

广州天嵌计算机科技有限公司	文档编号	产品版本	密级	文件总类
	V1.0.0		保密	说明书
	产品名称：TQIMX6Q 硬件编解码库说明书			共：9 页

TQIMX6Q 硬件编解码库说明书

(内部使用，严禁外传)

文档版本	修改人	修改时间	备注
V1.0.0	Quentin	2017-05-03	初稿

目录

目录..... 二

产品介绍..... 三

 功能特性..... 三

 产品资料..... 三

产品制作与使用..... 六

 libtqvpu.so 的制作..... 六

 libtqvpu.so 的使用..... 六

 libtqvpu.so 使用流程图..... 七

 软件接口说明..... 八

 测试程序说明..... 九

产品介绍

TQIMX6Q 硬件编解码动态库 (libtqvpu.so) 是广州天嵌计算机科技有限公司基于 imx6q-linux4.1.15 内提供的 vpu 库 (libvpu.so) 制作而成的, 是 imx6q 系列板卡支持的硬件编解码库 (vpu lib) 的应用简化 API , 为方便客户使用 vpu 而做的。

TQIMX6Q 硬件编解码动态库通过软件编程、调用飞思卡尔官方的硬件编解码库, 实例化飞思卡尔官方硬件编解码库的接口方法、及功能, 实现对官方提供的复杂的硬件编解码库的再封装和简化, 从而方便客户更加方便地使用硬件编解码库。

功能特性

支持 imx6q-linux4.1.15 内核平台

支持多种格式的视频文件的编码、解码功能

支持以帧为单位编码、解码视频文件

支持视频文件的旋转编码、旋转解码

支持与 ipu 动态库 (libtqipu.so) 连用, 实现视频文件的编码、解码及图形图像处理的混合使用

注: 关于功能的具体信息参见 tqvpu_lib.h、vpu_lib.h、vpu_io.h 文件

产品资料

工程源码: TQIMX6Q_LIB-tqvpu, 含简单的例程

文件树:

TQIMX6Q_LIB_tqvpu/

- ├─ **configs**//没有用到的一些文件，测试用
 - ├─ akiyo.mp4
 - ├─ config_dec
 - ├─ config_enc
 - ├─ config_encdec
 - └─ config_net
- ├─ decode_test.c //解码测试例程
- ├─ decode_test.out//解码测试例程可执行文件
- ├─ encode_test.c//编码测试例程
- ├─ encode_test.out//编码测试例程可执行文件
- ├─ **include**//库必须的头文件，含 ipu 头文件等
 - ├─ dptc_daemon.h //vpu 库依赖头文件
 - ├─ dut_probes_vts.h //vpu 库依赖头文件
 - ├─ ipu.h //ipu 库头文件
 - ├─ mxc_ipu_hl_lib.h //vpu 库依赖头文件
 - ├─ mxc_test.h
 - ├─ rng_test_driver.h
 - ├─ scc2_test_driver.h
 - ├─ scc_test_driver.h
 - ├─ soc_check.h
 - ├─ tq_ipu.h //ipu 库头文件
 - ├─ tqvpu_lib.h //vpu 库依赖头文件
 - ├─ vpu_io.h //vpu 库依赖头文件
 - ├─ vpu_jpegtable.h //vpu 库依赖头文件
 - └─ vpu_lib.h //vpu 库依赖头文件
- ├─ **ipu** //ipu 库调用测试程序，处理解码后的数据在 lcd 上显示
 - ├─ ipu_test.c
 - ├─ ipu_test.h
 - ├─ lcd.c
 - ├─ lcd.h
 - └─ videodev2.h
- ├─ **lib** //编译依赖的库，及本产品编译生成库存放的位置
 - ├─ libipu.so//官方 ipu 依赖库
 - ├─ libipu.so.0//官方 ipu 依赖库
 - ├─ libtqipu.so//天嵌产品的 ipu 简易 API 库
 - ├─ libtqvpu.so//本产品（vpu）编译生成后的库
 - ├─ libvpu.a//官方 vpu 依赖库
 - ├─ libvpu.so//官方 vpu 依赖库
 - └─ libvpu.so.4//官方 vpu 依赖库
- ├─ **linux**//头文件依赖包
 - ├─ ipu.h
 - └─ isl29023.h

```
|   ├── mxc_asrc.h
|   ├── mxc_dcic.h
|   ├── mxcfb.h
|   ├── mxc_mlb.h
|   ├── mxc_sim_interface.h
|   ├── mxc_v4l2.h
|   ├── pxp_device.h
|   ├── pxp_dma.h
|   └── videodev2.h
└── Makefile//Makefile 实现直接 make 编译
```

目录说明：仅对硬件编解码库依赖作说明

./lib 下为硬件编解码使用时必须的动态链接库，其中含有天嵌产品的 ipu 库 (libtqipu.so)，libtq*.so 这样的文件均为天嵌产品的动态链接库，具体说明请参见相关说明文档。

./include、和./linux 的文件均为 *.h，为源码及./lib 内所有库所依赖。

其它的详见文件树说明。

产品制作与使用

libtqvpu.so 的制作

内核版本：imx6q -linux4.1.15

编译器：gcc version 5.3.1 20160113 (Linaro GCC 5.3-2016.02)

制作方法：拷贝 TQIMX6Q_LIB-tqvpu 目录到内核中，执行 make 命令。

执行示例：

```
pu_test/TQIMX6Q_LIB-tqvpu$ make
arm-linux-gnueabi-gcc -Wall -O2 -c decode_test.c -o decode_test.o
arm-linux-gnueabi-gcc -o decode_test.out decode_test.o ipu/lcd.o ipu/ipu_test.o
-lrt -lpthread -L ./lib -lvpu -lipu -ltqipu -ltqvpu
/opt/EmbedSky/gcc-linaro-5.3-2016.02-x86_64_arm-linux-gnueabi/bin/../lib/gcc/arm-
linux-gnueabi/5.3.1/../../../../arm-linux-gnueabi/bin/ld: warning: size of symb
ol 'fb_info' changed from 4 in ipu/lcd.o to 380 in ipu/ipu_test.o
arm-linux-gnueabi-gcc -Wall -O2 -c encode_test.c -o encode_test.o
arm-linux-gnueabi-gcc -o encode_test.out encode_test.o ipu/lcd.o ipu/ipu_test.o
-lrt -lpthread -L ./lib -lvpu -lipu -ltqipu -ltqvpu
/opt/EmbedSky/gcc-linaro-5.3-2016.02-x86_64_arm-linux-gnueabi/bin/../lib/gcc/arm-
linux-gnueabi/5.3.1/../../../../arm-linux-gnueabi/bin/ld: warning: size of symb
ol 'fb_info' changed from 4 in ipu/lcd.o to 380 in ipu/ipu_test.o
```

结果说明：Makefile 自动链接需要编译的硬件编解码库 libtqvpu.so，自动图形图像处理

库 libtqipu.so，编译硬件解码测试例程 decode_test.c，在源码根目录下生成 decode_test.out，

编译硬件编码测试例程 encode_test.c，在源码根目录下生成 encode_test.out。

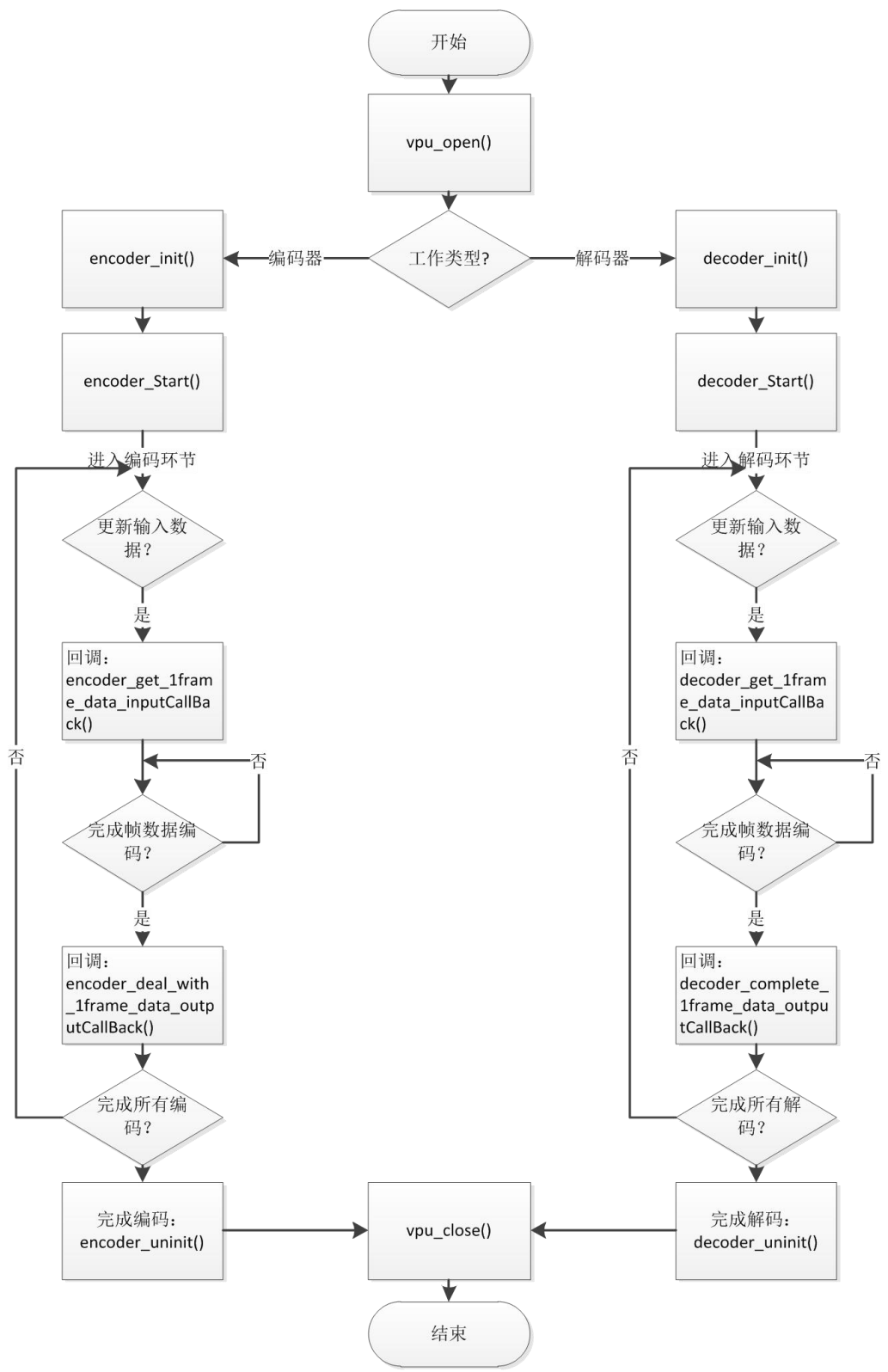
libtqvpu.so 的使用

源码移植：TQIMX6Q_LIB-tqvpu 源码下的 lib、include、linux 均为 libtqvpu.so 的

依赖文件，移植源码时必须确保所依赖的文件没有缺失，并且确保文件路径不被修改，否则会编译报错。

程序运行环境：调用 libtqipu.so、libtqvpu.so，或者其它的 libtq*.so 编译的源码程序，其可执行文件运行前必须搭建好环境，将相关的依赖库 libtq*.so 复制到板子文件系统上的 /usr/lib 目录下。

libtqvpu.so 使用流程图



软件接口说明

1. 对外开放的接口：

打开 vpu：vpu_open();

关闭 vpu：vpu_close();

解码器初始化：decoder_init();

解码器未初始化：decoder_uninit();

解码器开始工作：decoder_start();

解码器填入帧数据：decoder_get_1frame_data_inputCallBack();

解码器输出解码后帧数据：decoder_complete_1frame_data_outputCallBack();

编码器初始化：encoder_init();

编码器未初始化：encoder_uninit();

编码器开始工作：encoder_start();

编码器填入帧数据：encoder_get_1frame_data_inputCallBack();

编码器输出编码后帧数据：encoder_complete_1frame_data_outputCallBack();

2. 回调函数的重写：

回调函数接口的重写可参加例程，其中 ipu_test.c 文件中就有实例。

3. 需要重写的接口：

解码器填入帧数据：decoder_get_1frame_data_inputCallBack();

解码器输出解码后帧数据：decoder_complete_1frame_data_outputCallBack();

编码器填入帧数据：encoder_get_1frame_data_inputCallBack();

编码器输出编码后帧数据：encoder_complete_1frame_data_outputCallBack();

具体规则详见：tqvpu_lib.h、vpu_lib.h、vpu_io.h 等。

测试程序说明

解码实例：decode_test.c

命令：./decode_test.out -D "-i enc.264 -o test.264 -w 1280 -h 720 -f 0 -r 180"。

参数帮助：./decode_test.out -H 获取帮助，或者详见 tqvpu_lib.h、vpu_lib.h、vpu_io.h 等。

编码实例：encode_test.c

命令：./encode_test.out -E "-i enc.264 -o test.264 -w 1280 -h 720 -f 0 -r 180"。

参数帮助：./encode_test.out -H 获取帮助，或者详见 tqvpu_lib.h、vpu_lib.h、vpu_io.h 等。

具体的参数意义请查看：decode_test.c/encode_test.c 中的 `char *usage;`