



DragonBoard 测试使用指南

使用指南

版本号: 1.0
发布日期: 2022.02.14

版本历史

版本号	日期	制/修订人	内容描述
1.0	2022.02.14	AWA1685	初始版本

目 录

1 前言	1
1.1 简介	1
1.2 目的	1
2 DragonBoard 工具的使用	2
2.0.1 使用入门	2
2.0.2 配置测试项	2
2.0.3 编译 DragonBoard	2
2.0.3.1 配置环境变量	2
2.0.3.2 选择板级配置	2
2.0.3.3 测试用例前置条件	3
2.0.3.4 开启 DragonBoard	3
2.0.4 编译打包固件	4
2.0.5 烧写固件	4
2.0.5.1 SD 卡启动	4
2.0.5.2 板卡启动	4
2.0.6 测试流程	4
3 测试项配置与描述	6
3.0.1 模块配置示例	6
3.0.2 RTC 配置	6
3.0.2.1 测试状态	7
3.0.3 LED 配置	7
3.0.3.1 测试状态	7
3.0.4 U 盘配置	7
3.0.4.1 测试状态	8
3.0.5 DDR 配置	8
3.0.5.1 测试状态	8
3.0.6 TF 配置	8
3.0.6.1 测试状态	9
3.0.7 NOR flash 配置	9
3.0.7.1 测试状态	9
3.0.8 NAND falsh 配置	9
3.0.8.1 测试状态	9
3.0.9 CSI 配置	10
3.0.9.1 测试状态	10
3.0.10 MIC 配置	10
3.0.10.1 测试状态	11
3.0.11 VENC 配置	11
3.0.11.1 测试状态	11
3.0.12 VDEC 配置	12
3.0.12.1 测试状态	12

3.0.13 ISP 测试	13
3.0.13.1 测试状态	13
3.0.14 G2D 测试	13
3.0.14.1 测试状态	14
3.0.15 CE 配置	14
3.0.15.1 测试状态	14
3.0.16 VO 配置	14
3.0.16.1 测试状态	15



1 前言

1.1 简介

DragonBoard 是基于 Linux BSP 和 MPP，集成 MiniGUI 图形化板卡测试系统，旨在 PCBA 板测试，用于验证硬件的通路是否基本正常。DragonBoard 测试系统的固件可以直接烧录到板上，同时也支持卡启动，减少对 PC 的依赖。

DragonBoard 测试系统包括 DDR、RTC、VENC 等模块的自动测试，系统上电运行后会自动加载、运行用户启用的测试用例，并将结果显示到界面，用户可从中看到哪些测试项通过，哪些失败。

1.2 目的

本文档主要向用户阐述在 V853 平台上 DragonBoard 测试系统的使用方法及其测试用例的使用方法。

2 DragonBoard 工具的使用

2.0.1 使用入门

由于 DragonBoard 测试系统基于 Linux BSP，因此需要待测试平台的 Linux BSP，即 Linux 内核。

DragonBoard 测试系统的源码位于 external/eyesee-mpp/dragonboard 目录下，V853 SDK 默认不会开启 DragonBoard 需要在 menuconfig 中选中。

2.0.2 配置测试项

DragonBoard 测试系统提供了一个灵活的配置脚本 test_config.fex，方便用户定制自己的测试项目，从而提高系统的效率。test_config.fex 位于 external/eyesee-mpp/dragonboard/apps/DragonBoard 目录下，测试项目的各项配置说明详见第 2.2 节配置文件。

2.0.3 编译 DragonBoard

在 V853 SDK 根目录下执行如下指令：

2.0.3.1 配置环境变量

```
source build/envsetup.sh
```

2.0.3.2 选择板级配置

```
lunch

You're building on Linux

Lunch menu... pick a combo:
  1. v853_perfl-tina

Which would you like? [Default v853_perfl]: 1
=====
TINA_BUILD_TOP=/home/haoguo/workspace/exdroid/v853/tina-v853
```

```
TINA_TARGET_ARCH=arm
TARGET_PRODUCT=v853_perf1
TARGET_PLATFORM=v853
TARGET_BOARD=v853-perf1
TARGET_PLAN=perf1
TARGET_BUILD_VARIANT=tina
TARGET_BUILD_TYPE=release
TARGET_KERNEL_VERSION=4.9
TARGET_UBOOT=u-boot-2018
TARGET_CHIP=sun8iw2lp1
=====
clean buildserver
[1]+  Killed                  $T/tools/build/buildserver --path $T 2> /dev/null 1>&2
[1] 85471
```

2.0.3.3 测试用例前置条件

2.0.3.3.1 LED 测试打开 GPIO

- 开启内核选项

```
CONFIG_GPIO_SYSFS=y
```

2.0.3.3.2 CE 测试

- 系统配置

```
CONFIG_OPENSSL_ENGINE=y
CONFIG_OPENSSL_DYNAMIC_ENGINE=y
CONFIG_PACKAGE_libopenssl-afalg=y
```

- 内核选项

```
CONFIG_CRYPTO_USER_API_HASH=y
CONFIG_CRYPTO_USER_API_SKCIPHER=y
CONFIG_CRYPTO_USER_API_RNG=y
CONFIG_CRYPTO_USER_API_AEAD=y
CONFIG_CRYPTO_DEV_SUNXI=y
```

2.0.3.4 开启 DragonBoard

```
CONFIG_mpp_compile_dynamic_lib=y
CONFIG_PACKAGE_eyesee-mpp-dragonboard=y
```

2.0.4 编译打包固件

```
make -j32 && pack
```

2.0.5 烧写固件

2.0.5.1 SD 卡启动

使用 PhoenixCard 工具烧写制作卡启动固件，将之做好的启动卡，插入卡槽，上电，系统自动进入 DragonBoard 测试系统。

2.0.5.2 板卡启动

使用量产工具 PhoenixUSBpro 或升级工具 PhoenixSuit 将固件直接烧录到板卡上的 falsh 上，烧录完成系统自动进入 DragonBoard 测试系统。

2.0.6 测试流程

板卡上电之后系统自动启动，测试程序依照 test_config.fex 配置依次加载，并显示主界面。根据配置的不同，主页面的布局也将不同。当进入主页面之后，用户即看到各个测试项目的状态。待测试项为青色并显示 Waiting，测试项通过测试显示为绿色 PASS，不通过显示为红色 FAIL。



图 2-1：主页面

3 测试项配置与描述

DragonBoard 测试系统提供了一个灵活的配置脚本 `test_config.fex`，方便用户定制自己的测试项目，从而提高系统的效率。`test_config.fex` 位于 `external/eyesee-mpp/dragonboard/apps/DragonBoard` 目录下，方便用户定制自己的测试项目。

3.0.1 模块配置示例

```
[example]
enabled   = 1
category  = 0
run_type  = 0
```

- [example]

表示一个模块配置

- enabled

表示是否启用当前配置模块

- category

测试模式。0 代表自动测试模块，1 代表手动测试模块。

- run_type

0 代表当前等待测试程序执行完毕再运行下一个模块的测试程序。1 代表不等待当前模块的测试程序执行完毕。

3.0.2 RTC 配置

RTC 测试系统时钟，测试用例会获取三次系统事件并通过串口输出。

时钟测试配置如下。

```
[RTC]
enabled = 1
category = 0
run_type = 0
```

3.0.2.1 测试状态

测试正常，DragonBoard 显示主页面上 [RTC] 项显示为 [RTC]PASS，并且背景显示为绿色。

测试失败，DragonBoard 显示主页面上 [RTC] 项显示为 [RTC]FAIL，并且背景显示为红色。

测试等待，DragonBoard 显示主页面上 [RTC] 项显示为 [RTC]Waiting，并且背景显示为青色。

3.0.3 LED 配置

LED 测试 LED 指示灯，测试用例默认使用 GPIO 145，通过设置该 GPIO，判断其高低电平确认是否测试成功。

LED 指示灯测试配置如下。

```
[LED]
enabled = 1
category = 0
run_type = 0
```

3.0.3.1 测试状态

测试正常，DragonBoard 显示主页面上 [LED] 项显示为 [LED]PASS，并且背景显示为绿色。

测试失败，DragonBoard 显示主页面上 [LED] 项显示为 [LED]FAIL，并且背景显示为红色。

测试等待，DragonBoard 显示主页面上 [LED] 项显示为 [LED]Waiting，并且背景显示为青色。

3.0.4 U 盘配置

USB 测试板端 U 盘模式，需在板端启动前通过 OTG 连接线接入 U 盘。测试用例检测 PCBA 板是否可以正常识别到 U 盘设备，确认是否测试成功。

U 盘测试配置如下。

```
[USB]
enabled = 1
category = 0
runtype = 0
```

3.0.4.1 测试状态

测试正常，DragonBoard 显示主页面上 [USB] 项显示为 [USB]PASS，并且背景显示为绿色。

测试失败，DragonBoard 显示主页面上 [USB] 项显示为 [USB]FAIL，并且背景显示为红色。

测试等待，DragonBoard 显示主页面上 [USB] 项显示为 [USB]Waiting，并且背景显示为青色。

3.0.5 DDR 配置

DDR 测试 PCBA 板内存是否正常。

```
[DDR]
enabled = 1
category = 0
runtype = 0
```

3.0.5.1 测试状态

测试正常，DragonBoard 显示主页面上 [DDR] 项显示为 [DDR]PASS，并且背景显示为绿色。

测试失败，DragonBoard 显示主页面上 [DDR] 项显示为 [DDR]FAIL，并且背景显示为红色。

测试等待，DragonBoard 显示主页面上 [DDR] 项显示为 [DDR]Waiting，并且背景显示为青色。

3.0.6 TF 配置

TF 测试 PCBA 板是否可以正常识别到 TF 卡。

TF 测试配置如下。

```
[TF]
enabled = 1
category = 0
runtype = 0
```

3.0.6.1 测试状态

测试正常，DragonBoard 显示主页面上 [TF] 项显示为 [TF]PASS，并且背景显示为绿色。

测试失败，DragonBoard 显示主页面上 [TF] 项显示为 [TF]FAIL，并且背景显示为红色。

测试等待，DragonBoard 显示主页面上 [TF] 项显示为 [TF]Waiting，并且背景显示为青色。

3.0.7 NOR flash 配置

NOR 测试 PCBA 板 NOR flash 是否正常。

```
[NOR]
enabled = 1
category = 0
runtype = 0
```

3.0.7.1 测试状态

测试正常，DragonBoard 显示主页面上 [NOR] 项显示为 [NOR]PASS，并且背景显示为绿色。

测试失败，DragonBoard 显示主页面上 [NOR] 项显示为 [NOR]FAIL，并且背景显示为红色。

测试等待，DragonBoard 显示主页面上 [NOR] 项显示为 [NOR]Waiting，并且背景显示为青色。

3.0.8 NAND flash 配置

NAND 测试 PCBA 板 NAND flash 是否正常。

```
[NAND]
enabled = 1
category = 0
runtype = 0
```

3.0.8.1 测试状态

测试正常，DragonBoard 显示主页面上 [NAND] 项显示为 [NAND]PASS，并且背景显示为绿色。

测试失败，DragonBoard 显示主页面上 [NAND] 项显示为 [NAND]FAIL，并且背景显示为红色。

测试等待，DragonBoard 显示主页面上 [NAND] 项显示为 [NAND]Waiting，并且背景显示为青色。

3.0.9 CSI 配置

CSI 测试摄像头采集通路，测试用例会打开一路摄像头采集通路，并将采集的图像保存在/tmp 目录下。

保存的摄像头采集图像名称为 csitester_result.yuv，该文件为分辨率为 640x480 格式为 nv21 的 YUV 图像，可使用 YUVView 工具查看。

```
root@TinaLinux:/tmp# ll /tmp/csitester_result.yuv
-rw-r--r--  1 root   root   9216000 Jan  1 01:23 /tmp/csitester_result.yuv
```

CSI 测试配置如下。

```
[CSI]
enabled = 1
category = 0
runtype = 0
```

3.0.9.1 测试状态

测试正常，DragonBoard 显示主页面上 [CSI] 项显示为 [CSI]PASS，并且背景显示为绿色。保存的摄像头采集图像可以正常打开并正常显示。

测试失败，DragonBoard 显示主页面上 [CSI] 项显示为 [CSI]FAIL，并且背景显示为红色。摄像头采集图像没有保存或使用工具打开时出现异常。

测试等待，DragonBoard 显示主页面上 [CSI] 项显示为 [CSI]Waiting，并且背景显示为青色。

 说明

YUVView工具可在网址：<https://github.com/IENT/YUVview/releases/tag/v2.12.1> 上下载

3.0.10 MIC 配置

MIC 测试音频采集通路，测试用例会打开一路音频采集通路，并将采集到的音频数据，直接通过喇叭进行播放。

正常测试情况下，PCBA 板的喇叭会将 MIC 采集的音频进行播放。

MIC 测试配置如下。

```
[MIC]
enabled = 1
```

```
category = 0  
runtype = 0
```

3.0.10.1 测试状态

测试正常，DragonBoard 显示主页面上 [MIC] 项显示为 [MIC]PASS，并且背景显示为绿色。PCBA 板上的喇叭播放出采集的音频数据。

测试失败，DragonBoard 显示主页面上 [MIC] 项显示为 [MIC]FAIL，并且背景显示为红色。PCBA 板上的喇叭没有播放出采集的音频数据。

测试等待，DragonBoard 显示主页面上 [MIC] 项显示为 [MIC]Waiting，并且背景显示为青色。

3.0.11 VENC 配置

VENC 测试视频编码通路，将采集到的摄像头图像数据，送入到 VENC 视频编码模块，编码为 H264 格式视频，并将其封装为 MP4 格式文件。

测试用例将编码后并进行封装的视频文件，在/tmp 目录下保存为 ve_tester_result.mp4 文件。

```
root@TinaLinux:/tmp# ll ve_tester_result.mp4  
-rw-r--r-- 1 root root 3049692 Jan 1 01:53 ve_tester_result.mp4
```

VENC 测试配置如下。

```
[VENC]  
enabled = 1  
category = 0  
runtype = 0
```

3.0.11.1 测试状态

测试正常，DragonBoard 显示主页面上 [VENC] 项显示为 [VENC]PASS，并且背景显示为绿色。保存的视频编码文件可以正常打开并播放。

测试失败，DragonBoard 显示主页面上 [VENC] 项显示为 [VENC]FAIL，并且背景显示为红色。保存的视频编码文件打不开或者不可以播放。

测试等待，DragonBoard 显示主页面上 [VENC] 项显示为 [VENC]Waiting，并且背景显示为青色。

3.0.12 VDEC 配置

VDEC 测试视频解码，通过将一段 MP4 文件解码播放，来测试硬件解码模块是否正常。

VDEC 测试配置如下。

```
[VDEC]
enabled = 1
category = 0
runtype = 0
```

3.0.12.1 测试状态

测试正常，DragonBoard 显示主页面上 [VDEC] 项显示为 [VDEC]PASS，并且背景显示为绿色。解码并播放的视频显示在主页面下方。



图 3-1: vdectestester 页面

测试失败，DragonBoard 显示主页面上 [VDEC] 项显示为 [VDEC]FAIL，并且背景显示为红色。视频文件解码失败并无法显示。

测试等待，DragonBoard 显示主页面上 [VDEC] 项显示为 [VDEC]Waiting，并且背景显示为青色。

3.0.13 ISP 测试

ISP 测试 ISP 通路，ISP 处理摄像头采集到的图像，对其图像效果进行优化处理。

测试用例会将摄像头采集图像经过 ISP 处理优化，在/tmp 目录下保存为 fb0_yuv.bin，该文件分辨率为 512x288，nv21 格式的 YUV 图像。可使用 YUVView 工具查看。

ISP 配置如下。

```
[ISP]
enabled = 1
category = 0
runtype = 0
```

📖 说明

YUVView工具可在网址：<https://github.com/IENT/YUVview/releases/tag/v2.12.1> 上下载

3.0.13.1 测试状态

测试正常，DragonBoard 显示主页面上 [ISP] 项显示为 [ISP]PASS，并且背景显示为绿色。保存的文件可以使用工具正常打开并正常显示。

测试失败，DragonBoard 显示主页面上 [ISP] 项显示为 [ISP]FAIL，并且背景显示为红色。保存的文件使用工具无法打开或无法显示。

测试等待，DragonBoard 显示主页面上 [ISP] 项显示为 [ISP]Waiting，并且背景显示为青色。

3.0.14 G2D 测试

G2D 为硬件 2D 图像加速模块，通过将一张 480x320 nv21 格式的 YUV 进行 90 度旋转，测试该模块是否正常。

经过 G2D 90 度旋转的图片，在/tmp 目录下保存为 g2d_tester_result.yuv 文件，该文件分辨率为 320x480，nv21 格式的 YUV 图像。可使用 YUVView 工具打开。

G2D 测试配置如下。

```
[G2D]
enabled = 1
category = 0
runtype = 0
```

3.0.14.1 测试状态

测试正常，DragonBoard 显示主页面上 [G2D] 项显示为 [G2D]PASS，并且背景显示为绿色。保存的文件可以使用工具正常打开并正常显示。

测试失败，DragonBoard 显示主页面上 [G2D] 项显示为 [G2D]FAIL，并且背景显示为红色。保存的文件使用工具无法打开或无法显示。

测试等待，DragonBoard 显示主页面上 [G2D] 项显示为 [G2D]Waiting，并且背景显示为青色。

 说明

YUVView工具可在网址：<https://github.com/IENT/YUVView/releases/tag/v2.12.1> 上下载

3.0.15 CE 配置

CE 测试硬件加密模块，通过将摄像头采集的图像进行编码，获取到编码码流后将每一笔编码码流进行加密并解密，再与未加密之前的编码帧对比，来测试硬件加密模块是否正常。

CE 测试配置如下。

```
[CE]
enabled = 1
category = 0
runtype = 0
```

3.0.15.1 测试状态

测试正常，DragonBoard 显示主页面上 [CE] 项显示为 [CE]PASS，并且背景显示为绿色。

测试失败，DragonBoard 显示主页面上 [CE] 项显示为 [CE]FAIL，并且背景显示为红色。

测试等待，DragonBoard 显示主页面上 [CE] 项显示为 [CE]Waiting，并且背景显示为青色。

3.0.16 VO 配置

VO 测试显示功能，通过将摄像头采集到的图像直接显示在主页面下方，来测试显示功能是否正常。



图 3-2: votester 页面

VO 测试配置如下。

```
[VO]
enabled = 1
category = 0
runtype = 0
```

3.0.16.1 测试状态

测试正常，DragonBoard 显示主页面上 [VO] 项显示为 [VO]PASS，并且背景显示为绿色。

测试失败，DragonBoard 显示主页面上 [VO] 项显示为 [VO]FAIL，并且背景显示为红色。

测试等待，DragonBoard 显示主页面上 [VO] 项显示为 [VO]Waiting，并且背景显示为青色。

著作权声明

版权所有 © 2022 珠海全志科技股份有限公司。保留一切权利。

本文档及内容受著作权法保护，其著作权由珠海全志科技股份有限公司（“全志”）拥有并保留一切权利。

本文档是全志的原创作品和版权财产，未经全志书面许可，任何单位和个人不得擅自摘抄、复制、修改、发表或传播本文档内容的部分或全部，且不得以任何形式传播。

商标声明

、 **全志科技** （不完全列举）均为珠海全志科技股份有限公司的商标或者注册商标。在本文档描述的产品中出现的其它商标，产品名称，和服务名称，均由其各自所有人拥有。

免责声明

您购买的产品、服务或特性应受您与珠海全志科技股份有限公司（“全志”）之间签署的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所购买或使用的范围内。使用前请认真阅读合同条款和相关说明，并严格遵循本文档的使用说明。您将自行承担任何不当使用行为（包括但不限于如超压，超频，超温使用）造成的不利后果，全志概不负责。

本文档作为使用指导仅供参考。由于产品版本升级或其他原因，本文档内容有可能修改，如有变更，恕不另行通知。全志尽全力在本文档中提供准确的信息，但并不确保内容完全没有错误，因使用本文档而发生损害（包括但不限于间接的、偶然的、特殊的损失）或发生侵犯第三方权利事件，全志概不负责。本文档中的所有陈述、信息和建议并不构成任何明示或暗示的保证或承诺。

本文档未以明示或暗示或其他方式授予全志的任何专利或知识产权。在您实施方案或使用产品的过程中，可能需要获得第三方的权利许可。请您自行向第三方权利人获取相关的许可。全志不承担也不代为支付任何关于获取第三方许可的许可费或版税（专利税）。全志不对您所使用的第三方许可技术做出任何保证、赔偿或承担其他义务。