

第一章 微型计算机的基础知识（13 道）

1. 微处理器、微型计算机和微型计算机系统三者之间有什么不同？

将运算器与控制器集成在一起，被称为微处理器。微型计算机是由微处理器、存储器、输入/输出接口电路和系统总线所构成的裸机系统。微型计算机系统是以微型计算机为主机，配上系统软件和外设之后而构成的计算机系统。三者之间是有很大的不同的，微处理器是微型计算机的一个组成部分，而微型计算机又是微型计算机系统的一个组成部分。

2. 说明微型计算机系统的工作过程。

微型计算机系统的基本工作过程是执行程序的过程，也就是 CPU 自动从程序存放的第 1 个存储单元起，逐步取出指令、分析指令，并根据指令所规定的操作类型和操作对象，执行指令规定的相关操作。如此重复，周而复始，直至执行完程序的所有指令，从而实现程序的基本功能。

3. 说明微型计算机由哪些功能部件组成？试说明“存储程序控制”的概念。

微型计算机的硬件主要由运算器、控制器、存储器、输入设备和输出设备五大基本部件组成。

“存储程序控制”的概念：

（1）计算机应由运算器、控制器、存储器、输入设备和输出设备五大基本部件组成。

（2）计算机内部采用二进制来表示程序和数据。

（3）将编好的程序和原始数据事先存入存储器中，然后再启动计算机工作，使计算机在不需要人工干预的情况下，自动、高速的从存储器中取出指令并加以执行，这就是存储程序的基本含义。

（4）五大部件以运算器为中心进行组织。

4. 冯·诺依曼计算机的基本设计思想是什么？

（1）计算机应由运算器、控制器、存储器、输入设备和输出设备五大基本部件组成。

（2）采用二进制，计算机内部采用二进制来表示程序和数据。

（3）存储程序，将编好的程序和原始数据事先存入存储器中。

（4）程序控制，计算机在工作时从存储器取出指令并加以执行，自动完成计算任务。

（5）指令的执行是顺序的，即一般按照指令在存储器中存放的顺序执行。

5. 什么是总线？微型计算机采用总线结构有什么优点？

总线是计算机各种功能部件之间传送信息的公共通信线，它是 CPU、内存、输入和输出设备之间传递信息的公用通道，主机的各个部件通过总线相连接，外部设备通过相应的接口电路再与总线相连接，从而形成了计算机硬件系统。

微型计算机的总线结构是一个独特的结构，一个部件只要符合总线标准，就可以连接到采用这种总线标准的系统中，从而使系统功能得到扩展。

6. 微型计算机中共有哪几类总线？简述各类总线的应用场合？

按总线功能或信号类型分有：数据总线、地址总线、控制总线。

按总线的层次结构分为有：

（1）CPU 总线：微机系统中速度最快的总线，主要在 CPU 内部，连接 CPU 内部部件，在 CPU 周围的小范围内也分布该总线，提供系统原始的控制和命令。

（2）局部总线：在系统总线和 CPU 总线之间的一级总线，提供 CPU 和主板器件之间以及 CPU 到高速

外设之间的快速信息通道。

(3) 系统总线：也称为 I/O 总线，是传统的通过总线扩展卡连接外部设备的总线。由于速度慢，其功能已经被局部总线替代。

(4) 通信总线：也称为外部总线，是微机与微机，微机与外设之间进行通信的总线。

7. 定性分析微型计算机总线的性能指标。

微型计算机总线的主要职能是负责计算机各模块间的数据传输，对总线性能的衡量也是围绕这一性能而进行的。性能中最重要的是数据传输率，另外，可操作性、兼容性和性价比也是很重要的技术特征，任何系统的研制和外围模块的开发，都必须服从其采用的总线规范。

具体来说，总线的主要性能指标有以下几项：

(1) 总线宽度：以位数表示。

(2) 数据传输率 Mb/s：是总线工作频率与总线宽度的字节数之积。

(3) 时钟同步/异步：总线中与时钟同步工作的称为同步总线，与时钟不同步工作的称为异步总线，这取决于数据传输时源模块与目标模块间的协议约定。

(4) 信号线数：这是地址总线、数据总线和控制总线线数的总和。信号线数和系统的复杂程度成正比关系。

(5) 负载能力：以系统中可以连接的扩展电路板数表示。

(6) 总线控制方法：包括突发传输、并发工作、自动配置、仲裁方式、逻辑方式、中断方式等内容。

(7) 扩展电路板尺寸：这项指标对扩展电路板生产厂家很重要。

(8) 其它指标：电源是 5V 还是 3V，能否扩展 64 位宽度等。

8. 并行总线和串行总线的特点？

并行总线：优点是高速、高效；缺点是通信距离短，主要用于系统的内部总线。

串行总线：优点是通信距离远、接口简单；缺点是速度慢，主要用于系统的外部总线。

9. 并行总线的类型及特点？

并行总线由地址总线、数据总线、控制总线组成，分为同步、异步、半同步三种类型，特点是：

同步并行总线：总线时序以同一个总线时钟为参考，总线上的信息传送均以同步时钟为基准，优点是速度较快且容易实现；缺点是无法兼容总线上各种不同响应速度的设备。

异步并行总线：总线时序没有绝对的时间关系，只有相对的时间关系，总线上的信息传送由握手信号线决定，优点是可根据总线上设备的响应速度调整总线速度；缺点是速度较慢且控制复杂。

半同步并行总线：综合同步和异步并行总线的优点，克服其缺点，是目前微机系统总线的主流，ISA、PCI 都是半同步并行总线。

10. 串行总线的类型及特点？

串行总线分为同步串行总线和异步串行总线两类，它们的共同点是信息按位方式在单线上顺序传送，特点是：

同步串行总线：在信息传输中，字节内位与位之间，字节与字节之间与同步时钟有严格的时间关系。

异步串行总线：在信息传输中，字节内位与位之间，有相对的时间关系，但字节与字节之间无任何时间关系。

11. 什么是 PCI 总线？什么是 USB 总线？

(1) PCI 总线是微处理机机箱内底板各个插件板的一种数据传输标准，是 32/64 位标准总线。PCI 总

线是同步且独立于微处理器的，具有即插即用的特性，允许任何微处理器通过桥接口连接到 PCI 总线上。

(2) USB 总线是系统与系统之间、系统与外设之间的信息通道，USB 总线已成为目前电脑中的标准扩展接口，支持设备的即插即用的特性。

12. 简述 PCI 总线的特点。

(1) 高性能，低成本。

(2) 通用性强，使用方便。

(3) 与处理器无关，具有 32 位和 64 位数据总线，采用集中式总线仲裁、支持多处理器系统，通过桥电路兼容 ISA/EISA 总线，具有即插即用的自动配置能力。

(4) 可靠性高、数据完整性好。

13. 简述 USB 总线的特点。

(1) 即插即用，是开放性的工业标准。

(2) 可动态连接和重新配置外设，支持热插拔功能。

(3) 允许多台设备同时工作。

(4) 可以向 USB 总线上的设备供电，总线上的设备也可以自备电源。

(5) 通讯协议支持等时数据传输和异步消息传输的混合模式。

(6) 支持实时语音、音频和视频数据传输。

第二章 微型计算机的组成电路（16 道）

1. 简述半导体存储器的主要分类？说明它们的用途和区别。

根据存取方式的不同，主要分为随机存取存储器 RAM（静态 RAM，动态 RAM）和只读存取存储器 ROM（不可编程掩膜 MROM，可编程 PROM，紫外光擦除可编程 EPROM，电擦除可编程 EEPROM）。RAM 在程序执行过程中，能够通过指令随机地对其中每个存储单元进行读写操作，RAM 中存储的信息在断电后会丢失，是一种易失性存储器，主要用来存放正在执行的程序、原始数据、中间或运算结果、与 CPU 或外设交换的信息等。ROM 在程序执行过程中，只能对其进行读操作，不能随机地进行写操作，ROM 中存储的信息在断电后不会丢失，是一种非易失性存储器，主要用来存放固定不变的程序、汉字字型库、字符及图形符号等。

根据制造工艺的不同，随机存取存储器 RAM 主要有双极型和 MOS 型两类。双极型存储器具有存取速度快、集成度低、功耗大、成本高等特点，适用于高速缓冲存储器。MOS 型存储器具有集成度高、功耗小、成本低等特点，适用于内存储器。

2. 简述半导体存储器的主要技术指标？

(1) 存储容量。存储器可以存储的二进制信息总量称为存储容量。

(2) 存储速度。存储器的存储速度可以用两个时间参数表示，一个是“存取时间”，定义为从启动一次存储器操作到完成该操作所经历的时间；另一个是“存储周期”，定义为启动两次独立的存储器操作之间所需的最小时间间隔。

(3) 可靠性。存储器的可靠性用平均故障间隔时间 MTBF 来衡量。MTBF 越长，可靠性越高。

(4) 性价比。这是一个综合性指标，性能主要包括上述三项指标：存储容量、存储速度和可靠性，对不同用途的存储器有不同的要求。

3. 简述半导体存储器系统的层次结构及各层存储部件的特点。

(1) 为解决容量、速度和价格的矛盾，存储系统采用金字塔型层次结构，单位价格和速度自上而下逐层减少，容量自上而下逐层增加。存储系统的各层存储部件自上而下依次是：CPU 寄存器、高速缓存器、主存储器（RAM/ROM），辅助存储器（磁盘、光盘）等。

(2) CPU 寄存器、高速缓存器集成在 CPU 芯片上，对用户来说，是透明的，它们用于暂存主存储器和处理器交互的数据，以减少频繁读取主存而影响处理器速度；主存储器可以直接和处理器交换数据，而辅助存储器必须经过主存储器，然后才可以与处理器交换数据。

4. 简述 CPU 与 Cache、主存和外存的关系。

Cache、主存和外存为当前 PC 机的三级存储体系结构，CPU 首先访问速度最快 Cache，而 Cache 的数据由主存提供，称 Cache 中的数据为主存中数据的映射，而主存中的数据从速度最慢的外存获得。采用三级存储体系结构后，可大大提高 CPU 的工作效率。

5. 比较 SRAM 和 DRAM 的特点。

SRAM 的记忆单元是双稳态触发器，只要电源不切断，保存的信息就不会丢失，SRAM 存取速度快，集成度低，功耗大，一般用来组成高速缓冲存储器。

DRAM 的记忆单元是电容，因为电容上的电荷容易衰减，造成存储信息丢失，因此需要定期给 DRAM 的电容补充电荷以保证信息不丢失，也就是常说的“刷新”，DRAM 集成度高，功耗小，但存取速度较慢，一般用来组成大容量的主存系统。

6. DRAM 为什么要刷新？如何进行刷新？

DRAM 的记忆单元是电容，因为电容上的电荷容易衰减，造成存储信息丢失，因此需要定期给 DRAM 的电容补充电荷以保证信息不丢失，也就是常说的“刷新”。

存储系统的刷新控制电路提供刷新行地址，将 DRAM 芯片中的某一行选中刷新，实际上，刷新控制电路是将刷新行地址同时送达存储系统中的所有 DRAM 芯片，所有 DRAM 芯片都在同时进行一行的刷新操作，刷新控制电路设置每次行地址增量，并在一定时间间隔内启动一次刷新操作，这样就能够保证所有 DRAM 芯片的所有存储单元能够得到及时刷新。

7. 2164A 如何利用 8 条地址线在芯片内寻址 64K 单元？

2164A 地址线采用行地址线和列地址线分时工作，2164A 对外部只需引出 8 条地址线。芯片内部有地址锁存器，利用多路开关，由行地址选通信号 \overline{RAS} 把先送来的 8 位地址送至行地址锁存器，由随后出现的列地址选通信号 \overline{CAS} 把后送来的 8 位地址送至列地址锁存器，这样就能实现利用 8 条地址线在芯片内寻址 64K 单元。

8. 2164A 数据读写是如何实现的？用到了哪些引脚？

2164A 数据的读和写是分开的，由 \overline{WE} 信号控制读写。当 \overline{WE} 为高电平时，实现读，即所选中单元的内容经过三态输出缓冲器在 DOUT 引脚读出。当 \overline{WE} 为低电平时，实现写，即所选中单元的内容经过三态输出缓冲器在 DIN 引脚读入。

9. ROM、PROM、EPROM 和 EEPROM 在使用上各有什么特点？

(1) ROM 为只读存储器，制造时数据就已经固化好，使用中不可以改变。

(2) PROM 为一次可编程只读存储器，出厂时内容为空白，只可以写入一次数据。

(3) EPROM 为可擦除、可再编程只读存储器，出厂时内容为空白，写入数据后可以使用紫外线照射擦除，可以多次写入和擦除。

(4)EEPROM 为可擦除、可再编程只读存储器，出厂时内容为空白，写入数据后可以使用电信号擦除，可以多次写入和擦除。

10. 简述高速缓冲存储器 Cache 为什么能够实现高速的数据存取？

高速缓冲存储器 Cache 是根据程序的局部性原理来实现高速的数据存取。即在一个较小的时间间隔内，程序所要用到的指令或数据的地址往往集中在一个局部范围内，因而对局部范围内的地址访问频繁，而对局部范围外的地址则访问甚少的现象被称为程序的局部性原理。如果把正在执行的指令地址附近的一小部分指令或数据，即当前最活跃的程序或数据从主存成批调入 Cache，供 CPU 在一段时间内随时使用，就能在一定程度上大大减少 CPU 访问主存的次数，从而加速程序的运行。

11. 简述虚拟存储器的含义是什么？

虚拟存储器是以存储器访问的局部性为基础，建立在主存—辅存物理体系结构上的存储管理技术。在存储系统中，由于主存容量不能满足用户的需要，因而引入辅存作为后援。即辅存做主存用，扩大编程者的使用空间。

12. 什么是片内地址线？什么是片选地址线？

片内地址线：为对存储器芯片中的每一个存储单元都能进行寻址，CPU 为存储器芯片提供的地址线。

片选地址线：CPU 的地址线中，除了提供给存储器芯片的片内地址线外，其余均为片选地址线，用于确定某存储器芯片在存储系统空间中的具体存储范围。

13. 在存储器系统中，实现片选控制的全译码、部分译码和线选三种方法各有何特点？

(1) 全译码法，优点：每组芯片的地址范围是唯一确定的，而且是连续的，也便于扩展，不会产生地址重叠区。缺点：对译码电路要求较高。

(2) 部分译码法，优点：部分译码较全译码简单。缺点：存在地址重叠区。

(3) 线选法，优点：不需要地址译码、线路简单。缺点：不能充分利用系统的存储空间。

14. 触发器、寄存器及存储器之间有什么关系？

触发器是计算机记忆装置的基本单元，一个触发器能够存储一位二进制信息。寄存器是由触发器组成的，一个触发器就是一个一位寄存器，多个触发器就可以组成一个多位寄存器。存储器是由大量寄存器组成的，其中每一个寄存器被称为一个存储单元，它可以存放一个有独立意义的二进制信息。

15. 简述存储器与寄存器的异同。

相同点：存储器和寄存器均用于存放二进制信息。不同点：寄存器为 CPU 内部的存储单元，数量较少，每个寄存器都有专门用途和名称，编程时用寄存器名访问，例如 MOV AX, BX；存储器为 CPU 外部的存储单元，数量较大，每个存储单元都有地址，可存放指令和数据，编程时用存储单元地址访问，例如 MOV AX, [2000H]。

16. 简述存储器芯片中存储单元数量与存储容量大小的关系。

在存储器芯片中，存储单元数量与芯片的地址线数量有关，若地址线数量为 N，则存储单元数量为 2^N ；存储容量除了与存储单元数量有关外，还与芯片的数据线数量有关，若地址线数量为 N，数据线数量为 M，则存储容量为 $2^N \times M$ 。

第三章 微型计算机的微处理器（30 道）

1. 简述 8086 与 8088 的区别。

CPU 内部的区别：8086 的指令队列缓冲器为 6 字节，8088 为 4 字节；CPU 数据总线的区别：8086 的数据总线宽度为 16 位，8088 为 8 位；CPU 控制线的区别：因 8086 可一次进行 16 位数据的操作，可用控制线 $\overline{\text{BHE}}$ 和地址线 A0 完成对奇偶存储库的选择，8088 一次只能对 8 位数据的操作，无控制线 $\overline{\text{BHE}}$ 的功能。8086 与 8088 比较，存储器和 I/O 选择控制线的控制电平相反。

2. 在 8086/8088 系统中，请分组说明有哪些寄存器？

8 个通用寄存器 AX、BX、CX、DX、SP、BP、DI、SI，都可以用来暂存参加运算的数据或中间结果，但它们又有各自专门的用途。其中，4 个数据寄存器：AX 为累加寄存器，BX 为基址寄存器，CX 为计数寄存器，DX 为数据寄存器。4 个专用寄存器：SP 为堆栈指针寄存器，BP 为基址指针寄存器，DI 为目的变址寄存器，SI 为源变址寄存器。

4 个段寄存器 CS、DS、SS、ES，CS 代码段寄存器，用于装代码段的起始地址；DS 数据段寄存器，用于装数据段的起始地址；SS 堆栈段寄存器，用于装堆栈段的起始地址；ES 附加段寄存器，用于装附加段的起始地址。

2 个控制寄存器 FR、IP，用来存放有关的状态信息和控制信息。例如，状态标志寄存器 FR 用来存放状态标志和控制标志；指令指针寄存器 IP 用来存放下一条要取指令的有效地址。

3. 在 8086/8088 系统中，简述段与段寄存器的关系。

段在内存中，内存分段。段寄存器在 CPU 中，代码段的段基址放在 CS 中，数据段的段基址放在 DS 中，堆栈段的段基址放在 SS 中，附加段的段基址放在 ES 中。

4. 在 8086/8088 系统中，状态标志和控制标志分别有哪些？

状态标志位反应了当前运算和操作结果的状态条件，可作为程序控制转移与否的依据，它们分别是 CF、OF、ZF、SF、AF 和 PF。

控制标志位用来控制 CPU 的操作，由指令进行置位和复位，它们分别是 IF、DF 和 TF。

5. 在 8086/8088 系统中，6 个状态标志位的作用是什么？

6 个状态标志位为 CF、OF、ZF、SF、AF 和 PF。CF 是无符号数运算时的进位或借位标志，无进位或借位时 CF=0，有进位或借位时 CF=1；OF 为有符号数运算时的溢出标志，无溢出时 OF=0，有溢出时 OF=1；ZF 是两数运算时的值 0 标志，运算结果不为 0，ZF=0，运算结果为 0，ZF=1；SF 是有符号数运算时运算结果符号的标志，运算结果为正时 SF=0，运算结果为负时 SF=1；AF 是辅助进位标志，若 D3 位到 D4 位无进位时（或 D4 位到 D3 位无借位时），AF=0，若 D3 位到 D4 位有进位时（或 D4 位到 D3 位有借位时），AF=1；PF 是运算结果的奇偶校验标志，若运算为奇个 1，则 PF=0，若运算为偶个 1，则 PF=1。

6. 在 8086/8088 系统中，3 个控制标志位的作用是什么？

3 个控制标志位是 IF、DF 和 TF。IF 是可屏蔽中断中断允许控制位，当 IF=0 时，有可屏蔽中断请求但无中断响应产生，当 IF=1 时，有可屏蔽中断请求必有中断响应产生；DF 是数据串操作时的自动增量方向控制位，当 DF=0 时，地址增量方向为自动加，当 DF=1 时，地址增量方向为自动减；TF 是指令单步调试控制位，当 TF=0 时，无指令单步调试操作，当 TF=1 时，有指令单步调试操作。

7. 指令队列的作用是什么？

在执行指令的同时，从内存中取了下一条指令或几条指令，取来的指令放在指令队列中，这样它就不需要像以往的计算机那样让 CPU 重复的做取指和执行的工作了，从而提高 CPU 的利用率。

8. 简述堆栈操作原理。

堆栈是一段特殊定义的存储区，用于存放 CPU 在堆栈操作时的数据。在执行堆栈操作前，需先定义堆栈段 SS、堆栈深度和堆栈栈顶指针 SP。数据的入栈出栈操作类型均为 16 位，入栈操作时，栈顶指针值先自动减 2 ($SP=SP-2$)，然后 16 位数据从栈顶处入栈；出栈操作时，16 位数据先从栈顶处出栈，然后栈顶指针值自动加 2 ($SP=SP+2$)。

9. 什么是基本总线周期，什么是扩展总线周期？

基本总线周期由 4 个时钟周期组成，分别为 T1、T2、T3 和 T4。在 T1 时刻，CPU 的地址/数据复用线上发出地址信息，用于存储器单元或 I/O 端口的寻址。在 T2~T4 期间，在 CPU 的地址/数据复用线和存储器单元或 I/O 端口之间实现数据传送。扩展总线周期是在基本总线周期的基础上，根据特殊要求加入等待周期 T_w 和空闲周期 T_t 。为了保证高速 CPU 与低速存储器或 I/O 接口间的数据读写，在控制线 READY 的控制下，可在 T3 与 T4 间插入一个或多个等待周期 T_w 。当 CPU 暂时不需要经总线传送数据时，可在 T4 后插入一个或多个等待周期 T_t 。

10. 什么是地址锁存器？为什么要用地址锁存器？用什么信号控制锁存？

地址锁存器就是一个暂存器，它根据控制信号的状态，将总线上的地址信息暂存起来。

由于 CPU 受芯片封装的限制，只有 40 个管脚，所以地址线和数据线只能采用复用的方式共同使用某些管脚。对存储器进行读取数据或写入数据时，存储器要求在这个过程中地址信息必须稳定提供给存储器，而由于 CPU 地址线和数据线是复用的，就不可能在同一时刻具有地址和数据两种功能，这就需要在 CPU 提供地址信息时，将地址信息锁存起来，以保证下一个时刻，当这些复用的管脚起着数据线的功能时，存储器仍然有正确的地址信息。

通过 CPU 送出的 ALE 高电平信号来控制锁存的。锁存的信息包括这些复用管脚的地址 ($A_{19} \sim A_0$) 和 \overline{BHE} 等信号。

11. 什么是最小模式？什么是最大模式？

最小模式：在该系统中只有一个微处理器，所有的总线控制信号都由 8086 直接产生，因此，系统中总线控制电路被减到最少。

最大模式：在该系统中包含两个或者多个微处理器，其中一个主处理器就是 8086，其它处理器为协处理器，是协助主处理器工作的。它作用在中等规模或大型的 8086 系统中。一般情况下和 8086 配合的协处理器有两个，一个是数值运算协处理器 8087，一个是输入/输出协处理器 8089。

12. CPU 从功能上可分为哪两部分？这样设计的优点是什么？

从功能上分为两大部分，一是执行部件 (EU)，二是总线接口部件 (BIU)。EU 负责执行所有的指令，向 BIU 提供指令执行的结果数据和地址，并对通用寄存器和标志寄存器进行管理。BIU 负责执行外部总线周期，也就是负责 CPU 与存储器和外设之间的信息交换。

CPU 分成两部分后，BIU 负责取指令，EU 负责执行指令，它们之间既互相独立又互相配合，即实现了取指令和执行指令的并行工作，大大提高了 CPU 和总线的利用率，从而提高了指令的处理速度。

13. 简述流水线技术，怎样实现了最简单的指令流水线？

流水线是指在程序执行时，多条指令重叠进行操作的一种并行处理实现技术。它的并行处理是指完成一条指令的各个功能部件在同一时间上可以分别为多条指令的不同部分进行工作，通过提高各部件的利用率来提高指令的平均执行速度。它把一个重复的过程分解为若干个子过程，每个子过程由专门的功能部件来实现，将多个处理过程在时间上错开，依次通过各功能段，这样，每个子过程就可以与其它子过程并

行进行。

CPU 分成两部分后, BIU 负责取指令, EU 负责执行指令, 它们之间既互相独立又互相配合, 即实现了取指令和执行指令的并行工作, 大大提高了 CPU 和总线的利用率, 从而提高了指令的处理速度。也就是说, 在 EU 对一个指令进行执行时, BIU 可以同时对外部指令进行读取, 这就是最简单的指令流水线技术。

14. 在 8086 系统中, 简述在最小工作模式下, 如何响应一个总线请求?

外部总线主控模块经 HOLD 引线向 8086 发出总线请求信号, 8086 在每个时钟周期的上升沿采样 HOLD 引线, 若发现 HOLD=1, 则在当前总线周期结束时, 发出总线请求的响应信号 HLDA, 8086 使地址、数据及控制总线进入高阻状态, 让出总线控制权, 从而完成响应过程。

15. 什么是统一编址, 独立编址? 各有何特点?

统一编址: 存储器单元地址和 I/O 端口地址在同一个地址空间中分配。由于 I/O 端口地址占用存储器单元地址, 所以减少了存储器的寻址空间, 访问存储器单元和 I/O 端口可用相同的指令。

独立编址: 存储器单元地址和 I/O 端口地址在不同的地址空间中分配。存储器和 I/O 端口都有独立且较大的寻址空间, CPU 需要用门的控制线来识别是访问存储器还是访问 I/O 端口, 访问存储器单元和 I/O 端口要用不同的指令。

16. 什么是逻辑地址、偏移地址、物理地址?

逻辑地址是在程序中对存储器地址的一种表示方法, 由段地址和段内偏移地址两部分组成; 偏移地址是指段内某个存储单元相对该段首地址的差值, 是一个 16 位的二进制代码; 物理地址是 8086 芯片引线送出的 20 位地址码, 用来指出一个特定的存储单元。

17. 在 8086 系统中, 存储器的逻辑地址是由哪两部分组成? 物理地址由何器件如何生成? 每个段的逻辑地址与寄存器之间有何对应关系?

存储器的逻辑地址由段地址和段内偏移地址两部分组成。

存储单元的物理地址由地址加法器生成, 寻址时, CPU 首先将段地址和段内偏移地址送入地址加法器, 地址加法器将段地址左移 4 位并与段内偏移地址相加, 得到一个 20 位的物理地址。

代码段的段地址在 CS 寄存器中, 段内偏移地址在 IP 寄存器中。数据段的段地址在 DS 寄存器中, 段内偏移地址可能在 BX、BP、SI 或 DI 寄存器中。堆栈段的段地址在 SS 寄存器中, 段内偏移地址在 SP 寄存器中。扩展段的段地址在 ES 寄存器中, 段内偏移地址可能在 BX、BP、SI 或 DI 寄存器中。

18. 在 8086 系统中, 存储器为什么要分段? 哪几个段与分段有关?

8086CPU 提供了 20 位的地址总线, 可寻址 1MB 存储空间, 而 8086CPU 内部寄存器都是 16 位的, 寻址能力是 64KB; 因此, 为能实现对存储器寻址 20 位的物理地址, 可将 1MB 的存储空间划分为若干个逻辑段, 每个逻辑段可寻址 64KB; 各逻辑段之间可以部分、完全覆盖, 连续、不连续等非常灵活。

与分段有关的寄存器有: CS (代码段寄存器)、DS (数据段寄存器)、SS (堆栈段寄存器)、ES (附加段寄存器)。

19. 在 8086 系统中, 存储器是如何组织的, 它是如何与处理器总线连接的?

由于在 8086 系统中外部数据总线是 16 位的, 而存储器又是按字节编址的, 所以把 1MB 的存储空间分为两个 512KB 的存储体, 分别命名为偶地址存储体和奇地址存储体。偶地址存储体的数据线与系统数据总线低八位相连, 用 $A_0=0$ 作为选通信号。奇地址存储体的数据线与系统数据总线高八位相连, 用 $\overline{BHE}=0$ 作为选通信号。CPU 可以访问任何一个存储体, 读写一个字节, 也可以同时访问两个存储体, 读写一个字。

20. 在 8086 系统中，地址线 A0 为什么不能参加存储器的片内选择线？

由于在 8086 系统中外部数据总线是 16 位的，而存储器又是按字节编址的，所以把 1MB 的存储空间分为两个 512KB 的存储体，分别命名为偶地址存储体和奇地址存储体。偶地址存储体的数据线与系统数据总线低八位相连，用 $A_0=0$ 作为选通信号。奇地址存储体的数据线与系统数据总线高八位相连，用 $\overline{BHE}=0$ 作为选通信号。CPU 可以访问任何一个存储体，读写一个字节，也可以同时访问两个存储体，读写一个字。所以 A_0 就不能参加存储器的片内选择线，而是作为存储体的地址译码选择线。

21. 在 8086 系统中， \overline{BHE} 信号的作用是什么？试说明当起始地址为奇地址、偶地址、一次读写一个字节和一个字时， \overline{BHE} 和 A_0 的状态？

\overline{BHE} 信号的名称是高 8 位数据总线允许引脚。若 \overline{BHE} 为 0，则表示对一个字进行操作，即高 8 位有效，若 \overline{BHE} 为 1，则表示对一个字节进行操作，即高 8 位无效。

当起始地址为奇地址时，一次读写一个字节时， \overline{BHE} 为 0， A_0 状态为 1。

当起始地址为偶地址时，一次读写一个字节时， \overline{BHE} 为 1， A_0 状态为 0。

当起始地址为奇地址时，一次读写一个字时， $\left\{ \begin{array}{l} \text{第一个总线周期发送，}\overline{BHE}\text{为 0，}A_0\text{状态为 1。} \\ \text{第二个总线周期发送，}\overline{BHE}\text{为 1，}A_0\text{状态为 0。} \end{array} \right.$

当起始地址为偶地址时，一次读写一个字时， \overline{BHE} 为 0， A_0 状态为 0。

22. 在 8086 系统中，什么是存储器的规则字和非规则字？CPU 对一个规则字和非规则字读写时，有什么差别？

规则字是指在存储器中存储的起始地址为偶数（地址最低位 A_0 为 0）的字数据，非规则字是指在存储器中存储的起始地址为奇数（地址最低位 A_0 为 1）的字数据。

规则字读写需要一个总线周期，发送 \overline{BHE} 为 0， A_0 为 0，一个总线周期读写一个字；非规则字读写需要两个总线周期，第一个总线周期读取偶存储体数据，发送 \overline{BHE} 为 0， A_0 为 1，取得高 8 位数据，第二个总线周期读取奇存储体数据，发送 \overline{BHE} 为 1， A_0 为 0，取得低 8 位数据。

23. 在 8086 系统中，NMI、INTR、 \overline{INTA} 、ALE、HOLD、HLDA 引脚的名称各是什么？

NMI 是不可屏蔽中断请求信号，INTR 是可屏蔽中断请求信号， \overline{INTA} 是中断响应信号，ALE 是地址锁存允许信号，HOLD 是总线请求信号，HLDA 是总线请求响应信号。

24. 在 8086 系统中，控制线 ALE 的作用是什么？

控制线 ALE 的作用是在总线周期 T1 时，完成地址/数据复用线上地址信息的分离。ALE 用于控制锁存器的锁存控制端，在 T1 时 ALE 输出高电平锁存地址信息，在 T2~T4 期间 ALE 输出低电平保持地址信息。

25. 在 8086 系统中，控制线 \overline{DEN} 、 DT/\overline{R} 的作用是什么？

控制线 \overline{DEN} 的作用是完成对双向数据缓冲器的控制。当 $\overline{DEN}=0$ 时，双向数据缓冲器片选有效，CPU 的地址/数据复用线与数据总线连接有效。

控制线 DT/\overline{R} 的作用是控制双向数据缓冲器中数据传送的方向。当 $DT/\overline{R}=0$ 时，数据从数据总线上流入 CPU 内。当 $DT/\overline{R}=1$ 时，数据从 CPU 内流出数据总线上。

26. 在 8086 系统中，控制线 \overline{RD} 、 \overline{WR} 的作用是什么？

控制线 \overline{RD} 、 \overline{WR} 的作用是完成 CPU 对存储器单元或 I/O 端口的数据读写控制。当 $\overline{RD}=0$ 且 $\overline{WR}=1$ 时，CPU 经数据总线从选中的存储器单元或 I/O 端口中读取数据，当 $\overline{RD}=1$ 且 $\overline{WR}=0$ 时，CPU 经数据总线向选中

的存储器单元或 I/O 端口中写入数据。

27. 在 8086 系统中，控制线 $\overline{M}/\overline{IO}$ 的作用是什么？

控制线 $\overline{M}/\overline{IO}$ 的作用是确定在某一时刻 CPU 对存储器操作还是对 I/O 接口操作。当 $\overline{M}/\overline{IO}=0$ 时，CPU 对 I/O 接口操作有效，当 $\overline{M}/\overline{IO}=1$ 时，CPU 对存储器操作有效。

28. “8086 执行了一个总线周期”是指 8086 做了哪些可能的操作？在一个典型的读存储器总线周期中，地址信号、ALE 信号、RD 信号、数据信号分别在何时产生？

“8086 执行了一个总线周期”是指：①、8086 可能从片外的存储器取指令；②、8086 可能对片外的存储器或 I/O 接口进行了一次读/写数据的操作。

在一个典型的读存储器总线周期中，地址信号在 T1 周期内产生，ALE 信号在 T1 周期内产生，RD 信号在 T2 周期内产生，数据信号一般在 T3 周期内产生，若存储器在 T3 内来不及提供数据，8086 会在 T3 周期后插入等待周期 TW，存储器将在某 TW 中给出数据。

29. 论述指令周期、总线周期和时钟周期间的关系？如果在存储器读周期时，存储器的速度较慢，无法与 CPU 匹配应采取什么措施？

指令周期是指执行一条指令所需要的时间，不同指令的指令周期是不等长的。总线周期是指 CPU 与存储器或外设进行一次读/写操作所需要的时间。时钟周期（T 状态）是指时钟脉冲的重复周期，是 CPU 的时间基准，由计算机的主频决定。一个指令周期通常由一个或若干个总线周期组成。对 8086 而言，一个总线周期至少由 4 个时钟周期组成。

如果存储器的存取速度较慢，可设计一个等待电路向 CPU 发出请求延长总线周期的 READY 信号，CPU 接到此信号后就在 T3 和 T4 之间插入若干个等待周期 TW。

30. 什么是引脚的分时复用？请说出 8086CPU 有哪些引脚是分时复用的？

8086CPU 的数据线和地址线是分时复用的，所以常把 8086CPU 的总线称为多路总线，即某一时刻总线上出现的是地址，另一时刻总线上出现的是数据，正是这种引脚的分时复用的方法，才能使 8086CPU 用 40 条引脚实现 20 位地址、16 位数据及众多的控制信号和状态信号的传输。

8086CPU 的分时复用的引脚有：地址/数据复用引脚是：AD15~AD0，在总线周期的 T1 状态，传送地址信息，在其它状态则传送数据信息；地址/状态复用引脚是：A19/S6~A16/S3，在总线周期的 T1 状态，传送地址的高 4 位，在其它状态则传送状态信息。

第四章 微型计算机的指令系统（11 道）

1. 指令的寻址方式有哪几类？用哪一种寻址方式的指令执行速度最快？

数据操作数的寻址方式有 7 种，分别为：立即数寻址，直接寻址，寄存器寻址，寄存器间接寻址，寄存器相对寻址，基址加变址寻址和相对基址加变址寻址。

其中寄存器寻址的指令执行速度最快。

2. 直接端口寻址，间接端口寻址的特点？

8086CPU 在进行 I/O 端口访问时，有效地址线为 A15~A1，即 16 条地址线可访问 64K 个字节 I/O 端口。若仅用 16 条地址线的低 8 位地址进行访问 I/O 端口称为直接端口寻址，若 16 条地址线全用于访问 I/O 端口称为间接端口寻址。在直接端口寻址操作中，8 位地址操作数可直接出现在 IN/OUT 指令中。在间接端口寻址操作中，16 位地址操作数应先赋给寄存器 DX，IN/OUT 指令中仅出现间接地址 DX。

3. 存储器寻址中间址寄存器的使用特点？

存储器寻址中可用的间址寄存器有 BX、BP、SI、DI，当存储器间址寻址时，BX、SI、DI 对应的缺省段是 DS；BP 对应的缺省段是 SS。当存储器基址变址寻址时，BX+SI、BX+DI 对应的缺省段是 DS；BP+SI、BP+DI 对应的缺省段是 SS。

4. MOV 指令中源、目的操作数的禁用特点？

立即数不能作目的操作数；寄存器 CS、IP 不能作目的操作数；源、目的操作数不能同时为存储器操作数；源、目的操作数不能同时为段寄存器操作数；立即数不能赋给段寄存器。

5. CMP 指令对状态标志位的影响？

当两操作数比较是否相等时，影响状态标志位 ZF，当两数相等时 ZF=1，反之 ZF=0。当两操作数比较大小时，无符号数比较和有符号数比较会影响不同的状态标志位。若无符号数比较，当目的操作数小于源操作数时 CF=1，反之 CF=0。若有符号数比较，当目的操作数小于源操作数时 $OF \oplus SF=1$ ，反之 $OF \oplus SF=0$ 。

6. MUL、DIV 指令中专用寄存器的使用？

在 $8 \times 8 \rightarrow 16$ 模式中，专用被乘数寄存器是 AL、专用积寄存器是 AX。

在 $16 \times 16 \rightarrow 32$ 模式中，专用被乘数寄存器是 AX、专用积寄存器是 DX、AX。

在 $16 \div 8 \rightarrow 8$ 模式中，专用被除数寄存器是 AX、专用商/余寄存器是 AL、AH。

在 $32 \div 16 \rightarrow 16$ 模式中，专用被除数寄存器是 DX、AX、专用商/余寄存器是 AX、DX。

7. AAM、AAD 指令的使用特点？

AAM 是 BCD 乘法调整指令，作用是对积进行调整，即将真实的积调整为用 BCD 表示的积，应用时先用 MUL 指令后用 AAM 指令。AAD 是 BCD 除法调整指令，作用是对被除数进行调整，即将用 BCD 表示的被除数调整为真实的被除数，应用时先用 AAD 指令后用 DIV 指令。

8. 串操作指令中的专用寄存器？

专用源操作数串寄存器：串存放段寄存器 DS、串指针寄存器 SI。

专用目的操作数串寄存器：串存放段寄存器 ES、串指针寄存器 DI。

专用串操作数数量计数器 CX。

专用串指针自动移动方向控制标志 DF。

9. 串操作中控制标志 DF、指令尾符 B、W 的使用特点？

当控制标志 DF=0 时，串指针 SI、DI 为自动增址方式。

当控制标志 DF=1 时，串指针 SI、DI 为自动减址方式。

当尾符为 B 时，为字节串操作，自动增、减址量为 ± 1 。

当尾符为 W 时，为字串操作，自动增、减址量为 ± 2 。

10. JMP 指令对目的 CS：IP 的影响？

当 JMP 指令为段内无条件转移时，目的 CS 不变仅影响目的 IP。

当 JMP 指令为段间无条件转移时，将影响目 CS 和目的 IP。

11. 指令 CALL 和 INT 的相同点和不同点？

相同点均为调用一段功能程序，且都需要保护断点（现场）和恢复现场。不同点是保护断点（现场）和恢复现场的内容不同，若 CALL 为近调用则仅需保护和恢复 IP。若 CALL 为远调用则仅需保护和恢复 CS、IP。而 INT 需保护和恢复 PSW、CS、IP。

第五章 微型计算机的汇编程序（09 道）

1. 什么是变量，变量有哪三个属性？

变量通常指存放在存储单元里的值，在程序运行中是可以修改的。所有的变量都具有以下三个属性：段属性、偏移属性、类型属性。

2. 在汇编语言程序中，变量和标号有几种属性，各有什么含义

各有 3 种属性，分别是段属性、偏移属性和类型属性。

段属性是指变量和标号所在的段对应着段寄存器的内容。偏移属性是指变量和标号在相应段内的偏移地址。变量的类型属性是指字节、字、双字等数据宽度。标号的类型属性是指远（FAR）、近（NEAR）。

3. 指令助记符、伪指令、标识符在汇编语言中的作用？

汇编语言的源程序设计中，用指令助记符表示指令机器码，用伪指令管理整个源程序，用标识符表示存储器真实地址。源程序经汇编后，指令助记符被翻译为指令机器码，伪指令无任何翻译结果，而标识符则用真实地址替换。

4. 解释指令助记符、汇编语言、汇编语言程序、汇编程序的含义？

指令助记符：人们采用便于记忆并能描述指令功能的符号来表示指令机器码，该符号称指令助记符。

汇编语言：用指令助记符表示的指令以及使用它们编写程序的规则就形成汇编语言。

汇编语言程序：用汇编语言书写的程序就是汇编语言程序。

汇编程序：汇编语言程序要翻译成机器语言程序才可以由处理器执行，这个翻译的过程称为“汇编”，完成汇编工作的程序就是汇编程序（MASM.EXE）。

5. 在计算机上运行汇编语言程序的步骤是什么？

- （1）用编辑文件如 EDIT 编辑源文件，形成 .ASM 文件。
- （2）用汇编程序（MASM.EXE）把 .ASM 源文件汇编成目标文件 .OBJ。
- （3）用连接程序（LINK.EXE）把 .OBJ 文件转换成 .EXE 可执行文件。
- （4）运行可执行文件 .EXE。
- （5）若有错，使用 DEBUG 进行调试。

6. 比较宏指令和子程序，它们有何异同？它们的本质是什么？

两者的相同点为：都是用来处理在编程过程中多次使用的功能程序的方法，两者均能简化源程序。

两者的区别在于：

（1）宏调用通过宏指令名进行，汇编时，随时调用随时展开，并不简化目标程序；子程序调用是在程序运行期间通过执行 CALL 指令调用，代码只在目标程序中出现一次，所以也简化了目标程序。

（2）宏调用时的参数由汇编程序通过实参转换成形参的方式传递，具有很大的灵活性，宏定义中允许设置若干形式参数代替数值、指令、寄存器、各种字符串等，宏调用时可用程序所需要的实际参数来替换，使编程人员感觉非常灵活；而子程序的参数传递要麻烦得多。

（3）宏调用在汇编时完成，不需要额外的时间开销；子程序调用和子程序返回都需要时间，还涉及堆栈操作。

综上，故若优先考虑速度，用宏指令；若优先考虑存储空间，用子程序。

7. 什么是系统调用，如何进行系统调用？

系统调用就是 DOS 调用中的 INT 21H 调用。

其过程是：（1）置入口参数；（2）将系统功能号送入 AH；（3）执行系统调用 INT 21H。

8. 伪指令 END、ENDS、ENDP、ENDM 的作用？

END 为汇编源程序结束伪指令，出现在汇编源程序的最后一行。ENDS 为段定义结束伪指令，出现在段定义的最后一行。ENDP 为过程定义结束伪指令，出现在过程定义的最后一行。ENDM 为宏定义结束伪指令，出现在宏定义的最后一行。

9. 伪指令 DB 和 BYTE 的使用特点？

DB 为字节数据类型定义伪指令，用于在数据段中定义地址变量的数据类型；BYTE 为字节数据类型的说明伪指令，用于在代码段中对指令操作数进行数据类型说明。

第六章 微型计算机的中断技术（10 道）

1. 什么是中断？中断技术给计算机系统带来了什么作用？

中断概念：当计算机执行正常程序时，系统中出现某些异常情况或特殊请求，CPU 暂停它正在执行的程序，而转去处理所发生的事件，CPU 处理完毕后，自动返回到原来被中断的程序继续运行，则称为中断。

中断作用：

- （1）主机与外设并行工作。
- （2）实现实时处理。
- （3）硬件故障处理。
- （4）实现多道程序和分时操作。

2. 什么是中断源？识别中断源有哪些方法？

中断源概念：引起中断的原因或发出中断申请的来源，则称为中断源。

中断源识别：

（1）每个中断源都有一条中断请求信号线，且固定一个中断服务程序的入口地址，CPU 一旦检测到某条信号线有中断申请，就进入相应的中断服务程序。

（2）向量中断，使用向量中断系统的中断源，除了能输出中断请求信号外，还能在 CPU 响应它的中断请求后输出一个中断向量，CPU 根据这个中断向量能够获得该中断源程序的入口地址，从而为其服务。

3. 为了区别不同的中断，有以下概念，请说明他们的不同。

（1）可屏蔽中断和非屏蔽中断 （2）内部中断和外部中断

（1）受中断标志位控制的中断称为可屏蔽中断；必须立刻响应的中断请求，如电源掉电、机器故障等，且不受中断标志位控制的中断称为非屏蔽中断。

（2）内部中断又称软件中断，是指中断源来自主机内部，如运算出错、程序调试和软件中断等；外部中断又称硬件中断，是指中断源来自主机外部，往往通过 CPU 的中断请求引脚引入主机，如外部设备、实时时钟和硬件故障产生的中断等。

4. 硬件中断和软件中断的区别？

（1）硬件中断由硬件产生，通过 INTR 和 NMI 引脚发送给 CPU，软件中断由软件的中断指令或其它异常产生。

- (2) 硬件中断的中断号由中断控制器提供，软件中断的中断号在指令中提供或隐含。
- (3) 硬件中断具有随机性，软件中断具有确定性。
- (4) 大部分硬件中断需要 CPU 发送响应信号，软件中断不需要。
- (5) 硬件中断除 NMI 之外均可以屏蔽，软件中断不能屏蔽。

5. 简述响应可屏蔽中断的条件？

- (1) 无总线请求，即系统中其它的总线设备没有向 CPU 发出总线请求，无总线竞争。
- (2) 无非屏蔽中断或其它更高级中断源的请求。
- (3) CPU 执行完当前的指令。
- (4) 开中断，即 CPU 允许中断，IF 标志的状态为 1。
- (5) CPU 通过引脚连续发送两个负脉冲，外设接口在接到第二个负脉冲后，发送中断类型码，此时，CPU 才开始处理中断。

6. 一个可屏蔽中断响应周期般包括几个总线周期？主要进行什么操作？

一个可屏蔽中断响应周期一般包括 7 个总线周期。

- (1) 通知发送中断请求的设备，CPU 准备响应中断。
- (2) CPU 接收外设发来的中断类型码。
- (3) 将标志寄存器 PSW 的内容压入堆栈。
- (4) 将 CS 的内容压入堆栈。
- (5) 将 IP 的内容压入堆栈。
- (6) 将中断处理子程序入口地址的偏移量送 IP。
- (7) 将中断处理子程序入口地址的段值送 CS。

7. CPU 响应中断时，为什么要执行两个连续的中断响应周期？

CPU 响应中断时，执行两个连续的中断响应周期，每个响应周期都给出中断响应信号 INTA。第一个总线周期中，通知发送中断请求的设备，CPU 准备响应中断，应该准备好中断类型码，第二个总线周期中，CPU 接收中断类型码，以便得到中断向量，即中断服务程序的入口地址。

8. 外设向 CPU 申请中断，但 CPU 不响应，其原因可能有哪些？

- (1) 该中断请求持续时间太短。
- (2) CPU 未能在当前指令周期的最后一个时钟周期采样到中断请求信号。
- (3) CPU 处于关中断状态。
- (4) 该中断级被屏蔽。

9. 什么是中断类型码？什么是中断向量？什么是中断向量表？它们之间有什么联系？

8086 系统可以处理 256 种中断，为了区别每一种中断，为每个中断安排了一个号码，这个号码称为中断类型码。每一种中断服务程序在内存中的起始地址称为中断向量，以 32 位逻辑地址表示，即为 CS:IP。把所有中断向量存储在内存中的某一个连续的存储区中，这个连续的存储区称为中断向量表。当中断源发出中断请求时，即可通过中断类型码查找该表，找出其对应的中断向量，就可转入相应的中断服务程序。中断向量 CS:IP 在中断向量表中的位置为：中断向量表中偏移量为（中断类型码 $\times 4$ ）的单元中存放 IP 的值，中断向量表中偏移量为（中断类型码 $\times 4 + 2$ ）的单元中存放 CS 的值。

10. 中断时为什么要安排中断优先级？采用软件优先级排队和硬件优先级排队各有什么特点？

CPU 随时可能会收到多个中断源提出的中断请求，因此，为每个中断源分配一级中断优先权，根据它们的高低顺序决定响应的先后顺序。

软件优先级排队：各种中断源的优先权由软件安排。优点是电路比较简单，可以直接修改软件查询顺序来修改中断优先权，不必更改硬件。缺点是当中断源个数较多时，由逐位检测查询到转入相应的中断服务程序所耗费的时间较长，中断响应速度慢，服务效率低。硬件优先级排队：指利用专门的硬件电路或中断控制器对系统中各种中断源的优先权进行排队。优点是中断响应速度快，服务效率高，缺点是电路比较复杂，需要专门的硬件电路，如 8259A 芯片。

第七章 微型计算机的接口技术（22 道）

1. CPU 与外设通信为什么需要设置接口电路？

- (1) CPU 与外设二者的信号不兼容，包括信号线的功能定义、逻辑定义和时序关系。
- (2) CPU 与外设二者的速度不匹配，CPU 的速度快，外设的速度慢。
- (3) 若外设直接由 CPU 控制，会使 CPU 忙于与外设打交道，大大降低了 CPU 的效率。
- (4) 若外设直接由 CPU 控制，会使外设的硬件结构依赖于 CPU，对外设的发展不利。

2. 简述 I/O 接口的信号有几种？各有什么特点？

接口信号通常有以下四种：

- (1) 开关量：只有 2 种状态，用一位二进制数表示开或关。
- (2) 数字量：二进制形式的数据或是已经过编码的二进制形式的数据。
- (3) 脉冲量：脉冲信号是以脉冲形式表示的一种信号，关注的是信号发生的跳变情况。
- (4) 模拟量：用模拟电压或模拟电流幅值大小表示的物理量。

3. 简述 I/O 接口的概念及其功能？由哪几部分组成？

它的概念是：I/O 接口是位于系统与外设之间、用来协助完成数据传送和控制 I/O 任务的逻辑电路。

它的功能有：

- (1) 设置数据的寄存、缓冲逻辑，以适应 CPU 与外设之间的速度差异。
- (2) 进行信息格式的转换，如串行和并行的转换。
- (3) 协调 CPU 与外设的信息类型和电平上的差异，如电平转换驱动器、数/模和模/数转换器等。
- (4) 协调时序差异，同步 CPU 与外设的工作。
- (5) 地址译码和设备选择功能，使 CPU 在某一时刻只能选中一个 I/O 端口。
- (6) 提供联络信号，承担 CPU 与外设之间的联络工作，联络的具体信息有控制信息、状态信息和请求信号等，如外设的“Ready”、“Busy”等状态。
- (7) 设置中断和 DMA 控制逻辑，以保证在中断和 DMA 允许的情况下，产生中断和 DMA 请求信号，并在接收到中断和 DMA 应答之后完成中断处理和 DMA 传输。

它的组成有：I/O 接口电路一般都由端口寄存器和控制逻辑两大部分组成。

- (1) 端口寄存器包括数据缓冲寄存器、控制寄存器和状态寄存器，它们是接口电路的核心。
- (2) 控制逻辑电路包括数据总线缓冲器、地址译码、内部控制逻辑和联络控制逻辑。

4. 一般接口电路中应具有哪些电路器件？

- (1) 数据锁存器和缓冲器，以解决 CPU 与外设之间速度不匹配的矛盾，并起隔离和缓冲的作用。

- (2) 控制和状态寄存器，以存放 CPU 对外设的控制命令和状态信息。
- (3) 地址译码器，用于选择接口电路中的不同端口。
- (4) 读写控制逻辑。
- (5) 中断控制逻辑。

5. 在通常的输入传送中，为什么输入接口仅用缓冲器而不用锁存器？

在输入数据时，因简单外设输入数据的保持时间相对于 CPU 的接收速度而言较长，因而输入数据通常不用加锁存器来锁存，可直接使用三态缓冲器与 CPU 数据总线相连即可。

6. 在通常的输出传送中，为什么输出接口必须要锁存器？

CPU 在输出数据时，经过数据总线的时间是很短的，因而需要使用输出锁存器把要输出的数据保留，即用锁存器将数据锁存后输出给外设。

7. I/O 接口中数据信息、控制信息和状态信息的作用。

数据信息是 I/O 接口中的有用信息，CPU 通过数据总线经 I/O 接口中的数据端口与外设交换数据，多为 8 位数据，对输入设备数据流的方向为经数据端口送数据总线，对输出设备数据流的方向为经数据总线送数据端口；控制信息和状态信息为辅助信息，控制信息经数据总线送控制端口，控制外设开始工作或停止工作；状态信息经状态端口送数据总线，告诉 CPU 外设的工作状态，输入设备的状态信息是 READY（是否准备好数据），输出设备的状态信息是 BUSY（设备是否繁忙），控制信息和状态信息多为位操作。

8. 什么叫端口？通常有哪几类端口？通常有哪几种的编址方式？

端口概念：CPU 和外设进行数据传输时，各类信息在接口中进入不同的寄存器，这些寄存器称为端口。

端口分类：数据端口、状态端口、控制端口。

端口编址：统一编址、独立编址。

9. I/O 接口电路采用统一编址或独立编址时对微处理器有无特殊要求？

采用统一编址时对 CPU 无特殊要求，只需要将 I/O 接口电路当作存储器一样对待即可。采用独立编址时对 CPU 有特殊要求，需要 CPU 的指令系统中包含有访问内存指令和访问外设指令，还需要有专门的访问内存操作和访问外设操作的控制逻辑。

10. 简述主机与外设进行数据交换的几种常用方式？

(1) 无条件传送方式：常用于简单设备，CPU 认为它们总是处于就绪状态，随时进行数据传送，适合简单、慢速的数据传输。

(2) 查询式传送方式：CPU 首先查询外设工作状态，在外设就绪时进行数据传送，效率低，适合慢速的数据传输。

(3) 中断控制方式：在外设就绪时，通过请求引脚信号，主动向 CPU 提出交换数据的请求，CPU 无其它更紧迫任务，则执行中断服务程序完成一次数据传送，适合少量、高速的数据传输。

(4) DMA 传送方式：外设和内存间直接建立传输通道，传输过程由 DMAC 控制，传输过程可与 CPU 执行任务并行，效率高，适合大量、高速的数据传输。

11. 试述【DMA 传送方式】的工作原理？DMA 控制器 8237 的主要功能是什么？

DMA 的工作过程：

- (1) 由专用接口芯片 DMAC（DMA 控制器）控制传送过程。
- (2) 当外设需要传送数据时，通过 DMAC 向 CPU 发出总线请求信号。

- (3) CPU 发出总线响应信号，释放总线。
- (4) DMAC 接管总线，控制外设和内存之间的数据传送。

8237A 的主要功能：

- (1) 8237A 中有 4 个独立的 DMA 通道。
- (2) 每一个通道的 DMA 请求都可以分别被允许和禁止。
- (3) 每一个通道的 DMA 请求有不同的优先权，优先权可以固定，也可以改变。
- (4) 每一个通道一次传输数据最大长度可达 64KB。
- (5) 8237A 有 4 种工作方式：单字节传送方式，数据块传送方式，请求传送方式以及级联方式。
- (6) 8237A 可以级联，任意扩展通道数。

12. 简述【中断控制方式】和【DMA 传送方式】相比有什么不同？

- (1) 中断方式通过程序实现数据传送，而 DMA 方式通过硬件实现数据传送，传送速度快。
- (2) CPU 对中断的响应是在执行完一条指令之后，而对 DMA 的响应则可以在指令执行过程中的任何两个存储周期之间，请求响应快。
- (3) 中断方式必须切换程序，要进行 CPU 现场的保护和恢复操作。DMA 仅挪用了—个存储周期，不改变 CPU 现场，额外花销小。
- (4) DMA 请求的优先权比中断请求高。CPU 优先响应 DMA 请求，是为了避免 DMA 所连接的高速外设丢失数据。
- (5) 中断方式不仅具有 I/O 数据传送能力，而且还能处理异常事件，DMA 只能进行 I/O 数据传送。

13. 什么是并行传送？什么是串行传送？

- (1) 在数据传送过程中，一个字或一个字节的各位同时被传送出去，这种传送方式称为并行传送。
- (2) 在数据传送过程中，一个字或一个字节的各位逐位被传送出去，这种传送方式称为串行传送。

14. 什么是并行接口？什么是串行接口？

- (1) I/O 接口与 I/O 设备之间数据传送采用并行传送，则称为并行接口。
- (2) I/O 接口与 I/O 设备之间数据传送采用串行传送，则称为串行接口。

15. 比较串行通信与并行通信的优缺点？

- (1) 从传送距离上看：并行通信适宜于近距离的数据传送，通常小于 30 米，而串行通信适宜于远距离的数据传送，可以从几米到数千公里。
- (2) 从传送速度上看：并行通信传送数据的速度比串行通信快得多。
- (3) 从传送费用上看：在远距离传送中，通信线路的费用占很大的比重，而串行通信的费用由于传送线少而比并行通信的费用低得多。

16. 简叙串行通信的工作方式？

- 单工工作方式：只允许一个方向传输数据，从发送方到接收方。
- 半双工工作方式：允许两个方向传输数据，但不能同时传输，只能交替进行。
- 全双工工作方式：允许两个方向同时进行传输数据。

17. 什么是信号的调制与解调？

- 调制是将数字信号转换为模拟信号。
- 解调是将模拟信号转换为数字信号。

18. 在远距离数据传输时，为什么要使用调制解调器？

在远距离数据传输时，通常使用电话线进行传输，电话线的频带比较窄，一般只有几 KHz，因此传送音频的电话线不适于传输数字信号，高频分量会衰减的很厉害，从而使信号严重失真，以致产生错码。使用调制解调器，在发送端把将要传送的数字信号调制转换成适合在电话线上传输的音频模拟信号，在接收端通过解调，把模拟信号还原成数字信号。

19. 什么叫波特率？什么叫波特率因子？什么叫比特率？

波特率：是单位时间内通信系统所传送的信息量，即位传输率。

波特率因子：是时钟频率与波特率的比值称为波特率因子。

比特率：是信息量传送的速率单位，表示每秒传输二进制代码位数，即 bit/s。

20. 设波特率因子为 64，波特率为 1200，那么时钟频率为多少？

时钟频率=波特率因子×波特率=64×1200=76800Hz=76.8KHz

21. 某异步通信接口，其帧格式由 1 个起始位、7 个数据位、1 个奇偶校验位和 1 个停止位组成，当该接口每分钟传送 1800 个字符时，试计算出传送波特率？

波特率=每帧总位数×每秒字符数=(1+7+1+1)×(1800/60)=300bit/s

22. 若在串行通信中的波特率是 1200b/s，8 个数据位，1 个停止位，无校验位，传输 1KB 的文件需要多长时间？

需要时间=字符总数/每秒字符数=1024/(1200/10)=8.53s

第八章 常见的三大芯片（11 道）

1. 8259A 的初始化命令字和操作命令字有哪些？其功能分别是什么？哪些应写入奇地址？哪些应写入偶地址？

初始化命令字 ICW1、ICW2、ICW3、ICW4，功能依次是：

- (1) ICW1：确定 8259A 的触发方式和工作方式。
- (2) ICW2：确定中断类型码。
- (3) ICW3：确定主片上的信号线连接从片的方式。
- (4) ICW4：确定 8259A 的全嵌套方式、缓冲方式和结束方式。

操作命令字 OCW1、OCW2、OCW3，功能依次是：

- (1) OCW1：中断屏蔽控制字。
- (2) OCW2：中断结束和优先权循环控制字。
- (3) OCW3：设置屏蔽方式、中断查询和读状态控制字。

ICW2，ICW3，ICW4，OCW1 写入奇地址；ICW1，OCW2，OCW3 写入偶地址。

2. 8259A 对优先级管理方式有哪几种？

- (1) 完全嵌套方式：允许高级中断打断低级中断，禁止低级中断打断高级中断或同级中断互相打断。
- (2) 自动循环方式：当任何一级中断被处理完，它的优先级就被改变为最低，而将最高优先级赋给原来比它低一级的中断请求。
- (3) 中断屏蔽方式：包括普通屏蔽方式和特殊屏蔽方式。

3. 中断控制器 8259A 中，IRR、IMR 和 ISR 三个寄存器的作用是什么？

中断请求寄存器 IRR：用来存放从外设来的中断请求信号 $IR_0 \sim IR_7$ 。 DI 位为 1 表示 IR_i 有中断请求，为 0 表示 IR_i 无中断请求。

中断屏蔽寄存器 IMR：用来存放对中断请求信号的屏蔽状态。 DI 位为 1 表示 IR_i 中断被禁止，为 0 表示 IR_i 中断被允许。

中断服务寄存器 ISR：用来存放正在被 8259A 服务的中断状态。 DI 位为 1 表示 IR_i 中断正在服务中，为 0 表示 IR_i 中断没有被服务。

4. 简述 8259A 的基本组成部分。

- (1) IRR，8 位中断请求寄存器，用来存放从外设来的中断请求信号 $IR_0 \sim IR_7$ 。
- (2) IMR，8 位中断屏蔽寄存器，用来存放对中断请求信号的屏蔽状态。
- (3) ISR，8 位中断服务寄存器，用来存放正在被 8259A 服务的中断状态。
- (4) PR，优先级判别器，也称优先级分析器。
- (5) 控制逻辑。
- (6) 读写逻辑。
- (7) 数据总线缓冲器。
- (8) 级联缓冲器/比较器。

其中，IRR、IMR、ISR、PR 和控制逻辑五个部分是实现中断优先管理的核心部件。

5. 8259A 的中断屏蔽寄存器 IMR 与 8086 中断允许标志 IF 有什么区别？

IF 是 8086 内部标志寄存器的一位，若 $IF=0$ ，8086 就不响应外部可屏蔽中断请求 INTR 引线上的请求信号。8259A 有 8 个中断请求输入线 $IR_0 \sim IR_7$ ，若某位 IMR_n 为 1，则表示对应这位的中断请求被禁止掉，无法被 8259A 处理，也无法向 8086 产生 INTR 请求，即 IMR 仅条件屏蔽 8259A 的某些中断源，而 IF 将屏蔽 8259A 的所有中断源。

6. 8255A 的功能是什么？有哪几个控制字？

8255A 是一种通用的可编程序并行 I/O 接口芯片。

它有两个控制字，一个是方式选择控制字，它的作用是实现 8255A 的各个端口的选择。另一个是对 C 口进行置位或复位控制字，它的作用是实现 8255A 的端口 C 的每一位进行控制。

7. 如果 CPU 通过 8255A 端口 C 的某一条线向外部输出连续的方波信号，说出两种实现方法。

(1) 8255A 工作于方式 0 时，端口 C 可以指定为输出，即每隔 $1/2$ 方波周期，就改变其中一位的状态，其它位保持不变，即可从端口 C 的某一条线向外部输出连续的方波信号。

(2) 用对端口 C 某一位置位或复位的方法实现，即每隔 $1/2$ 方波周期，对端口 C 的某一位交替进行置位、复位，即可从端口 C 的某一条线向外部输出连续的方波信号。

8. 8255A 有哪几种工作方式？有什么区别？

3 种工作方式。各工作方式的特点是：

方式 0，基本输入/输出方式。适用于同步传送方式和查询传送方式。

方式 1，选通输入/输出方式。适用于外设能提供选通信号或数据接收信号的场合，并且采用中断传送方式比较方便。

方式 2，选通双向传送方式。适用于一个既可以作为输入又可以作为输出的并行外设，并且输入和输出不会同时进行的场合。

9. 8255A 的方式 0 一般使用在什么场合？在方式 0 时，如果要使用查询传送方式进行输入输出，应该如何处理？

方式 0 的使用场合有两种，一种是同步传送方式，另一种是查询传送方式。

在方式 0 情况下，没有规定固定的应答信号，所以，将端口 A 和端口 B 作为数据端口，把端口 C 的高 4 位或者低 4 位规定为输出口，用来输出一些控制信号，而把端口 C 的另外 4 位规定为输入口，用来输入一些外设状态，即利用端口 C 来配合端口 A 和端口 B 的输入输出操作。如果要使用查询传送方式进行输入输出，可利用端口 C 的某一位作查询，只有当该位为 1 时，才可以将数据送到输入口或者输出口中。

10. 8253 有哪几种工作方式？有什么区别？

6 种工作方式。各工作方式的特点是：

方式 0，计数结束中断。门控信号为低电平时，计数停止；门控信号为高电平时，进行计数。

方式 1，硬件触发单拍脉冲。门控信号为低电平或者高电平时，计数不受影响；门控信号为上升沿时，受触发开始计数，下一个时钟后，输出为低电平，直到计数为 0。

方式 2，频率发生器。门控信号为低电平时，计数停止，输出高电平；门控信号为高电平时，进行计数；门控信号为上升沿时，重新设置初始值，开始计数。

方式 3，方波发生器。门控信号为低电平时方式，计数停止，输出高电平；门控信号为高电平时，进行计数；门控信号为上升沿时，开始计数。

方式 4，软件触发选通。门控信号为低电平时，计数停止；门控信号为高电平时，进行计数。

方式 5，硬件触发选通。门控信号为低电平时，计数不受影响；门控信号为上升沿时，开始计数。

11. 定时器和计数器有什么相同和不同？

定时器和计数器是同一器件，其共同的特点是都有一个计数脉冲输入端，每输入一个脉冲，计数器就进行加 1 或减 1 计数。若计数器件的计数脉冲的频率固定，则可利用计数实现定时，这就是定时器。若计数器件的作用仅仅是记录输入脉冲的多少，则称为计数器。