

# VCB-Studio 教程 08 章节的处理与 BDMV 的分割

本教程旨在讲述 BDRip 压制中，对章节文件的分割，根据章节信息算切割帧数，以及如何操作来分割多集连在一起的 m2ts 文件。

## 0. 帧数标号，和 avs/vs 切割写法简述

本段其实来自之前的 VS/AVS 入门教程，但是教程 08 的设计是，读者并不需要完全了解这两篇所有详细内容，所以单独把这段提出来。已经详细阅读过 avs/vs 入门教程的可以跳过。

**以下是整个 vcb-s 教程体系中，我们对帧数标号的规定：**

在绝大多数场合下（除了 mkvtoolnix），视频的帧数是从 0 开始标号的。简单说，如果一个视频有 1000 帧，那么所有帧的标号为：

0, 1, 2...999

mkvtoolnix 是从 1 开始标号的: 1, 2, 3...1000。然而，除非指定了是 mkvtoolnix，任何讨论都假设帧数从 0 开始标号。

无论从 0 还是 1 开始标号，总帧数=末号-首号+1

如果我们说从 a 帧到 b 帧，我们默认是包括首尾的。比如 20-100 帧，就是 20,21,...99,100 帧，一共是  $100-20+1=81$  帧。

vs 中切割的写法：

```
src8 = core.std.Trim(src8,20,100)
```

avs 中切割的用法：

```
src8 = trim(src8, 20,100)
```

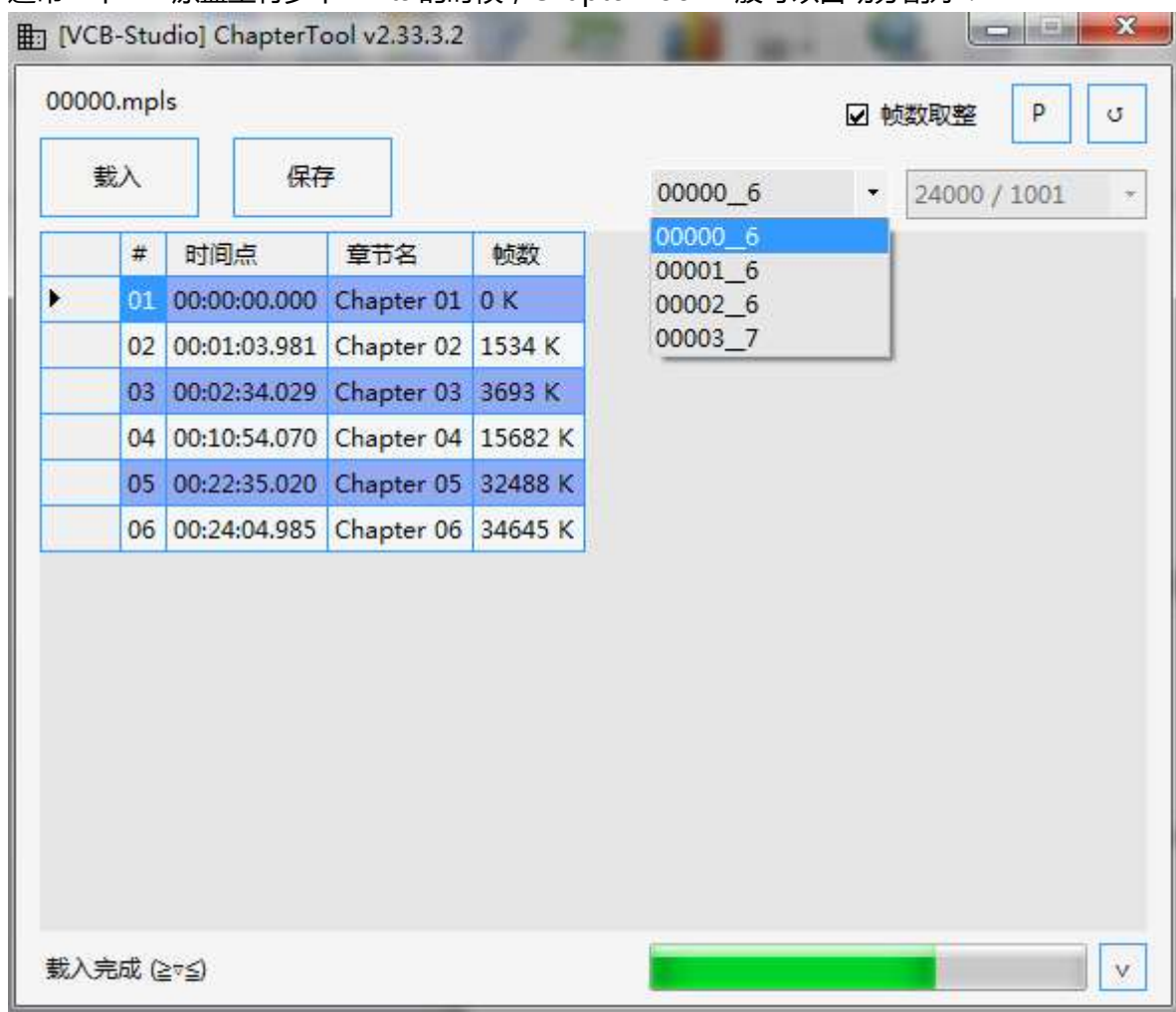
或者类似

```
trim(20,100)
```

```
src8 = last
```

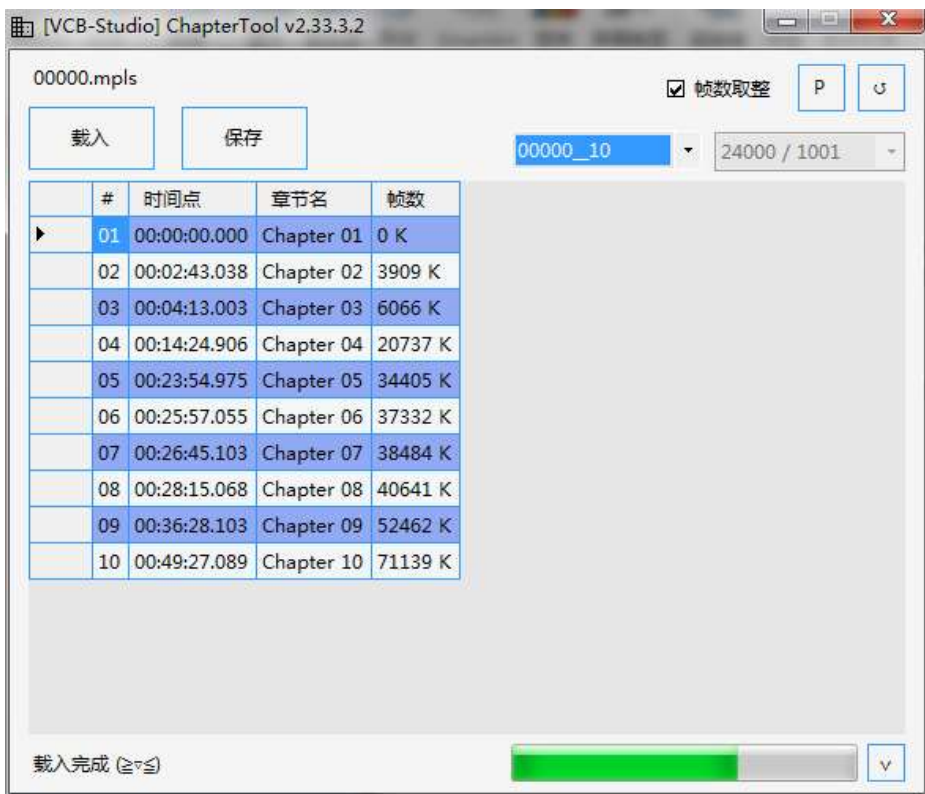
# 1. 章节的分割方法

通常一个 BD 原盘里有多 m2ts 的时候，Chapter Tool 一般可以自动分割好：

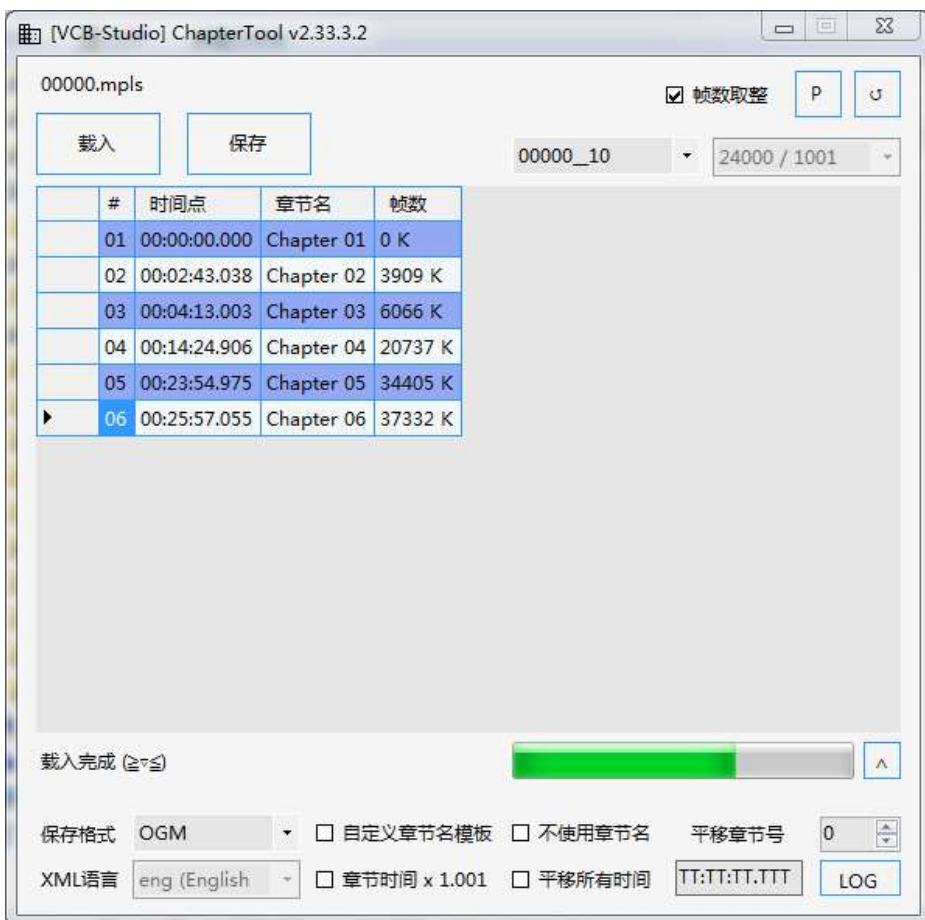


选择右上方的下拉框，就可以选择不同的 m2ts 标号（还有个数字表示章节的数量）  
选择好对应的保存就可以了。

但是，如果一个 BD 里面多集是连在一个 m2ts 文件的，情况就复杂了。典型的比如说化物语(Bakemonogatari)的 Vol.1，两集是连在 00000.m2ts 里面的。第一集的长度大约是 25 分 57 秒（这个可以通过播放 m2ts，人为的观察剧情分段来确定）。如果我们用 Chapter Tool 载入它的章节，效果如下：



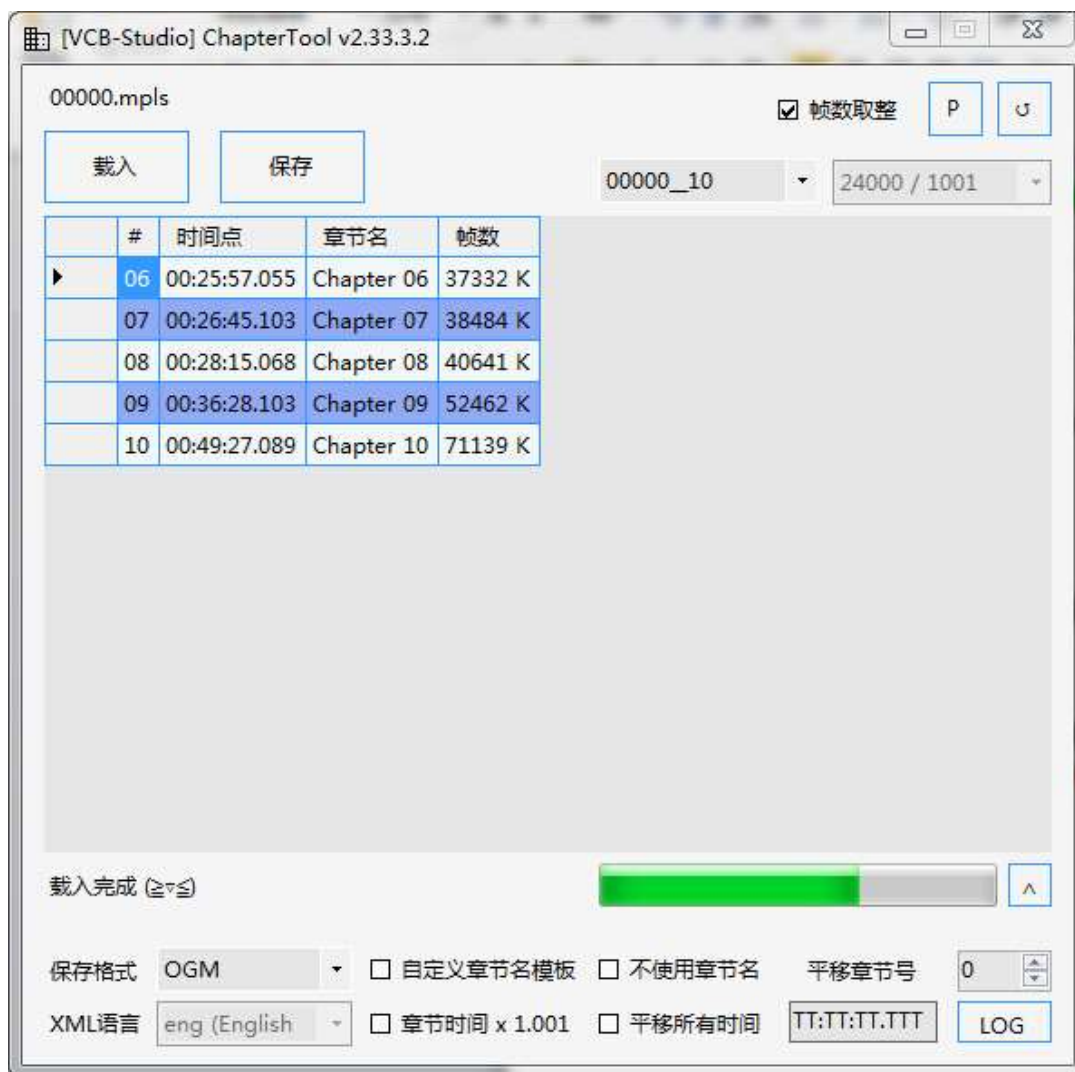
Chapter Tool 表格区域最左边的空格，是让你单击选中的。通过 ctrl 键多选，可以删掉不需要的部分。比如说 07 以后的部分都不需要了，我们可以选择 07-10，然后按键盘上的 delete 键删除：



然后就可以点击保存。注意，你完全可以连 06 一起删除。最末一个章节信息带不带在这里不影响。如此出来的就是第一集的章节信息文件。

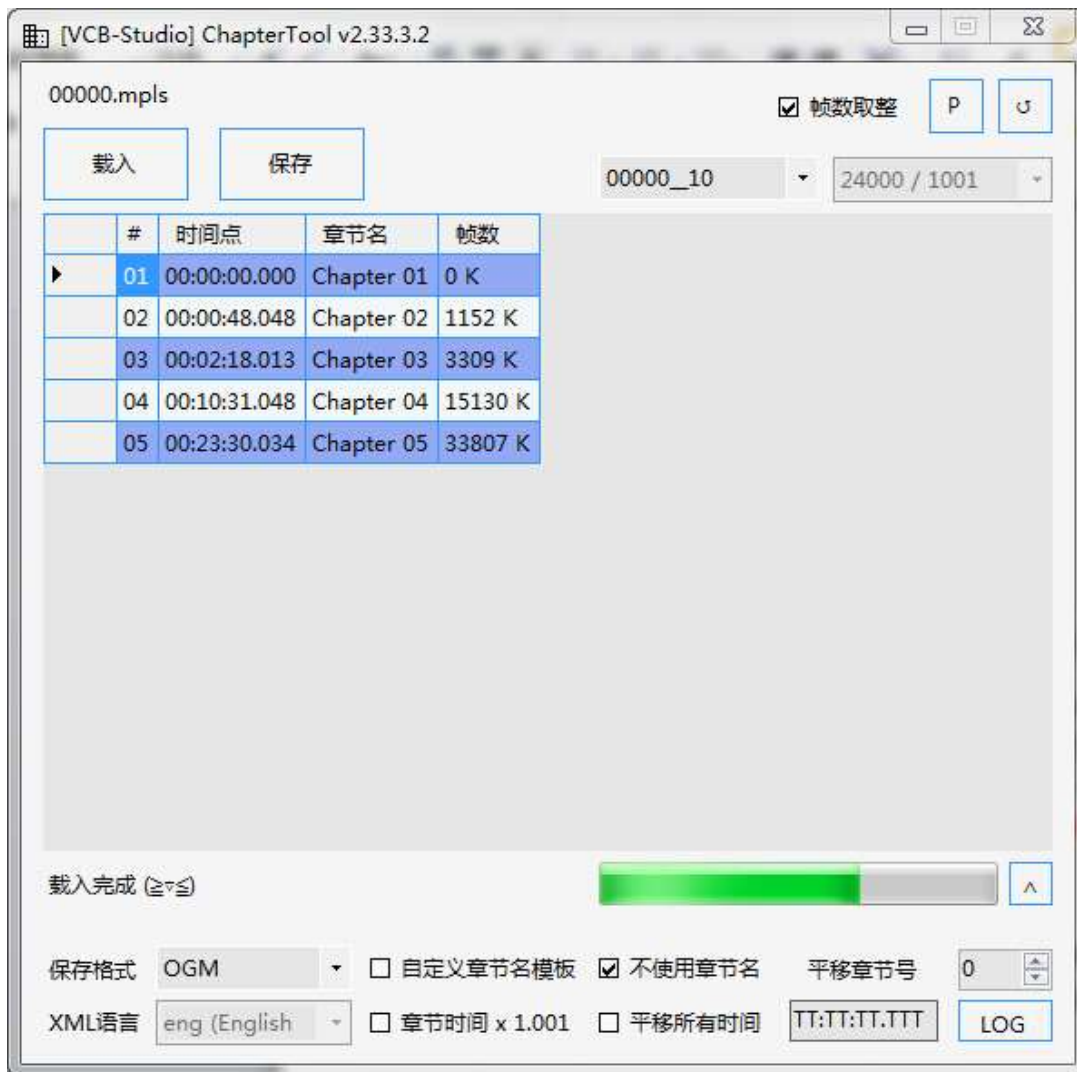
要抽取第二集的章节信息，我们就要删除掉第一集的内容，但是必须注意的是：

不要把第一集结束的时间戳（也是第二集开始的时间戳）给删除：



如图所示，删完了之后，开头的的时间戳应该是第一集结束的时间点。这个跟之前做第一集章节可留可不留不同，做第二集章节的时候必须予以保留。

切完了之后，其实 Chapter Tool 已经自动平移好时间轴，所有时间减去开头的时间（这就是为什么开头的时间不能弄错），点击右上角刷新按钮可以看到更新。但是章节名称依旧是 Chapter 06, Chapter 07.....这里可以勾选下方不适用章节名来重新标号。弄好之后刷新一下确认：



确认了之后，就可以点击保存，这样第二集的章节就做完了。

如果还有第三集，第四集，如法炮制：

1. 删掉不需要的章节，然而任何一集开头的时戳必须保留；
2. 选择不使用章节名平移章节名称标号。

## 2. 帧数的计算方法和视频切割

视频制作中，经常需要对视频按照时间切割。比如说化物语第一集是从 00:00 切割到 25.57.055 ( 大概的时间可以看 m2ts 播放，精确的时间就必须从章节信息中获取 )，第二集是从 25.57.055 切割到结束。原盘帧率恒定为 24000/1001fps ( 一般原盘时间小于 5 个小时的话，你可以用 23.976 近似代替 )

我们应该怎么把时间和帧率转换为切割用的帧数呢？

第一集的切割比较好理解：第一集总共是 25.57.055 秒，这意味着它有  $25*60+57.055=1557.055$  秒，每一秒有 23.976 帧，那么第一集应该有： $1557.055*23.976=37331.95068\approx 37332$  帧。一般运算结果不会是整数，因为章节显示时间只能是小数点后 3 位，且帧率我们用了近似。但是出来的结果应该非常非常接近整数，这样你可以四舍五入到整数。如果出来的结果离整数差距较远，那么你就要当心，很可能是算错了。

( 这里提一下为什么上文是 5 个小时内可以用近似：四舍五入的时候产生 1 帧的偏差，意味着结算结果有 0.5 的误差，那么  $0.5/(24000/1001-23.976)\approx 20854$  秒，大概是接近 6 小时。所以 24p 原盘长度小于 5 小时，可以用近似值代替。同理，30000/1001 fps 的源在 4 小时内可以用近似，60000/1001 fps 的源在 2 个半小时内可以用近似，因此 60000/1001 帧的视频建议算的时候用精确值，否则容易造成偏差 )

第一集的视频有 37332 帧，那么 trim 的时候就要写 trim(0,37331)。后面数字-1 的原因是，trim 的帧数是从 0 开始算起，总数是 N，标号是 0~N-1。第一集视频在原盘中，从第 0 帧开始，到第 37331 帧结束，包括首末。

第二集的视频，开头的帧数是原盘的 37332 帧，结尾就是原盘结尾。所以 Trim 可以这么写：

trim(37332, 0) # avs 中，第二个数字是 0 表示直接切割到结尾；

trim(src8, 37332, src8.num\_frames-1) # vs 中，可以调用 num\_frames 这个 clip property，返回总帧数。记好了总帧数是 N，最后一帧的标号是 N-1

如果我们想切一个第一集的 OP 怎么办呢？

1. 先从片源观察，OP 大约是 2 分 43 秒到 4 分 13 秒；
2. 在章节中找到对应的精确时间戳，2:43.038 到 4:13.003
3. 从开头到 OP 开始之前的开场白一共是  $(2*60+43.038)*23.976\approx 3909$  帧，意味着开场白最后一帧是 3908 帧，OP 开头是 3909 帧；
4. 从开头到 OP 结束，一共是  $(4*60+13.003)*23.976\approx 6066$  帧，意味着 OP 最后一帧是 6065 帧。
5. 可以知道 OP 的段落是 trim(3909, 6065)

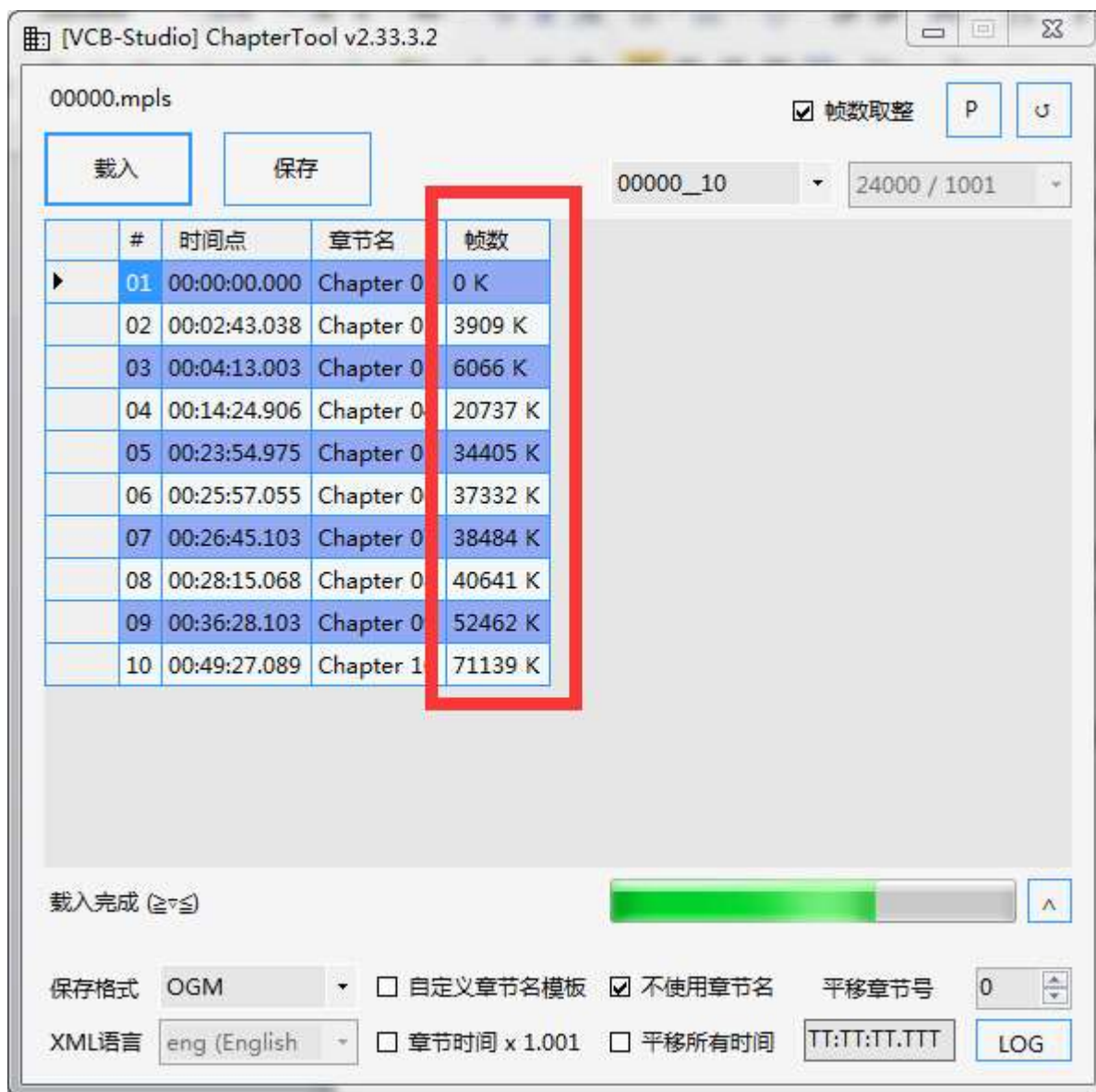
总结一下，开头和结尾两个时间点，分别转为秒数，乘以帧率，转为帧数之后，**开头的数字不用动，结尾的数字需要-1**。这个总结同样适用整集切割，比如说第一集，00:00-25.57.055，开头的时戳乘以帧率=0，结尾的时戳乘以帧率=37332，因此写法是 Trim(0,37331)。而第二集的开头时间是 25.57.055，转换为帧数之后，就无需-1 了。

这个做法简化为口诀是：**秒数乘帧率，后减前不减**。这个计算是非常严格的，帧数计算，以及是否要-1，这是必须谨慎对待的。算错/填错帧率导致出来的 BDRip 音画不同步/字幕不同步的例子比比皆是。



曾经自己写过一个小工具帮自己计算，time\_cal.exe（见附件）。这个工具可以在指定帧数和输入时间之后，计算出总帧数。比如指定帧率是 24000/1001，时间是 00:02:43.038，它会返回 3909。如果计算结果跟整数出入较大，它会给出警告。

Chapter Tool 同样有这样的功能，对于载入的章节信息，自动获取片源帧率，然后根据章节时间，算出整数：



在切割的时候就可以用这些帧数帮你忙。比如说切第一集 OP，很明显，给出的帧数是 3909 和 6066。根据后减前不减的法则，Trim 的写法为：Trim(3909,6065)。

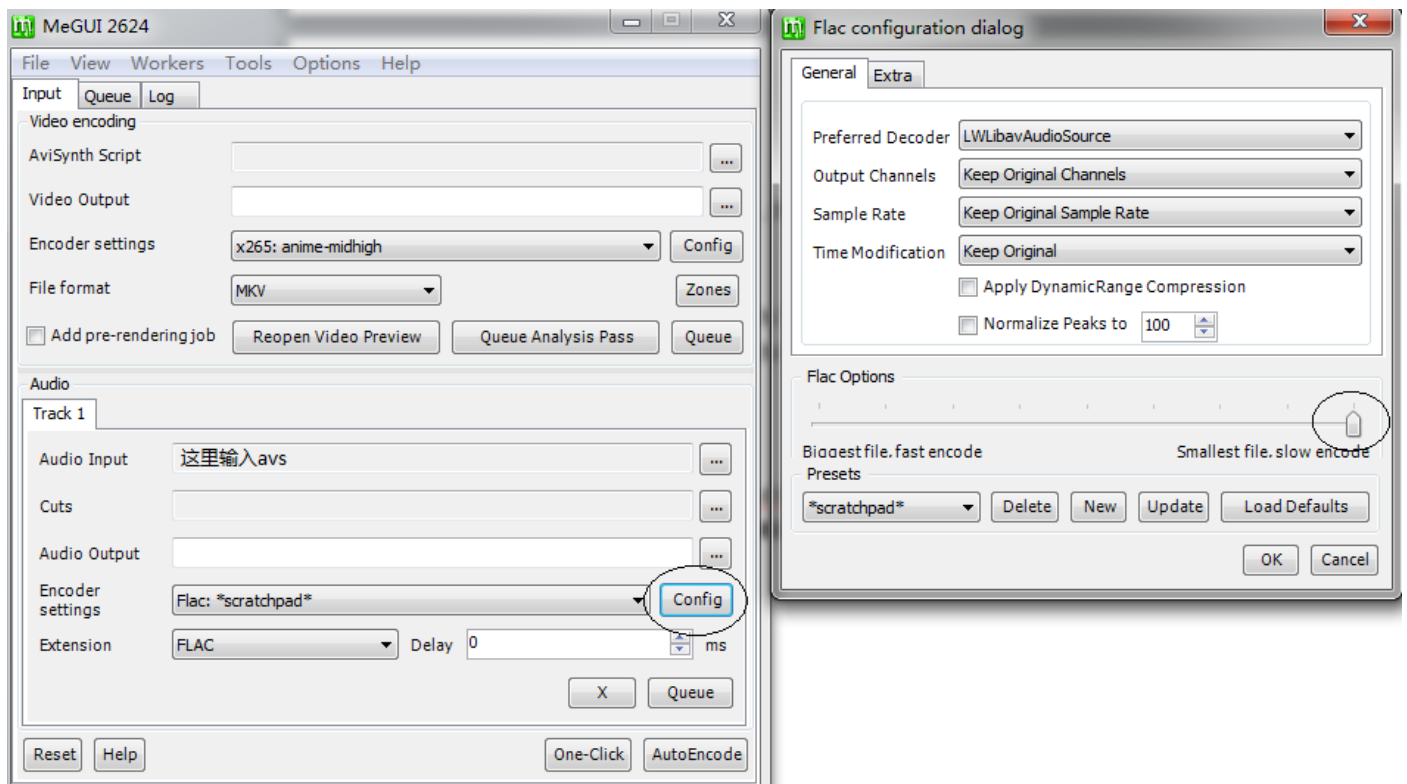
**注意切割到文件最后不能用这里面的数据，因为很可能文件结尾并没有对应的章节点。**比如说化物语第一卷 00000.m2ts 总长度为 51 分钟多，而章节里面最后的时间点仅为 49 分半。如果你需要知道结束的帧标号，你可以在 avs 中用 `src8.frameCount()-1`，vs 中会用 `src8.num_frames-1` 来获取，或者自己根据总时长计算。

### 3. 音频的切割方法

视频可以通过 Trim 在 avs/vs 脚本中做切割，然后压制，音频也是一样的原理。比较方便的是通过 avs 来实现，比如说我们想切割出化物语第二集的音频，你可以自己写一个 avs 脚本：

```
video = LWLibavVideoSource("00000.m2ts",threads=1)
audio = LWLibavAudioSource("00000.m2ts")
audiodub(video,audio)
trim(37332,0)
```

这样就是分别读取原盘的视频和音频，用 audiodub 联合起来，再按帧数进行切割。切的时候音频就被一并切了。然后这个 avs 可以直接丢给 MeGUI 的音频编码区域：



如上图，queue 之后执行，MeGUI 就会将 avs 输出的音频部分，用 flac 转码为无损编码，如果需要 aac 音轨，则选用 qaac，设置界面，一般 vcb-s 用 Contrained VBR 128Kbps。

如果有多条音轨，一般可以在 LWLibavAudioSource 中 用 stream\_index 指定。一般来说 ,BDMV 的 stream\_index 就是 mediainfo 中从上往下的顺序，0 是视频，1 是主音轨，2 是副音轨，3 是第二副音轨.....注意这样的标号并非永远成立；你需要在抽取之后听一下以作检验。如果不对，可以多试试其他标号，或者先把音轨抽取出来，用 lwas 单独载入 flac 音轨。

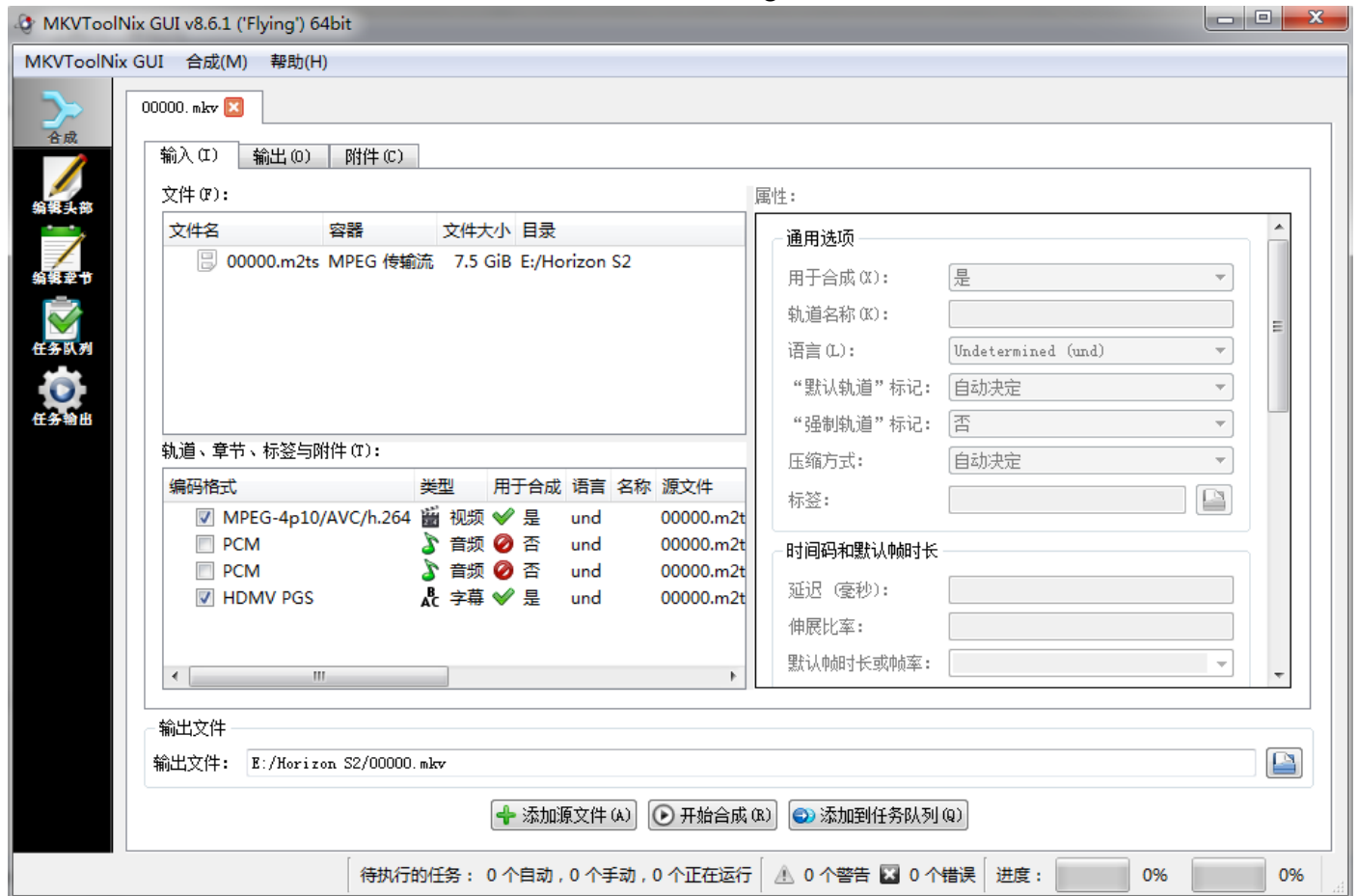
所以抽取副音轨的方法

```
video = LWLibavVideoSource("00000.m2ts",threads=1)
audio = LWLibavAudioSource("00000.m2ts",stream_index=2)
audiodub(video,audio)
trim(37332,0)
```

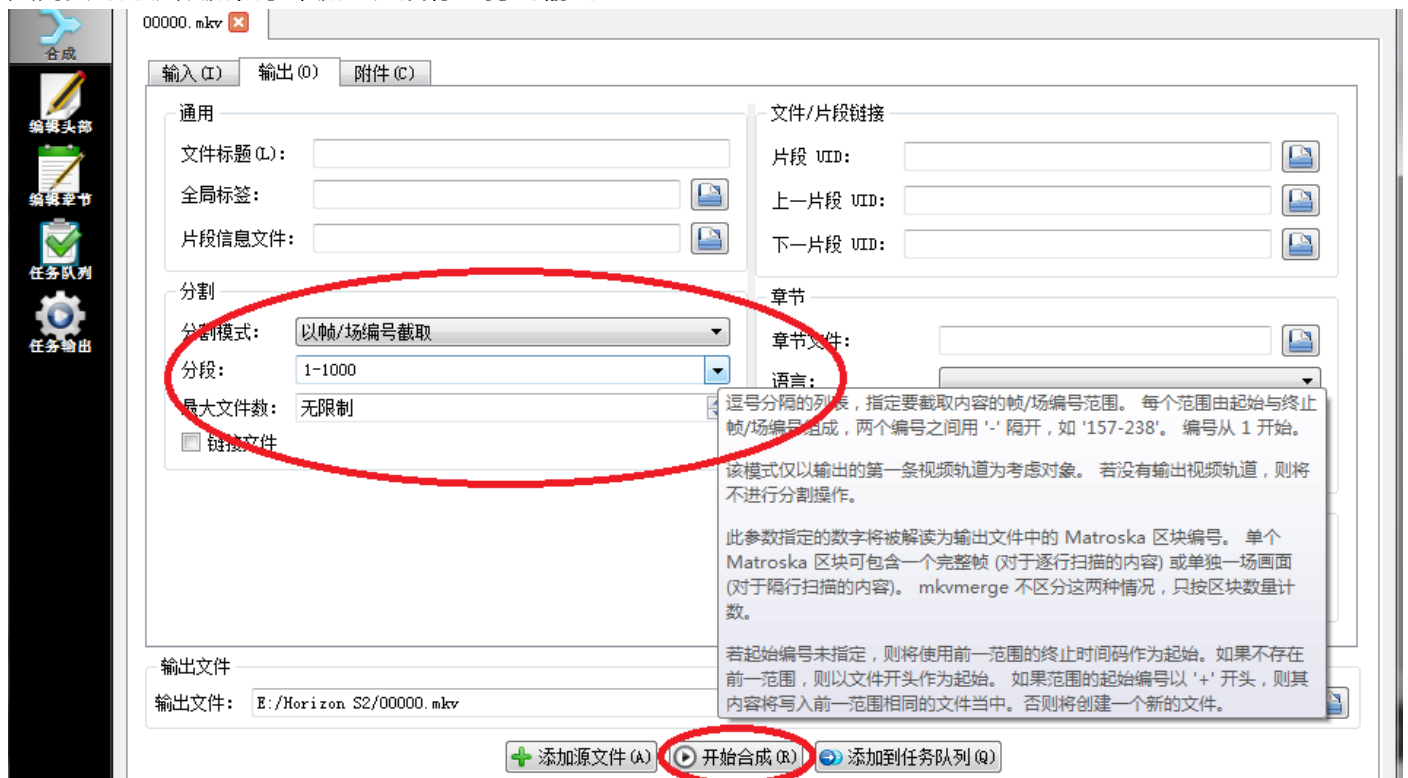


## 4. 字幕的切割方法

有时候，原盘还有 PGS 字幕，这在 BDRip 中也是需要切割的。不幸的是切割字幕可没法用音频的手段，得借助其他的工具，最常用的就是 MKVtoolnix。我们先把原盘拖入 mmg:

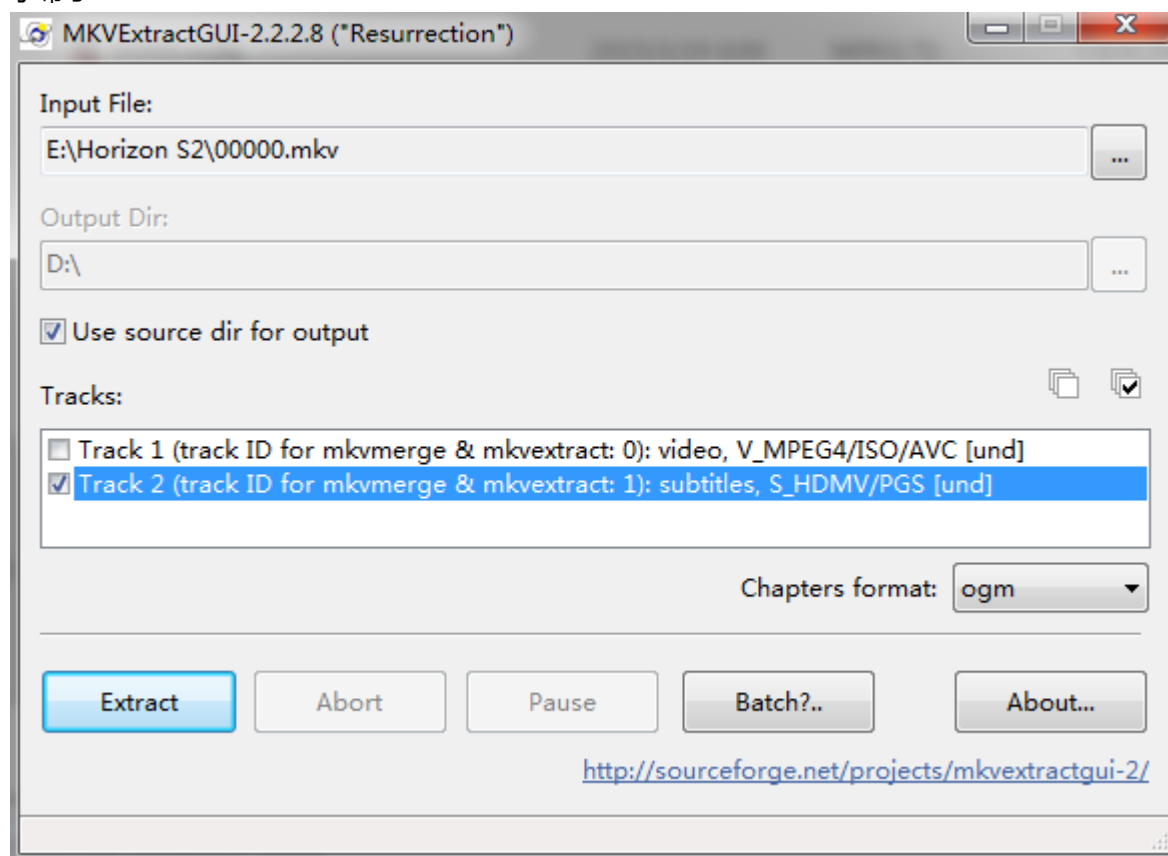


只需要勾选视频流和字幕流。然后标签打到输出:



需要注意的是，这是大概你唯一碰到的，帧数从 1 开始标号的地方。所以如果我们要切割出前 1000 帧，avs/vs 里面 trim 写法是 trim(0,999)，这里的写法是 1-1000，前后标号都需要+1。

设置好了之后，点击开始合成，mkvtoolnix 会输出分割后的，带字幕的 mkv。然后你就可以用 MKVExtract 抽取字幕了：



需要注意的是，这里**不能**使用 MeGUI 的 HD Stream Extractor (使用 eac3to)。原因未知，结果是抽取后的字幕轨道为空，大小不足 1MB，封装后没有任何字幕显示。

## 5. 复杂原盘的处理方法

以上我们介绍了原盘多集在一个 m2ts 当中的时候,如何根据章节信息确定精确的时间轴,并对章节、视频、音频、字幕进行分割处理。

部分时候,原盘组织格式,可能比你想象的要麻烦。比如中二病是 2 集在 3 个 m2ts 里,而且分界点是在第二个 m2ts 内。这时候遵从的法则是:先合并,再分割。

如果有 pgs 字幕,你可以用 mkvtoolnix 把所有的 m2ts 追加合并为一个 MKV,带音轨带字幕,然后当做一个 m2ts 去做。

如果没有 pgs 字幕,你可以在 avs/vs 中,通过读入多个 clip 合并为一个,再写 trim 的方式,来组织对应的视频和音频部分。