**2022“数智中原”河南省大学生电子设计大赛**

**（I 题）**

**【本科组】**



**2022年12月15日**

自动绕障行走小车（I题）

摘 要

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX.

关键词：XXXXX；XXXXX；XXXXX；XXXXX；XXX

Abstract

This system with STM32F407RBT6 plate core microcontroller as the main control hcip,quad bike models as themodel,the car through the digitalcamera K210 acquisition target target surface image,image information processing,getthe target position information of impact on enables the car along the corridor on the ground to draw some of the redsolid line,and complete the turn left and right,by Ann Pressure sensitive sensor is installed to load about 200g of drugs into the car at one time,and the car automatically starts to transport after detecting the completion of drugloading;

Key words:XXXX;XXXX;XXXXX;XXXX

目录

[一、题目分析与方案论证 1](#_Toc121941254)

[1.1、题目分析 1](#_Toc121941255)

[1.1.1题目任务 1](#_Toc121941256)

[1.1.2基本要求 1](#_Toc121941257)

[1.2、主控选择论证 1](#_Toc121941258)

[1.3、功能模块论证 1](#_Toc121941259)

[1.3.1驱动模块的选择论证 1](#_Toc121941260)

[1.3.2障碍识别的办法论证 2](#_Toc121941261)

[1.3.3转向的办法论证 2](#_Toc121941262)

[1.3.4终点位置的识别论证 2](#_Toc121941263)

[二、控制系统分析 2](#_Toc121941264)

[2.1、控制思路 2](#_Toc121941265)

[2.2、程序流程图 2](#_Toc121941266)

[2.3、PID算法辅助设计 3](#_Toc121941267)

[三、硬件设计 4](#_Toc121941268)

[3.1、供电设计 4](#_Toc121941269)

[3.1.1、电源选择 4](#_Toc121941270)

[3.1.2、主控供电 4](#_Toc121941271)

[3.1.3、电机供电 4](#_Toc121941272)

[3.2、PCB板设计 4](#_Toc121941273)

[四、理论分析及仿真设计 4](#_Toc121941274)

[4.1理论分析： 4](#_Toc121941275)

[4.2仿真设计分析： 5](#_Toc121941276)

[五、测试的方案与测试结果 5](#_Toc121941277)

[5.1、测试方案 5](#_Toc121941278)

[5.1.1检查电路 5](#_Toc121941279)

[5.1.2通电观察 5](#_Toc121941280)

[5.1.3观察波形与数据 5](#_Toc121941281)

[5.1.4组装测试 5](#_Toc121941282)

[5.2、测试条件与仪器 6](#_Toc121941283)

[5.3、测试结果（数据）及分析 6](#_Toc121941284)

[六、总结与感悟 6](#_Toc121941285)

[参考文献 6](#_Toc121941286)

[附录1电气原理图 7](#_Toc121941287)

[附录2主要程序代码 8](#_Toc121941288)

一、题目分析与方案论证

1.1、题目分析

1.1.1题目任务

题目要求设计制作一个能自动绕越障碍物的电动小车。实现图1所示绕障示意。

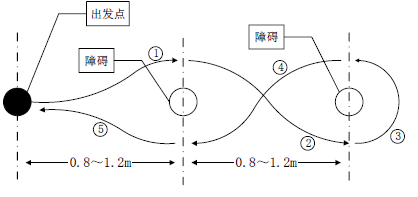


图1 基本要求绕障示意图

1.1.2基本要求

完成图1所示绕障要求。地面安放 2 个直径约15cm、高度20cm 的圆柱形红色障碍物，两障碍物间距离在0.8~1.2m 间可以调整。小车从处于二障碍物连线左侧0.8~1.2m 的黑色实心圆（直径15cm）出发，沿图示箭头轨迹运动，最后回到出发点。

行走过程中，小车不得碰撞障碍物；小车中心标志距离障碍物的距离不得超过40cm;返回出发点后小车中心标志应尽量与黑色实心圆中心重合；小车完成整个行程不得超过30 秒。

1.2、主控选择论证

方案一：51 单片机。采用通用的 51 单片机作为主控制器。51 单片机通用灵活、价格低廉、使用方便，但此单片机字长有限，数据处理能力很弱，处理速度较慢，资源不够丰富， 需要扩展较多的外围电路，降低了系统的可靠性，增加了制作的费用，难以满足本设计要求。

方案二：STM32F1单片机。基于 STM32F1 [系列的开发板](https://www.icxbk.com/ask.html?words=%E5%BC%80%E5%8F%91%E6%9D%BF)，性价比较高、资料非常丰富，这是最大的优势。对于这个项目，并不需要太多的 IO 口和寄存器，为了节约费用和空间，小车采用了 STM32F103C8T6 单片机作为主控芯片。芯片采用中容量芯片，一共48 个引脚，64KFLASH。37 个 GPIO 和 4 个 16bit 定时器/计数器。在减小主控模块的同时，保证了足够的 IO 口和定时器以供使用，能够完成题目设计要求的使用。

1.3、功能模块论证

1.3.1驱动模块的选择论证

方案一：使用 l298n 电机驱动模块，由 12v 稳压电池直接供电，再由微控制板给予 PWM控制信号来控制电机转速和正反转。

方案二：使用 Tb6612fng 电机驱动模块，由 12v 稳压电池直接供电，再由微控制板给予 PWM控制信号来控制电机转速和正反转。

经分析与要求，l298n 相对于 tb6612 模块，占用了更大的体积，对于集成度较高的小车来说，显然选用 tb661 驱动模块能为小车腾出更多的位置安装其他模块。最终选择方案二。

1.3.2障碍识别的办法论证

方案一：双目摄像头。采用双目摄像头进行识别定位，通过摄像头标定，可以掌握摄像头的所有信息（参数或系数)，从而可以确定现实世界中的三维点与摄像头捕获图像的二维投影(像素）之间的精确关系，从而确定目标的距离实现避障。双目摄像头较为昂贵，且需要算法进行优化运算，较为复杂。本项目采用STM32F103C8T6开发板，遂不采用双目摄像头。

方案二：红外测距避障模块。红外测距都是采用三角测距的原理。 红外发射器按照一定角度发射红外光束，遇到物体之后，光会反向回来，检测到反射光之后，通过结构上的几何三角关系，就可以计算出物体距离D。而本题目选用的障碍物为圆柱形红色障碍物，且受环境光线等环境因素的干扰，所以不采用红外测距模块。

方案三：超声波模块。超声波测距的原理是利用超声波在空气中的传播速度为已知，测量声波在发射后遇到障碍物反射回来的时间，根据发射和接收的时间差计算出发射点到障碍物的实际距离。超声波模块价格相比双目摄像头较低，相比红外测距模块较为稳定，干扰因素较少。故本方案采用超声波模块。

1.3.3转向的办法论证

方案一：三轮小车，使用简易万向轮，通过控制两后轮不同的转速来实现小车的转向。

方案二：四轮小车，用舵机经连轴器和小车两前轮相连，能实现单片机控制小车前轮左右摆动，从而控制小车转向。

方案三：四轮小车，通过控制左右轮不同的转速来达到转弯的目的，同时采用pid算法进行速度控制。

经分析与组建实物验证，方案一实物结构不稳定，行走过程易发生偏移。方案二实物结构有些复杂，且考虑到实物零部件购买组装存在部分问题。考虑采用方案三，在保证转向反应足够灵敏实现功能的同时，能够顺利完成实物的搭建。

1.3.4终点位置的识别论证

方案一：采用简易的光电传感器结合外围电路探测,但实际效果确实不理想， 对行驶过程中的稳定性要求较高，而且误测几率比较大。最重要的是小车整体容易因为该器件造成整个系统的不稳定。

方案二：采用两只 KY-032红外对管，分别至于小车身前轨道的两侧，根据两只光电开关接受到的线的情况来控制小车转向和速度来调整小车最终的停车位置。

经分析，最终选用方案二。

二、控制系统分析

2.1、控制思路

以单片机 STM32 F103C8T6 为核心,控制Tb6612fng电机驱动模块，利用超声波测距模块完成测距，根据测距距离完成转向的判别，由单片机对小车状态做出实时反应，做出相应的控制指令。系统通过调 PWM 输出可控制小车的前进、后退、转向、加速、减速等动作。

OLED 屏显示模块辅助显示表示当前超声波模块的测距距离。MPU6050陀螺仪模块辅助小车实现转弯绕障动作。

2.2、程序流程图

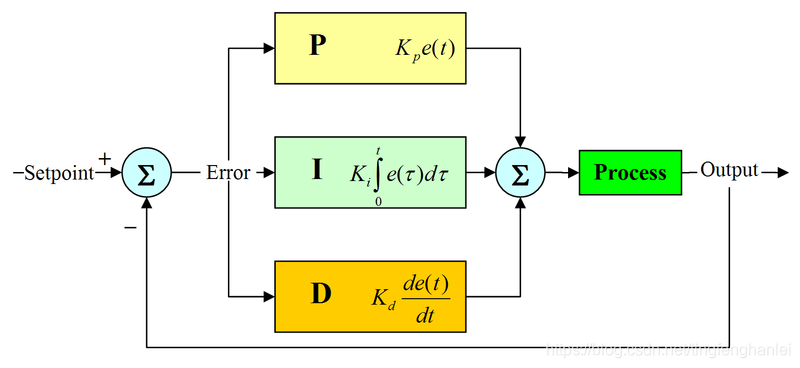
图示

描述已自动生成

图2 程序流程图

2.3、PID算法辅助设计

PID 是一种经典的控制方式，用于多种方式。本次的电子设计大赛用到的PID 主要是用于处理电机速度。因为在行走的过程中，电机速度是十分重要的，当有差速时，会导致直线行走路线不准确，发生偏移。所以本实验主要是用 PID 控制速度相等。PID 的三个参数 P、I、D ，在经过精确的调参后，电机的速度达到了完美的效果。

图3 PID算法控制原理图

三、硬件设计

3.1、供电设计

3.1.1、电源选择

选用12V直流电池组进行供电。

3.1.2、主控供电

本次设计的小车以 stm32f103c8t6 最小系统板作为主控芯片，所以用MP1584EN可调降压模块将12V的直流电源降压至 5v 作为主控模块供电来源。

3.1.3、电机供电

小车采用 12v 稳压电源为 tb6612 驱动供电控制双电机，双电机驱动保证小车动力足够。

3.2、PCB板设计

详细电路原理图详见附录1，图4为 PCB板设计平面图。

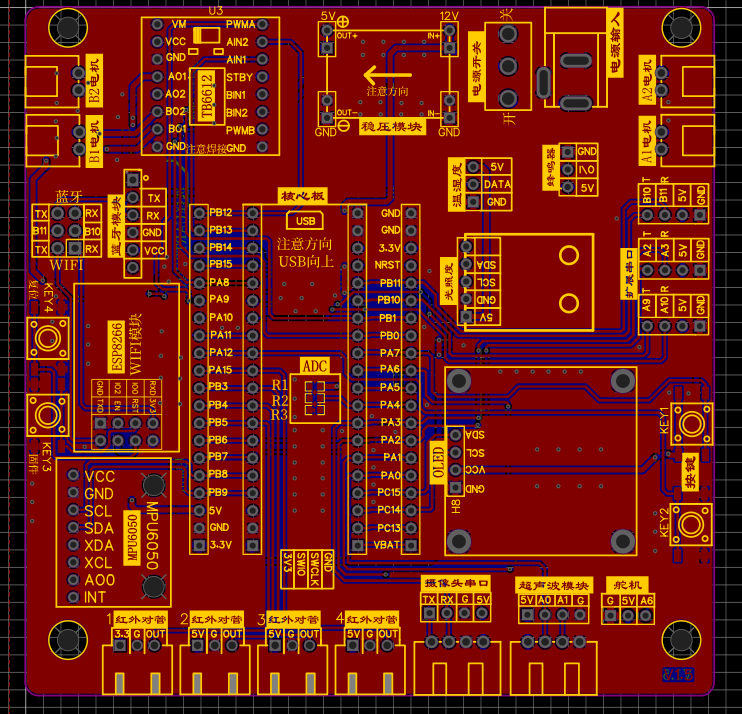


图4 PCB板设计平面图

四、理论分析及仿真设计

4.1理论分析：

测试是需要经过逻辑分析后进行实践检验。对测试结果进行分析后，将反馈用于修改参数，最终达到理想值。我我们测试的部分主要是：电机速度、单行程时间、转弯角度、转弯半径等等，主要修改的参数有控制电机PWM、超声波模块距离判别参数、MPU6050初始参数。经过反复调参，最终达到理想值。

4.2仿真设计分析：

采用Proteus进行仿真，通过不断改变超声波模块的距离值，观察左右侧电机转动的参数，实现题目要求避障的功能。

图示, 示意图

描述已自动生成

图5 PCB仿真设计分析图

五、测试的方案与测试结果

5.1、测试方案

5.1.1检查电路

任何电路，通电调试之前都必须再仔细的检查核对电路，，看是否有连线错误的的部分。最重要的是正负极有没有接反，短接，然后是元器件的正负极，还有元器件的引脚有没有焊实。

5.1.2通电观察

检查电路无误后，首先要做的不是用仪器观测波形和数据，二是先观察整个电路是否有冒烟、异常气味、发光放电、元器件发烫等现象。如果有，先断电， 继续检查电路，找出故障。再重新通电。

5.1.3观察波形与数据

通电观察无异常后，用电压表和示波器测试输入信号和输出信号是否和期望， 代码编写的结果一致。

5.1.4组装测试

待前几步骤都无误后，就可以组装整个硬件系统测试了。如果测试结果不符合设计要求，应检查问题所在，一般是对某些元件参数加以调整和改变，当然局部返工也是有可能的，这就需要我们有足够的耐心去处理出现的意料之外的问题。

5.2、测试条件与仪器

自行搭建测试场地，为尽量符合比赛场地，需要我们在室内日光灯的照明下完成测试。测试使用的仪器有便携式示波器，万用表，秒表等。

5.3、测试结果（数据）及分析

|  |  |  |  |
| --- | --- | --- | --- |
| 次数 | 完成时间/s | 碰撞障碍物次数 | 中心标志距离障碍物的距离超过 40cm次数 |
| 测试1 | 28 | 1 | 2 |
| 测试2 | 26 | 1 | 2 |
| 测试3 | 25.4 | 1 | 2 |
| 测试4 | 24 | 0 | 1 |
| 测试5 | 24 | 0 | 1 |
| 测试6 | 22 | 0 | 1 |
| 测试7 | 16 | 0 | 0 |
| 测试8 | 16 | 0 | 0 |
| 测试9 | 16 | 0 | 0 |

数据分析：

通过不断增大控制电机PWM参数，减少完成的总时间、调试修改超声波模块距离判别参数、MPU6050初始参数改变转弯的半径、转弯的角度，优化减少碰撞障碍物次数和中心标志距离障碍物的距离。通过测试，本作品可以顺利完成题目要求。

六、总结与感悟

通过这四天三夜的对本项目的设计，我们对大学期间所学知识进行了有效的实践，是一次宝贵的经历，对后期比赛，工作提供了经验，更加有效地加深了对硬件电路、嵌入式系统开发的学习，激发了对硬件电路、嵌入式系统开发的热爱。在做本项目的过程中，我们遇到了很多的困难，因为这次疫情的原因，比赛一直推迟到 12月。加上疫情快递停运，好在在网上购买的元器件在比赛前最终也陆续到达。我们拿到各种元器件之后，还有我们提前制作打好的PCB板，我们分工明确，将元器件进行焊接，在焊接过程中有很多贴片式的电阻电容很考验我们的焊接能力，经过一天的努力，焊接结束并且测试PCB板的功能无误。随后完成小车模型和试验场地的搭建，加上我们本身有STM32系统开发的经验，调试起来很迅速。在调试过程中，小车并不能很好的行走直线，我们查阅资料，最终确定加入PID算法，同时为有效确定小车转弯时转弯角度我们采用MPU6050陀螺仪进行辅助设计。当然整个项目中的困难还有很多，我和队友齐心协力一一克服，这也很大地增进了我们同学之间的友谊。我们将会向着硬件电路、嵌入式系统、机电一体化更加深入的方向前进。

参考文献

1. 杨和平等.单片机原理与应用.北京:机械工业出版社, 2008.
2. 陈炳和.计算机控制系统基础.北京:北京航天航空大学出版社，2001.
3. 康光华.模拟电子技术基础（第四版）.北京:高等教育出版社，2014.
4. 康光华.数字电子技术基础（第四版）.北京:高等教育出版社，2014.
5. 张菊鹏等.计算机硬件技术基础(第二版）.北京:清华大学出版社，2000.

**附录1电气原理图**

日历

描述已自动生成

**附录2主要程序代码**

int main(void)

{

static u8 z=0;

u16 adcx;

float temp;

//u8 key;

float pitch,roll,yaw; //欧拉角

short aacx,aacy,aacz;//加速度传感器原始数据

short gyrox,gyroy,gyroz;//陀螺仪原始数据

NVIC\_PriorityGroupConfig(NVIC\_PriorityGroup\_2); //设置NVIC中断分组2:2位抢占优先级，2位响应优先级

delay\_init();//延时函数初始化

uart\_init(115200);

uart3\_init(115200);

LED\_Init(); //初始化与LED连接的硬件接口

TB6612\_GPIO\_Init();

EXTIX\_Init();

TCRT5000\_Init();

OLED\_Init();

OLED\_Clear();

SR04\_GPIO\_Init(); //超声波PA0

Adc\_Init();//ADC初始化

MPU\_Init();//初始化MPU6050

TIM3\_PWM\_Init(999,1439); //舵机PWM

TIM1\_PWM\_Init(1999,719); //电机的PWM

TIM\_SetCompare1(TIM1,500);

TIM\_SetCompare4(TIM1,500);

TIM2\_Cap\_Init(0XFFFF,72-1);//以1Mhz的频率计数

while(1)

{

if(mpu\_dmp\_get\_data(&pitch,&roll,&yaw)==0)

{

MPU\_Get\_Accelerometer(&aacx,&aacy,&aacz);//得到加速度传感器数据

MPU\_Get\_Gyroscope(&gyrox,&gyroy,&gyroz);//得到陀螺仪数据

t++;

}

//OLED 显示ADC测量

adcx=Get\_Adc\_Average(ADC\_Channel\_4,10);

temp=(float)adcx\*(3.3/4096);

sprintf((char \*)string,"U:%.2f ",(temp\*5));

OLED\_ShowString(12,0,string,16);

//OLED显示距离功能

sprintf((char \*)string,"D:%d ",SR04\_Distance());

OLED\_ShowString(12,3,string,16);

//OLED显示模式功能

sprintf((char \*)string,"Mode:%d ",gyrox);

OLED\_ShowString(12,6,string,16);

//串口输出

UsartPrintf(USART3,"Mode:%d\r\n",gyrox);

UsartPrintf(USART3,"U:%.2f\r\n",(temp\*5));

UsartPrintf(USART3,"D:%d \r\n",SR04\_Distance());

if(SR04\_Distance()>30)// 前方无障碍物

{

Forward();

delay\_ms(500);

}

if(SR04\_Distance()==30)// 前方有障碍物

{

z++;

if(z==1)

{

Leftward();

delay\_ms(500);

if(pitch==30)

{

Forward();

delay\_ms(500);

Rightward();

delay\_ms(500);

if(pitch==-30)

{

Forward();

delay\_ms(500);

Rightward();

if(pitch==0)

{

Forward();

delay\_ms(500);

Rightward();

}

}

}

}

}

}

}