

1 题 省略

2 题

总线：计算机系统各部分之间传递地址、数据和控制信息的共同线路的集合。  
总线结构的好处：物理上是同时可以挂多个设备，但是总线上某一个时刻只有一个设备传送信息。

从空间上分为四类：片内总线，片间总线（主总线），系统总线（PCI，ISA 等标准总线）（是从 PC 机主板角度分类的，如图 1 所示），系统间通讯总线。  
标准总线好处：总线接口标准统一便于通用。

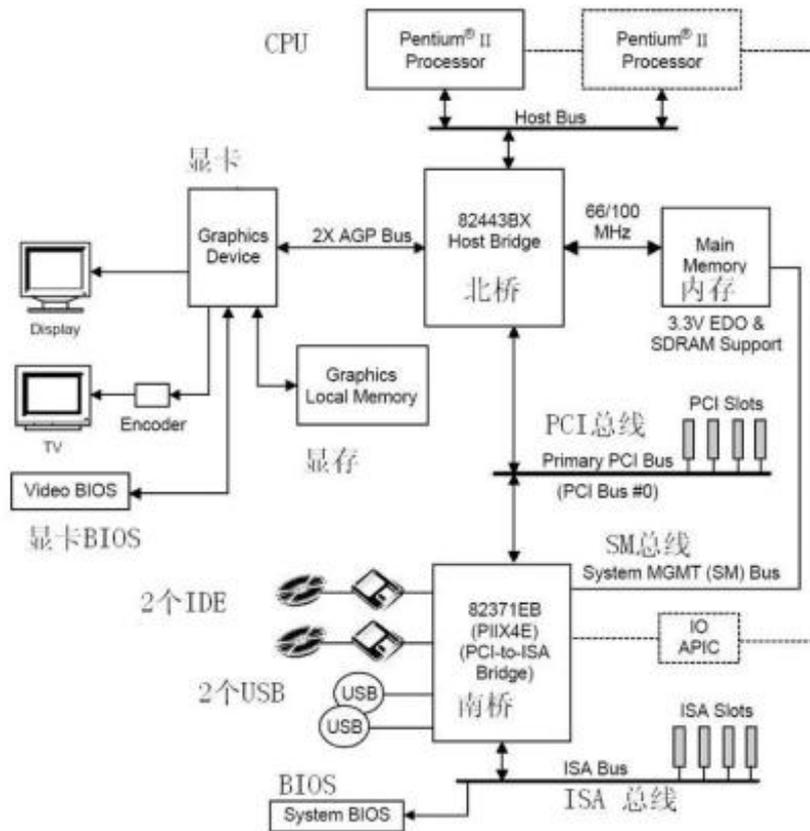


图 1 PII 的 CPU 主板系统总线

总线传递信息分为三大类：数据总线，地址总线，控制总线。系统总线等也是由是三大总线组成，只是根据不同的需要经过了转换。

3 题

总线接口单元(BIU)：

- 1) 形成访问内存或者 I/O 端口的物理地址；
- 2) 从内存的代码段中取指令送到队列缓冲器中；
- 3) 读取操作数参加 EU 运算或则存放运算结果到指定地址。

执行单元(EU)

- 1) 从 BIU 的指令队列缓冲器中去指令进行译码、执行；
- 2) 负责所有的算术，逻辑运算及 16 位偏移地址计算；
- 3) 向 BIU 提供指令的执行结果和 16 位偏移地址。

二者同步进行操作，相当两级流水线，提高了效率。

4 题 省略

5 题 T1 时刻是地址信息，用 ALE 信号锁存。**注：**凡是有 ALE 信号的 CPU 的某些管脚都具有分时复用的功能。（如 51 单片机，AVR 单片机等）。T3 时刻数据信息送出。通过地址锁存器保持地址信息。数据放大器（应为总线驱动器），加强总线的带载能力。

6 题略

7 题

7825H  
+ 5A1FH  
D244H            CF=0; OF=1; ZF=0; PF=1; AF=1; SF=1;

7825H  
-5A1FH  
1E06H            CF=0; **OF=0**; ZF=0; PF=1; AF=1; SF=0;

8 题

最大模式：除了 8086/8088 处理器外还有协处理器

最小模式：系统只有 8086/8088 处理器

9 题

访问存储器使用到哪些信号： WR RD M/IO ALE AD0-AD15  
A16/S3-A19/S6 等

10 题

一个指令周期至少包含一个总线周期（即：取指令读存取器的所花的总线周期），一个总线周期包含 4 个时钟周，可会在 T3 后插入的等待周期。

11 题 省略

12 题 主要点为以下三点，有以下原因的前两点还会引申出来的其他区别，如执行指令的效率。但是二者的程序是完全兼容的。

8086	8088
外部总线 16 位	外部总线 8 位
队列缓冲字节 6 个	队列缓冲字节 4 个
M/IO	IO/M

13 题 分时复用的信号，在总线周期前面出现的信号只要在后面的周期里还要维持其原来信号稳定不变的，就都要进行锁存。

14 题 指令队列缓冲区使得 CPU 可以取指和执行指令并行进行。8086 复位后，CS=0FFFFH，其他寄存器为 0，复位后程序指针指向了物理地址的 FFFF0H 的位置，一般在此处都是一条跳转指令，由此处跳到用户的自己的监控程序的入口处（开始的地方）。此方法在其他的 MCU 中也是这么用的，具有通用性。

15 题

实际地址（物理地址）是 20 位的，而内部数据总线是 16 位的，只能通过段加偏移的办法

段起始地址最低 4 位为 0。

16 题

地址线为 N，则寻址存储单位（字节）多少的关系为： $2^N$ 。也可能为 4 位，16 位等。

17 题

逻辑地址表示为 段地址: 偏移地址，是在编程时使用，计算机也是利用他们二者来找到实际地址的。

实际地址（物理地址）=段地址\*16+偏移地址

段地址只是段起始地址（是一个段开始的物理地址，为 20 位）的高 16 位。

18 题 二者都是 1117AH，同一物理地址可以写成不同逻辑地址的形式，但在实际中一般不会出现。

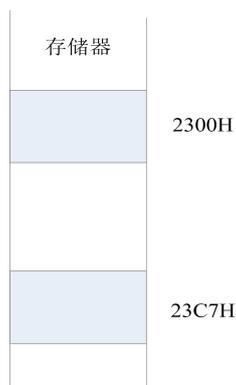
19 题

代码段的寻址就是通过  $CS*16+IP$ ，所以指令存放的第一个字节单元的物理地址

$$A7B00H+2B50H=AA650H$$

20 题

100 字 为 200 字节  $200=C8H$  偏移从 2300H 开始，



21 题

字节数据  $[123D4H]=78H$   $[123D6H]=34H$   
 字数据  $[123D4H]=5678H$   $[123D6H]=0E634H$

22 题



$CS=1000H, DS=1100H, SS=1180H$

23 题、 偶地址开始存放的字数据为规则字，奇地址开始存放的字数据为非规则字(8086 为 16 位机器)。当计算机为 32 是规则字应该是存放在能被 4 整除的

地址。