

微机原理与接口技术

8088 CPU 外部特性与

IBM PC总线结构

8088 CPU 外部特性与 IBM PC总线结构

•教学重点

- 最小组态下的基本引脚和总线形成
- 最小组态下的总线时序
- IBM PC总线

(1) 信号名称惯例

- 通常在信号名称加上划线（如： $\overline{\mathbf{MX}}$ ）或星号（如： \mathbf{MX}^* ）或井号（如： $\mathbf{\#MX}$ ）表示低电平有效

(2) 8086与8088的区别

- 1.数据线引脚的位数不同：8088数据线引脚为8个，是一个“准16位”的微处理器；8086数据线引脚为16个，是真正的16位微处理器。

- 2.指令队列容量的差别

8088CPU的指令队列只能容纳4个字节，当队列有一个字节空缺时就自动取指，当队列有一个字节就会执行。

8086CPU的指令队列可容纳6个字节，当队列有两个字节空缺时才自动取指，当队列有两个字节才会执行。

- 3.引脚特性的差别

8086定义AD15—AD0引脚为地址/数据复用总线；

8088由于只需用8条数据总线，因此AD7—AD0这8条引脚作为地址/数据复用。

（2）8086与8088的区别（续）

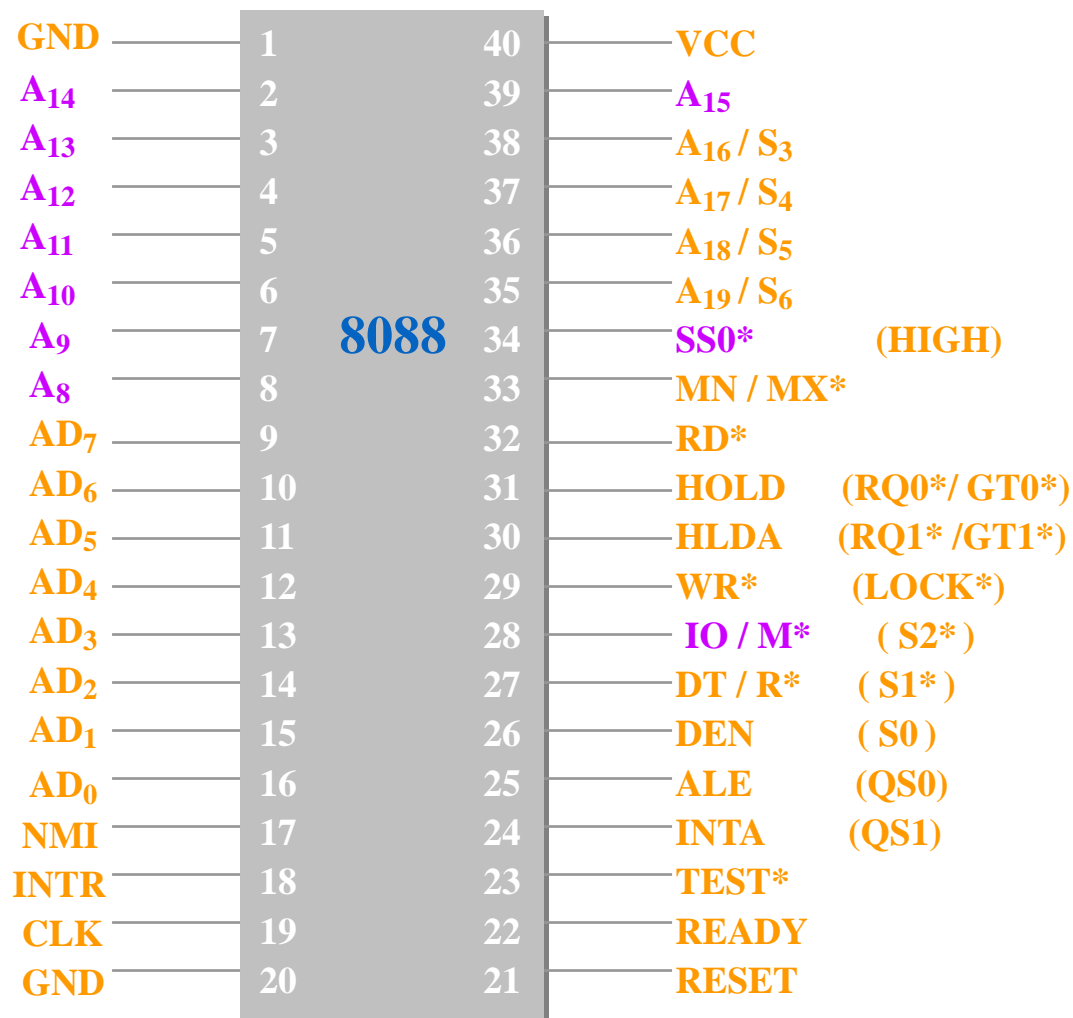
第28号引脚：

8088为IO/M*，8086为IO*/M。

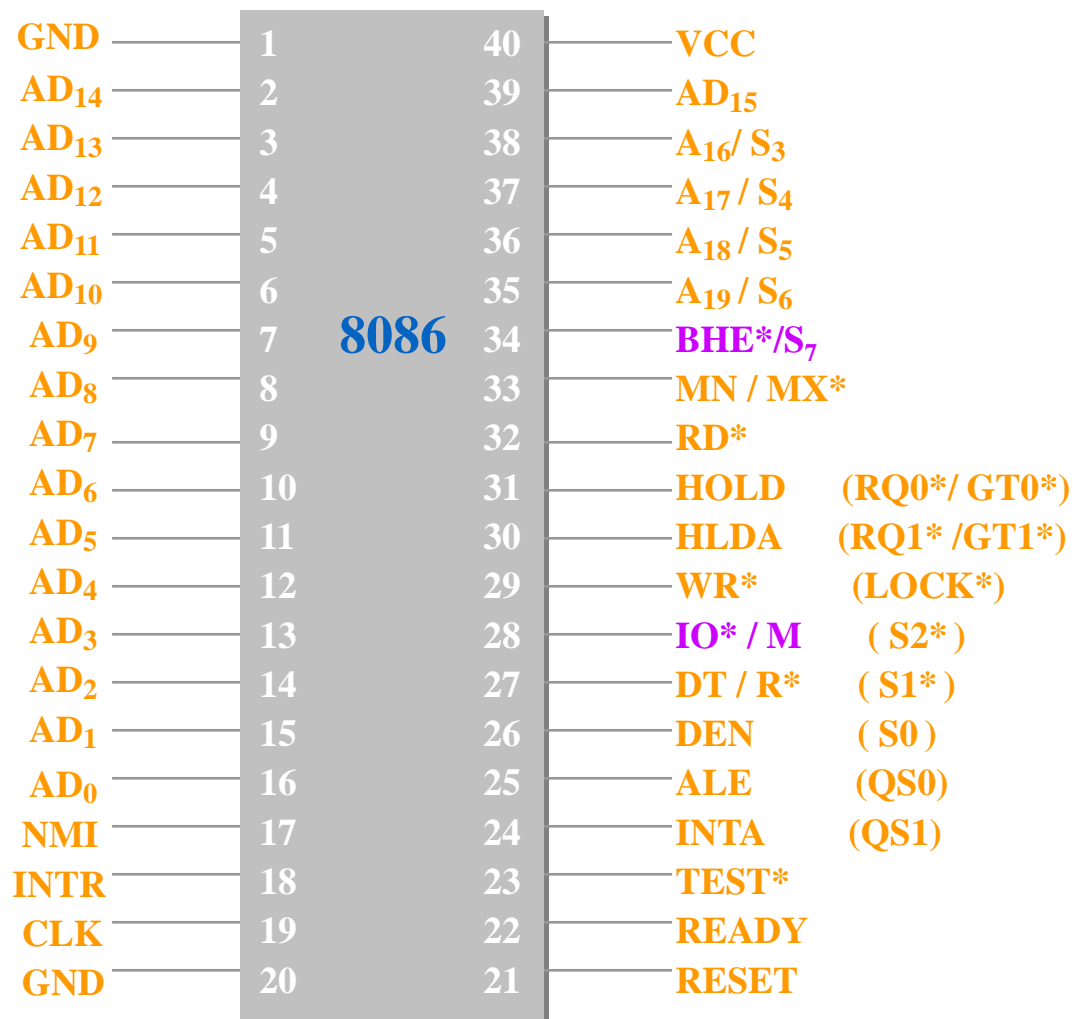
第34号引脚：

8088为SS0*，8086为BHE*/S7（系统未定义状态信号S7的功能）

8088的引脚图



8086的引脚图



(3) 8088的两种组态模式

- 两种组态构成两种不同规模的应用系统
- 最小组态模式
 - 构成小规模的应用系统
 - 8088本身提供所有的系统总线信号
- 最大组态模式
 - 构成较大规模的应用系统，例如可以接入数值协处理器8087
 - 8088和总线控制器8288共同形成系统总线信号

(3) 8088的两种组态模式

- 两种组态利用MN/MX*引脚区别
 - MN/MX*接高电平为最小组态模式
 - MN/MX*接低电平为最大组态模式

(4) 8088最小组态的引脚信号

1. 数据和地址引脚
2. 读写控制引脚
3. 中断请求和响应引脚
4. 总线请求和响应引脚
5. 其它引脚

1.数据 and 地址引脚

- $AD_7 \sim AD_0$ (Address/Data) (8088)
- 地址/数据分时复用引脚，双向、三态
- 分时复用就是一个引脚在不同的时刻具有两个甚至多个作用
- 总线复用的目的是为了减少对外引脚个数
- 8088 /8086CPU的数据地址线采用了总线复用方法

1.数据 and 地址引脚

- **双向：**指信号的传递有两种方向，可以由引脚外向引脚内或相反方向传递。
- **三态：**指引脚上的信号除“0”、“1”状态外，还具有有一种“高阻”状态；当处于“高阻”状态时，引脚可视为与总线间物理断开。

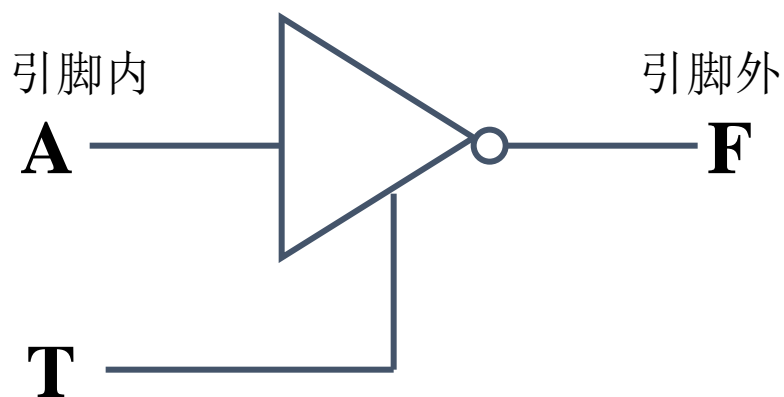
1.数据 and 地址引脚——三态

- 三态功能由三态门器件实现。
- 三态门：功率放大、导通开关
- 器件共用总线时，一般使用三态电路：
 - 需要使用总线的时候打开三态门；
 - 不使用的时候关闭三态门，使之处于高阻

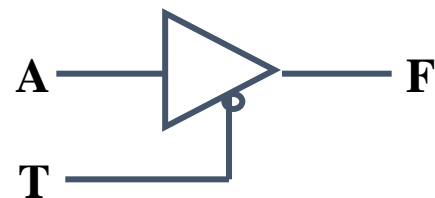
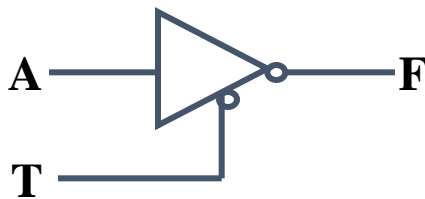
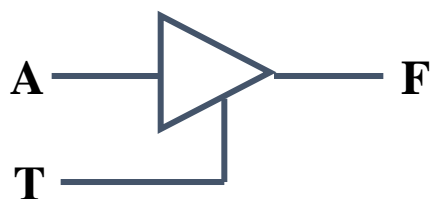
1.数据 and 地址引脚——三态

三态缓冲器（三态门）

具有单向导通和三态的特性



T为低平时：
输出为高阻抗（三态）
T为高电平时：
输出为输入的反相

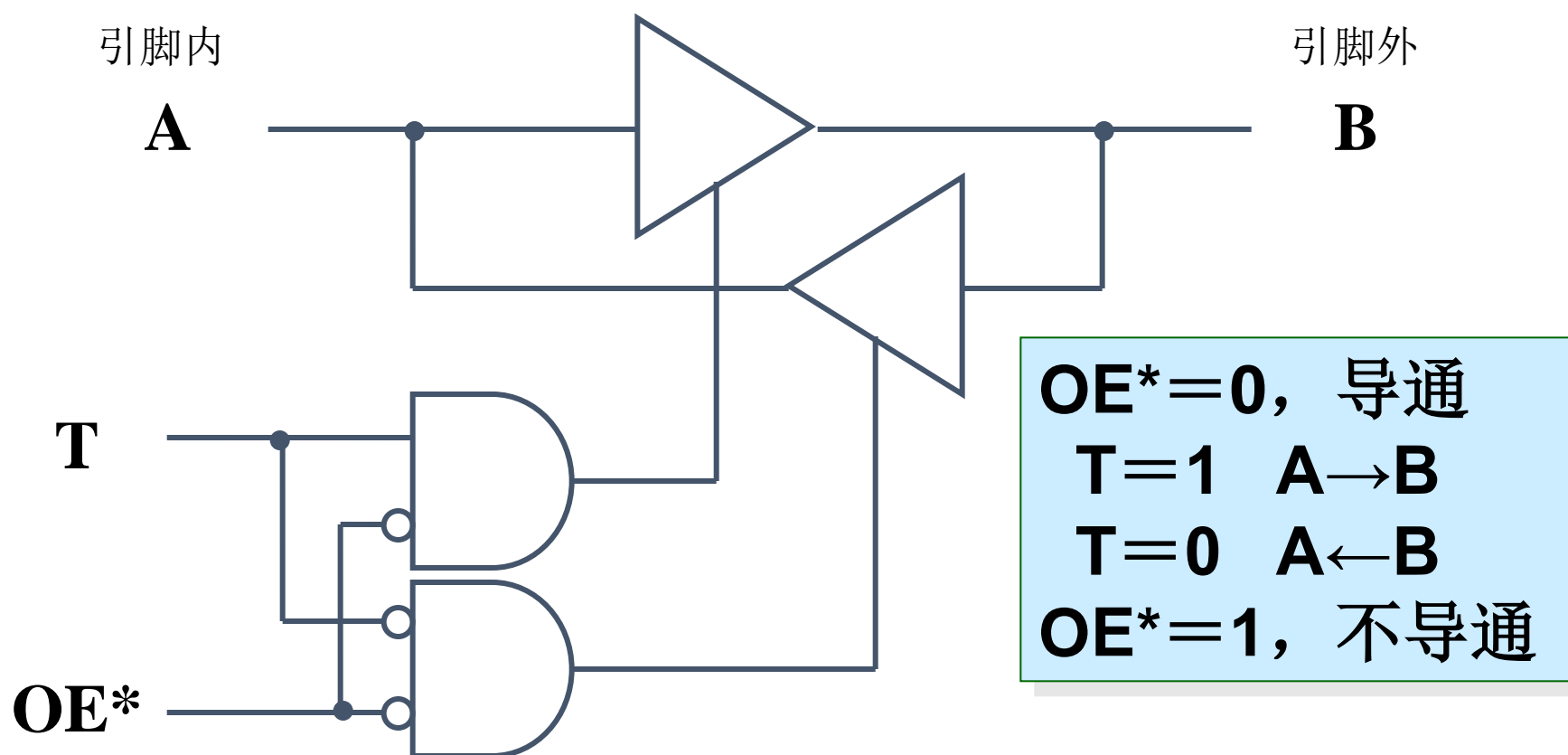


○ 表示反相或低电平有效

1.数据 and 地址引脚——三态

双向三态缓冲器

具有双向导通和三态的特性



1.数据和地址引脚

- 思考：
 - 1) 数据和地址引脚的“双向”特性是针对数据信号还是地址信号？
 - 2) CPU引脚的“三态”特性主要针对什么功能？

1.数据 and 地址引脚

- $AD_7 \sim AD_0$ (Address/Data) (8088)

地址/数据分时复用引脚，双向、三态

- 时序：

在访问存储器或外设的总线操作周期中，这些引脚在第一个时钟周期输出存储器或I/O端口的低8位地址 $A_7 \sim A_0$

T1状态：输出存储器地址或I/O地址

T2状态：输出控制信号

T3和Tw状态：总线操作持续，并检测READY以决定是否延长时序

T4状态：完成数据传送

- 其他时间用于传送8位数据 $D_7 \sim D_0$

1.数据 and 地址引脚

1. 数据 and 地址引脚

- $A_{15} \sim A_8$ (Address)
- 中间8位地址引脚，输出、三态
- 这些引脚在访问存储器或外设时，提供全部20位地址中的中间8位地址 $A_{15} \sim A_8$

1.数据 and 地址引脚

- $A_{19}/S_6 \sim A_{16}/S_3$ (Address/Status)
- 地址/状态分时复用引脚，输出、三态
- 这些引脚在访问存储器的第一个时钟周期输出高4位地址 $A_{19} \sim A_{16}$
- 在访问外设的第一个时钟周期全部输出低电平无效
- 其他时间输出状态信号 $S_6 \sim S_3$

1.数据 and 地址引脚

S_6 始终为低，指示8088是否连接在总线上；

S_5 为标志寄存器FR的中断允许标志IF的状态位，它在每一个时钟周期开始时被修改；

S_4 和 S_3 用以指示是哪一个段寄存器正在被使用，00为ES，01为SS，10为CS，11为DS。

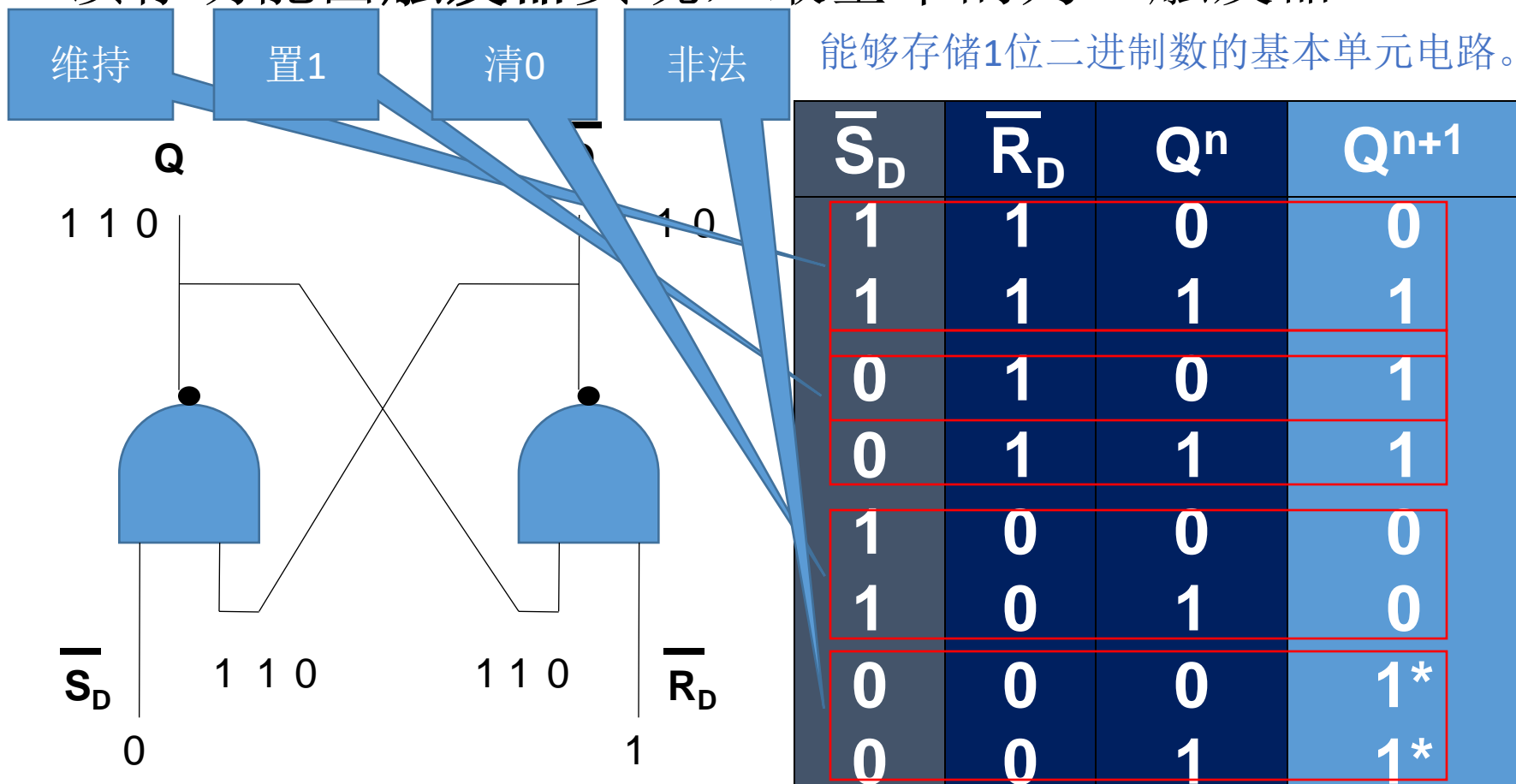
在DMA方式时这4条线处于高阻状态。

2. 读写控制引脚

- ALE (Address Latch Enable) 地址锁存允许引脚，输出、三态、高电平有效
- ALE引脚高有效时，表示复用引脚： $AD_7 \sim AD_0$ 和 $A_{19}/S_6 \sim A_{16}/S_3$ 正在传送地址信息
- 由于地址信息在这些复用引脚上出现的时间很短，所以系统可以利用ALE引脚将地址锁存起来

2. 读写控制引脚——锁存

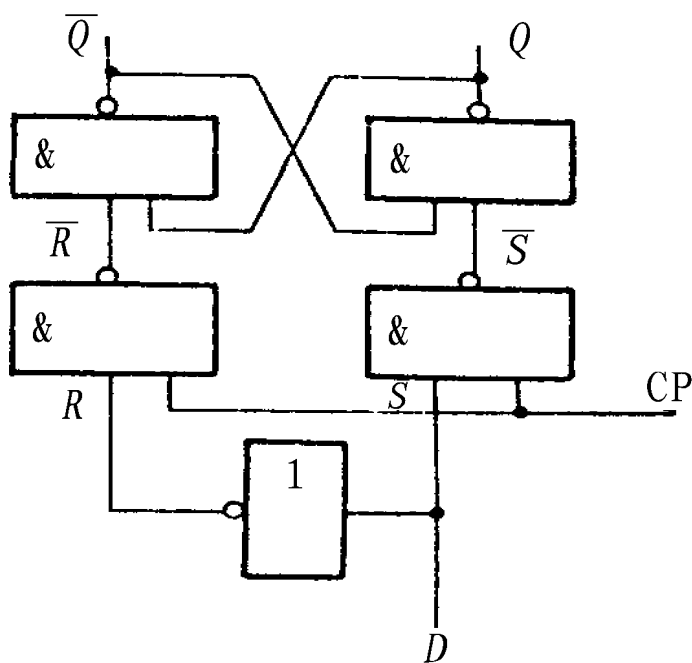
- 锁存功能由**触发器**实现，最基本的为RS触发器。



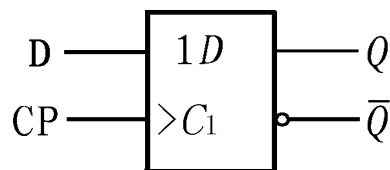
2. 读写控制引脚——锁存

D触发器：信号保持，也可用作导通开关

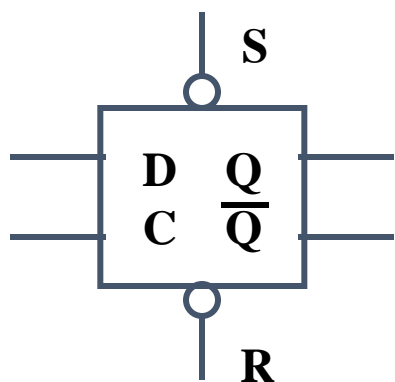
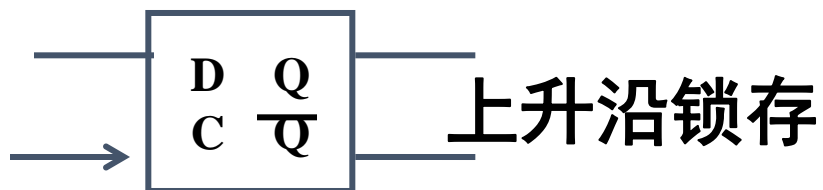
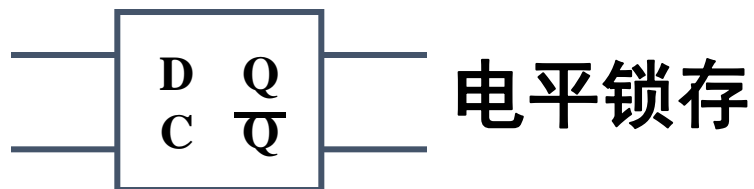
时钟D触发器工作原理： $S = \bar{R} = D$ 。其它与RS触发器相同。



- 避免瞬态干扰产生违反约束条件的输入信号。
- CP 控制脉冲或写入脉冲。



2. 读写控制引脚——锁存

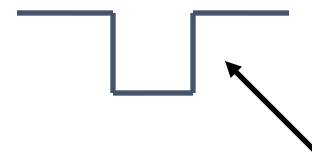


电平锁存:

高电平通过, 低电平锁存

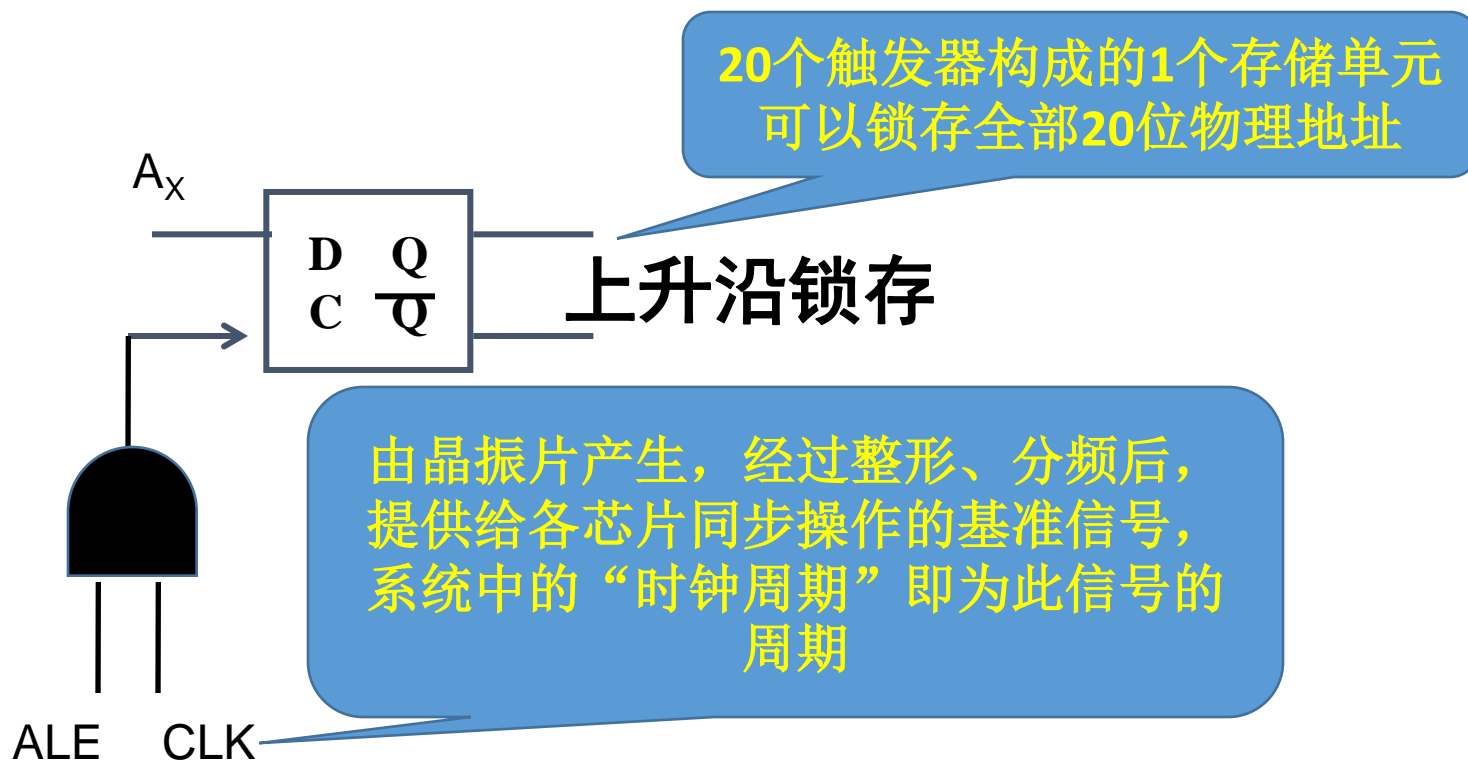
上升沿锁存:

通常用负脉冲触发锁存



2. 读写控制引脚

- 考虑地址引脚与ALE引脚如何在总线上形成地址锁存。



实际中，电平锁存也可不接CLK。

2. 读写控制引脚

- IO/M* (Input and Output/Memory)
- I/O或存储器访问，输出、三态
- 该引脚输出高电平时，表示CPU将访问I/O端口，这时地址总线 $A_{15} \sim A_0$ 提供16位I/O口地址
- 该引脚输出低电平时，表示CPU将访问存储器，这时地址总线 $A_{19} \sim A_0$ 提供20位存储器地址

2. 读写控制引脚

- **WR* (Write)**
- 写控制，输出、三态、低电平有效
- 有效时，表示CPU正在写出数据给存储器或I/O端口

2. 读写控制引脚

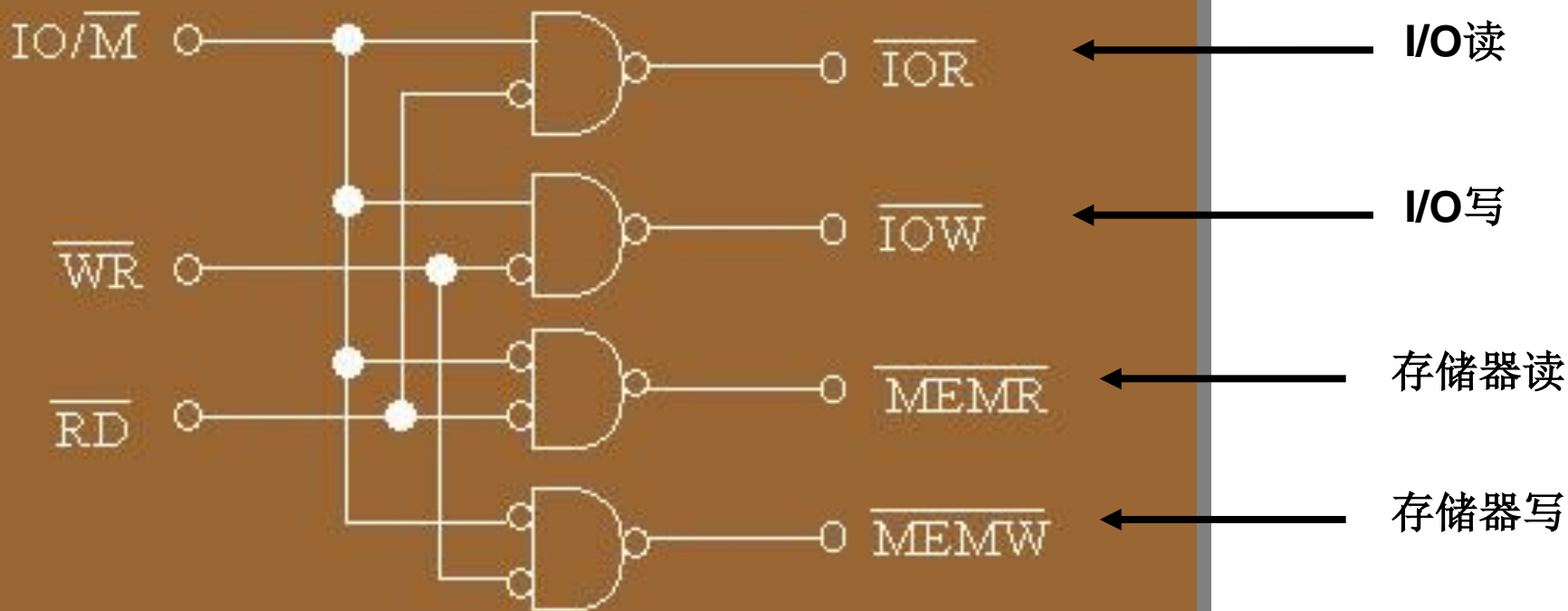
- RD* (Read)
- 读控制，输出、三态、低电平有效
- 有效时，表示CPU正在从存储器或I/O端口读入数据

2. 读写控制引脚

- IO/M*、WR*和RD*是最基本的控制信号
- 组合后，控制4种基本的总线周期

总线周期	IO/M*	WR*	RD*
存储器读	低	高	低
存储器写	低	低	高
I/O读	高	高	低
I/O写	高	低	高

2. 读写控制引脚——基本控制信号的组合方法（IBM-PC总线）



总线读写控制信号的转换

2. 读写控制引脚

T1状态：输出存储器地址或I/O地址

T2状态：输出控制信号

T3和Tw状态：总线操作持续，并检测READY以决定是否延长时序

T4状态：完成数据传送

- READY
- 存储器或I/O口就绪，**输入**、高电平有效
- 在总线操作周期中，8088 CPU会在**第3个时钟周期的前沿（下降沿）**测试该引脚
 - 如果测到高有效，CPU直接进入第4个时钟周期
 - 如果测到无效，CPU将插入等待周期Tw
- CPU在等待周期中仍然要监测READY信号，有效则进入第4个时钟周期，否则继续插入等待周期Tw。

2. 读写控制引脚

- DEN* (Data Enable)
- 数据允许，输出、三态、低电平有效
- 有效时，表示当前数据总线上正在传送数据，可利用它来控制对数据总线的驱动

2. 读写控制引脚

- DT/R* (Data Transmit/Receive)
- 数据发送/接收，输出、三态
- 该信号表明当前总线上数据的流向
- 高电平时数据自CPU输出（发送）
- 低电平时数据输入CPU（接收）

2. 读写控制引脚

- SS0* (System Status 0)
- 最小组态模式下的状态输出信号
- 它与IO/M*和DT/R*一道，通过编码指示CPU在最小组态下的8种工作状态：
 1. 取指
 2. 存储器读
 3. 存储器写
 4. 过渡状态
 5. 中断响应
 6. I/O读
 7. I/O写
 8. 暂停 (HLT)

系统级操作为低电平，
指令级操作为高电平。

3. 中断请求和响应引脚

- INTR (Interrupt Request)
- 可屏蔽中断请求，**输入**、高电平有效
- 有效时，表示请求设备向CPU申请可屏蔽中断
- 该请求的优先级别较低，并可通过关中断指令CLI清除标志寄存器中的**IF标志**、从而对中断请求进行屏蔽

3. 中断请求和响应引脚

- INTA* (Interrupt Acknowledge)
- 可屏蔽中断响应，**输出**、低电平有效
- 有效时，表示来自INTR引脚的中断请求已被CPU响应，CPU进入中断响应周期
- 中断响应周期是连续的两个，每个都发出有效响应信号，以便通知外设他们的中断请求已被响应、并令有关设备将中断向量号送到数据总线

3. 中断请求和响应引脚

NMI（Non-Maskable Interrupt）

- 不可屏蔽中断请求，**输入、上升沿有效**
- 有效时，表示外界向CPU申请不可屏蔽中断
- 该请求的优先级别高于INTR，并且不能在CPU内被屏蔽
- 当系统发生紧急情况时，可通过他向CPU申请不可屏蔽中断服务

3. 中断请求和响应引脚

- 主机与外设进行数据交换通常采用可屏蔽中断
- 不可屏蔽中断通常用于处理掉电等系统故障

4. 总线请求和响应引脚

- **HOLD**
- 总线保持（即**总线请求**），输入、高电平有效
- 有效时，表示总线请求设备向**CPU**申请占有总线
- 该信号从有效回到无效时，表示总线请求设备对总线的使用已经结束，通知**CPU**收回对总线的控制权
- **DMA**控制器等主控设备通过**HOLD**申请占用系统总线（通常由**CPU**控制）

4. 总线请求和响应引脚

- HLDA (HOLD Acknowledge)
- 总线保持响应 (即**总线响应**)，**输出、高电平有效**
- 有效时，表示CPU已响应总线请求并已将总线释放
- 此时CPU的**地址总线、数据总线及具有三态输出能力的控制总线**将全面呈现高阻，使总线请求设备可以顺利接管总线
- 待到总线请求信号HOLD无效，总线响应信号HLDA也转为无效，CPU重新获得总线控制权

5. 其它引脚

- RESET
- 复位请求，输入、高电平有效
- 该信号有效，将使CPU回到其初始状态；当它再度返回无效时，CPU将重新开始工作
- 8088/86复位后
CS=FFFFH、IP=0000H，
所以程序入口在物理地址FFFF0H

5. 其它引脚

- CLK (Clock)
- 时钟输入
- 系统通过该引脚给CPU提供内部定时信号。8088的标准工作时钟为5MHz
- IBM PC/XT机的8088采用了4.77MHz的时钟，其周期约为210ns
- IBM PC/XT机**时钟信号由8284时钟信号发生器提供。**

5. 其它引脚

- Vcc
- 电源输入，向CPU提供+5V电源
- GND
- 接地（输入），向CPU提供参考地电平

5. 其它引脚

- MN/MX* (Minimum/Maximum)
- 组态选择，输入
- 接高电平时，8088引脚工作在最小组态；反之，8088工作在最大组态

5. 其它引脚

- TEST*
- 测试，输入、低电平有效
- 该引脚与WAIT指令配合使用
- 当CPU执行WAIT指令时，他将在每个时钟周期对该引脚进行测试：如果无效，则程序踏步并继续测试；如果有效，则程序恢复运行
- 也就是说，WAIT指令使CPU产生等待，直到引脚有效为止
- 在使用协处理器8087时，通过引脚和WAIT指令，可使8088与8087的操作保持同步

浮点处理指令后面有WAIT指令。
有esc前缀的指令会由8087执行。

(4) 8088最小组态的引脚信号——小结

- **CPU、接口芯片**

脚信号上，
个方面：

- (1) 引脚的
- (2) 信号的
- (3) 有效电
- (4) 三态能力

输出正常的低电平、
高电平外，还可以输出
高阻的第三态

（4）8088最小组态的引脚信号——小结

CPU引脚是系统总线的基本信号
可以分成三类信号：

- 8位数据线： $D_0 \sim D_7$ （双向，三态）
- 20位地址线： $A_0 \sim A_{19}$ （单向，三态）
- 控制线：
 - ALE、IO/M*、WR*、RD*、READY
 - INTR、INTA*、NMI, HOLD、HLDA
 - RESET、CLK、V_{cc}、GND

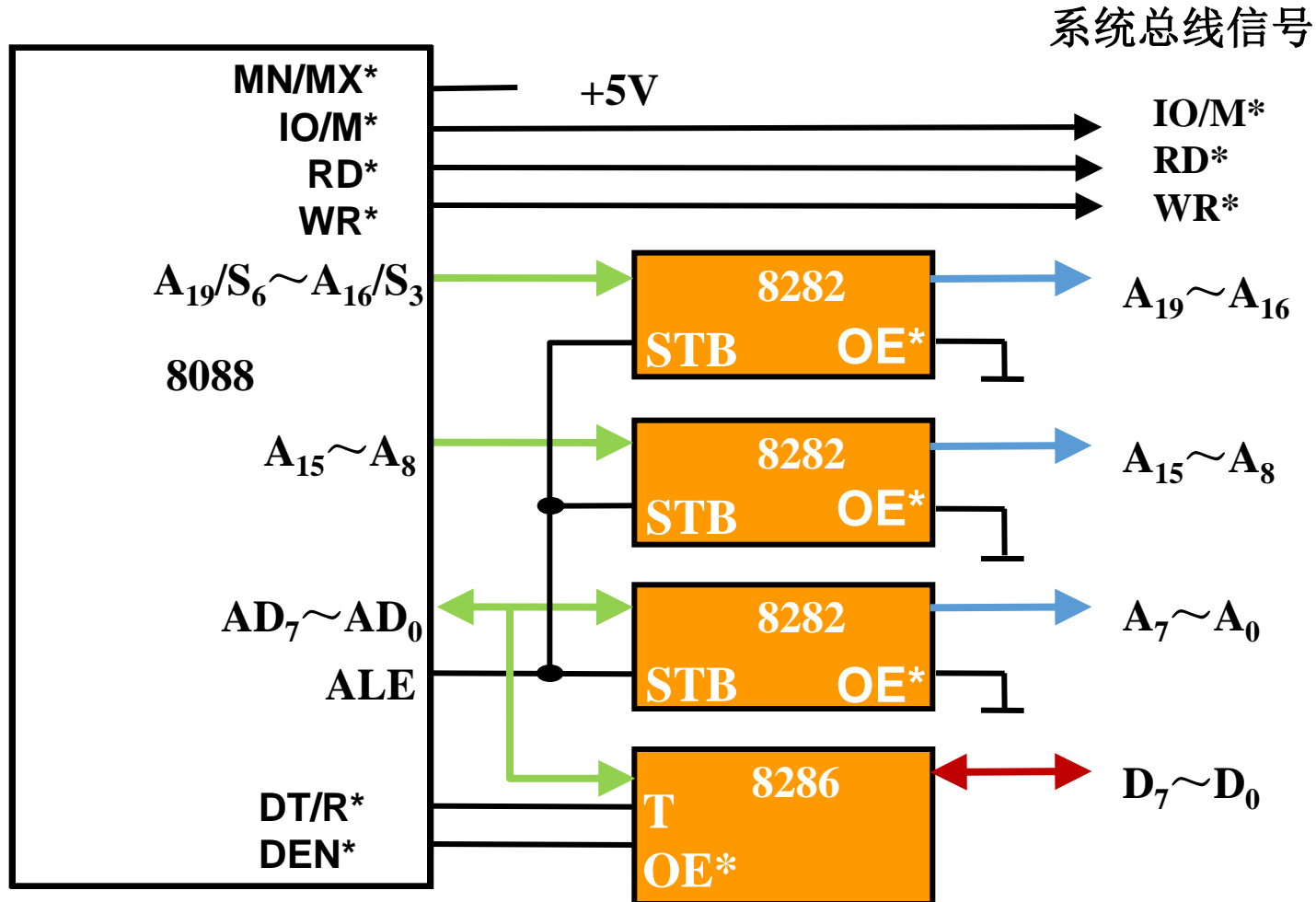
（4）8088最小组态的引脚信号——小结

英文缩写	功能定义	信号的流向	有效电平	三态能力
ALE	地址锁存允许	输出	高电平	—
IO/M*	I/O或存储器访问	输出	高电平/低电平	√
WR*	写控制	输出	低电平	√
RD*	读控制	输出	低电平	√
READY	存储器或I/O口就绪	输入	高电平	—
INTR	可屏蔽中断请求	输入	高电平	—
INTA*	可屏蔽中断响应	输出	低电平	—
NMI	不可屏蔽中断请求	输入	上升沿	—
HOLD	总线保持（即总线请求）	输入	高电平	—
HLDA	总线保持响应（即总线响应）	输出	高电平	—
RESET	复位请求	输入	高电平	—

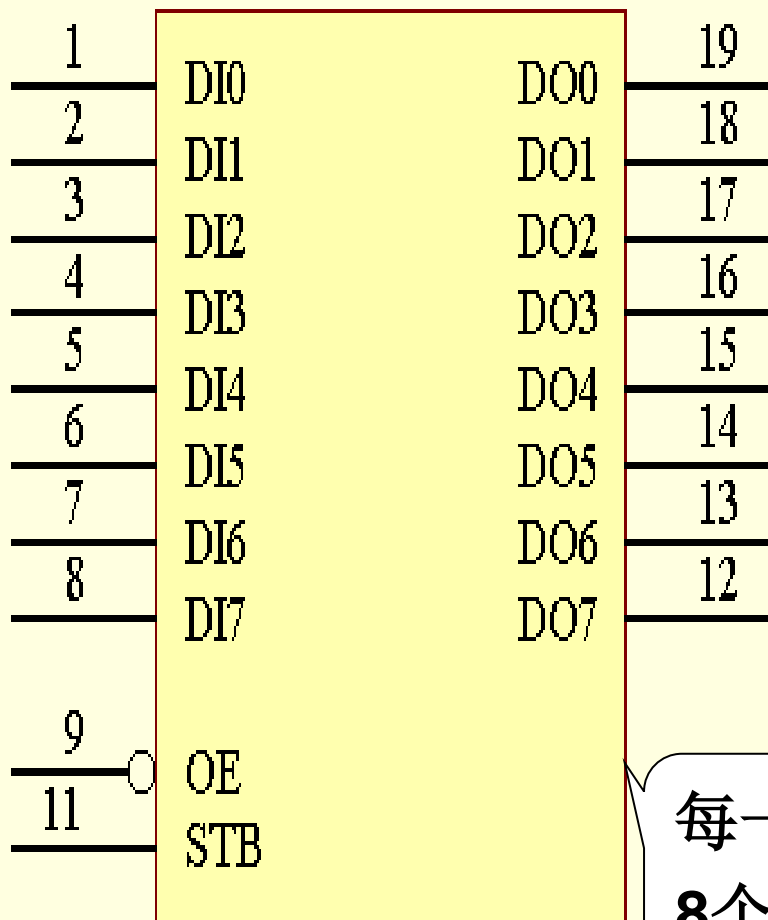
(4) 8088最小组态的引脚信号——小结

- 后续问题:
- CPU引脚如何与外部连接?
- 总线形成
- CPU引脚是如何相互配合，实现总线操作、控制系统工作的呢?
- 总线时序

(5) 最小组态的总线形成



1) 8282芯片外部引脚



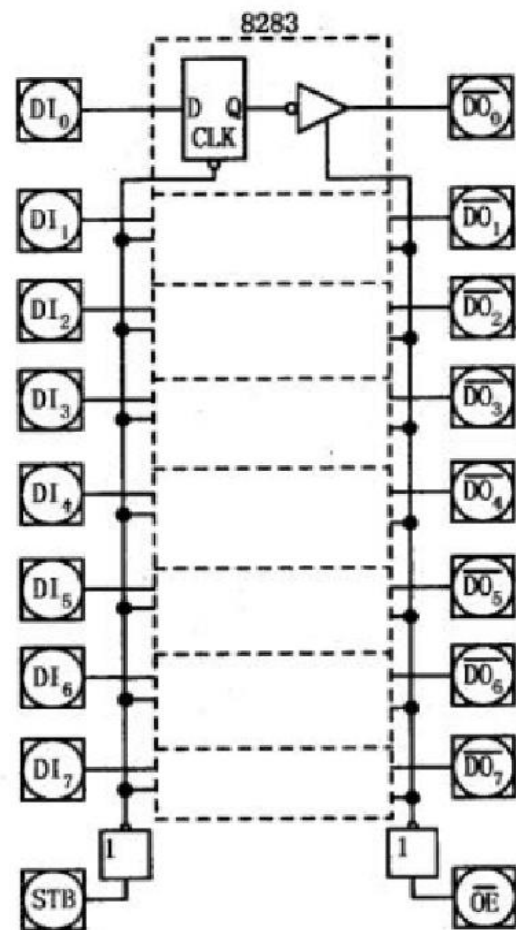
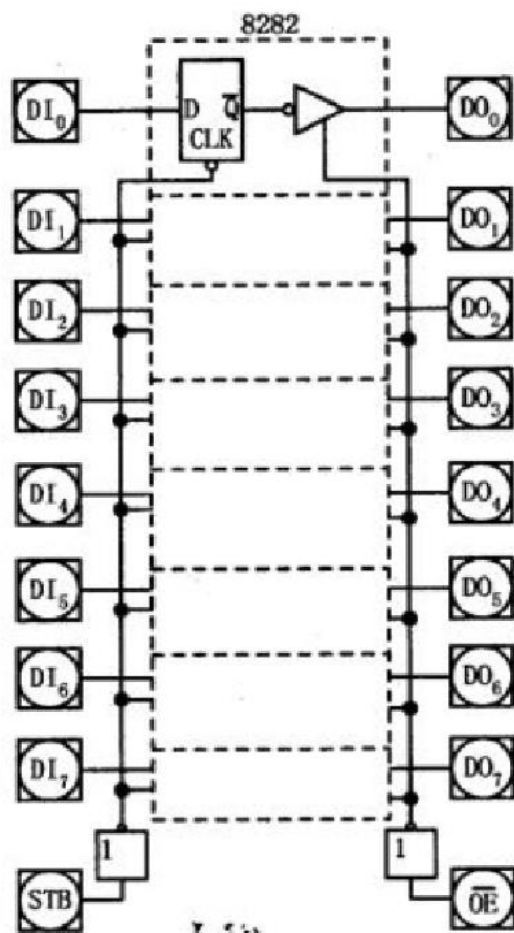
具有三态输出的
TTL电平锁存器

(+5V等价于逻辑“1”，0V等价于逻辑“0”)

STB 电平锁存引脚
OE* 输出允许引脚

每一位都是一个三态锁存器，
8个三态锁存器的控制端连在一起

2) 8282芯片内部结构



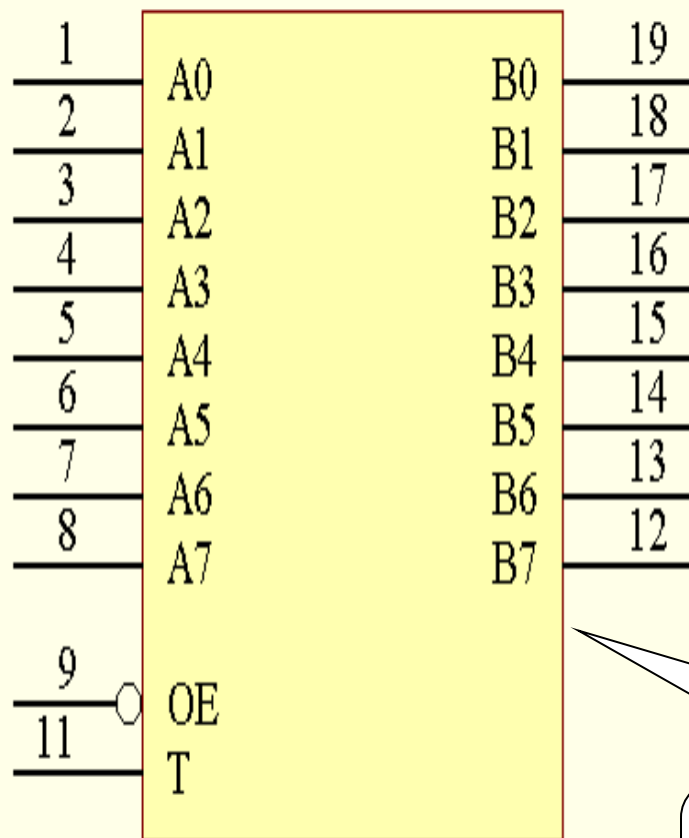
3) 20位地址总线的形成

- 采用3个8282进行锁存和驱动
- Intel 8282是三态透明锁存器，类似有Intel 8283和通用数字集成电路芯片74LS373。

比8282的驱动能力更强。

- **三态输出：**
 - 输出控制信号有效时，允许数据输出；
 - 无效时，不允许数据输出，呈高阻状态
- **透明：** 锁存器的输出能够跟随输入变化

4) 8286芯片外部引脚

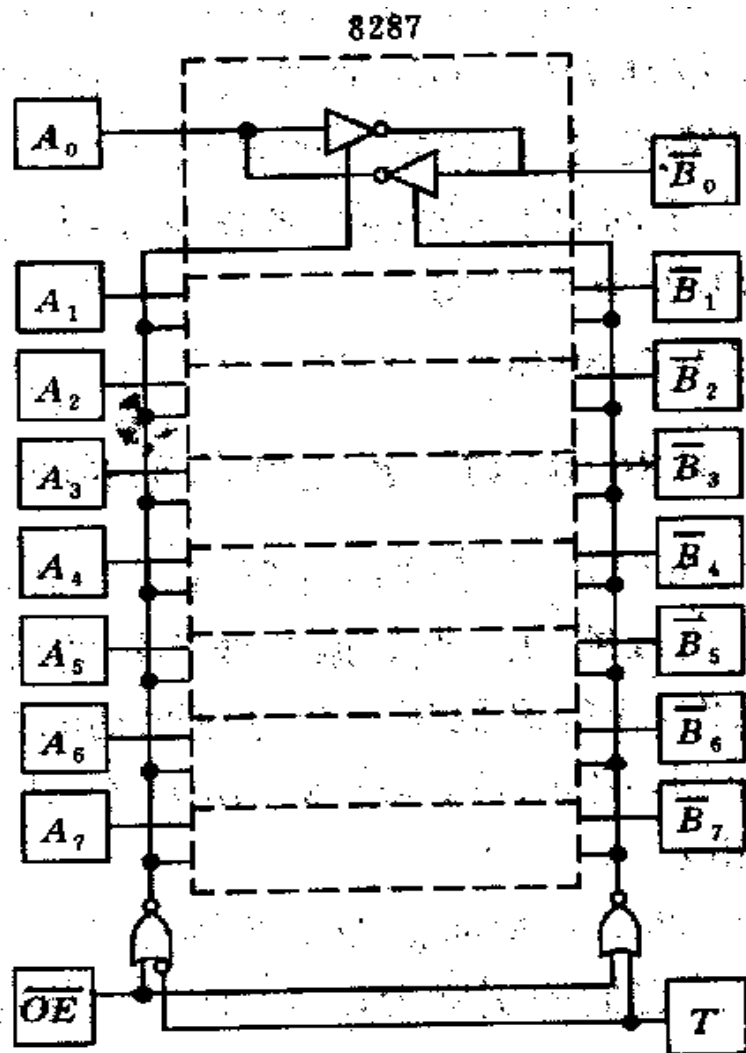
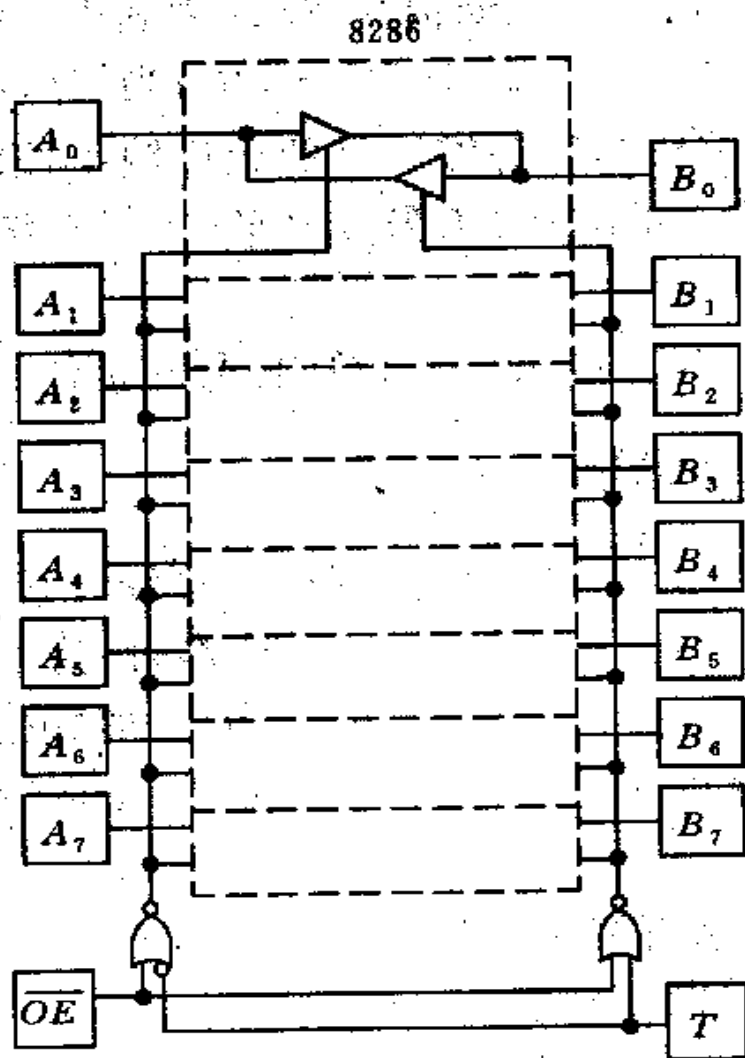


8286

OE*=0, 导通
T=1 A→B
T=0 A←B
OE*=1, 不导通

每一位都是一个双向三态门，
8位具有共同的控制端

5) 8286芯片内部结构



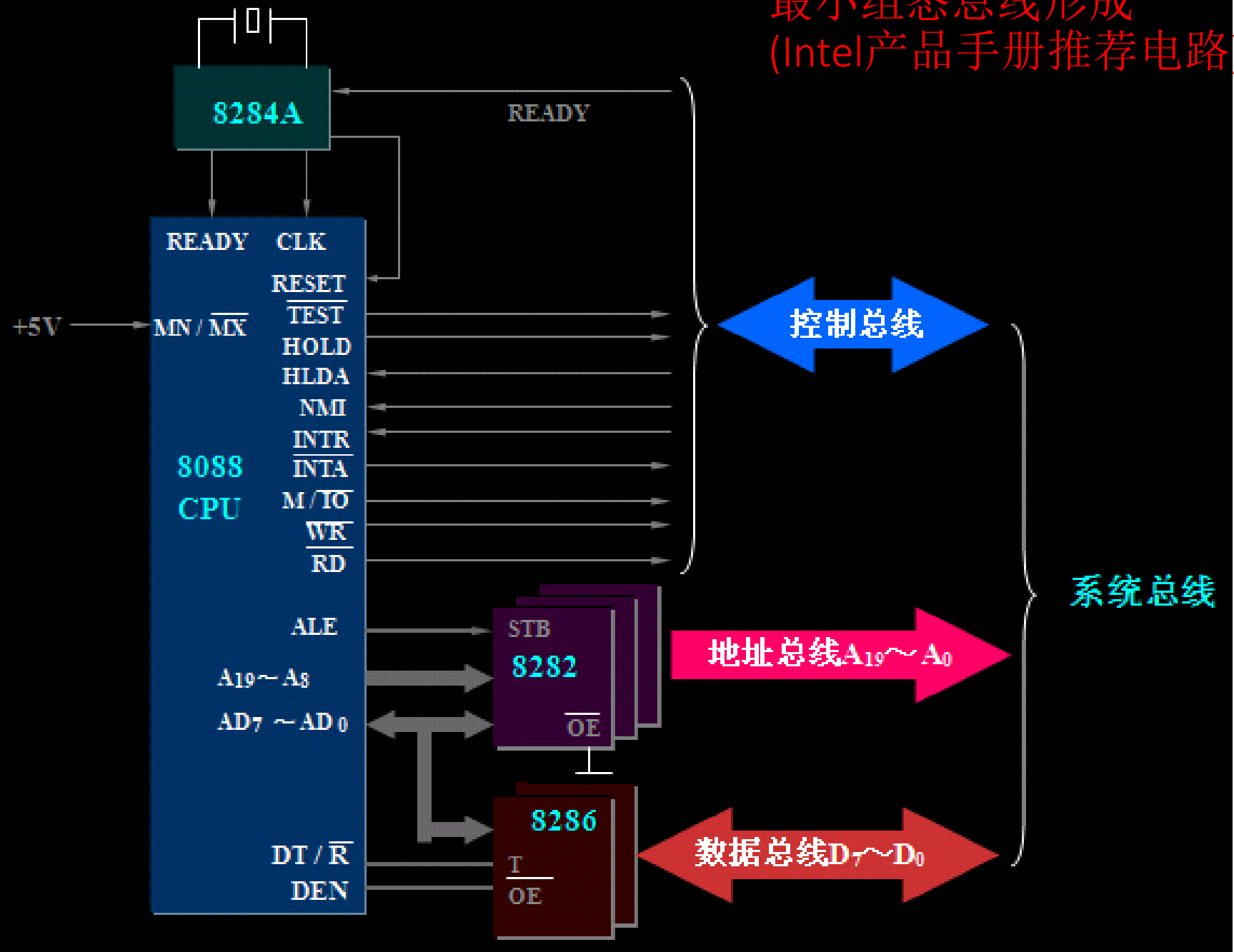
6) 8位数据总线的形成

- 采用数据收发器8286进行双向驱动
- Intel 8286是8位三态双向缓冲器，类似功能的器件还有Intel 8287、通用数字集成电路74LS245等
- 另外，接口电路中也经常使用三态单向缓冲器，例如通用数字集成电路74LS244就是一个常用的双4位三态单向缓冲器

(5) 最小组态的总线形成

- **(1) 20位地址总线——采用3个三态透明锁存器8282进行锁存和驱动**
- **(2) 8位数据总线——采用数据收发器8286进行驱动**
- **(3) 系统控制信号——由8088引脚直接提供（其中少量信号需要外接简单的组合逻辑电路来生成）**

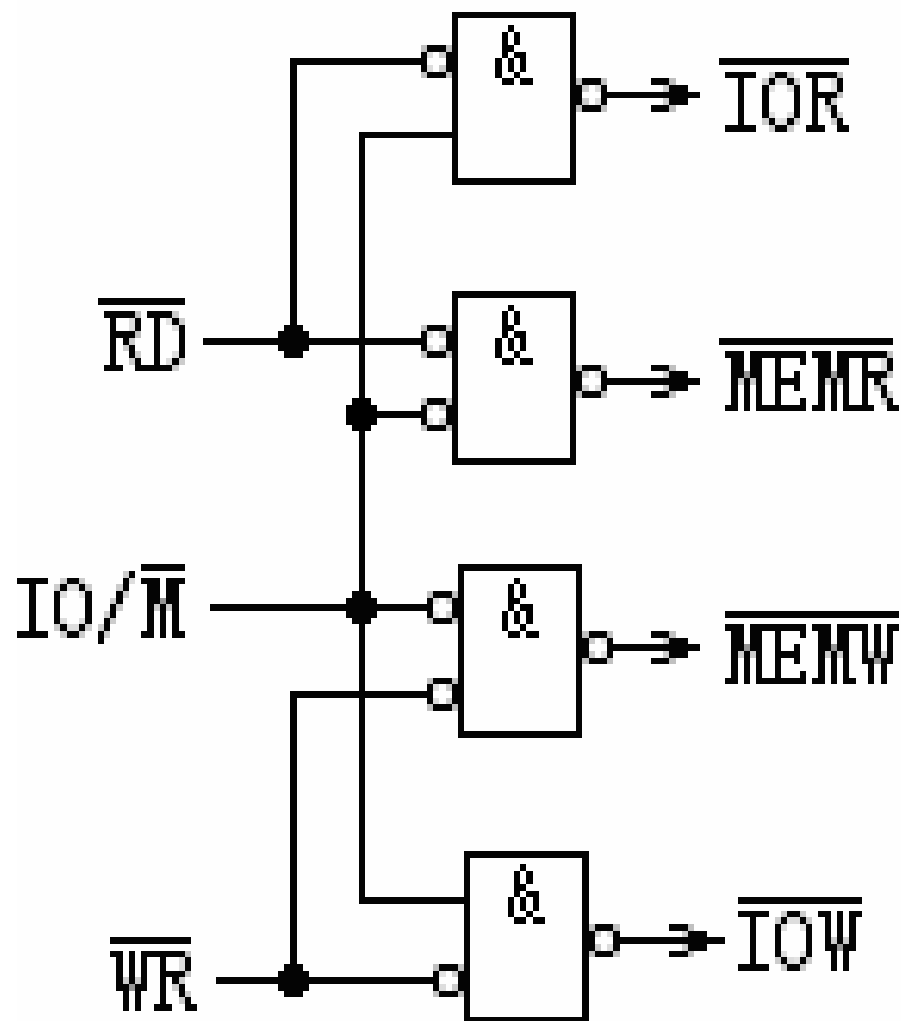
最小组态总线形成 (Intel产品手册推荐电路)



7) 系统控制信号的形成

8088的控制线不需驱动可以直接用作系统控制总线。

IO/M和RD、WR这3个信号经过如图所示的组合才能得到存储器读信MEMR，存储器写信MEMW、I/O读信号IOR和I/O写信号IOW。



8) 8284时钟发生器

8088内部没有时钟发生器，需要外接时钟发生器**8284**。

8284由三部分组成：

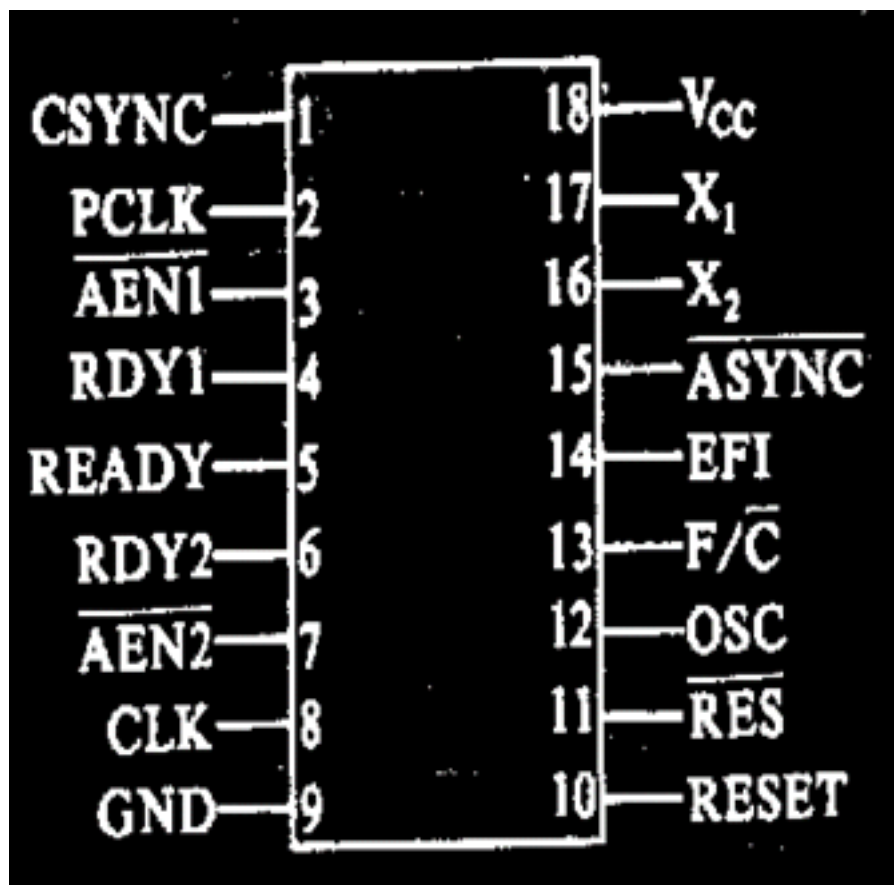
时钟发生电路

复位电路

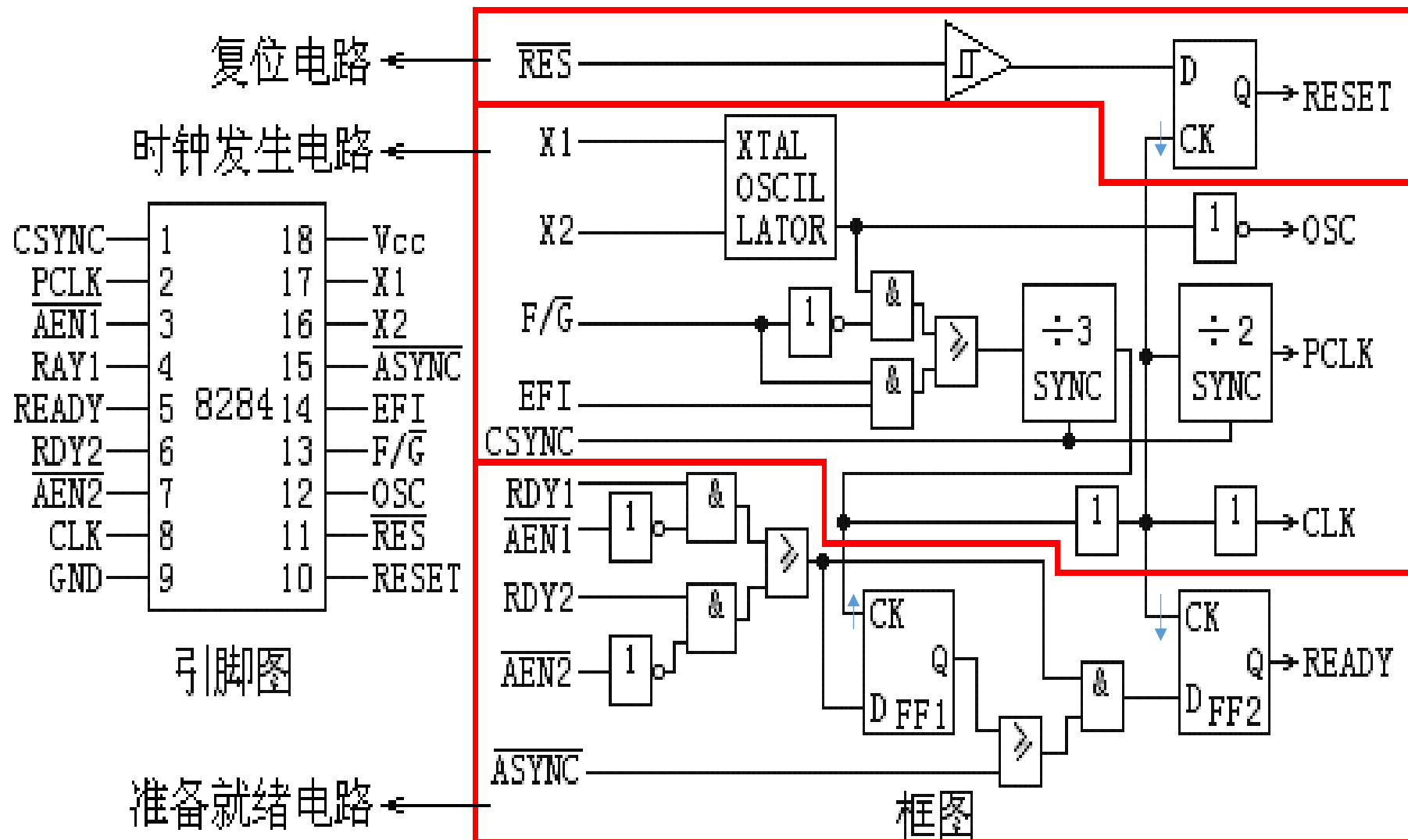
准备就绪电路

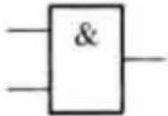
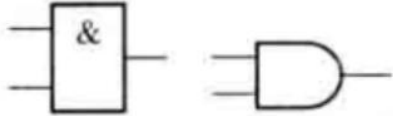
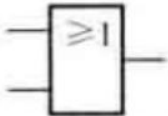
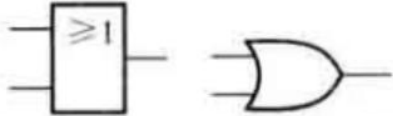
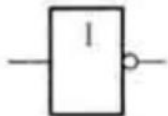
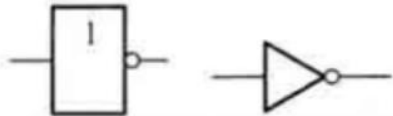
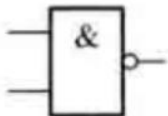
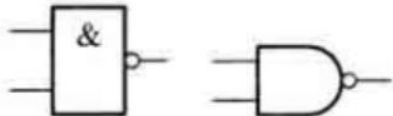
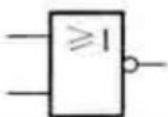
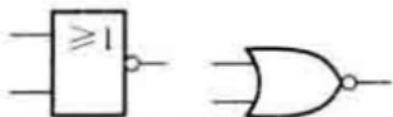
其内部电路的框图和引脚图如图所示。

8) 8284外部引脚图



8) 8284外部引脚与内部结构



名 称	国标符号	IEEE 标准符号
与门		
或门		
非门		
与非门		
或非门		

施密特触发器



8284的功能

1. 时钟信号产生（时钟发生电路）

- 内部震荡器X1,X2，外接晶体即可震荡。
- 外时钟输入端EFI。
- 时钟选择端F/G*，输入，为高选择外时钟，为低选择内时钟
- OSC: 输出，内部时钟同频信号。 提供给IBM显卡使用。
- PCLK: 输出，输入时钟6分频信号，占空比1/2。和8253一起，用于协助操作系统产生多任务、多线程、多进程的机制。
- CLK: 输出，输入时钟3分频信号，占空比为1/3。8086（88）时钟
- CSYNC: 输入，外部时钟的同步信号。使用内部时钟时，CSYNC接地

时钟信号产生

(1) 8284内部有一个石英晶体振荡器，只需在石英晶体连接端X1、X2两端外接石英晶体（晶振频率为14.31818MHz）即可。

也可由外振源输入端EFI输入一个TTL电平的振荡信号为时钟源。由外振源/晶体端F/C*来控制上述的两种选择。为高电平时，EFI端有效。

时钟信号产生

(2) 时钟同步输入端CSYNC是为多个8284的时钟同步而设置的。

在多个8284同时工作时，如果要求同相位的时钟信号，则把这些8284的EFI端接到同一个外振源，并用CSYNC信号来控制它们同步工作。当CSYNC为高电平时，8284的内部计数器复位；CSYNC变为低电平时，8284才开始工作。

使用晶体时，CSYNC应接地。

时钟信号产生

- (3) 振荡信号经3分频后由CLK输出一个占空比为1/3的MOS时钟信号。CLK信号再经二分频为供外部设备使用的外部时钟PCLK，这是一个占空比为1/2的TTL电平信号。

时钟信号产生

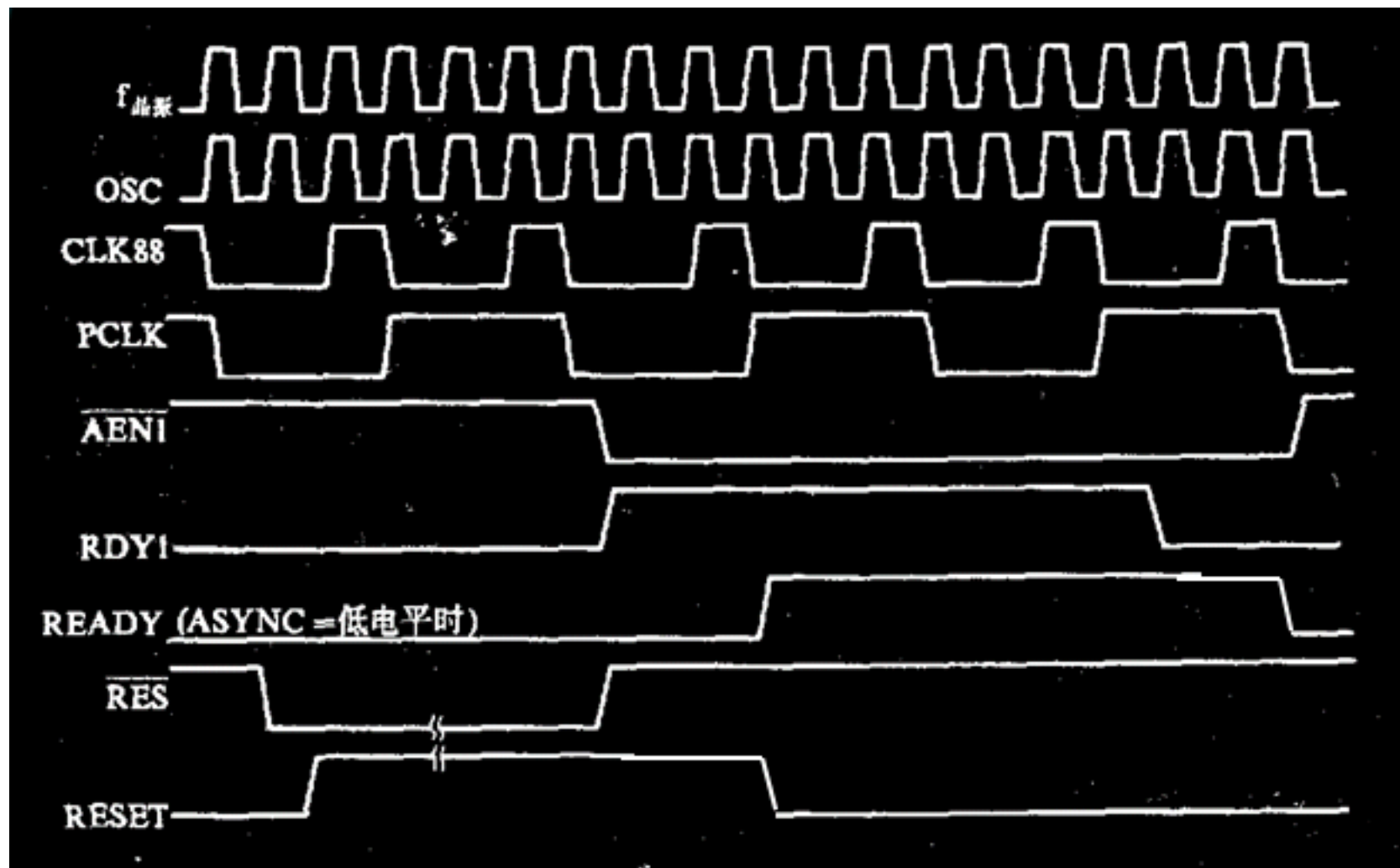
(4) PC XT微机只使用一片8284，外接14.31818MHz的晶体（这是IBM彩色图形卡上必须使用的频率），输出三种频率信号：

OSC端输出14.32MHz的振荡信号：供**显示器**使用；

CLK端输出4.77MHz的时钟信号：提供给**CPU和系统**；

PCLK端输出2.38MHz的外部时钟信号：给**8253定时/计数器**提供时钟输入。

8284 工作波形图



8284的功能

2. 复位逻辑

- 输入RES经斯密特触发器整形后，由系统时钟同步产生RESET信号，给计算机系统复位。

复位电路由一个**施密特触发器**和一个**同步触发器**组成。复位输入信号RES经过施密特触发器**整形**，在时钟脉冲**上升沿**送入同步触发器，产生系统复位信号RESET。

由于在同步触发器D端接有一个施密特触发器，故对复位信号要求不严格，由简单的RC放电回路即可生成。

8284的功能

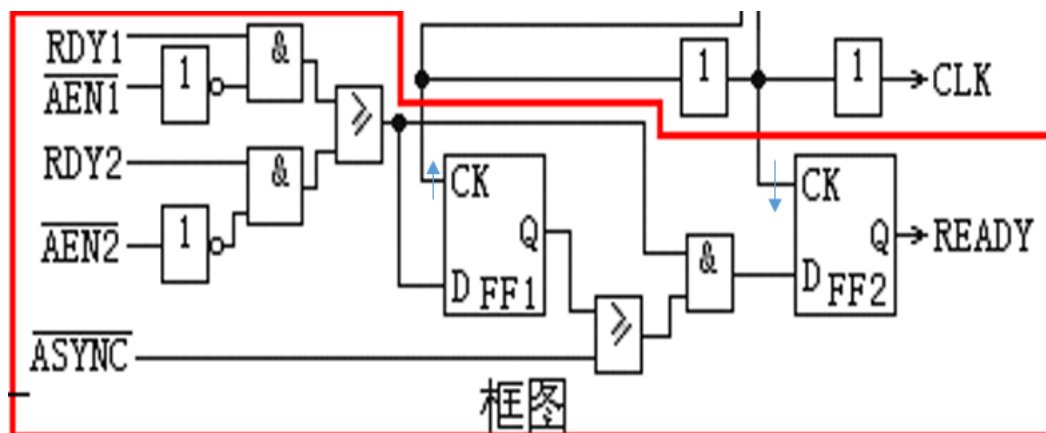
3. 准备好控制逻辑

- 准备好控制电路由有两组控制信号,
- $\text{RDY1} \bullet \overline{\text{AEN1}}$ 和 $\text{RDY2} \bullet \overline{\text{AEN2}}$

(1) **准备就绪电路**由两个D触发器和一些门电路组成。准备就绪信号 RDY_1 、 RDY_2 分别由对应的地址允许信号 AEN_1 、 AEN_2 来进行控制。

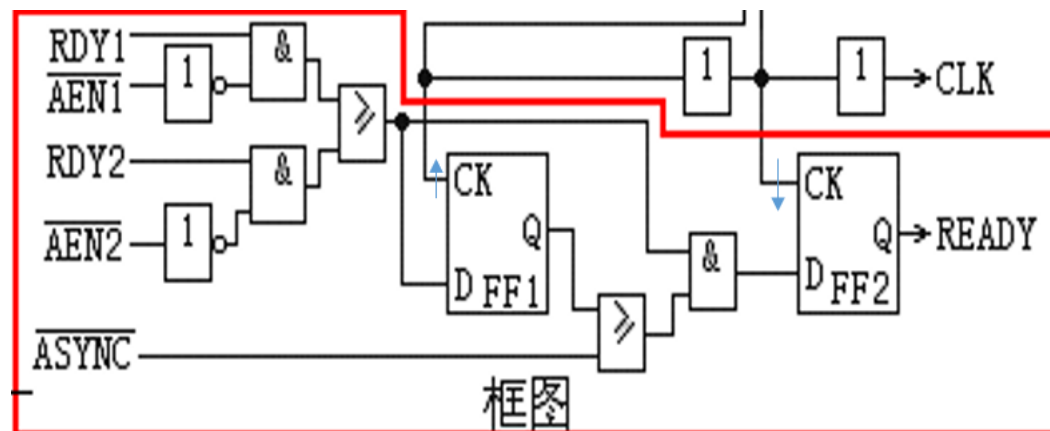
准备好控制逻辑

- **AEN**称为地址允许（高电平时说明总线上正在实施DMA操作），它在8284的内部经反相后和RDY一起作为与门的输入端。
- 若系统不提供**DMA**功能，**AEN**和**RDY**可以同时连接存储器或接口芯片提供的RDY信号（AEN需取反再输入）。
- 两组输入可仅使用一组，不用组的**RDY**接地，**AEN**接**Vcc**。



准备好控制逻辑

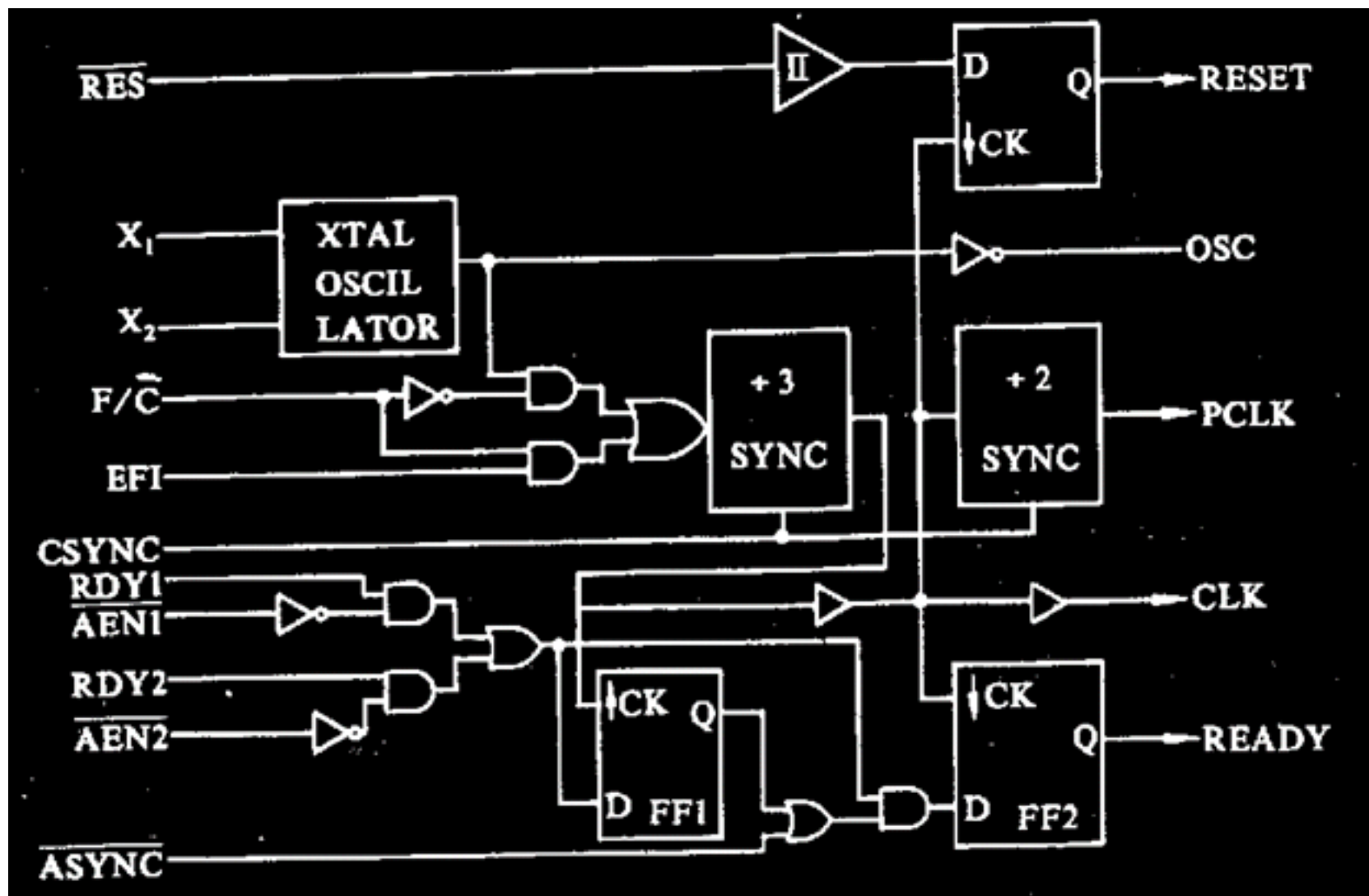
(2) 当准备就绪输入信号已和时钟同步时，可只使用一级同步方式，同步方式选择输入端ASYNC接高电平；否则应选用二级同步方式，ASYNC接低电平，ASYNC为低多插入一个时钟周期。



准备好控制逻辑

一级同步方式是在ASYNC接高电平时，将准备就绪输入信号直接在CLK的下降沿同步到FF2，处于这种方式时，要求外设能够提供建立时间要求的RDY信号。

8284内部结构图

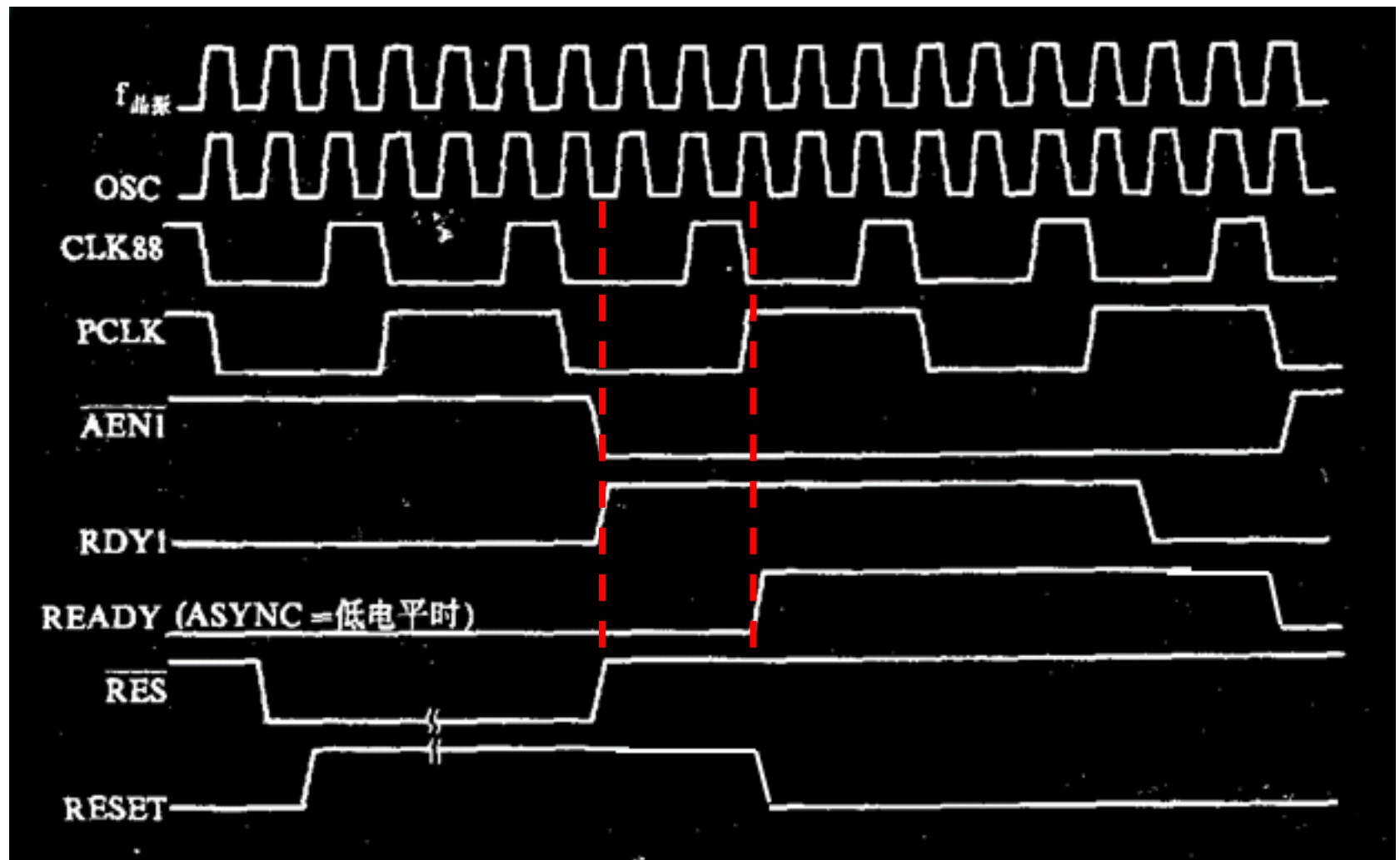


准备好控制逻辑

二级同步方式是在ASYNC接低电平时，准备就绪输入信号有效后，首先在CLK的上升沿同步到触发器1（FF1），然后在CLK下降沿同步到FF2，使READY有效（高电平）。

准备就绪输入信号无效时，将直接在CLK下降沿同步到FF2，使READY无效。

8284 工作波形图 例：RDY1和READY



（6）8088的基本总线时序

- **时序（Timing）**是指信号高低电平(有效或无效)变化及相互间的时间顺序关系。
- 时序主要由CPU中的控制器来控制，决定系统各部件间的同步和定时。
- **总线时序**描述CPU引脚如何实现总线操作。

（6）8088的基本总线时序

- 描述**总线操作**的微处理器时序有三级：
 - 指令周期 → 总线周期 → 时钟周期
- **指令周期**是指一条指令经取指、译码、读写操作数到执行完成的过程。若干总线周期组成一个指令周期
- **总线周期**是指CPU通过总线操作与外部（存储器或I/O端口）进行一次数据交换的过程

(6) 8088的基本总线时序

- 总线操作是指CPU通过总线对外的各种操作
- 8088的总线操作主要有：
 - 存储器读、I/O读操作
 - 存储器写、I/O写操作
 - 中断响应操作
 - 总线请求及响应操作
 - CPU正在进行内部操作、并不进行实际对外操作的空闲状态Ti

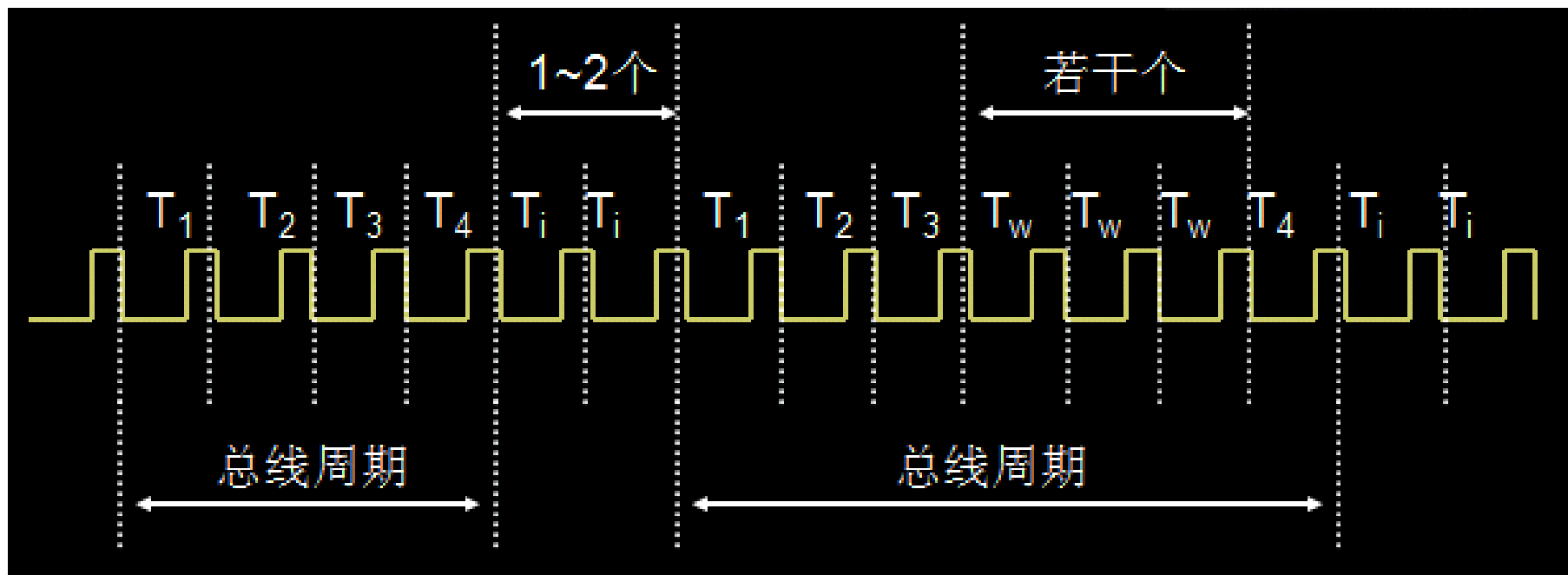
(6) 8088的基本总线时序

- 任何指令的**取指**阶段都需要存储器读总线周期，读取的内容是指令代码
- 任何一条以存储单元为源操作数的指令都将引起**存储器读**总线周期，任何一条以存储单元为目的操作数的指令都将引起**存储器写**总线周期
- 只有执行IN指令才出现**I/O读**总线周期，执行OUT指令才出现**I/O写**总线周期
- CPU响应可屏蔽中断时生成**中断响应**总线周期

(6) 8088的基本总线时序

- 8088的基本总线周期需要4个时钟周期
 - 4个时钟周期编号为 T_1 、 T_2 、 T_3 和 T_4
 - 总线周期中的时钟周期也被称作“T状态”
 - 时钟周期的时间长度就是时钟频率的倒数
- 当需要延长总线周期时需要插入等待状态 T_w

总线周期的构成



(6) 8088的基本总线时序

- 总线操作中如何实现CPU与内存、接口芯片间的时序同步是关键
- 同步时序：
 - 各部件都以系统时钟信号为基准
 - 当相互不能配合时，快速部件（CPU）插入等待状态等待慢速部件（I/O和存储器）
- 异步时序：

CPU与外设接口通过应答联络信号实现同步操作

(7) 8088最小组态的总线时序

本节展开微处理器最基本的4种总线周期

存储器读总线周期

存储器写总线周期

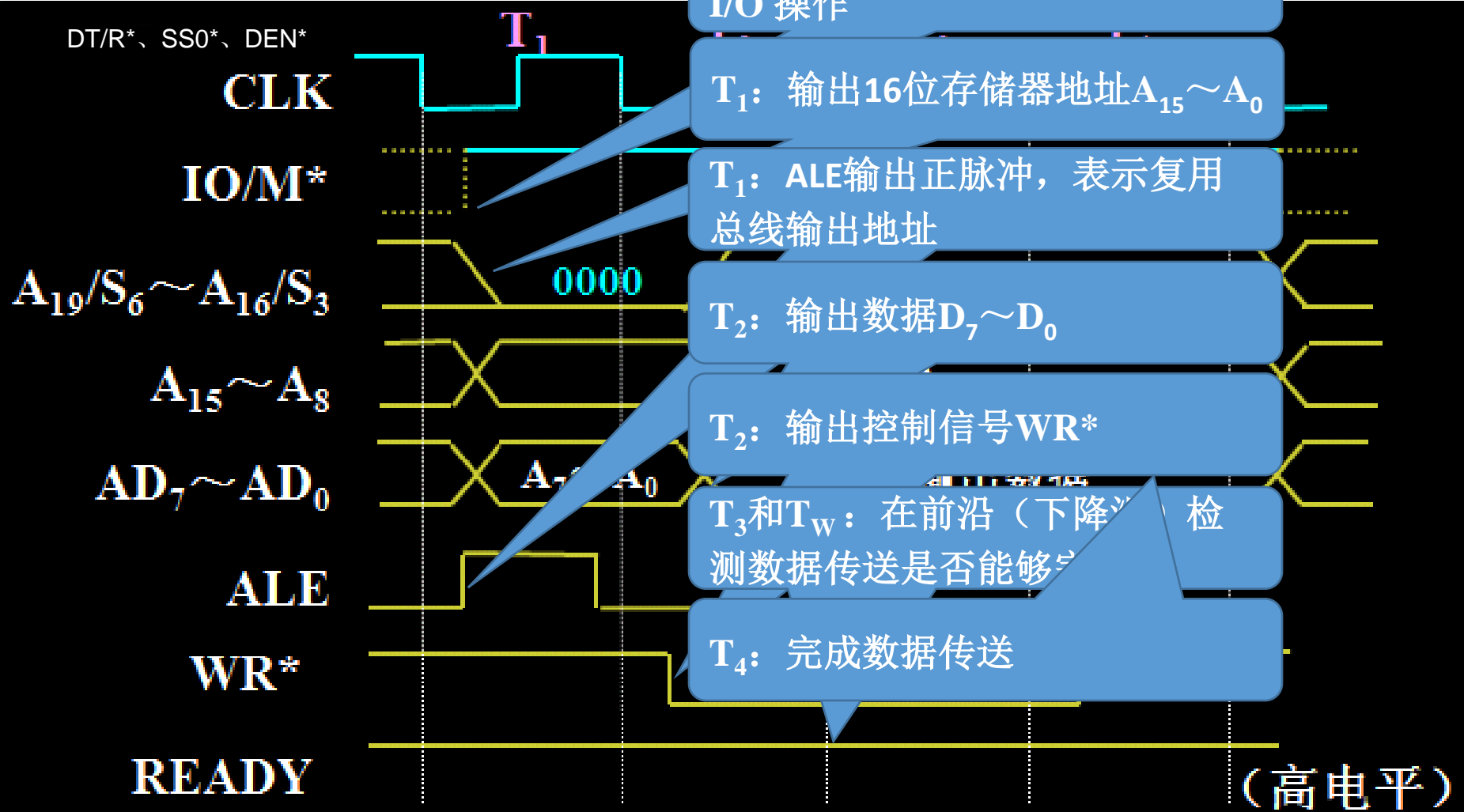
I/O读总线周期

I/O写总线周期

存储器写总线周期



I/O写总线周期



存储器读总线周期

DT/R*, SS0*, DEN*

CLK

IO/M*

A₁₉/S₆~A₁₆/S₃

A₁₅~A₈

AD₇~AD₀

ALE

RD*

READY

T₁

T₁: IO/M*输出低电平, 表示存储器操作

T₁: 输出20位存储器地址A₁₉~A₀

T₁: ALE输出正脉冲, 表示复用总线输出地址

T₂: 输出控制信号RD*

T₃和T_w: 在前沿(下降沿)检测数据传送是否能够

T₄: 前沿读取数据, 完成数据传送

(高电平)

The diagram illustrates the timing sequence for an I/O read operation. The signals shown are:

- DT/R***, **SS0***, **DEN***: Control signals, all high.
- CLK**: Clock signal, with a period T_1 indicated.
- IO/M***: I/O/Master/Slave select signal, active low pulse.
- A₁₉/S₆~A₁₆/S₃**: Address/data bus, showing address **0000** and status signals **S₆~S₃**.
- A₁₅~A₈**: Address/data bus, showing address **A₁₅~A₈**.
- AD₇~AD₀**: Address/data bus, showing address **A₇~A₀**.
- ALE**: Address Latch Enable, active low pulse.
- RD***: Read strobe, active low pulse.
- READY**: Ready signal, active low, shown as high.

Key timing points and operations:

- T₁**: Output 16-bit storage address **A₁₅~A₀**. ALE output positive pulse, indicating bus output address.
- T₂**: Output control signal **RD***.
- T₃ and T_w**: On the rising edge (falling edge) detection, data transmission is completed.
- T₄**: Rising edge reads data, completing data transmission.

(High level)

T₄: 前沿读取数据，完成数据传送

(高电平)

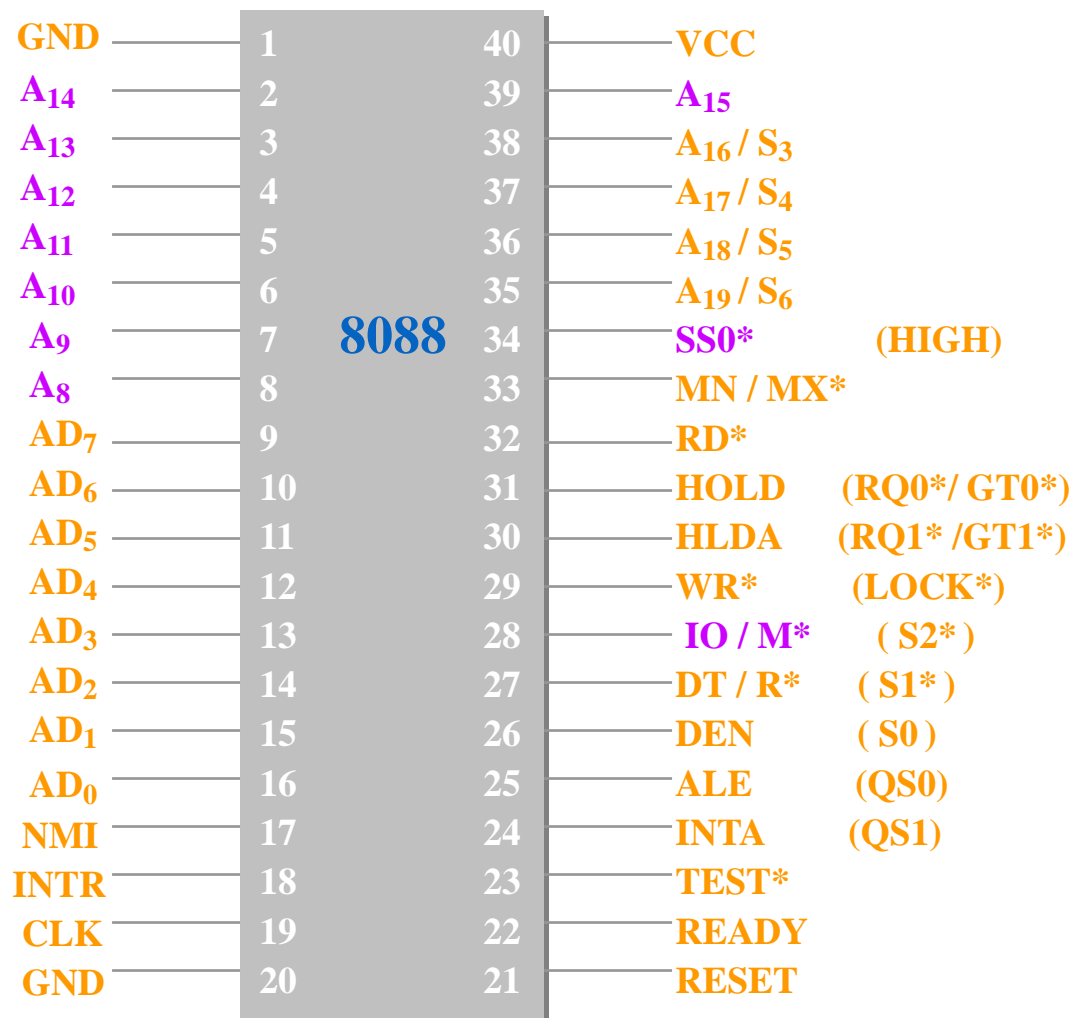
等待状态 T_w

- 同步时序通过插入等待状态，来使速度差别较大的两部分保持同步
- 在读写总线周期中，判断是否插入 T_w
 1. 在T3的前沿检测READY引脚是否有效
 2. 如果READY无效，在T3和T4之间插入一个等效于T3的 T_w ，转1
 3. 如果READY有效，执行完该T状态，进入T4状态

(8) 8088最大组态中的引脚定义

- 8088的数据/地址等引脚在最大组态与最小组态时相同
- 有些控制信号不相同，主要是用于输出操作编码信号，由总线控制器8288译码产生系统控制信号：
 - S2*、S1*、S0*——3个状态信号
 - LOCK*——总线封锁信号
 - QS1、QS0——指令队列状态信号
 - RQ*/GT0*、RQ*/GT1*——2个总线请求/同意信号

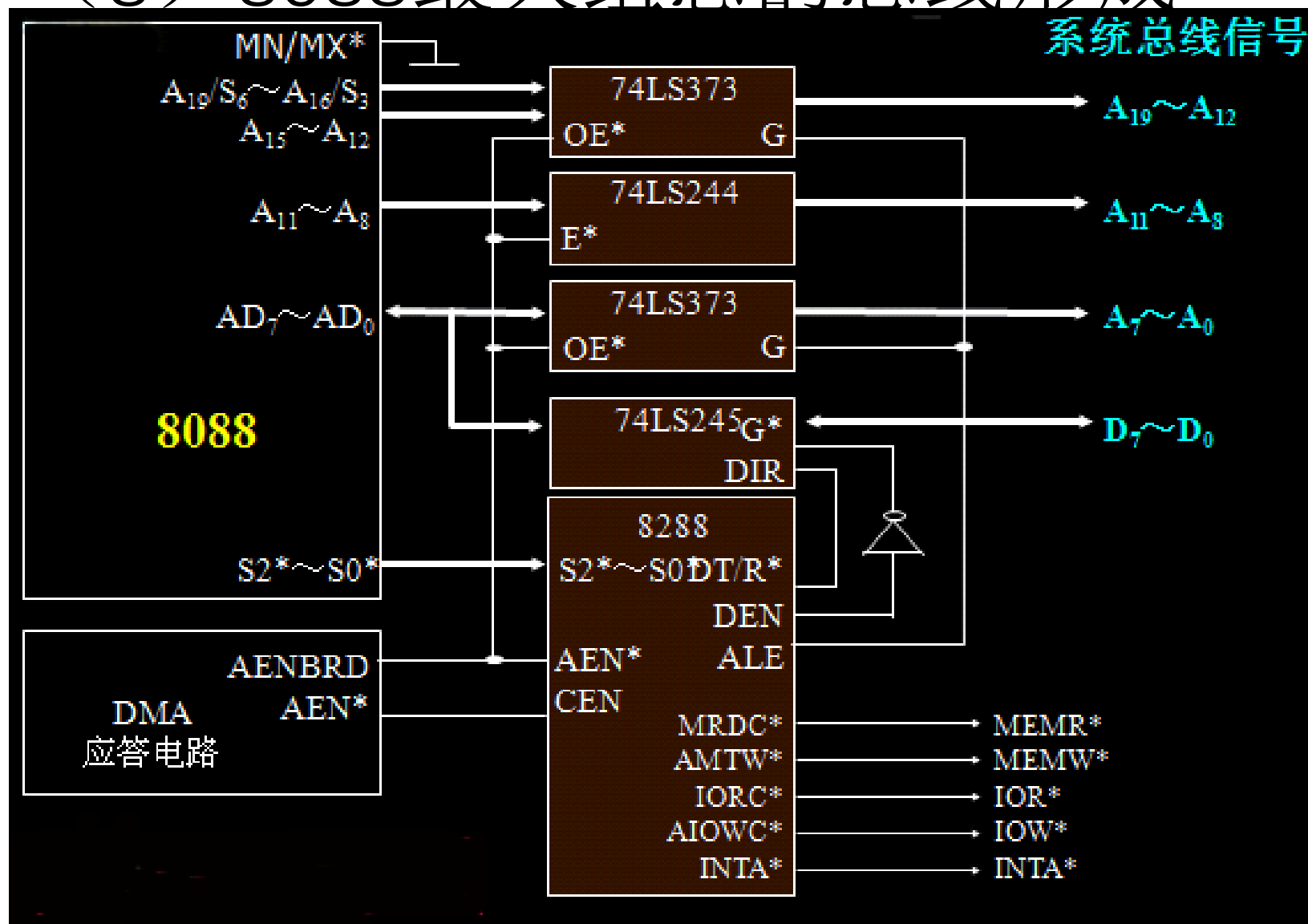
8088的引脚图



(8) 8088最大组态中的引脚定义

S2*	S1*	S0*	CPU的工作状态
0	0	0	中断响应
0	0	1	I/O 读
0	1	0	I/O 写
0	1	1	暂停
1	0	0	取指
1	0	1	存储器读
1	1	0	存储器写
1	1	1	过渡状态

(8) 8088最大组态的总线形成



(9) 8088最大组态的总线形成

(1) 系统地址总线

采用三态透明锁存器**74LS373**和三态单向缓冲器**74LS244**

(2) 系统数据总线

通过三态双向缓冲器**74LS245**形成和驱动

(3) 系统控制总线

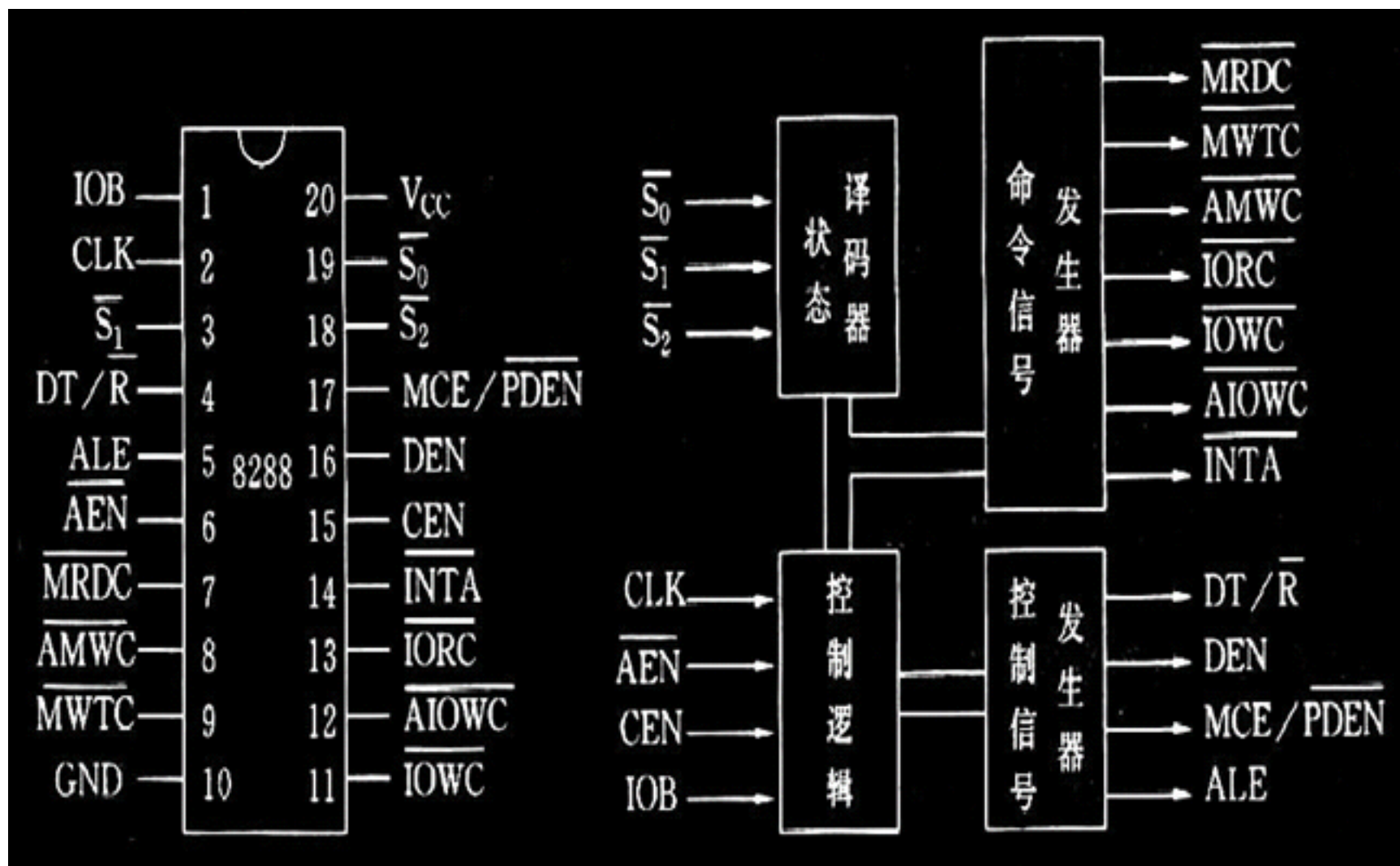
主要由总线控制器**8288**形成

MEMR*、**MEMW***、**IOR***、**IOW***、**INTA***

最大组态控制总线形成

S2	S1	S0	8086总线周期	8288命令输出
0	0	0	中断响应	INTA
0	0	1	读I/O口	IORC
0	1	0	写I/O口	IOWC,AIOWC
0	1	1	暂停	
1	0	0	取指令代码	MRDC
1	0	1	读存储器	MRDC
1	1	0	写存储器	MWTC,AMWC
1	1	1	过渡状态	

8288总线控制器内、外部结构



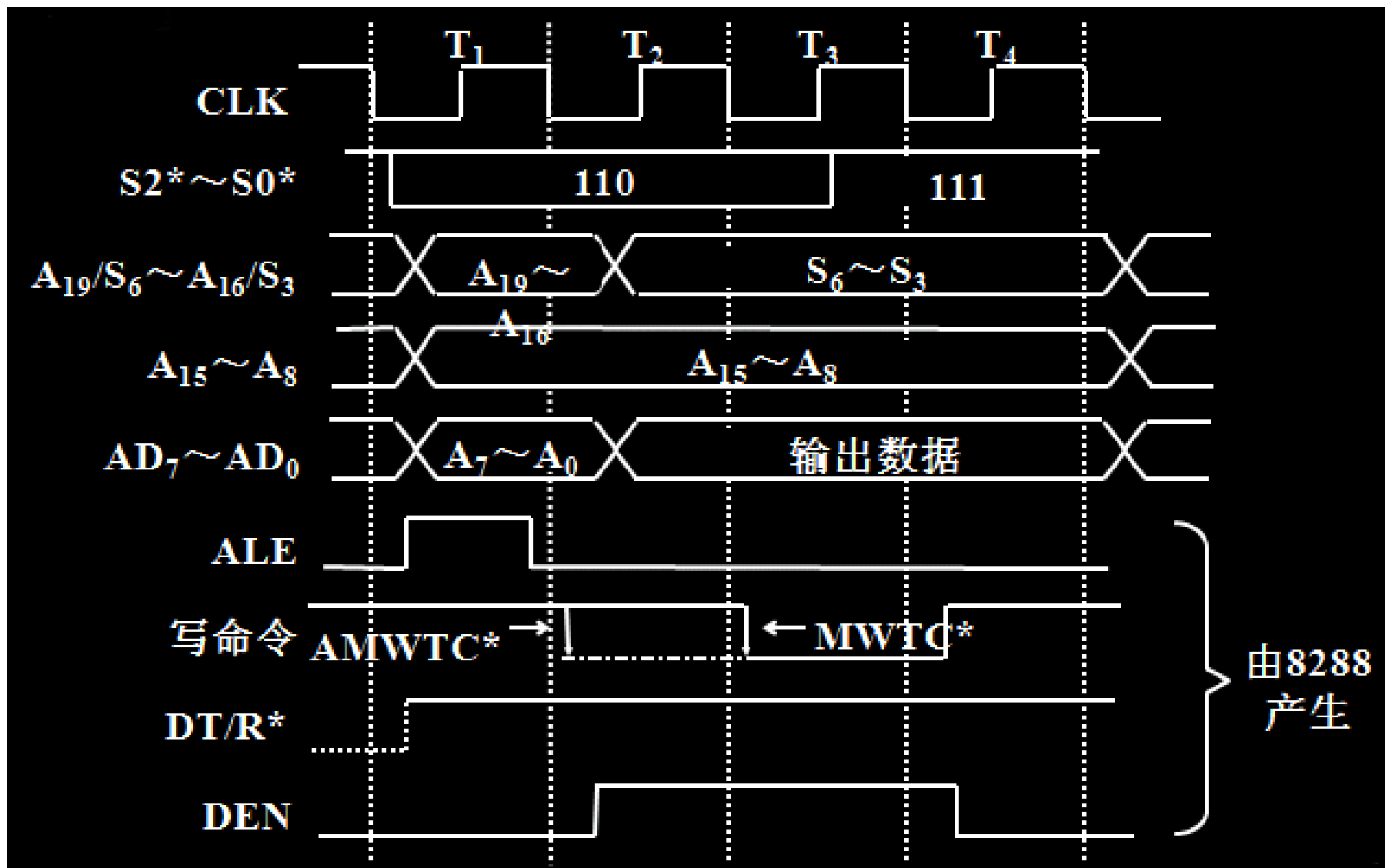
8288工作方式相关的引脚

- **CEN**片选信号，**CEN**为高电平允许**8288**工作，否则禁止工作
- **IOB**工作方式选择，**IOB**接地，**8288**为单处理器工作方式；**IOB**接+5V为多处理器工作方式。
- **AEN**芯片工作允许，低电平有效。多处理器系统中由总线仲裁器控制**8288**工作。

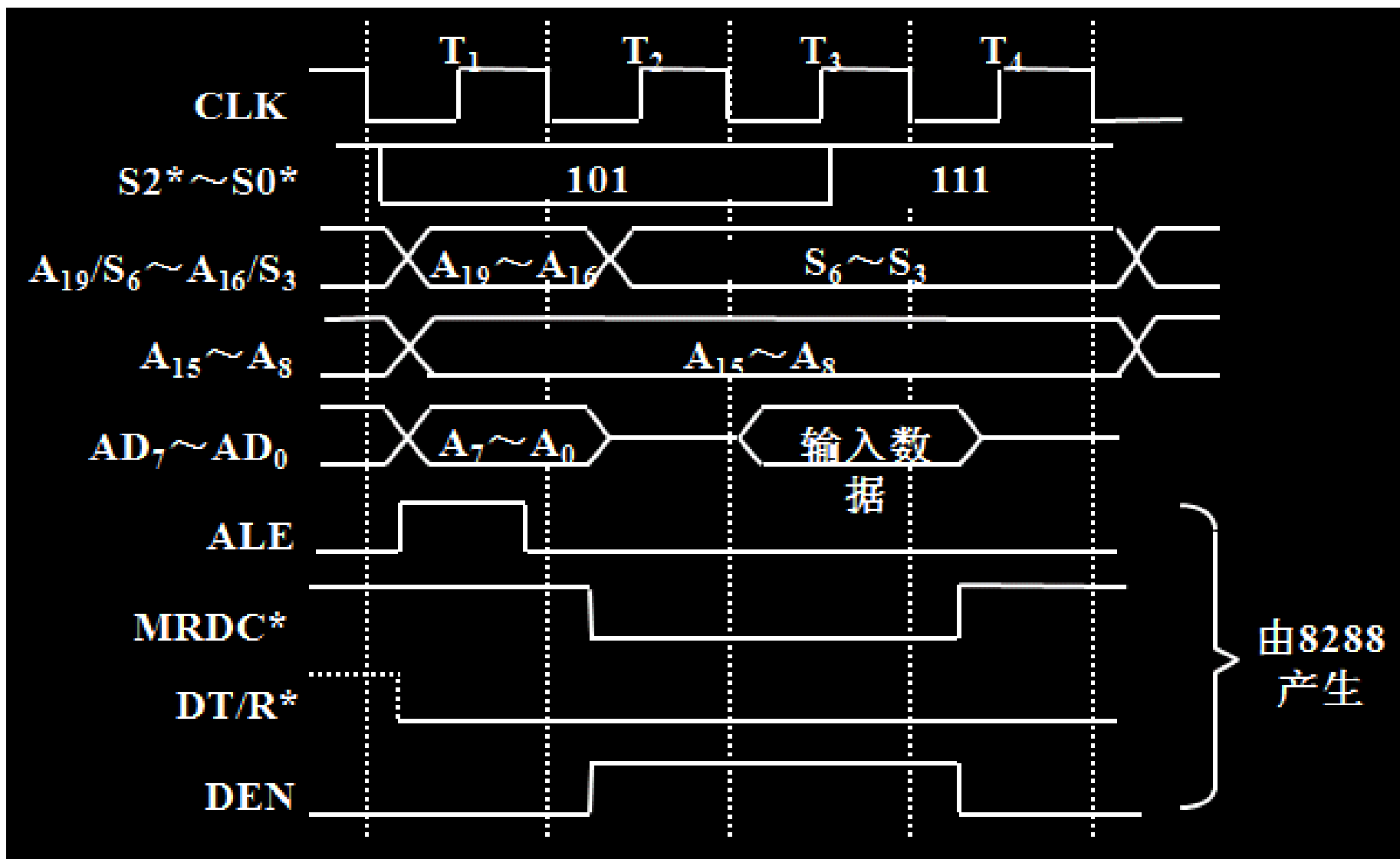
8288工作方式相关的引脚

- **MCE/PDEN**:总线主模块允许/外设数据允许。
- 单处理器工作时为**MCE**。此信号控制主从方式工作芯片协调工作。
- 多处理器系统中为**PDEN**。此信号作为数据总线收发器开启信号

8088最大组态的写总线时序（内存）



8088最大组态的读总线时序（内存）



8088最大组态下I/O读写时序

- 基本与内存读写时序相同，区别在于状态信号的差异。

（10）微机系统总线

- 微机系统采用**总线结构**。系统中主要部件通过系统总线相互连接、实现数据传输，并使微机系统具有**组态灵活、易于扩展**等诸多优点。
- 广泛应用的总线都实现了**标准化**，便于互连各个部件时遵循共同的总线规范。接口的任一方只需要根据总线标准的要求来实现和完成接口的功能，而不必了解对方的接口方式。
- **总线接口**也是一种通用的接口技术。

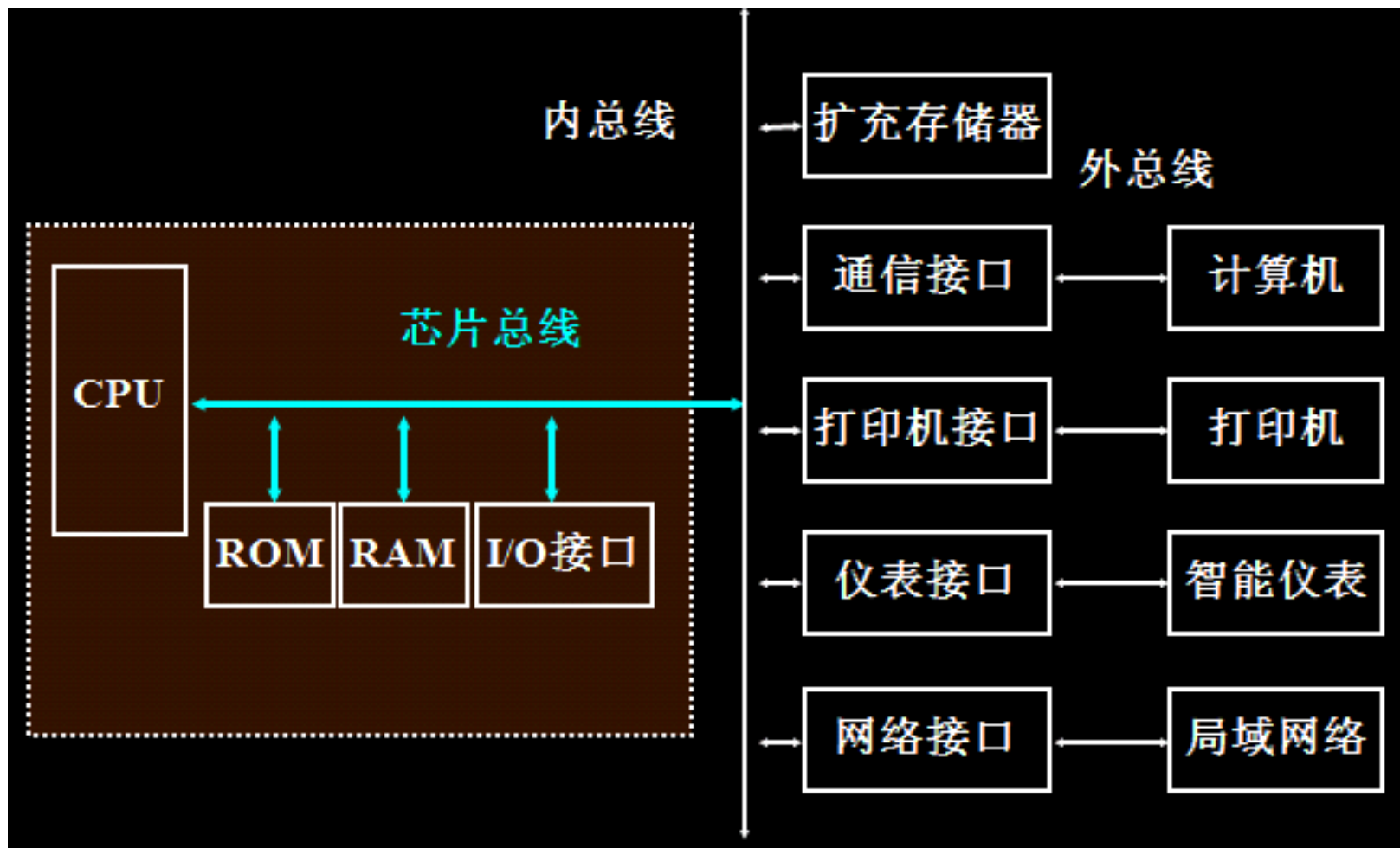
(10) 微机系统总线

- 总线连接方法广泛用于微机系统的各个连接层次上
 - 大规模集成电路芯片内部（如微处理器的内部总线）
 - 主机板中微处理器、存储器及I/O接口电路之间，主机模板与各种接口模板之间
 - 微机系统之间以及微机系统与外部设备之间

芯片总线（Chip Bus）

- 一个大规模集成电路芯片内部，或一个较小系统中各种不同器件连接在一起的总线；用于芯片级互连
- 芯片总线也称为局部总线（Local Bus）
- 微处理器的引脚信号就是芯片总线
- 微处理器内部的控制器、运算器、寄存器之间，还有系统主机板上CPU、存储器、接口电路等之间通常就是利用芯片级总线互连的

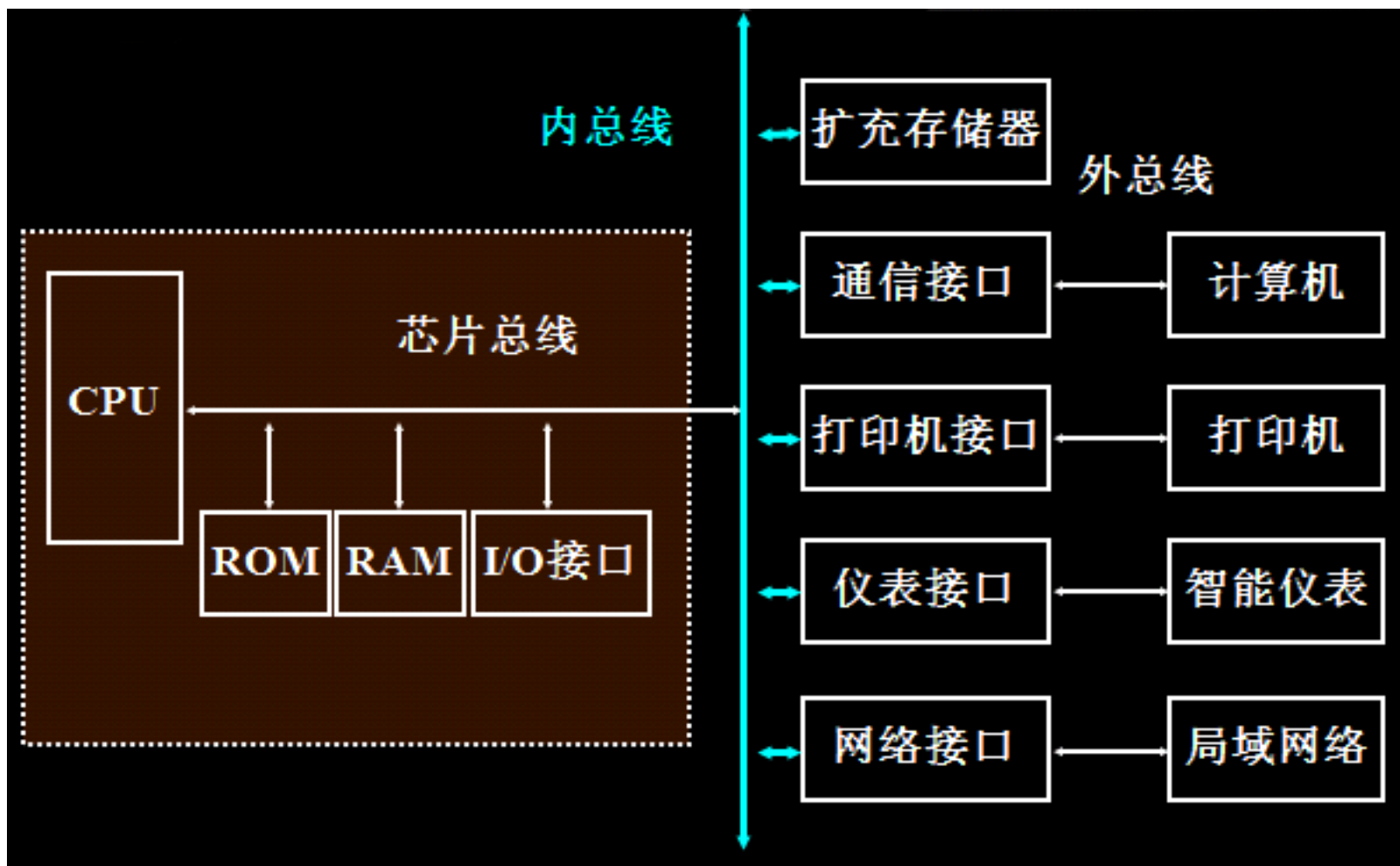
芯片总线 (Chip Bus)



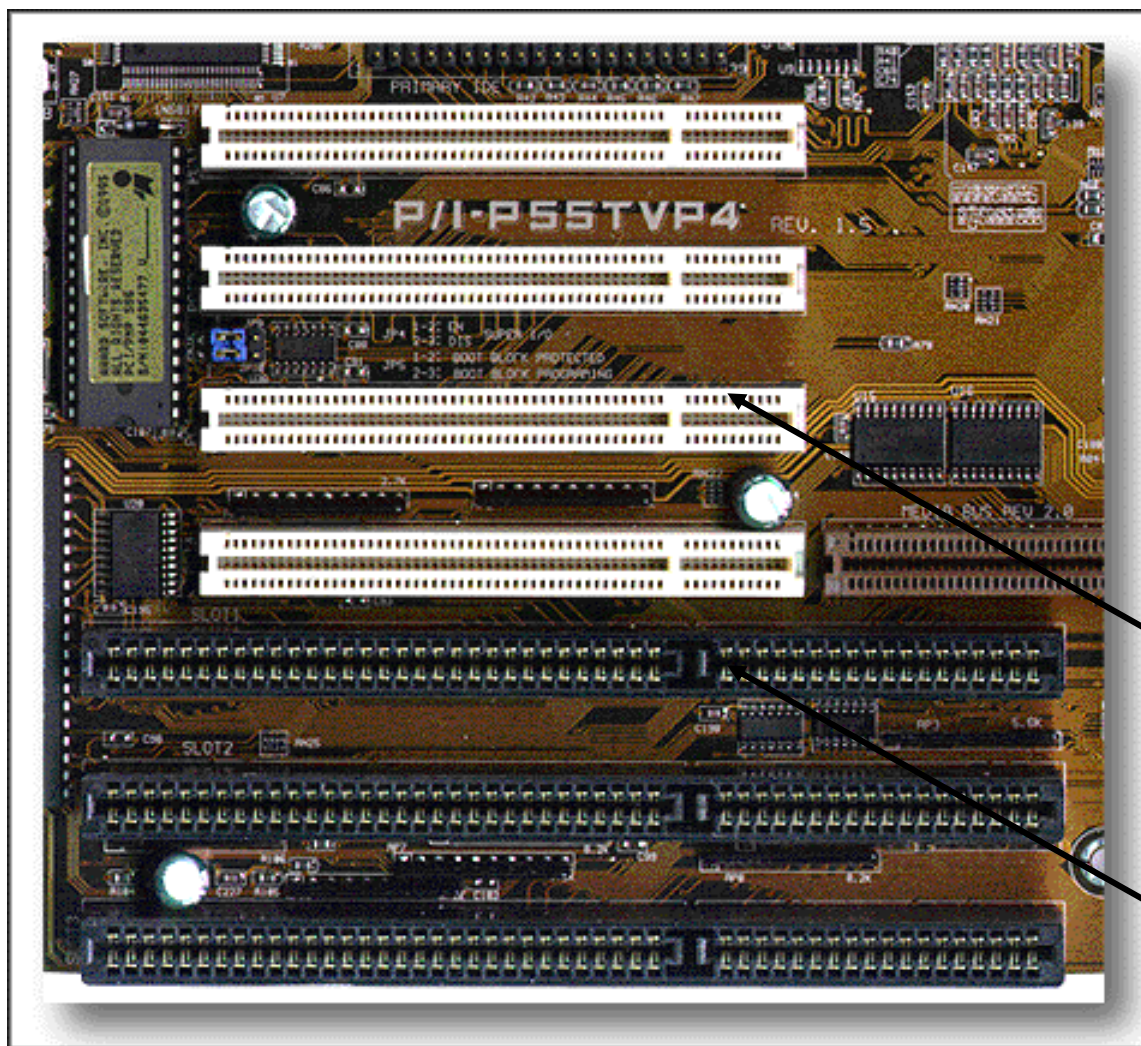
内总线（Internal Bus）

- 微机系统中**模板与模板间连接的总线**，是微机系统所特有的总线；用于模板级互连
- 内总线也被称为**板级总线或系统总线**（System Bus）
- 多数已实现标准化，例如**STD总线、ISA总线**等。
- 微机主板的各种扩展插槽多属于内总线

内总线（Internal Bus）



内总线（Internal Bus）



USB总线

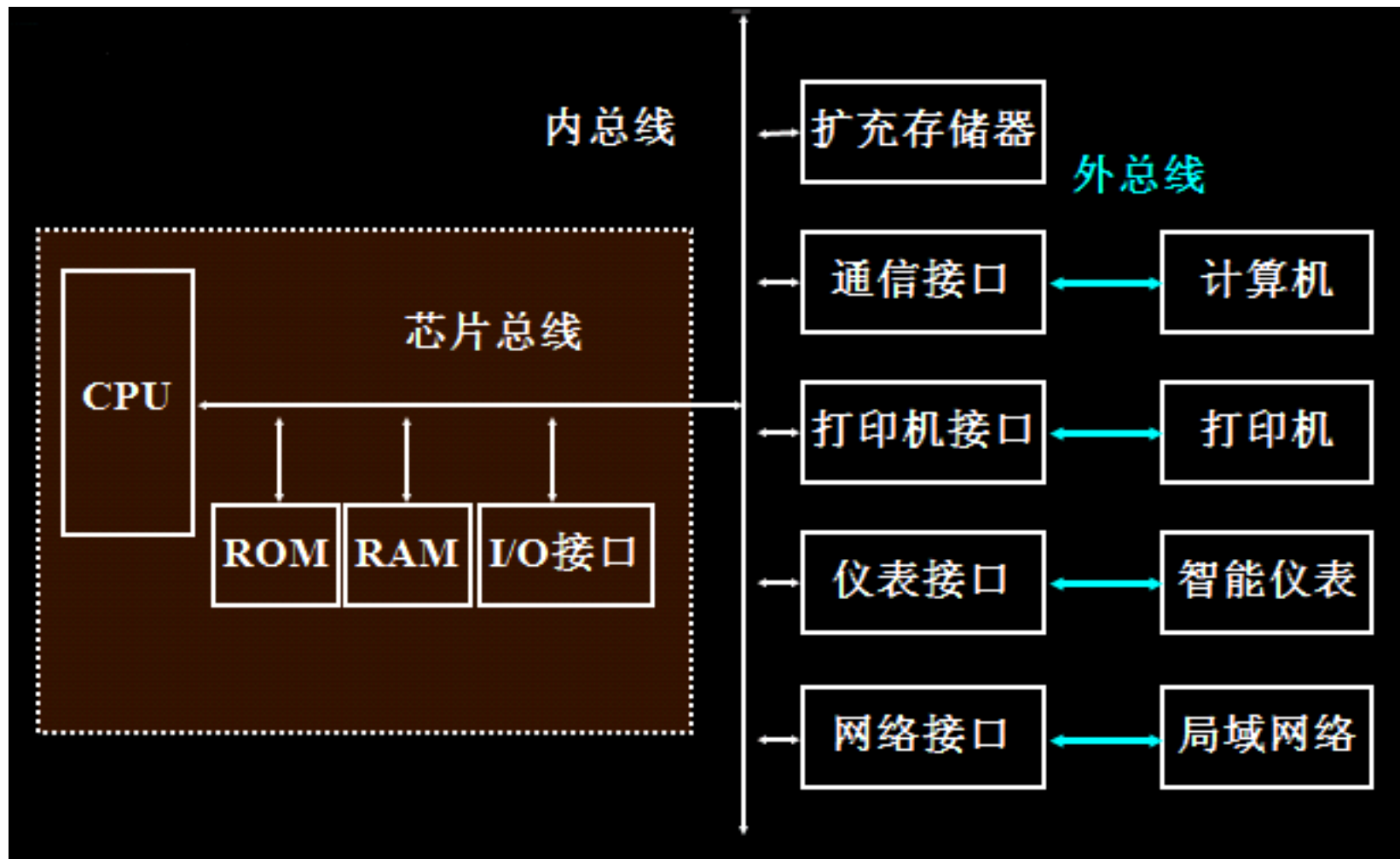
PCI总线

ISA总线

外总线（External Bus）

- 微机系统之间或微机系统与其外设通信的总线，用于设备级互连
- 外总线过去又称为通信总线，主要指串行通信总线，例如RS-232
- 现在，外总线的意义常延伸为外设总线，主要用于连接各种外设
- 外总线种类较多，常与特定设备有关，例如Centronics并行打印机总线、IEEE 488智能仪器仪表并行总线（又称为GPIB总线）

外总线（External Bus）



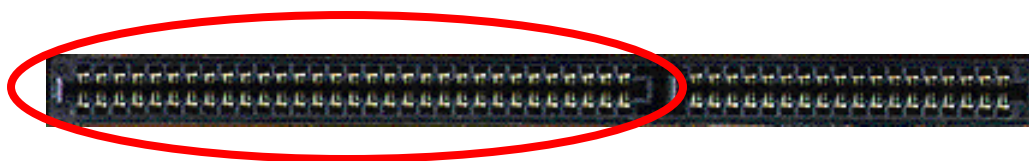
常见系统总线

- 在微机发展和应用中出现了许多种内、外总线标准
- 第一个标准化的微机总线S-100总线
- 面向工业控制领域的STD总线
- 32位PC机上的ISA系统总线、EISA总线、VESA总线、PCI总线、USB总线等

CPU组态与系统总线的区别与联系

- CPU的不同组态提供了不同的芯片总线形成方式
- 但CPU组态提供的总线信号并不一定被特定总线标准直接作为系统总线上的信号。
- 多数总线标准是在CPU引脚信号的基础上增添自定义的逻辑电路，生成符合自己标准的总线信号

IBM PC 总线（一种系统总线）



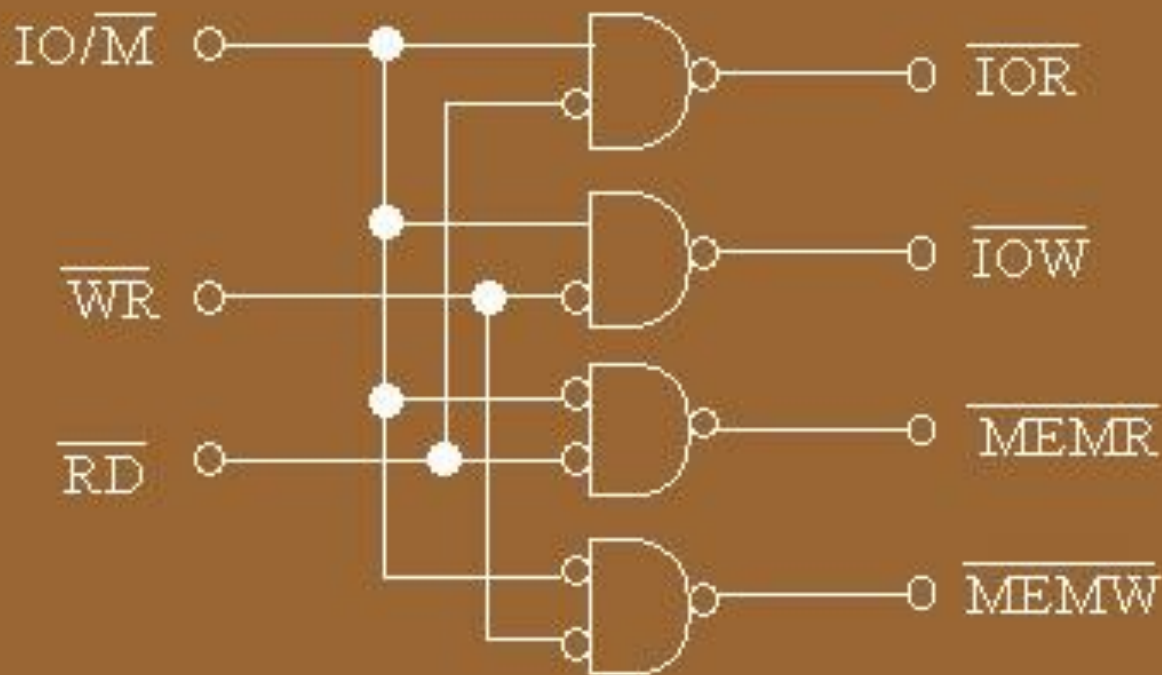
- IBM PC总线是IBM PC/XT机上使用的8位系统总线
- 有62条信号线，用双列插槽连接，分A面（元件面）和B面（焊接面）
- 实际上是8088 CPU核心电路总线的扩充和重新驱动
- 与最大组态下的8088总线相似

IBM PC 总线

- $D_0 \sim D_7$ ——8位双向数据线
- $A_0 \sim A_{19}$ ——20位输出地址线
- ALE——地址锁存允许，每个CPU 总线周期的T1 状态高电平有效
- MEMR*——存储器读，输出、低有效
- MEMW*——存储器写，输出、低有效
- IOR*——I/O读，输出、低有效
- IOW*——I/O写，输出、低有效
- I/O CH RDY——I/O通道准备好，输入、高有效

IBM PC 总线

最大、最小
组态兼容



总线读写控制信号的转换

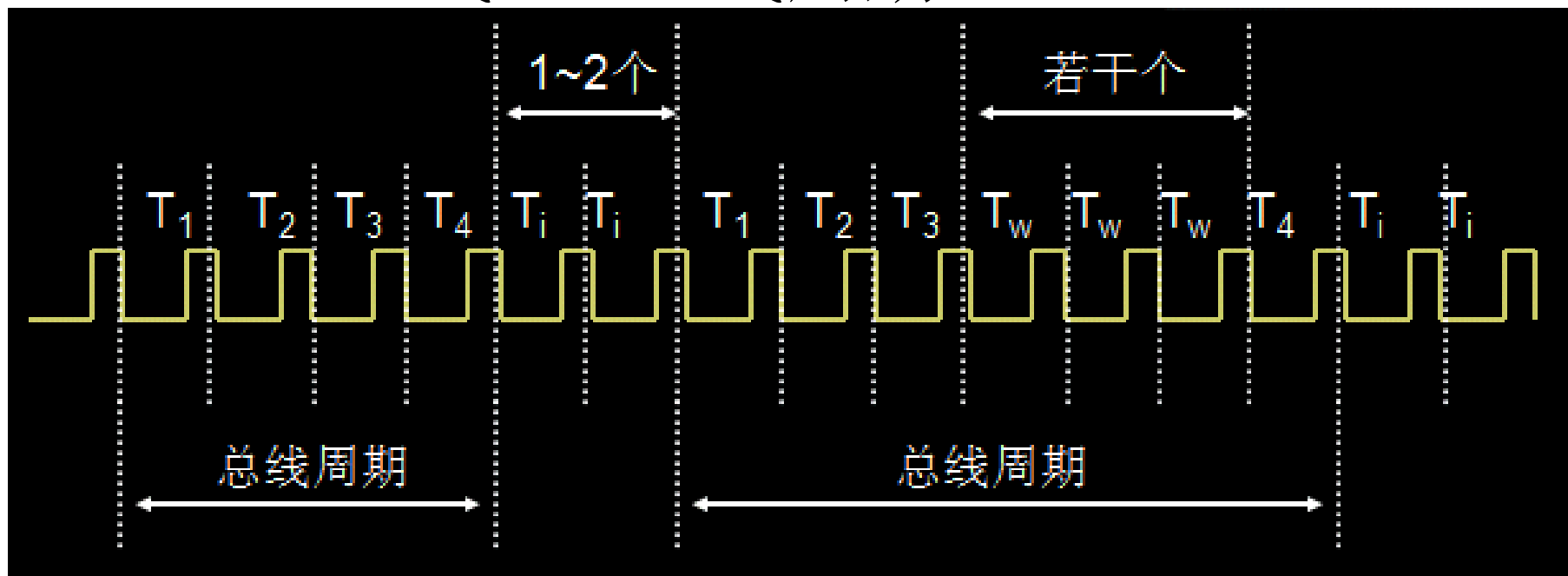
IBM PC 总线

- $IRQ_2 \sim IRQ_7$ ——中断请求信号，输入、高有效
- AEN——地址允许信号，输出、高有效，用于指示DMA总线周期
- $DRQ_0 \sim DRQ_3$ ——DMA请求信号，输入、高有效
- $DAK_0^* \sim DAK_3^*$ ——DMA响应信号，输出、低有效
- T/C——计数结束信号，输出、正脉冲有效

IBM PC 总线

- RESET——复位信号，输出、高有效
- IOCHCK*——I/O通道校验，输入、低有效
- OSC——晶振频率脉冲，输出14.31818MHz的主振频率信号
- CLK——系统时钟，输出4.77MHz的系统时钟信号
- +5V、-5V、+12V、-12V、GND——电源和地线

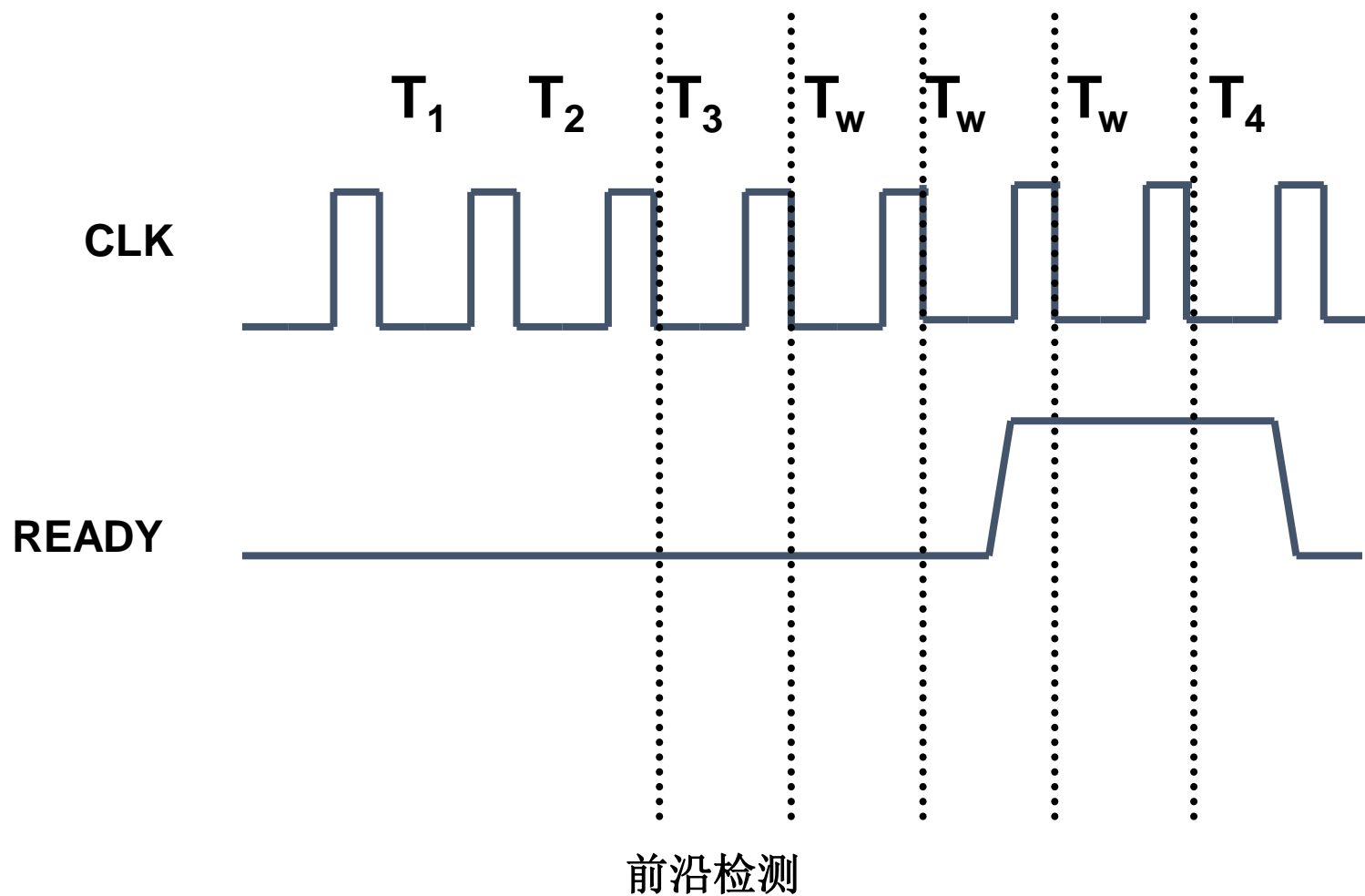
IBM PC 总线——总线周期



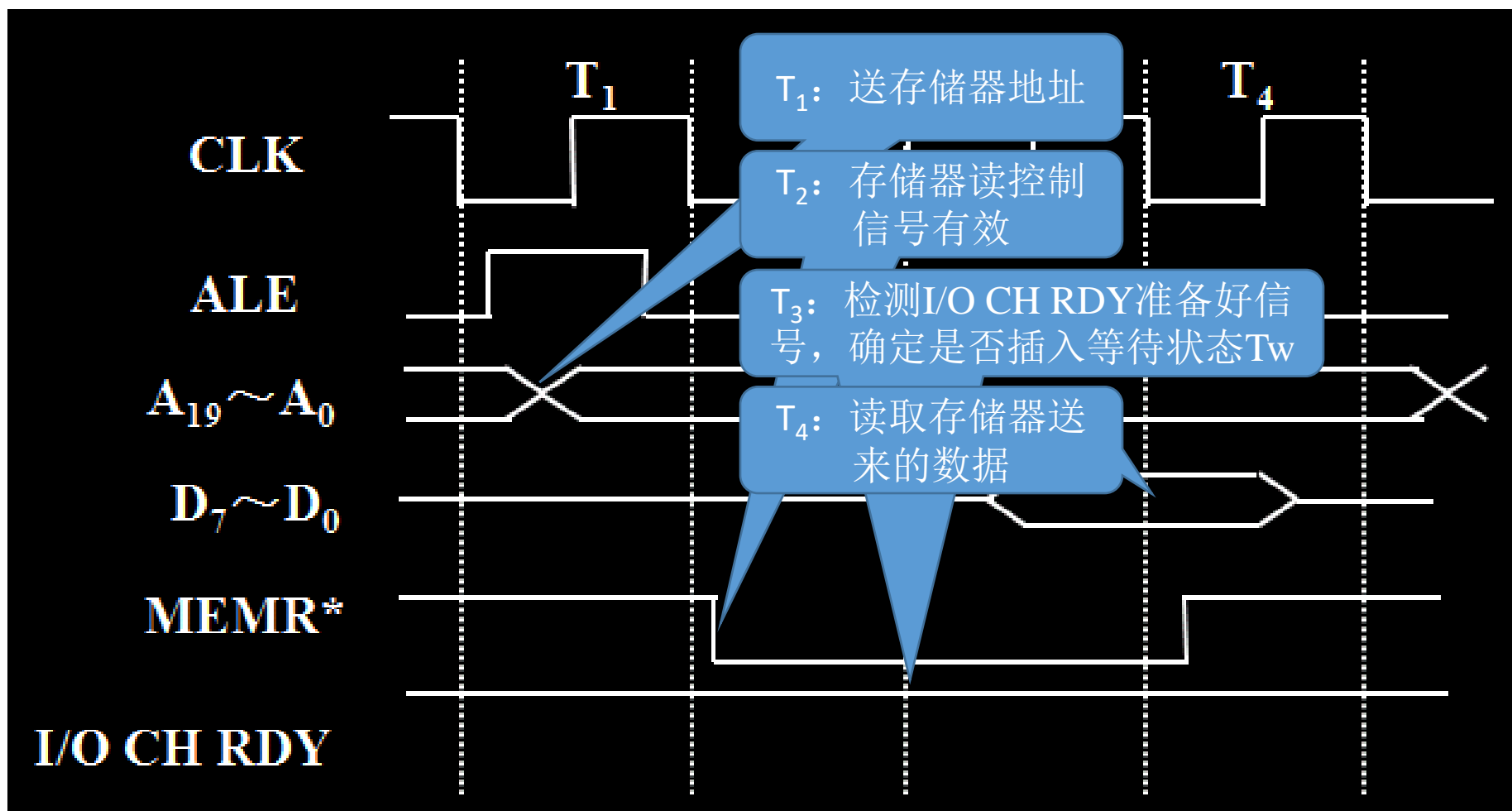
与8088CPU提供的总线周期一致：

基本总线周期由4个T状态组成： T_1 、 T_2 、 T_3 、 T_4
等待时钟周期 T_w ，在总线周期的 T_3 和 T_4 之间插入
空闲时钟周期 T_i ，在两个总线周期之间插入

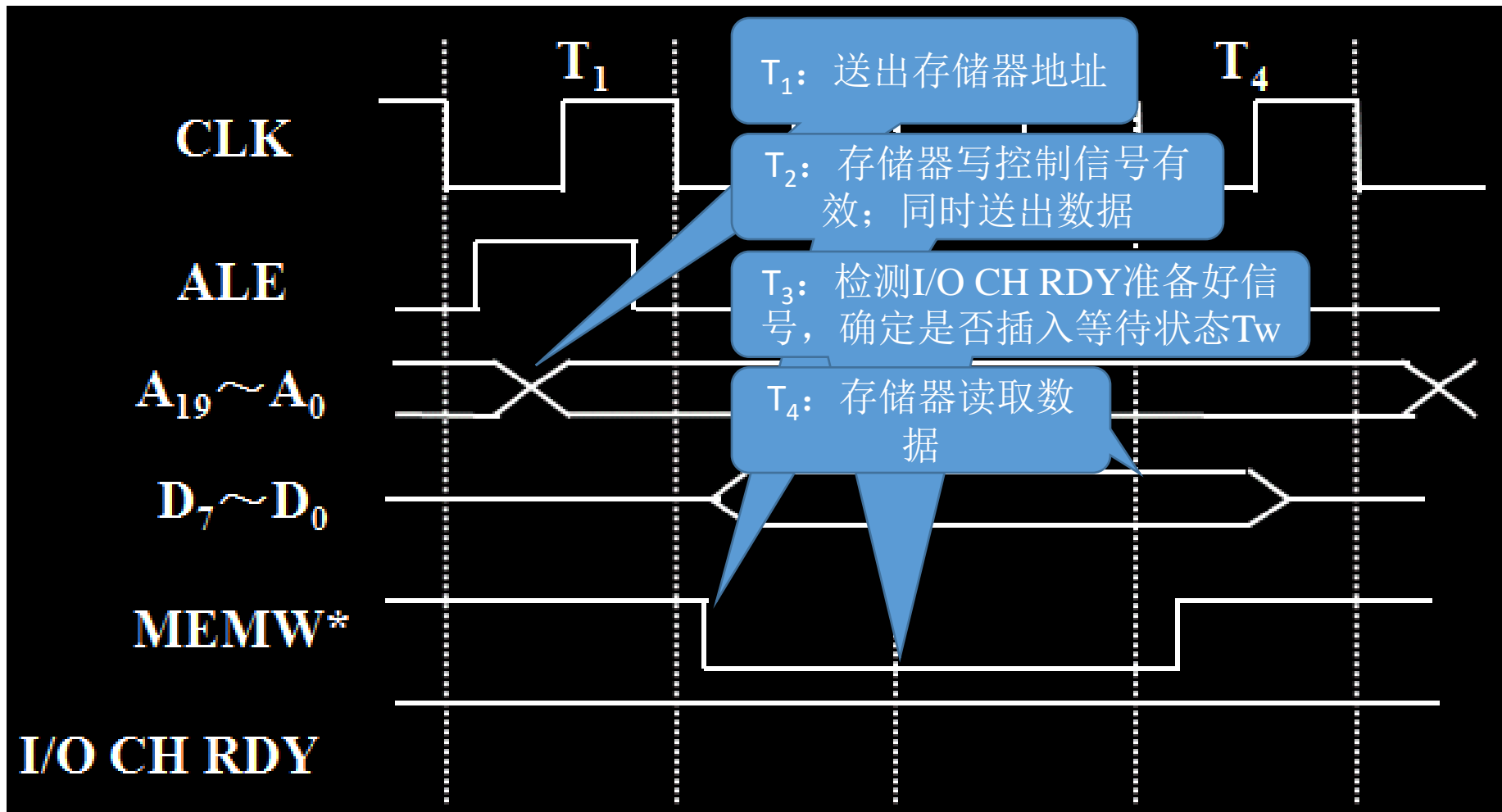
IBM PC 总线——总线周期



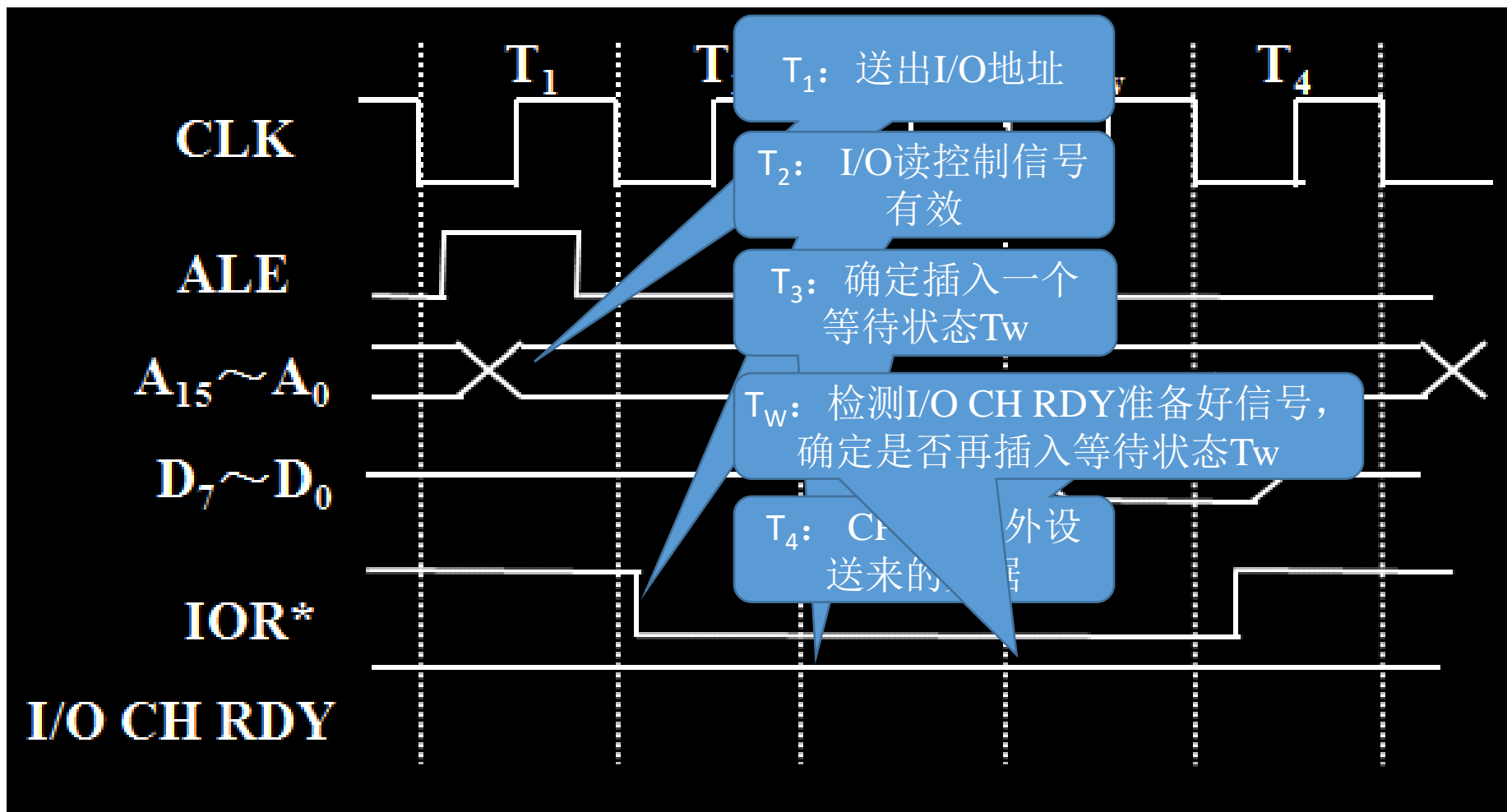
IBM PC存储器读总线周期



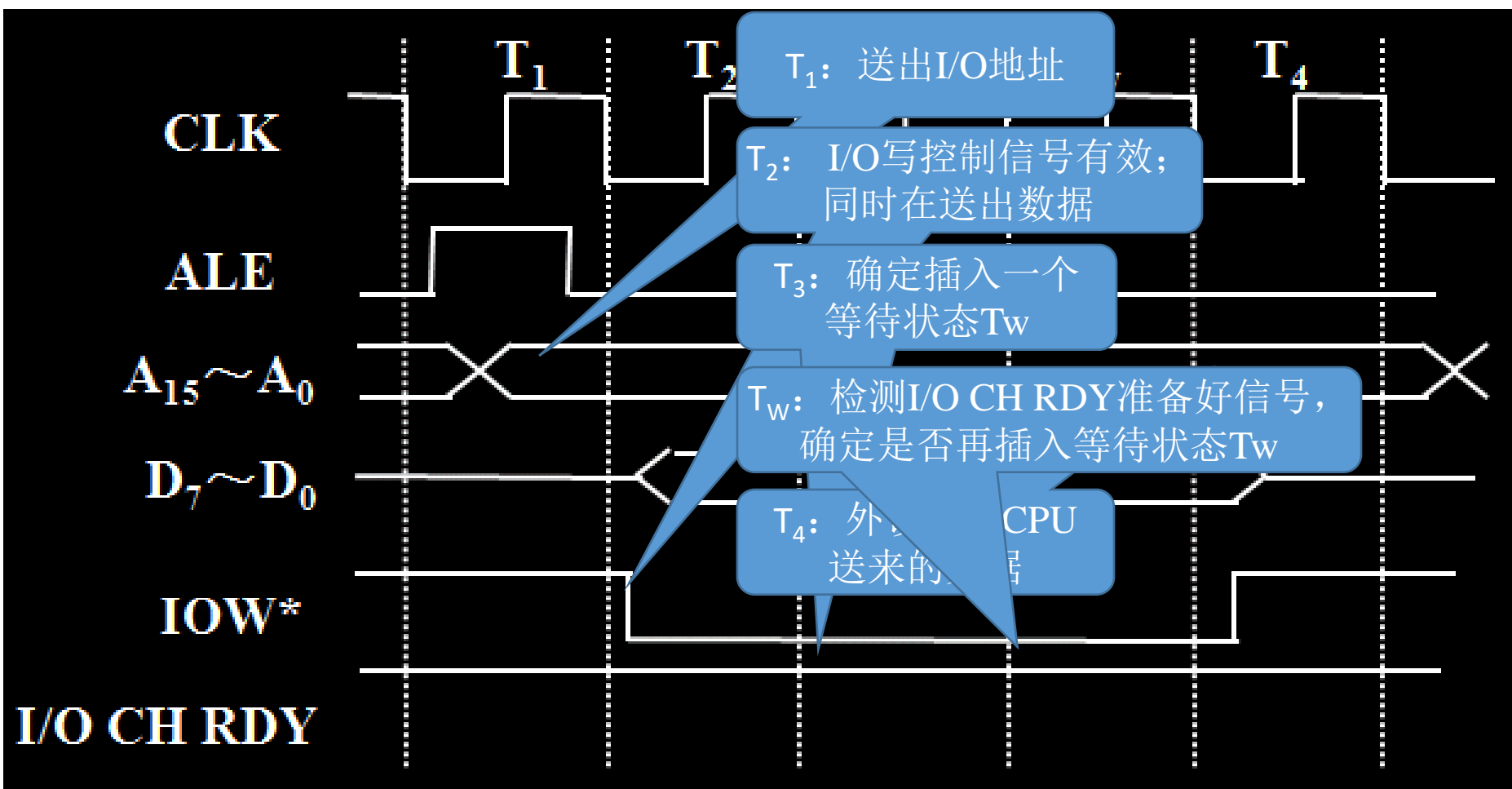
IBM PC存储器写总线周期



IBM PC I/O读总线周期



IBM PC I/O写总线周期



本章教学目的

- 1. 了解8088的两种组态形式；
- 2. 掌握最小组态下的引脚定义、总线形成和总线时序；
- 3. 了解最大组态下的引脚定义、总线形成和总线时序；
- 4. 理解总线及其层次结构
- 5. 掌握IBM PC总线的引脚及其总线周期。