

Flutter POS Program Manual

V1.2.0

1. Instruction

By reading this manual, developers can quickly learn how to use Flutter to implement the receipt printing function and apply it to actual development. This manual includes the use of the `PrinterManager` class and the `POSCCommand` class, as well as the meaning and usage of the constants in the `POSConst` class.

2. PrinterManager

2.1. PrinterManager

Constructor, create a printer management object

`PrinterManager()`

2.2. connectBt

This method is used for Bluetooth connection (Android uses classic Bluetooth, iOS uses BLE)

`Future<void> connectBt(String address, Function(int result) callback)`

[Parameter]

➤ address

Bluetooth Address

➤ callback

Connection status callback

2.3. connectUsb

This method is used for USB connection (only supports Android)

`Future<void> connectUsb(String path, Function(int result) callback)`

[Parameter]

➤ path

usb path

➤ callback

Connection status callback

2.4. getUsbPaths

This method is used to obtain the currently connected USB device list (only supports android)

`Future<List> getUsbPaths()`

[Return]

Usb List object: usb address list collection

2.5.searchCallback

Bluetooth search callback

searchCallback(Map args)

[Return]

Map dictionary object of Bluetooth information: key: address, value: name

2.6.sendESC

This method is used to send pos instructions

Future<void> sendESC(List<Map<String, dynamic>> data)

2.7.checkPermissions

This method is used to check Android Bluetooth authorization

Future<bool> checkPermissions()

[Return]

Bool object

2.8.startScan

This method is used for Bluetooth search (Android: Classic Bluetooth search, iOS: BLE search)

Future<void> startScan()

2.9.disconnect

Disconnect

Future<void> disconnect()

2.10.posprinterStatus

Query connection status

Future<int> posPrinterStatus()

status	Description
STS_CONNECT	connect
STS_DISCONNECT	disconnect

2.11.checkIsConnect

Query connection status

Future<int> checkIsConnect()
[Return]

status	Description
STS_CONNECT	connect
STS_DISCONNECT	disconnect

2.12.GetSerialNumber

Obtain the serial number of the printer
Future<String> getSerialNumber()

[Return]
Get the SN code queried by callback.

2.13.setBluetooth

Set Bluetooth information
void setBluetooth(String name, String pin)

[Parameter]
➤ name
bluetooth name
➤ pin
bluetooth pin code

[Return]
void

2.14.cashBoxCheck

This method is used to query the cash drawer status.

Future<int> cashBoxCheck()
The status-values are shown in the table below.

STS_UNKNOWN	Unknown state, read data timeout or received data length is not 1.
STS_CASH_OPEN	Cash drawer is open.
STS_CASH_CLOSE	Cash drawer is close.

3.POSCommand

3.1.POSCommand

Constructor to create print objects.
POSCommand()

3.2.printString

This function is used for text-printing.

POSCommand printString(String data)

[Parameter]

➤ data

Printed text string

[Return]

POSCommand Instance

3.3.printText

This function is used for format-specific text printing.

POSCommand printText(String text, {int alignment = POSConst.ALIGNMENT_LEFT, int attribute = POSConst.FNT_DEFAULT, int textSize = POSConst.TXT_1WIDTH | POSConst.TXT_1HEIGHT})

[Parameter]

➤ data

Printed text string

➤ alignment

The alignment of the text, and the default is ALIGNMENT_LEFT

Note: When using alignment, data needs to end with "\n", otherwise it may become invalid.

Variable	Description
ALIGNMENT_LEFT	Align left
ALIGNMENT_CENTER	Align center
ALIGNMENT_RIGHT	Align right

➤ attribute

This value is text attributes. It sets text attributes to print. default is FNT_DEFAULT

Variable	Description
FNT_DEFAULT	FontA, Set up as a standard
FNT_FONTB	Set up as FontB
FNT_BOLD	bold font
FNT_REVERSE	Set up as reverse print attribute
FNT_UNDERLINE	Set up as Underline attribute
FNT_UNDERLINE2	Set up as Bold Underline attribute

➤ textSize

The font size of the printed text font,default is TXT_1WIDTH|TXT_1HEIGHT

Variable(Set up width ratio)	Description
TXT_1WIDTH	Set up width ratio as x1

TXT_2WIDTH	Set up width ratio as x2
TXT_3WIDTH	Set up width ratio as x3
TXT_4WIDTH	Set up width ratio as x4
TXT_5WIDTH	Set up width ratio as x5
TXT_6WIDTH	Set up width ratio as x6
TXT_7WIDTH	Set up width ratio as x7
TXT_8WIDTH	Set up width ratio as x8

Variable(Set up height ratio)	Description
TXT_1HEIGHT	Set up height ratio as x1
TXT_2HEIGHT	Set up height ratio as x2
TXT_3HEIGHT	Set up height ratio as x3
TXT_4HEIGHT	Set up height ratio as x4
TXT_5HEIGHT	Set up height ratio as x5
TXT_6HEIGHT	Set up height ratio as x6
TXT_7HEIGHT	Set up height ratio as x7
TXT_8HEIGHT	Set up height ratio as x8

[Return]

POSCOMMAND Instance

3.4.printBitmap

This function is used for printing image files.

POSCOMMAND printBitmap(uint8List bitmap,int width, {int alignment = POSConst.ALIGNMENT_LEFT,int mode = POSConst.BMP_NORMAL})

[Parameter]

➤bitmap

Bitmap Object.

➤alignment

The alignment mode of the pictures.

Variable	Description
ALIGNMENT_LEFT	Align left
ALIGNMENT_CENTER	Align center
ALIGNMENT_RIGHT	Align right

➤width

Print the picture width.

➤model Print

mode

Variable	Description
BMP_NORMAL	Original(Normal) size
BMP_WIDTH_DOUBLE	Double width

BMP_HEIGHT_DOUBLE	Double height
BMP_WIDTH_HEIGHT_DOUBLE	Double size

[Return]

POSCommand Instance

3.5.printBarCode

This function is used for supporting barcode printing.

POSCommand printBarCode(String data, int codeType, {int width = 1, int height = 1, int alignment = POSConst.ALIGNMENT_LEFT, int textPosition = POSConst.HRI_TEXT_BELOW})

[Parameter]

➤data

Barcode string to be printed

➤codeType

Barcode type

Variable	Description
BCS_UPCA	UPC A
BCS_UPCE	UPCE
BCS_EAN8	EAN-8
BCS_EAN13	EAN-13
BCS_JAN8	JAN-8
BCS_JAN13	JAN-13
BCS_ITF	ITF
BCS_Codabar	Codabar
BCS_Code39	Code 39
BCS_Code93	Code 93
BCS_Code128	Code 128, For this type, the data must be added with {A, {B, {C, etc

➤height

Barcode height, range [1,255].Default is 162

➤width

This values barcode width in Dot Units, range [2, 6], Default is 3

➤alignment

It sets barcode alignment, Default is ALIGNMENT_CENTER

Variable	Description
ALIGNMENT_LEFT	Align left
ALIGNMENT_CENTER	Align center
ALIGNMENT_RIGHT	Align right

➤textPosition

This value is printing position of barcode HRI letters(barcode data).Default is HRI_TEXT_BELOW.

Variable	Description
HRI_TEXT_NONE	Do not print barcode data
HRI_TEXT_ABOVE	Print barcode data above the barcode
HRI_TEXT_BELOW	Print barcode data below the barcode
HRI_TEXT_BOTH	Print barcode data top and bottom

[Return]

POSCommand Instance

3.6.feedLine

This function is used for sending feeding command to printer.

POSCommand feedLine({int lineCount = 1})

[Parameter]

➤lineCount

This value is the number of lines for line feeding. Default is 1

[Return]

POSCommand Instance

3.7.printQRCode

This function is used for supporting QRCode barcode printing.

POSCommand printQRCode(String content,{int moduleSize = 8, int ecLevel = POSConst.QRCODE_EC_LEVEL_L, int alignment = POSConst.ALIGNMENT_CENTER})

[Parameter]

➤data

QRCode data to print

➤moduleSize

Module size. Range[1, 16], Default is 8.

➤ecLevel

Error Correction Level, Default is QRCODE_EC_LEVEL_L

Variable	Description
QRCODE_EC_LEVEL_L	Error correction Level L (7%)
QRCODE_EC_LEVEL_M	Error correction Level M (15%)
QRCODE_EC_LEVEL_Q	Error correction Level Q (25%)
QRCODE_EC_LEVEL_H	Error correction Level H (30%)

➤alignment

It sets QRCode alignment, Default is ALIGNMENT_CENTER

Variable	Description
ALIGNMENT_LEFT	Align left

ALIGNMENT_CENTER	Align center
ALIGNMENT_RIGHT	Align right

[Return]

POSCommand Instance

3.8.cutPaper

This method is used for cutting the paper

POSCommand cutPaper({int model = POSConst.CUT_HALF})

POSCommand cutHalfAndFeed(int distance)

[Parameter]

➤model

Cut paper mode, Default is CUT_HALF.

Variable	Description
CUT_ALL	Full cut
CUT_HALF	Half cut

➤distance

Feed distance

[Return]

POSCommand Instance

3.9.openCashBox

Open a cash drawer.

POSCommand openCashBox(int pinNum, {int onTime = 30, int offTime = 255})

[Parameter]

➤pinNum

Pin number to generate pulse.

Variable	Description
PIN_TWO	PIN 2
PIN_FIVE	PIN 5

➤onTime

Start tiime to generate pulse. onTime*2ms, range [0,255], Default is 30

➤offTime

Stop time to generate pulse. offTime*2ms, range [0,255], Default is 255

[Return]

POSCommand Instance

3.10.setCharSet

Set character encoding

POSCCommand setCharSet(String charSet)

[Parameter]

➤charSet

Character set name.

3.11.setAlignment

This method is used for set up the alignment of the text

POSCCommand setAlignment(int alignment)

[Parameter]

➤alignment

The alignment of the text, and the default is ALIGNMENT_LEFT

Variable	Description
ALIGNMENT_LEFT	Align left
ALIGNMENT_CENTER	Align center
ALIGNMENT_RIGHT	Align right

[Return]

POSCCommand Instance

3.12.setPrintArea

Set up the print area in page mode.

POSCCommand setPrintArea(int width, int height, {int x = 0, int y = 0})

[Parameter]

➤x

The x-coordinate of the starting position,Default is 0.

➤y

The y-coordinate of the starting position,Default is 0.

➤width

Width of printing area.

➤height

Height of printing area.

[Return]

POSCCommand Instance

3.13.setPageModel

Change to page mode or standard mode.

POSCommand setPageModel(bool isOpen)

[Parameter]

➤isOpen

Enable or Disable page mode. (TRUE, FALSE)

[Return]

POSCommand Instance

3.14.printPageModelData

Print and return to standard mode in page mode.

POSCommand printPageModelData()

[Return]

POSCommand Instance

3.15.setPrintDirection

Select print direction in page mode.

POSCommand setPrintDirection(int direction)

[Parameter]

➤direction

Print direction

Variable	Description
DIRECTION_LEFT_TOP	From top left to right
DIRECTION_LEFT_BOTTOM	From bottom left to top
DIRECTION_RIGHT_BOTTOM	From bottom right to top
DIRECTION_RIGHT_TOP	From top right to bottom

[Return]

POSCommand Instance

3.16.setAbsoluteHorizontal

Set absolute horizontal print position . (X axis)

POSCommand setAbsoluteHorizontal(int position)

[Parameter]

➤ position

Starting position.

[Return]

POSCommand Instance

3.17.setRelativeHorizontal

Set relative horizontal print position. (X axis)

POSCommand setRelativeHorizontal(int position)

[Parameter]

➤ position

Starting position.

[Return]

POSCommand Instance

3.18.setAbsoluteVertical

Set absolute vertical print position in page mode. (Y axis)

POSCommand setAbsoluteVertical(int position)

[Parameter]

➤ position

Starting position.

[Return]

POSCommand Instance

3.19.setRelativeVertical

Set relative vertical print position in page mode. (Y axis)

POSCommand setRelativeVertical(int position)

[Parameter]

➤ position

Starting position.

[Return]

POSCommand Instance

3.20.downloadNVImage

This function is used for save the NV images in flash.

POSCCommand downloadNVImages(List<Uint8List> bitmaps, int imageWidth)

[Parameter]

➤ bitmaps

The bitmap list

➤ imageWidth

This value is image width.

[Return]

POSCCommand Instance

3.21.printNVImage

This function is used to support the Bitmap Image printing stored in Flash Memory.

POSCCommand printNVImage(int index, int model)

[Parameter]

➤ index

It sets the index image stored in Flash Memory to print,range[1,255]

➤ model Print

model

Variable	Description
BMP_NORMAL	Normal size
BMP_WIDTH_DOUBLE	Double width
BMP_HEIGHT_DOUBLE	Double height
BMP_WIDTH_HEIGHT_DOUBLE	Double size

[Return]

POSCCommand Instance

3.22.initializePrinter

Initialize Printer, This function clears the print buffer data.

POSCCommand initializePrinter()

[Return]

POSCCommand Instance

3.23.selectBitmapModel

Select bitmap model

POSCCommand selectBitmapModel(int model, int width, Uint8List bmp)

[Parameter]

➤model

Bitmap model

➤Bitmap model

Variable	Description
SINGLE_DENSITY_8	8-point single density
DOUBLE_DENSITY_8	8-point double density
SINGLE_DENSITY_24	24-point single density(76 impact printers does not support)
DOUBLE_DENSITY_24	24-point double density(76 impact printers does not support)

➤width

Print the picture width.

➤bmp

Bitmap image

[Return]

POSCCommand Instance

3.24.feedDot

Print buffer data and run n points

POSCCommand feedDot(int n)

[Parameter]

➤n

The paper distance, in horizontal or vertical moving units. The default is point.

[Return]

POSCCommand Instance

3.25.setLineSpacing

Set line-height

POSCCommand setLineSpacing(int space)

➤space

Line-height,If you want to restore to the default height, use SPACE_DEFAULT.

[Return]

POSCCommand Instance

3.26.setTurnUpsideDownMode

Select / cancel the inverted printing mode.

POSCCommand setTurnUpsideDownMode(bool on)

[Parameter]

➤ on

True indicates selection, false indicates cancel.

[Return]

POSCCommand Instance

3.27.selectCodePage

Select character code page

POSCCommand selectCodePage(int page)

[Parameter]

➤ page

Code page

Value	Description	Value	Description
0	PC437(Std.Europe)	56	PC861(Icelandic)
1	Katakana	57	PC863(Canadian)
2	PC850(Multilingual)	58	PC865(Nordic)
3	PC860(Portugal)	59	PC866(Russian)
4	PC863(Canadian)	60	PC855(Bulgarian)
5	PC865(Nordic)	61	PC857(Turkey)
6	West Europe	62	PC862(Hebrew)
7	Greek	63	PC864(Arabic)
8	Hebrew	64	PC737(Greek)
9	East Europe	65	PC851(Greek)
10	Iran	66	PC869(Greek)
16	WPC1252	67	PC928(Greek)
17	PC866(Cyrillic#2)	68	PC772(Lithuanian)
18	PC852(Latin2)	69	PC774(Lithuanian)
19	PC858	70	PC874(Thai)
20	IranII	71	WPC1252(Latin-1)
21	Latvian	72	WPC1250(Latin-2)
22	Arabic	73	WPC1251(Cyrillic)
23	PT1511251	74	PC3840(IBM-Russian)
24	PC747	75	PC3841(Gost)
25	WPC1257	76	PC3843(Polish)

27	Vietnam	77	PC3844(CS2)
28	PC864	78	PC3845(Hungarian)
29	PC1001	79	PC3846(Turkish)
30	Uigur	80	PC3847(Brazil-ABNT)
31	Hebrew	81	PC3848(Brazil)
32	WPC1255(Israel)	82	PC1001(Arabic)
255	Thai	83	PC2001(Lithuan)
33	WPC1256	84	PC3001(Estonian-1)
50	PC437(Std.Europe)	85	PC3002(Eston-2)
51	Katakana	86	PC3011(Latvian-1)
52	PC437(Std.Europe)	87	PC3012(Tatv-2)
53	PC858(Multilingual)	88	PC3021(Bulgarian)
54	PC852(Latin-2)	89	PC3041(Maltese)
55	PC860(Portuguese)		

[Return]

POSCommand Instance

3.28.selectCharacterFont

Select font

POSCommand selectCharacterFont(int font)

[Parameter]

➤font

Font type

Variable	Description
FONT_STANDARD	Standard ascii font (12 × 24)
FONT_COMPRESS	Compress ASCII font (9 × 17)

[Return]

POSCommand Instance

3.29.sendData

This function is used to send data to the printer

POSCommand sendData(UInt8List data)

[Parameter]

➤data

Byte array to be sent

[Return]

POSCCommand Instance