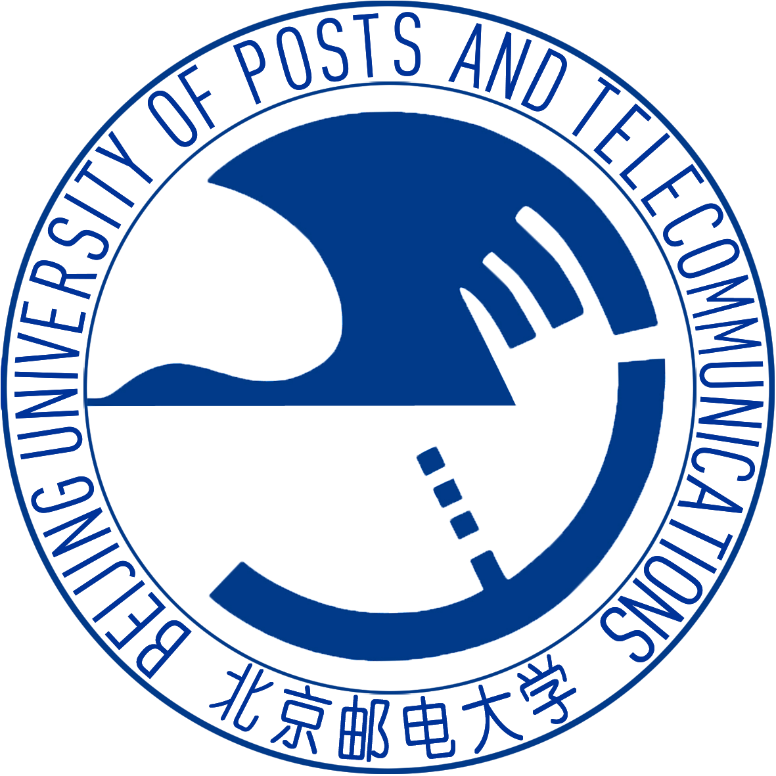
北京邮电大学

电子工艺实习

实习报告



学院：信息与通信工程学院

班级：2013211124

姓名：曹爽

序号：06

2015年7月17日

**摘　要**

电子工艺实习是电子信息类专业必须掌握的一门课程，本次实习从焊接入手，通过制作智能迷宫小车这一目标，综合练习了元器件焊接、电路焊接、小车装配、单片机程序设计、综合调试和异常处理等，同时也锻炼了团队合作能力。本文将以图文并茂的方式，具体详细地叙述实习的整个过程，并对实习过程中出现的问题和经验教训加以分析总结。

**关键词：**智能小车、焊接、单片机、电子工艺、实习

**目　录**

[一、 实习内容 4](#_Toc424886180)

[二、 实习过程 4](#_Toc424886181)

[1. 常用工具和耗材及其使用方法 4](#_Toc424886182)

[2. 焊接练习 5](#_Toc424886183)

[3. 发光二极管交替闪烁电路的实现 6](#_Toc424886184)

[(1) 任务要求 6](#_Toc424886185)

[(2) 原理图 6](#_Toc424886186)

[(3) 元器件清单 7](#_Toc424886187)

[(4) 元器件焊接注意事项 7](#_Toc424886188)

[(5) 焊接效果图 7](#_Toc424886189)

[4. 智能小车的安装与调试 8](#_Toc424886190)

[(1) 原理图和装配图 8](#_Toc424886191)

[(2) 材料清单 8](#_Toc424886192)

[(3) 焊接与装配 10](#_Toc424886193)

[(4) 控制电路测试 11](#_Toc424886194)

[(5) 驱动电路测试 12](#_Toc424886195)

[(6) 整机测试 13](#_Toc424886196)

[5. 程序的编写与调试 13](#_Toc424886197)

[(1) 对小车检测程序的理解 13](#_Toc424886198)

[(2) 程序的编写与测试 13](#_Toc424886199)

[(3) 算法的改进与优化 15](#_Toc424886200)

[6. 外观设计 15](#_Toc424886201)

[三、 实现功能 17](#_Toc424886202)

[四、 人员分工 17](#_Toc424886203)

[五、 问题总结和心得体会 17](#_Toc424886204)

[六、 参考文献 19](#_Toc424886205)

[七、 附：算法源程序 19](#_Toc424886206)

# 实习内容

1. 了解和掌握常用工具的使用方法。
2. 练习焊接技术，掌握各元件的焊接方法。
3. 焊接实现发光二极管交替闪烁电路。
4. 利用焊接技术进行智能小车的安装与调试。
5. 学习单片机的相关知识，并利用单片机C语言对小车程序进行调试。
6. 对小车进行外观设计和包装。
7. 智能小车的调试与验收。
8. 撰写电子工艺实习报告。

# 实习过程

## 常用工具和耗材及其使用方法

在课程的开始，老师先给我们讲解了电子工艺实习过程中常用的工具和耗材。焊接工具包括电烙铁（图2.1.1）、吸锡器（图2.1.2）、剥线钳（图2.1.3）等。耗材包括焊锡丝（图2.1.4）、助焊剂等。

**图2.1.1 外热式电烙铁 图2.1.2 吸锡器**

 ****

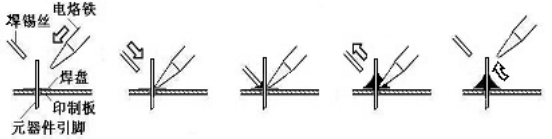
**图2.1.3 剥线钳 图2.1.4 焊锡丝**

随后，我又了解了它们各自的使用方法及注意事项，比如：

* 电烙铁通电前首先检查电烙铁的电源线有无破损，防止短路。
* 电烙铁不用时应放在烙铁架上，不能随意放置。
* 电烙铁使用前先用螺丝刀拧紧烙铁头，防止传热不良。
* 电烙铁使用时可以先在烙铁头涂一层锡以便于焊接。

## 焊接练习

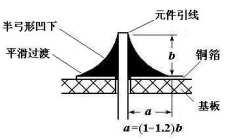
首先，老师给每个组发了一块万用板和一些废弃元件供我们练习焊接。在焊接过程中要遵循五步操作法（图2.2.1），即准备施焊、加热焊件、送入焊丝、移开焊丝、移开烙铁。



**图2.2.1 手工焊接五步操作法**

在实际焊接过程中，要注意以下问题：

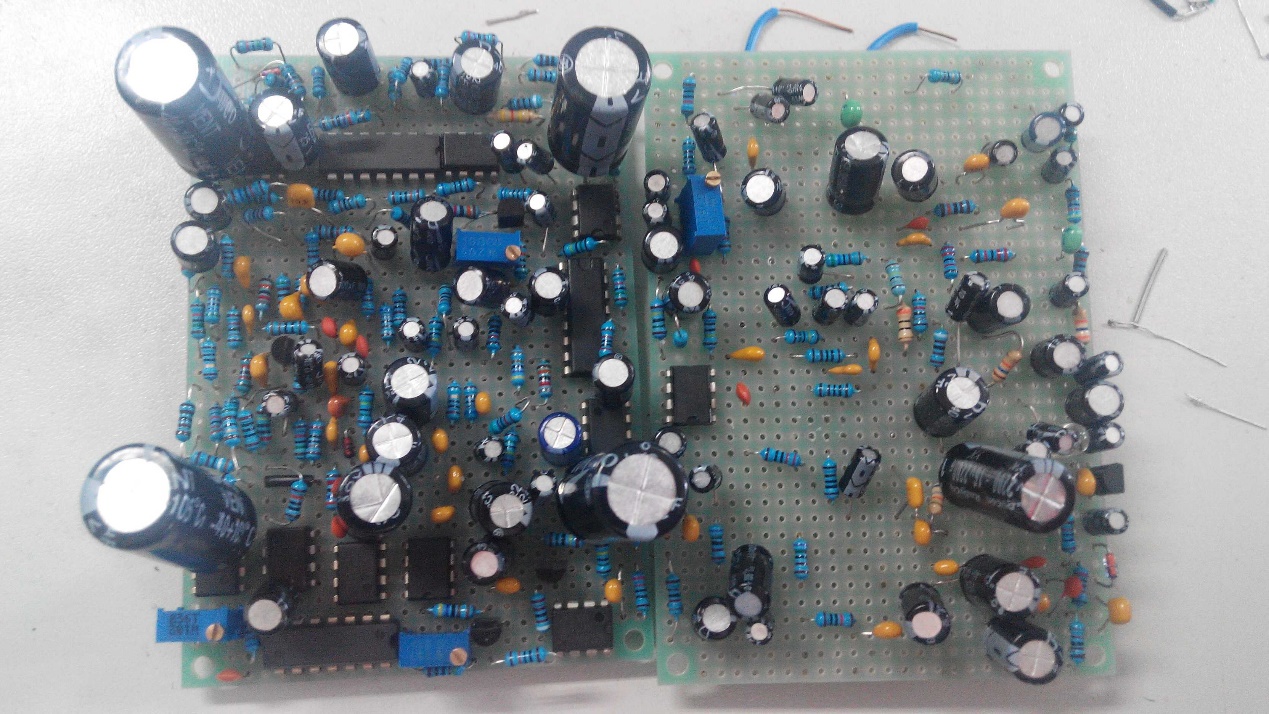
* 焊锡丝和烙铁头分别在引脚的两侧，各与引脚大致呈45度角，用热传导加热焊锡丝，焊锡丝不要碰到烙铁头。
* 焊接时不要过度挤压烙铁头。
* 焊接时要注意焊点应该与引脚充分接触，外观应平整光滑（图2.2.2），并不是锡量越多越好。尤其要避免虚焊、空洞、与焊盘剥离等情况。
* 焊接时元器件应尽量紧贴万用板。
* 焊接完毕过后要将元器件多出来的引脚剪断，只保留焊点即可。



**图2.2.2 焊点形状**

焊接时如果遇到焊接不理想的情况，还需要用吸锡器进行拆焊，具体操作是：先将吸锡器的活塞按压至最低处，之后用电烙铁将需要拆焊的焊点上的锡熔化，再将吸锡器对准熔化的锡，按压吸锡器上的按钮，使吸锡器内部压强迅速减小，从而将熔化的锡吸走，达到吸锡的目的。之后重新焊接即可。

图2.2.3是我们组焊接练习结束后的两块万用板。



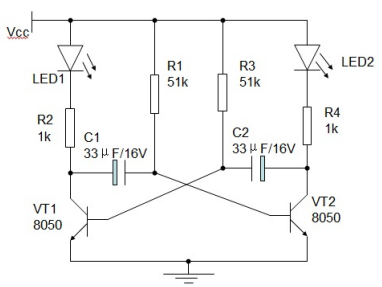
**图2.2.3 焊接练习后的万用板**

## 发光二极管交替闪烁电路的实现

### 任务要求

用万用板焊接安装一个发光二极管交替闪烁电路，

### 原理图



**图2.3.1 发光二极管交替闪烁电路原理图**

### 元器件清单

**表2.3.1 发光二极管交替闪烁电路元器件清单**

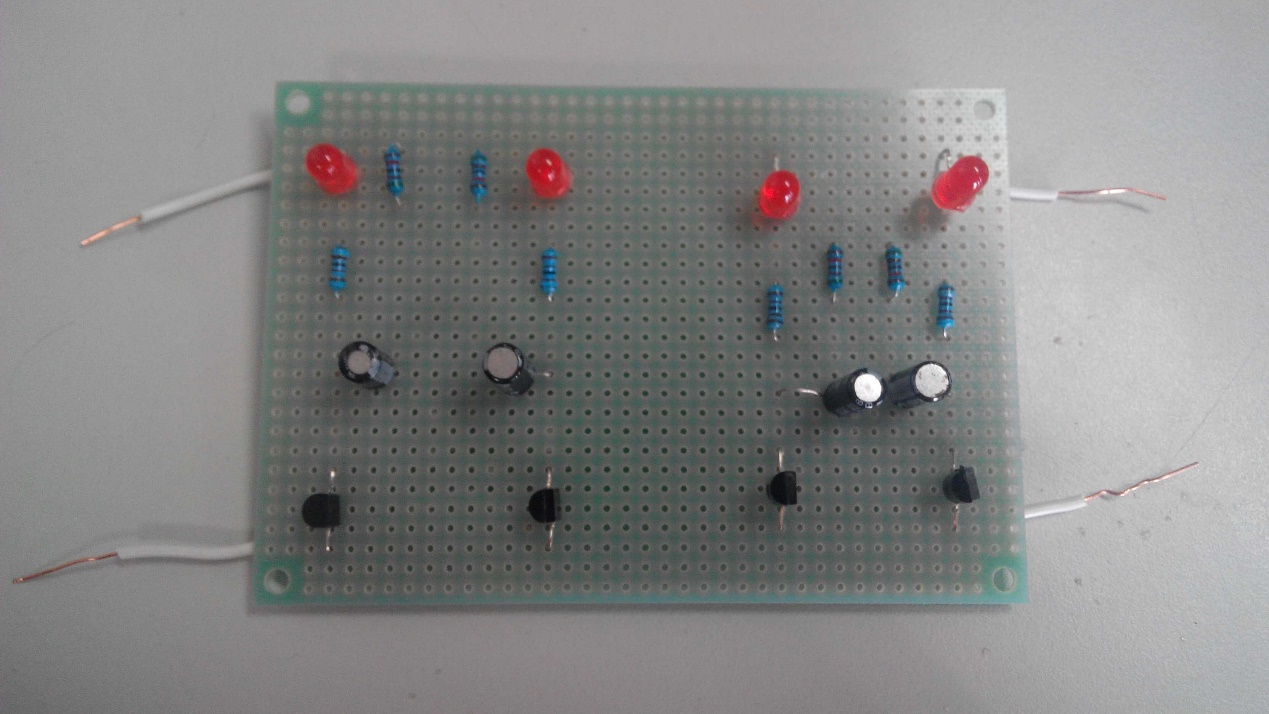
|  |  |  |  |
| --- | --- | --- | --- |
| 元器件名称 | 规格型号 | 数量 | 编号 |
| 电阻 | 1kΩ，1/8W | 2 | R1，R3 |
| 51kΩ，1/8W | 2 | R2，R4 |
| 电容 | 3μF，16V | 2 | C1，C2 |
| 三极管 | 8050 | 2 | VT1，VT2 |
| 发光二极管 |  | 2 | LED1，LED2 |

### 元器件焊接注意事项

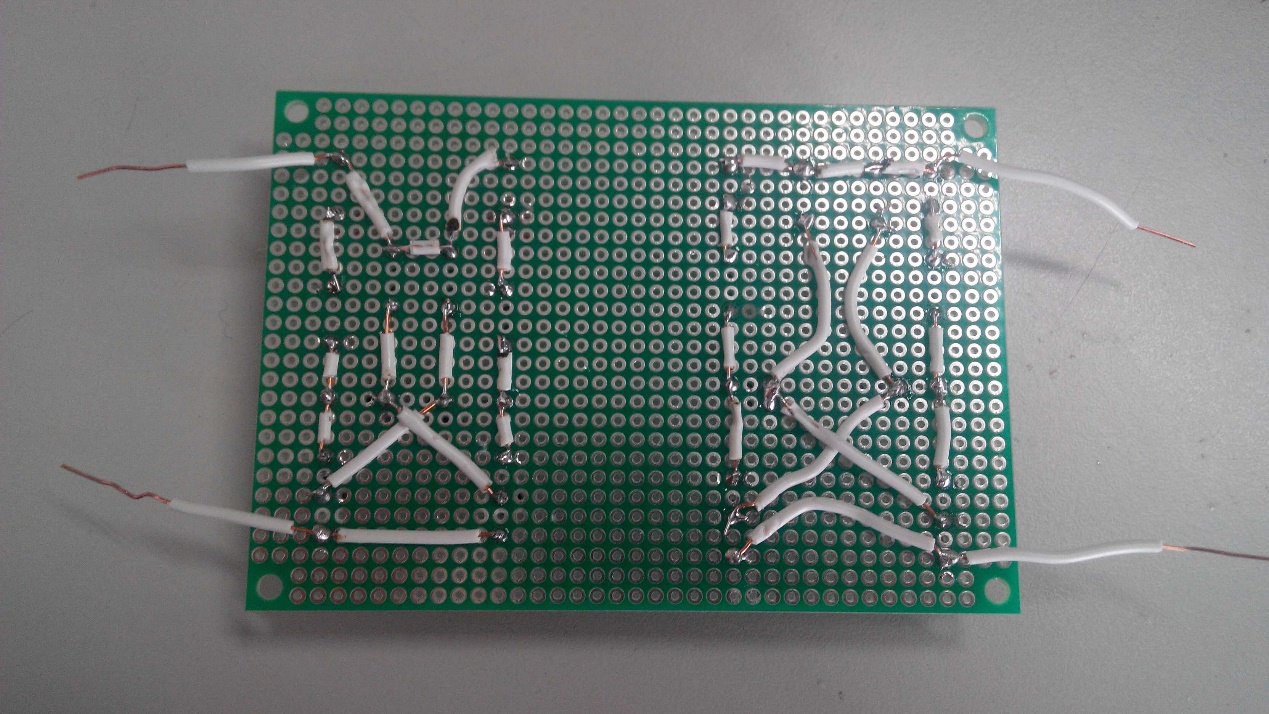
最好先在纸上规划好电路的安装图。在焊接元器件时，应注意元器件紧贴万用板表面。在焊接三极管等元件时，管脚应尽量分散开，以便于焊接和连线时能有较大空间。在焊接电容和发光二极管等有极性的元件时，应特别注意极性（因为剪掉管脚后就难以再检查极性），长管脚对应正极，短管脚对应负极。

在万用板背面连线时，应先剪取合适的长度之后剥线、镀上一层锡之后再与焊点进行焊接，这样更容易焊接。同时要注意不要烫坏绝缘外皮，以免短路。

### 焊接效果图



**图2.3.2 焊接效果（正面）**



**图2.3.2 焊接效果（背面）**

## 智能小车的安装与调试

### 原理图和装配图

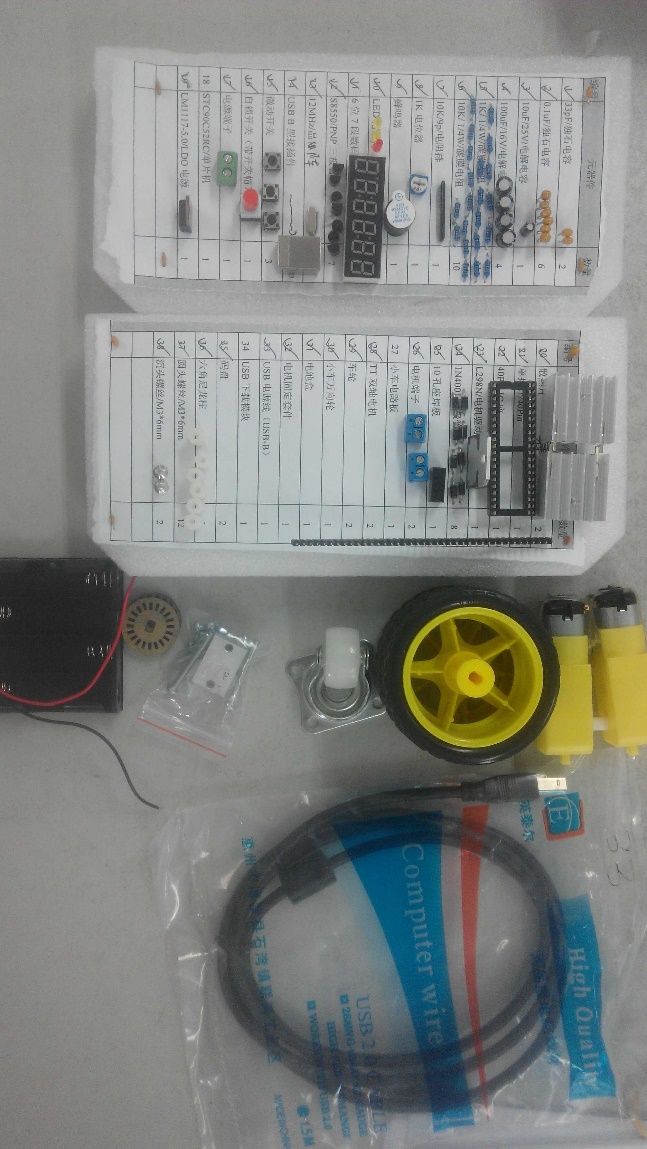
详见《电子工艺实习教程》第19页至22页。

### 材料清单

**表2.4.1 智能小车材料清单**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 编号 | 元器件 | 数量 |  | 编号 | 元器件 | 数量 |
| 1 | 33pF/独石电容 | 2 | 20 | 散热片 | 2 |
| 2 | 0.1uF/独石电容 | 6 | 21 | 单排针/40Pin | 1 |
| 3 | 10uF/25V/电解电容 | 1 | 22 | 40Pin IC座 | 1 |
| 4 | 100uF/16V/电解电容 | 4 | 23 | L298N/电机驱动芯片 | 1 |
| 5 | 1K/1/4W/碳膜电阻 | 11 | 24 | 1N4001/二极管 | 8 |
| 6 | 10K/1/4W/碳膜电阻 | 10 | 25 | 10孔座焊板 | 1 |
| 7 | 10K/9p/电阻排 | 1 | 26 | 电机端子 | 2 |
| 8 | 1K电位器 | 1 | 27 | 小车电路板 | 1 |
| 9 | 蜂鸣器 | 1 | 28 | TT双轴电机 | 2 |
| 10 | LED指示灯 | 3 | 29 | 车轮 | 2 |
| 11 | 6位7段数码管 | 1 | 30 | 小车万向轮 | 1 |
| 12 | S8550/PNP三极管 | 7 | 31 | 电池盒 | 1 |
| 13 | 12MHz/晶体 | 1 | 32 | 电机固定套件 | 1 |
| 14 | USB-B型接插件 | 1 | 33 | USB电源线(USB-B) | 1 |
| 15 | 微动开关 | 3 | 34 | USB下载模块 | 1 |
| 16 | 自锁开关(带开关帽) | 1 | 35 | 码盘 | 2 |
| 17 | 电源端子 | 1 | 36 | 六角尼龙柱 | 6 |
| 18 | STC90C52RC/单片机 | 1 | 37 | 圆头螺丝/M3\*6mm | 12 |
| 19 | LM1117-5.0/LDO电源 | 1 | 38 | 沉头螺丝/M3\*6mm | 2 |

实发材料如图2.4.1所示，小车底盘如图2.4.2所示。



**图2.4.1 实发材料**

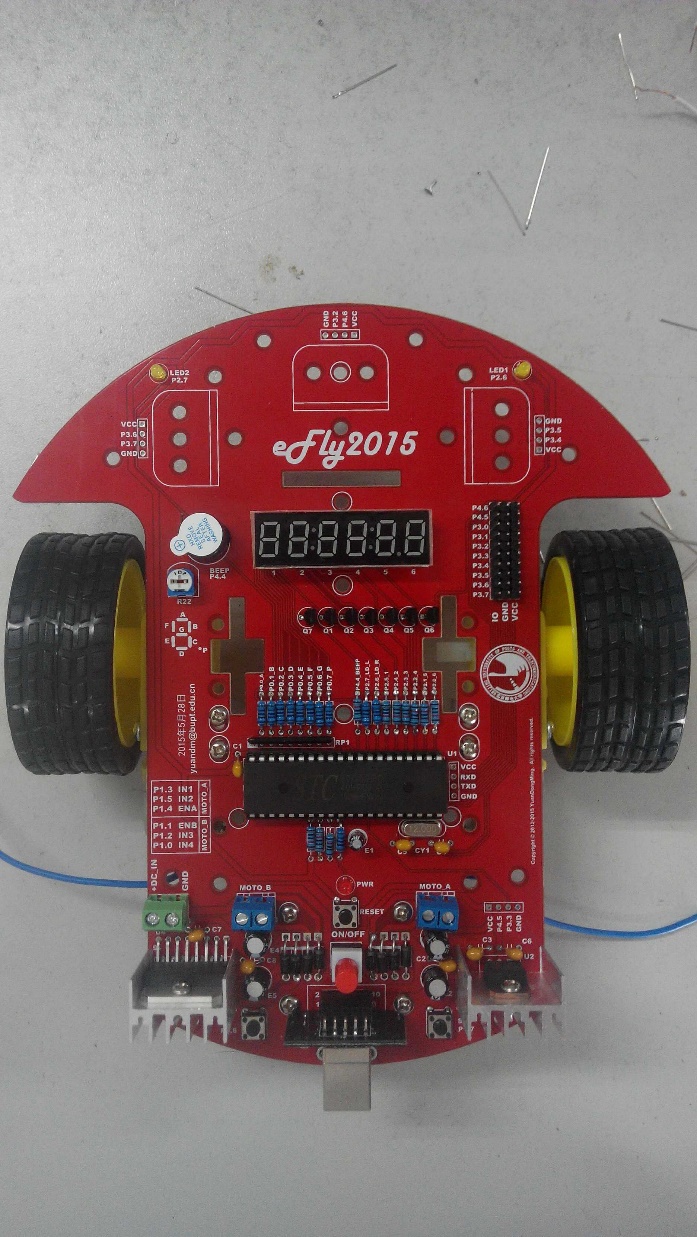


**图2.4.2 小车底盘**

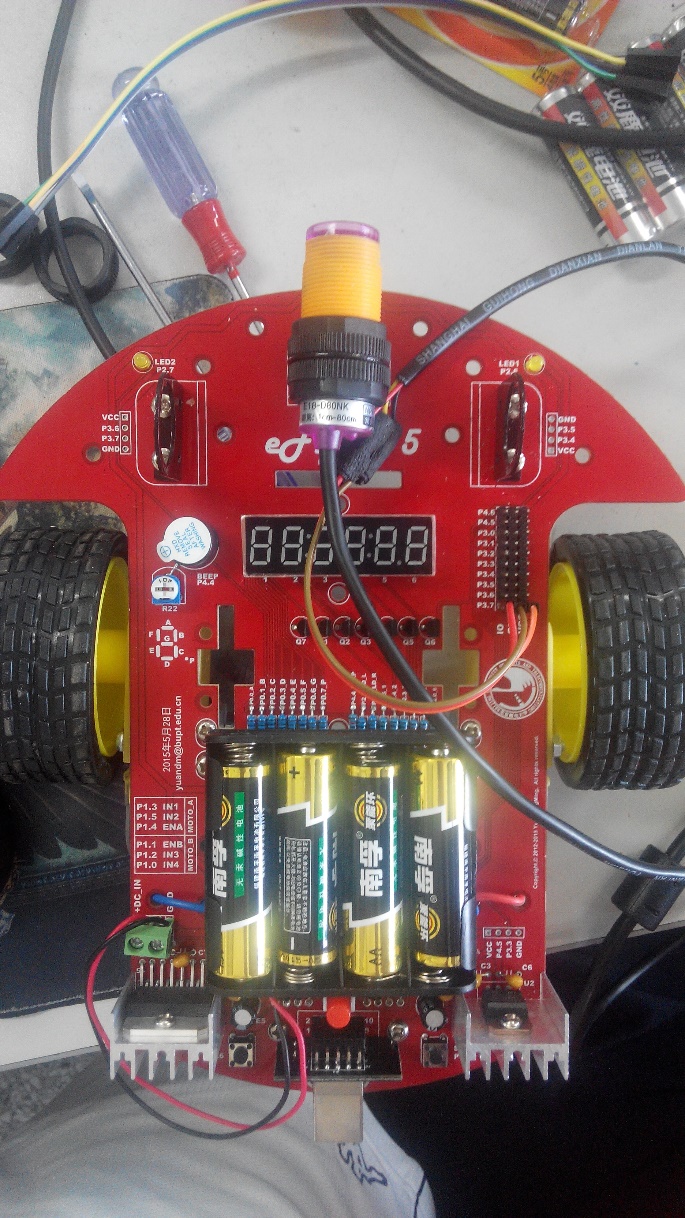
### 焊接与装配

焊接顺序详见《电子工艺实习教程》第24页。焊接时要注意，在焊接比较密集的元件管脚时，要防止因连焊引起的短路。装配时要注意螺丝与螺母之间应拧紧，必要时可以用尖嘴钳夹紧螺母，之后再用螺丝刀拧紧螺丝。单片机要先焊接底座，再将其插上去，不能直接焊接单片机管脚。

最后的效果如图2.4.3和图2.4.4所示。

****

**图2.4.3 焊接效果（同时装配了部分元件）**

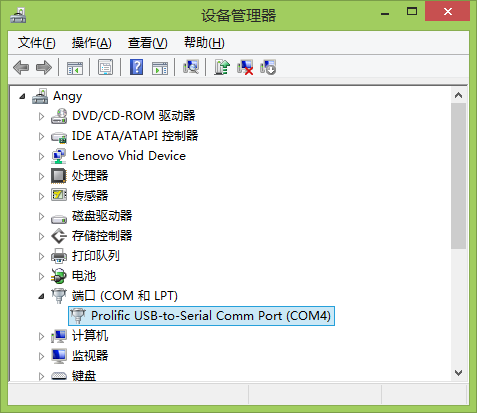
****

**图2.4.4 装配效果（同时安装了一个传感器）**

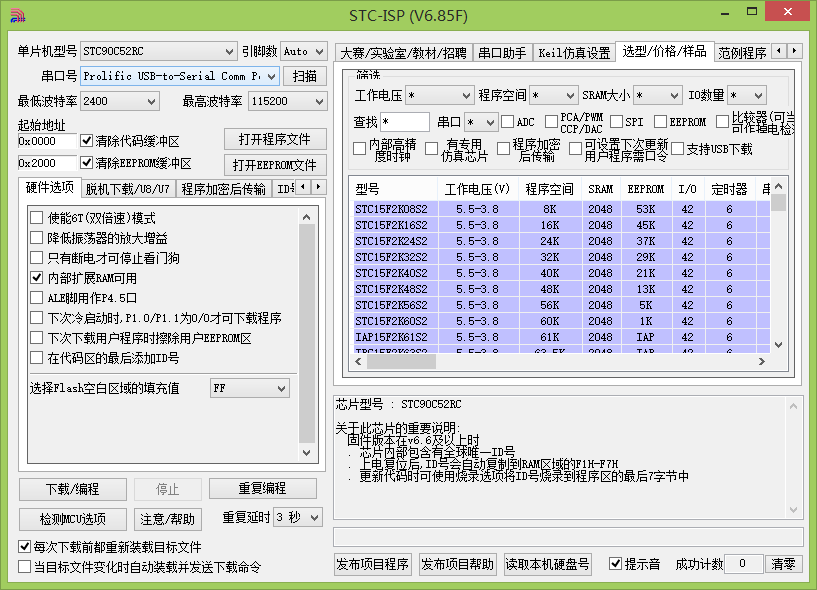
此外，在连接导线时，往往需要将导线焊住。这时同样最好现在导线的铜丝上镀上一层锡，之后再与其他的导线或其他元器件焊接。

### 控制电路测试

* 测量VCC与GND之间是否短路：将万用表调至欧姆档，红黑表笔分别放在VCC和GND管脚处，测得这两极之间的电阻大约为1kΩ，说明VCC与GND之间不短路。
* 同理，测量电源端子之间的电阻约为5kΩ，说明电源端子之间不短路。
* 插上USB线连接到电脑时，会自动安装相关的驱动程序（如图2.4.5）。待电脑识别到USB设备后，开关弹起时，红色LED灯（PWR灯）亮；按下开关后，灯熄灭，符合要求。
* 打开STC-ISP烧录软件，选择单片机型号，这次实习中使用的芯片为“STC90C52RC”。之后串口号若显示“Prolific USB-to-Serial Comm …”则表示与单片机成功连接。若显示“COMxx”（xx为数字），则说明未连接上。如图2.4.6是成功连接时的界面。

****

**图2.4.5 已安装好驱动的端口**

****

**图2.4.6 成功连接时的STC-ISP软件显示界面**

### 驱动电路测试

首先，我们从电路中心网站上下载到小车检测程序包“CHECK.zip”，在程序包附带的“readerme.txt”文档中写明了该程序包实现的功能，包括：

* 车轮的正转和反转
* 六个七段数码管显示012345
* 点亮左上和右上两个LED灯

其中要下载到单片机中的文件为“CHECK.HEX”文件，hex文件是可以烧录到MCU中，并被MCU执行的一种文件格式。如果用记事本打开可发现，整个文件以行为单位，每行以冒号开头，内容全部为16进制码（以ASCII码形式显示）。这样的文件不便于编程，所以在编程时，需要用单片机C语言开发软件“Keil”进行C语言程序编写，再编译生成hex文件供下载到单片机中。具体步骤将会在后面说明。

之后，打开STC-ISP烧录软件，选择单片机型号为“STC90C52RC”。在串口号显示“Prolific USB-to-Serial Comm …”后，点击“打开程序文件”，选择刚才的“CHECK.HEX”文件，确定后点击左下方“下载/编程”按钮。之后按下小车上的红色按钮，待软件界面右下方进度条满后，显示“下载成功”，则说明文件已成功下载。

测试的效果应该如“readerme.txt”文档中写明的相同。如果两个车轮旋转方向相反，则需要交换舵机的正负极重新焊接。如果数码管或LED灯不亮，则需要考虑重新焊接电路。

### 整机测试

整机测试要求小车在至少3秒钟内走直线，否则微调车轮。在之前的“CHECK.HEX”文件中已经实现了车轮的正转和反转，只要将反转和其他无用的代码删去，再延长正转时间即可。如果小车直行时左偏或右偏，则需要调节左右舵机的转速，左快右慢（右转）可以矫正左偏，左慢右快（左转）可以矫正右偏。根据左右舵机和轮子的不同特性，通过调节占空比来调节转速，最终可以使小车走直线。具体的代码调节方法将在下一部分说明。

## 程序的编写与调试

### 对小车检测程序的理解

在编写程序前，我们先对之前下载的小车检测程序进行了分析。用Keil软件打开“CHECK.C”文件，打开后是用C语言编写的小车检测程序。通过观察stop、qianjin、houtui等函数，我们猜测，IN1到IN4应该是控制左右车轮向前后转动方向的变量。随后，我们通过修改代码、下载并观察轮子确定了这四个变量的功能如表2.5.1所示。

**表2.5.1 四个变量对应的功能**

|  |  |
| --- | --- |
| 变量 | 功能 |
| IN1 | 右轮向前转 |
| IN2 | 右轮向后转 |
| IN3 | 左轮向后转 |
| IN4 | 左轮向前传 |

### 程序的编写与测试

在弄懂了原代码后，为了节省电量，我们删去了LED灯和数码管相关的代码，只对小车前进、后退、左转、右转等代码进行编写。

小车上的超声波传感器分别可以探测小车左、前、右三个方向的障碍物，传感器一共有8个状态。其中易于分析的基本状态如下：

* 在正常直行时，左右传感器能感应到墙壁，此时应继续直行，状态不变。
* 当到了应该右转的角落时，左中传感器感应到墙壁，此时右转90度。
* 同理，当右中传感器感应到墙壁时，应左转90度。
* 当走到死路时，左中右传感器都能感应到墙壁，此时应该掉头，即右转180度。
* 当刚走出迷宫时，三个传感器均检测不到障碍物，此时应继续直走。

其他情况再另行分析。

为简明起见，若分别用1和0代表能探测到障碍物和未探测到障碍物，则可以列出传感器的真值表如表2.5.2所示。

**表2.5.2 传感器的真值表**

|  |  |  |  |
| --- | --- | --- | --- |
| L  （左） | M  （中） | R  （右） | F  （输出） |
| 0 | 0 | 0 | 直行 |
| 0 | 0 | 1 | 另行分析 |
| 0 | 1 | 0 | 另行分析 |
| 0 | 1 | 1 | 左转90度 |
| 1 | 0 | 0 | 另行分析 |
| 1 | 0 | 1 | 直行 |
| 1 | 1 | 0 | 右转90度 |
| 1 | 1 | 1 | 右转180度\* |

\*注：此时右转180度是先右转90度后变为110状态，之后再右转90度。

现在，我们对于只有一个传感器感应到障碍物的状态（包括001、010、100）进行分析。我们的算法是向右优先算法，优先级从大到小为右转、直行、左转。在只有右传感器感应到障碍物时（001状态），由于客观上不应该向右转，所以仍然直行。同理，在前面有障碍物时（010状态）不能直行，所以右转；在左面有障碍物时（100状态）也优先选择右转。

于是，补全后的传感器真值表如表2.5.3所示。

**表2.5.3 补全后的传感器的真值表**

|  |  |  |  |
| --- | --- | --- | --- |
| L  （左） | M  （中） | R  （右） | F  （输出） |
| 0 | 0 | 0 | 直行 |
| 0 | 0 | 1 | 直行 |
| 0 | 1 | 0 | 右转90度 |
| 0 | 1 | 1 | 左转90度 |
| 1 | 0 | 0 | 右转90度 |
| 1 | 0 | 1 | 直行 |
| 1 | 1 | 0 | 右转90度 |
| 1 | 1 | 1 | 右转180度 |

### 算法的改进与优化

根据真值表，我们编写了相关函数控制小车的各项功能。我们采用分别控制左右轮转动的方法来控制小车的运动。比如youzhuan函数，在很短的周期内，先让左轮前进（youjin函数），再让右轮后退（zuotui函数）。通过控制youzhuan函数循环执行的时间（或次数，用for循环实现），来控制小车右转的角度。其他函数类似。

但是在实际测试的过程中，当小车遇到应该向右转弯的情况时（110状态），便会右转90度进入下一状态（100状态），此时，若按照向右优先算法的思想，应该继续右转90度，但实际情况是应该直行。此时，就需要我们进行改进。我们采取的解决方案是：在小车右转之后，加入代码让小车继续向前走一段路程，以离开100状态进入下一个正常的状态。

## 外观设计

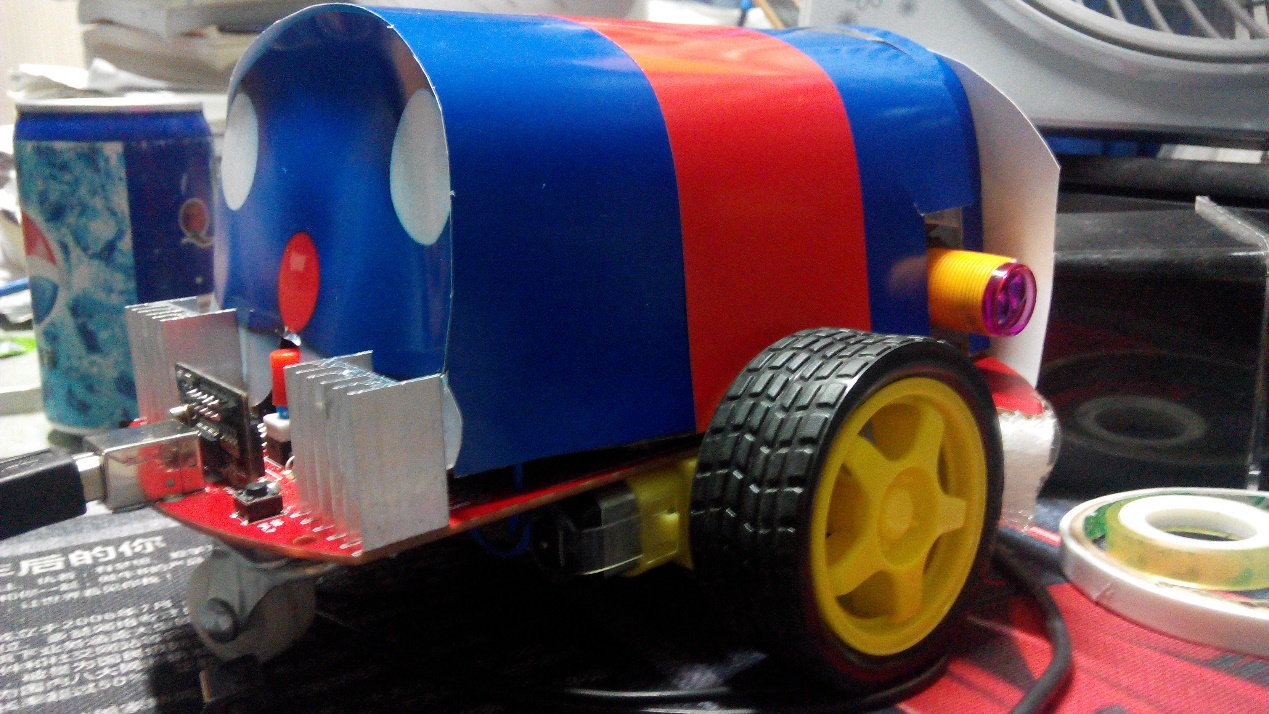
在将小车内部线路固定好之后，我开始进行外观设计。我预想的是将车身设计成一个趴着的哆啦A梦的造型。首先，我用一个透明的塑料饭盒倒扣住车身，在三个传感器伸出来的地方剪三个孔，并将散热片和USB口留在塑料饭盒的外面，以便于烧录程序和散热。并且也在塑料饭盒上剪了一个小口以便于按下按钮。

之后，我在网上找到了哆啦A梦正面和背面的素材，裁剪好后用胶版纸彩印出来，并同时打印了哆啦A梦身体的蓝色和铃铛系带的红色作为车身的装饰。

最后，我用双面胶和透明胶布将裁剪好的素材粘贴到车身上。除此之外，我还在车身前端的下面粘贴了一块泡沫，以减少撞击对车身电路和元器件造成的损害。最后的外观如图2.6.1—图2.6.3所示。

****

**图2.6.1 车身正面**

****

**图2.6.2 车身侧面**

****

**图2.6.3 车身背面**

# 实现功能

小车以单片机、传感器等为硬件基础，带有直行、左转、右转、掉头等功能，通过向右优先算法及其改进算法，最终能走出实验室提供的迷宫。

# 人员分工

|  |  |  |
| --- | --- | --- |
| 姓名 | 学号 | 分工 |
| 曹爽 | 2013210640 | 小车焊接、装配、外观制作 |
| 王建勇 | 2013210639 | 代码编写与调试、焊接监督 |

注：其他未尽事项为共同完成。

# 问题总结和心得体会

这次的电子工艺实习是我大学期间的第一门小学期动手实践课，而且这门课的任务是做出一个成品——能走迷宫的智能小车，最后还会有比赛等形式，这让我对这门课很感兴趣。

课程开始时，老师先简单讲解焊接的基本知识，之后我们亲自动手焊接各种元器件。在练习焊接的过程中，由于操作不熟练，我遇到了很多问题。有时掌握不好焊锡丝的用量，导致焊点锡量过大；有时用烙铁熔化的时间过长，导致焊盘剥离；还经常会有不小心将一个管脚与隔壁的管脚不小心连焊造成短路的情况。不过，在长期的练习之后，我终于初步掌握了这门技术，焊出了令人满意的焊点。在随后的发光二极管交替闪烁电路的制作时，也是一次性地焊通电路使其正常工作，这在很大程度上增强了我的自信心。

接下来的智能小车仍然由我来焊接。这是一项考验技术和耐心的工作，需要焊接的点又多又小又密，有些元器件动辄十几个管脚。焊接过程并不顺利，有时仍会遇到焊盘剥离的情况，不得已之后焊成一个较大的焊点。我以一个相对较慢的速度进行焊接，边焊接边检查焊点是否符合要求、电路是否连通，最大程度避免因焊接导致的电路问题。

之后是组装，这个比焊接要简单一些，就是用螺丝和螺母固定各种装置，并不是什么大问题。组装之后开始检查电路是否短路，很幸运，两个检验都显示中间有几千欧姆的电阻，说明电路不短路。之后插上USB线连接电脑后，也能正确识别。于是我们继续进行接下来的代码调试工作。

有关代码的工作主要由另一位组员完成，详细的原理已在前面说明。在调试代码的过程中，我们发现两个轮子的旋转情况总是不太对称，检查后发现原来我们用的舵机一个是单轴舵机，一个是双轴舵机。在和其他组商量之后，我们全部换成了单轴舵机。但是相比于双轴舵机，单轴舵机的缺点是摩擦大、耗电快。而且，我们的传感器对电压要求较高，电池稍微电压不足，传感器便会出现闪烁的情况，导致小车无法正常工作。所以在测试过程中，我们经常要更换新电池，并且电池电量还没有用完时，传感器就会出现问题，不得已只好继续更换新电池。这不仅浪费金钱，而且浪费资源。后来，我为了解决电压不足的问题，又在原来的基础上串联了一个电池盒，用八节使用过但电量未耗尽的电池为小车供电。这样，电压终于够了，并且舵机也比以前跑的更快（以前我们小车的速度很慢）。但是这样就要求我们不能长时间连续让小车工作，并保证良好的散热，以防止因过热造成的电路损坏。

调试代码的过程非常繁琐，每次都是修改了几个参数，之后再拿到迷宫上测试效果。如此重复，直到效果满意为止。在测试的过程中，我们也发现了许多特殊情况，并根据此改进了我们的算法。另外，我们的USB口接触不是很好，导致烧录程序经常失败。我曾重新焊接过一次，不过仍然没有多大效果，估计应该是USB模块内部的接触不良。

值得一提的是，在最开始传感器闪烁的时候，我们并不知道是什么原因。为此，我们重新连接了传感器相关的电路，更换了杜邦线，加了绝缘黑胶带防止短路，甚至修改了代码，但仍然无济于事。可是每次只要插上USB线便能正常工作，使用电池便会出现问题。于是我们才注意到电池电压的问题。在更换了全新的电池之后，传感器果然灵敏了。

最后，还是由我来进行外观制作。我希望把车包装成一个萌萌的卡通形象，最后选择了哆啦A梦。由于我的绘画水平不高，所以我选择彩印后粘贴到小车上。最终做出的效果还算差强人意。

可是，最后验收的时候，由于电池电压对小车舵机转速的影响太大，导致小车转动的角度过大，所以最后没能走出迷宫。这固然令人遗憾，可是也说明我们做东西时还需考虑各种因素，不能只从理论上解决问题，还要在实际中行之有效才行。

为期两周的电子工艺实习结束了，我从中体会到了团队合作的重要性，体会到了做东西的喜悦，也品尝了效果不尽人意的苦涩。不管结果如何，这些都是宝贵的经历，它们能让我体会到电子制作的无穷乐趣。

# 参考文献

［1］《电子工艺实习教程》，北京邮电大学电子工程学院电路中心，2015.6

［2］《89C51单片机实用教程》，张晓乡，电子工业出版社，2010.8

［3］HEX文件格式详解，http://www.forwhat.cn/post-240.html

# 附：算法源程序

#include<reg51.h>

#include <intrins.h>

#define uchar unsigned char

#define uint unsigned int

sbit IN1 =P1^3;

sbit IN2=P1^5;

sbit IN3=P1^2;

sbit IN4=P1^0;

sbit YOUHW=P3^3;

sbit ZHONGHW=P3^2;

sbit ZUOHW=P3^1;

unsigned int count=0;

unsigned int TURN=2;

zuojin()

{

IN1=0;

IN2=0;

IN3=0;

IN4=1;

}

youjin()

{

IN3=0;

IN4=0;

IN1=1;

IN2=0;

}

zuotui()

{

IN1=0;

IN2=0;

IN3=1;

IN4=0;

}

youtui()

{

IN3=0;

IN4=0;

IN1=0;

IN2=1;

}

qianjin()

{

uint i=0;

uint j=0;

for(;i<130;i++)

{

zuojin();

}

for(;j<100;j++)

{

youjin();

}

}

zuozhuan()

{

uint i=0;

uint j=0;

for(;i<110;i++)

{

zuotui();

}

for(;j<100;j++)

{

youjin();

}

}

youzhuan()

{

uint i=0;

uint j=0;

for(;i<100;i++)

{

zuojin();

}

for(;j<100;j++)

{

youtui();

}

}

void main()

{ P0=0xff;

P2=0xff;

P1=0xff;

TMOD=0x01;

TH0=0xfe; TL0=0x0c;

EA=1; ET0=1; TR0=1;

TURN=2;

while(1)

{

if(ZUOHW==0 &&

ZHONGHW==0 && YOUHW==1)

{

count=0;

for(;count<66;count++)

youzhuan();

count=0;

for(;count<130;count++)

qianjin();

}

else if(ZUOHW==1 &&

ZHONGHW==0 && YOUHW==0)

{

count=0;

for(;count<65;count++)

zuozhuan();

count=0;

for(;count<30;count++)

qianjin();

}

else if(ZUOHW==0 &&

ZHONGHW==0 && YOUHW==0)

{

count=0;

for(;count<30;count++)

youzhuan();

}

else if(ZUOHW==0 &&

ZHONGHW==1 && YOUHW==0)

{

count=0;

for(;count<100;count++)

qianjin();

}

else if(ZUOHW==1 &&

ZHONGHW==1 && YOUHW==0)

{

qianjin();

}

else if( ZUOHW==0 &&

ZHONGHW==1 && YOUHW==1)

{

count=0;

for(;count<66;count++)

youzhuan();

count=0;

for(;count<130;count++)

qianjin();

}

else if( ZUOHW==1 &&

ZHONGHW==0 && YOUHW==1)

{

count=0;

for(;count<66;count++)

youzhuan();

count=0;

for(;count<130;count++)

qianjin();

}

else if(ZUOHW==1 &&

ZHONGHW==1 && YOUHW==1)

{

count=0;

for(;count<100;count++)

qianjin();

}

}

}