**CoreSS接口帮助文档**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **变更（+/-）说明** | **作者** | **版本号** | **日期** | **批准** |
| 6 | 1)新增CEC的管理类，CecControlManager | 刘竹星 | V1.0.5 | 2016-10-22 |  |
| 5 | 1)在 SystemUpgrade新增加可持续化存储接口getPersistenceDir() 需要注意的是，  STB升级后，该路径下存储的文件信息将会被清除。 | 刘竹星 | V1.0.4 | 2016-10-12 |  |
| 4 | 1）为SystemUpgrade的rebootsystem方法添加了备注信息。即不是升级的情况下，重启系统请调用systemShell的reboot | 刘竹星 | V1.0.3 | 2016-9-26 |  |
| 3 | 1)将setWiFiConfiguration（）移除，并新增  updateWiFiInfo()来更新Wi-Fi的IP地址等信息。  2）新增加WiFiDevInfo 这个容器类，负责装载Wi-Fi IP地址等信息。 | 刘竹星 | V1.0.2 | 2016-09-05 |  |
| 2 | 1. 新增WifiAdmin这个类专门负责Wi-Fi的业务逻辑。其中包含的接口有：starScan（）和setWifiConfiguration（）和   getWiFiInfo()。   1. 新增了isS912()和isS905()的平台判断 2. 由于S912 及S905X的U盘路径为动态路径，暂时增加了StorageMounte这个类去获取U盘和SD卡的路径。   其中包含的方法有：getValidUSBPath()  和getExtendSDPath()，只有S912 S905X平台有具体的实现上述方法。   1. 移除了原唐义峰在905CoreSS中的Ethernet添加的isSupportEthernet() 和isSupportBluetooth() 和getWifiInfo()和isAndroid5（）。 | 刘竹星 | V1.0.1 | 2016-8-22 |  |
| 1 | 初稿 | 刘竹星 | V1.0.0 | 2016-8-1 |  |

CoreSS接口帮助文档目录

[1. 文档阅读帮助说明： 1](#_Toc464727488)

[2. 音频管理（Audio） 2](#_Toc464727489)

[2.1. 获取系统支持的所有音频 2](#_Toc464727490)

[2.2. 获取系统当前音频格式 2](#_Toc464727491)

[2.3. 设置当前音频格式 2](#_Toc464727492)

[3. 公共接口（CommonInterface） 3](#_Toc464727493)

[3.1. 开启Wi-Fi扫描 @deprecated 3](#_Toc464727494)

[3.2. 设置系统语言类型 3](#_Toc464727495)

[3.3. 擦除SD卡 3](#_Toc464727496)

[4. 显示模式 （DisplayMode） 4](#_Toc464727497)

[4.1. 手动设置HDMI视频的输出模式 4](#_Toc464727498)

[4.2. HDMI设置为自动模式 4](#_Toc464727499)

[4.3. 判断当前HDMI是否为自动模式 4](#_Toc464727500)

[4.4. 获取当前HDMI的输出模式 4](#_Toc464727501)

[4.5. 获取当前输出模式在模式列表中的位置信息 4](#_Toc464727502)

[5. DM2016检测（DM2016Checker） 5](#_Toc464727503)

[5.1. 检测DM2016 \* 5](#_Toc464727504)

[6. 以太网 (Ethernet) 6](#_Toc464727505)

[6.1. 以太网使能（开关） 6](#_Toc464727506)

[6.2. 设置以太网 6](#_Toc464727507)

[6.3. 获取当前以太网信息 6](#_Toc464727508)

[6.4. 获取以太网连接状态 6](#_Toc464727509)

[6.5. 获取以太网的MAC地址 6](#_Toc464727510)

[6.6. 获取以太网的广播信息 6](#_Toc464727511)

[7. Wi-Fi管理（WiFiAdmin） 7](#_Toc464727512)

[7.1. 开启Wi-Fi扫描 7](#_Toc464727513)

[7.2. 更新Wi-Fi配置信息 7](#_Toc464727514)

[7.3. 获取当前Wi-Fi网络配置信息 7](#_Toc464727515)

[8. 事件注入（EventInjecter） 8](#_Toc464727516)

[8.1. 按键事件注入系统 8](#_Toc464727517)

[8.2. 触摸事件注入系统 8](#_Toc464727518)

[8.3. 重力事件注入系统 8](#_Toc464727519)

[8.4. 移动鼠标 8](#_Toc464727520)

[8.5. 鼠标左击事件 8](#_Toc464727521)

[9. 工厂烧录(FactoryBurn) 9](#_Toc464727522)

[9.1. 烧录网卡Mac地址 \* 9](#_Toc464727523)

[9.2. 读取网卡Mac地址 \* 9](#_Toc464727524)

[9.3. 烧录Wi-Fi模组的Mac地址 9](#_Toc464727525)

[9.4. 读取Wi-Fi模组的Mac地址 9](#_Toc464727526)

[9.5. 烧录设备SN序列号\* 9](#_Toc464727527)

[9.6. 读取设备的SN序列号 \* 9](#_Toc464727528)

[9.7. 检查HDCP秘钥文件是否有效\* 10](#_Toc464727529)

[9.8. 获取HDCP秘钥文件中已被使用的秘钥数目\* 10](#_Toc464727530)

[9.9. 获取HDCP秘钥文件中未被使用的秘钥数目\* 10](#_Toc464727531)

[9.10. 烧录HDCP秘钥文件\* 10](#_Toc464727532)

[9.11. 获取HDCP秘钥烧录状态\* 10](#_Toc464727533)

[9.12. 烧录看门狗安全秘钥 \* 11](#_Toc464727534)

[9.13. 读取当前看门狗安全秘钥 \* 11](#_Toc464727535)

[9.14. 烧录MacID \* 11](#_Toc464727536)

[9.15. 烧录AC到S2 \* 11](#_Toc464727537)

[9.16. 烧录SN到S2 11](#_Toc464727538)

[9.17. 烧录ChipID到S2 11](#_Toc464727539)

[9.18. 读取S2的AC 12](#_Toc464727540)

[9.19. 读取S2的SN 12](#_Toc464727541)

[9.20. 读取S2的ChipID 12](#_Toc464727542)

[9.21. 读取S2的MacID 12](#_Toc464727543)

[9.22. 读取设备ID号 12](#_Toc464727544)

[9.23. 烧录设备的ID号 12](#_Toc464727545)

[9.24. DDR测试 12](#_Toc464727546)

[9.25. 设置系统权限秘钥 13](#_Toc464727547)

[10. LED总线控制（LedGPIO） 13](#_Toc464727548)

[10.1. 开启LED灯 13](#_Toc464727549)

[10.2. 关闭LED灯 13](#_Toc464727550)

[11. 输出位置（OutputPosition） 14](#_Toc464727551)

[11.1. 获取输出画面大小 14](#_Toc464727552)

[11.2. 设置输出画面大小 14](#_Toc464727553)

[11.3. 保存输出画面设置 14](#_Toc464727554)

[11.4. 获取系统最小的输出画面 14](#_Toc464727555)

[12. 存储信息（StorageMountedInfo） 15](#_Toc464727556)

[12.1. 获取外部SD卡路径 15](#_Toc464727557)

[12.2. 获取内部挂载SD卡路径 15](#_Toc464727558)

[12.3. 获取内部可读写SD卡路径 15](#_Toc464727559)

[12.4. 获取可以用的USB存储路径 15](#_Toc464727560)

[12.5. 获取SATA的路径 16](#_Toc464727561)

[13. 额外存储信息（StorageMounte） 16](#_Toc464727562)

[13.1. 获取USB的路径 16](#_Toc464727563)

[13.2. 获取SD卡的路径 16](#_Toc464727564)

[14. 系统信息（SystemInformation） 17](#_Toc464727565)

[14.1. 获取系统Model 17](#_Toc464727566)

[14.2. 获取系统的版本信息 17](#_Toc464727567)

[14.3. 获取系统的PID 17](#_Toc464727568)

[14.4. 获取系统的VID 17](#_Toc464727569)

[14.5. 获取STB有效的AV Code 17](#_Toc464727570)

[14.6. 获取与STB连接的HDIM设备信息 18](#_Toc464727571)

[14.7. 获取STB的Flash信息 18](#_Toc464727572)

[14.8. 获取STB的ID号 18](#_Toc464727573)

[14.9. 获取STB的硬件版本 18](#_Toc464727574)

[14.10. 获取STB的描述信息 18](#_Toc464727575)

[15. 系统Shell命令（SystemShell） 19](#_Toc464727576)

[15.1. 退出Shell操作 19](#_Toc464727577)

[15.2. 执行Shell命令 19](#_Toc464727578)

[15.3. 重启系统 19](#_Toc464727579)

[15.4. 同步数据 19](#_Toc464727580)

[15.5. 在新进程中执行Shell命令 19](#_Toc464727581)

[15.6. 开辟新进程执行Shell命令并执行完成后销毁进程 20](#_Toc464727582)

[16. 系统升级（SystemUpgrade） 21](#_Toc464727583)

[16.1. 获取升级包的版本号 \* 21](#_Toc464727584)

[16.2. 升级包检测 \* 21](#_Toc464727585)

[16.3. 通过升级包进行系统升级\* 21](#_Toc464727586)

[16.4. 重启系统并开始升级系统 21](#_Toc464727587)

[16.5. 重启系统并清除所有用户数据 22](#_Toc464727588)

[16.6. 清空用户资料 22](#_Toc464727589)

[16.7. 得到OTA升级的根路径 22](#_Toc464727590)

[16.8. OTA文件进行升级系统 22](#_Toc464727591)

[16.9. 恢复出厂设置 22](#_Toc464727592)

[17. TV输入资源（TVInputSource） 23](#_Toc464727593)

[17.1. 设置输入资源类型 23](#_Toc464727594)

[17.2. 获取当前输入资源类型 23](#_Toc464727595)

[17.3. 设置TV显示的画布 23](#_Toc464727596)

[18. 平台检测工具（BrandUtil） 24](#_Toc464727597)

[18.1. MX平台检测 24](#_Toc464727598)

[18.2. M3平台检测 24](#_Toc464727599)

[18.3. A20平台检测 24](#_Toc464727600)

[18.4. Mstar平台检测 24](#_Toc464727601)

[18.5. M8平台检测 24](#_Toc464727602)

[18.6. S905平台检测 24](#_Toc464727603)

[18.7. S912平台检测 24](#_Toc464727604)

[19. CEC功能（CecControlManager） 25](#_Toc464727605)

[19.1. 开启关闭CEC 25](#_Toc464727606)

[19.2. 开启关闭一键唤醒 25](#_Toc464727607)

[19.3. 开启设备语言自适应 25](#_Toc464727608)

[19.4. 判断CEC开启状态 25](#_Toc464727609)

[19.5. 判断一键唤醒开启状态 25](#_Toc464727610)

[19.6. 判断设备语言自适应状态开启 25](#_Toc464727611)

# 文档阅读帮助说明：

一级标题中（）里面的内容为接口类的类名。

例如：1.音频管理 （Audio） 其中Audio 为该接口类的类名。

二级标题为对应的一级标题中接口类提供的公共方法，具体方法名字可以点击目录相应的链接进行查看。其中 带\*号的方法需要用到盒子端的so库文件，库文件的相关信息在方法中有说明。

这份文档适用于所有CoreSS版本。华为项目与其他项目相比，华为项目只是新增了部分接口。

# 音频管理（Audio）

## 获取系统支持的所有音频

调用方法 : getAudios ()

返回值类型: Arraylist<AudioFormat> ，其中AudioFormat 包含了Audio的名字和ID号。

返回值：以Arraylist的格式返回当前系统支持的所有音频类型包括：PCM、HDMI\_RAW、SPDIF\_RAW。

## 获取系统当前音频格式

调用方法: getCurrentAudio ()

返回值类型：AudioFormat。其中AudioFormat 包含了Audio的名字和ID号。

返回值：返回当前系统的音频格式

## 设置当前音频格式

调用方法: setAudios (AudioFormat audio)

传入参数：audio表示音频的格式

# 公共接口（CommonInterface）

## 开启Wi-Fi扫描 @deprecated

调用方法: startScan()

返回值类型： boolean

返回值：如果开启扫描模式成功则返回true，否则则返回false

## 设置系统语言类型

调用方法：setLanguage(Locale locale)

传入参数：locale表示本地的语言

## 擦除SD卡

调用方法：eraseSdCard()

返回值类型：void

# 显示模式 （DisplayMode）

## 手动设置HDMI视频的输出模式

调用方法：setOutputModeFormat(DisplayModeFormat outputModeFormat)

参数类型：DisplayModeFormat（视频输出模式格式，例如1080P 720P）

## HDMI设置为自动模式

调用方法：setOutputModeAuto()

## 判断当前HDMI是否为自动模式

调用方法：getOutputModeAuto()

返回值类型： boolean

返回值：true表示HDMI处于自动模式。False表示HDMI处于手动设置模式。

## 获取当前HDMI的输出模式

调用方法: getCurrentOutputMode()

返回值类型：DisplayModeFormat

返回值：视频输出模式，包括模式名字和模式类型。

## 获取当前输出模式在模式列表中的位置信息

调用方法：getCurrentOutputModePst()

返回值类型：int

返回值：该视频输出模式在列表中对应的位置

# DM2016检测（DM2016Checker）

## 检测DM2016 \*

调用方法：checkDM2016()

返回值类型: boolean

返回值：true表示是本公司硬件。False表示不是。

相关联的so库文件：chkdm2016\_jni.so (位于盒子/system/lib目录下)

备注：检测DM2016实际上是去判断盒子的硬件是否为我们公司生产的。

# 以太网 (Ethernet)

## 以太网使能（开关）

调用方法：setNetEnabled(boolean enable)

参数类型： boolean

参数：当设置为true 的时候表示开启以太网。设置为false表示关闭以太网。

## 设置以太网

调用方法：updateEthInfo(EthernetInfo ethInfo)

参数类型：以太网基本信息，包括网关，IP地址，DNS，网络模式（DHCP，静态）等

参数：要设置的以太网的信息。

备注：这个方法可以设置，以太网获取IP的模式，DHCP及静态获取。

在静态获取时，可以设置IP地址，网关，DNS等信息

## 获取当前以太网信息

调用方法：getEthernetInfo()

返回值类型：以太网基本信息，包括网关，IP地址，DNS，网络模式（DHCP，静态）等等

返回值：能够得到网络模式（DHCP，静态）网关，IP地址，DNS等信息

## 获取以太网连接状态

调用方法：getEthernetStatus()

返回值类型：int

返回值：ETHER\_STATUS\_DISABLED（value=0） 连接失败 ，ETHER\_STATUS\_ENABLED （value=1）连接成功，ETHER\_STATUS\_UNKNOWN（value=2） 连接状态

不明。

## 获取以太网的MAC地址

调用方法：getEthMac()

返回值类型：String

返回值：以字符串的形式得到盒子以太网的Mac地址

## 获取以太网的广播信息

调用方法：getEthBroadcastAction(String action)

返回值类型：String

返回值：以字符串的形式得到盒子以太网的发出广播的信息

# Wi-Fi管理（WiFiAdmin）

## 开启Wi-Fi扫描

调用方法：startScan ()

返回值类型：boolean

返回值：true代表开启扫描成功

## 更新Wi-Fi配置信息

调用方法：updateWiFiInfo(WiFiDevInfo wifiDevInfo)

参数说明：WiFiDevInfo 包含了mode ，ip，gataway，netmask，dns。

## 获取当前Wi-Fi网络配置信息

调用方法：getWiFiInfo()

返回值类型：WiFiDevInfo

返回值：返回当前Wi-Fi的信息，包括mode，ip，gateway，netmask，dns。

# 事件注入（EventInjecter）

## 按键事件注入系统

调用方法：injectKeyEvent(KeyEvent keyEvent)

参数：要注入的按键事件（KeyEvent）

## 触摸事件注入系统

调用方法：injectMotionEvent(MotionEvent event)

参数：要注入的触摸事件（MotionEvent）

## 重力事件注入系统

调用方法：injectGravity(byte[] gravity)

参数：要注入的重力事件（MotionEvent）

## 移动鼠标

调用方法：moveMouse(int x ,int y)

参数：x代表移动的横坐标，y代表移动的纵坐标。

## 鼠标左击事件

调用方法：mouseLeftClick(int action)

参数：action表示注入的事件，包括：KeyEvent.ACTION\_DOWN, KeyEvent.ACTION\_UP, KeyEvent.ACTION\_MULTIPLE

# 工厂烧录(FactoryBurn)

## 烧录网卡Mac地址 \*

调用方法：writeEthMAC(String ethMac)

参数：用字符串的形式传入Mac地址

相关联so库文件：idburnerjni.so

可能抛出的异常：BurnPermissionException

## 读取网卡Mac地址 \*

调用方法：readEthMAC()

相关联so库文件：idburnerjni.so

返回值：以字符串形式得到网卡的Mac地址

## 烧录Wi-Fi模组的Mac地址

调用方法：writeWiFiMAC(String wifiMAC)

参数：以字符串形式传入Wi-Fi模组的Mac地址

可能抛出的异常：BurnPermissionException

## 读取Wi-Fi模组的Mac地址

调用方法：readWiFiMAC(String wifiModelType)

返回值类型：字符串（String）

返回值：以字符串形式得到Wi-Fi的Mac地址

## 烧录设备SN序列号\*

调用方法：writeDeviceSN(String sn, int len) 或writeDeviceSN(String sn)

参数：以字符串形式传入SN序列号，SN长度为可选的传入参数。

相关联so库文件：idburnerjni.so

可能抛出的异常：BurnPermissionException

## 读取设备的SN序列号 \*

调用方法：readDeviceSN(int snLen)或readDeviceSN()

返回值类型：字符串（String）

返回值：用字符串表示返回的SN序列号

相关联so库文件：idburnerjni.so

备注：若该方法需要传入SN的长度，其长度最大值为24。

## 检查HDCP秘钥文件是否有效\*

调用方法：checkHDCPFileValid(String keyFilePath)

返回值类型：布尔型（boolean）

参数:keyFilePath 表示秘钥所在的路径

返回值：true表示有效，fasle表示无效。

相关联so库文件：idburnerjni.so

备注：HDCP的意思是高带宽数字内容保护技术

## 获取HDCP秘钥文件中已被使用的秘钥数目\*

调用方法：getHDCPKeyUsedNum(String keyFilePath)

返回值类型：整型（int）

参数:keyFilePath 表示秘钥所在的路径

返回值：已被使用的秘钥数目。

相关联so库文件：idburnerjni.so

备注：HDCP的意思是高带宽数字内容保护技术

## 获取HDCP秘钥文件中未被使用的秘钥数目\*

调用方法：getHDCPKeyUnusedNum(String keyFilePath)

返回值类型：整型（int）

参数:keyFilePath 表示秘钥所在的路径

返回值：未被使用的秘钥数目。

相关联so库文件：idburnerjni.so

备注：HDCP的意思是高带宽数字内容保护技术

## 烧录HDCP秘钥文件\*

调用方法：writeHDCPKey(String keyFilePath)

方法作用：将HDCP秘钥文件烧录进入EPROM中

参数：keyFilePath 表示秘钥所在的路径。

相关联so库文件：idburnerjni.so

可能抛出的异常：BurnPermissionException

备注：HDCP的意思是高带宽数字内容保护技术

## 获取HDCP秘钥烧录状态\*

调用方法：isHDCPBurned()

返回值类型：布尔型（boolean）

返回值：true表示HDCP秘钥已经烧录，false未烧录

相关联so库文件：idburnerjni.so

备注：HDCP的意思是高带宽数字内容保护技术

## 烧录看门狗安全秘钥 \*

调用方法：writeDongleSecureKey(String secureKey)

参数：secureKey表示要烧录的安全秘钥。

相关联so库文件：istagedongle.so 和istage\_dongle\_jni.so

可能抛出的异常：BurnPermissionException

## 读取当前看门狗安全秘钥 \*

调用方法：readDongleSecureKey(int len)

返回值类型：字符串（String）

返回值：以字符串形式表示该看门狗安全秘钥

相关联so库文件：istagedongle.so 和istage\_dongle\_jni.so

备注：传入参数len表示秘钥的长度

## 烧录MacID \*

调用方法：writeDongleSecureKey(String secureKey)

参数：MacID表示要烧录的ID号一般默认为32位。

相关联so库文件：idburnerjni.so

可能抛出的异常：BurnPermissionException

## 烧录AC到S2 \*

调用方法：writeS2AC(final String acStr)

参数：acStr表示要烧录的AC。

相关联so库文件：idburnerjni.so

可能抛出的异常：BurnPermissionException

## 烧录SN到S2

调用方法：writeS2SN(final String snStr)

方法作用：将SN烧录进S2

参数：snStr表示要烧录的SN。

可能抛出的异常：BurnPermissionException

## 烧录ChipID到S2

调用方法：writeS2ChipID(final String cid)

方法作用：将ChipID烧录进S2

参数：cid表示要烧录的ChipID。

可能抛出的异常：BurnPermissionException

## 读取S2的AC

调用方法：readS2AC()

返回值类型： String

返回值：以字符串形式表示设备的AC信息

## 读取S2的SN

调用方法：readS2SN ()

返回值类型： String

返回值：以字符串形式表示设备的SN

## 读取S2的ChipID

调用方法：readS2ChipID ()

返回值类型： String

返回值：以字符串形式表示设备的ChipID

## 读取S2的MacID

调用方法：readMACID ()

返回值类型： String

返回值：以字符串形式表示设备的MacID

## 读取设备ID号

调用方法：readDeviceID ()

返回值类型： String

返回值：返回设备的ID号

备注：设备ID号即IMEI号

## 烧录设备的ID号

调用方法：writeDeviceID(String deviceID)

参数：deviceID表示要烧录的设备ID。

可能抛出的异常：BurnPermissionException

## DDR测试

调用方法：DDRTest(int testCount)

返回值类型： int

返回值：REC\_BURNER\_SUCCESS（value：0） 或者ERROR\_BURNER\_FAILE（value ： -1）

备注：DDR 意思是双倍速率同步动态随机存储器

## 设置系统权限秘钥

调用方法：setPermissionKey(String key)

参数：key表示系统的权限秘钥

备注：烧录任何的内容进入EPROM前，都需要调用该方法。

# LED总线控制（LedGPIO）

## 开启LED灯

调用方法：startLedFlash()

## 关闭LED灯

调用方法：stopLedFlash(boolean exit)

参数：exit若为true表示确认关闭。

# 输出位置（OutputPosition）

## 获取输出画面大小

调用方法：getArePercent()

返回值类型： int

返回值：返回输出画面占的百分数。

## 设置输出画面大小

调用方法：setArePercent()

## 保存输出画面设置

调用方法：saveArePercent()

返回值类型： boolean

返回值：true表示保存成功

## 获取系统最小的输出画面

调用方法：getMinPercent()

返回值类型： int

返回值：返回画面最小输出比例的百分数

# 存储信息（StorageMountedInfo）

## 获取外部SD卡路径

调用方法：getExtendSDPath()

返回值类型： String

返回值：返回外部SD卡路径

备注：该方法为静态方法可以直接通过类名访问，若不存在外部SD卡，则返回null

## 获取内部挂载SD卡路径

调用方法：getInnerSDMountedPath()

返回值类型： String

返回值：返回内部挂载SD卡路径

备注：该方法为静态方法可以直接通过类名访问，若不存在内部挂载SD卡，则返回null

## 获取内部可读写SD卡路径

调用方法：getInnerSDRWPath()

返回值类型： String

返回值：返回内部可读写SD卡路径

备注：该方法为静态方法可以直接通过类名访问，若不存在内部可读写SD卡，则返回null

## 获取可以用的USB存储路径

调用方法：getValidUSBPath()

返回值类型：ArryList< String>

返回值：返回可用USB存储路径的Arrylist链表

备注：该方法为静态方法可以直接通过类名访问，若不存在USB存储，则返回null

## 获取SATA的路径

调用方法：getSataPath()

返回值类型： String

返回值：返回SATA路径

备注：该方法为静态方法可以直接通过类名访问。

# 额外存储信息（StorageMounte）

## 获取USB的路径

调用方法：getValidUSBPath()

返回值类型： ArrayList<String>

返回值：返回外部USB的路径的数组（U盘，盒子支持同时挂载两个U盘）

备注：因为S912的路径为动态路径，所以新增加了类StorageMounte。原StorageMountedInfo 中的方法仍然可用。但是需要注意S912平台不可以用StorageMountedInfo中的方法，因为S912未在StorageMountedInfo中做出实现会报空指针异常。

## 获取SD卡的路径

调用方法：getExtendSDPath()

返回值类型： String

返回值：返回外部SD卡的路径

备注：因为S912的路径为动态路径，所以新增加了类StorageMounte。原StorageMountedInfo 中的方法仍然可用。但是需要注意S912平台不可以用StorageMountedInfo中的方法，因为S912未在StorageMountedInfo中做出实现会报空指针异常。

# 系统信息（SystemInformation）

## 获取系统Model

调用方法：getModel()

返回值类型： String

返回值：返回系统的Model

## 获取系统的版本信息

调用方法：getVersion()

返回值类型： String

返回值：返回系统的版本号

## 获取系统的PID

调用方法：getPID()

返回值类型：String

返回值：返回PID

## 获取系统的VID

调用方法：getVID()

返回值类型：String

返回值：返回VID

## 获取STB有效的AV Code

调用方法：getAvailableAVCodec()

返回值类型：String

返回值：返回AV Code

## 获取与STB连接的HDIM设备信息

调用方法：getHDMIDevices()

返回值类型：String

返回值：返回HDMI设备信息

## 获取STB的Flash信息

调用方法：getFlashInfo()

返回值类型：String

返回值：返回Flash信息

## 获取STB的ID号

调用方法：getSTBID()

返回值类型：String

返回值：返回STB的ID号

## 获取STB的硬件版本

调用方法：getHardwareVersion ()

返回值类型：String

返回值：返回STB的硬件版本

## 获取STB的描述信息

调用方法：getDescription()

返回值类型：String

返回值：返回STB描述信息

# 系统Shell命令（SystemShell）

## 退出Shell操作

调用方法：release()

可能抛出异常：IOException

## 执行Shell命令

调用方法：exec(String cmd)

传入参数：cmd表示要执行的Shell命令

返回值类型： boolean

返回值：true表示执行成功

可能抛出异常：SyncFailedException，IOException

## 重启系统

调用方法：reboot()

可能抛出异常：SyncFailedException，IOException

## 同步数据

调用方法：sync()

## 在新进程中执行Shell命令

调用方法：execNewProcess(String cmd)

传入参数：cmd表示要执行的Shell命令

可能抛出异常：SyncFailedException，IOException

备注：该方法为静态方法

## 开辟新进程执行Shell命令并执行完成后销毁进程

调用方法：execNewProcessWithResult(String cmd)

传入参数：cmd表示要执行的Shell命令

可能抛出异常：SyncFailedException，IOException

备注：该方法为静态方法

# 系统升级（SystemUpgrade）

## 获取升级包的版本号 \*

调用方法：getPackageVersion(String path)

传入参数：path表示升级包的路径

相关联的so库：sysup\_jni.so

返回值类型： String

返回值：返回版本号

## 升级包检测 \*

调用方法：checkFireware(String path)

传入参数：path表示升级包的路径

返回值类型： int

返回值：CHKFW\_SUCCESS（0），CHKFW\_FILE\_NOTEXIST（1）

CHKFW\_FILE\_INVALID（2），CHKFW\_OLDER\_THAN\_CURRENT（3）

CHKFW\_SAME\_WITH\_CURRENT（4），CHKFW\_NOT\_MATCHTYP（5）

CHKFW\_INC\_BASE\_VER\_NOT\_MATCH（6）

CHKFW\_INC\_VERSION\_NOT\_NEWER（7），CHKFW\_INC\_FS\_NOT\_MATCH（8）

CHKFW\_OTHERS\_ERR（9）

相关联的so库：sysup\_jni.so

## 通过升级包进行系统升级\*

调用方法：update(String path)

传入参数：path表示升级包的路径

返回值类型： boolean

返回值：true表示升级成功，false表示升级失败。

相关联的so库：sysup\_jni.so

## 重启系统并开始升级系统

调用方法：rebootSystem(String path)

传入参数：path表示升级包的路径

返回值类型：boolean

返回值：true表示升级成功，false表示升级失败。

备注：如果不进行升级的操作，只是单纯的重启系统，请不要调用该接口。

重启系统应该调用ShellSystem的reboot（）的接口；

## 重启系统并清除所有用户数据

调用方法：rebootWipeUserData()

返回值类型： boolean

返回值：true表示清除成功，false表示清除失败。

## 清空用户资料

调用方法：wipeUserData()

返回值类型： boolean

返回值：true表示清除成功，false表示清除失败。

## 得到OTA升级的根路径

调用方法：getOTAdir()

返回值类型： String

返回值：返回升级的根路径

## OTA文件进行升级系统

调用方法：upgrade(String filePath)

传入参数：filePath表示文件路径

## 恢复出厂设置

调用方法：masterClear()

# TV输入资源（TVInputSource）

## 设置输入资源类型

调用方法：setInputSource(int sourceType)

传入参数：sourceType表示资源类型,其中有E\_INPUT\_SOURCE\_CVBS（0）

E\_INPUT\_SOURCE\_STORAGE（1），E\_INPUT\_SOURCE\_HDMI（2）

## 获取当前输入资源类型

调用方法：getInputSourceType()

返回值类型： int

返回值：E\_INPUT\_SOURCE\_CVBS（0）E\_INPUT\_SOURCE\_STORAGE（1），E\_INPUT\_SOURCE\_HDMI（2）

## 设置TV显示的画布

调用方法：setTVPlayerSurface(SurfaceHolder surface)

传入参数：surface表示TV显示的画布

# 平台检测工具（BrandUtil）

## MX平台检测

调用方法：isMX()

返回值类型： boolean

返回值：true表示MX平台。

## M3平台检测

调用方法：isM3()

返回值类型： boolean

返回值：true表示M3平台。

## A20平台检测

调用方法：isA20()

返回值类型： boolean

返回值：true表示A20平台。

## Mstar平台检测

调用方法：isMstar()

返回值类型： boolean

返回值：true表示Mstar平台。

## M8平台检测

调用方法：isM8()

返回值类型： boolean

返回值：true表示M8平台。

## S905平台检测

调用方法：isM8()

返回值类型： boolean

返回值：true表示S905平台。

## S912平台检测

调用方法：isS912()

返回值类型： boolean

返回值：true表示S912平台。

# CEC功能（CecControlManager）

## 开启关闭CEC

调用方法：setCECEnable(boolean isEnable)

返回值类型：void

Note : Amlogic M8 平台以后 支持CEC功能， CEC（ Consumer Electronics Control） 是控制各种音视产品的协议；其功能是用单个遥控器控制连接 TV 上的所有设备。

## 开启关闭一键唤醒

调用方法：setOneTouchPlayEnable (boolean isEnable)

返回值类型：void

Note : 自动将处于休眠的电视唤醒、自动切换电视的输入源

## 开启设备语言自适应

调用方法：setAutoSwitchLanguageEnable (boolean isEnable)

返回值类型：void

Note : 电视 OSD 语言可以与设备的语言保持一致

## 判断CEC开启状态

调用方法：isOpenCEC（）

返回值类型：boolean

返回值：true开启

## 判断一键唤醒开启状态

调用方法：isOneTouchPlay（）

返回值类型：boolean

返回值：true开启

## 判断设备语言自适应状态开启

调用方法：isAutoSwitchLanguage（）

返回值类型：boolean

返回值：true开启