

# 计算机组成原理

●运算器由许多部件组成，其核心部分是（）。

- A、数据总线
- B、算术逻辑运算单元
- C、累加寄存器
- D、多路开关

**B**

●一个16K×32位的存储器，其地址线和数据线的总和是（）。

- A、48
- B、46
- C、36
- D、32

**B**

●DMA方式的接口电路中有程序中断部件，其作用是（）。

- A、实现数据传送
- B、向CPU提出总线使用权
- C、数据传输结束时向CPU发送中断请求
- D、实现周期挪用

**C**

●计算机中表示地址时，采用\_\_\_\_\_。

- A、原码
- B、补码
- C、反码
- D、无符号数。

**D**

●总线中地址线的作用是\_\_\_\_\_。

- A、只用于选择存储器单元
- B、由设备向主机提供地址
- C、用于选择指定存储器单元和I/O设备接口电路的地址
- D、即传送地址又传送数据

**C**

●设寄存器内容为80H，若它对应的真值是-127，则该机器数是。

- A、原码；
- B、补码；



- C、反码;
- D、移码。

C

●设机器字长为64位，存储容量为128MB，若按字编址，它的寻址范围是\_\_\_\_\_。

- A、16MB
- B、16M
- C、32M
- D、32MB

B

●DMA 访问主存时，让CPU 处于等待状态，等DMA 的一批数据访问结束后，CPU 再恢复工作，这种情况称作（）。

- A、停止CPU 访问主存
- B、周期挪用
- C、DMA 与CPU 交替访问
- D、DMA

A

●浮点数的表示范围和精度取决于\_\_\_\_\_。

- A、阶码的位数和尾数的机器数形式;
- B、阶码的机器数形式和尾数的位数;
- C、阶码的位数和尾数的位数;
- D、阶码的机器数形式和尾数的机器数形式。

C

●主机与设备传送数据时，采用\_\_\_\_\_，主机与设备是串行工作的。

- A、程序查询方式;
- B、中断方式;
- C、DMA方式;
- D、通道。

A

●常用的虚拟存储器寻址系统由\_\_\_\_\_两级存储器组成。

- A、主存 - 辅存;
- B、Cache - 主存;
- C、Cache - 辅存;
- D、主存—硬盘。

A

●下列描述中\_\_\_\_\_是正确的。



- A、控制器能理解、解释并执行所有的指令及存储结果;
- B、一台计算机包括运算器、控制器、存储器、输入设备和输出设备等五大组成成分;
- C、所有的数据运算都在CPU的控制器中完成;
- D、以上答案都正确。

**B**

●存储字长是指 ( ) 。

- A、存放在一个存储单元中的二进制代码组合个数
- B、存放在一个存储单元中的二进制代码位数
- C、存储单元的个数
- D、机器指令的位数

**B**

●下述说法中\_\_\_\_\_是正确的。

- A、EPROM是可改写的，因而也是随机存储器的一种
- B、EPROM是可改写的，但它不能用作随机存储器用
- C、EPROM只能改写一次，故不能作为随机存储器用
- D、EPROM是可改写的，但它能用作随机存储器用

**B**

●\_\_\_\_\_可区分存储单元中存放的是指令还是数据。

- A、存储器;
- B、运算器;
- C、控制器;
- D、用户。

**C**

●在单总线结构的CPU中，连接在总线上的多个部件 ( ) 。

- A、某一时刻只有一个可以向总线发送数据，并且只有一个可以从总线接收数据
- B、某一时刻只有一个可以向总线发送数据，但可以有多个同时从总线接收数据
- C、可以有多个同时向总线发送数据，并且可以有多个同时从总线接收数据
- D、可以有多个同时向总线发送数据，但可以有一个同时从总线接收数据

**B**

●存取周期是指\_\_\_\_\_。

- A、存储器的写入时间;
- B、存储器进行连续写操作允许的最短间隔时间;
- C、存储器进行连续读或写操作所允许的最短间隔时间;
- D、指令执行时间。。

**C**



●中断向量可提供\_\_\_\_\_。

- A、被选中设备的地址;
- B、传送数据的起始地址;
- C、中断服务程序入口地址;
- D、主程序的断点地址。

C

●一条指令中包含的信息有。

- A、操作码、控制码;
- B、操作码、向量地址;
- C、操作码、地址码;
- D、操作码、条件码。

C

●在浮点机中，判断补码规格化形式的原则是\_\_\_\_\_。

- A、尾数的符号位与第一数位不同
- B、尾数的第一数位为1，数符任意
- C、尾数的符号位与第一数位相同
- D、阶符与数符不同

A

●计算机中有关ALU的描述，\_\_\_\_\_是正确的。

- A、只做算术运算，不做逻辑运算;
- B、只做加法;
- C、能存放运算结果;
- D、以上答案都不对。

D

●在CPU的寄存器中，（）对用户是完全透明的。

- A、程序计数器
- B、指令寄存器
- C、状态寄存器
- D、通用寄存器

B

●I/O采用统一编址时，进行输入输出操作的指令是\_\_\_\_\_。

- A、控制指令
- B、访存指令
- C、输入输出指令
- D、程序指令

B



●微程序放在\_\_\_\_中。

- A、存储器控制器;
- B、控制存储器;
- C、主存储器;
- D、Cache。

**B**

●当定点运算发生溢出时，应该执行以下操作（）。

- A、向左规格化
- B、向右规格化
- C、发出出错信息
- D、舍入处理

**C**

●CPU不包括\_\_\_\_\_。

- A、地址寄存器
- B、指令寄存器IR
- C、地址译码器
- D、通用寄存器

**C**

●总线的异步通信方式\_\_\_\_\_。

- A、不采用时钟信号，只采用握手信号
- B、既采用时钟信号，又采用握手信号
- C、既不采用时钟信号，又不采用握手信号
- D、采用时钟信号，不采用握手信号

**A**

●存放欲执行指令的寄存器是\_\_\_\_\_。

- A、MAR;
- B、PC;
- C、MDR;
- D、IR。

**D**

●一个512KB的存储器，其地址线和数据线的总和是\_\_\_\_\_。

- A、17
- B、19
- C、27
- D、29

**C**



●一个16K×32位的存储器，其地址线和数据线的总和是\_\_\_\_\_。

- A、48;
- B、46;
- C、36;
- D、32。

**B**

●微指令操作控制字段的每一位代表一个控制信号，这种微程序的控制（编码）方式是\_\_\_\_\_。

- A、字段直接编码；
- B、直接编码；
- C、混合编码；
- D、字段间接编码。

**B**

●由于CPU内部操作的速度较快，而CPU访问一次存储器的时间较长，因此机器周期通常由一个\_\_\_\_\_来确定。

- A、指令周期；
- B、存取周期；
- C、间址周期；
- D、执行周期。

**B**

●在二地址指令中\_\_\_\_\_是正确的。

- A、指令的地址码字段存放的一定是操作数
- B、指令的地址码字段存放的一定是操作数地址
- C、运算结果通常存放在其中一个地址码所提供的地址中
- D、指令的地址码字段存放的一定是操作码

**C**

●计算机操作的最小单位时间是（）

- A. 时钟周期
- B. 指令周期
- C. CPU周期
- D. 中断周期

**A**

●下列器件中存取速度最快的是（）

- A. Cache
- B. 主存
- C. 寄存器
- D. 辅存

**C**



●主存和 CPU 之间增加高速缓冲存储器的目的是 ()

- A. 解决 CPU和主存之间的速度匹配问题
- B. 扩大主存容量
- C. 既扩大主存容量，又提高了存取速度
- D. 扩大辅存容量

A

●在中断周期中，将允许中断触发器置“0”的操作由 ()完成

- A. 硬件
- B. 关中断指令
- C. 开中断指令
- D. 软件

A

●下计算机中表示地址时，采用()

- A. 原码 B.
- 补码 C. 反
- 码 D. 无符号
- 数

D

●运算器由许多部件组成，其核心部分是()

- A. 数据总线
- B. 算术逻辑运算单元
- C. 累加寄存器
- D. 多路开关

B

●某计算机字长是 16 位，存储容量是 1MB，按字编址，它的寻址范围是()

- A. 512K
- B. 1M
- C. 512KB
- D. 1MB

A

●直接、间接、立即三种寻址方式指令的执行速度，由快至慢的排序是()

- A. 直接、立即、间接
- B. 直接、间接、立即
- C. 立即、直接、间接
- D. 立即、间接、直接

C



●以下叙述()是正确的

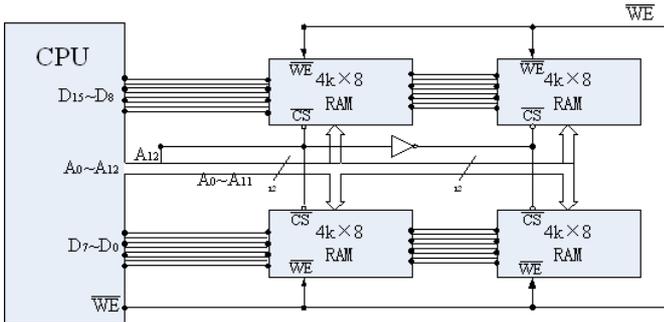
- A. 外部设备一旦发出中断请求, 便立即得到CPU的响应
- B. 外部设备一旦发出中断请求, CPU应立即响应
- C. 中断方式一般用于处理随机出现的服务请求
- D. 程序查询用于键盘中断

C

●采用4K×8 的RAM芯片, 构成8K×16的存储器。1) 画出存储器的逻辑框图, 图中标明信号线的种类、方向、条数。

2) 用十六进制写出该存储器占用的地址空间。

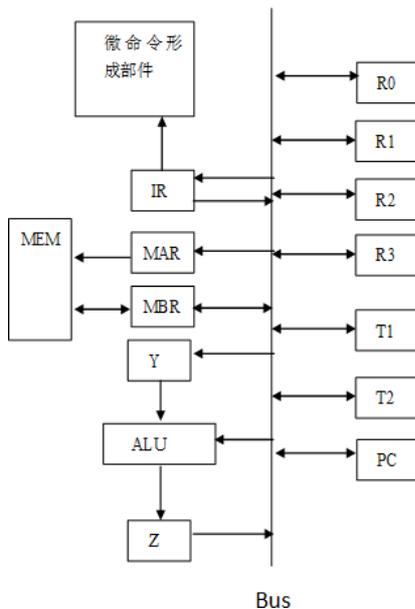
1) 存储器的逻辑框图如下: (8分)



2) 若按字编址, 存储器占用的地址空间是: 0000H~1FFFH;  
若按字节编址, 存储器占用的地址空间是: 0000H~2FFFH; (2分)  
上述两种答案均算对。

●单总线结构主机框图如下, 存储器按字编址。指令格式为: ADDR0, (R3);

源操作数R3为寄存器间接寻址, 目的操作数R0为寄存器直接寻址。操作形式为: ((R0)+((R3))→R0)。写出该指令的执行流程(从取指令开始)。



完成“ADDR0, (R3)”指令的执行流程如下  
(特别说明: 可以没有节拍安排):

取指周期

T0PC→Bus→MAR, 1→R



**T1MEM(MAR)→MBR,(PC)+1→PC**  
**T2MBR→Bus→IR,OP(IR)→微命令形成部件**  
**间址周期**  
**T0R3→Bus→MAR,1→R**  
**T1MEM(MAR)→MBR; 取操作数**  
**执行周期**  
**T0MBR→Bus→Y**  
**T1(Y)+(R0)→ALU→Z;R0通过总线送ALU**  
**T2Z→Bus→R0**

橙鹿学历宝  
 www.clxlb.com

●已知 $x=-0.1011$ ,  $y=+0.1101$ , 机器数的符号位采用双符号位表示。

求: (1)  $[X]$ 原=?  $[Y]$ 原=?

(2)  $[X]$ 补=?  $[Y]$ 补=?

(3)  $[X+Y]$ 补=?  $[X-Y]$ 补=? 并分别讨论其溢出情况。

(4)  $[x \times y]$ 原=?

①写出原码一位乘的运算步骤。

②写出原码乘法运算器的基本部件。

(1)  $[X]$ 原= $11.1011$   $[Y]$ 原= $00.1101$  (2分)

(2)  $[X]$ 补= $11.0101$   $[Y]$ 补= $00.1101$  (2分)

(3)  $[X+Y]$ 补= $[X]$ 补+ $[Y]$ 补= $11.0101+00.1101=00.0010$ 结果的两个符号位相同, 无溢出。

$[X-Y]$ 补= $[X]$ 补+ $[-Y]$ 补= $11.0101+11.0011=10.1000$ 结果的两个符号位不相同, 为10, 产生下溢。 (6分)

(4)

①设寄存器A= $00.0000$  (初始部分积)

B= $|X|=00.1011$  (存放被乘数), C= $|Y|=1101$  (存放乘数)

步数条件操作 ACC<sub>n</sub>+判断位

00.0000.1101

第1步 C<sub>n</sub>=1+B+00.1011

00.1011

→00.01011.110

第2步 C<sub>n</sub>=0+0+00.0000

00.0101

→00.001011.11

第3步 C<sub>n</sub>=1+B+00.1011

00.1101

→00.0110111.1

第4步 C<sub>n</sub>=1+B+00.1011

01.0001

→00.10001111

加符号位  $S_x S_y = 1 \oplus 1 = 0$ , 得原码乘法的乘积为 $[X \times Y]$ 原= $1.10001111$ 。 (6分)

②实现原码乘法运算器的基本部件包括:

(1) 存放部分积的寄存器;

(2) 存放乘数和被乘数的寄存器;

(3) 全加器;

(4) 计数器 (或记步电路)

(4分)

●将十进制数 $(60.25)_{10}$ 转换成二进制数、八进制数和十六进制数。

$(60.25)_{10} = (111100.01)_2 = (74.2)_8 = (3C.4)_{16}$

●已知 $x=0.1011$ ,  $y=-0.1101$ , 符号用双符号位表示。求  $[X+Y]_{补}$ ?  $[X-Y]_{补}$ ? 并分别讨论其溢出情况。

橙鹿学历宝  
 www.clxlb.com



$$2. [X]_{补} = 00.1011 \quad [Y]_{补} = 11.0011$$

$[X+Y]_{补} = [X]_{补} + [Y]_{补} = 00.1011 + 11.0011 = 11.1110$  结果的两个符号位相同，无溢出。

$[X-Y]_{补} = [X]_{补} + [-Y]_{补} = 00.1011 + 00.1101 = 01.1000$  结果的两个符号位不相同，为01，产生下溢。

●试分析比较DMA方式和中断传送方式的异同点，说明DMA方式为什么不能替代中断方式。

相同点：1) 两种方式均为目前在微型机中主机与外设广泛采用的信息交换方式。 2)

两种方式下主机和外设之间均可实现一定程度的并行工作。

不同点：1) 中断传送方式是通过中断服务处理程序来完成信息交换；而DMA方式则是用硬件代替软件来实现数据的传输。2) 中断传送方式不仅适合于一般的信息交换，还适合对随机事件的处理。3) DMA方式适合于高速外设和主机之间的信息交换，对高速外设采用程序中断方式传送数据往往回丢失数据。

DMA方式不能替代中断方式的原因：

1) DMA方式只能用于高速外设与内存直接交换信息，却不能像中断方式那样处理随机的异常现象。2) 在DMA方式的数据传输过程中，需用到中断方式。

●试简单说明CPU中控制器由哪几部分构成？简述各部分的功能。

CPU中控制器主要由程序计数器、指令寄存器、指令译码器、时序信号产生部件、微操作控制信号形成部件等部分构成。

各部分功能简述如下：

1)

程序计数器(PC)□在某些机器中用来存放正在执行的指令地址（或下条指令的地址）。

2) 指令寄存器(IR)用以存放从内存取出来的现行指令。□

3) 指令译码器对指令寄存器中的操作码进行分析解释，并产生相应的控制信号，提供给微操作控制信号形成部件。□

4) 时序信号产生部件具体产生不同指令相对应的周期、节拍、工作脉冲等时序信号。□ 5) 微操作控制信号形成部件是产生微命令的部件。

●简要说明动态RAM的各种刷新方式及其特点。

动态RAM的刷新方式有集中式刷新、分散式刷新、异步式刷新和透明式刷新等四种方式。

集中式刷新的特点：在最大刷新间隔时间内，集中安排一段时间进行刷新。其缺点是进行刷新时必须停止读、写操作。这对主机而言是个“死区”

分散式刷新的特点：刷新工作安排在系统的存取周期内进行，对主机而言不再有“死区”。但该方式加长了系统的存取周期，存在无谓刷新，降低了整机运行效率。因此，分散方式刷新不适用于高速存储器。

异步式刷新的特点：结合了上述两种方式的优点，充分利用了最大刷新间隔。相对于分散式刷新而言，它减少了刷新次数；相对于集中方式来说，主机的“死区”又缩短很多。因此，这种方式使用得比较多。

透明式刷新的特点：该方式不占用CPU时间，对CPU而言是透明的操作；但控制线路复杂。

●中断响应的条件是什么？中断响应主要完成哪些操作？

CPU响应中断的条件如下： 1)

有中断请求；

2) CPU允许中断，即中断允许状态IF=1（或EINT=1）；

3) 一条指令执行结束。

中断响应主要完成的操作包括：

1) 保护程序的断点；

2) 关中断；

3) 转中断服务程序入口。

上述操作，在中断周期中，由硬件（中断隐指令）完成。



### ●比较组合逻辑控制方式和微程序控制方式的优缺点。

- 1) 组合逻辑控制器的优点是运行速度快，缺点是设计与实现复杂，调试或修改困难，但随着EDA工具的成熟，该缺点已得到很大缓解。
- 2) 微程序的控制器优点是结构规整，设计效率高，易于修改，适用于实现系列计算机产品的控制器，缺点是运行速度慢。

### ●中断周期后是什么阶段？在中断周期CPU应完成什么操作？

- 1) 中断周期后是取指周期（取中断服务程序的第1条指令）；
- 2) 中断周期中，CPU由硬件（中断隐指令）完成如下操作：
  - ①保护程序的断点；
  - ②关中断；
  - ③转中断服务程序入口。

### ●存储器的层次结构主要体现在什么地方？为什么要分这些层次？

- 1) 存储器的层次结构主要体现在Cache—主存和主存—辅存这两个存储层次上。
- 2) Cache—主存层次主要解决CPU和主存速度不匹配的问题，在存储系统中主要对CPU访存起加速作用。从CPU的角度看，该层次的速度接近于Cache，而容量和每位价格却接近于主存。这就解决了存储器的高速度和低成本之间的矛盾；  
主存—辅存层次主要解决存储系统的容量问题，在存储系统中主要起扩容作用。从程序员的角度看，其所使用的存储器的容量和每位价格接近于辅存，而速度接近于主存。该层次解决了大容量和低成本之间的矛盾。

### ●说明微程序控制器的基本工作原理。

- 1) 将控制器所需要的微操作命令，以微代码的形式编成微指令，存在专门的控制存储器中；
- 2) CPU执行机器指令时，从控制存储器中取出微指令；
- 3) 对微指令中的操作控制字段进行解释，即产生执行机器指令所需的微操作命令序列。

### ●CPU中设有哪些寄存器？各寄存器的位数由何因素确定？

- 1) CPU中的寄存器主要包括通用寄存器，程序计数器PC，指令寄存器IR，存储器地址寄存器MAR，存储器数据寄存器MBR和状态标志寄存器等。
- 2) PC和MAR的位数取决于要访问的地址空间的大小。IR的位数取决于指令字长。通用寄存器及存储器数据寄存器MBR的位数取决于操作数（或操作数地址）的基本字长。

### ●以硬盘读写为例，说明在主机和外设之间进行数据传送，为什么需要采用DMA方式？

一些高速外设，如硬盘、光盘等I/O设备，经常需要和主存进行大批量的数据交换；若采用程序查询方式或程序中断方式来完成，即通过CPU执行程序来完成数据交换，速度较慢，极可能造成数据的丢失，因而不能满足批量数据的高速传递需求。因此，需要借助于硬件，比如DMA控制器来实现主存和高速外设之间的直接数据传送。

### ●计算机中设置Cache的作用是什么？能否扩大Cache的容量并取代主存，为什么？

- 1) 计算机中设置Cache主要是为了解决CPU和主存速度不匹配的问题，在存储系统中对CPU访存起加速作用。Cache和主存构成了Cache-主存存储层次，从CPU的角度看，该层次的速度接近于Cache，而容量和每位价格却接近于主存。这就解决了存储器的高速度和低成本之间的矛盾；
- 2) 不能把Cache的容量扩大到最后取代主存，主要是因为Cache通常由价格昂贵的双极型半导体器件构成，其每位价格远远高于普通的MOS型半导体器件构成的主存。若将其容量扩充到主存的容量，整个存储系统的成本会大幅度上升，因此不能取代主存。



●设机器数字长为 8 位 (含 1 位符号位在内) , 写出真值-13/64 对应的原码、反码和补码形式;

首先写出真值 -13/64 的二进制代码形式: 即  $-13/64 = -0.0011010$

- 1) 该真值对应的原码: 1.0011010
- 2) 由原码求出对应的反码: 1.1100101
- 3) 由原码求出对应的补码: 1.1100110。

●中断响应的条件是什么? 中断响应主要完成哪些操作?

CPU响应中断的条件如下: 1)

有中断请求;

2) CPU允许中断, 即中断允许状态  $IF=1$  (或  $EINT=1$ )

3) 一条指令执行结束

中断响应主要完成的操作包括:

1) 保护程序的断点;

2) 关中断;

3) 转中断服务程序入口。

上述操作, 在中断周期中, 由硬件 (中断隐指令) 完成。

●简要说明动态 RAM 的各种刷新方式及其特点

动态 RAM 的刷新方式有集中式刷新、分散式刷新、异步式刷新和透明式刷新等四种方式。

集中式刷新的特点: 在最大刷新间隔时间内, 集中安排一段时间进行刷新。其缺点是进行刷新时必须停止读、写操作。这对主机而言是个“死区”

分散式刷新的特点: 刷新工作安排在系统的存取周期内进行, 对主机而言不再有“死区”。但该方式加长了系统的存取周期, 存在无谓刷新, 降低了整机运行效率。因此, 分散方式刷新不适用于高速存储器。

异步式刷新的特点: 结合了上述两种方式的优点, 充分利用了最大刷新间隔。相对于分散式刷新而言, 它减少了刷新次数; 相对于集中方式来说, 主机的“死区”又缩短很多。因此, 这种方式使用得比较多。

透明式刷新的特点: 该方式不占用 CPU时间, 对 CPU而言是透明的操作; 但控制线路复杂

●比较组合逻辑控制方式和微程序控制方式的优缺点

1) 组合逻辑控制器:

优点:运行速度快

缺点:设计与实现复杂, 调试或修改困难, 但随着 EDA工具的成熟, 该缺点已得到很大缓解。

2) 微程序的控制器:

优点:结构规整, 设计效率高, 易于修改, 适用于实现系列计算机产品的控制器

缺点:运行速度慢。

●  
已知  $x=2^{-011} \times 0.101100$ ,  $y=2^{-010} \times (-0.100100)$

用浮点规格化补码加法求  $[x+y]$  补 (阶码、尾数均用补码表示), 要求写出计算步骤。



$[X]_{\text{补}}=1,101, 00.101100$        $[Y]_{\text{补}}=1,110, 11.011100$  浮点补码格式

1) 判零, 对阶,  $X_{\text{尾}}$  右移 阶码加 1,  $[X]_{\text{补}}=1,110, 00.010110$ 。

对阶后尾数求和,  $[X+Y]_{\text{补}}=1,110, 11.110010$ 。

2) 尾数规格化  $[X+Y]_{\text{尾}}$  左移 2 位, 阶码减 2, 得最后结果:

$[X+Y]_{\text{补}}=1,100, 11.001000$ , 即  $X+Y=2^{-100} \times (-0.111000)$



扫一扫上面的二维码图案, 加我微信

**另可提供视频代看服务, 有意加微**

橙鹿学历宝  
www.clxlb.com

升学历, 上橙鹿学历宝  
www.clxlb.com

