

《单片机原理及接口技术》

MiniSTC 实验板指导书 V1.1

南京工业大学计算机科学与技术学院

信息科学与技术实验教学中心

2023 年 3 月

目 录

第一章 实验设备简介.....	2
1.1 前言.....	2
1.2 设备特色.....	2
1.3 实验板布局介绍.....	2
1.4 模块原理图.....	4
第二章 软件使用说明.....	6
2.1 KEIL 集成开发环境简介.....	6
2.2 KEIL 集成开发环安装.....	6
2.3 uVision 应用说明	7
2.3.1 创建工程.....	7
2.3.2 建立源程序.....	10
2.3.3 程序编译.....	12
2.3.4 程序软件模拟调试.....	12
2.4 用户程序的下载.....	15
2.5 关于 STC 的头文件说明	15
第三章 软件实验.....	17
3.1 软件实验一 清零程序.....	17
3.2 软件实验二 拆字程序.....	18
3.3 软件实验三 拼字程序.....	19
3.4 软件实验四 数据区传送子程序.....	20
3.5 软件实验五 数据排序.....	21
3.6 软件实验六 数据查找.....	22
3.7 软件实验七 多分支程序.....	23
第四章 硬件实验.....	24
4.1 硬件实验一 跑马灯.....	24
4.2 硬件实验二 外部中断.....	25
4.3 硬件实验三 定时器.....	26
4.4 硬件实验四 脉冲计数.....	27
4.5 硬件实验五 串口通信.....	28
4.6 硬件实验六 AD 转换.....	29
4.7 硬件实验七 LCD 显示.....	30
4.8 硬件实验八 乐曲播放.....	31
第五章 综合性实验.....	32
5.1 综合性实验一 秒表.....	32
5.2 综合性实验二 电子钟.....	33

第一章 实验设备简介

1.1 前言

“从我多年来学习、开发、创办公司及服务客户的经验来看，单片机与CPLD的综合应用技术是一门实战非常强的学科，除了不断地学习之外，惟一的办法就是：实践！实践！！再实践！！！”这是周立功先生在一本技术书籍的序中的一段话。而作为工作在教学一线的单片教师，也总在思考这样的问题：怎样才能让学生切实有效的掌握这门知识，并很好的在实际中使用起来呢？显然仅靠课堂的理论学习以及有限时间的实验环节并不能很好地达到这个目的。结合自身的教学经验和对学生情况的了解，也非常赞同周先生的说法，所谓实践出真知！而实践教学需要硬件条件的配合，以往实验箱的方式在时间空间上都大大限制了学生学习的便利性。而本套实验设备突出了“口袋实验室”理念，同学们可以在寝室、图书馆、食堂，甚至任何一个可以工作和学习的地方，随时进行实验，算的上是一个全开放、全共享、课内外一体化的自主学习实践平台，让同学们可以很容易地“实践出真知”，很好地提高学习效率。

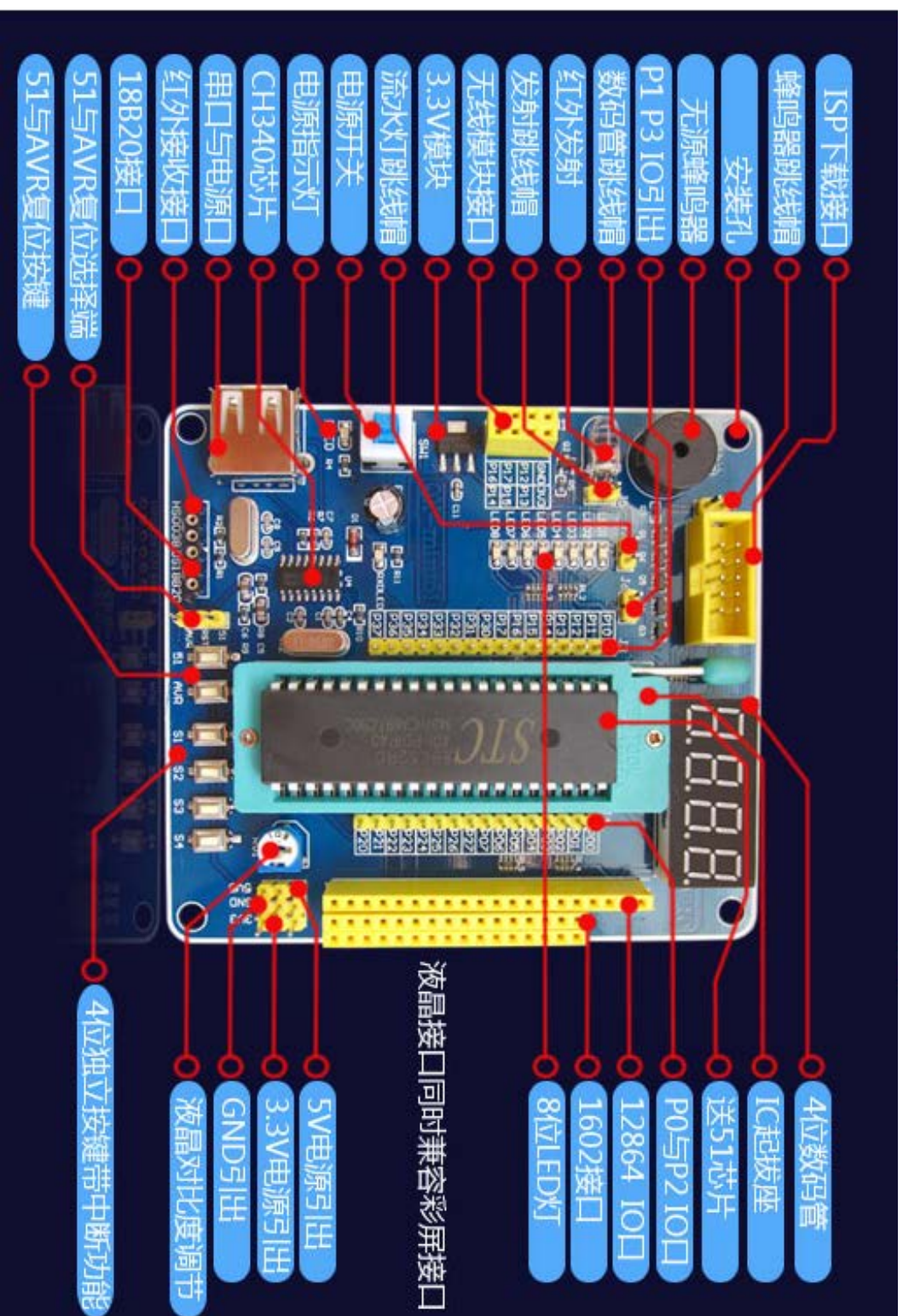
1.2 设备特色

本实验设备采用模块化组合设计，资源丰富且且无需额外配套工具，特别适合于单片机相关课程的实验，课程设计，实习和开放及创新实验等各类教学活动，当然也很适合作为自学者入门及提高的自备学习设备。

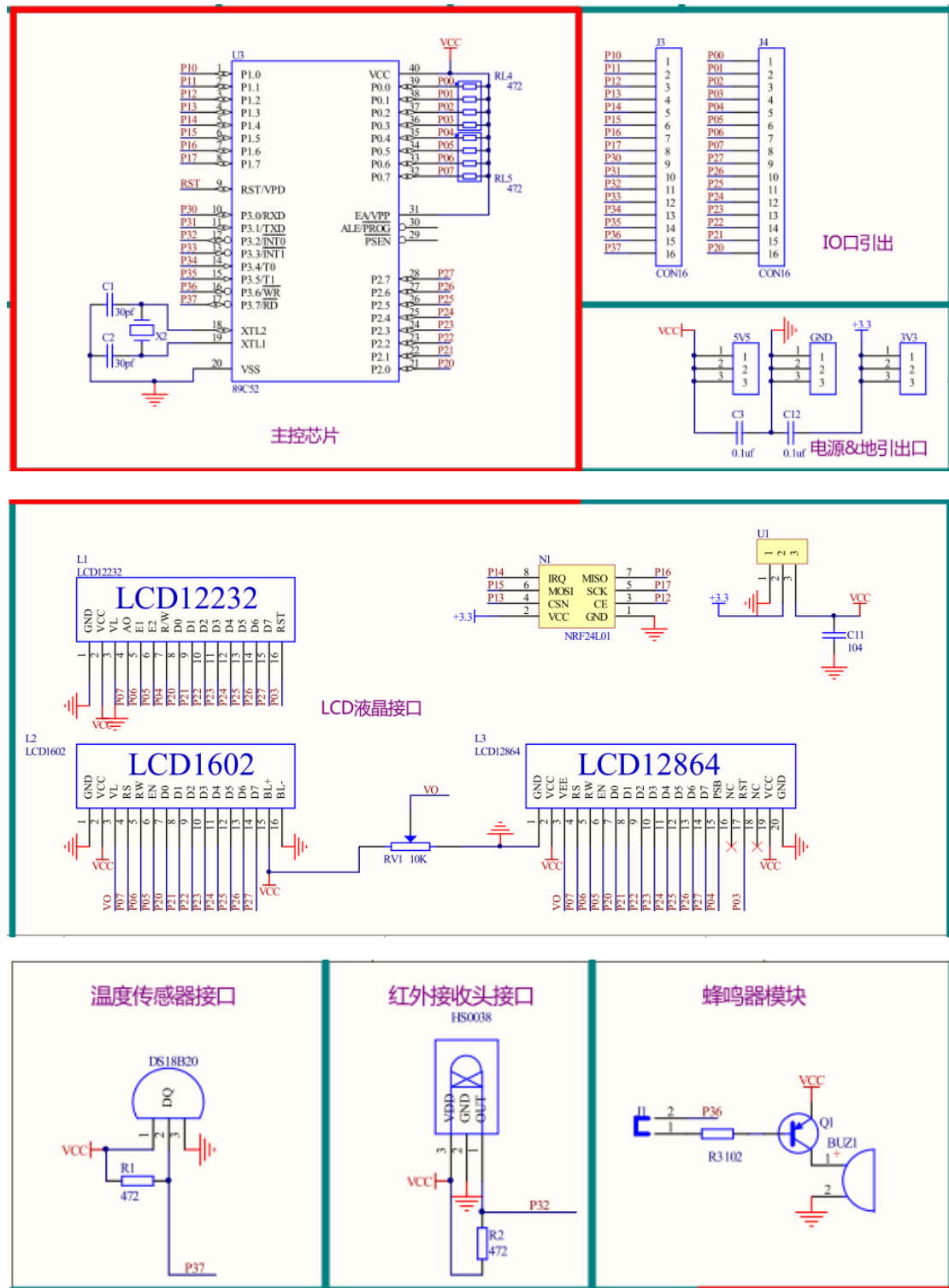
1. 全面兼容 WIN7/WIN8/WIN10/XP 32/64 位操作系统，出开发板外，只需一根 USB 接线即可完成功能，下载，串口通信等功能，特别方便携带和学习。
2. 本实验板硬件资源丰富，布局合理，性价比高，具体见 1.3 布局图。
3. 本实验板学习资料非常丰富，具体推荐见课件。

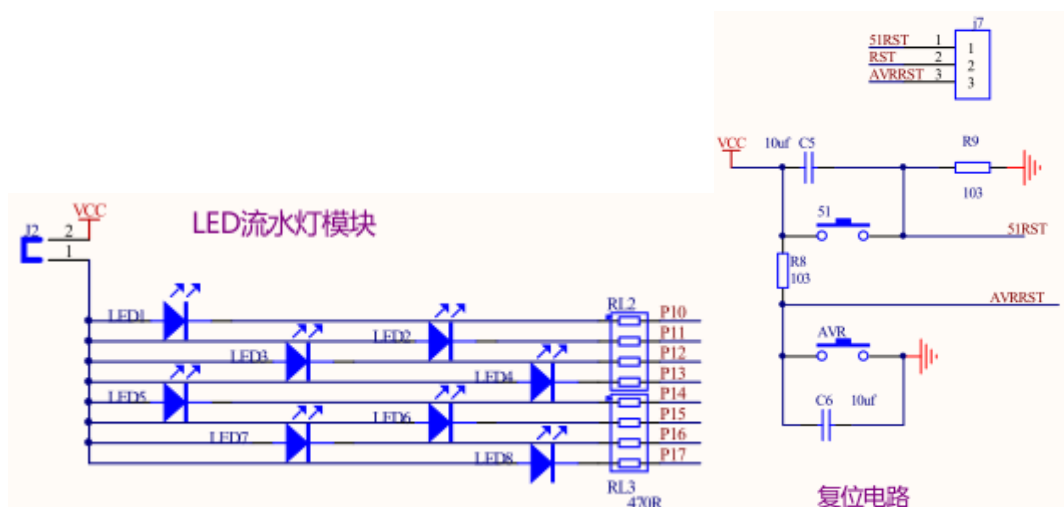
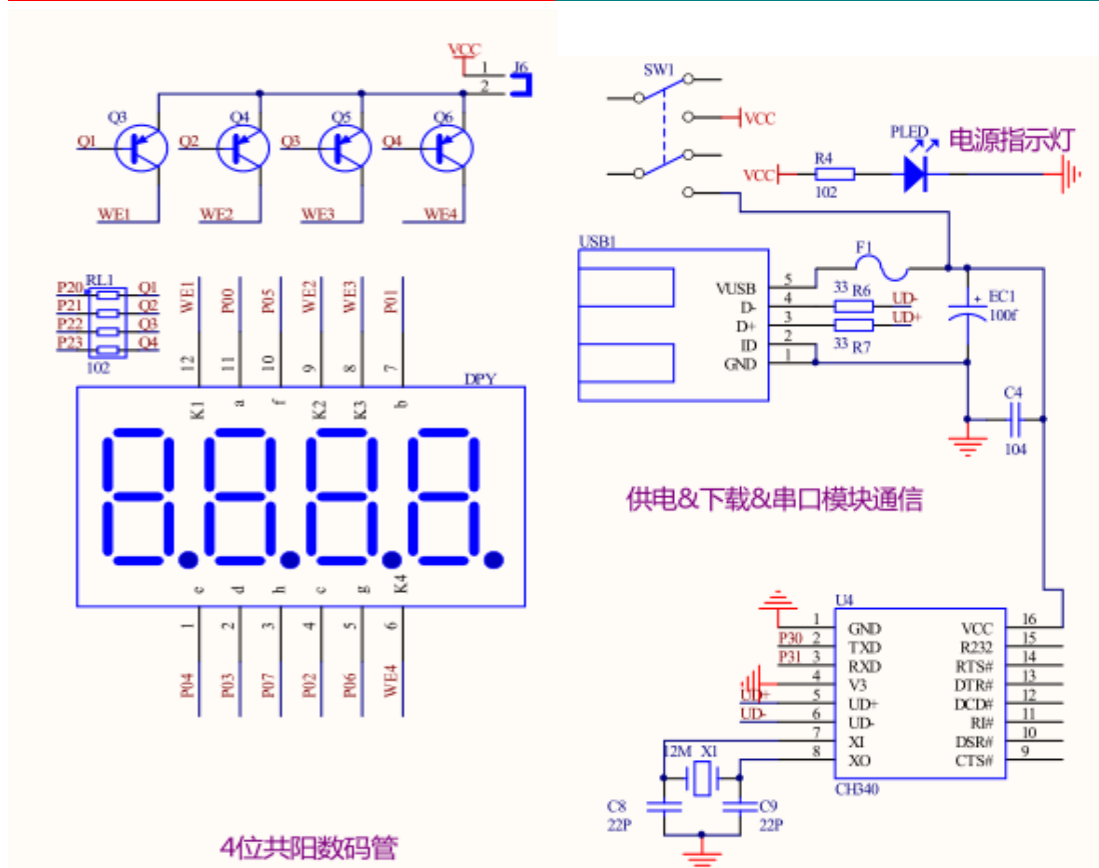
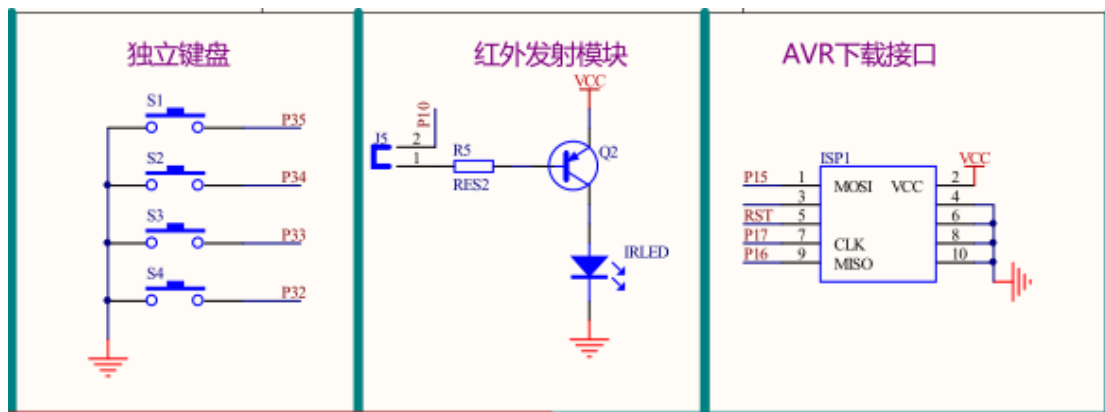
1.3 实验板布局介绍

实验板布局如下图：



1.4 模块原理图





第二章 软件使用说明

2.1 KEIL 集成开发环境简介

Keil Software 的8051开发工具提供以下程序，你可以用它们来编译你的C源码，汇编你的汇编源程序，连接和重定位你的目标文件和库文件，创建HEX文件，调试你的目标程序。

- Windows应用程序uVision2/3/4是一个集成开发环境，它把项目管理，源代码编辑，程序调试等集成到一个功能强大的环境中。
- C51美国标准优化C交叉编译器从你的C源代码产生可重定位的目标文件。
- A51宏汇编器从你的8051汇编源代码产生可重定位的目标文件。
- BL51连接/重定位器组合你的由C51和A51产生的可重定位的目标文件，生成绝对目标文件。
- LIB51库管理器组合你的目标文件，生成可以被连接器使用的库文件。
- OH51目标文件到HEX格式的转换器从绝对目标文件创建Intel HEX 格式的文件。
- RTX-51 实时操作系统简化了复杂和对时间要求敏感的软件项目。

uVision2/3/4包含一个器件数据库(device database)，可以自动设置汇编器、编译器、连接定位器和调试器选项，以满足用户选择各种不同型号微控制器的要求。此数据库包含：片上存储器和外围设备的信息、扩展数据指针(extra data pointer)和加速器(math accelerator)的特性。

2.2 KEIL 集成开发环安装

本实验板推荐安装 uVision4，软件在《STC 实验板资料》中的《3. 软件资料/开发环境(keil+uvision4)》的压缩包，安装教程的视频课参见在《STC 实验板资料》中的《0. mini51 板使用说明》的视频《二、编译软件的安装(C51V900).mp4》；

安装程序复制开发工具到基本目录的各个子目录中（以 uVision2 为例）。默认的基本目录是 C:\KEIL。下表列出的文件夹结构是包括所有开发工具的全部安装信息。

文件夹描述：

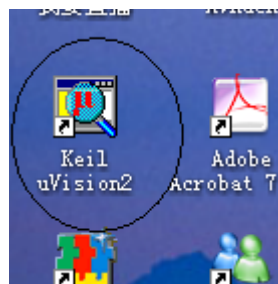
- C:\KEIL\C51\ASM 汇编 SFR 定义文件和模板源程序文件
- C:\KEIL\C51\BIN 8051 工具的执行文件
- C:\KEIL\C51\EXAMPLES 示例应用
- C:\KEIL\C51\RTX51 完全实时操作系统文件
- C:\KEIL\C51\RTX_TINY 小型实时操作系统文件
- C:\KEIL\C51\INC C 编译器包含文件
- C:\KEIL\C51\LIB C 编译器库文件启动代码和常规 I/O 资源
- C:\KEIL\C51\MONITOR 目标监控文件和用户硬件的监控配置
- C:\KEIL\UV2 普通 uVision2 文件

2.3 uVision 应用说明

本节描述了 uVision（以 uVision2 为例，uVision3/4 类似）的创建模式并向你演示如何使用它来创建一个简单的程序。以及生成和维护项目的一些选项；包括文件输出选项，C51 编译器的关于代码优化的配置，uVision2 项目管理器的特性等。

2.3.1 创建工程

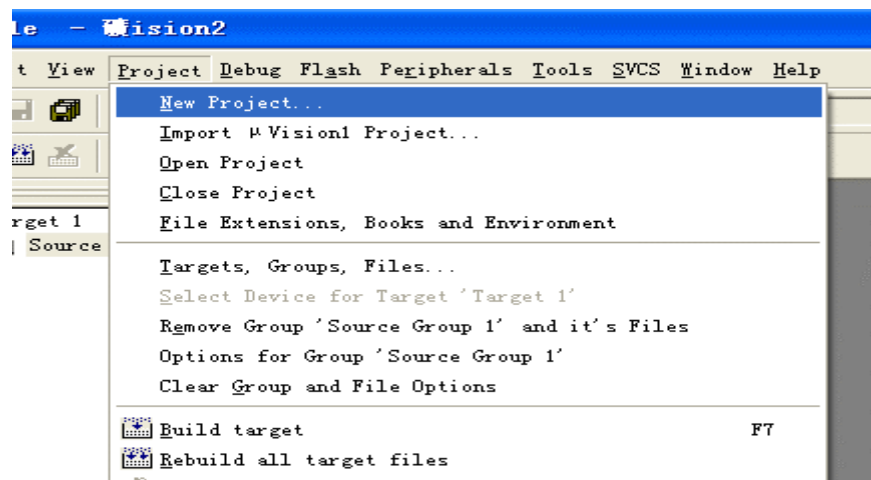
首先启动 KEIL 软件的集成开发环境，可以从桌面上直接双击 KEIL 快捷图标。如下图所示，进入 keil 集成开发环境。



KEIL 快捷图标

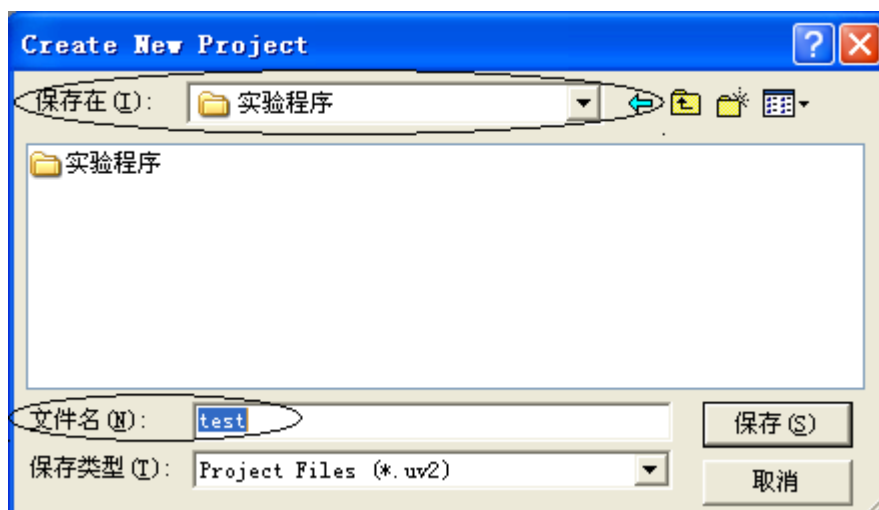
可以通过“PROJECT→OPEN PROJECT”直接进入已经建好的工程页面；也可以通过“PROJECT→NEW PROJECT”创建新的工程。

- (1) 点击“PROJECT→NEW PROJECT”创建新的工程，如下图



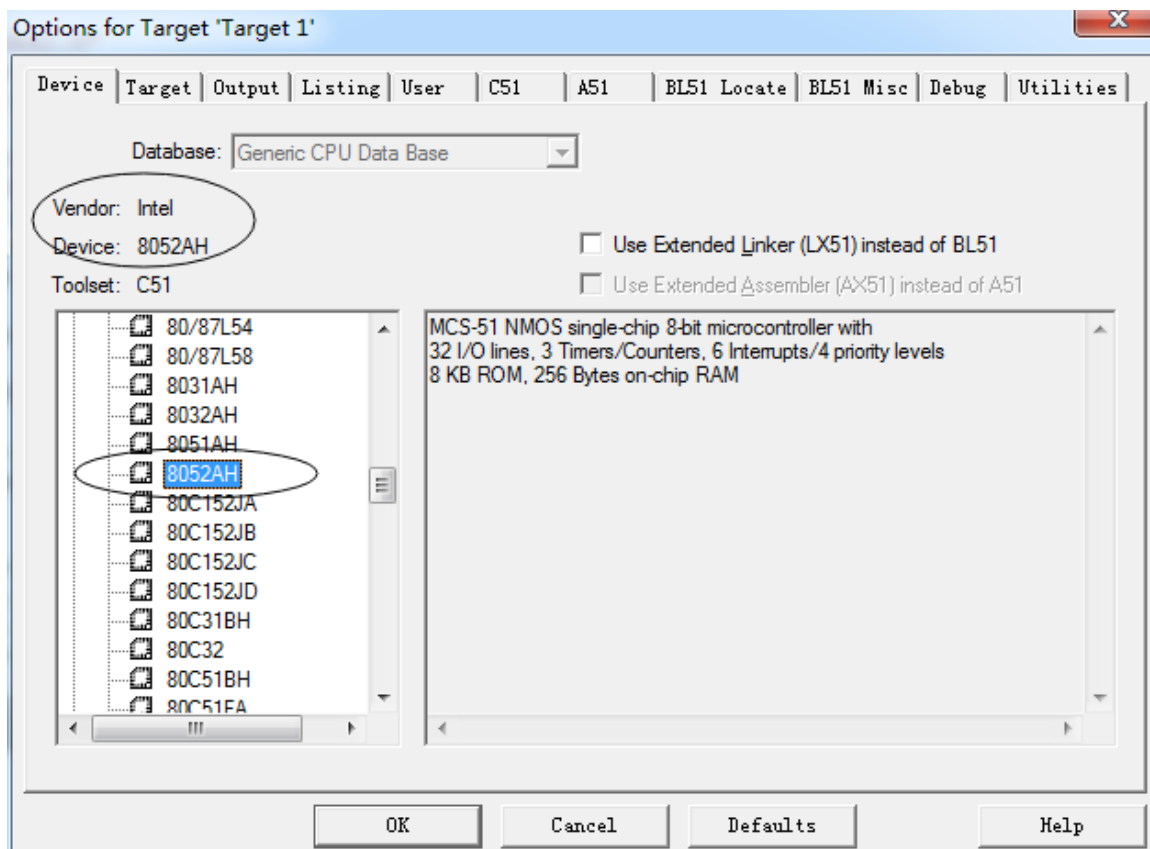
新建工程页面

- (2) 选择路径，文件名及类型并保存，如下图



为工程命名并保存在相应的目录下（注意保存的路径，对以后建立的其他源文件或头文件，尽量保存在该目录下）。

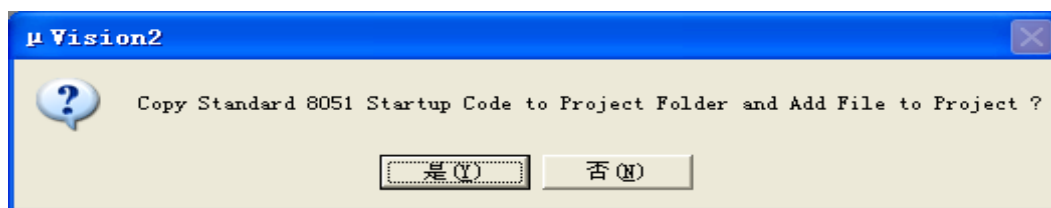
(3) 点击保存后进入器件和芯片选择框，我们选择 INTEL 公司提供的 8052，如下图：



器件和芯片选择界面

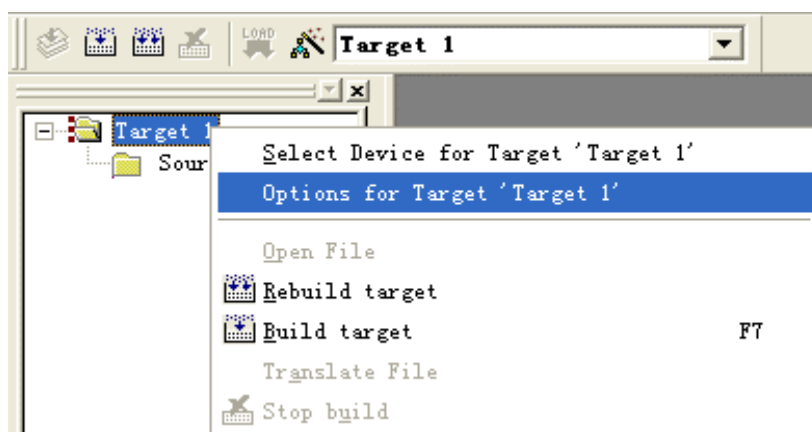
左边是我们具体选择的芯片，右边 Description 栏给出所选择的芯片的具体参数，如内核、端口数、定时器、内存大小等，以方便使用。

(4) 点击确定将弹出如下对话框，表示是否将系统提供的该芯片的初始化代码拷贝在该工程目录下，由于 KEIL 连接器能够自动识别目录下的头文件，所以我们最好选择“否 (NO)”进入下一步的设置。

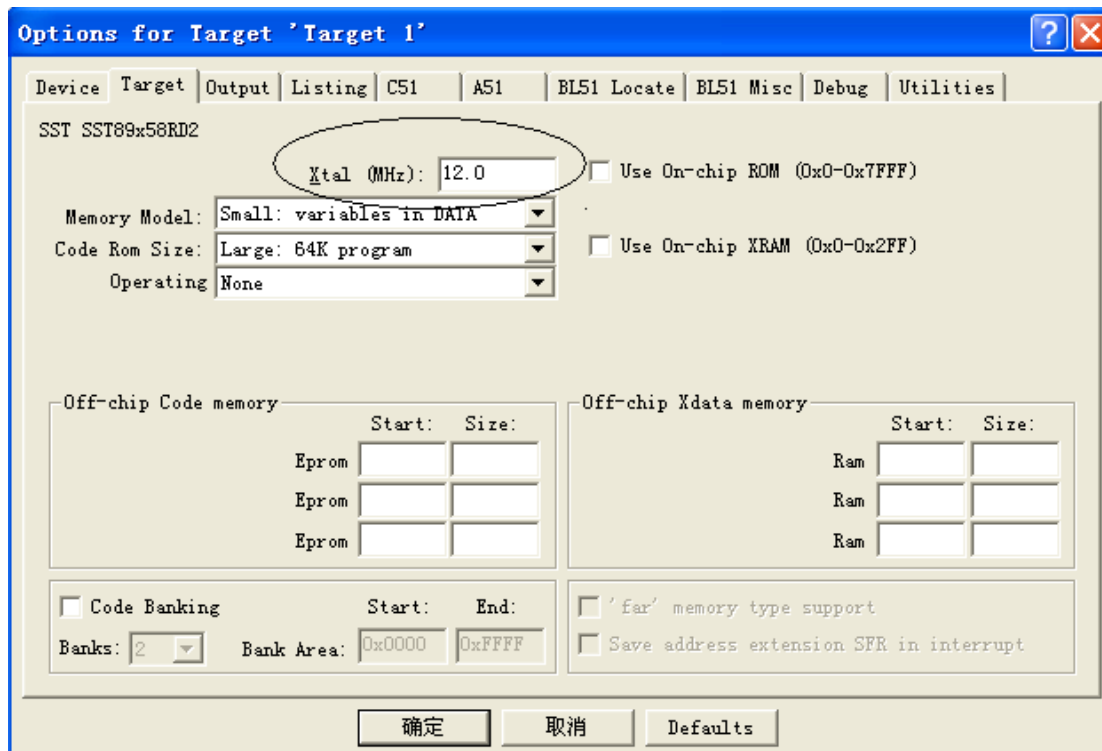


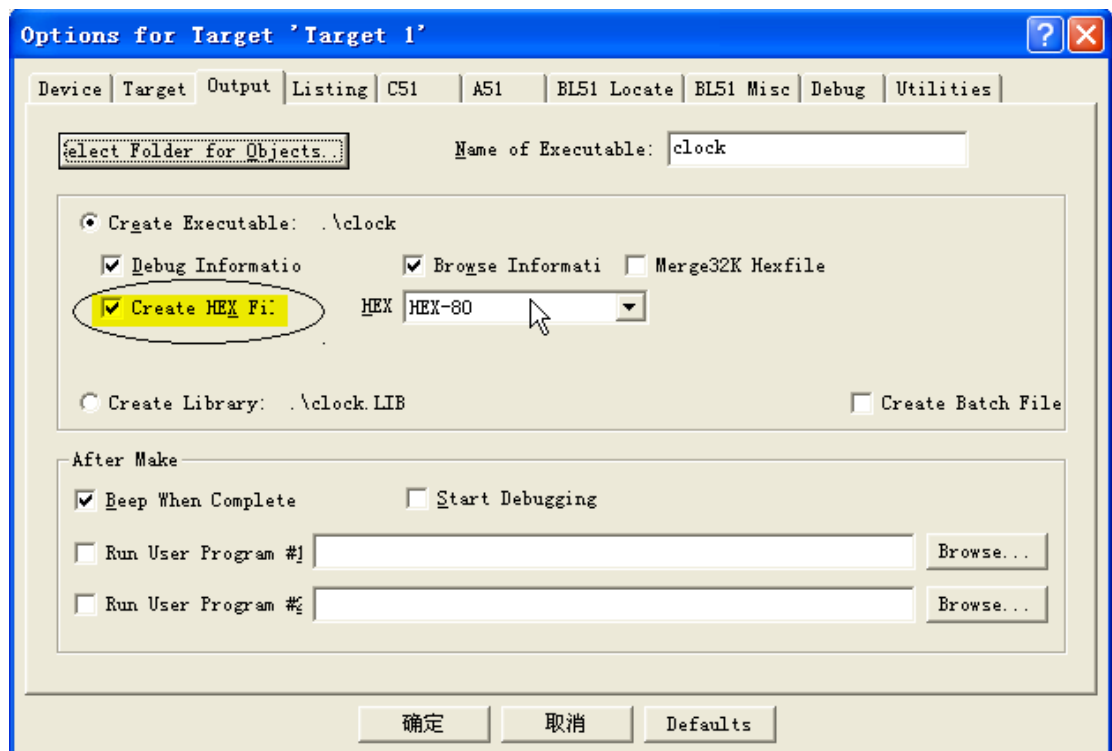
如果我们只利用 KEIL 来编译源程序并进行软件的调试或仿真，我们的工程就建立完毕，但如果需要将源文件编译后下载到实验板上运行，则还需要将工程编译完成后输出一个可执行的文件，因此还得对工程做如下设置。

(5) 在编译器的左边的“PROJECT WORKSPACE”中选择 TARGET1，点击右键，选择“Options for Target 1”进入对象的相关设置，如下图



(6) 单击“Options for Target 1”后，进入 TARGET 设置对话框。如下图

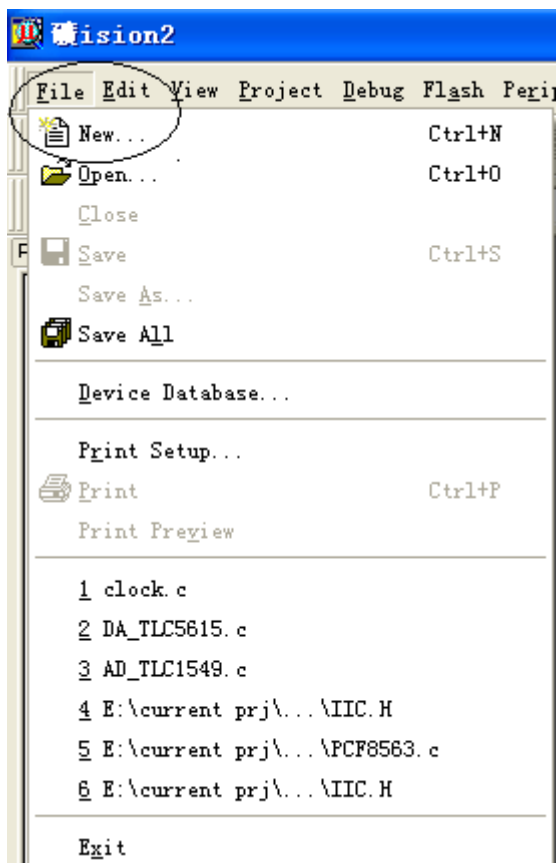




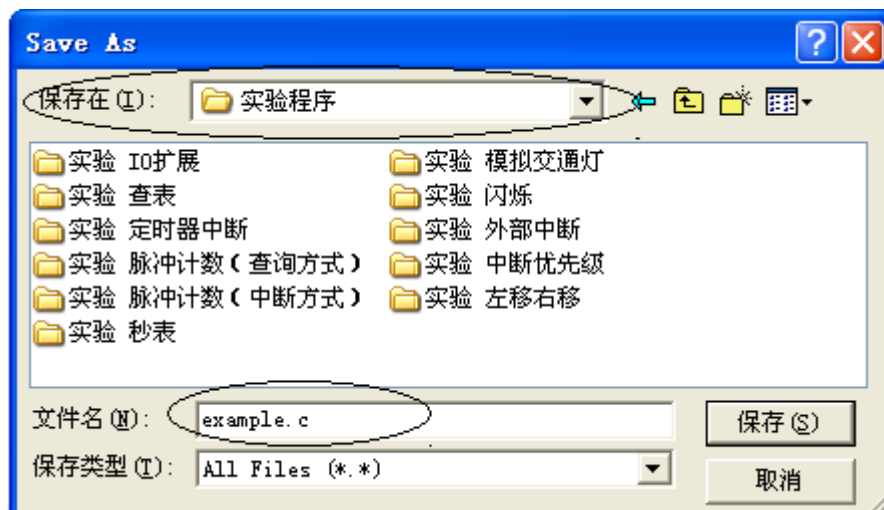
在 Target 选项中，设置 Xtal 为 12MHZ；在 OUTPUT 选项中，选中 Creat HEX 复选框，表示程序编译完成后生成可下载执行的 16 进制文件。生成的文件自动保存在工程目录下，点击确定，完成工程的建立。

2.3.2 建立源程序

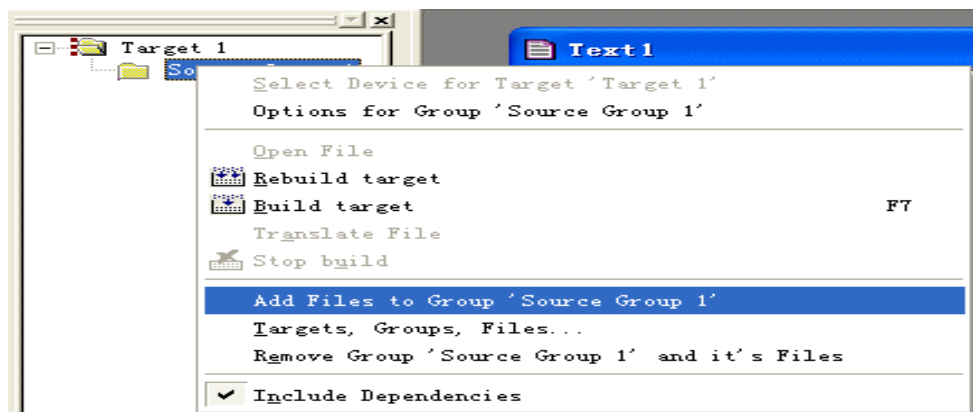
(1) 工程文件创建好后，要为工程建立一个源文件。可以用菜单选项 File-New 来新建一个源文件，也可以直接单击“New Text File ”图标来完成，如下图



(2) 文本文件建立后，就可以直接在文本文件中编辑程序了，但建议在编辑源程序以前，最好先将文件保存为特定的格式，如 C 程序保存为 example.c 文件，汇编程序保存为 example.asm 文件，如下图所示



(3) 文件保存后，还需要将该源文件添加到工程下，方法如下

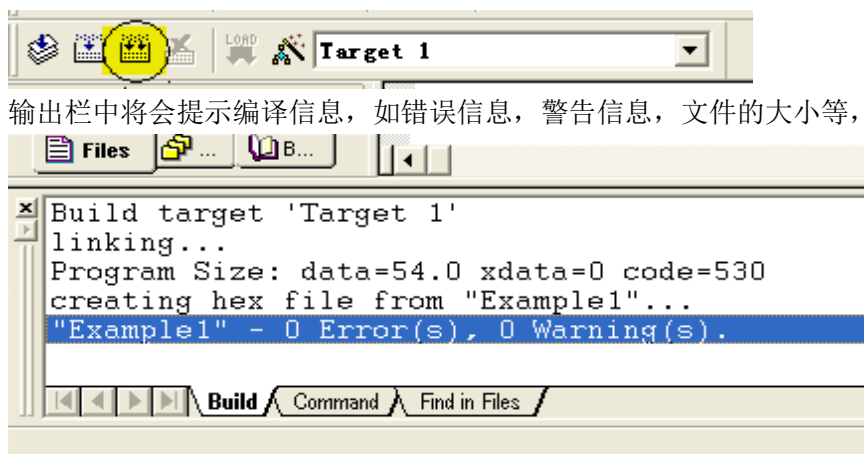


文件添加到工程中

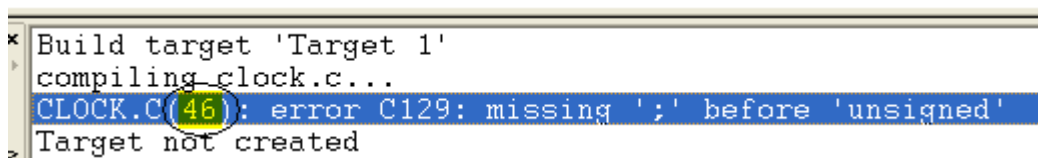
按照如上方法可以将所需要的文件添加到相关的工程文件中，例如，你可以单击 Add Files to Group ‘Source Group 1’，Add Files 选项打开一个标准的文件对话框，从对话框中选择你刚刚生成的文件 example.c。

2.3.3 程序编译

设置好工程后，即可进行编译、连接。选择菜单 Project->Build target 或者直接按 F7，或者编译按钮编译，如下图：



如果程序有语法错误，提醒信息如下图，包括出错的行数及错误描述：



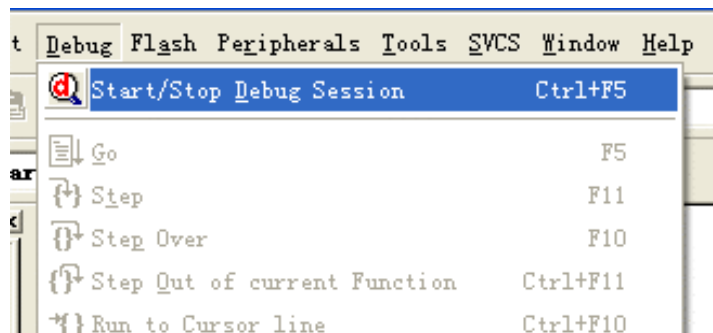
必须要把所有错误改正，而警告则可视情况而定；

2.3.4 程序软件模拟调试

(1) 如果程序没有错误，编译器同时将生成 16 进制文件。我们可以先用软件来调试程序的运行情况，先在 Options for Target ‘Target 1’ 的对话框中的 Debug 页中选中 Use Simulator，如下图所示：



(2) 选择 Debug 菜单的 Start/Stop Debug session, 如下图



Debug 选择菜单

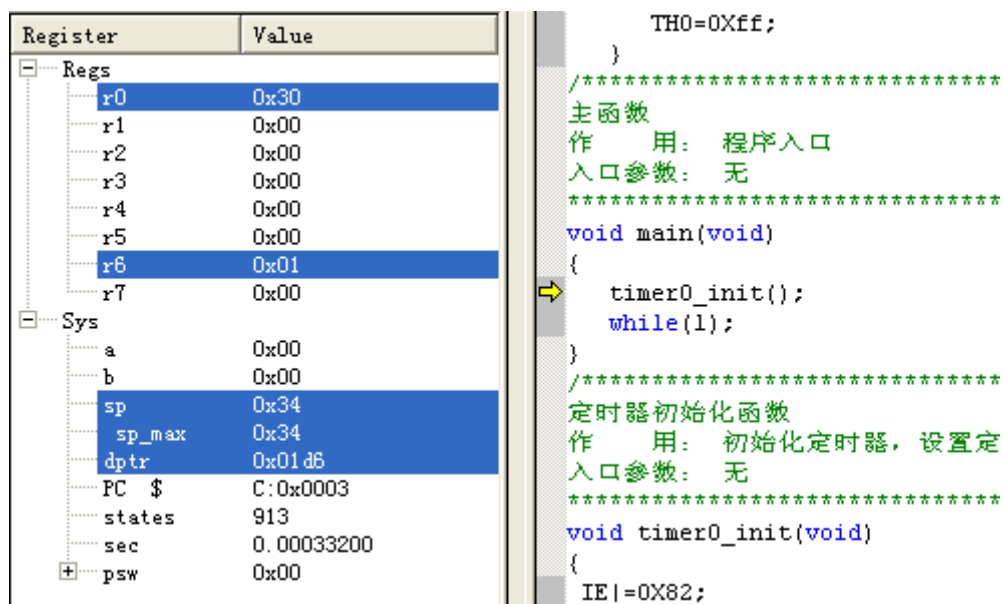
进入 Debug 后, 我们可以选择多种调试方式, 如断点, 单步执行, 跟踪等, 甚至可以直接打开与源程序对应的机器代码, 在程序运行过程中, 我们同样可以查看 CPU 的内部资源情况, 以及寄存器数据变化情况等。

(3) 在程序行设置或移出断点的方法是将光标定位于需要设置的断点的程序行:

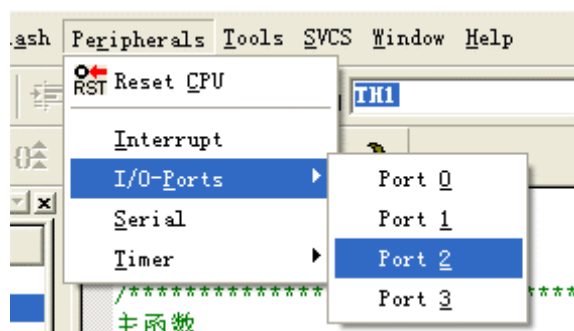
菜单 DEBUG->INSERT/REMOVE BREAKPOINT 设置或移出断点 (也可以用鼠标在该行双击实现断点的设置);

DEBUG->ENABLE/DISABLE BREAKPOINT 是开启或暂停光标所在行的断点的功能;

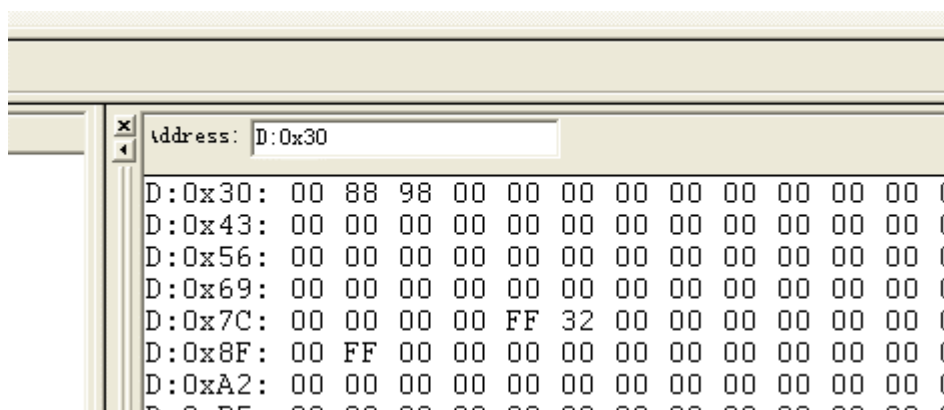
DEBUG->DISABLE ALL BREAKPOINT 暂停所有的断点; DEBUG->KILL BREAKPOINT 是清除所有的断点设置。



我们可以具体选择内部资源的某个模块进行观察，方便程序的调试，下图是资源查看选择方法。



从上图可以看出，在调试中，我们可以选择内部具体的资源情况进行查看，如中断，I/O 端口，串口，定时器等，显示所选器件的具体信息。下图为所选存储器的观察界面



存储器界面可以仿真内部存储器的内容。在地址框中输入需要观察的存储器地址，界面中将显示以该地址为起始地址的存储器内容。地址前面的前缀代表不同的显示内容。如果在输入的地址前面加前缀“D:”表示观察内部数据存储区内容；加前缀“C:”表示观察内部程序存储区内容；加前缀“I:”表示观察立即地址的内容。

我们也可以通过 VIEW 菜单项选择机器代码窗口，汇编代码窗口（C 源程序），程序性能分析窗口等。有了这些友好的界面，我们在程序调试过程中，就可以及时掌握程序的运行

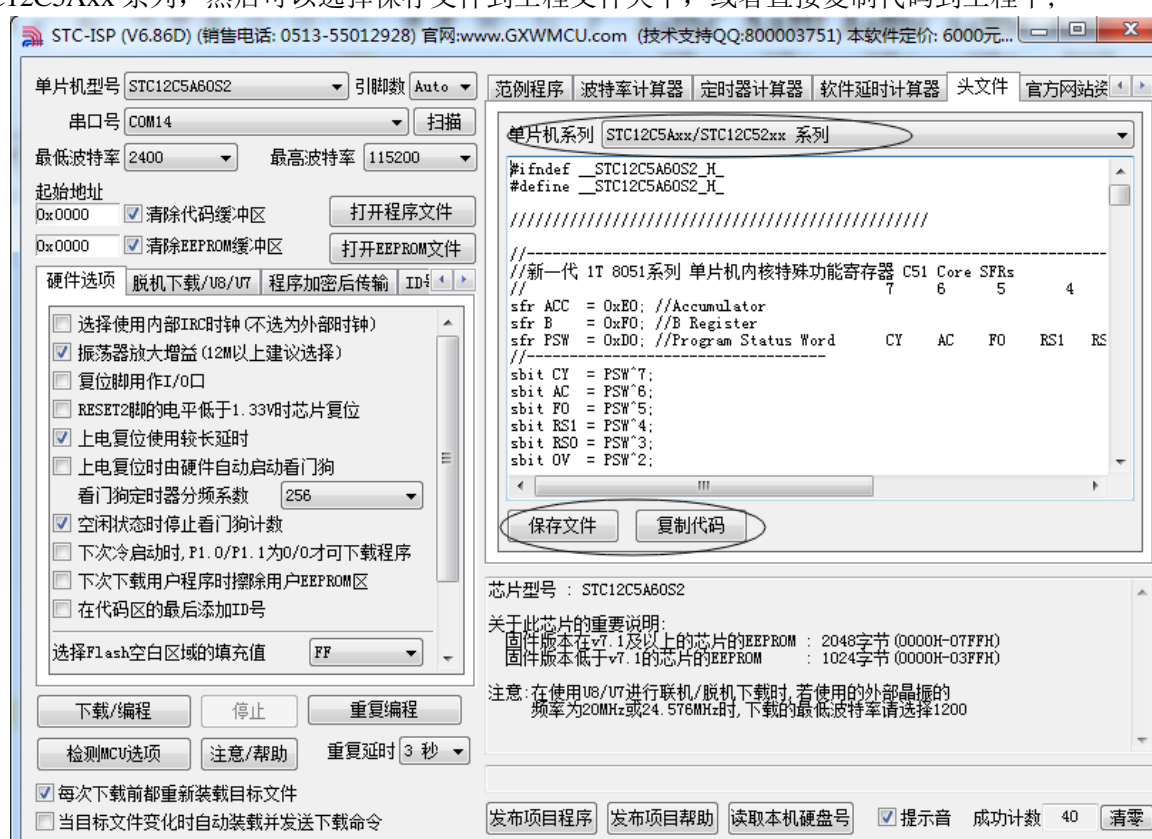
情况。

2.4 用户程序的下载

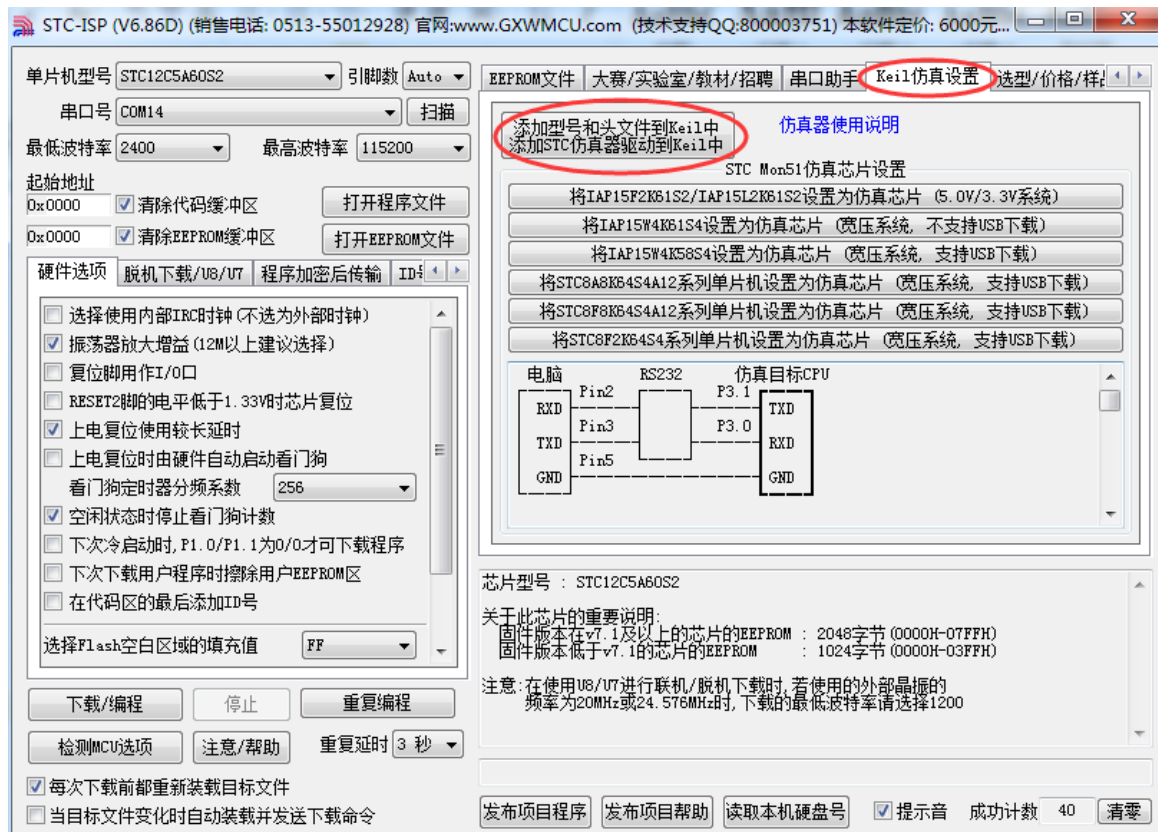
本实验设备使用 STC 公司提供的 ISP 软件“stc-isp-15xx-v6.86D.exe”进行程序的下载烧写，下载软件的使用可参考《STC 实验板资料》中的《四、烧写软件的使用.mp4》，当然在使用烧写软件之前，需要安装驱动程序，具体可参考《STC 实验板资料》中的《三、驱动软件的安装（CH340-USB 驱动）.mp4》。

2.5 关于 STC 的头文件说明

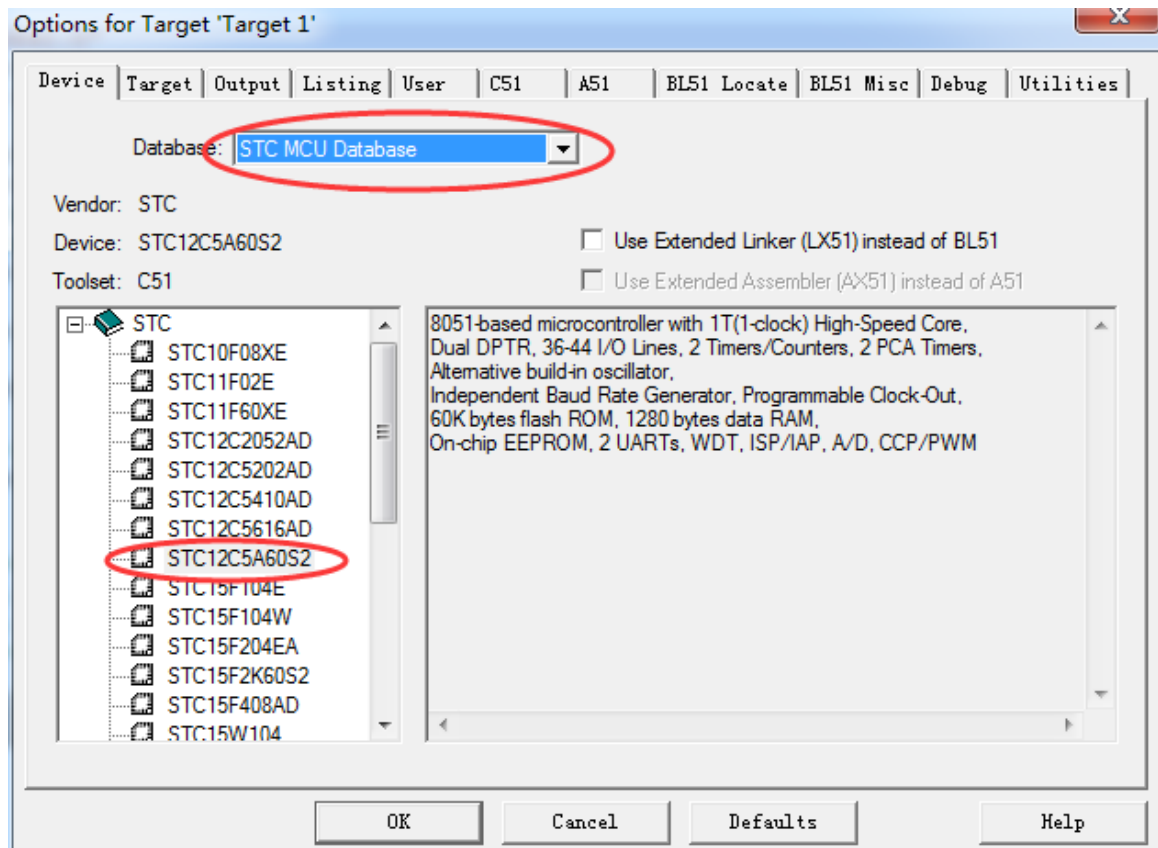
由于 keil 软件的器件列表中没有收集 STC 的产品，如果需要用到 STC 单片机的特殊功能（有别于通用 51 单片机的功能），则需要新的头文件（特殊功能对应的特殊寄存器的定义说明），有两类办法：其一，在工程所选的器件仍用 Intel 的 8052，但需要在源程序中工人替换掉原工程使用的 reg52.h，改为 STC 单片机型号对应的头文件，该头文件可以由烧写软件直接产生，如下图所示，实验板使用的型号是 STC12C5A60S2，故单片机系列选择 STC12C5Axx 系列，然后可以选择保存文件到工程文件夹中，或者直接复制代码到工程中；



另一选择（推荐），就是把 STC 单片机产品添加到 KEIL 软件中，然后在工程的器件列表中就可以直接选择所用的 STC 单片机型号（本板使用的型号是 STC12C5A60S2），具体如下：（1）首先把 STC 单片机产品添加到 KEIL 软件中，如下图，选择“keil 仿真设置”，然后点击“添加型号和头文件到 keil 中”，然后找到安装目录按要求执行即可：



添加完毕后, 再打开 KEIL 软件及工程, 如下图, 就可以选择对应的型号:



这样, 就可以在安装目录下 (例如 C:\Keil_v4\C51\INC\STC), 可以看到已经自动添加了头文件了。

第三章 软件实验

3.1 软件实验一 清零程序

一、实验目的：

1. 掌握汇编语言的设计，以及KEIL集成开发环境的使用，特别是程序调试方法的熟悉；

二、实验设备：

1. 硬件：PC 机 一台 ,实验开发板 一套，跳线若干；
2. 软件：Win7系统，KEIL集成开发环境；

三、实验内容：

把7000~70FFH的内容清零；

四、电路原理图及连接：

软件模拟调试，无需硬件连接；

五、预习要求：

1. 学习参考单片机方面的教程，掌握汇编语言的设计；

六、实验步骤：

1. 启动 keil 集成开发环境，建立工程，编写源程序；
2. 编译调试程序，验证功能是否正确实现；

七、参考程序流程：

略

八、思考题：

1. 把 7000~70FFH 的内容改为 FF；
2. 把 7000~710FH 的内容清零；

3.2 软件实验二 拆字程序

一、实验目的：

1. 掌握汇编语言的设计，以及KEIL集成开发环境的使用，特别是程序调试方法的熟悉；

二、实验设备：

1. 硬件：PC 机 一台，实验开发板 一套，跳线若干；
2. 软件：Win7系统，KEIL集成开发环境；

三、实验内容：

把7000H的内容按16进制数拆开，高4位送7001H低位，低4位送7002H低位，7001H7002H高位清零，一般本程序用于把数据送显示器缓冲使用。

四、电路原理图及连接：

软件模拟调试，无需硬件连接；

五、预习要求：

1. 学习参考单片机方面的教程，掌握汇编语言的设计；

六、实验步骤：

1. 启动 keil 集成开发环境，建立工程，编写源程序；
2. 编译调试程序，验证功能是否正确实现；

七、参考程序流程：

略

八、思考题：

1. 把 7000H 的内容按十进制数拆开，个位数送 7001H 低位，十位数送 7002H 低位，百位数送 7003H 低位；

3.3 软件实验三 拼字程序

一、实验目的：

1. 掌握汇编语言的设计，以及KEIL集成开发环境的使用，特别是程序调试方法的熟悉；

二、实验设备：

1. 硬件：PC 机 一台，实验开发板 一套，跳线若干；
2. 软件：Win7系统，KEIL集成开发环境；

三、实验内容：

把 7000H.7001H 的低位相拼后送入 7002H 中，一般本程序用于把显示缓冲区的数据取出拼装成一个字节。

四、电路原理图及连接：

软件模拟调试，无需硬件连接；

五、预习要求：

1. 学习参考单片机方面的教程，掌握汇编语言的设计；

六、实验步骤：

1. 启动 keil 集成开发环境，建立工程，编写源程序；
2. 编译调试程序，验证功能是否正确实现；

七、参考程序流程：

略

八、思考题：

1. 把 7000H(个位).7001H(十位)相拼后送入 7002H 中，一般本程序用于把显示缓冲区的数据取出拼装成一个字节；

3.4 软件实验四 数据区传送子程序

一、实验目的：

1. 掌握汇编语言的设计，以及KEIL集成开发环境的使用，特别是程序调试方法的熟悉；

二、实验设备：

1. 硬件：PC 机 一台，实验开发板 一套，跳线若干；
2. 软件：Win7系统，KEIL集成开发环境；

三、实验内容：

把内部 RAM 区以 R3 为源首址的 R7 个字节数据，传送到以 R5 为目的起始地址内部 RAM 区。

四、电路原理图及连接：

软件模拟调试，无需硬件连接；

五、预习要求：

1. 学习参考单片机方面的教程，掌握汇编语言的设计；

六、实验步骤：

1. 启动 keil 集成开发环境，建立工程，编写源程序；
2. 编译调试程序，验证功能是否正确实现；

七、参考程序流程：

略

八、思考题：

1. 把外部 RAM 区以 (R2. R3) 为源首址内的 (R6. R7) 个字节数据，传送到以 (R4. R5) 为目的起始地址外部 RAM 区。

3.5 软件实验五 数据排序

一、实验目的：

1. 掌握汇编语言的设计，以及KEIL集成开发环境的使用，特别是程序调试方法的熟悉；

二、实验设备：

1. 硬件：PC 机 一台，实验开发板 一套，跳线若干；
2. 软件：Win7系统，KEIL集成开发环境；

三、实验内容：

编写并调试一个排序程序，其功能为用冒泡法将内部 RAM 中几个单字节无符号整数按从小到大的次序重新排列。

四、电路原理图及连接：

软件模拟调试，无需硬件连接；

五、预习要求：

1. 学习参考单片机方面的教程，掌握汇编语言的设计；

六、实验步骤：

1. 启动 keil 集成开发环境，建立工程，编写源程序；
2. 编译调试程序，验证功能是否正确实现；

七、参考程序流程：

略

八、思考题：

1. 用冒泡法将内部 RAM 中几个单字节无符号整数，按从大到小的次序重新排列。

3.6 软件实验六 数据查找

一、实验目的：

1. 掌握汇编语言的设计，以及KEIL集成开发环境的使用，特别是程序调试方法的熟悉；

二、实验设备：

1. 硬件：PC 机 一台，实验开发板 一套，跳线若干；
2. 软件：Win7系统，KEIL集成开发环境；

三、实验内容：

在 7000H-7000FH 中查出有几个字节是零，统计“00”的个数。

四、电路原理图及连接：

软件模拟调试，无需硬件连接；

五、预习要求：

1. 学习参考单片机方面的教程，掌握汇编语言的设计；

六、实验步骤：

1. 启动 keil 集成开发环境，建立工程，编写源程序；
2. 编译调试程序，验证功能是否正确实现；

七、参考程序流程：

略

八、思考题：

1. 修改程序，查找其它内容，例如某个字符串。

3.7 软件实验七 多分支程序

一、实验目的：

1. 掌握汇编语言的设计，以及KEIL集成开发环境的使用，特别是程序调试方法的熟悉；

二、实验设备：

1. 硬件：PC 机 一台，实验开发板 一套，跳线若干；
2. 软件：Win7系统，KEIL集成开发环境；

三、实验内容：

编写**散转程序**，它根据不同的输入条件或运算结果，直接转向不同的处理子程序。

四、电路原理图及连接：

软件模拟调试，无需硬件连接；

五、预习要求：

1. 学习参考单片机方面的教程，掌握汇编语言的设计；

六、实验步骤：

1. 启动 keil 集成开发环境，建立工程，编写源程序；
2. 编译调试程序，验证功能是否正确实现；

七、参考程序流程：

略

八、思考题：

无

第四章 硬件实验

4.1 硬件实验一 跑马灯

一、实验目的：

1. 掌握51单片机IO的简单使用；
2. 掌握软件延时的方法；

二、实验设备：

1. 硬件：PC 机 一台，实验开发板 一套，跳线若干；
2. 软件：Win7系统，KEIL集成开发环境；

三、实验内容：

编写程序，实现流水灯的效果：使8个小灯自上而下依次点亮，闪烁间隔为500ms。

四、电路原理图及连接：

请根据自己的设计进行不同的跳线连接，具体电路请参考原理图；

五、预习要求：

1. 学习参考单片机方面的教程，掌握单片机的 I/O 输入输出的控制方式；
2. 熟悉实验开发板的硬件结构和原理，学习跳线连接器的使用；
3. 学习 LED 驱动原理及软件延时的方法；

六、实验步骤：

1. 确保 P1 口与 LED 引脚的正确连接；
2. 启动 keil 集成开发环境，建立工程，编写源程序；
3. 编译调试程序，验证功能是否正确实现；

七、参考程序流程：

略

八、思考题：

1. 控制 8 个 LED 灯不停地闪烁，闪烁间隔为 500ms；
2. 控制小灯从上下两端向中间依次点亮（每个时刻点亮两盏灯），然后再由中间向两端依次点亮，循环往复。
3. 用按键 S1 控制在两种不同的显示风格（比如实验内容中普通的流水灯效果和思考题 2 所定制的显示风格）中切换。

4.2 硬件实验二 外部中断

一、实验目的：

1. 了解单片机外部中断的原理，功能，掌握外部中断的使用方法；

二、实验设备：

1. 硬件：PC 机 一台，实验开发板 一套，跳线若干；
2. 软件：Win7系统，KEIL集成开发环境；

三、实验内容：

编写程序，利用LED灯亮暗表示按键按下次数（8位二进制数），即按一次加一；

四、电路原理图及连接：

请根据自己的设计进行不同的跳线连接，具体电路请参考原理图；

五、预习要求：

1. 学习参考单片机方面的教程，掌握单片机外部中断的原理，功能以及使用方法；
2. 熟悉实验开发板的硬件结构和原理；

六、实验步骤：

1. 确保 P1 口与 LED 引脚的正确连接，以及按键和外部中断口的连接；
2. 启动 keil 集成开发环境，建立工程，编写源程序；
3. 编译调试程序，验证功能是否正确实现；

七、参考程序流程：

略

八、思考题：

1. 用两个按键同时用两个外部中断，一加一减；
2. 当按键按下连续进行加（或减），直到按键松开；
3. 在思考题 1 的基础上，同时用数码管来显示按键次数（0-255）。为了用户界面直观友好，要求数码管以十进制形式显示。实现时注意两种显示方式（LED 灯的二进制形式和数码管显示的十进制形式）之间的**同步性**。
4. 仍用单个字节表示按键次数，只不过改用 BCD 码来表示，因此有效表示范围是 0-99，其余要求同思考题 3；
5. 将教材第 104 页的例程 6-2 的第 2 版代码**移植**到开发板上（注意电路的差异），另外要求同时呈现流水灯的效果。**实现提示**：显示流水灯的任务与指定的单个数码管上显示按键次数的任务相互独立。

4.3 硬件实验三 定时器

一、实验目的：

1. 了解单片机定时器的原理，功能，掌握定时器的使用方法；

二、实验设备：

1. 硬件：PC 机 一台，实验开发板 一套，跳线若干；
2. 软件：Win7系统，KEIL集成开发环境；

三、实验内容：

编写程序，通过定时器（查询方式）控制小灯每隔1秒左移一次；

四、电路原理图及连接：

请根据自己的设计进行不同的跳线连接，具体电路请参考原理图；

五、预习要求：

1. 学习参考单片机方面的教程，掌握单片机定时器的原理，功能以及使用方法；
2. 熟悉实验开发板的硬件结构和原理；

六、实验步骤：

1. 确保 P1 口与 LED 引脚的正确连接；
2. 启动 keil 集成开发环境，建立工程，编写源程序；
3. 编译调试程序，验证功能是否正确实现；

七、参考程序流程：

略

八、思考题：

1. 用中断方式实现；
2. 实现**秒表**功能，包括**复位**操作（时间清零）、**启动/暂停/继续**计时。由于开发板上只有四个数码管，因此需求分析中可以规定计时的有效范围为 00.00 – 99.99 秒，计时精度为 0.01 秒。最简性原则要求按键只用两个：一个用于复位清零；另一个用于启动/暂停/继续计时。**特别注意**如下**业务逻辑**：在计时的过程中不允许进行复位清零操作，只有当暂停计时后才允许复位清零；另外，在启动计时之前也不允许复位清零；最后，一旦计时溢出时，秒表停止工作，计时显示保持 99.99（秒）不变，禁用启动/暂停/继续按键，只有复位清零后才能重新计时。

4.4 硬件实验四 脉冲计数

一、实验目的：

1. 了解单片机I/O口扩展的原理，掌握8155芯片的使用；

二、实验设备：

1. 硬件：PC 机 一台，实验开发板 一套，跳线若干；
2. 软件：Win7系统，KEIL集成开发环境；

三、实验内容：

编写程序，把定时器0外部输入的脉冲进行计数，并送数码管显示；（外部）

四、电路原理图及连接：

请根据自己的设计进行不同的跳线连接，具体电路请参考原理图；

五、预习要求：

1. 学习参考单片机方面的教程，掌握单片机计数器原理及使用方法；
2. 熟悉实验开发板的硬件结构和原理；

六、实验步骤：

1. 确保各连线正确连接；
2. 启动 keil 集成开发环境，建立工程，编写源程序；
3. 编译调试程序，验证功能是否正确实现；

七、参考程序流程：

略

八、思考题：

1. 计算脉冲频率，并在数码管上显示；

4.5 硬件实验五 串口通信

一、实验目的：

1. 了解串口通信原理，掌握串口通信的应用设计方法；

二、实验设备：

1. 硬件：PC 机 一台，实验开发板 一套，跳线若干；
2. 软件：Win7系统，KEIL集成开发环境；

三、实验内容：

- 1、单片机通过串口接收数据（中断方式），点亮对应的LED灯。
- 2、串口数据格式如下：

帧头	长度	帧数据	校验字	结束符
2 字节	1 字节			

- (1) 帧头2字节：为固定两个字节，例如0xFA,0xAF，可自定义；
 - (2) 长度：帧数据的字节数
 - (3) 帧数据：即实际有用数据；
 - (4) 校验字：帧头、长度、帧数据所有字节累加和的低8位；
 - (5) 结束符：可选，可自定义
- 3、举例：在PC的串口调试助手（请自行下载）发送以下数据：
 - (1) FA AF 01 01 AB 则点亮第1个LED灯
 - (2) FA AF 01 03 AD 则点亮第1，2个LED灯

四、电路原理图及连接：

请根据自己的设计进行不同的跳线连接，具体电路请参考原理图；

五、预习要求：

1. 学习参考单片机方面的教程，掌握串口通信的应用方法；
2. 熟悉实验开发板的硬件结构和原理；

六、实验步骤：

1. 确保各连线正确连接；
2. 启动 keil 集成开发环境，建立工程，编写源程序；
3. 编译调试程序，验证功能是否正确实现；

七、参考程序流程：

略

八、思考题：

1. 在基本要求的功能基础上增加：单片机接收到正确数据，则单片机通过串口回送一串字符（表示接收到正确数据）；如果数据错误则回送另一串字符（表示接收到正确有误）。
2. 如果接收数据有误，则回送不同字符表示错误的原因；
3. 串口发送采用中断方式。

4.6 硬件实验六 AD 转换

一、实验目的：

1. 了解AD的原理，掌握STC单片机集成AD的使用；

二、实验设备：

1. 硬件：PC 机 一台，实验开发板 一套，跳线若干；
2. 软件：Windows7系统，KEIL集成开发环境；

三、实验内容：

通过STC单片机集成AD采集电压数据并在数码管上显示；

四、电路原理图及连接：

请根据自己的设计进行不同的跳线连接，具体电路请参考原理图；

五、预习要求：

1. 学习参考单片机方面的教程，掌握 AD 原理以及 STC 单片机集成 AD 的使用方法；
2. 熟悉实验开发板的硬件结构和原理；

六、实验步骤：

1. 确保各连线正确连接；
2. 启动 keil 集成开发环境，建立工程，编写源程序；
3. 编译调试程序，验证功能是否正确实现；

七、参考程序流程：

略

八、思考题：

1. 在数码管上直接显示出电压值；
2. 尝试不同的数字滤波方法；

4.7 硬件实验七 LCD 显示

一、实验目的：

1. 了解LCD的相关知识，掌握字符显示模块JM1602C的使用；

二、实验设备：

1. 硬件：PC 机 一台，实验开发板 一套，跳线若干；
2. 软件：Win7系统，KEIL集成开发环境；

三、实验内容：

在LCD上显示秒钟；

四、电路原理图及连接：

请根据自己的设计进行不同的跳线连接，具体电路请参考原理图；

五、预习要求：

1. 学习参考单片机方面的教程，了解 LCD 的相关知识以及 JM1602C 的使用方法；
2. 熟悉实验开发板的硬件结构和原理；

六、实验步骤：

1. 确保各连线正确连接；
2. 启动 keil 集成开发环境，建立工程，编写源程序；
3. 编译调试程序，验证功能是否正确实现；

七、参考程序流程：

略

八、思考题：

1. 实现显示模块中的光标功能；

4.8 硬件实验八 乐曲播放

一、实验目的：

1. 了解单片机控制蜂鸣器播放音乐的原理；

二、实验设备：

1. 硬件：PC 机 一台，实验开发板 一套，跳线若干；
2. 软件：Win7系统，KEIL集成开发环境；

三、实验内容：

实现单片机控制蜂鸣器进行播放音乐；

四、电路原理图及连接：

请根据自己的设计进行不同的跳线连接，具体电路请参考原理图；

五、预习要求：

1. 学习参考单片机方面的教程，了解单片机控制蜂鸣器播放音乐的原理；
2. 熟悉实验开发板的硬件结构和原理；

六、实验步骤：

1. 确保各连线正确连接；
2. 启动 keil 集成开发环境，建立工程，编写源程序；
3. 编译调试程序，验证功能是否正确实现；

七、参考程序流程：

略

八、思考题：

1. 通过键盘进行多首乐曲的切换播放；

第五章 综合性实验

5.1 综合性实验一 秒表

一、实验目的：

1. 能熟练运用单片机构建中小规模的应用系统，用C51可进行中等规模程序的编写；

二、实验设备：

1. 硬件：PC 机 一台 ,实验开发板 一套，跳线若干；
2. 软件：Win7系统，KEIL集成开发环境；

三、实验内容：

实现秒表功能，包括，启动计时，暂停计时，继续计时，时间清零等；要求，显示计时最小单位0.1秒，最大为999.9，使用按键数量最多不超过3个；

四、电路原理图及连接：

请根据自己的设计进行不同的跳线连接，具体电路请参考原理图；

五、预习要求：

1. 学习参考单片机方面的教程，了解单片机应用系统的构建过程；
2. 熟悉实验开发板的硬件结构和原理；

六、实验步骤：

1. 确保各连线正确连接；
2. 启动 keil 集成开发环境，建立工程，编写源程序；
3. 编译调试程序，验证功能是否正确实现；

七、参考程序流程：

略

八、思考题：

略；

5.2 综合性实验二 电子钟

一、实验目的：

1. 能熟练运用单片机构建中小规模的应用系统，用C51可进行中等规模程序的编写；

二、实验设备：

1. 硬件：PC 机 一台，实验开发板 一套，跳线若干；
2. 软件：Win7系统，KEIL集成开发环境；

三、实验内容：

设计一电子钟，以秒，分，时为单位进行计时，并通过数码管显示（由于只有四位，只显示分，秒时间数）；

四、电路原理图及连接：

请根据自己的设计进行不同的跳线连接，具体电路请参考原理图；

五、预习要求：

1. 学习参考单片机方面的教程，了解单片机应用系统的构建过程；
2. 熟悉实验开发板的硬件结构和原理；

六、实验步骤：

1. 确保各连线正确连接；
2. 启动 keil 集成开发环境，建立工程，编写源程序；
3. 编译调试程序，验证功能是否正确实现；

七、参考程序流程：

略

八、思考题：

1. 在数码管上通过按键选择分别显示秒，分，时的数字；
2. 实现闹钟功能，即通过键盘设置闹铃时间，到达该时间后能启动蜂鸣器报警；