第十七届全国大学生

智能汽车竞赛

**技 术 报 告**



学 校： 哈尔滨工业大学

队伍名称： 紫丁香十队

参赛队员： 王路任航 董鑫宇 申吴鑫

陈嘉慧 张旭

带队教师： 张依

关于技术报告和研究论文使用授权的说明

本人完全了解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：王路任航 董鑫宇 申吴鑫 陈嘉慧 张旭

带队教师签名： 张依

日 期： 2022年8月15日

**摘 要**

本文详细介绍了哈尔滨工业大学紫丁香十队在第十七届全国大学生智能汽车竞赛完全模型组的系统方案。我们选择使用官方指定I车模来搭建四轮车模，以赛曙科技公司生产的EdgeBoard作为上位机，以英飞凌公司生产的TC264单片机作为下位机，要求智能车实现沿赛道寻线、施工区泛行区加油站等任务，并以尽可能快的速度稳定完成整个比赛。智能车采用杰锐微通公司生产的GZ330工业模组摄像头对赛道寻线，用神经网络模型对赛道上的标志进行检测并进入相应区域，通过编码器监测车的实时速度，使用PID控制算法调节电机的转速，实现了车模在运动过程中速度和方向的闭环控制；为了提高模型车的速度和稳定性，使用上位机、按键、OLED 模块等调试工具，进行了大量硬件与软件测试。该智能车系统实现高度的智能化、人性化，并且具备良好的安全性、稳定性，可以为无人驾驶汽车的后续研究提供经验。

关键词：TC264，杰锐微通GZ330摄像头，I车模、PID

目录

[**第一章** **引言** 3](#_Toc111842130)

[**1.1 智能车大赛介绍** 3](#_Toc111842131)

[**1.2 第十七届完全模型组规则介绍** 3](#_Toc111842132)

[**1.3 报告内容** 4](#_Toc111842133)

[**第二章** **机械结构设计** 5](#_Toc111842134)

[**2.1 整体布局** 5](#_Toc111842135)

[**2.2 转向舵机安装** 5](#_Toc111842136)

[**2.3 前轮倾角调节** 6](#_Toc111842137)

[2.3.1 前轮外倾 6](#_Toc111842138)

[2.3.2 前轮前束 7](#_Toc111842139)

[**2.4悬挂调节** 7](#_Toc111842140)

[**2.5姿态传感器** 7](#_Toc111842141)

[**2.6 摄像头** 7](#_Toc111842142)

[**2.7 车壳设计** 7](#_Toc111842143)

[**2.8 车模主要技术参数** 8](#_Toc111842144)

[**第三章** **硬件电路的设计** 9](#_Toc111842145)

[**3.1 电源管理模块设计** 9](#_Toc111842146)

[**3.2 驱动电路设计** 10](#_Toc111842147)

[**3.3 拓展坞电路设计** 10](#_Toc111842148)

[**3.4 调试电路设计** 11](#_Toc111842149)

[**第四章** **软件方案** 13](#_Toc111842150)

[**4.1 图像采集与处理** 13](#_Toc111842151)

[**4.2 神经网络模型识别** 13](#_Toc111842152)

[**4.3 PID运动控制算法** 13](#_Toc111842153)

[**4.3.1 位置式PID** 16](#_Toc111842154)

[**4.3.2 增量式PID** 16](#_Toc111842155)

[**4.3.3 PID参数整定** 17](#_Toc111842156)

[**4.4 舵机控制（pd控制）** 17](#_Toc111842157)

[**4.5 电机控制（pi控制）** 17](#_Toc111842158)

[**第五章** **系统的开发环境与车模调试** 18](#_Toc111842159)

[**5.1 EdgeBoard的调试** 18](#_Toc111842160)

[**5.2 下位机的调试** 18](#_Toc111842161)

[5.2.1人机交互 18](#_Toc111842162)

[5.2.2 开发调试工具 19](#_Toc111842163)

[**5.3 WIFI及上位机调试** 20](#_Toc111842164)

[**第六章 总结** 21](#_Toc111842165)

[**6.1车模总结** 21](#_Toc111842166)

[**6.1.1控制方面** 21](#_Toc111842167)

[**6.1.2结构方面** 21](#_Toc111842168)

[**6.1.3不足之处** 21](#_Toc111842169)

[**6.2工作综述** 21](#_Toc111842170)

[**致谢** 22](#_Toc111842171)

[**参考文献** I](#_Toc111842172)

[**附录** II](#_Toc111842173)

1. **引言**

**1.1 智能车大赛介绍**

全国大学生智能车竞赛受教育部高等教育司委托，由教育部高等自动化专业教学指导分委员会（以下简称自动化分教指委）主办全国大学生智能汽车竞赛。该竞赛以智能汽车为研究对象的创意性科技竞赛，是面向全国大学生的一种具有探索性工程实践活动，是教育部倡导的大学生科技竞赛之一，为加强大学生实践、创新能力和团队精神的培养，促进高等教育教学改革。该竞赛以“立足培养，重在参与，鼓励探索，追求卓越”为指导思想，旨在促进高等学校素质教育，培养大学生的综合知识运用能力、基本工程实践能力和创新意识，激发大学生从事科学研究与探索的兴趣和潜能，倡导理论联系实际、求真务实的学风和团队协作的人文精神，为优秀人才的脱颖而出创造条件。该竞赛由竞赛秘书处为各参赛队提供/购置规定范围内的标准硬软件技术平台，竞赛过程包括理论设计、实际制作、整车调试、现场比赛等环节，要求学生组成团队，协同工作，初步体会一个工程性的研究开发项目从设计到实现的全过程。该竞赛融科学性、趣味性和观赏性为一体，是以迅猛发展、前景广阔的汽车电子为背景，涵盖自动控制、模式识别、传感技术、电子、电气、计算机、机械与汽车等多学科专业的创意性比赛。该竞赛规则透明，评价标准客观，坚持公开、公平、公正的原则，保证竞赛向健康、普及、持续的方向发展。

**1.2 第十七届完全模型组规则介绍**

（1）赛道环境

完全模型组比赛赛道以室内循环赛道为基础，赛道材质，赛道规格均保持一致。在导引方式上完全保留室内循环赛道的导引方式，并在此基础上添加完全模型组任务导引标志和锥桶，引导车模完成完全模型组赛道任务。

（2）任务描述

完全模型组的基本比赛任务为：

  选手制作的车模完成从车库出发沿着赛道运行两周。车模需要分别通过道路设置的各种元素，识别道路中心的标志完成特殊路段通行。

  比赛时间从车模驶出车库到重新回到车库为止。如果车模没有能够停止在车库内停车区内，比赛时间加罚5秒钟。对于未完成的任务会通过相应的加罚时间叠加在比赛时间上。

（3）传感器

摄像头，陀螺仪等

（4）车模

I车模

**1.3 报告内容**

在这份报告中,我们小组通过对小车设计制作的结构，电路，调试，车辆参数的介绍,尽力展现我们在制作调试过程中的思路和付出的汗水。具体表现在结构的设计制作,以及算法方面的思维想法。

1. **机械结构设计**

本章主要讲述我们在机械结构方面的设计与优化。

**2.1 整体布局**

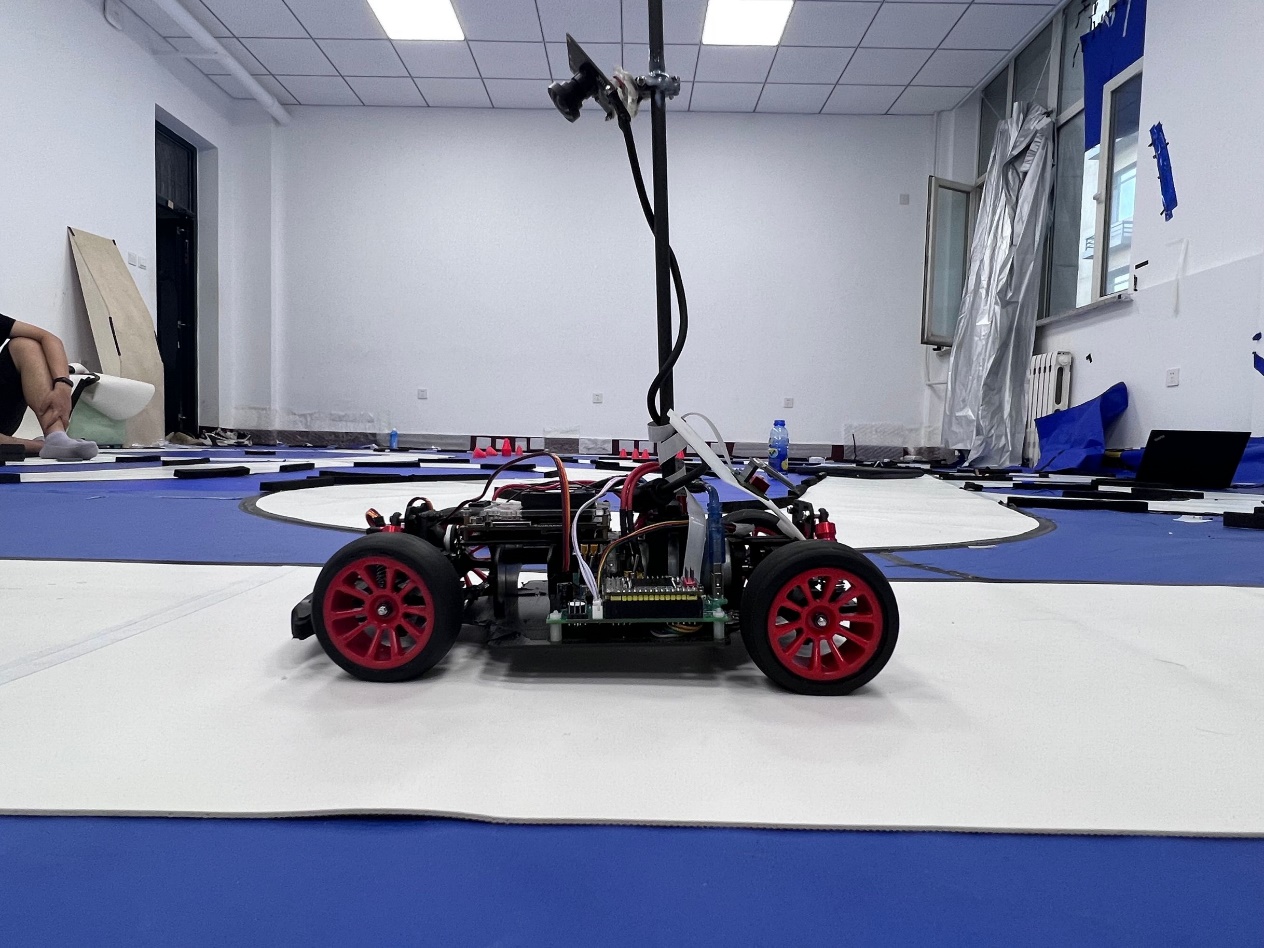


图2.1 车模整体结构

我们采用I车模推荐结构设计，考虑了电机与Edgeboard要摆放在中间的情况，主控与电源驱动安装在车体两侧，使其平衡车的重心，调试板放于车壳上，方便安装车壳后的调试。

**2.2** **转向舵机安装**

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，需要对舵机安装方式和位置进行研究和改进。通过分析舵机控制转向的原理可知，在相同转向条件下，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。

舵机安装方式有立式和卧式两种，通过比较得立式安装效果更优。 舵机安装时须保证左右对称，以保证舵机左右转向时力臂相等且处于最大范围，提升舵机响应速度。经理论分析，功率等于速度与扭矩的乘积，加大转向速度必然减少输出扭矩，扭矩过小会造成迟钝，故安装时须考虑到转向机构的响应速度与舵机扭矩之间的关系，获得最佳转向效果。

**2.3 前轮倾角调节**

为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。本系统所采用的智能车通过四只轮胎与地面接触，后轮同轴受到限位，无法调整，与智能车的前进方向保持平行，因此要改变智能车与地面的接触方式，调试出利于车转向、直线的四轮定位，只能通过调整前轮倾角各定位参数来实现。它的安装位置由前轮外倾和前轮前束两项目决定。

2.3.1 前轮外倾

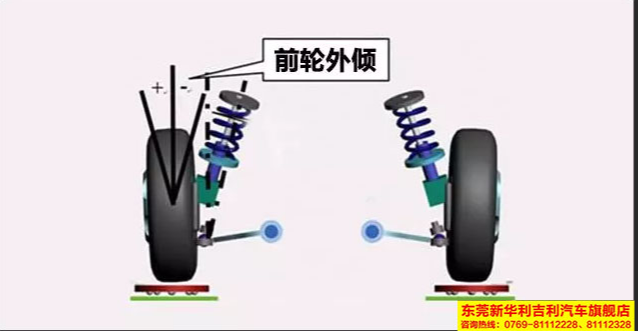
前轮外倾是指前轮安装后，其上端向外倾斜，于是前轮的旋转平面与纵向垂直平面间形成一个夹角，称之为前轮外倾角，其主要作用是使转向轻便，使车轮紧靠轮毂内轴承，以减少外轴承及轮毂螺母的负荷，有利于安全行驶。一般前轮外[倾角](https://baike.baidu.com/item/%E5%80%BE%E8%A7%92/3070523" \t "_blank)为1°左右，但对于有高速、急转向要求的车辆，前轮外倾角可减小甚至为负值。

图 2.2 前轮外倾

2.3.2 前轮前束

车轮前束如图2.3所示。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。从上往下看，两个车轮指向的方向在前端指向内称为车轮前束，指向外的则称为车轮后束。前轮前束的作用是保证汽车的行驶性能，减少轮胎的磨损。前轮在滚动时，其惯性力自然将轮胎向内偏斜，如果前束适当，轮胎滚动时的偏斜方向就会抵消，轮胎内外侧磨损的现象会减少。

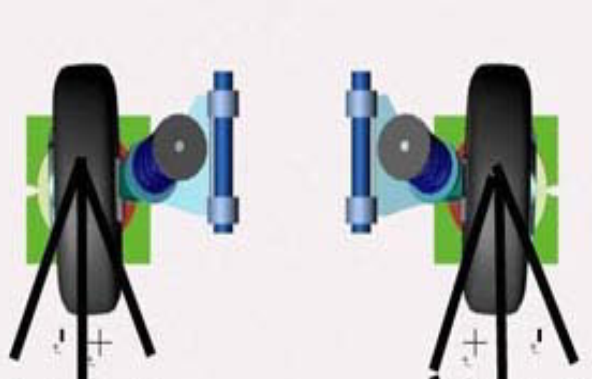


图2.3 前轮前束

**2.4悬挂调节**

通过调节悬挂，调整车模底板平行于地面，优化车模重心。

**2.5姿态传感器**

我们选用了龙邱科技公司的ICM-20602模块作为姿态传感器。ICM-20602是一款六轴陀螺仪模块，具有更低的温漂，支持SPI和IIC通信。

传感器尽可能地安装在车轮轴线上，以便于获得准确的姿态数据。

**2.6 摄像头**

我们使用了一个杰锐微通公司生产的GZ330工业模组摄像头，经过长时间的尝试，既能保证寻线的准确性，又可以保证模型的识别率。摄像头通过螺丝和AB胶固定在碳杆上，以确保车模在运行过程中摄像头的稳定性。

**2.7 车壳设计**

车壳长325.5mm、宽97.5mm、高190.8mm。此车壳是在I车模基础上，参考部分超跑车型设计而成。为了更好的利用空气动力学效应，车后加装了扩散器和尾翼。并在车前、车后和车顶打孔，以加强车内空气流通，可对舵机、电机、电路板、Edgeboard等器件起到散热作用。此外，为了适应调参和安装需要，分别在车壳中后部和车的两侧设计了三块活动窗口，并通过磁铁进行连接。

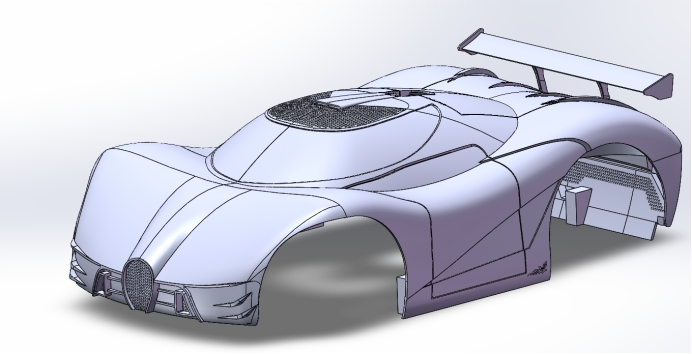


图 2.4 车壳

**2.8 车模主要技术参数**

智能车主要技术参数包括物理尺寸、电路指标等，具体参数见表 2.1。

表2.1 智能车主要技术参数

|  |  |  |
| --- | --- | --- |
| 模型车基本参数 | 长 | 32 cm |
| 宽 | 18 cm |
| 高 | 28 cm |
| 车重 | 2216g |
| 电池容量 | | 2700mah |
| 传感器 | 编码器 | 1个 |
| 彩色摄像头 | 1个 |
| 陀螺仪 | 1个 |
| 除了车模原有的驱动电机、舵机之外伺服电机个数 | | 0个 |
| 微控制器型号和数量 | TC264 | 1个 |

1. **硬件电路的设计**

这辆小车电路部分主要的模块包括：电源管理模块、传感器模块、驱动模块以及其他调试模块。各模块的总体设计原则是：稳定可靠且符合车模自身结构。

**3.1 电源管理模块设计**

电源输出的首要指标是可靠性，整个硬件系统的工作（无论是Edgeboard板卡、英飞凌单片机还是电机等等）完全由电源供电的可靠性决定，电源供电不稳定会引起损耗、单片机复位、传感器损毁等严重问题，尤其Edgeboard板卡的成本较高，因此对于我们组别来说电源输出的稳定性极为重要。

我们采用格氏3S锂离子动力电池为整个系统供电，电池电压保持在11.7~12.6V，电池连接在电源驱动板上通过三个接口分别向电机、Edgeboard板卡、主控板供电，其中未避免电机驱动带来的高频干扰带来的不良影响，我们在Edgeboard板卡与主控的电源输入之前加入了共模滤波器进行滤波，并在 Edgeboard板卡供电前端加入保险丝，以保障板卡的使用安全。

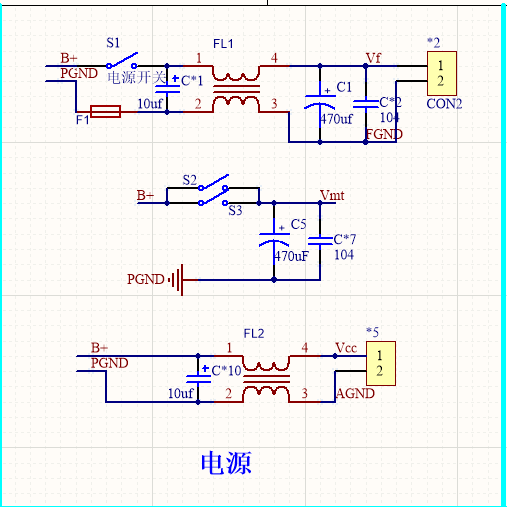


图 3.1电源管理原理图

此外我们通过ADC采集电池的电压，在电池电压低于11.7V蜂鸣器蜂鸣示警，以保证电池的正常使用、板卡的正常工作。

|  |  |
| --- | --- |
|  |  |
| ADC采样电路 | 蜂鸣器驱动电路 |

**3.2 驱动电路设计**

IR2104＋MOS管搭建的H 桥电路直接通过电池供电作为电机驱动电路，足以满足我们对于电机控制的要求，我们选用了IPB02606N作为驱动用的MOS管（该型号MOS，最大耐压60V，通流能力100A），实现了简单控制转速、实现转速闭环的需求，下面是电路原理图。

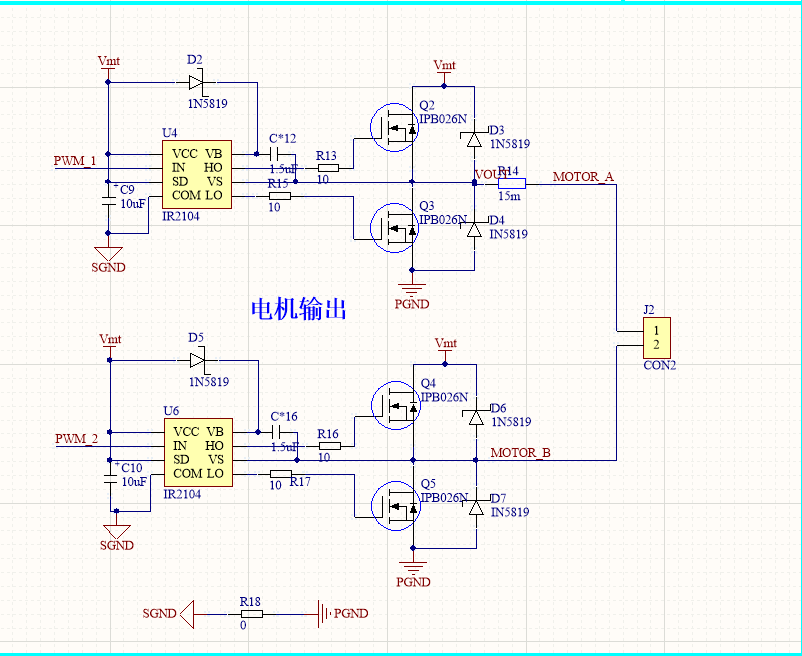
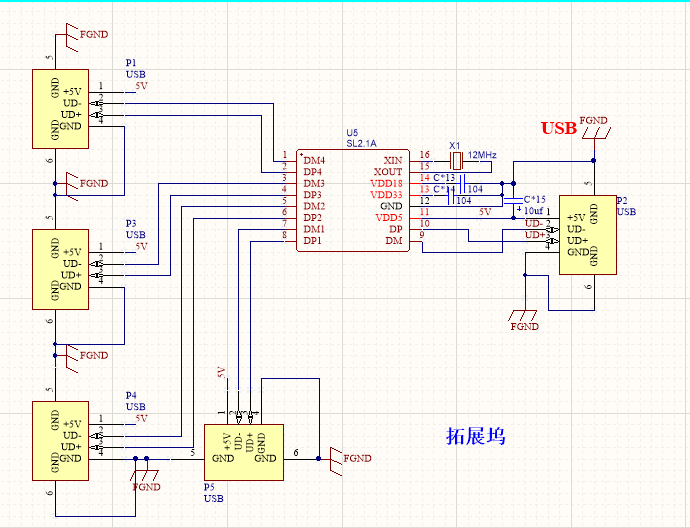


图 3.2 电机驱动电路

**3.3 拓展坞电路设计**

为了满足板卡同时与单片机、摄像头、无线网卡交互的要求，我们使用SL2.1芯片设计一个拓展坞电路，电路原理图如下。

图 3.3 拓展坞电路图

**3.4 调试电路设计**

调试电路在车模调试过程中是必不可少的，可以直观的显示智能车的参数和状态，并且可以方便地修改参数。调试电路包括：五向按键、OLED接口、拨码。由于完全模型组别车模必须装有车壳，因此我们将其单独放在一个小PCB板子粘在车壳上，提高了我们调试的效率。

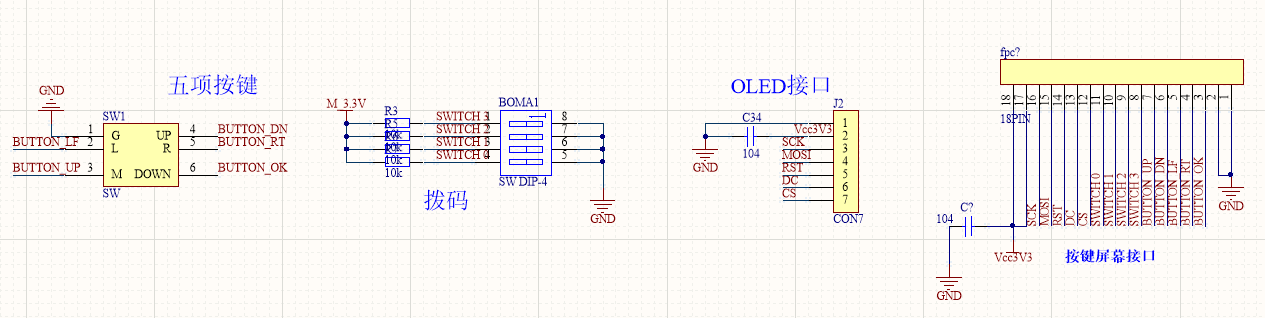


图 3.4 调试电路

**3.5 MCU控制电路**

车模的控制部分由英飞凌单片机（我们采用了龙邱科技的TC264DA核心板）完成，他负责向各个外设输出各种控制信号、从陀螺仪等传感器获取信息，并完成板卡图像处理数据的接收，对于小车来说其作用至关重要。

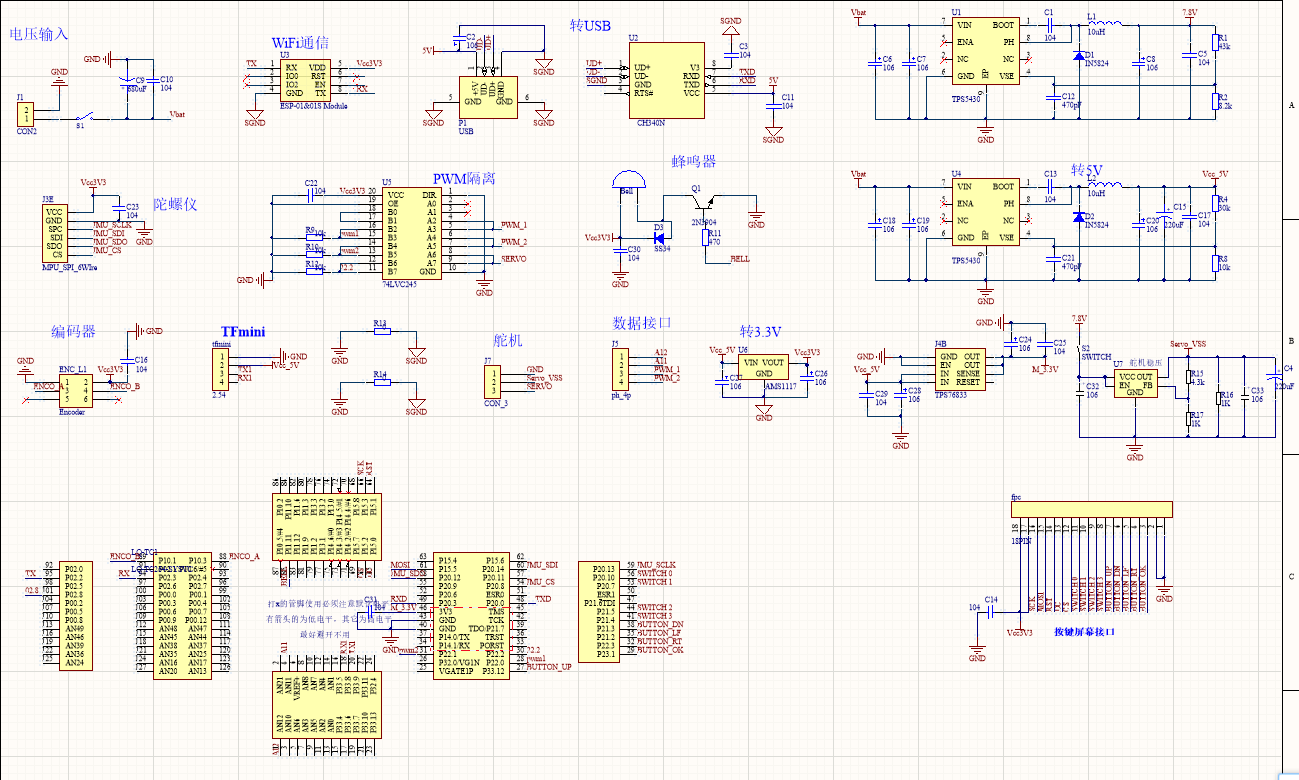


图 3.5 主控电路原理图

此外，我们在主控板上进行了各种电源变换给单片机以及其他外设供电，从电池输入的电源经过两个TPS5430转压电路分为转为5V和8V，5V分别通过TPS76833转压电路与AMS1117转压电路转为3.3V分别为单片机和编码器、陀螺仪等传感器供电 ；8V通过MIC29302将电压降为6.6V给舵机稳定供电。

1. **软件方案**

高效稳定的软件程序是智能车平稳快速循迹的基础。我们设计的智能车使用红外摄像头来获取信标位置信息，采用鲁棒性很好的PID算法控制智能车的直立、转向和速度控制，使得在寻灯的过程中智能车达到快速的效果。

**4.1** **图像采集与处理**

本智能车采用的130度RGB摄像头。在FPGA对图像进行采集后对其进行处理，以提取主干赛道中心线。考虑到FGPA强大的算力以及多样的内置环境，本款程序采用openCV进行辅助提取，通过色域提取、动态阈值、图形拟合等多种方式，较传统巡线算法更快更准得提取出主干赛道中心线。  
在图像处理中，我们提取的信息主要包括：赛道主干信息、模型图案、赛道外部信息、锥桶信息等等。  
 当场地光线足够理想时，通过固定阈值二值化，可将赛道与蓝色底布区分开，再通过连通域提取，即可精确地收集到赛道的所有信息。当场地光线不够理想时，我们采用动态阈值提取的方式，可将图像中光照不均匀、不充分的部分很好地与蓝色底布区分。考虑到我们收集的赛道信息需要对红色、蓝色以及白色进行区分，色域提取在图像预处理中占据了主要部分。  
 主体巡线采用骨架提取与补线相互辅助，部分区域进行特殊处理。

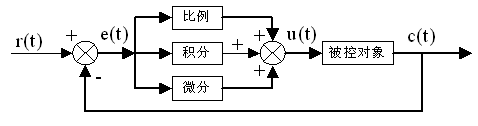
**4.2 神经网络模型识别**

首先通过前摄像头采集的图像进行图像处理，来判断前方是否存在标志。如果前方发现存在标志，将采集的图片进行剪裁后传给模型，利用模型去区分加油站，施工，泛行标志等等。  
 在对比了多种模型后，如yolo系列，fast系列，ssd系列，我们发现ssd\_moblient\_v1在速度，准确率方面均表现优良，因而最终采用改模型进行标志的识别。同时为了加大模型识别的准确率，我们对数据集进行了一定的增强，使其能够适应不同的光照，模糊，放缩等变化。

**4.3 PID运动控制算法**

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

PID控制器是一种线性控制器，它根据给定值与实际输出值构成控制偏差。将偏差的比例(P)、积分(I)和微分(D)通过线性组合构成控制量，对被控对象进行控制，故称PID控制器，原理框图如图5.1所示

图5.1 PID控制器原理框图

在计算机控制系统中，使用的是数字PID控制器，控制规律为：

 (公式5.1)

 (公式5.2)

式中

k——采样序号，k = 0，1，2…； r(k)——第k次给定值；

c(k)——第k次实际输出值； u(k)—— 第k次输出控制量；

e(k)—— 第k次偏差； e(k-1)—— 第k-1次偏差；

KP——比例系数； TI——积分时间常数；

TD——微分时间常数； T——采样周期。

简单说来，PID控制器各校正环节的作用如下：

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

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

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

数字PID控制算法通常分为位置式PID控制算法和增量式PID控制算法。

**4.3.1 位置式PID**

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

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

**4.3.2 增量式PID**

当执行机构需要的是控制量的增量(例如：驱动步进电机)时，可由式(5.2)

推导出提供增量的PID控制算式。由式(5.2)可以推出式(5.3)，式(5.2)减去式(5.3)可得式(5.4)。

………5.3

 ……………5.4

式中；；

公式(5.4)称为增量式PID控制算法，可以看出由于一般计算机控制系统采用恒定的采样周期T，一旦确定了KP、KI 、KD，只要使用前后三次测量值的偏差即可由式(5.4)求出控制增量。

增量式PID具有以下优点：

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

(2) 手动/自动切换时冲击小，便于实现无扰动切换。此外，当计算机发生故

障时，由于输出通道或执行装置具有信号的锁存作用，故能保持原值。

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

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

**4.3.3 PID参数整定**

运用PID控制的关键是调整KP、KI、KD三个参数，即参数整定。PID参数的整定方法有两大类：一是理论计算整定法。它主要是依据系统的数学模型，经过理论计算确定控制器参数；二是工程整定方法，它主要依赖工程经验，直接在控制系统的试验中进行，且方法简单、易于掌握，在工程实际中被广泛采用。由于智能车系统是机电高耦合的分布式系统，并且要考虑赛道的具体环境，要建立精确的智能车运动控制数学模型有一定难度，而且我们对车身机械结构经常进行修正，模型参数变化较为频繁，理论计算整定法可操作性不强，最终我们采用了工程整定的方法。

**4.4 舵机控制（pd控制）**

调节舵机占空比，观察舵机左右转动情况确定舵机左右极限值；初略设置舵机中值，占空比固定，车辆放置于直道行进，根据观察得行进路线进行微调，确定中值。在行驶中，运用pid将巡线偏差给到舵机占空比上实现方向控制。为适应不同速度、偏差大小等因素，我们采用一维模糊PID，通过不同段偏差对应不同的PID参数，我们车辆适应性和稳定性得到提升。

**4.5 电机控制（pi控制）**

通过调节给电机驱动输出的两路PWM信号的占空比来控制电机的转速，通过编码器采样得到电机的实际转速，通过恰当的比例控制与积分控制（不断将偏差输入到PI控制器，控制器输入不同的占空比给电机驱动）来使得转速环较硬能够满足我们的需求。

1. **系统的开发环境与车模调试**

在系统的设计制作和调试的过程中，不管是软件的开发还是硬件电路的仿真和电路板的制作都离不开PC 机。所以对于PC 机上的各种辅助设计软件必须要有一定的熟悉程度。这样可以提高开发的效率。

**5.1 EdgeBoard的调试**

我们主要采用了WinSCP以及XShell来进行Edgeboard的调试。

WinSCP是一款流行的SFTP客户端和FTP客户端!使用FTP、FTPS、SCP、SFTP、WebDAV或S3文件传输协议在本地计算机和远程服务器之间拷贝文件。

XShell可以在Windows界面下来访问远端不同系统下的服务器，从而比较好的达到远程控制终端的目的。它支持 RLOGIN、SFTP、SERIAL、TELNET、SSH2 和 SSH1，可以非常方便的对Linux主机进行远程管理。

此外，我们还应用了外接显示屏显图，可以用来调试图像。

**5.2 下位机的调试**

5.2.1人机交互

在调试过程中需要不断地修改变量的值来达到整定参数的作用，对此我们选用了液晶屏配合按键和拨码开关的调试方法。此外，比赛的时候，修改参数我们同样用这个模块进行修改。

其中液晶屏我们选用 OLED 液晶，该液晶具尺寸小，高分辨率等特点。界面如图6.2所示。按键采用五向按键其便于操作并且节省空间。设计五向按键以及4个拨码开关进行调参，显示，方案选择等部分。



图 6.1 液晶屏显示界面

5.2.2 开发调试工具

运用AURIX Development Studio作为软件开发工具。ADS是英飞凌公司于2019年底推出的免费的集成开发环境，支持英飞凌TriCore™内核AURIX™ 系列MCU；ADS是一个完整的开发环境，包含了Eclipse IDE、C编译器、Multi-core调试器、英飞凌底层驱动库（low-level driver,iLLD),同时对于编辑、编译及调试应用代码没有时间及代码大的限制。调试界面如图所示：

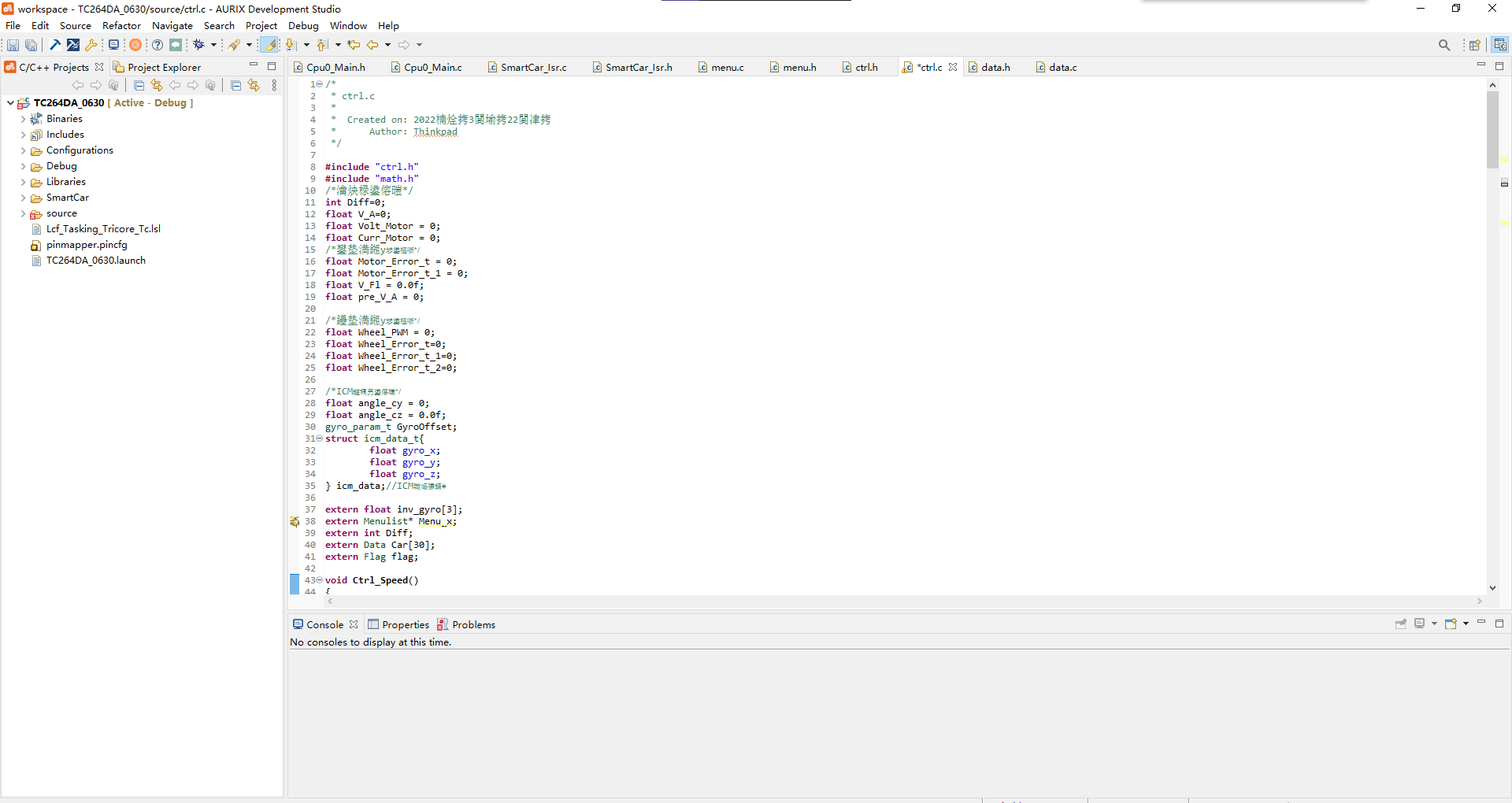


图 6.2 调试界面

**5.3 WIFI及上位机调试**

车在同样的赛道上走过的路都是不一样的，所以无论怎么考虑车的状态都是不够完全的，因此需要对运行中的车进行实时监控。为了解决这个问题，我们使用WIFI模块配合上位机进行实时观测车模运行状态。如图6.3所示。

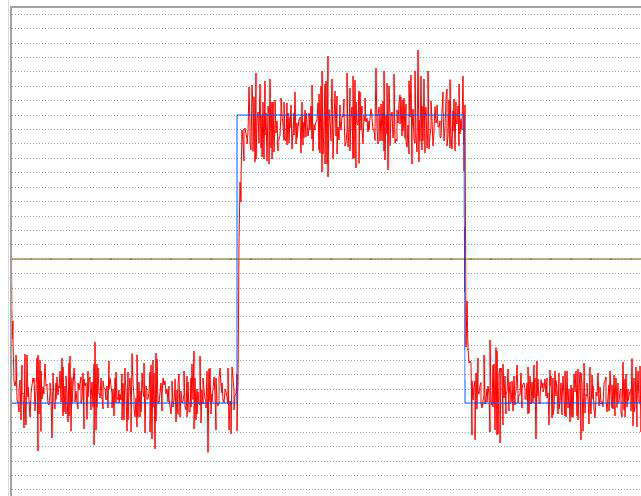


图 6.3 上位机示波器

**第六章 总结**

**6.1车模总结**

**6.1.1控制方面**

采用PID进行速度和转向控制，利用摄像头进行寻线以及模型识别。

**6.1.2结构方面**

尽力做到了结构的合理性与简洁性，将车模的重量和重心尽可能合理化。

电路板均为自己设计与焊接，完成了供电，采集，控制等各项任务。

**6.1.3不足之处**

车模的结构还有较大的优化空间。

**6.2工作综述**

在比赛的准备期间,我们小组成员涉猎机械学，电力电子学，控制科学，传感器应用，计算机科学等多个学科。通过数月的智能车制作，我们学到了很多实践能力，从方案选择、更改、重新设计到实施方案，细调参数，机械和硬件制作，软件编写，各方面都有深刻的体会，每一个细致的工作都会为小车这个系统的稳定性带来正面的影响，而每个疏漏，都会导致整个系统的不协调。这次比赛对我们的知识融合和实践动手能力的培养有极大的推动作用,同时也加深了队友之间的感情，为自己的未来添砖加瓦。我们感谢英飞凌公司提供了如此一个好的平台来让我们展现自我，小车在赛道上的驰骋，无疑也是我们那颗年轻的心的驰骋!

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

**致谢**

借书写技术报告的机会向帮助过我们的指导老师，哈工大智能车社团的学长们，哈工大学校、学院领导以及组委会的老师，表示深深地谢意！

更要向队内的队友表示深深地感谢，感谢陪伴在一起的日日夜夜！通过参赛，我们学会了很多！

最后，向没有机会出现在队员名单中的幕后英雄致敬！

**参考文献**

[1] 卓晴.第十七届全国大学智能汽车竞赛竞速比赛规则[EB/OL].

https://zhuoqing.blog.csdn.net/article/details/121187014

[2] 王盼宝. 智能车制作: 从元器件、机电系统、控制算法到完整的智能车设计[M]. 北京: 清华大学出版社, 2017.

[3] 王淑娟，蔡惟铮，齐明．模拟电子技术基础 [M]．北京：高等教育出版社．2009

[4] 卓晴，黄开胜，邵贝贝．学做智能车 [M]．北京：北京航空航天大学出版社．2007.

**附录**

**主要程序：#include** "Ifx\_Types.h"

**#include** "IfxCpu.h"

**#include** <ifxCpu\_Irq.h>

**#include** "IfxScuWdt.h"

**#include** "SmartCar\_Uart.h"

**#include** "SmartCar\_Upload.h"

**#include** "SmartCar\_PIT.h"

**#include** "SmartCar\_ADC.h"

**#include** "common.h"

**#include** "menu.h"

**#include** "ctrl.h"

**#pragma** section all "cpu0\_dsram"

//IfxCpu\_syncEvent g\_cpuSyncEvent;

Menulist\* Menu\_x = NULL;

Flag flag;

Data Car[30];

Commun\* Rx\_data = NULL ;

uint8 sector;

**int** r\_count = 0;

**extern** **float** Volt\_Motor;

**extern** **float** Curr\_Motor;

**extern** int32 dist;

**int** **core0\_main**(**void**)

{

/\*\* 关闭总中断\*/

IfxCpu\_disableInterrupts();

/\*\* 初始化时钟\*/

get\_clk();

/\* !!WATCHDOG0 AND SAFETY WATCHDOG ARE DISABLED HERE!!

\* Enable the watchdogs and service them periodically if it is required

\*/

IfxScuWdt\_disableCpuWatchdog(IfxScuWdt\_getCpuWatchdogPassword());

IfxScuWdt\_disableSafetyWatchdog(IfxScuWdt\_getSafetyWatchdogPassword());

/\* Wait for CPU sync event \*/

IfxCpu\_emitEvent(&g\_cpuSyncEvent);

IfxCpu\_waitEvent(&g\_cpuSyncEvent, 1);

/\*\* 初始化OLED\*/

SmartCar\_Oled\_Init();

/\*控制相关初始化\*/

Ctrl\_Init();

SmartCar\_OLED\_Fill(0);

/\*ADC通道初始化\*/

ADC\_Init(*ADC\_0*, *ADC0\_CH11\_A11*);

ADC\_Init(*ADC\_0*, *ADC0\_CH12\_A12*);

/\*Uart初始化\*/

//SmartCar\_Uart\_Init(IfxAsclin3\_TX\_P15\_6\_OUT,IfxAsclin3\_RXA\_P15\_7\_IN,115200,3);//USB连接板卡通信

//SmartCar\_Uart\_Init(IfxAsclin3\_TX\_P20\_0\_OUT,IfxAsclin3\_RXC\_P20\_3\_IN,115200,3);//USB连接板卡通信

SmartCar\_Uart\_Init(IfxAsclin2\_TX\_P33\_9\_OUT,IfxAsclin2\_RXE\_P33\_8\_IN,115200,2);//tfmini

SmartCar\_Uart\_Init(IfxAsclin1\_TX\_P02\_2\_OUT,IfxAsclin1\_RXG\_P02\_3\_IN,115200,1);//预留wifi（921600）或单片机向板卡发送接口（115200），

/\*定时中断初始化\*/

Pit\_Init\_ms(*CCU6\_0*, *PIT\_CH0*, 4);

Pit\_Init\_ms(*CCU6\_0*, *PIT\_CH1*, 15);

/\*接受数据链表创建\*/

Rx\_data = List\_Create(3);//链表数据

/\*菜单初始化\*/

Menulist\* Menu = NULL;

Menu = MenuInit();

Menu\_x = Menu;

/\*扇区选取\*/

**if**(GPIO\_Read(P20,10) == 0)

{

**if**(GPIO\_Read(P20,7) == 0)

sector = 1;

**else** sector = 2;

}

**else** {

**if**(GPIO\_Read(P20,7) == 0)

sector = 3;

**else** sector = 4;

}

/\*读取Flash\*/

**for**(**int** temp = 0; temp<27;temp++){

Car[temp].Fl = Page\_Read(sector,temp,**float**);

}

Car[27].In = Page\_Read(sector,27,uint32);

Car[28].In = Page\_Read(sector,28,uint32);

Car[29].In = Page\_Read(sector,29,uint32);

flag.a = 0;//环岛flag

flag.b = 0;//延时flag

flag.c = 0;//状态蜂鸣器

flag.d = 1;//出库

flag.sit\_t = 0;

/\*\* 开启总中断\*/

IfxCpu\_enableInterrupts();

/\*\* 主循环 \*/

**while**(TRUE)

{

**if**(Volt\_Motor <11.5){

GPIO\_Set(P11,11,1);

}

**else** GPIO\_Set(P11,11,0);

**if**(flag.b == 0 && !GPIO\_Read(P21,4)){

Delay\_ms(*STM0*,3000);

flag.b = 1;

}

Menu\_x = Button(Menu\_x);

// SmartCar\_Uart\_Transfer(&Car[28],1,1);

}

}

IFX\_INTERRUPT(**cc60\_pit\_ch0\_isr**, 0, CCU6\_0\_CH0\_ISR\_PRIORITY)

{

enableInterrupts();//开启中断嵌套

Tf\_Callback();

**if**((dist<42 && dist > 2)||(r\_count> 55)){

V\_E = 0;

}

**if**(flag.b){

Ctrl\_Speed();

r\_count++;

}

Volt\_Motor = ((**float**)(ADC\_Get(*ADC\_0*, *ADC0\_CH11\_A11*, *ADC\_12BIT*)))\*3.346\*4.05/(4095);

Curr\_Motor = (((**float**)(ADC\_Get(*ADC\_0*, *ADC0\_CH12\_A12*, *ADC\_12BIT*)))\*3.3/(4095\*0.5)) - 1.8/0.5;

PIT\_CLEAR\_FLAG(*CCU6\_0*, *PIT\_CH0*);

}

IFX\_INTERRUPT(**cc60\_pit\_ch1\_isr**, 0, CCU6\_0\_CH1\_ISR\_PRIORITY)

{

enableInterrupts();//开启中断嵌套

Rupt\_1();

PIT\_CLEAR\_FLAG(*CCU6\_0*, *PIT\_CH1*);

}

**#pragma** section all restore