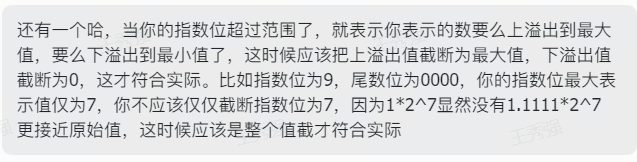
有关FPX的说明

这是针对龚成老师有关于世界先进的浮点数量化FPX的说明。先放OJ题，然后从三个不同的方式讨论，//最后提一下有关INF和NAN的处理问题，有点扯远了(也许是我不会hhh)，不写。

浮点数的量化指的是将高位宽的浮点数截断为低位宽的浮点数，其中对尾数位的量化截断是位截断，而对于指数位，由于其是经过移码而来的，则需要采用值截断(我也不知道为什么，他给的按他讲得做吧)。先看指数位，然后根据指数位的不同截断结果再来操作尾数位。

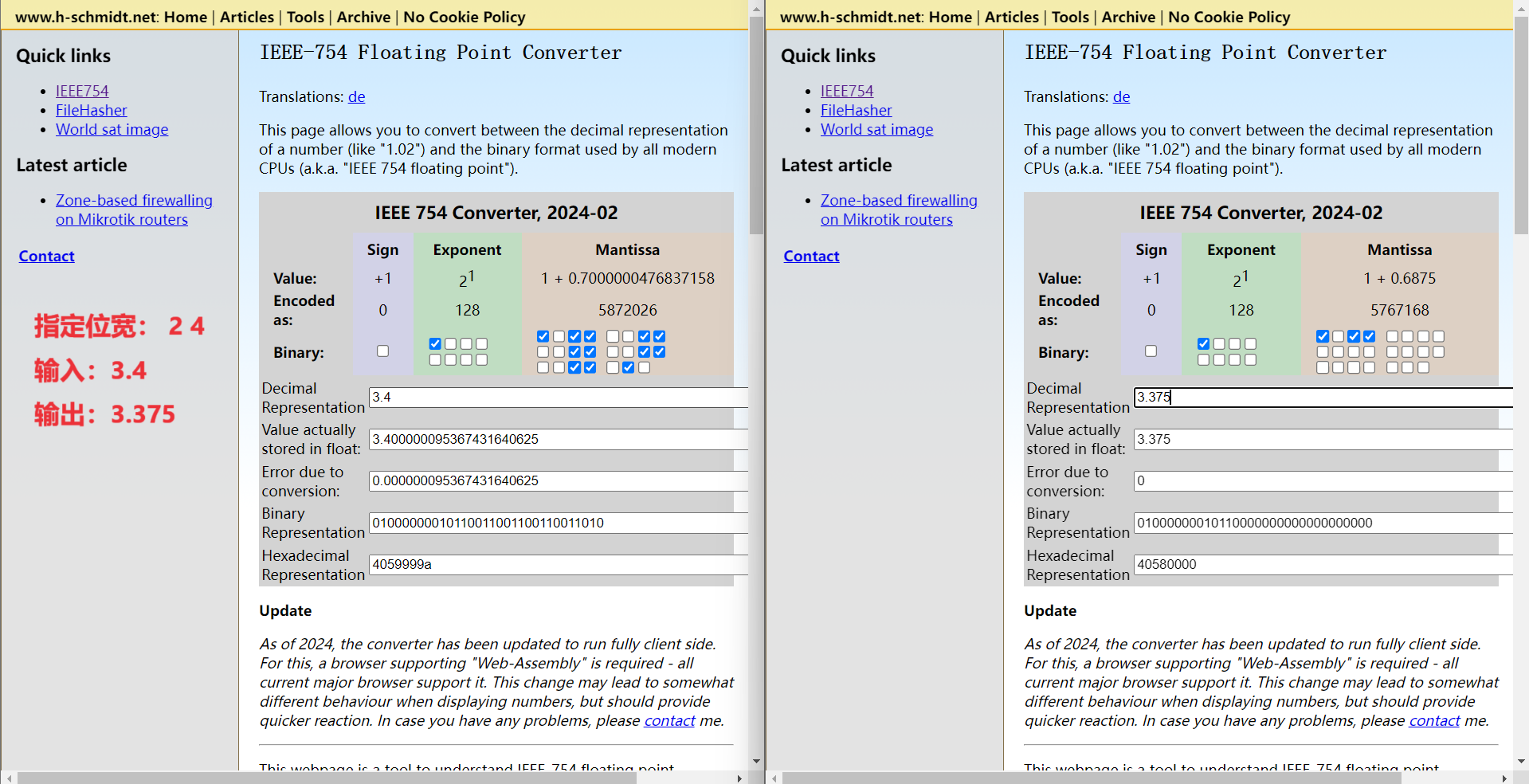
这里我们要先解释什么是指数位的值截断。在截断之前我们需要先将指数位假设我们指定位宽为4位，则这个4位能够表示的范围为，也就是signed类型的(指数默认为signed类型的，因为要表示较小的数，如果真实值都是大于0的，很显然，较小的数0.25之类的就无法表示，所以这里仍然是采用signed类型)。当指数处在这个区间时，我们就不用对指数进行截断。当指数上溢出这个区间时，我们需要将指数截断为区间上界，同时将尾数位尽可能地保留多一些一些值从而保证该值尽可能与原始值接近，也就是将尾数位指定位宽调整为1。

**

就是这里说的。

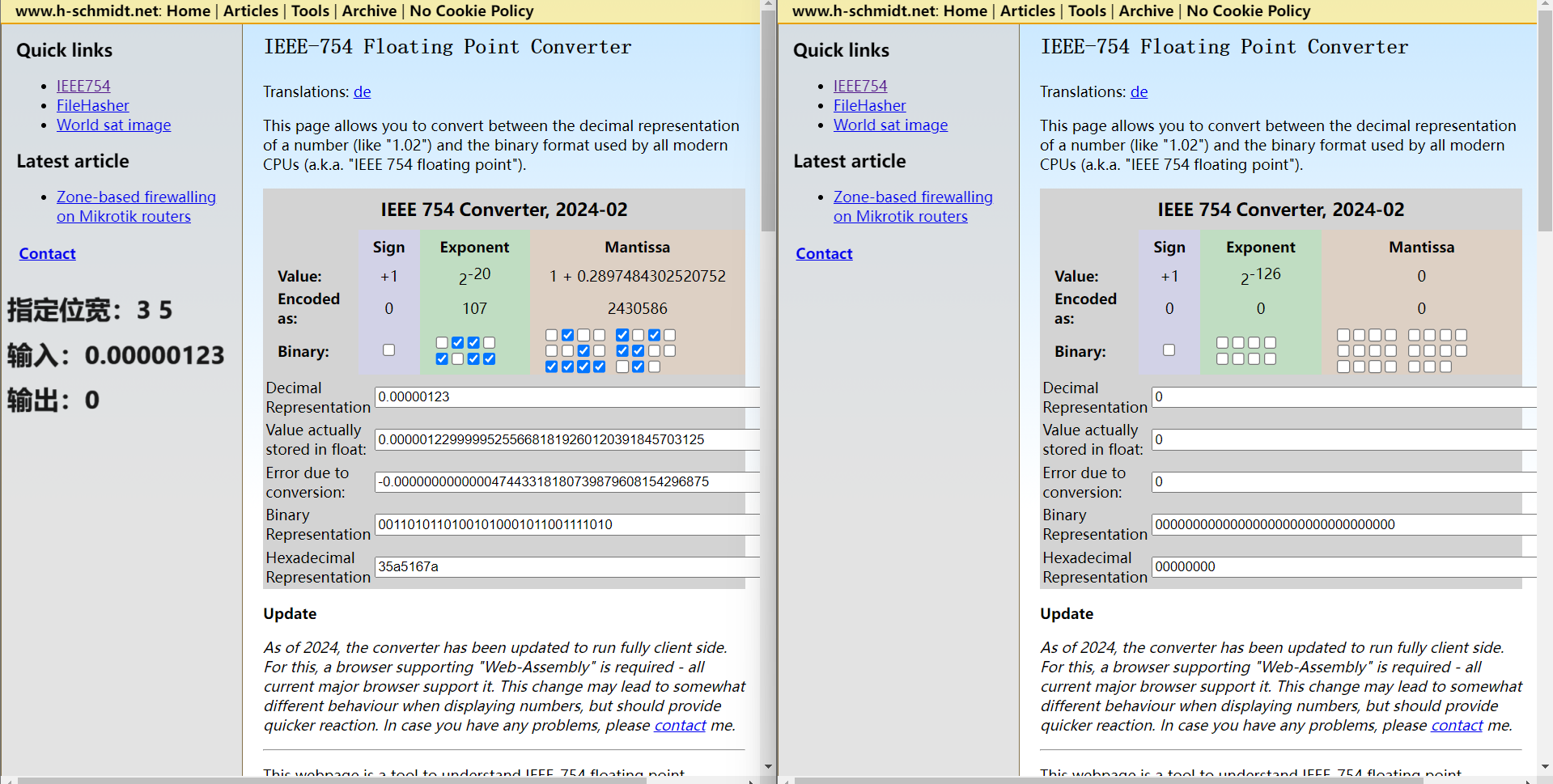
但是指数下溢的情况同指数上溢的情况有所不同，当指数下溢时，说明该数比要量化的对象还要小，如果像指数上溢一样将指数位对齐到区间下界，同时把尾数位指定位宽设置为0，就会造成数据扩大，精度损失的问题。因此，指数下溢的情况通常都不如直接将该浮点数直接量化为0。

1. 指数值在指定区间上



1. 指数下溢

对于指数下溢的问题，倘若我们指定位宽为5位，那么能表示的就是2^-5,如果我们的指数位是-8，显然相比于2^-5,0更加接近原始数据，因此下溢的问题统一处理为0.



1. 指数上溢



