



西安邮电大学  
XI'AN UNIVERSITY OF POSTS & TELECOMMUNICATIONS

# 《智能小车设计实验》课程 实验指导书

## 目 录

序 言 .....	1
实验一 车模组装及电源稳压模块实现.....	2
一、实验目的.....	2
二、实验器材.....	2
三、实验内容及步骤.....	2
四、测试.....	3
实验二 单片机最小系统模块设计与实现 .....	4
一、实验目的.....	4
二、实验器材.....	4
三、实验内容及步骤.....	4
四、参考代码.....	5
附 1: keil 建立工程及下载程序教程 .....	5
附 2: 下载文件.....	13
附 3: proteus7.5 简易教程 .....	17
实验三 驱动模块的实现 .....	25
一、实验目的.....	25
二、主要实验器材.....	25
三、实验内容及步骤.....	25
四、参考例程.....	26
实验四 人机接口模块实现 .....	30
一、实验目的.....	30
二、主要实验器材.....	30
三、实验内容及步骤.....	30
四、参考例程.....	30
实验五 循迹模块设计与实现.....	32
一、实验目的.....	32
二、主要实验器材.....	32
三、实验原理图.....	32
四、实验步骤.....	32
实验六 测速模块实现.....	34
一、实验目的.....	34
二、主要实验器材.....	34
三、实验原理.....	34
四、实验步骤.....	35
实验七 综合实验---循迹模块和驱动模块的综合 .....	36
一、实验目的.....	36
二、主要实验器材.....	36
三、实验内容及步骤.....	36

实验八 综合实验---循迹模块、驱动模块、测速模块及显示模块的综合 .....	37
一、实验目的.....	37
二、主要实验器材.....	37
三、实验内容及步骤.....	37

## 序 言

本课程实验要求学生设计并实现一个具有循迹、测速、时间或速度显示等功能的智能小车。该小车具体功能架构如下图所示。

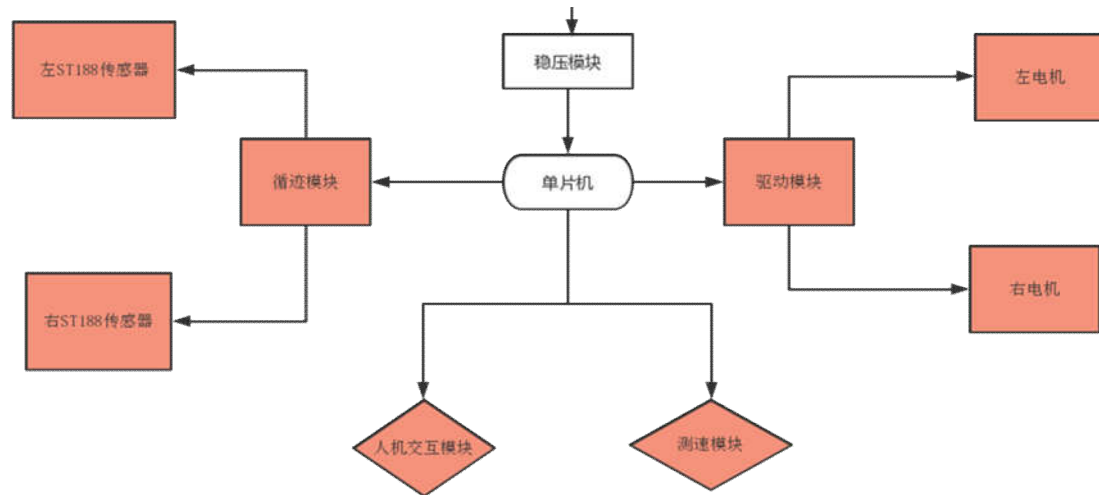


图 智能小车整体架构

## 实验一 车模组装及电源稳压模块实现

### 一、实验目的

- 1、熟练掌握车模组装的一般方法；
- 2、熟练掌握电源稳压电路的一般原理和实现方法。（51 单片机的工作电压为+5V，过高会损坏单片机。提供电池的电压为 7.5V 左右。）

### 二、实验器材

- 1、车模部件一套，7.4V 电池，充电器；
- 2、LM2940 芯片，47uf, 22uf 电解电容各一个，自锁开关，LED 发光二极管，470 欧电阻，1N4007 二极管，实验板，单排针。

### 三、实验内容及步骤

- 1、按照车模组装说明自行组装小车车模；
- 2、按照图 1.1 在实验板上焊接电路。在图 1.1 中，IN 代表电池的输出(DC7.4V)，OUT 代表稳压电路的输出（DC5V）。

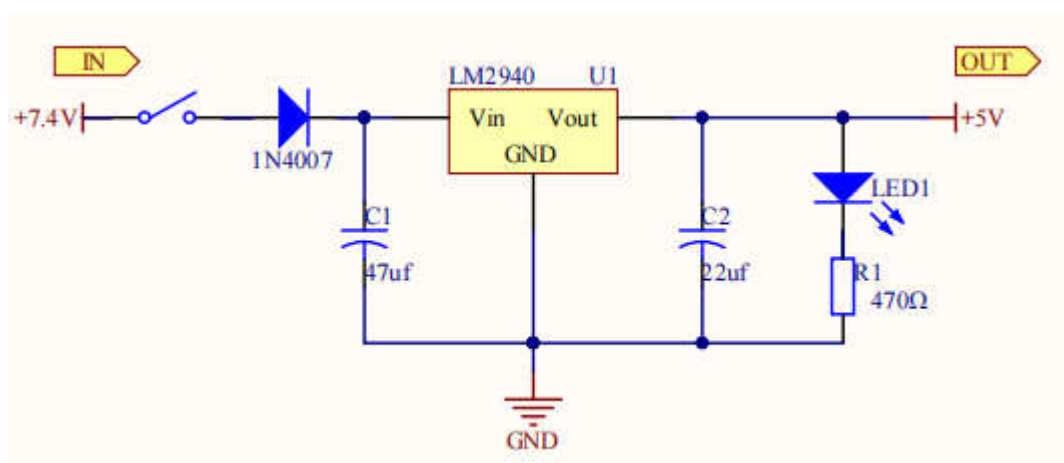


图 1.1 电源稳压电路原理图

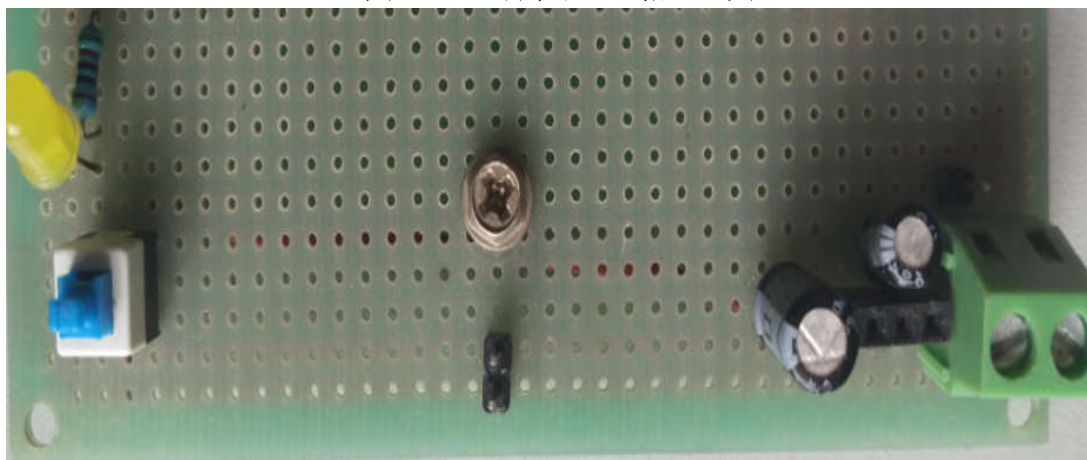


图 1.2 稳压电路实物图（正面）

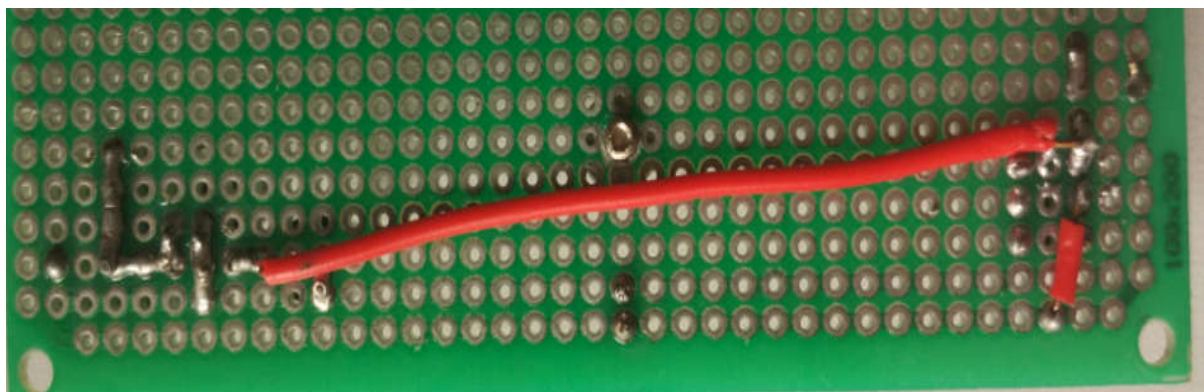


图 1.3 稳压电路实物图（背面）

#### 四、测试

##### 1、车模测试：

- 1)、机械结构是否完整、可靠；
- 2)、电机是否可以正常的正反转。

##### 2、稳压电路的测试：

使用万用表测量 out 口，若输出 5v（可有小范围偏差）则电路工作正常。

注：若发现芯片发烫严重应立刻断电，检查所焊电路。

## 实验二 单片机最小系统模块设计与实现

### 一、实验目的

- 1、掌握 51 单片机最小系统的组成；
- 2、初步掌握单片机 C 语言开发的一般过程（Keil 软件的使用）；
- 3、初步掌握电路仿真软件的使用方法（protues 软件的使用）；
- 4、掌握单片机程序下载的一般方法。

### 二、实验器材

STC89C52 单片机； 10k 电阻两个； 470 欧电阻； 30pf 瓷片电容； 10uf 电解电容； 12M 晶振； 按钮开关； LED 发光二极管

### 三、实验内容及步骤

- 1、按照图 2.1 在 Protues 仿真软件中绘制电路图；

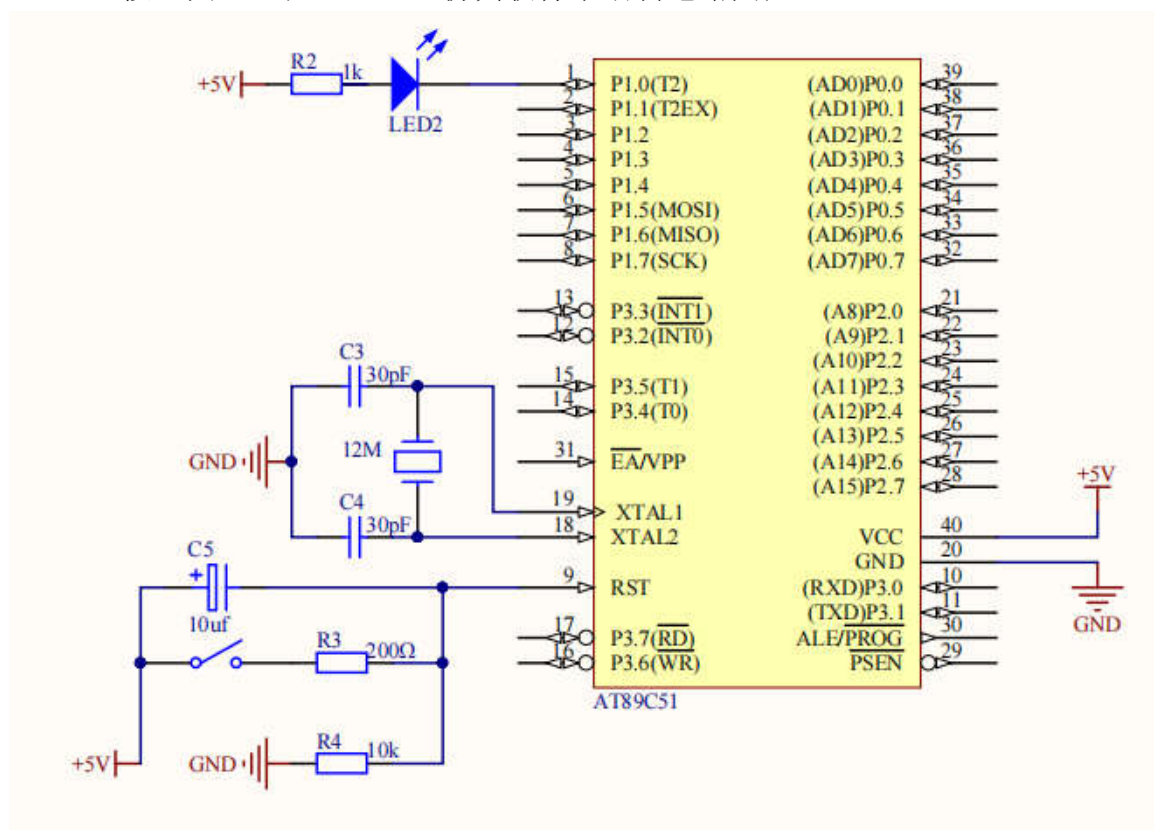


图 2.1 单片机最小系统电路图

- 2、参考流程图 2.2 在 Keil 环境下编写测试程序，编译生成目标文件（以 hex 为后缀名的文件），并将目标文件装载至 Protues 环境下的单片机中，仿真运行查看实验结果；



图 2.2 测试程序流程图

- 3、仿真运行无误后，按照图 2.1 焊接电路；
- 4、将目标文件下载至单片机（具体下载方法参见附录 1）；
- 5、系统上电，查看实验效果。（LED 应以一定频率闪烁）。

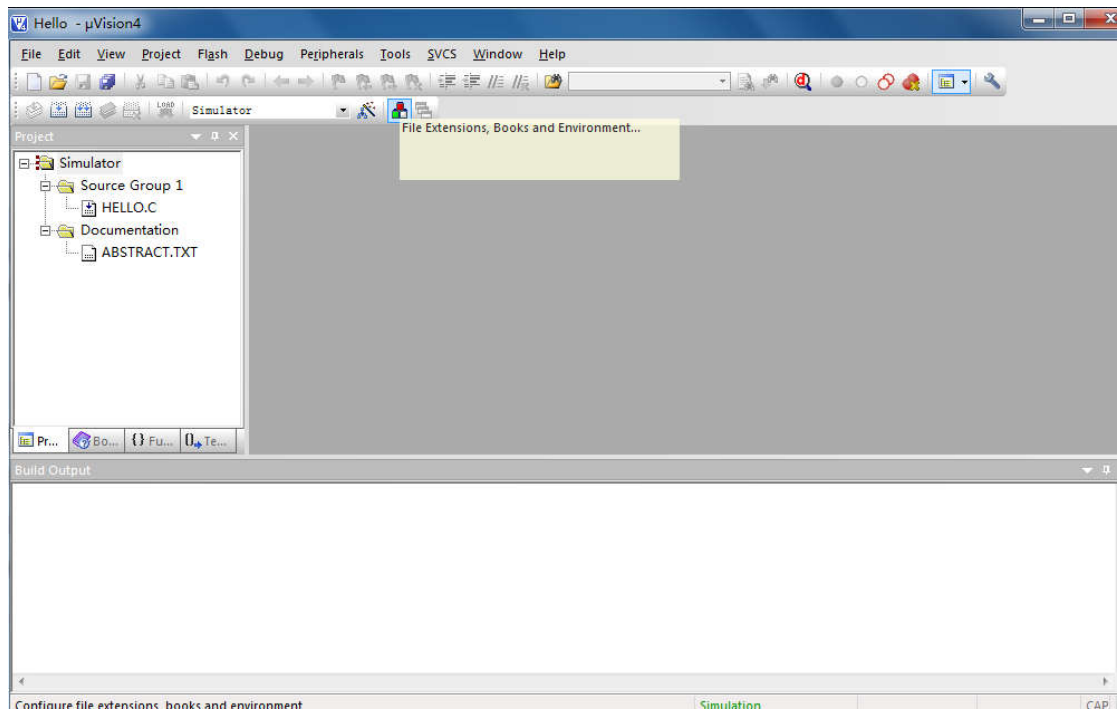
#### 四、参考代码

```
#include<reg52.h>
sbit P1^1 = Led;
void main()
{
    int i;
    while(1)
    {
        Led = 1;
        for( i = 0 ; i < 1000 ; i++)
        ;
        led = 0;
        for( i = 0 ; i < 1000 ; i++)
        ;
    }
}
```

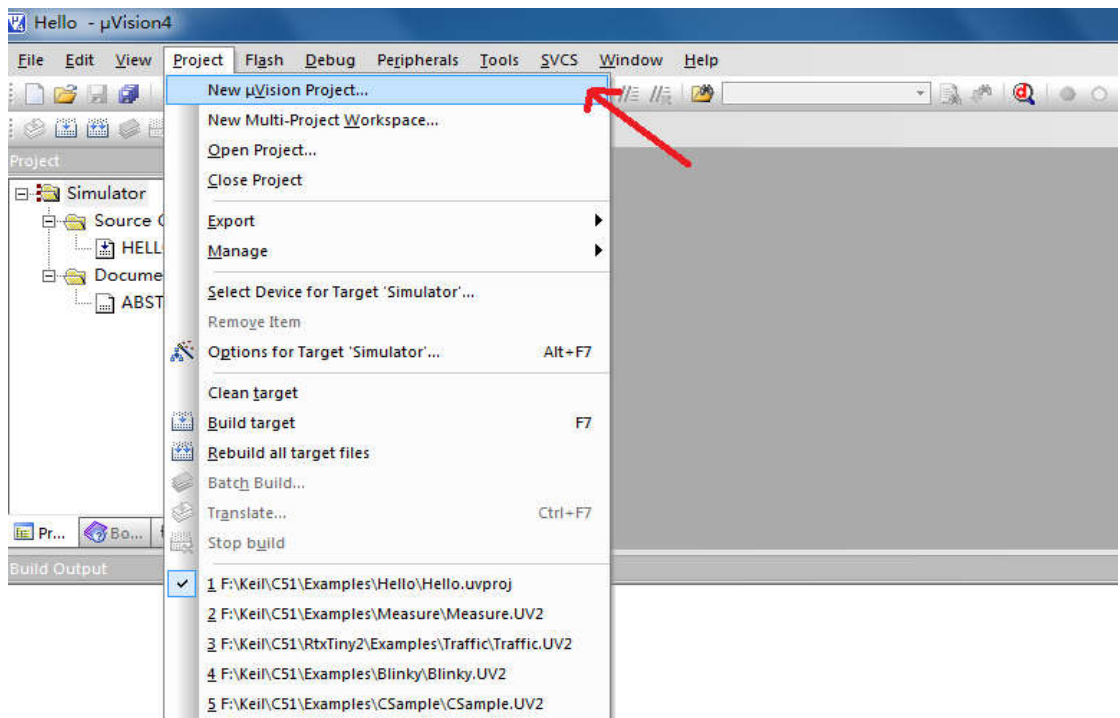
#### 附录 1: keil 建立工程及下载程序教程

- 1、打开 keil 软件

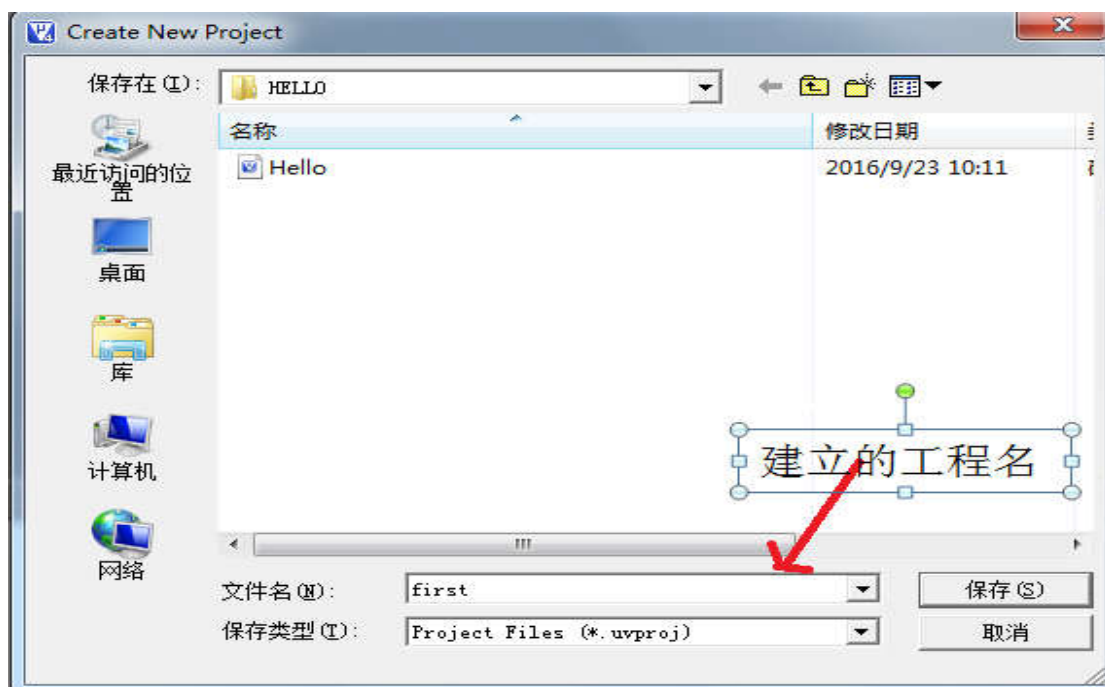




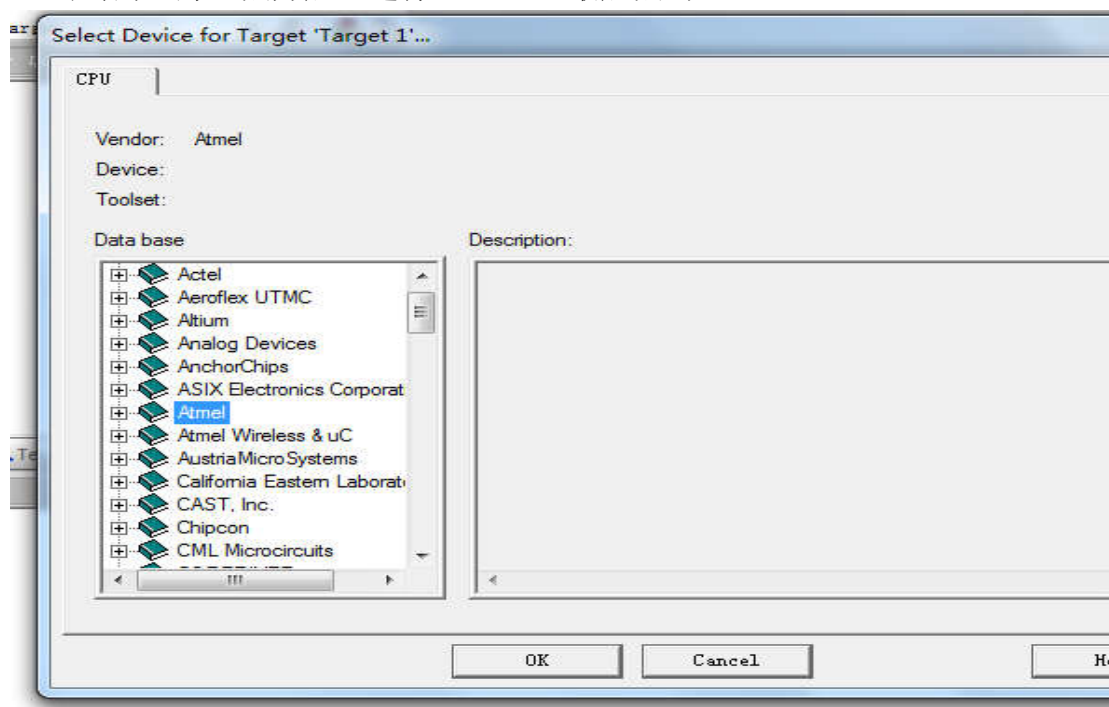
2 打开 keil 软件，点击菜单栏上的 Project，选择 New uVision Project。

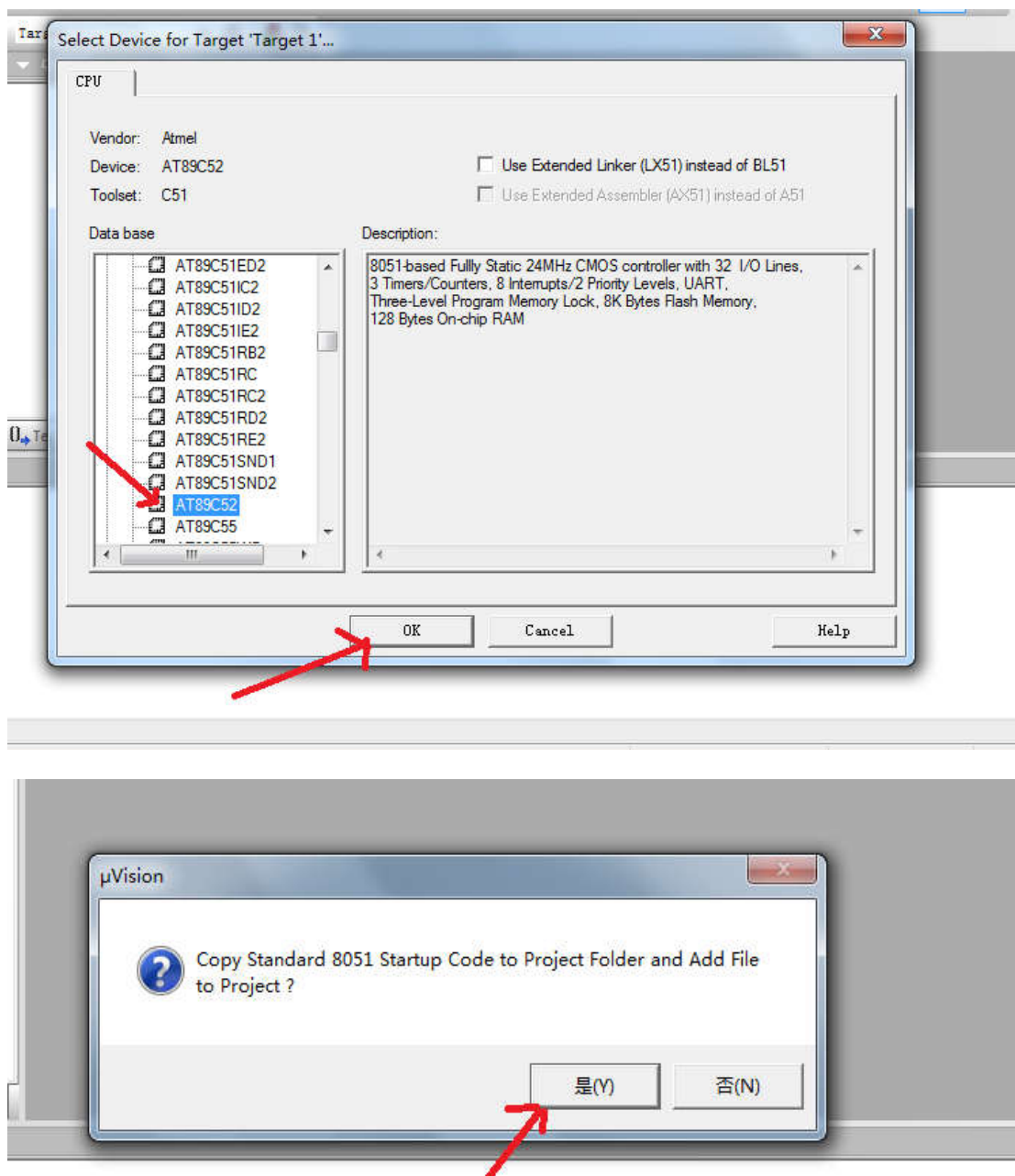


3 选择工程保存的路径与位置。给工程命名，然后保存

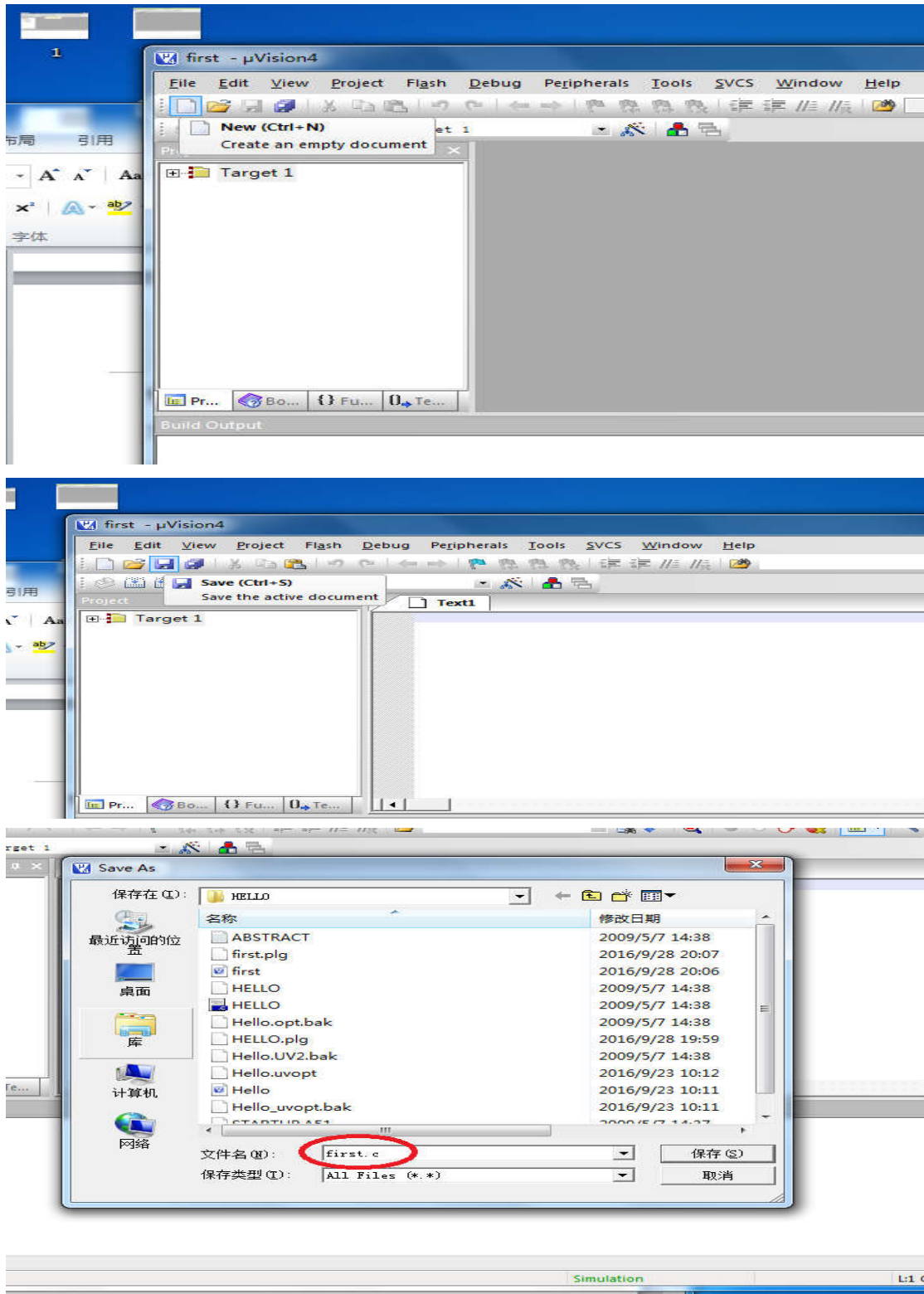


4 接着弹出选择芯片类型，一般是写 51 单片机系列的程序的话，就选择 Atmel 公司的芯片，点开后，选择 AT89C52. 最后点击 OK

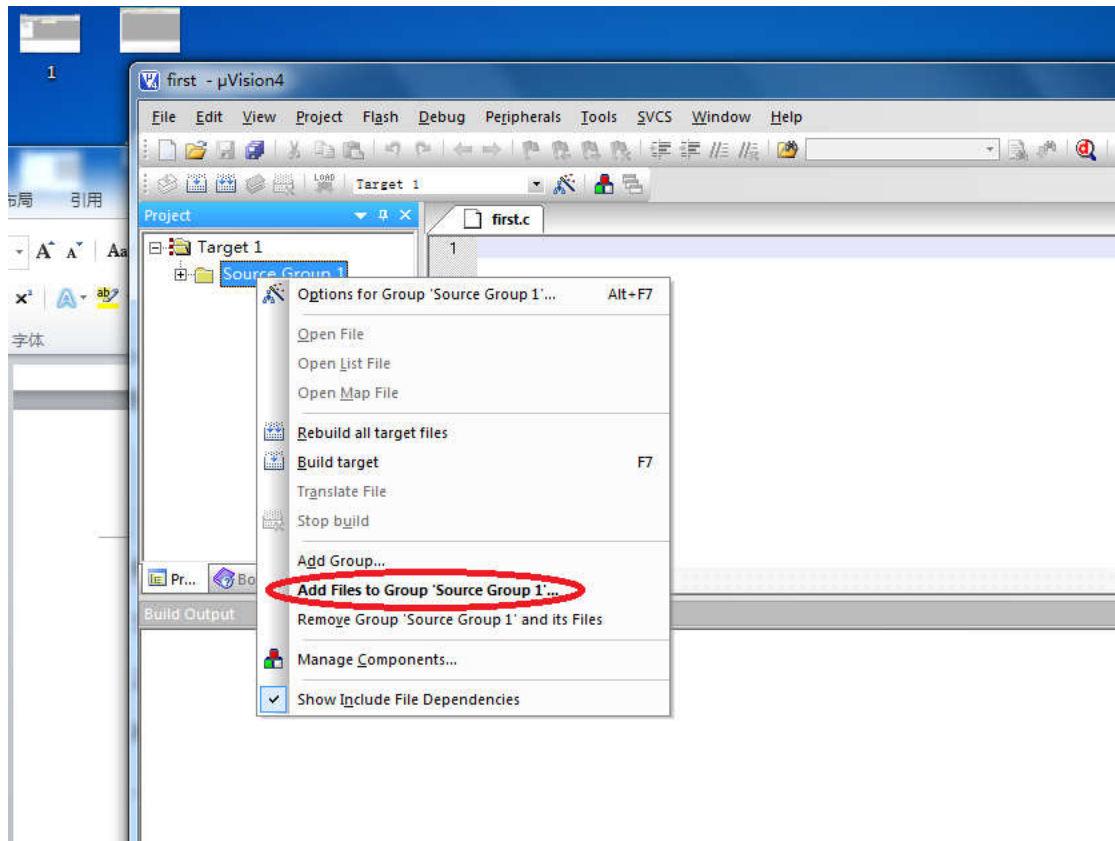




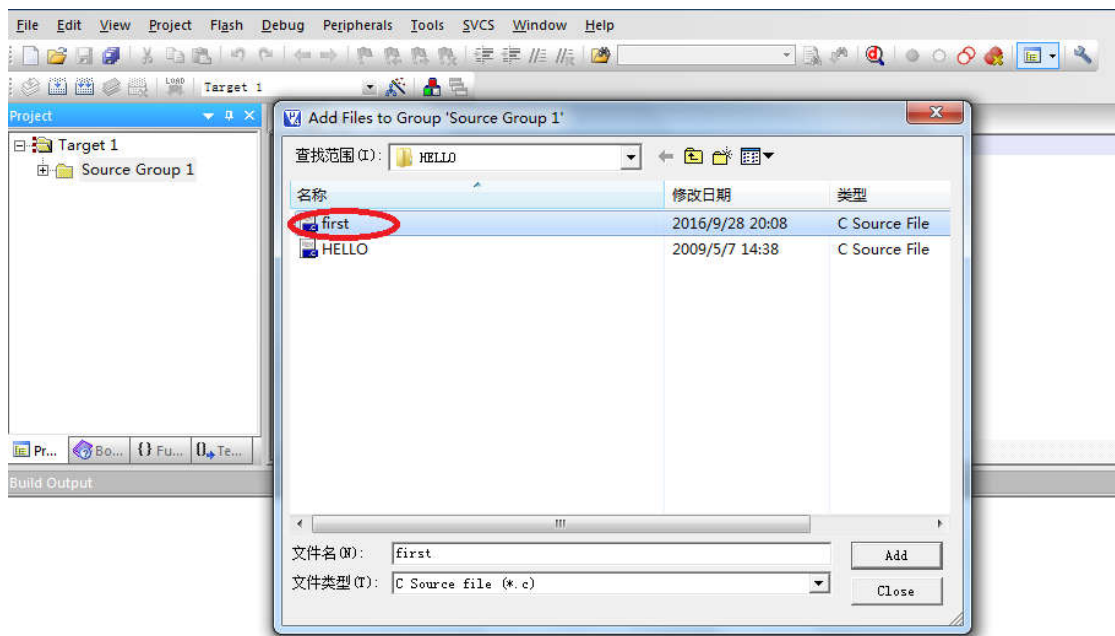
5 选择菜单 File，接着选择 New。新建一个文件。然后保存文件，保存在工程的路径下，保存的格式为 “.c”。



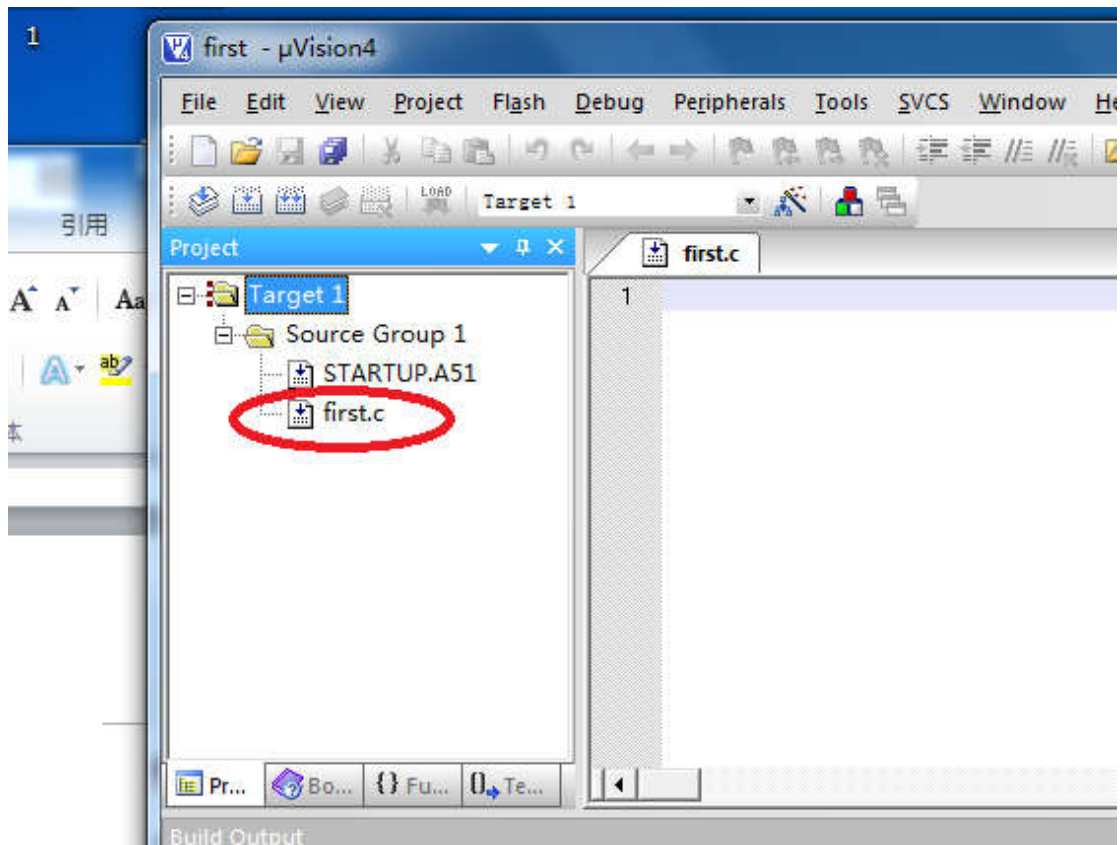
6 点开 Project 下的 Target 1 左边的加号。鼠标右击 Source Group1, 选择 Add File to Group 'Source Group1'...



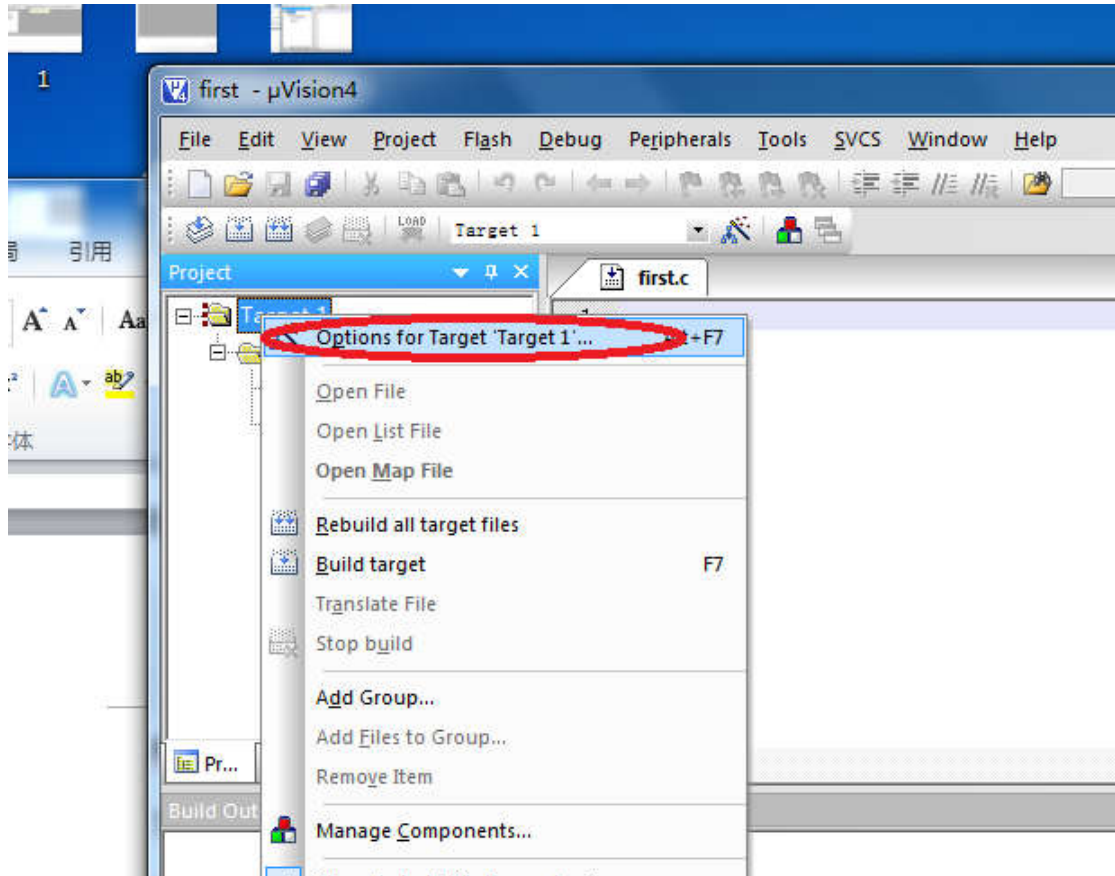
7 按截图的操作，找到.c 的文件，添加到工程里面。最后 close , 就可以了。



8 这个时候你看到 Source Group1 左边有加号标志。证明已经添加成功了。就可以编写程序了。

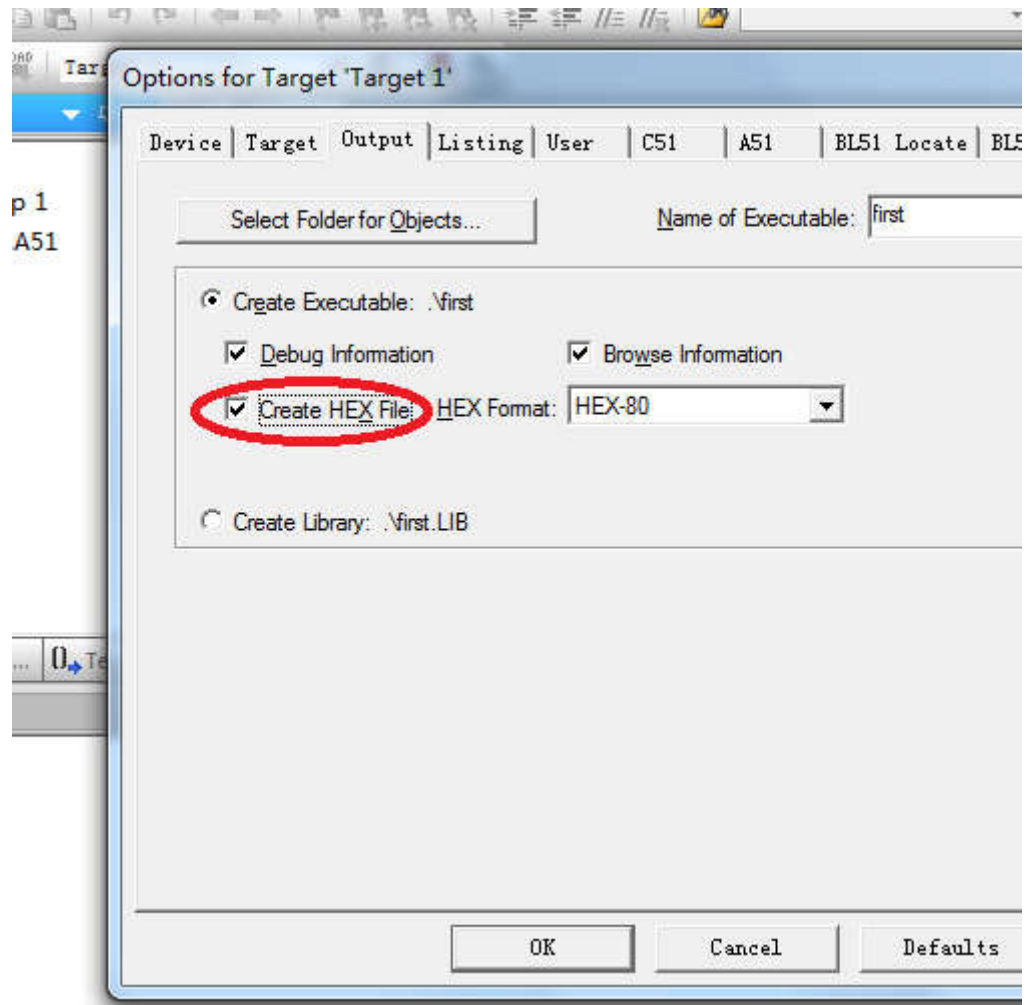


9 右击 project 下的 Target ， 选择 第一个

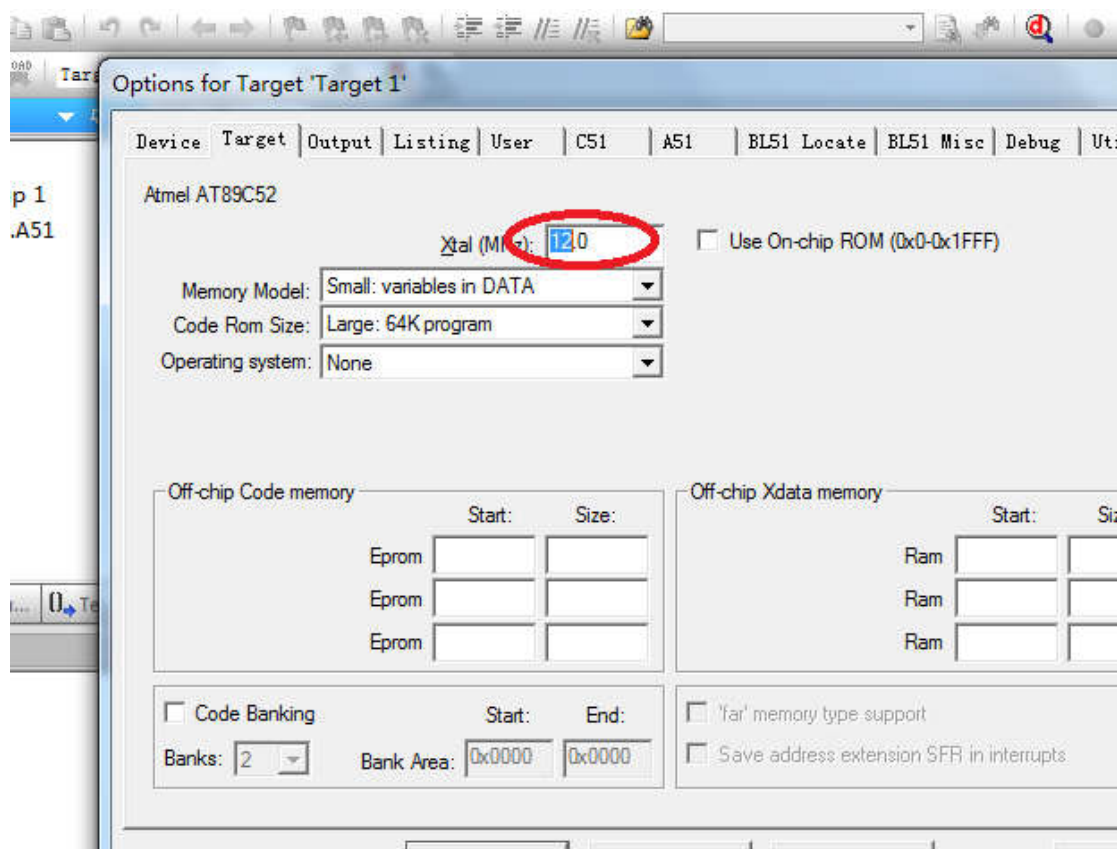




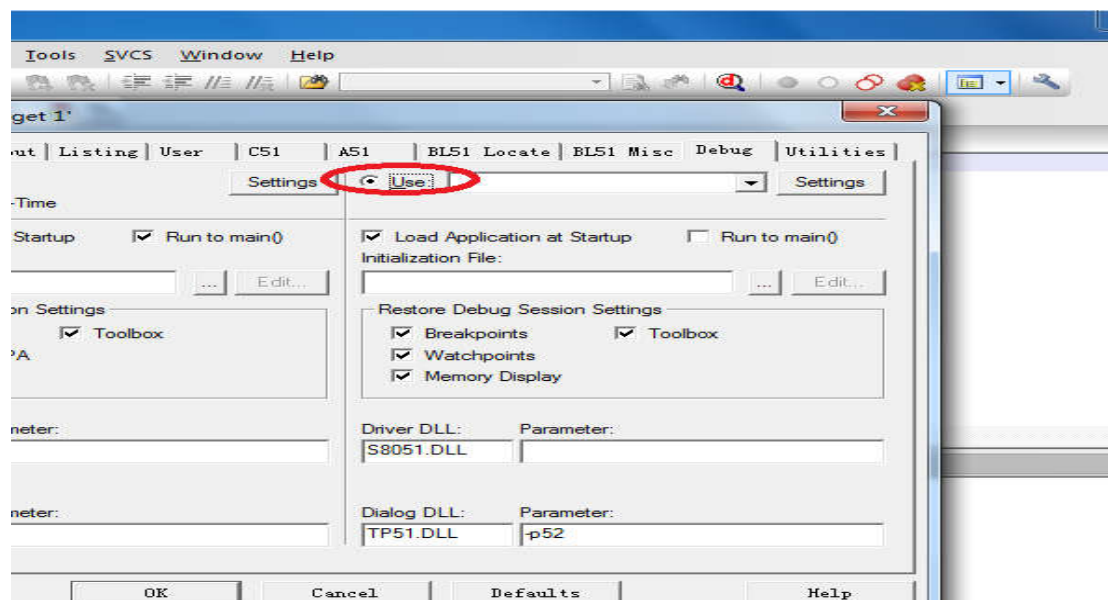
10 在 output 里面勾选 Create HEX File



11 在 Target 选项中将 Xtal 改成 12



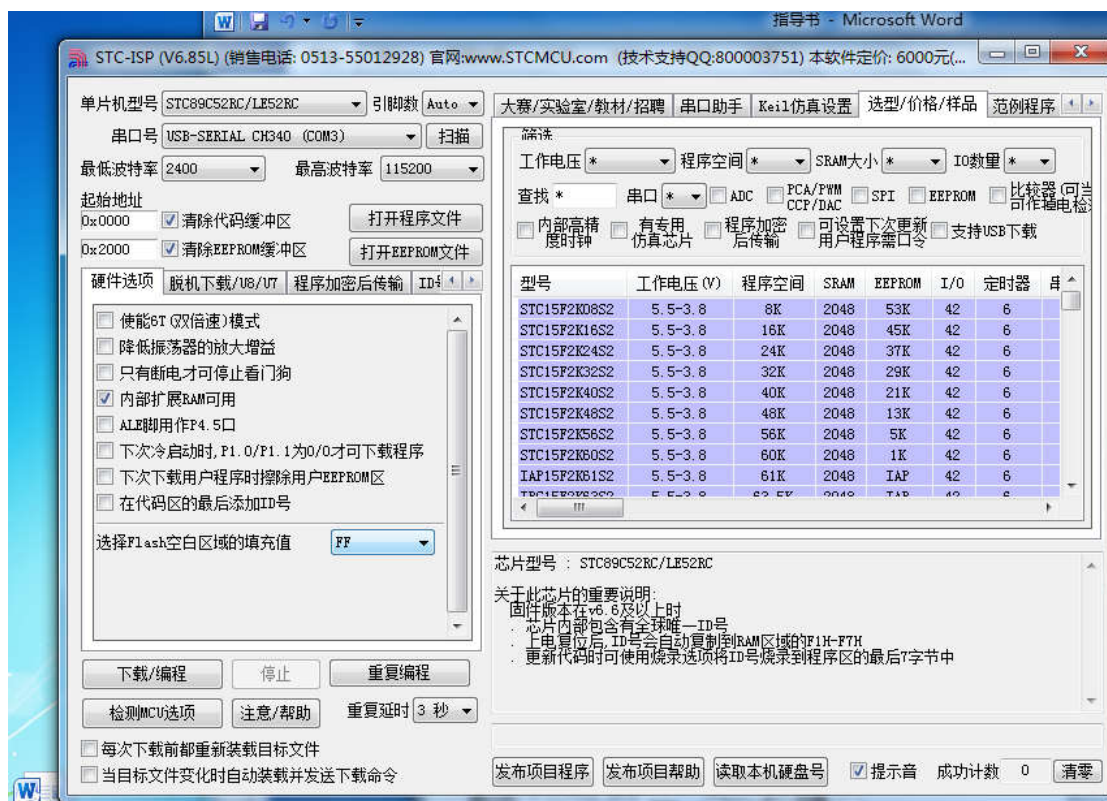
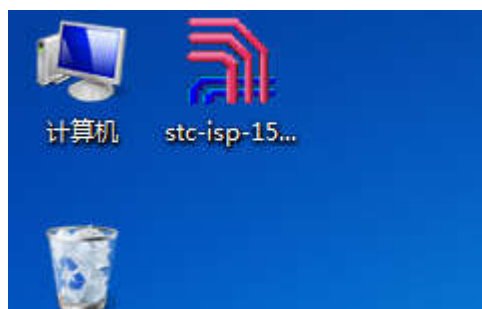
13 在 debug 选项下将右侧 Use 选上，点击 OK



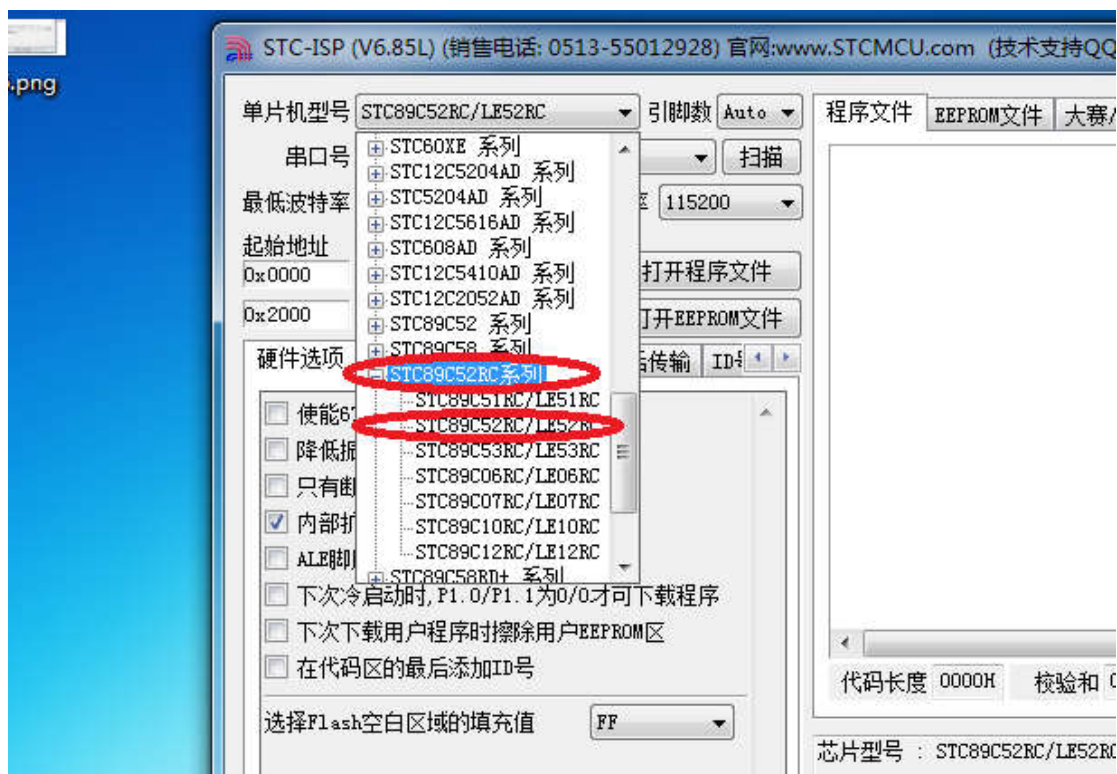
## 附 2：下载文件

- 1 连接电脑和下载器
- 2 打开下载软件

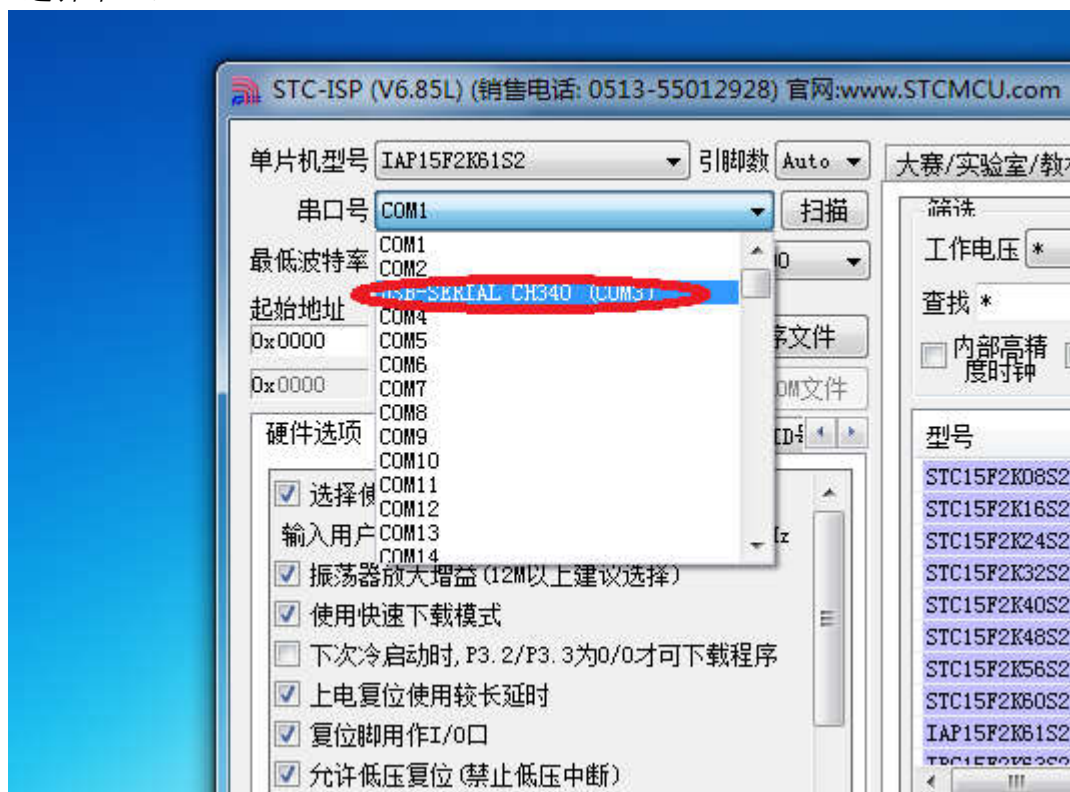




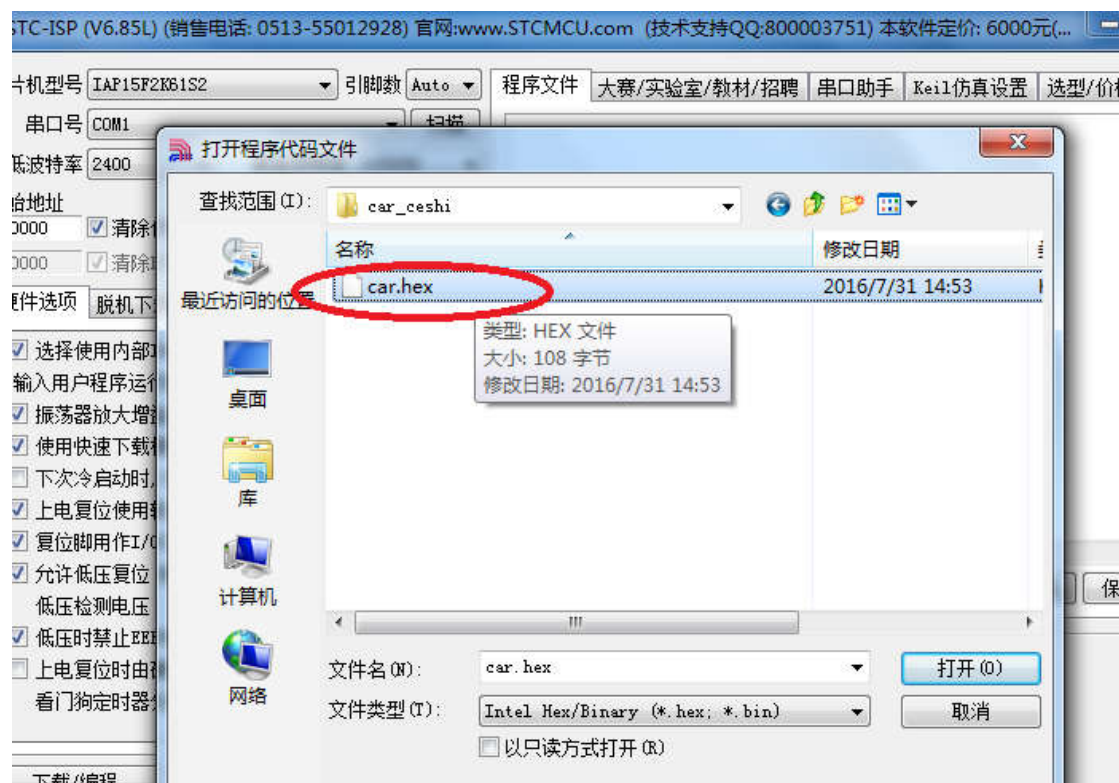
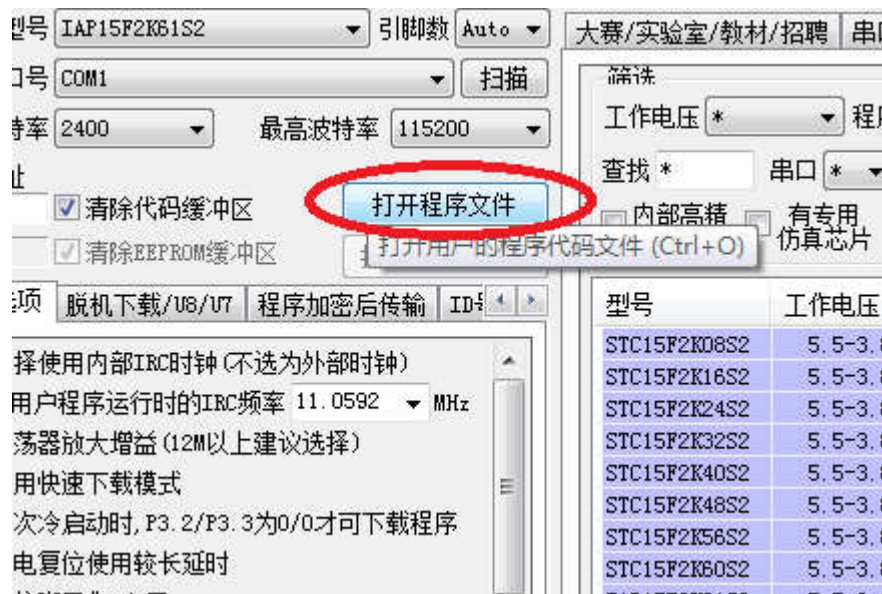
4 在单片机型号中选择 STC89C52RC 系列



3 选择串口，

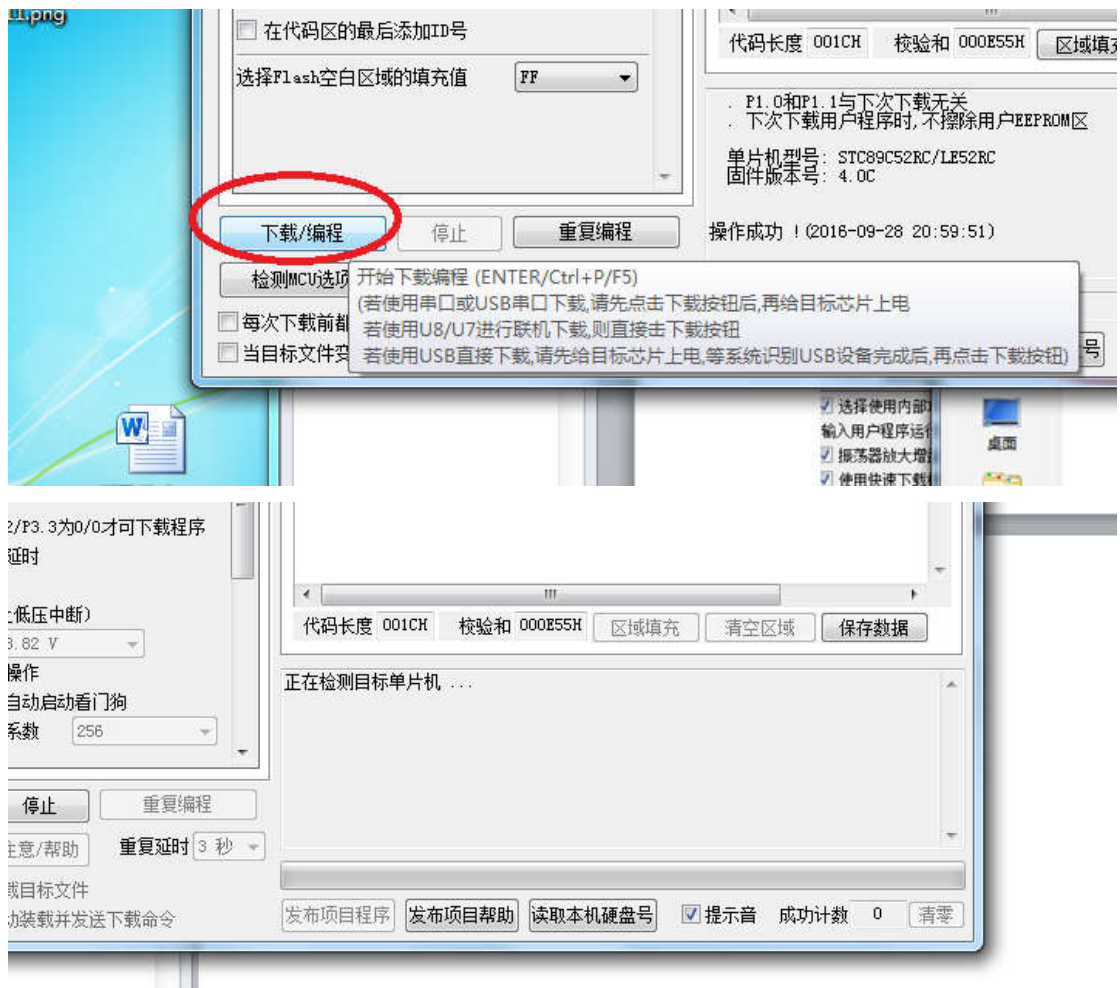


4 选择目标文件，选择自己所建的工程中的 .hex 文件



5 点击下载/编辑按钮





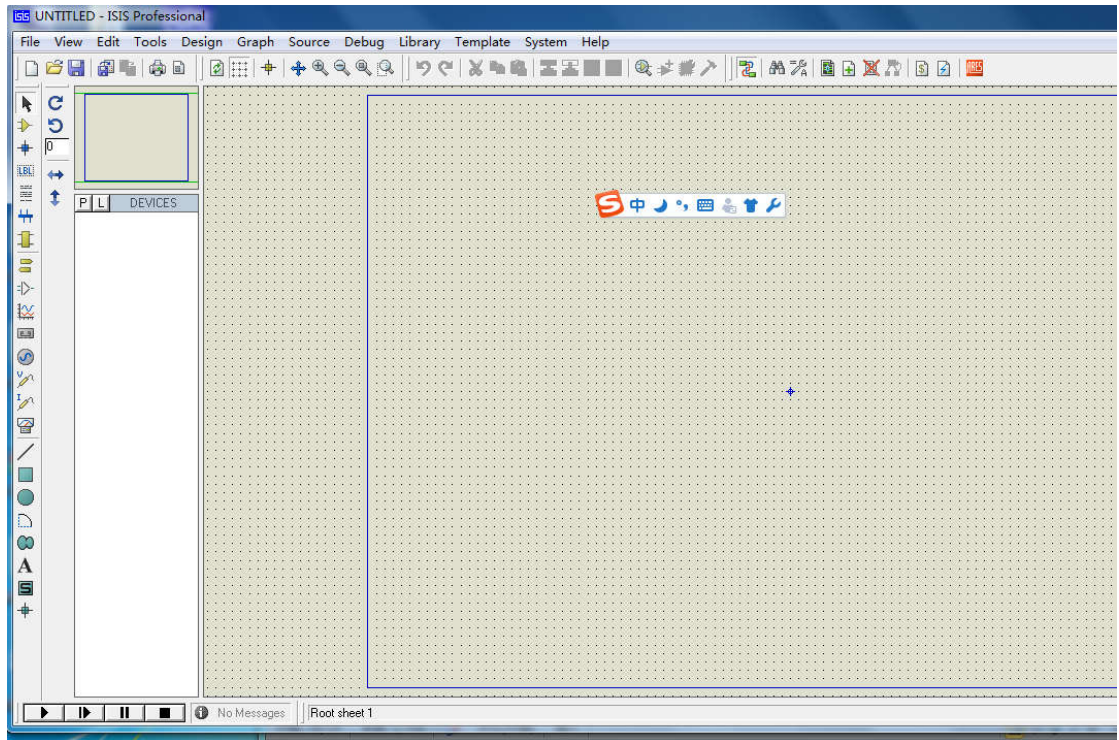
## 6 按下下载器的电源开关



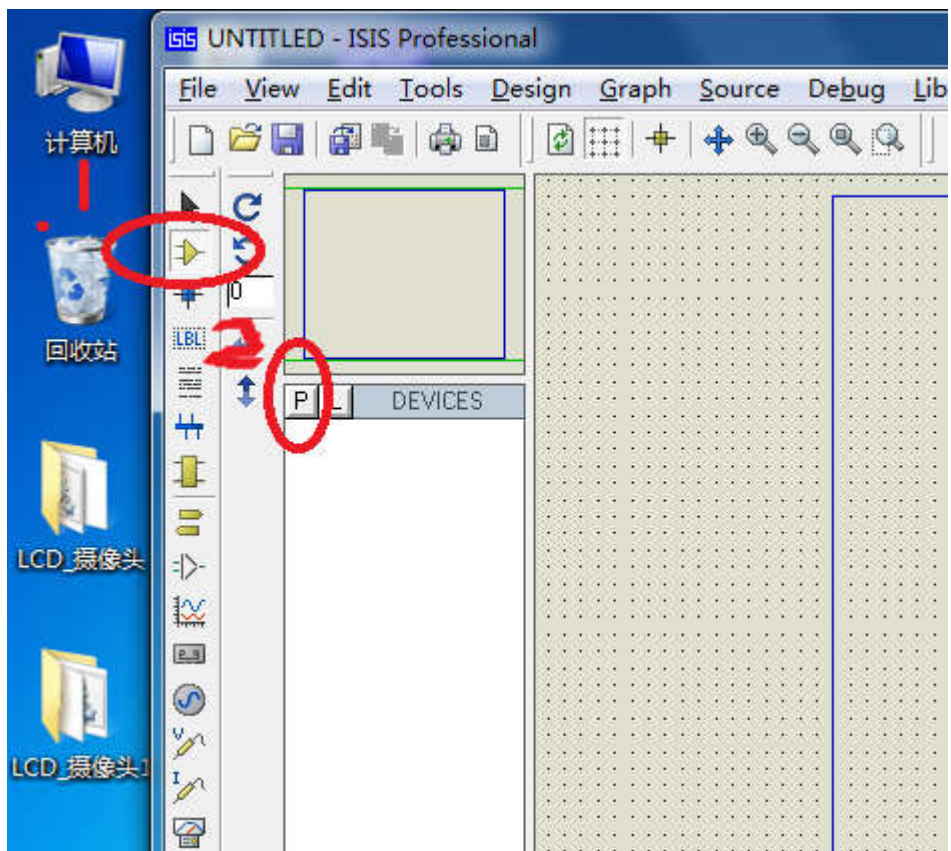
## 7 此时下载成功

### 附 3: proteus7.5 简易教程

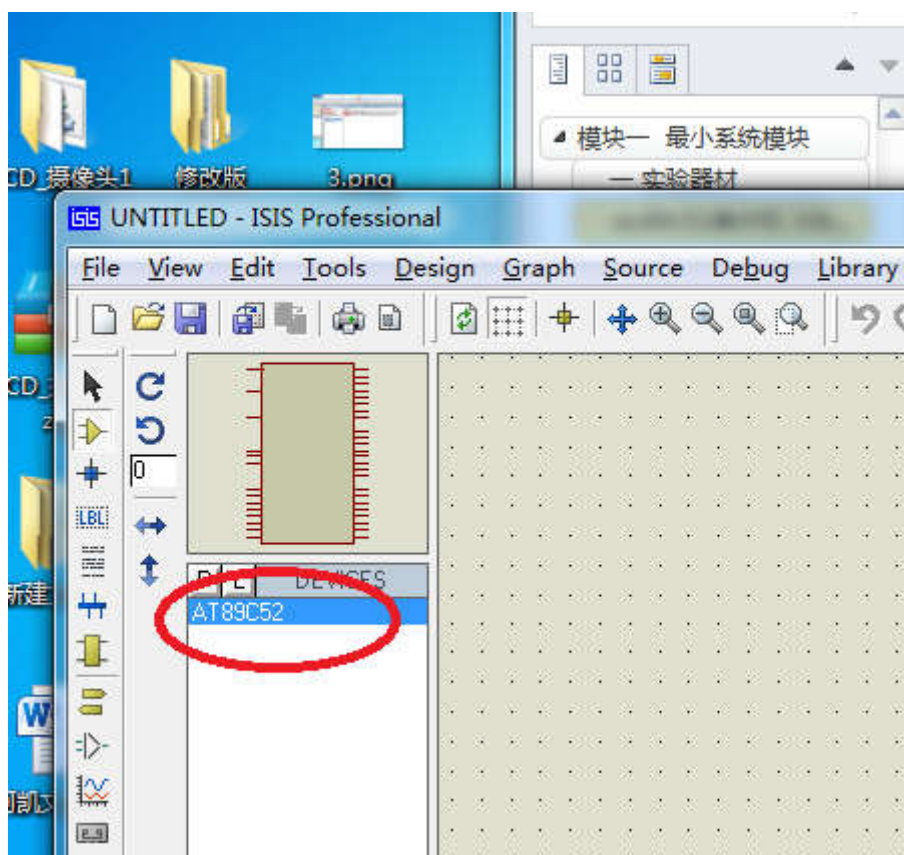
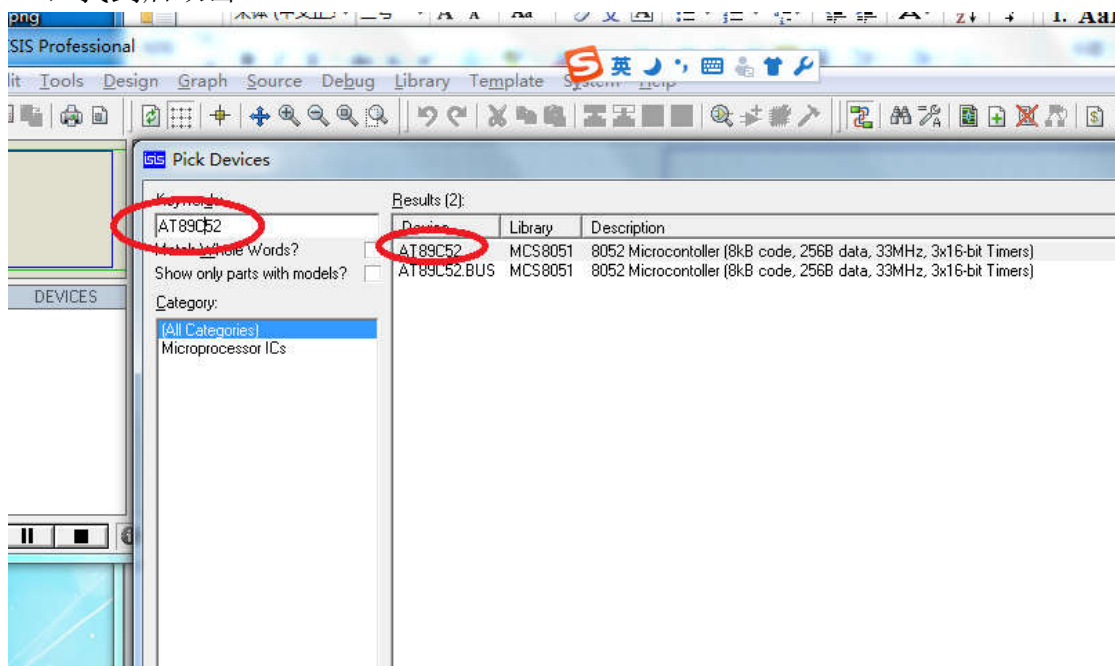
#### 1 打开 proteus 软件



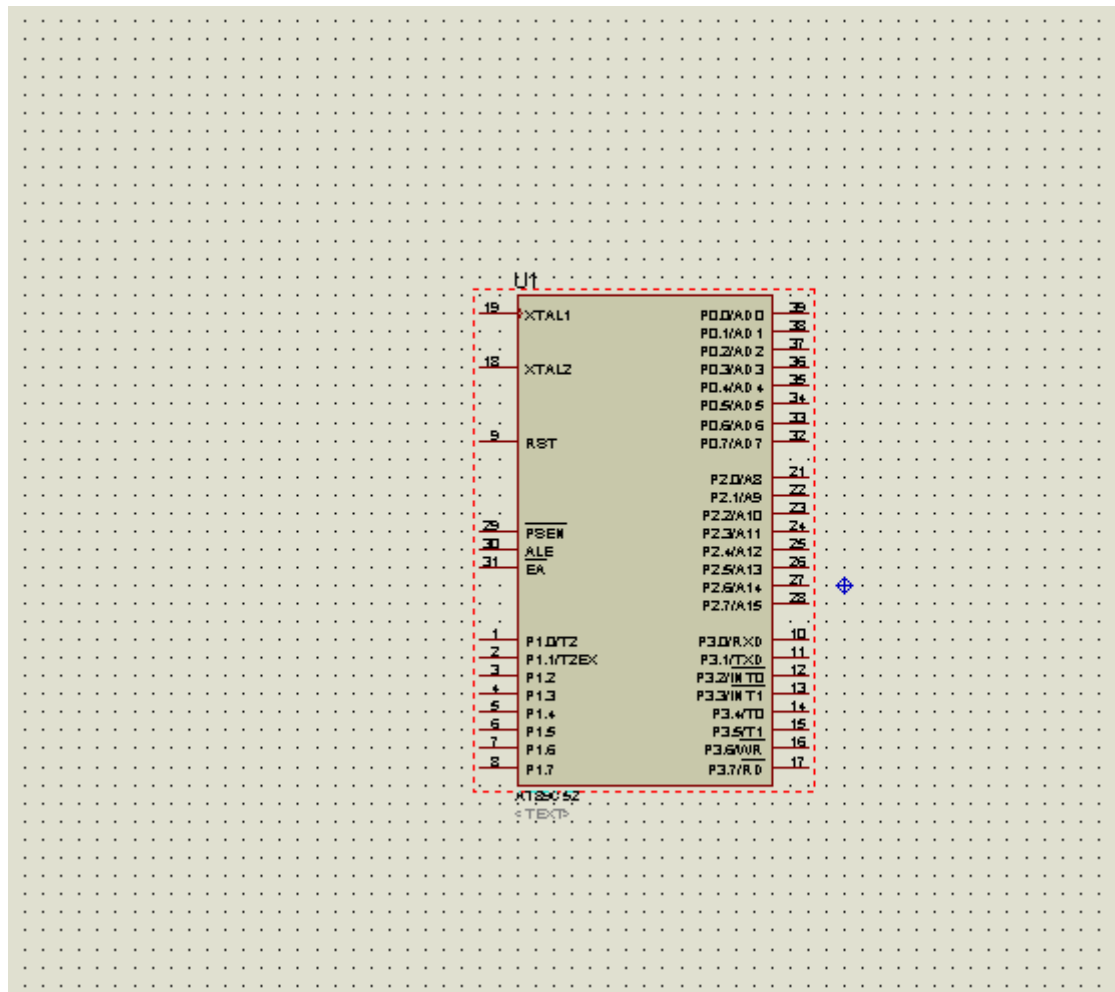
## 2 选择元器件（以选择单片机为例）



在弹出的对话框中的左上角 keywords 中输入所需元器件名称，例如 AT89C52，找到后双击



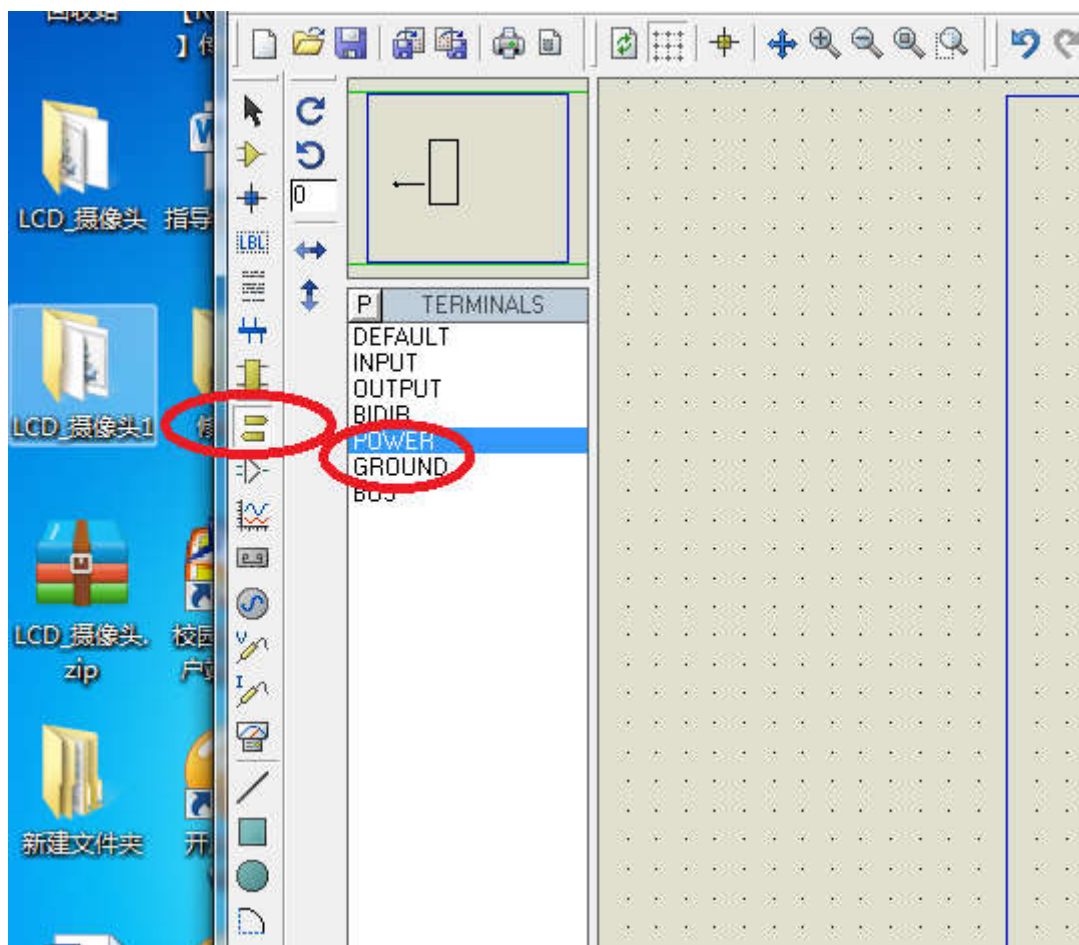
单击所需元件，然后在中间蓝色框内左击，调整元件位置，调整好后再左击



### 3 电源和地的选择

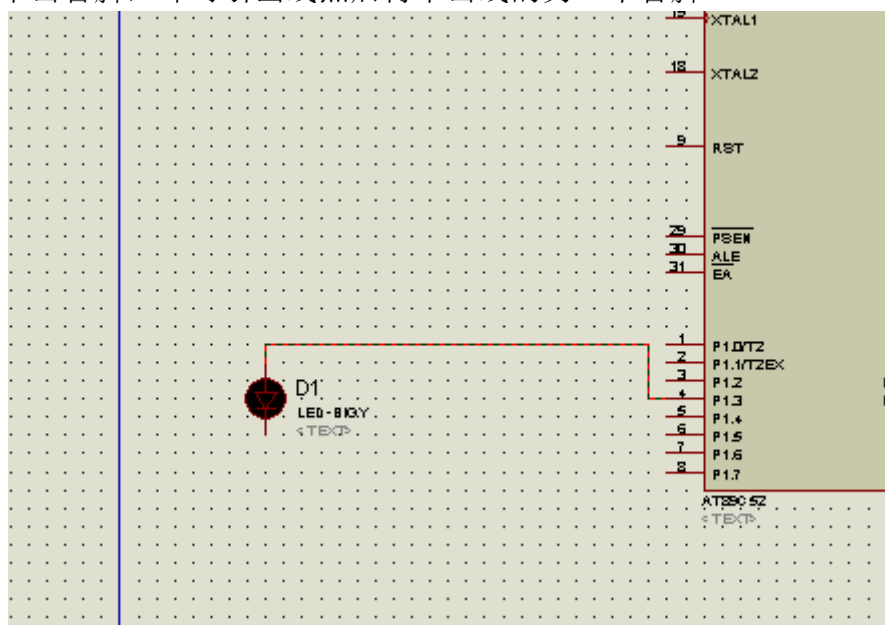
下图中的 power 代表电源，GROUND 代表地，选中





#### 4 连线

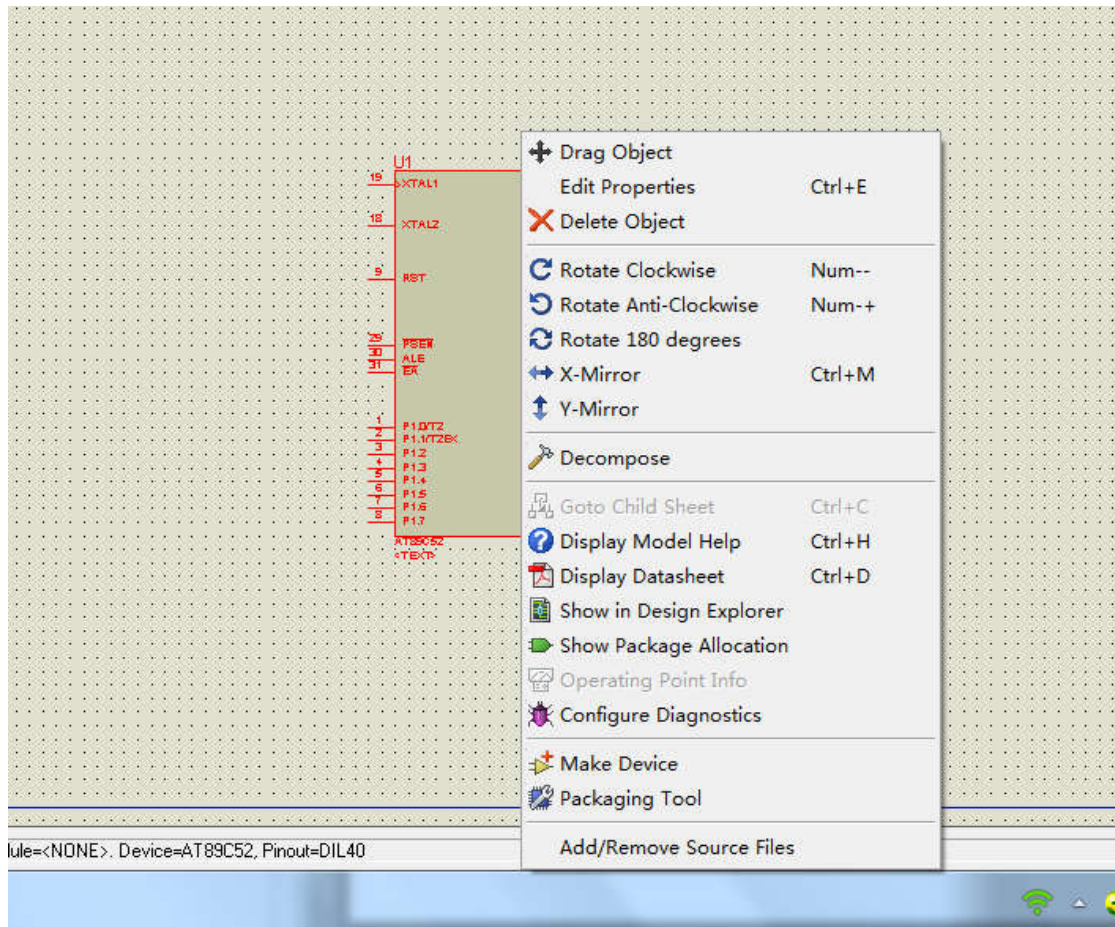
单击管脚，即可引出线然后再单击线的另一个管脚



#### 8 删除元件

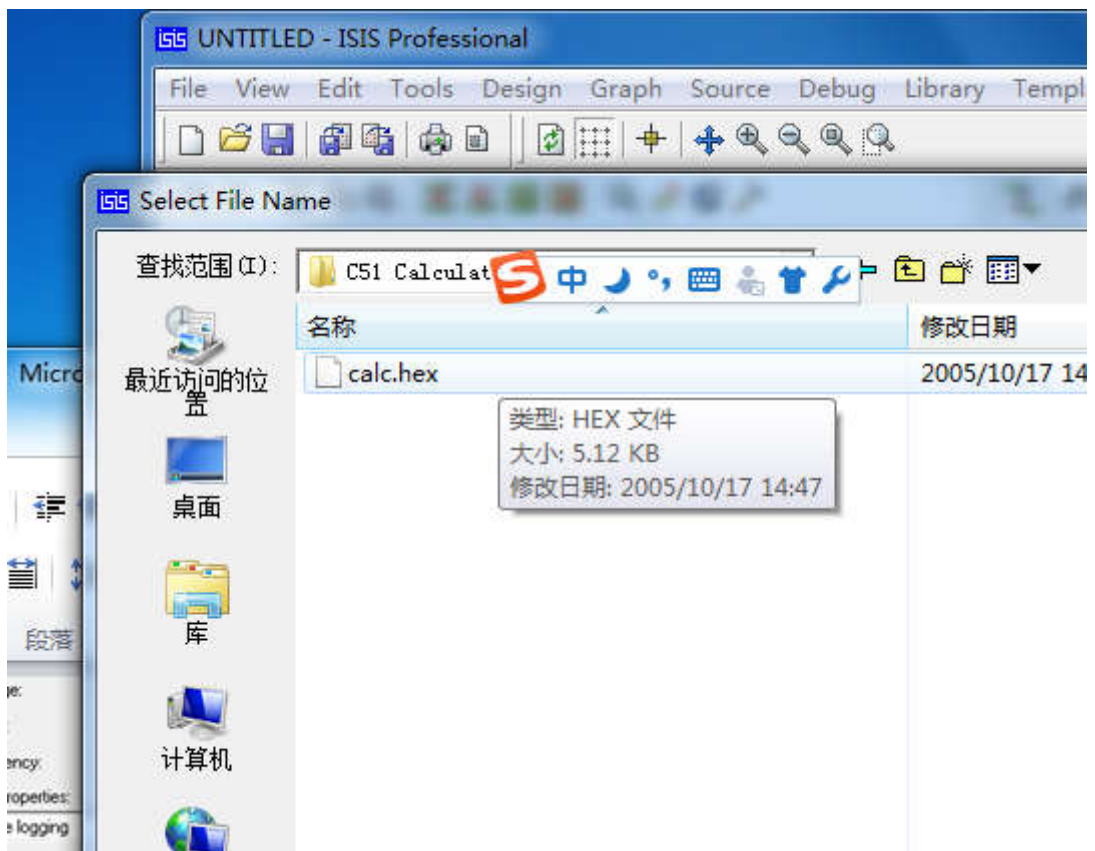
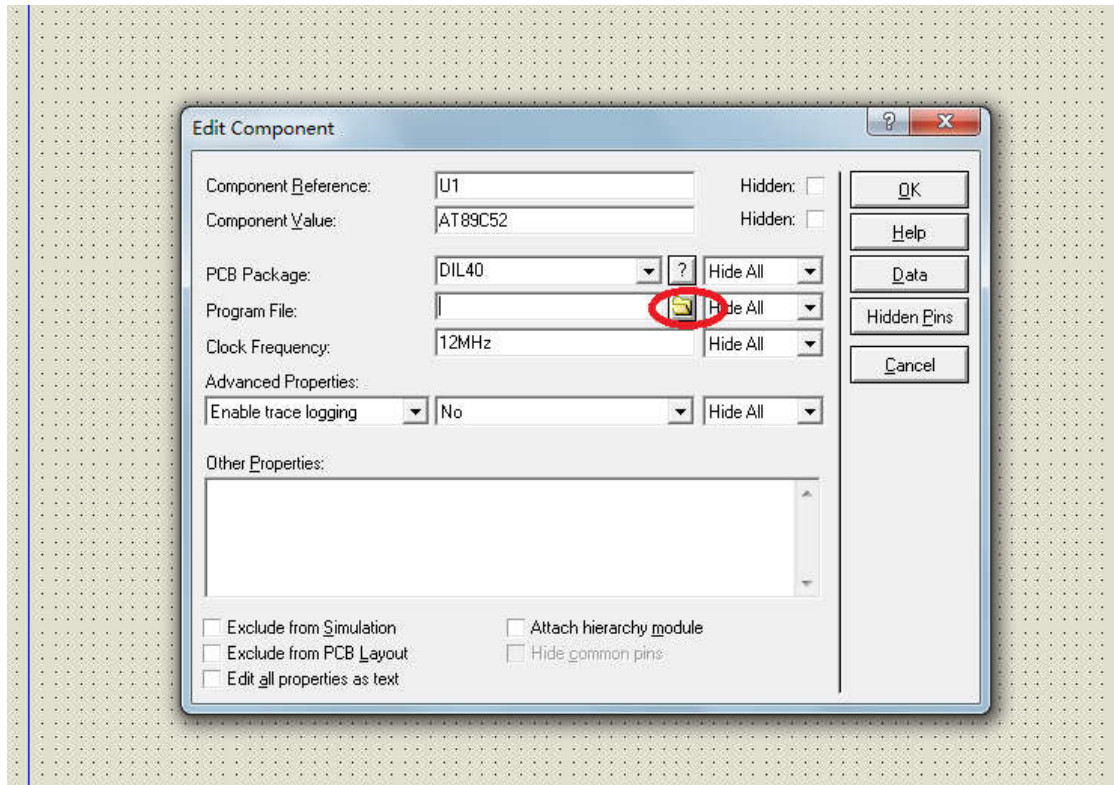
右击元件，在弹出对话框中选择删除，也可直接双右击直接删除



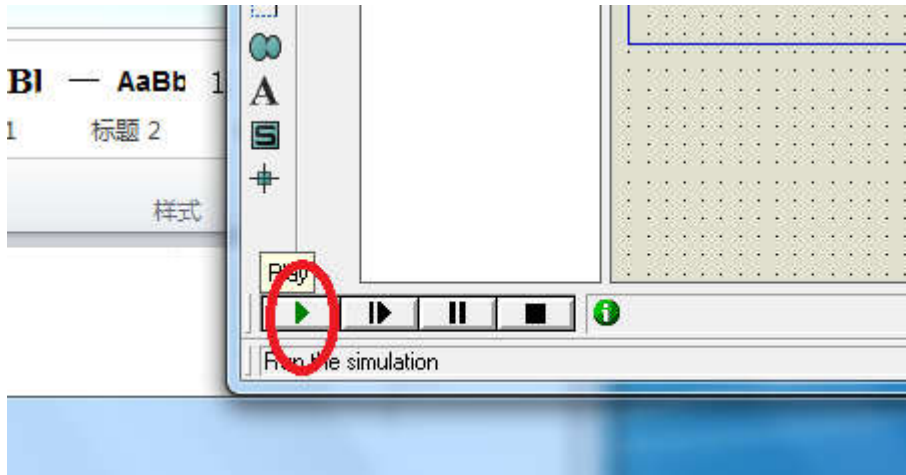


## 9 下载程序

双击单片机，在弹出对话框中单击 Program 后的按钮，选择自己的.hex 文件



然后单击 OK  
13 运行



这四个按钮以此为运行，单步运行，暂停，停止

## 实验三 驱动模块的实现

### 一、实验目的

- 1、掌握桥式驱动电路的运行原理；
- 2、掌握常用驱动模块的使用方法；
- 3、掌握直流电机正、反转的控制方法；
- 4、掌握直流电机调速的方法。

### 二、主要实验器材

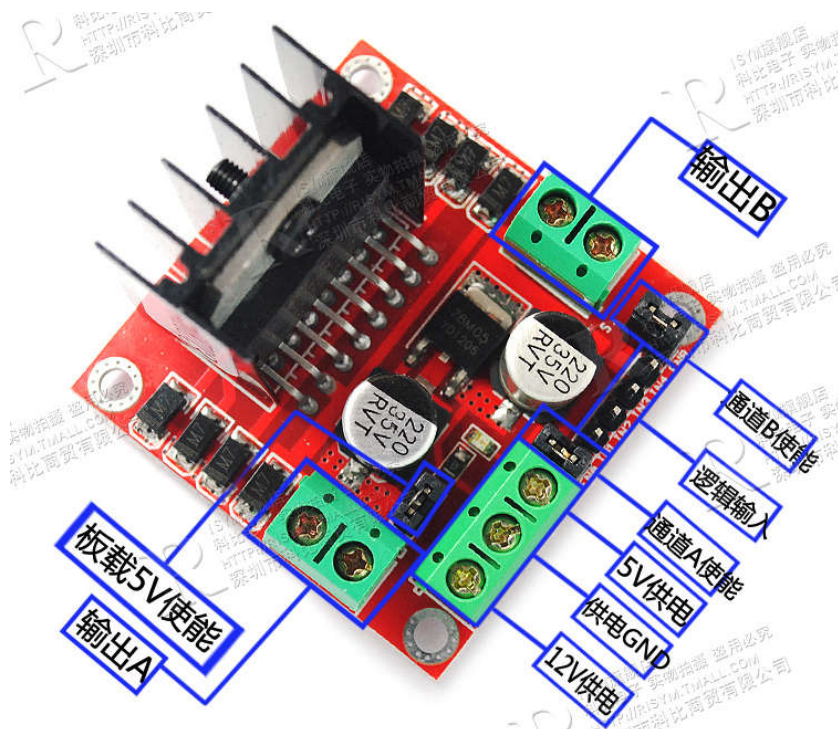


图 3.1 驱动模块

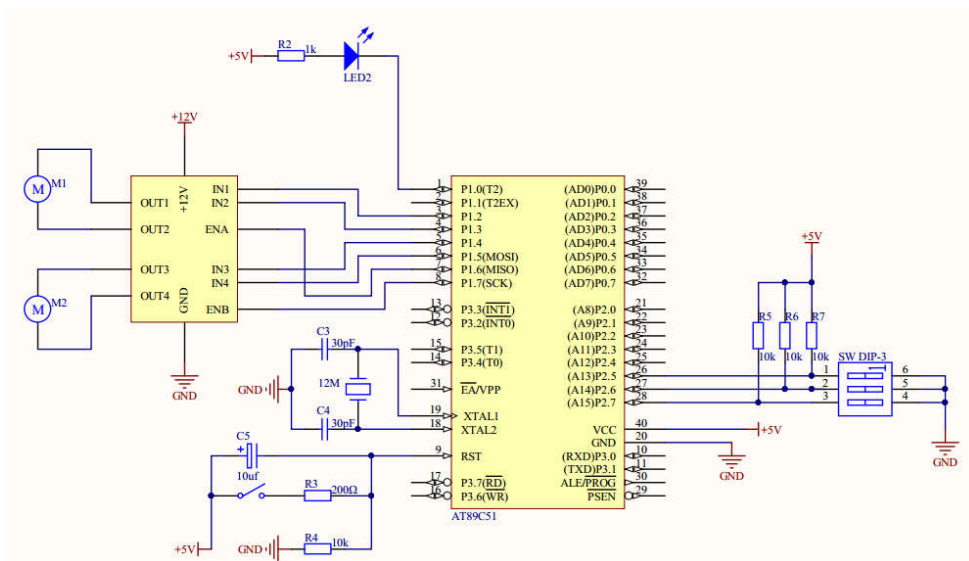


图 3.2 驱动电路

### 三、实验内容及步骤



1、驱动电路连接：将驱动模块上的 IN1—IN4 分别与单片机的 I/O 口（如 P1.2—P1.5）相连；驱动模块上的输出 A、输出 B 分别与左轮电机和右轮电机相连；驱动模块上的 12V 供电和供电 GND 分别与电池的正端及 GND 连接（**注意不能接反!**），通道 A 使能和通道 B 使能分别与单片机 I/O 口（如：P1.0—P1.1）相连。（**连接请使用杜邦线相连!**）

2、参考图 3.2 在 Keil 环境下编写电机正反转程序，在 proteus 中绘制电路图，进行仿真联合调试；仿真成功后，将目标文件下载至单片机，观察实验现象。（小车可自主前进和后退）

3、在 Keil 环境下编写电机调速程序（分别设置高、中、低三挡速度），在 proteus 中绘制电路图，进行仿真联合调试；仿真成功后，将目标文件下载至单片机，观察实验现象。（小车应以一定的速度前进 5 秒、后退 5 秒）

4、在 Keil 环境下编写小车走正方形程序，在 proteus 中绘制电路图，进行仿真联合调试；仿真成功后，将目标文件下载至单片机，观察实验现象。（小车应分别以高、中、低三挡速度走出规则正方形）

注：观察左右轮电机转动。若两轮转动方向不一致，可调整 IN1 与 IN2 或 IN3 与 IN4 的接线位置



图 3.2 电机正反转程序流程

#### 四、参考例程

程序 1-电机正反转程序：

```

#include<reg52.h>
Sbit P1^0 = ENA;
Sbit P1^1 = ENB;
Sbit P1^2 = IN1;
Sbit P1^3 = IN2;
Sbit P1^4 = IN3;
Sbit P1^5 = IN4;
  
```

```

void main()
{
    ENA = 1;
    ENB = 1;
    IN1 = 1;
    IN2 = 0;
    IN3 = 1;
    IN4 = 0;
    while(1)
    {
        ;
    }
}

```

程序 2-电机调速程序:

```

/* =====直流电机的 PWM 速度控制程序===== */
/* 晶振采用 11.0592M, 产生的 PWM 的频率约为 91Hz */
#include<reg51.h>
#include<math.h>
#define uchar unsigned char
#define uint unsigned int
sbit en1=P2^0; /* L298 的 Enable A */
sbit en2=P2^1; /* L298 的 Enable B */
sbit s1=P2^2; /* L298 的 Input 1 */
sbit s2=P2^3; /* L298 的 Input 2 */
sbit s3=P2^4; /* L298 的 Input 3 */
sbit s4=P2^5; /* L298 的 Input 4 */
uchar t=0; /* 中断计数器 */
uchar m1=0; /* 电机 1 速度值 */
uchar m2=0; /* 电机 2 速度值 */
uchar tmp1, tmp2; /* 电机当前速度值 */

/* 电机控制函数 index-电机号(1,2); speed-电机速度(-100—100) */
void motor(uchar index, char speed)
{
    if(speed>=-100 && speed<=100)
    {
        if(index==1) /* 电机 1 的处理 */
        {
            m1=abs(speed); /* 取速度的绝对值 */
            if(speed<0) /* 速度值为负则反转 */
            {
                s1=0;
                s2=1;
            }
        }
    }
}

```

```
    }
    else /* 不为负数则正转 */
    {
        s1=1;
        s2=0;
    }
}
if(index==2) /* 电机 2 的处理 */
{
    m2=abs(speed); /* 电机 2 的速度控制 */
    if(speed<0) /* 电机 2 的方向控制 */
    {
        s3=0;
        s4=1;
    }
    else
    {
        s3=1;
        s4=0;
    }
}
}
}

void delay(uint j) /* 简易延时函数 */
{
    for(j;j>0;j--);
}

void main()
{
    char i;
    TMOD=0x02; /* 设定 T0 的工作模式为 2 */
    TH0=0x9B; /* 装入定时器的初值 */
    TL0=0x9B;
    EA=1; /* 开中断 */
    ET0=1; /* 定时器 0 允许中断 */
    TR0=1; /* 启动定时器 0 */
    while(1) /* 电机实际控制演示 */
    {
        for(i=0;i<=100;i++) /* 正转加速 */
        {
            motor(1, i);
            motor(2, i);
```

```
    delay(5000);
}
for(i=100;i>0;i--) /* 正转减速 */
{
    motor(1,i);
    motor(2,i);
    delay(5000);
}
for(i=0;i<=100;i++) /* 反转加速 */
{
    motor(1,-i);
    motor(2,-i);
    delay(5000);
}
for(i=100;i>0;i--) /* 反转减速 */
{
    motor(1,-i);
    motor(2,-i);
    delay(5000);
}
}
}

void timer0() interrupt 1 /* T0 中断服务程序 */
{
    if(t==0) /* 1 个 PWM 周期完成后才会接受新数值 */
    {
        tmp1=m1;
        tmp2=m2;
    }
    if(t<tmp1) en1=1; else en1=0; /* 产生电机 1 的 PWM 信号 */
    if(t<tmp2) en2=1; else en2=0; /* 产生电机 2 的 PWM 信号 */
    t++;
    if(t>=100) t=0; /* 1 个 PWM 信号由 100 次中断产生 */
}
```



## 实验四 人机接口模块实现

### 一、实验目的

- 1、掌握七段数码管的显示原理；
- 2、掌握两位一体数码管的使用方法；
- 3、掌握按键消抖的一般方法。

### 二、主要实验器材

两位一体共阳数码管；NPN三极管两个；10k 排阻一个；小按键一个

### 三、实验内容及步骤

- 1、参考图 4.1 在 Protues 中绘制电路图；
- 2、在 Keil 中编写“数码管显示”程序，进行仿真综合调试，观察实验现象（显示固定数字如“21”）；
- 3、在 Keil 中编写“按键加一”程序，进行仿真综合调试，观察实验现象；（按键每按下一次，数码管显示的数字加一）
- 4、参考图 4.1 在实验板上焊接电路，下载程序，运行系统，观察实验现象（按键每按下一次，数码管显示的数字加一）；

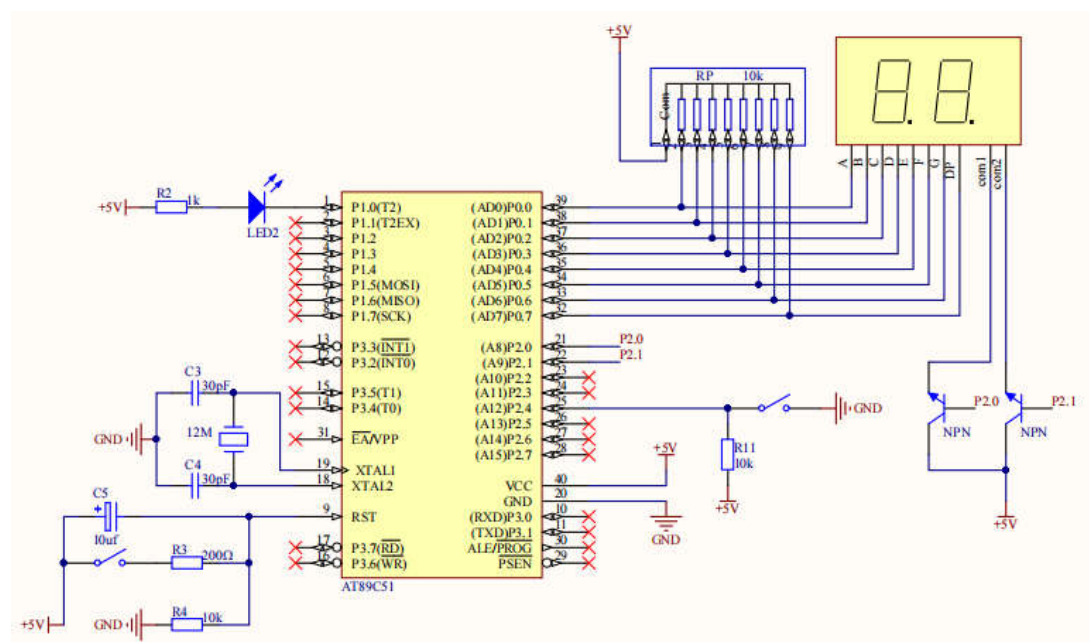


图 4.1 两位数码管应用电路

### 四、参考例程

程序 1----数码管显示 21

```
#include<reg52.h>
#define uint unsigned int
#define uchar unsigned char
uchar code DUAN_XUAN[]=
{0xc0,0xf9,0xa4,0xb0,0x99,0x92,0x82,0xf8,0x80,0x90,0x88,0x83,0xc6
```

```

, 0xa1, 0x86, 0x8e};//共阳极数码管编码数组
/*****
*程序目的：延时
*****/
void delay(uint x)
{
    uint i;
    while(x--)
    {
        for(i=0;i<121;i++);
    }
}
/*****
*函数作用：显示一个数字
*函数参数：y
*返回值：无
*****/
void display(uint y)
{
    ucharshi, ge;
    shi=y%100/10;
    ge=y%10;
    P0=0xff;//消隐
    P2=0x02;
    P0=DUAN_XUAN[ge];
    delay(10);
    P0=0xff;
    P2=0x01;
    P0=DUAN_XUAN[shi];
    delay(10);
}
void main()
{
    uint a;
    a=21;
    while(1)
    {
        display(a);
    }
}

```

程序 2-----按键加一（需自行完成）

## 实验五 循迹模块设计与实现

### 一、实验目的

- 1、掌握常用光电传感器的使用方法；
- 2、掌握循迹的一般原理。

### 二、主要实验器材

ST188；lm324；2 个 10k 的滑动变阻器；2 个 led 指示灯；10k、1k、500  $\Omega$  电阻各两个。

### 三、实验原理图

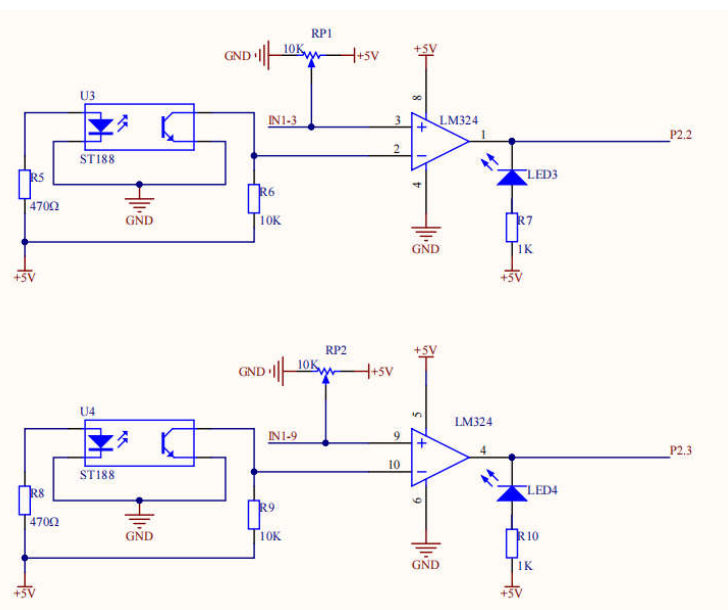


图 5.1 循迹模块原理图

如图 5.1 所示电路，其中 lm324 起比较放大作用，一个 lm324n 芯片里面有四组放大器，在这里选用其中两组即可。图 5.1 中接单片机口建议接到单片机中断口 P3.2 P3.3 上。具体实物如图 5.2 所示。

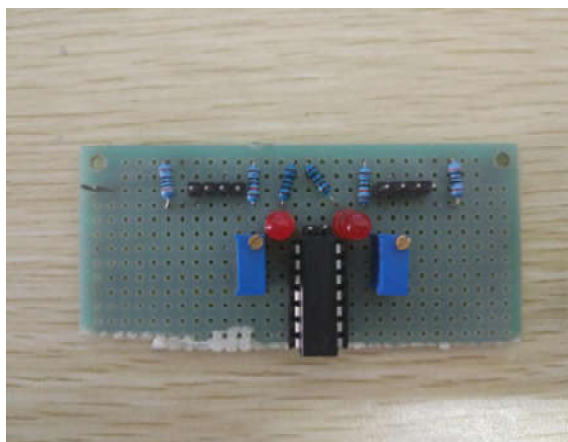


图 5.2 循迹模块实物图

注：图中两组四口排针接 st188 传感器即可。

### 四、实验步骤

- 1、参考图 5.1 在实验板上焊接硬件电路；
- 2、对步骤 1 焊接好的电路进行测试，具体测试方法如下：
  - ①改变滑动变阻器阻值，使得两个 led 灯处于常亮状态。
  - ②在桌子上粘贴一段黑胶带，将 st188 传感器放在合适的位置（一般为黑线上方 1cm~2cm）如果传感器检测到黑线灯灭，则整个电路焊接正确。如果没有，则继续检查电路。

## 实验六 测速模块实现

### 一、实验目的

- 1、掌握霍尔传感器的使用方法；
- 2、掌握利用脉冲数进行速度计算的方法。

### 二、主要实验器材

霍尔传感器 1 个；小磁铁 1 个

### 三、实验原理

当传感器未感应到磁铁后，传感器上只有电源指示灯会被点亮，传感器输出端口输出高电平。当传感器感应到磁铁后，传感器上相应的指示灯会被点亮，传感器输出端口输出低电平。

将传感器输出端口接至单片机相应端口进行检测，原理类似于按键检测，得到一分钟内检测的数值，即可测出小车的平均速度。

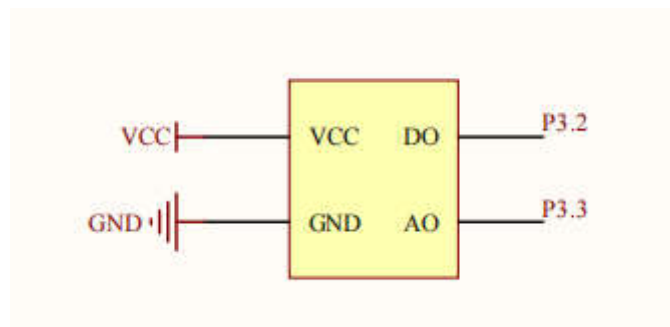


图 6.1 测速模块连接示意图

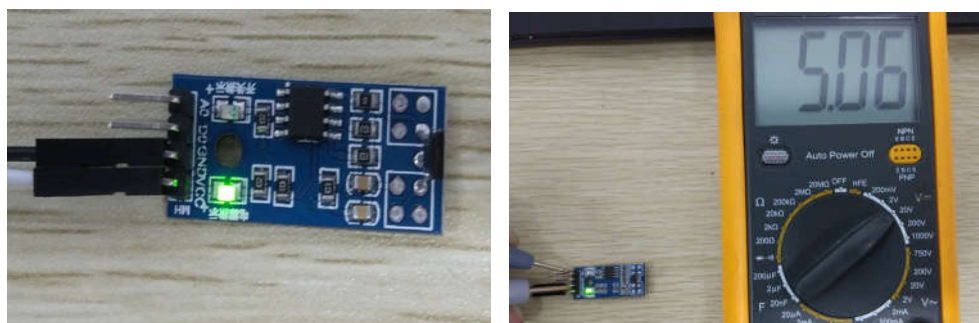


图 6.2 未检测到磁场

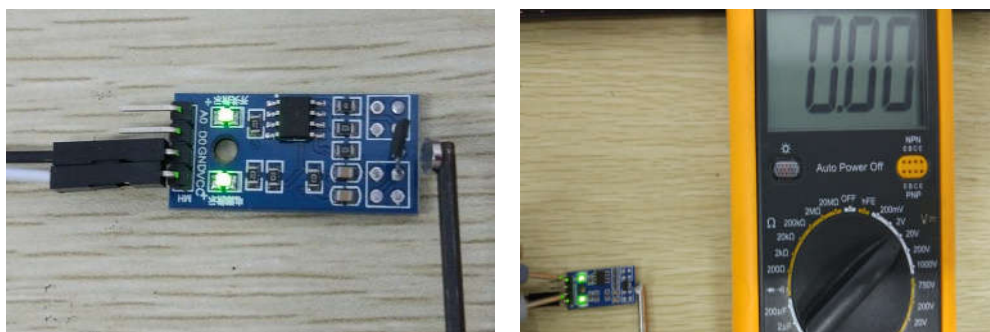


图 6.3 检测到磁场

#### 四、实验步骤

1、在 proteus 中绘制电路图，可将信号发生器接至单片机的 I/O 口，用信号发生器模拟产生一定频率的方波信号，然后通过单片机的 I/O 口，使用单片机的定时器/计数器对方波信号的个数进行计数，实现对方波信号频率的测量。

2、在 keil 中编写“计数程序”，进行仿真联合调试。将方波信号的频率显示在数码管上。

3、将传感器模块和小磁铁安装于小车相应位置，连接相关电路（给模块供电及与单片机的连接等）。

4、编写程序检测小车速度（单位：米/分钟）并显示在数码管上。

## 实验七 综合实验---循迹模块和驱动模块的综合

### 一、实验目的

- 1、掌握简单系统软硬件综合调试的一般方法；
- 2、掌握模块软硬件综合的一般方法。

### 二、主要实验器材

循迹模块；驱动模块。

### 三、实验内容及步骤

- 1、理清驱动模块、循迹模块同单片机最小系统硬件连接情况，查看是否存在冲突，若存在，则修改相应连接；若不存在，则连接好相应线路；
- 2、理清驱动模块、循迹模块对应的软件程序中变量定义、程序结构等方面的情况，将其综合整理成一个完整的程序；
- 3、将目标文件下载至单片机，系统上电运行，观察实验现象。重点查看小车是否可沿黑色引导线自主移动。

## 实验八 综合实验---循迹模块、驱动模块、测速模块及显示模块的综合

### 一、实验目的

- 1、进一步掌握系统软硬件综合调试的一般方法；
- 2、进一步掌握模块软硬件综合的一般方法。

### 二、主要实验器材

循迹模块；驱动模块；显示模块。

### 三、实验内容及步骤

- 1、理清驱动模块、循迹模块、测速模块、显示模块同单片机最小系统硬件连接情况，查看是否存在冲突，若存在，则修改相应连接；若不存在，则连接好相应线路；
- 2、理清驱动模块、循迹模块、测速模块、显示模块对应的软件程序中变量定义、程序结构等方面的情况，将其综合整理成一个完整的程序；
- 3、将目标文件下载至单片机，系统上电运行，观察实验现象。重点查看小车是否可沿黑色引导线自主、稳定、快速的移动，并实时显示小车速度。