\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**2024/9/13**

整理变量名

加了两个简单的降噪方式 移动平均滤波 和 中值滤波

同步至GITHUB仓库

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

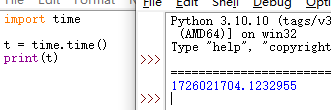
**2024/9/12**

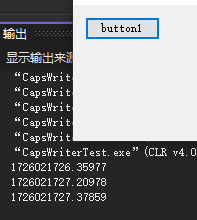
整理项目结构

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**2024/9/11**

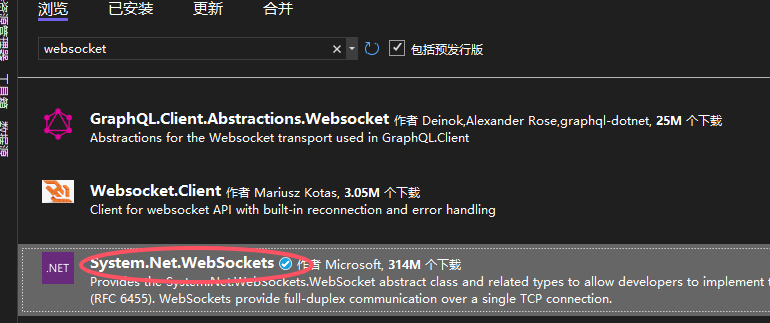
完成C#时间戳到python Time库的时间戳转换





完成了客户端

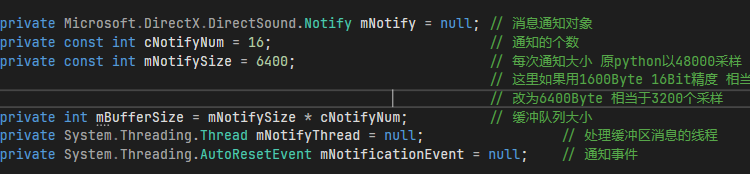
使用了System.Net.WebSockets



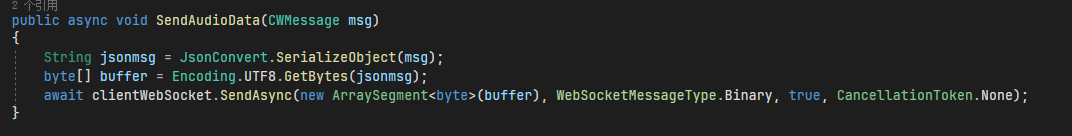
现在有如下问题，生产数据比消费数据要快，如果减少notifysize会导致发送来不及

一开始使用和python一样的配置，python是0.05\*48000=2400个采样，每3个平均后，以float32发送，相当于发送800个采样，3200Byte字节一次

同样800个采样，由于我这里使用16Bit，因此通知Byte为1600Byte，如果设置为1600Byte，



SendAudioData就会来不及，目前靠调大这个Size在使用，后续需要改进生产者消费者模式来处理



对于DB值进行了一下转换

采样的DB值，和我们通常理解的响度DB值之间有差异，查了资料才知道

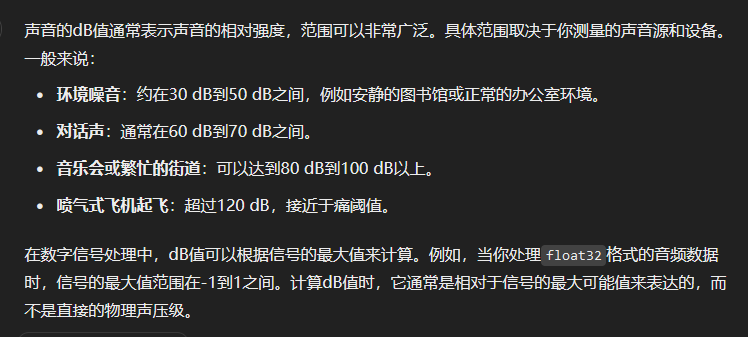
C#中 16Bit采样 值的范围为-32767至32768

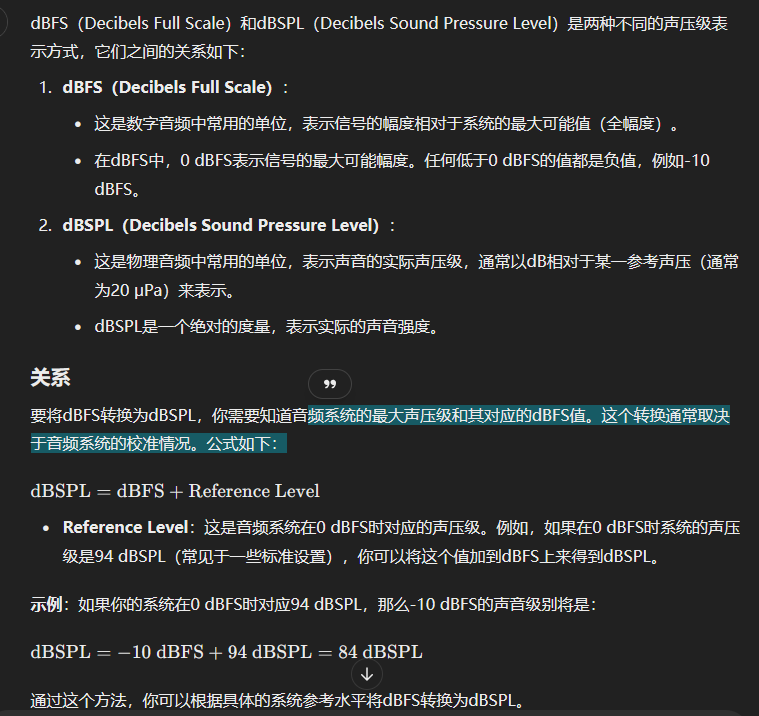
Python Float32采样 相当于上述值归一化 在-1.0至1.0之间

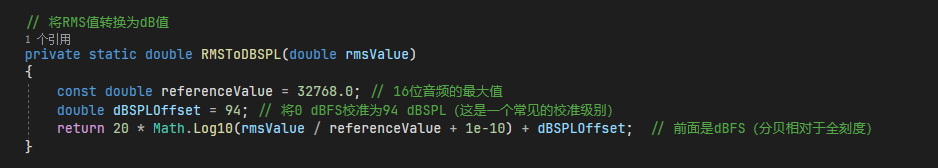
将各值平方后求均方根值，可以扩大DB值的量化

最后用公式20\*log10(rmsValue)求出RMS

Db有SPL和FS区别







写这个Client比多预计的要顺利

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*