

## 原 最简单的基于FFmpeg的移动端例子：IOS 视频转码器

2015年08月01日 16:34:20 阅读数：19307

=====

最简单的基于FFmpeg的移动端例子系列文章列表：

[最简单的基于FFmpeg的移动端例子：Android HelloWorld](#)

[最简单的基于FFmpeg的移动端例子：Android 视频解码器](#)

[最简单的基于FFmpeg的移动端例子：Android 视频解码器-单个库版](#)

[最简单的基于FFmpeg的移动端例子：Android 推流器](#)

[最简单的基于FFmpeg的移动端例子：Android 视频转码器](#)

[最简单的基于FFmpeg的移动端例子附件：Android 自带播放器](#)

[最简单的基于FFmpeg的移动端例子附件：SDL Android HelloWorld](#)

[最简单的基于FFmpeg的移动端例子：IOS HelloWorld](#)

[最简单的基于FFmpeg的移动端例子：IOS 视频解码器](#)

[最简单的基于FFmpeg的移动端例子：IOS 推流器](#)

[最简单的基于FFmpeg的移动端例子：IOS 视频转码器](#)

[最简单的基于FFmpeg的移动端例子附件：IOS自带播放器](#)

[最简单的基于FFmpeg的移动端例子：Windows Phone HelloWorld](#)

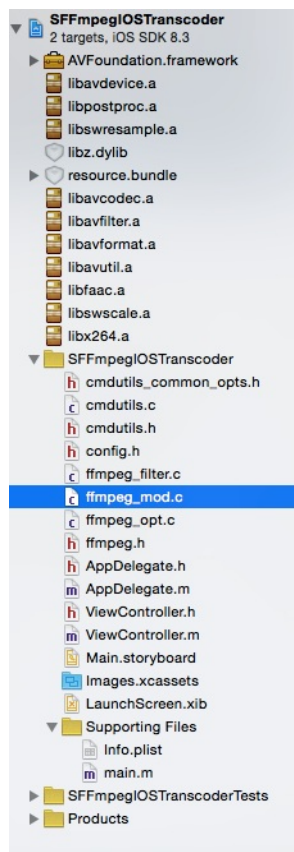
=====

本文记录IOS平台下基于FFmpeg的视频转码器。该转码器实际上移植自ffmpeg工程中的ffmpeg.c源代码。有关ffmpeg.c的源代码可以参考文章[《ffmpeg.c函数结构简单分析（画图）》](#)，在这里就不重复记录了。



## 源代码

项目的目录结构如图所示。



下列C语言文件拷贝自FFmpeg源代码：

*cmdutils.c*  
*cmdutils.h*  
*cmdutils\_common\_opts.h*  
*config.h*  
*ffmpeg.h*  
*ffmpeg\_filter.c*  
*ffmpeg\_opt.c*

此外在编译ffmpeg.c的时候，除了需要FFmpeg SDK中的头文件之外，还需要从源代码中拷贝下面头文件。

*libavformat/ffmpeg.h*  
*libavformat/network.h*  
*libavformat/os\_support.h*  
*libavformat/url.h*  
*libavutil/libm.h*

ffmpeg\_mod.c是修改后的ffmpeg.c文件。ffmpeg.c本身是用于命令行使用的程序，因此在改为类库调用的时候需要对源代码做一些修改。例如去掉一些exit(0)，释放一些变量，main()函数改名为ffmpegmain()等。

Objective-C调用FFmpeg的代码位于ViewController.m文件中，内容如下所示。

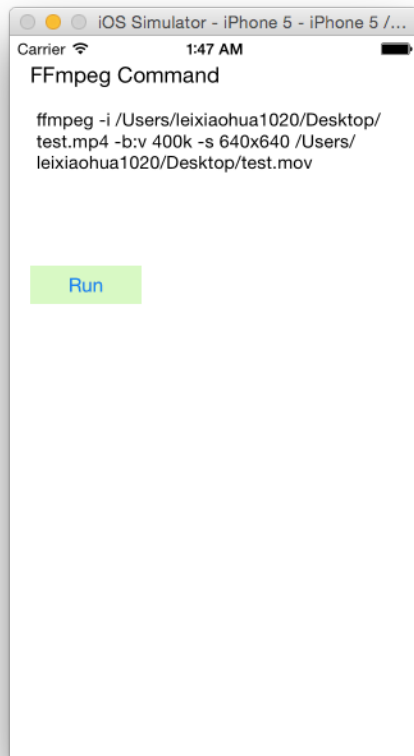
```

1.  /**
2.   * 最简单的基于FFmpeg的转码器-IOS
3.   * Simplest FFmpeg IOS Transcoder
4.   *
5.   * 雷霄骅 Lei Xiaohua
6.   * leixiaohua1020@126.com
7.   * 马小雨 Ma Xiaoyu
8.   * maxiaoyucuc@163.com
9.   * 中国传媒大学/数字电视技术
10.  * Communication University of China / Digital TV Technology
11.  * http://blog.csdn.net/leixiaohua1020
12.  *
13.  * 本程序是IOS平台下的转码器。它移植于ffmpeg.c命令行工具。
14.  *
15.  * This software is a Transcoder in IOS. It is transplanted from ffmpeg.c
16.  * command line tools.
17.  */
18.
19. #import "ViewController.h"
20. #include "ffmpeg.h"
21.
22. int ffmpegmain(int argc, char**argv);
23.
24. @interface ViewController ()
25.
26. @end
27.
28. @implementation ViewController
29.
30. - (void)viewDidLoad {
31.     [super viewDidLoad];
32.     // Do any additional setup after loading the view, typically from a nib.
33. }
34.
35. - (void)didReceiveMemoryWarning {
36.     [super didReceiveMemoryWarning];
37.     // Dispose of any resources that can be recreated.
38. }
39.
40. - (IBAction)clickRunButton:(id)sender {
41.     char command_str_full[1024]={0};
42.
43.
44.     NSString *command_str= [NSString stringWithFormat:@"%s",self.command.text];
45.     NSArray *argv_array=[command_str componentsSeparatedByString:@" "];
46.     int argc=argv_array.count;
47.     char** argv=(char**)malloc(sizeof(char*)*argc);
48.     for(int i=0;i<argc;i++)
49.     {
50.         argv[i]=(char*)malloc(sizeof(char)*1024);
51.         strcpy(argv[i],[argv_array objectAtIndex:i] UTF8String]);
52.     }
53.
54.     ffmpegmain(argc, argv);
55.
56.     for(int i=0;i<argc;i++)
57.         free(argv[i]);
58.     free(argv);
59. }
60. @end

```

## 运行结果

App在手机上运行后的结果如下图所示。



单击“Run”按钮之后，程序会根据输入的命令进行转码。例如对于上述截图的情况，就会将/Users/leixiaohua1020/Desktop/test.mp4转码为/Users/leixiaohua1020/Desktop/test.mov。生成的视频文件如下图所示。

## 下载

simplest ffmpeg mobile

### 项目主页

Github：[https://github.com/leixiaohua1020/simplest\\_ffmpeg\\_mobile](https://github.com/leixiaohua1020/simplest_ffmpeg_mobile)

开源中国：[https://git.oschina.net/leixiaohua1020/simplest\\_ffmpeg\\_mobile](https://git.oschina.net/leixiaohua1020/simplest_ffmpeg_mobile)

SourceForge：<https://sourceforge.net/projects/simplestffmpegmobile/>

CSDN工程下载地址：<http://download.csdn.net/detail/leixiaohua1020/8924391>

本解决方案包含了使用FFmpeg在移动端处理多媒体的各种例子：

[Android]

simplest\_android\_player: 基于安卓接口的视频播放器

simplest\_ffmpeg\_android\_helloworld: 安卓平台下基于FFmpeg的HelloWorld程序

simplest\_ffmpeg\_android\_decoder: 安卓平台下最简单的基于FFmpeg的视频解码器

simplest\_ffmpeg\_android\_decoder\_onelib: 安卓平台下最简单的基于FFmpeg的视频解码器-单库版

simplest\_ffmpeg\_android\_streamer: 安卓平台下最简单的基于FFmpeg的推流器

simplest\_ffmpeg\_android\_transcoder: 安卓平台下移植的FFmpeg命令行工具

simplest\_sdl\_android\_helloworld: 移植SDL到安卓平台的最简单程序

[IOS]

simplest\_ios\_player: 基于IOS接口的视频播放器

simplest\_ffmpeg\_ios\_helloworld: IOS平台下基于FFmpeg的HelloWorld程序

simplest\_ffmpeg\_ios\_decoder: IOS平台下最简单的基于FFmpeg的视频解码器

simplest\_ffmpeg\_ios\_streamer: IOS平台下最简单的基于FFmpeg的推流器

**simplest\_ffmpeg\_ios\_transcoder: IOS平台下移植的ffmpeg.c命令行工具**

simplest\_sdl\_ios\_helloworld: 移植SDL到IOS平台的最简单程序

版权声明：本文为博主原创文章，未经博主允许不得转载。<https://blog.csdn.net/leixiaohua1020/article/details/47072673>

文章标签：[FFmpeg](#) [视频](#) [转码](#) [移植](#) [IOS](#)

个人分类：[IOS多媒体](#) [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](mailto:liushidc@163.com)