



第7章 单片机I/O扩展及应用

7.1 I/O接口技术概述

7.2 MCS-51单片机I/O口的直接应用

7.3 单片机简单I/O口扩展

7.4 8255A可编程通用并行接口芯片

7.5 8155可编程并行接口芯片

7.6 单片机键盘接口技术

7.7 单片机LED显示器接口技术

7.8 MCS-51单片机打印机接口技术



7.1 I/O接口技术概述

- 一、I/O接口电路的功能：** I/O接口电路实现CPU与外设之间的数据输入/输出，主要功能包括：速度协调、数据锁存、三态缓冲、数据转换等。
- 二、端口：** I/O接口电路中能进行读/写操作的寄存器，一个接口电路有多个口。
- 三、数据总线隔离：** 数据总线与多个I/O设备连接，所以需要隔离。输出接口电路要提供锁存器，输入接口电路使用三态缓冲器。

三态缓冲器逻辑

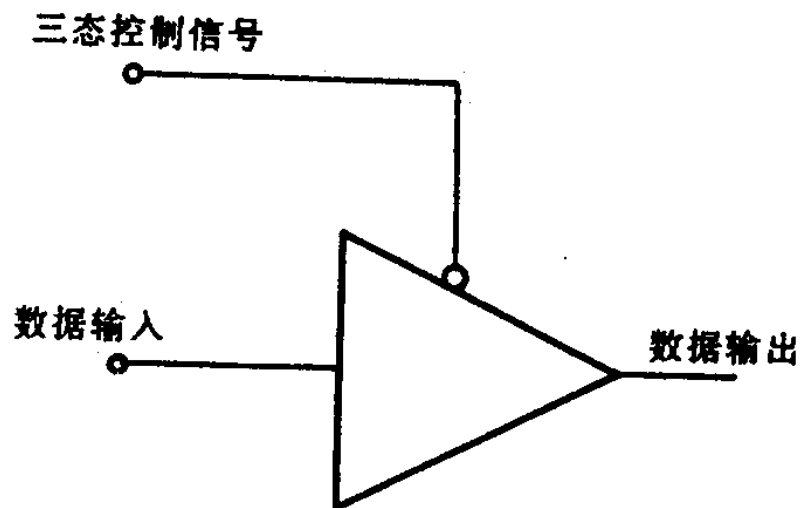


图 7.5 三态缓冲器逻辑符号

表 7-1 三态缓冲控制逻辑

三态控制信号	工作状态	数据输入	输出端状态
1	高阻抗	0	高阻抗
		1	高阻抗
0	驱动	0	0
		1	1

四、I/O口编址技术

独立编址方式：I/O口与存储器分开编址；

统一编址方式：I/O口与存储器统一编址。MCS-51采用这种方式。

五、I/O口数据传送的控制方式

无条件传送方式：不测试外设状态，根据需要随时传送数据；

程序查询方式：先以程序方法检测外设状态，再进行数据传送；

中断方式：先由外设提出中断请求，再进行数据传送。

7.2 MCS-51单片机I/O口的直接应用

位操作:

I/O操作指令

输出数据:

MOV Px, A

MOV Px, Rn

MOV Px, @Ri

MOV Px, direct

输入数据:

MOV A, Px

MOV Rn, Px

MOV @Ri, Px

MOV direct, Px

MOV Px.y, C

CLR Px.y

SETB Px.y

CPL Px.y

JB Px.y, rel

JNB Px.y, rel

JBC Px.y, rel

I/O口其它操作指令:

ANL Px, A

ORL Px, A

XRL Px, A

CJNE A, Px, rel

I/O口的直接使用

例：开关每扳动一次，将K0~K3的状态显示在LED0~LED3

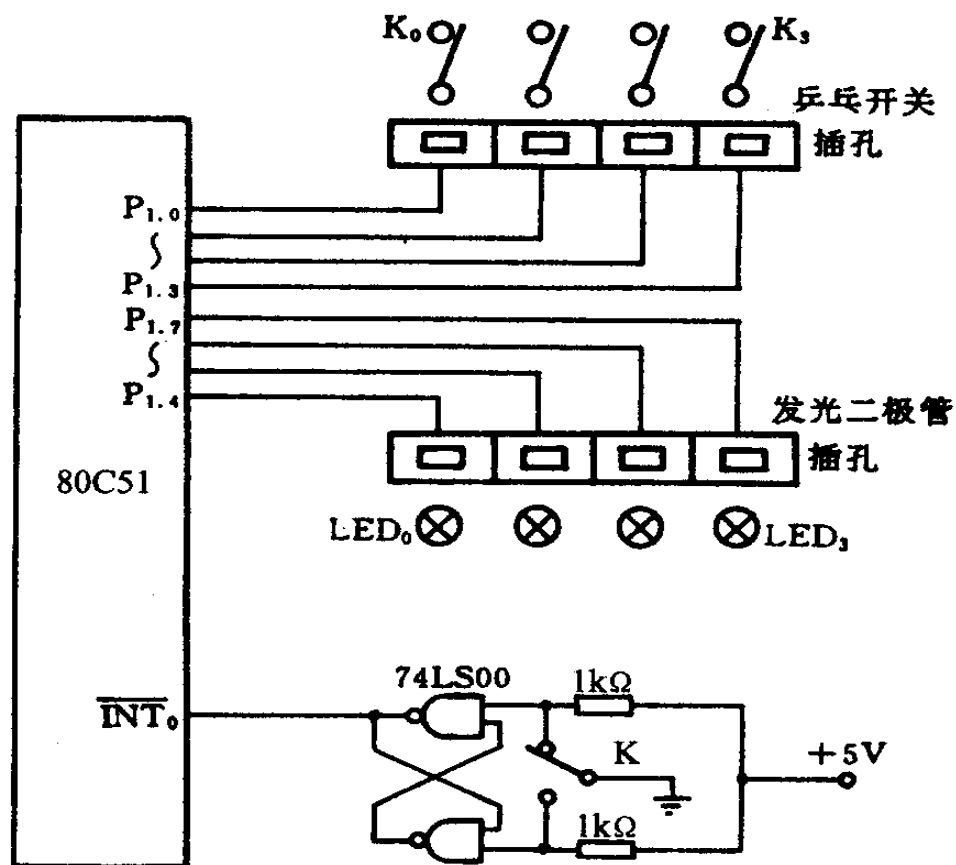


图 7.3 中断方式的简单 I/O 应用电路连接

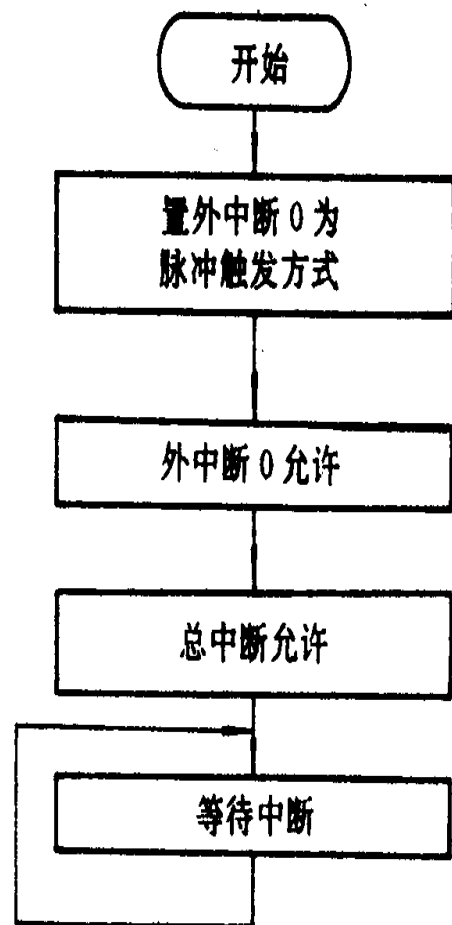


图 7.4 主程序流程图

```
STAR:  ORG 8000H
        AJMP MAIN
```

```
        ORG 8003H
        AJMP EXTR
```

```
MAIN:   ORG 8030H
        SETB IT0
        SETB EX0
        SETB EA
        HERE: AJMP HERE
```

;脉冲边沿触发
;外部中断0允许
;总中断允许
;等待中断

```
EXTR:   ORG 8200H
        P1 EQU 90H
        MOV A, #0FH
        MOV P1, A
        MOV A, P1
        CPL A
        ANL A, #0FH
        SWAP A
        MOV P1, A
        RETI
```

输出0熄LED，输出1
准备输入数据

;中断服务程序
;熄发光二极管
;输入开关状态
;状态取反
;屏蔽A的高半字节
;A高低半字节交换
;开关状态输出
;中断返回

7.3 单片机简单输入扩展

先介绍一个芯片

芯片**244**：2个4位三态缓冲器， **$1A_4 \sim 1A_1$** 、 **$2A_4 \sim 2A_1$** 为输入端； **$1Y_4 \sim 1Y_1$** 、 **$2Y_4 \sim 2Y_1$** 为输出端； **$/CE_1$** 、 **$/CE_2$** 为选通信号，低电平有效。

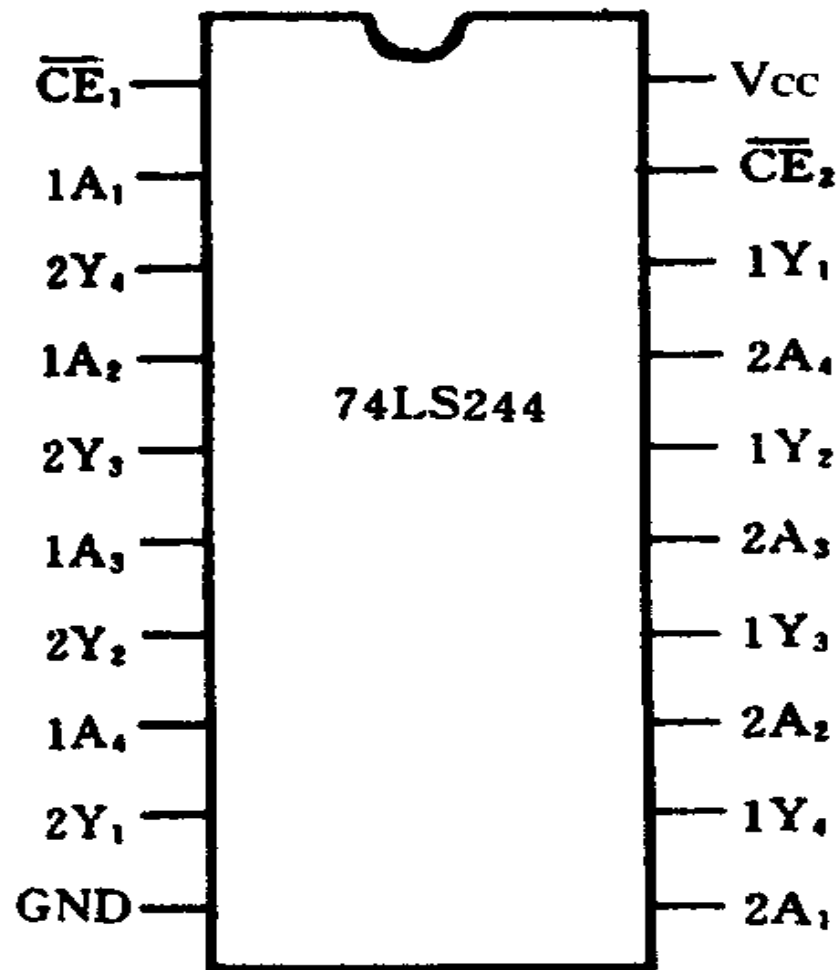


图 7.9 74LS244 引脚排列

单输入口扩展

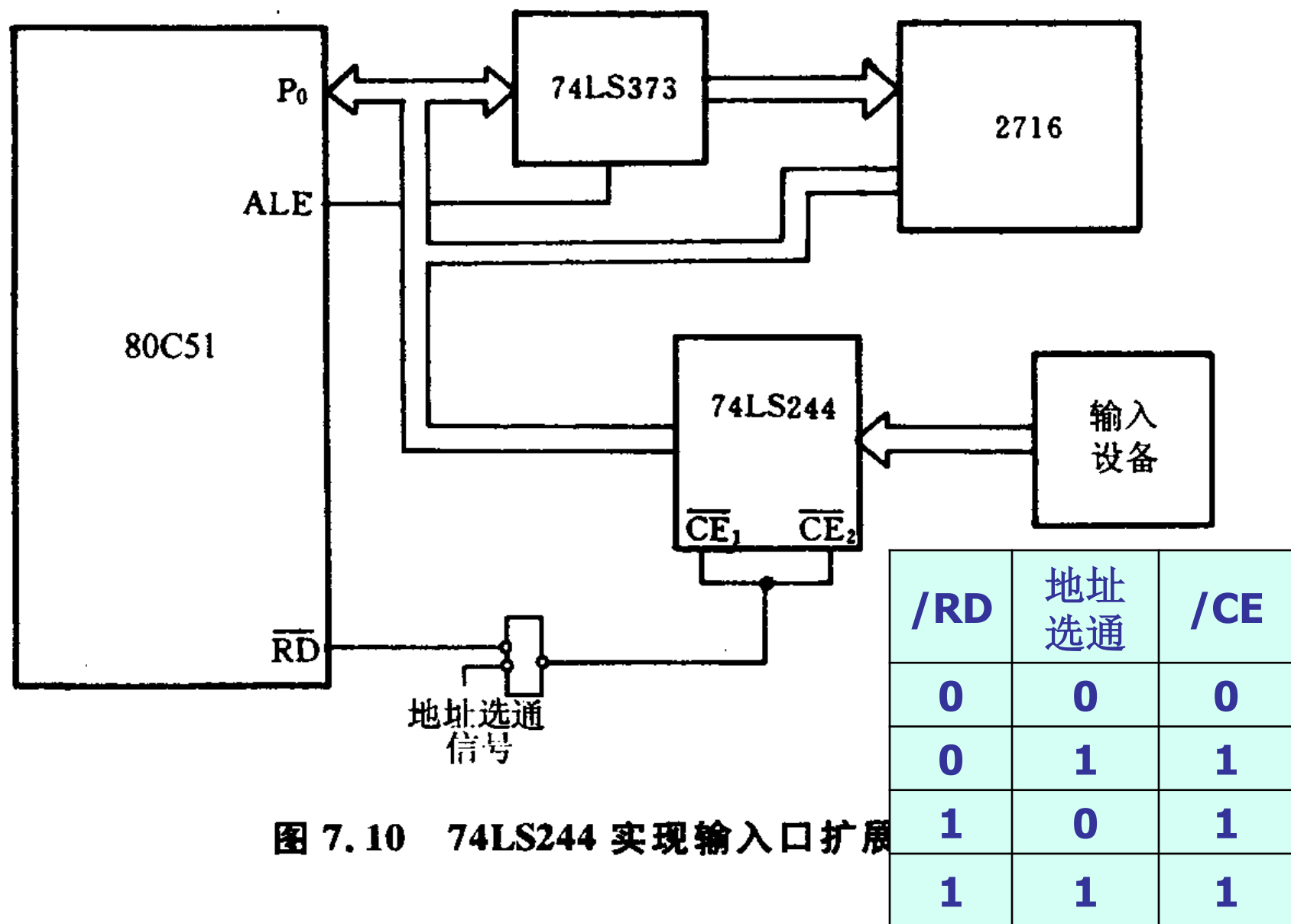


图 7.10 74LS244 实现输入口扩展

单片机基础



单片机基础

```
MOV DPTR, #7FFFH
MOVX A, @DPTR
MOV R4, A
```

BFFFH



简单输入输出扩展

介绍芯片**377**：8D锁存器。8D~1D：数据输入端；8Q~1Q：数据输出端；CK：时钟信号，上升沿锁存；/G：使能控制。

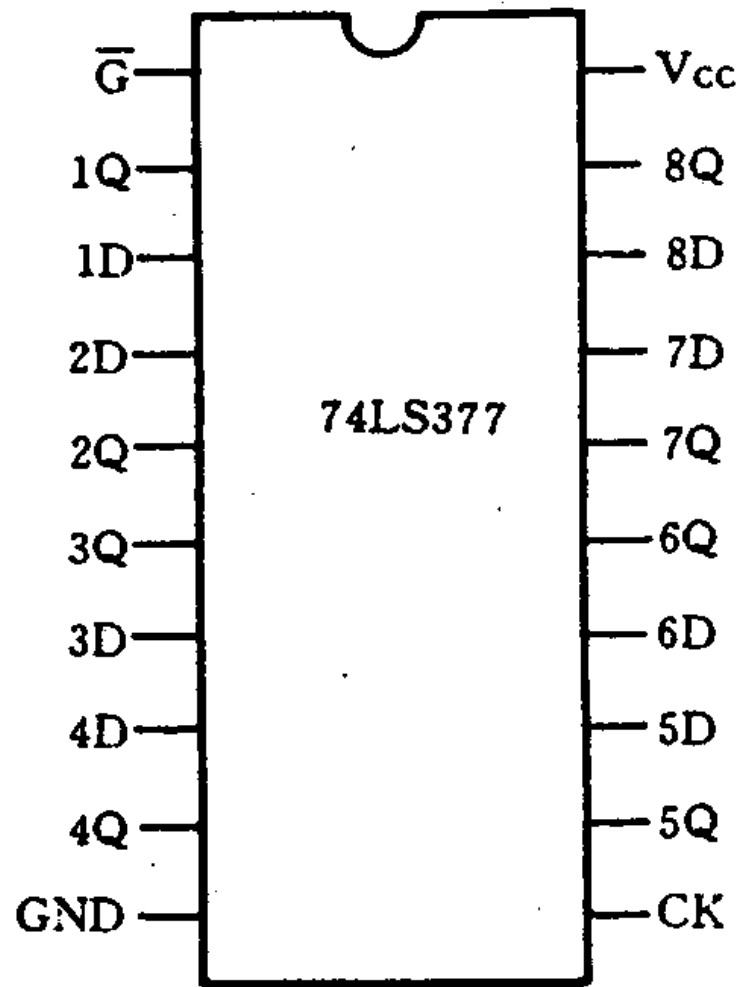


图 7.13 74LS377 引脚图

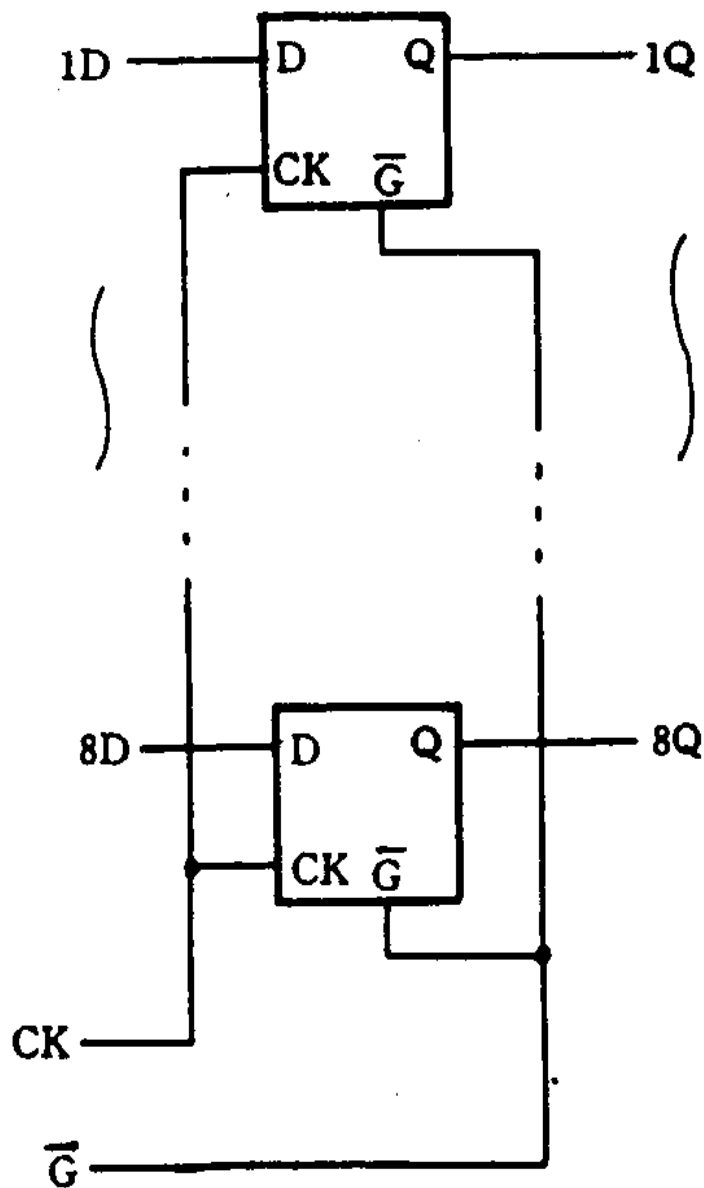


图 7.14 74LS377 逻辑电路

表 7-2 74LS377 真值表

\overline{G}	CK	D	Q
1	X	X	Q_0
0	\uparrow	1	1
0	\uparrow	0	0
X	0	X	Q_0

利用377扩展单个输出回

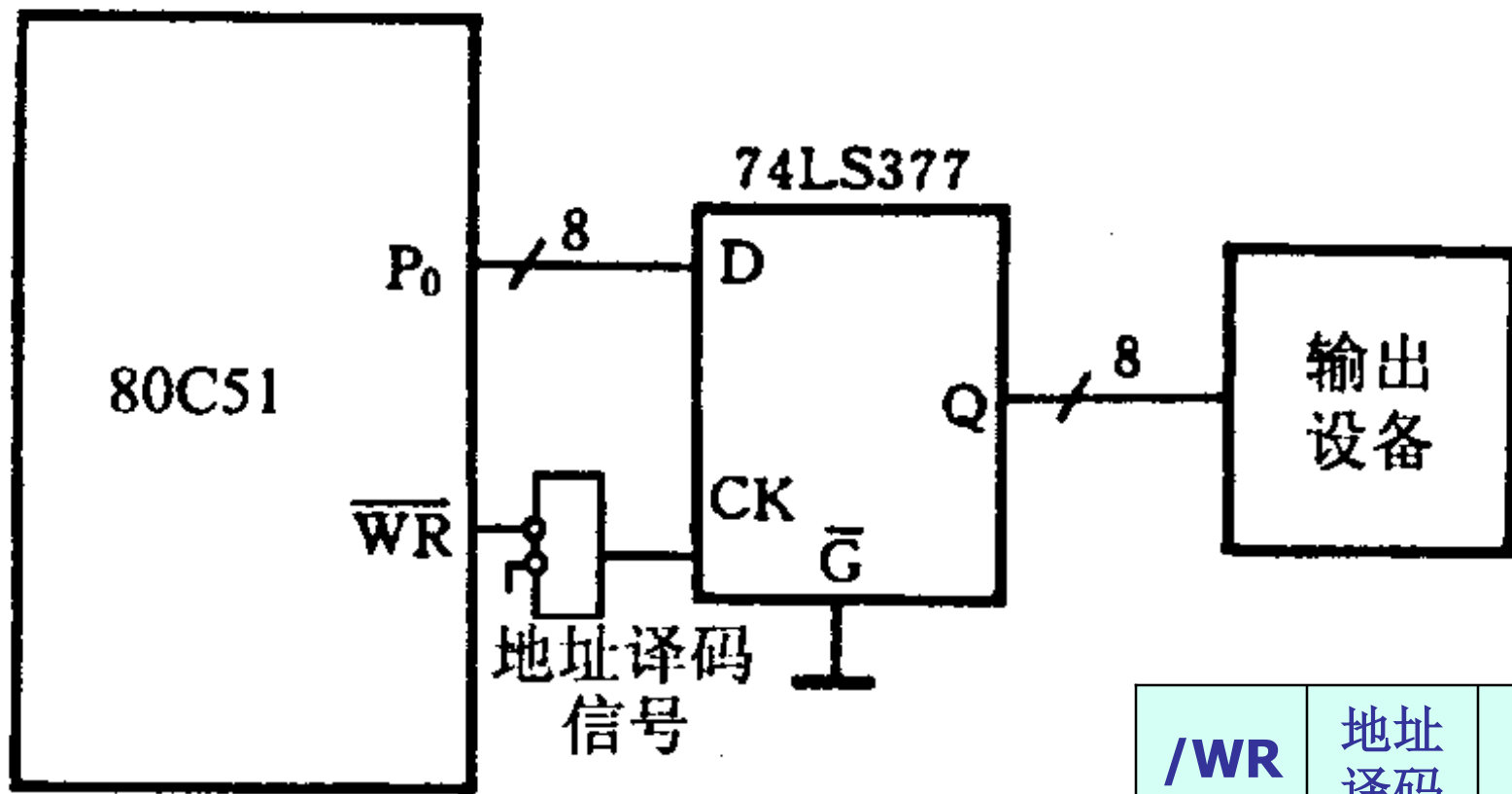


图 7.15 74LS377 作输出口

/WR	地址译码	CK
0	0	1
0	1	0
1	0	0
1	1	0

输出

锁存状态



常用的可编程接口芯片

8255A: 可编程通用并行接口;

8155: 带**RAM**和定时器/计数器的可编程并行接口;

8259: 可编程中断控制器;

8279: 可编程键盘/显示器接口;

8253: 可编程通用定时器

7.4 8255A可编程通用并行接口芯片

8255A的逻辑结构和信号引脚

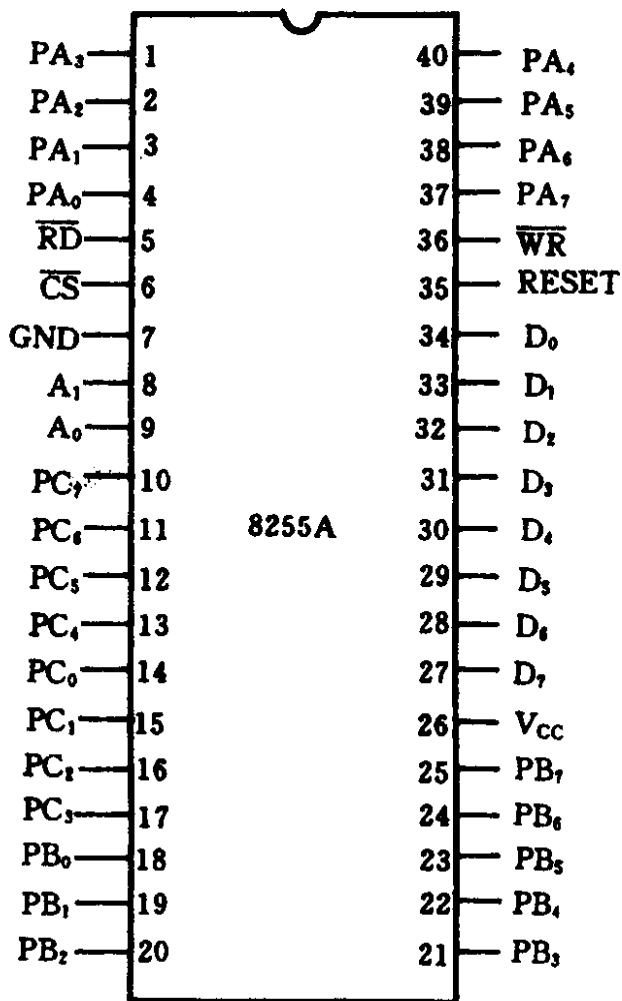


图 7.16 8255A 引脚图

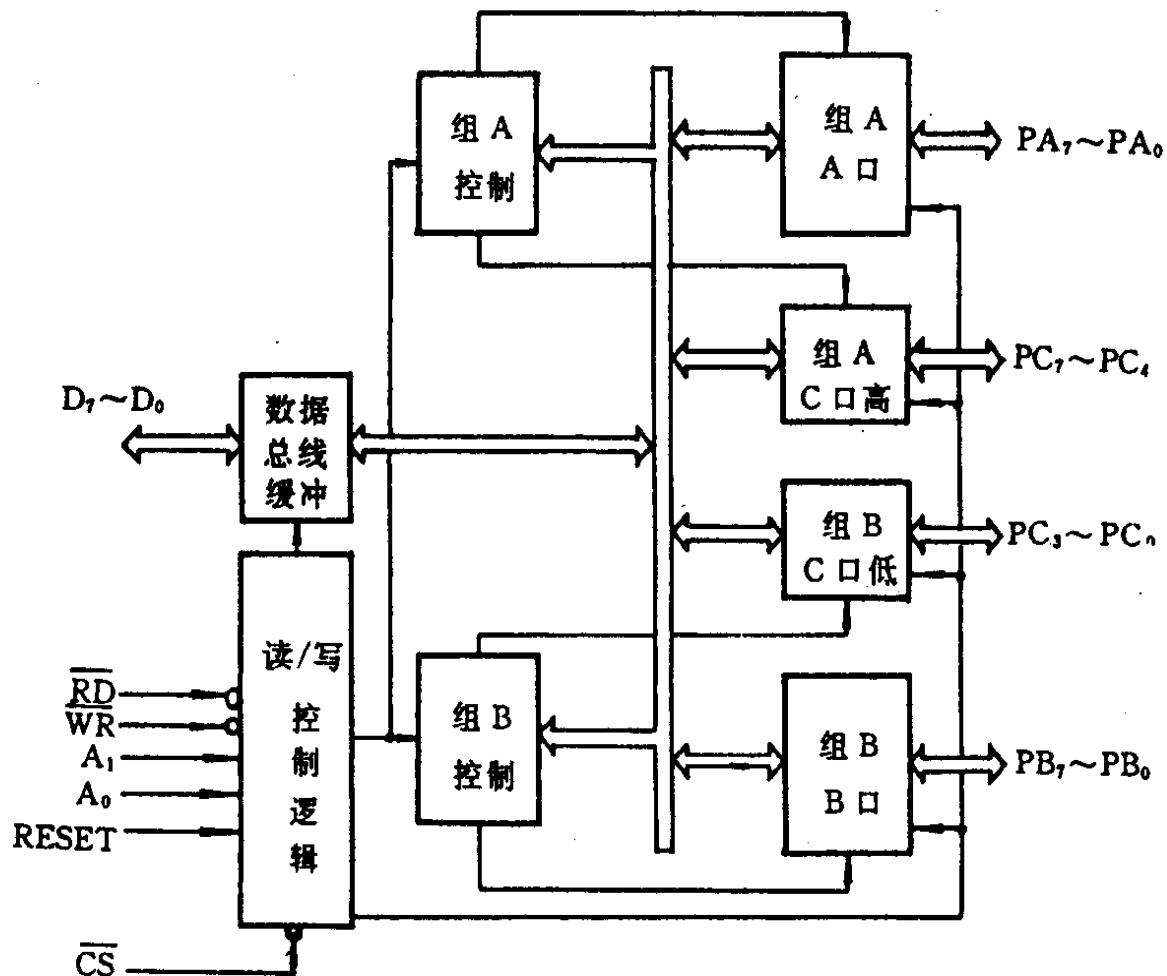


图 7.17 8255A 的逻辑结构

8255A的逻辑结构和信号引脚

A口 PA7~PA0: 数据输入/输出口。

B口 PB7~PB0: 数据输入/输出口。

C口 PC7~PC4: 数据输入/输出口或控制口 (**A**口控制)。

C口 PC3~PC0: 数据输入/输出口或控制口 (**B**口控制)。

/CS: 片选信号。

/RD: 读信号 (**CPU**读取**8255A**的数据)。

/WR: 写信号 (**CPU**把数据或控制字写到**8255A**中)。

A1、A0: 端口选择信号, 应与**P0.0**和**P0.1**相连。

RESET: 复位信号, 复位后, 控制寄存器清除, 各端口置为输入方式。

8255A的控制表及地址

表 7-3 8255A 读/写控制表

\overline{CS}	A_1	A_0	\overline{RD}	\overline{WR}	所选端口	操 作
0	0	0	0	1	A 口	读端口 A
0	0	1	0	1	B 口	读端口 B
0	1	0	0	1	C 口	读端口 C
0	0	0	1	0	A 口	写端口 A
0	0	1	1	0	B 口	写端口 B
0	1	0	1	0	C 口	写端口 C
0	1	1	1	0	控制寄存器	写控制字
1	×	×	×	×	/	数据总线缓冲器输出高阻抗



8255A工作方式及数据I/O操作

三种工作方式:

方式0: 基本输入/输出方式。A口、B口作为8位I/O口；C口高、C口低作为4位I/O口。

方式1: 选通输入/输出方式。A口、B口作为8位选通I/O口；C口相关口线用作数据传送的联络信号。

方式2: 双向数据传送方式。A口可作为I/O双向传送口（这时B口工作于方式0，C口相关口线作为控制信号）。

C口的联络信号

表 7-4 C 口联络信号定义

C 口位线	方 式 1		方 式 2	
	输入	输出	输入	输出
PC ₇		$\overline{\text{OBFA}}$		$\overline{\text{OBFA}}$
PC ₆		$\overline{\text{ACKA}}$		$\overline{\text{ACKA}}$
PC ₅	IBFA		IBFA	
PC ₄	$\overline{\text{STBA}}$		$\overline{\text{STBA}}$	
PC ₃	INTRA	INTRA	INTRA	INTRA
PC ₂	$\overline{\text{STBB}}$	$\overline{\text{ACKB}}$		
PC ₁	IBFB	$\overline{\text{OBFB}}$		
PC ₀	INTRB	INTRB		

选通方式下数据的输入过程

外设准备好数据→发出/**STB**信号→数据进入缓冲器→**IBF**信号有效（单片机可查询）→产生**INTR**信号（可使单片机中断）→单片机读入数据→**IBF**变低。

有关控制信号：

/STB：选通信号，低电平有效，外设送给**8255A**，将输入数据装入**8255A**锁存器。

IBF：输入缓冲器满，高电平有效，**8255A**输出给单片机的状态信号，表明数据已装入锁存器。

INTR：中断请求信号，高电平有效，**8255A**输出给单片机，请求中断。

选通方式下数据的输出过程

数据输出过程：外设接收完一组数据→发回/**ACK**信号→/**OBF**变高（供单片机查询）→**INTR**有效（可使单片机中断）→单片机输出下一个数据到**8255A**缓冲器→/**OBF**有效→外设取走数据。

有关控制信号：

/ACK：外设响应信号，低电平有效，外设发给**8255A**，表明数据取走。

/OBF：输出缓冲器满，低电平有效，**8255A**输出的状态信号，表明单片机已将数据装入锁存器。

INTR：中断请求信号，高电平有效，**8255A**输出给单片机，请求中断。

8255A 的控制字

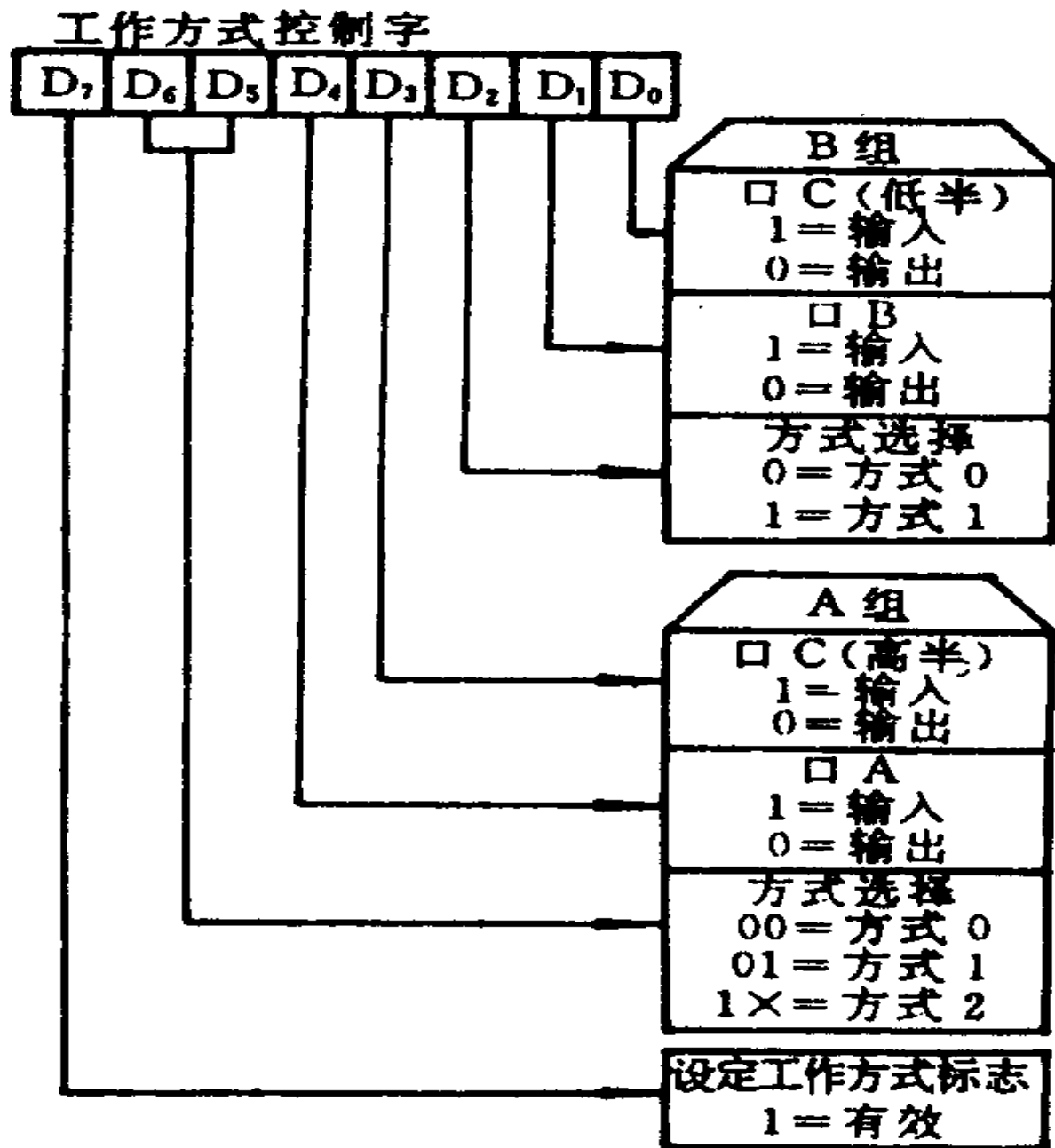


图 7.18 8255A 工作方式控制字格式

C口的位置位/复位

位置位/复位控制字

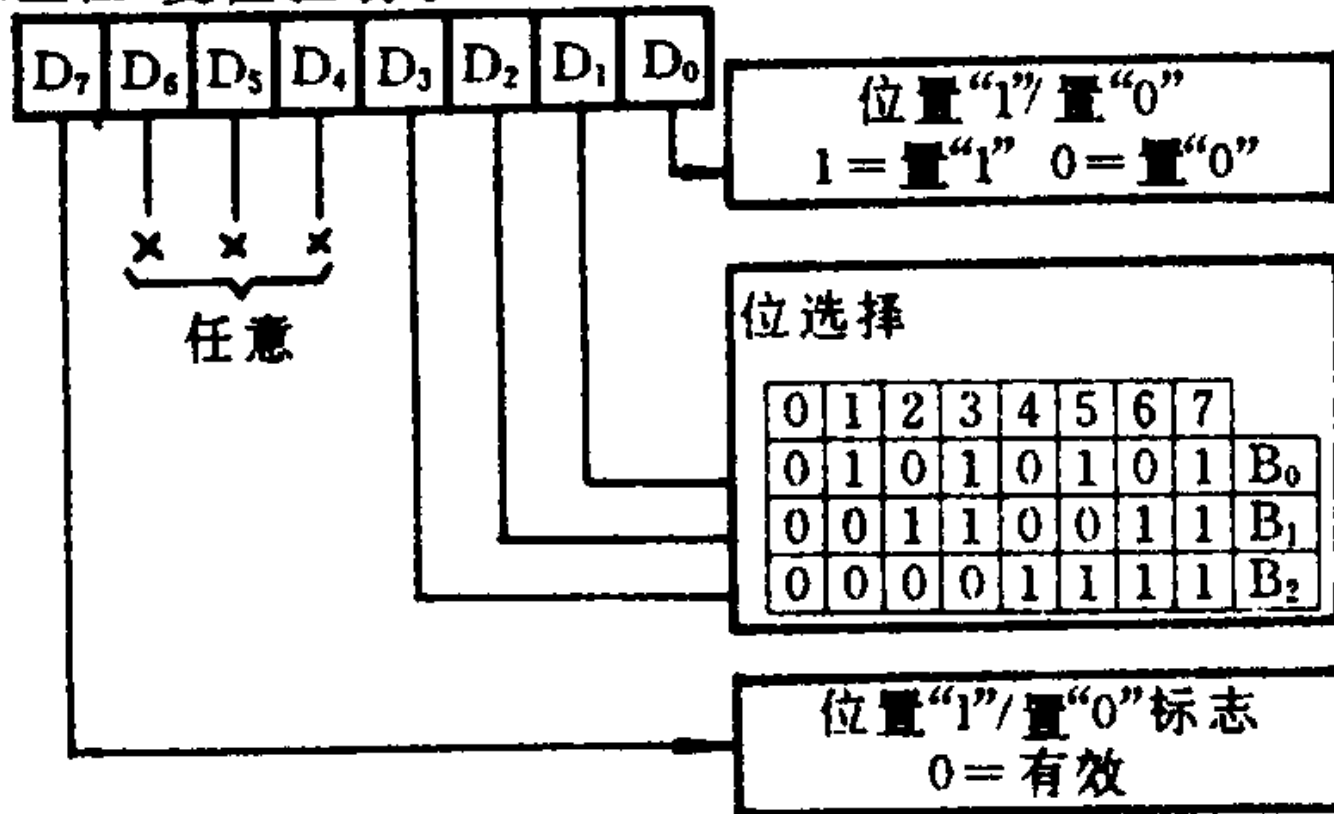


图 7.19 8255A 位置位/复位控制字格式



8255A的初始化例子

A口：方式0输入； B口：方式1输出， C口高：输出； C口低：输入。则控制字为10010101，即95H。

MOV R0, #03H; 设控制寄存器地址为0003H

MOV A, #95H

MOVX @R0, A

7.5 8155可编程并行接口芯片

8155的结构

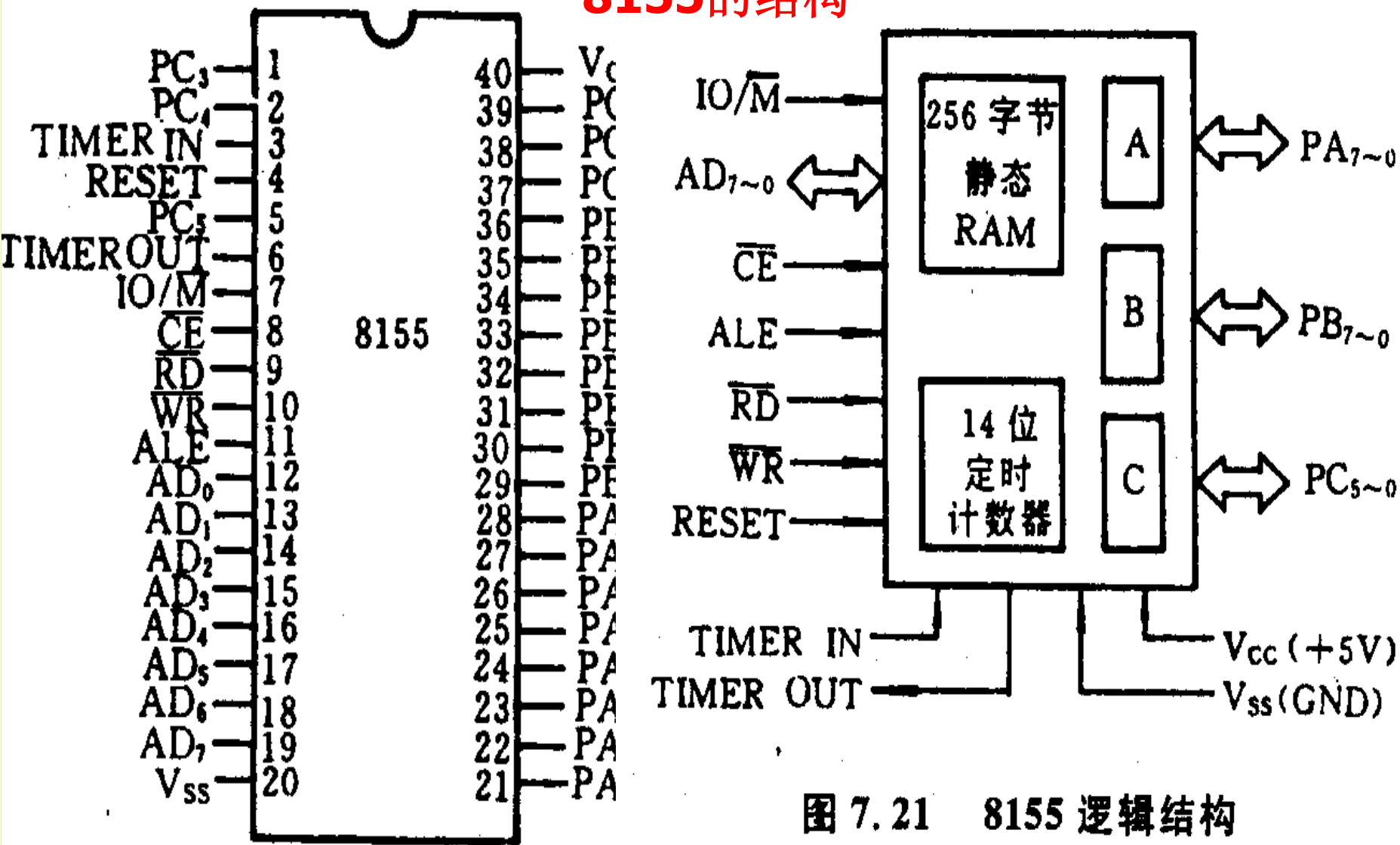


图 7.21 8155 逻辑结构



8155的管脚

AD7~AD0: 地址/数据复合线

ALE: 地址锁存信号（锁存地址信号、/CE信号、IO/M信号等）

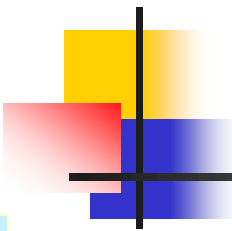
/RD: 读选通信号

/WR: 写选通信号

/CE: 片选信号

IO/M: I/O与RAM选择信号

RESET: 复位信号



8155的工作方式

A口（PA）工作方式：2种，输入方式，输出方式

B口（PB）工作方式：2种，输入方式，输出方式

C口（PC）工作方式：4种，输入方式，输出方式，PA控制端口方式，PA、PB口控制方式

C口联络信号

表 7-5 PC 口线联络信号定义

方式 口位	作 PA 控制端口	作 PA 和 PB 控制端口
PC ₀	INTRA	INTRA
PC ₁	ABF	ABF
PC ₂	$\overline{\text{ASTB}}$	$\overline{\text{ASTB}}$
PC ₃	输出	INTRB
PC ₄	输出	BBF
PC ₅	输出	$\overline{\text{BSTB}}$

INTR: 中断请求信号（输出），高有效

BF: 缓冲器满信号（输出），高有效

/STB: 选通信号（输入），数据输入时外设送来选通信号，数据输出时外设送来应答信号

8155的端口地址

表 7-6 8155 的可编程端口

AD ₇	AD ₆	AD ₅	AD ₄	AD ₃	AD ₂	AD ₁	AD ₀	选 择
×	×	×	×	×	0	0	0	命令/状态寄存器
×	×	×	×	×	0	0	1	PA 口
×	×	×	×	×	0	1	0	PB 口
×	×	×	×	×	0	1	1	PC 口
×	×	×	×	×	1	0	0	定时器/计数器低 8 位
×	×	×	×	×	1	0	1	定时器/计数器高 8 位

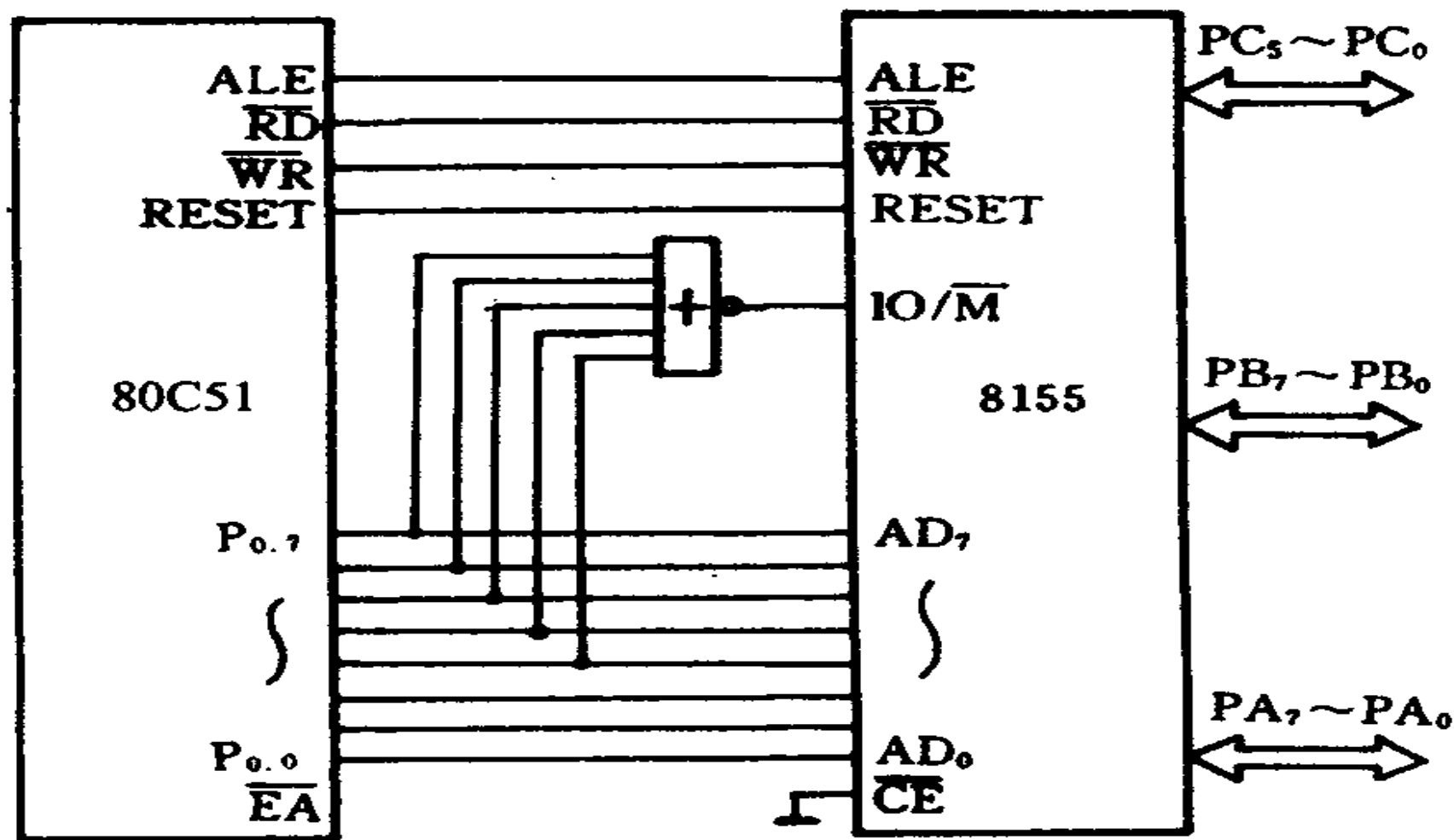
8155与MCS-51单片机的连接

兼容信号的连接

表 7-7 8155 与 MCS-51 的兼容信号

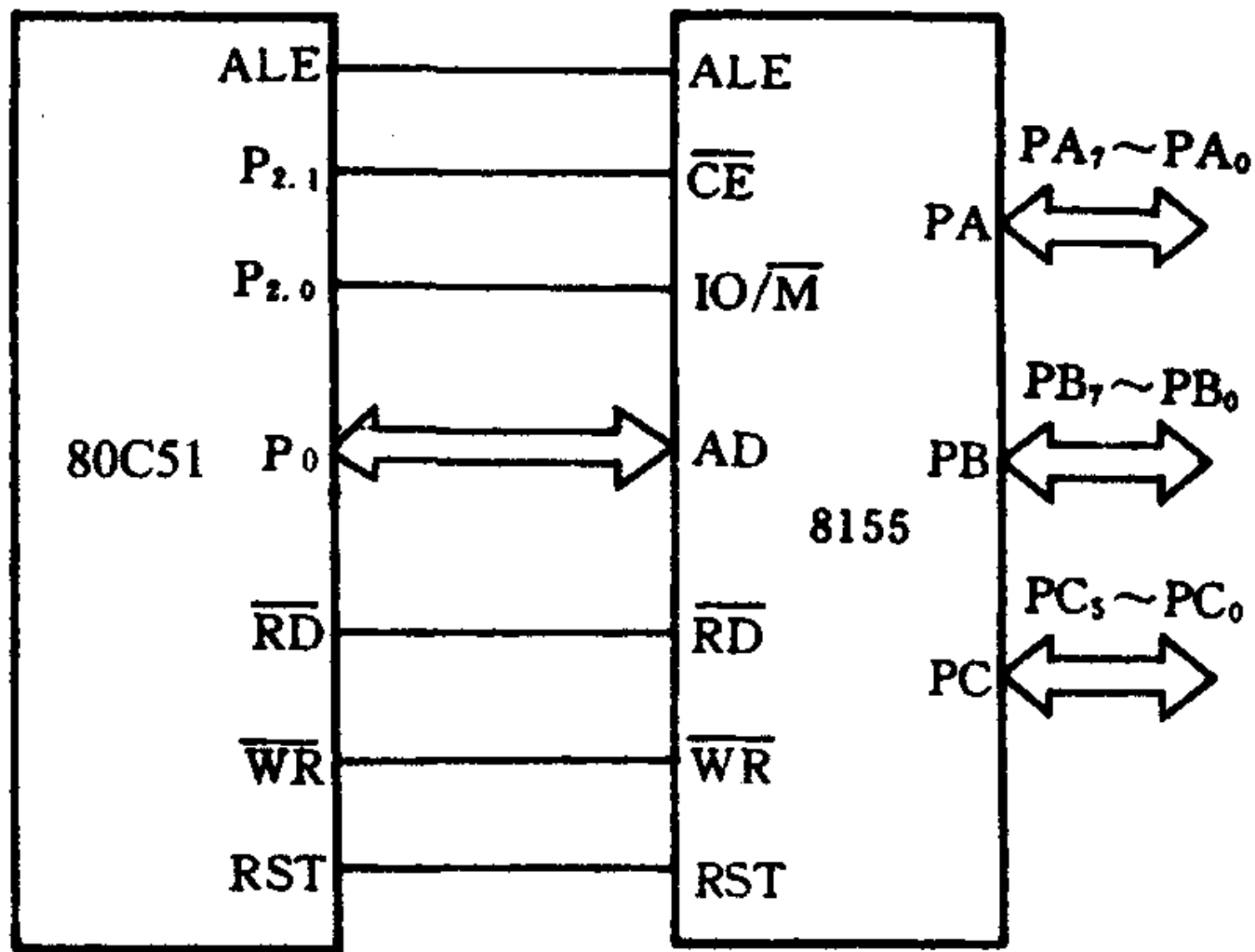
8155	MCS-51	8155	MCS-51
$AD_7 \sim AD_0$	P_0 口	\overline{RD}	\overline{RD}
ALE	ALE	\overline{WR}	\overline{WR}
RESET	RST		

8155的I/O/M信号与单片机的连接



方法一

或非门产生 IO/\overline{M} 信号



方法二

高位地址作 $\text{IO}/\overline{\text{M}}$ 信号

8155命令/状态字

地址相同的2个寄存器，写操作为命令寄存器，
读操作为状态寄存器

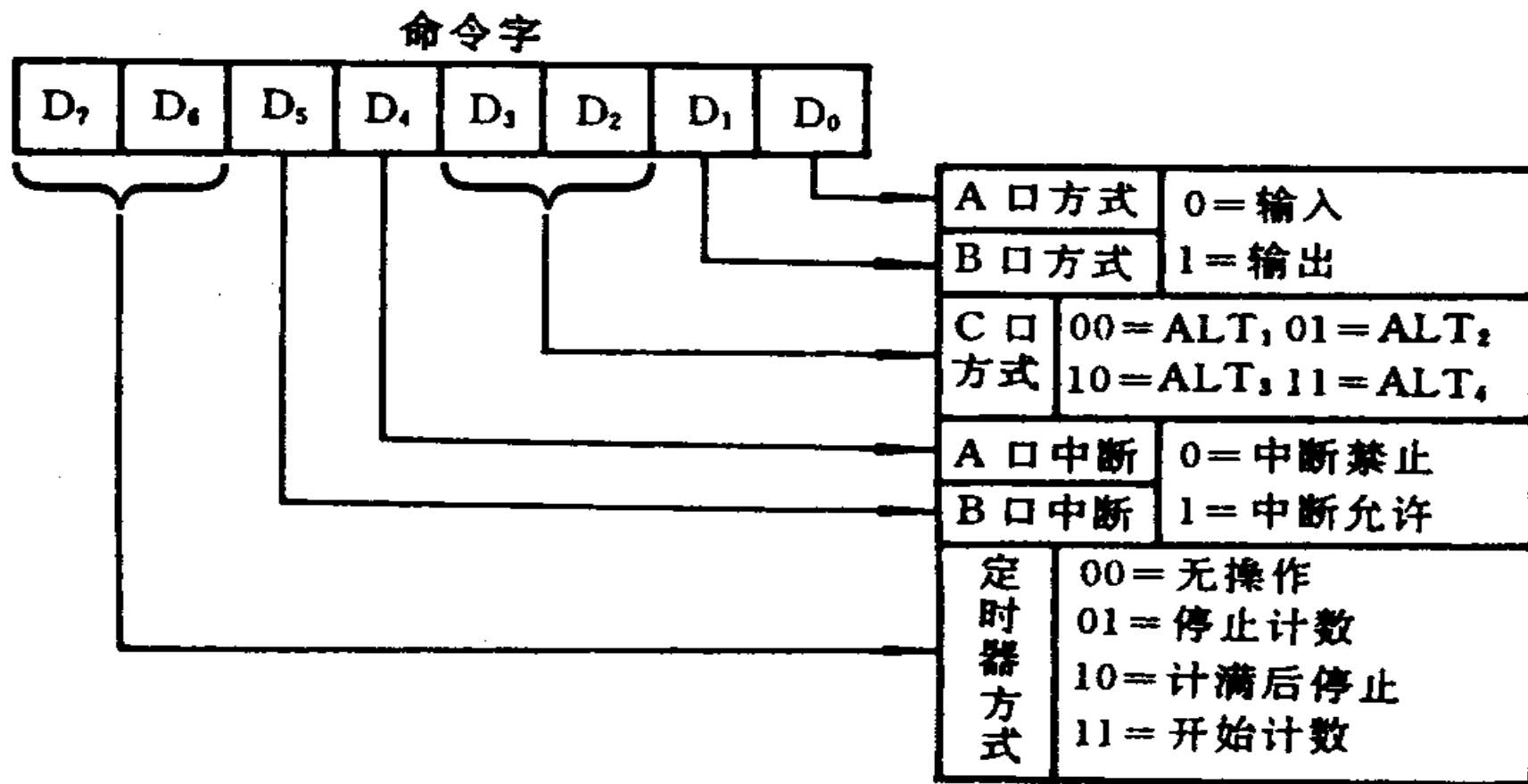


图 7.24 8155 命令字格式

8155状态字格式

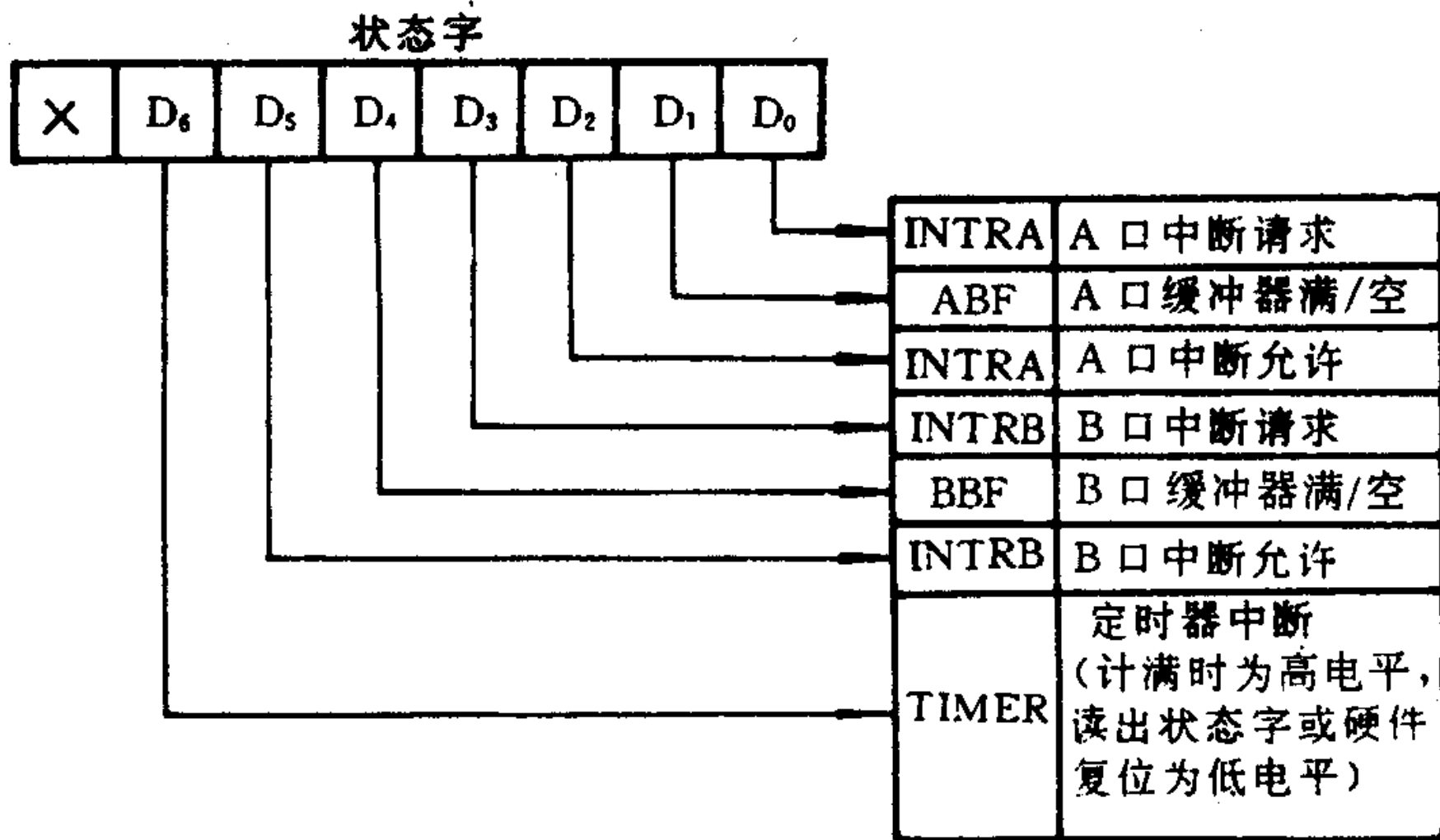


图 7.25 8155 状态字格式

8155的定时/计数器

计数结构如下：**14位减法计数器**

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
M ₂	M ₁	T ₁₃	T ₁₂	T ₁₁	T ₁₀	T ₉	T ₈

输出方式

计数器高6位

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
T ₇	T ₆	T ₅	T ₄	T ₃	T ₂	T ₁	T ₀

计数器低8位

8155定时/技术的输出方式

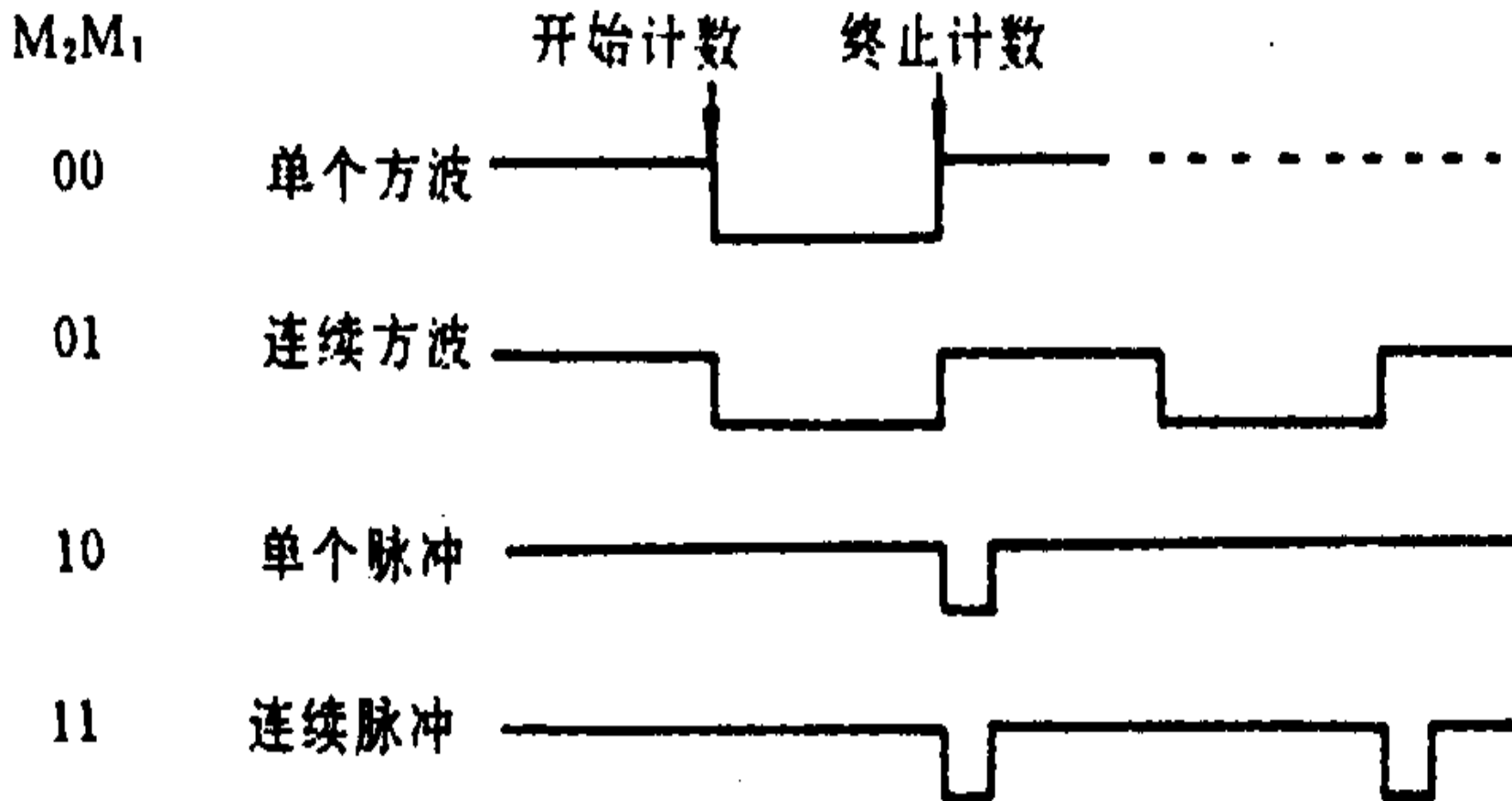


图 7.26 8155 定时器/计数器的输出方式

7.6 单片机键盘接口功能的实现

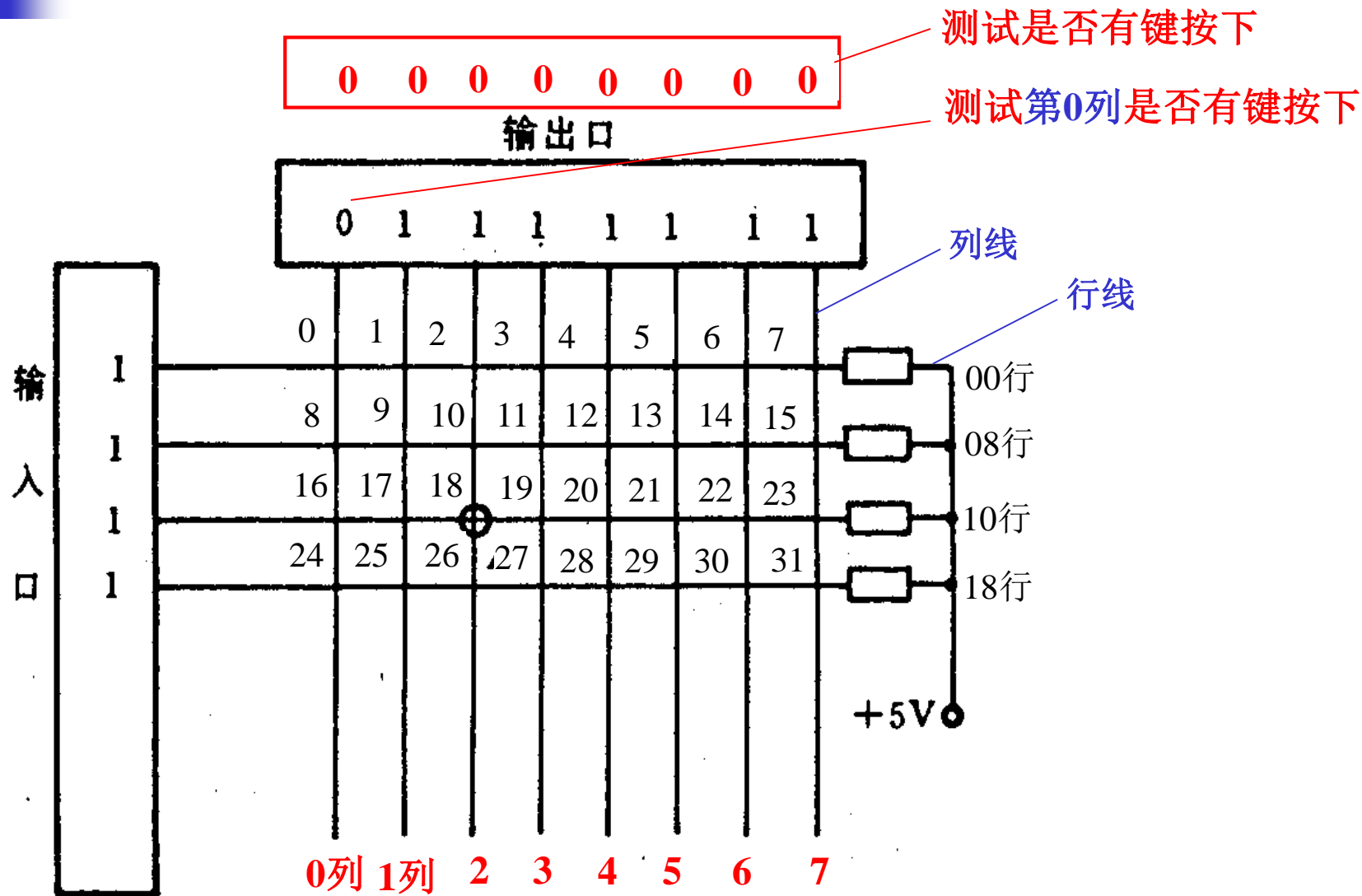
单片机键盘的种类：独立式、矩阵式

独立式键盘：每个按键独占一条口线，通过**I/O**口与单片机连接。

矩阵式键盘：按键按照行列组成矩阵。

矩阵式键盘接口的操作功能：键盘扫描、键识别、产生键码、去抖动。

测试是否有键按下



去抖动：软件延时10~20ms

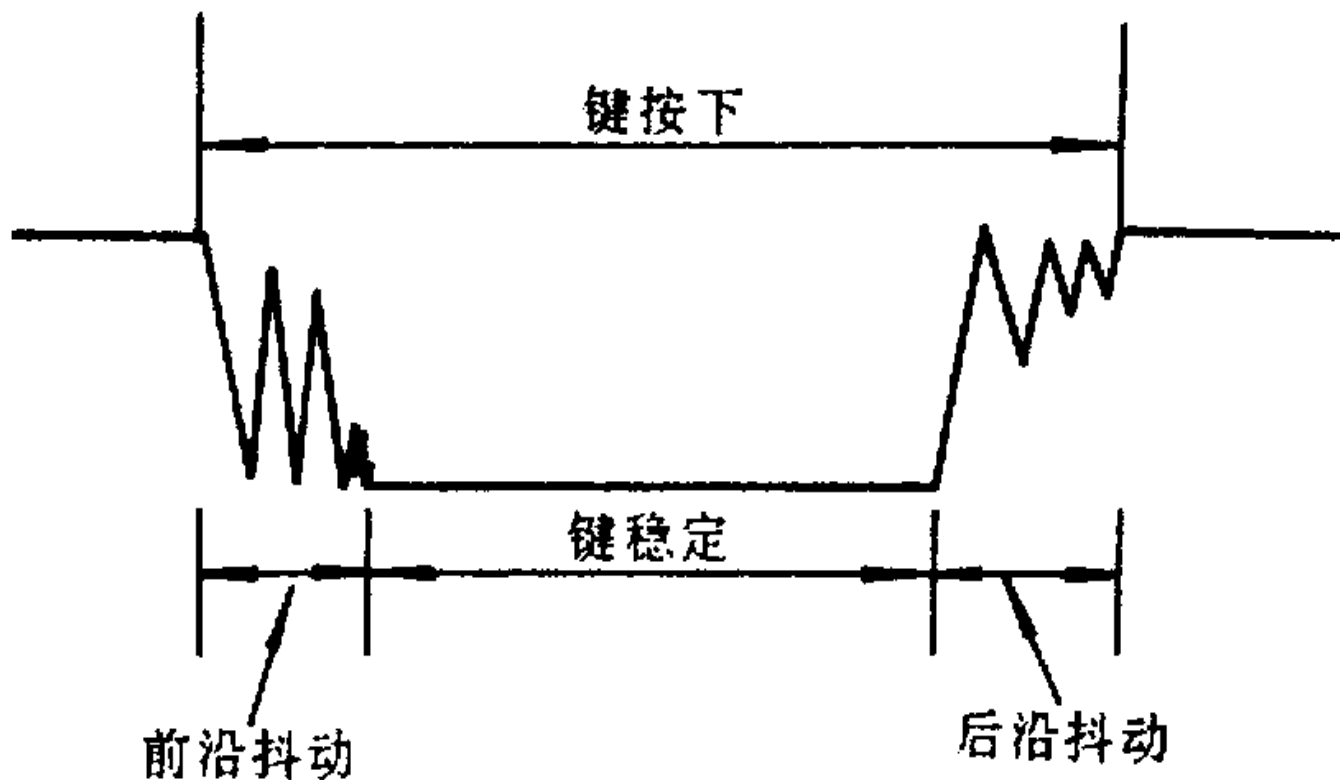


图 7.32 键闭合和断开时的电压抖动

行列扫描

输出口使第**0**列输出**0**，输入口判断行线状态，如没有低电平则输出口使第**1**列输出**0**，输入口继续判断行线状态，如此从第**0**列至第**7**列，直至找到按键位置。

键号 = 行号 + 列号。

键释放判断：延时后再测试按键，直至释放

采用8155作键盘接口

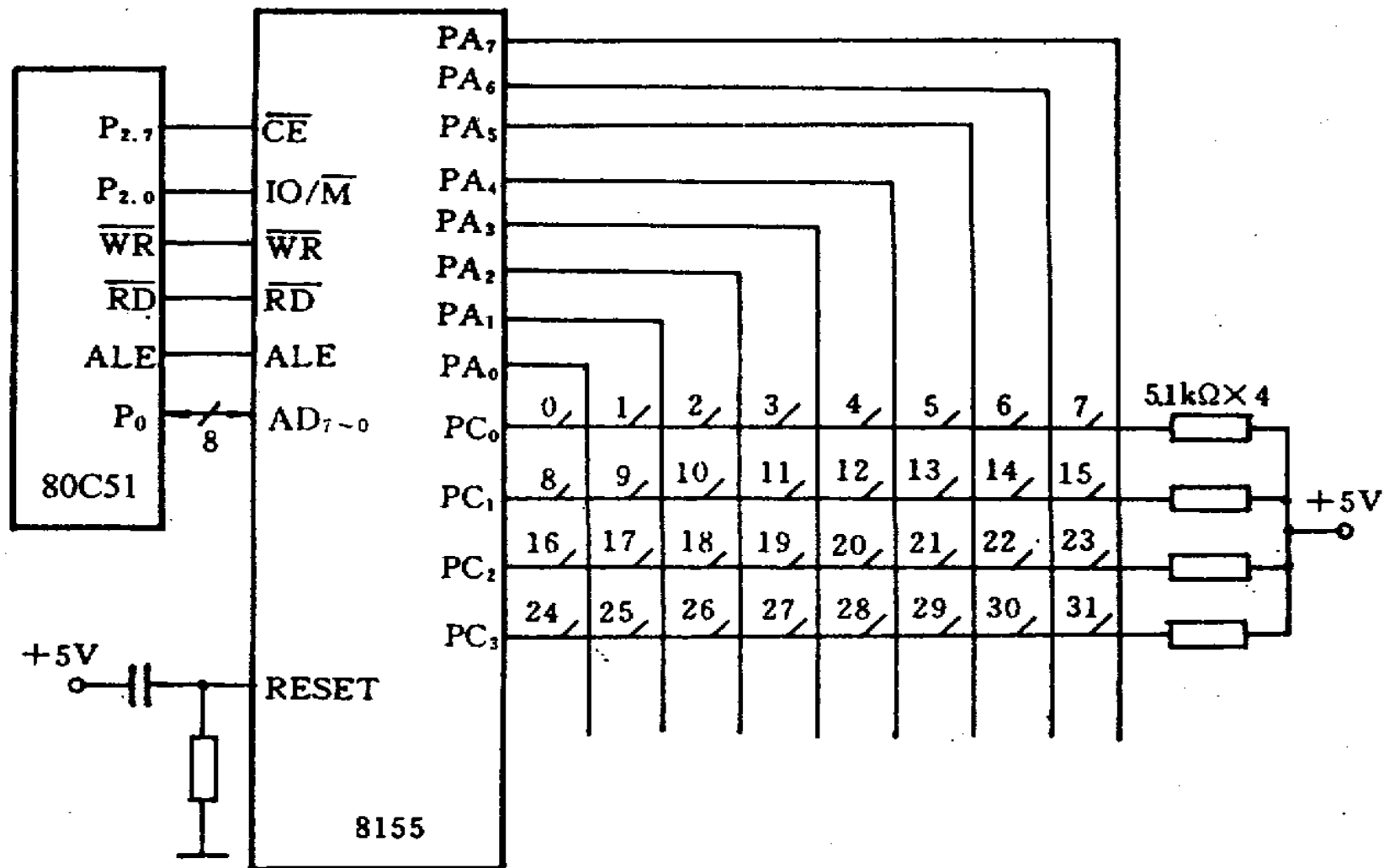
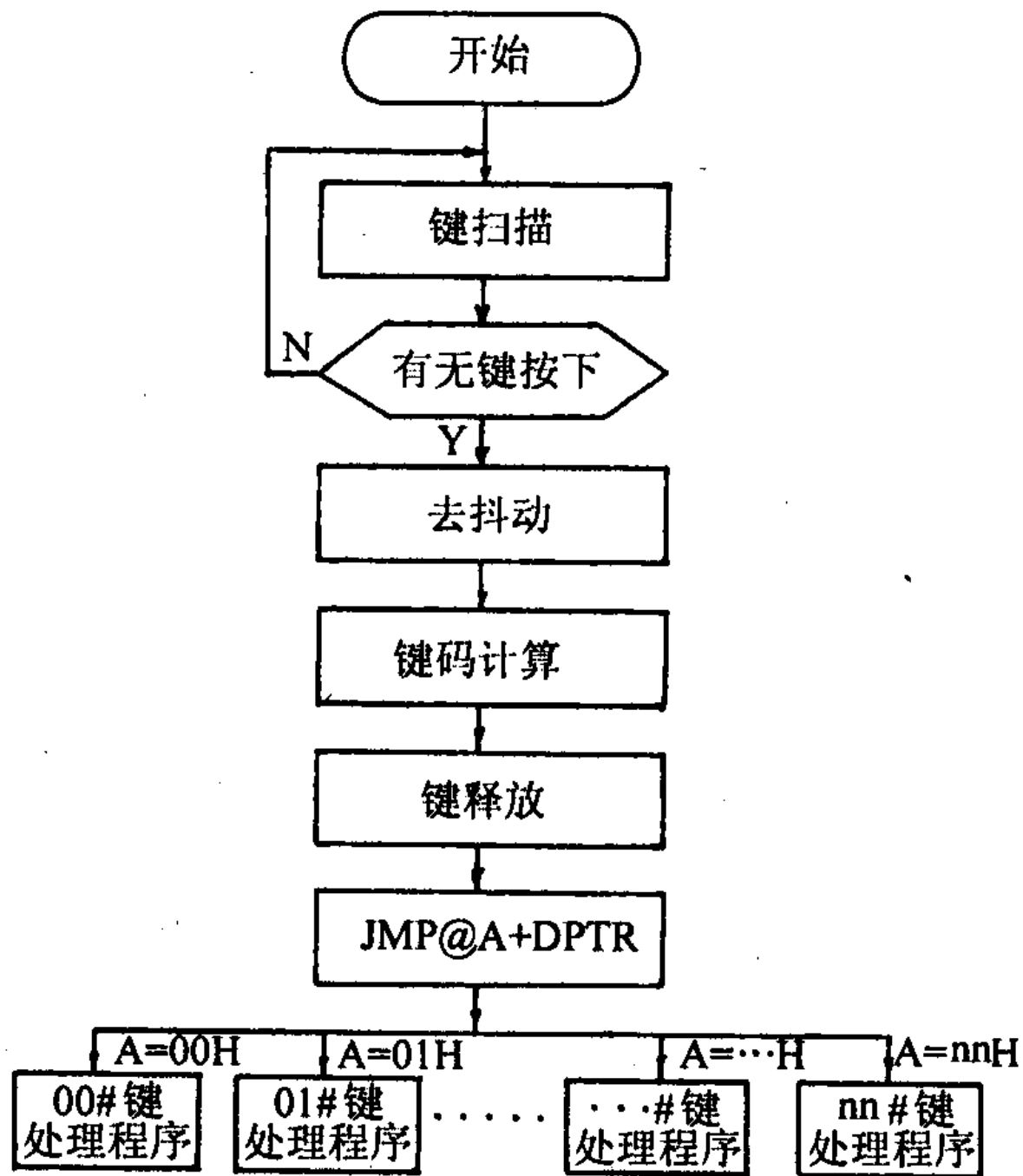


图 7.35 8155 作键盘接口

键处理流程图



判断有无闭合键的子程序

```
KS1:    MOV     DPTR, #0101H
        MOV     A, #00H           ;A 口送 00H
        MOVX    @DPTR, A
        INC     DPTR
        INC     DPTR             ;建立 C 口地址
        MOVX    A, @DPTR         ;读 C 口
        CPL     A                ;A 取反, 无键按下则全“0”
        ANL     A, #0FH          ;屏蔽 A 高半字节
        RET
```

键盘扫描程序

KEY1:	ACALL KS1	;检查有键闭合否
	JNZ LK1	;A 非“0”则转移
	ACALL DIR	;显示一次(延时 6 ms)
	AJMP KEY1	
LK1:	ACALL DIR	;有键闭合二次延时
	ACALL DIR	;共 12 ms 去抖动
	ACALL KS1	;再检查有键闭合否
	JNZ LK2	;有键闭合,转 LK2
	ACALL DIR	
	AJMP KEY1	;无键闭合,延时 6 ms 后转 KEY1
LK2:	MOV R ₂ , #0FEH	;扫描初值送 R ₂
	MOV R ₄ , #00H	;扫描列号送 R ₄
LK4:	MOV DPTR, #0101H	;建立 A 口地址
	MOV A, R ₂	
	MOVX @DPTR, A	;扫描初值送 A 口
	INC DPTR	
	INC DPTR	;指向 C 口
	MOVX A, @DPTR	;读 C 口
	JB ACC.0, LONE	;ACC.0=1,第 0 行无键闭合,转 LONE
	MOV A, #00H	;装第 0 行行值
	AJMP LKP	

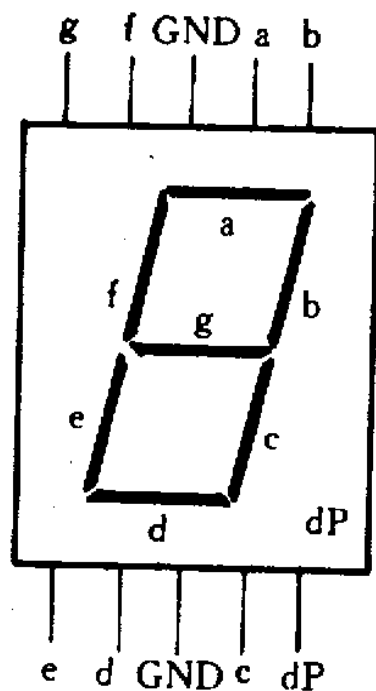
LONE:	JB	ACC. 1, LTWO	; ACC. 1=1, 第 1 行无键闭合, 转 LTWO
	MOV	A, #08H	; 装第 1 行行值
	AJMP	LKP	
LTWO:	JB	ACC. 2, LTHR	; ACC. 2=1, 第 2 行无闭合, 转 LTHR
	MOV	A, #10H	; 装第 2 行行值
	AJMP	LKP	
LTHR:	JB	ACC. 3, NEXT	; ACC. 3=1, 第 3 行无键闭合则转 NEXT
	MOV	A, #18H	; 装第 3 行行值
LKP:	ADD	A, R ₄	; 计算键码
	PUSH	ACC	; 保护键码
LK3:	ACALL	DIR	; 延时 6 ms
	ACALL	KS1	; 查键是否继续闭合, 若闭合再延时
	JNZ	LK3	
	POP	ACC	; 若键起, 则键码送 A
	RET		
NEXT:	INC	R ₄	; 扫描列号加 1
	MOV	A, R ₂	
	JNB	ACC. 7, KND	; 第 7 位为“0”, 已扫完最高列则转 KND
	RL A		; 循环左移一位
	MOV	R ₂ , A	
KND:	AJMP	LK4	; 进行下一列扫描
	AJMP	KEY1	; 扫描完毕, 开始新一轮

7.7 单片机LED显示器接口技术

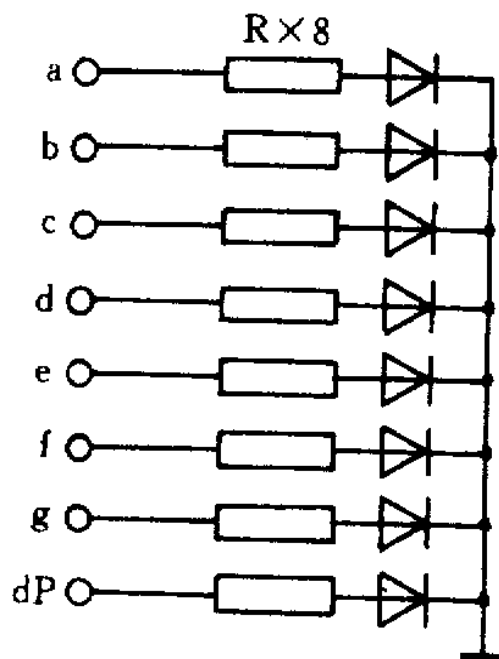
LED(发光二极管)显示器:

七段LED段码，外加小数点，共八段。

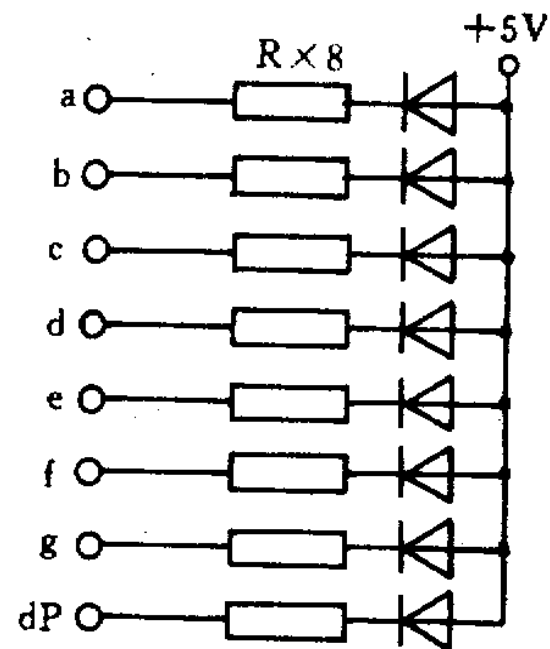
两种接法：共阴极，共阳极。



符号和引脚



共阴极



共阳极

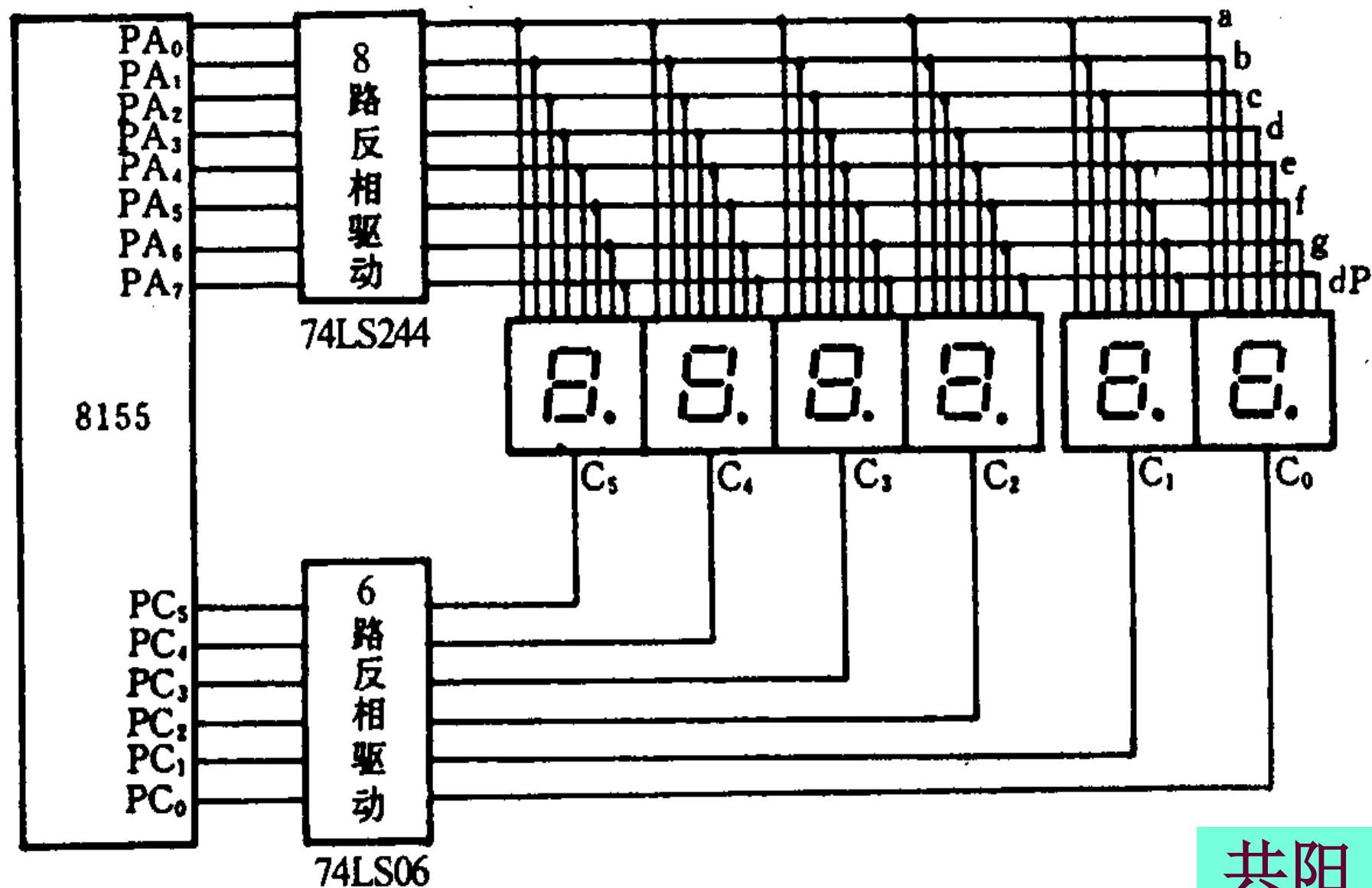
图 7.37 LED 显示器

十六进制数代码表

表 7-8 十六进制数字形代码表

字型	共阳极代码	共阴极代码	字型	共阳极代码	共阴极代码
0	C0H	3FH	9	90H	6FH
1	F9H	06H	A	88H	77H
2	A4H	5BH	b	83H	7CH
3	B0H	4FH	C	C6H	39H
4	99H	66H	d	A1H	5EH
5	92H	6DH	E	86H	79H
6	82H	7DH	F	8EH	71H
7	F8H	07H	灭	FFH	00H
8	80H	7FH			

8155作LED显示器



共阳

LED 显示程序

设显示缓冲区
为**79H~7EH**,
位控口地址
0103H, 段控
口地址**0101H**。
则**LED**显示程
序如下

DIR:	MOV	R ₀ , #79H	;建立显示缓冲区首址
	MOV	R ₃ , #01H	;从右数第一位显示器开始
	MOV	A, R ₃	;位控码初值
LD0:	MOV	DPTR, #0103H	;位控口地址
	MOVX	@DPTR, A	;输出位控码
	MOV	DPTR, #0101H	;得段控口地址
	MOV	A, @R ₀	;取出显示数据
DIR0:	ADD	A, #0DH	
	MOVC	A, @A+PC	;查表取字形代码
DIR1:	MOVX	@DPTR, A	;输出段控码
	ACALL	DL	;延时,维持点亮
	INC	R ₀	;转向下一缓冲单元
	MOV	A, R ₃	
	JB	ACC.5, LD1	;判是否到最高位,到则返回
	RL	A	;不到,向显示器高位移位
	MOV	R ₃ , A	;位控码送 R ₃ 保存
	AJMP	LD0	;继续扫描
LD1:	RET		
DSEG:	DB	C0H	;字形代码表
	DB	F9H	
	DB	A4H	
	:	:	

7.8 MCS-51单片机打印机接口技术

以 μP 系列打印机为例，其信号引脚如下：

DB7~DB0：数据线

/STB：数据选通信号，输入给打印机，
上升沿时数据读入

BUSY：忙信号，高电平有效，打印机输出，
表示打印机忙

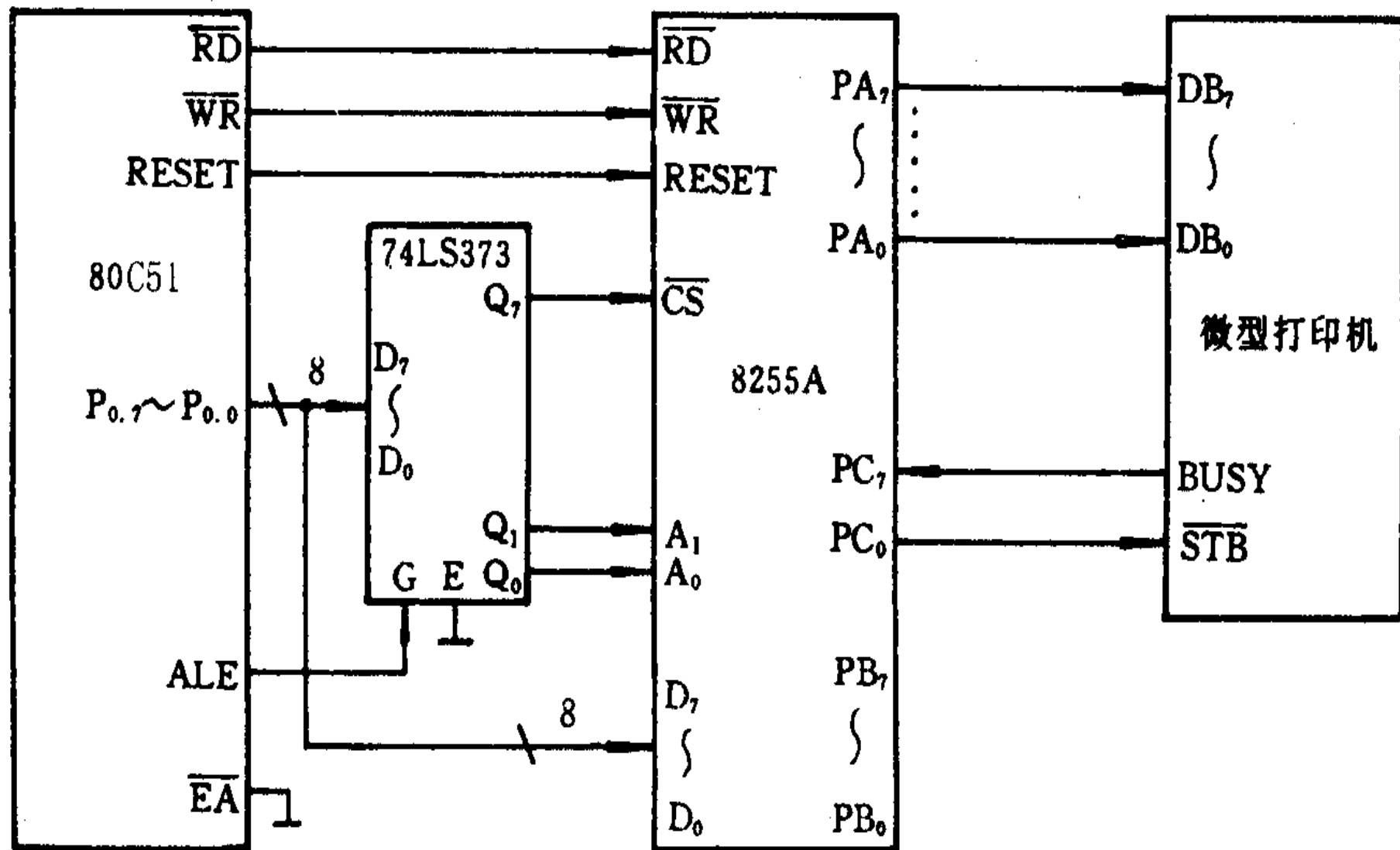
/ACK：应答信号，打印机输出，打印机
读入数据后返回应答信号

/ERR：出错信号，打印机输出。

打印机信号引脚排列

(2)	GND	GND	GND	GND	GND	GND	GND	GND	$\overline{\text{ACK}}$	$\overline{\text{ERR}}$	(20)
(1)	$\overline{\text{STB}}$	DB ₀	DB ₁	DB ₂	DB ₃	DB ₄	DB ₅	DB ₆	DB ₇	BUSY	(19)

使用8255作打印机接口



打印驱动程序

设

A口地址: $01111100=7C$;

B口地址: $01111101=7D$;

C口地址: $01111110=7E$;

控制口地址: $01111111=7F$

**A口方式: 方式0输出; B口不用;
C口高输入; C口低输出; 控制字:
 $10001000=88H$**

设**R1**存缓冲区首址, **R2**存缓冲区长度, 则驱动程序为

JB	ACC. 7, TP1	;BUSY=1,继续查询
MOV	R ₀ , #7CH	;A 口地址
MOV	A, @R ₁	;取缓冲区数据
MOVX	@R ₀ , A	;打印数据送 8255A
INC	R ₁	;指向下一单元
MOV	R ₀ , #7FH	;控制口地址
MOV	A, #00H	;输出 $\overline{\text{STB}}$ 脉冲
MOVX	@R ₀ , A	
MOV	A, #01H	
MOVX	@R ₀ , A	
DJNZ	R ₂ , TP	;数据长度减 1,不为 0 继续
RET		