

**电子工艺实习报告**



**姓 名**

**学 院 电子工程学院**

**专 业 电子信息科学与技术**

**班 级 2014211202**

**学 号 2014210**

**班内序号**

**指导教师 陈玉波**

**2016年9 月**

**北京邮电大学实习报告**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **实习名称** | **电子工艺实习** | | **学 院** | **电子工程学院** | |
| **学生姓名** |  | **班 级** | **2014211202** | **学 号** | **20142108** |
| **实习时间** | **2016年8.29—9.9** | **实习地点** | **主楼710** | | |
| **实**  **习**  **内**  **容** | 实习题目：电子工艺实习的基本技能（手工焊接技术、拆焊技术等）、发光二极管阵列的焊接与调测、智能平衡车的安装调测  实习进度安排：  8.29-8.30：认识焊接工具，学习并掌握手工焊接技术、拆焊技术等基本技能，大量训练掌握手工焊接的技巧    8.31-9.2：用万能板焊接安装一个发光二极管交替闪烁电路；初步了解单片机相关知识及其开发环境；发光二极管阵列的焊接与调测  9.5：熟悉电子原件市场、发元器件，准备安装小车  9.5-9.8：完成智能平衡车的安装调试，验收作品  9.9：总结得失，写实习报告 | | | | |
| **学生**  **实习**  **总结**  （附页，不少于2000字） | 见附页 | | | | |
| **实**  **习**  **成**  **绩**  **评**  **定** | 遵照实习大纲并根据以下三方面按五级分制（优秀、良好、中等、及格、不及格）综合评定成绩：  1、思想品德、实习态度、实习纪律等  2、技术业务考核、笔试、口试、实际操作等  3、实习报告、分析问题、解决问题的能力  **实习评语**:  **实习成绩**:  指导教师签名： 实习单位公章  年 月 日 | | | | |

目录：

[一、实习任务要求 4](#_Toc461362292)

[二、实习目的 4](#_Toc461362293)

[三、实验过程 4](#_Toc461362294)

[1.焊接 4](#_Toc461362295)

[2.简单的发光二极管交替闪烁电路 5](#_Toc461362296)

[2.1电路原理 5](#_Toc461362297)

[2.2材料清单 6](#_Toc461362298)

[2.3焊接要求 6](#_Toc461362299)

[2.4实验心得 6](#_Toc461362300)

[3.发光二极管阵列的焊接与调测 7](#_Toc461362301)

[3.1根据元器件清单准备元器件 7](#_Toc461362302)

[3.2根据元器件从低到高的顺序进行焊接 7](#_Toc461362303)

[3.3电路检测 8](#_Toc461362304)

[3.4单片机编程、电路调试 8](#_Toc461362305)

[3.5部分代码 9](#_Toc461362306)

[3.6字符设计 10](#_Toc461362307)

[3.7实验心得 12](#_Toc461362308)

[4.智能平衡车的安装调测 12](#_Toc461362309)

[4.1清点智能平衡车材料 12](#_Toc461362310)

[4.2组装只能平衡车 12](#_Toc461362311)

[4.3部分源代码 13](#_Toc461362312)

[4.4平衡车外观设计 16](#_Toc461362313)

[4.5实验心得 17](#_Toc461362314)

# 一、实习任务要求

（1）熟悉焊接用工具（外热式电烙铁、吸锡器等），了解焊接用材料（焊料、助焊剂），正确使用电烙铁，熟练掌握手工焊接技术，按照正确的焊接操作手法进行反复练习，知道什么是高质量的焊点，保障焊点的质量，学会手工拆焊技术。

（2）用万能板焊接安装一个发光二极管交替闪烁电路。

（3）焊接一个发光二极管阵列，通过单片机控制阵列显示图形或字符

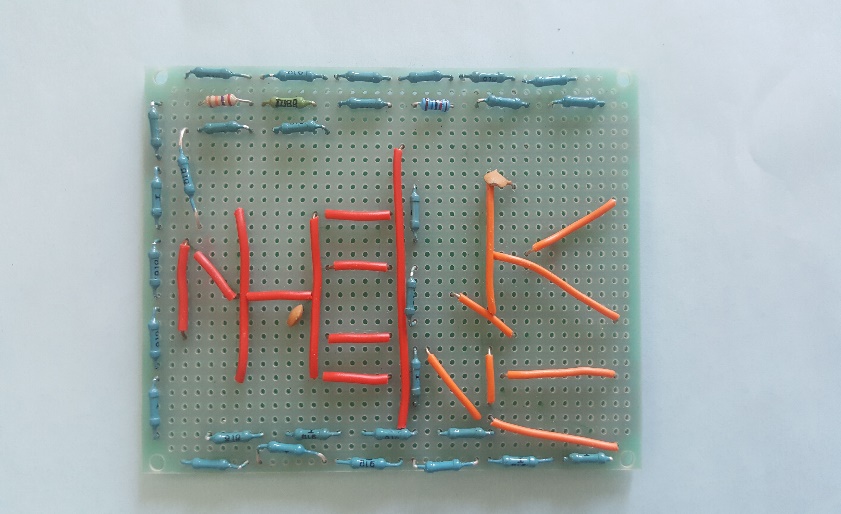
（3）安装调测智能平衡小车，基本要求实现平衡（静态平衡或动态平衡），提升要求实现直行、旋转、拐弯等。

# 二、实习目的

此次实验旨在通过焊接小车与调试小车，锻炼学生的综合能力。其中包括：焊接能力练习，代码编写，和综合调试等。通过此次实验，我们应该掌握基本的焊接技术，电路调试技术和单片机开发的基本能力。

# 三、实验过程

1.焊接

1.1焊接的基本知识

1.2焊接工具及其使用方法

1.3焊接技巧及使用的焊接训练

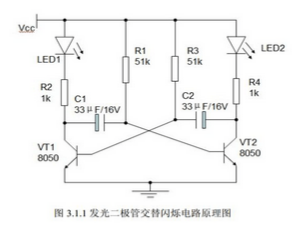
老师通过PPT演示的形势，给我们讲解了焊接时常用的工具有电烙铁、吸锡器、其它常用工具（烙铁架、尖嘴钳、剪刀、斜嘴钳、剥线钳、镊子、切刀等），还有焊接时用的焊料、助焊剂 等，讲解在使用电烙铁时要注意的事项，以及手工焊接的基本手法和要领。

最后，老师通过利用废弃的电容电阻等原件让我们焊接自己名字来锻炼我们的焊接技术。

【图为我与组员（孟琦与蒋欣欣）两名字各取一个组合而成】

2.简单的发光二极管交替闪烁电路

### 2.1电路原理



### 2.2材料清单



### 2.3焊接要求

      先将电路布局规划好，然后按照一定顺序将电路分块安装，再后固定原件，最后用

搭焊的方法焊接导线连接原件。

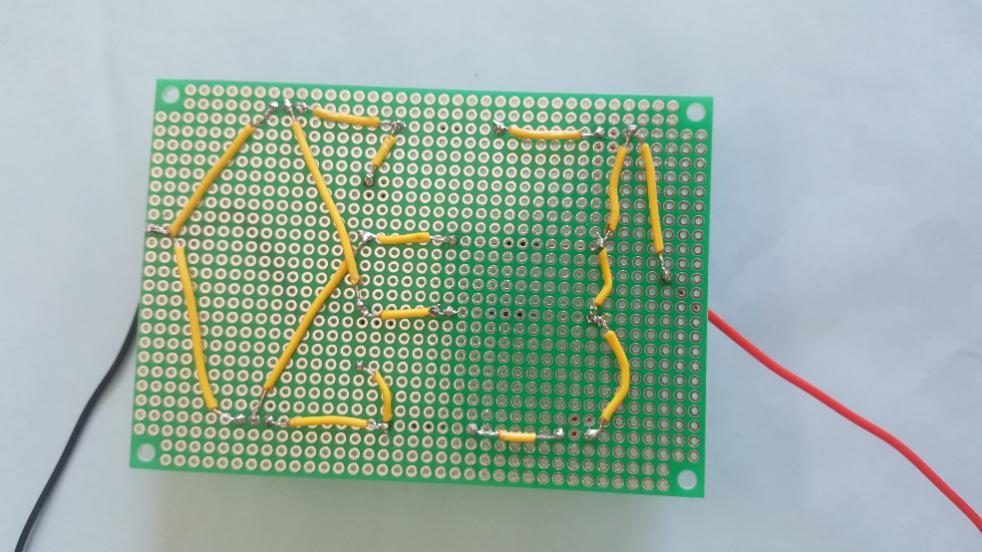
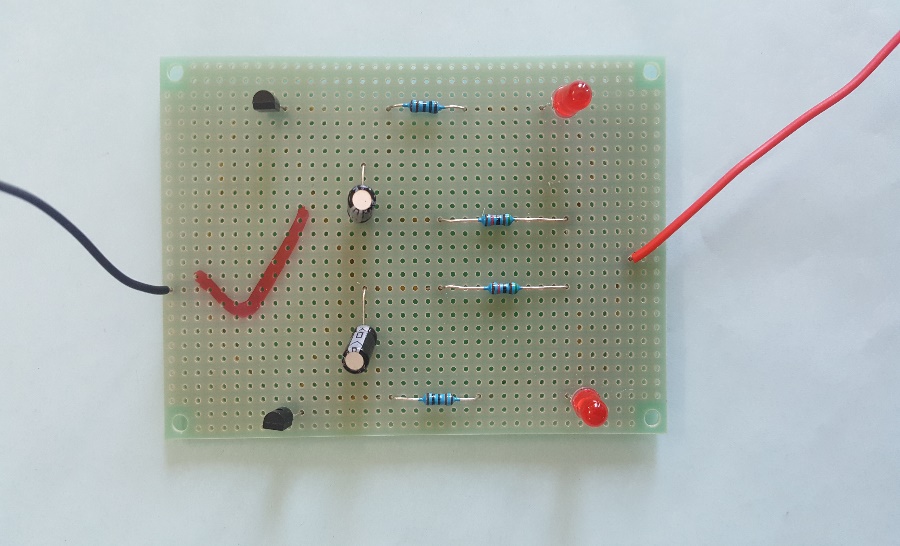
### 2.4实验心得

 在进行这次焊接之前我们已经进行了两天的焊接训练，学习了基本的焊接方法和焊

接技巧，所以在在进行焊接时并没有遇到什么困难。不过由于是第一次焊接实际电路，还是出现了一些可以再改进的状况，比如我们的每一条连接线都使用导线焊接相连，但是由于焊接技术含不成熟，有些短导线绝缘皮会被烫化，易短路，而且焊接位置的高低也不好控制，好在我们组成员合作较好，没有对电路的功能和外观造成太大的的影响。此次试验一次成功，没有经过再调试，为后来小车实验的焊接增加了信心。通过这次焊接，我意识到焊接练习与实际焊接电路的区别，并且初步收获了焊接电路的一些使用技巧。

     经过本次试验，进一步熟悉和掌握了焊接的技巧，了解了焊接实际电路时的必要步骤，同时学习了很多导线的焊接方法（绕焊、搭焊、勾焊等），提高了个人对于电路实验的兴趣，为今后的实际科研中电路实验奠定了基础。

实际效果图：



3.发光二极管阵列的焊接与调测

### 3.1根据元器件清单准备元器件

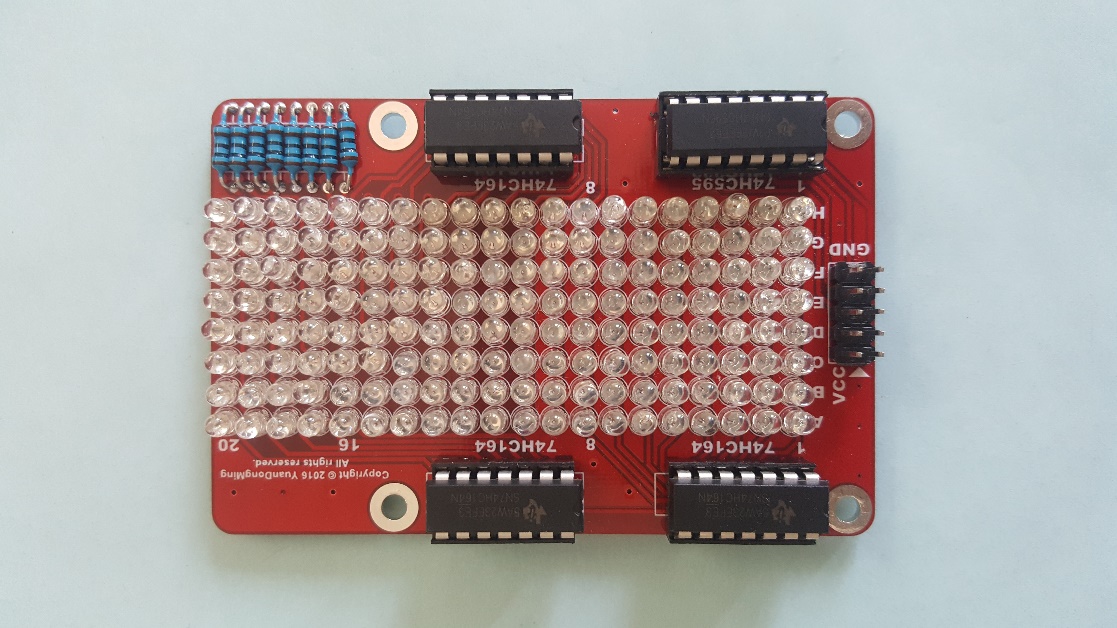
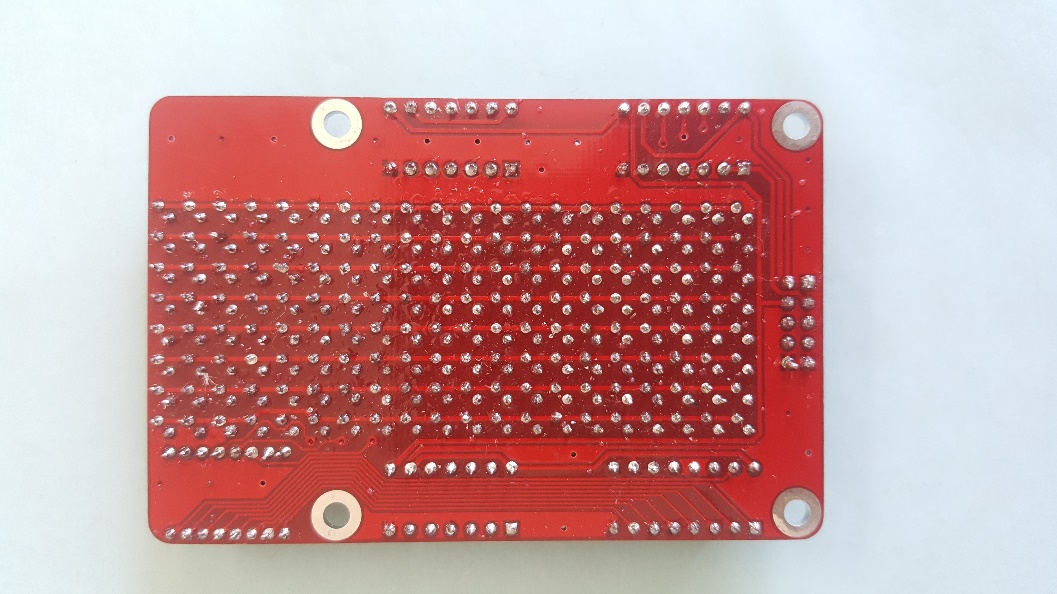


### 3.2根据元器件从低到高的顺序进行焊接

发光二极管，注意正负极性，长脚为正，短脚为负，先焊接，后剪脚；双排针，注意短的一面插入焊盘进行焊接。

### 3.3电路检测

电路检测并用万用表测量 VCC 与 GND 之间的电阻，确保没有短路。

### 3.4单片机编程、电路调试



### 3.5部分代码

int main()

{

RCC\_CFG();

GPIO\_CFG();

SPI1\_CFG();

while(p++<105)

{Led\_RollShow(data0);}

while(q++<105)

{Led\_RollShow(data1);}

while(m++<105)

{Led\_RollShow(data4);}

while(r++<105)

{

//Led\_RollShow(data); //动态输出

Led\_StaticShow(data4); //静态输出

}

while(n++<105)

{Led\_RollShow(data3);}

while(1)

{

//Led\_RollShow(data); //动态输出

Led\_StaticShow(data2); //静态输出

}

}

void Led\_StaticShow(u8 data[])//静态输出

{

u8 i=0;

for(i = 0; i < 20; i++) //20列一列一列输出

{

SPI1\_WriteByte(data[i]);

while(SPI\_I2S\_GetFlagStatus(SPI1, SPI\_I2S\_FLAG\_TXE) == RESET);//判断是否输出结束

delay\_us(50);

BK0\_LOW;

BK0\_HIGH; //时钟上升沿

if(i==0)

{

BK2\_LOW; //时钟上降沿

BK1\_LOW;

BK1\_HIGH;

BK2\_HIGH;

}

else

{

BK1\_LOW;

BK1\_HIGH; //时钟上升沿

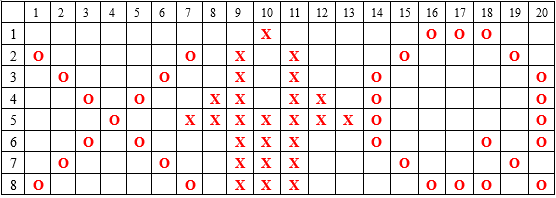
}

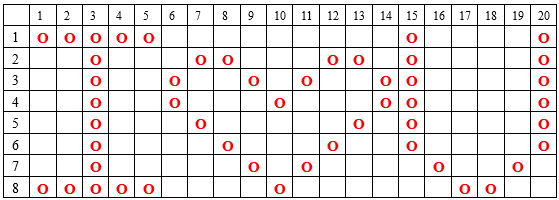
delay\_us(1000);

}

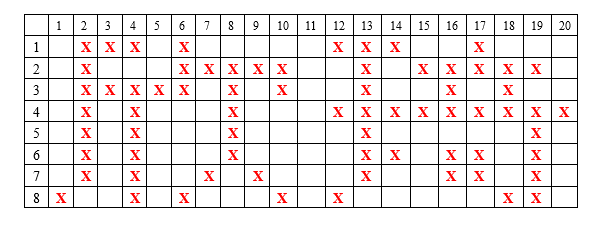
}

### 3.6字符设计

在设计显示的图形时，我们先用了word文档设计了图形，这样不仅方便了我们思维的跳跃性设计，还很方便快捷的来设计。

（表格设计为：X**⇮**Q）

（表格设计为：I**♡**U）

（表格设计为：欣琦）

部分代码：

void RCC\_CFG(void);

void GPIO\_CFG(void);

void SPI1\_CFG(void);

void SPI1\_WriteByte(u8 TxData);

void delay\_us(u32 nus);

void Led\_RollShow(u8 data[]);

void Led\_StaticShow(u8 data[]);

u8data0[20]={0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff};

u8data1[20]={0x82,0x44,0x28,0x10,0x28,0x44,0x92,0x18,0xFE,0xFF,0xFE,0x18,0x10,0x3C,0x42,0x81,0x81,0xA1,0x42,0xBC};

u8data2[20]={0x82,0x44,0x28,0x10,0x28,0x44,0x92,0x18,0xFE,0xFF,0xFE,0x18,0x10,0x3C,0x42,0x81,0x81,0xA1,0x42,0xBC};

u8data3[20]={0x81,0x81,0xFF,0x81,0x81,0x0C,0x12,0x22,0x44,0x88,0x44,0x22,0x12,0x12,0x3F,0x40,0x80,0x80,0x40,0x3F};

u8data4[20]={0x80,0x7f,0x05,0xfd,0x04,0x87,0x42,0x3E,0x42,0x86,0x00,0x89,0x7F,0x29,0x0A,0x6E,0x6B,0x8E,0xFA,0x08};

### 3.7实验心得

在这个发光二极管阵列的焊接与调测中，从一开始的焊接160个发光二极管到最后运用程序和单片机来显示图形中，有很大的收获。其中在焊接二极管时，由于引脚有正负之分，而需要焊接的数量又很多，所以需要很大的耐心来进行操作。在用杜邦线连接发光二极管阵列电路和单片机时，一定要一一对应连接管脚。

应用程序来编写图形时，是用十六进制来编写的，所以在换算进制方面时，头脑要清晰，否则容易出错误。

在整个焊接和代码的编写，外观的设计，都是我和队友一起交替完成的。所以说团结就是力量，最后我们完成的效果至少没有让我们失望，但是希望以后我们在各方面能力上更加有所突破，完成得更好。

4.智能平衡车的安装调测

### 4.1清点智能平衡车材料

### 4.2组装只能平衡车

A．通过焊接顺序，焊接电机驱动板；

B．在电池盒固定板上焊接电源端子，把电池盒焊接到电池盒固定板的另一面，把三根导线分别连接到电池盒板的蓝色端子上；

C．在每个TT电机上焊接2根电源线，铝块也根据要求焊接到电机上，吧车轮装配到TT机上；

D．将三根导线的另一端接到另外一个三端电源上，用万用表测试是否连通；

E．把单片机和电池板分别固定在小车底板的上下面，电机驱动板固定在单片机核心板上，电机电源线分别接到单片机核心板的MOTO-A和MOTO-B上；

F．用杜邦线将单片机核心板和电机驱动板的电源及地连在一起，检查所有安装，确保无误后，安装电池。

### 4.3部分源代码

A:

int main(void)

{

int i=0;

Stm32\_Clock\_Init(9);

delay\_init(72);

GPIO\_Configuration();

USART1\_Config();

I2C\_config();

InitSensor();

SPI1\_Init();

PWM\_Init(3599,0);

TIM3\_Init(99,7199);

for(i=0;i<20000;i++){

Angle\_t=0;

}

for(i=0;i<20000;i++){

Angle\_t=3;

}

for(;;)

{

Angle\_t=0;

}

for(;;)

{

Angle\_t=1;}

}

#include "init.h"

float Gyro\_y;

float Angle\_gy;

float Accel\_x;

float Angle\_ax;

float Angle;

float Angle\_t;

float Angle\_Control\_P = 100;

float Angle\_Control\_D = 8;

float Angle\_Control\_I =0.3;//2

float KIn=0;

float Angle\_Control\_Out\_Max = 3000.0;

float g\_Angle\_Control\_Out=0;

float g\_Left\_Motor\_Dead\_Value =1030;

float g\_Right\_Motor\_Dead\_Value = 1000;

B:

int main(void)

{

int i;

Stm32\_Clock\_Init(9);

delay\_init(72);

GPIO\_Configuration();

USART1\_Config();

I2C\_config();

InitSensor();

SPI1\_Init();

PWM\_Init(3599,0);

TIM3\_Init(99,7199); /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

for(i=1;i<30000;i++)

{

Angle\_Calculate();

Angle\_Control();

Set\_Motor\_Pwm\_go();

}

for(i=1;i<8000;i++)

{

Angle\_Calculate();

Angle\_Control();

Set\_Motor\_Pwm\_Left();

}

/\*for(i=1;i<8000;i++)

{

Angle\_Calculate();

Angle\_Control();

Set\_Motor\_Pwm\_go();

}\*/

for(i=1;i<8000;i++)

{

Angle\_Calculate();

Angle\_Control();

Set\_Motor\_Pwm\_Right();

}

for(i=1;i<800000;i++)

{

left\_forward(4000);

right\_backward(4000);

}

for(i=1;i<800000;i++)

{

right\_forward(4000);

left\_backward(4000);

}

for(;;)

{

Angle\_Calculate();

Angle\_Control();

Set\_Motor\_Pwm();

}

}

#include "init.h"

float Gyro\_y;

float Angle\_gy;

float Accel\_x;

float Angle\_ax;

float Angle;

//---------------------------- ------------------------------//

float Angle\_Control\_P = 10;

float Angle\_Control\_D = 0.5;

float Angle\_Control\_Out\_Max = 7000.0;

float g\_Angle\_Control\_Out=0;//½Ç¶È¿ØÖÆÊä³ö

//---------------------------- ---------------------------------//

float g\_Left\_Motor\_Dead\_Value =2400;//

float g\_Right\_Motor\_Dead\_Value = 2400;

/\*\*\*\*\*\*\*\*\*\*\*\*¨\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

### 4.4平衡车外观设计

平衡车的外观设计是我跟队友一起设计的，我们先是在网上选择好我们认为比较不错的的纸模，之后组装后放在智能平衡车上固定好。但是发生比较难过的事情就是，我们组装好之后，发现纸模有点重，放在智能平衡车上时，重心过高了，所以平衡小车难以平衡下来，最后通过减少一部分纸模之后，才调配好重量。重新更换代码数据之后，小车终于平衡了，换上另外一个功能的代码也能够实现了。

### 4.5实验心得

通过这个智能平衡小车的组装，调配，到程序设计，我们学会了怎么运用好Keil uVision5这个软件，发现不只是一个程序函数就能够改变小车的某个功能，发现了几个函数的数据组合之后，也能够改变这个小车的功能。自己添加小车旋转，直行的函数，这一系列的工作其实有苦有乐，发现在调配数据时是最难过的，因为要一组一组的调配，一个一个的试调，但是在最好程序问题解决之后，看到小车的各种功能都实现时，心情是欢呼雀跃的，也希望我们以后在各种实验中能够都学到更多的东西，从而运用到社会实践中去。加油加油！！