

1. Preface

1.1. Brief Introduction

1.1.1 Copyright Declaration

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.

1.1.2 Technical Support

If you have any questions about this documents, you can log on our official website www.shimeta.com.cn, and find the contact information of our technical support on the [Service and support](#) interface. You can contact the technical support by telephone, QQ or email during office hours (Monday to Friday 9:00–12:00/1:00–6:00 PM).

1.1.3 Resources Development

[OS resources package](#) The extract code: 0xfn

It contains JAR package, system signature file, Word form document, development DemoAPK and DemoApk source code, etc.

1.1.4 Special Matters

Most of the API in Lite version is not supported. Please pay attention to the platform because some interfaces are specially marked for use on Linux platform.

1.2. Version Modification Records

- The versions of the API that has been released to the public

version	Description	Date
V1.0	Established	2021-

		12-07
V2.0.4	Update	2021-12-10
V2.2.0	Update JAR package	2021-12-20
V2.3.1	Add NPU version acquisition Compatible with old API interfaces Optimize ApiDemo	2022-01-18
V2.5.0	Added a custom desktop shortcut icon interface	2022-03-04
V2.7.0	Add timing switcher status acquisition interface Add automatic application installation relevant function interface Add WiFi connection interface Add Internet guarding relevant interface Eliminate startup defaulted WiFi connection interface	2022-05-24
V2.8.0	Add startup Logo setting interface Add startup animation interface	2022-06-02
V2.9.0	Add screen parameter setting and gaining interface	2022-06-14
V2.11.0	Compatible with turnstile API interface (Wiegand/relay/three-color light/fan) Add device temperature interface Add device temperature interface	2022-07-29
V2.12.0	Add whether external storage devices are allowed to clone, upgrade, and automatically install application interfaces Restore the API part of compatible old turnstile JNI interface Add constant classes	2022-08-08
V2.14.1	Add WebView version acquisition interface (IMEI/ICCID/IMSI) Add SIM card information acquisition (IMEI/ICCID/IMSI) Add MCU power reset interface Add timing switching error codes Eliminate the setting of device model/producer/software version	2022-09-20
V2.16.3	Add network priority interface Add multiple networks coexist interface Add backlight switch interface Restore speaker volume gaining and audio inputting interface Add audio input type interface Eliminate the status of the microphone switch setting and acquisition interface Eliminate dual-screen collaboration interface	2022-11-11
V2.25.0	Add the concrete date of the timing switcher interface Add OTA-AB partitioning upgrade interface Update the OTA upgrade error codes	2024-03-25

1.3. How to Use the SMDT API

1.3.1 Get The Object Instance Method

Function name:: public static SmdtManagerNew getInstance(Context context)

Description: Get a single instance of the SmdtManagerNew object

Parameter:

Parameter name	Type	instruction
context	Context	context

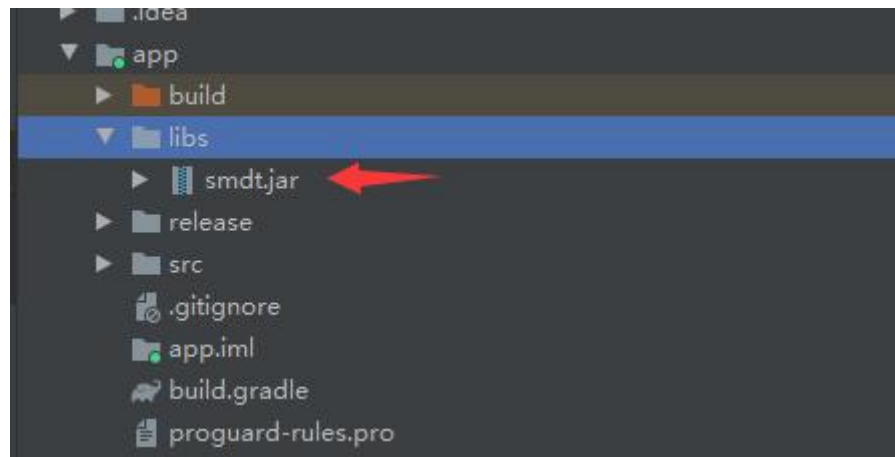
Return parameter description

Parameter name	Type	instruction
Return	SmdtManagerNew	Return to separate SmdtManagerNew object

Typical Example

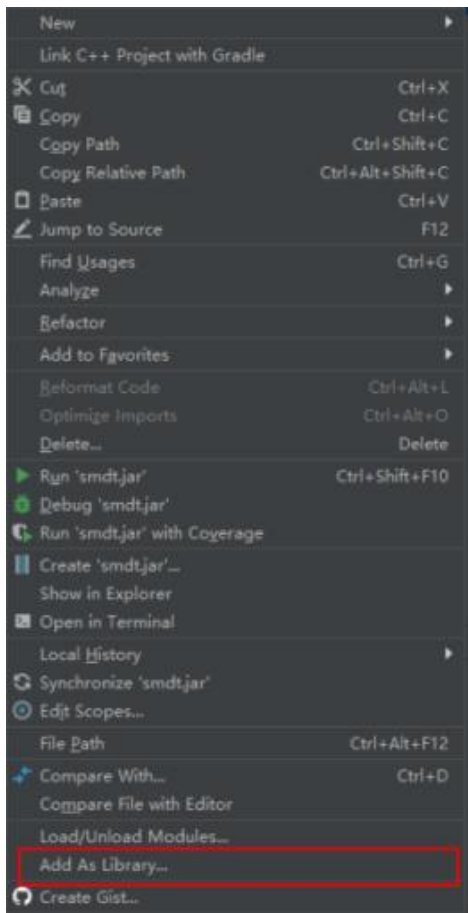
```
SmdtManagerNew smdt = SmdtManagerNew.getInstance(this);  
System.out.println(smdt.info_getApiVersion());  
Output Result : V1.0.0-release
```

Lead in JAR Package



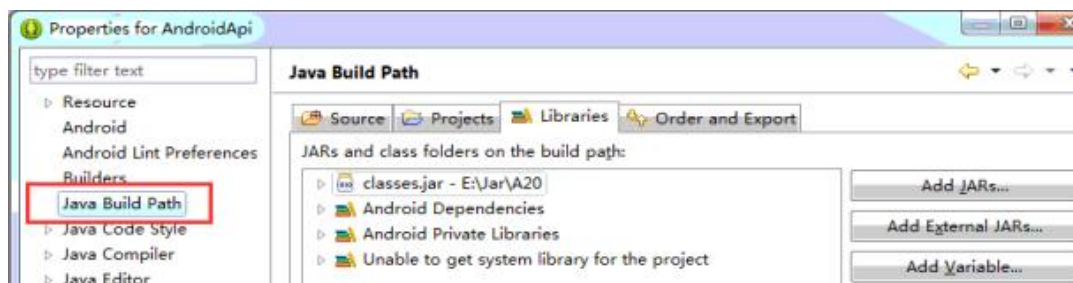
1.3.2 How to Use SMDT API in Android Studio

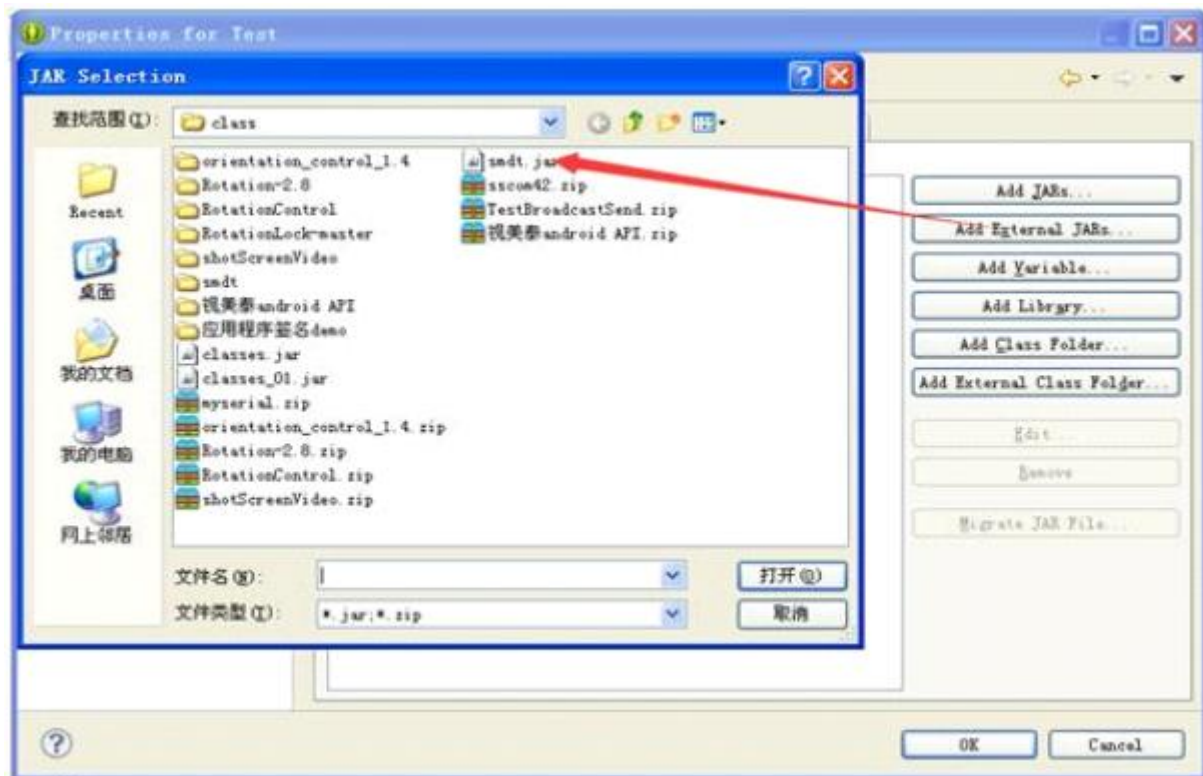
1. Copy smdt.jar to engineering catalogue\app\libs\ ;
2. Click the file jar of the file libs by the right mouse button to choose **add as Library**;
3. Then use instance object call methods in codes.



1.3.3 How to Use SMDT API in Eclipse

1. Open **eclipse** and choose item, then click the right mouse button and choose **properties**.
2. Then choose “Java Build Path” in dialogue box .





3. Click the "Add External Jars" button in the Libraries. Select "SmdtA20Api.jar" in the pop-up dialog box.
4. 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:



1.4、API Naming Rules

An API consists of three parts: the prefix name+ a verb + the name.
The name should correspond to the function of the API.

Prefix name

instruction

info	Device information category
disp	Display management category
net	Internet control category
sys	System control category
dev	Hardware control category
custom	Customer customization category
Verb	
instruction	
get	get
set	set
write	write
read	read
open	open
close	close
reset	reset
register	register
unregister	unregister
check	check
test	test
change	change
sync	update
add	add
del	delete
send	send
receive	receive
do	do

1.5. Global Error Codes

The Type of Error Codes

variable name	Error codes	instruction
RET_API_OK	0	success
RET_API_ERR_NG	-1	Not support
RET_API_ERR_PARA	-2	Parameter is incorrect
RET_API_ERR_FILE_EXISTS	-3	The document does not exist
RET_API_ERR_PERMISSION_DENIED	-4	have no permission
RET_API_ERR_EXCEPTION	-5	An exception is thrown
RET_API_ERR_PROPERTIES_EXISTS	-6	The attribute does not exist
RET_API_ERR_METHOD	-7	The method does not exist
RET_API_ERR_LITE	-8	The streamlined version is not supported

2. Device Information

2.1. Versions

2.1.1、 info_getApiVersion

Function name: public String info_getApiVersion()

Description: Get the current API version number

Parameter: no

Return parameter description

Parameter name	Type	Instruction
Return value	String	API version number

Example

```
SmdtManagerNew smdt = SmdtManagerNew.getInstance(this);
System.out.println(smdt.info_getApiVersion());
Output result : V1.0.0-release
```

Note: No

2.1.2、 info_getAndroidVersion

Function name: public String info_getAndroidVersion()

Description: acquire Android system version

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	Android system version

2.1.3、 info_getSoftwareVersion

Function name: public String info_getSoftwareVersion()

Description: Acquire device software version number

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	Equipment' s software version number

2.1.4、 info_getHardwareVersion

Function name: public String info_getHardwareVersion()

Description: Acquire device hardware version

Parameter: No

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	String	Equipment' s hardware version number

2.1.5、 info_getKernelVersion

Function name: public String info_getKernelVersion()

Description: Acquire device kernel version

Parameter: No

Return Parameter instruction

Parameter name	Type	Instruction
Return value	String	kernel version version

2.1.6、 info_getMCUVersion

Function name: public String info_getMCUVersion()

Description: acquire MCU version number

Parameter: No

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	String	MCU version number

2.1.7、 info_getSecurityVersion

Function name: public String info_getSecurityVersion()

Description: acquire Android security patch level

Parameter: No

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	String	security patch date

2.1.8、info_getNPUVersion

Function name: public String info_getNPUVersion()

Description: acquire NPU drive version number

Parameter: No

API version: above V2.3.0

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	String	NPU drive version number

2.1.9、info_getWebViewVersion

Function name: public String info_getWebViewVersion()

Description: acquire WebView version

API version: above V2.14.0

Parameter: No

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	String	WebView version

2.2、Memory

2.2.1、info_getTotalMemory

Function name: `public String info_getTotalMemory()`

Description: Acquire device total memory capacity

Parameter: No

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	String	total memory capacity

Note

The unit of return:

- Rest more than 1G: return GB
- Rest less than 1G: return MB

2.2.2、info_getAvailMemory

Function name: `public String info_getAvailMemory()`

Description: Acquire device remaining available memory capacity

Parameter: No

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	String	remaining available memory capacity

Note

Return unit:

- More than 1G:return GB
- Between 1G and 1M:return MB
- Between 1M and 1KB:return KB
- Less than 1KB:return B

2.2.3、info_getTotalStorage

Function name: `public String info_getTotalStorage()`

Description: Acquire device total storage capacity

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	Equipment total storage capacity

Note

- Return unit: GB

2.2.4、info_getAvailStorage

Function name: `public String info_getAvailStorage()`

Description: Acquire device remaining available storage capacity

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	remaining available storage capacity

Note

Return unit:

- More than 1G:return GB
- Between 1G and 1M:return MB
- Between 1M and 1KB:return KB
- Less than 1KB:return B

2.2.5、info_getAppUsedMemory

Function name: `public String info_getAppUsedMemory()`

Description: acquire the memory occupation of APP

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	the memory occupation of APP

Note

- Unit: MB

2.3、Model Number

2.3.1、info_getBoardType

Function name: `public String info_getBoardType()`

Description: acquire board model number

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	Board type number

2.3.2、info_getModel

Function name: `public String info_getModel()`

Description: Acquire device board number

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	Equipment model number

2.3.3、info_getFactoryCompany

Function name: `public String info_getFactoryCompany()`

Description: Acquire device producer

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	Equipment producer

2.4、Sequence Number

2.4.1、info_getSerialNumber

Function name: `public String info_getSerialNumber()`

Description: acquire sequence number

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	sequence number

2.5、CPU

2.5.1、info_getCpuTemperature

Function name: `public String info_getCpuTemperature()`

Description: Acquire device CPU temperature

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	CPU current temperature

Note

- Unit: °C

2.5.2、info_getCpuFrequency

Function name: `public String info_getCpuFrequency()`

Description: Acquire device CPU frequency

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
----------------	------	-------------

Return value	String	CPU frequency
--------------	--------	---------------

Note

- Unit:GHZ

2.5.3、info_getCpuUsage

Function name: `public String info_getCpuUsage()`

Description: Acquire device CPU usage ratio

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	CPU usage ratio

Note

- Unit:%

2.6、Face recognition

2.6.1、info_getFaceDetectSupport

Function name: `public int info_getFaceDetectSupport()`

Description: acquire whether face recognition is supported

Parameter: No

Return Parameter instruction

Parameter name	Type	instruction
Return value	int	1:support 0:Not support

2.7、Others

2.7.1、info_getDeviceTemperature

Function name: `public float info_getDeviceTemperature()`

Description: Acquire devicetemperature

API version: above V2.11.0

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	float	CPU current temperature

Note

- Unit: °C

3、Display Management

3.1、Display Screen

3.1.1、disp_setDispParams

Function name: public int disp_setDispParams(String params)

Description: Set the screen display parameter:

API version: above V2.9.0

Simplified version: Not support

Parameter:

Parameter name	Type	instruction
params	String	Screen parameter, refer to splicing alphabetic strings inputted by cfg file which is equipped with screen parameters, use ";" to separate parameters

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling the result refers to the error codes.

Example:

```
SmdtManagerNew smdt = SmdtManagerNew.getInstance(this);
String params = "cpu_id=3568;disp_main=lvds;disp_aux=hdmi;";
int result = smdt.disp_setDispParams(params);
```

Note

- Entering null means cancel the setting and restore the default screen parameters.

3.1.2、 disp_getDispParams

Function name: public int disp_getDispParams()

Description: acquire screen display parameters

API version: above V2.9.0

Simplified version Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	Screen display parameters:

Example:

```
SmdtManagerNew smdt = SmdtManagerNew.getInstance(this);
String params = smdt.disp_getDispParams();
System.out.println(params);
Output result: "cpu_id=3568;disp_main=lvds;disp_aux=hdmi;"
```

Note

- The output is dull by default, and the screen parameters is not configured by external equipment.

3.1.3、 disp_setDisplayOverScan

Function name: public int disp_setDisplayOverScan(int screen_id, String direction, int value)

Description: Set the display area (screen zoom)

Simplified version Not support

Parameters:

Parameter name	Type	instruction
screen_id	int	Screen ID 0:main screen 1:vice screen

direction	String	Direction: x:X axle(Right and left at the same time) y:Y axle (Up and down at the same time) left:left right:right top:up bottom:down all:all sides
value	int	Distance from the boundary

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling the result refers to the error codes

Note

- Set from 80 to 100
- The main and secondary screens depend on the system and screen parameters, not the number of screens. For example, set the main screen LVDS and the secondary screen HDMI. If you connect one HDMI screen, the HDMI is also the secondary screen, and the Parameter to be transmitted is also one secondary screen

3.1.4、 disp_getDisplayOverScan

Function name: public int[] disp_getDisplayOverScan(int screen_id)

Description: acquire display area value (Screen zoom)

Simplified version Not support

Parameter:

Parameter name	Type	instruction
screen_id	int	Screen ID 0:main screen 1:secondary screen

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int[]	An array of four side distances (member order 0 left, 1 right, 2 top, 3 bottom)

Note

- Set from 80 to 100

- The main and secondary screens depend on the system and screen parameters, not the number of screens. For example, set the main screen LVDS and the secondary screen HDMI. If you connect one HDMI screen, the HDMI is also the secondary screen, and the parameter to be transmitted is also one secondary screen

3.1.5、disp_getScreenWidth

Function name: `public int disp_getScreenWidth(int screen_id)`

Description: acquire the specified display resolution-width

Parameter:

Parameter name	Type	instruction
screen_id	int	Screen ID 0:main screen 1:secondary screen

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	width

Note

- The main and secondary screens depend on the system and screen parameters, not the number of screens. For example, set the main screen LVDS and the secondary screen HDMI. If you connect one HDMI screen, the HDMI is also the secondary screen, and the parameter to be transmitted is also one secondary screen.

3.1.6、disp_getScreenHeight

Function name: `public int disp_getScreenHeight(int screen_id)`

Description: acquire the specified display resolution-height

Parameter:

Parameter name	Type	instruction
screen_id	int	Screen ID 0:main screen 1:secondary screen

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	height

Note

- The main and secondary screens depend on the system and screen parameters, not the number of screens. For example, set the main screen LVDS and the secondary screen HDMI. If you connect one HDMI screen, the HDMI is also the secondary screen, and the parameter to be transmitted is also one secondary screen.

3.2、Backlight

3.2.1、disp_setLcdBackLight

Function name: public int disp_setLcdBackLight(int screen_id, int brightness, int frequency, boolean save)

Description: set the lightness of the screen backlight

Parameter:

Parameter name	Type	instruction
screen_id	int	0 the first backlight: system defaulted backlight 1 the second backlight:external PWM (MCU) interface
brightness	int	Lightness value
frequency	int	When setting the external PWM, please lead in frequency
save	boolean	conserve lightness value in data bank

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling result refers to error codes

- Note

- Frequency parameters:are used in external PWM interface, which can firstly acquire defualted frequency to directly set.
- In some platforms,the second backlight (such as; RK3568) does not pass the external PWM(MCU) interface. It passes the system defaulted interface which doesn' t need to lead in frequency.Although lead in frequency, it would not change.
- Before setting others,please acquire corresponding the maximum and minimum of lightness value
- When setting systematic defaulted interface, it' s recommended that using parameter “save” to conserve date in data bank after operation to reduce the times of operation.

3.2.2、 disp_getLcdBackLight

Function name: public int disp_getLcdBackLight(int screen_id)

Description: acquire the lightness of the backlight

Parameter:

Parameter name	Type	instruction
screen_id	int	0 the first backlight: system defaulted backlight 1 the second backlight:external PWM (MCU) interface

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Backlight lightness value

3.2.3、 disp_getLcdPwmFrequency

Function name: public int disp_getLcdPwmFrequency(int screen_id)

Description: acquire backlight frequency

Parameter:

Parameter name	Type	instruction
screen_id	int	0 the first backlight: system defaulted backlight 1 the second backlight:external PWM (MCU) interface

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	backlightfrequency

Note

- Platforms after 11.0 need not to set the acquisition of frequency, interface is in vain
- Backlight frequency is usually used in the setting of PWM interface backlight to lead in, Not support separate setting
- It is usually used to acquire the frequency of external PWM, for instance, systematic defaulted interface does not support acquisition. The two ways of RK3568 are systematic defaulted interface.

3.2.4、 disp_getLcdBackLightMaxMin

Function name: public int disp_getLcdBackLightMaxMin(int screen_id, String type)

Description: acquire the maximum or minimum of lightness value

Parameter:

Parameter name	Type	instruction
screen_id	int	0 the first backlight: system defaulted backlight 1 the second backlight:external PWM (MCU) interface
type	int	max:maximum value min:minimum value

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	acquire the maximum or minimum of lightness value

3.2.5、 disp_setLcdBackLightEnable

Function name: public int disp_setLcdBackLightEnable(int screen_id, boolean enable)

Description: set the switch of screen backlight

API version: above V2.16.0

Parameter:

Parameter name	Type	instruction
screen_id	int	0:the first backlight 1:the second backlight
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

3.2.6、 disp_getLcdBackLightEnable

Function name: public int disp_getLcdBackLightEnable(int screen_id)

Description: acquire the switch status of backlight

API version: above V2.16.0

Parameter:

Parameter name	Type	instruction
screen_id	int	0:the first backlight 1:the second backlight

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	int	1:open 0:close

3.3、 systemUI

3.3.1、 disp_setDisplayDensity

Function name: public int disp_setDisplayDensity(int value)

Description: set the display density of DPI

Simplified version Not support

Parameter:

Parameter name	Type	Instruction
value	int	DPI value

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	int	Calling results refers to error codes

Note

- The setting of DPI value should ≥ 72

3.3.2、disp_getDisplayDensity

Function name: public int disp_getDisplayDensity()

Description: acquire the current display density of DPI

Simplified version Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	DPI value

3.3.3、disp_getScreenShotBitmap

Function name: public Bitmap disp_getScreenShotBitmap()

Description: take a screenshot and return bitmap format picture

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	Bitmap	Picture data

3.3.4、disp_getScreenShot

Function name: public int disp_getScreenShot(String filepath)

Description: Take a screenshot and save it to the specified path

Parameter:

Parameter name	Type	instruction
filepath	int	The saving path of pictures

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- Save pictures by PNG format
- An automatic default path is saved when the message is sent: under the root directory Screenshots

3.3.5、 disp_setDisplayRotation

Function name: public int disp_setDisplayRotation(int screen_id, int degree)

Description: Set the screen rotation angle

Parameter:

Parameter name	Type	instruction
screen_id	int	Screen ID 0:main screen 1:secondary screen
degree	int	rotation angle: 0/90/180/270

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- Only support for angles: 0/90/180/270
- The main and secondary screens depend on the system and screen parameters, not the number of screens. For example, set the main screen LVDS and the secondary screen HDMI. If you connect one HDMI screen, the HDMI is also the secondary screen, and the Parameter to be transmitted is also one secondary screen

3.3.6、 disp_getDisplayRotation

Function name: public int disp_getDisplayRotation(int screen_id)

Description: acquire the rotation angel of screen

Parameter:

Parameter name	Type	instruction
screen_id	int	Screen ID 0:main screen 1:secondary screen

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Rotation angel:0/90/180/270

Note

- The main and secondary screens depend on the system and screen parameters, not the number of screens. For example, set the main screen LVDS and the secondary screen HDMI. If you connect one HDMI screen, the HDMI is also the secondary screen, and the Parameter to be transmitted is also one secondary screen

3.3.7、disp_setBootLogo

Function name: public int disp_setBootLogo(String filepath)

Description: replace the switch Logo

API version: above V2.8.0

Parameter:

Parameter name	Type	instruction
filepath	String	File path

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- Not support oversized pictures and 4K resolution
- If the file path is null, the setting will be canceled and the default will be restored
- File requirements: bmp format image, bit depth 8 (Allwinner platform 24bits)

Setting files that do not meet the requirements can easily lead to system exceptions

3.3.8、disp_getBootLogo

Function name: public int disp_getBootLogo()

Description: acquire the file path of start Logo

API version: above V2.8.0

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	the file path of start logo:If not null, it is custom; if null, it is default

3.3.9、disp_setBootAnimation

Function name: public int disp_setBootAnimation(String filepath)

Description: replace the start logo

API version: above V2.8.0

Parameter:

Parameter name	Type	instruction
Filepath	String	File path

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- If the file path is null, the setting will be canceled and the default will be restored
- File requirement:animation package which is made according to Android requirement,zip format.

- Setting files that do not meet the requirements can easily lead to system exceptions.

3.3.10、disp_getBootAnimation

Function name: public int disp_getBootAnimation()

Description: acquire the file path of smart animation file

API version: above V2.8.0

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	the file path of start logo:If not null, it is custom; if null, it is default

3.3.11、disp_setStatusBar

Function name: public int disp_setStatusBar(boolean enable)

Description: Set the status bar to display the status

Parameter:

Parameter name	Type	instruction
enable	boolean	true:display false:hide

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

3.3.12、disp_getStatusBar

Function name: public int disp_getStatusBar()

Description: acquire the display status of the current status bar

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	0:display 1:hide

3.3.13、 disp_setNavigationBar

Function name: public int disp_setNavigationBar(boolean enable)

Description: set the display status of navigation bar

Parameter:

Parameter name	Type	instruction
enable	boolean	true:display false:hide

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

3.3.14、 disp_getNavigationBar

Function name: public int disp_getNavigationBar()

Description: acquire the display status of navigation bar

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:display 0:hide

3.3.15、 disp_setStatusBarDrag

Function name: public int disp_setStatusBarDrag(boolean enable)

Description: Set whether the status bar can be dragged down

Parameter:

Parameter name	Type	instruction
enable	boolean	true:enable false:forbid

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

3.3.16、 disp_getStatusBarDrag

Function name: public int disp_getStatusBarDrag()

Description: acquire whether the status bar can be dragged down

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:enable 0:forbid

3.3.17、 disp_setGestureBar

Function name: public int disp_setGestureBar(boolean enable)

Description: 设置手势能否拉出导航栏

Parameter:

Parameter name	Type	instruction
enable	boolean	true:enable false:forbid

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

3.3.18、 disp_getGestureBar

Function name: public int disp_getGestureBar()

Description: acquire whether gestures can get the navigation bar
Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:enable 0:forbid

3.3.19、 disp_setSystemUIMode

Function name: `public int disp_setSystemUIMode(boolean enable)`

Description: set the mode of SystemUI

Parameter:

Parameter name	Type	instruction
enable	boolean	true:OS model false: original android

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- After the setting, the navigation bar needs to be hidden and then displayed again for changes
- In OS mode, the status bar does not support displays notifications

3.3.20、disp_getSystemUIMode

Function name: `public int disp_getSystemUIMode()`

Description: acquire the current SystemUI mode

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	true:OS model false: original android

3.3.21、disp_addAppLauncherHideList

Function name: `public int disp_addAppLauncherHideList(String packageName)`

Description: Add apps that need to hide icons

Simplified version: Not support

Parameter:

Parameter name	Type	instruction
----------------	------	-------------

packageName	String	Application package name
-------------	--------	--------------------------

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- Restart to take effect after setting

3.3.22、disp_getAppLauncherHideList

Function name: public List disp_getAppLauncherHideList()

Description: acquire the application that has been set with a hidden icon

Simplified version:Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	All set hidden APP package names

3.3.23、disp_delAppLauncherHideList

Function name: public int disp_delAppLauncherHideList(String packageName)

Description: Remove apps that need to hide icons

Simplified version Not support

Parameter:

Parameter name	Type	instruction
packageName	String	Application package name

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- Lead in 'clean' to delete app launch hide list

3.4、HDMI

3.4.1、disp_getHdmiInStatus

Function name: `public int disp_getHdmiInStatus()`

Description: acquire hdmi in status value

Simplified version Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:connected 0:disconnect

Note

- interface is not supported.

3.4.2、disp_setHdmiInAudio

Function name: `public int disp_setHdmiInAudio(boolean enable)`

Description: set the audio switch of hdmi in

Simplified version:Not support

Parameter:

Parameter name	Type	instruction
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- Interface is not supported.

3.4.3、disp_getHdmiInAudio

Function name: public int disp_getHdmiInAudio

Description: acquire the audio switch status of hdmi in

Simplified version:Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

Note

- Interface is not supported.

3.4.4、disp_setHdmiOutStatus

Function name: public int disp_setHdmiOutStatus(boolean enable)

Description: set the switch status of hdmi out

API version: above V2.13.0

Simplified version Not support

Parameter:

Parameter name	Type	Instruction
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	int	Calling results refers to error codes

3.4.5、disp_getHdmiOutStatus

Function name: public int disp_getHdmiOutStatus()

Description: acquire the switch status of hdmi out

API version: above V2.13.0

Simplified version:Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

4、 Network Management

4.1、 Information

4.1.1、 net_getMacAddress

Function name: public String net_getMacAddress(String type)

Description: Get the MAC address of the device Ethernet.

Parameter:

Parameter name	Type	instruction
type	String	Internet Type eth0:Ethernet eth1:Ethernetlwan0:WIFI

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	MAC address

Note

- API versionV2.12.0+available constant class
- The acquired WiFi MAC address is the MC address of this device

4.1.2、 net_getCurrentNetType

Function name: public String net_getCurrentNetType()

Description: acquire the current connected Internet type

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	WIFI:WIFI ETH:ethernet MOBILE:mobile network UNKNOWN:unknown Type

Note

- API versionV2.12.0+ available constant class

4.1.3、net_getNetWorkInf

Function name: public NetworkInfoData net_getNetWorkInf(String type)

Description: acquire the current connected internet type

Parameter:

Parameter name	Type	instruction
type	String	Network Type eth0:Ethernet eth1:Ethernet wlan0:WIFI

Return Parameter Instruction

Parameter name	Type	instruction
Return value	NetworkInfoData	An object that saves network information

Example

```
SmdtManagerNew smdt = SmdtManagerNew.getInstance(this);
NetworkInfoData mNetworkInfoData = smdtManagerNew.net_getNetWorkInf("eth0");
if (mNetworkInfoData != null) {
    String ip = mNetworkInfoData.getIp();
    String gateway = mNetworkInfoData.getGateway();
    String netmask = mNetworkInfoData.getNetmask();
    String dns1 = mNetworkInfoData.getDns1();
    String dns2 = mNetworkInfoData.getDns2();
}
```

Note

- API versionV2.12.0+ available constant class
- Look up the network information object class for more details

4.1.4、net_getWifiRssi

Function name: public int net_getWifiRssi(int level)

Description: acquire the connecting signal of WIFI

Parameter:

Parameter name	Type	instruction
level	int	Classification, such as: 5 copies are divided into 5 grades

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	The classification of the current signal

4.1.5、net_getImeiNumber

Function name: public String net_getImeiNumber()

Description: acquire IMEI number

API version: above V2.14.0

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	IMEI number

4.1.6、net_getIccidNumber

Function name: public String net_getIccidNumber()

Description: acquire ICCID number

API version: above V2.14.0

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	ICCID number

4.1.7、net_getIccidNumber

Function name: public String net_getImsiNumber()

Description: acquire IMSI number

API version: above V2.14.0

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	IMSI number

4.2、Setting

4.2.1、net_setNetWork

Function name: public int net_setNetWork(String type, boolean enable)

Description:Set the network switch status

Parameter:

Parameter name	Type	instruction
type	String	Network Type eth0:Ethernet eth1:Ethernet1 wlan0:WIFI mobile:mobile network
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- API versionV2.12.0+ available constant class

4.2.2、net_getNetWork

Function name: public int net_getNetWork(String type)

Description: acquire the network switch status

Parameter:

Parameter	Type	instruction
-----------	------	-------------

name		
type	String	Network Type eth0:Ethernet eth1:Ethernet1 wlan0:WIFI mobile:mobile network

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

Note

- API versionV2.12.0+ available constant class

4.2.3、net_setNetWorkModel

Function name: public int net_setNetWorkModel(String type, int model, String ip, String gaw, String mask, String dns1, String dns2)

Description:Set the network connection model

Parameter:

Parameter name	Type	instruction
type	String	Network Type eth0:Ethernet eth1:Ethernet1 wlan0:WIFI
model	int	Network mode; 0: dynamic 1: static
ip	String	Set the IP address in static mode
gaw	String	Set the gateway in static mode
mask	String	The static mode sets the sub-net mask
dns1	String	Set DNS1 in static mode
dns2	String	Set DNS2 in static mode

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- This interface WDI modifies the current WIFI connection. If you do not connect to WIFI, you will not be able to switch modes
- API versionV2.12.0+ available constant class

4.2.4、net_getNetWorkModel

Function name: `public int net_getNetWorkModel(String type)`

Description: acquire network connection mode

Parameter:

Parameter name	Type	instruction
type	String	Network Type eth0:Ethernet eth1:Ethernet1 wlan0:WIFI

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Network mode; 0: dynamic 1: static

Note

- API version V2.12.0+ available constant class

4.2.5、net_setWifiAp

Function name: `public int net_setWifiAp(boolean enable)`

Description: Set the WIFI hotspot switch status

Simplified version Not support

Parameter:

Parameter name	Type	instruction
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- The WIFI hotspot and the WIFI switch are mutually exclusive. The WIFI will be automatically turned off after the hotspot is opened

4.2.6、net_getWifiAp

Function name: `public int net_getWifiAp()`

Description: acquire WIFI hotspot switch status

Simplified version Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

4.2.7、net_setWifiConnect

Function name: public int net_setWifiConnect(String account, String pwd, int type, int mode, NetworkInfoData info)

Description: Set the WiFi connection account number password

API version: above V2.6.0

Parameter:

Parameter name	Type	instruction
account	String	Account name
pwd	String	password
type	int	Encryption mode 0: no password 1: WEP 2: WPA
mode	int	Connection mode 0: dynamic 1: static
info	NetworkInfoData	The connection mode is the configuration required when the connection mode is static

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- API version V2.12.0+ available constant class
- It takes effect in real time and connects to WIFI after calling it
- NetworkInfoData: Network information object class

4.2.8、net_setNetworkProtect

Function name: public int net_setNetworkProtect(boolean enable, int type, long time, String ip_internet, String ip_intranet, String log_path, boolean reboot)

Description: Set the network guard switch status

API version: above V2.7.0

Simplified version: Not support

Parameter:

Parameter name	Type	instruction
enable	boolean	true:open false:close
type	int	Type 0:automatic 1:Ethernet 2:WIFI 3:mobile net
time	long	The interval between the completion of diagnosis and the re-diagnosis
ip_internet	String	External IP address
ip_intranet	String	Internal IP address
log_path	String	Log save path
reboot	boolean	Whether to restart the network after a failure true:restart false:not restart

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- **Type:** Type is automatic by default: When the current network is detected as unavailable, it will attempt to fix all network Types
- **Time:**The default interval is 15 minutes: the interval between the diagnosis (including the repair after the problem occurs) and the diagnosis again. The interval should not be less than three minutes, and it is not recommended to set a too short time to cause frequent detection
- **ip_internet:** The default IP address of the external network is Baidu: used to detect whether the external network is communicating
- **ip_intranet:**The default IP address of the Intranet is Baidu: used to detect whether the external network is communicating

- log_path: The log saving path is in system root content/NetworkProtectLog by default
- Reboot whether to restart after network repair failure. It will restart by default. Restart: Restart system for re-diagnosis after repair failure. After multiple consecutive restarts, it will enter sleep mode, and the waiting time for each restart will be increased. Do not restart: It will enter sleep mode after repair failure.
- Sleep mode: When network changes are detected, the guardian will be restored and re-diagnosed. If the diagnosis fails, it will continue to sleep.

4.2.9、net_getNetworkProtectEnable

Function name: public int net_getNetworkProtectEnable()

Description: acquire the network guard switch status

API version: above V2.7.0

Simplified version:Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

4.2.10、net_getNetworkProtectEnable

Function name: public List net_getNetworkProtectConfig()

Description: acquire network guard configuration information

API version: above V2.7.0

Simplified version Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	List	Array order: 0:Type

	1:intervals 2:External IP address 3:Internal IP address 4:log path 5:Whether to restart the network after a failure
--	---

Example

```
SmdtManagerNew smdt = SmdtManagerNew.getInstance(this);
List<String> config = smdt.net_getNetworkProtectConfig();
int type = Integer.parseInt(config.get(0));
long time = Long.parseLong(config.get(1));
String ip_internet = config.get(2);
String ip_intranet = config.get(3);
String log_path = config.get(4);
boolean reboot = config.get(5).equals("1");
```

4.2.11、net_setNetworkPriority

Function name: public int net_setNetworkPriority(String[] types)

Description: Set network priority

API version: above V2.15.0

Simplified version Not support

Parameter:

Parameter name	Type	Instruction
types	String[]	Network priority sorting

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	int	Calling results refers to error codes

Note

- Types: Sort the networkType string from high to low, such as: new String[]{"eth0", "wlan0", "mobile"}; Ethernet/WIFI/mobile network
- Also can use new String[]{ TYPE_ETH0, TYPE_WLAN, TYPE_MOBILE};
- Restart to take effect after setting

4.2.12、net_getNetworkPriority

Function name: `public String[] net_getNetworkPriority()`

Description: acquire network priority

API version: above V2.15.0

Simplified version:Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String[]	Network priority sorting

Note

- Return value:Sort the networkType string from high to low, for example: `new String[]{"eth0", "wlan0", "mobile"};` Ethernet/WIFI/mobile network

4.2.13、net_setNetworkMultiEnable

Function name: `public int net_setNetworkMultiEnable(boolean enable)`

Description: Set the switch status of multi-network coexistence

API version: above V2.16.0

Simplified version Not support

Parameter:

Parameter name	Type	instruction
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- network priority is used in cooperation, and the network type for accessing the Internet is placed first

- Restart to take effect after setting

4.2.14、net_getNetworkMultiEnable

Function name: public int net_getNetworkMultiEnable()

Description: acquire Multi-network coexistence switch status

API version: above V2.16.0

Simplified version: Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

5、System Control

5.1、Setting

5.1.1、sys_setBluetooth

Function name: public int sys_setBluetooth(boolean enable)

Description: set the switch status of Bluetooth

Simplified version Not support

Parameter:

Parameter name	Type	instruction
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.1.2、sys_getBluetooth

Function name: public int sys_getBluetooth()

Description: acquire the switch status of Bluetooth

Simplified version Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

5.1.3、sys_setHwStack

Function name: public int sys_setHwStack(boolean enable)

Description: Set the switch status of HW overlay layer

Simplified version Not support

Parameter:

Parameter name	Type	instruction
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.1.4、sys_getHwStack

Function name: public int sys_getHwStack()

Description: acquire the switch status of HW stack

Simplified version Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

5.1.5、sys_setAirPlane

Function name: `public int sys_setAirPlane(boolean enable)`

Description: set the switch status of Air Plane

Parameter:

Parameter name	Type	instruction
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.1.6、sys_getAirPlane

Function name: `public int sys_getAirPlane()`

Description: acquire the switch status of Air Plane

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

5.1.7、sys_setLocation

Function name: `public int sys_setLocation(int mode)`

Description: set the switch status of location information

Parameter:

Parameter name	Type	instruction
mode	int	0:close 1:open 2:Power saving mode 3:high accuracy

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.1.8、sys_getLocation

Function name: public int sys_getLocation()

Description: acquire the switch status of location information

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	0:close 1:open 2:Power saving mode 3:high accuracy

5.1.9、sys_setSoftKeyboard

Function name: public int sys_setSoftKeyboard(boolean enable)

Description: set the display status of soft keyboard

Parameter:

Parameter name	Type	instruction
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.1.10、sys_getSoftKeyboard

Function name: public int sys_getSoftKeyboard()

Description: acquire the display status of soft keyboard

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

5.1.11、sys_setKeyReport

Function name: public int sys_setKeyReport(boolean enable)

Description: Set whether the button can be reported

Parameter:

Parameter name	Type	instruction
enable	boolean	true:enable false:forbid

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.1.12、 sys_getKeyReport

Function name: public int sys_getKeyReport()

Description: acquire whether the button can be reported

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:enable 0:forbid

5.1.13、 sys_setTouchReport

Function name: public int sys_setTouchReport(boolean enable)

Description: set whether the touch can be reported

Parameter:

Parameter name	Type	instruction
enable	boolean	true:enable false:forbid

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.1.14、 sys_getTouchReport

Function name: public int sys_getTouchReport()

Description: acquire whether the touch can be reported

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:enable 0:forbid

5.1.15、sys_copyFile

Function name: public void sys_copyFile(String oldpath, String newpath, CopyCallback callback)

Description: copy file

Parameter:

Parameter name	Type	instruction
oldpath	String	Source path
newpath	String	Target path
callback	CopyCallback	Copy status callback

Return Parameter instruction: No

Example

```
SmdtManagerNew smdt = SmdtManagerNew.getInstance(this);
smdt.sys_copyFile(oldpath, newpath, new SmdtManagerNew.CopyCallback
() {
    @Override
    public void onCopyProgress(int progress) throws RemoteException {
        ...
    }

    @Override
    public void onCopyFinished(int returnCode, String msg) throws Rem
oteException {
        ...
    }
});
```

Note

- Callback interface: [CopyCallback](#)

5.1.16、sys_setDefInputMethod

Function name: public int sys_setDefInputMethod(String default_input_method)

Description: Set the defaulted input method

Parameter:

Parameter name	Type	instruction
default_input_method	String	The package name and class name of the input method, such as:com.android.inputmethod.pinyin/.LatinIME

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- It is recommended to set the acquire input method list first.

5.1.17、sys_getDefInputMethod

Function name: public String sys_getDefInputMethod()

Description: acquire the current defaulted input method

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	The package name and class name of the input method, such as:com.android.inputmethod.pinyin/.LatinIME

5.1.18、sys_getDefInputMethodList

Function name: public String[] sys_getDefInputMethodList()

Description: acquire the overall input method

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	The package name and class name of the input method

5.1.19、 sys_setSystemFontSize

Function name: public int sys_setSystemFontSize(float size)

Description: set system font size

Simplified version Not support

Parameter:

Parameter name	Type	Instruction
size	float	Font size, such as: 0.85/1.00/1.15/1.30

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	int	Calling results refers to error codes

5.1.20、 sys_getSystemFontSize

Function name: public float sys_getSystemFontSize()

Description: acquire current system font size

Simplified version:Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	float	Font size, such as: 0.85/1.00/1.15/1.30

5.2、 Time and Language

5.2.1、 sys_setNtpServer

Function name: public int sys_setNtpServer(String url)

Description: Set the address of the NTP synchronization time server

Parameter:

Parameter name	Type	Instruction
----------------	------	-------------

url	String	Server url, such as:"time.nist.gov"
-----	--------	-------------------------------------

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	int	Calling results refers to error codes

5.2.3、sys_setNationallanguage

Function name: public int sys_setNationallanguage(String national, String language)

Description: set country and language

Parameter:

Parameter name	Type	Instruction
national	String	Country/region, such as:CN
language	String	Language, such as: zh

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	int	Calling results refers to error codes

Note

- Imported Parameter: Errors may cause system system exceptions. Use with caution.
- It is recommended to traverse the system language acquire to the corresponding country or region and language generation number first, and then call the interface to set it

5.2.2、sys_getNtpServer

Function name: public String sys_getNtpServer()

Description: acquire the current address of the NTP synchronized time server

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	Server address: url

5.2.4、sys_getNationallanguage

Function name: public String sys_getNationallanguage()

Description: acquire the current country and language

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	Country and language, such as: zh_CN

Note

- Return value zh_CN: country- CN, language- zh

5.2.5、sys_setTimeZone

Function name: public int sys_setTimeZone(String timeZone)

Description: set the current time zone

Parameter:

Parameter name	Type	Instruction
timeZone	String	Time zone: such as: Asia/Shanghai

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.2.6、sys_getTimeZone

Function name: public String sys_getTimeZone()

Description: acquire the current time zone

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	Current time zone, such as: Asia/Shanghai

5.2.7、sys_setTimeFormat

Function name: public int sys_setTimeFormat(String format)

Description: Set the system time to the default format

Parameter:

Parameter name	Type	instruction
format	String	Time format 12:12hours 24:24hours

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.2.8、sys_getTimeFormat

Function name: public String sys_getTimeFormat()

Description: acquire system time default format

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	12:12hours 24:24hours

5.2.9、sys_setTime

Function name: public int sys_setTime(long time_millisecond)

Description: set the current system time

Parameter:

Parameter name	Type	instruction
time_millisecond	long	Time, unit: millisecond

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- Before setting, please turn off the system automatic synchronization time

5.2.10、sys_setNetworkTimeSync

Function name: public int sys_setNetworkTimeSync(boolean enable)

Description: Set whether to synchronize network time

Parameter:

Parameter name	Type	instruction
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.2.11、sys_getNetworkTimeSync

Function name: public int sys_getNetworkTimeSync()

Description: acquire open status of synchronize network time

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

5.3、Application

5.3.1、sys_setDefaultLauncher

Function name: public int sys_setDefaultLauncher(String packageName)

Description: Set the default desktop application

Parameter:

Parameter name	Type	instruction
packageName	String	Set the package name and class name of the desktop application that needs to be set (with "/" in the middle)

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- The class name must be attached, such as: android.app.smdt.launcher/.Launcher

5.3.2、 sys_getDefaultLauncher

Function name: public String sys_getDefaultLauncher()

Description: acquire the default desktop application

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	Application package name and class name

5.3.3、 sys_setSystemBootTestApp

Function name: public int sys_setSystemBootTestApp(String packageName)

Description: Set the application that needs to start automatically

Parameter:

Parameter name	Type	instruction
packageName	String	Application package name (add "/" to the middle of the class name when you need to specify the class name)

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- If the application is set without a desktop icon, you need to specify the class name
- For example: Specify the package name android.app.smdt.apidemo; Specify the class name android.app.smdt.apidemo/.MainActivity

5.3.4、sys_getSystemBootApplication

Function name: public String sys_getSystemBootApplication()

Description: acquire the application that needs to start automatically

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	Application package name

5.3.5、sys_setDaemonsActivity

Function name: public int sys_setDaemonsActivity(String packageName, long time_millisecond, boolean broadcast_enable)

Description:Set up the daemon

Parameter:

Parameter name	Type	instruction
packageName	String	Set the application package name that needs to be guarded (add "/" after the class name when the class name needs to be specified)
time_millisecond	long	The time it takes to restart the application after it exits the front end, its unit is milliseconds
broadcast_enable	boolean	Whether a global broadcast is required when the application exits.

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- If the application is set without a desktop icon, you need to specify the class name
- For example: Specify the package name
android.app.smdt.apidemo; Specify the class name
android.app.smdt.apidemo/.MainActivity
- Global broadcast is "android.app.smdt.PROTECT_CHECK"

5.3.6、 sys_getDaemonsActivity

Function name: public String sys_getDaemonsActivity()

Description: acquire the current guarding application package name

Parameter: No **Return Parameter Instruction**

Parameter name	Type	instruction
Return value	String	guarding application package name

5.3.7、 sys_doSilentInstallApp

Function name: public void sys_doSilentInstallApp(String apkPath, InstallCallback callback)

Description: silent APP installation

Parameter:

Parameter name	Type	instruction
apkPath	String	The path where the application to be installed is located
callback	InstallCallback	Install the callback

Return Parameter instruction: No

Example

```
SmdtManagerNew smdt = SmdtManagerNew.getInstance(this);
smdt.sys_doSilentInstallApp(path, new SmdtManagerNew.InstallCallback
() {
    @Override
    public void onInstallFinished(String packageName, int returnCode,
String msg) throws RemoteException {
        ...
    }
});
```

Note

- Callback interface: [InstallCallback](#)

5.3.8、sys_doSilentUninstallApp

Function name: public void sys_doSilentUninstallApp(String packageName, DeleteCallback callback)

Description: silent app uninstall

Parameter:

Parameter name	Type	Instruction
packageName	String	The application package name that needs to be uninstalled.
callback	DeleteCallback	Uninstall callback

Return Parameter Instruction: No

Example

```
SmdtManagerNew smdt = SmdtManagerNew.getInstance(this);
smdt.sys_doSilentUninstallApp(pkg, new SmdtManagerNew.DeleteCallback
() {
    @Override
    public void onDeleteFinished(String packageName, int returnCode,
String msg) throws RemoteException {
        ...
    }
});
```

Note

- Callback interface: [DeleteCallback](#)

5.3.9、sys_addBlackWhiteList

Function name: public int sys_addBlackWhiteList(String packageName, int type, int function)

Description: add application in black and white list

Simplified version:Not support

Parameter:

Parameter name	Type	instruction
packageName	String	Application package name
type	int	Type 0:black list 1:white list
function	int	Function 0:uninstall 1:install

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- API versionV2.12.0+ available constant class
- When the black and white lists are null, all applications are allowed to install/uninstall
- When the whitelist is not null, the whitelist application can be installed/uninstalled, and other applications cannot be installed/uninstalled
- When the whitelist is null and the blacklist is not null, the blacklist application cannot be installed/uninstalled, while other applications can be installed/uninstalled.
- When both the white list and the black list are not empty, the white list takes effect and the black list does not take effect.

5.3.10、sys_getBlackWhiteList

Function name: public List sys_getBlackWhiteList(int type, int function)

Description: acquire black and white list

Simplified version Not support

Parameter:

Parameter name	Type	instruction
type	int	Type 0:black list 1:white list
function	int	Function 0:uninstall 1:install

Return Parameter Instruction

Parameter name	Type	instruction
Return value	List	Application package name

Note

- API versionV2.12.0+ available constant class
- When the black and white lists are null, all applications are allowed to install/uninstall

- When the whitelist is not null, the whitelist application can be installed/uninstalled, and other applications cannot be installed/uninstalled
- When the whitelist is null and the blacklist is not null, the blacklist application cannot be installed/uninstalled, while other applications can be installed/uninstalled.
- When both the white list and the black list are not empty, the white list takes effect and the black list does not take effect.

5.3.11、sys_delBlackWhiteList

Function name: public int sys_delBlackWhiteList(String packageName, int type, int function)

Description: delete application from black and white list

Simplified version Not support

Parameter:

Parameter name	Type	instruction
packageName	String	Application package name
type	int	Type 0:black list 1:white list
function	int	Function 0:uninstall 1:install

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- API versionV2.12.0+ available constant class
- Special value of application package : "clean"means to clear all data; "clean_type" clears the corresponding Type list in conjunction with the incoming type."clean_funtion" means to clear the corresponding function list in conjunction with incoming function."clean_type_function" means to clear the corresponding type and function list in conjunction with the incoming type and function.

5.3.12、sys_backupApplication

Function name: public void sys_backupApplication(String packageName, BackUpCallback callback)

Description: backup application data

Simplified version Not support

Parameter:

Parameter name	Type	instruction
packageName	String	Application package name
callback	BackUpCallback	Backup callback

Return Parameter instruction:No

Example

```
SmdtManagerNew smdt = SmdtManagerNew.getInstance(this);
smdt.sys_backupApplication(packageName, new SmdtManagerNew.BackUpCallback() {
    @Override
    public void onBackUpProgress(int progress) throws RemoteException
    {
        ...
    }

    @Override
    public void onBackUpFinished(int returnCode, String msg) throws RemoteException {
        ...
    }

    @Override
    public void onBackUpPath(String path) throws RemoteException {
        ...
    }
});
```

Note

- Callback interface: [BackUpCallback](#)
- Condition:Backup based on existent APP/ need root
- Can backup: /data/data/package name /data/user/0/package name/data/user_de/0/package name/Android/data/package name

5.3.13、sys_recoveryApplication

Function name: public void sys_recoveryApplication(String packageName, RecoveryCallback callback)

Description: recover application data

Simplified version: Not support

Parameter:

Parameter name	Type	instruction
packageName	String	Application package name
callback	RecoveryCallback	Recovery callback

Return Parameter instruction:No

Example

```
SmdtManagerNew smdt = SmdtManagerNew.getInstance(this);
smdt.sys_recoveryApplication(packageName, new SmdtManagerNew.Recover
yCallback() {
    @Override
    public void onRecoveryProgress(int progress) throws RemoteExcepti
on {
        ...
    }

    @Override
    public void onRecoveryFinished(int returnCode, String msg) throws
RemoteException {
        ...
    }

    @Override
    public void onRecoveryPath(String path) throws RemoteException {
        ...
    }
});
```

Note

- Recovery interface: [RecoveryCallback](#)
- Condition: recovery based on existent APP/ need root
- Can recover: /data/data/package name /data/user/0/package name /data/user_de/0/package name /Android/data/package name

5.3.14、sys_setAutoInstallEnable

Function name: public int sys_setAutoInstallEnable(boolean enable)

Description: Set whether to allow automatic installation of applications

API version: above V2.6.0

Parameter:

Parameter name	Type	instruction
enable	boolean	true:support false:not support

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.3.15、sys_getAutoInstallEnable

Function name: `public int sys_getAutoInstallEnable(boolean enable)`

Description: acquire Set whether to allow automatic installation of applications

API version: above V2.6.0

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:support 0:not support

5.3.16、sys_addAutoInstallAppList

Function name: `public int sys_addAutoInstallAppList(String path, String packageName)`

Description: Add automatic installation application policy configuration

API version: above V2.6.0

Parameter:

Parameter name	Type	instruction
path	String	the path of installation package
packageName	String	Application package name

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- The installation package path is a relative path, relative to the external storage. For example, if an external USB flash disk is inserted, the app directory is created in the root directory, and the test.apk is placed under the directory, then fill in: app/test.apk
- Multiple package name policies can be set under the same path
- If the package name is null, all policies under this path will be cleared

5.3.17、sys_getAutoInstallAppList

Function name: public List sys_getAutoInstallAppList()

Description: acquire automatic installation application policy configuration

API version: above V2.6.0

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	List	automatic installation application policy configuration list

Example

```
SmdtManagerNew smdt = SmdtManagerNew.getInstance(this);
List list = smdt.sys_getAutoInstallAppList();
if (list != null && list.size() > 0) {
    for (int i = 0; i < list.size(); i++) {
        String[] inf = list.get(i).split(",");
        String path = inf[0];
        String packageName = "";
        if (inf.length > 1) {
            packageName = inf[1];
        }
    }
}
```

Note

- Return list members are composed of path + "," + package name. For example: path root directory app.apk, package

name android.app.smdt.apidemo, then acquired values are :
app.apk, android.app.smdt.apidemo

5.3.18、sys_delAutoInstallAppList

Function name: public int sys_delAutoInstallAppList(String path, String packageName)

Description: delete automatic installation application policy configuration

API version: above V2.6.0

Parameter:

Parameter name	Type	instruction
path	String	The path of installation package
packageName	String	Application package name

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- The special value of installation package path: "clean" means to clear all data;
- When the name of installation package is null which means to delete all policy configuration under the path.

5.4、Audio

5.4.1、sys_setVolume

Function name: public int sys_setVolume(int value)

Description: set the volume of systematic media audio

Parameter:

Parameter name	Type	instruction
value	int	Volume value

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- Setting after acquiring the maximum and minimum volume of the audio

5.4.2、 sys_getVolume

Function name: public int sys_getVolume()

Description: acquire the volume of systematic media audio

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Volume value

5.4.3、 sys_getVolumeMaxMin

Function name: public int sys_getVolumeMaxMin(String type)

- **Description:** acquire the maximum and minimum volume of the media audio

Parameter:

Parameter name	Type	instruction
type	String	max:maximum value min:minimum value

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Maximum volume value /minimum volume value

5.4.4、 sys_setVolumeMute

Function name: public int sys_setVolumeMute(boolean enable)

Description: set the system media audio mute

Parameter:

Parameter name	Type	instruction
enable	boolean	true:mute false:not mute

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.4.5、sys_getVolumeMute

Function name: public int sys_getVolumeMute()

Description: acquire whether system media audio is mute

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:mute 0:not mute

5.4.6、sys_setOutVolume

Function name: public int sys_setOutVolume(int type, int value)

Description: set the increase of output volume of the device

API version: above V2.16.0

Simplified version:Not support

Parameter:

Parameter name	Type	instruction
type	int	Equipment Type 1:speaker sound 2:headphone sound 3:HDMI sound
value	int	Set the volume increase value range: 0<=value<=99

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- Only speaker gain is supported currently.

5.4.7、sys_getOutVolume

Function name: public int sys_getOutVolume(int type)

Description: acquire the increase of output volume of the device

API version: above V2.16.0

Simplified version: Not support

Parameter:

Parameter name	Type	instruction
type	int	Equipment type 1:speaker sound 2:headphone speaker 3:HDMI sound

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Volume increase: 0-99

Note

- Only speaker gain is supported currently.

5.4.8、sys_setAudioOutput

Function name: public int sys_setAudioOutput(int type)

Description: Set the audio output type

API version: above V2.16.0

Simplified version: Not support

Parameter:

Parameter name	Type	instruction
type	int	Audio output type 0: automatic 1: speaker 2: headphone 3: speaker+headphone 4: hdmi 5: speaker+HDMI

		6: headphone+HDMI 7: speaker+headphone+HDMI
--	--	--

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- Due to hardware , when selecting speaker-related configurations on some boards, the headset also outputs sound.

5.4.9、sys_getAudioOutput

Function name: public int sys_getAudioOutput()

Description: acquire the audio output type

API version: above V2.16.0

Simplified version: Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Audio output type 0: automatic 1: speaker 2: headphone 3: speaker+headphone 4: hdmi 5: speaker+HDMI 6: headphone+HDMI 7: speaker+headphone+HDM

5.4.10、sys_setAudioInput

Function name: public int sys_setAudioInput(int type)

Description: set the audio input type

API version: above V2.16.0

Simplified version Not support

Parameter:

Parameter name	Type	instruction
type	int	Audio input type 0: automatic 1: microphone 2: headphone

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- This interface is not supported for the time being.

5.4.11、 sys_getAudioInput

Function name: `public int sys_getAudioInput()`

Description: acquire the audio input type

API version: above V2.16.0

Simplified version:Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Audio input type 0: automatic 1: microphone 2: headphone

Note

- This interface is not supported for the time being.

5.5、 IO

5.5.1、 sys_setGpioDirection

Function name: `public int sys_setGpioDirection(int io, int direction, int value)`

Description: set the input and output status of GPIO

Parameter:

Parameter name	Type	instruction
io	int	Io ports, from 1 to 10
direction	int	GPIO direction 0:input 1:output
value	int	When the direction of GPIO is input, set 0:low level 1:high level

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- API versionV2.12.0+ available constant class

5.5.2、sys_getGpioDirection

Function name: public int sys_getGpioDirection(int io)

Description: acquire the input and out status of GPIO

Parameter:

Parameter name	Type	instruction
io	int	Io port, from 1 to 10

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	GPIO direction 0: input 1: output

- Note
- API versionV2.12.0+ available constant class

5.5.3、sys_getGpioValue

Function name: public int sys_getGpioValue(int io)

Description: acquire the level stratus of GPIO

Parameter:

Parameter name	Type	instruction
----------------	------	-------------

io	int	Io port, from 1 to 10
----	-----	-----------------------

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	0:low level 1:high level

Note

- API versionV2.12.0+ available constant class

5.5.4、sys_setExGpioDirection

Function name: public int sys_setExGpioDirection(int io, int direction, int value)

Description: set the input and output status of external GPIO

Parameter:

Parameter name	Type	instruction
io	int	Io port, from 1 to 10
direction	int	GPIO direction 0:input 1:output
value	int	When the direction of GPIO is output, set 0:low level 1:high level

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- API versionV2.12.0+ available constant class

5.5.5、sys_getExGpioDirection

Function name: public int sys_getExGpioDirection(int io)

Description: acquire the input and output status of external GPIO

Parameter:

Parameter name	Type	instruction
io	int	Io mouth, from 1 to 10

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	GPIO direction 0:input 1:output

- NoteAPI versionV2.12.0+ available constant class

5.5.6、sys_getExGpioValue

Function name: public int sys_getExGpioValue(int io)

Description: acquire the level status of external GPIO

Parameter:

Parameter name	Type	instruction
io	int	Io ports, from 1 to 10

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	0:low level 1:high level

Note

- API versionV2.12.0+available constant class

5.5.7、sys_setControl

Function name: public int sys_setControl(int type, boolean enable)

Description: Other IO related Settings (board status light, module power supply)

Parameter:

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

		16:TP reset 17:GBE reset 18:EDP BackLight power 19:Fan power
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- API version V2.12.0+ available constant class
- The common application only supports the control of LED lights (5) and fans (19)
- Other controls need to be at the systematic application level

5.5.8、sys_getControl

Function name: public int sys_getControl(int type)

Description: acquire other IO related Settings (board status light, module power supply)

Parameter:

Parameter name	Type	instruction
type	int	3:wifi power 4:wifi reset 5:led control 6:speak power 7:lvds power 8:lvds reset 9:4G power 10:4G reset 11:LAN power 12:LAN reset 13:SD power 14:SD reset 15:TP power 16:TP reset 17:GBE reset 18:EDP BackLight power 19:Fan power

Return Parameter Instruction

Parameter name	Type	instruction
----------------	------	-------------

Return value	int	1:open 0:close
--------------	-----	----------------

Note

- API version V2.12.0+ available constant class
- The common application only supports the control of LED lights (5) and fans (19)
- Other controls need to be at the systematic application level

5.6、Time switches

5.6.1、sys_setAutoPowerOnOff

Function name: public int sys_setAutoPowerOnOff(boolean enable, int[] week, int onHour, int onMinute, int offHour, int offMinute)

Description: set time switches

Parameter:

Parameter name	Type	instruction
enable	boolean	true:open false:close
week	int[]	date, new int[]{0,0,0,0,0,0,0}recurrents a repeat within 7 days, with Sunday being the first 1:set 0:not set
onHour	int	24-hour boot time (hour)
onMinute	int	Boot time (minute)
offHour	int	24-hour shutdown time (hour)
offMinute	int	Shutdown time (minute)

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling result refers to error codes.

Example

```
SmdtManagerNew smdt = SmdtManagerNew.getInstance(this);
String offTime = "11:00";
String onTime = "12:00";
int on_hour = Integer.parseInt(onTime.split(":")[0]);
int on_min = Integer.parseInt(onTime.split(":")[1]);
int off_hour = Integer.parseInt(offTime.split(":")[0]);
int off_min = Integer.parseInt(offTime.split(":")[1]);
//int[] week = null; //the setting day
```

```
int[] week = new int[]{1,0,0,0,0,0,0} //the first is Sunday, 1 means
set, 0means not set
int result = smdt.sys_setAutoPowerOnOff(true, week, on_hour, on_min,
off_hour, off_min);
```

- Note
- The first array of week parameters is Sunday, and so on
- Null enters week parameters means only setting once, namely, the setting day.
- If no repetition is set and only one setting is set, the shutdown time set needs to be after the current time.
- The shutdown time needs to be more than 3 minutes apart from the current time; the startup time and shutdown time also need to be more than 3 minutes apart.
- Timing switch error code category

5.6.2、 sys_getAutoPowerOnOffEnable

Function name: public int sys_getAutoPowerOnOffEnable(int type)

Description: acquire the condition of timing switch

API version: after V2.6.0

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	0:open 1:close

Note

- Before V2.6.0 version, the acquisition of timing switch status can [sys_getAutoPowerOnOff](#)

5.6.3、 sys_getAutoPowerOnOff

Function name: public String sys_getAutoPowerOnOff(int type)

Description: acquire the power on/off time of timing switch setting

API version: There are some changes after V2.6.0 , please refer to the parameters of note.

Parameter name	Type	Instruction
type	int	0: Scheduled shutdown time (24-hour format, including ":" number) 1: Time of scheduled startup (24-hour system, including ":" number) 2: New scheduled shutdown time (year, month, day, hour, minute, interface, millisecond) 3: New scheduled startup time (year, month, day, hour, minute, interface, millisecond)

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	String	Set the current on/off time (hours and minutes in 24-hour format)

Note

- Before V2.6.0 version, the interface can be used to determine whether the timer switch is open or not. When it is open, it has value; when it is closed, it is null
- Before V2.6.0 version, the acquisition of time can be achieved only when the timer is switched on.
- After V2.6.0 version, acquire is the last setting, regardless of the switch status
- After V2.24.0 version, supports the acquisition of new shutdown and startup time interfaces.

5.6.4、sys_getAutoPowerOnOffRepeat

Function name: public int[] sys_getAutoPowerOnOffRepeat()

Description: acquire the repetition status of the on/off setting at fixed intervals

API version: There are some changes after V2.6.0, please refer to Note.

Parameter: No

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	int	date, new int[] {0,0,0,0,0,0,0} represents 7 days, and the first number is Sunday 1:set 0:not set

Note

- The first element of week parameter array is Sunday, and so on
- Before V2.6.0 version, when timing switcher starts, it can be acquired, when timing switcher closes, it is null.
- Before V2.6.0 version, the acquisition belongs to the last time setting, which has nothing to do with the switch state

5.6.5、sys_setAutoPowerOnOffTime

Function name: public int sys_setAutoPowerOnOffTime(boolean enable, long onTimeMillis, long offTimeMillis)

Description: Set the time to turn on and off (year, month, day, hour and minute), and support to turn off on a certain day and turn on on a certain day

API version: V2.23.0+

Parameter:

Parameter name	Type	instruction
enable	boolean	true:open false:close
onTimeMillis	long	Fixed startup time (year, month, day, hour, minute, time converted to milliseconds) Unit: milliseconds
offTimeMillis	long	Set shutdown time (year, month, day, hour, minute, time converted to milliseconds) Unit: milliseconds

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Example

```
SmdtManagerNew smdt = SmdtManagerNew.getInstance(this);
String onTimeStr = "2024-03-25 12:00";
String offTimeStr = "2024-03-28 18:00";

SimpleDateFormat format = new SimpleDateFormat("yyyy-MM-dd HH:mm");
Date onTimeDate = format.parse(onTimeStr);
Date offTimeDate = format.parse(offTimeStr);

long onTime = onTimeDate.getTime();
long offTime = offTimeDate.getTime();
int result = smdtManagerNew.sys_setAutoPowerOnOffTime(true, onTime, offTime);
```

Note

- Only the shutdown time can be set. If the startup time is 0, only the shutdown time can be set.

5.7、 Watch dog

5.7.1、 sys_setWatchDog

Function name: public int sys_setWatchDog(boolean enable, int poweroff_time)

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

Parameter:

Parameter name	Type	instruction
enable	boolean	true:open false:close
poweroff_time	int	The shutdown time after not feeding the dog is 60 seconds by default. Unit: seconds

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- The poweroff_time parameters in interface:Not supported for the time being. The default is 60 seconds.

5.7.2、 sys_getWatchDog

Function name: public int sys_getWatchDog()

Description: acquire the open and close status of the watch dog.

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

5.7.3、 sys_setWatchDogFeed

Function name: `public int sys_setWatchDogFeed()`

Description: feed on the watch dog

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.7.4、sys_getWatchDogShutDownTime

Function name: `public int sys_getWatchDogShutDownTime()`

Description: acquire the time it takes after opening the door and not feeding the dog does it shut down.

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Shutdown time, unit: seconds

Note

- The interface is not available and is not supported for the time being.

5.8、Upgrade and Debug

5.8.1、sys_setPointerLoction

Function name: `public int sys_setPointerLoction(boolean enable)`

Description: Set the switch status of pointer position

Parameter:

Parameter name	Type	instruction
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.8.2、 sys_getPointerLoction

Function name: public int sys_getPointerLoction()

Description: acquire the switch status of pointer position

Parameter: No **Return Parameter Instruction**

Parameter name	Type	instruction
Return value	int	1:open 0:close

5.8.3、 sys_setPowerOff

Function name: public int sys_setPowerOff()

Description: system power-off

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.8.4、 sys_setReboot

Function name: public int sys_setReboot()

Description: system reboot

Parameter:

No **Return Parameter Instruction**

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.8.5、 sys_setRebootByMcu

Function name: public int sys_setReboot()

Description: system reboot (MCU reboot)

API version: above V2.13.0

Parameter: No

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	int	Calling results refers to error codes

5.8.6、sys_setDeveloperOptions

Function name: public int sys_setDeveloperOptions(boolean enable)

Description: Set the developer options switch status

Parameter:

Parameter name	Type	Instruction
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	int	Calling results refers to error codes

5.8.7、sys_getDeveloperOptions

Function name: public int sys_getDeveloperOptions()

Description: acquire the developer options switch status

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

5.8.8、sys_doUpdatePackage

Function name: public int sys_doUpdatePackage(int type, String filepath)

Description: OTA system update

Parameter:

Parameter name	Type	instruction
type	int	Update Type 0:OTA update
filepath	String	The absolute path of the OTA firmware package

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.8.9、sys_doUpdatePackageAB

Function name: public int sys_doUpdatePackageAB(String filepath, UpdateCallback callback)

Description: OTA-AB partition upgrade, online upgrades are supported without entering recovery

API version: V2.25.0+

system version: android12+

Parameter:

Parameter name	Type	Instruction
filepath	int	Update Type 0:OTA update
callback	UpdateCallback	Update callback

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	int	Calling results refers to error codes

Example

```

SmdtManagerNew smdt = SmdtManagerNew.getInstance(this);
String path = "/sdcard/xxx.zip";
int result = smdt.sys_doUpdatePackageAB(path, new SmdtManagerNew.UpdateCallback() {
    @Override
    public void onStatusUpdate(int status, float percent) throws RemoteException {
        if (status == UpdateEngineUtil.UpdateStatusConstants.DOWNLOADING) {
            Log.d(TAG, "update progress: " + percent);
            int progress = (int) (percent * 100);
        }
    }
});
@Override

```

```

    public void onPayloadApplicationComplete(int errCode) throws RemoteException {
        if (errCode == UpdateEngineUtil.ErrorCodeConstants.SUCCESS) {
            Log.d(TAG, "Installation succeeded!");
        } else {
            Log.d(TAG, "Installation fail:"+errCode);
        }
    }
});

```

- Not
- It is only applicable to system with AB partition. It depends on whether the equipment firmware is AB partition policy.
- For the status of onStatusUpdate and error code of onPayloadApplicationComplete in the callback, please refer to OTA-AB upgrade error code.

5.8.10、sys_rebootRecovery

Function name: public int sys_rebootRecovery()

Description: restore factory setting

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.8.11、sys_setSystemLog

Function name: public int sys_setSystemLog(boolean enable, int[] type, String dirpath, long fileMaxSize, boolean boot)

Description: Capture system logs to the specified path

Parameter:

Parameter name	Type	instruction
enable	boolean	true:open false:close
type	int[]	The logs that need to be captured Type null By default, all logs are output 0. All logs are not continuously output, only captured once. 1 Drive log 2 system Log 3 Main logs 4 Communication logs

		5system attribute 6 kernel Start Parameter:(need root) 7 Memory usage status 8 Non-pre-installed APK information 9 MCU information 10 Event Log 11. Crash log
dirpath	String	Storage directory The default storage directory is system root directory /SmdtLogNew/current time
fileMaxSize	long	The maximum storage value (in Byte) of the log file is 50MB by default
boot	boolean	Start the machine and capture logs true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- Before storing the log, it will automatically check the remaining system memory. If the memory is insufficient, the log file will be cleaned.
- When a single log file reaches the maximum storage value, a new file will be automatically generated. A maximum of 10 files will be generated under a single directory. After reaching the upper limit, the oldest file will be deleted.
- Setting Type: type include 0 which means that only the logs that need to be captured are captured at the current time and do not continue to output. After the input is finished, the log status will become close; 0,1,2 represents that the drive logs and Android logs are captured only once;
- Log storage path: It is recommended to use system root directory/folder name + current time.
- The type of event log and crash log are put in V2.19.0+, support separate acquisition.

5.8.12、sys_getSystemLog

Function name: public int sys_getSystemLog()

Description: Get the system log switch status

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

5.8.13、sys_getProcessLogcat

Function name: public void sys_getProcessLogcat(LogCallback callback)

Description: acquire system logs line by line

Parameter:

Parameter name	Type	instruction
callback	LogCallback	A callback for listening to each log line

Return Parameter instruction: No

Example

```
SmdtManagerNew smdt = SmdtManagerNew.getInstance(this);
int[] type = new int[]{2}; // 2 为 system 日志, 有且包括即可
smdt.sys_setSystemLog(true, type, null, 0, false);
smdt.sys_getProcessLogcat(new SmdtManagerNew.LogCallback() {
    @Override
    public void onSingleLine(String log) throws RemoteException {
        ...
    }
});
```

Note

- Callback interface: [LogCallback](#)
- You need to call the open system log switch first and select type: 2 System Log
- Close the system log switch or return null to release the listener

5.8.14、sys_getProcessAnrLog

Function name: public int sys_getProcessAnrLog(String filepath)

Description: acquire ANR log

Parameter:

Parameter name	Type	instruction
----------------	------	-------------

filepath	String	Save path
----------	--------	-----------

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- API version :V2.19.0+ The path requires a folder path to be passed in.

5.8.15、sys_setAdbDebug

Function name: public int sys_setAdbDebug(int type, boolean enable)

Description: Set the status of the ADB switch

Parameter:

Parameter name	Type	instruction
type	int	ADB type 0:USB 1:network
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- API versionV2.12.0+ available constant class

5.8.16、sys_getAdbDebug

Function name: public int sys_getAdbDebug(int type)

Description: acquire the status of the ADB switch

Parameter:

Parameter name	Type	instruction
type	int	ADB type 0:USB 1:network

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

Note

- API versionV2.12.0+ available constant class

5.8.17、sys_setOTGMode

Function name: public int sys_setOTGMode(int mode)

Description: Set OTG mode

Parameter:

Parameter name	Type	instruction
mode	int	0:HOST 1:DEVICE

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- API versionV2.12.0+ available constant class

5.8.18、sys_getOTGMode

Function name: public int sys_getOTGMode()

Description: acquire current OTG mode

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	0:HOST 1:DEVICE

Note

- API versionV2.12.0+ available constant class

5.8.19、 sys_setFloatBall

Function name: public int sys_setFloatBall(boolean enable)

Description: Set the floating ball status

Parameter:

Parameter name	Type	instruction
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.8.20、 sys_getFloatBall

Function name: public int sys_getFloatBall()

Description: acquire the floating ball status

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

5.8.21、 sys_setUpdateExState

Function name: public int sys_setUpdateExState(boolean enabled)

Description: Set whether to allow external storage equipment clone, upgrade system

API version: above V2.12.0

Simplified version Not support

Parameter:

Parameter name	Type	instruction
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

5.8.22、sys_getUpdateExState

Function name: public int sys_getUpdateExState()

Description: acquire whether to allow external storage equipment clone, upgrade system

API version: above V2.12.0

Simplified version Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	0:forbid 1:allow

5.8.23、sys_setApiDebugLevel

Function name: public int sys_setApiDebugLevel(int level)

Description: Set the level of API print log

Parameter:

Parameter name	Type	instruction
int	level	Level (0/1/2/3) The higher the level, the more information is printed

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- The logs in JNI SO database can not be controlled.

5.8.24、sys_getApiDebugLevel

Function name: public int sys_getApiDebugLevel()

Description: acquire the level of current API print log

Parameter: No **Return Parameter Instruction**

Parameter name	Type	instruction
Return value	int	Print grade (0/1/2/3)

5.8.25、sys_getErrorDescription

Function name: public String sys_getErrorDescription(int errorCode)

Description: acquire description corresponding to the error codes

Parameter:

Parameter name	Type	instruction
int	errorCode	Error codes

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	description corresponding to the error codes; support Chinese and English, can change the language description according to systematic language.

6、Hardware Control

6.1、Storage

6.1.1、dev_getSDcardPath

Function name: public String dev_getSDcardPath()

Description: acquire the path of the external storage SD card

Parameter: No

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	String	the path of the external storage SD card

6.1.2、dev_getUdiskPath

Function name: public List dev_getUdiskPath()

Description: acquire all USB flash disk paths for external storage

Parameter: No

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	List	all USB flash disk paths for external storage

6.1.3、dev_unmountExternalStorage

Function name: public int dev_unmountExternalStorage(String path, boolean force, boolean removeEncryption)

Description: uninstall external storage

Parameter:

Parameter name	Type	instruction
path	String	The absolute path of the external storage to be uninstalled
force	boolean	Whether to force uninstall
removeEncryption	boolean	Whether to remove the encryption device

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

6.1.4、dev_getPublicPartitionSize

Function name: public int dev_getPublicPartitionSize(int type, int deviceId)

Description: acquire public and partition size

Parameter:

Parameter name	Type	instruction
type	int	Type 0:emmc 1:eeprom
deviceId	int	When read eeprom, equipment ID starts with 0. Each ID represents a EEPROM.

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Total size of private partitions

6.1.5、dev_readPublicPartition

Function name: public byte[] dev_readPublicPartition(int type, int deviceId, int areaId, int start_addr, int size)

Description: Read the public and private partitions

Parameter:

Parameter name	Type	instruction
type	int	Type 0:emmc 1:eprom
deviceId	int	When read eeprom, equipment ID starts with 0. Each ID represents a EEPROM.
areaId	int	When read eeprom, equipment ID starts with 1. Each ID represents a area.
start_addr	int	starting address
size	int	The length of data

Return Parameter Instruction

Parameter name	Type	instruction
Return value	byte[]	Data that has been read

6.1.6、dev_writePublicPartition

Function name: public int dev_writePublicPartition(int type, int deviceId, int areaId, int start_addr, int size, byte[] buf)

Description: Write into the public and private partitions

Parameter:

Parameter name	Type	instruction
type	int	Type 0:emmc 1:eprom
deviceId	int	When read eeprom, equipment ID starts with 0. Each ID represents a EEPROM.
areaId	int	When read eeprom, equipment ID starts with 1. Each ID represents a area.
start_addr	int	starting address
size	int	starting address
buf	byte[]	Input data

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- It is recommended to distinguish the partition size before writing, and write cannot be done if it exceeds

6.2、Serial Port

6.2.1、dev_getUartPath

Function name: `public String dev_getUartPath(String uart)`

Description: acquire the absolute path of the serial port according to the number of ports.

Parameter:

Parameter name	Type	instruction
uart	String	The number of ports corresponding to the serial port, such as:uart0/uart1/uart2/uart3

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	The absolute path of the serial port, such as /dev/ttyS1, is subject to the actual serial port that needs to be operated.

6.2.2、dev_openUart

Function name: `public int dev_openUart(String uartNode, int baudrate, int databits, int stopbits, int parity, int flow_ctrl)`

Description: open serial port

Parameter:

Parameter name	Type	instruction
uartNode	String	The absolute path of the serial port
baudrate	int	Baud rate

databits	int	Data bit:8/7/6/5
stopbits	int	Stop bit :1/2
parity	int	Check bit: 0 no parity check, 1 parity check, 2 parity check
flow_ctrl	int	Flow control: 0 no flow control, 1 hardware flow control, 2 software flow control

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- Only support open a serial port to open and send and receive data.
- Multiple serial ports need to be developed by users themselves to open simultaneously.
- Remember to close the serial port after operation, otherwise it will be occupied all the time.

Example

```

SmdtManagerNew smdt = SmdtManagerNew.getInstance(this);
String uart = "/dev/ttyS1";
int baudrate = 9600;
int databits = 8;
int stopbits = 1;
int parity = 0;
int flow_ctrl = 0;
int result = smdt.dev_openUart(uart, baudrate, databits, stopbits, p
arity, flow_ctrl);//open 串口
if(result == RET_API_OK){
    //Keep receiving serial data after successful open
    smdt.dev_receiveUart(uart, new SmdtManagerNew.DataCallback() {
        @Override
        public void onDataReceive(byte[] buffer, int size) throws Remote
Exception {
            if (buffer != null) {
                DEBUG("data size :" + size);
            }
        }
    });
}

```

6.2.3、dev_closeUart

Function name: public int dev_closeUart(String uartNode)

Description: close serial port

Parameter:

Parameter name	Type	instruction
uartNode	String	The absolute path of the serial port

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

- Note
- Only support open a serial port to open and send and receive data.
- Multiple serial ports need to be developed by users themselves to open simultaneously.
- Remember to close the serial port after operation, otherwise it will be occupied all the time.

6.2.4、dev_sendUart

Function name: public int dev_sendUart(String uartNode, String data, boolean hex)

Description: Send serial data

Parameter:

Parameter name	Type	instruction
uartNode	String	The absolute path of the serial port
data	boolean	data
hex	boolean	true:hexadecimal false:decimal system

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- Only support open a serial port to open and send and receive data.
- Multiple serial ports need to be developed by users themselves to open simultaneously.

- Remember to close the serial port after operation, otherwise it will be occupied all the time.

6.2.5、dev_receiveUart

Function name: public void dev_receiveUart(String uartNode, DataCallback callback)

Description: Receive serial port data

Parameter:

Parameter name	Type	instruction
uartNode	String	The absolute path of the serial port
callback	DataCallback	Serial ports data callback

Return Parameter instruction: No

Example

```
SmdtManagerNew smdt = SmdtManagerNew.getInstance(this);
smdt.dev_receiveUart(uartNode, new SmdtManagerNew.DataCallback() {
    @Override
    public void onDataReceive(byte[] buffer, int size) throws RemoteException {
        if (buffer != null) {
            ...
        }
    }
});
```

Note

- Callback interface: [DataCallback](#)
- Only support open a serial port to open and send and receive data.
- Multiple serial ports need to be developed by users themselves to open simultaneously.
- Remember to close the serial port after operation, otherwise it will be occupied all the time.

6.3、Camera

6.3.1、dev_getCameraConfig

Function name: public int[] dev_getCameraConfig(int cameraId)

Description: acquire camera related ParameterS:

Simplified version: Not support

Parameter:

Parameter name	Type	instruction
cameraId	int	Camera ID, such as:camera0

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int[]	The array where camera parameters are stored in

Note

- The sequence meaning of an int[] array:
The first number: camera direction 0:rear 1:front-facing
The second number: camera rotation mode 0:follow system screen 1:fixed
The third number: when the rotation mode is fixed, the rotation angel is:0/90/180/270
The fourth number: Previewing picture is mirrored on the left and right 0:no 1:yes
The fifth number: Imaging picture is mirrored on the left and right 0:no 1:yes
The sixth number: Imaging picture is mirrored on the up and down 0:no 1:yes
The seventh number: Image rotation angle:0/90/180/270
The eighth number: video rotation angel:0/90/180/270

6.3.2、 dev_setCameraDirection

Function name: public int dev_setCameraDirection(int cameraId, int cameraDirection)

Description: Set the front and rear orientation of the camera

Parameter:

Parameter name	Type	instruction
cameraId	int	Camera ID, such as camera0
cameraDirection	int	The direction of camera 0:rear 1:front-facing

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- After setting, restart to take effect.

6.3.3、dev_setCameraRotationMode

Function name: public int dev_setCameraRotationMode(int cameraId, int rotationMode, int lockRotation)

Description: Set the camera rotation mode

Simplified version: Not support

Parameter:

Parameter name	Type	instruction
cameraId	int	Camera ID, such as camera0
rotationMode	int	Camera rotation mode 0:follow system screen 1:fixed
lockRotation	int	when the rotation mode is fixed, the rotation angel is :0/90/180/270

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- After setting, restart to take effect.

6.3.4、dev_setCameraMirror

Function name: public int dev_setCameraMirror(int cameraId, boolean previewMirrorHorizontal, boolean imageMirrorHorizontal, boolean imageMirrorVertical)

Description: set the mirror image of the camera

Simplified version:Not support

Parameter:

Parameter name	Type	instruction
cameraId	int	Camera ID, camera0
previewMirrorHorizontal	boolean	Previewing picture is mirrored on the left and right false:no true:yes
imageMirrorHorizontal	boolean	Imaging picture is mirrored on the left and right false:no true:yes
imageMirrorVertical	boolean	Imaging picture is mirrored on the up and down false:no true:yes

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- It takes effect immediately after setting.

6.3.5、dev_setCameraImageRotation

Function name: public int dev_setCameraImageRotation(int cameraId, int rotation)

Description: Set the camera image rotation

Simplified version:Not support

Parameter:

Parameter name	Type	instruction
path	String	Camera ID, such as, camera0
rotation	boolean	Image rotation angel:0/90/180/270

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	int	Calling results refers to error codes

Note

- It takes effect immediately after setting.

6.3.6、dev_setCameraVideoRotation

Function name: public int dev_setCameraVideoRotation(int cameraId, int rotation)

Description: set camera video rotation

Simplified version:Not support

Parameter:

Parameter name	Type	instruction
cameraId	int	Camera ID, such as:camera0
rotation	int	Video rotation angel:0/90/180/270

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- It takes effect immediately after setting.

6.4、I2C

6.4.1、dev_openI2c

Function name: public int dev_openI2c(String i2cNode)

Description: openi2c peripherals

Parameter:

Parameter name	Type	instruction
i2cNode	String	i2c device node

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- The interface is not available and is not supported for the time being.

6.4.2、dev_closeI2c

Function name: `public int dev_closeI2c(n)`

Description: `closei2c` peripherals

Parameter:

No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- The interface is not available and is not supported for the time being.

6.4.3、`dev_writeI2c`

Function name: `public int dev_writeI2c(int slave, int reg, byte[] data)`

Description: Write data to the device register of `i2c`

Parameter:

Parameter name	Type	instruction
slave	int	slave addr
reg	int	reg addr
data	byte[]	Data written

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Number of bytes written

Note

- The interface is not available and is not supported for the time being.

6.4.4、 dev_readI2c

Function name: public byte[] dev_readI2c(int slave, int reg, int len)

Description: read data to the device register of i2c

Parameter:

Parameter name	Type	instruction
slave	int	slave addr
reg	int	reg addr
len	int	the length of the data that has been read

Return Parameter Instruction

Parameter name	Type	instruction
Return value	byte[]	data

Note

- The interface is not available and is not supported for the time being.

6.5、 SPI

6.5.1、 dev_openSpi

Function name: public int dev_openSpi(String spiNode)

Description: openspi peripherals

Parameter:

Parameter name	Type	instruction
spiNode	String	spi device node

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- The interface is not available and is not supported for the time being.

6.5.2、 dev_closeSpi

Function name: public int dev_closeSpi()

Description: closespi peripherals

Parameter:

No **Return Parameter Instruction**

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- The interface is not available and is not supported for the time being.

6.5.3、 dev_readSpi

Function name: public byte[] dev_readSpi(int len)

Description: read spi equipment data

Parameter:

Parameter name	Type	instruction
len	int	The length of data

Return Parameter Instruction

Parameter name	Type	instruction
Return value	byte[]	data

Note

- The interface is not available and is not supported for the time being.

6.5.4、 dev_writeSpi

Function name: public int dev_writeSpi(byte[] data)

Description: write spi equipment data

Parameter:

Parameter name	Type	instruction
data	byte[]	Data written

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Number of bytes written

Note

- The interface is not available and is not supported for the time being.

6.6、Can

6.6.1、dev_openCan

Function name: public int dev_openCan(String canNode, int baudrate)

Description: opencan equipment

Parameter:

Parameter name	Type	instruction
canNode	String	can equipment node, 如 can0/can1
baudrate	int	can equipment baud rate , 如 125000

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- The interface is not available and is not supported for the time being.

6.6.2、dev_closeCan

Function name: public int dev_closeCan(String canNode)

Description: the can equipment of closeopen

Parameter:

Parameter name	Type	instruction
canNode	String	can equipment node, such as: can0/can1

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- The interface is not available and is not supported for the time being.

6.6.3、dev_receiveCan

Function name: public int dev_receiveCan()

Description: receive the data of can equipment

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- The interface is not available and is not supported for the time being.

6.6.4、dev_sendCan

Function name: public int dev_sendCan(long canid, byte[] data)

Description: write can equipment data

Parameter:

Parameter name	Type	instruction
canid	long	The ID of the data to be sent, corresponding to can_id of can_frame

data	byte[]	Data sent, which corresponding to hwctrl_canOpen, it will be written in node of canOpen
------	--------	---

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- The interface is not available and is not supported for the time being.

6.7、USB

6.7.1、dev_setUsbPower

Function name: public int dev_setUsbPower(int type, int usbId, boolean enable)

Description: Set the power switch status of USB port

Parameter:

Parameter name	Type	instruction
type	int	0:USBOTG 1:USBHOST
usbId	int	USB port ID: OTG:1;HOST:1, 2, 3
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- Some power ports do not support switching

6.7.2、dev_getUsbPower

Function name: public int dev_getUsbPower(int type, int usbId)

Description: acquire the power switch status of USB port

Parameter:

Parameter name	Type	instruction
type	String	0:USBOTG 1:USBHOST
usbId	boolean	USB 口 ID: ID: OTG:1;HOST:1, 2, 3

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

Note

- Some power ports do not support switching

6.8、Light

6.8.1、dev_setLedLighted

Function name: public int dev_setLedLighted(String ledColor, boolean lighted)

Description: Set the status of the three-color light

API version: above V2.11.0

Parameter:

Parameter name	Type	instruction
ledColor	String	Red light: LED_RED green light: LED_GREEN white light: LED_WHITE
lighted	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- API version V2.12.0+ available constant class

6.8.2、dev_getLedState

Function name: public int dev_getLedState(String ledColor)

Description: acquire the status of the three-color light

API version: above V2.11.0

Parameter:

Parameter name	Type	instruction
ledColor	String	Red light: LED_RED green light: LED_GREEN white light: LED_WHITE

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

Note

- API version V2.12.0+ available constant class

7、Industry customization

7.1、Wiegand

7.1.1、custom_sendWiegandCard

Function name: public int custom_sendWiegandCard(String idCard, int transform)

Description: the number of Wiegand send card

Parameter:

Parameter name	Type	instruction
idCard	String	Wiegand se number
transform	int	1:Wiegand26 2:Wiegand34

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

7.1.2、custom_sendWiegandCardHIDPID

Function name: public int custom_sendWiegandCardHIDPID(String HID_value, String PID_value, int transform)

Description: Wiegand uses hidden code and public code to send card number

Parameter:

Parameter name	Type	instruction
HID_value	String	Hidden code
PID_value	String	Public code
transformat	int	1:Wiegand26 2:Wiegand34

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

7.1.3、 custom_readWiegandData

Function name: public void custom_readWiegandData(WiegandCallback callback)

Description: read Wiegand input

Parameter:

Parameter name	Type	instruction
callback	WiegandCallback	Wiegand callback

Return Parameter instruction: No

Example

```
SmdtManagerNew smdt = SmdtManagerNew.getInstance(this);
smdt.custom_readWiegandData(new SmdtManagerNew.WiegandCallback() {
    @Override
    public void onReadData(String data) throws RemoteException {
        ...
    }
});
```

Note

- callback interface: [WiegandCallback](#)

7.1.4、 custom_releaseWiegandRead

Function name: public int custom_releaseWiegandRead()

Description: withdraw from Wiegand and input blocking

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

7.2、 Electric Relay

7.2.1、 custom_setRelayIoMode

Function name: public int custom_setRelayIoMode(int mode, int delay)

Description: Set relay mode and delay

Parameter:

Parameter name	Type	instruction
mode	int	0:Not automatic closing mode 1: Automatic closing mode (high effective-default low level, send the command high level for n seconds, and finally low level) 2: Automatic closing mode (low effective-default high level, send the command low level for n seconds, and finally high level)
delay	int	Delay in automatic closing mode

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- Set the delay for automatic closing mode in seconds, up to 63 seconds

7.2.2、 custom_getRelayIoMode

Function name: public int custom_getRelayIoMode

Description: acquire current mode of the relay

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	0:Not automatic closing mode 1: Automatic closing mode (high effective-default low level, send the command high level for n seconds, and finally low level) 2: Automatic closing mode (low effective-default high level, send the command low level for n seconds, and finally high level)

7.2.3、custom_setRelayIoEnable

Function name: `public int custom_setRelayIoEnable(boolean enable)`

Description: set the switch status of the relay

Parameter:

Parameter name	Type	Instruction
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	int	Calling results refers to error codes

7.2.4、custom_getRelayIoEnable

Function name: `public int custom_getRelayIoEnable()`

Description: acquire the switch status of the relay

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

7.3、Application

7.3.1、custom_dial

Function name: `public int custom_dial(String number)`

Description: Customized dialing without interface

Simplified version: Not support

Parameter:

Parameter name	Type	instruction
number	String	Telephone number

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

7.3.2、 custom_endCall

Function name: public int custom_endCall()

Description: customized hanging up without interface

Simplified version:Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

7.3.3、 custom_addAppliesEncryption

Function name: public int custom_addAppliesEncryption(String packageName, String oldpwd, String pwd)

Description: Add the application that needs to be encrypted

Simplified version:Not support

Parameter:

Parameter name	Type	instruction
packageName	String	Application package name
oldpwd	String	the old passwords is used to correct and check.If without any passwords,open directly.
pwd	String	passwords

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- This interface is called for adding and modifying passwords, and the old password must be entered to modify the password

7.3.4、 custom_getAppliesEncryption

Function name: public List custom_getAppliesEncryption(String packageName)

Description: acquire encrypted applications and their passwords

Simplified version:Not support

Parameter:

Parameter name	Type	instruction
packageName	String	Application package name:look for the passwords of designated package name.

Return Parameter Instruction

Parameter name	Type	instruction
Return value	List	acquire overall encrypted applications and their passwords

Note

- When the package name parameter is null, it returns all encrypted applications and corresponding passwords

7.3.5、 custom_delAppliesEncryption

Function name: public int custom_delAppliesEncryption(String packageName, String pwd)

Description: delete the app from the encryption list

Simplified version: Not support

Parameter:

Parameter name	Type	Instruction
----------------	------	-------------

packageName	String	Application package name
pwd	String	Passwords for the application

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	int	Calling results refers to error codes

7.4、 system

7.4.1、 custom_cleanRecentTasks

Function name: public int custom_cleanRecentTasks()

Description: Clear system recent running tasks

Simplified version:Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	int	Calling results refers to error codes

7.4.2、 custom_killPidProcess

Function name: public int custom_killPidProcess(int pid)

Description: end the process according to pid

Simplified version:Not support

Parameter:

Parameter name	Type	instruction
pid	int	Process PID

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- It kills the process directly. Use it with caution.

7.4.3、 custom_getPidProcess

Function name: `public int custom_getPidProcess(String packageName)`

Description: acquire PID according to package name

Simplified version Not support

Parameter:

Parameter name	Type	instruction
packageName	String	Application package name

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	The PID corresponding to the application package name

7.4.4、 custom_getPidProcessInfo

Function name: `public String custom_getPidProcessInfo(int pid)`

Description: acquire the package name corresponding to the pid process.

Simplified version: Not support

Parameter:

Parameter name	Type	instruction
pid	int	PID process

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	the package name corresponding to the pid process.

7.4.5、 custom_setDesktopApp

Function name: `public int custom_setDesktopApp(String packageName)`

Description: Set customized desktop shortcuts

API version: above V2.5.0

Simplified version:Not support

Parameter:

Parameter name	Type	Instruction
packageName	String	Application package name.

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- The setting is based on existent application and desktop icon.
- This interface replaces the first icon position on the OS-Launcher home screen .
- Passing in null means to clear the current binding state and restore the original icon. If removed, no content is displayed.

7.4.6、 custom_getDesktopApp

Function name: public String custom_getDesktopApp()

Description: acquire customized desktop shortcuts

API version: above V2.5.0

Simplified version:Not support

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	Application package name

8、 Others

8.1、 Constant class8.1.1 Constant class: VariableUtil

API version: above V2.12.0.

Connected network type:

Ethernet public static final CONNECT_TYPE_ETHERNET = "ETH";

WIFI public static final CONNECT_TYPE_WIFI = "WIFI";

Mobile network public static final CONNECT_TYPE_MOBILE = "MOBILE";

Network type

Ethernet 0 public static final String TYPE_ETH0 = "eth0";

Ethernet 1 public static final String TYPE_ETH1 = "eth1";

WIFI public static final String TYPE_WLAN = "wlan0";

Mobile network public static final String TYPE_MOBILE = "mobile";

Network connection mode

dynamics public static final int MODE_DHCP = 0;

statics public static final int MODE_STATIC = 1;

WIFI encryption type

No password public static final int SECURITY_NONE = 0;

WEP public static final int SECURITY_WEP = 1;

WPA public static final int SECURITY_WPA = 2;

Application list

Black list public static final int BLACK_LIST = 0;

White list public static final int WHITE_LIST = 1;

Uninstall list public static final int UNINSTALL_LIST = 0;

Install list public static final int INSTALL_LIST = 1;

GPIO

GPIO direction input public static final int GPIO_IN = 0;

GPIO direction output public static final int GPIO_OUT = 1;

GPIO output low level public static final int GPIO_LOW = 0;

GPIO output high level public static final int GPIO_HIGH = 1;

ADB OTG

USB ADB public static final int ADB_USB = 0;

network ADB public static final int ADB_NETWORK = 1;

OTG HOST public static final int OTG_HOST = 0;

OTG DEVICE public static final int OTG_DEVICE = 1;

Storage partitioning

EMMC public static final int PARTITION_EMMC = 0;

EEPROM public static final int PARTITION_EEPROM = 1;

Camers

Rear camera public static final int CAMERA_FACING_BACK = 0;

Front-facing public static final int CAMERA_FACING_FRONT = 1;

Camera follows the direction of screen public static final int
CAMERA_ROTATION_FOLLOW = 0;

The direction of camera is fixed public static final int CAMERA_ROTATION_LOCK = 1;

Light

White light public static final String LED_WHITE = "LED_WHITE"; **red light**
public static final String LED_RED = "LED_RED"; **green light** public static
final String LED_GREEN = "LED_GREEN";

Wiegand

Wiegand26 public static final int WIEGAND_26 = 1; **Wiegand34** public static
final int WIEGAND_34 = 2;

Log type

All logs output not continuously and only capture for once. public static final int
LOG_ONLY_ONE = 0;

KERNEL logs (Continuous output) public static final int LOG_KERNEL = 1;
Android system logs (Continuous output) public static final int LOG_ANDROID = 2;
The main log buffer area public static final int LOG_MEDIA = 3;
Relation information about the buffer area public static final int LOG_RADIO = 4;
System properties public static final int LOG_PROP = 5;
Kernel start parameters: (need root) public static final int LOG_CONFIG = 6;
Memory usage public static final int LOG_MISC = 7;
Information about APK that is not installed in advance public static final int LOG_APP = 8;

Mcu message public static final int LOG_MCU = 9;

IO

WIFI power public static final int WIFI_POWER = 3;

WIFI reset public static final int WIFI_RESET = 4;

LED light public static final int LED_CONTROL = 5;

Speaker power public static final int SPEAK_POWER = 6;

LVDS power public static final int LVDS_POWER = 7;

LVDS reset public static final int LVDS_RESET = 8;

mobile network module power public static final int MOBILE_POWER = 9;

mobile network module reset public static final int MOBILE_RESET = 10;

LAN power public static final int LAN_POWER = 11;

LAN reset public static final int LAN_RESET = 12;

SD power public static final int SD_POWER = 13;

SD reset public static final int SD_RESET = 14;

TP power public static final int TP_POWER = 15; T

P reset public static final int TP_RESET = 16;

GBE reset public static final int GBE_RESET = 17;

EDP backlight power `public static final int EDP_POWER = 18;`

Fan power `public static final int FAN_POWER = 19;`

8.2、Network information object class

Network information object class: `NetworkInfoData`

➤ Acquire IP address

Function name: `public String getIp()`

Description: acquire current IP address

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	IP address

➤ Acquire gateway

Function name: `public String getGateway()`

Description: acquire the current gateway

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	gateway

➤ Acquire the sub-net mask

Function name: `public String getNetmask()`

Description: acquire the sub-net mask of current network

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	Sub-net mask

➤ Acquire DNS1

Function name: `public String getDns1()`

Description: acquire the DNS1 of current network

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	DNS1

➤ Acquire DNS2

Function name: public String getDns2()

Description: acquire the DNS2 of current network

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	DNS2

8.3、Timing switcher error codes

Error codes: TimerCode

variable name	Error codes	instruction
SET_SUCCESS	0	success
ERROR_UNKNOWN	-1	error
ERROR_CONFIG	-2	Parameter:error
ERROR_EXCEPTION	-5	exception
ERROR_TIME_PASSED	100	The set time has expired
ERROR_COMPARE_ON_OFF_TIME	101	There is no difference of 3 minutes between the start time and shutdown time
ERROR_COMPARE_NOW_OFF_TIME	102	There is no difference of 3 minutes between the current time and shutdown time
ERROR_COMPARE_NOW_ON_TIME	103	There is no difference of 6 minutes between the current time and startup time

API version: above V2.14.1

Note

Before V2.14.1 version, do not return to error codes that starts with “10” .

8.4、OTA-AB Update error codes

Error codes: UpdateEngineUtil

UpdateStatusConstants Class:

variable name	Error codes	instruction
IDLE	0	idle
CHECKING_FOR_UPDATE	1	checking for update
UPDATE_AVAILABLE	2	Available update
DOWNLOADING	3	downloading
VERIFYING	4	Verifying
FINALIZING	5	Finishing update
UPDATED_NEED_REBOOT	6	Has updated and need to reboot
REPORTING_ERROR_EVENT	7	Report error event
ATTEMPTING_ROLLBACK	8	attempting rollback
DISABLED	9	disabled

ErrorCodeConstants Class

variable name	Error codes	instruction
SUCCESS	0	success
ERROR	1	error
FILESYSTEM_COPIER_ERROR	4	there are some errors during the systematic copy of files
POST_INSTALL_RUNNER_ERROR	5	there are errors in running script or program after installed
PAYLOAD_MISMATCHED_TYPE_ERROR	6	mismatches with the type of payload(the content of update package)
INSTALL_DEVICE_OPEN_ERROR	7	The device or related resources cannot be opened for installation
KERNEL_DEVICE_OPEN_ERROR	8	The kernel device or related resources could not be opened
DOWNLOAD_TRANSFER_ERROR	9	An error occurred during the download transmission
PAYLOAD_HASH_MISMATCH_ERROR	10	The hash value of the payload does not match the expected value, which may indicate that the file has been tampered with
PAYLOAD_SIZE_MISMATCH_ERROR	11	The size of the payload is not in line with expectations
DOWNLOAD_PAYLOAD_VERIFICATION_ERROR	12	the verification of payload downloaded failed
NEW_ROOTFS_VERIFICATION_ERROR	15	The systematic verification of new root files failed
DOWNLOAD_STATE_INITIALIZATION_ERROR	20	The initialization of the download status failed
DOWNLOAD_METADATA_SIGNATURE_MISMATCH	26	mismatches with the signature of metadata

		downloaded
USER_CANCELLED	48	canceled by users during the process of update
PAYLOAD_TIMESTAMP_ERROR	51	the timestamp of payload is wrong, not support update afterwards.
UPDATED_BUT_NOT_ACTIVE	52	The update has finished, but the new system 尚 has not yet activated.
NOT_ENOUGH_SPACE	60	the storage space of device is not enough
DEVICE_CORRUPTED	61	Equipment is corrupted or the system of the file is wrong
UPDATE_APPLY_PACKAGE_FAILED	100	Application update package failed

API version: above V2.25.0

system version: android12+

Note

It is only applicable to system with AB partition. It depends on whether the equipment firmware is AB partition policy.

8.5、InstallCallback

The callback of installed application

Function name: public void onInstallFinished(String packageName, int returnCode, String msg);

Description: installed

Parameter instruction:

Parameter name	Type	instruction
packageName	String	Application package name
returnCode	int	Refer to error codes
msg	String	message

8.6、DeleteCallback

The callback of deleted application

Function name: public onDeleteFinished(String packageName, int returnCode, String msg);

Description: deleted

Parameter instruction:

Parameter name	Type	instruction
packageName	String	Application package name
returnCode	int	Refer to error codes
msg	String	message

8.7、CopyCallback

The callback of copied file and content

Function name: `public void onCopyProgress(int progress);`

Description: copy progress

Parameter instruction:

Parameter name	Type	instruction
progress	int	progress

Function name: `public void onCopyFinished(int returnCode, String msg);`

Description: copied

Parameter instruction:

Parameter name	Type	instruction
returnCode	int	Refer to error codes
msg	String	message

8.8、DataCallback

The callback of serial port data read

Function name: `public void onDataReceive(byte[] buffer, int size);`

Description: receive serial port data

Parameter instruction:

Parameter name	Type	instruction
buffer	byte[]	data
size	int	The length of data

8.9、WiegandCallback

The callback of data read by Wiegand

Function name: `public void onReadData(String data);`

Description: read data

Parameter instruction:

Parameter name	Type	instruction
data	String	the data read by Wiegand

8.10、BackUpCallback

The callback of backup application data

Function name: `public void onBackUpProgress(int progress);`

Description: backup progress

Parameter instruction:

Parameter name	Type	instruction
progress	int	progress

Function name: `public void onBackUpFinished(int returnCode, String msg);`

Description: the backup has finished

Parameter instruction:

Parameter name	Type	instruction
returnCode	int	Refer to error codes
msg	String	message

Function name: `public void onBackUpPath(String path);`

Description: Return to the path of backup

Parameter instruction:

Parameter name	Type	instruction
----------------	------	-------------

path	String	path
------	--------	------

8.11、RecoveryCallback

The callback of recovery of backup application data

Function name: public void onRecoveryProgress(int progress);

Description: recovery progress

Parameter instruction:

Parameter name	Type	instruction
progress	int	progress

Function name: public void onRecoveryFinished(int returnCode, String msg);

Description: has recovered

Parameter instruction:

Parameter name	Type	instruction
returnCode	int	Refer to error codes
msg	String	message

Function name: public void onRecoveryPath(String path);

Description: Return to the path of backup recovery

Parameter instruction:

Parameter name	Type	instruction
path	String	path

8.12、LogCallback

The callback of capturing systematic logs line by line

Function name: public void onSingleLine(String log);

Description: callback each line of systematic logs

Parameter instruction:

Parameter name	Type	instruction
log	String	Progress

8.13、UpdateCallback

The callback of OTA-AB update

Function name: public void onStatusUpdate(int status, float percent);

Description: callback the update status and progress

Parameter instruction:

Parameter name	Type	instruction
status	int	status UpdateStatusConstants
percent	float	progress

Function name: public void onPayloadApplicationComplete(int errCode);

Description: callback the updated status

Parameter instruction:

Parameter name	Type	instruction
errCode	int	Return codes ErrorCodeConstants

9、Outdated Interface

9.1、Overview

The outdated API will be marked as updated and deprecated version, and it is not recommended to use it after that. If there is a replacement method, the new method is recommended. The general interface will be completely eliminated after three iterations of version, and the interface will be completely unavailable after elimination. Please refer to the complete outdated version for details.

- Outdated API list

Function name	Description	Eliminated date	Eliminated version	Complete elimination
net_setWifiDefaultPassword	set the defaulted connection with the passwords of WiFi after startup	2022-05-19	V2.6.0	V2.9.0

net_getWifiDefaultPassword	acquire the defaulted connection with the passwords of WiFi after startup	2022-05-19	V2. 6. 0	V2. 9. 0
info_setModel	Set the device model	2022-09-14	V2. 14. 0	V2. 14. 0
info_setFactoryCompany	Set the device producer	2022-09-14	V2. 14. 0	V2. 14. 0
info_setSoftwareVersion	set the version of the device software	2022-09-14	V2. 14. 0	V2. 14. 0
sys_setInVolume	set the switch status of microphone	2022-10-25	V2. 16. 0	V2. 16. 0
sys_getInVolume	Acquire the switch status of microphone	2022-10-25	V2. 16. 0	V2. 16. 0
disp_getScreenModel	Acquire the dual-screen with different displays	2022-10-25	V2. 16. 0	V2. 16. 3

9.2、net_setWifiDefaultPassword

Function name: public int net_setWifiDefaultPassword(String account, String pwd, int type)

Description: set the defaulted connection with the passwords of WiFi after startup

API version: the version before V2.6.0 can be replaced by the version after V2.6.0

Replace method:[net_setWifiConnect](#)

Parameter:

Parameter name	Type	instruction
account	String	Account name
pwd	String	passwords
type	String	way of encryption: 0:no passwords 1:WEP 2:WPA

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- Restart to take effect after setting
- Enter null on the account name to cancel the setting.

9.3、net_getWifiDefaultPassword

Function name: public String net_getWifiDefaultPassword()

Description: acquire the defaulted connection with the passwords of WiFi after startup

API version: Eliminate the version after V2.6.0, while the version before V2.6.0 still can be used.

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	String	Return the strings with commas in the middle: account + "," + password + "," + encryption mode

Note

- For example: return value: smdt, 123456, 1 represents the account is smdt, with its passwords-123456, its encryption type is WEP
- Encryption method: 0: no passwords 1: WEP 2: WPA

9.4、 info_setSoftwareVersion

Function name: public int info_setSoftwareVersion(String verison)

Description: set the version number of equipment software

API version: it is outdated, please use the burning tool to burn

Parameter:

Parameter name	Type	Instruction
verison	String	equipment software version number

Return Parameter instruction

Parameter name	Type	Instruction
Return value	int	Calling results refers to error codes

- Note: Restart to take effect after setting

9.5、 info_setModel

Function name: public int info_setModel(String modelname)

Description: set the equipment model

API version: it is outdated, please use the burning tool to burn

Parameter:

Parameter name	Type	instruction
modelname	String	Equipment model

Return Parameter instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- Restart to take effect after setting

9.6、info_setFactoryCompany

Function name: public int info_setFactoryCompany(String company)

Description: set the device producer

API version: it is outdated, please use the burning tool to burn

Parameter:

Parameter name	Type	instruction
company	String	Device producer

Return Parameter instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- Restart to take effect after setting

9.7、sys_setInVolume

Function name: public int sys_setInVolume(int type, boolean enable)

Description: set the switch status of microphone

API version: Eliminate the version after V2.16.0

Replace method:[sys_setAudioInput](#)

Parameter:

Parameter name	Type	instruction
type	int	Device type: 0:the microphone on the headphone 1:the microphone on the motherboard 2:the microphone on the USB
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	Calling results refers to error codes

Note

- The interface is not available and is not supported for the time being.

9.8、sys_getInVolume

Function name: public int sys_getInVolume(int type)

Description: Acquire the switch status of microphone on the device

API version: eliminate the version after V2.16.0

Replace method:[sys_getAudioInput](#)

Parameter:

Parameter name	Type	instruction
type	int	Device type: 0: 0:the microphone on the headphone 1:the microphone on the motherboard 2:the microphone on the USB

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	1:open 0:close

- **Note** The interface is not available and is not supported for the time being.

9.9、 disp_getScreenModel

Function name: public int disp_getScreenModel()

Description: acquire the combination of dual-screen with different displays

API version: eliminate the version after V2.16.0

Parameter: No

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	0:single display 4:dual displays by default

10、 Linux

10.1、 Brief introduction

Introduction to Linux OS-API interface

- The following interfaces are consistent with the Android API interface Parameter name. Please search for the following function names directly. The function names or Parameter that are different from Android are placed in the difference interface directory.

Return value	Function name	Description
char*	info_getSoftwareVersion	acquire software version number
char*	info_getKernelVersion	acquire the released message of the kernel
char*	info_getMCUVersion	acquire MCU version number
char*	info_setFactoryCompany	acquire 目前 API version number
char*	info_getApiVersion	set the version number of equipment software
char*	info_getTotalMemory	Acquire the total memory capacity of the device
char*	info_getTotalStorage	Acquire the total storage capacity of the device
char*	info_getSerialNumber	acquire array number

int	sys_getControl	Other IO related Settings (board status light, module power supply)
int	sys_getWatchDog	acquire the switch status of the watch dog
int	sys_setWatchDogFeed	feed on the watch dog
int	sys_setPowerOff	System shutdown
int	sys_setReboot	System reboot
int	custom_setRelayIoMode	set mode and delay of the relay
int	custom_getRelayIoMode	acquire the current mode of relay
int	disp_setLcdBackLight(int values)	Set the lightness of backlight
int	disp_getLcdBackLight()	acquire the lightness of backlight
int	disp_getLcdPwmFrequency()	Acquire the frequency of backlight
char*	net_getMacAddress()	acquire Ethernet mac
int	sys_setGpioDirection(int direction, int value, char *pData, int lenth)	Set the input and output status of GPIO
int	sys_getGpioDirection(char *pData, int lenth)	acquire the input and output status of GPIO
int	sys_getGpioValue(char *pData, int lenth)	acquire the level status of GPIO
int	sys_setControl(int type, int values)	Other IO related Settings (board status light, module power supply)
int	sys_setAutoPowerOnOff(char off_h, char off_m, char on_h, char on_m, char enable)	Set timing switchers
int	sys_setWatchDog(char enable)	Set the switch status of the door dog
int	dev_setUsbPower(int num, int values)	Set the switch status of the USB mouth
int	dev_getUsbPower(int num)	acquire the switch status of the USB mouth
int	dev_getUsbPower(int num)	acquire the switch status of the USB mouth
int	dev_setLedLighted(int led, int value)	Set the status of the three-color light
int	dev_getLedState(int led)	acquire the status of the three-color light
int	custom_sendWiegandCardHIDPID(char HID_value, char PID_value, char transformat)	Wiegand' s using hidden code and public code send card number
char*	custom_readWiegandData()	Read Wiegand input
int	custom_releaseWiegandRead()	Withdraw from Wiegandand input blocking

10.2、 Version Revision Record

- The API version of Linux OS-API released to the public

version	Description	Date
V1.0	established	2021-01-04
V2.0	updated	2022-10-24

10.3、 The usage method of API

Acquire object instance method

Platform : Linux

Function name: `const char * info_getApiVersion();`

Description: acquire current API version number

Parameter:

Parameter name	Type	Instruction
context	Context	context

Return Parameter instruction

Parameter name	Type	Instruction
Return value	char*	character string

Example

```
char * str = info_getApiVersion();  
Output result : V1.0.0-release
```

Add the library file

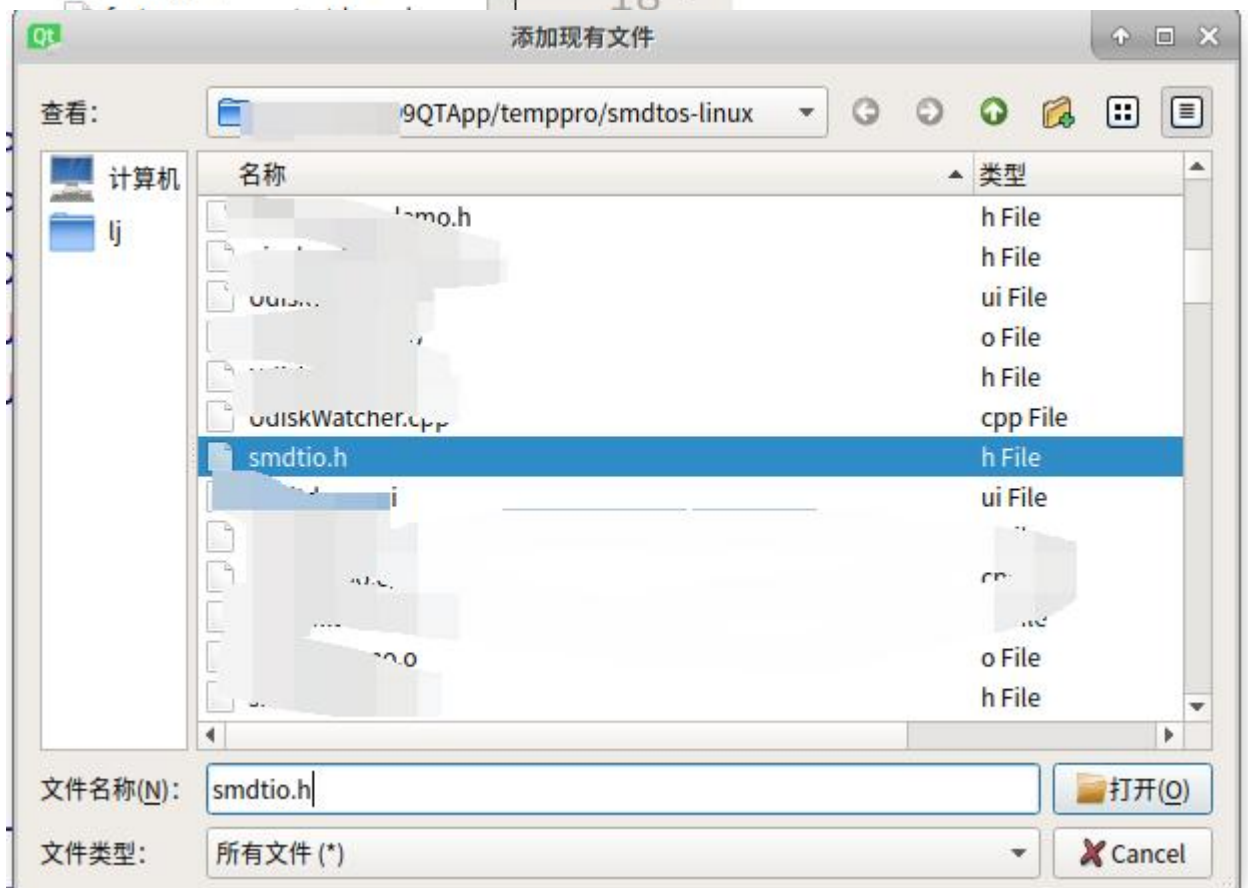
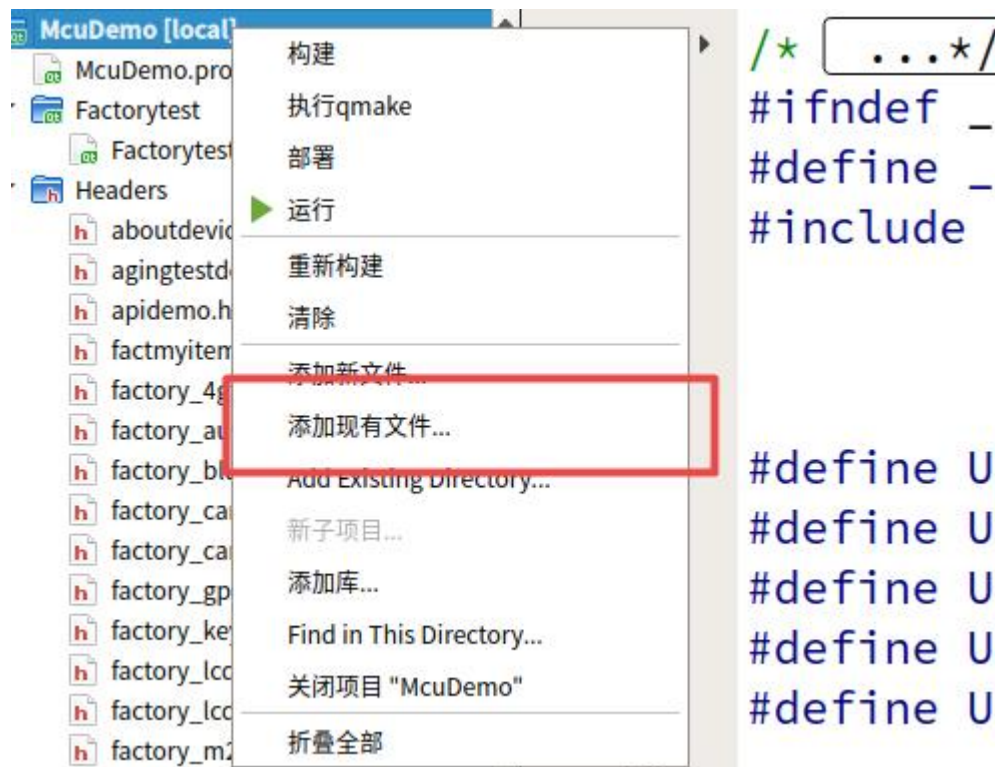
Linux Qt

1. Copy `smdtio.h`, `libsmdtio.so` to the engineering content;

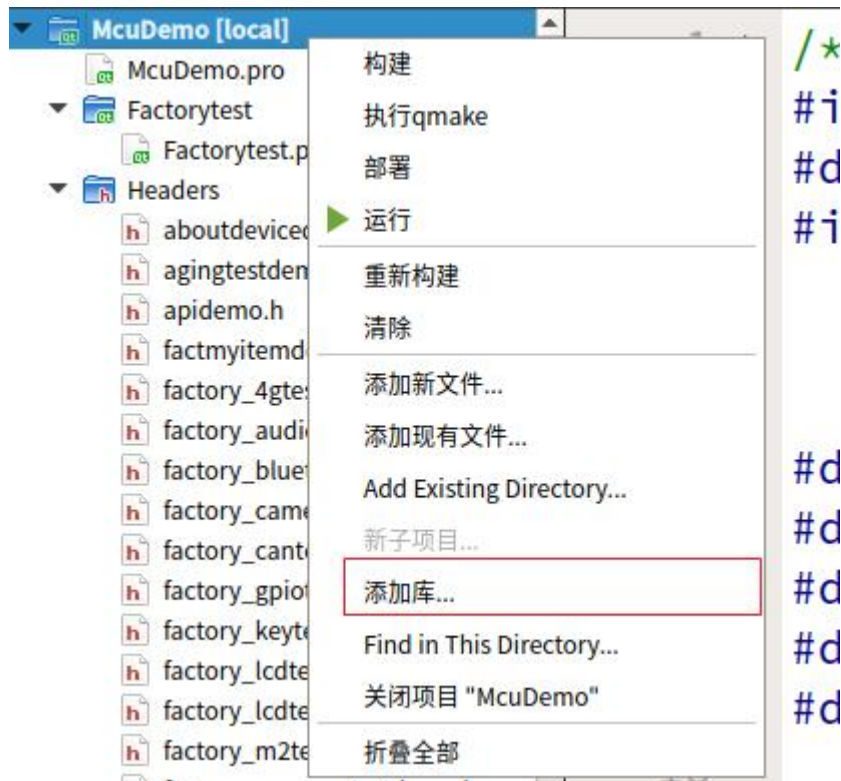
```
libsmdtio.so smdtio.h  
[App/test/smdtos-linux$
```

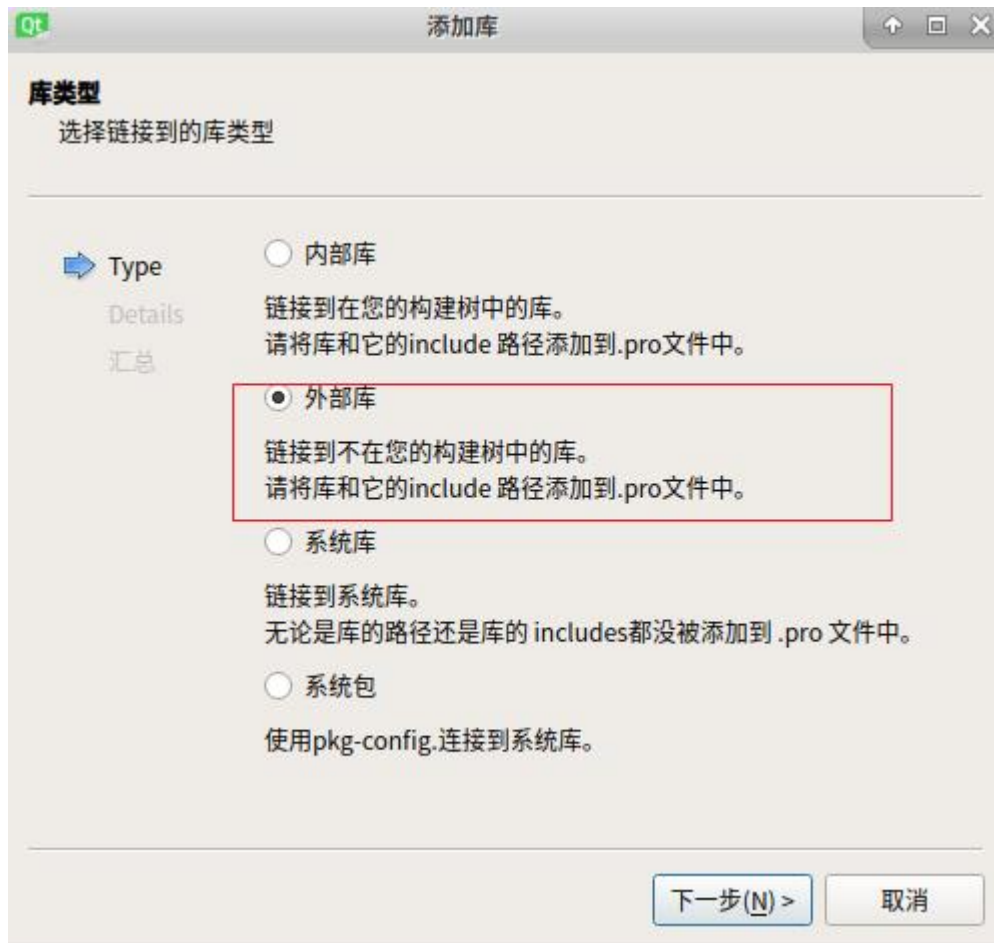
`smdtio.h`:

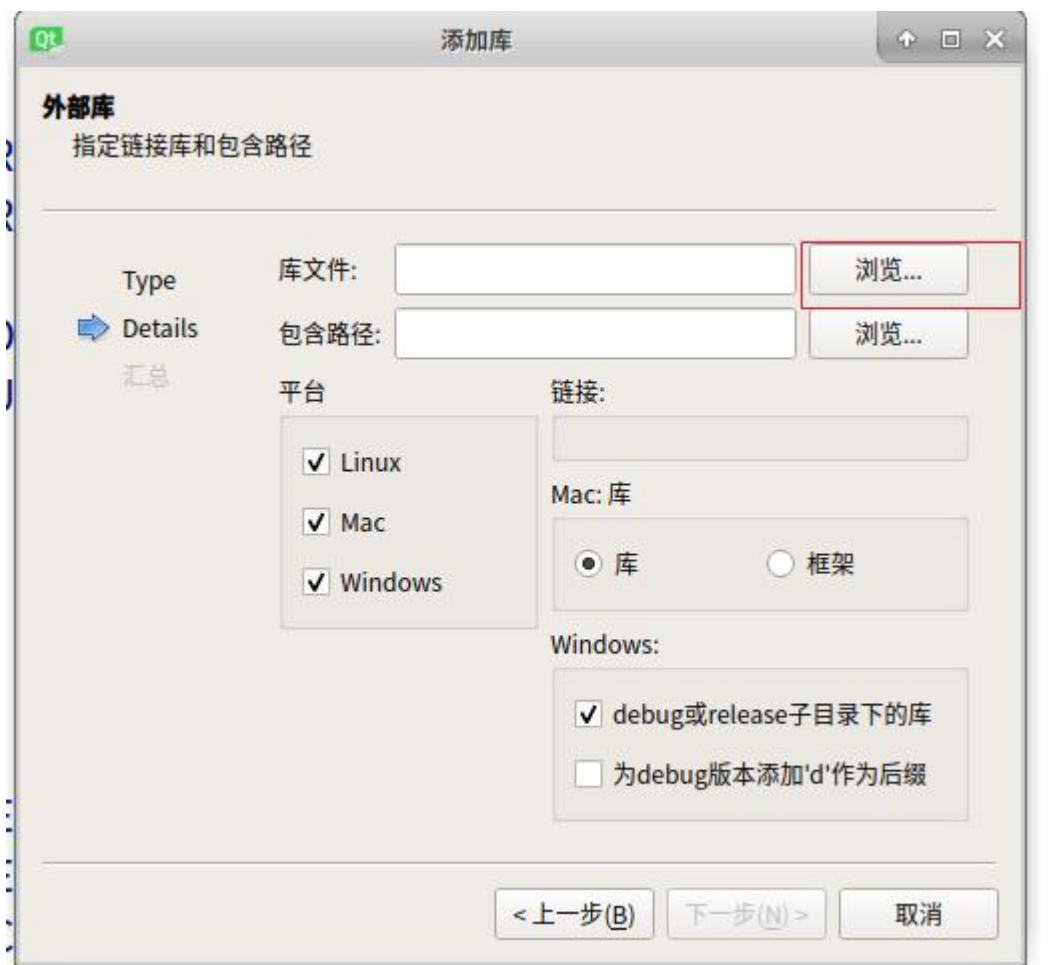
2. Add the first file and find



3. Select the project and right-click to add the library. Add it according to the following steps;







4. Then call the method after the instance object is instantiated in the code.

10.1、Special interface

10.1.1、sys_setBluetooth

Function name: `char *info_getRtcDateTime();`

Platform: Linux

Description: acquire the time and data information of rtc

Parameter:

Parameter name	Type	instruction
enable	boolean	true:open false:close

Return Parameter Instruction

Parameter name	Type	instruction
----------------	------	-------------

Return value	char*	Calling results, return toRTC information
--------------	-------	---

Note

There are parameter differences between this interface and Android interface.

10.1.2、 disp_setLcdPwmFrequency

Function name: int disp_setLcdPwmFrequency(int frequency, char ratio);

Platform: Linux

Description: set the frequency of backlight

Parameter:

Parameter name	Type	Instruction
frequency	int	frequency
ratio	char	Ratio

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	int	The call result is 0 for success and less than 0 for failure

10.1.3、 gpio_export

Function name: int gpio_export(int pin)

Platform: Linux

Description: Set pin to export

Parameter:

Parameter name	Type	instruction
pin	int	pin number

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	0:succeed, less than 0:failed

10.1.4、 gpio_unexport

Function name: int gpio_unexport(int pin)

Platform: Linux

Description: cancel the usage of pin to export

Parameter:

Parameter name	Type	instruction
pin	int	pin number

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	0:succeed, less than 0:failed

10.1.5、 gpio_direction

Function name: int gpio_direction(int pin, int dir)

Platform: Linux

Description: set the input and output of GPIO

Parameter:

Parameter name	Type	instruction
pin	int	pin number
dir	int	0 input, 1 output

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	0:succeed, less than 0:failed

10.1.6、 gpio_readdirection

Function name: int gpio_readdirection(int pin)

Platform: Linux

Description: read the input and output of GPIO

Parameter:

Parameter name	Type	instruction
pin	int	pin number

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	int	0:input 1:output

10.1.7、 gpio_write

Function name: `int gpio_write(int pin, char *buffer)`

Platform: Linux

Description: set the value of GPIO

Parameter:

Parameter name	Type	Instruction
pin	int	pin number
buffer	char *	0:low level, 1:high level

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	int	More than 0:succeed, 0: failed

10.1.8、 gpio_read

Function name: `int gpio_read(int pin);`

Platform: Linux

Description: read the value of GPIO

Parameter:

Parameter name	Type	instruction
pin	int	pin number

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	0:low level, 1:high level

10.1.9、sys_setTimerSwitchOffJson

Function name: `int sys_setTimerSwitchOffJson();`

Platform: Linux

Description: acquire the shutdown time

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	shutdown time

10.1.10、sys_setTimerSwitchOnJson

Function name: `int sys_setTimerSwitchOnJson();`

Platform: Linux

Description: acquire the startup time

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	startup time

10.1.11、sys_setUtcPowerOnOff

Function name: `int sys_setUtcPowerOnOff(int year, int mon, int offday, int offhour, int offmin, int onday, int onhour, int onmin, int enable);`

Platform: Linux

Description: set UTC power off at the fixed time

Parameter:

Parameter name	Type	instruction
year	int	year
mon	int	month
offday	int	Off days
offhour	int	Off hours
offmin	int	Off minutes
onday	int	On days

onhour	int	On hours
onmin	int	On minutes
enable	int	1, 3: set, other value:cancel

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	0:succeed, less than 0:failed

10.1.12、 sys_setUtcPowerOnOff

Function name: int sys_setUtcPowerOnOff(time_t timesoff, time_t timeson, int enable);

Platform: Linux

Description: set UTC power off at the fixed time

Parameter:

Parameter name	Type	instruction
timesoff	time_t	off time structural body
timeson	time_t	on time structural body
enable	int	1, 3:enable, other value:cancel

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	0:succeed, less than 0:failed

10.1.13、 dev_readEeprom

Function name: char* dev_readEeprom(int deviceId, int areaId, int start_addr, int size)

Platform: Linux

Description: read and write the EEPROM

Parameter:

Parameter name	Type	instruction
deviceId	int	Equipment ID, starting with 0, each ID represents one EEPROM
areaId	int	Area ID, starting with 1, each ID represents one area of EEPROM.
start_addr	int	Start address, means reading data from that area

size	int	the length of data that has been read
------	-----	---------------------------------------

Return Parameter Instruction

Parameter name	Type	instruction
Return value	char*	data that has been read

10.1.14、dev_readEeprom_3399

Function name: char* dev_readEeprom_3399(int deviceId, int areaId, int start_addr, int size)

Platform: Linux

Description: read and write the EEPROM

Parameter:

Parameter name	Type	instruction
deviceId	int	Equipment ID, starting with 0, each ID represents one EEPROM
areaId	int	Area ID, starting with 1, each ID represents one area of EEPROM.
start_addr	int	Start address, means reading data from that area
size	int	the length of data that has been read

Return Parameter Instruction

Parameter name	Type	instruction
Return value	char*	data that has been read

10.1.15、dev_writeEeprom

Function name: int dev_writeEeprom(int deviceId, int areaId, int start_addr, int size, char * pData);

Platform: Linux

Description: read and write the EEPROM

Parameter:

Parameter name	Type	instruction
deviceId	int	Equipment ID, starting with 0, each ID represents one EEPROM
areaId	int	Area ID, starting with 1, each ID represents one area of EEPROM.
start_addr	int	Start address, means reading data from that area
size	int	the length of data that has been read

pData	char*	the length of data that has been read
-------	-------	---------------------------------------

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	The length of data that has been written in

10.1.16、 dev_writeEeprom_3399

Function name: int dev_writeEeprom_3399(int deviceId, int areaId, int start_addr, int size, char * pData);

Platform: Linux

Description: read and write the EEPROM

Parameter:

Parameter name	Type	instruction
deviceId	int	Equipment ID, starting with 0, each ID represents one EEPROM
areaId	int	Area ID, starting with 1, each ID represents one area of EEPROM.
start_addr	int	Start address, means reading data from that area
size	int	the length of data that has been read
pData	char*	the length of data that has been read

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	The length of data that has been written in

10.1.17、 custom_setRelayIoValue

Function name: int custom_setRelayIoValue(int value)

Platform: Linux

Description: set the size of relay

Parameter:

Parameter name	Type	instruction
value	int	pin number

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	0:succeed, less than 0:failed

10.1.18、 custom_getRelayIoValue

Function name: int custom_getRelayIoValue()

Platform: Linux

Description: acquire the size of relay

Parameter:

Parameter name	Type	instruction
value	int	pin number

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	0:succeed, less than 0:failed

10.1.19、 dev_startFlag_on

Function name: int dev_startFlag_on()

Platform: Linux

Description: set MCU

Return Parameter Instruction

Parameter name	Type	instruction
Return value	int	0:succeed, less than 0:failed

10.1.20、 dev_startFlag_off

Function name: int dev_startFlag_off()

Platform: Linux

Description:set MCU

Return Parameter Instruction

Parameter name	Type	instruction
----------------	------	-------------

Return value	int	0:succeed, less than 0:failed
--------------	-----	-------------------------------

10.1.21、dev_getregval

Function name: unsigned char dev_getregval(unsigned char value)

Platform: Linux

Description: acquire the size of the relay

Parameter:

Parameter name	Type	instruction
value	unsigned char	Register acquired

Return Parameter Instruction

Parameter name	Type	instruction
Return value	unsigned char	The value of the register

10.1.22、dev_setregval

Function name: int dev_setregval(unsigned char reg, unsigned char value)

Platform: Linux

Description: set the value of mcu register

Parameter:

Parameter name	Type	instruction
reg	unsigned char	register
value	unsigned char	set the value of mcu register

Return Parameter Instruction

Parameter name	Type	instruction
Return value	unsigned char	register

10.1.23、dev_getMCUstatus

Function name: int dev_getMCUstatus()

Platform: Linux

Description: acquire the status of mcu

Return Parameter Instruction

Parameter name	Type	Instruction
Return value	int	the status of mcu

10.1.24、info_getArch

Function name: char *info_getArch()

Platform: Linux

Description: acquire the architecture

Return Parameter Instruction

Parameter name	Type	instruction
Return value	char*	architecture

10.1.25、sys_getSystemName

Function name: char * sys_getSystemName();

Platform: Linux

Description: acquire the system name

Return Parameter Instruction

Parameter name	Type	instruction
Return value	char*	System name

10.1.26、info_getRtcDateTime

Function name: char * info_getRtcDateTime();

Platform: Linux

Description: acquire the time and date information of RTC

Return Parameter Instruction

Parameter name	Type	instruction
Return value	char*	RTC information