



链滴

M1 Mac 原生编译 ffmpeg, 突破你的剪辑速度!

作者: [MingGH](#)

原文链接: <https://ld246.com/article/1663495096371>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

原文发布于: [M1 Mac原生编译ffmpeg, 突破你的剪辑速度!](#) , 欢迎关注获取最新更新

如果你的M1 Mac, 而且你也在使用ffmpeg, 这篇文章你可能需要好好看看。

通常我们在mac安装软件的时候都是通过homebrew进行安装, 但是对于M1 Mac来说, 如果你使用的是 [brew install ffmpeg](#) 安装的, 那么ffmpeg运行时会使用rosetta进行转义, 速度可以说是连微软的surface book都打不过。

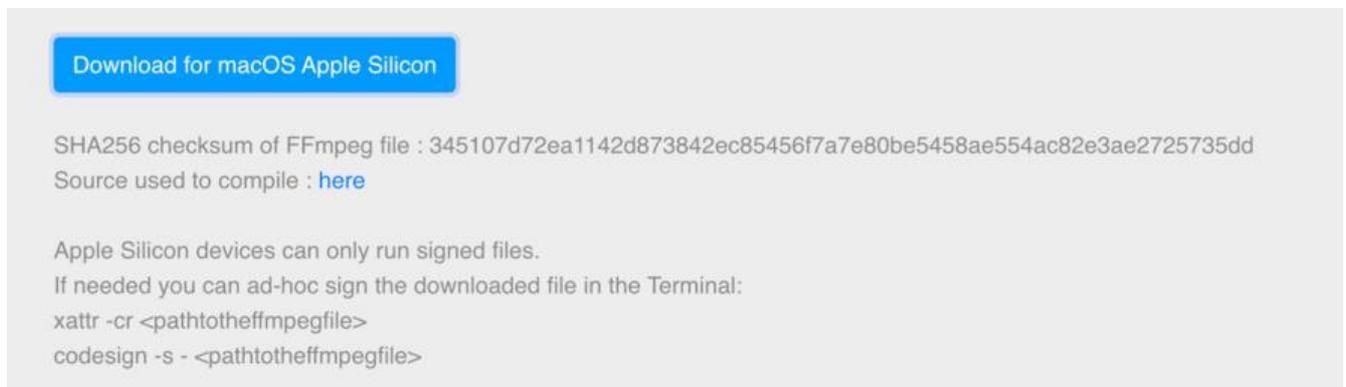
而最佳的解决办法就是, 使用NEON 扩展在你的M1 Mac上进行编译arm版本, 执行的速度比之前的度可以提高三倍以上, 真的不是开玩笑。

有两个办法可以让我们快速使用上M1 Mac下编译的ffmpeg

1. 使用已经编译好的ffmpeg

打开这个网址: [OSXExperts](#)

找到 [Download for macOS Apple Silicon](#) , 点击进行下载



如果打不开, 或者速度很慢, 可以直接点击这个[链接](#)进行下载, 我已经做了备份。

下载的这个ffmpeg是这个网址的大佬已经在他的电脑上编译好, 并且可以直接执行的二进制文件。

下载完成之后, 我们需要对文件进行一下SHA 256校验, 当然不校验也没关系。

校验的命令如下: ``

```
shasum -a 256 ffmpeg
```

得到结果: 345107d72ea1142d873842ec85456f7a7e80be5458ae554ac82e3ae2725735dd 表示证文件确实出自原始作者。

打开终端, 然后配置进你的Mac 环境变量,

```
vim ~/.zshrc
```

添加一行, 并加入PATH中

```
export FFMPEG_HOME=/Users/asher/workspace/software/ffmpeg
export PATH=$PATH:$MAVEN_HOME/bin:$FFMPEG_HOME:$JAVA_HOME/bin:$GRADLE_HOME/bin:.
```

关闭终端重新打开, 试试 `ffmpeg -version`, 可以看到已经有对应的输出。

```
ashher@Ashers-Mac Downloads % ffmpeg -version
ffmpeg version N-107981-g73fada029c Copyright (c) 2000-2022 the FFmpeg developer
s
built with Apple LLVM version 9.0.0 (clang-900.0.39.2)
configuration: --prefix=/Volumes/tempdisk/sw --extra-cflags=-fno-stack-check --a
rch=x86_64 --cc=/usr/bin/clang --enable-videtoolbox --enable-fontconfig --enabl
e-gpl --enable-libopenjpeg --enable-libopus --enable-libtheora --enable-libvorbi
s --enable-libmp3lame --enable-libass --enable-libfreetype --enable-libx264 --en
able-libx265 --enable-libvpx --enable-libwebp --enable-libaom --enable-libvidsta
b --enable-libsnpappy --enable-libzimg --enable-libsvtav1 --enable-version3 --pkg
-config-flags=--static --disable-ffplay --enable-postproc --enable-nonfree --ena
ble-runtime-cpudetect --disable-filter=yadif_videtoolbox --disable-indev=qtkit
--disable-indev=x11grab_xcb
libavutil      57. 35.100 / 57. 35.100
libavcodec     59. 42.103 / 59. 42.103
libavformat    59. 30.100 / 59. 30.100
libavdevice    59.  8.101 / 59.  8.101
libavfilter     8. 48.100 /  8. 48.100
libswscale      6.  8.104 /  6.  8.104
libswresample  4.  9.100 /  4.  9.100
libpostproc   56.  7.100 / 56.  7.100
ashher@Ashers-Mac Downloads %
```

无法打开?

如果无法打开, 可能是因为Apple Silicon只能运行已经签名的文件, 所以你需要在ffmpeg文件的目
执行以下两个命令。

```
xattr -cr ./ffmpeg
codesign -s - ./ffmpeg
```

此时用这个版本的ffmpeg去转码视频之类的, 比之前飞快。

2. 自行编译ffmpeg

这一部分适用于喜欢折腾的老哥使用, 也有算是相当简单的方式, 因为已经有大佬给我们做好了脚本。

克隆这个git仓库到本地: [ffmpeg-apple-arm64-build](#), 然后在编译前:

- 需要安装xcode, 可以直接从App Store中进行安装。安装之后记得先打开一下同意apple之类的协
。



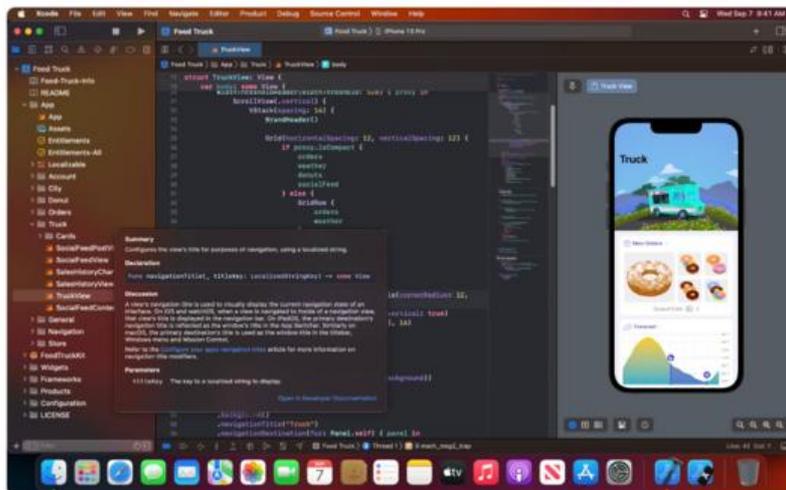
“xcode”的结果

Mac App iPhone与iPad App



Xcode
软件开发工具

打开



- 挂梯子，执行的时候会克隆很多外网的第三方依赖库
- 注册gitlab，并且添加sshkey 到你的账号，会有依赖库 [freetype](#)使用到

然后是执行部分，根据这个[ffmpeg-apple-arm64-build](#)的readme文档，推荐我们先随意创建一个目录，然后在这个目录中进行编译

```
mkdir ffmpeg-compile  
cd ffmpeg-compile
```

接着执行

```
../path/to/repository/build.sh
```

期间因为会克隆git仓库，速度取决于你的网速，也会有一些奇怪的错误抛出。当发生异常的时候需要删除这个目录，然后重新来一遍。。。祝你能编译成功！

参考内容

[FFMPEG on Apple Sillicon \(M1 Mac Mini 8Gb\)](#)

[M1 \(Mac ARM\) binary of FFMPEG?](#)