



厦门大学《计算机组成原理》课程期末试卷

软件学院 软件工程系 2017 级 软件工程专业

主考教师：曾文华 高星

试卷类型：(A 卷答卷)

一、 选择题（请从 A、B、C、D 中选择唯一的一个正确答案，15 小题，每小题 1 分，共 15 分；在答题纸填写答案时请写上每小题的对应编号）

- 冯·诺依曼计算机工作方式的基本特点是 B。
A. 多指令流单数据流 B. 按地址访问并顺序执行指令
C. 堆栈操作 D. 存储器按内容选择地址
- 不同的信号共用一组信号线，分时传送，这种总线传输方式是 D 传输。
A. 猝发 B. 并行
C. 串行 D. 复用
- 在三种集中式总线控制中， A 方式对电路故障最敏感。
A. 链式查询 B. 计数器定时查询
C. 独立请求 D. 链式查询和独立请求
- 在同步通信中，一个总线周期的传输过程是 C。
A. 先传输数据，再传输地址 B. 只传输数据
C. 先传输地址，再传输数据 D. 只传输地址
- 主存储器和 CPU 之间增加 Cache 的目的是 A。
A. 解决 CPU 和主存之间的速度匹配问题 B. 扩大主存储器的容量
C. 扩大 CPU 中通用寄存器的数量 D. 既扩大主存容量又扩大 CPU 通用寄存器数量
- 在程序的执行过程中，Cache 与主存的地址映射是由 C。
A. 操作系统来管理的 B. 程序员调度的
C. 硬件自动完成的 D. 编译器完成的
- 相联存储器是按 C 进行寻址的存储器。
A. 堆栈存取方式 B. 地址指定方式
C. 内容指定方式 D. 地址指定与堆栈存取方式结合
- 在 DMA 方式中，周期窃取是窃取一个 A。
A. 存取周期 B. 指令周期
C. CPU 周期 D. 总线周期
- 在小数定点机中，下述说法正确的是 B。
A. 原码能表示-1 B. 补码能表示-1
C. 反码能表示-1 D. 原码、补码、反码都能表示-1

10. 在定点补码运算器中, 若采用双符号位, 当 B 时表示结果溢出。
 A. 双符号位相同 B. 双符号位不同
 C. 双符号位均为 1 D. 双符号位均为 0
11. 长度相同但格式不同的两种浮点数, 假设前者阶码长、尾数短, 后者阶码短、尾数长, 其他规定均相同, 则它们可表示的数的范围和精度为 B。
 A. 两者可表示的数的范围和精度相同 B. 前者可表示的数的范围大但精度低
 C. 后者可表示的数的范围大且精度高 D. 前者可表示的数的范围大且精度高
12. 在 CPU 中跟踪指令后续地址的寄存器是 D。
 A. MAR B. IR
 C. MDR D. PC
13. 寄存器间接寻址方式中, 操作数在 C 中。
 A. 通用寄存器 B. 堆栈
 C. 主存单元 D. 指令本省
14. 为了缩短指令中地址码的位数, 应采用 D 方式。
 A. 立即数寻址 B. 间接寻址
 C. 直接寻址 D. 寄存器寻址
15. 在下列叙述中, D 能反映 RISC 的特征。
 A. 丰富的寻址方式 B. 难以用优化编译生成高效的目标代码
 C. 控制器采用微程序设计 D. 指令长度固定

二、 填空题（10 个空，每一空 1 分，共 10 分；在答题纸填写答案时请写上每个空格的对应编号）

- 总线判优控制是为了解决多个部件同时申请总线时的 使用权分配 问题，总线通信控制是为了解决通信双方 协调配合 问题。
- I/O 的编址方式可以分为独立编址和统一编址两大类，前者需有独立的 I/O 指令，后者可通过 访存指令 和 I/O 设备交换信息。
- 单重中断和多重中断的区别是：在单重中断的中断服务程序中，开中断指令位于 中断返回指令 之前；在多重中断的中断服务程序中，开中断指令位于 保护现场 之后。
- 毫微程序设计采用两级微程序设计方法，第一级微程序采用 垂直型 微指令，并行功能不强，但有严格的顺序结构；第二级微程序采用 水平型 微指令，具有很强的并行操作能力。
- 设总线的时钟频率为 8MHz，1 个总线周期等于 4 个时钟周期，总线宽度为 16 位，则总线的带宽为 32Mbps。（计算过程： $16/((1/8\text{MHz})*4)=16*8/4\text{ Mbps}=32\text{Mbps}$ ）
- 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、1 个停止位，要求每秒

传送 120 个字符，则传送的波特率为：1320 bps。（计算过程： $120 * (1+8+1+1) = 1320 \text{ bps}$ ）

7. 设转移指令的机器码为 2 个字节，其中第 1 个字节为操作码，第 2 个字节为地址码。已知当前指令地址为 240，要求转移到 200，则转移指令的第 2 个字节的机器码是（用十六进制表示）：D6H。
（ $200-242 = -42 = \text{D6H}$ ）

三、 判断题（下列表述正确的打√，错误的打×，10 小题，每小题 1 分，共 10 分；在答题纸填写答案时请写上每小题的对应编号）

1. 在 Cache 的写操作时，对 Cache 与主存单元同时修改的方法称为写回（或拷回）法。 ×
2. 汉明码可以发现 1 位出错，并能纠正 1 位出错。 √
3. 循环冗余校验码可以发现 1 位出错，但不能纠正 1 位出错。 ×
4. CPU 在指令执行周期结束时刻查询中断请求信号（在开中断情况下）。 √
5. 采用流水线技术的计算机一定是 RISC 计算机。 ×
6. 屏蔽技术可以改变中断响应的优先级次序。 ×
7. 在指令长度相同的情况下，所有指令的取指周期都是相同的。 √
8. 在微指令编码方式中，编码效率最低的是直接编码方式。 √
9. 动态微程序设计的控制存储器可采用掩模 ROM（Mask ROM）实现。 ×
10. 指令操作数所需的数据可以来自控制存储器。 ×

四、 名词解释（请写出下列英文缩写的中文全称，10 小题，每 1 小题 1 分，共 10 分；在答题纸填写答案时请写上每小题的对应编号）

- | | |
|--------------|-------------------------|
| 1. FLOPS | 每秒浮点运算次数 |
| 2. CPI | 平均执行一条指令所需的时钟周期 |
| 3. RLLC | 游程长度受限码 |
| 4. CRC | 循环冗余校验码 |
| 5. CAM | 相联存储器（或：按内容访问的存储器） |
| 6. VHDL | 超高速集成电路硬件描述语言（或：硬件描述语言） |
| 7. DDR SDRAM | 双倍速率同步动态随机存储器 |
| 8. LRU | 近期最少使用算法 |
| 9. RAID | 独立磁盘冗余阵列（或：廉价磁盘冗余阵列） |
| 10. VLIW | 超长指令字技术 |

五、 问答题（5 小题，每小题 3 分，共 15 分；在答题纸填写答案时请写上每
小题的对应编号）

1. （3 分）取指周期的任务是什么？间指周期的任务是什么？中断周期的任务是什么？

答：取指周期的任务是从存储器中得到指令的机器码，并送到 **IR** 中。（1 分）

间指周期的任务是由指令的形式地址，从存储器中得到指令的有效地址。（1 分）

中断周期的任务是：保护程序断点；中断服务程序入口地址送 **PC**；硬件关中断。（1 分）

2. （3 分）微指令的编码方式主要有哪几种？微指令序列地址（后续微指令的地址）的形成方法主要有哪几种？

答：微指令的编码方式主要有直接编码（直接控制）方式、字段直接编码方式、字段间接编码方式、混合编码方式等。（1.5 分）

微指令序列地址（后续微指令的地址）的形成方法主要有直接由微指令的下地址字段指出、根据机器指令的操作码形成等。（1.5 分）

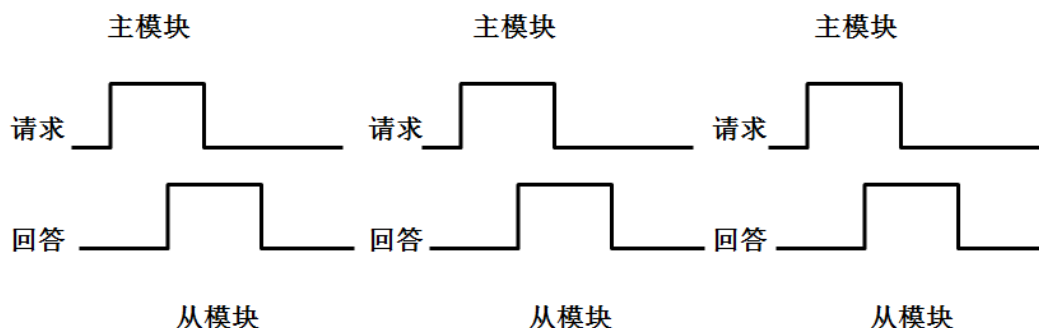
3. （3 分）**DRAM** 为什么需要刷新（再生）？通常有哪几种刷新方式？哪种刷新方式存在“死时间”？哪种刷新方式不存在“死时间”？哪种刷新方式可以缩短“死时间”？

答：由于 **DRAM** 采用电容存储电荷的原理来寄存信息，而电容上的电荷一般只能维持 1-2ms。因此必须在 2ms 内对 **DRAM** 所有存储单元恢复一次原状态，这个过程称为再生或刷新。（1 分）

通常有三种刷新方式：集中刷新、分散刷新、异步刷新。（1 分）

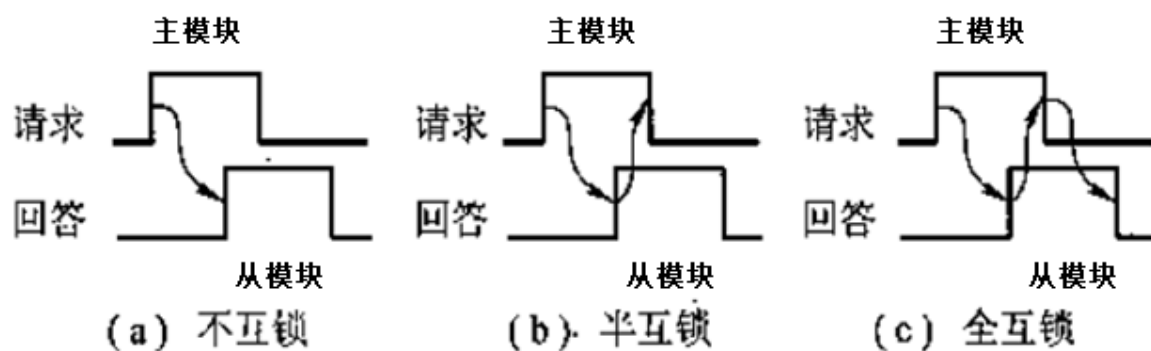
集中刷新存在“死时间”，分散刷新不存在“死时间”，异步刷新可以缩短“死时间”。（1 分）

4. （3 分）异步通信的应答方式分为哪三种类型？请在下面的图中画出三种类型主模块请求与从模块回答的关系（需同时在图上标明应答方式的类型）。



答：（1）不互锁、半互锁、全互锁 （1 分）

(2) 三种类型主模块请求与从模块回答的关系 (2分)



异步通信中请求与回答的互锁

5. (3分) 设某计算机有4个中断源A、B、C、D，其硬件排队优先次序为A → B → C → D。现要求将4个中断源的处理次序改为C → A → D → B。

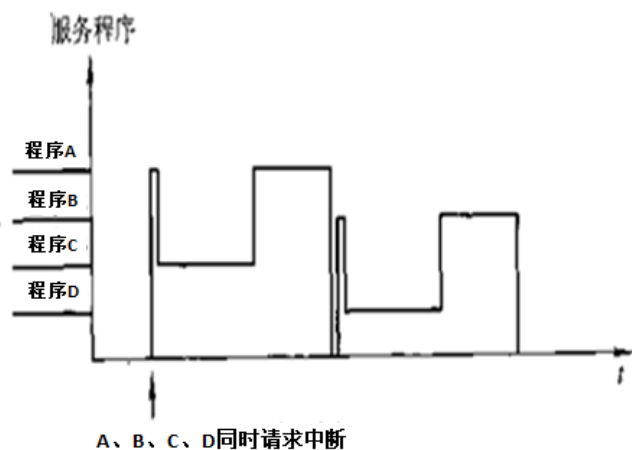
(1) 请写出4个中断源的屏蔽字。

(2) 若4个中断源同时有中断请求，请画出CPU执行程序的轨迹。

答: (1) (1分)

中断源	屏蔽字			
	A	B	C	D
A	1	1	0	1
B	0	1	0	0
C	1	1	1	1
D	0	1	0	1

(2) (2分)



CPU执行程序的轨迹

六、 设计题（共 40 分。其中 1-3 小题是选做题，请任选 2 小题；第 4 小题是必做题。在答题纸填写答案时请写上每小题的对应编号）

1. （12 分）某 Cache-主存层次结构中，主存容量为 1MB，Cache 容量为 4KB，块长为 4 个字，每个字为 32 位，访存地址为字节地址。要求：

（1）分别画出直接映射方式、全相联映射方式、四路组相联映射方式下，主存的地址格式和 Cache 的地址格式。

答：

①直接映射方式： （2 分）

主存地址 = 8 位（主存字块标志） + 8 位（Cache 字块地址） + 4 位（块内地址）

Cache 地址 = 8 位（Cache 字块地址） + 4 位（块内地址）

②全相联映射方式： （2 分）

主存地址 = 16 位（主存字块标志） + 4 位（块内地址）

Cache 地址 = 8 位（Cache 字块地址） + 4 位（块内地址）

③四路组相联映射方式： （2 分）

主存地址 = 10 位（主存字块标志） + 6 位（Cache 组地址） + 4 位（块内地址）

Cache 地址 = 6 位（Cache 组地址） + 2（Cache 字块地址） + 4 位（块内地址）

（2）直接映射方式下，地址为 12345H（十六进制）的主存单元，映射到 Cache 的哪个块？如果该 Cache 块的主存字块标志及有效位为以下三种情况，哪种情况 Cache 命中了？

第 1 种情况：主存字块标志 = 00010010 有效位 = 1

第 2 种情况：主存字块标志 = 00100001 有效位 = 1

第 3 种情况：主存字块标志 = 00010010 有效位 = 0

答：映射到 Cache 的 34H 字块。第 1 种情况 Cache 命中了。 （2 分）

（3）四路组相联方式下，地址为 6789AH（十六进制）的主存单元，映射到 Cache 的哪个组？如果该 Cache 组的四个主存字块标志及有效位为以下三种情况，哪种情况 Cache 命中了？如果命中，是命中到哪一个字块？

第 1 种情况：

字块号	主存字块标志	有效位
字块 0	1110011100	1
字块 1	1110011110	1
字块 2	1110011110	1
字块 3	1110011111	1

第 2 种情况：

字块号	主存字块标志	有效位
字块 0	0110011100	0

字块 1	0110011101	0
字块 2	0110011110	0
字块 3	0110011111	0

第 3 种情况:

字块号	主存字块标志	有效位
字块 0	0110011100	1
字块 1	0110011101	1
字块 2	0110011110	1
字块 3	0110011111	1

答: 映射到 Cache 的 09H 组。第 3 种情况 Cache 命中了, 命中到第 2 个字块。 (4 分)

2. (12 分) 某 CPU 有 20 根地址线、16 根数据线, M/-IO 为访问存储器与访问 I/O 的控制线 (高电平访问存储器, 低电平访问 I/O), -RD 为读控制信号 (低电平有效), -WR 为写控制信号 (低电平有效); CPU 通过 BHE 和 A₀ 来实现按字节或字 (1 个字=16 位) 两种形式访问存储器 (如下表所示)。现有 32KB ROM、64KB ROM、32KB RAM、64KB RAM 存储器芯片, 以及 74LS138 译码器、各种门电路芯片。请问:

(1) CPU 按字节访问存储器的地址范围是多少? CPU 按字访问存储器的地址范围是多少?

答: CPU 按字节访问存储器的地址范围是 1M, 按字访问的存储器地址范围是 512K。 (2 分)

(2) 画出 CPU 与存储器的连接图。要求最小 64KB 为用户存储区, 相邻 64KB 为系统程序区; CPU 既可以按字访问存储器, 也可以按字节访问存储器, 并且按字节访问存储器时, 需区分奇偶体。

答: (8 分)

用户程序区选用 2 片 32KB 的 RAM, 系统程序区选用 2 片 32KB 的 ROM

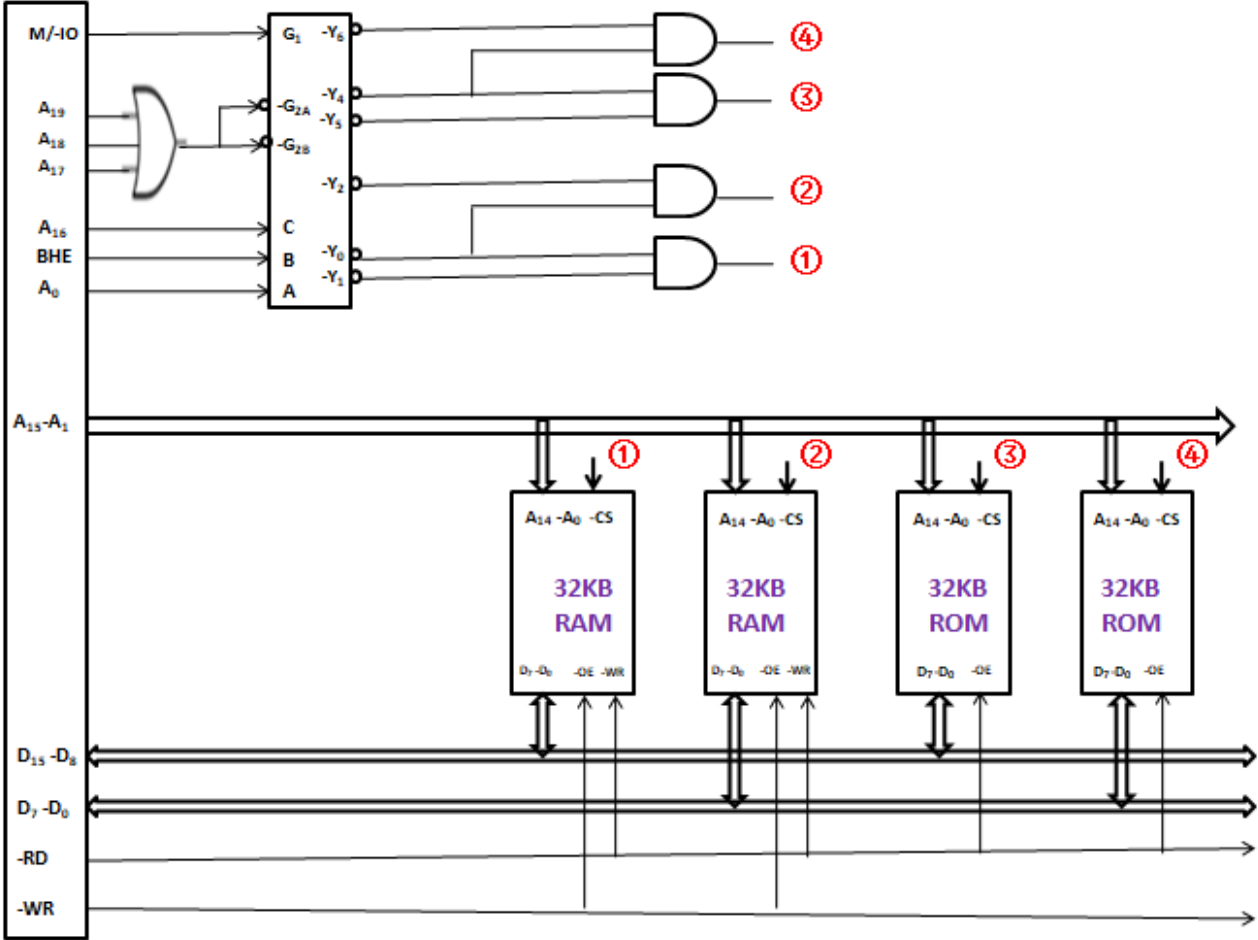
用户程序区的地址: 00000H – 0FFFFH

A19-A16	A15-A12	A11-A8	A7-A4	A3-A0	
0000	0000	0000	0000	0000	2 片 32KB RAM
0000	1111	1111	1111	1111	

系统程序区的地址: 10000H – 1FFFFH

A19-A16	A15-A12	A11-A8	A7-A4	A3-A0	
0001	0000	0000	0000	0000	2 片 32KB ROM
0001	1111	1111	1111	1111	

图如下所示：



(3) 分别写出 CPU 按字访问和按字节访问时，上图（第（2）步画出的图）中每个存储芯片的地址范围（用十六进制表示）。

CPU 访问形式与 BHE 和 A₀ 的关系

BHE	A ₀	访问形式
0	0	字
0	1	奇字节
1	0	偶字节
1	1	不访问

答：按字访问：

(1 分)

- ROM1: 00000H、00002H、00004H、.....、0FFFEH
- ROM2: 00000H、00002H、00004H、.....、0FFFEH
- RAM1: 10000H、10002H、10004H、.....、1FFFEH
- RAM2: 10000H、10002H、10004H、.....、1FFFEH

按字节访问:

(1 分)

ROM1: 00001H、00003H、00005H、.....、0FFFFH

ROM2: 00000H、00002H、00004H、.....、0FFFEH

RAM1: 10001H、10003H、10005H、.....、1FFFFH

RAM2: 10000H、10002H、10004H、.....、1FFFEH

3. (12 分) 已知 $X = 3.3125$, $Y = 6.125$, 请采用浮点加减法运算的方法, 完成 “ $X - Y = ?$ ” 的运算。要求: 先将 X 和 Y 用规格化的浮点数表示, 再进行浮点数的加减法运算, 运算结果也要用规格化浮点数表示。浮点数的格式为: 阶码为 5 位 (含 1 位符号位), 尾数为 11 位 (含 1 位符号位); 阶码采用移码表示, 尾数采用补码表示。

答:

先将 X 、 Y 、 $-Y$ 用规格化浮点数表示

$$X = 3.3125 = 11.0101 = 0.110101 \times 2^{10}$$

X 的浮点数规格化表示 = 1,0010 0.11 0101 0000 (2 分)

$$Y = 6.125 = 110.001 = 0.110001 \times 2^{11}$$

Y 的浮点数规格化表示 = 1,0011 0.11 0001 0000 (2 分)

$-Y$ 的浮点数规格化表示 = 1,0011 1.00 1111 0000 (2 分)

(1) 对阶: (2 分)

$$X = 1,0011 \quad 0.01 \ 1010 \ 1000$$

(2) 尾数求和: (2 分)

$$X - Y \text{ 的尾数} = 0.01 \ 1010 \ 1000 + 1.00 \ 1111 \ 0000 = 1.10 \ 1001 \ 1000$$

(3) 尾数规格化: (2 分)

$X - Y$ 的尾数为非规格化数 (尾数绝对值小于 0.5), 需要左规, 阶码减 1

$$X - Y = 1,0010 \quad 1.01 \ 0011 \ 0000$$

验证: (这一步不要求学生做)

$$X - Y = 1,0010 \quad 1.01 \ 0011 \ 0000 = -0.101101 \times 2^2 = -2.8125 = 3.3125 - 6.125$$

4. (16 分) 已知 TD-CMA 实验系统在简单模型机方式下的数据通路图如图 1 所示, ALU 的逻辑功能表如表 1 所示。该简单模型机有 5 条指令, 指令的助记符、机器码和说明如下:

助记符	指令机器码	说明
IN R0	0010 0000	IN -> R0

ADD R0, R0	0000 0000	R0 + R0 -> R0
OUT R0	0011 0000	R0 -> OUT
JMP addr	1110 0000 *****	addr -> PC
HLT	0101 0000	停机

该简单模型机的微指令格式如表 2 所示，5 条指令的微程序流程图如图 2 所示，微指令的代码如表 3 所示。请问：

(1) 该简单模型机的微指令采用什么编码方式（控制方式）？微指令的操作控制字段和顺序控制字段各有几位？

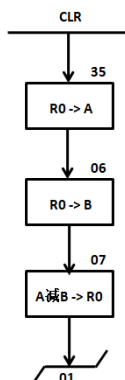
如果采用直接编码（直接控制）方式，该简单模型机微指令的操作控制字段需要多么位（假设最高 2 位 M23、M22 仍然需要保留）？

答：混合编码方式 (2 分)
18 位、6 位
20 位

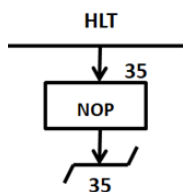
(2) 要求设计一条新的指令，代替原有 5 条指令中的“HLT”指令。该新指令完成将 R0 置为 0 的功能，即执行该指令后，R0=0，其助记符为“CLR R0”。请对图 2 的微程序流程图进行修改，对表 3 的微指令代码表进行修改，使该简单模型机具有新指令“CLR R0”的功能（代替原有的“HLT”指令）。注：只需给出图 2 的修改部分，表 3 的修改部分。

提示：可通过“R0 减 R0，结果送 R0”，实现将 R0 置为 0 的功能。

答：用 (2 分)



取代：



将 35 00 00 35 00000 0000 000 000 000 110101
修改为: 35 00 14 06 00000 0000 001 010 000 000110 (2分)

增加: 2 条微指令

06 00 24 07 00000 0000 010 010 000 000111 (2分)

07 05 B2 01 00000 1011 011 001 000 000001 (2分)

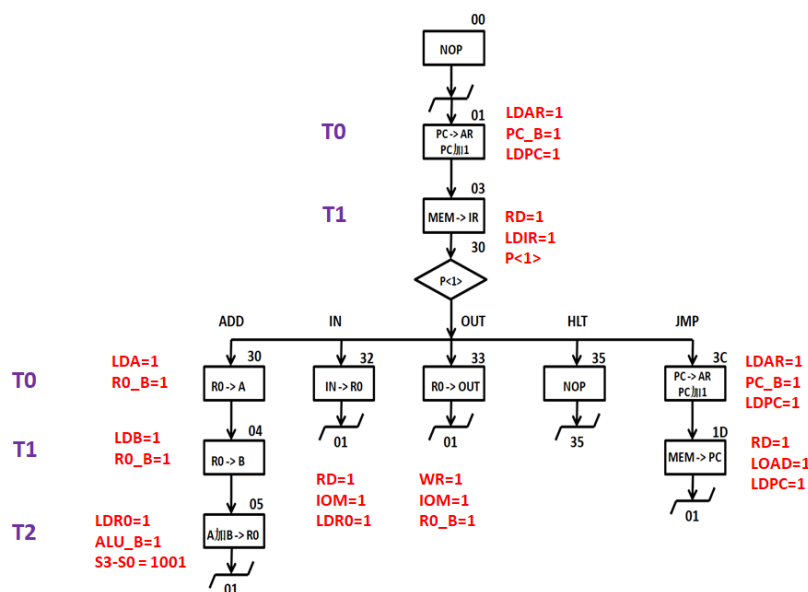
(3) 如果该简单模型机 (仍然是具有 IN、ADD、OUT、JMP、HLT 等 5 条指令的简单模型机) 的控制器采用组合逻辑设计方法进行设计, 请写出下列控制信号 (微操作命令信号) 的最简逻辑表达式:

- ① LDIR
- ② LDPC
- ③ RD

假设取指周期在 T₀、T₁ 节拍内完成 (对应微程序设计地址为 01 和 03 的 2 条微指令); ADD 指令的执行周期在 T₀、T₁、T₂ 节拍内完成 (对应微程序设计地址为 30、04 和 05 的 3 条微指令), IN 指令、OUT 指令、HLT 指令的执行周期都是在 T₀ 节拍内完成 (分别对应微程序设计地址为 32、33 和 35 的 1 条微指令), JMP 指令的执行周期在 T₀、T₁ 节拍内完成 (对应微程序设计地址为 3C 和 1D 的 2 条微指令)。用 FE 表示取指周期, EX 表示执行周期。

提示: 首先列出每条微指令所发出的控制信号, 然后再写出控制信号与取指周期 (FE)、执行周期 (EX)、节拍 (T₀、T₁、T₂)、指令 (ADD、IN、OUT、HLT、JMP) 之间的逻辑关系。

答: 每条微指令所发出的控制信号如下图所示: 其中, 2 条 NOP 微指令不发出任何控制信号。



(1) LDIR 控制信号在所有指令 (5 条指令) 的取指周期的 T₁ 节拍发出, 故有以下逻辑式:

$$LDIR = T_1 * FE * (ADD + IN + OUT + HLT + JMP) \quad (2 \text{ 分})$$

(2) LDPC 控制信号在所有指令（5 条指令）的取指周期的 T0 节拍发出，在 JMP 指令的执行周期的 T0 和 T1 节拍发出，故有以下逻辑式：

$$\text{LDPC} = \text{T0} * \text{FE} * (\text{ADD} + \text{IN} + \text{OUT} + \text{HLT} + \text{JMP}) + (\text{T0} + \text{T1}) * \text{EX} * \text{JMP} \quad (2 \text{ 分})$$

(3) RD 控制信号在所有指令（5 条指令）的取指周期的 T1 节拍发出，在 IN 指令的执行周期的 T0 和 T1 节拍发出，在 JMP 指令的执行周期的 T1 节拍发出，故有以下逻辑式：

$$\text{RD} = \text{T1} * \text{FE} * (\text{ADD} + \text{IN} + \text{OUT} + \text{HLT} + \text{JMP}) + \text{T0} * \text{EX} * \text{IN} + \text{T1} * \text{EX} * \text{JMP} \quad (2 \text{ 分})$$

其他的控制信号如下：（以下不要求学生做！）

$$(4) \text{LDAR} = \text{T0} * \text{FE} * (\text{ADD} + \text{IN} + \text{OUT} + \text{HLT} + \text{JMP}) + \text{T0} * \text{EX} * \text{JMP}$$

$$(5) \text{PC_B} = \text{T0} * \text{FE} * (\text{ADD} + \text{IN} + \text{OUT} + \text{HLT} + \text{JMP}) + \text{T0} * \text{EX} * \text{JMP}$$

$$(6) \text{P<1>} = \text{T1} * \text{FE} * (\text{ADD} + \text{IN} + \text{OUT} + \text{HLT} + \text{JMP})$$

$$(7) \text{LDA} = \text{T0} * \text{EX} * \text{ADD}$$

$$(8) \text{LDB} = \text{T1} * \text{EX} * \text{ADD}$$

$$(9) \text{R0_B} = (\text{T0} + \text{T1}) * \text{EX} * \text{ADD} + \text{T0} * \text{EX} * \text{OUT}$$

$$(10) \text{LDR0} = \text{T2} * \text{EX} * \text{ADD} + \text{T0} * \text{EX} * \text{IN}$$

$$(11) \text{ALU_B} = \text{T2} * \text{EX} * \text{ADD}$$

$$(12) \text{S3-S0} = \text{T2} * \text{EX} * \text{ADD}$$

$$(13) \text{IOM} = \text{T0} * \text{EX} * (\text{IN} + \text{OUT})$$

$$(14) \text{WR} = \text{T0} * \text{EX} * \text{OUT}$$

$$(15) \text{LOAD} = \text{T1} * \text{EX} * \text{JMP}$$

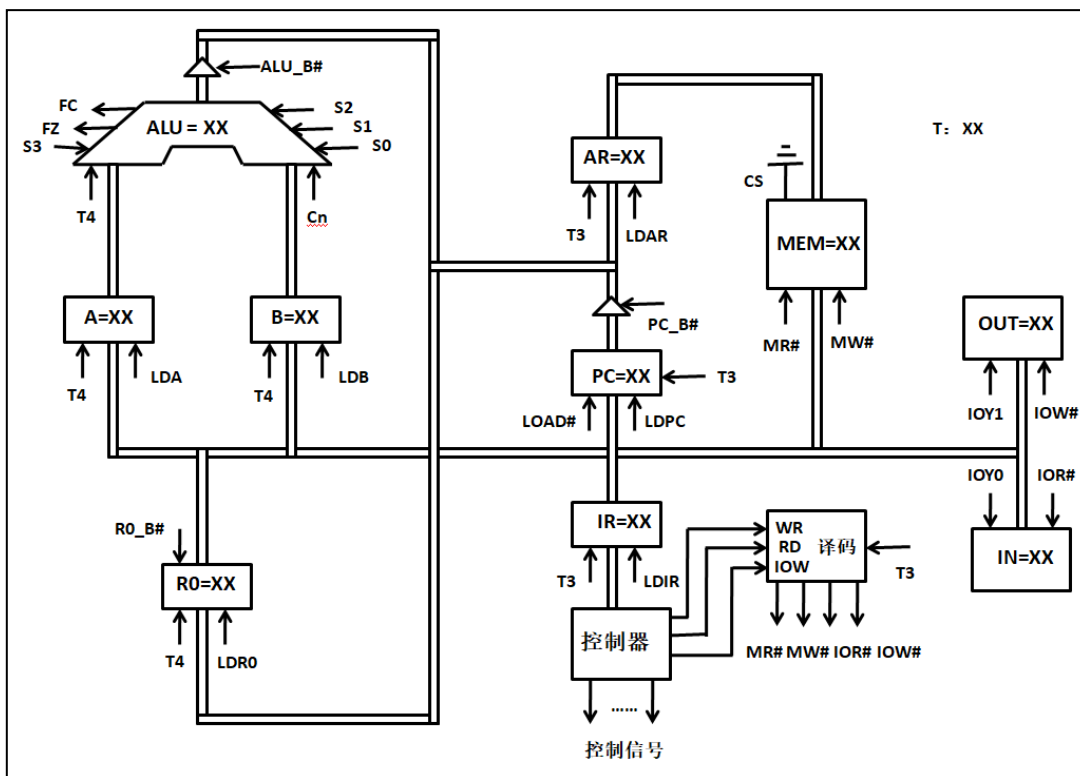


图 1 简单模型机的数据通路图

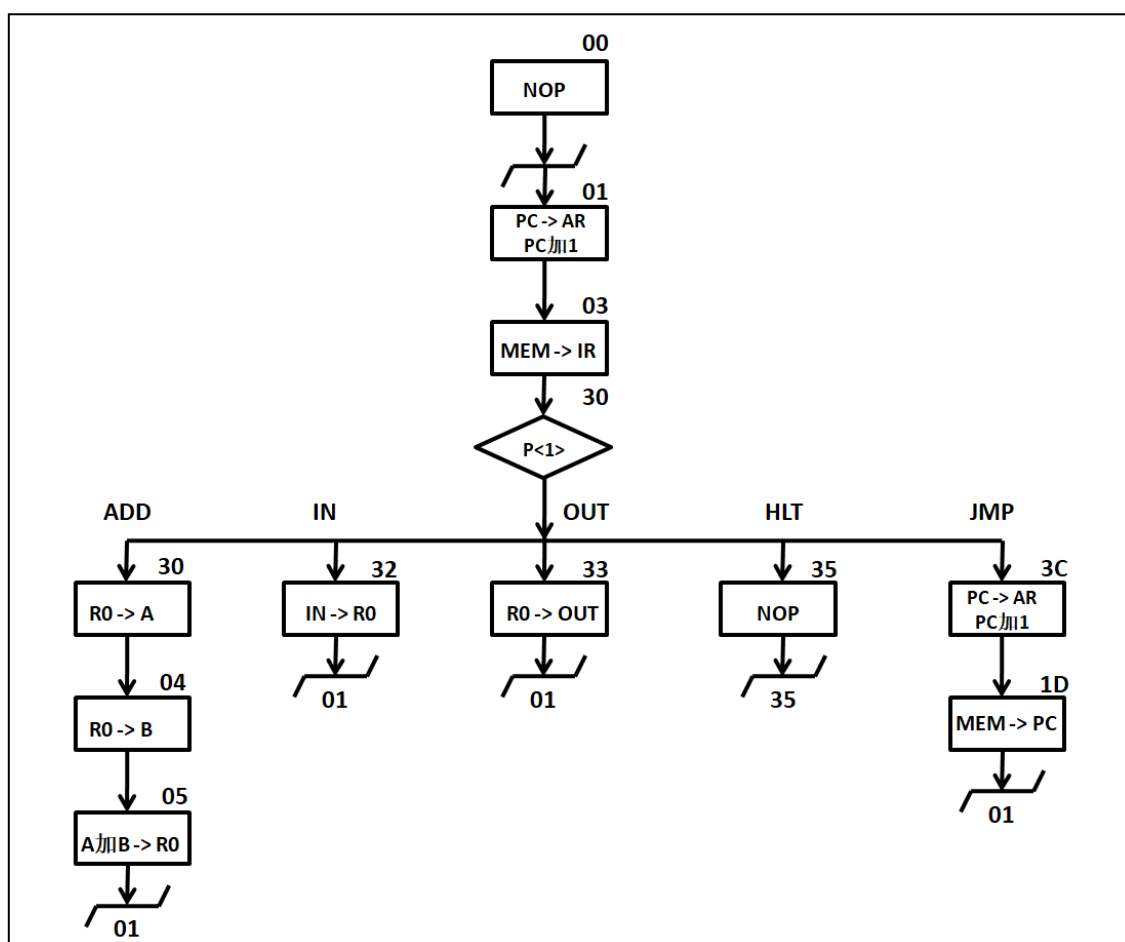


图 2 简单模型机微程序流程图

表 1 简单模型机 ALU 的逻辑功能表

运算类型	S3 S2 S1 S0	CN	功能
逻辑运算	0000	X	F=A (直通)
	0001	X	F=B (直通)
	0010	X	F=AB (FZ)
	0011	X	F=A+B (FZ)
	0100	X	F=/A (FZ)
移位运算	0101	X	F=A 不带进位循环右移 B (取低 3 位) 位 (FZ)
	0110	0	F=A 逻辑右移一位 (FZ)
		1	F=A 带进位循环右移一位 (FC, FZ)
	0111	0	F=A 逻辑左移一位 (FZ)
		1	F=A 带进位循环左移一位 (FC, FZ)
算术运算	1000	X	置 FC=CN (FC)
	1001	X	F=A 加 B (FC, FZ)
	1010	X	F=A 加 B 加 FC (FC, FZ)
	1011	X	F=A 减 B (FC, FZ)
	1100	X	F=A 减 1 (FC, FZ)
	1101	X	F=A 加 1 (FC, FZ)
	1110	X	(保留)
	1111	X	(保留)

*表中“X”为任意态，下同

表 2 简单模型机微指令的格式

23	22	21	20	19	18-15	14-12	11-9	8-6	5-0
M23	M22	WR	RD	IOM	S3-S0	A字段	B字段	C字段	MA5-MA0

A 字段

14	13	12	选择
0	0	0	NOP
0	0	1	LDA
0	1	0	LDB
0	1	1	LDRO
1	0	0	保留
1	0	1	LOAD
1	1	0	LDAR
1	1	1	LDIR

B 字段

11	10	9	选择
0	0	0	NOP
0	0	1	ALU_B
0	1	0	RO_B
0	1	1	保留
1	0	0	保留
1	0	1	保留
1	1	0	PC_B
1	1	1	保留

C 字段

8	7	6	选择
0	0	0	NOP
0	0	1	P<1>
0	1	0	保留
0	1	1	保留
1	0	0	保留
1	0	1	LDPC
1	1	0	保留
1	1	1	保留

表 3 简单模型机微指令的代码

地址	十六进制	高五位	S3-S0	A 字段	B 字段	C 字段	MA5-MA0
00	00 00 01	00000	0000	000	000	000	000001
01	00 6D 43	00000	0000	110	110	101	000011
03	10 70 70	00010	0000	111	000	001	110000
04	00 24 05	00000	0000	010	010	000	000101
05	04 B2 01	00000	1001	011	001	000	000001
1D	10 51 41	00010	0000	101	000	101	000001
30	00 14 04	00000	0000	001	010	000	000100
32	18 30 01	00011	0000	011	000	000	000001
33	28 04 01	00101	0000	000	010	000	000001
35	00 00 35	00000	0000	000	000	000	110101
3C	00 6D 5D	00000	0000	110	110	101	011101