

# **OPERATION MANUAL**

JavaPOS Driver Kit



This JavaPOS Driver Kit OPERATION MANUAL (hereinafter referred to as "the GUIDE") describes the procedures and precautions for using the JavaPOS Driver Kit (hereinafter referred to as "the Kit").

The GUIDE assumes that the reader is familiar with the following:

- General characteristics of POS peripheral devices
- Java terminology and architecture
- Java for Retail POS (JavaPOS for short) Programmer's Guide

Notes:

- It is prohibited to use or duplicate a part or whole of the GUIDE without the permission of Toshiba TEC Corporation.
- The GUIDE is subject to change without prior notice.
- \* Windows, Windows XP, Windows Vista and Windows 7 are registered trademarks of Microsoft Corporation in the United States and/or other countries.

The official name of Windows is the "Microsoft Windows Operating System".

\* Java and JavaPOS are trademarks of Sun Microsystems in the United States and/or other countries.



## **Table of Contents**

Introduction4
Overview of the MANUAL
1. How to Build An Operating Environment5
Supported Product
Operating Environment
Supported Devices
Installation of Java Runtime Environment6
Installation of Drawer Driver7
Patch application to Keyboard Driver14
Installation of the Linux Keyboard compatible Driver
Installation of iButton Driver17
Installation of TECUSB 21
Installation of Keyboard Driver
2. How to Check Performance of the JavaPOS Device Service30
PREPARE
Default value
OPERATION
3. How to Use the JavaPOS Device Service
Example of Creating An Application Using the JavaPOS Device Service
Coding Process
Creation of Window



## Introduction

The Kit (JavaPOS Driver Kit) provides the JavaPOS Device Service to be used to develop an application for key lock (Keylock), cash drawer (Drawer), magnetic stripe reader (MSR), line display (LineDisplay), scanner (Scanner) and POS printer (POSPrinter).

#### **Overview of the MANUAL**

The GUIDE consists of the following three steps, and explains the demo program enclosed in the Kit. Please follow the steps below:

#### Step 1. How to Build An Operating Environment

This step describes the method to build an environment for using the JavaPOS Device Service.

#### Step 2. How to Check Performance of the JavaPOS Device Service

This step describes the method to test whether or not the test program for checking performance of the JavaPOS Device Service (CheckHealth.jar) successfully runs.

#### Step 3. How to Use The JavaPOS Device Service

This step describes the method to create a unique application using the JavaPOS Device Service (Keylock, Drawer, MSR, LineDisplay, Scanner, POSPrinter).

#### Example of Creating An Application Using the JavaPOS Device Service

This chapter explains the method to operate the JavaPOS Device Service using an example. The demo program described here is a sample code which uses the JavaPOS Device Service to create the application.



## 1. How to Build An Operating Environment

This chapter describes the method to build an environment where the JavaPOS Device Service operates. Please take this step (Step 1) first, then go to Step 2 (Chapter 2 "How to Check Performance of the JavaPOS Device Service") or Step 3 (Chapter 3 "How to Use the JavaPOS Device Service").

#### **Supported Product**

- ST-A10 : ST-A10 does not support Linux OS.
- ST-A20

#### **Operating Environment**

Performance of the JavaPOS Device Service was checked under the following environment:

Operating system:Windows 2000, Windows XP, WEPOS, Windows Vista, Windows 7<br/>SUSE Linux Enterprise Desktop 10 SP1,<br/>SUSE Linux Enterprise Desktop 11 (\*none SP1)JavaRuntime:JRE1.4.2<br/>JavaPOS:JavaPOS:JavaPOS 1.11

#### **Supported Devices**

[Keylock]

- ➢ KITST-A10-BTNK
- PKBST-52 Keylock

#### [CashDrawer]

- DRWST-5x Cash Drawer
- DRWST-5x Cash Drawer (Extension)
- Cash Drawer connected to TRST-Axx

#### [LineDisplay]

- LIUST-A10
- ➢ WD-111

#### [MSR]

- ➢ MCRST-A10
- PKBST-52 MSR

(\*MSR JavaPOS Device Service does not support MCRST-76 under Linux OS.)



#### [Scanner]

- HS-530-RS Serial Barcode Scanner
- > HS-530-RS Serial Barcode Scanner (Extension)

(\*Scanner JavaPOS Device Service does not support HS-530-RS under SLED11.)

[POSPrinter]

- > TRST-A10-S-QM-R, TRST-A10-P-QM-R, TRST-A10-U-QM-R, TRST-A10-L-QM-R
- > TRST-A15-S-QM-R, TRST-A10-P-QM-R, TRST-A15-U-QM-R
- > TRST-A10-S-CN-R, TRST-A10-P-QM-R, TRST-A10-U-CN-R
- > TRST-A15-S-CN-R, TRST-A10-P-QM-R, TRST-A15-U-CN-R
- > TRST-A00-UF-QM-R, TRST-A00-DF-QM-R, TRST-A00-UC-QM-R, TRST-A00-DC-QM-R

#### Installation of Java Runtime Environment

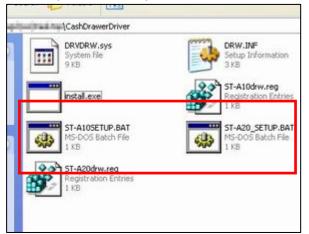
Download the file from the following web site and implement the Java Runtime Environment. <u>http://java.sun.com/products/archive/j2se/1.4.2\_16/</u>



#### **Installation of Drawer Driver**

<Windows>

1. Startup of driver setup batch file



#### ST-A10:

Select "CashDrawer Driver"  $\rightarrow$  "Windows", then double-click on the ST-A10SETUP.BAT file to execute.

#### ST-A20:

Select "CashDrawer Driver" → "Windows", then double-click on the ST-A20SETUP.BAT file to execute.

\*1 Under Windows Vista, right-click on the icon of the SETUP.BAT file, and select "Run As Administrator" to execute.
 \*2 DRVDRW.sys is a CashDrawer driver which runs on Windows. TECCashDrawerJni.dll is a library file which is used to access the Windows CashDrawer driver from Java.

#### 2. Start of installation



Press any key to start the installation.



#### 3. Permission to add registry



A confirmation window appears. Click on the [Yes] button.

#### 4. Result of registry addition

Registry	y Editor 🛛 🔀
į)	Information in DRW.REG has been successfully entered into the registry.

When the registry has been successfully added, the window shown above appears. Click on the [OK] button.

#### 5. Permission to install the driver



When the window shown above appears, click on the [Continue Anyway] button.



#### 6. Confirmation of installation

	Display adapters
rc	😟 🗃 Floppy disk controllers
è	🕀 🖾 Human Interface Devices
	😟 🚍 IDE ATA/ATAPI controllers
	庄 🦢 Keyboards
gε	😟 🐚 Mice and other pointing devices
r	🛨 夏 Monitors
	🛨 🎬 Network adapters
0	🖻 🥵 Other devices
	🕀 🍠 Ports (COM & LPT)
	🛨 🜨 Processors
	🛨 🧶 Sound, video and game controllers
	🛨 🥪 Storage volumes
	🖾 🧐 Suctem devices

Start the Device Manager and check that the CashDrawer driver has been successfully installed. Then, open the system32 folder (C:¥WINDOWS¥system32) and make sure that TECCashDrawerJni.dll has been copied.

\*When you install this driver on Windows Vista, you have to execute exceptional steps. Please follow the below.

1. Copy files

Copy the following file to particular place.

- DRVDRW.sys
   C:¥OPOS¥TEC¥SYS
- DRW.INF
   C:¥OPOS¥TEC¥SYS
- install.exe
   C:¥OPOS¥TEC¥SYS
- TECCashDrawerJni.dll C:¥Windows¥system32

#### 2. Add Hardware

Select "Add Hardware " icon in "Control Pane".

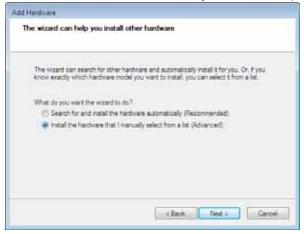




#### Click [Next] button.



Select "Install the hardware that I manually select from a list (Advanced)" radio button, then Click [Next] button.



Select "Show All Devices", then Click [Next] button.





#### Click [Have Disk] button.



Click [Browse] button and select "DRW.INF" file in the "C:¥OPOS¥TEC¥SYS" folder.



#### Click [Next] button.

Finally, click [Finish] button to finish the hardware addition.

Add Hardware	
The wizard is ready to	inatali yoor hardware
Hardware to install	
	IEC DRAWER Driver
To start installing your n	ww.hardware.click.Next.
	Back Ned > Cancel

3. Registry addition Execute "ST-A10drw.reg" or "ST-A20drw.reg" to add registry.



#### < Linux – SLED10SP1 and SLED11>

#### 1. Copy of driver module



Select "Driver"  $\rightarrow$  "CashDrawer Driver"  $\rightarrow$  "Linux"  $\rightarrow$  "Driver"  $\rightarrow$ "SLED10SP1" or "SLED11". (Please select OS used.) Copy the drw.ko driver module to any desired location.

\* drw.ko is a CashDrawer driver which runs on Linux. libTECCashDrawerJni.so.0.0 in "JNI" folder is a library file which is used to access LinuxCashDrawerDriver from Java.

#### 2. Installation of CashDrawer driver

#### ST-A10

]# insmod /home/tec/tecdrv/drw.ko

]# mknod /dev/drw c 242 0

Execute the above commands to install the driver.

(The above is an example when drw.ko has been copied to "/home/tec/tecdrv".)

#### ST-A20

]# insmod /home/tec/tecdrv/drw.ko port=0x48C

]# mknod /dev/drw c 242 0

Execute the above commands to install the driver.

(The above is an example when drw.ko has been copied to /home/tec/tecdrv.)

\*The shaded value is the ST-A20 drawer port address. As the port address differs for each model, please check it with the specifications of each model.

- \*1 The 1st line installs the module and the 2nd line creates a device file.
- \*2 This setup is cleared whenever the operating system is rebooted. This command must be executed every time the operating system is rebooted



#### 3. Confirmation of install

]# lsmod

Execute the above command to make sure the driver has been successfully installed.

4. Generation of link file

Select "Driver"  $\rightarrow$  "CashDrawer Driver"  $\rightarrow$  "Linux"  $\rightarrow$  "JNI". Copy the libTECCashDrawerJni.so.0.0 to the location you want to run the program. Then generate a link file.



#### Patch application to Keyboard Driver

#### <Linux - SLED10 SP1>

The source file of a keyboard driver is rewritten using "tec\_kbd-2.6.11.patch" file.

- /usr/src/linux/drivers/input/keyboard/atkbd.c
- /usr/src/linux/drivers/input/serio/i8042.c
- /usr/src/linux/drivers/char/keyboard.

1. Copy files

Copy the following files to "/home/tec/tecdrv/" from "Driver"  $\rightarrow$  "Linux Keyboard patch"  $\rightarrow$ "SLED10SP1" folder in this Kit.

• tec\_kbd-2.6.11.patch

Execute the following command. Then, a patch is applied and a source file can be rewritten.

]# cp -p /home/tec/tecdrv/tec\_kbd-2.6.11.patch /usr/src/linux/ ]# cd /usr/src/linux/ ]# patch -p0 < tec\_kbd-2.6.11.patch

Rebuild a kernel, after these two procedures are completed. *Cautions : Rebuild a kernel takes several hours.* 

#### 2. Rebuild the kernel

Execute the following commands in order from the top:

]# cd /usr/src/linux/	₽	Movement to a directory with the source file of a kernel.
]# make oldconfig	₽	Obtains configuration information of the kernel in operation.
]# make clean	₽	Deletes all interim files.
]# make	₽	Complies the kernel and driver modules.
]# make install	₽	Installs the kernel.
]# make modules_install	₽	Installs the drivers.

When the touch panel driver provided by Elo TouchSystems has been installed, dependency information among the touch panel modules will disappear. Execute the following commands and register the dependency information again.

]# cd /etc/opt/elo-ser/setup/	(or	/etc/opt/elo/)
]# ./install.sh		

Finally, restart the operating system.



#### <Linux – SLED11>

The source file of a keyboard driver is rewritten using "tec\_kbd-2.6.27.patch" file.

- /usr/src/linux/drivers/input/keyboard/atkbd.c
- /usr/src/linux/drivers/input/serio/i8042.c
- /usr/src/linux/drivers/char/keyboard.

1. Copy the following files to "/home/tec/tecdrv/" from "Driver" → "Linux Keyboard patch"

 $\rightarrow$  "SLED11" folder in this Kit.

tec\_kbd-2.6.27.patch

Execute the following command. Then, a patch is applied and a source file can be rewritten.

]# cp -p /home/tec/tecdrv/tec\_kbd-2.6.27.patch /usr/src/linux/ ]# cd /usr/src/linux/ ]# patch -p0 < tec\_kbd-2.6.27.patch

2. Patch application to w1 driver of Linux Kernel for TTEC iButton driver

The source file of a w1 driver is rewritten using "tec\_w1-2.6.27.patch" file.

/usr/src/linux/drivers/w1/masters/ds2490.c

Copy the following files to "/home/tec/tecdrv/" from "Driver"  $\rightarrow$  "Linux iButton Driver"  $\rightarrow$  "patch" folder in this Kit.

- w1-patch-install.sh
- ds2490.h
- tec\_w1-2.6.27.patch

Execute the following procedure, when using the iButton driver developed by TTEC.

]# /home/tec/tecdrv/w1-patch-install.sh

Rebuild a kernel, after these two procedures are completed. *Cautions : Rebuild a kernel takes several hours.* 



#### 3. Rebuild the kernel

Execute the following commands in order from the top:

]# cd /usr/src/linux/	¢	Movement to a directory with the source file of a kernel.
]# make oldconfig	₽	Obtains configuration information of the kernel in operation.
]# make clean	₽	Deletes all interim files.
]# make	₽	Complies the kernel and driver modules.
]# make install	₽	Installs the kernel.
]# make modules_install	₽	Installs the drivers.

When the touch panel driver provided by Elo TouchSystems has been installed, dependency information among the touch panel modules will disappear. Execute the following commands and register the dependency information again.

]# cd /etc/opt/elo-ser/setup/	(or	/etc/opt/elo/)
]# ./install.sh		

Finally, restart the operating system.

#### Installation of the Linux Keyboard compatible Driver

<Linux – SLED10SP1 and SLED11>

If a keyboard compatible driver is installed, "setkeycodes" command can be executed even when a PS/2 keyboard has not been connected.

Note: Restarting the operating system clears this setting. The procedure must be performed whenever the operating system is restarted.

Note: It is necessary to execute this procedure ahead of the procedure since 3.3.

1. Copy files

\*1

Copy the following files to "/home/tec/tecdrv/" from "Driver"  $\rightarrow$  "Linux compat kbd"  $\rightarrow$ "SLED10SP1" or "SLED11" (Please select OS used.) folder in this Kit.

compat\_keyb.ko

Execute the following command to install a Keyboard compatible driver.

This setup is cleared whenever the operating system is rebooted.

This command must be executed every time the operating system is rebooted

]# insmod /home/tec/tecdrv/compat\_keyb.ko



#### Installation of iButton Driver

<Windows>

Download the file from the following web site and implement the 1-Wire Drivers. <u>http://japan.maxim-ic.com/products/ibutton/software/tmex/index.cfm</u>

<Linux - SLED10SP1>

It is a required setup when using the iButton driver. Execute this procedure in advance.

[Copy files]

Copy the following files to "/home/tec/tecdrv/" from "Driver"  $\rightarrow$  "Linux iButton Driver"  $\rightarrow$  "Driver"  $\rightarrow$ "SLED10SP1" folder in this Kit.

- load\_usbibutton
- usbibutton.ko

[Source file modification of the kernel attached driver]

Modify two places of the "dscore.c" file as follows. Execute this procedure, when using the iButton driver developed by TTEC.

Target source file

/usr/src/linux/drivers/w1/dscore.c

[Contents of modification]

1. the 556th line from the top

(Before Modification) #if 0
int ds_search(struct ds_device *dev, u64 init, u64 *buf, u8 id_number, int conditional_search) {
(snip) }
int ds_match_access(struct ds_device *dev, u64 init) {



(After Modification)	
//#if 0	
int ds_search(struct ds_device *dev, u64 init, u64 *buf, u8 id_number, int conditional_search)	
{	
(snip)	
}	
#if O	
int ds_match_access(struct ds_device *dev, u64 init)	
{	

#### 2. the 789th line from the top

(Before Modification) #if 0 EXPORT\_SYMBOL(ds\_start\_pulse); EXPORT\_SYMBOL(ds\_set\_speed); EXPORT\_SYMBOL(ds\_detect); EXPORT\_SYMBOL(ds\_stop\_pulse); EXPORT\_SYMBOL(ds\_search); #endif

## ➡

(After Modification) #if 0 EXPORT\_SYMBOL(ds\_start\_pulse); EXPORT\_SYMBOL(ds\_set\_speed); EXPORT\_SYMBOL(ds\_detect); EXPORT\_SYMBOL(ds\_stop\_pulse); #endif EXPORT\_SYMBOL(ds\_search); //#endif



Rebuild a kernel, after these two procedures are completed. *Cautions : Rebuild a kernel takes several hours.* 

#### 3. Rebuild the kernel

Execute the following commands in order from the top:

]# cd /usr/src/linux/	<b>□</b> >	Movement to a directory with the source file of a kernel.
]# make oldconfig	⊏>	Obtains configuration information of the kernel in operation.
]# make clean	⊏>	Deletes all interim files.
]# make	⊏>	Complies the kernel and driver modules.
]# make install	<b>⊨</b> >	Installs the kernel.
]# make modules_install	<b>⊑</b> >	Installs the drivers.

When the touch panel driver provided by Elo TouchSystems has been installed, dependency information among the touch panel modules will disappear. Execute the following commands and register the dependency information again.

]# cd /etc/opt/elo-ser/setup/	(or	/etc/opt/elo/)
]# ./install.sh		

Finally, restart the operating system.

#### Installation of module

\*1 Before installing the USB iButton driver, connect the USB iButton adapter.

\*2 This setup is cleared whenever the operating system is rebooted.

This command must be executed every time the operating system is rebooted

]# insmod /home/tec/tecdrv/compat\_keyb.ko

]# cd /home/tec/tecdrv/ ]# chmod 777 load\_usbibutton ]# ./load\_usbibutton ]# setkeycodes 0x68 93



#### <Linux - SLED11>

[Copy files]

Copy the following files to "/home/tec/tecdrv/" from "Driver"  $\rightarrow$  "Linux iButton Driver"  $\rightarrow$  "Driver"  $\rightarrow$  "SLED11" folder in this Kit.

- load\_usbibutton
- usbibutton.ko

#### Installation of module

\*1 Before installing the USB iButton driver, connect the USB iButton adapter.
 \*2 This setup is cleared whenever the operating system is rebooted.
 This commend must be executed event time the executing system is rebooted.

This command must be executed every time the operating system is rebooted

]# insmod /home/tec/tecdrv/compat\_keyb.ko

]# cd /home/tec/tecdrv/

]# chmod 777 load\_usbibutton

]# ./load\_usbibutton

]# setkeycodes 0x68 93



#### Installation of TECUSB

<Windows>

1. Copy of driver module

C:\TRST-A1x JavaPO	S\TEC	USB Driver
d Folder Tasks	*	Name 🔺
ke a new folder plish this folder to the		C Windows

Copy a "¥TRST-A1x JavaPOS¥TECUSB Driver¥Windows" folder in the suitable place.

2. Execute of Batch file

[Windows 2000 / Windows XP / WEPOS]

Carry out "TECUSB\_LIBRARY\_SETUP.BAT" in the folder which I stored by procedure 1. After practice, the following file is copied by a folder of "¥Windows¥system32". (in the case of the Windows 2000, ¥Winnt¥system32)

- TECUSB.dll
- LogMngr.dll
- TECUSBJNI.dll
- TECUSBPM.exe

[Windows Vista / Windows 7] In case of Windows Vista and Windows 7, copy the following files to each directory manual operation.

"Windows¥system32" directory

- TECUSB.dll
- LogMngr.dll
- TECUSBJNI.dll

"Root directory of project

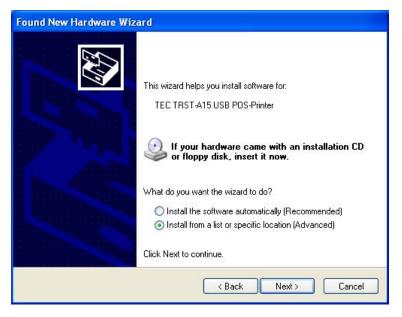
- ex) A folder same as "CheckHealth.bat"
  - TECUSBPM.exe



3. Installation of TRST-A1x TECUSB driver Connect TRST-A1x by USB and turn on a power supply. The following dialogue is displayed.

Found New Hardware Wizard
Welcome to the Found New Hardware Wizard         Windows will search for current and updated software by looking on your computer, on the hardware installation CD, or on the Windows Update Web site (with your permission). Read our privacy policy         Can Windows connect to Windows Update to search for software?         Yes, this time only         Yes, now and gvery time. I connect a device         No, not this time         Click Next to continue.
< <u>Back</u> <u>N</u> ext > Cancel

Choose "No, not this time". Click on the "Next >" button.



Choose "Install from a list or specific location [Advanced]". Click on the "Next >" button.

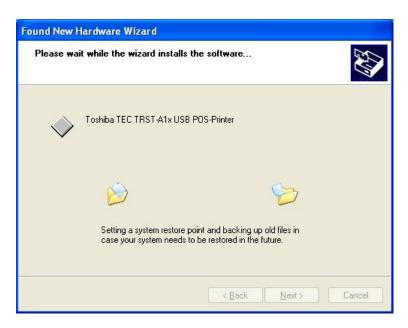


e choose your search and instal	llation options.
Search for the best driver in these loc	ations.
Use the check boxes below to limit or paths and removable media. The best	r expand the default search, which includes local t driver found will be installed.
Search removable <u>m</u> edia (flopp	у, CD-ROM)
☑ Include this location in the sear	rch:
C:\TECUSB Driver\Windows	Biowse
) Don't search. I will choose the driver Choose this option to select the devic the driver you choose will be the best	e driver from a list. Windows does not guarante

Exclude a check box of "Search removable media [floppy, CD-ROM...]".

Choose check box of "Include this location in the search.".

Click on the "Browse" button. And appoint a folder with the "TecUSBDEx.INF" file. Click on the "Next >" button.



The installation of the TECUSB driver is started.



Found New Hardware Wiz	Completing the Found New Hardware Wizard The wizard has finished installing the software for: Toshiba TEC TRST-A1x USB POS-Printer
	Click Finish to close the wizard.

If an above screen is displayed, it is installation completion. Click on the "Finish" button.

🖳 Device Manager	
Eile <u>A</u> ction <u>V</u> iew <u>H</u> elp	
Computer     Computer     Disk drives     System devices     Disk drives     Disk drives	

Finally start device manager. And confirm that it is installed as above.



< Linux – SLED10SP1 and SLED11>

1. Copy of driver module



Select "Driver"  $\rightarrow$  "TECUSB Driver"  $\rightarrow$  "Linux"  $\rightarrow$  "Driver"  $\rightarrow$ "SLED10SP1" or "SLED11". (Please select OS used.) Copy the "tecusbd.ko" driver module to any desired location. \* tecusbd.ko is a TECUSB driver which runs on Linux.

Select "Driver"  $\rightarrow$  "TECUSB Driver"  $\rightarrow$  "Linux"  $\rightarrow$  "Driver" Copy the "libtecusbd.so.0.0" module to any desired location.

2. Installation of driver

\*1 This setup is cleared whenever the operating system is rebooted. This command must be executed every time the operating system is rebooted

]# insmod /home/tec/tecdrv/tecusbd.ko

Execute the above commands to install the driver. (The above is an example when tecusbd.ko has been copied to "/home/tec/tecdrv".)

#### 3. Confirmation of install

]# lsmod

Execute the above command to make sure the driver has been successfully installed.

#### 4. Installation of library

```
]# cp -p /home/tec/tecdrv/libtecusb.so.0.0 /usr/lib/
]# ldconfig -n /usr/lib/
]# ln -s /usr/lib/libtecusb.so.0.0 /usr/lib/libtecusb.so
```

Execute the above commands to install the library.

(The above is an example when libtecusb.so.0.0 has been copied to /home/tec/tecdrv.)

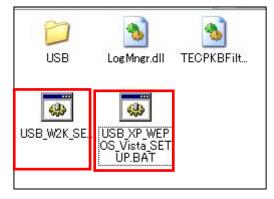


#### Installation of Keyboard Driver

<Windows USB>

Before this installation, execute "Installation of TECUSB" procedure. [Windows 2000, XP, and WEPOS]

1. Startup of driver setup batch file

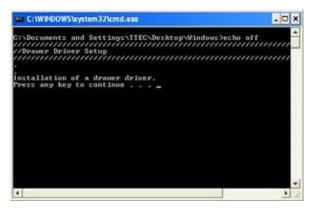


Select "Driver"  $\rightarrow$  "Keyboard Driver"  $\rightarrow$  "Windows", and then double-click on the BAT file to execute.

 Windows 2000:
 USB\_W2K\_SETUP.BAT

 Windows XP, WEPOS, Vista, 7:
 USB\_XP\_WEPOS\_Vista\_7\_SETUP.BAT

#### 2. Start of installation



Press any key to start the installation.



#### 3. Permission to add registry



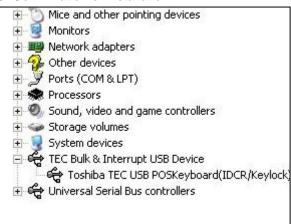
A confirmation window appears. Click on the [Yes] button.

#### 4. Result of registry addition

Registry	y Editor	
Ų.	Information in key.reg has been successfully entered into	) the registry,

When the registry has been successfully added, the window shown above appears. Click on the [OK] button.

#### 5. Confirmation of installation



Start the Device Manager and check that the Keyboard Filter driver has been successfully installed. Then, open the system32 folder (C:¥WINDOWS¥system32) and make sure that TECPKBFilterJNI.dll has been copied.



[Windows Vista, Windows 7]

When you install this driver on Windows Vista and Windows 7, you have to execute exceptional steps. Please follow the below.

#### 1. Run Command Prompt as Administrator

Right-Click "cmd" file in the "Windows" →"system32" folder, and select "Run as Administrator".

#### 2. Run BAT file

Change directory to "Driver"→"Keyboard Driver"→"Windows" folder, and click the "USB\_XP\_WEPOS\_Vista\_7\_SETUP.bat" file.

After finishing above steps, please follow [Windows 2000, XP, and WEPOS] steps.

#### 3. Reboot

When the installation succeeds, please reboot the computer.



<Linux USB - SLED10SP1 and SLED11>

Before this installation, execute "Installation of TECUSB" procedure.

Execute the following commands in order to install the USB POS keyboard driver. This procedure is necessary to operate MCR and Keylock attached to PKBST-52.

[Supported Product] PKBST-52

Note: Restarting the operating system clears this setting. The procedure must be performed whenever the operating system is restarted.

Note: In case of the simultaneous use with USB POS Printer, install usbposkbd.ko before installing tecusbd.ko.

Select "Driver"  $\rightarrow$  "Keyboard Driver"  $\rightarrow$  "Linux"  $\rightarrow$  "Driver"  $\rightarrow$ "SLED10SP1" or "SLED11" (Please select OS used.)  $\rightarrow$ "USB" folder.

Copy the "usbposkbd.ko" module to any desired location.

\*usbposkbd.ko is a USB Keyborad driver which runs on Linux.

1. Installation of module

\*1 This setup is cleared whenever the operating system is rebooted. This command must be executed every time the operating system is rebooted.

]# insmod /home/tec/tecdrv/usbposkbd.ko

Execute the above commands to install the driver. (The above is an example when usbposkbd.ko has been copied to "/home/tec/tecdrv".)

2. Keycode configuration [MCR]

]# setkeycodes 0x61 121

[Keylock]

]# setkeycodes 0x63 123



### 2. How to Check Performance of the JavaPOS Device Service

This chapter describes the method to check performance of the JavaPOS Device Service, assuming that the operating environment described in Chapter 1 has been built up.

Here, the setup method is referred to as "PREPARE" and the operation method of the Device Health Check Program as "OPERATION".

In this chapter, the device health check method is explained for the following devices:

- LineDisplay
- Keylock
- CashDrawer
- MSR
- Scanner
- POSPrinter

#### PREPARE

Copy of Device Health Check Program modules

0	Device Health Check Program	D
D	Windows CashDrawer Driver	PDF
PDF	RelNote.pdf Adobe Acrobat Document 142 KB	
	ST-A10V0900.001	

Open the Kit CD, then copy the Device Health Check Program folder to a desired location in the local computer.

The subsequent procedures are separately explained for Windows and Linux below.



#### <Windows>

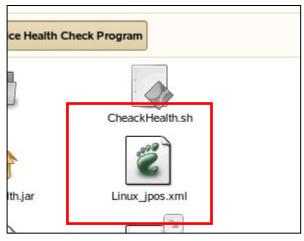
#### Change of setup file name



Change the name of the Windows\_jpos.xml file in the Device Health Check Program folder to jpos.xml.

<Linux>

1. Change of setup file name



Change the name of the Linux\_jpos.xml file in the Device Health Check Program folder to jpos.xml.



#### 2. Grant of execute authority

]# chmod 775 /home/tec/Device Health Check Program/CheackHealth.sh

Execute the above command to grant the CheackHealth.sh file an execute authority. (The above is an example when the Device Health Check Program folder has been copied to /home/tec.)

#### 3. Generation of CashDrawerJni link file

]# ln -s /home/tec/ Device Health Check Program/libTECCashDrawerJni.so.0.0 /home/tec/ Device Health Check Program/libTECCashDrawerJni.so

Execute the above command to generate link file. The above command is needed to use drawer driver.

#### 4. Generation of PKBFilterJNI link file

]# ln –s /home/tec/ Device Health Check Program/ libTECPKBFilterJNI.so.0.0 /home/tec/ Device Health Check Program/ libTECPKBFilterJNI.so

#### 5. Generation of TECUSBJNI link file

]# ln –s /home/tec/ Device Health Check Program/ libTECUSBJNI.so.0.0 /home/tec/ Device Health Check Program/ libTECUSBJNI.so

#### 6. Generation of TECUSB link file

]# ln –s /home/tec/ Device Health Check Program/ libtecusb.so.0.0 /home/tec/ Device Health Check Program/ libtecusb.so



#### Default value

Default value of major parameters is as follows. To change the default value, please refer to the chapter, "3. How to Use the JavaPOS Device Service" in the GUIDE or the setup method in the Application User Manual of each device service.

Category	LogicalName	deviceBus	PortName
Keylock	iButton	USB	-
Keylock	PKBST52	USB	-
MSR	MSRPKBST-52	USB	-
MSR	MCRST-A10	RS232	COM5
LineDisplay	LIUST-A10	RS232	COM4
LineDisplay	LIUST-53	RS232	COM4
LineDisplay	WD-111	RS232	COM4
Scanner	HS530RS	RS232	COM3
Scanner	HS530RSEx	RS232	COM3
CashDrawer	DRWST50	-	DRW1
CashDrawer	DRWST50Ex	-	DRW2
CashDrawer	TRSTA1S_CashDrawer	RS232	COM1
CashDrawer	TRSTA1P_CashDrawer	Parallel	LPT1
CashDrawer	TRSTA1L_CashDrawer	LAN	-
POSPrinter	TRSTA1S	RS232	COM1
POSPrinter	TRSTA1P	Parallel	LPT1
POSPrinter	TRSTA1L*x	LAN	-

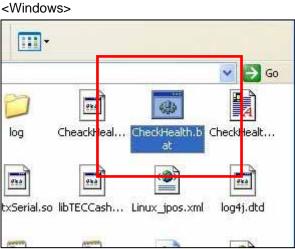
\*x=1 or 2 or 3

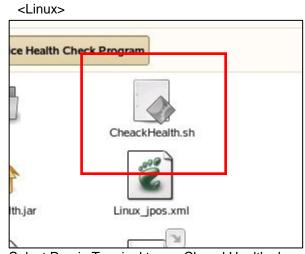
Category	LogicalName	deviceBus	vendorID	productID
CashDrawer	TRSTA1U_CashDrawer	USB	*1	*2
POSPrinter	TRSTA1U	USB	*1	*2

*1	TRST-A1x-U-QM: TRST-A1x-U-CN: TRST-A00-U:	61 70 82
*2	TRST-A00-U: Other:	2 1or2



### OPERATION

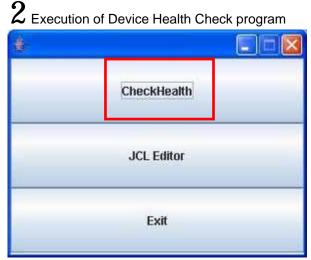




Execute the CheckHealth.bat file.

Select Run in Terminal to run CheackHealth.sh.

The subsequent OPERATION applies both to Windows and Linux.



Click on the [CheckHealth] button at the top.

\* Note the subsequent OPERATION differs for Keylock, Drawer, MSR, LineDisplay, and Scanner.
 \* The functions of the JCL Editor are not used this time.



## **iButton**

## A-1 Keylock panel display

+ POB Dentas Entry	Conf	ig			
+ Cris Dicplay	Calegory MSR LineDisplay CashDrawer CashDrawer	Logita/Hama MCRSTA18 LOGTA3 TRSTA19_CashDrawer TRSTA10_CashDrawer	Vender Todrieba TEC Corporation Todrieba TEC Corporation Todrieba TEC Corporation Todrieba TEC Corporation	Product Name TECMSR TECLINEDISISM TECCalinDraver TECCalinDraver	
Cash Draver	LineDrigtay MER CashOrawer Koylorik CashOrawer Neylorik Branner Moli	LAJOTAND MERTETST-TO DRWETSDEx Button DRWETSD PERSTS HISTOPHI MERTERST 4x	TODHILA TEC Corporative TODHILA TEC Corporative	TECLINEDISTRY TECMSP TECKSPORE TECKINDON TECKINOCH TECKINOCH TECKINOCH TECKINOCH TECKINOCH	
POS Patter	Scanner	HOSTORSEN	TOBHEA TEC Corporation	TECOLATIONES	Ŀ

Click on the [iButton] node under the [Keylock] node.

## A-2 Call to the Interactive Check Health method

PCG Device Entry	Keylock		
Line Dicatiay	Internal	External	Interactive
Cash Drawer Kinfock	Result:		-1.1
Pos Protector	CheckHealt	hText:	
Dianner			

Click on the [Interactive] button at the right.



## **iButton**

## $\operatorname{A-3}$ Installation of iButton

Keylock Interactive Che	ckHealth 🛛 🔀
Electronic Key Value Family Code Serial Number CRC Code	
ОК	NG

When the above window appears, make the iButton touch to the button contacts.

## A-4 Display of iButton data

Electronic Key Val	ue
Family Code	01
Serial Number	00000D532649
CRC Code	FA

The text boxes in the window show data stored in the iButton.

\* To exit, click on the [OK] or [NG] button.



## **iButton**

# $\mathrm{A}$ - $5\,$ Display of result



A value is displayed in the [Result] box and the [CheckHealthText] Text box. Either of the following two value combinations will be displayed in these boxes:

• When exited with th	e [OK] button in Step A-4.
Result	: SUCCESS
CheckHealthText	: Interactive Hcheck:Successful

When exited with the [NG] button in Step A-4.
 Result : SUCCESS
 CheckHealthText : Interactive Hcheck:Error



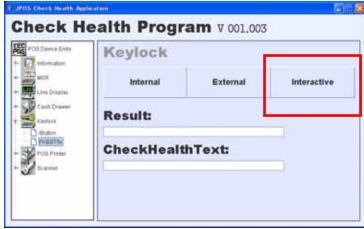
## Keylock

# $B{\ -}1$ Keylock panel display

POS Pros Device Entry	Conf	ig			
AGH     AGH     Cash Disper     Cash Disp	Calingany MDR CrestDrayer CashDrayer CashDrayer CashDrayer MGR CashDrayer Keylor CashDrayer Keylor CashDrayer Keylor ScantDrayer Keylor	Logith/fame MCRSTA18 ULUTATA3 TRISTA18_CastOrpwer TRISTA18_CastOrpwer ULUTA3 MORTFTSTA6 ORVITTA8 Bildton CONVITTA8 PriSBTTS MISSIONE MISSIONET 52	Tobrita TE C Approates Tobrita TE C Approates	Priduct Name TECLARS TECLARS Status TECCashOrawer TECCashOrawer TECCashOrawer TECCASS TECCAsOra TECCAsOra TECCAsOra TECCAsOra TECCASO TECCASO TECCASO TECCASO TECCASO TECCASO TECCASO	
FOS Patter	Geatrar	HOSTORIEX	TOBHEA TEC Corporation	TECOLANDERER	ŀ

Click on the [PKBST-52] node under the [Keylock] node.

# B-2 Call to the Interactive Check Health method



Click on the [Interactive] button at the right.



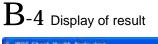
## Keylock

## B-3 Installation of keylock

ylock Interactive C		
KeyPosition Informa	tion	
Position Number	8	
PositionI Count	8	
ок	NG	
UN	NG	

When the above window appears, change key position.

To exit, click on the [OK] or [NG] button.





A value is displayed in the [Result] box and the [CheckHealthText] Text box. Either of the following two value combinations will be displayed in these boxes:

• When exited with the [OK] button in Step B-4.

Result	: SUCCESS
CheckHealthText	: Interactive Hcheck:Successful

• When exited with the [NG] button in Step B-4. Result : SUCCESS CheckHealthText : Interactive Hcheck:Error



# C-1 Drawer panel display

+ POB Device Entry	Conf	ig			
AGH     Line Display     Cash Display     TRETAIL CashDown     TRETAIL CashDown     DRWSTSDE     DOWSTSD     DOWSTSD     DOWSTSD     DAWSTSD     DAWSTSD     DAWSTSD	Californi MSR LineCrister CashOrawer CashOrawer LineCrister CashOrawer Keylorik CashOrawer Keylorik CashOrawer Keylorik CashOrawer Keylorik Di anter	Logitaria MCRETAIL LUCITAS TRETAILS_calibrater TRETAILS_calibrater LUCITAS MORTTES MORTTES Button DRWSTSOL PROSTS PRESS PRESS MORTS MORTS To MORTS To	Vender Toderska TEC Carporation Toderska TEC Carporation	Product Name TECLINSTRUM TECLINSTRUM TECCINSTRUM TECCINSTRUM TECCINSTRUM TECCINSTRUM TECCINSTRUM TECCINSTRUM TECCINSTRUM TECCINSTRUM TECCINSTRUM	
POG Porder	(Brather	HOSTOPSEN	TOBHBATEC Corporation	TECOLARCE	1.

Click on a node under the [CashDrawer] node. CashDrawer has the following eight kinds.

- DRWST50
- DRWST50Ex

• EPSON Drawer (TRSTA1xSDRW and TRSTA1xPDRW and TRSTA1xUDRW(QM or CN) and TRSTA1xLDRW and TRSTA00UDRW)

C-2 Call to the Interactive Check Health method



Click on the [Interactive] button at the right.



## CashDrawer

# C-3 Open the drawer

	Drawer Status
Drawer Open	CLOSED
OK	NG

Click on the [Drawer Open] button.

shDrawer Interactive CheckHeali	11
	Drawer Status
Drawer Open	OPEN
ОК	NG

The drawer opens and a message, "OPEN" is displayed on the [Drawer Status] box at the upper right in the window on the screen.

\* To exit, click on the [OK] or [NG] button.



## CashDrawer

# C-5 Display of result

A value is displayed in the [Result] box and the [CheckHealthText] Text box. Either of the following two value combinations will be displayed in these boxes:

• When exited with th	e [OK] button in Step C-4.
Result	: SUCCESS
CheckHealthText	: Interactive Hcheck:Successful

• When exited with the [NG] button in Step C-4. Result : SUCCESS CheckHealthText : Interactive Hcheck:Error

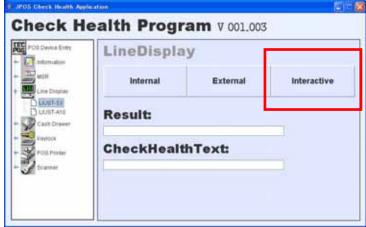


# LineDisplay

k Health	Program	▼ 001.003	1
Conf	-		
Calification MDR Calification Calification Calification Calification Calification Calification Calification Calification Netforta Netforta Net	LogitAtiama MCHSTA12 ULISTA13 THSTA12_CalifOrparet THSTA12_CalifOrparet THSTA12_CalifOrparet ULISTA13 MISHTF15 MISHTA15 MISHT51 MISHT5	Versier Todavida TEC Carporation Todavida TEC Carporation	Pridust Name TECLING TECLING TECLING TECCANDRAWE TECCANDRAWE TECCANDRAWE TECCANDRAWE TECCANDRAWE TECCANDRAWE TECCANDRAWE TECCANDRAWE TECCANDRAWE TECCANDRAWE

Click on the [LIUST-A10] node under the [LineDisplay] node.

D-2 Call to the Interactive Check Health method



Click on the [Interactive] button at the right.



# LineDisplay

## D-3 Display of LineDisplay

Indicates ANK Code	
Indicates Descriptors	012345678901234567

There are the following two CheckHealth functions for LineDisplay.

## Indicates ANK Code

Click on the [Indicates ANK Code] button at the upper left, and the same content, displayed in the two boxes on the right side of the button, is also displayed on the line display device.

## Indicates Descriptors

Click on the [Indicates Descriptors] button at the centre left, and a descriptor is displayed in the line display device at a location indicated by the value in the box on the right side of the button.



# D-4 Display of result

A value is displayed in the [Result] box and the [CheckHealthText] Text box. Either of the following two value combinations will be displayed in these boxes:

• When exited with the	e [OK] button in Step D-3.
Result	: SUCCESS
CheckHealthText	: Interactive Hcheck:Successful

When exited with the [NG] button in Step D-3.
 Result : SUCCESS
 CheckHealthText : Interactive Hcheck:Error



## MSR

# $E{\text{-}1}$ MSR panel display

POB Device Entry	Conf	ig		
MCH MARTYTET-TO MERTYSET-To Lone Creptiny Casin Drawer Rayoux	Calegory MDR UneChsplay CashDrawer CashDrawer UneChsplay MSR CashDrawer Keytorik CashDrawer Keytorik CashDrawer Keytorik CashDrawer Keytorik CashDrawer Keytorik	Logitatiana ACRESTASS LAUETAS TRESTALS_CARDEquest TRESTALS_CARDEquest LAUETAIS MORTYTET-TO ORWITTES Button ORWITTES PRESTS HISSORIS MORTORIS HISSORIS MORTORISTS	Vender Trösveika TEC Comparation Trösveika TEC Comparation	Product Name TECLARSH TECLARSHIP TECLARSHIP TECLARSHIP TECARSHIP TECARSHIP TECARSHIP TECCARDARSHIP TECCARDARSHIP TECCARDARSHIP TECCARDARSHIP TECCARDARSHIP
Pos Penter	Batter	HOLIOPIEN	TODHEA TEC Corporation	TECOLANDEREN

Click on the [MCRST-A10] node under the [MSR] node.

E-2 Call to the Interactive Check Health method



Click on the [Interactive] button at the right.



**MSR** 

SR Interar	tive Check	lealth	
Start	Result	Push start button	
	Track1	Can't Read Data now	
	Track2	Can't Read Data now	
	Track3	Can't Read Data now	
	Track4	Can't Read Data now	
		Fin	

Click on the [Start] button.

# E-4 Read of card data

R Interac	tive Checkl	lealth	Ŀ
Start	Result	Please swipe Card	
	Track1	Waiting	
	Track2	Waiting	
	Track3	Waiting	
	Track4	Waiting	
		Fin	

A message "Waiting" is displayed in the text boxes. Swipe a card.



## <Reading succeeded.>

	maria	and the second second	-11
Start	Result	Please swipe Card	
	Track1	ок	
	Track2	ок	
	Track3	ок	
	Track4	No Data	
		Fin	

To exit, click on the [Fin] button.

## <Reading failed.>

Start	Result	Please swipe Card	
	Track1	NG	
	Track2	NG	
	Track3	NG	
	Track4	NG	

# E-5 Display of result



Values displayed in the [Result:] and [CheckHealth] Text boxes differ depending on the reading result. There are the following two value combinations.



- When a reading operation in Step E-4 did not fail even once. Result : SUCCESS CheckHealthText : Interactive Hcheck:Successful
- When a reading operation in Step E-4 failed at least once.
   Result : SUCCESS
   CheckHealthText : Interactive Hcheck:Error



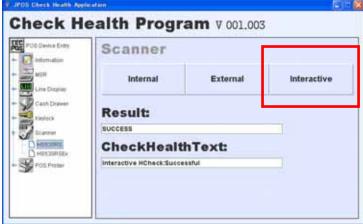
## Scanner

# F-1 Scanner panel display

POS Device Entry	onfi	ig			
MBB MC Line Depiley Can Cash Drawer Line Faylock Cash Startent Cash	eEnsplae InCrawer eDroplay B InCrawer dock InCrawer dock InCrawer facs some	Logitafiame ACRESTASS EVETASS EVETASS EVETASS EVETASS EVETAS EVETAS EVENTSS HORVETSS HORVETSS HORVETSS HORVETS HORVE	Viewier Trobella TEC Carporation Trobella TEC Carporation	Product Name TECLINSTRUM TECLINSTRUM TECCANDrawer TECCANDrawer TECCANDrawer TECCANDrawer TECCANDrawer TECCANDrawer TECCANDrawer TECCANDRAWER TECCANDRAWER TECCANDRAWER	
	enter.	HOLIOPSEX	TOBHBATEC Corporation	TECOLANDEREN	ţ

Click on the [HS530RS] node under the [Scanner] node.

# $F\mathchar`-2$ Call to the Interactive Check Health method



Click on the [Interactive] button at the right.



## Scanner

-3 Read of ba		Þ
Master		
ок	NG	
		1
		-
	End	

Read the bar code. The bar code read first time will be used as master data of the bar code. For subsequent bar code reading, it is judged OK if the bar code read is the same as that registered as master data. And, it is judged NG, if the bar code read is different from the master data.

#### <Same as master data>

icanner Interactive CheckHea	lth	×
Master F0008020010771	11 N-1018	
OK 1	NG 0	-01-
F3068320016771		*
A CLARING SHE WINDOW SHE WANTA		
		-
		-

To exit, click on the [End] button.

#### <Different from master data>

canner Interactive Che	ckHealth	×
Master F306832001 OK 1	6771 NG 1	
F3068320016771 F4901301231420		
	End	



## Scanner

# F-4 Display of result

" Scanner		
Internal	External	Interactive
Result:		
SUCCESS		1
CheckHeal	thText:	
Interactive HCheck:Suc	cessful	

Values displayed in the [Result:] and [CheckHealth] Text boxes differ depending on the reading result. There are the following two value combinations.

<ul> <li>When a reading op</li> </ul>	eration in Step F-3 did not fail even once.
Result	: SUCCESS
CheckHealthText	: Interactive Hcheck:Successful

 $\bullet$  When a reading operation in Step F-3 failed at least once.

Result: SUCCESSCheckHealthText: Interactive Hcheck:Error



## **POSPrinter**

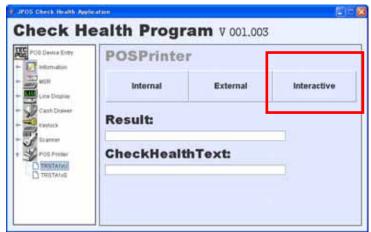
## G-1 POSPrinter panel display

POS Device Entry	Conf	ig			
- MOR	Category MSR LineDisplay CashDrawer CashDrawer	Logitafiame MCRET-A18 LKIST-A13 TRISTAT3_CashDrawer TRISTAT3_CashDrawer	Vender ToSHEA TEC Corporation TOSHEA TEC Corporation TOSHEA TEC Corporation TOSHEA TEC Corporation	Product Name TECMSR TECLINEDIState TECCalifOraver TECCalifOraver	
Cash Draver Earlock	CathQrawer LineDraplay MSR CashDrawer Reytora CashDrawer	LUGT-ATD MERTFTST-T6 DRWETSDEx Button DRWETSD	TODHEA TEC Corporation TODHEA TEC Corporation TODHEA TEC Corporation TODHEA TEC Corporation TODHEA TEC Corporation TODHEA TEC Corporation	TECLENDISEN TECMSR TECMSR TECCENDENNEES TECCENDENNE TECCENDENNE TECCENDENNE	
POS Printer	intervision Dicientine MDR Dicientine	PKBST54 340530R0 MDRPxB0T-54 340530R5Ex	TODHEA TEC Corporation TODHEA TEC Corporation TODHEA TEC Corporation TODHEA TEC Corporation	TECRANION TECREAMEN TECREAMEN TECREAMENES	

Click on a node under the [POSPrinter] node. POSPrinter has the following nine kinds.

- TRSTA1P
- TRSTAxS(TRSTA1S or TRSTA00S)
- TRSTAxU-x(TRSTA1U-QM, TRSTA1U-CN or TRSTA00U)
- TRSTA1Lx(TRSTA1L1, TRSTA1L2 or TRSTA1L3)

G-2 Call to the Interactive Check Health method



Click on the [Interactive] button at the right.

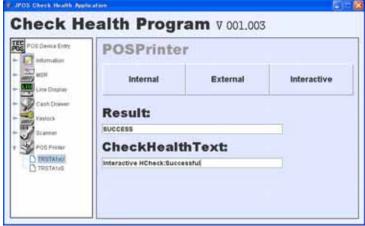


## **POSPrinter**

# G-3 Execute of Print

When the above window appears, execute each function. \* To exit, click on the [OK] or [NG] button.

## G-4 Display of result



A value is displayed in the [Result] box and the [CheckHealth] Text box. Either of the following two value combinations will be displayed in these boxes:

- When exited with the [OK] button in Step G-3.
   Result : SUCCESS
   CheckHealthText : Interactive Hcheck:Successful
- When exited with the [NG] button in Step G-3. Result : SUCCESS CheckHealthText : Interactive Hcheck:Error



## 3. How to Use the JavaPOS Device Service

This chapter describes the setups required to use the JavaPOS Device Service, assuming that the operating environment described in Chapter 1 has been built up.

#### **Required files**

In order to use the JavaPOS Device Service provided by the Kit, the following files are required besides the library file.

- jpos111.jar
- jpos.xml
- log4j.xml
- log4j.dtd
- log4j-1.2.12.jar
- commons-logging.jar
- RXTXcomm.jar
- swing-layout-1.0.3.jar
- xercesImpl.jar
- xml-apis.jar
- jcl\_editor.jar
- rxtxSerial.dll(\*1)
- librxtxSerial.so(\*2)
- rxtxParallel.dll(\*1)
- librxtxParallel.so(\*2)
- TECCashDrawerJni.dll(\*3)
- libTECCashDrawerJni.so(\*4)
- libTECiButtonJni.so(\*5)
- JimiProClasses.jar
- TECUSB.dll(\*6)
- libtecusb.so.0.0(\*7)
- TECUSBJNI.dll(\*6)
- libTECUSBJNI.so.0.0(\*7)
- TECUSBPM.exe(\*3)
- \*1 Required only for Windows
- \*2 Required only for Linux
- \*3 Required only for using CashDrawer Device Service under Windows
- \*4 Required only for using CashDrawer Device Service under Linux
- \*5 Required only for using iButton Device Service under Linux
- \*6 Required only for using POSPrinter USB Device Service under Windows
- \*7 Required only for using POSPrinter USB Device Service under Linux



## **Description of Files**

A destination to save a file may differ depending on the development environment. The following explanation is based on the development using the NetBeans5.5.

<u>Jpos111.jar</u>	
(Destination to save):	Any location
(Description):	JavaPOS Device Control. To be imported when creating an application.
(Available from):	JavaPOS-1.11.0-Dist.zip on the web site,
	http://www.javapos.com/samplecode.html, or
	http://www.javapos.com/index.html



jpos.xml (Destination to save):

(Description):

Root directory of project

A device setup file required to operate each Device Service. The following focuses on the major setup items descried in the file. This file is required for operating each Device Service.

\*1. Creation of jpos.xml file

An xml file is provided for each device in the JavaPOS folder. When using the xml files, compile all xml files into one file and name it "jpos.xml".

xml version="1.0" encoding="UTF-8"?
JposEntries PUBLIC "-//JavaPOS//DTD//EN" "jpos/res/jcl.dtd"
<jposentry logicalname="DefaultDisplay"> <creation <br="" factoryclass="jpos.toshibatec.linedisplay.loader.JavaPOSServiceFactory">serviceClass="jpos.toshibatec.linedisplay.services.LineDisplayService"/&gt; <vendor name="TOSHIBA TEC Corporation" url="http://www.toshibatec.co.jp"></vendor> <jpos category="LineDisplay" version="1.11"></jpos> <product <br="" description="TEC LUIST-51 Serial Line Display">name="TECLineDisplay" url="http://www.toshibatec.co.jp"/&gt;</product></creation></jposentry>
-!Other non JavaPOS required property (mostly vendor properties and bus specific properties i.e.
RS232 )>
<prop name="portName" type="String" value="COM4"></prop> """""""""""""""""""""""""""""""""""
<pre><pre>countryCode" type="String" value="3"/&gt;</pre></pre>
<pre><pre>cprop name="dataBits" type="String" value="8"/&gt; </pre></pre>
<prop name="parity" type="String" value="None"></prop> <prop name="modelName" type="String" value="LIUST-51"></prop>
<prop name="modelivanie_type=" string_value="Lios1-51"></prop> <prop name="flowControl" type="String" value="Xon/Xoff"></prop>
<pre><pre>cprop name= nowcontion type= string value= x0n/x0n //</pre></pre>
<prop name="deviceBus" type="String" value="RS232"></prop>
<prop name="baudRate" type="String" value="9600"></prop>
<jposentries></jposentries>
. //Descriptions for other devices

The following describes the major setup items. For details of the setup method, please refer to the Application User Manual of each JavaPOS Device Service.



- <JposEntry logicalName="DefaultDisplay" value="0"> A description to set a logical device name. Change the shaded area.
- <prop name="portName" type="String" value="COM4"/> A description to set COM ports of a device. Change the shaded area.
- <prop name="baudRate" type="String" value="9600"/> A description to set baud rate of a device. Change the shaded area.
- 2Defference in descriptions between Windows and Linux As for portName, "COMX" is used for Windows and "/dev/ttySX" is used for Linux. (X: serial port no.) Note that COMX starts from 1 while /dev/ttySX starts from 0.

[Windows]
value="COM1"
value="COM2"
value=

[Linux] value="dev/ttyS0"	
value="dev/ttyS1"	
value=	

## <u>log4j.xml</u>

(Destination to save): Root directory of project

(Description):

A setup file for a log to be output. To be copied in the directory where the execution file exists. The following focuses on the major setup items descried in the file. Please create your own file.

- <param name="file" value="log/ST-A20.log" /> A description to set a file name of log to be output.
- <priority value="info" />
   A description to set a log level.

Fatal:	Fatal error	error:	Error
warn:	Warning	info:	Information
debug:	Debug	trace:	Trace

Log4j.dtd

(Destination to save): Root directory of project

(Description):

A file to define XML tags. To be copied in the directory where the execution file exists. Please create your own file.



log4j-1.2.12.jar	
(Destination to save): (Description):	Any location A library file to output a log. As with JavaPOS DeviceService, this file must be imported in a project.
(Available from):	logging-log4j-1.2.12.zip on the web site, http://archive.apache.org/dist/logging/log4j/1.2.12/, or http://logging.apache.org/
commons-logging.jar (Destination to save): (Description): (Available from):	Any location A library file to output a log. To be imported when creating an application. commons-logging-1.0.4.zip on the web site, http://archive.apache.org/dist/commons/logging/binaries/, or http://commons.apache.org/logging/
<u>RXTXcomm.jar</u> (Destination to save): (Description):	Any location A library file to access a Device which uses a COM (component object model). To be imported when creating an application.
(Available from):	rxtx-2.1-7-bins-r2.zip on the web site, <u>http://rxtx.qbang.org/pub/rxtx/</u> , or <u>http://users.frii.com/jarvi/rxtx/download.html</u>
swing-layout-1.0.3.jar (Destination to save): (Description): (Available from):	Any location A library file to use swing. To be imported when creating an application. swing-layout-1.0.3.jar on the web site, http://java.sun.com/products/archive/jfc/1.0.3/index.html, or http://www.sun.com/
<u>xercesImpl.jar</u> (Destination to save): (Description):	Any location A library file to convert into text or other XML format. To be imported when creating an application.
(Available from):	Xerces-J-bin.2.9.0.zip on the web site, http://apache.adcserver.com.ar/xml/xerces-j/, or http://xerces.apache.org/



<u>xml-apis.jar</u> (Destination to save): (Description): (Available from):	Any location A library file to convert into text or other XML format. To be imported when creating an application. Xerces-J-bin.2.9.0.zip on the web site, <u>http://apache.adcserver.com.ar/xml/xerces-j/</u> , or <u>http://xerces.apache.org/</u>
<u>JposEntryEditor.jar</u> (Destination to save): (Description): (Available from):	Any location A library file to access an XML file. To be imported when creating an application. jcl2.2.0.zip on the web site, http://Availablefromforge.net/project/showfiles.php?group_id=128804&packag e_id=141062&release_id=306139, or http://jposloader.Available fromforge.net/downloads/?S=A
<u>JimiProClasses.jar</u> (Destination to save): (Description): (Available from):	Any location A library file to access an image file. To be imported when creating an application. jimi1_0.zip on the web site, http://java.sun.com/products/jimi/
<u>rxtxSerial.dll</u> (Destination to save): (Description): (Available from):	Root directory of project A library file to access a serial port under Windows. rxtx-2.1-7-bins-r2.zip on the web site, <u>http://rxtx.qbang.org/pub/rxtx/</u> , or <u>http://users.frii.com/jarvi/rxtx/download.html</u>
librxtxSerial.so (Destination to save): (Description): (Available from):	Root directory of project A library file used to access a serial port under Linux. rxtx-2.1-7-bins-r2.zip on the web site, <u>http://rxtx.qbang.org/pub/rxtx/</u> , or <u>http://users.frii.com/jarvi/rxtx/download.html</u>



rxtxParallel.dll	
(Destination to save):	Root directory of project
(Description):	A library file used to access a parallel port under Windows.
	This is a RXTX parallel library customized by TTEC.
	It is based on rxtx-2.1-7(LGPL). "rxtxSerial.dll" file is necessary to use this
(Available from):	library. This Kit. Click on "TEC RXTX Parallel Library Source".
	It is bundled with rxtxParallel.zip.
librxtxParallel.so	
(Destination to save):	Root directory of project
(Description):	A library file used to access a parallel port under Linux.
	This is a RXTX parallel library customized by TTEC. It is based on rxtx-2.1-7(LGPL). "librxtxSerial.so" file is necessary to use this
	library.
(Available from):	This Kit. Click on "TEC RXTX Parallel Library Source".
	It is bundled with rxtxParallel.zip.
TECCashDrawerJni.dll	
(Destination to save):	C:¥Windows¥system32 or C:¥WINNT¥system32
(Description):	An application programming interface (API) to be used to access the Windows
	CashDrawer driver from Java.
(Available from):	This Kit. Click on "Driver" $\rightarrow$ "Cash Drawer Driver" $\rightarrow$ "Windows".
libTECCashDrawerJni.s	<u>so.0.0</u>
(Destination to save):	Root directory of project
(Description):	An application programming interface (API) to be used to access the Linux. Make a link file called libTECCashDrawerJni.so and use it. Ex : ]# In –s libTECCashDrawerJni.so.0.0 libTECCashDrawerJni.so
(Available from):	This Kit. Click on "Driver" $\rightarrow$ "Cash Drawer Driver" $\rightarrow$ "Linux" $\rightarrow$ "JNI".
libTECiButtonJni.so	
(Destination to save):	Root directory of project
(Description):	An application programming interface (API) to be used to access the Linux
(Available from):	iButton driver from Java. This Kit. Click on "Driver" $\rightarrow$ "Linux iButton Driver" $\rightarrow$ "JNI".



TECUSB.dll (Destination to save): (Description): (Available from):	C:¥Windows¥system32 or C:¥WINNT¥system32 Library of TECUSB driver for Windows. This Kit. Click on "Driver" $\rightarrow$ "TECUSB Driver" $\rightarrow$ "Windows".
<u>LogMngr.dll</u> (Destination to save): (Description): (Available from):	C:¥Windows¥system32 or C:¥WINNT¥system32 Library of TECUSB driver for Windows. This Kit. Click on "Driver" $\rightarrow$ "TECUSB Driver" $\rightarrow$ "Windows".
libtecusb.so.0.0 (Destination to save): (Description): (Available from):	Root directory of project Library of TECUSB driver for Linux. Make a link file called libtecusb.so and use it. Ex : ]# In –s libtecusb.so.0.0 libtecusb.so This Kit. Click on "Driver" → "TECUSB Driver" → "Linux" → "Driver".
TECUSBPM.exe (Destination to save): (Description): (Available from):	C:¥Windows¥system32 or C:¥WINNT¥system32 In case of Vista, root directory of project An USB power management process for Windows. It is nesessary to use TRST-A1x-U on Windows This Kit. Click on"Driver" $\rightarrow$ "TECUSB Driver" $\rightarrow$ "Windows".
<u>TECUSBJNI.dll</u> (Destination to save): (Description): (Available from):	C:¥Windows¥system32 or C:¥WINNT¥system32 An application programming interface (API) to be used to access the Windows TECUSB driver from Java. This Kit. Click on "Driver" $\rightarrow$ "TECUSB Driver" $\rightarrow$ "Windows".
<u>libTECUSBJNI.so.0.0</u> (Destination to save): (Description): (Available from):	Root directory of project An application programming interface (API) to be used to access the Linux. Make a link file called libTECUSBJNI.so and use it. Ex : ]# In –s libTECUSBJNI.so.0.0 libTECUSBJNI.so This Kit. Click on"Driver" → "TECUSB Driver" → "Linux" → "JNI".



## libTECPKBFilterJNI.so.0.0

(Destination to save):	Root directory of project
(Description):	An application programming interface (API) to be used to access the
	Linux.
	Make a link file called libTECPKBFilterJNI.so and use it.
	Ex : ]# In -s libTECUSBJNI.so.0.0 libTECUSBJNI.so
(Available from):	This Kit. Click on "Driver" $\rightarrow$ "Keyboard Driver" $\rightarrow$ "Linux" $\rightarrow$ "JNI".



## Example of Creating An Application Using the JavaPOS Device Service

This chapter describes the method to create an application using the JavaPOS Device Service. For this purpose, the demo program enclosed in the Kit is used.

## **Coding Process**

There are the following processes to create an application using the JavaPOS Device Service.

- 1. Create a device class object.
- 2. Enable a device.
- 3. Call to a device-specific method and get a property.
- 4. Disable a device.

The subsequent sections explain each process.

#### 1. Create a device class object.

· · · · · · · · · · · · · · · · · · ·
<pre>import javax.swing.DefaultListModel; import jpos.*;</pre>
import jpos.events.*;
1 11
import java.util.*;
public class DrawerPanel extends javax.swing.JPanel implements StatusUpdateListener, DirectIOListener{
private CashDrawer drawer;
/** Creates new form DrawerPanel */
<pre>public DrawerPanel() {</pre>
initComponents();
drawer = new CashDrawer();
}

- Import Jpos.
- Specify a variable for CashDrawer type (for example, drawer).
- By specifying "new", create a Device Control object.
  - To receive events, implement events which are defined by each device. (For details of implementation, refer to "Supplemental Explanation 2 Receipt of events".)



#### 2. Enable a device.

drawer.open(LOGICALNAME); drawer.claim(100); drawer.setDeviceEnabled(true);

- Open a device by specifying a device logical name.
- Set a timeout and perform an exclusive process (\*).
- Set the DeviceEnabled property to TRUE.
- \*1 Specify a logical device name for LOGICALNAME.
- \*2 For devices which do not perform an exclusive processing, no claim handlings are required.
- \*3 For the claim handling, specify a timeout period in milliseconds in the round brackets as argument.

#### 3. Call to a device-specific method and get a property.

Bool status; drawer.openDrawer(); status = drawer.getOpened();

- Define a variable to get a status.
- By calling to the CashDrawer-specific method, a cash drawer opens.
- A status is obtained.
  - \*1 Device-specific methods differ for each device. For details, please refer to the Application User Manual of each Device Service.
  - \*2 Get method and Set method differ for each property. The example below explains the method to get YYY property in XXX object and set ZZZ value to the property.

Get: a=XXX.getYYY(); Set: XXX.setYYY (ZZZ);



#### 4. Disable a device.

drawer.setDeviceEnabled(flase); drawer.release(); drawer.close();

- Set the DeviceEnabled property to FALSE.
- Release the exclusive processing.
- Close the device.
- \*1 For the devices which do not perform an exclusive processing, the release processing is not required.
- \*2 When the device is closed, an operation speed increases. Please close the device when exiting from the application, as much as possible. To suspend the device, set the DeviceEnabled property to FALSE.

#### Supplemental Explanation 1. Method to set an exception

try	
{	
	//A process which uses JavaPOS Device Control
	drawer.open(LOGICALNAME);
	drawer.claim(100);
	drawer.setDeviceEnabled(true);
catch	n(JposException e)
{	
	//Describe an exception here.
	•

Perform an exception for all cases where the JavaPOS Device Control is used, for example, device open, claim, call to device-specific method, property handling.

In detail, describe a code which uses JavaPOS Device Control in braces "{}" and a handling when an exception occurs in braces of catch(JposException){}. JposException is an exception which is thrown when an exception occurs with the JavaPOS Device Control.



#### Supplemental Explanation 2. Receipt of events

```
public void statusUpdateOccurred(StatusUpdateEvent e){
     //Describe a handling here.
}
public void directIOOccurred(DirectIOEvent e){
     //Describe a handling here.
}
```

- Implement an event interface in a main class. (Please refer to "1. Create a device class object".)
- Implement functions which should be called when an event occurs.
- Add an appropriate description in the shaded areas.

```
*1 Event types differ for each device. For details, please refer to the Application User Manual of each device.
```



## **Creation of Window**

The figure below shows an example of window created when an application is created following the above-mentioned processes.

POS Test
POS TEST for Java POS
POS Device Entry     Drawer osversion:     Description:       Information     Device Name:     DRWST50Ex     RUN     2     STOP
<ul> <li>Line Display</li> <li>Cash Drawer</li> <li>TRSTA1S_CashDrawer</li> <li>TRSTA1U_CashDrawer</li> <li>DrawerOpen</li> </ul>
DRWST50     Keylock     Scanner     POS Printer

#### 1. Device Name (Logical Name) text box

A text box which is used to set a logical device name of a device to open the device.

#### 2. Device enable button

Performs a process required to enable the device. Specifically, performs Open and Claim, then set the DeviceEnabled property to TRUE.

#### 3. Device disable button

Performs a process required to disable the device. Specifically, set the DeviceEnabled property to FALSE, and then performs Release and Close.



#### 4. Calls to a method to operate a device

Calls to a device-specific device to operate a device. This example calls a drawerOpened property to explain the case of CashDrawer.

5. Property status label

A label is to display property information. This example displays the drawerOpened property to explain the case of CashDrawer.