

Windows 下使用 NDK 编译 ISP 效果库文件

（公开文档，欢迎转发）

版 本 号：	V1.5
B M 等 级：	<input type="checkbox"/> MM <input type="checkbox"/> JM <input type="checkbox"/> JUM <input checked="" type="checkbox"/> 公开
编 制：	
审 核：	

目录

1	NDK 简介.....	1
1.1	NDK 配置.....	1
1.2	验证	2
2	Chromatix 工程添加（新方法）	3
2.1	从源码拷贝头文件	3
2.2	重新整理头文件	4
2.3	添加 chromatix 效果工程文件	4
3	Chromatix 工程修改（该方法废弃）	5
3.1	新建工程目录	5
3.2	修改工程文件	6
3.3	目录下对应 Android.mk 的修改.....	7
3.4	源文件下对应 Android.mk 文件的修改.....	7
4	编译和 push 验证	8
4.1	脚本修改说明	9
4.1.1	编译脚本：buildSo_win_v2.0.bat	9
4.1.2	Push 库文件脚本： push.ver2.0.bat	10
5	可能遇到的错误	10
	附录：Chromatix 工程范例.....	11

1 NDK 简介

NDK 全称 Native Development Kit，是一系列工具的集合。它提供了一系列的工具，帮助开发者快速开发 C/C++ 的动态库，并能自动将 so 和 java 一起打包成 apk。本文不做任何关于 NDK 开发的介绍，只对 NDK 提供的工具的使用做一个简介，主要方便 Camera Tuning 人员使用 NDK 来编译生成 chromatix 对应的库文件，同时驱动大神也可以用来编译 eeprom 的库文件。

好处：编译时检查语法错；速度快于 buildSo 软件。

1.1 NDK 配置

工具下载，可在官网下载。

Google 官网下载地址：

<https://developer.android.google.cn/ndk/downloads/index.html>

平台	软件包	大小 (字节)	SHA1 校验和
Windows 32 位	android-ndk-r14b-windows-x86.zip	707533928	070443eaa7fa37ed337f91c655e02ca708d37c92
Windows 64 位	android-ndk-r14b-windows-x86_64.zip	769151176	a625e8c599bccdb9061b61dcf3d1f1a01071613f
Mac OS X	android-ndk-r14b-darwin-x86_64.zip	824705073	2bf582c43f6da16416e66203d158a6dfaba4277c
Linux 64 位 (x86)	android-ndk-r14b-linux-x86_64.zip	840626594	becd161da6ed9a823e25be5c02955d9cbca1dbeb

根据自己硬件选择对应平台，下载相应的 NDK 版本。

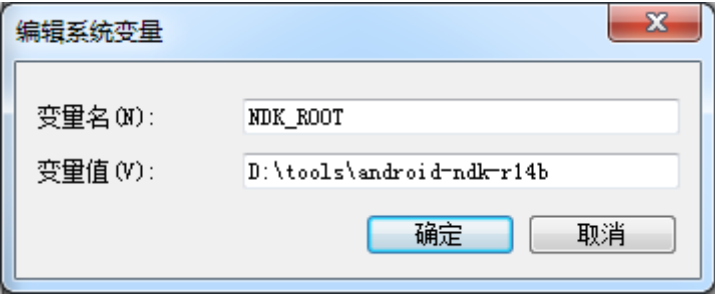
注：文档完成初稿时，官网最新 NDK 为 r14b，目前已更新到 r20 了，按照页面下载最新的版本即可，一般情况，里面会包含几乎所有的 android api，因此 NDK 版本影响不大。

Windows 配置（Linux 环境大同小异，编译时将 bat 命令全部替换为 sh 命令即可）：

下载 Window 版本之后，文件解压如下：

名称	修改日期	类型	大小
build	2017/10/26 17:16	文件夹	
platforms	2017/10/26 17:23	文件夹	
prebuilt	2017/10/26 17:23	文件夹	
python-packages	2017/10/26 17:23	文件夹	
shader-tools	2017/10/26 17:23	文件夹	
simpleperf	2017/10/26 17:23	文件夹	
sources	2017/10/26 17:24	文件夹	
sysroot	2017/10/26 17:25	文件夹	
toolchains	2017/10/26 17:25	文件夹	
CHANGELOG.md	2017/3/15 21:28	MD 文件	5 KB
ndk-build.cmd	2017/3/15 21:39	Windows 命令脚本	1 KB
ndk-depends.cmd	2017/3/15 21:39	Windows 命令脚本	1 KB
ndk-gdb.cmd	2017/3/15 21:39	Windows 命令脚本	1 KB
ndk-stack.cmd	2017/3/15 21:39	Windows 命令脚本	1 KB
ndk-which.cmd	2017/3/15 21:39	Windows 命令脚本	1 KB
source.properties	2017/3/15 21:39	PROPERTIES 文件	1 KB

配置环境变量：
“计算机” —“属性”—“高级系统设置”—“系统环境变量”—“新建”，输入如下内容：



注：变量值为 ndk-build 解压之后的根目录。
新建完成之后，点击“确定”，选择“Path”—“编辑”，在字符串末尾添加如下字符：
“;%NDK_ROOT%”
注：添加时若 path 字符串末尾如果没有分号（“;”），需手动添加，与上一个变量值分开。
添加完成之后，点击“确定”，完成环境变量配置。

1.2 验证

打开 cmd，终端中输入如下指令：
C:\Users\Alvin\Desktop>ndk-build -v
环境变量配置无误情况下，应输出如下内容：

```
C:\Users\101004341>ndk-build -v
GNU Make 3.81
Copyright (C) 2006 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.
There is NO warranty; not even for MERCHANTABILITY or FITNESS FOR A
PARTICULAR PURPOSE.

This program built for i586-pc-mingw32
```

接下来进行工程代码的修改。

2 Chromatix 工程添加（新方法）

如下这部分，已经存在整理好的工程和头文件，可通过 GitHub 直接下载使用。

地址：https://github.com/alvin08s/chromatix_demo

下载之后，可直接跳过 2.1 和 2.2 章节，直接进入 2.3 章节查看使用方法。若里面没有对应的平台头文件，可通过章节 2.1 和 2.2 手动添加。

同时，MSM8937/MSM8953/MSM8976/SDM450、SDM632/SDM636/SDM660 等使用的头文件一致，选择任意其一即可。

2.1 从源码拷贝头文件

使用 ndk-build 会进行语法，因此需要包含必要的头文件，这部分文件通过从对应平台的源码中拷贝，位置如下：

vendor\qcom\proprietary\mm-camerasdk\sensor\includes

名称	修改日期	类型	大小
0309	2018/1/18 10:28	文件夹	
0310	2018/1/18 10:28	文件夹	
actuator_driver.h	2018/1/18 10:28	H 文件	5 KB
chromatix_params.h	2018/1/18 10:28	H 文件	2 KB
debug_lib.h	2018/1/18 10:28	H 文件	6 KB
eeeprom_lib.h	2018/1/18 10:28	H 文件	9 KB
eeeprom_util.h	2018/1/18 10:28	H 文件	1 KB
flash_lib.h	2018/1/18 10:28	H 文件	3 KB
ois_driver.h	2018/1/18 10:28	H 文件	5 KB
pdaf_api.h	2018/1/18 10:28	H 文件	4 KB
pdaf_camif_api.h	2018/1/18 10:28	H 文件	2 KB
pdaf_lib.h	2018/1/18 10:28	H 文件	21 KB
sensor_lib.h	2018/1/18 10:28	H 文件	26 KB
sensor_sdk_common.h	2018/1/18 10:28	H 文件	4 KB

vendor\qcom\proprietary\mm-camera\common

名称	修改日期	类型	大小
cam_fifo.h	2018/1/18 10:28	H 文件	1 KB
cam_list.h	2018/1/18 10:28	H 文件	2 KB
cam_mmap.c	2018/1/18 10:28	C 文件	5 KB
cam_mmap.h	2018/1/18 10:28	H 文件	1 KB
camera.h	2018/1/18 10:28	H 文件	34 KB
camera_dbg.h	2018/1/18 10:28	H 文件	2 KB
camera_defs_i.h	2018/1/18 10:28	H 文件	1 KB
dsp_queues.h	2018/1/18 10:28	H 文件	4 KB
mm_camera_interface.h	2018/1/18 10:28	H 文件	8 KB
stacktrace.c	2018/1/18 10:28	C 文件	3 KB

Common 目录下的 camera_defs_i.h 和 camera_dbg.h 会被使用到。

2.2 重新整理头文件

因不同平台，头文件会有差异，因此以不同平台分别建立各自的文件夹：

名称	修改日期	类型	大小
msm8937	2018/1/18 10:15	文件夹	
sdm636	2018/1/18 10:14	文件夹	

不同的 3A header 也根据 3A version 来建立各自的目录：

名称	修改日期	类型	大小
0309	2018/1/17 14:53	文件夹	
0310	2018/1/17 15:01	文件夹	

将 common 中的如下两个头文件 copy 到 3A header 目录下。

名称	修改日期	类型	大小
camera_defs_i.h	2017/10/2 11:05	H 文件	1 KB
chromatix.h	2016/1/27 14:31	H 文件	75 KB
chromatix_3a.h	2016/1/27 14:31	H 文件	154 KB
chromatix_common.h	2016/1/27 14:31	H 文件	16 KB
chromatix_cpp.h	2016/1/27 14:31	H 文件	23 KB
chromatix_cpp_stripped.h	2016/1/27 14:31	H 文件	2 KB
chromatix_metadata.h	2016/1/27 14:31	H 文件	4 KB
chromatix_swpostproc.h	2016/1/27 14:31	H 文件	8 KB
sensor_dbg.h	2016/4/19 11:24	H 文件	1 KB

如上，已有整理好的资源，可下载直接使用，若有缺少的，自己根据当前代码添加即可。

下载位置：https://github.com/alvin08s/chromatix_demo

2.3 添加 chromatix 效果工程文件

从代码中拷贝一份 chromatx 文件，放置在任意目录下：

名称	修改日期	类型	大小
chromatix_hi846_hlt_lnd	2018/1/17 14:53	文件夹	
chromatix_hi846_hlt_lnd_1107	2018/1/18 9:50	文件夹	
chromatix_imx362	2018/1/18 9:55	文件夹	
chromatix_imx376	2018/1/18 9:51	文件夹	
includes	2018/1/18 10:15	文件夹	
Scripts	2018/1/18 10:44	文件夹	
LICENSE	2017/11/27 13:50	文件	11 KB

将 Scripts 中的脚本 buildSo_win_xxx.bat（通过 Github 查看和下载）拷贝到 chromatix 工程目录里下，记事本打开之后修改如下关键部分：

```
:: Defined by user
set INCLUDES_ROOT=C:\mm-camerasdk\includes
set BOARD_PLATFORM=msm8953
set HEADER_VERSION=0309
```

将 **BOARD_PLATFORM** 修改为自己工程对应的平台，如：msm8937，SDM660 等
将 **HEADER_VERSION** 修改对应的 3A 的版本号，如 0309，0310 等
将 **APP_PLATFORM** 根据当前使用机器的安卓版本准确填写，否则可能出现编译的库导致相机无法打开的异常，对应的 API 在脚本中已经备注说明。

如上两个信息决定查找头文件的位置，因此两个名字保证是在 includes 下存在并且有对应的头文件！！

修改完成之后直接双击运行即可直接编译，此方法不需要对 chromatix 的原始工程文件做任何的额外修改。注意，需要保证拷贝的工程是完整的，即每一级目录下都存在一个 Android.mk 文件。

若在编译中，提示缺少某个头文件，根据终端上的提示，可在源码中找到后复制到 includes 对应的目录下即可。

3 Chromatix 工程修改（该方法废弃）

该方法需要修改的 Makefile 较多，不太方便，保留可用于参考，借助该方法编译其他库文件。

因为 Chromatix 工程文件是存在源 Android 源码中的，同时 Android.mk 文件中存在多个宏开关来控制是否被编译，因为要单独编译对应的库文件，因此 Android.mk 文件要进行修改。

3.1 新建工程目录

本次以 Android 某项目 Chromatix 工程为例进行说明。

先创建一个工程目录，比如 Chromatix_NDK，添加需要修改的 Chromatix 工程文件和头文件：

rkspcace ▶ Y6_London ▶ chromatix_NDK ▶		
新建文件夹		
名称	修改日期	类型
chromatix	2017/11/19 11:52	文件夹
chromatix_hi846_hlt_ind_initial	2017/11/17 12:17	文件夹
chromatix_hi846_hlt_ind_initial	2017/11/20 15:13	文件夹
chromatix_hi846_hlt_ind_initial	2017/11/27 11:03	文件夹
includes	2017/11/17 12:17	文件夹
Scripts	2017/11/17 12:33	文件夹

所需头文件全部位于 includes 文件夹，所包含头文件如下（0309）：

rkspcace ▶ Y6_London ▶ chromatix_NDK ▶ includes ▶ 0309			
新建文件夹			
名称	修改日期	类型	大小
camera_defs_i.h	2017/10/2 11:05	H 文件	1 KB
chromatix.h	2016/1/27 14:31	H 文件	75 KB
chromatix_3a.h	2016/1/27 14:31	H 文件	154 KB
chromatix_common.h	2016/1/27 14:31	H 文件	16 KB
chromatix_cpp.h	2016/1/27 14:31	H 文件	23 KB
chromatix_cpp_stripped.h	2016/1/27 14:31	H 文件	2 KB
chromatix_metadata.h	2016/1/27 14:31	H 文件	4 KB
chromatix_swpostproc.h	2016/1/27 14:31	H 文件	8 KB
sensor_dbg.h	2016/4/19 11:24	H 文件	1 KB

头文件部分内容为高通对结构体的定义，header 版本只要为 0309，内容均是一致的，直接拷贝使用；若 header 为其他版本，在 Android 源码中，将对应的文件拷贝出来重新建立一个文件夹即可。

3.2 修改工程文件

需要修改的文件列表如下：

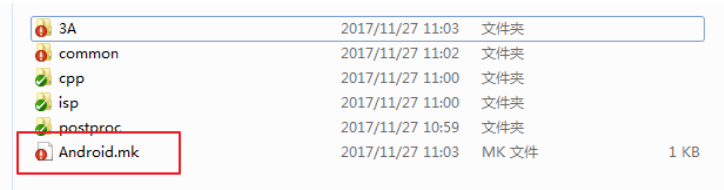
- 1、 各目录下的 Android.mk；

default_preview	2017/11/27 9:17	文件夹	
default_video	2017/11/27 9:15	文件夹	
zsl_preview	2017/11/27 9:16	文件夹	
zsl_video	2017/11/27 9:15	文件夹	
Android.mk	2017/10/4 9:03	MK 文件	1 KB

- 2、 各源文件下对应的 Android.mk

Android.mk	2017/10/4 9:03	MK 文件	2 KB
chromatix_hi846_hlt_ind_zsl_preview.c	2017/9/11 15:22	C 文件	1 KB
chromatix_hi846_hlt_ind_zsl_preview.h	2017/11/27 9:47	H 文件	945 KB

3.3 目录下对应 Android.mk 的修改



源代码如下：

```
1 ifeq ($(call is-vendor-board-platform,QCOM),true)
2 include $(call all-subdir-makefiles)
3 endif
```

源码中存在一个平台宏开关，因为是单独编译，取消开关控制：

```
1 - ifeq ($(call is-vendor-board-platform,QCOM),true)
2 1 include $(call all-subdir-makefiles)
3 - endif
```

（减号代表删除）

这样在编译的过程中方可包含目录下能找到子目录下存在的所有 Makefile 文件。

依次，将 3A、cpp、isp 下面的 Android.mk 全部修改掉。

3.4 源文件下对应 Android.mk 文件的修改

修改此处文件是因为，Android.mk 中指定的头文件路径是 Android 源码对应的路径，单独编译时，我们需要指定我们自己头文件所放的位置，否则编译会报错：无法找到对应的头文件。

以 3A 目录下 zsl_preview 为例：

未修改前，源代码如下：

```
LOCAL_C_INCLUDES := $(LOCAL_PATH)/../../../../../../../../mm-camerasdk/sensor/includes/$(CHROMATIX_VERSION)
```

此时，将 LOCAL_C_INCLUDES 指向的路径，修改为工程目录下 include 对应的路径：

```
LOCAL_C_INCLUDES := $(LOCAL_PATH)/../../../../includes/0309
```

相对路径可自己推算，只要指向工程目录下的“includes/0309”位置即可。若工程目录不改变，路径名也可以写成绝对路径，但是弊端是工程代码拷贝到其他地方时，运行就会报错。

修改前后的对比：

```
17 17 -include camera_defs_1.h
18 18
19 - LOCAL_C_INCLUDES := $(LOCAL_PATH)/../../../../../../../../mm-camerasdk/sensor/includes/$(CHROMATIX_VERSION)
19 + LOCAL_C_INCLUDES := $(LOCAL_PATH)/../../../../includes/0309
20 20 LOCAL_C_INCLUDES += $(LOCAL_PATH)/../../../../module/
```

此时，代码的修改工作已经完成，只要使用编辑好的 buildSo.bat 命令进行编译即可。

此处需要注意，includes 和 chromatx 工程目录的位置关系最好刚开始就做一个固定的设置，这样方便后续添加其他 sensor 的效果代码。

4 编译和 push 验证

将 Chromatix_NDK 目录中“Script”下的“buildSo_win_v2.0.bat”和“push.ver2.0.bat”拷贝到 Chromatix 工程根目录下：

名称	修改日期	类型	大小
3A	2017/10/26 17:17	文件夹	
common	2017/11/17 14:12	文件夹	
cpp	2017/10/26 17:17	文件夹	
isp	2017/10/26 17:17	文件夹	
libs	2017/11/20 9:46	文件夹	
obj	2017/11/20 9:46	文件夹	
postproc	2017/11/23 16:59	文件夹	
Android.mk	2017/10/20 20:46	MK 文件	1 KB
buildSo.bat	2017/11/17 14:20	Windows 批处理...	1 KB
clean.bat	2017/11/17 12:23	Windows 批处理...	1 KB
push.ver1.1.bat	2017/11/17 15:26	Windows 批处理...	1 KB
restartMediaServices_Y6.bat	2017/11/17 12:31	Windows 批处理...	1 KB

双击运行“buildSo.bat”，编译结果如下：

```
[armeabi] Compile thumb : chromatix_hi846_hlt_lnd_zsl_video <= chromatix_hi846_hlt_lnd_zsl_video.c
[armeabi] SharedLibrary : libchromatix_hi846_hlt_lnd_a3_default_preview.so
[armeabi] SharedLibrary : libchromatix_hi846_hlt_lnd_a3_default_video.so
[armeabi] SharedLibrary : libchromatix_hi846_hlt_lnd_common.so
[armeabi] SharedLibrary : libchromatix_hi846_hlt_lnd_cpp_liveshot.so
[armeabi] SharedLibrary : libchromatix_hi846_hlt_lnd_cpp_preview.so
[armeabi] SharedLibrary : libchromatix_hi846_hlt_lnd_cpp_snapshot.so
[armeabi] SharedLibrary : libchromatix_hi846_hlt_lnd_cpp_video.so
[armeabi] SharedLibrary : libchromatix_hi846_hlt_lnd_default_video.so
[armeabi] SharedLibrary : libchromatix_hi846_hlt_lnd_liveshot.so
[armeabi] SharedLibrary : libchromatix_hi846_hlt_lnd_postproc.so
[armeabi] SharedLibrary : libchromatix_hi846_hlt_lnd_preview.so
[armeabi] SharedLibrary : libchromatix_hi846_hlt_lnd_snapshot.so
[armeabi] SharedLibrary : libchromatix_hi846_hlt_lnd_zsl_preview.so
[armeabi] SharedLibrary : libchromatix_hi846_hlt_lnd_zsl_video.so
[armeabi] Install : libchromatix_hi846_hlt_lnd_a3_default_preview.so => libs/armeabi/libchromatix_hi846_hlt_lnd_a3...
[armeabi] Install : libchromatix_hi846_hlt_lnd_a3_default_video.so => libs/armeabi/libchromatix_hi846_hlt_lnd_a3_de...
[armeabi] Install : libchromatix_hi846_hlt_lnd_common.so => libs/armeabi/libchromatix_hi846_hlt_lnd_common.so
[armeabi] Install : libchromatix_hi846_hlt_lnd_cpp_liveshot.so => libs/armeabi/libchromatix_hi846_hlt_lnd_cpp_lives...
[armeabi] Install : libchromatix_hi846_hlt_lnd_cpp_preview.so => libs/armeabi/libchromatix_hi846_hlt_lnd_cpp_previe...
[armeabi] Install : libchromatix_hi846_hlt_lnd_liveshot.so => libs/armeabi/libchromatix_hi846_hlt_lnd_liveshot.so
[armeabi] Install : libchromatix_hi846_hlt_lnd_cpp_snapshot.so => libs/armeabi/libchromatix_hi846_hlt_lnd_cpp_snap...
[armeabi] Install : libchromatix_hi846_hlt_lnd_cpp_video.so => libs/armeabi/libchromatix_hi846_hlt_lnd_cpp_video.so
[armeabi] Install : libchromatix_hi846_hlt_lnd_default_video.so => libs/armeabi/libchromatix_hi846_hlt_lnd_default...
[armeabi] Install : libchromatix_hi846_hlt_lnd_postproc.so => libs/armeabi/libchromatix_hi846_hlt_lnd_postproc.so
[armeabi] Install : libchromatix_hi846_hlt_lnd_preview.so => libs/armeabi/libchromatix_hi846_hlt_lnd_preview.so
[armeabi] Install : libchromatix_hi846_hlt_lnd_snapshot.so => libs/armeabi/libchromatix_hi846_hlt_lnd_snapshot.so
[armeabi] Install : libchromatix_hi846_hlt_lnd_zsl_preview.so => libs/armeabi/libchromatix_hi846_hlt_lnd_zsl_previe...
[armeabi] Install : libchromatix_hi846_hlt_lnd_zsl_video.so => libs/armeabi/libchromatix_hi846_hlt_lnd_zsl_video.so

Build Finished!
请按任意键继续...
```

会在当前路径下生成两个目录：

名称	修改日期	类型	大小
3A	2017/11/27 9:15	文件夹	
common	2017/11/27 9:15	文件夹	
cpp	2017/11/27 9:15	文件夹	
isp	2017/11/27 9:15	文件夹	
libs	2017/11/27 9:15	文件夹	
obj	2017/11/27 9:15	文件夹	
postproc	2017/11/27 9:15	文件夹	
Android.mk	2017/10/4 9:00	MK 文件	1 KB
buildSo.bat	2017/11/17 14:20	Windows 批处理...	1 KB
push.ver1.2.bat	2017/11/24 20:55	Windows 批处理...	1 KB
restartMediaSevices_Y6.bat	2017/10/20 20:41	Windows 批处理...	1 KB

Obj 为生成的链接文件，libs 下为生成的库文件，需要 push libs 目录下的库文件到机器：

名称	修改日期	类型	大小
libchromatix_hi846_hlt_lnd_a3_default_preview.so	2017/11/27 14:16	SO 文件	63 KB
libchromatix_hi846_hlt_lnd_a3_default_video.so	2017/11/27 14:16	SO 文件	63 KB
libchromatix_hi846_hlt_lnd_common.so	2017/11/27 14:16	SO 文件	66 KB
libchromatix_hi846_hlt_lnd_cpp_liveshot.so	2017/11/27 14:16	SO 文件	46 KB
libchromatix_hi846_hlt_lnd_cpp_preview.so	2017/11/27 14:16	SO 文件	46 KB
libchromatix_hi846_hlt_lnd_cpp_snapshot.so	2017/11/27 14:16	SO 文件	46 KB
libchromatix_hi846_hlt_lnd_cpp_video.so	2017/11/27 14:16	SO 文件	46 KB
libchromatix_hi846_hlt_lnd_default_video.so	2017/11/27 14:16	SO 文件	56 KB
libchromatix_hi846_hlt_lnd_liveshot.so	2017/11/27 14:16	SO 文件	56 KB
libchromatix_hi846_hlt_lnd_postproc.so	2017/11/27 14:16	SO 文件	17 KB
libchromatix_hi846_hlt_lnd_preview.so	2017/11/27 14:16	SO 文件	56 KB
libchromatix_hi846_hlt_lnd_snapshot.so	2017/11/27 14:16	SO 文件	56 KB
libchromatix_hi846_hlt_lnd_zsl_preview.so	2017/11/27 14:16	SO 文件	63 KB
libchromatix_hi846_hlt_lnd_zsl_video.so	2017/11/27 14:16	SO 文件	63 KB

手机在开启 adb 并连接电脑的情况下，直接双击“push.ver1.2.bat”即可完成 push 操作。

4.1 脚本使用说明

4.1.1 编译脚本：buildSo_win_v2.0.bat

需要修改的有如下三个地方：

```
17  :: Defined by user
18  set INCLUDES_ROOT=C:\mm-camerasdk\includes
19  set BOARD_PLATFORM=msm8953
20  set HEADER_VERSION=0309
```

INCLUDES_ROOT:

为 2.2 章节中整理的头文件位置，这部分需要根据自己当前存放的位置进行修改；

BOARD_PLATFORM:

为当前项目使用的平台，MSM8937\MSM8953\SDM450、SDM636\SDM660 等平台使用的头文件均一致，等因此脚本只要任意指定其中一个平台即可；

HEADER_VERSION:

当前效果使用的 header，目前使用的有 0309 和 0310 版本，最新为 0310。

如上设置完成之后，其实实际位置即为：C:\mm-camerasdk\includes\sdm450\0309。

4.1.2 Push 库文件脚本： push.ver2.0.bat

Push 的路径可以是/system/vendor/lib/也可以是/vendor/lib，其实，/system/vendor/lib/即为路径/vendor/lib 的软链接。

5 可能遇到的错误

1.双击编译脚本之后报如下错误：

```
Build Starts...

GNU Make 3.81
Copyright (C) 2006 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.
There is NO warranty; not even for MERCHANTABILITY or FITNESS FOR A
PARTICULAR PURPOSE.

This program built for i586-pc-mingw32

D:/tools/android-ndk-r14b/build/./build/core/build-all.mk:89: Android NDK: WARNING: There are no modules to build in thi
make: Nothing to be done for 'all'.

Build Finished!
请按任意键继续. . .
```

解决方法：查看和编译脚本所在目录是否存在 Android.mk。

2.报错如下：

```
l error generated.
make: *** [obj/local/armeabi/objs/chromatix_hi846_hlt_lnd_a3_hfr_120/chromatix_hi846_hlt_lnd_a3_hfr_120.o] Error 1
make: *** [obj/local/armeabi/objs/chromatix_hi846_hlt_lnd_a3_default_preview/chromatix_hi846_hlt_lnd_a3_default_preview.o] Error 1
make: *** [obj/local/armeabi/objs/chromatix_hi846_hlt_lnd_a3_default_video/chromatix_hi846_hlt_lnd_a3_default_video.o] Error 1
[armeabi] Compile thumb : chromatix_hi846_hlt_lnd_zsl_video <= chromatix_hi846_hlt_lnd_zsl_video.c
make: *** [obj/local/armeabi/objs/chromatix_hi846_hlt_lnd_common/chromatix_hi846_hlt_lnd_common.o] Error 1
make: *** [obj/local/armeabi/objs/chromatix_hi846_hlt_lnd_a3_hfr_90/chromatix_hi846_hlt_lnd_a3_hfr_90.o] Error 1
make: *** [obj/local/armeabi/objs/chromatix_hi846_hlt_lnd_a3_hfr_60/chromatix_hi846_hlt_lnd_a3_hfr_60.o] Error 1
make: *** [obj/local/armeabi/objs/chromatix_hi846_hlt_lnd_postproc/chromatix_hi846_hlt_lnd_sw_post_proc.o] Error 1
(built-in):1:10: fatal error: 'camera_defs_i.h' file not found
#include "camera_defs_i.h"

l error generated.
[armeabi] SharedLibrary : libchromatix_hi846_hlt_lnd_zsl_preview.so
make: *** [obj/local/armeabi/objs/chromatix_hi846_hlt_lnd_zsl_video/chromatix_hi846_hlt_lnd_zsl_video.o] Error 1
[armeabi] Install : libchromatix_hi846_hlt_lnd_zsl_preview.so => libs/armeabi/libchromatix_hi846_hlt_lnd_zsl_previe
make: Target 'all' not remade because of errors.

Build Finished!
请按任意键继续. . .
```

头文件找不到，查看源码对应的 Android.mk 中，对头文件目录的指向是否正确、指向的目录下是否存在对应的文件。

3.报错如下：

```
common/chromatix_hi846_hlt_lnd_common.h:14:2: error: expected expression
{,,
common/chromatix_hi846_hlt_lnd_common.h:14:3: error: expected expression
{,,
common/chromatix_hi846_hlt_lnd_common.h:14:4: error: expected expression
{,,
3 errors generated.
make: *** [obj/local/armeabi/objs/chromatix_hi846_hlt_lnd_common/chromatix_hi846_hlt_lnd_common.o] Error 1
[armeabi] SharedLibrary : libchromatix_hi846_hlt_lnd_postproc.so
[armeabi] Install      : libchromatix_hi846_hlt_lnd_postproc.so => libs/armeabi/libchromatix_hi846_hlt_lnd_postproc.so
make: Target `all' not remade because of errors.

-----
Build Finished!
-----
请按任意键继续. . .
```

源码有语法错，第 14 行，多了几个逗号，实现了检查语法错的功能。

```
13  /* Header Version Info */
14  {,,
15      0x0309, /* Chromatix Version */
16      0, /* Revision */
17      /* Chromatix App Version */
18      {
19          6, /* Major */
20          13, /* Minor */
21          3, /* Revision */
22          0, /* Build */
23      },
```

附录：Chromatix 工程范例

工程 demo 模版已上传 GitHub：

地址：https://github.com/alvin08s/chromatix_demo

代码已验证过， ndk 配置没有问题的情况下，代码编译通过，下载后可直接在上面修改和验证。