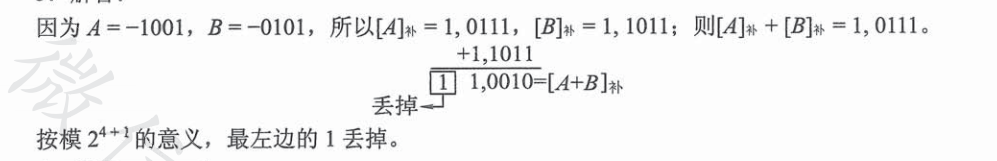
1. [X+Y]补=0.1011+1.1110=0.1001，[X-Y]补=0.1011+0.0010=0.1101

2．

3.

1)因为x=-68 =-(100 0100)2，则[-68]补=10111100=BCH;因y=-80=-(101 0000)，则[-80]补= 1011 0000= BOH，所以寄存器A和B中的内容分别是BCH、BOH。

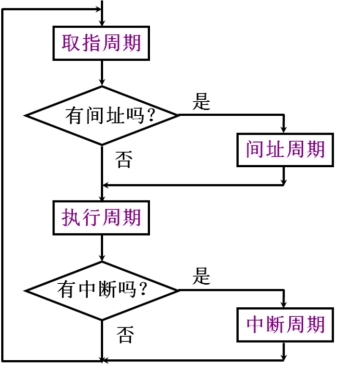
2) [x+y]补=[x]补+[y]补= 10111100+ 1011 0000 =101101100=6CH，所以寄存器C中的内容是6CH,其真值为108。此时，溢出标志位OF为1，表示溢出，即说明寄存器C中的内容不是真正的结果;符号标志位SF为0，表示结果为正数（溢出标志为1，说明符号标志有错);进位标志位CF为1，仅表示加法器最高位有进位，对运算结果不说明什么。

4. 1）静态RAM：依靠双稳态触发器保存二进制代码，只要不断电，信息就不会丢失；功耗较大，集成度较低，速度快，每位价格高，适合于作Cache或存取速度要求较高的小容量主存。

2）动态RAM：依靠电容存储电荷来保存二进制代码，需刷新电路进行动态刷新，存取速度较慢；功耗小，集成度高，每位价格低，适合于作大容量主存。

5.

流程图如下：



取指周期：完成取指令和分析指令的操作。

间址周期：取操作数的有效地址。

执行周期：执行指令的操作。

中断周期：将程序断点保存到存储器，转向中断服务程序入口

6.

**1）断定方式。**该方式直接由微指令的下地址字段指出，属于最基本的也是最常用的微指令地址形成方式。

**2）微地址形成部件生成。**该方式根据机器指令的操作码经微地址形成部件编码形成。

1. **增量计数器法（增量方式）。**该方式要求微指令在控存中顺序存储。

**4）分支转移。**该方式根据各种标志位决定微指令分支转移的地址，主要用于条件转移指令。

**5通过测试网络形成。**该方式下，微指令地址中的高段由H段地址码直接形成，低段为测试地址，通过检测若干测试源（比如触发器）的状态确定。

**6）由硬件产生微程序入口地址。**该方式下，诸如取指微程序入口地址、中断周期微程序入口地址等由专门的硬件电路产生

7.

1. 将控制器所需要的微操作命令（控制信号），以微代码的形式编成微指令，若干条微指令构成微程序，存在CPU的控制存储器中；
2. CPU执行机器指令时，找到微程序入口，从控制存储器中逐条取出构成微程序的微指令；
3. 对微指令中的操作控制字段进行解释，即产生执行机器指令所需的微操作命令序列。

8. DMA的数据传送过程可分为预处理、数据传送和后处理3个阶段。工作过程如下所示：

**各阶段完成的工作如下：**

1) **预处理阶段**：CPU执行主程序实现DMA传送的初始化设置；

2）**数据传送阶段**：由DMA控制器实现内存和外设间的数据传送。

3）**后处理阶段：**中断处理程序判断传送的正误，，对写入主存的数据进行校验，完成善后工作

9. Cache-主存地址映射有直接映射方式、全相联映射方式和组相联映射方式三种。

直接映射方式的特点：主存的字块只可以和固定的Cache字块对应，优点是方式直接，硬件实现电路简单，成本低；缺点是利用率低，同时命中率和效率较低。

全相联映射方式主存中的字块可以和Cache的任何字块对应，优点是方式灵活，利用率高；缺点是所需逻辑电路复杂，使用成本太高。

组相联映射方式是对前两种映射方式的折衷，组间全相联，组内直接映像。其特点是集中了两个方式的优点，成本也不太高，是目前应用最为广泛的Cache映射方式。

10.

1）根据操作数地址码为6位，得到二地址指令中操作码的位数为16-6-6=4，这4位操作码可有16种操作。由于操作码固定，因此除了零地址指令有M种，一地址指令有N种，剩下的二地址指令最多有16-M-N种。

2）采用扩展操作码技术，操作码位数可随地址数的减少而增加。对于二地址指令，指令字长16位，减去两个地址码共12位，剩下4位操作码，共16种编码，去掉一种编码（如1111）用于一地址指令扩展，二地址指令最多可有15种操作。

3)采用扩展操作码技术，操作码位数可变，二地址、一地址和零地址的操作码长度分别为4位、10位和 16位。这样，二地址指令操作码每减少一个，就可以多构成26条一地址指令操作码;一地址指令操作码每减少一个,就可以多构成26条零地址指令操作码。设一地址指令有R条，则一地址指令最多有(24-P)×26条，零地址指令最多有[(24-P)x26-R]×26条。根据题中给出零地址指令为Q条，即Q=[(24- P)×26-R]×26，得R=(24- P)x26-Q×2-6。