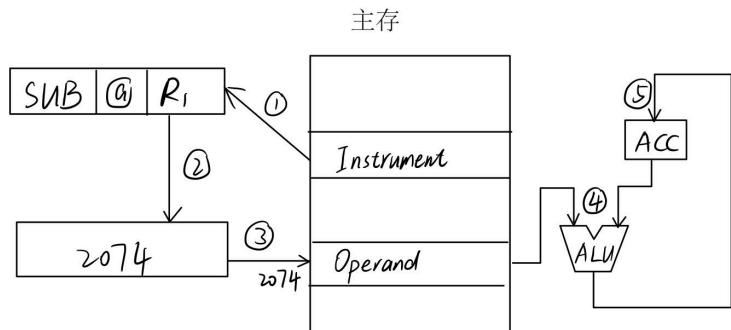


L7 指令系统

袁晨圃

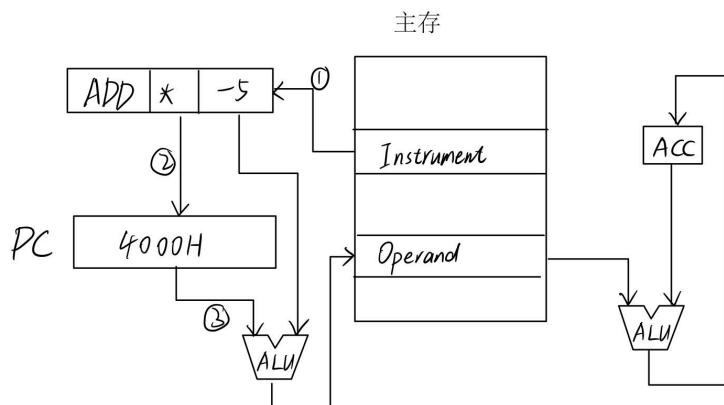
Exercise 1 (7.12): 画出 SUB @R1 指令对操作数的寻址及减法过程的流程图。设被减数和结果存于 ACC 中, @表示间接寻址, R1 寄存器的内容为 2074H

Solution:



Exercise 2 (7.13): 画出执行 ADD * -5 指令 (*为相对寻址地址) 的信息流程图, 设另一个和结果存于 ACC 中, 并假设(PC)=4000H

Solution:



Exercise 3 (7.15): 以相对寻址的转移指令占3个字节, 第一字节是操作码, 第二、三字节为相对位移量, 而且数据在存储器中采用以高字节为低地址的存放方式。假设 PC 当前值是 4000H, 当结果为 0, 执行 JZ * +35 和 JZ * -17 时, 该指令的第二、第三字节的机器代码为多少?

Solution: Big Endian

指令占 3 字节, 执行 JZ * +35 的时候 PC 已经是 4003H 了, 相对位移量 32 和 -20

大端序 32: 一二字节分别是 $00H$ 和 $20H$ -20: 一二字节分别是 FFH 和 ECH

00100011 00000000

11101111 11111111

Exercise 4 (7.16): 某主机内存为 $4M \times 16$ 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，试回答以下问题：

- (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) 一次间接寻址和多次间接寻址的寻址范围
- (4) 立即数的范围
- (5) 相对寻址的位移量（十进制）
- (6) 上述六种寻址方式的指令中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，使得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意位置，可采取什么措施？简要说明

Solution:

(1)	opcode	mode	address
-----	--------	------	---------

- opcode (7bit): 操作码
- mode (3bit): 寻址模式
- address (6bit): 形式地址

(2) $2^6 = 64$

(3) 一次间接寻址: 2^{16} , 多次间接寻址: 2^{15}

(4) 补码: $[-32]_{dec} \sim [31]_{dec}$, 无符号数: $0 \sim [64]_{dec}$

(5) $[-32]_{dec} \sim [31]_{dec}$

(6) 立即最短，因为它不需要寻址，直接在指令中给出操作数；间接最长，因为它需要多次寻址才能得到操作数。相对寻址便于程序浮动，变址寻址最适合处理数组问题。

(7) $4M : 22\text{位} = 6 + 16$

解决方案：将指令字长改为双字，另一个字表示剩下的部分地址

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) × 2⁶ + 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) 在 opcode 中加入直接寻址的转移指令，不使用 mode 部分位数，直接给出 8 位地址即可

（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—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。