

第十二届“恩智浦”杯全国大学生智能汽车竞赛

# 技术报告



北 華 航 天 工 業 學 院

参赛学校：北华航天工业学院

参赛队伍名称：北华航天工业学院摄像头队

队员名称：邵泽西，柯锐，曹兴瑞

指导教师：陈宪，李建义

## 关于技术报告和学术论文使用授权的说明

本人完全了解第十二届“恩智浦”杯全国大学生智能汽车竞赛关保留、使用技术报告和学术论文的规定，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：邵泽西 柯锐 曹兴瑞

带队教师签名：陈宪 李建义

日 期：2017/8/18

引言.....	2
第一章                系统方案设计.....	3
1.1 系统总体方案的选定.....	3
1.2 系统总体方案的设计.....	4
第二章 智能车的硬件安装与设计.....	5
2.1 智能车汽车传感器的安装.....	5
2.1.1 鹰眼摄像头的安装.....	5
2.1.2 速度传感器的安装.....	6
2.2 重心高度调整.....	7
2.2.1 底盘的设计.....	7
2.3 前轮角度定位的确定.....	7
2.3.1 主销后倾角.....	7
2.3.2 主销内倾角.....	8
2.3.3 前轮的外倾角.....	8
2.3.4 前轮前束.....	8
2.4 舵机的安装.....	9
2.5 机械部分小结.....	9
第三章 智能车硬件电路设计.....	11
3.1 控制板电路原理图.....	11
3.2 电机驱动电路原理图.....	13
3.3 硬件电路小结.....	14
第四章 智能车的软件设计.....	16
4.1 控制设计算法说明.....	16
4.1.1 PID 算法控制.....	16
4.1.2 舵机控制算法.....	17
4.1.3 速度控制算法.....	17
4.2 图像处理算法说明.....	17
4.2.1 路线的选择.....	17
4.2.2 圆环的处理.....	18
4.3 开发环境.....	21
4.4 软件部分小结.....	23
结论.....	24
车模技术参数.....	25
参考文献.....	26
附录 A.....	27

## 引言

随着科学技术的不断发展进步，智能控制的应用越来越广泛，几乎渗透到所有领域。智能车技术依托于智能控制，前景广阔且发展迅速。目前，掌握着汽车工业关键技术的发达国家已经开发了许多智能车的实验平台和商品化的车辆辅助驾驶系统。有研究认为智能汽车作为一种全新的汽车概念和汽车产品，在不久的将来会成为汽车生产和汽车市场的主流产品。

面向大学生的智能汽车竞赛最早始于韩国，在国内，全国大学生“飞思卡尔”杯智能汽车竞赛从 2006 年开始已经举办了十一届，得到了各级领导及各高校师生的高度评价。大赛为智能车领域培养了大量后备人才，为大学生提供了一个充分展示想象力和创造力的舞台，吸引着越来越多来自不同专业的大学生参与其中。

全国大学生“飞思卡尔”杯智能汽车竞赛包括光电组、摄像头组和电磁组，其中数摄像头组的智能车速度最快，备受关注。本技术报告主要包括机械系统、硬件系统、软件系统等，详尽地阐述了我们的设计方案，具体表现在硬件电路的创新设计以及控制算法的独特想法。智能车的制作过程包含着我们的辛勤努力，这份报告凝聚了我们智慧，是我们团队共同努力的成果。

在准备比赛的过程中，我们小组成员涉猎控制、模式识别、传感器技术、汽车电子、电气、计算机、机械等多个学科，几个月来的经历，培养了我们电路设计、软件编程、系统调试等方面的能力，锻炼了我们知识融合、实践动手的能力，对今后的学习工作都有着重大的实际意义。

# 第一章 系统方案设计

## 1.1 系统总体方案的选定



## 1.2 系统总体方案的设计



图 1.1 智能车整体图

遵照本届竞赛规则规定，我组智能汽车系统采用山外的 32 位微控制器 MK60FN1M0VLQ15 作为核心控制单元用于智能汽车系统的控制，利用摄像头采集赛道明暗信息，主控单片机摄像头采集的信息采用模糊控制算法控制舵机打角转向，由于考虑到 C 车没有机械差速器，我们采用电子差速来过弯，保证动力轮内侧轮胎的磨损均匀。为了保证控制的准确性和快速性，我们使用编码器作为速度传感器，通过编码器返回信号形成闭环，加以 PID 处理来控制电机的转速。我组智能车共包括四大模块：MK60FN1M0VLQ15 主控模块、传感器模块、电源模块、电机驱动模块。各模块的作用如下：MK60FN1M0VLQ15 主控模块，作为整个智能汽车的集中处理中心，摄像头传感器、和光电编码器等传感器的信号，根据控制算法做出控制决策，驱动舵机和直流电机完成对智能汽车的控制；传感器模块，是智能车识别赛道的中枢，可以通过一定的前瞻性，提前感知前方的赛道信息，为智能汽车的主控做出决策提供必要的依据和充足的反应时间；电源模块，为整个系统提供合适而又稳定的电源；电机驱动模块，驱动直流电机和伺服电机完成智能汽车的加减速控制和转向控制。

## 第二章 智能车的硬件安装与设计

### 2.1 智能车汽车传感器的安装

在小车安装的过程中，一些传感器的安装必不可少，我们主要安装了三个传感器，速度传感器（编码器）、摄像头、无线通讯传感器，编码器的安装可以较准确的测出左轮右轮的速度，对小车的速度控制有着很好的帮助；摄像头的安装可以很好的对图像进行处理；

#### 2.1.1 鹰眼摄像头的安装

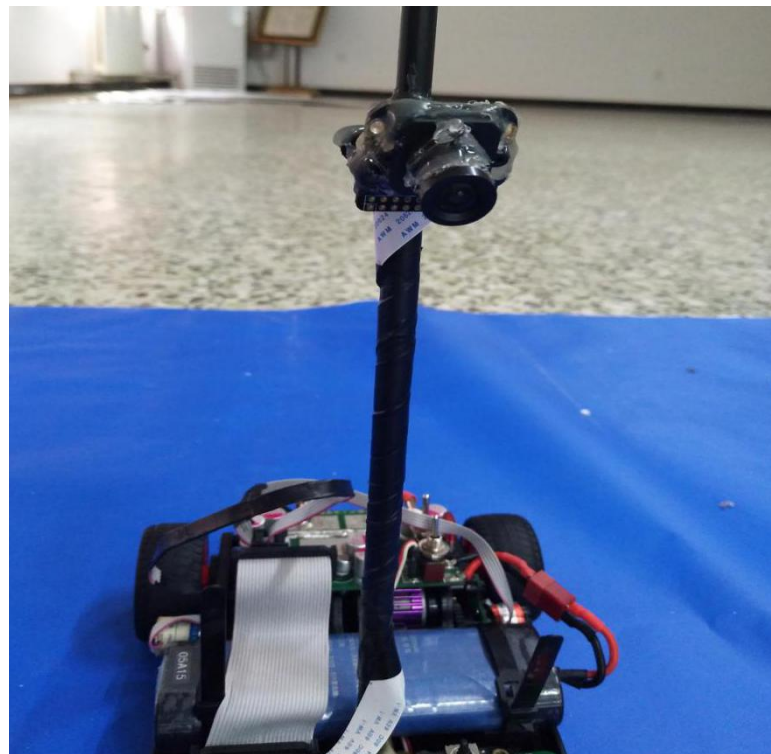


图 2.1 鹰眼摄像头

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模的中心位置，而且高度要适合于图像的采集和处理。通过多组对比试验，我们最后决定使用单杆结构来固定摄像头，单杆使得整个支架十分简约，但也会带来摄像头易抖动的问题。这时，单杆与底盘之间的连接可靠性就显得尤为重要。另外，摄像头的高度也会影响其抖动，架的越高则越容易抖动。为保证图像的稳定可靠，也为了在一定程度上降低车模重心，我们架低了摄像头使之与地面的距离在 26cm 摄像头与单杆之间的连接也十分重要，稳定牢靠的连接必不可少，连接件的重量也要控制在一定范围内，为此我们设计和加工了一种结构简质量轻巧的连接件；

### 2.1.2 速度传感器的安装



图 2.2 速度传感器

速度传感器一般可以选择对射式光栅或光电编码器。对射式光栅的重量轻，阻力小精度也高，然而光栅暴露在外界容易受到外界光线或粉尘等的影响，导致计数不准确；而光电编码器就不存在此类问题。所以我们选了蓝宙 512 线迷你编码器在安装编码器的时候要保证有合适的齿轮咬合。咬合完美的原则是：两个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；传动部分要轻松、顺畅，容易转动。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，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，咬合过紧，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。

## 2.2 重心高度调整

重心的高度和位置对小车的稳定很重要，高度要适中，位置要在车身的中心，为了使车在运行过程中更加平稳，我们对其重心和对称性进行调整，在做到左右对称的前提下再对重心加以改善。而车体的转动惯量和重心却是相互影响着，在重心降低后，极易造成车模前后的质量过大，增加转动惯量。对此，我们尽量降低重心，然后将车模的较重部位尽量贴近轮轴安置，从而减小转动惯量，这样，使在车模运行转向时更加容易控制。当然，还需要注意的一点是，根据杠杆原理，整辆车的质量分布不能过于集中，前后左右都需要做到尽量均衡。

### 2.2.1 底盘的设计

今年的车模较去年的简洁，底盘结构与第四届车模相似。对于底盘，我们没有做过多的改造，只是适当调整了前后底盘的高度使车模整体的重心下降到合适的位置，即保证小车以顺利过坡，且与赛道不会接触摩擦。用车模自带第十一届飞思卡尔全国大学生智能车竞赛零配件即可调整底盘前后的高度。合适的重心在小车过弯性能和小车速度这两个方面上起了很大的影响。

## 2.3 前轮角度定位的确定

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。在实际调试中，我们发现适当增大内倾角的确可以增大转弯时车轮和地面的接触面积，从而增大车了地面的摩擦程度，使车转向更灵活，减小因摩擦不够而引起的转向不足的情况

### 2.3.1 主销后倾角

主销后倾角是指主销轴线与地面垂直线在汽车纵向平面内的夹角主销后倾的作用是在车轮偏转后会产生一回正力矩 矫正车轮的偏转。后倾角越大 车速越高 车轮偏转后自动回正能力越强。但回正力过大 将会引起前轮回正过猛

加速前轮摆振 并导致转向沉重。通常后倾角值应设定在  $1^{\circ}$  到  $3^{\circ}$  。

### 2.3.2 主销内倾角

主销内倾角是指主销在汽车的横向平面内向倾斜一个角度 即主销轴线与地面垂直线在汽车横向断面内的夹角。主销内倾角也有使车轮自动回正的作用。当转向轮在外力作用下发生偏转时 由于主销内倾的原因 车轮连同整个汽车的前部将被抬起一定高度 当外力消失后 车轮就会在重力作用下恢复到原来的中间位置。另外 主销内倾还会使主销轴线延长线与路面的交点到车轮中心平面的距离减小 同时转向时路面作用在转向轮上的阻力力矩也会减小 从而可使转向操纵轻便 同时也减小了由于路面不平而从转向轮输出的力反馈。但其值不宜过小 即主销内倾角不宜过大 否则在转向时车轮主销偏转的过程中轮胎与路面将产生较大的滑动 从而增加轮胎与路面间的摩擦阻力 不仅会使转向变得沉重 还将加速轮胎的磨损。通常汽车主销内倾角不大于  $8^{\circ}$  。

### 2.3.3 前轮的外倾角

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称为“前轮外倾角”。轮胎呈现“八”字形张开时称为“负外倾” 而呈现“V”字形张开时称为“正外倾”。前轮外倾角一方面可以在汽车重载时减小或消除主销与衬套 轮毂与轴承等处的装配间隙 使车轮接近垂直路面滚动而滑动 同时减小转向阻力 使汽车转向轻便 另一方面还可以防止由于路面对车轮垂直反作用力的轴向分力压向轮毂外端的轴承 减小轴承及其锁紧螺母的载荷 从而增加这些零件的使用寿命 提高汽车的安全性。一般前轮外倾角为  $1^{\circ}$  左右。

### 2.3.4 前轮前束

当车轮有了外倾角后 在滚动时就类似于圆锥滚动 从而导致两侧车轮向外滚开。由于转向横拉杆和车桥的约束使车轮不可能向外滚开 车轮将在地面上出现边滚边向内滑移的现象 从而增加了轮胎的磨损。在安装车轮时 为消除车轮外倾带来的这种不良后果 可以使汽车两前轮的中心面不平行 并使两轮前边缘距离小于后边缘距离像内八字样前端小后端大的称为“前束” 而像外八字一样后端小前端大的称为“后束”或“负前束”。

## 2.4 舵机的安装

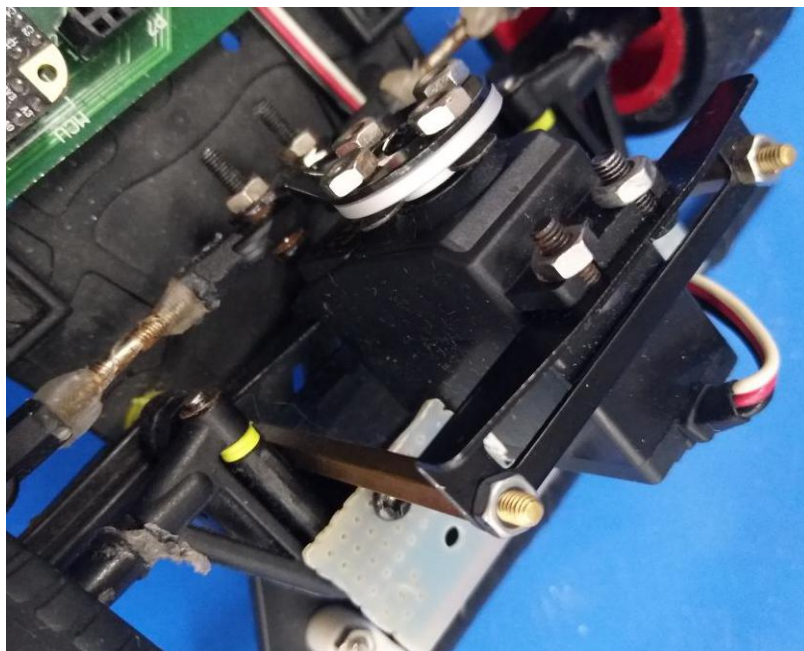


图 2.3 舵机

舵机对于整辆车的作用十分巨大，它起着对方向的控制作用，在整辆车的工作过程中，舵机的灵敏度和打角影响着整个行进过程。在安装过程中，我们为了主板的方便安装以及车模的转向性能，对舵机的安装结构进行了较大的调整。车模的转向是通过舵机带动左右横拉杆实现的，在舵机的转速和功率一定的情况下，想要加快转向机构的响应速度，唯一的方法就是优化舵机的安装位置及其力矩延长杆的长度。由于功率是速度与力矩的乘积函数，过分追求速度必然要损失力矩，力矩太小也会造成转向迟钝，因此我们利用实际参数进行计算最终设计出一套舵机连片，简化安装方式，通过直立式安装。

## 2.5 机械部分小结

小车机械方面的调整是相对比较繁琐的，一个设计不合理的零件往往会带来诸多的问题，要在不断地调试中找到最适合的，例如摄像头的高度调整，不同的高度看见的角度就不一样，在图像识别的代码就要随之改变；一个调整不当的前轮角度又有可能使小车跑得不流畅。机械部分在智能车比赛中已经成为

制胜的关键之一。只有有了好的机械，才能诞生一辆优秀的智能车。



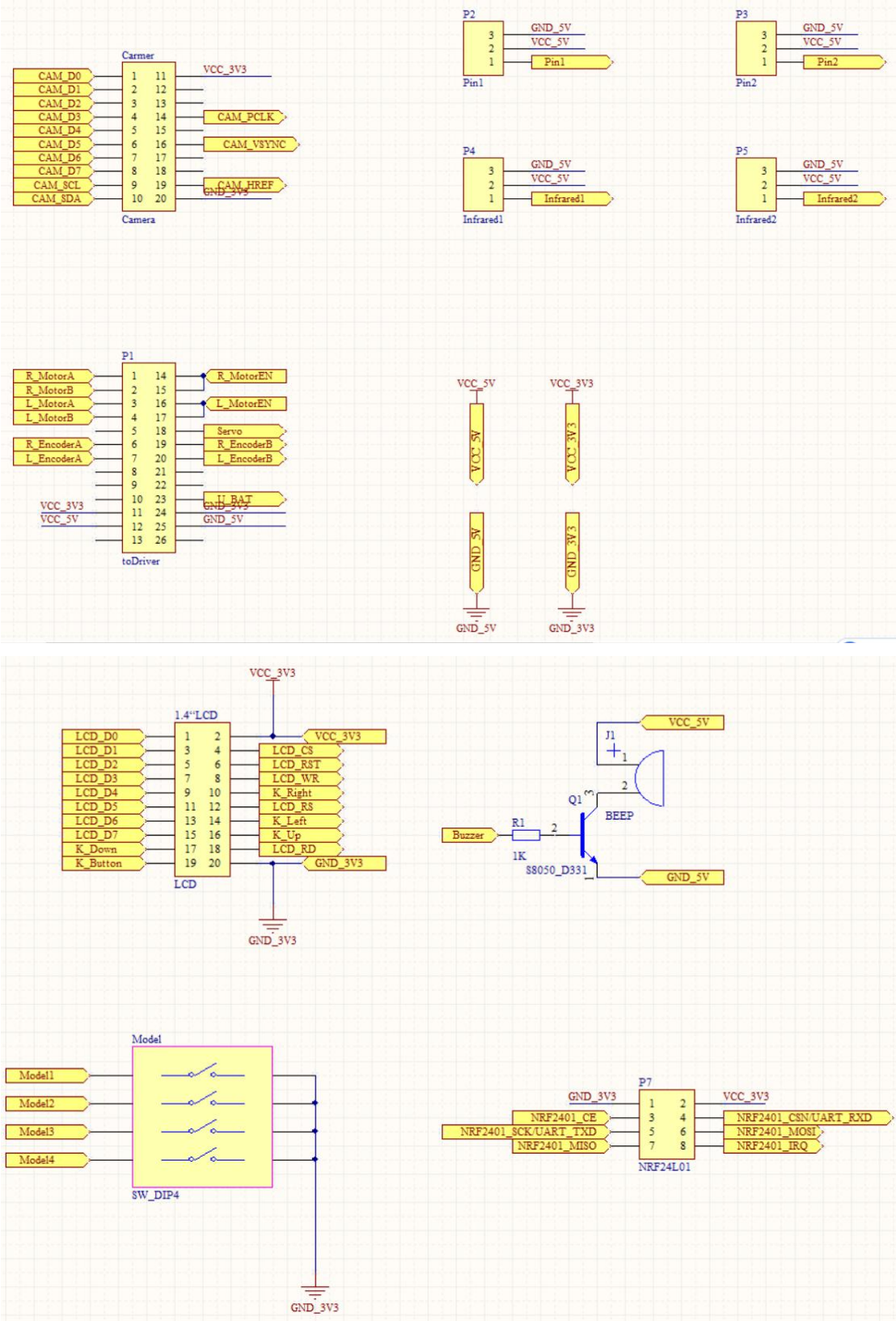
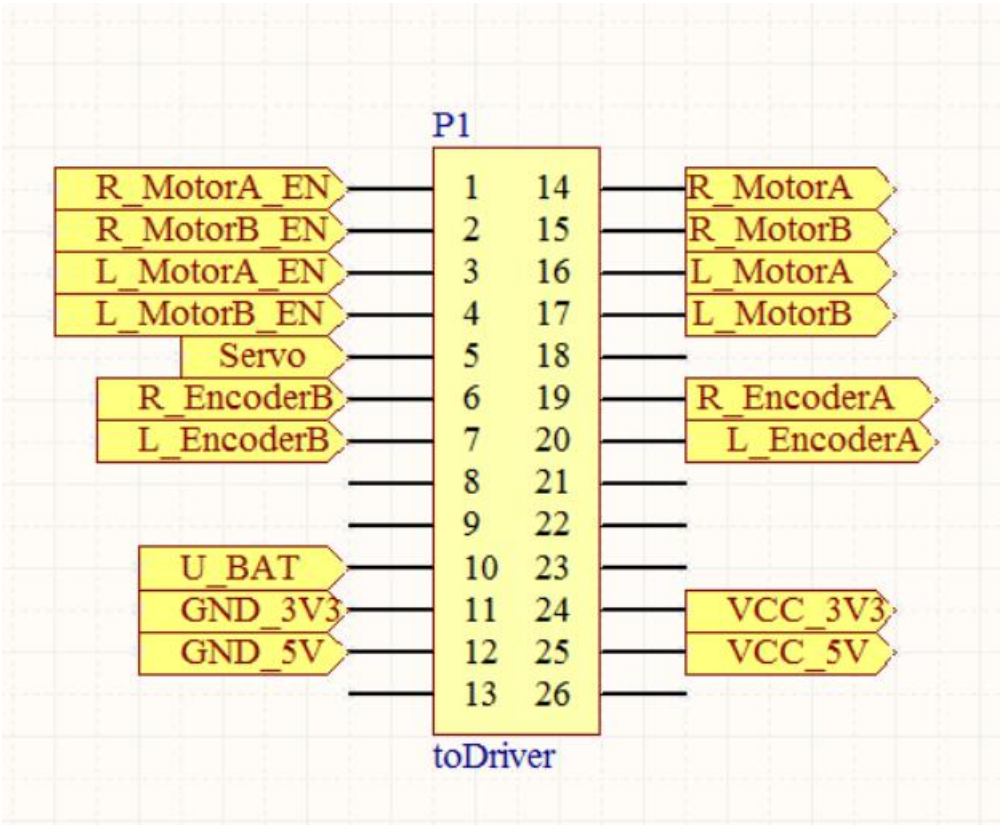


图 3.2

3.2 电机驱动电路原理图





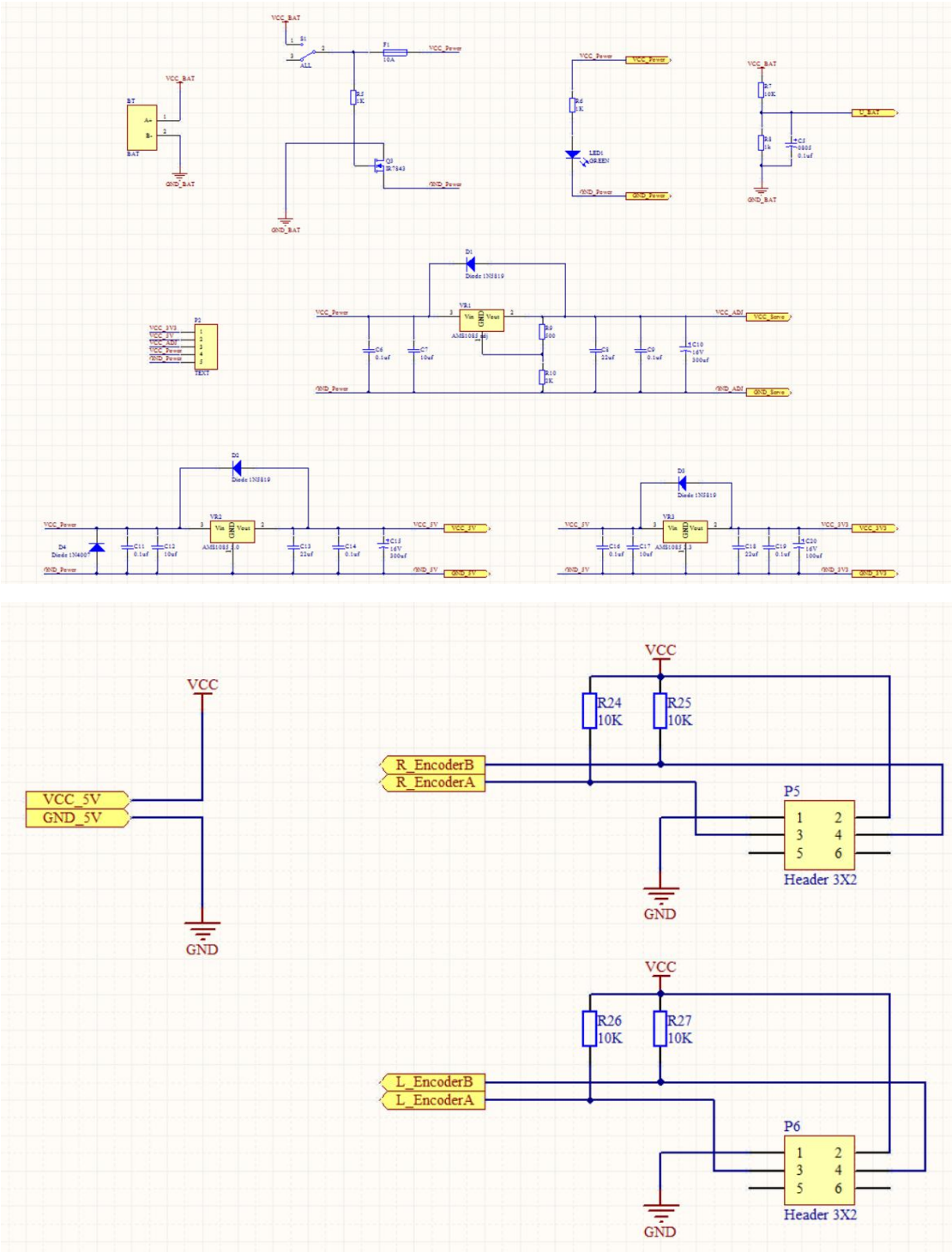


图 3.3

### 3.3 硬件电路小结

硬件电路是驱动整个智能车正常运行的根本，良好的硬件电路方案可以使



得小车在运行过程中得到最稳定的系统稳态。其中电源设计和电机驱动设计不可谓不重要。

电源是驱动小车的根本，在进行稳压电路 IC 选择时我们选择了以 LM1085—5.0 芯片为主的稳压电路设计方案输出 5V 作为为最小系统板和编码器的供电电源, 采用了以 AMS1085CD 为主的稳压设计电路为舵机提供稳定的 6V 电压，且在运行过程中，避免了舵机由于工作时会产生相当大的反向电流，提高了系统稳定性。

电机驱动部分，我们采用了比较稳定的电机驱动芯片，相较于 MOS 管驱动，虽然说明有精度较差等缺点，但贵在 BTN 驱动方案稳定，避免了重复焊接测试各种 Mos 管驱动方案的麻烦，省事省力。

## 第四章 智能车的软件设计

### 4.1 控制设计算法说明

在小车在赛道稳定的跑下来的时候，小车的差速控制、角度的控制、舵机的控制都需要算法控制来完成，在处理好摄像头拍摄的图片之后，根据图片的中线和车身的偏角控制舵机打角，再根据角度计算出小车左右轮的差速，使小车顺利通过弯道，我们的控制算法利用了好多高数、物理公式，把公式转变成代码，控制小车稳定过弯道。

#### 4.1.1 PID 算法控制

在过程控制中，按偏差的比例（P）、积分（I）和微分（D）进行控制的PID控制器（亦称PID调节器）是应用最为广泛的一种自动控制器。它具有原理简单，易于实现，适用面广，控制参数相互独立，参数的选定比较简单等优点；而且在理论上可以证明，对于过程控制的典型对象——“一阶滞后+纯滞后”与“二阶滞后+纯滞后”的控制对象，PID控制器是一种最优控制。PID调节规律是连续系统动态品质校正的一种有效方法，它的参数整定方式简便，结构改变灵活（PI、PD、…）。

PID 算法离散化：

假设采样时间间隔为  $T$ ，则在  $k$  时刻：

偏差为  $e(k)$ ；

积分为  $e(k) + e(k-1) + e(k-2) + \dots + e(0)$ ；

微分为  $(e(k) - e(k-1)) / T$ ；

从而公式离散化后如下：

比例系数： $K_p$ ，

积分系数： $K_p * T / T_i$ ，可以用  $K_i$  表示；

微分系数： $K_p * T_d / T$ ，可以用  $K_d$  表示；

则公式可以写成如下形式(公式 4-1)：

$$u(k) = K_p * e(k) + K_i * (e(k) + e(k-1) + e(k-2) + \dots + e(0)) + K_d (e(k) + e(k-1)) \quad (4-1)$$

PID 算法的离散形式就是这样了，这就是我们平时说的位置式 PID。

#### 4.1.2 舵机控制算法

舵机控制的是车模前进的方向，车模能否在赛道上走出最佳路径取决于转向舵机的正确控制。好的行车路径能够使车模行驶的时间最短，从而使时间最短，平均速度最高。

舵机控制用到了 PD 算法，偏差 error 是图像中心线与赛道中心线的差值，控制系统的输入(in)是偏差 error，输出(out)是舵机打角。

公式 4-2:

$$out = k_p * error + K_d * (error - last\_error)。$$

#### 4.1.3 速度控制算法

PID 控制在经典控制理论中技术成熟，该计数成熟可靠。相比于两位式控制，控制精度大大提高。如今，随着计算机特别是微机控制技术的发展和软件系统的灵活性，PID 控制可以得到修正而且更加完善，用计算机算法代替模拟式 PID 调节器，实现数字 PID 控制，使其控制作用更灵活、更易于改进和完善。速度控制用到了 PI 算法。

偏差 error 是设定速度与实际速度的差值，控制系统的输入(in)是偏差 error，输出(out)是 PWM。公式 4-3

$$out = k_p * error + K_i * (e(k) + e(k-1) + e(k-2) + \dots + e(0))。$$

### 4.2 图像处理算法说明

利用摄像头传感器得到图像信息，再经过二值化处理之后，再对图像进行找中线，图像是很重要的一部分，给小车找到一个很好的路线是小车稳定跑下来的前提，对不同的路况进行不同的处理，给控制算法提供必备的参数。

#### 4.2.1 路线的选择

在单片机采集图像信号后需要对其进行处理以提取主要的赛道信息，同时，由于交叉线、起点线的存在，光线、杂点、赛道连接处以及赛道外杂物的干扰，

图像效果会大打折扣。因此，在软件上必须排除干扰因素，对赛道进行有效识别，并提供尽可能多的赛道信息供决策使用。

我们利用摄像头采集回来的二值化的图像的信息，规划出来小车最终要跑的路线，就是简单的从最低行扫到白条黑或者黑跳白的点找到左右边线，根据左右边线得到最终的路线，我们没有分路况类型按照，统一的扫线方式，遇见十字路按照异常处理，相对来说代码量比较少，对于圆环另行处理。

中线提取算法的基本思想如下：

- (1) 直接逐行扫描原始图像，根据设定的阈值提取黑白跳变点；
- (2) 黑线宽度有一个范围，在确定的黑线宽度范围内提取有效黑块，这样可以滤除不在宽度范围内的黑点干扰；
- (3) 利用黑线的连续性，根据上一行黑线中心的位置来确定本行求出的黑线中心是否有效；
- (4) 求黑线中心时，因为近处的黑线稳定，远处黑线不稳定，所以采用由近及远的办法；
- (5) 图像数据量大，全部扫描一遍会浪费很多时间，利用前面已经求出的黑线中心位置判断出黑线的趋势，从而推断出下一行的黑线大概位置，确定出扫描范围，避免整行逐点扫描，节省时间；
- (6) 最终规划出来一条连续的路线；

#### 4.2.2 圆环的处理

对于今年新加的元素——圆环，我们把它另外处理了，在图像中出现黑色区域，并且；有一段连续的白色区域作为圆环的识别标志，因为摄像头的原因看不见圆环的出口，就强制左拐或者右拐，在模式中加上一左右模式，在出口的时候也进行强制补线，在十字路口车身不正也会当成圆环出口，这样对十字路就会有干扰，在做处理圆环出口补线的时候加上了条件这样就不干扰了，

```
for(H=88;H>30;H--)  
{  
    if(Image_Data[H][W_MID]==WHITE_VAL&&Image_Data[H-
```

```

1][W_MID]==BLACK_VAL)           //记录图像中心线有没有跳变

    {
        num=1;

        H_circle=120-(H-1);

        break;

    }

}

if(num==1)

{

    for(H=H_END;H>=120-H_circle;H--)

    {

        if(Image_Data[H][0]==WHITE_VAL)

            countL++;

        if(Image_Data[H][159]==WHITE_VAL)

            countR++;

    }

}

if(countL>=H_circle-30&&(countR>=H_circle-30)/*&&Image_Data[120-H_circle-3][159]==WHITE_VAL*/&&H_circle&&Image_Data[35][40]!=WHITE_VAL/*&&(!startline)*/) //H_circle&&Image_Data[35][40]!=WHITE_VAL: 在十字路处如果车身比较斜，十字路会与圆环出口差不多，所以加上这个这个条件

{

```

```
        if(direction)
        {
            for(H=H_END;H>=H_START;H--)
//圆环补右线，即圆环左转
            {
                HBoundR[H]=160-(H-(120-
H_circle))*160/(double)(H_circle);

                HBoundL[H]=159;

                if(H<(110-H_circle))
                {
                    HBoundR[H]=159;
                }

                HBoundM[H]=(HBoundL[H]+HBoundR[H])/2;
            }
        }
        else
            for(H=H_END;H>=H_START;H--)
//圆环补左线
            {
                HBoundL[H]=(H-(120-H_circle))*160/(double)(H_circle);

                HBoundR[H]=0;

                if(H<(110-H_circle))
                {
                    HBoundL[H]=0;
                }
            }
    }
```

$$\text{HBoundM}[H] = (\text{HBoundL}[H] + \text{HBoundR}[H]) / 2;$$

$$\}$$

$$\}$$

### 4.3 开发环境

软件开发环境：AR Embedded Workbench for ARM（下面简称 IAR EWARM），嵌入式 IAR Embedded Workbench 适用于大量 8 位、16 位以及 32 位的目前 IAREWARM 支持 ARM Cortex-M4 内核的最新版本是 6.3 版本。IAR EWARM 中包含一个全软件的模拟程序（simulator）。用户不需要任何硬件支持就可以模拟各种 ARM 内核、外部设备甚至中断的软件运行环境。从中可以了解和评估 IAR EWARM 的功能和使用方法。IAR EWARM 的主要特点如下：高度优化的 IAR ARM C/C++ Compiler、IARARMAsembler、一个通用的 IAR XLINK Linker、IAR XAR 和 XLIB 建库程序和 IAR DLIB C/C++ 运行库、功能强大的编辑器、项目管理器、命令行实用程序、IAR C-SPY 调试器（高级的高级语言调试器）、用 IAR 环境开发 Kinetis 是非常的方便的。图形界面如下：

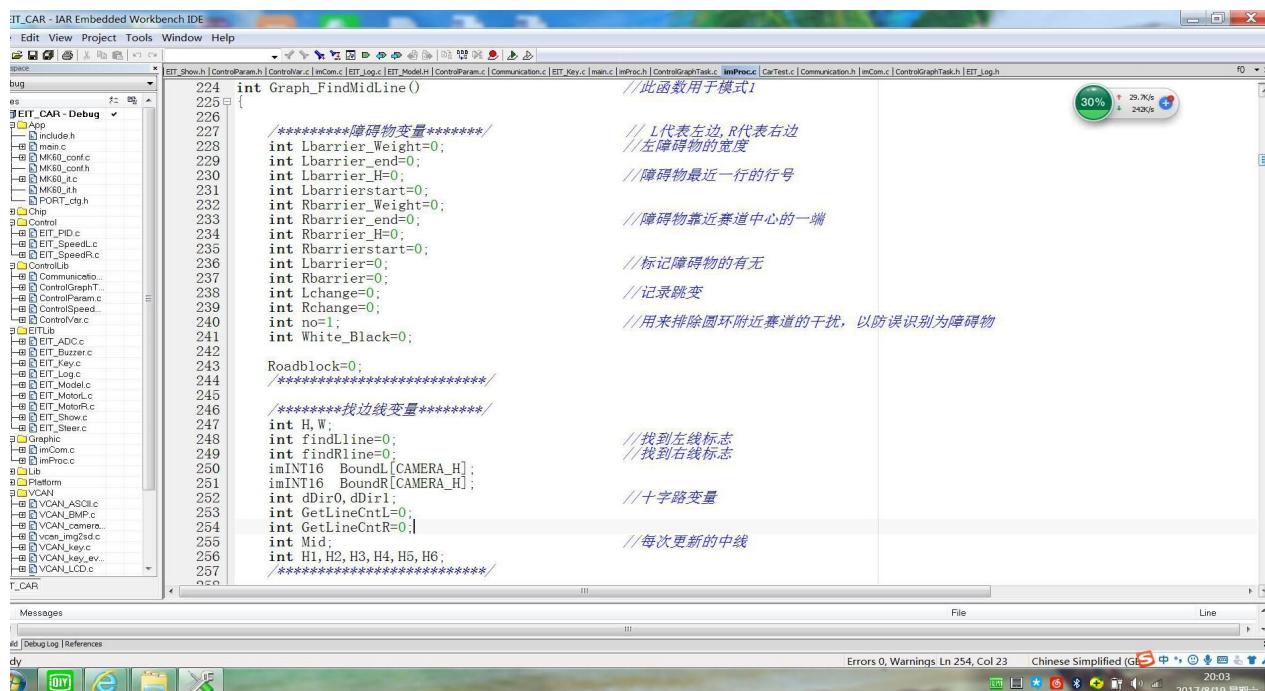


图 4.1

硬件开发环境：Altium Designer Winter 09 的板级设计功能可完全定义并实现设计中的物理元素。强大的规则驱动设计、版图和编辑环境可在直观高效的环境中完全控制设计中的各个方面：altium designer winter 管理库：Altium Designer Winter 09 提供综合的器件数据管理和库资源，无论您组织的大小或需求是什么，均可控制部件的使用；altium designer winter 设计制造：Altium Designer Winter 09 起到设计和制造间的桥梁作用，可管理所有制造数据的生成和验证；ltium designer winter 利用可编程器件 Altium Designer Winter 09 可充分利用了当今大容量可编程器件在设计中的潜力。您可在易于升级的[软件](#)领域中工作，令修改硬件与软件同样方便；altium designer winterFPGA/PCB 集成：可编程器件能否顺利集成到物理设计流程对设计生产率十分关键。Altium Designer Winter 09 提供 FPGA 设计项目的[无缝链接](#)，通过[板卡](#)设计把它们联系在一起；altium designer winter 管理电子开发流程：电子产品开发不再是独立的流程。Altium Designer Winter 09 统一了整个设计流程，可在单一、集成的设计流环境中管理开发的所有方面。图像界面如下：



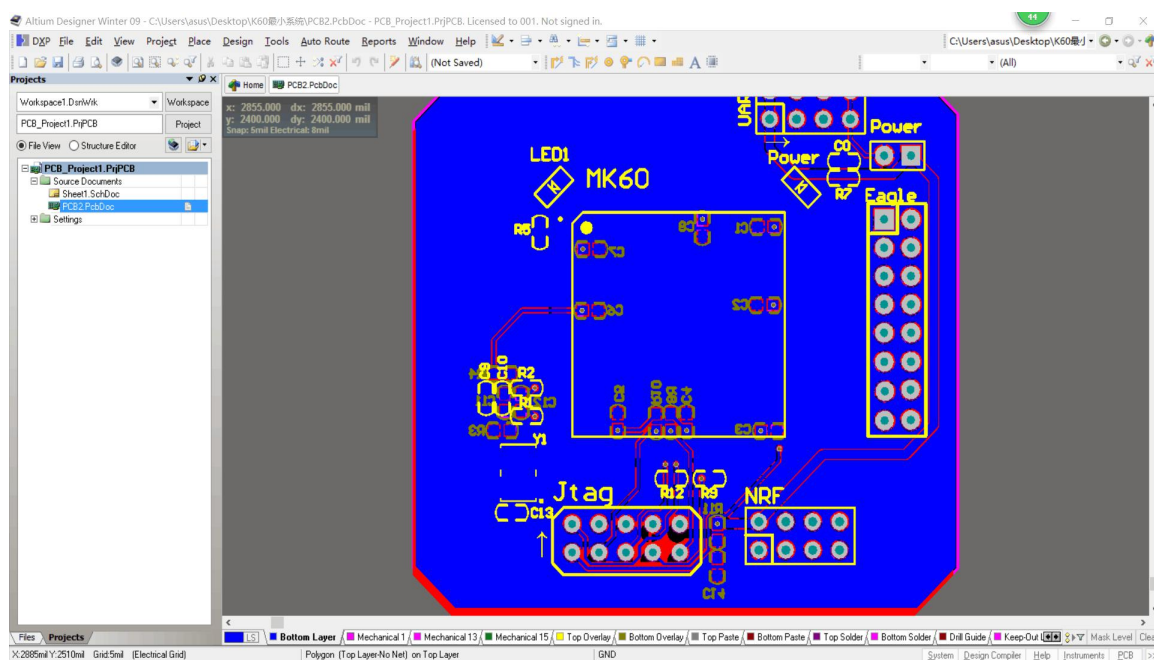


图 4.2

## 4.4 软件部分小结

软件部分对小车非常重要，首先要选择一个很好的路线，才能保证小车不会出界按照规定行驶，在小车控制算法上，对小车方向控制，小车的差速，对小车稳定的跑下来整个赛道起很关键的作用，软件控制的好，小车的平稳度才会高，在整个比赛中才会有很大的把握。

## 结论

这已是第十二届“恩智浦”杯全国大学生智能汽车竞赛了，其比赛内容涵盖了控制、模式识别、传感技术、电子、电气、计算机、机械等多个学科，培养了学生的知识融合和实践动手能力。整份技术报告我们介绍了准备比赛时的基本流程跟思路以及所需的准备，主要包括车体的机械机构、电路设计、以及软件程序的调政策略。对于初次准备智能车竞赛，由于自身知识积累不够，我们遇到很多困难，也犯了不少错误，付出了很大的成本。但是，通过不断地积累和查阅资料以及跟同学和老师的交流逐个将问题解决。从前期杂乱的机械结构到最终紧凑的模块整合，从起初频繁的系统复位、严重跳轮到最终的稳定运行提速冲刺，等等很多问题让我们感受颇深。从一开始速度快了过圆环会出界，过大环比较稳定，小环就直冲，过小圆环比较稳定，而过大圆环一直转圈，到后来的问题全部解决(在过圆环的时候检测出来就减速，在圆环里面以慢速通过，这样在小环里面就可以稳定的通过，在大圆环问题上改了判断补线的参数就可以稳定的出去了)，可是，这是一项比赛，更是一项挑战，唯有不断遇到问题，不断解决问题，才能真正从中收获知识和技能，这也正是比赛的真正意义所在。在即将进入智能时代的前夕，智能车也将有着很大的发展前景，通过参加“恩智浦”杯智能车竞赛我们对此兴趣更加浓厚，在未来的学习与实践中，将继续朝着这个方向深入学习。从最初的开始接触，到如今即将展示成果，回首过去，或许，这便是我们付出的回报，或许，这便是进步。最后，也希望我们能在这次恩智浦杯智能车竞赛中获得满意的成绩！

## 车模技术参数

车长	289mm
车宽	162mm
车高	313mm
车重	1030g
微处理器型号	MK60FN1M0VLQ15
传感器	摄像头
	编码器
舵机	FutabaS3010 一个
电路消耗	260w
赛道信息检验精度、频率	2/12.5mm 50HZ

## 参考文献

- [1] 冯冲, 段晓敏. 飞思卡尔 MC9S12(X) 开发必修课[M]. 北京. 北京航空航天大学出版社. 2014
- [2] 陈宋, 李立国, 黄开胜. 智能模型车底盘浅析[J]. 清华大学汽车工程系
- [3] 卓晴, 黄开胜, 邵贝贝. 学做智能车. 北京航空航天大学出版社[4] 张凯威、刘鹏、齐顺. 第八届全国大学生“飞思卡尔”杯智能汽车竞赛国防科技大学摄像铁军一师技术报告[R]. 国防科技学. 201
- [5]. 关文达. 汽车构造 (第 3 版). 北京. 机械工业出版社. 2011
- [6]. 刘莹, 吴宗泽. 机械设计教程 (第 2 版). 北京. 机械工业出版社. 20

## 附录 A

图像源代码:

```
#include "imProc.h"
#include "imCom.h"
#include "ControlParam.h"
#ifdef MK60F15
    #include "mex.h"
#endif
extern EIT_PARAM gParam;
extern imUINT8 Image_Data[CAMERA_H][CAMERA_W];
imUINT8 Image_DataF[CAMERA_H][CAMERA_W];
imINT32 gDir_Near=0;
imINT32 gDir_Mid =0;
imINT32 gDir_Far =0;
imINT16 HBoundL[CAMERA_H];
imINT16 HBoundR[CAMERA_H];
imINT16 HBoundM[CAMERA_H];
imINT16 HBoundM_F[CAMERA_H];
imINT16 HBoundM_REAL[CAM_MAX_LENGTH_CM+1];
imINT32 H_Min=H_START;
imINT32 H_MaxL=H_END;
imINT32 H_MaxR=H_END;
imINT32 D_Max =CAM_MAX_LENGTH_CM;

int MaxWhite=0;
int Out=0;
int MaxStart=W_START;
int MaxEnd=W_START;

int S=0;                //S 路标志位

int pixel_change=0;     //像素点突变次数
int num=0;               //用于初步判断是否是圆环
int Circle=0;
```

## 十二届全国大学生智能汽车邀请赛技术报告

```
int judge_startline1(); //新起跑线, 用于在起跑线处误识别
为圆环 或者十字路
void Graph_ExpandImag(unsigned char* src,int width, int height,int
strutWidth, int structHeigh);
int Graph_JudgeOut(void)
{
    static int LeftOut=0;

    int WhiteCnt;
    int W;
    int wStart,wEnd;

    MaxWhite=0;
    WhiteCnt=0;
    wStart=W_START;
    wEnd=W_START;

    pixel_change=0;
    for(W=W_START;W<W_END;W++)
    {

        if((Image_Data[H_END][W]>Image_Data[H_END][W+1] || Image_Data[H_END][W]<Image_Data[H_END][W+1])&&(W+1<=W_END)) //白变黑或者黑跳白
        {
            pixel_change++;
        }

        if(Image_Data[H_END][W]==WHITE_VAL&&Image_Data[H_END][W+1]==WHITE_VAL)
            WhiteCnt++;
        else
        {
            if(WhiteCnt>MaxWhite)
            {
                wEnd=W;
            }
        }
    }
}
```

```

        MaxWhite=WhiteCnt;
        MaxStart=wStart;
        MaxEnd=wEnd;
    }
    WhiteCnt=0;
    wStart=W;
}
}
if(WhiteCnt>MaxWhite)
{
    wEnd=W;
    MaxWhite=WhiteCnt;
    MaxStart=wStart;
    MaxEnd=wEnd;
}

switch(Out)
{
    case 0:
        if(MaxWhite<=40)
        {
            if(MaxStart<=W_START)
            {
                Out=1;
                LeftOut=1;
            }
            else if (MaxEnd>=W_END)
            {
                Out=1;
                LeftOut=0;
            }
        }
        break;
    case 1:
        if(MaxWhite>=70)
        {

```

## 第十二届全国大学生智能汽车邀请赛技术报告

```
        if(LeftOut==1 && (MaxStart<=W_START))
            Out=0;
        if(LeftOut==0 && (MaxEnd>=W_END))
            Out=0;
    }
    break;
}
return Out;
}

//腐蚀
void MorphErosion(unsigned char* src, unsigned char* dst, int width, int
height, int strutWidth, int structHeight)
{
    int i, j, m, n, mid;
    unsigned char val;
    if (width - strutWidth < 0 && height - structHeight < 0)return;
    if (strutWidth != structHeight)return;

    mid = (strutWidth + 1) / 2 - 1;
    for (i = mid; i < height - mid; i++)
    {
        for (j = mid; j < width - mid; j++)
        {
            val = WHITE_VAL;
            for (m = -mid; m <= mid; m++)
            {
                for (n = -mid; n < mid; n++)
                {
                    if (m == 0 && n == 0 )continue;
                    val &=  src[(i+m) * width + j + n];
                    if(!val)break;
                }
                if(!val)break;
            }
            dst[i * width + j] = val;
        }
    }
}
```



```

    }
}
}
//膨胀
void MorphDilation(unsigned char* src, unsigned char* dst, int width, int
height, int strutWidth, int structHeight)
{
    int i, j, m, n, mid;
    unsigned char val = 0;
    if (width - strutWidth < 0 && height - structHeight < 0) return;
    if (strutWidth != structHeight) return;

    mid = (strutWidth + 1) / 2 - 1;
    for (i = mid; i < height - mid; i++)
    {
        for (j = mid; j < width - mid; j++)
        {
            val = 0;
            for (m = -mid; m < mid; m++)
            {
                for (n = -mid; n < mid; n++)
                {
                    if (m == 0 && n == 0) continue;
                    val |= src[(i+m) * width + j + n];
                    if(val) break;
                }
                if(val) break;
            }

            dst[i * width + j] = val;
        }
    }
}
//先腐蚀后膨胀
void MorphOpen(unsigned char* src, unsigned char* tmp, int width, int
height, int strutWidth, int structHeight)

```

## 十二届全国大学生智能汽车邀请赛技术报告

```
{
    MorphErosion(src, tmp, width, height, strutWidth, structHeight);
    MorphDilation(tmp, src, width, height, strutWidth, structHeight);
    Graph_ExpandImag(src, width, height, strutWidth, structHeight);
}

void MorphClose(unsigned char* src, unsigned char* tmp, int width, int
height, int strutWidth, int structHeight)
{
    MorphDilation(src, tmp, width, height, strutWidth, structHeight);
    Graph_ExpandImag(tmp, width, height, strutWidth, structHeight);
    MorphErosion(tmp, src, width, height, strutWidth, structHeight);
    Graph_ExpandImag(src, width, height, strutWidth, structHeight);

}

void Graph_ExpandImag(unsigned char* src, int width, int height, int
strutWidth, int structHeight)
{
    int i, j, mid;
    mid = (strutWidth + 1) / 2 - 1;
    for (i = 0; i < mid+2; i++)
    {
        for (j = 0; j < width ; j++)
        {
            src[i*width+j]=src[(mid+2)*width+j];
        }
    }
    for (i = height-1; i > height-mid-2; i--)
    {
        for (j = 0; j < width ; j++)
        {
            src[i*width+j]=src[(height-mid-2)*width+j];
        }
    }

    for (j = 0; j < mid+1; j++)
    {
        for (i = 0; i < height ; i++)
```

```

        {
            src[i*width+j]=src[mid*width+mid+2];
        }
    }
    for (j = width-1; j > width-mid-1; j--)
    {
        for (i = 0; i < height ; i++)
        {
            src[i*width+j]=src[i*width+width-mid-2];
        }
    }
}

int Graph_FindMidLine()                                //此函数用于模式 1
{
    /*******障碍物变量*****/                          // L 代表左边, R 代表右边
    int Lbarrier_Weight=0;                               //左障碍物的宽度
    int Lbarrier_end=0;
    int Lbarrier_H=0;                                    //障碍物最近一行的行号
    int Lbarrierstart=0;
    int Rbarrier_Weight=0;
    int Rbarrier_end=0;                                  //障碍物靠近赛道中心的一端
    int Rbarrier_H=0;
    int Rbarrierstart=0;
    int Lbarrier=0;                                       //标记障碍物的有无
    int Rbarrier=0;
    int Lchange=0;                                         //记录跳变
    int Rchange=0;
    int no=1;                                              //用来排除圆环附近赛道的干
    扰, 以防误识别为障碍物
    int White_Black=0;
    /*******找边线变量*****/
    int H, W;

```

## 第十二届全国大学生智能汽车邀请赛技术报告

```
int findLline=0;           //找到左线标志
int findRline=0;           //找到右线标志
imINT16 BoundL[CAMERA_H];
imINT16 BoundR[CAMERA_H];
int dDir0,dDir1;           //十字路口变量
int GetLineCntL=0;
int GetLineCntR=0;
int Mid;                   //每次更新的中线
int H1,H2,H3,H4,H5,H6;
/*****/

/*****圆环变量*****/
// int num=0;               //用于初步判断是否是圆环
int H_circle=0;            //代表圆环近端的H
int countL=0,countR=0;
int direction=0;           //车行驶方向控制： 0:圆环左
转 1:圆环右转
int pixel_change1=0;
int lastL=0,nowL=0,lastR=0,nowR=0;
int no_getLineL=0,no_getLineR=0;
/*****/

/*****起始线变量*****/
int startline=0;           //用来判断是否是起跑线
/*****/

/*****S 路变量*****/
int White_pixel=0;         //图像中心白像素点个数
int sum;                   //60-120 行的中线之和
int S_Dir;                 //60-120 行的偏差均值
/*****/

H_Min = H_START;

/*****判断是否出界*****/
if(Graph_JudgeOut()&&(!(pixel_change>6)))
```

```

        return 0;
    else
    {
        for (H=0;H<=CAMERA_H;H++)
        {
            HBoundL[H]=MaxStart;
            HBoundR[H]=MaxEnd;
            HBoundM[H]=(MaxStart+MaxEnd)/2;
        }
        // H_Min = H_START;                //这个语句没作用，只是为了
        了程序的正确执行，因为如果 else 里没有语句，遇见圆环会死机
    }
    /*****/

    dDir0=0;//根据偏差算十字拐角
    dDir1=0;//根据偏差算十字拐角

    num=0;
    int point_A=0,point_B=0,point_C=0,point_D=0;        //圆环 上 下 左 右
    四个点
    int center=0;                //圆环中心
    DisBuzzer();
    Circle=0;
    for (H=110;H>30;H--)
    {
        if (Image_Data[H][W_MID]==WHITE_VAL&&Image_Data[H-
1][W_MID]==BLACK_VAL&&H_circle==0)        //记录图像中心线有没有跳变
        {
            num=1;
            H_circle=120-(H-1);
            //break;
        }
        if (H>10&&H<60)
        {
            if (Image_Data[H][W_MID]==WHITE_VAL&&Image_Data[H-
1][W_MID]==BLACK_VAL&&point_A==0)

```

## 十二届全国大学生智能汽车邀请赛技术报告

```
        {
            point_A=H-1;
            //EnBuzzer();
        }

if(Image_Data[H][W_MID]==WHITE_VAL&&Image_Data[H+1][W_MID]==BLACK_VAL&&point_B==0)
{
    point_B=H+1;
    //EnBuzzer();
}
if(point_A!=0&&point_B!=0)
    break;
}
}
if(point_A-point_B>15&&(point_A>25&&point_B<25))
{
    center=(point_A+point_B)/2;
}

S=0;
//DisBuzzer();
for(H=H_END;H>=H_START;H--)
{
    /*****S 路识别*****/
    if(Image_Data[H][79]==WHITE_VAL)
        White_pixel++;
    /******/

    H1=int_min(H+1,H_END);
    H2=int_min(H+2,H_END);
    H3=int_min(H+3,H_END);
    H4=int_min(H+4,H_END);
    H5=int_min(H+5,H_END);
    H6=int_min(H+6,H_END);
    Mid=HBoundM[H+1];
```

```

        if(Image_Data[H][Mid]!=WHITE_VAL || Image_Data[H][Mid-
1]!=WHITE_VAL || Image_Data[H][Mid+1]!=WHITE_VAL)
        {
            Mid = W_START+(W_END>>2);
            if(Image_Data[H][Mid]!=WHITE_VAL || Image_Data[H][Mid-
1]!=WHITE_VAL || Image_Data[H][Mid+1]!=WHITE_VAL)
            {
                Mid = W_END-(W_END>>2);
                if(Image_Data[H][Mid]!=WHITE_VAL || Image_Data[H][Mid-
1]!=WHITE_VAL || Image_Data[H][Mid+1]!=WHITE_VAL)
                {
                    Mid = W_START+(W_END>>4);
                    if(Image_Data[H][Mid]!=WHITE_VAL ||
Image_Data[H][Mid-1]!=WHITE_VAL || Image_Data[H][Mid+1]!=WHITE_VAL)
                    {
                        Mid = W_END-(W_END>>4);
                        if(Image_Data[H][Mid]!=WHITE_VAL ||
Image_Data[H][Mid-1]!=WHITE_VAL || Image_Data[H][Mid+1]!=WHITE_VAL)
                        {
                            HBoundL[H] = HBoundL[H+1] ;
                            HBoundR[H] = HBoundR[H+1] ;
                            HBoundM[H] = HBoundM[H+1] ;
                            continue;
                        }
                    }
                }
            }
        }

    }

    }

/*****找左线*****/
for(W=W_START;W<Mid;W++)
{
    if (Image_Data[H][W]<Image_Data[H][W+1])
    {

```

## 十二届全国大学生智能汽车邀请赛技术报告

```

        HBoundL[H] = W;
//        if(int_abs(HBoundL[H]-HBoundL[H1])>5)
//            HBoundL[H]=HBoundL[H1];
        GetLineCntL++;
        break;
    }
}
if(W==Mid)
{
    lastL=nowL;
    nowL=1;
    if((H>121-H_circle)&&(H<121-
H_circle+20))&&(lastL==1&&nowL==1)&&num==1)
        no_getLineL++;

    HBoundL[H] =HBoundL[H1] ;
}

/*****/

/*****找右线*****/
for(W=W_END;W>Mid;W--)
{
    if (Image_Data[H][W]<Image_Data[H][W-1])
    {

        HBoundR[H] = W;
//        if(int_abs(HBoundR[H]-HBoundR[H1])>5)
//            HBoundR[H]=HBoundR[H1];
        GetLineCntR++;
        break;
    }
}
if(W==Mid)
{
    lastR=nowR;
```



```

        nowR=1;
        if(((H>121-H_circle)&&(H<121-
H_circle+20))&&(lastR==1&&nowR==1)&&num==1)
            no_getLineR++;

        HBoundR[H] =HBoundR[H1] ;
    }

    if(Image_Data[center][W_MID]==BLACK_VAL&&Image_Data[center][W_MID-
1]==BLACK_VAL&&Image_Data[center][W_MID+1]==BLACK_VAL&&Image_Data[center][
W_MID-
5]==BLACK_VAL&&Image_Data[center][W_MID+5]==BLACK_VAL&&Image_Data[center][
W_MID+15]==BLACK_VAL&&Image_Data[center][W_MID-
15]==BLACK_VAL&&Image_Data[center][25]==WHITE_VAL&&Image_Data[center][135]
==WHITE_VAL&&(no_getLineR>10)&&(no_getLineL>10))
    {
        EnBuzzer();
        Circle=1;
    }
    /*****/

    /*****十字路口*****/
    dDir0=HBoundR[H]-HBoundR[H2];
    dDir1=HBoundR[H2]-HBoundR[H6];

    if((dDir0*dDir1<-4 && dDir0>0)/*&&(!startline) */|| ((dDir0 > 10
&& dDir1<=0)/*&&(!startline)*//(!startline):在起跑线处会误
识别为十字路，所以加上(!startline)
    {
        //十字路补右线
        HBoundR[H]=HBoundR[H3]+HBoundR[H3]-HBoundR[H6];
        HBoundR[H1]=HBoundR[H4]+HBoundR[H3]-HBoundR[H6];
    }

    dDir0=HBoundL[H]-HBoundL[H2];

```

## 十二届全国大学生智能汽车邀请赛技术报告

```
dDir1=HBoundL[H2]-HBoundL[H6];

if((dDir0*dDir1<-4 && dDir0<0)/*&&(!startline)* / || ((dDir0 <-10
&& dDir1>=0)/*&&(!startline)* /))
{
    //十字路补左线
    HBoundL[H]=HBoundL[H3]+HBoundL[H3]-HBoundL[H6];
    HBoundL[H1]=HBoundL[H4]+HBoundL[H3]-HBoundL[H6];
}

HBoundM[H]= (HBoundR[H]+ HBoundL[H])/2;
if( HBoundR[H] < W_START+LINE_WIDTH || HBoundL[H] > W_END-
LINE_WIDTH || (HBoundR[H]-HBoundL[H]) < LINE_WIDTH )
{
    H_Min = H;
    break;
}

/*****/

}

/****S 路识别****/
sum=0;
for(H=H_END;H>60;H--)
{
    sum+=HBoundM[H];
}
S_Dir=sum/(H_END-60)-79;

if(White_pixel>100&&((S_Dir<25&&S_Dir>1) || (S_Dir>-25&&S_Dir<-
1))&&(int_abs((HBoundM[60]+HBoundM[H_END])/2-HBoundM[90])>5))
{
    S=1; //S 路标志位
    //EnBuzzer(); //蜂鸣器响
}

/*****/
```

```

/*****圆环*****/

int countbl=0, countbr=0;
for(W=W_START;W<W_END;W++)
{
    if((Image_Data[121-H_circle-2][W]>Image_Data[121-H_circle-
2][W+1]||Image_Data[121-H_circle-2][W]<Image_Data[121-H_circle-
2][W+1])&&(W+1<=W_END)) //白变黑或者黑跳白
    {
        pixel_changel++; //像素点突变次数
    }
}

if(num==1&&(pixel_changel<6))
{
    for(H=120-H_circle;H<=H_END;H++)
    {
        if(Image_Data[H][0]==WHITE_VAL)
        {
            countL++;
//woxiedelalalallalla
            if(countL>=H_circle-30)
                break;
        }
    }
    else
    {
        countbl++;
        if(countbl>4)
        {
            countL=0;
            countbl=0;
            break;
        }
    }
}
}

```

## 十二届全国大学生智能汽车邀请赛技术报告

```
        for(H=H_END;H>=120-H_circle;H--)
        {
            if(Image_Data[H][159]==WHITE_VAL)
            {
                countR++;
//woxiedelalalallallalla
                if(countR>=H_circle-30)
                    break;
            }
            else
            {
                countbr++;
                if(countbr>4)
                {
                    countR=0;
                    countbr=0;
                    break;
                }
            }
        }
    }
    //Circle=0;
    //    int a,b,c,d;
    //    for(H=H_END;H>0;H--)
    //    {
    //
    //
    //        if(Image_Data[H][W_MID]==WHITE_VAL&&Image_Data[H-
    //1][W_MID]==BLACK_VAL)        //记录图像中心线有没有跳变
    //        {
    //            a=H-1;
    //            break;
    //        }
    //    }
    //    for(H=a;H>0;H--)
    //    {
    //        if(Image_Data[H][W_MID]==BLACK_VAL&&Image_Data[H-
```

```

1][W_MID]==WHITE_VAL)
//      {
//      b=H;
//      break;
//      }
//  }
//      for(H=b;H>0;H--)
//      {
//          if(Image_Data[H][W_MID]==WHITE_VAL&&Image_Data[H-
1][W_MID]==BLACK_VAL)
//          {
//              d=H;
//              break;
//          }
//      }
//      c=a-b;
//if((countL>=H_circle-30&&(countR>=H_circle-30)/*&&Image_Data[120-
H_circle-
3][159]==WHITE_VAL*/&&H_circle&&Image_Data[35][40]!=WHITE_VAL&&(GetLineCnt
R<H_circle))||((d>5)&&(c>40)&&(countL>=30)&&(countR>=30)&&H_circle&&Image_
Data[35][40]!=WHITE_VAL&&(GetLineCntR<H_circle))/*&&(!startline)*/)
//H_circle&&Image_Data[35][40]!=WHITE_VAL:在十字路处如果车身比较斜，十字路
会与圆环出口差不多，所以加上这个这个条件
    if(Circle==1||((H>30&&H<=80)&&(countL>=H_circle-
30)&&(countR>=H_circle-
30)&&H_circle&&Image_Data[35][40]!=WHITE_VAL&&(GetLineCntR<H_circle))||((H
>30&&H<=110)&&(countL>=H_circle-5)&&(countR>=H_circle-
5)&&H_circle&&Image_Data[35][40]!=WHITE_VAL)|| (no_getLineL>10&&no_getLineR
>10&&Image_Data[25][55]!=WHITE_VAL))
//H_circle&&Image_Data[35][40]!=WHITE_VAL:在十字路处如果车身比较斜，十字路
会与圆环出口差不多，所以加上这个这个条件
    {
        Circle=1;
        if(direction)
        {
            for(H=H_END;H>=H_START;H--)

```

## 十二届全国大学生智能汽车邀请赛技术报告

```
//圆环补右线，即圆环左转
{
    HBoundR[H]=160-(H-(120-H_circle))*160/(double)(H_circle);
    HBoundL[H]=159;
    if(H<(110-H_circle))
    {
        HBoundR[H]=159;
    }
    HBoundM[H]=(HBoundL[H]+HBoundR[H])/2;
}

else
    for(H=H_END;H>=H_START;H--)
//圆环补左线
{
    HBoundL[H]=(H-(120-H_circle))*160/(double)(H_circle);
    HBoundR[H]=0;
    if(H<(110-H_circle))
    {
        HBoundL[H]=0;
    }
    HBoundM[H]=(HBoundL[H]+HBoundR[H])/2;
}

/*****/

/*****障碍物*****/
for(H=100;H>20;H--) //50
开始是 100 到 40
{
    /*****识别左障碍物*****/
    Lbarrier_Weight=0;
    Lchange=0;
//在弯道,可能上一行 Rchange=1;
```

```

        for (W=HBoundM[H];W>HBoundL[H];W--)
检测左障碍物
        {
            if (Image_Data[H][W]>Image_Data[H][W-1] || Lchange)
            {
                Lchange=1;
有白到黑的跳变
                if (Lchange==1&&Image_Data[H][W-1]==BLACK_VAL&&Image_Data[H][W-2]==BLACK_VAL)
                    Lbarrier_Weight++;
障碍物宽度
            }
            else
            {
                Lbarrier_end=W;
                Lbarrier_H=H;
                Lchange=0;
                break;
            }
        }
        Lbarrierstart=Lbarrier_end+Lbarrier_Weight;
障碍物靠近赛道中心的一端
        /*****/

        /*****识别右障碍物*****/
        Rbarrier_Weight=0;
        Rchange=0;
在弯道, 可能上一行 Rchange=1;
        for (W=HBoundM[H];W<HBoundR[H];W++)
检测左障碍物
        {
            if (Image_Data[H][W]>Image_Data[H][W+1] || Rchange)
            {
                Rchange=1;
有白到黑的跳变

```

## 第十二届全国大学生智能汽车邀请赛技术报告

```
if(Rchange==1&&Image_Data[H][W+1]==BLACK_VAL&&Image_Data[H][W+2]==BLACK_VAL)
{
    Rbarrier_Weight++;
}
障碍物宽度
else
{
    Rbarrier_end=W;
    Rbarrier_H=H;
    Rchange=0;
    break;
}
}
if(HBoundM[H]-HBoundM[H-1]>10)
{
    由于赛道环境限制，在圆环那里，周围的赛道太靠近圆环，会误识别为障碍物，所以
    这个 if 语句用来排除周围赛道的干扰
    {
        no=0;
    }
    代表不是障碍物，而是周围赛道干扰
}
Rbarrierstart=Rbarrier_end-Rbarrier_Weight;
障碍物靠近赛道中心的一端
/*****/

if(Lbarrier_Weight>20&&HBoundM[Lbarrier_H]!=BLACK_VAL)
{
    排除远处干扰    : HBoundM[Rbarrier_H]!=BLACK_VAL
    {
        Lbarrier=1;
        break;
    }
    if(Rbarrier_Weight>20&&HBoundM[Rbarrier_H]!=BLACK_VAL)
    //Lbarrier_Weight 设为 10 时，起跑线会当成障碍物，所以改为 20
    {
        Rbarrier=1;
        break;
    }
}
```



```

/*****障碍物补线*****/
if (Lbarrier==1&&no&&(Lbarrier_H>H_Min))
// (Lbarrier_H>H_Min&&(H_Min!=H_START)):排除弯道干扰, 急弯时, 会误识别为障碍物
{
    for (H=Lbarrier_H;H<H_END;H++)
    {
        HBoundL[H]=Lbarrierstart-(H-
Lbarrier_H)*Lbarrierstart/(H_END-Lbarrier_H)+10;        //障碍物补线
    }
    for (H=Lbarrier_H-1;H>=H_START;H--)
    {
        HBoundL[H]=HBoundL[H+1];
    }
}

if (Rbarrier==1&&no&&(Rbarrier_H>H_Min))
{
    for (H=Rbarrier_H;H<H_END;H++)
    {
        HBoundR[H]=Rbarrierstart+(H-Rbarrier_H)*(160-
Rbarrierstart)/(H_END-Rbarrier_H)-10;
    }
    for (H=Rbarrier_H-1;H>=H_START;H--)
    {
        HBoundR[H]=HBoundR[H+1];
    }
}

for (H=H_END;H>=H_START;H--)                                //障
碍物中线规划
{

    HBoundM[H]=(HBoundL[H]+HBoundR[H])/2;
}

```

## 十二届全国大学生智能汽车邀请赛技术报告

```
    }  
    /*****  
  
    /*for (H=int_min (H_MaxL, H_MaxR) ;H<=H_END;H++)  
    {  
        HBoundM[H]=(HBoundL[H]+HBoundR[H])/2;  
    }*/  
    return S;  
}
```