

会议平板统一 SDK

2024. 5. 14

目录

一、模块划分	3
二、接口类型划分	4
三、基础接口定义	4

一、模块划分

一级模块	二级模块
设备固件信息 (说明：设备物理硬件或整机系统程序相关参数)	硬件参数 (说明：设备物理硬件参数设置/获取)
	软件参数 (说明：设备整机系统程序相关参数设置/获取)
网络 (说明：设备网络连接相关配置，包括有线、无线、热点、WiFi P2P 等)	有线网络 (说明：有线网络配置相关)
	无线网络 (说明：无线网络配置相关)
	网络策略 (说明：网络策略配置相关)
	SADP 管理 (说明：SADP 相关业务接口)
显示 (说明：设备屏幕显示相关)	显示参数 (说明：设备屏幕显示参数相关)
	多屏异显 (说明：多屏异显的功能支持)
性能 (说明：设备性能优化配置、运行时性能参数获取)	运行时参数 (说明：设备运行时性能参数获取)
安全 (说明：敏感功能权限配置、隐私相关配置、系统安全相关配置)	USB 权限 (说明：USB 相关权限配置)
应用设置 (说明：针对具体应用的配置模块，如：为应用增加系统保护、应用权限管理、应用多开等)	系统保护 (说明：针对特定应用进行系统保护的配置)
电源 (说明：设备电源开关配置)	定时开关机 (说明：定时开关机配置)
	开机参数 (说明：开机过程相关配置)
更新 (说明：程序更新升级模块，包括组件升级和系统升级)	组件升级 (说明：组件升级相关接口)
	系统升级 (说明：整机系统程序升级相关接口)
测试 (说明：开发者调试相关功能配置)	ADB 设置 (说明：ADB 状态获取与设置)
相机 (说明：设备相机相关拓展功能配置)	

音频 (说明：设备相机相关拓展功能配置)

二、接口类型划分

类别	定义说明	允许开放	强制实现	评议方式	管理平台	迭代频率
基础接口	海康设备必须实现的最小接口集，一般为高频调用、核心组件依赖的接口	是	是	评委会评审	SDK 管理平台	低
扩展接口	设备按需实现，一般为业务类接口、智能类接口	是	否	评委会评审	SDK 管理平台	中
私有接口	部门内部的私有接口，一般为强业务属性接口、临时定制接口等	否	否	部门内部评审	SDK 管理平台	高

三、基础接口定义

一级模块	二级模块	接口
设备固件信息 deviceinfo	硬件参数 HardwareInfoHolder	描述： 获取当前设备温度 接口： double getDeviceTemperature(int modeld)

		<p>说明： @description 获取设备模块的温度 @param mode 模块号 @return 温度</p>
	<p>软件参数 SoftwareInfoHolder</p>	<p>描述： SDK 版本号</p> <p>接口： int getSDKVersionCode()</p> <p>说明： @description 获取 SDK 版本号 @return SDK 版本号</p>
		<p>描述： SDK 版本名称（缺省则为“V 版本号”）</p> <p>接口： String getSDKVersionName()</p> <p>说明： @description 获取 SDK 版本名称 @return SDK 版本名称（缺省则为“V 版本号”）</p>
		<p>描述： SDK 版本说明（可缺省）</p> <p>接口： String getSDKVersionDescription()</p> <p>说明： @description 获取 SDK 版本说明 @return SDK 版本说明</p>

一级模块	二级模块	接口
网络 network	有线网络 EthernetManager	<p>描述： 打开有线网络</p> <p>接口： void openEthernet()</p> <p>说明： @description 打开有线网络开关</p>

		<p>描述： 关闭有线网络</p> <p>接口： void closeEthernet()</p> <p>说明： @description 关闭有线网络开关</p>
		<p>描述： 获取有线网络状态</p> <p>接口： boolean getEthernetState()</p> <p>说明： @description 更新系统有线网络打开状态 @return 有线网络状态</p>
	<p>网络策略 NetworkPolicy</p>	<p>描述： 是否支持 IPV6</p> <p>接口： boolean isIPv6Supported()</p> <p>说明： @description 是否支持 IPv6 @return IPv6 的支持结果</p>
	<p>SADP 管理 SADP</p>	<p>描述： 开启 SADP</p> <p>接口： void start()</p> <p>说明： @description 开启 SADP</p> <p>描述： 关闭 SADP</p> <p>接口： void stop()</p> <p>说明： @description 停止 SADP</p> <p>描述： 获取 Sadp 功能是否启用</p>

		<p>接口： boolean isSADPEnabled()</p> <p>说明： @description 是否启用 SADP @return SADP 启用状态</p>
		<p>描述： 启用 Sadp 功能</p> <p>接口： void enable()</p> <p>说明： @description 启用 SADP 功能</p>
		<p>描述： 禁用 Sadp 功能</p> <p>接口： void disable()</p> <p>说明： @description 禁用 SADP 功能</p>
		<p>描述： 激活密码校验</p> <p>接口： boolean checkPassword(String password)</p> <p>说明： @description 校验 SADP 激活密码 @param password 输入 SADP 激活密码 @return SADP 激活结果</p>
		<p>描述： 获取 SADP 密码</p> <p>接口： String getPassword()</p> <p>说明： @description 获取 SADP 激活密码 @return SADP 的激活密码</p>
		<p>描述： 修改 SADP 密码</p>

		<p>接口： void setPassword(String password)</p> <p>说明： @description 设置 SADP 激活密码 @param password 输入 SADP 激活密码</p>
		<p>描述： 获取设备是否激活</p> <p>接口： boolean isActive()</p> <p>说明： @description 设备是否激活 SADP @return 设备 SADP 的激活状态</p>
		<p>描述： 恢复未激活</p> <p>接口： void resetActiveState()</p> <p>说明： @description 重置 SADP 的激活状态</p>
		<p>描述： 是否在线</p> <p>接口： boolean isOnline()</p> <p>说明： @description 设备是否在线 @return 在线状态</p>
		<p>描述： 获取 SADP 版本</p> <p>接口： String getVersion()</p> <p>说明： @description 获取 SADP 版本 @return SADP 版本</p>

一级模块	二级模块	接口
显示 display	显示参数 DisplayManager	描述： 获取系统亮度 接口： int getSystemBrightness() 说明： @description 获取系统的屏幕亮度 @return 屏幕的亮度值 (0-100.0F)
		描述： 设置系统亮度 接口： void setSystemBrightness(int value) 说明： @description 设置系统屏幕的的亮度值 @param value 屏幕的亮度值 (0-100.0F)
		描述： 获取屏幕旋转角度 接口： float getScreenRotationAngle() 说明： @description 获取屏幕旋转角度 @return 屏幕旋转角度(0-360.0F)
		描述： 设置屏幕旋转角度 接口： int setScreenRotationAngle(float value) 说明： @description 设置屏幕旋转角度 @param value 屏幕旋转角度(0-360.0F) @return 返回-1 表示不支持，返回 0 表示设置成功
		描述： 获取屏幕 DPI

		<p>接口： int getScreenDPI()</p> <p>说明： @description 获取屏幕的 DPI @return 屏幕的 DPI</p>
		<p>描述： 设置屏幕 DPI</p> <p>接口： void setScreenDPI(int value)</p> <p>说明： @description 设置屏幕的 DPI @param value 屏幕的 DPI</p>
		<p>描述： 获取屏幕分辨率</p> <p>接口： Point getScreenResolution()</p> <p>说明： @description 获取屏幕的分配率 @param 屏幕的分辨率(x 为横向, y 为纵向)</p>
		<p>描述： 设置屏幕分辨率</p> <p>接口： void setScreenResolution(Point value)</p> <p>说明： @description 设置屏幕的分配率 @param value 屏幕的分辨率(x 为横向, y 为纵向)</p>

一级模块	二级模块	接口
安全 safety	USB 管理 USBManager	<p>描述： 设置 USB 使能</p> <p>接口： boolean isUSBEnabled()</p> <p>说明：</p>

		<p>@description 是否支持 USB 功能 @return 支持 USB 功能</p>
		<p>描述： 获取 USB 模式</p> <p>接口： int getUSBMode()</p> <p>说明： @description 获取当前的 USB 模式 @return USB 的模式</p>
		<p>描述： 设置 USB 模式</p> <p>接口： void setUSBMode(int mode)</p> <p>说明： @description 设置当前的 USB 模式 @param mode USB 的模式</p>
		<p>描述： 获取是否启用 U 盘挂载</p> <p>接口： boolean isMountUDiskEnabled()</p> <p>说明： @description 是否支持 U 盘挂载 @return 支持 U 盘挂载</p>
		<p>描述： 启用 U 盘挂载</p> <p>接口： void enableMountUDisk()</p> <p>说明： @description 启用 U 盘挂载</p>
		<p>描述： 禁用 U 盘挂载</p> <p>接口： void disableMountUDisk()</p> <p>说明：</p>

		@description 禁用 U 盘挂载
--	--	-----------------------

一级模块	二级模块	接口
应用设置 app	系统保护 PrivilegeManager	描述： 判断白名单功能是否开启 接口： boolean isWhiteListFunctionSupported(int functionId) 说明： @description 白名单功能是否支持 @param functionId 白名单的功能号 @return 功能开启状态
		描述： 开启白名单功能 接口： void openWhiteListFunction(int functionId) 说明： @description 打开白名单功能 @param functionId 白名单的功能号
		描述： 关闭白名单功能 接口： void closeWhiteListFunction(int functionId) 说明： @description 关闭白名单功能 @param functionId 白名单的功能号
		描述： 获取白名单支持列表 接口： List<Integer> getWhiteListFunctionList() 说明： @description 获取定义白名单功能号列表 @return 白名单功能列表
		描述： 应用添加到白名单

		<p>接口： void addWhiteList(int functionId, String packageName)</p> <p>说明： @description 添加到白名单 @param functionId 白名单的功能号 @param packageName 应用的包名</p> <hr/> <p>描述： 应用从白名单移除</p> <p>接口： void removeWhiteList(int functionId, String packageName)</p> <p>说明： @description 从白名单移除 @param functionId 白名单的功能号 @param packageName 应用的包名</p> <hr/> <p>描述： 获取白名单</p> <p>接口： List<String> getWhiteList(int functionId)</p> <p>说明： @description 获取白名单列表 @param functionId 白名单的功能号 @return 白名单列表</p> <hr/> <p>描述： 是否在白名单内</p> <p>接口： boolean isInWhiteList(int functionId, String packageName)</p> <p>说明： @description 判断是否在白名单列表中 @param functionId 白名单的功能号 @param packageName 应用包名 @return 是否在白名单中</p> <hr/> <p>描述： 注册开机启动组件</p> <p>接口： void registerStartupComponent(ComponentName name, Intent intent)</p>
--	--	--

		<p>说明： @description 注册开机启动组件 @param name 启动的组件名 @param intent 启动参数，如启动模式</p>
		<p>描述： 取消注册开机启动组件</p> <p>接口： void unregisterStartupComponent(ComponentName name)</p> <p>说明： @description 取消注册开机启动组件 @param name 启动的组件名</p>
		<p>描述： 获取开机启动组件列表</p> <p>接口： List<ComponentName> getStartupList()</p> <p>说明： @description 获取开机启动组件列表 @return 开启启动组件列表</p>

一级模块	二级模块	接口
电源 power	开机参数 BootManager	<p>描述： 替换开机 Logo</p> <p>接口： void replaceBootLogo(String path)</p> <p>说明： @description 替换开机 Logo @param path 开机 Logo 的路径</p>

一级模块	二级模块	接口
更新 update	组件升级 ComponentManager	<p>描述： 本地升级</p> <p>接口： void updateComponent(String path)</p> <p>说明：</p>

		<p>@description 更新组件, 本地升级</p> <p>@param path 升级文件的路径</p> <p>描述： 获取组件当前版本</p> <p>接口： int GetComponentVersion(String name)</p> <p>说明： @param name 组件名称</p>
		<p>描述： 获取组件列表</p> <p>接口： List<String> GetComponentList()</p> <p>说明： @description 获取组件列表 @return 组件名称列表</p>
	系统版本 SystemVersion	<p>描述： 获取系统版本</p> <p>接口： int getSystemVersion()</p> <p>说明： @description 获取系统版本 @return 系统的版本</p>
		<p>描述： 系统升级(本地)</p> <p>接口： void updateSystem(String path)</p> <p>说明： @description 系统升级 @param path 系统包本地路径</p>

一级模块	二级模块	接口
测试 debug	ADB 设置 ADBManager	<p>描述： 开启网络 ADB</p> <p>接口：</p>

		<p>void enableInternetADB()</p> <p>说明： @description 开启网络 ADB</p> <hr/> <p>描述： 禁用网络 ADB</p> <p>接口： void disableInternetADB()</p> <p>说明： @description 禁用网络 ADB</p> <hr/> <p>描述： 判断网络 ADB 是否开启</p> <p>接口： boolean isInternetADBSupported()</p> <p>说明： @return 网络 ADB 开启状态</p> <hr/> <p>描述： 开启 ADB</p> <p>接口： void enableADB()</p> <p>说明： @description 开启 ADB</p> <hr/> <p>描述： 禁用 ADB</p> <p>接口： void disableADB()</p> <p>说明： @description 禁用 ADB</p> <hr/> <p>描述： 判断 ADB 是否开启</p> <p>接口： boolean isADBSupported()</p> <p>说明： @description 是否支持 ADB</p>
--	--	---

	@return ADB 开启状态
--	------------------