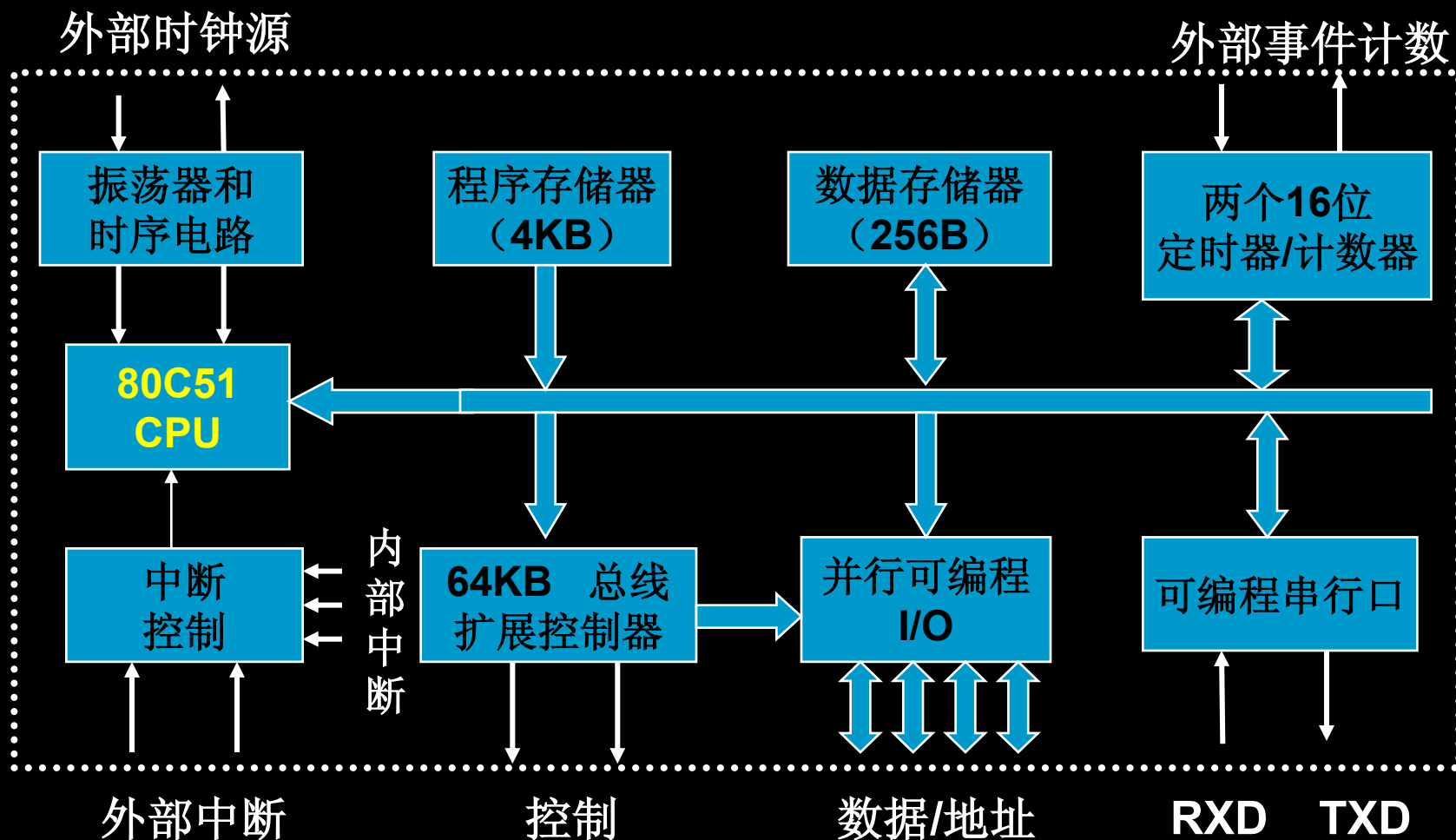
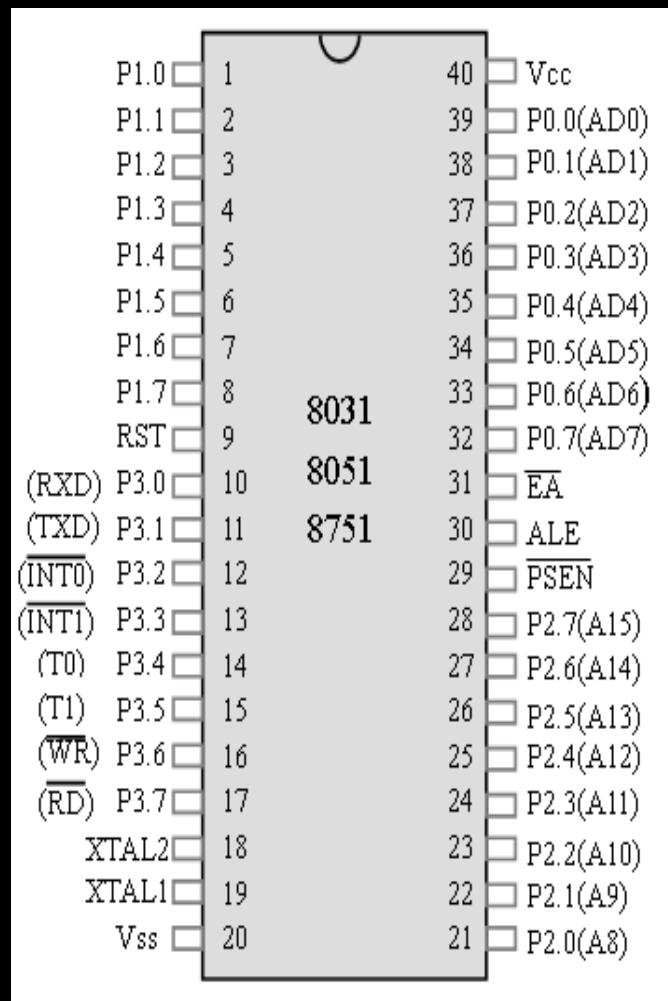


第五部分 功能部件使用



§ 5.1 P0~P3端口结构与应用

80C51有四个8位的并行双向口**P0**、**P1**、**P2**、**P3**，计有**32**根输入输出线。各口的每一位均有锁存器、输出驱动器和输入缓冲器组成。由于它们在结构上存在一些差异，故各口的性质和功能也有差异。它们之间的区别如图。



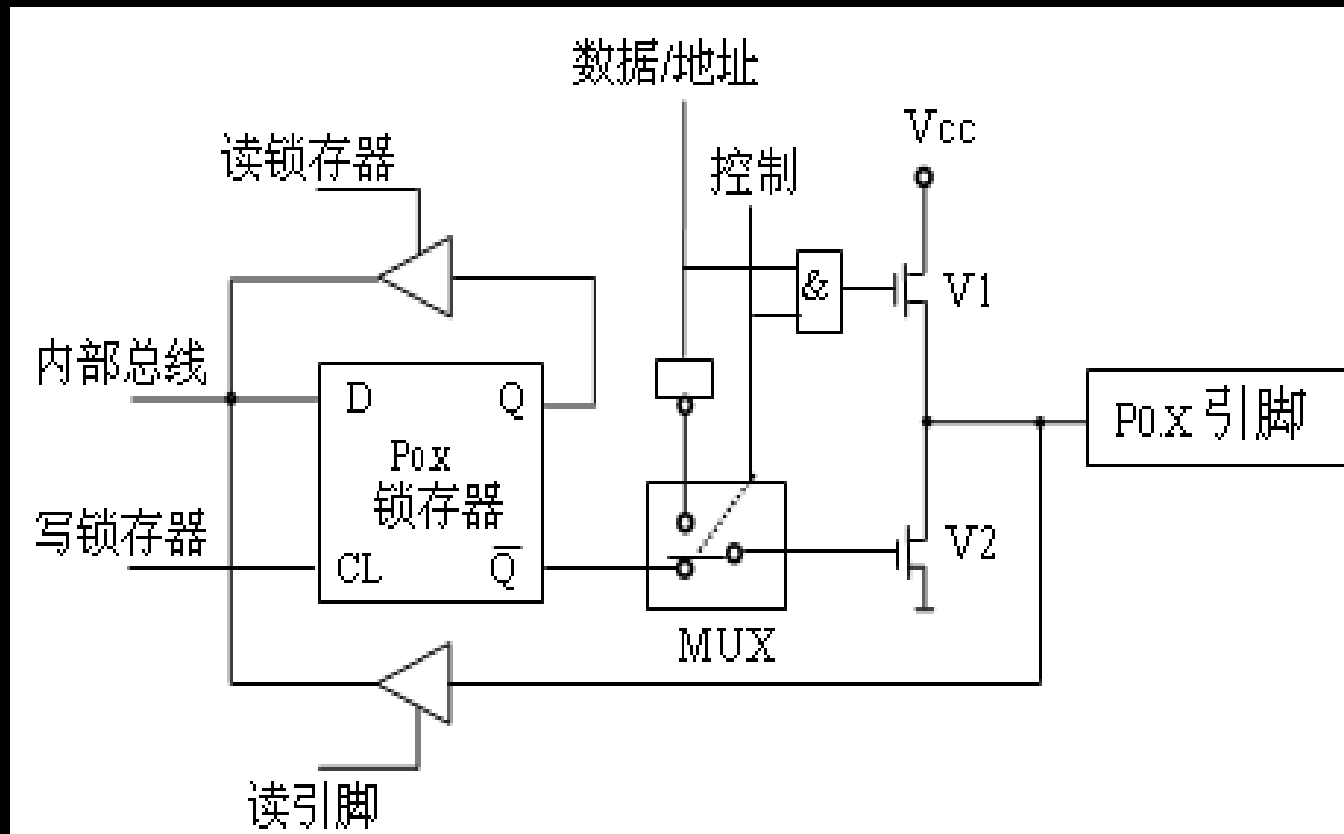
一、P0口

P0口是一个多功能的8位口，可以字节访问，也可以位访问，其字节地址为80H，位地址为80~87H。

1、P0口的功能和特点：

- 做输入输出时，需加上拉电阻。
- 并行扩展时，做地址/数据复用总线。输出地址总线的低8位（A7~A0）和数据总线（D7~D0）。
- 能驱动8个TTL负载。

2、P0口的结构



做输入输出口时，P0要加上拉电阻。做输入时要先输出1，将口设为输入状态。

二、 P1口

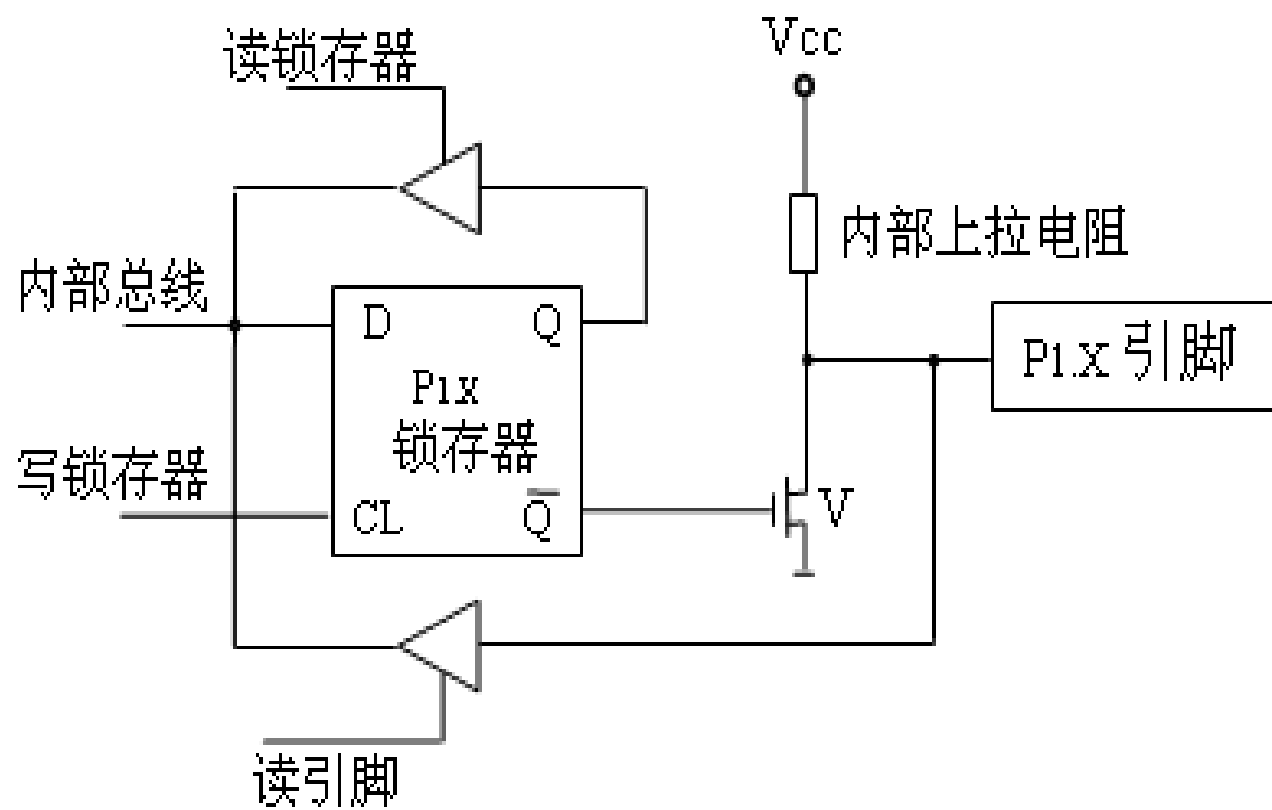
1、 P1口是一个8位口

可以字节访问也可位访问，其字节地址为90H，位地址为90H~97H。

2、 P1口的特点

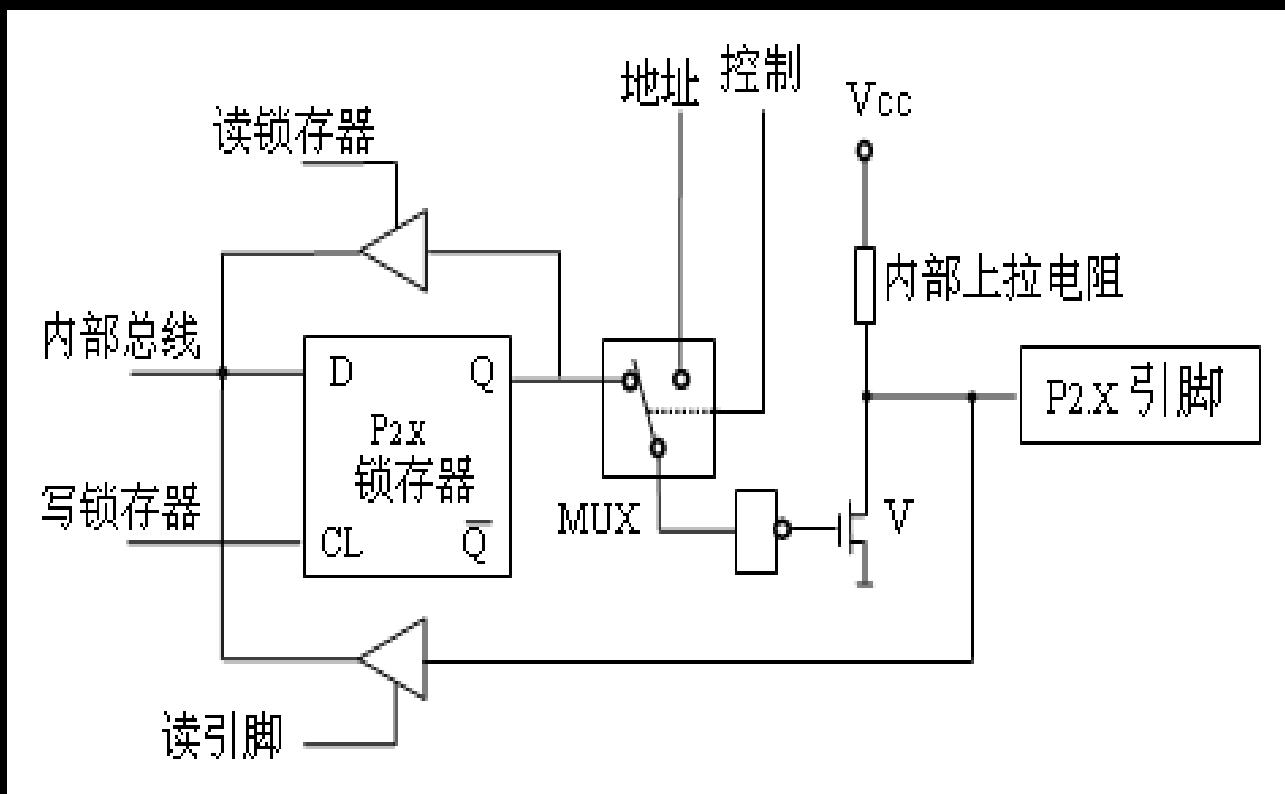
- 1) 无条件输出，输出带锁存；
- 2) 输入缓冲，输入时有条件，即需要先输出1，将口设为输入状态；

P0、P1、P2、P3同样具有上述两个特点。



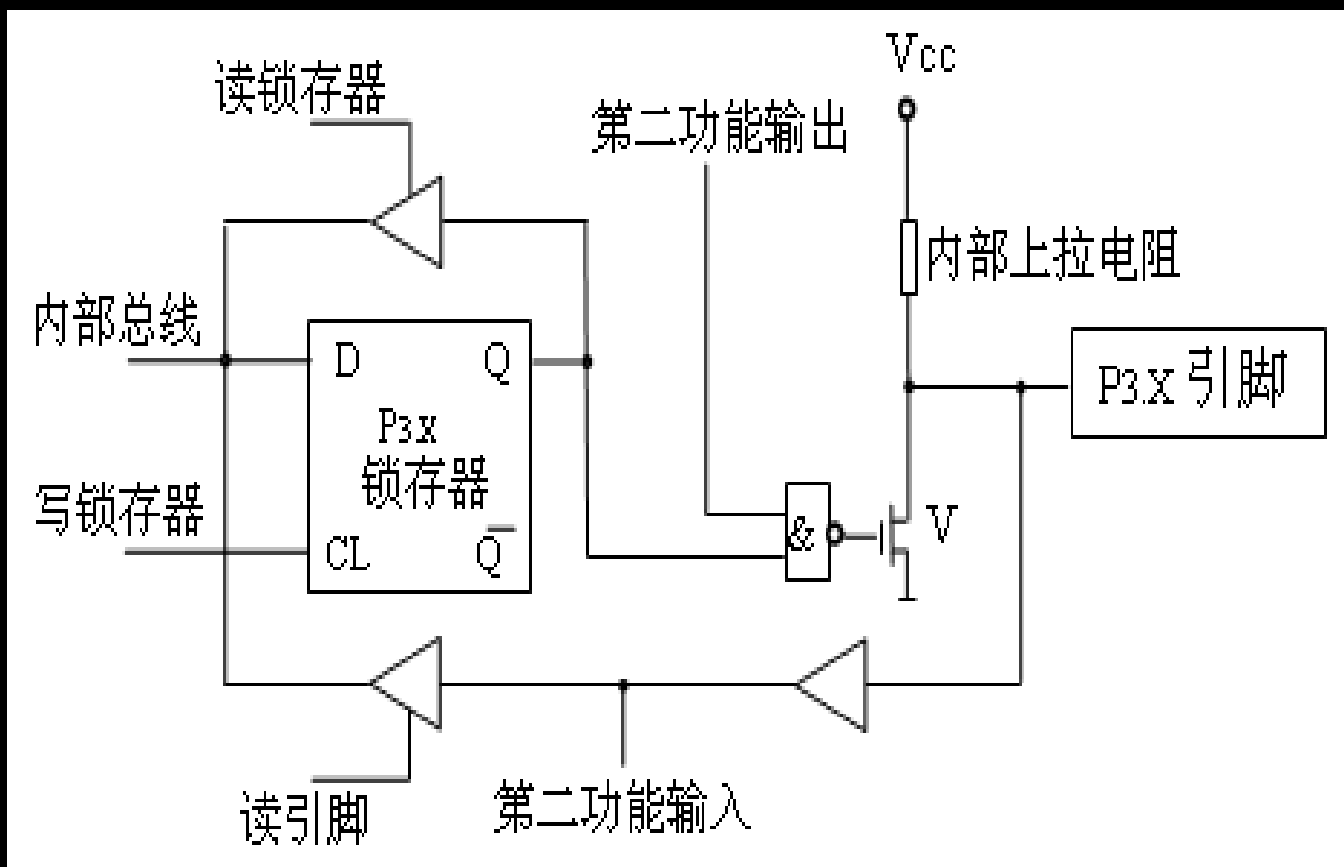
三、P2口

P2口是一个多功能的8位口，可以字节访问也可以位访问，其字节地址位A0H，位访问地址为A0H~A7H。在做并行扩展时，做为地址线的高8位（A8~A15）。



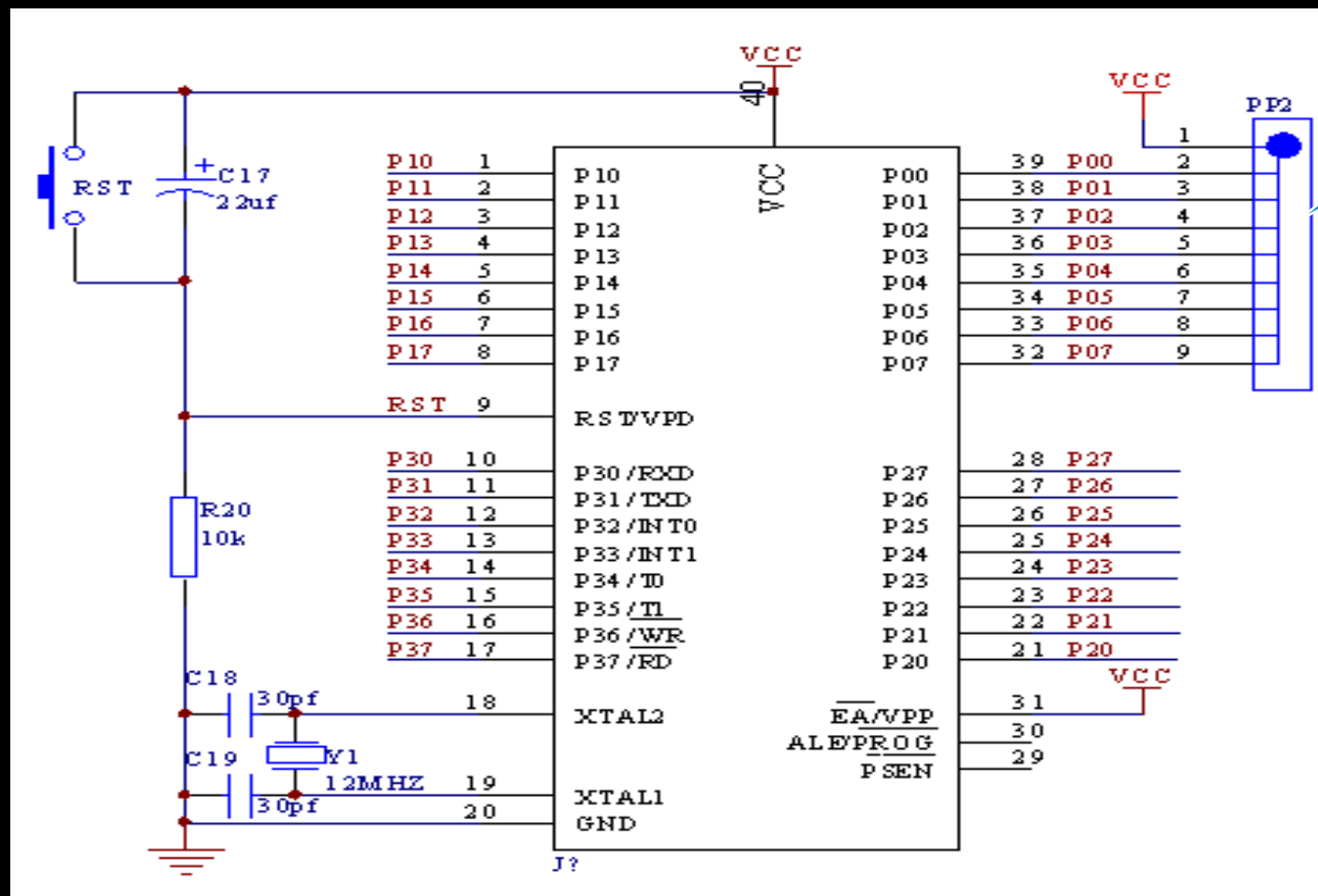
四、P3口

P3口是一个**多功能8位口**，可以访问字节也可访问位。
其字节访问地址为B0H, 位访问地址为B0H~B7H。



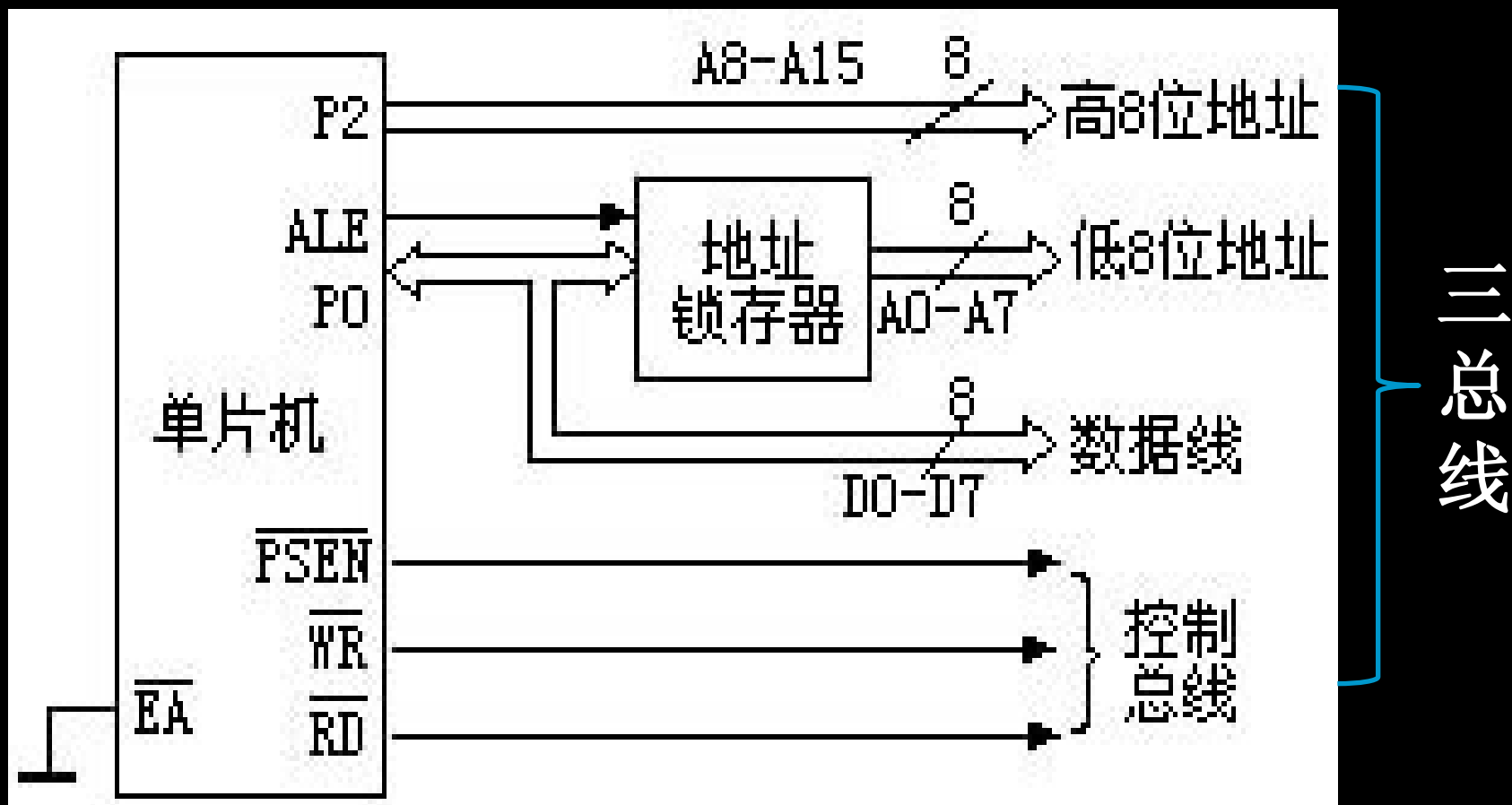
51最小应用系统

P0~P3口做普通的输入输出口。P0要加上拉电阻。
输入时有条件，要先输出1，将口设为输入状态。



五、并行扩展举例

三总线结构形成



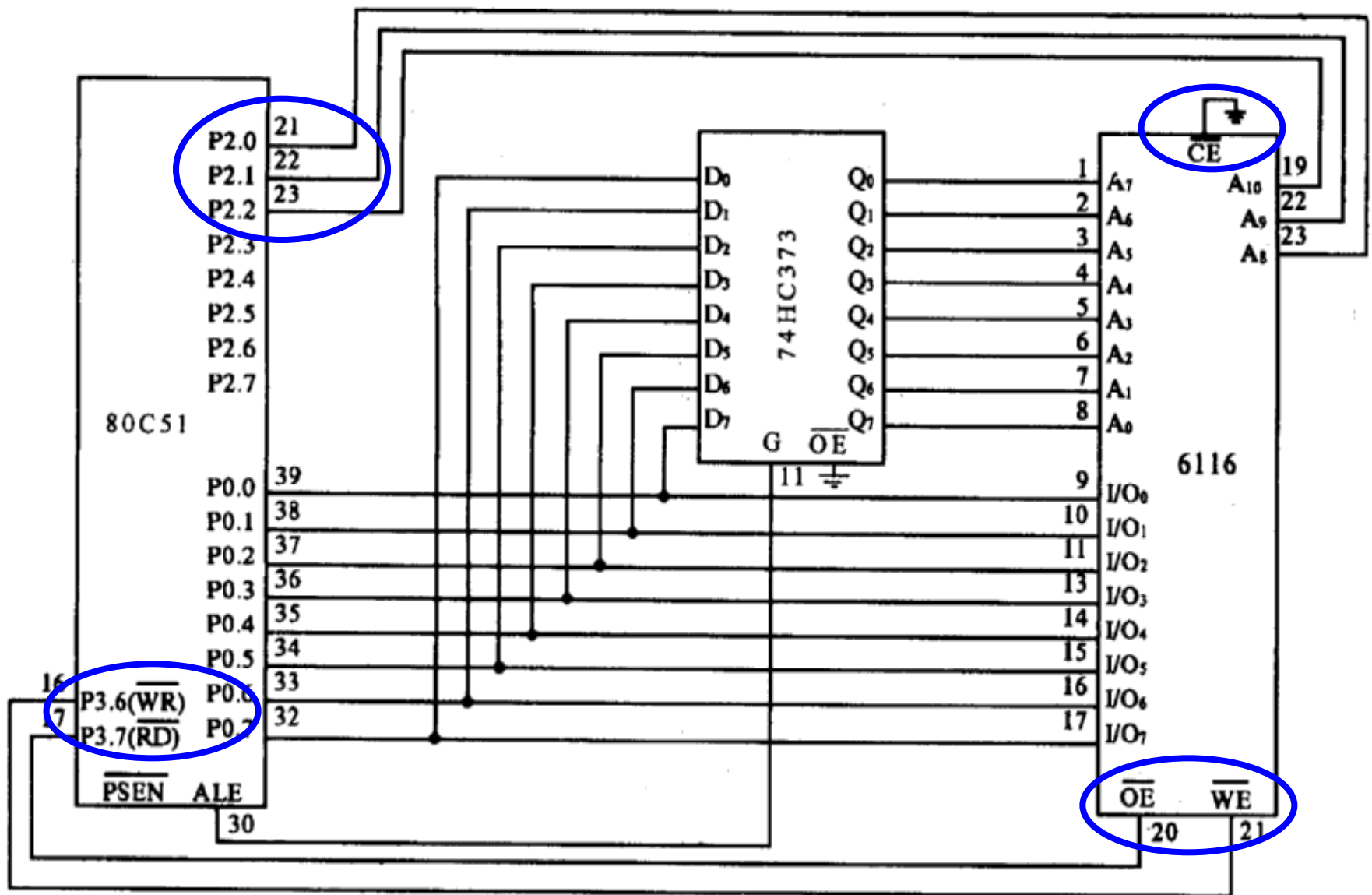
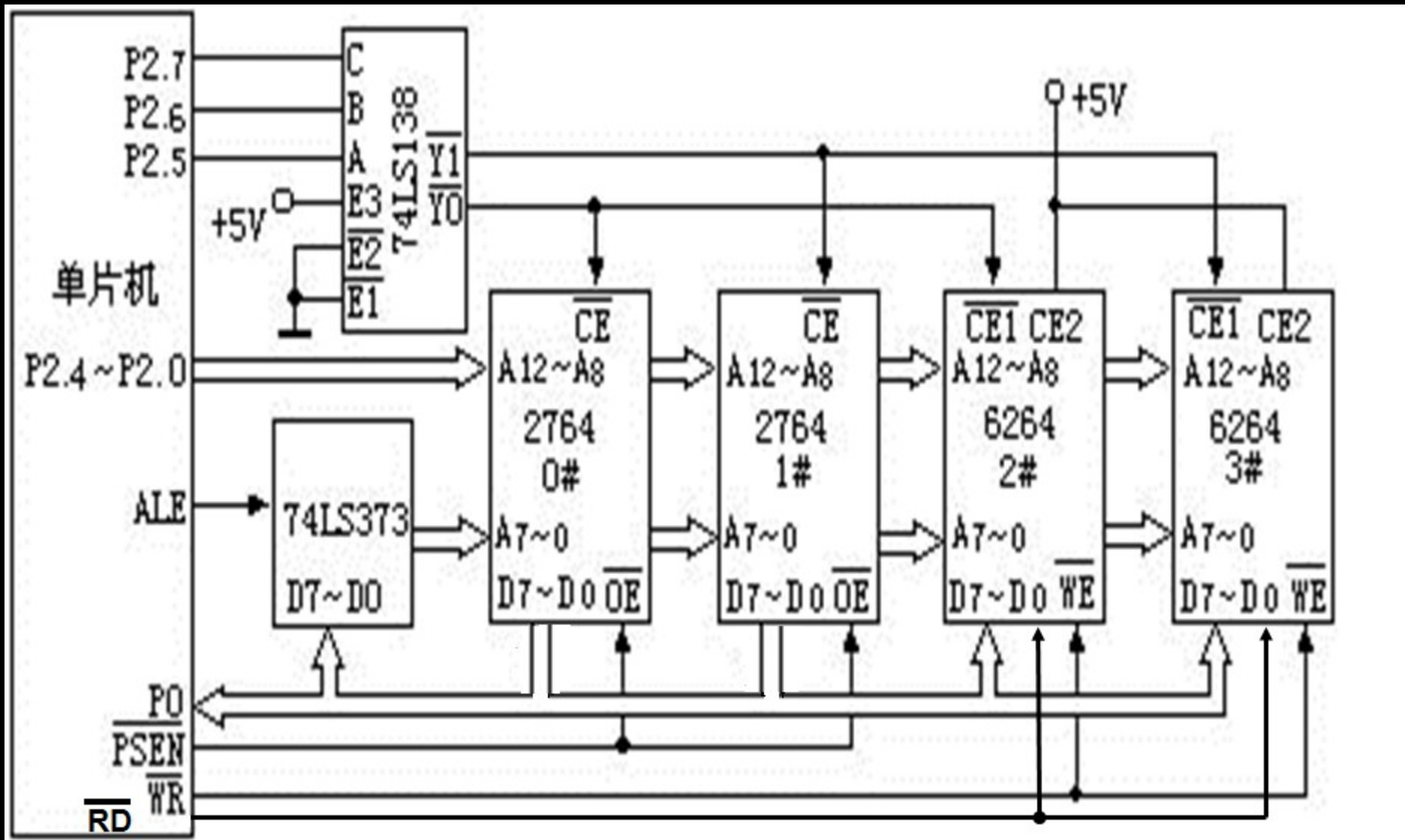


图 9-3 扩展 2 KB RAM

6116的地址空间为： 0000H ~ 07FFH



0#、2#地址：0000H ~ 1FFFH
 1#、3#地址：2000H ~ 3FFFH

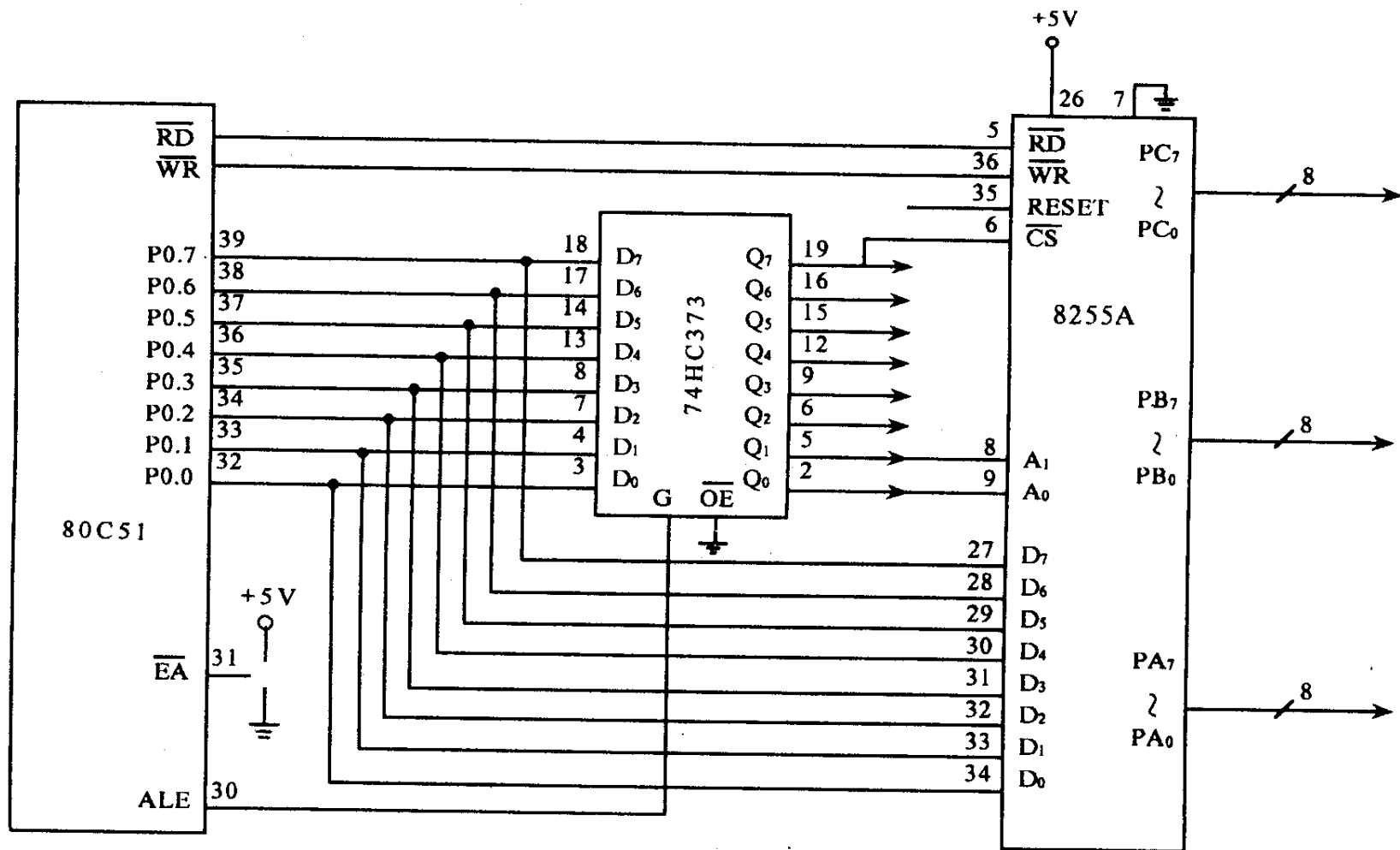
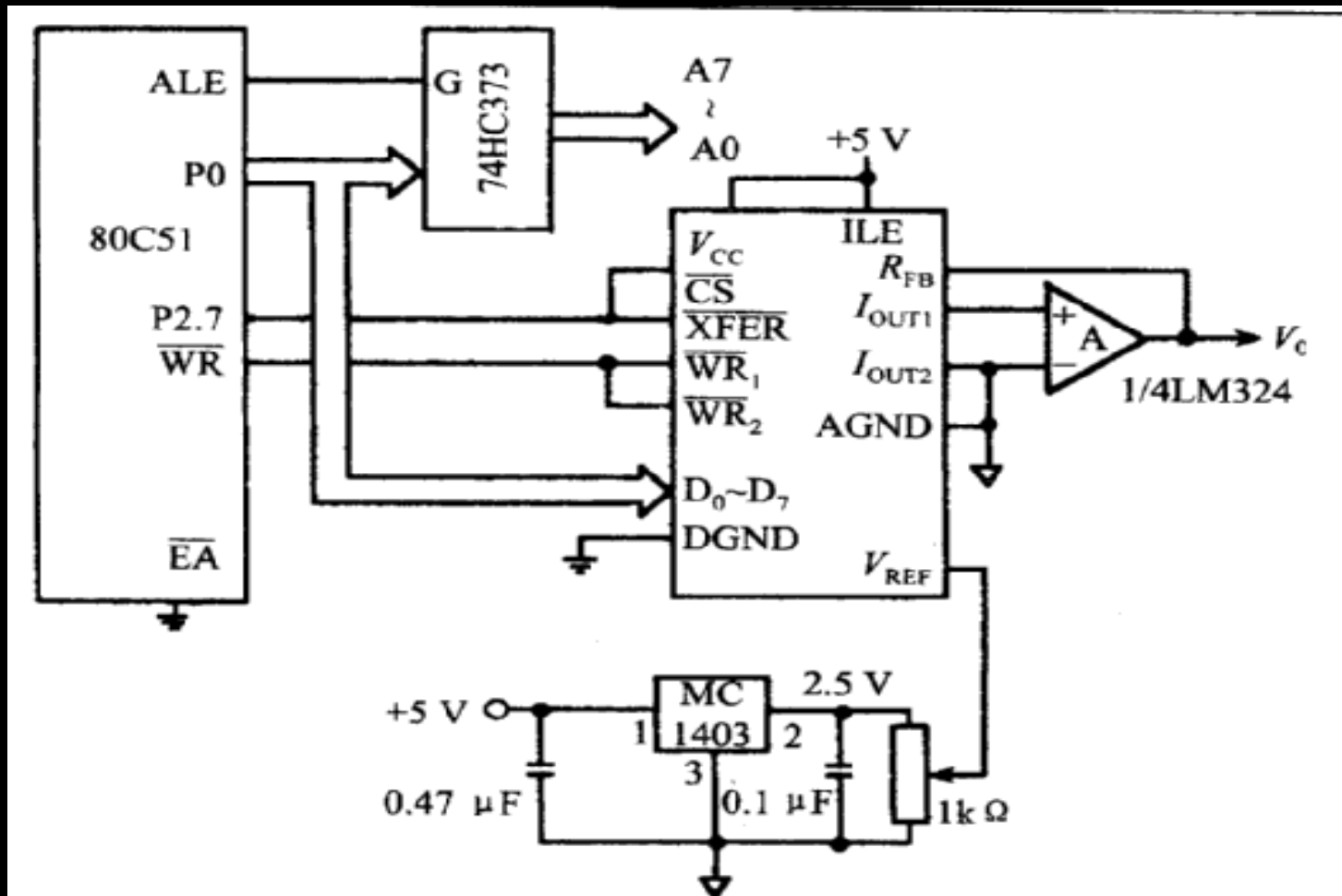


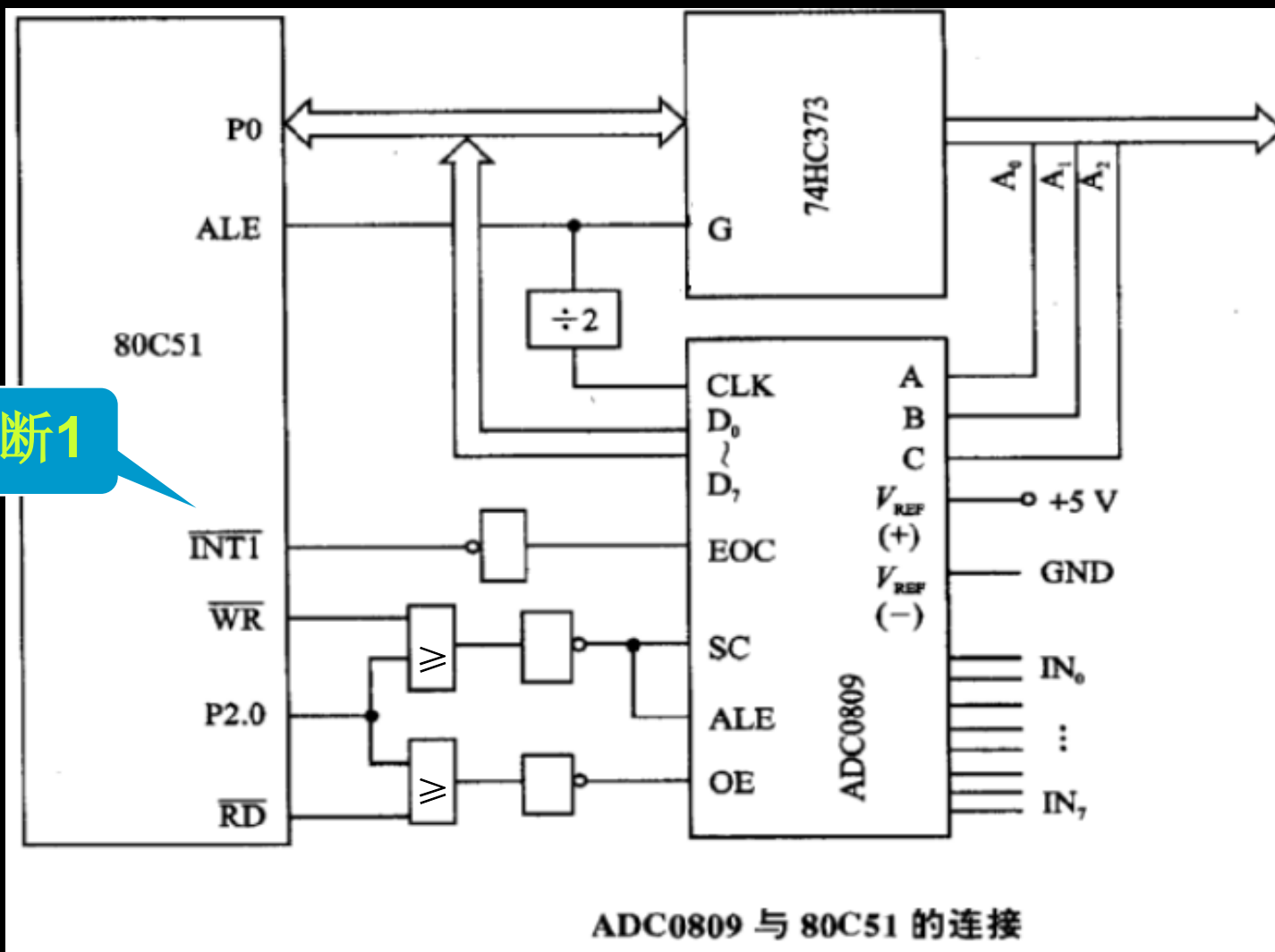
图 9-9 80C51 与 8255 的接口电路

8255地址: XXXXXXXX 0XXX XX00、
 XXXXXXXX 0XXX XX01、
 XXXXXXXX 0XXX XX10、
 XXXXXXXX 0XXX XX11

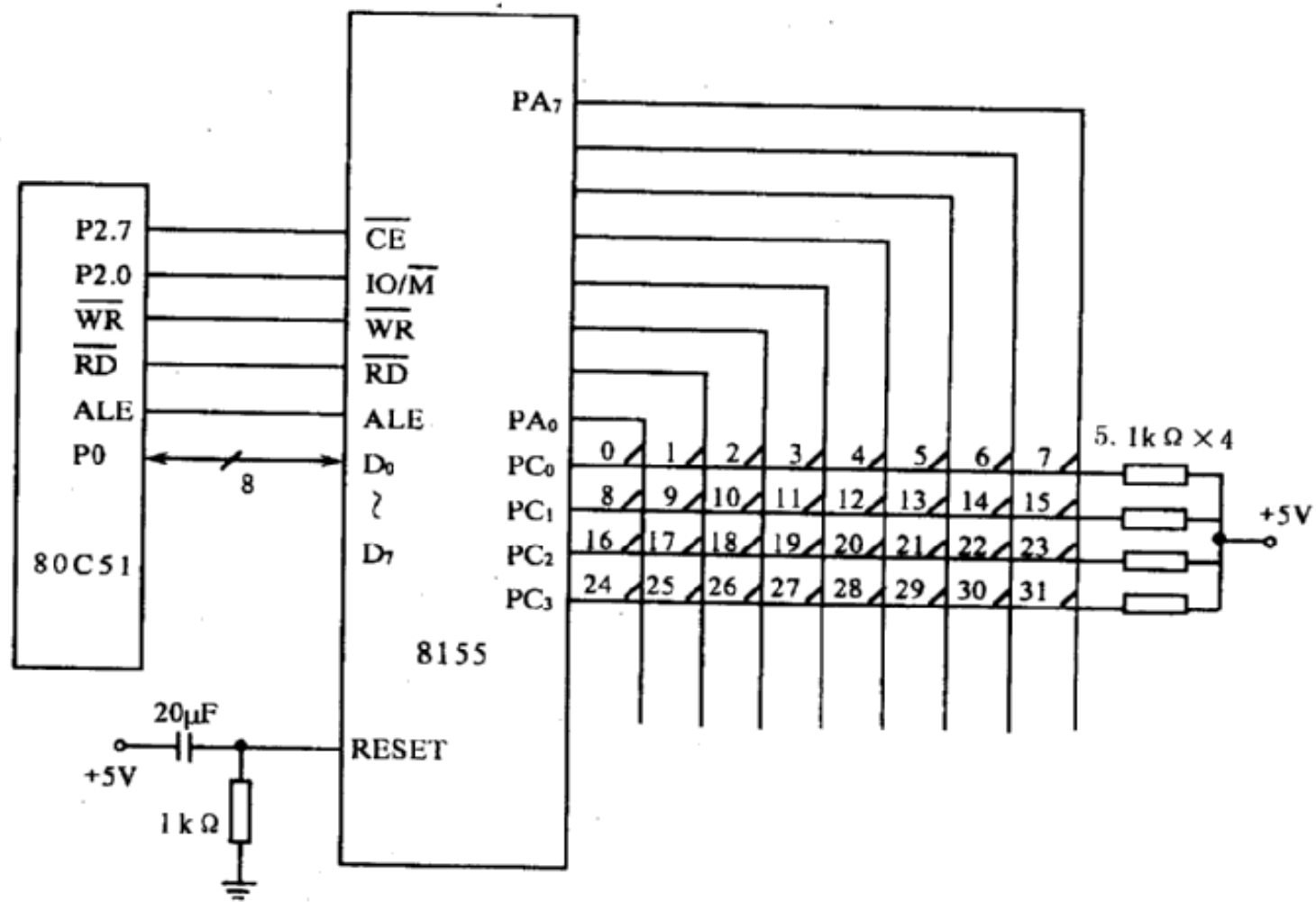


DAC0832地址: 0XXX XXXX XXXX XXXX

外中断1



ADC0809地址: XXXX XXX0 XXXX X000、
 XXXX XXX0 XXXX X001、 . . .
 XXXX XXX0 XXXX X111



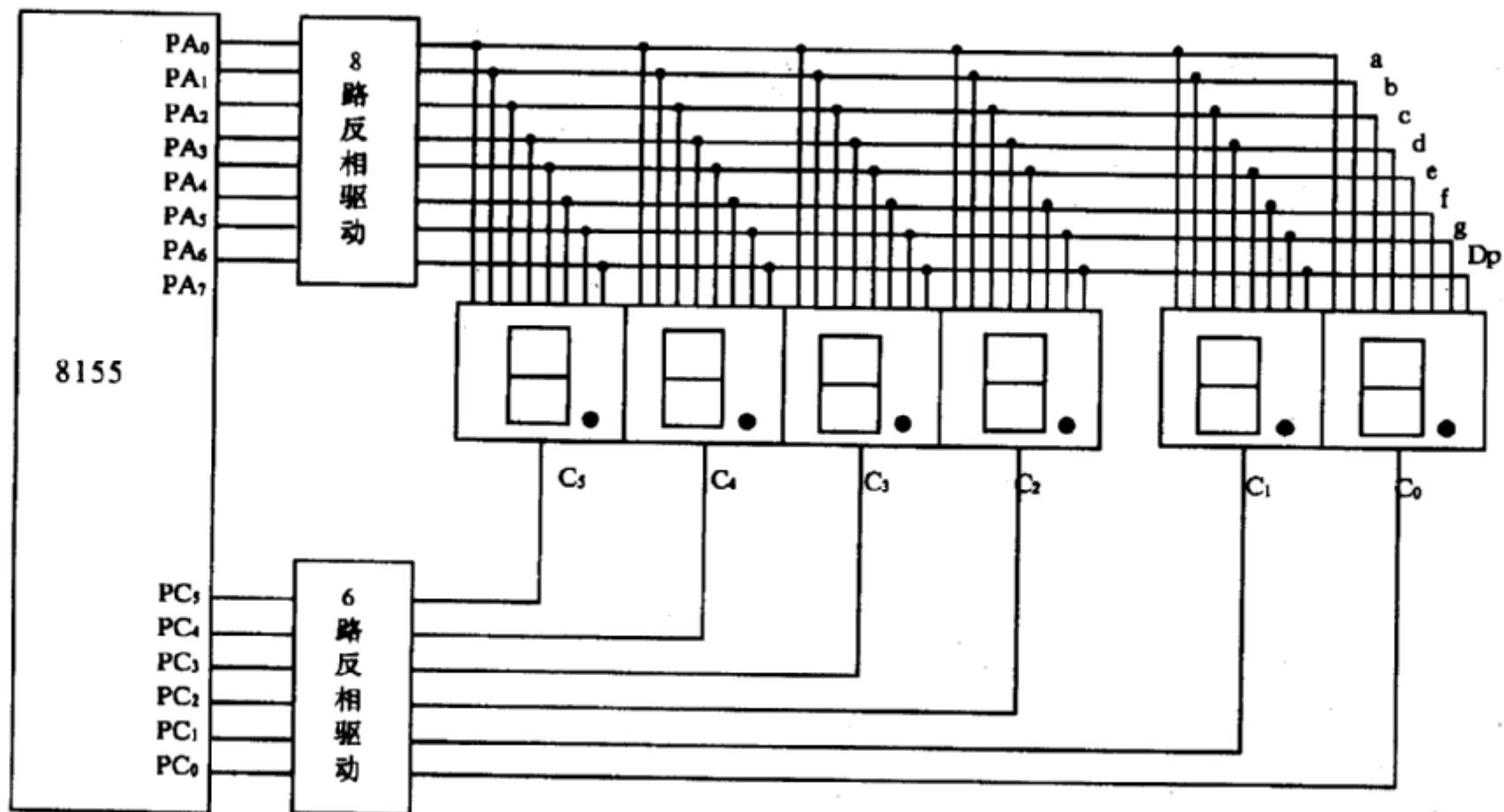


图 10-8 扫描式显示电路

§ 5.2 定时器/计数器

80C51包含有两个16位的定时器/计数器：T0、T1；

在单片机中，定时功能和计数功能的设置和控制都是通过软件进行的。

定时器/计数器的核心是一个加1计数器，其基本功能是加1功能。

一、 定时器/计数器的结构和原理

1、 定时器/计数器T0、T1的内部结构

- 计数器TH0、TL0和TH1、TL1;
- 特殊功能寄存器TMOD、TCON;
- 时钟分频器
- 输入引脚T0 (P03.4)、T1 (P03.5)
/INT0、/INT1

2、 最短定时周期、最大计数频率

- 定时： 每一个机器周期，计数器加1;
- 计数： 输入引脚T0 (P03.4)、T1 (P03.5)
上的脉冲频率最大为时钟频率的1/24。

3、定时器/计数器T0、T1的特殊功能寄存器

(1) 定时器/计数器T0、T1的方式寄存器—TMOD

方式寄存器TMOD是一个逐位定义的8位寄存器，是只能字节寻址的寄存器，字节地址为89H。

D7	D6	D5	D4	D3	D2	D1	D0
GATE	C/ \overline{T}	M1	M0	GATE	C/ \overline{T}	M1	M0

← T1 → T0 →

当**GATE=0**时，T0、T1的启动只受TR_x控制，TR_x =1则启动，TR_x=0则不启动。

当**GATE=1**时，**T0、T1的启动受TR_x和INT_x控制**，当INT_x为高电平时，TR_x置位，才能启动。

TMOD只能按字节寻址

前4位控制T1
与后4位功能相同

M1,M0: 工作方式
0 , 0: T0方式0
0 , 1: T0方式1
1 , 0: T0方式2
1 , 1: T0方式3



0: T0启动与INT0无关
1: T0启动与INT0有关

0: T0定时器模式
1: T0计数器模式

(2) 定时器/计数器T0、T1的控制寄存器—TCON

控制寄存器TCON是一个逐位定义的8位寄存器，既可字节寻址也可以位寻址，字节地址为**88H**，位寻址的地址为**88H~8FH**。

位地址	8FH	8EH	8DH	8CH	8BH	8AH	89H	88H
位功能	TF1	TR1	TF0	TR0				

T1的溢出标志

T1的运行控制位

T0的溢出标志

T0的运行控制位

(3) 定时器/计数器T0、T1的数据寄存器 — TH1、TL1, TH0、TL0。

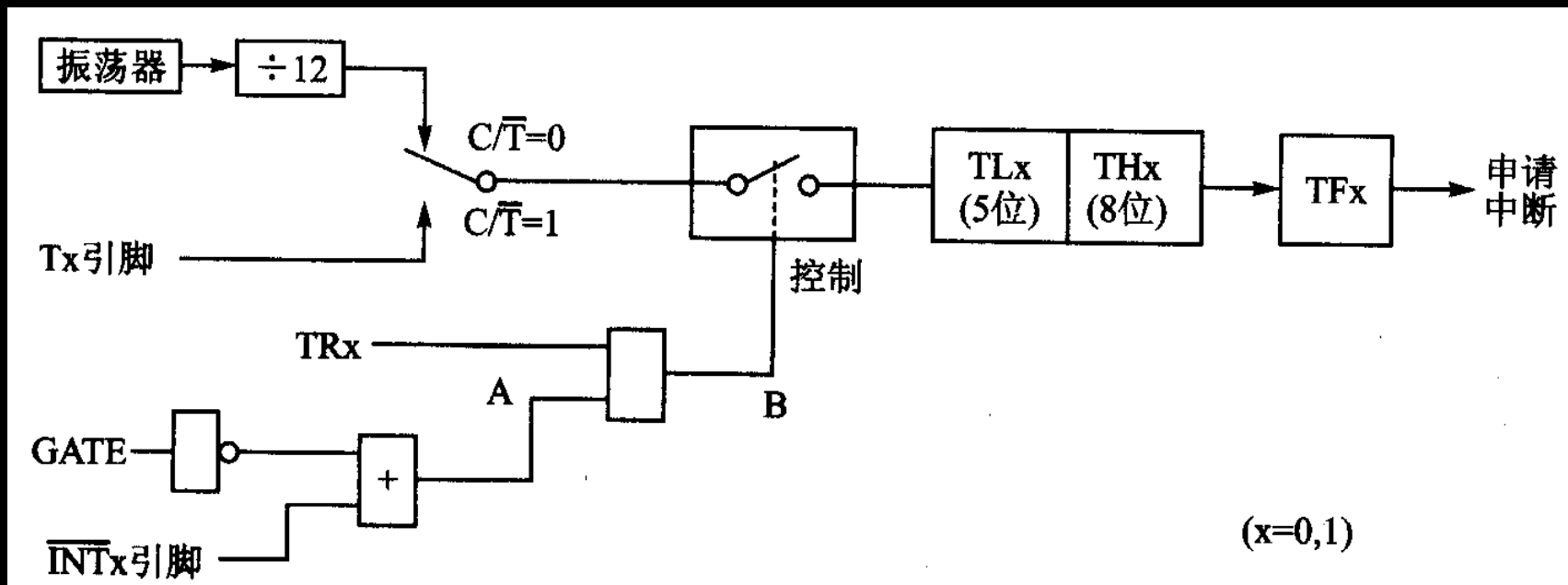
T0、T1各有一个16位数据计数器，可分为高8位和低8位。它们是读写寄存器，任何时候可以进行读写。复位后清零。

5、定时器/计数器T0、T1的工作方式

(1) 方式0

计数寄存器由13位组成，即TLx的高3位未用。

计数时，TLx的低5位溢出后向THx进位，THx溢出后将TFx置位，并向CPU申请中断。

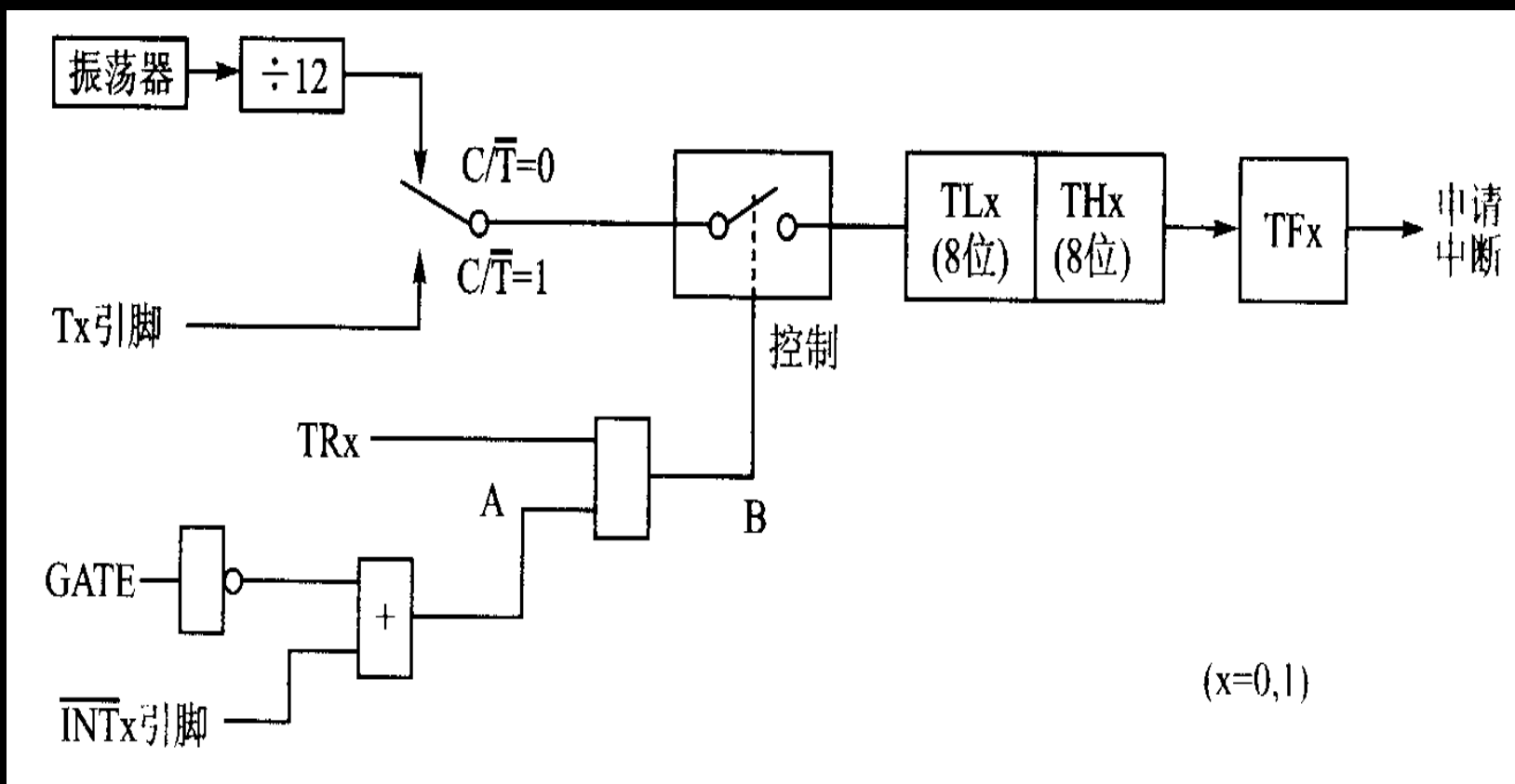


当**GATE=0**时，A点为高电平，定时器/计数器的启动/停止由**TRx**决定。**TRx=1**，定时器/计数器启动；**TRx=0**，定时器/计数器停止。

当**GATE=1**时，A点的电位由**INTx**决定，因而B点的电位就由**TRx**和**INTx**决定，即定时器/计数器的启动/停止由**TRx**和**INTx**两个条件决定。

(2) 方式1

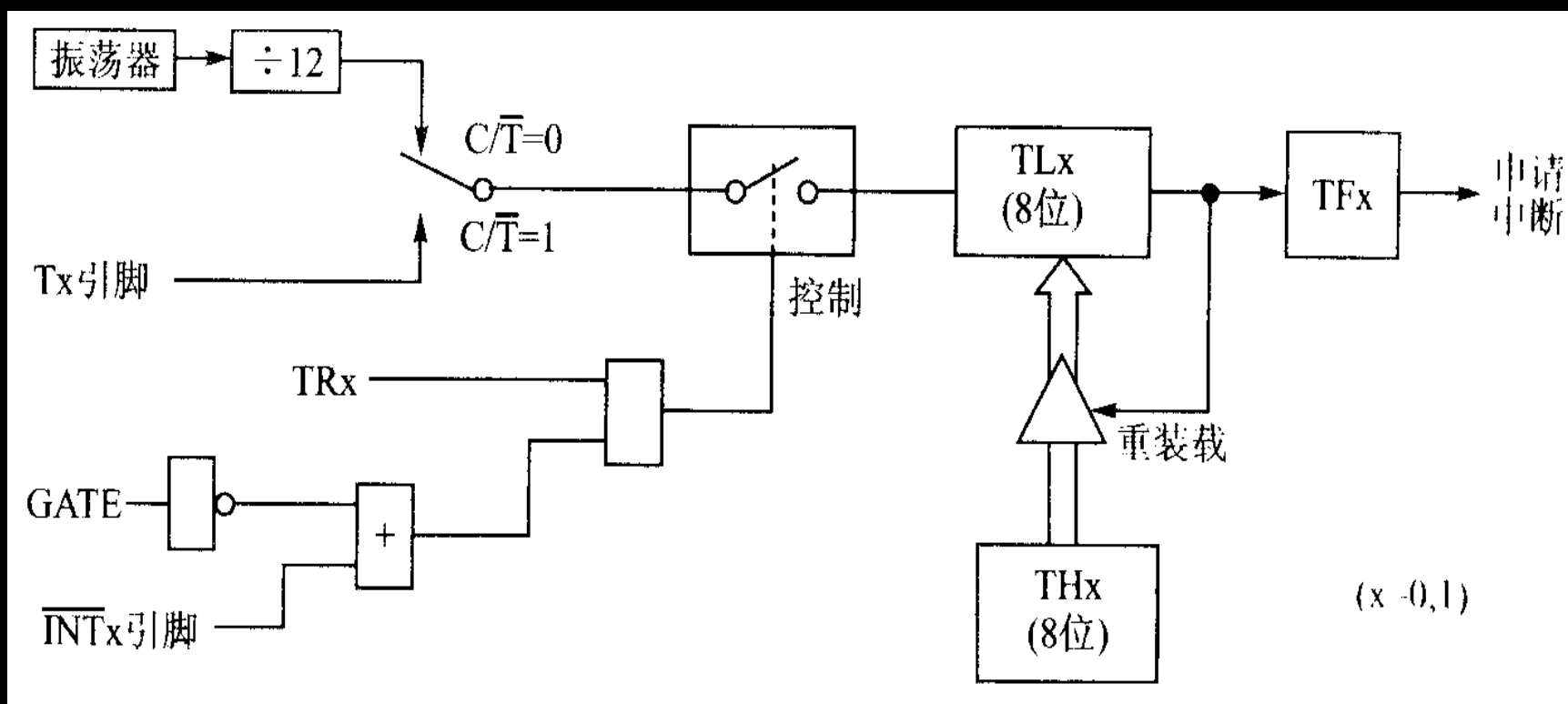
计数寄存器由**16**位组成，其余与方式0相同。



(3) 方式2

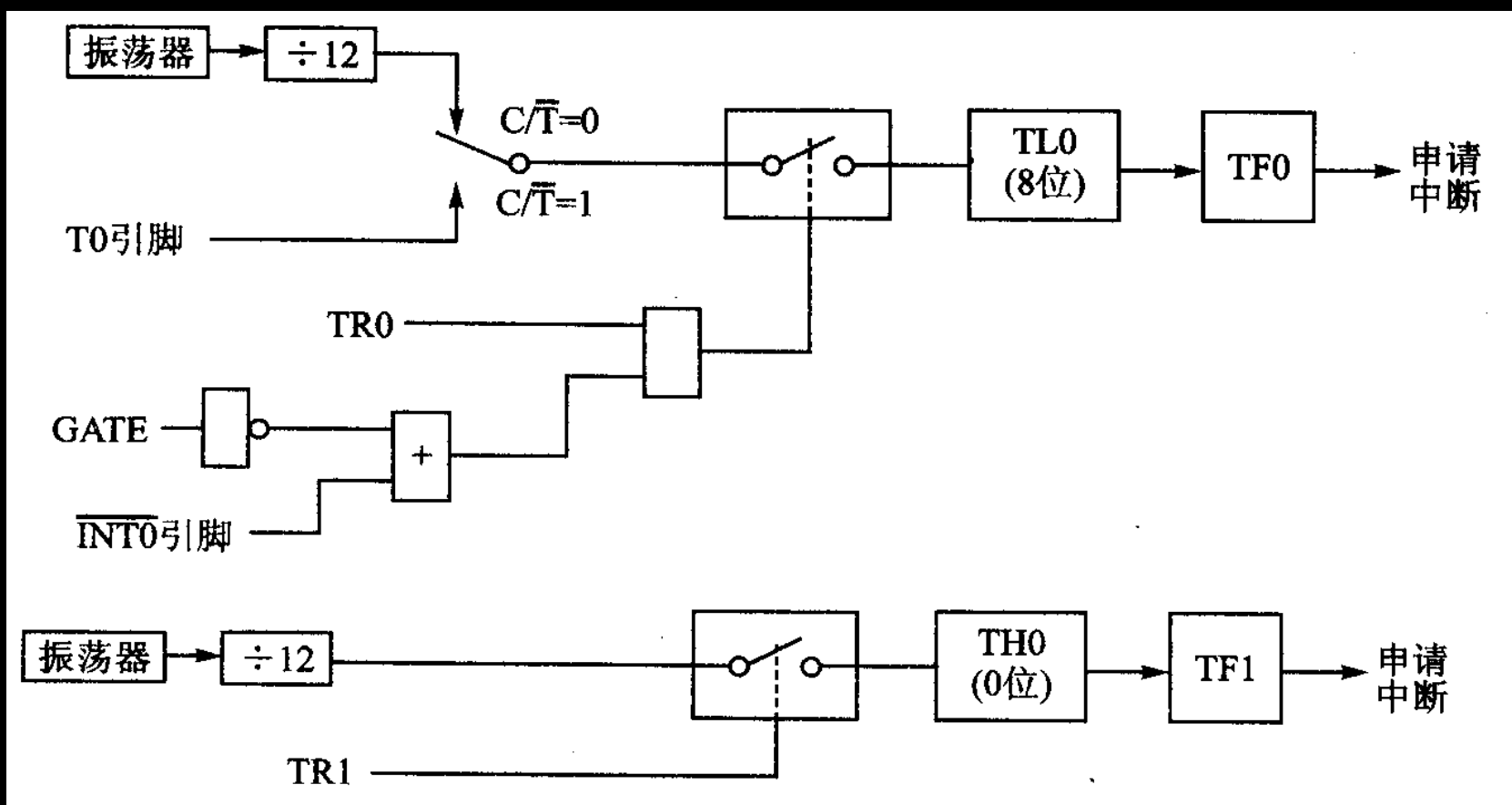
适合用于串行口波特率发生器

将16位计数寄存器分为两个8位寄存器，组成一个可重装入的8位计数寄存器。当TLx计数溢出时，一方面将TFx置位，另一方面将THx的内容重新装入TLx，继续计数。可重复装入。

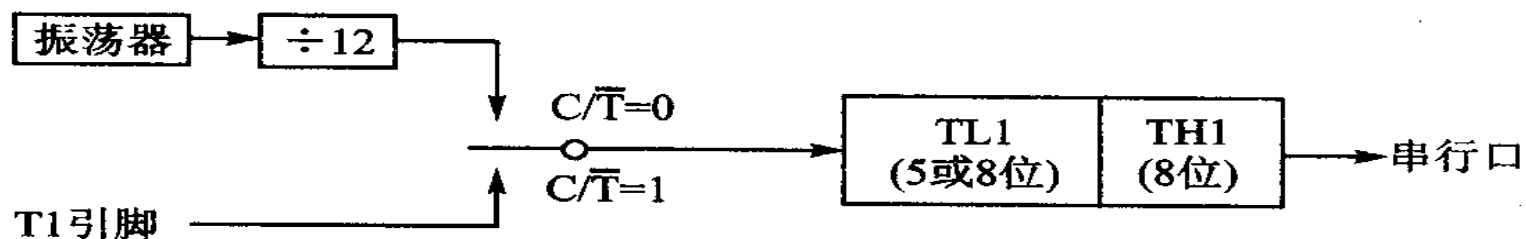


(4) 方式3

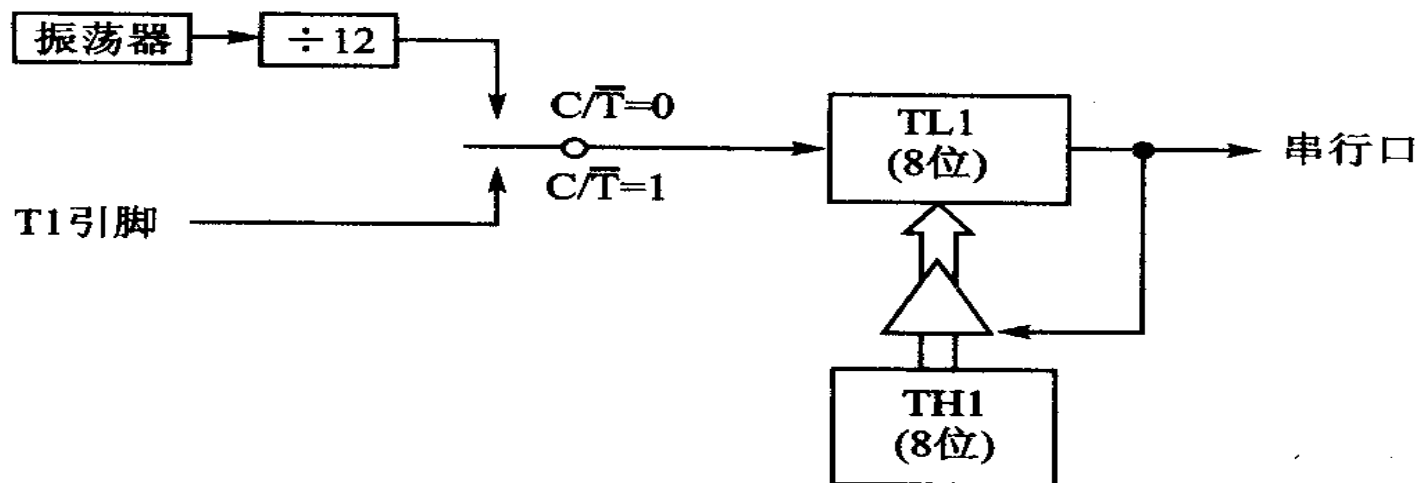
将T0分为一个8位定时器/计数器（TL0），一个8位定时器（TH0）。



当T0工作在方式3下，T1只能工作在方式0、1、2下，且不能用中断。



(a) T1的方式1或0



(b) T1的方式2

定时时间公式（T为定时时间）：

方式0： $T = (2^{13} - \text{初值}) * \text{时钟周期} * 12$

方式1： $T = (2^{16} - \text{初值}) * \text{时钟周期} * 12$

方式2： $T = (2^8 - \text{初值}) * \text{时钟周期} * 12$

二、 定时器/计数器的编程和使用

1. 写TMOD；只能用字节寻址

设置工作方式、功能选择、门控信号等

2. 将初值（时间常数）写入THi和TLi。只能字节寻址；

3. 启动定时或计数；

SETB TRi ; 启动计数器

SETB TCON. 4 (T0)

SETB TCON. 6 (T1)

CLR TRi ; 停止计数器

4. 定时器中断开放或禁止，即写IE。

(IE. 7 (EA) , IE. 3 (ET1) , IE. 1 (ET0))

SETB ETi

SETB EA

CLR ETi

CLR EA

四、定时器 / 计数器的应用举例

例1. 使用定时器 / 计数器T1的方式0。在P1.1引脚上产生周期为1ms的方波。晶振的频率为fosc=6MHz。

①定时常数计算

振荡器的频率fosc=6MHZ

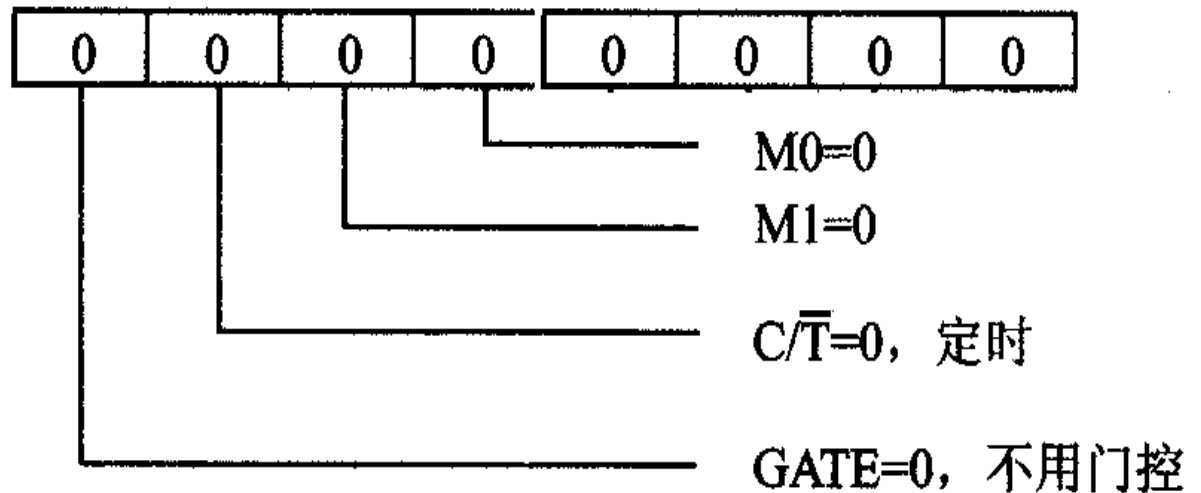
方式0计数器长度L=13

定时时间T=500μs

$T = (2^{13} - T0\text{初值}) * \text{时钟周期} * 12$

T0=7942D=1111100000110B

11111000 00000110B=F806H



控制字为 #00H

```
ORG 2000H
MOV TMOD,#00H
MOV TH1,#0F8H
MOV TL1,#06H
SETB TR1
LP1: JBC TF1, LP2
      AJMP LP1
```

```
LP2: MOV TL1, #06H
      MOV TH1, #0F8H
      CPL P1.1
      AJMP LP1
```


§ 5.3 串行接口

一、串行口结构与工作原理

在80C51中有一个串行接口（serial port），全双工的UART。80C51中的串行口是一个全双工的异步串行通信接口，它可以作为UART（通用异步接收和发送器）用，也可做同步移位寄存器用。

所谓**全双工**的异步串行通信接口，是说该接口可以**同时进行接收和发送数据**。

在8051中，口内的接收缓冲器和发送缓冲器在物理层上是隔离的，即是完全独立的。

可以通过访问特殊功能寄存器**SBUF**，来访问接收缓冲器和发送缓冲器。

接收缓冲器还具有双缓冲的功能，即它在接收第一个数据字节后，能接收第二个数据字节。但是，在他完成接受第二个数据字节后，若第一个字节仍未取走，那么该字节数据将丢失。

UART串行口的结构

UART串行口的结构如图所示。

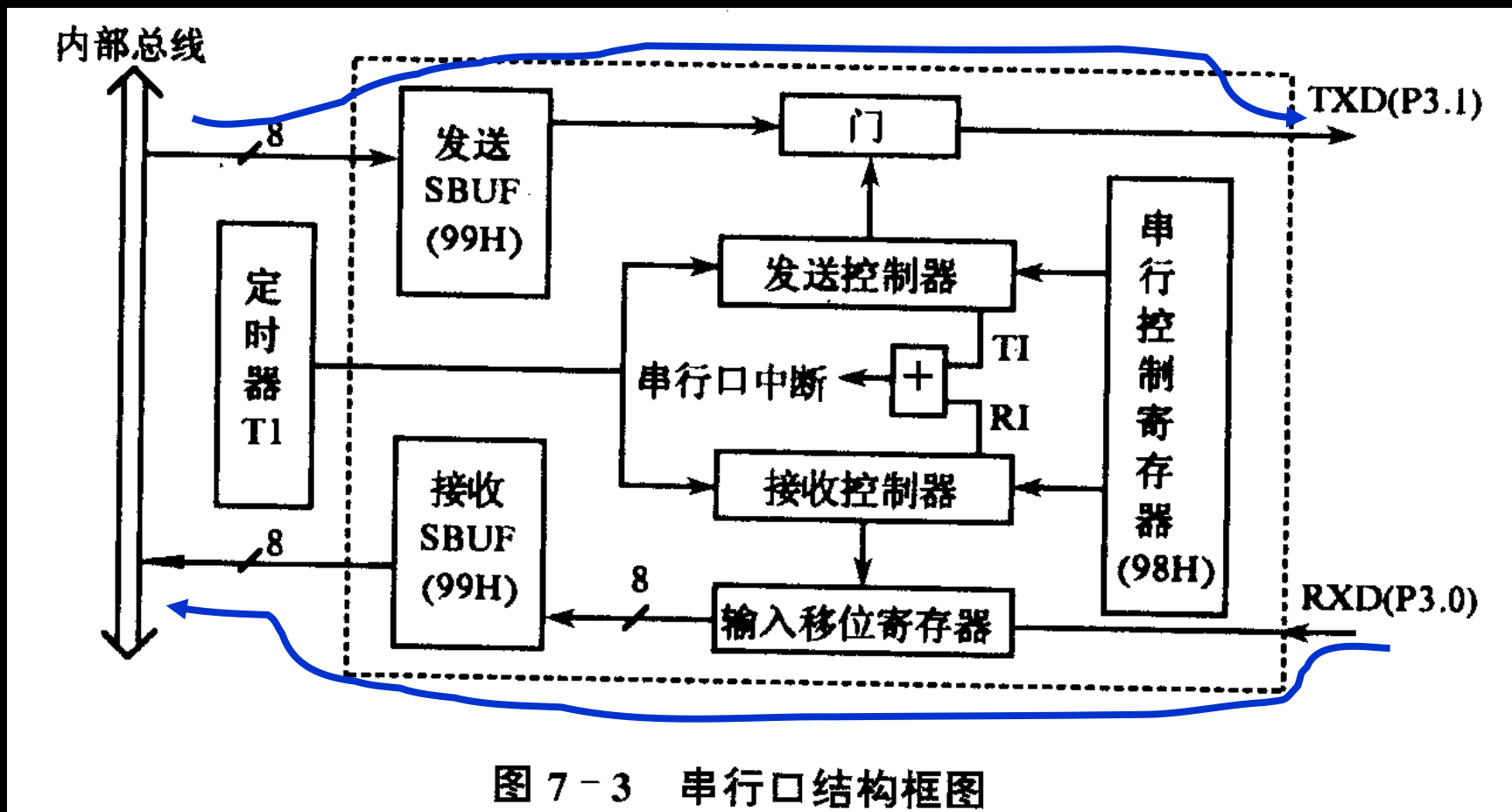


图 7-3 串行口结构框图

二、串行口的特殊功能寄存器

1、状态控制寄存器SCON

是一个逐位定义的8位寄存器，由它控制串行通信的方式选择、接收和发送、指示串行口的状态。

寄存器SCON既可字节寻址也可位寻址，字节地址为98H，位地址为98H-9FH。

位地址	9FH	9EH	9DH	9CH	9BH	9AH	99H	98H
位功能	SM0	SM1	SM2	REN	TB8	RB8	TI	RI

多机通信

0: 单机对单机

1: 多机通信

发送数据第9位

发送中断标志

SM0

SM1

SM2

REN

TB8

RB8

TI

RI

串口工
作方式

接收控制
0: 禁止接收
1: 允许接收

接收数据第9位

接收中断标志

2、控制寄存器PCON

是一个逐位定义的8位寄存器，目前仅有几位有定义，其中仅最高位—**SMOD**与串行口控制有关，其他位与掉电方式有关。

D7	D6	D5	D4	D3	D2	D1	D0
SMOD	-	-	-	GF1	GF0	PD	IDL

波特率选择位SMOD：

特殊功能寄存器**PCON**（电源控制寄存器）

中的**第7位**为波特率选择位**SMOD**。**SMOD=1**时，方式**1、2、3**的波特率加倍。复位时为**0**。

3、串行数据寄存器SBUF

包含在物理上是隔离的两个8位寄存器：发送数据和接收数据寄存器，共用一个地址99H。

D7	D6	D5	D4	D3	D2	D1	D0
SD7	SD6	SD5	SD4	SD3	SD2	SD1	SD0

写SBUF: MOV SBUF, A ; 发送
读SBUF: MOV A, SBUF ; 接收

三、串行口的工作方式及多机通信方式

1、方式0

当SM0=0、SM1=0时，串行口选择方式0。实质上是一种同步移位寄存器方式。

...	D0	D1	D2	D3	D4	D5	D6	D7	...
-----	----	----	----	----	----	----	----	----	-----

方式0的波特率为振荡频率的1/12。

2、方式1

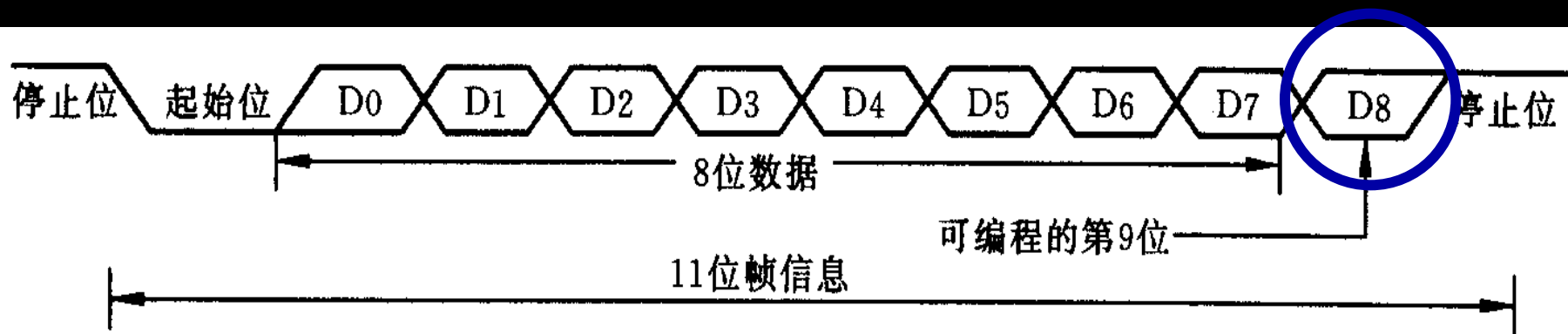
当SM0=0、SM1=1时，串行口选择方式1。

发送或者接收一帧信息为10位，格式如下：



3、方式2（SM0=1、SM1=0）

每帧包括：1位起始位（0），8位数据位（低位在先），**1位可编程的第9位数据**，1位停止位（1）。



当接收时，只有接收到第9位为1时才将串行口的中断标志RI置“1”。

按方式2进行多机数据通讯时，不允许进行奇/偶校验，第9位数据可作为数据/地址标志位。

4、方式3

SM0=1、SM1=1时，串行口选择方式3。数据帧格式与方式2相同。

各种工作方式的波特率：

SM0、SM1：串行口工作方式选择位。

SM0	SM1	工作方式	功能	波特率
0	0	0	8 位的移位寄存器	$f_{osc}/12$
0	1	1	8 位的 UART	由定时器 T1 的溢出率确定
1	0	2	9 位的 UART	$f_{osc}/32$ 或 $f_{osc}/64$
1	1	3	9 位的 UART	由定时器 T1 的溢出率确定

(1) 方式0的波特率为振荡频率的1/12。

(2) 方式2的波特率：

SMOD=0：波特率为振荡频率的1/64（复位）

SMOD=1：波特率为振荡频率的1/32

$$\text{波特率} = 2^{\text{SMOD}} \times f_{\text{osc}} / 64$$

(3) 方式1、3的波特率：

由定时器1的溢出速率决定。

$$\text{波特率} = 2^{\text{SMOD}} \times (\text{定时器T1溢出率}) / 32$$

当用定时器1的方式2时，溢出周期为：

$$T = 12 / f_{osc} * (256 - X)$$

$$\text{波特率} = 2^{SMOD} * (\text{定时器T1溢出率}) / 32$$

$$\text{波特率} = 2^{SMOD} * f_{osc} / 12 / 32 / (256 - X)$$

表 7-1 定时器 T1 的常用波特率

波特率/(b/s)	$f_{\text{osc}}/\text{MHz}$	SMOD	定时器 1		
			C/T	模式	初值
方式 0: 1×10^6	12	×	×	×	×
方式 2: 375 000	12	1	×	×	×
方式 1, 3: 62 500	12	1	0	2	FFH
19 200	11.059	1	0	2	FDH
9 600	11.059	0	0	2	FDH
4 800	11.059	0	0	2	FAH
2 400	11.059	0	0	2	F4H
1 200	11.059	0	0	2	E8H

方式0的应用

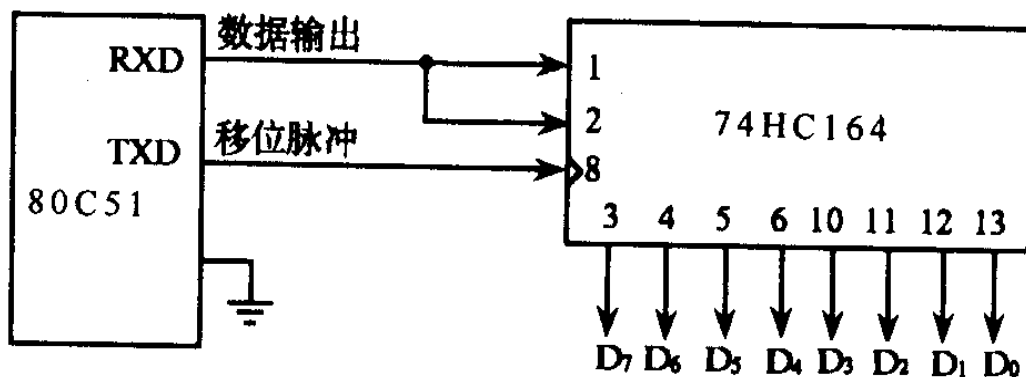


图 7-4 方式 0 用于 I/O 扩展输出

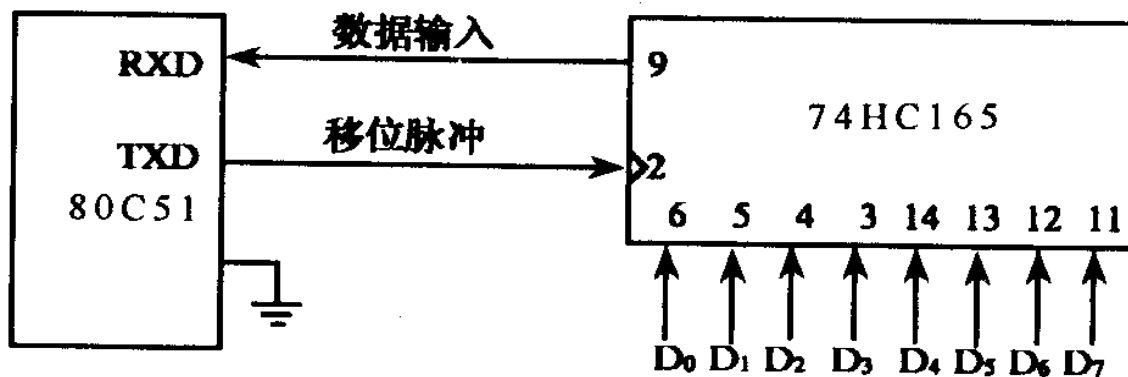
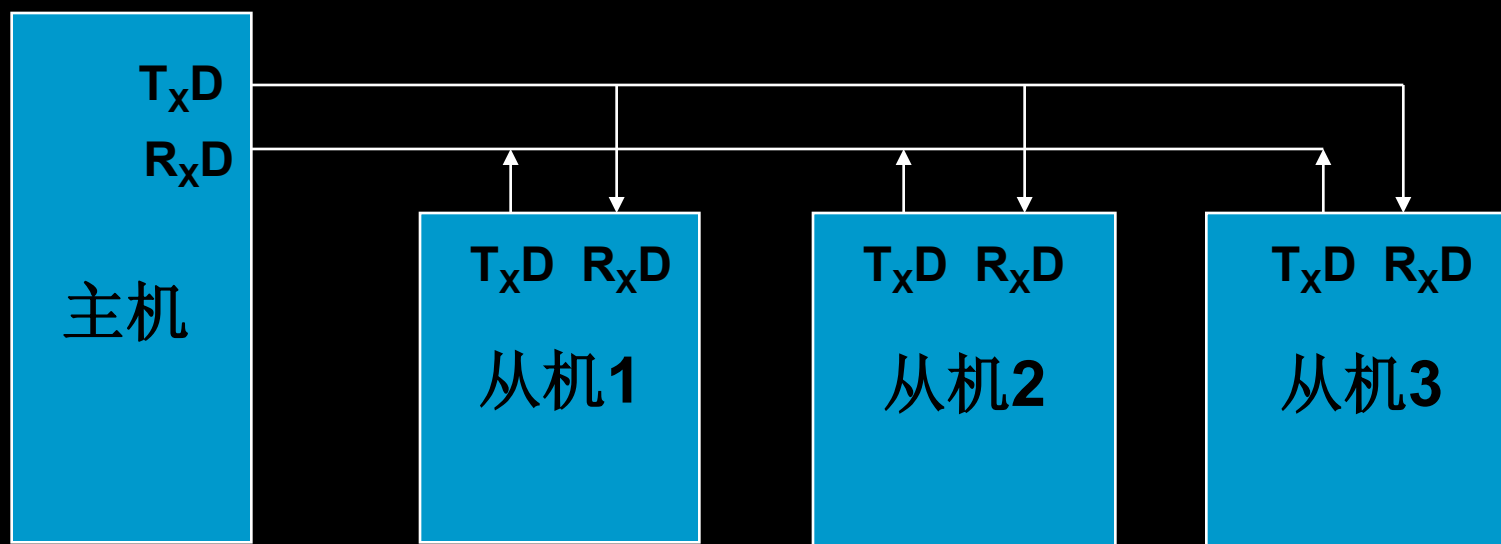


图 7-5 方式 0 用于 I/O 扩展输入

多处理机通信方式

在串行口控制寄存器SCON中，设有多处理机通信位SM2。当串行口以方式2或3接收时，若SM2=1，如果接收到的第九位数据为1，才将数据送入接收缓冲器SBUF，并RI置1发中断，否则数据丢失；SM2=0，则无论第九位是1还是0，都能将数据送入SBUF，并引发中断。利用这一特性，变可以实现多机之间的通信。



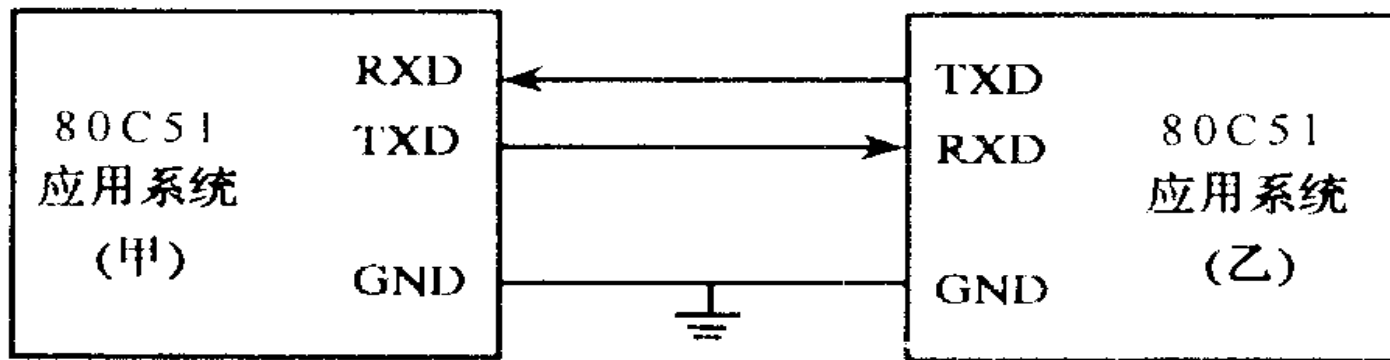
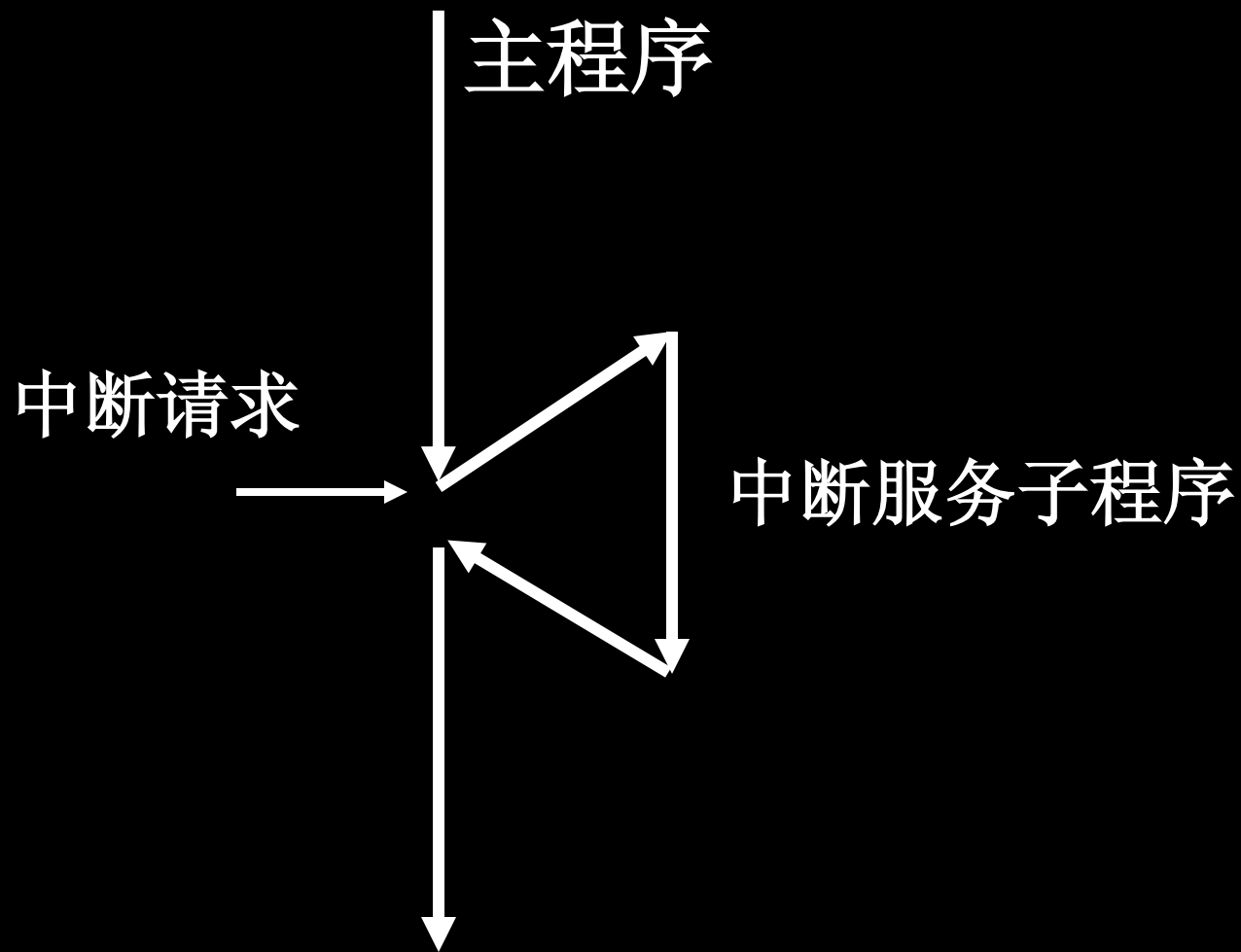


图 7-9 双机异步通信连接图

例3：编程把甲机片内RAM 50H -- 5FH单元中的数据块从串行口输出。用方式3发送，TB8作奇偶校验位。采用定时器1方式2作波特率发生器，波特率为1200波特，**fosc** = 11.0592MHz，预置值TH1=0E8H。

编程使乙机从甲机接收16个字节数据块，并存入片外3000H -- 300FH单元。接收过程中要求判奇偶校验标志TB8。若出错则置F0标志为1，若正确则置F0标志为0，然后返回。

§ 5.4 中断系统



一、中断请求源

中断源：能发出中断请求信号的来源称为中断源。

8051系列单片机系列有**5个中断源**。

- 1、外部中断请求源INT0
- 2、外部中断请求源INT1
- 3、内部定时器请求源T0
- 4、内部定时器请求源T1
- 5、片内串口请求源

二、中断标志

(1) 定时器控制寄存器TCON

TCON寄存器前4位与定时器有关，后4位与外中断有关。

- **TCON**可以字节寻址，也可以位寻址。而位寻址使用更灵活。

外部中断两种触发方式：

1、电平触发方式：低电平有效

2、跳变触发方式：负边沿触发



TCON

0: 硬件复位
1: INT0上有中断

0: INT0电平触发
1: INT0负边沿触发

与T0、T1有关



0: 硬件复位
1: INT1上有中断

0: INT1电平触发
1: INT1负边沿触发

(2) 串行口控制寄存器SCON

SCON

串行口接收标志位
0: 无接收中断
1: 有接收中断



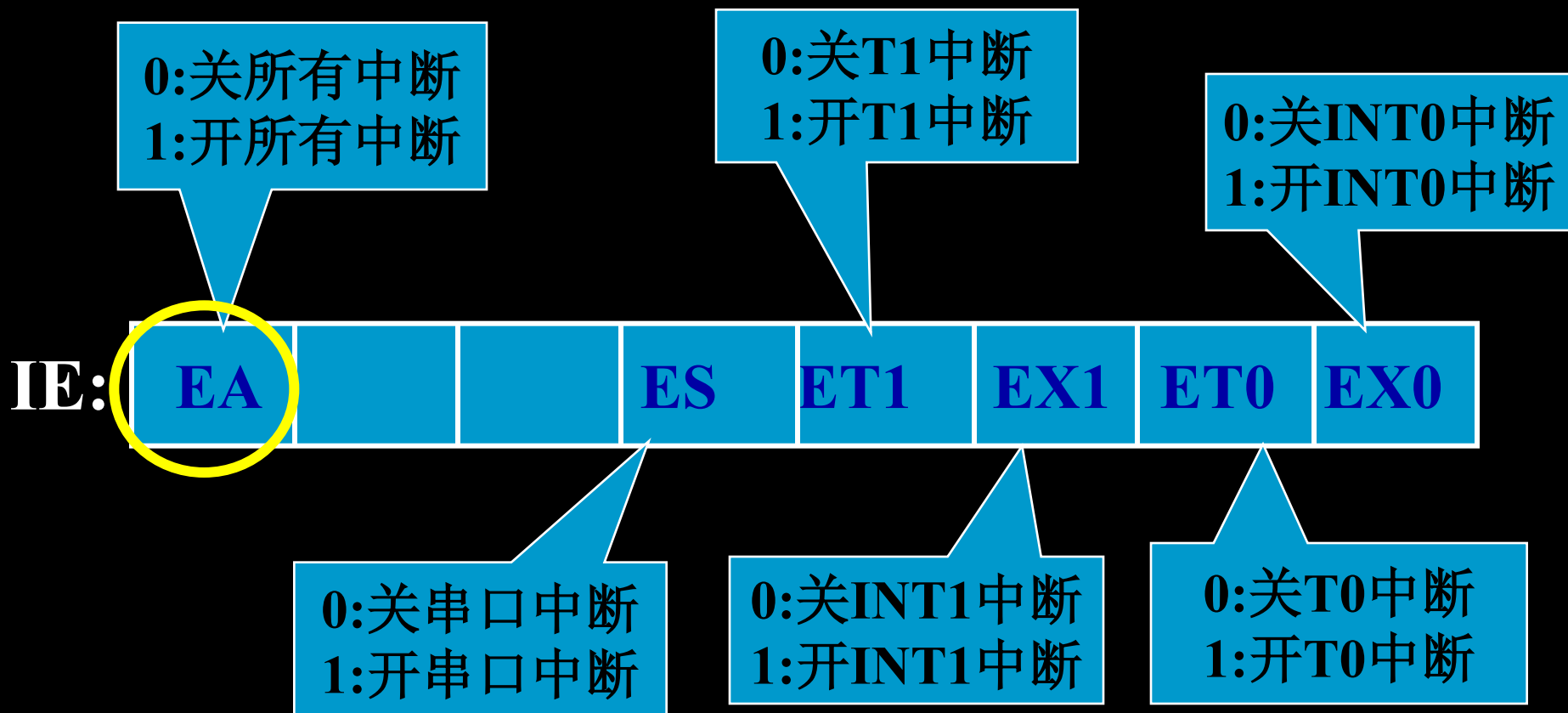
串行口发送标志位
0: 无发送中断
1: 有发送中断

- 当串口发送完一组数据，**TI**变成**1**（即有发送中断）请求**CPU**进行发送后的处理。
- 当串口接收完一组数据，**RI**变成**1**（即有接收中断）请求**CPU**进行接收后的处理。

TI、RI值需要软件清零。以便**CPU**处理下一个发送、接收中断。

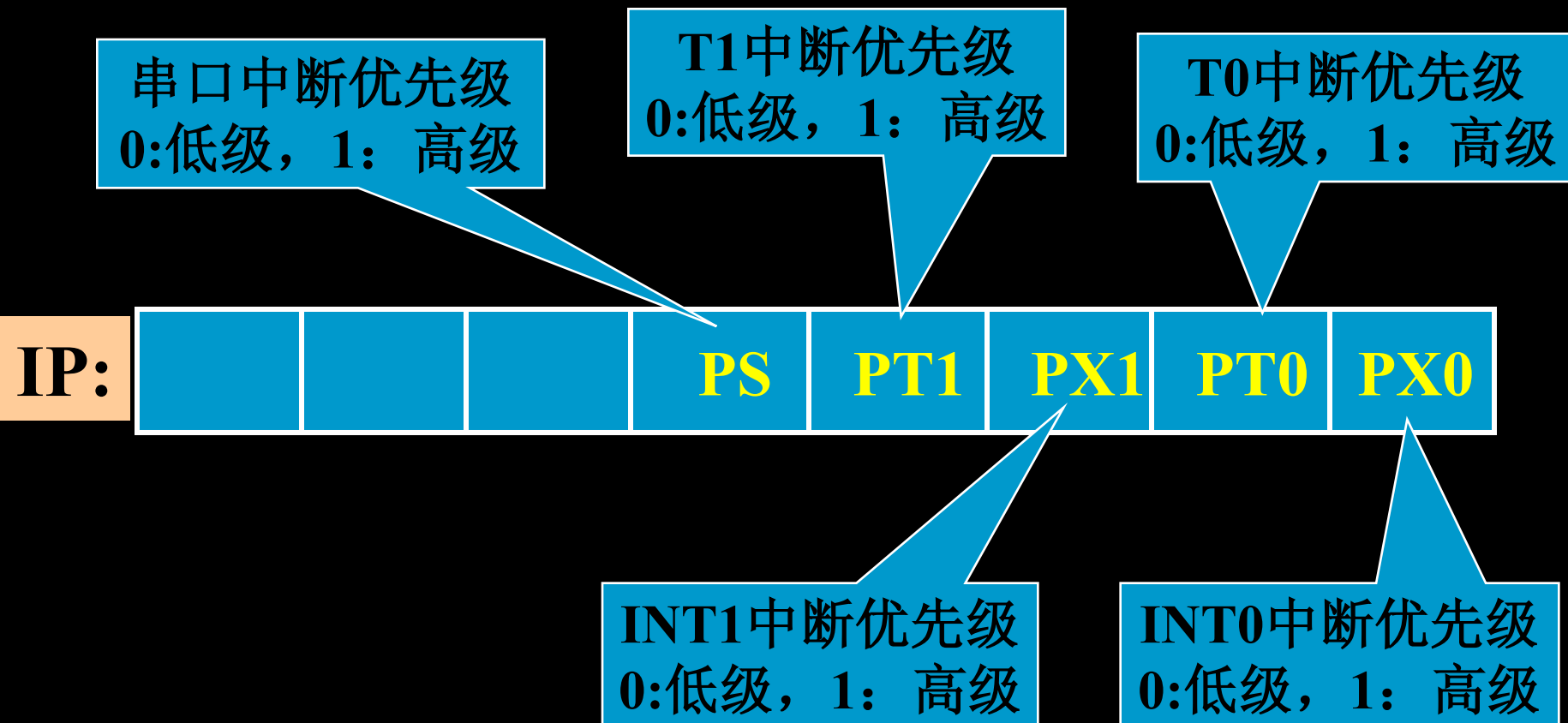
三、中断允许寄存器IE

该寄存器用来反映CPU对中断源的开放或屏蔽。
相当于一组开关。



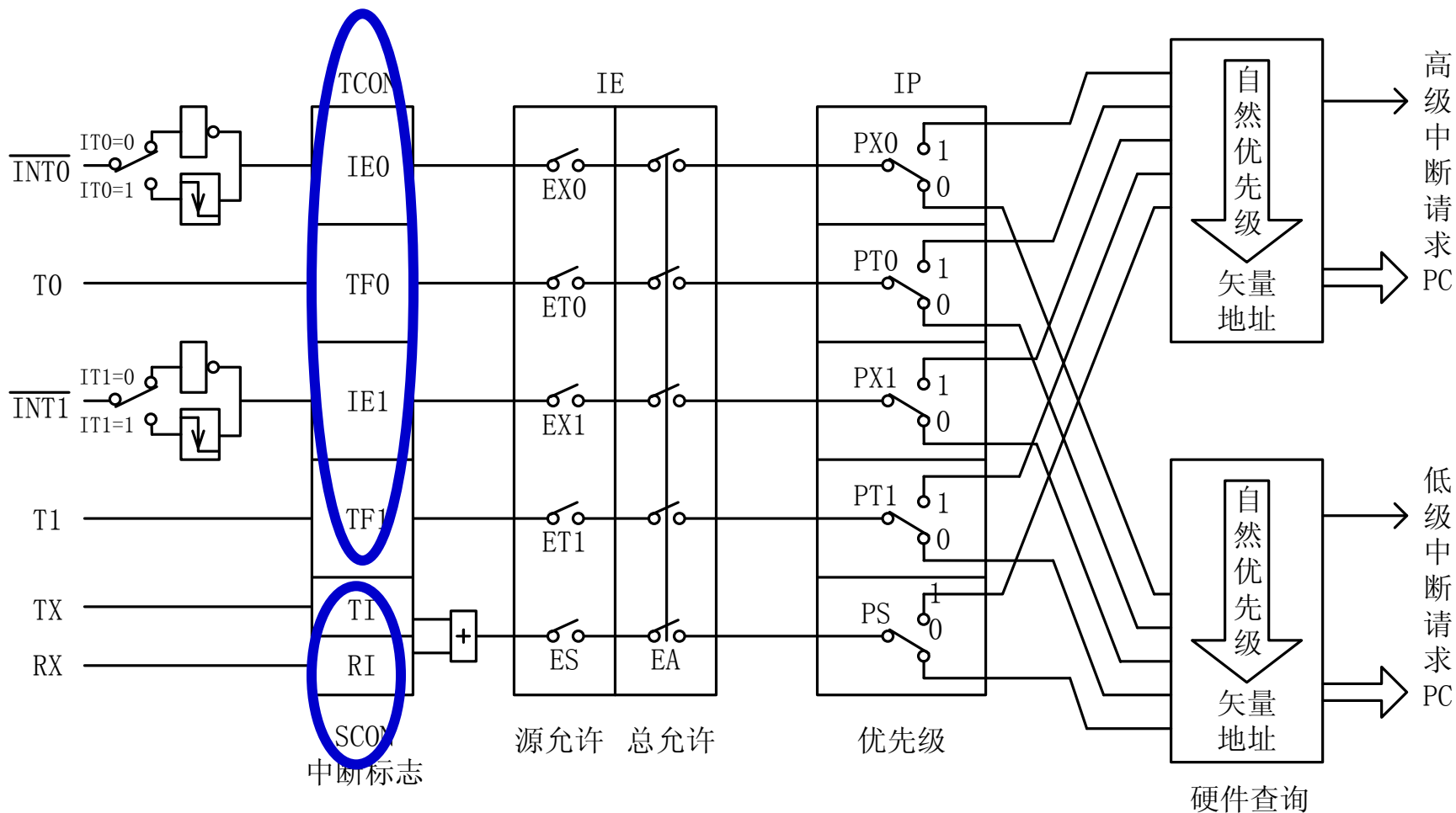
四、中断优先级寄存器IP

所有中断都可设定位高、低两个优先级，以便CPU对中断实现中断嵌套。



同一级别的优先顺序:

外部中断0	(IE0)	高	
T0溢出中断	(IF0)		
外部中断1	(IE1)		
T1溢出中断	(IF1)		
串口中断	(TI或RI)	低	



中断系统

五、 中断响应过程和中断中断矢量地址

1、CPU响应中断请求过程：

- (1)、保存中断点地址：把当前程序计数器PC内容压入堆栈。
- (2)、关闭中断：防止响应中断期间有其它中断干扰。
- (3)、根据中断源入口地址转入相应中断服务程序执行。

2、80C51中断入口地址

$\overline{\text{INT0}}$	0003H
T0	000BH
$\overline{\text{INT1}}$	0013H
T1	001BH
串行口中断	0023H

3、中断请求的撤除

在中断被响应前，中断源发出的中断请求是由**CPU**锁存在特殊功能寄存器中的**TCON**和**SCON**的相应标志位中的。一旦某个中断请求得到响应，**CPU**必须把它的中断标志位复位成“0”状态，称为中断请求的撤除。

系统自动
清除

(1)、定时器溢出中断的撤除

TF0和TF1是定时器溢出中断标志位，在中断得到响应后会自动复位成“0”状态

用户清除

(2)、串行口中断请求的撤除

TI和RI是串行口中断标志位，中断系统不能将它们复位。需要用户在中断服务程序的适当位置将它们复位。可用如下指令：

CLR TI 或 CLR RI

(3)、外部中断请求的撤除

IE0和IE1是外部中断的标志位。

(1) 当外部中断是负边沿触发时，由CPU自动复位。

(2) 当外部中断是低电平触发时，采用硬件复位。

例：

在下图中P1.4~P1.7接有四个发光二极管，P1.0~P1.3接有四个开关，消抖电路用于产生中断请求信号。当消抖电路的开关来回拨动一次将产生一个下降沿信号，通过/INT0向CPU申请中断。要求：初时发光二极管全黑，每中断一次，P1.0~P1.3所接的开关状态反映到发光二极管上，且要求开关断开的对应发光二极管亮。

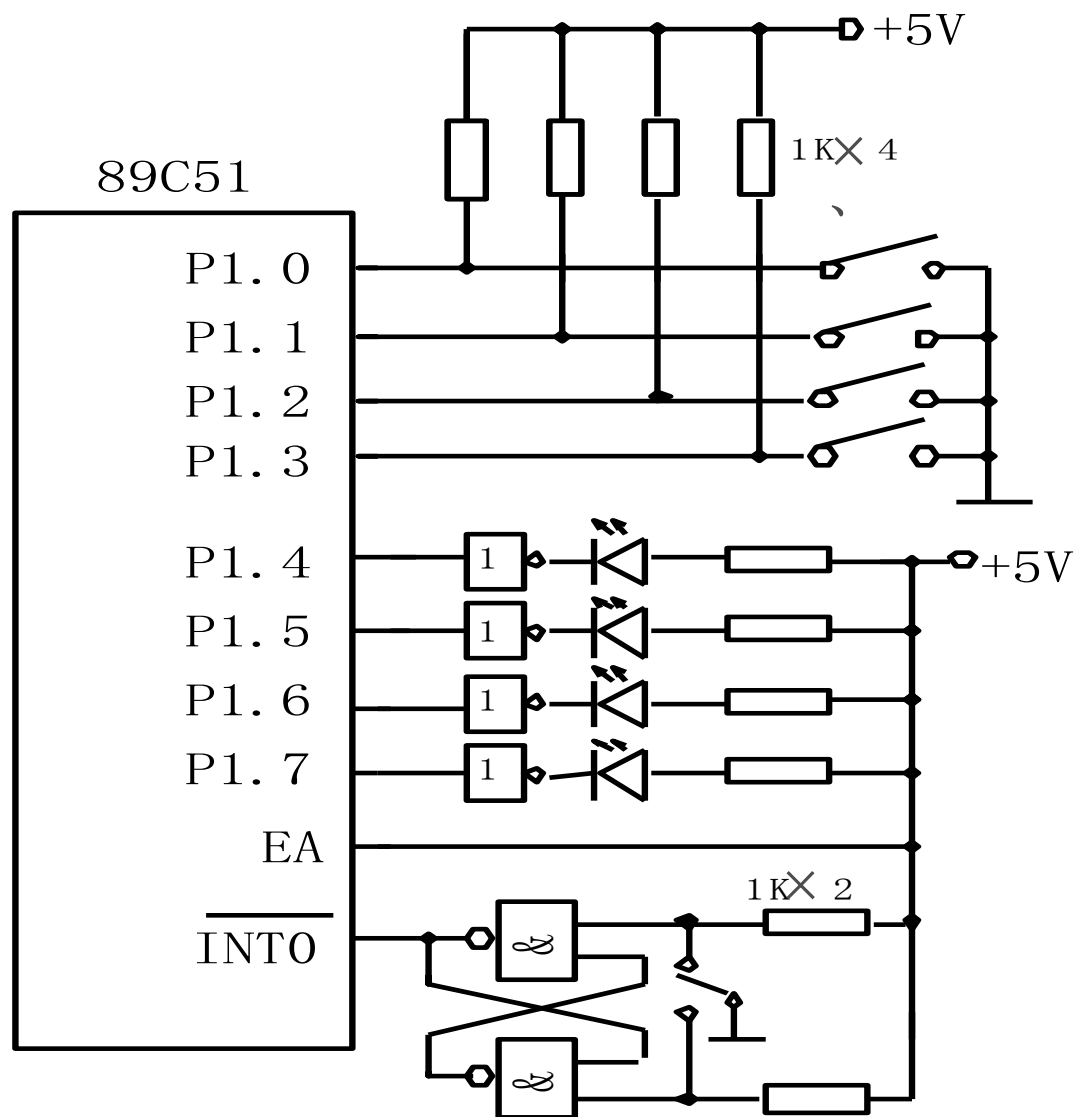


图6. 3

```

ORG    0000H
AJMP   MAIN
ORG    0003H           ; INT0中断入口
AJMP   WBI             ; 转中断服务程序
ORG    0030H           ; 主程序
MAIN:  MOV    P1, #0FH  ; 全灯灭，低四位输入
      SETB   IT0       ; 边沿触发中断
      SETB   EX0       ; 允许外中断0中断
      SETB   EA        ; 开中断开关
      SJMP   $
WBI:   MOV    P1, #0FH  ; P1先写入“1”且灯灭
      MOV    A, P1      ; 输入开关状态
      SWAP   A
      MOV    P1, A      ; 输出到P1高4位
      RETI
      END

```

在Keil中测量程序执行时间:

```
ORG    0000H
LJMP    MAIN
MAIN:
    MOV    P0,#0FEH
    MOV    R7,#0FEH
MAIN_LP:
    LCALL    DELAY
    MOV    A,R7
    RL     A
    MOV    R7,A
    MOV    P0,A
    JMP    MAIN_LP
```

;-----

```
DELAY:
    MOV    R0,#0FFH
    MOV    R1,#0FFH
DLY_LP:
    NOP
    NOP
    DJNZ    R0,DLY_LP
    MOV    R0,#0FFH
    DJNZ    R1,DLY_LP
    RET
END
```

首先设置晶振:

Options for Target 'Target 1'

Device Target Output Listing C51 A51 BL51 Locate BL51 Misc Debug Utilities

Atmel AT89C51

Xtal (MHz): 11.0592 ☒ Use On-chip ROM (0x0-0xFFFF)

Memory Model: Small: variables in DATA

Code Rom Size: Large: 64K program

Operating: None

Off-chip Code memory

	Start:	Size:
Eprom		
Eprom		
Eprom		

Off-chip Xdata memory

	Start:	Size:
Ram		
Ram		
Ram		

☐ Code Banking

Banks: 2 Bank Area: 0x0000 0xFFFF

Start: End:

☐ 'far' memory type support

☐ Save address extension SFR in interrupt

确定 取消 Defaults

设置两个断点，求出时间差即可。

Project Workspace

Register	Value
Regs	
r0	0x00
r1	0x00
r2	0x00
r3	0x00
r4	0x00
r5	0x00
r6	0x00
r7	0x00
Sys	
a	0x00
b	0x00
sp	0x07
sp_max	0x07
dptr	0x0000
PC \$	C:0x0000
sec	0.00000000
psw	0x00

02-LED流水灯

版本: 2.0
语言: 汇编
操作: 将JP1短接。
实验描述: 控制与P0口相连的8个LED指
网址: www.hkd168.com

```
ORG 0000H
LJMP MAIN

MAIN:
    MOV P0, #0FEH
    MOV R7, #0FEH
MAIN_LP:
    LCALL DELAY
    MOV A, R7
    RL A
    MOV R7, A
    MOV P0, A
    JMP MAIN_LP

DELAY:
    MOV R0, #0FFH
    MOV R1, #0FFH
DLY_LP:
    NOP
    NOP
    DJNZ R0, DLY_LP
    MOV R0, #0FFH
    DJNZ R1, DLY_LP
    RET
END
```

Project Workspace

Register	Value
Regs	
r0	0x00
r1	0x00
r2	0x00
r3	0x00
r4	0x00
r5	0x00
r6	0x00
r7	0xfe
Sys	
a	0x00
b	0x00
sp	0x07
sp_max	0x07
dptr	0x0000
PC \$	C:0x0008
status	5
sec	0.00000543
...	...

02-LED流水灯

版本: 2.0
语言: 汇编
操作: 将JP1短接。
实验描述: 控制与P0口相连的8个LED指示灯
网址: www.hkd168.com

```
ORG 0000H
LJMP MAIN

MAIN:
MOV P0, #0FEH
MOV R7, #0FEH
MAIN_LP:
LCALL DELAY
MOV A, R7
RL A
MOV R7, A
MOV P0, A
JMP MAIN_LP

DELAY:
MOV R0, #0FFH
MOV R1, #0FFH
DLY_LP:
NOP
NOP
DJNZ R0, DLY_LP
MOV R0, #0FFH
DJNZ R1, DLY_LP
RET

END
```

Project Work **Halt**

Register	Value
Regs	
r0	0xff
r1	0x00
r2	0x00
r3	0x00
r4	0x00
r5	0x00
r6	0x00
r7	0xfe
Sys	
a	0x00
b	0x00
sp	0x07
sp_max	0x09
dptr	0x0000
PC \$	C:0x000B
sec	0.28306858
psw	0x00

02-LED流水灯

版本: 2.0
语言: 汇编
操作: 将JP1短接。
实验描述: 控制与P0口相连的8个LED
网址: www.hkd168.com

```
ORG 0000H
LJMP MAIN

MAIN:
MOV P0, #0FEH
MOV R7, #0FEH
MAIN_LP:
LCALL DELAY
MOV A, R7
RL A
MOV R7, A
MOV P0, A
JMP MAIN_LP

DELAY:
MOV R0, #0FFH
MOV R1, #0FFH
DLY_LP:
NOP
NOP
DJNZ R0, DLY_LP
MOV R0, #0FFH
DJNZ R1, DLY_LP
RET

END
```


单片机作业二：

- 1、将51学习板例程中的1-6 的C语言程序改为汇编语言程序，并能够正确运行。