



SigmaStar Camera Verify Sample 说明



© 2019 SigmaStar Technology Corp. All rights reserved.

SigmaStar Technology makes no representations or warranties including, for example but not limited to, warranties of merchantability, fitness for a particular purpose, non-infringement of any intellectual property right or the accuracy or completeness of this document, and reserves the right to make changes without further notice to any products herein to improve reliability, function or design. No responsibility is assumed by SigmaStar Technology arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights, nor the rights of others.

SigmaStar is a trademark of SigmaStar Technology Corp. Other trademarks or names herein are only for identification purposes only and owned by their respective owners.



REVISION HISTORY

Revision No.	Description	Date
{000001}	• {Initial release}	{06/06/2019}



TABLE OF CONTENTS

REVISION HISTORY	i
TABLE OF CONTENTS.....	ii
1. 概述.....	1
1.1. 概述.....	1
2. RELEASE samPLE 说明 :	2
2.1. gfx:.....	2
2.2. disp+panel:.....	2
2.3. ai/ao:.....	2
2.4. vdec	2
2.5. application	3



1. 概述

1.1. 概述

文档主要描述 release 出来的 SDK 的测试 demo，主要是单模块或者多模块简单测试，用于指导客户对单模块封装 API 参考。

2. RELEASE SAMPLE 说明：

2.1. gfx:

gfx 是一个比较单纯的模块，它需要使用到物理内存，所以会引入 mi_sys 模块，具体操作包括了 quickfill、bitblit、blending 等参考。

代码位置：/sdk/verify/feature/gfx

编译方法：vi /project/release/customer_tailor/nvr_i2m_display_glibc_tailor.mk
在尾部加入 verify_gfx:=enable

然后在 cd /sdk/verify/feature;make gfx 即可

客户也可以自行编码 makefile 来编译 sample code（推荐）

注意：demo 中生成的文件格式以及生成的路径需要客户阅读代码获取，也可以自行定义相关内容

2.2. disp+panel:

代码主要实现了将 disp 和 panel 输出显示打通，从 disp 要一张 input buffer，填充 YUV 数据后推进 disp 模块，这样 panel 就可以看到显示你传入的相关 YUV 文件了。

代码位置：/sdk/verify/feature/disp

编译方法：vi /project/release/customer_tailor/nvr_i2m_display_glibc_tailor.mk
在尾部加入 verify_disp:=enable

修改/sdk/verify/feature/disp/config.mk，SRCS:= disp_ut.c panel_ut.c init_panel_driver.c

然后在 cd /sdk/verify/feature;make disp 即可，生成 bin 文件：prog

运行：

```
./prog -f YUV420SP_352_288.yuv -w 352 -h 288 -o ttl  
-f YUV 文件参数,文件格式为 semiplaner420,  
-w 在 panel 上显示的宽  
-h 在 panel 上显示的宽  
-o ttl/mipi(panel 类型, Demo 版本自带是 800*480 TTL)
```

2.3. ai/ao:

代码主要提供 ai/ao 的样片代码，包含了 capture、play、vqe、aec、nr 等音频相关功能使用

代码位置：/sdk/verify/mi_demo/geonosis/audio_all_test_case

编译方法：cd /sdk/verify/mi_demo/geonosis

make audio_all_test_case

音频接插以及运行方法参考：“SSD201_Demo 板音频使用说明.pdf”

2.4. vdec

代码主要实现了将 vdec 和 disp 串接起来，然后给 vdec 送 es 流，解码和 disp 绑定后，disp 会最终显示 es 流的画面。

代码位置：/sdk/verify/feature/disp

编译方法：vi /project/release/customer_tailor/nvr_i2m_display_glibc_tailor.mk
在尾部加入 verify_disp:=enable

修改/sdk/verify/feature/disp/config.mk，SRCS:= (disp_ut.c) vdec_ut.c init_panel_driver.c

另外 USE_CASE:=disp 也要做对应的修改，使用 vdec_ut.c 则改成 USE_CASE:=vdec_disp

然后在 `cd /sdk/verify/feature;make disp` 即可，生成 bin 文件：prog
运行：

```
./prog -f 720P265.h264 -w 640 -h 360  
-f H264 es 流,文件有加一个 header, 用于读 nalu, 在 application 的下载目录有一个 es 流  
-w 在 panel 上显示的宽(并不是指视频流的宽高, 视频流宽高需要在代码写)  
-h 在 panel 上显示的高
```

2.5. application

配合 demo 板提供的 demo 参考需要从 github 获取，获取到之后如：SSD20x_DisplayDemo-master.zip
也可以通过 git clone 的方式下载。

下载后 zip 包解压，然后将 application 目录拷贝到/sdk/verify 下面
编译：

```
cd /sdk/verify/application/smarttalk/demoApp  
./build_sample.sh
```

编译会生成 mginit 在 build 目录

如果需要做到开机运行，需要按如下流程执行：

- 1、修改/project/release/customer_tailor/nvr_i2m_display_glibc_tailor.mk，尾部添加
verify_smarttalk:=enable
- 2、cd /project
- 3、./setup_config.sh configs/nvr/i2m/8.2.1/nor.glibc-squashfs.011a.64
- 4、cd /sdk/verify/application/smarttalk/demoApp
- 5、./build_sample.sh
- 6、cd /project/image
- 7、make clean;make image //这样会去 application 目录取相关资源，然后打包进 images
- 8、这时候编译会自动进行将 application 相关的内容拷贝至 image 并打包成烧录 image

注：由于 flash 空间问题需要将如下 2 行注释掉

```
#@cp $(APPLICATION_PATH)/smarttalk/RunFile/720P25.h264 $(IMAGE_INSTALL_DIR)  
/tvconfig/config/  
#@cp $(APPLICATION_PATH)/smarttalk/RunFile/  
8K_16bit_MONO_30s.wav $(IMAGE_INSTALL_DIR)/tvconfig/config/
```