

JposPrinter JavaPOS Driver

-User Manual-

1. Introduction

JposPrinter JavaPos driver was developed to be compatible with JavaPOS standard version 1.13, supporting POS80 relevant products of JposPrinter.

Support communication interfaces below:

USB

Serial/RJ232C

Lan/Ethernet

This software package provides POSPrinter driver support for JAVA POS, the documents below need to be included to support the main functions:

jcl.jar

jpos.xml

jpos113-controls.jar

Xercesimpl.jar

xml-apis.jar

JposPrinterJavaPOS.jar

JposPrinterJIOx32.dll

JposPrinterJIOx64.dll

2. Operating Environment

2.1 operating system requirements

x86 framework operating system below are supported by this software.

- Windows 10 64-bit
- Windows 10 32-bit
- Windows 8.1 64-bit
- Windows 8.1 32-bit
- Windows 8 64-bit
- Windows 8 32-bit
- Windows 7 64-bit
- Windows 7 32-bit
- Windows XP 32-bit

2.2 Environment requirements

This software package must be operated under JRE Ver 1.42 or higher version. JRE Ver1.5 or later is recommended.

If you are developer user, please make sure that your JDK environment configuration completed.

The following application should be included:

Javac.exe

Java.exe

Javadoc.exe

And use them in the command line normally.

```
C:\Users>java -version
java version "1.8.0_91"
Java(TM) SE Runtime Environment (build 1.8.0_91-b14)
Java HotSpot(TM) 64-Bit Server VM (build 25.91-b14, mixed mode)

C:\Users>javac -version
javac 1.8.0_77
```

3. JavaPOS Driver Setting

JposPrinter JavaPOS mainly set by attribute of jpos.xml document.

```
<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE JposEntries PUBLIC "-//JavaPOS//DTD//EN" "jpos/res/jcl.dtd">

<JposEntries>

    <JposEntry logicalName="JposPrinter_serial">

        <creation                                factoryClass="net.jposprinter.javaposdrv.ServiceInstanceFactory"
serviceClass="net.jposprinter.javaposdrv.POSPrinterService" />

        <vendor name="JposPrinter" url="www.JposPrinter.net" />

        <jpos category="POSPrinter" version="1.13" />

        <product description="JposPrinter receipt printer" name="JposPrinter POSPrinter controller"
url="www.JposPrinter.net" />

        <prop name="model" type="String" value="POS80" />

        <prop name="portName" type="String" value="COM1" />

        <prop name="portSettings" type="String" value="19200" />

        <prop name="ioTimeoutMillis" type="Integer" value="5000" />

        <prop name="DBCS" type="String" value="" />

    </JposEntry>

</JposEntries>
```

3.1 Interface setting

See in jpos.xml document comes with software package.

Serial/RJ232C:

```
<prop name="portName" type="String" value="COM1" /> ----port name:COMx
```

```
<prop name="portSettings" type="String" value="19200" />----baud rate
```

This driver support Serial/RJ232C baud rate as below:

9600, 19200, 38400, 115200

If it is not the above number, the communication will not be established correctly.

```
<prop name="ioTimeoutMillis" type="Integer" value="5000" />---IO timeout, unit: ms
```

USB:

```
<prop name="portName" type="String" value="usb" /> ---port name: fixed to usb(notice, lowercase)
```

```
<prop name="portSettings" type="String" value="" /> ----not need to set
```

```
<prop name="ioTimeoutMillis" type="Integer" value="5000" />---IO timeout, unit:ms
```

Lan/Ethernet

```
<prop name="portName" type="String" value="tcp:192.168.1.123" />----printer IP address
```

```
<prop name="portSettings" type="String" value="" />----no need to set
```

```
<prop name="ioTimeoutMillis" type="Integer" value="5000" />----IO timeout, unit:ms
```

3.2 Printing character traits setting

Double-byte setting:

```
<prop name="DBCS" type="String" value="GB2312" />---double-byte character code setting
```

the following are supported, according to the language that printer supported

SJIS-----Japanese

GB2312-----Chinese-simplified

GB18030-----Chinese-simplified

Big5-----Chinese Traditional

EUC_KR-----Korean

4. Testing routine

Pls mainly refer to Source Code file "JposPrinterReceiptTest.java" of the software package and handle as follows

Ensure the following files are in the same catalog:

```
jcl.jar
jpos.xml
jpos113-controls.jar
xercesimpl.jar
xml-apis.jar
JposPrinterJavaPOS.jar
JposPrinterJIOx32.dll
JposPrinterJIOx64.dll
JposPrinterReceiptTest.java
posprinter.bmp
```

1. Configure "jpos.xml" file first according to the situation.

After configured, using the command:

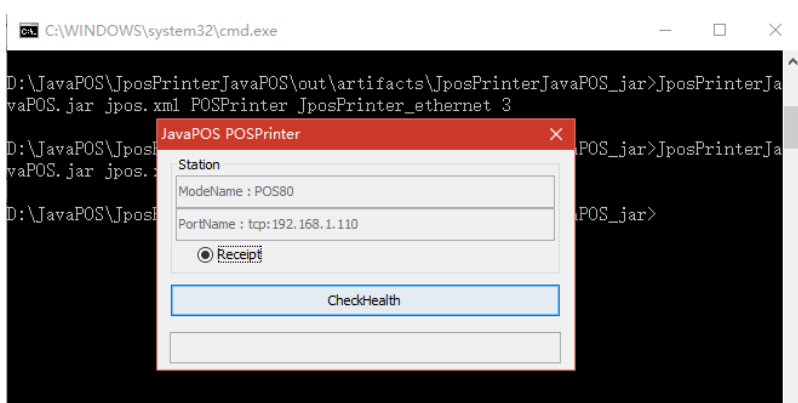
```
JposPrinterJavaPOS.jar jpos.xml POSPrinter JposPrinter_XXXXXX 3
```

(XXXXXX represent <JposEntry logicalName="JposPrinter_XXXXX"> of configuration)

For example: <JposEntry logicalName="JposPrinter_ethernet">

So command as:

```
JposPrinterJavaPOS.jar jpos.xml POSPrinter JposPrinter_ethernet 3
```



If all configuration and the printer is normal, the detection information will be printed out

2. Edit **JposPrinterReceiptTest.java**, Note: The file path here must specify "jpos.xml" correctly

```
* If you want to place the jpos.xml file elsewhere on your local file
* system then uncomment the following line and specify the full path to
* jpos.xml.
*/
System.setProperty( JposPropertiesConst.JPOS_POPULATOR_FILE_PROP_NAME, "jpos.xml");
```

3. As described in the "jpos.xml" file.

```
// open the printer object according to the entry names defined in jpos.xml
//printer.open("JposPrinter_USB");
//printer.open("JposPrinter_serial");
printer.open("JposPrinter_ethernet");
```

4. Confirm the file without mistakes, then save it, making sure your JDK environment is well configured, finally use the javac command to compile the application.

Edit the command(All files are in the same directory):

```
javac -encoding utf-8 -classpath jpos113-controls.jar;jcl.jar JposPrinterReceiptTest.java
```

```
D:\JavaPOS\JposPrinterJavaPOS\out\artifacts\JposPrinterJavaPOS_jar>javac -encoding utf-8 -classpath jpos113-controls.jar;jcl.jar JposPrinterReceiptTest.java
```

If no compilation error occurs, the JposPrinterReceiptTest.class file is generated in the directory

5. Execution test application: JposPrinterReceiptTest

```
java -classpath .;JposPrinterJavaPOS.jar;jpos113-controls.jar;jcl.jar;xercesimpl.jar;xml-apis.jar JposPrinterReceiptTest
```

```
D:\JavaPOS\JposPrinterJavaPOS\out\artifacts\JposPrinterJavaPOS_jar>java -classpath .;JposPrinterJavaPOS.jar;jpos113-controls.jar;jcl.jar;xercesimpl.jar;xml-apis.jar JposPrinterReceiptTest
Async transaction print submitted: time = 1468509139249 output id = 1
OutputCompleteEvent received: time = 1468509139328 output id = 1
JposPrinterReceiptTest finished.
```

5. JavaPOS Programming object

This software is mainly developed based on JAVAPOS standard, if more detailed JAVAPOS programming are needed, please refer to the JAVAPOS programming reference manual. Specific details pls visit:

<http://www.javapos.com/>

When using PrintBarCode, only support the following type of barcode

PTR_BCS_UPCA, PTR_BCS_UPCE, PTR_BCS_JAN8, PTR_BCS_JAN13, PTR_BCS_ITF, PTR_BCS_Codabar, PTR_BCS_Code39, PTR_BCS_Code93, PTR_BCS_Code128, PTR_BCS_QRCODE,---*1 PTR_BCS_PDF417 ----*2

Notice:

*1,*2 When using both types, the size setting parameter will not work, but must be set to within the normal print range

1. When "text" is sent by default, the default parameter is used to generate QRCODE.To set the size and error correction level of QRCODE, pls use the following format:

Hexadecimal 1b N1 N2 "text"

N1: Unit point size

N2: correct level

Please refer to the QRCODE instructions in the programming manual of printer for details

2. When "text" is sent by default, PDF417 is generated using the default parameter. To set the relevant parameters of PDF417, pls use the following format:

Hexadecimal 1b N1 N2 N3 N4 N5 N6 N7 "text"

N1:column number

N2: number of rows

N3: width

N4: row height

N5: correction rating description type

N6: correction level value

N7: set / cancel truncation mode

Please refer to the PDF417 instructions in the programming manual of printer for details