
第十二届“恩智浦”杯全国大学生智能汽车竞赛

技 术 报 告



中南民族大学

SOUTH-CENTRAL UNIVERSITY FOR NATIONALITIES

学 校：中南民族大学

队伍名称：中南民大麒麟队

参赛队员：谭学乾 许大新 张静 叶梓荣

带队老师：朱宗晓 孔德龙

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有关保留、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：

带队教师签名：

日 期：

目录

第一章 引言.....	1
第二章 智能车整体设计.....	2
2.1 系统机械结构设计与调整.....	2
2.2 控制简述.....	3
第三章 机械设计与修改.....	4
3.1 前轮定位.....	4
3.2 地盘高度的调整.....	4
3.3 舵机安装结构的调整.....	4
3.4 电磁传感器安装.....	5
3.5 其他机械结构的调整.....	5
第四章 传感器选择和布局.....	6
4.1 磁场传感器选择.....	6
4.2 电磁感应线圈在磁场中的特性.....	7
4.3 磁场传感器布局.....	9
4.3.1 确定导线位置布局.....	9
4.3.2 前瞻设计.....	9
4.4 编码器的安装.....	10
第五章 硬件电路模块设计.....	11

5.1 K60 控制器模块.....	11
5.2 前瞻传感器模块.....	12
5.3 电源控制模块.....	14
5.4 测速反馈模块.....	16
5.5 电机驱动模块.....	16
5.5.1 BTS7960 电路连接.....	16
5.6 LED 数码管显示与键盘模块.....	17
5.7 起跑线检测模块.....	18
5.8 超声波测距模块.....	18
5.9 舵机驱动电路模块.....	19
5.10 系统主板电路的设计.....	19
5.11 硬件电路的小结.....	20
第六章 软件部分设计.....	21
6.1 整体框架.....	21
6.2 舵机控制流程.....	22
6.3 速度控制.....	23
6.4 PID 算法.....	23
6.4.1 模拟 PID 算法.....	24
6.4.2 PID 参数作用.....	24
6.4.3 PID 算法的数字实现.....	24

第七章 开发与调试.....	26
7.1 开发环境.....	26
7.2 程序下载.....	31
7.3 测试环境.....	32
第八章 总结.....	33
8.1 制作成果.....	33
8.2 问题与思考.....	33
8.3 不足与改进.....	33
参考文献.....	34
附录 A 智能车整车图片.....	35
附录 B 部分程序原代码.....	36

第一章 引言

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，受教育部高等教育司委托(教高司函[2005]201号文，附件1)，由教育部高等学校自动化专业教学指导分委员会主办全国大学生智能汽车竞赛。

我校为响应教育部的号召，积极组队参加第十二届“恩智浦”杯全国大学生智能汽车竞赛。在老师与学长的指导下，我们集继承与创新于一体，设计理念的不断进步，制作精度的不断提高，最终设计出一套完整的智能车开发、调试平台。作为电磁组的中南民族麒麟队秉承简洁美观的优良传统，采用轻质量、低重心的机械设计。在前瞻距离、噪声抑制、驱动优化、整车布局等方面加强研究创新，最终使智能小车获得了良好的综合性能和赛场表现。

本文将对智能车的总体设计和各部分的详细设计进行逐一介绍。

第二章 智能车整体设计

根据电磁车循迹的要求，电磁车设计包括赛车的设计和恒流源的设计。

根据电磁传感器方案设计，赛车共包括五大模块：道路检测传感器模块、速度传感器、CPU模块，电机驱动模块、电源管理模块。系统框图如图2.1所示。

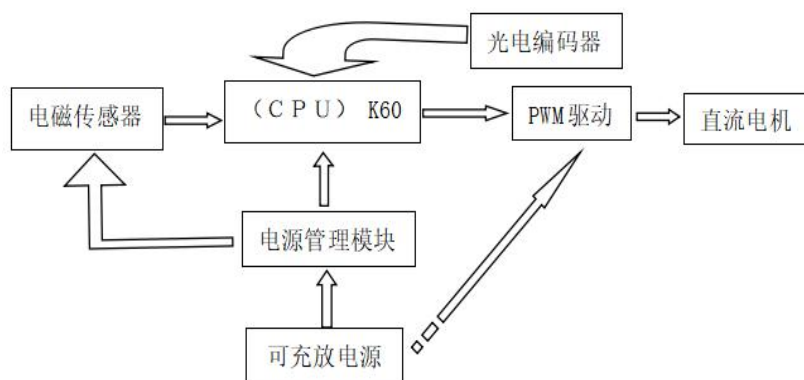


图 2.1 系统结构框图

2.1 系统机械结构设计与调整

智能车的机械设计与调整是否合适，对于又快又稳的寻迹运动要求具有非常关键的作用。

鉴于赛道的摩擦力和复杂性，我们提出在不损害赛道的前提下尽可能地降低车子的重心，因为车子的重心高低和位置对车子的行驶、转弯和提速都有很大的影响，尤其是提速的时候效果非常明显。重心不合适的车子在速度提升时很容易翻车，过弯道也过的不顺容易甩出赛道，这是我们在实验室调试的时候都遇到过的。为了尽可能的降低重心，我们决定先降低地盘的高度，使用的是PCB板把底盘降到我们需要的高度。同时在电池安装方面，我们也是尽可能使电池紧贴着车子底盘安装，一切都是为了尽最大努力降低重心，为后面的提速打下基础。

由于电磁组的电磁信号较为单一，从采集到的信号当中获取的信息要远远少于摄像头组，所以为了获得更好的前瞻性，我们将传感器用支架尽可能地向前延伸，事实证明这样做在提速时是非常有必要的，它可以提前对弯道或其他特殊赛道元素进行判断，这使得车子的转向性能更好，这样就达到了我们预期想要的效果。

关于前瞻支架的选择以及安装方式的选择我们尝试很多方案，最后我们采用的是强度高、重量轻、加工固定方便的支架方案，使得车子的转动惯量达到最小化，从而使车子在转弯时更加迅速灵敏。

最后是车子闭环控制的问题，我们采用的是龙邱科技512线的编码器，支架是自己根据车子量身定做的，可以直接使编码器齿轮咬合车模主动轮的方法来获得小车的速度。最后在程序中根据编码器采集回来的信号设计速度控制策略来实现车速的闭环控制。

- 麒麟队主体部分：
- 底盘及其附属部分：驱动电机、转向舵机、电池等等；
- 信号检测部分：前瞻；
- 编码器测速部分；
- 通信部分：超声波、NRF；
- 单片机最小系统及其相应电路，包括电源处理及检波；
- 机械制作：传感器支架，编码器支架、舵机支架等。
- 干簧管停车及键盘调试模块。

2.2 控制简述

由于传感器感应到的信号只有几十毫伏左右，所以我们采用三级管放大电路进行放大，然后将其检波为直流电平送入K60的AD口。与此同时，512线编码器将采集到的信号也同步输入，有K60的输入捕捉通道进行上升沿或者下降沿计数从而获得车子行驶的速度。

舵机和电机采用的控制算法都是经典的P I D原理控制的算法，首先是将采集到的数据进行整理，再经过长时间的参数整定来达到预期效果。

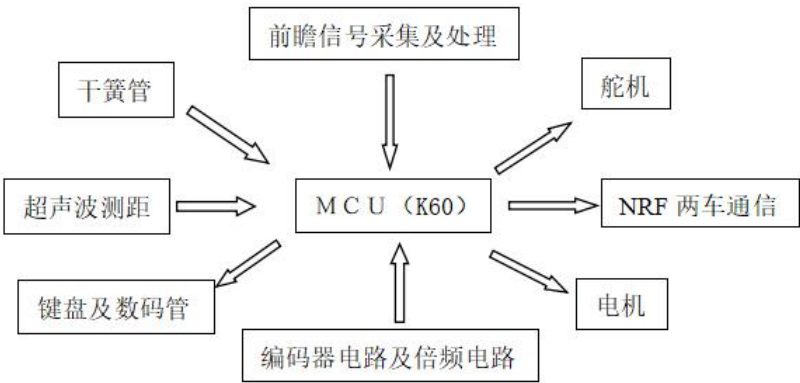


图2.2芯片控制图

第三章 机械设计与修改

3.1 前轮定位

为了使模型车在直线行驶时更加稳定，转向轻便，转向后能自动回正，并减少轮胎和转向系零件的磨损等，在转向轮、转向节和前轴之间形成一定的相对安装位置，叫车轮定位。其中包括主销后倾、主销内倾、车轮外倾和前束。

在调试过程中，我们发现由于前轮轴和车轮之间的间隙较大，对车高速时转向中心的影响较大，会引起高速转向下模型车的转向不足。然而这里是规则中严禁改动的地方。所以为了尽可能降低转向舵机负载，我们对前轮的安装角度，及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、主销和前轴等三者车架上的位置关系。

在实际的调试中，我们发现适当增大内倾角度的确可以增大转弯时车轮和地面的接触面积，从而增大了地面的摩擦程度，是车转向可更加灵活，减少因摩擦不够而引起的转向不足的情况。

3.2 地盘高度的调整

在保证顺利通过坡道和障碍的前提下，底盘尽量降低，从整体上降低车模的重心，可使模型车转弯时更加稳定、高速。

3.3 舵机安装结构的调整

原装车模的舵机为卧式安装，考虑到主板的安装方便以及车模转向性能，我们对舵机安装结构进行了较大的调整。比赛车模转向是通过舵机带动左右横拉杆实现的。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，必然损失力矩，力矩太小也会使转向迟钝。因此在设计时就要综合考虑。

关于舵机的安装方式，我们经过直立式和卧式安装的实验比较，最终选择直立的安装方式。

3.4 电磁传感器安装

本设计方案中的电磁传感器为三个工字型的10毫亨电感。为了能够更好的检测前面的道路，我们将这三个工字型的电感用较长的前瞻伸向车模的前方。

由于电磁线圈在车模的最前方，在车模失控之后跑出赛道撞击其它物体，最先遭到破坏的就是电磁感应线圈。为了保护电磁感应线圈，我们把向前伸出的杆有意延长，加上防护套，用以保护电磁线圈。

3.5 其他机械结构的调整

- 1 为提高小车的稳定性，我们尽量做到使小车重心降低。
- 2 系统板和电池尽量往后靠近电机，这样做可以使重心的分布更合理，提高车模的转向性能。

第四章 传感器选择和布局

4.1 磁场传感器选择

由于赛道铺设20KHz交变电流的导线，因此需要通过检测导线周围所产生的电磁场确定道路与小车的相对位置。磁场传感器利用了物质与磁场之间的各种物理效应，如磁电效应（电磁感应，霍尔效应，磁致电阻效应），磁机械效应，核磁共振等。现代检测磁场的传感器有很多，常见的有磁通门磁场传感器，磁阻抗磁场传感器，半导体霍尔传感器、磁敏二极管，磁敏三极管。因为各种传感器测量磁场所依据的原理不相同，测量的磁场精度和范围相差也很大， $10^{-11} \sim 10^7 \text{G}$ 。图4.1为各类磁场传感器的测量范围示意图。

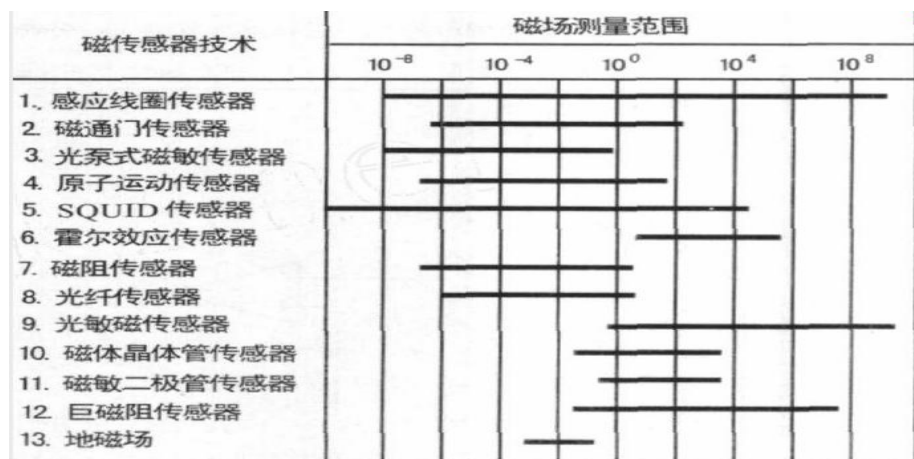


图 4.1 各类磁场传感器的测量范围示意图

先估算赛道的磁场强度。把赛道看作无限长直导线，载流为直流100mA，距离导线 $r=5\text{cm}$ 时，由毕奥-萨伐尔定律知，磁场强度 B 如（公式1）：

$$B = \frac{\mu_0 I}{2\pi r} = \frac{4\pi \times 0.1}{2\pi \times 0.05} \times 10^{-7} = 4 \times 10^{-7} \text{ T} = 4 \times 10^{-3} \text{ G} \quad (\text{公式1})$$

一般霍尔元件的检测范围在1mT以上，即10G以上，可以想象到需要贴着地面进行检测，而且精度大大受到限制。

磁阻传感器如Honeywell的高灵敏度磁阻HMC1001，分辨率可达27微高斯，还可以使用多轴的磁阻传感器检测不同方向的磁场。

普通的电感线圈测量范围广，理论上只要加上合适的谐振电容和放大电路，不但能够筛选出特定频段进行放大，而且有较强的抗干扰能力。

我们需要选择适合车模竞赛的检测方法，除了检测磁场的精度之外，还需要对于检测磁场的传感器的频率响应、尺寸、价格、功耗以及实现的难易程度进行考虑。由于霍尔元件和磁阻传感器的检测精度比较低，价格比较高。

我们选取最为传统的电磁感应线圈的，它具有原理简单、价格便宜、体积小、频率响应快、电路实现简单等特点。感应线圈可以自行绕制，也可以采用市面上的工字电感。

4.2 电磁感应线圈在磁场中的特性

为了讨论方便，我们作以下约定：

(1) 小车车体坐标系中，定义小车前进的方向为Y轴正向，顺着Y轴的右手边为X轴的正向，Z轴指向小车正上方，如图4.2所示；

(2) 水平线圈是指轴线平行于Z轴的电感线圈，垂直线圈是指轴线平行于X轴的线圈，轴线平行于Y轴的线圈所感应到的电动势远小于上述两类线圈，但该类摆放线圈在回环路检测中将可以用到。

(3) B_X 是指向载流导线右手边的电磁感应强度， B_Z 是指向载流导线正上方的电磁感应强度。显然，垂直线圈感应的是 B_X 变化率，水平线圈感应的是 B_Z 的变化率。

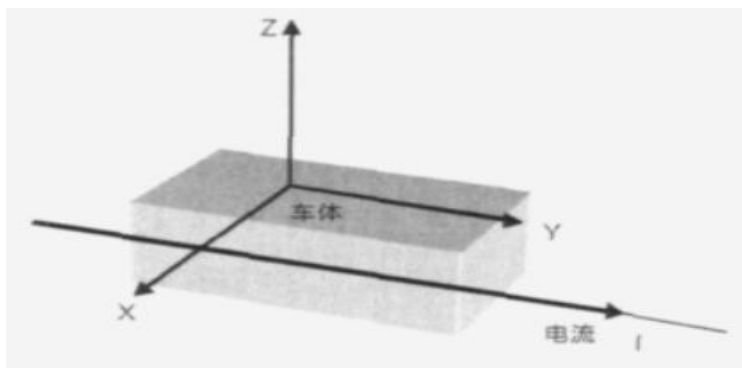


图 4.2 假定车体坐标系

直道附近的磁场分布，可以近似为无限长的直导线上的磁场分布，容易算得距离长直导线距离为 r 的点的磁感应强度如（公式2）：

$$B = \frac{\mu_0 I}{2\pi r} \quad (\text{公式2})$$

进而可以推出：

$$B_x = \frac{\mu_0 I}{2\pi} \times \frac{h}{x^2 + h^2} \quad (\text{公式3})$$

$$B_z = \frac{\mu_0 I}{2\pi} \times \frac{x}{x^2 + h^2} \quad (\text{公式4})$$

其中h是电感线圈距离地面的垂直距离。为了讨论的方便，记

$$B'_x = \frac{h}{x^2 + h^2} \quad (\text{公式5})$$

$$B'_z = \frac{x}{x^2 + h^2} \quad (\text{公式6})$$

则从（公式3）、（公式4）、（公式5）、（公式6）可以得出B' X、B' Z分别和BX、BZ有相同的变化趋势。图4. 3和图4. 4显示了当分别取h为5、8、10时B' X和B' Z的变化趋势。由图可知：

(1) B' X是x的偶函数，在Y轴两侧单调；B' Z是x的奇函数，在Y轴两侧没有单调关系；

(2) 在相同的高度下，B' X幅值是B' Z的两倍，但是在x=20的时候，B' X只有B' Z的一半左右了，因此B' X的衰减较B' Z快很多。综上可推知，水平线圈比较适合做x的正负判别，垂直线圈比较适合用来解算x的具体数值，B' Z较B' X衰减慢得多，说明水平线圈对远处道路状况相对比较敏感，可以用来预测前方的弯道。

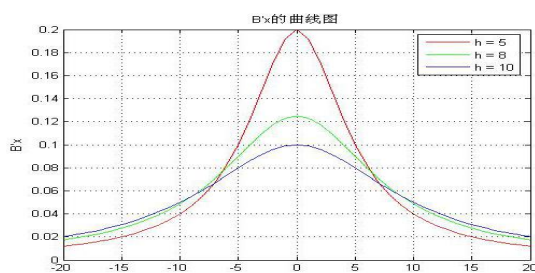


图 4.3 B' X 的曲线图

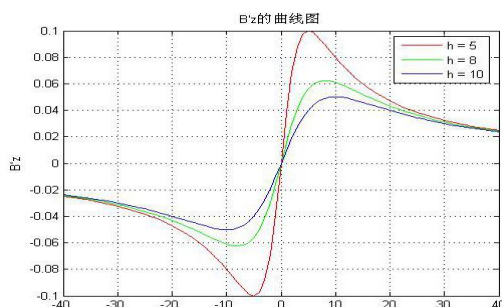


图 4.4 B' Z 的曲线图

4.3 磁场传感器布局

由4.2所述的性质，我们可以知道，垂直线圈可以比较容易地得出小车与导线的相对位置，水平线圈可以预测前方弯道以及传感器摆放的一些要求。根据这些性质，分检测导线位置和前瞻两部分论述传感器的布局方案。

4.3.1 确定导线位置布局

以垂直线圈作为检测小车与导线的相对位置，原则上采用双垂直线圈就可以判断导线的位置。然而增加传感器可以增加检测的精度，有利于小车的精确控制。我们选用三个垂直线圈平均间隔一字排开，另外两个电感平行于导线安装有利于圆环、十字等特殊元素的判断。如图4.5所示。

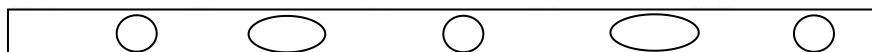


图4.5 加入前瞻前传感器排列图

4.3.2 前瞻设计

个前瞻是智能车的“眼睛”，“看“得见、”看“得清、尽量”看“得远才能保证智能车正确寻迹运动。

由于电磁小车的前瞻长度不受限制。所以前瞻在不影响整车的稳定性的前提下可以做的更长，这无疑就增加了电磁车的前瞻性。所以我们就选择了三个垂直线圈、两个竖直线圈平均间隔一字排开。

此方案的设计缺点就是道路的检测信息单一。但它最大的一优点就是虽然信息单一，但在眼睛看不到的磁场中我们就可以很好的利用这看似单一的信息来进行控制，这简化了程序的复杂性。最主要的一点是在十字交叉位置车子不受半点影响。这也是我们最终选择的原因。在固定电感的时候我们想过两种方式，一、尽量保持三个线圈呈水平位置，并且在一条直线上。如果电磁线圈安装不在同一条水平线上，那么车模在过十字交叉路口的时候，水平方向的电线中的磁场就会在这两个线圈上产生额外的感应电动势。这些电动势会造成车模方向控制的不稳定性。二、让左右两个电感摆出八字角，这样可以在过弯的时候提前预判出弯道，增加车模的转向性能。但是这种安装方式很难保证车模在

过十字弯时的流畅性，几乎办不到。但是在十字弯处可以通过对参数的细致调节使晃动尽可能减少的。综合考虑，我们还是选择了第二种可以预判弯道的安装方式。

在智能车制作之初我们想了两个方案，一个是前瞻“摇头”方案；另一个是固定前瞻方案。

1) 前瞻“摇头”方案

此方案对机械性能的要求很高，对于我们来说是很难实现的，再者因为今年的电磁组前瞻没有了长度的限制，要想获的很好的前瞻性必定前瞻会很长，那么采用摇头方案的话就会得不偿失，最终选择放弃。

2) 固定前瞻方案

此方案是一种传统的设计方案，前瞻加长了之后只要与车身接触的地方固定的好的话，不会出现抖动的现象，但会出现一个问题，如果车在高速跑的过程中转弯时惯性会很大有可能车子会发生侧翻，为了解决这个问题，我们选择了轻质的前瞻固定架子碳棒，并且前瞻电路也做到了很轻。所以在车跑的过程中以上的这些问题都没有发生过，并且也有了一定的前瞻性。

4.4 编码器的安装

为了使编码器和差速盘咬合得适当，既保证编码器和差速盘的齿轮牙合充分，避免空程，使编码器采回来的脉冲值更精确。又让编码器和差速盘间的摩擦阻力更小，以免影响车的正常行驶或者是发出很大的噪声，我们自己用PCB量身定做了一种编码器的安装配件，再加上自己的打磨，安装后达到了比较理想的效果。如图所示：



图 4.6 编码器的安装图

在本次智能车设计中，使用了单片机内部 PWM 模块、脉冲捕捉模块、A/D 模块、SCI 模块、SPI 模块及 I/O。PWM 模块分为三路输出，其中一路 PWM 输出频率为 50Hz，正脉宽约为 1.5ms~2.5ms 的信号用于控制伺服器。不同正脉宽对应不同的转向值，并且恒定某一输出值后，伺服器会保持转角不变。另外两路 PWM 输出频率为 1KHz 的信号，当 A 路输出低电平，B 路输出 PWM 信号时电机实现正转，通过改变正脉宽可以实现电机转速调节。当 A 路输出 PWM 信号，B 路输出低电平时则实现电机反转，用于减速控制。脉冲捕捉模块配置为上升沿、下降沿检测方式，用于捕捉编码器输入脉冲数，通过测量输入脉冲得到智能车运动速率。A/D 模块用于对传感器输入电压采样，获得传感器电压数值。采用 12 位精度采样，时钟配置为 2M/s，配置为 5 路通道循环采样，关闭中断采用查询方式获取转换数值。使用 SCI 模块与无线模块通讯，发送数据至电脑端，其波特率配置为 9600bit/s，大约一秒能够发送 1200 字节数据。SPI 模块用于读写 TF 卡，速率配置为 2M/s，能够实现高速读写 TF 卡，记录智能车的重要数据。

5.2 前瞻传感器模块

经过前述理论分析可知，采用线圈作为传感器有很强的适应能力，能够满足电磁场检测的要求，并且容易实现，比较适合检测变化的电磁场。在实验的初期，测试过磁阻以及磁敏三极管，对于本电磁场信号（20KHz，100mA）响应较差，几乎无法感受到电磁场信号，输出非常微弱甚至没有输出。在采用线圈作为电磁场传感器后，通过串联谐振电路，输出信号较大，并且随着位置改变输出信号也有很明显的变化。根据传感器总体结构图，设计传感器电路图包括 LC 串联谐振电路感应电磁场信号，经过后级放大电路放大弱小信号，通过半桥检波和低通滤波器后输出直流信号。在选取 LC 串联谐振电路时，实验了使用电感和自制线圈方法，由于自制线圈不易固定其电感值，无法找到匹配电容。考虑到谐振电感和谐振电容 Q 值等因素，最终选取 10mH 电感。通过公式：

$$f = \frac{1}{2\pi\sqrt{LC}} \quad (\text{公式 7})$$

可以计算出 $C=6.33\text{nF}$ ，市场上最接近的电容值是 6.8nF ，所以最终选取 10mH 电感和 6.8nF 电容作为 LC 串联谐振电路。放大级可以选用三极管和运算放大器实现，考虑电路的频率响应和成本等因素，最终选取三极管作为放大元件。

半波检波电路使用两个二极管实现，在初期实验时，使用的是普通 1N4007 二极管，经过测试后发现频率响应特性较差。最终选择了 1N5819 高效肖特基二极管。前级输出半波波形经过 π 型低通滤波器后输出直流信号。整个传感器电路原理图如图 5.2 所示，实物如图 5.3 所示。

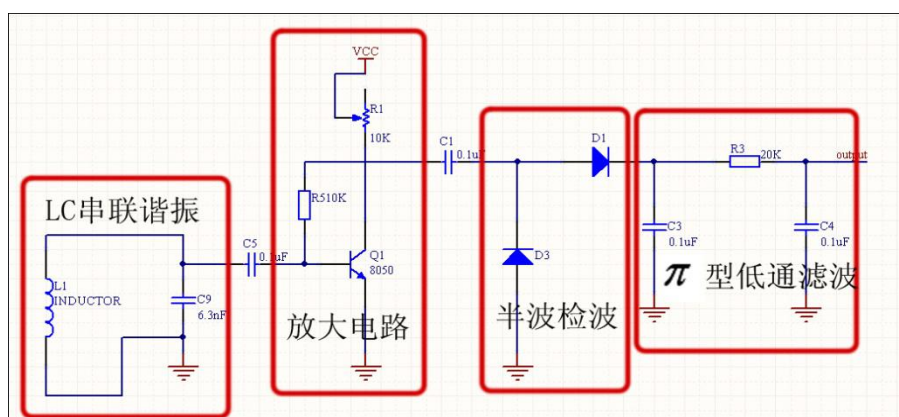


图 5.2 传感器电路图

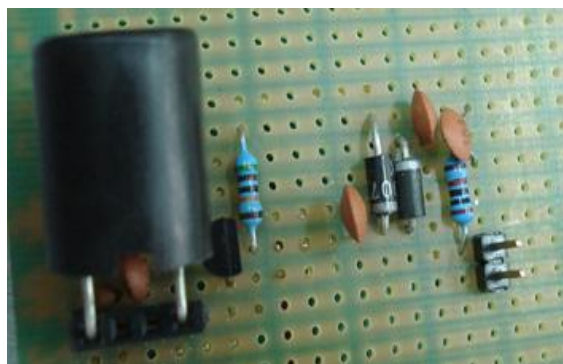


图 5.3 传感器电路

经过 LC 串联谐振电路感应出的波形为频率 20KHz 的正弦波，当传感器置于通电漆包线中时，波形如图 5.4 所示，其峰-峰值大约为 54mV。当传感器靠近漆包线时，波形如图 5.5 所示，其峰-峰值大约为 452mV。可以看出随着偏移位置的改变输出波形强度会有很大的变化。

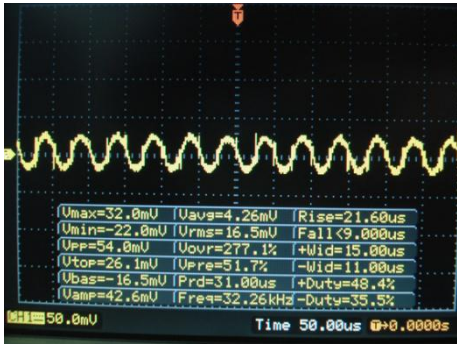


图 5.4 传感器远离漆包线时波形

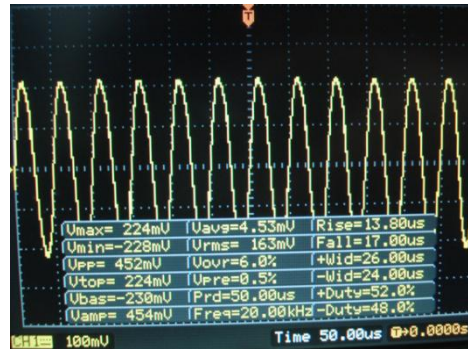


图 5.5 传感器靠近漆包线时波形

在相同距离下，经过后级放大、检波、滤波电路后输出直流信号如图 5.6、图 5.7 所示。

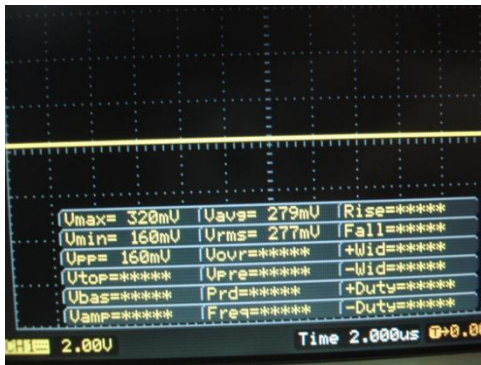


图 5.6 远离漆包线时直流输出信号



图 5.7 靠近漆包线时直流输出信号

由图5.6和5.7可以看出，在远离通电漆包线时，其直流输出信号平均值大约为279mV,当靠近漆包线时，其直流输出信号平均值可以达到4.32V。此信号可以直接输入单片机A/D接口采样，使用12位A/D进行采样大约数值变化围为44~3539。然后对A/D采回来的值作处理相应的去控制电机和舵机。

5.3 电源控制模块

电源是一个系统正常工作的基础，电源模块为系统其他各个模块提供所需要的能源保证，因此电源模块的设计至关重要。模型车系统中接受供电的部分包括：传感器模块、单片机模块、电机驱动模块等。设计中，除了需要考虑电压范围和电流容量等基本参数外，还要在电源转换效率、噪声、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流容量各不相同，因此电源模块应该包含多个稳压电路，将充电电池电压转换成各个模块所需要的电压。

智能车所需电源如图5.8所示。

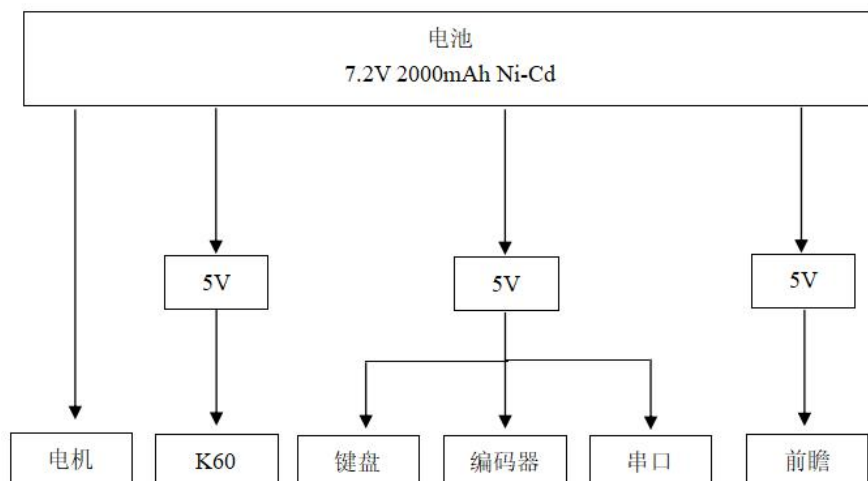


图 5.8 系统所需电源

电源模块由若干相互独立的稳压电源电路组成。在本系统中，除了电机驱动模块的电源是直接取自电池外，其余各模块的工作电压都需要经电源管理芯片来实现。

由于智能车使用7.2V镍镉电池供电，在小车行进过程中电池电压会有所下降，故使用低压差电源管理芯片LM2940。LM2940是一款低压稳压芯片，能提供5V的固定电压输出。LM2940低压差稳压芯片克服了早期稳压芯片的缺点。与其它的稳压芯片一样，LM2940需要外接一个输出电容来保持输出的稳定性。出于稳定性考虑，需要在稳压输出端和地之间接一个22 μ F低等效电阻的电容器。

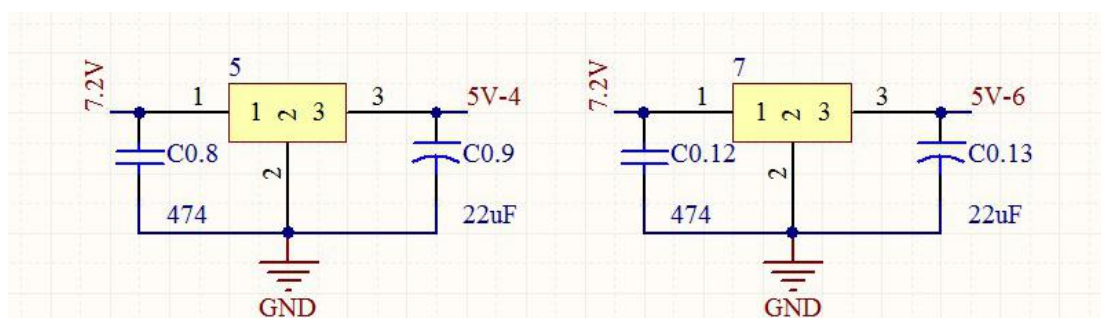


图5.9 电源电路原理图

5.4 测速反馈模块

测速采用光电脉冲编码器，光电脉冲编码器是一种数字式角度传感器，它
能将角位移量转换为与之对应的电脉冲进行输出,主要用于机械转角位置和旋
转速度的检测与控制。为了使编码器采回来的值精确度更高，我们对编码器采
回来的脉冲进行了倍频处理。

5.5 电机驱动模块

电机驱动使用集成芯片BTS7960。BTS7690具有过热、过压、欠压、过流、
短路保护等功能。BTS7960主要参数如表5.1所示。

表 5.1 BTS7960 主要参数

工作电压	5.5~27.5V
导通电阻	9mΩ
输入信号	TTL/CMOS
PWM 频率	<10KHZ
最大输出电流	6A

5.5.1 BTS7960 电路连接

本设计采用 2 片全桥驱动方式，每个电机由两块芯片控制。此方案电路
设计简单、稳定，有效提高驱动能力，避免调试过程中芯片发热严重现象，但
成本高。BTS7960 驱动电路如图 5.10 所示。

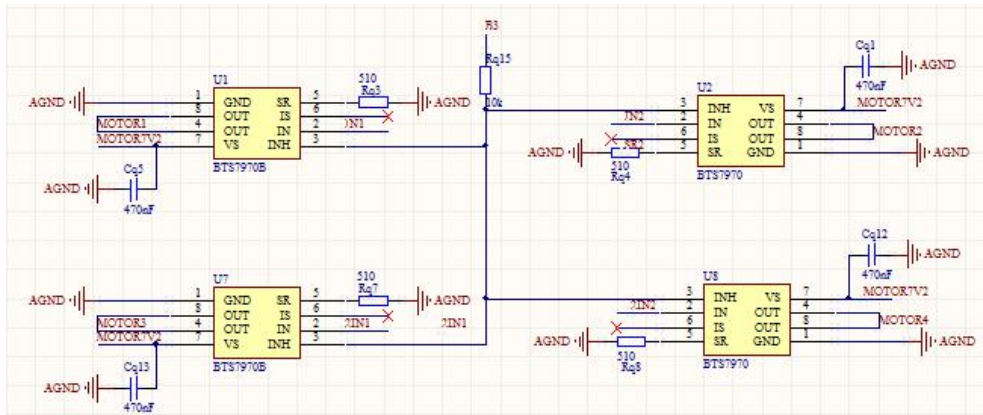


图5.10 BTS7960驱动电路

5.6 LED 数码管显示与键盘模块

这部分属于人机界面，主要是在调试的过程方便一些参数的确定和速度档的选择，比如在调节传感器的值的时候可以直接看数码管上的值，这样在调试的过程中方便了很多。而键盘的作用无异于就是在调试的过程中对一次参数的调整和设置。一般不同的赛道会用到不同的参数，这样就可以适应其他不同的赛道了。LED数码管显示与键盘如图5.11所示：

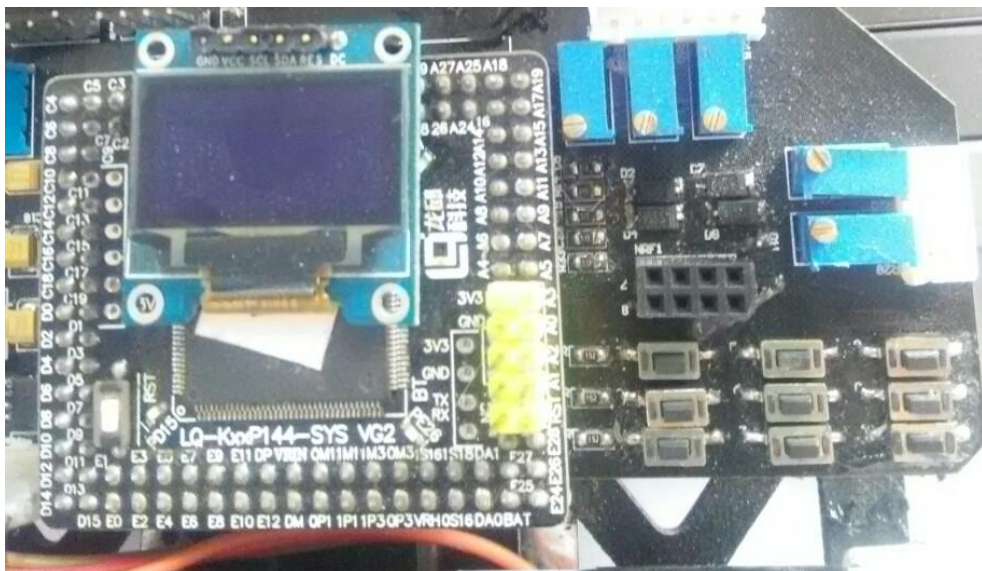


图5.11 LED数码管显示与键盘

5.7 起跑线检测模块

按照比赛规则，起跑线是6个按照一定间隔摆放的磁钢组成，而且磁钢的极性是随机放置的。

如果使用霍尔方式检测那么极性问题就一定要考虑进去，所以一对霍尔传感器至少是两个。

但还是如果使用干簧管，极性问题就不用考虑。干簧管不用供电，将干簧管的一端接地，一段接I0，然后利用单片机的内部上拉，用程序去检测低电平。这样就可以大大简化电路的设计和整个车的复杂性。

5.8 超声波测距模块

测距模块是利用的鸳鸯超声波。这个模块上有超声波模块和红外模块，发送模块上是超声波发送和红外发送，接收模块有超声波模块和红外接收模块。发送模块同时发出超声波和红外，由于光速远远大于声速，当红外信号到达接收模块时作为一个起点，用程序把管脚电平拉高，然后等待声波的到来，当声波信号到来时，从而得到一个时间差，最终计算得距离。数据 20 毫秒更新一次，通过计算，测得较为准确的距离。该模块如图 5.12 所示。图 5.13 为安装图。

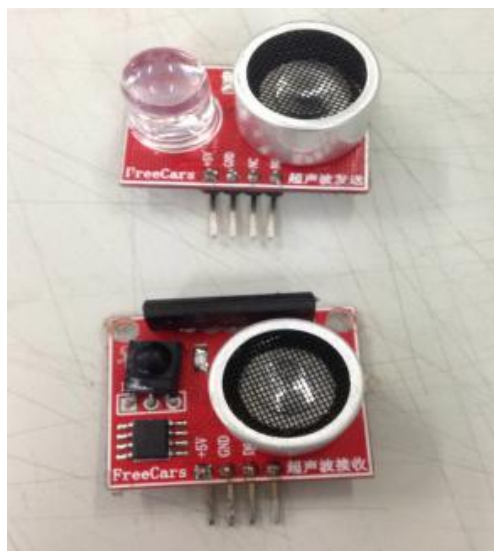


图 5-12 超声波模块



图 5-13 超声波安装

5.9 舵机驱动电路模块

今年使用的舵机是S-D5，采用的是LM1117可调稳压电路输出的5.5V供电，通过不断地实验，我们发现S-D5这款舵机在5.5V时会有比较快的响应速度，使得车体的转向更加地灵敏，同时也不至于电压过高而使舵机损坏。效果比较理想。

5.10 系统主板电路的设计

由于硬件是分模块进行设计的，最终必然要互相连在一起才能顺利工作。系统主板电路主要包括：电机驱动模块，光电隔离模块、CPU系统模块、信号检测模块，舵机驱动模块、起跑线检测模块、无线通信模块、各部分的电源电路等。

电路布局过程中遇到的最大问题就是信号干扰，直接导致车无法完整的跑完一圈，在布线过程中我们总结出以下应该注意的地方，使我们的电路板抗干扰性能达到了一定的高度：

由于前瞻信号检测电路是车的眼睛，是非常非常重要的，也是最易受干扰的部分，所以在该部分布线时我们特别注意“隔离”，这里的“隔离”我们具体是这样做的，前瞻部分的供电电源和其他部分的稳压电源隔离，前瞻部分铺的地也和其他部分的铺的地隔离，只是通过一个电源芯片和外界相连，事实效果是这种接法干扰降到了很小很小。

前瞻部分元件布局时引脚间尽可能的近，线也要粗一点，能保证信号无损耗同时“不给干扰窜进来的机会”来进行放大。

对于前瞻线的处理也是比较有讲究的，传统的用两根线直接从前瞻将信号引回来的方式没有任何的抗干扰能力，我们采用的是自制的双绞线。这种做法有一定的屏蔽干扰信号的能力。

为了解决一部分的静电干扰，我们在P C B板上引出了好多地线的接口，将一些容易产生放电和容易受到电干扰的器件如：电机、编码器等等的外壳进行接地处理，对消除车体的静电有一定的效果。

5.11 硬件电路的小结

对于硬件电路部分，一定要一步一步地做好，尤其是抗干扰问题。单片机、舵机、前瞻电路等等的供电电压一定要稳，电源纹波一定要小。车在运行的过程中，电机的强电流干扰是十分巨大的，会严重影响信号的检测，所以要单独把电路隔开处理。

硬件电路是智能车的基础，只有基础打得很好了才能继续软件方面的工作。

第六章 软件部分设计

高效稳定的软件程序是智能车平稳快速寻线的基础。本车采用 6*8 工字型电感作为寻线传感器，在智能车的转向和速度控制方面，我们使用经典的 PID 控制算法，是在寻线中智能车达到了稳定快速的效果。

6.1 整体框架

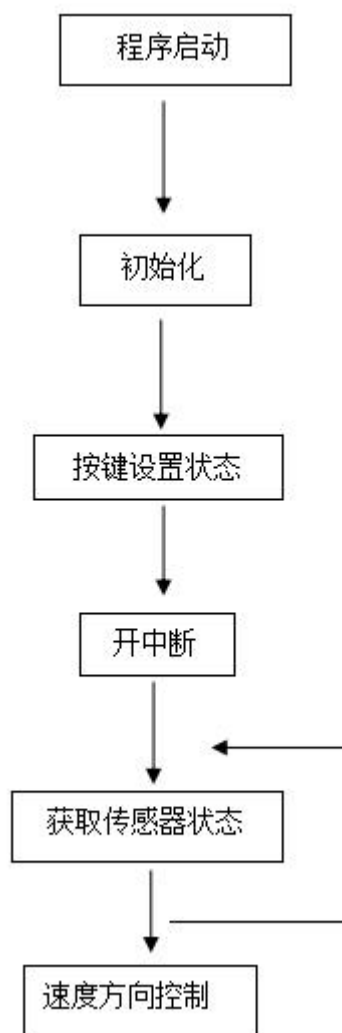


图 6.1 整体框架图

6.2 舵机控制流程

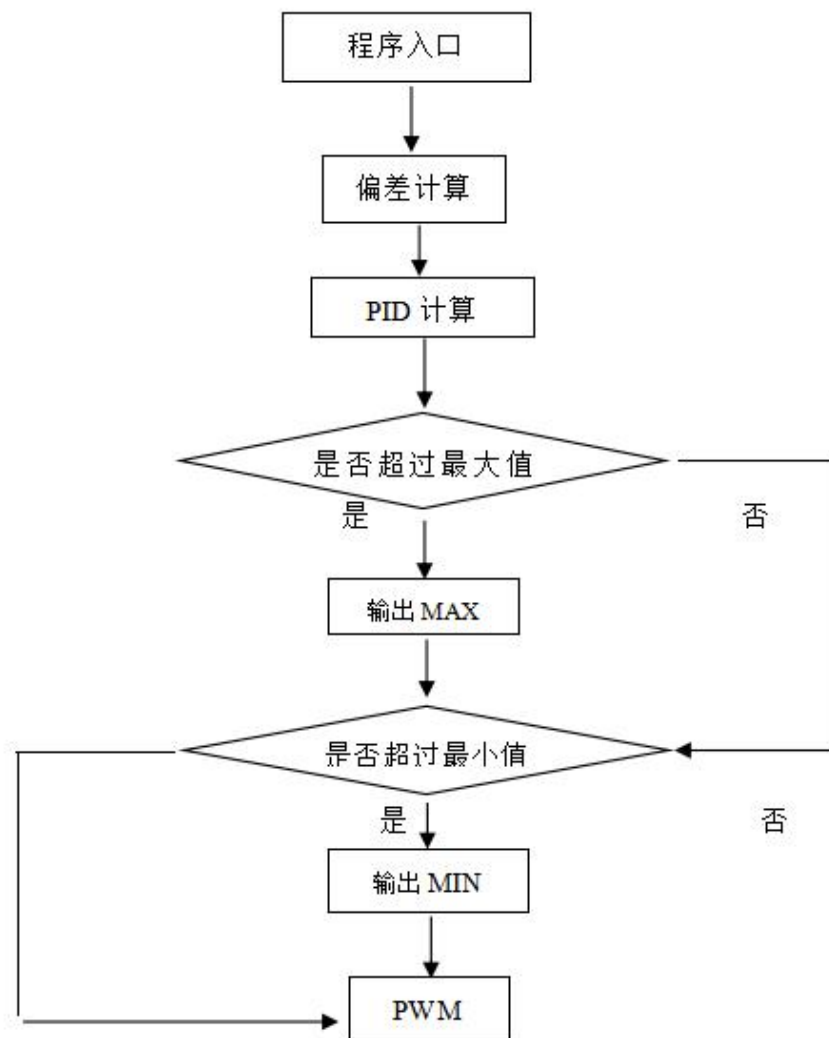


图 6.2 舵机控制

6.3 速度控制

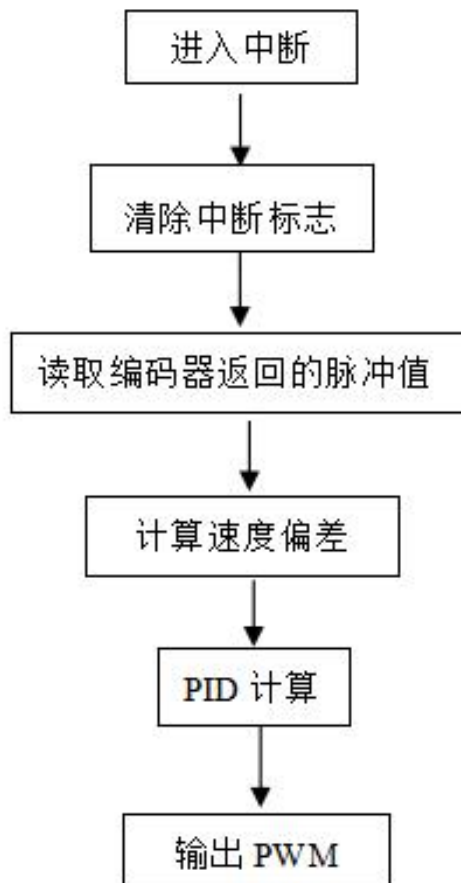


图 6.3 速度控制

6.4 PID 算法

PID控制是按偏差的比例，积分和微分进行控制的一种控制规律。它具有原理简单，易于实现，参数整定方便，结构改变灵活，适用性强等优点。有模拟PID调节器和数字PID算法两种。

用计算机实现的数字 PID 算法代替了模拟 PID 调节器，是当今工业过程计算机控制系统中应用最广泛的一种。

6.4.1 模拟 PID 算法

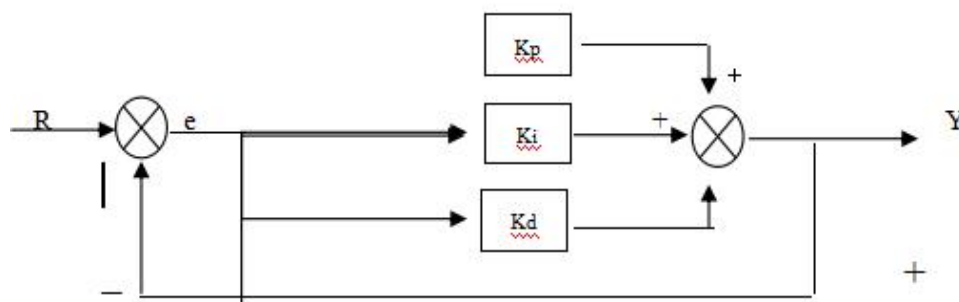


图 6.4 模拟 PID

6.4.2 PID 参数作用

K_p 越大，系统的响应速度越快，调节精度越高，但是容易产生超调，超过一定范围会导致系统振荡加剧甚至不稳定。

K_i 越大，系统的静态误差消除越快，但是容易在初期产生积分饱和现象，从而引起响应过程的较大超调。

K_d 的作用是在回应过程中抑制偏差向任何方向的变化，对偏差变化进行提前预测。但是会使响应过程提前制动，从而延长调节时间。

6.4.3 PID 算法的数字实现

在数字计算机中，PID 控制规律的实现，也必须用数值逼近的方法。当采样周期相当短时，用求和代替积分，用差商代替微商，使 PID 算法离散化，将描述连续-时间 PID 算法的微分方程，变为描述离散-时间 PID 算法的差分方程。

位置式 PID 控制算法

$$U(K) = K_p * e(K) + K_i \sum_{j=0}^K e(j) + K_d [e(k) - e(k-1)] + U_0$$

式中 U_0 ——控制量的基值，即 $K=0$ 时的控制； $U(k)$ ——第 K 个采样时刻的控制量； K_p ——比例系数； K_i ——积分放大系数； K_d ——微分放大系

数。这种算法输出的是控制量的绝对数值。

增量式 P I D 算法

$$\begin{aligned}\Delta U(k) &= U(k) - U(k-1) \\ &= Kp*[e(k) - e(k-1)] + Ki*e(K) + Kd[e(k) - 2*e(k-1) + e(k-2)]\end{aligned}$$

第七章 开发与调试

本智能车以FIRE_kineti-IAR-EMBDDLE开发工具，加上良好的测试环境使智能车开发和调试能够顺利而快速地完成。

7.1 开发环境

程序的开发是在组委会提供的CodeWarrior IDE 下进行的，包括源程序的编写、编译和链接，并最终生成可执行文件。

CodeWarrior for S12 是面向以HC1 和S12 为CPU 的单片机嵌入式应用开发软件包。包括集成开发环境IDE、处理器专家库、全芯片仿真、可视化参数显示工具、项目工程管理器、C 交叉编译器、汇编器、链接器以及调试器。下面以一个简单的实例来说明使用 Corewarrior IDE 开发全过程。

一、用 IAR 首先要新建工作区，在工作区里再建立工程。

1) 建立工作区间

新建 IAR 工作区间，首先是菜单 File 里选择 Open 再选择 Workspace, 如下图 7.1 蓝色圈所示。



图 7.1 新建 IAR 示意图

接着就会看到一片空白。这时就是“新的办公区”了。



图 7.2 工作空间示意图

有了工作空间就可以建立工程了。

2) 建立工程

菜单 Project>Create New Project。点了之后出现下面这个对话框。如图 7.3 所示。

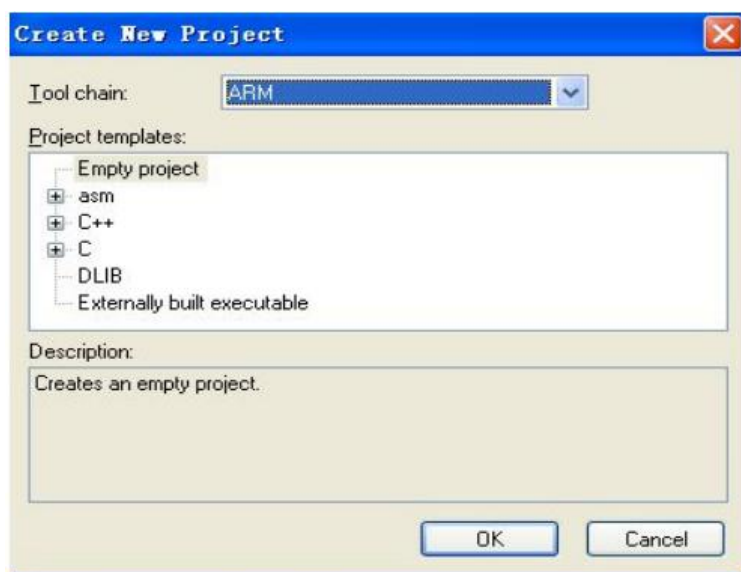


图 7.3 新建工程示意图

上面图中有个 Tool Chain:ARM, 这里选的是芯片的类型, IAR 有很多很多专用的环境, 比如 ARM 的, 430 的, 8051 的等。这里只有 ARM, 相当于没得选。它的下面有个 Project templates(工程模块), 然后是 asm, C++, C 等, 选择用户使用的开发语言, 我们选 C。展开 C 前面的+号, 显示如下图(上个图的局部)。

二、项目的设置

1) 弹出设置对话框

在工程名上点右键，选择 options，或者点菜单——“Project”——“Options”进入，或者按快捷键 AL+F7

下面进入设置，这里只说明几种基本设置，具体的还得根据自己需要来调整。没有出现设置栏目就保持默认。

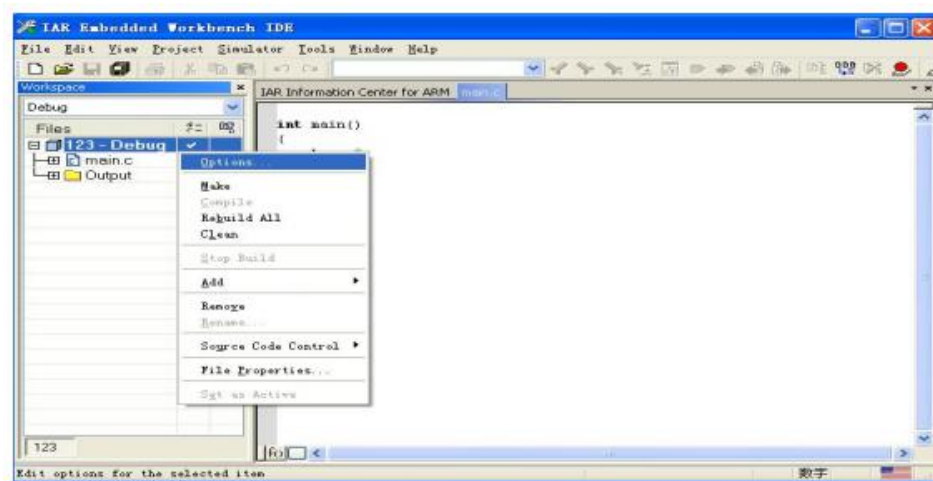


图 7.4 项目设置

2) 先选择芯片的型号。我们用的是 K60，这里选 Freescale K60Xxxx。

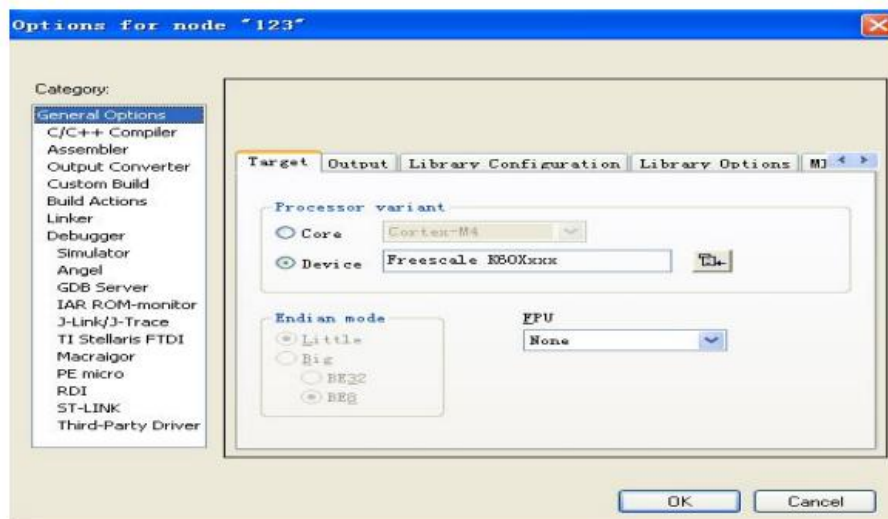


图 7.5 芯片选择示意图

3) 编程语言栏目设置。以 C 语言为例，如果是 C++就选 C++。

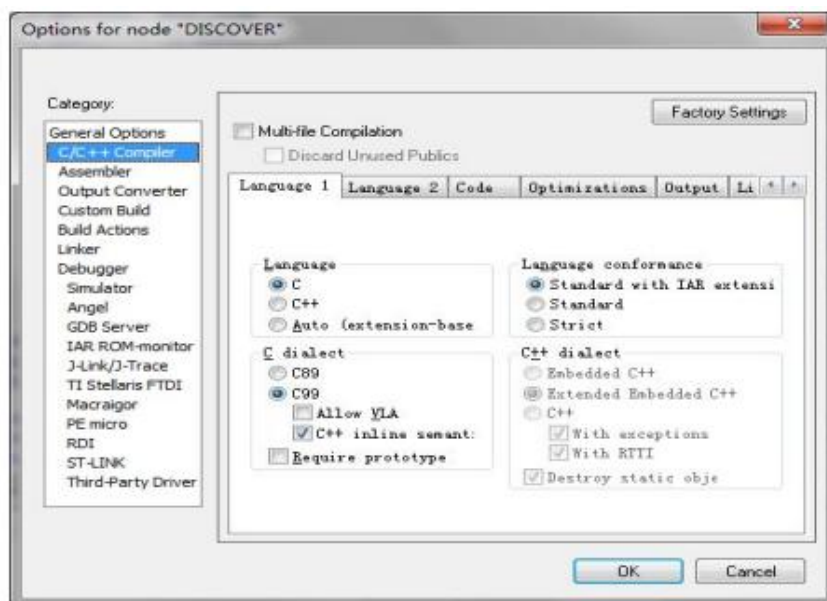


图 7.6 编程语言选择

4) 这里设置的是编译后是否要生成 BIN、HEX 文件。

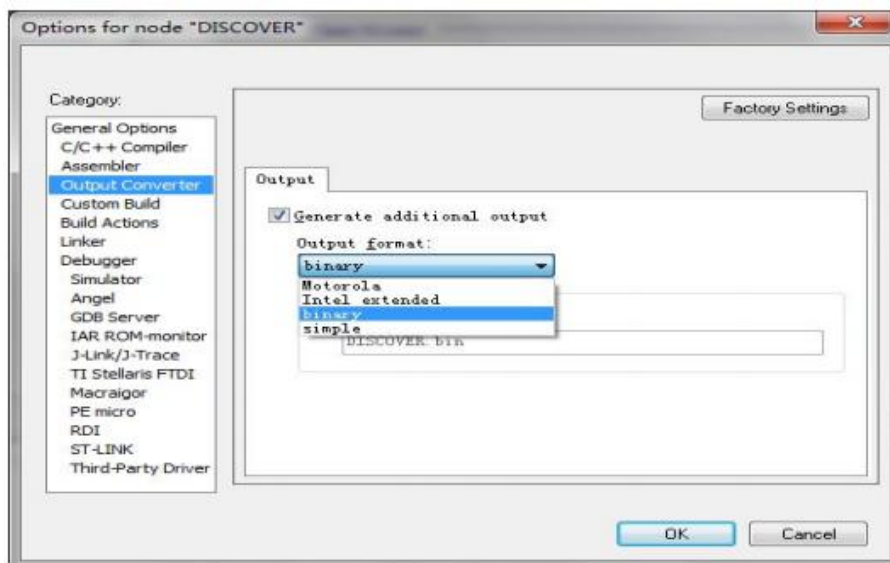


图 7.7 设置生成文件

5) Linker 的设置，这里记录着储存的地址。

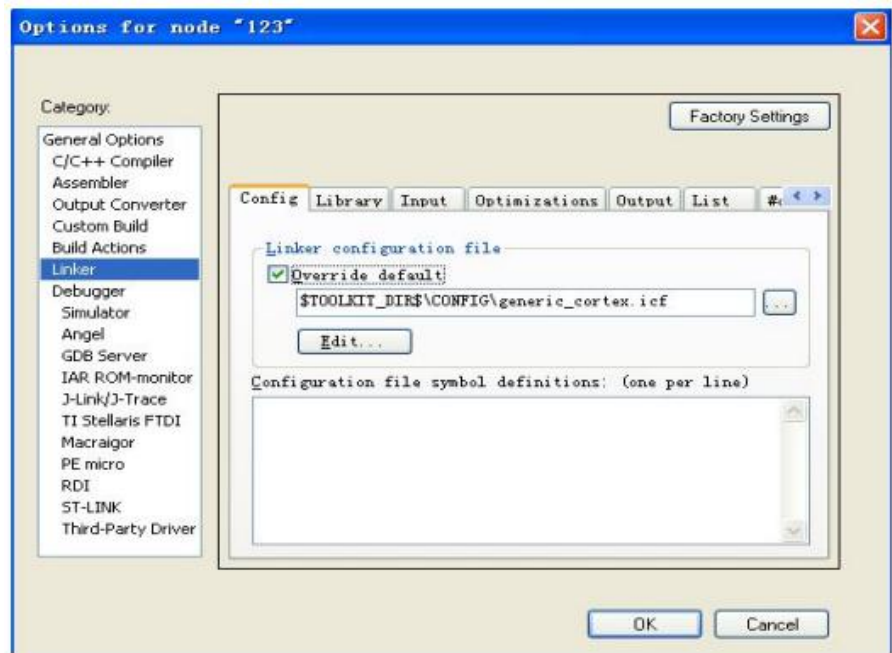


图 7.8 Linker 设置

6) 调试工具的设置。

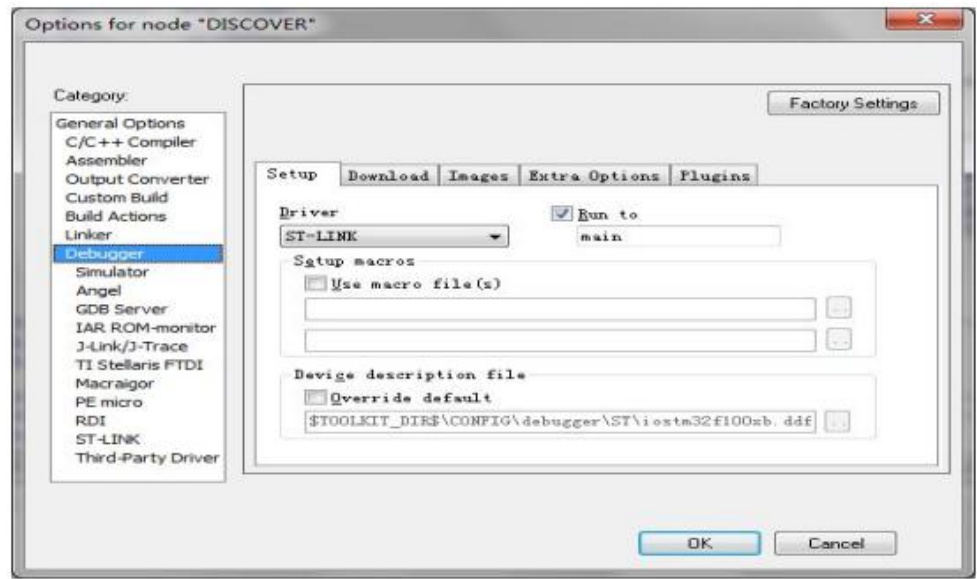


图 7.9 调试工具设置

A) 如果使用 JTAG 下载器，则在驱动中选择

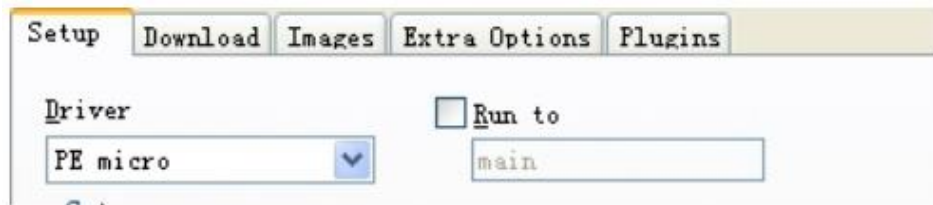


图 7.10 下载器选择 (a)

B) 如果使用 J-Link 下载器，则在驱动中选择



图 7.11 下载器选择 (b)

基本设置就这样了，具体的根据自己情况修改。

7.2 程序下载

BDM 链接成功后，首先先将程序进行编译，先观察是否有问题，所有的功能如图：



图 7.12 程序下载 (a)

程序下载成功后，如图：

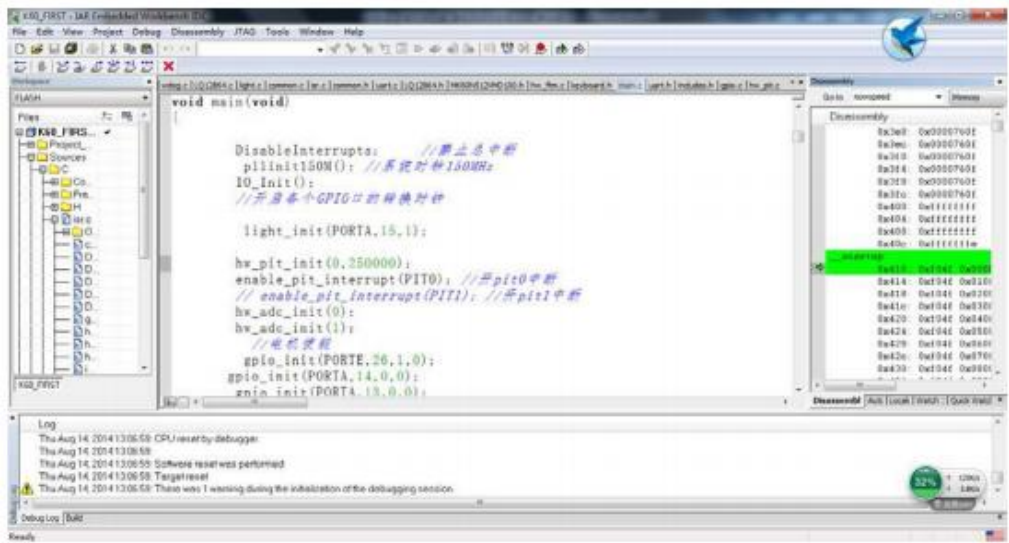


图 7.13 程序下载 (b)

7.3 测试环境

我们用PVC赛道铺设一自行设计测试跑道，方便调试。调试中主要考虑跑道摩擦系数、电流源输出电流过大和过小对智能车的影响。试验结果表明，智能车运行最佳状态为跑道摩擦系数较大、漆包线通电电流为 $100\text{mA} \pm 5\text{mA}$ 、频率为 $20\text{KHz} \pm 2\text{KHz}$ 。自行设计跑道如图7-14所示。



图 7.14 自行设计赛道

第八章 总结

8.1 制作成果

比赛无极限，追求无止境。在已经取得的调试结果之上，我们仔细观察，认真思考，对于发现的问题，逐步完善控制方案，突破现有观念的限制，寻求更加先进完美的方案，在比赛规定的技术范围之内，追求卓越。在不断的改进与实验中，收获的不仅是最终的比赛成绩，而且还有自己动手的能力和勇于探索的精神。

8.2 问题与思考

在车模的制作过程中，我们的小车曾经出现了很大问题，如小车在行驶的过程中抖动厉害，前瞻采集信号不稳定使小车突然跑出赛道，电机驱动烧毁，稳压芯片烧坏以至于烧坏了很多块小系统。因为信号微弱经常冲出赛道等问题，或多或少拖慢了我们的进度，但经队员的不懈努力与老师的精心指导，我们从硬件和软件上做了一步步的改良和升级，渐渐对各模块有了深入的认识，对整体有了清晰的把握，我们在不断的前进的同时，小车也不断的加速前进。

8.3 不足与改进

从小车总体性能各因素考虑，小车系统分为采集、处理、控制三部分。由于电磁组特点决定了小车的前瞻比较小，小车容易出现严重的滞后，因此信号的采集影响到小车的控制策略。在实践中发现，工字电感采集回来的信号存在一定的滞后，而且单片机 AD 精度有限，电感位置变动时存在变化的死区造成检测精度不足。在处理这些精度不太高的数据时如何合理的利用其中规律我们有待研究。在控制算法上，有待尝试多种思路进行比较。当小车行驶较快的时候，机械问题突出，严重影响了小车的运行，制约了小车的提速空间。

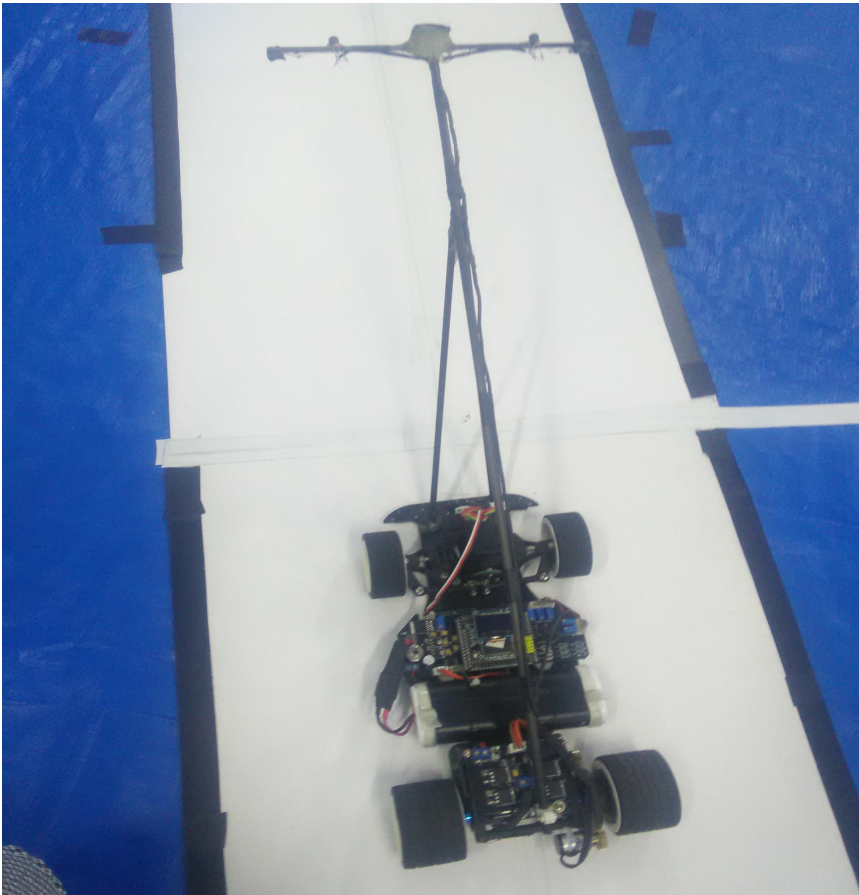
参考文献

- 【1】. 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
- 【2】. 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.
- 【3】. Freescale Semiconductor, S12XS-Family Reference Manual, Rev. 1.03.
- 【4】. 童诗白, 华成英. 模拟电子技术基础[M]. 北京. 高等教育出版社. 2000
- 【5】. 余志生. 汽车理论. 北京:机械工业出版社. 2006 年5 月.
- 【6】. 胡寿松. 自动控制原理. 北京:科学出版社. 2001 年2 月第4 版.
- 【7】. <http://www.freescale.com.cn/>.

附录 A 智能车整车图片

模型车参数说明

项目	参数
车模尺寸（长X宽X高）（mm）	约550X180X250
车模重量（带电池）	约1.2
电路功耗（匀速行驶电流mA）	约1300
电容重量（uF）	约1200
传感器种类和个数	赛道检测X5，速度检测X1
除了车模原有的驱动电机、舵机外伺服机的个数（个）	0
赛道信息的检测精度（mm）	12.5
赛道信息检测频率（次/秒）	50



附录 B 部分程序原代码

```
/******初始化******/
void Init(void) {

/******双电机PWM初始化******/

    PWME=0x00;          //禁止PWM输出
    PWMCTL=0xFF;         //级联方式
    PWMPRCLK=0x00;       //ClockA=Busclock=40MHz
    ClockB=Busclock=40MHz

    PWMCLK=0x00;         //全选择A, B
    PWMSCLA=0;           //SClockA=ClockA/(2*PWMSCLA)
    PWMSCLB=0;           //SClockB=ClockB/(2*PWMSCLB)
    PWMPOL=0xFF;         //PWM输出起始为高电平
    PWMCAE=0x00;         //左对齐方式
    PWCNT01=0;
    PWCNT23=0;
    PWCNT45=0;
    PWCNT67=0;
    PWMDTY01=0;          //电机反转
    PWMPER01=4000;
    PWMDTY23=0;          //电机正转
    PWMPER23=4000;       //输出频率=
    (40MHz)/(4000)=10k, 舵机控制

    PWME=0xFF;

/******双电机PWM初始化******/

/******ATD初始化修改过******/
```

```

//以下为16位精度数据
ATDOCTL1=0X3F;    //十位精度
ATDOCTL2=0X60;    //硬件清楚CCFx，禁止中断
ATDOCTL3=0X88;    //右对齐、序列长度为1, 禁止FIFO模

式

ATDOCTL4=0X49;    //采样时间4ADCclock，预分频系数10
ATDOCTL5=0X20;    //单通道、连续，起始输入通道0
ATDODIEN=0x00;

/*****ATD初始化*****/

/*****用于捕捉编码器脉冲的T口初始化*****/
*****/

PACTL=0X50;        //PT7 PIN, PACN32
16BIT, FALLing edge, NOT INTERRUPT
TCTL3=0x40;        //TCTL34输入捕捉控制器 , 0、6
通道上升沿捕捉，其他禁止
TCTL4=0x02;        //40表示ICx禁止，1表示上升沿，
2表示下降沿，3表示任何沿
TIE =0x00;        //定时器中断使能寄存器, 0表示
相应IC/OC通道禁止中断
TIOS =0x00;        //输入捕捉输出比较寄存器： 0
输入捕捉, 1输出比较

/*修改过*/
//TCTL3_EDG7x=1;    //c-输入捕捉7任何沿有效,
TSCR1=0X80; //定时器允许工作
TSCR2=0X01; //定时器溢出中断允许，及分频因子选择
TFLG1=0X01; //清空中断标志 TFLG1指示了中断发生在
哪个通道?

//要对相应位清零时，可以对它进行置1
操作。 C7F-C0F： IC/OC的中断标志。

```

```
PACNT=0;          //这个应该是PAC通道的脉冲计数器

/*****用于捕捉编码器脉冲的T口初始化
*****/

/*****定时器中断初始化
*****/

PITCFLMT_PITE=0; //定时中断通道0关
PITCE_PCE0=1; //定时器通道0使能
PITMTLD0=40-1; //8位定时器初值设定, 40分频, 在
40MHzBusClock下, 为1us
PITLDO=1000-1; //16位定时器初值设定。
PITTIME*1us=10000*1us=10ms
PITINTE_PINTE0=1; //定时器中断通道0中断使能
PITCFLMT_PITE=1; //定时器通道0使能

/*****定时器中断初始化
*****/

/*注释: 这里的中断是指定时器定时候时间到了后就中断, 与定时器的输入捕捉无关,
在输入捕捉中已经禁止输入捕捉中断*/

/*****DFLASH初始化*****/
FCLKDIV=0X0f;
IRQCR&=0xbf;      //IRQCR. 6 (IRQEN) =0禁止IRQ中断
(默认开)
COPCTL=0x00;      //COPCTL. 2-0 (cr2:cr0) =000禁止看
门狗

/*****DFLASH初始化*****/

/*****端口初始化*****/
DDRK=0xFF;
```

位

```

        DDRM=0xFF;
        DDRS=0xFF;
        DDRJ=0xFF;
        DDRH=0xFF;
        DDRE=0x00; //编码器输入端，PORTB_PB6，PORTE，共九

        DDRB=0xBF; //PB6是一个输入端口
        PORTB=0;
        PORTB_PB5=1; //编码器先清零，再启动计数
        PORTB_PB5=0;
        PORTB_PB2=1;
        PORTB_PB3=1; //开电机驱动使能端
        IO_ini();
        CH451_ini();
        CH451_WriteCommand(CH451_BCD);
        PORTK=0xFF;
        PORTE=0;

        /*****端口初始化*****/
        SCIOBD=260;
        SCIOCR1=0;
        SCIOCR2=0x2c; //接收发送均允许

    }

    /*****初始化*****/

    void data_init(void) {
        speed_set=20*70;
        differ_Kp=80;
        differ_Kd=95;
        speed_Kp=2.8;
        speed_s=20*63;
        speed_m=20*63;
        speed_b=20*63;
    }

```

```
}
```

```
void keyanalysis(void) {  
    if(KeyValue<3) {  
        key[0]=KeyValue;  
    }else if(KeyValue<15) {  
        key[1]=KeyValue;  
    }else {  
        key[2]=KeyValue;  
    }  
}
```

//获取到按键, 根据按键值存在不同的变量中

//key[0]作为功能选择专用, 为0时作为显示控制, 为1时做为速度调档

//key[1]功能的分支, 例:当按下显示键0后, 按下3~7可分别显示5个电感的

当前采值

//key[2]为电机启动键

```
}
```

```
void dispchoose(void) {  
    print_s(KeyValue);  
}
```