

成 绩: \_\_\_\_\_

指导教师: \_\_\_\_\_

## 电磁循迹智能车的设计和实现

计算机科学与技术二班      2020416240      彭清元

指导教师 黄宝贵      依托课程 单片机原理与应用

**摘要：**本文是基于 STC16F 单片机结合 512 编码器、S3010 舵机、10 路 ADC 口、ICM20602 等传感器，通过编码器获得速度之后，利用增量式 PID 进行速度反馈得到稳定的运行速度，将 ADC 采集出的数据进行转换后结合差比和差算法实现自动循迹功能，利用蓝牙、上位机等第三方模块进行辅助调参。硬件模块主要包括电压模块，电驱模块，放大器模块。电压模块给各个功能进行稳定供电，电驱模块采用 MOS 管组成全桥驱动直流电机对 PWM 输出更加稳定，放大器模块将 ADC 采集的电磁信号放大处理。在优化过程中引入阿克曼转向原理，实现双电机进行差速控制。

**关键字：**智能小车 电磁寻迹 PID 控制 阿克曼转向 差比和差算法

### 引言

随着科学技术的不断发展进步，人工智能迅速崛起，人工智能不断地参与到生活的方方面面，在响应社会的进步，作为在校的大学生追随时代的发展，研究基于电磁传感器控制小型智能车的自动循迹过程，为以后更加深入了解无人驾驶，人工智能这一块领域奠定基础<sup>[1]</sup>。

## 1 系统总体方案设计

### 1.1 车辆设计思路

智能车整体运行主要由三个部分组成：检测系统，控制系统，动力系统。首先是检测系统采用的是工字型 10mH 电感作为传感器，用于采集电磁信号。其次控制系统是采用 STC16F 单片机作为主控芯片，动力系统采用的是 512 线编码器获取速度信息，利用 PID 控制进行速度控制。整体的运行过程为，通过电感采集赛道的电磁信息，将电磁信息转换成数字信息传送到 STC16F 主控芯片上，根据当前的位置和状态做出相应的调整，计算出当前的 PWM，输出给控制电机进行运转，从而实现转向、加减速、识别不同元素的速度控制。为了更好了解智能车内部程序的运行，设计了系统流程图，如图 1.1 所示。在为了取得更好的效果，在设计过程中因该尽量的简化硬件结构，降低硬件出错的可能性。

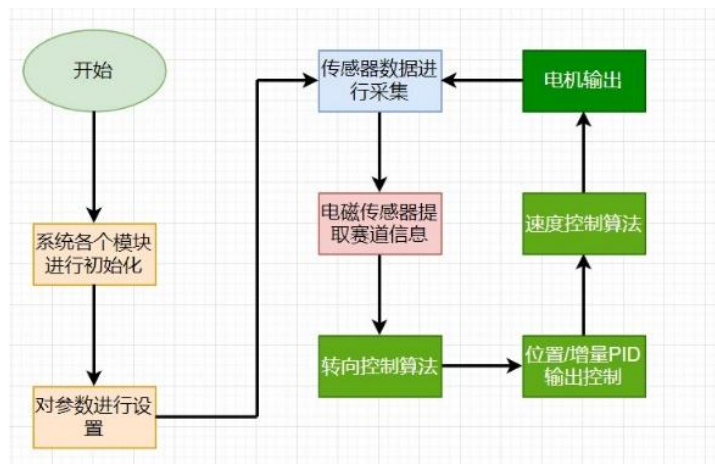


图 1.1 系统结构图

## 1.2 整车布局

- (1) 为降低车的整体重心，将主控板置于车模上。
- (2) 在小车高速循迹下，电感会剧烈上下摇晃，为避免电感上下摇晃导致数据不准确，设计了碳素杆支架，用碳素杆连接成三角形排列，更加稳定。
- (3) 为了在高速下能够更快的做出反应，并且使其有效测量更长，将电感固定在碳素杆上，如图 1.2 所示。



图 1.2 整车布局图

## 2 机械系统设计及实现

### 2.1 编码器及齿轮咬合调整

选用 512 编码器进行速度的获取。512 编码器用螺钉通过支架固定在后轮支

架上，固定好了之后，紧密咬合，增大了测速的精确性，但是过度咬合会一定程度

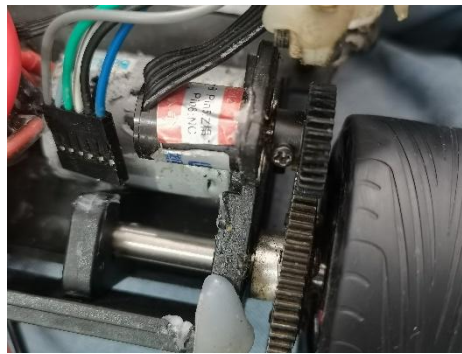


图 2.1 编码器安装

度上增大摩擦力，对齿轮有一定的损坏，为了避免噪点对速度控制的影响，在软件方面对获取的速度进行滤波处理，如图 2.1 所示。

## 2.2 电感的设计与安装

为了能够获取更多的信息，使用了双排电感进行采集信息，当第一排的信息丢失，利用第二排电感的信息进行补充，在正常的情况下分别给前后排电感赋予权重比，进行对获取的信息处理。在高速运行或者上下坡道的时候，避免电感长度过长而导致的摆动或者上下跳动，使用三通和两个小型碳素杆进行固定成三角形形状。安装如图 2.2 所示。

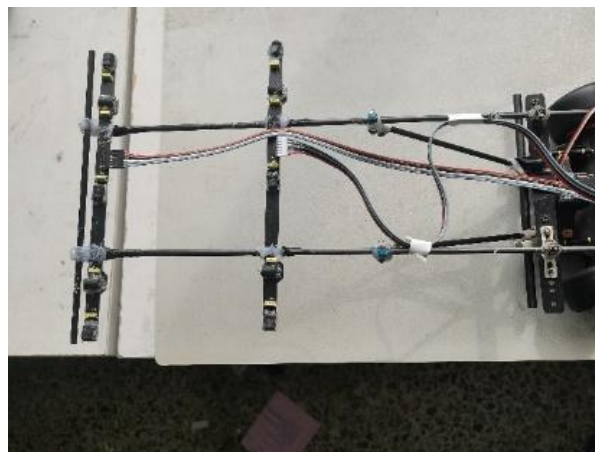


图 2.2 电感的安装

## 2.3 ICM20602 陀螺仪安装

为了能够更加精确的获取当前小车运行状态，在舵机上安装型号为 ICM20602 陀螺仪时时获取陀螺仪的角速度，从而计算出小车转向的角度，反馈

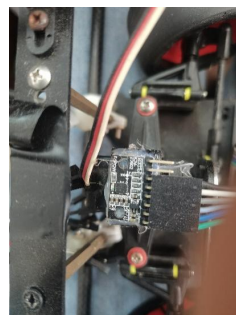


图 2.3 ICM20602 的安装

给位置 PID，对小车转向做一个矫正作用。ICM20602 传感器的安装如图 2.3 所

示。

### 3 硬件系统设计与实现

本车硬件系统设计为主控、电机驱动、运放分板，主要分为 CPU 模块，电源管理模块，驱动模块，放大器模块。

#### 3.1 CPU 模块

本车系统以 STC16F 为控制芯片，将电感、编码器、陀螺仪、蓝牙等传感器采集到的信号传入芯片，根据编写的算法进行控制，驱动直流电机完成运动轨迹。STC16F 系列单片机是不需要外部晶振和外部复位的单片机，是以超强抗干扰、超低价的一款单片机。STC16F 包括了复位电路、3.3V 稳压电路、按键、液晶接口等。使用逐飞科技智能车专用的开源库，能够进行多模块的开发，如图 3.1 所示。

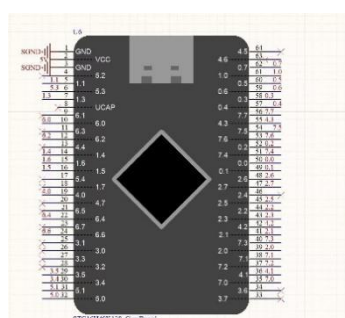


图 3.1 控制芯片

#### 3.2 电源管理模块

电源模块是为系统中各个模块提供电源，设计时不仅需要考虑用到的电器电压，电流等基本参数，还需要降低噪流，提高电源的转换效率和供电的稳定性，这将关系到小车在高速运行过程中的稳定性和传感器的灵敏性。电路接口及开关电路图如图 3.2 所示<sup>[2]</sup>。

#### 3.3 驱动模块

电机驱动模块是智能车的动力系统，选用了 MOS 管组成全桥驱动直流电机，MOS 管型号 IR7843，具有集成度高、外接电路少、内阻小等特点。为小车提供强有力的动力保障，电压稳定输出。具体电路如图 3.3 所示：

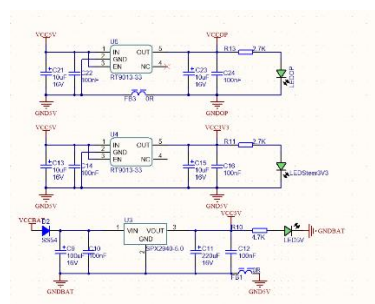


图 3.2 电源模块

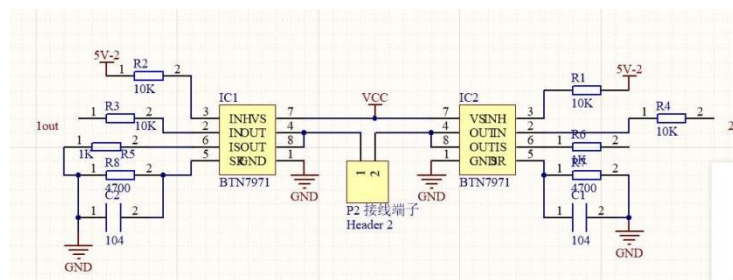


图 3.3 电机驱动模块

### 3.4 放大器模块

经电磁传感器采集后的信号不能直接使用，需要经过集成运算放大器的进一步放大后才能输入单片机中进行分析处理，采用的运放芯片为 OPA2350, 此芯片相较于 OPA4377 性能更加稳定，其共有两路集成运放器，选用 5 个芯片以满足 10 个通道电磁传感器，可通过调节滑动变阻器修改电磁信号的放大倍数，进而修改采集到的信号值，如图 3.4 所示。

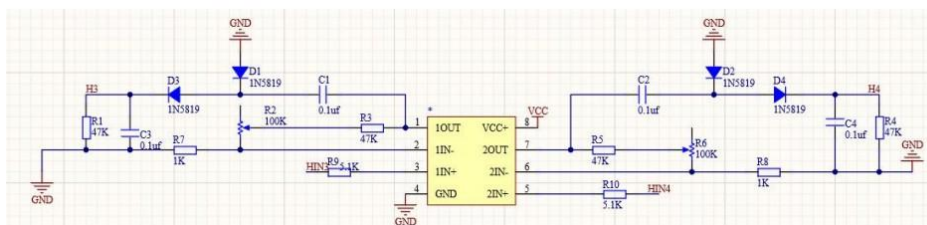


图 3.4 放大器模块

## 4 软件系统设计与实现

高效的软件设计是智能车最为核心的一部分，高速运行下，对算法的复杂度以及判断条件的精密性都会影响小车正常循迹。此程序基于 STC16F 单片机，采用 10 个 ADC 通道进行电磁读数，将采集到的电磁信号通过转化成数字信号进行使用，通过位置 PID 算法计算出当前对应的脉冲数量输送给舵机转向，结合增量 PID 提供动力系统，协调进行正常循迹，主流程图如图 4.1 所示。

### 4.1 电磁识别系统的设计

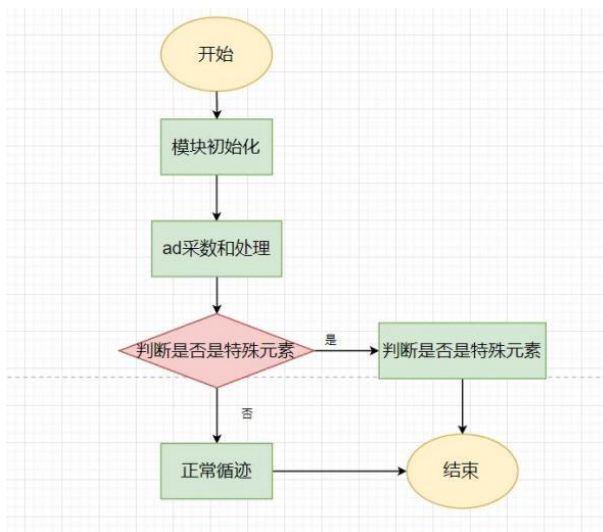


图 4.1 主程序流程图



#### 4.1.1 电磁信号的采集

采用了 10 个 12 位的 ADC 进行采集信号，双排电感来回切换读数，将电磁线产生的电磁信号通过 ADC 口进行采集，通过 OPA2350 放大器放大，通过 AD 功能换成数字信号，传送到 STC16F 核心板上进行存储。

#### 4.1.2 电磁信号的处理

将存储在核心板上的数字信号，多次采集，在采集过程中不可避免有一些噪点，因此采用较为常见的均值滤波进行处理这些噪点，将一些无效值去除，重新存储，将这些数据进行位置 PID 控制，如图 4.2 所示。



图 4.2 处理之后的电感值

### 4.2 方向和速度控制

智能车主要采用增量式 PID 控制和位置式 PD 控制，使智能车能够在赛道上平稳快速的行驶。

#### 4.2.1 方向控制

C 车模的方向控制是依靠 S3010 舵机进行的，电感通过采集赛道上的电磁信号，为了解决电磁噪点影响，选用 12 位 ADC 通道进行采数，采数之后进行归一化处理，最后将归一化好的数据进行滤波处理，得到较为可信的数据。通过差分和差算法进行计算出与舵机中值的偏差值，如果位置 PID 计算出的角度小于目标值则向左打角，反之向右打角。

#### 4.2.2 速度控制

C 车模是带有双电机的车模系统，可以实现前后左右自动运行，在进行速度控制过程中，使用的是传统的增量 PID 进行控制，在调节 PID 参数过程中，借助蓝牙、上位机等设备进行观测调试效果，最终实现闭环操作，能够实现快速的加减速操作，在实现阿克曼差速模型中提供动力如图 4.3 所示。

#### 4.2.3 阿克曼转向原理

阿克曼转向是一种现代汽车的转向方式，在汽车转弯的时候，内外轮转过的角度不一样，内侧轮胎转弯半径小于外侧轮胎。智能车在过弯的时候实现两轮的差速有利于快速过弯，于是将阿克曼模型和舵机结合使用<sup>[3]</sup>。

阿克曼转向核心公式：

$$\cot \beta - \cot \alpha = \frac{w}{L} \quad (1)$$

其中， $\beta$ ：汽车前外轮转角， $\alpha$ ：汽车前内轮转角， $w$ ：两主销中心距， $L$ ：轴距

通过数学公式计算：

$$L_{in} = R - \frac{W}{2} \quad (2)$$

$$L_{out} = R + \frac{W}{2} \quad (3)$$

$$\tan \beta = \frac{w}{L_{out}} \quad (4)$$

$$\tan \theta = \frac{L}{R} \quad (5)$$

阿克曼车模在转向的时候其实是在做圆周运动，这个圆心就和曲率有关<sup>[4]</sup>，因此两个轮子的角速度因该是一致的所以可以计算出：

$$\frac{V_{in}}{V_{out}} = \frac{L_{in} * w}{L_{out} * w} = \frac{R - \frac{W}{2}}{R + \frac{W}{2}} \quad (6)$$

假设当前设定的速度是两轮中心的速度，那么有：

$$\frac{V_s}{V_{in}} = \frac{R}{R - \frac{W}{2}} \quad (7)$$

$$\frac{V_s}{V_{out}} = \frac{R}{R + \frac{W}{2}} \quad (8)$$

所以得出两轮的速度分别为：

$$V_{in} = \frac{R - \frac{W}{2}}{R} * V_s \quad (9)$$

$$V_{out} = \frac{R + \frac{W}{2}}{R} * V_s \quad (10)$$

通过计算 R、W 和  $V_s$  参数可以实现两个轮子的差速控制

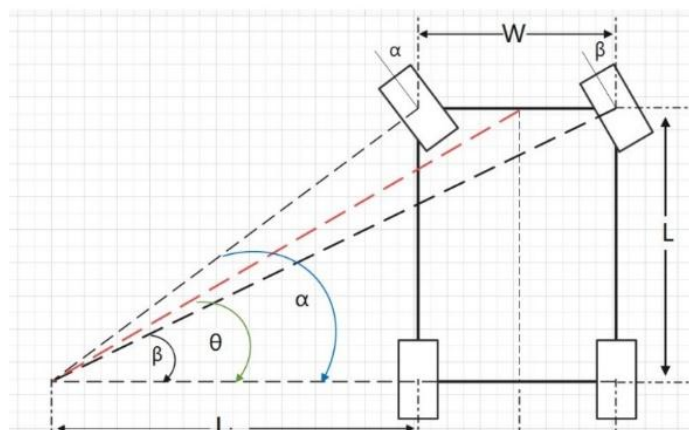


图 4.3 阿克曼转向模型

### 4.3 特殊元素判断系统设计

通过位置解算法进行判断当前的位置是处于特殊元素位置还是处于正常循迹,比如当中间电感示数大于预先设定的阈值时,再进一步判断其他电感的示数大小,如果水平电感示数之和同时大于预先设定的示数时,可认为此处是环岛标志处 如图 4.4 所示,当判断到了特殊元素位置之后,进入相应的子函数进行下一步处理。

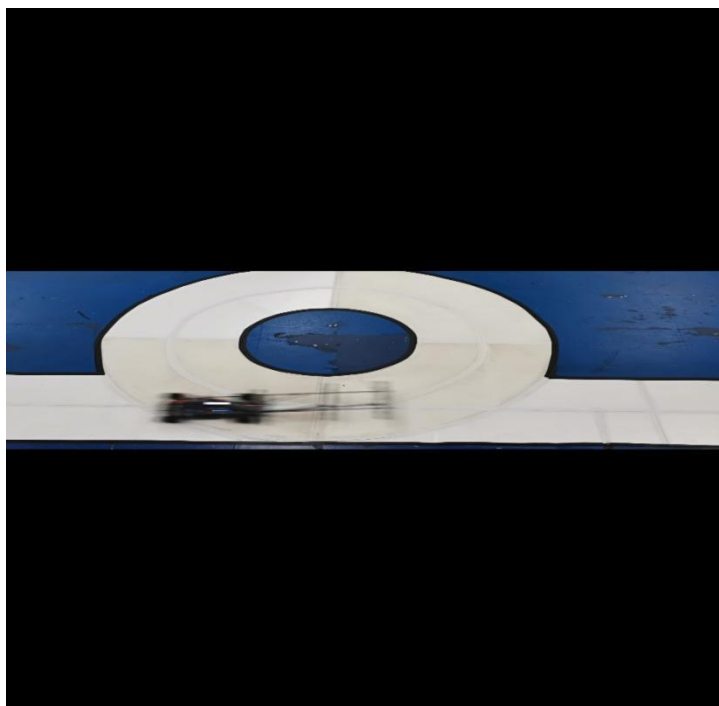


图 4.4 经过圆环



图 4.5 经过圆环液晶显示

如果是在正常循迹过程，那么进入正常循迹的子函数，在正常循迹过程中不断进



行检测特殊元素的标志点，进行来回转换如图 4.5 所示。

#### 4.4 路径检测算法

电磁传感器通过 ADC 模块采集电压转换成对应的数字信号，根据数字信号的不同进行差比和差处理判断智能车的位置。差比和差算法推到过程如下：其中 L：左边水平电感 R：右边水平电感 ML：左边竖直电感 MR：右边竖直电感 Mid：中间电感 A、B、C、limit 为参数，Err 为误差值，差比和差公式：

$$\text{Err} = \frac{(A * (L - R) + B * (ML - MR))}{(A * (L + R) + B * (ML + MR))} * \text{limit} \quad (11)$$

但是在使用过程中发现在过弯的时候非常外切，于是进行改良：

$$\text{Err} = \frac{(A * (L - R) + B * (ML - MR))}{(A * (L + R) + B * (ML - MR))} * \text{limit} \quad (12)$$

接下来在调试过程中发现在出拐角的时候会自动的进行回扣，再次改良之后的结果<sup>[5]</sup>：

$$\text{Err} = \frac{(A * (L - R) + B * (ML - MR))}{(A * (L + R) + B * (ML - MR + \text{Mid}))} * \text{limit} \quad (13)$$

#### 4.5 增量 PID 调试

增量 PID 是速度控制的核心，PID 本质是经过负反馈调节，不断输入输出的结果，自动计算出符合预先的结果。PID 实指“比例 proportional”、“积分 integral”、“微分 derivative”，这三项构成 PID 基本要素。每一项完成不同任务，对系统功能产生不同的影响。它的结构简单，参数易于调整，是控制系统中经常采用的控制算法<sup>[6]</sup>。代码实现如图 4.6 所示

```
double PID_Increase1(double ActualSpeed,double SetSpeed)
{
    double ierror;

    ierror = SetSpeed - ActualSpeed;

    pidincl = kp_1 * ( ierror - pidl.last_error )
              + ki_1 * ierror
              + kd_1 *(ierror - 2 * pidl.last_error + pidl.pre_error);

    pidl.pre_error = pidl.last_error;

    pidl.last_error=ierror;

    pidl.last_speed=ActualSpeed;

    return pidincl;
}
```

图 4.6 增量 PID

#### 4.7 蓝牙调试和上位机组合

在传统调试过程中是进行手动调试参数，由于此方法过于繁琐，于是采用了上位机和蓝牙组合的形式进行调参，如图 4.7 所示，实现蓝牙和单片机的通信，将数据结果显示在上位机上。首先将 STC16F 单片机上的 RX，TX 接口和蓝牙的 TX，RX 对应连接，将蓝牙模块和 USB 转 TTL 进行配对使用，然后实现软件上的通信，数据如图 4.8 所示。

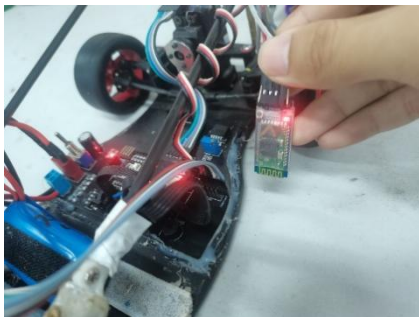


图 4.7 蓝牙连接

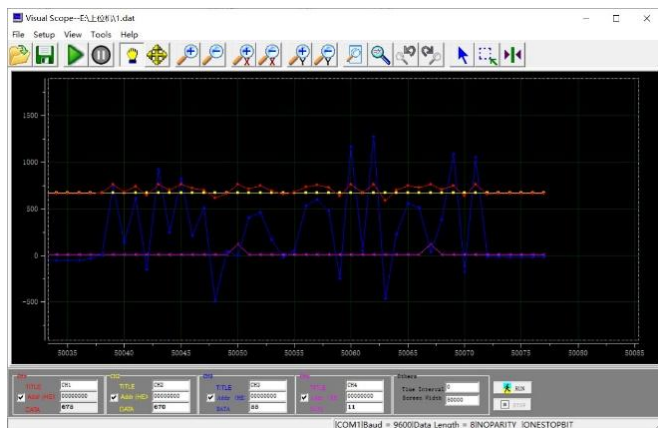


图 4.8 上位机数据显示

## 5 基于传统 PID 改良的模糊控制

### 5.1 传统 PID

普通的 PID 结构简单、稳定性好、可靠性较强，成为当代控制算法的主流，但是普通的 PID 控制在非线性的变化中，系统的鲁棒性不强，因此达到预期结果并不理想，传统 PID 主要是由比例、积分、微分组成，通过比例增大反应速度，通过积分作用消除误差，而微分控制可缩小超调量，三者共同控制，构成一个负反馈调节，流程如图 5.1 所示。

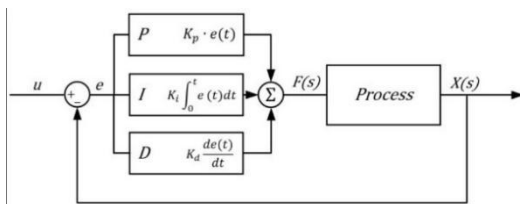


图 5.1 PID 控制过程流程

### 5.2 模糊 PID 控制

基于传统 PID 的鲁棒性不强，于是在原来的基础上引入模糊控制，模糊控制在当下实际应用中已经十分常用，如家用空调、智能洗衣机、吸尘器等。模糊控制是基于智能推理算法，逐步求精的思想，一个完整的模糊算法主要是由模糊化、模糊推理机、清晰化三个方面构成的<sup>[6]</sup>。

#### 5.2.1 输入量的量化

输入量的量化就是将输入量通过量化函数进行映射成一定的数字级别，一般

而言都是相对 0 的对称区间。首先引入 e 和 ec 模糊集对应的论域，定义{-6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6}。对于任何一个输入量都会有一个范围，预先标记为 $V_{max}$ 和 $V_{min}$ ，在进行 PID 进行调节的时候设定值的范围的预期一致。此处的量化采用线性函数进行处理：

$$f(e) = \frac{6 * e}{V_{max} - V_{min}} \tag{13}$$

$$f(ec) = \frac{6 * ec}{2 * (V_{max} - V_{min})} \tag{14}$$

### 5.2.2 输入量的模糊化

当收集了 e 和 ec 的模糊子集,对于区间的范围规定为: 负大[NB]、负中[NM]、负小[NS]、零[Z0]、正小[PS]、正中[PM]、正大[PB]等七个变量进行描述模糊范围，所以定义 e 和 ec 的模糊子集位：{NB, NM, NS, Z0, PS, PM, PB}。正常的隶属度是在 0-1 之间的一个值，通过这个值来描述变量的模糊程度，在这里采用线性的隶属度函数，如图 5.2 所示：

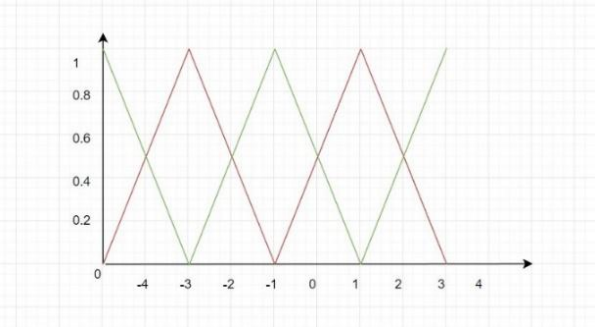


图 5.2 隶属度函数图像

### 5.2.2 建立模糊化表

在传统 PID 中，比例的取值取决于响应速度，增大比例能够提高反应速度，但是当比例过大的时候会发生超调现象，导致系统震荡严重，为了减小震荡，适当的将比例减小，当比例过小的时候，会导致反应速度过慢。因此设定比例的模糊表的规则：当调节初期因该取较大值，在调节中期取中间值，在调节末期取较小值，最终达到具有较小的超调并且保证了反应速度较快的效果，其中水平表示 ec，竖直表示 e，具体规则表 5-1：

表 5-1 比例模糊规则

$\Delta K_p$	NB 负大	NM 负中	NS 负小	Z0 零	PS 正小	PM 正中	PB 正大
NB 负大	PB 正大	PB 正大	PM 正中	PM 正中	PS 正小	Z0 零	Z0 零
NM 负中	PB 正大	PB 正大	PM 正中	PS 正小	PS 正小	Z0 零	NS 负小
NS 负小	PM 正中	PM 正中	PM 正中	PS 正小	Z0 零	NS 负小	NS 负小
Z0 零	PM 正中	PM 正中	PS 正小	Z0 零	NS 负小	NM 负中	NM 负中
PS 正小	PS 正小	PS 正小	Z0 零	NS 负小	NS 负小	NM 负中	NM 负中
PM 正中	PS 正小	Z0 零	NS 负小	NM 负中	NM 负中	NM 负中	NB 负大
PB 正大	Z0 零	Z0 零	NM 负中	NM 负中	NM 负中	NB 负大	NB 负大

在 PID 调节过程中，积分控制主要是用来消除稳态偏差，在调节过程中可能产生积分饱和，导致在调节过程中发生超调现象，所以在调节初期，为了避免积分过饱和，积分因适当小一点，在调节中期，积分作用适中，在调节末期，因提

高积分效果，减小稳态偏差，其中水平表示  $e_c$ ，竖直表示  $e$ ，具体积分的模糊具体见表 5-2：

表 5-2 积分模糊规则

$\Delta K_i$	NB 负大	NM 负中	NS 负小	ZO 零	PS 正小	PM 正中	PB 正大
NB 负大	NB 负大	NB 负大	NM 负中	NM 负中	NS 负小	ZO 零	ZO 零
NM 负中	NB 负大	NB 负大	NM 负中	NS 负小	NS 负小	ZO 零	ZO 零
NS 负小	NB 负大	NM 负中	NS 负小	NS 负小	ZO 零	PS 正小	PS 正小
ZO 零	NM 负中	NM 负中	NS 负小	ZO 零	PS 正小	PM 正中	PB 正大
PS 正小	NM 负中	NS 负小	ZO 零	PS 正小	PS 正小	PM 正中	PB 正大
PM 正中	ZO 零	ZO 零	PS 正小	PS 正小	PM 正中	PB 正大	PB 正大
PB 正大	ZO 零	ZO 零	PS 正小	PM 正中	PM 正中	PB 正大	PB 正大

为了减小超调现象，于是引入微分运算，除了在减小超调以外，微分也起到可加快调节的作用，微分的作用是抑制偏差的变化。微分的调节对动态特性影响很大，由于微分过大，调节就会超前，微分过小制动就会落后。因此对于微分的调节，在初期，加大微分的作用，中期适当减小，保持固定，在末期因稍作减小。其中水平表示  $e_c$ ，竖直表示  $e$ ，具体微分的模糊表见表 5-3：

表 5-3 微分模糊规则

$\Delta K_d$	NB 负大	NM 负中	NS 负小	ZO 零	PS 正小	PM 正中	PB 正大
NB 负大	PS 正小	NS 负小	NB 负大	NB 负大	NB 负大	NM 负中	PS 正小
NM 负中	PS 正小	NS 负小	NB 负大	NM 负中	NM 负中	NS 负小	ZO 零
NS 负小	ZO 零	NS 负小	NM 负中	NM 负中	NS 负小	NS 负小	ZO 零
ZO 零	ZO 零	NS 负小	NS 负小	NS 负小	NS 负小	NS 负小	ZO 零
PS 正小	ZO 零	ZO 零	ZO 零	ZO 零	ZO 零	ZO 零	ZO 零
PM 正中	PB 正大	NS 负小	PS 正小	PS 正小	PS 正小	PS 正小	PB 正大
PB 正大	PB 正大	PM 正中	PM 正中	PM 正中	PS 正小	PS 正小	PB 正大

### 5.2.3 清晰化

在实际引用过程中，需要用到的是清晰的比例、积分、微分，因此根据模糊表将模糊之后的数据推理出来得到最终使用的结果。这里采用的是  $\{-6, 6\}$  的论域，这里采用重心法计算个输出的量化值。计算公式，其中  $M$  是隶属度， $F$  为模糊规则表数值：

$$V_0 = \frac{\sum_{i=0}^n (M_i * F_i)}{\sum_{i=0}^n M_i} \quad (15)$$

将最终得出的数据带入到 PID 公式中进行计算，具体流程图 5.3 如下<sup>[7]</sup>：

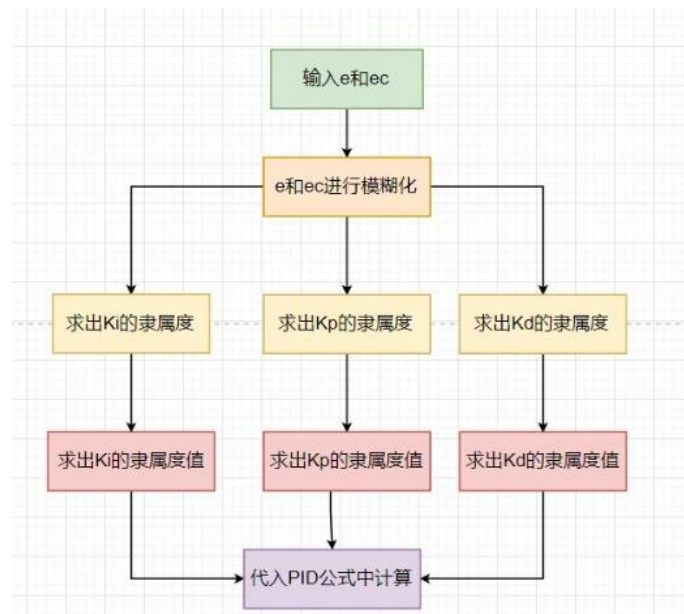


图 5.3 模糊 PID 流程

通过引入模糊 PID 的使用，十分明显的提高了小车运行鲁棒性和稳定性，反应速度一定程度上得到了极大的改良。

## 6 总结

在本文中，主要介绍了实现智能车运行过程需要实现的基本思路，包括电路、算法、传感器、蓝牙与上位机等软件以及最重要的电磁信息处理以及差比和差算法的优化，引入了模糊 PID 的处理增强了系统的鲁棒性和反应速度。在结构方面，根据 C 车模的特点进行降低地盘，S3010 舵机立式安装。在电路方面，使用了驱动直流电机、全桥驱动电路设计电驱、使用 OPA2350 芯片对放大器进行稳压输出，使得电磁性能更加稳定。在第三方模块上使用蓝牙与上位机，使用蓝牙和单片机通信，将数据采集显示在上位机上进行分析处理。在程序方面，使用 Keil5 开发环境调试 C 语言编程程序，在不断调试改进的过程中设计出了一套比较通用稳定的程序。在该套稳定的算法中，优化路径，行驶过程中加减速控制，保证能在较短时间内完成比赛。

### 参考文献

- [1] 邵贝贝. 单片机嵌入式应用的在线开发方法 [M]. 北京. 清华大学出版社. 2004.
- [2] 张鑫. 单片机原理及应用. 北京: 清华大学出版社, 2010.
- [3] 孟刚. 车辆的转向特性与阿克曼转向原理的分析[J]. 机械研究与应用, 2007(04): 36-38.
- [4] 徐立平. 基于阿克曼理论的转向梯形设计分析[J]. 机械工程师, 2017(05): 69-70.
- [5] 谭浩强著. C 程序设计. 北京: 清华大学出版社, 2003.
- [6] 仲志丹, 张洛平, 张青霞. PID 调节器参数自寻优控制在运动伺服中的应用[J]. 洛阳工学院学报, 2000, 21 (1): 57~60.
- [7] 李太福. 基于在线参数自整定的模糊 PID 伺服控制系统[J]. 交流伺服系统 2005, 4: 203~215.