

杭州电子工业学院
2001 年研究生入学考试
《计算机组成原理》试卷

(本试卷共 4 页, 三大题)

请将答案写在答题纸上, 写在试卷上的不计分。

一. 下面的两组句子中各有 5 个正确的, 请选出, 多选则以前 5 个为准。(共 20 分, 每题 10 分)

(1) 以下描述中, 正确的有: ____。

☒ 若一个机器数的表示形式为 10000000, 则这个机器数可能是 $(-0)_{10}$ 的原码, 或者是 $(-127)_{10}$ 的补码, 或者是 $(-127)_{10}$ 的反码, 或者是 $(+0)_{10}$ 的移码。
X -128 的补码!

☒ 尾数决定了浮点数的精度, 阶码决定了浮点数的范围; 浮点数的溢出取决于阶码是否溢出, 而非尾数。*这跟溢出和阶码有关*

☒ 一个指令周期由若干个机器周期组成, 而一个机器周期通常又包含若干个时钟周期。

☒ 补码机器数算术左移时, 低位补 0, 符号位不变; 算术右移时, 高位补 1, 符号位不变。*P18*

☒ 线性流水线的时钟周期, 应选择它的过程段中处理时间最长的一个。

☒ 在微命令的分段直接编码方法中, 为实现并行操作, 相容性的微操作应被安排在同一字段, 而相斥性的微操作应被安排在不同字段中。*在 P180*

☒ 多模块交叉存储器实质上是一种并行存储器, 它能并行执行多个独立的读写操作, 它的体选信号是由地址总线的低位地址译码产生的。

☒ SRAM 是非易失性存储器, 而 DRAM 是易失性存储器。*都是易失性*

☒ Cache 与主存统一编址, 即主存空间的某一部分属于 Cache。*P160*

☒ Cache 的容量小, 速度快, 价格贵, 其功能全由硬件实现, 对用户是透明的。

(2) 以下描述中, 正确的有: ____。

☒ 当系统采用存储器与 I/O 设备统一编址的方案时, 通常不设置专用的 I/O 指令, 而用访存指令来访问 I/O 设备。*P247*

☒ 一台机器中, 所有机器指令和微指令的集合被称为这台计算机的指令系统。*所有机器指令的集合*

☒ 汇编语言与高级语言一样都是符号化的语言, 因此, 汇编语言不是面向机器的语言, 即与计算机的硬件结构无关。

☒ $N+1$ 位的补码机器数所能表示的数据范围是 $-2^n \sim +(2^n - 1)$ 。*N=1 10=-2 01=1*

☒ 微程序控制器的执行速度较硬联逻辑控制器的速度快, 而且易扩充, 因此目前多采用微程序控制器。*11=-1*

☒ 某中断源的中断向量包含该中断源的中断服务程序的入口地址, 而向量地址则是指该中断源的中断服务程序的入口地址的地址。

☒ CPU 响应一个中断请求的条件是一个 CPU 周期结束, 并且 CPU 允许中断。*当前一条指令执行完毕*

☒ 在程序查询方式中, CPU 与 I/O 设备是串行工作的, 并且 CPU 被 I/O

设备独占。

9. 四片 Am2901 和一片 Am2902 器件相配合，可以具有组内串行进位，组间超前进位的传递功能。
10. 相联存储器是按内容访问的存储器，通常用于快速查询的场合。

二. 计算题：(共 30 分)

1. (21 分) 设浮点数的格式为：阶码 6 位，包含一位符号位，尾数 10 位，包含一位符号位，阶码用移码表示，尾数用补码表示，排列顺序为：

阶符	阶码	数符	尾数
----	----	----	----

则按上述浮点数的格式：

$$[X]_{\text{移}} = 110010$$

$$[Y]_{\text{补}} = 10010$$

$$[M]_{\text{补}} = 0.110101000$$

$$= \frac{13}{64} \times 2^{18} = 53 \times 2^{13}$$

- ① (2 分) 若数 Z 的浮点数的 16 进制形式为 C9A8 H，求 Z 的十进制的真值。

- ② (4 分) 若 $(X)_{10} = 69/256$ ， $(Y)_{10} = -3.8125$ ，则求 X 和 Y 的规格化浮点数表示形式。

- ③ (5 分) 求 $[X-Y]_{\text{补}}$ (要求尾数用补码计算，列出计算步骤)。

- ④ (10 分) 求 $[X*Y]_{\text{补}}$ (要求阶码用移码计算，尾数用补码两位乘法计算，列出计算过程和算式)。

2. (9 分) 设某机寄存器字长 16 位，用 16 进制表示，已知：

变址寄存器内容为 0005H，PC 的内容为 0003H，内存中部分单元内容如下：

地址：内容	地址：内容
0002H: 000AH	0007H: 0006H
0003H: 0002H	0008H: 0007H
0004H: 0003H	0009H: 0008H
0005H: 0004H	000AH: 0009H
0006H: 0005H	000BH: 000AH

指令格式为：

操作码及寻址方式(8 位)	形式地址(8 位)
---------------	-----------

若当前指令分别为下列寻址方式时，试求出操作数有效地址及操作数各为多少？

寻址方式	有效地址	操作数
直接	① 0002H	② 000AH
间接	③ 000AH	④ 0009H
立即	_____	⑤ 0002H
变址	⑥ 0007H	⑦ 0006H
相对	⑧ 0005H	⑨ 0004H

三. 设计题：(共 50 分)

1. (25 分) 某机字长 16 位，CPU 地址总线 20 位，数据总线 16 位，存储器按字编址，CPU 的控制信号线有：MREQ# (存储器访问请求，低

电平有效), R/W# (读写控制, 低电平为写信号, 高电平为读信号)。

试问:

$$2^{20} = 1M8 = 1M \times 16 \text{ 位} = 2MB$$

① (2分) 该机可以配备的最大主存容量为多少?

② (4分) 若该机主存采用 $64K \times 1\text{bit}$ 的 DRAM 芯片 (内部为 4 个 128×128 阵列) 构成最大主存空间, 则共需多少个芯片? 若采用分散刷新方式, 单元刷新周期为 2ms , 则刷新信号的周期为多少时间?

主地址位为 20 位
数据地址总线位数!

③ (6分) 若为该机配备 $16K \times 16$ 位的 Cache, 每字块 16 字, 采用 4 路组相联映象, 试写出对主存地址各个字段的划分 (各个字段的位数); 若主存地址为 $03287H$, 则该地址映象到的 Cache 的哪一组?

④ (13分) 若已知该机已有 $4K \times 16$ 位的 ROM 存储器, 地址处于主存的最高端; 现在再用若干个 $16K \times 8$ 位的 SRAM 芯片形成 $128K \times 16$ 位的 RAM 存储区域, 起始地址为 $00000H$, 假设 SRAM 芯片有 CS# (片选, 低电平有效) 和 WE# (写使能, 低电平有效) 信号控制端; 试写出 RAM、ROM 的地址范围, 并画出 SRAM、ROM 与 CPU 的连接图, 请标明 SRAM 芯片个数、译码器的输入输出线、地址线、数据线、控制线及其连接。

2. (25分) 设某实验模型机中, 指令系统 64 条, 指令寻址空间 $64K$ 字, 机器字长 16 位, 通用寄存器 16 个; 采用微程序控制器方案。

① (4分) 如果有一条加法指令, 采用直接寻址方式, 功能如下:

$$(SR) + (ADR) \rightarrow DR$$

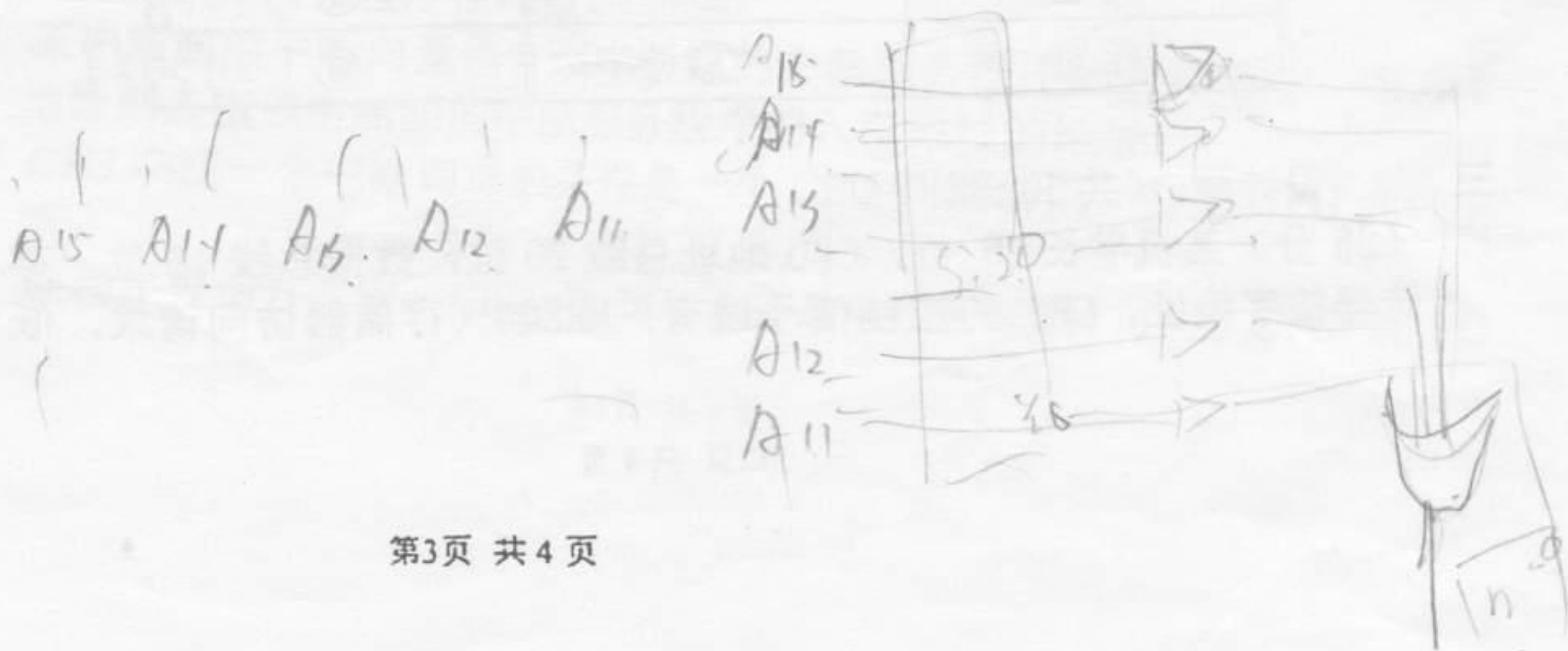
其中, SR, DR 分别为源寄存器号和目的寄存器号, ADR 为 16 位直接地址, 试设计其指令格式, 并写出各字段的位数。

② (6分) 系统中有 25 种微操作命令 (采用直接控制法), 有 4 个转移控制状态 (采用译码形式), 微指令格式中的下址字段 12 位, 则操作控制字段和判别测试字段各有几位? 控存的容量为多少 (字数 \times 字长)?

操作控制字段	判别测试字段	下址字段
--------	--------	------

③ (7分) 某指令的微程序流程图如下所示。其中: PC 是存放增 1 以前的 PC 值的寄存器; AR 是地址寄存器; $AR \rightarrow IR$ 是将 AR 的内容当作地址, 读存储器, 并将相应信息单元的内容送 IR; $MAP=0$ 是根据 IR 中的操作码转移到该指令的微程序入口; DR 是目的寄存器, 问: 这条指令包含了几个字? 每个字包含了什么内容? 是什么寻址方式? 描述指令功能, 及微指令的操作。

④ (8分) 设 ADD DR, IMME 指令由两个字组成, 立即数 IMME 位于指令第二字, 功能为: 将 DR 的内容加上立即数 IMME, 结果送回 DR 中; 试仿照上题写出该指令的微程序流程图。



PC → AR
PC → PC*

PC+1 → PC
MEM → IR

MAP# = 0

PC → AR
PC+1 → PC

MEM → AR

MEM → DR

PC →
PG1 →

~~DR → A~~ DR → AC₁

~~PC → AR~~ PC → AD

AR → MEM

MEM → AC₂

AC₁ + AC₂ → DR

PL → BUS
BUS → AR
PL+1

RAM → BUS
BUS → IR
J₁ = 0
数据

PL → BUS BUS → DA1
PL+1

DR → BUS
BUS → DA2

DA1 + DA2 → BUS
BUS → DR

