

图中共画出了五种指令格式, 它们的操作码位数均为 8 位。RR 格式是寄存器-寄存器格式, 两个操作数均在寄存器中, 完成 $(R_1) \text{ OP } (R_2) \rightarrow R_1$ 的操作。RX 是二地址格式的寄存器-存储器型指令, 一个操作数在寄存器中, 另一个操作数在存储器中, 其有效地址由变址(X)和基址(B)寻址方式求得, 可以完成 $(R_1) \text{ OP } M[(X)+(B)+D] \rightarrow R_1$ 的操作。RS 格式是三地址格式的寄存器-存储器型指令, 完成 $(R_3) \text{ OP } M[(B)+D] \rightarrow R_1$ 操作。SI 格式中的 I 为立即数, 它完成立即数 $\rightarrow M[(B)+D]$ 的操作。SS 格式是存储器-存储器型指令, 两个操作数均在存储器, 这类指令用于十进制运算和字符串处理, 数据长度字段 L 可定义一个长度(1~256 个字符)或两个长度(每一个为 1~16 个十进制数), 它完成 $M[(B_1)+D_1] \text{ OP } M[(B_2)+D_2] \rightarrow M[(B_1)+D_1]$ 的操作。

4. Intel 8086/80486 系列机

Intel 8086/80486 系列微型计算机的指令字长为 1~6 个字节, 即不定长。例如, 零地址格式的空操作指令 NOP 只占一个字节; 一地址格式的 CALL 指令可以是 3 字节(段内调用)或 5 字节(段间调用); 二地址格式指令中的两个操作数既可以是寄存器-寄存器型、寄存器-存储器型, 也可以是寄存器-立即数型或存储器-立即数型, 它们所占的字节数分别为 2、2~4、2~3、3~6 个字节。有关该系列机的指令格式, 读者可以查阅有关资料自行分析。

7.4.3 指令格式设计举例

例 7.4 某机字长 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? 若安排寄存器寻址的一地址指令, 还能容纳多少条?

解: (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 7.22(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_x 的编号。剩下的 5 位可作操作码, 其格式如图 7.22(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。

(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 位可作操作码, 比格式(b)的操作码扩展了 3 位, 其格式如图 7.22(c) 所示。8 条这种指令的操作码为

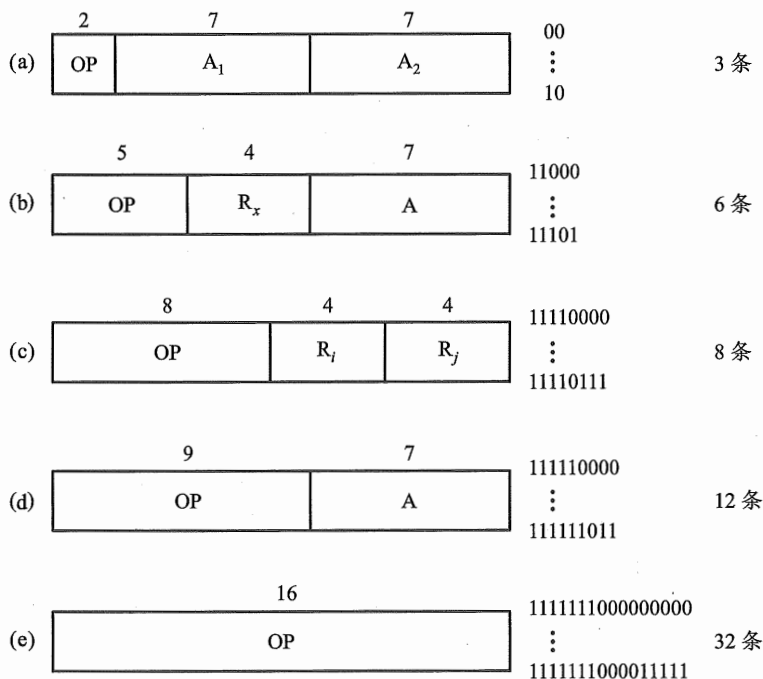


图 7.22 例 7.4 五种指令格式

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(c)的操作码扩展了 1 位,与格式(c)剩下的 8 个编码组合,可构成 16 个 9 位编码。以 11111 作为格式(d)指令的操作码特征位,12 条这种指令的操作码为 111110000~111111011,如图 7.22(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(d)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 7.22(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。

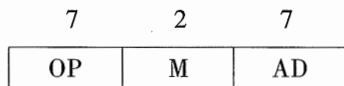
例 7.5 设某机配有基址寄存器和变址寄存器,采用一地址格式的指令系统,允许直接和间接寻址,且指令字长、机器字长和存储字长均为 16 位。

(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 16 MB 的主存?

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,

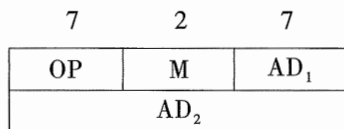
且有基址寄存器和变址寄存器,故取2位寻址特征位,其指令格式如下:



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 $2^7=128$,一次间接寻址的寻址范围是 $2^{16}=65536$ 。

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



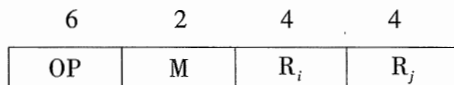
其中,形式地址为AD₁//AD₂,共 $7+16=23$ 位。 $2^{23}=8\text{ M}$,即可直接访问主存的任一位置。

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1 MB,按字节编址。

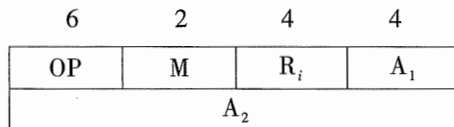
要求设计算逻指令、取数/存数指令和相对转移指令的格式,并简述理由。

解:(1) 算逻指令格式为寄存器-寄存器型,取单字长16位。



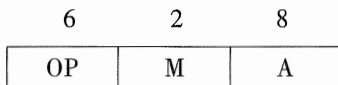
其中,OP为操作码,6位,可实现64种操作;M为寻址模式,2位,可反映寄存器寻址、直接寻址、相对寻址;R_i和R_j各取4位,指出源操作数和目的操作数的寄存器(共16个)编号。

(2) 取数/存数指令格式为寄存器-存储器型,取双字长32位,格式如下:



其中,OP为操作码,6位不变;M为寻址模式,2位不变;R_i为4位,源操作数地址(存数指令)或目的操作数地址(取数指令);A₁和A₂共20位,为存储器地址,可直接访问按字节编址的1 MB存储器。

(3) 相对转移指令为一地址格式,取单字长16位,格式如下:



其中, OP 为操作码, 6 位不变; M 为寻址模式, 2 位不变; A 为位移量 8 位, 对应位移量为 $-128 \sim +127$ 。

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 4 M 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。

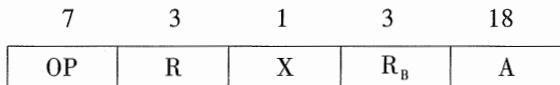
(2) 若在上述设计的指令字中设置一寻址特征位 X, 且 $X=1$ 表示某个寄存器作基址寄存器, 画出指令格式。试问基址寻址可否访问主存的任一单元? 为什么? 如果不能, 提出一种方案, 使其可访问主存的任一位置。

(3) 若主存容量扩大到 4 G 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存储器的任一位置?

解: (1) 欲使指令可直接访问 4 M 字存储器的任一单元, 采用寄存器-存储器型指令, 该机指令应包括 22 位的地址码、3 位寄存器编号和 7 位操作码, 即指令字长取 $22+3+7=32$ 位, 指令格式如下:



(2) 在上述指令格式中增设一寻址特征位, 且 $X=1$ 表示某个寄存器作基址寄存器 R_B 。其指令格式如下:



由于通用寄存器仅 16 位, 形式地址 18 位, 不足以覆盖 4 M 地址空间, 可将 R_B 寄存器内容左移 6 位, 低位补 0, 形成 22 位基地址, 然后与形式地址相加, 所得的有效地址即可访问 4 M 字存储器的任一单元。

(3) 若主存容量扩大到 4 G 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, 采用一次间接寻址即可访问存储器的任一单元, 因为间接寻址后得到的有效地址为 32 位, $2^{32}=4\text{ G}$ 。

7.5 RISC 技术

RISC 即精简指令系统计算机 (Reduced Instruction Set Computer), 与其对应的是 CISC, 即复杂指令系统计算机 (Complex Instruction Set Computer)。