FP162INT8算法说明

该工程实现FP16规格化数转换为INT8

1. Step1：pre\_operation

该阶段需要对输入数做预处理，分离出符号位，根据阶数判断上下溢出或者分离出应保存的尾数，根据舍弃的尾数判断舍入

* 符号位直接根据输入的第16位保存即可
* INT8只有7位用于表示数字，对应到FP16规格化的阶数为21-14（14时仍有可能通过入来摆脱下溢出），除此之外大于21判断为上溢出overflow，小于14判断为下溢出zero
* 阶数在21-14时，根据阶数的不同，拼接规格化省略的1和相应的尾数
* 舍入原则为舍入到最近可以表示的数，若距离一样，则偏好舍入到偶数。算法为 [被舍弃部分的最高位 && (保留部分的最低位 | | 被舍弃部分的次高位及以下是否有1)]

1. Step2：round to nearest even

该阶段根据step1中舍入的判断依据ifround，来对保留数rmcache进行+1与否的修正

* 如果没有上溢出也没有下溢出，则判断舍入与否，入则+1，舍则不变
* 如果有上溢出或者下溢出，保留溢出标志，其他不做操作

1. Step3：overflow and carry

由于step2中的入操作可能再一次导致溢出，该阶段检查是否溢出并修正

* 如果没有上溢出或者下溢出，检查保留数的第8位，该位为1则溢出，上溢出标志置1，其余不做操作；为0则没有溢出，上溢出标志置0，保留数rmcache不变
* 如果有上溢出或者下溢出，保留溢出标志，其他不做操作、

1. Step4：result

该阶段可以得出结果

* 如果上溢出，输出置0xff
* 如果下溢出，输出置0x00
* 如果正常，拼接{符号位，保留数低七位}输出