第十一届“恩智浦”杯全国大学生

智能汽车竞赛

**技术报告**



学 校：太原理工大学

队伍名称：晋豹六队

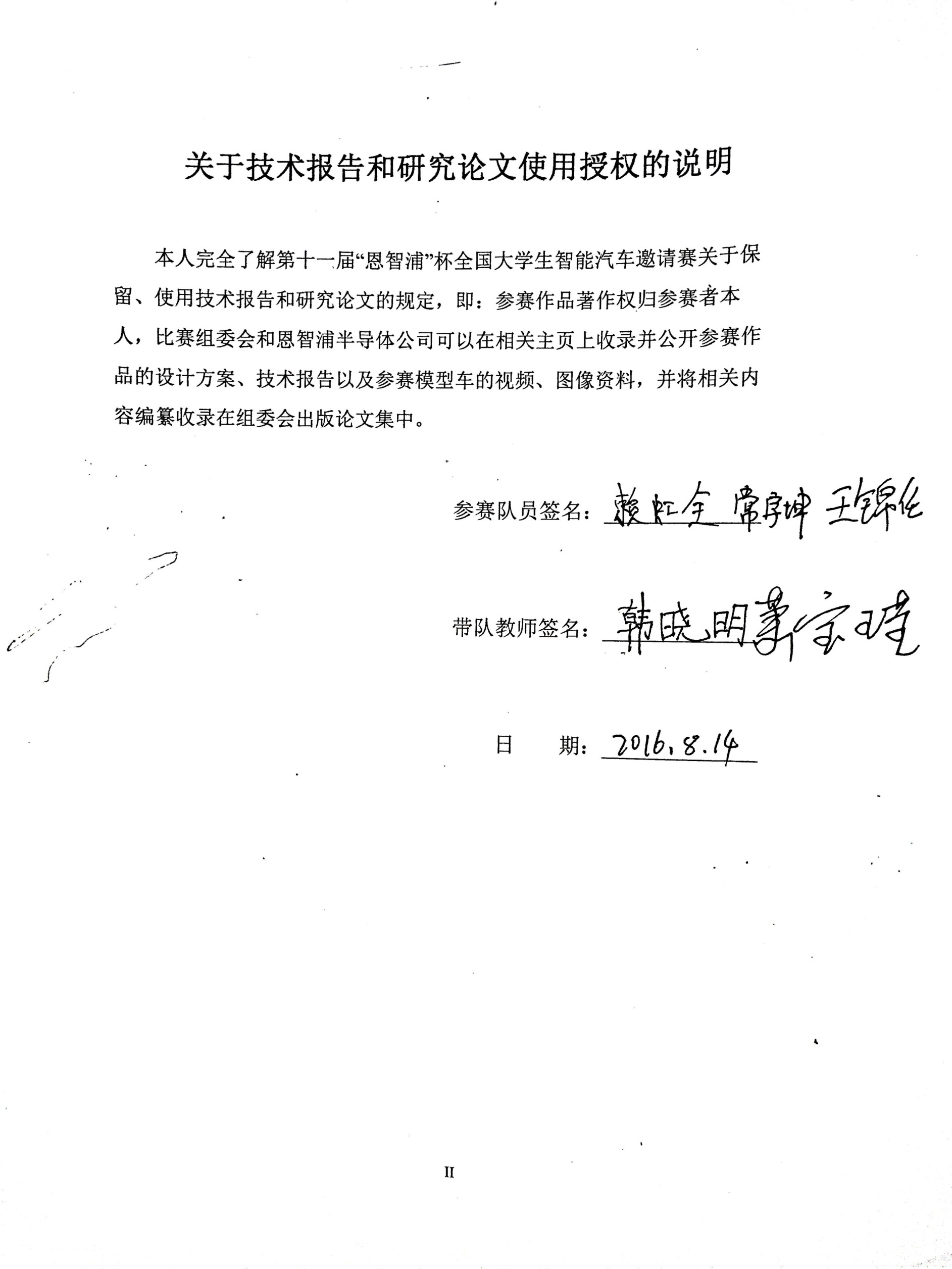
参赛队员：赖虹全

王锦任

常宇坤

带队教师：萧宝瑾

韩晓明



摘要

本设计以第十一届“恩智浦”杯全国智能车大赛为背景，以大赛组委会指  
定的B型车模为平台，制作一辆以恩智浦32位微控制器MK60DN256ZVLL10 作为核心控制单元。以 IAR 为开发环境，利用模拟摄像头实现赛道两边黑道的识别，解决十字、障碍、坡道、灯塔起始线等特殊赛题，并沿着赛道以尽可能快的速度运行。该智能车车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，可以为无人驾驶汽车及环保个人交通工具的后续研究提供经验。  
【关键词】智能车; MK60DN256ZVLL10; 摄像头;

目 录

第一章 引 言......................................................................................................1

1.1“恩智浦”杯智能车大赛介绍......................................................................1

1.2第十一届智能车大赛规则介绍.....................................................................1

1.3章节安排..........................................................................................................2

第二章机械结构部份的设计及调整............................................................3

2.1机械结构制作部分设计............................................................................. ….3

2.1.1 舵机改装..................................................................................................3  
2.1.2 底盘改装................................................................................................3

2.1.3 前轮定位..................................................................................................4

2.1.4 车轮外倾角............................................................................................5

2.1.5 前轮前束................................................................................................5

2.2摄像头的选型和架设.......................................................................................6

2.3摄像头信息参数...............................................................................................7

2.4光电编码器的选择和安装..............................................................................7

2.5智能车主要技术参数......................................................................................8  
第三章 模块规划................................................................................................. 9  
第四章 硬件电路的设计...............................................................................11

4.1电源模块设计.................................................................................................11

4.2传感器电路设计与实现.................................................................................12

4.3驱动电路设计.................................................................................................13

4.4其他硬件电路.................................................................................................13

第五章 总体理论分析及控制器软件算法实现................................. 15  
5.1 系统控制总体设计.....................................................................................15

5.2 路径识别部分...............................................................................................15

5.3 速度控制部分...............................................................................................17

5.4 舵机的PID控制算法...................................................................................17

5.5 辅助车模调试程序设计...............................................................................21

第六章 系统的开发环境与车模试............................................................23

6.1 IAR 的使用.....................................................................................................23

6.2基于 LABVIEW 的上位机................................................................................23

第七章 总 结.......................................................................................................25

7.1 总结................................................................................................................25

7.1.1 控制方面...................................................................................................25

7.1.2 特色设计...................................................................................................25

7.2 结束语............................................................................................................25

致谢..........................................................................................................................27

参考文献.................................................................................................................28

附录 A MK60DN256ZVLL10 最小系统原理图…………………………………29

附录 B 程序源代码..................................................................................................30

# 第一章 引 言

1.1“恩智浦”杯智能车大赛介绍

本文以第十一届“恩智浦”杯全国大学生智能汽车竞赛为背景，该比赛受教育  
部高等教育司委托，由教育部高等自动化专业教学指导分委员会（以下简称自  
动化分教指委）主办全国大学生智能汽车竞赛。该竞赛以智能汽车为研究对象  
的创意性科技竞赛，是面向全国大学生的一种具有探索性工程实践活动，是教  
育部倡导的大学生科技竞赛之一，为加强大学生实践、创新能力和团队精神的  
培养，促进高等教育教学改革。该竞赛以“立足培养，重在参与，鼓励探索，追  
求卓越”为指导思想，旨在促进高等学校素质教育，培养大学生的综合知识运用  
能力、基本工程实践能力和创新意识，激发大学生从事科学研究与探索的兴趣  
和潜能，倡导理论联系实际、求真务实的学风和团队协作的人文精神，为优秀  
人才的脱颖而出创造条件。该竞赛由竞赛秘书处为各参赛队提供/购置规定范围  
内的标准硬软件技术平台，竞赛过程包括理论设计、实际制作、整车调试、现  
场比赛等环节，要求学生组成团队，协同工作，初步体会一个工程性的研究开  
发项目从设计到实现的全过程。该竞赛融科学性、趣味性和观赏性为一体，是  
以迅猛发展、前景广阔的汽车电子为背景，涵盖自动控制、模式识别、传感技  
术、电子、电气、计算机、机械与汽车等多学科专业的创意性比赛。该竞赛规  
则透明，评价标准客观，坚持公开、公平、公正的原则，保证竞赛向健康、普  
及，持续的方向发展。  
 该竞赛以恩智浦半导体公司为协办方，得到了教育部相关领导、恩智浦  
公司领导与各高校师生的高度评价，已发展成全国 30 个省市自治区近 300  
所高校广泛参与的全国大学生智能汽车竞赛。2008年起被教育部批准列入国家教学质量与教学改革工程资助项目中科技人文竞赛之一（教高函[2007]30号文）。

1.2 第十一届智能车大赛规则介绍

本届大赛要求摄像头组使用B型车模，可以向两个方向运行。须采用恩智浦半导体公司的 8 位、16 位、32 位处理器作为唯一的微控制器。参加摄像头组的车模可以使用摄像头传感器进行道路检测，禁止使用激光传感器。  
 摄像头组的比赛采用起跑线控制的方式。参赛队伍的车模需要能够识别  
赛道上的起跑线标志完成车模的出发和停止控制。

1.3章节安排

技术报告的章节安排如下：第 1 章引言，大赛介绍和规则介绍。第 2 章智能车机械调整，主要介绍对小车的机械结构的改装及一些模块的安装方法。第 3章各模块的规划。第 4 章介绍智能车硬件系统设计，主要介绍智能车硬系统的设计思想、设计过程和设计结果。第 5 章智能车软件系统设计，主要介绍智能车软件系统的理论分析、控制器设计过程和调试。第 6 章系统的开发环与  
辅助设计软件介绍，主要介绍了 IAR 作为系统的开发环境的使用，还介绍了其  
它在系统设计过程用到过的计算机辅助设计软件。第 7 章结论与展望，总结了  
在本大赛当中所取得的成绩与不足之处。第 6 章系统的开发环境与辅助设计软  
件介绍，主要介绍了 IAR作为系统的开发环境的使用，还介绍了其它在系统设  
计过程用到过的计算机辅助设计软件。

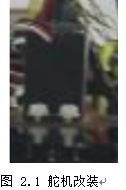
# 第二章机械结构部份的设计及调整

智能车机械部分设计主要包括制作和调整两部分内容，制作部分的内容主要是对车模没有的部分进行设计，包括传感器支架、电路板固定、防撞、测速码盘安装等。调整部分则主要是针对智能车车模本身已经有的机械部分，在规则允许范围内进行调整，改装，提高其运动性能，以适应高速行驶和快速控制，这部分主要包括舵机改装、底盘调整、避震调整、四轮定位等。本章内容主要对电路板安装、传感器支架以及机械调校部分进行介绍。

2.1 机械结构制作部分设计

由于大赛提供车模本身是运动型模型车通用车模，并没有提供专门为智能车安装电路、传感器等电路部分的部件，因此这部分机械结构需要自行设计制作并安装。制作部分主要原则为：轻、牢、简，所以我们主要选择铝合金、尼龙等原材料制作，所有自制结构，都是由我们手工制作。 车模本身的机械结构是通用结构，并不适合智能车竞赛的要求，因此要对这些部分进行改装，另外，为了提高车模的运动性能，对一些机械结构还需要调整，比如车轮前束等。这部分着重介绍舵机改装、底盘等部分的调整和改装。

2.1.1 舵机改装

为了提高舵机反应速度，在相同转角下，有尽可能大的线行程，因此需要延长舵机臂。另一方面，由于舵机扭矩和转角精度的限制，不能无限制延长舵机臂，这样就确定了舵机臂的长度，并使用铝合金片加工成形，尺寸为 24mm× 38mm。图 2.1 是舵机安装实物图，四个螺丝将舵机牢固安装在支架上。

2.1.2 底盘改装

由于赛道特性，底盘改装目标是尽可能低(能保证通过坡道)，这样可以最大  
程度保证智能车行驶的稳定性。因此我们降低了底盘高度。另外，由于赛道整  
体属于平坦路面，没有较大较多的颠簸，所以我们把后避震拆除，并紧固后桥  
连接件。  
 2.1.3 前轮定位  
 B型车模前轮可以调整的角度有主销前倾、内倾、前束等，这些角度的调  
整根据每个车的机械性能不同而不同调整，我们的智能车由于重心位置在中心  
偏后，因此前轮压力较小，转向负担不大，因此为了增加抓地力和稳定性，选  
择了主销内倾和负前束的调整。另外，由于车模本身的精度限制，这部分角度  
的调整并不是主要的，仅仅是为了避免负面影响以及修正车模本身的不对称和  
不平衡问题。图 2.2 所示 A 圈是用于调整主销内倾角的丝杠，B 圈是用于调整前轮前束角的丝杠。

图 2.2 前轮定位

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°。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°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。

2.1.4 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，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。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°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°即可，并且要与前轮前束  
匹配。

2.1.5 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。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 012mm。

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 02mm 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。

2.2 摄像头的选型和架设

摄像头分为两种，数字的和模拟的，我们选择了模拟摄像头 OV5116，通  
过硬件二值化，将图像信息二值化以后传输给单片机，传输回来的还有行场中  
断信号。摄像头分为 CMOS 和 CCD 两类，也分数字和模拟的摄像头。

摄像头和模拟摄像头性能差别不大，数字摄像头电路更为简单，只需要直接将信号接入单片机的 IO 口，而模拟摄像头需要使用芯片（LM1881 等）进行视频信号同步分离。对于摄像头的选择，主要考虑以下几个参数：  
1 芯片大小  
2 自动增益  
3 分辨率  
4 最小照度  
5 信噪比  
6 标准功率  
7 扫描方式 我们最终选定了使用使用ov5116 芯片自制摄像头，相比其他摄像头，这款摄像头图像和动态性能不差，由于自己设计电路，可以把  
摄像头尺寸和重量尽可能的做小做轻。 摄像头选择碳纤维杆，通过热熔胶  
制作固定。安装好的摄像头最远扫描行 1.8 米。摄像头的假设高度大致为  
30cm，这样在兼顾远端的同时不丢失近端，同时使近端扫描行宽度足够。  
便于处理今年新增加的特殊赛题。

2.3 摄像头信息参数

1)摄像头扫描行的选择 最远端扫描行距离：1.8 米左右;最近段扫描行距  
离：车前 10cm 左右 ;主控制行位置：车前 40cm 左右 ;这样选择主要考虑到：  
首先要能看较远，这样可以对赛道和特殊赛题做出预判；不能太远，因为太远  
之后，远端信息可靠性极差，正确赛道提取并不容易，同时还会受到临道、坡  
道等特殊状况的干扰；近处行需要可靠有效，作为远处行信息的基础；控制行  
40cm足够。

2) 分辨率  
OV5116 的分辨率理论可以达到 320x240，但如果所有点都提取出来，  
将占用巨大的数据空间，处理速度也会减慢。而我们在车辆控制上所需要的  
分辨率并没有这么高，最终我们有针对性的挑选出部分扫描行和部分像素点。  
最终分辨率大概近似为 140x100。这样既保证了数据量不要过大又能保证扫  
描行间距足够采集到中心引导线。2.4 光电编码器的选择和安装。

2.4 光电编码器的选择和安装

测速采用光电编码器来测量车轮的转速。

图 2.8 光电编码器及码盘

2.5 智能车主要技术参数

智能车主要技术参数包括物理尺寸、电路指标等，具体参数见表 2.1。

表2.1 智能车主要参数表

|  |  |  |
| --- | --- | --- |
| **项目** | **单位** | **参数** |
| 车模尺寸(长×宽×高) | 毫米(mm) | 270×180×230, |
| 车模重量(带电池) | 克(g) | 1980 |
| 车模轴距 | 毫米(mm) | 190 |
| 车模平均电流(匀速行驶) | 毫安(mA) | ≈8600 |
| 直立传感器 |  | 陀螺仪、加速度计 |
| 传感器(模拟摄像头OV5116) | 个 | 1 |
| 赛道信息检测空间精度 | 毫米(mm) | 12 |
| 赛道信息检测频率 | 次/秒 | 30 |
| 核心处理器种类及个数 |  | MK60DX256ZVLL10 1 个 |
| 智能车名称 |  |  |

注：由于设计报告书写期间智能车仍然在进行改进，因此有些数据未能更新。之后的内容中涉及到的参数、程序、图片也有类似情况，不做逐一说明。

# 第三章模块规划

## 电路是整个系统中相当重要的一部分，是系统运行的基础，所以每一个电容每一个电阻都需要去考究。硬件电路的设计与制作是一个漫长的过程，也是一个充满乐趣的过程，从电路原理的研究到元器件的选择与使用再到电路板的设计与焊接，如果电路效果不佳，或是有改进，则会重复这一过程，所以为了保证电路的可靠性，同时也要让电路板集成度更高，我们花了很多精力去追求完美。电路系统框图如图 3.1：

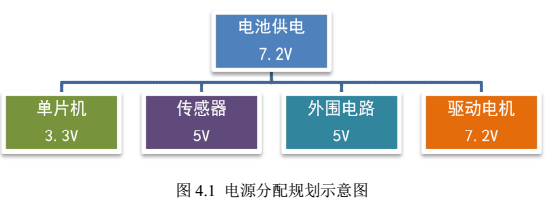
电路板总共有两块，一块是单片机最小系统板，它是系统的控制核心，引  
出了我们所需要的接口。另一块是集成了传感器、驱动电路、电源模块和一些  
接口。  
系统的硬件电路板部分大体可概括为四个部分：  
(1) 单片机最小系统板：我们采用的恩智浦公司的 MK60DN256ZVLL10  
单片机,它是系统的控制核心，具有丰富的资源，足以满足小车的控制。由于是  
新出的单片机，基于 Cortex-M4 内核，对于我们来说，掌握起来难度不小；电源管理电路模块：此模块为整个电路系统提供电源，要求电源的质量高，电压稳定。我们采用的是 2940 和 ASM1117 分别为系统提供 5V 和 3.3 V 的稳定  
电压，电机则直接由电池供电。  
(2) 电机驱动电路模块：本模块为单片机与电机之间提供一个接口，让单片  
机可以有效地控制电机。要求驱动电路的驱动电流足够大，内阻小，开关频率  
高等。我们采用的是英飞凌公司的 BTS7970 半桥驱动芯片，它的驱动能力强，  
足够满足 C 型车模 260 电机的需求。

(3) 为循迹设计的摄像头电路：我们采用的是自己围绕 OV5116 搭建的简  
易摄像头电路，OV5116 的具体优势已在前面阐述，在此就不做赘述。  
(4) 其它附加模块：这包括无线数传模块、SD 卡、液晶显示和键盘模块，  
在调试过程中起辅助作用。

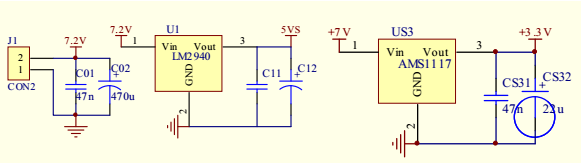
# 第四章硬件电路的设计

智能车电路部分主要的模块包括：单片机最小系统、电源模块、传感器模  
块、驱动模块以及其他周边调试模块。各模块的总体设计原则是：紧凑、易于  
拆换、稳定可靠。但根据各模块的不同，又有不同的设计要求，单片机最小系  
统我们今年购买了龙丘的最小系统板，这里就不做更多介绍，本章对其他各个  
模块的设计进行详细描述。

4.1 电源模块设计

电源的首要指标是可靠性，整个硬件系统的工作完全由电源供电的可靠性  
决定，电源供电不稳定会引起电池损耗、单片机复位、舵机及传感器损毁等严  
重问题，因此电源的设计是最重要的硬件电路设计部分之一。  
电源设计中主要考虑到需要的电压和电流，另外还设计了电池电压显示，  
便于直观发现电池电量是否正常。如图 4.1 所示，智能车需要的电源要求包括  
3.3V，5V，6V，7.2V 等。

根据规划，5V 供电我们选择了 LM2940-5，比起 7805，2940 的优点是低压差稳压，它的稳压压差可以小于 500mV，这样保证电池在低电压的情况下，仍能使单片机和传感器正常工作，同时，LM2940 的输出电流可以达到 1A[11]足够供应放大电路和键盘显示电路的工作。设计原理图如图 4.2 所示。3.3V 供电我们使用了 AMS1117-3V3 芯片作为稳压芯片。其电路原理是该芯片的典型电路，在这里不做赘述。其原理图如 4.3 。

图 4.2 LM2940-5 稳压电路原理图 图 4.3 AMS1117 稳压电路原理图

其中，为了不互相干扰工作，单片机 3.3V 和传感器的 5V 分开供电，单点共地。为了消除纹波，需要在电源输出位置加入电容滤波，提高供电稳定性。。  
驱动电机供电选择电池直接供电。最终电源模块电路板尺寸 45mm×40mm。  
4.2 传感器电路设计与实现

本设计中，传感器分为两部分：巡线传感器（摄像头）和速度传感器（光  
电码盘）。

4.2.1 巡线感器电路设计

用 OV5116 摄像头作为主要的巡线传感器，这部分也是硬件电路设计的主  
要模块，是将来控制程序的基础。在设计时，要充分考虑到赛道情况，包括赛题中的各种特殊赛道类型，要尽可能通过硬件保证检测的有效性和可靠性。另外考虑到摄像头架设偏高，因此要尽可能的制作的轻巧，以免导致车体重心偏高。特别的，本届比赛赛题加入直角弯道，这样就需要摄像头的远端图像足够清晰可靠，从而提前识别直角弯道，因此我们还要考虑摄像头的畸变问题和镜头的可视角度。摄像头最终成品如图 4.4  
4.3 驱动电路设计

驱动电路为智能车驱动电机提供控制和驱动，这部分电路的设计要求以能

够通过大电流为主要指标。驱动电路的基本原理是 H 桥驱动原理，目前流行的  
H 桥驱动电路有： H 桥集成电路，如 MC33886；集成半桥电路，如 BTS7960 等；MOS 管搭建的 H 桥电路。

我们对三种电路都进行了搭建并测试，MC33886 的优点是电路简单[12]，外围元件少，但缺点是内阻较大，通过电流有限，可以通过两片 MC33886 并联方式进行改善。MOS 管搭建的 H 桥电路可以通过较大电流，但由于每个 MOS 管体积较大，因此电路板面积较大，另外，MOS 管 H 桥电路可能会发生脉冲上升沿下降沿由于不够陡峭而带来的瞬间短路、功率过高等问题，为了避免这些问题，无疑要增加电路复杂程度。  
最终我们选择了目前比较流行的两 BTS7970 半桥集成电路组成 H 桥。  
BTS7960 的内阻为上半臂 7mΩ，下半臂 9 mΩ，完全导通后全部内阻为 16 mΩ，可通过电流为 43A[13]，远远大于智能车电机需求。

# 第五章 总体理论分析及控制软件算法实现

# 5.1 系统控制总体设计

整个系统有两部分，速度和方向。这些都是通过控制车模两个后轮驱动电机完成的。可以假设车模的电机可以虚拟地被拆解成三个不同功能的驱动电机，它们同轴相连，分别控制车模的直立平衡、前进行走、左右转向，其实三个环节并不是完全的独立的控制环路，而是相互协同配合的。为了使电路板紧凑，较少车重，系统板仅对所用到的必要引脚引出，适当留有备用端口，其中包括PWM 接口、计数器接口、普通I/O 口、BDM 接口、

SCI 接口等。主要端口作用规划见表5.1。

表5.1 单片机主要输入输出端口规划表

|  |  |  |
| --- | --- | --- |
| **端口** |  | **作用** |
| B1 | 摄像头信息读取 |  |
| A13, B2 | 行场中断 |  |
| A19,17,6,5 | 电机驱动控制 |  |

# 5.2 路径识别部分

# 路径识别传感器我们选择的是 OV5116 模拟摄像头，高效的软件程序是智能车高速平稳自动寻线的基础，图像采集及校正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，我们使用了鲁棒性很好的经典 PID 控制算法，配合使用理论计算和实际参数补偿的办法，使智能车能够稳定快速寻线。

5.2.1原始图像的处理

在采集图像的过程中，由于感光芯片的误差、赛道杂点的影响，以及赛道

的反光作用，会对采集造成一定的干扰，因此原始图像处理首先要做的是去除这些错误信息的影响。去除了噪点等影响后，就要对图像畸变进行处理。由于摄像头架设有倾角，因此产生了几何透视关系，加上镜头本身的光学畸变，不能使用直接采集到的信息，直接采集的信息如图5.2 所示，这里的直道、弯道、十字交叉有严重的变形，容易产生误判。

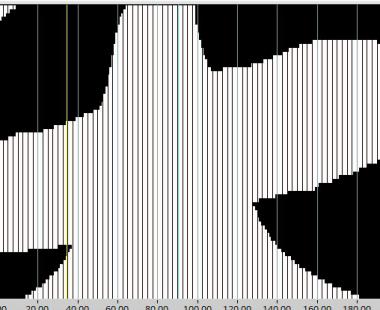
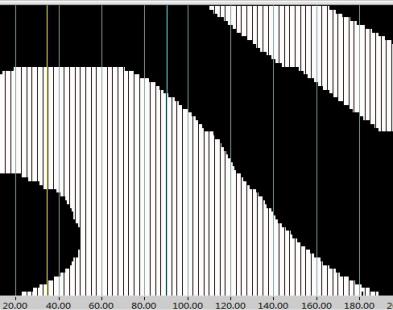
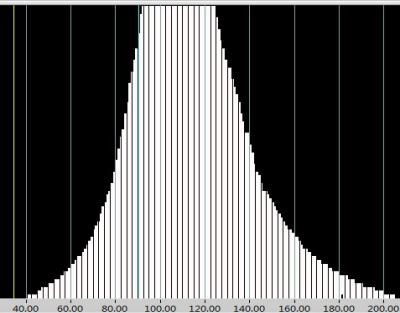
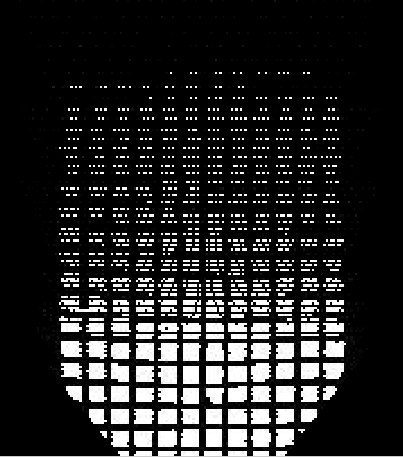
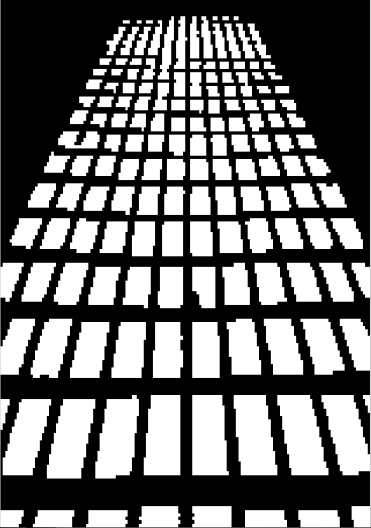


图5.2未处理图像

要对信息进行处理，最基本的处理是畸变矫正，在这里，畸变矫正包括两方面：一、光学畸变的校正，包括桶形失真等；二、透视变形的矫正，根据摄像头的视角进行校正，对校正图像进行修正。图5.3所示为原始图像经过光学畸变校正和几何畸变矫正之后的图像对比。这里，为了便于解释和观察，我们用摄像图拍摄了一个贴有网格的KT 板。(图5.3 畸变校正效果)



a）对梯形失真矫正前的效果 b ）对几何透视畸变进行矫正后的效果

5.2.2循迹处理

在基本图像处理完成后，就要从图像中提取出有效的赛道信息，也就是赛

道边缘。循迹算法思路如下：

 1)由视频分离电路获取行场同步信号，读取整幅图像的Buff值（即电

平值）；

 2)按照循迹算法开始循迹，对于前三行，采用由视野中心向两边找点

的方式，找出近处赛道边沿线；

 3)对于接下来的赛道，采用单边循迹的方式，确定阈值，在阈值范围

内找赛道边沿；

 4)对于十字，由于双侧赛道边沿的丢失，确定找点的初始位置，选择

由中间向两边找的方式找的路径，同时根据矫正图像滤除十字拐角线，从而

顺利连线；

 5)路障的处理，是通过记录黑块的行的个数来确定路障的位置，从而

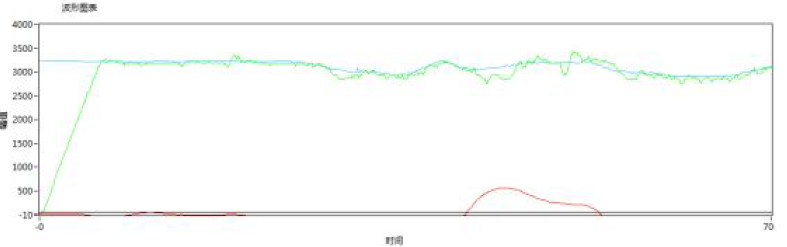
在控制上有效的避开路障；

6)直角的处理，是通过计算斜率来判断直角，从而控制舵机打较大的角度过直角；

7）中心引导线，是从视野了两边往中间扫描，判断中心引导线并控制舵机打角，让小车小车沿引导线行驶。

# 5.3 速度控制部分

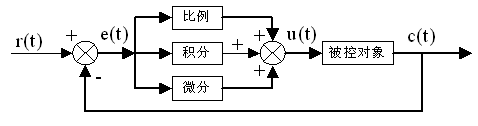
# 速度控制并不像方向控制那样精确，只要在短时间内达到目标速度即可，而误差是允许的，因为作为智能车这个惯性系统来说，速度不会徒增骤减（与控制周期比较），因此我们使用了P 控制，经过调试，选择合适的P 系数后，加速可以很快也不会过冲，控制效果很好，如图5.4.

5.4 速度控制测试曲线

5.3.1 方向控制的理论分析

在工程实际中，应用最为广泛的调节器控制规律为比例、积分、微分控制，简称PID控制，又称PID调节。PID控制器问世至今已有近70年历史，它以其结构简单、稳定性好、工作可靠、调整方便而成为工业控制的主要技术之一。当被控对象的结构和参数不能完全掌握，或得不到精确的数学模型时，控制理论的其它技术难以采用时，系统控制器的结构和参数必须依靠经验和现场调试来确定，这时应用PID控制技术最为方便。即当我们不完全了解一个系统和被控对象，或不能通过有效的测量手段来获得系统参数时，最适合用PID控制技术。PID控制，实际中也有PI和PD控制。

PID控制器是一种线性控制器，它根据给定值与实际输出值构成控制偏差。将偏差的比例(P)、积分(I)和微分(D)通过线性组合构成控制量，对被控对象进行控制，故称PID控制器，原理框图如图4.11所示。

图4.11 PID控制器原理框图

在计算机控制系统中，使用的是数字PID控制器，控制规律为：

 (公式4.1)

 (公式4.2)

式中

k——采样序号，k = 0，1，2…； r(k)——第k次给定值；

c(k)——第k次实际输出值； u(k)—— 第k次输出控制量；

e(k)—— 第k次偏差； e(k-1)—— 第k-1次偏差；

KP——比例系数； TI——积分时间常数；

TD——微分时间常数； T——采样周期。

简单说来，PID控制器各校正环节的作用如下：

比例环节：及时成比例地反映控制系统的偏差信号，偏差一旦产生，控制器立即产生控制作用，以减少偏差。

积分环节：主要用于消除静差，提高系统的无差度。积分作用的强弱取决于积分时间常数，越大，积分作用越弱，反之则越强。

微分环节：能反映偏差信号的变化趋势(变化速率)，并能在该偏差信号变得太大之前，在系统中引入一个有效的早期修正信号，从而加快系统的动作速度，减小调节时间。

数字PID控制算法通常分为位置式PID控制算法和增量式PID控制算法。

位置式PID

位置式PID中，由于计算机输出的u (k) 直接去控制执行机构(如阀门)，u(k)的值和执行机构的位置(如阀门开度)是一一对应的，所以通常称公式(4.2)为位置式PID控制算法。

位置式PID控制算法的缺点是：由于全量输出，所以每次输出均与过去的状态有关，计算时要对过去e(k)进行累加，计算机工作量大；而且因为计算机输出的u(k)对应的是执行机构的实际位置，如计算机出现故障，u(k)的大幅度变化，会引起执行机构位置的大幅度变化，这种情况往往是生产实践中不允许的，在某些场合，还可能造成严重的生产事故。因而产生了增量式PID 控制的控制算法，所谓增量式PID 是指数字控制器的输出只是控制量的增量△u(k)。

增量式PID

当执行机构需要的是控制量的增量(例如：驱动步进电机)时，可由式(4.2)

推导出提供增量的PID控制算式。由式(4.2)可以推出式(4.3)，式(4.2)减去式(4.3)可得式(4.4)。

(公式4.3)

(公式4.4)

式中；；

公式(4.4)称为增量式PID控制算法，可以看出由于一般计算机控制系统采用恒定的采样周期T，一旦确定了KP、TI 、TD，只要使用前后三次测量值的偏差，即可由式(4.4)求出控制增量。

增量式PID具有以下优点：

(1) 由于计算机输出增量，所以误动作时影响小，必要时可用逻辑判断的方法关掉。

(2) 手动/自动切换时冲击小，便于实现无扰动切换。此外，当计算机发生故障时，由于输出通道或执行装置具有信号的锁存作用，故能保持原值。

(3) 算式中不需要累加。控制增量△u(k)的确定仅与最近k次的采样值有关，所以较容易通过加权处理而获得比较好的控制效果。

但增量式PID也有其不足之处：积分截断效应大，有静态误差；溢出的影响大。使用时，常选择带死区、积分分离等改进PID控制算法。

PID参数整定

运用PID控制的关键是调整KP、KI、KD三个参数，即参数整定。PID参数的整定方法有两大类：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；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、易于掌握，在工程实际中被广泛采用。由于智能车系统是机电高耦合的分布式系统，并且要考虑赛道的具体环境，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修正，模型参数变化较为频繁，理论计算整定法可操作性不强，最终我们采用了

工程整定的方法。

5.4舵机的PID控制算法

从采集回来的图像中提取控制量来控制舵机的转向，实现智能车的自动循迹是整个程序算法的核心部分，只有舵机控制好了，才可能实现在不同的赛道

元素上迅速、稳定的前进。

5.5 辅助车模调试程序设计

辅助调试程序有两个部分，一是基于键盘显示与按键的PID 参数输入程序，二是基于蓝牙模块的无线通信数据发送程序。

第六章系统的开发环境与车模调试

在系统的设计制作和调试的过程中，不管是软件的开发还是硬件电路的仿

真和电路板的制作都离不开PC 机。所以对于PC 机上的各种辅助设计软件必须要有一定的熟悉程度。这样可以提高开发的效率。

## 6.1 IAR的使用

今年我们选定了恩智浦最新的32 位单片机，它的开发成了一个很难问题，我们选择了用IAR 来进行软件的开发。由于IAR 才刚开始支持恩智浦kinetis系列的单片机，所以没有很全的库可以使用，得从最原始的资料开始做起，包括单片机的要用到的所有的模块初始化代码。在老师的帮助指导下，最终完成了单片机的底层开发，逐渐熟悉了在IAR 下对kinetis系列单片机进行开发。

## 6.2 基于LABVIEW的上位机

在后期车模调试的过程中，为了让小车的速度更快，就需要很好了解车模运行的详细状态，而光靠人眼是无法完成对车模很好的监控的，尤其是速度、方向融合在一起之后，只有通过PC 机才能很好地监控小车的运行状态，并纪录数据以供分析。那么一个良好的上位机就显得格外重要了。我们选择了LABVIEW 来进行上位机的开发，对数据进行处理，纪录并用波形显示。



图6.1最终的用户界面

最终的用户界面如图6.1，本上位机可以显示4 个通道的数据，并可以保存这4 个通道的数据。在进一步的调试过程中，我们开发出用SD 卡记录数据的方式，来记录车模在行进过程中的图像采集以及各种运行状态中的数据记录，如图6.2。

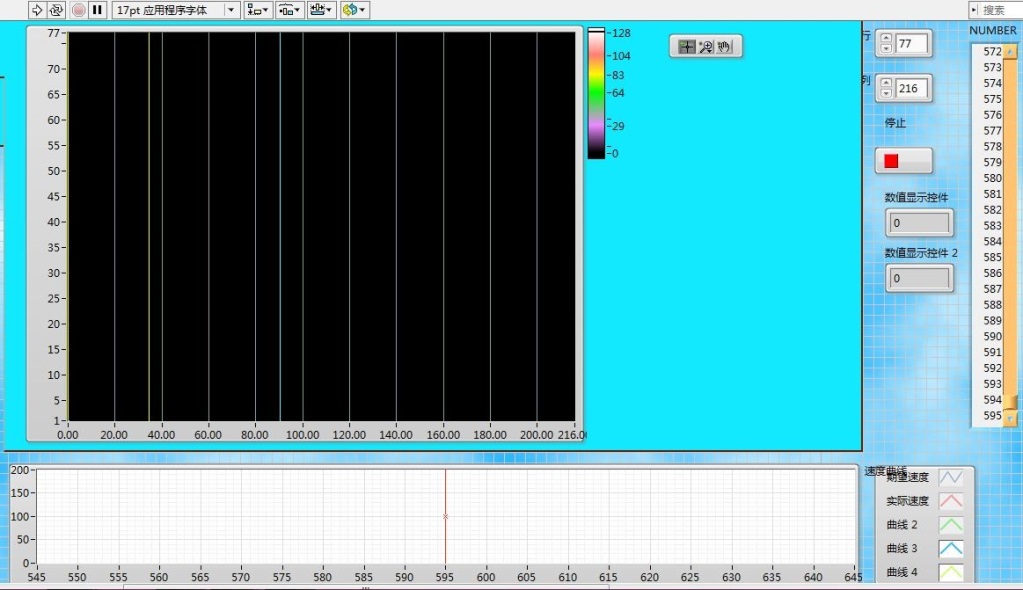


图6.2 SD 卡数据回放上位机界面

第七章总 结

7.1 总结

7.1.1 控制方面

总体来说对车模方向、速度的控制方案是在组委会提供的参考方案基础上设计的。在十字交叉、加减速、终点线、直角弯、障碍等特殊赛题上面进行了一些设计。然而对直角弯和障碍的处理，是我们设计的短板，没有做出更好的解决方案，从而使车模运行到这些位置的时候行驶明显不够流畅。这是我们以后需要改进的地方。

7.1.2.特色设计

1)所有硬件电路，包括测速传感器、摄像头，都是亲自绘制电路板，自己焊接制作，这样能更好的定制符合自己车模的电路模块；

2)摄像头图像的畸变矫正使图像处理更加方便简单；

3)自己编写的上位机，为车模调试提供了更多的调试手段。

## 7.2 结束语

通过数月的智能车制作，我们学到了很多实践能力，从方案选择、更改、重新设计到实施方案，细调参数，机械和硬件制作，软件编写，各方面都有深刻的体会，每一个细致的工作都会为小车这个系统的稳定性带来正面的影响，而每个疏漏，都会导致整个系统的不协调。在制作智能车的过程中，我们苦恼过，也因为小车跑的好而开心的笑过，我相信所有的智能车爱好者在制作小车以及智能控制小车的过程中，体会到的不仅仅是制作小车的乐趣，还有在制作

小车的过程中队友们的齐心协力。小车转过第一个弯道的记忆如今还深深地留存在我的脑海里，就像你倾心栽培的一株花终于开了花，也像一个蹒跚学步的小孩突然小跑起来，那种感觉很让人兴奋。为了小车跑的好，我们熬过无数夜，翻阅各种资料，

也曾在睡梦中无数次做梦梦到小车完美的直道加速、转弯、冲过十字道等等。如今回头看，欣喜地发现，在这个过程中，我们收获的不是一两点说的完的，不论是学知识，还是学做人，都是一次非常值得的经历。我们感谢恩智浦公司提供了如此一个好的平台来让我们展现自我，小车在赛道上的驰骋，无疑也是我们那颗年轻的心的驰骋!

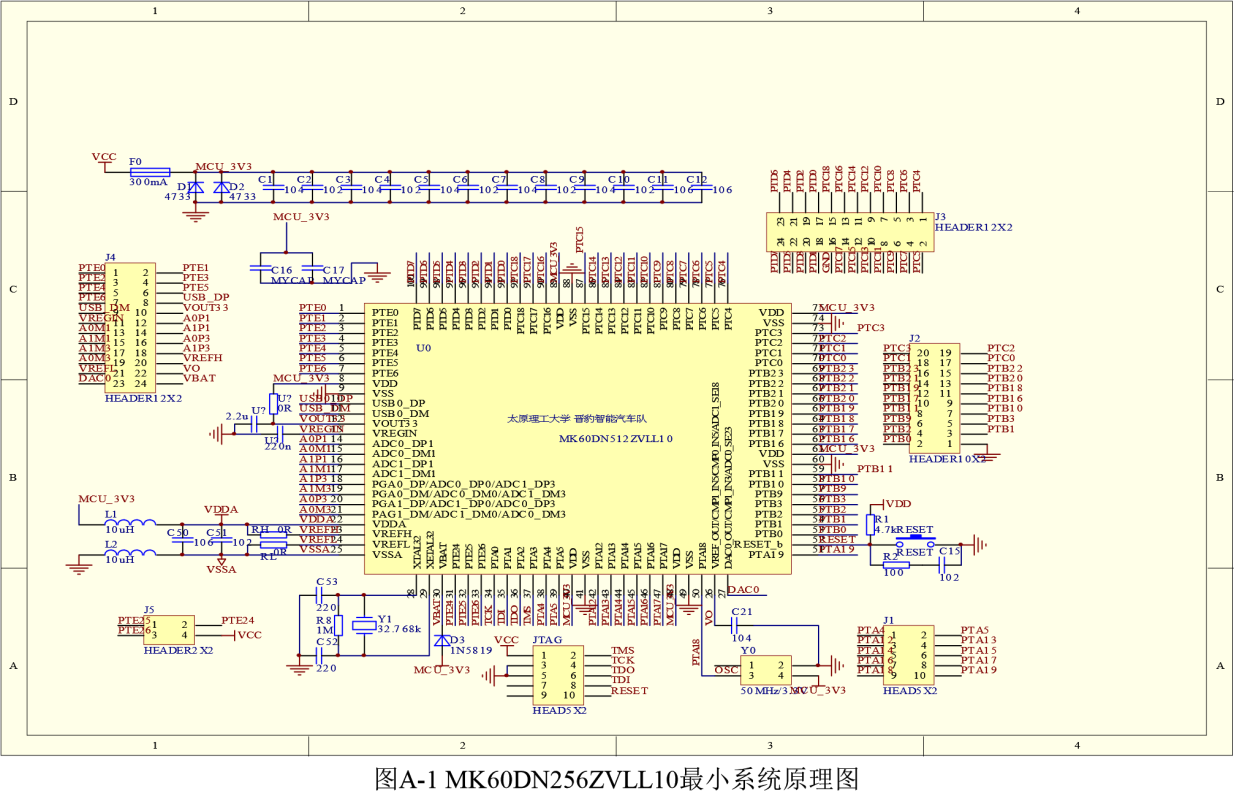
因为时间仓促，水平有限，有很多疏漏或者错误，也很遗憾没有更加细致的写出整个设计方案，衷心希望老师对我们的工作提出宝贵的意见，给我们指点，让我们在今后的工作学习中获取更多的进步。

致谢

借书写技术报告的机会向帮助过我们的萧宝瑾老师、同学，学校、学院领导以及组委会的老师，表示深深地谢意！更要向队内的队友表示深深地感谢，感谢陪伴在一起的日日夜夜！通过参赛，我们学会了很多！最后，向没有机会出现在队员名单中的幕后英雄致敬！

参考文献

1. 关于举办第十一届全国大学生“恩智浦”智能汽车竞赛的通知.
2. 第十一届全国大学生“恩智浦”杯智能汽车竞赛竞速比赛规则与赛场纪律.
3. 华成英, 童诗白. *模拟电子技术基础*. 北京, 高等教育出版社, 2006
4. 邵贝贝. *单片机嵌入式应用的在线开发方法*. 北京, 清华大学出版社, 2001
5. 刘慧银, 龚光华, 王云飞等. *Motorola(Freescale)微控制器MC68HC08原理及其嵌入式应用(新版)*. 北京, 清华大学出版社,1993
6. 卓晴, 黄开胜, 邵贝贝等. *学做智能车*. 北京, 北京航空航天大学出版社, 2001
7. 潘新民, 王燕芳. *微型计算机控制技术*. 北京, 高等教育出版社, 2001
8. 何立民. *单片机应用系统设计*. 北京, 北京航天航空大学出版社, 1997
9. Karl J.Astrom, BjormWittenmark. *Computer-Controlled Systems Theory and Design*[M]. 北京, 电子工业出版社，2001.
10. Freescale Semiconductor, Inc. MK60DN256ZVLL10 Sub-family Reference Manual[M], Rev.6, Nov. 2011
11. LM2940 datasheet
12. MC33886 datasheet
13. BTS7960 datasheet
14. 夏路易, 石宗义. *电路原理图与电路板设计教程Protel 99SE*[M]. 北京, 北京希望电子出版社
15. 曹丽，刘扬，刘伟.利用加速度计和陀螺仪的笔杆运动姿态的检测[M].仪器仪表学报
16. 阎石. *数字电子电路基础*. 北京, 高等教育出版社, 2005

附录 A MK60DN256ZVLL10 最小系统原理图

附录 B 程序源代码

void pit0\_isr(void)

{

DisableInterrupts;

//关总中断

if((PIT\_TFLG(0)&PIT\_TFLG\_TIF\_MASK)!=0)

{

ZJWW++; s

witch (ZJWW)

{

Case1: if(FLAG\_ALT==-1)

{

count\_ALT2++; if (count\_ALT2<=10)

{

if ( FlagPWMleft==0)

{

Current\_Speedleft+=get\_counter\_value();

}

else

{

Current\_Speedleft-=get\_counter\_value();}

} else

{

AdCurrent\_Speed\_left=Current\_Speedleft; count\_ALT2=0;

Current\_Speedleft=0;

AdCurrent\_Speed=(AdCurrent\_Speed\_left+AdCurrent\_Speed\_right);

AdCurrent\_Speedtest=AdCurrent\_Speed;

Speed\_Ctrl\_zlr(); Speedcounter=0; positionold=position;

}

lptmr\_pulse\_counter(LPTMR\_ALT1);

}

Else

{

count\_ALT1++; if (count\_ALT1<=10)

{

if ( FlagPWMright==0)

{

Current\_Speedright+=get\_counter\_value();

}

else {Current\_Speedright-=get\_counter\_value();}

} else

{

AdCurrent\_Speed\_right=Current\_Speedright; count\_ALT1=0;

Current\_Speedright=0;

} lptmr\_pulse\_counter(LPTMR\_ALT2);

}

FLAG\_ALT=0-FLAG\_ALT; break;

case 2:

GetSAD(); GetAD();

break;

case 3:

AngleCalculate(); AngleControl(); break;

case 4: Speedcounter++;

Speedpwmcaculate(); break;

case 5:

DirecionCalculate();

DirectionControl();

SetMotorVoltage(); ZJWW=0; break;

default: break;

}

PIT\_TFLG(0)|=PIT\_TFLG\_TIF\_MASK; //清标志

}

EnableInterrupts; //开总中断 }

voidAngleCalculate(void)

{

FloatfDeltaValue=0; if(SpeedErrNew<spdcontrol) angle\_center=angle\_center\*SpeedErrNew\* Cp; else angle\_center=angle\_center;

Angle\_current=(Angle\_Center-angle\_center-Angle\_AD)\*180;

Angle\_currenttest=Angle\_current+10000;

AngleSpeed\_current=(AngleSpeed\_AD-Angle\_Speed\_Center)\*5;

Angle\_current\_renew[1]=Angle\_current\_renew[0]; Angle\_current\_renew[0]=Angle\_current; if(Angle\_current\_renew[1]>0&&Angle\_current\_renew[0]<0)

{STARTFLAG=1;} if(STARTFLAG==1)

{

Angle=AngleSpeed\_Integral;

Angletest=Angle+10000;

fDeltaValue =(Angle\_current - Angle)\*25;

AngleSpeed\_Integral+=(AngleSpeed\_current + fDeltaValue);

}

}

voidSpeed\_Ctrl\_zlr()

{

intCarSpeed;

floatfP; float fD;

floatfI;

SpeedErrOld=SpeedErrNew;

SpeedPWMold=SpeedPWMnew;

CarSpeed = AdCurrent\_Speed;

SpeedErrNew=DesiredSpeed-CarSpeed; fP=SP\*SpeedErrNew; fD=Sd\*(SpeedErrNew-SpeedErrOld); if(STARTFLAG==1)

{fI+=Si\*SpeedErrNew\*0.1;}

SpeedPWMnew=fP+fD+fI;

}

voidSpeedpwmcaculate()

{

float SpeedPWMDeltaValue;

SpeedPWMDeltaValue=SpeedPWMnew-SpeedPWMold;

SpeedPWM=SpeedPWMDeltaValue\*Speedcounter+SpeedPWMold;}

void main(void)

{

inti=0;

DisableInterrupts; //禁止总中断init\_all();

keyboardinput();

EnableInterrupts;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/ delay(15000);

GPIOA\_PDOR |= GPIO\_PDOR\_PDO(GPIO\_PIN(15));

for(i=0;i<40;i++)

{

hw\_adc\_convertstart(0, 19, 12);

while(( ADC0\_SC1A & ADC\_SC1\_COCO\_MASK ) != ADC\_SC1\_COCO\_MASK);

{

Angle\_Speed\_Center+=ADC0\_RA;

ADC0\_SC1A &= ~ADC\_SC1\_COCO\_MASK;

}

}

Angle\_Speed\_Center=Angle\_Speed\_Center\*0.025;

delay(1500);

GPIOA\_PDOR &= ~GPIO\_PDOR\_PDO(GPIO\_PIN(15)); delay(15000);

hw\_pit\_init(0,50000); enable\_pit\_interrupt(PIT0);

GPIOA\_PDOR |= GPIO\_PDOR\_PDO(GPIO\_PIN(15));

for(;;)

{

//Parameter\_sent();

keyboard();

}

}