



# H6 DragonBox 使用说明 书

confidential

V1.0  
2017.07.12

## 文档履历

版本号	日期	制/修订人	内容描述
V0.9	2017.07.01	AWA0279	初始版本
V1.0	2017.07.12	AWA0961	增加、规范测试项

Confidential

# 目录

1. 概述 . . . . .	1
1.1 编写目的 . . . . .	1
1.2 适用范围 . . . . .	1
1.3 相关人员 . . . . .	1
2. 功能与工具介绍 . . . . .	3
3. 测试步骤及使用方法 . . . . .	4
3.1 制作启动 U 盘/SD 卡 . . . . .	4
3.2 修改配置 . . . . .	4
3.3 测试界面 . . . . .	6
3.4 功能与判断标准 . . . . .	6
4. 模块功能与流程设计 . . . . .	9
4.1 使用流程 . . . . .	9
4.2 功能模块 . . . . .	10
4.3 扩展测试用例 . . . . .	10
4.4 常用接口说明 . . . . .	13
5. Declaration . . . . .	15

# 1. 概述

## 1.1 编写目的

描述 DragonBox 工厂测试工具的现有实现方式、代码结构、模块功能等。以便相关人员快速熟悉二次开发或软件维护等。

## 1.2 适用范围

适用 Android 4.0 以上，SDK 支持列表如下：

芯片平台	SDK 版本	内部发布版本（或以上）	是否支持
A10s	4.0	2.0	是
A20	4.0	1.2	是
A20	4.2	2.0	是
A31	4.2	1.0	是
A31s	4.4.2	4.4	是
A80	4.4.2	1.0	是
H8	4.4.2	1.0	是
H3	4.4.2	1.0	是
H64	5.1	1.0	是
H5	5.1	3.0	是
H5	7.0	4.0	是
H6	4.4	4.0	是
H6	7.0	4.0	是

## 1.3 相关人员

开发人员：罗伯龙、黄龙、郑湘娜、李思东、曲军政等。

使用人员：工厂产测人员。

Confidential

## 2. 功能与工具介绍

该工具使用于工厂，用于测试机器是否能正常工作。当机器出厂前，都需要使用该工具测试运行，过滤明显的不良机器。应用可于二次开发扩展需要的测试项。现有的测试项如下为：

序号	测试项目名称	自动化程度	重测支持	硬件需求
1	版本测试	自动	支持	无
2	LED 测试	半自动	支持	至少 2 个 LED
3	SPDIF 测试	半自动	支持	自带 spdif 输出
4	视频测试	半自动	支持	连接显示器
5	性能测试	自动	支持	无
6	USB 测试	自动	支持	插入 U 盘
7	SD 卡测试	自动	支持	有 sd 卡槽并插入 sd 卡
8	以太网测试	自动	支持	连接有线以太网
9	WIFI 测试	自动	支持	连接 wifi
10	HDMI 测试	半自动	支持	以 hdmi 方式连接显示器
11	CVBS 测试	半自动	支持	以 cvbs 方式连接显示器
12	Speaker 测试	半自动	支持	自带 Speaker 输出
13	录音测试	半自动	支持	自带 MIC 输入或者连接 usb MIC
14	Camera 测试	半自动	支持	自带 Camera 输入或者连接 usb Camera
15	蓝牙测试	自动	支持	自带蓝牙模块

## 3. 测试步骤及使用方法

### 3.1 制作启动 U 盘/SD 卡

如图，在 U 盘或 SD 卡根目录建立一个名为 DragonBox 的文件夹，将 custom\_cases.xml 文件拷贝到 DragonBox 文件夹下：

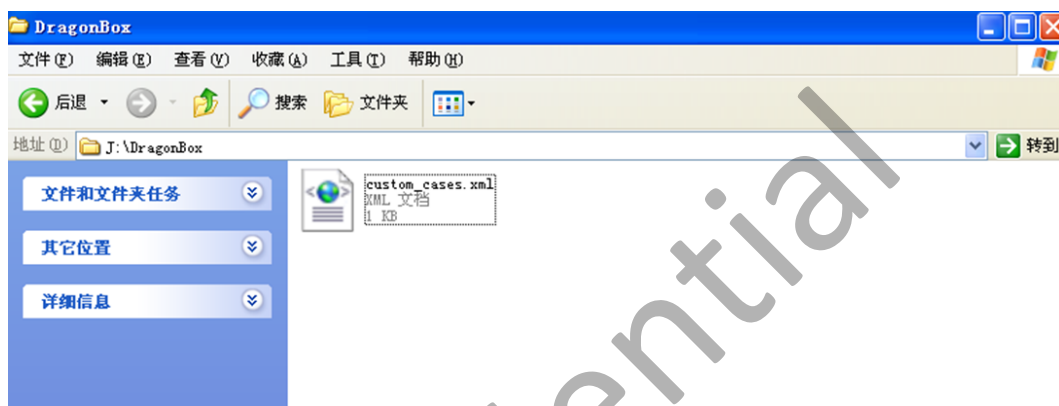


图 1: DragonBox 测试文件夹

custom\_cases.xml 文件可以到以下路径获取：

```
/DragonBox/assets/default_cases.xml
```

### 3.2 修改配置

针对不同的平台，custom\_cases.xml 需要一定的修改。

```
<CaseVersion fireware="xxx" display="xxx" model="xxx"></CaseVersion>
<CaseLed></CaseLed>
<CaseSpdif></CaseSpdif>
<CaseVideo></CaseVideo>
<CasePerformance></CasePerformance>
```

```
<CaseUsbVolume num="xxx"></CaseUsbVolume>
<CaseSDVolume></CaseSDVolume>
<CaseEthernet></CaseEthernet>
<CaseWifi maxRSSI="xxx" wifiSSID="xxx" wifiPWD="xxx"></CaseWifi>
<CaseBluetooth></CaseBluetooth>
<CaseHdmi></CaseHdmi>
<CaseSpeaker></CaseSpeaker>
<CaseMicRecorded recordedtime="xxx"></CaseMicRecorded>
<CaseCamera recordedtime="xxx"></CaseCamera>
<CaseCvbs></CaseCvbs>
```

上表中标记为 ``xxx" 的值都需要针对测试环境进行修改，如

测试项	键名	键值	注释
CaseVersion	fireware	ro.product.firmware	系统版本号
CaseVersion	display	ro.build.display.id	编译信息 ID
CaseVersion	model	ro.product.model	产品系列名称
CaseWifi	maxRSSI	Wifi RSSI (db)	保证的最低连接信号强度 (负数), 建议数值"-65"
CaseWifi	wifiSSID	Wifi SSID	Wifi 名称 (SSID)
CaseWifi	wifiPWD	Wifi Password	Wifi 密码
CaseMicRecorded	recordedtime	大于 0 的整数	录音的时间 (秒), 建议数值"5"
CaseCamera	recordedtime	大于 0 的整数	录像的时间 (秒), 建议数值"5"
CaseUsbVolume	num	大于 0 的整数	建议数值为机器上 USB 口数量

其中 fireware、display、model 可以在编译后的系统中的/system/build.prop 文件中复制获取，maxRSSI、wifiSSID、wifiPWD 值需要按照测试环境的 wifi SSID、密码进行配置，注意 maxRSSI，这是用来测试 wifi 连接强度的门限值，此值为负数，其绝对值越小，表示 wifi 连接得越稳定。在工厂生产中常常有 wifi 天线焊接不良的情况，这种情况下往往 wifi 是连接上的，但 wifi 信号很弱，因此需要一个测试门限值进行判断，通常取距离 wifi 5~10m 距离内，RSSI 值大于 -65db 为通过，否则不通过。recordedtime 表示录音或者录像



的时间，太长影响工厂的测试效率。num 表示需要测试的 U 盘挂载数量，通常取机器上 USB 口的数量，测试时，每一个 USB 口都插入 U 盘。

### 3.3 测试界面

插入 U 盘或 SD 卡，启动 DragonBox 应用，应用界面如下：

1	[版本测试]	固件版本:homlet4.4.2-h8-v1.0rc2		设备型号:eagle_fvd_p1-eng 4.4.2 KOT49H 20141103 test-keys		版本:eagle	[成功]
		Wifi MAC:08:d8:33:03:bf:95	eth MAC:36:c9:e3:f1:b8:05	DDR:746 MB	NAND:4.87 GB		
2	[Led测试]	LED 测试结果:					等待测试
3	[Spdif测试]	左声道测试:			右声道测试:		等待测试
4	[视频测试]	视频测试结果:					等待测试
5	[性能测试]	CPU最低频率: 480MHZ		CPU最高频率: 2016MHZ		CPU测试状态: [测试成功]	[成功]
6	[USB测试]	USB卡挂载:[成功]			USB读写测试:[成功]		[成功]
7	[SD测试]	SD卡挂载:[失败]		[Led测试]		[失败]	[失败]
8	[以太网测试]	网线连接状态: [已连接]		请观察LED灯能否点亮, 且闪烁03			[成功]
9	[WIFI测试]	WIFI热点:[PD1-test]	IP地址: [192.168.5	[成功]	[失败]	[30B]	WIFI状态: [搜索结束]
10	[HDMI测试]	HDMI接入状态:			HDMI声音输出:		等待测试
11	[CVBS测试]	CVBS状态:		左声道测试:		右声道测试:	等待测试

图 2: DragonBox 启动界面

首先进入的是自动测试模式，版本号、wifi、以太网、USB、SD 卡等可自动化测试的项目会首先进行自动化测试，LED、HDMI、CVBS、音频等需要人工参与的项目会提示对话框确认是否测试正常，测试结果会在最右边的结果栏显示出来，绿色表示通过，红色表示测试失败。如果某项测试没有通过，可以使用遥控器移动最左边的光标到需要重测的项目，按下遥控器 OK 键进行重测。

### 3.4 功能与判断标准

下表为各个功能判断需求与判断标准：

测试项目	需求内容	案例规范	判断标准	人工判断
版本测试	显示系统基础信息，包括硬件信息、固件信息、特殊号码	1. 显示固件版本号 2. 显示设备型号 3. 显示版本信息 4. 显示wifi和以太网MAC地址 5. 显示SN 6. 显示DDR信息 7. 显示NAND信息	1. 配对固件版本的号 2. 配对设备信号 3. 配对版本信息 4. 可获取除SN号的其他号码（非空）	否
TF卡测试	测试TF卡能否被检测到并挂载成功，且可读写	1. 显示挂载信息 2. 显示读结果 3. 显示写结果 4. 根据设备节点的major为179，判断为TF卡	1. 挂载成功 2. 可读写	否
USB存储测试	测试USB存储能否被检测到，并挂载成功	1. 显示挂载信息 2. 显示读结果 3. 显示写结果 4. 根据设备节点的major为8，判断为U盘。 5. 可以在配置文件中配置要测试的U盘数量	1. 每一个U盘均挂载成功 2. 每一个U盘均可读写	否
HDMI测试	测试切换HDMI输出显示，声音输出是否正常，显示RGB颜色	1. HDMI有输出，显示RGB颜色块 2. 播放音频 3. 测试时系统音量调到最大 4. 测试时系统音频通道为audio_hdmi，只有hdmi发声 5. 测试完毕后，将音量和音频通道恢复至测试前状态。	1. HDMI输出正常 2. HDMI声音输出正常 3. HDMI输出显示RGB颜色正常	是
CVBS测试	测试切换CVBS输出显示，左右声道输出是否正常，显示RGB颜色	1. CVBS有输出，显示RGB颜色块 2. 分别播放左右音频，显示左右声道测试结果 3. 测试时系统音量调到最大 4. 测试时系统音频通道为audio_codec，只有cvbs发声 5. 测试完后将音量和音频通道恢复至测试前状态。	1. CVBS输出正常 2. CVBS左右声道输出正常 3. CVBS输出显示RGB颜色正常	是
以太网测试	测试以太网网络连通性，判断是否已获取eth0的ip地址。以太网是否接入。	1. 显示网口接入状态 2. 显示获取IP地址	1. 物理网线接入 2. IP地址已获取	否
WIFI测试	测试wifi连通性，与配置文件设置默认连接的host和最低强度值，连接成功且强度满足要求	1. 显示所有WIFI热点。已配置热点排序优先。 2. 显示配置热点连接情况 3. 显示采样情况	1. 可获取热点 2. 可连接已配置的热点 3. 信号强度采样3次，需达到参考样机强度	否

图 3: DragonBox 测试标准

蓝牙测试	测试蓝牙设备扫描	1. 显示蓝牙扫描结果	1. 可扫描蓝牙信息	否
视频测试	测试播放视频能否成功，声音是否正常	1. 显示循环播放的视频与播放进度状态 2. 测试时系统音量调至最大。 3. 测试时音频通道为 audio_hdm 和 audio_codec，hdm 和 cvbs 同时发声。 4. 测试完毕后，将系统音量和音频通道恢复至测试前状态。	1. 视频循环播放正常 2. 声音输出正常	是
LED测试	测试LED是否点亮（闪烁）	1. LED蓝红交替闪烁	1. LED蓝红交替闪烁	是
Spdif测试	测试Spdif声音输出左右声道是否正常	1. 分别播放左右声道。 2. 显示左右声道的测试结果 3. 测试时系统音量调至最大。 4. 测试时音频通道为 audio_spdif，只有 spdif 发声。 5. 测试完后将系统音量和音频通道恢复至测试前状态。	1. 左右声道分别正常	是
性能测试	CPU切频测试	切换最低、最高频率，显示当前频率	1. 依次切换cpu频率，系统无异常	否
Speaker测试	测试Speaker声音输出是否正常	1. 播放音频 2. 测试时将系统音量调至最大 3. 测试时系统音频通道为 audio_speaker 4. 测试完后将系统音量和音频通道恢复至测试前状态。	1. 机器上的喇叭有声音	是
录音测试	测试Mic的录音功能	1. 按照配置文件中设置的时间录音 2. 回放录音 3. 测试结束后，删除录音文件。	1. 回放录音正常，与录音内容一致	是
录像测试	测试摄像头的录像功能	1. 按照配置文件中设置的时间录像 2. 回放录像 3. 测试结束后，删除录像文件。	1. 回放录像正常，画质清晰，没有明显的偏色现象，与录像时的预览画面一致。	是

图 4: DragonBox 测试标准

## 4. 模块功能与流程设计

### 4.1 使用流程

下图为一般测试流程，在启动时，读取配置好的 xml 后，分类型启动测试项，需要人工交互的测试会独立以 Dialog 方式弹出并需要用户选择判断结果，可自动判断的测试项会自动测试并判断结果。当测试完成后回到主界面，可以独立测试单个测试项。

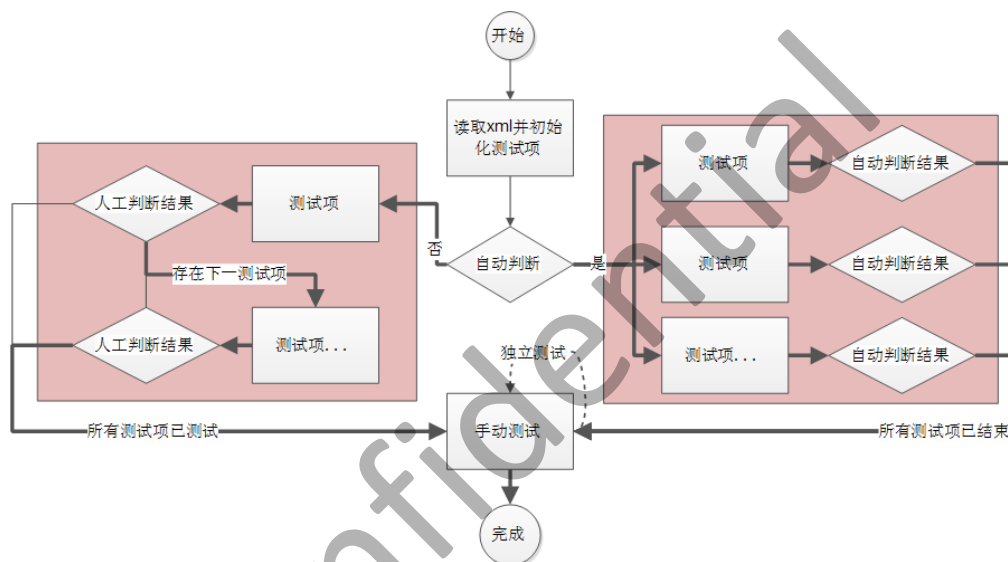


图 5: DragonBox 流程

## 4.2 功能模块

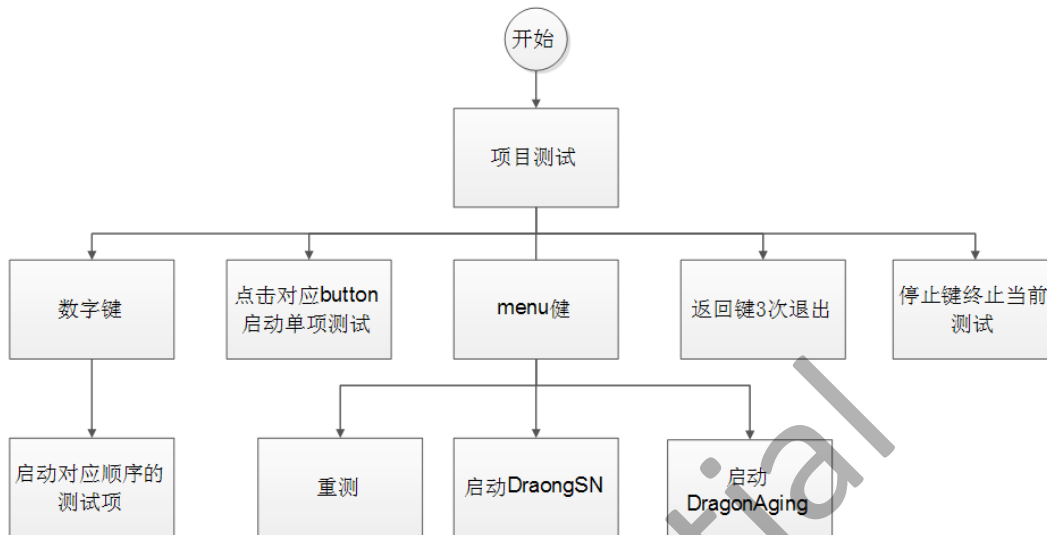


图 6: DragonBox 功能

## 4.3 扩展测试用例

客户或者 Dragon 工具维护工程师可以根据实际需求，在现有代码基础上快速进行二次开发，添加新的测试用例。代码结构如下：

```

|—— assets: DragonBox 的参考配置文件和的一些视频、声音文件。
|—— platform: 放置不同平台的功能接口代码，提高应用兼容性。
|—— res: 放置应用配置文件
|—— src
|   |—— com
|       |—— softwinner
|           |—— dragonbox
|               |—— config: 解析配置文件，生成测试用例
|               |—— entity: 测试用例涉及的实体类
|               |—— manager: 封装系统接口，进行功能测试
|               |—— platform: 一些接口类
  
```

—— **testcase**: 包含所有的测试用例  
—— **utils**: 使用到的工具类  
—— **view**: 显示列表用到的数据适配器

添加测试用例的具体步骤如下:

在 **testcase** 目录下新建新的测试类, 如 **CaseTest.java** (下面均以 **CaseTest.java** 为例说明)。

1. **CaseTest** 类需要继承所有测试类的父类 **IBaseCase**
2. 同时覆盖父类未实现的方法 **public void onStartCase()** 和 **public void onStopCase()**。这两个方法分别在该测试用例开始测试时和停止测试时被调用到, 开发人员可以根据测试用例的实际需求、设计思想或者参考已有测试用例填写这两个方法。
3. 建议覆盖父类 **IBaseCase** 中的方法 **public void reset()**。该方法用于测试结束后按遥控器 **menu** 键进行重新测试时, 对资源进行释放或重置。
4. 添加 **CaseTest** 的构造函数, 传参必须为两个: **Context** 和 **XmlPullParser**。不然在解析配置文件 **custom\_cases.xml** 构造测试用例时会因找不到构造函数而反射失败。在构造函数中需要通过 **super** 向父类传递: 测试用例的名字、测试时显示 **dialog** 使用的布局、测试按钮右边显示测试用例详细执行结果使用的布局、测试类型 (手动或者自动)。测试用例使用的布局需要开发人员根据需求来设计。

代码如下:

```
public class CaseTest extends IBaseCase {
    Context mContext;
    @Override
    public void onStartCase() {
    }

    public CaseTest(Context context, XmlPullParser xmlParser) {
        /**
         * @param context
         * @param R.string.case_test_name the name of the button
         * @param R.layout.case_test_max the dialog's view of this test case when testing
         * @param R.layout.case_test_min the main mini view on the right of buttons
         * @param TYPE_MODE_AUTO or TYPE_MODE_MANUAL
         */
    }
}
```

```
super(context, R.string.case_test_name, R.layout.case_test_max,  
        R.layout.case_test_min, TYPE_MODE_AUTO);  
mContext = context;  
}  
  
@Override  
public void onStopCase() {  
}  
  
@Override  
public void reset() {  
    // TODO Auto-generated method stub  
    super.reset();  
}  
}
```

写好新的测试用例后，需要将测试用例添加到 ConfigManager.java 中，以便解析 U 盘或者 SD 卡中的配置文件时匹配到对应的测试用例。代码如下：

```
private static final Map<String, Class<? extends IBaseCase>> mAllValidCases  
    = new HashMap<String, Class<? extends IBaseCase>>() {  
    private static final long serialVersionUID = 1L;  
    {  
        put(CaseHdmi.class.getSimpleName(), CaseHdmi.class);  
        .....  
        put(CaseYSTMicRecord2.class.getSimpleName(), CaseYSTMicRecord2.class);  
        put(CaseTest.class.getSimpleName(), CaseTest.class);  
    }  
};
```

接下来只需要在配置文件 custom\_cases.xml 中添加以新测试用例的类名为 key 的测试项即可。代码如下：

```
<CaseTest></CaseTest>
```

## 4.4 常用接口说明

下面提供一些编写测试用例时经常用到的 api

```
/**
 * 设置点击测试用例后弹出的 Dialog 中的"通过"按钮是否可用
 * @param enable true enable or false disable
 */
public void setDialogPositiveButtonEnable(boolean enable){
    .....
}
```

```
/**
 * 设置执行测试用例时弹出的对话框中是否包含 Dialog 自带的"通过"和"失败"按钮。
 * 当不显示 Dialog 自带的 Button 时，可以在 Layout 布局文件中自定义"通过"和"失败"按钮。
 * 适用于测试项包含多个子测试项，如 CVBS 测试包含左声道测试和右声道测试
 * @param showDialogButton true 显示通过和失败按钮，false 时不显示通过失败按钮
 */
public void setShowDialogButton(boolean showDialogButton){
    .....
}
```

```
/**
 * 设置该测试用例的测试结果：成功或失败，该接口仅在 setShowDialogButton(false)
 * 添加自定义的"通过"和"失败"按钮时使用
 * @param caseResult 解释结果：成功 true 或失败 false
 */
public void setCaseResult(boolean caseResult){
    .....
}
```



```
/**
 * 当某个测试项由于特殊原因不能立即测试时，可以在 DragonBoxMain 的
 * restartAllTest 函数中使用该接口延时测试
 * @param millisecond 延时时长，单位 ms
 */
public void startCaseDelay(int millisecond){
    .....
}
```

```
/**
 * 当两个测试项互斥不能同时测试时，可以在 DragonBoxMain 的 restartAllTest
 * 函数中使用该接口互斥两个测试项
 * @param afterCase 当前测试项会在 afterCase 测试项执行完毕后开始测试
 */
public void startCastAfterCase(IBaseCase afterCase){
    .....
}
```

```
/**
 * 已在 DragonBoxMain 类中实现该接口，可以监听测试结果。如：当测试结果
 * 全部通过时启动 DragonSN 进行烧号。
 * @param baseCase 测试项
 * @param caseResult 该测试项的测试结果
 */
public interface onResultChangeListener{
    public void onResultChange(IBaseCase baseCase,boolean caseResult);
}
```

## 5. Declaration

This document is the original work and copyrighted property of Allwinner Technology ( “Allwinner” ). Reproduction in whole or in part must obtain the written approval of Allwinner and give clear acknowledgement to the copyright owner. The information furnished by Allwinner is believed to be accurate and reliable. Allwinner reserves the right to make changes in circuit design and/or specifications at any time without notice. Allwinner does not assume any responsibility and liability for its use. Nor for any infringements of patents or other rights of the third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Allwinner. This datasheet neither states nor implies warranty of any kind, including fitness for any particular application.

Confidential