

Label Printer

LPAPI Interface Description

Android Version

V2.2

DothanTech Thanmore

8th Jan. 2026

Contents

Contents	2
Document modification records	5
1. LPAPI Interface Description	7
2. Simple introduction to use LPALPI	7
3. Create LPAPI object	8
4. Related introduction methods of LPAPI.Callback	8
4.1 onProgressInfo Callback function when Bluetooth device state changes	11
4.2 onStateChanged	12
4.3 onPrintProgress	13
4.4 onPrinterDiscovery	13
5. Related methods to connect the printer	14
5.1 isDeviceSupported	14
5.2 isPrinterSupported	14
5.3 getAllPrinters	15
5.4 getAllPrinterAddresses	15
5.5 getFirstPrinter	16
5.6 getFirstPrinterAddress	16
5.7 OpenPrinter Open the printer with the specified name or model asynchronously	17
5.8 OpenPrinter Open the specified Bluetooth device asynchronously	17
5.9 OpenPrinterByAddress Open the specified printer object asynchronously	18
5.10 OpenPrinterSync Open the printer with the specified name or model synchronously	18
5.11 openPrinterByAddressSync	18
5.12 getPrinterName	19
5.13 getPrinterInfo	19
5.14 getPrinterState	20
5.15 isPrinterOpened	20
5.16Cancel Cancel the print job	21
5.17 closePrinter	21
5.18 reopenPrinter	21
5.19 reopenPrinterSync	22
5.20 printBitmap Print Bitmap Image	22
5.21 waitPrinterState wait for printer state	23
6. Related methods of USB connection to printer	25
6.1 hasUsbPrinter	26
6.2 hasUsbPrinter	26
6.3 GetUsbPrinter	27

6.4	getUsbPrinter	27
7.	The method introduction of print jobs and pages	27
7.1	startJob start the drawing job	27
7.2	abortJob cancel the drawing job	28
7.3	commitJob commit param to print	28
7.4	commitJobWithParam	28
7.5	startPage start to draw a page	29
7.6	endPage end drawing the page	29
7.7	endJob end drawing the job	29
7.8	getJobPages	30
8.	Settings or obtaining the printer content parameters	30
8.1	setPrintPageGapType	31
8.2	setPrintPageGapLength	31
8.3	setPrintDarkness	32
8.4	setPrintSpeed	32
8.5	setDrawParam	32
8.6	setItemOrientation	32
8.7	getItemOrientation	33
8.8	setItemHorizontalAlignment	33
8.9	getItemHorizontalAlignment	33
8.10	setItemVerticalAlignment	33
8.11	getItemVerticalAlignment	34
9.	Description of drawing text string method	34
9.1	Uniform font used when drawing text	34
9.2	drawText	35
9.3	drawTextRegular	35
9.4	drawRichText	36
9.5	drawTextWithIndent	37
9.6	drawTextWithScale	38
9.7	measureFontHeight	39
10.	Description of drawing barcode method	40
10.1	draw1DBarcode	41
10.2	draw2DQRCode	41
10.3	draw2DDataMatrix	42
11.	Description of drawing vector graphic method	42
11.1	drawRectangle	42
11.2	fillRectangle	43
11.3	drawRoundRectangle	43
11.4	fillRoundRectangle	44

11.5 drawEllipse	44
11.6 fillEllipse	45
11.7 drawCircle	45
11.8 fillCircle	46
11.9 drawLine	46
11.10 drawDashLine	47
12. Description of drawing image method	48
12.1 drawImage	48
12.2 drawImageWithActualSize	48
12.3 drawImageWithThreshold	49
12.4 drawBitmap	49
12.5 drawBitmapWithActualSize	50
12.6 drawBitmapWithThreshold	50
12.7 drawBitmapStream	51
12.8 drawBitmapStreamWithActualSize	52
12.9 drawBitmapStreamWithThreshold	52

Document modification records

Num				Modification date
1	V0.1	Document establishment	YangLingmei	27th Apr,2017
2	V0.2	StartJob function, remove the DPI parameter	YangLingmei	15th May,2017
3	V0.3	Update synchronously according to the actual code	HuDianxing	12th Aug,2017
4	V1.0	Update synchronously according to the latest code	HuDianxing	10th Oct,2017
5	V1.1	Document detection and optimization	YangLingmei	11th Nov,2017
6	V1.2	Add documents related to LPAPI.Callback function	HuDianxing	30th Nov,2017
7	V1.3	Add setDrawParam function	YangLingmei	20th Mar,2018
8	V1.4	Add hardwareFlags and other attributes in PrinterInfo	YangLingmei	22th Oct,2018
9	V1.5	Add documents related to print parameters	HuDianxing	30th Mar,2019
10	V1.6	Modify document format	YangLingmei	1st Apr,2019
11	V1.7	PrintParamName, Add horizontal and vertical offsets to the print parameters	YangLingmei	9th Sep,2019
12	V1.8	ANTI_COLOR parameter, indicate whether to print in anti-color	YangLingmei	11th Oct,2019
13	V1.9	Add isPrinterSupported parameter to judge whether supported printers	YangLingmei	27th Nov,2019
14	V2.0	Add ANTI_COLOR and FONT_WRAP parameter to setDrawParam function	YangLingmei	24th Apr,2020

15	V2.1	Add relevant document of USB connection print	YangLingmei	22th Jan,2021
16	V2.1	Added permission descriptions Adjusted LPAPI.Callback callback Added setPrintPageGapType function to set gap type Added setPrintPageGapLength function to set gap length (gap distance) Added setPrintDarkness function to set print darkness Added setPrintSpeed function to set print speed Added getJobPages function to obtain the image array of the completed jobs Added draw2DDataMatrix function to draw DataMatrix code	Yang Zongyan	12th Dec,2023
17	V2.1	Adapt to Android14	Yang Zongyan	17th Apr,2024
18	V2.1	Adjust the second parameter of LPAPI.Callback's onPrintProgress function from Object type to PrintData type (click to view) Added the function of PAGE_KEY to set the print page identification, which can be set when submitting a job (click to view)	Yang Zongyan	2nd Aug,2024
19	V2.1	Add 3 types when draw barcode: CODE128_A、 CODE128_B、 CODE128_C	Yang Zongyan	21 st Oct, 2024
20	V2.2	Add details of reasons for print failures	Yang Zongyan	14 th Feb 2025
21	V2.2	Support printer function search Add example of print custom table templates in the demo Add example of batch print in the demo	Yang Zongyan	8 th Jan 2026

1. LPAPI Interface Description

The LPAPI interface description document is an interface description proposed for label drawing, convenient for users to call interface in the secondary development, shorten the development cycle and speed up the development progress.

The LPAPI interface provides caller with a simple method to complete label drawing. This interface provides a variety of methods for drawing objects such as text, 1D bar code, 2D codes, images, and various graphics. While it can also perform the rotation of the drawing object and the drawing page, the caller can also call the method to obtain the finished label image for label preview, which makes label drawing easier.

2. Simple introduction to use LPALPI

1. Apply for permissions

The following permissions need to be applied for in AndroidManifest.xml

Permissions:

```
<uses-permission android:name="android.permission.BLUETOOTH" />
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
// Android 12 and above also request the following two permissions
<uses-permission android:name="android.permission.BLUETOOTH_SCAN" />
<uses-permission android:name="android.permission.BLUETOOTH_CONNECT" />
```

Android 12 and above request dynamic permission before using Bluetooth function

Request activity permission

```
// Request activity permission
```

```
private void requestPermission() {
    String[] permissions;
    if (Build.VERSION.SDK_INT < 31) {
        permissions = new String[] { Manifest.permission.BLUETOOTH
    };
    } else {
        permissions = new String[] { Manifest.permission.BLUETOOTH,
        Manifest.permission.BLUETOOTH_SCAN,
        Manifest.permission.BLUETOOTH_CONNECT,
    };
    }
}
```

2.The basic calling process of the LPAPI interface is as follows:

Code

```
// Obtain the LPAPI object
LPAPI api = LPAPI.Factory.createInstance();
// Connect to the first paired printer
api.openPrinter(null);
// Start drawing task, input parameters (page width, page height)
api.startJob(40, 30, 0);
// Start the drawing of a page, task,, draw a text string
api.drawText(content, 4, 5, 40, 30, 4);
// End the drawing task and commit for drawing
api.commitJob();
```

3. Create LPAPI object

There is a public static class Factory in the LPAPI interface for creating LPAPI objects. The method is as follows

Function Introduction	Create LPAPI Objects	
Method	public static LPAPI createInstance(Callback callback)	
Parameter Name	callback	Related callback functions in the printing process, please refer to: LPAPI.Callback ;
Return Value	LPAPI Object	
Introduction	Each call to this method will create a new LPAPI object.	

4. Related introduction methods of LPAPI.Callback

[ProgressInfo](#) Progress Information:

ProgressInfo	Value	Description
ProgressInfo.AdapterEnabling	0	Turn on the Bluetooth adapter; addInfo is Boolean data, indicates whether it is relevant to this object;
ProgressInfo.AdapterEnabled	1	The Bluetooth adapter has been turned on; addInfo is Boolean data, indicates whether it is relevant to this object;
ProgressInfo.AdapterDisabled	2	The Bluetooth adapter is not turned on; addInfo is Boolean data, indicates whether it is relevant to this object;

ProgressInfo. DeviceBonding	3	Printer pairing in progress; addilInfo is PrinterAddress data, indicates the target printer;
ProgressInfo. DeviceBonded	4	The printer has been paired successfully; addilInfo is PrinterAddress data, indicates the target printer;
ProgressInfo.DeviceUnbonded	5	Bonded printer has canceled; addilInfo is PrinterAddress data, indicates the target printer;

PrinterState Printer State :

PrinterState	Value	Description
PrinterState.Connecting	0	Connecting to the printer;
PrinterState.Connected	1	The printer is connected;
PrinterState.Connected2	2	The printer is connected, connect it again to test if the connection works? If valid, this state message will be issued;
PrinterState. Printing	3	Printing...;
PrinterState. Working	4	Other working besides printing;
PrinterState.Disconnected	5	The printer is disconnected;

PrintProgress Print Progress :

PrintProgress	Value	Description
PrintProgress.Connected	0	The printer is connected, addilInfo is PrinterAddress data, indicates the target printer;
PrintProgress.StartCopy	1	The printer has started printing, addilInfo is Integer data, indicates the page copy number to start printing, starting from 1;
PrintProgress.DataEnded	2	Printer data transfer has ended, addilInfo is Integer data, indicates the page copy number to start printing, starting from 1;
PrintProgress.Success	3	The printer prints successfully., addilInfo is Integer data, indicates the number of pages printed successfully;
PrintProgress. Failed	4	The printer printing failed, addilInfo is PrintFailReason data, indicates the failure reason;

PrintFailReason Print Progress:

PrintFailReason	Value	Description
PrintFailReason.OK	0	Non-failure reason, for placeholder only;
PrintFailReason.IsPrinting	1	The printer is printing;
PrintFailReason.IsRotating	2	The printer motor is turning;
PrintFailReason.Cancelled	3	The print job is canceled;
PrintFailReason.EnvNotReady	4	The printing environment is not currently available;
PrintFailReason.VolTooLow	5	The power level is too low, please charge the device first and continue printing when the power level is sufficient;
PrintFailReason.VolTooHigh	6	The power level is too high, please discharge the device first and continue printing after the power level drops a little;
PrintFailReason.TphNotFound	7	The print head is not detected;
PrintFailReason.TphTooHot	8	The use temperature is too high, please let it stand for a few minutes before continuing to print;
PrintFailReason.CoverOpened	9	The top cover is not closed, please close the top cover before continuing to print.
PrintFailReason.No_Paper	10	No paper detected. If you have run out of paper, replace it with new before continuing to print; If you have not run out of paper, please check the adjusting plates in the paper compartment if they fix the paper correctly;
PrintFailReason.TphOpened	11	The ribbon is not locked, please tighten the lock before continuing to print;
PrintFailReason.No_Ribbon	12	Ribbon is not detected, please install the ribbon before continuing to print ;
PrintFailReason.Unmatched_Ribbon	13	The ribbon currently in use does not match the printer, please replace the ribbon with a matching one before continuing to print;.
PrintFailReason.TphTooCold	14	The use temperature is too low, place the printer indoors or in a warmer location for use;

PrintFailReason.Usedup_Ribbon	15	Ribbon has been used up, please replace the ribbon with a new one before continuing to print.
PrintFailReason.Usedup_Ribbon2	16	Label cartridge has been used up, please replace the ribbon with a new one before continuing to print.
PrintFailReason.No_Label	17	No label paper detected, if you have run out of label paper, replace with a new roll before continuing to print;
PrintFailReason.Unmatched_Label	18	The label roll currently in use does not match the printer, please replace with a matching roll of before continuing to print.
PrintFailReason.Usedup_Label	19	If you have run out of label paper, replace with a new roll before continuing to print;
PrintFailReason.No_Ribbon2	20	Label cartridge is not detected, please install it before continuing to print ;
PrintFailReason.Unmatched_Ribbon2	21	The current label cartridge does not match the printer, please replace with a matching one before continuing to print.
PrintFailReason.LabelCannotOpen	22	The label cartridge is not locked, please tighten the lock before continuing to print.
PrintFailReason.Disconnected	23	Printer is disconnected
PrintFailReason.Timeout	24	Timeout error
PrintFailReason.Other	25	Other unknown errors

GeneralProgress Current general progress of searching for printers:

GeneralProgress	Value	Description
GeneralProgress.Start	0	Indicates starting printer search, and address is empty at this time;
GeneralProgress.Success	1	Indicates the printer search has ended, and address is empty at this time;
GeneralProgress.Canceled	4	Indicates the printer search process was canceled, and address is empty at this time;
GeneralProgress.Timeout	5	Indicates the printer searched previously has not been searched again for a long time. Usually it is because the printer has been turned off, or connected to another mobile phone;
GeneralProgress.Info	6	Indicates the printer has been searched a printer;

4.1 onProgressInfo Callback function when Bluetooth device

Function Introduction	This callback function will be called when the Bluetooth device state changes.	
Method	void onProgressInfo(ProgressInfo info, Object addInfo);	
Parameter Name	info	Bluetooth device or printer pairing state;
Parameter Name	addInfo	Additional information when the Bluetooth device state changes, please refer to : ProgressInfo ;
Return Value	None	
Introduction	None	
Notice		

4.2 onStateChanged

Callback function when printer state changes

Function Introduction	This callback function will be called when the Bluetooth device state changes.	
Method	void onStateChanged(PrinterAddress address, PrinterState state);	
Parameter Name	address	The target printer object;
Parameter Name	state	Current printer state, please refer to: PrinterState ;
Return Value	none	
Introduction	none	
Notice		

4.3 onPrintProgress

Callback function when the printing progress changes

Function Introduction	This callback function will be called when the printing progress changes.	
Method	void onPrintProgress(PrinterAddress address, PrintData printData, PrinterProgress progress, Object addilInfo);	
Parameter Name	address	The Bluetooth address object of the target printer;
Parameter Name	printData	printData.PrintObj, Print data object, Bitmap object or IAtESCPOS object; printData.printParam, Printing parameters input when printing, Bundle; printData.printCopy, The serial number of the printing copies, starting from 1; printData.pageKey, The page ID actually used when printing (1-65535, it will restart from 1 after exceeding 65535) ;
Parameter Name	progress	The current printing progress;
Parameter Name	addilInfo	For additional information about the printing progress, please refer to: PrintProgress ;
Return Value	None	
Introduction	None	
Notice	The second parameter of this function is changed from Object type to PrintData type. The original return object of Object type is changed to printData now.PrintObj (refer to the introduction of this function parameter)	

4.4 onPrinterDiscovery

Callback function when scanning to the printer via NFC

Function Introduction	This function will be called when the phone scans to the printer via NFC.
Method	void onPrinterDiscovery(PrinterAddress address, ObjectaddilInfo);

Parameter Name	address	The target printer object; for the searched printer, its type is DiscoveredPrinter, which contains more search related information; the printer information scanned by NFC, its type is PrinterAddress.
Parameter Name	addInfo	PrinterInfo, when NFC scans, it indicates the printer information scanned by NFC.; GeneralProgress, additional information about searching the printing progress, please refer to: GeneralProgress
Return Value	None	
Introduction	None	
Notice		

5. Related methods to connect the printer

5.1 isDeviceSupported

Function Introduction	Is it a supported printer if return the specific Bluetooth device	
Method	public boolean isDeviceSupported(BluetoothDevice printer, String modelName);	
Parameter Name	printer	Bluetooth device;
Parameter Name	modelName	Printer model name, It can be a list of multiple printer model names, separated by English semicolons; the model name starts with English minus sign, indicates the model is excluded; a null or empty string indicates no restriction on the printer model.
Return Value	true: indicates the supported printer; false: indicates the printer is not supported;	
Introduction	None	
Notice		

5.2 isPrinterSupported

Function Introduction	Is it a supported printer if return the specific Bluetooth device name	
Method	public boolean isPrinterSupported(String printer, String modelName);	
Parameter Name	printer	Bluetooth device name;
Parameter Name	modelName	Printer model name, It can be a list of multiple printer model names, separated by English semicolons; the model name starts with English minus sign, indicates the model is excluded; a null or empty string indicates no restriction on the printer model.
Return Value	true: indicates the supported printer;	

	False: indicates the print is not supported;
Introduction	None
Notice	

5.3 getAllPrinters

Get all printers of the specified model name in the paring list

Function Introduction	Get all printer objects of the specified model name in the paring list	
Method	public String getAllPrinters(String modelName);	
Parameter Name	modelName	Printer model name, It can be a list of multiple printer model names, separated by English semicolons; the model name starts with English minus sign, indicates the model is excluded; a null or empty string indicates no restriction on the printer model.
Return Value	Returns all printer names that meet requirements, different name separated by English ","	
Introduction		
Notice		

5.4 getAllPrinterAddresses

Get all printers of the specified model name in the paring list

Function Introduction	Get all printer objects of the specified model name in the paring list	
Method	public List<PrinterAddress> getAllPrinterAddresses(String modelName);	
Parameter Name	modelName	Printer model name, It can be a list of multiple printer model names, separated by English semicolons; the model name starts with English minus sign, indicates the model is excluded; a null or empty string indicates no restriction on the printer model.
Return Value	Returns all printers that meet requirement.	
Introduction		
Notice		

5.5 getFirstPrinter

Get the first printer of the specified model name in the pairing list

Function Introduction	Get the first printer of the specified model name in the pairing list	
Method	public String getFirstPrinter(String modelName);	
Parameter Name	modelName	Printer model name, It can be a list of multiple printer modelnames, separated by English semicolons; the model name starts with English minus sign, indicates the model is excluded; a null or empty string indicates no restriction on the printer model.
Return Value	<u>If there is no printer of the corresponding model in the pairing list, return null, otherwise return the first printer name.</u>	
Introduction		
Notice	Please make sure the Bluetooth is turned on before using, and the corresponding printer is paired and turned on.	

5.6 getFirstPrinterAddress

Get the first printer of the specified model name in the pairinglist

Function Introduction	Get the first printer object of the specified model name in the pairing list	
Method	public PrinterAddress getFirstPrinterAddress(String modelName);	
Parameter Name	modelName	Printer model name, It can be a list of multiple printer modelnames, separated by English semicolons; the model name starts with English minus sign, indicates the model is excluded; a null or empty string indicates no restriction on the printer model.
Return Value	<u>If there is no printer of the corresponding model in the pairing list, return null, otherwise return the first printer object.</u>	
Introduction		
Notice	Please make sure the Bluetooth is turned on before using, and the corresponding printer is paired and turned on.	

5.7 OpenPrinter Open the printer with the specified name or model asynchronously

Function Introduction	Open the printer with the specified name or model asynchronously	
Method	public boolean openPrinter(String modelName);	
Parameter Name	modelName	Printer model name, It can be a list of multiple printer modelnames, separated by English semicolons, the model name supports regular expressions; the model name starts with English minus sign, indicates the model is excluded; a null or empty string indicates no restriction on the printer model.
Return Value	Success or failure	
Introduction	If the specified printer name is null or an empty string, open the first installed printer. If the specified printer is not found, it returns failed.	
Notice	Before starting a printing task, this method must be called first, or the same series of methods.	

5.8 OpenPrinter Open the specified Bluetooth device asynchronously

Function Introduction	Open the specified Bluetooth device asynchronously	
Method	public boolean openPrinter(BluetoothDevice device);	
Parameter Name	device	Bluetooth devices are supported by the interface, Bluetooth devices are invalid if not provided by the manufacturer.
Return Value	Success or failure; true indicates the target printer is supported by the interface, otherwise it is false.	
Introduction	Connect the printer	
Notice	Before starting a printing task, this method must be called first, or the same series of methods	

5.9 OpenPrinterByAddress Open the specified printer object asynchronously

Function Introduction	Open the specified printer object asynchronously	
Method	public boolean openPrinterByAddress(PrinterAddress address);	
Parameter Name	address	The target printer object
Return Value	Success or failure	
Introduction	If the specified printer is not found, it returns failed.	
Notice		

5.10 OpenPrinterSync Open the printer with the specified name or model synchronously

Function Introduction	Open the printer with the specified name or model synchronously	
Method	public boolean openPrinterSync(String modelName);	
Parameter Name	modelName	Printer model name, It can be a list of multiple printer model names, separated by English semicolons, the model name supports regular expressions; the model name starts with English minus sign, indicates the model is excluded; a null or empty string indicates no restriction on the printer model.
Return Value	Success or failure	
Introduction	If the specified printer name is null or an empty string, open the first installed printer. If the specified printer is not found, it returns failure.	
Notice	Please make sure the printer is powered on, and the pairing operation has been performed.	

5.11 openPrinterByAddressSync

Open the specified printer object synchronously

Function	Open the specified printer object synchronously
----------	---

Introduction		
Method	public boolean openPrinterByAddressSync(PrinterAddress address);	
Parameter Name	address	User-specified parameters of PrinterAddress type;
Return Value	Success or failure	
Introduction	If the specified printer is not found, it returns failure..	
Notice		

5.12 getPrinterName

Get the connected printer name

Function Introduction	Get the connected printer name	
Method	public String getPrinterName();	
Parameter Name	None	
Return Value	If the printer is connected, the corresponding printer name will be returned, otherwise an empty string will be returned.	
Introduction		
Notice		

5.13 getPrinterInfo

Get the details information of the connected printer

Function Introduction	Get the details information of the connected printer	
Method	public PrinterInfo getPrinterInfo();	
Parameter Name	None	
Return Value	Examples of information related to the current printer.	
Introduction	Common attributions in PrinterInfo: deviceType: DEVICE_TYPE_xxx, thermal printer/thermal transfer printer, etc deviceName: device name, including model and series numbers deviceVersion: hardware version number softwareVersion: software version number deviceAddress: MAC address deviceAddrType: BLE/SPP/WiFi etc	

	deviceDPI: print resolution, DPI deviceWidth: print width, mm hardwareFlags: hardware flags, use this value to match different firmware upgrade packages. softwareFlags: software flags
Notice	Please make sure the printer is powered on successfully.(connected)

5.14 getPrinterState

Get the printer connection states

Function Introduction	Get the printer connection states
Method	public PrinterState getPrinterState();
Parameter Name	None
Return Value	The connection state of the current printer, please refer to: PrinterState
Introduction	
Notice	

5.15 isPrinterOpened

Judge whether the current printer is opened and connected successfully

Function Introduction	Judge whether the current printer is opened on and connected successfully
Method	public boolean isPrinterOpened();
Parameter Name	None
Return Value	true: indicates the printer has been connected successfully and can print; false: indicates the printer is not connected, the printer needs to be connected before printing;
Introduction	
Notice	

5.16 Cancel Cancel the print job

Function Introduction	Cancel the print job
Method	public void cancel();
Parameter Name	None
Return Value	None
Introduction	Cancel the current printing, used to execute the cancel operation after committing the print job,
Notice	

5.17 closePrinter

Disconnect the current connected printer

Function Introduction	Disconnect the current connected printer
Method	public void closePrinter();
Parameter Name	None
Return Value	None
Introduction	After printing is completed, this function can be used to disconnect the current connected printer;
Notice	Try to avoid calling this function before printing is completed, otherwise the printing job may be terminated.

5.18 reopenPrinter

Reopen the last connected printer asynchronously

Function Introduction	Reopen the last connected printer asynchronously
Method	public boolean reopenPrinter();
Parameter Name	None

Return Value	Success or failure
Introduction	Reconnect the last connected printer.
Notice	Returning success only indicates that the operation is committed, and does not indicate the connection is successful. The specific connection result will be notified through the callback function.

5.19 reopenPrinterSync

Reopen the last connected printer synchronously

Function Introduction	Reopen the last connected printer synchronously
Method	public boolean reopenPrinterSync();
Parameter Name	None
Return Value	Success or failure
Introduction	Reconnect the last connected printer.
Notice	

5.20 printBitmap Print Bitmap Image

Function Introduction	Print bitmap image	
Method	public boolean printBitmap(Bitmap bitmap, Bundle printParam);	
Parameter Name	bitmap	The image to be printed;
	printParam	Printing parameters, if it is null, the default printing parameters will be used for printing;; Parameters type can refer to: PrintParamName ;
Return Value	Success or failure	
Introduction		
Notice	1、The background color of the bitmap object cannot be transparent, it is recommended to use white background color; 2、Returning success only indicates that the operation is committed, and does not indicate the connection is successful. The specific connection result will be notified through the callback function.	

5.21 waitPrinterState wait for printer state

Function	wait for printer state	
Introduction		
Method	public boolean waitPrinterState(printerState state, int millis);	
Parameter Name	state	Desired printing states;
	millis	Waiting timeout in milliseconds, less than 0 indicates keeping waiting;
Return Value	true : after specified time, the printer connection state is the given connection state; false : after specified time, the printer connection state is inconsistent with the given connection state;	
Introduction	This method is usually used to determine whether the printer is connected successfully after connecting to the printer asynchronously. Because the printer connection is time-consuming, the connection may not success immediately after initiating the printer connection request, in this case, this function is needed to determine the printer connection state;	
Notice		

PrintParamName :

PrintParamName	Value	Description
PRINT_DARKNESS	"PRINT_DENSITY"	Used to set the print darkness, the vale refer to PrintParamValue ;
PRINT_SPEED	"PRINT_SPEED"	Used to set the print speed, the value refer to PrintParamValue ;
PRINT_DIRECTION	"PRINT_DIRECTION"	Used to set the print direction, the value is 0, 90, 180, 270;
PRINT_COPIES	"PRINT_COPIES"	Used to set the print copies, the value is any valid integer;
GAP_TYPE	"GAP_TYPE"	Used to set the paper type, the value refer to PrintParamValue ;
GAP_LENGTH_01MM	"GAP_LENGTH_01MM"	Used to set the length of paper gap, the unit is 0.01mm
HORIZONTAL_OFFSET_01MM	"HORIZONTAL_OFFSET_01MM"	Used to set the horizontal offset, the unit is mm;
HORIZONTAL_OFFSET_PX	"HORIZONTAL_OFFSET_PX"	Used to set the horizontal offset, unit pixel;
VERTICAL_OFFSET_01MM	"VERTICAL_OFFSET_01MM"	Used to set the vertical offset, the unit is mm;
VERTICAL_OFFSET_PX	"VERTICAL_OFFSET_PX"	Used to set the vertical offset, unit pixel;
ANTI_COLOR	"ANTI_COLOR"	Used to indicate whether it is "inverted color printing";

PAGE_KEY	"PAGE_KEY"	When committing a print job in
		commitJobWithParam, it is used to set the page ID actually used for printing (1-65535, and it will restart from 1 after exceeding 65535);

ThanMore

PrintParamValue

PrintParamValue	Value	Description
MIN_PRINT_DARKNESS	0	The reference value of print darkness, range 0 to 14, 0 indicates the minimum print darkness, 14 indicates the maximum print darkness, 5 indicates the default print darkness;
DEFAULT_PRINT_DARKNESS	5	
MAX_PRINT_DARKNESS	14	
MIN_PRINT_SPEED	0	The reference value of print speed, range 0 to 4, 0 indicates the minimum print speed, 2 indicates the maximum print speed, 4 indicates the default print speed;
DEFAULT_PRINT_SPEED	2	
MAX_PRINT_SPEED	4	
GAP_NONE	0	Paper type: paper type is default, set with the printer settings;
GAP_HOLE	1	Paper type: use label paper with position holes for positioning;
GAP_GAP	2	Paper type: use gap paper for printing;
GAP_BLACK	3	Paper type: use black label paper for printing;

6. Related methods of USB connection to printer

The method of USB connection requires connecting the mobile phone and the printer device through the OTG data cable. The USB connection will only prompt to access the device when the application is opened in the foreground, and will not prompt when entering the background. After the USB connection is

successful, it will automatically disconnect the current printer device connected via Bluetooth, and automatically identify and get the printer device connected via USB. You can also connect the specified printer object by calling [connect \(PrinterAddress.USB\)](#); Automatically reconnects to the last printer device connected via Bluetooth after USB disconnection

6.1 hasUsbPrinter

Returns whether there is a printer device connected to the phone via USB

Function Introduction	Returns whether there is a printer device connected to the phone via USB
Method	public static boolean hasUsbPrinter();
Parameter Name	None
Return Value	true: indicates the supported printers; false: indicates unsupported printers;
Introduction	None
Notice	

6.2 hasUsbPrinter

Return and check whether there is a printer device connected to the phone via USB

Function Introduction	Start the drawing job.	
Method	public static boolean hasUsbPrinter(String prefixName);	
Parameter Name	prefixName	The prefix name of printer type, such as: DP23, DT60, DP30 Label Printer.
Return Value	true: Indicates the supported printers; false: Indicates the unsupported printers;	
Introduction	None	
Notice		

6.3 GetUsbPrinter

Get the currently available USB connected printers

Function Introduction	Return and check whether there is a printer device connected to the phone via USB
Method	public static PrinterAddress getUsbPrinter();
Parameter Name	None
Return Value	Get the currently available USB connected printers, return null indicates no printer is connected to the phone via USB.
Introduction	None
Notice	

6.4 getUsbPrinter

Get the currently available USB connected printers

Function Introduction	Start the drawing job	
Method	public static PrinterAddress getUsbPrinter(String prefixName);	
Parameter Name	prefixName	The prefix name of printer type, such as DP23, DT60, DP30 Label Printer.
Return Value	Get the currently available USB connected printers, return null indicates no printer is connected to the phone via USB.	
Introduction	None	
Notice		

7. The method introduction of print jobs and pages

7.1 startJob start the drawing job

Function Introduction	start the drawing job	
Method	public boolean startJob(double width, double height, int orientation);	
Parameter Name	width	Label width (based on the drawing view, without considering of label rotation. Unit mm)

	height	Label height (based on the drawing view, without considering of label rotation. Unit mm)
	orientation	The drawing job's label page rotates in a clockwise direction, 0/90/180/270, the default is 0.
Return Value	Success or failure	
Introduction	This method shall be called when starting a drawing job.	
Notice	When starting a drawing job, all previous jobs or data will be discarded.	

7.2 abortJob cancel the drawing job

Function Introduction	Cancel the drawing job	
Method	public void abortJob();	
Parameter Name	None	
Return Value	None	
Introduction	Cancel the drawing task, used to cancel all drawing operations before commit.	
Notice	All the current jobs or data will be discarded, but the parameters settings will be remained.	

7.3 commitJob commit param to print

Function Introduction	Commit param to print	
Method	public boolean commitJob();	
Parameter Name	None	
Return Value	Success or failure	
Introduction	Commit the printing job to do the real printing.	

7.4 commitJobWithParam

Function Introduction	Commit param to print	
Method	public boolean commitJobWithParam(Bundle printParam);	
Parameter Name	printParam	Set the print parameters in Bundle format, for the parameters types, please refer to: PrintParamName ;

Return Value	Success or failure
Introduction	Commit the printing job and set the print parameters

7.5 startPage start to draw a page

Function Introduction	Start to draw a page
Method	public boolean startPage();
Parameter Name	None
Return Value	Success or failure
Introduction	Start a print page

7.6 endPage end drawing the page

Function Introduction	End drawing the page
Method	public void endPage();
Parameter Name	None
Return Value	None
Introduction	End drawing the page

7.7 endJob end drawing the job

Function Introduction	End drawing the job
Method	public void endJob();
Parameter Name	None
Return Value	None
Introduction	End drawing the job

7.8 getJobPages

Get the image param of completed job page

Function Introduction	End drawing the job
Method	<code>public List<Bitmap> getJobPages();</code>
Parameter Name	None
Return Value	A Bitmap collection of the completed job interfaces
Introduction	Get the image param of the completed job page, used after endJob().

8. Settings or obtaining the printer content parameters

ItemAlignment Types of drawing barcode:

ItemAlignment	Value	Description
ItemAlignment.LEFT	0	Left alignment
ItemAlignment.CENTER	1	Horizontally centered / Vertically centered
ItemAlignment.RIGHT	2	Right alignment
ItemAlignment.TOP	0	Top alignment
ItemAlignment.MIDDLE	1	Vertically centered / Horizontally centered
ItemAlignment.BOTTOM	2	Bottom alignment
ItemAlignment.SAMEASITEM	3	The alignment of the object' s sub-elements are the same as the object alignment, currently used in the horizontal alignment of the 1D barcode text

DrawParamName Types of drawing font:

DrawParamName	Description
DrawParamName.FONT_NAME	The name of the font file stored in the assets directory of the project: the vale of FONT_NAME is String type. Such as: HeiTi.ttf. Please note that the font file name is not the font name necessarily. The .ttf will be added automatically if parameters value does not specify a suffix
DrawParamName.QRCODE_VERSION	The code version of QRCode (1~40) ; QRCODE_VERSION value is the Integer type. For specific meaning, please refer to the coding standards of QRCode. When the version number is not specified, it

	indicates the appropriate one will be adopted automatically according to the code content.
DrawParamName.ERROR_CORRECTION	Error correction level of QRCode (0~3) ; The ERROR_CORRECTION value is Integer type. The default is 0. For label printing, it is recommended to use level 0, setting the black dots as large as possible, so that the QR code is easier to scan and recognize;
DrawParamName.CHARACTER_SET	String encoding type of QRCode ; CHARACTER_SET value is String type. The default is UTF-8
DrawParamName.MARGIN	Margin of barcode; The MARGIN value is Integer type. The margin value defined in the QRCode encoding specification is 0/2/4. The default is 0, that is, no blank-leaving;
DrawParamName.ANTI_COLOR	Whether the text is ANTI_COLOR; The ANTI_COLOR value is Boolean type. The default is false, that is, not ANTI_COLOR
DrawParamName.FONT_WRAP	Whether the text wraps; The FONT_WRAP value is of Boolean type. The default is true, that is, the text will wrap automatically;

8.1 setPrintPageGapType

Function Introduction	Set print page gap type	
Method	public void setPrintPageGapType(int gapType);	
Parameter Name	gapType	Gap type, 0: Receipt; 1: Positioning hole; 2: Label; 3: Cardstock
Return Value	无	

8.2 setPrintPageGapLength

Function Introduction	Set print page gap length	
Method	public void setPrintPageGapLength(int gapLength);	
Parameter Name	gapLength	Label gap length, unit is mm;
Return Value	None	

8.3 setPrintDarkness

Function Introduction	Set print darkness	
Method	public void setPrintDarkness(int darkness);	
Parameter Name	darkness	Print darkness, range 0-15, 0: base on printer settings; 1: lightest; 6: normal; 15: darkest;
Return Value	None	

8.4 setPrintSpeed

Function Introduction	Set print speed	
Method	public void setPrintSpeed(int speed);	
Parameter Name	speed	Print speed, value range 0-5, 0: base on printer settings; 1: slowest; 3: normal; 5: fastest;
Return Value	None	

8.5 setDrawParam

Function Introduction	Set the related draw param	
Method	public void setDrawParam(String name, Object value);	
Parameter Name	name	Param name, Defined in the static string variable of the DrawParamName interface. please refer to DrawParamName
	value	The value, refer to different param names, please refer to DrawParamName
Return Value	None	

8.6 setItemOrientation

Function Introduction	Set the clockwise rotation angle of the drawing item	
Method	public void setItemOrientation(int orientation);	
Parameter Name	orientation	Clockwise rotation angle of subsequent drawing item (0/90/180/270), the default is 0.
Return Value	None	

8.7 getItemOrientation

Function Introduction	Get the rotation angle of the current drawing item
Method	public int getItemOrientation();
Parameter Name	None
Return Value	Get the rotation angle of the current drawing item (0/90/180/270)

8.8 setItemHorizontalAlignment

Function Introduction	Set item horizontal alignment
Method	public void setItemHorizontalAlignment(int alignment);
Parameter Name	alignment The horizontal alignment of subsequent drawing actions (ItemAlignment), the default is ItemAlignment.LEFT .
Return Value	None
Introduction	The horizontal direction is based on the horizontal drawing direction of the subsequently drawn item, which follows the clockwise rotation angle of the drawn item.

8.9 getItemHorizontalAlignment

Function Introduction	Get the horizontal alignment of the current drawing action
Method	public int getItemHorizontalAlignment();
Parameter Name	None
Return Value	the horizontal alignment of the current drawing action (ItemAlignment)
Introduction	The horizontal direction is based on the horizontal drawing direction of the subsequently drawn item, which follows the clockwise rotation angle of the drawn item.

8.10 setItemVerticalAlignment

Function Introduction	Set the vertical alignment of subsequent drawing actions
Method	public void setItemVerticalAlignment(int alignment);
Parameter	alignment The vertical alignment of subsequent drawing actions (ItemAlignment),

Name		the default is ItemAlignment.TOP .
Return Value	None	
Introduction	The vertical direction is based on the vertical drawing direction of the subsequently drawn item, which follows the clockwise rotation angle of the drawn item.	

8.11 getItemVerticalAlignment

Function Introduction	Get the vertical alignment of the current drawing action	
Method	public int getItemVerticalAlignment();	
Parameter Name	None	
Return Value	the vertical alignment of the current drawing action (ItemAlignment)	
Introduction	The vertical direction is based on the vertical drawing direction of the subsequently drawn item, which follows the clockwise rotation angle of the drawn item.	

9. Description of drawing text string method

[FontStyle](#)

FontStyle	Value	Description
FontStyle.REGULAR	0	Normal
FontStyle.BOLD	1	Bold
FontStyle.ITALIC	2	Italic
FontStyle.BOLDITALIC	3	Bold italic
FontStyle.UNDERLINE	4	Underline
FontStyle.STRIKEOUT	8	Strike through

9.1 Uniform font used when drawing text

There are great differences in the default fonts of different brands and mobile phones in the Android system. therefore, for systems which require a higher consistency of the printed labels, it is necessary to unify the fonts used when drawing text. Unifying text fonts can be done by the following two methods:

1. If one font is used in the entire label, please directly put the font file in the assets directory of the Android APP project and change the file name to FONT.ttf. The interface function for drawing text will use this font to draw. If the font file cannot be found, the text drawing function will use the default font;
2. If multiple fonts are used in the entire label, please put multiple font files that need to be used in the assets directory of the Android APP project (for font files with Chinese names, it is recommended to change the file name to English, because some Android system does not support the Chinese language named files in the assets directory very well),and specify the font file by invoking

[setDrawParam\(IAtBitmap.DrawParamName.FONT_NAME,"xxx"\)](#) to draw them before drawing text.

9.2 drawText

Function Introduction	Drawing a text string	
Method	public boolean drawText(String text, double x, double y, double width, double height, double fontHeight);	
Parameter Name	text	The text string needs to be drawn.
	x	The horizontal position of the upper left corner of the drawn textbox (unit: mm)
	y	The vertical position of the upper left corner of the drawn textbox (unit: mm)
	width	The horizontal width of the drawn textbox (unit:mm). If width is 0, the drawn text will be aligned left, center, and right based on the display width of the text and the current alignment mode with x as the reference point. The default is 0.
	height	The vertical height of the drawn textbox (unit:mm). If the height is 0, it will align the top, middle and bottom with y as the base point according to the current alignment and the display height of the drawn text. The default is 0
	fontHeight	Font size(unit:mm)
Return Value	Success or failure	
Introduction	● If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, and then draw a text string. ● The upper left corner vertical position, horizontal width, and vertical height of the drawing are based on the vertical directions of the current drawing page. ● fontHeight is in millimeters as unit. The conversion formula for font size and millimeter is: Font size = mm*72/25.4 For example: size 9 is 3.175mm, size 12 is 4.233mm ● Refer to setDrawParam to set the parameters related to the text font.	

9.3 drawTextRegular

Function Introduction	Drawing a text string
Method	public boolean drawTextRegular(String text, double x, double y, double width,

	double height, double fontHeight, int fontStyle, float lineSpace);	
Parameter Name	text	The text string needs to be drawn.
	x	The horizontal position of the upper left corner of the drawn textbox (unit: mm)
	y	The vertical position of the upper left corner of the drawn textbox (unit: mm)
	width	The horizontal width of the drawn textbox (unit:mm). If width is 0, the drawn text will be aligned left, center, and right based on the display width of the text and the current alignment mode with x as the reference point. The default is 0.
	height	The vertical height of the drawn textbox (unit:mm). If the height is 0, it will align the top, middle and bottom with y as the base point according to the current alignment and the display height of the drawn text. The default is 0
	fontHeight	Font size(unit:mm)
	fontStyle	The font style under the initial state, normal, bold, italic, bold italic, underline, strike through (FontStyle)
	lineSpace	Line spacing, the default is 0, 1 means double the line spacing of the current font.
Return Value	Success or failure	
Function Introduction	● If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, and then draw a text string. ● The upper left corner vertical position, horizontal width, and vertical height of the drawing are based on the vertical directions of the current drawing page. ● fontHeight is in millimeters as unit. The conversion formula for font size and millimeter is: Font size = mm*72/25.4 For example: size 9 is 3.175mm, size 12 is 4.233mm ● Refer to setDrawParam to set the parameters related to the text font.	

9.4 drawRichText

Function Introduction	Draw text string with font style setting symbol	
Method	public boolean drawRichText(String text, double x, double y, double width, double height, double fontHeight, int fontStyle);	
Parameter	text	Need to draw text string with font style setting symbol.

Name	x	The horizontal position of the upper left corner of the drawn textbox (unit: mm)
	y	The vertical position of the upper left corner of the drawn textbox (unit: mm)
	width	The horizontal width of the drawn textbox (unit:mm). If width is 0, the drawn text will be aligned left, center, and right based on the display width of the text and the current alignment mode with x as the reference point. The default is 0.
	height	The vertical height of the drawn textbox (unit:mm). If the height is 0, it will align the top, middle and bottom with y as the base point according to the current alignment and the display height of the drawn text. The default is 0
	fontHeight	Font size(unit:mm)
	fontStyle	The font style under the initial state, normal, bold, italic, bold italic, underline, strike through (FontStyle)
Return Value	Success or failure	
Function Introduction	● If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, and then draw a text string. ● The upper left corner vertical position, horizontal width, and vertical height of the drawing are based on the vertical directions of the current drawing page. ● fontHeight is in millimeters as unit. The conversion formula for font size and millimeter is: Font size = mm*72/25.4 For example: size 9 is 3.175mm, size 12 is 4.233mm ● This method will set the font style of the drawn text according to the font style setting symbol which contained in the string data. ● Refer to setDrawParam to set the parameters which related to the text font.	

9.5 drawTextWithIndent

Function Introduction	Draw a text string with indentation function	
Method	public boolean drawTextWithIndent(String text, double x, double y, double width, double height, double fontHeight, int fontStyle, floatlineSpace);	
Parameter Name	text	Need to draw text string with font style setting symbol.
	x	The horizontal position of the upper left corner of the drawn textbox (unit: mm)
	y	The vertical position of the upper left corner of the drawn textbox (unit: mm)

	width	The horizontal width of the drawn textbox (unit:mm). If width is 0, the drawn text will be aligned left, center, and right based on the display width of the text and the current alignment mode with x as the reference point. The default is 0.
	height	The vertical height of the drawn textbox (unit:mm). If the height is 0, it will align the top, middle and bottom with y as the base point according to the current alignment and the display height of the drawn text. The default is 0
	fontHeight	Font size(unit:mm)
	fontStyle	The font style under the initial state, normal, bold, italic, bold italic, underline, strike through (FontStyle)
	text	Need to draw text string with font style setting symbol.
Return Value	Success or failure	
Function Introduction	● If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, and then draw a text string. ● The upper left corner vertical position, horizontal width, and vertical height of the drawing are based on the vertical directions of the current drawing page. ● fontHeight is in millimeters as unit. The conversion formula for font size and millimeter is: $\text{Font size} = \text{mm} * 72 / 25.4$ For example: size 9 is 3.175mm, size 12 is 4.233mm ● This method will set the font style of the drawn text according to the font style setting symbol which contained in the string data. ● Refer to setDrawParam to set the parameters which related to the text font.	

9.6 drawTextWithScale

Function Introduction	Draw a text string with scale function	
Method	public boolean drawTextWithScale(String text, double x, double y, double width, double height, double fontHeight, int fontStyle, float lineSpace, boolean autoIndent);	
Parameter Name	text	Need to draw text string with font style setting symbol.
	x	The horizontal position of the upper left corner of the drawn textbox (unit: mm)
	y	The vertical position of the upper left corner of the drawn textbox (unit: mm)
	width	The horizontal width of the drawn textbox (unit:mm). If width is 0, the drawn text will be aligned left, center, and right based on the display

		width of the text and the current alignment mode with x as the reference point. The default is 0.
	height	The vertical height of the drawn textbox (unit:mm). If the height is 0, it will align the top, middle and bottom with y as the base point according to the current alignment and the display height of the drawn text. The default is 0
	fontHeight	Font size(unit:mm)
	fontStyle	The font style under the initial state, normal, bold, italic, bold italic, underline, strike through (FontStyle)
	text	Need to draw text string with font style setting symbol.
	autoIndent	true: When a colon (":") is detected in the string, the parts before and after the colon are aligned respectively.
Return Value	Success or failure	
Function Introduction	● If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, and then draw a text string. ● The upper left corner vertical position, horizontal width, and vertical height of the drawing are based on the vertical directions of the current drawing page. ● fontHeight is in millimeters as unit. The conversion formula for font size and millimeter is: Font size = mm*72/25.4 For example: size 9 is 3.175mm, size 12 is 4.233mm ● This method will set the font style of the drawn text according to the font style setting symbol which contained in the string data. ● Refer to setDrawParam to set the parameters which related to the text font.	

9.7 measureFontHeight

Function Introduction	Measure the height of the text string	
Method	public double measureFontHeight(String text, double x, double y, double width, double height, double fontHeight, float lineSpace);	
Parameter Name	text	Need to draw text string with font style setting symbol.
	x	The horizontal position of the upper left corner of the drawn textbox (unit: mm)
	y	The vertical position of the upper left corner of the drawn textbox (unit: mm)
	width	The horizontal width of the drawn textbox (unit:mm). If width is 0, the drawn text will be aligned left, center, and right based on the display

		width of the text and the current alignment mode with x as the reference point. The default is 0.
	height	The vertical height of the drawn textbox (unit:mm). If the height is 0, it will align the top, middle and bottom with y as the base point according to the current alignment and the display height of the drawn text. The default is 0
	fontHeight	Font size(unit:mm)
	lineSpace	Line spacing, the default is 0, 1 means double the line spacing of the current font.
Return Value	The height of the text string (unit: mm).	
Function Introduction	● If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, and then draw a text string. ● The upper left corner vertical position, horizontal width, and vertical height of the drawing are based on the vertical directions of the current drawing page. ● fontHeight is in millimeters as unit. The conversion formula for font size and millimeter is: Font size = mm*72/25.4 For example: size 9 is 3.175mm, size 12 is 4.233mm	

10. Description of drawing barcode method

[BarcodeType](#)

BarcodeType	Value	Description
BarcodeType.UPC_A	20	
BarcodeType.UPC_E	21	
BarcodeType.EAN13	22	
BarcodeType.EAN8	23	
BarcodeType.CODE39	24	
BarcodeType.ITF25	25	
BarcodeType.CODABAR	26	
BarcodeType.CODE93	27	
BarcodeType.CODE128	28	
BarcodeType.ISBN	29	
BarcodeType.ECODE39	30	
BarcodeType.AUTO	60	
BarcodeType.CODE128_A	61	
BarcodeType.CODE128_B	62	
BarcodeType.CODE128_C	63	

10.1 draw1DBarcode

Function Introduction	Draw a 1D barcode	
Method	public boolean draw1DBarcode(String text, int type, double x, double y, double width, double height, double textHeight);	
Parameter Name	text	The content of the 1D barcode needs to be drawn.
	type	1D barcode encoding type (BarcodeType), the default is AUTO
	x	The horizontal position of the upper left corner of the drawn 1D barcode .
	y	The vertical position of the upper left corner of the drawn 1D barcode (unit: mm).
	width	The overall display width of a 1D barcode.
	height	Display height of 1D barcode (including the readable text).
	textHeight	The height of the readable text (unit:mm), 3mm is recommended.
Return Value	Success or failure	
Function Introduction	- If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw a 1D barcode. - The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page.	

10.2 draw2DQRCode

Function Introduction	Draw a QR code	
Method	public boolean draw2DQRCode(String text, double x, double y, double width);	
Parameter Name	text	The content of the QR code needs to be drawn.
	x	The horizontal position of the upper left corner of the drawn QR code.
	y	The vertical position of the upper left corner of the drawn QR code (unit: mm).
	width	The horizontal width of the drawn QR code (unit mm).
Return Value	Success or failure	
Function Introduction	- If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw a QR code. - The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page.	

- Refer to the [setDrawParam](#) function to set the parameters which related to drawing QR code.

10.3 draw2DDataMatrix

Function Introduction	Draw 2D DataMatrix	
Method	public boolean draw2DDataMatrix(String text, double x, double y, double width);	
Parameter Name	text	The content of the 2D DataMatrix needs to be drawn.
	x	The horizontal position of the upper left corner of the drawn 2D DataMatrix.
	y	The vertical position of the upper left corner of the drawn 2D DataMatrix (unit: mm).
	width	The horizontal width of the drawn 2D DataMatrix (unit mm).
Return Value	Success or failure	
Function Introduction	- If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw a 2D DataMatrix. - The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page.	

11. Description of drawing vector graphic method

11.1 drawRectangle

Function Introduction	Draw a rectangle with the specified line width.	
Method	public boolean drawRectangle(double x, double y, double width, double height, double lineWidth);	
Parameter Name	x	The horizontal position of the upper left corner of the drawn rectangle (unit mm).
	y	The vertical position of the upper left corner of the drawn rectangle (unit mm).
	width	The horizontal width of the drawn rectangle (unit mm).
	height	The vertical height of the drawn rectangle (unit mm).
	lineWidth	The line width of the rectangle (unit: mm).The line width of the rectangle extends towards the inside of rectangle.
Return Value	Success or failure	

Function Introduction	- If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw a rectangle. - The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page.
------------------------------	--

11.2 fillRectangle

Function Introduction	Draw a filled rectangle with the specified line width	
Method	public boolean fillRectangle(double x, double y, double width, double height);	
Parameter Name	x	The horizontal position of the upper left corner of the drawn filled rectangle (unit mm).
	y	The vertical position of the upper left corner of the drawn filled rectangle (unit mm).
	width	The horizontal width of the drawn filled rectangle (unit mm).
	height	The vertical height of the drawn filled rectangle (unit mm).
Return Value	Success or failure	
Function Introduction	- If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw a rectangle. - The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page.	

11.3 drawRoundRectangle

Function Introduction	Draw a rounded rectangle with the specified line width	
Method	public boolean drawRoundRectangle(double x, double y, double width, double height, double cornerWidth, double cornerHeight, double lineWidth);	
Parameter Name	x	The horizontal position of the upper left corner of the drawn rounded rectangle (unit mm).
	y	The vertical position of the upper left corner of the drawn rounded rectangle (unit mm).
	width	The horizontal width of the drawn rounded rectangle (unit mm).
	height	The vertical height of the drawn rounded rectangle (unit mm).
	cornerWidth	Corner width (unit mm).

	cornerHeight	Corner height (unit mm). the default is the same as cornerWidth.
	lineWidth	The line width of the rounded rectangle (unit: mm).The linewidth of the rounded rectangle extends towards the inside of itself.
Return Value	Success or failure	
Function Introduction	● If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw a rounded rectangle. ● The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page.	

11.4 fillRoundRectangle

Function Introduction	Draw a filled rounded rectangle with the specified linewidth.	
Method	public boolean fillRoundRectangle(double x, double y, double width, double height, double cornerWidth, double cornerHeight);	
Parameter Name	x	The horizontal position of the upper left corner of the drawn filled rounded rectangle (unit mm).
	y	The vertical position of the upper left corner of the drawn filled rounded rectangle (unit mm).
	width	The horizontal width of the drawn filled rounded rectangle (unit mm).
	height	The vertical height of the drawn filled rounded rectangle (unit mm).
	cornerWidth	Corner width (unit mm).
	cornerHeight	Corner height (unit mm). the default is the same as cornerWidth.
Return Value	Success or failure	
Function Introduction	● If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw a filled rounded rectangle. ● The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page.	

11.5 drawEllipse

Function Introduction	Draw a ellipse with the specified line width.	
Method	public boolean drawEllipse(double x, double y, double width, double height, double lineWidth);	
Parameter Name	x	The horizontal position of the upper left corner of the drawn ellipse (unit mm).

	y	The vertical position of the upper left corner of the drawn ellipse (unit mm).
	width	The horizontal width of the drawn ellipse (unit mm).
	height	The vertical width of the drawn ellipse (unit: mm). if the default is the same as width, it means to draw a circle.
	lineWidth	The line width of the ellipse(unit: mm). The line width of the ellipse extends towards the inside of itself.
Return Value		Success or failure
Function Introduction		● If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw a filled rounded rectangle. ● The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page. ● When the width and height are equal, a circle will be printed.

11.6 fillEllipse

Function Introduction	Draw a filled ellipse with the specified line width.	
Method	public boolean fillEllipse(double x, double y, double width, double height);	
Parameter Name	x	The horizontal position of the upper left corner of the drawn ellipse (unit mm).
	y	The vertical position of the upper left corner of the drawn ellipse (unit mm).
	width	The horizontal width of the drawn ellipse (unit mm).
	height	The vertical width of the drawn ellipse (unit: mm). if the default is the same as width, it means to draw a circle.
Return Value		Success or failure
Function Introduction		● If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw a filled rounded rectangle. ● The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page. ● When the width and height are equal, a circle will be printed.

11.7 drawCircle

Function Introduction	Draw a circle centered at the specified position
Method	public boolean drawCircle(double x, double y, double radius, double lineWidth);

Parameter Name	x	The x-axis center position of the drawn circle (unit:mm).
	y	The y-axis center position of the drawn circle (unit:mm).
	radius	Draw the radius of the circle (unit: mm).
	lineWidth	The line width of the circle (unit:mm). The line width of the circle extends towards the inside of the circle.
Return Value	Success or failure	
Function Introduction	● If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw a circle. ● The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page.	

11.8 fillCircle

Function Introduction	Draws a filled circle centered at the specified position	
Method	public boolean fillCircle(double x, double y, double radius);	
Parameter Name	x	The x-axis center position of the drawn circle (unit:mm).
	y	The y-axis center position of the drawn circle (unit:mm).
	radius	Draw the radius of the circle (unit: mm).
Return Value	Success or failure	
Function Introduction	● If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw a filled circle. ● The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page.	

11.9 drawLine

Function Introduction	Draw a line (straight line/slash line)	
Method	public boolean drawLine(double x1, double y1, double x2, double y2, double lineWidth);	
Parameter Name	x1	The horizontal position of the starting point of the line (unit:mm)
	y1	The vertical position of the starting point of the line (unit:mm).

	x2	The horizontal position of the end of the line (unit: mm).
	y2	The vertical position of the end of the line (unit: mm).
	lineWidth	Line width(unit: mm). The line width extends downward the line.
Return Value	Success or failure	
Function Introduction	- If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw a straight line/slash line. - The vertical position of the drawing is based on the vertical direction of the current drawing page.	

11.10 drawDashLine

Function Introduction	Draw a dot-dash line	
Method	<pre>public boolean drawDashLine(double x1, double y1, double x2, double y2, double lineWidth, double[] dashLen, int dashCount); public boolean drawDashLine2(double x1, double y1, double x2, double y2, double lineWidth, double dashLen1, double dashLen2); public boolean drawDashLine4(double x1, double y1, double x2, double y2, double lineWidth, double dashLen1, double dashLen2, double dashLen3, double dashLen4);</pre>	
Parameter Name	x1	The horizontal position of the starting point of the line (unit: mm)
	y1	The vertical position of the starting point of the line (unit: mm).
	x2	The horizontal position of the end of the line (unit: mm).
	y2	The vertical position of the end of the line (unit: mm).
	lineWidth	Line width(unit: mm). The line width extends downward the line.
	dashLen1	The length of the first segment of the dot-dash line (unit: mm)
	dashLen2	The length of the second segment of the dot-dash line (unit: mm)
	dashLen3	The length of the third segment of the dot-dash line (unit: mm)
	dashLen4	The length of the fourth segment of the dot-dash line (unit: mm)
	dashLen	Array of line segments length of dot-dash line (unit: mm)
	dashCount	The number of elements in the dot-dash line segment length array

Return Value	Success or failure
Function Introduction	- If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw a dot-dash line. - The vertical position of the drawing is based on the vertical direction of the current drawing page.

12. Description of drawing image method

12.1 drawImage

Function Introduction	Draw an image of the specified file	
Method	public boolean drawImage(String imageFile, double x, double y, double width, double height);	
Parameter Name	imageFile	Image path
	x	The horizontal position of the printed object (unit: mm).
	y	The vertical position of the printed object (unit: mm).
	width	The width of the printed object (unit: mm). if width is 0, use the width of the loaded bitmap. The default is 0.
	height	The height of the printed object (unit: mm). if height is 0, use the width of the loaded bitmap. The default is 0.
Return Value	Success or failure	
Function Introduction	- If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw an image. - The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page.	

12.2 drawImageWithActualSize

Function Introduction	Draw image with actual size	
Method	public boolean drawImageWithActualSize(String imageFile, double x, double y);	
Parameter Name	imageFile	Image path
	x	The horizontal position of the printed object (unit: mm).
	y	The vertical position of the printed object (unit: mm).

Return Value	Success or failure
Function Introduction	- If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw an image. - The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page.

12.3 drawImageWithThreshold

Function Introduction	Draw an image with gray-scale threshold	
Method	public boolean drawImageWithThreshold(String imageFile, double x, double y, double width, double height, int threshold);	
Parameter Name	imageFile	Image path
	x	The horizontal position of the printed object (unit: mm).
	y	The vertical position of the printed object (unit: mm).
	width	The width of the printed object (unit: mm). if width is 0, use the width of the loaded bitmap. The default is 0.
	height	The height of the printed object (unit: mm). if height is 0, use the width of the loaded bitmap. The default is 0.
	threshold	Draw bitmap with the gray-scale threshold. 256 indicate to draw gray-scale image; 257 indicate to draw original color image; 0~255 indicate to draw monochrome image, points with original image color > gray threshold will be considered as white, <= will be considered as black, the default is 192.
Return Value	Success or failure	
Function Introduction	- If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw an image. - The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page.	

12.4 drawBitmap

Function Introduction	Draw a bitmap
Method	public boolean drawBitmap(Bitmap bitmap, double x, double y, double width, double height);

Parameter Name	bitmap	Bitmap object to be drawn.
	x	The horizontal position of the printed object (unit: mm).
	y	The vertical position of the printed object (unit: mm).
	width	The width of the printed object (unit: mm). if width is 0, use the width of the loaded bitmap. The default is 0.
	height	The height of the printed object (unit: mm). if height is 0, use the width of the loaded bitmap. The default is 0.
Return Value	Success or failure	
Function Introduction	● If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw an image. ● The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page. ● Be scaled to the specified width and height while drawing bitmap.	

12.5 drawBitmapWithActualSize

Function Introduction	Draw bitmap with actual size	
Method	public boolean drawBitmapWithActualSize(Bitmap bitmap, double x, double y);	
Parameter Name	bitmap	Bitmap object to be drawn
	x	the upper left corner horizontal position of the drawn bitmap(unit mm).
	y	the upper left corner vertical position of the drawn bitmap(unit mm).
Return Value	Success or failure	
Function Introduction	● If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, and then draw the bitmap. ● The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page. ● Be scaled to the specified width and height while drawing bitmap.	

12.6 drawBitmapWithThreshold

Function Introduction	Draw bitmap with gray-scale threshold	
Method	public boolean drawBitmapWithThreshold(Bitmap bitmap, double x, double y, double width, double height, int threshold);	

Parameter Name	bitmap	Bitmap object to be drawn.
	x	the upper left corner horizontal position of the drawn bitmap(unit mm).
	y	the upper left corner vertical position of the drawn bitmap(unit mm).
	width	The width of the printed object (unit: mm). if width is 0, use the width of the loaded bitmap. The default is 0.
	height	The height of the printed object (unit: mm). if height is 0, use the width of the loaded bitmap. The default is 0.
	threshold	Draw bitmap with the gray-scale threshold. 256 indicate to draw gray-scale image; 257 indicate to draw original color image; 0~255 indicate to draw monochrome image, points with original image color > gray threshold will be considered as white, <= will be considered as black, the default is 192.
Return Value	Success or failure	
Function Introduction	● If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw a bitmap. ● The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page. ● Be scaled to the specified width and height while drawing bitmap.	

12.7 drawBitmapStream

Function Introduction	Draw the bitmap object corresponding to the input stream.	
Method	public boolean drawBitmapStream(InputStream is, double x, double y, double width, double height);	
Parameter Name	is	The input stream object corresponding to the bitmap object to be drawn.
	x	the upper left corner horizontal position of the drawn bitmap(unit mm).
	y	the upper left corner vertical position of the drawn bitmap(unit mm).
	width	The width of the printed object (unit: mm). if width is 0, use the width of the loaded bitmap. The default is 0.
	height	The height of the printed object (unit: mm). if height is 0, use the width of the loaded bitmap. The default is 0.
Return Value	Success or failure	
Function Introduction	● If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw a bitmap.	

- The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page.
- Be scaled to the specified width and height while drawing bitmap.

12.8 drawBitmapStreamWithActualSize

Function Introduction	Draw the bitmap object corresponding to the input stream.	
Method	public boolean drawBitmapStreamWithActualSize(InputStream is, double x, double y);	
Parameter Name	is	The input stream object corresponding to the bitmap object to be drawn.
	x	the upper left corner horizontal position of the drawn bitmap(unit mm).
	y	the upper left corner vertical position of the drawn bitmap(unit mm).
Return Value	Success or failure	
Function Introduction	• If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw a bitmap. • The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page. • Be scaled to the specified width and height while drawing bitmap.	

12.9 drawBitmapStreamWithThreshold

Function Introduction	Draw the bitmap object corresponding to the input stream.	
Method	public boolean drawBitmapStreamWithThreshold(InputStream is, double x, double y, double width, double height, int threshold);	
Parameter Name	is	The input stream object corresponding to the bitmap object to be drawn.
	x	the upper left corner horizontal position of the drawn bitmap(unit mm).
	y	the upper left corner vertical position of the drawn bitmap(unit mm).
	width	The width of the printed object (unit: mm). if width is 0, use the width of the loaded bitmap. The default is 0.
	height	The height of the printed object (unit: mm). if height is 0, use the width of the loaded bitmap. The default is 0.
	threshold	Draw bitmap with the gray-scale threshold. 256 indicate to draw gray-scale image; 257 indicate to draw original color image; 0~255 indicate to draw monochrome image,

		points with original image color > gray threshold will be considered as white, <= will be considered as black, the default is 192.
Return Value	Success or failure	
Function Introduction	● If invoke this method directly without invoking startPage before, this method will invoke startPage automatically to start drawing a page, then draw a bitmap. ● The upper left corner vertical position, width, and height of the drawing are based on the vertical directions of the current drawing page. ● Be scaled to the specified width and height while drawing bitmap.	