

## 第七章 指令系统

作业：P335 7.1、7.2、7.3、7.4、7.5、7.6、7.7、7.14、7.16、7.19

### 7.1 什么叫机器指令？什么叫指令系统？为什么说指令系统与机器的主要功能以及硬件结构之间存在着密切的关系？

解：机器指令：能够被计算机识别的、完成预设功能的一组二进制机器代码。

指令系统：一台计算机所有指令的集合。

从 CPU 的设计角度看，应该先着眼于实现何种功能，在此前提下完成硬件的设计，再根据硬件结构，设计指令以实现计算机实现的基本功能。而执行一条指令的所有微操作即完成了指令的功能。故指令系统与机器的主要功能以及硬件结构之间存在着密切的关系。

### 7.2 什么叫寻址方式？为什么要学习寻址方式？

解：寻址方式分为指令寻址和数据寻址两类。

指令寻址指的是寻找下一条要执行的存放指令的所在位置。这主要用于转移指令和子程序调用指令。

数据寻址指的是寻找操作数的有效地址。

学习寻址方式是能够掌握一条指令完整的执行过程，以及掌握计算机运行中指令流和数据流的走向，有助于掌握计算机组成的基本原理。

### 7.3 什么叫指令字长、机器字长和存储字长？

解：指令字长：机器指令中二进制代码的总位数。（不同的指令长度可以不同）。

机器字长：计算机进行一次（定点）整数运算所能处理的二进制数据的位数。

存储字长：一个存储单元存储一串二进制代码的位数。

### 7.4 零地址指令的操作数来自哪里？各举一例说明。

答：零地址指令的操作数来自 ACC，为隐含约定。

在一地址指令中，另一个操作数的地址通常可采用 ACC 隐含寻址方式获得。

### 7.5 对于二地址指令而言，操作数的物理地址可安排在什么地方？举例说明。

答：对于二地址指令而言，操作数的物理地址可安排在寄存器内、指令中或内存单元内等。

### 7.6 某指令系统字长 16 位，地址码 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。（换一种做法）

解：

三地址指令：一个地址码 4 位，三地址即 12 位，则操作码字段 4 位。4 位操作码可有如下组合可以选用：0000~1111，共 16 种。由于有 8 条三地址指令，在此我们选择三地址指令的操作码为：0000~0111。这样，操作码字段中，最高位“0”已被使用，只剩下“1”可用。

二地址指令：二个地址码占 8 位，操作码字段 8 位，则 8 位操作码可有如下组合可以选用：10000000~11111111，共 128 种。由于有 16 条二地址指令，在此我们选择二地址指令的操作码为：10000000~11111000，这样，操作码字段中，高 5 位“11111”已被使用。未被使用的高 8 位为：11111000~11111111。

一地址指令：一个地址码占 4 位，操作码字段 12 位。则 12 位操作码可有如下组合可以选用：  
 $111110000000 \sim 111111111111$ ，共 128 种组合，由于有 100 条一地址指令，在此我们选择一地址指令的操作码为： $111110000000 \sim 111111000111$ 。这样，在 12 位操作码字段中，其高 7 位为“1111111”和低 2 位为“00011”已被使用。未被使用的高 12 位组合为： $11111100100 \sim 111111111111$ ，只剩下 8 种组合。

**7.7 设指令字长为 16 位，采用扩展操作码技术，每个操作数的地址为 6 位。如果定义了 13 条二地址指令，试问还可以安排多少条一地址指令？**

解：

解法一：

扩展操作码技术，意味着操作码长度可变，指令字长 16 位，操作数的地址 6 位。则二地址指令中，2 个操作数的地址占 12 位，余下 4 位作为操作码。指令格式如下：

OP (4 位)	A1 (6 位)	A2 (6 位)
0000	A1	A2
0001	A1	A2
...		
1100	A1	A2

高 4 位 1101、1110、1111 可扩展。

一地址指令格式如下：

OP (12 位)	A1 (6 位)
-----------	----------

操作码可有如下种类：

1101, 0000, 00~1101, 1111, 11; 共 64 条。

1110, 0000, 00~1101, 1111, 11; 共 64 条。

1111, 0000, 00~1101, 1111, 11; 共 64 条。

还可以安排 192 条一地址指令。

解法二：

当操作码占 4 位时，最多 16 种组合，意味着二地址指令最多 16 条。而实际上二地址指令为 13 条，高四位还剩下 3 种组合，这 3 种组合与后续 6 位构成一地址指令的操作码，故还可以安排多少条一地址指令为：

$$3 * 2^6 = 3 * 64 = 192 \text{ 条。}$$

**7.8 某机指令字长 16 位，每个操作数的地址码为 6 位，设操作码长度固定，指令分为零地址、一地址和二地址三种格式。若零地址指令有 M 条，一地址指令有 N 种，则二地址指令最多有几种？若操作码位数可变，则二地址指令最多允许有几种？**

解：1) 若采用定长操作码时，二地址指令格式如下：

OP (4 位)	A1 (6 位)	A2 (6 位)
----------	----------	----------

设二地址指令有 K 种，则： $K = 2^4 - M - N$

当 M=1 (最小值)，N=1 (最小值) 时，二地址指令最多有： $K_{\max} = 16 - 1 - 1 = 14$  种

2) 若采用变长操作码时，二地址指令格式仍如 1) 所示，但操作码长度可随地址码的个数而变。此时， $K = 2^4 - (N/2^6 + M/2^{12})$ ；

当  $(N/2^6 + M/2^{12}) \leq 1$  时 ( $N/2^6 + M/2^{12}$  向上取整), K 最大, 则二地址指令最多有:  
 $K_{\max} = 16 - 1 = 15$  种 (只留一种编码作扩展标志用。)

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:

$$EA = [(IX) + A], IX \rightarrow (IX) + 1$$

2) 先间址再变址寻址过程简单示意如下:  $EA = (IX) + (A), IX \rightarrow (IX) + 1$

7.14 设相对寻址的转移指令占 2 个字节, 第一个字节是操作码, 第二个字节是相对位移量, 用补码表示。假设当前转移指令第一个字节所在地址是 2000H 且 CPU 每取出一个字节便自动完成(PC)+1 的操作。试问: 当执行“JMP \* +8”和“JMP \* -9”指令时, 转移指令第二个字节的内容各是多少?

解:

依题意, 转移指令的操作码为 8 位, 补码表示的位移量为 8 位。当前转移指令第一个字节所在地址是 2000H, 取出一个字节便自动完成 (PC) +1, 则本指令取完后, PC=2002H。

“JMP \* A”执行后, 应跳至“本指令下条指令所在地址 (本指令取完后 PC 的值) +位移量, 再送至 PC”, 即 (PC) +2+位移量的位置。

执行“JMP \* +8”指令时, “+8”是 8 位补码表示的位移量, 故该位移量为: 0000, 1000。  
 当指令“JMP \* +8”执行结束时, PC=2002H+8=200AH。

执行“JMP \* -9”指令时, “-9”是 8 位补码表示的位移量, 故该位移量为: 1111, 0111。  
 当指令“JMP \* -9”执行结束时, PC=2002H+(-9)=1FF9H。

7.16 某机主存容量为 4M×16 位, 且存储字长等于指令字长, 若该机指令系统可完成 108 种操作, 操作码位数固定, 且具有直接、间接、变址、基址、相对、立即等六种寻址方式, 试回答: (1) 画出一地址指令格式并指出各字段的作用;

(2) 该指令直接寻址的最大范围;

(3) 一次间址和多次间址的寻址范围;

(4) 立即数的范围 (十进制表示);

(5) 相对寻址的位移量 (十进制表示);

(6) 上述六种寻址方式的指令哪一种执行时间最短? 哪一种最长? 为什么? 哪一种便于程序浮动? 哪一种最适合处理数组问题?

(7) 如何修改指令格式, 使指令的寻址范围可扩大到 4M?

(8) 为使一条转移指令能转移到主存的任一位置, 可采取什么措施? 简要说明之。

解: (1) 单字长一地址指令格式:

OP (7 位)	M (3 位)	A (6 位)
----------	---------	---------

OP 为操作码字段, 共 7 位, 可反映 108 种操作;

M 为寻址方式字段, 共 3 位, 可反映 6 种寻址操作;

A 为地址码字段, 共 16-7-3=6 位。

(2) 直接寻址的最大范围为  $2^6=64$ 。

(3) 由于存储字长为 16 位, 故一次间址的寻址范围为  $2^{16}$ ; 若多次间址, 需用存储字的最高位

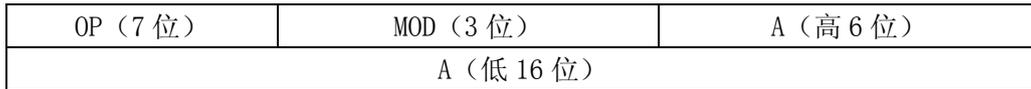
来区别是否继续间接寻址，故寻址范围为  $2^{15}$ 。

(4) 立即数的范围为-32——31（有符号数），或 0——63（无符号数）。

(5) 相对寻址的位移量为-32——31。

(6) 上述六种寻址方式中，因立即数由指令直接给出，故立即寻址的指令执行时间最短。间接寻址在指令的执行阶段要多次访存（一次间接寻址要两次访存，多次间接寻址要多次访存），故执行时间最长。变址寻址由于变址寄存器的内容由用户给定，而且在程序的执行过程中允许用户修改，而其形式地址始终不变，故变址寻址的指令便于用户编制处理数组问题的程序。相对寻址操作数的有效地址只与当前指令地址相差一定的位移量，与直接寻址相比，更有利于程序浮动。

(7) **方案一**：为使指令寻址范围可扩大到 4M，需要有效地址 22 位，此时可将单字长一地址指令的格式改为双字长，如下图所示：



**方案二**：如果仍采用单字长指令（16 位）格式，为使指令寻址范围扩大到 4M，可通过段寻址方案实现。安排如下：

硬件设段寄存器 DS（16 位），用来存放段地址。在完成指令寻址方式所规定的寻址操作后，得到有效地址 EA（6 位），再由硬件自动完成段寻址，最后得 22 位物理地址。即：物理地址 = (DS) × 2<sup>6</sup> + EA

注：段寻址方式由硬件隐含实现。在编程指定的寻址过程完成、EA 产生之后由硬件自动完成，对用户是透明的。

**方案三**：在采用单字长指令（16 位）格式时，还可通过页面寻址方案使指令寻址范围扩大到 4M。安排如下：

硬件设页面寄存器 PR（16 位），用来存放页面地址。指令寻址方式中增设页面寻址。当需要使指令寻址范围扩大到 4M 时，编程选择页面寻址方式，则：EA = (PR) || A（有效地址 = 页面地址“拼接”6 位形式地址），这样得到 22 位有效地址。

(8) 为使一条转移指令能转移到主存的任一位置，寻址范围须达到 4M，除了采用(7) 方案一中的双字长一地址指令的格式外，还可配置 22 位的基址寄存器或 22 位的变址寄存器，使 EA = (BR) + A（BR 为 22 位的基址寄存器）或 EA = (IX) + A（IX 为 22 位的变址寄存器），便可访问 4M 存储空间。还可以通过 16 位的基址寄存器左移 6 位再和形式地址 A 相加，也可达到同样的效果。

总之，不论采取何种方式，最终得到的实际地址应是 22 位。

### 7.19 CPU 内有 32 个 32 位的通用寄存器，设计一个能容纳 64 种操作的指令系统。假设指令字长等于机器字长，试回答以下问题：

(1) 如果主存可直接或间接寻址，采用寄存器-存储器型指令，能直接寻址的最大存储空间是多少？画出指令格式并说明各字段的含义。

(2) 在满足(1)的前提下，如果采用通用寄存器作为基址寄存器，则上述寄存器-存储器型指令的格式有何特点？画出指令格式并指出这类指令可访问多大的存储空间？

解：

已知条件如下：

64 种操作的指令系统，操作码需要 6 位即可。

32 个 32 位的通用寄存器，说明寄存器编号需要 5 位，且机器字长为 32 位。由于指令字长等于机器字长，则指令字长为 32 位。

(1) 寄存器-存储器型指令，说明一个地址码可为寄存器，另一个地址码为存储器。而根据题意，地址码可为寄存器时只有寄存器寻址，地址码为存储器时可有直接寻址和间接寻址。

故有 2 种寻址方式：存储器时直接寻址和间接寻址，寻址特征 M 需要 1 位。

故指令格式如下：

OP (6 位)	Ri (5 位)	M	A (20 位)
----------	----------	---	----------

形式地址 A 为 20 位，直接寻址时，形式地址即为有效地址。故其最大可寻址范围为： $2^{20}=1M$  字。

(2) 寄存器-存储器型指令，如果采用通用寄存器作为基址寄存器，寻址特征 M 需要 2 位。指令格式如下：

OP (6 位)	Ri (5 位)	M	Rj (5 位)	A (14 位)
----------	----------	---	----------	----------

寄存器间接寻址和基址寻址的有效地址码都是 32 位，故其最大可寻址范围为： $2^{32}=4G$  字。

**P301 例题** 指令字长 16 位，其中 4 位为基本操作码字段 OP，另有 3 个 4 位地址码字段 A1、A2、A3。设计形成：三地址指令 15 条，二地址指令 12 条，一地址指令 31 条，零地址指令 16 条，共 74 条。

解：

OP	A1	A2	A3	
0000	A1	A2	A3	三地址指令 15 条
0001	A1	A2	A3	
...				
1110	A1	A2	A3	
1111	0000	A1	A2	二地址指令 12 条
1111	0001	A1	A2	
...				
1111	1011	A1	A2	
1111	1100	0000	A1	一地址指令 16 条
1111	1100	0001	A1	
...				
1111	1100	1111	A1	
1111	1101	0000	A1	一地址指令 15 条
1111	1101	0001	A1	
...				
1111	1101	1110	A1	
1111	1110	0000	0000	零地址指令 16 条
1111	1110	0000	0001	
...				
1111	1110	0000	1111	

P302 例 7.1 假设指令字长 16 位，操作数的地址码 6 位，指令有零地址、一地址、二地址三种格式。

- (1) 设操作码固定，若零地址指令有 P 种，一地址指令有 Q 种，则二地址指令最多有几种？
- (2) 采用扩展操作码技术，若二地址指令有 X 种，零地址指令有 Y 种，则一地址指令最多有几种？

解：

(1) 由于操作数地址码为 6 位，则二地址指令的操作码位数为： $16-6-6=4$  位。而操作码固定，意味着操作码只有 4 位。最多可有  $2^4=16$  种操作，即最多可有 16 条指令。因零地址指令有 P 种，一地址指令有 Q 种，则二地址指令最多有： $16-P-Q$  种。

(2) 采用扩展操作码技术，操作码位数可变，则二地址指令、一地址指令和零地址指令操作码的位数分别为：4 位、10 位和 16 位。

因二地址指令（使用高 4 位作为操作码）有 X 种，则高 4 位未用的组合为  $2^4-X$  种，这些未用的组合才能作为一地址和零地址指令操作码的高 4 位。

而二地址指令操作码每减少 1 种，就能够多构成  $2^6$  种一地址指令操作码；同理，一地址指令操作码每减少 1 种，就能够多构成  $2^6$  种零地址指令操作码。

则一地址指令最多有  $(2^4-X) * 2^6$  种（这些种如果都被使用，则零地址指令就没有了）。设一地址指令 M 种，则零地址指令最多有  $Y=[(2^4-X) * 2^6-M] * 2^6$ ，则  $M=(2^4-X) * 2^6-Y/2^6$ 。

P318 例 7.2 设相对寻址的转移指令占 3 个字节，第一个字节为操作码，第二、第三字节为相对位移量（补码表示），而且数据在存储器中采用以低（位数据）字节地址为字地址存放方式。每当 CPU 从存储器取出一个字节时，即自动完成  $(PC) + 1$ 。

- (1) 若 PC 当前值为 240（十进制），要求转移到 290（十进制），则转移指令的第二、第三字节的机器代码是什么？
- (2) 若 PC 当前值为 240（十进制），要求转移到 200（十进制），则转移指令的第二、第三字节的机器代码是什么？

解：

依据“数据在存储器中采用以低（位数据）字节地址为字地址存放方式”题意，可知：指令中的形式地址采用小端模式存放。

(1) PC 当前值指向的是本条相对寻址的转移指令，取完本指令后， $(PC)=243$ 。若要跳转到 290 单元，其位移量应为： $290-243=47$ ，其补码为 002FH。题意为：以低字节地址为字地址存放方式，即采用小端模式存储方式。因此，该指令的第二、第三字节的机器代码分别是：2FH 和 00H。（详见 P306 的图 7.4 及其解释）

(2) 取完本指令后， $(PC)=243$ 。若要跳转到 200 单元，其位移量应为： $200-243=-43$ ，其补码为 FFD5H，同理，则该指令的第二、第三字节的机器代码分别是：D5H 和 FFH。

P319 例 7.3 一条双字长直接寻址的子程序调用指令，其第一个字为操作码和寻址特征，第二个字为地址码 5000H。假设 PC 当前值为 2000H，SP 的内容为 0100H，栈顶内容为 2746H，存储器按字节编址，而且进栈操作是先执行  $(SP) - \Delta$ ，后存入数据。试回答下列几种情况下，PC、SP 及栈顶内容各是多少？

- (1) CALL 指令被读取前。
- (2) CALL 指令被执行后。
- (3) 子程序返回后。

解：

根据题意，本条要执行的指令在取出之前机器的状态为：PC=2000H，SP=0100H，栈顶内容仍为2746H。

- (1) 从“第二个字为地址码 5000H”可以看出，字长为 2 个字节，故子程序调用指令为 4 个字节。PC 的当前值指向的是本条双字长直接寻址的子程序调用指令，CALL 指令被读取前：PC=2000H，SP=0100H，栈顶内容仍为 2746H。
- (2) 因子程序调用指令为 4 个字节，取完本指令后，PC=2004H。因存储器按字节编址，说明 $\Delta=1$ 。子程序调用指令执行中，断点地址应该压栈，即将 PC 的内容（即 2004H）压栈，则 SP=0100H-2=00FEH，栈顶内容应为 2004H（题中没有说明先压高字节还是低字节，故只能这样说明）。而本条子程序调用指令为双字长直接寻址，说明地址码 5000H 即为子程序入口地址，PC=5000H。
- (3) PC=2004H，SP=0100H，栈顶内容为 2746H。

P323 例 7.4 某机器字长 16 位，存储器直接寻址空间为 128 字，变址时的位移量为-64~+63, 16 个通用寄存器均可作为变址寄存器。设计一套指令系统格式，满足下列寻址类型的要求。

- (1) 直接寻址的二地址指令 3 条。
- (2) 变址寻址的一地址指令 6 条。
- (3) 寄存器寻址的二地址指令 8 条。
- (4) 直接寻址的一地址指令 12 条。
- (5) 零地址指令 32 条。

试问：还有多少种代码未用？以上各种指令均具备的前提下，若安排寄存器寻址的一地址指令，还能容纳多少条？

解：已知条件如下：

根据“机器字长 16 位，存储器直接寻址空间为 128 字”可知：按字寻址

根据“存储器直接寻址空间为 128 字”可知：存储器直接寻址的形式地址码占 7 位

根据“变址时的位移量为-64~+63”可知：位移量占 7 位

根据“16 个通用寄存器均可作为变址寄存器”可知：通用寄存器可用 4 位表示具体的寄存器

- (1) “存储器直接寻址空间为 128 字”说明，地址码为 7 位。机器字长 16 位，说明指令最短应 16 位，因此，该机器的存储器也应该按字寻址。

对于二地址指令：每个地址码 7 位，则操作码只有 2 位。即：

OP(2 位)	A1 (7 位)	A2 (7 位)
---------	----------	----------

由于直接寻址的二地址指令 3 条，故 OP 可为：00、01、10（高两位只剩下 11 未用）。

- (2) 由于“16 个通用寄存器均可作为变址寄存器”，故变址寻址的一地址指令中：变址寄存器（作为一个地址）编号 4 位。而“变址时的位移量为-64~+63”，则位移量 7 位。因此，操作码 5 位（正常情况下，由于 16 个通用寄存器均可作为变址寄存器，变址寄存器应采用“显式”，即应该再有一个“寻址特征”位，但本题中没有存在寄存器-存储器型，即 R-M 型指令，故可认为只有这样一类指令，且操作码为 5 位。否则操作码只有 4 位，无法解决本例题）。

即：

OP (5 位)	Rx (4 位)	A (7 位)
----------	----------	---------

由于变址寻址的一地址指令 6 条，故 OP 可为：11000、11001、11010、11011、11100、11101（高五位只剩下 11110 和 11111 未用）。

(3) 寄存器寻址的二地址指令：指的是两个地址均为寄存器寻址，有 16 个通用寄存器，故可用 4 位作为寄存器编号。这样，操作码可为 8 位。即：

OP (8 位)	Ri (4 位)	Rj (4 位)
----------	----------	----------

由于寄存器寻址的二地址指令 8 条，故 OP 可为：11110,000、11110,001、11110,010、11110,011、11110,100、11110,101、11110110、11110,111（高五位只剩下 11111 未用）。

(4) “直接寻址的一地址指令 12 条”说明：地址码 7 位，操作码 9 位。即：

OP (9 位)	A (7 位)
----------	---------

11111,0011、11111,0100、11111,0101、11111,0110、11111,0111、11111,1000、11111,1001、11111,1010、11111,1011（高七位只剩下 1111111 未用）。

(5) “零地址指令 32 条”意味着：操作码可为 16 位。

OP (16 位)
-----------

由于高七位只剩下 1111111 组合（共 7 位）未用，低位操作码有 9 位。故还有  $2^9 - 32 = 480$  种代码未用。

32 条零地址指令操作码（注意：是操作码不是地址码）可为：

1111111,0000,00000~1111111,0000,11111

若安排寄存器寻址的一地址指令（16 个通用寄存器），需要 4 位表示寄存器号即可，具体格式如下：

OP (12 位)	Ri (4 位)
-----------	----------

除去末位 4 位作为寄存器编号外，操作码 12 位。

因零地址指令的高十二位为 1111111,0000,0~1111111,0000,1，高十二位只剩下：1111111,0001,0~1111111,1111,1 组合未用（共 30 条），故操作码正好剩余 30 种，还能容纳 30 条。

**P324 例 7.5** 设某机器配有基址寄存器和变址寄存器，采用一地址指令格式的指令系统，允许直接和间接寻址，且指令字长、机器字长和存储字长均为 16 位。

(1) 若采用单字长指令，共能完成 105 种操作，则指令可直接寻址的范围是多少？一次间接寻址的寻址范围是多少？画出其指令格式并说明各字段的含义。

(2) 若存储字长不变，可采用什么方法直接访问容量为 16MB 的主存？

解：通常，寻址方式有两种实现方法：一种是在 OP 中直接确定，另一种是用寻址特征位确定，而后一种是指令设计最常用的方法，本题用寻址特征位确定寻址方式。

(1) 单字长指令，共能完成 105 种操作，因  $2^6 = 64 < 105 < 2^7 = 128$ ，故操作码需要 7 位。

有基址寄存器和变址寄存器，说明寻址方式有基址寻址和变址寻址；允许直接和间接寻址，

说明还有直接寻址和间接寻址。这样，寻址方式有 4 种，寻址特征位 M 需要 2 位。剩余 7 位可作为形式地址。指令格式如下：

OP (7 位)	M(2 位)	A (7 位)
----------	--------	---------

直接寻址即形式地址为有效地址，则指令可直接寻址的范围是： $2^7=128$ 。

由于指令字长、机器字长和存储字长均为 16 位，而寄存器应该与机器字长相同（通常都是这样设计，处理起来方便快速），故间接寻址（操作数有效地址是 16 位）、基址寻址（基址寄存器 16 位）、变址寻址（变址寄存器 16 位）的寻址范围  $2^{16}=65536$ 。

(2) 因为机器字长和存储字长均为 16 位，主存  $16\text{MB}=8\text{M 字}=2^{23}$  字，即有效地址应为 23 位。故指令至少是双字长。这样， $M=2$ ， $A=23$ ， $OP=7$ 。格式如下：

OP (7 位)	M	A (23 位)
----------	---	----------

P325 例 7.6 某模型机共有 64 种操作，操作码位数固定，且具有以下特点：

- (1) 采用一地址或二地址格式。
- (2) 有寄存器寻址、直接寻址和相对寻址（位移量为  $-128\sim+127$ ）三种寻址方式。
- (3) 有 16 个通用寄存器，算术运算和逻辑运算的操作数均在寄存器中，结果也在寄存器中。
- (4) 取数/存数指令在通用寄存器和存储器之间传送数据。
- (5) 存储器容量为 1MB，按字节编址。

要求设计算逻指令、取数/存数指令和相对转移指令的格式，并简述理由。

解：

已知条件如下：

操作码位数固定，且有  $64=2^6$  种操作，说明操作码 OP 占 6 位。

寻址方式有寄存器寻址、直接寻址、相对寻址三种，则寻址特征位 M 需要 2 位。

相对寻址的位移量为  $-128\sim+127$ ，需要 8 位。

16 个通用寄存器，则寄存器编号需要 4 位。

算术运算和逻辑运算的操作数均在寄存器中，结果也在寄存器中，说明是二地址寄存器-寄存器型算逻指令。

取数/存数指令在通用寄存器和存储器之间传送数据，说明是二地址寄存器-存储器型传送类指令。

按字节编址的存储器容量为 1MB。说明操作数的有效地址需要 20 位，操作数应该为 8 位。

- (1) 算逻类指令：二地址，寄存器-寄存器型， $OP=6$ ， $M=2$ ， $R_i=4$ ， $R_j=4$ ，共 16 位，单字长即可。  
格式如下：

OP (6 位)	M	$R_i$ (4 位)	$R_j$ (4 位)
----------	---	-------------	-------------

- (2) 取数/存数传送类指令：在通用寄存器和存储器之间传送数据，二地址，寄存器-存储器型，的， $OP=6$ ， $M=2$ ， $R_i=4$ ， $A=20$ （能够进行 1MB 的寻址），共 32 位，双字长可行。  
格式如下：

OP (6 位)	M	Ri (4 位)	A (20 位)
----------	---	----------	----------

或

OP (6 位)	M	A (20 位)	Ri (4 位)
----------	---	----------	----------

(3) 相对寻址类指令：一地址，该地址就是位移量，OP=6，M=2，A=8（满足位移量-128~+127），共 16 位，单字长即可。

格式如下：

OP (6 位)	M	A (8 位)
----------	---	---------

P325 例 7.7 设某机共能完成 110 种操作，CPU 有 8 个通用寄存器（16 位），主存容量为 4M 字，采用寄存器-存储器型指令。

(1) 欲使指令可直接访问主存的任一地址，指令字长应取多少位？画出指令格式。

(2) 若在上述设计的指令字中设置一个寻址特征位 X，且 X=1 表示某个寄存器作为基址寄存器，画出指令格式。试问基址寻址可否访问主存的任一单元？为什么？如果不能，提出一种方案，使其可访问主存的任一位置。

(3) 若主存容量扩大到 4G 字，且存储字长等于指令字长，则在不改变上述硬件结构的前提下，可采用什么方法使指令可访问主存的任一位置。

解：已知条件如下：

某机共能完成 110 种操作，因  $2^6=64 < 110 < 2^7=128$ ，故操作码 7 位即可。

8 个 16 位通用寄存器，则寄存器编号需要 3 位。

主存容量为 4M 字= $2^{22}$  字，按字寻址需要 22 位有效地址。

从题目其它部分也可了解：存储字长=16 位，且按字寻址。

(1) 采用寄存器-存储器型指令，则 Ri=3 位；指令可直接访问主存的任一地址，则 EA=22 位；而 OP=7 位，故以上共 32 位，该指令字长必须为双字长指令。

格式如下：

OP (7 位)	A (22 位)	Ri (3 位)
----------	----------	----------

或

OP (7 位)	Ri (3 位)	A (22 位)
----------	----------	----------

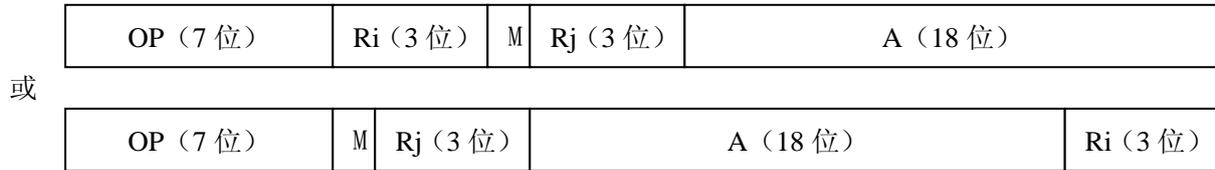
(2) 依据题意，仍然采用寄存器-存储器型指令，则寄存器寻址 Ri=3 位。

而在存储器访问方面采用基址寻址方式解决，寻址特征位 M=1 位；而任何寄存器都可以作为基址寄存器，Rj=3 位，M 和 Rj 构成基址寻址，需要 4 位。

这样，在这类指令中，OP、Ri、M、Rj 占 14 位，再需要若干位的形式地址，构成寄存器-存储器型指令。

若指令字长 16 位，形式地址只能占 2 位，不可行。故该种指令应为双字长 32 位，其中形式地址占 18 位。

格式如下：



由于 M 和 Rj 构成基址寻址，根据基址寻址的特点，即基址寄存器的值与位移量之和作为操作数地址。而基址寄存器的值为 16 位，位移量 A 为 18 位，之和仍为 18 位，只能寻址  $2^{18}=256K$ 。

为保证可以寻址 4M 字，可以采用以下方法：

- ① Rj 左移 6 位，低位补“0”，形成 22 位基地址，再与位移量 A 相加，生成有效地址；
- ② 增加一个 16 位的段寄存器，段寄存器左移 6 位，低位补“0”，再与 Rj、位移量 A 相加，生成有效地址。

(3) 若主存容量扩大到 4G 字，且存储字长等于指令字长，则在不改变上述硬件结构的前提下，可采用什么方法使指令可访问主存的任一位置。

所谓不改变上述硬件结构：指的是寄存器和存储器仍然为 16 位。

所谓存储字长等于指令字长：指的是最短指令的字长为存储字长，本题为 16 位，符合不改变硬件结构的要求。

若要寻址达 4G 字，存储器的有效地址应该 32 位。要想得到 32 位有效地址，必须采用间接寻址方式，即按照形式地址访问主存，先取得 16 位有效地址，形式地址+1，再次访问主存，再取得 16 位有效地址，两个 16 位有效地址拼接成 32 位有效地址。