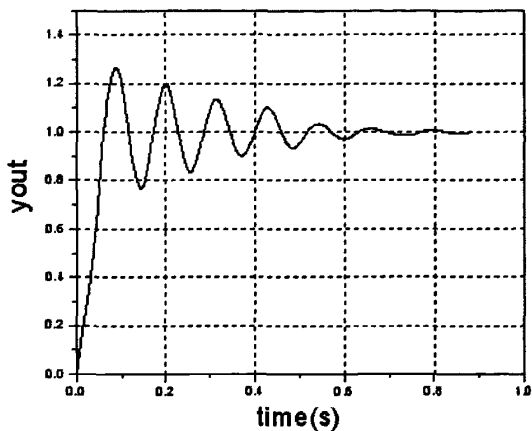


图3-10 增加比例环节Ki值的变化响应曲线

在  $q_2=18$ , 其他参数不变的情况下, 仿真效果图如下:

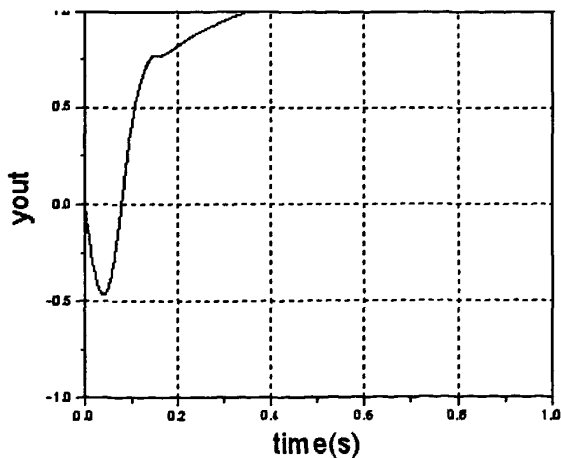


图3-11 增加比例环节Kd的变化响应曲线

由上可以看出,  $q_0=8$ ,  $q_1=0.1$ ,  $q_2=10$ , 采样时间设置为20ms系统响应效果最好。固小车的速度控制就采用PID控制和BANG-BANG相结合的方法实现。使得小车的速度调节能到快速响应以及超调量小, 几乎避免了振荡现象。

### 3.2.3 舵机 PID 控制器的设计

舵机的 PID 控制器设计原理与速度 PID 控制器设计是相同的。只是输入偏差量与输出量代表的意义不同。在这里舵机的我们给舵机设定一个原点, 也就是舵机处于正中央的位置时我们定义为零度。将这个作为偏差的参考点, 建立一个一维的坐标系。左边为负, 右边为正。这样我们先建立一个简单的比例控制器。根据前面我们对 PID 控制理论和速度 PID 控制器的设计我们可以举一反三得到舵机的 P 控制器数学表达式:

$$u(t) = k_p e(t)$$

为了防止舵机的振荡, 我们将 I 参数和 D 参数定为零。通过试验采用舵机在速度较慢或者弯道不很弯的情况效果较好, 能够正确的控制舵机安全的跑完整个赛道。但是这就回到本次智能小车的目标: 安全的在最短时间内跑完整个赛道。而采用常规 PID 控制舵机显然不能达到这个效果。因此我们采用了模糊 PID 控制算法, 在下一章中将详细阐述模糊控制理论以及基于该算法对舵机的控制。

## 3.3 本章小结

本章首先介绍了常规 PID 控制基本理论和控制器设计及其参数整定的一般方法, 然后在这基础上对建立了小车运动控制系统常规 PID 控制器, 并进行试验仿真。我们发现, 基于常规 PID 控制的速度调节能够达到很好的效果, 具体表现在准确性、快速响应性、消除超调现象和振荡。但是舵机的常规 PID 控制效果却很不好, 在实验过程经常会多跑路线甚至过冲现象。因此, 在下一章中我们将讲解基于模糊 PID 的舵机控制。

## 第四章 基于模糊PID的智能小车系统控制

本章通过模糊控制的思想来对智能车舵机的控制算法进行了优化。实际解决了智能车在拐弯时超调过大的问题。文中介绍了模糊控制的理论和S12特有的模糊控制指令。在实际经验的基础上设计出了模糊控制算法，并给出了控制规则。实验结果表明，设计的算法在实际减小智能车在转弯时的超调具有较好的效果。并且在最后，研究了将记忆算法与两种PID控制算法结合起来，使得小车的运行效果达到一个新的层次。

模糊控制（Fuzzy Control）是指模糊理论在控制技术上的应用。它用语言变量代替数学变量或两者结合应用；用模糊条件语句来刻画变量间的函数关系；用模糊算法来刻画复杂关系，是具有模拟人类学习和自适应能力的控制系统。模糊控制技术适用于被控过程没有数学模型或很难建立数学模型的工业过程，这些过程参数变动、时变、呈现极强的非线性特征、并伴有时变及滞后，这类系统的数学模型非常复杂或根本就不存在，不能用常规控制方法控制系统。但是因为模糊控制不需要精确的数学模型，所以在解决这类问题时更有优势。

而我们的智能车舵机之所以采用模糊控制的方法也正是基于这样的考虑。

1. 赛道预先未知，无法采用清晰的算法来对小车进行合适的控制。
2. 小车本身由于重量，机械结构的偏差导致我们在建立车体的数学模型时有很大的困难。
3. 为保证以最快的速度跑完全程，在不同类型的赛道上（如直道，大小弯，S弯）小车的运动策略是不一样的，用其他的常规算法可以保证跑完赛道，但是无法保证时间，使用模糊控制算法可以使我们采用更为灵活的控制策略。

### 4.1 模糊控制理论概述

介绍了模糊控制系统和模糊控制器的构架，并介绍了设计模糊控制算法的经典方法和流程<sup>[35]</sup>。

#### 4.1.1 模糊控制系统的组成

模糊控制属于计算机数字控制的一种形式，因此，模糊控制系统的组成类似于一般的数字控制系统，其结构如图4-1所示：



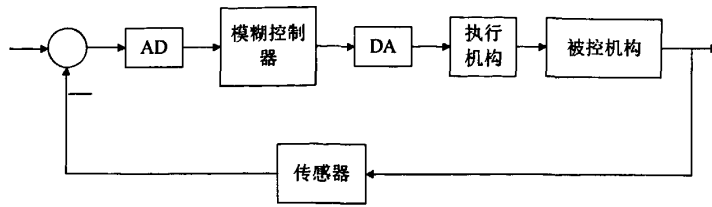


图4-1 模糊控制系统结构

模糊控制系统一般可分为五个组成部分：

- (1) 模糊控制器。是模糊控制系统中的核心部分。
- (2) 输入 / 输出接口。模糊控制器通过输入 / 输出接口，从被控对象获取数字量信号，并将模糊控制器决策的输出数字信号经过数模转换，将其变换为模拟信号，然后送给被控对象。
- (3) 执行机构。包括直流电动机、步进电动机。
- (4) 被控对象。被控对象可以是确定的或模糊的、单变量的、有滞后或无滞后的，也可以是线性的或非线性的、定常的或时变的，以及具有强耦合和干扰等多种情况。对于那些难以建立精确数学模型的复杂对象，更适宜采用模糊控制。
- (5) 传感器。传感器是将被控对象或各种过程的非电量转换为电信号的一类装置。

#### 4.1.2 模糊控制器的基本结构

模糊控制器的基本结构由4个重要部件组成，如图4-2所示：知识库、模糊推理单元、模糊化输入接口与去模糊化输出接口。

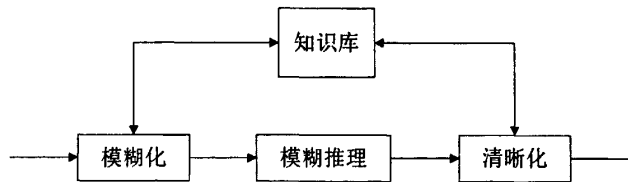


图4-2 模糊控制器基本结构

知识库又包含模糊if-then规则库和数据库，规则库中的模糊规则定义体现了与邻域问题有关的专家经验或知识，而数据库则定义隶属函数、尺度变化因子以及模糊分级数等。推理单元按照这些规则和所给的事实执行推理过程，求得合理的输出。模糊输入接口将明确的输入转换模糊量，并用模糊集合表示，根据模糊输入得到控制量，控制量也是模糊量，因此，要求清晰化过程，把模糊控制量转换为清晰值作为模糊控制器的输出，去模糊输出接口就是将模糊的计算结果转换为明确的输出。

### 4.1.3 设计模糊控制器的流程

#### 1、确定模糊化接口

针对模糊控制器每个输入空间定义一个语言变量。在模糊控制系统中，通常取系统的误差值 $e$ 和误差变化率 $ec$ 为模糊控制器的两个输入。

(1)在 $e$ 上定义语言变量“误差E”，在 $ec$ 上定义语言变量“误差变化EC”；

(2)定义各语言变量的论域。通常E和EC的论域均设置为：

$$X = Y = \{-6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6\};$$

在输入量进行模糊化之前，需将 $e$ 和 $ec$ 的论域变换到E和EC的论域。

(3)定义各语言变量的语言值。通常 $e$ 和 $ec$ 的取值为：

$$\begin{aligned} & \{ \text{“正大”}, \text{“正中”}, \text{“正小”}, \text{“零”}, \text{“负小”}, \text{“负中”}, \text{“负大”} \} \\ & = \{ PB, PM, PS, Z, NS, NM, NB \}. \end{aligned}$$

(4)各语言变量的论域上定义其语言值的隶属函数。通常隶属函数的形状取正态分布函数，例如高斯基函数。隶属函数的形状和分布如图4-3所示：

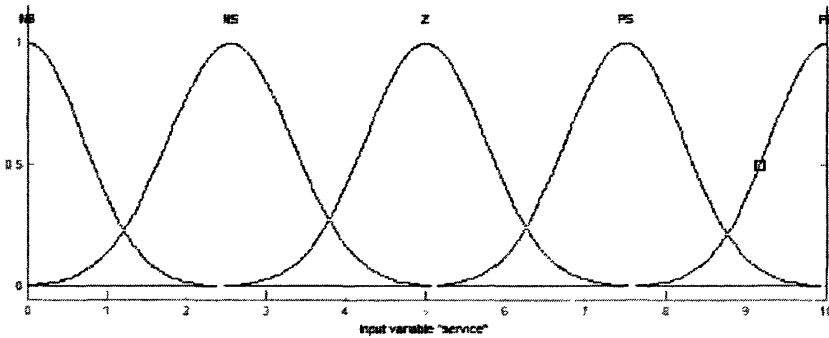


图4-3 隶属度函数示例图

#### 2、确定模糊控制器的规则

在模糊控制器中主要使用Mamdani模糊模型和Takagi——Sugeno模糊模型。

##### (1) Mamdani模糊模型

Mamdani模糊模型是一种语言模型，利用Mamdani模型构成的模糊逻辑系统实质上是一组IF——THEN规则，在这组规则中，前件变量和后件变量均为模糊语言集合，其一般形式如下：

$R$  : 如果 $x$ 是 $A_1$ , and  $y$ 是 $B_1$ , 则 $z$ 是 $C_1$ ;

$R_2$ : 如果 $x$ 是 $A_2$ , and  $y$ 是 $B_2$ , 则 $z$ 是 $C_2$ ;

.....

$R_n$  : 如果 $x$ 是 $A_n$ , and  $y$ 是 $B_n$ , 则 $z$ 是 $C_n$ 。

在现有的模糊控制系统中，大多采用这种形式。

##### (2) Takagi——Sugeno模糊模型

这种模型也是基于IF—THEN规则，规则的前件含有模糊语言值，后件是前件变量的函数。

即：如果 $x$ 是 $A$ ，and  $y$ 是 $B_i$ ，则 $z=f_S(x, \dots, y)$ 。

模糊控制器的控制规则是由一组彼此通过“或”的关系连结起来的模糊语句来描述的。集中每一条模糊条件语句，当输入、输出语言变量在各自论域上反映各语言值的模糊子集为已知时，都可以表达为论域的积集上的模糊关系，即 $R_1, R_2, \dots, R_n$ ；计算出每一条模糊条件语句决定的模糊关系 $R_i (i=1, 2, \dots, n)$ 之后，考虑到每条模糊条件语句间的“或”关系，可得描述整个系统的控制规则的总模糊关系 $R$ 为：

$$R = R_1 \cup R_2 \cup \dots \cup R_n = \bigcup_{i=1}^n R_i \quad (4-1)$$

模糊控制规则可通过总结模糊控制关系 $R$ 来实现。

### 3、输出量的去模糊化

有了模糊关系 $R_i (i=1, 2, \dots, n)$ ，则输出语言变量论域上的模糊子集 $U$ 记为：

$$U = \bigcup_{i=1}^n (E, EC) \cdot R_i \quad (4-2)$$

但此时模糊控制器的输出是一个模糊集，它包含控制量的各种信息。但被控对象仅能接受一个精确的控制量，应从中选择哪一个控制量施加到被控制对象中去，必须进行模糊判决（模糊决策），把模糊量转化为精确量。把模糊量转化为精确量的过程称为清晰化，又称为去模糊化，或者称为模糊判决。

模糊自动控制中常用的模糊判决方法有三种：

#### (1) 最大隶属法

这个方法是选取隶属度最大的论域元素为去模糊的结果。如对应的模糊判决的模糊子集 $U$ ，则取该模糊子集中隶属度最大的那个元素 $u_{\max}$ 作为执行量，即

$$u_u(u_{\max}) \geq u_u(u)。$$

这种方法的优点是简单易行，缺点是它概括的信息量很少，因为这种方法排除了其他隶属度较小的元素的影响和作用。

#### (2) 加权平均法

这种判决方法有以下两种形式：

普通加权平均法，其执行量 $u_{\max}$ 由下式决定：

$$u_i = \frac{\sum u(u_i)u_i}{\sum u(u_i)} \quad (4-3)$$

权系数加权平均法，其执行量 $u_{\max}$ 由下式决定：

$$u_{\max} = \frac{\sum k_i u_i}{\sum k_i} \quad (4-4)$$

其中，权系数 $K_i$ 的选择应根据实际情况来决定。加权系数的决定直接影响着系统的响应特性，对模糊控制系统来说，要改变系统的响应特性，选取和调整有关系数是关键。

### (3) 取中位数法

为了充分利用模糊子集中所有的信息量，可以求出把隶属度曲线与横坐标围成的面积平分成两部分的数作为去模糊的结果。最后，加到被控过程的控制量 $U$ 应是去模糊的结果 $u$ 与比例因子 $k_u$ 的乘积： $U = k_u * u$ 。

## 4、模糊控制查询表

经过模糊化、模糊推理和去模糊化，对于一组输入（ $e$ ,  $ec$ ）可以得到控制器的输出 $U$ 。

即得到模糊控制器的准确查询表。例如表4-1所示：

表4-1 模糊控制查询表

e U ec	-5	-4	-3	-2	-1	0	1	2	3	4	5
-6	6	6	6	6	6	5	5	4	4	0	0
-5	6	6	6	6	6	5	5	4	4	0	0
-4	6	6	6	6	5	4	4	4	4	0	0
-3	6	6	6	6	5	4	4	4	0	0	0
-2	4	4	4	4	4	4	2	0	0	0	0
-1	4	4	4	4	4	2	2	0	0	0	0
0	4	4	3	2	2	0	0	0	-2	-2	-4
1	2	2	0	0	0	0	-3	-3	-4	-4	-4
2	2	2	0	0	0	0	-2	-3	-4	-4	-4
3	0	0	0	0	-3	-4	-4	-5	-5	-6	-6
4	0	0	0	0	-4	-4	-4	-4	-6	-6	-6
5	0	0	0	0	-4	-4	-4	-5	-6	-6	-6
6	0	0	0	0	-4	-4	-5	-6	-6	-6	-6

该表被存储于计算机中，在线运行时根据 $e$ ,  $ec$ 的实测值，查表就可以得到控制输出 $U$ 。

综上所述，模糊控制器的设计步骤可以概括为：

- (1) 确定模糊控制器的输入变量和输出变量；
- (2) 确定输入输出的论域；

- (3) 总结模糊控制规则;
- (4) 选择推理算法;
- (5) 确立模糊化和逆模糊化的方法。

## 4.2 舵机模糊控制器的设计

在小车的实际运行中,我们先采取的是PD控制。在该控制算法下小车跑直线时,表现非常好,但是在进曲率半径较小的弯道时,往往会冲出去,这样的话就相当于出局了。我们发现这是和比例系数不够大有关的,因此我们采用了模糊控制算法,对不同的弯道曲率用不同的比例参数。

### 4.2.1 模糊控制算法设计

模糊指令是S12的特色指令,他们类似于高级语言指令,多数执行时间较长,而且执行过程中允许中断响应,中断返回后可以继续运行,采用他们可以简化模糊运算程序,缩短运行时间,而且不易出错。

如图4-4所示, S12指令的模糊指令在处理模糊运算时分为三个阶段:

- (1) 首先根据隶属度函数将过程变量变成模糊输入,这部分工作在模糊化接口完成,然后将结果放入存储器;
- (2) 推理运算部分根据规则库和模糊运算输入得到模糊输出,也放入存储器;
- (3) 解模糊部分通过隶属度函数及相关规则将模糊输出变成控制输出,送往执行机构。

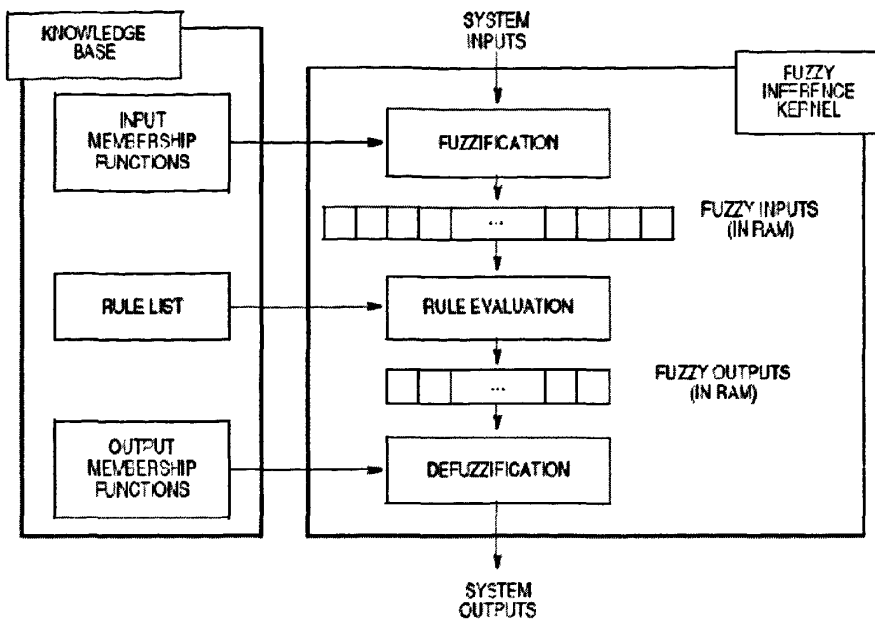


图4-4 S12模糊控制指令的执行流程

S12的模糊控制指令在实现时相应的用了MEM、REV、REVW和WAV四条指令，并为他们制定了必要的表达方式。下面将具体介绍这几条指令的用法<sup>[36]</sup>。

1、MEM指令

MEM指令是单个隶属度函数计算的核心指令，由他将过程变量变成模糊输入，该函数正常运行的前提是：

- (1) 建立一个隶属度函数；
- (2) 在内存建立描述该隶属度函数的数据结构；
- (3) 制定保存运算结果及模糊输入值的内存空间。

特别是在建立隶属度函数的数据结构时要遵照一定的语法规则，如图4-5所示：

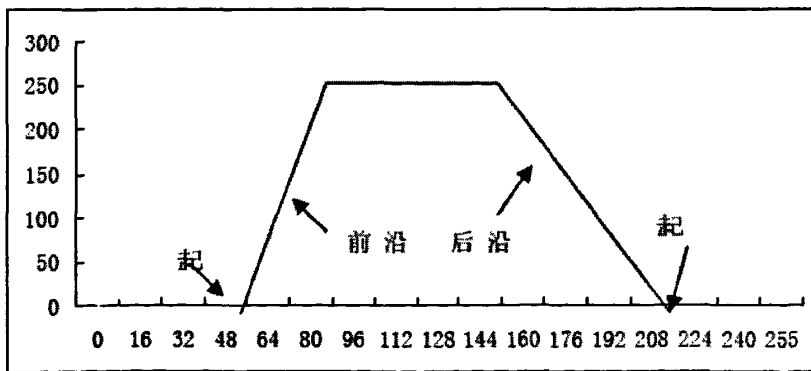


图4-5 S12的隶属度函数建立规则示例图

每个隶属度函数的数据结构必须用四个字节无符号数的数据结构来描述：即{起点的X坐标，终点的X坐标，前沿斜率，后沿斜率}，这里起点和终点的X坐标

均采用类似归一化的数值，即必须将输入的范围映射到\$00-\$FF中去，而前后沿斜率的符号固定不变，只需要给出具体数值，分别为（ $\$FF / (\text{饱和点X坐标} - \text{起点X坐标})$ ）和（ $\$FF / (\text{终点X坐标} - \text{饱和点X坐标})$ ）。当斜率为无穷大的时候，用\$00表示。

设定数据结构时采用如下的语法：

IINPUT\_MPFS:

FCB \$4 0 ; 起点X坐标  
 FCB \$D0 ; 终点X坐标  
 FCB \$0 8 ; 前沿斜率  
 FCB \$4 0 ; 后沿斜率

## 2、REV指令

REV指令是规则推理的核心指令，由他根据模糊输入确定模糊输出，但必须先按照约定为其设定必要的条件：

- (1) 按照需要提供若干个魔术模糊输入，输出关系的规则；
- (2) 在内存建立描述该规则的数据结构；
- (3) 制定运算结果及模糊输出值的内存空间，并事先清0；
- (4) 设置有关寄存器；

所谓模糊逻辑的规则就是一个模糊逻辑表达式：

IF前方是弯道AND小车当前的车速很高THEN车辆减速AND车辆预先偏转。

用模糊指令的语言表示：

RULE\_START

FCB I0,I1,I2.....,\$FE,O00,O01.....\$FE:

FCB I0,I1,I2.....,\$FE,O10,O11.....\$FE:

FCB I0,I1,I2.....,\$FE,O20,O21.....\$FE:

.....

FCB I0,I1,I2.....,\$FE,O00,O01.....\$FE:

FUZ\_INS

RMB MI0+MI1+...Min+MO0+MO1...Mon;

## 3、RE VW指令

RE VW为加权推理指令，与REV指令功能类似，但也存在着不同点，RE VW指令处理每个规则时，将前件表达式的运算结果乘以一个权值后，作为模糊后件，而REV指令直接将结果送给后件。在使用RE VW指令时需要额外加一个权值表：

RULE\_START

FCB I0, I1, I2....., \$FE, 000, 001.....\$FE:

```

FCB I0, I1, I2....., $FE, 010, 011.....$FE:
FCB I0, I1, I2....., $FE, 020, 021.....$FE:
.....
FCB I0, I1, I2....., $FE, 000, 001.....$FE:
FUZ_INS
RMB MIO+MI1+...Min+MO0+MO1...Mon;
FUZ_WT
FCB W0, W1, ..... Wn // 权值数据表

```

其中表内的权值均为0-1之间的小数，小数点位于最高位左边。

#### 4、WAV指令

WAV指令是解模糊阶段的核心指令，它与EDIV指令配合完成模糊输出到控制输出的变换。

常用的算法之前已有介绍，WAV指令使用的是其中的权系数加权平均法，该方法将模糊输出作为权值，对输出模糊子集对应的模糊单点集的值(Singleton)求加权平均，即采用下式确定控制输出：

$$output = \frac{\sum_{i=1}^n F_i S_i}{\sum_{i=1}^n F_i} \tag{4-5}$$

式中Si: 各个输出模糊自己对应的模糊单点集的值；

Fi: 输出对该子集的隶属度；

n : 输出模糊子集数。

下面具体介绍模糊算法和S12模糊控制指令的实现。

在计算曲率时我们取了三个有效点，如图4-6所示：

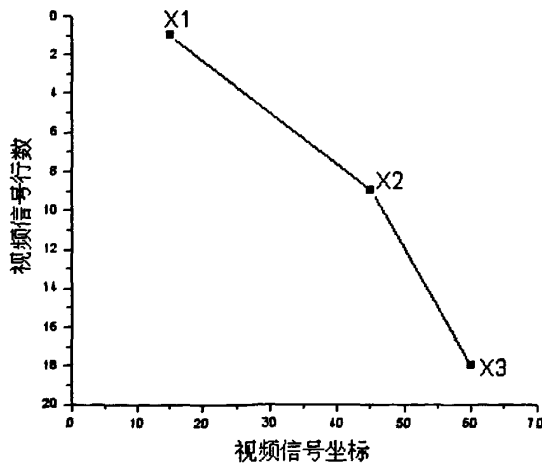


图4-6 曲率计算示意图



这三个有效点的参数都是在视界坐标下取得的,纵坐标是摄像头采集到的视图的行数,行与行之间的距离在实际作表中表示的距离是不变的,横坐标用来表示点在横轴方向上的位置,也是在视界坐标下确定的。实际计算曲率时,我们不用把所得坐标都转换到实际坐标上去算,因为在视界坐标下算出的曲率同样可以反映出赛道的弯曲程度。我们通过式6来计算曲率CURVE:

$$CURVE = |(X1 - X2) - (X2 - X3)| = |X1 - 2 * X2 + X3| \quad (4-6)$$

之所以不用标准的曲率计算公式一方面是他的计算量太大,另一方面我们不需要那么精确的曲率,上面定义的曲率公式已足以反映出赛道的弯曲程度。计算出来的曲率值恒为正,因为我们有专门的程序判别是左弯还是右弯,所以我们只需知道赛道的弯曲程度就可以了。将CURVE模糊化,模糊子集定义为:

CURVE={零, 极小, 小, 大, 非常大}

注:如果为零,表示曲率半径无穷大,该赛道为直道。

将CURVE的语言变量定义为:

Z = 零;

VS = 极小;

S = 小;

B = 大;

VB = 极大。

将CURVE量化为7个等级,分别为 0, 1, 2, 3, 4, 5, 6。那么CURVE的论域为: CURVE={0, 1, 2, 3, 4, 5, 6}。

根据调试小车在弯道上的表现给出CURVE量的隶属度表,如表4-2所示:

表4-2 CURVE的隶属度函数表

量化等级 语言变量	0	1	2	3	4	5	6
零	1	0.5	0	0	0	0	0
极小	0	1	0.5	0	0	0	0
小	0	0	1	0.5	0	0	0
大	0	0	0	0.5	1	0.5	0
极大	0	0	0	0	0	0.5	1

由上表我们得到CURVE的隶属度函数，如图4-7所示：

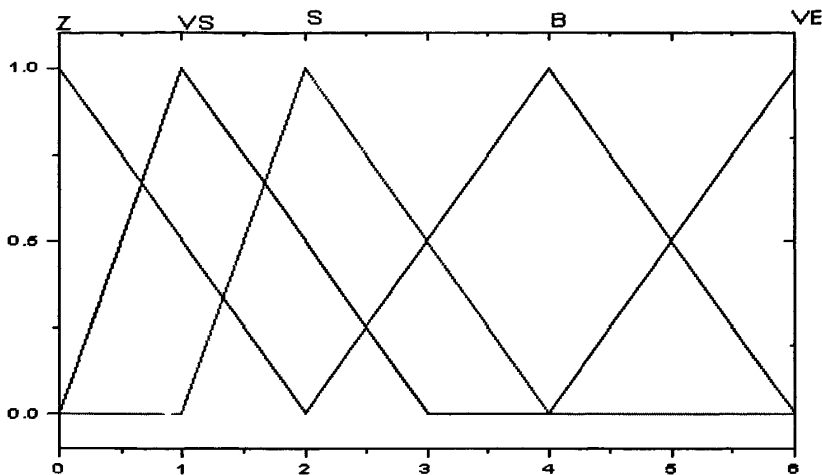


图 4-7 CURVE 隶属度函数

对于Kp值我们同样进行模糊化，模糊子集定义为：

$K_p = \{零, 极小, 小, 大\}$ 。

将Kp的语言变量定义为：

Z = 零；

VS = 极小；

S = 小；

B = 大。

将Kp量化为7个等级，分别为 0, 1, 2, 3, 4, 5, 6。那么Kp的论域为： $K_p = \{0, 1, 2, 3, 4, 5, 6\}$ 。同样根据调试的经验给出Kp量的隶属度表4-3：

表4-3 Kp量的隶属度表

量化等级 语言变量	0	1	2	3	4	5	6
零	1	0.5	0	0	0	0	0
极小	0	0.5	1	0.5	0	0	0
小	0	0	0	0.5	1	0.5	0
大	0	0	0	0	0	0.5	1

从而得到 $K_p$ 量的隶属度函数如图4-8所示:

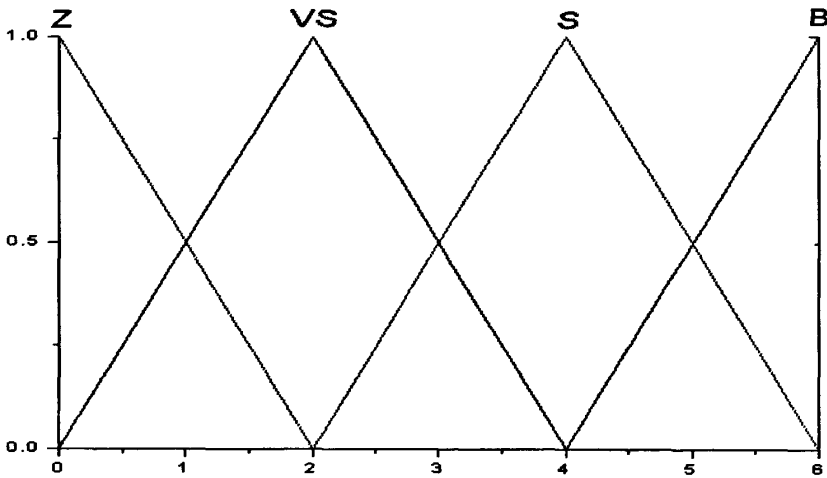


图4-8  $K_p$ 的隶属度函数

这部分的S12程序如下:

```
RAM: section
absentry fuzvar
fuzvar: ds.b 8      ; inputs      // CURVE语言变量进行量化
Z: equ 0           ; 零
VS: equ 1          ; 极小
S: equ 2           ; 小
B: equ 3           ; 大
VB: equ 4          ; 极大

output membership variables // 输出语言变量进行量化
absentry fuzout
fuzout: ds.b 4     ; outputs
Z: equ 0           ; 零
VS: equ 1          ; 极小
S: equ 2           ; 小
B: equ 3           ; 大

EEPROM: section
fuzzification      //建立CURVE的隶属度函数
s_tab: dc.b 0,80,0,3 ; 零
dc.b 0,120,12,3    ; 极小
dc.b 20,160,12,3   ; 小
dc.b 80,240,3,3    ; 大
```

dc.b 160,255,3,0 ; 极大

注：把0-6的等级放大40倍，该定义规则见S12指令部分。

然后建立模糊控制规则如表4-4所示：

表4-4 模糊控制规则表

CURVE	Z	VS	S	B	VB
Kp	Z	VS	S	B	B

上述模糊规则可以用模糊关系R来表示：

$$R = (Z * Z) \cup (VS * VS) \cup (S * S) \cup (B * B) \cup (VB * B) \quad (4-7)$$

S12程序表示为：

rules:

dc.b Z, \$FE,Z,\$FE

dc.b VS, \$FE,VS,\$FE

dc.b S, \$FE,S,\$FE

dc.b B, \$FE,B,\$FE

dc.b VB, \$FE,B,\$FE

最后需要用权系数加权平均法来得到输出。我们定义Kp各个量化因子对应的权重为：{Z,VS,S,B,VB}={0,1,2,3,4}。其S12指令为：

addsingleton:

dc.b 0, 1, 2, 3, 4 ;

然后对模糊控制指令的返回值Y进行处理，得到新的Kp值<sup>[37-42]</sup>。

#### 4.2.2 舵机模糊控制器仿真

由于电机的转向与输出的波成正比例的关系，所以可以近似的认为小车转向速度与电机的输出成正比例的关系。由于小车的转向动力学模型为<sup>[43,44]</sup>：

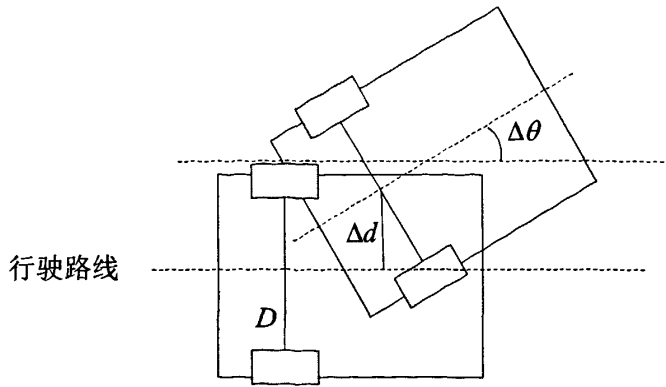


图 4-9 小车追踪路线示意图

由上图所示， $\Delta t$  为初始时刻和下一时刻之间的时间间隔； $\Delta d$  为自动导向车两个驱动轮的连线中点与路径中线垂直距离； $\Delta \theta$  为自动导向车两个驱动轮的连线中垂线与路径中心线之间的夹角； $v_l$  为自动导向车左驱动轮速度， $v_r$  为自动导向车右驱动轮速度。为自动导向车轮距中心速度。由图 4-9 可得：

$$\Delta \theta = (v_r - v_l) \times \Delta t / D, \quad \Delta \theta = 0.5(v_r + v_l) \times \Delta t \times \sin \theta$$

由于小车的行进的过程中其路径是连续的，故有：

$$\Delta \theta = (v_r - v_l) \times t / D, \quad \Delta d = 0.5(v_r + v_l) \times t \times \sin \theta$$

对时间  $t$  进行拉普拉斯变换可以得到：

$$\theta(s) = \frac{1}{Ds(v_r - v_l)}, \quad d(s) = \frac{1}{2s \times (v_r + v_l) \times \sin \theta}$$

假设电机的转速与轮速相等，两个轮的直径相等，电机的时常数  $T_m$ ，则转动速度与电枢电压的关系可表示为：

$$v_r = \frac{k}{1 + T_m} u_s, \quad v_l = \frac{k}{1 + T_m} u_s$$

由上式可知小车的传递函数为：

$$d(s) = \frac{T_m \times v_c}{2D_s^2 (T_m s + 1)} \times \Delta U(s)$$

由上式传递函数可知，该系统是一个三阶系统，易知该系统是一个不稳定的系统，我们可以采用多点信号采样，使用模糊控制方法使小车的路径偏差能得到更好的改善，输出能够很好的跟随输入。其中误差 CURVE 可参照前面的建立的隶属度函数，误差变化率  $\Delta CURVE$  的隶属度与误差同步，小车转动角度  $u$  的范围在  $(-30^\circ, 30^\circ)$ 。参照前面的模糊控制器建立步骤，可对误差 CURVE，误差变化率  $\Delta CURVE$ ，转角  $u$  各个变量建立隶属度函数和模糊控制规则，得到模糊控制的输

出转角  $u$ ，从而用于实时仿真，检测模糊前面的模糊控制器的建立是否合理<sup>[45]</sup>。

设仿真的采样时间为  $0.2s$ ， $v_c$  为  $1m/s$ ，输入周期为  $25s$ ，幅值为  $10$  的正弦波。为了仿真方便，我们采用三角隶属度函数。仿真图如下<sup>[46]</sup>：

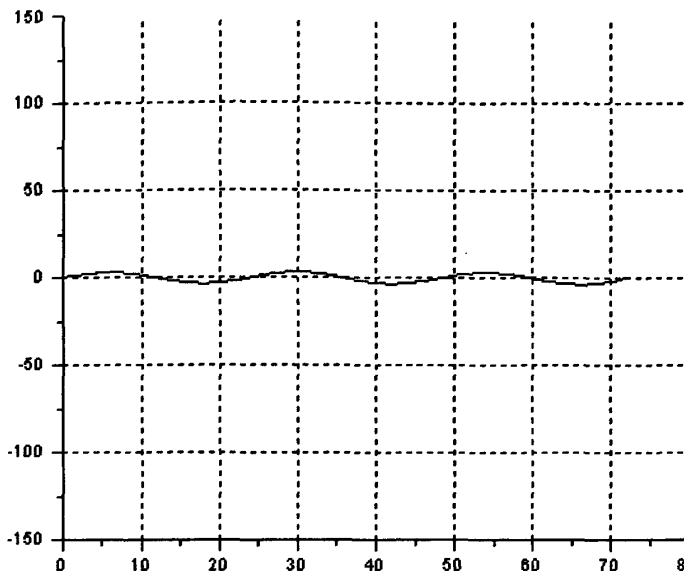


图 4-11 小车系统在模糊控制下角度的输出

在上图中能够看到舵机在模糊控制作用下小车角度输出响应图，可见使用模糊控制方法输出能够很好的跟随输入。由此可见，模糊控制能够很好的保证被控制对象的稳定性。更重要的是，说明这样建立的模糊输入隶属度函数和模糊规则的建立是合理的，能够实现多舵机的稳定快速响应的控制。下面将给出模糊控制舵机和 PD 舵机控制两种效果的实际效果比较。

#### 4.2.3 两种算法实验结果比较

以下是使用模糊控制算法前后小车的实际轨迹，如图4-11所示：

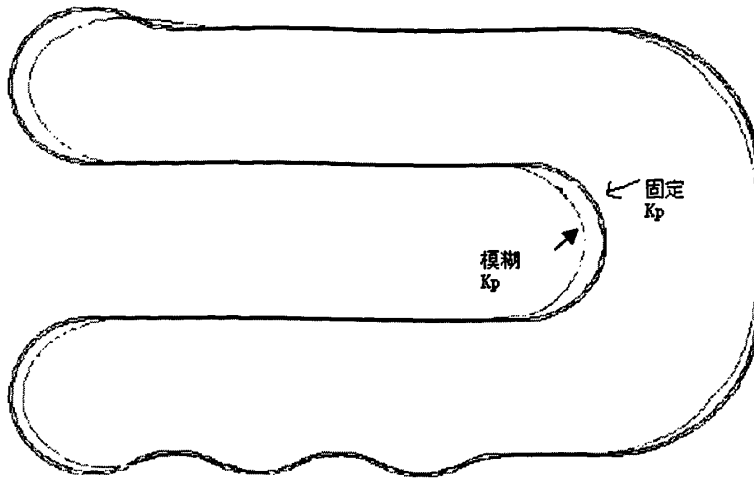


图4-11 使用模糊控制算法前后小车在赛道上的实际轨迹

其中的红线是赛车行驶轨迹。

注：这张图的绘制方法是这样的，先通过串口把误差值和速度通过无线串口发到上位机，然后通过计算机上描点画出小车实际的轨迹，有偏差但是与实际情况非常近似<sup>[47, 48]</sup>。

从表4-5中可以非常明显的看到使用模糊控制算法之后，小车虽然也有偏出赛道的情况存在，但是比用 $K_p$ 参数恒定的算法情况就好多了。

表4-5 参数对比

	小车跑完全程的时间	误差的样本总体的方差	速度的平均值
$K_p$ 固定	18	278.3003	1.84m/s
使用模糊控制算法	15	220.6543	2.208m/s

这主要是因为弯道上 $K_p$ 参数变大的缘故。 $K_p$ 变大时，一旦检测到偏差，系统就能更快的恢复到平衡位置。从表中可以看到小车偏差的方差实际减小了，因而小车走的实际距离也减小了，自然跑完赛道的时间也减小了。

### 4.3 基于模糊PID控制的记忆算法的研究

#### 4.3.1 记忆算法介绍

本节在对智能车控制中采用模糊PID控制算法基础上进行改进，将小车控制策略通过与记忆算法相结合进行了达到小车的更优化空空告知。主要从赛道记忆算法的实现前提、初圈记忆、数据分析与处理以及如何充分利用记忆得到的信息四个方面进行介绍。小车速试验表明，对于相对简单的比赛赛道，基于赛道记忆

控制算法的智能车可以取得较好的成绩，随着赛道的日趋复杂，赛道记忆算法也有很大的潜力。

本次智能车邀请赛规则中明确指出，智能车在赛道上连续跑两圈，并记其中最好圈成绩为比赛成绩。这个规则的确立使得赛道记忆算法成为了可行。如图4-12所示：

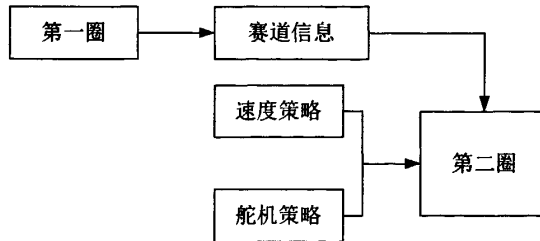


图 4-12 记忆算法流程图

赛道记忆算法在第一圈以最安全的速度缓慢驶过一圈，并将赛道信息保存下来，第二圈根据保存下来的信息进行车速和转角决策的相应最优化，从而在第二圈取得好成绩。无论智能车的传感器前瞻距离有多远，在跑圈时它都只能预测在一段有限距离内赛道的情况。而采用赛道记忆算法的智能车，在第二圈时已对整个赛道有了全面的认识，从而在相同条件下，将比不使用赛道记忆的智能车更具优势<sup>[49]</sup>。

当然，要想成功实现赛道记忆算法，必须有以下几大前提：

(1) 赛车在第一圈必须安全走完全程。在第一圈，赛车的最主要目的是采集赛道信息，使用赛道记忆算法的智能车，在第一圈一般都会采用较缓慢的匀速走完全程。

(2) 赛车必须能明确分辨出赛道起点线和十字交叉线的区别。

(3) 赛车必须能够有足够记录一圈赛道数据的内存空间。由于单片机芯片的RAM容量有限，算法研究者需要对算法进行优化处理，改变保存的格式以节省内存空间。

(4) 在跑第一圈时，赛车需即时地对当前赛道数据进行处理，避免第二圈时还在进行数据处理而对赛车的控制实时性带来不良的影响。

#### 4.3.2 记忆算法控制策略

采用赛道记忆算法的智能车，第一圈的主要目的就是将赛道的重要特征按特定的格式记录下来，作为第二圈的参考。一般而言，赛道是由一系列的直道和与之相切的不同曲率半径和弧度的弯道组成的。因此，初圈最终要记录的数据的横坐标一般为行驶的路程，而纵坐标则为赛道此点的曲率半径。对于横坐标，由于初圈一般匀速驶完，一般可以直接用时间来表示，也可以用车速传感器的当前脉



冲数来表示。而对于纵坐标,则无法第一时间取得当前点的曲率半径,它需要对采集到的原始数据进行相关处理后才能得出。在初圈中,我们能记录的只是当前的转角设定值,可用舵机占空比或者直接转换过来的角度值表示。我们以此为一个试验赛道建立了一个二维的数据结构,具体如下表格所示:

表4-6 道路记忆数据结构

x/0.05s	0	10	20	30	40	50	60	70	80	90
y(duty)	1150	1150	1150	1150	1150	1150	1150	1150	1150	1150
x	100	110	120	130	140	150	160	170	180	190
y	1160	1160	1162	1160	1155	1150	1150	1150	1142	1150
x	200	210	220	230	240	250	260	270	280	290
y	1158	1150	1142	1130	1170	1165	1168	1165	1152	1150
x	300	310	320	330	340	350	360	370	380	390
y	1148	1150	1160	1150	1145	1150	1130	1145	1125	1130
x	400	410	420	430	440	450	460	470	480	490
y	1150	1155	1150	1165	1120	1170	1120	1170	1120	1170
x	500	510	520	530	540	550	560	570	580	590
y	1120	1170	1135	1150	1165	1150	1150	1150	1150	1150

其中,舵机占空比 1150 时是处于中间位置,大于 1150 向左转,小于 1150 向右转。因此,通过这个数据表格能够较全面的反应整个赛道的路况:

0—100: 直道

100—170: 左大弯道

170—190: 右小弯道

190—210: 左小弯道

210—230: 右小急转弯

230—280: 左大急转弯

280—350: 小弯道(可近似直道)

350—400: 右大弯道

400—440: 左大弯道

440—510: S 弯道(处理后可当直道跑)

对于使用赛道记忆算法的智能车来说,第二圈才是最关键和吸引人的部分。在第二圈,如何充分利用根据第一圈收集的赛道信息,跑出最优的成绩,这个是仁者见仁的事情,这里只从车速策略和转角策略简单地介绍一些自己的思路<sup>[50]</sup>。

## 1、车速策略

由于车本身的某些局限性，如舵机响应时间、赛道摩擦系数、轮胎抓地力、传感器布局等因素的影响，对于某种特定弯道，车是有它所谓的极限速度存在。如果我们利用初圈得到弯道的曲率半径后，在过各弯道时将车速调至它相应的极限速度，那么理论上这就是它所能跑出的最优成绩了。对于直道，特别是长直道，更能发挥赛道记忆算法的优势。例如，在初圈记忆后得到某段直道为 3000 个脉冲点长(这里脉冲点直接反应时间)，那么在决策中就可在前 2700 个脉冲时启动电机用最大占空比加速，超过 2700 脉冲时直接减速，将速度调在下一个弯道的极限速度进行过弯。由于事先已经知道直道的长度，因此调速策略更能有的放矢。

## 2、转角策略

如果赛车采用的是一字型传感器布局的话，当你将中心点设偏的话，那么赛车也会出现一定的左右倾向性，从而给在弯道时“抄近路”奠定了基础。在初圈记忆后，赛车跑第二圈时可以直接得出下个弯道是左拐还是右拐。这样赛车就可以按比规定更小的曲率半径通过赛道，从而减少了经过路径，也节约了时间。同样，一般赛车在直道上高速行驶时经常会出现来回振荡的现象。如果经过初圈记忆，赛车判断自己正处于直道时，可以进行相应的措施，很好地解决这个问题。其他的就直接采用与模糊控制舵机的方法实现对转角的控制。

## 4.4 本章小结

本章在第三章的基础上进一步对小车的控制进行了优化，主要体现在舵机的控制上采用模糊 PID，并给出了两种舵机控制算法的效果比较，较好的解决了前面所出现的过冲等问题。并在最后提出了将记忆算法与模糊 PID 相结合实现对小车的控制的方法。这个算法暂时还未能达到非常理想的效果，主要是在对道路完整而精确的记忆，在此算是抛砖引玉，也将是后续努力的研究方向。

## 第五章 总结与展望

### 5.1 总结

经过长达3个月的细致紧张准备，我们终于完成了计划中所需完成的内容。从车模安装，机械结构调整，到电路板制作，传感器方案制定，最终经过长时间的算法调试改进，赛道测试，后期改进研究，进一步完善了我们的智能小车系统。回顾这次智能小车系统设计研究，从以下几个方面进行一下总结。

在整个车模开发过程中，并不是一帆风顺的，期间遇到了各种软硬件问题。在此，对一些重点问题进行一下说明。

#### 1、遇到的问题与解决办法

##### (1) 舵机响应延时问题。

大赛组委会所提供的转向舵机，在实际赛道测试时，遇到了严重的响应延时问题。由于赛车高速运行，控制算法频率为50Hz，因此每20ms，就向舵机发送一次控制命令。而参考舵机说明书，经过实际测试，舵机响应延时大于20ms。因此，无法及时响应控制命令。首先，加大舵机工作电压，增加舵机抗负载能力。其次，通过算法进行弥补，发挥摄像头探测距离远的特点，有足够的时间进行舵机响应。实际赛道测试，通过调整算法参数，可以很好地弥补响应延时。

##### (2) 赛车弯道速度控制问题。

实际赛道测试发现，如果不控制赛道在弯道速度，会出现转向不足的问题。除了调整赛车机械结构，只能通过弯道减速的办法，实现平稳顺滑过弯。由于赛车没有安装刹车机构，车辆减速只能依靠永磁电机特性和运动负载减速，因此赛车减速特性较差，需要预留足够长的减速距离，完成入弯减速。因此对摄像头提前识别弯道距离提出更高的要求。

##### (3) 赛车实际速度获取。

除了寻线传感器获取到的车身偏移量，赛车运行速度是车身运动控制算法中重要参数。稳定可靠获取当前速度，对于赛车调试来说，具有相当重要的意义。但是，比较成熟的速度传感器，有的体积过大，不方便安装，有的极大增加赛车负载，影响赛车性能。因此我们并没有购买安装光电码盘等现成的设备作为我们的速度传感器方案，而是根据光电码盘原理，自己制作了一个简易的光电传感器，在控制精度上不及光电码盘，对于此次比赛，不要求很高的控制精度，因此这样的简易机构完全满足我们的使用要求。

## 2、设计中的亮点

### (1) 自主设计制作的速度传感器。

根据光电编码器原理，利用光电管黑白线识别功能，配合S12 强大的ECT模块，自主设计了一个简易的速度传感器。精度和稳定性达到使用要求。

### (2) 记忆算法的研究。

这种算法虽然最后未能用来参加比赛，但试验效果尚可。如果在道路的数据记忆方面能够得到进一步的改善，一定能起到非常好的效果。

## 5.2 展望

展望未来，如果有机会参加今后的比赛，在软硬件方面可以进行更多的尝试。因为实验和摸索是提高成绩的根本，只有经过反复实验才能得到可靠的算法。具体来说，还可以在以下几方面继续探索：

### (1) 小车机械性能方面。

现在速度不能提得太高的一个主要原因是舵机性能不好，由于比赛规定舵机不允许改动，我们可以考虑外加一个制动装置在入弯时进行减速。这样直道速度可以提高，平均速度更快。但如何选择合适的制动装置并对其进行控制，还需要进一步深入。

### (2) 硬件传感器方面。

目前，主要试过的路面检测传感器有光电管和摄像头，经后考虑尝试其它传感器。

### (3) 软件控制算法方面。

首先，可以继续模糊控制算法和记忆算法上进行深入。其次，还可以考虑仿真平台的搭建，这对算法调试来说是很有意义的，毕竟我们不能每次为了测试一个新的路况都搭建一次赛道。

## 参考文献

- [1] 孙增析,张再兴,邓志东.智能控制理论与技术.北京:清华大学出版社,2004.
- [2] Saridis G.N..Towerd the Realization of Intelligent Controls.IEEE Proc.,1979,67(8).
- [3] 孙增析,张再兴,邓志东.智能控制理论与技术北京:清华大学出版社,2004.
- [4] 张幽彤,陈宝江,翟涌,等,汽车电子技术原理及应用,北京:北京理工大学出版社,2006,2.
- [5] 吴斌华,黄卫华,程磊等.基于路径识别的智能车系统设计.电子技术应用,2007年第3期
- [6] 孙立宁,彭龙刚,祁亮自动引导车轨迹偏差的智能控制哈尔滨工业大学学报2003,(2):62-65.
- [7] 卢韶芳,刘大维.自主式移动机器人导航研究现状及其相关技术.农业机械学报.2002,(2):112-116.
- [8] 史恩秀等.全方位 AGV 复合转向方法的运动分析与仿真.西安理工大学学报.2001,(20):126-130.
- [9] 王华坤.基于扰动的自主移动机器人跟踪控制与驱动策略的研究.南京理工大学硕士学位论文.2004年.
- [10] 张进.AGV 视觉导引控制问题研究. 硕士论文. 上海交通大学 2004. 2. 19.
- [11] 张燕羽.基于微机模糊控制技术的 AGV 导向系统设计.[硕士学位论文].上海交通大学,1996.
- [12] MC9S12DP256B Device User Guide V02.149S12DP256BDGV21D Motorola Inc.
- [13] 杨国田,白焰.摩托罗拉 68HC12 系列微控制器原理,应用与开发关系.中国电力出版社.2003.9.
- [14] 熊慧,尤一鸣,MC68 单片机入门与实践,北京:北京航空航天大学出版社,2006,9.
- [15] Freescale Semiconductor, Inc. MC9S12DG128 Device User Guide.
- [16] Background Debug Module(BDM)V4 S 12BDMV4/D Motorola Inc.
- [17] 孙仲康.数字图像处理及其应用.国防工业出版社.1985.
- [18] Milan Sonka, Vaclav Hlavac, Roger Boyle. Image Processing, Analysis, Vision, Second Edition. USA: Thomson Asia Pte Led. 2002, January.
- [19] 张晓飞,袁祥辉.基于 DSP 成像系统的视频图像采集部分的实现.压电与声光,第24卷,第3期.2002年6月.

- [20] National Semiconductor. LM1881 Video Sync Separator General Description. April 2001.
- [21] C. K. Lee and W. H. Pang. A Brushless DC Motor Speed Control System Using Fuzzy Rules Power Electronics and Variable-Speed Drives, 26—28 October 1994.
- [22] 王荣本, 游峰, 崔高健, 郭列. 基于计算机视觉高速智能车辆的道路识别. 计算机工程与应用 2004. 26.
- [23] 万永伦, 丁杰雄. 一种机器人寻线控制系统. 电子科技大学学报, 第 32 卷, 第 1 期. 2003 年 2 月.
- [24] 林欣凡等. 清华三角洲队技术报告.
- [25] MILICR. STOJIC. Design of the Microprocessor-Based Digital System for DC Motor Speed Control. IEEE, Transactions on Control Systems Technology, Vol, IE-31, No3, August 1984.
- [26] (美) Todd D. Morton 著 严隽永译. 嵌入式微控制器. 机械工业出版社, 2005.
- [27] PWM\_8B8C Block User Guide V01.17 S12PWM8B8CV1/D Motorola Inc.
- [28] 关军等. 上海交通大学 CyberSmart 技术报告.
- [29] 陶永华. 新型 PID 控制及其应用. 机械工业出版社, 2002.
- [30] 王朝盛. 基于 16 位单片机 MC9S12DG128B 智能车系统的设计. [硕士学位论文]. 天津工业大学.
- [31] 张红涛, 闰林, 徐晓辉等. 基于单片机 PID 算法的无刷直流电机控制系统的研究. 现代电子技术, 2007 年第 10 期: 52-54.
- [32] 马磊, 基于模糊 PI 控制的推进电机调速研究. [硕士学位论文]. 武汉: 华中科技大学, 2006.
- [33] 魏巍. MATLAB 控制工程工具箱技术手册. 国防工业出版社, 2005.
- [34] 薛定宇. 控制系统计算机辅助设计语言及 MATLAB 应用. 北京: 清华大学出版社, 1996 年.
- [35] 诸静等. 模糊控制原理与应用. 机械工业出版社. 1995.
- [36] 邵贝贝. 单片机嵌入式应用的在线开发方法. 清华大学出版社, 2007.
- [37] T. W. Vaneck. Fuzzy Controller for a Autonomous Boat. IEEE Control System Magazine, Vol. 17, No. 2, April 1997.
- [38] C. W. Tao, Jin-ShiuhTau. Flexible Complexity Reduced PID-Like Fuzzy Controllers. IEEE Transactions on Systems, MAN, and, Cybernetics-Part B: Cybernetics Vol.30, No. 4, August 2000.
- [39] A. Visioli. Tuning of PID Controllers with Fuzzy Logic. IEE Proc

- Control Theory Appl, Vol. 148, No 1, January 2001.
- [40] Kiam Heong Ang, Gregory Chong. PID Control System Analysis, Design, and Technology. IEEE Transactions on Control Systems Technology, Vol 13, NO 4, July 2005.
- [41] Mao-Fu Lai, Chen Chang. Fuzzy Logic in The Phase-Locked Loop DC Motor Speed Control System. IEEE Catalog Number: 97TH8280.
- [42] R. Ferreiro Garcia, F. J. PCrez Castelo, Fuzzy Adaptive PID Controller Using Frequency Techniques IEEE 1996.
- [43] 程柏林. 基于模糊 PID 控制的智能小车控制器的研制. [硕士学位论文]. 合肥工业大学.
- [44] 陈宋, 李立国, 黄开胜. 智能模型车底盘浅析. 电子产品世界, 2006. 11.
- [45] 谢守勇. 基于模糊控制的小车运动仿真研究. [硕士学位论文]. 西南农业大学.
- [46] 刘金馄等. 先进控制及 MATLAB 仿真. 北京: 电子工业出版社, 2005.
- [47] 周斌, 蒋荻南, 黄开胜. 基于虚拟仪器技术的智能车仿真系统. 电子产品世界, 2006. 3.
- [48] 楚现知, 吴吉祥, 李锦忠. 基于 LabVIEW 的监控界面设计与单片机的串行通信. 工业控制计算机, 2005 年 18 卷第 7 期.
- [49] 周斌, 刘旺等. 智能车赛道记忆算法的研究. 电子产品世界. 2006 年 8 期.
- [50] 卓晴, 黄开胜, 邵贝贝. 学做智能车——挑战“飞思卡尔”杯. 北京航空航天大学, 2007.

## 致 谢

在完成本论文之际,我首先衷心感谢我的导师鲁五一教授,感谢他在我攻读硕士研究生学位期间对我学业上的悉心培养和不倦教诲,以及生活上无微不至的关怀与照顾。鲁五一老师渊博的知识功底,深厚的专业造诣,丰富的工程经验,让我在几年的学习中受益匪浅。鲁老师乐观的人生态度、积极的奋斗精神、通达的处世襟怀使我受益匪浅,让我在今后的工作和学习中有了更多的借鉴和参照。借此机会,向鲁五一导师致以最崇高的敬意和最真挚的谢意。

同时感谢熊红云老师和王莉老师,两位老师严谨的治学之风、积极探索的奋斗精神、高尚的师德和高境界的个人修为都将是我以后学习、工作的参照与指导。王老师不论是生活上还是学业上,都给了我最深的关怀和指导,并给予我无私的帮助。

感谢张宏亮、李长德师兄以及实验室所有的兄弟姐妹对我的帮助、关心和支持。最后,感谢我的家人、朋友、同学以及所有关心、支持、理解和帮助过我的人。

许寰

2009年4月



## 攻读硕士学位期间科研及论文完成情况

### 一. 参与科研项目情况:

1. 基于红外传感器的智能小车设计。负责硬件的设计与改良、程序和远程通讯调试等。(2007.4-2007.7)
2. 制氧控制系统设计。负责整个系统的软硬件设计和实验调试。(2007.10-2007.12)
3. 基于 CMOS 摄像头的智能小车设计。负责整个系统方案的选择、软硬件设计以及实验调试。(2008.4-2007.5)

### 二. 发表论文情况:

- [1]许寰, 鲁五一. 摄像头路径识别和小车控制策略. 计算机工程与科学, 2009, 5.

作者: [许寰](#)  
学位授予单位: [中南大学](#)

## 相似文献(2条)

### 1. 学位论文 [于虹](#) 基于16位单片机MC9S12DG128智能模型车系统开发研究 2009

随着汽车的普及,智能汽车及其相关技术的研究已经成为热门。本文以全国大学生智能汽车竞赛为背景,设计开发一辆具有路径检测功能,能实现自主线路识别的智能车。本文采用模块化的设计方法,完成了智能车系统的硬件设计。通过系统分析,将整个系统分为7大模块,包括核心控制模块、电源模块、电机控制模块、舵机控制模块、路径检测模块、车速检测模块及调试模块。本文使用MC9S12DG128开发板作为系统的核心控制模块;路径识别模块采用10对红外光电传感器,一字型等间距布局;本文采用MC33886两片并联驱动电机,增加了驱动能力;车速检测模块采用光电旋转编码器,既满足驱动电机控制的精度要求,又方便传感器的安装。为了系统调试方便,设计了RS-232通信模块及LCD显示模块,并制定了相应的通信协议。本文设计了二层结构的控制系统,底层系统包括“转向控制器”和“车速控制器”。本文在路径识别采用了软件细分算法及基于传感器离散布局的连续路径识别算法,克服了红外光电传感器离散布局所采集到的赛道信息有限的缺点,提高了路径识别的精度;设计了时间最优意义下的全加速、紧急制动和闭环控制等多种模式中平稳切换的“多模式车速控制器”;由于模型车在封闭赛道上连续行驶两圈,在控制策略上采用了基于脉冲点的道路记忆算法,明显的提高了第二圈的成绩;本文给出了车速与转向协调控制规则及模型车调试规则,并经过多次实验,验证了该规则的有效性。

### 2. 学位论文 [胡杰](#) 基于16位单片机MC9S12DG128智能模型车系统开发研究 2008

我国于2007年举办了第二届全国大学生“飞思卡尔杯”智能汽车竞赛,这是以快速发展的汽车电子为背景,涵盖了控制、模式识别、传感技术、电子、电气、计算机、机械等多个学科交叉的科技创新比赛。根据比赛规则要求,设计并制作了智能模型车控制系统,整个系统功能完善、运行稳定、可靠性高。模型车控制系统主要由核心控制模块、电源管理模块、路径识别及记忆模块、舵机控制模块、后轮驱动电机控制模块、车速检测模块、电池电压监测模块及无线通信模块组成。系统核心控制模块采用了freescale16位单片机MC9S12DG128。该系统是在CodeWarrior集成开发环境的基础上设计完成的,采用C语言和汇编语言混合编程。为了深入的了解各个关键系统及器件的特性,在系统设计之前,对其进行了相关的标定试验,为后续的硬件设计、软件设计及控制算法和控制策略的制定和选择奠定基础;系统在路径识别模块采用了软件细分算法及基于传感器离散布局的连续路径识别算法,克服了由于光电传感器离散布局所采集到赛道信息有限的缺点,提高了路径识别的精度;舵机控制模块控制采用了增量式数字PID控制算法,该控制算法技术成熟,结构简单,参数容易调整,不需要具体的数学模型,在工业控制中得以广泛应用,取得了理想的控制效果;后轮驱动电机控制模块采用了模糊自适应PID控制算法,充分的利用了MC9S12DG128内部提供的模糊推理机;车速检测模块设计了基于光电编码器设计思想的虚拟编码器,既满足驱动电机控制的精度要求,又方便传感器的安装;针对比赛规则的要求,在控制策略上采用了基于脉冲点的道路记忆算法,明显的提高了第二圈的成绩;为了系统的调试方便,设计了基于PTR2000模块的无线通信模块,并制定了相应的通信协议;为了提高系统的开发效率及更为深入的研究模型车系统,基于Visual Basic语言开发智能模型车系统开发平台。通过对智能模型车系统设计、开发及研究,取得了一定的成果,但仍有不完善的地方,有待进一步深入研究。

本文链接: [http://d.g.wanfangdata.com.cn/Thesis\\_Y1536431.aspx](http://d.g.wanfangdata.com.cn/Thesis_Y1536431.aspx)

下载时间: 2010年4月7日