



Weiji Yuanli yu
Jiekou Jishu Shiyān
ji Kecheng Sheji

微机原理与接口技术 实验及课程设计

杨斌
编著



西南交通大学出版社
[Http://press.swjtu.edu.cn](http://press.swjtu.edu.cn)

微机原理与接口技术 实验及课程设计

杨 斌 编著

西南交通大学出版社

· 成 都 ·

图书在版编目 (C I P) 数据

微机原理与接口技术实验及课程设计 / 杨斌编著.
成都: 西南交通大学出版社, 2005.10
ISBN 7-81104-169-3

I. 微... II. 杨... III. ①微型计算机-理论-高等学校-教学参考资料②微型计算机-接口-高等学校-教学参考资料 IV. TP36

中国版本图书馆 CIP 数据核字 (2005) 第 113325 号

微机原理与接口技术实验及课程设计

杨 斌 编著

*

责任编辑 黄淑文

责任校对 李 梅

封面设计 何东琳设计工作室

西南交通大学出版社出版发行

(成都二环路北一段 111 号 邮政编码: 610031 发行部电话: 028-87600564)

<http://press.swjtu.edu.cn>

E-mail: cbsxx@swjtu.edu.cn

四川森林印务有限责任公司印刷

*

成品尺寸: 185 mm × 260 mm 印张: 14.125

字数: 353 千字 印数: 1—3 000 册

2005 年 10 月第 1 版 2005 年 10 月第 1 次印刷

ISBN 7-81104-169-3/TP · 046

定价: 20.00 元

图书如有印装问题 本社负责退换

版权所有 盗版必究 举报电话: 028-87600562

前 言

“微机原理与接口技术”是计算机类及大多数电类专业学生的必修课程，也是一门实用性和操作性都很强的计算机硬件类课程。掌握这门课程的内容并具备一定的实际电路设计和编程能力，将会给学生深刻理解计算机软、硬件关系及在以后的计算机应用开发中打下很好的基础。另外，我国正在加快开发具有自主知识产权的计算机软、硬件产品，并进行大规模生产技术改造，因此，对既具备软件编程能力、又了解硬件知识的复合型人才的需求在不断提高。

在传统的教学过程中，由于条件的限制以及对实验课作用认识上的偏差，使得对实验课内容的组织、具体实验的实施都不是十分重视。尽管“微机原理与接口技术实验”是独立设课，但其目的仅仅是为了配合理论教学环节，实验内容也主要是为了验证理论课教学的内容。在当今大力提倡素质教育和创新能力培养为目标的教學理念下，必须重新认识实验课的地位和作用，强化实验内容，改进实验手段和方法，建立一套完整的认知体系，使整个实践环节起到它应有的作用。

经过多年的实践和探索，我们将微机原理与接口技术的实践环节分为两个阶段来实施：第一阶段为认知性入门阶段，主要安排了以验证性为主的硬件基本功能模块实验；第二阶段为综合与设计性的提高阶段，对此，我们自行开发了紧密结合理论教学内容的多项课程设计题目。

硬件基本功能模块实验安排了以理解掌握单元内容为目的的实验，实验过程基于实验器进行。实验器中各功能模块与 CPU 的数据线、地址线及读写控制线已在电路板上连接好，实验时只需连接片选以及输入/输出信号线，以减少因接线错误或接触不良造成的不成功率，从而增强学生完成实验的信心。初学者可将更多的注意力集中于对接口电路功能的理解、编程方法及接口电路工作中软硬件的配合。这部分实验内容基本按照理论教学进度安排，包括：基于 8255 的交通灯实验，基于 8253 的定时/计数器实验，基于 8279 的键盘显示实验，基于 8259 的中断实验，基于 0832 的 D/A 实验，基于 0809 的 A/D 实验，基于 8251 的串口通信实验，以及综合了前面所有功能器件的综合实验。

课程设计是实践性教学环节的重要内容，目的是培养学生的动手能力以及综合运用所学知识解决实际问题的能力，激发学生的创造性。因此，在具体选题和实现方式上力求做到：

- ① 与理论课教学内容相配合，并在实现技术和芯片运用知识方面作适当拓展，主要设计内容应是已学过或正在学习的内容，不致使学生感到无从下手。
- ② 设计和制作不再依赖实验器进行，而是自行焊接电路板或制作 PCB 板。
- ③ 设计分基本要求部分和发挥部分。基本部分要求不可过高，要让大多数同学经过努力在较短的时间内都可实现，从而树立同学的成就感和自信心。
- ④ 各部分设计题目之间既有一定的独立性又相互配合，设计内容循序渐进，后面的设计内容紧密结合前面的设计内容。

设计内容的组织形式为:

- (1) 基于 8088 的最小模式 (或最大模式) 硬件核设计。
- (2) 基于以上硬件核的最小系统设计 (结合存储器和外设地址译码)。
- (3) 在上述最小系统基础上实现的 8255 应用设计题目。包括 4×4 键盘和 4 位数码管显示电路 (结合 8255 内容)、 16×16 LED 点阵汉字显示电路。
- (4) 在上述最小系统基础上实现的 8279 键盘/数码管电路。
- (5) 在上述最小系统基础上实现的 0809 及 0832 A/D 和 D/A 电路。
- (6) 在上述最小系统基础上实现的 8253 和单级 8259 及 8250 串行接口电路。
- (7) 一个结合 PC 机编程的利用打印口的数据采集及控制电路。

在课程设计内容中, 为了引导同学的设计思路, 在开始的设计题目中对必要的设计原理进行了叙述, 并补充了一些硬件设计的知识、经验以及元器件选型方面的常识。在每个设计题目中, 还提供了各题目的参考电路设计和程序。

本教材所列的课程设计题目已经过几届学生的使用, 并且其中部分参考电路和参考程序也出自他们之手。学生的设计制作过程和完成情况表明, 这些题目内容都是比较合适的。学生在设计报告中表达的体会, 反映了通过这一实训环节他们在能力方面的提高以及成就感, 也更说明了这一环节的重要性。

由于编者的经验和水平所限, 书中难免会有不足和错漏之处。恳请本行同仁及读者指正。本人邮箱地址为: yangbin_1748@sina.com.cn。

作者

2005 年 9 月

目 录

1 实验器简介	1
1.1 实验器的组成结构	1
1.2 实验器的工作方式	2
1.2.1 单板计算机工作方式	2
1.2.2 PC 机串口工作方式	2
1.2.3 ISA 总线方式	2
1.3 实验器内部存储空间和 I/O 空间的分配	2
1.3.1 存储空间的分配	2
1.3.2 I/O 空间的分配	3
2 实验器的单板机工作方式及操作	4
2.1 键盘的定义及功能	4
2.1.1 键盘的排布	4
2.1.2 键盘的功能	4
2.2 键盘监控命令	6
2.2.1 键盘监控命令的定义及格式	6
2.2.2 键盘监控命令的使用方法举例	8
3 实验器的串行接口通信（监控）工作方式及操作	9
3.1 概 述	9
3.1.1 串行接口通信工作方式下的编程限制	9
3.1.2 操作步骤简述	9
3.1.3 串行监控方式与单板机方式的区别	9
3.1.4 软件快速入门	9
3.2 串行监控方式下的调试软件使用详解	10
3.2.1 系统主窗口	10
3.2.2 文件（File）	11
3.2.3 汇编编译（Masm）	13
3.2.4 加载/下载（UpLoad/Download）	15
3.2.5 运行（Run）	17
3.2.6 设 置（Settings）	19
3.3 串行监控命令	20

3.3.1	如何使用该选项	20
3.3.2	串行监控命令集	21
3.3.3	串行监控命令格式	22
3.3.4	串行监控命令的具体使用方法	22
4	硬件实验	25
4.1	基于 8255 的双色灯及开关量 I/O 实验	25
4.1.1	实验目的	25
4.1.2	开关状态输入及显示实验	25
4.1.3	模拟交通信号灯实验	26
4.1.4	8255 方式 1 下 B 口到 A 口的数据传输实验	28
4.2	定时/计数器 8253 的编程实验	31
4.2.1	实验目的	31
4.2.2	8253 定时方式下分频系数的设置及通道级联应用实验	31
4.2.3	8253 计数方式实验	33
4.3	键盘/显示控制器 8279 编程实验	35
4.3.1	实验目的	35
4.3.2	实验内容和要求	35
4.3.3	实验说明	36
4.3.4	实验电路及实验连线	36
4.3.5	验证性实验参考程序	37
4.4	中断控制器 8259 的应用及编程实验	39
4.4.1	实验目的	39
4.4.2	实验内容及要求	40
4.4.3	实验说明	40
4.4.4	实验电路及实验连线	41
4.4.5	验证性实验参考程序	42
4.5	D/A 变换实验	45
4.5.1	实验目的	45
4.5.2	实验内容及要求	45
4.5.3	实验说明	45
4.5.4	实验电路及连线	46
4.5.5	实验参考程序	46
4.6	A/D 变换实验	47
4.6.1	实验目的	47
4.6.2	实验内容及要求	47
4.6.3	实验电路及实验连线	48
4.6.4	实验参考程序	48
4.7	多功能数据采集系统综合设计实验	50

4.7.1	实验目的	50
4.7.2	实验内容及要求	50
4.7.3	实验电路及实验连线	50
4.7.4	程序参考流程图	51
4.8	8251 可编程串行接口通信实验	52
4.8.1	实验目的	52
4.8.2	实验内容及要求	52
4.8.3	实验原理图及实验说明	52
4.8.4	实验电路及实验连线	53
4.8.5	实验参考程序	53
5	综合实验及课程设	55
5.1	8088 最小模式下的小系统设计	55
5.1.1	最小模式下的最简结构硬件核设计	55
5.1.2	采用最简硬件核实现的简单微机系统	62
5.2	FLASH ROM 的接口电路设计及在线编程	72
5.2.1	FLASH ROM 简介	72
5.2.2	FLASH ROM 的编程特点	72
5.2.3	FLASH ROM 读写 ID 号操作的进入和退出	73
5.2.4	FLASH ROM 的擦除	73
5.2.5	FLASH ROM 数据保护的无效化编程	74
5.2.6	FLASH ROM 的写入编程	74
5.2.7	FLASH ROM 在线编程扩展电路及参考程序	75
5.3	采用 8255 的 4×4 键盘及 8 位数码管接口电路设计	78
5.3.1	设计要求	78
5.3.2	设计思想	78
5.3.3	整体电路设计	84
5.3.4	参考电路	84
5.3.5	程序设计	86
5.4	采用 8279 的键盘及显示接口电路设计	91
5.4.1	设计要求	91
5.4.2	电路设计思想及实现	91
5.4.3	程序设计思想及实现	93
5.5	LED 点阵汉字显示屏的接口设计	114
5.5.1	设计要求	114
5.5.2	LED 点阵显示器的结构及汉字显示原理	114
5.5.3	显示电路的总体结构设计及工作原理	116
5.5.4	核定驱动器的驱动能力	116
5.5.5	参考电路	117
5.5.6	实现汉字显示的程序流程框图	117

5.5.7 参考程序	118
5.6 点阵型液晶显示器的接口设计	121
5.6.1 点阵型液晶显示器简介	121
5.6.2 12864 点阵型液晶显示器的显示原理	122
5.6.3 12864 点阵型液晶显示器的内部编程结构及外部引脚	123
5.6.4 12864 点阵型液晶显示器的接口电路设计	124
5.6.5 12864 点阵型液晶显示器的编程	125
5.7 A/D、D/A 的接口电路设计	135
5.7.1 设计要求	135
5.7.2 设计思想	136
5.7.3 参考电路	137
5.7.4 参考程序	137
5.8 基于 8088 最小系统的定时、中断和串行接口电路设计	139
5.8.1 设计内容及要求	139
5.8.2 电路原理图及设计思想描述	139
5.8.3 MAX232 使用简介	141
5.8.4 参考电路	143
5.8.5 调试用参考程序	143
5.9 基于打印口的 A/D 和 D/A 接口电路设计	148
5.9.1 设计内容及要求	148
5.9.2 PC 机打印口简介	148
5.9.3 电路设计思想及参考电路	149
5.9.4 软件编程思想及参考程序	151
附录一 8279 键盘/显示控制器的系统组成结构与编程	169
一、8279 的内部编程结构及工作原理	169
(一) 8279 的内部编程结构	169
(二) 8279 的外部引脚及功能	170
(三) 8279 的外部电路构成原理	171
二、8279 的工作方式及编程	172
(一) 键盘控制部件的结构及工作方式	172
(二) 显示控制部件的结构及工作方式	174
(三) 8279 的编程	176
(四) 8279 的命令字	177
(五) 8279 的状态寄存器(状态字)	178
附录二 实验器中的实验模块	180
一、频率源模块	180
二、可读写存储器模块	180

三、可编程并行口 8255 模块	180
四、双色 LED 灯模块	182
五、单色 LED 灯模块	182
六、参考电压模块	182
七、电位器模块	183
八、分频器模块	183
九、单脉冲电路模块	184
十、8279 键盘及 LED 数码显示模块	184
十一、A/D 转换器模块	185
十二、D/A 转换器模块	185
十三、8251 可编程串行通讯模块	186
十四、可编程定时/计数器模块	186
十五、I/O 译码电路模块	187
十六、8259 中断控制器模块	187
十七、开关量输入模块	188
附录三 调试程序 DEBUG	189
一、DEBUG 的功能	189
二、DEBUG 中的参数说明	189
(一) DEBUG 命令的一些通用信息	189
(二) DEBUG 命令中的地址和地址范围参数	189
(三) DEBUG 支持的语法和规则	190
(四) DEBUG 的启动	191
三、DEBUG 的常用命令	191
(一) D 显示内存单元内容命令	191
(二) E 修改内存单元内容命令	192
(三) R 显示和修改寄存器内容命令	192
(四) A 汇编命令	193
(五) U 反汇编命令 (Unassemble Command)	194
(六) G 执行程序命令	195
(七) T 跟踪程序执行命令 (Trace Command)	195
(八) N 设置文件名命令	196
(九) L 装入磁盘文件命令 (Load Command)	196
(十) W 写磁盘文件命令	197
附录四 8256 异步串行通信控制器系统组成结构与编程	198
一、8250 的内部编程结构及工作原理	198
(一) 8250 的内部编程结构	198
(二) 8250 的数据串行收发工作原理	198

(三) 8250 的引脚功能	199
(四) 8250 的内部寄存器	201
(五) 8250 的内部寄存器寻址	206
二、8250 的编程	207
(一) 8250 的初始化编程	207
(二) 8250 的数据发送编程	207
(三) 8250 的数据接收编程	208
三、PC 微机提供的串口通信系统服务	210
(一) BIOS 调用的 INT 14H 服务	211
(二) DOS 调用的 INT 21H 服务	212
附录五 ASCII 码表	213
附录六 PC 机键盘扫描码表	215

1 实验器简介

1.1 实验器的组成结构

本实验器由许多独立的硬件实验模块组成，用户可用它们组成各种各样的硬件实验。主要包括频率源模块、存储器 RAM 模块、DMA 模块、8255 并行口模块、双色灯模块、单色灯模块、参考电压模块、键盘及显示模块（8279）、分频器模块、A/D 转换模块、D/A 转换模块、串行通讯 8251 模块、定时/计数器 8253 模块、8259 中断控制器模块、8088 CPU 及监控模块、单脉冲触发模块和地址译码模块等 20 几种模块。实验器的内部组成结构原理图如图 1.1 所示。

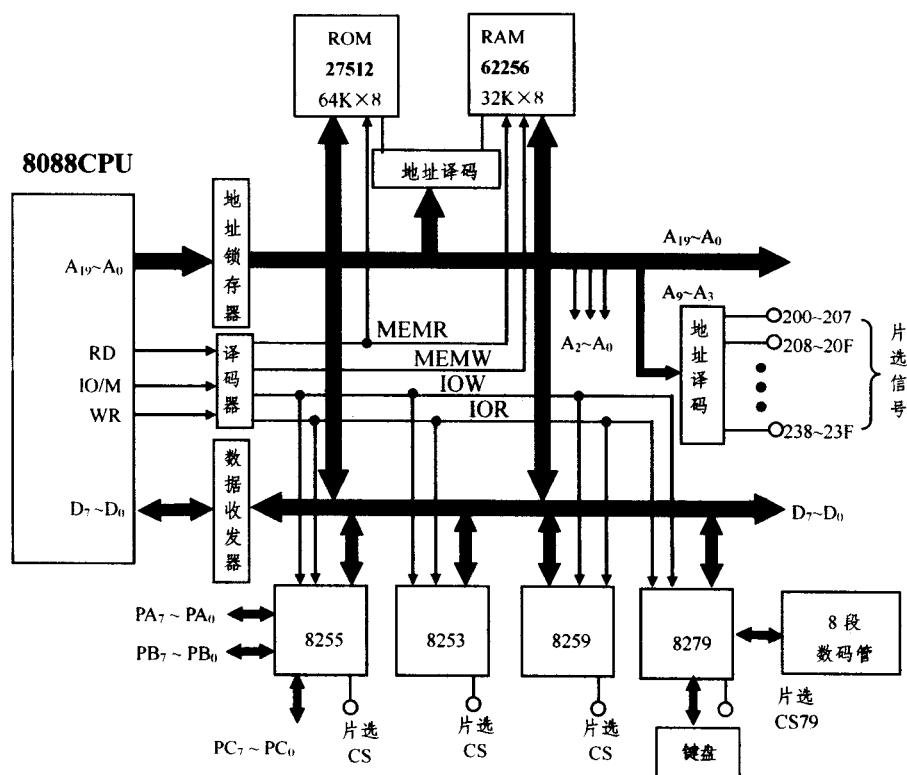


图 1.1 实验器的内部组成结构原理图

为了减少一些不必要的重复连线 and 减少连线错误，实验器内部已将 CPU 与各模块之间的地址线 and 数据线连接好，实验中只需连接主要的控制信号、片选信号和 I/O 接口信号。这些信号的连接点通过板上的“O”形圆孔接出，用于连接硬导线或作为测试点。

1.2 实验器的工作方式

本实验器以 8088 CPU 最小模式组成系统, 配置有 64K 的 EPROM 存放监控程序, 32K 的 RAM 存放实验程序。实验器有三种工作方式可供选择:

1.2.1 单板计算机工作方式

这种方式不需要另外的计算机支持, 实验器自成系统 (构成 1 单板计算机)。系统加电后, 实验板上的 8088 CPU 将运行固化在 64K 字节的 EPROM (27C512) 中的监控程序, 实现对实验器上资源的管理和控制。实验时直接由实验器的小键盘将机器码程序输入到 RAM 中并在其中运行。这是一种最贴近计算机硬件的实验方式, 尽管机器码程序的录入过程比较烦琐。但是通过这种方式可以使同学直观地了解机器码程序的指令格式以及在计算机内存中的具体存放、内存中绝对地址和相对地址的概念、字数据的存放顺序等内容。

单板计算机工作方式下的具体操作在第 2 章详细说明。

1.2.2 PC 机串口通信方式

这种方式是用串口电缆把 PC 机和实验器连接起来, 在 PC 机上运行一个综合实验管理程序, 用于编辑、编译和链接源程序并通过串口将最终的执行程序传送到实验器的 RAM 中去运行, 实现 PC 机对实验器实验过程的远程操控。

PC 机串口通信方式下的具体操作在第 3 章详细说明。

1.2.3 ISA 总线方式

由于本实验只用到了前两种工作方式, 所以对这种方式不做过多的介绍。

1.3 实验器内部存储空间和 I/O 空间的分配

1.3.1 存储空间的分配

8088 尽管有 1 兆字节空间的寻址能力, 而本实验器在单板机和串行监控方式下, 理论上提供给用户的可使用 RAM 空间为: 00000H~6FFFFH, 80000H~FFFFFFH, 包括总清入口共 960K 存储空间, 外加 64K 的系统程序存储器 ROM 空间。但实际上, 实验器在满足所有实验内容需要的前提下, 只在实验板上放置了 64K 字节的 EPROM 和 32K 字节的 RAM。

其中, 64K EPROM 只读存储器地址空间为 70000H~7FFFFH, 用于存放固化的实验器监控程序, 用户不可占用。

另外的 32K RAM 对应的地址空间为 80000H~87FFFH, 用于存放用户的实验程序和数据。(在 ISA 总线方式下, 这段区间被调整到 D0000H~D7FFFFH 地址空间)。

需要说明的是, 地址空间 80000H~800FFH 与 00000H~000FFH 是重合的, 它们在内部硬件电路连接上都对应实验器内相同的 RAM 区, 即通过程序读写 80000H~800FFH 地址空间时, 也就是读写 00000H~000FFH 地址空间, 反之亦然。所以此区间可用于建立系统和用

户的中断向量表。其中监控程序占用 80000H~80013H (00000H~00013H) 5 个类型号共 20 个字节，作为单步、断点、无条件暂停等系统功能的中断向量，而其后的空间则可用于存放用户的中断向量。

另外，80100H~802FFH 为系统监控程序所需要的数据区和堆栈区，所以向实验器键入（单板计算机工作方式）或载入（PC 机串口通信方式）用户编写的实验程序时，起始地址需大于 80300H。通常采用从 81000H 开始，即段地址为 8100H，偏移地址为 0000H 的 RAM 区开始存放用户程序。

这 96K 存储空间的分配情况如图 1.2 所示。

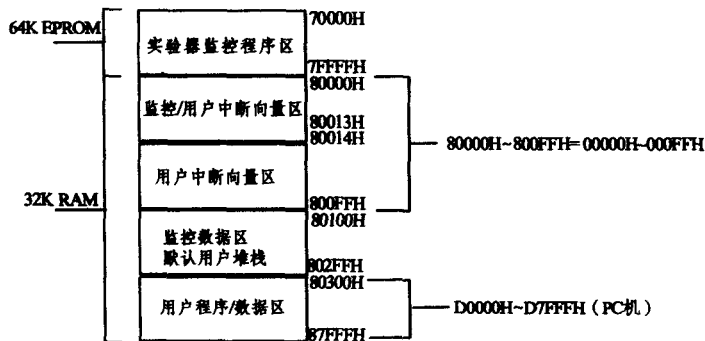


图 1.2 存储空间分配表

1.3.2 I/O 空间的分配

8088 CPU I/O 指令可操纵的 0000H~FFFFH 共 64K I/O 空间都可由用户使用，但实验器内部译码器已译码的 I/O 地址范围为 200H~23FH。它们在实验器上分别对应 8 个引出点（插孔），可直接作为片选信号使用。共分为：200H~207H、208H~20FH、210H~217H、218H~21FH、220H~227H、228H~22FH、230H~237H、238H~23FH 等 8 个独立的 I/O 地址空间。

2 实验器的单板机工作方式及操作

在单板机方式下，实验器自成系统。系统加电后，实验板上的 8088 CPU 将运行固化在 64K 字节的 EPROM (27C512) 中的监控程序，实现对实验器上资源的管理和控制。用户通过实验器内 24 键的键盘直接将十六进制的机器码程序（由宏汇编程序 MASM.EXE 对源程序编译后产生的列表文件*.LST 文件中提供）输入到实验器的 RAM 存储器中，然后再使用键盘监控命令操作实验器内的实验程序的运行，并且可以选择连续运行、单步运行或设置断点的分段运行。

实验器在单板配置方式下，通过 8 个 LED 数码管显示输入过程中的存储器地址信息和机器码数据。8 个数码管按 4 个字符一组被分成两组：左边一组为“地址段”，右边一组为“数据段”。所有显示都用十六进制给出。

本章着重介绍键盘监控命令的含义和使用方法，以及如何利用键盘监控命令操作实验器上的硬件实验。

将电源开关打开后，LED 数码管应显示“AEDK8688”字样。按实验器小键盘上的任意键（RST 键除外），即进入键盘监控状态，键盘监控状态提示符为“-”。

2.1 键盘的定义及功能

2.1.1 键盘的排布

键盘的排布如图 2.1 所示。

EB/AX 0	ER/BX 1	GO/CX 2	ST/DX 3	IB/SP 4	OB/BP 5	MV/SI 6	EW/DI 7
IW/CS 8	OW/DS 9	SS A	ES B	IP C	FLAG D	E	F
.	,	-	+	:	REG	DMA HALT	RST

图 2.1 实验器的键盘排布

2.1.2 键盘的功能

在键盘程序控制下，用户可以通过按下键盘上的键来输入命令和数据（通过显示来进行用户和监控程序之间的通讯）。键盘共有 24 键，上面两排为十六进制数字键，第三排为 8 个功能键。

0~D 共 14 个十六进制数字键均是复合功能键，其功能符号印在键上，即十六进制数上面的英文字母是监控命令（斜线左边）和寄存器名（斜线右边）的首字母缩写。表 2.1 是 14 个复合功能键上字符的说明及功能，表 2.2 是其他功能键操作说明。

表 2.1 复合键的功能说明表

十六进制键	操 作 命 令		寄 存 器	
	缩 写 符	功 能	缩 写 符	名 称
EB/AX 0	EB	字节读写	AX	AX 累加器
ER/BX 1	ER	寄存器读写	BX	BX 基址寄存器
GO/CX 2	GO	执行程序	CX	CX 计数寄存器
ST/DX 3	ST	单步运行	DX	DX 数据寄存器
IB/SP 4	IB	I/O 口输入字节	SP	堆栈指针寄存器
OB/BP 5	OB	I/O 口输出字节	BP	基址指针寄存器
MV/SI 6	MV	成组搬迁	SI	源变址寄存器
EW/DI 7	EW	字读写	DI	目的变址寄存器
IW/CS 8	IW	I/O 口字输入	CS	代码分段寄存器
OW/DS 9	OW	I/O 口字输出	DS	数据分段寄存器
SS A			SS	堆栈分段寄存器
ES B			ES	扩展分段寄存器
IP C			IP	指令指针寄存器
FLAG D			FLAG	状态标志寄存器

表 2.2 其他功能键操作说明表

功能键	操 作
.	· (句号) 键是命令的终止符。当按下此键时，当前命令就被执行。注意：使用 GO 命令时，按下此键就开始执行指定地址处的程序。平时或出错时，按下此键时，返回监控，显示提示符 “-”
,	， (逗号) 键用来分隔键盘项目，并用来将地址段增量，以指向相邻的下一个存储单元
-	- (负号) 键允许用户从一个十六进制数中减去另一个十六进制数
+	+ (正号) 键允许用户作两个十六进制数的加法
:	: (冒号) 键将要输入的地址分成两部分：段地址和偏移量，还用来在 EB/EW 命令中作地址减量 (-1 或 -2)
REG	REG 键。允许用户使用任意一个寄存器中的内容作为一个地址或数据项
DMA HALT	DMA 为 DMA 实验请求键 (ISA 配置方式使用)。HALT 无条件暂停键 (串行监控和单板配置方式使用)，中断当前的活动，并保护所有寄存器的内容，返回监控，等待用户输入命令
RST	系统复位键。允许用户终止任何当前的活动，返回监控

2.2 键盘监控命令

2.2.1 键盘监控命令的定义及格式

以下键盘监控命令必须在系统处于监控命令状态下进行（最左端数码管显示“-”）。系统复位后按任意键将进入监控命令状态。

（1）机器码的输入、修改和显示命令（EB 键）

使用方法：先按 EB 键，输入 4 位十六进制存储器段地址（建议值为 8100H）；按“:”键，输入 4 位十六进制存储器偏移地址；最后按“,”键。左 4 位数码管将显示当前存储单元的偏移地址，最右端 2 位数码管将显示该存储单元的十六进制内容。

使用说明：地址由段和偏移两部分组成，用“:”作分隔符，如果缺省段值，则该地址的段值为代码段 CS 中的当前值。输入地址后按下“,”，在地址段中显示段地址，2 s 后在地址段显示偏移地址，并在数据段显示该地址单元的内容。键入“,”，使地址加 1 再显示；键入“:”，使地址减 1 再显示；键入十六进制数据后再键入“,”或“.”，则将改写读单元内容；键入“.”则返回监控；键入其他键则出错，在地址段显示“-ERR”。

（2）显示/修改存储字单元命令（EW 键）

使用方法：具体操作同 EB，只是按“,”和“:”时地址的增量或减量为 2。另外，最右端 4 位数码管将显示存储器指定地址的字内容。

（3）显示/修改寄存器内容命令（ER 键）

使用方法：先按 ER 键，再按指定的寄存器符号键（最右 4 位数码管将显示指定寄存器当前值），然后输入新内容并按“,”键确定。

使用说明：若要查看所有寄存器内容，可先按 ER 键，再按 AX 寄存器符号键，此后不断键入“,”，则将按照 AX、BX、CX、DX、SP、BP、SI、DI、CS、DS、SS、ES、IP、FL 次序依次显示下一个寄存器内容，直到 FL 寄存器为止，不循环。寄存器若键入数据再按“,”或“.”，则该寄存器的内容被修改；若键入“.”，修改后返回键盘监控；按其他键出错，地址段上显示“-ERR”，并返回监控。

（4）从 I/O 端口读入字节或字数据命令（IB 键）

使用方法：先按 IW 键，再输入 I/O 端口地址，按“,”键确认。右端数码管将显示该端口的当前内容。

使用说明：若从指定的输入端口地址输入一个 8 位的字节或 16 位的字显示在数据段后，继续键入“,”，将再次从该口读入数据，并显示在数据段上。由于 I/O 端口地址最大寻址范围为 64K，因而对于端口地址不允许用段值。

（5）输出字节或字数据到输出口命令（OB 键）

使用方法：先按入 OB 键，再输入 I/O 端口地址，按“,”键，接着输入数据，最后按“,”键确认。