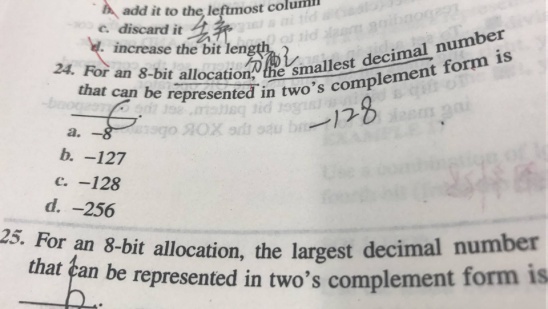
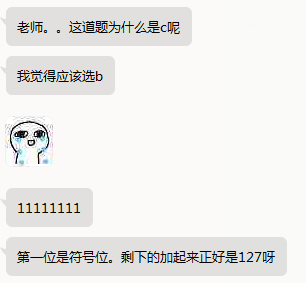
**问题1. 练习卷PartI上的24题**





答：题目：8位的补码表示的最小十进制数是多少？答案为-128，原因如下：

补码10000000对应的十进制数-128.

11111111对应的十进制数-1.

10000001对应的十进制才是-127.

启发：1. 给出补码形式需要用对应的变换方法求出其实际的十进制数。可以使用补码减1得到反码再求原码的方法。也可以用书上的方法：高位符号位1，则表示数为负数，然后从右边开始，开始遇到的0不变，当遇到第一个1时不变但1后面的就要取反了，得到的二进制数转成十进制再加上负号就对了。

2. 对于高位是1的反码和原码对应的十进制数都是负数，求值时如果是反码要先取反，而原码除掉符号位的部分就是数值了。

3. 正数的原码、反码和补码都相同。

4. 下面的图以4位二进制为例，可以对照图1原码和图2补码表示。

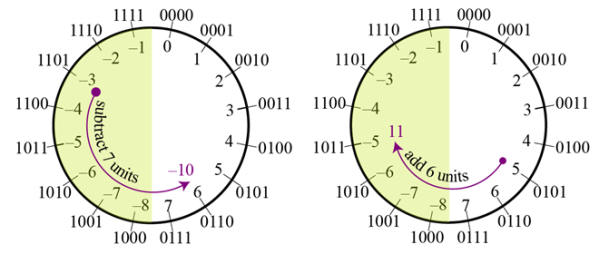
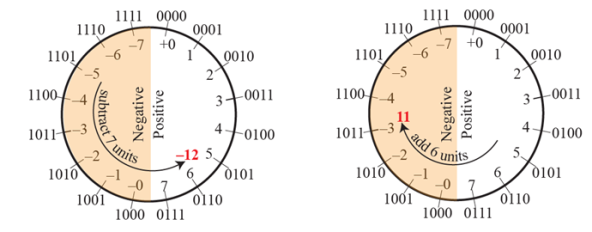


图1. 原码（可以看出原码1000为-0） 图2补码（可以看出补码1000为-8，补码没有-0）

**问题2. 关于第五章教材上的80题及相关问题。**

**数据寄存器**的作用是用来存储数据，本题中每个寄存器使用寄存器的名字R0-R3来标识，所以对于4个寄存器，需要2位二进制来标识。

**存储器（内存）**是用来存储数据和指令，存储器的地址用来标识每个可以访问的存储单元，本题中存储器一共有1024个字，说明本题中一共有1024个存储单元可以访问，因为每个存储单元都需要一个唯一的地址来标识，所以主要10位二进制来标识每个存储单元。所以本题中的存储器**地址空间**是从0到1023（一共1024个），如果写出二进制就是0000000000--1111111111（10个1），如果写出16进制就是0x000—0x3FF；

一条**指令**是由操作码（即所做的操作，如ADD,LOAD,STORE,SUBTRUCT等）和操作数（对谁进行操作）来组成的，所以指令的长度就是操作码的长度加上操作数的长度。本题中，给出的指令格式为：add 565 R2。此条指令的含义是将存储器地址565中存储的数和寄存器R2中存储的数相加。其中操作码为add，两个操作数分别为存储器地址565中的数和寄存器R2中的数。本题中一共支持16种操作，所以为了标识每一种具体的操作，需要4位二进制，再加上一个存储器中的操作数和一个寄存器中的操作数（即一个存储器地址的长度和一个寄存器标识的长度）一共为4+10+2=16。所以指令长度为16。

81.题中问：如果数据和指令的字长相同即长度相同，则数据寄存器多大？

因为给出数据和指令一样长，而数据寄存器是用来临时存放数据的，所以数据寄存器的大小也为16.也就是说本题中的R0-R3四个寄存器都是16位的。

82.问：指令寄存器instruction register的长度？

因为指令寄存器（IR）是用来临时存储指令的，所以指令寄存器的长度为16位。

83.问：程序计数器program counter的长度？

因为程序计数器是用来存储下一条指令的地址的，也就是给出下一条指令在存储器中哪个存储单元，本题中一共有1024个存储单元，前面也提到一共需要10位二进制来标识，所以PC的长度为10。

84.问：数据总线多宽？

因为数据总线是在CPU和memory之间传输数据的通道（这里的数据是广义的，也就是说不仅指操作数本身，还包括指令等），所以数据总线的宽度也是16位的，以保证存储单元中的一个字可以一起从内存读入CPU或者从CPU写入内存。

85.问: 地址总线的宽度？

地址总线是地址信息的通道，当CPU要访问某个内存单元时，该内存单元的地址信息就是通过地址总线来传输的，比如CPU在读取每条指令的时候，PC中的指令地址就是通过地址总线来传输的，当控制器发出存储器读命令时，存储器按给定的地址读出指令，又如本题中的565这个存储器地址也是通过地址总线传输的。

86.问：控制总线的宽度？

控制总线主要用来传输控制信号（和时序信号）的，因为本题中的控制信号数量为16种，即计算机支持的16种不同的指令/操作（ADD,LOAD,STORE,SUBTRUCT等），所以控制总线的宽度为log216

**其他问题**：假设本例中的简单计算机一共支持16种操作，其中ADD的编码为0010，寄存器R0到R3的编码为00,01,10,11，那么指令ADD 565 R2的二进制编码是什么？

答：该指令的二进制编码为：0010100011010101。因为add的编码为0010,565的编码为1000110101，即中间红色的10位，R2的编码为01。一共是16位。

当然如果假设指令时ADD 20 R2（把存储单元20中的内容与R2相加），那么指令的长度也为16，二进制编码为：0010 0000010100 01，中间红色的部分为存储单元20的编码，因为地址长度为10，所以10100前面需补齐00000，凑成10位。一共16位。