SMDT Android API File

1. Version and Modification Records

Version	Description	Date
V1.0	Creation	2015-11-06
V1.1	The instructions of modifying some functions	2016-02-19
V1.2	Add some interfaces	2016-03-15
V1.3	Uniform modification	2016-03-21
V1.4	Add the setting of static IP	2016-05-24
V1.5	Increase the status value of the current screen switches	2016-07-05
V1.6	The modification according to R18 test	2016-07-26
V1.7	Add API method of firmware upgrade	2016-08-18
V1.8	Add the method of setting the lightness interface	2016-11-08
V1.9	Add the method of setting the mic switches interface	2017-05-05
V2.0	A83 add the uninstall interface of SD cards/U disk interface	2017-06-22
V2.1	Increase to get dual screen display screen_number value	2017-06-22
V2.2	Modify the reset usage method of smdtsetcontrol operation Add an interface to get HDMI IN status	2017-07-02
V2.3	Increase the interface to set the GPIO direction	2018-02-05
V2.4	Add interface to switch HDMIIN	2018-05-08
V2.5	Unified management of each platform API	2019-07-11
V2.6	Solve the problem that some APIs require system permission Add EDP screen switch	2019-12-18
V2.7	Add CPU usage, install APP callback, etc.	2020-02-28

2. Copyright Notice

This manual is copyrighted by Shenzhen Smart Device Technology Co., Ltd and reserves all rights. No entity or individual may extract part or all of this manual without the consent (in writing form) of SMDT. Any violation of this manual will be prosecuted for legal liability.

Attention：

The manuals for the products to be sold are updated frequently. Please download the latest manual at http://www.smdt.com.cn. We will not have any further notice.

3. Technical Support

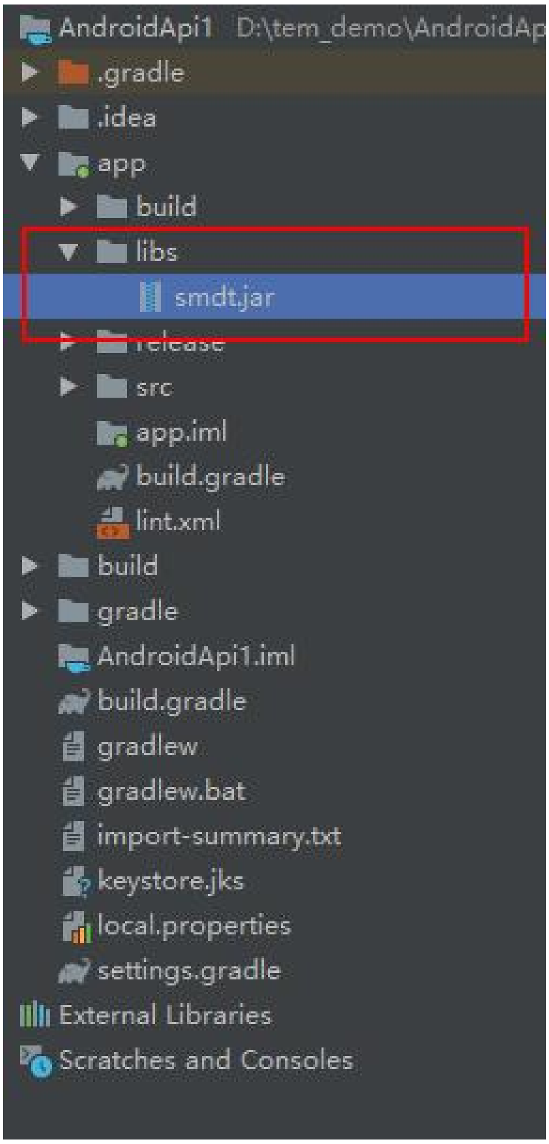
If you have questions about the documents, you can call the technical supporters during office hours (Monday to Friday 9:00 to 12:00; 1:30 to 6:00 pm; Saturday 9:00 to 12:00) or contact us by E-mail.

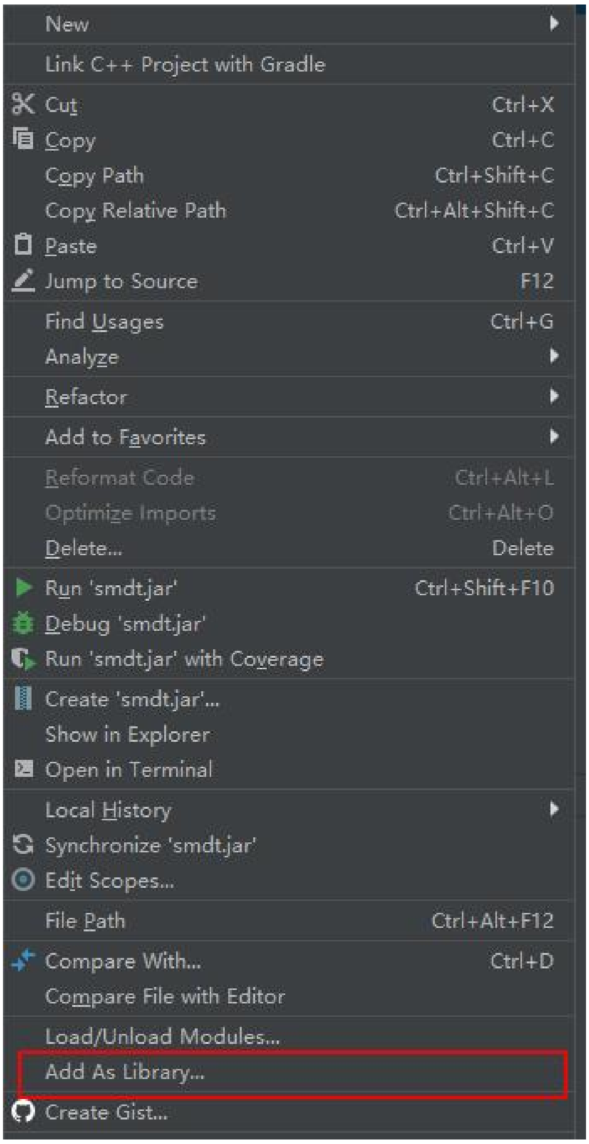
Web：www.smdt.com.cn
Telephone： 0755-6166 2980–8899, 8888
Development community： www.armboard.org
Technical support QQ group：206823686
E-mail： sales@smdt.com.cn

Contents

4. How to use the SMDT API in Android Studio

Copy smdt.jar to [Project directory \app\libs\];
Right-click on the JAR file in the LIBS folder and select *add as Library...*





Start using the SMDT API

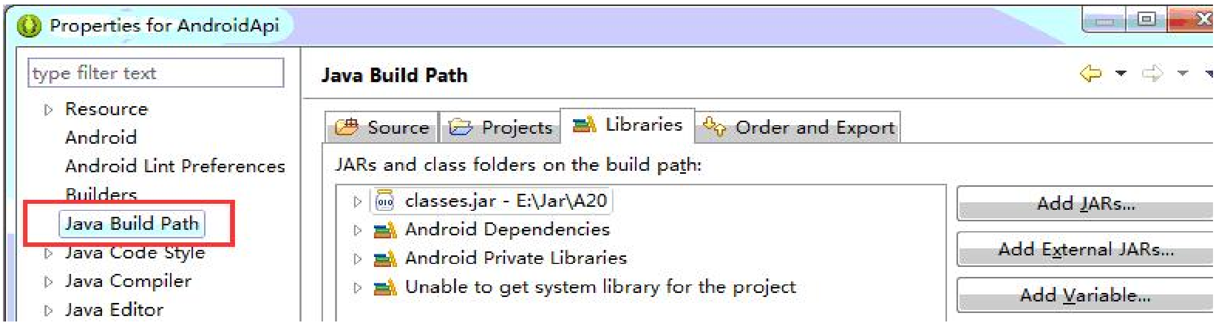
First declare the SmdtManager object, then you can start using the API. Take the following example:

```
    //Declare the SmdtManager object
    private SmdtManager smdt;
smdt = SmdtManager.create(this);
//Use API
smdt.smdtWatchDogEnable((char) 1);
```

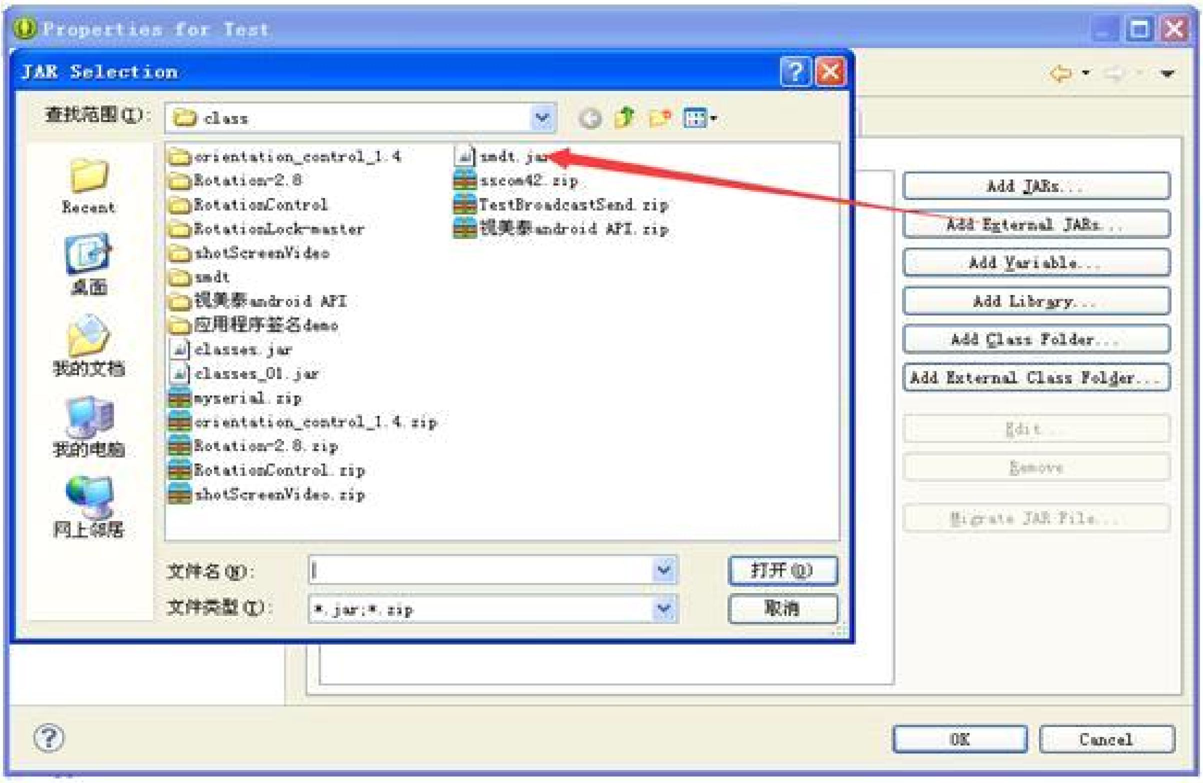
5. How to use SMDT API in eclipse

1. Open eclipse, select the project, click the right mouse button, select properties

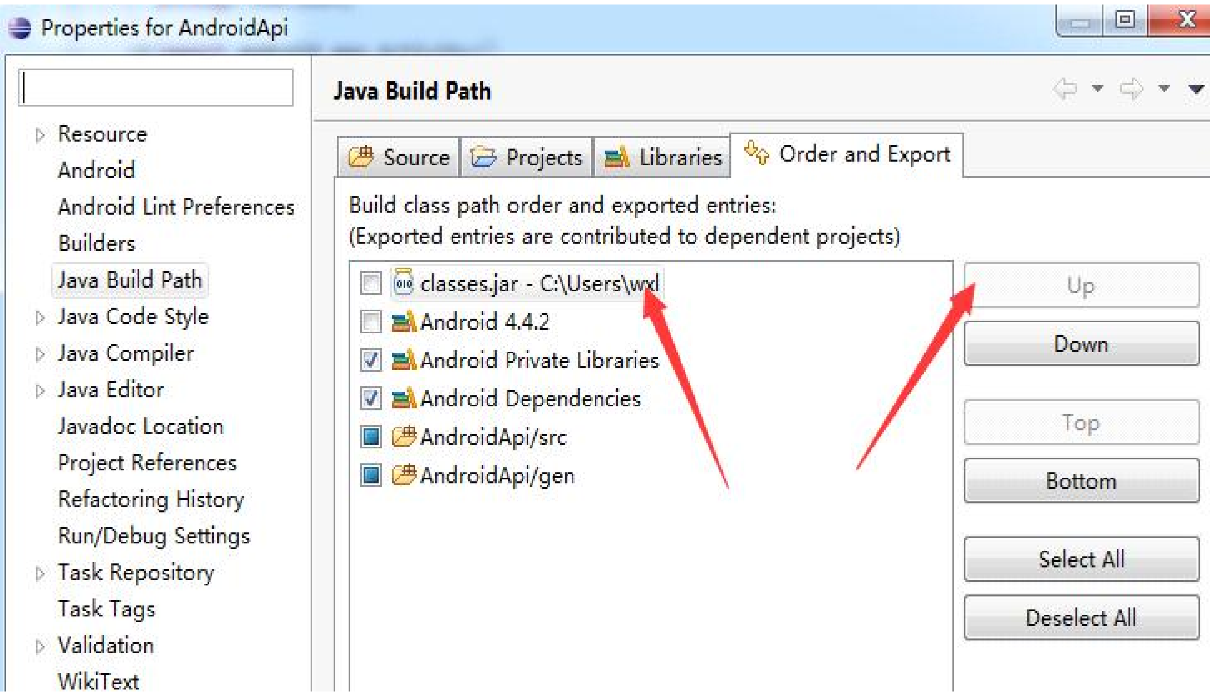
2. Select "Java Build Path" in the pop-up dialog box.



3. Click the "Add External Jars" button in the Libraries. Select "SmdtA20Api.jar" in the pop-up dialog box.



In the "Order and Export" option, select the Jar library you just imported, click the "UP" button in the upper right corner, and always raise the position of this library to the top. As shown below：



Build class path order and exported entries:
(Exported entries are contributed to dependent projects)

□ classes.jar - C:\Users\wxl
□ Android 4.4.2
☑ Android Private Libraries
☑ Android Dependencies
■ AndroidApi/src
■ AndroidApi/gen

Notes: All api calls

SmdtManager smdt = SmdtManager.create(this);

Start using the SMDT API， first declare the SmdtManager object, then you can start using the API. As shown in the following example:

private SmdtManager smdtManager; //Declare the SmdtManager object

smdtManager = SmdtManager.create(this); //use API

smdtManager.reboot("reboot");

6. Get System Information

6. 1. Function：public String smdtGetAPIVersion ()

Description：get the current information of API platform-version-data.

The suitable API version for this interface：

parameter name/returned value	types	instructions	examples
return value	String	The information of AIP platform, version and data.Format is as follows: XXX-Vn-201YMMDD.	A83T-V1-20160311

Notes：

For examples：

Log.e("API Version=” SmdtManager. smdtGetAPIVersion());

// output API Version= A83T-V1-20160311

6. 2. Function：public String getAndroidModel()

Description: get the model number of the current device.

The suitable API version for this interface：V1-20160321

parameter name/returned value	types	instructions	examples
return value	String	model of the devices	DS83X

Notes：

For examples：

Log.e("HW Model=” SmdtManager. getAndroidModel ());

/output HW Model= DS83X

6. 3. Function：public String getAndroidVersion ()

Description：get the android version of the current device.

The suitable API version for this interface：

parameter name/returned value	types	instructions	examples
return value	String	the version of android	4.4.4,5.1

Notes：

For examples：

Log.e("Android SDK=" SmdtManager. getAndroidVersion ());

/output Android SDK=4.4.4

6. 4. Function：public String getRunningMemory ()

Description：et the hardware memory size of the device.

The suitable API version for this interface：

parameter name/returned value	types	instructions	examples
return value	String	hardware memory size in GB	2GB

Notes：

For examples：

Log.e("DDR size=" SmdtManager. getRunningMemory ());

/output DDR size =1GB

6. 5. Function：public String getInternalStorageMemory ()

Description：get the capacity of hardware internal storage size of the device.

The suitable API version for this interface：

parameter name/returned value	types	instructions	examples

return value	String	Hardware internal storage size in GB	8GB

Notes：

For examples：

Log.e("EMMC size=" SmdtManager. getInternalStorageMemory ());

//output EMMC size =16GB

6. 6. Function：public String getFirmwareVersion ()

Description：get the firmware SDK version of the device.

The suitable API version for this interface：

parameter name/returned value	types	instructions	examples
return value	String	the firmware SDK version of the device	

Notes：

For examples：

Log.e("SW SDK=" SmdtManager. getFirmwareVersion ());

//output SW SDK = v1.2rc3

6. 7. Function：public String getFormattedKernelVersion()

Description: Get the firmware kernel version of the device.

The suitable API version for this interface：

parameter name/returned value	types	instructions	examples
return value	String	the firmware SDK version of the device	

Notes：

For examples：

Log.e("SW kernel Version=" SmdtManager. getFormattedKernelVersion());

//output SW kernel Version =3.4.39 wxl@server-109 #14

6. 8. Function: public String getAndroidDisplay ()

Description: Get the firmware system version and compilation date of the device.

The suitable API version for this interface:

parameter name/returned value	types	instructions	examples
returned value	String	Firmware system version and compilation date of the device.	

Notes:

For examples:

Log.e("SWAndroidDisplay=" SmdtManager. getAndroidDisplay ());

//output SW AndroidDisplay= octopus_perf_eng 4.4.4 KTU84Q 20160318 test-keys

6. 9. Function: public String getCpuFreq()

Description: Gets the device CPU frequency.

The suitable API version for this interface: V2.7

parameter name/returned value	types	types	examples
returned value	String	CPU frequency	

Notes:

For examples:

Log.e(TAG, smdtManager. getCpuFreq ());

6. 10. Function: public double getSystemCpuUsed()

Description: Gets the current device CPU usage.

The suitable API version for this interface: V2.7

parameter name/returned value	types	instructions	examples
returned value	Double	CPU usage	

Notes：

For examples：

Log.e(TAG, “” + smdtManager. getSystemCpuUsed ());

6. 11. Function：public String getSystemAvailableMemory ()

Description：Gets the system available memory.。

The suitable API version for this interface：V2.7

parameter name/returned value	types	instructions	examples
returned value	String	System available memory in MB	

Notes：

For examples：

Log.e(TAG, smdtManager. getSystemAvailableMemory ());

6. 12. Function：public double getAppMemory(int pid)

Description：Get the memory occupied by the APP.

The suitable API version for this interface：V2.7

parameter name/returned value	types	instructions	examples
PID	Int	Progress ID	
returned value	String	The memory occupied by APP is in MB	

Notes：

For examples：

Log.e(TAG, smdtManager. getAppMemory (android.os.Process.myPid()));

6. 13. Function：public void setUSBDebug(boolean debug)

Description：Turn USB debugging mode on or off.

The suitable API version for this interface：V2.7

parameter name/returned value	types	instructions	examples
debug	Boolean	turn on/off	

Notes：

//turn on USB debugging

```
Log.e(TAG, smdtManager. setUSBDebug (true));
```

6. 14. Function：public void setNetworkDebug(boolean debug)

Description：Turn on or off the network adb debug mode.

The suitable API version for this interface：V2.7

parameter name/returned value	types	instructions	examples
debug	Boolean	turn on/off	

Notes：

For examples：

```
//Turn on the network ADB debugging
Log.e(TAG, smdtManager. setNetworkDebug (true));
```

7. Time switches

7. 1. Function:public void smdtSetTimingSwitchMachine (String offTime,String OnTime, String enable)

Description：Time switch machine (first turn off and then turn on).

The suitable API version for this interface：

parameter name/returned value	types	instructions	examples
offTime	String	24-hour shutdown time	“9:50”
onTime	String	24-hour boot time	“20:10”
enable	String	(1 means use, 0 means not used)	1
return value	void	Failed:-1 , success:0	

Notes：

After setting the switch time, it will take effect every day, that is, the cycle is 1 day.

After power off, the MCU will reset and the time switch will be lost.

The time of power-off must be before the scheduled power-on time. If the scheduled power-on time is before the scheduled power-off time, it must be automatically turned off.

For examples：

```
//Now is 07:00. Turn off the computer at 9:50, and turn on the computer at 20:10 every day.
SmdtManager smdt = SmdtManager.create(this);
smdt. smdtSetTimingSwitchMachine ("9:50", "20:10"， 1);
//The current time is 07:00, and it can be switched on at 8:50 every day, and switched on or off at 22:10 every day.
```

7. 2. Function：public void shutDown()

Description：normally shut down the system.

The suitable API version for this interface：

parameter name/returned value	types	instructions	examples
return value	Void		

Notes：

For examples：

```
SmdtManager smdt = SmdtManager.create(this);
smdt.shutDown();
```

7. 3. Function：public void smdtReboot(String reason)

Description：normally restart the system.

The suitable API version for this interface：

parameter name/returned value	types	instructions	examples
reason	String	reserve	
return value	Void		

Notes：

For examples：

```
SmdtManager smdt = SmdtManager.create(this);
smdt. smdtReboot(“reboot”);
```

7. 4. Function： public int smdtSetPowerOnOff(char off_h, char off_m, char on_h, char on_m, char enable)

Description:： Set the switch time interval.

The suitable API version for this interface：

parameter name/returned value	types	instructions	examples
ff_h	char	shutdown hours	
off_m	char	Shutdown minutes	
on_h	char	Boot hours	
on_m	char	Boot minutes	
enable	char	enable 3: on, 0: off	
return value	int	fail:-1, others: success	

Notes：

For examples：

```
SmdtManager smdt = SmdtManager.create(this);
smdt.smdtSetPowerOnOff((char)0, (char)3, (char)0, (char)4, (char)3);
//The boot time interval is 4 minutes and the shutdown time interval is 3 minutes.
```

8. Watch dog

8. 1. Function: public int smdtWatchDogEnable(char enable)

Description：Enable or close the Watch Dog in the application layer.

The suitable API version for this interface：

parameter name/returned value	types	instructions	examples

enable	char	1=Enable the watchdog;
		0=Close the watchdog;
			1

return value	int	reserve	

Notes：

Watch Dog will be overrun after 60 seconds. Please make sure to feed Watch Dog by functions at least once in 60 seconds. It is recommended to feed the Dog in 10 to 20 seconds.

When Watch Dog is overtime,the system will automatically restart.

3.To exit the application, you must close the function of Watchdog. Otherwise the watchdog will overrun and the system will restart.

For examples：

8. 2. Function：public int smdtWatchDogFeed()

Description: Feed Watch Dog once, then resetting the counts of watchdog.

The suitable API version for this interface：

parameter name/returned value	types	instructions	examples

enable	char	1=Enable the watchdog;
		0=Enable the watchdog;
			1

return value	int	reserve	

Notes：

Make sure the watchdog is enabled before calling this function.

For example：

```
SmdtManager smdt = SmdtManager.create(this);
//Enable watchdog
smdt smdtWatchDogEnable ((char)1);
//Feed the dog once every 10 seconds
smdt. smdtWatchDogFeed ();
//Close the watchdog
smdt. smdtWatchDogEnable ((char)0);
```

9. Display

9. 1. Function: public void smdtTakeScreenshot(String path,String name ,Context context)

Description: Capture the current full screen as a png format image and rename it to the appropriate location.

The suitable API version for this interface：

parameter name/returned value	types	instructions	examples
path	String	Save absolute path	“/storage/usbcard1”
name	String	Renamed png format image	“aaa.png”
context	Context	Context	context

Notes：

- Whether path is a readable and writable directory.
- Name contains no illegal characters. The suffix is .png at the same time.
- This method must be signed by APK to be valid.

For examples:

```
//Take a screenshot and save to"/storage/usbcard1/aaa.png"
SmdtManager smdt = SmdtManager.create(this);
smdt. smdtTakeScreenshot (“/storage/usbcard1”,“aaa.png”,this);
```

9. 2. Function: public Bitmap smdtScreenShot(Context context)

Description: Screenshot adds a method to return directly to the bitmap method

For examples：

```
SmdtManager smdt = SmdtManager.create(this);
smdt. smdtScreenShot(this);
```

9. 3. Function: public void setRotation (String rotationDegree)

Description: Sets the screen to rotate the N angle counterclockwise.

The suitable API version for this interface：

parameter name/returned value	types	instructions	examples
rotationDegree	String	Only supports 0, 90, 180, 270 angles.	"0" "90" "180" "270"
return value	void		

Notes：

The function will automatically restart after the system has taken effect.

For examples：

SmdtManager smdt = SmdtManager.create(this);

smdt.setRotation ("0");

smdt.setRotation ("90");

smdt.setRotation ("180");

smdt.setRotation ("270");

9. 4. Function：public int smdtGetScreenWidth(Context context)

Description：Get the display resolution to a wide X pixels.

The suitable API version for this interface：

parameter name/returned value	types	instructions	examples
context	Context	context	
return value	int	Actual wide pixel of the screen	

Notes：

For example：

9. 5. Function：public int smdtGetScreenHeight (Context context)

Description：Get the display resolution high Y pixels.

The suitable API version for this interface：

parameter name/returned value	types	instructions	examples
context	Context	context	
return value	int	Actual high pixel of the screen	

Notes：

For example：

```
//Get resolution in 1080p screen
    SmdtManager smdt = SmdtManager.create(this);
int X=smdt. smdtGetScreenWidth (this)；
int Y= smdt. smdtGetScreenHeight (this);
Log.e(TAG,"Width: Height=”+X+”:”+Y);
//Will output Width: Height= 1920:1080
```

9. 6. Function：public int getExtendScreenWidth()

Description: Get the secondary screen resolution X pixels wide.

The suitable API version for this interface：

parameter name/returned value	types	instruction	examples
Return	int	Actual wide pixel of the screen	

value			

Notes：

For example：：

9. 7. Function：public int getExtendScreenHeight()

Description: The secondary screen resolution is high Y pixels.

The suitable API version for this interface:

parameter name/returned value	types	instruction	examples
Return value	int	Actual high pixel of the screen	

Notes：

For example：

//Get the resolution in the secondary screen with 1080p inserted

    SmdtManager smdt = SmdtManager.create(this);

  int X=smdt. getExtendScreenWidth()；

int Y= smdt. getExtendScreenHeight();

Log.e(TAG,"Width: Height=”+X+”:”+Y);

//wil output Width: Height= 1920:1080

9. 8. Function：public int smdtSetStatusBar(Context context, boolean enable)

Description：Set to show or hide the dynamic status bar.(Only the System system signature APP can be called)

The suitable API version for this interface：

parameter name/returned value	types	instruction	examples

context	Context	context	
enable	boolean	True：Display status bar False：Hide status bar	
Return value	Int	0：success N: error	

Notes：

For examples：

SmdtManager smdt = SmdtManager.create(this);

// Display status bar

smdt. smdtSetStatusBar（getApplicationContext，TRUE）；

/Hide status bar

smdt. smdtSetStatusBar（getApplicationContext，FALSE）；

9. 9. Function：public int smdtGetStatusBar(Context context)

Description：Get the current dynamic status bar display or hide status.

The suitable API version for this interface：

parameter name/returned value	types	instruction	examples
context	Context	context	
return value	Int	0：Hide status bar 1：Display status bar N: Error	

Notes：

For examples：

Int ret= SmdtManager. smdtSetStatusBar（getApplicationContext）；

if(ret == 0) {

    Log.e("StatusBar hide.”);

```
} else if (ret == 1) {
        Log.e("StatusBar show.”);
} else {
        Log.e("something error=”+ ret);
}
```

9. 10. Function：public void smdtSetLcdBackLight(int on)

Description： Turn off the screen, only turn off the backlight, but do not go to sleep, the software continues to run.

The suitable API version for this interface：

parameter name/returned value	types	instruction	examples
On	Int	0 represents off 1 representative open	
return value			

Notes：

For example：

```
SmdtManager smdt = SmdtManager.create(this);
//Turn off the backlight
smdt,smdtSetLcdBackLight(0);
//Turn on the backlight
smdt.smdtSetLcdBackLight(1);
```

9. 11. Function：public void smdtGetLcdLightStatus();

Gets the current state of the screen, on or off

The suitable API version for this interface：

return value	instruction	instruction	examples

1	int	The current screen is open	
0	int	The current screen is off	

For example：

SmdtManager smdt = SmdtManager.create(this);

int status = smdt. smdtGetLcdLightStatus();

Log.i("status====", status+"");

9. 12. Function：public void smdtSetEDPBackLight(int on)

Description: Turn off the screen, only turn off the backlight, but do not go to sleep, the software continues to run.(3288 5.1 not supported)

The suitable API version for this interface：

parameter name/returned value	types	instruction	examples
On	Int	0 represents off 1 representative open	
return value			

Notes：

For exampl：

SmdtManager smdt = SmdtManager.create(this);

//Turn off the backlight

smdt,smdtSetEDPBackLight(0);

//Turn on the backlight

smdt.smdtSetEDPBackLight(1);

9. 13. Function：public void smdtGetEDPBackLight();

Gets the current state of the screen, on or off. (3288 5.1 not supported)

The suitable API version for this interface：

returned value	types	instruction	examples
1	int	The current screen is on	
0	int	The current screen is off	
-1	int	Get results fail	

For example：

```
SmdtManager smdt = SmdtManager.create(this);
int  status  =  smdt. smdtGetLcdLightStatus();
Log.i("status====",  status+"");
```

9. 14. Function：public void setBrightness(ContentResolver resolver, int brightness);

Description: Set the backlight brightness.

The suitable API version for this interface：

Parameter name / return value	types	instruction	examples
resolver	ContentResolver	ContentResolver object	
brightness	int	Brightness value	0～255
return value	void		

Notes：

For example：

```
SmdtManager smdt = SmdtManager.create(this);
smdt.setBrightness(getContentResolver(), 255);
```

9. 15. Function：public void setGestureBar(boolean enable)

Description：Sets whether gestures can pull out the navigation bar.

The suitable API version for this interface：V2.7

Parameter name / return value	types	instruction	examples
enable	Boolean	whether gestures can pull out the navigation bar	

Notes：

For example：

//Set to pull out the status bar with a gesture

smdtManager. setGestureBar (true);

9. 16. Function：public boolean getGestureBar()

Description：Sets whether gestures can pull out the navigation bar.

The suitable API version for this interface：V2.7

Parameter name / return value	types	instruction	examples
enable	Boolean	whether gestures can pull out the navigation bar	

Notes：

For example：

//Set to pull out the status bar with a gesture

smdtManager. getGestureBar ();

9. 17. Function：public void hideSoftKeyboard(boolean hide)

Description：Hide the soft keyboard.

The suitable API version for this interface：V2.7

Parameter name / return value	types	instruction	examples
hide	Boolean	Hide/show soft keyboard	

		When hidden, the input field does not pop up the soft keyboard	

Notes：

For example：

//Hide soft keyboard

smdtManager. hideSoftKeyboard (true);

9. 18. Function：public void setKeyReject(boolean reject)

Description：Do not press button to report.

The suitable API version for this interface：V2.7

Parameter name / return value	types	instruction	examples
reject	Boolean	Disable/allow button escalation	

For example：

//Prohibit key press report

smdtManager. setKeyReject (true);

9. 19. Function：public void setTouchReject(boolean reject)

Description: Disable touch and click.

The suitable API version for this interface：V2.7

Parameter name / return value	types	instruction	examples
reject	Boolean	Disable/allow touch and click	

For example：

//Disable touch and click.

smdtManager. setTouchReject (false);

9. 20. Function：public void setOnKeyListener(OnClickListener listener)

Description：Listen for key click events, which can be used in non-Activity classes.

The suitable API version for this interface：V2.7

Parameter name / return value	types	instruction	examples

listener	SmdtManager. OnClickListener	callback class	

For example：

//Listen for key click events

```
    smdt.setOnKeyListener(new SmdtManager.OnClickListener() {
        @Override
        public void onKeyDown(int keyCode) throws
RemoteException {
            Log.i(TAG, “keyCode:” + keyCode);
        }
    });
```

10. Installation upgrade

10. 1. Function: public void smdtInstallPackage(Context context ,File file)

Description：The update differential package file upgrade will be restarted.

The suitable API version for this interface：

Parameter name / return value	types	instruction	examples
file	File	Update differential package	
return value	void		

Notes：

The function will automatically restart the system for an update.zip upgrade.

Ensure that the file file exists and is fully available.

The file name must be update.zip.

For examples：

SmdtManager smdt = SmdtManager.create(this);

//Upgrade the differential package update.zip under the SD card.

smdt.smdtInstallPackage(getApplicationContext,new File("/mnt/sdcard/update.zip"));

10. 2. Function: public void smdtRebootRecovery ()

Description ： will reboot into recovery mode, update differential package upgrade (same as smdtInstallPackage, but do not specify the upgrade package location).

The suitable API version for this interface：

Parameter name / return value	types	instruction	examples

None	None		

Notes：

The function will automatically restart the system for an update.zip upgrade.

Ensure that the update.zip file exists on the internal SD card or external SD card and is fully available.

For example：

SmdtManager smdt = SmdtManager.create(this);

smdt. smdtRebootRecovery ();

10. 3. Function: public void smdtSilentInstall (String path,Context context)

Description：Silently install the APK app.

The suitable API version for this interface：

Parameter name / return value	Types	instruction	examples
path	String	Absolute path of the APK to be installed	"/mnt/sdcard/test.apk"
context	Context	context	context
return value	void		

Notes：

The APK file exists and is fully available.

For example：

//Install test.apk

SmdtManager smdt = SmdtManager.create(this);

smdt. smdtSilentInstall ("/mnt/sdcard/test.apk",getApplicationContext);

10. 4. Function: public void installApp(String apkPath, InstallCallback installCallback)

Description：Install APK application silently, callback installed successfully.

The suitable API version for this interface：V2.7

Parameter name / return value	Types	instruction	examples
apkPath	String	Absolute path to APK to install	"/mnt/sdcard/test.apk"
installCallback	InstallCallback	callback class	
return value	void		

Notes：

The APK file exists and is fully available.

For example：

```
    String path =
Environment.getExternalStorageDirectory().getPath().toString()
+ File.separator + "Update.apk";
    if (!new File(path).exists()) {
        Toast.makeText(getApplicationContext(), "安装失败，
Update.apk 不存在", Toast.LENGTH_SHORT).show();
        break;
    }
    smdt.installApp(path, new SmdtManager.InstallCallback() {
        @Override
        public void onInstallFinished(String packageName, int
returnCode, String msg) throws RemoteException {
    // returnCode 代表成功，其它代表失败
            Log.i(TAG, "packageName:" + packageName +
" ,returnCode:" + returnCode + ",msg:" + msg);
          }
    Parameter name / return value
```

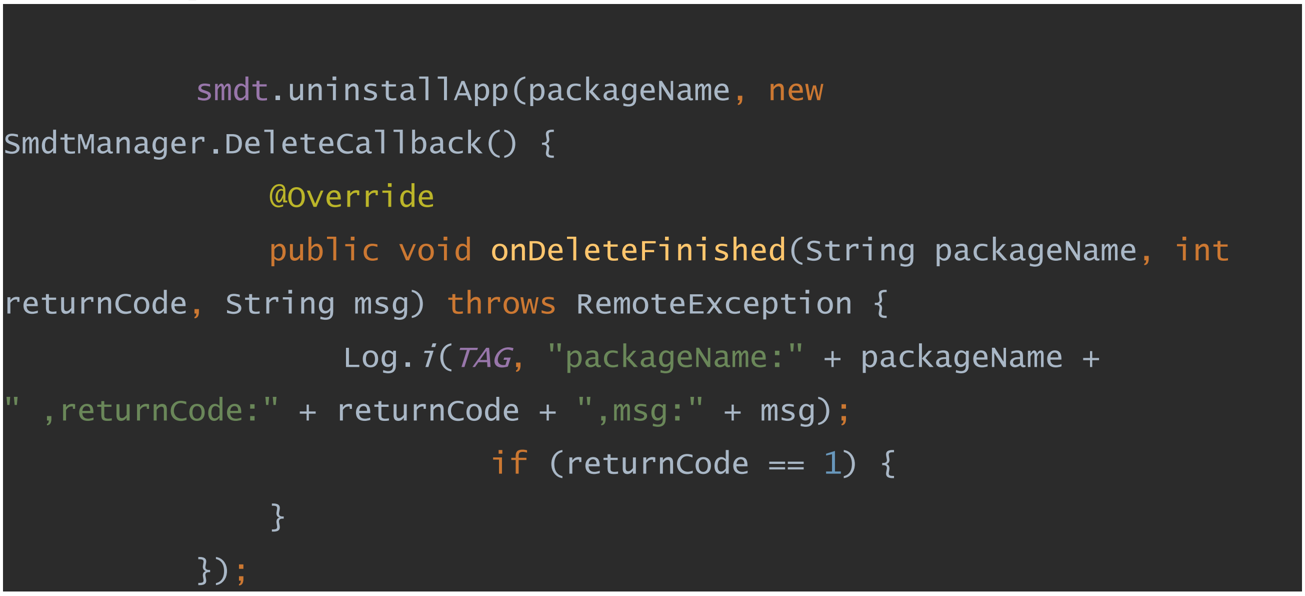
10. 5. Function: public void uninstallApp(String packageName, DeleteCallback deleteCallback)

Description：Silent uninstall APK application with successful callback.

The suitable API version for this interface：V2.7

Parameter name / return value	Types	instruction	examples
packageName	String	Name of the package to be unloaded	Com.android.test
deleteCallback	DeleteCallback	callback class	
return value	void		

For example：



```
          smdt.uninstallApp(packageName, new
SmdtManager.DeleteCallback() {
              @Override
              public void onDeleteFinished(String packageName, int
returnCode, String msg) throws RemoteException {
                  Log.i(TAG, "packageName:" + packageName +
" ,returnCode:" + returnCode + ",msg:" + msg);
                          if (returnCode == 1) {
              }
          });
```

11. Network

11. 1. Function：public String smdtGetEthMacAddress ()

Description：Get the MAC address of the device Ethernet.

The suitable API version for this interface：

Parameter name / return value	types	instruction		examples

return value	String	Ethernet MAC address		12:23:34:56:AA:BB

Notes：

For example：

Log.e("ETH MAC=" SmdtManager. smdtGetEthMacAddress ());

//Output ETH MAC =XX：XX：XX：XX

11. 2. Function：public String smdtGetEthIPAddress()

Description: Get the IP address of the device Ethernet.

The suitable API version for this interface：

Parameter name / return value	types	instruction	examples
return value	String	Ethernet IP address	192.168.1.100

Notes：

For example：

Log.e("ETH IP=" SmdtManager. smdtGetEthIPAddress());

//Output ETH IP =192.168.1.103

11. 3. Function：public void smdtSetEthIPAddress(String mIpaddr, String mMask, String mGw, String mDns)

Description：Set the IP address of the device Ethernet.

The suitable API version for this interface：

Parameter name / return value	types	instruction	examples
mIpaddr	String	IP address	

mMask	String	Subnet mask	
mGw	String	Gateway	
mDns	String	DNS	
return value	String	Ethernet IP address	

For example：

smdt.smdtSetEthIPAddress("192.168.1.100", "255.255.255.0", "192.9.50.1", "202.96.134.133");

11. 4. Function：public String getCurrentNetType()

Description：Get the type of current network connection.

The suitable API version for this interface：

Parameter name / return value	types	instruction	examples
return value	String	Network Types 2G: GSM network 3G: WCDM/EVDO network 4G: FDD and other 4G networks WIFI: WIFI wireless network ETH: Ethernet wired network null: no network	

Notes：

Notes：

```
Log.e("NET TYPE=”+smdtManager.getCurrentNetType());
//output NET TYPE=3G
```

11. 5. Function：public String smdtGetEthernetState

Description：Get the IP address of the device Ethernet.

The API version that implements this interfac：

Parameter name / return value	types	instruction	examples
return value	boolean	True: open, false:close	

Notes：

For example：

boolean isEnable = smdtManager.smdtGetEthernetState());

11. 6. Function：public WifiUtils getWifiInterface(Context context)

Description：Control Wifi switch.

The suitable API version for this interface：

Parameter name / return value	types	instruction	examples
return value	WifiUtils	Class that controls the WIfi switch	

Notes：Need permission，

<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />

<uses-permission android:name="android.permission.CHANGE_WIFI_STATE" />

For example：

smdtManager.getWifiInterface(getApplicationContext()).wifiOpen(); //打开 wifi

smdtManager.getWifiInterface(getApplicationContext()).wifiClose(); //关闭 wifi

12. External storage media

12. 1. Function：public String smdtGetSDcardPath(Context context)

Description：Get the external storage SD card path.

The API version that implements this interface：

Parameter name / return value	types	instruction	examples
return value	String	External storage SD card absolute path	

Notes：

For example：

Log.e("SD=”+smdtManager. smdtGetSDcardPath (context));

//Output SD=/mnt/extsd

12. 2. Output：public String smdtGetUSBPath(Context context, int num)

Description：et the external storage U disk path.

The API version that implements this interface：

Parameter name / return value	types	instruction	examples
num	int	U disk drive letter	0,1,2,3
return value	String	External storage U disk absolute path	

Notes：

For example：

Log.e("USB=”+smdtManager. smdtGetUSBPath (0);
//Output USB="/storage/usbhost0";

12. 3. Function : public void unmountVolume(String path,boolean force,boolean removeEncryption)

Description: Uninstall external storage. (3288 5.1 Not supported)

The API version that implements this interface：

Parameter name / return value	types	instruction	examples
return value	void		
path	String	The absolute path of the external storage to be uninstalled (specifically, the actual path of the platform)	/storage/emulated/0 /storage/extsd /storage/usbhost0 /storage/usbhost1 /storage/usbhost2 /storage/usbhost3 /storage/usbhost4
force	boolean	Whether to force uninstall	true
removeEncryption	boolean	Whether to remove the encryption device	false

Notes：

For example：

smdt.unmountVolume("/storage/usbhost2", true, false);

12. 4. Function：public byte[] smdtReadExtROM(int deviceId, int areaId, int start_addr, int size)

Description: Read external EEPROM storage.

The API version that implements this interface：

Parameter name / return value	types	instruction	examples
deviceId	Int	Device ID, starting from 0, each ID represents an EEPROM	
areaId	Int	Area ID, starting from 1, each ID represents an area of an EEPROM	
start_addr	Int	Start address, indicating reading data from this area	
size	Byte	Data length	
return value	byte[]	Read Buffer data	

Notes：The maximum capacity currently only supports 2k

For example：

1, write:mdtManager.smdtWriteExtROM(0,1,300, 5, “12345”);

2, read:Byte [] data = smdtManager.smdtReadExtROM(0,1, 300, 5);

Log.d("data=”+new String(data));

//output data=12345;

12. 5. Function：public int smdtWriteExtROM(int deviceId, int areaId, int start_addr, int size , byte[] buf)

Description：Write to external EEPROM storage.

The API version that implements this interface：

Parameter	types	instruction	examples

name / return value			
deviceId	Int	Device ID, starting from 0, each ID represents an EEPROM	
areaId	Int	Area ID, starting from 1, each ID represents an area of an EEPROM	
start_addr	Int	Start address, indicating that data is written from this area	
size	int	Data length	
buf	byte[]	Buffer data written	
return value	Int	0: successful, non-0: failed	

Notes：Maximum capacity only supports 2K

For example：

smdtManager.smdtWriteExtROM(0,1,300, 5, “12345”);

13. Hardware interface

13. 1. Function：public String getUartPath(String uart)

Description: Get the absolute path of the serial port.

The API version that implements this interface：

Parameter name / return value	types	instruction	examples
Uart	String	Port number corresponding to the serial port - uart 0, uart 1, uart 2, uart 3	
return value	String	UART absolute path	

Notes：

For example：

Log.e("UART-1=”+smdtManager. getUartPath (uart1));

//output UART-1=/dev/ttyS1

13. 2. Function：public int smdtSetUsbPower(int type, int num, int values)

Description: Set the USB port power.

The API version that implements this interface：

Parameter name / return value	types	instruction	examples
type	int	USB types OTG/HOST: 1 HUB: 2	

num	int	USB code: 1~3	
valuse	int	On: 1, Off: 0	
return value	int	Success: 0, Failure: -1	

Notes:

For example:

smdtManager.smdtSetUsbPower(1, 1, 1);

//Set usb1 to open

smdtManager.smdtSetUsbPower(1, 1, 0);

//Set usb1 to off

13. 3. Function: public int smdtReadExtrnalGpioValue (int io)

Description: Get the IO port input status.

The API version that implements this interface:

Parameter name / return value	types	instruction	examples
io	int	Io corresponding port number - 0, 1, 2, 3	1
return value	int	1: IO input is high 0: IO input is low	

Notes:

Determine the serial number of the io port of the hardware connection.

Make sure the io port of the hardware connection is in input mode.

For example:

```
If（smdtManager. smdtReadExtrnalGpioValue (1)）{
        Log.e("IO-1=1");
}
```

13. 4. Function：public void smdtSetExtrnalGpioValue(int io, boolean state)

Description：Set the output status of the IO port.

he API version that implements this interface：

Parameter name / return value	types	instruction	examples
io	int	Io corresponding port number - 0, 1, 2, 3	1
state	boolean	TRUE: IO output is high FALSE: IO output is low	
return value	Void		

Notes：

Determine the serial number of the io port of the hardware connection.

Make sure the io port of the hardware connection is in output mode.

For example：

//Output IO-1 port is low

smdtManager. smdtSetExtrnalGpioValue (1,false)；

13. 5. Function：public int smdtSetGpioDirection(int io, int value)

Description：Set the direction of the IO port.

The API version that implements this interface：

Parameter name / return value	types	instruction	examples
io	int	Io corresponding port number -1, 2, 3...	1

direction	int	0: IO is set to input 1: IO is set to output	
value	int	Enter a value of 0 or 1	
return value	int	0: successful, -1: failed	

Notes：

Determine the serial number of the io port of the hardware connection.

Determine the io mode supported by the hardware.

The GPIO level may be restored to the default value after being set to the output mode.

For example：

```
/Output IO-1 port is input
if (0 == smdtManager. smdtSetGpioDirection(1, 0))
{
//Set successfully
}
```

14. Voice Control

14. 1. It is recommended to implement various sound controls using the Android standard interface.

14. 2. Function：public boolean smdtSetVolume(Context context, int volume)

Description：Set the current sound size. (Recommended to use Android standard interface)

The API version that implements this interface：

Parameter name / return value	types	instruction	examples
context	Context	Context	
volume	int	Sound value (1~15)	
return value	boolean	TRUE: Set successfully FALSE: Setup failed	

Notes：

Set the current channel sound size.

For example：

//Set the current channel sound=1

smdtManager.smdtSetVolume(getApplicationContext(), 1);

14. 3. Function：public int smdtGetVolume(Context context)

Description：Get the current channel sound. (Recommended to use Android standard interface)

The API version that implements this interface：

Parameter name / return value	types	instruction	examples
context	Context	context	
return value	int	Sound value (1~15)	

Note：

1.Get the current channel volume value.

For example：

//Get the current channel sound

Int v=smdtManager.getSetVolume(getApplicationContext());

14. 4. Function：public int setHeadsetMicOnOff(int value)

Description：Switch the headset microphone.

The API version that implements this interface：

Parameter name / return value	types	instruction	examples
value	int	0: off, 1: on	
return value	int		

Notes：

1, value must be 0 or 1

2, It only works if you plug in headphones

For example：

```
smdtManager.setHeadsetMicOnOff(1);  //Open the headset
smdtManager.setHeadsetMicOnOff(0);  //Turn off the headset
```

15. Camera

15. 1. **No special, it is recommended to use Android standard interface to achieve various camera control.**

16. GPS

16. 1. **No special, it is recommended to use the Android standard interface to achieve GPS control.**

17. Others

17. 1. Function: public void setTime (Context context, int year, int month, int day, int hour, int minute)

Description：Set and save the system time.

The API version that implements this interface：

Parameter name / return value	types	instruction	examples
context	Context	Context	context
year	int	year	
month	int	month	
day	int	day	
hour	int	(24-hour) hour	
minute	int	(24-hour) minute	
return value	void		

Note：

Due to system time setting permission restrictions, the application APK cannot set the system time. Therefore, it is also possible to call the Android standard setting interface by authorizing the APK signature.

If the setting time is February 30th, the system will automatically adjust to the next month's No. 1 (leap year)

If the setting time is February 29 or 30, the system will automatically adjust to the next month's No. 1 or No. 2（common year ）

For example：

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

smdtManager. setTime (getApplicationContext, 2016, 3, 16, 13, 44);

17. 2. Function: public void execSuCmd (String command)

Description：The shell command will be run with ROOT privileges.

The API version that implements this interface：

Parameter name / return value	types	instruction	examples
command	String	Shell command	“ls”
return value	void		

Notes：

For example：

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

smdtManager.execSuCmd ("ls /mnt/sdcard/ ");

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

17. 3. Function：public void smdtGetSystemLogcat(String folderPath)

Description：Grab the LOG of the Android layer and save the corresponding directory.

The API version that implements this interface：

Parameter name / return value	types	instruction	examples
folderPath	String	Log saves the absolute path	“/storage/usbcard1/”
return value	Void		

For example：

```
//Save to a USB flash drive，
SmdtManager smdt = SmdtManager.creat(this)；
smdt. smdtGetSystemLogcat (“/storage/usbcard1/”) ；
//View log: /storage/usbcard1/ logcat.txt
```

17. 4. Function：public int smdtSetControl(int type, int values)

Description：Other device related settings.

he API version that implements this interface：

Parameter name / return value	types	instruction	examples
type	int	3:wifi power 4:wifi reset 5:led control 6:speak power 7:lvds power 8:lvds reset 9:3G power 10:3G reset 11:LAN power 12:LAN reset 13:SD power 14:SD reset 15:TP power 16:TP reset	

values	int	0: off, 1: on	
return value	int		

Notes：

For example：

//Turn on the Ethernet power

smdt.smdtSetControl(11, 1);

Function： public String getCurrentNetType()

All reset operations need to be operated as follows, otherwise they will not succeed.

```
//reset 3G(3G reset)
try {
smdt.smdtSetControl(10, 0);
Thread.sleep(1000);
smdt.smdtSetControl(10, 1);
} catch (Exception e) {
    e.printStackTrace();
}
```

17. 5. Function： public int getScreenNumber()

Description: Get the value of the dual display interface

The API version that implements this interface：

Parameter name / return value	type	instruction	examples
return value	int		

For example：

int screen_number = smdt.getScreenNumber();

17. 6. Function：public int getHdmiinStatus()

Description: Get the Hdmi in status value

The API version that implements this interface：

Parameter name / return value	type	instruction	examples
return value	int		

For example：

int screen_number = smdt.getHdmiinStatus();

//1 means insert, 0 means no insert

17. 7. Function：public int setHdmiInAudioEnable()

Description: Switch Hdmi in sound

The API version that implements this interface：

Parameter name / return value	type	instruction	examples
context	Context	context	
enable	boolean	True: on false: off	
return value	boolean	true:success false:fail	

For example：

int ret= smdt.setHdmiInAudioEnable(getApplicationContext(), false); //Turn off the HDMIIN sound

int ret= smdt.setHdmiInAudioEnable(getApplicationContext(), true); //Turn on the HDMIIN sound

17. 8. Function：public void setAllowinstall(boolean allowed)

Description: Prohibit install the APP switch

The API version that implements this interface：V2.7

Parameter name / return value	type	instruction	examples
allowed	Boolean	Allow/disable APP installation	

For example：

/ Prohibit install the APP

smdtManager.setAllowinstall(false);

17. 1. Function：public boolean isAllowinstall()

Description: Prohibit install the APP on/off state

The API version that implements this interface：V2.7

Parameter name / return value	type	instruction	examples
return value	Boolean	Allow/disable APP installation	

For example：

// isAllowInstall = true, the code can install the APP

Boolean isAllowInstall = smdtManager. isAllowinstall ();

17. 2. Function：public void setAllowUninstall(boolean allowed)

Description: Prohibit uninstall APP

The API version that implements this interface：V2.7

Parameter name / return value	type	instruction	examples
allowed	Boolean	Allow/disable uninstall APP	

For example：

//Prohibit uninstall APP

smdtManager. setAllowUninstall (false);

17. 3. Function：public boolean isAllowUninstall ()

Description: Get permission/prohibit to uninstall APP status

The API version that implements this interface：V2.7

Parameter name / return value	type	instruction	examples
return value	Boolean	Allow/disable uninstall APP status	

For example：

// isAllowUninstall = true, The code can uninstall the APP

Boolean isAllowUninstall = smdtManager. isAllowUninstall ();

17. 4. Function：public void addInstallWhiteList(String packageName)

Description: Set the whitelist to install APPs. When banned, the whitelist APP can still be installed

The API version that implements this interface：V2.7

Parameter name / return value	type	instruction	examples
packageName	String	package name	

For example：

smdtManager. addInstallWhiteList (“com.android.test”);

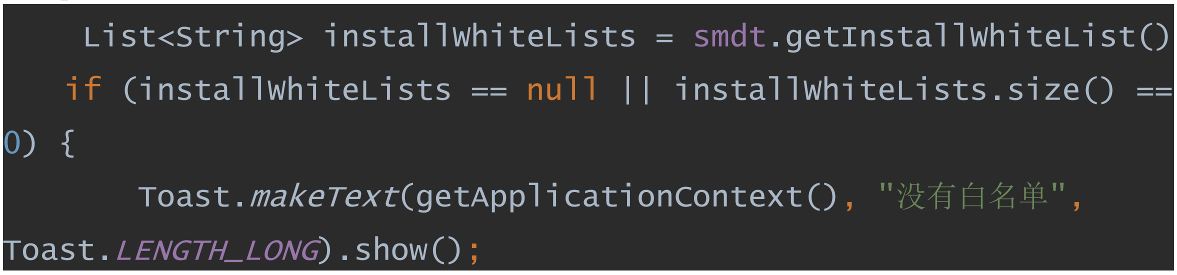
17. 5. Function：public List<String> getInstallWhiteList()

Description: Get the whitelist list

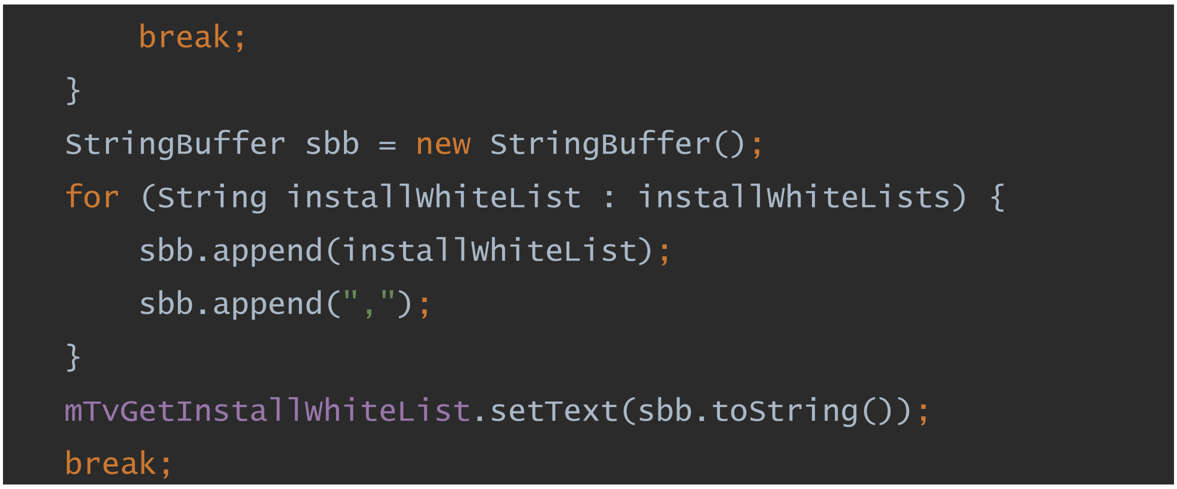
The API version that implements this interface：V2.7

Parameter name / return value	type	instruction	examples
return value	List<String>	Whitelist	

For example：



```
    List<String> installWhiteLists = smdt.getInstallWhiteList();
   if (installWhiteLists == null || installWhiteLists.size() ==
0) {
       Toast.makeText(getApplicationContext(), "没有白名单",
Toast.LENGTH_LONG).show();
```



```
        break;
    }
    StringBuffer sbb = new StringBuffer();
    for (String installWhiteList : installWhiteLists) {
        sbb.append(installWhiteList);
        sbb.append(",");
    }
    mTvGetInstallWhiteList.setText(sbb.toString());
    break;
```

17. 6. Function：public void setNtpServer(String server)

Description: Set the address of the time NTP server

The API version that implements this interface：V2.7

Parameter name / return value	type	instruction	examples
server	String	The address of the NTP server	

For example：

smdtManager. setNtpServer (“time.windows.com”);

17. 7. Description：public String getNtpServer()

Description: Set the whitelist to install apps. When banned, the whitelist APP can still be installed

The API version that implements this interface：V2.7

Parameter name / return value	type	instruction	examples
return value	String	The address of the NTP server	

For example：

smdtManager. getNtpServer ();