

# 定时/计数技术

## 3.1 定时/计数

微机系统需要为处理机和外设提供时间标记和对外部事件进行计数

- 分时系统的程序切换
- 向外设周期性地输出控制信号
- 外部事件发生次数达到规定值后产生中断
- 统计外部事件发生的次数
- .....

## 3.1 定时/计数

### 一、计数/定时

#### — 计算机处理计数问题

1. 首先要将计数量转化为电脉冲的形式（引发计数工作的电脉冲称为计数脉冲）
2. 计数的任务即对电脉冲的出现次数进行度量
3. 要求计数系统具有良好的实时性

# 3.1 定时/计数

## 一、计数/定时

- 计数和定时本质相同

1. 都是对一个输入脉冲进行计数
2. 如果输入脉冲的**频率一定**，那么记录脉冲的个数与所需的时间是一一对应的关系，**计数可以当定时用**

- 定时/计数功能应用

1. 对外部事件发生次数进行**计数**
2. 为处理器和外设提供时间标记（**定时信号**）  
(例如，分时系统的程序切换，键盘去抖，DRAM刷新)

# 3.1 定时/计数

## 二、时间-频率-声音

### — 时间/时间序列

- Newton

绝对时间，不受任何影响，在任何场所都总是以同样速度流动

- Leibniz

时间不过是用来表示事物发生变化先后顺序的一个用词

### — 频率/声音

## 3.1 定时/计数

### 三、定时的分类

- 内部定时

计算机本身运行的时间基准或时序关系，使计算机每个操作都按照严格的时间节拍执行

- 外部定时

外部设备实现某种功能时，在外设与CPU之间或外设与外设之间的时序配合

## 3.1 定时/计数

### 三、定时的分类

- 内部定时

由CPU硬件结构确定，有固定的时序关系，无法更改

- 外部定时

由外设和被控对象的任务、功能决定，无一定模式，需要用户根据I/O设备的需求设定

- 时序配合

外设与CPU连接时，以计算机的时序关系为依据，设计外部定时机构

## 3.1 定时/计数

### 四、定时方法

为获得所需要的定时，要求准确而稳定的时间基准

#### 1. 软件定时

- 基于CPU内部定时机构，运用软件编程利用每执行一条指令需要若干个指令周期的原理，循环执行一段程序而产生等待延时，主要用于**短时延时**



## 3.1 定时/计数

- 优点

不需要增加设备，只需编制相应的延时程序以备调用

- 缺点

延时等待增加了CPU的时间开销，延时时间越长，等待开销越大，降低了CPU的效率，浪费了CPU的资源

## 3.1 定时/计数

— 例如

```
mov cx,0ffffh
```

```
l2: dec cx
```

```
cmp cx,0
```

```
jnz l2
```

```
jmp l1    ;l1为延迟后要跳转的程序段
```

## 3.1 定时/计数

— 例如

delay: push cx ;延时子程序

mov cx, 0100h

dl1: push ax

pop ax

loop dl1

pop cx

ret

## 3.1 定时/计数

### 2. 硬件定时

- 采用可编程通用的定时/计数器或单稳延时电路产生的定时或延时
- 优点

硬件定时不占用CPU的时间，定时时间长，使用灵活，故得到广泛应用

## 3.2 可编程定时/计数器8253

### 可编程定时/计数器

1. 8253的**基本结构**及其**作用**
2. 8253的**工作方式**和**控制字**
3. 8253的读写操作
4. 各种外部定时技术

# 可编程大规模接口芯片

芯片原理

芯片结构

芯片功能

如何编程

初始化

读/写

芯片应用

硬件连接

软件代码

## 3.2 可编程定时/计数器8253

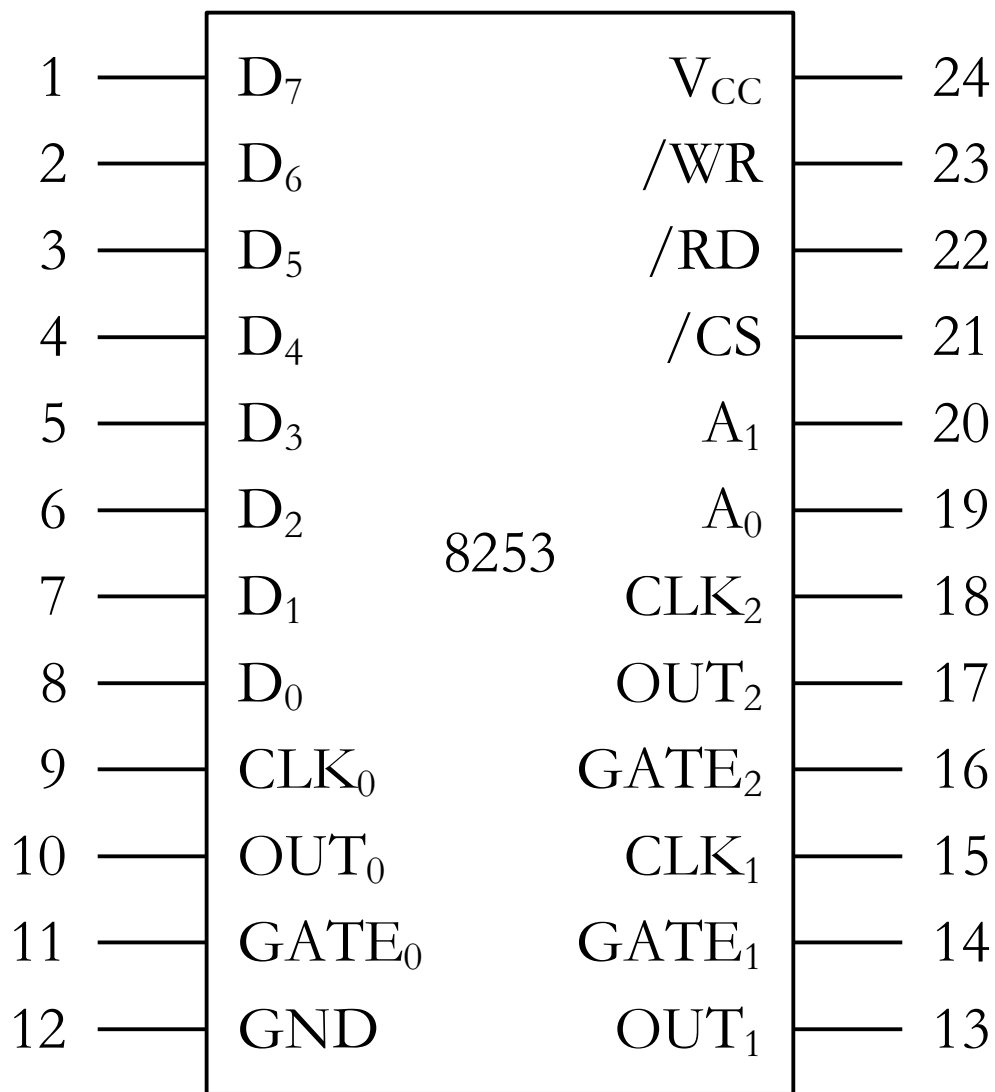
- 可编程定时/计数器8253/8254
  - 8253-5 (5MHz) , 8254-2 (10MHz) , 8253 (2MHz) , 8254 (8MHz) , 8254-5 (5MHz)
  - 3个独立的16位计数器
  - 6种工作方式供选择
  - 数据总线缓冲器和读写控制逻辑
  - 外形、引脚、功能兼容, 最高频率不同

## 3.2 可编程定时/计数器8253

### 一、8253外部特性

- 8253是24引脚双列直插式芯片
- +5V供电
- 数据总线 $D_0-D_7$ ，RD#和WD#分别是读写控制引脚，CS#是片选信号。 $A_1$ ， $A_0$ 是片内地址选择引脚
- 8253的三个计数通道在结构上和功能上完全一样，每个通道均有两个输入引脚CLK和GATE，一个输出信号引脚OUT





定时/计数器8253引脚图

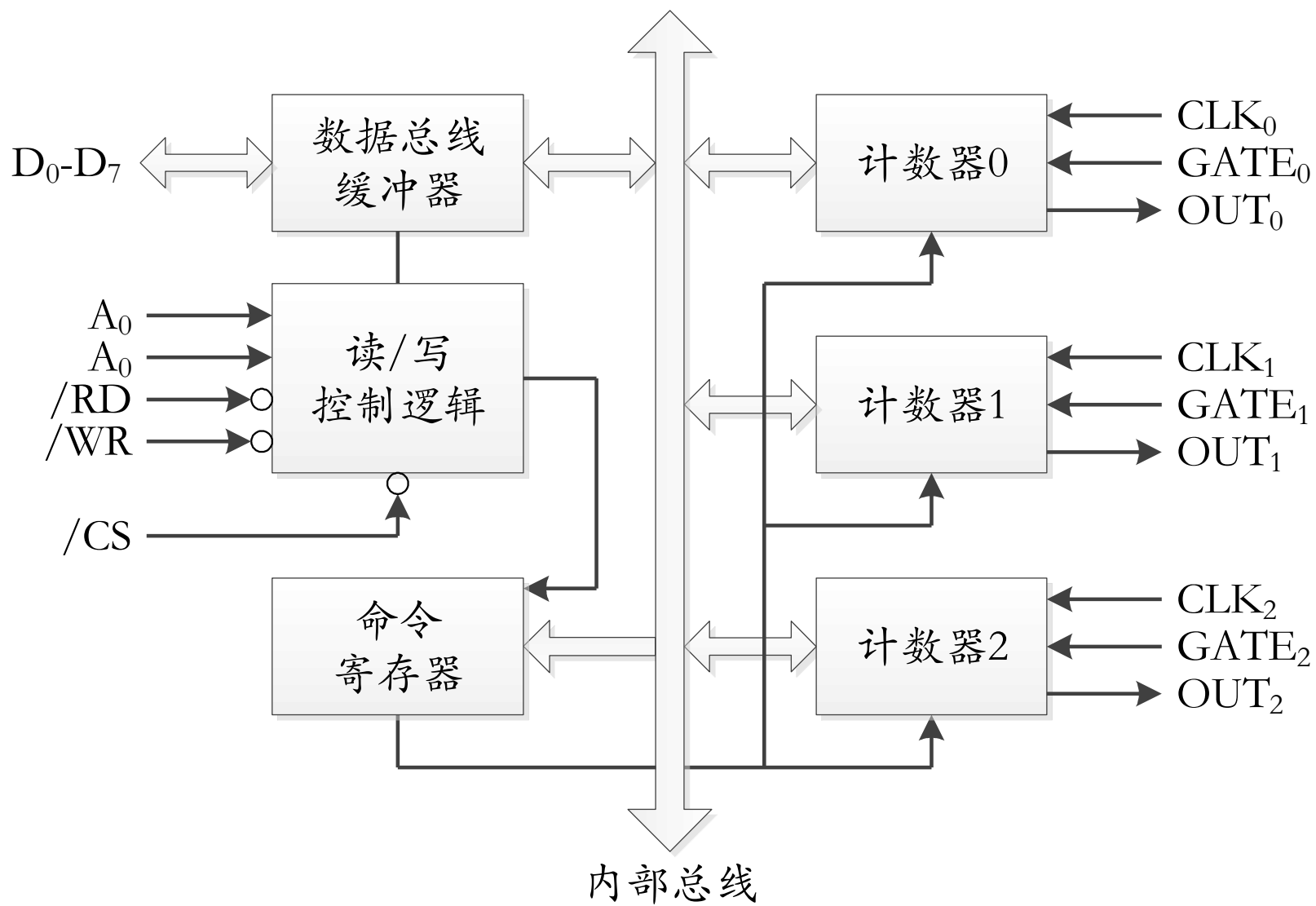
## 3.2 可编程定时/计数器8253

### 二、内部逻辑

#### 1. 8253内部逻辑结构

- 数据缓冲寄存器
- 读写逻辑
- 控制命令寄存器
- 计数器

#### 2. 计数通道内部逻辑



8253内部结构框图

数据总线缓冲器

双向三态8位寄存器

与系统总线D<sub>0</sub>-D<sub>7</sub>相连

写入数据/命令

确定8253工作方式

向寄存器装入初值

读取数据/状态

读出初值/当前值

计数器

3个独立计数通道

16位减1计数器

16位计数初值寄存器

16位输出锁存器

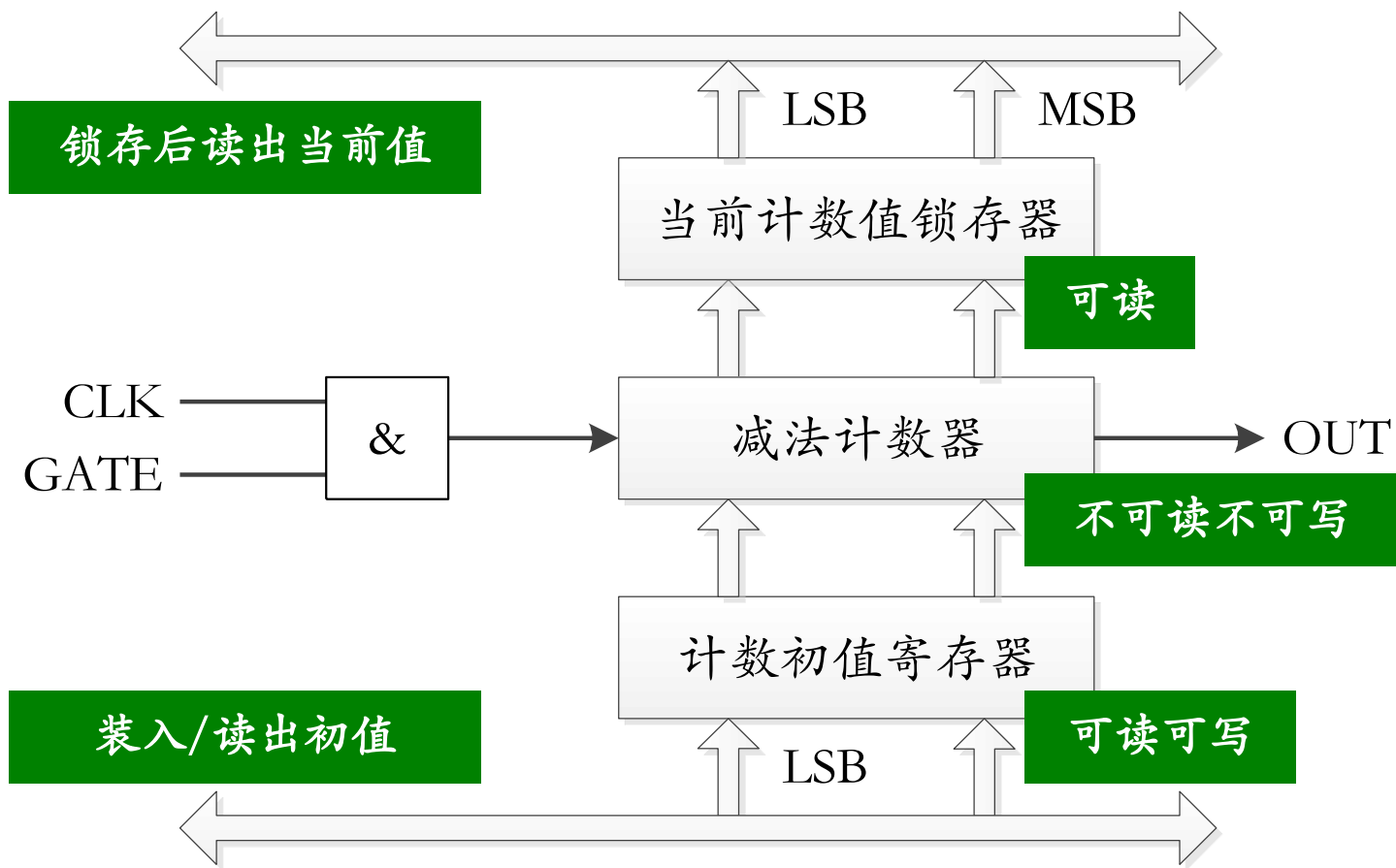
控制寄存器

CPU写入控制字

读/写逻辑

读/写控制

片内地址译码



计数器通道内部逻辑

## 3.2 可编程定时/计数器8253

### 三、计数初值和编程命令

- 计数初值

- 计数初值寄存器用来寄存计数初值
- 计数工作单元为16位减1计数器，它的初值是计数初值寄存器内容
- 计数单元对CLK脉冲计数，每出现一个CLK脉冲，计数器减1，当减为零时，通过OUT输出指示信号表明计数单元已为零

## 3.2 可编程定时/计数器8253

### — 作为定时器工作

当计数单元为零时，计数寄存器内容会自动重新装入计数单元，因为CLK输入是均匀的脉冲序列，所以OUT输出是频率降低了的脉冲序列（相对于CLK信号频率）

### — 作为计数器工作

只关注在CLK端出现（代表事件）的脉冲个数，当CLK端出现了规定个数的脉冲时，OUT输出一个脉冲信号



## 3.2 可编程定时/计数器8253

- 计数器工作过程

1. 将控制字写入控制寄存器，指示8253的工作方式；
2. 将计数初值写入计数寄存器；
3. 计数单元开始工作，对CLK脉冲计数，每出现一个CLK脉冲，计数器减1；
4. 当计数单元减为零时，通过OUT输出指示信号表明计数单元已为零

## 3.2 可编程定时/计数器8253

- 读写操作及编程命令

2种情况：写命令字操作、读当前计数值操作

## 3.2 可编程定时/计数器8253

- 写操作（芯片初始化）

芯片加电后，其工作方式是不确定的，为了正常工作，要对芯片初始化

初始化的工作有两点：

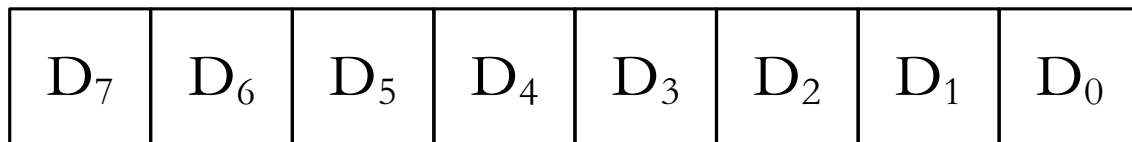
1. 向控制寄存器写入方式控制字
  - 选择计数器（三个中的一个）
  - 确定工作方式（六种方式之一）
  - 指定计数器计数初值的长度、装入顺序、计数值的码制（BCD码或二进制码）
2. 向已选定的计数器按方式控制字的要求写入计数初值

## 3.2 可编程定时/计数器8253

计算公式为：

$$C_i = \text{CLK} / \text{OUT}$$

$C_i$ 计数初值，CLK输入时钟频率，OUT输出时钟频率



00 选择通道 0

01 选择通道 1

10 选择通道 2

11 非法

00 锁定当前计数值

01 读写计数器低 8 位

10 读写计数器高 8 位

11 先读写计数器低 8 位  
再读写计数器高 8 位

0 二进制

1 十进制

000 方式 0

001 方式 1

010 方式 2

011 方式 3

100 方式 4

101 方式 5

工作方式命令格式

## 3.2 可编程定时/计数器8253

例1 选择2号计数器，工作方式3，计数器的初值533H，采用二进制

MOV DX, 307H	;命令口
MOV AL, 10110110B	;2号计数器初始化命令字
OUT DX, AL	;写命令口寄存器
MOV DX, 306H	;2号计数器数据口
MOV AX, 533H	;计数初值
OUT DX, AL	;先送低字节到2号计数器
MOV AL, AH	;取高字节送AL
OUT DX, AL	;后送高字节到2号计数器

## 3.2 可编程定时/计数器8253

例2 选择计数器通道1，工作方式1，按BCD码计数，计数器的初值为十进制4000

MOV AL, 63H

;设置控制字0110 0011B

OUT 43H, AL

;写命令口寄存器

MOV AL, 40H

;设置计数初始值4000H

OUT 41H, AL

;送高字节到计数通道1

## 3.2 可编程定时/计数器8253

例3 选择计数器通道0，工作方式0，8位二进制计数，计数器的初值为4

MOV AL, 10H

;设置控制字0001 0000

OUT 43H, AL

;写命令口寄存器

MOV AL, 4

;设置计数初始值4

OUT 40H, AL

;送初值到计数通道0



## 3.2 可编程定时/计数器8253

例4 选择计数器通道2, 工作方式2, 二进制计数, 计数初值为0304H

MOV AL, B4H

;设置控制字1011 0100

OUT 43H, AL

;写命令口寄存器

MOV AL, 04H

;设置计数器低字节

OUT 42H, AL

;送初值到计数通道2

MOV AL, 03H

;设置计数器高字节

OUT 42H, AL

;送初值到计数通道2

## 3.2 可编程定时/计数器8253

- 读当前计数值（锁存后读操作）

在事件计数器的应用中，需要读出计数过程中的计数值，以便根据这个值做计数判断

8253内部逻辑提供了将当前计数值锁存后读操作功能，具体作法是：

1. 先发一条锁存命令（即方式控制字中的  $RL_1RL_0=00$ ），将当前计数值锁存到输出计数器；
2. 执行读操作，得到锁存器的内容。

## 3.2 可编程定时/计数器8253

- 例如，要求读出并检查1号计数器的当前计数值是否是全“1”（假定计数值中只有低8位），其程序段为：

L: MOV AL, 01000000B	;1号计数器的锁存命令
OUT TIMER+3, AL	;写入控制寄存器
IN AL, TIMER+1	;读1号计数器当前计数值
CMP AL, 0FFH	;比较
JNE L	;非全“1”，再读
HLT	;是全“1”，暂停

## 3.2 可编程定时/计数器8253

### 四、工作方式和特点

8253作为一个可编程计数器/定时器，可以用6种工作模式，不论工作在那种模式，都遵守下面几条基本规则：

1. 控制字写入寄存器时，所有控制逻辑电路立即**复位**，输出端OUT进入**初始状态**；
2. 初值写入后，要经过一个时钟上升沿和一个下降沿，计数执行部件才开始进行计数；

## 3.2 可编程定时/计数器8253

3. 通常在时钟脉冲CLK的上升沿，门控信号GATE被采样
  - 模式0，4中，门控信号为电平触发
  - 模式1，5中，门控信号为上升沿触发
  - 模式2，3中，门控信号为电平或上升沿触发（二种）
4. 在时钟脉冲CLK的下降沿，计数器作减1计数
  - 0是计数器所能容纳的最大初始值  
二进制时，0相当于 $2^{16}$ ；BCD码时，0相当于 $10^4$

## 3.2 可编程定时/计数器8253

工作模式决定以下内容

### 1. 门控信号的影响

- 高电平允许，当GATE=0，即使出现CLK，也不计数（模式0，2，3，4）
- 上升沿允许（上升沿触发）（模式1，5）

## 3.2 可编程定时/计数器8253

### 2. OUT信号的状态

- 写入控制字后
- 计数过程中
- 计数终了

### 3. 计数操作可否重复

- 不可重复 (模式0, 4)
- 自动重复 (模式2, 3)
- 条件重复 (模式1, 5) (GATE上升沿)

## 3.2 可编程定时/计数器8253

### 6种工作模式主要区别

- 输出波形不同
- 启动计数器的触发方式不同
- 计数过程中门控信号GATE对计数操作的影响不同
- 有的工作方式具备“初值自动重装”的功能  
(当计数值减到规定的数值后, 计数初值将会自动地重新装入计数器)



		方式0	方式1	方式2	方式3	方式4	方式5
功能		计数结束输出正跃变信号	单脉冲发生器	频率发生器	方波发生器	单脉冲发生器	单脉冲发生器
启动方式		“软件”启动	“硬件”启动	“软件/硬件”启动	“软件/硬件”启动	“软件”启动	“硬件”启动
输出波形		写入初值后，经过大于N个CLK，输出为高	宽度为N个CLK周期的负脉冲	宽度为1个CLK周期的负脉冲	N为偶数，正脉冲宽均为N/2个CLK周期；奇数时，正脉冲宽为(N+1)/2个CLK周期，负脉冲(N-1)/2个CLK周期	宽度为1个CLK周期的负脉冲	宽度为1个CLK周期的负脉冲
初值重装		——	需GATE上升沿	初值自动重装	初值自动重装	——	需GATE上升沿
计数过程中改变计数初值		立即有效	外部触发后有效	计数到1后有效	外部触发有效/计数结束后有效	立即有效	外部触发后有效
GATE信号的作用	GATE=0	停止计数	——	停止计数	停止计数	停止计数	——
	上升沿	——	启动计数	——	——	——	启动计数
	GATE=1	允许计数	——	允许计数	允许计数	允许计数	——

## 3.2 可编程定时/计数器8253

8253的六种工作方式可归为两类：

- 充当频率发生器
- 作为计数器

从这个角度讨论总结OUT和GATE门的作用

## 3.2 可编程定时/计数器8253

- 与频率发生器有关的工作方式
  - 8253有两种方式与频率发生器有关（方式2和方式3）
  - 对OUT端，方式2提供给用户的是负脉冲，方式3提供给用户的是方波
  - GATE信号要始终保持为高

## 3.2 可编程定时/计数器8253

- 与计数器有关的工作方式
  - 方式0、1和方式4、5与计数器有关
  - 启动计数器的方式有两种
    1. CPU把时间常数写入相应通道后（/WR），计数器就开始工作，称之为**软件启动**方式，在这种启动方式下，GATE要始终保持为高电平，**方式0**和**方式4**为软件启动方式

## 3.2 可编程定时/计数器8253

2. 硬件启动计数器，CPU把时间常数写入计数器后（/WR），即使GATE为高电平，计数器并不工作，只有GATE发生跳变，其上升沿启动计数器工作，**方式1和方式5为硬件启动方式**
- 计数器溢出时，OUT有两种输出形式，**电平（方式0和方式1）或负脉冲（方式4和方式5）**

## 3.2 可编程定时/计数器8253

工作方式（计数器）		输出OUT	
		电平	负脉冲
启动方式	硬件 (GATE上升沿)	方式1	方式5
	软件 (写入初值/WR)	方式0	方式4

## 3.2 可编程定时/计数器8253

例1 现有一个高精密晶体振荡电路，输出信号是脉冲波，频率为1MHz。要求利用8253做一个秒信号发生器，其输出接一发光二极管，以0.5秒点亮，0.5秒熄灭的方式闪烁指示。设8253的通道地址为80H~86H（偶地址）

## 3.2 可编程定时/计数器8253

### 1、时间常数计算

要求用8253作一个分频电路，而且输出是方波，否则发光二极管不可能等间隔闪烁指示。频率为1MHz信号的周期为1微秒，而1Hz信号的周期为1秒，所以分频系数

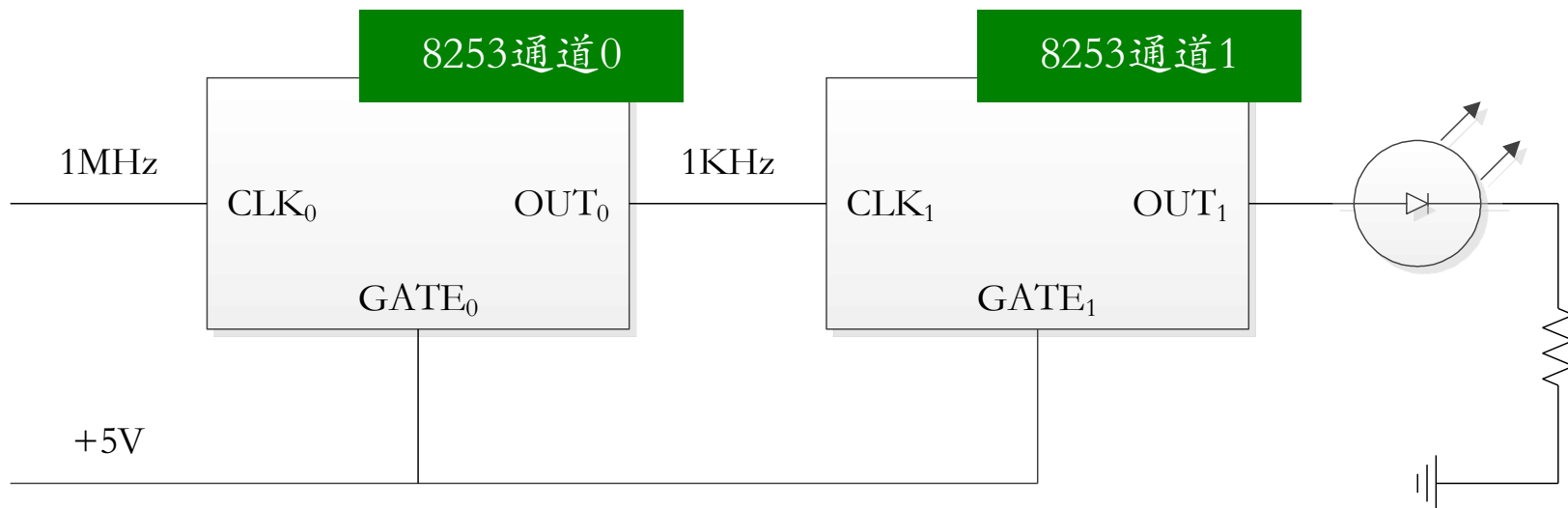
$$N=1000000$$

由于8253一个通道最大的计数值是65536，所以一个通道不能完成上述分频要求

$$N=1000000=1000 \times 1000 = N_1 \times N_2$$

取两个计数器，采用级联方式





秒信号发生器

## 3.2 可编程定时/计数器8253

### 3、工作方式选择

由于通道1要输出方波信号推动发光二极管，  
所以通道1应选工作方式3

对于通道0，只要能起分频作用，对输出波形  
不做要求，所以方式2和方式3都可以选用

对于通道0，取工作方式2，BCD计数；

对于通道1，取工作方式3，二进制计数（也  
可选BCD计数）

## 3.2 可编程定时/计数器8253

### 4、程序

```
mov al, 00110101b      ;通道0控制字
out 86h, al
mov al, 00              ;通道0初始计数值, BCD码计数
out 80h, al
mov al, 10h
out 80h, al
mov al, 01110110b      ;通道1控制字
out 86h, al
mov al, 0e8h            ;通道1初始计数值, 3E8H=1000
out 82h, al
mov al, 03h
out 82h, al
```

## 3.2 可编程定时/计数器8253

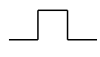
例2 计件系统的功能是记录脉冲的个数，一个脉冲代表一个事件，例如交通道路检测系统中通过检测点的车辆，工业控制系统中流水线上已加工好的工件；要求在计件过程中，PC机可以显示当前计数器的内容，当完成10000个工件记录后，系统发出1KHz信号推动喇叭发音通知用户（外部提供时钟1MHz）

## 3.2 可编程定时/计数器8253

解：1、需要两个通道，一个作为计数，选用通道0。  
另一个产生1KHz信号，选用通道1。

传感器电路把物理事件转换为脉冲信号输入到通道0计数，当记录10000个事件后，通道0计数器溢出，GATE端输出高电平，这时通道1开始工作，产生1KHz信号推动喇叭发音。

1MHz



计件脉冲

CLK<sub>0</sub>

8253通道0

OUT<sub>0</sub>

GATE<sub>0</sub>

+5V

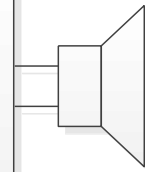
CLK<sub>1</sub>

8253通道1

OUT<sub>1</sub>

GATE<sub>1</sub>

驱动  
放大



计件系统

## 3.2 可编程定时/计数器8253

### 3、工作方式选择

- 对于通道1，由于要产生1KHz信号，故选用**工作方式3**
- 对于通道0， $GATE_0=1$  (+5V)，要求初始计数值写入计数通道后，计数器就可以工作，则通道0的启动方式应是**软件启动**，故可选的工作方式为**方式0**和**方式4**

由于要求计数溢出后产生一个信号来启动一个事件，即喇叭发音，对于图所示方案，通道1的 $GATE_1$ 信号由通道0的OUT信号产生，这个OUT信号应该是**电平型**，所以通道0应选用**工作方式0**

## 3.2 可编程定时/计数器8253

### 4、时间常数

$$N_0 = 10000$$

$$N_1 = 1\text{MHz}/1\text{KHz} = 1000$$



## 3.3 8253应用举例

### 一、8253在PC/XT中的应用

PC机中，8253的端口地址为40H~43H

1. 计数器0 - 向系统日历时钟提供定时中断  
方式3，控制字36H，计数器初始值0
2. 计数器1 - 动态RAM刷新  
方式2，控制字54H，计数器初始值18 (12H)
3. 计数器2 - 控制扬声器发声  
方式3，控制字B6H，计数器初始值1331  
(533H)

## 3.3 8253应用举例

### – BIOS对8253初始化

#### 计数器2

MOV AL, 0B6H ;控制字10,11,011,0

OUT 43H, AL ;通道2, 16位, 方式3, 二进制

MOV AX, 533H ;初值 533H = 1331

OUT 42H, AL ;写入计数器低8位

MOV AL, AH;

OUT 42H, AL ;写入计数器高8位

OUT2输出两次变高的间隔为 $840\text{ns} \times 1331 = 1.12\text{ms}$ ,  
频率为894Hz

(输入时钟频率 $\text{CLK0} = 1.1931816\text{MHz} = 1/840\text{ns}$ )

## 3.3 8253应用举例

### – BIOS对8253初始化

计数器1

MOV AL, 54H ;控制字01,01,010,0  
OUT 43H, AL ;通道1, 低8位, 方式2, 二进制  
MOV AL, 12H ;初值18  
OUT 41H, AL ;写入计数器低8位, 高8位自动为0

OUT1输出两次变高的间隔为 $840\text{ns} \times 18 = 15\mu\text{s}$ , 于是  
2ms内可有132次动态RAM刷新 (128次是系统的  
最低要求)

## 3.3 8253应用举例

### – BIOS对8253初始化

#### 计数器0

MOV AL, 36H	;控制字00,11,011,0
OUT 43H, AL	;通道0, 16位, 方式3, 二进制
MOV AL, 0	;初值0, 计数65536次 (最大)
OUT 40H, AL	;写入计数器的低计数值
OUT 40H, AL	;写入计数器的高计数值

OUT0输出两次变高的间隔为 $840\text{ns} * 65536 = 55\text{ms}$   
(计时单位), 即每隔55ms发出一次请求 (输入时钟频率 $\text{CLK0} = 1.1931816\text{MHz} = 1/840\text{ns}$ )

## 3.3 8253应用举例

### 日时钟运行原理

- 0号计数器，工作方式3，计数初值65536  
输出频率为 $1.1931816\text{MHz}/65536=18.2\text{Hz}$   
也即每隔54.9ms产生一次0级中断，并执行一次中断08H服务程序。
- 中断服务程序TIMER-INT主要工作是对BIOS数据区中**双字**变量的TIMER\_LO和TIMER\_HI进行操作。

## 3.3 8253应用举例

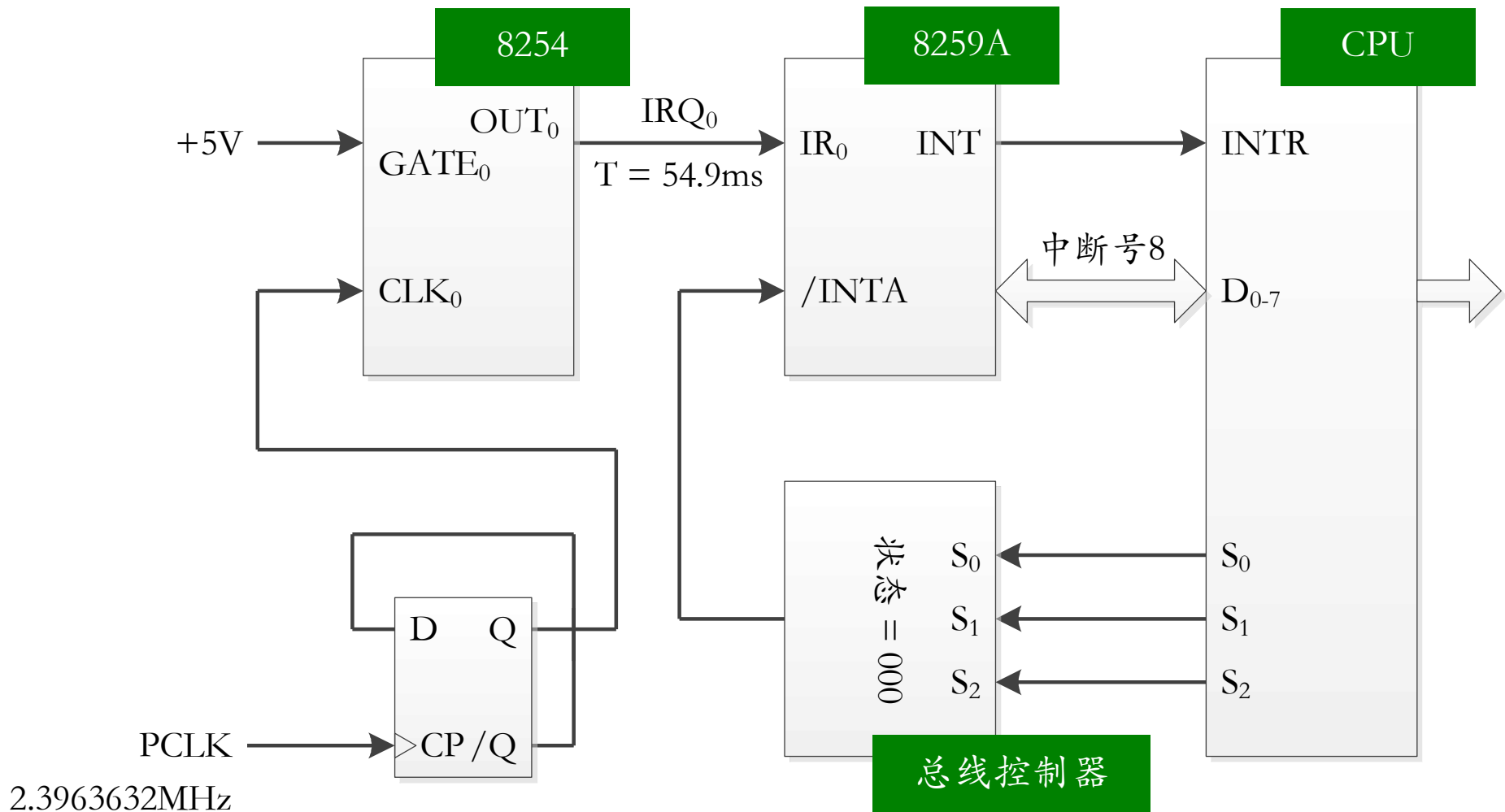
- 日时钟计时

一个小时65543个单位

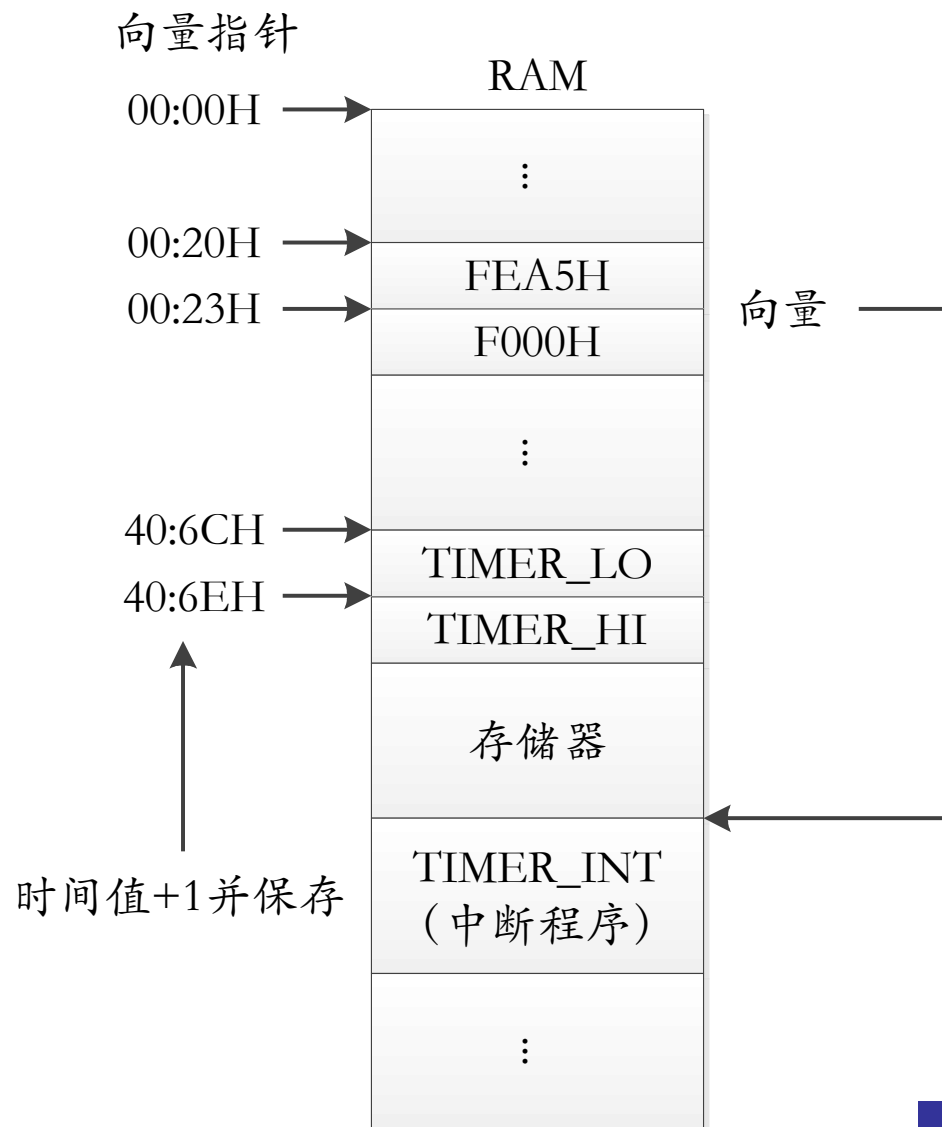
$$3600\text{s}/54.925493\text{ms}=65543$$

24个小时1573040 (001800B0H) 个单位:

$$24 \times 3600\text{s}/54.925493\text{ms} = 1573040$$



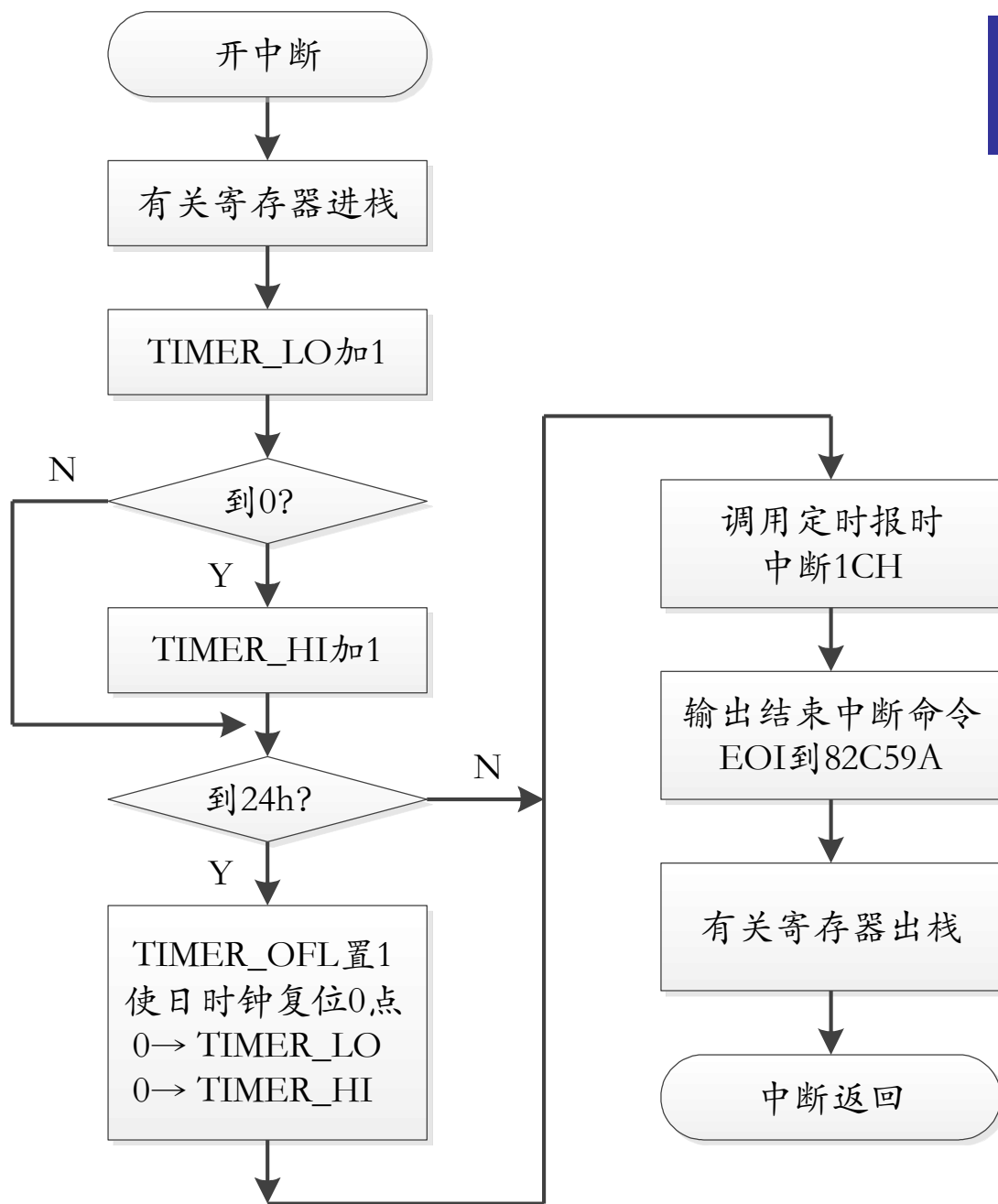
日时钟运行原理



日时钟运行原理



## 日时钟中断 8



## 3.3 8253应用举例

### 二、扬声器控制

设计一个程序，使扬声器发出600Hz频率（音高）的声音，按下任意键声音停止（p43/p55）

- PC机的发声系统以计数器2为核心，CLK2的输入频率1.19MHz
- 改变计数器初值可以由OUT2得到不同频率的方波输出

对于600Hz，计数初值 $1.19\text{MHz}/600\text{Hz}=1983$

## 3.3 8253应用举例

- 发声系统受8255芯片B口的两个输出端线PB0、PB1的控制

PB0为1，使GATE2为1，计数器2能正常计数

PB1为1，打开输出控制门

```

CODE    SEGMENT
        ASSUME CS:CODE
START:
        .....
        IN AL, 61H           ;读8255A的PB口
        OR AL, 03H ;00000011
        OUT 61H, AL         ;打开PB1和PB0
        MOV AX, 1983
        OUT 42H, AL         ;写入计数初值（低位字节）
        MOV AL, AH
        OUT 42, AL          ;写入计数初值（高位字节）
        MOV AH, 01H
        INT 21H             ;DOS中断，等待从键盘输入字符（参见p306）
        IN AL, 61H          ;读8255A的PB口
        AND AL, 0FCH        ;11111100
        OUT 61H, AL         ;关闭PB1和PB0
        MOV AH, 4CH
        INT 21H             ;DOS中断（终止当前程序）
        .....
CODE    ENDS
        END START

```

## 3.3 8253应用举例

### 三、波特率发生器

- 定时常数的计算公式

$$T_c = \text{CLK} / (\text{Baud} * \text{Factor})$$

其中，输入时钟CLK，波特率因子Factor为每传输1位所需要的时钟脉冲个数，波特率Baud为每秒传输的数据位数

## 3.3 8253应用举例

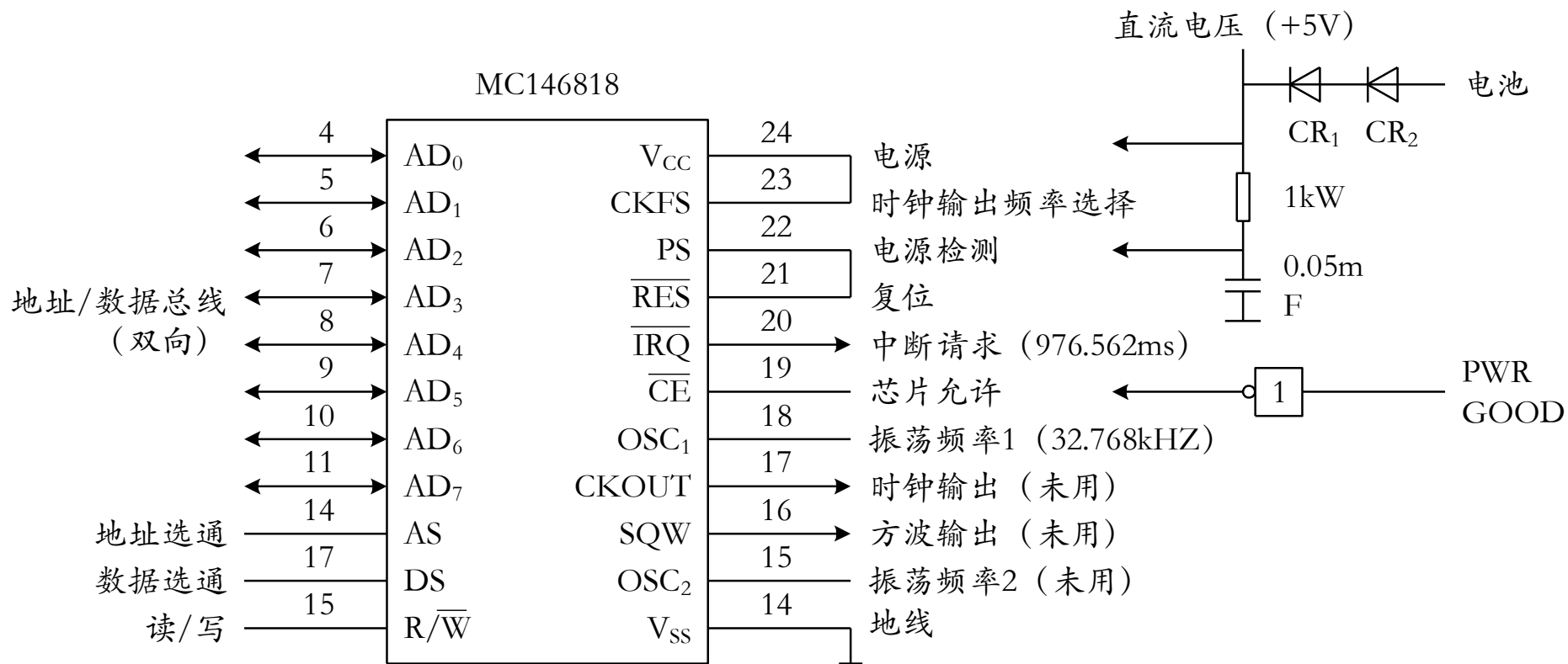
### 四、利用系统硬件定时器延时5秒钟

- 硬中断08H用户无法直接使用，ROM-BIOS的软中断INT1A功能调用，为用户提供了使用8253和实时钟的途径
- 定时系统的BIOS调用建立在定时器8253和实时钟RTC芯片（MC146818）的定时硬中断基础上

## 3.4 实时钟电路MC146818及其应用

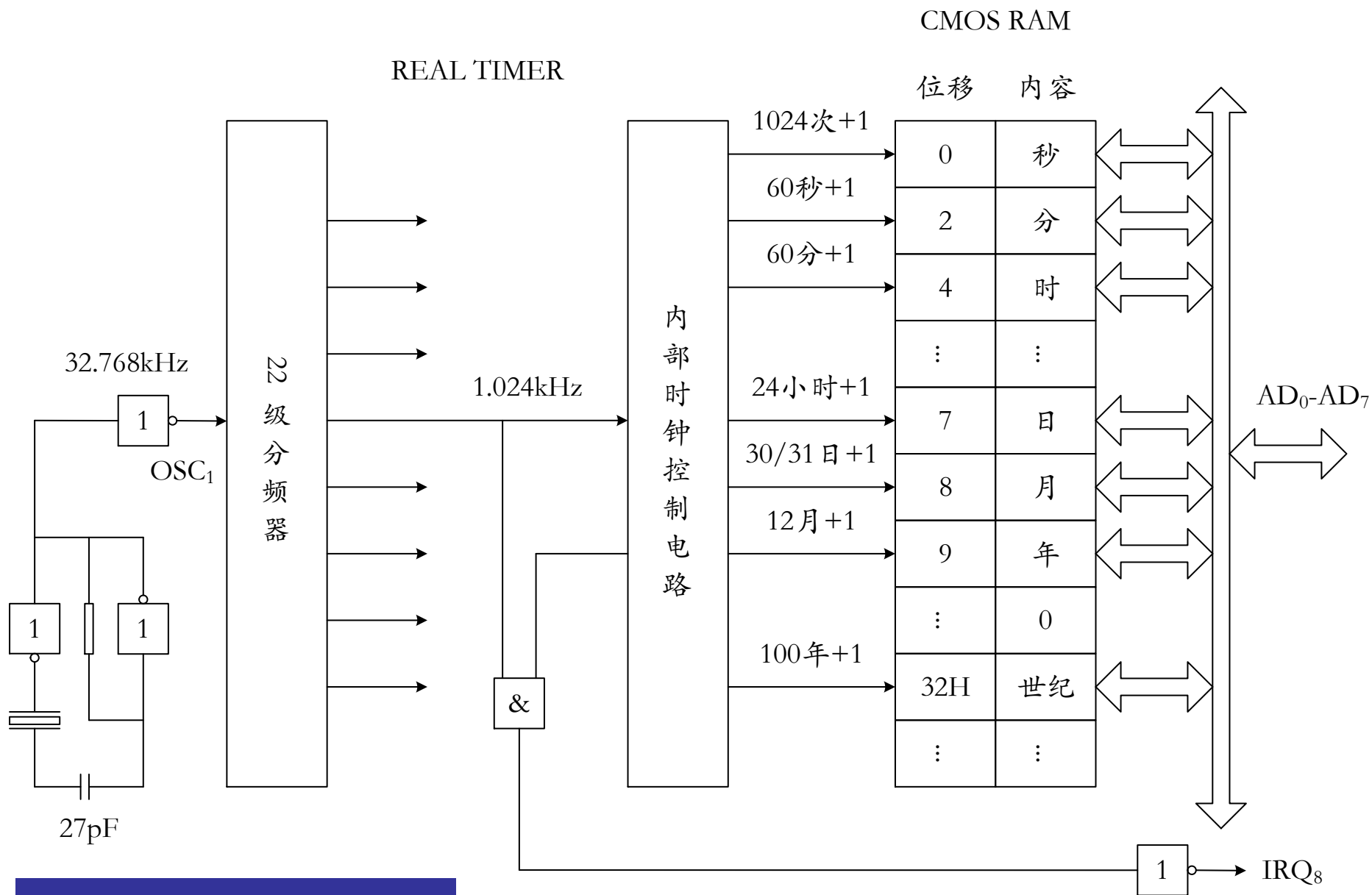
### 一、MC146818的引脚信号和内部结构

- MC146818的引脚信号
- MC146818的内部运行原理
- 实时钟定时信息在CMOS-RAM中的地址分配
- 内部状态寄存器
- 实时钟信息的读写操作和初始化编程



MC146818的引脚及用途





实时钟工作原理

CMOS-RAM实时钟信息存放单元

地址	功能	地址	功能
0	秒	7	日
1	报警秒	8	月
2	分	9	年
3	报警分	0AH	状态寄存器A
4	时	0BH	状态寄存器B
5	报警时	0CH	状态寄存器C
6	星期几	0DH	状态寄存器D
32H	日期世纪		

D<sub>7</sub>      D<sub>6</sub>      D<sub>5</sub>      D<sub>4</sub>      D<sub>3</sub>      D<sub>2</sub>      D<sub>1</sub>      D<sub>0</sub>

A

UIP	DV <sub>2</sub>	DV <sub>1</sub>	DV <sub>0</sub>	RS <sub>3</sub>	RS <sub>2</sub>	RS <sub>1</sub>	RS <sub>0</sub>
时钟 更新	选择22级分频器 输入基准频率			选择22级分频器 输出信号频率			

D<sub>7</sub>      D<sub>6</sub>      D<sub>5</sub>      D<sub>4</sub>      D<sub>3</sub>      D<sub>2</sub>      D<sub>1</sub>      D<sub>0</sub>

B

TE	PIE	AIE	UIE	SQWE	DM	M24	DSE
----	-----	-----	-----	------	----	-----	-----

D<sub>7</sub>      D<sub>6</sub>      D<sub>5</sub>      D<sub>4</sub>      D<sub>3</sub>      D<sub>2</sub>      D<sub>1</sub>      D<sub>0</sub>

C

IRF	PF	AF	UF	保留			
中断 请求	周期 中断	报警 中断	计时 更新				

D<sub>7</sub>      D<sub>6</sub>      D<sub>5</sub>      D<sub>4</sub>      D<sub>3</sub>      D<sub>2</sub>      D<sub>1</sub>      D<sub>0</sub>

D

VRT	保留						
-----	----	--	--	--	--	--	--

状态寄存器内容

## 3.4 实时钟电路MC146818及其应用

- 实时钟信息的读写操作

对RT/CMOS RAM芯片内部的64个字节内容读写操作时，要分两步进行

### 读操作

1. 先把需读出的RAM单元位移地址送到地址端口（070H）；
2. 再从数据端口（071H）读出该单元的内容。

## 3.4 实时钟电路MC146818及其应用

例 要求读取“星期几”的日期信息，则其读出操作的程序段如下：

MOV AL, 6H ;6H是存放“星期几”的单元位移地址

OUT 070H, AL ;送地址端口

JMP SHORT \$+2 ;芯片I/O延时要求

;一条语句2个字节

;直接转移指令需15个时钟周期

IN AL, 071H ;读数据端口

## 3.4 实时时钟电路MC146818及其应用

### 写操作

1. 先把需写入的RAM单元的位移地址送到地址端口（070H）；
2. 再把待写的内容写入数据端口（071H）。

## 3.4 实时钟电路MC146818及其应用

例 要求向4单元写入3小时的时间信息，则其写入操作的程序段如下：

MOV AL, 4H	;4H是存放小时的单元位移地址
OUT 070H, AL	;送地址端口
JMP SHORT \$+2	;芯片I/O延时要求
MOV AL, 3	;3是要求写入的小时数
OUT 071H, AL	;写数据端口

## 3.4 实时钟电路MC146818及其应用

### 二、MC14818在PC/AT机中的应用

- BIOS实时钟的I/O功能调用

实时钟I/O程序INT 1AH的功能（BIOS调用）

- 实时钟对日时钟的初始化

PC/AT配置两种硬件电路分别支持日时钟和实时钟，为使两者协调一致，每次系统启动时要用实时钟信息初始化日时钟的双字变量



## 实时钟I/O程序INT 1AH功能

AH值	功能名称	参数	参数所用的寄存器
0	读日时钟计数	输出	CX=计数高字
1	置日时钟计数	输入	DX=计数低字
2	读实时钟时间	输出	CH=小时 CL=分
3	置实时钟时间	输入	DH=秒 DL=1/0 夏令时/非夏令时
4	读实时钟日期	输出	CH=世纪 (19H或20H)
5	置实时钟日期	输入	CL=年 DH=月 DL=日
6	置实时钟报警	输入	CH=小时 CL=分 DH=秒
7	清实时钟报警	无	

## 3.4 实时时钟电路MC146818及其应用

### 实时钟对日时钟的初始化程序

CMOS-PORT EQU	70H	;CMOS端口基址
CMOS-S	EQU 00H	;实时钟“秒”位移
CMOS-M	EQU 02H	;实时钟“分”位移
CMOS-H	EQU 04H	;实时钟“小时”位移
COUNTS-S	EQU 18	;日时钟“秒单位”
COUNTS-M	EQU 1092	;日时钟“分单位”
COUNTS-H	EQU 7	;日时钟“时单位”

## 3.4 实时时钟电路MC146818及其应用

```
MOV  AL, CMOS-S
OUT  CMOS-PORT, AL
JMP  SHOPT $+2
IN   AL, CMOS-PORT+1 ;读实时钟 “秒”
CMP  AL, 59H
JA   ERROR           ;高于跳转（错误）
CALL CVT-BINARY;转换为二进制
MOV  BL, COUNTS-S
MUL BL;乘以日时钟 “秒单位（18）” 结果AX
MOV  CX, AX
```

## 3.4 实时钟电路MC146818及其应用

```
MOV  AL, CMOS-M
OUT  CMOS-PORT, AL
JMP  SHORT $+2
IN   AL, CMOS-PORT+1 ;读实时钟“分”
CMP  AL, 59H
JA   ERROR;高于跳转（错误）
CALL CVT-BINARY
MOV  BX, COUNTS-M
MUL  BX      ;乘以日时钟“分单位（1092）”
      ;结果DX:AX
ADD  CX, AX
```

## 3.4 实时时钟电路MC146818及其应用

```
MOV    AL, CMOS-H
OUT    CMOS-PORT, AL
JMP     SHORT $+2
IN      AL, CMOS-PORT+1 ;读实时钟的“小时”
CMP     AL, 23H
JA      ERROR ;高于跳转（错误）
CALL    CVT-BINARY
MOV     BL, COUNTS-H
MOV     DX, AX
MUL     BL ;乘以日时钟“时单位（7）”，结果AX
ADD     AX, CX ;其中CX内容来自“分单位”的计算
ADC     DX, 0000H ;带进位位加法
MOV     TIMER-HT, DX ;高字送日时钟
MOV     TIMER-LO, AX ;低字送日时钟
```