

微机原理与接口技术

第八章 定时计数控制接口

第8章 定时计数控制接口

- **教学重点**

- 8253的引脚和6种工作方式
- 8253的编程
- 8253在IBM PC系列机上的应用

定时器和计数器

定时控制在微机系统中极为重要

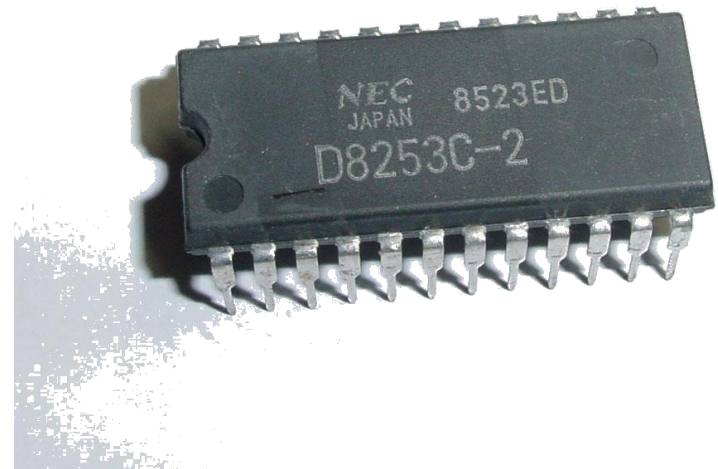
- 定时器由数字电路中的计数电路构成，通过记录高精度晶振脉冲信号的个数，输出准确的时间间隔
- 计数电路如果记录外设提供的具有一定随机性的脉冲信号时，它主要反映脉冲的个数（进而获知外设的某种状态），常又称为计数器

定时功能的实现方法

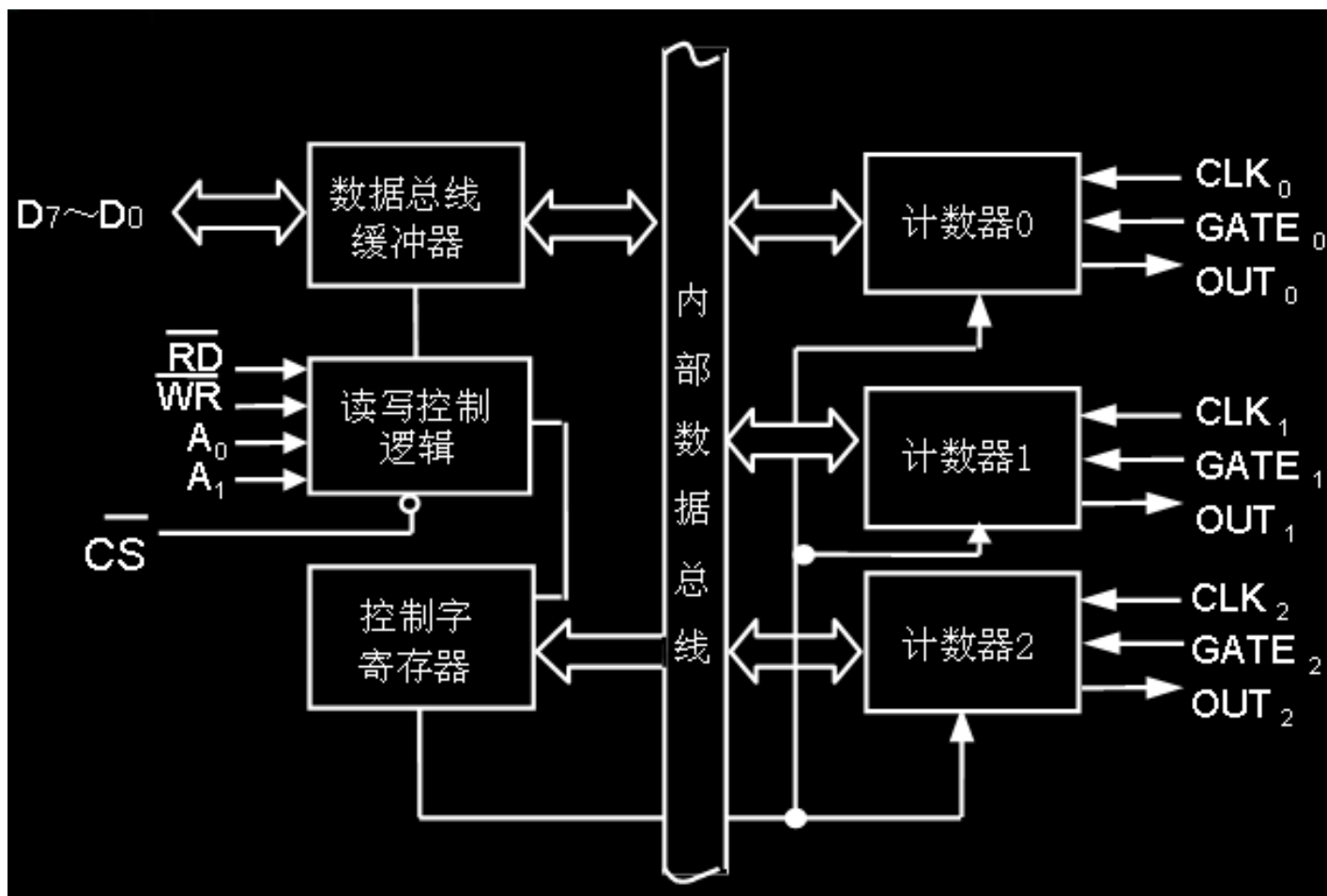
- 软件延时——利用微处理器执行一个延时程序段实现
- 不可编程的硬件定时——采用分频器、单稳电路或简易定时电路控制定时时间
- 可编程的硬件定时——软件硬件相结合、用可编程定时器芯片构成一个方便灵活的定时电路

8.1 8253/8254定时计数器

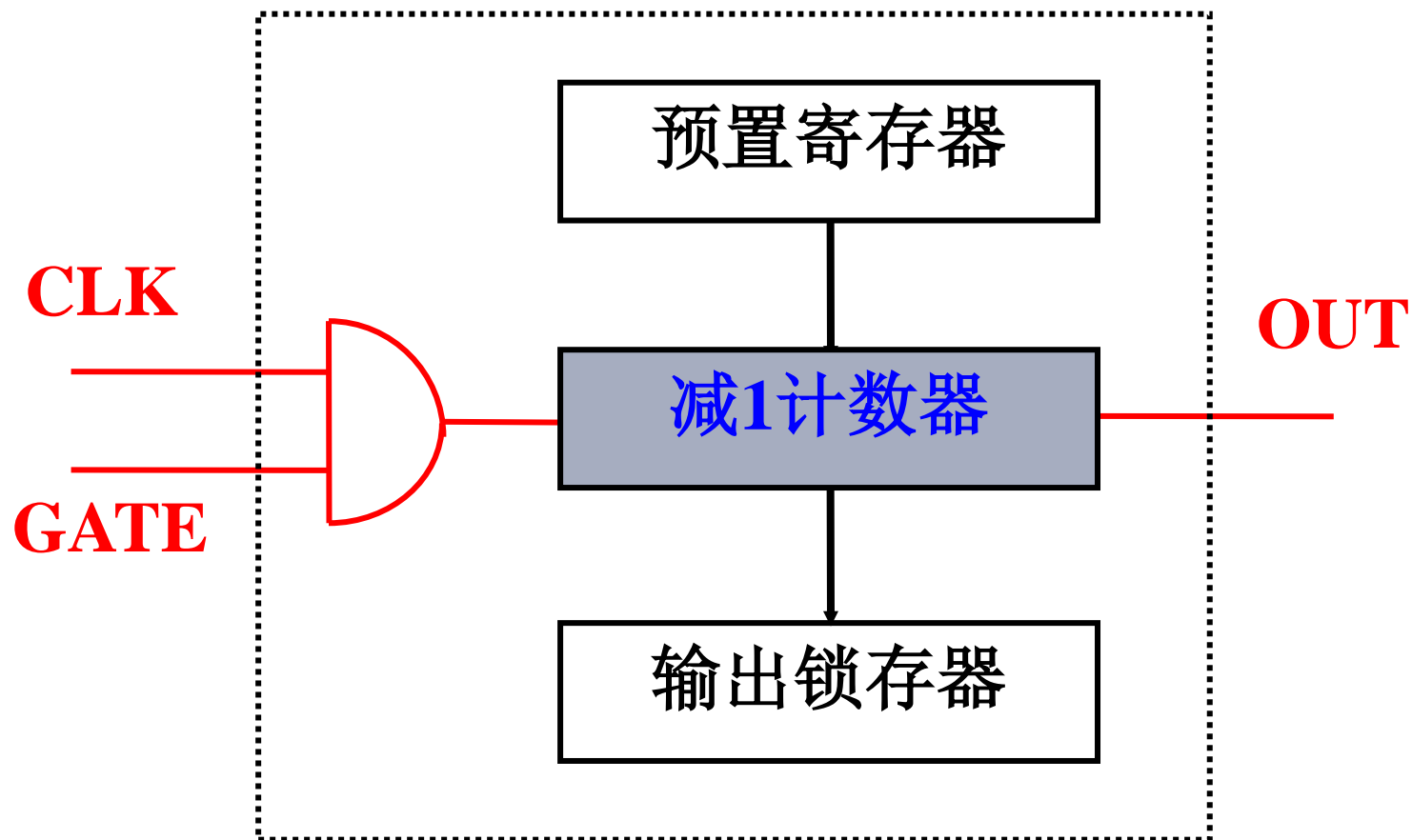
- 3个独立的16位计数器通道
- 每个计数器有6种工作方式
- 按二进制或十进制（BCD码）计数
- **8254是8253的改进型**



8.1.1 8253/8254的内部结构和引脚



计数器结构示意图



计数器结构分析

- 计数初值存于**预置寄存器**；
- 在计数过程中，**减法计数器**的值不断递减，而预置寄存器中的预置不变。
- **输出锁存器**用于写入锁存命令时，锁定当前计数值

计数器的3个引脚

- **CLK**时钟输入信号——在计数过程中，此引脚上每输入一个时钟信号（下降沿），计数器的计数值减1
- **GATE**门控输入信号——控制计数器工作，可分成电平控制和上升沿控制两种类型
- **OUT**计数器输出信号——当一次计数过程结束（计数值减为0），OUT引脚上将产生一个输出信号

2. 与处理器接口

- D0 ~ D7 数据线
 - RD* 读信号
 - CS* 片选信号
- A0 ~ A1 地址线
 - WR* 写信号

CS* A1 A0	I/O地址	读操作RD*	写操作WR*
0 0 0	40H	读计数器0	写计数器0
0 0 1	41H	读计数器1	写计数器1
0 1 0	42H	读计数器2	写计数器2
0 1 1	43H	无操作	写控制字

8253/8254的I/O地址

$\overline{\text{CS}}$	$\overline{\text{RD}}$	$\overline{\text{WR}}$	A1	A0	功 能
0	1	0	0	0	对计数器0设置计数初值
0	1	0	0	1	对计数器1设置计数初值
0	1	0	1	0	对计数器2设置计数初值
0	1	0	1	1	设置控制字
0	0	1	0	0	从计数器0读出计数值
0	0	1	0	1	从计数器1读出计数值
0	0	1	1	0	从计数器2读出计数值

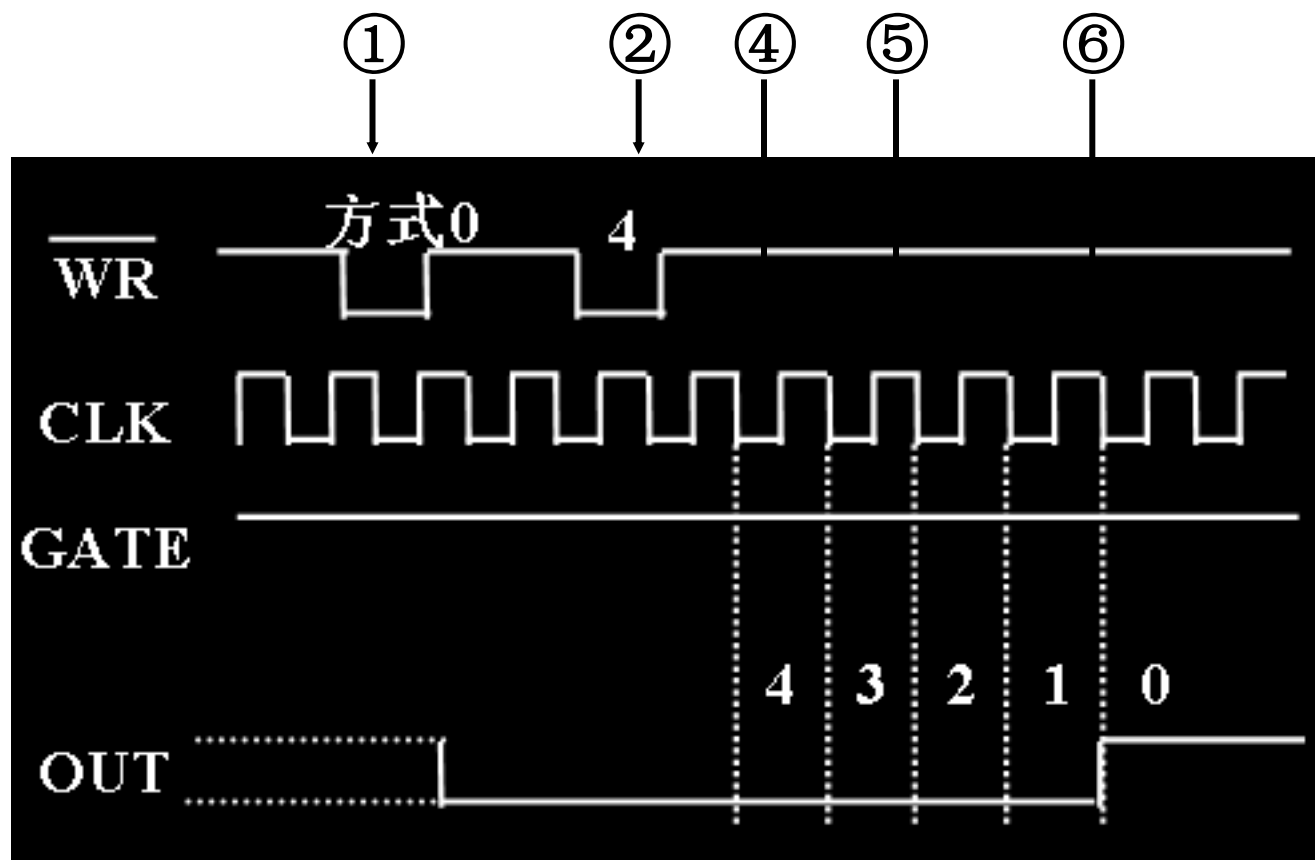
8.1.2 8253/8254的工作方式

- 8253有6种工作方式，由方式控制字确定
- 熟悉每种工作方式的特点才能根据实际应用问题，选择正确的工作方式

8.1.2 8253/8254的工作方式

- 每种工作方式的过程类似：
 - (1) 设定工作方式
 - (2) 设定计数初值
 - 〔 (3) 硬件启动 〕
 - (4) 计数初值进入减1计数器
 - (5) 每输入一个时钟（脉冲）使计数器减1的计数过程
 - (6) 计数过程结束

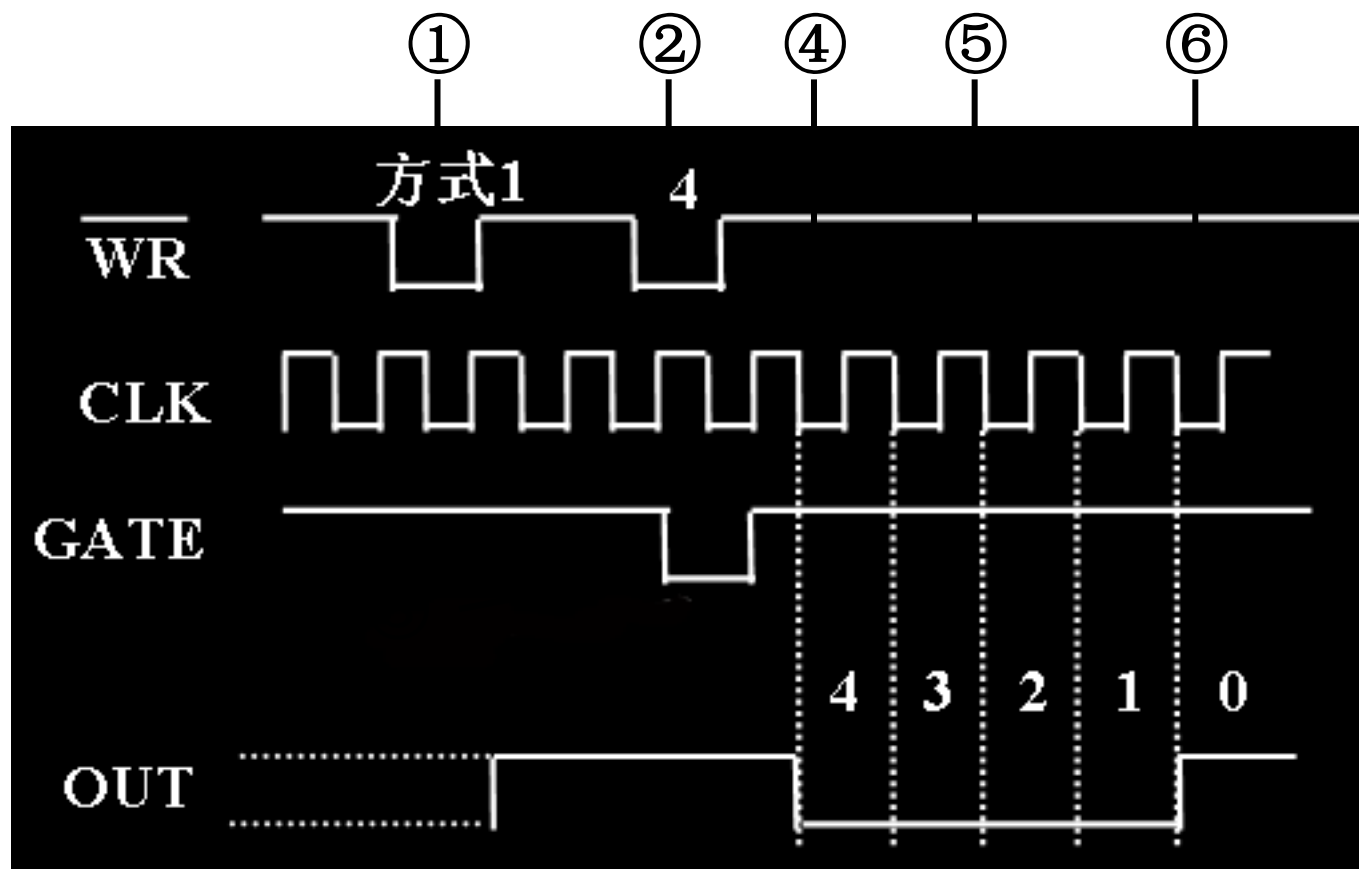
方式0 计数结束中断



⑥

计数结束

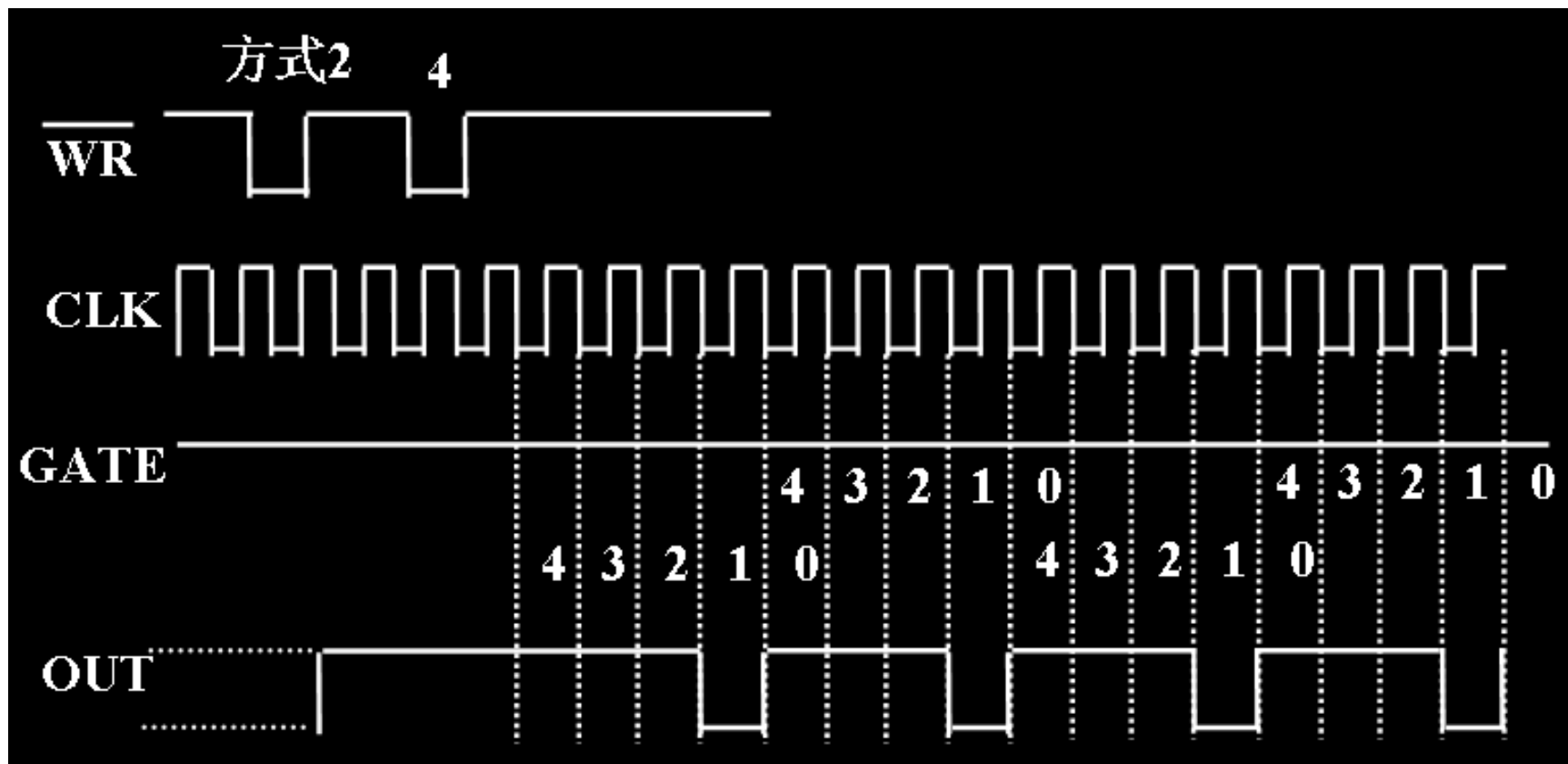
方式1 可编程单稳脉冲



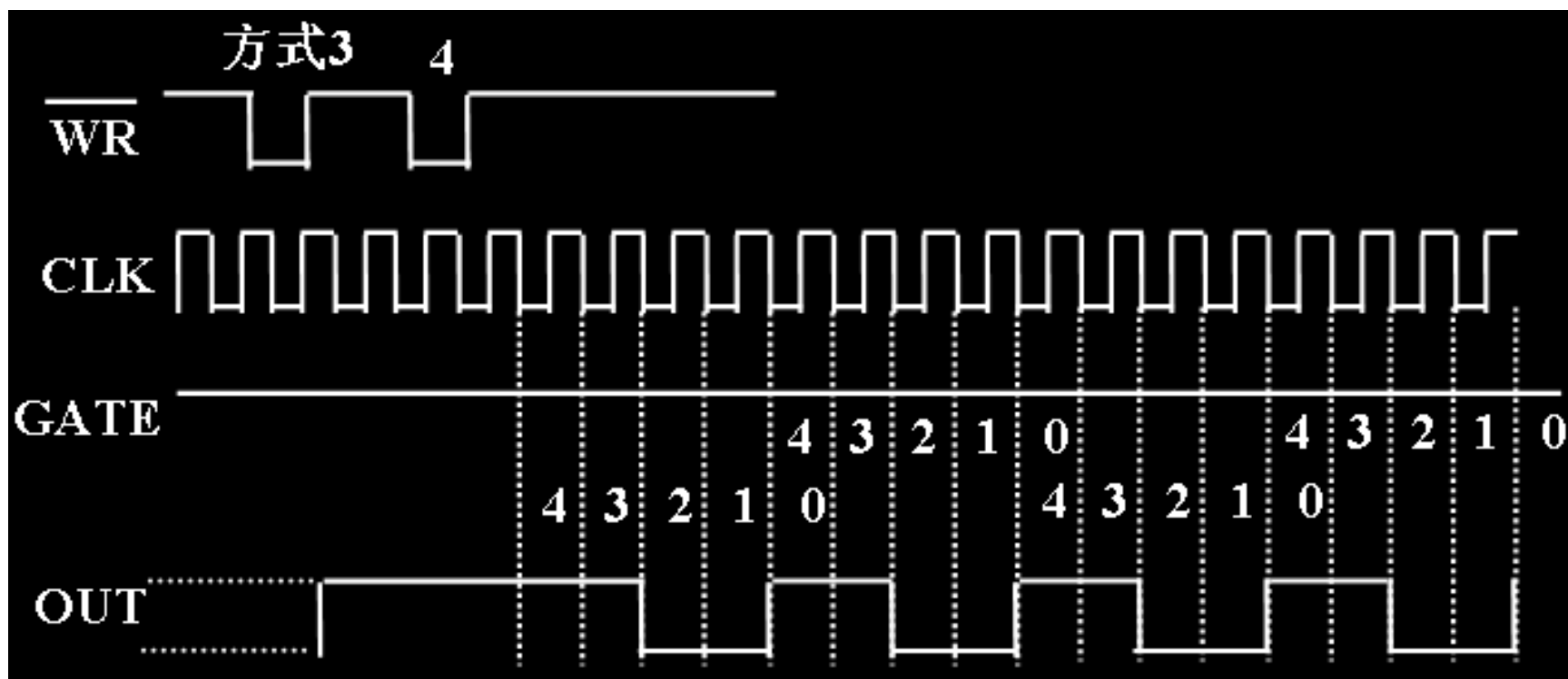
⑥

计数结束

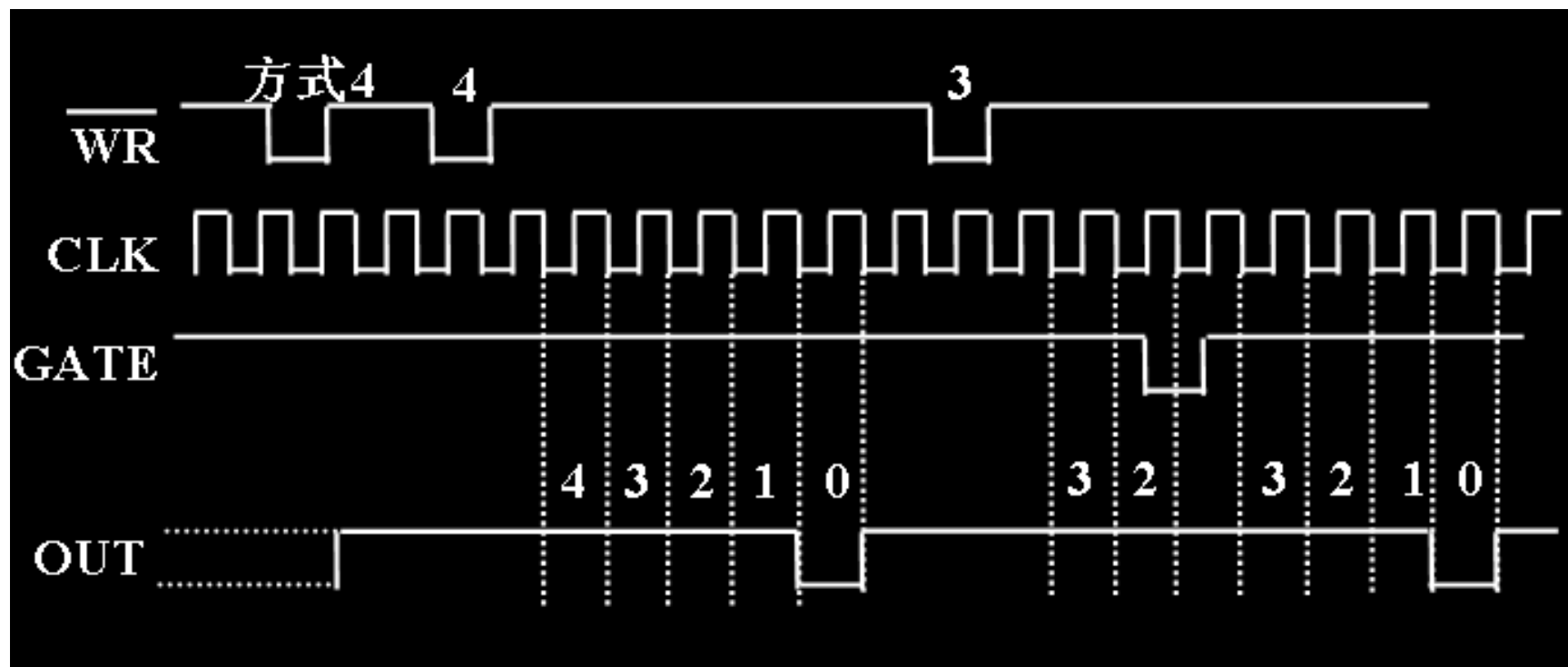
方式2 频率发生器（分频器）



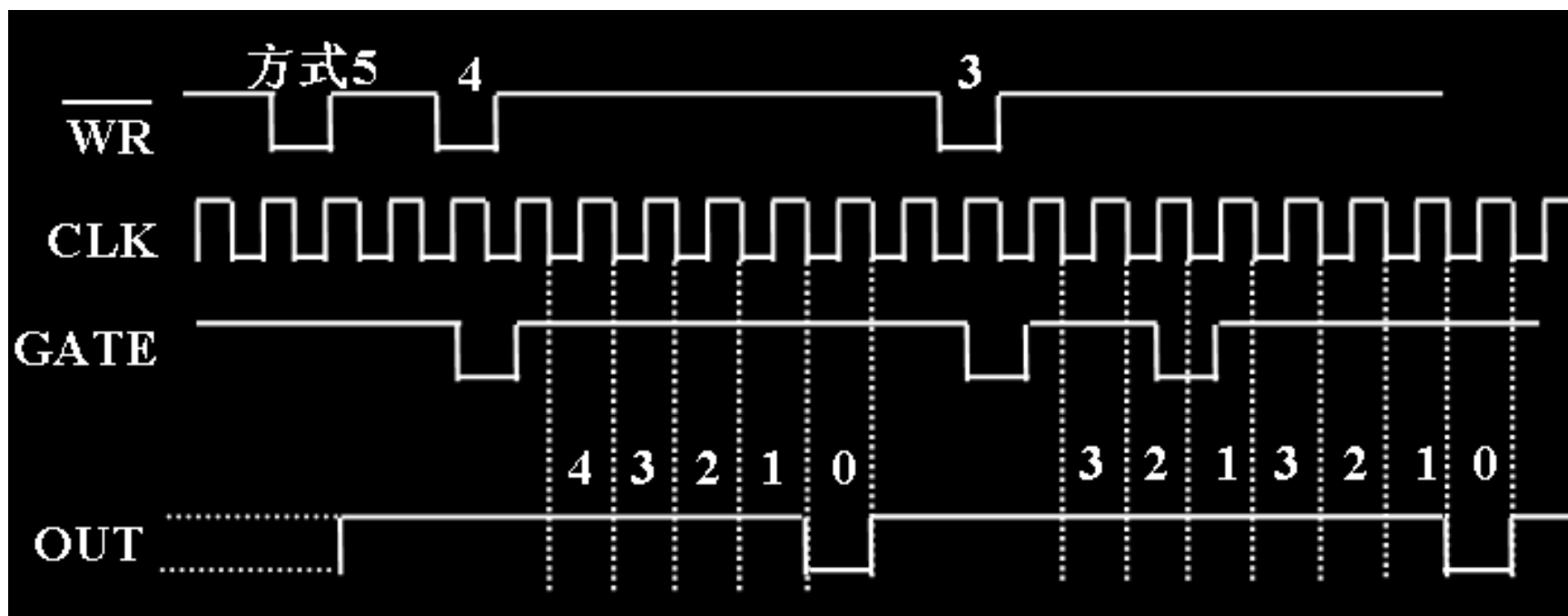
方式3 方波发生器



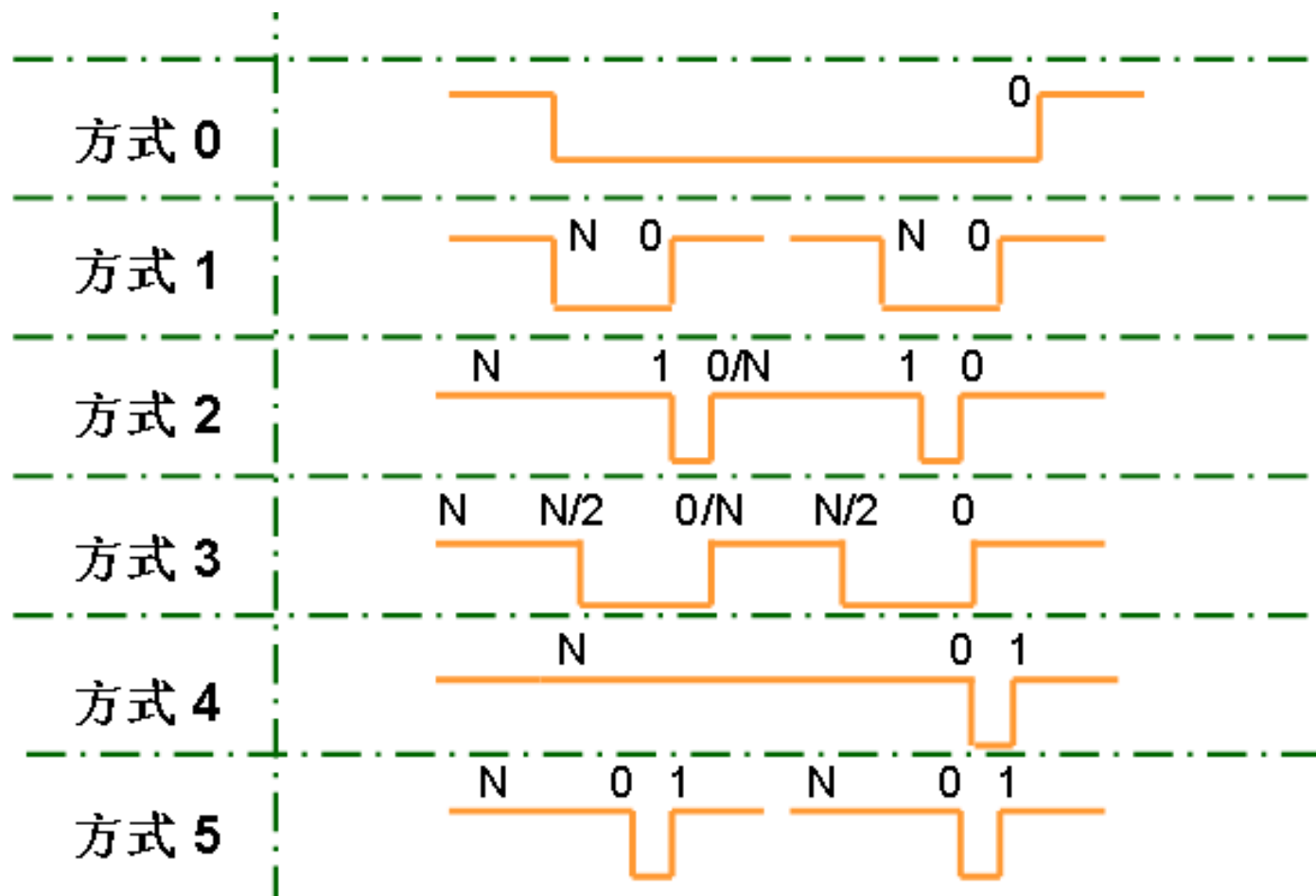
方式4 软件触发选通信号



方式5 硬件触发选通信号



各种工作方式的输出波形



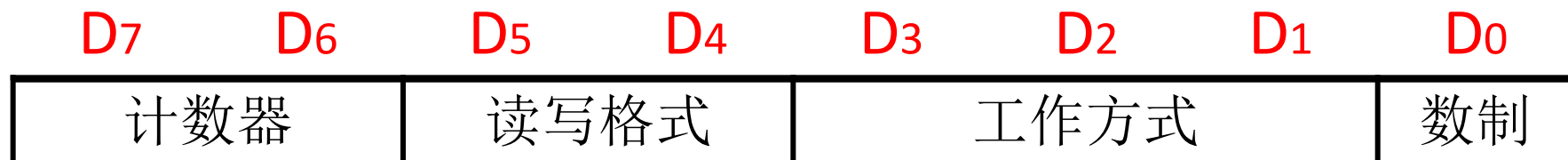
计数开始的时刻

- 需要注意：
- 处理器写入8253的计数初值只是写入了预置寄存器，之后到来的第一个CLK输入脉冲（需先由低电平变高，再由高变低）才将预置寄存器的初值送到减1计数器。
- 从第二个CLK信号的下降沿，计数器才真正开始减1计数。

8.1.3 8253/8254的编程

- 8253加电后的工作方式不确定
- 8253必须初始化编程，才能正常工作
- 写入控制字
 - 写入计数初值
 - 读取计数值
 - 8254新增读回命令

1 写入方式控制字



00 计数器

00 计数器锁存

000 方式0

0 二进制

控制字写入控制字I/O地址 ($A_1A_0=11$)

11 先读写后
后读写高

100 方式
101 方式

8253的控制字编程

- ； 某个8253的计数器0、1、2端口和控制端口地址依次是40H~43H
- ； 设置其中计数器0为方式0，采用二进制计数，先低后高写入计数值

mov al,30h

- ； 方式控制字：30H=00 11 000 0B

out 43h,al

- ； 写入控制端口：43H

2 写入计数值

- 选择二进制时
 - 计数值范围：0000H～FFFFH
 - 0000H是最大值，代表65536
- 选择十进制（BCD码）
 - 计数值范围：0000～9999
 - 0000代表最大值10000
- 计数值写入计数器各自的I/O地址

3 读取计数值

- 对8位数据线，读取16位计数值需分两次
- 计数在不断进行，应该将当前计数值先行锁存，然后读取：
 - 向控制字I/O地址：给8253写入锁存命令
 - 从计数器I/O地址：读取锁存的计数值
- 读取计数值，要注意读写格式和计数数制

8253的计数初值编程

- ； 某个8253的计数器0、1、2端口和控制端口地址依次是40H~43H
- ； 设置计数器0采用二进制计数，写入计数初值：1024（=400H）

`mov ax,1024` ； 计数初值： 1024（=400H）

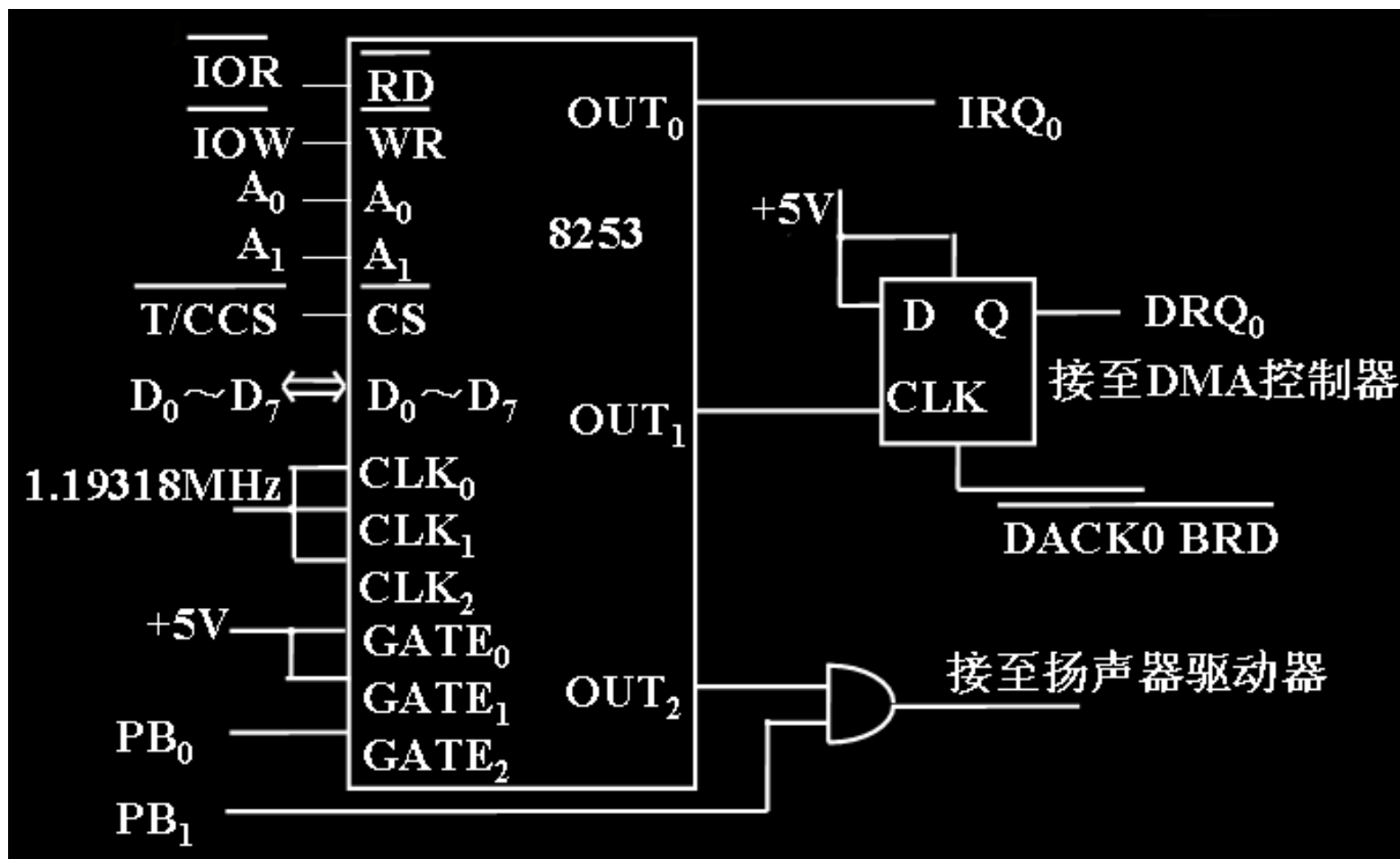
 ； 写入计数器0地址： 40H

`out 40h,al` ； 写入低字节计数初值

`mov al,ah`

`out 40h,al` ； 写入高字节计数初值

8.2 8253在IBM PC系列机上的应用



8.2.1 定时中断和定时刷新

- 从阅读初始化程序段
- 看计数器0作为定时中断的作用
- 将计数器1作为定时刷新
- 看如何编写初始化程序段

8253初始化

- **mov al,36h**
- ;计数器0为方式3，采用二进制计数，
- ;先低后高写入计数值
- **out 43h,al** ;写入方式控制字
- **mov al,0** ;计数值为0
- **out 40h,al** ;写入低字节计数值
- **out 40h,al** ;写入高字节计数值

计数器0：定时中断

- 计数器0：方式3，计数值：65536，输出频率为 $1.19318\text{MHz} \div 65536 = 18.206\text{Hz}$ 的方波
- 门控为常启状态，这个方波信号不断产生
- OUT0端接8259A的IRQ0，用作中断请求信号
- 每秒产生18.206次中断请求，或说每隔55ms（54.925493ms）申请一次中断
- DOS系统利用计数器0的这个特点，通过08号中断服务程序实现了日时钟计时功能

计数器1： 定时刷新

- 需要重复不断提出刷新请求
门控总为高，选择方式2或3
- 2ms内刷新128次，即15.6 μ s刷新一次
计数初值为18

8253初始化--定时刷新

- **mov al,54h**
- ;计数器1为方式2，采用二进制计数，只写低8位计数值
- **out 43h,al** ;写入方式控制字
- **mov al,18** ;计数初值为18
- **out 41h,al** ;写入计数值

8.2.2 扬声器控制

- 计数器2的输出控制扬声器的发声音调
- 计数器2只能工作在方式3，才能输出一定频率的方波，经滤波后得到近似的正弦波，进而推动扬声器发声
- 扬声器还受控于并行接口（8255芯片）
- 必须使PB0和PB1同时为高电平，扬声器才能发出预先设定频率的声音

扬声器控制--频率设置

- **speaker proc**
- **push ax**
- **mov al,0b6h**
- **out 43h,al ;写入控制字**
- **pop ax**
- **out 42h,al ;写入低8位计数值**
- **mov al,ah**
- **out 42h,al ;写入高8位计数值**
- **ret**
- **speaker endp**

扬声器控制--扬声器开

- **speakon proc**
- **push ax**
- **in al,61h**
- **or al,03h**
- **;D1D0=PB1PB0=11B, 其他位不变**
- **out 61h,al**
- **pop ax**
- **ret**
- **speakon endp**

扬声器控制--扬声器关

- **speakoff proc**
- **push ax**
- **in al,61h**
- **and al,0fch**
- **;D1D0=PB1PB0=00B, 其他位不变**
- **out 61h,al**
- **pop ax**
- **ret**
- **speakoff endp**

扬声器控制--主程序

- ;数据段
- **freq dw 1193180/600**
- ;代码段
- **mov ax,freq**
- **call speaker ;设置扬声器音调**
- **call speakon;打开扬声器声音**
- **mov ah,1 ;等待按键**
- **int 21h**
- **call speakoff ;关闭扬声器声音**

8.2.3 可编程硬件延时

- 利用日时钟每隔55ms中断一次不变的特点，可以编写一段不随系统时钟频率变化的固定延时程序
- 由于日时钟中断的时间单位是55ms，所以无法实现更短时间的延时
- 这时只有利用实时时钟中断，不过它的最短延时约是1ms（976 μ s）

可编程硬件延时--日时钟

- ;延时开始
- **mov ah,0**
- **int 1ah**
- **add dx,90** ;加5秒 ($5 \times 18 = 90$)
- **mov bx,dx** ;期望值送bx
- **repeat: int 1ah** ;再读日时钟
- **cmp bx,dx** ;与期望值比较
- **jne repeat** ;不等, 则循环
- **.....** ;相等, 延时结束

可编程硬件延时--实时时钟

- ;延时开始
- **mov cx,0**
- **mov dx,1952**
- ;延时 $1.952\text{ms} = 2 \times 976\mu\text{s}$
- **mov ah,86h**
- **int 15h**
- ;功能调用返回时，定时时间到

8.3 扩充定时计数器的应用

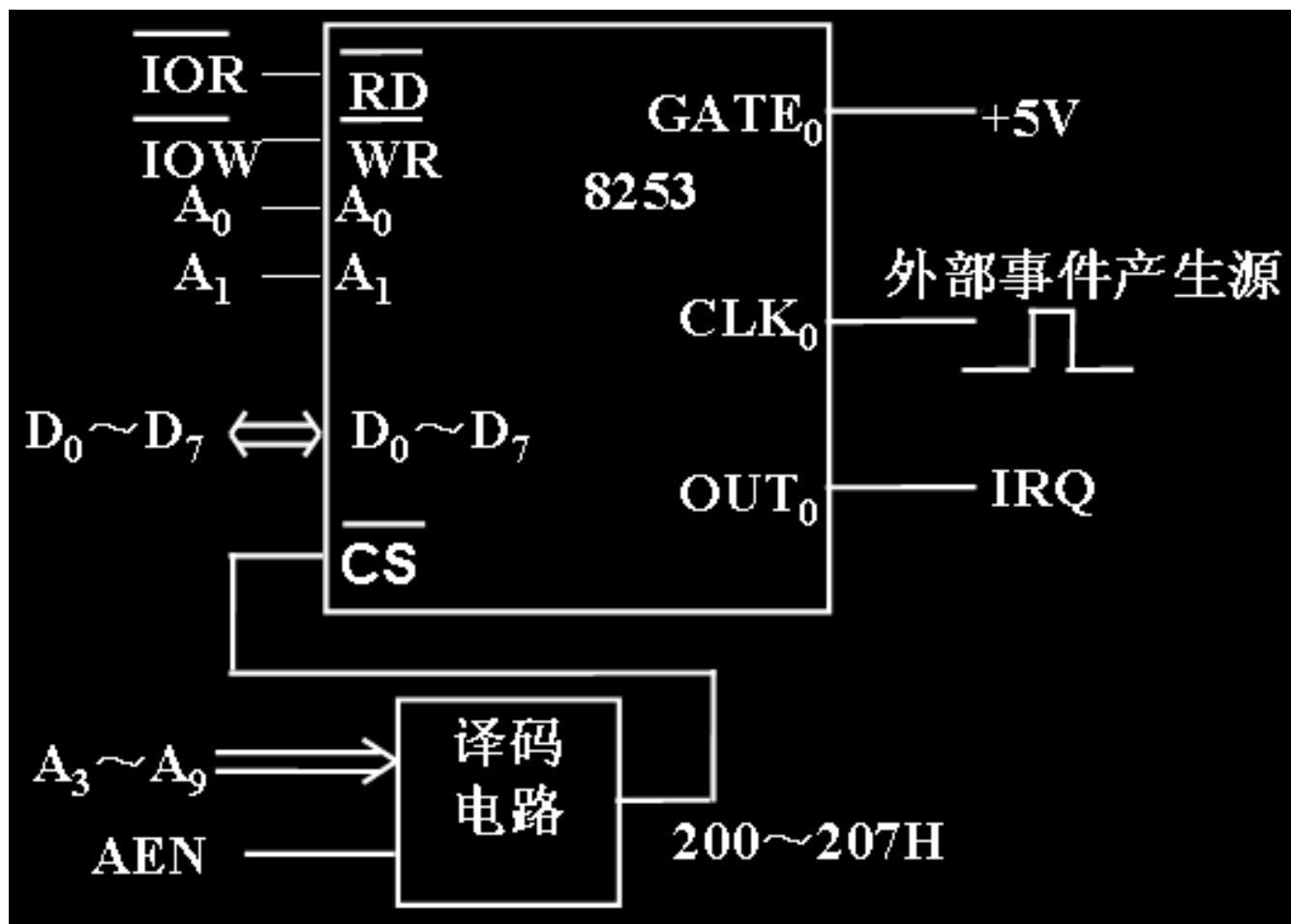
- 例题8.2

利用扩充定时计数器对外部事件的计数

- 例题8.3

为A/D转换电路提供可编程的采样信号

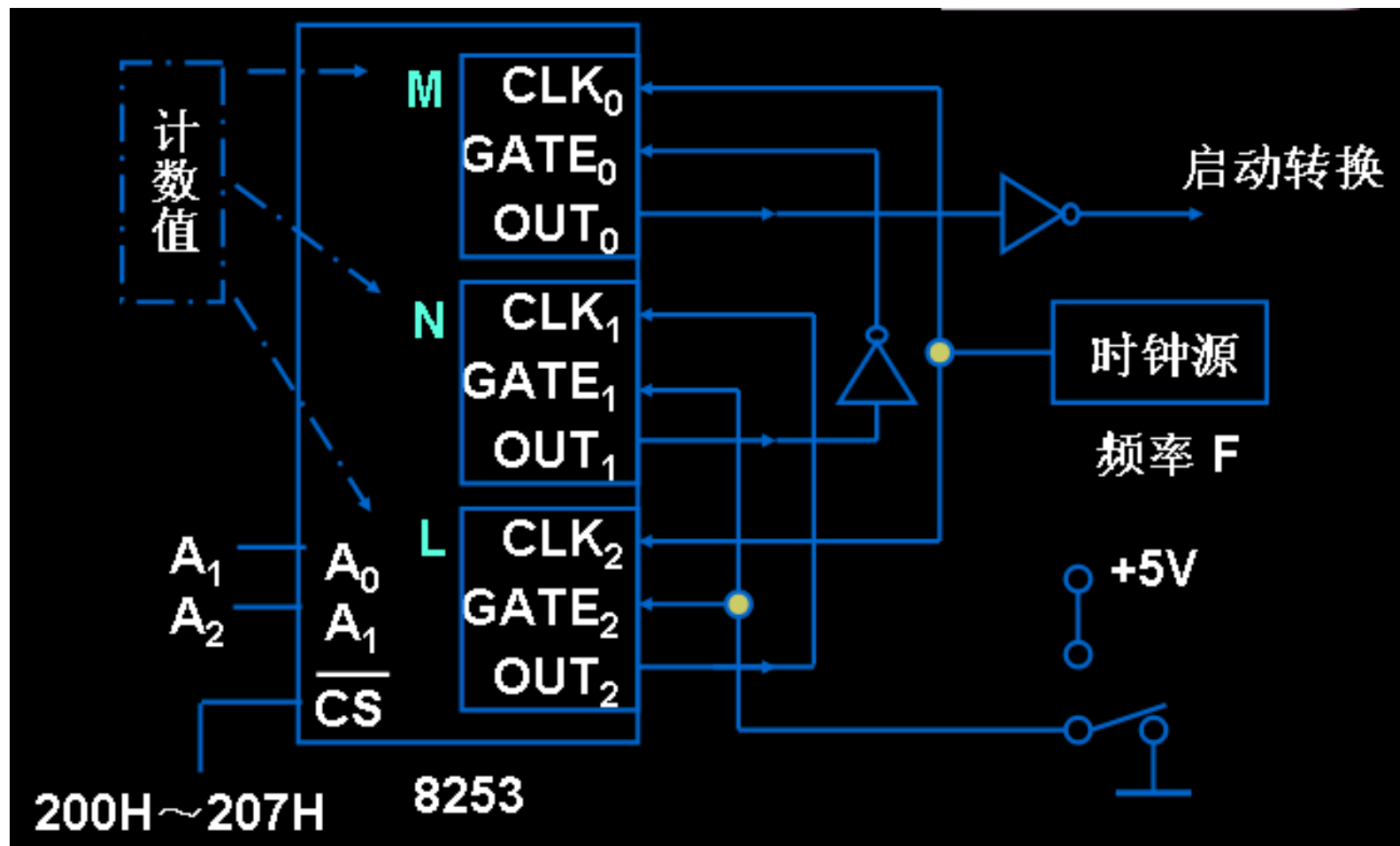
例8.2



例8.2--初始化程序段

- 输出：明确向哪个端口输出什么数据
- 输入：清楚从哪个端口输入什么数据
- **mov dx,203h** ;设置方式控制字
- **mov al,10h**
- **out dx,al**
- **mov dx,200h** ;设置计数初值
- **mov al,64h** ;计数初值为100
- **out dx,al**

例8.3



例8.3--初始化计数器0

- **mov al,14h**
- **mov dx,206h**
- **out dx,al**
- **mov al,cnt0**
- **mov dx,200h**
- **out dx,al**

例8.3--初始化计数器1

- **mov al,52h**
- **mov dx,206h**
- **out dx,al**
- **mov al,cnt1**
- **mov dx,202h**
- **out dx,al**

例8.3--初始化计数器2

- **mov al,96h**
- **mov dx,206h**
- **out dx,al**
- **mov al,cnt2**
- **mov dx,204h**
- **out dx,al**

第8章教学要求

- 1. 掌握8253引脚，尤其是CLK、OUT、GATE引脚的功能
- 2. 掌握8253的六种工作方式、编程和在IBM PC系列机上的应用