

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：安徽工业大学

队伍名称：皖 E007

参赛队员：赵庆

王杰

张松

带队教师：武卫华、乔莹

关于技术报告和研究论文使用授权的说明

本人完全了解第十一届“恩智浦”杯全国大学生智能汽车竞赛关保留、使用技术报告和研究论文的规定，即：参赛作品著作权归参赛者本人，比赛组委会和恩智浦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： 王杰

赵庆

张松

带队教师签名： 武卫华

乔莹

日 期： 2016.08.14

摘要

本文以第十一届全国大学生智能车竞赛为背景，利用 Freescale 半导体公司生产的 32 位单片机 MK60FN1M0VLQ15 和 CMOS 摄像头的配合来实现自动识别道路，让赛车可以在特定的跑道上行驶。通过对 CMOS 摄像头采集到的数据进行外部电路二值化处理，再由电子线路传送到单片机，在单片机内计算出赛车面对的道路，选择最优行进路线，并采用优化后的控制策略使赛车能够快速安全的行驶。本团队设计的系统是一个软硬件与机械相结合的复杂整体，硬件部分主要包括电源管理模块、电机驱动模块，速度测量模块、辅助调试模块、图像采集处理模块、舵机控制模块和单片机模块等；软件部分需要解决的是单片机初始化程序、速度测量程序、速度设定程序、速度控制程序、舵机控制程序、图像识别程序等方面的内容，另外，本届比赛对机械的改进和创新也成为提高速度必不可少的组成部分。

关键词：智能车、恩智浦、MK60FN1M0VLQ15、PID 控制算法、摄像头

目录

第一章 引言.....	-1-
1.1 研究背景.....	-1-
1.2 章节安排.....	-2-
第二章 系统整体方案设计.....	-4-
2.1 系统整体结构.....	-4-
2.2 智能车软件系统总体设计.....	-5-
2.3 车模方向控制.....	-5-
第三章 机械结构设计说明.....	-6-
3.1 车体机械建模.....	-6-
3.2 底盘高度的调整与固定.....	-6-
3.3 前轮定位的调整.....	-6-
3.3.1 主销后倾.....	-7-
3.3.2 主销内倾.....	-7-
3.3.3 前轮外倾.....	-8-
3.3.4 前轮前束.....	-8-
3.4 转向机构调整优化.....	-9-
3.5 后轮减速齿轮机构调整.....	-9-
3.6 编码器的安装.....	-10-
3.7 摄像头的安装.....	-11-
3.8 车体重心位置的调整.....	-11-
3.9 本章小结.....	-12-
第四章 硬件系统设计.....	-13-
4.1 单片机系统.....	-13-
4.2 电源管理芯片.....	-14-
4.2.1 3.3V 稳压芯片.....	-14-
4.2.2 12V 稳压芯片.....	-15-
4.3 电机驱动模块.....	-16-
4.4 图像采集模块.....	-16-
4.5 速度检测模块.....	-17-
4.6 人机交互模块.....	-18-

4.8 本章小结.....	-18-
第五章 软件系统设计.....	-19-
5.1 总体控制流程图.....	-19-
5.2 程序初始化.....	-20-
5.3 图像采集算法.....	-21-
5.4 中心线提取及优化处理.....	-21-
5.4.1 原始图像的特点.....	-21-
5.4.2 赛道边缘提取.....	-23-
5.4.3 图像校正.....	-25-
5.4.4 推算重心.....	-26-
5.4.5 路径选择.....	-27-
5.5 控制算法.....	-27-
5.5.1 PID 控制算法简介.....	-27-
5.5.2 电机控制算法.....	-31-
5.5.3 舵机控制算法.....	-31-
5.6 本章小结.....	-31-
第六章 调试平台及调试场地.....	-32-
6.1 调试平台.....	-32-
6.2 调试场地.....	-33-
6.3 辅助调试.....	-33-
6.4 本章小结.....	-34-
第七章 模型的主要技术参数.....	-35-

第一章 引言

1.1 研究背景

教育部为了加强大学生实践、创新能力和团队精神的培养，在已举办全国大学生数学建模、电子设计、机械设计、结构设计等大竞赛的基础上，经研究决定，委托教育部高等学校自动化教学指导分委会主办每年一度的全国大学生智能汽车竞赛，并成立了由教育部、自动化分教指委、清华大学、飞思卡尔半导体公司等单位领导及专家组成的组委会。该竞赛是为了提高大学生的动手能力和创新能力而举办的，具有重大的现实意义。与其它大赛不同的是，这个大赛的综合性很强，是以迅猛发展的汽车电子为背景，涵盖了控制、模式识别、传感、电子、电气、计算机和机械等多学科交叉的科技创意性比赛，这对进一步深化高等工程教育改革，培养本新意识，培养硕士从事科学、技术研究能力，培养博士生知识、技术创新能力具有重要意义。

以智能汽车为研究背景的科技创意性制作，是一种具有探索性的工程实践活动，其本质也是人类创造有用人工物的一种训练性实践，其过程属性是综合，而结果属性很可能是创造。通过竞赛，参赛的同学们培养了对已学过的基础与专业理论知识与实验的综合运用的能力；带着背景对象中的各种新问题，学习控制、模式识别、传感技术、电子、电气、计算机、机械等多个学科新知识，包括来自不同学科背景大学生的相互学习，逐渐学会了在学科交叉、集成基础上的综合运用；若是以实用为目的，还必须考虑考虑可靠性、寿命、外观工业设计、集成科学与非科学，在具体约束条件下融合形成整体的综合运用。这样的训练是很有意义的。

在智能车的开发过程中，各参赛队伍需要改装竞赛车模，完成智能巡线小车的制作。在此过程中需要学习嵌入式系统开发环境与在线编程方法、单片机接口电路设计，自行设计实现识别引导黑线的硬件电路、电机的驱动电路、车速反馈电路、智能车舵机控制电路及能使小车在不驶出赛道的前提下尽可能快速行驶的控制策略与软件算法。飞思卡尔智能车竞赛到目前为止已经成功组织了十届，在赛制、规则、组别的不断改进中，增加了比赛的观赏性、技术性和实用性。于今年举办第十一届，并更名为恩智浦杯。

智能车的开发与设计涉及到多个专业领域，对于大学生综合素质的培养，知识面的拓展和分析问题解决问题的能力提高很有意义，并且有利于提高大学生的动手能力、激发创新能力。此外，制作这样一个高性能智能小车的过程，也是需要同组成员相互协作、紧密配合的过程，在此过程中，团队成员的交流与合作也显得尤为重要。

1.2 章节安排

第一章引言：主要介绍研究背景及智能车比赛的发展历程与竞赛内容，并介绍本组别规则产生的较大变化。

第二章整体方案设计：简要叙述智能车摄像头组的运行原理，并将整体控制任务进行分解，分模块介绍相关控制原理。

第三章机械结构设计说明：主要介绍了对原有车模进行的改装，针对重心位置及质量分布所做的机械改造，并简要介绍车上硬件的安装方法。

第四章硬件系统设计：以功能为依据，分模块对智能车电路进行介绍，并简述在制作及调试过程中遇到的相关问题及解决办法。

第五章软件系统设计：介绍软件开发环境, 根据控制任务分解，简单介绍相关算法实现。

第六章开发与调试：主要讲述智能车调试过程，以及在调试过程中遇到的主要问题及解决方法。

第七章智能车技术参数说明：对最终制作完成的智能车主要技术参数进行说明。

第二章 系统整体方案设计

根据大赛的统一要求，须在组委会统一提供的车模平台上，自主选择传感器类型，设计系统硬件电路，开发软件算法。车模平台、硬件电路系统、软件算法是本系统主要包括的三个大部分。每一个部分又由各个小模块构成。所以要构建一个完整的小车系统，必须先对各个模块进行论证和设计，再将整个系统组合成一个完整系统，进行系统联调。

2.1 系统总体结构

智能车主要由三个部分组成：检测系统，控制决策系统，动力系统。其中检测系统采用 CMOS 摄像头 OV7725，控制决策系统采用 MK60FN1M0VLQ15 作为主控芯片，动力系统主要控制舵机的转角和直流电机的转速。整体的流程为：通过视觉传感器来检测前方的赛道信息，并将赛道信息发送给单片机。同时，通过光电编码器构成的反馈渠道将车体的行驶速度信息传送给主控单片机。根据所取得的赛道信息和车体当前的速度信息，由主控单片机做出决策，并通过 PWM 信号控制直流电机和舵机进行相应动作，从而实现车体的转向控制和速度控制。

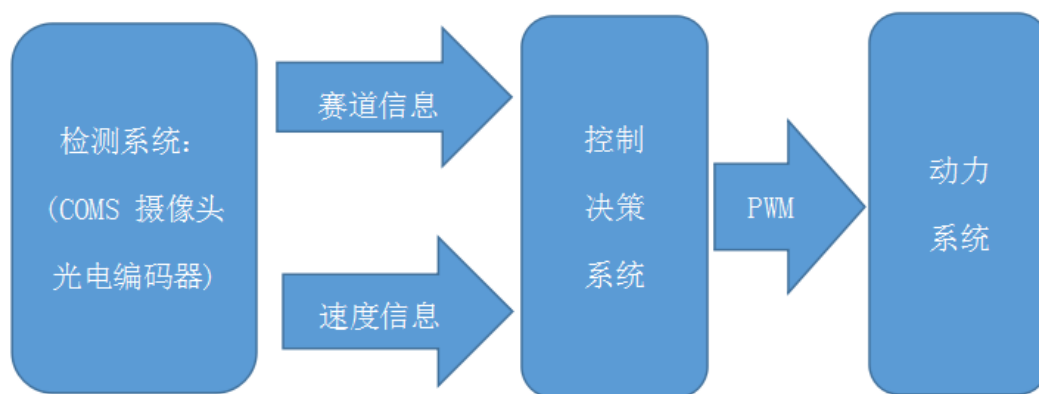


图 2.1 智能车系统总体结构框图

2.2 智能车软件系统总体设计

系统硬件对于赛车来说是最基础的部分，软件算法则是赛车的核心部分，该部分会在第五章中详细介绍。

2.3 车模方向控制

在方向控制中，采用摄像头对赛道信息进行采集，经过相关处理之后，计算得方向偏差。以此作为方向控制的输入量，通过舵机的转角来控制车模前行的方向。通过调整车模的方向，再加上车前行运动，可以逐步消除车模距离中心线的距离差别。这个过程是一个积分过程，因此车模差动控制一般只需要进行简单的比例控制就可以完成车模方向控制。

本章主要介绍了智能汽车控制系统的工作原理和赛车整体结构设计。硬件、软件和机械部分的有效融合是赛车能否跑出好成绩的关键因素。根据系统的需求简要给出了系统的总体设计方案，并具体将整个系统划分成检测系统、控制决策系统和动力系统，并对系统的总体框架作了一个简单的介绍。

第三章 机械结构设计说明

3.1 车体机械建模

此次竞赛的赛车车模选用由北京科宇通博科技有限公司提供的 B 型车模。车模外形如图 3.1 所示。

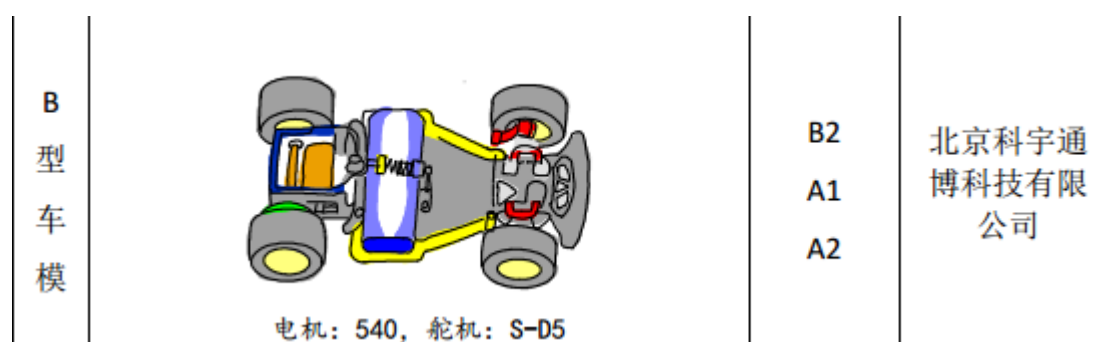


图 3.1 车体机械模型

3.2 地盘高度的调整与固定

在保证顺利通过坡道的前提下，底盘尽量降低，从整体上降低模型车的重心，使模型车转弯时更加稳定、高速。于是我们增加前轮垫片，协调后轮的垫片，使车模重心达到了允许范围内的最低，防止了允许过程中翻车的发生。使车模更加稳定，有利于速度的提升。

此外我们参照组委会的相关规定，用废弃的 PCB 板对底盘与后轮电机连接部分进行了固定，这样使底盘与后轮电机连接部分连接成为一个刚性结构，从而大大地减小了车模行驶过程中摄像头的抖动，提高了车模的整体稳定性。

3.3 前轮定位的调整

现代汽车在正常行驶过程中，为了使汽车直线行驶稳定，转向轻便，转向后能自动回正，减少轮胎和转向系零件的磨损等，在转向轮、转向节和前轴之间须形成一定的相对安装位置，叫车轮定位。车轮定位包括主销后倾角、主销内倾角、前轮外倾角和前轮前束四个内容。如图 3.3。

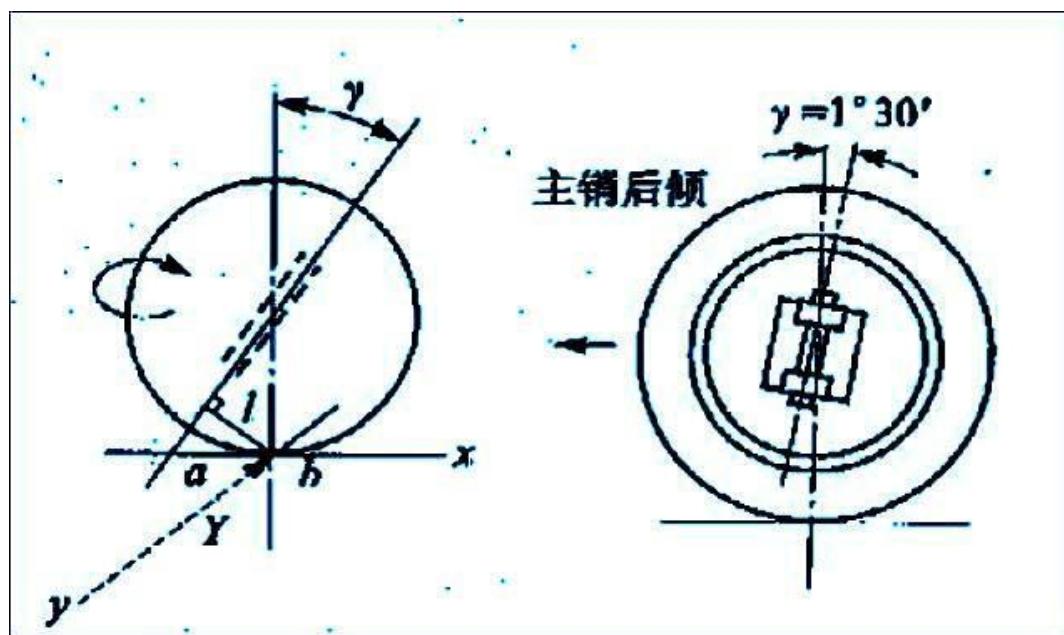


图 3.3 前轮定位原理图

3.3.1 主销后倾

所谓主销后倾，是将主销（即转向轴线）的上端略向后倾斜。从汽车的侧面看去，主销轴线与通过前轮中心的垂线之间形成一个夹角，即主销后倾角。主销后倾的作用是增加汽车直线行驶时的稳定性和在转向后使前轮自动回正。由于主销后倾，主销（即转向轴线）与地面的交点位于车轮接地点的前面。这时，车轮所受到的阻力的作用点总是在主销轴线之后，相当于主销拖着车轮前进。这样，就能保持行驶方向的稳定性。当车转弯时，由于车轮所受阻力作用线，不通过主销轴线，这样，车轮所受阻力在主销方向有力矩作用产生，迫使车轮自动偏转直到到车轮所受阻力作用线通过主销轴线，此时，车轮已回正，这就是转向车轮的自动回正功能。

主销后倾角越大，方向稳定性越好，自动回正作用也越强，但转向越沉重。汽车主销后倾角一般不超过 3° ，由前悬架在车架上的安装位置来保证。

现代轿车由于采用低压宽幅子午线轮胎，高速行驶时轮胎的变形加大，接地点后移，因此主销后倾角可以减小，甚至为负值（变成主销前倾），以避免由于回正力矩过大而造成前轮摆振。

模型车通过增减黄色垫片的数量来改变主销后倾角的，由于竞赛所用的转向舵机力矩不大，过大的主销后倾角会使转向变得沉重，转弯反应迟滞，所以设置为 0° ，以便增加其转向的灵活性。

3.3.2 主销内倾

所谓主销内倾，是将主销（即转向轴线）的上端向内倾斜。从汽车的前面看去，主销轴线与通过前轮中心的垂线之间形成一个夹角，即主销内倾角。主销内倾的作用是使车轮转向后能及时自动回正和转向轻便。对于模型车，通过调整前桥的螺杆的长度可以改变主销内倾角的大小，由于过大的内倾角也会增大转向阻力，增加轮胎磨损，所以在调整时可以近似调整为 $0^{\circ} \sim 3^{\circ}$ 左右，不宜太大。主销内倾和主销后倾都有使汽车转向自动回正，保持直线行驶的功能。不同之处是主销内倾的回正与车速无关，主销后倾的回正与车速有关，因此高速时主销后倾的回正作用大，低速时主销内倾的回正作用大。

3.3.3 前轮外倾

车轮面与地面不垂直。从车头望向车轮，车轮与铅垂线的夹角称为外倾角。若轮胎上端向外倾斜即左右轮呈“\”形，称为正外倾角，向内倾斜为负外倾角。基本上，正外倾角的设定有较佳的灵活度，而负外倾角具较稳定的直进性。其角度的不同能改变轮胎与地面的接触点及施力点，直接影响轮胎的抓地力及磨损状况。并改变了车重在车轴上的受力分布，避免轴承产生异常磨损。此外，外倾角的存在可用来抵消车身荷重后，悬吊系统机件变形及活动面间隙所产生的角度变化。外倾角的存在也会影响车子的行进方向，这正如摩托车可利用倾斜车身来转弯，因此左右轮的外倾角必须相等，在力的平衡下不致影响车子的直进性，再与束角(Toe)配合，提高直进稳定性及避免轮胎耗不均。增加负的外倾角需配合增加 Toe-out；增加正的外倾角则需配合增加 Toe-in。

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。

3.3.4 前轮前束

前轮前束，是使汽车两前轮的前端距离小于后端距离。其距离之差叫做前束值。从汽车的上面往下看，左右两个前轮形成一个开口向后的“八”字形。前轮外倾有使前轮向外转向的趋势，前轮前束有使车轮向内转向的趋势，可以抵消因前轮外倾带来的不利影响，使车轮直线滚动而无横向滑拖的现象，减少轮胎磨损。悬架系统铰接点的变形，也使前轮有向外转向的趋势，也要靠前轮前束来补偿。

在模型车中，前轮前束是通过调整伺服电机带动的左右横拉杆实现的。主销在垂直方向的位置确定后，改变左右横拉杆的长度即可以改变前轮前束的大小。在实际的调整过程中，我们发现较小的前束，约束 $0 \sim 2\text{mm}$ 可以减小转向阻力，使模型车转向更为轻便，但实际效果不是十分明显。

虽然模型车的主销后倾角、主销内倾角、车轮外倾角和前束等均可以调整，但是由于车模加工和制造精度的问题，在通用的规律中还存在着不少的偶然性，一切是实际调整的效果为准。如图 3.3.4 所示。

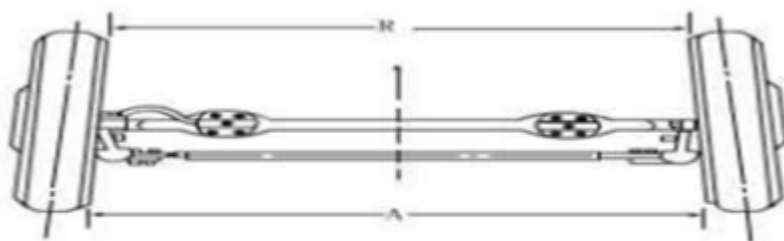


图 3.3.2 前轮前束示意图

3.4 转向机构调整优化

原装 B 车模的舵机为 SD-5 型号的舵机，比赛车模的转向是通过舵机带动左右横拉杆实现。舵机的转动速度和功率是一定，要想加快转向机构的响应速度，唯一的办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乘积的函数，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，因此设计时就要综合考虑转向机构响应速度与舵机力矩之间的关系，通过优化得到一个最佳的转向效果。利用实际参数经计算，我们得出了一套可以稳定高效工作的参数及结构。最终，我们设计了一套舵机连片(转向拉杆)，综合考虑了速度与力矩的关系，并根据模型车底盘的具体结构，简化了安装方式，实现了预期目标。

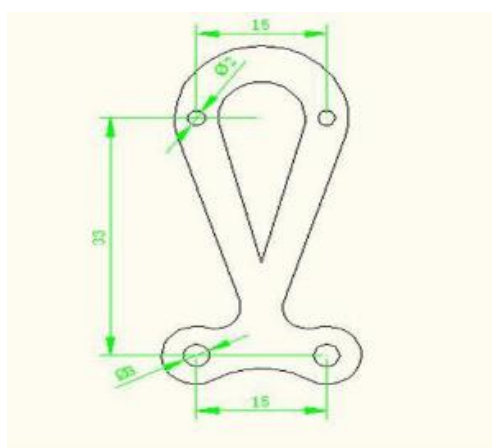


图 3.4.1 转向拉杆

关于舵机的安装方式，较为主流的有直立式安装和倒置式安装，我们的舵机安装如图 3.4.2 所示。与连片的啮合类似齿轮的啮合，由于连片长度以及齿

宽的影响，很难调节到合适的舵机机械中值。为了弥补这种连片的缺陷，我们采用通过软件调整的方法调节舵机中值。

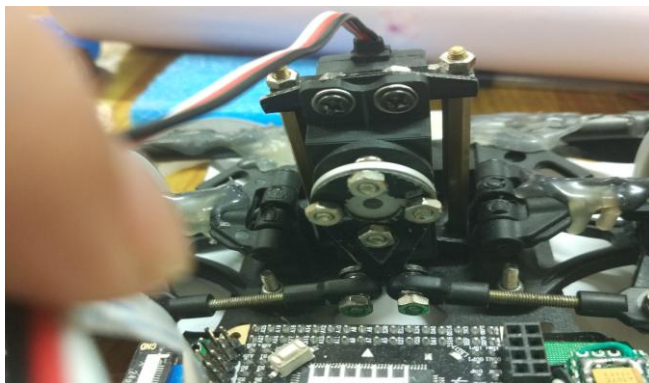


图 3.4.2 舵机安装图

3.5 后轮减速齿轮机构调整

模型车采用 540 电机驱动，该电机相较于其他车模的电机而言，功率更大，跑的更快。齿轮传动机构对车模的驱动能力有很大的影响。齿轮传动部分安装不恰当，会增大电机驱动后轮的负载；齿轮配合间隙过松则容易打坏齿轮过紧则会增加传动阻力。所以我们在电机安装过程中尽量使得传动齿轮轴保持平行，传动部分轻松、流畅，不存在卡壳或迟滞现象。噪音很小，并且不会有碰撞类的杂音，后轮减速齿轮机构就基本上调整好了，动力传递十分流畅。

3.6 编码器的安装

编码器在智能车当中起到了测速的作用，通过编码器对电机转速的测定我们可以知道车模在某个时刻的速度，从而通过软件调整差值，是车模的速度达到我们想要的效果，让车模电机运转处于我们所要求的理想状态，从而形成一个速度闭环控制系统，我们使用了测速编码器去测量脉冲数，就可以得到当前电机的转速。如图 3.6 所示。

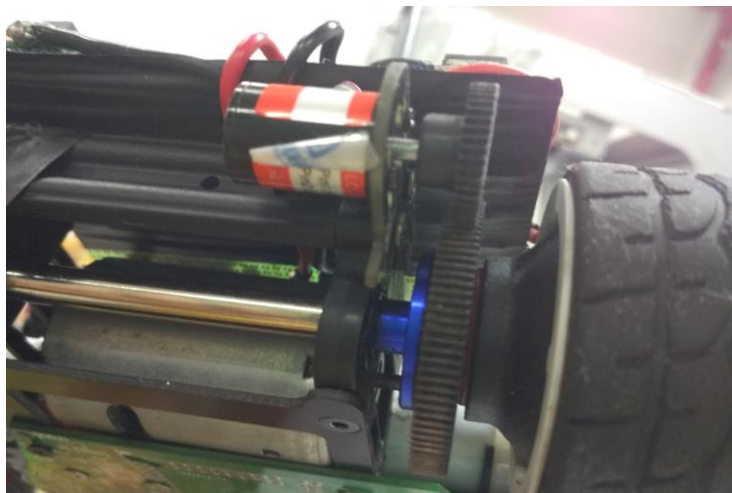


图 3.6 编码器安装示意图

3.7 摄像头的安装

由于线性摄像头镜头视角的限制，只有将摄像头架的比较高才能得到想要的赛道宽度，于是额外增加了两块铝片使竖直的碳素竿稳定。使用轻质的纤维杆，尽量降低重心。见下图 3.7 所示。



图 3.7 摄像头安装示意图

3.8 车体重心位置的调整

为了提高车的速度，小车就需要有一个比较长的前瞻，然而前瞻的长度增加使整车的重心前移。为了达到一个较长的前瞻长度同时保证车的重心尽可能

的靠后，我们把舵机和主板尽可能的靠后安装，同时把电机的驱动板从主板上分离开来，安装到后桥的电机上方，使整车的重心后移，并加大了后轮的垂直下压力提高后轮的抓地力又不至于后轮过重出现甩尾的现象。

主板和驱动板的安装实物如下：

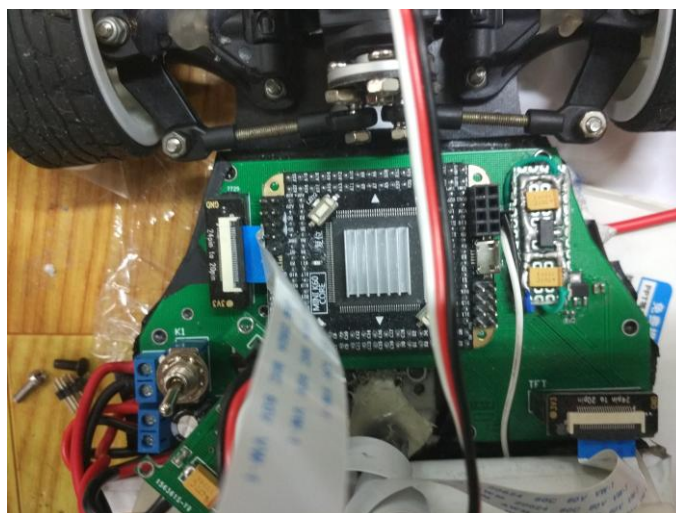


图 3.8.1 主板的安装



图 3.8.2 驱动板的安装

3.9 小结

本章通过对 B 型车模进行系统的建模分析，在比赛规则的允许范围内，制定了整车机械系统的设计安装和改进方案，使其能够符合智能车的竞技要求，

主要包括模型车机械部分安装及改造、传感器的设计安装、系统电路板的固定及连接等。车速较高时，小车的机械结构显得尤为重要，只有好的机械结构才能提高小车的最高速度。

第四章 硬件系统设计

本系统硬件电路的设计目标为：可靠、高效、简洁。可靠性是系统设计的第一要求，我们对电路设计的所有环节都进行了电磁兼容性设计，做好各部分的接地、屏蔽、滤波等工作，将高速数字电路与模拟电路分开，并用 0Ω 电阻将数字地与模拟地连接，有效克服电平漂移，使本系统工作的可靠性达到了设计要求。高效是指本系统的性能要足够强劲，具有较强的带载能力。简洁是指在满足了可靠、高效的要求后，为了尽量减轻整车重量，降低车体重心位置，应使电路设计尽量简洁，尽量减少元器件使用数量，缩小电路板面积，使电路部分重量轻，易于安装。硬件部分主要由 MK60FN1M0VLQ15 32 位单片机，电源模块，图像采集模块，速度检测模块，电机驱动模块，舵机驱动模块，人机交互模块（TFT 小液晶）等组成。

4.1 单片机系统

采用 freescale 公司 MK60DN512ZVLQ10, 32bit 的单片机，使用方便，功能强大。核心板主要用来处理传感器采集到数据，同时输出控制信号，根据摄像头采集到的图像（硬件二值化后的数据），求得黑线的实际位置，并根据 PID 算法对舵机和电机进行控制，同时它还控制速度参数的设定等工作，因此核心板是整个系统的控制中心。单片机的硬件资源是有限的，合理的利用单片机的硬件资源是保证系统稳定的前提。在本系统中，既涉及到信号的采集，同时涉及到控制部分。在采集部分，使用的是中断捕捉方式进行，而在控制部分，使用的是单片机自带的 PWM 控制波。

电源模块基本分为三大块，一是电池电压直接给电机驱动供电和转 5V 之后供给伺服电机的供电；二是通过升压芯片 MC34063 转换为 12V 给电机驱动 IC 供电；三是稳压芯片 LT1117 转换为 3.3V 电压供给摄像头、单片机、编码器、液晶屏幕。电源管理模块如图 4.2 所示。

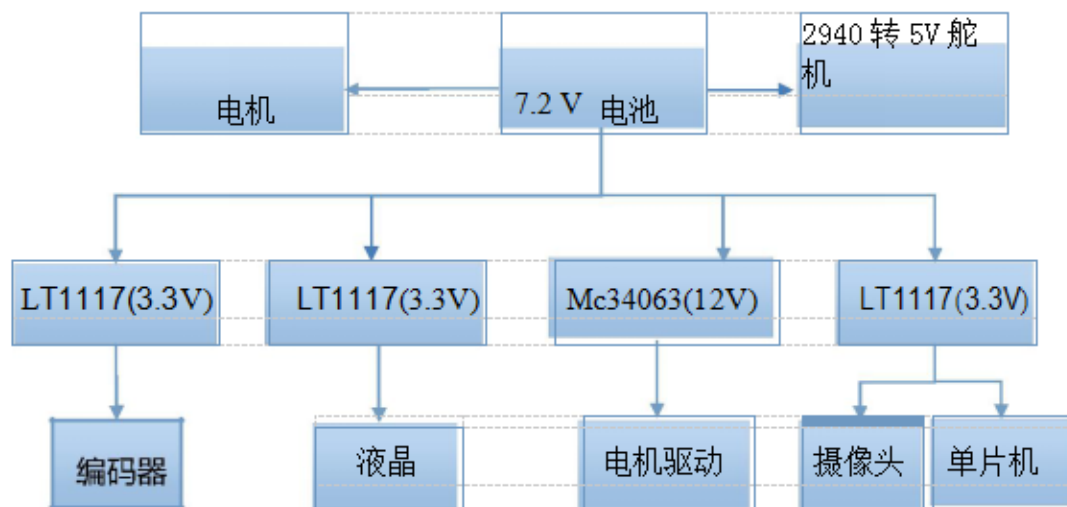


图 4.2 电源管理模块

为了避免各个模块之间供电相互干扰，车速检测电路供电、摄像头和单片机供电、液晶供电我们分别用一块 LT1117-3.3V 芯片以保证系统正常运行。在电路设计中，考虑到由于电机驱动所引起的电源不稳定（主要为瞬态脉冲），在电源输入端，各芯片电源引脚都加入了滤波电路和稳压电容。为了避免由于驱动电机转动时所引起的电磁干扰，在电路板设计中，在印制板上做了敷铜处理，将电路中的“地”与敷铜面相连接。

4.2.1 3.3V 稳压电路

3.3V 稳压电路如下 4.2.1 图所示。

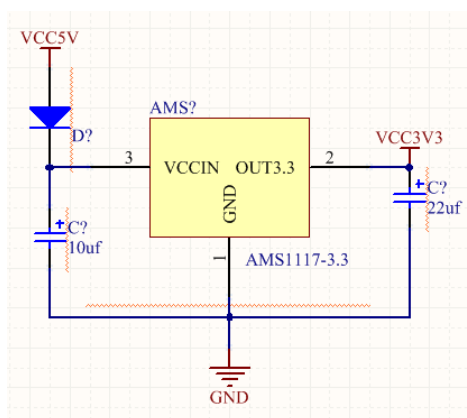


图 4.2.1 3.3V 稳压电路

4.3 电机驱动模块

小车的驱动模块是小车的动力之源。小车的驱动可以采用组委会提供的 MC33886 驱动芯片，也可采用最常用的 BTS7970 驱动芯片。但是根据以往经验，MC33886 在带载情况下会出现发热比较严重的现象，BTS7970 虽然带载能力虽然教 MC33886 要强，但对于 B 车电机来说依然会出现严重发热的现象，甚至会出现过热保护的现象。参考了往届的经验，我们采用 IR 公司的 N 沟 MOSFET LR7843 和专用栅极驱动芯片 IR2184s 设计。由于分立的 N 沟道 MOSFET 具有极低的导通电阻，大大减小了电枢回路总电阻。另外，专门设计的栅极驱动电路可以提高 MOSFET 的开关速度，使 PWM 控制方式的调制频率可以得到提高，从而减少电枢电流脉动。并且专用栅极驱动芯片通常具有防同臂导通、硬件死区、欠电压保护等功能，可以提高电路工作的可靠性。最终通过比较我们采用用两片 IR2184 及四个 NMOSFET 组成全桥的电机驱动电路，经实际应用表明，该驱动电路完全能解决严重发热的问题，大大地提高了电机的驱动能力，但是最后由于驱动损坏，我们不得不更换了由手焊的 7971 洞洞板驱动，效果还可以。电路如图 4.3 所示。

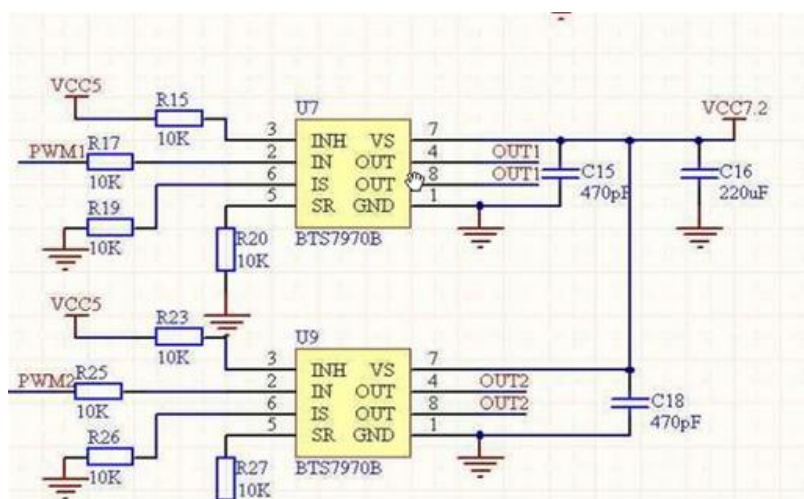


图 4.3 电机驱动电路

4.4 图像采集模块

CCD 摄像头具有对比度高、动态特性好的优点，但需要工作在 12V 电压下，对于整个系统来说过于耗电，而且 CCD 体积大，质量重，会抬高车体的重心，这对于高速情况下小车的行驶非常不利。与之相比，COMS 摄像头具有体积小、质量轻、功耗低，图像动态特性好等优点，因为小车队图像的清晰度，分辨率要求并不高，所以选用 COMS 摄像头。对于摄像头的选择，主要考虑以下几个参数：芯片大小、自动增益、分辨率、最小照度、信噪比、标准功率、扫描方式。

其中芯片大小主要会对视场的范围会有影响，扫描方式主要有追行扫描以及隔行扫描。

市面上的摄像头主要分为数字和模拟两种，数字摄像头主要有 OV7620, OV6620, OV7670, OV7725, 模拟摄像头主要有 OV5116, BF3003, MT9V136。大多数摄像头都支持 sccb 通信，可以很好的实现单片机与摄像头之间的交互。

智能车的摄像头对图像的分辨率要求并不高，但是对动态特性要求非常高，特别是小车在高速行驶入弯或者出弯的时候，图像变化较大，这就对摄像头的自动增益有较高的要求。一般来说，在摄像头图像发生突变时，感光芯片本身会有一段适应时间，这段时间要求越小越好。

我们先后试用了 OV7620、OV6620、OV7670、OV7725 以及 PC1030N 等摄像头，后两种均为彩色摄像头，在经过硬件二值化后可以取其有效信息，与黑白摄像头 OV762 得到的信息类似。经过对比，虽然后两种摄像头图像效果更好，动态性能也更好。经过最后使用效果，我们选用山外出厂的鹰眼 OV7725，自带硬件二值化电路，这样讲硬件二值化融合到摄像头上，使数据输出更加稳定，更有利于我们的算法处理。

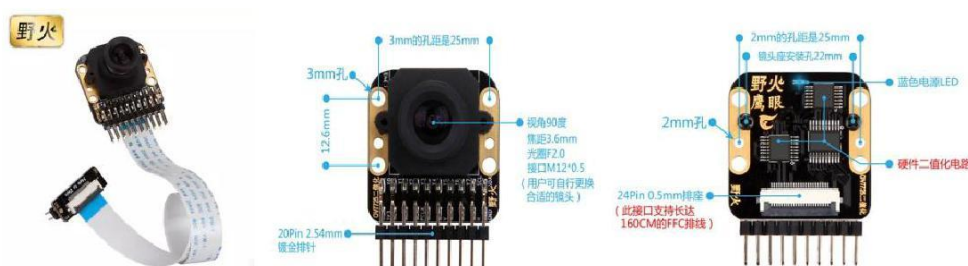


图 4.4 野火鹰眼摄像头实物图

4.5 速度检测模块

为了使用闭环控制，我们在车模上附加了编码器。经过机械和电路性能的考虑和挑选，对比于欧姆龙的 E6A2-CW3C 我们选用龙邱 512 线的 3 相增量式编码器。

和其他编码器相比，该编码器具有体积小，质量轻，稳定性高的特点，外壳直径 14mm，长度 18mm，完胜大个头光电式编码器，输出型号类型为 CMOS 信号，抗干扰能力强，不需上拉电阻直接和单片机连接，供电电压 3.3V 到 5V 可选，编码器可输出编码器的运转方向，即不需要正交解码也可以完成方向识别，同时该编码器也可使用正交解码程序。如下图所示。



图 4.5.1 编码器实物图

输出信号时序图:

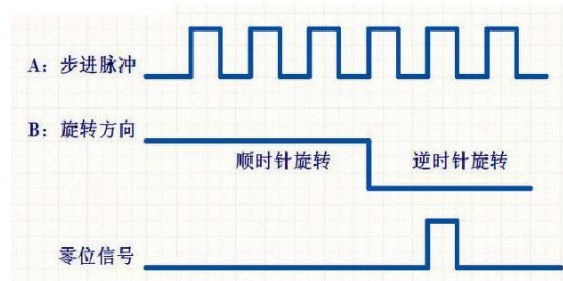


图 4.5.2 输出信号时序图

4.6 人机交互模块

在整个系统的调试过程中,通过人机交互模块可以显示智能车在寻迹过程中的一些相关信息和工作参数,以及改变相应的参数对应不同的线路。由于赛道在比赛前是不对外公布的,所以需要为小车设定了不同档的速度,以便对于不同难度的赛道可以灵活改变各个参数,使小车具有一定的适应性。同时也可以检测起跑线时设置不同的时间。所以结合以往的经验我们选用山外的 1.44 寸液晶显示屏,它自带六轴按键能很好满足我们的调试要求。如图 4.7 所示。

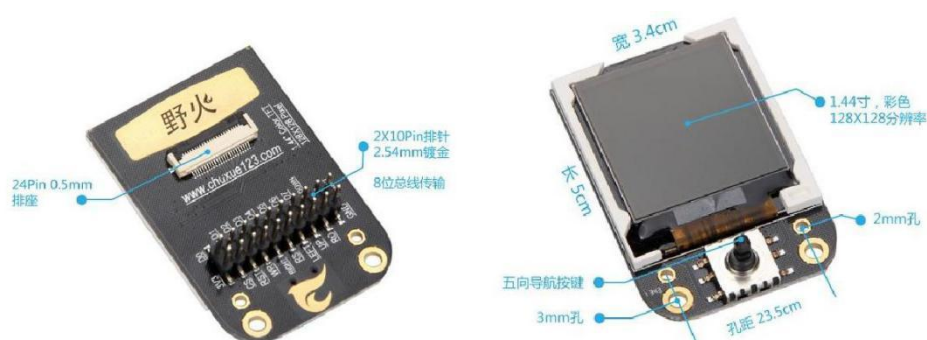


图 4.6 野火 1.44 寸液晶显示屏

4.7 本章小结

本章详细介绍了智能车系统的硬件电路结构和原理，系统硬件电路简单、可靠、工作稳定。

第五章 软件系统设计

5.1 总体控制流程图

程序主要用到 单片机 芯片中的 PWM 模块、输入捕捉中断模块，I/O 模块等。PWM 模块主要用来控制电机的运转；脉冲计数模块主要是用在了测速，捕捉中断并计算瞬时速度；输入捕捉中断模块用于采集行场中断；I/O 模块主要是用来读取按键信息和采集图像信息以及输出信息。

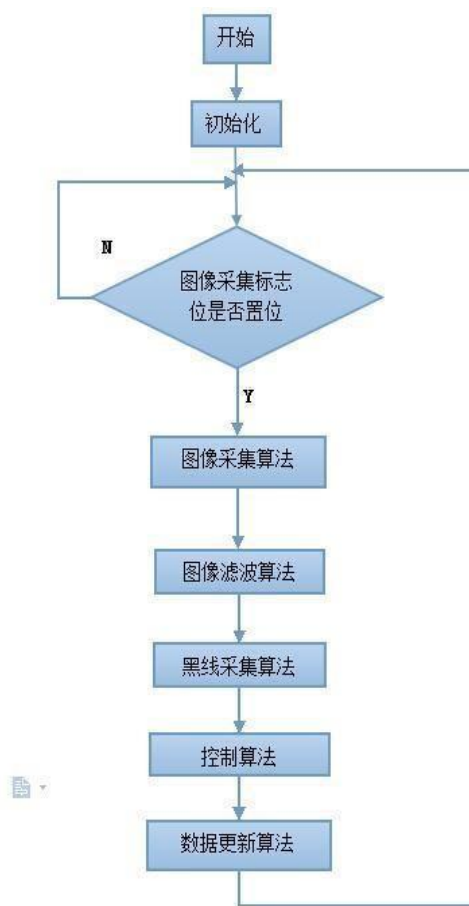


图 5.1 控制流程

5.2 程序初始化

程序开始运行后，在进入功能循环之前首先要对很多参数进行初始化，以便在后面的程序中才能正确调用，芯片的许多功能都需要初始化才能正常工作。初始化一般没有先后顺序，但一般有一定的书写习惯，本程序按照以下流程初始化，流程图如图 5.2.1 所示：

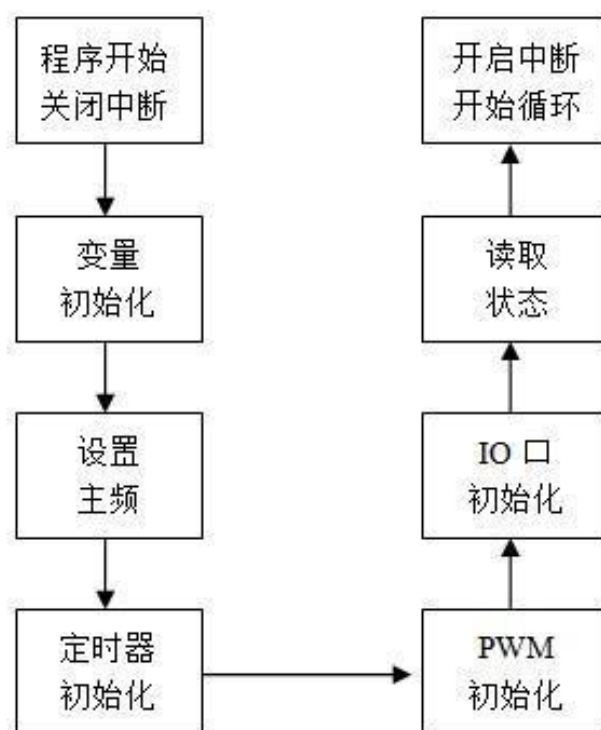


图 5.2.1 初始化流程

5.3 图像采集算法

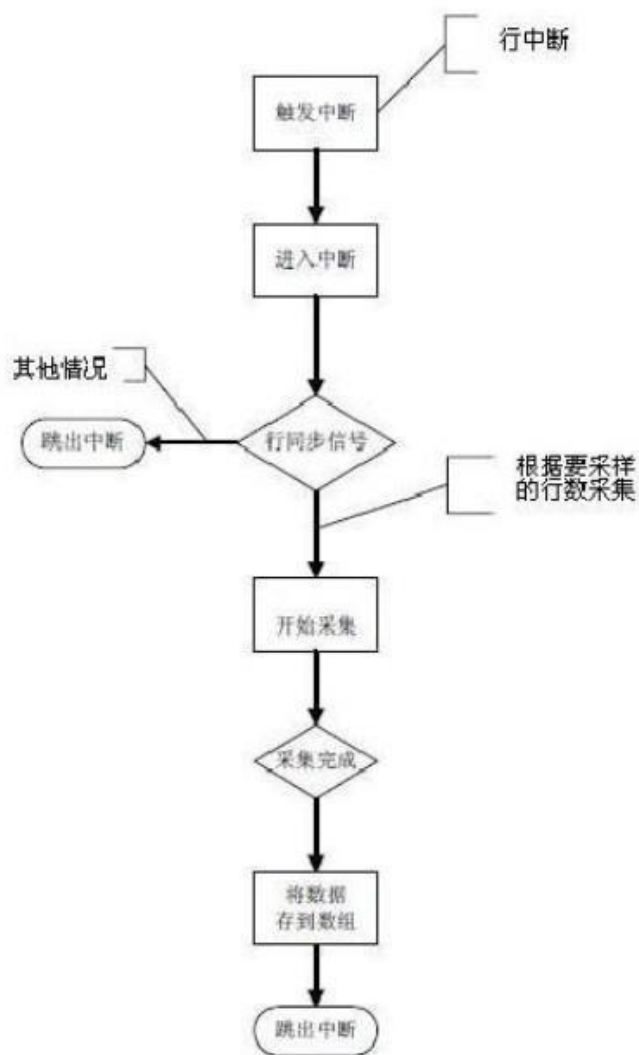


图 5.3 图像采集流程

5.4 中心线提取及优化处理

5.4.1 原始图像的特点

在单片机获取采集二值化后的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，由于交叉道、起点线的存在，光线、杂点、赛道远处图像不清楚的干扰，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。

在图像信号处理中我们提取的赛道信息主要包括：赛道两侧边沿点位置、通过校正计算的赛道中心位置，中心点规划面积，赛道变化幅度，赛道类型判别。

由于摄像头自身的特性，图像会产生梯形式变形，这使得摄像头看到的信息不真实。因此我们利用赛道进行测量，创建函数还原出了真实赛道信息。原始图像是一个将模拟图像经模拟电路转换得到的二维数据矩阵，矩阵的每一个元素对应一个像素点，图像的第一行对应最远处，大约 220cm，图像的最底部一行对应最近处，大约 5cm。远处的图像小，近处的图像大，黑线为梯形状。

单片机通过比较器电路将每一行的黑白跳变点（跳变点按从右到左的顺序）记录下来，保存到两个二维数组里（分别表示上升沿、下降沿）。通过遍历上升沿和下降沿可以完成赛道边沿的提取。摄像头采集到几种典型赛道图像如图 5.4.1~图 5.4.3 所示。

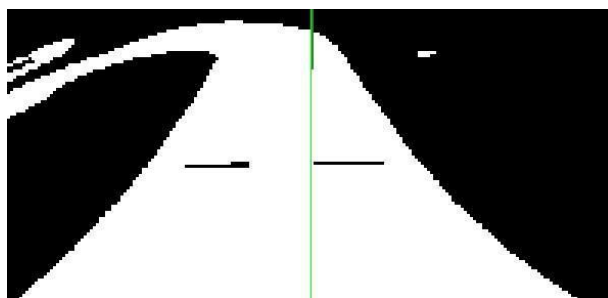


图 5.4.1 起点原始图像



图 5.4.2 弯道原始图像



图 5.4.3 十字交叉原始图像

5.4.2 赛道边沿提取

边沿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. 赛道宽度有一个范围，在确定的赛道宽度范围内提取有效赛道边沿，这样可以滤除不在宽度范围内的干扰；
3. 利用赛道的连续性，根据上一行白块的位置和边沿的位置来确定本行的边沿点；
4. 求边沿点时，因为近处的图像稳定，远处图像不稳定，所以采用由近及远的办法；
5. 进出十字的时候，通过校正计算出边沿角度可较好的滤除十字并补线；
6. 由于权重分配的问题，如果不对障碍进行一定的处理的话，控制量对于远端的障碍相应比较小，可能在很接近障碍的时候才有响应，这样很容易造成撞到障碍，为了消除这种影响，我们利用曲线包络的形式 将障碍作用的区域人为扩大，这样有效避免了碰撞障碍的危险。

边沿提取算法的程序流程如图 5.4.4 所示。

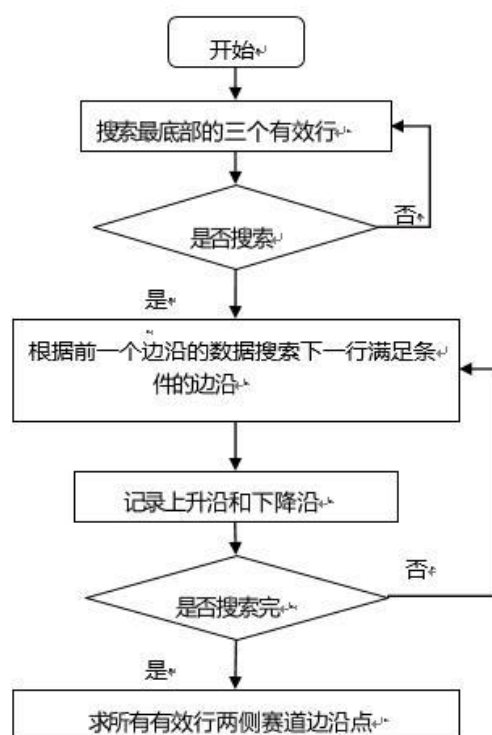


图 5.4.4 赛道边沿提取流程图

处理后得到的黑线中心如图 5.4.5~4.8 图所示。

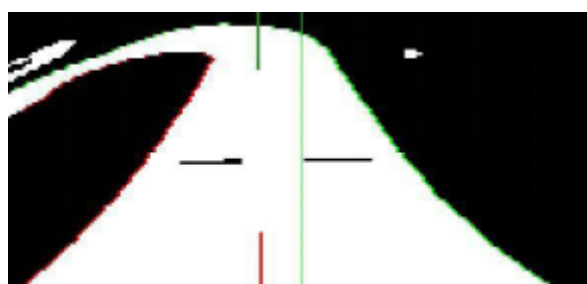


图 5.4.5 起点处理后图像

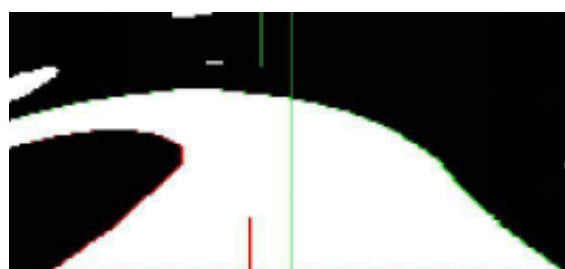


图 5.4.6 弯道处理后图像

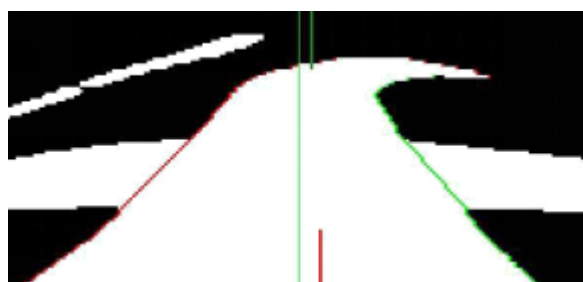


图 5.4.7 十字交叉处理后图像

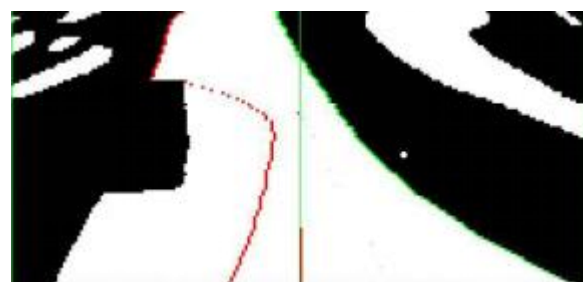


图 5.4.8 障碍处理图像

5.4.4 推算中心

通过之前提取的赛道边沿数据推算中心：当左右边沿点总数较少时返回；若只有单边有边沿点数据，则通过校正对单边数据按法线平移赛道宽度一半的距离；当能找到与一边能匹配上的另一边沿点时则直接求其中心作为中心点。推算完中心点后，对中心点进行均匀化，方便之后的控制。

5.4.5 路径选择

根据往届比赛以及本届安徽赛区的经验，赛车能否以最短的时间完成比赛，

与赛车的速度和路径都有着密切的关系，因此，如何使赛车以一个最合理、最高效的路径完成比赛是提高平均速度的关键。

对于赛车路径的优化，我们从以下三个方面来完成：

1. 增加视场的长度和宽度

根据我们的分析，当赛车采集到的图像能够覆盖一个比较完整的 S 弯道时，通过加权算法计算出来的中心就会处于视场中央附近，此时赛车会以一个比较好的路径快速通过 S 弯道；相反，如果视场无法覆盖一个完整的 S 弯道，赛车就会误处理为普通的单向弯道，这样赛车的速度就会大大减慢。因此，尽量增大视场的长度和宽度就很有必要了。

为了增加视场宽度，增加每行采集的图像点数之外，我们采用了广角镜头，从而有效地增加了视场宽度。

2. 优化加权算法

对整场有效行的中心求加权平均值的算法，在低速情况下可以有效地优化赛车路径，但在赛车速度提高到一定程度之后由于过弯时的侧滑，路径不是很

好。而由于图像分布不均，三分之二的行分布于车体前方 40cm 的范围内，求出的加权平均值受车体近处的图像影响较大，因此整场图像求加权的算法对于高速情况下的路径优化效果不是很明显。

为了解决这个问题，我们对于参与加权计算的图像行数及权重进行了处理，减小了车体前部 50cm 范围内的图像参与加权的行数和权重，同时增大视场前部图像的权重。在经过长期调试之后，得到了一套比较合适的参数，能够有效优化高速情况下的赛车路径。

3. 对不合理的中心点进行处理

对于在校正后的图像数据中求得的中心线，反校正到原始图像后存在一行中含有多个中心点的情况。在通常情况下，这种情况出现在较远的视野中，但由于我们增大了视场前部图像的权重，这些中心点对权重的影响极大，导致车模容易出现掉轮甚至冲出赛道的现象。

为了解决这个问题，我们利用数学方法求出了中心线的折点，对折点之后的中心点单独处理，使车模不再出现掉轮的现象。

5.5 控制算法

5.5.1 PID 控制算法简介

控制策略方面，使用经典控制方法里的 PID 控制。由于 PID 算法具有对模型参数要求不高、参数整定方便的优点，比较适合用来做小车寻迹。PID 控制算法思路：

经典 PID 算法介绍

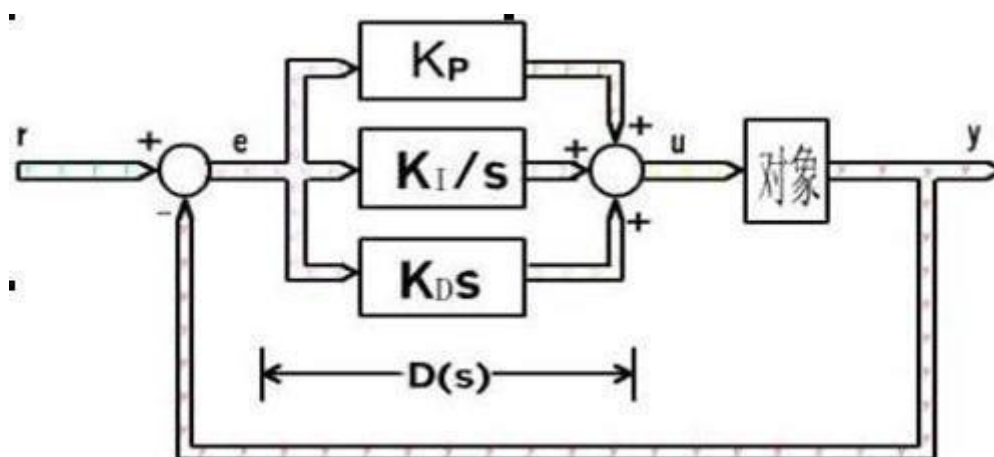


图 5.5.1 经典 PID 算法

1, PID 是一个闭环控制算法。因此要实现 PID 算法, 必须在硬件上具有闭环控制, 就是得有反馈。比如控制一个电机的转速, 就得有一个测量转速的传感器, 并将结果反馈到控制路线上, 下面也将以转速控制为例。

2, PID 是比例(P)、积分(I)、微分(D)控制算法。但并不是必须同时具备这三种算法, 也可以是 PD, PI, 甚至只有 P 算法控制。我以前对于闭环控制的一个最朴素的想法就只有 P 控制, 将当前结果反馈回来, 再与目标相减, 为正的话, 就减速, 为负的话就加速。现在知道这只是最简单的闭环控制算法。

3, 比例(P)、积分(I)、微分(D)控制算法各有作用:

比例, 反应系统的基本(当前)偏差 $e(t)$, 系数大, 可以加快调节, 减小误差, 但过大的比例使系统稳定性下降, 甚至造成系统不稳定;

积分, 反应系统的累计偏差, 使系统消除稳态误差, 提高无差度, 因为有误差, 积分调节就进行, 直至无误差;

微分, 反映系统偏差信号的变化率 $e(t)-e(t-1)$, 具有预见性, 能预见偏差变化的趋势, 产生超前的控制作用, 在偏差还没有形成之前, 已被微分调节作用消除, 因此可以改善系统的动态性能。但是微分对噪声干扰有放大作用, 加强微分对系统抗干扰不利。

积分和微分都不能单独起作用, 必须与比例控制配合。

4, 控制器的 P, I, D 项选择。

下面将常用的各种控制规律的控制特点简单归纳一下:

比例控制规律 P: 采用 P 控制规律能较快地克服扰动的影响, 它的作用于输出值较快, 但不能很好稳定在一个理想的数值, 不良的结果是虽较能有效的克服扰动的影响, 但有余差出现。它适用于控制通道滞后较小、负荷变化不大、控制要求不高、被控参数允许在一定范围内有余差的场合。如: 金彪公用工程部下设的水泵房冷、热水池水位控制; 油泵房中间油罐油位控制等。

比例积分控制规律(PI): 在工程中比例积分控制规律是应用最广泛的一种控制规律。积分能在比例的基础上消除余差, 它适用于控制通道滞后较小、负荷变化不大、被控参数不允许有余差的场合。如: 在主线窑头重油换向室中 F1401 到 F1419 号枪的重油流量控制系统; 油泵房供油管流量控制系统; 退火窑各区温度调节系统等。

比例微分控制规律(PD): 微分具有超前作用, 对于具有容量滞后的控制通道, 引入微分参与控制, 在微分项设置得当的情况下, 对于提高系统的动态性能指标, 有着显著效果。因此, 对于控制通道的时间常数或容量滞后较大的场合, 为了提高系统的稳定性, 减小动态偏差等可选用比例微分控制规律。如: 加热型温度控制、成分控制。需要说明一点, 对于那些纯滞后较大的区域里, 微分项是无能为力, 而在测量信号有噪声或周期性振动的系统, 则也不宜采用微分控制。如: 大窑玻璃液位的控制。

比例积分微分控制规律(PID): PID 控制规律是一种较理想的控制规律, 它在比例的基础上引入积分, 可以消除余差, 再加入微分作用, 又能提高系统的

稳定性。它适用于控制通道时间常数或容量滞后较大、控制要求较高的场合。如温度控制、成分控制等。鉴于 D 规律的作用，我们还必须了解时间滞后的概念，时间滞后包括容量滞后与纯滞后。其中容量滞后通常又包括：测量滞后和传送滞后。测量滞后是检测元件在检测时需要建立一种平衡，如热电偶、热电阻、压力等响应较慢产生的一种滞后。而传送滞后则是在传感器、变送器、执行机构等设备产生的一种控制滞后。纯滞后是相对与测量滞后的，在工业上，大多的纯滞后是由于物料传输所致，如：大窑玻璃液位，在投料机动作到核子液位仪检测需要很长的一段时间。

总之，控制规律的选用要根据过程特性和工艺要求来选取，决不是说 PID 控制规律在任何情况下都具有较好的控制性能，不分场合都采用是不明智的。

如果这样做，只会给其它工作增加复杂性，并给参数整定带来困难。当采用 PID 控制器还达不到工艺要求，则需要考虑其它的控制方案。如串级控制、前馈控制、大滞后控制等。

5，公式：

比例 (P) 控制器

$$u(t) = K_p e(t) \quad (3-2)$$

比例+积分 (PI) 控制器

$$u(t) = K_p \left[e(t) + \frac{1}{T_I} \int_0^t e(\tau) d\tau \right] \quad (3-3)$$

比例+积分+微分 (PID) 控制器

$$u(t) = K_p \left[e(t) + \frac{1}{T_I} \int_0^t e(\tau) d\tau + T_D \frac{d(e(t))}{dt} \right] \quad (3-4)$$

式中 K_p ——比例放大系数； T_I ——积分时间； T_D ——微分时间。

6, 问题。 K_p, T_i, T_d 三个参数的设定是 PID 控制算法的关键问题。一般说来编程时只能设定他们的大概数值, 并在系统运行时通过反复调试来确定最佳值。因此调试阶段程序须得能随时修改和记忆这三个参数。

7, 参数的自整定。在某些应用场合, 比如通用仪表行业, 系统的工作对象是不确定的, 不同的对象就得采用不同的参数值, 没法为用户设定参数, 就引入参数自整定的概念。实质就是在首次使用时, 通过 N 次测量为新的工作对象寻找一套参数, 并记忆下来作为以后工作的依据。

8, pid 算法流程图:

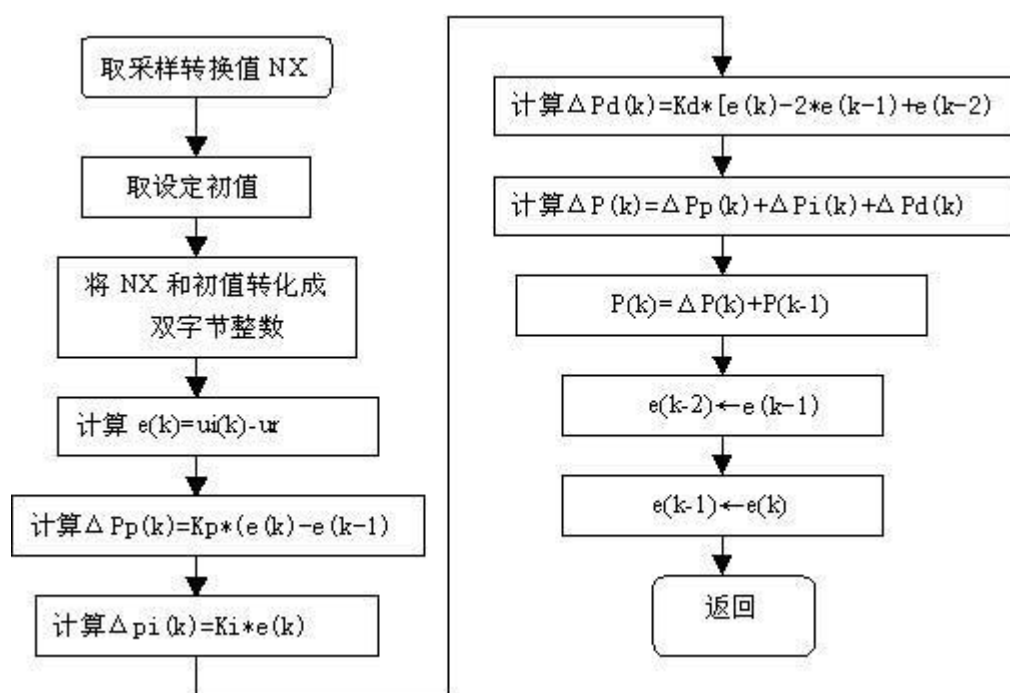


图 5.5.2 PID 算法流程图

5.5.2 电机控制算法

小车行驶途中, 根据不同的路况, 摄像头采集到的图像和计算出来的偏差是不同, 利用偏差的大小, 可以将路况分段, 再根据分段, 来给定适合的速度, 例如直道上偏差很小, 那么我们可以给一个较快的速度, 使小车在直道上加速, 当偏差越来越大时, 我们可以适当的降低车子的速度, 以至于不让车子冲出跑道。

给定的速度通过上面介绍的 PID 算法进行闭环, 通过我们的实际实验, 在给了合适的 P 值, I 值, D 值后, 通过 PID 的闭环能用够车子快速的到达我们指定的速度。使得车子, 能够根据路况的不同进行加减速, 从而顺利的通过不同的赛道类型。

5.5.3 舵机控制算法

同上, 利用摄像头采集到的图像和计算出来的偏差, 将路况分段, 不过, 由于舵机存在比较大的滞后, 所以我们采用了 PD 进行闭环, 让舵机可以更快

的响应，在过弯时，根据不同的偏差大，给定不同的 K_P 值和 K_D 值，具体的 K_P 值和 K_D 值由测试和实验得出。

经过实际调试测量控制后，小车的转向控制能满足要求。

5.6 本章小结

本章详细介绍了系统软件控制思想和算法，包括图像处理算法、电机、舵机控制算法。经试验调试，小车能够满足不同的赛道要求，具有一定的抗干扰能力。

第六章 调试平台及调试场地

6.1 调试平台

此次智能车大赛的软件开发平台为 IAR 集成开发环境。其使用界面如图 6.1 所示：

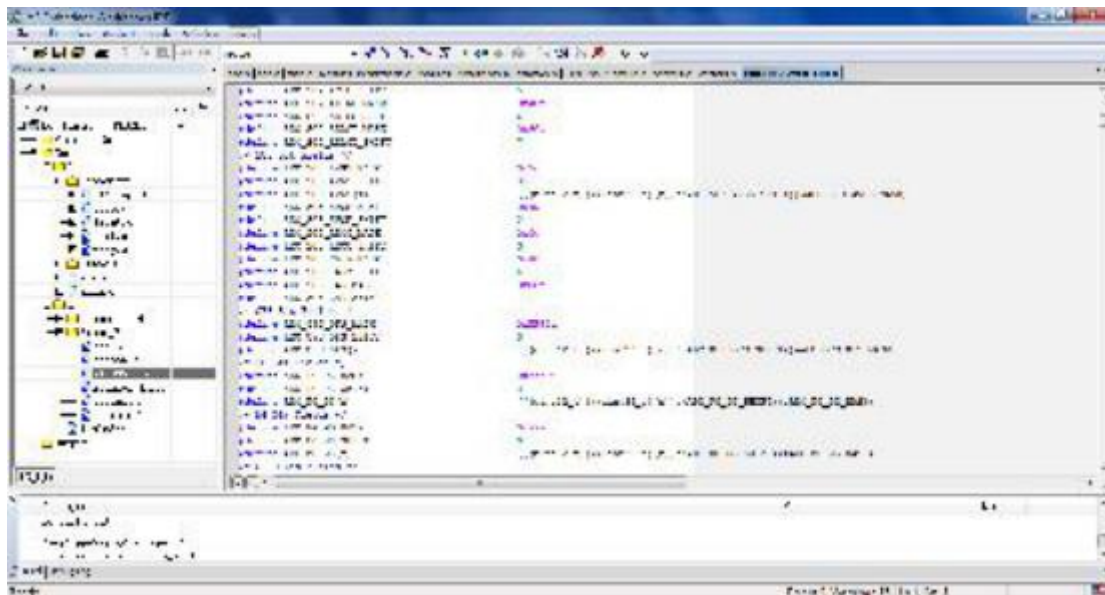
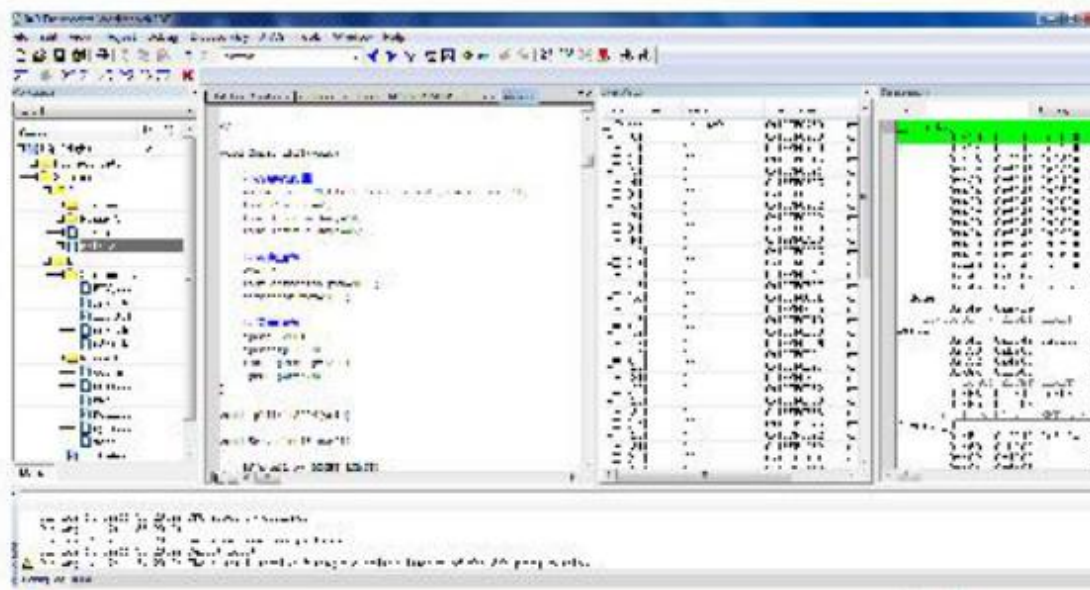


图 6.1 开发平台

IAR 的功能非常强大，可用于绝大部分单片机、嵌入式系统的开发。用户可在新建工程时将芯片的类库添加到集成环境开发环境中，工程文件一旦生成就是一个最小系统，用户无需再进行繁琐的初始化操作，就能直接在工程中添加所需的程序代码。如图 5.9 所示，利用 IAR6.0 和 OSJLINK 下载器，用户可以进行一系列的调试工作，如动态调试、监视寄存器状态、修改 PC 指针、设置断点等，这样能快速地帮助我们找到软件或硬件的问题。



6.2 调试场地

为了能够让车模能够适应各种不同的环境及不同的路径，我们设计了比较复杂的调试场地，该场地基本包括比赛中可能出现的各种情况。场地图片如下：



图 6.2 调试场地

6.3 辅助调试

为了调试的方便，我们加入了蓝牙模块，利用蓝牙，将小车行驶中的数据发送回电脑，再利用虚拟示波器，将数据显示出来，这样，我们便获得了动态的信息。利用这些及时的动态信息，我们能够更加直观掌控小车的行驶状况，也更有利于我们对控制参数的调试。调试界面如图 6.3 。

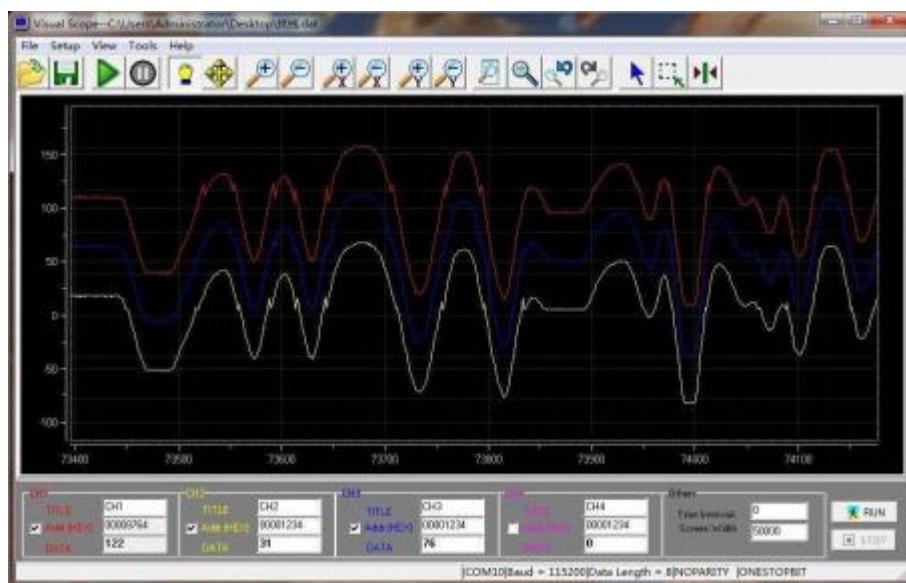


图 6.4 虚拟示波器调试界面

6.4 本章小结

本章介绍了小车调试过程所用到的调试方案，良好的辅助调试手段更有利于小车的调试，但是由于水平有限，所用调试方案仍然不是很好。希望以后发现更优的调试方案。

第七章 模型的主要技术参数

项目	参数
车长	27cm
车宽	17.7cm
车高	39cm
车重	1.12kg
传感器种类及个数	鹰眼摄像头 1 个 光电编码器 1 个
车模之外伺服电机个数	0

总结

经过近半年的努力，我们最终完成了摄像头 B 车模的智能车制作与调试。现在总结整个做车流程及方案设计等方面，综合考虑，我们的智能车具有以下特色：

- (1) 在本系统中，巧妙地使用迭代滤波和逻辑滤波，减少了检测误差，同时提高了小车运动的精度。
- (2) 在机械上精心设计，增大了整个车模的倾角，实现了模块化的电路设计，在重心的调节上更加方便。
- (3) 在连接件上，自行设计，由加工厂代工制造了机械加工零件，提高了安装精度，使得小车安装简洁、美观。

尽管我们作了很大的努力，但是由于时间紧迫以及缺乏经验等原因，本系统仍然存在着一些问题，主要有以下几点：

- (1) 由于车模的模块化电路管理，在小车上出现连线不稳，易松动的情况。
- (2) 为了提高 B 车模的摩擦力，尽量均匀分布质量，降低车模重心，导致车模转动惯量较大，对转向造成一定的影响。
- (3) 路径检测传感器 CMOS 的安装不稳固，由于路径检测的需求与整体车模结构导致 CMOS 只能架空安装，即使额外使用了支架进行加固仍然很容易随车体抖动，增大了路径检测的误差。

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] 卓晴, 黄开胜, 邵贝贝. 学做智能车. 北京-北京航空航天大学出版社. 2007.
- [3] 李宁, 刘启. 新电机自动控制系统. 北京-机械工业出版社. 2003.
- [4] 谭浩强. C++程序设计. 北京-清华大学出版社. 2004.
- [5] 潘松, 黄继业. 现代数字电路基础教程. 北京-科学出版社 2008.
- [6] 王水平. 开关稳压电源原理及设计. 北京-人民邮电出版社. 2008.
- [7] 仇慎谦. PID 调节规律和过程控制. 南京-江苏科学技术出版社. 1987, 12.

附录:

部分源代码:

```

void All_init(void)
{
    SCB->CPACR |=((3UL << 10*2)|(3UL << 11*2));    //开硬件浮点
    //初始化 flash
    //flash_init();
    //DELAY_MS(10);    //flash 初始化需要适当的延时
    //    data16_0=    flash_read(255,0,uint16);    //    data16_1 =
    flash_read(255,4,uint16);    //    data16_2 =
    flash_read(255,8,uint16);    //    data16_3 =
    flash_read(255,12,uint16);    //    data16_4 =
    flash_read(255,16,uint16);    //    data16_5 =
    flash_read(255,20,uint16);    //    data16_6 =
    flash_read(255,24,uint16);    //    data16_7 =
    flash_read(255,28,uint16);

    //初始化 LCD 屏
    LCD_init();

    //初始化摄像头
    camera_init(imgbuff);

    //初始化编码器
    gpio_init(PTC3,GPI,0);    //编码器方向
    lptmr_pulse_init(LPT0_ALT2,0xffff,LPT_Rising);    //初始化脉冲
    计数器,用 LPT0_ALT2,即 PTC5 输入,上升沿触发
    lptmr_pulse_clean();    //清空 lptmr
    脉冲计数器计算值

    //初始化 PIT
    pit_init_ms(PIT0,5);    //5ms 定时
    pit_init_ms(PIT1,10000);    //5ms 定时

```

```

//初始化系统板上 LED
led_init(LED0);
led_init(LED1);
led_init(LED2);
led_init(LED3);

//初始化 PWM 模块
ftm_pwm_init(FTM1, FTM_CH0, 300, 10000); //8800
ftm_pwm_init(FTM0, FTM_CH0, 10000, 0); //电机初始
化

ftm_pwm_init(FTM0, FTM_CH1, 10000, 0);

//初始化串口
//uart_init(UART4, 57600);

//初始化按键
key_init(KEY_MAX);

//配置中断服务函数
set_vector_handler(PORTA_VECTORn ,PORTA_IRQHandler); //设置
LPTMR 的中断服务函数为 PORTA_IRQHandler
set_vector_handler(DMA0_VECTORn ,DMA0_IRQHandler); //设置
LPTMR 的中断服务函数为 PORTA_IRQHandler
set_vector_handler(PIT0_VECTORn,pit_hander);
set_vector_handler(UART4_RX_TX_IRQn , uart0_handler);

//设置中断优先级
//NVIC_SetPriority(PORTA_IRQn, 0); //配置优先级
NVIC_SetPriority(DMA0_IRQn, 1); //配置优先级
NVIC_SetPriority(PIT0_IRQn, 2); //配置优先级
NVIC_SetPriority(UART4_RX_TX_IRQn , 0); //配置优先级
// NVIC_SetPriority(PORTE_IRQn, 3); //配置优先级

display_menu(); //主界面显示

//开中断
uart_rx_irq_en (UART4); //开串
口接收中断
//enable_irq(PIT_IRQn); //
开定时器中断

```

```
}

void set_motorspeed(void)
{

    Motor.LastLastError=Motor.LastError;
    Motor.LastError=Motor.Error;
    Motor.LastVaule=Motor.OutputVaule;
    Motor.Error=Motor.SetVaule-lptmr_pulse;
    Motor.OutputVaule=(int) (Motor.LastVaule+

CS.Motor_PID.Kpp*(Motor.Error-Motor.LastError)+
                        CS.Motor_PID.Kii*Motor.Error+

CS.Motor_PID.Kdd*(Motor.LastLastError-2*Motor.LastError+Motor.Error)
                        );

    if (Motor.OutputVaule>500)           //限制输出
    {
        Motor.OutputVaule=500;
    }
    if (Motor.OutputVaule<-500)
    {
        Motor.OutputVaule=-500;
    }

    if (Motor.OutputVaule>0)
    {
        ftm_pwm_duty (FTM0,FTM_CH0, Motor.OutputVaule);
        ftm_pwm_duty (FTM0,FTM_CH1, 0);
    }
    if (Motor.OutputVaule<0)
    {
        ftm_pwm_duty (FTM0,FTM_CH0, 0);
        ftm_pwm_duty (FTM0,FTM_CH1, -Motor.OutputVaule);
    }
}
```

```
void Duoji_P_Refresh(void)
{
    if(Err_Rate<=7 && Err_Rate>=-7)
    {
        CS.Duoji_L_PID.Kp=CS.Duoji_L_PID.KP;
        CS.Duoji_R_PID.Kp=CS.Duoji_R_PID.KP;
    }
    else if((Err_Rate<-7 && Err_Rate>=-12 )||( Err_Rate >7 &&
Err_Rate<=12))
    {

CS.Duoji_L_PID.Kp=CS.Duoji_L_PID.KP+CS.Duoji_Fenduan_P.P1;

CS.Duoji_R_PID.Kp=CS.Duoji_R_PID.KP+CS.Duoji_Fenduan_P.P1;
    }
    else if((Err_Rate<-12 && Err_Rate>=-23 )||( Err_Rate >12 &&
Err_Rate<=23))
    {

CS.Duoji_L_PID.Kp=CS.Duoji_L_PID.KP+CS.Duoji_Fenduan_P.P2;

CS.Duoji_R_PID.Kp=CS.Duoji_R_PID.KP+CS.Duoji_Fenduan_P.P2;
    }
    else if((Err_Rate<-23 && Err_Rate>=-40 )||( Err_Rate >23 &&
Err_Rate<=40))
    {

CS.Duoji_L_PID.Kp=CS.Duoji_L_PID.KP+CS.Duoji_Fenduan_P.P3;

CS.Duoji_R_PID.Kp=CS.Duoji_R_PID.KP+CS.Duoji_Fenduan_P.P3;
    }
    else
    {

CS.Duoji_L_PID.Kp=CS.Duoji_L_PID.KP+CS.Duoji_Fenduan_P.P4;

CS.Duoji_R_PID.Kp=CS.Duoji_R_PID.KP+CS.Duoji_Fenduan_P.P4; //4
    }
}
```

}