

学 号：15507108

天津商业大学宝德学院毕业设计（论文）

ARM 智能小车控制系统 设计与实现

**Design and Implementation of ARM Intelligent
Vehicle Control System**

系： 计算机与信息技术

专业： 计算机科学与技术

班级： 1501

学生姓名： 相佩豪

指导教师： 董晶 讲师

2019 年 5 月

目 录

摘要.....	I
Abstract.....	II
1 导言.....	1
1.1 课题的研究背景及意义.....	1
1.2 智能小车控制系统的国内外发展现状.....	1
1.3 课题的目的和主要工作.....	3
1.4 论文的组织结构.....	3
2 系统需求分析和规格说明.....	4
2.1 系统总体设计需求.....	4
2.2 系统详细设计要求.....	4
2.3 规格说明.....	4
3 系统总体设计.....	5
3.1 系统结构设计.....	5
3.2 系统软件结构设计.....	5
3.3 系统硬件结构设计.....	6
4 系统详细设计.....	7
4.1 硬件构件设计.....	7
4.2 软件模块设计.....	12
5 系统集成与测试.....	28
5.1 系统实现.....	28
5.2 系统测试.....	30
5.3 系统展示.....	30
6 总结.....	35
参考文献.....	36
致谢.....	38

摘 要

传统的汽车主要是人们自己进行驾驶，但是驾驶员可能会受到未知因素的干扰，因此存在一定的安全隐患。而随着人工智能的不断发展，在智能科学与大数据技术的支持下，智能化汽车已成为一种趋势。而智能车辆的设计和使用不仅可以帮助人们节省资源，还可以解决人们无法做到的工作。本文结合 ARM 芯片，利用嵌入式技术设计并实现了智能小车控制系统，旨在不断推动汽车智能化的发展。

基于增量式 PID 控制算法和图论算法，设计并实现 ARM 智能小车控制系统。以 ARM Cortex-M3 的微控制器 LM3S615 为核心控制芯片，通过一体化 1024 线编码器输出小车的速度，并将速度和方向参数反馈给直流电机和陀螺仪，利用一体化红外传感器 IRM8601S 探测并反馈墙壁的信息，结合等高图实现 ARM 智能小车自动行驶、自动避障和自动搜索道路，选择最优路径等功能。

关键词：ARM 嵌入式；增量式 PID 控制；智能小车；图论；闭环控制

Abstract

With the continuous development of the automotive industry, with the support of science and technology, automotive intelligence has become a trend. Traditional automobile driving is mainly manual operation, and drivers are sometimes disturbed by other factors, so there are certain safety risks. The design and use of intelligent vehicles can not only help people save resources, but also solve the work that people cannot complete. This paper combines ARM embedded chip, designs and implements intelligent vehicle control system, aiming at continuously promoting the development of automobile intelligence.

Based on the incremental PID control algorithm and graph theory algorithm, the ARM intelligent car control system is designed and implemented. With ARM Cortex-M3 microcontroller LM3S615 as the core control chip, the speed of the car is output by an integrated 1024-line encoder, and the speed and direction parameters are fed back to the DC motor and gyroscope. The wall information is detected and fed back by the integrated infrared sensor IRM8601S, and the contour map is used to realize the automatic driving, obstacle avoidance and road search of the ARM intelligent car. The best choice is the best. Optimal path and other functions.

Key Words: ARM Embedded Incremental PID Control Intelligent Vehicle
Graph Theory Closed-loop Control

1 引言

近年来，人工智能技术的出现，让人类生活发生了巨大的改变。改变了传统的机械制造、电子芯片、传感器、交通系统的发展，使得智能化技术不断发展。通过设计与实现 ARM 智能小车控制系统，可以解决传统的人工驾驶出现的诸多问题，将自动搜索道路、选择最优路径、道路途中自动避障等功能得以实现。把人工驾驶转变为自动无人驾驶，通过 ARM 智能小车控制系统，实现智能化^[1]。

1.1 课题的研究背景及意义

当今社会，能源、环境、交通等问题不断升级，让人们意识到可持续发展的重要性，尤其是对于智能化车辆的发展，造成了极大地限制。只有通过智能化可持续发展的技术解决难题，而 ARM 智能小车控制系统，将是智能网联汽车发展的第一步，通过设计与实现 ARM 智能小车控制系统，将解决人工驾驶带来的问题，促进无人驾驶技术的发展，提高国家智能化、互联化水平。

1.2 智能小车控制系统的国内外发展现状

国外智能车辆控制系统的起步相对较早，相关性研究始于上世纪 50 年代，它的发展历程大致可分为三个阶段：

第一阶段：1954 年，第一辆带有自主引导的汽车于美国问世，它是智能车辆系统研究的里程碑，它代表着车辆将进行智能化发展，这个系统是一个拖车式的用于运货的平台。尽管车辆只能沿着固定路线运行，但是，它可以实现无人驾驶，这一优势也是车辆智能化的重要特征。

第二阶段：在这一阶段，智能车辆控制系统进入了新时代，各个发达国家都开始了对它的研究。1986 年，普罗米修斯在欧洲开启；1995 年在美洲，为了促进智能车辆控制系统的可行性研究，加快智能车辆控制技术的实用化步伐，美国创立了国家自动高速公路系统联盟；1996 年在亚洲，日本把重心放在自动车辆导航的方法研究上。

第三阶段：九十年代到现在，智能车辆控制系统的研究更加深入，不断向系统化、大规模的方向发展。其中，美国卡耐基-梅陇大学在这期间接连完成 Navlab 系列十部自主车辆的研制。进入 21 世纪，为促进智能车辆控制系统的研发，在德国举办了欧洲陆地机器人比赛（European Land Robot Trial），最终的赢家是德国队的智能汽

车，该智能车具有整体反馈机制，可以对检测到的视野进行成像。在该竞赛中，智能车可以将图像传输到内部计算机系统中，以控制车辆并识别行人和树木。而美国也在积极开发智能车辆控制系统，其国防部主管研究项目局（DARPA）举办机器人挑战比赛来促进智能车辆控制技术的发展与创新。

目前，国外智能车辆控制技术的发展保持在第三阶段，该阶段的研究代表了当前智能车辆控制系统的主要发展方向。智能车辆控制系统在国内的研究也在进行，整体性的研究尚未产生巨大成果，但是很多针对性的研究领域取得了不少成果，并且随着我国经济发展、人才培养，在该领域的许多方面投入了时间与精力，并取得了一系列成果，主要有：

（1）1992 年，中国科学院大学通过设想，将计算机放入到汽车中，进行试验编程控制系统。将计算机与相关传感器相连接，测出需要控制的数据，采用早期液体压力原理来控制系统。将完成后的控制系统安装在汽车的计算机中进行测试，成功研制出我国第一辆自主控制型汽车，该车拥有人工驾驶与自动驾驶相互切换功能。

（2）2003 年，我国单项研究追赶上世界的研究步伐，尤其是在驾驶时速的控制上，达到国际标准。其事例就是中国科学院大学同一汽集团成功研制出自动化汽车控制系统，可以控制车辆在道路上行驶的速度，范围是 130 公里/小时~170 公里/小时，具备安全可靠的控制保障。这项科研成果为未来我国的智能车辆控制系统的发展具有划时代的意义。

（3）2006 年，吉林大学交通学院智能车辆课题组研制出一种微型低速区域交通智能车 Cyber Car。这款智能车采用单目视觉导航，在当时属世界首创。该车辆具有无噪声、无污染、使用方便和安全可靠等特点。控制中心可对多台车辆进行集中的调度和管理。

（4）2014 年，由清华大学自动化系牵头，北京交通大学、同济大学、北京航空航天大学、东南大学、武汉大学、武汉理工、长安汽车等 10 家单位参与合作的国家 863 计划主题项目“智能车路协同关键技术研究”研讨会在河北清华发展研究院举行。对提升我国交通系统智能化水平具有积极的促进作用^[2]。

（5）2018 年，在智能车辆控制系统领域中，百度公司率先开展业务，大力发展智能车辆控制系统，与多个企业签署战略合作协议，共同研究控制系统程序与无人驾驶汽车的研发。

1. 3 课题的目的和主要工作

此次，本文是要设计出一个性能优越的智能小车控制系统，实现自动搜索道路、选择最优路径、道路途中自动避障等功能。在迷宫复杂环境下，可以改善灵敏度，实现自动化、智能化。这个系统是一个集自动感知、自主引导、自动驾驶等功能于一体的综合性系统。它将是未来实现无人驾驶汽车上路的一个重要过程，可以帮助目前形势下智能网联汽车发展的需求，进行模拟的开发与试验，达到具体环境下的需求。可以更好地解决能源短缺、环境污染、交通拥堵、事故频发等问题，在道路上实现自动驾驶，更好的方便人们的出行与安全。同时，在设计的过程中，也可以将大学期间学习的计算机科学理论知识与具体实践技术相结合，提高自身分析与解决问题的能力。

1. 4 论文的组织结构

本文结构主要分为以下六章：

第一章主要阐述课题的研究背景、意义及发展现状，并提出本课题的研究内容。

第二章主要阐述系统需求和规格说明，分析系统的主要要求。

第三章主要阐述系统总体设计，分别为系统结构设计、软件结构设计和硬件结构设计。

第四章主要阐述系统详细设计，主要包括硬件的构件设计和软件的模块设计。

第五章主要阐述系统的集成与测试，展示实物图，对功能进行具体测试。

第六章总结全文，对 ARM 智能小车控制系统的研究和发展做出展望。

2 系统需求分析和规格说明

本章是对系统提出设计需求，针对具体的需求提炼出详细的设计要求，并撰写出详尽的规格说明。

2.1 系统总体设计需求

ARM 智能小车控制系统总体设计的需求如下：

- (1) 在平面迷宫中，控制 ARM 智能小车自动行驶；
- (2) 在平面迷宫中，控制 ARM 智能小车自动避障；
- (3) 在平面迷宫中，控制 ARM 智能小车自动搜索道路，选择最优路径。

2.2 系统详细设计要求

经过对总体设计需求分析，对于 ARM 智能小车的详细设计要求如下：

- (1) 通过直流电机控制速度；
- (2) 通过红外收发器检测墙壁信息；
- (3) 通过陀螺仪控制转向角度；
- (4) 通过外设存储器存储墙壁信息；
- (5) 稳定的电源系统；
- (6) 实现最优路径的选择。

2.3 规格说明

通过对系统详细设计要求分析，整理出需求规格说明，如表 1 所示。

表 1 需求规格说明

项目	说明
名称	ARM 智能小车控制系统
目的	实现智能小车在迷宫中自主驾驶
输入	一个电源开关，两个操作按钮（2 个数字量），红外接收信息输入
输出	红外发射信号输出，速度大小 v ，转向角度 θ
功能	控制智能小车自动行驶、自动避障和选择最优路径
性能	最大速度 5m/s，实时避开障碍物，平面迷宫下工作
功耗	7.4v/350mAH 可充电锂电池
尺寸	体长 96mm，宽 84mm
重量	不多于 300g

3 系统总体设计

本章是对 ARM 智能小车控制系统进行总体设计，分为系统结构设计、软件结构设计和硬件结构设计。从这三方面阐述控制系统的具体结构情况，将软硬一体化，进行协同设计，共同完成基本功能，在设计同时考虑非功能情况，加强控制系统稳定。

3.1 系统结构设计

针对系统的需求，该系统需要软硬件相互结合。在功能上完成自动行驶需求，需要直流电机驱动，让小车动起来；同时要实现智能化，需要加入控制程序，控制电机速度；并且在考虑非功能因素情况下，还需要加入稳定的电源系统；完成自动避障需求，需要给智能小车加上“眼睛”，通过红外收发器，可以实现检测功能，发现并反馈障碍物信息；通过陀螺仪控制程序，控制智能小车转向的角度；完成最优路径选择，需要墙壁数据，存储到存储器中，给智能小车进行自主判断。通过以上分析，得到系统结构图，如图 1 所示。

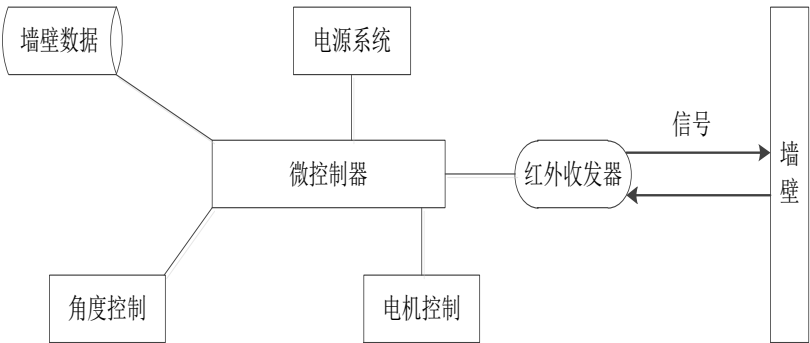


图 1 系统结构图

3.2 系统软件结构设计

红外收发器采集到红外线信号后，传递到输入/输出控制系统，并将检测的墙壁信息写入存储器中。系统接收到路径信息，判断当前位置情况，通过这些数据进行理论演算，计算出智能小车的理论速度及转向参数，当接收到墙壁信息时，控制电机驱动，计算速度差值，传输速度参数，得到反馈量。系统软件工作框图如图 2 所示。同时，针对上述的系统结构图，可以分析出软件功能需求模块，如速度控制程序、传感器接口程序、角度控制程序、数据读写控制程序和输入/输出控制程序。系统软件结构图如图 3 所示。

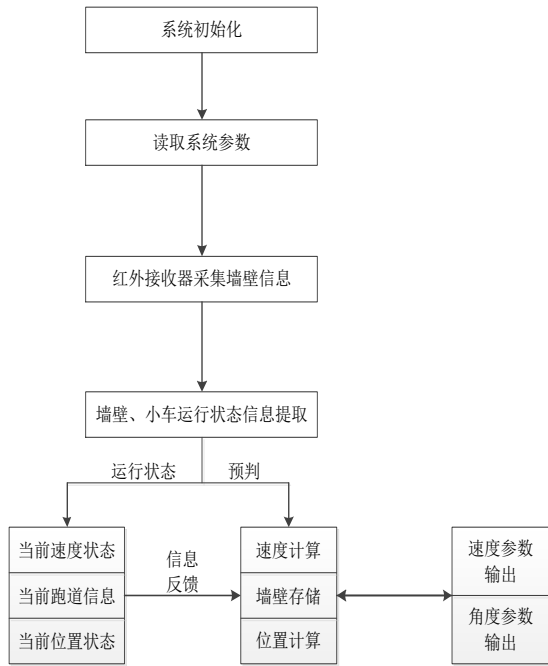


图 2 系统软件工作框图

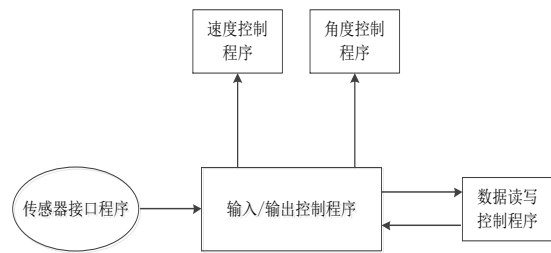


图 3 系统软件结构图

3. 3 系统硬件结构设计

在系统硬件设计中，各个器件应当互相兼容，减少干扰，通过系统总线将各个器件连接起来，由 CPU 作为主控制器，调整兼容器件间的电磁干扰。结构设计如图 4 所示。

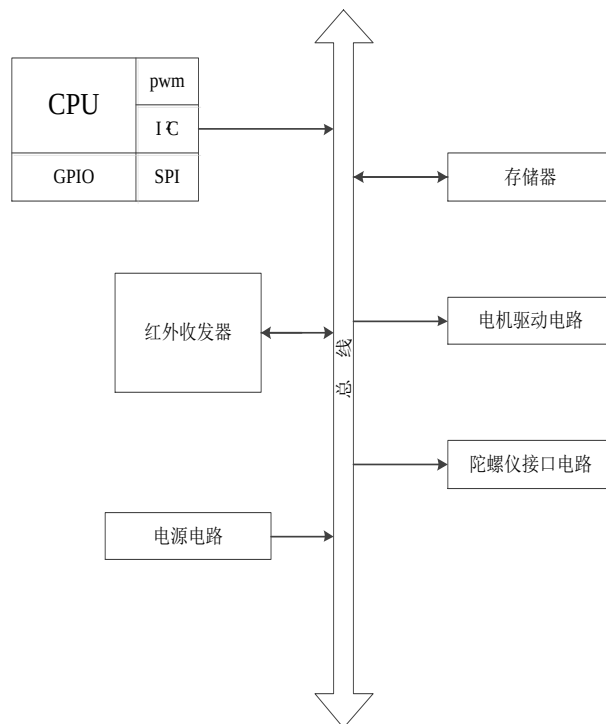


图 4 系统硬件结构图

4 系统详细设计

4.1 硬件构件设计

对于硬件主要芯片的选择，应当从多方面的角度考虑，既要实现系统的基本功能，又要追求各个器件相互兼容处理。根据前文提出的功能需求，分析本系统硬件构件的组成成分，合理妥善的选择需要的元器件。

4.1.1 微控制器

LM3S615 是基于 ARM[®]系列的 CortexTM-M3 的控制器，价格与 8 位和 16 位器件相同，却能为用户提供 32 位器件的性能，并且所有器件都是以小型封装的形式提供^[3-6]。

4.1.2 电机驱动电路

通过对功能需求的分析，直流电机采用德国制造的 1717SR 空心杯直流电机，输入电压 7.4V，可以输出 1.96W 的功率，最大转速为 14000rmps^[7-8]。

(1) 直流电机的 PWM 驱动电路

图 5 是最基本的 PWM 驱动电机的电路。开关管 V1 决定着电机的转动控制，由电平信号的高低控制电机的电流，以此来对平均电压进行控制。当 U_i 输入高电平时，PWM 信号传输，导通开关管 V1，使得电机的两端电压为 U_s 。在图中 t_1 秒后， U_i 的输入由高电平转向低电平，此时，V1 被关闭，无电流流入。但是 t_2 秒后，又重新开始循环。由此得到电机的电枢绕组两端的平均电压 U_o 如式（1）所示。

$$U_o = (t_1 * U_s) / (t_1 + t_2) = U_s * (t_1 / T) = \alpha * U_s \quad (1)$$

式中 α 是占空比。

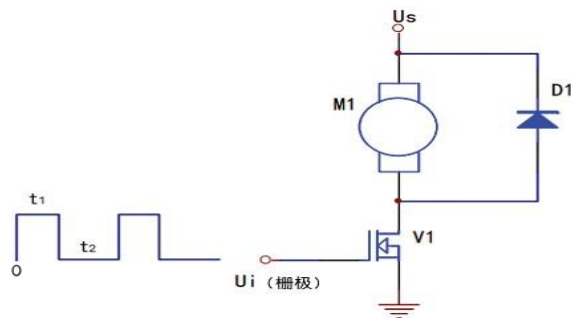


图 5 基本 PWM 驱动电路

而改变电枢电压是直流调速的主要方法。其控制电机正反转 PWM 波形如图 6 所示，通过改变 PWM 占空比调节电机的速度大小^[9-10]。

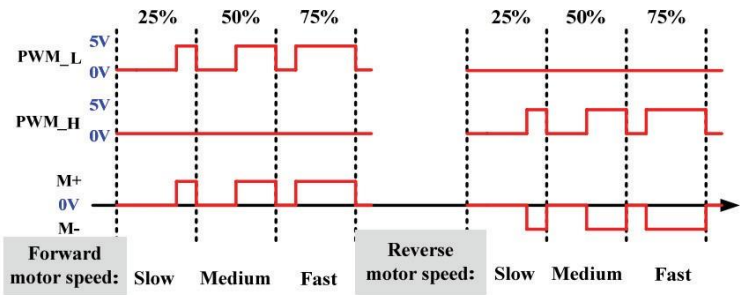


图 6 单极性 PWM 驱动方式

(2) 全桥驱动电路

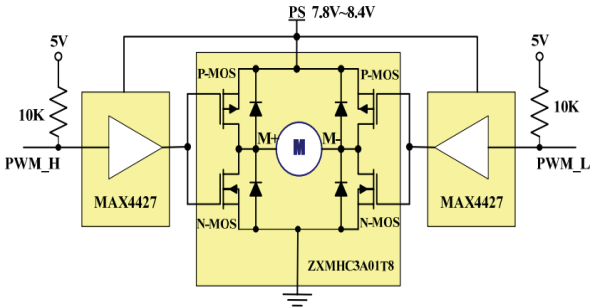


图 7 全桥电路结构图

直流电机的驱动方式使用图 7 的全桥结构，这个结构的主要功能是完成对于电机的驱动控制，来实现调整智能小车的速度与转向。

在驱动电路中，由 PWM 信号控制电机速度。当 PWM_H 的 L 和 PWM_L 的 H 导通时，电机的状态为顺时针转动；当 PWM_H 的 H 和 PWM_L 的 L 导通时，电机的状态为逆时针转动；当 PWM_H 的 L 和 PWM_L 的 L 导通时，电机的状态为停止转动，此时电流顺时针回转；当 PWM_H 的 H 和 PWM_L 的 H 导通时，电机的状态也为停止转动，此时电流逆时针回转。

在同一电源系统下控制直流电机的电流，再结合 PWM 技术，通过传输信号来达到电机正转和反转的目的。表 2 是全桥电路的真值表，图 8 为其四种导通模式。

表 2 全桥电路真值表

PWM_H	PWM_L	状态
L	L	正转刹车
L	H	正转
H	L	反转
H	H	反转刹车

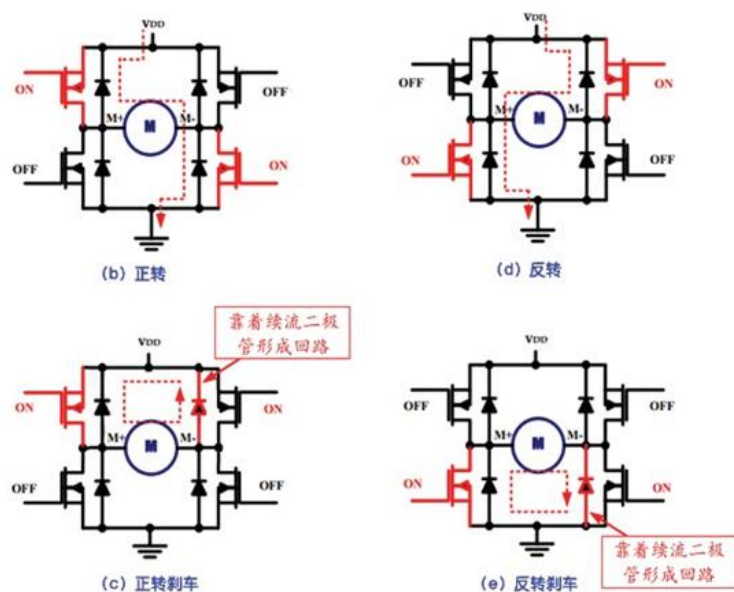


图 8 四种全桥电路导通模式

(3) 电机驱动硬件电路

电机驱动的硬件电路原理图如图 9 所示。其中 MAX4427 芯片可以增强电机驱动电流；HEADER 是电机电路，控制旋转速度；ZXMH3A01T8 是 H 桥电机驱动芯片，实现电机正反转控制。

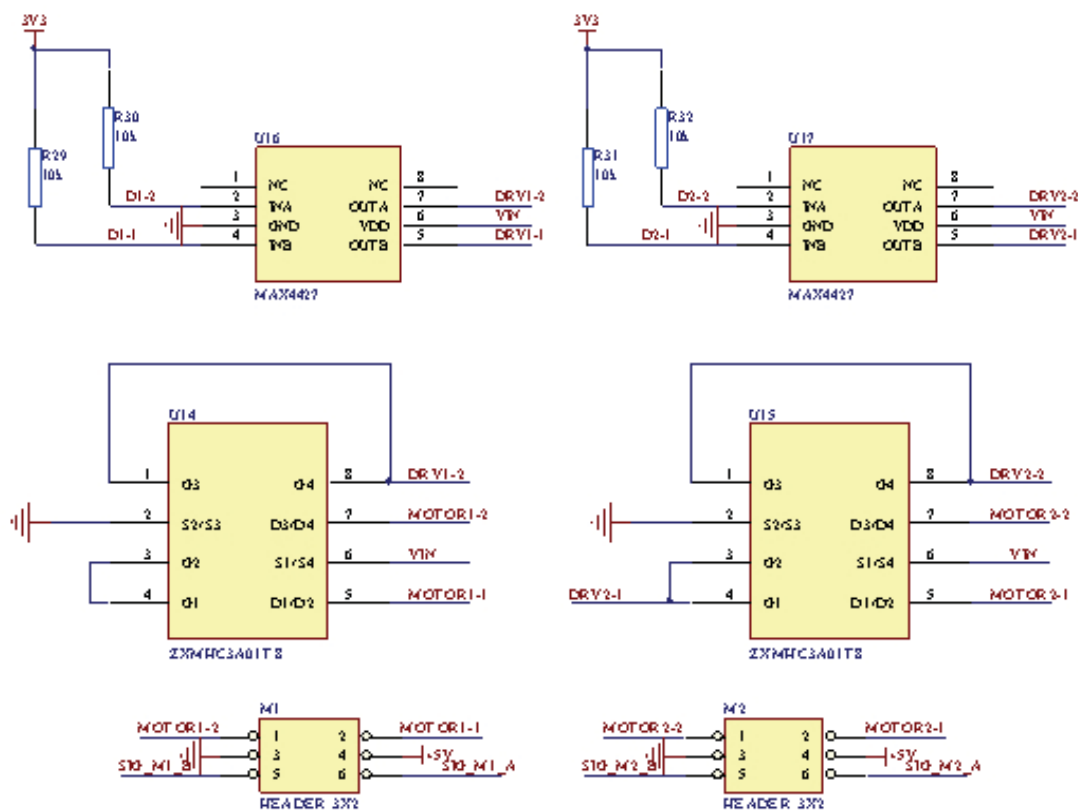


图 9 直流电机驱动硬件电路

4. 1. 3 红外收发器

ARM 智能小车上共设计了 6 组红外接收传感器，每个传感器由两部分组成，一部分是发射器，另一部分是接收器。它的主要作用是检测视野范围内的障碍物，粗略判断障碍物的远近距离，用于迷宫格信息的识别和转弯的控制。

一体化接收头含有一个中心频率为 38KHz 的带通滤波器^[11-13]，如图 10 所示，在不改变驱动电流和电压的情况下，改变载波频率就可以判断出障碍物的远近。

当传感器中的带通滤波器的频率为 38KHz 时，传感器的灵敏度最高。通过这个特性完成障碍物距离的测定，当离中心频率越近时，信号衰减越少；当离中心频率越远时，信号衰减越多。即小车前进时，当离障碍物越近时，频率也就越高，信号也就越强；当离障碍物越远时，频率也就越低，信号也就越弱。

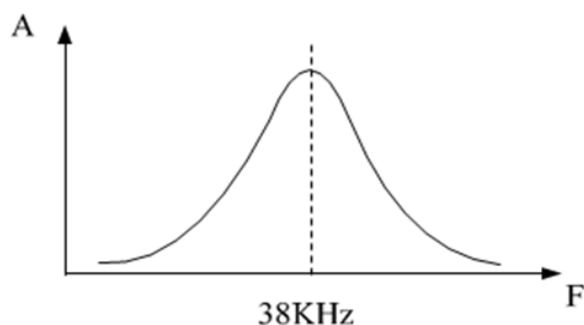


图 10 带通滤波示意图

ARM 智能小车红外线收发传感器电路图，如图 11 所示。其中 IR1 是发射红外线电路，为了增强红外发射强度采用 SN74 作为红外驱动芯片，使红外信号更容易检测和接收。

对红外接收器而言，可以调节灵敏度，通过使用带通滤波中心频率的特性，调节小车距离障碍物的最小距离，设置参数为 4.2 厘米。

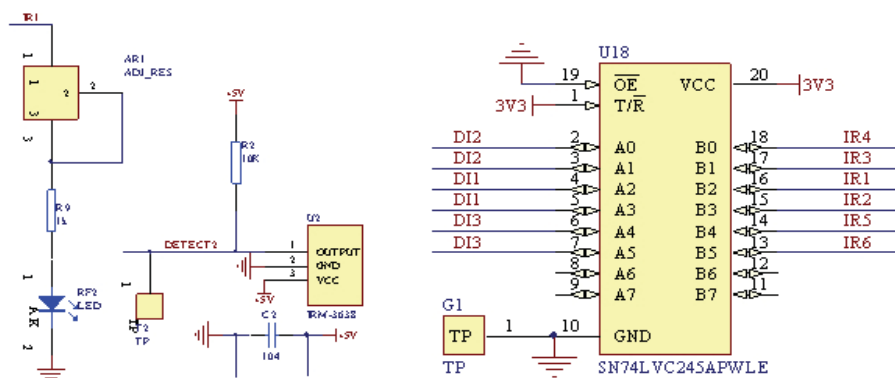


图 11 红外检测电路

4. 1. 4 陀螺仪接口电路

本系统中调节小车转弯角度的硬件设计采用 MPU6050 陀螺仪，MPU6050 内部集成 16 位 ADC^[14-16]。电路原理图如图 12 所示。

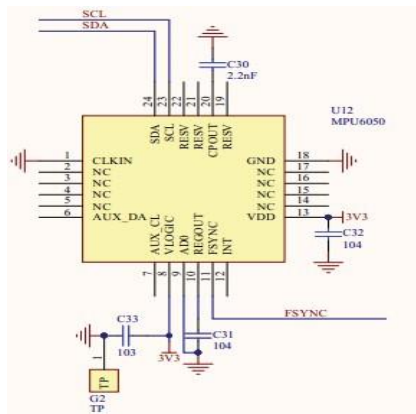


图 12 MPU6050 电路图

4. 1. 5 电源电路

ARM 智能小车电源控制系统包含 7.4V、5V 和 3.3V 三个电压电路。5V 电源电路为门电路、传感器提供电源；3.3V 电源电路为系统 CPU 提供电源；7.6V 电源电路为电机以及电机驱动芯片电路提供电源。

供电电路使用 LM311 比较器将电池进行分压，设置 3.3v 作为阈值与电压进行比较，如果当电压过低时，低电压报警电路将会亮起红灯进行提醒，以防止电池电压过低^[17-19]。

4. 1. 6 存储器电路

该系统采用 24WC08 E PROM 存储器作为外部存储系统，且 E PROM 具有掉电不丢失数据的特性。因此，用以存储一些运行参数，诸如红外传感器发射频率参数，系统 PI 参数等。电路原理如图 13 所示^[20-21]。

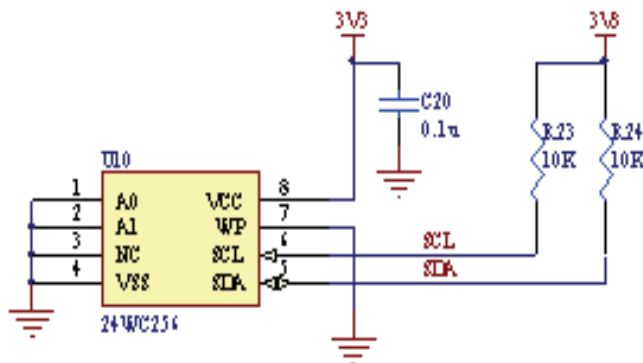


图 13 24WC08 E PROM 电路图

4. 2 软件模块设计

在软件模块设计中，将程序设计分为底层驱动程序和顶层算法程序，底层驱动程序主要控制速度和角度，完成小车的转弯设置，通过驱动连接实现速度和角度结合，控制电机转速和陀螺仪调节角度；顶层算法程序主要完成智能小车控制系统运行状态、存储墙壁信息、建立等高图和选择最优路径。

4. 2. 1 速度控制程序设计

(1) 增量式 PID 算法

由上面可知直流电机的速度的大小是通过调节输出 PWM 波的占空比的大小来改变的，那么在控制电机速度时依据什么去改变 PWM 占空比的大小从而使电机快速平稳达到所需速度，在这里采用通用的增量式数字 PI 控制器，速度 PI 控制器如图 14 所示。

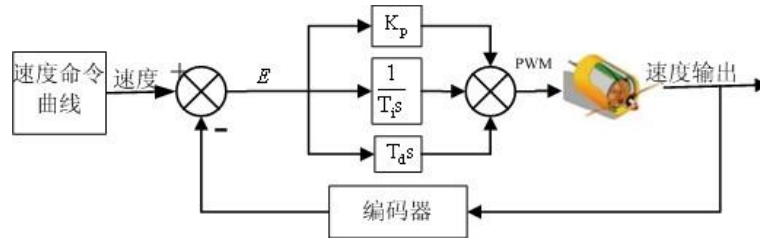


图 14 速度 PI 控制器的示意图

从图 14 可知，实际的转速由编码器计算得出，通过理论速度与平均实际转速相比较，将产生的差值 e 代入 PI 控制器中进行输出 PWM 波，将得出的占空比结果控制电机电源电路，来控制电流驱动 DC 电动机以改变其旋转速度，之后不断重复这个过程，直至产生的差值 e 为 0 为止。

所谓增量式 PID 是指数字控制器的输出量是其控制量的增量 ΔU_k 。将增量 ΔU_k 看作是理论速度与实际速度的差值。

增量式 PID 控制算法可以通过式 (2) 推导出^[22]。由式 (2) 可以得到控制器的第 $k-1$ 个采样时刻的输出值式 (3)。

$$u_k = k_p \left[e_k + \frac{T}{T_i} \sum_{j=0}^k e_j + T_d \frac{e_k - e_{k-1}}{T} \right] \quad (2)$$

$$u_{k-1} = k_p \left[e_{k-1} + \frac{T}{T_i} \sum_{j=0}^{k-1} e_j + T_d \frac{e_{k-1} - e_{k-2}}{T} \right] \quad (3)$$

将式（2）与式（3）相减整理，就可以得到增量式 PID 控制算法如式（4）所示。

$$\begin{aligned}
 \Delta u_k = u_k - u_{k-1} &= k_p(e_k - e_{k-1} + \frac{T}{Ti}e_k + Td\frac{e_k - 2e_{k-1} + e_{k-2}}{T}) \\
 &= k_p(1 + \frac{T}{Ti} + \frac{Td}{T})e_k - k_p(1 + \frac{2Td}{T})e_{k-1} + k_p\frac{Td}{T}e_{k-2} \\
 &= Ae_k + Be_{k-1} + Ce_{k-2}
 \end{aligned} \tag{4}$$

其中

$$A = k_p(1 + \frac{T}{Ti} + \frac{Td}{T});$$

$$B = k_p(1 + \frac{2Td}{T});$$

$$C = k_p\frac{Td}{T}。$$

由式（4）可以看出，确定 T、A、B、C，就可以由式（4）求出控制量。

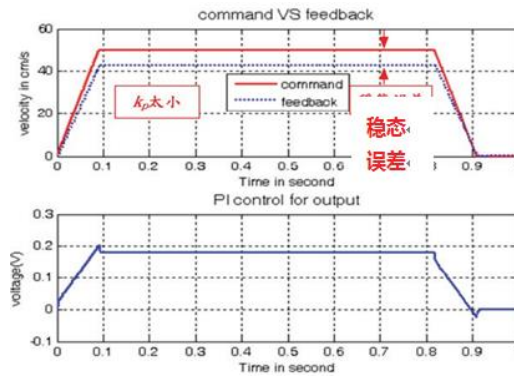
其中比例系数 k_p 、积分系数 k_i 的具体数值。设定步骤如下：

①先调整积分增益 k_i 为 0，接着调整比例增益 k_p ，直到速度反馈信号产生振荡，如图 15-a 输出结果波形图所示。

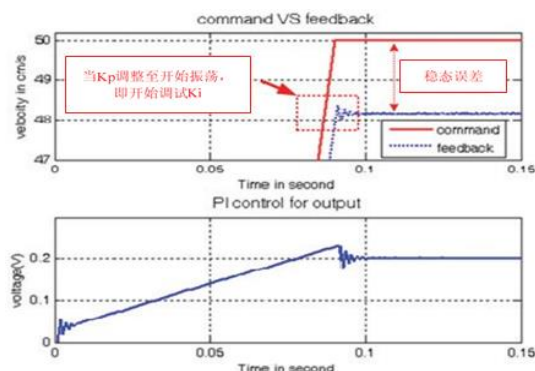
②调整积分增益 k_i ，直到消除稳态误差, 如图 15-b 输出结果波形图所示。

③输出结果如果不理想，重复步骤①和②，加大比例系数 k_p 的值，让振荡现象更明显，直到输出理想结果为止，如图 15-c 输出结果波形图所示

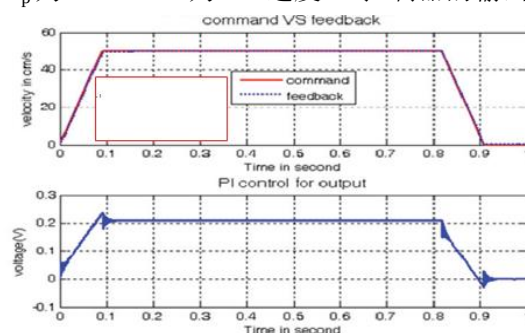
其中 Feedback 为反馈实际值，command 为命令设定值。



a. 当 k_p 为 0.025， k_i 为 0，速度 PI 控制器的输出结果



b.当 k_p 为 0.011, k_i 为 0, 速度 PI 控制器的输出结果



c.当 k_p 为 0.011, k_i 为 3, 速度 PI 控制器的输出结果

图 15 仿真波形图

(2) 直流电机闭环控制系统

在直流电机闭环控制设计中, 根据具体功能情况, 整个速度控制系统如图 16 所示, 由两个 PI 闭环组成:

第一个 PI 闭环为: 设定值为给定的直行速度 ($xSpeed$), 用左轮的速度和右轮的速度之和再除以 2 ($(rightSpeed+leftSpeed)/2$) 作为实际直行速度的反馈量, 将它与给定的直行速度相比较, 做偏差得到 $xError$ ($xError=xSpeed-(R+L)/2$), 再把偏差 $xError$ 送到 PI 控制器进行闭环运算得到输出 $U1$ 。在 PI 闭环的控制下, 不断调整 $U1$ 输出, 控制 PWM 波的占空比, 改变左右轮的直行转速, 使得实际直行速度反馈量不断接近给定的直行速度。当 $xError$ 无限趋于 0 时, $xSpeed=(R+L)/2$, 此时, 第一个 PI 闭环控制系统完成直行速度调节。

第二个 PI 闭环为: 设定值为给定的旋转速度 ($wSpeed$) 和位置校正输入之和, 旋转速度的改变在于左右轮转速的不一致或者方向不同, 当左轮的转速大于右轮的转速时, 小车将会向右偏移; 当左轮的转速小于右轮的转速时, 小车将会向左偏移; 当左轮与右轮的旋转方向相反时, 小车将会向后旋转。用右轮速度减去左轮速度再除以 2 ($(rightSpeed-leftSpeed)/2$) 得到实际旋转速度的反馈量, 将它与给定的旋转速度相比较, 做偏差得到 $wError$ ($wError=wSpeed-(R-L)/2$), 再把偏差 $wError$ 送

到 PI 控制器进行闭环运算得到输出 U2。在 PI 闭环的控制下，不断调整 U2 输出，控制 PWM 波的占空比和输出流向，改变左右轮的旋转转速，使得实际旋转速度反馈量不断接近给定的旋转速度。当 wError 无限趋于 0 时，wSpeed= (R-L) /2，此时，第二个 PI 闭环控制系统完成旋转速度调节。

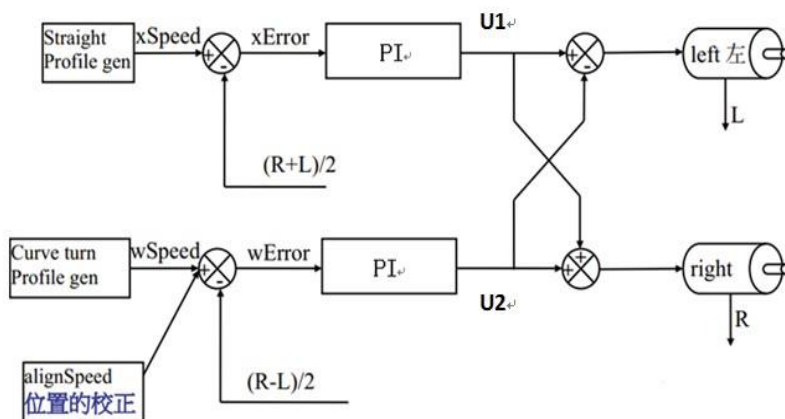


图 16 直流电机闭环控制系统框图

其中 $\text{leftSpeed} = \text{xSpeed} - \text{wSpeed}$

$\text{rightSpeed} = \text{xSpeed} + \text{wSpeed}$

$\text{xSpeed} = (\text{rightSpeed} + \text{leftSpeed}) / 2$

$\text{wSpeed} = (\text{rightSpeed} - \text{leftSpeed}) / 2$

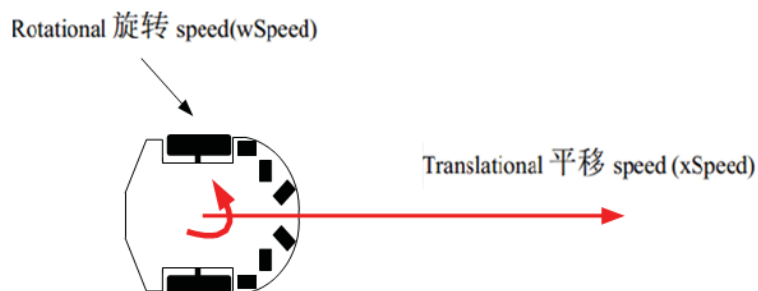


图 17 直行和旋转速度示意图

把 U1 减去 U2 的值作为左电机的驱动信号，把 U1 加 U2 的值作为右电机的驱动信号，将数据传输到控制系统中，调整智能小车的速度，如图 17 所示。这两个 PI 闭环就是控制小车直行和旋转速度的调制器。

小车的运行状态将基于两个 PI 闭环进行控制，输出直行和旋转两种运行方式，结合陀螺仪达到旋转的输出目的。通过闭环系统，可以控制直流电机的转速和转向，将 PWM 信号与差值结合，用差值 e 计算 t1 的时间，调整 t1 占 T 的比例，改变 PWM 波的占空比，从而调整电枢电压。

(3) 速度控制程序流程图

闭环控制系统流程，如图 18 所示。

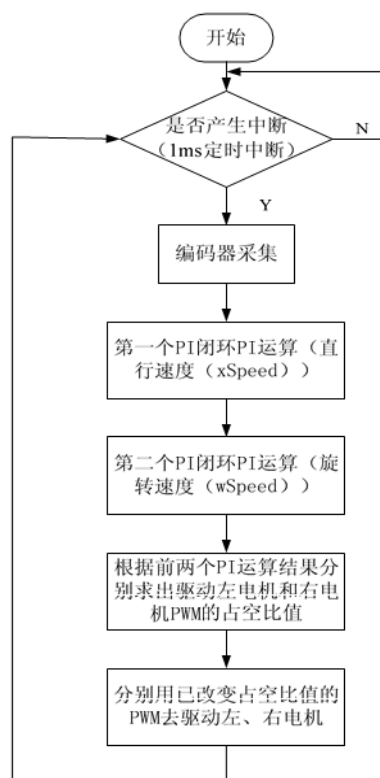


图 18 闭环控制系统流程图

增量式 PID 算法流程，如图 19 所示。

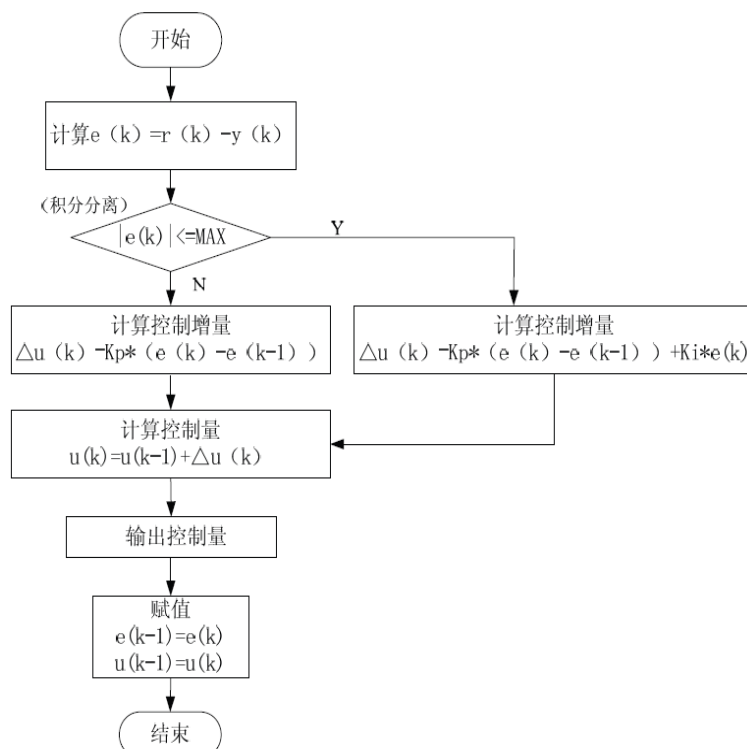


图 19 增量式 PID 算法流程图

4. 2. 2 传感器接口程序设计

(1) 传感器主程序流程图

红外线传感器接口程序是将接收到的信息发送到主芯片中，对于是否接收信息设置标志位，进行中断。其中，主程序流程图，如图 20 所示。

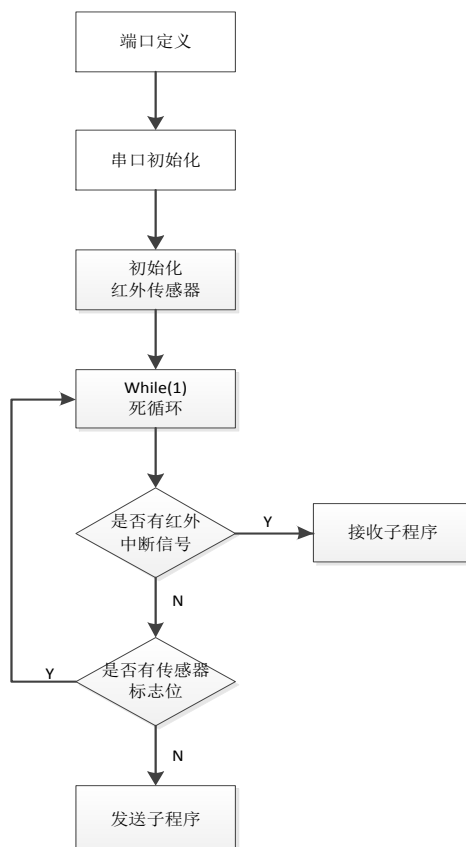


图 20 主程序流程图

(2) 传感器接口程序

```
void__sensorInit(void)
{
    TimerDisable(TIMER1_BASE,TIMER_B);
    TimerControlLevel(TIMER1_BASE,TIMER_B,true);
    TimerDisable(TIMER1_BASE,TIMER_B
        __IRSEND3_LR);
    PWMGenConfigure(PWM_BASE,PWM_GEN_1,PWM_GEN_MODE_UP_DOWN
        __IRSEND3_LR);
    PWMOinPutState(PWM_BASE,PWM_OUT_7_BIT
```

```

        |PWM_OUT_3_BIT,false);

PWMGENEYable(PWM_BASE, PWM_GEN_7);

TimerControlLevel(TIMER1_BASE,TIMER_B,true);

TimerDisable(TIMER1_BASE,TIMER_B,

        |__IRSEND3_LR);

TimerDisable(TIMER1_BASE,TIMER_B);

TimerDisable(TIMER1_BASE,TIMER_B);

TimerControlLevel(TIMER1_BASE,TIMER_B,true);
}

```

4. 2. 3 角度控制程序设计

(1) 转弯角度值的计算

在陀螺仪硬件设计中可知，智能小车旋转角度由陀螺仪控制，其数字信号输出角速度，而角度值需对其进行定积分。

当陀螺仪的量程为 $\pm 2000^\circ/\text{s}$ ，采样周期为 1ms 时。获取 MPU6050 陀螺仪的输出值 GYRO_ZOUT，通过资料中的公式，将数据各个代入，可以计算出 MPU6050 陀螺仪的角速度值 $w = \text{GYRO_ZOUT} / 16.384^{[23-24]}$ 。

计算机积分的思想就是离散求和，设转弯角度为 θ ，如公式（5）所示。

$$\theta = \int_0^t w dt = \sum w(i) * t(i) \quad (5)$$

将采样周期设为 1ms，只需将每个周期的采样值与采样周期相乘，再求和，便得到角度积分。

(2) 角度控制程序

```

void GYRO_Z_Angle(void)
{
    int16 w=0;

    zlg7289Download(0,0,0,GW%10);

    w=w/16.4;

    GWH=GWH+w;
}

main (void)

```

```

{
    int16 Val;

    TimerControlLevel (TIMER1_BASE,TIMER_B,true _MAIN
                        |__IRSEND3_LR);

    TimerControlLevel(TIMER1_BASE,TIMER_B,true);

    TimerDisable(TIMER1_BASE,TIMER_B);

    zlg7289Download(0,0,0,GW%10);

    mazeInit();

    MPU6050_init(1);

    zlg7289Download(0,0,0,GW%10)
    {
        GYRO_Z_Angle();

        zlg7289Download(0,3,0,GW%10);

        zlg7289Download(0,0,0,GW%10);

        zlg7289Download(0,0,0,GW%10);

        zlg7289Download(0,3,0,GW%10);

        mazeInit();

    }
}

```

4. 2. 4 最优路径程序设计

(1) 建立坐标和方向

为了便于智能小车对最优路径的选择与判断，需要让智能小车通过记忆功能记录所走过的路径，为了方便记录路径，现将整个迷宫抽象化，并以智能小车的绝对方向为参照物，建立直角坐标系。通过坐标对迷宫所有地方进行标记，用栈的方式记录迷宫坐标。

在本系统中，为了便于方向的辨别和管理，将把小车在迷宫中向前的方向定义为 0、向右的方向定义为 1、向下的方向定义为 2、向左的方向定义为 3。

(2) 墙壁信息存储

在智能小车行驶的过程中，它需要不断的去记忆所走过的路程信息，其中，最

主要就是墙壁信息，用于判断该路径是否搜索过。因此，在智能小车的内部存储器中，将用字节的低四位记录智能小车走过的墙壁信息，如表 3 所示。

表 3 墙壁信息存储方式

变量位	绝对方向	相对方向
bit0	上方 0	1: 有路，0: 有墙壁
bit1	右方 1	1: 有路，0: 有墙壁
bit2	下方 2	1: 有路，0: 有墙壁
bit3	左方 3	1: 有路，0: 有墙壁

(3) 迷宫搜索方法

1) 右手法则

该法则基于数据结构中的广度优先原则，当前小车面临两种以上的选择时，小车优先考虑向右转弯，其次向前，最后才考虑向左转弯。示意图如图 21 所示，图中坐标（0，0）为小车出发点，虚线为小车运动路径。

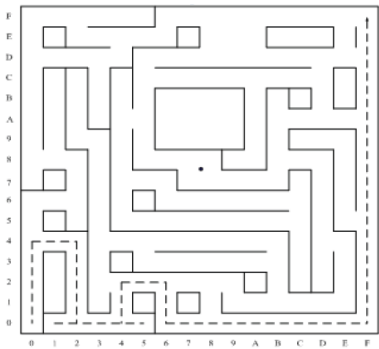


图 21 右手法则示意图

2) 左手法则

与右手法则相似，在迷宫搜索方法策略上，采用中序遍历，小车优先考虑左转弯，其次是向前行，最后考虑向右转弯。示意图如图 22 所示，图中小车出发点依然是坐标（0，0），虚线依然为小车运动路径。

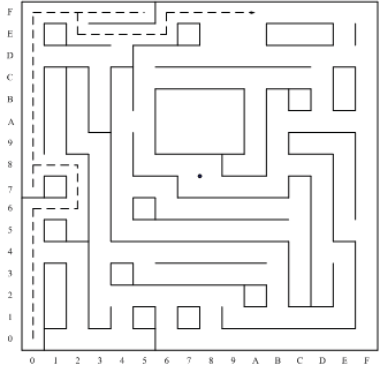


图 22 左手法则示意图

3) 中心法则

中心法则的关键原理在于往迷宫中心靠近行驶，每当小车需要做出选择时，优先考虑靠近中心的选择。

如何确定哪个方向更能接近目标呢？如图 23 所示，通过对整个迷宫进行区域划分，分成四个等份，标注为 1、2、3、4。可以得出结论，观察区域 1 为迷宫左下角可知，智能小车只要向右和向上行驶就可以靠近中心。观察区域 2 为迷宫右下角，仅向左和向上便能靠近中心。同样还可以观察出区域 3 和区域 4 靠近中心的方向。

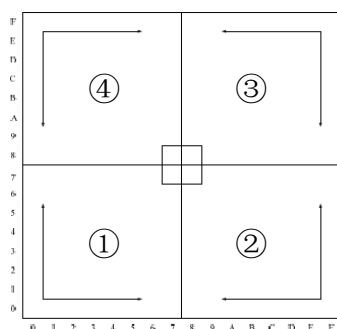


图 23 迷宫分区示意图

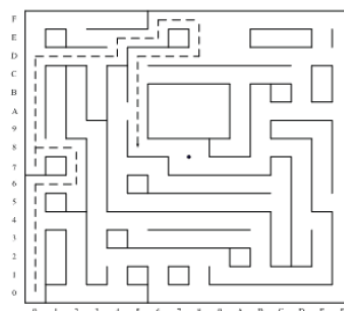


图 24 中心法则示意图

当小车行进的方向有两个选择时，这两个方向都有可能靠近中心，在中心法则的作用下优先选择直线行驶的方向，如果不能够直行，就选择转弯。

当小车行进远离中心时，在中心法则的作用下优先选择直线行驶的方向，如果不能够直行，就选择转弯。其搜索示意图如图 24 所示^[25-26]。

(4) 最优路径算法

在该算法中，如果智能车通过迷宫搜索的方法已经完成了对迷宫的搜索，并在搜索的过程中存储行进迷宫时的相关墙壁信息，那么它是如何通过这些路径信息判断出从起点到终点的最优路径？而系统采用图论方法，利用等高线图将迷宫坐标的值做上标记为迷宫格上从起始点到目的点的位移步数，使智能车独立选择最短路径。

其中，假设小车当前的位置为数值 1，在进行的过程中，每前进一步就对设置的标识位自加 1，以此类推，小车不断行进，标识位就不断的进行自加。

1) 图论算法流程

本系统通过一个 4*4 的小迷宫详细说明图论算法，设原点当前位置为起点。

首先，将每个前进的位移数设置为 1，并将迷宫上的等高值填充到 0xff。当原点前进并改变时，变量 Step 从 0xff 增加到 1。记录等高值并将起点坐标 (0, 0) 存储在堆栈中，如图 25 所示。

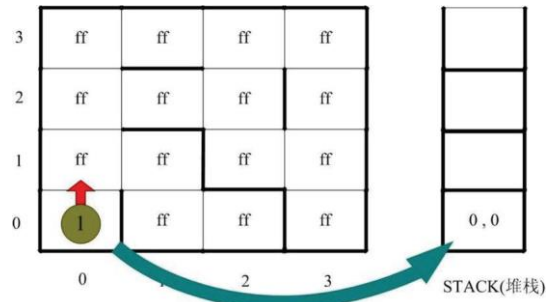


图 25 等高图一

图 25 可知，原点在 (0, 0) 四周仅有一个方格可以行进，进入箭头指定方向的方格，此时，坐标为 (0, 1)，变量 $Step=Step+1$ ，得到下图 26。

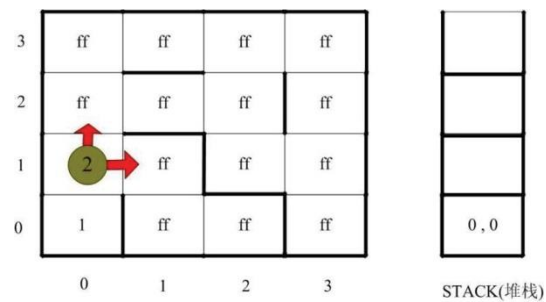


图 26 等高图二

图 26 可知，在 (0, 1) 有两个方格可以行进，所以存储选择坐标在堆栈中并选择任意方格。此时，坐标为 (1, 1)，变量 $Step = Step + 1$ ，得到下图 27。

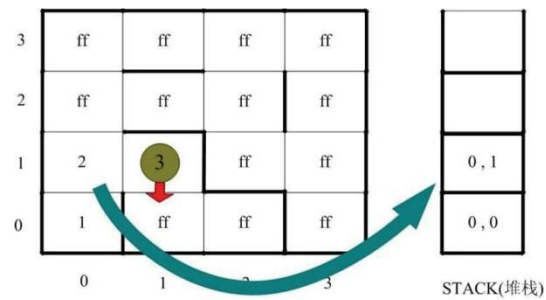


图 27 等高图三

图 27 可知，只有一个方格可以前进，进入指定方格，此时，坐标为 (1, 0)，变量 $Step=Step+1$ ，得到下图 28。

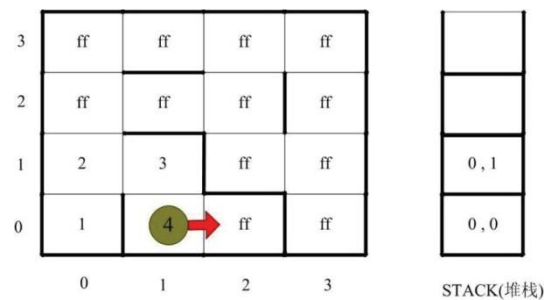


图 28 等高图四

图 28 可知，原点处仅有一个方格可以行进，进入箭头指定方格，此时，坐标为 (2, 0)，变量 $Step=Step+1$ ，得到下图 29。

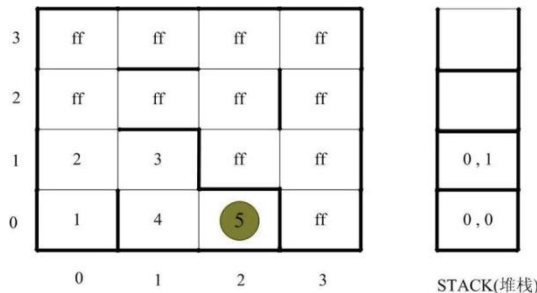


图 29 等高图五

图 29 可知，取出原先堆栈内容，跳转到之前的坐标 (0, 1)，读出该点的坐标 $Step$ 值， $Step=2$ ，以解决四周没有可以行进的方格，变量 $Step=Step+1$ ，得到下图 30。

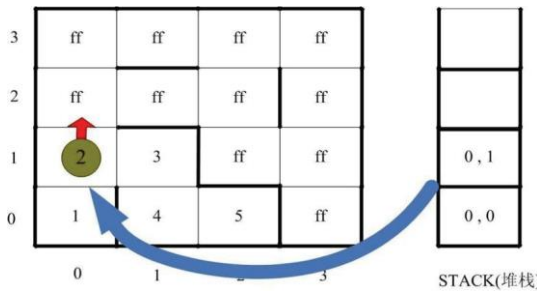


图 30 等高图六

以此类推，当堆栈中没有需要处理的坐标点，并且当前原点没有可以行进的方格为止，最终可以得到如下图 31 所示的等高图。

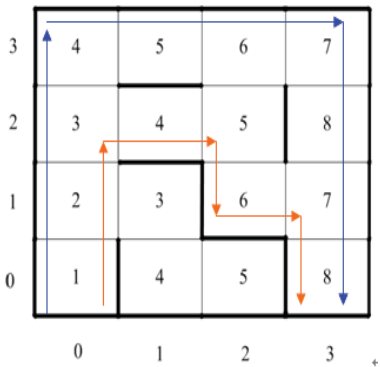


图 31 等高图最终的示意图

2) 转弯加权的图论算法

在图 31 中可以看出，(0, 2) 点和 (1, 1) 点的等高值都为 3，由于智能小车转弯要浪费时间，择优考虑智能小车选择 (0, 2) 点。

所以可以通过加权的方法来寻找一条最优的路径。设转弯的权值为 1，即当智能小车在行进的过程中遇到转弯点，则 $Step$ 自加 2。加权后的等高图如下图 32 所示。

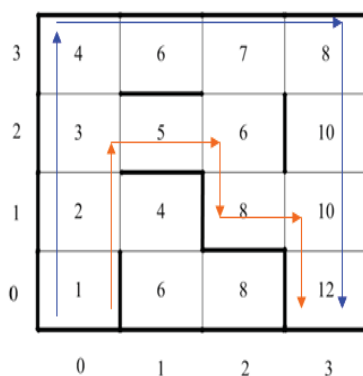


图 32 转弯加权后的等高图

4. 2. 5 输入/输出控制程序设计

(1) 运行状态程序设计

1) 智能小车直行

智能小车的直行原理在于调整电机左右速度和调整陀螺仪度数为 0，将左右轮速度对等，不能产生差值，调整旋转速度为空。通过分析可得，将控制系统中的第一个 PI 控制闭环的参数设置相等，调整反馈量 $(R+L)/2$ ，对于第二个 PI 控制闭环，将它所负责的转弯速度调为空，且在第一个 PI 闭环的调节下使偏差 $xError$ 逐渐趋于 0，即 $xSpeed = (R+L)/2$ ，在第二个 PI 闭环的调节下会使偏差 $wError$ 逐渐趋于 0，即 $wSpeed = (R-L)/2 = 0$ ，这就使得左轮速度等于右轮速度 ($R=L$)，这样就使得智能小车以 $xSpeed$ 为直线速度做直行运动。

2) 智能小车转弯

智能小车转弯的原理在于将左右电机的速度和陀螺仪进行调整，一慢一快产生差值进行九十度转弯。图 33 为智能小车转弯示意图。

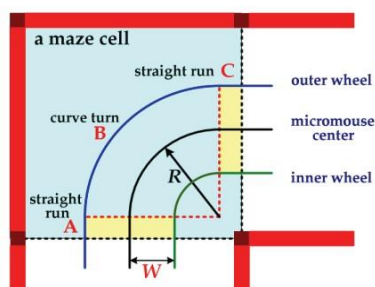


图 33 前进中 90°转弯轨迹示意图

①左转弯

左转弯的控制原理在于调整智能小车左右轮的旋转速度，当智能小车同时向前运动时，增加右轮的速度，减小左轮的速度。在控制速度公式中，将第一个 PI 闭环

中的参数设置为固定值，将旋转速度设置为正值，将速度大小调整成适合旋转的速度，便于控制智能小车在前进中左转。通过分析可得，其中第一个 PI 闭环的设定值直行速度（xSpeed）为所需值，反馈为 $(R+L)/2$ ，而第二个 PI 闭环的设定值为正值，反馈为 $(R-L)/2$ ，通过计算，可以调节第一个 PI 闭环使偏差 xError 逐渐趋于 0，则 $xSpeed = (R+L)/2 = \text{正值}$ ，而通过调节第二个 PI 闭环会使偏差 wError 趋于 0，则 $wSpeed = (R-L)/2 = \text{正值}$ ，这使得左轮速度小于右轮速度 $(R>L>0)$ ，这样就使得智能小车向左偏移实现在前进中左转弯。部分程序代码如下：

```
void mouseTurnleft(void)
{
    while((GmRight.uiPulseCtr+200)<=GmRight.uiPulse);
    while((GmLeft.uiPulseCtr+200 )<=GmLeft.uiPulse);
    GmLeft.uiPulse =3000;
    GmLeft.uiPulseCtr=0;
    GmRight.uiPulse =3000;
    GmRight.uiPulseCtr=0;
    mazeInit();
    break;
    GucMouseState=TURNLEFT;
    GucMouseDir=(GucMouseDir+3)%4;
    while(1)
    {
        if(GPIOPinRead(GPIO_PORTD_BASE, __DIR1) == 0)
        {
            break;
        }
    }
    while((GmRight.uiPulseCtr+200)<=GmRight.uiPulse);
    while((GmLeft.uiPulseCtr+200)<=GmLeft.uiPulse);
}
```

②右转弯

右转弯的控制原理在于增加左轮的速度，减小右轮的速度。在控制速度公式中，将直行速度设置为固定值，将旋转速度设置为正值，将速度调整成适合旋转的速度，便于控制智能小车在前进中右转。通过分析可得，其中第一个 PI 闭环的设定值直行速度 ($xSpeed$) 为所需值，反馈为 $(R+L)/2$ ，而第二个 PI 闭环的设定值为正值，反馈为 $(R-L)/2$ ，通过计算，可以调节第一个 PI 闭环使偏差 $xError$ 逐渐趋于 0，则 $xSpeed = (R+L)/2 = \text{正值}$ ，而通过调节第二个 PI 闭环会使偏差 $wError$ 趋于 0，则 $wSpeed = (R-L)/2 = \text{正值}$ ，这使得左轮速度大于右轮速度 ($0 < R < L$)，这样就使得智能小车向右偏移实现在前进中右转弯。图 34 为前进中转弯两轮的速度曲线图。

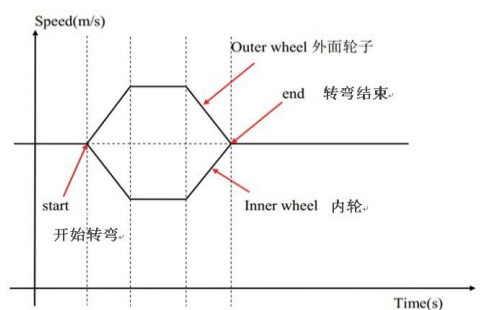


图 34 前进中转弯两轮的速度曲线图

(2) 运行过程程序设计

ARM 智能小车在运行中可以划分为四个阶段，等待阶段、启动阶段、搜索迷宫阶段和选择最优路径阶段。

1) 等待阶段

在等待阶段中，通过开启按键，设置智能小车中启动程序，激活内部程序，进行初始化操作，但是智能小车并不运行，需增加开启条件。

2) 启动阶段

在启动阶段中，需要完成的任务是判断启动坐标所在的位置。如果是左转，那么起点就在 $(0, 0)$ ，否则 $(F, 0)$ 。其程序流程图见图 35 所示。

3) 搜索迷宫阶段

在搜索阶段中，需要完成的任务是记忆行驶过程中迷宫的路径和墙壁信息。图 36 中采用右手搜索法则演示搜索的整体流程。

4) 选择最优路径阶段

前一节已说明判断的图论算法，在选择最优路径阶段中，智能小车会根据之前

搜索迷宫的相关信息，选择并判断到达终点的最优路径。

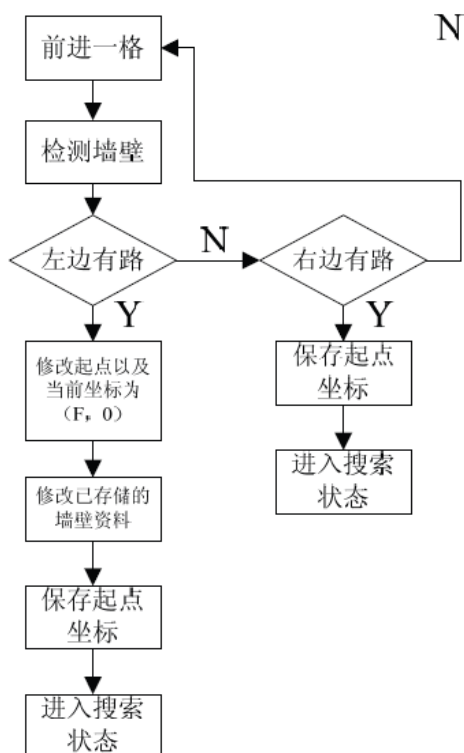


图 35 判断起点坐标程序流程图

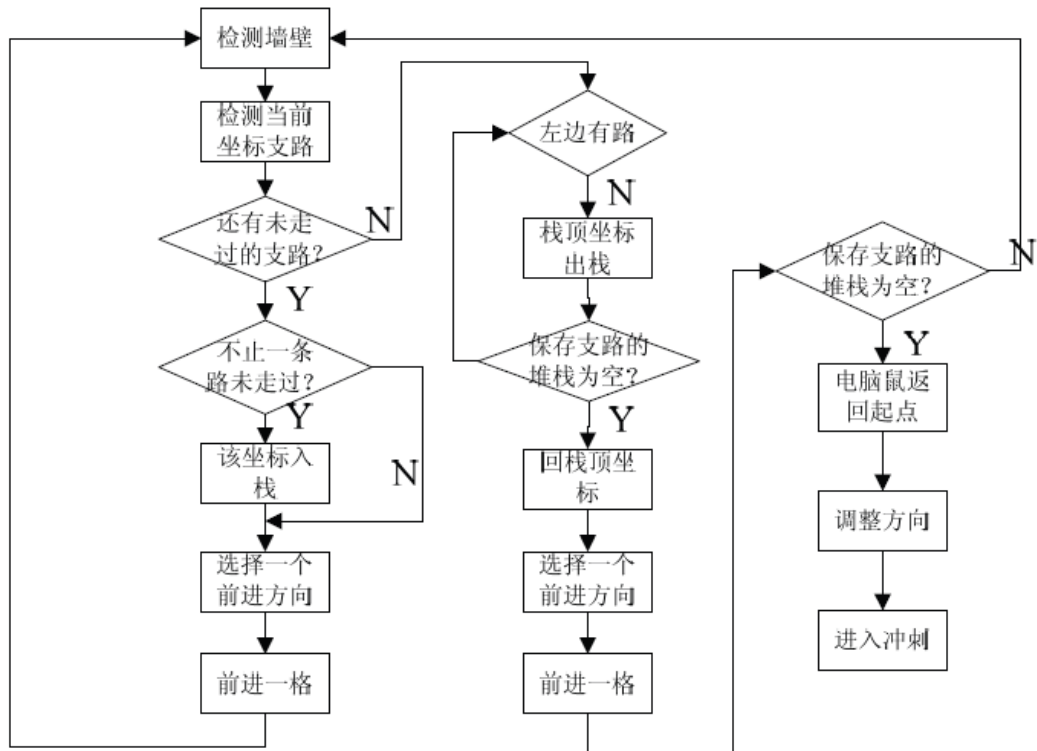


图 36 迷宫搜索流程图

5 系统集成与测试

5.1 系统实现

ARM 智能小车上使用了 LM3S615 的 PWM 发生器、GPIO 口、SPI 接口计数器/定时器、串行口及 I²C 接口。其电路原理图如图 37 所示。

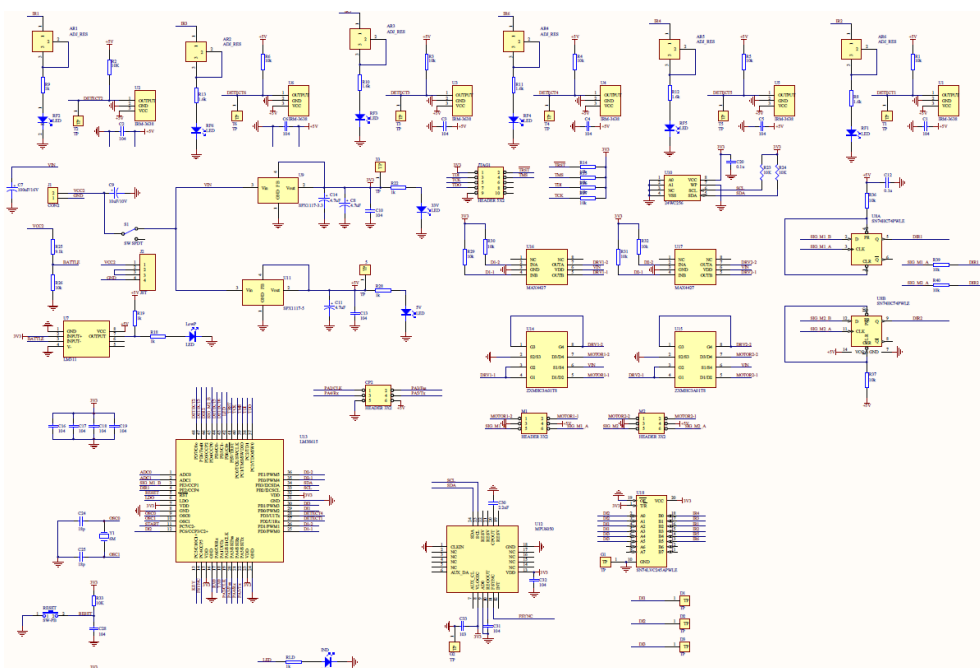


图 37 电路原理图

电机驱动 PWM D1_1、D1_2、D2_1、D2_2 中，电机驱动电路 M4427 的 INB 和 INA 与 D1_1、D1_2 相连接，作用是提供左电机驱动信号，电机驱动电路 M4427 的 INB 和 INA 与 D2_1、D2_2 相连接，作用是提供右电机驱动信号；为了便于通过编码器判断左右电机的正反转，让编码器输出信号接口与相对应的 E、D 口连接，将 SIG_M1_A、SIG_M2_A、SIG_M1_B、SIG_M2_B 四个接口与 D 型号触发器的 CLK、D 口相连接。

在红外传感器中，74LS245 引脚 4、3、1、2、5 和 6 输出 6 通道红外线，依次对应右侧、左侧、左前、右前、左 45°、右 45°；处理器上的输出端口 B1，C6 和 B0 分别对应三路红外发射器驱动信号的接口，提供 74LS245 输入信号，6 通道红外接收信号，对应输入端口 B4，B5，D7，D6，D3，D2，连接 5、6、2、3、4、1 红外接收器的输出。

陀螺仪接口 SDA、SCL、FSYNC，与 GPIOB3、GPIOB2、GPIOC4 分别对应，

且与 MPU6050 的 SCL、SDA 和 FSYNC 相连接；而串行口，分别与芯片的输入输出 A1，A0 接口连接；AT24C08 芯片的 SDA、SCL 端口通过记忆芯片的两个接口，分别对应 GPIOB3、GPIOB2；处理器上 RESET、C7 和 C5 接口与功能按键的三个接口连接。

其实物图如图 38、39、40、41 所示。



图 38 ARM 智能小车右侧图



图 39 ARM 智能小车前端图



图 40 ARM 智能小车俯视图



图 41 ARM 智能小车迷宫运行图

系统采用 IAR EWARM 作为集成开发环境，通过编译连接处理，生成可调式代码文件。弹出 Build 消息窗口，通过该窗口可以进行调试，如图 42 所示。

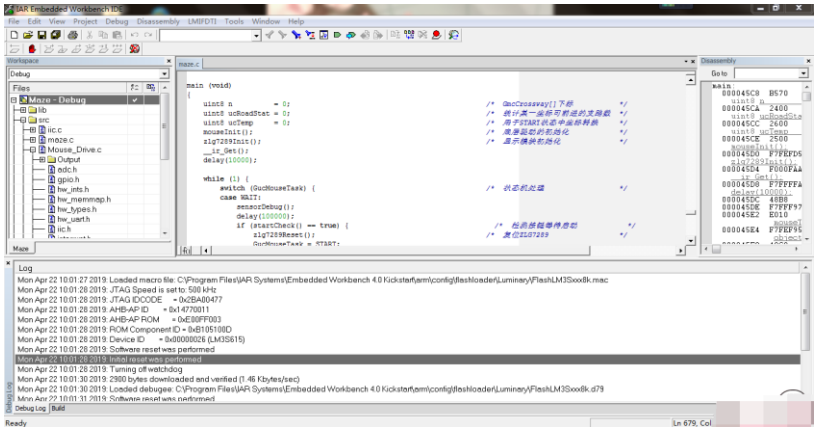


图 42 Build 消息窗口

5. 2 系统测试

ARM 智能小车控制系统以图 43 迷宫道路情况作为测试数据，虚线为搜索路径，实线为最优路径，通过小车实际运行状态作为实际输出，测试用例如表 4 所示。

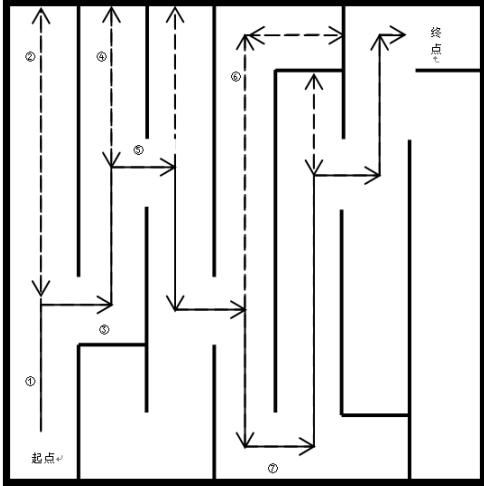


图 43 迷宫道路图

表 4 测试用例

测试编号	测试数据	预期输出	实际输出
1	无墙壁	直行	直行
2	有墙壁	向后	向后
3	有墙壁	向左	向左
4	有墙壁	向后	向后
5	无墙壁	向左	向左
6	有墙壁	向右	向右
7	无墙壁	向左	向左

5. 3 系统展示

(1) 自动搜索迷宫

在自动搜索迷宫中，实现 ARM 智能小车自动行驶和自动避障的基本功能，并对迷宫路径和墙壁信息进行记录。

接下来将展示 ARM 智能小车对 7*7 迷宫的探索过程，以中心法则作为核心算法搜索迷宫，设置左下角 3*3 的迷宫格为区域 1；右下角 3*4 的迷宫格为区域 2；左上角 4*3 的迷宫格为区域 3；右上角 4*4 的迷宫格为区域 4。在区域 1 中采用中右左算法；区域 2 采用中左右算法；区域 3 采用中左右算法；区域 4 采用中左右算法。

采用加权等高图记录算法，设置 ARM 智能小车每走一格，记录该坐标的等高值和四周墙壁信息，当遇到转弯时，加入权值 Step。设置 ARM 智能小车的起点坐标为 (0, 0)，终点坐标为 (6, 6)。

首先，将 ARM 智能小车开启，放置在起点 $(0, 0)$ 上，如图 44 所示。紧接着，智能小车启动，开始直行，如图 45 所示。



图 44 智能小车起点图



图 45 智能小车直行图

在图 45 中，智能小车遇到第一个选择口，根据中心法则区域 1 的算法，智能小车将继续直行，到达 $(0, 6)$ 后返回，如图 46 所示。返回到第一个选择口 $(0, 3)$ ，向左行驶，如图 47 所示。

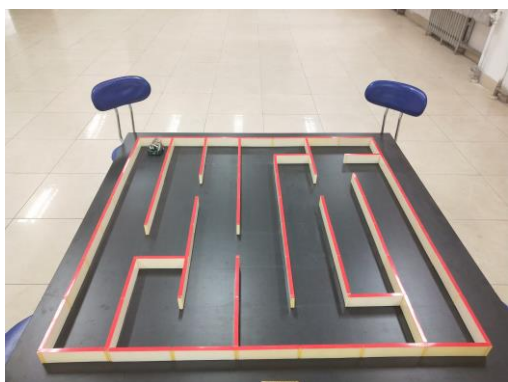


图 46 智能小车返回图



图 47 智能小车向左行驶图

智能小车继续转弯向前行驶，到达 $(1, 6)$ 点后，再返回到 $(1, 4)$ 点向左转弯，根据中心法则区域 3 的算法，智能小车再向左转弯，继续前行，如图 48 所示。



图 48 智能小车连续转弯图

到达 (2, 2) 点后, 为靠近中心, 智能小车向左转弯, 如图 49 所示。到 (3, 2) 点根据区域 2 的算法, 智能小车向左转弯, 到达 (3, 6) 点, 向右转弯, 进入死胡同后返回, 到达 (3, 0) 后向左转弯, 到达 (5, 0) 点, 如图 50 所示。



图 49 智能小车靠近中心图



图 50 智能小车回转图

智能小车返回前行, 向右转弯前进, 如图 51 所示。智能小车继续直行, 遇到死胡同 (4, 5) 后, 返回再根据中心法则区域 4 算法向左转弯, 如图 52 所示。

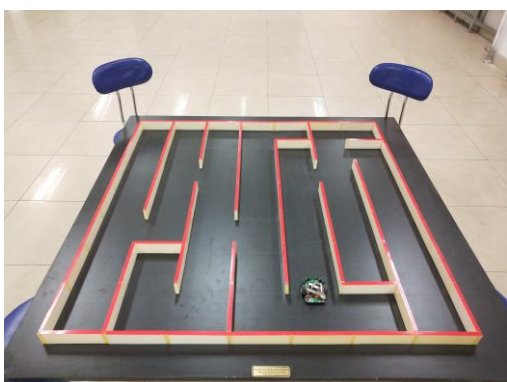


图 51 智能小车向右转弯图



图 52 智能小车向左转弯图

在区域 4 中, 智能小车继续向左转弯, 如图 53 所示。之后直行再转弯, 到达终点 (6, 6), 结束整个探索的过程, 如图 54 所示。



图 53 区域 4 向左转弯图



图 54 智能小车到达终点图

(2) 自动选择最优路径

通过从起点到终点局部迷宫的探索，已经构建好加权等高图，ARM 智能小车将自主判断并选择从起点到终点的最优路径。

ARM 智能小车从起点 $(0, 0)$ 出发，如图 55 所示。到达第一个选择口 $(0, 2)$ 点，如图 56 所示。智能小车不再直行，而是直接向右转，如图 57 所示。

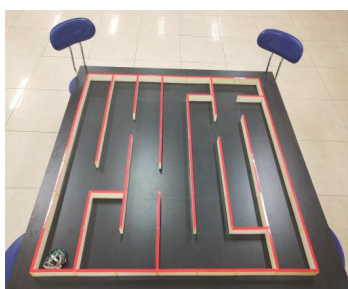


图 55 起点图

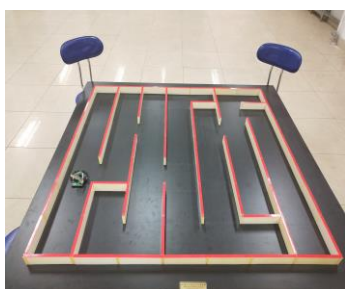


图 56 路口图

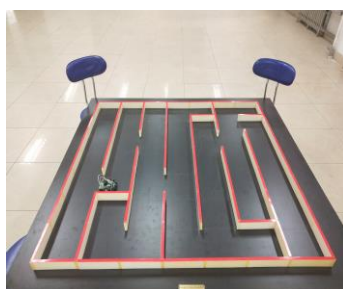


图 57 右转图

紧接着，ARM 智能小车继续直走，到达 $(1, 4)$ ，如图 58 所示。之后直接右转，如图 59 所示。根据加权等高值，智能小车再往右转直行，如图 60 所示。

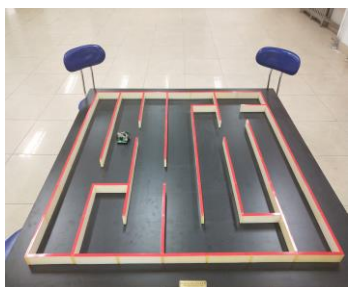


图 58 直行图



图 59 右转图

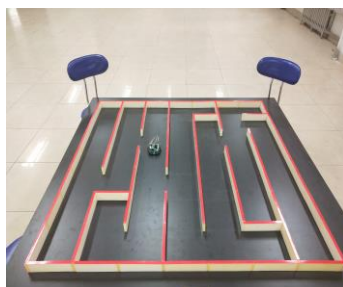


图 60 右转直行图

智能小车到达 $(2, 2)$ 点选择口，如图 61 所示。选择最优路径，直接左转，如图 62 所示。智能小车左转后，再往右转为等高值最小，如图 63 所示。



图 61 选择口路径图

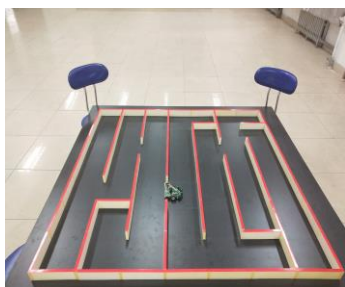


图 62 选择路径左转图

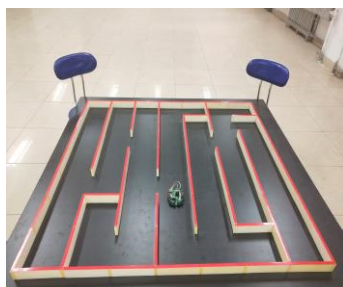


图 63 选择路径右转图

当到达 $(3, 0)$ 点时，ARM 智能小车向左转，如图 64 所示。根据等高图的等高值，智能小车将周围的等高值进行对比，判断最小等高值，选择最小等高值的路径，再次左转，如图 65 所示。左转之后，智能小车继续直行，当到达点 $(4, 4)$ 时，智能小车遇到选择口，如图 66 所示。



图 64 智能小车左转图

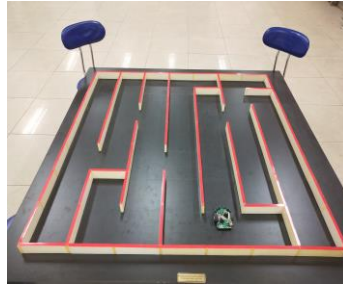


图 65 连续左转图



图 66 选择路口图

接下来，智能小车直接右转，如图 67 所示。面对两条路径选择，智能小车比对等高值，选择最小的等高值，直接左转行驶，如图 68 所示。紧接着，智能小车直行，在（5，5）点时，智能小车根据等高值进行判断，如图 69 所示。选择最小等高值，之后直接右转直行到达终点，如图 70 所示。



图 67 直接右转图



图 68 直接左转图



图 69 判断路径图



图 70 终点图

6 总结

该 ARM 智能小车控制系统，利用嵌入式微控制器、机电运动、一体化红外传感器和 PID 控制算法等技术，完成了自动行驶、自动避障和自动搜索道路，选择最优路径等功能。比较系统地介绍了硬件的组成及设计方法。利用嵌入式 C 语言完成了系统软件的设计。

该设计的主要特点：

(1) 以 Cortex-M3 内核微控制器 LM3S615 作为控制核心，丰富的外设，快速中断响应，高效管理各类事件；

(2) 高性能低功耗德国进口 1717SR 空心杯直流电机，配合一体化 1024 线编码器，实现高精度速度和位置控制；

(3) 六组无干扰独特创新红外数字式收发传感器，准确检测智能小车周围墙壁数据，进行智能避障；

(4) 高精度 MPU-6050 陀螺仪模块，测量智能小车转弯角速度，实现智能小车转弯的精确控制；

(5) 整个系统以图论算法为基础，完成自动搜索和最优路径选择功能。系统软硬件搭配合理，设计、开发、维护方便，性价比高。

ARM 智能小车控制系统具有精确的判断能力，快速的行走能力和优秀的避障能力，可以促进无人驾驶汽车的发展，进行模拟的开发与试验。可以更好地解决能源短缺、环境污染、交通拥堵、事故频发等问题，在道路上实现自动驾驶，更好的方便人们的出行与安全。

但该系统在设计和实现过程中仍有较多问题和漏洞，实时性和灵敏度没有达到预期的结果，并且在程序编写的过程中出现了许多错误，通过查阅大量的嵌入式系统资料才得以解决。通过这次毕业设计，本人学到了很多专业性的知识，也明白了坚持的重要性，面对困难，想的应该是去解决，而不是去逃避。

参考文献

- [1] 崔胜民. 智能网联汽车新技术[M]. 北京:化学工业出版社,2016:1-3
- [2] 杜永博,李萌,樊跃新,何婧. 无人驾驶技术在低速环境下的智能车应用研究[J]. 技术与市场. 2017,24(06): 346
- [3] 解 俊. 基于 ARM 的智能小车控制软件系统的设计与实现[D]. 电子科技大学,2014: 10-15
- [4] 吕美高,刘国权. 基于 STC89C52 单片机的 GPS 定位智能小车[J]. 电子质量. 2018(06): 36-40
- [5] 古训,郑亚利. 基于 MK60 单片机的微缩型无人驾驶智能车设计与实现[J]. 贵阳学院学报(自然科学版). 2017,12(04): 80-85
- [6] 周立功. LM3S615 微控制器数据手册 [DB]. <https://www.docin.com/p-377162029.html>, 2007-03-31
- [7] 陈二阳,张修军,袁姜红. 基于图像识别的自动驾驶智能小车的设计与实现[J]. 现代电子技术. 2018,41(18): 162-165+169
- [8] DR.FRITZ FAULHABER GMBH. EN_1717_SR_DFF[EB/OL].Technical Information, 2011- 2012
- [9] 潘晓贝. 基于 ARM 嵌入式的智能小车的控制系统设计研究[J]. 时代农机. 2018,45(03): 136-137
- [10] 梁明亮,孙逸洁. 嵌入式智能小车的设计与实现[J]. 制造业自动化. 2012,34(22): 87-89+94
- [11] Everlight Electronics Co.,Ltd. IRLM-8601S[EB/OL].<http://www.everlight.com>, 2004- 08- 19
- [12] 林粤伟,牟森,姜海君. 基于 ARM 和嵌入式 Linux 的智能小车技术实践[J]. 计算机系统应用. 2018,27(08): 241-246
- [13] 侯益坤,侯聪玲,刘益标. 基于 ARM 嵌入式的智能小车的控制系统设计[J]. 机电工程技术. 2010,39(01): 21-23+105
- [14] InvenSense Inc. MPU6050[EB/OL].<http://www.invensense.com>, 2011- 10- 24
- [15] 董宗祥,石红瑞,杨杰. 嵌入式智能小车测控系统的设计与实现[J]. 计算机测量与控制. 2010,18(02): 357-359+362
- [16] 石 亿. 嵌入式智能小车运动控制系统的研制[D]. 湘潭大学,2012: 2-35
- [17] 刘崇翔. 基于 ARM 的智能小车的设计与研究[D]. 江南大学,2012: 35-45
- [18] Maxim Integrated Products. max4427[EB/OL].<http://www.alldatasheet.com>, 1993- 03- 01

- [19] TEXAS INSTRUMENTS. cd74hct74[EB/OL].<http://www.ti.com>, 2003- 09
- [20] Zetex Semiconductors plc. zxmhc3a01t8[EB/OL].<http://www.zetex.com>, 2004
- [21] 张晓明. 基于嵌入式的智能小车设计[J]. 现代制造技术与装备. 2016(06): 73-74
- [22] 段建民,石慧,刘丹,于宏啸. 无人驾驶智能车导航系统控制研究[J]. 计算机仿真. 2016,33(02): 198-203+442
- [23] TEXAS INSTRUMENTS. SN74LVC245A[EB/OL].<http://www.ti.com>, 2013- 03
- [24] 邢晓敏,杨正祥. 嵌入式智能小车设计[J]. 自动化应用. 2017(05): 62-64
- [25] Sipex Corporation. SPX1117[EB/OL].<http://www.maxlinear.com>, 2004- 11- 29
- [26] 傅海军,段少飞,李炫君. 复杂路况下智能小车的设计[J]. 电子产品世界. 2018,25(10): 36-38+49

致 谢

本次毕业设计和论文在天津商业大学宝德学院计算机与信息技术系完成。借此机会，我要感谢天津商业大学宝德学院计算机与信息技术系常庆英老师提供实验环境；感谢所有给予转载和引用权的资料、图片、文献、研究思想和设想的所有者；同时特别感谢我的论文指导教师董晶老师，她以渊博的专业知识、严谨的治学态度和平易近人的生活作风深深地感染了我。在本次课题研究阶段中，董老师始终以严格的要求和孜孜不倦的教诲对我悉心地指导，培养了我分析问题、解决问题的能力 and 做事认真踏实的态度。在生活上也给予了我无微不至的关怀和帮助。在这里我向她表示我们最衷心的感谢和敬意！