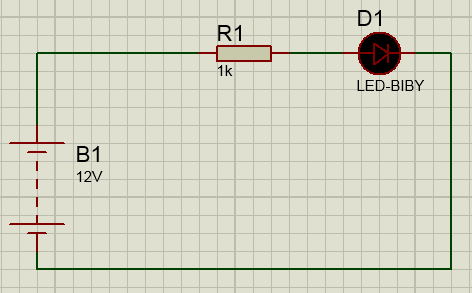
**电子线路设计实验报告**

自动化71任泽华 2171411498

第一次实验

# 例程1

## 电路图：

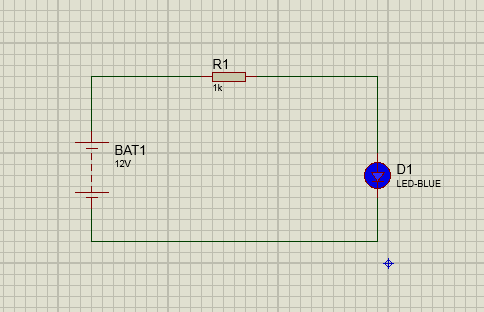


## 用到的器件：

|  |  |  |
| --- | --- | --- |
| 图中器件 | 中文名 | 英文名 |
| 1 | 固定电阻 | RES |
| 2 | 发光二极管 | LED（可选择颜色） |
| 3 | 直流电源 | BATTERY |

## 实验过程：

调用器件库中的元件，按照要求连接电路，点击运行，二极管被点亮（0.pdsprj）



## **学到的东西：**

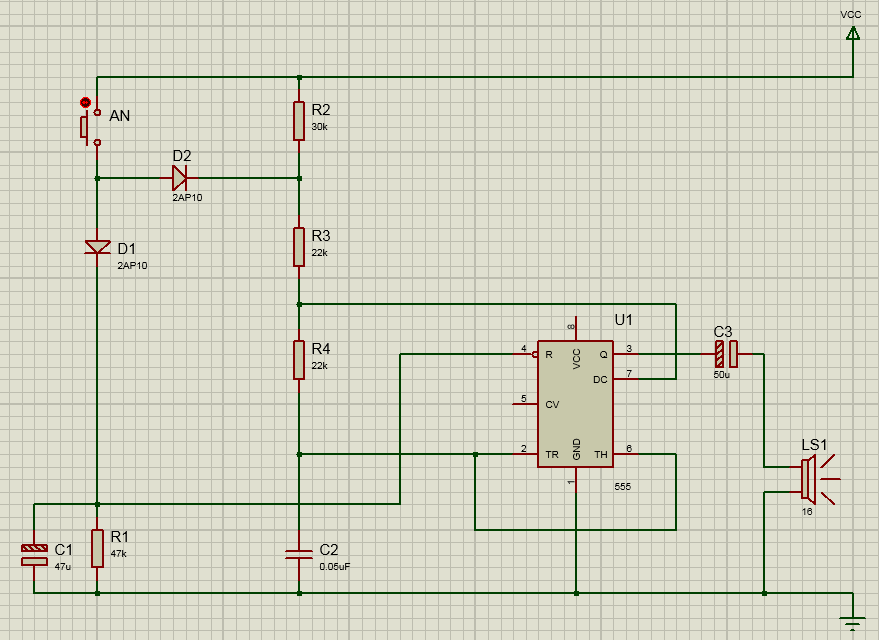
1.元器件库的位置，刚开始在黑底的编辑窗口找不到P标志，后来发现应该是在白底的窗口才能找到。

2.连线的方法，只要鼠标放到管脚上面就可以连接。

3.元器件的旋转方法，右键点击然后选择选择就好，经过同学提醒发现一个快捷键ctrl和+或者-，可以实现顺时针、逆时针的旋转。

# 例程2

## 电路图：



## 用到的器件：

|  |  |  |
| --- | --- | --- |
| 图中器件 | 中文名 | 英文名 |
| 1 | 固定电阻 | RES |
| 2 | 二极管 | DIODE |
| 3 | 电容 | 定值无极性电容（CAP）有极性电容（CAPACITOR POL） |
| 4 | 集成定时器 | 555 |
| 5 | 喇叭（蜂鸣器） | SPEAKER（BUZZER） |
| 6 | 按钮 | BUTTON |
| 7 | 电源 | VCC GND |

## 实验过程

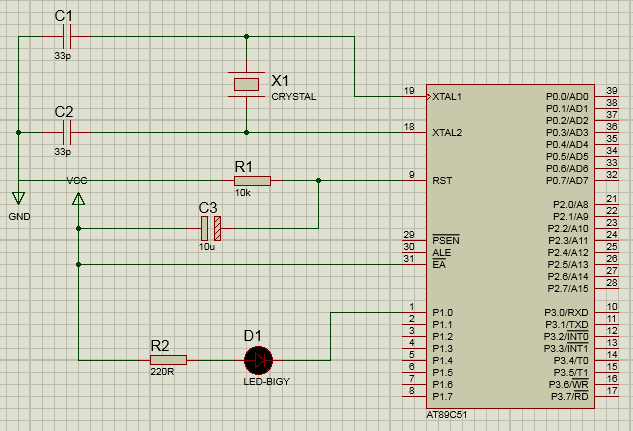
输出声音见文件夹下的 蜂鸣.mp4（1.pdsprj）蜂鸣器发出了蜂鸣声。

## 学到的东西

学会了VCC和 GND的放置方法，在左边侧栏选择terminals即可，需要注意的是，添加电源和电源地之后要再次双击修改参数，比如说是VCC还是VPP。、

# 例程3

## 电路图：



## 用到的器件：

|  |  |  |
| --- | --- | --- |
| 图中器件 | 中文名 | 英文名 |
| 1 | 固定电阻 | RES |
| 2 | 发光二极管 | LED（可选择颜色） |
| 3 | 电容 | 定值无极性电容（CAP）有极性电容（CAPACITOR POL） |
| 4 | 晶振 | CRYSTAL |
| 5 | 单片机 | AT89C51（时钟频率：11.0592） |
| 6 | 电源 | POWER（VCC GND） |

## 程序：

#include<reg51.h>

void main(void)

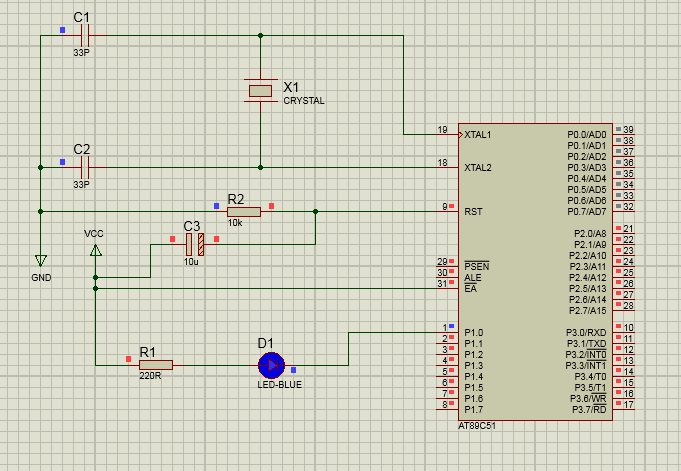
{

P1=0xfe;

}

## 实验过程

点击运行，发光二极管被点亮。（2.pdsprj）



## 学到的东西：

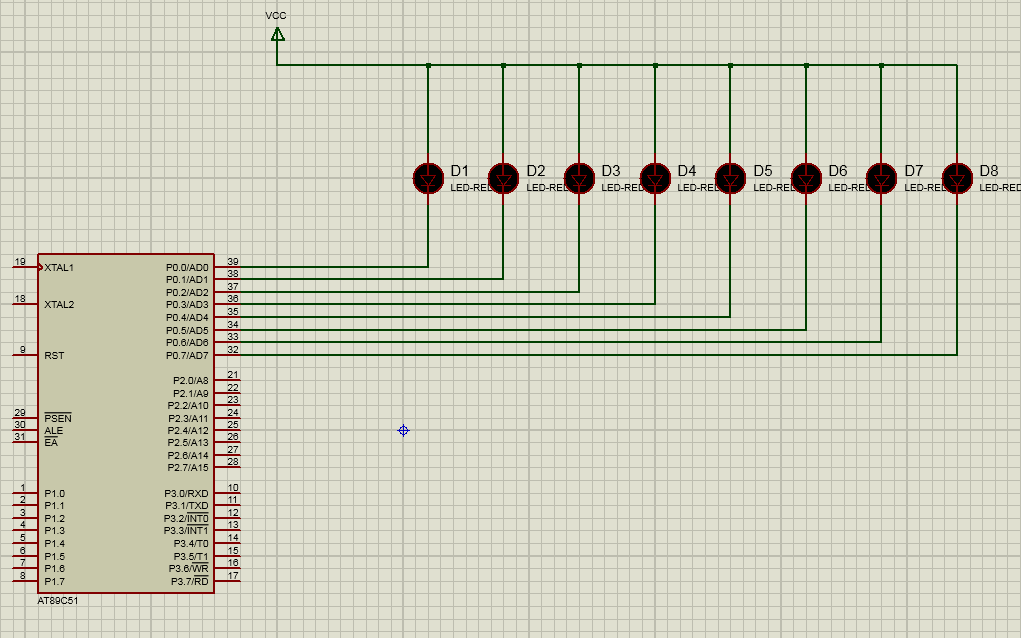
1.学到了如何在建立工程的时候添加单片机部件，在刚开始选错单片机后后续操作会比较麻烦。

2.学会了怎样在keil当中编辑程序并且把它导入到Proteus当中，当时建立了.c文件以后忘记了把它添加到工程里面，直接运行以后发现什么也没有，要注意敲完代码以后一定要记得添加到工程里面。

3．要记得选择勾选“创建二进制文件”的按钮

# 例程4

## 电路图：



## 用到的器件：

|  |  |  |
| --- | --- | --- |
| 图中器件 | 中文名 | 英文名 |
| 1 | 发光二极管 | LED（可选择颜色） |
| 2 | 单片机 | AT89C51（时钟频率：11.0592） |
| 3 | 电源 | POWER（VCC GND） |

## 实验过程：

实验程序老师在说明文件当中给出，此处不予列出。工程文件3.pdsprj

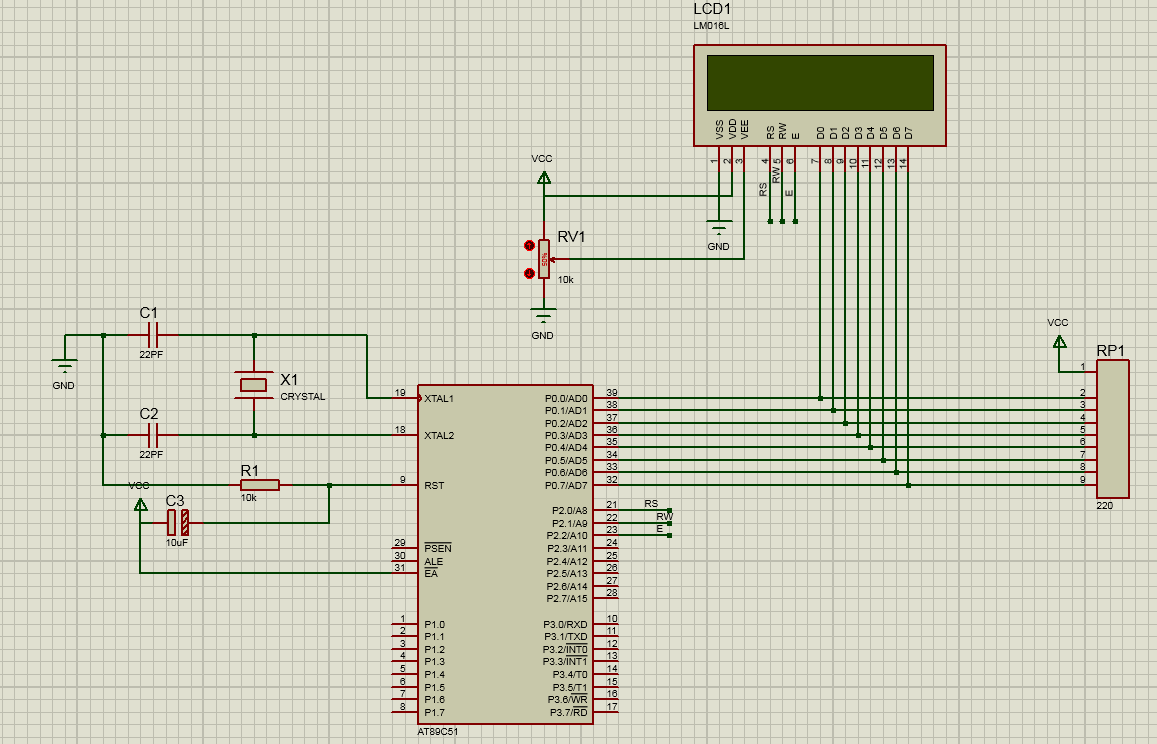
点击运行，发光二极管依次被点亮，形成“跑马灯”的效果。见 跑马灯.mp4

## 学到的东西：

学到了复杂程序的调试方法，粘贴代码时出了一点小问题，通过下方的编译提示找出了出问题的位置并进行了改正。

# 例程5

## 电路图：



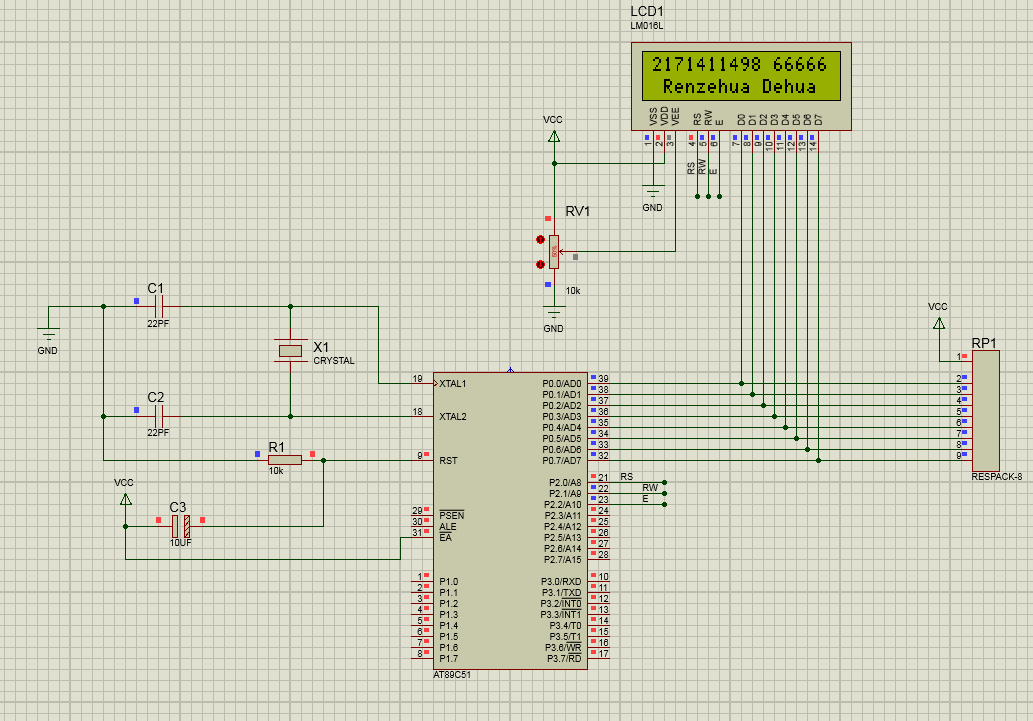
## 用到的器件：

|  |  |  |
| --- | --- | --- |
| 图中器件 | 中文名 | 英文名 |
| 1 | 固定电阻 | RES |
| 2 | 发光二极管 | LED（可选择颜色） |
| 3 | 电容 | 定值无极性电容（CAP）有极性电容（CAPACITOR POL） |
| 4 | 晶振 | CRYSTAL |
| 5 | 滑线变电阻 | POT-HG |
| 6 | 电阻排 | RES中选RESPACR |
| 7 | 液晶 | LM016L |
|  | 单片机 | AT89C51（时钟频率：11.0592） |
|  | 电源 | POWER（VCC GND） |

## 实验过程

实验程序老师在说明文件当中给出，此处不予列出。工程文件4.pdsprj

点击运行，液晶屏显示出想要显示的信息，此处为姓名学号。



## 学到的东西：

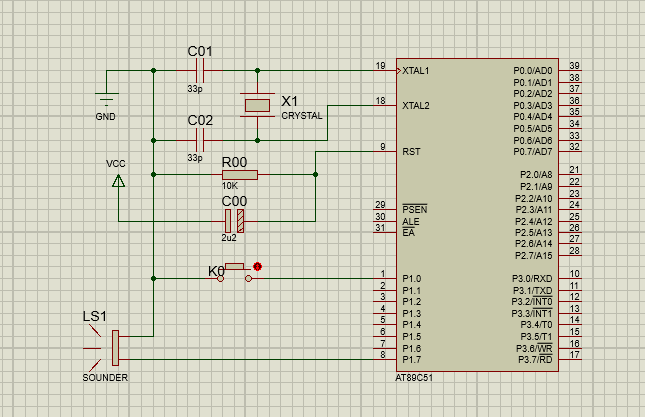
1.在此例中，我学到了跨接线的接法，在左边工具栏点击wire label model，可以对指定线路进行编号，相同编号认为相连接。

2.这是同学碰到的问题，程序编译正确，电路图也找不到错误，经过搜索发现是在电路图上面应该相连的位置出现了节点，原因是连过了头。所以在连接线路的时候一定要注意不要产生这样的“虚接点”

**注：上述例程工程文件在lib1文件夹内，2、3、4.pdsprj相关程序文件在里面的AT89C51文件夹下，名字相同；下方补充练习工程文件为在mylib文件夹内的song.pdsprj，它的程序在该路径下AT89C51文件夹下，happybirthday.hex和jiaoaodeshaonian.hex**

# 补充练习

## 电路图：



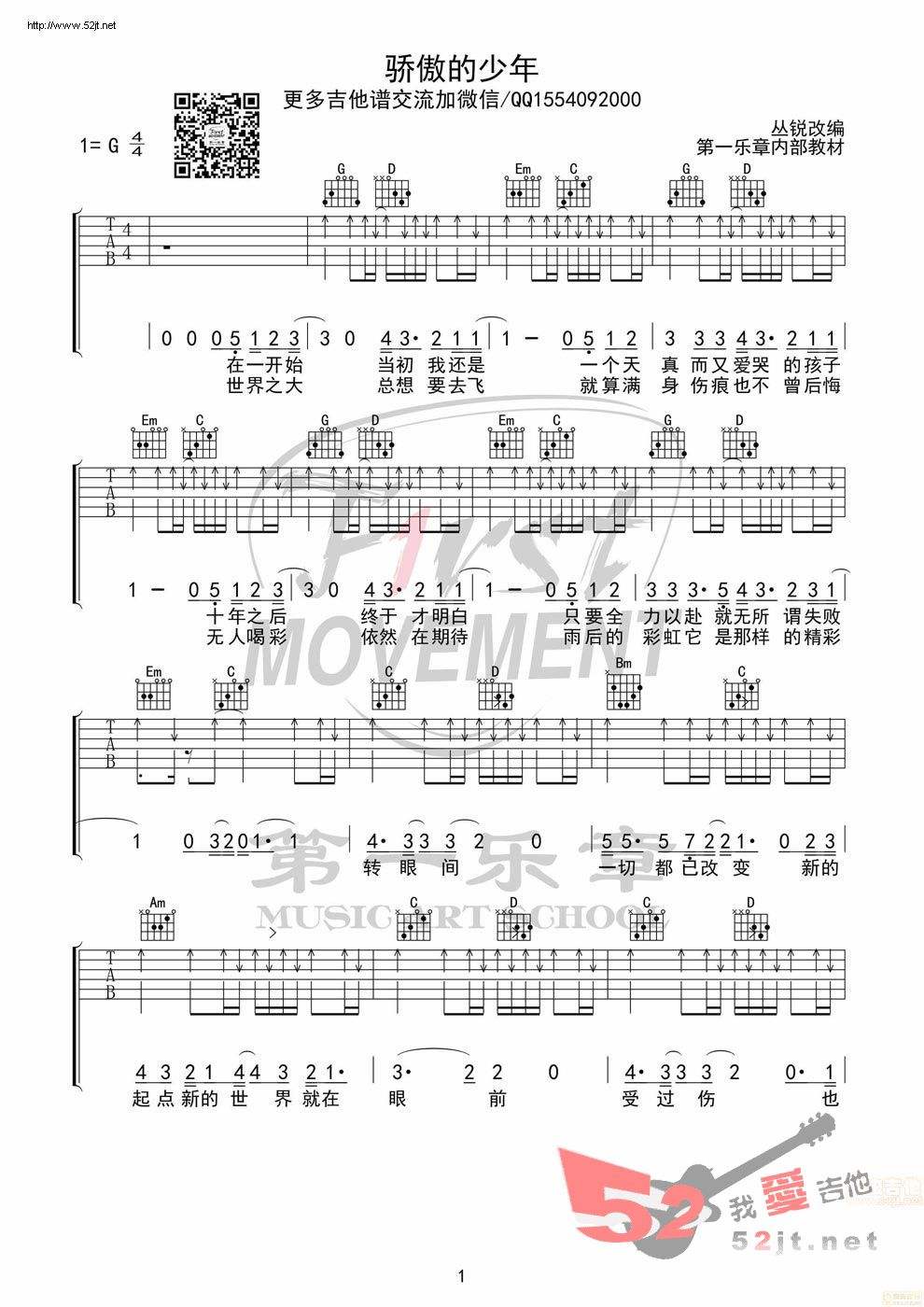
## 用到的器件：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 名称 | 编号 | 大类 | 子类 | 型号/标称值 | 数量 |
| 80C51 | U1 | Microprocessor\_Ics | 80C51\_family | AT89C51 | 1 |
| 石英晶体 | X1 | Miscellaneous | CRYSTAL | 12MHz | 1 |
| 电阻 | R00 | Resistors | Chip\_Resistor\_1/8W\_5% | 10kΩ | 1 |
| 电容 | C00 | Capacitors | Miniature\_Electronlytic | 2μ2 | 1 |
| . | C01 C02 | Capacitors | Ceramic\_Disc | 33P | 2 |
| 按键 | K0 | Switches&Relays | Switches | BUTTON | 1 |
| 发声器 | LS1 | Speakers&Sounders | . | SOUNDER | 1 |

## 实验过程：

按照电路图连接，在keil里面敲入程序，编译导入Proteus，按下运行键。此时按下按键，音乐开始播放，播放完后再按下按键，音乐重复播放。改变程序里面的与音符、节拍有关的数组，可以播放不同的歌曲。(song.pdsprj)改变定时器参数，可以改变播放顺序。刚开始播放的是《生日快乐》歌，一拍800ms，然后用更快的速度播放《骄傲的少年》，一拍320ms。见生日快乐+骄傲的少年.mp4、骄傲的少年.mp4。不同的程序文件在happybirthday.hex和jiaoaodeshaonian.hex中。

《骄傲的少年》曲谱如下：



此处只选取了前两行内容。

程序如下：

#include <reg51.h>

sbit K0=P1^0;

sbit SOND=P1^7;

unsigned char i,j;

unsigned char code th[22]={

0,196,202,208,211,216,220,224,226,229,232,233,236,238,240,241,242,244,244,246,247,248};

unsigned char code tl[22]={

0,12,27,21,7,4,16,12,4,12,9,20,2,8,6,2,22,5,26,1,4,3};

unsigned char code s[26]={

12,12,13,12,15,14,12,12,13,12,16,15,12,12,19,17,15,14,13,0,18,18,17,15,16,15};

unsigned char code L[26]={

4,4,8,8,8,16,4,4,8,8,8,16,4,4,8,8,8,8,16,8,4,4,8,8,8,16};

void main(){

unsigned int t;

TMOD =0x10;

TH1 =0x3c;TL1=0xb0;

IP =0x02;

IE =0x8a;

while(1){

while(K0==1);

while(K0==0);

for(i=0;i<26;i++){

if(s[i]==0) {SOND=0;

TR0=0;}

else {TH0=th[s[i]];

TL0=tl[s[i]];

TR0=1;}

j=L[i];TR1=1;

while(TR1==1);

TR0=0;

SOND=0;

for(t=0;t<2000;t++);}}}

void t0() interrupt 1{

SOND=~SOND;

TH0=th[s[i]];TL0=tl[s[i]];}

void t1() interrupt 3{

TH1=0x3c;TL1=0xb0;

if((j--)==0) TR1=0;}

## 学到的东西：

1.进一步巩固了对于Proteus和keil软件的应用，对于电路图的设计、程序的编译运行调试有了进一步的认识，也提高了自己的上手操作能力。

2.学习了定时器的相关知识，TH1、TL1分别对应着高八位和低八位，定时/计数器初值计算公式：

又得知在此例中，是按方式1工作的，N=16，，1机周=1μs，按照此公式可以设计快放（一拍320ms）时的参数，即TH1 =0xc5;TL1=0x68;

3.对于乐理知识有了一定的了解