

## 转 vc2005编译ffmpeg以及ffplay

2013年09月25日 21:35:02 阅读数：7018

### ffmpeg编译过程：

1 <http://ffmpeg.zeranoe.com/builds/>

下载官方提供的源码，win32库和dll。

2 新建vc2005 console空工程，把ffmpeg.h,ffmpeg.c,cmdutils.c,cmdutils.h,cmdutils\_common\_opts.h,ffmpeg\_filter.c,ffmpeg\_opt.c加到工程。

3 [http://drv.nu/ffmpeg/dxygen/config\\_8h-source.html](http://drv.nu/ffmpeg/dxygen/config_8h-source.html)

从这里生成 config.h

4 从mingw下复制stdint.h和inttypes.h

5 把.h里的inline换成\_\_inline

或

```
[cpp]
1.  #if defined(WIN32) && !defined(__cplusplus)
2.  #define inline __inline
3.  #endif
```

或

project->[setting]->[c/c++]->Preprocessor definitions:编辑框里输入inline=\_\_inline

6 找不到colospace.h等一些文件，从源码里找到复制过来。

7 有几个系统文件找不到，注释掉config.h相关定义，如

```
[cpp]
1.  // #define HAVE_SYS_RESOURCE_H 1
```

另外，需要用到socket，修改：

```
[cpp]
1.  #define HAVE_WINSOCK2_H 1
2.  #define HAVE_STRUCT_ADDRINFO 1
```

7 print\_all\_libs\_info 里找不到什么东西，就注释掉。

8参考 <http://www.cnblogs.com/zy1910/archive/2012/08/08/c99int.html>

在config.h里面加上

```
#define PRIu64      "I64u"
```

```
#define PRId64      "I64d"
```

9

```
// #define AV_TIME_BASE_Q      (AVRational){1, AV_TIME_BASE}
```

改成

```
#define AV_TIME_BASE_Q      _AVRational(1, AV_TIME_BASE)
```

再加：

```
[cpp]
1.  AVRational _AVRational(int num, int den) {
2.  AVRational r = {num, den};
3.  return r;
4.  }
```



或者，在用AV\_TIME\_BASE\_Q 的地方，手写AVRational r = {1,AV\_TIME\_BASE };，再用上这个r。

最麻烦就是const OptionDef options[] 这个数组的初值处理，主要是里面的函数，参考ffmpeg\_vc5的代码，把每一行改成这样：{ "f", HAS\_ARG | OPT\_STRING | OPT\_OFFSET, { (void\*)OFFSET(format) },

就是把原来函数的地方{.fun = xxx}改成{ (void\*) xxx}。编译通过了，常规的一些输入参数都好用。复杂的还没试。

最后会提示几个链接问题，手动加上const AVPixFmtDescriptor av\_pix\_fmt\_descriptors[PIX\_FMT\_NB];

然后，在config.h里加：

```
[cpp]    
1. #define AVCONV_DATADIR "c:/ffmpeg1010/"  
2.  
3.  
4. #define snprintf _snprintf  
5. #define lrint(f) (f>=0?(int32_t)(f+(double)0.5):(int32_t)(f-(double)0.4))  
6. #define lrintf(f) (f>=0?(int32_t)(f+(float)0.5):(int32_t)(f-(float)0.4))  
7. #define llrintf(f) (f>=0?(int64_t)(f+(double)0.5):(int64_t)(f-(double)0.4))  
8. #define llrint(f) (f>=0?(int64_t)(f+(float)0.5):(int64_t)(f-(float)0.4))  
9.  
10.  
11. #define log2(x) (log(x) * 1.44269504088896340736)
```

这个AVCONV\_DATADIR 是随便写的，暂时还没用上。

到此，基本就完全成功了。编译出来，运行ffmpeg -i rtsp://xxx/test.264 out.mp4  
保存的质量非常好。

## ffplay编译过程

ffplay编译比ffmpeg简单的多，新建工程，加上代码，稍改几行就OK了。然后ffplay rtsp，出了一个黑窗口，播放效果不错。再再生成一个mfc工程，给sdl指定一个窗口，视频就在指定窗口播放了。不过这时有个问题，拖动窗口时，视频就花掉了。估计sdl需要一些参数，以后有空再研究了。

源码下载：<http://download.csdn.net/detail/sxcong/4672795>

原文地址：<http://blog.csdn.net/sxcong/article/details/8097594>

文章标签：[vc](#) [ffplay](#) [编译](#)

个人分类：[FFMPEG](#)

所属专栏：[FFmpeg](#)

此PDF由spygg生成,请尊重原作者版权!!!

我的邮箱:liushidc@163.com