

8098 单片机及扩展接口

<http://www.tup.tongji.edu.cn>

T23681-33

T 33

412706

8098 单片机及扩展接口 实验指导

唐庆玉 编著

清华大学出版社

(京)新登字 158 号

JS158/16
内 容 简 介

本书的实验内容可分为三种类型:第一类是关于 8098 单片机本身功能的实验,这类实验有助于读者学习 8098 单片机的原理;第二类是关于存储器扩展的实验,这类实验使读者学习如何用 8098 单片机构成最小系统;第三类是关于各种常用可编程接口的实验,这类实验可使读者学习用 8098 单片机控制各种可编程接口的接线方法及编程方法。在前 26 个实验中,每个实验又包括 1~6 个实验题目,每个实验题目都提供有详细接线图及程序清单。在实验 27 中提出一些课程设计实验题目。

本书的实验内容十分丰富,且具有实用性,可作为大专院校“微机原理”、“微机接口技术”以及“智能电子仪器”等课程的教学实验指导书,也可供从事科研及维修的工程技术人员参考。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

8098 单片机及扩展接口实验指导/唐庆玉编著. —北京:清华大学出版社,1998
ISBN 7-302-02914-8

I. 80… II. 唐… III. 单片微型计算机接口实验 IV. TP368.1

中国版本图书馆 CIP 数据核字(98)第 08601 号

出版者:清华大学出版社(北京清华大学校内,邮编 100084)

因特网地址:www.tup.tsinghua.edu.cn

印刷者:北京市清华园胶印厂

发行者:新华书店总店北京科技发行所

开 本:787×1092 1/16 印张:14 字数:331 千字

版 次:1998 年 7 月 第 1 版 1998 年 7 月 第 1 次印刷

书 号:ISBN 7-302-02914-8/TP·1543

印 数:0001~8000

定 价:15.00 元

前 言

自 1989 年美国 Intel 公司推出高性能的 8098 单片机以来,在我国的仪器仪表、工业自动化控制、机电一体化、电力电子、医疗电子等行业的科学研究及工业生产中得到广泛的应用。在各高等院校的电子学、自动化、计算机、电力电子、仪器仪表、机械制造及自动控制、医疗电子仪器等有关专业都设置了以 8098 单片机为主要教学内容的课程,这些课程有“微机原理”、“微机应用”、“微机控制”、“微机接口技术”、“智能电子仪器”、“医疗电子仪器”等。

微机原理及应用这一类课程,是实践性很强的课程,要求学生不但听懂课堂教学内容,而且必须做实验。只有通过做实验,才能真正掌握其基本原理,从而进一步达到熟练应用的目的。

多年来,作者从事关于 8098 单片机应用的教学及科研,以教学和科研的经验为基础编写了此实验指导书,并在大学本科教学及培训班中多年使用并反复修改,力求使本书体现先进性、系统性、实用性和趣味性。在内容上有基本性的实验,用于学生掌握 8098 单片机的基本性能;有提高性的实验,用于学生掌握各种常用接口技术;还有综合性的实验,用于学生对 8098 单片机及各种接口进行综合应用。每个实验都有实验要求及设计方法,并提供详细的电路图和参考程序清单。这样,能使学生由浅入深、循序渐进掌握 8098 单片机技术,并能达到举一反三的程度。本书提供的实验内容相当丰富,多达 80 多个实验可分为必做和选做,以适应不同程度的学生。

本书的实验都以 ATD-98B 8098 仿真板为开发工具而编排的。实际上,本书中提供的电路图及程序清单都适用于任何型号的 8098 开发系统。

本书中提供的一些电路图及程序设计,大都是作者多年科研教学工作的经验积累,例如数据采集、波形显示、DMA 数据传送、键盘控制、打印机控制、CRT 字符显示、计算机通信、电子表等。因此,本书不但可作为大专院校的学生进行教学实验的指导书,也可作为从事微机应用的科研设计人员和智能仪器设备维修运行人员进行设计和自学的参考书。

鉴于作者水平有限,书中难免有不妥之处,恳请读者予以指正,不胜感谢。

唐庆玉
1997 年 8 月

目 录

第1部分 实 验

实验 1 8098 单片机开发系统操作实验	3
1-1 屏幕编辑/汇编/单步执行操作实验	3
1-2 反汇编/设置断点/连续执行/存储器检查操作实验	5
实验 2 8098 单片机汇编语言程序设计实验	8
2-1 加减法程序设计	8
2-2 乘法程序设计	9
2-3 除法程序设计	10
实验 3 8098 单片机 I/O 接口扩展与控制实验	12
3-1 一般输出接口扩展与 LED 控制实验	12
3-2 一般输入接口扩展与蜂鸣器控制实验	14
3-3 用一般输入输出接口控制矩阵键盘实验	15
实验 4 8098 单片机外部中断实验	17
4-1 由 EXTINT 输入的外部中断实验	17
4-2 由 ACH7 输入的外部中断实验	19
实验 5 8098 单片机存储器扩展实验	20
5-1 8kB RAM 6264 扩展实验	20
5-2 8kB EPROM 2764 扩展实验	22
5-3 8098 单片机最小系统实验	23
实验 6 8098 单片机片内 A/D 转换器应用实验	24
6-1 单通道查询方式数据采集	24
6-2 单通道数据采集软件延时控制采样率	25
6-3 单通道中断方式数据采集	26
6-4 多通道中断方式数据采集	27
实验 7 8098 单片机 PWM 输出实验	30
7-1 直接获得 PWM 输出	30

• II •

7-2	用 PWM 产生锯齿波	31
7-3	8098 片内 A/D 采样、PWM-D/A 输出、冻结显示	31
7-4	8098 片内 A/D 采样、PWM-D/A 输出、实时跟踪显示	33
实验 8	8098 单片机串行口应用实验	35
8-1	8098 单片机串行口方式 0 发送实验	35
8-2	8098 单片机串行口方式 0 键盘扫描实验	37
8-3	8098 单片机串行口方式 1 发送实验	39
8-4	8098 单片机串行口方式 1 两机通信实验	40
8-5	8098 单片机串行口方式 2 查询方式发送实验	41
8-6	8098 单片机与 PC 机的串行通信实验	42
实验 9	8098 单片机高速输入 HSI 实验	45
9-1	利用 HSI 测量时钟周期实验	45
9-2	利用 HSI 测量脉冲宽度实验	46
9-3	利用 HSI 中断方式统计输入脉冲数	47
9-4	HSI. 0 输入中断实验	49
9-5	HSI 引脚作为一般输入口应用实验	50
实验 10	8098 单片机高速输出 HSO 实验	51
10-1	利用 HSO 产生脉冲,用 HSO 中断方式	51
10-2	利用 HSO. 0 和 HSO. 1 输出相同波形,用 HSO 查询方式	52
10-3	HSO 作为一般输出口应用实验	52
10-4	用 HSO 定时启动 A/D,A/D 转换完成产生中断	53
实验 11	8098 单片机软件定时器应用实验	56
11-1	用软件定时器定时,P2. 5 口控制蜂鸣器产生报警信号	56
11-2	8098 片内 A/D 单通道数据采集,软件定时器控制采样率	57
实验 12	8098 单片机内部硬件定时器 T2 应用实验	60
12-1	用高速输出 HSO 控制蜂鸣器产生报警信号,用 T2 定时器定时	60
12-2	用 T2 对外部事件进行计数实验	61
实验 13	8098 单片机监视定时器(WDT)应用实验	63
13-1	应用 WDT 防止因外界干扰使主程序走飞实验	63
实验 14	8255 可编程并行 I/O 接口实验	65
14-1	8255 方式 0 对 LED 和开关的控制实验	65

14-2	8255 方式 0 对微型继电器的控制实验	66
14-3	8255 方式 0 对模拟电子开关的控制实验	67
14-4	8255 方式 0 对矩阵键盘的控制实验	68
实验 15	8253 可编程定时器/计数器实验	70
15-1	方波发生器实验	70
15-2	脉冲信号发生器实验	71
15-3	利用 8253 对外部事件进行计数实验	72
实验 16	8279 可编程键盘扫描显示控制器实验	74
16-1	8279 编码方式扫描键盘及 LED 发光二极管控制实验	74
16-2	8279 编码方式扫描键盘及数码管控制实验	75
16-3	8279 译码方式扫描键盘及 LED 发光二极管控制实验	79
实验 17	8155 可编程并行 I/O 接口/定时器实验	82
17-1	8155 内部 RAM 应用实验	83
17-2	8155 内部定时器应用实验	83
17-3	8155 并行口编程实验	83
实验 18	8259 可编程中断控制器实验	85
18-1	利用 8259 查询方式判别中断源	86
18-2	利用 8259 提供中断向量	87
18-3	2 片 8259 级连实验	88
实验 19	8251 可编程串行通信接口实验	90
19-1	8251 异步方式发送实验	90
19-2	8251 同步方式发送实验	91
实验 20	MC146818(HM6818)实时时钟芯片实验	93
20-1	用 MC146818 的 SQW 输出产生方波	93
20-2	用 MC146818 的 $\overline{\text{IRQ}}$ 产生定时中断请求信号	94
20-3	电子表编程实验(用 8279 控制 4 位数码管产生小时和 分,秒用小数点闪烁表示)	95
实验 21	DAC0832 数模转换器实验	99
21-1	DAC0832 输出模拟电压测试	99
21-2	利用 DAC0832 产生三角波和阶梯波	100
21-3	模拟心电信号发生器实验	101

实验 22 ADC0804 模数转换器实验	103
22-1 ADC0804 查询方式数据采集	104
22-2 ADC0804 中断方式数据采集	104
22-3 ADC0804 数据采集,8253 定时器中断方式控制采样率.....	105
22-4 ADC0804 数据采集,DAC0832 数模输出,示波器波形刷新显示实验	106
实验 23 8237 可编程 DMA 控制器实验	109
23-1 8237 控制 RAM 到 RAM 的 DMA 数据传送实验	109
23-2 8237 控制 RAM 到 D/A 的 DMA 数据传送,示波器波形刷新显示实验	110
23-3 8237 控制 A/D 到 RAM 的 DMA 数据传送实验	114
实验 24 TPμP-16A 微型打印机控制实验	117
24-1 用 8098 及扩展输出接口控制 TPμP-16A 打印字符	117
24-2 用 8255 控制 TPμP-16A 打印字符	119
24-3 用 8255 控制 TPμP-16A 打印图形及汉字	121
24-4 用 8255 控制 TPμP-16A 打印曲线	122
实验 25 PP40 四色绘图打印机控制实验	124
25-1 用 8155 控制 PP40 打印字符	124
25-2 用 8155 控制 PP40 绘制图形	125
25-3 用 8155 控制 PP40 绘制波形图	126
实验 26 MC6845 可编程 CRT 控制器实验	127
26-1 MC6845 控制 CRT 字符显示实验	127
26-2 8279 控制键盘输入字符,MC6845 控制 CRT 字符显示实验	129
实验 27 综合性实验及课程设计实验题目	131
参考文献	132

第 2 部分 附 录

附录 1 8098 单片机性能速查表	135
附录 2 MCS-96 指令系统速查表	148
附录 3 8098 单片机开发系统使用说明	158
附录 4 8255 可编程并行 I/O 接口使用速查表	177

附录 5	8253 可编程定时器/计数器使用速查表	180
附录 6	8279 可编程键盘扫描显示控制器使用速查表	182
附录 7	8155 可编程并行 I/O 接口使用速查表	186
附录 8	8259 可编程中断控制器使用速查表	189
附录 9	8251 可编程串行通信接口使用速查表	193
附录 10	MC146818 实时时钟芯片使用速查表	195
附录 11	DAC0832 数模转换器使用速查表	199
附录 12	ADC0804 模数转换器使用速查表	202
附录 13	8237 可编程 DMA 控制器使用速查表	204
附录 14	TPP-16A 微型打印机使用速查表	208
附录 15	PP40 四色绘图打印机使用速查表	210
附录 16	MC6845 可编程 CRT 控制器使用速查表	212
附录 17	TL-1A 型 8098/8031 单片机扩展接口学习机简介	215

第1部分 实 验

实验 1 8098 单片机开发系统操作实验

1. 实验目的

学习 8098 单片机开发系统的使用方法及基本命令。做此实验前,必须详细阅读附录 3“8098 单片机开发系统使用说明”。

2. 实验内容

1-1 屏幕编辑/汇编/单步执行操作实验

(1) 软件启动

将 8098 开发系统仿真板与 PC 机的 RS-232 串行通信口连接好,打开 8098 仿真板电源开关和 PC 机电源开关。

在操作系统状态下键入:“ATD96”,再按〈回车〉键,PC 机屏幕上显示主菜单如下:

ATD-96/98 MAIN MENU
Version 2. 20

- | | |
|------------------------|---------------|
| 1. Line Editor (EDLIN) | ;行编辑软件 |
| 2. Screen Editor (PE) | ;屏幕编辑软件 |
| 3. MCS-96 Assembler | ;MCS-96 汇编软件 |
| 4. MCS-96 Disassembler | ;MCS-96 反汇编软件 |
| 5. Debug | ;调试软件 |
| 6. Linker | ;连接软件 |
| 0. Return to DOS | ;返回 DOS |
- Choose[0-6]:

按数字键 2 则进入第 2 项屏幕编辑状态,屏幕上出现请用户输入文件名的提示如下:

PE File name[. ASM]:

输入文件名,再按〈回车〉键。如果该文件是磁盘里已有的文件,则屏幕上将显示该文件的程序清单;如果该文件是新建立的文件,则屏幕上提示为“New file”,并有如下显示:

=TOP of file=
=Bottom of file=

这时,系统处于屏幕编辑状态。

(2) 屏幕编辑

按〈ESC〉键,进入屏幕编辑命令状态,光标从屏幕左下角跳到左上角,此时用户可以使用屏幕编辑命令,输入或修改源程序。

输入以下源程序:

```
                ORG    2080H
LD      20H, #0000H
LD      26H, #6000H
LOOP: ST      20H, [26H]
LD      22H, 20H
LD      24H, 20H
INC     20H
SJMP    LOOP
END
```

程序输入完毕,检查程序有无错误,如有错误,用屏幕编辑命令进行修改;如无错误,则按〈ESC〉键,退出屏幕编辑命令状态,这时光标移回左下角。

按〈F2〉键再按〈回车〉键,将输入的当前文件存盘;按〈F4〉键,退出屏幕编辑状态,返回主菜单。或者按〈F3〉键再按〈回车〉键,存盘并退出屏幕编辑状态,返回主菜单。

(3) 汇编

在主菜单下,按数字键 3 选择第 3 项“MCS-96 Assembler”,进入汇编软件,可将你输入的源程序汇编成目标文件(机器码)。这时,屏幕上出现如下提示:

Source File name [.ASM]:

输入源程序名,并按〈回车〉键,屏幕上又出现如下提示:

List File name [.LST]:

输入列表文件名(一般与源文件同名),再按〈回车〉键。屏幕上又出现如下提示:

Object File name [.OBJ]:

输入目标文件名(一般与源文件同名),再按〈回车〉键。若源文件无错误,则汇编结束并显示:

9 Lines Passed. (9 行汇编通过)

Assembling Successful. (汇编成功)

而且还显示询问用户是否打印符号表的提示:

Label -Address output to printer(Y/N)?

若要打印符号表则打开打印机,键入 Y 并按〈回车〉键,即打印出符号表;若不需要打印,则键入 N,则返回主菜单。

若源程序有错误,汇编时屏幕上显示错误的行号及错误类型。汇编结束时,显示:

Assembling failure

以及

Save Error Message (Y/N)?

问是否存储错误信息。一般不需要存储,只要键入 N,返回主菜单,再重新进入编辑软件,对源程序进行修改。

(4) 联机仿真调试

此项操作请参阅附录 3 第 5 项。

在主菜单下,按数字键 5 选择第 5 项“Debug”,进入调试软件。

按〈F2〉键,执行 LOAD 命令,将 PC 机磁盘上的目标文件传送到 8098 仿真板的存储器中去。屏幕提示为:

File name [_.OBJ]:

输入目标文件名并按〈回车〉键,屏幕上出现如下提示:

0015 Loading OK. (15H 表示传送的字节数)

说明传送完毕。再按〈回车〉键,返回 Debug 命令状态。

如果传送不成功,屏幕上将出现通信错误的提示如下:

Reading Error on RS-232

这时需要按 8098 仿真板上的复位键(红色 RST 键),再按 PC 机的〈回车〉键,回到 Debug 命令状态下,重复上述按〈F2〉键的 LOAD 过程。

若传送成功,则可以进行单步执行程序。单步执行程序时屏幕左半部跟踪显示 5 个字寄存器单元的内容以及 PC,PSW 和 SP 的当前值,这样便于观察程序执行情况。按〈F10〉键即可单步执行程序,注意观察 20H,22H,24H 字单元中的内容为加 1 操作,26H 字单元中为地址 6000H 不变,28H 字单元没有使用,是一个随机数。按任意键可以停止单步执行。

按〈Alt〉+〈X〉键,可以退出 Debug,返回主菜单状态。

注意:如果 8098 仿真板电源未开或串行通信不成功,屏幕上将有红色提示:

Press RST Key on ATD - 96 then hit any key to restore.

即按 8098 仿真板上的复位键后,再按 PC 机的任一键,回到主菜单。

1-2 反汇编/设置断点/连续执行/存储器 检查操作实验

按照实验 1-1 的顺序输入如下程序并将其进行汇编:

```
ORG 2080H
LDB 20H, #00H ;赋值 0
```

	CLRB	21H	;清 0
	LD	28H, # 6000H	
LOOP:	STB	20H, [28H] ⁺	
	LDB	22H, 20H	
	ANDB	22H, # 0FH	;“与”运算
	LDB	23H, 20H	
	ORB	23H, # 0FH	;“或”运算
	LDB	24H, 20H	
	XOR	24H, # 0FH	;“异或”运算
	LDB	25H, 20H	
	NOTB	25H	;求反
	LDB	26H, 20H	
	NEGB	26H	;求补
	LDB	27H, 20H	
	SHRB	27H, # 1	;右移一位
	INCB	20H	;加 1
BRK:	DJNZ	21H, LOOP	;计数循环
HERE:	SJMP	HERE	

(1) 反汇编

将上述源程序经屏幕编辑和汇编后,返回 DOS 状态,用 DIR 和〈回车〉检查磁盘上是否有你的源文件和目标文件。假设你用的文件名为 TQY,则用命令“DEL TQY·ASM 〈回车〉”将你的源文件删除,目标文件仍保留。

键入命令“ATD 96〈回车〉”,进入主菜单。按数字键 4,即进入第 4 项“MCS-96 Disassembler”反汇编软件。这时屏幕提示如下:

Object Filename [.OBJ]:

输入目标文件名并〈回车〉,屏幕又提示:

Disassembler Filename [.ASM]:

输入反汇编后的源文件名并〈回车〉,则反汇编软件将你输入的目标文件的机器码转换成 MCS-96 汇编语言源程序,并以输入的反汇编后的源文件名存于磁盘上。屏幕提示:

Hit any key return main menu

即按任一键返回主菜单。这时你可进入文本编辑列出反汇编后的程序清单。当你将反汇编后的程序与源程序相对照时,可以发现反汇编后的程序标号都已不存在,而以绝对地址代替。

(2) 设置断点并连续执行

进入 Debug 状态,按〈F2〉键完成 LOAD 操作。

按〈F8〉键,屏幕上出现设置断点的提示,总共可设置 8 个断点。若你键入“BRK”并按〈回车〉键,就在上述程序的 BRK 标号的那一语句设置了一个断点。再按〈F8〉键,返回 De-

bug 命令状态。

按〈F9〉键,屏幕提示:

Start Address:

键入起始地址“2080”并按〈回车〉键(注意 2080 后不要打 H),程序从 2080H 单元开始连续执行,并到断点处停止,屏幕上显示断点语句及 20H~29H 寄存器中的内容,你可核对运行结果是否正确。

按〈F6〉键可以检查 1AH~FFH 所有寄存器中的内容,再按〈F6〉键返回 Debug 命令状态。按〈F5〉键可以检查所有特殊功能寄存器中的内容,再按〈F5〉键返回 Debug 命令状态。

按〈Alt〉+〈L〉键可以按输入的首地址显示存储器 128 个字节的内容。这时屏幕显示:

First Address:

键入“6000”并按〈回车〉键就可以检查从 6000H 开始的存储器中的内容。

按〈F2〉重新 LOAD 或者按〈F8〉后用空格键将设置的断点抹掉,断点可以消除。

(3) 连续执行

按〈F9〉键再键入起始地址并按〈回车〉键可以连续全速执行程序。由于上述程序执行结果看不到任何现象,程序执行到最后陷于“HERE”那一语句的死循环,必须按〈ESC〉键,屏幕上出现红色提示:

Press RST key on ATD-96 then hit any key to restore

即按 8098 仿真板上的复位键,再按 PC 机的任一键,则返回 Debug 命令状态。

同时按〈Alt〉和〈L〉键,可以检查存储器中的内容,屏幕提示:

First Address:

键入“6000”并按〈回车〉键,屏幕上将显示出 6000H 单元开始的存储器中的内容。检查这些内容是否与预期的程序执行结果相符合。

按〈F3〉键,修改存储器 6000H~6007H 单元中的内容,重新执行上述程序,再利用〈Alt〉+〈L〉键检查这些单元中的内容。

3. 思考题及选做题

- (1) 参照附录 3 第 2 项的说明,练习屏幕编辑的其他命令的使用方法。
- (2) 参照附录 3 第 5 项的说明,练习 Debug 的其它命令的使用方法。

实验 2 8098 单片机汇编语言程序设计实验

1. 实验目的

掌握 MCS-96 汇编语言指令的使用,进一步熟悉 8098 单片机开发系统的操作命令。

2. 实验内容

2-1 加减法程序设计

(1) 四字节无符号数加法

设计一个能做如下两个四字节无符号数的加法程序:

$$769F3256H + 556677CCH = CC05AA22H$$

程序设计方法:被加数装入 23H~20H 单元,加数装入 27H~24H 单元,结果装入 23H~20H 单元,将被加数冲掉。

参考程序如下:

```
ORG      2080H
LD        20H, # 3256H
LD        22H, # 769FH
LD        24H, # 77CCH
LD        26H, # 5566H
ADD       20H, 24H
ADDC      22H, 26H
HERE: SJMP HERE
END
```

标志说明:如果溢出标志 $V=1$,则说明发生溢出;溢出陷阱标志 $VT=1$,则说明有溢出发生过。ADDC 指令执行结果不能将零标志 Z 置 1。

(2) 四字节无符号数减法

设计一个能做如下两个四字节无符号数的减法程序:

$$769F3256H - 556677CCH = 2138BA8AH$$

程序设计方法:被减数装入 23H~20H 单元,减数装入 27H~24H 单元,结果装入 23H~20H 单元,将被减数冲掉。

参考程序如下: