第2章 80X86微处理器

> 课件来源：《第二章 80X86微处理器.ppt》（91 页，本章唯一版本，以下页码均指该课件 Slide 编号）。本章是全书的硬件基础章：物理地址计算、奇偶存储体访问、标志位判断、引脚功能与总线时序是期末必考点，几乎每年都以计算题/选择题/简答题形式出现。

0. 本章考点清单

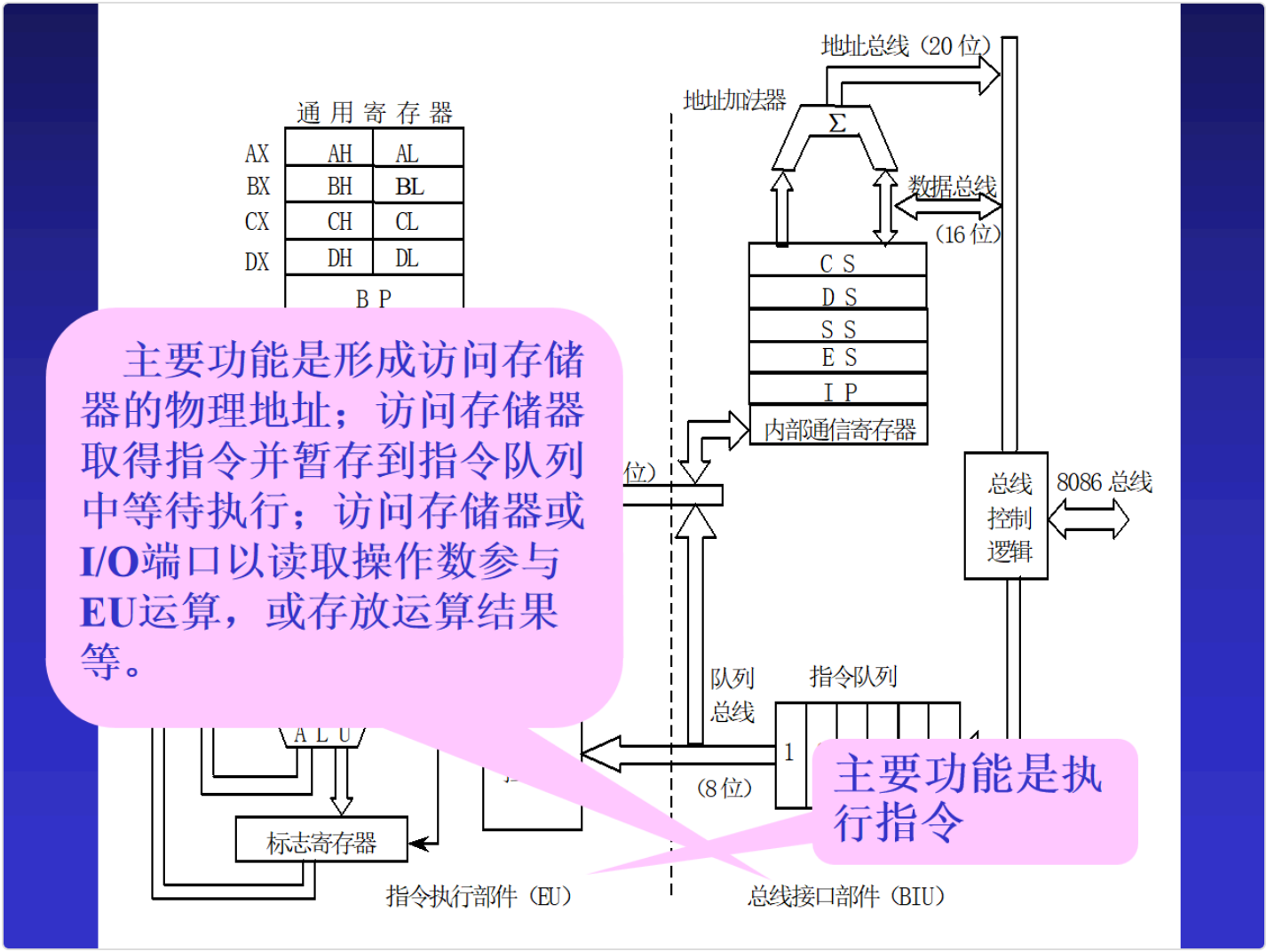
考点	常见题型	重要度
物理地址 = 段基址×16 + 偏移地址；逻辑地址不唯一	计算、填空	★★★
奇偶存储体结构；BHE、A0 组合与四种读/写情况（奇地址字 = 2 个总线周期）	选择、计算、综合	★★★
FLAGS 六个条件标志 + 三个控制标志的含义与运算后判断	选择、计算	★★★
寄存器专用用途（AX/CX/DX/SP/BP/SI/DI）与默认段搭配	填空、选择	★★★
引脚功能：RESET、READY、HOLD/HLDA、INTR/NMI、MN/MX、ALE、DEN、DT/R	选择、填空、简答	★★★
总线周期 T1~T4、Tw、Ti 各状态动作；读/写时序区别	简答、选择	★★★
最小模式典型配置：8284×1、8282×3、8286×2 及连接关系	简答、填空	★★
EU 与 BIU 的组成、分工与并行工作（指令队列）	简答、选择	★★
存储器分段：节(paragraph)、段基址、段重叠、内存专用区域	填空、选择	★★
最小/最大模式概念；最小模式控制信号分类（读写/中断/DMA）	选择	★★
复位操作后寄存器状态与第一条指令地址 FFFF0H	填空、简答	★★
时钟周期与总线周期换算（T = 1/f）	计算	★★

1. 8086/8088 概述与编程结构（EU 与 BIU）

概念精讲：8086 是 Intel 系列的 16 位微处理器，对外有 **16 根数据线、20 根地址线**，可寻址 2^{20} = **1MB** 空间（课件 p.2）。8088 的内部寄存器与指令功能与 8086 完全相同，但**对外数据总线只有 8 位**，处理一个 16 位数据需两步操作，故称**准 16 位**微处理器（课件 p.2）。本章讨论的内部结构是从指令执行全过程、即程序员角度看到的结构，称为**编程结构**，与真实物理结构有所区别（课件 p.2）。

编程结构分为两大部件（课件 p.3~5、p.22）：

部件	主要功能	组成
执行部件 EU	执行指令：从指令队列取指令、译码、发出控制命令完成运算	内部寄存器组（8 个 16 位）、算术逻辑单元 ALU 及标志寄存器 FLAGS、内部控制逻辑电路（课件 p.5、p.17）
总线接口部件 BIU	形成访问存储器的物理地址；取指令暂存指令队列；访问存储器或 I/O 端口读操作数/写结果	4 个段地址寄存器 CS/DS/ES/SS、16 位指令指针 IP、20 位地址加法器、6 字节指令队列缓冲器、总线控制逻辑（课件 p.22）



要点：- EU 不直接连外部总线；凡访问存储器/I/O，EU 向 BIU 申请总线周期，由 BIU 完成（课件 p.38）。- ALU 完成 16 位或 8 位算术逻辑运算，结果送通用寄存器或 BIU 内部寄存器等待写存储器，结果特征置入 FLAGS（课件 p.11）。- 内部控制逻辑：从队列取指令、译码、产生控制信号协调各部件（课件 p.17）。

EU 与 BIU 的并行管理（课件 p.38）：两部分并行工作但又不完全独立。EU 每执行一条指令就从队列头部取出，后续指令自动前推；EU 要访存/访 I/O 时向 BIU 申请总线周期，BIU 空闲则立即响应，**若 BIU 正在取指令则等取指完成后再响应**；遇到转移类指令，BIU 将队列中剩余指令作废、从新地址重新取指；一般应保证队列填满，使 EU 不断有指令可执行。二者既非同步也非无关，而是互相配合。

易错警示：「EU 负责取指令」是错的——取指、送队列是 BIU 的事，EU 只从队列取已取好的指令执行；「8088 是 8 位机」也是错的——它内部仍是 16 位，只是外部数据总线 8 位（课件 p.2）。

2. 寄存器组（EU 内部寄存器）

概念精讲：8086 共有 **8 个 16 位内部寄存器**，分两组：通用数据寄存器（AX、BX、CX、DX）和指针与变址寄存器（BP、SP、SI、DI）（课件 p.6）。四个通用数据寄存器既可作 16 位用，也可拆成 8 位用：AH/AL、BH/BL、CH/CL、DH/DL（课件 p.6）。

寄存器	名称	专用用途（课件页码）
AX(AH,AL)	累加器	有些指令约定以 AX/AL 为源或目的；**I/O 指令都通过 AX 或 AL 与接口传送信息**；中间运算结果多放 AX（p.6）
BX(BH,BL)	基址寄存器	间接寻址的地址寄存器、基地址寄存器；作基址时默认数据段 DS（p.7、p.9）
CX(CH,CL)	计数寄存器	**循环和串操作中充当计数器**，指令执行后 CX 内容自动修改（p.7）
DX(DH,DL)	数据寄存器	I/O 指令中作**端口地址寄存器**；乘除指令中作辅助累加器；32 位乘除法存放高 16 位（p.8）
BP	基址指针寄存器	访问内存时存放偏移地址；用作**堆栈区的基地址寄存器**，默认堆栈段 SS（p.8~9）
SP	堆栈指针寄存器	堆栈操作中存放**栈顶的偏移地址**，随堆栈指令自动修改（p.9、p.28）
SI	源变址寄存器	串操作中存放**源操作数**偏移地址；间接/变址寻址（p.10）
DI	目的变址寄存器	串操作中存放**目的操作数**偏移地址（p.10）

要点：- 指针和变址寄存器主要存放操作数的**偏移地址**（段内地址）（课件 p.9）。- 变址寄存器在串操作传送完成后地址**自动修改**（如传 1 字节后加 1），「变址」由此得名（课件 p.10）。- **BX 与 BP 的区别**：作通用寄存器都可存数据；作基址寄存器时，**BX 表示数据在数据段，BP 表示数据在堆栈段**（课件 p.9）。- 段寄存器与偏移寄存器的默认搭配（课件 p.29、p.61）：CS:IP → 代码段；DS(或 ES):SI/DI/BX → 数据段；SS:SP/BP → 堆栈段（图见第 4 节）。

易错警示：SP 只指向栈顶、不能当通用寄存器随意用；BP 默认配 SS 而不是 DS；端口地址大于 8 位时必须用 DX 间接寻址（课件 p.8 的 DX 用途）；串操作计数器是 CX 不是 SI。

3. 标志寄存器 FLAGS

概念精讲：FLAGS 共 **16 位，其中 7 位未用**；有效 9 位 = **6 位条件标志**（寄存程序运行状态，作后续指令判断依据）+ **3 位控制标志**（控制机器或程序的运行过程）（课件 p.12）。位分布（课件 p.12 图）：

位	15~12	11	10	9	8	7	6	5	4	3	2	1	0
标志	未用	OF	DF	IF	TF	SF	ZF	未用	AF	未用	PF	未用	CF

条件标志（课件 p.13~15）：

标志	名称	含义与置位条件
CF	进位标志	最高位（D7 或 D15）有进位（加）或借位（减）则 CF=1；无符号运算结果超范围（8 位加法 >255、减法差 <0）时 CF=1；**对有符号数运算无直接意义**（p.13）
PF	奇偶标志	结果**低 8 位**中「1」的个数为偶数则 PF=1，奇数则 PF=0；用于判断传送是否出错（p.14）
AF	辅助进位标志	加减时 **D3 向 D4** 有进位/借位则 AF=1；BCD 码运算时用于十进制调整（p.14）
ZF	零标志	结果为 0 则 ZF=1，否则 ZF=0（p.15；课件原文误写为「零进位」，存疑：应为零标志）
SF	符号标志	结果最高位为 1 则 SF=1（带符号数为负），否则 SF=0（p.15）
OF	溢出标志	补码表示的有符号数运算结果超出该字长表数范围则 OF=1；**对无符号数运算无意义**（p.15）

OF 的硬件判断方法：**OF = C7 ⊕ C6**（符号位的进位 ⊕ 次高位向符号位的进位），课件例 1 即如此计算（课件 p.18）。

控制标志（课件 p.16~17）：

标志	名称	作用与设置指令
DF	方向标志	串操作方向：DF=0 地址自动增量、低地址→高地址；DF=1 自动减量、高地址→低地址。**STD 置位、CLD 复位**（p.16）
IF	中断允许标志	IF=1 允许响应可屏蔽中断；**STI 置位、CLI 复位**（p.16）
TF	陷阱标志	TF=1 进入单步方式：每执行完一条指令自动产生一次内部中断，用于程序调试（p.17）

要点（标志的使用选择）：有符号数运算用 OF 判断溢出、不必关心 CF；无符号数运算用 CF 判断超范围、无需关心 OF（课件 p.19）。由例 2、例 3 可归纳有符号数比较结论：**OF=0 时 SF 即真实符号；OF=1 时 SF 与真实符号相反**，即 SF⊕OF=1 表示「小于」（课件 p.20~21 归纳）。

易错警示：PF 只统计低 8 位，不是整个 16 位结果；CF 与 OF 各管各的数制，不能混用；ZF=1 表示「结果为零」而不是「有进位」；TF 单步产生的是内部中断，与 INTR 引脚无关。

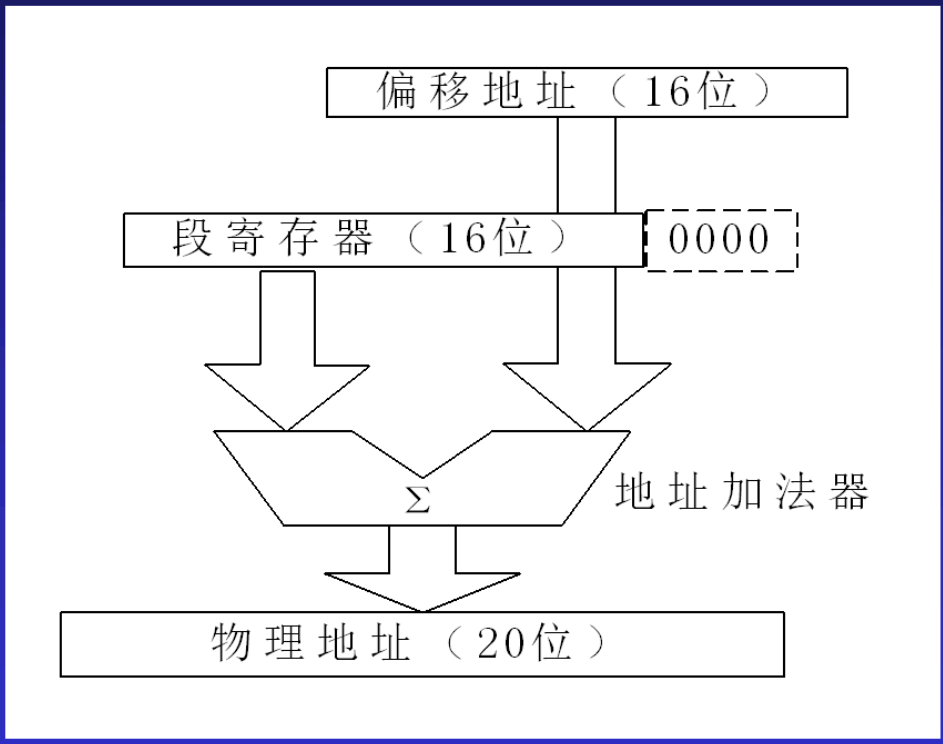
4. BIU 与存储器分段：逻辑地址与物理地址

概念精讲：8086 有 20 位地址线（1MB），但内部寄存器只有 16 位（只能寻址 64KB），故采用**地址分段**：把 1MB 分成若干**逻辑段**，每段容量 ≤64KB，存放一种类型的程序或数据；若均分不重叠，1MB 可分成 **16 个逻辑段（0~15）**（课件 p.23~24）。段内仍用 16 位地址寻址。

- **节(paragraph)**：16 字节的存储空间称一节；逻辑段必须从节的整数边界开始，即**段首地址低 4 位为 0**；段首地址的高 16 位称**段基址**，存于段寄存器（课件 p.24）。合法段首地址如 00000H、00010H、00020H……FFFF0H（课件 p.24）。
- **段浮动与重叠**：段可在整个空间浮动，段间可**部分重叠、完全重叠、连续排列**，可设置若干个逻辑段；任一物理地址可属于一个或多个重叠的逻辑段（课件 p.25）。
- **逻辑地址表示**：段基址:偏移地址，如 2345H:1100H 表示段起始 23450H、段内偏移 1100H 的单元（课件 p.26）。偏移地址 = 单元距段首的距离，又称段内地址。
- **四类段与段寄存器**（课件 p.27~28）：代码段 CS（存程序指令，正在执行段的段基址；超 64K 需分几段）、数据段 DS（当前使用的数据）、附加段 ES（需要第二个数据段时用）、堆栈段 SS（存调用子程序的入口参数、返回地址等，**先进后出**；SS 存段基址、SP 存栈顶偏移，堆栈指令自动修改 SP）。

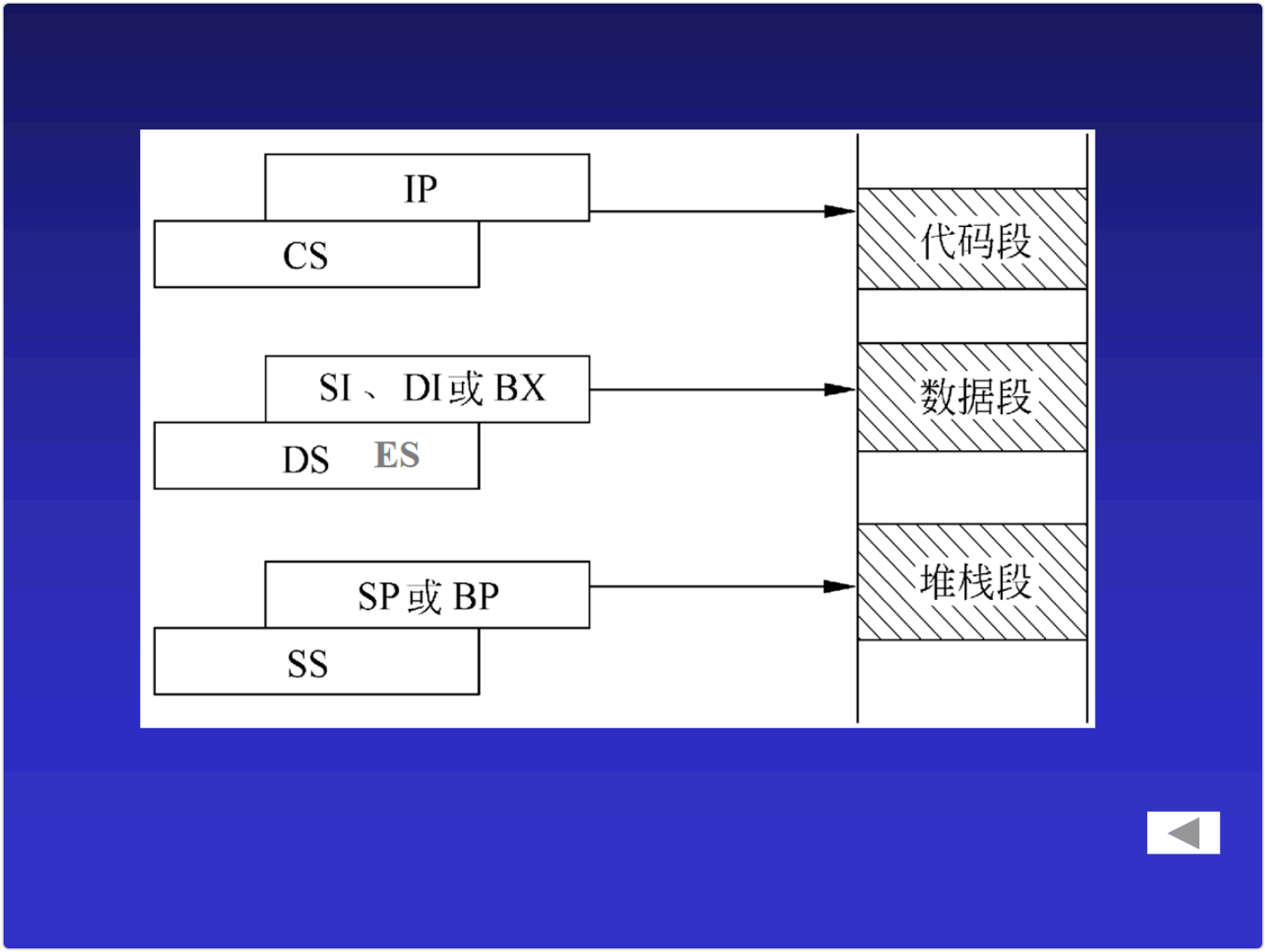
物理地址形成（课件 p.32）：BIU 中的 **20 位地址加法器**把段基址左移 4 位（即 ×16）与 16 位偏移地址相加：

```
物理地址 = 段基址 × 16 + 偏移地址        （20 位）
例：2345H:1100H → 23450H + 1100H = 24550H    （课件 p.34）
```



重要结论：一个存储单元的**物理地址唯一，逻辑地址不唯一**。如 24550H 可表示为 2455H:0000H，也可表示为 2400H:0550H（课件 p.34）；又如 CS=2000H、IP=1000H 与 CS=2100H、IP=0000H 都对应 21000H（课件 p.60）。

段寄存器与偏移寄存器的默认搭配（课件 p.29、p.61）：



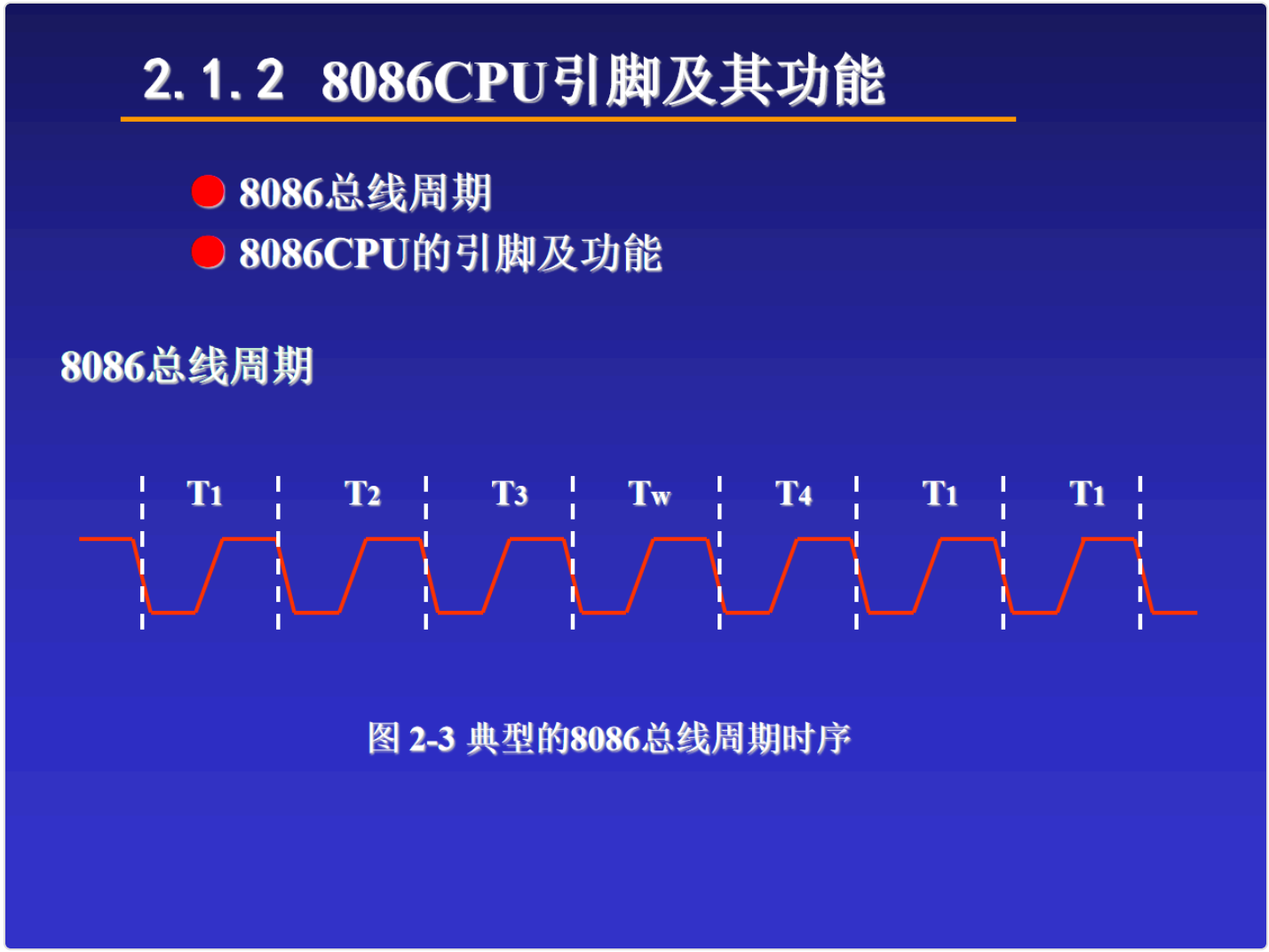
采用段地址的好处（课件 p.31）：① 解决 16 位寄存器访问大于 64KB 空间的问题；② 实现**程序重定位**——不到 64KB 的程序只需改变段寄存器内容（一般由操作系统完成）即可放在 1MB 任意段位置，便于多道程序运行。段地址在程序装入时由操作系统分配，用户程序一般只涉及偏移地址（课件 p.60）。

BIU 其余部件： - **IP（16 位）**：存放**将要执行的下一条指令**在代码段中的偏移地址；BIU 自动修改使其始终指向下一条指令；**程序不能直接访问 IP**，但转移指令会把目标地址送入 IP 实现转移（课件 p.35）。 - **指令队列缓冲器（6 字节）**：BIU 预取指令送入队列；**一旦队列空出 2 个字节，BIU 自动读指令填满**；收到 EU 的操作数地址立即形成物理地址完成读/写；**遇转移类指令队列剩余指令作废**、重新取指（课件 p.36）。 - **总线控制电路**：连接 CPU 内部总线与引脚外部总线（16 数据 + 20 地址 + 若干控制线），是 CPU 与外部交换信息的必经之路（课件 p.37）。

易错警示：×16 是左移 4 位（十六进制末尾添一个 0），不是乘 4 也不是乘 10H 以外的数；段基址本身不是段首物理地址（段首 = 段基址×16）；偏移地址最大 FFFFH，故段范围是「段首 ~ 段首+FFFFH」；IP 不能用 MOV 直接赋值。

5. 8086 总线周期与时钟

概念精讲（课件 p.41）： - **时钟周期 T**：系统主时钟 CLK 一个周期的持续时间，T = 1/f，是 CPU 基本时间计量单位。例：f = 5MHz → T = 200ns；f = 100MHz → T = 10ns（课件 p.41）。 - **总线周期**：CPU 通过外部总线对存储器或 I/O 端口进行**一次读/写操作**的过程；需发地址、发读/写命令、传数据，故一个总线周期由若干个 T 组成。基本总线周期 = **T1、T2、T3、T4 四个状态**（课件 p.40、p.86）。 - （补充常识）指令周期 = 取指 + 执行一条指令所需时间，由若干总线周期组成；三者关系：指令周期 ≥ 总线周期 ≥ 时钟周期。



各状态动作（课件 p.42~43）：

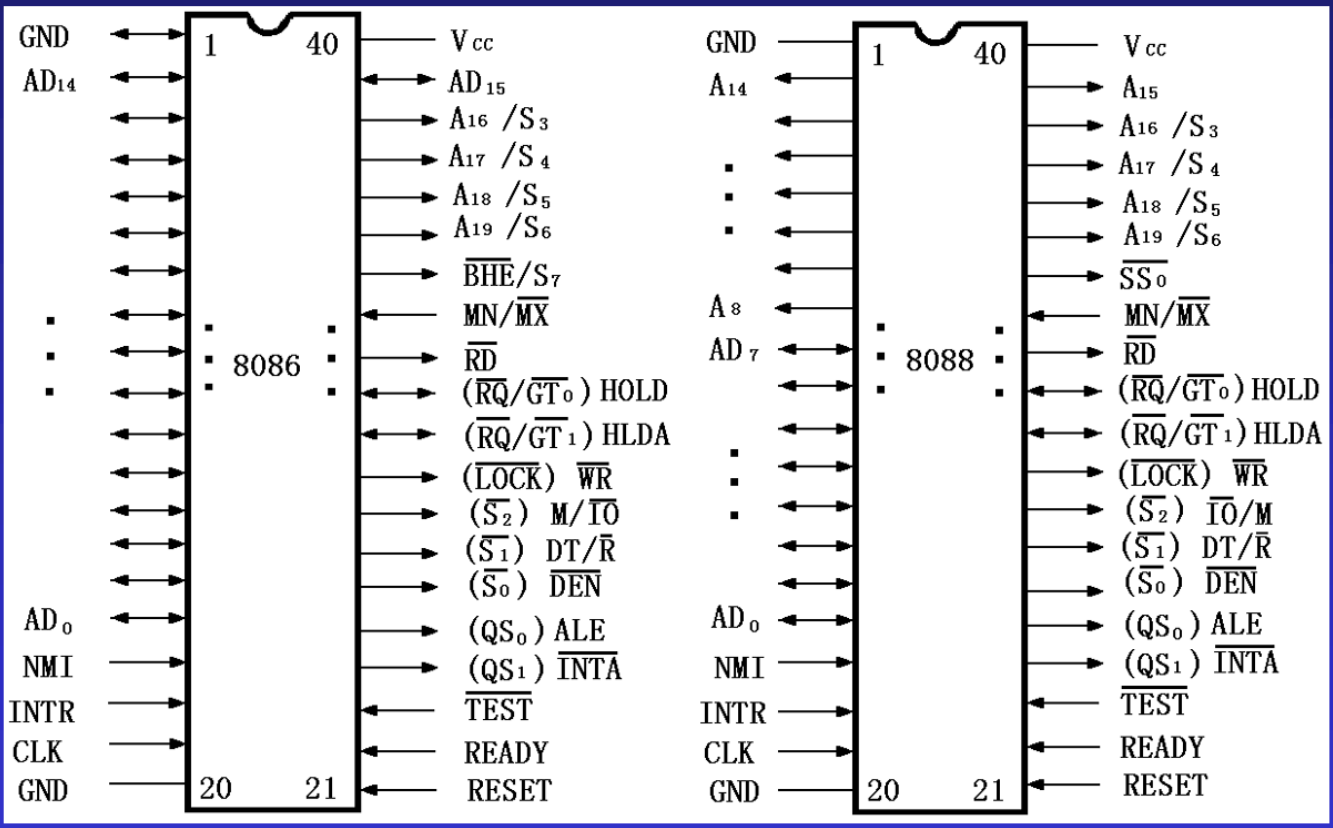
状态	动作
T1	CPU 向复用总线发地址 A19~A0；送出 **ALE 正脉冲**，其**下降沿**把地址锁入地址锁存器
T2	撤销地址，低 16 位总线呈高阻（为数据传输作准备）；高 4 位 A19~A16 输出状态信息 S6~S3（中断允许状态、所用段寄存器名）
T3	高 4 位状态不变；低 16 位出现要写出的数据或准备读入的数据；若外设/内存来不及，通过 **READY** 在 T3 前沿（下降沿）之前申请插入 Tw
Tw	等待状态；在 T3 及每个 Tw 的前沿查询 READY，高电平则结束等待进入下一状态，否则继续插 Tw
T4	总线周期结束；读周期在 T4 前沿将数据读入 CPU
Ti	总线空闲周期（两个总线周期之间）

易错警示：Tw 插在 **T3 与 T4 之间**，不是 T2 与 T3 之间；READY 采样点是 T3/Tw 的**前沿（下降沿）**；ALE 是正脉冲、用下降沿锁存；基本周期 4 个 T，插 n 个 Tw 就是 4+n 个 T。

6. 8086 CPU 引脚及其功能

概念精讲：8086 向外需 16 数据 + 20 地址 + 控制信号，为减少引脚采用**分时复用**（同一根线在不同时间传不同信息），构成 **40 引脚双列直插封装**（课件 p.39）。**最小/最大两种工作模式下，引脚 24~31 功能不同**，引脚图括号内为最大模式重新定义的信号（课件 p.46）。信号传输类型：输出、输入、双向、三态（高/低之外还有悬浮态，内外隔离）（课件 p.46）。

2. 8086 CPU的引脚及功能



8086/8088 引脚分配图

引脚编号速记（依课件 p.44 图）：1 GND；2~16 AD14~AD0；17 NMI；18 INTR；19 CLK；20 GND；21 RESET；22 READY；23 TEST；24 INTA(QS1)；25 ALE(QS0)；26 DEN(S0)；27 DT/R(S1)；28 M/IO(S2)；29 WR(LOCK)；30 HLDA(RQ/GT1)；31 HOLD(RQ/GT0)；32 RD；33 MN/MX；34 BHE/S7；35~38 A19/S6~A16/S3；39 AD15；40 VCC。

各引脚功能：

引脚	方向/有效电平	功能要点（课件页码）
AD15~AD0	双向、三态	T1 输出地址；读周期 T2 浮空（T3 由存储器送数据上总线），写周期 T2~T3 传数据（p.47；课件原文「T2~T3 读周期浮空」，存疑：按 p.83 读时序 T3 数据已上总线，浮空指 CPU 侧不驱动）
A19/S6~A16/S3	输出、三态	T1 输出地址高 4 位（**访问 I/O 时保持 0**）；T2~T4 输出状态 S6~S3（p.47）
BHE/S7	输出、三态、低有效	T1 输出 BHE=0 表示高 8 位数据线 D15~D8 有效（访问奇体）；T2~T4 输出 S7（**S7 无实际意义**）（p.49）
NMI	输入	**上升沿（正跳沿）触发**，不受 IF 影响、软件不可屏蔽；当前指令执行完后转**类型号 2** 的中断服务程序；级别高于 INTR（p.51）
INTR	输入	**高电平有效**；CPU 在每条指令**最后一个时钟周期**采样；IF=1 时结束当前指令后响应（p.51）
RD	输出、三态、低有效	读信号，T2、T3、Tw 期间有效；保持响应期间浮置高阻；读存储器还是 I/O 由 M/IO 决定（高=存储器，低=I/O）（p.52）
CLK	输入	时钟；**占空比 1/3**（1/3 高、2/3 低）；所有操作在时钟同步下进行（p.52）
RESET	输入、高有效	**至少维持 4 个时钟周期高电平**；FR、IP、DS、SS、ES 清 0，队列清空，**CS=FFFFH**；复位结束后从 CS:IP 取指，第一条指令物理地址 **FFFF0H**（该处一般放转移指令转引导程序）（p.53）
READY	输入、高有效	被访问设备发来的「准备好」响应；慢速设备在 T3 之前给低电平；CPU 在 T3 前沿采样，低则插 Tw，Tw 前沿继续查询直到变高才进 T4（p.54）
TEST	输入、低有效	与 **WAIT 指令**配合：WAIT 使 CPU 空转等待，TEST 收到低电平结束等待继续执行；实现 CPU 与外设同步（p.55）
MN/MX	输入	**接 +5V = 最小模式；接地 = 最大模式**（p.55）

状态信号 S6~S3（课件 p.48）：S6=0 表示 8086 占用系统总线（T2~T4 总为 0）；S5 反映 IF（IF=1 则 S5=1）；S4、S3 指明正在使用的段寄存器：

S4 S3	00	01	10	11
段寄存器	ES	SS	CS	DS

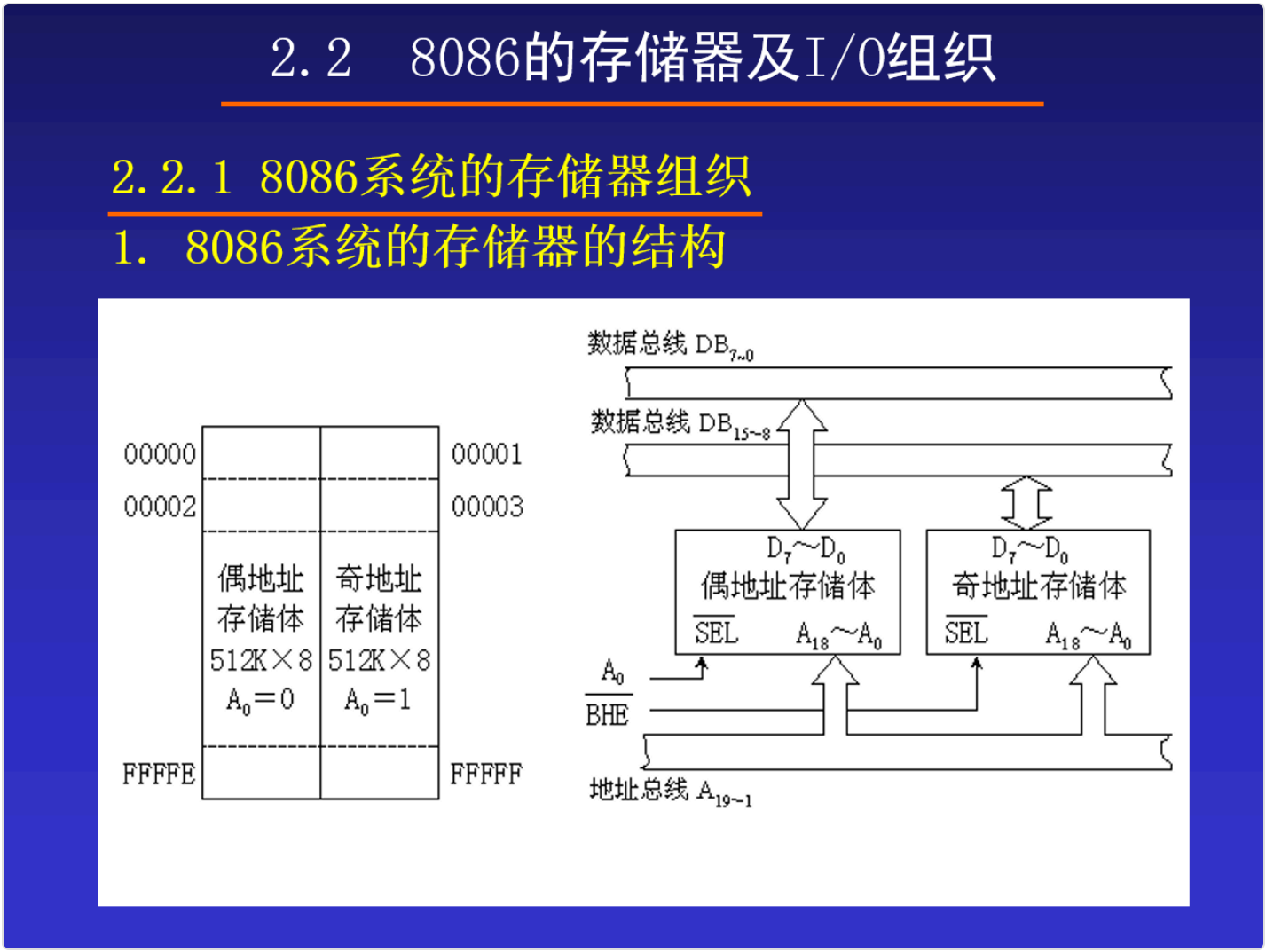
BHE 与 A0 的组合意义（课件 p.50，低 8 位数据线接偶体、高 8 位接奇体，见 p.49）：

BHE A0	操作	使用数据线
0 0	从偶地址开始读/写一个字	AD15~AD0
1 0	从偶地址读/写一个字节	AD7~AD0
0 1	从奇地址读/写一个字节	AD15~AD8
0 1（第一周期）→ 1 0（第二周期）	从奇地址开始读/写一个字（**2 个总线周期**）	先 AD15~AD8，后 AD7~AD0

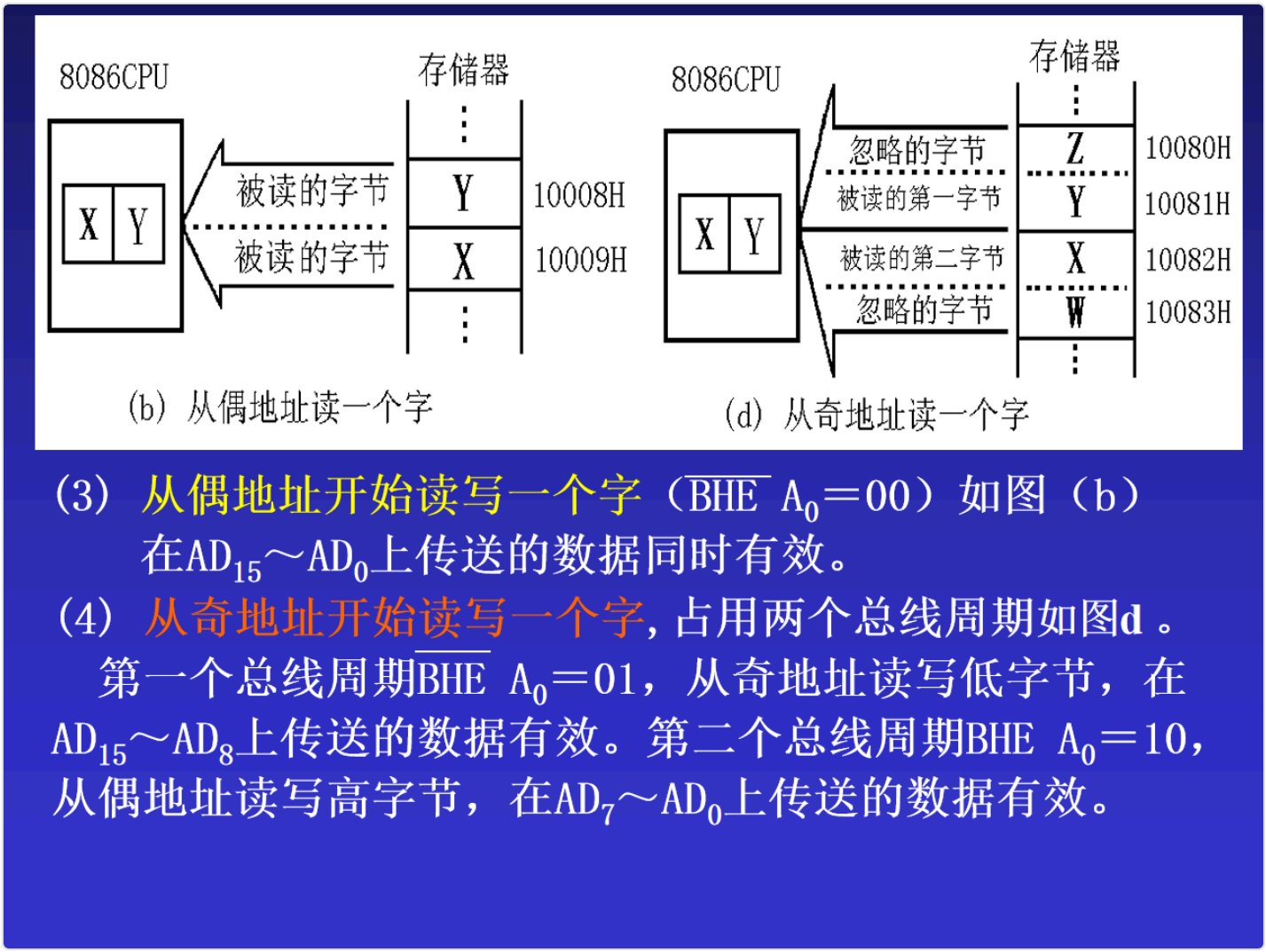
易错警示：NMI 是边沿触发、INTR 是电平触发，二者不可互换描述；复位后 CS=FFFFH 而**不是** 0，故首指令地址 FFFF0H 而非 00000H；访问 I/O 时 A19~A16 恒 0（端口只用低 16 位地址线中的 A15~A0）；S7 无意义，不要编造功能。

7. 8086 系统的存储器组织与 I/O 组织

概念精讲：1MB 空间分成两个 512K×8 的存储体：**偶地址存储体**（A0=0，接数据总线低 8 位 D7~D0）与**奇地址存储体**（A0=1，接高 8 位 D15~D8）；两体的体内地址均接 **A19~A1**；选体信号：偶体由 A0（=0 选中）、奇体由 BHE（=0 选中）（课件 p.49、p.57）。



四种读/写情况（课件 p.58~59）： 1. **偶地址字节**（BHE A0=10）：AD15~AD8 数据被忽略，字节经 AD7~AD0 传送（图 a）。 2. **奇地址字节**（BHE A0=01）：AD15~AD8 数据有效，AD7~AD0 被忽略（图 c）。 3. **偶地址开始一个字**（BHE A0=00）：AD15~AD0 同时有效，**1 个总线周期**（图 b）。 4. **奇地址开始一个字**：**占 2 个总线周期**（图 d）：第一周期 BHE A0=01，从奇地址读/写**低字节**（AD15~AD8 有效）；第二周期 BHE A0=10，从偶地址读/写**高字节**（AD7~AD0 有效）。



由图 (b)(d) 可见：**低字节存低地址、高字节存高地址**（如字单元 10008H 的低字节在 10008H、高字节在 10009H）（课件 p.59 图）。

存储器地址（课件 p.60）：物理地址 = 段地址×16 + 偏移地址；段地址由段寄存器提供，偏移地址由 IP、SP、BP、SI、DI 等寄存器或存储器提供，也可计算得到，**不同指令有不同组合方式**；段地址引入为程序浮动创造条件，段地址由操作系统装入时分配；同一物理地址可有不同段:偏移表示。

内存专用区域（课件 p.62，必背）：

地址范围	容量	用途
00000~003FFH	1KB	**中断向量表**：256 个中断服务程序入口地址，每个占 4 字节
B0000~B0FFFH	4KB	单色显示器显示缓冲区（屏幕字符 ASCII 码）
B8000~BBFFFH	16KB	彩色显示器显示缓冲区（屏幕像素代码）
FFFF0H	—	**起动地址**：一般放一条无条件转移指令，转系统初始化程序

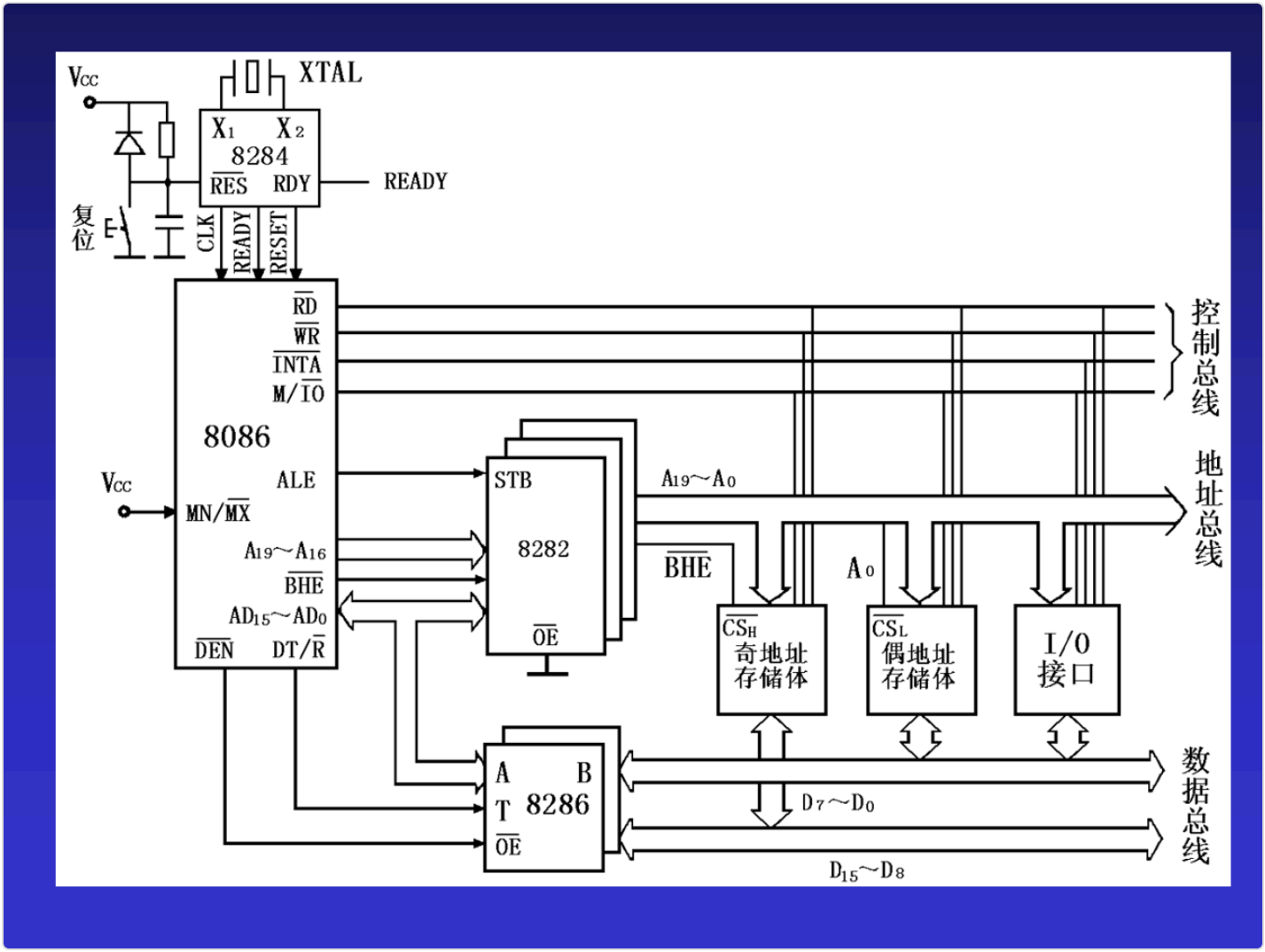
I/O 组织（课件 p.63）：8086 系统有专有的输入 **IN**、输出 **OUT** 指令，用于**外设端口**（外设接口中的内部寄存器）寻址；I/O 数据传送经 AX/AL（课件 p.6），端口地址可放 DX（课件 p.8）。

易错警示：奇地址字访问是 2 个总线周期而偶地址字是 1 个，这是 8086 与「字对齐」编程建议的由来；8088 因外部数据总线仅 8 位，**任何字访问都要 2 个总线周期**（课件 p.2）；中断向量表在最低 1KB，不要与显示缓冲区地址记混。

8. 最小/最大模式与最小模式系统配置

概念精讲（课件 p.63~64）：8086/8088 提供两种工作模式，由**硬件连接**（MN/MX 引脚电平）决定： - **最小模式**：系统中只有一个微处理器 8086/8088，**所有总线控制信号由它产生**，总线控制逻辑减到最小。MN/MX 接 +5V（课件 p.65）。 - **最大模式**：系统包含两个以上处理器，一个 8086/8088 作主处理器，其余作协处理器。MN/MX 接地（课件 p.55）。

最小模式典型配置（课件 p.65~66）：**1 片 8086 + 1 片时钟发生器 8284 + 3 片地址锁存器 8282 + 2 片总线驱动器 8286（总线数据收发器）**。

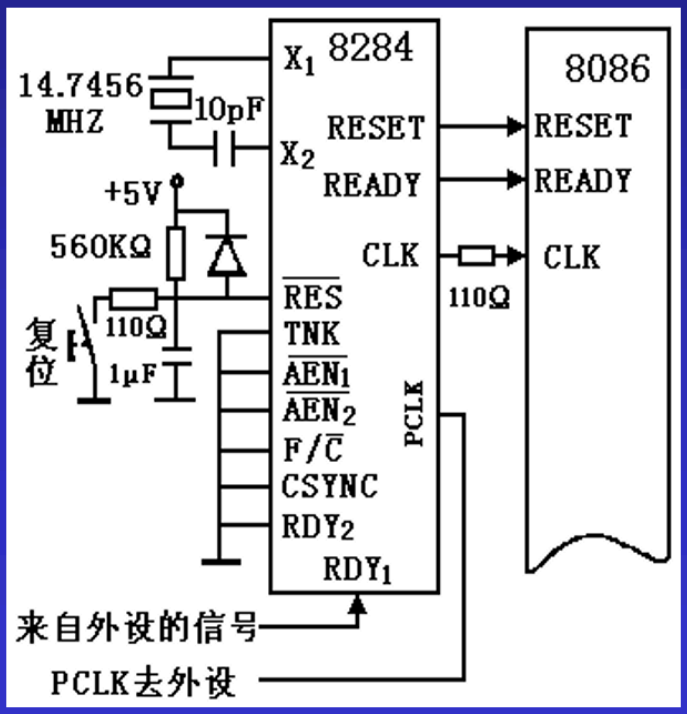


CPU 及其外围芯片合称 **CPU 子系统**，外围芯片作用（课件 p.68）：① 提供时钟、对外界控制/联络信号同步处理；② 分离地址/数据分时复用信号，得到独立地址总线与数据总线并增强驱动能力；③ 组合 CPU 控制信号，产生稳定可靠、便于使用的系统总线信号。

8284 时钟发生器（课件 p.69）：外接晶振产生时钟并 **3 分频**，得到**占空比 1/3** 的 CLK 送 8086；还对外部输入的 RESET、READY 进行**同步**，产生与 CLK 同步的复位/准备好信号送 8086（图中晶振示例 14.7456MHz，3 分频后 CLK≈4.9152MHz）。

2. 时钟发生器8284与8086的连接

8284通过外接晶振芯片产生时钟信号，并对这个信号3分频，产生占空比为1/3的时钟信号CLK送往8086 CPU。8284还对外部输入的RESET和READY信号进行同步，产生与CLK同步的复位信号RESET和准备就绪信号READY送往8086。

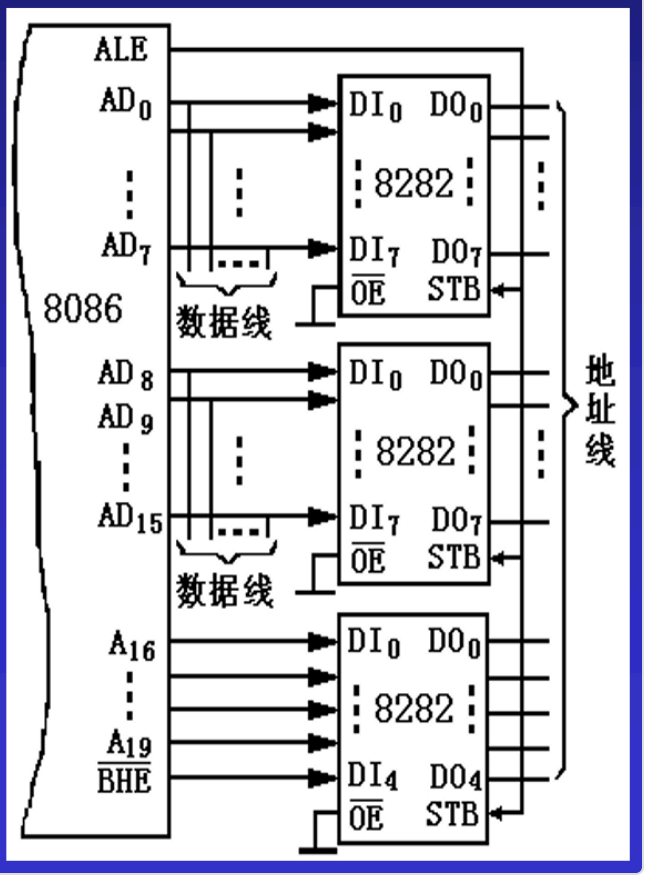


8282 地址锁存器（课件 p.70~71）：8 位锁存器；**STB 高电平**时 DI7~DI0 输入进入锁存器，**STB 下降沿**锁定数据；**OE 低电平**时经三态缓冲器从 DO7~DO0 输出。连接：**8086 的 ALE 接 8282 的 STB**；T1 期间 AD15~AD0、A19/S6~A16/S3 上的地址被 **3 片 8282** 锁存（第三片还锁存 BHE），8282 输出即系统地址总线，整个访问周期稳定输出 20 位地址；最小模式下 8282 还同时锁存控制信号送系统总线；可用其他三态输出锁存器代替。

3. 地址锁存器8282与8086的连接

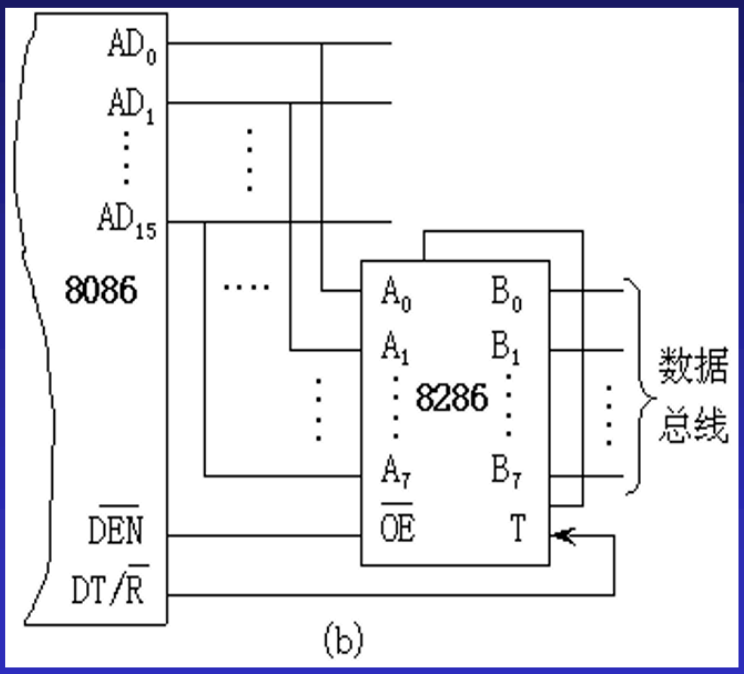
地址锁存器用来锁存8086输出的地址信号。

8282是一个8位锁存器，STB是它的数据锁存/选通信号。STB为高电平时，DI_7~DI_0上输入的信号进入锁存器；STB由高变低出现下降沿时，输入数据被锁定，锁存器的状态不再改变。8282具有三态输出功能，$\overline{OE}$是数据输出允许端，它为低电平时，锁存器的内容通过内部的三态缓冲器从引脚DO_7~DO_0输出。



8286 总线驱动器（课件 p.72~74）：三态输出的 **8 位双向总线收发器/驱动器**，两组 8 位数据线 A7~A0、B7~B0，驱动能力强。控制信号：**T**（方向）与 **OE**（输出允许，低有效）：OE=1 高阻、双向都不通；OE=0、T=1 时 A→B；OE=0、T=0 时 B→A。连接：**T 接 DT/R，OE 接 DEN**；两片 8286 的 A 侧接 AD15~AD0，B 侧成为系统数据总线；保证只有 CPU 访问存储器/I/O 时数据才通过 8286。**若系统规模不大且不用 DMA（总线永远由 8086 独自控制），可不用收发器，AD15~AD0 直接作系统数据总线。**

8286用作数据总线驱动器时，T与8086数据收发信号相连，用于控制数据传送方向；$\overline{OE}$与8086数据允许信号相连，保证只有在CPU需要访问存储器或I/O端口时才允许数据通过8286。二片8286的A_7~A_0与8086的AD_{15}~AD_0相连，而二组B_7~B_0则成为系统数据总线。



如果系统规模不大，并且不使用DMA传输（这意味着总线永远由8086独自控制），可以不使用总线收发器，将8086的引脚AD_{15}~AD_0直接用作系统数据总线。

最大模式说明：课件正文只给出最大模式概念（p.64）与最大模式总线读时序（p.87，见第 10 节）：CPU 输出状态信号 S2~S0，读/写命令（图中带 * 的 MRDC、IORC 及 ALE、DT/R、DEN）由总线控制器产生（补充常识：该控制器为 8288）。小结目录列有「2.3.3 最大模式系统」但课件无专门展开页（课件 p.90）。

易错警示：8282 锁**地址**（ALE→STB）、8286 驱**数据**（DEN→OE、DT/R→T），二者控制信号不可接反；8282 用 3 片是因为要锁 20 位地址 + BHE 共 21 位（3×8=24）；8284 是 3 分频不是 2 分频；MN/MX 接 +5V 是最小模式，接地才是最大模式。

9. 最小模式控制信号归纳（读写 / 中断 / DMA）

读写控制信号（课件 p.75~77）：控制传输方向（读/写）、种类（存储器/I/O）、方式（奇/偶字节/字）、设备是否准备好、分时总线信号类型。

信号	方向/有效电平	作用
M/IO	三态输出	高 = 访问存储器；低 = 访问 I/O 端口（p.75）
RD	三态输出、低有效	正在读存储器或 I/O（p.76）
WR	三态输出、低有效	正在写存储器或 I/O；M/IO、WR、RD 合起来决定数据传输方向（p.76）
READY	输入、高有效	设备已准备好；无效则插 Tw 直到有效（p.76）
BHE/S7	三态输出、低有效	高字节有效；非数据传送期间作 S7 输出状态（p.76）
ALE	输出、高有效	地址锁存允许：复用引脚上正在输出地址；送 8282 作锁存信号（p.77）
DEN	三态输出、低有效	数据允许：复用引脚上正在传数据；作 8286 输出允许；**DMA 传输时置高阻**（p.77）
DT/R	三态输出	高 = 发送（CPU 写）；低 = 接收（CPU 读）；控制 8286 方向（p.77）

中断控制信号（课件 p.78）：INTR 可屏蔽中断请求（输入）；**INTA 中断响应**（输出、三态、低有效）：含**两个负脉冲**，在两个连续中断响应周期发出——第一个通知外设 CPU 已收到并响应 INTR（主要用于中断优先权排队），第二个要求申请设备向 CPU 发送**中断类型号**；NMI 非屏蔽中断请求（输入）。

DMA 控制信号（课件 p.79）：DMA 是不经过 CPU、在内存与 I/O 设备间直接传数据的方法，使用前须申请总线并获认可。**HOLD 总线请求**（输入、高有效）：其他设备请求使用总线；**HLDA 总线响应**（输出、高有效）：CPU 每个时钟周期检测 HOLD，若允许让出，则在**当前总线周期的 T4 状态**发 HLDA，同时所有三态总线置高阻（不影响外部存储器与 I/O 交换数据）；申请部件接到 HLDA 后接管总线，操作完成撤销 HOLD，CPU 才重新接管。

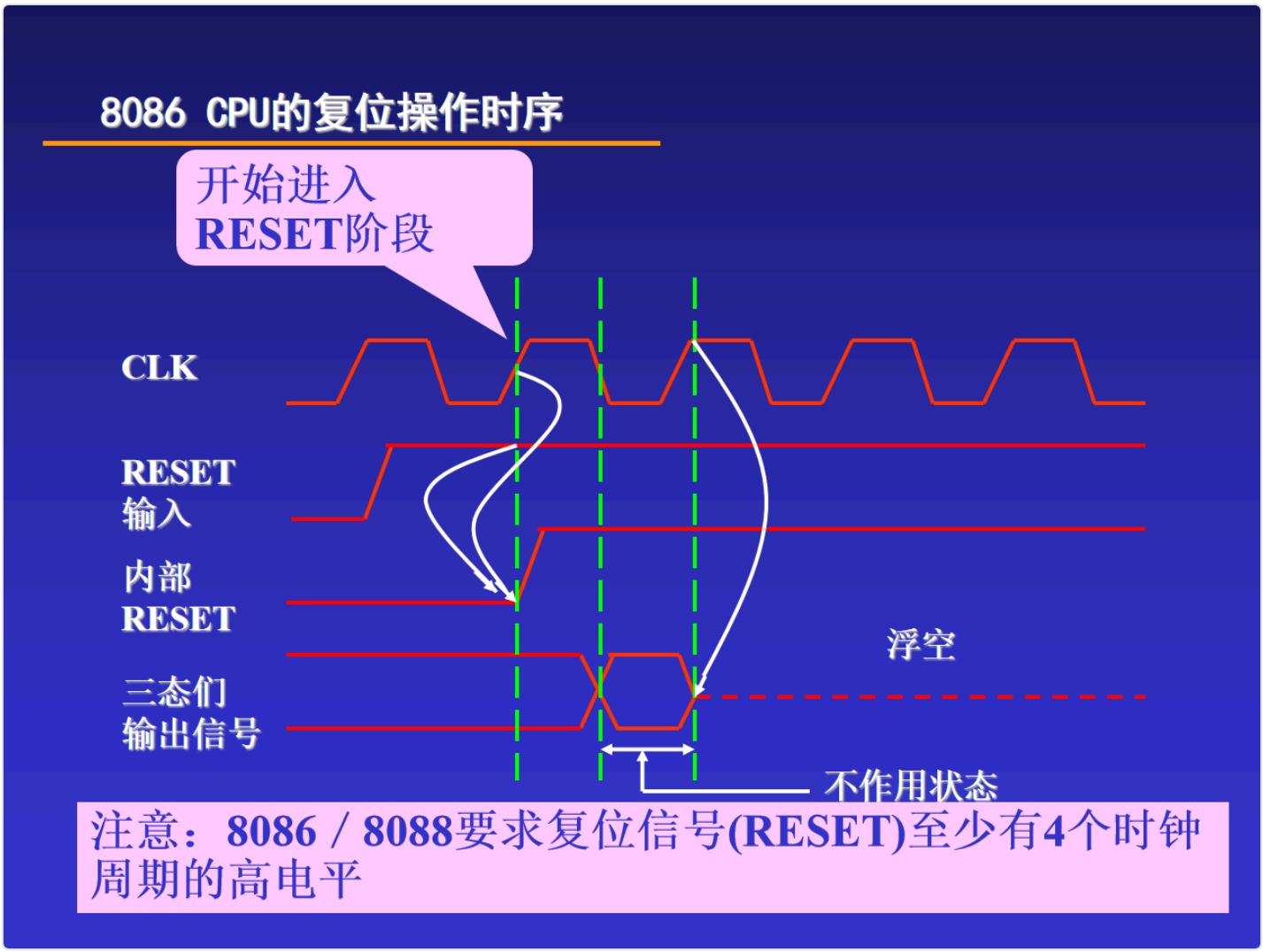
最小模式控制总线信号归纳（课件 p.80）： - 控制存储器/I/O 读写：M/IO、RD、WR、BHE、DEN、DT/R、READY； - 中断联络与控制：INTR、INTA、NMI； - DMA 联络与控制：HOLD、HLDA。 这些信号是构建微型计算机系统的核心，后面章节反复使用。

易错警示：INTA 是**两个**负脉冲、各有分工；HLDA 在 **T4** 发出（总线保持时序图中为 T4 或 Ti，课件 p.88）；DMA 期间 DEN 高阻、CPU 让出的是三态总线而非全部引脚（RESET、CLK 等仍工作）。

10. 8086 CPU 的操作时序

10.1 复位操作时序（课件 p.81~82、p.89）

复位内容：① 所有内部寄存器、标志寄存器及 ES、SS、DS、IP 清 0，指令队列清空，**CS 置全 1 = FFFFH**；复位信号消失后程序从 CS×16+IP = **FFFF0H** 开始执行；② 复位时所有三态输出总线变高阻。时序要求：**RESET 至少 4 个时钟周期高电平**（课件 p.82 图 2-14）。

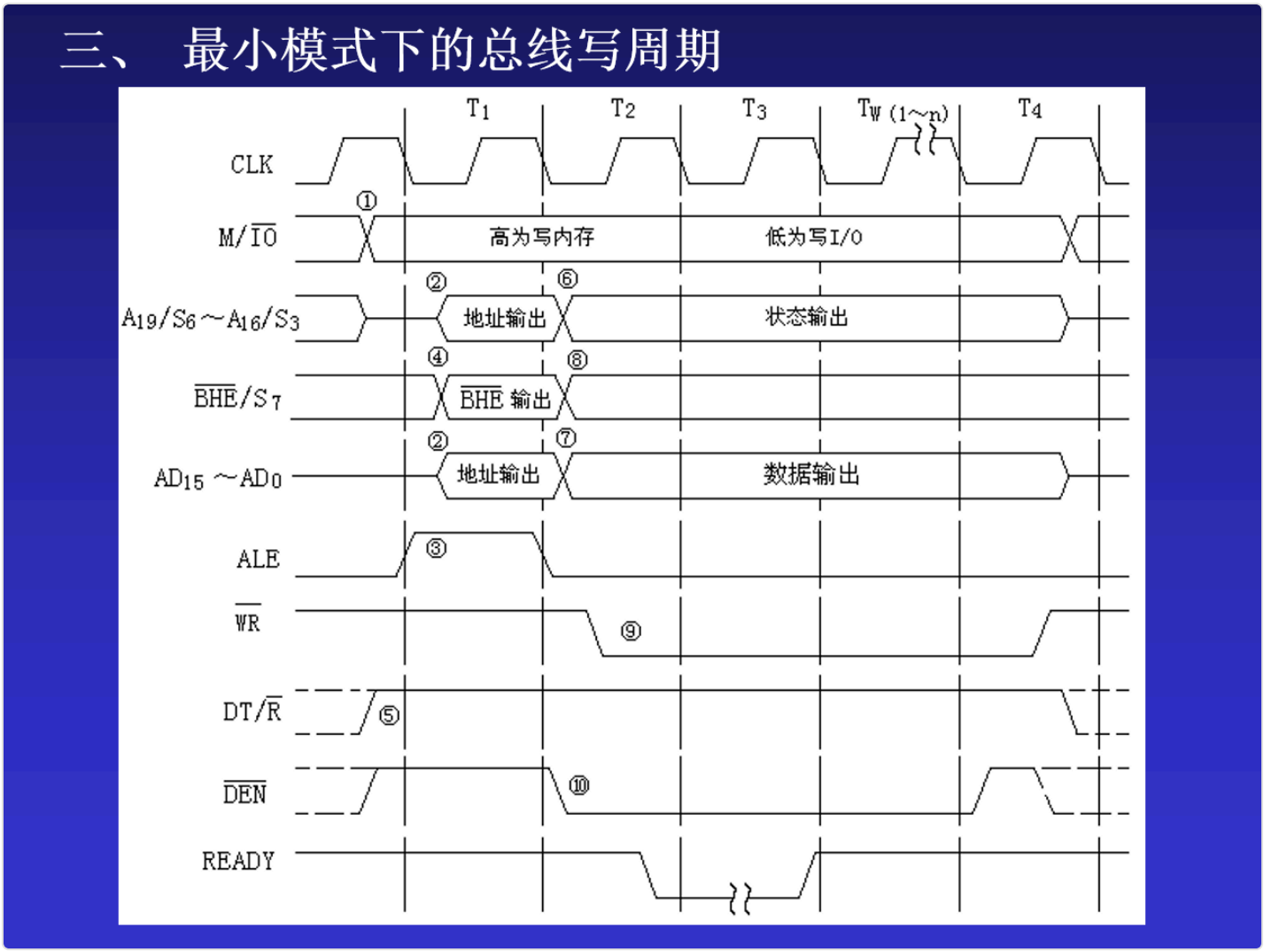


10.2 最小模式总线读周期（课件 p.83~84）

状态	动作
T1	M/IO 有效（定存储器/I/O）；20 位地址有效；ALE 有效、T1 结束变低，地址锁入 8282；读奇体时 BHE=0；DT/R 输出低（读周期，8286 接收）
T2	地址撤销，A19/S6~A16/S3 与 BHE/S7 输出状态 S7~S3 维持到 T4；AD15~AD0 高阻为读作准备；RD 变低送所有存储器/I/O（只对选中者起作用）；DEN 变低开放 8286（维持到 T4 中期）；DT/R 保持低
T3	存储器/I/O 把数据送上总线，CPU 经 AD15~AD0 接收；**T3 前沿（下降沿）采样 READY**：高 = 已准备好，T3 结束读数据；低 = 在 T3、T4 间插 Tw
Tw	每个 Tw 前沿采样 READY：低继续插；高则在该 Tw 结束时读数据
T4	在 T3(Tw) 与 T4 交界下降沿采样数据总线完成读取；T4 后半周数据撤销，控制/状态信号无效，读周期结束

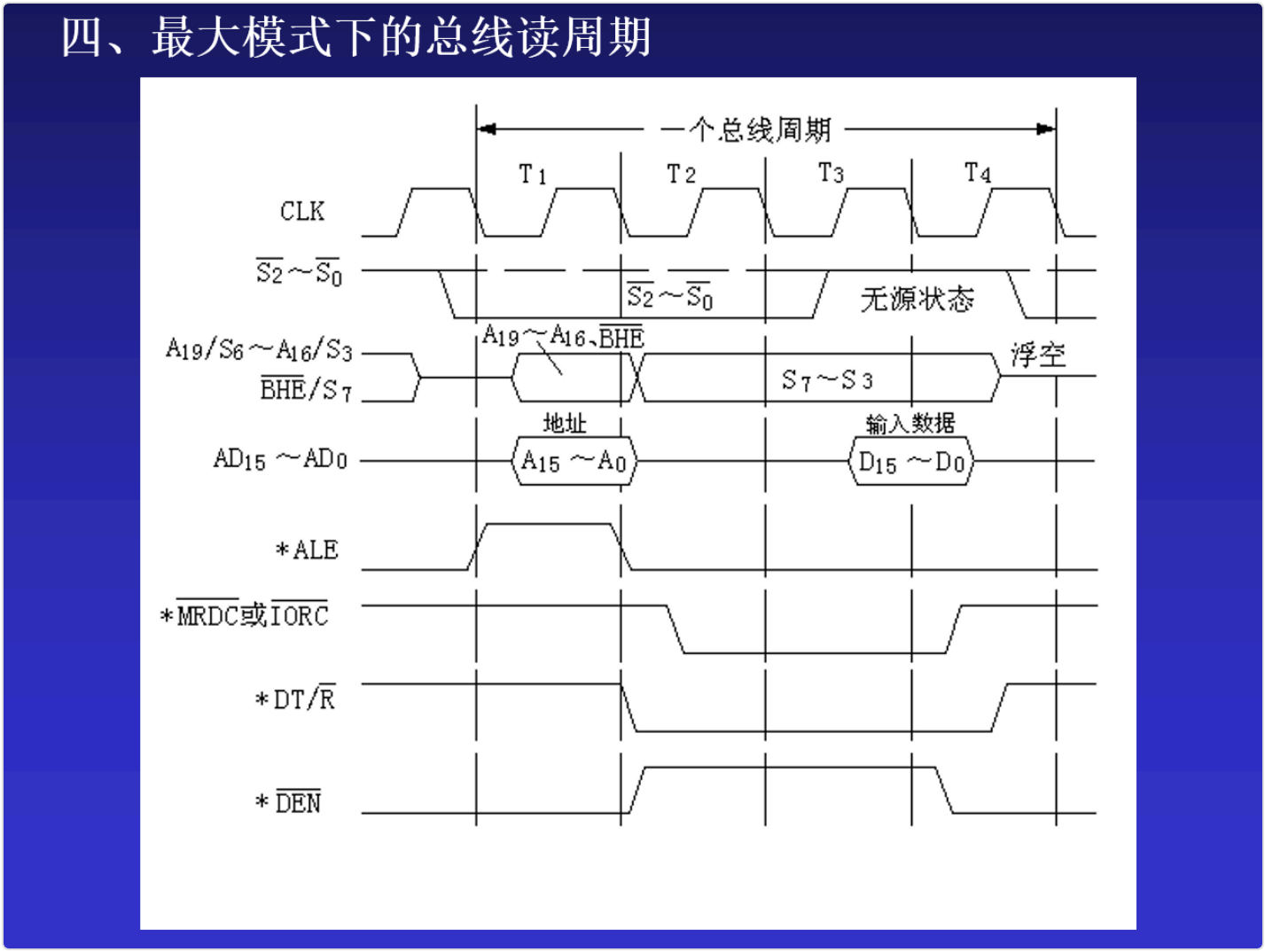
归纳（课件 p.84）：**T1 送地址及相关信号；T2 发读命令和 8286 控制命令；T3、Tw 等待数据出现；T4 将数据读入 CPU**。

10.3 最小模式总线写周期（课件 p.85~86）



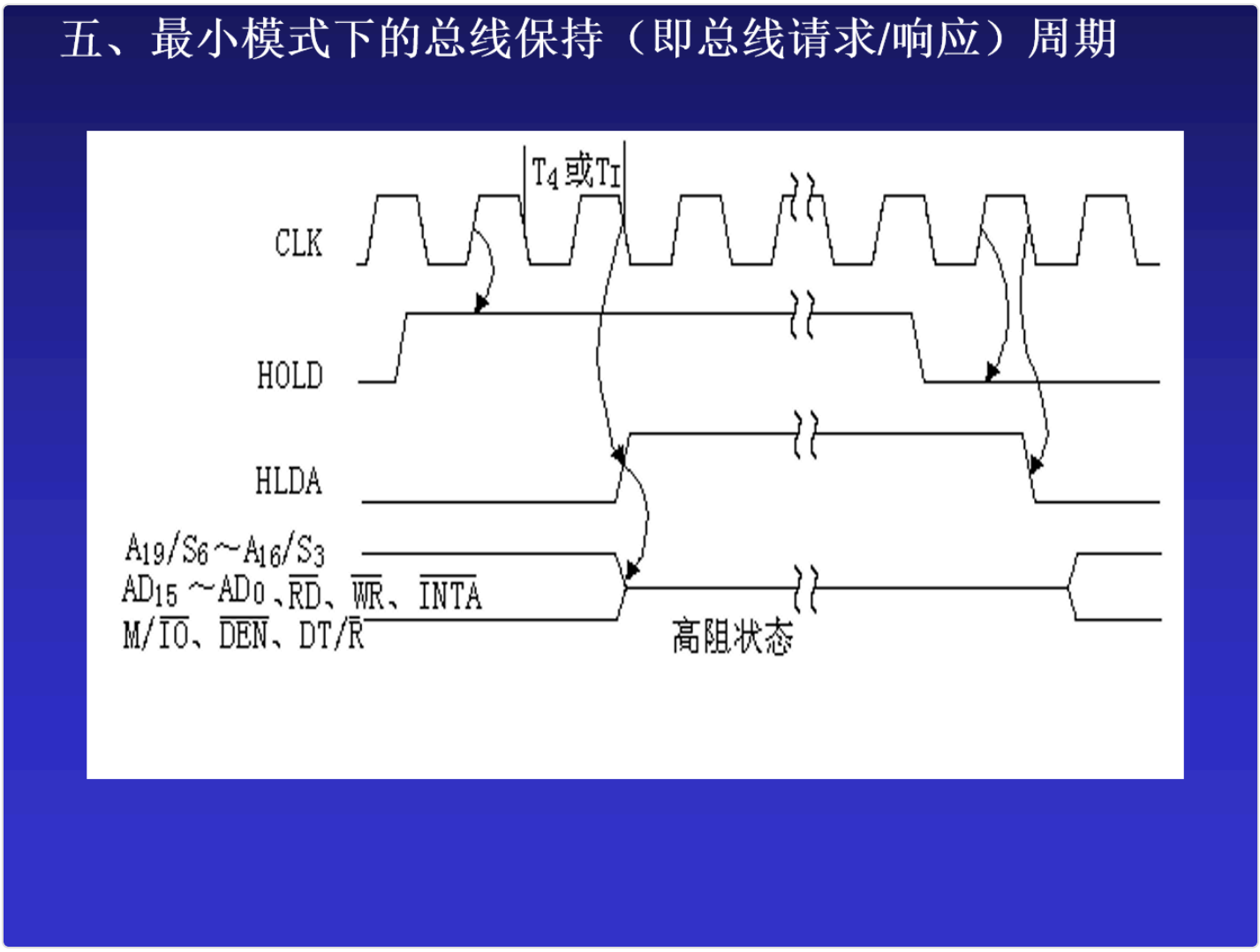
写周期同样含 T1~T4，慢速设备在 T3、T4 间插 Tw。**与读周期的区别**（课件 p.86）： 1. 写周期中地址（T1）与数据（T2）在 AD15~AD0 上**同方向**输出，故 **T2 不再需要读周期那样的高阻缓冲**，发完地址立即发数据，设备一旦准备好即可取走； 2. **DT/R 为高电平**（写周期，控制 8286 向外发送）； 3. **WR 有效、RD 无效**，但出现时间类似。

10.4 最大模式总线读周期（课件 p.87）



与最小模式读周期框架相同（T1 地址、T2 发命令、T3 数据、T4 结束），但 CPU 引脚 24~31 改为输出状态 **S2~S0** 等，读命令 **MRDC/IORC** 及 ALE、DT/R、DEN（图中带 * 者）由总线控制器根据状态译码产生（补充常识：8288）。

10.5 最小模式总线保持（总线请求/响应）周期（课件 p.88）



过程：外部设备拉高 HOLD → CPU 在 **T4 或 Ti** 发 HLDA → A19/S6~A16/S3、AD15~AD0、RD、WR、INTA、M/IO、DEN、DT/R 等进入**高阻状态**，DMA 控制器接管总线 → 传送完成撤销 HOLD → HLDA 随之撤销，CPU 重新接管总线（课件 p.79、p.88）。

易错警示：读周期 T2 有高阻缓冲、写周期没有，这是读/写时序最常被考的区别；DT/R 读低写高；复位让 CS=FFFFH 与总线保持让出总线是两回事；最大模式不是「引脚更多」，而是 24~31 引脚改义、控制信号外移。

11. 本章小结与目录展望

课件小结（p.90~91）给出的框架：2.1 8086 微处理器（编程结构 EU/BIU、引脚功能）；2.2 存储器及 I/O 组织（存储器结构、I/O 组织）；2.3 系统配置（最小/最大模式概念、最小模式系统、最大模式系统）；2.4 操作时序（复位、最小模式读/写、最大模式读/写、最小模式总线保持、最大模式总线请求/允许）。

小结目录还列出 **2.5 80386（寄存器组、地址转换、存储管理与保护、流水线结构）、2.6 80486（特点、内部结构）、2.7 Pentium（体系结构特点、相对 486 的增强点）**，但课件正文**没有对应展开页**（课件 p.91），本笔记不作展开；若考试按目录涉及，需另行参考教材（补充常识：80386 为 32 位寄存器/32 位数据总线，80486 集成 Cache 与 FPU，Pentium 采用超标量双流水线——仅供衔接，非课件内容）。

12. 易错点与高频陷阱汇总

1. 陷阱：把 8088 当 8 位机。正解：8088 内部 16 位、仅外部数据总线 8 位，是准 16 位（课件 p.2）。
2. 陷阱：物理地址 = 段基址×16 + 偏移，误算成段基址×4 或直接把段基址当段首。正解：段首 = 段基址×16（左移 4 位、末尾添 0H）（课件 p.32）。
3. 陷阱：认为一个物理地址只对应一个逻辑地址。正解：物理地址唯一、逻辑地址不唯一（课件 p.34、p.60）。
4. 陷阱：PF 统计整个结果的 1 个数。正解：只看低 8 位，偶数个 1 才 PF=1（课件 p.14）。
5. 陷阱：用 CF 判断有符号溢出、用 OF 判断无符号超范围。正解：CF 管无符号、OF 管有符号，互不通用（课件 p.13、p.15、p.19）。
6. 陷阱：OF=1 时仍按 SF 判断大小。正解：OF=1 时真实符号与 SF 相反（SF⊕OF=1 为小于）（课件 p.20~21）。
7. 陷阱：BP 作基址默认 DS。正解：BP 默认 SS，BX 才默认 DS（课件 p.9）。
8. 陷阱：奇地址字访问 1 个总线周期。正解：8086 奇地址字 = 2 个总线周期（先 01 后 10）；偶地址字才 1 个（课件 p.50、p.59）。
9. 陷阱：Tw 插在 T2、T3 之间或 READY 在 T4 采样。正解：Tw 在 T3、T4 之间，T3/Tw 前沿采样 READY（课件 p.43、p.54）。
10. 陷阱：复位后 CS=0、从 00000H 启动。正解：CS=FFFFH、IP=0，首指令物理地址 FFFF0H（课件 p.53、p.81）。
11. 陷阱：NMI 可被 CLI 屏蔽 / INTR 边沿触发。正解：NMI 上升沿、不可屏蔽、类型号 2；INTR 高电平、受 IF 控制（课件 p.51）。
12. 陷阱：8282 接 DEN、8286 接 ALE。正解：ALE→8282 的 STB（锁地址）；DEN→8286 的 OE、DT/R→T（驱数据）（课件 p.71、p.74）。
13. 陷阱：写周期 T2 也浮空一个周期。正解：写周期地址后直接送数据，无高阻缓冲；DT/R 写高读低（课件 p.86）。
14. 陷阱：MN/MX 接地为最小模式。正解：接 +5V 最小、接地最大（课件 p.55）。
15. 陷阱：INTA 一个脉冲完成响应。正解：两个负脉冲，第一个应答/排队、第二个取中断类型号（课件 p.78）。
16. 陷阱：队列空 1 字节就预取 / 转移后队列继续用。正解：8086 队列 6 字节、空出 2 字节即预取；转移指令使队列作废重取（课件 p.36、p.38）。

13. 典型例题精讲

例 A（标志位全判断，课件 p.18~19 例 1 加法部分）：AL=3BH、AH=7DH，求 (AL)+(AH) 后六标志。 步骤：① 展开二进制 0011 1011 + 0111 1101；② 逐位相加得 1011 1000 = B8H；③ 最高位无进位 → CF=0；④ D3 向 D4 有进位（B+D=18H）→ AF=1；⑤ 结果 4 个 1（偶）→ PF=1；⑥ D7=1 → SF=1；⑦ C7=0、C6=1 → OF=C7⊕C6=1；⑧ 结果非 0 → ZF=0。教学点：先写二进制竖式再逐标志对照定义，OF 用「双进位异或」最稳。

例 B（溢出时的有符号比较，课件 p.21 例 3）：AL=-125（83H）、BL=50（32H），执行 AL-BL 后判断大小。步骤：① 1000 0011 - 0011 0010 = 0101 0001（51H=81）；② D6 向 D7 有借位、D7 向更高位无借位 →

OF=1；③ SF=0；④ 负减正应得负却得正 → 溢出，真实符号与 SF 相反 → AL<BL。教学点：验证 -125-50=-175 超出 8 位补码范围 [-128,127]，与 OF=1 自洽；结论「OF=1 且 SF=0 → 小于」。

例 C（物理地址 + 奇偶体综合）：设 DS=2000H，字变量偏移地址 0051H。(1) 求物理地址；(2) 8086 读该字需几个总线周期？各周期 BHE、A0 与有效数据线？ 步骤：(1) 20000H+0051H=20051H；(2) 20051H 为奇地址 → 2 个周期：第一周期 BHE A0=01，读低字节（20051H），AD15~AD8 有效；第二周期 BHE A0=10，读高字节（20052H），AD7~AD0 有效。教学点：把「地址计算 → 奇偶判断 → 查 BHE/A0 表」串成一条链，是期末综合题的标准套路。

附：本章速查表

公式与数据 - 物理地址 = 段基址×16 + 偏移地址（20 位）；寻址空间 2^{20} = 1MB；段 ≤64KB；节 = 16B（课件 p.2、p.24、p.32） - T = 1/f；5MHz→200ns；100MHz→10ns；基本总线周期 4T，插 n 个 Tw 为 (4+n)T（课件 p.41） - 复位：RESET ≥4 个时钟高电平；CS=FFFFH，其余段/IP/FR=0；首指令 FFFF0H（课件 p.53） - 指令队列 6 字节，空 2 字节预取；CLK 占空比 1/3；8284 三分频（课件 p.36、p.52、p.69）

寄存器用途一句话：AX 累加/I/O 必经；BX 基址(默认 DS)；CX 循环/串计数；DX 端口地址/乘除高位；SP 栈顶(SS)；BP 堆栈基址(SS)；SI 串源；DI 串目的（课件 p.6~10）。

FLAGS 位：11 OF、10 DF、9 IF、8 TF、7 SF、6 ZF、4 AF、2 PF、0 CF；15~12、5、3、1 未用（课件 p.12）。设置指令：STD/CLD 控 DF，STI/CLI 控 IF（课件 p.16）。

BHE A0：00 偶字(AD15~0)；10 偶字节(AD7~0)；01 奇字节(AD15~8)；奇字 = 01→10 两周期（课件 p.50）。

S4 S3：00 ES、01 SS、10 CS、11 DS；S5=IF；S6=0（课件 p.48）。

专用内存区：00000~003FFH 中断向量表(256×4B)；B0000~B0FFFH 单显 4KB；B8000~BBFFFH 彩显 16KB；FFFF0H 起动转移（课件 p.62）。

最小模式配置：8284×1（时钟/同步 RESET、READY）、8282×3（ALE→STB 锁 20 位地址+BHE）、8286×2（DEN→OE、DT/R→T 驱数据）（课件 p.66、p.69~74）。

T 状态动作：T1 地址+ALE；T2 读=高阻+RD/DEN 有效、写=直接送数据+WR 有效；T3 数据+采样 READY；Tw 等待；T4 读入/结束（课件 p.42~43、p.84、p.86）。

信号分类：读写 M/IO、RD、WR、BHE、DEN、DT/R、READY；中断 INTR、INTA、NMI；DMA HOLD、HLDA（课件 p.80）。