Main function description

public static void POS_S_SetBarcode(String strCodedata, int nOrgx, int nType, int nWidthX, int nHeight, int nHriFontType, int nHriFontPosition)..... 3
public static void POS_S_SetQRcode(String strCodedata, int nWidthX,int nErrorCorrectionLevel)..... 6
public static void POS_S_TextOut(String pszString, int nOrgx, int nWidthTimes, int nHeightTimes, int nFontType, int nFontStyle)..... 7
public static void POS_FeedLine()..... 9
public static void POS_Reset() 10
public static void POS_SetLineSpacing(int nDistance)..... 12
public static void POS_SetRightSpacing(int nDistance)..... 13
public static void POS_S_SetAreaWidth(int nWidth)..... 14
public static void POS_SetCharSetAndCodePage(int nCharSet, int nCodePage)..... 15
public static void POS_SetKey(byte[] key)..... 17

public static void POS_PrintPicture(Bitmap mBitmap, int nWidth, int nMode)

Description:
Scaling the bitmap to the specified width and print the bitmap

Parameter:
mBitmap
The needed print bitmap

nWidth
The width needs to scale; the Max of printer is 384 pixels. And nWidth scaling aligned with 8.

nMode
Pls use 0

Return value:
N/A

Remark:
N/A

public static void POS_S_SetBarcode(String strCodedata, int nOrgx, int nType, int nWidthX, int nHeight, int nHriFontType, int nHriFontPosition)

Description

Setting and print bar code.

Parameter

pszInfoBuffer

[in] Bar code character string

nOrgx

[in] Assign the level start point and the distance points of left boundary which bar code will print

It can be 0 to 65535

nType

[in] Assign specified bar code type

Can be the number listed in the following table.Moreover pls refer “appendix B bar code state”.

Value	Meaning
0x41	UPC-A
0x42	UPC-C
0x43	JAN13(EAN13)
0x44	JAN8(EAN8)
0x45	CODE39
0x46	ITF
0x47	CODEBAR
0x48	CODE93
0x49	CODE 128

nWidthX

[in] Assign the basic element width of the bar code.

Can be the number(n) listed in the following table.

n	Single basic module width (continuous)	Double basic module width(discrete type)	
		Narrow element width	Wide element width
2	0．25mm	0．25mm	0．625mm
3	0．375mm	0．375mm	1．0mm
4	0．5mm	0．5mm	1．25mm
5	0．625mm	0．625mm	1．625mm
6	0．75mm	0．75mm	1.875mm

nHeight

[in] Assign the height point of bar code.

Can be 1 to 255.Default is 162 dot.

nHriFontType

[in]Assign HRI（Human Readable Interpretation）character’s font type.
Can be the number listed in the following table.

Value	Meaning
0x00	Standard ASCII
0x01	Compress ASCII

nHriFontPosition

[in] Specify the position of HRI（Human Readable Interpretation）character.
Can be one of the values listed in the following list.

Value	Meaning
0x00	No printing
0x01	Just print above the barcode
0x02	Just print below the barcode
0x03	Print both above and below the barcode

Returned value:
None
Remark:
If the barcode is too wide beyond the maximum printing width of printers, it will not be printed.

public static void POS_S_SetQRcode(String strCodedata, int nWidthX,int nErrorCorrectionLevel)

Description:
Print 2D Code

Parameter:
strCodedata
2D Code data

nWidthX
2D Code per unit width, range from 2 to 6

nErrorCorrectionLevel
Error correction level, range from 1 to 4

public static void POS_S_TextOut(String pszString, int nOrgx, int nWidthTimes, int nHeightTimes, int nFontType, int nFontStyle)

Description:
Send the character string data that will be printed to print buffer, and specify the absolute starting point position of X direction (horizontal), specify amplification factor, type and style of each character in both width and height direction.

Parameter:

pszString
[in] character strings need to print.

nOrgx

[in] Specify dots number that the starting point position in X direction (horizontal) separates from the left margin.
Can be 0 to 65535.。

nWidthTimes

[in] specify the amplification factor of characters in width.
Can be 0 to 1.。

nHeightTimes

[in] specify the amplification factor of characters in height.
Can be 0 to 1.

nFontType

[in] specify character font type.
Can be one of the values listed in the following list.

Value	Meaning
0x00	Standard ASCII

0x01	Compress ASCII

nFontStyle

[in] specify character font style.
Can be one or several value listed in the following table.

Value	Meaning
0x00	Normally
0x08	bold
0x80	Underline 1 dot thick
0x100	Underline 2 dot thick
0x200	Upside-down（Effective only at the beginning）
0x400	white/black reverse（white on black）
0x1000	Each character rotates 90 degrees clockwisely.

Remark:
If the length of text is less than one line, it needs **POS_Feedline** to feed paper to print completely.

public static void POS_FeedLine()

Description:
Feed paper a line

Parameter:
None

Return value:
None

Remark:
None

public static void POS_Reset()

Description:
Reset the printer

Parameter:
None

Return value:
None

Remark:
None

public static void POS_SetMotionUnit(int nHorizontalMU, int nVerticalMU)

Description:

Set the printer motion units

Parameter:

nHorizontalMU

[in] set the Horizontal motion units to 25.4 / nHorizontalMU mm.
Can be 0 to 255.

nVerticalMU

[in] set the Vertical motion units to 25.4 / nVerticalMU mm.
Can be 0 to 255.
Return value: None
Remark: None

public static void POS_SetLineSpacing(int nDistance)

Description:
Set characters Line Spacing

Parameter:
nDistance
[in] specifies the dots of line-height.
Can be 0-255. The distance of each dots is associated with the resolution of printer head.

Return value: None

Remark: None

public static void POS_SetRightSpacing(int nDistance)

Description:
Set the right side spacing of characters (the clearance distance between two characters)

Parameter:
nDistance
[in] Specify the dots number of right spacing.
Can be 0 to 255. The distance of each dots is associated with the resolution of printer head.

Return value:
None
Remark:
None

public static void POS_S_SetAreaWidth(int nWidth)

Description:
Set printing area width in standard mode

Parameter:
nWidth
[in] specifies the width of the printing area.
Can be 0 to 65535 dots.

Return value:
None
Remark:
None

public static void POS_SetCharSetAndCodePage(int nCharSet, int nCodePage)

Description:
Choose international character sets and code page.

Parameter:
nCharSet
[in] Specify the international character sets. The symbol definition of different international character sets corresponding to ASCII value from 0x23 to 0x7E is different

Can be one of the values in following table.

Value	Meaning
0x00	U.S.A
0x01	France
0x02	Germany
0x03	U.K.
0x04	Denmark I
0x05	Sweden
0x06	Italy
0x07	Spain I
0x08	Japan
0x09	Nonway
0x0A	Denmark II
0x0B	Spain II
0x0C	Latin America
0x0D	Korea

nCodePage

[in] specify the character code page. The symbol definition of different code page corresponding to ASCII value from 0x23 to 0x7E is different.

Can be one of the values in the following table.

Value	Meaning
0x00	PC437 [U.S.A. Standard Europe
0x01	Reserved
0x02	PC850 [Multilingual]

0x03	PC860 [Portuguese]
0x04	PC863 [Canadian-French]
0x05	PC865 [Nordic]
0x12	PC852
0x13	PC858

Return value:
None
Remark:
None

public static void POS_SetKey(byte[] key)

Description:
Set 8 bytes secret key.

Parameter:
key
8 bytes secret key.

Return value:
None

Remark:

Pls use 8 character(English or number)as parameter.

public static boolean POS_CheckKey(byte[] key, byte[] random)

Description:

Setting 8 key.

Parameter:
key
8 byte key

random

8 byte random number

Returned value:

If the key of lower machine is same as the above KEY,then go back to TRUE,if not,go back to FALSE.

Remark:

The principle of Check Key is like this:

Upper machine create 8 byte random number group,then send to lower machine,lower machine received the data and encrypt,after that goes back to upper machine.

Upper machine use own key,encrypt for random number group.

By comparing the encryption result with these two machine,judge the KEY is the same or not.