

# 第七届“飞思卡尔”杯全国大学生 智能汽车竞赛 技 术 报 告



## 湖北汽车工业学院

HuBei Automotive Industries Institute

学 校：湖北汽车工业学院

队伍名称：摄像头1队

参赛队员：盛 昌

杨 超

段佳伟

带队老师：吕 科

王新超

## 关于技术报告和研究论文使用授权的说明

本人完全了解第七届“飞思卡尔”杯全国大学生智能汽车邀请赛关保留、使用技术报告和研究论文的规定，即：参赛作品著作权归参赛者本人，比赛组委会和飞思卡尔半导体公司可以在相关主页上收录并公开参赛作品的设计方案、技术报告以及参赛模型车的视频、图像资料，并将相关内容编纂收录在组委会出版论文集中。

参赛队员签名: \_\_\_\_\_

带队教师签名: \_\_\_\_\_

日 期: \_\_\_\_\_

---

## 摘要

随着数字图像处理技术的发展，面阵 CCD 摄像头在自动控制领域得到了越来越广泛的应用。

本文在对 CCD 图像处理技术和单片机嵌入式应用进行深入研究的基础上，借鉴国内外先进技术，研制出了具有自动循迹，转弯，加减速功能的智能小车。本文介绍了基于面阵 CCD 图像处理技术以及 MC9S12XS128 微控制器嵌入式技术的智能小车的设计原理及研制过程。论述了智能小车自动控制系统的实现方法，着重讨论了智能小车的硬件设计和以 Codewarrior C 交叉编译器为开发工具的软件设计。

在智能小车的研制过程中，利用 CCD 摄像头采集到的图像存在杂点和误差，这对单片机的控制影响较大，因此需要运用图像处理技术对数字图像进行处理，以达到对黑线正确识别的目的。

### 关键词：

**智能小车；CCD；MC9S12XS128 微控制器；图像处理**

## **Abstract**

With the development of treatment technology of the digital picture , plane array CCD camera has got more and more extensive application on the automatic controlled field.

This text is on the basis of the thing that use and further investigate to CCD image processing technology and single-chip computer embeddedly, learns from domestic and international advanced technology, develop and follow the mark automatically, turn, add the intelligent car which moderates the function. The introduction to this text is on the basis of plane array CCD image processing technology and the design principle of the intellectual car of MC9S12XS128 microcontroller embedded technology and research course. Have described the implementation method of the automatic control system of intellectual car , have discussed emphatically the hardware of the intellectual car is designed and as the software design of the developing instrument with Code-warrior C compiling device alternately.

In the course of research of the intelligent car , utilize the picture that CCD camera gathers to noise and error, this makes a great influence on control of the single-chip computer, so need to use the image processing technology to deal with the digital picture , in order to achieve the correct purpose that discerns to the black line.

**Keywords: Intelligentcar;CCD; MC9S12XS128; microcontroller ;**

**Image processing**

---

# 目录

摘要.....	I
ABSTRACT.....	II
第一章 引言.....	1
第二章 智能车整体设计.....	2
2.1 技术方案的实现.....	2
2.2 系统机械结构设计.....	2
2.3 智能车前轮定位的调整.....	4
2.3.1 主销后倾角.....	4
2.3.2 主销内倾角.....	5
2.3.3 车轮外倾角.....	5
2.3.4 前轮前束.....	5
2.4 其它机械结构的调整.....	6
第三章 硬件电路设计说明.....	8
3.1 系统硬件结构.....	8
3.2 系统电源模块设计.....	9
3.3 电机驱动模块设计.....	10
3.4 转速传感器电路设计.....	10
3.5 摄像头供电模块电路设计.....	11
3.6 最小系统模块.....	12
第四章 视频采集和图像处理.....	13
4.1 视频采集.....	13

4.1.1 摄像头工作原理.....	13
4.1.2 摄像头的选择.....	14
4.1.3 信号分离电路.....	14
4.2 图像处理.....	16
4.2.1 目标指引线的提取.....	16
第五章 智能车软件设计与实现.....	17
5.1 主程序流程框图.....	18
5.2 初始化.....	18
5.3 测量.....	19
5.4 舵机和速度的PID 算法.....	19
5.4.1 经典PID 控制算法介绍.....	19
5.4.2 经典PID 算法在本智能车上的应用.....	21
第六章 智能车技术参数说明.....	23
参考文献.....	24
附录：部分程序代码.....	I

# 第一章 引言

本智能小车以飞思卡尔16位微控制MC9S12XS128作为唯一的控制单元，采用一组线圈作为传感器，感应由赛道中心导线产生的交变磁场检测道路信息，通过单片机处理优化，把控制信号发送给电机和舵机。同时通过自制的转速传感器获取小车速度，进行速度反馈处理，最后利用PID控制方式作为电机驱动。本文对线圈的选择及对跑道位置的检测进行了简要的分析，对小车的硬件与软件设计进行了介绍。

技术报告以智能小车的设计为主线，包括小车的构架设计、软硬件设计，以及控制算法研究等，共分为五章。其中，第一章为引言部分。第二章主要介绍了小车的总体方案的选取。第三章介绍了硬件设计，主要介绍了电路的设计；检测模块的设计。第四章对小车的软件设计进行了详细的介绍。第五章描述了小车的软件调试过程。

## 第二章 智能车整体设计

### 2.1 技术方案的实现

介于飞思卡尔智能车比赛的特征，我们不难发现赛道多是以弯道为主，在此我们需要考虑其弯道的极限速度，为了配合响应的控制，我们在机械，硬件电路，软件调试做了相应的配比。像系统板的位置固定，传感器方案，转向机构的优化，转弯策略的调整。

系统采用主办方提供的飞思卡尔 16 位微控制器 MC9S12XS128 作为核心控制单元，采用 PWM 和 PID 技术，控制舵机的转向和电机转速。系统还扩展了液晶和键盘模块作为人机操作界面，以便于智能小车的调试与相关参数调整。此外，软件系统增加了复位快速自动重启技术，防止小车受到干扰后不能正常运行情况的出现。信号由安装在车前部的光电传感器采集，经过选频，滤波电路传入 MCU，MCU 对信号进行处理后，由 PWM 发生模块发出 PWM 波，分别对转向主舵机，电机进行控制，完成智能车的转向，前进。在智能车后轮上安有光电编码器，采集关于车轮转速的脉冲信号，经由 MCU 进行 PID 计算后自动调节对电机的 PWM 波的占空比，控制车速度。

本系统中，使用了摄像头来黑线位置的检测，从而可以进行识别路径。摄像头传感器的路径检测作为一个关键部分，对其处理的好坏直接决定了小车运行状态和动态性能。本文析了采用位置模糊控制的理论来判断黑线的位置，采用较大视野的镜头来减少盲区。本系统显示了高度的智能化、人性化，并且具备良好的安全性、稳定性与反应的灵敏性，后面的各个章节将分别与之介绍。

### 2.2 系统机械结构设计

#### 2.2.1 摄像头的安装

摄像头是整辆车的眼睛，摄像头的安装是最重要的。摄像头的安装要求使得摄像头位于整个车的中心位置而且高度要适合信号的采集和处理通过多组对比试验我们最终决定试用单杆结构来固定摄像头单杆结构使得整个车十分简洁，但也会带来摄像头易抖动的问题，我们让摄像头与车连成整体来减小抖动量。



如图 2-1 摄像头的安装：

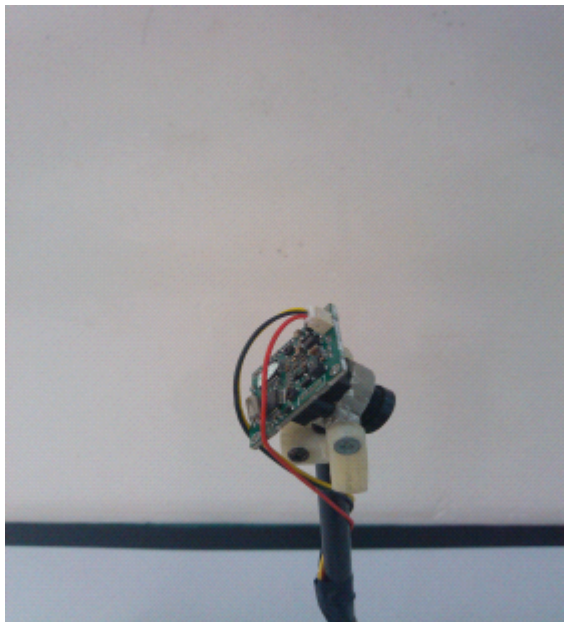


图 2.1

### 2.2.2 舵机的固定

经过我们不断测量发现，舵机摆臂长度加长有助于机械响应，但由于 FUTABA S3010 是 5KG，当舵机摆臂过长之后会导致响应下降，经试验得到一个较为理想的值。

我们舵机采用了立式架法，立式架法相对其他架法优势在于机械结构间隙较小，左右调整方便对称性好。

具体安装如图：

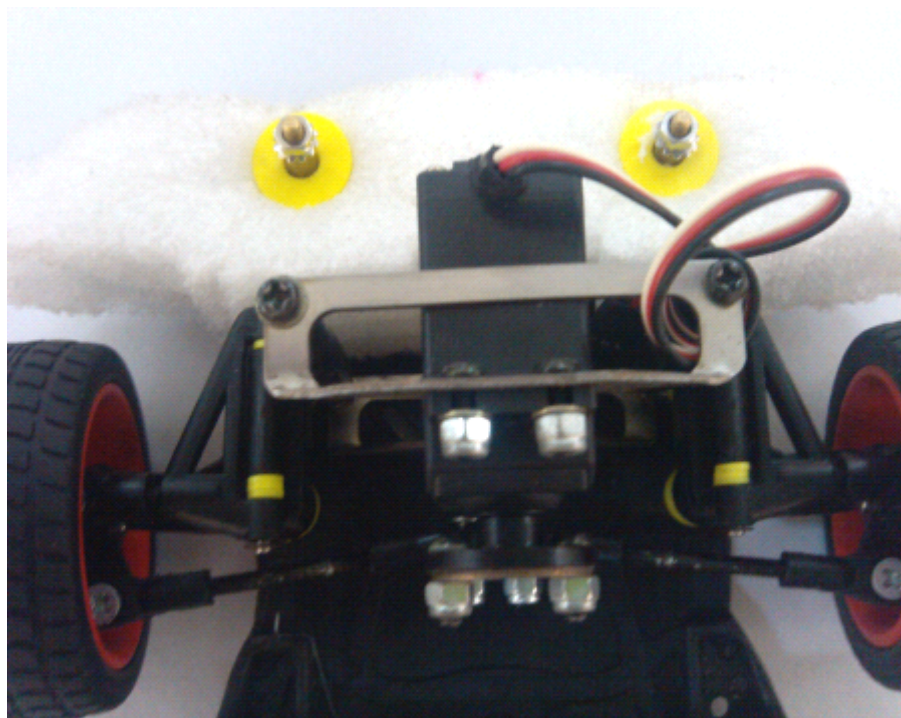


图 2.2

## 2.3 智能车前轮定位的调整

现代汽车在正常行驶过程中，为了使汽车直线行驶稳定，转向轻便，转向后能自动回正，减少轮胎和转向系零件的磨损等，在转向轮、转向节和前轴之间须形成一定的相对安装位置，叫车轮定位，其主要的参数有：主销后倾、主销内倾、车轮外倾和前束。智能车竞赛模型车的四项参数都可以调整，但是由于模型车加工和制造精度的问题，在通用的规律中还存在着一些偶然性。

### 2.3.1 主销后倾角

主销后倾角是指在纵向平面内主销轴线与地面垂直线之间的夹角。它在车辆转弯时会产生与车轮偏转方向相反的回正力矩，使车轮自动恢复到原来的中间位置上。所以，主销后倾角越大，车速越高，前轮自动回正的能力就越强，但是过大的回正力矩 黄色垫片的数量来改变主销后倾角的，由于竞赛所用的转向舵机力矩不大，过大的主销后倾角会使转向变得沉重，转弯反应迟滞，所以设置为  $0^\circ$ ，以便增加其转向的灵活性。

### 2.3.2 主销内倾角

主销内倾角是指在横向平面内主销轴线与地面垂直线之间的夹角，它的作用也是使前轮自动回正。角度越大前轮自动回正的作用就越强，但转向时也就越费力，轮胎磨损增大；反之，角度越小前轮自动回正的作用就越弱。通常汽车的主销内倾角不大于  $8^{\circ}$ 。

对于模型车，通过调整前桥的螺杆的长度可以改变主销内倾角的大小，由于过大的内倾角也会增大转向阻力，增加轮胎磨损，所以在调整时可以近似调整为  $0^{\circ} \sim 3^{\circ}$  左右，不宜太大。

主销内倾和主销后倾都有使汽车转向自动回正，保持直线行驶的功能。不同之处是主销内倾的回正与车速无关，主销后倾的回正与车速有关，因此高速时主销后倾的回正作用大，低速时主销内倾的回正作用大。

### 2.3.3 车轮外倾角

前轮外倾角是指通过车轮中心的汽车横向平面与车轮平面的交线与地面垂线之间的夹角[2]，对汽车的转向性能有直接影响，它的作用是提高前轮的转向安全性和转向操纵的轻便性[1]。在汽车的横向平面内，轮胎呈“八”字型时称为“负外倾”，而呈现“V”字形张开时称为正外倾。如果车轮垂直地面一旦满载就易产生变形，可能引起车轮上部向内倾侧，导致车轮联接件损坏。所以事先将车轮校偏一个正外倾角度，一般这个角度约在  $1^{\circ}$  左右，以减少承载轴承负荷，增加零件使用寿命，提高汽车的安全性能。

模型车提供了专门的外倾角调整配件，近似调节其外倾角。由于竞赛中模型主要用于竞速，所以要求尽量减轻重量，其底盘和前桥上承受的载荷不大，所以外倾角调整为  $0^{\circ}$  即可，并且要与前轮前束匹配。

### 2.3.4 前轮前束

所谓前束是指两轮之间的后距离数值与前距离数值之差，也指前轮中心线与纵向中心线的夹角[2]。前轮前束的作用是保证汽车的行驶性能，减少轮胎的磨损。前轮在滚动时，其惯性力自然将轮胎向内偏斜，如果前束适当，轮胎滚动时的偏斜方向就会抵消，轮胎内外侧磨损的现象会减少。像内八字那样前端小后端大的称为“前束”，反之则称为“后束”或“负前束”。在实际的汽车中，一般前束为  $0 \sim 12\text{mm}$ 。

在模型车中，前轮前束是通过调整伺服电机带动的左右横拉杆实现的。主销在垂直方向的位置确定后，改变左右横拉杆的长度即可以改变前轮前束的大

小。在实际的调整过程中，我们发现较小的前束，约束  $0\sim 2\text{mm}$  可以减小转向阻力，使模型车转向更为轻便，但实际效果不是十分明显。

虽然模型车的主销后倾角、主销内倾角、车轮外倾角和前束等均可以调整，但是由于车模加工和制造精度的问题，在通用的规律中还存在着不少的偶然性，一切是实际调整的效果为准。

## 2.4 其它机械结构的调整

1 为有效地防止由于轮胎与轮辋错位而引起的驱动力损失，我们对前轮进行了粘胎处理。

2 为了防止小车在高速时急转弯时发生侧滑甩尾等现象，在底盘下部车尾与车身也进行了粘接处理。

3 为提高小车的稳定性，我们尽量做到使小车重心降低且使小车的重心后移，如将电池后移。

制作完的小车整体布局如图：

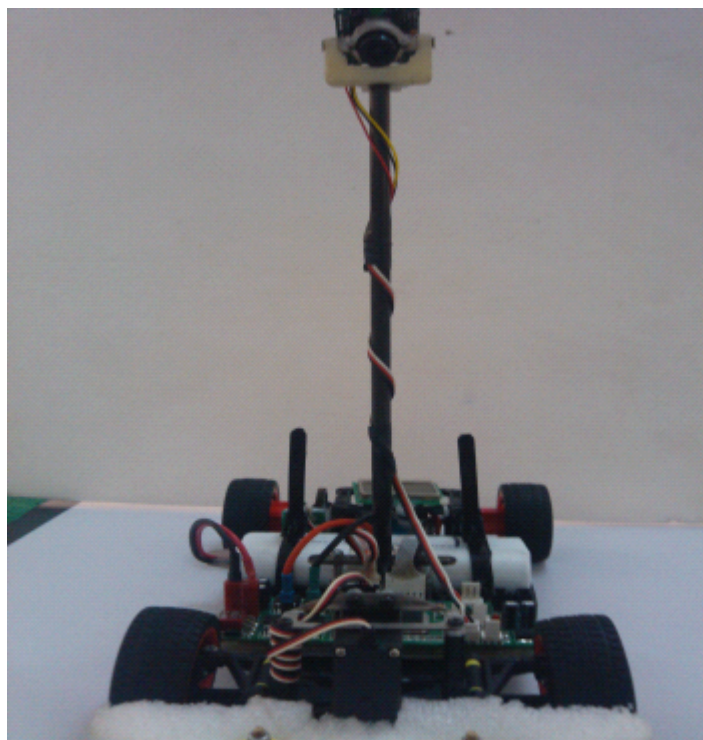


图 2.3 前视图

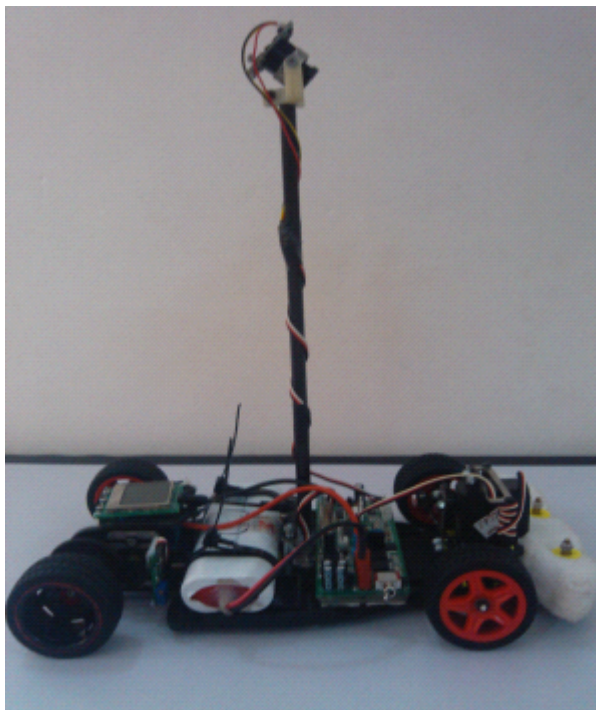


图 2.4 侧视图

## 第三章 硬件电路设计说明

### 3.1 系统硬件结构

此智能车辆定位系统用摄像头拍车辆前方的赛道，通过 MC9S12XS128 采样视频信号，获得图像数据。然后用合适的算法，如跟踪边缘检测法，分析图像数据，提取目标指引线。然后，系统根据目标指引线的位置信息，对舵机和电机施以合适的控制。本智能车定位系统的结构图如图 3-1 所示：

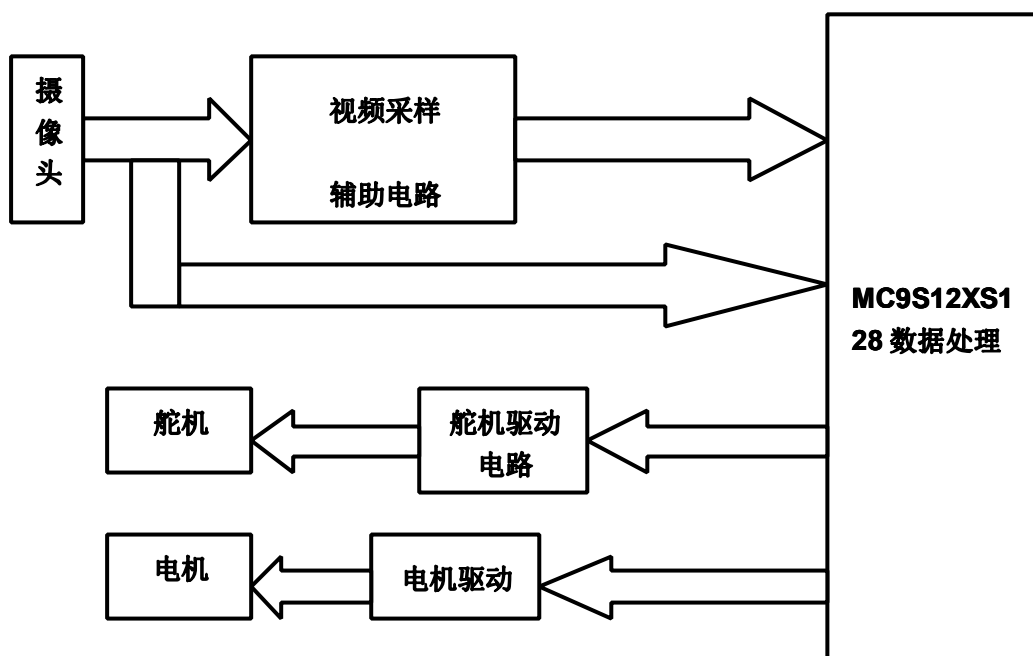


图3.1 智能车定位系统结构图

因为系统是一个有机的整体，所以任何一部分的改进都能提高小车的性能。虽然轮胎、驱动电机、舵机和电池等车模主要结构不能作改动，但是一个机械结构上的细节仍然会对小车的性能产生影响，为此我们对这些细节进行了一定的调整。系统所用的传感器包括光电（1 个用于测速）、JY043W（黑线识别）和摄像头（路径识别），它们可以完成赛道信息的采集和小车行驶参数的获取，算法部。

系统硬件位于底层，是整个系统的基础，系统软件结构则根据硬件和控制需要来制定。系统的基本软件流程为：首先，对各功能模块和控制参数进行初始化。然后，通过图像采集模块获取前方的图像数据，同时通过速度传感器模块获取小车的速度，用前排传感器获取路面的信息，采用模糊算法来控制舵机

的转向。另外根据检测到的速度，结合速度控制策略，对小车速度进行不断的调整，使小车在符合比赛规则的前提下，沿赛道快速行驶。系统的基本软件结构如图 3-2 所示。

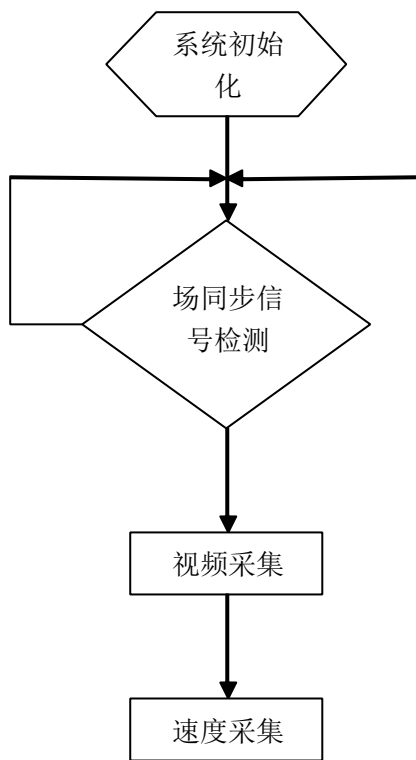


图 3.2 系统的基本软件机构

### 3.2 系统电源模块设计

全部硬件电路的电源由 7.2V, 2A/h 的可充电镍镉电池提供。由于电路中的不同电路模块所需要的工作电压和电流容量各不相同，因此电源模块应该包括多个稳压电路，将充电电池电压换成各个模块所需要的电压。主要包括如下不同的电压：

- 5V 电压。主要为单片机、信号调理电路及部分接口电路（如速度传感器）提供电源，电压要求稳定、噪声小，电流容量大于 500mA。
- 6V 电压。主要是为舵机提供工作电压。实际工作时，舵机所需要的工作电

流一般在几十毫安左右，电压无需十分稳定。

- 7.2V 电压。这部分直接取自电池两端电压，主要为后轮电机驱动模块提供电源。
- 12V 电压。采用摄像头进行路径检测时，需要12V 工作电源。整个电源模块的电路结构为：

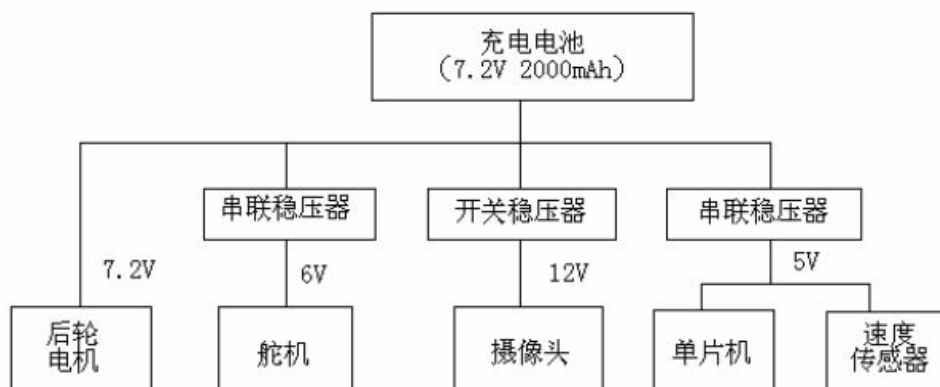


图 3.3 电源模块电路结构

### 3.3 电机驱动模块设计

根据汽车机械式差速器的原理，在过弯时，汽车内外车轮行驶的路径不同，所以长短不一，这样导致两轮的转速不一样，差速器就是通过机械结构决汽车在行驶过程中的相互干扰。我们这套车模提供的是滚珠差速器，为了让汽车具有更换的差速功能，我们通过添加润滑油，来保障小更好的行驶。

### 3.4 转速传感器电路设计

车速传感器为对射式光电传感器，具体电路如图 3.4。由光电对管采集输出为类似正弦波的信号，为使最终的车速信号为单片机所能处理的数字信号，在光电对管输出端接电压比较器 LM2903。



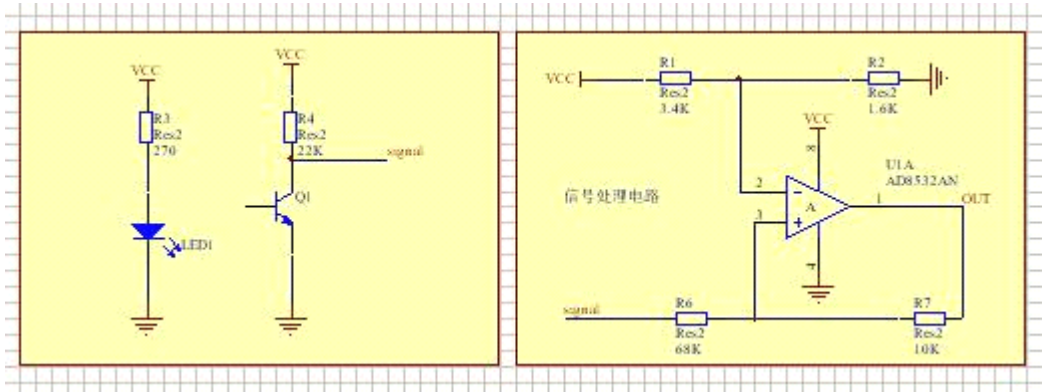


图 3.4 车速传感器电路

3.5 摄像头供电模块电路设计

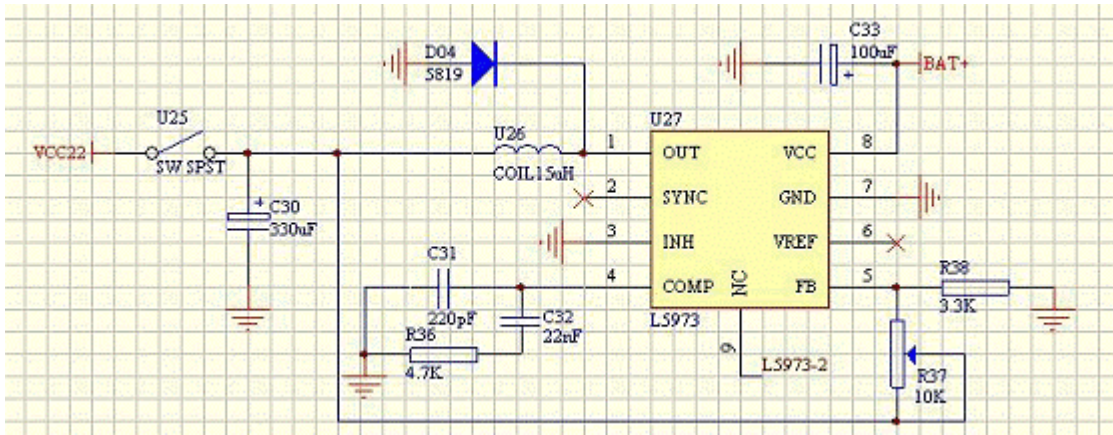


图3.5 摄像头供电模块电路

### 3.6 最小系统模块

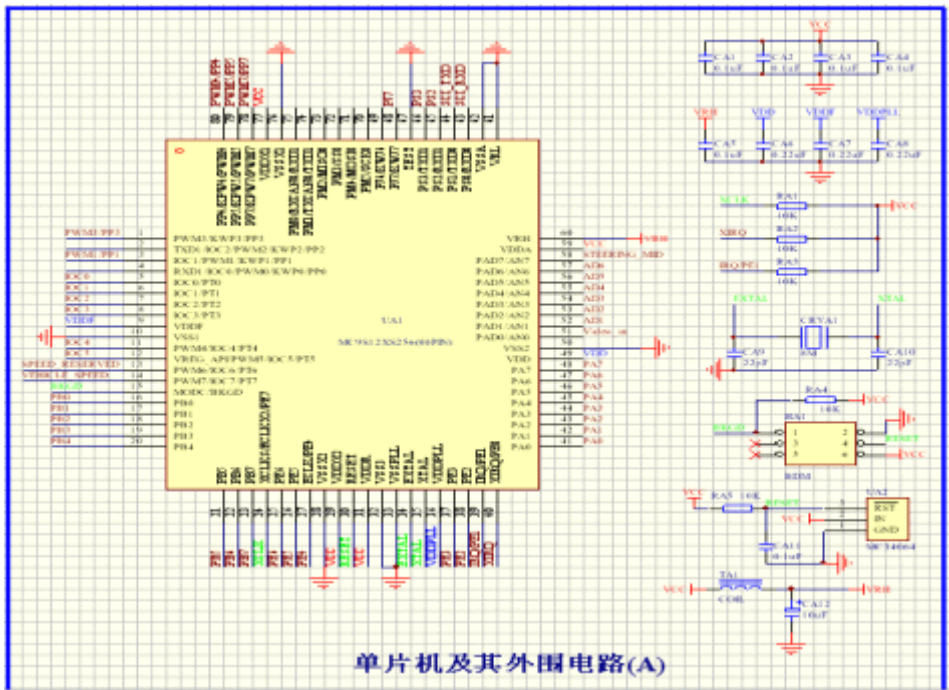


图3.6 最小系统电路图

## 第四章 视频采集和图像处理

### 4.1 视频采集

视频采集模块由摄像头、1881 视频采集信号分离芯片以及 S12 的 AD 模块构成。视频信号是 AD 采集的基础，下面先介绍视频信号的特征，然后再展开。

#### 4.1.1 摄像头工作原理

摄像头分为黑白和彩色两种，根据赛道特点可知，为达到寻线目的，只要提取到画面的灰度信息，而不必要提取其色彩信息，所以本设计中采用的是黑白摄像头。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。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。具体而言（参见图 5-1），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（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（即持续时间远长于）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，上一场的视频信号才真正到来。摄像头每秒扫描 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。

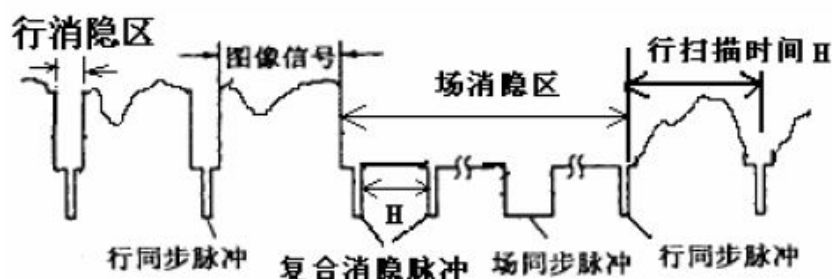


图 4.1 摄像头视频信号

### 4.1.2 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  $7\mu\text{s}$ ，所以如果选用一个分辨率为 320 的摄像头，则单行视频信号持续时间约为  $20\text{ms}/320=62.5\mu\text{s}$ ，AD 对单行视频信号采样的点数将不超过  $[62.5/7]+1=9$  个。若使用分辨率越高的，则其采到的点数将更少。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  $20\text{ms}/300=66\mu\text{s}$ ，AD 采样每行视频信号的点数至多为  $[66/7]+1=10$  个（不超频的情况下），这对赛车定位来说是不够的，所以在选用摄像头时，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率能力（但降低都仍然远远够用），却可以增加单片机采样视频信号的点数，提高横向分辨能力。在本次设计过程中，在市场上我们找到了索尼的 CCD 摄像头，其分辨率为 200 万像素，以下章节中的视频采集工作就是基于此摄像头。

### 4.1.3 信号分离电路

要对有效地对视频信号进行采样，首先要处理好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，为了减轻其处理负担，采用了第二种方法进行信号的时序信息，如行同步脉冲、场同步脉冲和奇、偶场信号信息等，并将它们转换成 TTL 电平直接输出给单片的 I/O 口作控制信号之用。1881 的端口接线方式如图 4.2 所示。

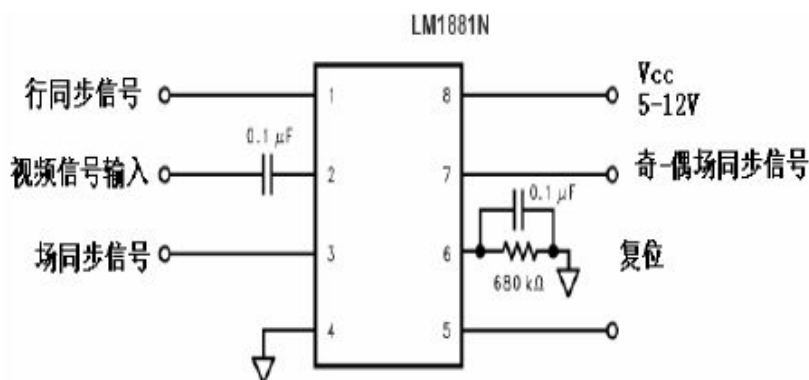


图 4.2 LM1881

其中, 引脚 2 为视频信号的输入端, 引脚 1 为行同步信号输出端(图 5-3 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 4.3 中的 c)。引脚 7 为奇-偶场同步信号端输出端, 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, 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。

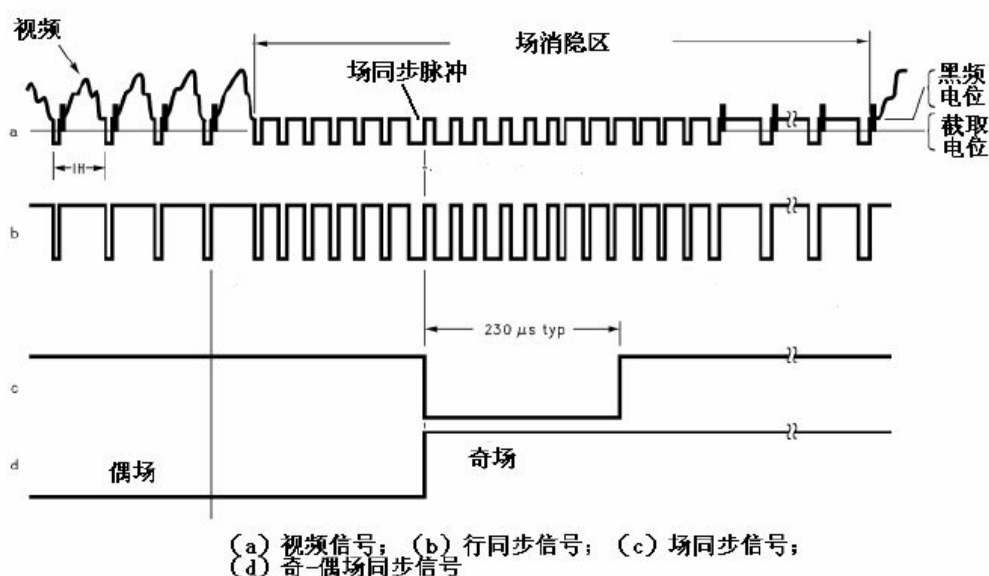


图 4.3 LM1881 信号时序图

## 4.2 图像处理

### 4.2.1 目标指引线的提取

智能车通过图像采样模块获得车前方的赛道图像信息，往下介绍如何分析此二维数组来提取黑线，我们采用边缘检测的方法。二维数组的行数和列数即为像素的图像坐标，我们若求出了黑线边缘的图像坐标，就知道了黑线的位置。黑线边缘的特点是其左、右两像素为一黑一白，两像素值的差的绝对值大于某阈值，大于可根据试验确定；而其余处的相邻两像素或全白，或全黑，像素值差的绝对值小于该阈值。这样，只要我们对两数组每行中任何相邻两点做差，就可以根据差值的大小是否大于该阈值来判断此两点处是否为黑线边缘，还可以进行根据差值的正负来判定边缘处是左白右黑，还是右黑左白。从最左端的第一个有效数据点开始依次向右进行阈值判断：由于实际中黑白赛道边缘可能会出现模糊偏差，导致阈值并不是个很简单介于两相邻之间，很可能要相隔两个点。因此：第  $line$  为原点，判断和  $line+3$  的差是否大于该阈值，如果是则将  $line+3$  记为  $i$ ，从  $i$  开始继续在接下的从  $i+3$  到该行最末一个点之间的差值是否大于阈值，如果大于则将  $line+i/2+1$  的坐标赋给中心给黑线中心位置值，如 4.4

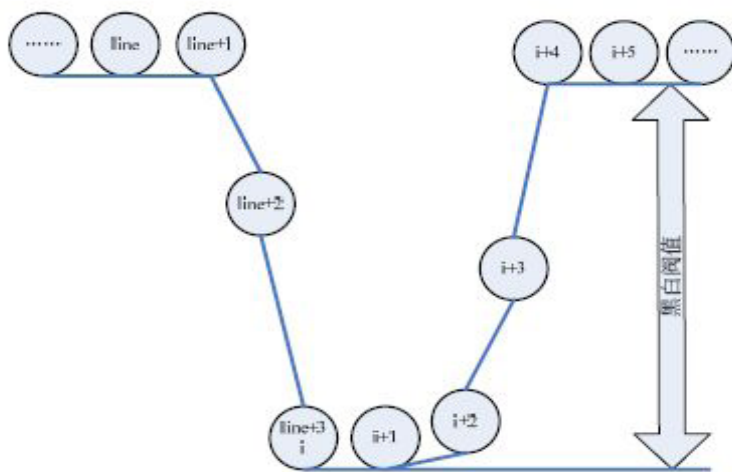


图4.4 黑线提取法

利用该算法所得到的黑线提取效果不仅可靠，而且实时性好；在失去黑线目标以后能够记住是从左侧还是从右侧超出视野，从而控制舵机转向让赛车回到正常赛道。试验表明：只要阈值取得合适，该算法不仅可靠，而且实时性较好。如果更进一步可以设置阈值根据现场情况的变化而变化。在黑线引导线已经能够可靠提取的基础上，我们可以利用它来进行相应的弯、直道判定，以及速度和转向舵机控制算法的研究。

## 第五章 智能车软件设计与实现

高效稳定的软件程序是智能车平稳快速寻线的基础。本智能车采用摄像头作为寻线传感器，采集数据的处理就成了整个软件的核心内容。在智能车的转向和速度控制方面，我们使用了增量式 PID 控制算法，配合使用理论计算和实际参数补偿，使在寻线中智能车达到了稳定快速的效果。

### 5.1 主程序流程框图

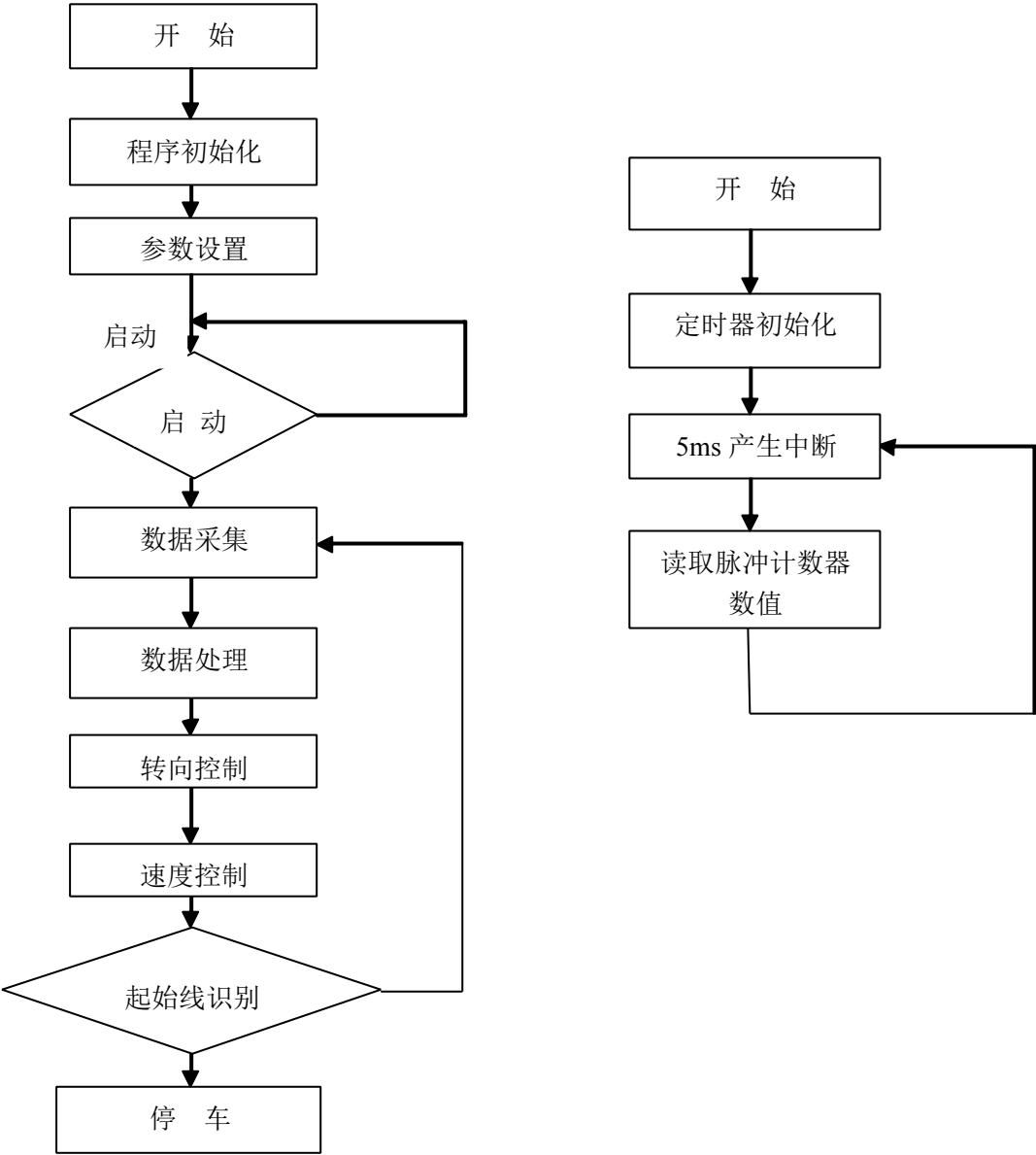


图5.1流程图

## 5.2 初始化

初始化包括：

总线时钟的设置，

I/O 端口的设置，

ADC 通道选择和工作方式配置，

PWM 通道选择和工作方式设置，

中断设置。

## 5.3 测量

由硬件设计部分可知，速度采用脉冲计数器，光电编码器的输出接在定时器接口上。在软件里，通过每隔 5ms 的中断处理程序读取一次脉冲次数，读完后立即清零，等待下一次的中断。并将读取数据经过处理，除去偶然误差，做为速度值。

## 5.4 舵机和速度的 PID 算法

### 5.4.1 经典 PID 控制算法介绍

PID 控制是工程实际中应用最为广泛的调节器控制规律。问世至今 70 多年来，它以其结构简单、稳定性好、工作可靠、调整方便而成为工业控制的主要技术之一。单位反馈的 PID 控制原理框图如图 5.2：

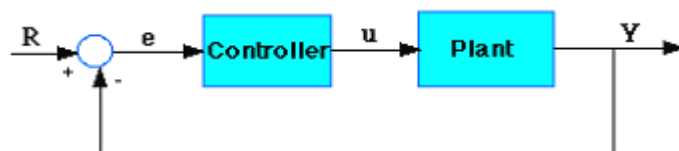


图 5.2 单位反馈的 PID 控制原理图

单位反馈  $e$  代表理想输入与实际输出的误差，这个误差信号被送到控制器，



控制器算出误差信号的积分值和微分值，并将它们与原误差信号进行线性组合，得到输出量  $u$ 。

$$u = k_p e + k_i \int e dt + k_d \frac{de}{dt} \quad \text{公式 1}$$

其中， $k_p$ 、 $k_i$ 、 $k_d$  分别称为比例系数、积分系数、微分系数。 $u$  接着被送到了执行机构，这样就获得了新的输出信号  $u$ ，这个新的输出信号被再次送到感应器以发现新的误差信号，这个过程就这样周而复始地进行。

#### (1) PID 各个参数作用基本介绍：

增大微分项系数可以加快动态系统响应，但容易引起震荡。一般增大比例系数能够减小上升时间，但不能消除稳态误差。增大积分系数能够消除稳态误差，但会使瞬时响应变差。增大微分系数能够增强系统的稳定特性，减小超调，并且改善瞬时响应。对连续系统中的积分项和微分项在计算机上的实现，是将上式转换成差分方程，由此实现数字 PID 调节器。

#### (2) 位置式 PID 控制算法

用矩形数值积分代替上式中的积分项，对导数项用后向差分逼近，得到数字 PID 控制器的基本算式（位置算式）：

$$u_n = k_p (e_n + \frac{1}{T_i} \sum_{k=1}^n e_k T + T_d \frac{e_n - e_{n-1}}{T}) \quad \text{公式 2}$$

其中  $T$  是采样时间， $k_p$ 、 $T_i$ 、 $T_d$  为三个待调参数，我们在实际代码实现算法时，处理成以下形式：

$$\text{PreU} = Kp * \text{error} + Ki * \text{Integral} + Kd * \text{derror} \quad \text{公式 3}$$

#### (3) 增量式 PID 控制算法

对位置式加以变换，可以得到 PID 算法的另一种实现形式（增量式）：

$$\Delta u_n = u_n - u_{n-1} = k_p [(e_n - e_{n-1}) + \frac{1}{T_i} e_n + \frac{T_d}{T} (e_n - 2e_{n-1} + e_{n-2})] \quad \text{公式 4}$$

我们在实际代码实现时，处理成以下形式：

$$\text{PreU} += (Kp * d\_error + Ki * \text{error} + Kd * dd\_error) \quad \text{公式 5}$$

这种算法用来控制步进电机特别方便，对直流电机也可以采用，其实如果对控制有更高的要求或者干扰因素较多，我们可以对 PID 算法做各种改进，比如用梯形法做数值积分以提高精度，将差分改成一阶数字滤波等等，在实际调车的过程中，我们确实遇到过由于自制码盘采样得到的脉冲上升下降沿不够陡，使得速度采样出现不稳定和失真，但由于这些附加处理比较耗费代码的运行时间，出于代码效率和实际效果的比较，我们没有采用这些改进的方案，另外可以考虑加反向器来整波形得到较为理想的方波。

运用 PID 控制的关键是调整三个比例系数，即参数整定。PID 整定的方法有两大类：一是理论计算整定法。它主要是依据系统的数学模型，经过理论计算确定控制器参数。由于智能车整个系统是机电高耦合的分布参数系统，并且要考虑赛道具体环境，要建立精确的智能车运动控制数学模型有一定难度，而且我们对车身机械结构经常进行不断修正，模型参数变化较频繁，可操作性不强；二是工程整定方法，它主要依赖工程经验，直接在控制系统的试验中进行，且方法简单，我们采用了这种方法，同时，我们先后实验了几种动态改变 PID 参数的控制方法。

### 5.4.2 经典 PID 算法在本智能车上的应用

#### (1) 速度 PID

本次设计中，我们采用增量式 PID。

增量型算法与位置型算法相比，具有以下优点：增量型算法不需要做累加，增量的确定仅与最近几次偏差采样值有关，计算精度对控制量的计算影响较小，而位置型算法要用到过去偏差的累加值，容易产生大的累加误差；增量型算法得出的是控制量的增量，而位置型算法的输出是控制量的全量输出，误动作影响大；采用增量型算法，易于实现手动到自动的无冲击切换。在实际应用中，采样的反馈值，即为脉冲累加器中的 PACN10 中的脉冲数，预设门限值在参数整定时根据实际情况调节，输出值并不能直接用来控制电机，需要将其转换为控制 PWM 占空比，然后用增大或减小 PWM 占空比的方法来实现对电机的加减速的控制。换句话说，在求偏差量时，实际上用的是每 10ms 电机转过的齿轮数和实际期望电机转过的齿轮数，通过二者的差值，再乘以相应的系数，即  $K_P$ 、 $K_I$ 、 $K_D$  的协调控制，计算出相应的 PWM 占空比，实际上用的是 PWMDTY

的值。

本设计中综合考虑各种因素，最后选用的采样周期为 5ms，即每 5ms 对电机进行一次 PID 调节。经过我们反复测试，选择的 PID 调节策略具体在程序中的代码如下：

$$\text{speed\_out} = k * \text{es1} / 10 + k_i * \text{sum\_error} / 300 + k_d * (\text{es1} - \text{es2}) / 10$$

其中：es1—期望速度与实测速度的偏差值

es2—上次的偏差值

sum\_error—偏差值的累加和

k, ki, kd—PID 参数值

## (2) 舵机的控制

我们对舵机的控制采用了位置式 PID 算法，根据往届的资料和自己的测试，将传感器的反馈值经过算法处理后得到的黑线位置和对应的舵机 PID 参照角度处理成一次线性关系。

开始在较低速（2m/s 以下）试验时，在偏离黑线很少的某个范围，将  $K_p$  直接置零，在偏离黑线较少的某个范围，将  $K_p$  值减小为原来的一半，在偏离较大的其他情况，则保持  $K_p$  原来的大小。取得的实际效果在弯道较多、直道较短的赛道上，车子转弯流畅，直道也能基本保持直线加速，车身左右抖动较小。但在提高车速至高速（2.5m/s 以上）时，发现车身在跑直道特别是长直道时，车身左右震荡比较严重，究其原因，软件上自身编写的 PID 舵机控制还不够精细，动态适应能力不够，在从弯道进直道的过程中，由于小车循黑线本质上是一个随动系统，积分项在弯道累积的偏差错误地加在直道的跟踪上，造成在进入直道时转向不够准确，跑直道时虽然能跟踪黑线，但是转向调整往往超调，导致车身在直道上左右震荡，这种震荡严重影响了车的整体表现，导致小车开始转向，由于中间一直检测到弯道，小车会沿 S 弯道左右震荡，同时相应会减速，经过反复调试 PID 参数，我们发现光调整 PID 参数很难使车在跑 S 弯和长直道时都选择跑最佳路径，并且不影响车拐普通弯。这就要求我们要智能地识别出当前赛道是哪种类型，我们没有选择赛道记忆等方法来判断当前赛道是哪种类型，仅仅根据黑线位置动态改变 PID 参数，得到了较好的控制效果，

经过我们反复测试，选择的 PID 调节策略具体在程序中的代码如下：

$\text{angle\_out} = e1 * ks + ksd * (e1 - e2)$

其中：e1—当前位置偏差值，e2—上一次的偏差值，ks，ksd—PID 参数

## 第六章 智能车技术参数说明

项目	参数
路径检测方法（赛题组）	摄像头组
车模几何尺寸（长、宽、高）（毫米）	290* 160*310
车模轴距/轮距(前/后)（毫米）	200/ (150/170)
车模平均电流（匀速行驶）（毫安）	3000
电路电容总量（微法）	1100
传感器种类及个数	摄像头传感器、测速传感器共 2 个
新增加伺服电机个数	0
赛道信息检测空间精度（毫米）	15
赛道信息检测频率（次/秒）	200
主要集成电路种类/数量	MC9S12XS128*1
车模重量（带有电池）（千克）	1.20

## 参考文献

- [1] 邵贝贝. 嵌入式实时操作系统[LC / OS-II(第2版)][M]. 北京. 清华大学出版社. 2004
- [2] 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
- [3] 王晓明. 电动机的单片机控制[M]. 北京. 北京航空航天大学出版社. 2002
- [4] 臧杰, 阎岩. 汽车构造[M]. 北京. 机械工业出版社. 2005
- [5] 安鹏, 马伟. S12 单片机模块应用及程序调试[J]. 电子产品世界. 2006. 第 211 期. 162-163
- [6] 童诗白, 华成英. 模拟电子技术基础[M]. 北京. 高等教育出版社. 2000
- [9] 沈长生. 常用电子元器件使用一读通[M]. 北京. 人民邮电出版社. 2004
- [10] 宗光华. 机器人的创意设计与实践[M]. 北京. 北京航空航天大学出版社. 2004
- [11] 张伟等. Protel DXP 高级应用[M]. 北京. 人民邮电出版社. 2002
- [12] 张文春. 汽车理论[M]. 北京. 机械工业出版社. 2005

## 附录：部分程序代码

```

Filter:
    LDX      #CCDInterpolation_Aryy
    LDAA     #50
Filter_Initial1:
    MOVB     #$FF, 1, X+
    DBNE     A, Filter_Initial1
    LDX      #Miss_White_Piece_Array
    LDAA     #50
Filter_Initial2:
    MOVB     #$FF, 1, X+
    DBNE     A, Filter_Initial2
;-----
    CLR      JiZhun_No_Effect_Flag

    MOVB     #25, Filter_Compute_Max_Line
    CLR      Basement_Num
    CLR      Over_Not_Nomal
    MOVW     #Basement_Array, Basement_Array_Address
    CLR      Current_Found_Line1_Point_Count
    CLR      Current_Found_Line2_Point_Count

    MOVB     #$00, DeltaNearLine1
    MOVB     #$00, DeltaNearLine2
    MOVB     #$00, DeltaFarLine1
    MOVB     #$00, DeltaFarLine2

    MOVB     #$FF, DeltaLine1_Far_Flag
    MOVB     #$FF, DeltaLine1_Near_Flag
    MOVB     #$FF, DeltaLine2_Far_Flag
    MOVB     #$FF, DeltaLine2_Near_Flag

;
#$A1      ;#####$#####
;      JSR      SCI_TX
LDAB

```

```

        JSR      Distortion_revert
        JSR      CuLv_Filter

;
#$A2      ;#####$#####
;        JSR      SCI_TX
LDAB

```

```

        MOVW      #Filter_BelowArray,Filter_EntryArray_Address
;   MOVW      #DEBUG_BelowArray,Filter_EntryArray_Address
        MOVW
Filter_EntryArray_Address,BelowArray_Count_Address

```

```

;-----
;   LDAB      Ji_Zhun_No_Point
;   CMPB      #2
;   BLO       Filter_Initial4
;   JMP       Recognise_JiZhun_Contiunue_B12
;-----

```

```

        JSR      Compute_Effect_Whith

```

```

;-----

```

```

        JSR      Filter_B101

```

```

LDAB

```



---

```
#$A3          ;#####$#####  
;      JSR      SCI_TX
```

```
;===== TEST
```

```
; LDAB      #$B1  
; JSR      SCI_TX
```

```
; LDAB      Effect_Whithmin  
; JSR      SCI_TX
```

```
; LDAB      Effect_Whithmax  
; JSR      SCI_TX
```

```
; LDAB      Effect_Start_Point  
; JSR      SCI_TX
```

```
; LDAB      Effect_Final_Point  
; JSR      SCI_TX
```

```
; LDX      #Effect_Whith_Array  
; LDAA     #5
```

```
;Caishu_Filter2:
```

```
; LDAB      1,X+  
; JSR      SCI_TX  
; DBNE     A,Caishu_Filter2
```

```
;=====
```

```
;-----
```

```
Filter_Initial4:
```

```
;      BRCLR      Start_Zhui_Line_Flag,$80,Filter_Initial3  
;      JSR      Find_All_Line  
;      LBRA      Filter_End
```

```
Filter_Initial3:
```

```

        CLR          Filter_Compute_Line

Filter_A:

        LDAB        Filter_Compute_Line
        LDX         #Line1_Boundary_Tab
        MOVB        B, X, Line1_Boundary_Point
        LDX         #Line2_Boundary_Tab
        MOVB        B, X, Line2_Boundary_Point


        LDX         BelowArray_Count_Address
        LDAA        1, X+
        STAA        Filter_A_Line_Count
        TSTA
        LBEQ        Filter_A6
        INX
        INX
        STX         BelowArray_Whith_Address
;----- Width
Filter_A1:
        LDX         BelowArray_Whith_Address
        LDAA        , X
        CMPA        Effect_Whithmin
        BGE         Filter_A2
        LBRA        Filter_A2B3
Filter_A2:
        CMPA        Effect_Whithmax
        BLE         Filter_A3
        LBRA        Filter_A2B5
;-----
Filter_A2B3:
        LDAB        Effect_Whith_Array+4
        CMPB        #15

```

---

```

        LBL5      Filter_A5
        LDAB      Effect_Whithmin
        SUBB      ,X
        CMPB      #5
        BLS       Filter_A3
        LBRA      Filter_A5
;-----

Filter_A2B5:
        LDAB      Effect_Whith_Array+4
        CMPB      #$FF
        LBEQ      Filter_A5
        CMPB      #20
        BHS       Filter_A3
        LDAB      ,X
        SUBB      Effect_Whithmax
        CMPB      #5
        BLS       Filter_A3
        LBRA      Filter_A5

;----- Repeat Restrict
Filter_A3:

        LDAA      Effect_Final_Point
        CMPA      #$FF
        LBEQ      Filter_A7
        CMPA      #Line2_Boundary_Point_Const
        BNE       Filter_A32
        LDAA      Line2_Boundary_Point
Filter_A32:
        STAA      RegisterA_Position
        DEX
        LDAA      0,X
        CMPA      #Line2_Boundary_Point_Const
        BNE       Filter_A31
        LDAA      Line2_Boundary_Point
Filter_A31:
        STAA      RegisterAH_Position
        CMPA      RegisterA_Position
        BLS       Filter_A33

```

```

        LDAA      RegisterA_Position
        STAA      RegisterAL_Position

;-----
        LDAB      Effect_Start_Point
        CMPB      #Line1_Boundary_Point_Const
        BNE       Filter_A42
        LDAB      Line1_Boundary_Point
Filter_A42:
        STAB      RegisterB_Position
        DEX
        LDAB      0, X
        CMPB      #Line1_Boundary_Point_Const
        BNE       Filter_A41
        LDAB      Line1_Boundary_Point
Filter_A41:
        STAB      RegisterBH_Position
        CMPB      RegisterB_Position
        BHS       Filter_A43
        LDAB      RegisterB_Position
Filter_A43:
        STAB      RegisterBL_Position

;-----
        LDAA      RegisterBL_Position
        CMPA      RegisterAL_Position
        BHS       Filter_A5
        LDAB      RegisterBH_Position
        CMPB      RegisterB_Position
        BLO       Filter_A45
        LDAA      RegisterAH_Position
        CMPA      RegisterA_Position
        BHI       Filter_A45
;=====
;   LDAB      #$B2
;;   JSR      SCI_TX
;   LDAB      #$FF
;   JSR      SCI_TX
;-----
        LBRA      Filter_A7

```

---

```

Filter_A45:
    LDAA    RegisterA_Position
    SUBA    RegisterB_Position
    SEX     A, X

    LDAA    RegisterAL_Position
    SUBA    RegisterBL_Position
    LDAB    #10
    MUL

    IDIV
    XGDX
    CMPB    #6
    BLO     Filter_A5
;=====

```

```

TEST
;   PSHB
;   LDAB    #$B2
;   JSR     SCI_TX
;   PULB
;   JSR     SCI_TX
;-----

    LBRA    Filter_A7

```

```

;=====
;-----

```

```

Filter_A5:
    DEC     Filter_A_Line_Count
    LDAA    Filter_A_Line_Count
    TSTA
    BEQ     Filter_A6
    LDX     BelowArray_Whith_Address
    LDAB    #3

```

```

        ABX
        STX      BelowArray_Whith_Address
        LBRA     Filter_A1

;-----
Filter_A6:
        LDAB     Basement_Num
        TSTB
        LBNE     Filter_A8

;-----
;--扩大宽度
;-----

;-----
;前 1 行宽度
;-----

        LDAB     Filter_Compute_Line
        LSLB
        LDX      #Last_CCDInterpolation_Aryy
        ABX
        LDAA     1, X
        CMPA     0, X
        BEQ      Filter_A66
        CMPA     #$FF
        BNE      Filter_A61
        LDAA     Filter_Compute_Line
        LDY      #Line2_Boundary_Tab
        LDAA     A, Y
Filter_A61:
        LDAB     0, X
        CMPB     #$FF
        BNE      Filter_A62
        LDAB     Filter_Compute_Line
        LDY      #Line1_Boundary_Tab
        LDAB     B, Y
Filter_A62:
        SBA
        TAB

;-----

```

---

LDAA	3, X
CMPA	2, X
BEQ	Filter_A66
CMPA	#\$FF
BNE	Filter_A64
LDAA	Filter_Compute_Line
INCA	
LDY	#Line2_Boundary_Tab
LDAA	A, Y

Filter\_A64:

PSHB	
LDAB	2, X
CMPB	#\$FF
BEQ	Filter_A65
LDAB	Filter_Compute_Line
INCB	
LDY	#Line1_Boundary_Tab
LDAB	B, Y

Filter\_A65:  
SBA

;-----

PULB	
SBA	
TSTA	
BGE	Filter_A651

LDAB	Effect_Whithmin
SBA	
TSTA	
BGE	Filter_A66
NEGA	
BRA	Filter_A66

Filter\_A651:

LDAB	Effect_Whithmax
ABA	
STAA	Effect_Whithmax
BRA	Filter_A66

;-----

```
Filter_A66:
    INC      Filter_Compute_Line
    LDAB     Filter_Compute_Line
    CMPB     #Ji_Zhun_Line-Allow      ;
有效行不能大于 4
    LBHI     Filter_A99
```

;-----

```
Filter_A63:
    LDX      BelowArray_Count_Address
    LDAB     1, X+
    LDAA     #3
    MUL
    ABX
    STX      BelowArray_Count_Address

    LBRA     Filter_A
```

;-----

Filter\_A7:

;=====

```
    ; LDAB     RegisterA_Position
    ; JSR      SCI_TX
    ; LDAB     RegisterB_Position
```



---

```

;   JSR      SCI_TX

;   LDAB     RegisterAL_Position
;   JSR      SCI_TX
;   LDAB     RegisterBL_Position
;   JSR      SCI_TX

;   LDAB     RegisterA_Position
;   SUBB     RegisterB_Position
;   JSR      SCI_TX

;   LDAB     RegisterAL_Position
;   SUBB     RegisterBL_Position

;   JSR      SCI_TX
;=====

```

```

INC      Basement_Num
LDX      Basement_Array_Address
MOVW     BelowArray_Whith_Address, 2, X+
STX      Basement_Array_Address
LBRA     Filter_A5

```

```

Filter_A8:
LDAB     Basement_Num
CMPB     #1
BNE      Filter_A9
MOVW     Basement_Array, BelowArray_Whith_Address
BRA      Filter_A0B3

```

```

;=====

```

```

Filter_A9:
BSET     Over_Not_Nomal, #$1
LDX      #Basement_Array
MOVB     #$FF, RegisterA_Position
LDAA     Basement_Num
Filter_A91:
LDY      2, X+
LDAB     0, Y
SUBB     Effect_Whith_Array+4

```

```

        TSTB
        BGE      Filter_A92
        NEGB
Filter_A92:
        CMPB     RegisterA_Position
        BHS      Filter_A93
        STY      BelowArray_Whith_Address
        STAB     RegisterA_Position
Filter_A93:
        DBNE     A, Filter_A91
        BRA      Filter_A0B3

;-----

;=====
Filter_A0B3:
        MOVW
BelowArray_Whith_Address, Right_BelowArray_Whith_Address
        JSR      Recognise_JiZhun_Contiunue
;-----
        BRCLR   Over_Not_Nomal, #$40, Filter_A0B6
        CLR     Basement_Num

```