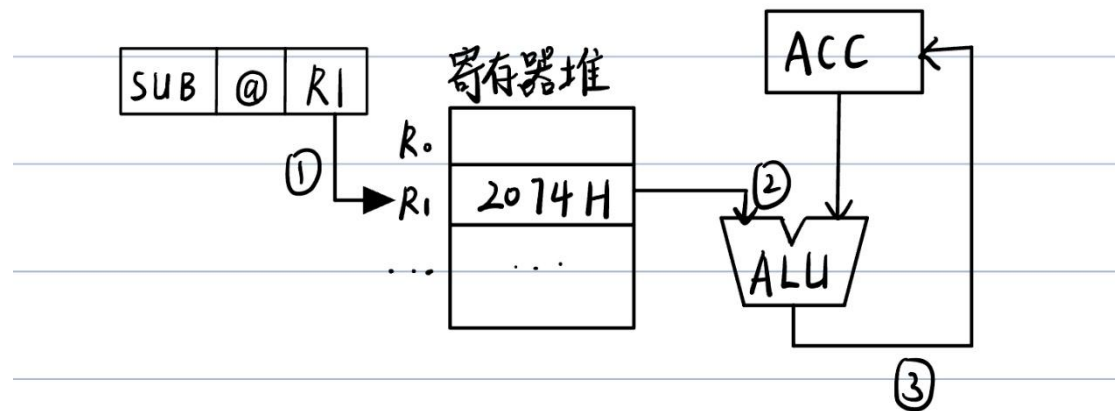


## 7.12

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，

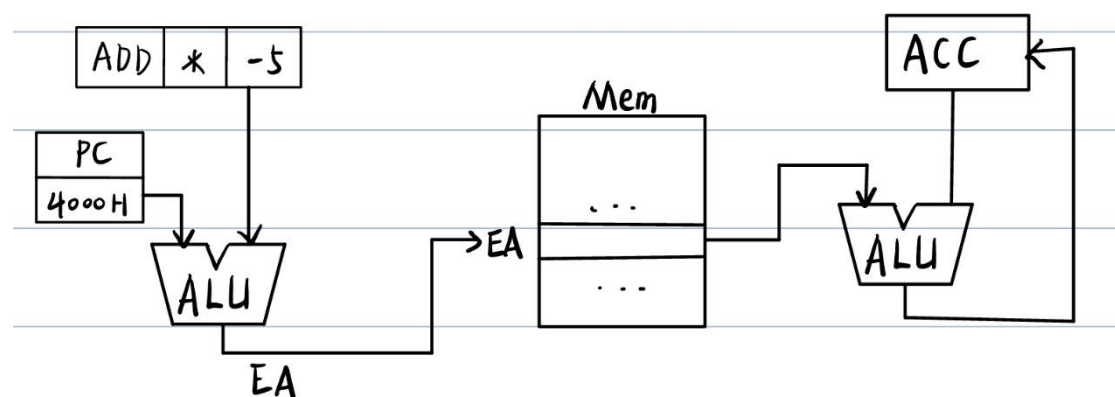
@表示间接寻址，R1 寄存器的内容为 2074H



## 7.13

画出执行“ADD \*-5”指令（\*为相对寻址特征）的信息流程图。设另一个操作数和结果存于 ACC 中，并假设 (PC) = 4000H。

注：图中画出了两个 alu，实际上只有一个



## 7.15

一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地址为字地址的存放方式。假设 PC 当前值是 4000H。试问当结果为 0，执行“JZ\*+35”和“JZ\*-17”指令时，该指令的第二、第三字节的机器代码各为多少？

PC 当前值为 4000H，取出三个字节的转移指令后，PC 值修改为 4003H。

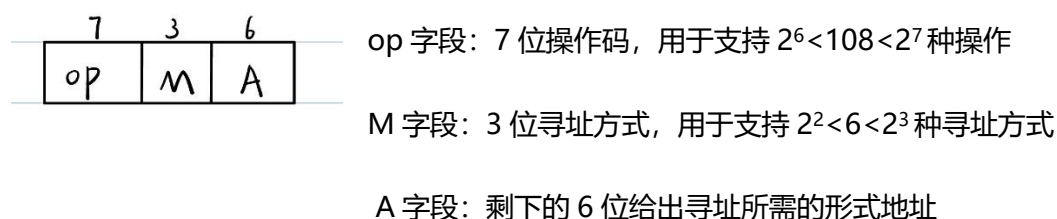
JZ\*+35 的相对位移量为  $35-3=32$  (十进制)，该指令的第二字节是 00H，第三字节是 20H。

JZ\*-17 的相对位移量为  $-17-3=-20$  (十进制)，该指令的第二字节是 FFH，第三字节是 ECH。

## 7.16

某机主存容量为  $4\text{ M}\times 16$  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。

直接寻址的最大范围为  $2^6=64$  字

(3) 一次间接寻址和多次间接寻址的寻址范围。

一次间接寻址范围为  $2^{16}=64\text{ K}$  字

多次间接寻址需用最高位表示是否继续间接寻址，范围为  $2^{15}=32\text{ K}$

(4) 立即数的范围 (十进制表示)。

有符号数:  $-32 \sim 31$

无符号数:  $0 \sim 63$

(5) 相对寻址的位移量 (十进制表示)。

$-32 \sim 31$

(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?

**立即寻址**的指令执行时间最短, 因为地址由指令中的立即数直接给出, 无需再转化为有效地址。

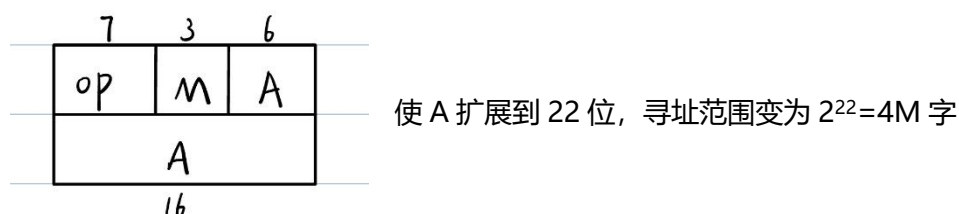
**间接寻址**执行时间最长, 因为其在指令的执行阶段需多次访存。

**相对寻址**便于程序浮动, 因为操作数位置可随程序存储区的变动而改变, 但操作数的有效地址只与当前指令地址相差位移量是一定的。

**变址寻址**最适合处理数组问题, 因为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, (变址值可自动修改而不需要修改程序)。

(7) 如何修改指令格式, 使指令的寻址范围可扩大到 4 M?

可采用双字长一地址指令, 即指令字长为存储字长的 2 倍, 一地址指令格式变为



(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。

使用(7)中的双字节一地址指令格式, 使用直接寻址即可转移到主存的任一位置