



型号：XH-3568D2

API开发文档

广州芯伙智能科技有限公司

www.aixinhuo.com

目录

第一章 Android Studio 中使用 API.....	1
第二章 获取系统信息.....	3
2.1 获取API 版本-日期信息.....	3
2.2 获取设备的型号.....	3
2.3 获取设备的系统版本.....	4
2.4 获取设备内存容量.....	5
2.5 获取设备存储容量.....	6
2.6 获取设备固件版本.....	7
2.7 获取设备固件内核版本.....	8
2.8 获取设备固件版本和编译日期.....	8
2.9 获取固件编译时间.....	9
2.10 获取设备CPU 型号.....	10
2.11 获取厂商识别码.....	11
2.12 获取SN 号.....	11
第三章 关机和重启.....	13
3.1 关机.....	13
3.2 重启.....	13
第四章 显示.....	15

4.1 截屏保存	15
4.2 设置屏幕旋转角度	16
4.3 获取屏幕宽像素	17
4.4 获取屏幕高像素	17
4.5 导航栏状态设置	18
4.6 导航栏状态查询	19
4.7 设置滑出导航栏打开或关闭	20
4.8 查询滑出导航栏是否打开	21
4.9 设置滑出通知栏打开或关闭	22
4.10 查询滑出通知栏状态	23
4.11 设置屏幕亮度	23
4.12 关闭背光	24
4.13 开背光	25
4.14 获取背光是否打开	25
4.15 获取背光亮度值	26
4.16 关闭HDMI 输出	27
4.17 打开HDMI 输出	27
4.18 状态栏状态设置	28
4.19 状态栏状态查询	28
4.20 副屏截图	29
4.21 主屏幕类型	30

4.22 副屏幕类型	31
4.23 获取DPI	32
4.24 设置DPI	32
第五章 安装升级	34
5.1 固件升级	34
5.2 recovery 模式	34
5.3 静默安装APK	35
5.4 设置升级固件是否弹窗	36
5.5 设置升级固件时后，固件包是否从文件中删除	37
5.6 静默卸载apk	37
第六章 网络	39
6.1 查询以太网MAC 地址	39
6.2 查询以太网连接模式	39
6.3 查询以太网开关状态	40
6.4 查询以太网的子网掩码	41
6.5 查询以太网的网关	42
6.6 查询以太网的Dns1	43
6.7 查询以太网Dns2	44
6.8 设置以太网为动态获取模式	45
6.9 设置以太网为静态地址模式	46
6.10 获取以太网动态IP 地址	47

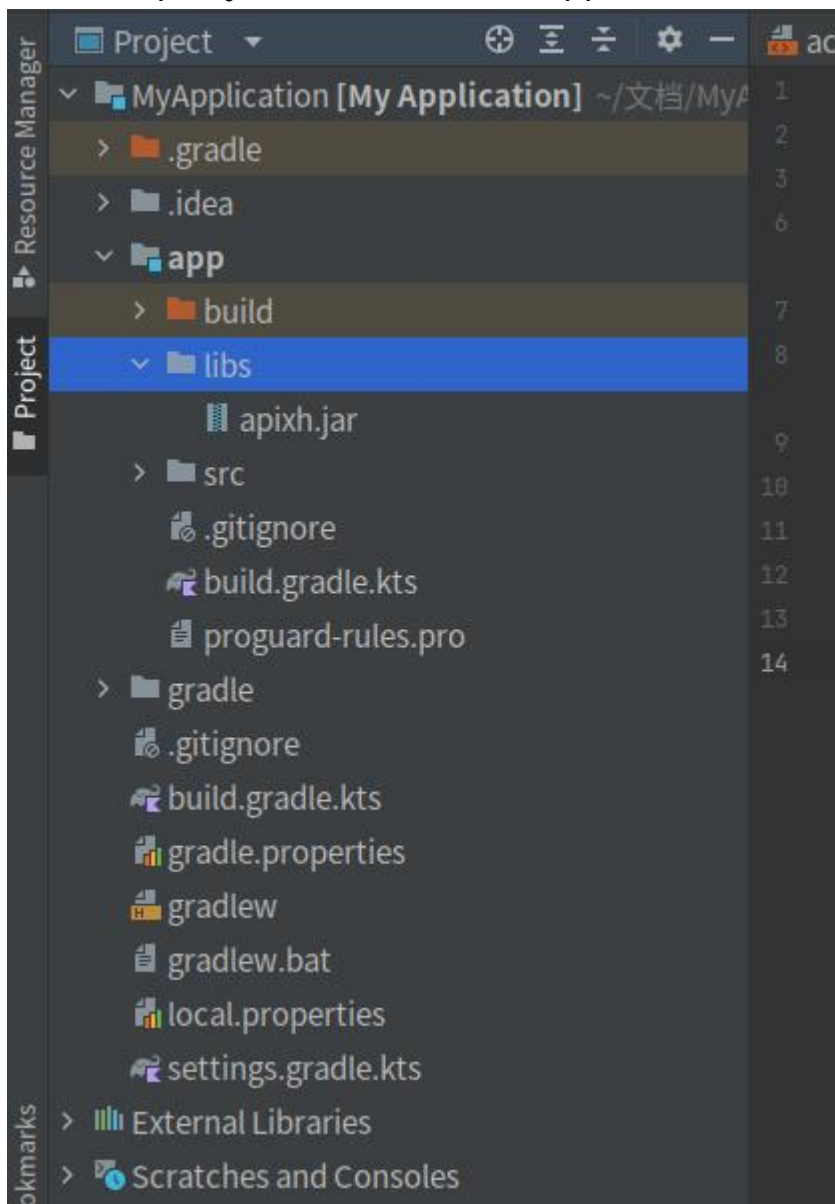
6.11 查询以太网的静态IP 地址	48
6.12 控制以太网开关	49
6.13 pppoe 拨号上网	49
第七章 存储	51
7.1 查询外部存储SD 卡路径	51
7.2 查询外部存储U 盘路径	51
7.3 查询串口绝对路径	52
第八章 定时开关机	54
8.1 设置周模式的定时开关机	54
8.2 设置一组定时开关机	55
8.3 获取当前定时开关机模式	56
8.4 获取当前定时开关机的开机时间	57
8.5 获取当前定时开关机的关机时间	57
8.6 获取设备上一次的定时开关机的开机时间	58
8.7 获取设备上一次的定时开关机的关机时间	59
8.8 清除定时开关机数据	60
8.9 获取定时开关机版本	60
8.10 获取设备是否设置了定时开关机	61
第九章 GPIO 索引值控制	62
9.1 设置io 口的状态是输入或输出	62
9.2 获取当前gpio 是输入口还是输出口	63

9.3 设置gpio 的电平	63
9.4 获取当前gpio 的电平	64
第十章 其他	66
10.1 设置并保存系统时间	66
10.2 打开或关闭自动确定时间	67
10.3 控制软键盘是否能弹出	67
10.4 休眠时间的控制	68
10.5 查询自动确定网络时间状态	69
10.6 以ROOT 权限运行shell 命令	70
10.7 查询设备上网方式	70
10.8 查询HDMI 输入状态	71
10.9 设置默认输入法	72
10.10 获取默认输入法	73
10.11 设置系统语言	74
10.12 获取设备CPU 温度	75
10.13 打开或关闭adb 连接	75
10.14 替换开机动画	76
10.15 设置默认Launcher	77
10.16 执行关机操作	78
10.17 设置开机自启	78
10.18 设置守护进程应用	79

10.19 应用安装白名单.....	80
10.20 应用安装黑名单.....	80
10.21 打开或关闭网络adb 连接.....	81

第一章 Android Studio 中使用 API

(1) 将apixh.jar 复制到【工程目录\app\libs\】下;



(2) 右键点击 libs 文件夹中的 jar 文件选择 add as Library...然后选择 Model, 这样也可以导入成功。

注意：开始api调用（获取对象实例后 务必 bind 和 unbind AIDL 进行跨进程通讯，方可使用API）； bind和unbind必须要成对出现

MyManager manager = MyManager.getInstance(this);

```
1 package com.chipamte.pxsjfloat;
2
3 > import ...
4
5 public class MainActivity extends AppCompatActivity {
6     3 usages
7     private MyManager myManager;
8     @Override
9     protected void onCreate(Bundle savedInstanceState) {
10         super.onCreate(savedInstanceState);
11         myManager = MyManager.getInstance(context: this);
12         myManager.bindAIDLService(context: this);
13     }
14
15     @Override
16     protected void onDestroy() {
17         super.onDestroy();
18         myManager.unbindAIDLService(context: this);
19     }
20 }
```

第二章 获取系统信息

2.1 获取 API 版本-日期信息

函数：public String getApiVersion()

描述：获取目前 API 的版本-日期信息

参数名/返回值	类型	说明	举例
返回值	String	版本-日期信息。	V1.0-20100000

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
Log.d(TAG,"API Version = " + manager.getApiVersion());
```

2.2 获取设备的型号

函数：public String getAndroidModle()

描述：获取目前 API 的版本-设备型号

参数名/返回值	类型	说明	举例
返回值	String	设备的型号	Mstar Android TV

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
Toast.makeText(getApplicationContext(),"Model"+manager.getAndroidModle(),Toast.LENGTH_SHORT).show();
```

2.3 获取设备的系统版本

函数：public String getAndroidVersion()

描述：获取目前设备的 android 系统的版本

参数名/返回值	类型	说明	举例
返回值	String	android 系统的版本	19

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
Toast.makeText(getApplicationContext(),"Version="+manager.getAndroidVersion(),Toast.LENGTH_SHORT).show();
```

2.4 获取设备内存容量

函数：public String getRunningMemory()

描述：获取设备的硬件内存大小容量。

参数名/返回值	类型	说明	举例
返回值	String	硬件内存大小，以 GB 为单位	2GB

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
Toast.makeText(getApplicationContext(),"RunningMemory"+manager.getRunningMemory(),Toast.LENGTH_SHORT).show();
```

```
//输出 RunningMemory2GB
```

2.5 获取设备存储容量

函数：public String getInternalStorageMemory ()

描述：获取设备的硬件内部存储大小容量

参数名/返回值	类型	说明	举例
返回值	String	硬件内部存储大小，以 GB 为单位	16GB

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
Toast.makeText(getApplicationContext(),"InternalStorageMemory:"+manager.getRunningMemory(),  
Toast.LENGTH_SHORT).show();
```

```
//输出 InternalStorageMemory:16GB
```

2.6 获取设备固件版本

函数：public String getFirmwareVersion()

描述：获取设备的固件 SDK 版本。

参数名/返回值	类型	说明	举例
返回值	String	设备的固件版本	1.0

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
Toast.makeText(getApplicationContext(),"FW:"+manager.getFirmwareVersion(),Toast.LENGTH_S  
HORT).show();
```

```
//输出 FW:1.0
```

2.7 获取设备固件内核版本

函数：public String getKernelVersion()

描述：获取设备的固件内核版本

参数名/返回值	类型	说明	举例
返回值	String	设备的固件内核版本	4.19.232

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
Log.e("SW kernel Version=" manager.getKernelVersion());
```

```
//输出 SW kernel Version =3.10.86
```

2.8 获取设备固件版本和编译日期

函数：public String getAndroidDisplay()

描述：获取设备的固件系统版本和编译日期

参数名/返回值	类型	说明	举例
返回值	String	设备的固件系统版本和编译日期	

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
Log.e("SWAndroidDisplay=" manager.getAndroidDisplay ());
```

```
//输出 SW AndroidDisplay= aosp_almond_dtmb-userdebug 6.0 MRA58K eng.ace.20180523.103817  
test-keys
```

2.9 获取固件编译时间

函数：public String getFirmwareDate()

描述：获取固件编译的时间

参数名/返回值	类型	说明	举例
返回值	String	设备的固件编译日期	2025-03-21-11-23

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
Log.d("API 显示","getFirmwareDate(): "+ manager.getFirmwareDate());
```

```
//输出 API 显示： : getFirmwareDate(): 20180602
```

2.10 获取设备 CPU 型号

函数：public String getCPUType()

描述：获取设备 CPU 型号

参数名/返回值	类型	说明	举例
返回值	String	设备的 CPU 型号	arm64-v8a

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
Log.d("API 显示","设备 CPU 型号: "+ manager.getCPUType());
```

2.11 获取厂商识别码

函数：public String getVendorID()

描述：获取厂商识别码

参数名/返回值	类型	说明	举例
返回值	String	返回厂商识别码	XH6212

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
ToastUtils.showToast(this, "厂商识别码=" + manager.getVendorID());
```

2.12 获取 SN 号

函数：public String getSn()

描述：获取 SN 号

参数名/返回值	类型	说明	举例
返回值	String	SN 号	2CE4WW1BKN

注意点：

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
ToastUtils.showToast(this, "sn=" + manager.getSn());
```

第三章 关机和重启

3.1 关机

函数: `public void shutdown()`

描述: 关机

例:

```
MyManager manager = MyManager.getInstance(this);  
manager.shutdown();
```

3.2 重启

函数: `public void reboot()`

描述: 重启

例:

```
MyManager manager = MyManager.getInstance(this);  
manager.reboot();
```

第四章 显示

4.1 截屏保存

函数：public boolean takeScreenshot(String path)

描述：截取当前全屏为图片并重命名到相应位置

参数名/返回值	类型	说明	举例
参数名	String	保存绝对路径	“/mnt/internal_sd/001.jpg”
返回值	boolean	截图是否成功	True 成功保存截图 false 失败

注意点：

- 1、path 是否是可读写目录包括文件名。
- 2、使用此方法需要绑定 aidl 服务，在创建 MyManager 对象的时候绑定，方法如下：
manager.bindAIDLService(this);
- 3、在 activity 销毁的时候需要调用 manager.unbindAIDLService(this);

例：

```
//截屏并保存到"/mnt/internal_sd/001.jpg"
```

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.takeScreenshot(Environment.getExternalStorageDirectory().getPath() + "/001.jpg")
```

4.2 设置屏幕旋转角度

函数：public void rotateScreen(Context context, String degree)

描述：设置屏幕顺时针旋转 N 角度

参数名/返回值	类型	说明	举例
degree	String	只支持 0 , 90, 180, 270 四个角度。	"0"、"90"、"180"、"270"

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.rotateScreen(context,"90");
```

4.3 获取屏幕宽像素

函数：public int getDisplayWidth(Context context)

描述：获取显示屏分辨率宽 X 像素

参数名/返回值	类型	说明	举例
参数名	Context	上下文对象	
返回值	int	屏幕实际宽像素	

例：MyManager manager = MyManager.getInstance(this);

```
manager.getDisplayWidth(context);
```

4.4 获取屏幕高像素

函数：public int getDisplayHeight()

描述：获取显示屏分辨率高 Y 像素

参数名/返回值	类型	说明	举例
参数名	Context	上下文对象	
返回值	int	屏幕实际高像素	

例：

```
//在 1080p 的屏获取分辨率
MyManager manager = MyManager.getInstance(this);

int x = manager.getDisplayWidth(context);

int y = manager.getDisplayHeight(context);

Log.d("API 显示: ", "Width = "+x+" Height = "+y);

//将输出 Width: Height= 1920:1080
```

4.5 导航栏状态设置

函数：public void hideNavBar(boolean hide)

描述：显示或隐藏导航栏

参数名/返回值	类型	说明	举例
参数名	boolean	True: 隐藏导航栏 False: 显示导航栏	

例：

```
MyManager manager = MyManager.getInstance(this);

manager.hideNavBar(true);//隐藏导航栏的写法
```

4.6 导航栏状态查询

函数：public boolean getNavBarHideState()

描述：导航栏是否隐藏

参数名/返回值	类型	说明	举例
返回值	boolean	true:隐藏， false:显示	

例：

```
MyManager manager = MyManager.getInstance(this);  
  
if(manager.getNavBarHideState()){  
    Log.e("StatusBar hide.");  
}  
else{  
    Log.e("StatusBar show.");  
}
```

4.7 设置滑出导航栏打开或关闭

函数：public void setSlideShowNavBar(boolean flag)

描述：打开或关闭滑出导航栏

参数名/返回值	类型	说明	举例
参数名	boolean	True: 打开滑出状态栏 False: 关闭滑出状态栏	

例：

```
MyManager manager = MyManager.getInstance(this);  
  
manager.setSlideShowNavBar(true);//打开滑出导航栏
```

4.8 查询滑出导航栏是否打开

函数：public boolean isSlideShowNavBarOpen

描述：滑出导航栏是否打开

参数名/返回值	类型	说明	举例
返回值	boolean	true:打开, false:关闭	

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
if(manager.isSlideShowNavBarOpen){  
    Log.e("SlideShowNavBar open.");  
}else{  
    Log.e("SlideShowNavBar close");  
}
```

4.9 设置滑出通知栏打开或关闭

函数：public void setSlideShowNotificationBar(boolean enable)

描述：打开或关闭滑出通知栏

参数名/返回值	类型	说明	举例
参数名	boolean	True: 打开滑出状态栏 False: 关闭滑出状态栏	

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.setSlideShowNavBar(true);//打开下拉通知栏
```

4.10 查询滑出通知栏状态

函数：public boolean isSlideShowNotificationBarOpen()

描述：滑出通知栏是否打开

参数名/返回值	类型	说明	举例
返回值	boolean	true:打开， false:关闭	

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
if(manager.isSlideShowNavBarOpen){
```

```
    Log.e("SlideShowNavBar open.");
```

```
}else{
```

```
    Log.e("SlideShowNavBar close");
```

```
}
```

4.11 设置屏幕亮度

函数：public void changeScreenLight(int value)

描述：调节屏幕亮度

参数名/返回值	类型	说明	举例
参数名	int	亮度的具体值（1--100）	50

注意点：参数值是按照 1 到 100 设置的，设置 100 即代表最大亮度，1 代表最小亮度

例：manager.changeScreenLight(50);

4.12 关闭背光

函数：public void turnOffBacklight()

描述：熄灭屏幕，只关背光，却不进入休眠，软件继续运行。

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.turnOffBacklight();
```

4.13 开背光

函数：public void turnOnBacklight()

描述：开背光。

例：manager.turnOnBacklight();

4.14 获取背光是否打开

函数：public boolean isBacklightOn()

描述：获取背光是否打开。

参数名/返回值	类型	说明	举例
返回值	boolean	True 表示背光打开 false 表示背光关闭	

例：

```
MyManager manager = MyManager.getInstance(this);  
manager.isBacklightOn();
```

4.15 获取背光亮度值

函数：public int getSystemBrightness()

描述：获取背光亮度值

参数名/返回值	类型	说明	举例
返回值	int	取到的亮度范围是从 0 到 100	

例：

```
MyManager manager = MyManager.getInstance(this);  
manager.getSystemBrightness();
```

4.16 关闭HDMI输出

函数: `public void turnOffHDMI()`

描述: 关闭 HDMI 输出信号

例: `MyManager manager = MyManager.getInstance(this); manager.turnOffBacklight();`

4.17 打开HDMI输出

函数: `public void turnOnHDMI()`

描述: 打开 HDMI 输出信号

例:

`MyManager manager = MyManager.getInstance(this);`

`manager.turnOnBacklight();`

4.18 状态栏状态设置

函数: `public void hideStatusBar(boolean hide)`

描述: 显示或隐藏状态栏

参数名/返回值	类型	说明	举例
参数名	boolean	True: 隐藏状态栏 False: 显示状态栏	

例:

`MyManager manager = MyManager.getInstance(this);`

`manager.hideStatusBar(true);`//隐藏状态栏的写法

4.19 状态栏状态查询

函数：public boolean getStatusBar()

描述：状态栏是否隐藏

参数名/返回值	类型	说明	举例
返回值	boolean	true:隐藏, false:显示	

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
if(manager.getStatusBar()){
```

```
    Log.e("StatusBar hide.");
```

```
}else{
```

```
    Log.e("StatusBar show.");
```

```
}
```

4.20 副屏截图

函数：public boolean viceScreenshot(String path)

描述：截取当前全屏为图片并重命名到相应位置

参数名/返回值	类型	说明	举例
参数名	String	保存绝对路径	“/mnt/internal_sd/001.jpg”
返回值	boolean	截图是否成功	True 成功保存截图 false 失败

注意点：

1、path 是否是可读写目录包括文件名。

2、使用此方法需要绑定 aidl 服务，在创建 MyManager 对象的时候绑定，方法如下：

```
manager.bindAIDLService(this);
```

3、在 activity 销毁的时候需要调用 manager.unbindAIDLService(this);

例：

```
//截屏并保存到"/mnt/internal_sd/001.jpg"
```

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.viceScreenshot(Environment.getExternalStorageDirectory().getPath() + "/001.jpg");
```

4.21 主屏幕类型

函数：public String getHomeScreenType()

描述：主屏幕类型

参数名/返回值	类型	说明	举例
返回值	String	主屏幕类型	HDMI

例：

```
MyManager manager = MyManager.getInstance(this); ToastUtils.showToast(this,
```

```
"主屏幕类型为:" + manager.getHomeScreenType());
```

4.22 副屏幕类型

函数：public String getSecondaryScreenType()

描述：副屏幕类型

参数名/返回值	类型	说明	举例
返回值	String	副屏幕类型	HDMI

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
ToastUtils.showToast(this, "副屏幕类型为:" + manager.getSecondaryScreenType());
```

4.23 获取 DPI

函数：public String getDpi()

描述：获取 DPI

参数名/返回值	类型	说明	举例
参数	String	返回当前 DPI	160

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.getDpi();
```

4.24 设置 DPI

函数：public boolean setDpi(int value)

描述：设置 DPI

参数名/返回值	类型	说明	举例
参数	int	传入的 dpi 值： 0=160,1=240 2=360,3=480	0

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.setDpi(0);
```

第五章 安装升级

5.1 固件升级

函数: `public void upgradeSystem(String absolutePath)`

描述: 升级固件

参数名/返回值	类型	说明	举例
参数名	String	固件存放的绝对路径	/mnt/internal_sd/update.img

注意点:

1. 函数将会自动重启系统进行 `update.img` 升级。
2. 保证 `img` 文件存在, 完整可用。
3. 文件名一定为 `update.img` 或 `updat.zip`。

例:

```
MyManager manager = MyManager.getInstance(this);  
manager.upgradeSystem("/mnt/internal_sd/update.img");
```

5.2 recovery 模式

函数: `public void rebootRecovery()`

描述: 将重启进入 `recovery` 模式

例:

```
MyManager manager = MyManager.getInstance(this);  
manager.rebootRecovery();
```

5.3 静默安装APK

函数：public boolean silentInstallApk(String apkPath)

描述：静默安装 APK 应用。

参数名/返回值	类型	说明	举例
参数名	String	待安装 APK 的绝对路径	"/mnt/internal_sd/test.apk"
返回值	Boolean	静默安装是否成功	

注意点：

1. APK 文件存在，完整可用。

例：

//安装 test.apk

```
MyManager manager = MyManager.getInstance(this);
```

```
boolean success = manager.silentInstallApk("/mnt/usb/0000-0000/YiShengTest.apk");
```

```
ToastUtils.showToast(this,"静默升级是否成功" + success);
```

5.4 设置升级固件是否弹窗

函数：public void setUpgradeSystemWithDialog(boolean flag)

描述：设置升级固件是否弹窗

参数名/返回值	类型	说明	举例
参数名	boolean	控制升级固件是否弹窗	True 弹窗 false 不弹窗

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.setUpdateSystemWithDialog(true);//设置升级固件弹窗确认
```

5.5 设置升级固件时后，固件包是否从文件中删除

函数：public void setUpdateSystemDelete(boolean flag)

描述：设置升级固件时后，固件包是否从文件中删除

参数名/返回值	类型	说明	举例
参数名	boolean	控制升级固件是否删除	True 删除 false 不删除

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.setUpdateSystemDelete(true);//设置升级成功删除固件
```

5.6 静默卸载apk

函数：public boolean unInstallApk(String packagename)

描述：静默卸载 apk

参数名/返回值	类型	说明	举例
参数名	String	需要卸载的包名	com.bjw.ComAssistant

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.selfStart("com.bjw.ComAssistant");
```

第六章 网络

6.1 查询以太网MAC 地址

函数：public String getEthMacAddress()

描述：获取以太网的 MAC 地址

参数名/返回值	类型	说明	举例
返回值	String	以太网的 MAC 地址	30:1F:9A:61:BA:8F

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
Log.e("ETH MAC=" manager.getEthMacAddress ());
```

6.2 查询以太网连接模式

函数：public String getEthMode()

描述：获取当前以太网的模式

参数名/返回值	类型	说明	举例
返回值	String	以太网模式	StaticIp（静态） DHCP（动态）

注意点：

①、在获取以太网静态 IP 之前，需要绑定获取以太网静态 IP 的 aidl 服务，在创建 MyManager 对象的时候绑定，方法如下：manager.bindAIDLService(this);

②、在 activity 销毁的时候需要调用 manager.unbindAIDLService(this);

例：

```
MyManager manager = MyManager.getInstance(this);  
  
Log.d("API 显示","当前模式 =" +manager.getEthMode());  
  
//输出 当前模式 = DHCP
```

6.3 查询以太网开关状态

函数：public boolean getEthStatus()

描述：获取当前以太网开关是否开启

参数名/返回值	类型	说明	举例
返回值	boolean	以太网开关状态	true（打开） false（关闭）

注意点：

- ①、在获取以太网是否开启之前，需要绑定 aidl 服务，在创建 MyManager 对象的时候绑定，方法如下：manager.bindAIDLService(this);
- ②、在 activity 销毁的时候需要调用 manager.unbindAIDLService(this);

例：

```
MyManager manager = MyManager.getInstance(this);  
  
Log.d("API 显示","以太网是否打开 =" +manager.getEthStatus());
```

6.4 查询以太网的子网掩码

函数：public String getEthNetMask()

描述：获取当前以太网的子网掩码

参数名/返回值	类型	说明	举例
返回值	String	子网掩码	255.255.255.0

注意点:

①、在获取以太网是否开启之前，需要绑定 aidl 服务，在创建 MyManager 对象的时候绑定，方法如下：manager.bindAIDLService(this);

②、在 activity 销毁的时候需要调用 manager.unbindAIDLService(this);

例:

```
MyManager manager = MyManager.getInstance(this);
```

```
Log.d("API 显示","以太网的子网掩码 "+manager.getNetMask());
```

6.5 查询以太网的网关

函数：public String getGateway ()

描述：获取当前以太网的网关

参数名/返回值	类型	说明	举例
返回值	String	网关	192.168.1.1

注意点:

①、在获取以太网是否开启之前，需要绑定 aidl 服务，在创建 MyManager 对象的时候绑定，方法如下：manager.bindAIDLService(this);

②、在 activity 销毁的时候需要调用 manager.unbindAIDLService(this);

例:

```
MyManager manager = MyManager.getInstance(this);
```

```
Log.d("API 显示","以太网的网关 "+manager.getGateway());
```

6.6 查询以太网的Dns1

函数：public String getEthDns1 ()

描述：获取当前以太网的 Dns1

参数名/返回值	类型	说明	举例
返回值	String	Dns1	192.168.1.1

注意点：

①、在获取以太网是否开启之前，需要绑定 aidl 服务，在创建 MyManager 对象的时候绑定，方法如下：manager.bindAIDLService(this);

②、在 activity 销毁的时候需要调用 manager.unbindAIDLService(this);

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
Log.d("API 显示","以太网的 dns1 =" +manager.getGateway());
```

6.7 查询以太网Dns2

函数：public String getEthDns2()

描述：获取当前以太网的 Dns2

参数名/返回值	类型	说明	举例
返回值	String	Dns2	192.168.1.1

注意点：

①、在获取以太网是否开启之前，需要绑定 aidl 服务，在创建 MyManager 对象的时候绑定，方法如下：manager.bindAIDLService(this);

②、在 activity 销毁的时候需要调用 manager.unbindAIDLService(this);

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
Log.d("API 显示","以太网的 dns2 =" +manager.getGateway());
```

```
//输出 以太网的 dns2 = 192.168.1.1
```

6.8 设置以太网为动态获取模式

函数：public void setDhcpIpAddress(Context context)

描述：设置以太网模式为动态获取

参数名/返回值	类型	说明	举例
参数名	Context	上下文对象	

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.setDhcpIpAddress(this);
```

6.9 设置以太网为静态地址模式

函数：public void setStaticIPAddress(String IPAddr, String gateWay, String mask, String dns1, String dns2)

描述：设置以太网的模式为静态地址

参数名/返回值	类型	说明	举例
IPAddr	String	IP 地址	"192.168.1.125"
gateWay	String	网关	"192.168.1.1"
mask	String	子网掩码	"255.255.255.0"
dns1	String	DNS	"192.168.1.1"
ipLength	String	DNS	"0.0.0.0"

注意点：网关、子网掩码和 DNS 都要设置为合法数据

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.setEthIPAddress("192.168.1.125", "192.168.1.1", "255.255.255.0", "192.168.1.1", "0.0.0.0");
```

6.10 获取以太网动态IP 地址

函数：public String getDhcpIpAddress()

描述：获取以太网的动态 IP 地址

参数名/返回值	类型	说明	举例
返回值	String	以太网的动态 IP 地址	192.168.1.100

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
Log.d("API 显示", "ETH IP =" + myManager.getDhcpIpAddress());
```

6.11 查询以太网的静态IP 地址

函数：public String getStaticEthIPAddress()

描述：获取以太网的静态 IP 地址

参数名/返回值	类型	说明	举例
返回值	String	以太网的静态 IP 地址	192.168.1.100

注意点：

①、在获取以太网静态 IP 之前，需要绑定获取以太网静态 IP 的 aidl 服务，在创建 myManager 对象的时候绑定，方法如下：myManager.bindAIDLService(this);

②、在 activity 销毁的时候需要调用 myManager.unbindAIDLService(this);

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
Log.d("API 显示","ETH IP =" +manager.getStaticEthIPAddress());
```

6.12 制以太网开关

函数：public void ethEnabled(boolean enable)

描述：开关以太网

参数名/返回值	类型	说明	举例
返回值	boolean	开关以太网	true 开 false 关

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.ethEnabled(false);//关以太网
```

6.13 pppoe 拨号上网

函数：public void setPppoeDial(Context context,String userName,String password)

描述：设置 pppoe 拨号上网

参数名/返回值	类型	说明	举例
参数名	Context	上下文对象	
参数名	String	拨号上网的用户名	
参数名	String	拨号上网的密码	

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.setPppoeDial(this,"username","password");
```

第七章 存储

7.1 查询外部存储SD 卡路径

函数：public String getSDcardPath()

描述：获取外部存储 SD 卡路径

参数名/返回值	类型	说明	举例
返回值	String	外部存储 SD 卡绝对路径	

例：

```
MyManager manager = MyManager.getInstance(this);  
Log.d("API 测试","SD="+manager.getSDcardPath());  
//输出 API: 外置 SD 卡路径 = /mnt/external_sd
```

7.2 查询外部存储U 盘路径

函数：public String getUSBStoragePath(int num)

描述：获取外部存储 U 盘路径

参数名/返回值	类型	说明	举例
num	int	U 盘盘符	0,1,2,3
返回值	String	外部存储 U 盘绝对路径	

注意点：

需在源码设置路径，才能正常运行 Demo

例：

```
MyManager manager = MyManager.getInstance(this);  
Log.d("API 测试","USB Path = "+manager.getUSBStoragePath(0));  
//输出/mnt/usb_storage/USB_DISK2/udisk0;
```

7.3 查询串口绝对路径

函数：public String getUartPath(String uart)

描述：获取串口绝对路径。

参数名/返回值	类型	说明	举例
Uart	String	串口对应的端口号- uart 0, uart 1, uart 2, uart 3	
返回值	String	UART 绝对路径	

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
Log.d("API 测试","UART-1="+manager. getUartPath (uart1));
```

```
//输出 UART-1=/dev/ttyS1
```

第八章 定时开关机

8.1 设置周模式的定时开关机

函数：`public void setPowerOnOffWithWeekly(int[] powerOnTime,int[] powerOffTime,int[] weekdays)`

描述：设置定时开关机，周模式，一天只有一组开关机时间

参数名/返回值	类型	说明	举例
powerOnTime	Int[]	开机时间，时分	{8,30}
powerOffTime	Int[]	关机时间，时分	{18,30}
weekdays	Int[]	周一到周日的工作状态	{1,1,1,1,0,0,1}

注意点：

此方法一天只能有一组时间设入，并且开机时间在前，关机时间在后

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
int [] timeonArray = new int {8,30};
```

```
int [] timeoffArray = new int {18,30};
```

```
Int[] weekdays = new int {1,1,1,1,1,0,0}; //周一到周日工作状态,1 为开机，0 为不开机
```

```
manager.setPowerOnOffWithWeekly(timeonArray,timeoffArray,weekdays);
```

设置上述时间将会在每周一到周五的 8:30 开机，18:30 关机。

8.2 设置一组定时开关机

函数：public void setPowerOnOff(int[] powerOnTime,int[] powerOffTime)

描述：设置一组定时开关机，一天可设置多次，需传入年月日时分

参数名/返回值	类型	说明	举例
powerOnTime	Int[]	开机时间，年月日时分	{2020,01,13,18,40}
powerOffTime	Int[]	关机时间，年月日时分	{2020,01,13,18,30}

注意点：

此方法设置的时候，关机时间在前，开机时间在后

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
int [] timeoffArray = new int{2018,1,10,20,38};
```

```
int [] timeonArray = new int{2018,1,10,20,48};
```

```
manager.setPowerOnOff(timeonArray,timeoffArray);
```

设置上述时间将会在 2018 年 1 月 10 号，20:38 关机，20:48 开机

注意：该方法同样适用于只设置开机时间，只需将关机时间传 0 即可，即

```
int [] timeoffArray = new int{0,0,0,0,0};
```

8.3 获取当前定时开关机模式

函数：public String getPowerOnMode()

描述：获取定时开关机模式

参数名/返回值	类型	说明	举例
返回值	String	开关机模式	“0”

例：

```
MyManager manager = MyManager.getInstance(this);  
manager.getPowerOnMode()
```

8.4 获取当前定时开关机的开机时间

函数：public String getPowerOnTime()

描述：获取定时开关机开机时间

参数名/返回值	类型	说明	举例
返回值	String	定时开关机开机时间	202001132025

例：

```
MyManager manager = MyManager.getInstance(this);  
manager.getPowerOnTime();
```

8.5 获取当前定时开关机的关机时间

函数：public String getPowerOffTime()

描述：获取定时开关机关机时间

参数名/返回值	类型	说明	举例
返回值	String	定时开关机关机时间	202001132020

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.getPowerOffTime();
```

8.6 获取设备上一次的定时开关机的开机时间

函数：public String getLastestPowerOnTime()

描述：获取设备上一次执行过的开机时间

参数名/返回值	类型	说明	举例
返回值	String	上一次的定时开关机开机时间	202001132025

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.getLastestPowerOnTime();
```

8.7 获取设备上一次的定时开关机的关机时间

函数：public String getLastestPowerOffTime()

描述：获取设备上一次执行过的关机时间

参数名/返回值	类型	说明	举例
返回值	String	上一次的定时开关机关机时间	202001132020

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.getLastestPowerOffTime();
```

8.8 清除定时开关机数据

函数：public void clearPowerOnOffTime()

描述：清除定时开关机时间

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.clearPowerOnOffTime();
```

8.9 获取定时开关机版本

函数：public String getVersion()

描述：获取定时开关机版本号

参数名/返回值	类型	说明	举例
返回值	String	定时开关机版本	XH_1.0_20180312

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.getVersion();
```

8.10 获取设备是否设置了定时开关机

函数：public boolean isSetPowerOnTime()

描述：获取设备是否设置了定时开关机

参数名/返回值	类型	说明	举例
返回值	boolean	是否设置了定时开关机	True

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.isSetPowerOnTime();
```

第九章 GPIO 索引值控制

9.1 设置 io 口的状态是输入或输出

函数：public boolean setGpioDirection(int gpio, int arg)

描述：根据 io 口值设置该 gpio 是输入还是输出

参数名/返回值	类型	说明	举例
返回值	Boolean	设置是否成功	成功-true 失败-false
参数值	Gpio	Gpio 的值	1
参数值	Arg	输入还是输出	输入口-1 输出口-0

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.setGpioDirection(1, 0);
```

9.2 获取当前gpio 是输入口还是输出口

函数：public String getGpioDirection(int gpio)

描述：根据具体的 gpio 值获取当前 gpio 的状态是输入还是输出。该方法也可判断该 gpio 是否有效，如果没有返回值，就说明该 gpio 不可用

参数名/返回值	类型	说明	举例
返回值	String	输入还是输出	输入-in 输出-out
参数值	Int	Gpio 值	1

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.getGpioDirection(1);
```

9.3 设置 gpio 的电平

函数：public boolean writeGpioValue(int gpio, String arg)

描述：根据具体的 gpio 值控制该 gpio 的电平

参数名/返回值	类型	说明	举例
返回值	Boolean	写入是否成功	成功--true 失败--false
参数名	Int	Gpio 值	1
参数名	String	高低电平	高--1 低--0

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.writeGpioValue(1, 1);
```

9.4 获取当前 gpio 的电平

函数：public String getGpioValue(int gpio)

描述：获取当前 gpio 的电平

参数名/返回值	类型	说明	举例
返回值	String	高电平或低电平	高电平--1 低电平--0
参数名	Int	Gpio 值	1

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.getGpioValue(1);
```

第十章 其他

10.1 设置并保存系统时间

函数：public void setTime(int year, int month, int day, int hour, int minute,int sec)

描述：设置并保存系统时间

参数名/返回值	类型	说明	举例
year	int	年	
month	int	月（1 到 12）	
day	int	日	
hour	int	（24 小时制）小时	
minute	int	（24 小时制）分钟	
sec	int	秒	
返回值	void		

例：

```
//设置 2016-03-16, 13:44:20
```

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.setTime (2016, 3, 16, 13, 44,20);
```

10.2 打开或关闭自动确定时间

函数：public void switchAutoTime(boolean open)

描述：设置打开或关闭自动确定时间

实

参数名/返回值	类型	说明	举例
参数名	boolean	True 打开自动确定时间 False 关闭自动确定时间	

例：

```
MyManager manager = MyManager.getInstance(this);
manager.switchAutoTime(false); // 关闭自动确定日期和时间
```

10.3 控制软键盘是否能弹出

函数：public void setSoftKeyboardHidden(boolean hidden)

描述：控制软键盘是否能弹出

参数名/返回值	类型	说明	举例
参数名	boolean	True 软键盘可以弹出 False 软键盘不能弹出	

例：

```
MyManager manager = MyManager.getInstance(this);
manager.setSoftKeyboardHidden(false);
```

10.4 休眠时间的控制

函数：public void setDormantInterval(Context context, long time)

描述：休眠时间的控制

参数名/返回值	类型	说明	举例
---------	----	----	----

返回值	long	间隔时间（如果关闭休眠，传的值是 2147483647）	如果间隔 15s 休眠，传的参 数是 15000
-----	------	------------------------------	-----------------------------

例：

```
MyManager manager = MyManager.getInstance(this);
manager.setDormantInterval(context,15000);
```

10.5 查询自动确定网络时间状态

函数：public boolean isAutoSyncTime()

描述：获取自动确定网络时间的开关是否开启

参数名/返回值	类型	说明	举例
返回值	boolean	True 打开自动确定时间 False 关闭自动确定时间	

例：

```
MyManager manager = MyManager.getInstance(this);
manager.isAutoSyncTime();
```

10.6 以 ROOT 权限运行 shell 命令

函数：public void execSuCmd (String command)

描述：将以 ROOT 权限运行 shell 命令

实

参数名/返回值	类型	说明	举例
command	String	Shell 命令	“reboot”
返回值	void		

例:

```
//查看目录并用 shell 命令安装 APK
```

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.execSuCmd ("ls /mnt/sdcard/ ");
```

```
manager.execSuCmd ("system/bin/pm -install -r " + "mnt/sdcard/Update.apk ");
```

10.7 查询设备上网方式

函数: `public int getCurrentNetType()`

描述: 获取当前的上网类型

参数名/返回值	类型	说明	举例
返回值	int	0 以太网, 1 wifi, 2 移动网络 -100 未知网络	

例:

```
MyManager manager = MyManager.getInstance(this);
```

```
Int type = manager.getCurrentNetType();
```

10.8 查询 HDMI 输入状态

函数: `public int getHdmiinStatus()`

描述: HDMI 输入状态

参数名/返回值	类型	说明	举例
返回值	int	返回 0 没有 HDMI 输入, 1 有 HDMI 输入	

例:

```
MyManager manager = MyManager.getInstance(this);
```

```
Toast.makeText(getApplicationContext(), " HDMI" +  
manager.getHdmiinStatus(), Toast.LENGTH_SHORT).show();
```

10.9 设置默认输入法

函数： `public boolean isSetDefaultInputMethodSuccess(String defaultInputMethod)`

描述： 设置默认输入法

参数名/返回值	类型	说明	举例
参数名	String	默认输入法 apk 的包名	谷歌拼音
			<code>com.google.android.inputmethod.pinyin/.PinyinIME</code>
返回值	Boolean	是否设置成功	成功返回 true 失败返回 false

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.isSetDefaultInputMethodSuccess("com.google.android.inputmethod.pinyin/.PinyinIME");
```

10.10 获取默认输入法

函数：public String getDefaultInputMethod()

描述：获取默认输入法

参数名/返回值	类型	说明	举例
参数名	String	当前输入法 apk 的包名	谷歌拼音 com.google.android.inputmet hod.pinyin

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.getDefaultInputMethod();
```

10.11 设置系统语言

函数：public void setLanguage(String language,String country)

描述：设置系统语言

参数名/返回值	类型	说明	举例
参数名	String	设置的语言（小写字母）和国家（大写字母）	简体中文 ch CN 中文台湾 zh TW 加拿大英语 en CA

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.setLanguage("zh","CN");
```

10.12 获取设备CPU 温度

函数：public String getCPUTemperature()

描述：获取设备 CPU 温度。

参数名/返回值	类型	说明	举例
参数名	String	设备的 CPU 温度	

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
Log.d("API 显示","设备 CPU 温度: "+ manager.getCPUTemperature());
```

10.13 打开或关闭adb 连接

函数：public void setADBOpen(boolean open)

描述：打开或关闭 adb

参数名/返回值	类型	说明	举例
参数	Boolean	是否打开 adb，true 打开，false 关闭	

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.setADBOpen(true);
```

10.14 替换开机动画

函数：public void replaceBootanimation(String path)

描述：替换开机动画

参数名/返回值	类型	说明	举例
参数	String	开机动画所在路径	/storage/emulated/0/bootanimation.zip

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.replaceBootanimation("/storage/emulated/0/bootanimation.zip");
```

10.15 设置默认Launcher

函数：public void setDefaultLauncher(String packageAndClassName)

描述：系统有一个以上的 Launcher，可设置默认 Launcher

参数名/返回值	类型	说明	举例
参数	String	默认 Launcher 的包名和启动类名	Launcher3 的包名和启动类名 "com.android.launcher3/com.android.launcher3.Launcher"

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.setDefaultLauncher("com.android.launcher3/com.android.launcher3.Launcher");
```

10.16 执行关机操作

函数：public void shutdown()

描述：休眠之后唤醒系统的方法

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.shutdown();
```

10.17 设置开机自启

函数：public void selfStart(String packagename)

描述：设置开机自启 app

参数名/返回值	类型	说明	举例
参数	String	应用包名	rkandroidapi.ys.com.rkandroidapi

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.selfStart("rkandroidapi.ys.com.rkandroidapi");
```

10.18 设置守护进程应用

函数：daemon(String packageName, int value)

描述：设置守护进程应用

参数名/返回值	类型	说明	举例
参数	String	需要守护应用的包名	rkandroidapi.ys.com.rkandroidapi
参数	int	守护的时间	0

例：

```
MyManager manager = MyManager.getInstance(this);  
manager.daemon("rkandroidapi.ys.com.rkandroidapi",1);
```

10.19 应用安装白名单

函数：public void setAppInstallWhitelist(String isopen, String packageName)

描述：应用安装白名单

参数名/返回值	类型	说明	举例
参数	String	true 启用白名单 false 不启用	true
参数	String	需要增加白名单的应 用包名	rkandroidapi.ys.com.rkandroidapi

例：

```
MyManager manager = MyManager.getInstance(this);  
manager.setAppInstallWhitelist("true","rkandroidapi.ys.com.rkandroidapi");
```

10.20 应用安装黑名单

函数：public void setAppInstallBlacklist(String isopen, String packageName)

描述：删除应用安装白名单

参数名/返回值	类型	说明	举例
参数	String	true 启用黑名单 false 不启用	true
参数	String	需要增加黑名单的应 用包名	rkandroidapi.ys.com.rkandroidapi

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.setAppInstallBlacklist("true","rkandroidapi.ys.com.rkandroidapi");
```

10.21 打开或关闭网络adb 连接

函数：public void setADBOpen(boolean open)

描述：打开或关闭 adb

参数名/返回值	类型	说明	举例
参数	Boolean	是否打开网络 adb，true 打开，false 关闭	

例：

```
MyManager manager = MyManager.getInstance(this);
```

```
manager.setNetworkAdb(true);
```