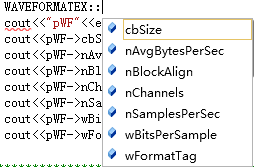
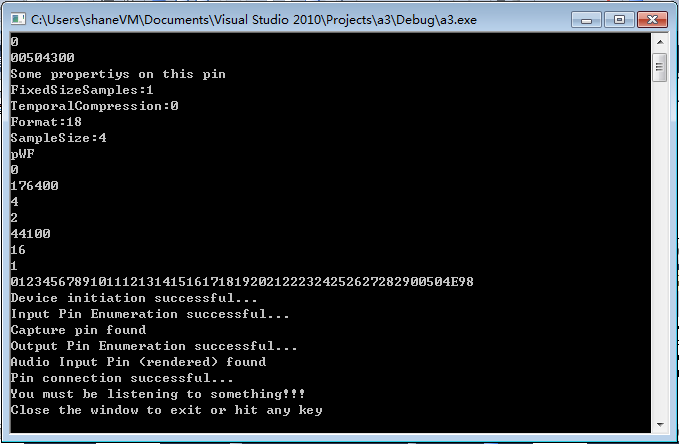
# 媒介参数以及缓存参数的获取

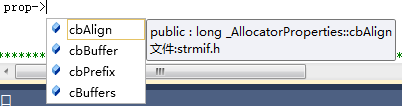
根据设置用结构体

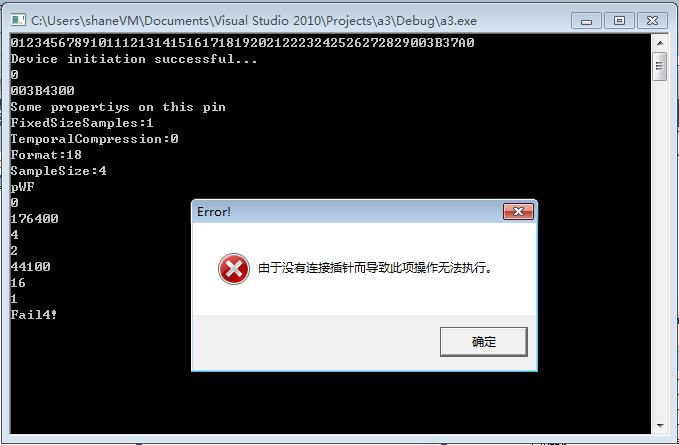






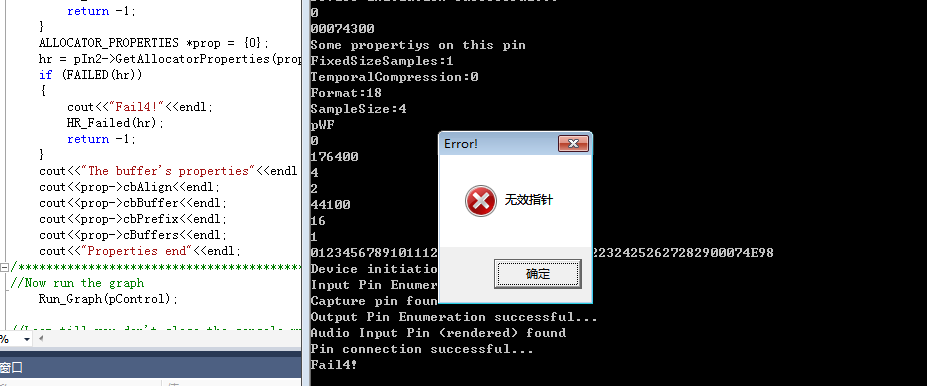
还有缓存优化信息可以使用结构体ALLOCATOR\_PROPERTIES来描述。



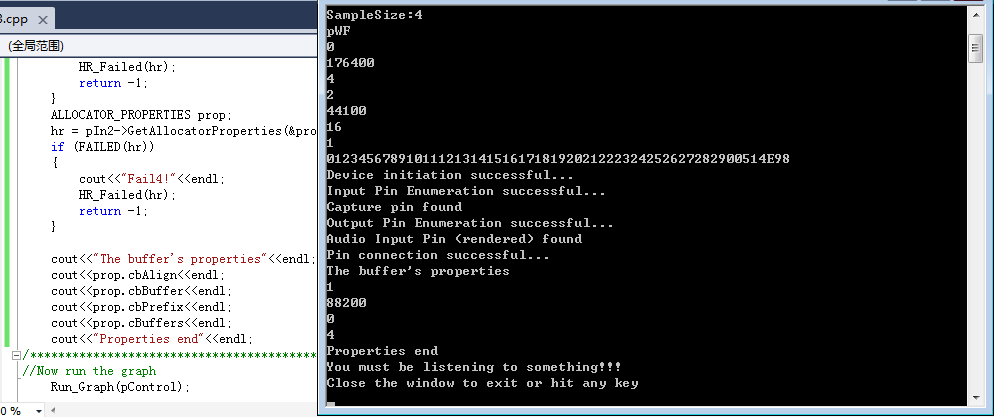


缓存只有在连接Graph之后才会分配缓存。

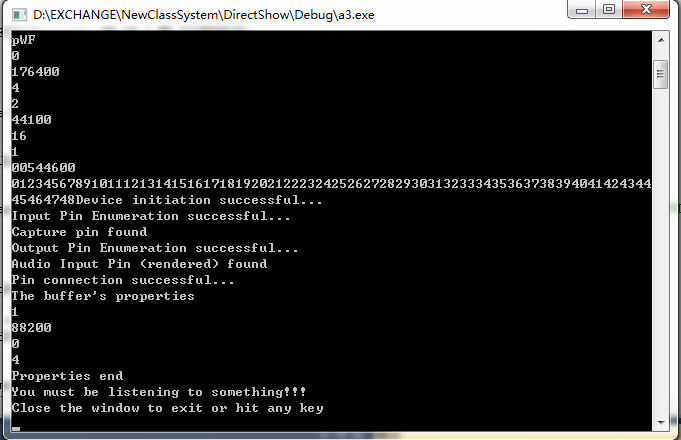
但是更改在Graph建立连接之后又出现了新的问题，就是出现无效指针，原因为只能使用结构体而不能使用指针进行参数的获取。



之后只要吧prop改为结构体的声明并且传入指针就可以正常的获取缓存的参数了。



最后捕获下来的USB声卡的相关媒体以及缓存信息。

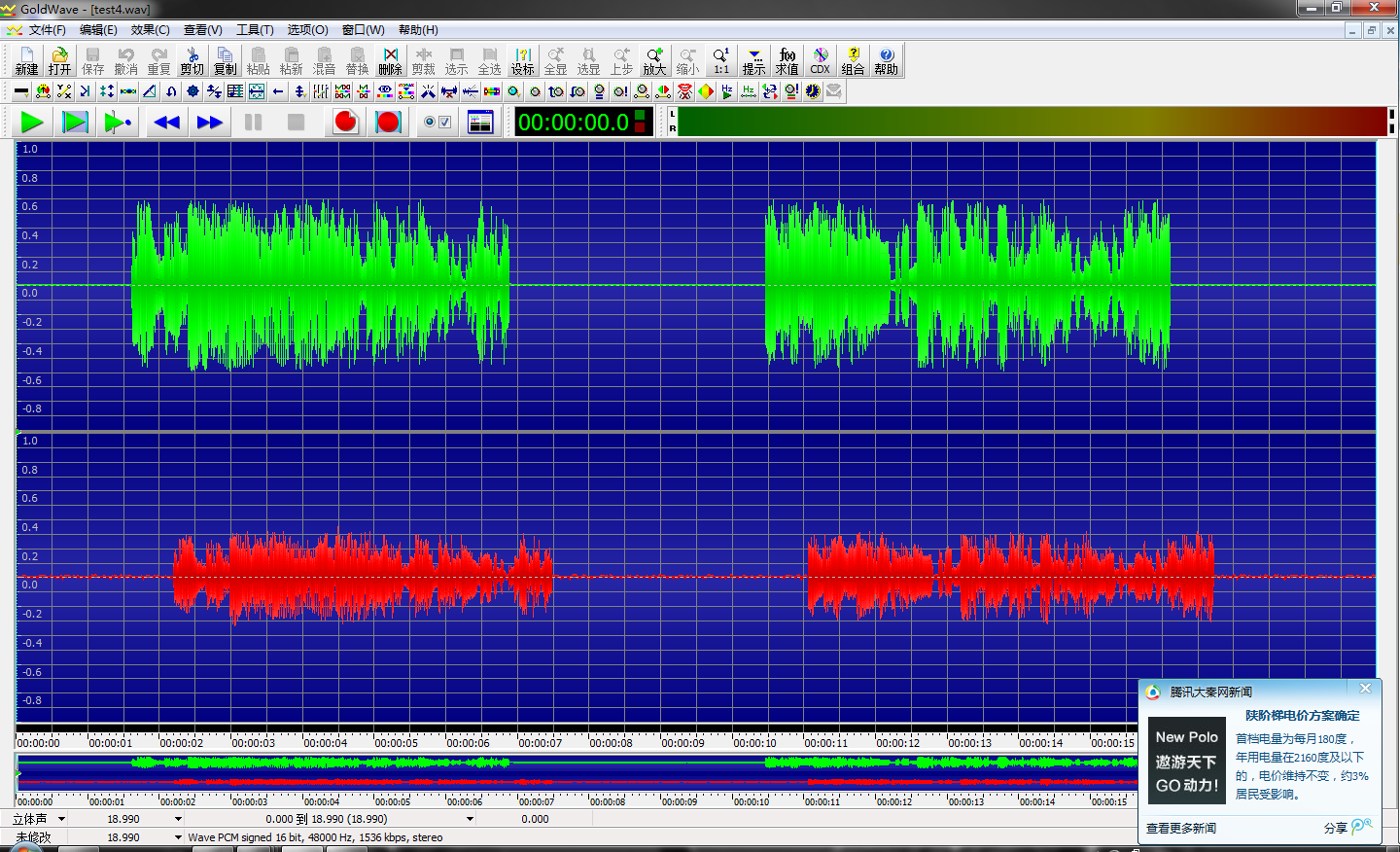


# 降低的延迟的主要思路有两种

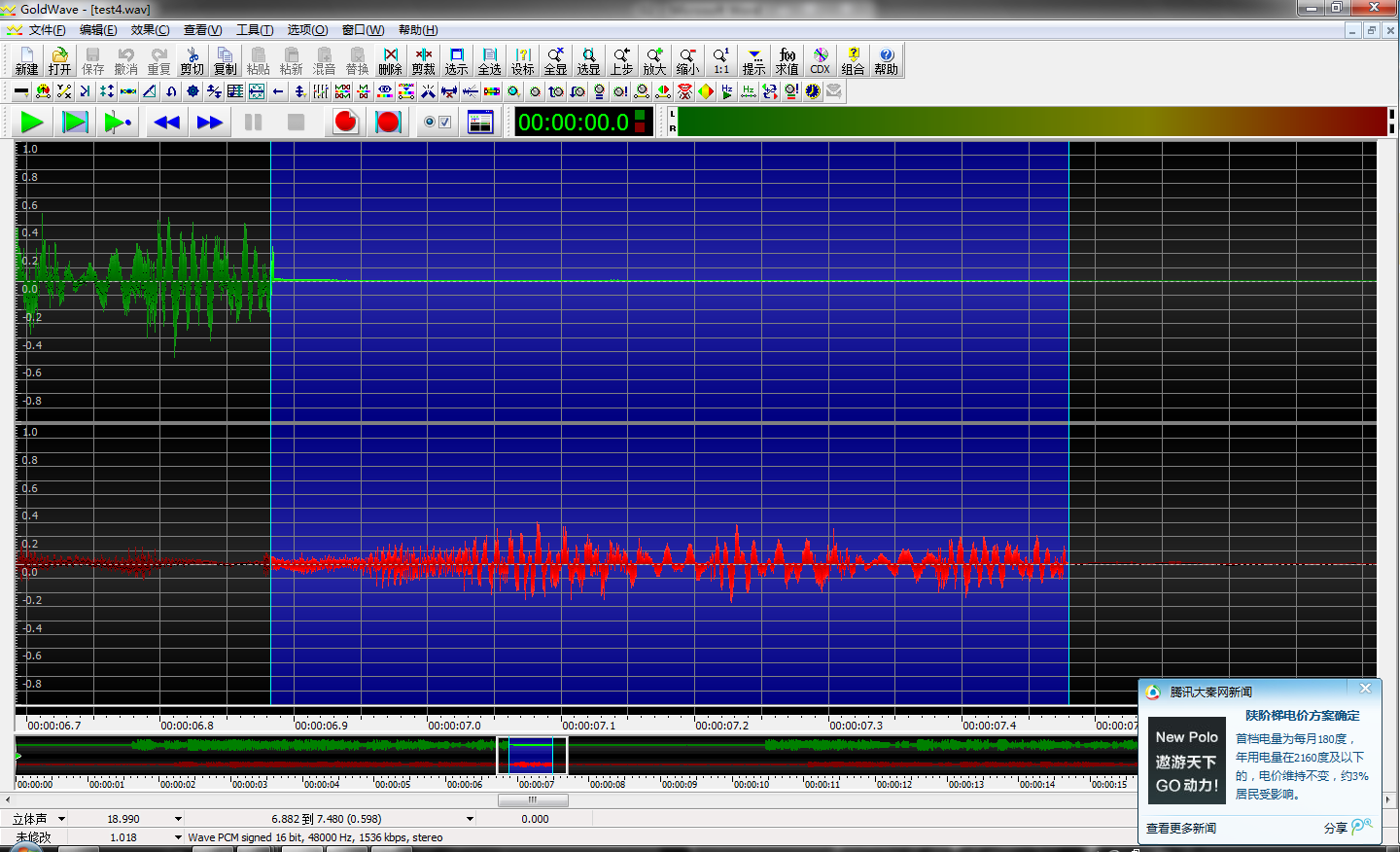
根据相关资料，Audio Capture默认会设置容纳0.5s媒体的缓存区，

1. 所以一种方法是简单粗暴的减小缓存区的大小；
2. 另外一种就是通过相关的公式以及录制的媒体的性质，一方面减小线入媒体的音质，另一个方面减小缓存区的大小。

# 再测延迟



具体的延迟段



# 音频媒体参数汇总

## 音频媒体信息

|  |  |  |
| --- | --- | --- |
| cbSize | 0 | 未知 |
| nAvgBytesPerSec | 176400 | 平均码率 |
| nBlockAlign | 4 | 未知 |
| nChannels | 2 | 声道数 |
| nSamplesPerSec | 44100 | 平均采样率 |
| wBitesPerSample | 16 | 采样大小 |
| wFormatTag | 1 | 未知 |

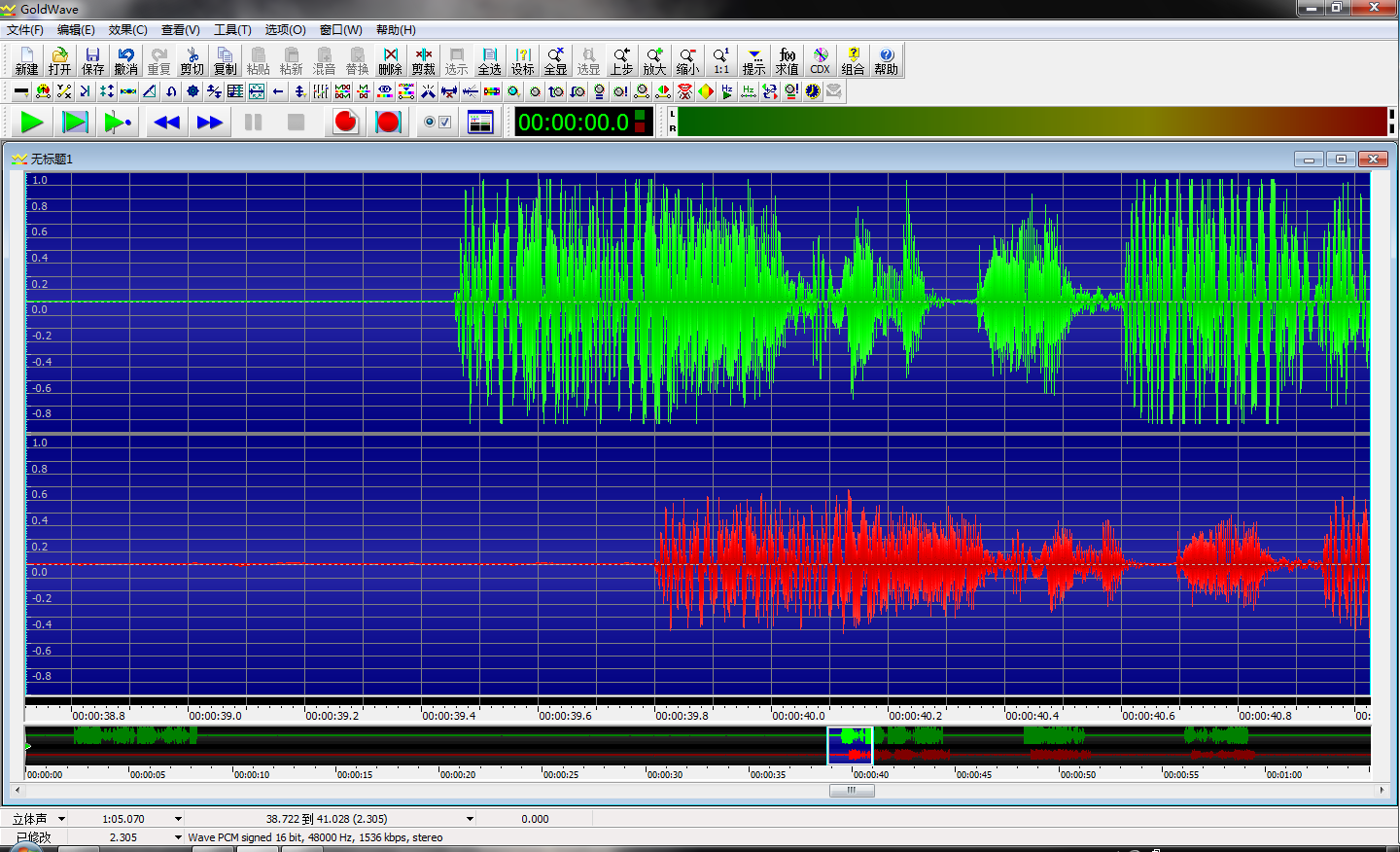
## 缓存信息

|  |  |  |
| --- | --- | --- |
| cbAlign | 1 | 未知 |
| cbBuffer | 88200 | 缓存大小 |
| cbPrefix | 0 | 未知 |
| cBuffers | 4 | 未知 |

## 公式总结可得

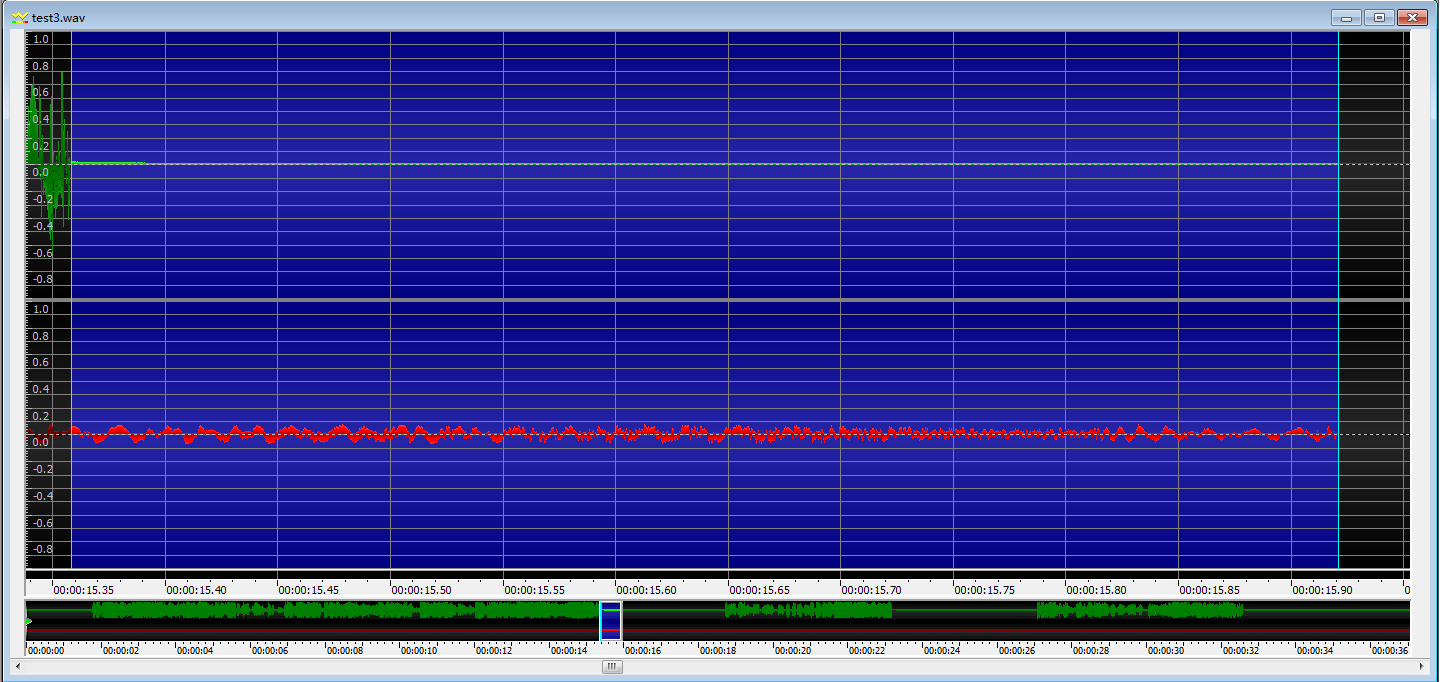
# 系统延迟优化

需要注意的是在使用方法IAMBufferNegotiation:: SuggestAllocatorProperties(&prop)的时候，务必要保证Capture接口未连接。



## 500ms缓存延迟

未改之前的波形为



未改之前的延迟为



563ms

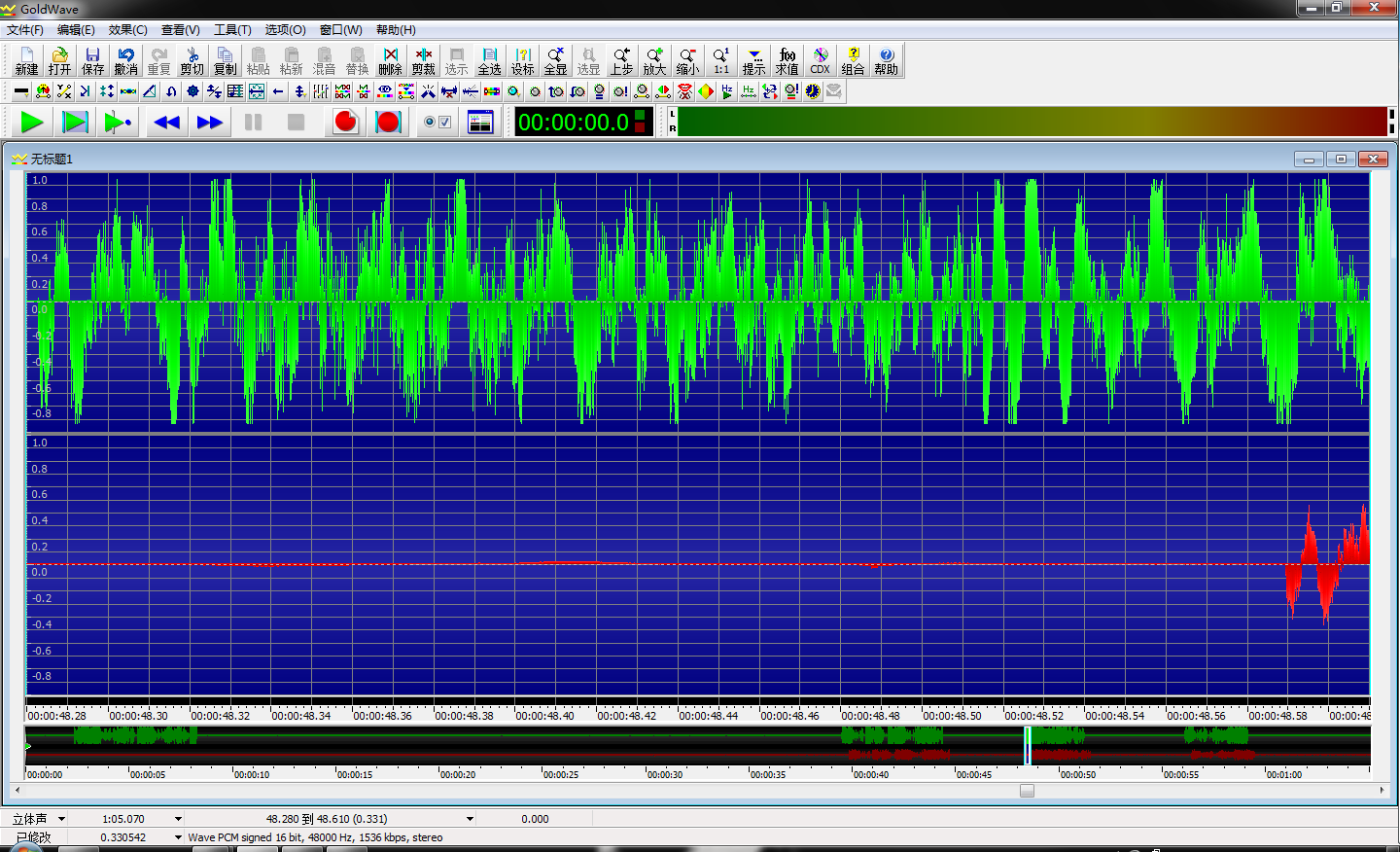
可以基本上确定系统延迟为50-70ms之间。

## 250ms缓存延迟

当使用下列的Buffer属性优化之后，延迟得到改善。

|  |  |  |
| --- | --- | --- |
| cbAlign | 1 | 未知 |
| cbBuffer | 44100 | 缓存大小 |
| cbPrefix | 0 | 未知 |
| cBuffers | 4 | 未知 |

具体延迟见下图：



起始点位48.283

结束点位48.589

延迟为307ms

延迟有了较大的改观



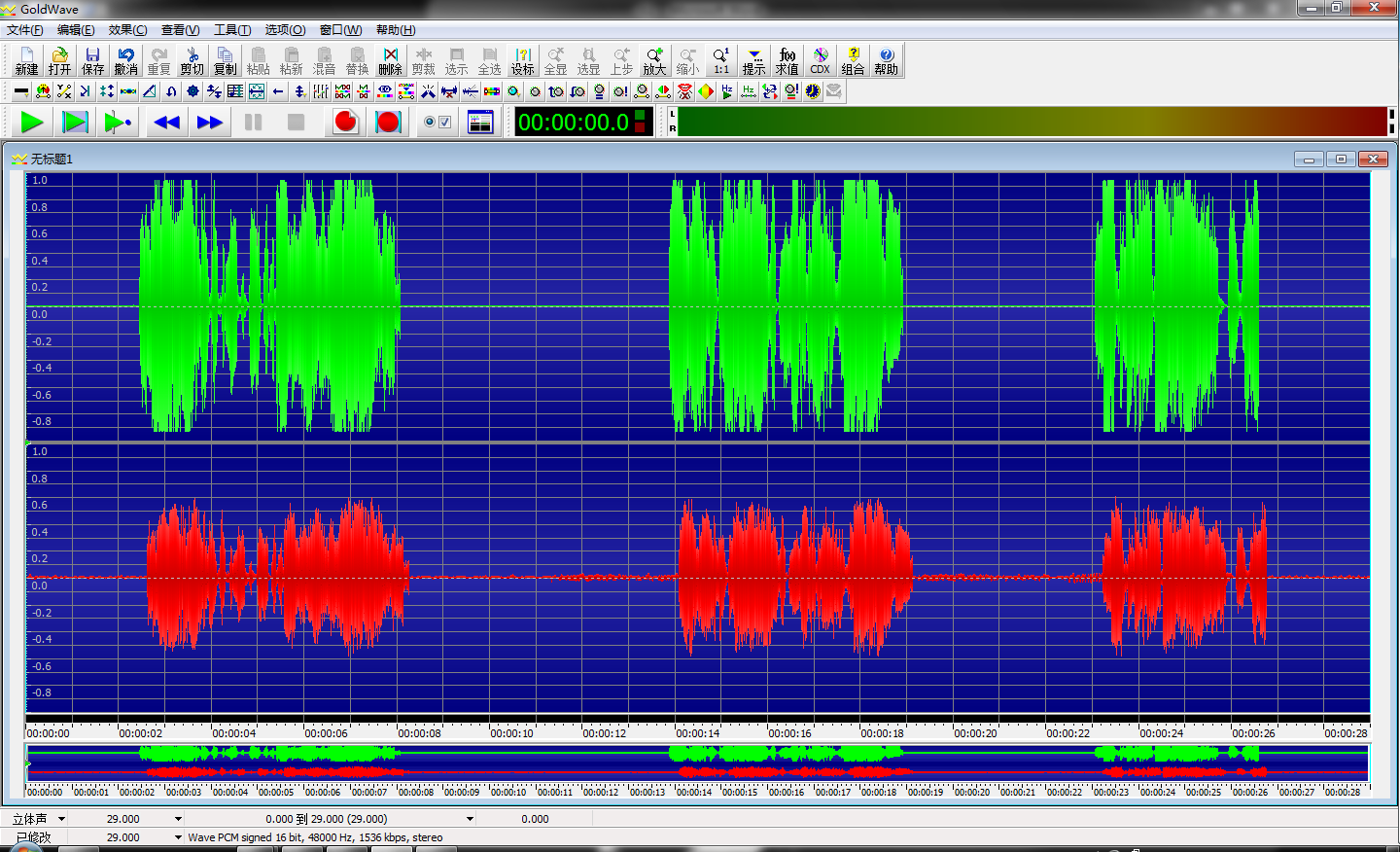
文件为test5.wav

## 25ms缓存延迟

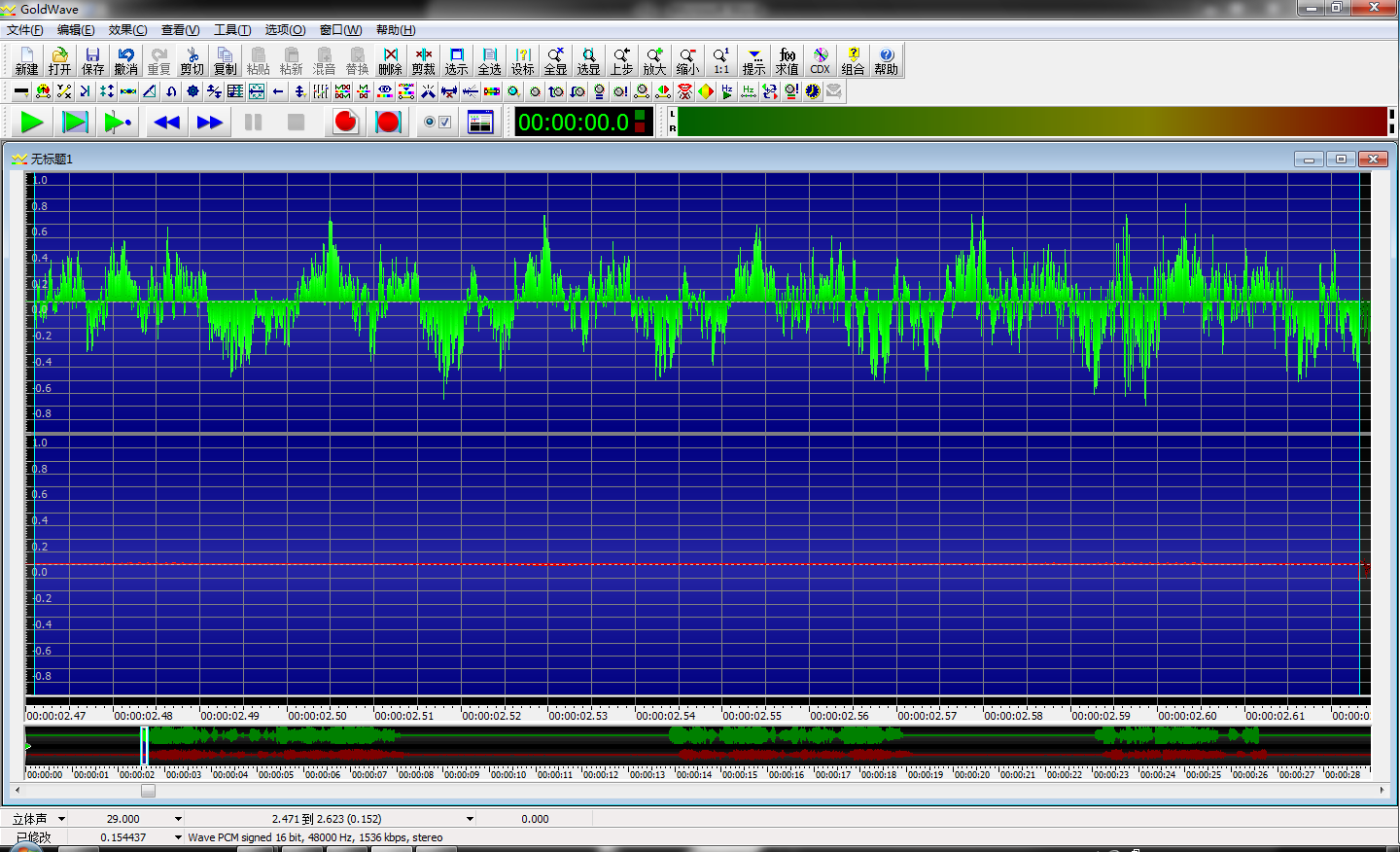
这次使用的配置为：

|  |  |  |
| --- | --- | --- |
| cbAlign | 1 | 未知 |
| cbBuffer | 4410 | 缓存大小 |
| cbPrefix | 0 | 未知 |
| cBuffers | 4 | 未知 |

最后得到的设置缓存大小为4408，应该是由于内存对齐的问题。本次测试的输出波形为：



最终延迟为150ms





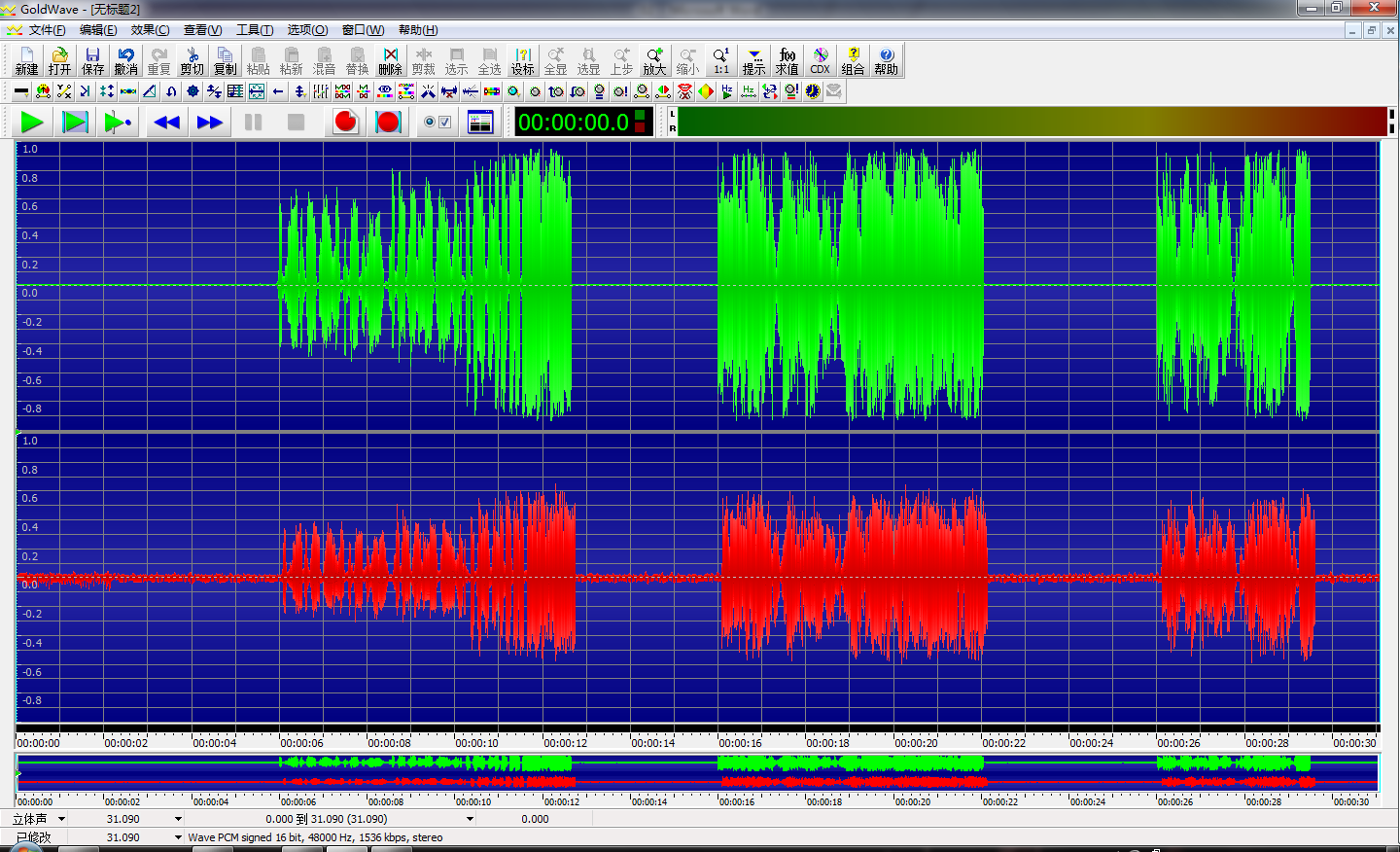
测量文件最后为test6.wav

## 回返延迟测量后有如下的关系

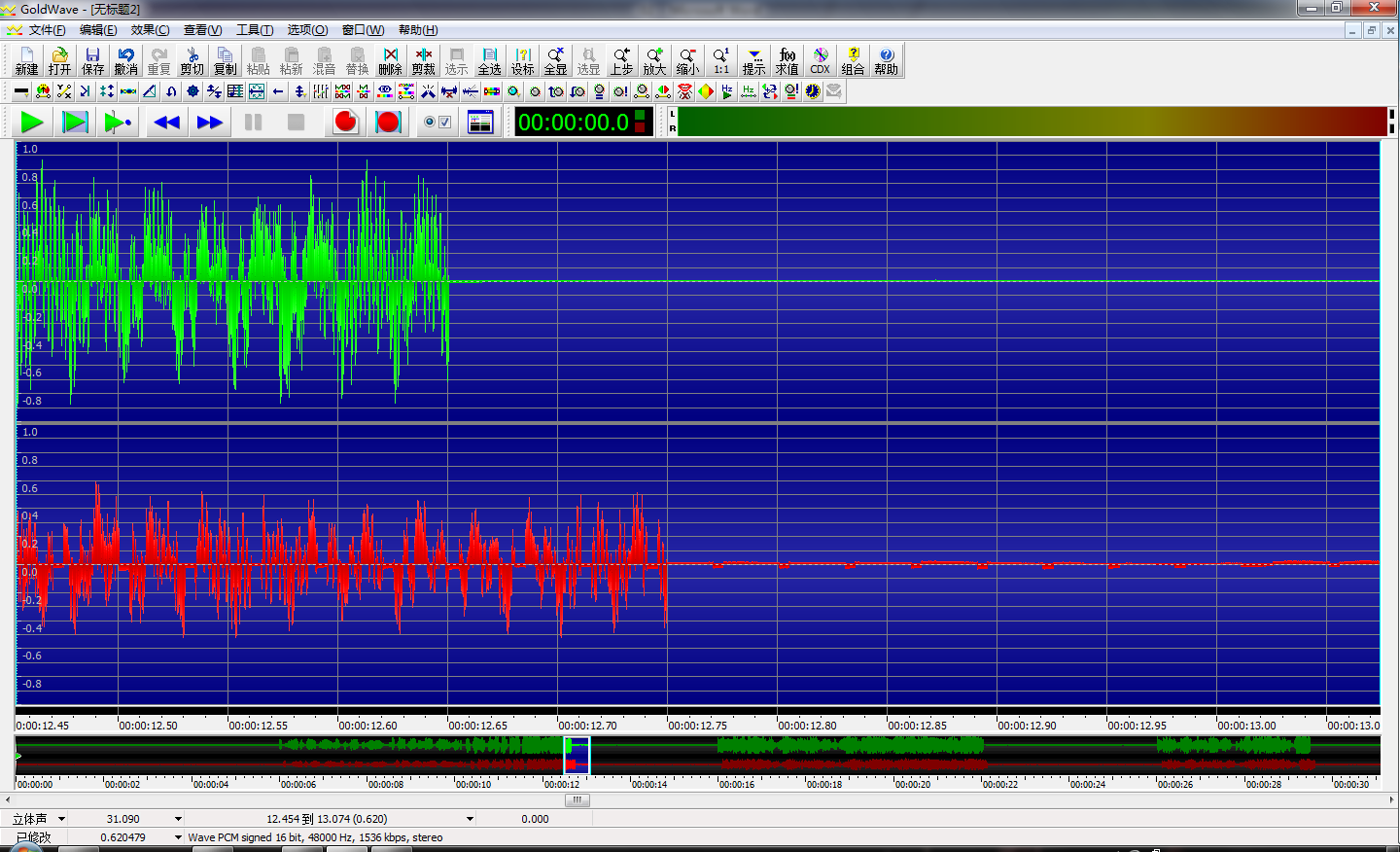
# 对于缓存块个数的研究

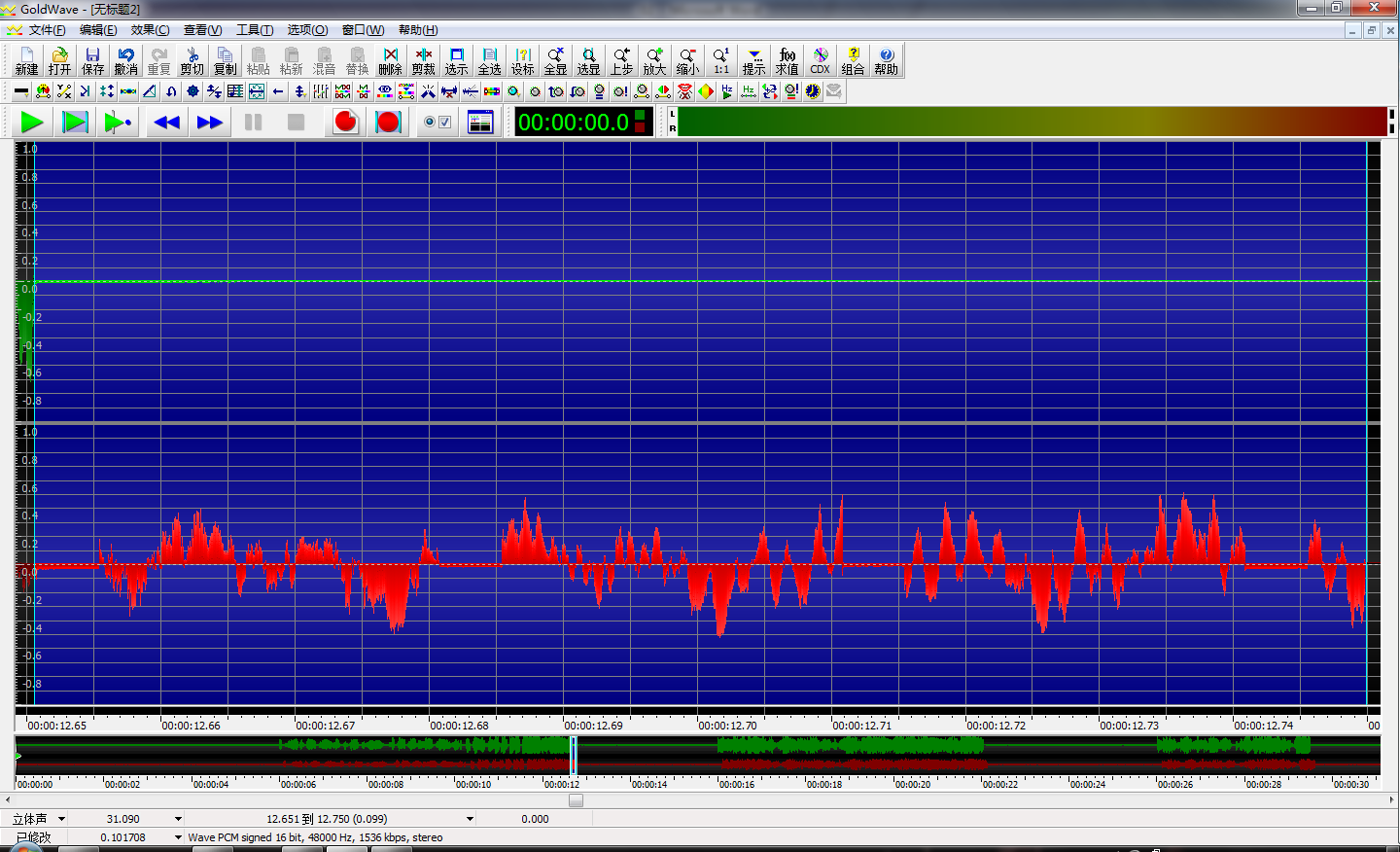
在缺省情况下，缓存块格式为4块。

改成1块之后，音质明显受损。之后进行具体的测量，看在1个缓存块的情况小到底延迟的表现能有多少。



延迟有所改善，但是音质损失严重







延迟为99ms。

文件为test7.wav。