

21世纪

高等院校计算机系列教材

单片微型计算机 原理及接口技术

实验指导与实训

林 军 等编著



中国水利水电出版社
www.waterpub.com.cn

21 世纪高等院校计算机系列教材

单片微型计算机原理及接口技术 实验指导与实训

林 军 等编著

中国水利水电出版社

内 容 提 要

本书是《单片微型计算机原理及接口技术》(杨光友、朱宏辉主编)的配套教材。本书以 MCS-51 单片机为例,详细介绍了如何设计和调试 MCS-51 汇编语言程序。全书共分三部分。第一部分是 MCS-51 单片机上机指导。第二部分是 MCS-51 单片机实例,其中包含有 3 个单片机应用开发实例。第三部分是习题与参考答案。读者在第一部分实验的指导下上机做实验,最后通过设计一到两个具体的单片机应用系统来掌握单片机的开发技术。第三部分的练习有利于读者巩固所学习的单片机知识。

本书由浅入深、循序渐进、前后呼应、通俗易懂、图文并茂、内容丰富,可以使读者轻松地掌握使用开发机进行在线仿真的方法,学会如何开发单片机应用系统。

本书可作为高等学校单片机原理与应用课程的配套教材,也可作为高职高专的单片机的配套教材,是初学者和自学者的好帮手,同时对于从事单片机开发的技术人员也具有一定的参考价值。

图书在版编目(CIP)数据

单片微型计算机原理及接口技术实验指导与实训/林军等编著. —北京:中国水利水电出版社, 2004

(21 世纪高等院校计算机系列教材)

ISBN 7-5084-1800-X

I. 单… II. 林… III. ①单片微型计算机—理论—高等学校—教材
②单片微型计算机—接口—高等学校—教材 IV. TP368.1

中国版本图书馆 CIP 数据核字(2004)第 108288 号

书 名	单片微型计算机原理及接口技术实验指导与实训
作 者	林 军 等编著
出版 发行	中国水利水电出版社(北京市三里河路 6 号 100044) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn 电话: (010) 63202266 (总机)、68331835 (营销中心)、82562819 (万水)
经 售	全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京蓝空印刷厂
规 格	787mm×1092mm 16 开本 14 印张 320 千字
版 次	2004 年 10 月第 1 版 2004 年 10 月第 1 次印刷
印 数	0001—5000 册
定 价	20.00 元

凡购买我社图书,如有缺页、倒页、脱页的,本社营销中心负责调换

版权所有·侵权必究

前 言

单片机原理与应用课程已经在高校的许多专业中开设，课程主要介绍单片机的硬件工作原理、指令系统、存储器的扩展以及单片机的接口技术。目前，单片机的开发与应用已成为自动化领域中最活跃的课题之一。在全国大学生电子制作竞赛中，也常常用到单片机。本课程的学习可以为学生毕业后从事实时控制方面的工作打下一个基础，也可以为参加全国大学生电子制作竞赛的学生提供参考。本课程的特点是计算机的硬件和软件紧密结合，是一门实践性很强的课程。因此，实验训练是非常重要的。为了培养学生的实际动手能力，加强实践性教学这个重要的教学环节，我们编写了用于指导学生实验的《单片微型计算机原理及接口技术实验指导与实训》一书。

本书是《单片微型计算机原理及接口技术》（杨光友、朱宏辉主编）的配套教材，是结合高等学校单片机原理与应用课程教学的特点，由高等学校长期从事单片机课程教学与科研开发的一线教师编写而成。

全书共分三部分。第一部分是单片机上机实验指导，共安排了 13 个实验，指导读者由浅入深、循序渐进地学习和掌握上机操作的方法。在各个实验指导中，都有实验的具体要求和详细的操作步骤，引导读者一步步地完成实验。实验中的大部分程序设计题都提供了流程图；实验中的程序设计题都有参考答案，这些参考程序，都已经在开发机上调试成功。第二部分是 MCS-51 单片机应用实例，其中包含有 3 个单片机控制的实时控制系统。实例 1 是单片机控制的彩灯，实例 2 是单片机控制的交通信号灯，实例 3 是单片机控制的数字钟。实例中的代码都是经过调试成功的程序。第三部分是单片机习题及参考答案。读者在第一部分实验的指导下上机做实验，最后通过设计一到两个具体的单片机应用系统来掌握单片机的开发技术。第三部分的练习，剪性强，题型多样，可以对学生进行基本的训练，有利于读者巩固所学习的单片机知识。并且，习题还有参考答案，便于自学者自我测试。

书中的每一个实验都是一个相对独立和较为完整的单元，读者可以根据自己所使用教材中的章节顺序来决定做实验的顺序，因此，本书既可以与《单片微型计算机原理及接口技术》（杨光友、朱宏辉主编）的教材配套使用，也可以与各种单片机原理与应用的教材配套使用。

本书是在日常教学的基础上编写而成的。书中大部分实验都经过了学生多次上机操作验证。书中的程序都通过了调试，并且书中还配有大量的插图，可以帮助初学者快速入门。

书中的程序是在编者所在学校所使用的开发机上调试的。目前，单片机开发机的种类繁多，读者可以参考书中的程序，再针对具体应用的开发机进行小的调整或修改，就能使用这些程序。总之，这些程序具有很强的通用性。

参加本书大纲讨论、画图、程序录入、校对等编写工作的有余铮、王虹、桂超、陈莹、王文剑、陈恒、李荆洪、李翔、关培超、宋崇杰、周美珍、陈开经、杨凤霞等。

由于编者水平有限，书中难免有疏漏不当之处，恳请广大读者批评指正。

编者

2004年8月

目 录

前言

第一部分 单片机上机实验指导	1
实验 1 开发机的使用	1
实验 2 开发机与计算机的连接与使用	8
实验 3 程序设计（一）	11
实验 4 程序设计（二）	19
实验 5 并行端口	30
实验 6 串行端口	36
实验 7 定时器与中断	39
实验 8 简单并行 I/O 端口扩展	47
实验 9 使用 8255 扩展并行 I/O 端口	51
实验 10 显示器接口	58
实验 11 A/D 转换	64
实验 12 D/A 转换	69
实验 13 继电器控制	75
第二部分 MCS-51 单片机应用实例	77
实例 1 单片机控制的彩灯	77
实例 2 单片机控制的交通信号灯	83
实例 3 单片机控制的数字钟	97
第三部分 习题与参考答案	110
第 1 章 单片微型计算机概述	110
第 2 章 MCS-51 单片机的硬件结构	117
第 3 章 MCS-51 单片机的指令系统	125
第 4 章 MCS-51 单片机汇编语言程序设计	133
第 5 章 定时器/计数器与中断系统	143
第 6 章 串行通信及其接口	154
第 7 章 MCS-51 单片机系统扩展技术	160
第 8 章 MCS-51 单片机的接口技术	167
附录 1 实验参考程序	179
附录 2 MCS-51 系列单片机指令表	211
附录 3 分类指令表	215

第一部分 单片机上机实验指导

实验 1 开发机的使用

实验目的

1. 熟悉开发机的键盘及功能。
2. 掌握开发机的使用方法。
3. 掌握汇编、汇编语言源程序、机器语言目标程序以及机器码等基本概念。

实验内容

1. 练习通过键盘输入程序的机器代码。
2. 练习通过键盘运行程序。
3. 练习设置断点。

实验设备

1. 单片机开发机 1 台。
2. 微机电源一台。

开发机的键盘介绍

用户可以通过开发机上的 32 个键向开发机发出各种操作命令。键盘的示意图如图 1.1 所示。键盘分为数字键和命令键两种键，图 1.1 中左边的 16 个键是数字键，右边的 16 键是命令键，大多数键有两种功用，例如数字键 0 可以作为数字 0 使用，也可以作为当前工作寄存器 R0 使用。

常用键的功能如下所示：

(1) 0~F 键——数字键。0~F 分别表示十六进制数，用于输入地址、数据或机器语言程序代码。0~7 除了表示数字以外，还用于表示寄存器。0~7 分别表示当前工作寄存器 R0~R7；8~F 分别表示特殊功能寄存器 DPL、DPH、ACC、B、PSW、SP 和程序计数器的 PCL 和 PCH。

(2) MON 键——返回待命状态键。按 MON 键可以迫使开发机进入待命状态。通常用 MON 键进行如下的操作：

- 清除已输入的数字。
- 退出一些操作，例如退出存储器读写状态，退出寄存器读写状态等。

- 强迫终止用户程序的执行。

7/R7	8/R8	9/R9	A/ACC	TV/ME	EG/DL	PRT	EX/FV
4/R4	5/R5	6/R6	B/B	RG/FS	RW/IS	COMP	SCAL
1/R1	2/R2	3/R3	C/PSW	F1/LS	EV/UN	MOVE	STEP
0/R0	F/PCH	E/PCL	D/SP	F2/NX	EC/EP	DAR	MON

图 1.1 开发机键盘示意图

在用户执行程序时，按 MON 键则立即停止用户程序的执行，并在显示器上显示终止的地址及当时 A 累加器的内容，再按一次 MON 键，就返回到待命状态 0，即显示闪烁的“P.”。在程序终止时，用户所有 CPU 寄存器的内容都被保护起来，即数据被压入堆栈，通过寄存器读写命令可以查看寄存器的内容。

按 MON 键不会影响用户的存储区、寄存器以及已设置的断点。

说明：当开发机处于待命状态 0 时，显示器的最左端显示一个闪动的“P.”提示符，表示开发机实验系统处于初始化状态，等待用户操作。以下操作都可以使开发机进入待命状态 0。

- 在实验系统接通电源后，机器自动复位，处于待命状态 0。
- 按 RESET 复位键后，机器处于待命状态 0。
- 在大多数情况下，按 MON 键也可以使机器处于待命状态 0。

(3) ME、LS 和 NX 键——程序存储器读写命令。这一组命令用来检查（读出）或更改（写入）内存单元的内容，因此通过这些键命令的操作可以给开发机输入程序和数据。

说明：在单片机中程序存储器是由只读存储器构成的，而在开发机中，为了使用户能够调试和修改程序，其程序存储器是由随机读写存储器构成的。

使用 LS (LAST) 键，可以进入上一个存储单元；使用 NX (NEXT) 键，则可以进入下一个存储单元。

(4) RG、LS 和 NX 键——寄存器读写、特殊功能寄存器检查和片内 RAM 读写命令。使用这些键可以对内部数据 RAM、特殊功能寄存器进行读写操作。其键盘的代号或地址所表示的存储单元如表 1.1 所示。

表 1.1 键盘代号或地址所表示的存储单元

键盘代号/地址	寄存器	键盘代号/地址	寄存器	键盘代号/地址	寄存器
0	R0	E	PCL	90	P1
1	R1	F	PCH	98	SCON
2	R2	00H~7FH	8032 片内 RAM	99	SBUF
3	R3	80H	P0	A0	P2
4	R4	81H	SP	A8	IE

续表

键盘代号/地址	寄存器	键盘代号/地址	寄存器	键盘代号/地址	寄存器
5	R5	82H	DPL	B0	P3
6	R6	83H	DPH	B8	IP
7	R7	87	PCON	D0	PSW
8	DPL	88	TCON	E0	ACC
9	DPH	89	TMOD	F0	B
A	SP	8A	TL0		
B	ACC	8B	TH0		
C	B	8C	TL1		
D	SP	8D	TH1		

(5) RW、LS 和 NX 键——外部数据存储单元读写命令。当键入 2 位地址时，读写开发机 8032 的 80H~FFH 单元的内容；当键入 4 位地址时，则实现外部数据存储单元的读写和 I/O 的读写以及扩展的外部 8155RAM 的 256 字节单元的读写。

(6) STEP 键——单步执行命令。按一次 STEP 键，开发机单步执行一条用户指令。

(7) EX 键——执行程序命令。从当前 PC 所指向的地址开始执行用户程序。当程序执行时，显示器显示执行标记“r”。用执行键 EX 执行用户程序，在程序中遇到断点时会停下来，并保护所有的寄存器，显示断点地址和该单元的内容。

(8) IS 键——插入命令。在当前的地址单元插入一条指令，当前地址单元的内容向下移动一个单元，依此类推。

(9) DL 键——删除命令。删除当前地址单元的一条指令，后面单元的内容依次向上移动一个单元。

(10) F1 键——标志设置命令。有多种作用，例如，在待命状态 1 时按 F1 键，进入仿真 2 状态，即仿真 EPROM，PC 值指向外部程序空间，LED 显示“H.....”。也可以利用 F1 键来设置程序运行的断点。

(11) F2 键——标志设置命令。有多种作用，例如，在待命状态 1 时按 F2 键，进入仿真 1 状态，即仿真 RAM，PC 值指向外部存储器空间，LED 显示“P.....”。

(12) RESET——复位键。复位键位于开发机的右侧，用于按键复位，其作用与上电复位相同。在任何时刻按复位键，都使单片机进入初始状态。复位后，开发机处于待命状态 0 的状态。复位操作设置用户堆栈指针为 07H，并使实验系统中的 8155、8255 芯片复位。

按复位键不改变用户存储区的内容，也不改变 8032 片内 RAM 的内容以及外部数据存储区的内容。

实验要求与步骤

1. 练习在开发机上运行程序。要求通过键盘输入以下程序的机器码。运行程序，并记录程序运行后各个相关的寄存器单元的内容。汇编程序和机器代码如下所示。

地址	机器码	汇编语言源程序
		ORG 6000H
6000	7840	MOV R0, #40H
6002	7F0A	MOV R7, #0AH
6004	7400	MOV A, #00H
6006	F6	LOOP1: MOV @R0, A
6007	08	INC R0
6008	04	INC A
6009	DFFB	DJNZ R7, LOOP1
600B	7840	MOV R0, #40H
600D	902000	MOV DPTR, #2000H
6010	7F0A	MOV R7, #0AH
6012	E6	LOOP2: MOV A, @R0
6013	F0	MOVBX @DPTR, A
6014	08	INC R0
6015	A3	INC DPTR
6016	DFFA	DJNZ R7, LOOP2
6018	80FE	SJMP \$
		END

操作步骤如下：

- (1) 接好微机电源线，将开发机与微机电源正确地连接起来。
- (2) 打开微机电源。
- (3) 开发机进入待命状态 0，即显示器显示闪烁的“P.”。
- (4) 运行 8 字循环程序，检查开发机是否工作正常。

提示：键入 FFC1，然后按 EX 键运行 8 字循环程序。FFC1 是 8 字循环程序的首地址。

说明：开发机的型号不同，8 字循环程序的首地址是不同的，因此，要查看所使用的开发机的手册来确定其首地址。

- (5) 如果 8 字循环程序运行正常，则按复位键，使开发机回到待命状态 0 的状态。
- (6) 键入程序的首地址 6000。
- (7) 按 ME 键进入程序存储器读写状态。
- (8) 输入十六进制的机器码。
- (9) 按 NX 键进入下一个程序存储器单元，依次写入相应的机器代码。
- (10) 当全部程序输入完毕后，使开发机进入待命状态 0。

(11) 键入程序的首地址，然后按 EX 键运行程序。这时显示器最左边显示执行标记“P”，表示程序正在运行中。

说明：因为 MCS-51 的指令系统中没有结束指令，因此，本程序是采用动态停机的方法来结束程序的执行的，即执行 SJMP \$ 指令，该指令是一条短转移指令，它的功能是转

到本指令执行，形成一个无限循环，从而形成动态停机。由于机器在不停地执行这条语句，所以实验系统显示“r”。

(12) 按复位键，使开发机回到待命状态 0 的状态。

(13) 检查各个相关寄存器的内容，并记录运行结果。

2. 练习单步执行程序。要求使用 STEP 键单步执行下列程序，并记录每一条指令执行后相关寄存器的内容。汇编程序和机器代码如下所示。

(1) 程序 1 的汇编程序和机器代码如下所示。

地址	机器码	汇编语言源程序
		ORG 6000H
6000	7800	MOV R0, #00H
6002	7901	MOV R1, #01H
6004	7A02	MOV R2, #02H
6006	7B03	MOV R3, #03H
6008	EB	MOV A, R3
6009	08	INC R0
600A	09	INC R1
600B	0A	INC R2
600C	0B	INC R3
600D	28	ADD A, R0
600E	FC	MOV R4, A
600F	80FE	SJMP \$

操作步骤如下：

1) 输入程序，其操作与上题相同。

2) 从程序的第一条指令开始，单步执行程序，并记录每条指令执行后的中间结果。

填写单步执行程序表，如表 1.2 所示。

表 1.2 单步执行程序表

执行指令	寄存器	记录内容
MOV R0, #00H	R0	
MOV R1, #01H	R1	
MOV R2, #02H	R2	
MOV R3, #03H	R3	
MOV A, R3	A	
INC R0	R0	
INC R1	R1	
INC R2	R2	
INC R3	R3	
ADD A, R0	A	
MOV R4, A	R4	

提示：使开发机处于待命状态 0，键入程序的首地址，然后按 STEP 键，单步执行程序，每按一次 STEP 键，机器单步执行一条指令。

(2) 程序 2 的汇编程序和机器代码如下所示。要求填写单步执行程序表，如表 1.3 所示。

地址	机器码	汇编语言源程序
		ORG 6000H
6000	7820	MOV R0, #20H
6002	7930	MOV R1, #30H
6004	7633	MOV @R0, #33H
6006	7744	MOV @R1, #44H
6008	E6	MOV A, @R0
6009	27	ADD A, @R1
600A	08	INC R0
600B	F6	MOV @R0, A
600C	80FE	SJMP \$

表 1.3 单步执行程序表

执行指令	寄存器	记录内容
MOV R0, #20H	R0	
MOV R1, #30H	R1	
MOV @R0, #33H	R0, (20H)	
MOV @R1, #44H	R1, (30H)	
MOV A, @R0	A, R0, (20H)	
ADD A, @R1	A, R1, (30H)	
INC R0	R0	
MOV @R0, A	R0, (21H), A	

3. 练习设置程序的断点。要求在地址 100B 处设置断点，即在指令 SJMP \$ 的首地址处设置断点，然后从程序的首地址执行程序到断点处，并记录程序运行后有关存储器单元的内容。汇编程序和机器代码如下所示。

地址	机器码	汇编语言源程序
		ORG 1000H
1000	904000	MOV DPTR, #4000H
1003	7810	MOV R0, #10H
1005	74FF	MOV A, #0FFH
1007	F0	L1: MOVX @DPTR, A
1008	A3	INC DPTR
1009	D8FC	DJNZ R0, L1

100B 80FE SJMP \$

说明：在指令 SJMP \$ 的首地址处设置断点，指令 SJMP \$ 并不被执行，程序只执行到断点处就停下来。

操作步骤如下：

- (1) 输入程序。
- (2) 在地址 100B 处设置断点，即按 100B，再按 F1 键。
- (3) 按程序的首地址 1000。
- (4) 按 EX 键，则机器从首地址开始执行程序到断点处。
- (5) 观察机器显示的地址是否是断点的地址，并记录外部数据存储器 4000H~400FH 存储单元中的内容。

4. 关机。

5. 思考题。

- (1) 什么是汇编语言源程序？
- (2) 什么是汇编？
- (3) 什么是机器语言？

实验 2 开发机与计算机的连接与使用

实验目的

1. 掌握开发机与计算机的连接与使用方法。
2. 学习使用开发机的软件来调试程序。

实验内容

1. 练习从计算机输入程序到开发机。
2. 练习调试程序。
3. 练习设置断点。

实验设备

1. 单片机开发机一台。
2. 微机电源一台。
3. 计算机一台。

开发机软件简介

用于 MCS-51 系列单片机的开发机种类繁多,各个厂家都纷纷推出自己的产品。目前的单片机开发机都能与计算机连接起来,利用计算机来调试程序。一般来说,开发人员都使用计算机来调试开发的单片机应用程序,而不使用实验 1 中的方法来调试开发的单片机应用程序。实验 1 介绍的在开发机上调试程序的方法只是为了使初学者熟悉开发机的使用,理解所学的基本概念。真正实际工作中和以后的实验中,是将开发机与计算机连接后,使用计算机来调试程序。要在计算机上调试 MCS-51 系列单片机程序,必须将开发机与计算机通过通信线连接起来,并且必须具有相应的软件支持——这就是开发机的软件。在这个软件的支持下,用户可以编辑 MCS-51 汇编语言源程序、汇编源程序以及调试程序。

实验要求与步骤

1. 准备工作。

操作步骤提示如下:

- (1) 将单片机与计算机连接起来,插好插头。
- (2) 打开单机电源,使开发机进入待命状态,单片机显示“P.”(如果不能正常显示提示符“P.”,应立即关闭单机电源,查明故障原因)。
- (3) 打开计算机显示器电源。

- (4) 打开计算机主机电源, 启动计算机。
- (5) 在计算机上运行与开发机配套的调试程序所使用的软件。
- (6) 输入和编辑源程序, 其扩展名为.ASM。
- (7) 汇编所编辑的源程序。
- (8) 运行程序。

说明: 由于生产开发机的厂家不同, 其相应的软件也有所不同, 在使用开发软件之前, 应认真阅读软件使用说明书, 然后再联机使用。

注意: 对于初学者, 有一个常见的问题, 就是计算机端口的选择, 例如, 如果将开发机连接在计算机的串行端口 COM2 上, 而计算机上软件中的设置为 COM1, 则会出现计算机与开发机连接不上的错误。这时, 只需要将计算机中的端口设置成 COM2, 与实际连接相符即可。

2. 设计程序, 交换内部数据 RAM 的 30H 和 40H 单元中的值。要求在计算机上编辑、汇编和运行程序, 并查看程序运行结果。参考程序如下:

方法 1:

```
ORG          2000H
MOV          R7, 30H
MOV          30, 40H
MOV          40, R7
SJMP        $
```

方法 2:

```
ORG          2000H
MOV          R0, #40H
MOV          R1, #30H
MOV          A, @R0
MOV          B, @R1
MOV          @R0, B
MOV          @R1, A
S JMP       $
```

方法 3:

```
ORG          2000H
MOV          A, 30H
XCH         A, 40H
MOV          30H, A
SJMP        $
```

方法 4:

```
ORG          2000H
PUSH        30H
```

PUSH	40H
POP	30H
POP	40H
SJMP	\$

3. 练习单步执行程序。单步执行程序，并记录每一条指令执行后相关寄存器的内容。
4. 练习连续执行程序，并查看程序的运行结果。
5. 练习设置程序的断点，执行程序到断点处，并记录程序运行后有关寄存器的内容。
6. 关机。

操作步骤如下：

- (1) 关闭计算机。
- (2) 关闭计算机显示器电源。
- (3) 关闭单机电源。
- (4) 拔下开发机与计算机的连接线插头。

7. 思考题。

- (1) 汇编语言源程序文件的扩展名是什么？
- (2) 源程序汇编后产生的目标程序文件的扩展名是什么？
- (3) 十六进制可执行程序文件的扩展名是什么？

实验3 程序设计（一）

实验目的

1. 熟悉 MCS-51 的指令系统。
2. 掌握 MCS-51 的汇编语言程序的设计方法。
3. 练习编写顺序结构、分支结构和循环结构的程序。

实验内容

1. 练习设计数据传送程序。
2. 练习设计二进制的加法和减法程序。
3. 练习设计 BCD 码加法程序。

实验设备

1. 单片机开发机一台。
2. 微机电源一台。
3. 计算机一台。

实验要求与步骤

1. 编制数据传送程序，并在开发机上运行程序，查看运行结果。

(1) 将外部数据 RAM 2000H~200FH 中的 16 个数分别传送到内部数据 RAM 40H~4FH 中，设外部数据 RAM 2000H~200FH 中的数据分别为 00H、01H、02H、…、0FH。

提示：由于外部数据 RAM 2000H~200FH 中的数据是加 1 递增的，因此可以使用手工的方法或者程序的方法来设置这些单元的数据初值。这里使用程序的方法来设置其初值。程序的流程图如图 3.1 所示。其中寄存器：

DPTR——存放外部数据存储器的地址，作为外部数据存储器的地址指针。

R7——循环变量，在这里为 16D。

A——累加器，作为中间单元。

(2) 将外部数据 RAM 2000H~200FH 单元中的 16 个数分别传送到外部数据 RAM 4000H~400FH 中，设外部数据 RAM 2000H~200FH 中的数据分别为 00H、01H、02H、…、0FH。

设计思想：先将 2000H~2000FH 单元中的 16 个数传送到 40H~4FH 的中间单元中，然后再将中间单元中的数据传送到 4000H~400FH 中。