[ESC Linux SDK]

[Printer ESC Command Development Manual v2.0.4]

1. Information of the Manual

This SDK manual provides the so file information for Linux application development.
We continuously promote and update the function and quality of all our products.   Any change to the product specification and the manual will be without any further notice.

2. Operation System

Linux debian 5.10.0 and above.

3. remark

1.   When error code Return Value is greater than 0, it is the internal error of Linux system, please refer to related help file.
2.   The printer resolution is 200 dpi,1 mm=8 dot;The printer resolution is 300 dpi,1 mm=12 dot.
3.   The SDK references third-party libraries: libserialport, libusb-1.0. Please install it in the operating system in advance.
4.   Serial port connection requires root privileges.

4.   Method

4.1.   InitPrinter

Set up the target printer of specified model (the printer object must be created before any printer operation).

```
void* InitPrinter (
      const TCHAR* model
);
```

Parameter：
const TCHAR model*
      [in]   Specify the model of target printer.

Return Value：
      success:Returns a handle to the printer object
      fail:Return NUL

4.2.   ReleasePrinter

The method is to release the resources of the printer object (the created printer object must be released after the operation is completed).

```
int ReleasePrinter    (
      void* hPrinter
);
```

Parameter:
void hPrinter*
    [in]   Handle to the target printer object that needs to be released

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-1	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory

4.3.   OpenPort

Open the communication port and connect with the printer. After successfully   connected, other functions can be used. If failed connecting, please check the error information. Currently it supports USB, internet, serial interface.

int OpenPort (
    void* hPrinter,
    const TCHAR* setting
);

Parameter:
void hPrinter*
    [in]   The created target printer object.
const TCHAR setting*
    [in]   Set the communication port parameters to connect to the target printer. See the table below for details:
    Configuration List:

Type	Configuration	Description	Sample
USB	USB,path	USB,USB path	USB,/001/007
NET	NET, IP address (IPV4)[,port]	Spacify the IPAddress and port.If no port is specified,The default port is 9100.	NET,192.168.1.10 NET,192.168.1.10,9100
COM	COM,path,rate	Specify the connected serial port path and baud rate.	COM10,19200

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_USB_DEVICE_NOT_FOUND	-17	Failed, device not found
ERROR_IO_OPEN_FAILED	-8	Failed to open port

4.4.   ClosePort

This function is to close the communication port and disconnect with the printer.

int ClosePort (
        void* hPrinter
);

Parameter:
void hPrinter*
        [in]   The created target printer object.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-3	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-2	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory

4.5.   WriteData

This function is to send data to the printer.

int WriteData(
        void* hPrinter,
        unsigned char* buffer,
        unsigned int size
);

Parameter:
void hPrinter*
        [in]   The created target printer object.
unsigned char buffer*
        [in]   The data sent to the printer (hex string).
unsigned int size
        [in]   The length of the sent data.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out

4.6.   ReadData

This function is to read the printer data.

```
int ReadData(
      void* hPrinter,
      unsigned char* buffer,
      unsigned int size
);
```

Parameter:
void hPrinter*
      [in]   The created target printer object.
unsigned char buffer*
      [in]   Printer data to be read.
unsigned int size
      [in]   The length of the data to be read.

Return Value:

Code	Value	Description
>0	>0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_OPEN_FAILED	-8	Failed to open port

4.7.   PrinterInitialize

Clear the data in the print buffer and reset the printer modes to the modes that were in effect when power was turned on.
Any macro definitions are not cleared.
Offline response selection is not cleared.
Contents of user NV memory are not cleared.
NV graphics (NV bit image) and NV user memory are not cleared.
The maintenance counter value is not effected by this command.
The specifying offline response isn't cleared.

```
int PrinterInitialize(
      void* hPrinter
);
```

Parameter：
void hPrinter*
      [in] The created target printer object.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.8.   SetTextLineSpace

Set the line spacing to lineSpace × (vertical or horizontal motion unit).
When standard mode is selected, the vertical motion unit is used.
When page mode is selected, the vertical horizontal motion unit is used for the print direction.

int SetTextLineSpace(
        void* hPrinter,
        int lineSpace
);

Parameter：
void hPrinter*
        [in] The created target printer object.
int lineSpace
        [in] Set the line spacing of characters   0≤linespace≤255.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.9.   CancelPrintDataInPageMode

In page mode, delete all the print data in the current print area.

int CancelPrintDataInPageMode(
        void* hPrinter
);

Parameter：
void hPrinter*
        [in] The created target printer object.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.10. GetPrinterState

This function is for getting the printer real-time state.

int GetPrinterState(
      void* hPrinter,
      unsigned int* printerStatus
);

Parameter：
void hPrinter*
      [in] The created target printer object.
*unsigned int * printerStatus*
      [in] Pinter real-time state, when returning to multi-state, value is showed as   accumulation,
return state please refer to below:

printerStatus	bit	function	value	decimal value
1	0	fixed to 0	0	0
	1	fixed to 1	1	2
	2	One or two cash boxes open	0	0
		Both cash boxes are closed	1	4
	3	online	0	0
		Offline	1	8
	4	fixed to 1	1	16
	5,6	fixed to 0	0	0
	7	fixed to 0	0	0
2	0	fixed to 0	0	0
	1	fixed to 1	1	2
	2	The upper cover is closed	0	0
		The upper cover is open	1	4
	3	Paper feed key not pressed	0	0
		Press the paper feed key	1	8
	4	fixed to 1	1	16
	5	Printer has paper	0	0
		Printer is out of paper	1	32
	6	No error conditions	0	0
		error condition	1	64
	7	fixed to 0	0	0
3	0	fixed to 0	0	0
	1	fixed to 1	1	2
	2	undefined	0	0
	3	Cutter without error	0	0
		Cutter error	1	8
	4	fixed to 1	1	16
	5	No unrecoverable errors	0	0
		There is an unrecoverable error	1	32
	6	Print head temperature and voltage are normal	0	0
		Print head temperature or voltage is out of range	1	64
	7	fixed to 0	0	0
4	0	fixed to 0	0	0
	1	fixed to 1	1	2
	2,3	enough paper	0	0
		Paper is about to run out	1	12

	4	fixed to 1	1	16
	5,6	Printer has paper	0	0
		printer is out of paper	1	96
	7	fixed to 0	0	0

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.11. SetCodePage

Select character code table.

int SetCodePage(
      void* hPrinter,
      int characterSet
);

Parameter：
void hPrinter*
      [in] The created target printer object.
int characterSet
      [in] Select code page setting.

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 Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.12. SetInternationalCharacter

Select an international character set.

int SetInternationalCharacter(
        void* hPrinter,
        int characterSet
);

Parameter：
void hPrinter*
        [in] The created target printer object.
int characterSet
        [in] Select international character setting.
        Default: U.S.A.

Value	Description
0	U.S.A
1	France
2	Germany
3	U.K.
4	Denmark I
5	Sweden
6	Italy
7	Spain
8	Japan
9	Norway

10	Denmark II
11	Spain II
12	Latin America
13	Korean
14	Slovenia / Croatia
15	Chinese

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.13. CutPaper/CutPaperWithDistance

Feed paper to (cutting position + distance × vertical motion unit) and execute a full cut (cuts the paper completely) or execute a partial cut (one point left uncut), then feed paper to the print start position.

```
int CutPaper(
      void* hPrinter,
       int cutMode
);
int CutPaperWithDistance(
      void* hPrinter,
       int distance
);
```

Parameter：
void hPrinter*
    [in] The created target printer object.
int cutMode
    [in] Paper cut mode. Execute a full cut or a partial cut.

Paper Cut Mode	Value	Description
FULL_CUT	0	Full cut
PARTIAL_CUT	1	Partial cut

int distance
    [in] Specify a range of paper cut 0≤distance≤255.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out

Other values	Other values	the error code returned by the Linux system

4.14. FeedLine

Print the data in the print buffer and feed lines, when printer in page mode, only the print position moves, and the printer does not perform actual printing.

int FeedLine(
      void* hPrinter,
      int lines
);

Parameter：
void hPrinter*
      [in] The created target printer object.
int lines
      [in] Set lines of paper feed   0≤lines≤255.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.15. PrintAndFeedLine

The function is to print the data in the print buffer and feed one line of paper.

int PrintAndFeedLine(
      void* hPrinter
)

Parameter：
void hPrinter*
      [in] The created target printer object.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.16. OpenCashDrawer

This function is for opening the cash drawer(printer should be connected with cash drawer).

int OpenCashDrawer(
        void* hPrinter,
        int pinMode,
        int onTime,
        int ofTime
);

Parameter：
void hPrinter*
        [in] The created target printer object.
int pinMode
        [in] Select the pin which cash drawer connected.

Pin	Value	Description
CASDRAWER_1	0	Pin 2
CASDRAWER_2	1	Pin 5

int onTime
        [in] Set the start time of pulse, onTime*2ms.
int ofTime
        [in] Set the end time of pulse, offTime*2ms.
        Remark: When the setting value of end time is less than that of start time, end time is equal to the start time.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.17. PrintText

This function is for printing printer text with attribute.

int PrintText(
        void* hPrinter,
        const char* data,
        int alignment,
        int textSize
);

Parameter：
void hPrinter*

[in] The created target printer object.

const char data,*
    [in] The text data which needs to print.

int alignment
    [in] The alignment method of text.

Alignment Method	Value	Description
ALIGNMENT_LEFT	0	Left alignment
ALIGNMENT_CENTER	1	Center alignment
ALIGNMENT_RIGHT	2	Right alignment

int textSize
    [in] Set the text size (It will not be printed if the text length exceeds print paper area).
    Set text width:

Text Width	Value	Description
TEXT_SIZE_0WIDTH	0	Text width × 1
TEXT_SIZE_1WIDTH	16	Text width × 2
TEXT_SIZE_2WIDTH	32	Text width × 3
TEXT_SIZE_3WIDTH	48	Text width × 4
TEXT_SIZE_4WIDTH	64	Text width × 5
TEXT_SIZE_5WIDTH	80	Text width × 6
TEXT_SIZE_6WIDTH	96	Text width × 7
TEXT_SIZE_7WIDTH	112	Text width × 8

    Set text height:

Text Height	Value	Description
TEXT_SIZE_0HEIGHT	0	Text height × 1
TEXT_SIZE_1HEIGHT	1	Text height × 2
TEXT_SIZE_2HEIGHT	2	Text height × 3
TEXT_SIZE_3HEIGHT	3	Text height × 4
TEXT_SIZE_4HEIGHT	4	Text height × 5
TEXT_SIZE_5HEIGHT	5	Text height × 6
TEXT_SIZE_6HEIGHT	6	Text height × 7
TEXT_SIZE_7HEIGHT	7	Text height × 8

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.18. SetRelativeHorizontal

Set relative horizontal printing position.

int SetRelativeHorizontal(
    void* hPrinter,

```
        int position
)
```

Parameter:
void hPrinter*
    [in] The created target printer object.
int position
    [in]   relative position

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.19. PrintTextS

This function is for printing printer text.

```
int PrintTextS(
        void* hPrinter,
        const char* data
);
```

Parameter:
void hPrinter*
    [in] The created target printer object.
const char data,*
    [in] The text data which needs to print.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.20. PrintBarCode

This function is for printing bar code. In standard mode, only when bar code print position is in new line or without data in buffer can print normally. In page mode, when the command of printing bar code is not received, the bar code data will be saved in buffer and print will not be executed.

```
int PrintBarCode(
    void* hPrinter,
    int bcType,
    const char* bcData,
    int width,
    int height,
    int alignment,
    int hriPosition
);
```

Parameter：
void hPrinter*
    [in] The created target printer object.
int bcType
    [in] Set bar code type.
const char bcData,*
    [in] Bar code data.

Bar Code Type	Value	Bar Code Data Length	Effective Data Range
BARCODE_UPC_A	65	11≤n≤12	48≤data≤57
BARCODE_UPC_E	66	11≤n≤12	48≤data≤57
BARCODE_EAN13 BARCODE_JAN13	67	12≤n≤13	48≤data≤57
BARCODE_EAN8 BARCODE_JAN8	68	7≤n≤8	48≤data≤57
BARCODE_CODE39	69	1≤n≤255	48≤data≤57,65≤data≤90, data=32,36,37,43,45,46,47
BARCODE_ITF	70	1≤n≤255 (even number)	48≤data≤57
BARCODE_CODABAR	71	1≤n≤255	48≤data≤57,65≤data≤68, data =36,43,45,46,47,58
BARCODE_CODE93	72	1≤n≤255	0≤data≤127
BARCODE_CODE128	73	2≤n≤255	0≤data≤127
BARCODE_STANDARD PDF417	101	2≤n≤928	0≤data≤255
BARCODE_TRUNCATE D_PDF417	102	2≤n≤928	0≤data≤255
BARCODE_QRCODE1	103	2≤n≤928	0≤data≤255
BARCODE_QRCODE2	104	2≤n≤928	0≤data≤255

int width
    [in] Bar code width effective value range: 1-6, when bar code print width exceeds printable area, bar code print will not be executed. The parameter is ineffective for 2D code.
int height
    [in] Set bar code print height. Effective range:1-255, this parameter is ineffective for 2D code.
int alignment
    [in] Set bar code alignment method.

Alignment Method	Value	Description
ALIGNMENT_ LEFT	0	Left alignment
ALIGNMENT_CENTER	1	Center alignment
ALIGNMENT_ RIGHT	2	Right alignment

int hriPosition

[in] Set bar code visible character position.

Position	Value	Description
BRACODE_ HRI_ NONE	0	Not print visible character
BRACODE_ HRI_ABOVE	1	Print visible character above bar code
BRACODE_ HRI_ BELOW	2	Print visible character below bar code
BRACODE_ HRI_ BOTH	3	Print visible character above/below bar code

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.21.　PrintSymbol

This function is for printing QR code.

int PrintSymbol(
　　　void* hPrinter,
　　　int type,
　　　const char* data,
　　　int errLevel,
　　　int width,
　　　int height,
　　　int alignment
);

Parameter：
void hPrinter*
　　　[in] The created target printer object.
int type
　　　[in] QR code type.

Type	Value	Description
BARCODE_STANDARD_PDF417	101	Standard sample PDF417 code
BARCODE_TRUNCATED_PDF417	102	Simple sample PDF417 code
BARCODE_QRCODE1	103	QR Code sample 1
BARCODE_QRCODE2	104	QR Code sample 2

const char data,*
　　　[in] QR code data.

Data Length	Data Range
1≤n≤7089	0≤data≤255

int errLevel
　　　[in] QR code setting error correction level.

Error Correction	Value	Code or Error-tolerant Rate
PDF417_ERROR_CORRECTION_LEVEL_0	48	2

PDF417_ERROR_CORRECTION_LEVEL_1	49	4
PDF417_ERROR_CORRECTION_LEVEL_2	50	8
PDF417_ERROR_CORRECTION_LEVEL_3	51	16
PDF417_ERROR_CORRECTION_LEVEL_4	52	32
PDF417_ERROR_CORRECTION_LEVEL_5	53	64
PDF417_ERROR_CORRECTION_LEVEL_6	54	128
PDF417_ERROR_CORRECTION_LEVEL_7	55	256
PDF417_ERROR_CORRECTION_LEVEL_8	56	512
QRCODE_ERROR_CORRECTION_LEVEL_L	48	7%
QRCODE_ERROR_CORRECTION_LEVEL_M	49	15%
QRCODE_ERROR_CORRECTION_LEVEL_Q	50	25%
QRCODE_ERROR_CORRECTION_LEVEL_H	51	30%

int width
    [in] QR code width   0≤n≤255.
int height
    [in] QR code height   0≤n≤255 (This parameter is ineffective for QR Code).
int alignment
    [in] QR code alignment method

Alignment Method	Value	Description
ALIGNMENT_LEFT	0	Left alignment
ALIGNMENT_CENTER	1	Center alignment
ALIGNMENT_RIGHT	2	Right alignment

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.22. PrintImage

Print specified image (Only supports monochrome bmp format). In page mode, the bit images is only stored in the print buffer and is not printed.

int PrintImage(
    void* hPrinter,
    const char*filePath,
    int scaleMode
);

Parameter：
void hPrinter*
    [in] The created target printer object.
*const char*filePath*
    [in] The complete path of image.
int scaleMode

[in] The scale mode for printing image.

Mode	Value	Description
PRINT_IMAGE_NORMAL	0	Normal mode
PRINT_IMAGE_DOUBLE_WIDTH	1	Double-width mode
PRINT_IMAGE_DOUBLE_HEIGHT	2	Double-height mode
PRINT_IMAGE_QUADRUPLE	3	Quadruple mode

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.23. DefineNVImageCompatible

Define the NV bit image in the NV graphics area. It is able to download several pictures at the same time. The downloaded pictures are numbered from 1. This function is supported only by some printer models and may not be supported by future models.It is recommended to use NV graphics function <DefineNVImage>.

```
int DefineNVImageCompatible(
      void* hPrinter,
      const char**filePathList,
      int imageQty
);
```

Parameter：
void hPrinter*
      [in] The created target printer object.
*const   char**filePathList*
      [in] The specified image path list.
int imageQty
      [in] The specified image quantity.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.24. PrintNVImageCompatible

Print the NV bit image downloaded by <DefineNVImageCompatible>. This function is supported only by some printer models and may not be supported by future models. It is recommended to use NV graphics function <PrintNVImage>.
int PrintNVImageCompatible(
      void* hPrinter,
      int imgNo,
      int scaleMode
);

Parameter：
void hPrinter*
      [in] The created target printer object.
int imgNo
      [in] Print specified nth image (the image serial number which undefined in NV buffer area will not
      be printed)   1≤n≤255.
int scaleMode
      [in] The scale mode for printing image.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.25. PrintDownloadedImageCompatible

Print downloaded bit image.This function is supported only by some printer models and may not be supported by future models. It is recommended to use NV graphics function <PrintDownloadedImage>.
int PrintDownloadedImageCompatible(
      void* hPrinter,
      int scaleMode
);

Parameter：
void hPrinter*
      [in] The created target printer object.
int scalemode
      [in] The scale mode for printing image.

Mode	Value	Description
PRINT_IMAGE_NORMAL	0	Normal mode
PRINT_IMAGE_DOUBLE_WIDTH	1	Double-width mode
PRINT_IMAGE_DOUBLE_HEIGHT	2	Double-height mode
PRINT_IMAGE_QUADRUPLE	3	Quadruple mode

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.26. SelectPageMode

Switch from standard mode to page mode (only effective when printer supports page mode and in standard mode).
int SelectPageMode(
        void* hPrinter
);

Parameter：
void hPrinter*
        [in] The created target printer object.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.27. SelectStandardMode

Switch from page mode to standard mode (only effective in page mode).

int SelectStandardMode(
        void* hPrinter
);

Parameter：
void hPrinter*
        [in] The created target printer object.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data

| ERROR_IO_WRITE_TIMEOUT | -10 | Write data timed out |
| Other values | Other values | the error code returned by the Linux system |

4.28. SelectPrintDirectionInPageMode

In page mode, select printer print direction. This function is only effective in page mode.

int SelectPrintDirectionInPageMode(
    void* hPrinter,
     int direction
);

Parameter：
void hPrinter*
    [in] The created target printer object.
int direction
    [in] Select print direction.

Print Direction	Value	Description	Start Position
PRINT_DIRECTION_LEFT_TO_RIGHT	0	Left->Right	Top left corner
PRINT_DIRECTION_BOTTOM_TO_TOP	1	Bottom->Top	Bottom left corner
PRINT_DIRECTION_RIGHT_TO_LEFT	2	Right->Left	Bottom right corner
PRINT_DIRECTION_TOP_TO_BOTTOM	3	Top->Bottom	Top right corner

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.29. SetAbsoluteVerticalPrintPositionInPageMode

In page mode, set the vertical print position (when print start position is top left corner or bottom right corner, is vertical position setting. When print start position is bottom left corner or top right corner, is horizontal setting).

int SetAbsoluteVerticalPrintPositionInPageMode(
    void* hPrinter,
    int position
);

Parameter：
void hPrinter*
    [in] The created target printer object.

int position
    [in] Set vertical position.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.30. PrintAndReturnStandardMode

Print and return standard mode (only effective in page mode).

int PrintAndReturnStandardMode(
    void* hPrinter
);

Parameter：
void hPrinter*
    [in] The created target printer object.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.31. SetPrintAreaInPageMode

In page mode, set the size and the logical origin of the print area. Both print area width and height cannot be set to 0.
int SetPrintAreaInPageMode(
    void* hPrinter,
    int horizontal,
    int vertical,
    int width,
    int height
);

Parameter：
void hPrinter*
    [in] The created target printer object.

int horizontal
    [in] Set horizontal position of print start(range: 0-32000, unit: dot).
int vertical
    [in] Set vertical position of print start(range: 0-32000, unit: dot).
int width
    [in] Set horizontal width of printable area.
int height
    [in] Set vertical height of printable area.
    When print width is 80mm: horizontal star point = 0, vertical start point = 0, width = 576, height =

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.32. PrintDataInPageMode

Print data in page mode, and not return standard mode after printing (only effective in page mode).
int PrintDataInPageMode(
    void* hPrinter
);

Parameter：
void hPrinter*
    [in] The created target printer object.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.33. SetAbsolutePrintPosition

Moves the print position to n ×(horizontal or vertical motion unit) from the left edge of the print area.
The printer ignores any setting that exceeds the print area.
When standard mode is selected, the horizontal motion unit is used.
When page mode is selected, the horizontal or vertical motion unit is used for the print direction.

int SetAbsolutePrintPosition(

```
        void* hPrinter,
        int position
);
```

Parameter：

void hPrinter*

      [in] The created target printer object.

int position

      [in] Horizontal print start position.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.34. PositionNextLabel

Print the label and locate the start position of next label.

```
int PositionNextLabel(
        void* hPrinter
);
```

Parameter：

void hPrinter*

      [in] The created target printer object.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.35. PrintNVImage

Print the NV graphics data defined by the key codes (kc1 and kc2).
```
int PrintNVImage(
        void* hPrinter,
        unsigned char kc1,
        unsigned char kc2,
);
```

Parameter：
void* hPrinter
    [in] The created target printer object.
unsigned char kc1
    [in] Key code 1. 32≤kc2≤126.
unsigned char kc2
    [in] Key code 2. 32≤kc2≤126.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.36. PrintDownloadedImage

Prints the downloaded graphicsdata defined by the key codes (kc1 and kc2).

int PrintDownloadedImage(
    void* hPrinter,
    unsigned char kc1,
    unsigned char kc2
);

Parameter：
void* hPrinter
    [in] The created target printer object.
unsigned char kc1
    [in] Key code 1. 32≤kc2≤126.
unsigned char kc2
    [in] Key code 2. 32≤kc2≤126.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.37. SetAlign

Set print justification.The justification has no effect in page mode.

```
int SetAlign(
      void* hPrinter,
      int     align
);
```

Parameter:
void hPrinter*
      [in] The created target printer object.
int align
      [in] Set the justification.

Align	Justification
0,48	Left
1,49	Center
2,50	Right

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.38. SetTextBold

Turn emphasized mode on or off.
```
int SetTextBold(
      void* hPrinter,
      int     bold
);
```

Parameter:
void hPrinter*
      [in] The created target printer object.
int bold
      [in] Set the emphasized mode for the text.
      0 : emphasized mode is turned off.
      1 : emphasized mode is turned on.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the

		Linux system

4.39. SetTextFont

Set the text font.
int SetTextFont(
        void* hPrinter,
        int   font
);

Parameter:
void hPrinter*
        [in] The created target printer object.
int font
        [in] Set the font type.

Font	Type
0,48	Font A
1,49	Font B

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.40. SetHorizontalAndVerticalMotionUnits

Set the motion units in the horizontal direction and vertical direction.

int SetHorizontalAndVerticalMotionUnits(
        void* hPrinter,
        int horizontal,
        int vertical
);

Parameter:
void hPrinter*
        [in] The created target printer object.
int horizontal
        [in] Horizontal motion unit   0≤horizontal≤255.
int vertical
        [in] Vertical motion unit   0≤vertical≤255.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success

ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.41. EnableBlackMark

This function is used to enable/disable the black mark function.

int EnableBlackMark (
        void* hPrinter,
        int enable
);

Parameter:
void hPrinter*
        [in] The created target printer object.
int enable
        [in] 0 means to turn off the black mark function, non-zero means to enable the black mark
        function

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.42. SetBlackMarkDistance

This function is used to set the distance between black marks.

int SetBlackMarkDistance(
        void* hPrinter,
         int distance
);

Parameter:
void hPrinter*
        [in] The created target printer object.
int distance
        [in]   Set the black mark interval length, in dots, the value range is 400 <= distance <= 4000

Return Value:

Code	Value	Description

ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.43. SetBlackMarkHeight

This function is used to set the distance between black marks.

int SetBlackMarkHeight(
    void* hPrinter,
    int height
);

Parameter:
void hPrinter*
    [in] The created target printer object.
int height
    [in]   Set the height of the black mark, in points, the value range is 24 <= height <= 240

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.44. DefineUserDefinedCharacters

This function is used to customize the download character set.

int DefineUserDefinedCharacters(
    void* hPrinter,
    unsigned char* data
    int size
);

Parameter:
void hPrinter*
    [in] The created target printer object.
unsigned char* *data*
    [in]   Custom character set data, including header information. The specific structure is as follows
    ：fonttype c1 c2 name[32] fontdata[d1…dk]

When fonttype=0, the font is fontA;

When fonttype=1, the font is fontB;

c1 c2 represents a custom character range(32<=c1<=c2<=255)

name:32 characters can be used to represent the name of the user-defined character set

fontdata：font data

int size

[in]   incoming data length

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.45. DeleteUserDefinedCharacter

This function is used to cancel the user-defined character set.

int DeleteUserDefinedCharacter(
    void* hPrinter,
    int n
);

Parameter:

void hPrinter*

[in] The created target printer object.

int n

[in]   Cancel the character whose code is n among the user-defined characters.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

4.46. GoHomeWithBlackMark

Feed the black mark paper to the printing start position.

int GoHomeWithBlackMark(
    void* hPrinter
);

Parameter:
void hPrinter*
    [in] The created target printer object.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system

[note]   • This command is activated only when the black mark function is enabled.
        • This command sets the next print position at the beginning of a line.
        • Even if this command is executed at the print start position of the marked paper, the printer does not feed the paper to the next print start position.
        • This command needs to be used when using the black mark function

4.47. SetBlackMarkAdjust

Feed the black mark paper to the printing start position.

```
int SetBlackMarkAdjust(
    void* hPrinter,
    int func,
    int direction,
    int offset
);
```

Parameter:
void hPrinter*
    [in] The created target printer object.
int func
    [in]   The value 1 is used to set the starting printing position, and the value 2 is used to set the starting cutting position.
int direction
    [in]   Specify the direction of adjustment, 0/48 is specified as the direction of paper feeding, and 1/49 is specified as the opposite direction of the system.
int offset
    [in]   Position distance, unit:dot.

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out

Other values	Other values	the error code returned by the Linux system

4.48. FirmwareUpgrade

This function is used to upgrade printer firmware

int FirmwareUpgrade(
    void* hPrinter,
    const char* cFileName,
    void (*progressCallback)(float)
);

Parameter:
void hPrinter*
    [in] The created target printer object.
const char cFileName*
    [in] Firmware file path
*void (*progressCallback)(float)*
    update progress callback

describe	Value
update progress	0~1
Update success	ERROR_CM_SUCCESS
Not enough memory	ERROR_CM_INSUFFICIENT_MEMORY
Failed to read file	ERROR_IO_READ_FAILED
Failed to send data	ERROR_IO_WRITE_FAILED

Return Value:

Code	Value	Description
ERROR_CM_SUCCESS	0	success
ERROR_CM_INVALID_HANDLE	-2	failed with invalid handle
ERROR_CM_INVALID_PARAMETER	-1	Invalid argument
ERROR_CM_INSUFFICIENT_MEMORY	-4	failed, out of memory
ERROR_IO_WRITE_FAILED	-9	Failed to send data
ERROR_IO_WRITE_TIMEOUT	-10	Write data timed out
Other values	Other values	the error code returned by the Linux system