

目录

【注】确认收货后评价+带 3 图以上联系客服加 VIP 群 圆梦工大

前言.....	1
序.....	2
目录.....	3
期末试题部分	5
西北工业大学 2004-2005 学年第一学期期末考试	5
西北工业大学 2006-2007 学年第一学期期末考试	6
西北工业大学 2009-2010 学年第一学期期末考试(教育实验学院).....	8
西北工业大学 2011-2012 学年第一学期期末考试(教育实验学院).....	10
西北工业大学 2012-2013 学年期末考试.....	12
西北工业大学 2012-2013 学年第一学期期末考试(教育实验学院).....	14
西北工业大学 2013-2014 学年第一学期期末考试	16
西北工业大学未知年份期末考试(一).....	17
西北工业大学未知年份期末考试(二).....	19
西北工业大学未知年份期末考试(三).....	20
西北工业大学未知年份期末考试(四).....	21
西北工业大学未知年份期末考试(五).....	24
西北工业大学期末考试(软微 A 卷).....	26
西北工业大学期末考试(软微 B 卷)	28
南京大学计算机系统基础 2014-2015 学年第一学期期末考试(A 卷/开卷).....	28
南京大学计算机系统基础 2015-2016 学年第一学期期末考试(A 卷/开卷).....	30
历年真题部分	32
西北工业大学 2001 年研究生入学考试	32
西北工业大学 2002 年研究生入学考试(460)	34
西北工业大学 2003 年研究生入学考试(460)	36
西北工业大学 2004 年研究生入学考试(814)	38
西北工业大学 2004 年研究生入学考试(401)	39
西北工业大学 2004 年研究生入学考试(401)	39
西北工业大学 2007 年研究生入学考试(814)	41
2009 年研究生入学考试计算机统考 408.....	42
2010 年研究生入学考试计算机统考 408.....	44
2011 年研究生入学考试计算机统考 408.....	47
西北工业大学 2012 年研究生入学考试(801)	49
西北工业大学 2013 年研究生入学考试(801)	49
西北工业大学 2015 年研究生入学考试(801)	50
西北工业大学 2016 年研究生入学考试(801)	52
西北工业大学 2017 年研究生入学考试(801)	52
西北工业大学 2018 年研究生入学考试(801)	54
西北工业大学 2019 年研究生入学考试(801)	54
西北工业大学 2020 年研究生入学考试(847)	55
西北工业大学 2020 年研究生入学考试(879)	56

西北工业大学 2020 年研究生入学考试(801)	57
附录一	59
本科生期末试卷一	59
计算机组成原理期末试题及答案(一)	60
计算机组成原理期末试题及答案(二)	62
计算机组成原理期末试题及答案(三)	65
计算机组成原理期末试题及答案(四)	68
附录二 知识点总结	71
第一章 计算机系统概论	71
第三章 系统总线	73
第四章 存储器	76
第五章 输入输出系统	89
第六章 计算机的运算方法	94
第七章 指令系统及第八、九章	101

期末试题部分

西北工业大学 2004-2005 学年第一学期期末考试

一. 选择题 (每小题 1.4 分, 共 21 分)

答案速查: AACDB BCBAB BDDBA

- 1、A 【解析】5 的二进制编码为 101, 即 1.01×2^{10} , 即第一位符号位为 0, 阶码采用移码形式, 为 $2+127=1000\ 0001$ 尾数省略小数点之前的 1, 应为 010 0000 0000 0000 0000 0000
- 2、A 【解析】由于该 SRAM 芯片的容量为 $1K \times 4$ 位, 即数据线为 4 位, 地址线为 10 位, 加上 1 位片选线和 1 位读写控制线, 故最少应为 16 根。
- 3、C 【解析】由于该机器字长为 32 位, 采用双字编址, 即 64 位 (即 8B), 故寻址范围为存储容量除以字长 $=8MB/8B=1M$, 即 0——1M-1
- 14、B 【解析】通道对 CPU 的请求形式为中断, CPU 对通道的请求形式为通道命令

二. 填空题 (每空一分, 共 39 分)

- 1、定时协议 同步 异步 2、内存 CPU I/O 设备
- 3、先行 16 4 4、集中刷新 分散刷新 异步刷新
- 5、硬件判优 软件判优 异常处理 6、符号位 数值位 整数
- 7、3A 【解析】先将 26 和 63 按位或, 然后将结果和 135 按位异或, 将 16 进制和 8 进制均转换为 2 进制进行计算
- 8、SRAM 速度 主存-缓存 9、算数运算 数据传送 逻辑运算
- 10、主设备 控制权 总线仲裁 11、接口 选择 多路
- 12、可擦写 存储器 固态硬盘 13、地址译码 设备选择 传送控制信息状态信息
- 14、10011 4

三. 应用题 (共 40 分)

$$1、[\frac{1}{2}x]_{\text{补}}=0.01011 \quad [\frac{1}{4}x]_{\text{补}}=0.001011 \quad 2[-x]_{\text{补}}=1.1010 \quad [\frac{1}{2}y]_{\text{补}}=1.11011 \quad [\frac{1}{4}y]_{\text{补}}=1.111011 \quad 2[-y]_{\text{补}}=0.101$$

【解析】求 $[\frac{1}{2}x]_{\text{补}}$ 即将 $[x]_{\text{补}}$ 右移一位, 求 $2[-x]_{\text{补}}$ 即将 $[-x]_{\text{补}}$ 左移一位

2、操作码字段 OP 为 6 位, 故最多有 64 种不同的指令。源寄存器和基址寄存器均为 4 位, 即最多有 16 个通用寄存器。偏移量为 16 位, 即该指令为二地址指令, 可以进行寄存器直接寻址, 寄存器间接寻址和基址寻址, 立即数寻址, 间接寻址等寻址方式。

3、CPU 时钟周期为计算机主频的倒数, 即 $1/200\text{MHz}=5\text{ns}$

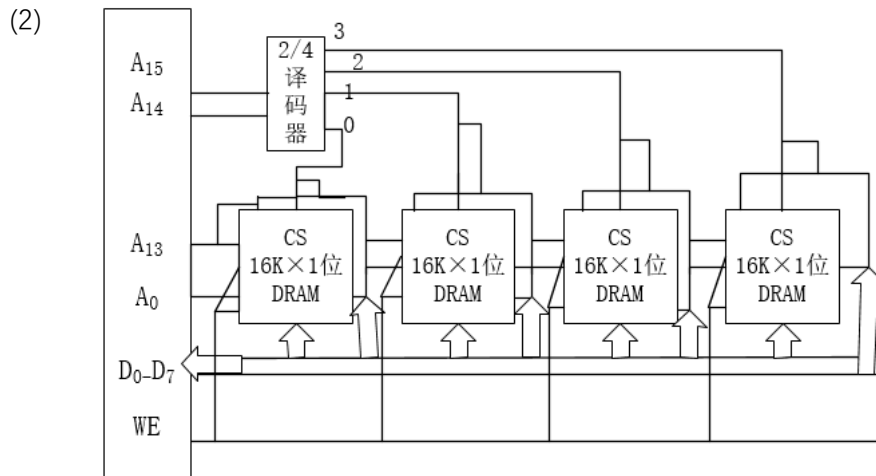
指令周期为 2.5 个 CPU 周期, 每个 CPU 周期又包含两个主频周期, 即一个指令周期包含 5 个时钟周期。即执行一条指令需要的时间为 $5\text{ns} \times 5=25\text{ns}$ 。

平均指令执行速度为 $1/25\text{ns}=40\text{MIPS}$

当每条指令平均包括 5 个 CPU 周期，每个 CPU 周期包含 4 个主频周期时，即一条指令包含 20 个时钟周期，即执行一条指令需要的时间为 $5\text{ns} \times 20 = 100\text{ns}$

平均指令执行速度为 $1/100\text{ns} = 10\text{MIPS}$

4、(1) 由于采用 $16\text{K} \times 1$ 位的 DRAM 芯片构成 $64\text{K} \times 8$ 位的存储器，所以首先需要按位扩展，使用 8 个 $16\text{K} \times 1$ 位芯片构成 $16\text{K} \times 8$ 位的存储器。然后按字扩展，需要 4 个 $16\text{K} \times 8$ 位的存储器构成 $64\text{K} \times 8$ 位的存储器。即共需要 $4 \times 8 = 32$ 片 $16\text{K} \times 1$ 位 DRAM 芯片。



(3)略(4)采用异步刷新较为合理，根据该芯片容量，该芯片的行列应为 256×256 。

两次刷新的最大时间间隔应为 $2\text{ms}/256 = 7.8\mu\text{s}$ 。将全部存储单元刷新一遍，多需的实际刷新时间应为 $256 \times 7.8\mu\text{s} = 1996.8\mu\text{s}$

西北工业大学 2006-2007 学年第一学期期末考试

一.选择题(每小题 1 分，共 10 分)

答案速查：1.C 2.BC 3.D 4.C 5.B 6.C 7.B 8.ABCD 9.C 10.A

1、C 【解析】将所有数转化为 10 进制进行比较

3、D 【解析】该 SRAM 芯片的容量为 512×8 位，即数据线 8 条，地址线 9 条，片选线 1 条，读写控制线 1 条，共 19 条。

4、C 【解析】该机器字长为 32 位，按字编址，存储容量为 64MB，即寻址范围为 64MB/4B

二.填空题(每空 1 分，共 20 分)

1、I/O 相同

2、操作数地址 ACC

3、选址 传送命令

4、节拍电位 节拍脉冲

5、定时协议 同步 异步

6、先行 16 4

7、集中刷新 分散刷新 异步刷新

8、-0.5 9、数据流 指令流

三.简答题(每小题 4 分, 共 20 分)

1、通过不同的时间段来区分指令和数据, 即在取指令阶段 (或取指微程序) 取出的为指令, 在执行指令阶段 (或相应微程序) 取出的即为数据。

通过地址来源区分, 由 PC 提供存储单元地址的取出的是指令, 由指令地址码部分提供存储单元地址的取出的是操作数。

2、存储程序并按地址顺序执行。

存储程序。也就是将用指令序列描述的解题程序与原始数据一起存储到计算机中。计算机只有一个启动, 就能自动的取出一条条指令并执行之, 直至程序执行完毕, 得到计算结果为止。

3、相同点:这两种方式下, 主机和 I/O 设备都是并行工作。

DMA 指的是直接内存访问。外部设备与计算机主存之间直接进行数据交换, 而无需经过 CPU。中断控制方式: 程序的切换需要保护和恢复现场, 对中断请求的相应发生在每条指令执行完毕时, 数据传送为程序传送。

DMA: 除了预处理和后处理, 不占用 CPU 资源, 对 DMA 请求的响应发生在每个机器周期结束, 传送过程不需要 CPU 干预, 数据传输率高, 请求优先级高, 数据传输靠硬件传送

4、先行进位解决进位的传递速度问题。让每一位的进位与其低一位的进位无关, 仅与两个参加操作的数以及最低位的进位有关。由于每位的操作数是同时给出的, 各进位信号几乎可以同时产生, 和数也随之产生, 所以先行进位可以提高进位的传递速度, 从而提高加法器的运算速度。

5、同步通信: 指由统一时钟控制的通信, 总线周期的长度是固定的, 控制方式简单, 灵活性差, 当系统中各部件工作速度差异较大时, 总线工作效率明显下降。适合于速度差别不大的场合。具有高的传输频率, 缺点: 同步总线必须按最慢的模块来设计公共时钟, 当各模块存取时间相差很大是, 会大大损失总线效率。

异步通信: 指没有统一时钟控制的通信, 部件间采用应答方式进行联系, 控制方式较同步复杂, 灵活性高, 当系统中各部件工作速度差异较大时, 有利于提高总线工作效率。优点: 总线周期长度可变, 不把响应时间强加到功能模块上, 因而允许快速和慢速的功能模块都连接到同一总线上; 缺点: 总线复杂, 成本较高。

四.计算应用题(共 50 分)

$$1、[X]_{\text{补}}=1.10001 \quad \left[\frac{1}{2} X\right]_{\text{补}}=1.110001 \quad \left[\frac{1}{4} X\right]_{\text{补}}=1.1110001 \quad [Y]_{\text{补}}=0.11001 \quad [-Y]_{\text{补}}=1.00111$$

$$\left[\frac{1}{2} Y\right]_{\text{补}}=0.011001 \quad \left[\frac{1}{4} Y\right]_{\text{补}}=0.0011001 \quad [X+Y]_{\text{补}}=0.0101 \quad [X-Y]_{\text{补}}=0.11000$$

求 $\left[\frac{1}{2} X\right]_{\text{补}}$ 即将 $[X]_{\text{补}}$ 右移一位, $\left[\frac{1}{4} X\right]_{\text{补}}$ 即将 $[X]_{\text{补}}$ 右移两位

2、(1) 由于采用 $64K \times 1$ 位的 DRAM 芯片构成 $256K \times 8$ 位的存储器, 所以首先需要按位扩展, 使用 8 个 $64K \times 1$ 位芯片构成 64×8 位的存储器。然后按字扩展, 需要 4 个

$64K \times 8$ 位的存储器构成 $256K \times 8$ 位的存储器。即共需要 $4 \times 8 = 32$ 片 $64K \times 1$ 位 DRAM 芯片。

(2) 应采用异步刷新, 两次刷新的最大间隔为 $2ms/128 = 15.625\mu s$, 故可取 $15\mu s$, 将全部存储单元刷新一遍, 需要的时间为 $15\mu s \times 128 = 1.92ms$

3、操作码字段 OP 为 7 位, 故最多有 128 条指令。源寄存器占 5 位, 即最多含有 32 个通用寄存器, 变址寄存器占 2 位, 即最多含有 4 个变址寄存器。

寻址方式特征为可为 2 位, 可以进行寄存器寻址, 寄存器间接寻址, 变址寻址, 相对寻址。

4、(1) a 为 MDR (数据寄存器) b 为 IR (指令寄存器) c 为 MAR (地址寄存器) d 为 PC (程序计数器)

因为 a 可以和主存交换数据, 而 c 只能向主存单向传送数据, 故 a 为 MDR, c 为 MAR。由于 d 需要进行 +1 操作, 显然 d 应为 PC。又 b 和操作控制器与 PC 相连, 故 b 应为 IR。

(2) (PC) \rightarrow MAR M(MAR) \rightarrow MDR MDR \rightarrow IR PC \rightarrow PC+1

(3) 读: M(MAR) \rightarrow MDR MDR \rightarrow ALU ALU \rightarrow ACC

写: ACC \rightarrow MDR MDR \rightarrow M(MAR)

5、5 的二进制编码为 101, 即 1.01×2^{10} , 即第一位符号位为 0, 阶码采用移码形式, 为 $2+127=1000\ 0001$ 尾数省略小数点之前的 1, 应为 010 0000 0000 0000 0000 0000

故组合起来应为 0100 0000 1010 0000 0000 0000 0000 0000

西北工业大学 2009-2010 学年第一学期期末考试(教育实验学院)

一、解释并比较 (每小题 6 分, 共 30 分)

1) 存取周期是指存储器进行连续 2 次独立的存储器操作 (如连续两次读操作) 所需的最小时间间隔 (2 分)。存取时间又称为访问时间, 是指启动一次存储器操作 (读或者写) 到完成该操作所需的全部时间 (2 分)。两者都用来表示存储速度, 存取周期通常大于存取时间 (2 分)。

2) 中断响应是指 CPU 识别中断源到确定中断服务程序入口地址的过程 (2 分); 中断处理是指执行中断服务程序的过程 (2 分)。两者共同构成了一个完整的中断服务过程, 中断响应通常是由硬件完成, 中断处理则是软件完成 (2 分)。

3) 基址寻址方式是指操作数的有效地址由指令码中的形式地址与基址寄存器中的内容相加形成 (2 分); 变址寻址方式是指操作数的有效地址由指令码中的形式地址与变址寄存器中的内容相加形成 (2 分)。基址寻址主要用于为程序或数据分配存储空间, 故基址寄存器的内容通常由操作系统或管理程序确定, 在程序的执行过程中其值不变; 编制寻址主要用于数组寻址等, 变址寄存器的内容由用户设定, 在程序执行过程中其值可变 (2 分)。

4) 总线的主设备是指拥有总线控制权, 启动总线上信息传送的设备 (2 分); 总线的从设备是指总线上只能接受总线命令没有总线控制权的设备 (2 分)。总线主设备获得总线控制权后才开始启动一次总线上的信息传送, 从设备响应主设备的总线命令共同完成一次总线传送 (2 分)。

5) 指令周期是指 CPU 取出并执行一条指令所需的全部时间 (2 分)。机器周期是所有指令执行过程中的一个基准时间 (2 分)。一般情况下, 一个指令周期包含若干个机器周期, 每个指令周期内的机器周期数可以不等。 (2 分)

【说明】每小题 6 分, 答对了其中的关键点就可以得 6 分, 部分答对都可以酌情给分 1-5 分。

二、【解】(1) x, y 为小数, 补码加法的基本公式为 $[x]_{\text{补}} + [y]_{\text{补}} = [x+y]_{\text{补}} \pmod{2}$

x, y 为整数数, 补码加法的基本公式为 $[x]_{\text{补}} + [y]_{\text{补}} = [x+y]_{\text{补}} \pmod{2^{n+1}}$

(两个中的任何一个都可以, 4 分)

可以根据 x, y 的符号, 分 4 种情况, 分别讨论。以小数为例:

【要点 1】 x 为正数时, $[x]_{\text{补}} = x$; x 为负数时, $[x]_{\text{补}} = 2 + x$

【要点 2】按照补码定义, $[x+y]_{\text{补}} = 2 + x + y \pmod{2}$

【要点 3】 $2 + 2 + x + y \pmod{2} = 2 + x + y \pmod{2}$

证明过程略 (证明过程 10 分)

(2) $A = -0.1110$ $[A]_{\text{补}} = 1.0001$ (1 分) $B = 0.1101$ $[B]_{\text{补}} = 0.1101$ (1 分)

按照补码加法计算方法, 计算过程 2 分, 结果正确 1 分。

结果没有溢出, 但需要说明没有溢出 (1 分)

【说明】只要证明过程是对的, 就可以给 10 分。加法计算需要有计算过程, 如果只有结果, 没有过程将扣去计算过程对应的分数。

三、(10 分)【解】(1) 在从缓冲区取出并进行处理的过程可中断的情况下, 完成 N 个数据的输入及处理共需时间 $P \times N + Q$ 秒 (2 分), 因此每个参数输入的平均时间为 $(P \times N + Q) / N$ 秒 (2 分), 若中断请求均匀到达, 则每秒可跟踪到 $N / (P \times N + Q)$ 个中断 (1 分)。

(2) 在从缓冲区取出并进行处理的过程不可中断的情况下, 第 N 个数据到达后将进行数据的取出和处理, 共需时间 $P + Q$ 秒 (3 分), 这是两次中断允许的最短间隔时间, 因此, 若中断请求均匀到达, 每秒可跟踪到的中断次数为 $1 / (P + Q)$ (2 分)。

四、(12 分)【解】(1) 16MB 容量的存储器, 掩字节编制, 地址 24 位 (1 分)

Cache 每块 32 字节, 块内地址 5 位 (1 分)

Cache 共有 $8K/32=256$ 块, 每组 4 块, 因此, 共有 64 组, 组地址 6 位 (1 分)

主存字块标记共有 $24-5-6=13$ 位 (1 分)

因此, 主存地址字段各段格式为

主存字块标记	组地址	块内地址	
13 位	6 位	5 位	(1 分)

(2) Cache 初始为空, 访问 0 号字时, Cache 不命中, 访问 1-7 号字时, Cache 命中, 同理, 可知, 访问 8, 16, ..., 96 号字时, Cache 不命中, 其余访问时, Cache 命中 (1 分)。重复访问, 即第 2-10 次访问时, Cache 命中 (1 分), 因此, 命中率为:

$(100 \times 10 - 13) / (100 \times 10) \times 100\% = 98.7\%$ (1 分)

无 Cache 时, 总访问时间为 $5t \times 100 \times 10 = 5000t$ (1 分)

有 Cache 时, 总访问时间为 $5t \times 13 + t \times (100 \times 10 - 13) = 948t$ (1 分)

速度提高的倍数为 $5000t / 948t - 1 = 3.75$ (倍) (1 分)

系统效率为 $t / (0.987 \times t + (1 - 0.987) \times 5t) \times 100\% = 95\%$ (1 分)

【说明】有依赖关系时, 第一次错误点扣分, 后续错误点可以酌情不扣分

五、(10 分)【解】(1) 4 个步骤中, 时间最长的一步的时间是 90ns, 因此, 流水线的时钟周期应为 90ns;

(2) 相邻的指令数据相关, 情况为

IF	ID	EX	WR
IF		ID	EX
		WR	

因此, 第 2 条指令推迟 2 个周期才能不发生错误 (3 分)

(说明: 若只有结果, 没有给出相应的说明, 酌情扣 1 分)

若相邻两指令发生数据相关, 不推迟第 2 条指令的执行, 可以采取数据旁路技术, 此时的情况为:

IF	ID	EX	WR
	IF	ID	EX
		WR	

上一条指令执行 (EX 阶段) 的结果直接作为下一条指令执行 (EX 阶段) 的操作数。(3 分)

(说明: 若只有结果, 没有给出相应的说明, 酌情扣 1 分)

六、(18 分)【解】(1) 47 种操作, 固定长度操作码, 需要 6 位 (1 分)

4 种寻址方式, 寻址模式字段 2 位 (1 分) 剩下的 $16-6-2=8$ 位可以用作形式地址 (1 分)

指令字格式为:

操作码	寻址模式	形式地址	
6 位	2 位	8 位	(3 分)

直接寻址范围为 $2^8=256$ (1 分) 一次间接寻址的寻址范围 $2^{16}=65536$ (1 分)

(2) 5 个控制字段分别包含 5、8、2、15、22 个微命令, 分别需要给出 6、9、3、16、23 个状态, 分别需要 3、4、2、4、5 位, 共 18 位 (5 分)

2 个可判定的外部条件, 条件测试字段需要 2 位 (1 分)

微指令字长 28 位, 因此, 下地址字段长度为 $28-18-2=8$ 位。(1 分)

微指令格式为: (2 分)

5 个微命令 8 个微命令 2 个微命令 15 个微命令 22 个微命令 条件测试 下地址

3 位 4 位 2 位 4 位 5 位 2 位 8 位

微程序存储器的容量为 256×28 位 (1 分)

【说明】有依赖关系时，第一次错误点扣分，后续错误点可以酌情不扣分

西北工业大学 2011-2012 学年第一学期期末考试(教育实验学院)

一、解释并比较(4 小题，每小题 6 分，共 24 分)

1) 存取周期：又称读周期或者写周期，即连续两次访问存储器操作所需的最小时间间隔。

存取时间：启动一次存储操作直到完成该操作所需的时间，具体地说从译码、驱动启动，直到被访问的存储单元内容读出或写入为止。

比较：存储周期包含存取时间和恢复时间。

2) 机器字长：计算机一次能直接处理的二进制数据的最长位数，一般等于内部通用寄存器的大小。

指令字长：一个指令字中包含二进制代码的位数，由操作码和地址码组成。

比较：现代 CPU 的机器字长和指令字长不一定相等，但两者都是字节的整数倍。

3) 基址寻址：将 CPU 中基址寄存器的内容，加上指令格式中的形式地址而形成操作数的有效地址。

变址寻址：把变址寄存器的内容（通常是首地址）与指令地址码部分给出的地址（通常是位移量）之和作为操作数的地址来获得所需要的操作数就称为变址寻址。

比较：基址寻址通常用作操作系统为多道程序分配存储空间，基址寄存器的值在寻址时不变；变址寻址立足于用户，主要用于处理数组问题，变址寄存器的值在寻址时会变。

4) 机器周期：计算机完成一次基本操作所需的时间，如取指，间址等。其时间通常等于存储周期。

指令周期：从取指开始到这条指令执行完成所需时间。

比较：指令周期由若干机器周期组成。

二、简要回答下列问题(3 小题，每小题 8 分，共 24 分)

(1) ①为了解决多个部件同时申请总线时的使用权分配问题。

②仲裁时机：总线空闲不忙时。

③申请分配、寻址、传输、结束阶段

(2) ①区别：程序查询方式下处理器处于轮询(死等)状态，只专注于读取数据，处理器利用率很低。而中断方式下，在中断来临之前处理器可以处理其他任务，只有中断信号来临并产生中断之后才传送数据，处理器利用率大大提高。

②响应时间不同：

CPU 对中断的响应是在中断产生时正在执行的那条指令执行完成之后才响应。

而对 DMA 的响应可以发生在每个机器周期结束时，只要 CPU 不占用总线就可以响应

(3) 定点补码判断方法：

法 1.当符号相同的两数相加，如果结果的符号和加数(或被加数)不相同，则为溢出

法 2.任意两符号两数相加时，若数值最高位的进位与符号位的进位不一致，则为溢出

法 3.如果采用双符号位，正数符号位为 00，负数为 11，两符号位不一致则为溢出，01 为正溢出，10 为负溢出。

浮点补码加减法溢出判断：如果阶码为全 1，若尾数为全 0 表示溢出（尾数符号为 1 表示负无穷大即下

溢，0 表示上溢)

原码定点除法和补码定点出发溢出判断：

以小数除法为例原码除法以第一次上商的商值来判断是否溢出若上商“1”即为溢出。补码除法以第一次上商的商值(即商符)与两操作数的符号位异或结果进行比较若比较结果不同即为溢出。例如两操作数符号位异或结果为 1 而第一次上商为 0 即为溢出

三、证明题(1 小题，共 12 分)

将 x, y 分为四种情况讨论

(1) $x=0.1010, y=0.0101$; $[x+y]_{\text{补}}=[0.1010+0.0101]_{\text{补}}=[0.1111]_{\text{补}}=0.111$

$$[x]_{\text{补}}+[y]_{\text{补}} = 0.1010+0.0101 = 0.1111$$

(2) $x=0.1010, y=-0.0101$; $[x+y]_{\text{补}}=[0.1010+(-0.0101)]_{\text{补}} = 0.0101$

$$[x]_{\text{补}}+[y]_{\text{补}} = 0.1010+[-0.0101]_{\text{补}} = 0.1010+2-0.0101=2+0.0101=0.0101 \bmod 2$$

(3) $x=-0.1010, y=0.0101$; $[x+y]_{\text{补}}=[-0.1010+0.0101]_{\text{补}}=[-0.0101]_{\text{补}}=1.1011$

$$[x]_{\text{补}}+[y]_{\text{补}} = [-0.1010]_{\text{补}}+[0.0101]_{\text{补}} = 1.0110+0.0101=1.1011$$

(4) $x=-0.1010, y=-0.0101$; $[x+y]_{\text{补}}=[-0.1010+(-0.0101)]_{\text{补}}=[-0.1111]_{\text{补}}=1.0001$

$$[x]_{\text{补}}+[y]_{\text{补}} = 1.0110+1.1011 = 11.0001 = 1.0001 \bmod 2$$

上述四个例子包括了 x, y 各种组合，均证明了 $[x+y]_{\text{补}}=[x]_{\text{补}}+[y]_{\text{补}}(\bmod 2)$;

四、应用题(3 小题，共 40 分)

1. (1) 主存根据字节编址那么 $4\text{MB}=2^{22}\text{B}$, 主存地址长度为 22;

由于是按字访问，每个存储字是 4B, 所以地址线为: $\log_2(2^{22}/2^2)=20$;

数据线数与机器字长相等: 32;

地址线数+数据线数=20+32=52;

(2) cache 字块的大小为: $8 \times 4\text{B}=32\text{B}=2^5\text{B}$, 所以组内地址长度为 5;

每个 cache 组的大小为: $4 \times 2^5\text{B}=2^7\text{B}$;

总共的 cache 组数为: $16\text{KB}/2^7\text{B}=2^7$; 所以 cache 组号占 7 位;

标记位: $22-5-7=10$;

标记位	组号	块内地址
10bit	7bit	5bit

(3) 由四路组相联映射图得:

在第一轮读入主存单元时，每读 8 个字产生一次缺块，100 个字共产生 $100/8 \approx 13$ 次内存块缺失。之后的 9 轮读取，都可以直接在 cache 中获得，无需访问主存。

命中率: $(100 \times 10 - 13)/(100 \times 10) = 98.7\%$;

设访问一次 cache 时间为 t ，访问一次主存时间为 $5t$; 平均访问时间为 $T = t \times 98.7\% + 5t \times 1.3\%$;

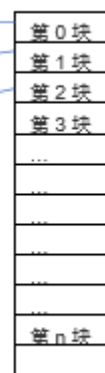
速度提高倍数: 访问主存的时间/平均访问时间 $= 5t/T = 4.78$ 倍;

效率: 访问 cache 的时间/平均访问时

cache :



主存 :



间 $= (t/T) \times 100\% = 95\%$;

2.(1)流水线周期应取 IF,ID,EX,WR 周期中的最大值, 即 90ns;

(2)推迟两个流水线时钟周期

IF	ID	EX	WR				
		IF			ID	EX	WR

即等待上一条指令的计算结果写入相应寄存器之后, 第二条指令才能执行 ID 周期

(3)采用数据旁路技术(定向技术),不需要等到上一条指令执行完并存放到缓冲或数据寄存器中之后再取数, 而是直接将上一条指令的执行结果送入第二条指令的输入数寄存器中。

3.(1)操作码为 6 位, 寻址方式表示位 1 位, 形式地址 9 位

OP	M	A
6bit	1bit	9bit

可直接寻址范围为: $0 \sim 2^9 - 1$; 一次间址寻址范围: $0 \sim 2^{16} - 1$

【注意】大家要区别好寻址范围和寻址空间的区别。如果是问寻址空间, 那么此题的答案就是, 直接寻址空间: $2^9 \times 2B$ 间址寻址空间: $2^{16} \times 2B$!!! 寻址空间一般指的是 CPU 对于内存寻址的能力。通俗地说, 就是能最多用到多少内存的一个问题, 即能够寻址的最大容量。一般用 MB 和 G 来表示。而范围是数的区间概念。

(2)时钟周期其实就是 CPU 的主频的倒数, 此题即是 $1/20\text{MHZ}$;

由题可知: 每条指令占用 4×2.5 个时钟周期, 所以执行速度: $20\text{MHZ} / (4 \times 2.5) = 2\text{MIPS}$

(3)由于是采用直接编码方式, 分为 5 个互斥组, 分别为 5、8、2、15、22 个微命令, 那么每个组需要产生的微操作分别为 6、9、3、16、23 种 (每组都需要加一个表示不产生任何操作信号的操作码, 通常全 0 表示), 那么表征每组微操作数目所需的二进制字段数目分别为 3bit, 4bit, 2bit, 4bit, 5bit; 外部条件是两个, 那么需 1bit 表示 (注意: 有些辅导书上的题目在此会取两位, 即有多少个条件判断就占几位。我们这里采取《计算机组成与结构》(第五版) 王爱英编著的方式, 即 1bit 表示两种外部条件。如有异议欢迎讨论)。剩下的 9 位为下址字段。那么水平型微指令格式为:

uOP1	uOP2	uOP3	uOP4	uOP5	BCF	BAF
3bit	4bit	2bit	4bit	5bit	1bit	9bit

由于微指令的下址字段直接给出后续微指令地址, 所以控制存储器容量: $2^9 \times 28\text{bit} = 14336\text{bit}$

西北工业大学 2012-2013 学年期末考试

一. 简单题(4x5=20 分)

1. 【解】计算机区分指令和数据有以下 2 种方法: 通过不同的时间段来区分指令和数据, 即在取指令阶段(或取指微程序)取出的为指令, 在执行指令阶段(或相应微程序)取出的即为数据。通过地址来源区分, 由 PC 提供存储单元地址的取出的是指令, 由指令地址码部分提供存储单元地址的取出的是操作数。

2. 【解】总线判优控制解决多个部件同时申请总线时的使用权分配问题; 常见的集中式总线控制有三种: 链式查询、计数器定时查询、独立请求; 特点: 链式查询方式连线简单, 易于扩充, 对电路故障最敏感; 计数器定时查询方式优先级设置较灵活, 对故障不敏感, 连线及控制过程较复杂; 独立请求方式速度最快, 但硬件器件用量大, 连线多, 成本较高。

3. 【解】指令周期: 取出并执行一条指令的时间。机器周期: 通常用内存中读取一个指令字的最短时间来规定 CPU 周期。(也就是计算机完成一个基本操作所花费的时间) 时钟周期: 处理操作的最基本单位。(CPU 的主频)

三者之间的关系：指令周期通常用若干个机器周期表示，而机器周期时间又包含有若干个时钟周期。

4. 【解】刷新：对 DRAM 定期进行的全部重写过程；

刷新原因：因电容泄漏而引起的 DRAM 所存信息的衰减需要及时补充，因此安排了定期刷新操作；

常用的刷新方法有三种：集中式、分散式、异步式。集中式：在最大刷新闻隔时间内，集中安排一段时间进行刷新，存在 CPU 访存死时间；分散式：在每个读/写周期之后插入一个刷新周期，无 CPU 访存死时间；异步式：是集中式和分散式的折衷。

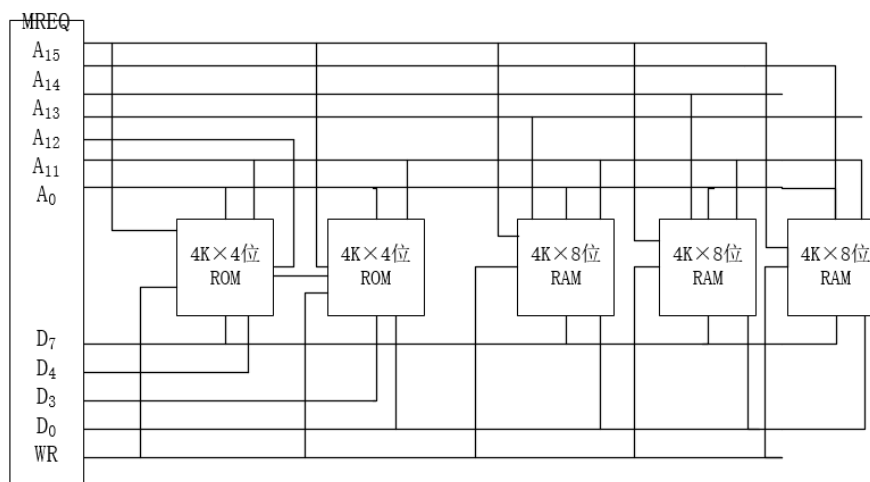
二.计算题(80 分)

1. 【解】(1)地址空间分配图：

系统程序区(ROM 共 4KB)0000H-0FFFH 用户程序区(RAM 共 12KB): 1000H-FFFFH

(2)选片：ROM：选择 4K×4 位芯片 2 片，位并联 RAM：选择 4K×8 位芯片 3 片，字串联(RAM1 地址范围为:1000H-1FFFH,RAM2 地址范围为 2000H-2FFFH,RAM3 地址范围为:3000H-3FFFH)

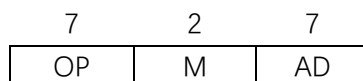
(3)各芯片二进制地址分配如下：



2. 【解】根据字符设备的传输率为 9600bps,得每秒能传输 $9600 / 8 = 1200\text{B}$,即 1200 个字符若采用 DMA 方式,传送 1200 个字符共需 1200 个存取周期,考虑到每传 400 个字符需中断处理一次,因此 DMA 方式每秒因数据传输占用处理器的时间是 $100\text{ns} \times 1200 + 5\mu\text{s} \times (1200 / 400) = 135\mu\text{s}$ 若采用中断方式,每秒因数据传输占用处理器的时间是 $5\mu\text{s} \times 1200 = 6000\mu\text{s}$

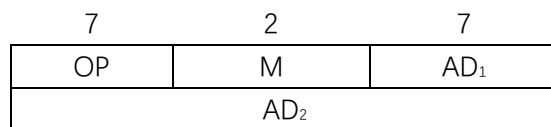
【备注】这里 CPU 和 DMA 之间采用的是周期挪用，要占用存取周期，也就是说 DMA 运行时通过周期挪用的方式占用总线，唐朔飞的书有相关题目

3. 【解】(1)在单字长指令中，根据能完成 105 种操作，取操作码 7 位。因允许直接和间接寻址，且有基址寄存器和变址寄存器，故取 2 位寻址特征位，其指令格式为：



其中 OP 为操作码，可完成 105 种操作；M 为寻址特征，可反映四种寻址方式；AD 为形式地址。这种指令格式可直接寻址 $2^7 = 128$ ，一次间址的寻址范围是 $2^{16} = 65536$ 。

(2)容量为 16MB 的存储器，正好与存储字长为 16 位的 8M 存储器容量相等，即 $16\text{MB} = 8\text{M} \times 16$ 位。欲使指令直接访问 16MB 的主存，可采用双字长指令，其操作码和寻址特征位均不变，其格式为：



其中形式地址为 AD1//AD2, 共 7+16=23 位。 $2^{23}=8M$, 即可直接访问主存的任一位置。

5. 【解】 $[x]_{\text{补}}=1.10001$ $[-x]_{\text{补}}=0.01111$ $[0.25x]_{\text{补}}=1.11101$ $[y]_{\text{补}}=0.11001$ $[0.25y]_{\text{补}}=0.00110$ $x+y=0.01010$ $x-y=1.010001$

6. 【解】根据题目给出的字符格式, 有效数据位为 8 位, 而传送一个字符需 $1+8+1+1=11$ 位, 故 比特率为: $1200 \times (8/11)=872.72\text{bps}$ 。

7. 【解】101 先用转换到 GF(2)上的多项式, 就是 $s(x)=x^2+1$ 在用生成多项式去对 信息进行编码: $g(x)*s(x)=x^6+x^5+x^3+1$, 注意这是有限域 GF(2)上的多项式运算, 系数要模 2 才行, 所以码字是: 1101001

8. 【解】 $(0.15625)_{10}=(0.00101)_2$, 二进制浮点为 1.01×2^{-3} , 阶码 $E=(127-3)_{10}=(124)_{10}=01111100B$, IEEE754 代
码是 1 01111100 0100000000000000000000

【注】单精度浮点数字长 32 位, 其中符号位占 1 位, 0 表示整数, 1 表示负数(双精度同), 指数部分占 8 位, 指数偏移量 127, 尾数部分占 23 位; 双精度浮点数字长 64 位, 尾数 52 位, 指数长度 11, 指数偏移量 1023。

西北工业大学 2012-2013 学年第一学期期末考试(教育实验学院)

一、(共 18 分, 每小题 3 分)解释下列名词的含义

- 1)机器字长: 计算机进行一次运算所能处理的二进制数据的位数。
- 2)总线的主设备: 拥有总线控制权, 启动总线上数据传输的设备, 如中央控制器、DMA 控制器等。
- 3)存取周期: 指连续启动两次独立的存储器操作 (例如连续两次读操作) 所需间隔的最小时间。
- 4)周期挪用: CPU 访存时, 必须待存取周期结束后, CPU 再将总线占有权让出, 当同时请求访存, CPU 暂时放弃总线占有权, 由 I/O 设备挪用一個或几个周期
- 5)寻址方式: 就是处理器根据指令中给出的地址信息来寻找有效地址的方式, 是确定本条指令的数据地址以及下一条要执行的指令地址的方法。
- 6)流水线的控制相关: 使下一条指令无法在设计时钟周期内执行

二、(共 24 分, 每小题 4 分)简要回答下列问题

- 1)通过不同的时间段来区分指令和数据, 即在取指令阶段 (或取指微程序) 取出的为指令, 在执行指令阶段 (或相应微程序) 取出的即为数据。
通过地址来源区分, 由 PC 提供存储单元地址的取出的是指令, 由指令地址码部分提供存储单元地址的取出的是操作数。
- 2)四个阶段
申请分配阶段: 由需要使用总线的主模块提出申请, 经总线仲裁机构决定将下一传输周期的总线使用权授予某一申请者;
寻址阶段: 取得了使用权的主模块通过总线发出本次要访问的从模块的地址及有关命令, 启动参与本次传输的从模块;
传输阶段: 主模块和从模块进行数据交换, 可单向或双向进行数据传送;
结束阶段: 主模块的有关信息均从系统总线上撤除, 让出总线使用权。
- 3)原因: 因电容放电而引起的 DMAR 所存信息的衰减需要及时补充。
集中刷新: 在一个刷新周期内, 利用一段固定的时间, 依次对存储器的所有行逐一再生, 在此期间停止对存储器的读写操作, 称为“死时间”
分散刷新: 把对每一行的刷新分散到各个工作周期中去, 无 CPU 访存“死区”异步刷新

4)①计算机是电子设备,而外设的工作方式多种多样

②CPU 只能识别数字量,而有些外设的信息是模拟量

③CPU 只能处理并行的信息,而有些外设的信息是串行的,有些外设的信息是并行的

④CPU 工作速度较快,而外设工作的速度慢的多

⑤CPU 和外设之间的信号电平有很多是不一致的

5)可以使用双符号位法,运算的两个符号位不同时,则表示溢出。

6)时钟周期:主频的倒数,计算机主时钟的周期时间,是计算机运行时最基本的时序单位,对应完成一个微操作所需的时间。

机器周期:在同步控制的机器中,执行指令周期中一步相对完整的操作(指令步)所需时间,通常安排机器周期长度等于主存周期;

指令周期:是取出一条指令并执行这条指令的时间。一般由若干个机器周期组成,是从取指令、分析指令到执行完所需的全部时间。

一个指令周期包含若干个机器周期,一个机器周期又包含若干个时钟周期,每个指令周期内的机器周期数可以不等,每个机器周期内的节拍数也可以不等。

三、(6 小题,共 58 分)应用题

1)若采用定点小数表示,条件是: $|X| < 1$, $|Y| < 1$, $|X+Y| < 1$ 。所以分四种情况证明:

① $X > 0$, $Y > 0$, 则 $X+Y > 0$ 。

因为 X 、 Y 都是正数,而正数的补码和原码是一样的,所以得:

$$[X]_{\text{补}} + [Y]_{\text{补}} = X + Y = [X+Y]_{\text{补}} \pmod{2}$$

② $X > 0$, $Y < 0$, 则 $X+Y > 0$ 或 $X+Y < 0$; $[X]_{\text{补}} = X$, $[Y]_{\text{补}} = 2+Y$, 即 $[X]_{\text{补}} + [Y]_{\text{补}} = X+2+Y = 2+(X+Y)$

当 $X+Y > 0$, $2+(X+Y) > 2$, 进位 2 必丢失,又因 $(X+Y) > 0$, 故: $[X]_{\text{补}} + [Y]_{\text{补}} = X+Y = [X+Y]_{\text{补}} \pmod{2}$

当 $X+Y < 0$, $2+(X+Y) < 2$, 又因 $(X+Y) < 0$, 故: $[X]_{\text{补}} + [Y]_{\text{补}} = 2+(X+Y) = [X+Y]_{\text{补}} \pmod{2}$

③ $X < 0$, $Y > 0$, 则 $X+Y > 0$ 或 $X+Y < 0$; 同 (2), 把 X 和 Y 的位置对调即可

④ $X < 0$, $Y < 0$, 则 $X+Y < 0$; 两负数相加,则其和也一定是负数。

因为 $[X]_{\text{补}} = 2+X$, $[Y]_{\text{补}} = 2+Y$, 即 $[X]_{\text{补}} + [Y]_{\text{补}} = 2+X+2+Y = 2+(2+X+Y)$

又 $|X+Y| < 1$, $1 < (2+X+Y) < 2$, $2+(2+X+Y)$ 进位 2 必丢失,而 $X+Y < 0$, 故:

$$[X]_{\text{补}} + [Y]_{\text{补}} = 2+(X+Y) = [X+Y]_{\text{补}} \pmod{2}$$

综上,得证。

2)根据图中已知, ROM_1 的空间地址为 0000H——3FFFH, ROM_2 的地址空间地址为 4000H——7FFFH, RAM_1 的地址空间为 C000H——DFFFH, RAM_2 的地址空间为 E000H——FFFFH。

对应上述空间,地址码最高 4 位 A_{15} —— A_{12} 状态如下:

0000——0011 ROM_1 ; 0100——0111 ROM_2 ; 1100——1101 RAM_1 ; 1110——1111 RAM_2

2 : 4 译码器对 $A_{15}A_{12}$ 两位进行译码,产生四路输出,其中: $y_0 = 00$ 对应 ROM_1 , $y_1 = 01$ 对应 ROM_2 , $y_3 = 11$ 对应 RAM_1 和 RAM_2 。然后用 A_{13} 区分是 RAM_1 ($A_{13} = 0$) 还是 RAM_2 ($A_{13} = 1$),此处采用部分译码。

由此,两组端子的连接方法如下: 1——6,2——5,3——7,8——12,11——14,9——3

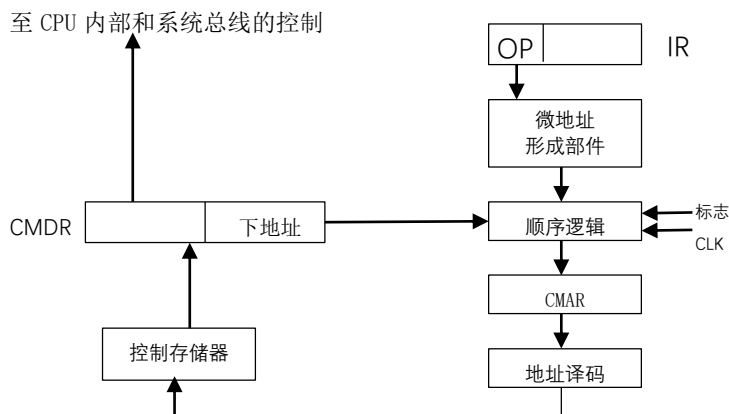
3) 因为控制存储器的容量为 512×40 位, 故微指令长为 40 位, 其中下地址字段为 9 位。因为有 6 个转移条件, 故判别测试字段为 3 位, 则微命令字段为 28 位。

控制寄存器: 核心部件, 用于存放各指令对应的微程序 (ROM)

微指令寄存器: 用于存放从 CM 中取出的指令

微地址形成部件: 用于产生初始微地址和后继微地址

微地址寄存器: 接受微地址形成部件送来的微地址



4) 平均访问时间 = $(3800 \times 50\text{ns} + 200 \times 250\text{ns}) / 4000 = 60\text{ns}$; 效率为: $250/60 = 4.17$

西北工业大学 2013-2014 学年第一学期期末考试

一、选择题(每小题 2 分, 共 10 分)

答案速查: ADABA

- 3、C 【解析】总线的数据传输率 = 总线的工作频率 \times (总线的宽度/8), 总线的时钟频率为 100MHz, 得 1 个时钟周期为 $0.01\mu\text{s}$; 一个总线传输周期等于 4 个时钟周期, 则一个总线传输周期为 $0.01\mu\text{s} \times 4 = 0.04\mu\text{s}$; 由于总线宽度为 32 位, 等于 4B, 所以总线数据传输速率为 $4/0.04 = 100\text{MB/s}$ 。
- 4、B 【解析】平均指令执行速度 = $8\text{MHz} / (2.5 \times 2) = 1.6\text{MIPS}$

二、填空题(每小题 1 分, 共 24 分)

- 程序 存储 存储器 运算器
- 设备 独立请求 寻址阶段 传输阶段
- SRAM DRAM 集中刷新 分散刷新
- 指令寄存器 程序计数器
- 指令译码 执行指令
- 阶码 对阶 规格化 溢出判断
- 算数运算 数据传送 逻辑运算 程序控制

三、应用题(66 分)

1、 $[X]_{\text{补}} = 1.10001$ $[-X]_{\text{补}} = 0.01111$ $[\frac{1}{2}X]_{\text{补}} = 1.110001$ $[Y]_{\text{补}} = 0.11001$

$[X+Y]_{\text{补}} = 1.10001 + 0.011001 = 0.0101010$; $[X-Y]_{\text{补}} = 1.10001 + 1.100111 = 0.011000$

2、(1) Cache 可分为 1K 个块, 按四路组相联方式映射, 故可分为 256 个组。

主存容量为 16MB, 故可分为 2M 个块。

主存字块标记 (21 位)	组地址 (8 位)	字块内地址 (3 位)
---------------	-----------	-------------

(2) 因为每字块有 8 个字, 故依次从 0, 1, 2, ..., 99 号单元读出 100 个字, 需要访存 13 次。由于已将字块调入 cache 中。后续读取不需要访存。故命中率为 $(1000 - 13) / 1000 = 98.7\%$ 。

若 cache 的速度是主存速度的 5 倍。则系统的平均访问时间为 $98.7\% \cdot t + 1.3\% \cdot 5t = 1.052t$
 速度提高了 $5t/1.052t = 4.75$ 倍，即提高了 3.75 倍。

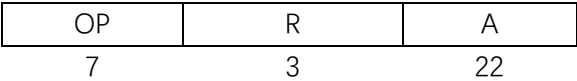
3、根据字符设备的传输率为 9600b/s，得每秒能传输：9600/8=1200B，即 1200 个字符。

1) 若采用 DMA 方式，传输 1200 个字符共需要 1200 个存取周期，考虑到每传 400 个字符需中断处理一次，因此 DMA 方式每秒因数据传输占用处理器的时间：

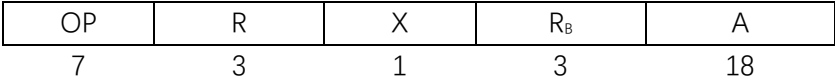
$$100ns \cdot 1200 + 5us \cdot (1200/400) = 135us$$

2) 若采用中断方式，每秒因数据传输占用处理器的时间是：5us*1200=6000us

4、(1)因可实现的操作有 110 种，故操作码需 7 位进行编码；要能直接访问 $4M = 2^{22}$ 的内存，需地址码为 22 位；8 个通用寄存器需 3 位进行编码，则指令字长为 7+22+3=32 位。指令格式如下：



(2)在上述指令格式中设置一寻址特征位 X，且 X=1 表示某个寄存器作基址寄存器 R_s ，因而需要加入基址寄存器的编码 3 位。指令格式为：



因为形式地址 A 为 18 位，通用寄存器也 16 位，均不够 20 位，无法对 4M 字的主存进行寻址。

【解决方案】可将基址寄存器的内容左移 6 位，达到 22 位的基地址，再与形式地址相加，即可对 4M 空间寻址。

(3)若主存容量扩大到 4G，则需要 32 位。因存储字长等于指令字长，即为 32 位，故可考虑采用一次间接寻址，即有效地址于内存单元中恰为 32 位。

5、	<div>取指令：(PC) ->MAR M(MAR)->MDR (PC)+1->PC (MDR)->IR</div>	<div>执行阶段：(R1) +X->AC1 AC0+AC1->AC1</div>
----	---	---

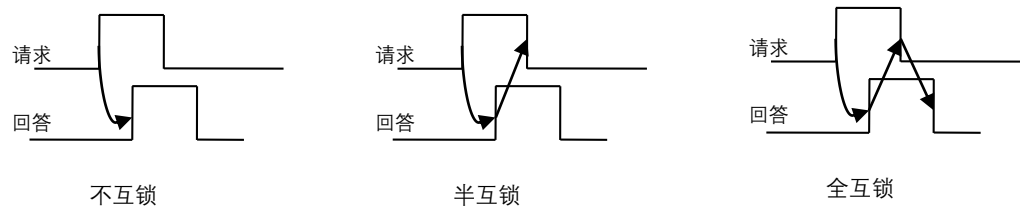
西北工业大学未知年份期末考试(一)

一、填空题

- 1、-0.5 2、同步 异步 3、操作数 ACC 的内容
 4、先行 16 4 5、集中刷新 分散刷新 异步刷新

二、问答题

1、3 种互锁关系



2、时钟周期：主频的倒数，计算机主时钟的周期时间，是计算机运行时最基本的时序单位，对应完成一个微操作所需的时间。

机器周期：在同步控制的机器中，执行指令周期中一步相对完整的操作（指令步）所需时间，通常安排机器周期长度等于主存周期；

指令周期：是取出一条指令并执行这条指令的时间。一般由若干个机器周期组成，是从取指令、分析指令到执行完所需的全部时间。

一个指令周期包含若干个机器周期，一个机器周期又包含若干个时钟周期，每个指令周期内的机器周期数可以不等，每个机器周期内的节拍数也可以不等。

3、中断向量地址是指中断服务程序的入口地址的地址。入口地址即中断向量，是中断服务程序的首地址。

4、不一样，CPU 响应中断后，需要保存现场并执行中断服务程序。而 DMA 方式只需要 CPU 执行几条 I/O 指令，完成预处理。

5、顺序执行和并发执行

PC 自加自动生成下一条指令地址。或转移指令（通过改变 PC 值实现）

6、冯诺依曼计算机的设计思想为计算机的硬件是由有形的电子器件等构成的。冯诺依曼计算机由五部分组成，分别为：

输入设备：将程序和数据以机器所能识别和接受的信息形式输入到计算机

输出设备：将计算机处理的结果以人们所能接受的形式或其他系统所要求的信息形式输出

存储器：用来存放数据和程序

运算器：计算机的执行部件，用于对数据进行加工处理，完成算术运算和逻辑运算。

控制器：计算机的指挥中心，由其“指挥”各部件自动协调的进行工作。

传统上将运算器和控制器称为 CPU，而将 CPU 和存储器称为主机。

7、总线定时是指总线在双方交换数据的过程中需要时间上配合关系的控制，这种控制称为总线定时，主要有同步定时和异步定时。

同步定时：指由统一时钟控制的通信，总线周期的长度是固定的，控制方式简单，灵活性差，当系统中各部件工作速度差异较大时，总线工作效率明显下降。适合于速度差别不大的场合。具有高的传输频率，缺点：同步总线必须按最慢的模块来设计公共时钟，当各模块存取时间相差很大是，会大大损失总线效率。

异步定时：指没有统一时钟控制的通信，部件间采用应答方式进行联系，控制方式较同步复杂，灵活性高，当系统中各部件工作速度差异较大时，有利于提高总线工作效率。优点：总线周期长度可变，不把响应时间强加到功能模块上，因而允许快速和慢速的功能模块都连接到同一总线上；缺点：总线复杂，成本较高。

8、顺序存储：从始端按顺序查找（磁带）

直接存储：先直接指出该存储器的某个区域再顺序查找（磁盘）

随机存储：可以直接访问任意存储单元（RAM、ROM）

关联存储：按内容查找 相联存储器（快表）

9、中断：中断就是使 CPU 暂停运行原来的程序而应更为急迫事件的需要转向去执行为中断源服务的程序，待该程序处理完后，再返回运行原程序。

中断源：即引起中断的事件或原因，或发出中断申请的来源。

中断系统：实现中断功能的硬件和软件称为中断系统。

中断向量：由发出中断请求的设备通过输入输出总线主动向 CPU 发出一个识别代码。

程序中断：指在计算机执行现行程序的过程中，出现某些急需处理的异常情况或特殊请求，CPU 暂时中止现行程序，而转去对这些异常情况或特殊情况请求进行处理，在处理完毕后 CPU 继续执行原程序。

多重中断：当 CPU 执行某个中断服务程序的过程中，发生了更高级、更紧迫的事件，CPU 暂停现行中断服务程序的执行，转去处理该事件的中断，处理完返回现行中断服务程序继续执行的过程

二、1 10000001 010 0000 0000 0000 0000 0000

【解析】第 1 位为符号位，第 2-9 位为阶码，用移码表示。后面为尾数，用原码表示，省略第一位

-5 的二进制形式为 -101; 可写为 -1.01×2^{10} 。显然, 第 1 位符号位为 1, 尾数应为 010 0000 0000 0000 0000; 在计算移码时需要加上 127, 即 $127+2=129$ (1000 0001)

三、(1) $[Y]_{\text{原}}=1.110011$ $[Y]_{\text{反}}=1.001100$ $[Y]_{\text{补}}=1.001101$

$[-Y]_{\text{补}}=0.110011$ $[Y]_{\text{移}}=0.001101$

(2) $[X+Y]_{\text{补}}=00.101101+11.001101=11.111010$

$[X-Y]_{\text{补}}=00.101101+00.110011=01.100000$

四、(1) 主存容量为 128KB, 可用 17 位二进制位表示。Cache 容量为 1KB, 可用 10 位二进制位表示。

(2) 每块大小为 4B, 故主存可分为 $128\text{KB}/4\text{B}=32\text{K}$ 块。Cache 可分为 $1\text{KB}/4\text{B}=256$ 块。

(3) 主存字块只能唯一的装入 Cache 中的唯一位置。

主存字块标记 (14 位)	Cache 字块地址 (8 位)	字块内地址 (2 位)
---------------	------------------	-------------

(4) 直接映像: $n\%256$

在全相联方式中主存的第 n 块可以映像到 Cache 中的任意位置。

西北工业大学未知年份期末考试(二)

一、填空题

- 1、数据 程序
- 2、每秒执行多少百万条指令 每秒执行多少百万次浮点运算
- 3、链式查询 独立请求
- 4、DRAM DRAM 集中刷新 异步刷新 分散刷新 集中刷新
- 5、通用寄存器 程序状态字寄存器 指令寄存器 程序计数器
- 6、数据流 指令流
- 7、选址 传送命令
- 8、32767 65535
- 9、机器周期 节拍
- 10、集中式 分布式
- 11、I/O 指令 独立
- 12、指令寄存器 程序计数器
- 13、取指令 分析指令 执行指令
- 14、存储程序

二、简答题

1、通过不同的时间段来区分指令和数据, 即在取指令阶段 (或取指微程序) 取出的为指令, 在执行指令阶段 (或相应微程序) 取出的即为数据。

通过地址来源区分, 由 PC 提供存储单元地址的取出的是指令, 由指令地址码部分提供存储单元地址的取出的是操作数。

2、解决多个部件同时申请总线时的使用权分配问题

链式查询: 连线简单, 易于扩充, 对电路故障最敏感, 优先级不灵活, 近的有最高优先权

计数器定时查询: 优先级设置较灵活, 对故障不敏感, 连线及控制过程较复杂

独立请求: 速度最快, 但硬件器件用量大, 连线多成本高

3、DMA 指的是直接内存访问。外部设备与计算机主存之间直接进行数据交换, 而无需经过 CPU。中断控制方式: 程序的切换需要保护和恢复现场, 对中断请求的相应发生在每条指令执行完毕时, 数据传送为程序传送。

DMA: 除了预处理和后处理, 不占用 CPU 资源, 对 DMA 请求的响应发生在每个机器周期结束, 传送过程不需要 CPU 干预, 数据传输率高, 请求优先级高, 数据传输靠硬件传送

4、时钟周期: 主频的倒数, 计算机主时钟的周期时间, 是计算机运行时最基本的时序单位, 对应完成一个微操作所需的时间。

机器周期: 在同步控制的机器中, 执行指令周期中一步相对完整的操作 (指令步) 所需时间, 通常安排机

器周期长度等于主存周期；

指令周期：是取出一条指令并执行这条指令的时间。一般由若干个机器周期组成，是从取指令、分析指令到执行完所需的全部时间。

一个指令周期包含若干个机器周期，一个机器周期又包含若干个时钟周期，每个指令周期内的机器周期数可以不等，每个机器周期内的节拍数也可以不等。

5、刷新：对 DRMA 定期进行的全部重写的过程。刷新是一行一行进行的，由 CPU 自动完成。

原因：因电容放电而引起的 DMAR 所存信息的衰减需要及时补充。

常用的刷新方法有三种：集中式、分散式、异步式。集中式：在最大刷新间隔时间内，集中安排一段时间进行刷新，存在 CPU 访存死时间；分散式：在每个读/写周期之后插入一个刷新周期，无 CPU 访存死时间；异步式：是集中式和分散式的折衷。

三、1、(1) 地址寄存器为 20 位，数据寄存器为 8 位。编址范围为 2^{20}

(2) 地址寄存器为 19 位，数据寄存器为 16 位。编址范围为 2^{19}

(3) 地址寄存器为 18 位，数据寄存器为 32 位。编址范围为 2^{18}

2、采用扩展码操作技术，操作码位数可变，则二地址、一地址和零地址的操作码长度分别为 8 位、16 位、24 位。这样二地址指令每减少一个，就可以多构成 2^8 条一地址指令操作码。一地址指令操作码每减少一个，就可以多构成 2^8 条零地址指令操作码。

假设有 X 条双操作数指令。则：

单操作数指令最多有： $(2^8 - X) \times 2^8$ ；无操作数指令最多有： $[(2^8 - X) \times 2^8 - M] \times 2^8$

又由题目得无操作数指令有 N 条，即 $N = [(2^8 - X) \times 2^8 - M] \times 2^8$ ，可解得： $X = 2^8 - [N \times 2^{-8} + M] \times 2^{-8}$

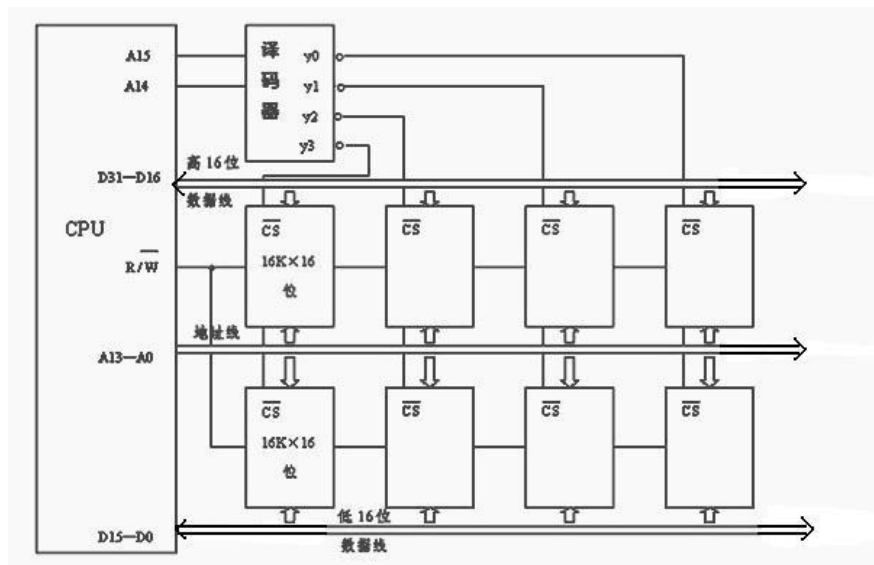
西北工业大学未知年份期末考试(三)

一.简答题

1.存储器所有存储单元的总数称为存储器的存储容量；存储器中是由许多存储单元组成的，每个存储单元都有编号，称为单元地址；如果某字代表要处理的数据，则称为数据字；如果某字为一条指令，则称为指令字。

2.指令是机器所能领会的一组编排成特定格式的代码串，他要求机器在一个指令周期内，完成一组特定的操作；程序是为特定问题求解而设计的指令序列。

3.所需芯片总数 $(64K \times 32) \div (16K \times 16) = 8$ 片，因此存储器可分为 4 个模块，每个模块 $16K \times 32$ 位，各模块通过 A15、A14 进行 2:4 译码



4. DRAM 存储元是通过栅极电容存储电荷来暂存信息。由于存储的信息电荷终究是有泄露的, 电荷数又不能像 SRAM 存储元那样由电源经负载管来补充, 时间一长, 信息就会丢失。为此必须设法由外界按一定规律给栅极充电, 按需要补给栅极电容的信息电荷, 此过程叫“刷新”。
5. 主存储器的性能指标有存储容量、存取时间、存取周期和存储带宽。
- 在一个存储器中可以容纳的存储单元总数通常称为该存储器的存储容量。
- 存取时间又称为存储器访问时间, 是指从启动一次存储器操作到完成该操作所经历的时间。
- 存储周期是指连续启动两次读操作所需间隔的最小时间。
- 存储器带宽是单位时间里存储器所存取的信息量。
6. 指令是机器所能领会的一组编排成特定格式的代码串, 他要求机器在一个指令周期内, 完成一组特定的操作。
- 一台计算机中所有机器指令的集合称为这台计算机的指令系统。
7. 由于设定全部指令采用 8 位固定的 OP 字段, 故这台计算机最多的指令条数为 $2^8=256$ 条。因此最多还可以设计出 $(256-m-n)$ 条单操作数指令。
8. 指令周期是完成一条指令所需的时间。包括取指令、分析指令和执行指令所需的全部时间。机器周期也称为 CPU 周期, 是指被确定为指令执行过程中的归一化基准时间, 通常等于取指时间(或访存时间)。时钟周期是时钟频率的倒数, 也称为节拍脉冲或 T 周期, 是处理操作的最基本单位。一个指令周期由若干个机器周期组成, 每个机器周期又由若干个时钟周期组成
9. (1) 最大存储容量。单总线系统中, 最大内存容量必须小于由计算机字长所决定的可能的地址总线;
- 双系统总线中, 存储容量不会受到外围设备数量的影响。
- (2) 指令系统。双总线系统, 必须有专门的 I/O 指令系统; 单总线系统, 访问内存和 I/O 使用相同指令。
- (3) 吞吐量。总线数量越多, 吞吐量越大
10. 分五个阶段: 总线请求, 总线仲裁, 寻址(目的地址), 信息传送, 状态返回(或错误报告)。
11. 连接到总线上功能模块有主动和被动两种形态。主方可以启动一个总线周期, 而从方只能响应主方的请求。每次总线操作, 只能有一个主方启用总线控制权, 但同一时间里可以有一个或多个从方。除 CPU 模块外, I/O 功能模块也可以提出总线请求。为了解决多个主设备同时竞争总线控制权, 必须具有总线仲裁部件, 以某种方式选择其中一个主设备作为总线的下一次主方。
- 一般来说, 采用优先级或公平策略进行仲裁。在多处理器系统中对 CPU 模块的总线请求采用公平原则处理, 而对 I/O 模块的总线请求采用优先级策略。
12. 集中式仲裁有链式查询方式、计数器定时查询方式、请求独立方式。
13. 一个计算机系统总线, 大致分为三类:
- ① CPU 内部连接各寄存器及运算部件之间的总线, 称为内部总线;
- ② CPU 同计算机系统的其他高速功能部件, 如存储器、通道等相互连接的总线, 称为系统总线;
- ③ 中、低速 I/O 设备之间相互连接的总线, 称为 I/O 总线。
14. ① 外设发出 DMA 请求; ② CPU 响应请求, DMA 控制器从 CPU 接管总线的控制; ③ 由 DMA 控制器执行数据传送操作; ④ 向 CPU 报告 DMA 操作结束; ⑤ 主要优点是数据传送速度快。

西北工业大学未知年份期末考试(四)

一、(21 分) 完成下列运算

1、X 的二进制: 1100 0001 八进制: -101 十六进制: -41

$[X]_{原}=1100\ 0001$ $[X]_{反}=1001\ 1110$ $[X]_{补}=1001\ 1111$ $[X]_{移}=0001\ 1111$

$[X+Y]_{补}=1110\ 0001$ $[X-Y]_{补}=0101\ 1101$ [溢出]

二、(52 分) 回答下列问题

1、CPU 主要由运算器和控制器两大部分组成

运算器：

暂存寄存器：用于暂存从主存读来的数据。

累加寄存器：通用寄存器，暂时存放 ALU 运算的结果，可以作为加法的一个输入端。

程序状态字寄存器：保留由算数逻辑运算指令或测试指令的结果而建立的各种状态信息。

通用寄存器组：如 AX、BX、CX、DX、SP 等，用于存放操作数（包括源操作数、目的操作数及中间结果）和各种地址信息。

控制器：

程序计数器：用于指出下一条指令在主存中的存放地址。

指令寄存器：用于保存当前正在执行的那条指令。

地址寄存器：用于存放所要访问的主存单元的地址。

数据寄存器：用于存放向主存写入的信息或从主存中读出的信息。

2、①方法：随机地选择被替换得块；**优点**：简单，易于硬件实现，并且对于硬件调试很有用；**缺点**：没有考虑块被使用的情况，反映不了程序的局部性

②方法：最先装入的块最先被替换；**优点**：简单，易于实现；**缺点**：虽然利用了各块进入 Cache 的顺序这一“历史”信息，但是还是不能正确反映程序的局部性。因为最先进入的块，很可能是经常用到的块。

③方法：最近最少被使用的块作为被替换的块；**优点**：反映了程序的局部性，因而失效率在三种方法中最低；**缺点**：比较复杂，硬件实现比较困难，特别是当 Cache 容量增加时，实现代价越来越高，而且经常只能是近似的实现（选最久没有被使用过的块作为被替换的块）

3、中断：中断就是使 CPU 暂停运行原来的程序而应更为急迫事件的需要转向去执行为中断源服务的程序，待该程序处理完后，再返回运行原程序。

中断源：即引起中断的事件或原因，或发出中断申请的来源。

中断系统：实现中断功能的硬件和软件称为中断系统。

断点：即程序执行中断的地方，保存断点以便处理完中断继续执行该程序

向量中断：是指一种识别中断源的技术或方式。识别中断源的目的就是要找到中断源对应的中断服务程序的入口地址的地址，即获得向量地址

单重中断：处理一个中断请求期间不允许被其他中断请求打断

4、【解析】为了解决多个主设备同时竞争总线控制权，必须具有总线仲裁部件，以某种方式选择其中一个主设备作为总线的下一次主方。对多个主设备提出的占用总线请求，一般采用优先级或公平策略进行仲裁。按照总线仲裁电路的位置不同，仲裁方式分为集中式仲裁和分布式仲裁两类。

集中式仲裁中每个功能模块有两条线连到中央仲裁器：一条是送往仲裁器的总线请求信号线 BR，一条是仲裁器送出的总线授权信号线 BG。

(1)链式查询方式

链式查询方式的主要特点：总线授权信号 BG 串行地从一个 I/O 接口传送到下一个 I/O 接口。假如 BG 到达的接口无总线请求，则继续往下查询；假如 BG 到达的接口有总线请求，BG 信号便不再往下查询，该 I/O 接口获得了总线控制权。离中央仲裁器最近的设备具有最高优先级，通过接口的优先级排队电路来实现。

链式查询方式的优点：只用很少几根线就能按一定优先次序实现总线仲裁，很容易扩充设备。

链式查询方式的缺点：对询问链的电路故障很敏感，如果第 i 个设备的接口中有关链的电路有故障，那么第 i 个以后的设备都不能进行工作。查询链的优先级是固定的，如果优先级高的设备出现频繁的请求时，优先级较低的设备可能长期不能使用总线。

(2)计数器定时查询方式

总线上的任一设备要求使用总线时，通过 BR 线发出总线请求。中央仲裁器接到请求信号以后，在 BS 线为“0”的情况下让计数器开始计数，计数值通过一组地址线发向各设备。每个设备接口都有一个设备地址判别电路，当地址线上的计数值与请求总线的设备地址相一致时，该设备置“1”BS 线，获得了总线使用权，此时中止计数查询。

每次计数可以从“0”开始，也可以从中止点开始。如果从“0”开始，各设备的优先次序与链式查询法相同，优先级的顺序是固定的。如果从中止点开始，则每个设备使用总线的优先级相等。

计数器的初值也可用程序来设置，这可以方便地改变优先次序，但这种灵活性是以增加线数为代价的。

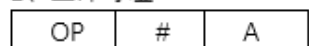
(3)独立请求方式

每一个共享总线的设备均有一对总线请求线 BR_i 和总线授权线 BG_i。当设备要求使用总线时，便发出该设备的请求信号。中央仲裁器中的排队电路决定首先响应哪个设备的请求，给设备以授权信号 BG_i。

独立请求方式的优点：响应时间快，确定优先响应的设备所花费的时间少，用不着一个设备接一个设备地查询。其次，对优先次序的控制相当灵活，可以预先固定也可以通过程序来改变优先次序；还可以用屏蔽(禁止)某个请求的办法，不响应来自无效设备的请求。

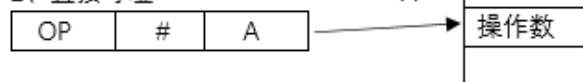
三、(10 分) 请画出如下寻址方式的示意图

1、立即寻址

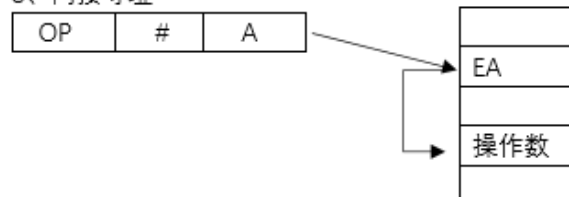


#为寻址特征，A 为立即数

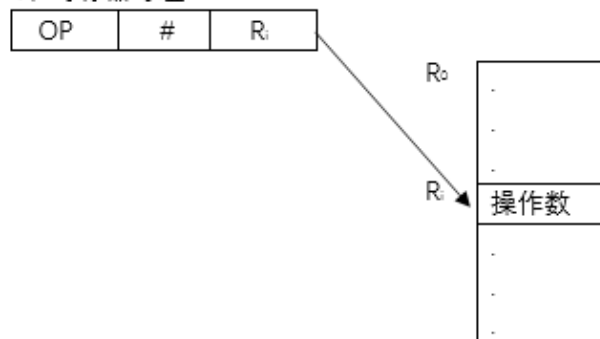
2、直接寻址



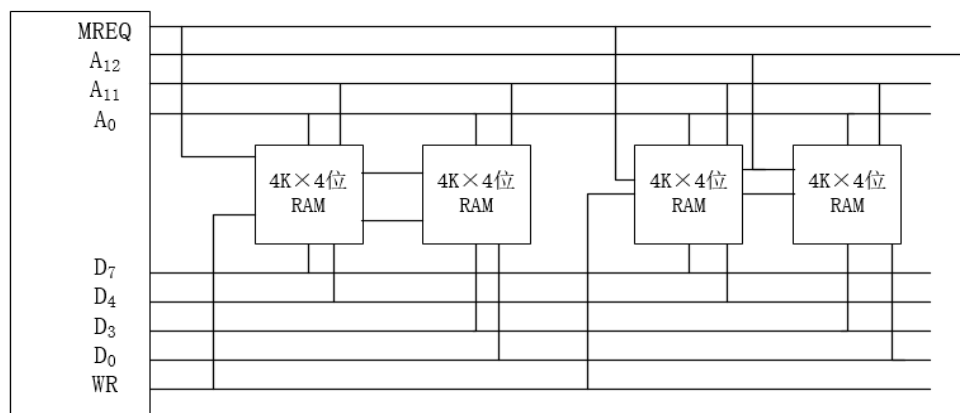
3、间接寻址



4、寄存器寻址



四、(10 分)



五、(7 分) 计算机字长为 16 位，故指令长为 16 位。指令系统有 60 条指令，故操作码为 6 位。四种寻址方

式，故寻址特征为 2 位。指令为单字长单地址指令。主存容量为 64K，故操作数部分需要 16 位

操作码	寻址特征	操作数
6 位	2 位	8 位

立即寻址：0~2⁸ - 1 直接寻址：2⁸ 间接寻址：2¹⁶ 相对寻址：2¹⁶

西北工业大学未知年份期末考试(五)

一 名词解释（每小题 2 分，共 10 分）

- 1、总线：是连接计算机各部件之间的一组公共的信号线。一般情况下，可分为系统总线和外总线
- 2、DMA：直接存储器访问，直接依靠硬件实现主存与外设之间的数据直接传输，传输过程本身不需 CPU 程序干预。
- 3、虚拟存储器：将主存或辅存的地址空间统一编址，形成一个庞大的地址空间
- 4、动态刷新：定期向电容补充电荷
- 5、微程序：定期向电容补充电荷

二 填空题

- 1、162.25 2、65535 3、11000111 4、1.010110 5、-1—1-2⁻⁵
- 6、同号 7、+[-X]_补 8、余数为负 9、寄存器间接
- 10、双稳态触发器 电容 11、16 16 12、程序计数器

三 简答题

- 1、浮点数中，为了在尾数中表示最多的有效数据位，同时使浮点数具有唯一的表示方式，浮点数的编码所采用的规范格式。
对阶：求阶差，然后以小阶向大阶看齐的原则，将阶码小的尾数右移一位，阶加 1，直到两个数的阶码相等为止。
尾数求和：将对阶后的尾数按定点数加（减）运算规则运算
规格化：当尾数的最高数值位与符号位不同时，即为规格化形式。
舍入：在对阶和右规的过程中，可能会将尾数低位丢失，引起误差，影响精度。
溢出判断：浮点数的溢出与否是由阶码的符号决定的。
- 2、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。
Cache-主存层次在存储系统中主要对 CPU 访存起加速作用，即从整体运行的效果分析，CPU 访存速度加快，接近于 Cache 的速度，而寻址空间和价位却接近于主存。
主存-辅存层次在存储系统中主要起扩容作用。
- 3、(1)按其功能可分为随机存取存储器（RAM）和只读存储器（ROM）
RAM：包括 DRAM 和 SRAM，当关机或断电时，其中的 信息都会随之丢失。 DRAM 主要用于主存，SRAM 主要用于高速缓存存储器。
ROM：只能读出不能随意写入信息，在主板上的 ROM 里面固化了一个基本输入/输出系统，其主要作用是完成对系统的加电自检、系统中各功能模块的初始化、系统的基本输入/输出的驱动程序及引导操作系统。

(2)按其制造工艺可分为：双极晶体管存储器和 MOS 晶体管存储器。

①双极型：运算速度比磁芯存储器速度约快 3 个数量级，而且与双极型逻辑电路型式相同，使接口大为简化。

②MOS：集成度高、容量大、体积小、存取速度快、功耗低、价格便宜、维护简单。

(3)按其存储原理，可分为：静态和动态两种。

①静态存储器：需要电源才能工作，只要电源正常，就能长期稳定的保存信息。

②动态存储器：超大容量的存储技术，跟其它类型的存储器相比，每兆比特的价格为最低。

4、物理地址是内存中各存储单元的编号，即存储单元的真实地址，它是可识别、可寻址并实际存在的。用户程序经过编译或汇编形成的目标代码，通常采用相对地址形式，其首地址为零，其余指令中的地址都是相对首地址而定。这个相对地址就称为逻辑地址或虚拟地址。逻辑地址不是内存中的物理地址，不能根据逻辑地址到内存中存取信息。为了保证 CPU 执行程序指令时能正确访问存储单元，需要将用户程序中的逻辑地址转运行时可由机器直接寻址的物理地址，这一过程称为地址映射或地址重定位。

5、1)从主存中取出一条指令，并指出下一条指令在主存中的位置。

2)对指令进行译码或测试，产生相应的操作控制信号，以便启动规定的动作。

3)指挥并控制 CPU、主存、输入和输出设备之间的数据流动方向。

程序计数器：用于指出下一条指令在主存中的存放地址。

指令寄存器：用于保存当前正在执行的那条指令。

指令译码器：仅对操作码字段进行译码，向控制器提供特定的操作信号。

MAR：用于存放所要访问的主存单元的地址。

MDR：用于在放向主存写入的信息或从主存中读出的信息。

时序系统：用于产生各种时序信号，它们都是由统一时钟(CLOCK)分频得到。

微操作信号发生器：根据 IR 的内容(指令)、PSW 的内容(状态信息)及时序信号，产生控制整个计算机系统所需的各种控制信号，其结构有组合逻辑型和存储逻辑型两种。

6、根据控制器产生微操作控制信号的方式的不同，控制器可分为硬布线控制器和微程序控制器，两类控制器中的 PC 和 IR 是相同的，但确定和表示指令执行出步骤的办法，以及给出控制各部分运行所需要的控制信号的方案是不同的。

硬布线控制器：根据指令的要求、当前的时序及外部和内部的状态情况，按时间的顺序发送一系列微操作控制信号。它由复杂的组合逻辑门电路和一些触发器构成，因此又称为组合逻辑控制器。速度快，应用 RISC 的 CPU，一旦设计完成，不能通过其他额外修改添加新功能。

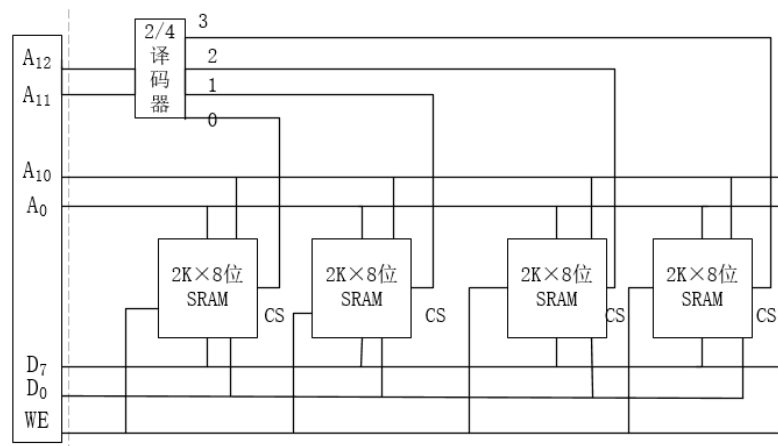
微程序控制器：采用存储逻辑实现，也就是把微操作信号代码化，使每条机器指令转化成为一段微程序并存一个专门的存储器(控制存储器)中，微操作控制信号由微指令产生。这些微程序可以存放在一个控制存储器中，用寻址用户程序机器指令的办法来寻址每个微程序的微指令。具有规整性，灵活性，可维护性，速度较慢，用于 CISC 的 CPU

四 计算题

(1) 16 位地址可以访问 64K 的地址单元，又该存储器的字长为 8 位，即该存储器能存储 $64K \times 8$ 位的信息，即 64KB

(2) 如果采用 $2K \times 8$ 位的芯片来构成该存储器，需要按字扩展，即需要 32 片该芯片。

(3) 由于采用 $2K \times 8$ 位的芯片来构成该存储器，故该芯片的片内地址应为 11 位，由于共需要 32 片该芯片，因此片选地址为 5 位。



西北工业大学期末考试(软微 A 卷)

一.(10 分)名词解释

系统总线：计算机系统内各功能部件之间相互连接的总线
中断向量：由发出中断请求的设备通过输入输出总线主动向 CPU 发出一个识别代码。
微操作：一条机器指令可以分解成一个微操作序列，是计算机中最基本、不可分解的操作
存储元：存储单元是存储器中可存放一个字或若干字节的基本单位
EEPROM：电可擦写只读存储器，能够用电子的方法擦除其中的内容。

二.(32 分)完成下列运算

- 1、 $X_2 = -0.101$ $X_8 = -0.5$ $X_{16} = -0.8$ $[X]_{原} = 1.101$ $[X]_{反} = 1.010$ $[X]_{补} = 1.011$ $[-X]_{补} = 0.101$ $[X]_{8421} = -0.0110$
0010 0101
2、8 位定点整数原码：-127——127； 8 位定点整数补码：-128——127
8 位定点小数原码：-1+2⁻⁷——1-2⁻⁷； 8 位定点小数补码：-1——1-2⁻⁷
3、C140 0000H
4、7_补=00.0111 -3_补=11.1101 -7_补=11.1001，采用 Booth 算法所得结果为 1.00010101

【解析】运算规则为：①符号位参与运算，运算的数均以补码表示；②2、被乘数一般取双符号位参与运算，部分积取双符号位，初值为 0，乘数可取单符号位；③乘数末位增设附加位 y_{n+1}，且初值为 0；④根据 (y_n, y_{n+1}) 的取值来确定操作；⑤移位按补码右移规则进行；⑥按照上述算法进行 n+1 步操作，但第 n+1 步不再移位（共进行 n+1 次累加和 n 次右移）仅根据 y_n 与 y_{n+1} 的比较结果做相应的运算即可

三.(38 分)回答下列问题

- 1、数据总线：用来传输各功能部件之间的数据信息，它是双向传输总线，其位数与机器字长、存储字长有关。
地址总线：用来指出数据总线上的源数据或目的数据所在的主存单元或 I/O 端口的地址

控制总线：传输控制信息，包括 CPU 送出的控制命令和主存返回 CPU 的反馈信号。

2、中断：中断就是使 CPU 暂停运行原来的程序而应更为急迫事件的需要转向去执行为中断源服务的程序，待该程序处理完后，再返回运行原程序。

保存程序断点，硬件关中断，将向量地址送至程序计数器（硬件向量法）或将中断识别程序入口地址送至程序计数器（软件查询法）

因为在系统总线中，地址总线是单向的，地址线上的信号只能用来选择内存单元或选择接口中的端口，而中断向量地址的内容需要在中断周期中被送至 PC，地址线做不到

3、程序计数器：用于指出下一条指令在主存中的存放地址

指令寄存器：用于存放当前正在执行的那条指令

指令译码器：仅对操作码字段进行译码，向至 CPU 内部和系统总线的控制

存储器地址寄存器：用于存放所要访问的主存单元的地址

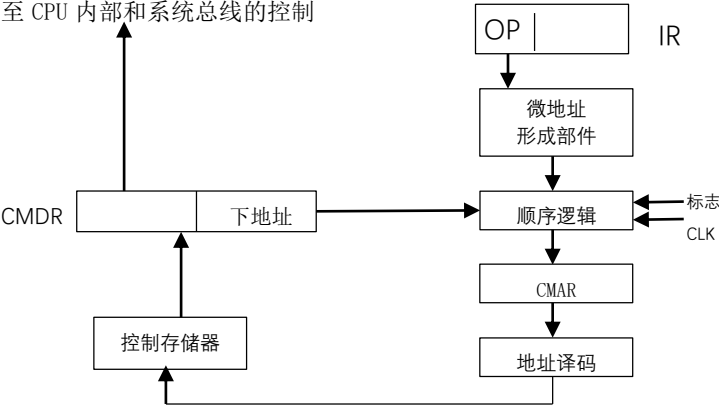
存储器数据寄存器：用于存放向主存写入的信息或从主存中读出的信息

4、控制寄存器：核心部件，用于存放各指令对应的微程序（ROM）

微指令寄存器：用于存放从 CM 中取出的指令

微地址形成部件：用于产生初始微地址和后继微地址

微地址寄存器：接受微地址形成部件送来的微地址



四. (10 分) (1)

操作码	寄存器 R	地址 A1	地址 A2
000	R	A1	A2
.....
110	R	A1	A2

(2)

操作码	寄存器 R	地址 A1
111 0000 0000 0000 000	R	A1
.....
111 0000 0111 1111 000	R	A1

(3)

操作码
1110 0000 1111 1110 0100 0000 0000 0000
.....
1110 0000 1111 1110 0100 0000 0011 0001

【注】答案不唯一

五.(10 分)请分析下列问题：

(1)行的长度较大，可以充分利用程序访问的局部性原理，使一个比较大的局部空间被一起调到 Cache 中因可以增加命中机会，但是行长也不能太长，行长大使失效损失变大。如果未命中的话，需要更多的时间从主存读快。行长太大，则 Cache 项数变少，命中可能性变小

(2)统一 Cache 比分立 Cache 有更高的命中率，因为它在获取指令和数据时可以自动实现平衡。只须设计和实现一个 Cache，设计难度低。分立 Cache 允许 CPU 在同一个 Cache 存储周期内同时取指令和数据，保证不

同指令可以同时访问

(3)提高计算机运算速度, 由于片内缓存在芯片内, 其容量不可能很大, 这就可能致使 CPU 欲访问的信息不在缓存内, 必须通过系统总线访问主存, 次数多了, 导致速度下降

西北工业大学期末考试(软微 B 卷)

一、(10 分)名词解释

ALU: 算术逻辑运算单元, 负责执行各种算术运算和逻辑运算。

中断: 中断就是使 CPU 暂停运行原来的程序而应更为急迫事件的需要转向去执行为中断源服务的程序, 待该程序处理完后, 再返回运行原程序。

【解析】其他见《西北工业大学期末考试(软微 A 卷)》一.名词解释

二、(32 分)完成下列运算

1、 $X_2 = -0.10001$ $X_8 = -0.42$ $X_{16} = -0.88$ $[X]_{原} = 1.10001$ $[X]_{反} = 1.01110$ $[X]_{补} = 1.01111$ $[-X]_{补} = 0.10001$

2、【解析】参考《西北工业大学期末考试(软微 A 卷)》二.完成下列运算

3、C0C0 0000H 【解析】-6 写成二进制位-110, 即 -1.10×2^{10} , 阶码采用移码表示, 即 $127+2=129$ (1000 0001), 尾数省略小数点之前的 1, 即 100 0000 0000 0000 0000 0000, 组合起来即为 1100 0000 1100 0000 0000 0000 0000 0000, 写为十六进制即为 C0C00000H

4、 $7_{补} = 00.0111$ $-3_{补} = 11.1101$ $-7_{补} = 11.1001$ $3_{补} = 00.0011$

【解析】运算规则为参考《西北工业大学期末考试(软微 A 卷)》二.完成下列运算的 4 小题

三、(38 分)回答下列问题

1、【解析】参考《西北工业大学期末考试(软微 A 卷)》三.回答下列问题的 3 小题

2、【解析】参考《西北工业大学期末考试(软微 A 卷)》三.回答下列问题的 4 小题

3、【解析】参考《西北工业大学期末考试(软微 A 卷)》三.回答下列问题的 1 小题

4、【解析】参考《西北工业大学期末考试(软微 A 卷)》三.回答下列问题的 2 小题

四、五同《西北工业大学期末考试(软微 A 卷)》

南京大学计算机系统基础 2014-2015 学年第一学期期末考试(A 卷/开卷)

一、(4 分) 略

二、(6 分) 在 0x080496e0 到 0x080496e3 这 4 个单元中存放的是-100, $-100 = -(64+32+4) = -1100100B$, 其对应的 8 位补码表示为 10011100, 在机器中的 32 位补码表示为 FFFFFFF9CH。因为 IA-32 是小端方式, 因此在 0x080496e0 到 0x080496e3 这 4 个单元的内容依次为: 9CH、FFH、FFH、FFH。(2 分)

三、CISC (1 分) 因为指令的长度参差不齐, 有一个、两个、三个、四个字节等不同长度 (1 分)

四、(4 分) 如图所示。

五、(10 分) cmpl 指令通过做减法来生成标志信息, 其执行将影响 EFLAGS 寄存器中的 OF、CF、ZF 和 SF 这

几位条件标志位。(4分)

当 $i=4$ 时, `cmpl` 指令中的操作数 $M[R[ebp]-4]$ 实际上就是 4, 因此, `cmpl` 指令实际上是在以下电路中实现“4 减 3”的功能, 也即在电路中 $X=0000\ 0004H$, $Y=0000\ 0003H$, $Y'=FFFF\ FFFCH$, $Sub=1$, 因此 $result=0000\ 0001H$, $OF=0$, $CF=0$, $ZF=0$, $SF=0$, 执行到 `jle` 指令时, 根据指令功能得知是带符号整数的比较, 因为 $OF=SF$, 因而是大于关系, 不满足 `jle` 指令的“小于或等于”关系, 因而不会跳到循环体内执行, 而是跳出循环执行 (6分)。

六、采用“比例变址+位移量”寻址方式 (1分) `EAX` 中存放的是 i (1分) 比例因子为 4 是因为 a 的每个元素占 4B, 且 IA-32 按字节编制 (1分) 有效地址 $EA=R[eax]*4+0x80496dc=i*4+0x80496dc$ (其中 $0x80496dc$ 是数组 a 的首地址) (1分) (略)

七、(10分) 如图所示。

根据 `sum` 反汇编结果可知只读数据和代码段的起始地址、可读写数据段的起始地址分别是 $0x8048000$ 和 $0x8049000$ (7分)

因为符号 a 和 val 都是初始化的全局变量, 所以被定义在可读写数据段中, `sum` 是函数名, 所以被定义在只读数据和代码段 (3分)

八、数据 `cache` 共有 $32KB/64B=512$ 行, 共有 $512/4=128$ 个 `cache` 组。因此 32 位主存地址中, 块内地址占 6 位, 组索引占 7 位, 高 19 位是标记 (3分)

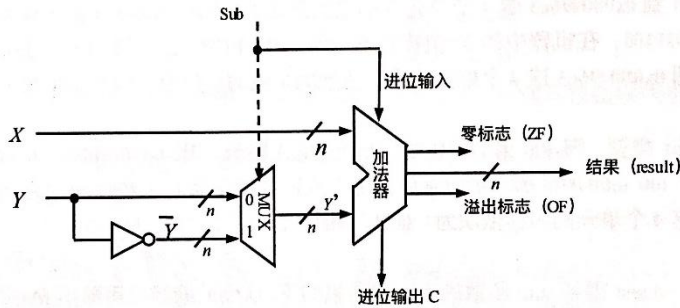
数组 a 中有 100 个元素, 每个元素占 4 个字节, 因此 a 共占 400B, $400B/64B=6.25$ 。其首地址为 $0x406dc$, 因此其最后一个元素的地址为 $0x406dc+396=0x403dc+0x18c=0x40868$ (3分)

将 $0x406dc$ 展开后为 100 000 0011011 011100, 将 $0x40868$ 展开为 100 000 0100001 101000, 因此可以看到 a 中的元素依次映射到 `chche` 的第 11011B=27 组、28 组、...、100001=33 组, 共 7 组。因此共有 7 次未命中, 命中率为 $93/100=93\%$ (4分)

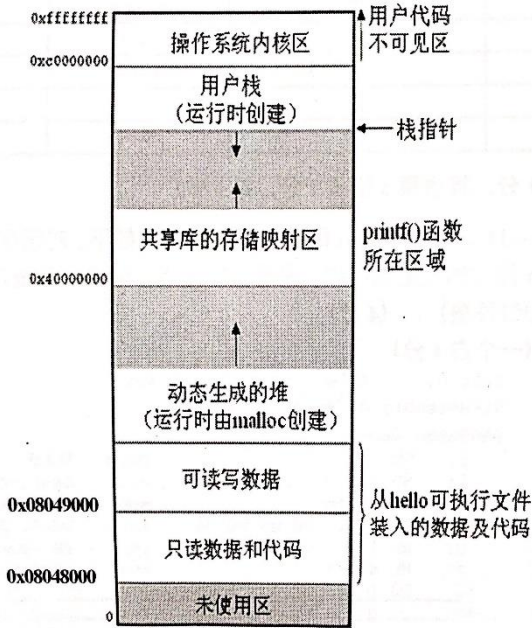
九、(i 和 N 各占 1 分, 其余每 3 格占 1 分, 共 6 分)



第四题图



第五题图



第七题图

标识符	在 test.o 的符号表中?	定义模块	符号类型	节
a	是	main.o	外部	---
val	是	test.o	全局	.data
sum	是	test.o	全局	.text
i	否	---	---	---
N	否	---	---	---

十、(4分) 如图所示。

十一、因为 `main.c` 中使用了标准 I/O 函数 `printf()`, 这个函数的原型说明在 `stdio.h` 中, 所以应该在 `main.c` 的开头加 `#include<stdio.h>` (1分)

因为在 `main.c` 的开头加了 `#include<stdio.h>`, 因此在预处理后, `printf()` 函数的原型说明被嵌入到了 `main.c` 中, 因此在 `main` 函数中引用 `printf` 函数不会发生错误。 (2分)

因为 printf 函数隐含的输出文件名是 stdout，即标准输出，因此在执行 test 程序后会在屏幕上显示字符串。(1 分)

十二、十三、(10 分) 略

```
test.o:      file format elf32-i386
Disassembly of section .text:
00080000 <sum>:
 0: 55                push    %ebp
 1: 89 e5             mov     %esp,%ebp
 3: 83 ec 10          sub     $0x10,%esp
 6: c7 45 fc 80 00 00 00 movl    $0x0,-0x4(%ebp)
 d: eb 1a            jnp     29 <sum+0x29>
 f: 8b 45 fc          mov     -0x4(%ebp),%eax
12: 8b 14 85 80 00 00 00 mov     0x0(,%eax,4),%edx
19: a1 00 00 80 00    mov     0x0,%eax
1e: 81 d8             add     %edx,%eax
28: a3 00 00 80 00    mov     %eax,0x0
25: 83 45 fc 81       addl    $0x1,-0x4(%ebp)
29: 83 7d fc 83       cmpl    $0x3,-0x4(%ebp)
2d: 7e e8             jle     f <sum+0xf>
2f: a1 00 00 80 00    mov     0x0,%eax
34: c9               leave   %eax
35: c3               ret
```

第十题图

南京大学计算机系统基础 2015-2016 学年第一学期期末考试(A 卷/开卷)

一、简答题（每题 4 分，共 20 分）

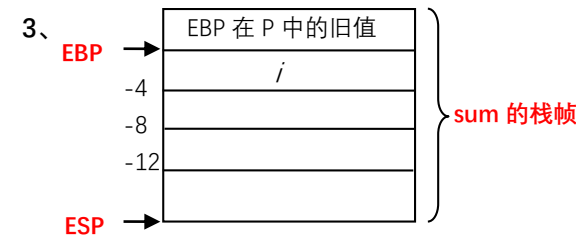
- 1、DMA 方式：I/O 设备和内存间直接通过 DMA 硬件批量传输数据。
- 2、用户程序需通过自陷指令向操作系统提供系统服务对应的编号来使用内核提供的相应服务。
- 3、控制器通过对指令进行译码，得到每条指令执行所需要的控制信号来控制指令的执行。
- 4、如溢出、缺页、越界、非法指令、除数为 0、堆栈溢出、断点设置、打印机缺纸、电源掉电、硬件故障、采用时间到等。
- 5、活跃页表在高速缓存中的镜像。

二、1、略 2、在 0x080496e0 到 0x080496e3 这 4 个单元中存放的是 3，3 =11B， 在机器中的 32 位原码表示为 00000003H。因为 IA-32 是小端方式，因此，在 0x080496e0 到 0x080496e3 这 4 个单元的内容依次为：03H、00H、00H、00H。(2 分)

在 0x080496e8 到 0x080496eb 这 4 个单元中存放的是浮点数-0.75，3 = BF400000H，因为 IA-32 是小端方式，因此，在 0x080496e8 到 0x080496eb 这 4 个单元的内容依次为：00H、00H、40H、BFH。(2 分)

Val 的机器值是 FFH，(2 分)

printf 输出的值是 255 (2 分)



4、cmpl 指令通过做减法来生成标志信息，其执行将影响 EFLAGS 寄存器中的 OF、CF、ZF 和 SF 这几位条件标志位。(4 分)

因为 $a = -1$ ，cmpl 指令中的操作数 $M[R(\%ebp) + 8]$ 实际上就是 a ，因此，cmpl 指令实际上是在以下电路中实现“-1 减 0”的功能。如果“-1 <= 0”，则跳转到 0x804843a 处执行，否则执行 0x804842f 语句。(6 分)

5、基址+位移 (2 分)

$i = a + b;$ (2 分)

leave 执行的是 movl %ebp, %esp

popl %ebp (2 分)

ret 执行的是 1) 取出返回地址，2) 将返回地址送入 EIP 中 (2 分)

三、【分析解答】(1) Save 数组的每个元素占 4 个字节。因为每次循环取数组元素时，其下标地址都要乘以 4。

(2) MIPS 中有 32 个通用寄存器，因为寄存器的编号是 5 位。

(3) “addi”指令的操作码是 001000B，因为其 OP 字段为 8，相当于 6 位二进制编码 001000。

(4) 标号 exit 的值是 80024，其含义是循环结束时跳出循环后执行的首条指令的地址。它由当前分支指令（条件转移指令）的地址 80012 加上 4 得到下条指令的地址，然后再加上相对位移量 2×4 得到，即 $80012 + 4 + 2 \times 4 = 80024$ 。

(5) 标号 loop 的值为 80000，是循环入口处首条指令的地址，由跳转指令 j 的 32 位地址 80020 的高 4 位 (0000B)，与指令中给出的低 26 位 (20000) 拼接成 30 位地址，然后再在低位添两个 0（相当于 $\times 4$ ）得到，即 $20000 \times 4 = 80000$ 。因为跳转指令的地址与其跳转到的目标指令地址的高 4 位一样，所以，如果将 4GB 的主存空间分割成 16 个 256MB 的子空间，那么跳转到的目标指令总是和跳转指令在同一个子空间，不可能跳出它本身所在的 256MB 的子空间，所以跳转目标地址范围的大小是 256M，也即，假定跳转指令地址的高 4 位为 X，则跳转目标地址范围是 X000 0000H ~ XFFF FFFCH。

(6) 单周期处理器的时钟周期最小为 $200 + 50 + 100 + 200 + 50 = 600\text{ps}$ ；多周期处理器的时钟周期最小为 200ps。对于单周期数据通路中的 10 次循环执行，第 1~4 条指令执行了 10 次，第 5~6 条指令执行了 9 次，因此，共用了 $(4 \times 10 + 2 \times 9) \times 600 = 34800\text{ps} = 34.8\text{ns}$ 。对于多周期数据通路中的 10 次循环执行，sll、add 和 addi 指令都需要 4 个时钟周期，bne 和 j 指令需要 3 个时钟周期，lw 指令需要 5 个时钟周期，因此，一共用了 $(4 + 4 + 5 + 3) \times 10 \times 200 + (4 + 3) \times 9 \times 200 = 44600\text{ps} = 44.6\text{ns}$ 。

(7) 第 1 和 2、2 和 3、5 和 1 条指令之间的数据相关可以被消除，但不能消除第 3 和 4 条指令之间的 load-use 冒险。

10 次循环共有 10 个时钟的 load-use 阻塞；此外，对于 bne 控制（分支）冒险，第 1 次和最后 1 次预测错误，所以有 2 次需要对预取执行的指令进行冲刷。因为条件检测和转移目标地址修改都在“执行 (Ex)”阶段进行，因此，分支延迟损失时间片（分支延迟槽）为 2，也即每次冲刷掉 2 条指令。因此，2 次共被冲刷掉 4 条指令，使流水线阻塞了 4 个时钟周期；对于最后一条 jump 指令，因为在“译码 (ID)”阶段进行跳转目标地址修改，所以每次有一个时钟阻塞，10 次循环 jump 指令共执行了 9 次，因而有 9 个时钟周期的阻塞。综上所述，10 次循环总共有 $10 + 4 + 9 = 23$ 次阻塞，且第 1~4 条指令各执行了 10 次，第 5~6 条指令各执行了 9 次，因此，10 次循环所用的时钟周期数为 $4 \times 10 + 2 \times 9 + 23 = 81$ ，总时间为 $81 \times 200\text{ps} = 16.2\text{ns}$ 。执行上述程序段时，流水线处理器的速度大约是单周期处理器速度的 $34.8 / 16.2 = 2.15$ 倍；大约是多周期处理器速度的 $44.6 / 16.2 = 2.75$ 倍。

四、(1) 16 位虚拟地址中低 7 位为页内偏移量，高 9 位为虚页号；虚页号中高 7 位为 TLB 标记，低 2 位为 TLB 组索引。

(2) 12 位物理地址中低 7 位为页内偏移量，高 5 位为物理页号。

(3) 地址 067AH = 0000 0110 0111 1010B，所以，虚页号为 0000011 00B，映射到 TLB 的第 00 组，将 0000011B = 03H 与 TLB 第 0 组的四个标记比较，虽然和其中一个相等，但对应的有效位为 0，其余都不等，所以 TLB 缺失，需要访问主存中的慢表。直接查看 0000011 00B = 00CH 处的页表项，有效位为 1，取出物理页号 19H = 11001B，和页内偏移 111 1010B 拼接成物理地址：11001 111 1010B。根据中间 4 位 1110 直接找到 cache 第 14 行（即：第 E 行），有效位为 1，且标记为 33H = 110011B，正好等于物理地址高 6 位，故命中。根据

物理地址最低两位 10，取出字节 2 中的内容 4AH=01001010B。

五、 A: world!hello

B: world!
hello

C: hello
World!

六、(1) 因为该外设接口中有一个 32 位数据缓存器，所以，若用中断方式进行输入/输出的话，可以每 32 位数据进行一次中断请求，因此，中断请求的时间间隔为 $106 \times 4B / 40kB = 100\mu s$ 。

对应的中断服务程序的执行时间为 $(1/1GHz) \times 500 \times 106 = 0.5\mu s$ ，因为中断响应过程就是执行一条隐指令的过程，所用时间相对于中断处理时间（即执行中断服务程序的时间）而言，几乎可以忽略不计，因而整个中断响应并处理的时间大约 $1\mu s$ 多一点，远远小于中断请求的间隔时间。因此，可以用中断方式进行该外设的输入输出。若用中断方式进行该设备的输入/输出，则该设备持续工作期间，CPU 用于该设备进行输入/输出的时间占整个 CPU 时间的百分比大约为 $0.5/100 = 0.5\%$ （也可以通过考察 1 秒钟内 500M 个时钟周期中有多少时钟周期用于中断来计算百分比，其计算公式为 $(106/100 \times 500)/500M = 1\%$ ）。

(2) 若外设的最大传输率为 4MBps，则中断请求的时间间隔为 $106 \times 4B / 4MB = 1\mu s$ 。而整个中断响应并处理的时间大约 $0.5\mu s$ 多一点，中断请求的间隔时间和中断响应处理时间太接近，虽然可以用中断方式进行该外设的输入输出，但不太合适

若用周期挪用 DMA 方式，则一秒钟内产生的 DMA 次数为 $4MB / 1000B = 4000$ ；每次 DMA 传送前都需要 2000 个时钟周期进行 DMA 初始化和 DMA 结束处理，所以，CPU 用于 DMA 处理的总开销为 $4000 \times 2000 = 8000000 = 8M$ 个时钟周期；而 CPU 的时钟频率为 1GHz，即 CPU 每秒钟内产生 1000M 个时钟周期，故 CPU 用于该外设 I/O 的时间占整个 CPU 时间的百分比 $8M / 1000M = 0.8\%$ （也可通过考察相邻两次 DMA 请求间隔时间内 CPU 用于该外设 I/O 的时间来计算，即 $(2000 \times 1/1GHz) / (1000B / 4MB) = 0.8\%$ ）。

历年真题部分

西北工业大学 2001 年研究生入学考试

一.(10 分)名词解释

向量中断：是指一种识别中断源的技术或方式。识别中断源的目的就是要找到中断源对应的中断服务程序的入口地址的地址，即获得向量地址

系统总线：计算机系统内各功能部件之间相互连接的总线

微程序：完成某一个指令的一系列微指令结合。

寻址方式：处理器根据指令中给出的地址信息来寻找有效地址的方式，是确定本条指令的数据地址以及下一条要执行的指令地址的方法。

存储单元：存储单元是存储器中可存放一个字或若干字节的基本单位

二.(20 分)填空

答案速查：1、 $-1 \sim 1 - 2^{-n+1}$ $-2^{n-1} \sim 2^{n-1} - 1$ 2、补码 3、10 32 $15.6\mu s$

4、提高访存速度 硬件 扩充主存容量 操作系统和硬件 逻辑地址 物理地址 地址映射

5、指令 寄存器 主存 2 6、总线请求 (BR) 总线允许 (BG) 总线忙 (BS)

3.【解析】DRAM 芯片行列地址复用，地址线需要 7 根，数据线 1 根、片选线 1 根，读写控制线 1 根，如果是

选择题的话读写控制线也可能是两根；异步刷新的周期为 2/128

三.(8 分)答：CPU 主要由运算器和控制器两大部分组成

运算器	暂存寄存器：用于暂存从主存读来的数据。 累加寄存器：通用寄存器，暂时存放 ALU 运算的结果，可以作为加法的入端。 程序状态字寄存器：保留由算数逻辑运算指令或测试指令的结果而建立的各种状态信息。 通用寄存器组：如 AX、BX、CX、DX、SP 等，用于存放操作数（包括源操作数、目的操作数及中间结果）和各种地址信息。
控制器	程序计数器：用于指出下一条指令在主存中的存放地址。 指令寄存器：用于保存当前正在执行的那条指令。 地址寄存器：用于存放所要访问的主存单元的地址。 数据寄存器：用于存放向主存写入的信息或从主存中读出的信息。

四.(7 分)【解析】可画出时空图，可直观计算出

1、 $80(t_{fh}+t_{d1}+t_{d2}+t_{ex})$ 2、 $80(t_{fh}+t_{d1}+t_{d2})+t_{ex}$ 3、 $80t_{fh}+t_{d1}+t_{d2}+t_{ex}$

五.(15 分)①停止 CPU 访问内存：当外设要求传送一批数据时，由 DMA 控制器发一个信号给 CPU。DMA 控制器获得总线控制权后，开始进行数据传送。一批数据传送完毕后，DMA 控制器通知 CPU 可以使用内存，并把总线控制权交还给 CPU。

②周期挪用：CPU 访存时，必须待存取周期结束后，CPU 再将总线占有权让出，当同时请求访存，CPU 暂时放弃总线占有权，由 I/O 设备挪用一個或几个周期，既实现了 I/O 传送，又较好的发挥了主存与 CPU 的效率，适用于 I/O 设备读写周期大于主存周期。

③DMA 与 CPU 交替访内：一个 CPU 周期可分为 2 个周期，一个专供 DMA 控制器访内，另一个专供 CPU 访内。不需要总线使用权的申请、建立和归还过程。适用于 CPU 工作周期比主存周期长的情况。

六.(10 分)【解析】计算机字长为 16 位，故指令长为 16 位。有 64 条指令，故操作码为 6 位。四种寻址方式，故寻址特征为 2 位。CPU 内部有 4 个通用寄存器，故寄存器为 2 位

1、

操作码	寻址特征	X	操作数
6 位	2 位	2 位	6 位

2、立即寻址： $0\sim 2^6-1$ 直接寻址： 2^6 间接寻址： 2^{16} 相对寻址： 2^{16}

七.(6 分)计算机体系结构是程序员所看到的计算机的属性，即概念性结构与功能特性。按照计算机系统的多级层次结构，不同级程序员所看到的计算机具有不同的属性。包括指令集、数据类型、存储器寻址技术等，大都属于抽象的属性。计算机组成指的是系统结构的逻辑实现，包括机器机内的数据流和控制流的组成及逻辑设计等。确定主存周期、逻辑上是否采用并行主存、逻辑设计等。

计算机实现是计算机组成的物理实现。如选择存储芯片类型、微组装技术、线路设计等
计算机组成是计算机系统结构的逻辑实现。一种体系结构可以有多种组成。一种组成可以有多种实现。

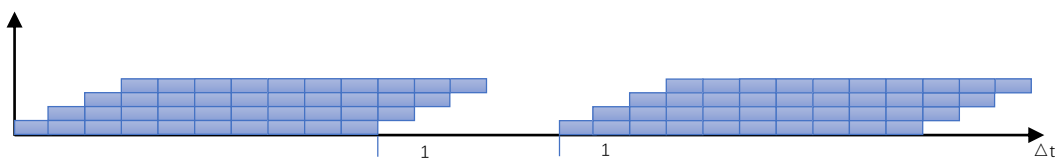
八.(6 分)Cache 的透明性指 cache 功能的具体实现对程序员不可见。程序员不需要理解其内部实现原理。

Cache 的功能主要采用硬件实现，以实现透明性。

九.(10 分)中央处理器访问主存的逻辑地址分解成页号 a 和页内地址 b，并对页号 a 进行地址变换，即将逻辑页号 a 作为索引，先查找快表。若找到要访问的页，便修改页表项中的访问位，然后利用页表项中给出的物理块号和页内地址形成物理地址。若在快表中未找到该页的页表项，应到内存中去查找页表，再对比页表项中的状态位，看该页是否已调入主存，未调入则产生缺页中断，请求从外存把该页调入主存。若内存中没有空闲块，还需根据页面置换算法淘汰某页。

最后将得到的物理页号同页内地址组合得到物理地址，根据物理地址从主存中存取必要的信息。

十.(8 分)吞吐率 = $\frac{20}{(13 \times 2 + 2)\Delta t} = \frac{5}{7}\Delta t$ 效率 = $\frac{n \text{ 个任务占用 } k \text{ 时空区有效面积}}{n \text{ 个任务所用时间与 } k \text{ 个流水段所围成的时空区总面积}} = \frac{4 \times 10 \times 2}{28 \times 4} = 71.4\%$



西北工业大学 2002 年研究生入学考试(460)

一.(10 分)名词解释

随机存储器：存储器的任何一个存储单元的内容都可以随机存取，而且存取时间与存储单元的物理地址无关

取指周期：是指令周期的第一个阶段，也称取指令阶段。即控制单元执行程序周期内，根据 PC 的内容从主存中取出所要执行的指令并存放在 IR 中的过程。

流水线：把一个重复的过程分解为若干子过程，每个子过程与其他子过程并行执行

通道：一个独立于 CPU 的专门 I/O 控制的处理器，控制设备与内存直接进行数据交换

程序：完成某一个功能的一系列指令结合。

二.(10 分)请解释下列中断方式

简单中断：当处理机接受中断时，只需暂停一个或几个周期而不执行处理程序的中断

自愿中断：是在编写程序中产生的，能在程序的任意一个地方执行，转而去执行一些特定的。

向量中断：是指一种识别中断源的技术或方式。识别中断源的目的就是要找到中断源对应的中断服务程序的入口地址的地址，即获得向量地址

多重中断：当 CPU 执行某个中断服务程序的过程中，发生了更高级、更紧迫的事件，CPU 暂停现行中断服务程序的执行，转去处理该事件的中断，处理完返回现行中断服务程序继续执行的过程

单重中断：处理一个中断请求期间不允许被其他中断请求打断

三.(5 分)试说明下列寄存器的作用

程序计数器(PC)：用于指出下一条指令在主存中的存放地址。

指令寄存器(IR)：用于保存当前正在执行的那条指令。

标志寄存器(FR)：保留由算术逻辑运算指令或测试指令的结果而建立的各种状态信息

存储器地址寄存器(MAR)：用于存放所要访问的主存单元的地址。

存储器缓冲寄存器(MBR)：用于存放向主存写入的信息或从主存中读出的信息。

四.(5 分)说明下列寻址方式寻找操作数的方法

立即寻址：指令的地址字段即操作数

直接寻址：指令的地址字段即操作数在内存中的地址

间接寻址：指令的地址字段即存放操作数地址的地址

寄存器操作数寻址：操作数在所给的寄存器中存放

相对寻址：指令的地址字段加上程序计数器（PC）的内容形成操作数的有效地址

五.(36 分)回答下列问题

1.采用简单的单总线结构，虽然可以构成计算机系统，但是它的信息吞吐量受到限制，工作效率和计算机的使用范围受到很大的限制。随着对微机性能越来越高的要求，现代微机的体系结构已不再采用单总线或双总线的结构，而是采用更复杂的多总线结构。

2.总线接口：连接在总线上的设备与总线的连接电路；外设的速度与 CPU 的速度不匹配

3.指令流：指机器执行的指令序列；数据流：指由指令流调用的数据序列。

所有的程序和数据都是以二进制码在存储器中存放的。

当 EIP 指针指向哪 CPU 就加载那段内存中的数据,如果是不正确的指令格式,CPU 就会发生错误中断. 被 CS:IP 指定的是程序，也可以作为数据，所有的都可以作为数据用，但被 CS:IP 指定的通常不作为数据用。

4.指令：指挥机器工作的指示和命令

微指令：若干微命令的集合，存放微指令的控制存储器的单元地址称为微地址

程序：一系列按一定顺序排列的指令

微程序：完成某一个指令的一系列微指令结合

关系：①一条机器指令对应一个微程序，这个微程序是由若干条微指令构成的。因此，一条机器指令的功能是若干条微指令组成的序列来实现的。简而言之，一条机器指令所完成的操作划分成若干条微指令来完成，由微指令进行解释和执行。

②从指令与微指令，程序与微程序，地址与微地址的一一对应关系上看，前者与内存储器有关，而后者与控制存储器（它是微程序控制器的一部分。微程序控制器主要由控制存储器、微指令寄存器和地址转移逻辑三部分组成。其中，微指令寄存器又分为微地址寄存器和微命令寄存器两部分）有关，与此相关也有相对应的硬设备。

③从一般指令的微程序执行流程图可以看出。每个 CPU 周期就对于一条微指令。这就告诉我们怎么设计微程序，也将使得我们进一步体验到机器指令很微指令的关系。

5.①RISC 更能充分利用 VLSI 芯片的面积，CISC 的控制器大都采用微程序控制，其控制存储器在芯片所占面积为 50%以上，而 RISC 控制器采用组合逻辑控制，其硬布线逻辑只占 CPU 芯片 10%左右

② RISC 更能提高运算速度，RISC 指令数寻址方式和指令格式种类少，又设有多个通用寄存器，采用流水线技术，所以运算速度更快，大多数指令在一个时钟周期内完成。

③RISC 便于设计，可降低成本，提高可靠性，指令系统简单，机器设计周期短，逻辑简单，可靠性高

④RISC 有利于编译程序代码优化，代码执行更高效

六.(10 分) $[X]_{\text{原}}=0.10100000$ $[X]_{\text{补}}=0.10100000$ $[X]_{\text{反}}=0.01000000$

$[-X]_{\text{补}}=0.01100000$ $[X]_{\text{移}}=1.10100000$ $[X]_{8421}=0.0110\ 0010\ 0101$

$[X+Y]_{\text{补}}=0.0001111$ $[X-Y]_{\text{补}}=1.0010001$ 不正确，溢出

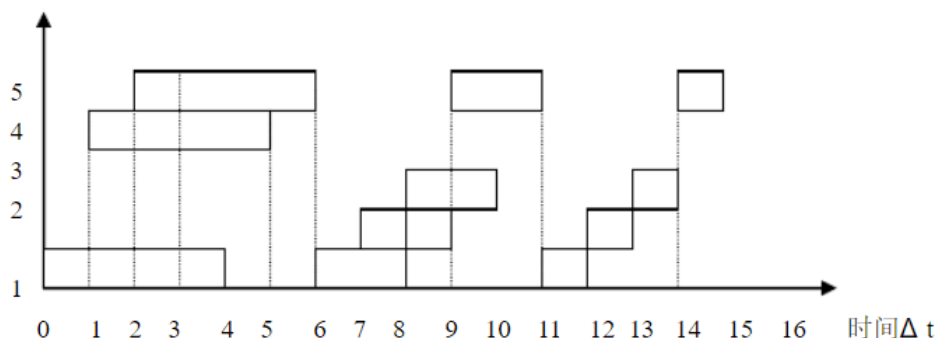
七.(4 分)1.5 2.4 【解析】如果仅考虑纠正 1 位错的情况，只要满足 $2^k \geq n+k+1$ 就可以了(设校验位的位数为 k，信息位的位数为 n)。此题中因为 $n=7$ ，所以 $k \geq 4$ 。

八.(10 分)(1): 2048—4095 5120—6143 7168—8191

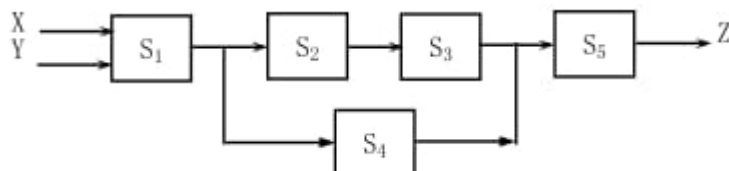
(2) 0---3072; 1023---4095; 1024---1024; 3728---无; 4096---2048; 7800---无

九.(10 分)【解析】应先选择适合于静态流水线工作的算法使完成向量点积 A、B 所用的时间最短。本题可连续计算 $a_1 \times b_1$ 、 $a_2 \times b_2$ 、 $a_3 \times b_3$ 、 $a_4 \times b_4$ 4 个乘法，然后功能切换，按 $((a_1 \times b_1 + a_2 \times b_2) + (a_3 \times b_3 + a_4 \times b_4))$ 经过三次加法来求得最后的结果，在 15 个 Δt 时间内流传 7 个结果。其实际吞吐率 TP 为 $7/(15\Delta t)$ ，而顺序方式所用的时间为

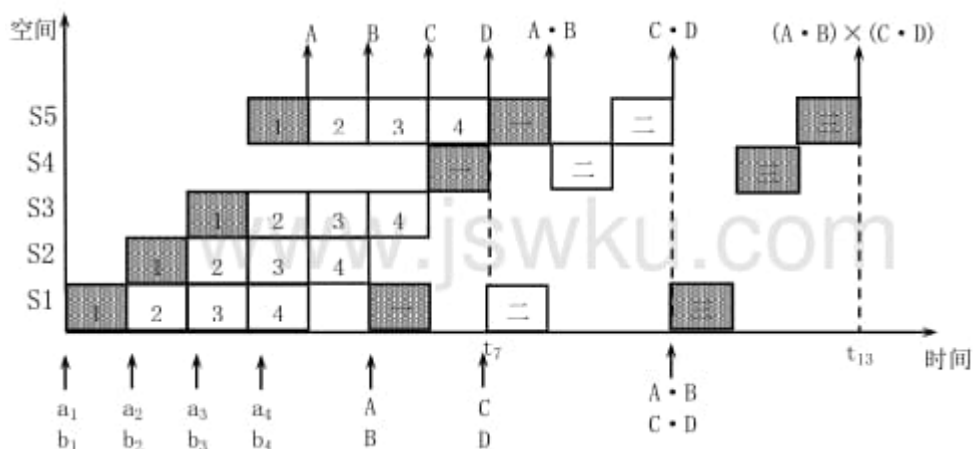
$4 \times 3\Delta t + 3 \times 4\Delta t = 24\Delta t = 24 \times 10\mu s = 240\mu s$ ，该流水线的 $\eta = \frac{4 \times 3\Delta t + 3 \times 4\Delta t}{5 \times 15\Delta t} = 32\%$



【扩展】类似题：有一条 5 个功能段的线性动态多功能流水线如图所示，其中 1→2→3→5 功能段组成加法流水线，1→4→5 功能段组成乘法流水线，设每个功能段的延迟时间均相等为 Δt 。用这条流水线计算 $F = \prod_{i=1}^4 (a_i + b_i)$ ，画出流水线时空图，并计算流水线的实际吞吐率、加速比和效率。



【解】由于该流水线为动态双功能流水线，计算要求先加后乘，因此应先设置加法功能，连续计算出 (a_1+b_1) 、 (a_2+b_2) 、 (a_3+b_3) 、 (a_4+b_4) 四个加法后，再设置乘法功能，而且按照 $[(a_1+b_1) \times (a_2+b_2)] \times [(a_3+b_3) \times (a_4+b_4)]$ 顺序做 3 个乘法。因此可画出该流水线的时空图如图所示，图中 $A = a_1+b_1$ ， $B = a_2+b_2$ ， $C = a_3+b_3$ ， $D = a_4+b_4$



由时空图可以看出，在总共 12 个 Δt 的时间内输出 7 个结果，所以有： $TP = n/T_n = 7/12 \Delta t$

而当用串行方法完成操作时，需要四次加法和三次乘法，完成一次加法需要 $4 \Delta t$ ，完成一次乘法需要 $3 \Delta t$ ，完成该运算总共需要时间为： $T_0 = 4 \times 4 \Delta t + 3 \times 3 \Delta t = 25 \Delta t$ ；所以 $S = T_0/T_n = 2.08$ ； $E = \text{有效时空区面积} / \text{全部时空区面积} = (4 \times 4 \Delta t + 3 \times 3 \Delta t) / (5 \times 12 \Delta t) = 0.42$

西北工业大学 2003 年研究生入学考试(460)

一.(40 分)填空题

- 1、硬件系统 软件系统 应用软件 2、10011 4 3、16 4

- 4、双稳态触发器 栅极电容 5、主存-辅存 主存容量 6、硬件和操作系统 直接
7、程序访问的局部性原理 CPU 与主存速度不匹配 8、PC 操作数
9、CISC RISC 10、分析指令 执行指令 11、控制存储器 水平型 垂直型 水平型 并行
12、同步 异步 13、时序 电平 14、接口 总线控制权 15、关中断 保护现场
16、改变中断处理的优先级 屏蔽一些不允许产生的中断 17、IDE USB
18、控制逻辑 地址寄存器

二.(30 分)简答题

2.CRC 码被 $G(x)$ 整除, 所得余数与出错位之间有唯一对应关系

3.先行进位解决进位的传递速度问题。让每一位的进位与其低一位的进位无关, 仅与两个参加操作的数以及最低位的进位有关。由于每位的操作数是同时给出的, 各进位信号几乎可以同时产生, 和数也随之产生, 所以先行进位可以提高进位的传递速度, 从而提高加法器的运算速度。

4.RAM, 它相当于 PC 机上的移动存储, 用来存储和保存数据的。它在任何时候都可以读写, RAM 通常是作为操作系统或其他正在运行程序的临时存储介质 (内存)

ROM, 它相当于 PC 机上的硬盘, 用来存储和保存数据。ROM 数据不能随意更新, 但是在任何时候都可以读取。即使是断电, ROM 也能够保留数据。但是资料一旦写入后只能用特殊方法或根本无法更改, 因此 ROM 常在嵌入式系统中担任存放作业系统的用途。

两者的最大区别是 RAM 在断电以后保存在上面的数据会自动消失, 而 ROM 就不会。

易失性: DRAM、SRAM、Cache 非易失性: ROM、光盘

5.操作码和地址码 采用程序计数器实现顺序执行和转移

6.相同点:这两种方式下, 主机和 I/O 设备都是并行工作。

DMA 指的是直接内存访问。外部设备与计算机主存之间直接进行数据交换, 而无需经过 CPU。中断控制方式: 程序的切换需要保护和恢复现场, 对中断请求的相应发生在每条指令执行完毕时, 数据传送为程序传送。

DMA: 除了预处理和后处理, 不占用 CPU 资源, 对 DMA 请求的响应发生在每个机器周期结束, 传送过程不需要 CPU 干预, 数据传输率高, 请求优先级高, 数据传输靠硬件传送

三.(30 分)判断改错题

答案速查: $\times \times \sqrt{\times} \sqrt{\quad} \sqrt{\quad} \times \sqrt{\times}$

1、错【解析】规格化尾数有原码形式和补码形式 2、错【解析】都考虑了进位

4、错【解析】访存时间与存储器容量无关 8、错【解析】计算机速度不仅仅由主频决定

10、错【解析】不可以同时使用总线, 同一时间只允许其中一个使用总线

四.(10 分) (1) 原码: 0.0010111 反码: 0.0010111 补码: 0.0010111 移码: 1.0010111

(2) 原码: 无 反码: 无 补码: 10000000 移码: 00000000

(3) 原码: 1.1000110 反码: 1.0111001 补码: 1.0111001 移码: 0.0111001

(4) 原码: 无 反码: 无 补码: 1.0000000 小数无移码定义

(5) 原码: 10000001 反码: 11111110 补码: 11111111 移码: 01111111

五.(10 分)【解析】逻辑左移时将操作数看做无符号数; 算数左移时, 只有当操作数为补码形式的负数是, 右移需补 1

1.[X] 算数左移: 1.0110 算数右移: 1.01011

逻辑左移: 1.0110 逻辑右移: 1.01011

[Y] 算数左移: 0.0110 算数右移: 0.01011

逻辑左移: 1.0110 逻辑右移: 0.01011

2.2[-Y]_反=1.1000 [-X]_补=0.1011

六.(10 分)【解析】由题意得, 主存容量为 256K, 故需要 18 位二进制才可完全表示。故地址字段为 18 位, 又知该计算机共有 64 条指令, 故操作码 OP 为 6 位, 相加可得指令长为 32 位。

(1) 18 6 32

(2) 可采用移位寄存器将基址寄存器内地址左移 4 位, 然后加上偏移地址

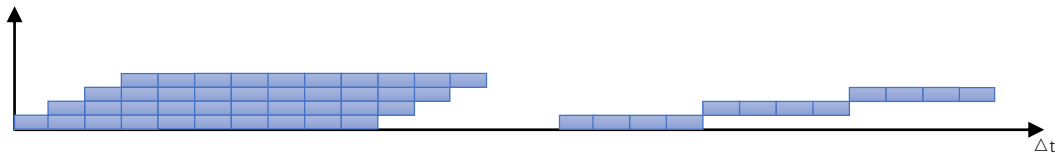
七.(10 分)(1)100ns (2)200ns (3)100ns

【解析】流水线的操作周期应等于每一流水操作的最长时间。(2) 问可画出时空图易得

(3) 由于相邻指令发生数据相关, 故还需延时 100ns 等待上一条指令运算完毕

(王道书上有原题, 计算机组成原理(唐朔飞)上也有原题)

八.(10 分)假设一条指令的执行可分为四个步骤, 如下图所示, 显然流水 CPU 具有更高的吞吐量



西北工业大学 2004 年研究生入学考试(814)

五、(15 分)1.二进制: 10011111 八进制: -37 十六进制: -1F 反码: 11100000

补码: 11100001 移码: 01100001 8421 码: - 0011 0001

2.结果为: 1.10011111 无溢出

六、(8 分) (1) C0A0000F (2) 114.25

七、(12 分)顺序存储: 从始端按顺序查找 (磁带)

直接存储: 先直接指出该存储器的某个区域再顺序查找 (磁盘)

随机存储: 可以直接访问任意存储单元 (RAM、ROM)

关联存储: 按内容查找 相联存储器 (快表)

八、(8 分)定长操作码: 固定长度操作码是指操作码所占的二进制位数固定不变, 指令字最高位若干部分固定长度表示操作码。有利于简化硬件设计, 提高指令译码和识别速度

可变操作码: 是操作码扩展技术的应用, 即操作码的长度是可变的, 且分散地放在指令的不同字段中。这种格式有利于压缩程序中操作码的平均长度, 在字长较短的微型机中被广泛应用。

九、(15 分)指令流: 指机器执行的指令序列; 数据流: 指由指令流调用的数据序列。

所有的程序和数据都是以二进制码在存储器中存放的。

当 EIP 指针指向哪 CPU 就加载那段内存中的数据,如果是不正确的指令格式,CPU 就会发生错误中断. 被 CS:IP 指定的是程序, 也可以作为数据, 所有的都可以作为数据用, 但被 CS:IP 指定的通常不作为数据用。

十、(17 分)中断: 中断就是使 CPU 暂停运行原来的程序而应更为急迫事件的需要转向去执行为中断源服务的程序, 待该程序处理完后, 再返回运行原程序。

中断向量: 由发出中断请求的设备通过输入输出总线主动向 CPU 发出一个识别代码。

向量中断: 是指一种识别中断源的技术或方式。识别中断源的目的就是要找到中断源对应的中断服务程序的入口地址的地址, 即获得向量地址

CPU 处理中断的过程: 关中断, 保存断点, 引出中断服务程序, 保存现场和屏蔽字, 开中断, 执行中断服务程序, 关中断, 恢复现场和屏蔽字, 开中断, 中断返回

CPU 通过数据总线获得中断向量。因为地址总线为单向的。

西北工业大学 2004 年研究生入学考试(401)

八、解释下列问题(6 分)

(1) 153 (2) -103 (3) 0.1953125 (4) 25 (5) 76 (6) 99 (8421 码)

九、填空(24 分)

1、1111101001 2、11010010 3、1.0111 (溢出) 4、33*4MB 5、4 6、64 25

7、62.5 μ s 【解析】刷新周期为 8/128ns

8、指令 149

9、B17880H 【解析】由于 cache 块的大小为 128 字节，可用 7 位二进制表示，使 b178af 低 7 位为 0 即可

10、 $(2^9-1) \times (2^5-1)$ (可能不对)

十、选择题(12 分，每个题有一个以上的正确答案，请将正确的答案全部选择出来)

1、BF 【解析】考察中断向量的概念，中断向量是中断服务程序的入口地址，是通过数据线传输的，因为地址线是单向的，故不可以用地址线传输

2、BCEF [解析]注意题目问的是描述不正确的，控制存储器中存放的是微程序，并不是程序。，也不是存放的指令。

3、ADH 【解析】主存储器采用的是半导体存储器，一般选用 DRAM，SRAM 一般用于 cache。

4、CE 【解析】DMA 方式是由硬件实现的，CPU 并不参与数据交换

5、AB 【解析】考察微操作的概念

十一、(8 分)顺序执行和并发执行

PC 自加自动生成下一条指令地址。或转移指令（通过改变 PC 值实现）

西北工业大学 2004 年研究生入学考试(401)

一、问答题(共 10 分)

1、软件层次 2、以运算器为中心 3、略 4、停止 CPU 访存

二、简答题(共 25 分，每小题 5 分)

1、先行进位解决进位的传递速度问题。让每一位的进位与其低一位的进位无关，仅与两个参加操作的数以及最低位的进位有关。由于每位的操作数是同时给出的，各进位信号几乎可以同时产生，和数也随之产生，所以先行进位可以提高进位的传递速度，从而提高加法器的运算速度。

2、SRAM 的存储元是用双稳态触发器来记录信息的。

DRAM 是利用存储元电路中的栅极电容上的电荷来存储信息的。

3、通过不同的时间段来区分指令和数据，即在取指令阶段（或取指微程序）取出的为指令，在执行指令阶段（或相应微程序）取出的即为数据。

通过地址来源区分，由 PC 提供存储单元地址的取出的是指令，由指令地址码部分提供存储单元地址的取出的是操作数。

4、在物理层提高总线的性能只要是提高总线信号速度，其主要措施有：增加总线的宽度，增加传输数据的数据场长度，缩短总线长度，降低信号电平，采用差分信号，采用多条总线等等。

在逻辑层可通过改进总线协议提高总线的性能。具体措施有：简化总线传输协议，采用总线复用技术，采用消息传输协议。

5、由于一般外围设备都有他自身的独立时钟，故吧他们接入主机时，必须解决两个异步工作的系统之间的同步或通讯联络问题；由于外围设备的工作速度远远比主机慢，有的相差达几个数量级。故将他们接入主机时，必须解决数据格式的转换问题。

三、计算题(共 40 分，每小题 10 分，4 和 5 小题任选一题)

1、 $16/(2 \times 2) = 4\text{MIPS}$ $16/5 = 3.2\text{MIPS}$

2、200ns 两个执行周期长的可采用两个机器周期完成执行周期操作

3、【解析】分别计算出有 cache 和没有 cache 两种情况所需要的时间。题目没给出是先在 cache 中查找，如没找到则在主存中查找。还是同时查找 cache 和主存，若 cache 找到，则停止访存。默认为同时查找 cache 和主存。

有 cache：执行 5t 个指令所需的时间为： $20 \times 98\% \times 5t + 20 \times 95\% \times t + 120 \times 2\% \times 5t + 120 \times 5\% \times t = 135t$

无 cache：执行 5t 个指令所需的时间为： $120 \times 5t + 120 \times t = 720t$

$(720/135) - 1 = 4.33$ 即大约快了 4.33 倍

4、(1)

OP	A1	A2
----	----	----

(2)

0000	A1	A2	15 条二地址指令
0001	A1	A2	
.....	
1110	A1	A2	
1111	000000	A2	32 条一地址指令
1111	000001	A2	
.....	
1111	011111	A2	
1111	100000	000000	110 条零地址指令
1111	100000	000001	
.....	
1111	100001	101101	

(多种画法，结果不唯一)

(3) $(15 \times 4 + 32 \times 10 + 110 \times 16) / (15 + 32 + 110) = 13.63$

5、 10^{38} 约 2^{27} ， \therefore 阶取 8 位，含一个符号位。 10^{-7} 约 $2^{-20} \times 2^{-4} = 2^{-24}$ ， \therefore 尾数取 24 位，另加一个符号(数符)阶码采用移码，才能使 0 全为机器数。 \therefore 浮点数格式为

阶码	数浮	尾数
0 7	8 9	32

西北工业大学 2007 年研究生入学考试(814)

一、完成下列表示(8 分)

二进制原码: 11111011 八进制: 173 十六进制: -7B

反码: 10000100 补码: 10000101 移码: 00000101 8421 码: -0001 0010 0011

二、填空(15 分)

- 1、 2^{16} 2、1 3、存储元 4、快 低 高 5、寄存器 辅存
6、11010101 7、00101000 8、总线控制器 9、一条微指令
10、下一条指令的地址 11、块 12、中断服务程序入口地址

三、名词解释(15 分)

- 1、标志寄存器: 保留由算数逻辑运算指令或测试指令的结果而建立的各种状态信息。
- 2、即插即用: 指计算机上加一个新的外部设备时, 能自动侦测与配置系统的资源, 而不需要重新配置或手动安装驱动程序。
- 3、微操作: 一条机器指令可以分解成一个微操作序列, 是计算机中最基本、不可分解的操作。
- 4、存储单元: 存储单元是存储器中可存放一个字或若干字节的基本单位。
- 5、指令周期: 是取出一条指令并执行这条指令的时间。一般由若干个机器周期组成, 是从取指令、分析指令到执行完所需的全部时间。

四、回答下列问题(22 分)

- 1、(1)寄存器寻址 (2)寄存器间接寻址 (3)立即寻址 (4)直接寻址 (5)相对寻址
- 2、同步通信: 指由统一时钟控制的通信, 总线周期的长度是固定的, 控制方式简单, 灵活性差, 当系统中各部件工作速度差异较大时, 总线工作效率明显下降。适合于速度差别不大的场合。具有高的传输频率, 缺点: 同步总线必须按最慢的模块来设计公共时钟, 当各模块存取时间相差很大是, 会大大损失总线效率。
异步通信: 指没有统一时钟控制的通信, 部件间采用应答方式进行联系, 控制方式较同步复杂, 灵活性高, 当系统中各部件工作速度差异较大时, 有利于提高总线工作效率。优点: 总线周期长度可变, 不把响应时间强加到功能模块上, 因而允许快速和慢速的功能模块都连接到同一总线上; 缺点: 总线复杂, 成本较高。

五、应用题(15 分)

(1) 因为采用断定方式, 所以要给出下一条微指令的地址, 而控制存储器容量为 512×32 位, 所以后继地址字段应为 9 位。又知控制微程序转移的条件有 4 个, 故判断测试字段应为 3 位, 操作控制字段为 $32-9-3=20$ 位。

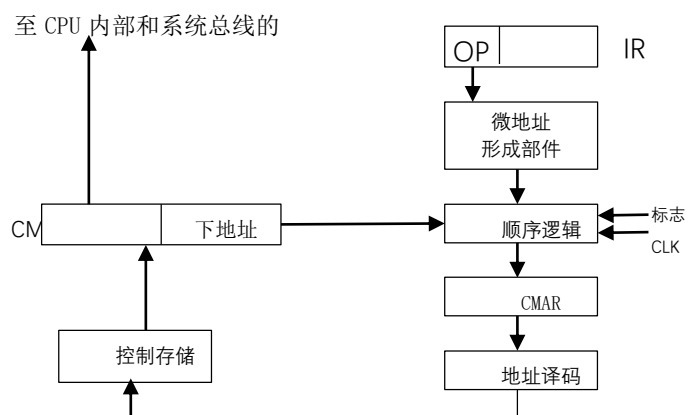
操作控制 (20 位)	判断测试字段 (3 位)	后继地址字段 (9 位)
-------------	--------------	--------------

(2) 控制寄存器: 核心部件, 用于存放各指令对应的微程序 (ROM)

微指令寄存器: 用于存放从 CM 中取出的指令

微地址形成部件：用于产生初始微地址和后继微地址

微地址寄存器：接受微地址形成部件送来的微地址



2009 年研究生入学考试计算机统考 408

一.单项选择题：每小题 2 分。

答案速查：CDDCD CAADB DA

11. C 【解析】考查指令的执行过程。

通常完成一条指令可分为取指阶段和执行阶段。在取指阶段通过访问存储器可将指令取出；在执行阶段通过访问存储器可以将操作数取出。这样，虽然指令和数据都是以二进制代码形式存放在存储器中，但 CPU 可以判断在取指阶段访问存储器取出的二进制代码是指令；在执行阶段访存取出的二进制代码是数据。

12. D 【解析】考查符号位的扩展。

结合题干及选项可知，int 为 32 位，short 为 16 位；又 C 语言的数据在内存中为补码形式，故 x、y 的机器数写为 0000007FH、FFF7H。

执行 $z=x+y$ 时，由于 x 是 int 型，y 为 short 型，故需将 y 的类型强制转换为 int，在机器中通过符号位扩展实现，由于 y 的符号位为 1，故在 y 的前面添加 16 个 1，即可将 y 强制转换为 int 型，其十六进制形式为 FFFFFFF7H。然后执行加法，即 $0000007FH + FFFFFFF7H = 00000076H$ ，其中最高位的进位 1 自然丢弃。故选 D。

13. D 【解析】考查浮点加法运算。

根据题意，X 可记为 00,111; 00,11101 (分号前为阶码，分号后为尾数)，Y 可记为 00,101; 00,10100。

首先对阶，X、Y 阶码相减，即 $00,111-00,101=00,111+11,0111=00,010$ ，可知 X 的阶码比 Y 的阶码大 2，根据小阶向大阶看齐的原则，将 Y 的阶码加 2，尾数右移 2 位，可得 Y 为 00,111; 00,00101。

尾数相加，即 $00,11101+00,00101=01,00010$ ，尾数相加结果符号位为 01，故需进行右规。

规格化，将尾数右移 1 位，阶码加 1，得 $X+Y$ 为 01,000; 00,1000，阶码符号位为 01，说明发生溢出。

14. C 【解析】考查 Cache 与主存的映射方式。

由于 Cache 共有 16 块，采用 2 路组相联，因此共有 8 组，0, 1, 2, ..., 7。主存的某一字块按模 8 映射到 Cache 某组的任一字块中，即主存的第 0, 8, 16...字块可以映射到 Cache 第 0 组 2 个字块的任一字块中，而 129 号单元是位于第 4 块主存块中，因此将映射到 Cache 第 4 组 2 个字块的任一字块中。

【注意】由于在计算机系统结构中和计算机组成原理的某些教材中介绍的组相联跟此处的组相联并不相同，导致部分考生理解错题目。考生应以真题为准，以后再出现类似题目，应以此种解答为标准。

15. **D** 【解析】考查存储器的扩展。

首先确定 ROM 的个数, ROM 区为 4KB, 选用 $2K \times 8$ 位的 ROM 芯片, 需要 $\frac{4K \times 8}{2K \times 8} = 2$ 片, 采用字扩展方式; 60KB 的 RAM 区, 选用 $4K \times 4$ 位的 RAM 芯片, 需要 $\frac{60K \times 8}{4K \times 4} = 30$ 片, 采用字和位同时扩展方式。

16. **C** 【解析】考查相对寻址。

相对寻址 $EA = (PC) + A$, 首先要求的是取指令后 PC 的值。转移指令由两个字节组成, 每取一个字节 PC 自动加 1, 因此取指令后 PC 值为 2002H, 故:

$$EA = (PC) + A = 2002H + 06H = 2008H。$$

17. **A** 【解析】考查 RISC 的特性。

相对于 CISC 计算机, RISC 计算机的特点是指令条数少; 指令长度固定, 指令格式和寻址种类少; 只有取数/存数指令访问存储器, 其余指令的操作均在寄存器之间进行; CPU 中通用寄存器多; 大部分指令在一个或者小于一个机器周期内完成; 以硬布线逻辑为主, 不用或者少用微程序控制。

18. **A** 【解析】考查流水线中时钟周期的特性。

时钟周期应以最长的执行时间为准, 否则用时长的流水段的功能将不能正确完成。

19. **D**。考查硬布线控制器的特点。

硬布线控制器的速度取决于电路延迟, 所以速度快; 微程序控制器采用了存储程序原理, 每条指令都要访控存, 所以速度慢。硬布线控制器采用专门的逻辑电路实现, 修改和扩展困难。

20. **B** 【解析】考查总线的基本概念。

总线带宽是指单位时间内总线上可传输数据的位数, 通常用每秒钟传送信息的字节数来衡量, 单位可用字节/秒 (B/s) 表示。根据题意可知, 在 $2 \times (1/10\text{MHz})$ 秒内传输了 4B, 所以 $4B \times 10\text{MHz} / 2 = 20\text{MB/s}$ 。

21. **D** 【解析】考查 Cache 的命中率。

命中率 = Cache 命中的次数 / 所有访问次数, 有了这个公式这道题就很容易看出, 要注意的一点是看清题, 题中说明的是缺失 50 次, 而不是命中 50 次, 仔细审题是做对题的第一步。

22. **A** 【解析】考查中断的分类。

选项中能引起外部中断的只能是输入设备键盘。

二.综合应用题

43. (8 分) (1) 按题意, 外设每秒传送 0.5MB, 中断时每次传送 4B。中断方式下, CPU 每次用于数据传送的时钟周期为 $5 \times 18 + 5 \times 2 = 100$ 。

为达到外设 0.5MB/s 的数据传输率, 外设每秒申请的中断次数为 $0.5\text{MB} / 4B = 125\ 000$ 。1s 内用于中断的开销为 $100 \times 125\ 000 = 12\ 500\ 000 = 12.5\text{M}$ 个时钟周期。

CPU 用于外设 I/O 的时间占整个 CPU 时间的百分比为 $12.5\text{M} / 500\text{M} = 2.5\%$ 。

(2) 当外设数据传输率提高到 5MB/s 时, 改用 DMA 方式传送, 每次 DMA 传送 5 000B, 1s 内需产生的 DMA 次数为 $5\text{MB} / 5\ 000B = 1\ 000$ 。

CPU 用于 DMA 处理的总开销为 $1\ 000 \times 500 = 500\ 000 = 0.5\text{M}$ 个时钟周期。

CPU 用于外设 I/O 的时间占整个 CPU 时间的百分比为 $0.5\text{M} / 500\text{M} = 0.1\%$ 。

【评分说明】如果考生只给出正确的计算结果, 未给出计算过程, 每个给 2 分。

44. (13 分) 【解答】题干已给出取值和译码阶段每个节拍的功能和有效控制信号, 我们应以弄清楚取指阶段中数据通路的信息流动作为突破口, 读懂每个节拍的功能和有效控制信号。然后应用到解题思路中, 包括划分执行步骤、确定完成的功能、需要的控制信号。

先分析题干中提供的示例 (本部分解题时不做要求):

取指令的功能是根据 PC 的内容所指主存地址, 取出指令代码, 经过 MDR, 最终送至 IR。这部分和后面

的指令执行阶段的取操作数、存运算结果的方法是相通的。

C1: (PC)→MAR

在读写存储器前，必须先将地址（这里为(PC)）送至 MAR。 C2: M(MAR)→MDR, (PC)+1→PC

读写的数据必须经过 MDR，指令取出后 PC 自增 1。 C3: (MDR)→IR

然后将读到 MDR 中指令代码送至 IR 进行后续操作。

指令“ADD (R1),R0”的操作数一个在主存中，一个在寄存器中，运算结果在主存中。根据指令功能，要读出 R1 的内容所指的主存单元，必须先将 R1 的内容送至 MAR，即(R1)→MAR。而读出的数据必须经过 MDR，即 M(MAR)→MDR。

因此，将 R1 的内容所指主存单元的数据读出到 MDR 的节拍安排如下：

C5: (R1)→MAR C6: M(MAR)→MDR

ALU 一端是寄存器 A，MDR 或 R0 中必须有一个先写入 A 中，如 MDR。 C7: (MDR)→A

然后执行加法操作，并将结果送入寄存器 AC。 C8: (A)+(R0)→AC

之后将加法结果写回到 R1 的内容所指主存单元，注意 MAR 中的内容没有改变。

C9: (AC)→MDR C10: (MDR)→M(MAR)

有效控制信号的安排并不难，只需看数据是流入还是流出，如流入寄存器 X 就是 Xin，流出寄存器 X 就是 Xout。还需注意其他特殊控制信号，如 PC+1、Add 等。

于是得到参考答案如下：

时钟	功能	有效控制信号
C5	MAR←(R1)	R1out, MARin
C6	MDR←M(MAR)	MemR, MDRinE
C7	A←(MDR)	MDRout, Ain
C8	AC←(A)+(R0)	R0out, Add, ACin
C9	MDR←(AC)	ACout, MDRin
C10	M(MAR)←(MDR)	MDRoutE, MemW

本题答案不唯一，如果在 C6 执行 M(MAR)→MDR 的同时，完成(R0)→A（即选择将(R0) 写入 A），并不会发生总线冲突，这种方案可节省 1 个节拍，见下表。

时钟	功能	有效控制信号
C5	MAR←(R1)	R1out, MARin
C6	MDR←M(MAR), A←(R0)	MemR, MDRinE, R0out,
C7	AC←(MDR)+(A)	MDRout, Add, ACin
C8	MDR←(AC)	ACout, MDRin
C9	M(MAR)←(MDR)	MDRoutE, MemW

2010 年研究生入学考试计算机统考 408

一.单项选择题

答案速查：DBBDA DBADA D

12 . D 【解析】考查计算机的性能指标。

I . CPU 的时钟频率，也就是 CPU 主频率，一般说来，一个时钟周期内完成的指令数 是固定的，所以主频越高，CPU 的速度也就越快，程序的执行时间就越短。

II. 数据在功能部件之间传送的路径称为数据通路, 数据通路的功能是实现 CPU 内部的运算器和寄存器以及寄存器之间的数据交换。优化数据通路结构, 可以有效提高计算机系统的吞吐量, 从而加快程序的执行。

III. 计算机程序需要先转化成机器指令序列才能最终得到执行, 通过对程序进行编译优化可以得到更优的指令序列, 从而使得程序的执行时间也越短。

13. B 【解析】考查定点数的运算。

用补码表示时 8 位寄存器所能表示的整数范围为 $-128 \sim +127$ 。由于 $r1=-2$, $r2=-14$, $r3=-112$, $r4=-8$, 则 $r2 \times r3 = 1568$, 结果溢出。

14. B 【解析】考查不同精度的数在计算机中的表示方法及其相互转换。

由于 (int) $f=1$, 小数点后面 4 位丢失, 故 II 错。IV 的计算过程是先将 f 转化为双精度浮点数据格式, 然后进行加法运算, 故 $(d+f)-d$ 得到的结果为双精度浮点数据格式, 而 f 为单精度浮点数据格式, 故 IV 错。

15. D 【解析】考查存储器的组成和设计。

用 $2K \times 4$ 位的芯片组成一个 $8K \times 8$ 位存储器, 每行中所需芯片数为 2, 每列中所需芯片数为 4, 各行芯片的地址分配如下。

第一行 (2 个芯片并联): $0000H \sim 07FFH$ 。第二行 (2 个芯片并联): $0800H \sim 0FFFH$ 。第三行 (2 个芯片并联): $1000H \sim 17FFH$ 。第四行 (2 个芯片并联): $1800H \sim 1FFFH$ 。于是地址 $0B1FH$ 所在芯片的最小地址即为 $0800H$ 。

16. A 【解析】考查半导体随机存取存储器。

一般 Cache 采用高速的 SRAM 制作, 比 ROM 速度快很多, 因此 III 是错误的, 排除法 即可选 A。RAM 需要刷新, 而 ROM 不需要刷新。

17. D 【解析】考查 TLB、Cache 及 Page 之间的关系。

TLB 即为快表, 快表只是慢表 (Page) 的小小副本, 因此 TLB 命中, 必然 Page 也命中, 而当 Page 命中, TLB 则未必命中, 故 D 不可能发生; 而 Cache 的命中与否与 TLB、Page 的命中与否并无必然联系。

18. B 【解析】考查 CPU 内部寄存器的特性。

汇编程序员可以通过指定待执行指令的地址来设置 PC 的值, 而 IR、MAR、MDR 是 CPU 的内部工作寄存器, 对程序员不可见。

19. A 【解析】考查指令流水线的基本概念。

有三种相关可能引起指令流水线阻塞: ①结构相关, 又称资源相关; ②数据相关; ③控制相关, 主要由转移指令引起。数据旁路技术, 其主要思想是不必待某条指令的执行结果送回到寄存器, 再从寄存器中取出该结果, 作为下一条指令的源操作数, 而是直接将执行结果送到其他指令所需要的地方, 这样可以使流水线不发生停顿。

20. D 【解析】考查典型的总线标准。

目前典型的总线标准有: ISA、EISA、VESA、PCI、PCI-Express、AGP、USB、RS-232C 等。

21. A 【解析】考查中断处理过程。

单级中断系统中, 不允许中断嵌套。中断的处理过程为: ①关中断; ②保存断点; ③识别中断源; ④保存现场; ⑤中断事件处理 (开中断、执行中断服务程序、关中断); ⑥恢复现场; ⑦开中断; ⑧中断返回。其中, ①~③由硬件完成, ④~⑧由中断服务程序完成。

22. D 【解析】考查显示器的相关概念。

刷新所需带宽 = 分辨率 \times 色深 \times 帧频 = $1600 \times 1200 \times 24\text{bit} \times 85\text{Hz} = 3916.8\text{Mbit/s}$, 显存总带宽 的 50% 用来刷屏, 于是需要的显存总带宽为 :

$$3916.8\text{Mbit/s} / 0.5 = 7833.6\text{Mbit/s} \approx 7834\text{Mbit/s}.$$

二、综合应用题

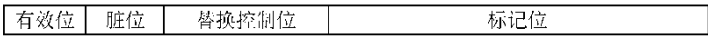
43. (11 分) (1) 操作码占 4 位, 则该指令系统最多可有 $2^4=16$ 条指令; 操作数占 6 位, 寻址方式占 3

位，于是寄存器编号占 3 位，则该机最多有 $2^3=8$ 个通用寄存器；主存容量为 128KB，按字编址，计算机字长为 16 位，划分为 $128KB/2B=216$ 个存储单元，故 MDR 和 MAR 至少各需 16 位。

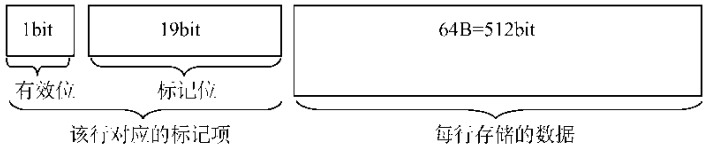
(2) PC 和 R_n 可表示的地址范围均为 $0 \sim 216-1$ ，而主存地址空间为 216，故转移指令的目标地址范围为 $0000H \sim FFFFH$ ($0 \sim 216-1$)。

(3) 汇编语句“add(R_4), (R_5) +”，对应的机器码为 $0010\ 0011\ 0001\ 0101B=2315H$ 。该指令执行后，寄存器 R_5 和存储单元 5678H 的内容会改变。执行后， R_5 的内容从 5678H 变成 5679H。存储单元 5678H 中的内容变成该加法指令计算的结果 $5678H+1234H=68ACH$ 。

44. (12 分) (1) 每个 Cache 行对应一个标记项，如下图所示。

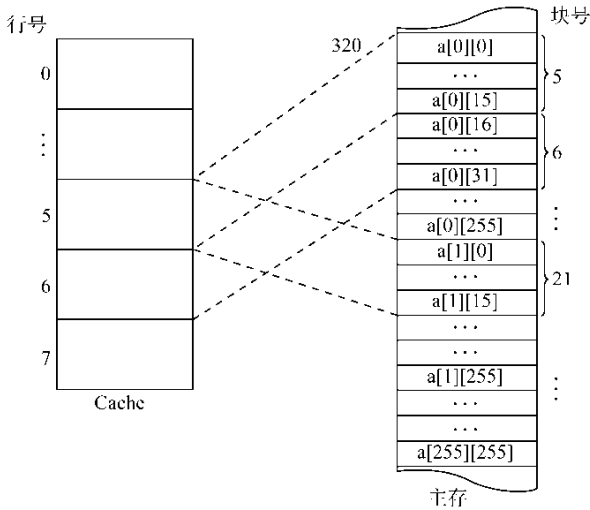


不考虑用于 Cache 一致性维护和替换算法的控制位。地址总长度为 28 位 ($2^{28}=256M$)，块内地址 6 位 ($2^6=64$)，Cache 块号 3 位 ($2^3=8$)，故 Tag 的位数为 $28-6-3=19$ 位，还需使用一个有效位，故题中数据 Cache 行的结构如下图所示。



数据 Cache 共有 8 行，因此数据 Cache 的总容量为 $8 \times (64+20/8)B=532B$ 。

(2) 数组 a 在主存的存放位置及其与 Cache 之间的映射关系如下图所示。



数组按行优先方式存放，首地址为 320，数组元素占 4 个字节。 $a[0][31]$ 所在的主存块对应的 Cache 行号为 $(320+31 \times 4)/64=6$ ； $a[1][1]$ 所在的主存块对应的 Cache 行号为 $(320+256 \times 4+1 \times 4)/64 \% 8=5$ 。

【另解】由 (1) 可知主存和 Cache 的地址格式如下图所示。



数组按行优先方式存放，首地址 320，数组元素占 4 个字节。 $a[0][31]$ 的地址为 $320+31 \times 4=1\ 1011\ 1100B$ ，故其对应的 Cache 行号为 $110B=6$ ； $a[1][1]$ 的地址为 $320+256 \times 4+1 \times 4=1348=101\ 0100\ 0100B$ ，故其对应的 Cache 行号为 $101B=5$ 。

(3) 数组 a 的大小为 $256 \times 256 \times 4B=218B$ ，占用 $218/64=212$ 个主存块，按行优先存放，程序 A 逐行访问数组 a，共需访问的次数为 216 次，未命中次数为 212 次（即每个字块的第一个数未命中），因此程序 A 的

命中率为 $(216-212)/216 \times 100\% = 93.75\%$ 。

【另解】数组 a 按行存放，程序 A 按行存取。每个字块中存放 16 个 int 型数据，除访问的第一个不命中，随后的 15 个全都命中，访问全部字块都符合这一规律，且数组大小为字块大小的整数倍，故程序 A 的命中率为 $15/16 = 93.75\%$ 。

程序 B 逐列访问数组 a，Cache 总容量为 $64B \times 8 = 512B$ ，数组 a 一行的大小为 1KB，正好是 Cache 容量的 2 倍，可知不同行的同一列数组元素使用的是同一个 Cache 单元，故逐列访问每个数据时，都会将之前的字块置换出，也即每次访问都不会命中，命中率为 0。由于从 Cache 读数据比从主存读数据快很多，所以程序 A 的执行比程序 B 快得多。

2011 年研究生入学考试计算机统考 408

一、单项选择题

答案速查：DABDA CDCCD C

12 . D 【解答】考查计算机的性能指标。

MFLOPS 指每秒百万条浮点数运算。

13 . A 【解答】考查浮点数的表示。

x 的二进制表示为 $-1000.01 = -1.000\ 01 \times 2^{11}$ ，根据 IEEE 754，标准隐藏最高位的“1”，又 $E-127=3$ ，所以 $E=130=1000\ 0010B$ ，还因数据存储顺序为 1 位数符+8 位阶码（含阶符）+23 位尾数。故，FR1 内容为 1 10000 0010 0000 10000 0000 0000 0000 000 即，1100 0001 0000 0100 0000 0000 0000 0000=C104000H。

14 . B 【解答】考查随机存取存储器。

随机存取方式是指存储器的任何一个存储单元的内容都可以存取，而且存取时间与存储单元的物理位置无关。选项 A、C、D 均采用随机存取方式，CD-ROM 即光盘，采用串行存取方式。

15 . D 【解答】考查存储器的编制。

按字节编址，64MB 的主存地址空间，MAR 的寻址范围是 64M，故而是 26 位，而实际的主存的空间不能代表 MAR 的位数。

16 . A 【解答】考查寻址方式。

间接寻址不需要寄存器， $EA=(A)$ 。基址寻址： $EA=A+$ 基址寄存器内同；相对寻址： $EA = A+PC$ 内容；变址寻址： $EA = A+$ 变址寄存器内容。

17 . C 【解答】考查条件转移指令。

无符号整数比较 $A>B$ 的情况，bgt 指令会将两个无符号数进行比较，也就是将 A 和 B 相减。 $A-B$ 无进位/借位，也不为 0（为 0 时表示两数相同），故而 CF 和 ZF 均为 0。18 . D 【解答】考查指令流水线的优化。

指令定长、对齐、仅 Load/Store 指令访存，以上 3 个都是 RISC 的特征，使取指令、取操作数操作简化且时间长度固定，能够有效地简化流水线的复杂度。

19 . C 【解答】考查指令周期。

A 由于没有采用指令预取技术，故而取指令要访存；B 时钟周期对指令是不可分割的；C 空指令能够使 PC 寄存器自动加 1，故而不正确；D 在指令执行结束时刻，CPU 查询是否有中断请求，故而可能被中断。

20 . C 【解答】考查总线的特点。

数据线可能传输指令，当取指令时，指令便是在数据线上传输的；操作数显然在数据线上传输。中断类型号是用以指出中断向量地址的，CPU 响应中断请求后，将中断应答信号（INTR）发回到数据总线上，CPU 从数据总线上读取中断类型号后，就可以通过这个中断类型号在中断向量表上找到该设备的中断服务程序入口

地址，转入中断服务程序，而握手（应 答）信号应该在通信总线上传输。

21 . **D** 【解答】考查中断屏蔽字。

高优先级置 0 表示可被中断，比该中断优先级低（相等）的置 1 表示不可被中断，只能屏蔽 L3 和其自身。

22 . **C** 【解答】考查程序查询方式。

每秒 200 次查询，每次 500 个时钟周期，则每秒最少占用 $200 \times 500 = 100000$ 个时钟周期，占 CPU 时间比为 $100000 \div 50M = 0.20\%$ 。

二、综合应用题

43. (11 分) (1) $134 = 128 + 6 = 1000\ 0110B$, 所以 x 的机器数为 $1000\ 0110B$, 故 $R1$ 的内容为 $86H$ 。 $246 = 255 - 9 = 1111\ 0110B$, 所以 y 的机器数为 $1111\ 0110B$ 。 $x - y$: $1000\ 0110 + 0000\ 1010 = (0)1001\ 0000$, 括弧中为加法器的进位, 故 $R5$ 的内容为 $90H$ 。 $x + y$: $1000\ 0110 + 1111\ 0110 = (1)0111\ 1100$, 括弧中为加法器的进位, 故 $R6$ 的内容为 $7CH$ 。

(2) m 的机器数与 x 的机器数相同, 皆为 $86H = 1000\ 0110B$, 解释为带符号整数 m (用补码表示) 时, 其值为 $-111\ 1010B = -122$ 。 $m - n$ 的机器数与 $x - y$ 的机器数相同, 皆为 $90H = 1001\ 0000B$, 解释为带符号整数 $k1$ (用补码表示) 时, 其值为 $-111\ 0000B = -112$ 。

(3) 能。 n 位加法器实现的是模 $2n$ 无符号整数加法运算。对于无符号整数 a 和 b , $a + b$ 可以直接用加法器实现, 而 $a - b$ 可用 a 加 b 的补数实现, 即 $a - b = a + [-b]_{\text{补}} \pmod{2n}$, 所以 n 位无符号整数加/减运算都可在 n 位加法器中实现。

由于带符号整数用补码表示, 补码加/减运算公式为: $[a + b]_{\text{补}} = [a]_{\text{补}} + [b]_{\text{补}} \pmod{2n}$, $[a - b]_{\text{补}} = [a]_{\text{补}} + [-b]_{\text{补}} \pmod{2n}$, 所以 n 位带符号整数加/减运算都可在 n 位加法器中实现。

(4) 带符号整数加/减运算的溢出判断规则为: 若加法器的两个输入端 (加法) 的符号相同, 且不同于输出端 (和) 的符号, 则结果溢出, 或加法器完成加法操作时, 若次高位的进位和最高位的进位不同, 则结果溢出。

最后一条语句执行时会发生溢出。因为 $1000\ 0110 + 1111\ 0110 = (1)0111\ 1100$, 括弧中为加法器的进位, 根据上述溢出判断规则, 可知结果溢出。

44. (12 分) (1) 虚拟地址为 24 位, 其中高 12 位为虚页号。物理地址为 20 位, 其中高 8 位为物理页号。

(2) 20 位物理地址中, 最低 5 位为块内地址, 中间 3 位为 Cache 行号, 高 12 位为标志。

(3) 在主存中。虚拟地址 $001C60H = 0000\ 0000\ 0001\ 1100\ 0110\ 0000B$, 故虚页号为 $0000\ 0000\ 0001B$, 查看 $0000\ 0000\ 0001B = 001H$ 处的页表项, 由于对应的有效位为 1, 故虚拟地址 $001C60H$ 所在的页面在主存中。

页表 $001H$ 处的页框号 (物理页号) 为 $04H = 0000\ 0100B$, 与页内偏移 $1100\ 0110\ 0000B$ 拼接成物理地址: $0000\ 0100\ 1100\ 0110\ 0000B = 04C60H$ 。

对于物理地址 $0000\ 0100\ 1100\ 0110\ 0000B$, 所在主存块只能映射到 Cache 的第 3 行 (即第 $011B$ 行); 由于该行的有效位=1, 标记 (值为 $105H$) $\neq 04CH$ (物理地址高 12 位), 故访问该地址时 Cache 不命中。

(4) 虚拟地址 $024BACH = 0000\ 0010\ 0100\ 1011\ 1010\ 1100B$, 故虚页号为 $0000\ 0010\ 0100B$; 由于 TLB 只有 $8/4 = 2$ 个组, 故虚页号中高 11 位为 TLB 标记, 最低 1 位为 TLB 组号, 它们的值分别为 $0000\ 0010\ 010B$ (即 $012H$) 和 $0B$, 因此, 该虚拟地址所对应物理页面只可能映射到 TLB 的第 0 组。

由于组 0 中存在有效位=1、标记= $012H$ 的项, 所以访问 TLB 命中, 即虚拟地址 $024BACH$ 所在的页面在主存中。

西北工业大学 2012 年研究生入学考试(801)

一、(本题满分 20 分)计算题

- 1、 $[X]_{\text{补}}=0.1001$ $[Y]_{\text{补}}=1.0101$ $[-Y]_{\text{补}}=0.1011$ $[X+Y]_{\text{补}}=1.1110$ $[X-Y]_{\text{补}}=1.0100$ (溢出)
2、 $X=0.1011 \times 2^{10}$ $Y=0.0101 \times 2^{10}$

二、(本题满分 30 分)回答下列问题

- 1、(8 分)总线带宽：单位时间内总线上可传输数据的位数，通常用每秒钟传送信息的字节数来衡量。总线带宽 = $16\text{MHz} \times (32/8) = 64\text{MB/s}$
- 2、(4 分)断点：即程序执行中断的地方，保存断点以便处理完中断继续执行该程序
现场：当前所执行程序的 CPU 环境，即 CPU 内各寄存器内的信息
- 3、(10 分)微操作：一条机器指令可以分解成一个微操作序列，是计算机中最基本、不可分解的操作
微指令：若干微命令的集合，存放微指令的控制存储器的单元地址称为微地址
微程序：完成某一个指令的一系列微指令结合
指令：指挥机器工作的指示和命令
程序：一系列按一定顺序排列的指令
一个程序由多条指令组成，一条指令可由一个微程序来解释，一个微程序是一系列微指令的集合，一个微指令可分为一个微操作序列。微操作是最基本的，不可再分的。
- 4、(4 分)DRAM 刷新：刷新的过程实质上是先将原存信息读出，再由刷新放大器形成原信息并重新写入的再生过程。可分为集中刷新，异步刷新，分散刷新。
- 5、(4 分)1 5

西北工业大学 2013 年研究生入学考试(801)

一、(本题满分 10 分)名词解释

定点数：小数点位置固定不变的机器数，小数点不再使用“.”表示，而是约定它的位置。

总线复用：一种信号线在不同的时间传输不同的信息。可以使用较少的线传输更多的信息，从而节省了空间和成本。

有效地址：操作数所在的单元到段首的距离即逻辑地址的偏移地址。

直接寻址：指令字中的形式地址 A 就是操作数的真实地址 EA，即 $EA=A$ 。

存储单元：若干个存储一位二进制代码的存储元件。

二、(本题满分 25 分)回答问题

1. (10 分) (1)提取阶段：由输入设备把原始数据输入给计算机存储器存起来；
(2)解码阶段：根据 CPU 的指令集架构定义将数值解译为指令；

(3)执行阶段：再由控制器把需要处理或计算的数据调入运算器；

(4)最终阶段：由输出设备把最后的结果输出。

2. (15 分)中断源向 CPU 发送中断请求信号；

CPU 结束当前指令的执行，检测中断请求信号；

CPU 向提出请求的中断源发出中断响应信号；

CPU 进行断点保护，PSW、断点地址保存在堆栈中，被响应的中断源撤销中断请求信号，并为 CPU 提供中断处理程序入口地址的相关信息；

CPU 识别中断源，获得中断处理程序入口地址，并将其装入 PC，转向执行中断处理程序；

CPU 执行中断处理程序；

CPU 执行中断返回指令，恢复被保存在堆栈中的断点信息，回到断点处继续执行源程序。

【优点】使处理器的利用率提高，且能支持多道程序和 I/O 设备的并行操作。

【缺点】①各种 I/O 设备都通过中断处理方式进行并行操作，那么中断次数的急剧增加会造成 CPU 无法响应中断和出现数据丢失现象；②如果 I/O 控制器的数据缓冲区比较小，在缓冲区装满数据之后将会发生中断。那么，在数据传送的过程中，发生中断的机会较多，这将消耗大量的 CPU 处理时间。

三、(本题满分 15 分) 完成下列运算(要求写出分析过程)

(1) $4\text{GB}=2^{32}\text{B}$ ； $16\text{MB}=2^{24}\text{B}$

因为 cache 的容量为 2^{24}B ，块大小为 $16\text{B}=2^4\text{B}$ ，所以 cache 的块数是 $2^{24}/2^4=2^{20}$ (块)。

因为主存的容量为 2^{32}B ，块大小为 $16\text{B}=2^4\text{B}$ ，所以主存的块数为 $2^{32}/2^4=2^{28}$ (块)。

(2)两路组相连，一组有 2 块，cache 共分为 $2^{20}/2=2^{19}$ 组，所以组地址是 19 位。因为按字编址，字块大小 16B，一个主存块 4 个字， $16\text{B}/32\text{bit}=4$ ，所以 2 位来表示偏移，字块内地址为 2 位。主存容量为 2^{32}B ，因为是按字编址， $2^{32}\text{B}/4\text{B}=2^{30}$ ，地址位数为 30 位，所以主存字块标记 $30-19-2=9$ 位。所以主存地址地段设计为：

主存字块标记	组地址	字块内地址
9 位	19 位	2 位

西北工业大学 2015 年研究生入学考试(801)

一、(本题满分 8 分)回答问题

A=13.75 B=-11 C=14.046875 D=-8.3125 E=161.00390625

$[X]_{\text{原}}=-13$ $[Y]_{\text{反}}=73$ $[Z]_{\text{补}}=-27$

二、(本题满分 15 分)名词解释

微指令：在微程序控制的计算机中，将控制部件向执行部件发出的各种控制命令称为微命令。

直接寻址：指令字中的形式地址 A 就是操作数的真实地址 EA，即 $EA=A$ 。

浮点数：以适当的形式将比例因子表示在数据中，让小数点的位置根据需要而浮动的机器数。

中断向量：中断服务程序的入口地址。

总线主设备：获得总线控制权的设备。

三、(本题满分 42 分)回答下列问题

1. (12 分)高速器件、Cache、双端口存储器、多模块存储器

2. (15 分)①停止 CPU 访问主存。

【原理】当外设要求传送一批数据时，由 DMA 接口向 CPU 发一个停止信号，要求 CPU 放弃地址线、数据线和有关控制线的使用权。DMA 接口获得总线控制权后，开始进行数据传送，在数据传送结束后，DMA 接口通知 CPU 可以使用主存，并把总线控制权交回给 CPU。

【优点】控制简单。

【缺点】DMA 接口在访问主存时，CPU 基本上处于不工作状态或保持原状态。

②DMA 与 CPU 交替访存

【原理】这种方式适用于 CPU 的工作周期比主存取周期长的情况。例如，CPU 的工作周期是 $1.2\mu s$ ，主存周期小于 $0.6\mu s$ ，那么可将一个 CPU 周期分为 C1 和 C2 两个周期，其中 C1 专供 DMA 访存，C2 专供 CPU 访存。

【优点】不需要总线使用权的申请、建立和归还过程，总线使用权是通过 C1 和 C2 分时控制的。

【缺点】相应的硬件逻辑变得更为复杂。

③周期挪用

【原理】当 I/O 设备没有 DMA 请求时，CPU 按程序的要求访问主存，一旦 I/O 设备有了 DMA 请求，会遇到 3 种情况：

第一种是此时 CPU 不需要访问主存，故 I/O 的访问请求与 CPU 未发生冲突；

第二种是 CPU 正在访存，则必须待存取周期结束后，CPU 再将总线占有权让出；

第三种是 I/O 和 CPU 同时访问请求，出现了访存冲突，此刻 CPU 要暂时放弃总线占有权，由 I/O 设备挪用一或几个存取周期。

【优点】既实现了 I/O 传送，又较好地发挥了主存与 CPU 的效率。

【缺点】可能需要复杂的时序电路，而且数据传输过程是不连续的和不规则的。

3. (5 分)【目的】使两个操作数的小数点位置对齐，即使得两个数的阶码相等。

【原则】先求阶差，然后以小阶向大阶看齐的原则，将阶码小的尾数右移一位，阶码加 1，直到两个数的阶码相等为止。

4. (10 分)指令周期：CPU 从主存中每取出并执行一条指令所需的全部时间。

机器周期：完成一个基本操作所需要的时间。

时钟周期：时钟频率的倒数。

【关系】指令周期常常用若干个机器周期表示，一个机器周期又包含若干个时钟周期。

四、(本题满分 10 分)(要求写出分析过程)

【解】总线的时钟频率为 200MHZ，得 1 个时钟周期为 $0.005\mu s$ ；

一个总线传输周期等于 4 个时钟周期，则一个总线传输周期为 $0.005\mu s \times 4 = 0.02\mu s$ ；

由于总线宽度为 16 位，等于 2B，所以总线数据传输速率为 $2/0.02 = 100\text{MB/s}$ 。

【方法】①保持时钟频率不变，总线宽度改为 32 位；②保持总线宽度不变，时钟频率变为 400MHZ。

西北工业大学 2016 年研究生入学考试(801)

一、选择题(单选, 每小题 2 分, 共 20 分)

答案速查: CCCAD CACAB

1.C 【解析】MAR: 地址寄存器 MDR: 数据寄存器 PC: 程序计数器

4.A 【解析】15 根地址线, 8 根数据线, 1 根片选线, 1 根读写线

5.D 【解析】地址没有正负, 用无符号数

8.C 【解析】字长是 $n+1$ 时, 补码整数范围是 -2^n-2^n-1

9.A 【解析】指令周期由若干个机器周期表示, 一个机器周期又包含若干时钟周期。CPU 周期又称机器周期。存取周期是连续两次独立地访问存储器操作之间所需的最小时间间隔。

二、应用题(共 55 分)

1. (10 分)总线的时钟频率为 200MHZ, 得 1 个时钟周期为 0.005us;

一个总线传输周期等于 4 个时钟周期, 则一个总线传输周期为 $0.005\text{us} \times 4 = 0.02\text{us}$;

由于总线宽度为 32 位, 等于 4B, 所以总线数据传输速率为 $4/0.02 = 200\text{MB/s}$ 。

【方法】①保持时钟频率不变, 总线宽度改为 64 位; ②保持总线宽度不变, 时钟频率变为 400MHZ。

2. (10 分) $[X]_{\text{补}} = 1.1101$, $[Y]_{\text{补}} = 1.0001$, $[X+Y]_{\text{补}} = 1.1101 + 1.0001 = 0.1110$, 有溢出

3. (10 分)①CPU 每秒对鼠标进行 30 次查询, 所需时钟周期数为 $100 \times 30 = 3000$ 。CPU 的时钟频率为 50MHz, 即每秒 50×10^6 个时钟周期, 故对鼠标的查询占用 CPU 的时间比率为 $[3000/(50 \times 10^6)] \times 100\% = 0.006\%$ 。

可见, 对鼠标的查询基本不影响 CPU 的性能。

②对于硬盘, 每 32 位被 CPU 查询一次, 故每秒查询次数为 $2 \times 2^{20}\text{B}/4\text{B} = 512\text{K}$ 。则每秒查询的时钟周期数为 $100 \times 512 \times 1024 = 52.4 \times 10^6$

故对硬盘的查询占用 CPU 的时间比率为 $[52.4 \times 10^6/(50 \times 10^6)] \times 100\% = 105\%$

结论: 以上结果表明, 对鼠标的查询基本不影响 CPU 的性能, 而即使 CPU 将全部时间都用于对磁盘的查询也不能满足磁盘传输的要求, 所以 CPU 一般不采用程序查询方式与磁盘交换信息。

4. (10 分)cache 命中率 $h = N_c/(N_c + N_m) = 4800/(4800 + 200) = 0.96$;

平均访问时间: $T_a = h \cdot T_c + (1-h) \cdot T_m = 30 \times 0.96 + 150 \times (1-0.96) = 34.8\text{ns}$

效率: $e = T_c/T_a = 30/34.8 = 0.86$; 性能提高: $T_m/T_a - 1 = 150/34.8 - 1 = 3.3$ 倍

5. (15 分)【解析】参考《西北工业大学 2013-2014 学年第一学期期末考试》三应用题 4 小题

西北工业大学 2017 年研究生入学考试(801)

一、简答题

1.指令: 计算机执行某种操作的命令。

程序: 使计算机执行一个完整的任务的指令的集合。

CPU 控制程序的执行, 实际是对指令的执行, 那么就是指令执行过程。计算机每执行一条指令都可分为

三个阶段进行。即取指令、分析指令、执行指令。

取指令的任务是：根据程序计数器 PC 中的值从程序存储器读出现行指令，送到指令寄存器。

分析指令阶段的任务是：将指令寄存器中的指令操作码取出后进行译码，分析其指令性质。如指令要求操作数，则寻找操作数地址。

计算机执行程序的过程实际上就是逐条指令地重复上述操作过程，直至遇到停机指令可循环等待指令。

2.通常完成一条指令可分为取指阶段和执行阶段。在取指阶段通过访问存储器可将指令取出；在执行阶段通过访问存储器可以将操作数取出。因此虽然指令和数据都是以二进制代码形式存放在存储器中，但 CPU 可以根据指令周期的不同阶段判断出从存储器取出的二进制代码是指令还是数据。

3.① $4K=2^{12}$ ，cache 地址有 12 位； $4KB/16B=256$ ，则可容纳 256 块；② $128K*32\text{ 位}=512KB=2^{19}B$ ，主存地址为 19 位； $512KB/16B=32768$ ，则可以容纳 32768 块。

【注】题目说主存 $128k*32\text{ 位}$ ，暗示按字寻址，一字 32 位。

4.中断是指在计算机执行现程序的过程中，出现某些急需处理的异常情况或特殊请求，CPU 暂时中止现程序，而转去对这些异常情况或特殊请求进行处理，在处理完毕后 CPU 又自动返回到现程序的断点处，继续执行原程序。

中断方式工作原理：

(1)中断源向 CPU 发送中断请求信号；

(2)CPU 结束当前指令的执行，检测中断请求信号；

(3)CPU 向提出请求的中断源发出中断响应信号；

(4)CPU 进行断点保护，PSW、断点地址保存在堆栈中，被响应的中断源撤销中断请求信号，并为 CPU 提供中断处理程序入口地址的相关信息；

(5)CPU 识别中断源，获得中断处理程序入口地址，并将其装入 PC，转向执行中断处理程序；

(6)CPU 执行中断处理程序；

(7)CPU 执行中断返回指令，恢复被保存在堆栈中的断点信息，回到断点处继续执行源程序。

【优点】使处理器的利用率提高，且能支持多道程序和 I/O 设备的并行操作。

【缺点】①各种 I/O 设备都通过中断处理方式进行并行操作，那么中断次数的急剧增加会造成 CPU 无法响应中断和出现数据丢失现象。

②如果 I/O 控制器的数据缓冲区比较小，在缓冲区装满数据之后将会发生中断。那么，在数据传送的过程中，发生中断的机会较多，这将消耗大量的 CPU 处理时间。

二、计算题

1. X 化成十进制为 0.5625；八进制为 0.44；十六进制为 0.9。

2. Y 的原码为 1.11；反码为 1.00；补码为 1.01；移码为 0.01；BCD 码为 1.01110101。

3. $[X]_{\text{补}}=0.1001$ ， $[Y]_{\text{补}}=1.0100$ ， $[X+Y]_{\text{补}}=[X]_{\text{补}}+[Y]_{\text{补}}=1.1101$ 。

4. $[-Y]_{\text{补}}=1.1100$ ， $[X-Y]_{\text{补}}=[X]_{\text{补}}+[-Y]_{\text{补}}=0.1001+1.1100=0.0101$

三、总线传输速率：单位时间内总线上可传输数据的位数，可以理解为总线带宽。

影响因素：总线工作频率和总线宽度。

时钟频率为 66MHz，那么时钟周期为 $1/66\mu s$ ；总线周期也为 $1/66\mu s$ ；总线带宽为 $4B/(1/66\mu s)=264MB/s$

西北工业大学 2018 年研究生入学考试(801)

一、选择题(单选, 每小题 2 分, 共 20 分)

答案速查: CCAAA BACCB

3、A 【解析】链式查询对故障最敏感, 因为链式查询的总线响应线是串行的从一个部件传送到下一个部件。

4、A 【解析】每个字符一共 11 个比特位, 每秒传输 120 个字符, 所以波特率为 120×11

6、B 【解析】数据线为 32 根, 地址线为 14 根, 共计 46 根

7、A 【解析】若采用程序查询方式, CPU 一旦启动 I/O, 必须停止现程序的运行, CPU 不断查询 I/O 设备状态, 直到外设准备就绪。CPU 和外设是串行工作的。

二、1.(10 分) 【解析】考察补码的加减法, 并判断结果是否溢出, 往年均有类似题目, 较为基础, 需熟练掌握。

2.(10 分) 【解析】总线的工作频率为: $66\text{MHz}/4=16.5\text{MHz}$

总线带宽=总线工作频率* (总线宽度/8) = $16.5\text{MHz} \times (32/8) = 66\text{B/s}$

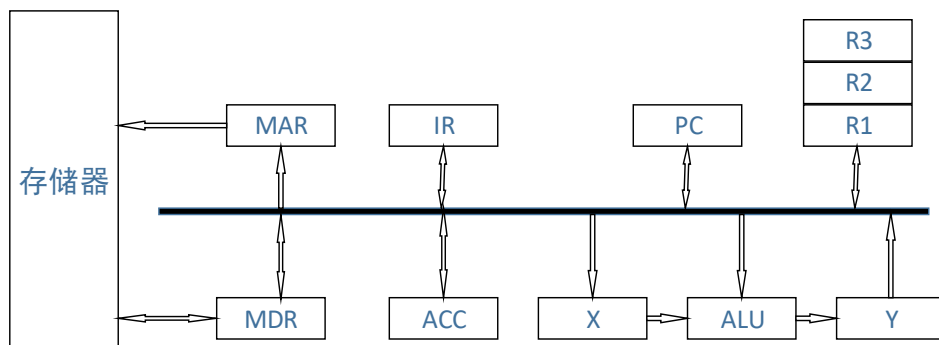
3.(15 分) 【解析】已知 Cache 的容量为 1Gb, 即 128MB; Cache 每块为 64B, 即 Cache 共有 $128\text{MB}/64\text{B}=2\text{M}$ 块。又由题意知, 该 Cache 和主存的映射关系为四路组相联, 即 Cache 可分为 $2\text{M}/4=512\text{K}$ 个组

(1) (2) Cache 共有 512K 个组, 即组地址需要用 19 个二进制位表示。又 Cache 每块 64B, 即块内地址应为 6 位。

主存标记字段	组地址	块内地址
--------	-----	------

(3) 根据所给地址现在 Cache 中查询, 通过查看地址主存标记字段的的有效位是否为 1, 若为 1, 则表示在 cache 中, 若不为 1, 则需调入 cache。

4.(10 分) ADD R1, R2, R3 先进行取指令操作, 根据 PC 的值在存储器中取出该指令, 然后依次将 R1 和 R2 寄存器中的数据分别送入 X 和 ALU, 经过加法计算后, 将 R3 寄存器中的数据送至 X, 将 X 和 ALU 中的数据进行加法运算, 将所得结果送至 Y 寄存器, 最后将数据保存至 R1 寄存器。



西北工业大学 2019 年研究生入学考试(801)

一、选择题(10×2=20 分)

1、B 【解析】考察冯诺依曼计算机的基本特点。冯诺依曼计算机是以运算器为中心, 按地址顺序执行程序。

2、【解析】考察八位二进制补码的范围。八位二进制整数补码: $-2^7 \sim 2^7 - 1$; 八位二进制小数补码: $-1 \sim -2^{-7}$

3、B 【解析】一个字符为 10 位，每秒钟需发送 480 个字符。故数据传送速率应为 $480 \times 10 = 4800\text{bps/s}$

二、问答题

1、【解析】本题考察补码的加减法，同时要求使用一种方法判断结果是否溢出。例如可使用双符号位法判断结果是否溢出，若符号位为 01，则为正溢出，符号位为 10，则为负溢出

2、(10 分) 【解析】CPI 即执行一条指令所需的时钟周期数

P1 方式下总的 CPI 为： $(1 \times 10\% + 2 \times 20\% + 3 \times 50\% + 3 \times 20\%) \times 1.0 \times 10^6 = 2.6 \times 10^6$

P2 方式下总的 CPI 为： $(2 \times 10\% + 2 \times 20\% + 2 \times 50\% + 2 \times 20\%) \times 1.0 \times 10^6 = 2 \times 10^6$

时钟周期即为主频的倒数

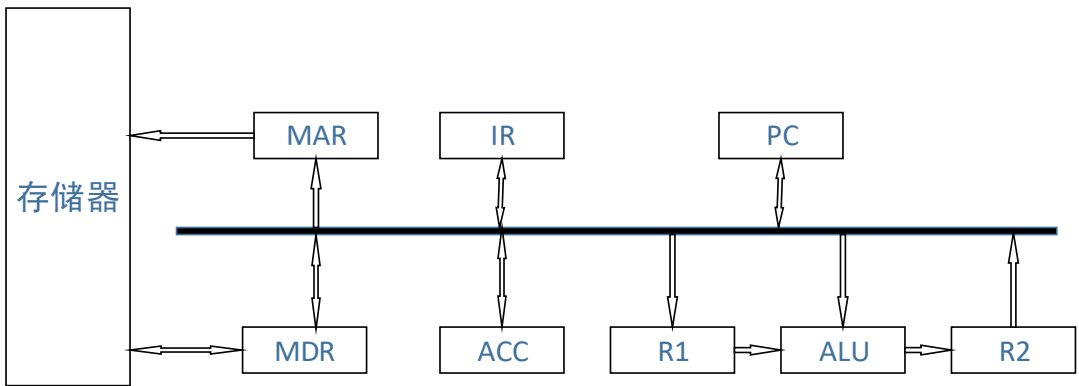
P1 方式下的时钟周期为 $1/2.5\text{GHz}$

P2 方式下的时钟周期为 $1/3\text{GHz}$

3、(10 分) 【解析】参考《西北工业大学 2016 年研究生入学考试(801)》二应用题的 3 小题。

4、【解析】参考王道或课本，均有组相联映射示意图

5、ADD A1, A2, A3 【解析】先进行取指令操作，根据 PC 的值在存储器中取出该指令，根据指令中的操作数地址依次取出操作数 A1 和 A2，分别存放在 R1 和 ALU 中，然后进行加法运算并将结果保存到 R2 中。将 R2 中的结果送到 R1 中，然后再取出操作数 A3 到 ALU 中，并与 R1 中的值相加将结果存放到 R2 中，并将结果保存到地址为 A1 的内存单元中。



西北工业大学 2020 年研究生入学考试(847)

一.填空题(每空 1 分，共 30 分)

二.选择题(每小题 1 分，共 20 分)

三.名词解释(每个 6 分，共 30 分)

总线传输周期：总线传输周期指传送一次数据总线上的数据所花时间。

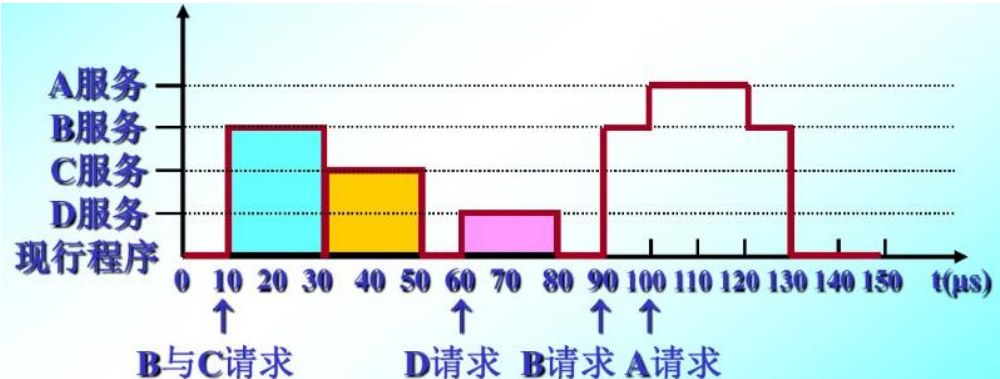
存取周期、存取时间：见《西北工业大学 2009-2010 学年第一学期试题(教育实验学院)》一名词解释(1)小题

RARP：逆地址解析协议。

频分复用：将用于传输信道的总带宽划分成若干个子频带（或称子信道），每一个子信道传输 1 路信号。

四.简答题(每小题 6 分, 共 30 分)

- 1.Cache 命中率= $4800/(4800+200)=0.96$
平均访问时间= $30 \times 0.96 + (150+30) \times (1-0.96) = 36\text{ns}$
效率=访问 Cache 的时间/平均访问时间= $30/36=83.3\%$
性能提高=主存访问时间/平均访存时间= $150/36=4.1$ 倍
- 2.这是一个多重中断的程序运行轨迹, 图中忽略了中断响应时间。



3. (1) 16 在慢启动--拥塞避免算法中, 拥塞窗口初始为 1, 窗口大小开始按指数增长。当拥塞窗口大于慢启动门限后, 停止使用慢启动算法, 改用拥塞避免算法。由于慢启动的门限值初始为 16, 当拥塞窗口增大到 16 时改用拥塞避免算法, 窗口大小按线性增长, 每次增加 1 个报文段。
- (2) TCP 在进行流量控制时是以分组的丢失作为产生拥塞的标志, 发送方通过收到三个重复的确认检测到丢失的报文段。 12 轮次。
- (3) 慢启动--拥塞避免
- (4) 2046、2013 【解析】若甲收到 1 个来自乙的 TCP 段, 该段的序号 seq=1913、确认序号 ack=2046、有效载荷为 100 字节, 则甲立即发送给乙的 TCP 段的序号 seq1=ack=2046 和确认序号 ack1=seq+100=2013。
4. (1) 128.96.39.11, 转发端口为 (A) (2) 128.96.40.1 转发端口为 (C)
- (3) 128.96.40.126, 转发端口为 (C) (4) 192.4.153.11 转发端口为 (D)

五.计算题(每小题 10 分, 共 40 分)

西北工业大学 2020 年研究生入学考试(879)

一.名词解释

CPI: CPU 的指令时钟数。表示每条计算机指令执行所需的时钟周期数。由于不同指令的功能不同, 造成指令执行时间不同, 即指令执行所用的时钟数不同, 所以通常所说的 CPI 是一个平均值。

二.CPI= $2 \times 0.5 + 3 \times 0.35 + 4 \times 0.15 = 2.65$

三.略 四.(1)105 种操作, 故操作码取 7 位; 按题意有 4 种寻址方式, 故寻址方式特征位取 2 位。

指令格式如下:

OP	M	A
----	---	---

7 位 2 位 7 位

指令可直接寻址范围: $2^7=128$; 一次间接寻址范围: $2^{16}=65536$

(2)16MB=8M×16 位, 需要 23 位的地址位才能访问该主存的全部单元

可在(1)指令格式的基础上, 采用双字长指令; 格式如下:

7 位	2 位	7 位
OP	M	A
A2		

形式地址 A=A1/A2 共 23 位；刚好寻址 8M 个 16 位。

五.DMA：直接存储器访问，直接依靠硬件实现主存与外设之间的数据直接传输，传输过程本身不需 CPU 程序干预。

三种访存方式见《西北工业大学 2001 年研究生入学考试》五大题。

西北工业大学 2020 年研究生入学考试(801)

- 一、(15 分) 1.数据总线宽度为 x 位，时钟主频位 zMHZ，所以时钟周期 1/z 微秒，总线周期为 2 个时钟周期，既 2/z 微秒，所以总线理论带宽为 xz/2 Mbps。
- 2.假设计算机中一个字长为 m 位，且 $m < x$ ，那么一个总线周期只能传送 m 位信息，这将使实际带宽低于理论带宽。
- 3.可以将数据总线宽度提高 2 倍或者将时钟主频提高 2 倍。

二、(25 分) 1.寄存器、cache、主存、辅存、光盘

2.计算机中 CPU 和主存速度差异较大，CPU 直接访存会降低 CPU 的效率，因此引入 cache-主存层次结构可以缩短平均访存时间，提高 CPU 利用率；另外主存容量一般较小，为了逻辑上使主存容量扩大，引入了主存-辅存层次结构。

- 3.(1)行内地址占 5 位，所以一个 cache 块大小位 32B，映射位 5 位，所以 cache 总共 32 行
- (2) 一个 cache 块大小为 32B，所以 0、4、8、16 装入 cache 的同一行中，160、180 装到同一行中，130、140 装入到同一行，其他都各占一行。由于从通电开始，所以 cache 初始为空，访问 0、160、130、232、1024、3100、2180 均未命中，所以命中率为：5/12=41.7%

三、(15 分) 1. I/O 中断方式、DMA 控制方式；

程序查询方式：CPU 一旦启动 I/O，CPU 必须停止现行程序的运行，CPU 处在踏步等待的状态。这种方式接口设计简单，设备量少，但 CPU 会花费很多时间进行查询和等待，大大降低了 CPU 的工作效率。

I/O 中断方式：实现了 CPU 和 I/O 设备的并行工作，提高了 CPU 的利用率，但由于数据中的每个字在存储器与 I/O 控制器之间传输必须经过 CPU，这就导致了 I/O 中断方式仍然要消耗较多的 CPU 时间。

DMA 控制方式：也实现了 CPU 与 I/O 设备的并行操作，而且进行数据传输时不需要 CPU 干预，仅在开始和传输结束需要 CPU 干预，进一步提高了 CPU 的利用率，而且 DMA 方式也适合大批数据的传输。

2.由于键盘是字符流设备，以串行方式向 CPU 输入所按的键，DMA 方式适合大批数据的交换，现代计算机一般不采用程序查询方式，所以对键盘的访问一般采用 I/O 中断方式。

四、(10 分) 假设数据长度为 n 位，

当 $X > 0$ ， $Y > 0$ 时， $X_{补}=X$ ， $Y_{补}=Y$ ， $X_{补}-Y_{补}=X-Y=X+(-Y)=X_{补}+(-Y)_{补}$ ；

当 $X > 0$ ， $Y < 0$ 时， $X_{补}=X$ ， $Y_{补}=Y+2^n$ ， $X_{补}-Y_{补}=X+(-Y-2^n)=X_{补}+(-Y-2^n)_{补}$ ，

而 $-Y-2^n < 0$ ， $(-Y-2^n)_{补}=-Y-2^n+2^n=-Y$ ， $-Y > 0$ ， $(-Y)_{补}=-Y$ ，

所以 $X_{补}-Y_{补}=X+(-Y-2^n)=X_{补}+(-Y-2^n)_{补}=X_{补}+(-Y)_{补}$ ；

当 $X < 0$ ， $Y > 0$ 时， $X_{补}=X+2^n$ ， $Y_{补}=Y$ ， $X_{补}-Y_{补}=X+2^n-Y=X_{补}+(2^n-Y)_{补}$ ，

$2^n-Y > 0$ ，所以 $(2^n-Y)_{补}=2^n-Y$ ，而 $2^n-Y=(-Y)_{补}$ ，所以 $X_{补}+(2^n-Y)_{补}=X_{补}+(-Y)_{补}$ ；

当 $X < 0$ ， $Y < 0$ 时， $X_{补}=X+2^n$ ， $Y_{补}=Y+2^n$ ， $X_{补}-Y_{补}=X+2^n-Y-2^n=X-Y=X_{补}+(-Y)_{补}$ ；

五、(10 分) 1.Reduced Instruction Set Computer Complex Instruction Set Computer

2.		CISC	RISC
----	--	------	------

指令数目	复杂，庞大	简单，精简
CPI	各指令执行时间相差很大	绝大数指令在一个 CPI 内完成
寻址方式	寻址方式较多，一般大于 4	寻址方式较少，一般小于 4

本科生期末试卷一

一.选择题(每小题 1 分，共 10 分)

答案速查：DBCBC CACAA

二.填空题(每小题 3 分，共 24 分)

- 1.A.程序 B.地址 C.冯·诺依曼 2.A.浮点数 B.指数 C.对阶
3.A.可编程存储器 B.体系结构 C.固态盘 4.A.物理 B.RR C.RS
5.A.软件 B.操作控制 C.灵活性 6.A.总线带宽 B.传输速率 C. 264MB/s
7.A.刷新 B.显示 C. ROM BIOS 8.A.外围设备 B. DMA 控制器 C. 内存

三. (11 分) (1) 最大正数：数值 = $(2^{31} - 1)_{10}$ ；最大负数：数值 = $-(2^{31} - 1)_{10}$

(2) 最大正数值 = $(1 - 2^{31})_{10}$ ；最大负数值 = $-(1 - 2^{31})_{10}$

四. (11 分) 顺序存储器和交叉存储器读出 4 个字的时间分别是： $t_2 = m T = 4 \times 200\text{ns} = 8 \times 10^{-7} \text{ (s)}$

$t_1 = T + (m - 1)\tau = 200 + 3 \times 50 = 3.5 \times 10^{-7} \text{ (s)}$

顺序存储器带宽是： $W_1 = q / t_2 = 32 \times 10^7 \text{ (位/ S)}$

交叉存储器带宽是： $W_2 = q / t_1 = 73 \times 10^7 \text{ (位/ S)}$

五. (11 分) (1) 操作码字段为 6 位，可指定 $2^6 = 64$ 种操作，即 64 条指令。

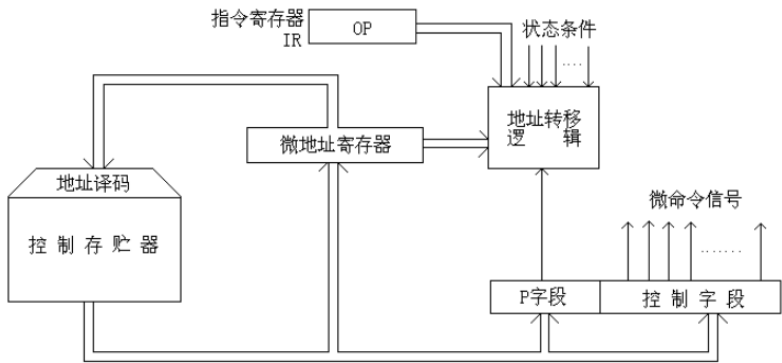
(2) 单字长(32)二地址指令。

(3) 一个操作数在原寄存器(共 16 个)，另一个操作数在存储器中(由变址寄变址寄存器和位移量决定)，所以是 RS 型指令

(4) 这种指令结构用于访问存储器

六. (11 分) 【解】(1) 假设判别测试字段中每一位为一个判别标志，那么由于有 4 个转移条件，故该字段为 4 位，(如采用字段译码只需 3 位)，下地址字段为 9 位，因为控制容量为 512 单元，微命令字段是 $(48 - 4 - 9) = 35$ 位。

(2) 对应上述微指令格式的微程序控制器逻辑框图如下：



其中微地址寄存器对应下地址字段，P 字段即为判别测试字段，控制字段即为微命令子段，后两部分组成微指令寄存器。地址转移逻辑的输入是指令寄存器 OP 码，各状态条件以及判别测试字段所给的判别标志(某

一位为 1)，其输出修改微地址寄存器的适当位数，从而实现微程序的分支转移。

计算机组成原理期末试题及答案(一)

一.选择题(共 20 题, 每题 1 分, 共 20 分)

答案速查: BBDBA CCCBD CCCBC DBCAB

20.B 【解析】16K 是 2 的 14 次方, 也就是 14 根地址线了, 后面的 32 就是数据线的根数了。所以总线数就是 46。

二.填空题(共 7 题, 每空 1 分, 共 20 分)

- 1.硬件 系统软件 应用软件 2.数据总线 控制总线 3.-16~+15 4. 触发器 电容
5.集中式 分散式 异步式 6. 间址 执行 中断 结构 数据
7.直接映射 全相连 组相连

三.简答题(共 2 题, 每题 5 分, 共 10 分)

1. 指令是执行某种操作的命令。一台计算机的所有指令的集合叫做指令系统。
2. ①中断请求②中断判优③中断响应④中断服务⑤中断返回

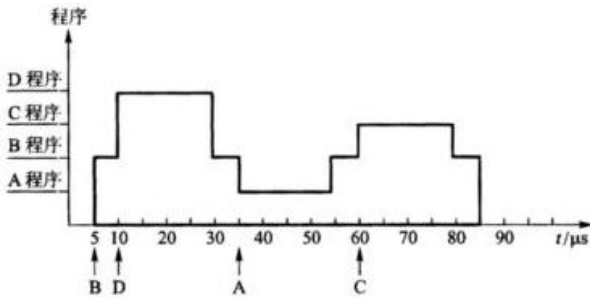
四.应用题(共 5 题.每题 10 分.共 50 分)

1. 【解】 $8/(2*2.5)=1.6\text{MIPS}$ $8/(5*4)=0.4\text{MIPS}$ 结论: 主频相同, 但是机器周期和时钟周期不同的机器, 平均执行指令速度 也不同, 说明主频不是唯一判断计算机性能的指标

2. 【解】(1)对角线必须是 1, 否则会死循环。由于 $D>A>C>B$, D 在 A 前 (故对于 A, 将 A B C 均置为 1, D 置为 0<即不屏蔽 D 的中断请求>); (对于 B, D A C 均在 B 前, 由于 A 本身硬件优先级就在 B 前, 故不需设置屏蔽字, 而 D C 设为 0); (对于 C, D A 在 C 前, 由于 A 本身硬件优先级就在 C 前, 故不需设置屏蔽字, 而 D 设置为 0); (对于 D, 由于是第一个运行, 故屏蔽所有的中断请求, 均置为 1)。每个中断源新的屏蔽字如表所示。

(2)根据新的处理次序, CPU 执行程序的轨迹如图所示。

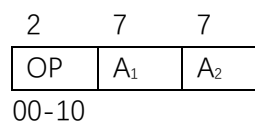
中断码	中断屏蔽字			
	A	B	C	D
A	1	1	1	0
B	0	1	0	0
C	0	1	1	0
D	1	1	1	1



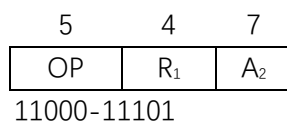
3. 【解】 $\because A = +15 = +0001111, B = +24$
 $+0011000 \therefore [A]_{补} = 0,0001111, [B]_{补} = 0,0011000, [-B]_{补} = 1,1101000$
则 $[A-B]_{补} = [A]_{补} + [-B]_{补} = 0,0001111 + 1,1101000 = 1,1101111$
 $\therefore [A-B]_{补} = 1,1101111$, 故 $A-B = -0001001 = -9$

4. (1)在直接寻址的二地址指令中,根据题目给出直接寻址空间为 128 字,则每个地址码为 7 位,其格式如图(a)所示。3 条这种指令的操作码为 00、01 和 10,剩下的 11 可作为下一种格式指令的操作码扩展用。

(2)在变址寻址的一地址指令中,根据变址时的位移量为-64 ~ +63,形式地址 A 取 7 位。根据 16 个通用寄存器可作为变址寄存器,取 4 位作为变址寄存器 R_1 的编号。剩下的 5 位可作操作码,其格式如图(b)所示。6 条这种指令的操作码为 11000 ~ 11101,剩下的两个编码 11110 和 11111 可作为扩展用。



图(a)



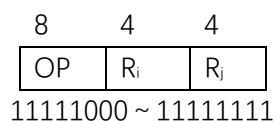
图(b)

(3)在寄存器寻址的二地址指令中,两个寄存器地址 R_i 和 R_j 共 8 位,剩下的 8 位可作为操作码,比格式(2)的操作码扩展了 3 位,其格式如图(c)所示。8 条这种指令的操作码为 11110000 ~ 11110111。剩下的 11111000 ~ 11111111 这 8 个编码可作为扩展用。

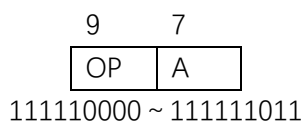
(4)在直接寻址的一地址指令中,除去 7 位的地址码外,可有 9 位操作码,比格式(3)的操作码扩展了 1 位,与格式(3)剩下的 8 个编码组合,可构成 16 个 9 位编码。以 11111 作为格式(4)指令的操作码特征位,12 条这种指令的操作码为 111110000 ~ 111111011,如图(d)所示。剩下的 111111100 ~ 111111111 可作为扩展用。

(5)在零地址指令中,指令的 16 位都作为操作码,比格式(4)的操作码扩展了 7 位,与上述剩下的 4 个操作码组合后,共可构成 4×2^7 条指令的操作码。32 条这种指令的操作码可取 1111111000000000 ~ 1111111000011111,如图(e)所示。

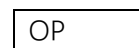
还有 $2^9 - 32 = 480$ 种代码未用,若安排寄存器寻址的一地址指令,除去末 4 位为寄存器地址外,还可容纳 30 条这类指令。



图(c)

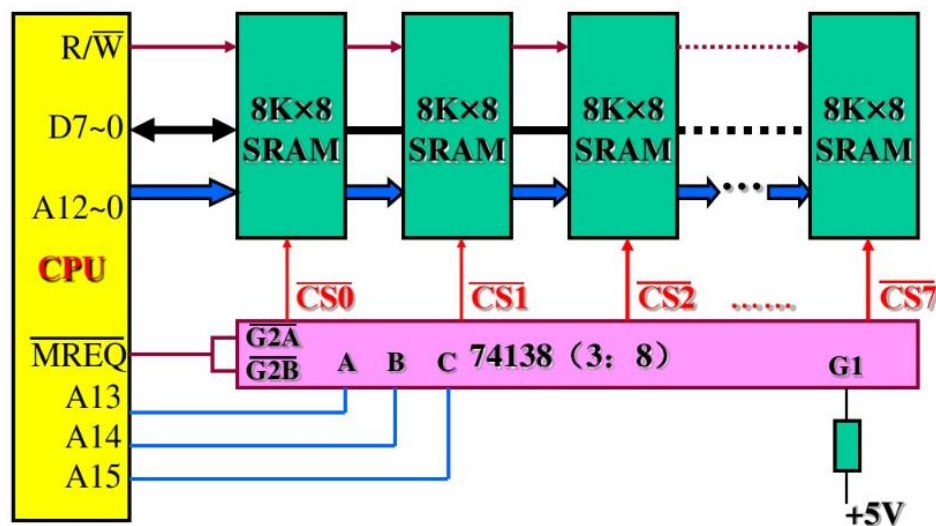


图(d)



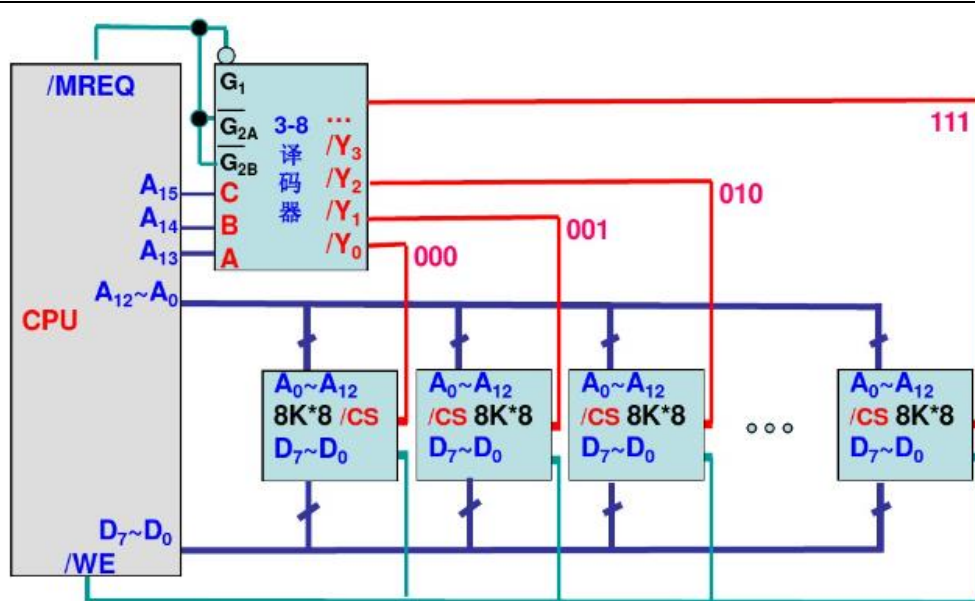
图(e)

5. 【解】(1)CPU 与存储器芯片连接逻辑图:



或:

第一片地址 0 ~ 8K:	000	0 0000 0000 0000 ~ 000	1 1111 1111 1111
第二片地址 8K~16K:	001	0 0000 0000 0000 ~ 001	1 1111 1111 1111
第三片地址 16K~24K:	010	0 0000 0000 0000 ~ 010	1 1111 1111 1111
第四片24K~32K; 第五片32K~40K; 第六片40K~48K; 第七片48K~56K			
第八片地址 56K~64K:	111	0 0000 0000 0000 ~ 111	1 1111 1111 1111



(2)地址空间分配图:

Y0	8K*8 RAM	0~8191
Y1	8K*8 RAM	8192~16383
Y2	8K*8 RAM	16384~24575
Y3	8K*8 RAM	24576~32767
Y4	8K*8 RAM	32768~40959
Y5	8K*8 RAM	40960~49151
Y6	8K*8 RAM	49152~57343
Y7	8K*8 RAM	57344~65535

(3)如果地址线 A13 与 CPU 断线, 并搭接到高电平上, 将会出现 A13 恒为“1”的情况。此时存储器只能寻址 A13=1 的地址空间, A13=0 的另一半地址空间将永远访问不到。若对 A13=0 的地址空间进行访问, 只能错误地访问到 A13=1 的对应空间中去。

【补充】一般还会问: 如果运行时发现无论往哪片 RAM 写入数据后, 以 A000H 为起始地址的存储芯片都有与其相同的数据, 分析故障原因。

【解】如果运行时发现无论往哪片 RAM 写入数据后, 以 A000H 为起始地址的存储芯片都有与其相同的数据, 则根本的故障原因为:

- ①该片的-CS 端与-WE 端错连或短路;
- ②该片的-CS 端与 CPU 的-MREQ 端错连或短路;
- ③该片的-CS 端与地线错连或短路;

在此, 假设芯片与译码器本身都是好的

计算机组成原理期末试题及答案(二)

一.选择题

答案速查: CCDDC AB AABC BDDAC BAACA ABDCE

11.B 【解析】颜色数为 256 色即颜色位数为 8bit。进行 2D 应用时占用显存容量=水平分辨率×垂直分辨率×

颜色位数/8bit=1024x1024x8bit/8bit=1048576byte=1MB。

18.A【解析】因为字符总共有 (7+1+1+1) =10, 波特率是说单位时间内传送的比特数 (比特率是说单位时间内传送的有效比特数), 所以这题的字符传送速率是 9600/10=960b/s。

二.填空题

- 1.A.内存 B.外存 C.内存 2. A.集中式策略 B.分布式策略 C.总线
3. A.结构 B. CPU C.技术无关的开发标准
4. A.高速缓冲存储器 B.速度不匹配 C.指令 cache
5. A. ②CPU③cache①主存⑤磁盘④磁带 B.①③ C.④⑤
6.A.指令流 B. 指令 C.数据流 7. A.数据 B. 算术 C. 逻辑
8. A.高速外围总线 B.同步 C.集中 9.A.寄存器 B.间接 C. 立即寻址
10. A.存储器 B.控制器 C.运算器 11. A.容量 B.存取时间 C.存取周期 12. A.32 B.17 C.32

三.简答题

1.【解】取指周期中从内存读出的信息流是指令流, 而在执行器周期中从内存读出的信息流是数据流

2.【解】指令周期: 取出一条指令并且执行这条指令的时间;

机器周期: 又称 cpu 周期, cpu 一次访问内存的时间, 通常用内存中读取一个指令字的最短时间来规定 CPU 周期

时钟周期: 处理操作的最基本单位。(CPU 的主频)

指令周期、机器周期和时钟周期之间的关系: 指令周期通常用若干个机器周期表示, 而机器周期时间又包含有若干个时钟周期。

3.【解】PCI 总线上有 HOST 桥、PCI/LAGACY 总线桥、PCI/PCI 桥。桥在 PCI 总线体系结构中起着重要作用, 它连接两条总线, 使彼此间相互通信。桥是一个总线转换部件, 可以把一条总线的地址空间映射到另一条总线的地址空间上, 从而使系统中任意一个总线主设备都能看到同样的一份地址表。桥可以实现总线间的猝发式传送, 可使所有的存取都按 CPU 的需要出现在总线上。由上可见, 以桥连接实现的 PCI 总线结构具有很好的扩充性和兼容性, 允许多条总线并行工作。

4.【解】存储器-存储器型时间最长, 寄存器-寄存器型时间最短, 因为前者要两次访问内存, 而后者不用访问内存。

5.【解】Cache 的基本原理: 1.CPU 与 Cache、Cache 与主存之间的数据交换是以块为单位。一个块由通常若干定长的字组成。2.因此 Cache 的基本原理是 当 CPU 字的内存地址同时发给 Cache Cache 该字当前是否已在 Cache CPU CPU CPU 与此同时把含有这个字的数 据块从主存读出 并装入到 Cache 中 Cache 中较旧的内容块替换掉。这种替换控制由始终管理 Cache 使用情况的硬件逻辑电路来实现最常用的替换算法为 LRU。

6.【解】与、或、非、异或, 和 01010101 相与

7.【解】a. ($2^{18} \times 8$)/(32k*8)=8(个); b. ($2^{15} / 2^{12}$)*(8 位/4 位)=16 即(32K*8)/4K* 4=16(片); c. 4*16=64(片)

8.【解】刷新: 对 DRAM 定期进行的全部重写过程;

集中式刷新、分布式刷新。集中式刷新: DRAM 的所有行在每一个刷新周期中都被刷新; 分布式刷新: 每一行的刷新插入到正常的读、写周期之中。

9.【解】原码: 11110001 反码: 10001110 补码: 10001111

10.【解】流水线中有三类数据相关冲突: 写后读(RAW)相关; 读后写(WAR)相关; 写后写 (WAW)相关。

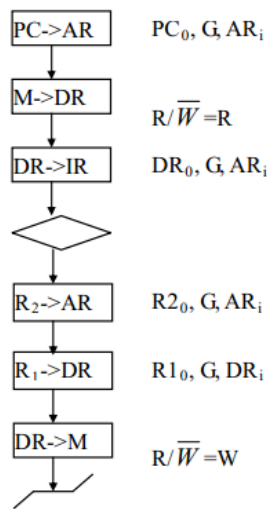
解决数据相关冲突的办法: 在流水 CPU 的运算器中设置若干运算结果缓冲寄存器, 暂时保留运算结果, 以

便于后继指令直接使用，这称为“向前”或定向传送技术。

四.分析与设计

1.
$$h = \frac{N_c}{N_c + N_m} = \frac{3800}{3800 + 200} = 0.95$$
$$r = \frac{t_m}{t_c} = \frac{250ns}{50ns} = 5$$
$$e = \frac{1}{r + (1-r)h} = \frac{1}{5 + (1-5) \times 0.95} = 83.3\%$$
$$t_a = \frac{t_c}{e} = 50ns / 0.833 = 60ns$$

平均访问时间 60ns 效率 83.3%



第 5 题图

2.①双字长二地址指令 ②操作字段 op 为 6 位，可以指定 64 种操作(指令) ③一个操作数在源寄存器(共 4 个)，另一个操作数在存储器中(由变址寄存器和位移量决定)，所以是 RS 型指令

3.【解】40 条指令至少需要操作码字段 6 位，所以剩下的长度为 26 位。主存的容量为 64M 字，则设寻址模式(X)2 位，格式如下：X= 00 直接寻址 有效地址 E=D X= 01 立即寻址 D 字段为立即数 X= 10 变址寻址 有效地址 E=(RX)+D(可寻址 64M 个存储单元) X= 11 相对寻址 有效地址 E=(PC)+D(可寻址 64M 个存储单元) 其中 RX 为变址寄存器(32 位)，PC 为程序计数器(32 位)。在相对寻址时，位移量 D 可正可负

4.数据寄存器 16 位，地址寄存器 17 位 共需要 4 个。 5.见图

6.【解】顺序存储器和交叉存储器连续读出 m=8 个字的信息总量是：64*8=512 位

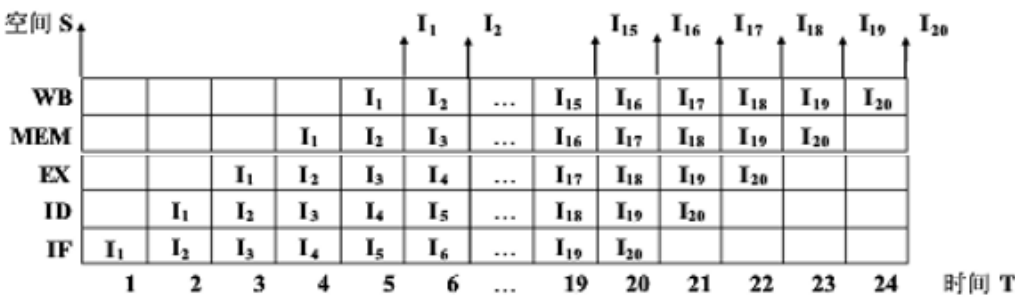
顺序存储器和交叉存储器连续读出 m=8 个字的时间分别是：t1=mT=8*200ns=1600ns T2=T+(m-1)t=200ns+7*50ns=550ns

所以带宽分别为：W1=q/t1=512/(1600x 10⁻⁹)=32X 10⁶(位/秒) W2=q/t2=512/(550X 10⁻⁹)=93.1x 10⁶(位/秒)

7.【解】-15/64=-0.001111=-1.111x2⁻³； E=-3+127=124=01111100 S=1
M=1111 0000 0000 0000 0000 0000；
最后表示为：1 01111100 111100000000000000000000

8.【解】刷新所需带宽=分辨率 x 每个像素点颜色深度x刷新速率= 1024x1024x3x72=226MB/S ； 刷存总带宽 =226MB/Sx100/60=453MB/S

9.【解】(1)流水处理的时空图如下，其中每个流水操作周期为 100ns:



(2)流水线的实际吞吐量：执行 20 条指令共用 5+1*19=24 个流水周期，共 2400ns，所以实际吞吐率为：

$$\frac{20}{2400 \times 10^{-9}} \approx 8.33 \text{ 百万条指令/秒}$$

(3)流水线的加速比为：设流水线操作周期为 T，则 n 指令串行经过 k 个过程的时间为 n*k*T；而 n 条指令经过

可并行的 k 段流水线时所需的时间为 $(k+n-1) \times T$; 故 20 条指令经过 5 个过程段的加速比为: $\frac{20 \times 5 \times \tau}{(5+19) \times \tau} \approx 4.17$

10. 【解】 (1) $275 \times 12288 \times 4 = 12.89\text{MB}$

(2)最高位密度 D1按最小磁道半径 R1 计算($R1=115\text{mm}$): $D1=12288 \text{ 字节}/2\pi R1=17 \text{ 字节}/\text{mm}$; 最低位密度 D2 按最大磁道半径 R2 计算: $R2=R1+(275 \div 5)=115+55=170\text{mm}$; $D2=12288 \text{ 字节}/2\pi R2=11.5 \text{ 字节}/\text{mm}$

(3) $(3000/60) \times 12288 = 600\text{KB/s}$ (4) $\frac{1}{2} \times \frac{60}{3000} \times 1000 = 10\text{ms}$

(5)	16	15 14	6 5	4 3	0
	台号	柱面(磁道)号	盘面(磁头)号	扇区号	

此地址格式表示有 4 台磁盘，每台有 4 个记录面，每个记录面最多可容纳 512 个磁道，每道有 16 个扇区。

计算机组成原理期末试题及答案(三)

一.选择题

答案速查：BDCCB AADAB DCDDB

二.名词解释

1.CPU 周期也叫机器周期。是指 CPU 访问一次主存或输入输出端口所需要的时间。一个 CPU 周期由若干个时钟周期组成。

2.存取时间又称存储器访问时间，是指启动一次存储器操作到完成该操作所需的时间。具体地说，存取时间从存储器收到有效地址开始，经过译码、驱动，直到将被访问的存储单元的内容读出或写入为止。

3.存储设备在单位时间内向主机传送数据的二进制位数或字节数称为数据传输率。数据传输率(Dr)与存储设备的构造及主机接口逻辑有着密切的关系，与记录密度 D 和记录介质的运动速度 V 成正比。

4.微指令格式是指由于编译微操作码时采用不同的方法而形成的不同格式的微指令的方式，微指令格式有水平型微指令和垂直型微指令两种。

5.总线上的部件通过总线进行信息传送时，用一个公共的时钟信号进行同步，这种方式称为同步通信。这个公共的时钟信号可以由总线控制部件发送到每一个部件或设备，也可以每个部件有自己的时钟发生器，但是它们都必须由 CPU 发出的时钟信号进行同步。

6.接口是计算机系统总线与外围设备之间的一个逻辑部件，它的基本功能有两点：一是为信息传输操作选择外围设备；二是在选定的外围设备和主机之间交换信息，保证外围设备用计算机系统特性所要求的形式发送或接收信息。

7. 是指组成一台计算机的各种物理装置,它们是由各种实实在在的器件组成的,是计算机进行工作的物质基础。计算机的硬件由输入设备、输出设备、运算器、存储器和控制器五部分组成。

8.正逻辑是把逻辑电路中电平的高低和逻辑变量值 0、1 联系起来的一种概念。指定逻辑电路中高电平为“1”，低电平为“0”，称为正逻辑。

9.指令的编码格式是指指令中操作码的长度是否可以变化而形成指令的方式。指令操作码通常有两种编码格式：固定格式和可变格式。

10.指令周期是执行一条指令所需要的时间。也就是从取指令开始到执行完这条指令为止的全部时间。一个指令周期由若干个机器周期组成。

11.存储周期又称访内周期，是指连续启动两次独立的存储器操作所需间隔的最小时间。它是衡量主存储器

工作性能的重要指标。

12.微命令是指构成一条指令所对应的各种基本命令，它是构成控制信号序列的最小单位，微命令通常是指那些能直接作用于某部件控制门的命令。

三.简答题

1.是由计算机电路所采用的器件决定的。计算机中采用了具有两个稳态的二值电路，用二值电路只能代表两个数码：0 和 1。比如，采用正逻辑表示，是以低电位表示数码“0”，高电位表示数码“1”；负逻辑表示，则以高电位表示数码“0”，低电位表示数码“1”。在计算机中采用二进制，具有运算规则简单，物理上实现方便，成本低廉，数码“1”和“0”正好与逻辑命题中的两个值“True”、“False”相对应，为计算机中实现逻辑运算和程序中的逻辑判断提供了便利条件等优点。

2.主存储器的性能指标主要是存储容量、存取时间、存储周期可靠性和性能价格比。在一个存储器中可以容纳的存储单元总数通常称为该存储器的存储容量。存取时间又称存储访问时间，是指从启动一次存储器操作到完成该操作所经历的时间。存储周期是指连续两次独立的存储器操作(如连续两次读操作)所需间隔的最小时间。可靠性是指在规定的时间内，存储器无故障工作的时间。通常用平均无故障时间 MTBF 来衡量。性能与价格的比值是衡量存储器经济性能好坏的综合性指标。

3.计算机的主要技术性能指标有下面几项：主频、字长、存储容量、存取周期和运算速度等。

(1)主频：主频即时钟频率，是指计算机的 CPU 在单位时间内发出的脉冲数。

(2)字长：字长是指计算机的运算部件能同时处理的二进制数据的位数，它与计算机的功能和用途有很大的关系。字长决定了计算机的运算精度，字长长，计算机的运算精度就高。字长也影响机器的运算速度，字长越长，计算机的运算速度越快。

(3)存储容量：计算机能存储的信息总字节量称为该计算机系统的存储容量存储容量的单位还有 MB (兆字节)、GB(吉字节)和 TB(太字节)。

(4)存取周期：把信息代码存入存储器，称为“写”；把信息代码从存储器中取出，称为“读”。存储器进行一次“读”或“写”操作所需的时间称为存储器的访问时间(或读写时间)，而连续启动两次独立的“读”或“写”操作(如连续的两次“读”操作)所需的最短时间，称为存取周期(或存储周期)。

(5)运算速度：运算速度是一项综合性的性能指标。衡量计算机运算速度的单位是 MIPS(百万条指令/秒)。因为每种指令的类型不同，执行不同指令所需的时间也不一样。过去以执行定点加法指令作标准来计算运算速度，现在用一种等效速度或平均速度来衡量。等效速度由各种指令平均执行时间以及相对应的指令运行比例计算得出来，即用加权平均法求得。

4. (1)a 为存储器数据寄存器 MDR，b 为指令寄存器 IR，c 为存储器地址寄存器 MAR，d 为程序计数器 PC；

(2)取指令的数据通路：PC→MAR→MM→MDR→IR

(3)数据从主存中取出的数据通路(设数据地址为 X)：X→MAR→MM→MDR→ALU→AC

5.(1) 用来保存当前正在执行的一条指令；(2) 用来确定下一条指令的地址；

(3) 用来保存当前 CPU 所访问的内存单元的地址；

(4) <1>作为 CPU 和内存、外部设备之间信息传送的中转站。 <2>补偿 CPU 和内存、外围设备之间在操作速度上的差别。 <3>在单累加器结构的运算器中，缓冲寄存器还可兼作为操作数寄存器；

(5)当运算器的算术逻辑单元(ALU)执行全部算术和逻辑运算时，为 ALU 提供一个工作区。

(6) 保存由算术指令和逻辑指令运行或测试的结果建立的各种条件码内容。除此之外，还保存中断和系统工作状态等信息，以便使 CPU 和系统能及时了解机器运行状态和程序运行状态。

6.指令周期是执行一条指令所需要的时间。也就是从取指令开始到执行完这条指令为止的全部时间。一个指令周期由若干个机器周期组成。CPU 周期也叫机器周期。是指 CPU 访问一次主存或输入输出端口所需要的时间。一个 CPU 周期由若干个时钟周期组成。时钟周期是 CPU 处理操作的最小时间单位，也叫 T 周期。

7.18,8

8.微命令通常是指那些能直接作用于某部件控制门的命令，它是构成控制信号序列的最小单位。执行部件接受微命令后进行的操作称为微操作。在一个 CPU 周期中，一组实现一定操作功能的微命令的组合，称为微指令。对应于程序指令的概念，微指令的有序集合称为微程序。一般一条机器指令的功能由对应的一段微程序来实现。

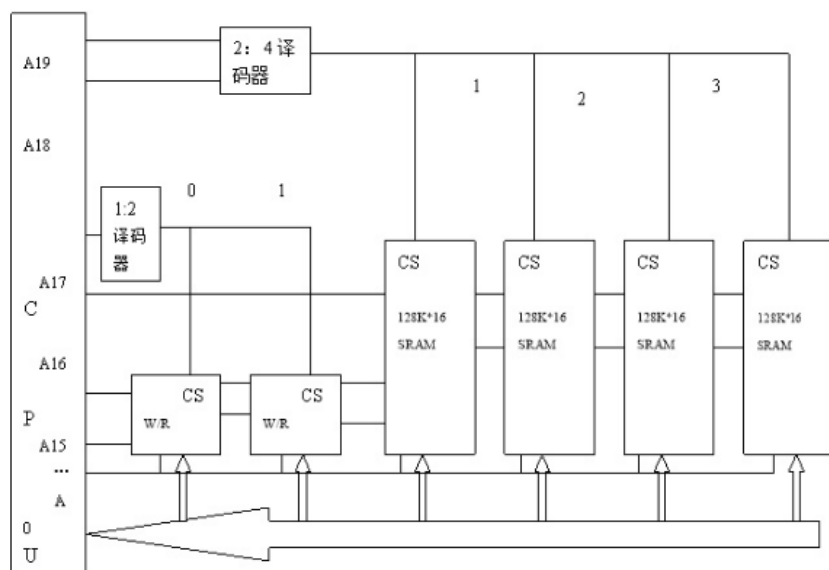
9.根据总线所传输的信息内容的不同，总线可分为地址总线、数据总线和控制总线。根据总线在计算机系统中所处的位置不同，总线又有片级总线、内部总线和外部总线之分。根据二进制数码的传送方式，总线有并行总线和串行总线两种。根据数据的传送方向，总线有单向总线和双向总线两种。数据总线的特点是：双向传输；地址总线的特点是：单向传输。

四.计算机设计题

1. 平均存取时间=20ns*0.9+(20 ns +60 ns)*0.6*0.1+(20 ns +60 ns+1200us)*0.4*0.1=48 us

2. (1)16 位；(2)地址寄存器需要 20 位，其中只读存储器需要地址线 16 位，其中有一位为 1:2 译码器；SRAM 组成的 512K*16 的存储器需要 19 位地址线，且与只读存储器共用 A0~A15 这 16 根线，其中 A18, A19 两根组成 2:4 译码器。

(3)共需要 EPROM8 片；(4)各个芯片与 CPU 连线如图：

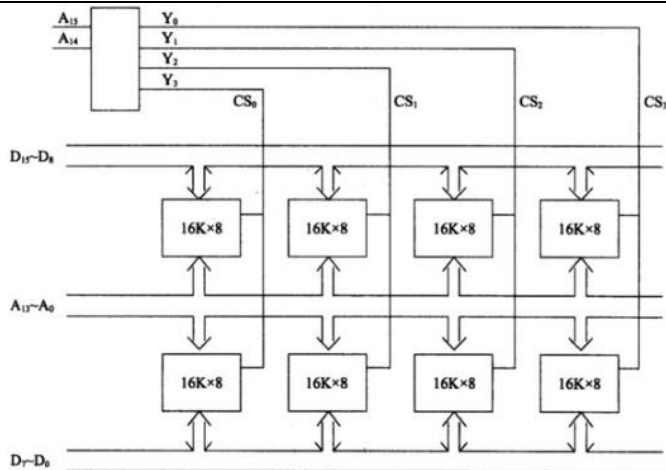


第 2 题(4)

3. ①定点原码整数表示时最大正数= $(2^{15}-1)_{10} = (32767)_{10}$ 最小负数= $-(2^{15}-1)_{10} = (-32767)_{10}$ ②定点原码小数表示时最大正数= $(1-2^{-15})_{10}$ ；最小负数= $-(1-2^{-15})_{10}$

4.如图所示。

5. $(1101.0111)_2 = (13.4375)_{10} = (E.7)_{16}$



第4题图

6.① $X = 0.1011$ $Y = 0.1100$ $\therefore [X]_{\text{补}} = 00\ 1011$, $[Y]_{\text{补}} = 00\ 1100$ 。 $00\ 1011 + 00\ 1100 = 01\ 0111$ 有溢出 ② $X = -0.1011$ $Y = 0.1001$ $\therefore [X]_{\text{补}} = 11\ 0101$ $[Y]_{\text{补}} = 00\ 1001$ 。 $11\ 0101 + 00\ 1001 = 11\ 1110$ 不溢出。

7.【解】 $(-0.276)_{10} = (-0.0100011)_2$ $(47)_{10} = (0101111)_2$

原码	1 0100011	0101111
反码	1 1011100	0101111
补码	1 1011101	0101111

计算机组成原理期末试题及答案(四)

一、选择题 (每小题选出一个最合适的答案, 每小题 2 分, 共 20 分)

答案速查: DBAAB ABBDD

二、名词解释 (每小题 4 分, 共 20 分)

1.全相联映像: 就是让主存中的任何一个块均可以映像装入到 Cache 中任何一个块的位置上。

2.指令系统: 是指一台计算机的所有指令的集合。

3.指令周期: 是指从取指令、分析取数到执行完该指令所需的全部时间。

CPU 周期: 也叫机器周期, 通常把一个指令周期划分为若干个机器周期, 每个机器周期完成一个基本操作。

4.向量中断: 是指那些中断服务程序的入口地址是由中断事件自己提供的中断。

5.微指令: 是指控制存储器中的一个单元的内容, 即控制字, 是若干个微命令的集合。

微指令 在微程序控制的计算机中, 将由同时发出的控制信号所执行的一组微操作称为微指令

三、改错题 (在下列各小题的表述中均有错误, 请改正。每小题 3 分, 共 12 分)

1、在中央处理器中, 运算器可以向控制器发出命令进行运算操作。

改为: 在中央处理器中, 控制器可以向运算器发出命令进行运算操作。

2、在单处理机总线中, 相对 CPU 而言, 地址线 and 数据线一般都为双向信号线

改为：在单处理机总线中，相对 CPU 而言，地址线为单向信号和数据线一般都为双向信号线

3、多重中断方式，是指 CPU 同时处理多个中断请求

改为：多重中断是指具有中断嵌套的功能，CPU 在响应较低级别的中断请求时，如果有更高级别的中断请求，CPU 转去响应更高级别中断请求。

4、在“半互锁”异步通信方式中，“请求”信号的撤消取决于“回答”信号的来到，而“请求”信号的撤消又导致“回答”信号的撤消

改为：在“半互锁”异步通信方式中，“请求”信号的撤消取决于“回答”信号的来到，而“回答”信号的撤消由从设备自己决定。

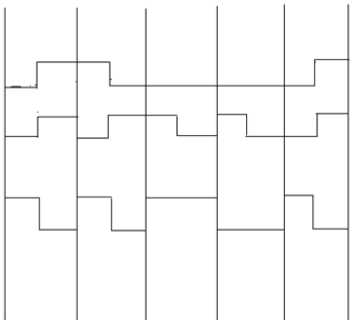
四、简答题（每小题 5 分，共 15 分）

1.000 XXX YYY ZZZ

101 XXX YYY ZZZ
110 000 YYY ZZZ

110 111 YYY ZZZ

2. NRZ-1
 PE
 FM



3.相同点：都是能在不需要 CPU 干预下实现外设和内存间的数据交换（2 分）

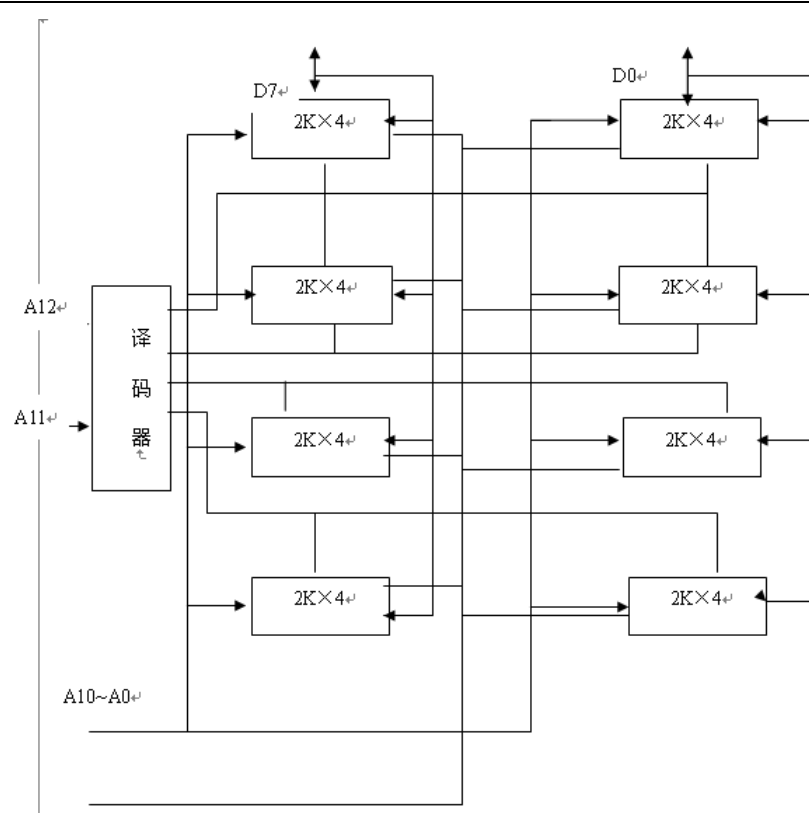
不同点：1) DMA 控制器是通过专门设计的硬件控制逻辑来实现对数据传递的控制，而通道具有自己的指令和程序，是一个有特殊功能的处理器 2) DMA 仅能控制一台或几台同类设备，而通道能控制多台同类或不同类设备

五、计算题（10 分）

解题要领：首先要转化为 ， 然后进行列算式计算。没有转化， 但会列算式， 且最后结果正确给 7 分。

六、设计题（第一小题 12 分， 第二小题 11 分， 共 23 分）

1.(每小题 4 分， 共 12 分)(2)取指令的数据通路：PC→MAR→MM→MDR→IR
(3)数据从主存取出的数据通路（设数据地址为 X）X→MAR→MM→MDR→ALU→AC
数据存入主存的数据通路（设数据地址为 Y）Y →MAR,AC→MDR→MM
2.(1)共需 8 片（5 分）(2)如下逻辑图（6 分）



附录二 知识点总结

【注】这是软件学院本校期末总结的考点，我们在第 6 版进行了增加，大家可以着重看看，尤其是针对之前概念或解题思路不清晰的，用这个进行规范

第一章 计算机系统概论

☞1-1 计算机功能：数据处理/数据存储/数据传送/操作控制

☞1-2 名词解释：外围设备

☞1-3 冯·诺依曼计算机特征：程序存储二进制运算

- 1.计算机由运算器、存储器、控制器、输入设备、输出设备五大部分组成
- 2.指令和数据以同等地位存放于存储器内，可按地址寻访
- 3.指令和数据均用二进制表示
- 4.指令由操作码和地址组成，操作码用来表示操作的性质，地址码用来表示操作数在存储器中的位置
- 5.指令在存储器内按顺序存放。
- 6.机器以运算器为中心，输入输出设备与存储器间的数据传送通过运算器完成

原因？？

☞1-4 计算机硬件主要技术指标：机器字长，存储容量，运算速度

☞1-5 机器字长：指 CPU 一次能处理数据的位数，通常与 CPU 寄存器位数有关。

☞1.1 什么是计算机系统、计算机硬件和计算机软件？硬件和软件哪个更重要

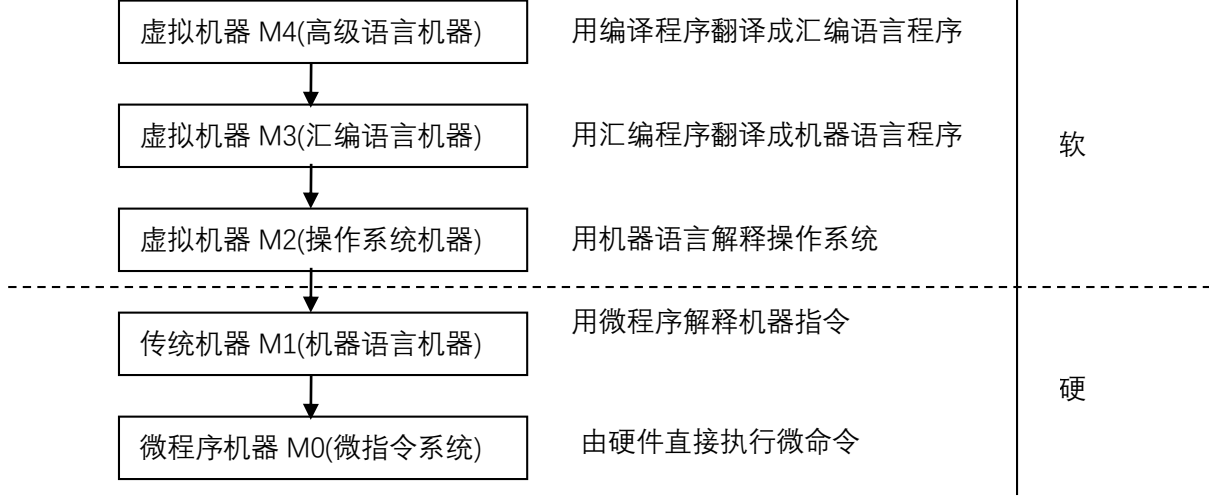
计算机系统——计算机硬件、软件和数据通信设备的物理或逻辑的综合体。

计算机硬件——计算机的物理实体。

计算机软件——计算机运行所需的程序及相关资料。

硬件和软件在计算机系统中相互依存，缺一不可，因此同样重要。

☞1.2 如何理解计算机系统的层次结构？



☞1.3 说明高级语言、汇编语言和机器语言的差别及联系

机器语言：用 0、1 代码表示的语言，可直接在机器上执行

汇编语言：用某些人类便于理解的符号代替相应的机器代码，必须先翻译成机器语言程序才能被机器接受

高级语言：接近人类语言，要先翻译成汇编语言或其他中间语言程序，再翻译成机器语言程序，也可直接

联系：高级语言、汇编语言都被翻译成机器语言才能被机器执行。

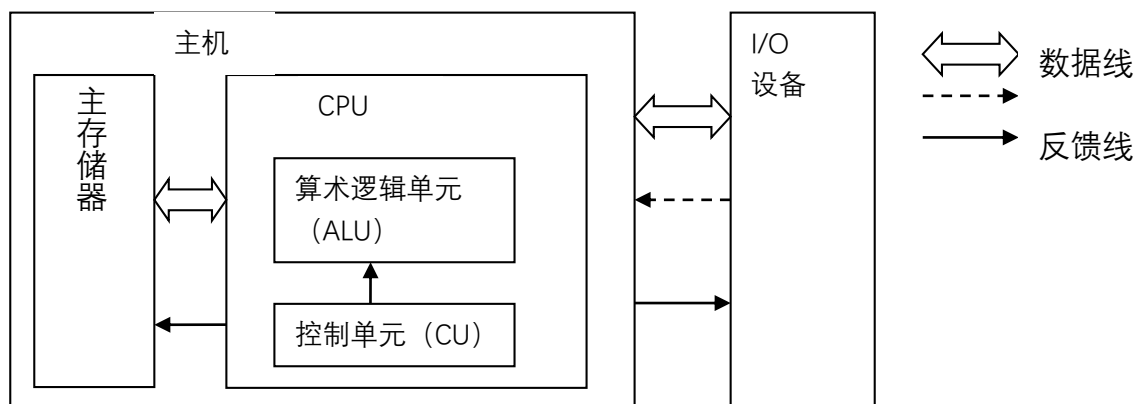
1.4 如何理解计算机组成和计算机体系结构？

答：计算机体系结构是指程序员所见到的计算机系统的属性，即概念性的结构与功能特性。通常是指用机器语言编程的程序员（也包括汇编语言程序设计者和汇编程序设计者）所看到的传统机器的属性，包括指令集、数据类型、存储器寻址技术、I/O 机理等，大都属于抽象的属性。

计算机组成是指如何实现计算机体系结构所体现的属性，它包含了许多对程序员来说是透明的（即程序员不知道的）硬件细节。例如一台机器是否具备乘法指令是一个结构问题，而实现乘法指令采用什么方式是一个组成问题。

1.5 冯·诺依曼计算机特点是什么？见 1-3

1.6 画出计算机硬件组成框图，说明各部件的作用及计算机硬件的主要技术指标



主存储器：用来存放程序和数据，可直接和 CPU 交换信息

ALU 完成算数逻辑运算

CU 用来解释存储器中的指令，并发出各种操作命令，执行程序

I/O 设备：完成相应的输入输出操作

主要技术指标：机器字长，存储容量，运算速度

1.7 解释概念：主机、CPU、主存、存储单元、存储元件、存储基元、存储元、存储字、存储字长、存储容量、机器字长、指令字长。

主机——是计算机硬件的主体部分，由 CPU+MM（主存或内存）组成；

CPU——中央处理器（机），是计算机硬件的核心部件，由运算器+控制器组成；（早期的运、控不在同一芯片上）

主存——计算机中存放正在运行的程序 and 数据的存储器，为计算机的主要工作存储器，可随机存取；（由存储体、各种逻辑部件及控制电路组成）

存储单元——可存放一个机器字并具有特定存储地址的存储单位；

存储元件——存储一位二进制信息的物理元件，是存储器中最小的存储单位，又叫存储基元或存储元，不能单独存取；

存储字——一个存储单元所存二进制代码的逻辑单位；

存储字长——一个存储单元所存二进制代码的位数；

存储容量——存储器中可存二进制代码的总量；（通常主、辅存容量分开描述）

机器字长——CPU 能同时处理的数据位数；

指令字长——一条指令的二进制代码位数

1.11 指令和数据都存于存储器中,计算机如何区分它们？

计算机硬件主要通过不同的时间段来区分指令和数据，即：取指周期（或取指微程序）取出的既为指令，执行周期（或相应微程序）取出的既为数据。另外也可通过地址来源区分，从 PC 指出的存储单元取出的是指令，

由指令地址码部分提供操作数地址。

①1.12 什么是指令？什么是程序？

指令是计算机执行某种操作的命令，由一串二进制代码组成。

程序是为实现特定的目标或解决特定的问题而用计算机语言编写的命令序列(指令)的集合。

第三章 系统总线

②®3-1 总线：连接多个部件的信息传输线，是各部件共享的传输介质(多个模块共享的信息通路)->名词解释

③3-2 总线组成：传输线，总线接口逻辑，总线仲裁部件

④3-3 系统总线：CPU，主存，I/O 设备(通过 I/O 接口)各大部件之间的信息传输线

⑤3-4 传输线：数据线，地址线，控制线

数据线：传送数据的通路，双向，并行传送能力

地址线：传送数据地址的通路，单向，由 CPU 发出

控制线：传送控制信号的通路，单向，命令。响应，定时

⑥3-5 通信总线：按传输方式分为：串行通信/并行通信

串行通信：数据在单条 1 位宽的传输线上，一位一位地按顺序分时传送

并行通信：数据在多条并行 1 位宽的传输线上，同时由源传送到目的地

⑦3-6 总线的特性：机械特性，电气特性，功能特性，时间特性

机械特性：总线在机械连接方式上的一些性能，如插头与插座使用的标准，几何尺寸、形状、引脚个数、排列顺序，接头处的可靠接触等

电气特性(逻辑联系)：总线每一根传输线上信号的传递方向和有效的电平范围

功能特性：总线中每根传输线的功能

时间特性：总线中的任一根线在什么时间内有效

⑧3-7 总线性能指标：

1.总线宽度：通常指总线的根数，用 bit 表示

2.总线带宽：总线的数据传输速率，即单位时间内总线上传输数据的位数，通常用每秒传输的字节数衡量，单位 MBps

3.时钟同步/异步：总线上数据与时钟同步工作的总线称为同步总线，与时钟不同步工作的总线为异步总线

4.总线复用：一条信号线上分时传送两种信号

5.信号线数：地址总线、数据总线、控制总线三种总线数的总和

6.即插即用

⑨3-8 总线结构：单总线结构/双总线结构

单总线结构：CPU、主存、I/O 设备(通过 I/O 接口)都挂在一组总线上

优：总线结构改变灵活/CPU 可访问所有设备/两设备间也可通信

缺：总线宽度(提高频率，增加宽度—有限)/冲突

多总线结构：将速度较低的 I/O 设备从单总线上分离出来，形成主存总线与 I/O 总线分开的结构。

®P47 表 3.1

⑩3-9 集中控制优先权仲裁方式：链式查询/计数器定时查询/独立请求方式 (P58)

1.菊花链：优点：简单，只有三根控制线

缺点：不公平，速度慢，故障敏感性强及故障传播

2.独立请求：请求—排队—授权—获得设备置 BB 忙

优点：速度快，优先权灵活，公平，解决故障敏感性

缺点：复杂， $1+2n$ 根控制线

3.计数器定时查询：复杂性剧中， $2+\log_2 n$ 根控制线；优先权不固定，取决于计数器初值，有一定的公平性。

3-10 总线通信：同步通信/异步通信/半同步通信(准同步)/分离式通信

同步通信：只有发送方发送了，接收方才能接收；一旦发送方停止发送，接收方停止接收

① 定时方法：时钟

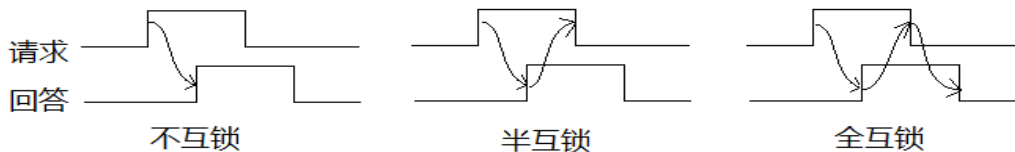
② 优缺点：优：规定明确统一，模块间配合简单一致，可最大限度容忍时钟误差

缺：主从模块时间配合属于强制性“同步”，必须在限定的时间内完成规定的要求

可靠性低，不适合远距离传送(始终漂移)，对存储器要求高

异步通信：克服同步通信缺点，允许各模块速度不一致性，给设计者充分的灵活性和选择余地。采用应答方式，当主模块发出请求信号时，一直等待从模块反馈回来“响应”信号后才开始通信。

互锁：可解决可靠性问题。代价：复杂性。



异步优点：可靠性高，适合远距离传送，对存储器要求不高

缺点：复杂程度高，效率低

异步串行通信数据传送速率用波特率来衡量。单位时间内传送二进制数据的位数 bps

波特率：信号变化的频率

数据传送速率：单位时间传输的代码位数，位/秒，或 B/s

3.1 什么是总线？总线传输有何特点？为了减轻总线的负载，总线上的部件都应具备什么特点？

总线是连接各个部件的信息传输线，是各个部件共享的传输介质。

总线上信息传输的特点：某一时刻只允许有一个部件向总线发送信息，但多个部件可以同时从总线上接收相同的信息。

以 CPU 片内总线为例，在每个需要将信息送至总线的寄存器输出端接三态门，由三态门控制端控制什么时刻由哪个寄存器输出。当控制端无效时，寄存器和总线之间呈高阻状态。

3.2 总线如何分类？什么是系统总线？系统总线又分为几类，它们各有何作用，是单向的还是双向的。它们与机器字长、存储字长、存储单元有何关系？

总线分类：按数据传送方式：并行/串行传输总线；使用范围：计算机总线、测控总线、网络通信总线；连接部件：系统总线、片内总线、通信总线

系统总线：CPU，主存，I/O 设备(通过 I/O 接口)各大部件之间的信息传输线。分为数据总线，地址总线，控制总线。

数据总线：用来传输各功能部件之间的数据信息，双向，其位数与机器字长、存储字长有关，一般为 8 位、16 位、32 位

地址总线：主要用来指出数据总线上的源数据或目的数据在主存单元的地址或 I/O 设备的地址，单向，位数与存储单元的个数有关，eg:地址线 20 根，存储单元个数 2^{20}

控制总线：用来发出各种控制信号的传输线，单向

3.3 常用的总线结构有几种？不同的总线结构对计算机的性能有什么影响？举例说明

单/多总线结构。单：简单，便于扩充，但所有传送都通过这组总线共享，易形成计算机系统瓶颈。多：工作效率高

3.5 解释概念：总线宽度、总线带宽、总线复用、总线的主设备（或主模块）、总线的从设备（或从模块）、总线的传输周期、总线的通信控制。

总线宽度——指数据总线的位（根）数，用 bit（位）作单位。

总线带宽——指总线在单位时间内可以传输的数据总量，相当于总线的数据传输率，等于总线工作频率与总线宽度（字节数）的乘积。

总线复用——指两种不同性质且不同时出现的信号分时使用同一组总线，称为总线的“多路分时复用”。

总线的主设备（主模块）——指一次总线传输期间，拥有总线控制权的设备（模块）；

总线的从设备（从模块）——指一次总线传输期间，配合主设备完成传输的设备（模块），它只能被动接受主设备发来的命令；

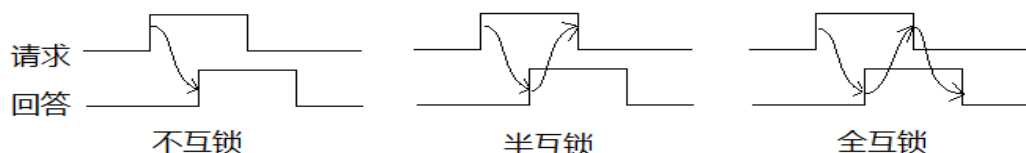
总线的传输周期——总线完成一次完整而可靠的传输所需时间；

总线的通信控制——指总线传送过程中双方的时间配合方式。

☞3.6 试比较同步通信和异步通信。

同步通信和异步通信的主要区别是前者有公共时钟，总线上的所有设备按统一的时序统一的传输周期进行信息传输，通信双方按约定好的时序联络；后者没有公共时钟，没有固定的传输周期，采用应答方式通信，具体的联络方式有不互锁、半互锁和全互锁三种。不互锁方式通信双方没有相互制约关系；半互锁方式通信双方有简单的制约关系；全互锁方式通信双方有完全的制约关系。其中全互锁通信可靠性最高。

☞3.7 画图说明异步通信中请求与回答有哪几种互锁关系？



☞3.8 为什么说半同步通信同时保留了同步通信和异步通信的特点？

半同步通信既能像同步通信那样由统一时钟控制，又能像异步通信那样允许传输时间不一致，因此工作效率介于两者之间

☞3.10 什么是总线标准？为什么要设置总线标准？目前流行的总线标准有哪些？什么是即插即用？哪些总线有这一特点？

总线标准——可理解为系统与模块、模块与模块之间的互连的标准界面。

总线标准的设置主要解决不同厂家各类模块化产品的兼容问题；

目前流行的总线标准有：ISA、EISA、PCI 等；

即插即用——指任何扩展卡插入系统便可工作。EISA、PCI 等具有此功能。

☞3.13 什么是总线的数据传送速率，它与哪些因素有关？

单位时间内总线上传输数据的位数，通常用每秒传输的字节数衡量，单位 MBps

☞3.14 设总线的时钟频率为 8MHz，一个总线周期等于一个时钟周期。如果一个总线周期中并行传送 16 位数据，试问总线的带宽是多少？

$$\text{总线宽度} = 16 \text{ 位} / 8 = 2\text{B}$$

$$\text{总线带宽} = 8\text{MHz} \times 2\text{B} = 16\text{MB/s}$$

☞3.15 在一个 32 位的总线系统中，总线的时钟频率为 66MHz，假设总线最短传输周期为 4 个时钟周期，试计算总线的最大数据传输率。若想提高数据传输率，可采取什么措施？

$$\text{总线的最大数据传输率为: } 4\text{B} / (4 \times T) = 1\text{B} \times f = 66 \times 10^6 \text{Hz} \approx 63\text{MBps}$$

若想提高数据传输率，可采用以下两种措施：外部数据总线宽度改为 64 位；时钟频率提高到 132 MHz。

☞3.16 在异步串行传送系统中，字符格式为：1 个起始位、8 个数据位、1 个校验位、2 个终止位。若要求每秒传送 120 个字符，试求传送的波特率和比特率。

根据题目给出的字符格式，一个字符包含 $1+8+1+2=12$ 位

$$\text{故波特率为 } 120 \times 12 = 1440\text{bps}$$

$$\text{比特率为 } 1440 \times (8/12) = 960\text{bps}$$

第四章 存储器

☞4-1 存储器：计算机系统记忆设备，用来存放程序和数据。

重要性：影响计算机系统的类型，技术组织性能，价格

CPU, Register

内存 { cache
主存储器=主存

外存 eg. 磁盘，通过I/O与CPU进行信息交换

内存=主存

☞4-2 存储器位置：内存=cache+主存

☞4-3 容量：位/字节/字

字长：字所包含的2进制位

存储容量：所有字的总和

☞4-4 传送单位：主存：Byte/word，与DB宽度有关，指每次读入/写入主存的数据位数

外存：块，每次传送一个块，块的大小是Byte的整数倍

☞4-5 存储方法：顺序/直接/随机/关联存取（顺序/直接/随机—按地址 关联—数据(快)速度快，容量小，成本高）

顺序存取：访问时间和数据位置有关 eg. 磁带

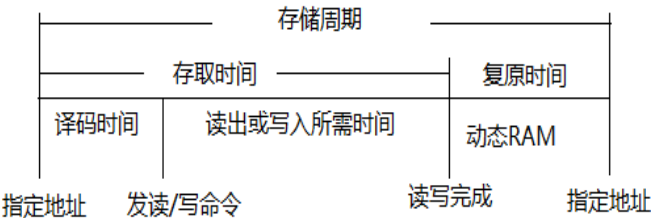
直接存取：(半顺序存取) eg. 硬盘，寻道位置和时间有关，查找不同扇区时间相等

随机存取：eg. 内存 访问时间和位置有关，共享地址译码

存储位置	性能	物理性质	容量	传送单位	物理性质	存取方法
CPU(寄存器)	存取时间	半导体	字长	字=存储单元	易失，不可擦	顺序直接
内存(主存)	存储周期	磁表面	字数	块	可擦，非易失	随机关联
外存(辅存)	传送速度	光介质	存储容量			

☞4-6 性能：①存取时间：写入时间和读出时间；②存储周期；③传送速度

② 存储周期：连续读写所需时间，对存储器连续访问所需最短的时间间隔



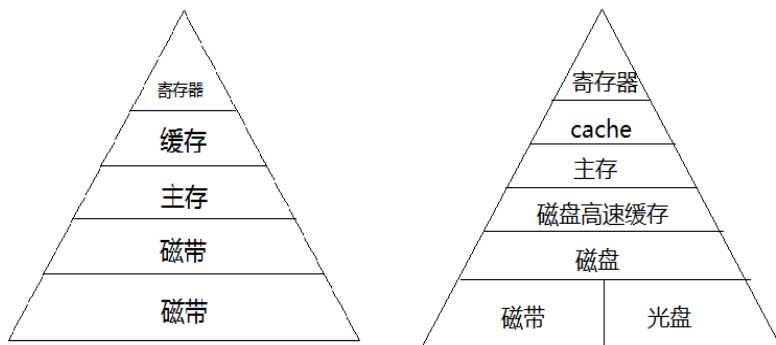
③ 传送速度：传输率，存储器带宽。

对于主存，单位时间内数据传入/传出主存的位数 位/秒 bps 字节/秒 Bps

例：DB=8bit，存储周期 250ns，求主存带宽？

主存单元长度和 DB 相等是最佳匹配， $8 \times 1s / 250ns = 4 \times 10^6 \text{Byte/s}$

☞4-7 存储器层次结构



追求：大容量，高速度，低成本

☞4-8 存储体组成：

最小存储单元：存储元，一个二进制位

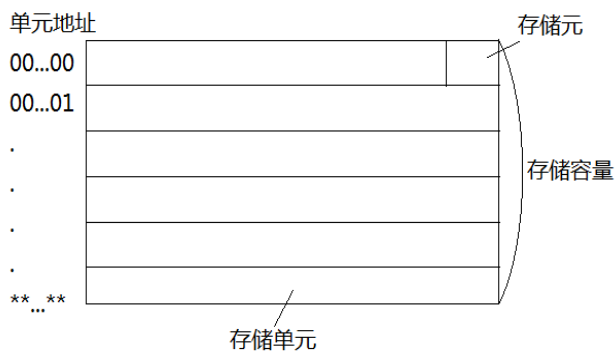
存储单元：若干个存储元构成一个存储单元，有唯一编号：单元地址

单元地址：存储器地址，存储单元地址，简称地址

存储容量：组成存储器所有存储元的个数。

◎存储器：计算机中重要组成部件，作用是存储数据/程序，自从引入了存储器，计算机就开始了自动化。->

考：名词解释



☞4-9 存储体编制方式：①按字。②按字节

☞4-10 主存技术指标：存储容量，存储速度

① 存储容量：主存能存放二进制代码的总位数：存储容量=存储单元格数*存储字长
也可用字节总数表示：存储容量=存储单元格数*存储字长/8

② 存储速度：由存取时间和存取周期来表示

③ 存储器带宽：与存取周期密切相关，表示单位时间内存储器存取的信息量。

提高带宽：缩短存取周期/增加存储器字长/增加存储体

☞4-11 半导体主存



4-12 书 P76 图 4.9

4-13SRAM

6 管 SRAM 存储元双稳态：->了解

稳态 1: $A \uparrow$ T1 截止 $\rightarrow A \uparrow \rightarrow T2$ 通 $\rightarrow B \downarrow$

稳态 2: T2 截止 $\rightarrow A \downarrow \rightarrow T1$ 通 $\rightarrow B \uparrow$

信息存储原理：电阻分压 优点：速度快，稳定性好 缺点：成本高，复杂

4-14SRAM 存储器组成：存储元IC芯片 排成行列矩阵结构

芯片包含存储元的个数叫容量；芯片一次可访问的存储元个数叫字长。

©用 Intel2114 设计->考：设计

容量：地址线 $A0 \sim Ax$ eg.1K, $A0 \sim A9$

字长：I/O 线 eg.4bit, I/O 线 4 根 $I/O0 \sim I/O3$

SRAM 集成度低，所以有了 DRAM

4-15DRAM

©Intel4116 单管 DRAM 算引脚数：地址减半，数据线分开，两个片选->考：算引脚数

4-16DRAM 刷新：靠电容存储信息，外界给 C2 补充电荷刷新，靠读操作完成。

例题：DRAM16K*1bit,128*128 矩阵，刷新周期 2ms，读写周期 0.5 μ s

① 集中式刷新

在规定的一个刷新周期内，对全部存储单元集中一段时间逐行刷新，此刻必须停止读/写操作。一旦开始，把所有行都刷新完毕。

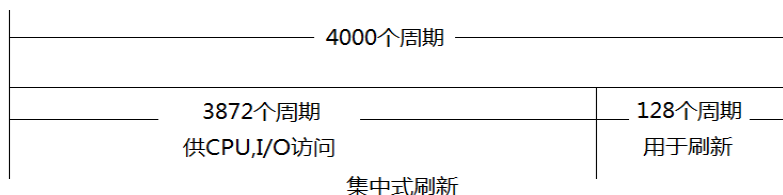
优点：简单

缺点：刷新时间长，死时间 CPU 不能访问，效率低

应用：高速存储器

例题：每次一行，一个读操作，128 个读周期

2ms/0.5 μ s=4000 个周期，128 个用于刷新，即 64ms



② 分散刷新

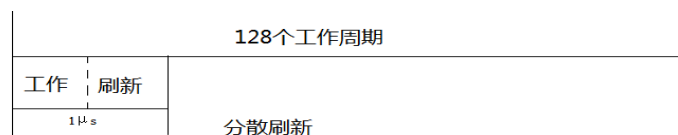
对每行存储单元刷新分散到每个存取周期内完成。

每个工作周期带一个刷新，把死时间分散，2ms 内刷新若干次

缺点：影响效率，慢；优点：能有效分散死时间

应用：低速存储器

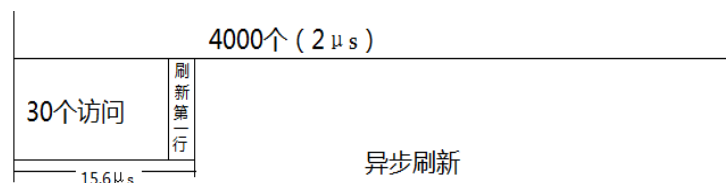
例：若读取周期为 $0.5\mu\text{s}$ ，则存取周期为 $1\mu\text{s}$ ，每 $128\mu\text{s}$ 就可将存储芯片全部刷新一遍。



③ 异步刷新

前两种方式结合，既可缩短死时间，又充分利用最大刷新间隔 2ms。

例：2ms 内对 128 行各刷新一遍，即每隔 $15.6\mu\text{s}$ ($2000\mu\text{s}/128=15.6\mu\text{s}$) 刷新一行，每行刷新时间仍为 $0.5\mu\text{s}$ ，这样刷新一行只停止一个存取周期，但对每行来说，刷新间隔时间仍为 2ms，而死时间缩短为 $0.5\mu\text{s}$ 。 $15.6/0.5=31$ 个周期。

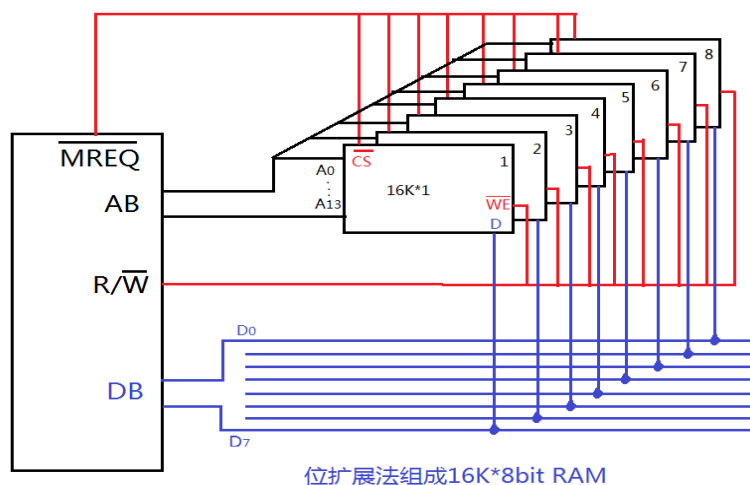
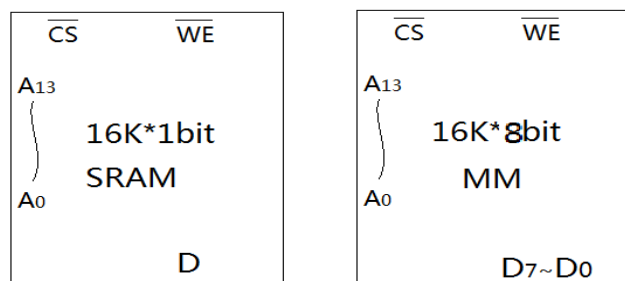


☞ 4-17ROM 原始定义：一旦注入原始信息即不能改变，但随着用户需要，出现了 PROM/ EPROM/ EEPROM
PROM: 可以实现一次性编程的只读存储器，不能再修改。

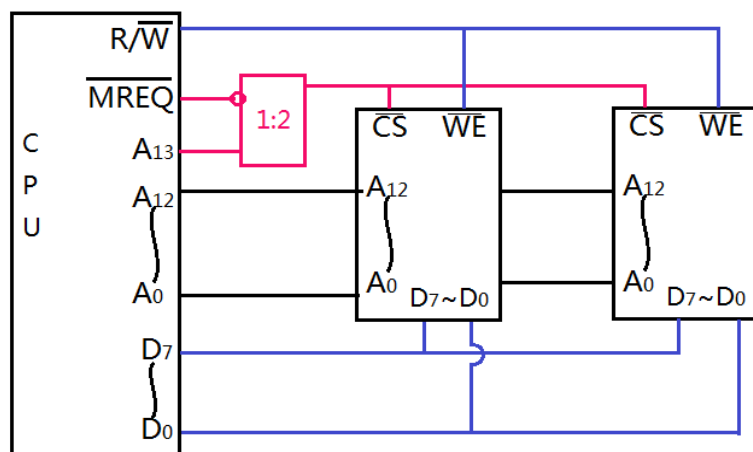
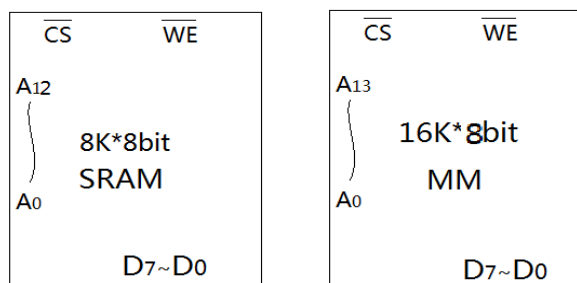
EPROM: 可擦出可编程只读存储器。可以由用户对其所存信息作任意次的改变。

☞ © 4-18 存储容量的扩展->考

© 位扩展：增加存储字长

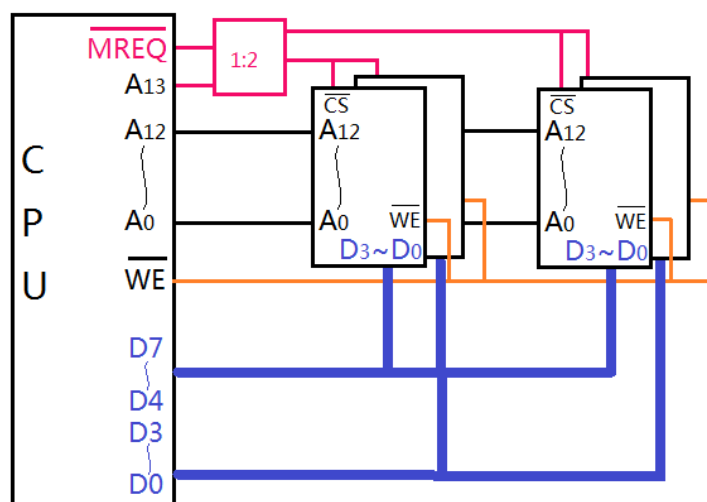
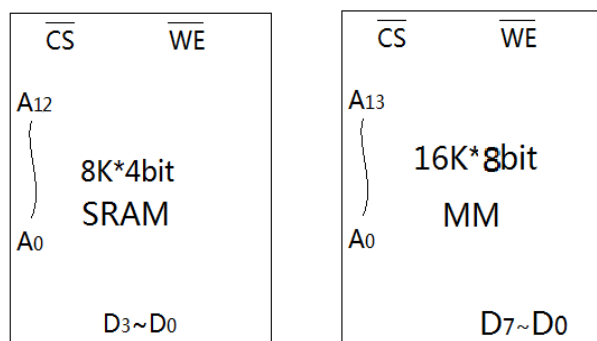


© 字扩展：增加存储器字的数量。



字扩展

© 字位扩展：既增加存储字的数量，又增加存储字长



字位扩展

4-19 存储器的校验

差错：发送方发送信息与接收方接收的不一致，读入与输出不一致。

差错产生：噪声/热/冲击

4-20 纠错码:海明码，正反码

检错码：循环冗余码(CRC)，奇偶校验码，定比码

4-21 奇偶校验码优点:简单易实现；缺点：检错能力差，只能发现奇数位错误，检错率 50%

4-22 海明码

信息位 m bit 校验位 r bit 码字 n=(m+r) bit $2^r \geq n+1=m+r+1 \rightarrow$ 记忆

例：m=4，求 r

R=3， m4 m3 m2 c4 m1 c2 c1

M=7 r=4 m7 m6 m5 c8 m4 m3 m2 c4 m1 c2 c1

例：m=4, r=3, n=4+3=7

7	6	5	4	3	2	1
M4	M3	M2	C4	M1	C2	C1

指误表 1

S2	S1	S0	误码位
0	0	0	无错
0	0	1	C1
0	1	0	C2
0	1	1	M1
1	0	0	C4
1	0	1	M2
1	1	0	M3
1	1	1	M4

S2=C4+M1+M3+M4 S1=C2+M1+M3+M4 S0=C1+M1+M2+M4

m=7, r=4, n=7+4=11

指误表 2

S3	S2	S1	S0	误码位
0	0	0	0	无错
0	0	0	1	C1
0	0	1	0	C2
0	0	1	1	M1
0	1	0	0	C4
0	1	0	1	M2
0	1	1	0	M3
0	1	1	1	M4
1	0	0	0	C8
1	0	0	1	M5
1	0	1	0	M6
1	0	1	1	M7

S3=C8+M5+M6+M7 S2=C4+M2+M3+M4 S1=C2+M1+M3+M4+M6+M7

S0=C1+M1+M2+M4+M5+M7

根据码字确定指误表，根据指误表监督关系式

例：m7m6m5m4m3m2m1=1001000，检测位值，编码序列=？

由指误表 2 后的关系式，推出 C 与 M 的关系， $C8=M5+M6+M7=1$ 以此类推，
m7m6m5c8m4m3m2c4m1c2c1=10011001000

例：m4m3m2c4m1c2c1=0010101，检查是否有错。

$S2=C4+M1+M3+M4=0$ $S1=C2+M1+M3+M4=0$ $S0=C1+M1+M2+M4=1$

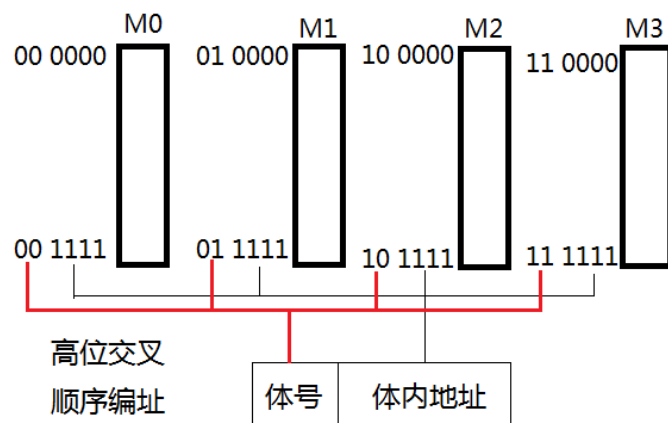
由指误表 1,001---C1 错误，正确序列：0010100

书 P102，例 4.4

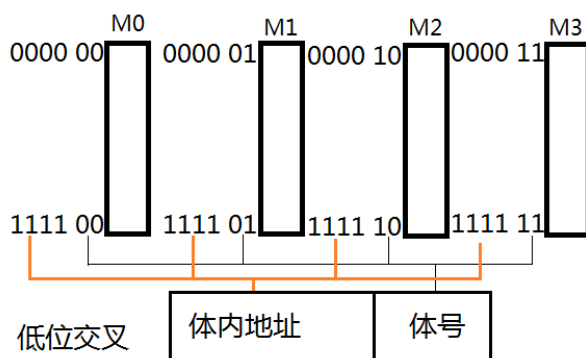
4-23 提高访存速度的措施：单体多字系统，多体并行系统

单体多字：增加存储器宽度 P103，图 4.41

多体并行(多体交叉)：高位交叉：4 体，8 体 低位交叉



低位交叉：



并行性好，个体轮流编址

特点：在不改变存储周期的前提下，增加存储器宽度 书 P105，图 4.44

©书 P106 例 4.6->考 设有 4 个模块组成的四体存储器结构，每个体的存储字长为 32 位，存取周期 200ns，假设数据总线宽度为 32 位，总线传输周期为 50ns，试求顺序存储和交叉存储的存储器带宽。

解：顺序存储(高位交叉编址)和交叉存储(低位交叉编址)连续读出 4 个字的信息量是 $32 \times 4 = 128$ 位。

顺序存储存储器连续读出 4 个字的时间： $200\text{ns} \times 4 = 800\text{ns} = 8 \times 10^{-7}\text{s}$

交叉存储存储器连续读出 4 个字的时间： $200\text{ns} + 50\text{ns} \times (4-1) = 350\text{ns} = 3.5 \times 10^{-7}\text{s}$

顺序存储器带宽： $128 / (8 \times 10^{-7}) = 16 \times 10^7 \text{bps}$

交叉存储器带宽： $128 / (3.5 \times 10^{-7}) = 37 \times 10^7 \text{bps}$

4-24 高速缓冲存储器——所有掌握(书 P109)

4-26 为避免 CPU 与 I/O 设备争抢访存, 可在 CPU 与主存之间加一级缓存

由高速缓存 cache 解决主存与 CPU 速度不匹配问题

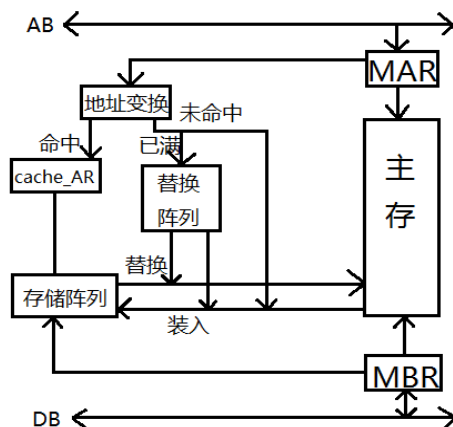
Cache 出现使 CPU 可以不直接访问主存, 而与高速 cache 交换信息

Cache 的使用目的: 优化主存速度; 依据: 程序执行的局部性

$$CPU \xrightarrow[\text{字/字节}]{\text{字/字节} \times \text{cache 块}} \text{主存}$$

4-27 信息交换

4-28 cache 中存放信息 -> 考填空题 结构: 透明性



地址变换: 主存地址→cache 地址, 对 CPU 要访问的内容进行快速访问, 用一个存储器实现, 速度快

4-28 工作原理 -> 考问答

① 命中: 命中率 75% 以上, SRAM 按地址访问, 主存地址转换为 cache 地址

② 不命中 miss: a, MAR→主存→MBR

b, MAR→主存→存储阵列→MBR

不同计算机不同。

4-29 地址转换方法: 全相联映像/直接相联映像/主相联映像 -> 考, 一般考组 王老师喜欢考全/直接

例: block=4B, cache=64B, MM=1024B

析: cache:16block, MM:256block

① 全相联: 结构: cache 和 MM 分成大小相同的块

映像规则: MM 中一块可以放到 cache 中任意一块, 只要空闲。块内地址相同。Cache 中每块长出一个单元记录 MM 号。

地址阵列: 相联存储器, 单元地址=cache 块号, 单元内容=MM 块号

同时找, 速度快, 复杂度高

优点: 只要 cache 有空闲块, 主存就能放进来, cache 利用率高

缺点: 比较逻辑复杂

② 直接相连: 结构: MM 按 cache 大小分区

映像规则: MM 中第 n 块只能放到 cache 中第 n 块

地址阵列: 相联存储器, 单元地址=cache 块号, 单元内容=MM 区号

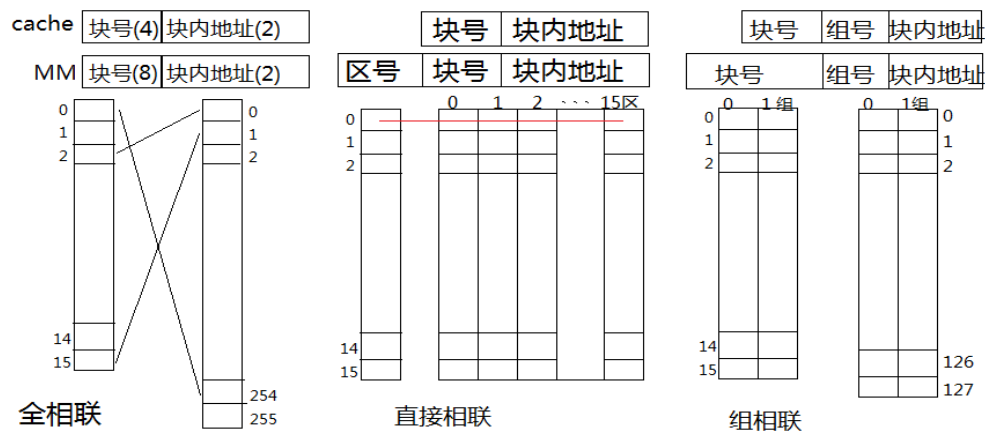
解决地址转换问题

优点: 简单; 缺点: 浪费空间, 利用率低

③ 组相连 路: 2 路组相连, 每组有 2 块; 4 路组相连, 每组有 4 块

图为 8 路组相连

组间直接相联, 组内全相联



©例题 4.8~4.11

☞4-30 替换策略：能很好的实现程序的局部性

常用替换算法：先进先出(FIFO),近期最少使用(LRU),随机法

随机：随机选择被替换的块(随机函数发生器)

优点：简单，易于实现，对硬件调试很有用

缺点：没有考虑块被使用的情况，反映不了程序的局部性

FIFO:选择最早调入 cache 的字块进行替换，不需记录各种字块的使用情况

优点：简单，易于实现；

缺点：虽利用各块进入 cache 顺序这一"历史"信息，但还不能正确反映程序局部性，最先进入的块可能是经常使用的块

LUR:最近最少被使用的作为替换块

优点：反映了程序的局部性，失效率是三种方法最低的

缺点：复杂，硬件实现困难，特别是 cache 容量增加时，实现代价越来越高，而且经常只能近似实现，选最久没有被使用的块作为替换。

☞4-31 写策略->了解即可

写直达：写 cache 同时写主存

优点：简单，MM 中数据总是最新的，对 I/O 多处理机系统非常重要

缺点：写停顿，write stall

回写：只写 cache，当替换的时候才回写主存，需设置修改位

优点：写速度快

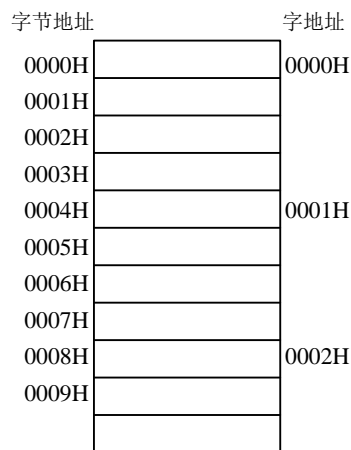
缺点：cache 与 MM 一致性问题

写失效：要写的单元不在 cache 中

☞4.6 某机字长为 32 位，其存储容量是 64KB，按字编址其寻址范围是多少？若主存以字节编址，试画出主存字地址和字节地址的分配情况。

答：存储容量是 64KB 时，按字节编址的寻址范围就是 64K，

如按字编址，其寻址范围为： $64K / (32/8) = 16K$



主存字地址和字节地址的分配情况：如图

4.7 一个容量为 $16K \times 32$ 位的存储器，其地址线和数据线的总和是多少？当选用下列不同规格的存储芯片时，各需要多少片？

1K×4 位，2K×8 位，4K×4 位，16K×1 位，4K×8 位，8K×8 位

地址线和数据线的总和 = $14 + 32 = 46$ 根；

各需要的片数为：

1K×4： $16K \times 32 / 1K \times 4 = 16 \times 8 = 128$ 片

2K×8： $16K \times 32 / 2K \times 8 = 8 \times 4 = 32$ 片

4K×4： $16K \times 32 / 4K \times 4 = 4 \times 8 = 32$ 片

16K×1： $16K \times 32 / 16K \times 1 = 32$ 片

4K×8： $16K \times 32 / 4K \times 8 = 4 \times 4 = 16$ 片

8K×8： $16K \times 32 / 8K \times 8 = 2 \times 4 = 8$ 片

4.11 一个 $8K \times 8$ 位的动态 RAM 芯片，其内部结构排列成 256×256 形式，存取周期为 $0.1\mu s$ 。试问采用集中刷新、分散刷新及异步刷新三种方式的刷新间隔各为多少？

(1) 集中刷新间隔为：2ms（共 20000 个存取周期）

(2) 分散刷新间隔为：256 个存取周期(存取周期为 $0.2\mu s$) $256 \times (0.1\mu s + 0.1\mu s) = 51.2\mu s$

(3) 异步刷新间隔为： $2ms / 256 = 7.8\mu s$ ，即每隔 $7.8\mu s$ 刷新一行，但对每行来说，刷新间隔仍为 2ms。

4.13 设有一个 $64K \times 8$ 位的 RAM 芯片，试问该芯片共有多少个基本单元电路(简称存储基元)?欲设计一种具有上述同样多存储基元的芯片，要求对芯片字长的选择应满足地址线和数据线的总和为最小，试确定这种芯片的地址线和数据线，并说明有几种解答。

存储基元总数 = $64K \times 8$ 位 = $512K$ 位 = 219 位；

思路：如要满足地址线和数据线总和最小，应尽量把存储元安排在字向，因为地址位数和字数成 2 的幂的关系，可较好地压缩线数。

设地址线根数为 a，数据线根数为 b，则片容量为： $2^a \times b = 2^{19}$ ； $b = 2^{19-a}$ ；

若 $a = 19$ ， $b = 1$ ，总和 = $19 + 1 = 20$ ；

$a = 18$ ， $b = 2$ ，总和 = $18 + 2 = 20$ ；

$a = 17$ ， $b = 4$ ，总和 = $17 + 4 = 21$ ；

$a = 16$ ， $b = 8$ ，总和 = $16 + 8 = 24$ ；

.....

由上可看出：芯片字数越少，芯片字长越长，引脚数越多。芯片字数减 1、芯片位数均按 2 的幂变化。

结论：如果满足地址线和数据线的总和为最小，这种芯片的引脚分配方案有两种：地址线 = 19 根，数据线 = 1 根；或地址线 = 18 根，数据线 = 2 根。

4.15 设 CPU 共有 16 根地址线, 8 根数据线, 并用 \overline{MREQ} (低电平有效) 作访存控制信号, R/\overline{W} 作读/写命令信号 (高电平为读, 低电平为写)。现有这些存储芯片: ROM (2K×8 位, 4K×4 位, 8K×8 位), RAM (1K×4 位, 2K×8 位, 4K×8 位) 及 74138 译码器和其他门电路 (门电路自定)。试从上述规格中选用合适的芯片, 画出 CPU 和存储芯片的连接图。要求如下:

- (1) 最小 4K 地址为系统程序区, 4096~16383 地址范围为用户程序区。
- (2) 指出选用的存储芯片类型及数量。
- (3) 详细画出片选逻辑。

答: (1) 地址空间分配图:

系统程序区 (ROM 共 4KB): 0000H-0FFFH

用户程序区 (RAM 共 12KB): 1000H-3FFFH

(2) 选片: ROM: 选择 4K×4 位芯片 2 片, 位并联

RAM: 选择 4K×8 位芯片 3 片, 字串联 (RAM1 地址范围为: 1000H-1FFFH, RAM2 地址范围为 2000H-2FFFH, RAM3 地址范围为: 3000H-3FFFH)

(1) 写出对应的二进制地址码。

ROM 区地址范围为 (4K×8 位):

A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1

因而可选 2 片 4K×4 位的 ROM 芯片。

RAM 区地址范围为:

A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1

RAM 容量为 12K×8 位, 因此可选 3 片 4K×8 位的 RAM 芯片。

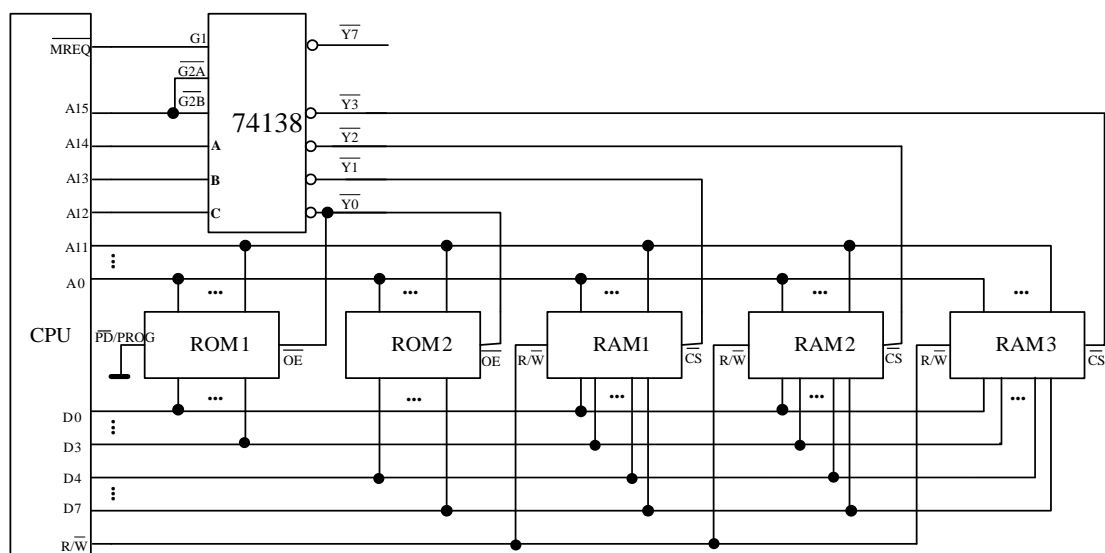
(2) 分配地址线:

A₁₁~A₀ 接 4K×4 位的 ROM 芯片的地址线和 4K×8 位的 RAM 芯片的地址线。

(3) 确定片选信号:

A₁₄A₁₃A₁₂ 连 3-8 译码器的 C、B、A 端, A₁₅ 连译码器的 G_{2A} 非使能端, \overline{MREQ} 连 3-8 译码器的 G_{2B} 非使能端, 译码器的 G₁ 使能端接 +5V 电源。译码器的 Y₀ 输出端同时接 2 片 ROM 芯片片选端; Y₁、Y₂、Y₃ 输出端分别接 3 片 RAM 芯片片选端。

(4) CPU 和存储器连接逻辑图及片选逻辑如下图(3)所示:



4.18 已知收到的海明码(配偶原则)为 1100100、1100111、1100000、1100001，检查上述代码是否出错？第几位出错？

假设接收到的汉明码为：c1'c2'b4'c3'b3'b2'b1'

纠错过程如下：

$$P1=c1'\oplus b4'\oplus b3'\oplus b1'$$

$$P2=c2'\oplus b4'\oplus b2'\oplus b1'$$

$$P3=c3'\oplus b3'\oplus b2'\oplus b1'$$

如果收到的汉明码为 1100100，则 p3p2p1=011，说明代码有错，第 3 位 (b4') 出错，有效信息为：1100

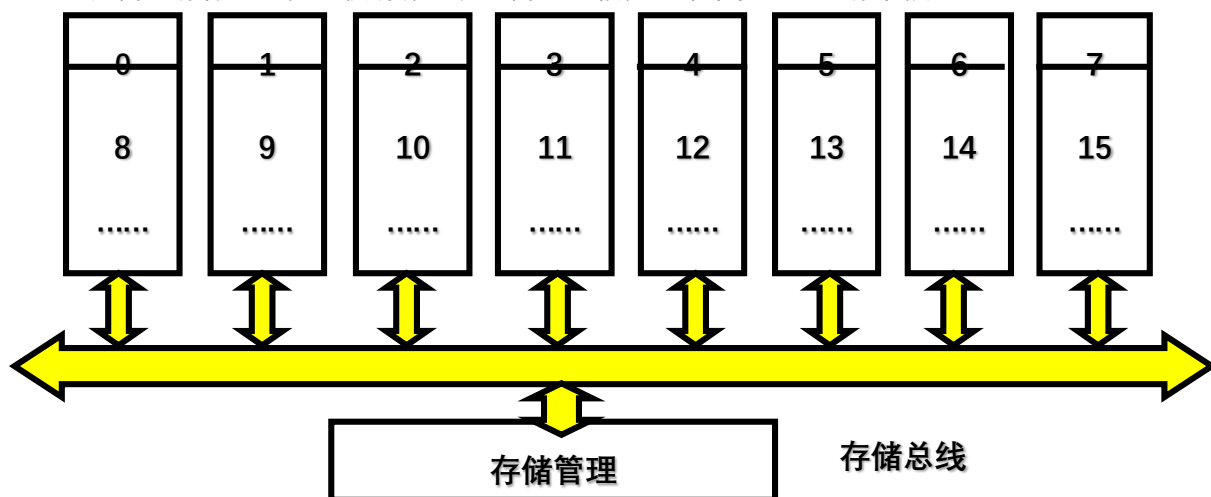
如果收到的汉明码为 1100111，则 p3p2p1=111，说明代码有错，第 7 位 (b1') 出错，有效信息为：0110

如果收到的汉明码为 1100000，则 p3p2p1=110，说明代码有错，第 6 位 (b2') 出错，有效信息为：0010

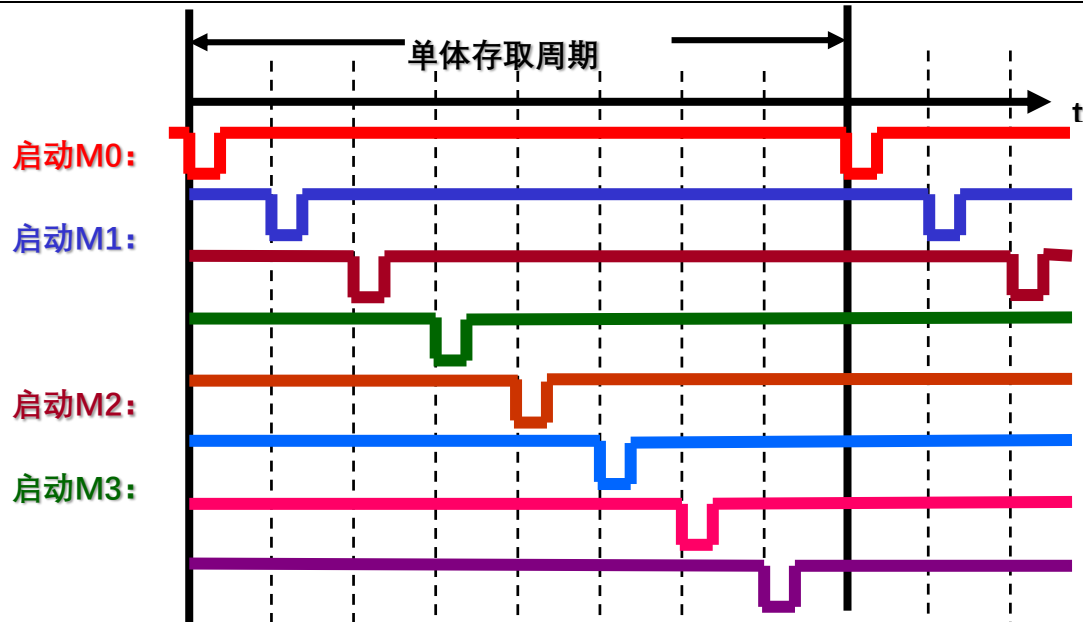
如果收到的汉明码为 1100001，则 p3p2p1=001，说明代码有错，第 1 位 (c1') 出错，有效信息为：0001

4.22 某机字长为 16 位，常规的存储空间为 64K 字，若想不改用其他高速的存储芯片，而使访存速度提高到 8 倍，可采取什么措施？画图说明。

若想不改用高速存储芯片，而使访存速度提高到 8 倍，可采取多体交叉存取技术



八体交叉访问时序



由图可知：每隔1/8个存取周期就可在存储总线上获得一个数据。

4.28 设主存容量为 256K 字，Cache 容量为 2K 字，块长为 4。

- (1) 设计 Cache 地址格式，Cache 中可装入多少块数据？
- (2) 在直接映射方式下，设计主存地址格式。
- (3) 在四路组相联映射方式下，设计主存地址格式。
- (4) 在全相联映射方式下，设计主存地址格式。
- (5) 若存储字长为 32 位，存储器按字节寻址，写出上述三种映射方式下主存的地址格式。

答：(1) 根据 Cache 容量为 2K 字= 2^{11} ，得 Cache 字地址为 11 位。根据块长为 4，且访存地址为字地址，得字块内地址为 2 位，即 $b=2$ ，且 Cache 共有 $2048/4=512=2^9$ 块，即 $c=9$ 。所以 Cache 地址格式为：

Cache 块号	字块内地址
9 位	2 位

(2) 根据主存容量为 256K 字= 2^{18} ，得主存字地址为 18 位。在直接映射方式下，主存字块标记为 $18-11=7$ ，故主存地址格式为：

主存字块标记	Cache 字块地址	字块内地址
7 位	9 位	2 位

(3) 根据四路组相联的条件，一组内有 4 块，得 Cache 共分 $512/4=128=2^7$ 组，即 $q=7$ ，主存字块标记为 $18-q-b=18-7-2=9$ 位，其地址格式为：

主存字块标记	组地址	字块内地址
9 位	7 位	2 位

(4) 在全相联映射方式下，主存字块标记为 $18-b=18-2=16$ 位，其地址格式为：

主存字块标记	字块内地址
16 位	2 位
~ 88 ~	

(5) 若存储字长为 32 位，存储器按字节寻址，上述三种映射方式下主存的地址格式分别为：

直接映射：

主存字块标记	Cache 字块地址	字地址	字节地址
7 位	9 位	2 位	2 位

四路组相联映射：

主存字块标记	组地址	字地址	字节地址
--------	-----	-----	------

全相联映射：

9 位	7 位	2 位	2 位
主存字块标记	字地址	字节地址	
16 位	2 位	2 位	

4.31 设主存容量为 1MB，采用直接映射方式的 Cache 容量为 16KB，块长为 4，每字 32 位。试问主存地址为 ABCDEH 的存储单元在 Cache 中的什么位置？

解：主存和 Cache 按字节编址，

Cache 容量 $16KB=2^{14}B$ ，地址共格式为 14 位，分为 $16KB/(4*32/8B)=2^{10}$ 块，每块 $4*32/8=16B=2^4B$ ，Cache 地址格式为：

Cache 字块地址 (10 位)	字块内地址 (4 位)
-------------------	-------------

主存容量 $1MB=2^{20}B$ ，地址共格式为 20 位，分为 $1MB/(4*32/8B)=2^{16}$ 块，每块 2^4B ，采用直接映射方式，主存字块标记为 $20-14=6$ 位，主存地址格式为：

主存字块标记 (6 位)	Cache 字块地址 (10 位)	字块内地址 (4 位)
--------------	-------------------	-------------

主存地址为 ABCDEH=1010 1011 1100 1101 1110B，主存字块标记为 101010，Cache 字块地址为 11 1100 1101，字块内地址为 1110，故该主存单元应映射到 Cache 的 101010 块的第 1110 字节，即第 42 块第 14 字节位置。或者在 Cache 的第 11 1100 1101 1110=3CDEH 字节位置。

第五章 输入输出系统

5-1 名词术语：

输入设备:外界信息转换成计算机能是别的信息

输出设备:计算机 01 信息形式转换为设备或人能接收的信息

外围设备:输入设备，输出设备统称外围设备

输入/输出模块

5-2 设备与 CPU 连接方式：

① 直接连接：主存 \leftrightarrow CPU \leftrightarrow I/O

② 总线连接：P157 图 5.2 优点：屏蔽设备复杂性；关键：I/O 接口

5-3 I/O 基本方法

① 无条件传送：慢速设备

② 接口逻辑：功能：连接器，转换器。

I/O 接口功能:设备选择，数据缓冲，格式转换，电平转换，传送控制命令，反馈设备状态

5-4 I/O 接口组成：数据寄存器，状态/控制寄存器，设备选择 I/O 控制 三部分

5-5 设备编址解决的问题：CPU 如何找到指定的设备。设备编址=I/O 接口编址

设备地址=I/O 地址=I/O 接口地址=I/O 端口地址=端口

实际上是 I/O 接口中寄存器的地址

☞5-6 编址方法：存储器映射 I/O，端口寻址

对 I/O 地址码的编址，两种方式：统一编址，不统一编址

☞5-7 I/O 基本方法：

①编程 I/O：程序查询，条件查询

I/O 由 CPU 直接完成，外设与 CPU 完全串行工作，最简单，最经济

②中断驱动 I/O

编程 I/O 存在的问题：高速 CPU 与低速设备之间的矛盾

☺中断：设备主动通知 CPU 进行数据交换的过程，中断是随机的->考：名词解释

计算机在执行程序过程中，当出现异常情况或特殊请求时，计算机停止现在程序的运行，转向对这些异常情况或特殊请求的处理，处理结束后再返回到现程序的间断处，继续执行原程序

中断源：中断产生的来源，复位，时钟中断，键盘中断，页面失效中断

中断系统：实现中断的软硬件系统

中断处理：

主程序：被中断的程序

中断服务子程序：处理中断事物的程序

☞5-8 I/O 设备与主机信息传送的控制方式

①程序查询方式：CPU 通过程序不断查询 I/O 设备是否已做好准备，从而控制 I/O 设备与主机交换信息。

只要一启动 I/O 设备，CPU 便不断查询 I/O 设备的准备情况，从而终止了原程序的执行。CPU 在反复查询过程中，犹如就地“踏步”。另一方面，I/O 设备准备就绪后，CPU 要一个字一个字地从 I/O 设备取出，经 CPU 送至主存，此刻 CPU 也不能执行原程序，这种方式使 CPU 和 I/O 设备出去串行工作状态，CPU 的工作效率不高。

②程序中断方式：CPU 只有当 I/O 设备准备就绪并向 CPU 发出中断请求后才予以响应。

书 P163 图 5.9.5.11

中断好处：实时处理，并行处理，故障处理

☞5-9 接口：可以看作是两个系统或部件之间的交接部分，既可以是两种硬件设备之间的连接电路，也可以是两个软件之间的共同逻辑边界。

I/O 接口通常指主机与 I/O 设备之间设置的一个硬件电路及相应的软件控制。

☞5-10 I/O 总线包括数据线，设备选择线，命令线和状态线。

☞5-11 接口功能：选址功能，传送命令功能，传送数据功能，反映 I/O 设备工作状态的功能

☞5-12 I/O 接口分类：

按数据传送方式：并行，串行

按功能选择的灵活性：可编程接口，不可编程接口

按通用性：通用接口，专用接口

按数据传送的控制方式：程序型接口(慢速设备)，DMA 型接口(高速设备)

☞5-13 书 P191 图

☞5-14 CPU 一旦响应了 I/O 中断，就要暂停现执行程序，转去执行该设备的中断服务程序。不同的设备有不同的中断服务程序，每个服务程序都有一个入口地址，CPU 必须去找到这个入口地址。

☞5-15 书 P198 图 5.41

☞5-16 中断服务程序流程：保护现场，中断服务，恢复现场，中断返回

硬件：①响应中断，②关中断，③保护断点，④向量指向 PC，转设备服务

☞5-17 书 P201 图 5.43->考

☞5-18 中断类型：

①单重中断，多重中断

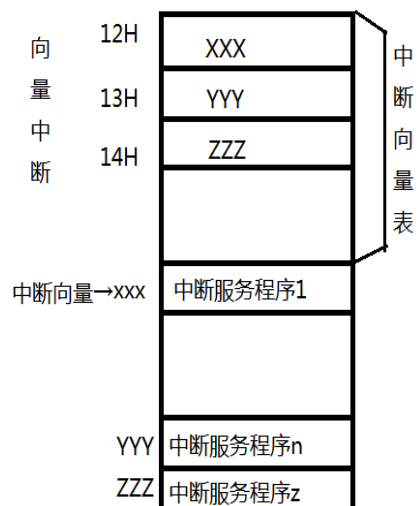
②单线中断，多线中断

③ ④ 向量中断，非向量中断->考

中断向量=中断服务例程的起始地址

向量地址=存储中断向量的存储单元地址

中断向量表：通常是系统内存的第 0 页



5-19 DMA 方式

目的：减少大批量数据传送时 CPU 的开销

方法：硬件实现 DMA，采用专用部件生成访存地址并控制访存过程

特点：数据交换不经过 CPU

DMAC：DMA 控制器，组成：

- { *counter*: 长度计数器，保存传送数据的字数
- { *Data Reg*: 数据寄存器
- { *Address Reg*: 地址寄存器，向地址总线提供访存地址
- { *DMA* 控制逻辑
- { *DMA* 状态逻辑
- { 中断控制逻辑

DMA 传送：DMA 时，CPU 对总线的使用权临时禁止

① 预处理：初始化 DMAC：

设置数据传送方向(读/写)

设置 I/O 接口地址

设置存储器起始地址

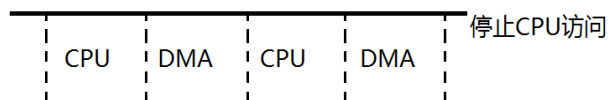
设置传送的数据量(计数器置初值)

② 数据传送：完成一批数据的传送，CPU 不参与

③ 后处理：DMA 结束后通过中断方式请求 CPU 处理

问题：CPU 与设备访存冲突：设备优先(设备无足够存储资源，不及时处理会丢失)

方法：① 停止 CPU 访问。

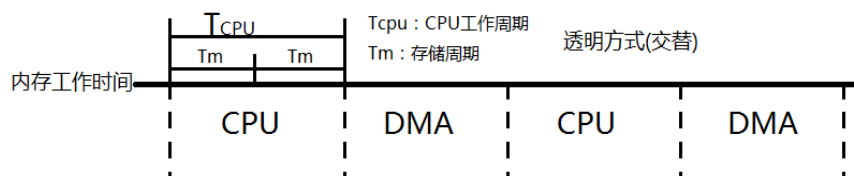


优点：控制简单

缺点：CPU，内存利用率低(可能同时空闲)

应用：高速设备

②透明方式(交替)

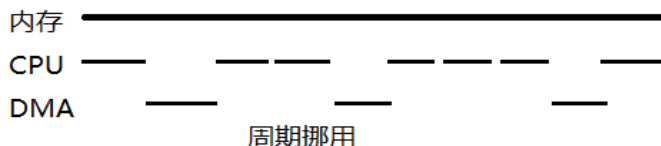


优点：不需总线控制权的申请建立和归还，总线控制权的转移很快，高效

缺点：控制复杂

应用：CPU 工作周期比存储周期长的情况

③周期挪用



方法：DMA 直接访存，**访存冲突->解释**

优点：既实现了 I/O，较好发挥了 CPU 内存的效率

缺点：单字传送，每次申请建立归还总线使用权，因而速度慢

应用：慢速设备

DMA 特点：CPU 对总线控制临时被禁止

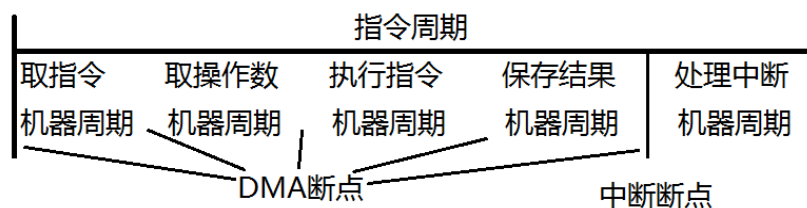
DMAC 接管总线控制权

CPU 前后处理，在 DMA 数据交换的过程中 CPU 可继续运行程序(并行)

与 I/O 设备在更大程度上并行工作，效率更高

适合高速、批量数据传送，eg.视频显示刷新，磁盘存储系统读写，存储器到存储器传输

DMA 响应时机



CPU 工作最小时间单位：CPU 周期，不同类型时间长度相同

DMA 可以中断指令的执行(无响应条件,当前机器周期结束);中断不可以(有响应条件,开中断)

DMA VS. 中断

① CPU 控制：数据交换是否经过 CPU

② 响应时机：机器周期 开中断

③ 适用情况：大数据量批量

DMA 接口类型:选择型：高速，每次逻辑只连一个设备，独占带宽(物理上连接多个)

多路型：低速，物理、逻辑都是多个，1 个设备不能独占接口

5.1 I/O 有哪些编址方式？各有何特点？

解：常用的 I/O 编址方式有两种：I/O 与内存统一编址和 I/O 独立编址；

特点：I/O 与内存统一编址方式的 I/O 地址采用与主存单元地址完全一样的格式，I/O 设备和主存占用同一个地址空间，CPU 可像访问主存一样访问 I/O 设备，不需要安排专门的 I/O 指令。

I/O 独立编址方式时机器为 I/O 设备专门安排一套完全不同于主存地址格式的地址编码，此时 I/O 地址与主存地址是两个独立的空间，CPU 需要通过专门的 I/O 指令来访问 I/O 地址空间。

5.10 什么是 I/O 接口？它与端口有何区别？为什么要设置 I/O 接口？I/O 接口如何分类？

I/O 接口一般指 CPU 和 I/O 设备间的连接部件；

I/O 端口一般指 I/O 接口中的各种寄存器。为了便于程序对这些寄存器进行访问，通常给每个寄存器分配一个地址编号，这种编号被称为 I/O 端口地址，相应的寄存器也叫作 I/O 端口。

I/O 接口和 I/O 端口是两个不同的概念。一个接口中往往包含若干个端口，因此接口地址往往包含有若干个端口地址。

由于 I/O 设备的物理结构和工作速率一般与主机差异很大，无法直接相连，因此通常通过 I/O 接口进行连接。

I/O 接口分类方法很多，主要有：

按数据传送方式分，有并行接口和串行接口两种；

按数据传送的控制方式分，有程序控制接口/程序中断接口/DMA 接口三种。

5.13 说明中断向量地址和入口地址的区别和联系。

中断向量地址和入口地址的区别：

向量地址是硬件电路（向量编码器）产生的中断源的内存中断向量表项地址编号，中断入口地址是中断服务程序首址。

中断向量地址和入口地址的联系：

中断向量地址可理解为中断服务程序入口地址指示器（入口地址的地址），通过它访存可获得中断服务程序入口地址。（两种方法：在向量地址所指单元内放一条 JMP 指令；主存中设向量地址表。参考 8.4.3）

5.14 在什么条件下，I/O 设备可以向 CPU 提出中断请求？

解：I/O 设备向 CPU 提出中断请求的条件是：I/O 接口中的设备工作完成状态为 1 ($D=1$)，中断屏蔽码为 0 ($MASK=0$)，且 CPU 查询中断时，中断请求触发器状态为 1 ($INTR=1$)。

5.15 什么是中断允许触发器？它有何作用？

解：中断允许触发器是 CPU 中断系统中的一个部件，他起着开关中断的作用（即中断总开关，则中断屏蔽触发器可视为中断的分开关）。

5.16 在什么条件和什么时间，CPU 可以响应 I/O 的中断请求？

(1) CPU 响应 I/O 设备提出中断请求的条件是：必须满足 CPU 中的允许中断触发器 EINT 为“1”；I/O 本身有请求又未被屏蔽；经排队后又被选中。

(2) CPU 响应中断的时间一定是在每条指令执行阶段的结束时刻。

CPU 响应 I/O 中断请求的条件和时间是：当中断允许状态为 1 ($EINT=1$)，且至少有一个中断请求被查到，则在一条指令执行完时，响应中断。

5.20 试比较单重中断和多重中断服务程序的处理流程，说明他们不同的原因。

5.21 中断向量通过什么总线送至什么地方？为什么？

5.22 程序查询方式和程序中断方式都是通过“程序”传送数据，两者的区别是什么？答：程序查询方式通过“程序”传送数据时，程序对 I/O 的控制包括了 I/O 准备和 I/O 传送两段时间。由于 I/O 的工作速度比 CPU 低得多，因此程序中要反复询问 I/O 的状态，造成“踏步等待”，严重浪费了 CPU 的工作时间。而程序中断方式虽然也是通过“程序”传送数据，但程序仅对 I/O 传送阶段进行控制，I/O 准备阶段不需要 CPU 查询。故 CPU 此时照样可以运行现行政程序，与 I/O 并行工作，大大提高了 CPU 的工作效率。

5.26 什么是多重中断？实现多重中断的必要条件是什么？

多重中断是指：当 CPU 执行某个中断服务程序的过程中，发生了更高级、更紧迫的事件，CPU 暂停现行中断服务程序的执行，转去处理该事件的中断，处理完返回现行中断服务程序继续执行的过程。

实现多重中断的必要条件是：在现行中断服务期间，中断允许触发器为 1，即开中断。

第六章 计算机的运算方法

☞6-1BCD 码：二进制编码的十进制数

☞6-2 数的表示形式：

① 无符号数

b_{n-1}	...	b_0
-----------	-----	-------

有符号数

S	b_{n-1}	...	b_0
---	-----------	-----	-------

② 定点数：小数点位置固定不变的数 定点小数，定点整数

③ 浮点数：小数点位置不是固定不变的数

现在的计算机都是定点计算机

☞6-3 机器码：把符号“数字化”的数称为机器数，带+-符号的数称为真值

☞6-4 原码

$$[x]_{\text{原}} = \begin{cases} 0, x & 2^n > x \geq 0 \\ 2^n - x & 0 \geq x > -2^n \end{cases} \quad \begin{matrix} x = +1110 & [x]_{\text{原}} = 0,1110 \\ x = -1110 & [x]_{\text{原}} = 1,1110 \end{matrix}$$

整数

x 为真值，n 为整数位数

$$[x]_{\text{原}} = \begin{cases} x & 1 > x \geq 0 \\ 1-x & 0 \geq x > -1 \end{cases} \quad \begin{matrix} x = +0.1101 & x_{\text{原}} = 0.1101 \\ x = -0.1101 & x_{\text{原}} = 1.1101 \end{matrix}$$

小数

例：求 x=0 的原码

解：设 $x = +0.0000$ $[+0.0000]_{\text{原}} = 0.0000$

设 $x = -0.0000$ $[-0.0000]_{\text{原}} = 1.0000$

对于整数： $[+0]_{\text{原}} = 0,0000$ $[-0]_{\text{原}} = 1,0000$

☞6-5 补码

$$[x]_{\text{补}} = \begin{cases} 0, x & 2^n > x \geq 0 \\ 2^{n+1} + x & 0 > x \geq -2^n \end{cases} \quad \begin{matrix} x = +1010 & [x]_{\text{补}} = 0,1010 \\ x = -1011000 & [x]_{\text{补}} = 1,0101000 \end{matrix}$$

整数

$$[x]_{\text{补}} = \begin{cases} x & 1 > x \geq 0 \\ 2 + x & 0 > x \geq -1(\text{mod } 2) \end{cases} \quad \begin{matrix} x = +0.1110 & [x]_{\text{补}} = 0.1110 \\ x = -0.1100000 & [x]_{\text{补}} = 1.0100000 \end{matrix}$$

小数

当真值为负，补码可用原码符号位外的每位取反，末位加 1 求的；原码可用补码除符号位外每位取反，末位加 1 求的。

或者：从右到左，找到第一个不为 0 的位，右边不变，左边取反

☞6-6 反码

$$[x]_{\text{反}} = \begin{cases} 0, x & 2^n > x \geq 0 \\ (2^{n+1} - 1) + x & 0 \geq x > -2^n (\text{mod } (2^{n+1} - 1)) \end{cases} \quad \begin{matrix} x = +1101 & [x]_{\text{反}} = 0,1101 \\ x = -1101 & [x]_{\text{反}} = 1,0010 \end{matrix}$$

整数

$$[x]_{反} = \begin{cases} x & 1 > x \geq 0 \\ (2 - 2^{-n}) + x & 0 \geq x > -1(\text{mod}(2 - 2^{-n})) \end{cases}$$

$x = +0.1101$
 $[x]_{反} = 0.1101$

$x = -0.1010$
 $[x]_{反} = 1.0101$

例：求 0 的反码

设 $x = +0.0000$, $[+0.0000]_{反} = 0.0000$

设 $x = -0.0000$, $[-0.0000]_{反} = 1.1111$

整数： $[+0]_{反} = 0,0000$

$[-0]_{反} = 1,1111$

☞6-7 三种机器码小结：

最高位为符号位，书写上用，(整数).(小数)将数值部分和符号位隔开

对于正数：原码=补码=反码

对于负数：符号位为 1，其数值部分：原码除符号位外每位取反，末位+1 变成补码

原码除符号位外每位取反，变成反码

☞6-8P225 例 6.1

二进制代码	无符号数	原码对应	补码对应	反码对应
00000000	0	+0	+0	+0
00000001	1	+1	+1	+1
00000010	2	+2	+2	+2
..
01111111	127	+127	+127	+127
10000000	128	-	-128	-127
10000001	129	-	-127	-126
..
11111101	253	-125	-	-
11111110	254	-126	-	-
11111111	255	-127	-	-

☞6-9 移码

$[x]_{移} = 2^n + x \quad (2^n > x \geq -2^n)$ x 为真值， n 为整数位数

$x = 10100 \quad [x]_{移} = 2^5 + 10100 = 1,10100$

$x = -10100 \quad [x]_{移} = 2^5 - 10100 = 0,01100$ 用逗号将符号与数值部分隔开

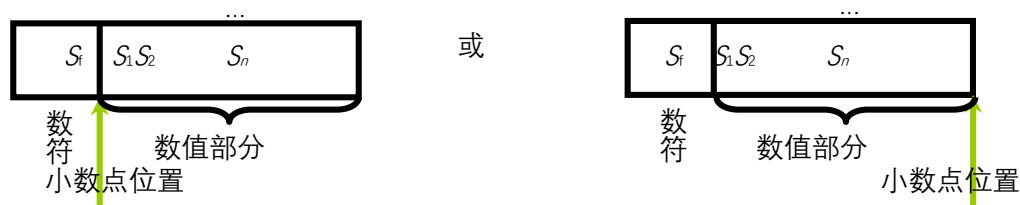
补码与移码只差一个符号位

$[+0]_{移} = 2^5 + 0 = 1,00000 \quad [-0]_{移} = 2^5 - 0 = 1,00000 \quad [+0]_{移} = [-0]_{移}$

最小真值的移码全为 0

用移码表示浮点数的阶码能方便地判断浮点数阶码的大小

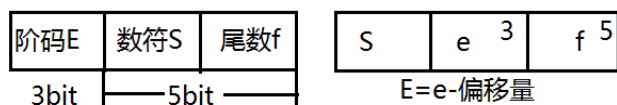
6-10 定点表示小数点固定在某一位置的数为定点数，有两种格式->考：定义



小数定点机： $-(1-2^{-n}) \sim (1-2^{-n})$ 整数定点机： $-(2^n-1) \sim (2^n-1)$

纯小数，纯整数

浮点表示 $N = (-1)^S \times 2^E \times f$ ->考：定义，为什么用？

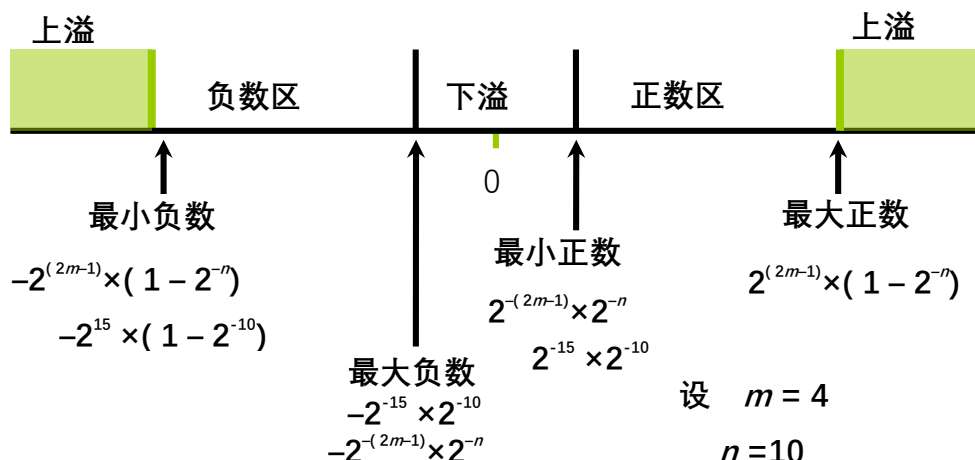


$$x = 0.253125 = 0.01001_2 = 0.1001 \times 2^{-1}$$

① $E = -1$, $f = 0.1001$ E 补 = 111 f 补 = 0.1001 x 补 = 111,01001

② e : 3bit, $e = E + \text{偏移值} = -1 + 3 = 2 = 010$, x 补 = 0010 1001

浮点数表示范围：



定点表示是特殊的浮点表示

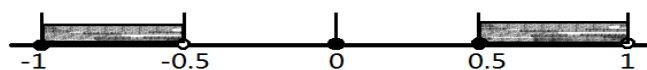
为什么用浮点表示：①扩大取值范围②提高表示精度

6-11 规格化

目的：①在尾数中表示更多的位，提高精度②数据表示的唯一性

机器零：全为 0，特殊的数据编码

规格化：将非规格化数转化成规格化数的过程



为什么不包含 -0.5？原补码一样

定 1 用丁点小数，定点整数都能表示

00.1XXX } 规格化表示
11.0XXX }

00.0XXX } 左规：尾数左规1位，阶码-1
11.1XXX } 左规至规格化数为止

01.XXXX } 右规：尾数左规1位，阶码+1
10.XXXX } 右规至规格化数为止

->考 填空 判断 双符号位不同，右规一次，不是溢出

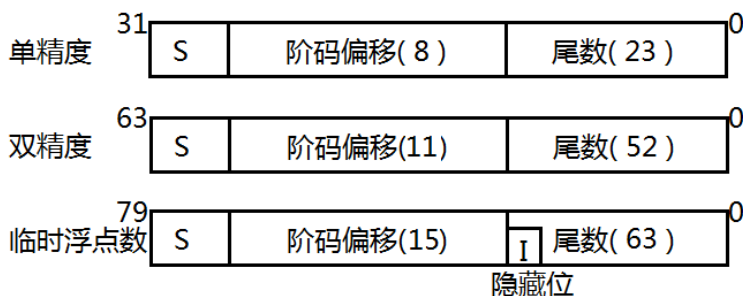
基数=2，尾数最高位为 1 的数为规格化数

基数=4，尾数最高两位不全为 0 的数为规格化数。规格化时，尾数左移两位，阶码-1；尾数右移两位，阶码+1。

基数=8，尾数最高三位不全为 0 的数为规格化数。规格化时，尾数左移 3 位，阶码-1；尾数右移 3 位，阶码+1。

一般基数 r 越大，可以表示的浮点数范围越大，而且所表示的数的个数越多。但 r 越大，浮点数的精度反而下降。如 r=16，因其规格化的尾数最高三位可能出现零，故与其尾数尾数相同的 r=2 的浮点数相比，后者可能比前者多 3 位精度。

6-12 IEEE754 标准



单精度：23 位尾数表示 24 位有效数字；双精度：53—54。有隐藏位

尾数的整数位，临时实数显示存储，单双精度是隐含的

浮点数运算用 80 位的临时浮点数，一般至少 2 位保护位

$$N = (-1)^S \times 2^E \times f$$

e8bit, 偏移值： $2^{8-1} - 1 = 127$

$2^8 = 256, 0 \sim 255$

全 0 全 1 有特殊指派，1~254 共用 e，共 254 个

真指数 $E = e - \text{偏移值} = e - 127, -126 \sim 127$

规格化数： $(-1)^S \times 1.f \times 2^{e-127}$

非规格化： $(-1)^S \times 0.f \times 2^{e-126}$

例：转成单精度浮点数 -5， -6

-5：二进制 -101；尾数 1.f $(-1)^1 \times 1.01 \times 2^2$ ， e=129

$-5 = (-1)^1 \times 2^2 \times 1.0100 = (-1)^1 \times 2^{129-127} \times 1.0100$ ， 1 10000001 0100 0000

-6: $(-1)^1 \times 2^{129-127} \times 1.1$, 1 1000 0001 1000 0000

例：单精度转十进制 42E48000

0100 0010 1110 0100 1000 0000 0000 0000

0|100 0010 1|110 0100 1000 0000 0000 0000

$(-1)^0 \times 2^6 \times (1+1/2+1/4+1/32+1/256) = 114.25$

☞6-13 定点运算-移位运算

算术移位：符号位不变

	码 制	添补代码
正数	原码、补码、反码	0
负	原 码	0
	补 码	左移 添 0
		右移 添 1
	反 码	1

☞6-14 定点数加法

整数 $[A]_{补} + [B]_{补} = [A + B]_{补} \pmod{2^{n+1}}$

小数 $[A]_{补} + [B]_{补} = [A + B]_{补} \pmod{2}$

证明补码加法公式->考：证明

证明：小数 $[x]_{补} = \begin{cases} x & 1 > x \geq 0 \\ 2 + x & 0 > x \geq -1 \pmod{2} \end{cases}$ ，约束条件： $|x| < 1, |y| < 1, |x+y| < 1$

	x	y	x+y
1	=0	=0	=0
2	>0	>0	>0
3	>0	<0	>0 或 <0
4	<0	>0	>0 或 <0
5	<0	<0	<0

情况 1：左=0=右

情况 2：左=x+y=x+y=右边

情况 3：左=x+(2+y)=x+y+2

若>0,右边=x+y 若<0,右边=2+x+y

mod2,左边=右边

情况 4：同 3

情况 5：左=2+x+2+y 右=x+y+2 mod2,左边=右边

☞6-15 定点数减法

整数 $[A]_{\text{补}} - [B]_{\text{补}} = [A - B]_{\text{补}} \pmod{2^{n+1}}$

小数 $[A]_{\text{补}} - [B]_{\text{补}} = [A - B]_{\text{补}} \pmod{2}$

温馨提示: $[-Y]_{\text{补}} = -[Y]_{\text{补}}$ 包含符号位各位取反, 末位加 1

从左到右, 第一位不为 0 的数, 右边不变, 左边取反

$[B]_{\text{补}} = 00011000$

Eg. $[-B]_{\text{补}} = 11101000$

6-16 溢出检测: 补码定点加减法判断溢出有两种方法: 一/两位符号位判断

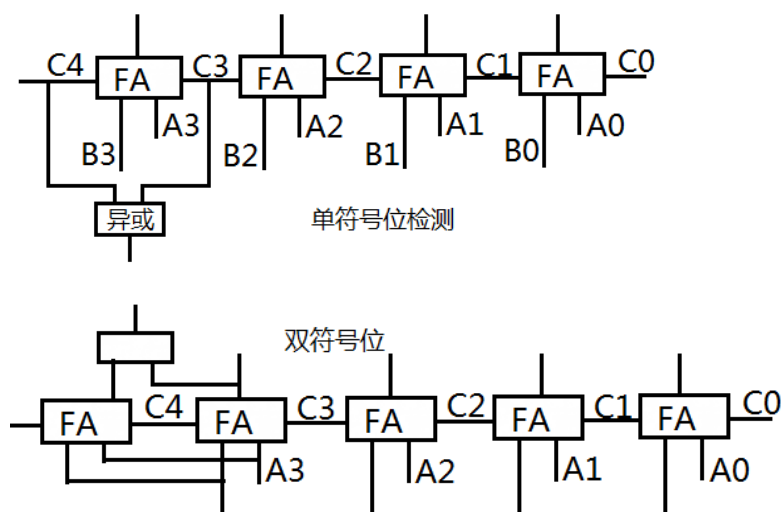
双符号位: 11,00 不溢出

Fc: 数值位最高进位位, Fs: 符号位进位位

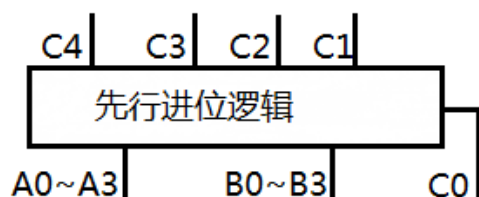
$v = fc \oplus fs = 1$ 溢出, $=0$ 不溢出

6-17 一位全加器:

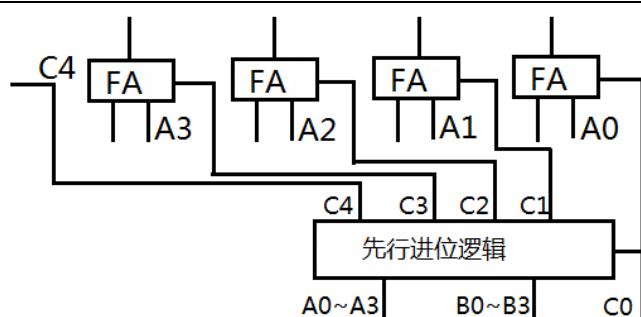
行波进位加法器:



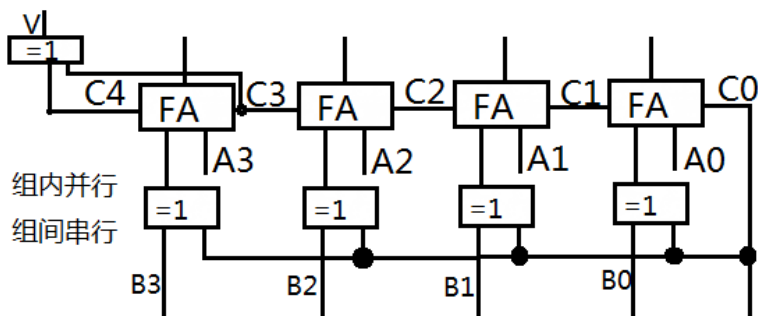
先行进位逻辑->考设计 理解



书 p285 图 6.21, 思想了解, 电路图不管



检测位同以上两图



6-18 定点数乘法

1 位 Booth 算法->背

已知 $[x]_{\text{补}}, [y]_{\text{补}} = y_0 y_1 y_2 \dots y_{n+1}$, 求 $[x \square y]_{\text{补}}, [x]_{\text{补}} \square y$

算法: ① $[y]_{\text{补}} = y_0 y_1 y_2 \dots y_{n+1}$, 且 $y_{n+1} = 0$

② $[z]_{\text{补}} = [x \square y]_{\text{补}}$, 且 $[z_0]_{\text{补}} = 0$

③ $[y]_{\text{补}} = y_0 y_1 y_2 \dots y_{n+1}$ 从低位到高位每次用乘数的两位 $y_n y_{n+1}$

运算求的部分积 $[z_i]_{\text{补}} = 2^{-1}((y_{n-i+2} - y_{n-i+1}) \cdot [x]_{\text{补}} + [z_{i-1}]_{\text{补}})$

④重复③ $n+1$ 次, 最后一次不移位, 得 $2n+1$ 位积, 包括1位符号位

$$[x]_{\text{补}} = x_0 x_1 \dots x_n, \text{ 则 } x = -x_0 + \sum_{i=1}^n x_i 2^i$$

补码与真值:

补码移位: 若移位以后不溢出, 则以下结论成立

右移一位相当于除 2, 左移一位相当于乘 2。

$$[x \cdot y]_{\text{补}} = [x]_{\text{补}} \cdot y$$

例: $x = +0.0011_2$, $y = -0.1011_2$, 求 $[x \cdot y]_{\text{补}}$ ->考: 几次加法? 几次移位? 大题: 运算过程

$$[x]_{\text{补}} = 0.0011, [-x]_{\text{补}} = 1.1101, [y]_{\text{补}} = 1.0101$$

n 次加法, $n-1$ 次右移

$$[x \cdot y]_{\text{补}} = 1.1101 \ 1111$$

0	0	.	0	0	0	0	1	.	0	1	0	1	0		
1	1	.	1	1	0	1									$+[-x]_{\text{补}}$
1	1	.	1	1	0	1									
1	1	.	1	1	1	0	1		1	0	1	0	1		$\rightarrow 1$
0	0	.	0	0	1	1									$+ [x]_{\text{补}}$
0	0	.	0	0	0	1	1								
0	0	.	0	0	0	0	1	1							$\rightarrow 1$
1	1	.	1	1	0	1									$+ [-x]_{\text{补}}$
1	1	.	1	1	0	1	1	1							
1	1	.	1	1	1	0	1	1	1			1	0	1	$\rightarrow 1$
0	0	.	0	0	1	1									$+ [x]_{\text{补}}$
0	0	.	0	0	0	1									
0	0	.	0	0	0	0	1	1	1	1			1	0	$\rightarrow 1$
1	1	.	1	1	0	1									$+ [-x]_{\text{补}}$
1	1	.	1	1	0	1	1	1	1	1					最后一步不移位

☞6-19 浮点数加减法

步骤：对阶，尾数加减，规格化，舍入，检查溢出

① 对阶->考：作用，解决什么问题，结果是什么？

$$x = fx \square 2^{Ex} \quad y = fy \square 2^{Ey}$$

$$\Delta E = \begin{cases} 0 & Ex = Ey \\ > 0 & Ex > Ey \quad y \text{向} x \text{看齐}, fy \rightarrow 1, Ey + 1 \\ < 0 & Ex < Ey \quad x \text{向} y \text{看齐}, fx \rightarrow 1, Ex + 1 \end{cases}$$

② 小阶向大阶看齐，尾数右移

③ 舍入：

舍入前		舍入后末位数据		
末位	欲舍入位	截断法	恒置 1	0 舍 1 入
0	0	0	1	0
0	1	0	1	1
1	0	1	1	1
1	1	1	1	10

☞6-20 浮点数乘法

阶码加减，尾数乘除，规格化，舍入，检查溢出

第七章 指令系统及第八、九章

☞7-1 指令，程序->考：名词解释

指令：计算机执行某种操作的命令

程序：由有序的指令串构成，程序要解决一个具体的问题

指令系统：一台计算机能执行的全部指令的集合

指令系统的重要性：软件编程的基础，硬件设计的依据，综合考虑计算机的软硬件是计算机设计的关键因素。

☞7-2 操作码->考：设计 定长，变长

操作码用来指明该指令所要完成的操作。通常位数反映了机器的操作种类，即机器允许的指令条数，如 7 位

→ $2^7=128$ 条指令

指令的一般格式:

操作码字段	地址码字段
-------	-------

操作码: 定长, 变长

固定长度操作码: 操作码长度(占二进制位数)固定不变

硬件设计相对简单

指令译码时间开销小

指令空间利用率较低

可变长度操作码: 操作码长度随指令地址数目的不同而不同(可平均缩短指令长度)

硬件设计相对复杂

指令译码时间开销较大

指令空间利用率较高

例: 某机器采用固定长度指令系统, 16 位, 包括 3 地址指令 15 条, 双地址指令 10 条, 单地址指令若干, 每个地址占 4 位。问: 该指令系统最多容纳多少个单地址指令, 并设计该指令系统的操作码编码方案

析: 每条指令: 一个唯一操作码编码, 不同类型指令具有不同标识, 用扩展操作码方案

三 15 条, 1111

双 10 条, 6 个没用 $6 \times 16 = 96$ 个

7.3 什么是指令字长、机器字长和存储字长?

7.6 某指令系统字长为 16 位, 地址码取 4 位, 试提出一种方案, 使该指令系统有 8 条三地址指令、16 条二地址指令、100 条一地址指令。

固定操作码为 4 位。

8 条三地址指令操作码为: 0000~0111 (剩下 1000~1111 共 8 个扩展窗口)

16 条二地址指令操作码为: 1000 0000~1000 1111

(剩下 1001 0000~1111 1111 共 112 个扩展窗口)

100 条一地址指令操作码为: 10010000 0000~10010000 1111

10010001 0000~10010001 1111

10010010 0000~10010010 1111

10010011 0000~10010011 1111

10010100 0000~10010100 1111

10010101 0000~10010101 1111

10010110 0000~10010110 0011

7.9 试比较间接寻址和寄存器间接寻址。

7.10 试比较基址寻址和变址寻址。

1) 都可有效地扩大指令寻址范围。

2) 基址寻址时, 基准地址由基址寄存器给出, 地址的改变反映在位移量 A 的取值上; 变址寻址时, 基准地址由 A 给出, 地址的改变反映在变址值的自动修改上, 变址值由变址寄存器给出。

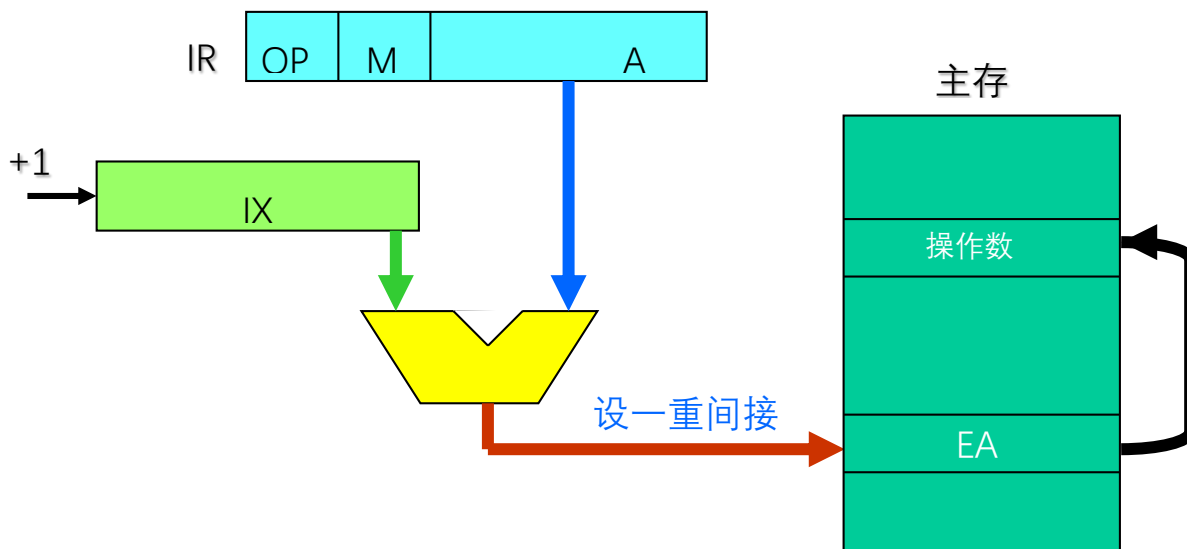
3) 基址寄存器内容通常由系统程序设定, 变址寄存器内容通常由用户设定。

4) 基址寻址适用于程序的动态重定位, 变址寻址适用于数组或字符串处理, 适用场合不同。

7.11 画出先变址再间址及先间址再变址的寻址过程示意图。

解: 1) 先变址再间址寻址过程简单示意如下:

$$EA = [(IX) + A], \quad (IX) + 1 \rightarrow IX$$



7.14 设相对寻址的转移指令占两个字节，第一字节是操作码，第二字节是相对位移量（用补码表示），假设当前转移指令第一字节所在地址为 2000H，且 CPU 每取出一个字节便自动完成 $(PC) + 1 \rightarrow PC$ 的操作。试问当执行“JMP * +8”和“JMP * -9”指令时，转移指令第二字节的内容各为多少？

(1) 根据汇编语言指令 JMP * +8，即要求转移后的目标地址为 $2000H + 08H = 2008H$ ，但因为 CPU 取出该指令后 PC 值已修改为 2002H，故转移指令第二字节的内容应为 +6（十进制），写成补码为 06H。

(2) 同理，执行“JMP * -9”指令时，要求转移指令第二字节的内容应为 -11（十进制），写成补码为：F5H

8.1 CPU 有哪些功能？画出其结构框图并简要说明各个部件的作用。

答：参考 P328 和图 8.2。

8.2 什么是指令周期？指令周期是否有一个固定值？为什么？

指令周期是 CPU 每取出并执行一条指令所需的全部时间，也即 CPU 完成一条指令的时间。由于各种指令操作功能不同，因此各种指令的指令周期是不同的，指令周期的长短主要和指令在执行阶段的访存次数和执行阶段所需要完成的操作有关。

8.3 画出指令周期的流程图，分析说明图中每个子周期的作用。

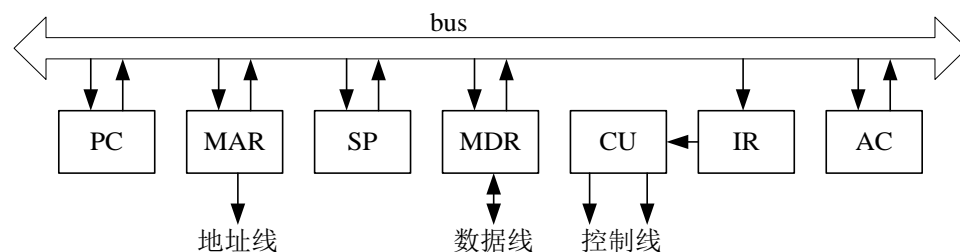
答：参看 P343 及图 8.8。

8.4 设 CPU 内有下列部件：PC、IR、SP、AC、MAR、MDR 和 CU，要求：

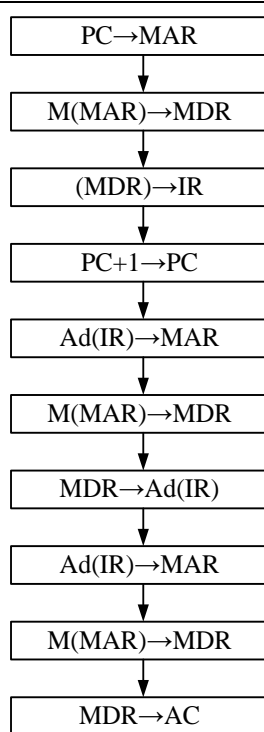
(1) 画出完成间接寻址的取数指令 LDA@X（将主存某地址单元 X 的内容取至 AC 中）的数据流（从取指令开始）。(2) 画出中断周期的数据流。

解：CPU 中的数据流向与所采用的数据通路结构直接相关，不同的数据通路中的数据流是不一样的。常用的数据通路结构方式有直接连线、单总线、双总线、三总线等形式，目前大多采用总线结构，直接连线方式仅适用于结构特别简单的机器中。

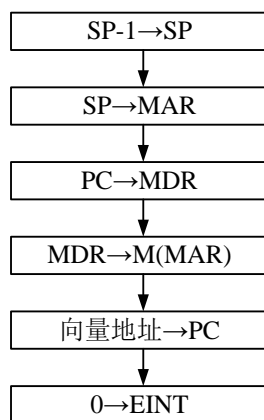
为简单起见，本题采用单总线将题中所给部件连接起来，框图如下：



(1) LDA@X 指令周期数据流程图：



(2) 中断周期流程图如下:



8.5 中断周期前是什么阶段？中断周期后又是什么阶段？在中断周期 CPU 应完成什么操作？

从 CPU 机器周期的时序层次来看，中断周期前是指令的执行阶段。中断周期后是取指令阶段。在中断周期 CPU 应完成关中断、保存断点和转中断服务程序入口三个操作。

9.2 控制单元的功能是什么？其输入受什么控制？

答：控制单元的主要功能是发出各种不同的控制信号。其输入受时钟信号、指令寄存器的操作码字段、标志和来自系统总线的控制信号的控制。

9.3 什么是指令周期、机器周期和时钟周期？三者有何关系？

指令周期是 CPU 每取出并执行一条指令所需的全部时间，也即 CPU 完成一条指令的时间。机器周期是所有指令执行过程中的一个基准时间，通常以存取周期作为机器周期。时钟周期是机器主频的倒数，也可称为节拍，它是控制计算机操作的最小单位时间。

一个指令周期包含若干个机器周期，一个机器周期又包含若干个时钟周期，每个指令周期内的机器周期数可以不等，每个机器周期的时钟周期数也可以不等。

9.4 能不能说 CPU 的主频越快，计算机的运行速度就越快？为什么？

不能说 CPU 的主频越快，计算机的运行速度就越快。因为机器的速度不仅与主频有关，还与机器周期中所含的时钟周期数以及指令周期中所含的机器周期数有关。同样主频的机器，由于机器周期所含时钟周期数不同，机器的速度也不同。机器周期中所含时钟周期数少的机器，速度更快。

此外，机器的速度还和其他很多因素有关，如主存的速度、机器是否配有 Cache、总线的数据传输率、硬盘的速度以及机器是否采用流水技术等。

⑩10.15 设控制存储器的容量为 512×48 位，微程序可在整个控存空间实现转移，而控制微程序转移的条件共有 4 个（采用直接控制），微指令格式如下：

试问微指令中的三个字段分别为多少位？

解：微指令各字段位数分配如下：操作控制 35，转移条件 4，下地址 9

⑩10.21 下表给出 8 条微指令 I1~I8 及所包含的微命令控制信号，设计微指令操作控制字段格式，要求所使用的控制位最少，而且保持微指令本身内在的并行性。

解：为使设计出的微指令操作控制字段最短，并且保持微指令本身内在的并行性，应采用混合编码法。首先找出互斥的微命令组，为便于分析，将微命令表重画如下：

由表中微命令的分布情况可看出：a、b、c、d、e 微命令的并行性太高，因此不能放在同一字段中。另外，由分析可知，在 2、3、4 分组的互斥组中，3 个一组的微命令互斥组对控制位的压缩作用最明显。因此，应尽可能多的找出 3 个一组的互斥组。现找出的互斥组有：cfj, dij, efh, fhi, bgj, ehj, efj……等等。

从中找出互不相重的互斥组有两个：dij, efh。则：微指令操作控制字段格式安排如下：

1	1	1	1	2	2
A	b	c	g	efh	dij

各字段编码分配如下：a: 0 无操作, 1 a 微命令；

b: 0 无操作, 1 b 微命令；

c: 0 无操作, 1 c 微命令；

g: 0 无操作, 1 g 微命令；

dij : 00 无操作；01 d 微命令；10 i 微命令；11 j 微命令；

efh: 00 无操作；01 e 微命令；10 f 微命令；11 h 微命令

与采用直接控制法比较：直接控制法：10 个微命令需 10 位操作控制位；本方案中 10 个微命令需 8 位操作控制位，压缩了 2 位。