

HDCP KEY **工具使用指南**

文档版本 09

发布日期 2016-03-15

版权所有 © 深圳市海思半导体有限公司 2016。保留一切权利。

非经本公司书面许可,任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部,并不得以任 何形式传播。

商标声明



(上) 、HISILICON、海思和其他海思商标均为深圳市海思半导体有限公司的商标。

本文档提及的其他所有商标或注册商标,由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受海思公司商业合同和条款的约束,本文档中描述的全部或部分产 品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定,海思公司对本文档内容不 做任何明示或默示的声明或保证。

由于产品版本升级或其他原因,本文档内容会不定期进行更新。除非另有约定,本文档仅作为使用 指导,本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

深圳市海思半导体有限公司

地址: 深圳市龙岗区坂田华为基地华为总部 邮编: 518129

网址: http://www.hisilicon.com

客户服务邮箱: support@hisilicon.com



前言

概述

本文档主要介绍海思 HDCP key 工具的使用方法,以及应用 HDCP key 方案时的工厂操作流程。

产品版本

与本文档相对应的产品版本如下。

产品名称	产品版本
Hi3716C	V2XX
Hi3751	V1XX
Hi3719C	V1XX
Hi3719M	V1XX
Hi3718C	V1XX
Hi3718M	V1XX
Hi3716C	V1XX
Ні3716Н	V1XX
Hi3716M	V2XX
Hi3716M	V3XX
Hi3712	V1XX
Hi3716M	V4XX
Hi3798C	V1XX
Hi3798C	V2XX
Hi3796C	V1XX
Hi3798M	V1XX
Hi3796M	V1XX



产品名称	产品版本
Hi3110E	V5XX

读者对象

本文档(本指南)主要适用于以下工程师:

- 技术支持工程师
- 软件开发工程师

作者信息

章节号	章节名称	作者信息
全文	全文	Z00213260/Y00229039/H00222638

修订记录

修订记录累积了每次文档更新的说明。最新版本的文档包含以前所有文档版本的更新内容。

修订日期	版本	修订说明
2013-07-31	00B01	第一次临时版本发布。
2013-11-19	01	增加 3.2.4.1、3.2.4.2、3.2.4.3 章节。
2014-01-18	02	修改文档引用错误;增加 Hi3716MV400。
2014-04-16	03	增加海思 Android 平台使能 HDCP 1.x 特性的方法; 增加 Hi3798CV100,Hi3796CV100 芯片支持的说明。
2014-07-07	04	更新 HDCP_Key 工具版本、操作描述以及配图。
2014-11-05	05	增加 Hi3716MV310、Hi3719CV100、Hi3798MV100、 Hi3796MV100 芯片支持说明;修改第 2 章工具界面图。
2015-03-10	06	新增支持 Hi3110E V500 芯片。
2015-04-23	07	新增支持 Hi3798CV200、Hi3716MV410/420 芯片。
2015-11-10	08	修改 3.2.5 章节。
2016-3-15	09	增加 Hi3716MV330 芯片支持说明。



目录

前	〕 言	iii
1	概述	1-1
	1.1 工具概述	1-1
	1.2 环境准备说明	1-1
	1.3 快速入门	1-2
	1.3.1 安装	1-2
	1.3.2 主界面概览	1-2
2	界面及功能说明	2-1
3	工具应用参考	3-1
	3.1 Hi3716CV100/Hi3716HV100/Hi3716MV200/Hi3716MV300 芯片 HDCP key 方案	3-1
	3.1.1 应用接口及相关结构体	3-1
	3.1.2 Sample 介绍	3-2
	3.1.3 工厂烧写 HDCP key	3-2
	3.1.4 验证 HDCP key	3-4
	3.2 Hi3712/Hi3716CV200 及后续芯片 HDCP key 方案	3-5
	3.2.1 应用接口及相关结构体	3-5
	3.2.2 Sample 介绍	3-6
	3.2.3 工厂生产	3-7
	3.2.4 Linux 平台使能 HDCP 1.x 特性	3-8
	3.2.5 Android 平台使能 HDCP 1.x 特性	3-11
1	注音車 項	4_1



插图目录

图 1-1 工具主界面图	1-2
图 2-1 HDCP key 工具使用图(一)	2-2
图 2-2 HDCP key 工具使用图(二)	2-3
图 2-3 HDCP key 工具使用图(三)	2-4
图 2-4 HDCP key 工具使用图(四)	2-5
图 2-5 HDCP key 工具使用图(五)	2-6
图 3-1 HDCP key 烧写错误提示(一)	3-3
图 3-2 HDCP key 烧写错误提示(二)	3-3
图 3-3 HDCP key 烧写成功	3-3
图 3-4 查看获取 HDMI event 是否成功(一)	3-4
图 3-5 查看获取 HDMI event 是否成功(二)	3-5
图 3-6 执行 sample 生成加密的 HDCP key	3-6
图 3-7 Hi3712/Hi3716CV200 及后续芯片 HDCP key 方案工厂生产流程	3-7
图 3-8 烧写并锁定 HDCP root key 流程	3-8
图 3-9 menuconfig 菜单	3-9
图 3-10 HDCP 选项	3-9
图 3-11 sample 默认配置	3-9
图 3-12 运行 HDMI sample 时没有准备被加密的 HDCP key	3-11
图 3-13 HDMI 模块加载了错误的被加密的 HDCP key	3-11
图 3-14 HDMI 模块加载 HDCP key 成功	3-11
图 3-15 定义的加密的 HDCP Key 路径	3-12
图 3-16 打开 HI_HDCP_SUPPRT 开关	3-13
图 3-17 完 ∀ σ HDCPFlag 的初始值	3_13



1 概述

1.1 工具概述

HDCP_key.exe 工具是拆分从 Digital Content Protection LLC 组织购买的 HDCP key 的工具,运行环境为 Windows 平台。

使用 HDCP key.exe 工具前,请注意:

- HDCP 是 HDMI 数据加密传输方式。HDCP key 必须从 Digital Content Protection LLC 组织购买。
- Hi3716CV100/Hi3716MV200/Hi3716MV300 芯片和 Hi3712/Hi3716CV200 及后续版本芯片采用不同的 HDCP key 方案。

□ 说明

Hi3712/Hi3716CV200及后续版本芯片包括: Hi3712、Hi3716CV200、Hi3719MV100、Hi3719CV100、Hi3798CV100、Hi3798CV100、Hi3798CV100、Hi3796CV100、Hi3796MV100、Hi3796CV100、Hi3716MV310、Hi3716MV330、Hi3716MV400。文中后续提到的"Hi3712/Hi3716CV200及后续芯片"的说法均参照此说明。

- 使用 Hi3716CV100/Hi3716MV200/Hi3716MV300 芯片时, HDCP key 直接烧写到芯片 OTP 内部; 使用 Hi3712/Hi3716CV200 及后续版本芯片时, 芯片内部 OTP 区域可存储专门用于加/解密 HDCP key 的根密钥 HDCP root key。
- Hi3712/Hi3716CV200 及后续版本芯片时,可使用 HDCP root key 对 HDCP key 进行加密,并将加密后的 HDCP key 数据存储到 flash 指定区域。需要使用 HDCP key 对 HDMI 的数据进行加密传输时,HDMI 模块从 flash 区域读取加密的 HDCP key 数据,并将其解密到芯片内部,使用解密后的 HDCP key 对数据进行加密传输。其中,解密后的 HDCP key 数据对外不可读。

1.2 环境准备说明

HDCP key.exe 工具使用的环境如下:

- 把 HDCP key.exe 工具拷贝到 windows 操作系统本地硬盘。
- 双击运行即可。



1.3 快速入门

1.3.1 安装

本工具无需安装,双击运行即可。

1.3.2 主界面概览



注意

以下工具说明适用于 HDCP Key 工具 V3.0.0.0(包括)至 V3.1.0.0(不包括)之间的版本。

本工具主界面显示如图 1-1 所示。

图1-1 工具主界面图





2 界面及功能说明

HDCP_key.exe 工具只有一个界面,HDCP key 的拆分功能在此界面可完成。可参考如下步骤使用 HDCP_key.exe 对 HDCP key 进行拆分:

步骤 1 选择 HDCP key 包 bin 文件,HDCP key 总数会自动显示,个数一般为 10000、100000 或 1000000。 如需分解 HDCP2.2 版本的包,点击下拉框切换至 HDCP 2.2。如图 2-1 与图 2-2 所示。



图2-1 HDCP key 工具使用图 (一)





图2-2 HDCP key 工具使用图(二)



步骤 2 如果是切割 HDCP 1.4 的源 Key 包,需要设置 Client ID,由 32BYTE 的英文字母或数字组成,输入框已设 32BYTE 输入限制。比如设定客户 ID 为: 1234567890123456789012。切割 HDCP 2.2 的包不需要输入客户号。如图 2-3 所示



图2-3 HDCP key 工具使用图(三)



- **步骤** 3 选择密钥存放文件夹、切割密钥的模式、切割的起始序号和生产的密钥数目。模式下 拉框提供如下三种方式:
 - ALL Sets。生产所有密钥。
 - Multiple Sets。生产 1-n 内连续的分 key (n 为 HDCP key 包的 key 数目)。多个 Key 下,两个文本框需要分别填写希望生产的多个 Key 的起始序号以及希望生产 Key 个数。
 - One Set。生产指定序号的密钥。
 密钥存放目录可以任意指定已存在的文件夹。
 - □ 说明

下面所有图以分解 HDCP 1.4 的 Key 包为示例,分解 HDCP2.2 的 Key 包同理。



图2-4 HDCP key 工具使用图(四)





图2-5 HDCP key 工具使用图(五)



□ 说明

在切割 HDCP 1.4Key 包时,分 key 的 bin 文件大小为 384 个字节,各字段意义如下:

- 第 0~7(8)字节:海思标记位:HISI_xxx
- 第 8~15(8)字节:工具版本号: V0000001
- 第 16~47(32)字节:客户的 ID 号:xxxxxxxxx
- 第 48~367(320)字节:加密后的 320bytes HDCP key(16 字节对齐)。
- 第 368~384(16)字节:后续使用,暂定为 0。

在切割 HDCP 2.2Key 包时,分 key 的 bin 文件大小为 902

- 第 0~39(40)字节:源 Key 包的 40 字节头
- 第 40~901(862)字节: HDCP 2.2 的 Key

----结束



3 工具应用参考

3.1 Hi3716CV100/Hi3716HV100/Hi3716MV200/Hi3716MV 300 芯片 HDCP key 方案

3.1.1 应用接口及相关结构体

- 接口 HI_UNF_HDCP_SetHDCPKeyEx 用于往芯片 OTP 区域烧写 HDCP key: HI_S32 HI_UNF_HDCP_SetHDCPKeyEx(HI_UNF_OTP_HDCPKEY_S stHdcpKey)
- 接口 HI_UNF_HDCP_LockHDCPKeyEx 用于锁定 OTP 区域烧写的 HDCP key: HI_S32 HI_UNF_HDCP_LockHDCPKeyEx(HI_VOID)
- 接口 HI_UNF_HDCP_GetKeyBurnFlag 用于获取 HDCP key 的烧写标志位,判断 HDCP key 是否已烧写:

HI_S32 HI_UNF_HDCP_GetKeyBurnFlag(HI_BOOL *pbKeyBurnFlag)

相关结构体如下:

```
typedef struct hiUNF_DECRYPT_HDCP_S
{
    HI_U8 u8KSV[5]; /**<HDCP KSV:40bits, Orignal data */
    HI_U8 u8PrivateKey[280];
}HI_UNF_DECRYPT_HDCP_S;
typedef struct hiUNF_ENCRYPT_HDCP_S
{
    HI_U8 u8EncryptKey[384]; /**<HDCP Encryption key */
}HI_UNF_ENCRYPT_HDCP_S;

typedef struct
{
    HI_BOOL EncryptionFlag;
    union
{
    /**< Key = DecryptData, if EncryptionFlag == HI_FALSE.if EncryptionFlag ==</pre>
```



HI_TRUE,key = Encrpytiondata */

HI_UNF_DECRYPT_HDCP_S DecryptData;

HI_UNF_ENCRYPT_HDCP_S EncryptData;

}key;

HI_U32 Reserved; /**<Reserved for future use */</pre>

}HI_UNF_OTP_HDCPKEY_S;

□ 说明

HI_UNF_OTP_HDCPKEY_S 结构体中 Reserved 成员为 0xabcd1234 时,不烧录 HDCP key,直接返回成功,用作调试。

HDCP key 的写入操作只能进行一次操作,第二次操作将不能够成功。

HI_UNF_HDCP_SetHDCPKey 的接口实现改为使用 HI_UNF_HDCP_SetHDCPKeyEx 和 HI_UNF_HDCP_LockHDCPKeyEx。

3.1.2 Sample 介绍

位于 SDK 发布包目录 sample/sethdcpkey 下有 3 个烧 HDCP key 的 sample。

- sethdcp orgkey.c。使用原始单个 HDCP key 直接烧写。
- sethdcp orgpacketkey.c。使用原始 HDCP 包,选择其中的某个 key 烧写。
- sethdcp encrykey.c。使用海思拆分工具产生的加密 HDCP key 烧写。

其参数请参看 sdk/sample/sethdcp/Readme.txt

下文以 sample: sethdcp_encrykey.c(编译后可执行文件为 sample_sethdcp_encrykey)为例进行说明。

3.1.3 工厂烧写 HDCP key

请参考如下流程烧写 HDCP key。

步骤 1 检测所需的 ko 模块是否插入,hi_sethdcp.ko, hi_cipher.ko 和 hi_otp.ko. 可以使用:lsmod 察看,如果上面的模块没有加载,需要手动执行,如:insmod /kmod/ hi_sethdcp.ko

- **步骤 2** 拷贝加密之后的分 key(384 字节大小,使用工具生成的 BIN 文件)至 SDK 发布包目录 sample/sethdcpkey。
- 步骤 3 烧写 HDCP key。执行:

./sample_sethdcp_encrykey

10000_fake_key.bin_hisi_encry_v00000001_key0000001.bin

烧写 HDCP key 时,主要提示信息及截图如图 3-1 所示。

未插入 hi sethdcp.ko。



图3-1 HDCP key 烧写错误提示(一)

```
# | 1s | Makefile | Readme.txt | Orignalpacket2key.bin_hisi_encry_v0000001_key0000002.bin | sample_sethdcp_encrykey | sample_sethdcp_orgkey | sample_sethdcp_orgkey | sample_sethdcp_orghex.c | sethdcp_orghex.c | sethdcp_org
```

● 已烧写 HDCP key 或烧写 HDCP key 错误。

图3-2 HDCP key 烧写错误提示(二)

```
# ./sample_sethdcp_encrykey orignalpacket2key.bin_hisi_encry_v0000001_key0000002
.bin
binfilelen=384
In drv_sethdcp:Start to Decrypt, chnNum = 0x7!
arg=0x7.
In Drv_burnhdcp:Decrypt OK, chnNum = 0x7!
HDCP is already locked
HDCPKey_Burn error
Error burnhdcp ioctl CMD_SETHDCP_SETHDCPKEY err. ret = -2142371830
HI UNF_HDCP_SetHDCPKey failed
#
```

● 烧写 HDCP key 成功。

图3-3 HDCP key 烧写成功

```
# 1s
Makefile
Readme.txt
Orignalpacket2key.bin_hisi_encry_v0000001_key0000002.bin
sample_sethdcp_encrykey
sample_sethdcp_orgkey
sample_sethdcp_orgpacketkey
sethdcp_encrykey.c
sethdcp_encrykey.c
sethdcp_orgpacketkey.c
# ./sample_sethdcp_encrykey orignalpacket2key.bin_hisi_encry_v0000001_key0000002
.bin
binfilelen=384
In drv_sethdcp:Start to Decrypt, chnNum = 0x7!
arg=0x7.
In Drv_burnhdcp:Decrypt OK, chnNum = 0x7!
do_precharge_apb_result: status = 0
do_precharge_apb_result: status = 0
set_hdcp_cmd = 0x41885401_ok !
```

----结束



3.1.4 验证 HDCP key

成功烧写 HDCP key 后,切换到 SDK/sample/hdmi_tsplay 目录播放高清 TS 文件,可验证烧入的 HDCP key 是否可用。步骤如下:

- 步骤 1 准备好需要播放的 TS 码流, 存放至 SDK/sample/hdmi_tsplay 目录。
- 步骤 2 运行./sample_hdmi_tsplay 1254-SALON_AUTO_2-1080i25@30MplusAud.ts 1080i_50 1 (1 为 HDCP 播放 flag)。

查看获取 HDMI event 是否成功。

图3-4 查看获取 HDMI event 是否成功(一)

```
# pwd
/mnt/Hi3716MV100R001C00SPC040/sample/hdmi_tsplay
#
# ls
1254-SALON_AUTO_2-1080i25@30MplusAud.ts
Makefile
hdmi_test_cmd.c
hdmi_test_cmd.h
hdmi_tsplay.c
readme.txt
sample.jpg
sample hdmi_tsplay
simhei.ttf
#
# //sample_hdmi_tsplay 1254-SALON_AUTO_2-1080i25@30MplusAud.ts 1080i_50 1
argv[3]:1, g_HDCPFlag:1

!!! The format is '1080i_50'/mce_Open,218: ok !
6.
mce_Release,231: ok !
reset register
g_HDMTForceMode:0
HDMI_Init_Mode:0
stHDMIInit_enForceMode:0
[8194000 ERROR-user]:PSI_DataRead[99]:read_date_num:1 len :0x14
[8194001 ERROR-user]:PSI_DataRead[99]:read_date_num:1 len :0x26

ALL Program Infomation:
Channum = 0, Program ID = 1, Channel Num = 1
[8194100 ERROR-hdmi]:SI_TimerHandler[481]:force_HDCP_stop

Audio_Stream_Type_MP3
Video_Stream_Type_MP3
Video_Stream_Type_MP3
Video_Stream_Type_MP3
Video_Stream_Type_MP3
```



图3-5 查看获取 HDMI event 是否成功(二)

```
MPEG2: 0
ACC: 0
DTS: 0
ATRAC: 0
One Bit Audio: 0
Dolby Digital +: 0
DTS-HD: 0
MAT: 0
DST: 0
WMA Pro: 0
Max Audio PCM channels: 2
Support Audio Sample Rates: 32000 44100 48000 88200 96000 192000
Support Audio channels: LFE RL/RR
bIsPhyAddrValid:1
Physical Address:01.00.00.00
bDVI Dual:0x00, bDC Y444:0x01, bSupports AI:0x00 bDC 30bit:0x01, bDC 36bit:0x01, bDC 48bit:0x00 bSupportYCbCr:1, xvYCC601:0, xvYCC709:0, MaxTMDS:205
Output result: Video Pid: 4113 mpeg VO: 1920x1080I@50 24bit
Output result: Audio Pid: 4352 mp3 AO: uncompress
      Get HDMI event: HDCP SUCCESS.
Output result: Video Pid: 4113 mpeg Vo:1920x10801@50 24bit
Output result: Audio Pid: 4352 mp3 AO:uncompress
```

当成功获得 "--- Get HDMI event: HDCP_SUCCESS. --- "信息时,表明 HDCP key 烧写已成功。

----结束

3.2 Hi3712/Hi3716CV200 及后续芯片 HDCP key 方案

3.2.1 应用接口及相关结构体

接口 HI_UNF_HDCP_EncryptHDCPKey 可选择使用芯片内部保存的 HDCP root key 或海思自定义的秘密 key 两种方式中的一种对 HDCP key 进行加密。请注意:海思自定义的秘密 key 仅用于调试,量产时请使用客户自定义的 HDCP root key 对 HDCP key 进行加密保护。

/** Encrypted HDCP key */



3.2.2 Sample 介绍

HDCP root key 用于对从 DCP LLC.组织购买的 HDCP key 加密,其长度为 16bytes,客户自定义并维护其值。烧写 HDCP root key 到芯片内部 OTP 区域的方法请参考发布包中 sample: sample/otp/sample_otp_sethdcprootkey.c。

修改 sample_otp_sethdcprootkey.c 中待写入的 HDCP root key 值,编译并执行,即可烧写 HDCP root key。烧写完成后,HDCP root key 将会被锁定,锁住之后 HDCP root key 不可读。

生成加密的 HDCP key 数据。请参考发布包中 sample: sample/hdcpkey/sample encryptHdcpKey.c:

图3-6 执行 sample 生成加密的 HDCP key

```
# ./sample_encryptHdcpKey
Usage: ./sample_encryptHdcpKey num inputFilename
########HDCP SAMPLE########
[1] Encrypt HDCP Key using OTP Root key
[2] Encrypt HDCP Key using key defined by Hisilicon
```

烧写完 HDCP root key 后,执行./sample_encryptHdcpKey,选择使用 HDCP root key 加密 HDCP key。

- 如果执行:

./sample_encryptHdcpKey 1



10000_fake_key.bin_hisi_encry_v00000001_key0000001.bin

表示使用芯片内部 OTP 区域的 HDCP root key 加密 HDCP key。

- 如果执行:

./sample_encryptHdcpKey 2

10000_fake_key.bin_hisi_encry_v00000001_key0000001.bin

则表示使用海思自定义的秘密 key 加密 HDCP key。

□ 说明

输入数据 10000_fake_key.bin_hisi_encry_v00000001_key0000001.bin 为使用工具 HDCP_Key.exe 产生的长度为 384bytes 的分 key。

每颗芯片中海思自定义的秘密 key 的值相同,仅供调试使用,不建议用于量产。所以在运行 sample_encryptHdcpKey 时,第一个参数请配置为"1"。

3.2.3 工厂生产

3.2.3.1 数据准备

工厂生产前, 需预先准备以下数据:

- HDCP root key 数据
- 被加密的 HDCP key 数据

其中,HDCP root key 可自定义,长度为 16bytes,工厂生产时将被写入芯片内部 OTP 区域,lock 后不可读。

生成加密后的 HDCP kev 数据步骤如下:

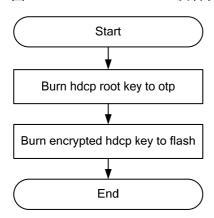
- 步骤 1 产生待加密的 HDCP key。使用 HDCP key 拆分工具拆分 HDCP key 包,将生成多个长度为 384bytes 的用于 HDCP 的分 key。详细用法请参考章节"3.1.2 Sample 介绍"。
- 步骤 2 生成加密的 HDCP key。使用海思 unf 接口加密 HDCP key,可选择使用芯片内部保存的根密钥或使用海思自定义的秘密密钥加密 HDCP key。详细用法请参考上文。

----结束



3.2.3.2 工厂生产流程

图3-7 Hi3712/Hi3716CV200 及后续芯片 HDCP key 方案工厂生产流程

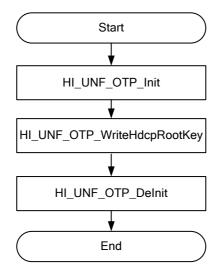


□ 说明

如果使用 HDCP root key 对 HDCP key 进行加密,预先准备的数据 HDCP root key 和 encrypted HDCP key 务必要匹配,否则将导致 HDCP key 校验失败,无法达到 HDMI 数据传输保护的目的。

其中,往芯片 OTP 区域中烧写并锁定 HDCP root key 的流程可参考。

图3-8 烧写并锁定 HDCP root key 流程





警告

HDCP root key 只能烧写一次, lock 后不可读取。



3.2.4 Linux 平台使能 HDCP 1.x 特性

3.2.4.1 打开 HDMI 对 HDCP 的支持

步骤 1 在 SDK 根目录下输入 make menuconfig, 进入 menuconfig 菜单。如图 3-9 所示。

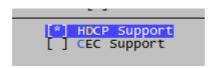
图3-9 menuconfig 菜单

```
Arrow keys navigate the menu. <Enter> selects submenus --->. Highlighted letters are hotkeys. Pressing </>
features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in [] excluded <M> module <> module capable

Base ---> Board ---> Uboot ---> Common ---> Component ---> Msp ---> Component ----> Component ----
```

进入 msp->HDMI config 下勾选 "HDCP support",如图 3-10 所示。

图3-10 HDCP 选项



步骤 2 sample 里打开 HDCP 选项。

打开 SDK/sample/common/hi_adp_hdmi.c 文件下的 HDCP 宏:

#define HI_HDCP_SUPPORT

图3-11 sample 默认配置

```
//#define HI_HDCP_SUPPORT
#ifdef HI_HDCP_SUPPORT
const HI_CHAR * pstencryptedHdcpKey = "EncryptedKey_332bytes.bin";
#endif
```

步骤 3 重新 SDK 和 Sample



3.2.4.2 HDMI 相关接口说明

接口 HI UNF HDMI LoadHDCPKey 用于导入加密后的 key:

HI_S32 HI_UNF_HDMI_LoadHDCPKey(HI_UNF_HDMI_ID_E enHdmi,
HI_UNF_HDMI_LOAD_KEY_S *pstLoadKey);

相关数据结构体如下:

/**HDMI HDCP key struct*/

typedef struct hiUNF_HDMI_LOAD_KEY_S

{

HI_U8 *pu8InputEncryptedKey; /**<Encrypted key pointer */</pre>

HI_U32 u32KeyLength; /**<Encrypted key length*/</pre>

}HI_UNF_HDMI_LOAD_KEY_S;

具体使用方式可参考 sdk 目录/sample/common/hi_adp_hdmi.c。

3.2.4.3 运行参考用例验证 HDCP key

步骤 1 使用海思工具 HDCP_Key.exe 生成一个 HDCP Key: org_xxxxx.bin;



注音

步骤 2 org_xxxxx.bin 为通过海思拆分工具 HDCP_Key.exe 生成的单个 HDCP 1.x key。使用 HDCP 的 sample 加密 org_xxxxx.bin:

./sample_encryptHdcpKey 1 org_xxxx.bin

将生成加密后的 HDCP Key 文件 EncryptedKey 332bytes.bin。

步骤 3 赋予 EncryptedKey 332bytes.bin 可读写权限:

chmod 777 EncryptedKey 332bytes.bin

- 步骤 4 存放 EncryptedKey_332bytes.bin 文件到 SDK/sample/hdmi_tsplay 目录下;
- 步骤 5 准备好需要播放的 TS 码流,也存放至 SDK/sample/hdmi tsplay 目录;
- 步骤 6 运行: ./sample hdmi tsplay stream.ts 1080i 50;
- 步骤7 输入: hdmi hdcp 1,此时开启 HDCP 握手过程。

步骤 7 完成后,可做如下判断:

- HDMI 模块初始化时,将解密并加载 flash 中加密的 HDCP key 数据到芯片内部,如果加载失败,将提示错误信息: Load HDCP key error!
- HDMI 模块加载 HDCP key 到芯片内部后,会对其进行校验,如果校验失败,将提示错误信息: HDCP crc check is error!
- 如果加载 HDCP key 或校验 HDCP key 失败,很有可能是因为加密的 HDCP key 和芯片内部的 key 不一致,重新生成 flash 中加密的 HDCP key 可解决此问题。



图 3-12、图 3-13、图 3-14 为 HDCP key 加载时部分调试信息截图,请参考。

图3-12 运行 HDMI sample 时没有准备被加密的 HDCP key

```
can't find key file
Set hdcp erro:0xffffffff
```

图3-13 HDMI 模块加载了错误的被加密的 HDCP key

```
--- Get HDMI event: HDCP_FAIL. ---
HDCP AUTO Check RI
[132360 ERROR-HI_HDMI]:SI_ReAthentication[870]:L[870] epst = 0x80.
[132380 ERROR-HI_HDMI]:SI_ReAthentication[870]:L[870] epst = 0x83.
[132386 ERROR-HI_HDMI]:SI_ReAthentication[875]:hdcp crc check is error!
--- Get HDMI event: HDCP_FAIL. ---
HDCP AUTO Check RI
[132820 ERROR-HI_HDMI]:SI_ReAthentication[870]:L[870] epst = 0x80.
[132840 ERROR-HI_HDMI]:SI_ReAthentication[870]:L[870] epst = 0x83.
[132846 ERROR-HI_HDMI]:SI_ReAthentication[875]:hdcp crc check is error!
```

图3-14 HDMI 模块加载 HDCP key 成功

```
[907880 ERROR-HI_HDMI]:SI_ReAthentication[870]:L[870] epst = 0x80.
[907904 ERROR-HI_HDMI]:SI_ReAthentication[870]:L[870] epst = 0x81.
--- Get HDMI event: HDCP SUCCESS. ---
```

----结束

3.2.5 Android 平台使能 HDCP 1.x 特性

请参考如下步骤:

步骤 1 通过海思提供的工具将 HDCP 1.x Key 包进行拆包,得到单个的 key 为 org_xxxxx.bin,这个 key 已经被工具完成加密。该 key 的大小为 384 个字节。



注音

org_xxxxx.bin 为通过海思拆分工具 HDCP_Key.exe 生成的单个 HDCP 1.x key。如需了解海思拆分工具 HDCP_Key.exe 的使用方法,请参考海思 SDK 发布包中文档《HDCP Key 工具使用指南》。

步骤 2 将步骤 1 中得到 384 字节的 key 放置到单板的可读写的目录,比如/mnt/sdcard/目录。 并且赋予该 key 可读可写的权限:比如: chmod 777 /mnt/sdcard/org_xxxxx.bin。



使用海思的 HiSysManager 类中的 setHdmiHDCPKey 接口,将加密的 org_xxxxx.bin 进行解密,解密完成后,用 otp 里的 hdcp root key 对其进行再次加密。加密后的 key 大小为 332 字节。并且将此 key 存放于 deviceinfo 分区中。

setHdmiHDCPKey 接口原型为:

public int setHdmiHDCPKey(String OrgKeyPath)

OrgKeyPath: org_xxxxx.bin 文件的路径字符串。如: "/mnt/sdcard/org_xxxxx.bin" setHdmiHDCPKey 接口调用完毕后,org_xxxxx.bin 将自动被删除。



注意

HDMI HDCP 的 Key 存放在 deviceinfo 分区尾部。

deviceinfo 分区的尾部 128K 已经被使用存放 DRM 的 Key 信息。版本中默认使用尾部的 128K+4K 的偏移量,作为存放 HDMI 的 HDCP Key 的位置。

此 Key 存放的位置支持自行配置,通过将偏移量写入 persist.sys.hdmi.hdcp.offset 来指 定 HDMI 的 HDCP Key 存放位置。注意,此偏移量的单位是字节,偏移是相对 deviceinfo 尾部的偏移,而且此偏移量必须大于 HDMI 的 HDCP Key 加上 DRM 的 Key 存放偏移(128K+4K:128*1024 + 4*1024)小于 deviceinfo 分区的总长减去 HDMI 的 HDCP Key 存放大小(2M – 4K:2*1024*1024 – 4*1024)。

若不自行配置存放 HDMI 的 HDCP Key 值,则默认使用海思的 deviceinfo 分区尾部的 128K+4K 的偏移量,作为存放 HDMI 的 HDCP Key 的位置。

步骤 3 修改 device/hisilicon/bigfish/frameworks/hidisplaymanager/hal/hi_adp_hdmi.c 文件,打开 HI HDCP SUPPRT 开关: #define HI HDCP SUPPORT。如图 3-16 所示。

图3-15 打开 HI_HDCP_SUPPRT 开关

- 47 static HDMI ARGS S g stHdmiArgs;
- 48 #define HI HDCP SUPPORT
- 49 #ifdef HI HDCP SUPPORT

步骤 4 重新编译 hidisplay.bigfish.so, 并 push 到单板/system/lib/hw 目录。

cd device/hisilicon/bigfish/frameworks/hidisplaymanager/hal m -B

push 生成的文件 out/target/product/Hi3798MV100/system/lib/hw/hidisplay.bigfish.so 到单板/system/lib/hw 目录。

此时重启单板,插上 HDMI 线,单板启动后 HDCP 1.x 特性默认使能。

----结束



4 注意事项

使用 $HDCP_{key.exe}$ 工具前,请务必确认当前芯片类型和与之对应的 $HDCP_{key}$ 方案,并请参考"3工具应用参考"章节进行相应的操作。