



K10-A7

API Development Manual

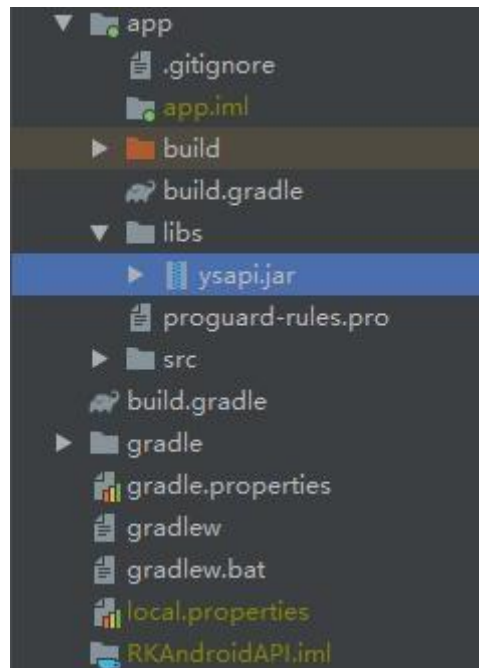
Table of Contents

Chapter 1 How to use API in Android Studio.....	1
Chapter 2 Obtaining System Information.....	3
Chapter 3 Shutdown and Restart.....	12
3.1 Shutdown.....	12
3.2 Restart.....	12
Chapter 4 Display.....	14
4.11 Set screen brightness.....	22
4.12 Turn off the backlight.....	23
4.13 Turn on the backlight.....	24
4.15 Get backlight brightness value.....	25
4.16 Turn off HDMI output.....	26
4.17 Turn on HDMI output.....	26
Chapter 5 Installation and Upgrade.....	28
5.1 Firmware upgrade.....	28
5.2 recovery mode.....	28
5.3 Silently install APK.....	29
5.4 Set whether the upgrade firmware pops up.....	30
Chapter 6 Network.....	32
6.1 Query Ethernet MAC address.....	32
6.2 Query Ethernet connection mode.....	32
6.3 Query Ethernet switch status.....	33
6.4 Query Ethernet subnet mask.....	34
6.5 Query Ethernet gateway.....	35
6.6 Query Ethernet Dns1.....	36
6.7 Query Ethernet Dns2.....	37
6.8 Set Ethernet to dynamic acquisition mode.....	38
6.9 Set Ethernet to static address mode.....	39
6.10 Get Ethernet dynamic IP address.....	40

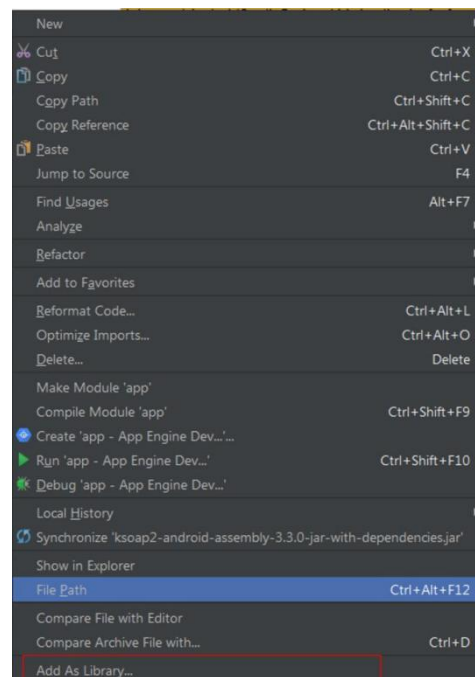
6.11 Query the static IP address of the Ethernet.....	41
6.12 Control Ethernet switch	42
6.13 pppoe dial-up internet.....	42
Chapter 7 Storage	44
7.1 Query external storage SD card path.....	44
7.2 Query the path of the external storage U disk.....	44
7.3 Query the absolute path of the serial port.....	45
Chapter 8 Timer Switch	47
Chapter 9 GPIO Index Value Control	55
9.1 Apply permission for export	55
9.6 Set gpio level	59
9.7 Get the current gpio level.....	60
Chapter 10 Other	61
10.4 Sleep time control	64
10.6 Run shell commands with ROOT privileges.....	65
10.8 Query HDMI input status.....	68
10.9 Query the current number of screens	69
10.10 Query Android layer log	70
10.11 Get the Kernel layer log.....	71
10.12 Set default input method	71
10.13 Get the default input method	72
10.14 Set system language	73
10.15 Get device CPU temperature	74
10.16 Turn on or off adb connection	75
10.17 Replace boot animation.....	75
10.18 Set default Launcher	76

Chapter 1 How to use API in Android Studio

- Copy ysapi.jar to [Project Directory]\app\libs\];



- Right-click the jar file in the libs folder, select add as Library... and then select Model, so that the import can be successful.



Note: all api calls

MyManager manager = MyManager.getInstance(this);

Start using API

First, declare the MyManager object, and then you can start using the API. For example:

```
//Declare MyManager object  
private MyManager manager;  
manager = MyManager.getInstance(this);
```

```
//Use API  
manager.getApiVersion();
```

Chapter 2 Obtaining System Information

2.1 Get API version-date information

function: public String getApiVersion ()

description: Get the current API version-date

Information API version that implements this interface: V1.0-20180602

```
/**  
 * @method getApiVersion()  
 * @description Get current API platform-version-date information, if the API is modified,  
 * modify here  
 * @date 20180602  
 * @author sky  
 * @return Version information of the current API  
 */
```

Parameter name/return value	Types	Description	For example
return value	String	Version-date information	V1.0-20180602

example:

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

```
Log.d(TAG, "API Version = " +
```

```
manager.getApiVersion());
```

```
//output APIINFO: API Version = V1.0-20180602
```

2.2 Get the model number of the device

Function: public String getAndroidModle()

Description: Get the current device model number

API version that implements this interface: V1.0-20180602

```
/**  
 * @method getAndroidModle()  
 * @description Get the current device model, for example rk3288. rk3399  
 * @date 20180602  
 * @author sky  
 * @return device model number  
 */
```

Parameter name/return value	Types	Description	For example
return value	String	Device model number	Mstar Android TV

example:

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

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

```
//output Model= rk3288
```

2.3 Get the system version of the device

function: public String getAndroidVersion()

description: Get the version of the Android system of the current device to

implement the API version of this interface: V1.0-20180602

```
/**  
 * @method getAndroidVersion()  
 * @description Get the Android system version of the current device, for example, 7.1  
 returns 25  
 * @date 20180602
```

```

* @author sky
* @return android system version
*/

```

Parameter name/return value	Types	Description	For example
return value	String	android system version	19

example:

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

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

```
//output Version=19
```

2.4 Get device memory capacity

function: public String getRunningMemory() description: Get the hardware memory capacity of the device. API version that implements this interface: V1.0-20180602

```

/**
 * @method getRunningMemory()
 * @description Get the hardware memory capacity of the device
 * @date 20180602
 * @author sky
 * @return memory size
 */

```


Parameter name/return value	Types	Description	For example
return value	String	Hardware memory size in GB	2GB

example:

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

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

```
//output RunningMemory2GB
```

2.5 Get device storage capacity

```
function: public String getInternalStorageMemory ()
```

description: Get the device's hardware internal storage size and capacity to implement the API version of this interface: V1.0-20180602

```
/**
```

```
 * @method getInternalStorageMemory()
```

```
 * @description Get the storage capacity of the device
```

```
 * @date 20180602
```

```
 * @author sky
```

```
 * @return Device internal storage size
```

```
*/
```

```
public String getInternalStorageMemory();
```

Parameter name/return value	Types	Description	For example
return value	String	Hardware internal storage size, in GB	16GB

example:

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

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

```
//output InternalStorageMemory:16GB
```

2.6 Get device firmware version

```
function: public String getFirmwareVersion()
```

description: Obtain the firmware SDK version of the device.

API version that implements this interface: V1.0-20180602

```
/**  
 * @method getFirmwareVersion()  
 * @description Get the firmware SDK version of the device  
 * @date 20180602  
 * @author sky  
 * @return Device SDK version  
 */
```

Parameter name/return value	Types	Description	For example
return value	String	The firmware version of the device	1.0

example:

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

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

```
//output FW:1.0
```

2.7 Get device firmware kernel version

function: public String getKernelVersion()

description: Get device firmware kernel version

API version that implements this interface: V1.0-20180802

```
/**  
 * @method getKernelVersion()  
 * @description Get the firmware kernel version of the device  
 * @date 20180802  
 * @author sky  
 * @return Kernel version  
 */
```

Parameter name/return value	Types	Description	For example
return value	String	The firmware kernel version of the device	3.10.86

example:

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

```
Log.e("SW kernel Version="
```

```
manager.getKernelVersion());
```

```
//output SW kernel Version =3.10.86
```

2.8 Get device firmware version and compilation date

function: public String getAndroidDisplay()

description: Get the device's firmware system version and compilation date to

implement the API version of this interface: V1.0-20180602

```
/**
```

```
 * @method getAndroidDisplay()
```

```
 * @description Get the firmware version number
```

```
 * @date 20180602
```

```
 * @author sky
```

```
 * @return The firmware version number of the device
```

```
*/
```

Parameter name/return value	Types	Description	For example
return value	String	The firmware system version and compilation date of the device	

example:

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

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

```
());
```

```
//outputSW AndroidDisplay= aosp_almond_dtmb-userdebug 6.0 MRA58K  
eng.ace.20180523.103817
```

```
test-keys
```

2.9 Get firmware compilation time

```
function: public String getFirmwareDate()
```

```
description: Get firmware compilation time
```

```
API version that implements this interface: V1.0-20180602
```

```
/**
```

```
 * @method getFirmwareDate()
```

```
 * @description Compile time of firmware
```

```
 * @date 20180602
```

```
 * @author sky
```

```
 * @return Compile time of firmware, for example 20180602
```

```
*/
```

Parameter name/return value	Types	Description	For example
-----------------------------	-------	-------------	-------------

return value	String	The device's firmware compilation date	20180602
--------------	--------	--	----------

example:

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

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

```
//output API display : : getFirmwareDate(): 20180602
```

2.10 Get device CPU model

```
function: public String getCPUType()
```

description: Get device CPU model

API version that implements this interface: V3.0-20190118

```
/**
 * @method getCPUType()
 * @description Get device CPU model
 * @date 20190118
 * @author sky
 * @return device CPU model
 */
```

Parameter name/return value	Types	Description	For example
return value	String	The CPU model of the device	

example:

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

```
Log.d("API display ", "The CPU model of the device: "+ manager.getCPUType());
```

Chapter 3 Shutdown and Restart

3.1 Shutdown

function: public void shutdown()

description: Shutdown

API version that implements this interface: V1.0-20180602

```
/**
 * @method shutdown()
 * @description Perform shutdown operation, follow Android standard shutdown process
 * @date 20180602
 * @author sky
 */
```

example:

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

3.2 Restart

function:: public void reboot()

description: restart

API version that implements this interface: V1.0-20180602

```
/**
 * @method reboot()
 * @description Perform the restart operation, follow the Android standard restart
 process
```

```
* @date 20180602
* @author sky
*/
public void reboot();
```

example:

```
MyManager manager =
```

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


Chapter 4 Display

4.1 Screenshot save

function: public boolean takeScreenshot(String path) description: Capture the current full screen as a picture and rename it to the corresponding location API version that implements this interface: V1.0-20180602

```
/**
 * @method takeScreenshot
 * @description Screenshot,Executing this method can save the screenshot of the
   current display screen of the system in the specified path
 * @date 20180602
 * @author sky
 * @param path, Path to save the picture。 For example/mnt/internal_sd/001.jpg
 * @return Whether the screenshot is successful, true success, false failure
 *
 */
```

Parameter name/return value	Types	Description	For example
parameter name	String	Save absolute path	"/mnt/internal_sd/001.jpg"
return value	boolean	Whether the screenshot is successful	True Screenshot saved successfully false failure

Points to note:

1. Whether path is a readable and writable directory including file name
2. To use this method, you need to bind the aidl service, which is bound when creating the MyManager object. The method is as follows:
manager.bindAIDLService(this);
3. You need to call manager.unbindAIDLService(this) when the activity is destroyed;

example:

```
//Screenshot and save to"/mnt/internal_sd/001.jpg"  
MyManager manager = MyManager.getInstance(this);  
  
manager.takeScreenshot(Environment.getExternalStorageDirectory().getPath()  
+ "/001.jpg");
```

4.2 Set the screen rotation angle

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

description: Set the screen to rotate clockwise by N angle

API version that implements this interface: V1.0-20180602

```
/**  
 * @method rotateScreen(Context context, String degree)  
 * @description Rotate the screen orientation, 0 degrees, 90 degrees, 180 degrees and  
 * 270 degrees  
 * @date 20180602  
 * @author sky  
 * @param context, Context object. degree, Angle of rotation (0/90/180/270)  
 */
```

Parameter name/return value	Types	Description	For example
degree	String	Only supports 0, 90, 180, 270 four corners degree.	"0" "90" "180" "270"

example:

```
MyManager manager =
```

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

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

4.3 Get screen width pixels

function: public int getDisplayWidth(Context context)

description: Get the display resolution width

X pixels API version that implements this

interface: V1.0-20180602

```
/**  
 * @method getDisplayWidth(Context context)  
 * @description Get screen width  
 * @date 20180602  
 * @author sky  
 * @param context, Context object  
 * @return Screen width  
 */  
  
public int getDisplayWidth(Context context)
```

Parameter name/return value	Types	Description	For example
parameter name	Context	Context object	
return value	int	Actual screen width in pixels	

```
example: MyManager manager =  
MyManager.getInstance(this);  
manager.getDisplayWidth(context);
```

4.4 Get screen high pixels

function: public int getScreenHeight()

description: Get high Y pixels of display resolution

API version that implements this interface: V1.0-20180602

```

/*      ption Get screen height
 *
 *      * @date 20180602
 * @
 * m
 * e
 * t
 * h
 * o
 * d */

```

Parameter name/return value	Types	Description	For example
parameter name	Context	Context object	
return value	int	Actual high pixels of the screen	

example:

```

//Get resolution on 1080p screen
g MyManager manager =
h
t MyManager.getInstance(this); int x =
(
C manager.getDisplayWidth(context);
o
n int y =
t
e manager.getDisplayHeight(context);
x
t Log.d("API display: ", "Width = "+x+"
    Height = "+y);
c
o//set output Width: Height= 1920:1080
n
t
e

```

4.5 Navigation bar status settings

function: public void hideNavBar(boolean hide)

* @description: Show or hide the navigation bar

API version that implements this interface: V1.0-20180602

```

/*
 *
 */

```

e navigation

```
* @
m
e
t
h
o
d
h
i
d
e
N
a
v
B
a
r(
b
o
o
l
e
a
n
h
i
d
e)
*/
* @
d
e
s
c
r
i
p
t
i
o
n
S
e
t
t
o
s
h
o
w
o
r
h
i
d
```

Parameter name/return value	Types	Description	For example
parameter name	boolean	True: Hide navigation bar False: Show navigation bar	

example:

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

```
manager.hideNavBar(true);//Hidden navigation bar
```

4.6 Navigation bar status query

function: public boolean getNavBarHideState()

description: Whether the navigation bar is hidden

API version that implements this interface: V1.0-20180602

```
/**
 * @method getNavBarHideState()
 * @description Get the status of the navigation bar
 * @date 20180602
 * @author sky
 * @return Return true when the navigation bar is hidden, false when displayed
 */
```

Parameter name/return value	Types	Description	For example
return value	boolean	true:hide, false:display	

example:

```
MyManager manager =
```

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

```

if(manager.getNavBarHideState()){
    Log.e("StatusBar hide.");
}else{
    Log.e("StatusBar show.");
}

```

4.7 Set the slide-out navigation bar to turn on or off

function: public void setSlideShowNavBar(boolean flag)

description: Open or close the slide out navigation bar

API version that implements this interface: V1.0-20180602

```

/**
 * @method setSlideShowNavBar(boolean flag)
 * @description Open or close the slide out navigation bar
 * @date 20180602
 * @author sky
 * @param flag, Turn on the slide-out navigation bar to pass in true, turn off and pass in false
 */

```

Parameter name/return value	Types	Description	For example
-----------------------------	-------	-------------	-------------

parameter name	boolean	True: Open the slide-out status bar False: Turn off the status bar	
----------------	---------	---	--

example:

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

```
manager.setSlideShowNavBar(true);//Open the slide out navigation bar
```

4.8 Query whether the slide out navigation bar is open

function: public boolean isSlideShowNavBarOpen

description: Whether the slide out navigation bar is open

API version that implements this interface: V1.0-20180602

```
/**
```

```
 * @method isSlideShowNavBarOpen()
```

```
 * @description Query whether the navigation bar option is turned on
```

```
 * @date 20180602
```

```
 * @author sky
```

```
 * @return Slide out the navigation bar to open the return true, close the return false
```

```
*/
```

Parameter name/return value	Types	Description	For example
return value	boolean	true:Turn on, false:Turn off	

example:

```
MyManager manager =
```

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

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

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

```

}else{
    Log.e("SlideShowNavBar close");
}

```

4.9 Set the slide-out notification bar to turn on or off

function: public void setSlideShowNotificationBar(boolean enable)

Description: Turn on or off the slide out notification bar

API version that implements this interface: V1.0-20180602

```

/**
 * @method setSlideShowNotificationBar(boolean flag)
 * @description Settings Turn on or off the drop-down notification bar
 * @date 20180602
 * @author sky
 * @param flag, false Turn on the drop-down notification bar to pass true, and prohibit
  the drop-down notification bar from passing false
 */

```

Parameter name/return value	Types	Description	For example
parameter name	boolean	True: Open the slide-out status bar False: Turn off the status bar	

example:

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

```
manager.setSlideShowNavBar(true);//Open the drop-down notification bar
```

4.10 查询滑出通知栏状态

function: public boolean isSlideShowNotificationBarOpen()

description: 滑出通知栏是否打开

API version that implements this interface: V1.0-20180602

```
/**  
 * @method isSlideShowNotificationBarOpen()  
 * @description Query whether the drop-down notification bar is open  
 * @date 20180602  
 * @author sky  
 * @return Returns true when the notification bar is opened, otherwise it returns false  
 */
```

Parameter name/return value	Types	Description	For example
return value	boolean	true: Turn on, false: Turn off	

example:

```
MyManager manager =
```

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

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

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

```
}else{
```

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

```
}
```

4.11 Set screen brightness

function: public void changeScreenLight(int value)

Description: Adjust screen brightness

API version that implements this interface: V1.0-20180828

```
/**  
 * @method changeScreenLight(int value)  
 * @description Set screen brightness  
 * @date 20180828  
 * @author sky  
 * @param value, It is set from 1 to 100, setting 100 means maximum brightness, 1  
   means minimum brightness  
 */
```

Parameter name/return value	Types	Description	For example
parameter name	int	Specific value of brightness (1--100)	50

Note point: The parameter value is set from 1 to 100, setting 100 represents the maximum brightness, and 1 represents the minimum brightness

example: `manager.changeScreenLight(50);`

4.12 Turn off the backlight

function: `public void turnOffBacklight()`

description: Turn off the screen, only turn off the backlight, but do not enter sleep, the software continues to run. API version that implements this interface: V1.0-20180602

```
/**  
 * @method turnOffBackLight()  
 * @description Turn off the screen backlight, just turn off the backlight, other system  
   functions as usual  
 * @date 20180602  
 * @author sky
```

```
*/
```

example:

```
MyManager manager =
```

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

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

4.13 Turn on the backlight

function: public void turnOnBacklight()

description: Turn on the backlight.

API version that implements this interface: V1.0-20180602

```
/**
```

```
 * @method turnOnBackLight()
```

```
 * @description Turn on the screen backlight, corresponds to the turnOffBackLight() method
```

```
 * @date 20180602
```

```
 * @author sky
```

```
*/
```

example: manager.turnOnBacklight();

4.14 Get whether the backlight is on

function: public boolean isBacklightOn()

Description: Get whether the backlight is on

API version that implements this interface: V2.1-20181122

```
/**
```

```
 * @method isBacklightOn()
```

```
 * @description Get whether the backlight is on
```

```
 * @date 20181122
```

```
 * @author sky
```

* @return The current backlight is on, return true, otherwise return false

*/

Parameter name/return value	Types	Description	For example
return value	boolean	True means Turn on the backlight false means Turn off the backlight	

example:

```
MyManager manager =
```

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

```
manager.isBacklightOn();
```

4.15 Get backlight brightness value

function: public int getSystemBrightness()

description: Get backlight brightness value

API version that implements this interface: V2.1-20181122

/**

* @method getSystemBrightness()

* @description Get current device's backlight brightness value

* @date 20181122

* @author sky

* @return int, The returned brightness range is 0-100

*/

Parameter name/return value	Types	Description	For example
return value	int	The brightness range taken is from 0 to 100	

example:

```
MyManager manager =
```

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

4.16 Turn off HDMI output

function: public void turnOffHDMI()

description: Turn off HDMI output signal

API version that implements this interface: V1.0-20180602

```
/**  
 * @method turnOffHDMI()  
 * @description Turn off HDMI output, just turn off signal output, other system functions  
 * will not be affected  
 * @date 20180602  
 * @author sky  
 */
```

example:

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

4.17 Turn on HDMI output

function: public void turnOnHDMI()

description: Turn on the HDMI output signal

API version that implements this interface: V1.0-20180602

```
/**  
 * @method turnOnHDMI()  
 * @description Turn on HDMI output, corresponding to turnOffHDMI()  
 * @date 20180602
```

```
* @author sky
```

```
*/
```

example:

```
MyManager manager =
```

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

```
manager.turnOnBacklight();
```


Chapter 5 Installation and Upgrade

5.1 Firmware upgrade

function: public void upgradeSystem(String absolutePath)

description: Upgrade firmware

API version that implements this interface: V1.0-20180602

```
/**
 * @method upgradeSystem(String absolutePath)
 * @description Upgrade firmware
 * @date 20180602
 * @author sky
 * @param absolutePath, The absolute path of the firmware to be upgraded
 */
```

Parameter name/return value	Types	Description	For example
parameter name	String	The absolute path where the firmware is stored	/mnt/internal_sd/update.img

Note point:

- 1 function will automatically restart the system for the update.img upgrade.
- 2 Ensure that the img file exists and is completely usable.
- 3 The file name must be update.img or updat.zip.

example:

```
MyManager manager =
```

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

```
upgradeSystem("/mnt/internal_sd/update.img");
```

5.2 recovery mode

function: public void rebootRecovery()

description: Will reboot into recovery mode

API version that implements this interface: V1.0-20180602

```
/**
 * @method rebootRecovery()
 * @description Restore default setting
 * @date 20180602
 * @author sky
 */
```

example:

```
MyManager manager =
```

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

```
manager.rebootRecovery();
```

5.3 Silently install APK

function: public boolean silentInstallApk(String apkPath)

description: Install the APK application silently.

API version that implements this interface: V1.0-20180602

```
/**
 * @method silentInstallApk(String apkPath)
 * @description To install apk silently, the method used is pm install -r
 * @date 20180602
 * @author sky
 * @param apkPath, The absolute path of the apk to be installed
 * @return Return true if silent installation is successful, otherwise false
 */
```

Parameter name/return value	Types	Description	For example
parameter name	String	The absolute path of the APK to be installed	"/mnt/internal_sd/test.apk"
return value	Boolean	Whether the silent installation was successful	

Note point:

1. The APK file exists and is

fully usable.example:

//install test.apk

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

```
boolean success = manager.silentInstallApk("/mnt/usb/0000-0000/YiShengTest.apk"); ToastUtils.showToast(this, "Whether the silent upgrade is successful" + success);
```

5.4 Set whether the upgrade firmware pops up

function: public void setUpdateSystemWithDialog(boolean flag)

description: Set whether the upgrade firmware pops up

API version that implements this interface: V4.2-20200330

```
/**
 * @method setUpdateSystemWithDialog(boolean flag)
 * @description 设置升级固件是否弹窗, 默认是不弹窗Set whether the upgrade firmware pops up, the default is not to pop up
 * @date 20200330
 * @author sky
 * @param flag, true Pop-up window during upgrade, false does not pop up during upgrade
 */
```

Parameter name/return value	Type s	Description	For example
parameter name	boolean	Control whether the upgrade firmware pops up	True pops up false not pops up

example:

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

```
manager. setUpdateSystemWithDialog(true);//Confirmation of setting upgrade firmware pop-up window
```

Chapter 6 Network

6.1 Query Ethernet MAC address

function: public String getEthMacAddress()

description: Get the MAC address of the Ethernet

API version that implements this interface: V2.0-20180806

```
/**
```

```
 * @method getEthMacAddress()
```

```
 * @description Get the MAC address of the Ethernet
```

```
 * @date 20180806
```

```
 * @author sky
```

```
 * @return Return the Ethernet mac address, for example 30:1F:9A:61:BA:8F
```

```
*/
```

Parameter name/return value	Types	Description	For example
return value	String	Ethernet MAC address	30:1F:9A:61:BA:8F

example:

```
MyManager manager =
```

```
MyManager.getInstance(this); Log.e("ETH
```

```
MAC=" manager.getEthMacAddress ());
```

```
//output ETH MAC =30:1F:9A:61:BA:8F
```

6.2 Query Ethernet connection mode

function: public String getEthMode()

description: Get the current Ethernet mode

API version that implements this interface: V1.0-20180602

```

/**
 * @method getEthMode()
 * @description Get the mode of Ethernet, static or dynamic
 * @date 20180602
 * @author sky
 * @return Static mode returns StaticIp, dynamic mode returns DHCP
 */

```

Parameter name/return value	Types	Description	For example
return value	String	Ethernet mode	StaticIp(Static) DHCP(dynamic)

Note point:

① Before obtaining the Ethernet static IP, you need to bind the aidl service that obtains the Ethernet static IP, and bind it when creating the MyManager object. The method is as follows:
`manager.bindAIDLService(this);`

② Need to be called when the activity is destroyed `manager.unbindAIDLService(this);`

example:

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

```
Log.d("API display", "Current mode
```

```
="+manager.getEthMode());
```

```
//output Current mode = DHCP
```

6.3 Query Ethernet switch status

function: `public boolean getEthStatus()`

description: Get whether the current Ethernet switch is on API version that

implements this interface: V2.0-20180806

```

/**
 * @method getEthStatus()
 * @description Get Ethernet switch status
 * @date 20180806
 * @author sky
 * @return Return true when the Ethernet switch is turned on, and false when the switch is
        turned off
 */

```

Parameter name/return value	Types	Description	For example
return value	boolean	Ethernet switch status	True(Turn on) False(Turn off)

Note point:

① Before getting whether the Ethernet is turned on, you need to bind the aidl service, and bind it when creating the MyManager object. The method is as follows:
`manager.bindAIDLService(this);`

② Need to be called when the activity is destroyed `manager.unbindAIDLService(this);`

example:

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

```
Log.d("API display", "Whether the Ethernet is on =" + manager.getEthStatus());
```

```
//output Whether the Ethernet is on = false
```

6.4 Query Ethernet subnet mask

```
function: public String getNetMask ()
```

```
description: Query Ethernet subnet mask
```

```
API version that implements this interface: V2.0-20180806
```

```
/**
```

```
 * @method getNetMask()
```

```

* @description Query Ethernet subnet mask
* @date 20180806

* @author sky

* @return Return the current device Ethernet subnet mask, for example 255.255.255.0

*/

```

Parameter name/return value	Types	Description	For example
return value	String	Subnet mask	255.255.255.0

Note point:

① Before obtaining whether the Ethernet is turned on, you need to bind the aidl service, and bind it when creating the MyManager object. The method is as follows:
`manager.bindAIDLService(this);`

② Need to be called when the activity is destroyed `manager.unbindAIDLService(this);`

example:

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

```
Log.d("API display","Ethernet subnet mask =" +manager.getNetMask());
```

```
//output Ethernet subnet mask = 255.255.255.0
```

6.5 Query Ethernet gateway

```
function: public String getGateway ()
```

```
description: Get current Ethernet gateway
```

```
API version that implements this interface: V2.0-20180806
```

```
/**
```

```
* @method getGateway()
```

```
* @description Get Ethernet gateway
```

```
* @date 20180806
```


* @author sky

* @return Return the Ethernet gateway of the current device, for example 192.168.1.1

*/

Parameter name/return value	Types	Description	For example
return value	String	gateway	192.168.1.1

Note point:

① Before obtaining whether the Ethernet is turned on, you need to bind the aidl service, and bind it when creating the MyManager object. The method is as follows:
manager.bindAIDLService(this);

② Need to be called when the activity is destroyed manager.unbindAIDLService(this);
example:

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

```
Log.d("API display", "Ethernet gateway
```

```
="+manager.getGateway());
```

```
//output Ethernet gateway = 192.168.1.1
```

6.6 Query Ethernet Dns1

```
function: public String getEthDns1 ()
```

```
description: Query Ethernet Dns1
```

```
API version that implements this interface: V2.0-20180806
```

```
/**
```

```
* @method getEthDns1()
```

```
* @description Query Ethernet Dns1
```

```
* @date 20180806
```

```
* @author sky
```

```
* @return Return the DNS1 of the current device Ethernet, for example 192.168.1.1
*/
```

Parameter name/return value	Types	Description	For example
return value	String	Dns 1	192.168.1.1

Note point:

① Before obtaining whether the Ethernet is turned on, you need to bind the aidl service, and bind it when creating the MyManager object. The method is as follows:

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

② Need to be called when the activity is destroyed `manager.unbindAIDLService(this);`

example:

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

```
Log.d("API display", "Ethernet dns1
```

```
="+manager.getGateway());
```

```
//output Ethernet dns1 = 192.168.1.1
```

6.7 Query Ethernet Dns2

```
function: public String getEthDns2()
```

```
description: Get the current Ethernet Dns2
```

```
API version that implements this interface: V2.0-20180806
```

```
/**
```

```
* @method getEthDns2()
```

```
* @description Get the Ethernet Dns2
```

```
* @date 20180806
```

```
* @author sky
```

```
* @return Return the DNS2 of the current device Ethernet, for example 192.168.1.1
```

```
*/
```

Parameter name/return value	Types	Description	For example
return value	String	Dns 2	192.168.1.1

Note point:

① Before obtaining whether the Ethernet is turned on, you need to bind the aidl service, and bind it when creating the MyManager object. The method is as follows:

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

② Need to be called when the activity is destroyed `manager.unbindAIDLService(this);`

example:

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

```
Log.d("API display", "Dns2 for ethernet
```

```
="+manager.getGateway());
```

```
//output Dns2 for ethernet = 192.168.1.1
```

6.8 Set Ethernet to dynamic acquisition mode

function: `public void setDhcpIpAddress(Context context)`

description: Set Ethernet to dynamic acquisition mode

API version that implements this interface: V1.0-20180602

```
/**
 * @method setDhcpIpAddress(Context context)
 * @description Set the Ethernet mode to dynamic
 * @date 20180602
 * @author sky
 * @param context, Context object
 */
```

Parameter name/return value	Types	Description	For example
-----------------------------	-------	-------------	-------------

parameter name	Context	Context object	
----------------	---------	----------------	--

example:

MyManager manager =

MyManager.getInstance(this);

manager.setDhcpIpAddress(this);

6.9 Set Ethernet to static address mode

function: public void setStaticEthIpAddress(String IPAddr, String gateWay,String mask, String dns1, String dns2)

description: Set Ethernet to static address mode

API version that implements this interface: V1.0-

20180602

/**

* @method setStaticEthIpAddress(String IPAddr, String gateWay, String mask, String dns1, String dns2)

* @description Set the Ethernet mode to static and specify related parameters

* @date 20180602

* @author sky

* @param IPAddr, The set IP address. gateWay, Set the gateway.mask, Set the subnet mask.

dns1 and dns2 are the set dns values

*/

Parameter name/return value	Types	Description	For example
IPAddr	String	IP address	"192.168.1.125"
gateWay	String	gateway	"192.168.1.1"

mask	String	Subnet mask	"255.255.255.0"
dns1	String	DNS	"192.168.1.1"
ipLength	String	DNS	"0.0.0.0"

Note point: The gateway, subnet mask and DNS must be set as legal data.

example:

```
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 Get Ethernet dynamic IP address

function: public String getDhcpIpAddress()

description: Get Ethernet dynamic IP address

API version that implements this interface: V1.0-20180602

```
/**
 * @method getDhcpIpAddress()
 * @description Get Ethernet dynamic IP address
 * @date 20180602
 * @author sky
 * @return Return Ethernet dynamic ip address
 */
```

Parameter name/return value	Type s	Description	For example
return value	String	Ethernet dynamic IP address	192.168.1.100

example:

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

```
Log.d("API display", "ETH IP
="+myManager.getDhcpIpAddress());
//output ETH IP =192.168.1.103
```

6.11 Query the static IP address of the Ethernet

```
function: public String getStaticEthIPAddress()
```

```
description: Get the static IP address of the Ethernet
```

```
API version that implements this interface: V1.0-20180602
```

```
/**
 * @method getStaticEthIPAddress()
 * @description Get the static IP address of the Ethernet
 * @date 20180602
 * @author sky
 * @return Return to the static IP address of the Ethernet
 */
```

Parameter name/return value	Type s	Description	For example
return value	String	static IP address of the Ethernet	192.168.1.100

Note point:

①. Before obtaining the Ethernet static IP, you need to bind the aidl service that obtains the Ethernet static IP. When creating the myManager object, bind it as follows:
myManager.bindAIDLService(this);(this);

②Need to be called when the activity is destroyed myManager.unbindAIDLService(this);

example:

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

```
Log.d("API display ", "ETH IP
```

```
="+manager.getStaticEthIPAddress());
```

```
//output ETH IP =192.168.1.103
```

6.12 Control Ethernet switch

function: public void ethEnabled(boolean enable)

description: Switch Ethernet

API version that implements this interface: V1.0-20180602

```
/**  
 * @method ethEnabled(boolean enable)  
 * @description Control Ethernet switch  
 * @date 20180602  
 * @author sky  
 * @param enable, Turn on the Ethernet switch and pass in true, turn off the Ethernet switch  
 and pass in false  
 */
```

Parameter name/return value	Type s	Description	For example
return value	boolean	Switch Ethernet	true turn on false Turn off

example:

```
MyManager manager =
```

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

```
manager.ethEnabled(false);// Turn off Ethernet
```

6.13 pppoe dial-up internet

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

description: Set pppoe dial-up internet

API version that implements this interface: V4.2-20200330

Parameter name/return value	Types	Description	For example
parameter name	Context	Context object	
parameter name	String	Dial-up user name	
parameter name	String	Password for dial-up internet	

example:

```
MyManager manager =
```

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

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


Chapter 7 Storage

7.1 Query external storage SD card path

function: public String getSDcardPath()

description: Get external storage SD card path

API version that implements this interface: V1.0-20180602

```
/**  
 * @method getSDcardPath()  
 * @description Get the external SD card path  
 * @date 20180602  
 * @author sky  
 * @return Return to the external SD card path  
 */
```

Parameter name/return value	Types	Description	For example
return value	String	Absolute path of external storage SD card	

example:

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

```
Log.d("API test", "SD="+manager.getSDcardPath());
```

```
//output API: External SD card path = /mnt/external_sd
```

7.2 Query the path of the external storage U disk

function: public String getUSBStoragePath(int num)

description: Get the path of the external storage U disk

API version that implements this interface: V1.0-20180602

```

/**
 * @method getUSBStoragePath(int num)
 * @description Get the path of the external U disk
 * @date 20180602
 * @author sky
 * @param num, Start from 0, which means the U disk
 * @return the path of the u disk
 */

```

Parameter name/return value	Types	Description	For example
num	int	U Disk name	0,1,2,3
return value	String	Absolute path of external storage U disk	

Note point:

You need to set the path in the source code to run Demo normally

example:

```

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

```

7.3 Query the absolute path of the serial port

function: public String getUartPath(String uart)

description: Get the absolute path of the serial port.

API version that implements this interface: V1.0-20180602

```

/**
 * @method getUartPath(String uart)
 * @description Get the absolute path of the serial port

```

* @date 20180602

* @author sky

* @param uart, Enter the serial number of the serial port, such as serial port 0, enter TTYS0

* @return Return the absolute path of the specified serial port

*/

Parameter name/return value	Types	Description	For example
Uart	String	Port number corresponding to the serial port- uart 0, uart 1, uart 2, uart 3	
return value	String	UART Absolute path	

example:

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

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

```
//output UART-1=/dev/ttyS1
```

Chapter 8 Timer Switch

8.1 Set the timer switch of the week mode

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

description: Set timer switch, weekly mode, only one set of switch time per day.

API version that implements this interface: V4.0-20200115

```
/**  
 * @method setPowerOnOffWithWeekly(int[] powerOnTime,int[]  
 powerOffTime,int[] weekdays)  
 * @description Weekly mode setting time switch machine, only one set of switch  
 machine time a day  
 * @date 20200113  
 * @author sky  
 * @param powerOnTime, Boot time, for example {8,30}. powerOffTime, Shutdown time, for  
 example{18,30}.  
 * weekdays, Working status from Monday to Sunday, 1 means power on, 0  
 means no power on. For example, {1,1,1,1,1,0,0} means to switch the machine from  
 Monday to Friday  
 */
```

Parameter name/return value	Types	Description	For example
powerOnTime	Int[]	Boot time, Hour and minute	{8,30}
powerOffTime	Int[]	Shutdown time, Hour and minute	{18,30}
weekdays	Int[]	Working status from Monday to Sunday	{1,1,1,1,0,0,1}

Note point:

In this method, only one set of time can be set in a day, and the boot time is before and the shutdown time is after.

example:

```
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}; //Working status from Monday to Sunday, 1 means booting, 0 means not booting
```

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

Setting the above time will start at 8:30 and shut down at 18:30 from Monday to Friday.

8.2 Set up a set of timing switches

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

description: Set a set of time switch machine, can be set multiple times a

day, need to input the year, month, day, hour and minute

API version that implements this interface: V4.0-20200115

```
/**
```

```
 * @method setPowerOnOff(int[] powerOnTime,int[] powerOffTime)
```

```
 * @description Set a set of time data for timing switch machine, need to pass in year, month, day, hour and minute
```

```
 * @date 20200113
```

```
 * @author sky
```

```
 * @param powerOnTime, boot time, for example{2020,1,10,20,48}, powerOffTime, shutdown time, for example{2020,1,10,20,38}.
```

```
*/
```

Parameter name/return value	Types	Description	For example
powerOnTime	Int[]	Boot time, year, month, day, hour and minute	{2020,01,13,18,40}

powerOffTime	Int[]	Shutdown time, year month day hour minute	{2020,01,13,18,30}
--------------	--------	---	--------------------

Note point:

When this method is set, the shutdown time is

first, and the boot time is later

example:

```
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,timeoffArr
```

```
ay);
```

Setting the above time will be on January 10, 2018, shutdown at 20:38, and boot at 20:48

Note: This method is also suitable for only setting the power-on time, just pass the power-off time to 0, that is

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

8.3 Get the current timing switch mode

function: public String getPowerOnMode()

description: Get timing switch mode

API version that implements this interface: V4.0-20200115

```
/**
```

```
 * @method getPowerOnMode()
```

```
 * @description Get timing switch mode
```

```
 * @date 20200113
```

```
 * @author sky
```

```
 * @return "0"refers to the switch time set locally at the timer switch. "2"refers to calling the setPowerOnOffWithWeekly method by broadcasting. "1"refers to calling the setPowerOnOff method by broadcasting.
```

```
*/
```

Parameter name/return value	Types	Description	For example
return value	String	Switch mode	"0"

example:

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

8.4 Get the current boot time of the timing switch

function: public String getPowerOnTime()

description: Get the boot time of the timing switch

API version that implements this interface: V4.0-20200115

```
/**
 * @method getPowerOnTime()
 * @description Get the boot time of the current device
 * @date 20200113
 * @author sky
 * @return Return the currently set boot time, for example 202001132025, means boot on 13th
Jan of 2020 20:25.
 */
```

Parameter name/return value	Types	Description	For example
return value	String	Timer switch boot time	202001132025

example:

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

8.5 Get the shutdown time of the current timing switch

function: public String getPowerOffTime()

description: Get the time of the timer switch off

API version that implements this interface: V4.0-20200115

```
/**  
 * @method getPowerOffTime()  
 * @description Get the shutdown time of the current device  
 * @date 20200113  
 * @author sky  
 * @return Return the currently set boot time, for example 202001132020, mean shutdown  
 on13th Jan of 2020 20:20.  
 */
```

Parameter name/return value	Types	Description	For example
return value	String	Time switch off time	202001132020

example:

```
MyManager manager =
```

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

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

8.6 Get the last boot time of the timer switch for the device.

function: public String getLastestPowerOnTime()

description: Get the last boot time of the device

API version that implements this interface: V4.0-20200115

```
/**  
 * @method getLastestPowerOnTime()  
 * @description Get the last boot time of the device
```


* @date 20200113

* @author sky

* @return Return to the last boot time of the device, for example 202001132025, refers to the boot operation performed at 20:25 on January 13, 2020

*/

Parameter name/return value	Types	Description	For example
return value	String	Last boot time of the timer switch	202001132025

example:

```
MyManager manager =
```

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

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

8.7 Get the last shutdown time of the timer switch for the device

function: public String getLastestPowerOffTime() description: Get the

last shutdown time of the device API version that implements this

interface: V4.0-20200115

/**

* @method getLastestPowerOffTime()

* @description Get the last shutdown time of the device

* @date 20200113

* @author sky

@return Returns the last boot time of the device, for example 202001132020, means that the boot operation was performed at 20:20 on January 13, 2020

*/

Parameter name/return value	Types	Description	For example
return value	String	Last shutdown time of the timer switch	202001132020

example:

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

8.8 Clear time switch data

function: public void clearPowerOnOffTime()

description: Clear timing switch time

API version that implements this interface: V4.0-20200115

```
/**
 * @method clearPowerOnOffTime()
 * @description Clear timing switch time
 * @date 20200113
 * @author sky
 */
```

example:

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

8.9 Get the timing switch version

function: public String

getVersion() description: Get the version number of the timer switch

API version that implements this interface: V4.0-20200115

Parameter name/return value	Types	Description	For example
return value	String	Time switch version	YS_1.0_20180312

example:

```
MyManager manager =
```

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

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

Chapter 9 GPIO Index Value Control

9.1 Apply permission for export

function: public void upgradeRootPermissionForExport ()

description: Apply permission for export, the operation that must be performed before gpio index value operation.

API version that implements this interface: V4.1-20200213

```
/**  
 * @method upgradeRootPermissionForExport()  
 * @description Apply for export permission, it is recommended to execute it when the  
 boot broadcast or the application starts  
 * @date 20200213  
 * @author sky  
 */
```

example:

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

9.2 Configure the gpio path according to the specific index value

function: public boolean exportGpio(int gpio) description:

Configure the gpio path, what must be done API version that implements this interface: V4.1-20200213

```
/**  
 * @method exportGpio(int gpio)  
 @description Configure the export path according to the specific index value, and it is  
 recommended to execute it when the boot broadcast or the application starts  
 * @date 20200213
```

* @author sky

* @return Return true if configuration is successful, false if configuration fails

*/

Parameter name/return value	Types	Description	For example
return value	Boolean	Whether the configuration export path is successful	success-true Fail-false
parameter name	int	Index value	136

example:

```
MyManager manager =
```

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

```
manager.exportGpio(136);
```

9.3 Apply permission for the configured gpio path

function: public void upgradeRootPermissionForGpio(int gpio)

description: To apply for permission for the configured gpio path, the necessary operations are performed after the 9.2 method

API version that implements this interface: V4.1-20200213

/**

* @method upgradeRootPermissionForGpio(int gpio)

* @description Apply permission for the configured gpio path, it is recommended to execute it when the boot broadcast or the application starts, after the exportGpio(int gpio) method

* @date 20200213

* @author sky

*/

Parameter name/return	Types	Description	For example
-----------------------	-------	-------------	-------------

value			
parameter name	int	Index value	136

example:

```
MyManager manager =
```

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

```
manager.upgradeRootPermissionForGpio(136);
```

9.4 Set the state of the io port as input or output

function: public boolean setGpioDirection(int gpio, int arg) description: Set whether the gpio is input or output according to the specific index value

API version that implements this interface: V4.1-20200213

```
/**
 * @method setGpioDirection(int gpio, int arg)
 * @description Set the state of the io port as input or output
 * @date 20200213
 * @author sky
 * @param gpio, the gpio index value need to be operated. arg, 1 refer setting to
input port, 0 refer estting to output port.
 * @return Return true if successful, false if failed
 */
```

Parameter name/return value	Types	Description	For example
return value	Boolean	Whether the setting is successful	Success -true Fail-false
Parameter value	Gpio	Gpio index value	136

Parameter value	Arg	Input or output	Input port-1 output port-0
-----------------	-----	-----------------	-------------------------------

example:

MyManager manager =

MyManager.getInstance(this); manager.

setGpioDirection(136, 0);

9.5 Get the current gpio is input port or output port

function: public String getGpioDirection(int gpio)

description: According to the specific gpio index value, get the current gpio state is input or output. This method can also determine whether the gpio is valid, if there is no return value, it means that the gpio is not available

API version that implements this interface: V4.1-20200213

/**

* @method getGpioDirection(int gpio)

* @description According to the specific gpio index value to get the current gpio status is input or output

* @date 20200213

* @author sky

* @param gpio, gpio index value

* @return input port return in, output port return out

*/

Parameter name/return value	Types	Description	For example
return value	String	Input or output	input-in output-out
Parameter value	Int	Gpio index value	136

example:

MyManager manager =

MyManager.getInstance(this);

```
manager.getGpioDirection(136);
```

9.6 Set gpio level

```
function: public boolean writeGpioValue(int gpio, String arg)
```

description: Control the level of the gpio according to the specific gpio index value

API version that implements this interface: V4.1-20200213

```
/**  
 * @method writeGpioValue(int gpio, String arg)  
 * @description control gpio level, only for output port  
 * @date 20200213  
 * @author sky  
 * @param gpio, gpio index value. Arg, high level--1, low level--0  
 * @return Return true if writing is successful, false if it fails  
 */
```

Parameter name/return value	Types	Description	For example
return value	Boolean	Whether it writes successful	successful--true fail--false
parameter name	Int	Gpio index value	136
parameter name	String	High and low level	high--1 low--0

example:

```
MyManager manager =
```

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

```
manager.writeGpioValue(136, 1);
```


9.7 Get the current gpio level

function: public String getGpioValue(int gpio)

description: Get the current gpio level

API version that implements this interface: V4.1-20200213

```
/**  
    * @method getGpioValue(int gpio)  
    * @description Get the current gpio level  
* @date 20200213  
  
    * @author sky  
  
    * @param gpio, gpio 的index value  
  
    * @return High level returns 1, low level returns 0  
  
*/
```

Parameter name/return value	Types	Description	For example
return value	String	High level or low level	High level--1 Low level--0
parameter name	Int	Gpio index value	136

example:

```
MyManager manager =
```

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

```
manager.getGpioValue(136);
```

Chapter 10 Other

10.1 Set and save system time

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

description: Set and save system time

API version that implements this interface: V1.0-20180602

```
/**  
 * @method setTime(int year, int month, int day, int hour, int minute, int sec)  
 * @description Set system time  
 * @date 20180602  
 * @author sky  
 * @param Enter the year, month, day, hour, minute, and second to be set, where the  
 month is from 1-12, and the time is in 24-hour format  
 */
```

Parameter name/return value	Types	Description	For example
year	int	Year	
month	int	Month (1 to 12)	
day	int	day	
hour	int	(24-hour format) hour	
minute	int	(24-hour format) minute	
return value	void		

example:

```
//Set 2016-03-16, 13:44:00
```

```
MyManager manager =
```

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

(2016, 3, 16, 13, 44);

10.2 Turn on or off automatic time determination

function: public void switchAutoTime(boolean open)

description: Set to turn on or off automatic time determination

API version that implements this interface: V1.0-

20180602

```
/**
 * @method switchAutoTime(boolean open)
 * @description Control the switch that automatically determines the date and time
 * @date 20180602
 * @author sky
 * @param open, Enter true to turn on the switch, false to turn off the switch
 */
```

Parameter name/return value	Type s	Description	For example
parameter name	boolean	True Turn on automatic time determination False Turn off automatic time determination	

example:

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

```
manager.switchAutoTime(false);//Turn off automatic date and time determination
```

10.3 Control whether the soft keyboard can pop up

function: public void setSoftKeyboardHidden(boolean hidden)

description: Control whether the soft keyboard can pop up

API version that implements this interface: V1.0-20181126

```
/**
```

```
* @method setSoftKeyboardHidden(boolean hidden)
```

```
* @description Control whether the soft keyboard can pop up(Mainly EditText control)
```

```
* @date 20181126
```

```
* @author sky
```

```
* @param hidden, Enter true to hide the soft keyboard, and false to show the soft keyboard
```

```
*/
```

Parameter name/return value	Types	Description	For example
parameter name	boolean	True Soft keyboard can pop up False The soft keyboard cannot pop up	

example:

```
MyManager manager =
```

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

```
manager.setSoftKeyboardHidden(false);
```

10.4 Sleep time control

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

description: Sleep time control

API version that implements this interface: V2.2-20181129

```
/**
```

```
 * @method setDormantInterval(Context context, long time)
```

```
 * @description Sleep time control
```

```
 * @date 20181129
```

```
 * @author sky
```

```
 * @param context, Context object. time, Interval time (no operating system enters sleep within the interval)
```

```
*/
```

Parameter name/return value	Types	Description	For example
return value	long	Interval time (if sleep is turned off, the value passed is 2147483647)	If the sleep interval 15s, the parameters are passed 15,000

example:

```

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

```

10.5 Query automatically determine the network time status

function: public boolean isAutoSyncTime()

description: Get whether the switch to automatically determine the network time is on

API version that implements this interface: V2.0-20180806

```

/**
 * @method isAutoSyncTime()
 * @description Get the switch status that automatically determines the network time
 * @date 20180806
 * @author sky
 * @return Return true to indicate that the automatic network time determination is
 *         enabled, and false to indicate that the switch is turned off
 */

```

Parameter name/return value	Type s	Description	For example
return value	boolean	True Turn on automatic time determination False Turn off automatic time determination	

example:

```

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

```

10.6 Run shell commands with ROOT privileges

function: public void execSuCmd (String command)

description: Will run the shell command with ROOT authority

version that implements this interface: V1.0-20180602

/**

* @method execSuCmd(String command)

* @description Run shell commands under root authority

* @date 20180602

* @author sky

* @param command, Pass in the shell command that needs to be executed, such as
reboot

*/

Parameter name/return value	Types	Description	For example
command	String	Shell command	"reboot"
return value	void		

example:

```
//View the directory and install the APK with shell commands
```

```
MyManager manager =
```

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

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

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

10.7 Query device's Internet access method

function: public int getCurrentNetType()

description: Get the current Internet type

API version that implements this interface: V1.0-20180602

```
/**
```

```
* @method getCurrentNetType()
```

```
* @description Get the current Internet type
```

```
* @date 20180602
```

```
* @author sky
```

```
* @return Returns an int value, 0 means Ethernet, 1 means WIFI, 2 means mobile network, -100 means unknown network
```

```
*/
```


Parameter name/return value	Types	Description	For example
return value	int	0 Ethernet, 1 wifi, 2 mobile network -100 unknown network	

example:

```
MyManager manager =
```

```
MyManager.getInstance(this); Int type =
```

```
manager.getCurrentNetType();
```

10.8 Query HDMI input status

```
function: public int getHdmiinStatus()
```

description: HDMI input status

API version that implements this interface: V1.0-20180602

```
/**
```

```
 * @method getHdmiinStatus()
```

```
 * @description Get HDMI connection status
```

```
 * @date 20180602
```

```
 * @author sky
```

```
 * @return HDMI has output return 1, otherwise return 0
```

```
*/
```

Parameter name/return value	Types	Description	For example
return value	int	Return 0 no HDMI input, 1 have HDMI input	

example:

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

```
Toast.makeText(getApplicationContext(),"HDMI" +
```

```
manager.getHdmiinStatus(),Toast.LENGTH_SHORT).show();
```

10.9 Query the current number of screens

```
function: public int getScreenNumber()
```

description: Query the current number of screens

API version that implements this interface: V1.0-20180602

```
/**  
 * @method getScreenNumber()  
 * @description Get the number of screens connected to the current device  
 * @date 20180602  
 * @author sky  
 * @return the device will return the number when it connect how many screens  
 */
```

Parameter name/return value	Types	Description	For example
return value	int	Default return 1, return 2 when it connect with two screens	

example:

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

```
Toast.makeText(getApplicationContext(), " numbers of
screens " +
manager.getScreenNumber(),Toast.LENGTH_SHORT).s
how();
```

10.10 Query Android layer log

function: public void getAndroidLogcat(String path)

description: Query Android layer log

API version that implements this interface: V1.0-20180602

```
/**
 * @method getAndroidLogcat(String path)
 * @description Get Android layer log
 * @date 20180602
 * @author sky
 * @param path, Path to save logs
 */
```

Parameter name/return value	Types	Description	For example
parameter name	String	Path to save logs	"/mnt/sdcard/logcat.txt"

Note

point:

example:

MyManager manager =

MyManager.getInstance(this);

manager.getAndroidLogcat("/mnt/sdcard/logcat.t

xt")

10.11 Get the Kernel layer log

function: public void getKmsgLog(String path)

description: Get the Kernel layer log

API version that implements this interface: V1.0-20180602

```
/**
 * @method getKmsgLog(String path)
 * @description Get the Kernel layer log
 * @date 20180602
 * @author sky
 * @param path, Path to save logs
 */
```

Parameter name/return value	Type s	Description	For example
parameter name	String	Path to save logs	

example:

```
MyManager manager =
```

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

```
manager.getKmsgLog("/mnt/sdcard/logcat.txt")
```

10.12 Set default input method

function: public boolean isSetDefaultInputMethodSuccess(String defaultInputMethod)

description: Set default input method

API version that implements this interface: V3.1-20190228

```
/**
 * @method isSetDefaultInputMethodSuccess(String defaultInputMethod)
```

* @description Set default input method, and determine whether the setting is successful

* @date 20190228

* @author sky

* @param defaultInputMethod, The package name of the input method that needs to be set, such as Google Pinyin input method "com.google.android.inputmethod.pinyin/.PinyinIME"

* @return Return true if successful, false if failed

*/

Parameter name/return value	Types	Description	For example
parameter name	String	Default input method apk package name	Google Pinyin com.google.android.inputmethod.pinyin/.PinyinIME
return value	Boolean	Whether the setting is successful	Return true on success, false on failure

example:

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

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

10.13 Get the default input method

```
function: public String getDefaultInputMethod()
```

description: Get the default input method

API version that implements this interface: V3.1-20190228

```
/**
```

```
* @method getDefaultInputMethod()
```

```

* @description Get the current system input method
* @date 20190228

* @author sky
* @return Returns the package name of the current input method, for example
com.google.android.inputmethod.pinyin/.PinyinIME

*/

```

Parameter name/return value	Types	Description	For example
parameter name	String	current input method apk package name	Google PINYIN com.google.android.inputmethod.pinyin

example:

```

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

```

10.14 Set system language

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

description: Set system language

API version that implements this interface: V3.2-20190513

```

/**

* @method setLanguage(String language, String country)
* @description Set the current system language
* @date 20190513
* @author sky
* @param language, language, like zh。 country, like CN

```

*/

Parameter name/return value	Types	Description	For example
parameter name	String	Set language (lowercase letters) and country (uppercase letters)	Simplified Chinese ch CN Chinese Taiwan zh TW Canada English en CA

example:

```
MyManager manager =
```

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

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

10.15 Get device CPU temperature

function: public String getCPUTemperature()

description: Get device CPU temperature

API version that implements this interface: V3.3-20190619

/**

* @method getCPUTemperature()

* @description Get the current system CPU temperature

* @date 20190619

* @author sky

* @return Returns an int value, the unit is Celsius

*/

Parameter name/return value	Types	Description	For example
-----------------------------	-------	-------------	-------------

parameter name	String	Device CPU temperature	
----------------	--------	------------------------	--

example:

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

```
Log.d("API display","Device CPU temperature: "+ manager.getCPUTemperature());
```

10.16 Turn on or off adb connection

function: public void setADBOpen(boolean open)

description: Tur on or off adb

API version that implements this interface: V3.4-20190628

```
/**
 * @method setADBOpen(boolean open)
 * @description Turn on or off adb connection
 * @date 20190628
 * @author sky
 * @param open, true, tum on adb connection switch。 false, Turn off adb connection
 */
```

Parameter name/return value	Types	Description	For example
parameter	Boolea n	Whether turn on adb, true turn on, false turn off	

example:

```
MyManager manager =
```

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

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

10.17 Replace boot animation

function: public void replaceBootanimation(String path)

description: Replace boot animation

API version that implements this interface: V3.4-20190628

```
/**  
 * @method replaceBootanimation(String path)  
 * @description Replace boot animation  
 * @date 20190628  
 * @author sky  
 * @param path, Boot animation the absolute path where bootanimation.zip is located  
 */
```

Parameter name/return value	Types	Description	For example
parameter	String	Path of boot animation	/storage/emulated/0/bootanimation.zip

example:

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

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

10.18 Set default Launcher

function: public void setDefaultLauncher(String packageAndClassName)

description: Launcher If the system has more than one Launcher, it can set the default Launcher

API version that implements this interface: V3.5-20190719

```
/**  
 * @method setDefaultLauncher(String packageAndClassName)  
 * @description Set system default Launcher  
 * @date 20190719
```

* @author sky

* @param packageAndClassName, 设置的 Launcher 的包名和启动类名, 例如
"com.android.launcher3/com.android.launcher3.Launcher"

*/

Parameter name/return value	Types	Description	For example
parameter	String	Default Launcher package name and the start class name	Launcher3 package name and the start class name "com.android.launcher3/com.android.launcher3.Launcher"

example:

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

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