Print development guide

Document revision record

Version number	*Change state	Brief description	date	Changed by	Approval date	Approver
V1.0	C	Initial version	2021/8/9	LinJinXing		

*Change status: C = created, a = added, M = modified, d = deleted

Document approval record

Serial number	Approver	role	Approval date	sign	remarks

catalogue

1 Summary

In order to facilitate secondary development, we provide a function library that can run on the Java platform. The library is written in Java language.

2 Introduction method

1. Integration adds the AAR package to the project.
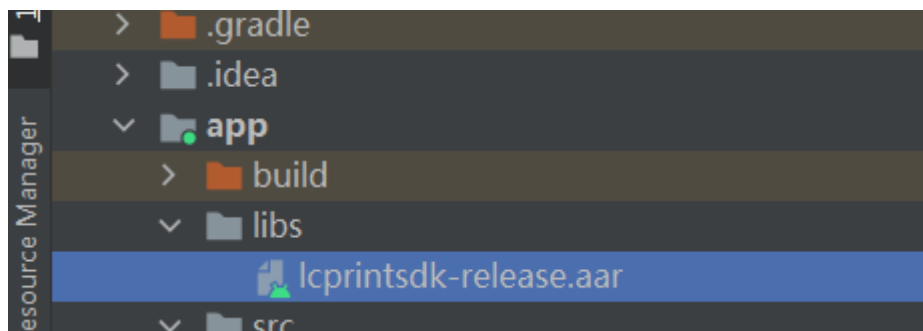
2.Add references to repositories and arr packages under gradle in the project

(repositories and Android nodes are at the same level).
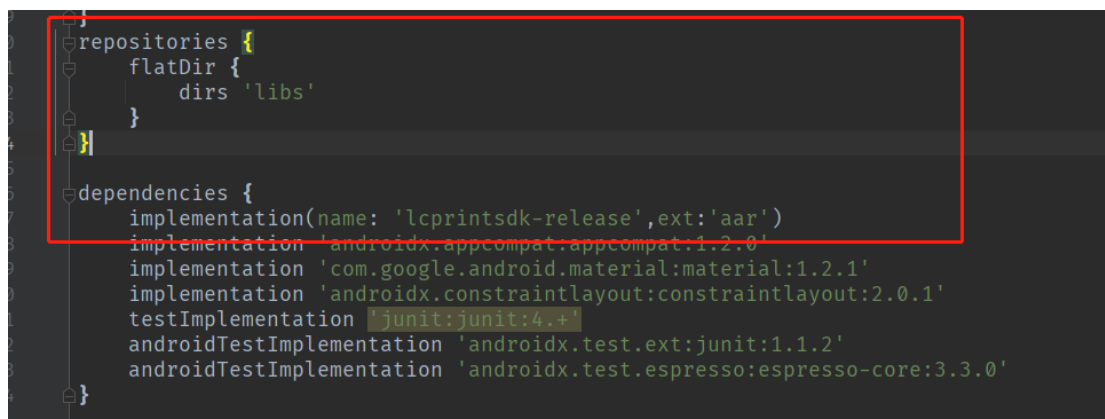
3 Basic printing process
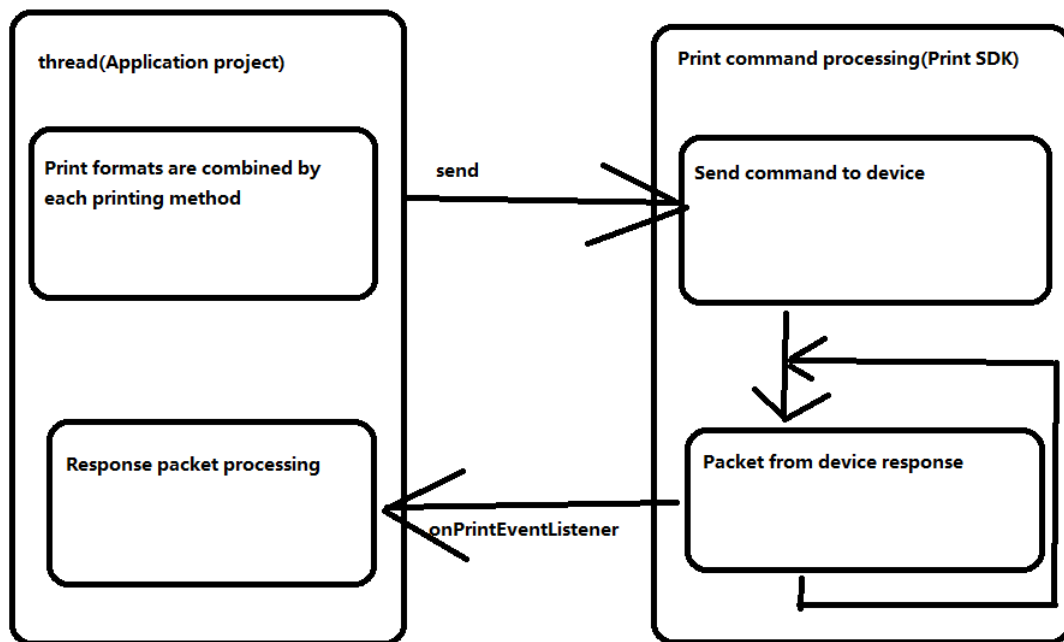
4 Get started quickly

4.1 Label printing

// Step 1: get a print example

printUtil=PrintUtil.getInstance (TestLabelTemplateActivity.this);

// Step 2: set print result callback listening

printUtil.setPrintEventListener (this);// Set listening

// Step 3: label settings

printUtil.setUnwindPaperLen (return_distance);// Set paper return distance

printUtil.printEnableMark (true);// Open Black Label

printUtil.printConcentration (25);// Set concentration

// Step 4: print text

printUtil.printText（"Print text 1"）;

printUtil.printText（"Print text 2"）;

printUtil.printText（"Print text 3"）;

// Step 5: paper feeding

printUtil.printGoToNextMark ();//Paper feeding

4.2 Small ticket printing

// Step 1: get a print example

printUtil=PrintUtil.getInstance (this);// Get instance

// Step 2: set print result callback listening

printUtil.setPrintEventListener (this);// Set listening

// Step 3: label settings

printUtil.printEnableMark (false);// Close Black Label

printUtil.printConcentration (25);// Print density

// Step 4: print text

printUtil.printText ("text ");// Print text

printUtil.printLine (4);// Print lines

// Step 5: paper feeding

printUtil.start ();//Start moving paper

4.3 Print listening example

public class NoLabelTemplateActivity extends BaseActivity implements PrintUtil.PrinterBinderListener {

protected void onCreate(Bundle savedInstanceState) {

// Set print listening

printUtil.setPrintEventListener (this);

}

@Override

public void onPrintCallback(int state) {

if (PrintConfig.IErrorCode.ERROR_NO_ERROR == state) {

```
// Printing succeeded

showToast (getString (R.string.toast_print_success));

}

}

@Override

public void onVersion(String s) {

// Version number

showToast (s);

}

}
```

See demo for details

5 Interface description

5.1 Get printer instance

Function interface	**PrintUtil** getInstance(Context context)
Function description	Gets the print interface instantiation object.
Parameter description	context
Return value	Printutil object

5.2 Add print monitor

Function interface	**void** setPrintEventListener(PrintUtil.PrinterBinderListener Listener)
Function description	Add print monitor
Parameter	Print listener,

description	Onprintcallback (int state) needs to be implemented, Listening status callback state: see Chapter 6.1 status description table for details. Onversion (String Version) needs to be implemented Version: version number Example: see 4.3 print listening example for details.
Return value	nothing

5.3 Remove print monitor

Function interface	**void** removePrintListener(PrintUtil.PrinterBinderListener aListener)
Function description	Remove print monitor
Parameter description	PrintUtil.PrinterBinderListener monitor
Return value	nothing

5.4 Get print version

Function interface	**String** getVersion()
Function description	Get print version
Parameter description	nothing
Return value	Version number

5.5 Open print

Function interface	**boolean** open()
Function	Open print

description	
Parameter description	nothing
Return value	true success : false fail

5.6 Turn off printing

Function interface	**boolean** close()
Function description	Turn off printing
Parameter description	nothing
Return value	true success : false fail

5.7 Set concentration

Function interface	**void** printConcentration(int density)
Function description	Set concentration
Parameter description	Concentration value【1~39】
Return value	nothing

5.8 Reset printer

Function interface	**int** resetPrint()
Function description	Reset printer
Parameter description	nothing

Return value	int

5.9 Set font size

Function interface	**void** printFontSize(int fontSize)
Function description	Set font size
Parameter description	int(refer to chapter 6.1, constant class for detailed values)
Return value	nothing

5.10 Get font size

Function interface	**int** getFontSize()
Function description	Get font size
Parameter description	nothing
Return value	int

5.11 Set bold

Function interface	**void** printTextBold(boolean fontBold)
Function description	Set bold
Parameter description	boolean
Return value	nothing

5.12 Bold

Function interface	**boolean** isFontBold()
Function description	Bold
Parameter description	nothing
Return value	boolean

5.13 Set black mark detection

Function interface	**void** printEnableMark(boolean enable)
Function description	Set Black Label
Parameter description	boolean true: open、false : close
Return value	nothing

5.14 Detect black mark status

Function interface	**boolean** isBlackLabel()
Function description	Detect black mark status
Parameter description	nothing
Return value	boolean

5.15 Set underline

Function interface	**void** setUnderLine(boolean enable)
Function description	Set underline
Parameter description	boolean true: open、false : close
Return value	nothing

5.16 Detect underline

Function interface	**boolean** isUnderLine()
Function description	Detect underline
Parameter description	nothing
Return value	boolean

5.17 Set paper feed distance

Function interface	**void** setFeedPaperSpace(int space)
Function description	Set paper feed distance
Parameter description	Int: distance (effective in label mode, the maximum value is 20001000, about 12.5cm)
Return value	nothing

5.18 Get paper distance

Function interface	**int** getFeedPaperSpace()
Function description	Get paper distance
Parameter description	nothing
Return value	int

5.19 Set paper return distance

Function interface	**void** setUnwindPaperLen(int length)
Function description	Set the paper return distance (black label is required for label printing to have actual effect)
Parameter description	int
Return value	nothing

5.20 Get paper return distance

Function interface	**int** getUnwindPaperLen()
Function description	Get paper return distance
Parameter description	nothing
Return value	int

5.21 Add printed text

Function interface	**void** printText(int offset, int fontSize, boolean isBold, boolean isUnderLine, String content)
Function description	Add printed text
Parameter description	offset： PrintConfig.Align.ALIGN_LEFT be at the left side 　　　　PrintConfig.Align.ALIGN_CENTER Center 　　　　PrintConfig.Align.ALIGN_RIGHT be at the right fontSize：PrintConfig.FontSize font size isBold：Bold isUnderLine：Underline Content：Text content
Return value	nothing

5.22 Add printed text

Function interface	**void** printText(String content)
Function description	Add printed text
Parameter description	Content：Text content
Return value	nothing

5.23 Add 1D code

Function interface	**void** printBarcode(int offset, int height, String content, int barcodeType, int hriPosition)
Function description	Add 1D code
Parameter description	offset：PrintConfig.Align.ALIGN_LEFT be at the left side 　　　　PrintConfig.Align.ALIGN_CENTER Center

	PrintConfig.Align.ALIGN_RIGHT be at the right
	height : One dimensional code height
	content : One dimensional code content
	barcodeType : One dimensional code type
	hriPosition : One dimensional code text content location
	PrintConfig.HRIPosition
Return value	nothing

5.24 Add 1D code

Function interface	**void** printBarcode(int height, String content, int barcodeType)
Function description	Add 1D code
Parameter description	height : One dimensional code height
	content : One dimensional code content
	barcodeType : One dimensional code type
Return value	nothing

5.25 Add QR code

Function interface	**void** printQR(int offset, int height, String content)
Function description	Add QR code
Parameter description	offset : PrintConfig.Align.ALIGN_LEFT be at the left side
	PrintConfig.Align.ALIGN_CENTER Center
	PrintConfig.Align.ALIGN_RIGHT be at the right
	height : QR code height
	content : QR code content
Return value	nothing

5.26 Add QR code

Function interface	**void** printQR(String content)
Function description	Add QR code
Parameter description	content：QR code content
Return value	nothing

5.27 Add picture

Function interface	**void** printBitmap(int offset, Bitmap image)
Function description	Add picture
Parameter description	offset：PrintConfig.Align.ALIGN_LEFT be at the left side 　　　　　PrintConfig.Align.ALIGN_CENTER Center 　　　　　PrintConfig.Align.ALIGN_RIGHT be at the right image：Bitmap
Return value	nothing

5.28 Add picture

Function interface	**void** printBitmap(Bitmap image)
Function description	Add picture
Parameter description	image：Bitmap
Return value	nothing

5.29 Add picture

Function interface	**void** printBitmap(int offset, String imagePath)
Function description	Add picture
Parameter description	offset : PrintConfig.Align.ALIGN_LEFT be at the left side 　　　　PrintConfig.Align.ALIGN_CENTER　Center 　　　　PrintConfig.Align.ALIGN_RIGHT　be at the right imagePath : Picture path
Return value	nothing

5.30 Add picture

Function interface	**void** printBitmap(String imagePath)
Function description	Add picture
Parameter description	imagePath : Picture path
Return value	nothing

5.31 Add blank line

Function interface	**void** printLine(int lines)
Function description	Add blank line
Parameter description	lines : Number of rows
Return value	nothing

5.32 Start moving paper

Function interface	**void** start()
Function description	Start moving paper (Thermal use)
Parameter description	nothing
Return value	nothing

5.33 Set black and white reversal

Function interface	**void** setReverse(boolean reverse)
Function description	Set black and white reversal
Parameter description	Reverse ：true、false
Return value	nothing

5.34 Whether to reverse black and white

Function interface	**boolean** isReverse()
Function description	Whether to reverse black and white
Parameter description	nothing
Return value	Boolean

5.35 Start moving paper

Function interface	**void** printGoToNextMark()
Function description	Start moving paper（Label use）
Parameter description	nothing
Return value	nothing

5.36 Start moving paper

Function interface	**void** printGoToNextMark(int distance)
Function description	Start moving paper（Label use）
Parameter description	Distance: Paper feeding distance
Return value	nothing

5.37 Set text line spacing

Function interface	**void** setLineSpacing(float spacing)
Function description	Set text line spacing
Parameter description	spacing: Unit multiple example：spacing=1　　1 line spacing 　　　spacing=3　　　3x line spacing
Return value	nothing

5.38 Get text line spacing

Function interface	**float** getLineSpacing()
Function description	Get text line spacing
Parameter description	nothing
Return value	float Row spacing multiple

5.38 Get current support module

Function interface	**int** getsupportprint()
Function description	Get current support module
Parameter description	nothing
Return value	int 1: New module, 0: old module

6 appendix

6.1 Constant class

Constant class：android.bld.print.configuration.PrintConfig

//IErrorCode：Error code

public class IErrorCode {

      public static final int ERROR_CMD = 19;// Instruction error

      public static final int ERROR_DATA_INPUT = 165;// Input parameter error

      public static final int ERROR_DATA_INVALID = 18;// data is invalid

```java
        public static final int ERROR_DEV_BMARK = 7;// Abnormal black mark
detection

        public static final int ERROR_DEV_FEED = 5;// Moving paper

        public static final int ERROR_DEV_IS_BUSY = 1;// Device busy

        public static final int ERROR_DEV_NOT_OPEN = 16;// The device is not
turned on

        public static final int ERROR_DEV_NO_BATTERY = 4;// Low electricity

        public static final int ERROR_DEV_PRINT = 6;// print now

        public static final int ERROR_GRAY_INVALID = 20;// Illegal concentration

        public static final int ERROR_NO_DATA = 17;// Print data cannot be empty

        public static final int ERROR_NO_ERROR = 0;// Printing succeeded

        public static final int ERROR_PRINT_BARCODE = 162;// Error printing
barcode

        public static final int ERROR_PRINT_BITMAP = 161;// Error printing bitmap

        public static final int ERROR_PRINT_BITMAP_WIDTH_OVERFLOW = 164;//
Print bitmap width overflow

        public static final int ERROR_PRINT_DATA_MAC = 167;// Mac check error
        public static final int ERROR_PRINT_HOT = 2;// high temperature

        public static final int ERROR_PRINT_ILLEGAL_ARGUMENT = 166;//
Parameter error

        public static final int ERROR_PRINT_NOPAPER = 3;// Paper shortage
        public static final int ERROR_PRINT_QRCODE = 163;// Error printing QR code

        public static final int ERROR_PRINT_TEXT = 160;// Print text error

        public static final int ERROR_PRINT_UNKNOWN = 255;// unknown error

        public static final int ERROR_RESULT_EXIST = 168;// Result already exists

        public static final int ERROR_TIME_OUT = 169;// overtime

}
//StateType : Status type

public class StateType {

        public static final int CHECK_ALL = 1;

        public static final int CHECK_BMASK = 7;

        public static final int CHECK_BUSY = 2;

        public static final int CHECK_FEED = 5;
```

```java
        public static final int CHECK_PAPER = 4;

        public static final int CHECK_PRINT = 6;

        public static final int CHECK_TEMP = 3;

}

//HRIPosition : One dimensional code text position

public static class HRIPosition {

        public static final int POSITION_ABOVE = 2;

        public static final int POSITION_BELOW = 3;

        public static final int POSITION_BOTH = 4;

        public static final int POSITION_NONE = 1;

}

//BarCodeType : Barcode Type

public static class BarCodeType {

        public static final int TOP_TYPE_CODABAR = 71;

        public static final int TOP_TYPE_CODE128 = 73;

        public static final int TOP_TYPE_CODE39 = 69;

        public static final int TOP_TYPE_CODE93 = 72;

        public static final int TOP_TYPE_EAN13 = 67;

        public static final int TOP_TYPE_EAN8 = 68;

        public static final int TOP_TYPE_ITF = 70;

        public static final int TOP_TYPE_UPCA = 65;

        public static final int TOP_TYPE_UPCE = 66;


}

//ALIGN : Alignment

public static class Align {

        public static final int ALIGN_CENTER = 2;// Center

        public static final int ALIGN_LEFT = 1;// be at the left side

        public static final int ALIGN_RIGHT = 3;// be at the right
```

```
}
//FontSize：Character font
public static class FontSize {
        public static final int TOP_FONT_SIZE_LARGE = 5;
        public static final int TOP_FONT_SIZE_MIDDLE = 3;
        public static final int TOP_FONT_SIZE_SMALL = 1;
        public static final int TOP_FONT_SIZE_SUPER = 7;
        public static final int TOP_FONT_SIZE_XLARGE = 6;
        public static final int TOP_FONT_SIZE_XMIDDLE = 4;
        public static final int TOP_FONT_SIZE_XSMALL = 2;
        public static final int TOP_FONT_SIZE_XSUPER = 8;

}
```

7 Q&A

3.1 Q: If you already have an old SDK, how can you upgrade it

A: Obtain the difference interface between the old and new modules, and add judgment compatibility processing. (distinguish between old and new modules, refer to chapter 5.38)

3.2 Q: What if the old and new dependency packages conflict

A: Need to cancel core_ 3.4.0.jar package dependency.

3.3 Q: How to migrate an old SDK to a new SDK

A: You can refer to the S60 and s60a print SDK interface differences. Docx document.Let's first understand the differences between the SDK. The printing process between the old and new printing is basically not different. There are changes in individual methods. It's OK to replace them.