

单片机原理及接口技术

实验指导书

(计算机科学与技术专业适用)

计算机科学与技术学院
信息科学与技术实验教学中心
江苏省省级示范实验教学中心
2020 年再版修订

前 言

《单片机原理及接口技术》是一门实践性很强的专业课，其实验是软件与硬件结合。只有通过大量的实验才能掌握教学内容要求的 MCS-51 系列单片机的结构特点和组成原理，做到理论联系实际。在实验中通过对单片机软硬件开发过程所出现的故障而进行的分析和寻找，程序的设计和调试，来培养学生分析问题和解决问题的能力。通过对单片机应用技术的设计，也培养学生综合运用各种知识的实践能力。

结合《单片机原理及接口技术》课程理论教学内容，单片机原理及接口技术实验选用西安唐都科教仪器公司开发的实验教学系统作为实验环境。通过实验可以使具备以下能力：会使用 MCS-51 系列单片机；熟练运用汇编语言进行编程，调试运行程序；初步掌握单片机的一些典型应用来解决一些实际问题。

实验指导书共分为四大部分：概述、单片机原理实验、单片机集成功能模块实验和单片机系统扩展实验。

编者 2020 年 3 月

目录

第 1 章 概 述	1
1.1 SST89E554RC 简介	1
1.2 实验项目	1
1.3 Keil C51 的安装	2
1.4 μ Vision3 集成开发环境	2
第 2 章 单片机原理实验	3
2.1 系统认识实验	3
2.2 数据排序实验	10
第 3 章 单片机集成功能模块实验	12
3.1 中断系统实验	12
3.2 定时/计数器实验	14
第 4 章 单片机系统扩展实验	15
4.1 键盘扫描及显示设计实验	15

第1章 概述

单片机进入我国已20多年了，随着科学技术的发展，计算机技术的普及，单片机已经成为工科院校的一门技术基础课。西安唐都科教仪器公司自成立以来一直致力于单片机教学实验平台的开发，从2002 年开始，陆续推出了以SST89E554RC 单片机为核心的一系列单片机教学实验平台，以满足不同用户的不同需要。

1.1 SST89E554RC 简介

SST89E554RC是SST公司推出的8位51微控制器MCU家族中的一员，具有如下特征：

- 与8051兼容，嵌入Flash 存储器
 - 软件完全兼容
 - 开发工具兼容
 - 引脚全兼容
- 工作电压5V，工作时钟0~40MHz
- 1Kbyte 内部RAM
- 三个16位的定时器/计数器
- 全双工、增强型UART
- 八个中断源，四级优先级
- 可编程看门狗定时器（WDT）
- 标准每周期12个时钟，器件提供选项可使速度倍增，达到每周期6个时钟
- 低功耗模式
 - 掉电模式，可由外部中断唤醒
 - 空闲模式

1.2 实验项目

本实验指导书包含以下单片机实验项目：

1. 单片机原理实验
 - (1) 系统认识实验
 - (5) 数据排序实验
2. 单片机集成功能模块实验
 - (2) 中断系统实验
 - (3) 定时器/计数器实验
3. 单片机系统扩展实验
 - (2) 8255 键盘及数码显示实验

1.3 Keil C51 的安装

Keil C51 μ Vision3 集成开发环境是美国Keil公司开发的基于80C51 内核的微处理器软件开发平台，内嵌多种符合当前工业标准的开发工具，可以完成从工程建立到编译、链接、目标代码生成、软件仿真、硬件仿真等完整的开发流程。

1.3.1 系统要求

安装Keil C51软件，必须满足最小的软、硬件要求，以确保程序功能的正常。

- (1) Pentium、Pentium-II 或兼容处理器的PC;
- (2) Windows98、Windows2000 或Windows XP 操作系统;
- (3) 至少16MB RAM; 至少20MB 硬盘空间。

1.3.2 软件安装

进入Keil C51 软件的Setup 目录下，双击SETUP.EXE 开始安装（略）

1.4 μ Vision3 集成开发环境

MVision3 支持所有的Keil 80C51 的工具软件，包括C51 编译器、宏汇编器、链接器/定位器、软硬件调试器和目标文件到HEX 格式文件转换器等， μ Vision3 可以自动完成编译、汇编、链接程序等操作。

双击桌面上的Keil μ Vision3 快捷图标，可以进入如图1-4-1 所示的集成开发调试环境，各种调试工具、命令菜单都集成在此开发环境中。菜单栏提供了各种操作菜单，如编辑器操作、工程维护、程序调试、窗体选择以及操作帮助等。工具栏按钮和快捷键可以快速执行 μ Vision3命令。

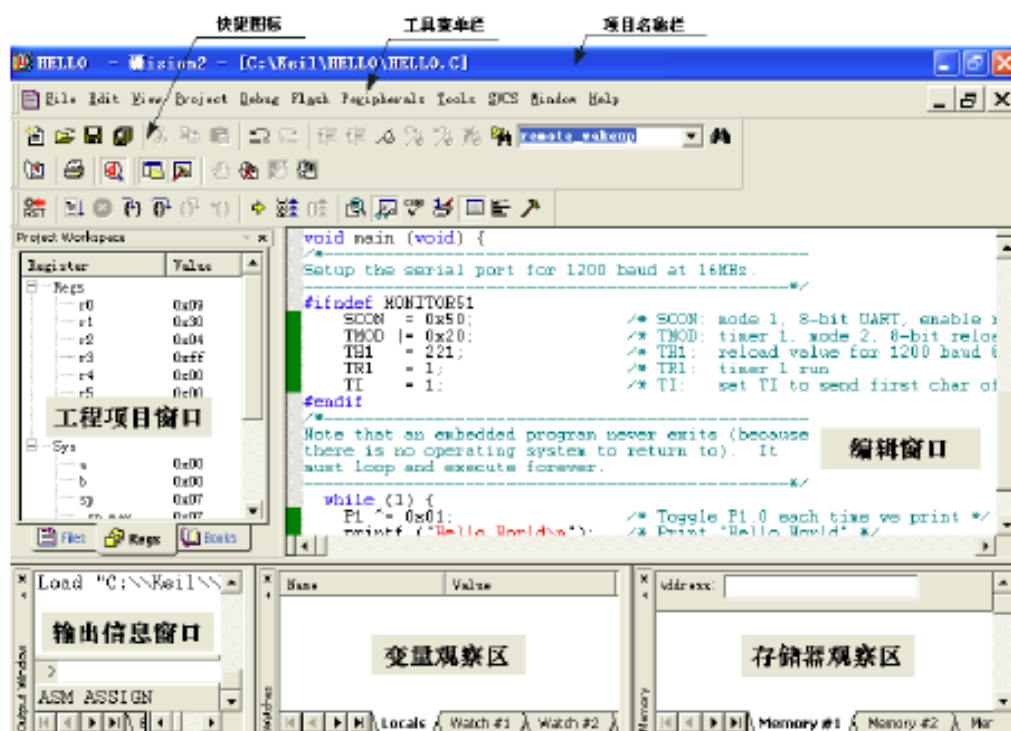


图1-4-1 μ Vision3 集成环境界面

第2章 单片机原理实验

通过本章的实验，旨在使学生掌握Keil C51 的操作方法，学习80C51 的指令系统及汇编语言的程序设计方法。

2.1 系统认识实验

2.1.1 实验目的

1. 学习Keil C51集成开发环境的操作；
2. [熟悉TD-51 系统板的结构及使用。]

2.1.2 实验设备

PC机一台，[TD-PITE 教学实验系统+TD-51 系统平台]

2.1.3 实验内容

编写实验程序，将00H~0FH 共16 个数写入单片机内部RAM 的30H~3FH 空间。

通过本实验，学生需要掌握Keil C51 软件的基本操作，便于后面的学习。

2.1.4 实验步骤

1. 创建Keil C51 应用程序

在Keil C51 集成开发环境下使用工程的方法来管理文件，所有的源文件、头文件甚至说明性文档都可以放在工程项目文件里统一管理。

下面创建一个新的工程文件Asm1.Uv2，以此详细介绍如何创建一个Keil C51 应用程序。

(1) 运行Keil C51 软件，进入Keil C51 集成开发环境。

(2)选择工具栏的**Project** 选项，如图2-1-1 所示，弹出下拉菜单，选择**NewProject** 命令，建立一个新的 μ Vision3 工程。这时会弹出如图2-1-2 所示的工程文件保存对话框，选择工程目录并输入文件名Asm1 后，单击保存。

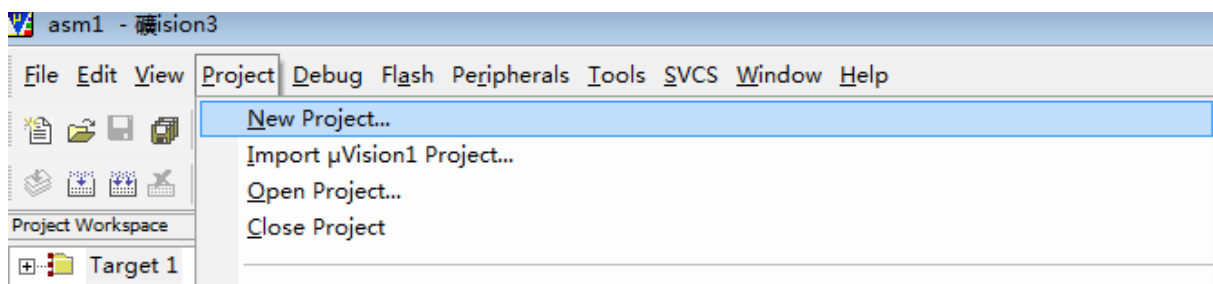


图2-1-1 工程下拉菜单

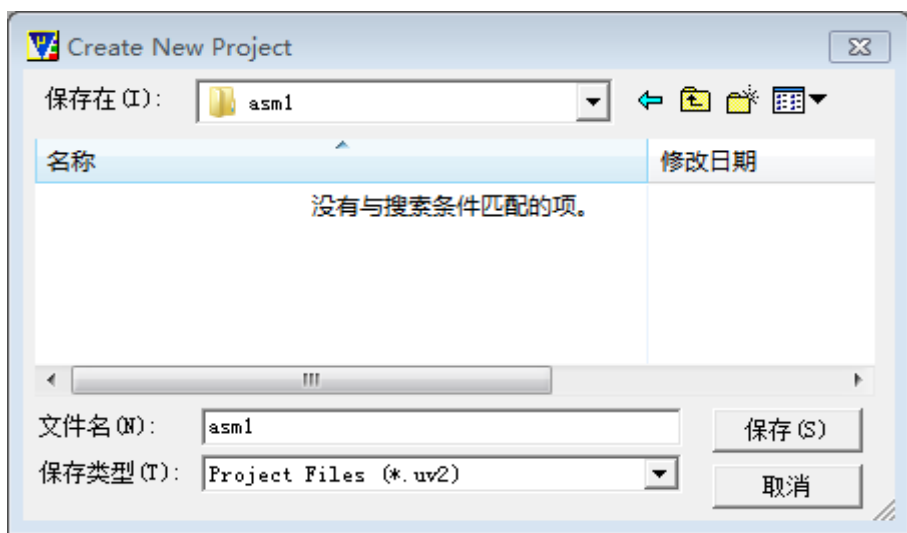


图2-1-2 工程保存对话框

(3) 工程建立完毕后， μ Vision3 会马上弹出如图2-1-3 所示的器件选择窗口。器件选择的目的是告诉 μ Vision3 使用的80C51 芯片的型号是哪一个公司的哪一个型号，不同型号的51 芯片内部资源是不同的。此时选择SST 公司的SST89E554RC。另外，可以选择Project 下拉菜单中的“Select Device for Target ‘Target 1’”命令来弹出图2-1-3 所示的对话框。

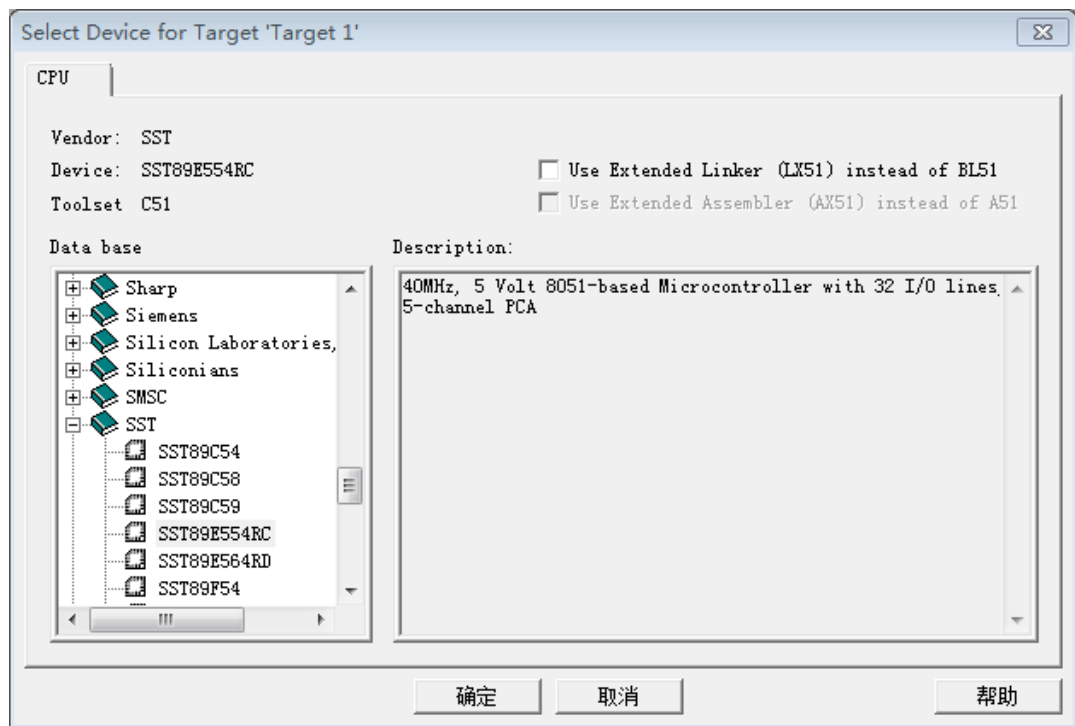



图2-1-3 器件选择对话框

(4) 到此建立好一个空白工程，现在需要人工为工程添加程序文件，如果还没有程序文件则必须建立它。选择工具栏的File 选项，在弹出的下拉菜单中选择New 目录，如图2-1-4 所示，或点击。此时会在文件窗口出现如图2-1-5 所示的新文件窗口 Text1，若多次执行New 命令，则会出现Text2、Text3 等多个新文件窗口。

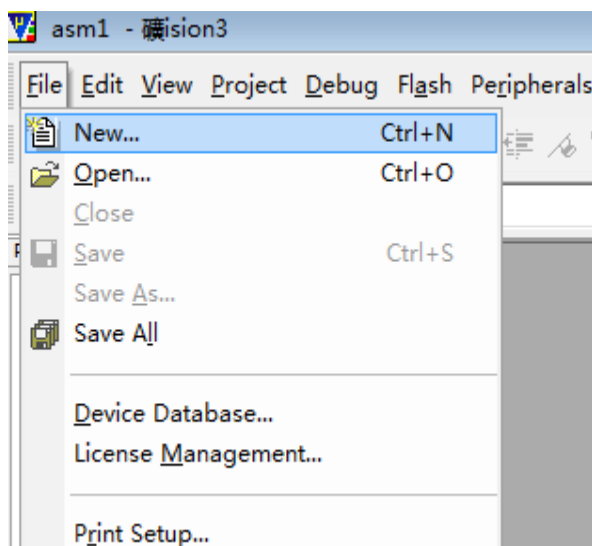


图2-1-4 新建源文件下拉菜单

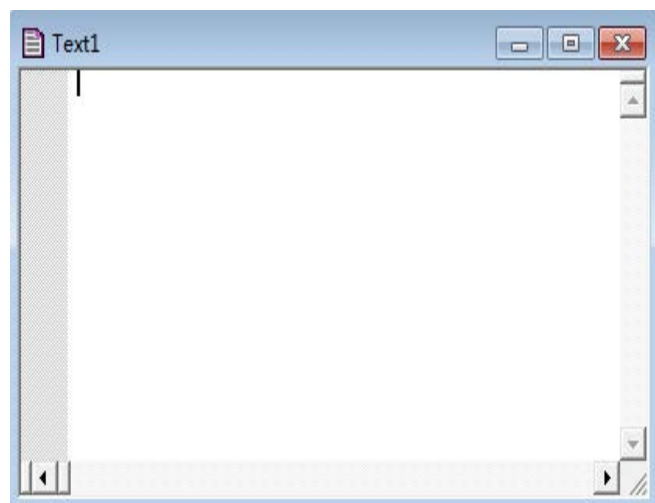


图2-1-5 源程序编辑窗口

(5) 输入程序，完毕后点击“保存”命令保存源程序，如图2-1-6 所示，将Text1 保存成Asm1.asm。Keil C51 支持汇编和C 语言， μ Vision3 会根据文件后缀判断文件的类型，进行自动处理，因此保存时需要输入文件名及扩展名.ASM 或.C。保存后，文件中字体的颜色会发生一定变化，关键字会变为蓝色。

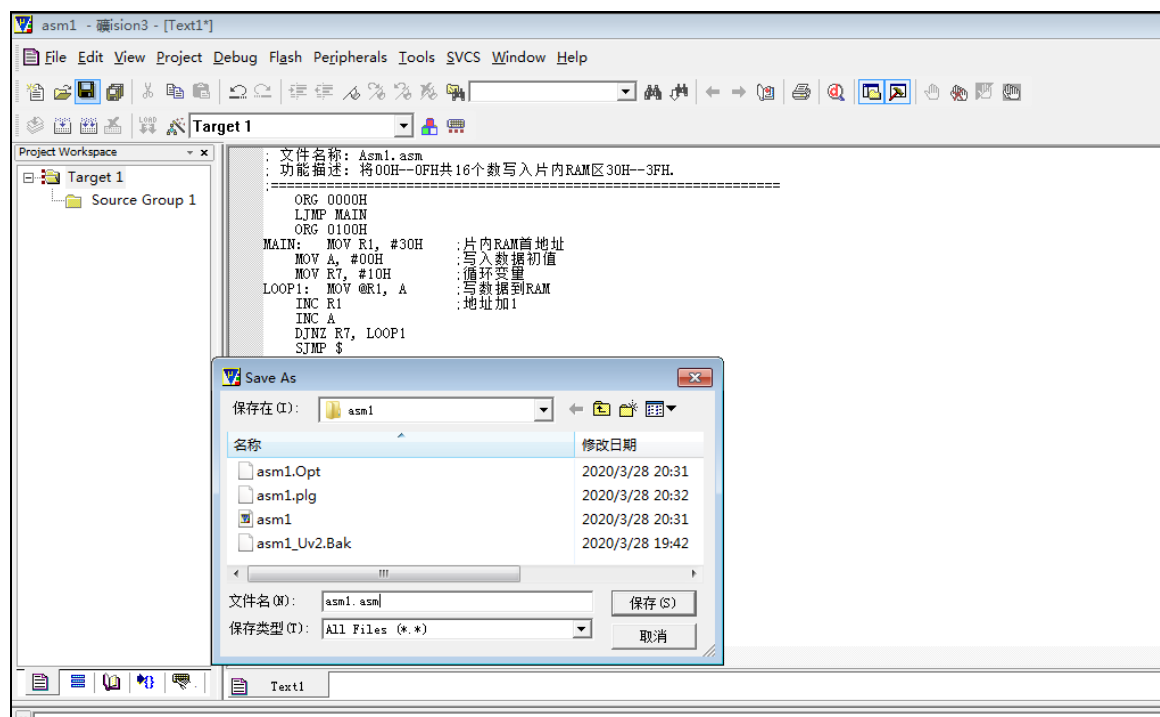


图2-1-6 源文件保存对话框

(6) 程序文件建立后，并没有与Asm1.Uv2 工程建立任何关系。此时，需要将Asm1.asm源程序添加到Asm1.Uv2 工程中，构成一个完整的项目。在Project Window 窗口内，选中Source Group1 点击鼠标右键，会弹出如图2-1-7 所示的快捷菜单，选择Add Files to Group ‘SourceGroup1’ 命令，此时弹出如图2-1-8 所示的添加源程序文件对话框，选择文件Asm1.asm，点击Add 命令按钮即可将源程序文件添加到工程中。

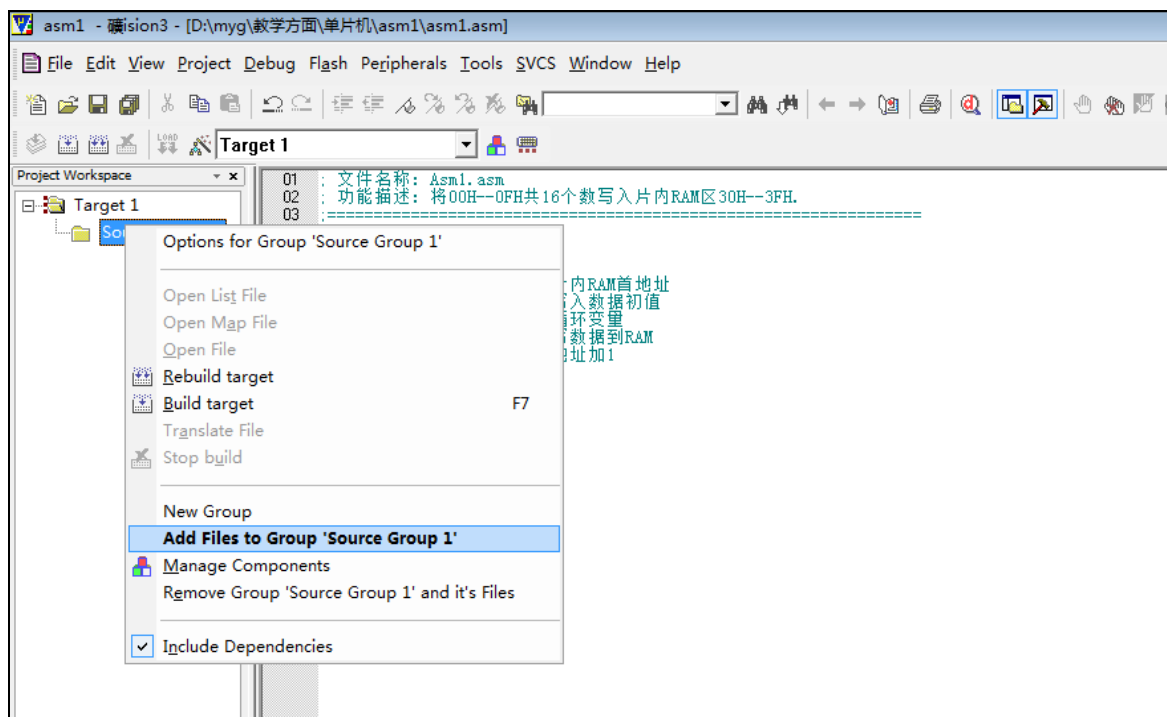


图2-1-7 添加源程序文件快捷菜单

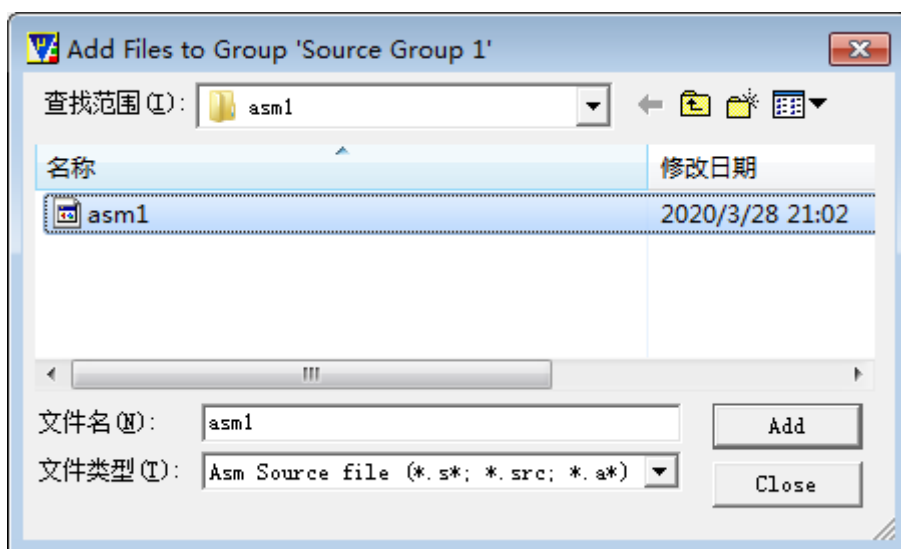



图2-1-8 添加源程序文件对话框

2. 编译、链接程序文件

(1) 设置编译、链接环境，点击  命令，会出现如图2-1-9 所示的调试环境设置窗口，在这里可以设置目标系统的时钟。单击Output 标签，在打开的选项卡中选中 Create Hex File 选项，在编译时系统将自动生成目标代码*.Hex。点击Debug 标签会出现如图2-1-10 所示的调试模式选择窗口。

从图2-1-10 可以看出，μVision2 有两种调试模式：Use Simulator（软件仿真）和Use（硬件仿真）。这里选择硬件仿真，点击Settings 可以设置串口。

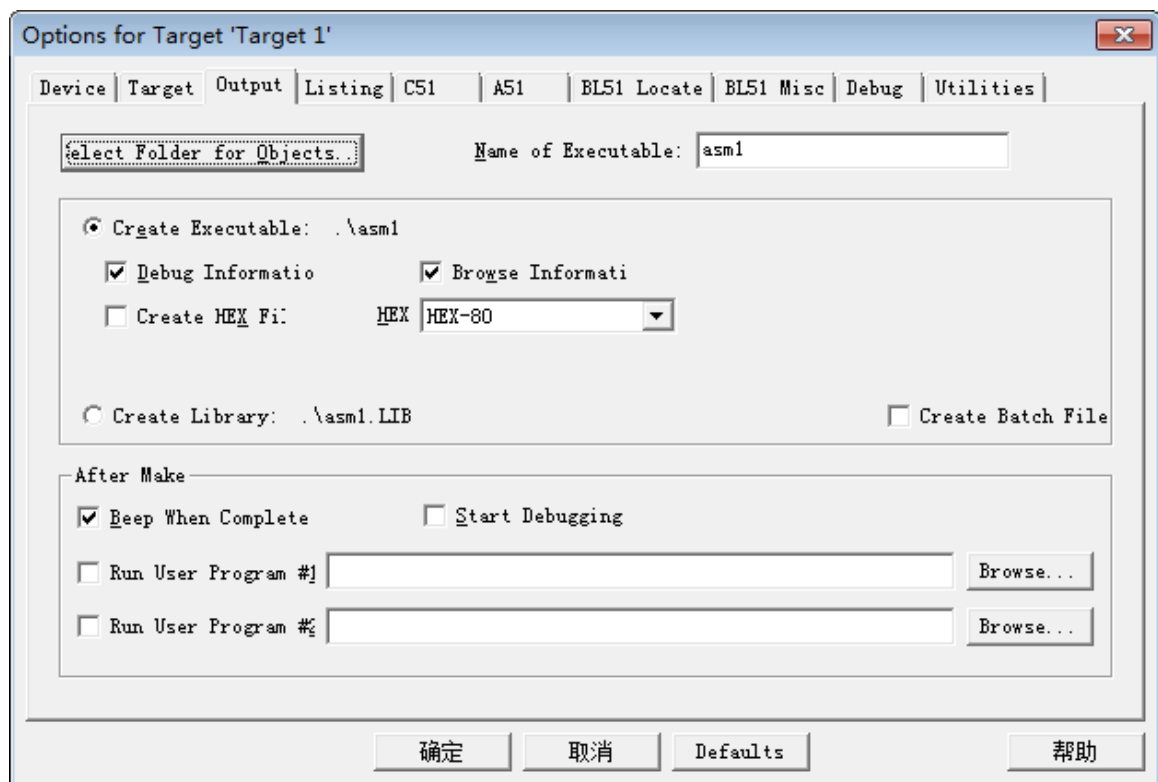


图2-1-9 Keil C51 调试环境设置窗口(Output 标签)

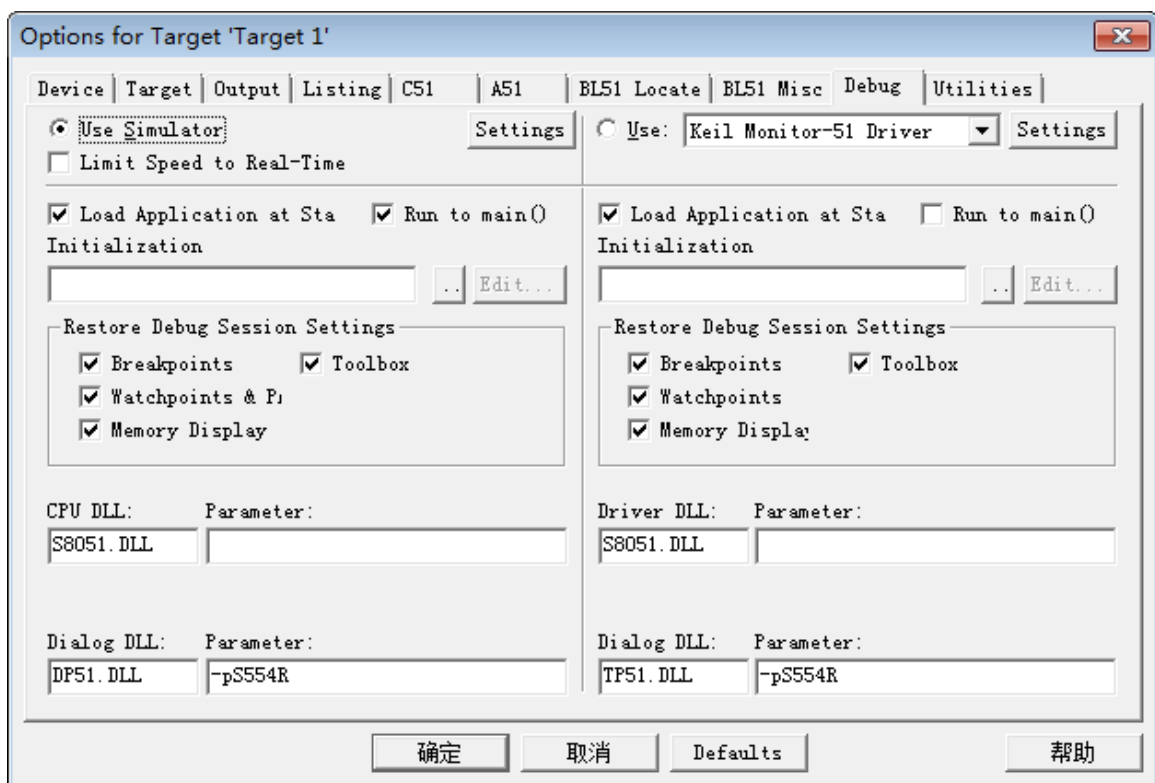




图2-1-10 调试设置窗口(Debug 标签)

(2) 点击  或  命令编译、链接程序，此时会在Output Window 信息输出窗口输出相关信息，如图2-1-11 所示。

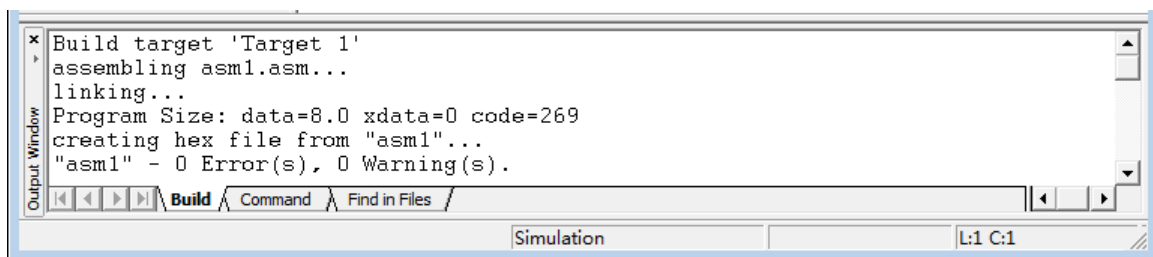



图2-1-11 编译、链接输出窗口

3. 调试仿真程序

(1) 打开系统板的电源，给系统复位后点击  调试命令（注：每次进入调试状态前确保系统复位正常），将程序下载到单片机的FLASH 中，此时出现如图2-1-12 所示调试界面。

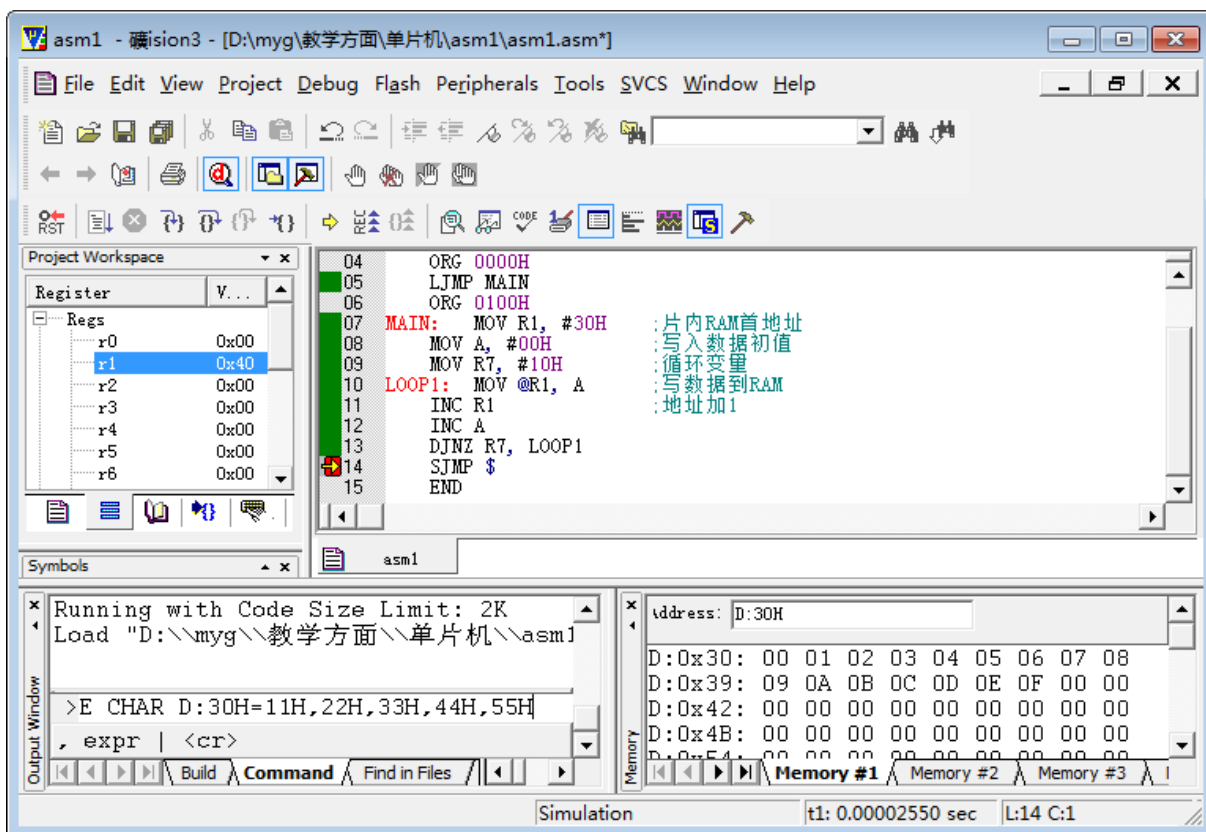





图2-1-12 调试界面

(2) 点击  命令，可以打开存储器观察窗口，在存储器观察窗口的‘Address:’栏中输入D:30H（或0x30）则显示片内RAM30H 后的内容，如图2-1-12 所示。如果输入‘C:’表示显示代码存储器的内容，‘I:’表示显示内部间接寻址RAM 的内容，‘X:’表示显示外部数据存储器中的内容。

(3) 将光标移到SJMP \$语句行，点击  命令，在此行设置断点。

(4) 接下来点击  命令，运行实验程序，当程序遇到断点后，程序停止运行，观察存储器中的内容，如图2-1-13 所示，验证程序功能。

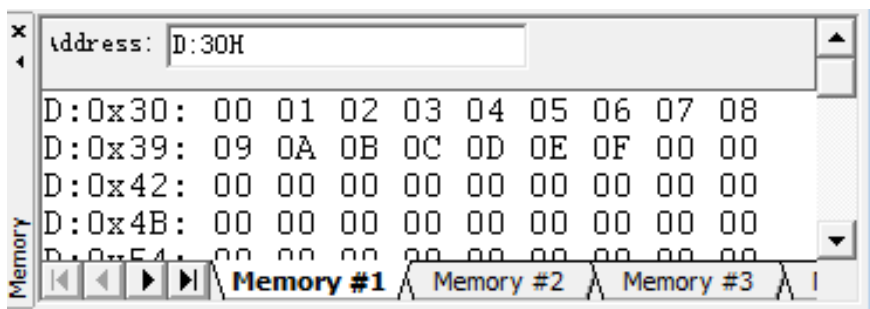


图2-1-13 运行程序后存储器窗口

(5) 如图2-1-12 所示，在命令行中输入 ‘E CHAR D:30H=11H, 22H, 33H, 44H, 55H’ 后回车，便可以改变存储器中多个单元的内容，如图2-1-14 所示。

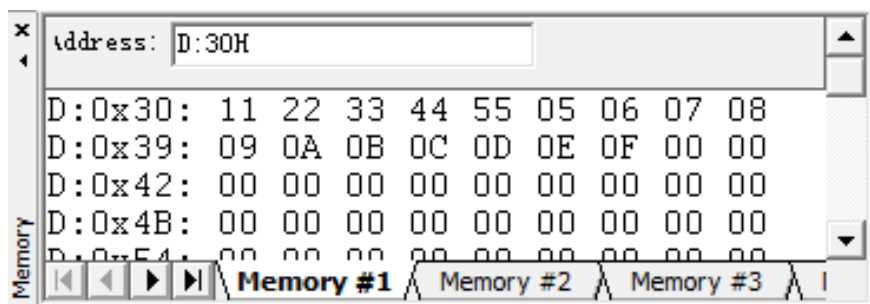


图2-1-14 修改存储器内容

(6) 修改存储器的内容的方法还有一个，就是在要修改的单元上点击鼠标右键，弹出快捷菜单，如图2-1-15 所示，选择 ‘Modify Memory at D:0x35’ 命令来修改0x35 单元的内容，这样每次只能修改一个单元的内容。

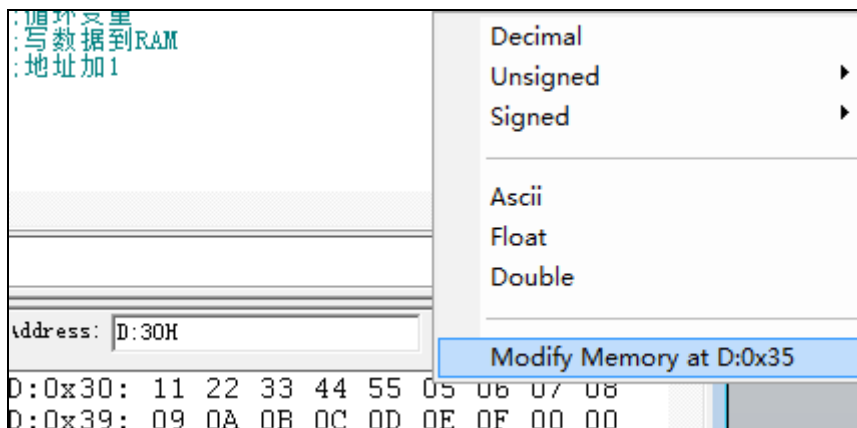





图2-1-15 存储器修改单元

(7) 点击  命令，可以复位CPU，重新调试运行程序，点击  命令，单步跟踪程序。

(8) 实验结束，按系统的复位按键可以复位系统，点击  命令，退出调试。

在此以Asm1.Uv2 工程为例简要介绍了Keil C51 的使用，Keil C51 功能强大，关于Keil C51的使用需要通过日后的使用慢慢掌握。

2.2 数据排序实验

2.2.1 实验目的

熟悉MCS-51单片机的指令系统，掌握数据排序程序的设计方法。

2.2.2 实验设备

PC机一台，[TD-PITE 教学实验系统+TD-51 系统平台]

2.2.3 实验内容

在单片机片内 RAM 的 30H~39H 写入 10 个数，编写实验程序，将这 10 个数按照由小到大的顺序排列，仍写入 RAM 的 30H~39H 单元中。

2.2.4 实验步骤

- (1) 编写实验程序，编译、链接无误后联机调试；（程序流程图可参考图2-2-1）
- (2) 为30H~39H 赋初值，如：在命令行中键入E CHAR D:30H=9, 11H, 5, 31H, 20H, 16H, 1, 1AH, 3FH, 8 后回车，可将这10 个数写入30H~39H 中；
- (3) 将光标移到语句行SJMP \$处，点击命令，将程序运行到该行；
- (4) 查看存储器窗口中30H~39H 中的内容，验证程序功能；
- (5) 重新为30H~39H 单元赋值，反复运行实验程序，验证程序的正确性。

思考题：

修改程序，将这10个数按照由大到小的顺序排列，仍写入RAM 的30H~39H 单元中。

实验程序清单：（Asm5.asm）

```
ORG 0000H
LJMP MAIN
ORG 0100H
MAIN:  MOV R0, #30H    ;数据起始地址
        MOV R7, #09H   ;排序数据个数: 10 个数(比较9次)
LP1:   MOV A, R7
        MOV R6, A
        MOV A, R0
        MOV R1, A
        INC R1
LP2:   MOV A, @R0      ;取出一个数据
        CLR C
        SUBB A, @R1    ;与第二个数进行比较
        JC LP3         ;R0中的数小于R1 中的数则跳转
        MOV A, @R0     ;R0中的数大于R1 中的数, 交换数据
        XCH A, @R1
        MOV @R0, A
LP3:   INC R1
        DJNZ R6, LP2
        INC R0
        DJNZ R7, LP1
        SJMP $
END
```

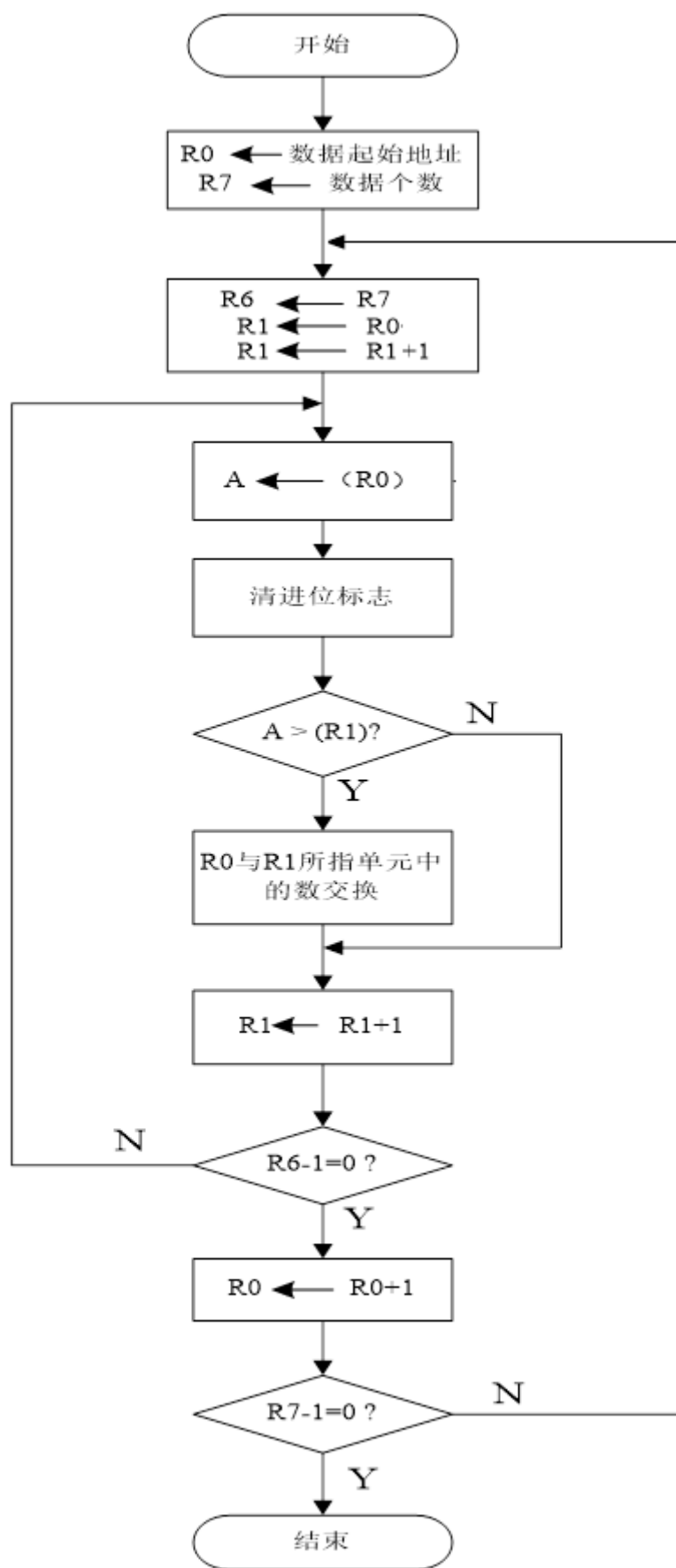


图2-2-1 实验程序流程图

第3章 单片机集成功能模块实验

通过本章实验，学习中断、定时/计数器等功能模块的使用及其程序设计方法。

3.1 中断系统实验

3.1.1 实验目的

了解MCS-51 单片机的中断原理，掌握中断程序的设计方法。

3.1.2 实验设备

PC机一台，TD-PITE 教学实验系统+TD-51 系统平台

3.1.3 实验内容

手动扩展外部中断INT0、INT1，当INT0产生中断时，使LED 8亮8灭闪烁4次；当INT1产生中断时，使LED由右向左流水显示，一次亮两个，循环4次。

3.1.4 实验步骤

- (1) 按图3-1-1 连接实验电路；
- (2) 编写实验程序，编译、链接无误后启动调试；
- (3) 运行实验程序，先按KK1—，观察实验现象，然后按KK2—，观察实验现象；
- (4) 验证程序功能，实验结束按复位按键退出调试。

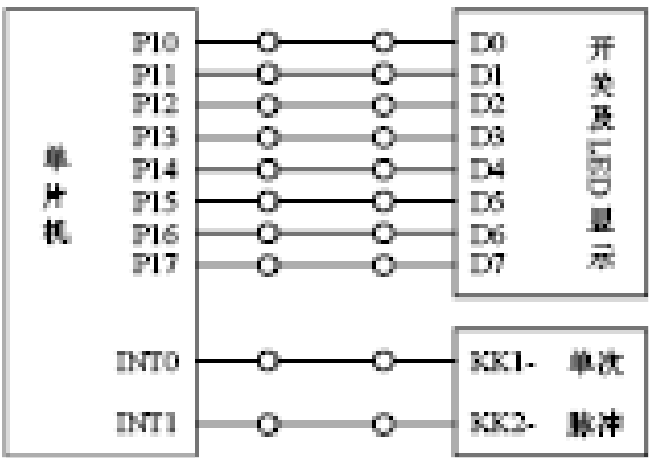


图3-1-1 外中断实验接线图

思考题：

修改程序，当INT1产生中断时，使LED由左向右流水显示，一次亮1个，循环5次。

实验参考程序：

```
=====
; 文件名: INT2.ASM
; 功能说明: 外部中断INT0, INT1操作.
;           外部中断0: 使发光二极管闪烁4次.
;           外部中断1: 使发光二极管由右向左流水显示, 一次亮两个LED.
=====

        ORG 0000H
        LJMP START
        ORG 0003H
        LJMP EINT0
        ORG 0013H
        LJMP EINT1
START:   MOV P1, #00H
        SETB IT0      ;定义外部中断0为边沿触发
        SETB EX0      ;使能外部中断0
        SETB IT1      ;定义外部中断1为边沿触发
        SETB EX1      ;使能外部中断1
        SETB EA        ;使能总中断
        SJMP $

EINT0:   MOV R4, #04H   ;外部中断0中断服务程序
ELP0:    MOV P1, #0FFH
        ACALL DELAY     ;调用延时程序
        MOV P1, #00H
        ACALL DELAY
        DJNZ R4, ELP0
        RETI

EINT1:   MOV A, #03H    ;外部中断1中断服务程序
        MOV R7, #10H
ILP1:    MOV P1, A
        RL A
        RL A
        ACALL DELAY
        DJNZ R7, ILP1
        RETI

DELAY:   MOV R6, #0FFH  ;延时子程序
DLP1:    MOV R5, #0FFH
DLP2:    DJNZ R5, DLP2
        DJNZ R6, DLP1
        RET
        END
```


3.2 定时/计数器实验

3.2.1 实验目的

1. 了解MCS-51 单片机定/计数器的工作原理与工作方式；
2. 掌握定时/计数器T0 和T1 在定时器或计数器两种方式下的编程；

3.2.2 实验设备

PC机一台，TD-PITE 教学实验系统+TD-51 系统平台。

3.2.3 实验内容

将定时/计数器1 设定为计数器方式，每次计数到10在P1.0 引脚上取反一次，观察发光二极管的状态变化。

3.2.4 实验步骤

- (1) 按图3-2-1连接实验线路图；
- (2) 编写程序，联机调试；
- (3) 运行实验程序，按单次脉冲KK1，观察发光管D0 的状态，每10 次变化一次；
- (4) 实验结束，按复位按键退出调试。

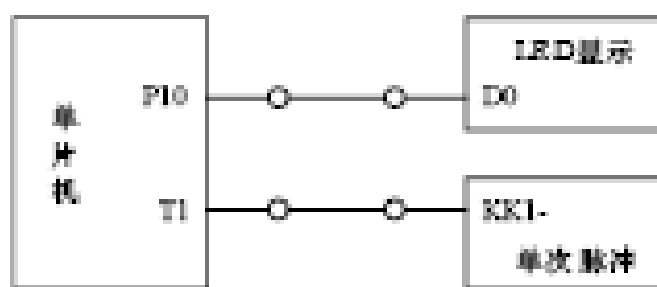


图3-2-1 计数器实验接线图

思考题：

修改程序，用中断方式实现此功能。

实验参考程序：

```
=====
;
; 文件名称：COUNT.ASM
; 功能描述：利用计数器1，在计数10次后对P1.0引脚取反一次
;=====

        ORG 0000H
        AJMP START
        ORG 0100H
START:   MOV TMOD, #60H      ;设置定时器/计数器1为计数器方式，且为方式2
        MOV TH1, #0F6H      ;装入计数初值
        MOV TL1, #0F6H
        SETB TR1             ;启动计数
MLP1:    JNB TF1, MLP1        ;判溢出标志
        CPL P1.0             ;取反P1.0
        CLR TF1              ;清溢出标志
        AJMP MLP1
        END
```

第 4 章 单片机系统扩展实验

4.1 键盘扫描及显示设计实验

4.1.1 实验目的

- 1. 了解8255 的工作方式及应用；
- 2. 了解键盘扫描及数码显示的基本原理，熟悉8255 的编程。

4.1.2 实验设备

PC机一台，TD-PITE 教学实验系统+TD-51 系统平台

4.1.3 实验内容

将8255 单元与键盘及数码管显示单元连接，设计并编写实验程序，实现在键盘按任意键能在LED上显示学生本人的学号（6位, 如:170105）。

4.1.4 实验原理

8255内部结构及引脚如图4-1-1 所示，工作方式控制字和C口按位置位/复位控制字格式如图4-1-2 所示。

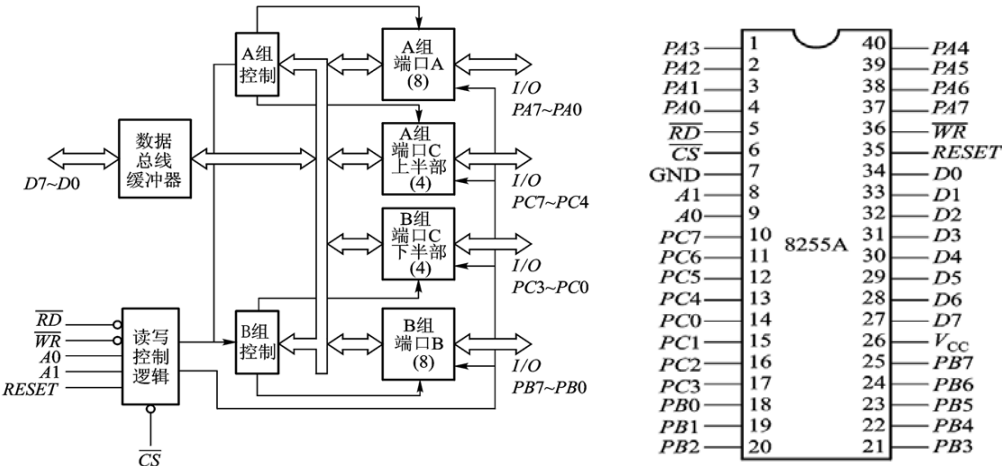


图4-1-1 8255 内部结构及外部引脚图

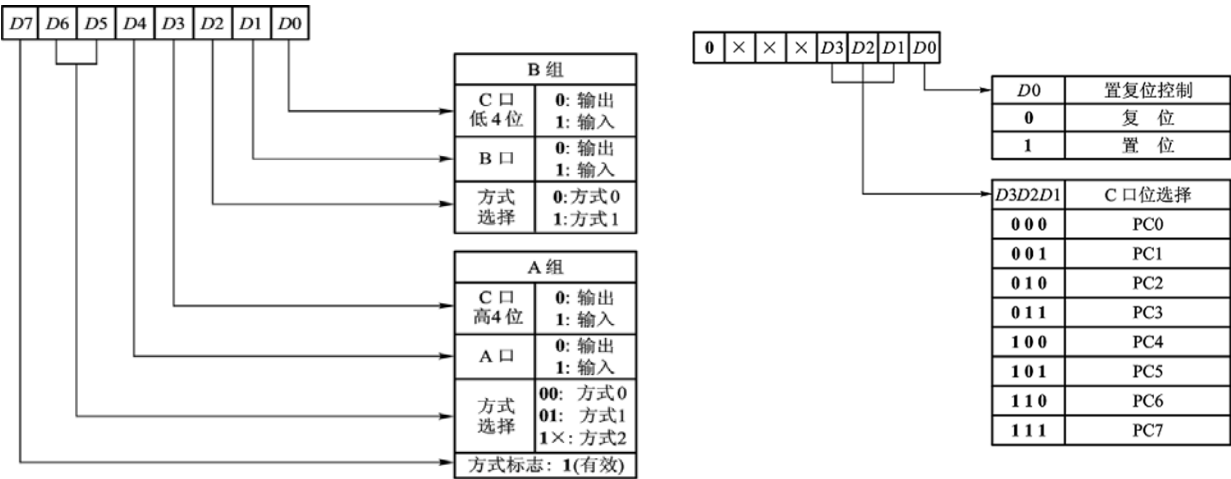


图4-1-2 8255 控制字格式

键盘扫描及数码管显示单元原理图如图4-1-3 所示。

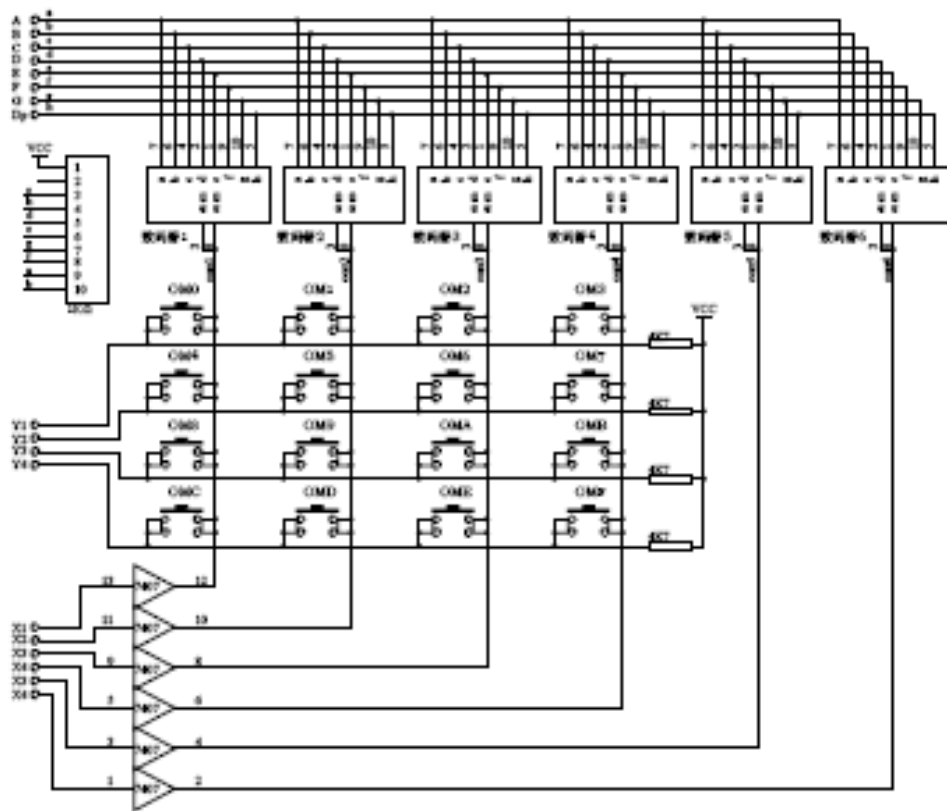


图4-1-3 键盘扫描及数码管显示单元原理图

4.1.5 实验步骤

- 1. 按图4-1-4 连接线路图；
- 2. 设计并编写实验程序，检查无误后编译、连接后启动调试；
- 3. 运行实验程序，按下按键，观察数码管的显示，验证程序功能。

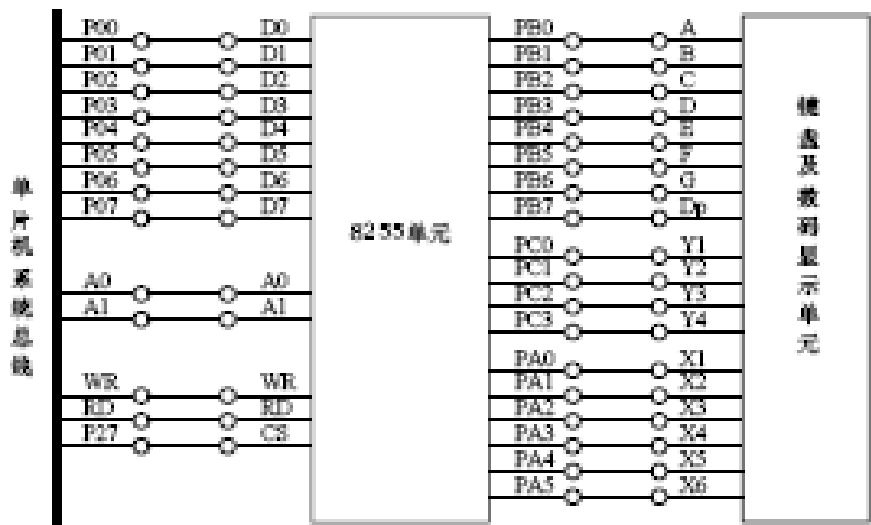
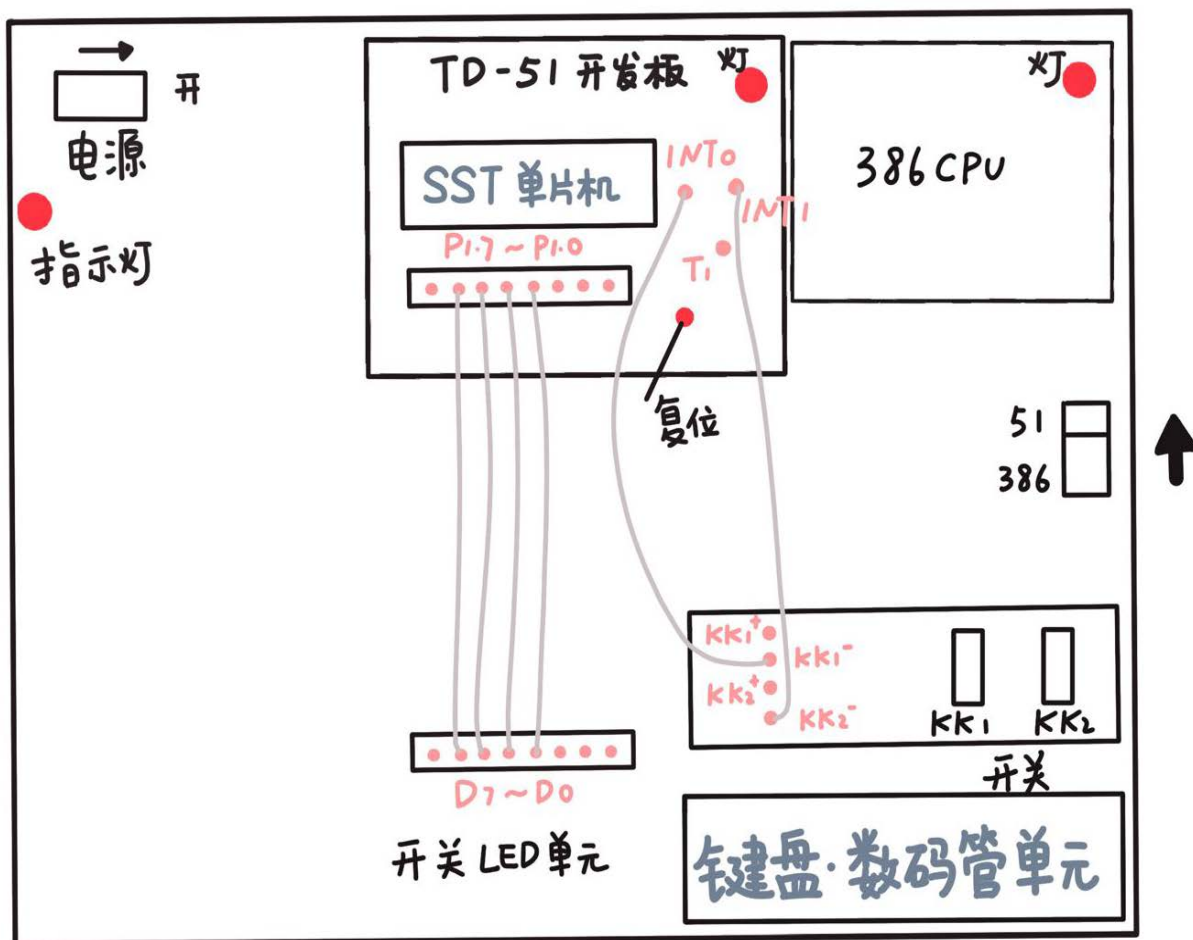


图4-1-4 键盘扫描及数码显示实验接线图

附录:



附图1: 实验箱分布图