

朱世鸿 编

# 微机接口和 IBM-PC 编程应用实验

国科学技术大学出版社

1

D36  
ZSH/1

# 微机接口和IBM-PC编程

## 应用实验

朱世鸿 编



中国科学技术大学出版社

1990·合肥

1019464

**微机接口和 IBM-PC 编程应用实验**

责任编辑: 黄 德      封面设计: 王瑞荣

\*

中国科学技术大学出版社出版

(安徽省合肥市金寨路 96 号)

中国科学技术大学印刷厂印刷

安徽省新华书店发行

\*

开本: 787×1092/16 印张: 17.75 字数: 424 千

1990 年 9 月第 1 版      1990 年 9 月第 1 次印刷

印数: 1—5000 册

ISBN7-312-00169-6/TP·16 定价: 3.50 元

# JS250/50

## 内 容 简 介

本书着重讨论 I/O 可编程接口芯片的使用，并系统介绍了目前通用的内、外总线结构，书中阐述的原则，不仅适用于 INTEL8086/8088 CPU 系统，也可用于其它类型的 CPU 系统中。

本书还详细介绍了 IBM PC 微机的系统功能调用，对汇编语言上机操作和高级语言对汇编语言的调用进行了系统阐述，并给出了实例。

本书内容充实，由浅入深，实用性强，每个实验还附有实际意义的思考题，以巩固基本概念的掌握。

本书可作为大专院校和各类微机培训班的实验教材，也可供从事微机开发和应用的工程技术人员参考。



## 前 言

微处理器的出现是70年代大规模集成电路发展的重大成果之一。其应用范围从科学计算、工农业实时控制、到航天和军事领域。由于16位微处理器的功能远超过八位微处理器，在一定程度上可与小型机相媲美。所以，16位微处理器在国际上得到广泛的应用，我国对16位微处理器的研究和应用也日臻成熟。

现在，国内高校的许多专业开设了微机原理及其实验课程，但有关16位微机的实验指导书尚不多见，为此，该书是在总结了多年教学经验的基础上编写而成。

编写本书的指导思想是着重微机的实际应用。因此了解I/O接口与外界进行各种信息交换的方法和合理设计接口是本书的重点。每个实验均强调其基本概念和基本用法，使初学者能正确运用基本理论，设计出实际的应用系统。

本书还向读者介绍了IBM-PC DOS的中断和功能调用，系统阐述了微机原理和实现接口的基本工具——汇编语言上机编写工作的全过程，并提供了一些程序设计的实例。

为了培养学生独立的实验技能，本书省略了常规的实验步骤（指导教师可视学生的实际情况予以引导）。书中提供的参考程序和硬件设计方法不一定是最佳方案。二十三个实验均可在SDK-86, TP-86单板机上实现。

本书编写过程中得到了孙适副教授、景中起副教授的支持；沈兰荪教授、曾绍统教授、汪炳权副教授对本书原稿进行了悉心审阅，第五章—高级语言与汇编于程序的连接由郝炳焜、翟锐锋同志执笔。在此谨致谢意。

限于编者水平，书中缺点和错误在所难免，恳请读者批评指正。

编 者  
1989年8月于合肥

# 目 录

## 前 言

第一章 KD86/88 单板微型计算机简介和使用方法 .....	( 1 )
§ 1.1 主要技术指标 .....	( 1 )
§ 1.2 KD86/88 单板机原理简介 .....	( 2 )
§ 1.3 KD86/88 单板机使用方法 .....	( 6 )
第二章 KD86/88 单板机实验 .....	( 19 )
实验一 单板机键盘操作 .....	( 19 )
实验二 程序调试及代码转换 .....	( 28 )
实验三 四则运算 .....	( 33 )
实验四 数字滤波 .....	( 39 )
实验五 8255 和 LED 显示器接口 .....	( 42 )
实验六 交通灯控制 .....	( 45 )
实验七 A/D 转换 .....	( 49 )
实验八 D/A 转换 .....	( 57 )
实验九 用 D/A 在示波器上显示图形 .....	( 64 )
实验十 EPROM, EEPROM 编程和 RAM 掉电保护 .....	( 68 )
实验十一 CPU 和打印机通讯 .....	( 76 )
实验十二 8251A 基本性能实验 .....	( 79 )
实验十三 8251 双机通讯 .....	( 82 )
实验十四 8253 基本性能实验 .....	( 86 )
实验十五 用 8253 实现频率测试 .....	( 90 )
实验十六 8259 中断实验 .....	( 94 )
实验十七 8279 键盘/显示接口实验 .....	( 97 )
实验十八 8086/8088 中断实验 .....	( 102 )
实验十九 磁带转储 .....	( 105 )
实验二十* 8253, 8255, 8259 离板扩充设计 .....	( 110 )
实验二十一* 用 8255 实现 IEEE-488 接口设计 .....	( 112 )
实验二十二* IBM-PC 全双工通讯设计 .....	( 114 )
实验二十三* DMA 数据采集系统设计 .....	( 125 )
(一) 归一化 .....	( 126 )
(二) 放大器 .....	( 127 )
(三) 滤波器 .....	( 129 )
(四) 模拟开关 .....	( 133 )
(五) 取样保持电路 .....	( 135 )

(六) A/D 转换 .....	(137)
(七) D/A 转换 .....	(142)
(八) DMA 控制器 .....	(143)
(九) 系统噪声干扰的抑制 .....	(144)
<b>第三章 IBM-PC DOS 使用指南 .....</b>	<b>(150)</b>
§ 3.1 IBM-PC 操作系统简介 .....	(150)
一、PC-DOS 简介 .....	(150)
§ 3.2 系统调用和中断处理 .....	(152)
一、系统功能调用类型 .....	(152)
1. I/O 设备功能调用 .....	(152)
2. 文件操作功能调用 .....	(154)
3. 目录操作功能调用 .....	(156)
4. 其它功能调用 .....	(156)
5. DOS 3.x 增加功能调用 .....	(157)
6. 功能调用错误代码信息 .....	(159)
7. 使用功能调用注意的问题 .....	(161)
二、中断调用 .....	(162)
1. INT 5H 打印屏幕 .....	(162)
2. INT 10H CRT 屏幕显示 .....	(162)
3. INT 13H 软盘 I/O .....	(164)
4. INT 13H 硬盘 I/O .....	(165)
5. INT 14H 串行 I/O 通讯 .....	(165)
6. INT 16H 键盘 I/O .....	(166)
7. INT 17H 打印机 I/O .....	(166)
8. INT 20H DOS 程序终止 .....	(167)
9. INT 21H DOS 功能调用 .....	(167)
10. INT 25H DOS 磁盘读 .....	(167)
11. INT 26H DOS 磁盘写 .....	(167)
§ 3.3 PC-DOS 常用命令键 .....	(167)
一、启动主机 .....	(167)
二、编辑键 .....	(167)
三、常用键 .....	(167)
<b>第四章 汇编语言程序上机操作的过程 .....</b>	<b>(169)</b>
§ 4.1 汇编编辑程序简介 .....	(169)
一、WORDSTAR 编辑程序 .....	(169)
二、MINCE 编辑程序 .....	(172)
§ 4.2 汇编语言编辑过程 .....	(175)
§ 4.3 汇编语言程序设计举例 .....	(178)
§ 4.4 汇编错误信息提示 .....	(187)

§ 4.5	动态调试程序 DEBUG .....	(192)
§ 4.6	8086 指令系统简介 .....	(204)
第五章	高级语言与汇编子程序的连接 .....	(210)
§ 5.1	混合编程原理简介.....	(210)
§ 5.2	BASIC 程序调用汇编子程序 .....	(211)
§ 5.3	FORTRAN 程序调用汇编子程序 .....	(215)
§ 5.4	C 程序调用汇编子程序.....	(219)
§ 5.5	PASCAL 程序调用汇编子程序 .....	(224)
第六章	附 录 .....	(227)
§ 6.1	可编程接口芯片方式命令字简介.....	(227)
一、	8237 高性能可编程 DMA 控制器 .....	(227)
二、	8251A 可编程通讯接口 .....	(230)
三、	8253 可编程定时计数器 .....	(232)
四、	8255A 可编程并行 I/O 接口 .....	(234)
五、	8259A 可编程中断控制器.....	(237)
六、	8279 可编程键盘/显示器接口 .....	(240)
七、	8250 可编程通讯接口 .....	(242)
§ 6.2	总线标准.....	(245)
一、	S-100 总线 .....	(245)
二、	多总线(MULTIBUS) .....	(250)
三、	STD 总线 .....	(253)
四、	IEEE—488 总线 .....	(256)
五、	RS—232C 串行接口总线 .....	(258)
六、	IBM—PC 总线 .....	(261)
§ 6.3	A/D 和 D/A 转换器特性解释.....	(263)
一、	A/D 转换器 .....	(263)
二、	D/A 转换器 .....	(264)
§ 6.4	常用集成电路电气性能和引脚简介.....	(265)
一、	TTL 系列 .....	(265)
二、	MOS 系列 .....	(266)
三、	色标电阻 .....	(266)
四、	电容标志 .....	(267)
五、	集成电路引脚简介 .....	(268)



# 第一章 KD86/88 单板微型计算机 简介和使用方法

KD86/88 单板微型计算机的中央处理单元是 INTEL 8086 和 INTEL 8088 CPU (该机的设计吸取了国内外同种类型机器的优点)。其中 A 型机 CPU 工作为最小模式, B 型机 CPU 工作为最大模式。兼容了 INTEL 公司的 SDK-86 和国产 TP-86 单板机的全部软、硬件功能。

MMS-KD (MULTIMICROPROCESSOR SYSTEM) 是 KD86/88 的改进型, 其功能键在 KD86/88 的基础上进行了扩充, 操作更为方便。结构上只需更换少量器件, 便满足用户对 8086, 8088, 8096, 8098 和 MCS51, HD64180, Z80, 8085CPU 的在线一次开发成功的要求。特别适合大专院校学生做各种规模的实验 (可提供学生实验板, 重复焊接次数大于百次, 同时为用户提供多种功能板, 加快开发速度)。本书是专为 8086 和 8088 CPU 而编写的实验指南, 其它 CPU 系统, 也编写了类似本书的实验。

下面我们介绍 KD86/88 A 单板微型计算机的功能。

## § 1.1 主要技术指标

### (一) 中央处理器(CPU)

KD86/88 单板微型计算机 CPU 分别是 INTEL 8086 和 INTEL 8088。工作模式为最大和最小两种型号, 时钟为 5/2.5MHz。

### (二) 存储器

1. 板上提供 32K 字节的 ROM, 可立即扩展为 64K 字节, 其中 0FE000H~0FFFFFH 为监控存储区(8088 CPU), 0FDFFFH~0F8000H 为用户工作区。

2. 板上提供 32K 字节的 RAM, 具有掉电保护功能, 可立即扩展为 128K 字节, 其中 00H~0FFH 为监控工作区, 100H~7FFFH 为用户工作区。

### (三) 中 断

1. 外部中断: 用户能使用屏蔽中断。NMI 已用于“INTR”功能键。

2. 内部中断: 中断 1, 3 为监控程序使用。

### (四) EPROM 编程

对 2716-27256 EPROM 和 E<sup>2</sup>PROM 具有快速检测, 分区和整片编程功能, 不需外接编程电压。

### (五) I/O 可编程接口

1. 串行 I/O 接口: 串行接口由 INTEL 8251A 构成标准 RS-232C 接口。

2. 并行 I/O 接口: 并行接口由两片 INTEL 8255 构成, 可进行 16 位数据或 8 位数据的并行传输。

3. 定时/计数接口: 定时/计数接口由 INTEL 8253 构成, 其三个通道全供用户使用。
4. 中断控制接口: 中断控制接口由 INTEL 8259 构成, 可以单独地处理 8 级优先权的中断请求。
5. DMA 控制接口: DMA 接口由 INTEL 8237 构成, 其 4 个通道全供用户使用。
6. 键盘/显示接口: 键盘/显示接口由 INTEL 8279 构成, 板上设置了 24 个按键, 8 只八段 LED 显示器。
7. 提供 8087 数值协处理器插座, 由用户根据需要自选。
8. 用户通过 I/O 接口可配接盒式磁带机、打印机、绘图仪等设备。

#### (六) 总线扩展

考虑用户离板扩展系统的需要, 提供了和 TTL 电平完全兼容的地址、数据和控制总线。

## § 1.2 KD 86/88 单板机原理简介

KD86/88A 单板微型计算机硬件结构框图见图 1.2.1, 其整机硬件电路主要由 CPU, 存储器和 I/O 接口三部份组成。

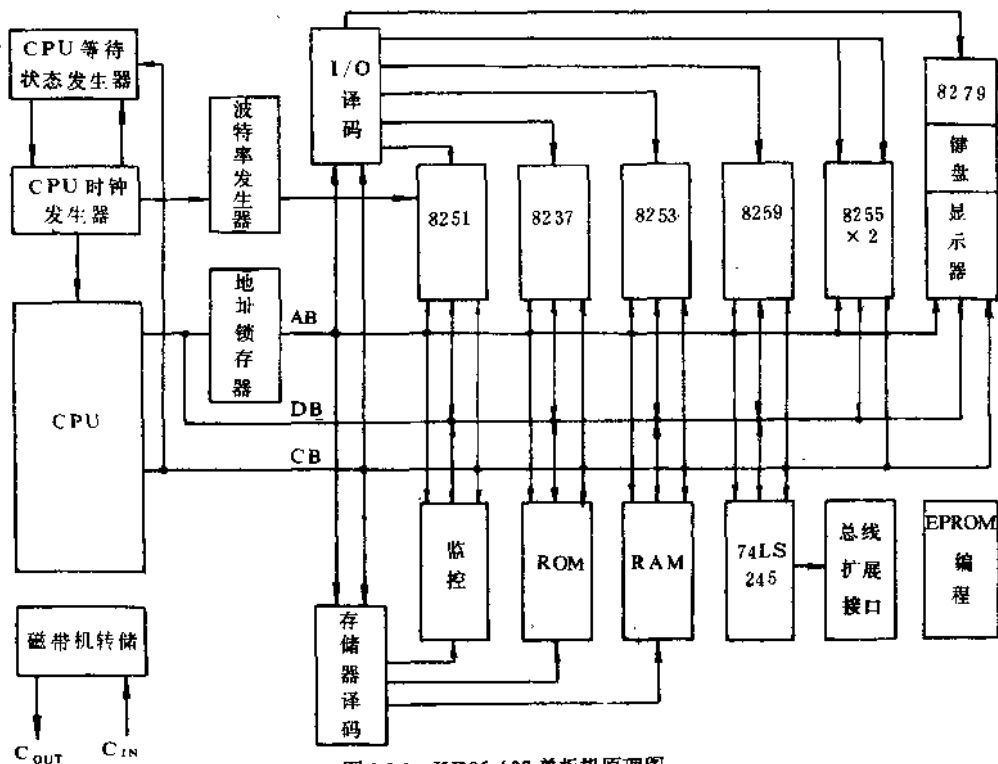


图 1.2.1 KD86/88 单板机原理图

### 一、CPU

KD 86/88 单板微型计算机采用 INTEL 8284 时钟信号发生驱动器为整个系统提供

时钟，以满足 CPU 所要求的前后沿时间（小于 10ns）及占空度 33% 的要求（当提供给 CPU 的时钟，接近 CPU 的极限工作频率时，其占空度必需是 33%）。图 1.2.2 是 CPU 的外围电路图。

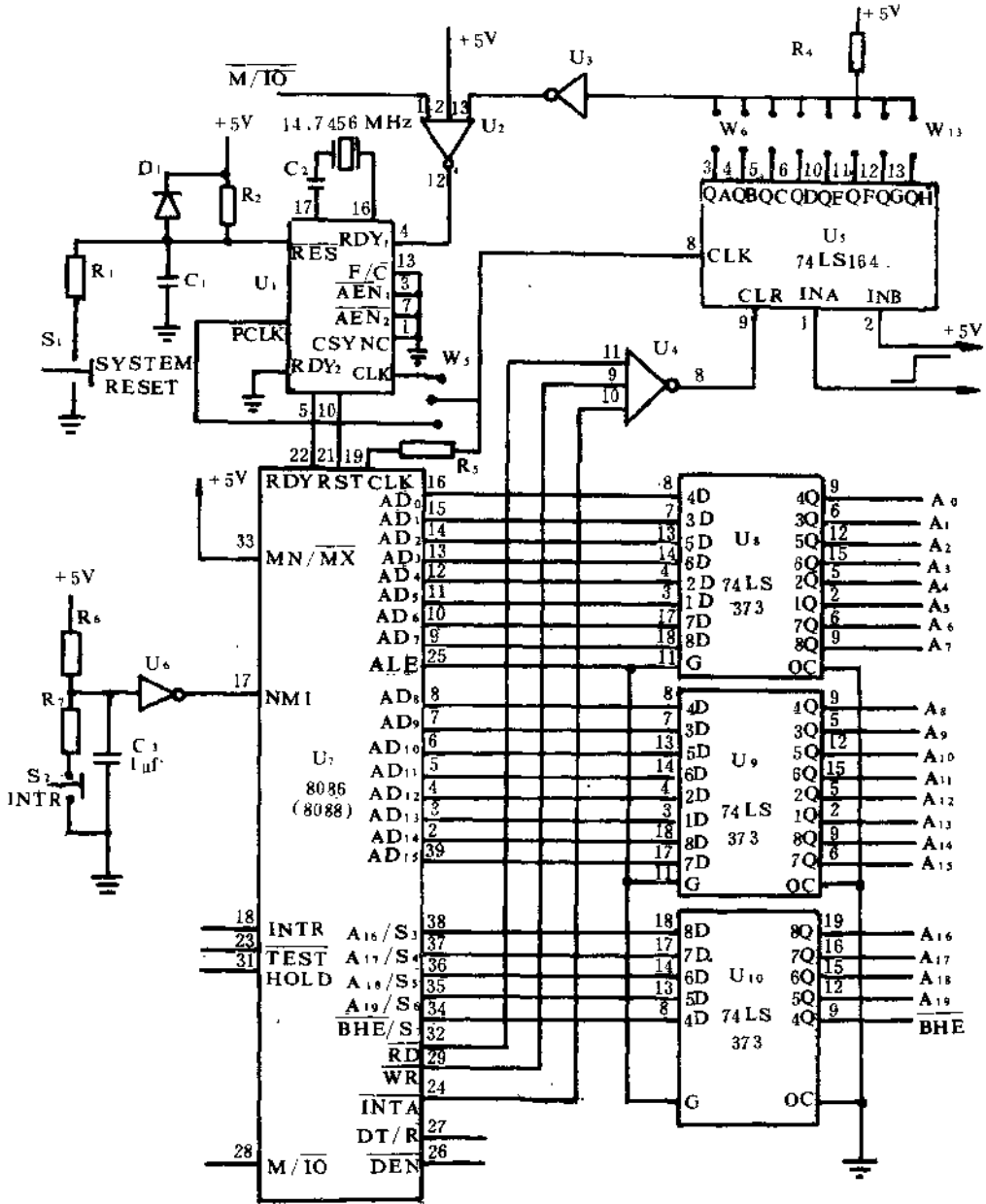


图 1.2.2 CPU 外围电路

8284 时钟信号发生器同时还提供两个和 CPU 时钟信号同步的输出控制信号 RDY 和 RST，当输入端 RES 有效时间大于 4 个时钟周期时，由 RST 输出“1”信号使 CPU 内部寄存器初始化为表 1.1.1 的状态。

表 1.1.1 CPU 寄存器初始化状态表

CPU 部件	内 容
标志寄存器 F	0 0 0 0 H
指令指示器 IP	0 0 0 0 H
CS 寄存器	F F F F H
DC 寄存器	0 0 0 0 H
SS 寄存器	0 0 0 0 H
ES 寄存器	0 0 0 0 H
QUEUE 队列	Empty 空

我们知道 8086(8088)CPU 在一个总线周期内, 通过  $AD_0 \sim AD_{15}$ ,  $A_{16} / S_3 \sim A_{19} / S_6$  20 个引脚分时进行地址、数据和状态操作。在  $T_1$  状态期间, CPU 在 20 位地址/数据/状态总线上发出地址, 通过 ALE 信号, 将地址状态信息锁存在 74LS373 八 D 锁存器中。在  $T_3$  状态期间, CPU 在  $AD_0 \sim AD_{15}$  总线上进行数据的读或写操作, 若存储器或 I/O 设备不能以 CPU 最大的传送速率传送数据时, CPU 自动引入等待状态  $T_w$ , 直到存储器或 I/O 设备完成数据传送, CPU 自动脱离  $T_w$  状态进行  $T_4$  状态操作。

等待状态发生器由 74LS164 并行输出串行移位寄存器组成。当系统要求插入  $T_w$  等待状态时, 8284 的  $RDY_1$  和  $RDY$  为低电平, CPU 停在  $RDY$  信号下降沿状态上; 等待状态发生器在时钟作用下, 使“1”在寄存器中移位, 当移到所选择的等待状态位置时, 跨越的短路开关的输出为“1”, 使  $RDY_1$  和  $RDY$  有效, CPU 便继续正常工作。表 1.2.2 是  $T_w$  等待状态选择表。

表 1.2.2 等待状态选择表

短路开关位置	等待状态数	短路开关位置	等待状态数
W6*	0	W10	4
W7	1	W11	5
W8	2	W12	6
W9	3	W13	7

\* 不跨越任何等待短路开关时, 等效于零等待状态

## 二、存 储 器

KD86/88 单板微型机具有 32K 字节 ROM 和 32K 字节 RAM 区。

### (一) ROM 存储器

ROM 存储器的片选信号由八中取一译码器 74LS138 组成。表 1.2.3 是存储空间分配表。

表 1.2.3 ROM 内存空间分配表

地 址	A <sub>19</sub> -A <sub>13</sub>	A <sub>12</sub> -A <sub>0</sub>	译码器有效输出
F0000H-F1FFFH	1 1 1 1 0 0 0	可 变	Y <sub>0</sub> 离板扩展
F2000H-F3FFFH	1 1 1 1 0 0 1	可 变	Y <sub>1</sub> 离板扩展
F4000H-F5FFFH	1 1 1 1 0 1 0	可 变	Y <sub>2</sub> 离板扩展
F6000H-F7FFFH	1 1 1 1 0 1 1	可 变	Y <sub>3</sub> 离板扩展
F8000H-F9FFFH	1 1 1 1 1 0 0	可 变	Y <sub>4</sub>
FA000H-FBFFFH	1 1 1 1 1 0 1	可 变	Y <sub>5</sub>
FC000H-FDFFFH	1 1 1 1 1 1 0	可 变	Y <sub>6</sub>
FE000H-FFFFFH	1 1 1 1 1 1 1	可 变	Y <sub>7</sub> 监控

(二) RAM 存储器

RAM 存储器的片选信号由八中取一译码器 74LS138 完成, 表 1.2.4 是存储空间分配表.

表 1.2.4 RAM 内存空间分配表

地 址	A <sub>19</sub> -A <sub>13</sub>	A <sub>12</sub> -A <sub>0</sub>	译码器有效输出
00000H-01FFFH	0 0 0 0 0 0 0	可 变	Y <sub>0</sub>
02000H-03FFFH	0 0 0 0 0 0 1	可 变	Y <sub>1</sub>
04000H-05FFFH	0 0 0 0 0 1 0	可 变	Y <sub>2</sub>
06000H-07FFFH	0 0 0 0 0 1 1	可 变	Y <sub>3</sub>
08000H-09FFFH	0 0 0 0 1 0 0	可 变	Y <sub>4</sub> 离板扩展
0A000H-0BFFFH	0 0 0 0 1 0 1	可 变	Y <sub>5</sub> 离板扩展
0C000H-0DFFFH	0 0 0 0 1 1 0	可 变	Y <sub>6</sub> 离板扩展
0E000H-0FFFFH	0 0 0 0 1 1 1	可 变	Y <sub>7</sub> 离板扩展

三、I/O 接口电路

INTEL 8086(8088)CPU 的 I/O 空间可容纳 64K 八位端口或为 32K 十六位端口, 利用 IN、OUT 指令可使累加器 (AX,AL) 同固定的 I/O 端口进行信息传输. 在 64K 字节的 I/O 空间中不能分段, INTEL 公司还规定 0F8H~0FFH(8 个单元)供将来硬件和软件产品使用.

KD86/88 单板机的 I/O 接口芯片有 INTEL 8255, INTEL 8251, INTEL8253, INTEL8237, INTEL 8259, INTEL 8279. 采用累加器 I/O 寻址方式同 CPU 进行通讯. 其片选信号由 16 选一译码器 74LS154 完成. 表 1.2.5 是 I/O 空间端口地址分配表.

表 1.2.5 I/O 端口地址分配表

地 址	A <sub>15</sub> ~ A <sub>3</sub>	A <sub>0</sub>	A <sub>1</sub> , A <sub>2</sub>	备 注
FFFFH	1111111111111,	1	可 变	Y' = 8255
FFF8H	1111111111111,	0	可 变	Y <sub>15</sub> = 8255
FFF0H	1111111111110,	0	可 变	Y <sub>14</sub> = 8251
FFE8H	1111111111101,	0	可 变	Y <sub>13</sub> = 8279
FFE0H	1111111111100,	0	可 变	Y <sub>12</sub> = 8253
FFD8H	1111111111011,	×	可 变	Y <sub>11</sub> = 8237
FFD0H	1111111111010,	×	可 变	Y <sub>10</sub> = 8237
FFC8H	1111111111001,	0	可 变	Y <sub>9</sub> = 8259
FFC0H	1111111111000,	0	可 变	Y <sub>8</sub> 离板扩展
FFB8H	1111111111011,	0	可 变	Y <sub>7</sub> 离板扩展
FFB0H	1111111111010,	0	可 变	Y <sub>6</sub> 离板扩展
FFA8H	11111111110101,	0	可 变	Y <sub>5</sub> 离板扩展
FFA0H	11111111110100,	0	可 变	Y <sub>4</sub> 离板扩展
FF98H	11111111110011,	0	可 变	Y <sub>3</sub> 离板扩展
FF90H	11111111110010,	0	可 变	Y <sub>2</sub> 离板扩展
FF88H	11111111110001,	0	可 变	Y <sub>1</sub> 离板扩展
FF80H	11111111110000,	0	可 变	Y <sub>0</sub> 离板扩展

### § 1.3 KD86 / 88 单板机使用方法

本节介绍 KD86 / 88 单板机在监控程序控制下同用户进行通讯的方法。

KD86 / 88 单板机在接通电源或按下“SYSTEM RESET”（系统复位）键时，KD86 / 88 单板机就从 0FFFF0H 为起始地址开始执行监控程序，此时单板机上的 LED 显示器便显示出系统提示符 8086 -CPU (或 8088 -CPU)。该提示符的显示表示监控程序已对系统进行初始化，这时 CPU 寄存器内容如表 1.3.1 所示。

表 1.3.1 初始化后 CPU 寄存器状态表

寄 存 器	内 容
CS (代码段)	0 0 0 0 H
DS (数据段)	0 0 0 0 H
ES (附加段)	0 0 0 0 H
SS (堆栈段)	0 0 0 0 H
IP (指令指示器)	0 0 0 0 H
FL (标志)	0 0 0 0 H
SP (堆栈指示器)	0 1 0 0 H

同时中断矢量初始化为：

中断 1: 用于单步命令

中断 2: 用于 INTR 功能键

中断 3: 用于 GO 命令

在完成上述操作后, KD86/88 单板微型机便可通过键盘和显示器完成下列各种操作:

检测和修改微处理器内的寄存器内容。

检测和修改存储器单元内容。

输入和开始执行用户程序或子程序。

通过监控程序的单步和断点功能来鉴定用户程序的执行 (调试)。

将所选的存储器块由一个单元传送到另一个单元去。

往 I/O 端口写数据或从 I/O 端口读数据。

### 一、键 盘

KD86/88 单板机共有 24 个按键, 在键盘监控程序的控制下, 分为两个逻辑组: 右边 16 个键为十六进制键, 左边 8 个为功能键, 该键盘在机器上的排列如图 1.3.1 所示。

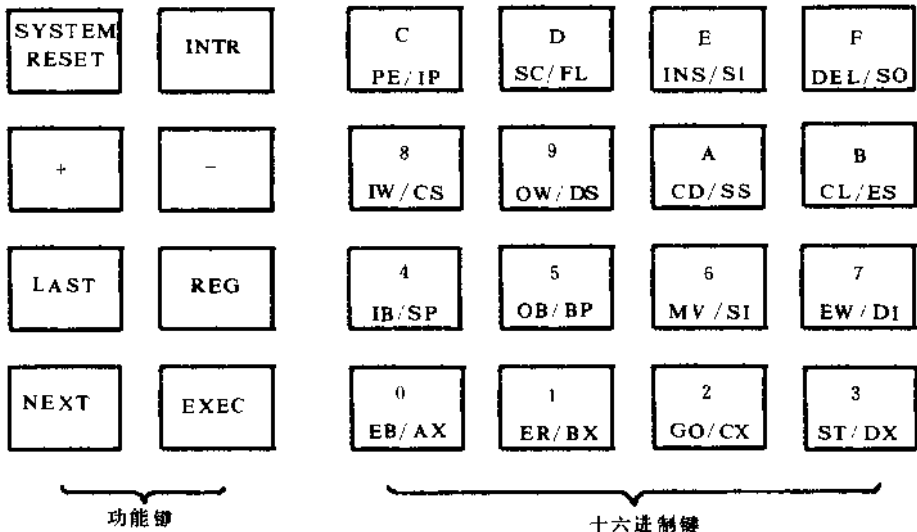


图 1.3.1 键 盘

#### 功能键说明:

**SYSTEM RESET:** 按下此键时则终止当前操作, 并对系统进行初始化和显示系统提示符

```
8086 -CPU
```

**INTR:** 该键用来产生 NMI 中断. 按下该键时将保护 CPU 的当前寄存器内容。

**+** : 该键可以进行两个 16 进制数的加法, 并可计算相对于一个基地址的地址。

**-** : 该键可以进行两个 16 进制数的减法。

**LAST:** 该键将输入的地址分为段地址和偏移量, 配合 EB/EW 命令键使地址减量 (减 1, 减 2)。

REG: 该键可使 CPU 中的寄存器内容作为一个地址或数据使用。

NEXT: 该键用来区分键盘命令键, 并可使地址增量, 指向下一个相邻的存储单元。

EXEC: 该键是命令的终止符, 按下此键时, 即执行当前命令, 当用 GO 命令时, 按下此键就开始执行指定地址处的程序。

十六进制键说明:

0 EB / AX	: EB 检查字节	AX 累加器
1 ER / BX	: ER 检查寄存器	BX 基址寄存器
2 GO / CX	: GO 转向指定程序	CX 计数寄存器
3 ST / DX	: ST 单步操作	DX 数据寄存器
4 IB / SP	: IB 输入字节	SP 堆栈指示器
5 OB / BP	: OB 输出字节	BP 基址指示器
6 MV / SI	: MV 传送	SI 源变址寄存器
7 EW / DI	: EW 检查字	DI 目的变址寄存器
8 IW / CS	: IW 输入字	CS 代码段寄存器
9 OW / DS	: OW 输出字	DS 数据段寄存器
A CD / SS	: CD 磁带转储	SS 堆栈段寄存器
B CL / ES	: CL 磁带输入	ES 附加段寄存器
C PE / IP	: PE EPROM 编程	IP 指令指示器
D SC / FL	: SC 串行通讯	FL 标志寄存器
E INS / SI	: INS 插入一个字节	SI 串行输入
F DEL / SO	: DEL 删除一个字节	SO 串行输出



## 二、键盘操作过程

KD86/88A 单板机接通电源并显示出提示符后，便可执行各种命令。每个命令键的具体使用方法如下：

### (一) EB 和 EW 命令：

EB 和 EW 命令的功能是用来检测和修改指定存储单元内容，该命令的格式为：

<u>EB</u>	(地址)	<u>NEXT</u>	[(数据) <u>NEXT</u> ]	<u>EXEC</u>	地址增 1
<u>EB</u>	(地址)	<u>NXET</u>	[(数据) <u>LAST</u> ]	<u>EXEC</u>	地址减 1
<u>EW</u>	(地址)	<u>NEXT</u>	[(数据) <u>NEXT</u> ]	<u>EXEC</u>	地址增 2
<u>EW</u>	(地址)	<u>NEXT</u>	[(数据) <u>LAST</u> ]	<u>EXEC</u>	地址减 2

操作过程：当出现系统提示符后，即可使用 EB 和 EW 键。当按下上述两键中的任意一个时，地址段右边的小数点亮，这时便可输入存储单元的段值和偏移值。当没有指定段值时，缺省的段值是当前 CS 寄存器的值。当指定了段值后，首先应输入段值，紧接着应按“LAST”键（段值分隔符），再输入地址的偏移值。输入地址后，按下“NEXT”键，这时数据段显示器便显示出当前存储单元的内容（字节或字）。若要修改该存储单元内容，即通过键盘将要修改的数据输入。如需检查存储单元中的内容，则可按“NEXT”键或“LAST”键使存储单元地址以字节或字的形式增量或减量，同时也可修改存储单元的内容。按下“EXEC”键后，即结束本条命令返回监控。

例 1 检查相对于 CS 寄存器内容的存储单元内容（地址增量）。

按键	地址段显示	数据段显示	注 释
SYSTEM RESET	8 0 8 6	- C P U	; 系统复位提示符
<sup>0</sup> EB/AX			; EB 命令
<sup>1</sup> ER/BX			; 输入存储单元高位地址
<sup>0</sup> EB/AX			; 输入存储单元次高位地址
<sup>A</sup> CD/SS	1 0 A.		; CS:10A 单元地址输入完
NEXT	1 0 A.	× ×.	; 显示 10AH 单元的内容 × ×
NEXT	1 0 B.	× ×.	; 显示 10BH 单元的内容
NEXT	1 0 C.	× ×.	; 显示 10CH 单元的内容
<sup>8</sup> IW/CS	1 0 C.	× 8.	; 更改 10CH 单元内容