

第二届“飞思卡尔”杯 全国大学生智能汽车邀请赛

技 术 报 告



学 校： 上海交通大学

队伍名称： CyberSmart 队

参赛队员： 胡晨晖

陆佳南

陈立刚

带队教师： 杨 明

摘 要

本文介绍了Cyber Smart队员们在准备第二届Freescale智能车大赛过程中的工作成果。智能车的硬件平台采用带MC9S12DG128 处理器的S12 环境，软件平台为CodeWarrior IDE 开发环境，车模采用大赛组委会统一提供的1：10 的仿真车模。文中介绍了智能小车控制系统的软硬件结构和开发流程。

整个系统涉及车模机械结构调整、传感器电路设计及信号处理、控制算法和策略优化等多个方面。为了提高智能赛车的行驶速度和可靠性，对比了不同方案的优缺点，并结合Labview 仿真平台进行了大量底层和上层测试，最终确定了现有的系统结构和各项控制参数。

关键字：智能车，摄像头，视频采样，速度控制

Abstract

In this article, we present the results of our team members' working during the period of preparing the second session of the Freescale Intelligence Car Competition. The hardware platform of the smart car bases on S12 development board which features MC9S12DG128 micro-controller, Code warrior IDE is used as its software development tool, and 1/10 EP on-racing car is adopted as its mechanism. The article gives introduction to the Hardware/Software structure and the designing process of the car's control system.

The whole system includes the aspects of the mechanism structure adjustment, the sensor circuit design and signal process, control algorithm and strategy optimization etc. In order to increase the speed and the reliability of the car, the advantage and disadvantage of the different schemes are compared, and a great number of the bottom layer and the upper layer tests are carried on combined with the Labview simulation platform. At last, the current system structure and each control parameters are determined.

Key Word: Intelligent car, Camera, Video Sample, Speed Control

目 录

摘 要	2
第一章 绪论	6
1.1 研究背景	6
1.1.1 智能车的发展历史	6
1.1.2 应用前景	7
1.2 智能车大赛介绍	8
1.2.1 大赛简介	8
1.2.2 比赛规则	8
1.2.3 规则分析	11
第二章 主要思路及技术方案概要	13
2.1 系统硬件结构	13
2.2 系统软件结构	14
第三章 模型车机械设计说明	15
3.1 车体机械建模	15
3.2 前轮倾角的调整	16
3.3 齿轮传动机构调整	17
3.4 后轮差速机构调整	18
3.5 舵机安装方式	19
3.6 其他机械模块调整	19
第四章 电路设计说明	20
4.1 驱动电路板	20
4.1.1 系统电源模块	21
4.1.2 电机驱动模块	22
4.2 核心控制板	25
第五章 视频采集和图象处理	28
5.1 视频采集	28
5.1.1 摄像头工作原理	28
5.1.2 摄像头选择	29
5.1.3 摄像头安装	30
5.1.4 信号分离电路	32
5.1.5 摄像头初始化	34
5.2 图像处理	35
5.2.1 目标指引线的特征位置	35
5.2.2 二值化算法	36
5.2.3 边缘检测算法	39
5.2.4 跟踪边缘检测算法	44
5.2.5 用摄像头识别坡道	50

5.2.6 双摄像头的研究	51
5.2.7 摄像头镜头畸变	52
5.2.8 超频试验	55
第六章 车体控制	57
6.1 速度控制	57
6.1.1 速度传感器模块	57
6.1.2 速度传感器初始化	59
6.1.3 速度获取	59
6.1.4 速度传感器准确性测试	60
6.2 舵机控制	60
6.2.1 简单沿线行驶策略	61
6.2.2 最佳行驶路线策略	63
6.3 加速度控制	64
6.3.1 工作方式:	64
6.3.2 干扰因素	64
6.3.3 安装方式分析	65
6.3.4 坡道检测	65
6.3.5 分析结论	66
6.4 赛道记忆	66
6.4.1 路径记录方式	67
6.4.2 数据存储	68
6.4.3 第二圈控制优化	70
6.4.4 问题和局限	71
第七章 开发工具、制作调试过程	72
7.1 软件开发平台Codewarrior IDE.....	72
7.1.1 Codewarrior IDE 功能介绍.....	72
7.1.2 Codewarrior IDE 基本使用方法.....	73
7.2 辅助调试方法	76
7.3 无线调试模块	77
7.3.1 Zigbee无线通信模块.....	77
7.3.2 天线的摆放	80
7.4 数码开关	81
7.4.1 原理简介	81
7.4.2 程序使用	82
7.4.3 操作方法	83
7.5 计时器	83
第八章 结论	85
参考文献	87
鸣谢	88
附录	89

第一章 绪论

1.1 研究背景

1.1.1 智能车的发展历史

1953 年，美国 Barrett Electric 公司制造了世界上第 1 台采用埋线电磁感应方式跟踪路径的自动导向车，也被称作“无人驾驶牵引车”。20 世纪 60 年代和 70 年代初，AGV 仍采用这种导向方式。但是，20 世纪 70 年代中期，具有载货功能的 AGV 在欧洲得到了应用并被引入到美国。这些自动导向车主要用于自动化仓储系统和柔性装配系统的物料运输。在 20 世纪 70 年代和 80 年代初，AGV 的应用领域扩大而且工作条件也变得多样化，因此，新的导向方式和技术得到了更广泛的研究与开发。

在最近的 10-15 年里，各种新型 AGV 被广泛地应用于各个领域。单元式 AGV 主要用于短距离的物料运输并与自动化程度较高的加工设备组成柔性生产线。例如，自动导向叉车用于仓储货物的自动装卸和搬运；小型载货式 AGV 用于办公室信件的自动分发和电子行业的装配平台。除此以外，AGV 还用于搬运体积和重量都很大的物品，尤其是在汽车制造过程中用多个载货平台式 AGV 组成移动式输送线，构成整车柔性装配生产线。最近，小型 AGV 应用更为广泛，而且以长距离不复杂的路径规划为主。AGV 从仅由大公司应用，正向小公司单台应用转变，而且其效率和效益更好。至此出现了智能车的概念

下表是 1993~1997 年 AGV 在国外各个行业的销售情况。

表1-1 1993~1997年AGV在国外各个行业的销售情况（%）

年份	1993	1994	1995	1996	1997
汽车制造	27.2	27.6	28.1	28.6	29.1
航空国防	10.6	9.9	9.2	8.5	7.9
电子行业	12.1	12.3	12.5	12.6	12.8
服务行业	20.7	21.0	21.3	21.6	22

工程机械	8.7	8.6	8.6	8.6	8.6
其 他	20.7	20.6	20.3	20.1	19.6

1.1.2 应用前景

城市公共交通是与人民群众生产生活息息相关的重要基础设施。然而，目前世界上许多大城市都面临着由私人汽车过度使用而带来的诸多问题，例如道路堵塞、停车困难、能源消耗、噪声污染和环境污染等，这些问题严重降低了城市生活的质量。优先发展城市公共交通是提高交通资源利用效率，缓解交通拥堵的重要手段。国务院总理温家宝于 2005 年 10 月做出重要批示，要求优先发展城市公共交通，这是贯彻落实科学发展观和建设节约型社会的重要举措。大容量城市公共交通，如地铁、轻轨等，其最大优点是空间利用率和能源利用率较高。然而，由于缺乏足够的时间、空间、运力灵活性，在客流量不足的情况下，系统效率将大大降低，运营成本过高，难以大力推广和应用。回顾汽车发展的百年历史，不难发现其控制方式从未发生过根本性改变，即由人观察道路并驾驶车辆，形成“路一人一车”的闭环交通系统。随着交通需求的增加，这种传统车辆控制方式的局限性日益明显，例如安全性低（交通事故）和效率低（交通堵塞）。最新调查表明，95%的交通事故是由人为因素造成，交通堵塞也大都与驾驶员不严格遵守交通规则有关。如果要从根本上解决这一问题，就需要将“人”从交通控制系统中请出来，形成“车一路”闭环交通系统，从而提高安全性和系统效率。这种新型车辆控制方法的核心，就是实现车辆的智能化。

智能车有着极为广泛的应用前景。结合传感器技术和自动驾驶技术可以实现汽车的自适应巡航并把车开得又快又稳、安全可靠；汽车夜间行驶时，如果装上红外摄像头，就能实现夜晚汽车的安全辅助驾驶；他也可以工作在仓库、码头、工厂或危险、有毒、有害的工作环境里，此外他还能担当起无人值守的巡逻监视、物料的运输、消防灭火等任务。在普通家庭轿车消费中，智能车的研发也是很有价值的，比如雾天能见度差，人工驾驶经常发生碰撞，如果用上

这种设备，激光雷达会自动探测前方的障碍物，电脑会控制车辆自动停下来，撞车就不会发生了。

1.2 智能车大赛介绍

1.2.1 大赛简介

受教育部高等教育司委托，高等学校自动化专业教学指导委员会负责主办全国大学生智能车竞赛。该项比赛已列入教育部主办的全国五大竞赛之一。首届“飞思卡尔”杯全国大学生智能车邀请赛已于去年八月在清华大学举行。此项赛事，在韩国已经举办过多届，其专业知识涉及控制、模式识别、传感技术、汽车电子、电气、计算机、机械等诸多学科，对学生的知识融合和动手能力的培养，对高等学校控制及汽车电子学科学术水平的提高，具有良好的推动作用。

今年已是第二届比赛，在本次比赛中，参赛选手仍须使用大赛组委会统一提供的竞赛车模，以 Freescale 公司生产的 16 位微控制器 MC9S12DG128B 作为核心控制单元，自主构思控制方案及系统设计，包括传感器信号采集处理、控制算法及执行、动力电机驱动、转向舵机控制等，最终实现一套能够自主识别路线，并且可以实时输出车体状态的智能车控制软硬件系统。各参赛队完成智能车工程制作及调试后，于指定日期与地点参加比赛。参赛队伍之名次以赛车现场成功完成赛道比赛时间为主，技术方案及工程制作质量为辅来决定。但与去年不同的是，今年将先分五个赛区进行预赛选拔，然后再进行决赛，决赛地点定于我校。同时，今年的赛道还将增加坡道，难度有所加大。本文所述的内容即是为本届比赛而准备的技术方案。

1.2.2 比赛规则

参赛选手须使用大赛组委会统一提供的竞赛车模，采用飞思卡尔 16 位微控

制器 MC9S12DG128 作为核心控制单元，自主构思控制方案及系统设计，包括传感器信号采集处理、控制算法及执行、动力电机驱动、转向舵机控制等，完成智能车工程制作及调试，于指定日期与地点参加场地比赛。参赛队伍之名次（成绩）由赛车现场成功完成赛道比赛时间为主，技术方案及制作工程质量评分为辅来决定。组委会制定如下竞赛规则，在实际可操作性基础上力求公正与公平参与。组委会将邀请独立公证人监督现场赛事及评判过程。

比赛过程中，如果赛车碰到赛道两边的立柱并使之倾倒或移动，裁判员将判为赛车冲出跑道。赛车前两次冲出跑道，选手可以申请恢复比赛，即将冲出跑到赛车重新放置在裁判指定的赛车冲出跑道的位置，恢复比赛。整个恢复比赛过程中计时不间断。选手也可以在赛车冲出跑道后放弃比赛。

比赛过程中如果出现有如下一种情况，判为比赛失败：

- 1) 裁判点名后，2 分钟之内，参赛队没有能够进入比赛场地并做好比赛准备；
- 2) 比赛开始后，赛车在 30 秒之内没有离开出发区；
- 3) 赛车在离开出发区之后 10 分钟之内没有跑完两圈；
- 4) 赛车冲出跑道的次数超过两次；
- 5) 比赛开始后未经裁判允许，选手接触赛车；
- 6) 决赛前，赛车没有通过技术检验。

如果比赛失败，则不计成绩。

比赛中禁止：

- 1) 不允许在赛道周围安装辅助照明设备及其它辅助传感器等；
- 2) 选手进入赛场后，不允许进行任何硬件和软件的修改；
- 3) 比赛场地内，除了裁判与 1 名队员之外，不允许任何其他人员进入场地；
- 4) 不允许其它影响赛车运动的行为。

对于智能竞赛车模

- 1) 禁止改动车底盘结构、轮距、轮径及轮胎；
- 2) 禁止改动驱动电机的型号及传动比；

- 3) 禁止改造滚珠轴承;
- 4) 禁止改动舵机;
- 5) 禁止改动驱动电机以及电池, 车模主要前进动力来源于车模本身直流电机及电池;
- 6) 为了车模的行驶可以安装电路、传感器等, 允许在底盘上打孔或安装辅助支架等。

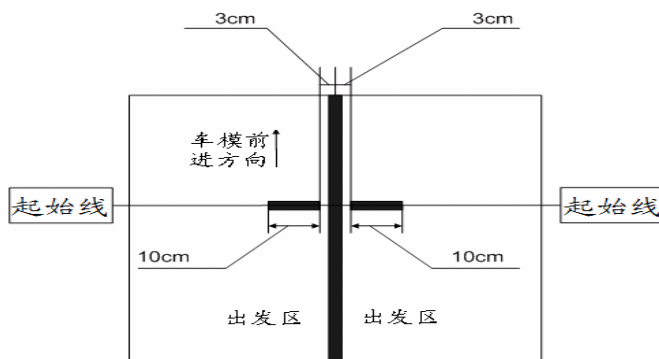
电路器件及控制驱动电路限制

- 1) 核心控制模块可以采用组委会提供的 HCS12 模块, 也可以采用 MC9SDG128 自制控制电路板, 除了 DG128MCU 之外不得使用辅助处理器以及其它可编程器件;
- 2) 伺服电机数量不超过 3 个;
- 3) 传感器数量不超过 16 个(红外传感器的每对发射与接受单元计为 1 个传感器, CCD 传感器记为 1 个传感器);
- 4) 直流电源使用大赛提供的电池;
- 5) 禁止使用 DC-DC 升压电路为驱动电机以及舵机提供动力;
- 6) 全部电容容量和不得超过 2000 微法; 电容最高充电电压不得超过 25 伏。可以选择参数:
 - 1) 开发软件可以选择 CodeWarrior 3.1, 也可以另行选择;
 - 2) 开发调试硬件可以选择 BDM(清华大学制作)工具, 也可以另行选择;
 - 3) 电路所使用元器件(传感器、各种信号调理芯片、接口芯片、功率器件等)种类与数量都可以自行设计选择。

赛道基本参数(不包括拐弯点数目、位置以及整体布局)

- 1) 赛道路面用纸制作, 跑道所占面积不大于 5000mm* 7000mm, 跑道宽度不小于 600mm;
- 2) 跑道表面为白色, 中心有连续黑线作为引导线, 黑线宽 25mm;
- 3) 跑道最小曲率半径不小于 500mm;
- 4) 跑道可以交叉, 交叉角为 90° ;

- 5) 赛道为二维水平平面；
- 6) 赛道有一个长为 1000mm 的出发区，如下图所示，计时起始点两边分别有一个长度 100mm 黑色计时起始线，赛车前端通过起始线作为比赛计时开始或者与结束时刻。



车模照片如下：

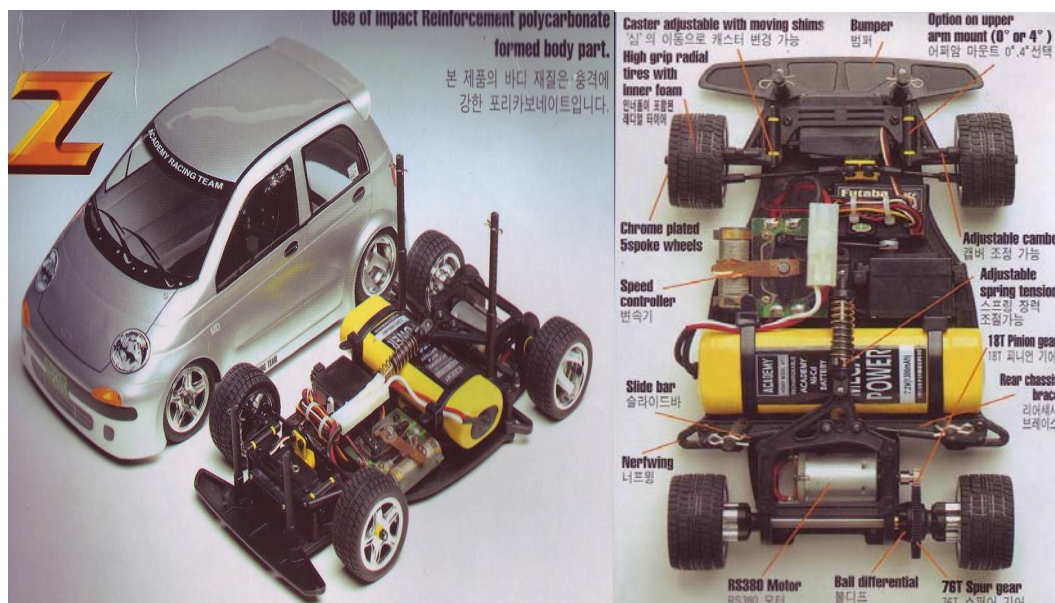


图 1-1 智能车大赛车辆模型

1. 2. 3 规则分析

从大赛的规则上来看我们的选择已被限制在一个很小的范围内。车模本身的机械结构是禁止做改动的。电路板的控制芯片也只能用 freescale 的

s12dg128, 而且不能添加辅助处理器和存储电路。最多只能使用 3 个伺服电机, 除去一个用于舵机, 一个用于后轮驱动, 只有一个伺服电机可供自由使用。在对电机操作时我们也不能使用升压芯片, 电容最高的充电电压不能超过 25 伏, 在这个意义上就是说各个队的车速不会相差太明显。因此我们队得出这样一个结论, 要让我们的智能车在赛道上有较好的表现, 关键就在控制方法和控制算法上。

第二章 主要思路及技术方案概要

2.1 系统硬件结构

此智能车辆定位系统用摄像头拍摄车辆前方的赛道，通过 MC9S12DG128 采样视频信号，获得图像数据。然后用合适的算法，如跟踪边缘检测算法，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。本智能车辆定位系统的结构图如图 2-1 所示。。

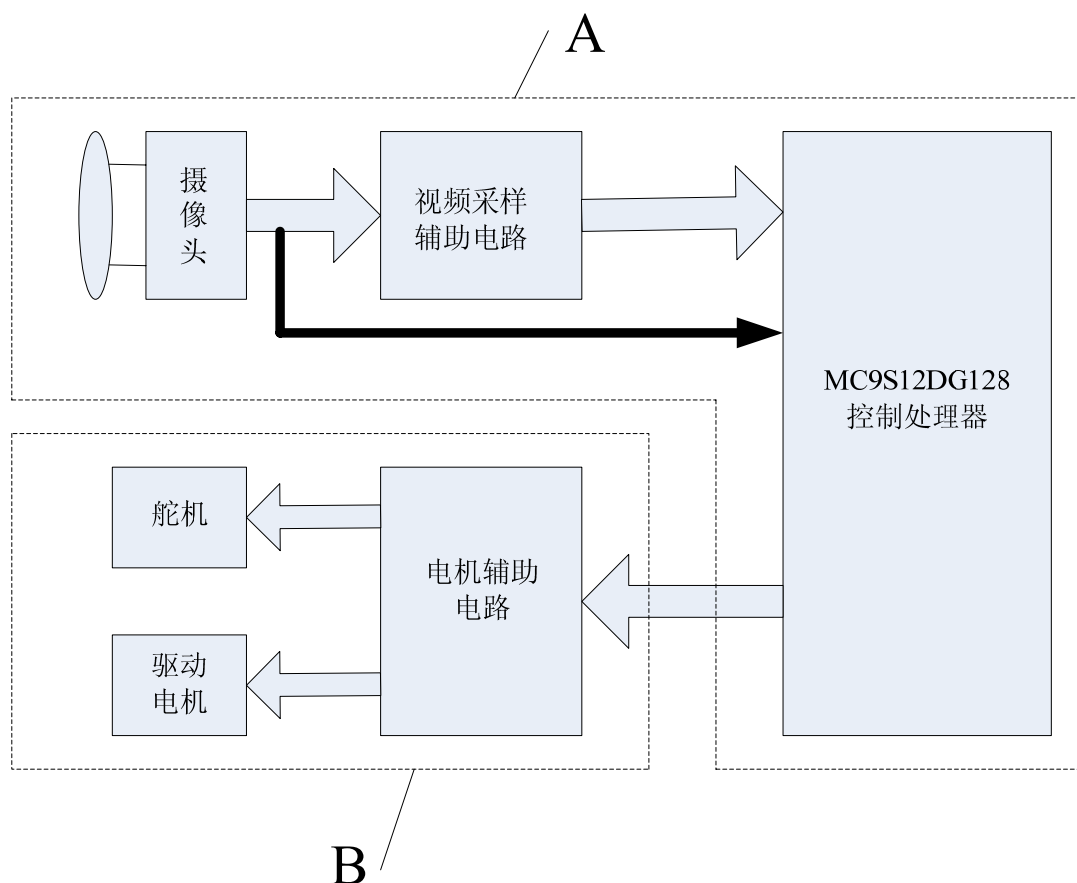


图 2-1 智能车辆定位系统结构图

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，但是一些机械结构上的细节仍然会对小车性能产生影响，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编码器和摄像头，它们可以完成赛道信息的采集和小车行驶参数的获取，算法部分主要涉及到 PID 控制算法和预判算法等，这部分将在后面详述。

2.2 系统软件结构

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求来制定。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赛道的图像数据，同时通过速度传感器模块获取赛车的速度。采用 PID 对舵机进行反馈控制。另外根据检测到的速度，结合速度控制策略，对赛车速度不断进行适当调整，使赛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 2-2 所示。

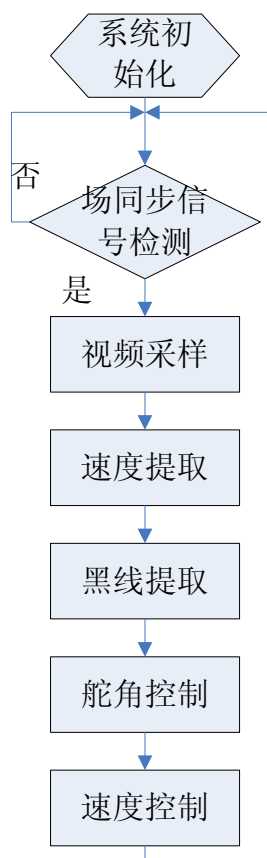


图 2-2 系统软件结构

第三章 模型车机械设计说明

任何的控制算法和软件程序都是需要一定的机械结构来执行和实现的，因此在设计整个软件架构和算法之前一定要对整个车模的机械结构有一个感性的认识，然后建立相应的数学模型。从而再针对具体的设计方案来调整赛车的机械结构。本章的将主要介绍赛车车模的机械特点和调整方案。

3.1 车体机械建模

此次比赛选用的赛车车模采用 1/10 的仿真车模。赛车机械结构只使用竞赛提供车模的底盘部分及转向和驱动部分。控制采用前轮转向，后轮驱动方案。具体车模数据如下：

表 3-1 车模基本尺寸参数

基本参数	尺寸
轴距	197cm
前轮距	124cm
后轮距	136cm
车轮直径	50cm
车长	316mm
车宽	172m
传动比	18/76

如 图 3-1 所示：其中虚线部分为轮胎，A点为右轮的转动轴点，同理，对应左侧相应位置。

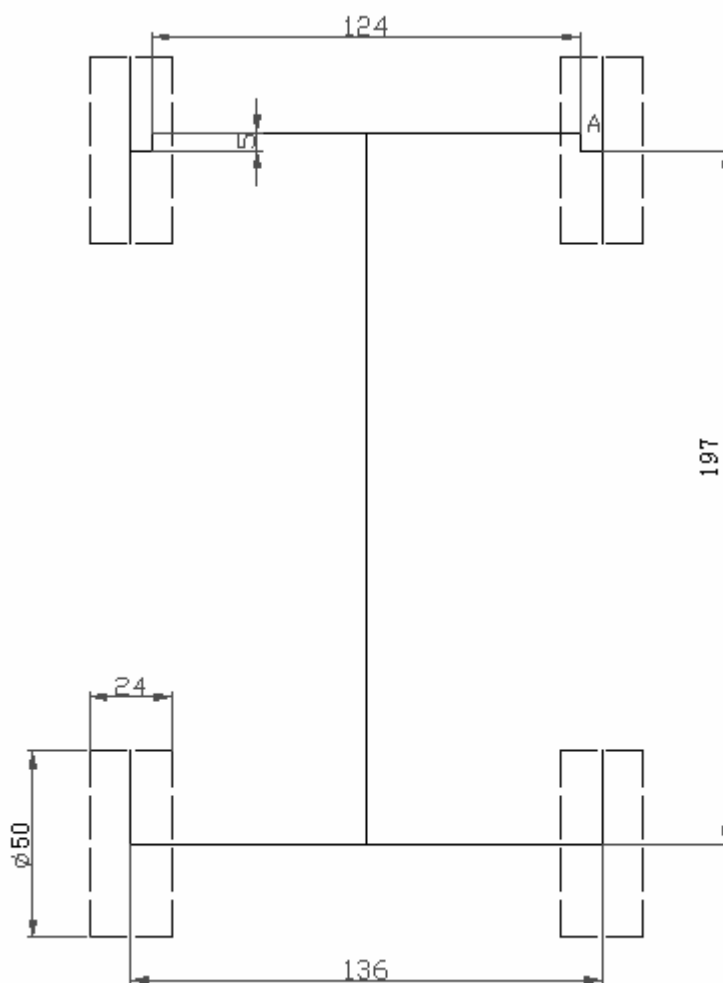


图 3-1 车模尺寸示意图

3.2 前轮倾角的调整

调试中发现，在车模过弯时，转向舵机的负载会因为车轮转向角度增大而增大。为了尽可能降低转向舵机负载，对前轮的安装角度，即前轮定位进行了调整。前轮定位的作用是保障汽车直线行驶的稳定性，转向轻便和减少轮胎的磨损。前轮是转向轮，它的安装位置由主销内倾、主销后倾、前轮外倾和前轮前束等 4 个项目决定，反映了转向轮、主销和前轴等三者在车架上的位置关系。

主销内倾是指主销装在前轴略向内倾斜的角度，它的作用是使前轮自动回

正。角度越大前轮自动回正的作用就越强烈，但转向时也越费力，轮胎磨损增大；反之，角度越小前轮自动回正的作用就越弱。

主销后倾是指主销装在前轴，上端略向后倾斜的角度。它使车辆转弯时产生的离心力所形成的力矩方向与车轮偏转方向相反，迫使车轮偏转后自动恢复到原来的中间位置上。由此，主销后倾角越大，车速越高，前轮稳定性也愈好。

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，因此高速时后倾的回正作用大，低速时内倾的回正作用大。

前轮外倾角对汽车的转弯性能有直接影响，它的作用是提高前轮的转向安全性和转向操纵的轻便性。前轮外倾角俗称“外八字”，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外八字角度，这个角度约在 1° 左右。

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，其惯性力会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。

3.3 齿轮传动机构调整

车模后轮采用 RS-380SH-4045 电机驱动，由竞赛主办方提供。电机轴与后轮轴之间的传动比为 9:38（电机轴齿轮齿数为 18，后轮轴传动轮齿数为 76）。

齿轮传动机构对车模的驱动能力有很大的影响。齿轮传动部分安装位置的不恰当，会大大增加电机驱动后轮的负载，从而影响到最终成绩。调整的原则是：两传动齿轮轴保持平行，齿轮间的配合间隙要合适，过松容易打坏齿轮，过紧又会增加传动阻力，白白浪费动力；传动部分要轻松、顺畅，容易转动，不能有卡住或迟滞现象。

判断齿轮传动是否调整好的一个依据是,听一下电机带动后轮空转时的声音。声音刺耳响亮,说明齿轮间的配合间隙过大,传动中有撞齿现象;声音闷而且有迟滞,则说明齿轮间的配合间隙过小,或者两齿轮轴不平行,电机负载加大。调整好的齿轮传动噪音小,并且不会有碰撞类的杂音。

3.4 后轮差速机构调整

差速机构的作用是在车模转弯的时候,降低后轮与地面之间的滑动;并且还可以保证在轮胎抱死的情况下不会损害到电机。

当车辆在正常的过弯行进中(假设:无转向不足亦无转向过度),此时4个轮子的转速(轮速)皆不相同,依序为:外侧前轮>外侧后轮>内侧前轮>内侧后轮。此次所使用车模配备的是后轮差速机构。差速器的特性是:阻力越大的一侧,驱动齿轮的转速越低;而阻力越小的一侧,驱动齿轮的转速越高。以此次使用的后轮差速器为例,在过弯时,因外侧前轮轮胎所遇的阻力较小,轮速便较高;而内侧前轮轮胎所遇的阻力较大,轮速便较低。

差速器的调整中要注意滚珠轮盘间的间隙,过松过紧都会使差速器性能降低,转弯时阻力小的车轮会打滑,从而影响车模的过弯性能。好的差速机构,在电机不转的情况下,右轮向前转过的角度与左轮向后转过的角度之间误差很小,不会有迟滞或者过转动情况发生。

3.5 舵机安装方式

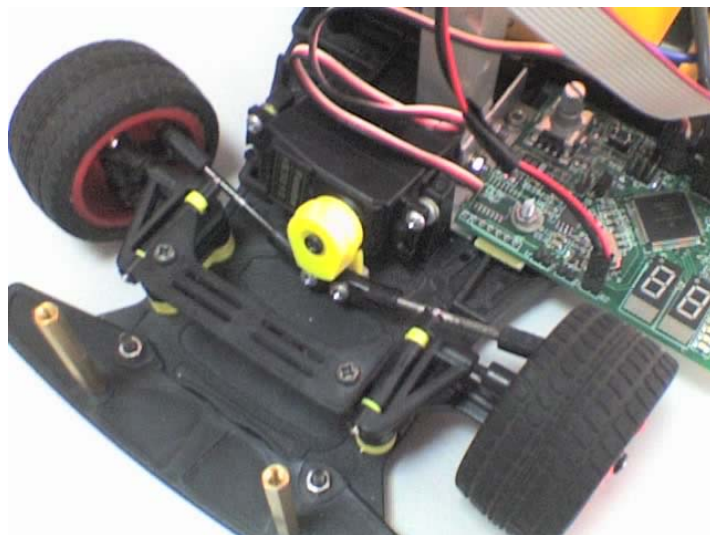


图3-2 舵机安装实物图

舵机转向是整个控制系统中延迟较大的一个环节，为了减小此时间常数，通过改变舵机的安装位置，而并非改变舵机本身结构的方法可以提高舵机的响应速度。分析舵机控制转向轮转向的原理可以发现，在相同的舵机转向条件下，转向连杆在舵机一端的连接点离舵机轴心距离越远，转向轮转向变化越快。这相当于增大力臂长度，提高线速度。针对上述特性，改变了去年舵机的安装方式，将舵机安装在了相对靠后的位置。这样安装的优点是：1) 改变了舵机的力臂，使转向更灵敏2) 舵机安装在了正中央，使左右转向基本一致 3) 重心更靠后，减轻了舵机负载。

3.6 其他机械模块调整

在调试过程中，除去对以上部分的调整以外，其他机械部分也要酌情调整。悬挂臂、转向连杆的动作要灵活自如，准确无误；主悬架要松紧适度；驱动电机的螺丝一定要上紧，并要经常检查，一旦在行驶中松动就会造成零件的损坏。调试中还发现主悬架的第二个螺丝容易松动，测试时要检查是否上紧。

第四章 电路设计说明

系统硬件电路由驱动电路板和核心控制板两部分组成，本章将先对这两个电路进行总体介绍，然后分模块详细说明各部分的功能

4.1 驱动电路板

驱动电路板（如图 4-1 所示）安装在车体后部，它主要负责系统各部分电压的分配以及电机的驱动。由于这部分电路功耗较大，单独设计一块电路板后可使散热性能更好。

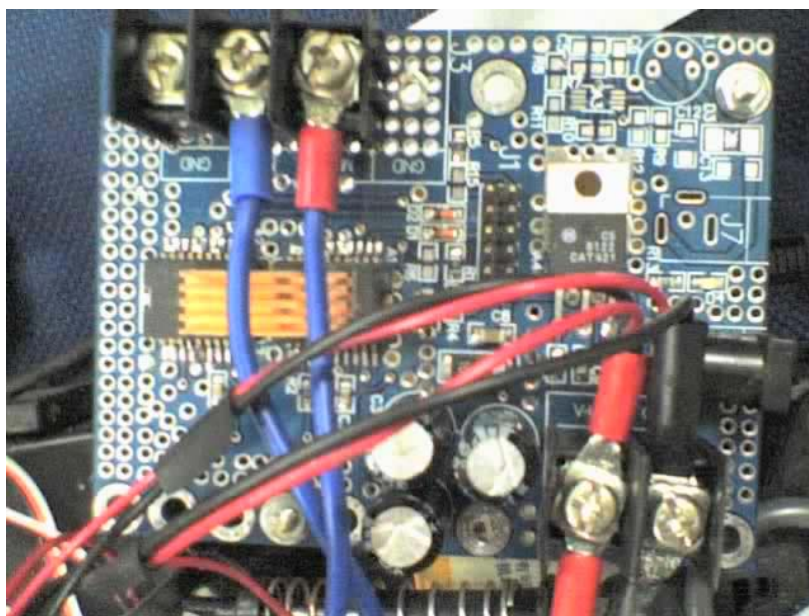


图 4-1 驱动电路板安装位置

驱动电路板的原理图见附表，其各个端口的分配情况如下：

- 1) J1的1-3口提供VCC；4口为使能信号，控制电机运转；6口状态位查询，检查电机运转情况；9-10口提供PWM，控制电机运转方向和运转速度。
- 2) J2, J5 电池电源输入，J5为并联备用。
- 3) J3、J4 通过比较2路PWM输入控制电机正反转。

- 4) J7 给摄像头提供工作电源，由于选用的摄像头额定工作电压为8V，在电池电压高于7.2V 的范围内能可靠工作。
- 5) A1、A2 33886运行电机，2路并联输入，4路并联输出，可以提高输入和输出的电流值，提高系统稳定性。

4.1.1 系统电源模块

驱动电路板中的电源模块为系统其它各个模块提供所需要的电源。设计中，除了需要考虑电压范围和电流容量等基本参数之外，还要在电源转换效率、降低噪声、防止干扰和电路简单等方面进行优化。可靠的电源方案是整个硬件电路稳定可靠运行的基础。

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，将充电电池电压转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、信号调理电路以及部分接口电路（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。采用摄像头进行道路检测时，需要12V工作电源。

整个电源模块的电路结构为：

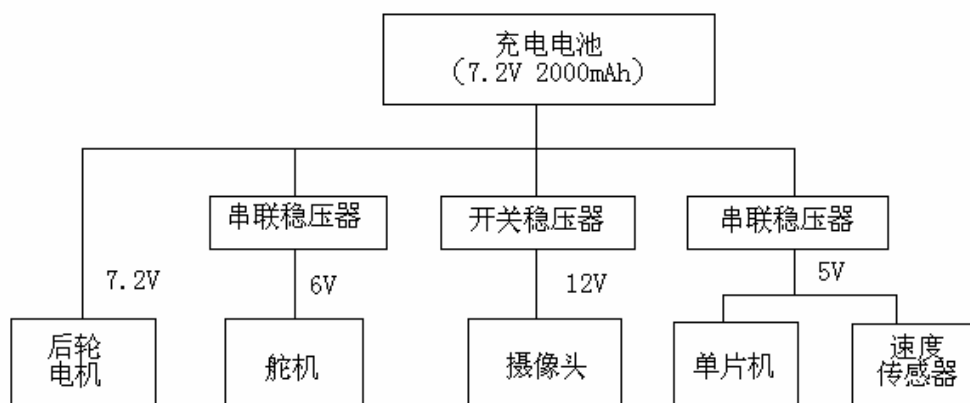


图4-2 电源模块电路结构

4.1.2 电机驱动模块

目前电路板上将两片33886并联，PWM信号一路直接输入到A1，一路经过反向后输入到A2。这样如果PWM波的占空比高于50%时，电机朝一个方向转；占空比低于50%时，电机朝另一个方向转。通过这种方式，可以在程序中实现反向制动，而这对于赛车在直道上提高速度是有帮助的。

33886 作为一个单片电路H-桥，是理想的功率分流直流马达和双向推力电磁铁控制器。它的集成电路包含内部逻辑控制，电荷泵，门控驱动，及低读选通(on) 金属-氧化物半导体场效应晶体管输出电路。

33886 能够控制连续感应直流负载上升到5.0 安培，输出负载脉宽调制(PWM-ed)的频率可达10 千赫一个故障状态输出可以报告欠压，短路，过热的情况。两路独立输入控制两个半桥的推拉输出电路的输出。两个无效输入使H-桥产生三态输出(呈现高阻抗)。

33886制定的参数范围是 $-40^{\circ}\text{C} \leq T_A \leq 125^{\circ}\text{C}$ 、 $5.0\text{V} \leq V_+ \leq 28\text{V}$ 。集成电路也可以工作在40V通过降低规定的定额值。集成电路能够在表面安装带散热装置的电源组件。

特点

- 与MC33186DH1类似的增强特性
- 5.0 V至40 V连续运转
- 120 mΩ RDS(ON) H桥 MOSFETs
- TTL/CMOS兼容输入
- PWM的频率可达10千赫
- 通过内部常定时关闭对PWM有源电流限制(依靠降低温度的阈值)
- 输出短路保护
- 欠压关闭
- 故障状况报告

33886简化内部结构图:

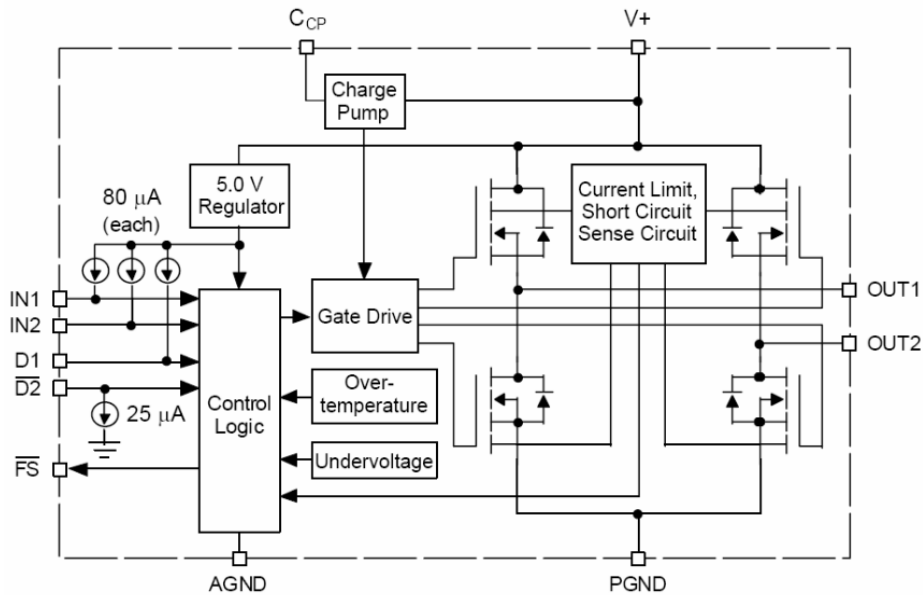


图 4-3 MC33886 芯片内部封装图

芯片的封装：

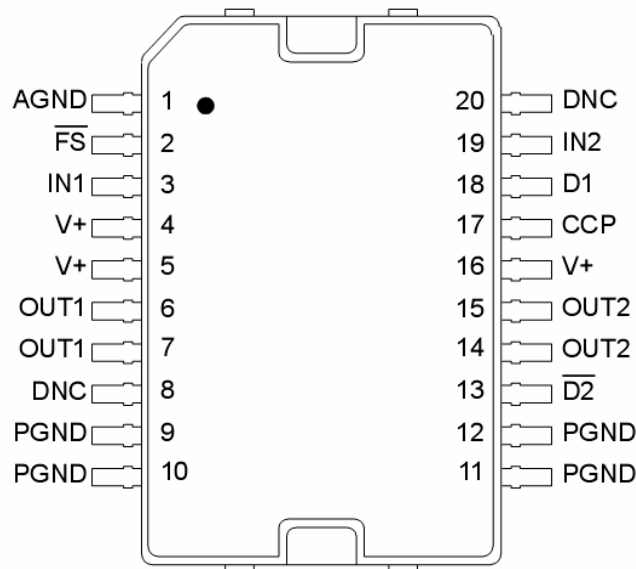


图 4-4 MC33886 芯片外部引脚的封装图

各个引脚的功能：

表 4-1 MC33886 引脚对应功能列表

终端	终端名称	正式名称	定义
1	AGND	模拟接地	低电流模拟信号接地
2	FS	H桥故障状态	故障状态场效应管低电位有效，要求拉电阻上升到5V。
3	IN1	逻辑输入控制1	实际逻辑输入控制的1口
4,5,16	V+	电源供电	正电源连接。
6,7	OUT1	H桥输出1	H桥输出1
8,20	DNC	静止连接	在应用中不连接或接地。他们仅在制造中用于测试模式终端。
9-12	PGND	电源接地	装置电流高功率接地。
13	D2	无效2	输入低电位有效用于使两个H桥输出同时三态无效。当D2为逻辑低,输出都是三态。
14,15	OUT2	H桥输出2	H桥输出2
17	CCP	电荷泵电容器	外部充电电容器连接内部电荷泵电容器。
18	D1	无效1	输入低电位有效用于使两个H桥输出同时三态无效。当D1为逻辑低,输出都是三态。
19	IN2	逻辑输入控制2	实际逻辑输入控制的2口

4.2 核心控制板

核心控制板（如图 4-5 所示）负责视频采集、获取速度并输出舵机和驱动电机的控制信号。电路板上包括了 S12 单片机工作所必需的稳压模块、时钟模块和复位模块，同时还包括了视频信号分离电路、BDM 调试接口、串口以及无线 Zigbee 接口等。



图 4-5 控制电路板

核心控制板的原理图见附表，单片机各个端口的分配情况如表 4-2 所示：

表 4-2 单片机端口分配表

AD 模块	PAD0	摄像头视频信号
	PAD6	电池电压
	PAD7	摄像头电压
外部中断	IRQ	行同步信号

ECT 模块	PT0	速度传感器信号
	PT1	场同步信号
	PT2	奇偶场信号
PWM 模块	PWM0, PWM1	舵机, 由 PWM1 输出
	PWM2, PWM3	33886 (电机)
	PWM4, PWM5	LED4, LED5
串口通讯 模块	RXD0	串口接收
	TXD0	串口发送
无线通讯 模块	RX10	无线接收
	TX10	无线发送
IO	PA0—PA7	LED#2
	PB0—PB7	LED#1
	PJ6	33886 状态检测
	PJ7	33886 使能
	PM0	视频选测信号
	PT5-PT7	数码开关信号

电路板上所有的接口功能说明如下：

- 1) 摄像头视频信号接口J1、J2、J3 三个接口，J2、J3用来连接摄像头，另外一个可以外接电视机，以方便信号调试。
- 2) 视频同步分离模块U2 根据视频输入，分离出场同步信号、奇偶场信号等。
- 3) BDM 调试接口J9 按照清华的定义调整了针脚信号定义，以防止误插反时损坏电路。
- 4) MEMS加速度传感器模块 根据赛车加速度的不同，把某一特定方向的加速度转化为信号输入PAD3口。
- 5) 视频信号处理子模块J8 通过外接硬件积分电路，对视频信号进行处理，用

以识别路线和控制。

- 6) **舵机控制接口J5** 经过测试，在电池电压范围内舵机能可靠工作，所以没有降压而直接用电池电压给舵机供电。
- 7) **串行通讯接口J10** 提供了TTL 电平的串口通讯接口，可以配合多种通讯模块与PC 通讯，通过该接口，在调试时可以方便地获取小车的实时运行参数。
- 8) **无线通讯接口J11** 使用Zigbee无线传输各种数据，方便地获取小车的实时运行参数。
- 9) **速度传感器接口J4** 通过光电编码器产生的脉冲获得赛车的运行速度，该信息可以用于速度闭环控制，保证小车能在不同路段上基本以指定的速度运行，而不受电池电压变化的影响。
- 10) **电池电源接口J12** 连接车模供电电池组。
- 11) **运行电机接口J6** 将两片33886并联使用驱动运行电机，尽量增加驱动电路的电流驱动能力。
- 12) **数码开关S1** 通过左旋、右旋、确认三种方式调节QESA、QESB、QESP，用来现场调整运行速度及控制策略等。
- 13) **LD0稳压** 提供3.3V电压。
- 14) **复位按钮S2** 复位单片机系统。
- 15) **2位数码管DB1、DB2** 通过控制PA0-7，PB0-7用来实时显示系统的工作状态。
- 16) **状态指示灯** 用LED1-LED5显示系统的工作状态。

第五章 视频采集和图象处理

5.1 视频采集

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

5.1.1 摄像头工作原理

摄像头分黑白和彩色两种，根据赛道特点可知，为达到寻线目的，只需提取画面的灰度信息，而不必提取其色彩信息，所以本设计中采用的是黑白摄像头。

摄像头主要由镜头、图像传感芯片和外围电路构成。图像传感芯片是其最重要的部分，但该芯片要配以合适的外围电路才能工作。将芯片和外围电路制作在一块电路板上，称为“单板”。若给单板配上镜头、外壳、引线和接头，就构成了通常所见的摄像头，如聊天用的摄像头；若只给单板配上镜头，这就是“单板摄像头”。单板摄像头日常生活中不多见，生产单板的公司通常将它们卖给其它公司，其它公司再按自己的要求包装这些单板。

单板通常有三个端子：电源端、地端和视频信号端（有的还多出一个端子，那是音频信号端）。电源接的电压要视具体的单板而定，目前一般有两种规格，6-9V 或 9-12V。视频信号的电压一般位于 0.5V-2V 之间。

摄像头的工作原理是：按一定的分辨率，以隔行扫描的方式采集图像上的点，当扫描到某点时，就通过图像传感芯片将该点处图像的灰度转换成与灰度一一对应的电压值，然后将此电压值通过视频信号端输出。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的灰度变化。当扫描完一行，视频信号端

就输出一个低于最低视频信号电压的电平（如 0.3V），并保持一段时间。这样相当于，紧接着每行图像信号之后会有一个电压“凹槽”，此“凹槽”叫做行同步脉冲，它是扫描换行的标志。然后，跳过一行后（因为摄像头是隔行扫描的），开始扫描新的一行，如此下去，直到扫描完该场的视频信号，接着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，下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。

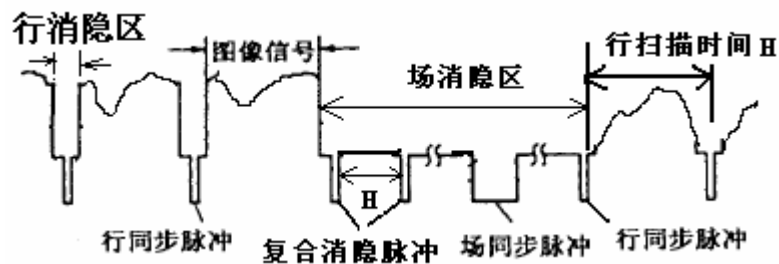


图 5-1 摄像头视频信号

摄像头有两个重要的指标：分辨率和有效像素。分辨率实际上就是每场行同步脉冲数，这是因为行同步脉冲数越多，则对每场图像扫描的行数也越多。事实上，分辨率反映的是摄像头的纵向分辨能力。有效像素常写成两数相乘的形式，如“320x240”，其中前一个数值表示单行视频信号的精细程度，即行分辨能力；后一个数值为分辨率，因而有效像素=行分辨能力×分辨率。

5.1.2 摄像头选择

因为 S12 单片机的 AD 转换时间在不超频的情况下最短为 7 μ s，所以如果选用一个分辨率为 320 线的摄像头，则单行视频信号持续的时间约为 20ms/320=62.5 μ s，AD 对单行视频信号采样的点数将不超过 $[62.5/7]+1=9$ 个。

若使用分辨率为 640 线的摄像头，则单行视频信号持续的时间约为 $20\text{ms}/640=31\mu\text{s}$ ，AD 对单行视频信号采样的点数将不超过 $[31/7]+1=5$ 个。这就是说，分辨率越高，单行视频信号持续的时间就越短，AD 对单行视频信号所能采样的点数就越少。

如前所述，摄像头的分辨率越高，尽管可提高纵向分辨能力，却会减少单片机 AD 采样单行信号的点数，削弱了横向分辨能力。现在市场上摄像头的分辨率通常都在 300 线以上，由此推得单行视频信号的持续时间至多为 $20\text{ms}/300=66\mu\text{s}$ 左右，AD 采样每行视频信号的点数至多为 $[66/7]+1=10$ 个（不超频情况下），这对赛车定位来说是不够的。所以，在选择摄像头时，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能力（但降低后仍然远远够用），却可以增加单片机采样单行视频信号的点数，提高横向分辨能力。

在本次设计过程中，在市场上所能找到的分辨率最低的摄像头是采用 1/3 Omni Vision CMOS 为传感芯片的摄像头，其分辨率为 320 线。另外，为了使得视野大小合适，选用了 3.6mm 的镜头。以下章节中的视频采集工作就是基于此种摄像头的。

5.1.3 摄像头安装

摄像头安装方式如图 5-2 所示：

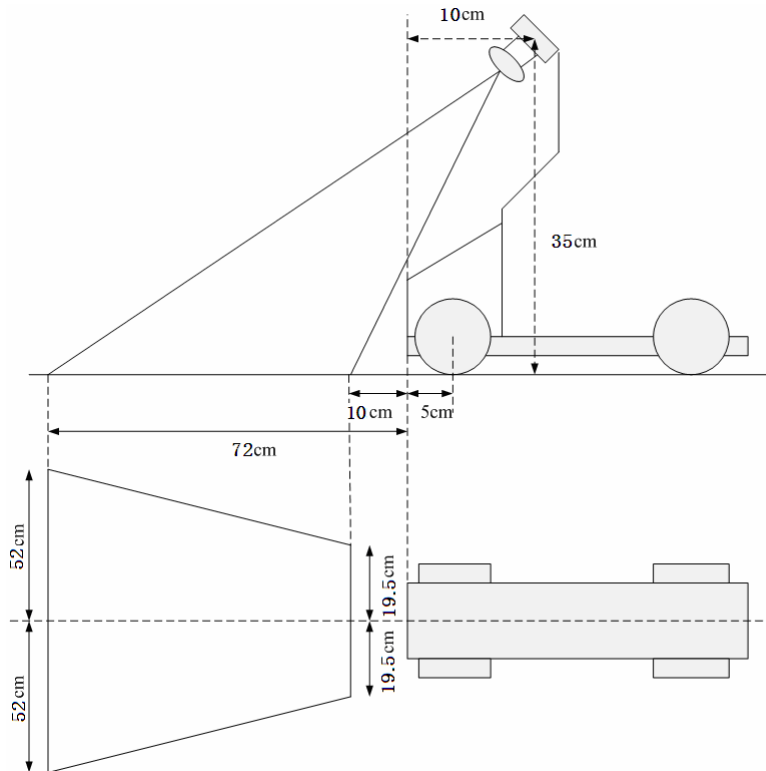


图 5-2 摄像头安装示意图

摄像头的安装位置应合适选取。安装位置太低，会导致视域不够广阔，影响寻线的有效范围；安装位置太高，导引线会变得过窄而无法被检测到，而且赛车系统会因重心抬高而稳定性变差。

安装位置合适的一个标准是：在此位置的拍摄范围能满足控制的需要。控制的策略简单，则所需的拍摄范围就可以小一些；反之策略复杂，需获得的赛道信息较多，则拍摄范围就应大一些。作为本设计的赛车系统，控制策略涉及到赛车入弯时能区分S弯还是普通的单向弯。如图 5-3 所示（摘自韩国比赛用赛道），对于S弯和单向弯，两者前半部分均为一段相同的 30 度的弧（Arc），只不过单向弯的后半部分仍沿原方向偏转，而S弯则沿相反方向偏转。为区别S弯和单向弯，拍摄范围至少要能包含Arc，因此拍摄范围的前后距离至少为 50cm，左右距离至少为 $2 \times 13.4\text{cm}$ （因为弯有可能向左，也有可能向右）=26.8cm。按图 5-3 中的安装参数安装，拍摄范围即可满足要求。

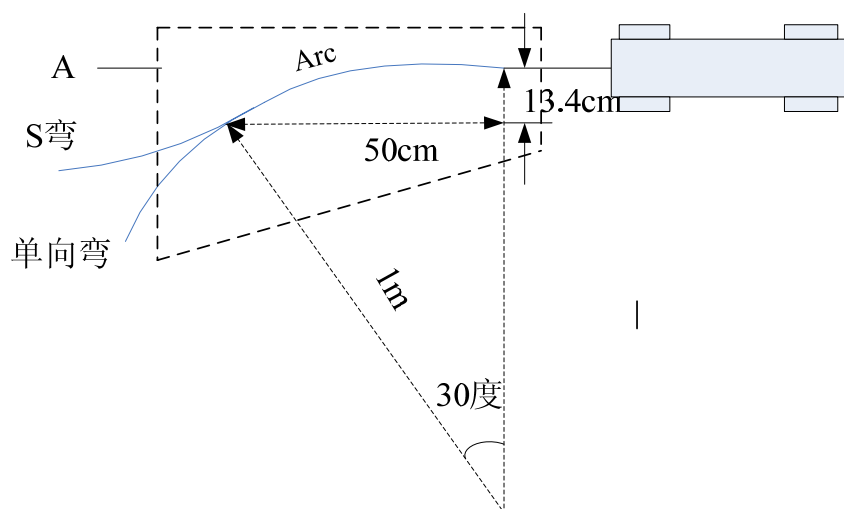


图 5-3 S 弯-单向弯

5.1.4 信号分离电路

要能有效地对视频信号进行采样，首先要处理好的问题是如何提取出摄像头信号中的行同步脉冲、消隐脉冲和场同步脉冲。这里有两种可行的方法。第一，直接通过单片机 AD 进行提取。因为行同步脉冲、消隐脉冲或场同步脉冲信号的电平低于这些脉冲以外摄像头信号的电平，所以据此可设定一个信号电平阈值来判断 AD 采样到的信号是否为上述三类脉冲。第二，就是给单片机配以合适的外围芯片，此芯片要能够提取出摄像头信号的行同步脉冲、消隐脉冲和场同步脉冲以供单片机作控制之用。

考虑到单片机的速度有限，而一些脉冲的间隔时间又较短，为了减轻其处理负担，采用了第二种方法进行信号提取。LM1881 视频同步信号分离芯片（简称 1881）可从摄像头信号中提取信号的时序信息，如行同步脉冲、场同步脉冲和奇、偶场信息等，并将它们转换成 TTL 电平直接输给单片机的 I/O 口作控制信号之用。1881 的端口接线方式如图 5-4 所示。

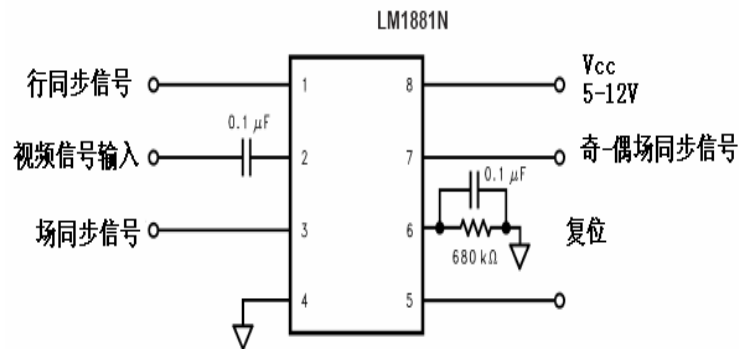


图 5-4 LM1881

其中，引脚 2 为视频信号输入端，引脚 1 为行同步信号输出端（如图 5-5 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 5-5 中的 c）。引脚 7 为奇-偶场同步信号输出端，当摄像头信号处于奇场时，该端为高电平，当处于偶场时，为低电平。事实上，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。

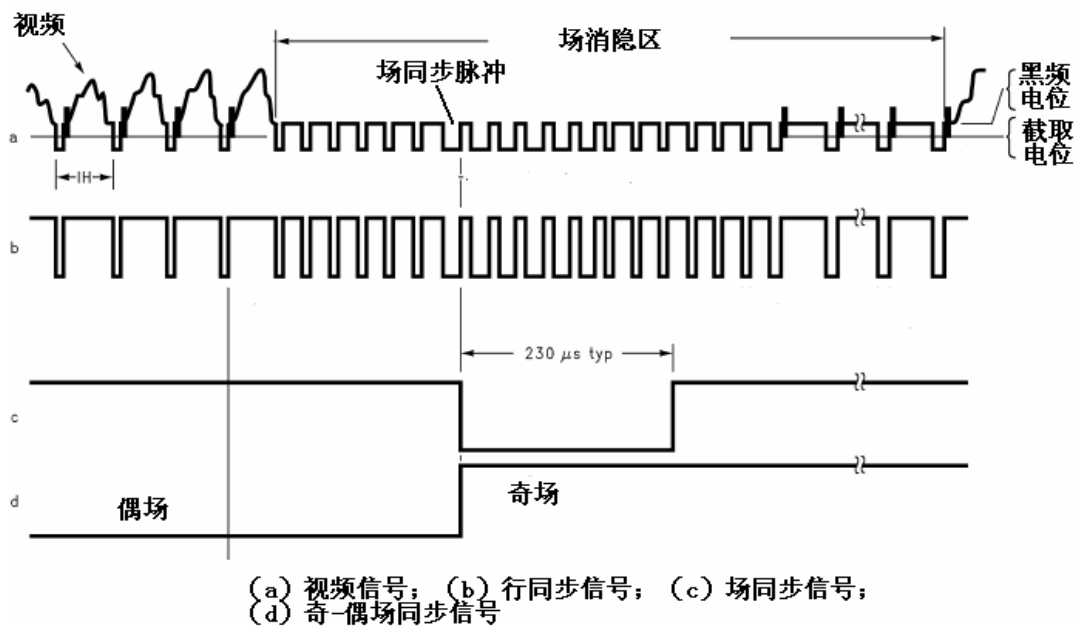


图 5-5 LM1881 信号时序图

由 1881 及其外围电路构成的摄像头采样电路如图 5-6 所示。摄像头视频信号端接 1881 的视频信号输入端，同时也接入 S12 的一路 AD 转换端口(选用 AD0)。1881 的行同步信号端（引脚 1）接入外部中断引脚（IRQ），同时将 LM1881 的场同步信号和奇-偶场同步信号输入到 ECT 模块中（选用 PT1，PT2），这样，既可以采用查询方式获取奇偶场信号跳变，又可以采用脉冲捕捉方式获取电平变化。通过这样的接线，为软件开发提供了多种选择的机会，使程序更加灵活。

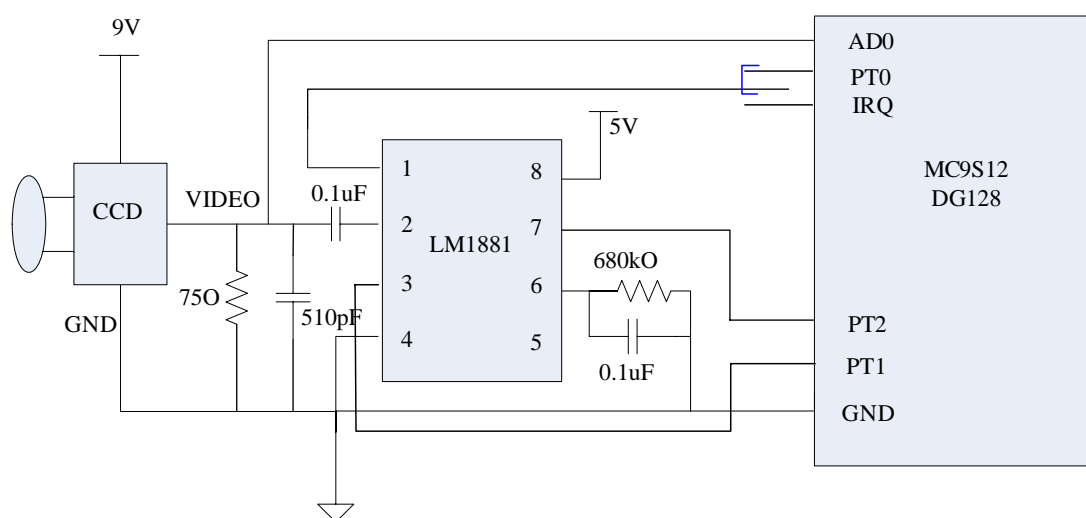


图 5-6 摄像头采样电路图

5.1.5 摄像头初始化

由于摄像头行同步信号接到了 S12 单片机的外部中断 IRQ 引脚，所以当每行视频信号到来时，会产生一个中断。此时如果需要采集该行，就开始进行 AD 转换。摄像头的初始化包括初始化中断和 AD 模块，具体过程为：

```
DDRS_DDRS2=0;           //奇偶场信号输入
INTCR_IRQE=1;           //外部IRQ使能
INTCR_IRQEN=1;          //IRQ选择为边沿触发
ATD0CTL2=0xc0;          //AD上电，快速清零，无等待，关闭外部触发
ATD0CTL3=0x08;          //每通道转换一次，无 FIFO
```

```
ATD0CTL4=0x81;          //8位精度，2个时钟周期，ATD时钟=8MHz  
ATD0CTL5=0xa0;          //右对齐，无符号，单通道，通道0  
  
ATDODIEN=0x00;          //关闭数字输入
```

5.2 图像处理

5.2.1 目标指引线的特征位置

目标指引线是有宽度的（25mm），只要能探测的目标指引线，指引线的宽度信息对智能车定位系统并无额外的帮助。为达到寻线目的，实际上只要提取目标指引线的某些特征点，要求这些特征点合在一起能反映出指引线的形状。称这些特征点的矩阵坐标为特征位置，只要知道目标指引线的特征位置，我们就可以进一步推知目标指引线的形状和位置。提取目标指引线的矩阵坐标，就是指取一些能代表它的特征点，然后求取这些特征点的矩阵坐标。

目标指引线有两类比较重要的特征：中间点和边缘点（如二维数组矩阵中颜色信息为黑色的像素点）。我们可以取每列的中间点、或边缘点作为该列的特征点。二维数组矩阵共 17 列，则共可取出 17 个特征点，每个特征点的横坐标值就是其所在列的列值，而纵坐标值（行值）就是我们将通过算法求出的。若取中间点为特征点，具体做法是，我们采用二值化的方式逐列检测图像数据（详细内容见“二值化算法”一节），判断出每列中颜色信息为黑色的像素点，取每列中这些像素点纵坐标值大小排于最中间的那点（若有两个，则取纵坐标小的那个）为该列的特征点，记录下该特征点的纵坐标值。若取边缘点为特征点，我们逐列检测图像数据以找出每列的边缘点（上边缘或下边缘），记录下该边缘点的纵坐标值（详细内容见“边缘检测方法一”和“跟踪边缘检测方法”两节）。

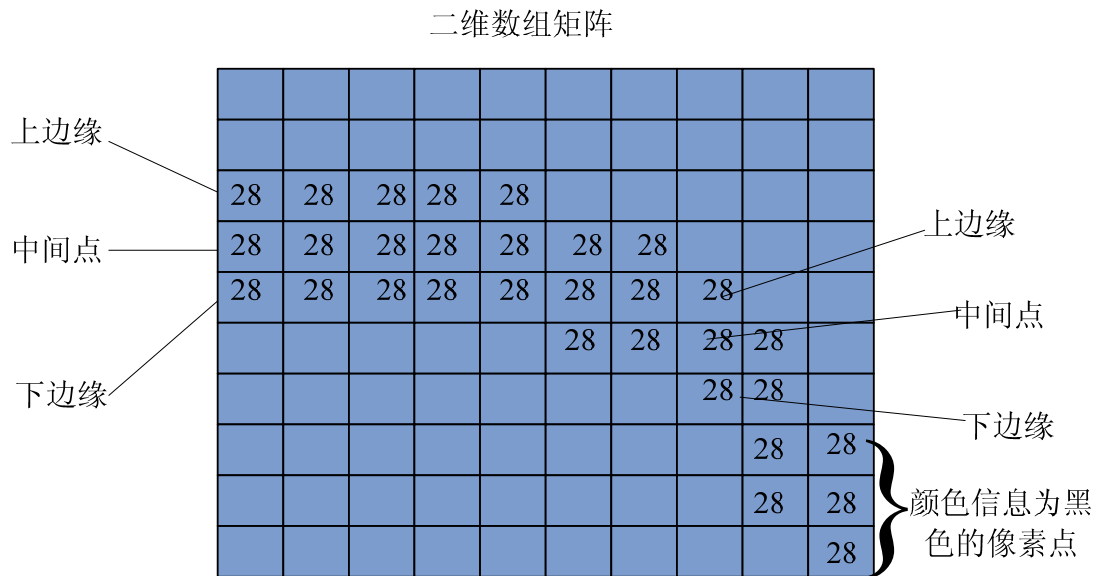
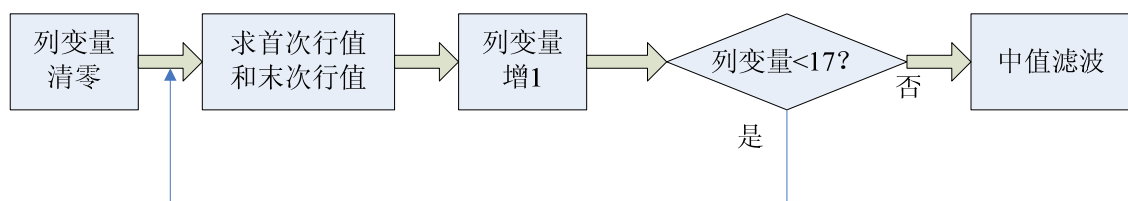


图 5-7 二维数组矩阵

5.2.2 二值化算法

算法思路是：设定一阈值（例如 42），对于二维数组矩阵中每一列，从上至下比较各像素值和阈值的大小。若像素值大于等于阈值，则判定该像素对应的是白色赛道；若小于阈值，则判定对应的是目标指引线。记下第一次出现像素值小于阈值时的像素点纵坐标值（简称首次纵坐标值）和最后一次出现像素值小于阈值时的像素点纵坐标值（简称末次纵坐标值），算出两纵坐标值的平均值（称为中间值），以此平均值作为该列（横坐标）上目标指引线的纵坐标（该列中行值为此纵坐标值的点即为该列的特征点）。

程序流程图如图 5-8 所示。



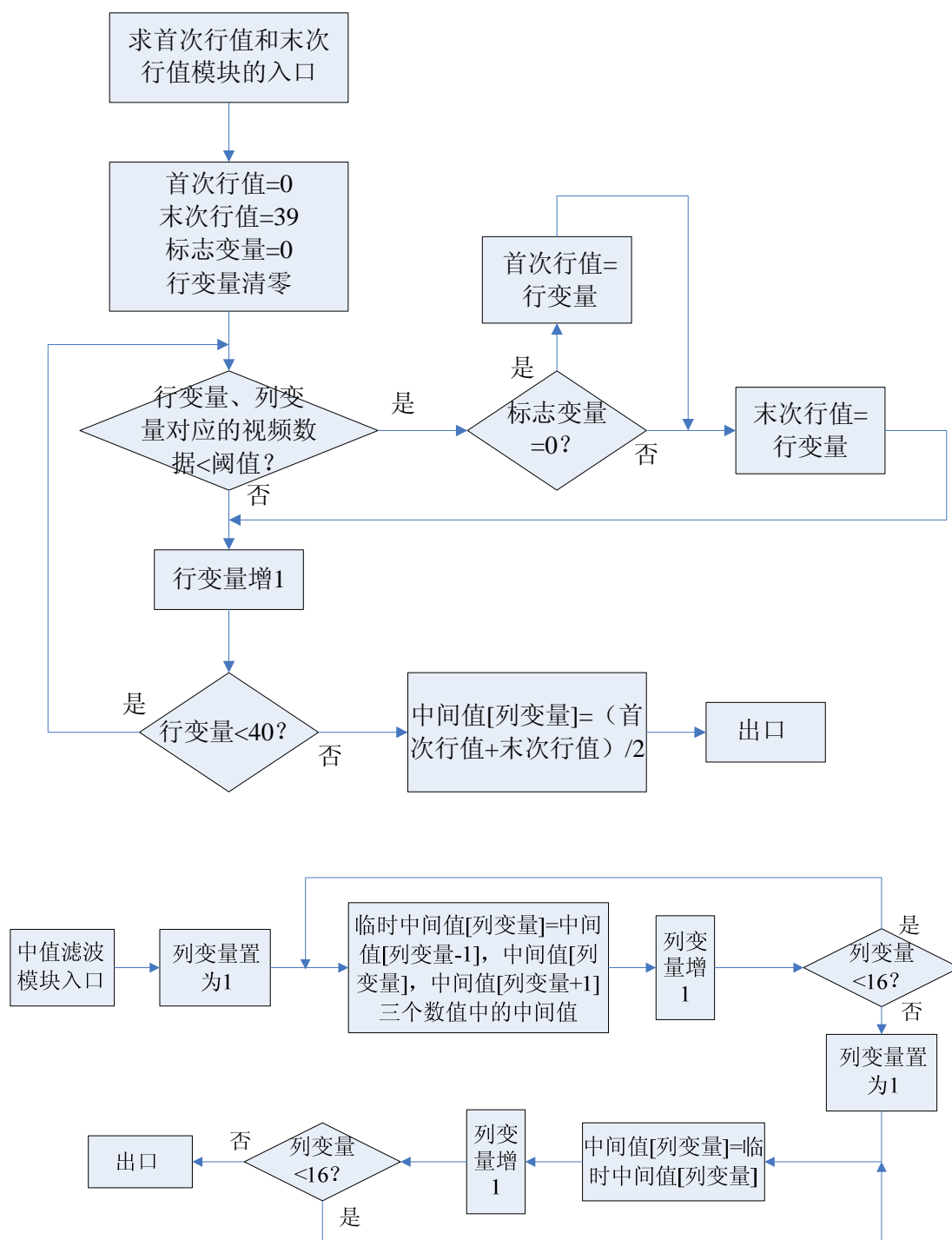


图 5-8 二值化方法

实测该算法的运算时间约为 2580us。当拍摄图像中只有目标指引线一条黑

线时，该算法能准确提取出目标指引线（如图）。该算法的抗干扰性不强，当出现光强有大幅度的变化时，或当拍摄图像中出现其他黑色图迹的干扰时，该算法提取的位置就会有可能与目标指引线的实际位置偏离较大。

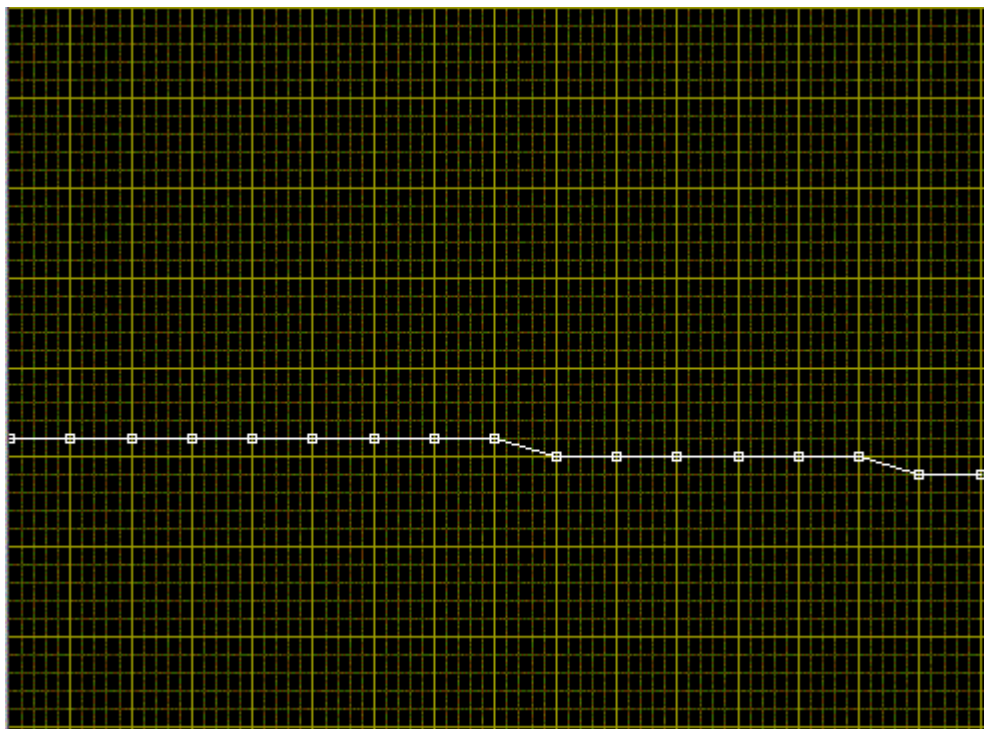
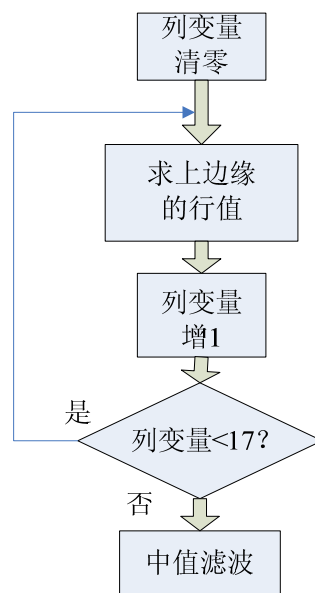


图 5-9 二值化算法效果

5.2.3 边缘检测算法

我们检测目标指引线的上边缘。算法思路是：设定一阈值（例如 15），对于二位数组矩阵中每一列，从上至下求得相邻两像素值间的差值（上减下）。若差值大于等于阈值，则判定其下的像素点对应的是黑色指引线的上边缘，以此像素点作为该列的特征点，记录下此像素点的纵坐标值（即为相应的上边缘纵坐标），作为该列上目标指引线的纵坐标。有可能始终不会出现差值大于等于阈值的情况，则让该列上目标指引线纵坐标值保持不变（即同于分析上一场图像数据时求得的纵坐标）。

程序流程图参见 图。



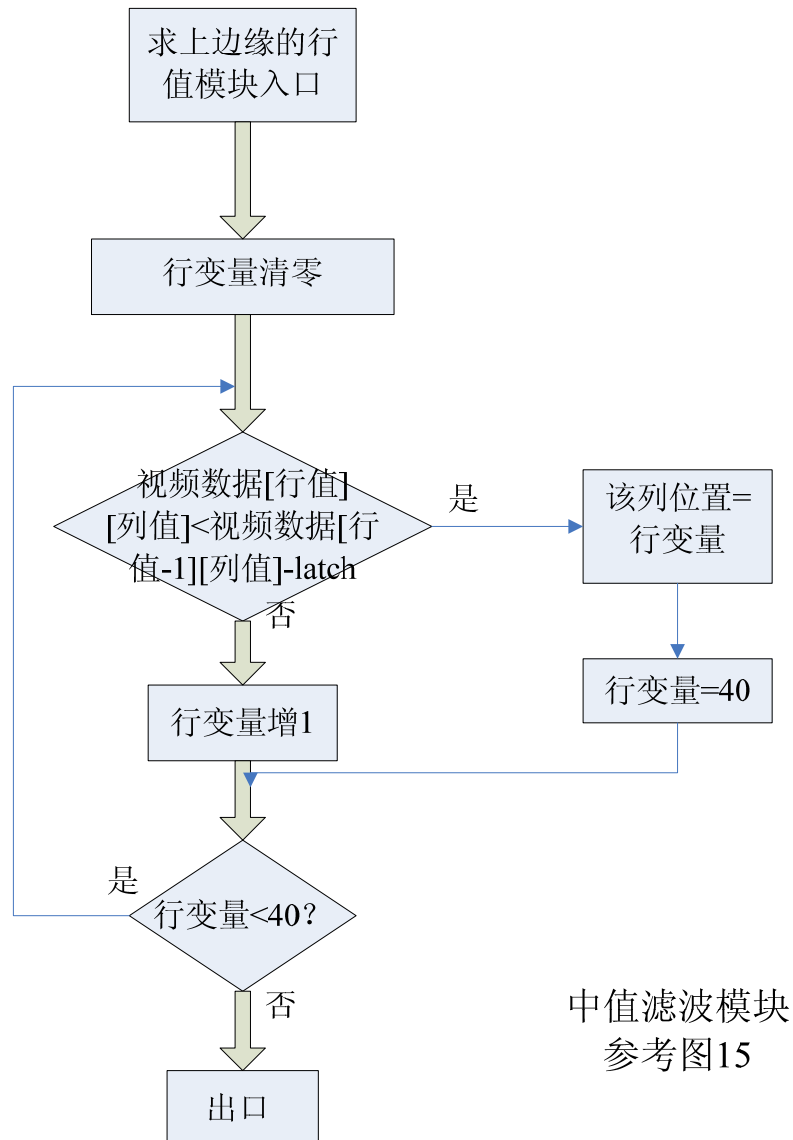


图 5-10 边缘检测方法

实测该算法的运算时间平均约为 1500us。当拍摄图像中只有目标指引线一条黑线时，该算法能准确提取出目标指引线（如图 5-11）。该算法抗环境光强变化干扰的能力较强。该算法能削弱或消除垂直交叉黑色指引线的干扰（如

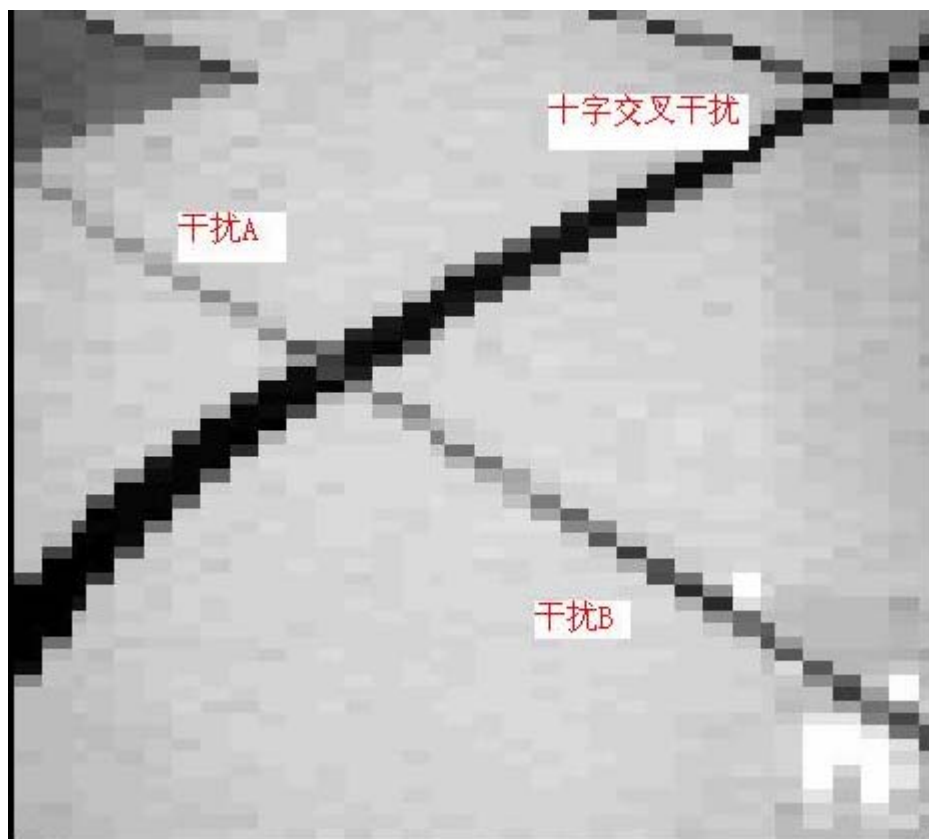
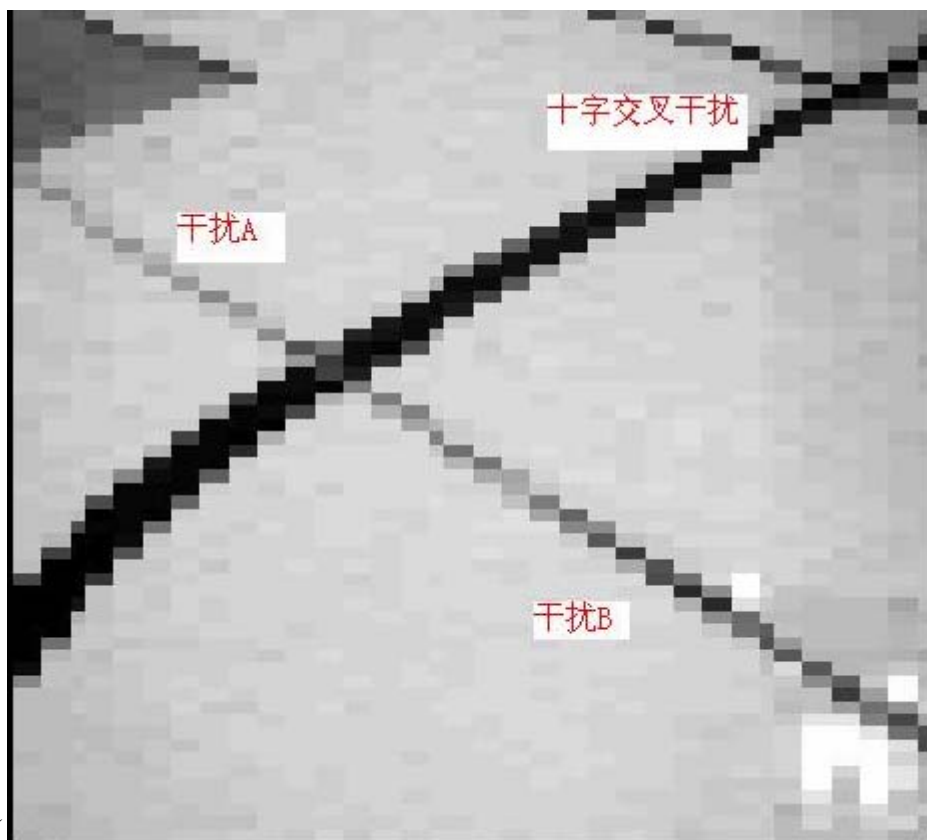
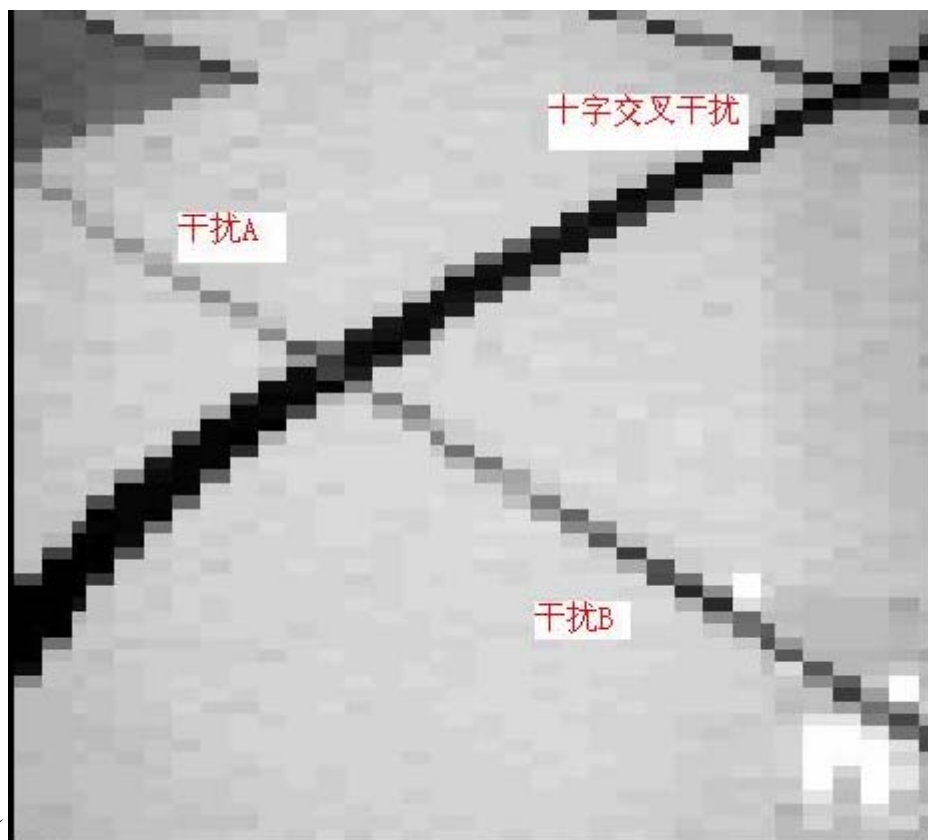


图 5-12)。因为该算法在二维数组矩阵中是由上至下来寻找目标指引线的上边缘，当黑色图迹干扰出现在目标指引线上方



图中的黑色图迹干扰A) 时，该算法无法排除其干扰，而会误将干扰图迹的上边缘当成是目标指引线的上边缘；当出现在下面



图中的黑色图迹干扰 B) 时，可以排除干扰。

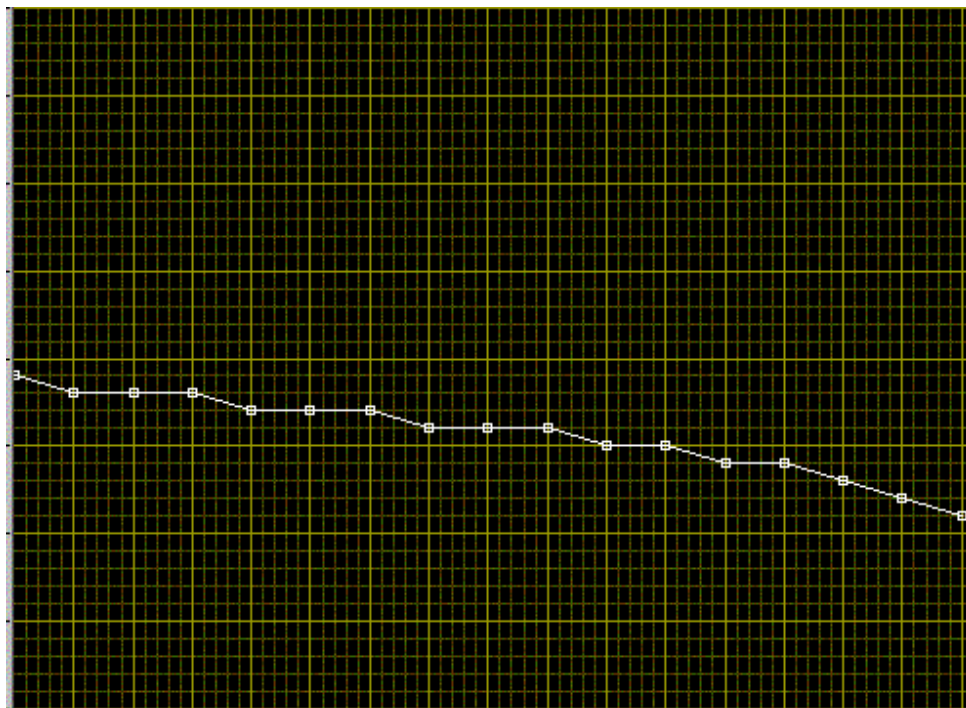


图 5-11 边缘检测方法一效果

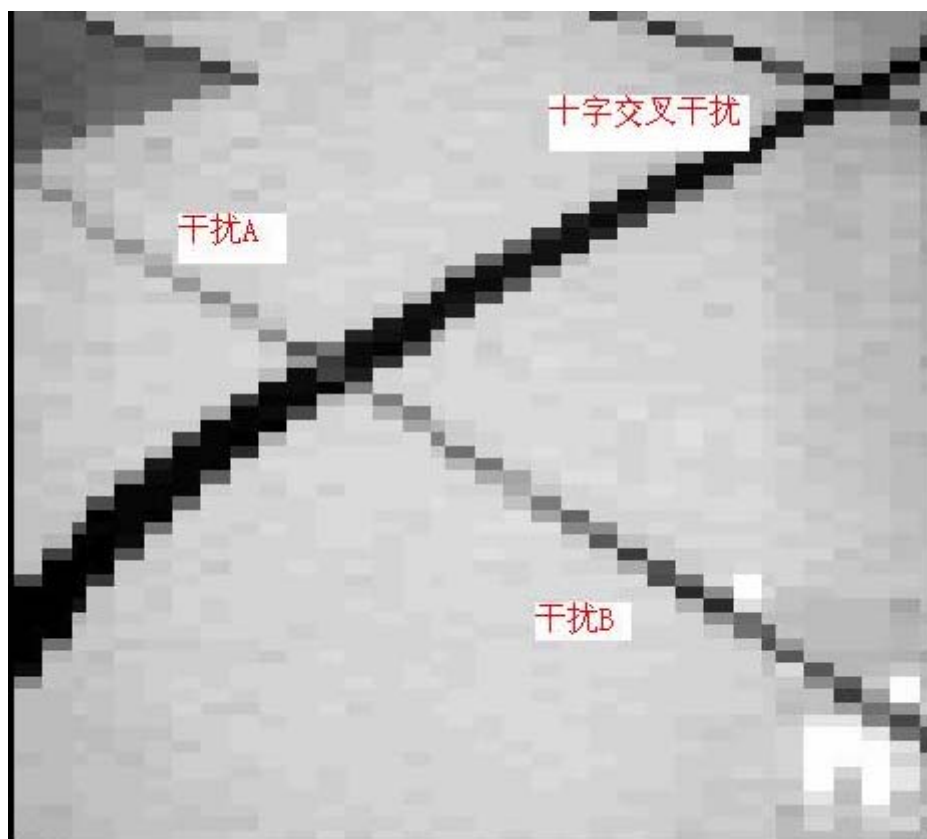


图 5-12 赛道模版

5.2.4 跟踪边缘检测算法

跟踪边缘检测算法跟上一节介绍的边缘方法一样，也是寻找出目标指引线的上边缘，仍然用上边缘的位置代表目标指引线的位置。但跟踪边缘检测从二维数组矩阵每列中寻找上边缘的方法与上一节中介绍的不同。

因为目标指引线是连续的，所以相邻两列的上边缘点比较接近。跟踪边缘检测正是利用了这一特性，其主要思路是：当已寻找出某列的上边缘，若在该位置附近寻找下一列的上边缘，则只用花较少的步骤就可以找到（如图中所示，对比图中的搜寻方式）。

这种方法的特点就是始终跟踪在每列上边缘的附近，去寻找下一列的上边缘，所以就称这种方法为“跟踪”边缘检测算法。

二维数组矩阵

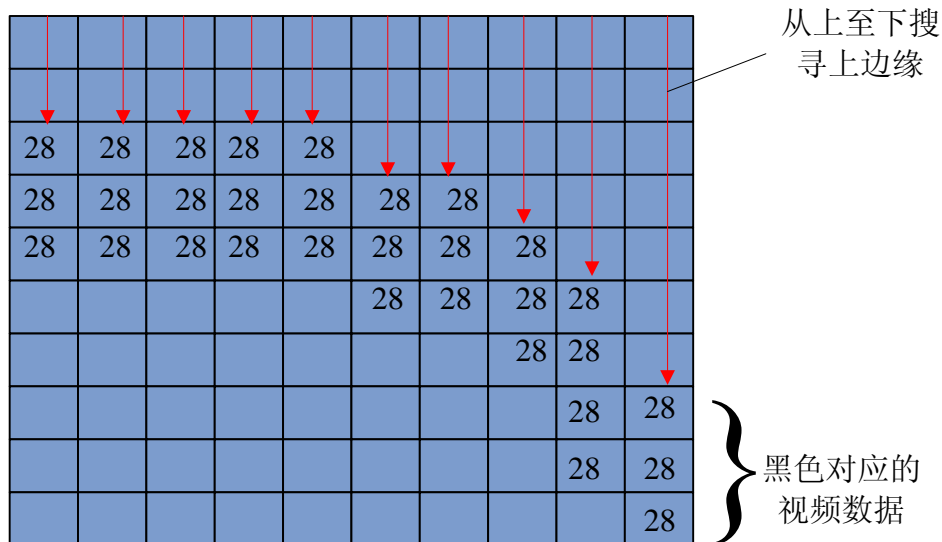


图 5-13 边缘检测方法一搜寻上边缘的方式

二维数组矩阵

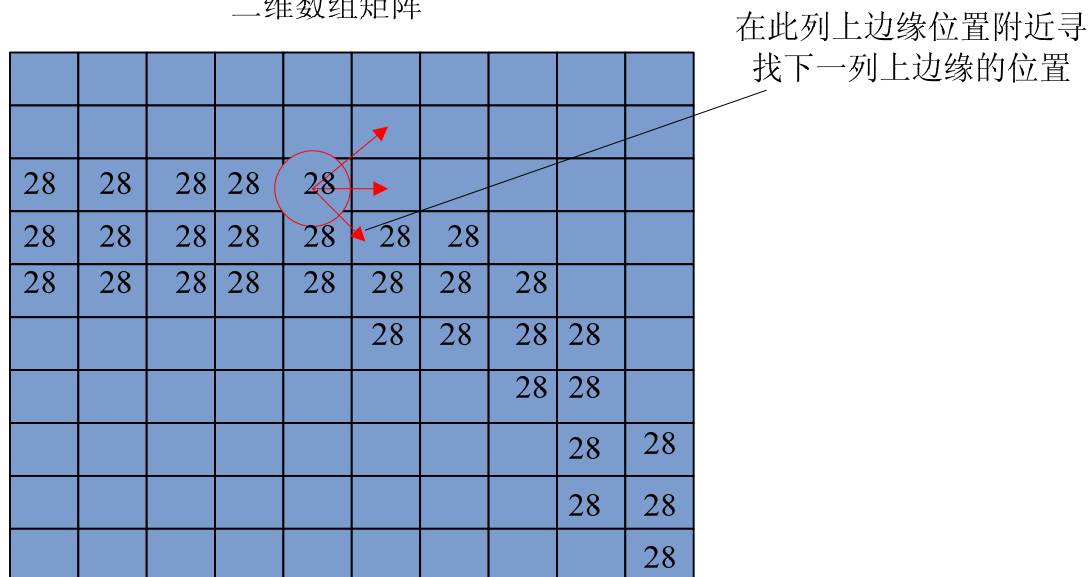
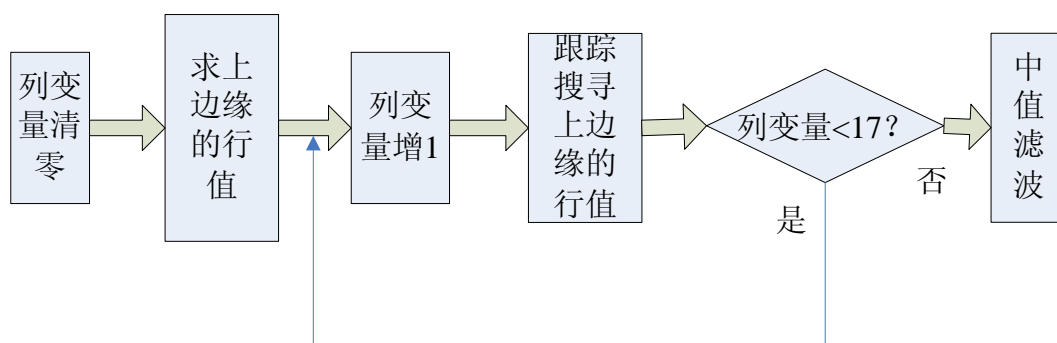
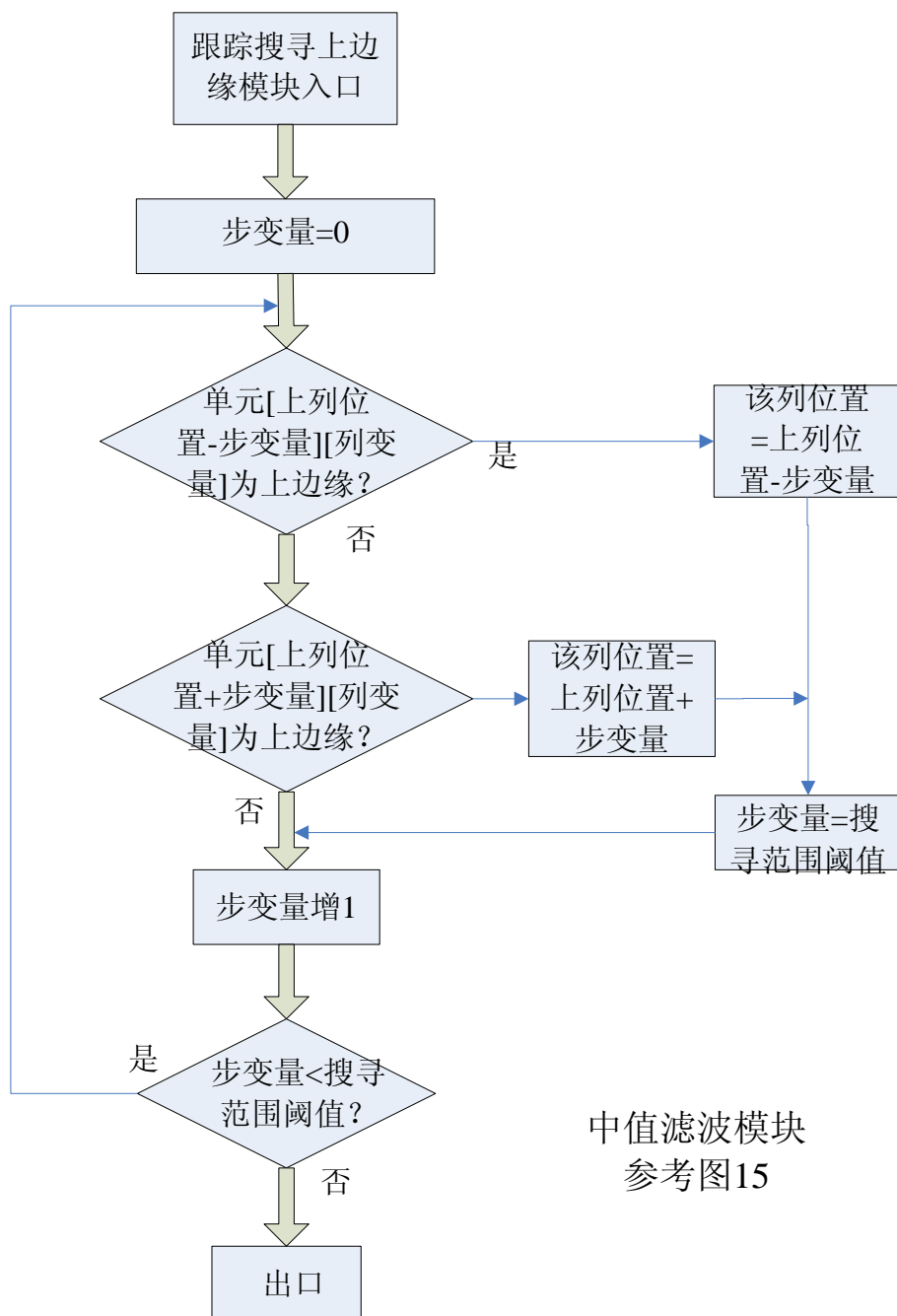


图 5-14 跟踪边缘检测算法搜寻上边缘的方式

该算法的程序流程图如 图所示。

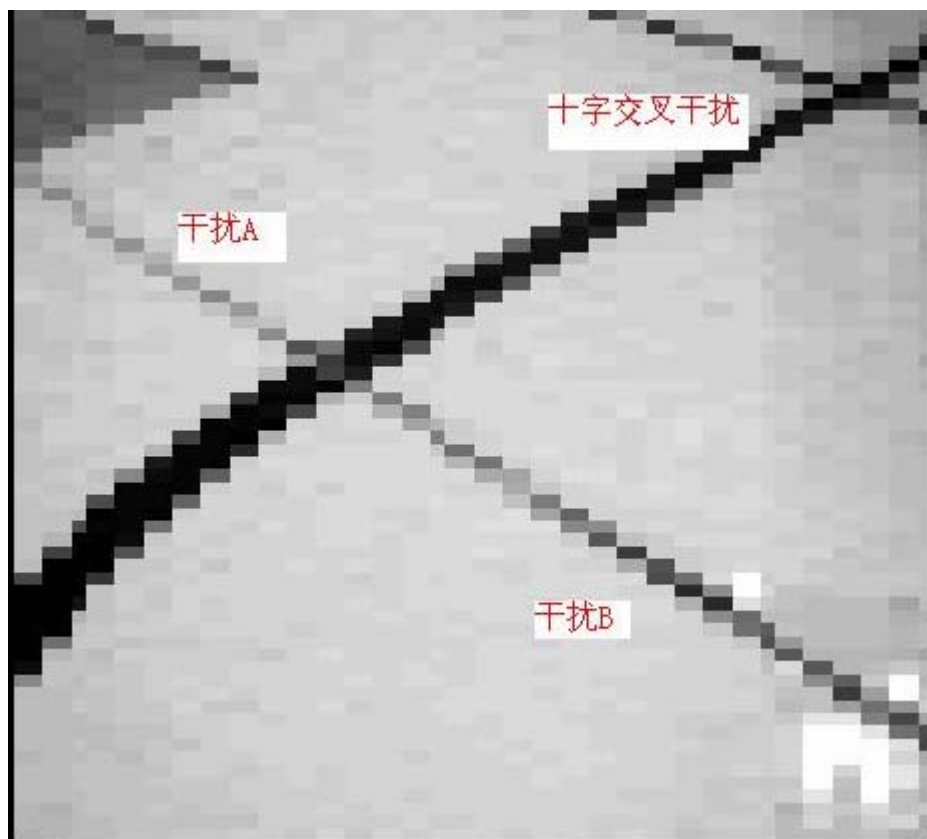




中值滤波模块
参考图15

图 5-15 跟踪边缘检测算法

实测该算法的运算时间平均为 450 μ s。该算法能准确提取出目标指引线，而且具有较强的抗干扰性。该算法同边缘检测方法一样具有较强的抗光强变化的能力，而且该算法能更有效地削弱甚至消除垂直交叉黑色指引线的干扰，该算法还能排除画面内干扰图迹的影响，始终跟踪住目标指引线（如图）。图对应的实际拍摄画面见



图，从图可看出，该算法已经排除了垂直交叉指引线、黑色图迹A、B的干扰，正确提取出目标指引线。

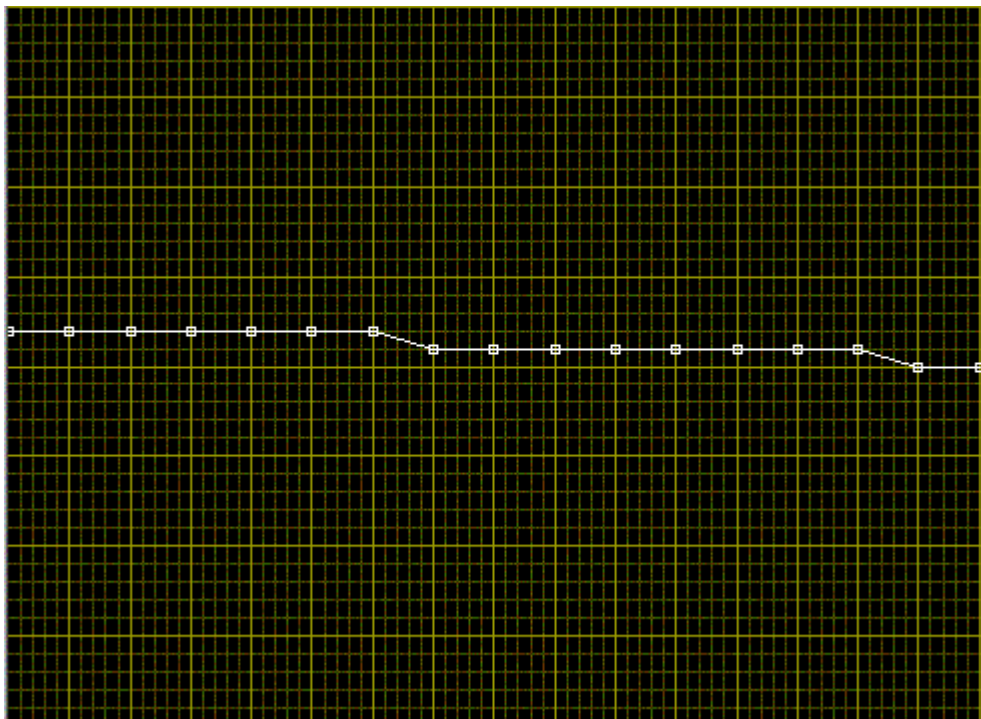


图 5-16 跟踪边缘检测效果

三种目标指引线提取算法的特性总结如表 5-1 所示。

表 5-1 三种目标指引线提取算法的特性比较

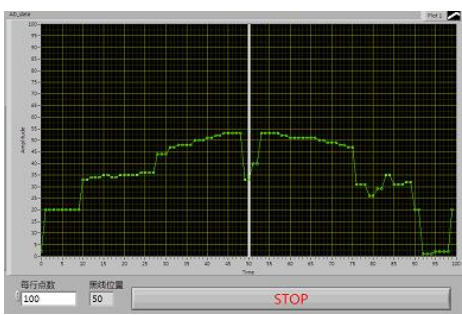
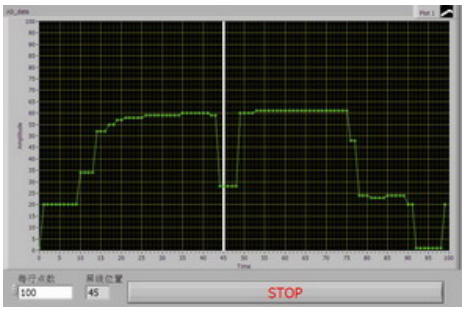



	运算时间	普通情况下*的位置提取效果	抗干扰性
二值化算法	约 2580us	准确	抗光强变化和黑色图迹干扰的能力都较弱
边缘检测方法一	平均约为 1500us	准确	抗光强变化的能力较强, 抗黑色图迹干扰的能力较弱。能削弱或排除垂直交叉指引线的干扰
跟踪边缘检测方法	平均约为 450us	准确	抗光强变化和黑色图迹干扰的能力都较强, 能削弱或排除垂直交叉指引线的干扰

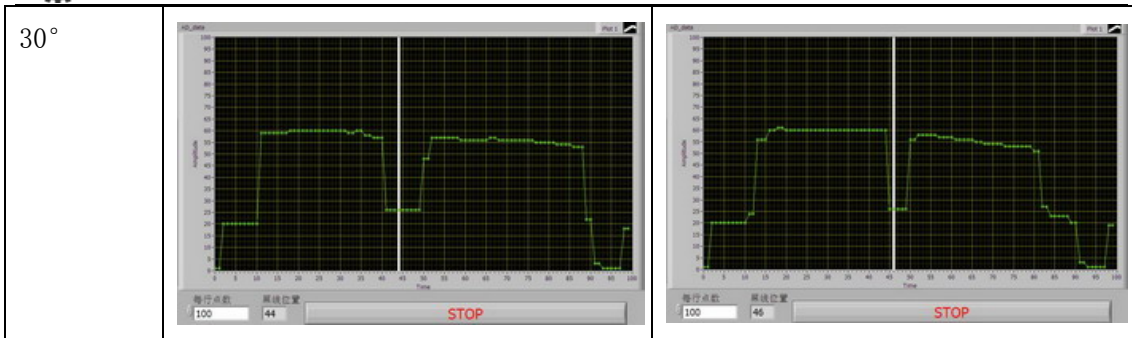
* 普通情况即指，环境光强为普通室内日光灯照射下的光强，而且拍摄画面内只有目标指引线一条黑色指引线。

5.2.5 用摄像头识别坡道

由于摄像头采集同样符合近大远小的原则，所以想通过判断坡道引起的距离变化来识别出不同角度。对不同距离的黑线的不同角度进行识别，观察摄像头采集到的黑线宽度变化。如下表所示：

表 5-2 摄像头测试

角度\距离	48 厘米	80 厘米
-15°	无	
0°		
15°		



结论:

同一距离不同角度黑线宽度有一定区别,可以在一定程度上判断坡道和直道,只是现阶段采集精度下区别不是特别明显,不太容易准确判别,所以需要提高采样精度。

5.2.6 双摄像头的研究

由于今年新的电路板有 2 个视频输入口,所以尝试采用双摄像头来丰富视频采集的多样性。电路板上的 J2, J3 是视频输入端口,默认情况下是选择 J2。J1 可作为信号输出端,在电视机上实时观察摄像头的情况。

使用 PM0 进行摄像头的选择,选择过程如下。

```
DDRM_DDRM0=1;    // PM0 口方向
PTM_PTM0          // 视频信号选择
PTM_PTM0=1;       // J3 口信号
PTM_PTM0=0;       // J2 口信号
```

摄像头切换需要 10ms,纯 1 隔 1 切换每秒可以切换 100 次;1 隔 1 切换等待一场完整信号(采到 1 个场信号到下一个场信号来临)可以采集 33 场每秒。

$$1s/100=10ms \quad 1s/(10+20)=33ms$$

等于 30ms 一个周期时,可以稳定交替采集 2 个信号。

理论上计算也是如此,从一个信号切换到另一信号时,处于一场信号中的

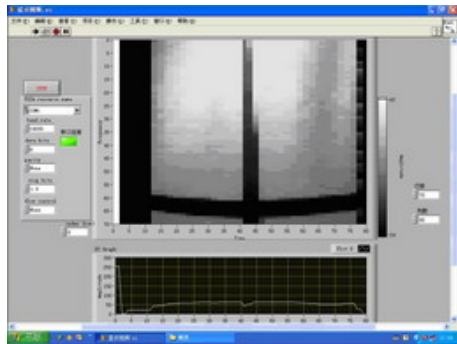
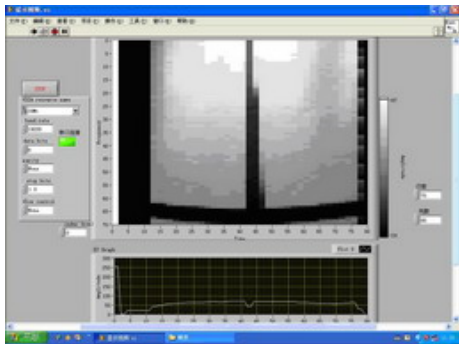
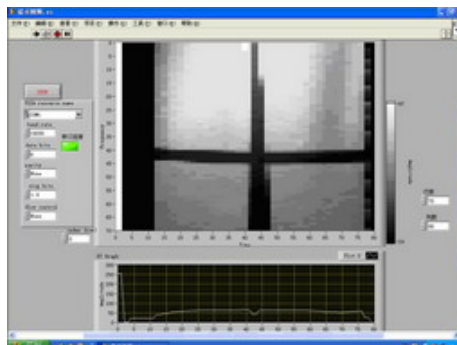
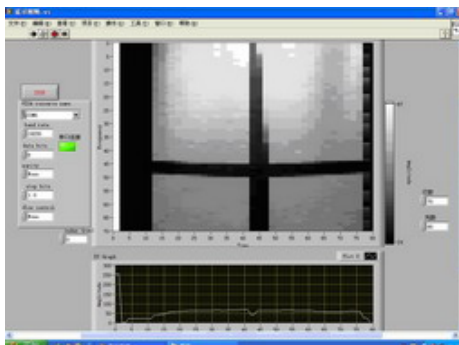
位置是不一定的, 如果刚巧一场开始, 则等待 20ms, 刚巧一场结束, 则等待 0ms, 按统计平均正好是 10ms.

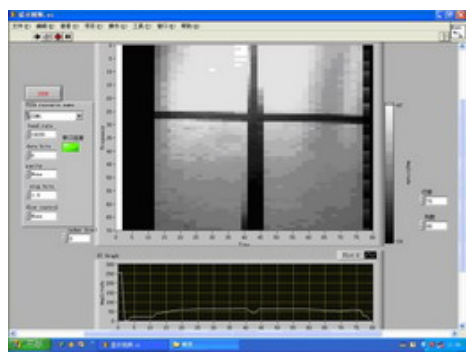
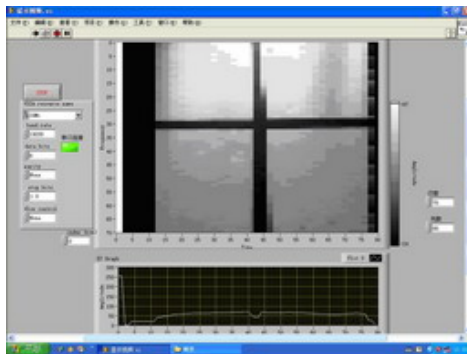
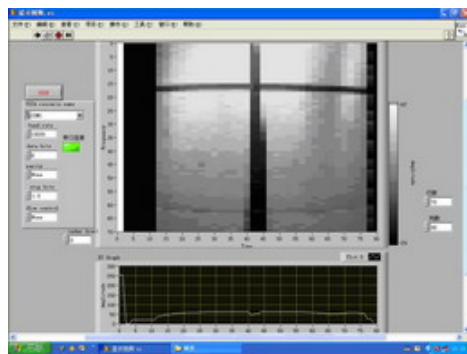
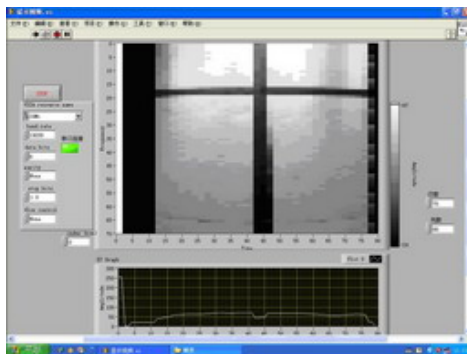
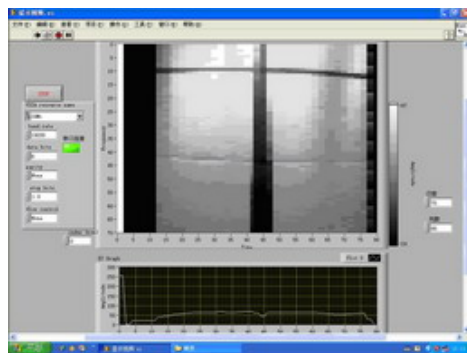
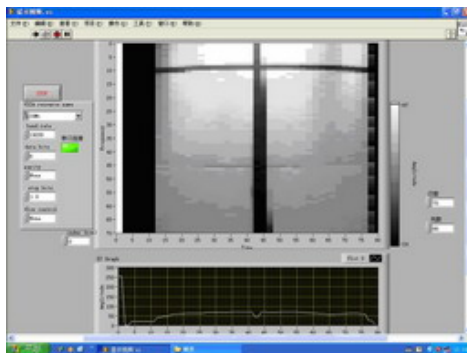
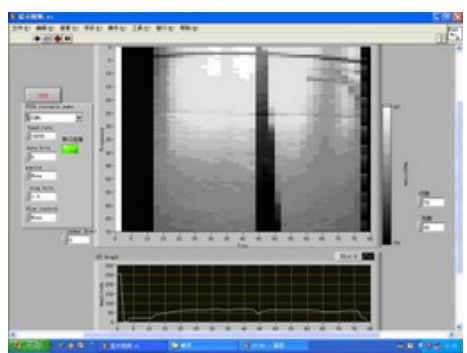
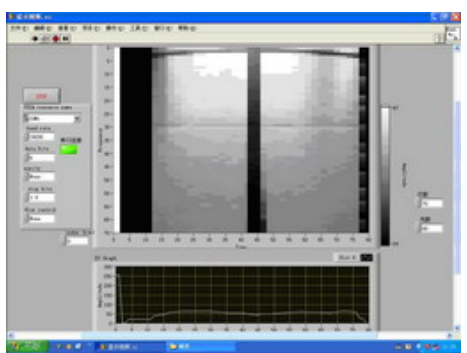
5.2.7 摄像头镜头畸变

通过去年和今年两种安装高度的对比, 可以看出对图像采集的影响。

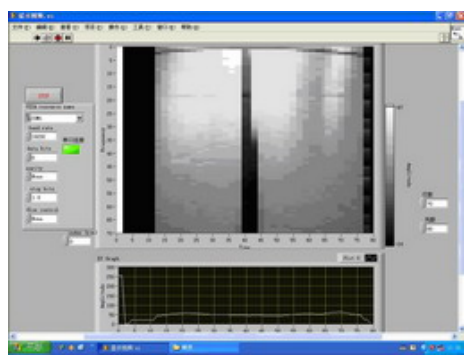
2 车摄像头的安装高度分别是 27cm, 37cm; 使最近可视距离均为 13cm 处, 则最远处分别可达 75cm 和 67cm。

表 5-3 摄像头畸变测试

距离\高度	27CM	37CM
15 CM		
25 CM		

35 CM		
45 CM		
55 CM		
65 CM		

75 CM



超出范围

摄像头在视频采集时，边缘会产生一定的畸变，从图上可以看出，在图像的最上方和最下方直线会有一定的弯曲，在 35cm 处图像畸变程度最小。通过不同高度的对比，可以看出安装高度对畸变的影响远小于距离对畸变的影响，2 种安装高度在相同距离的畸变基本一致，都是 2 头弯，中间直。所以在识别起始线和交叉线时尽量使用中间采集，当然可以适当移近（因为近处的线较宽）。摄像头的高度对采集区域的宽度有一定影响，安装高度越高采集区域越宽，适当调高可以增大视野范围。

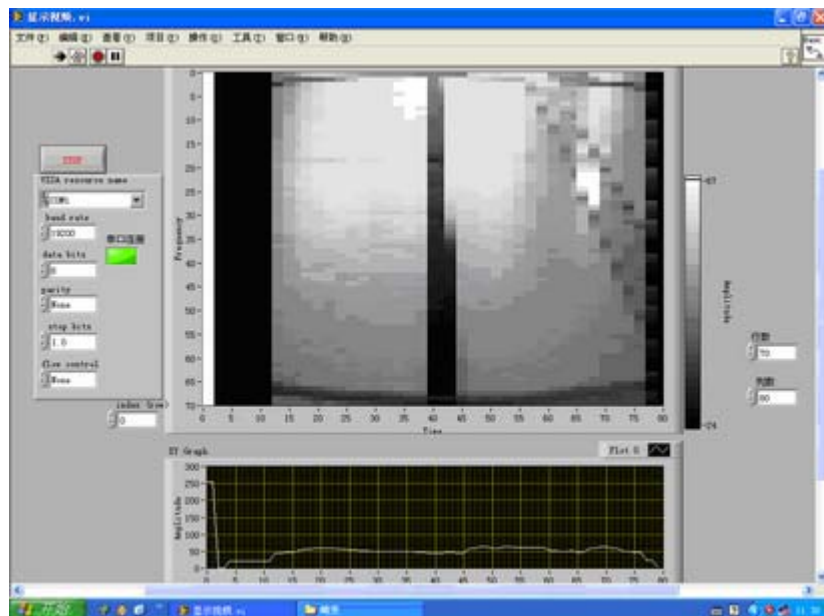


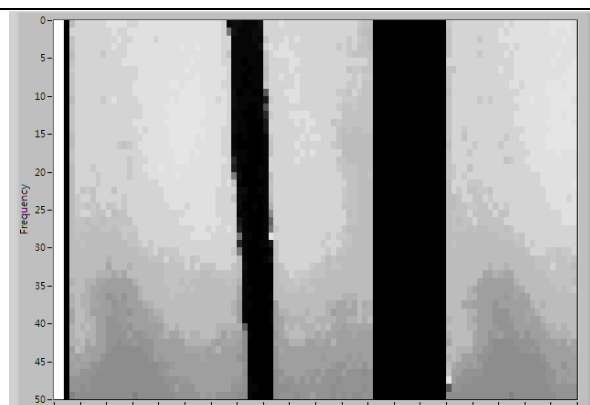
图 5-17

对于沿黑线方向，从图 5-17 可以看出，是近处可视范围窄，远处可视范围宽(上图黑线和旁边的阴影实际上和黑线是平行的)，所以直线不在中间位置时，分析计算时应当也要注意区别。

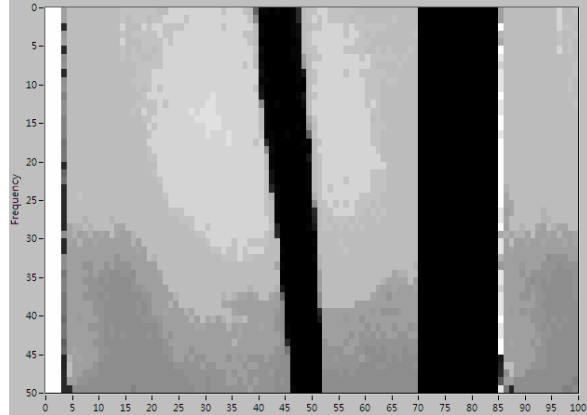
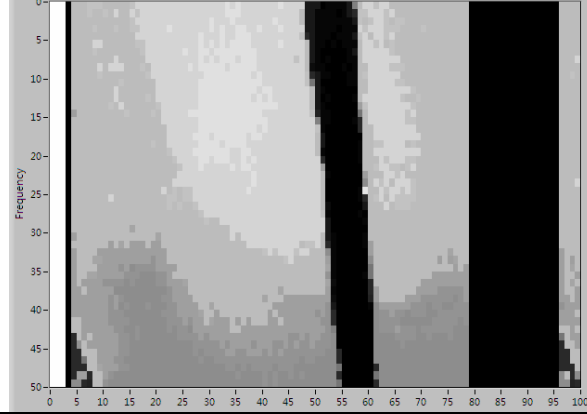
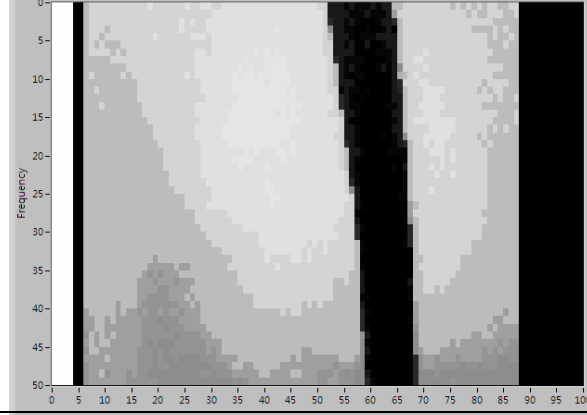
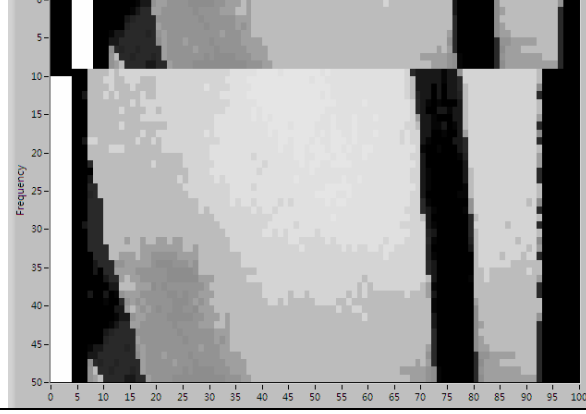
5. 2. 8 超频试验

由于今年核心电路板和驱动电路板分离，加上板子散热情况较好，所以尝试对其进行超频试验。结果如下：

表 5-4 超频测试

单片机频率	图像	每行采集到的点数
32 MHz		62 个点



36 MHz		70 个点
40 MHz		78 个点
44 MHz		87 个点
46 MHz		92 个点

48 MHz	没有接收到图像	
--------	---------	--

结论：44MHz 可以稳定采集，46MHz 有图像，但有一定错位，48MHz 没有信号。

此外，AD不能单独进行超频， $ATDclock = BusClock \times (PRS + 1) \times 0.5$ ，但改动ATDCTL4没有反应，即PRS置0和1采集到的点数是一样的。

第六章 车体控制

6.1 速度控制

6.1.1 速度传感器模块

为了使得赛车能够平稳地沿着赛道运行，需要控制车速，使赛车在急转弯时速度不至过快而冲出赛道。通过控制驱动电机上的平均电压可以控制车速，但是如果开环控制电机转速，会受很多因素影响，例如电池电压、电机传动摩擦力、道路摩擦力和前轮转向角度等。这些因素会造成赛车运行不稳定。通过速度检测，对车模速度进行闭环反馈控制，即可消除上述各种因素的影响，使得车模运行得更稳定。此外，在记忆算法中为了记录道路信息，需要得到赛车运行距离，这也可以通过车速检测来实现。

车速检测的方式有很多种，例如用测速发电机、转角编码盘、反射式光电

检测、透射式光电检测和霍尔传感器检测。经过对去年测速方案和其它学校方案的比较，本次设计中速度传感器采用的是OMRON公司生产的E6A2-CS100型光电编码器。它由5-12V的直流供电，安装方式如图6-1所示。速度传感器用螺钉固定在塑料片上，塑料片固定在后轮支架上，这样固定好之后，就有了较高的稳定性。速度传感器通过后轮轴上的齿轮与电机相连，车轮每转一圈，速度传感器转过2.75圈。



图 6-1 速度传感器安装位置

速度传感器有三根引线，其中棕色线接 VCC（即图 6-2 中的引脚 1），蓝色线接地（引脚 3），黑色线为输出信号，需要上拉后接入单片机（引脚 2）。车轮每前进一段距离时，速度传感器便产生一定数目的脉冲，单片机利用 ECT 对脉冲进行计数，从而得到速度值。



图 6-2 速度传感器接线

6.1.2 速度传感器初始化

通过对每个控制周期内速度传感器产生的脉冲数进行累加，可以测出这一周期内的平均速度。由于目前速度传感器的精度较高，所以用了 16 位的脉冲累加器 PB 进行脉冲计数。其初始化过程为：

```
TIOS=0x00;           //所有通道设置为输入捕捉
TSCR1_TEN=1;         //主计时器使能
TCTL4=0x03;
PBCTL_PBen=1;        //16 位脉冲累加器 B 使能
```

6.1.3 速度获取

在每一控制周期开始，S12 读取脉冲累加器中的数值，然后与第m个控制周期前的脉冲累加器值作差，这样就求得了之前m个控制周期时间里共有多少个脉冲数累积，从而由可求得赛车速度值。程序流程如图 6-3。

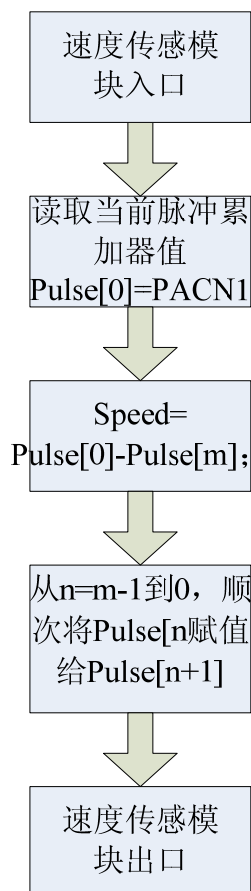


图 6-3 速度获取

6.1.4 速度传感器准确性测试

让赛车在赛道上行驶，每 20ms 就将赛车当前速度值通过串口发送到电脑上一次。对赛车行驶的正好一圈中的速度值进行累加求和，再乘以 20ms，得到的总共行驶距离为 27m，而赛道总长为 26m。两者的相对误差不到 4%。这说明，速度传感器测量基本准确。

6.2 舵机控制

按照比赛规则，赛车应沿赛道行驶，不得触及两侧的标志杆，所以必须根据导引线和赛车的相对位置，及时对舵机进行调整。通常赛车行驶时间最短的路径（最佳行驶路线）与导引线位置接近，因此较为简单的行驶策略是使赛车

基本沿着导引线走。但是，赛道上存在一种特殊情况，如图 6-4 所示，导引线相对于一条中心线左右摆动呈“S 形”。对于这种道路，赛车最佳的行驶路线应该是其中心线方向。以下分别介绍简单沿线行驶策略和最佳行驶路线的舵机控制策略。

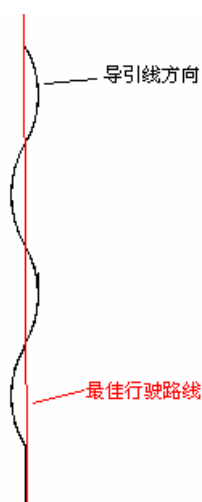


图 6-4 S 道

6.2.1 简单沿线行驶策略

简单沿线行驶策略的目标是控制舵机使得赛车尽可能沿着导引线前进。如果采用光电管寻线方案，由于探测距离有限，即使赛车检测到导引线进入弯道时，立即开始按照距离偏差的正比执行转向，开始时角度偏转也将很小。若采用 PID 控制的 D 分量，由于微分量反映了路径曲率的变化，增加了这一项，可以提前对舵机进行转向控制。

在 2005 年韩国比赛赛道上用光电管方案、PID 对舵机转角进行控制，发现添加 I 参数后系统变得非常不稳定，所以这里只用了 PD 控制。改变 P、D 值，记录赛车运行时间和实际行驶路线与导引线偏差的程度，得到下表结果：

表 6-1 不同 PD 参数下赛车运行情况

P 值和 D 值	赛车跑完全程的时间	路径偏差程度
----------	-----------	--------

KP=6; KD=0;	22s	64.81925
KP=8; KD=0;	20s	50.21506
KP=10; KD=0;	21s	50.63976
KP=8; KD=1;	21s	55.6323
KP=8; KD=3;	21s	53.64936
KP=10; KD=3;	21s	51.1258
KP=6; KD=5;	23s	295. 3797
KP=8; KD=5;	22s	52.66921

从表中可以发现，当路径偏差小的时候赛车跑完全程的时间就短，当方差等于 50.2 时，赛车的速度最快。因为路径偏差是赛车调整次数和调整程度的反映，当赛车调整次数少并且调整量小的时候，赛车就减少了走无效路程的时间，总的时间当然就小了。

为了说明 PID 各个参数对系统的影响，可以分析赛车在第一个急弯上冲出赛道的时间，这个时间可以认为是黑线丢失的持续时间。该时间表示的是赛车的动态响应特性。但仅凭这一时间无法知道赛车的超调程度，因为冲出赛道后检测到的黑线位置都是在最边上。因此，只能近似分析出赛车的动态响应特性。根据 PID 理论，P 系数能影响小车的动态响应特性。随着 P 值的增大，赛车的动态响应越来越快。从表 6-2 中可以看到，随着 P 从 60 变到 100，赛车在赛道外的时间逐渐变小，这说明赛车的动态响应变快了，因为赛车一检测到冲出赛道马上做出了调整。

表 6-2 不同参数下赛车冲出第一个弯道的时间

p=60	p=80	p=100	p=80 d=10	p=80, d=30	p=100 d=30	p=60 , d=50	p=80, d=50
1.8s	1.2s	1s	1.2s	1.1s	0.8s	0.6s	1s

从理论上分析，当加入 D 分量后，赛车的动态特性基本不变，而超调变小。验证时可以假定赛车的动态响应基本不变，那么参照上表当 D 分量变大时小车

冲出赛道的持续时间变小，就可以看到是 D 分量使得超调变小的结果。因为在动态响应不变的情况下，只有超调变小，才能使得冲出赛道的持续时间变小。从这个意义上讲，当时使用光电管方案时用 PD 方法对赛车进行控制是可行的。

然而如果采用摄像头对导引线进行识别，就可以在较早的时候检测到弯道，这样只要 P 参数足够大，便可保证赛车有足够的转向，本设计中也就只用了 P 控制。但是由于赛车在不同路段上的速度不同，为了保证系统稳定性，赛车的预瞄距离和对每行黑线偏差进行校正的 P 系数是动态选取的。

6.2.2 最佳行驶路线策略

前面已经提到，对于 S 道最佳的行驶路线是沿着中心线行驶，这样可以大大提高赛车速度，缩短行驶时间。如何能做到这一点呢？这要建立在两个基础之上：其一，需将 S 弯和普通弯道、直道区分开来；其二，要有合适的控制策略使舵机在 S 道上以较小幅度摆动，甚至直接冲过去。

识别 S 道的方法基本上可以分为两种：一种直接通过摄像头采集到的信号进行判别；一种通过赛道记忆信息进行判别。这里先讨论第一种方法，赛道记忆在后面将述及。 S 道和直道一样，它们的黑线位置都在视野中间的某一区域内，而普通弯道会从视野的一侧拐出去，从这一基本事实出发就可以识别出 S 道。当然，也可以采用其它算法进行识别，例如对导引线的趋势进行分析，如果是单独超一个方向延伸则认为是普通弯道或者是直道，反之则视为 S 道。

区分出 S 道后，下面就要对舵机施以适当的控制策略，使其尽可能地冲过 S 道。由于舵机目前采用的是 P 控制，如果在 S 道上将 P 系数取得很小，同时预瞄的距离也取得远一些，那么舵机就能降低对 S 道上导引线的敏感程度，以较小的摆动幅度冲过 S 道。如图 6-5 所示，逐渐减小 P 参数后，可以看出赛车在 S 道上的行驶效果。图中黑色曲线为 S 形导引线，路线 1、路线 2 和最佳路线的 P 参数依次递减。

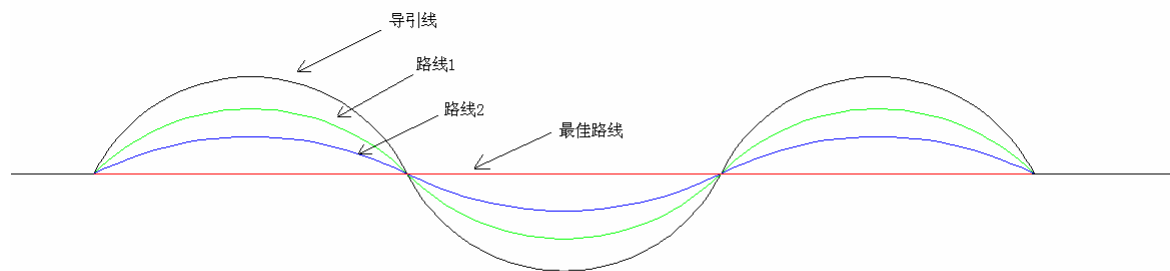


图 6-5 不同 P 参数下 S 道行驶效果

6.3 加速度控制

这次加速度传感器使用的是 freescale 公司出品的 MMA2260。

6.3.1 工作方式：

- MMA2260 的端口 4 负责输出会随加速度变化的电压值。
- 端口 8 为自检测（Self-Test）端口。
- 端口 5 为状态位（Status）。
- 端口 4 连接至 S12 的 PAD3，用 A/D 转换端口 3 来转换电压信号。

6.3.2 干扰因素

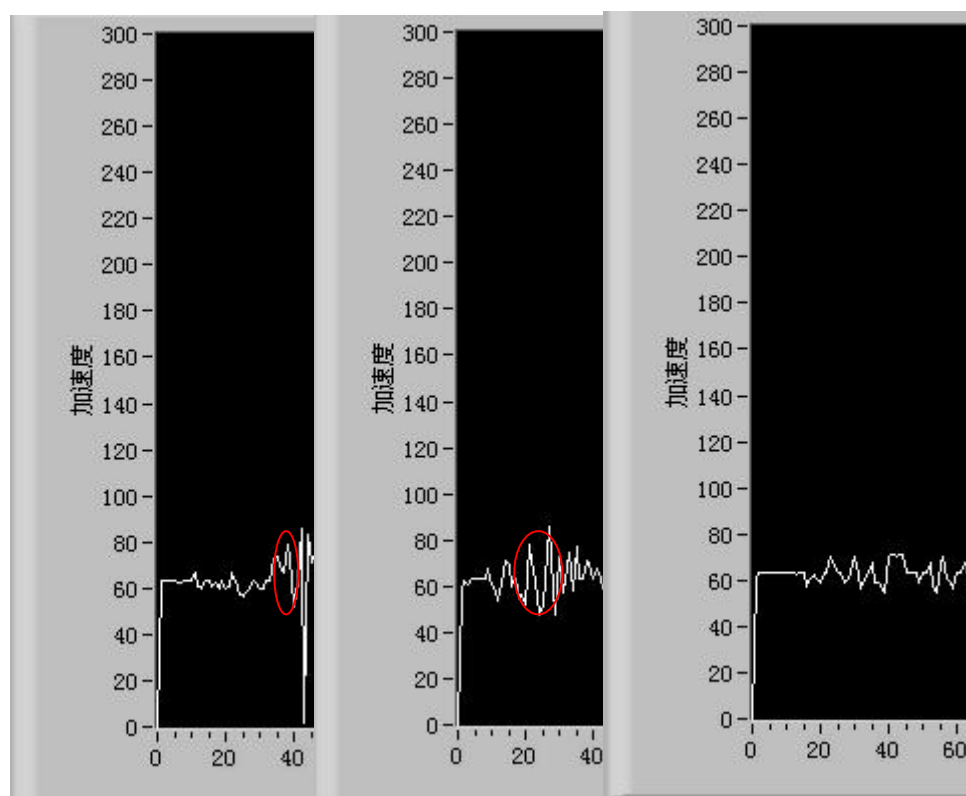
这块芯片可以测试一个方向上的加速度变化。所以采用了不同的安装方式，其间要考虑到比较多的干扰因素。

- 车体行进过程中的机械振动
- 速度控制不平滑对车体运动状态的影响
- 常规情况下的误差和正常时，加速度传感器产生的输出区别大小

6.3.3 安装方式分析

- 若安装于安装摄像头的立杆上，杆子在小车加减速的时候有比较明显的左右晃动，通过 Labview 观察发现干扰过大。
- 若安装于小车后部，由于电机自身的振动，对 MEMS 产生的影响会较之前更为明显。
- 经测试，如果测量水平 X 方向的加速度，宜将装有 MMA2260 的电路板装载在车前部。
- 若想要测量竖直 Z 方向上的加速度，板子只有采用立式安装的方式。

6.3.4 坡道检测

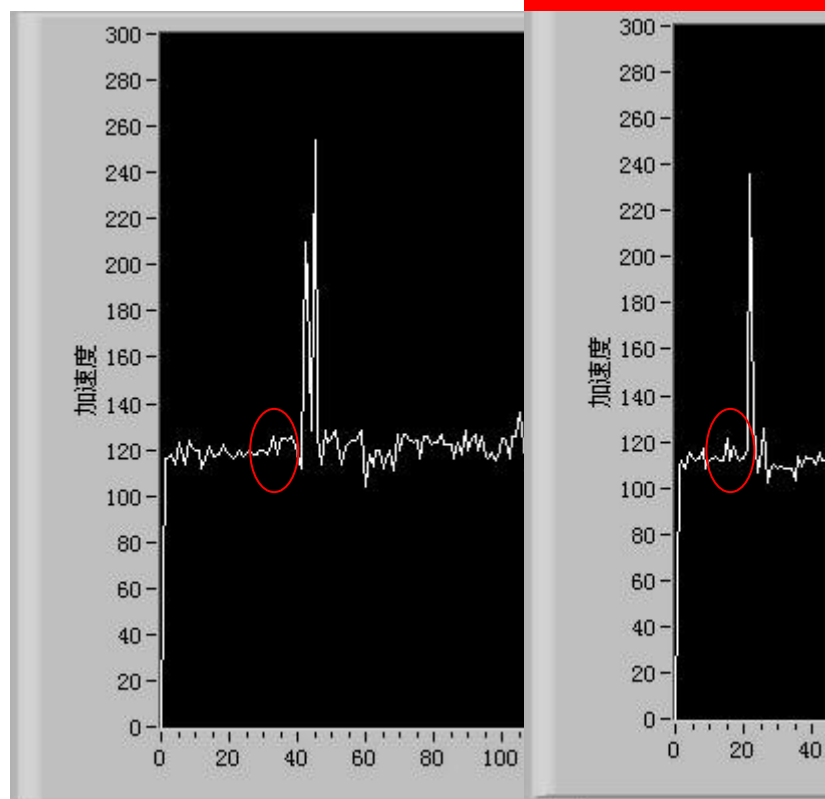


下坡

上坡

直道

工作方式：Z 方向加速度，安装于立杆上。



下坡

上坡

工作方式：X 方向加速度，安装于车头。

6.3.5 分析结论

由 Labview 发回的曲线图可以见得，加速度变化主要在坡顶和坡底两个部分。而其变化量在 $(-20, 20)$ 的范围内，从直道的加速度情况对比看来在 $(-10, 10)$ 范围内波动，相比而言不够明显。而且小车走完坡道，由于坡度的突变，由急变到缓，小车重新调整速度，会随之产生一个巨大的波动。如果需要利用 MEMS 加速度传感器，还需要进一步试验。

6.4 赛道记忆

智能车邀请赛规则中明确指出，赛车在赛道上连续跑两圈，并记其中单圈

最好成绩为比赛成绩。这个规则提示参赛队员，可以在第一圈时将赛道信息记录下来并进行分析，然后在第二圈时对控制算法进行优化，以提高比赛成绩。去年已经有某些队伍使用了这类赛道记忆算法，并收到了很好的效果。前面提到，为了提高直道速度必须使预判距离尽可能远才行，记忆算法就能满足这一要求。

6.4.1 路径记录方式

第一圈的信息蕴含在赛车所走过的路径中，因此选择何种方式将路径记录下来是记忆算法的关键之一。其中较为直接的一种方式就是记录赛车的里程和对应的舵机转角。图 6-6 是通过独立开发的仿真平台得到的里程-转角关系曲线，其中舵机转角的单位为弧度，里程单位为米。从图中可以看出，赛车先从直道（舵机转角在 0 附近）驶入一个左弯（舵机转角大于 0），然后经过一段 S 弯，接着又是一小段直道加左弯……，最后是 S 弯右弯左弯到直道。如图 6-7 所示，由于赛车的舵机转角控制策略采用的是简单沿线方式，通过这一曲线得到的赛车行驶路径和真实地图是非常吻合的。

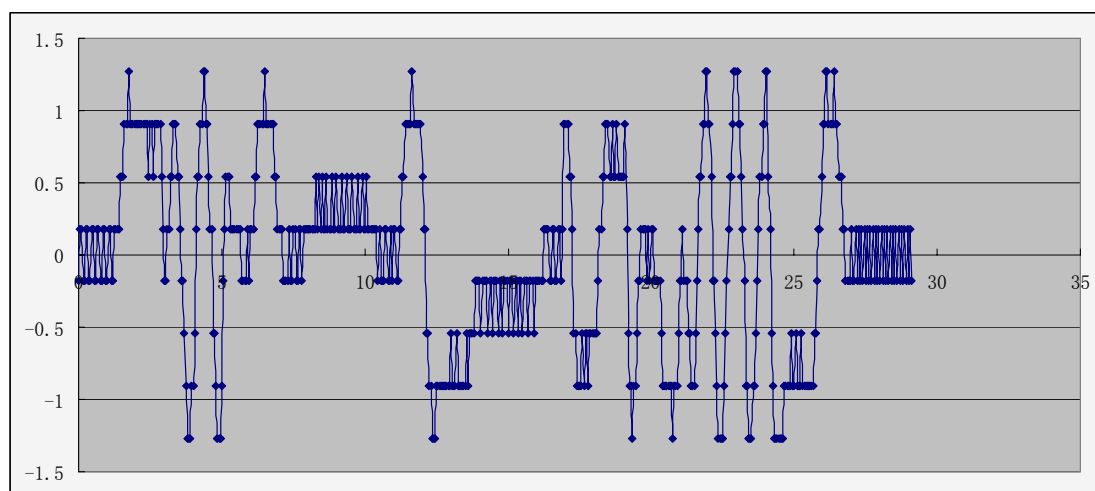


图 6-6 仿真平台中导出的转角-里程关系



图 6-7 仿真地图

由此可见，通过记录里程-转角的方式来对路径进行记忆是可行的。事实上，在自主移动机器人领域经常使用里程计进行定位。但是实际应用时，必须充分考虑到里程计的误差。

里程计的误差可以分为两类：系统误差和非系统误差。系统误差是由于轮子直径、轴距等安装的精度所造成的，而非系统误差是由死锁、打滑等不连续滑动所造成的。通常里程计的误差会随着路程长度累积，如果赛道长度较长，则势必造成较大的误差。实际测试下来，控制赛车在 21.5 米长的赛道上沿线行驶，最后的里程误差在 0.1-0.2 米左右。这一距离和赛车长度在一个数量级上，应该对第二圈的控制不会产生太大影响。

6.4.2 数据存储

要成功实现赛道记忆算法，必须有以下几大前提：

- 1、赛车在第一圈必须安全走完全程以记录数据。

- 2、赛车必须能明确分辨出第二圈。
- 3、赛车必须能够有足够记录一圈赛道数据的存储空间。
- 4、赛车记忆和处理信息的时间必须不影响赛车的控制时间。

第一个条件通常都能满足，另外前面已经对起始线进行了识别，因此第二个条件也已满足。这里主要讨论存储空间和处理时间的问题。

由于单片机芯片的RAM容量有限，而记忆算法需要大量的存储空间记录赛道的信息，第一圈记录的数据越详细，第二圈可用于控制的信息就越丰富，因此如何记录和存储足够的数据也是运用记忆算法的基础。当然，算法研究者可以对算法进行优化处理，改变保存的格式以节省内存空间。例如采用一定的压缩算法，把直道，弯道分成若干级别，分别记录他们的距离和顺序。优化压缩虽然可以节省存储空间，但对于赛道信息多少都会有丢失，不利于记忆算法的准确使用，因此如果可以获得更大的存储空间，就可以完整记录赛道的特征信息。

S12 单片机提供了 2K Byte 的 EEPROM，EEPROM 是一种电可擦除可编程只读存储器，其内容在掉电的时候也不会丢失。目前赛车的平均速度一般都在 2 米左右，对于一个 50 米长的赛道，25 秒内就可以跑完。由于摄像头更新频率 50Hz 的限制，赛车的控制周期一般都是 20ms，由此可以得出跑完一圈才运行 1000 多个周期，就算每 2 个周期记录一个 16 位的信息，EEPROM 的容量也已足够。

在对 EEPROM 进行编程和擦除时，要严格地遵守下列步骤^[11]：

首先，访问 ECLKDIV 寄存器，进行时钟分频，EEPROM 需要编程时钟在 150-200KHz 之间。如果没有设定时钟分频，对 EEPROM 的一切操作无效。

检查 CBEIF 标志位，确定地址、数据和命令缓冲区是否为空。然后，进行如下的三步操作（注意，操作顺序不能改变，中间也不可插入其他操作）：

向需要编程的地址写入一个字（2Byte）。如果是全擦除操作，可向 EEPROM 中任意一个地址写入数据；如果是向量擦除，则二进制地址的最后两位将被忽略。

向 ECMD 寄存器写入命令字。

向 CBEIF 位写入 1 开始命令。CCIF 位置 1 表示命令被执行完毕。用户可以通过中断机制确定命令的完成。

如果出现无效的命令字或是企图修改保护区内容的操作，ACCERR 和 PVIOL 位会置位。

EEPROM 的写和读函数 `ee_write()` 和 `ee_read()`。其中 `ee_write()` 向一个给定的地址里写入一个整型数，而 `ee_read()` 从给定的地址读出数据。测试表明，对 EEPROM 每次进行写操作所需的时间大约为 0.9ms，而每次读的时间仅需 2us。写操作的时间较长，这应引起足够的重视，因为对于摄像头控制方案，保证控制周期在 20ms 以内是非常必要的，否则赛车在高速行驶时将来不及反应。除此以外，对于记忆信息的处理时间也应尽量压缩。由于第一圈控制算法通常较为简单，这给信息处理提供了一定的时间裕量，所以只要算法得当，赛道记忆算法的几个条件都能满足。

6.4.3 第二圈控制优化

记录第一圈信息的关键是为了第二圈能跑得更好，所以对于使用赛道记忆算法的智能车来说，第二圈才是最关键和最吸引人的部分。在第二圈，如何充分利用第一圈收集的赛道信息，跑出最优的成绩，是需要仔细研究和考虑的。

根据转角大小可以判断赛道某处的曲率，如果利用第一圈得到的弯的曲率半径，在过各弯道时将车速调至它相应的极限速度，那么理论上这就是它所能跑出的最优成绩了。对于直道，特别是长直道，更能发挥赛道记忆算法的优势。例如，在第一圈记忆后得到某段直道为 2800 个脉冲点长，那么在决策中就可在前 2500 个脉冲时启动电机用最大占空比加速，超过 2500 脉冲时直接减速，将速度调在下一个弯道的极限速度进行过弯。由于事先已经知道直道的长度，因此调速策略更能有的放矢。

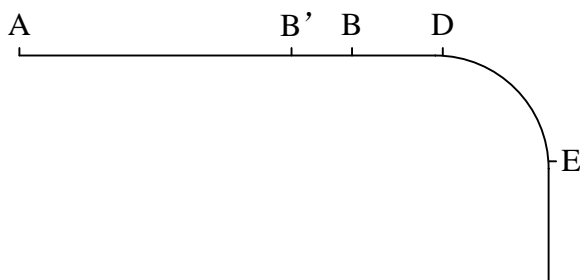


图6-8 记忆算法速度控制示意图

如图6-8所示，第一圈中摄像头的预判距离有限，假设由直道入弯时在B点开始减速，那么为了保证经过BD后车速减到过弯的安全速度 V_{safe} ，直道最高速度限制在 V_1 以内。第二圈有了记忆信息以后，相当于知道了整个赛道，那么可以提高直道速度 V_1 ，但减速距离会有所增加（此时可能从B'处开始减速）。一般地，只要AD远大于B'D，第二圈的平均速度就会更快一些。

6.4.4 问题和局限

目前大学生智能车比赛的赛道有着越来越复杂、越来越长的趋势，十字交叉线也将不只出现一个，这对于第一圈的记忆和数据处理都产生了很大的挑战。同时由于决赛之前赛道是不公布的，因此赛道记忆算法存在很大的风险，较保守和安全的算法在决赛中可能更有利。除此以外，记忆算法还有一个很大的局限，那就是第二圈和第一圈通常要保证行驶路线的重合，这在一定程度上限制了记忆算法的性能。

尽管如此，记忆算法独特的思路还是值得继续发展下去的，有着很大的发展潜力。预计在将来的比赛中，这种处理方式将得到广泛地应用，第一圈和第二圈将被区分开来，第二圈将在第一圈的基础上有所提高，而不是简单的重复。

第七章 开发工具、制作调试过程

前面几章从硬件设计到软件构架对智能车系统做了详尽的介绍和分析。一个系统方案制定好后，其实主要的工作是不断细化和调试。在对程序进行开发和软硬件联调的过程中需要一整套的软件开发与调试工具。在整个程序开发过程中，除采用汇编语言开发的方式外，还可使用 Metrowerks 公司为 MC9S12 系列单片机专门提供的全套开发工具（Metrowerks Codewarrior IDE）。这是一套用 C 语言进行编程的集成开发环境——本文智能车系统的软件设计部分就是在此开发环境下完成的。同时为了观察摄像头采集和控制算法的效果，还采用了 NI 公司的虚拟仪器 Labview 作为辅助开发调试工具。

7.1 软件开发平台 Codewarrior IDE

Codewarrior 是由 Metrowerks 公司提供的专门面向 Freescale 所有 MCU 与 DSP 嵌入式应用开发的软件工具。其中包括集成开发环境 IDE、处理器专家、全芯片仿真、可视化参数显示工具、项目工程管理、C 交叉编译器、汇编器、链接器以及调试器。其中在本设计方案中最为重要的部分就是集成开发环境 IDE 以及调试器，所以接下来将主要介绍这两部分。

7.1.1 Codewarrior IDE 功能介绍

CodeWarrior IDE 能够自动检查代码中的语法错误，它通过一个集成的调试器和编辑器来扫描代码，以找到并减少明显的错误，然后编译、链接程序以便计算机能够理解和执行程序。每个应用程序都经过了像 CodeWarrior 这样的开发工具进行编辑、编译、链接和调试的过程。

具体到比赛所应用 MC9S12DG128 的一个显著特点就是片上外围设备众多，每个外围设备对应的寄存器也较多，有的甚至达到三十多个。Metrowerks Codewarrior IDE 中的 mc9s12dg128.h 文件对所有寄存器对应的存储映射地址都

进行了宏定义，开发者在软件开发时直接调用这些宏就可以了（如图 7-1）。这些宏的名称都与说明文档上相应寄存器的名称相同或近似，这样便于对 MC9S12DG128 的开发。

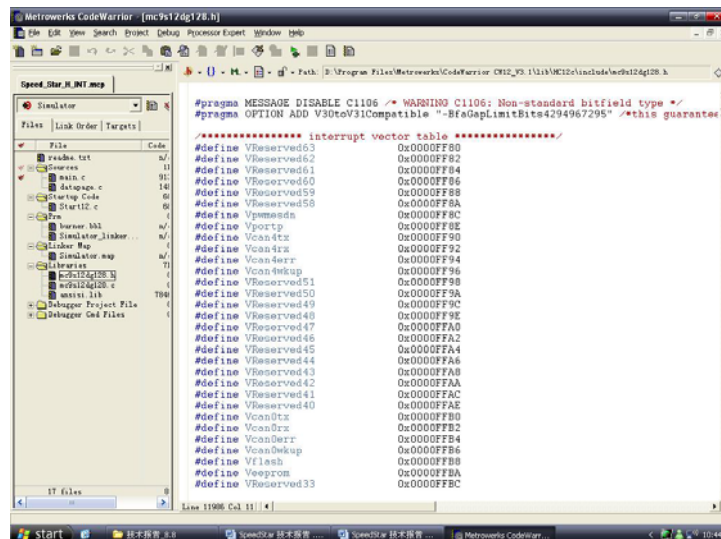


图 7-1 S12DG128B 的库文件

7.1.2 Codewarrior IDE 基本使用方法

A 运行“开始菜单—>所有程序—>Metrowerks CodeWarrior—>CW12 V3.1—>CodeWarrior IDE”，选择“File—>New”，出现如下的对话框

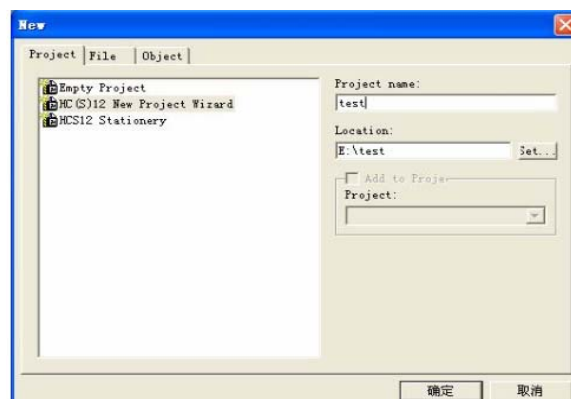


图 7-2 建立软件环境过程 1

接着选择 MC9S12DG128B，按照提示选取期望的选项。直至建立工程文件如下：

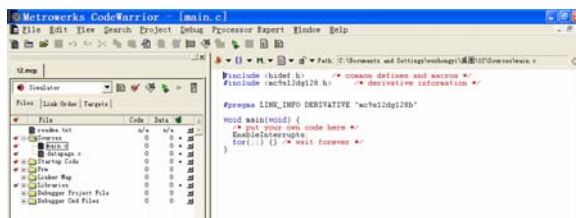


图 7-3 建立软件环境过程 2

B 打开 Start12.c 文件，找到代码

```
#ifndef _HCS12_SERIALMON
```

```
#define __INITRM      (*(volatile unsigned char *) 0x0010)
```

```
#define __INITRG      (*(volatile unsigned char *) 0x0011)
```

```
#define __INITEE      (*(volatile unsigned char *) 0x0012)
```

```
#endif
```

修改成：

```
//ifndef _HCS12_SERIALMON
```

```
#define __INITRM      (*(volatile unsigned char *) 0x0010)
```

```
#define __INITRG      (*(volatile unsigned char *) 0x0011)
```

```
#define __INITEE      (*(volatile unsigned char *) 0x0012)
```

```
//endif
```

之所以要经过这一步骤是因为，初始化代码默认程序中有监控程序，而实际使用 BDM 调试工具就不需要监控程序了。如果不注释掉这两条语句，一旦将程序烧写到片内的 flash，会发现程序跑不起来，因为初始化过程在等待监控程序的响应，如果没有监控程序，那么程序就会进入死循环的状态。

C 添加代码：将自己的代码加入到工程文件中。

D 如果文件编译通过，则可用 BDM 进行下载。启动 CodeWarrior 4.1 自带的 Hiwave.exe 程序，该程序一般位于 CodeWarrior 4.1 的安装目录

“\Metrowerks\CW for HCS12X V4.1\prog”中，是用来往单片机下载程序和调试单片机程序的。启动后程序界面如图 7-4 所示。

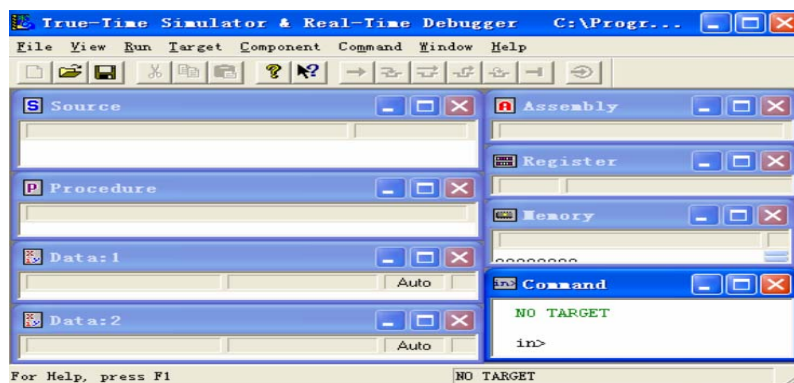


图 7-4 Hiwave 程序烧写环境

装好相关驱动后，界面如图 7-5 所示：

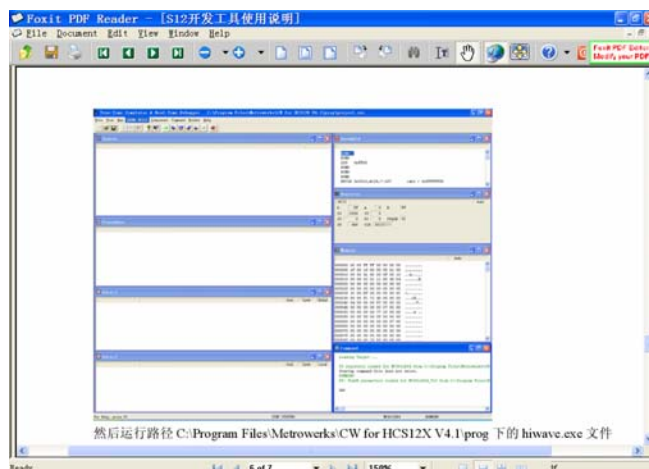


图 7-5 烧写环境建立界面

之后运行路径 C:\Program Files\Metrowerks\CW for HCS12X V4.1\prog 下的 hiwave.exe 文件选择 TBDML HCS12，在下拉菜单中选择“set speed”，输入晶振频率 16.00，然后这个下拉菜单中会出现“Flash”选项，点击这个选项，出现如下的对话框：

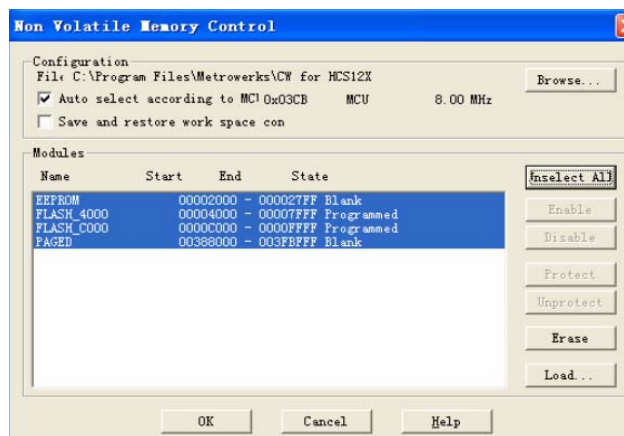


图 7-6 程序擦除烧写界面

在下载自己的程序前应该将板子中已经存在的程序擦除，选择有程序的部分，点击 Erase。擦除完成后，点击 Load，选择自己所建立工程文件夹中 bin 文件夹下后缀为“.abs”的文件，点击打开，下载完成。

至此就基本完成了从程序代码编写和设置，到程序烧写的主要过程。这些步骤是开发一个完整工程所必须具备的。

7.2 辅助调试方法

根据仿真平台需要实现的各个功能，设计出仿真平台程序框图，利用LabVIEW编程软件编写程序。由于在LabVIEW环境下，程序基于数据流运行，所以宏观上，程序按照框图中各个功能模块的先后顺序结构执行；微观上，具体到每个功能模块内部实现时，其执行顺序由数据流的先后顺序决定^[6]。如：小车的转角显示和速度显示都是在同一个循环体内实现，但其执行的先后顺序是随机的。尽管如此，这种随机性并不会影响到程序的正确执行。

在LabVIEW编程环境下，除了图形化的编程方式，与文本混合编程的特色，还值得注意的是其控件库包含了很多常用的功能模块。除了常用的数学函数，矩阵数组、显示、计时及文件I/O等模块外，还包括测量I/O模块、仪器I/O模块、应用程序接口模块、图形声音模块、信号处理模块、数据通信模块等。这些模

块体现了LabVIEW与外界设备进行交互的强大功能，也符合了“仪器”的本质^[8]。每个功能模块集成了与外设交互的相关程序，因此程序员无需了解程序是如何跟外接设备建立连接的，也无需了解如何驱动外界设备，只需要根据功能模块的接口，接收/发送相关的数据即可。所以，与外接设备交互的繁琐细节被LabVIEW屏蔽了，程序员可以集中精力处理主程序。这为程序员提供了极大的方便，也最大限度的减小了程序体出现漏洞的可能性。在本硬件在环仿真平台中，用到LabVIEW提供的通信协议里的VISA配置串口，利用串口进行无线通信，轻易的就能实现数据的收发。

7.3 无线调试模块

7.3.1 Zigbee 无线通信模块

在智能车的制作和调试过程中，需要将赛车检测到的路面信息以及速度等参数实时地发送给 PC，以便对算法进行有针对性的分析。由于小车在行驶时不能通过有线的方式获得其运行参数，就需使用无线方式。



图 7-7 无线传输示意图

这里选择的是 Zigbee 无线通信模块，它可以设置传输速率，以数据包的形式对数据进行收发。下图为 Zigbee 模块的示意图。

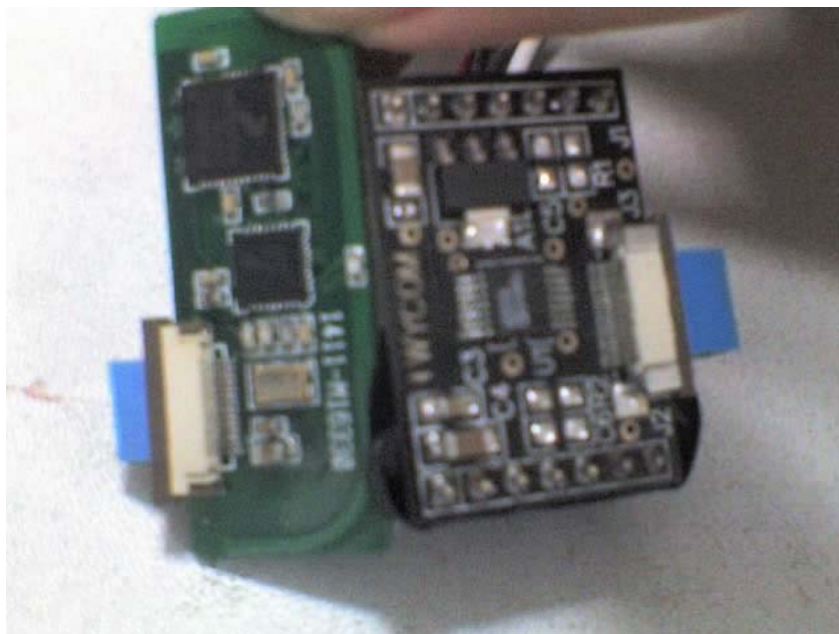


图 7-8 Zigbee 无线通信模块

功能板上装有传感器、发光二极管、显示屏等元件，PC工具软件发送的控制指令通过无线网络传递到各功能板上，达到对所有元器件的监控目的，用这些简单的演示来模拟各种现场的实际应用。

工具中的控制板是USB接口，可与PC直接连接，并通过PC端的工具软件对其进行控制



图 7-9

接口标准

模块尺寸：40*16*3（mm）

接口：14PIN 扁平电缆

规格：0.5mm 间距，下接方式

推荐使用与模块相同的排线座

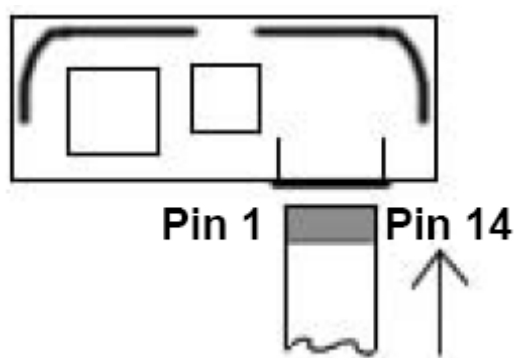


图 7-10

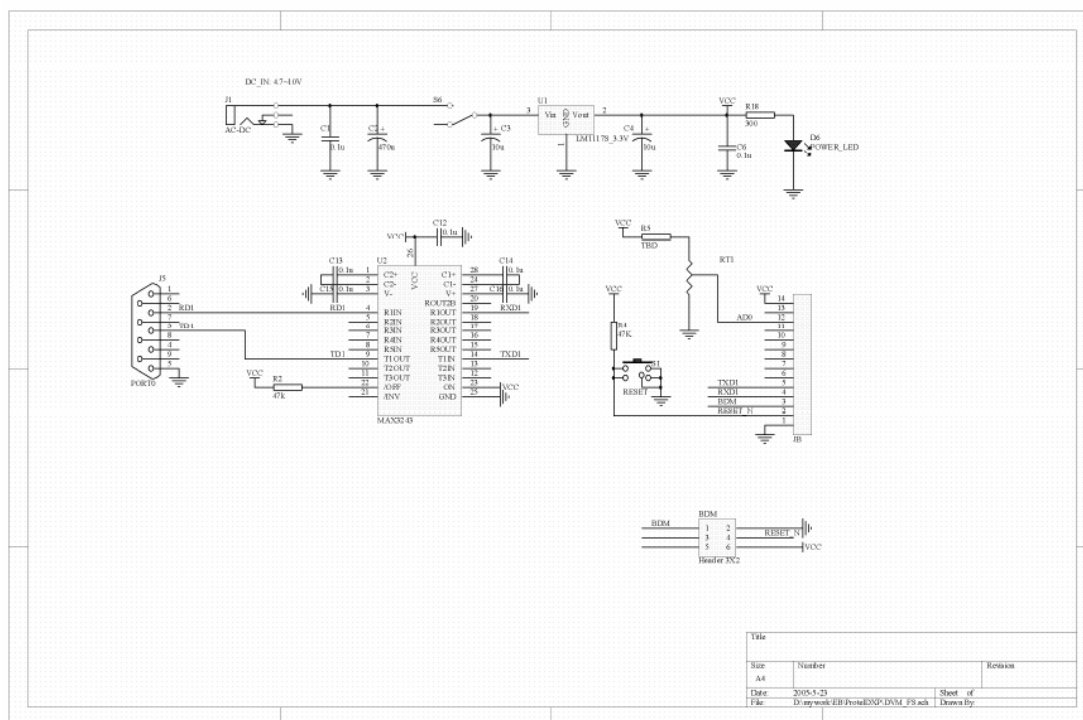


图 7-11 参考设计原理图

7.3.2 天线的摆放

如下图，两个模块处在同一高度上，且可以在y轴上进行适当的旋转，但需尽量使天线处于同侧，这时通讯效果最好。请注意尽量不要在其他轴上进行旋转。

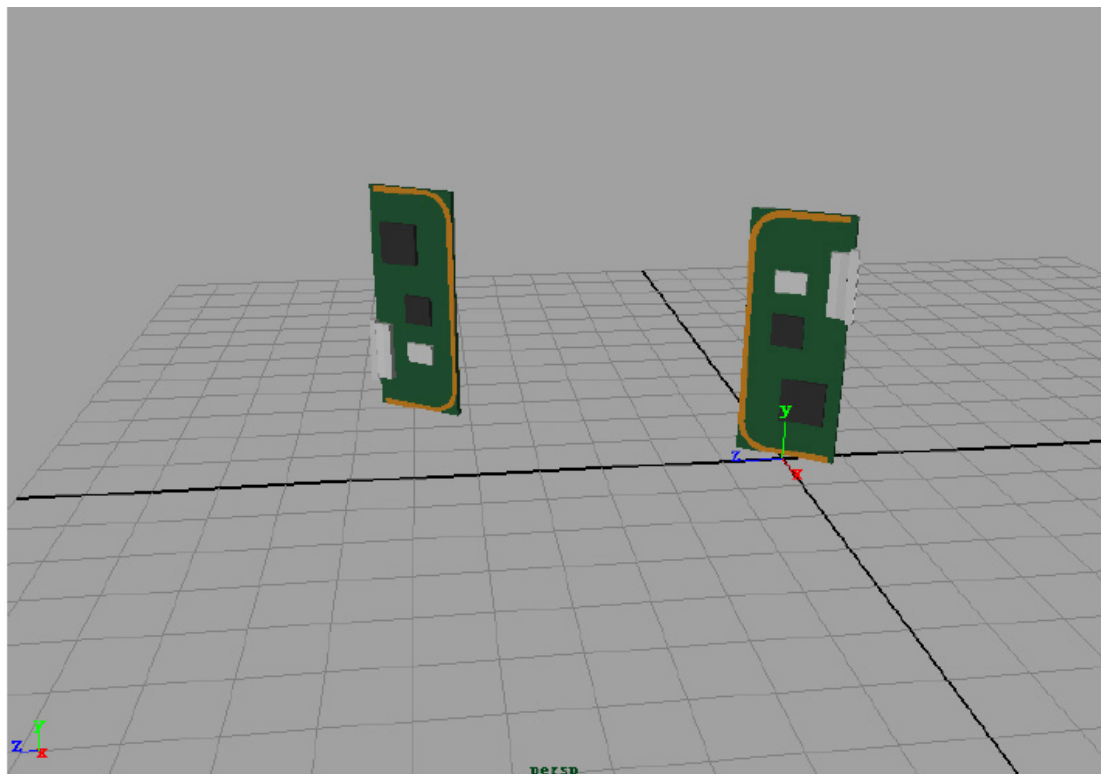


图 7-12 正确的天线收发示意图

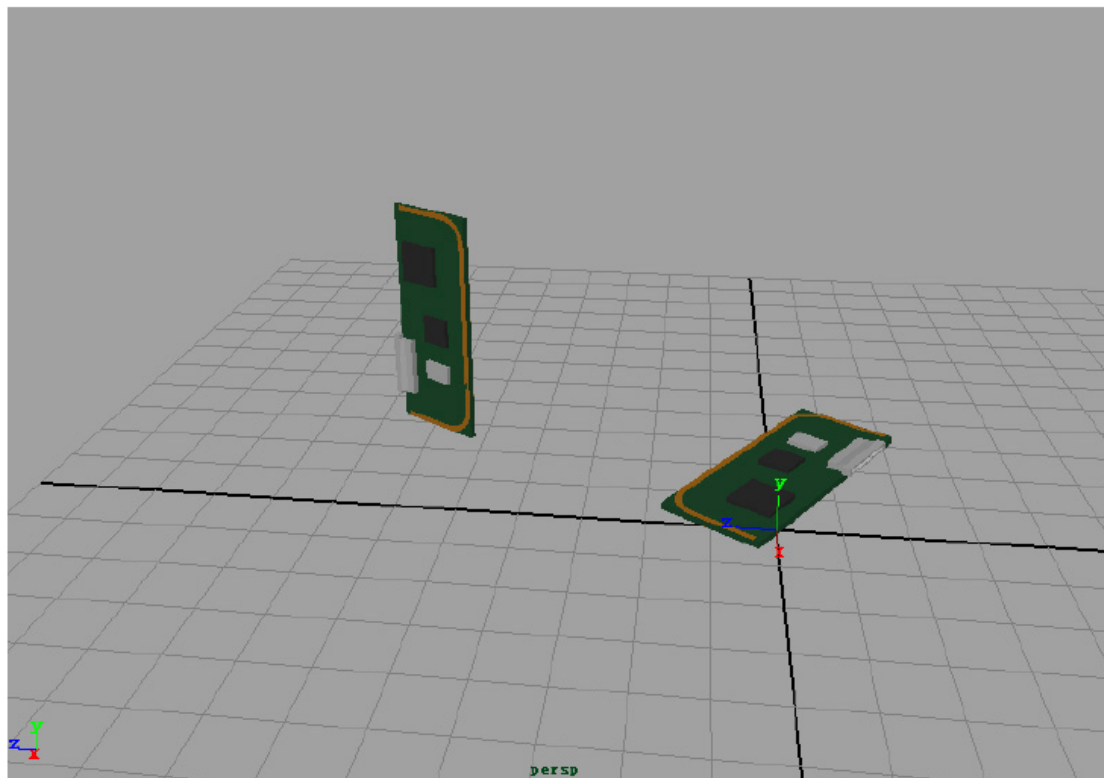


图7-13 尽量避免的天线收发示意图

注意事项

- 1) 在使用时，不要用手直接接触模块的芯片，以免芯片受到静电损伤。
- 2) 在拔插扁平接口电缆时，不要用力过猛，以免损坏接口。

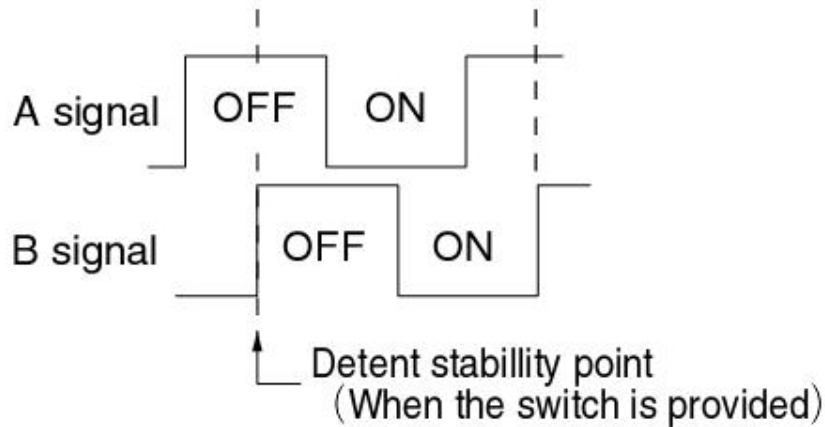
7.4 数码开关

7.4.1 原理简介

新的 S12 电路板上将拨码开关换成了数码开关，数码开关可以左右旋转并可以按下，如 S12P80V1.mdi 图所示，开关有三个输出端口 QESA, QESB, QESP，它们分别与单片机的 PT5-PT7 相连。

如下图所示，当数码开关顺时针旋时，A, B 口的电平从右向左变化，依次是 10, 11, 01, 00，而当数码开关逆时针旋时，A, B 口的电平从右向左变化，

依次是 00, 01, 11, 10。



因此可通过不断读 PT5 和 PT6 口判断开关是否在旋转并确定方向，通过软件或硬件计脉冲数确定旋转量（由于一圈有 32 个脉冲，所以我将脉冲数除以 5，操作效果较好）。而 QESP 在位按下时为高电平，按下后即变为低电平，所以可用排脉冲捕捉，通过读标志位判断是否按下。

板子上有两个八段数码管和 5 个并列的 LED 灯可用于数码开关的操作和程序运行时的显示。

7.4.2 程序使用

了解了各元件的控制端口后我开始编写操作和显示的程序。程序已经写成了函数，可移植到其它程序中直接使用，只要执行 void operate(void) 函数即可。

程序开头是对一些端口做的宏定义，还定义了一些全局变量，其中 LEDvalue[5] 存储操作后所定的参数值（未操作前初值为零，若要改变初值，可在 void operate(void) 函数的开头部分对 LEDvalue[] 数组赋值），中除第一个元素外，其余四个均可赋值给控制程序中的任何参数。

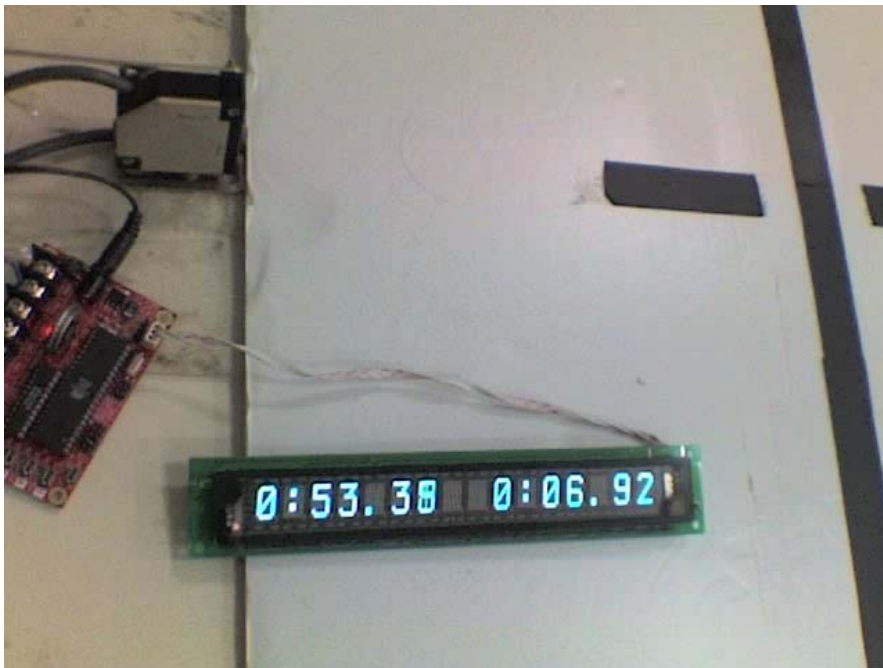
若要单独使用数码管显示可直接调用 void dis_digital(char number) 函数。

7.4.3 操作方法

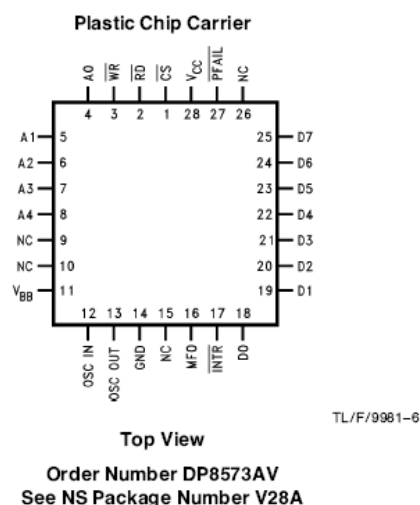
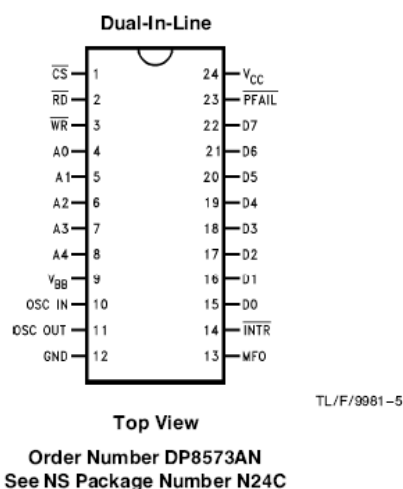
数码开关顺时针旋转为增加，逆时针为减少，按下为确定。

进入操作程序，最上方一个 LED 灯亮，数码管显示为“0”，此时若按下数码开关，即不改变任何参数，跳出操作界面。若旋转数码开关可选择第几个 LED 亮（即选择要调整的参数），移动到所需要的灯后，按下即可调整该灯所对应的参数，调整完，按下确定后修改即生效。然后可继续调整其它参数。若要退出，选择第一个 LED 按下确定键即可。

7.5 计时器



计时器芯片是 DP8573ARTC 芯片。引脚定义如图



定时器精度为 0.01 秒。完成计时器程序大致分为两个部分，一个是计时功能实现，另一个就是计时器初始化（端口初始化和显示），最后将光电触发功能加入计时器。在按键扫描中增加了对 P1.2 口的扫描，为了防止小车通过时多次触发，加入了延时程序（在计时开始后程序主体经过 20 次循环即 0.53s 后才再次响应其他触发），最终的程序比较稳定。

计时器操作方法：

1. 接线：发射管接 VIN01 和 VIN02 不分极性

接受管：灰—GND 蓝—GND 棕—VPP 黑—SIN 白—GND

显示屏将白色接口插上即可

2. 操作：接上电源，屏幕显示 smartcar 即正常工作，注意对射装置要对准。若不正常工作，断开电源在插上。若要重启可按下右边第一个按键 reset。计时器也可手动操作，右边第二个按键是开始，第三个是停止。
3. 注意事项：最好不要长时间通电，防止板子过热。

第八章 结论

本文详细介绍了为第二届全国智能车大赛而准备的智能车系统方案。该系统以 Freescale 16 位单片机 MC9S12DG128 作为系统控制处理器，采用基于的摄像头的图像采样获取赛道图像信息，通过边缘检测方法提取赛道黑线，求出小车与黑线间的位置偏差，采用 PID 方式对舵机转向进行反馈控制。通过速度传感器对小车形成速度闭环控制。

文中介绍了赛车机械结构和调整方法，赛车各个主要模块的工作原理和设计思路，并叙述了系统开发过程中所用到的开发工具、软件以及各种调试、测试。

综合来看，智能赛车分硬件和软件 2 部分。硬件部分主要是赛车的安装和调整，以及电路板的设计。软件部分是本文的重点，除了基础工作外，主要分为三部分：

（1）视频采集和图像处理

视频采集和图像处理是所有算法的基础，只有看得清楚，才能控制准确。与上届比赛相比，由于采用了跟合理的硬件设计和优化的软件算法，图像采集的精度有所提高，虽然比赛不一定用到如此高的精度，但更高的采集精度可以提供更多的图像信息。同时在图像识别上也有进步，从原理单一的识别黑线发展到现在的特定图形识别。

（2）车体控制

车体控制主要分为速度控制和舵角控制。速度控制相比去年有了很大进步，更短的控制周期使加减速更及时，也是速度更加平滑。新使用的速度传感器也大大提高了闭环控制的精度，能够更加准确的控制小车在不同赛道跑出响应的极限速度。舵机控制关系到方向的选择，好的舵机控制不仅要准确循线，还要能够沿最优路线前进。

（3）记忆算法

作为智能赛车，就要有一定的预判能力，虽然摄像头的视野比光电管大出

数倍，但还是有限的，而记忆算法的采用是预判距离变为整圈赛道，这使得我们可以从全局的高度来控制小车，使小车的速度和角度在不同的路段上都达到最佳。这也是未来的发展方向和研究重点。

总结整个设计过程，不仅使我们得到了实战的锻炼机会，更是培养了一定的科研能力。展望未来，智能车技术必将在更广阔的领域得到广泛应用。

参考文献

- [1] 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. 2004 年 10 月第 1 版.
- [2] 杨国田, 白焰. 摩托罗拉 68HC12 系列微控制器原理, 应用与开发关系. 中国电力出版社. 2003.9.
- [3] Freescale Semiconductor, Inc. MC9S12DG128 Device User Guide.
- [4] <http://www.freescale.com.cn/>
- [5] 张晓飞, 袁祥辉. 基于 DSP 成像系统的视频图像采集部分的实现. 压电与声光, 第 24 卷, 第 3 期. 2002 年 6 月.
- [6] National Semiconductor. LM1881 Video Sync Separator General Description. April 2001.
- [7] 万永伦, 丁杰雄. 一种机器人寻线控制系统. 电子科技大学学报, 第 32 卷, 第 1 期. 2003 年 2 月.
- [8] 王荣本, 游峰, 崔高健, 郭列. 基于计算机视觉高速智能车辆的道路识别. 计算机工程与应用 2004.26.
- [9] 李华. 自主轮式机器人 THMR-V 的混和模糊逻辑控制. 机器人 2003.11.
- [10] 朱卫华. 一种微机与单片机无线串行通信的设计方法. 南华大学, 硕士学位论文, 2002 年 8 月.
- [11] 楚现知, 吴吉祥, 李锦忠. 基于 LabVIEW 的监控界面设计与单片机的串行通信. 工业控制计算机, 2005 年 18 卷第 7 期.
- [12] 孙鑫, 余安萍. 编著. VC++深入详解. 电子工业出版社, 2006 年 1 月.
- [13] 鲍晓东, 张仙妮. 智能交通系统的现状及发展. 交通论坛, 2006 年 8 月:15-18.
- [14] 黄开胜, 金华民, 蒋狄南. 韩国智能模型车技术方案分析. 电子产品世界, 2006(5): 150-152.
- [15] 许欢, 唐竞新. 调制光寻线系统及其在自动行走机器人中的应用. 清华大学学报 (自然科学版), 2002:第 42 卷, 第 1 期.

- [16] 卓晴, 王珽, 王磊. 基于面阵 CCD 的赛道参数检测方法. 电子产品世界, 2006(4):141-143.
- [17] 关军, 吴中仪, 胡晨晖. 上海交通大学 CyberSmart 队技术报告.
- [18] 贾秀江, 李颢, 戴丰, 田兴华. 上海交通大学 SpeedStar 队技术报告.

鸣谢

在本次智能车大赛的准备期间, 我们遇到了许多困难。从最开始的程序烧写到后来的传感器调试, 从电路板设计到系统搭建, 一个个问题的解决见证了我们这支队伍的成长。期间离不开老师、同学们的大力支持, 是他们的帮助让我们做得更好。

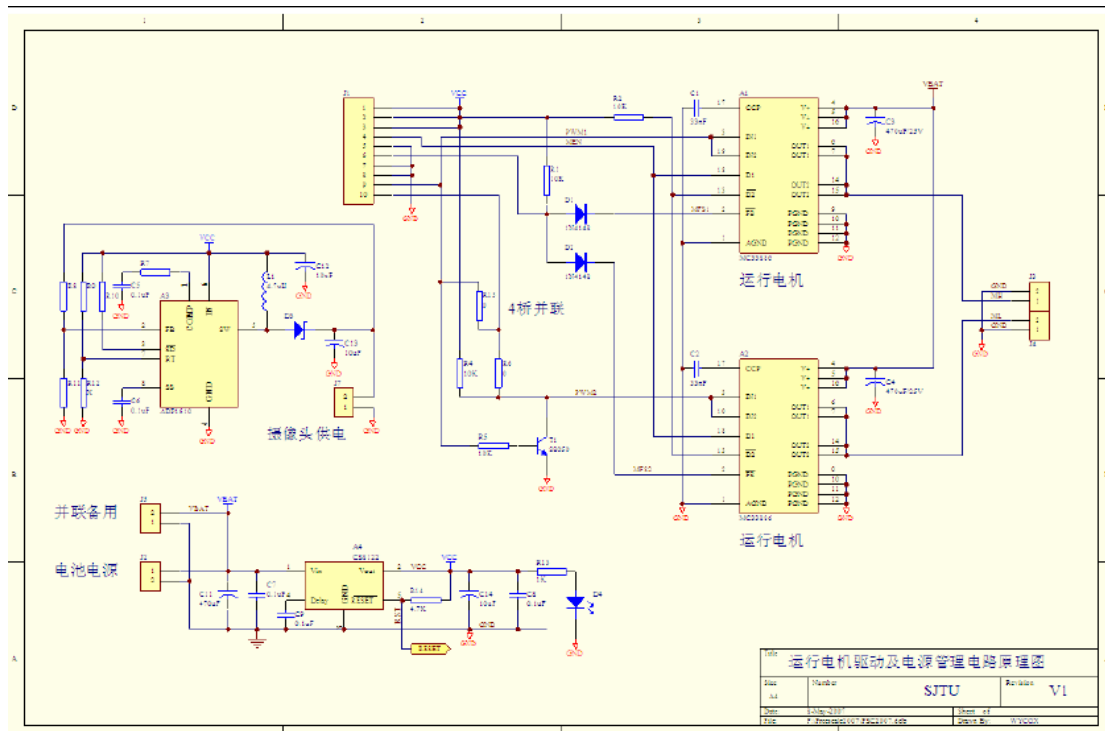
首先, 要感谢杨明老师和王冰老师的悉心指导。没有他们在思路上的指点, 在场地、器材方面的支持, 我们将进展缓慢。

其次, 要感谢上海交通大学自动化系田主任的关心。在准备比赛期间, 主任经常询问比赛进度, 为我们解决了许多困难。

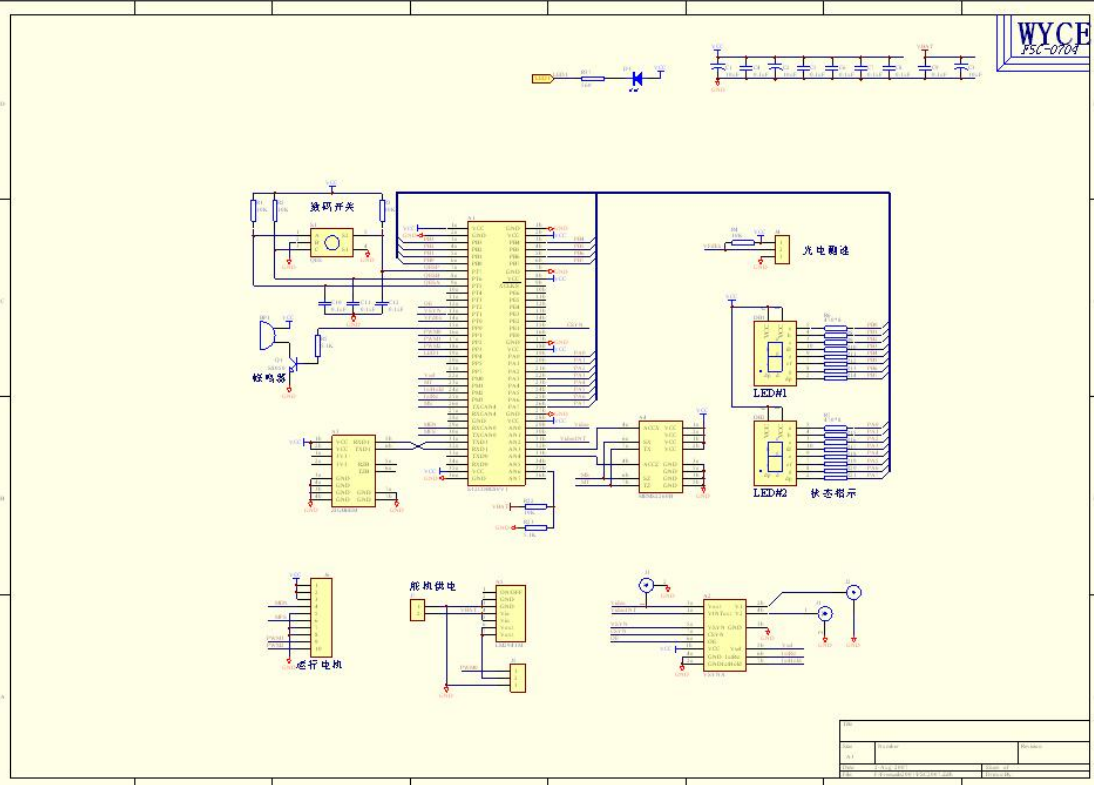
最后, 要感谢实验室其他成员的帮助。他们在比赛期间为我们提供了很多方便。

附录

附录1 驱动电路板原理图



附录2 核心控制板原理图



附录3 摄像头标定原理

一、基本方法：逆透视变换，IPM (inverse perspective mapping)

摄像头标定的目的是建立世界坐标系和图像坐标系之间的映射关系。

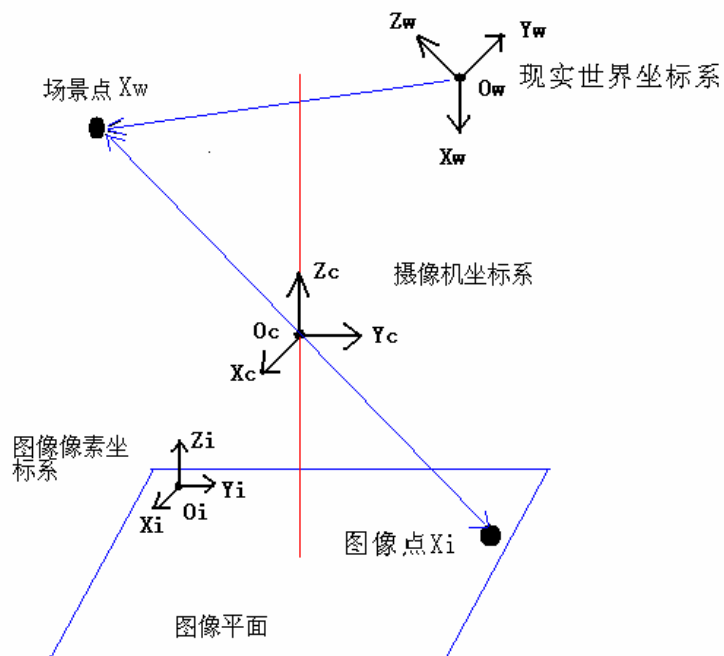


图 A-1

记图像坐标系 $U(u, v, 1)$ ，世界坐标系 $W(x, y, z, 1)$ ，可以得到如下关系： $U = MW$ 。

通过几个已知的对应点的坐标可以解出 M 。将上式展开可以得到：

$$\alpha \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = M \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \quad (\text{公式 A-1})$$

由于我们需要的是从图像坐标得到世界坐标的信息，所以实际过程是上面提到的逆过程。

要从二维的图像坐标知道真实世界的坐标由于缺少环境的深度信息，需要一个基本假设即“地面平坦”，然后就可以通过逆透视变换（IPM）建立映射关系。

有了地面平坦的假设，可以令 $z = 0$ 。这样（1）式可以变为：

$$\beta \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = N \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \quad (\text{公式 A-2})$$

进一步展开：

$$\begin{aligned} \beta x &= n_{11}u + n_{12}v + n_{13} \\ \beta y &= n_{21}u + n_{22}v + n_{23} \\ \beta &= n_{31}u + n_{32}v + n_{33} \end{aligned}$$

消去 β ，写成矩阵形式：

$$\begin{pmatrix} u & v & 1 & 0 & 0 & 0 & -ux & -vx \\ 0 & 0 & 0 & u & v & 1 & -uy & -vy \end{pmatrix} \vec{N} = \begin{pmatrix} x \\ y \end{pmatrix} \quad (\text{公式 A-3})$$

$$\text{其中，} \vec{N} = \frac{1}{n_{33}} (n_{11}, n_{12}, n_{13}, n_{21}, n_{22}, n_{23}, n_{31}, n_{32})^T$$

理论上只要 4 个点的对应坐标就可以解出 \vec{N} ，通常我们选取更多的点，用最小二乘的方法解出 \vec{N} 。

二、最小二乘法解出 N

若选取 n 个点，则可以得到 $2 \times n$ 个方程，记为：

$$AX = B \quad (A: 2n \times 8, X: 8 \times 1, B: 2n \times 1)$$

将上式进行变换： $A^T AX = A^T B$ （公式 A-4）

$A^T A$ 为实对称矩阵且非奇异。利用对称线性方程组的三角分解法，可以解出这个方程组。

$A^T A = LDL^T$ (公式 A-5), L 为一个三角矩阵, D 为一个对角矩阵。

$$L = \begin{pmatrix} 1 & & & \\ l_{21} & 1 & & 0 \\ \vdots & & \ddots & \\ l_{n1} & l_{n2} & \cdots & 1 \end{pmatrix}, \quad D = \text{diag}(d_{11} \quad d_{22} \quad \cdots \quad d_{nn})$$

$$\begin{cases} d_{11} = a_{11} \\ d_{ii} = a_{ii} - \sum_{k=1}^{i-1} l_{ik}^2 d_{kk} \\ l_{ij} = (a_{ij} - \sum_{k=1}^{j-1} l_{ik} l_{jk} d_{kk}) / d_{jj} \quad (j < i) \\ l_{ij} = 0 \quad (j > i) \end{cases} \quad (\text{公式 A-6})$$

当 L 和 D 确定后, 令 $DL^T X = Y$ (公式 A-7), 这样, $LY = B$ (公式 A-8)

由此, 按回代过程可解出 Y , 然后再由方程组 (7) 可以求出解向量 X 。

$$\begin{cases} y_0 = b_0 \\ y_i = b_i - \sum l_{ik} y_k \quad (i = 1, 2, \cdots, n-1) \\ x_{n-1} = y_{n-1} / d_{n-1, n-1} \\ x_i = y_i / d_{ii} - \sum_{k=i+1}^{n-1} l_{ki} x_k \quad (i = n-2, \cdots, 0) \end{cases} \quad (\text{公式 A-9})$$

三、得到对应的世界坐标

若解得 $N' = (n'_{11}, n'_{12}, n'_{13}, n'_{21}, n'_{22}, n'_{23}, n'_{31}, n'_{32})$

其中, $n'_{ij} = n_{ij} / n_{33}$

$$\text{则, } x = \frac{n'_{11}u + n'_{12}v + n'_{13}}{n'_{31}u + n'_{32}v + 1}$$

$$y = \frac{n'_{21}u + n'_{22}v + n'_{23}}{n'_{31}u + n'_{32}v + 1}$$

附录 4 模型车技术参数统计

项目	参数
车模几何尺寸（长、宽、高）（毫米）	279、158、428
车模轴距/轮距（毫米）	197
车模平均电流（匀速行驶）（毫安）	600
电路电容总量（微法）	1625
传感器种类及个数	摄像头，1 个
新增加伺服电机个数	0
赛道信息检测空间精度（毫米）	2.5
赛道信息检测频率（次/秒）	50
除 MC9S12DG128 之外其它主要芯片	LM1881,MC33886
车模重量（带有电池）（千克）	1.2