

# **NCR Retail**

Release 3.7

## **OPOS User's Guide**



B005-0000-1619

Issue D

---

The product described in this book is a licensed product of NCR Corporation.

NCR is a registered trademark of NCR Corporation.

NCR RealPOS, NCR RealPrice, NCR RealScan, NCR EasyPoint and NCR FastLane are either registered trademarks or trademarks of NCR Corporation in the United States and/or other countries.

It is the policy of NCR Corporation (NCR) to improve products as new technology, components, software, and firmware become available. NCR, therefore, reserves the right to change specifications without prior notice.

All features, functions, and operations described herein may not be marketed by NCR in all parts of the world. In some instances, photographs are of equipment prototypes. Therefore, before using this document, consult with your NCR representative or NCR office for information that is applicable and current.

To maintain the quality of our publications, we need your comments on the accuracy, clarity, organization, and value of this book.

Address correspondence to:

Manager, Information Products  
NCR Corporation  
2651 Satellite Blvd.  
Duluth, GA 30096

Copyright © 2004  
By NCR Corporation  
Dayton, Ohio U.S.A.  
All Rights Reserved

# Preface

## Audience

This book is written for hardware installer/service personnel, system integrators, and field engineers.

**Notice:** This document is NCR proprietary information and is not to be disclosed or reproduced without consent.

## References

- NCR Retail Systems Manager Software User's Guide  
(B005-0000-1518)

# Table of Contents

## OPOS for Windows

Additional Software Requirements .....	1-4
Using Retail Systems Manager LE for Windows.....	1-4
Creating a New Profile .....	1-5
Cash Drawer - Attached to Printer .....	1-8
Cash Drawer Configuration Entries.....	1-8
CashDrawer Data Capture Configuration Entries.....	1-10
Cash Drawer - Attached to POS Terminal.....	1-12
Cash Drawer Configuration .....	1-12
CashDrawer Configuration Entries.....	1-13
CashDrawer Data Capture Configuration Entries.....	1-14
Hard Totals.....	1-16
Hard Totals Configuration Entries.....	1-16
HardTotals Data Capture Configuration Entries .....	1-17
Keylock .....	1-18
Keylock Configuration Entries.....	1-18
Keylock Data Capture Configuration Entries.....	1-19
Line Display .....	1-20
NCR5972LineDisplay .....	1-20
Line Display Configuration Entries .....	1-20
NCR5973LineDisplay .....	1-22
Line Display Configuration Entries .....	1-22
NCR59752x20LineDisplay .....	1-23
Line Display Configuration Entries .....	1-23
NCR5975IntLineDisplay .....	1-24
Line Display Configuration Entries .....	1-24
NCR7402LineDisplayAPA .....	1-25

Line Display Configuration Entries .....	1-25
NCR7402LineDisplay2x20 .....	1-27
Line Display Configuration Entries .....	1-27
NCR7443/7446LineDisplay2x20.....	1-29
Line Display Configuration Entries .....	1-29
LineDisplay Data Capture Configuration Entries.....	1-30
MICR .....	1-32
DirectIO - MICR Parsing .....	1-32
DirectIO - Set Remove Non Digits.....	1-39
Clear Exception Table.....	1-40
MICR - Magnetic Ink Character Recognition Configuration Entries .....	1-41
MICR Data Capture Configuration Entries.....	1-42
MotionSensor .....	1-43
MotionSensor - Motion Sensor Configuration Entries .....	1-43
Motion Sensor Data Capture Configuration Entries.....	1-44
MSR .....	1-45
MSR - Magnetic Stripe Reader Configuration Entries.....	1-45
MSR Data Capture Configuration Entries.....	1-46
POS Printer .....	1-47
DirectIO - Raw Output.....	1-48
DirectIO - Set Barcode Width.....	1-49
DirectIO - Set Bitmap Type.....	1-50
DirectIO - Sound Audible Tone .....	1-51
DirectIO - Set Minimum Number of Characters Per Line.....	1-52
DirectIO - Get the Current Printer Status .....	1-54
DirectIO - Get Kiosk Specific Sensor Status.....	1-56
DirectIO - Set Autoeject Timeout .....	1-57
DirectIO - Eject Receipt.....	1-58
DirectIO - Get Kiosk Specific Status .....	1-59
DirectIO - Change Paper Low Reporting.....	1-60

---

DirectIO - Set Black Mark Configuration.....	1-61
DirectIO - Set Minimum Receipt Length .....	1-62
DirectIO - Set Receipt Retain Length.....	1-64
POS Printer Configuration Entries .....	1-66
Kiosk Specific Configuration Entries .....	1-70
POS Printer Data Capture Configuration Entries .....	1-72
Two-Sided Thermal Printing.....	1-73
OPOS / JavaPOS Configuration for Selection of Printer- Defined Modes.....	1-74
Escape Sequences to Control the Two-Sided Functionality within a Mode.....	1-80
Special Escape Sequences to Switch between Modes.....	1-83
Select Print Side Sequence .....	1-84
OPOS / JavaPOS Mode Functional Definition.....	1-85
Scale.....	1-87
DirectIO - Scale Status .....	1-87
DirectIO - Scale Read ROM .....	1-88
DirectIO - Scale ROM Version.....	1-89
DirectIO - Scale Live Weight .....	1-90
DirectIO - Scale Direct Access .....	1-92
Method Updates.....	1-93
ReadWeight Method <i>Updated in UPOS Release 1.9</i> .....	1-93
zeroScale Method <i>Updated in UPOS Release 1.9</i> .....	1-95
Scale Configuration Entries .....	1-96
Scale Data Capture Configuration Entries .....	1-97
Scanner.....	1-99
DirectIO - Scanner NOT-ON-FILE .....	1-99
DirectIO - Scanner Tone Control .....	1-100
DirectIO - Scanner Reset .....	1-101
DirectIO - Scanner Status.....	1-102
DirectIO - Scanner Read ROM .....	1-103

DirectIO - Scanner ROM Version .....	1-104
DirectIO - Scanner Pacesetter Options.....	1-105
DirectIO - Scanner Direct Access .....	1-106
Scanner Configuration Entries .....	1-107
Scanner Data Capture Configuration Entries .....	1-116
Scanner Frequently Asked Questions .....	1-117
7837-3000 and 7838 Scanner setup for UPOS.....	1-117
Money Center Integrated Self Service Scanner .....	1-118
7401/7402 Kiosk Integrated Scanner .....	1-118
HID USB Scanner Configuration.....	1-118
[Scanner Configuration Program Sequences]:.....	1-119
[Scanner Configuration Bytes]:.....	1-119
NCR 2357-xxxx( 3800, 4600, 5600 and 5620) Scanner setup for UPOS .....	1-123
ToneIndicator.....	1-125
DirectIO - Set Keyclick Tone.....	1-125
ToneIndicator - Tone Indicator Configuration Entries.....	1-126
Tone Indicator Data Capture Configuration Entries .....	1-127



## Revision Record

Issue	Date	Remarks
A	Dec 2005	Updates to USB Scale.
B	Apr 2006	Updates for Retail Controls 3.6 scanner/scale
C	May 2006	Updates to POS Printer for Two-Sided Thermal Printing
D	Aug 2006	Updates to Scanner, Cashdrawer, and POS Printer



# OPOS for Windows

---

The following table shows the devices that are supported for this version of the Retail Controls for Windows.

UnifiedPOS Control	NCR Support	Devices Supported
Bump Bar	NO	
Cash Changer	NO	
Cash Drawer - On Printer	YES	
Cash Drawer - Terminal	YES	
		NCR 7167, NCR7197 Printers
		NCR 7402/7443/7446/7456/7457/7458.
		NCR 7452/53-3xxx
CAT - Credit Authorization Term	NO	
Check Scanner	NO	
Coin Dispenser	NO	
Fiscal Printer	NO	
Hard Totals	YES	
Keylock	YES	Disk Based Media
		NCR 5932 USB Keyboard or 5952 USB Dynakey
Line Display	YES	NCR 5972 VFD, LCD, and Occular LCD. Serial only for all models. NCR 7402 APA, NCR 7402 2x20. NCR 7443/7446 2x20
MICR	YES	Connected to Printer
Motion Sensor	YES	NCR 7401 and 7402
MSR	YES	NCR 5932 USB Keyboard or 5952 USB Dynakey
PIN Pad	NO	
Point Card Reader Writer	NO	
POS Keyboard	NO	
POS Power	NO	

POS Printer	YES	NCR 7167, NCR 7197, NCR 7168
Remote Order Display	NO	
Scale	YES	NCR 7872, NCR 7875, 7876, 7883 (Serial, USB)
Scanner	YES	NCR 7872/75/76 (Serial, USB), NCR 7837 (Serial, NCR USB), NCR 7838 (Serial, NCR USB) 3800, 4600, 5600/20 (Serial, NCR USB)
Note1: USB – OS supported USB		NCR 7880, NCR 7882 (Serial, NCR USB)
Note2: NCR USB – USB through Virtual Serial COM port emulation drivers. Requires additional driver install.		NCR 7883, NCR 7892 (Serial, USB) NCR 7832 (Serial)
Signature Capture	NO	
Tone Indicator	YES	NCR 5932 USB Keyboard or 5952 USB Dynakey

NCR uses standard JavaPOS 1.7 Device Controls as provided by the JavaPOS committee (they are open source). Similarly, NCR OPOS Service Objects are compatible with the Common Controls for OPOS 1.7. The OPOS and JavaPOS specifications have now been combined into one specification called UPOS (Unified Point of Service). The latest UPOS specification can be found at: <http://www.nrf-arts.org/> then select **UnifiedPOS** from the frame on the left.

Additional information on JavaPOS can be obtained from: <http://www.javapos.com> and/or <http://www.javapos.com/samplecode.html>

This document only describes the NCR specific differences from the UnifiedPOS specification and is to be used in conjunction with that specification.

The term Retail controls is defined as an overall term which includes both OPOS and JavaPOS for Linux or Windows. The following table shows the relationship of some of the terms between OPOS and JavaPOS.

<b>Retail Controls</b>
------------------------

<b>JavaPOS</b>	<b>OPOS</b>
Device Control	Control Object
Device Service	Service Object

## Additional Software Requirements

### Using Retail Systems Manager LE for Windows

The Retail Systems Manager LE permits you to control power-up functionality and report some of the system parameters.

Note: Make all changes to a peripherals configuration using the Retail Systems Manager interface. If you change a peripheral's configuration directly in the registry, the new value may not be recognized.

Refer to the NCR Retail Systems Manager Software User's Guide (B005-0000-1518) for additional information on using Retail Systems Manager LE.

## Creating a New Profile

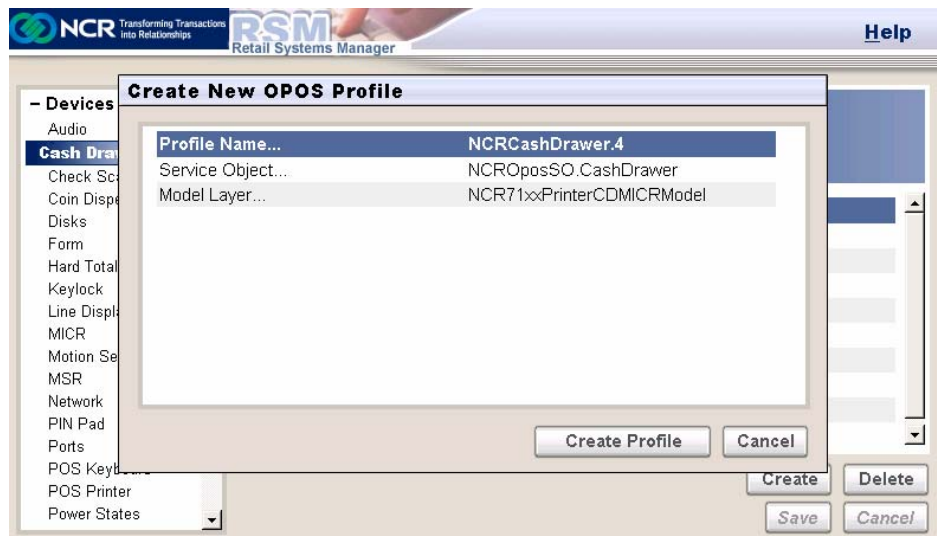
If you wish to add a new profile for a particular Retail Control, Select the type of device type on the left side of the screen and then select the “Create” button.

The screenshot shows the NCR Retail Systems Manager (RSM) interface. On the left, a tree view under 'Devices' has 'Cash Drawer' selected. The main window is titled 'Cash Drawer' and has two tabs: 'OPOS' and 'Diagnostics'. The 'OPOS' tab is active, displaying a configuration table for a profile named 'NCRCashDrawer.3'.

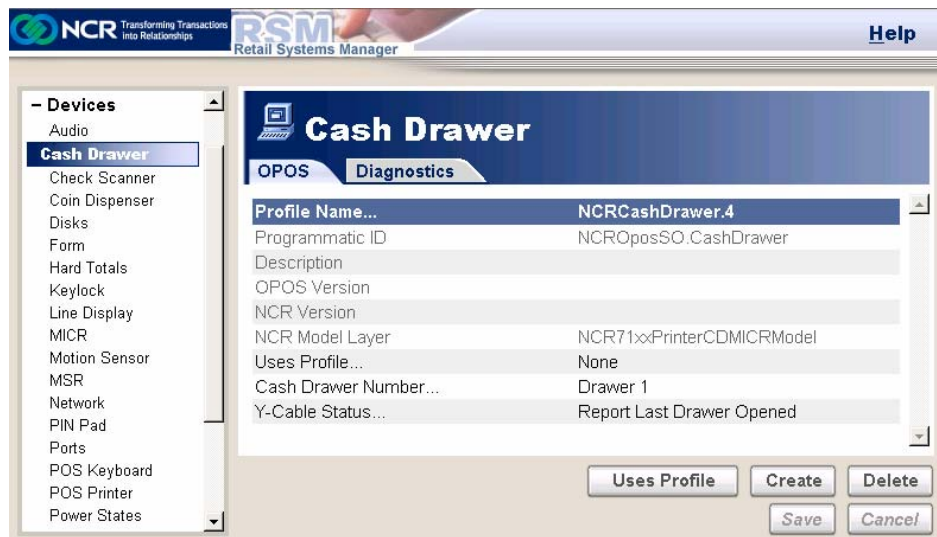
Profile Name...	NCRCashDrawer.3
Programmatic ID	NCROposSO.CashDrawer
Description	
OPOS Version	
NCR Version	
NCR Model Layer	NCRCashDrawerProcessor
Cash Drawer Number...	Drawer 1
Y-Cable Status...	Report Last Drawer Opened
Connection Type	Motherboard
Model	Integrated

At the bottom right of the configuration area are four buttons: 'Create', 'Delete', 'Save', and 'Cancel'.

Enter the Profile name that identifies this particular configuration. Select the Service Object line and then select the Service Object for the Retail Control being used. Other entries may appear depending on the Service Object selected, such as the Model Layer parameter shown in the following example.

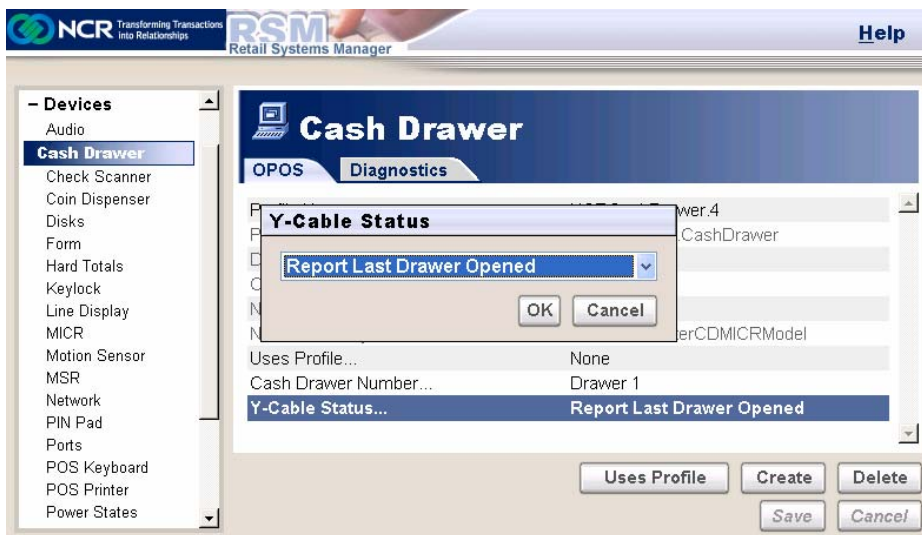


After these parameters have been entered, select “Create Profile”. The specific values for the profile with default values assigned is shown.





If you wish to change a particular parameter, select that parameter and you are provided with the configuration values to select. For example, if you wish to change the reporting of the method of the Y-Cable Status shown above, select “Y-Cable Status” and then select a valid value from the drop down list. After selecting new parameter values, select the Save button at the bottom of the screen. For additional information on using the Retail Systems Manager (RSM), refer to the *NCR Retail Systems Manager Software User’s Guide (B005-0000-1518)*.

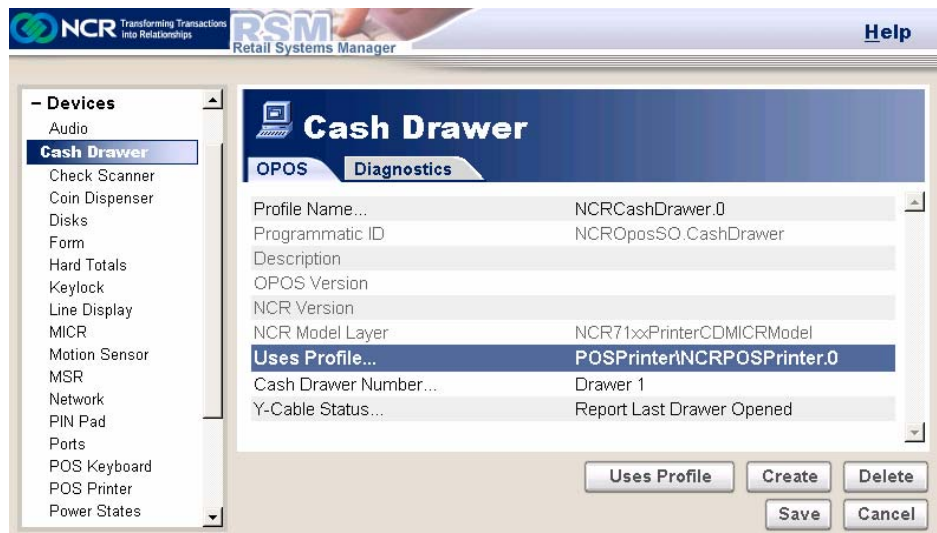


## Cash Drawer - Attached to Printer

The CashDrawer supports 1 or 2 cash drawers connected to the following printers:

- NCR 7167 Printer.
- NCR 7197 Printer.

## Cash Drawer Configuration Entries



Parameter	Description	Valid Values
Profile Name	The name of a specific configuration.	A configuration name such as NCRCashDrawer.0, (Any name you wish)
Programmatic ID	The Service Object being used by this control.	NCR OposSO.CashDrawer
NCR Model Layer	The cash drawer(s) are connected to a printer. Therefore, the printer must also be available. This is the Device Service for the printer.	NCR71xxPrinterCDMICRModel
Uses Profile	The printer profile that manages this drawer . It uses the configuration parameters of the Printer in addition to the parameters described here. Therefore, the Printer configuration entries must be set up first. If you delete the Printer service object, the cash drawer can not be used.	POSPrinter\NCRPOSPrinter.0  If the Printer profile has not been setup, this field shows "None."
Cash Drawer Number	Number of the cash drawer to be controlled.	Drawer 1 (default) or Drawer 2
YCableStatus	This setting determines how the cash drawer status is reported back to the application. This is intended to permit applications to work around the hardware limitation caused by the use of a Y-Cable in order to support two cash drawers. Most of the settings only have an effect when used with printers that support dual cash drawers with a Y-Cable. The following are the values for this parameter:  <b>Report Drawer 1 Only.</b> Backwards Compatibility = Reports the cash drawer status as all previous releases of this OPOS Service Object. No matter which drawer is open, the status is always reported as drawer 1.  <b>Report Last Drawer Opened</b> - In this case when a drawer open is detected, the open status is reported for the last drawer which was sent an Open command. If the Open is sent to drawer 1 then drawer 1 shows open, if to drawer 2 then drawer 2 shows open. If drawer 1 or 2 is already open, and an Open is sent to the other drawer, then both drawers report open. Both drawers remain reporting open until both drawers are closed, because we cannot detect	Report Drawer 1 Only Report Last Drawer Opened (default) Report Each Drawer Don't Report Status

Parameter	Description	Valid Values
	close until both are closed.	
	<b>Report Each Drawer</b> - This basically reports exactly what the hardware tells us. If either drawer 1 or drawer 2 is open, then both drawers are reported open. If drawer 1 is open then both 1 and 2 report open. If drawer 2 is open then both 1 and 2 report open. Both are reported open until both drawers are closed.	
	<b>Don't Report Status</b> - This affects all cash drawer types. In this mode the CapStatus capability reports FALSE, and both drawers always show closed regardless of the actual drawer status.	

## CashDrawer Data Capture Configuration Entries

Data capture for the CashDrawer is controlled through settings stored in the NCRDatacap.conf configuration file. The following example shows how to enable maximum data capture for all of the CashDrawer components.

```
[NCROposSO.CashDrawer]
```

```
LevelMask = 0xFFFFFFFF
```

```
TraceMask = 0xFFFFFFFF
```

```
UseDefault = F
```

```
[NCR71xxCashDrawerModel(0)]
```

```
LevelMask = 0xFFFFFFFF
```

```
TraceMask = 0xFFFFFFFF
```

```
UseDefault = F
```

```
[NCR71xxCashDrawerModel(1)]
```

LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F

[NCR71xxPrinterCDMICRModel]

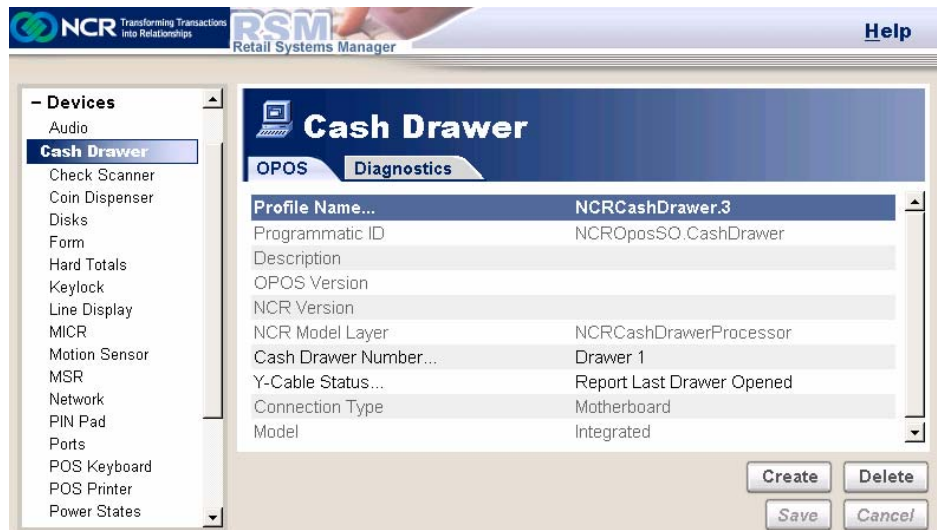
LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F

## Cash Drawer - Attached to POS Terminal

The CashDrawer supports 1 or 2 cash drawers connected to the following terminals.

- NCR 7453 - 3xxx
- NCR 7456
- NCR 7457
- NCR 7458
- NCR 7402
- NCR 7443
- NCR 7446
- NCR 7449

## Cash Drawer Configuration



## CashDrawer Configuration Entries

Parameter	Description	Valid Values
Profile Name	The name of a specific configuration.	A configuration name such as NCRCashDrawer.3 (Any name you wish)
Programmatic ID (OPSOS)	The Service Object being used by this control.	NCROposSO.CashDrawer
serviceClass (JavaPOS)		Or com.ncr.retail.jpos.services.cashdrawer.CashDrawerService
NCR Model Layer	The cash drawer(s) are connected to the terminal.	NCRCashDrawerProcessor
ConnectionType	Type of cashdrawer interface to be used. M – Motherboard (integrated).	M (default) 7443 744321xx 7446 7449
Model	Type of cashdrawer.	INTEGRATED (default)
nDeviceNumber	Number of the cash drawer to be controlled.	Drawer 0 (default) or Drawer 1
Uses Profile	The cashdrawer profile that manages this drawer . It uses the configuration parameters of the managing cashdrawer in addition to the parameters described here. Therefore, the managing cashdrawer configuration entries must be set up first. If you delete the managing cashdrawer service object, the cash drawer can not be used.	POSPrinter\NCRPOSPrinter.0  If the Printer profile has not been setup, this field shows “None.”
YCableStatus	This setting determines how the cash drawer status is reported back to the application. This is intended to permit applications to work around the hardware limitation caused by the use of a Y-Cable in order to support two cash drawers. Most of the settings only have an effect when used with printers that support dual cash drawers with a Y-Cable. The following are the values for this parameter:  <b>Report Drawer 1 Only.</b> Backwards Compatibility = Reports the cash drawer status as all previous releases of this OPOS Service Object. No matter which drawer is open, the	Report Drawer 1 Only Report Last Drawer Opened (default) Report Each Drawer Don't Report Status

Parameter	Description	Valid Values
	status is always reported as drawer 1.	
	<b>Report Last Drawer Opened</b> - In this case when a drawer open is detected, the open status is reported for the last drawer which was sent an Open command. If the Open is sent to drawer 1 then drawer 1 shows open, if to drawer 2 then drawer 2 shows open. If drawer 1 or 2 is already open, and an Open is sent to the other drawer, then both drawers report open. Both drawers remain reporting open until both drawers are closed, because we cannot detect close until both are closed.	
	<b>Report Each Drawer</b> - This basically reports exactly what the hardware tells us. If either drawer 1 or drawer 2 is open, then both drawers are reported open. If drawer 1 is open then both 1 and 2 report open. If drawer 2 is open then both 1 and 2 report open. Both are reported open until both drawers are closed.	
	<b>Don't Report Status</b> - This affects all cash drawer types. In this mode the CapStatus capability reports FALSE, and both drawers always show closed regardless of the actual drawer status.	

## CashDrawer Data Capture Configuration Entries

Data capture for the Terminal CashDrawer is controlled through settings stored in the NCRDatacap.conf configuration file. The following example shows how to enable maximum data capture for all of the CashDrawer components.

```
[NCROposSO.Cashdrawer]
```



LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F

[NCRCashDrawerModel]

LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F

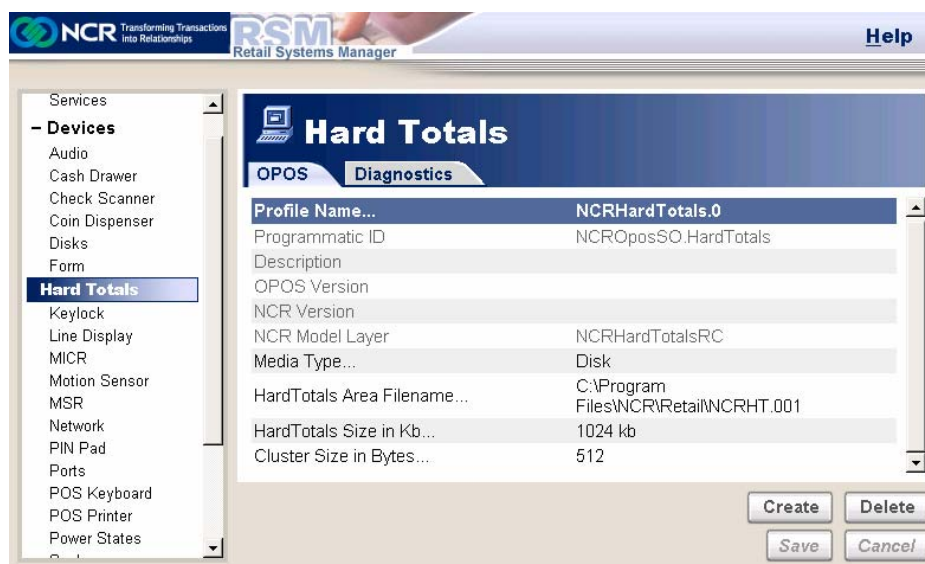
[NCRCashDrawerHydra]

LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F

## Hard Totals

Hard Totals provide a method to store various totals information obtained from the retail application. These totals can be stored on Disk or any device that supports standard disk I/O such as the network or a Compact Flash card. The Hard Totals use the synchronous processing model.

## Hard Totals Configuration Entries



Parameter	Description	Valid Values
Profile Name	The name of a specific configuration.	HardTotals.0 (Any name you wish)
Programmatic ID	This is the Service Object name for the Hard Totals	NCRPosSO.HardTotals
NCR Model Layer		NCRHardTotalsRC
Media Type	Indicates the media on which to store the Hard Totals data. This value may only be set to <i>Disk</i> . A media type of disk does not mean that the Hard Totals file must reside on a local hard drive; the Hard Totals file can	"Disk"

	reside on any device that supports the standard disk I/O operations, including a network drive or compact flash.	
HardTotals Area Filename ...	The file name of the totals file. If the file does not exist, then a file with the requested size is created in the desired directory. If the directory component of the file name is invalid, the service object will return an error at open time.	Example: C:\Program Files\NCR\Retail\NCRHT .001
HardTotals Size in KB	The size of the totals file, in kilobytes. The default value is 1024. This value is only used if the specified disk file does not exist. If a file size is specified, the size is rounded down to the nearest multiple of the cluster size (512 bytes for disk).	Example: 1024
ClusterSize in Bytes	The file system layout within the Hard Totals area. The Hard Totals file must contain at least 3 clusters to be useful, 1 cluster for the header, 1 for the file table, and 1 cluster to hold file data. If the size is less than 3 clusters, the size will be set to 3 clusters	Cluster size in bytes 128 256 512 (Default) 1024

## HardTotals Data Capture Configuration Entries

Data capture for the HardTotals is controlled through settings stored in the NCRDataCap.conf configuration file. The following example shows how to enable maximum data capture for all of the HardTotals components.

```
[NCROposSO.HardTotals]
```

```
LevelMask = 0xFFFFFFFF
TraceMask = 0xFFFFFFFF
UseDefault = F
```

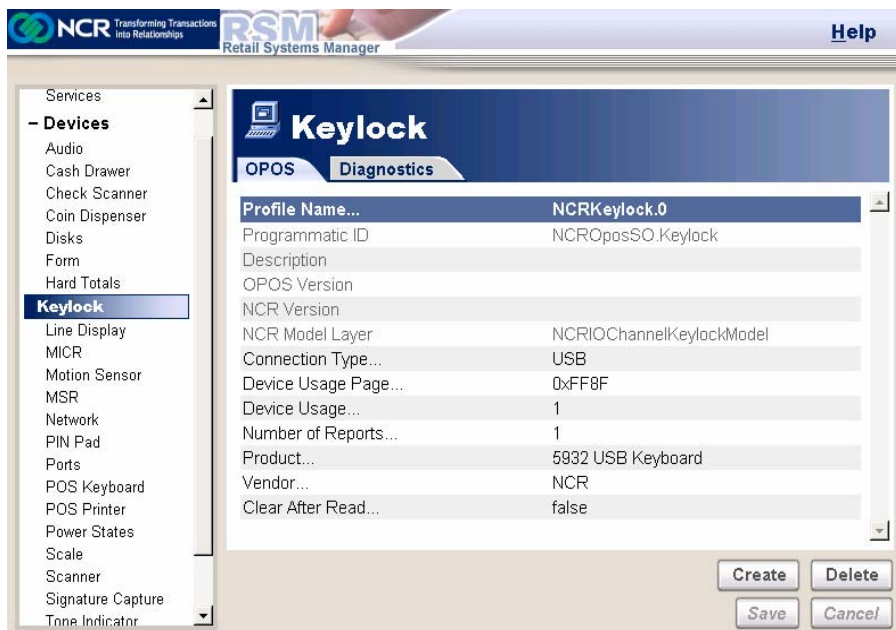
```
[NCRHardTotals]
```

```
LevelMask = 0xFFFFFFFF
TraceMask = 0xFFFFFFFF
UseDefault = F
```

# Keylock

The Keylock supports the keylock on the NCR 5932 USB keyboard or the NCR 5953 USB Dynakey.

## Keylock Configuration Entries



Parameter	Description	Valid Values
Profile Name	The name of a specific configuration.	NCRKeylock.0 (Any name you wish)
Programmatic ID	This is the Service Object name for the Hard Totals	NCROposSO.Keylock
NCR Model Layer		NCRIOChannelKeylockModel
Connection Type	The device interface used for the Keylock. USB is the only option.	USB (Read-Only)
Device Usage Page	USB Configuration - Indicates the usage page for this Keylock device.	0xFF8F (Read-Only)

Device Usage	USB Configuration -Indicates the usage for this Keylock device.	1 (Read-Only)
Number of Reports	USB Configuration - Indicates the number of reports for the Keylock device.	1 (Read-Only)
Product	USB Only - The name of the device supported.	Any HID USB Keylock (Default) 5932 USB Keyboard 5952-6xxxx/8xxx USB Dynakey 5952-9xxx USB Dynakey
Vendor	USB Only - The manufacturer of the Keylock.	Any Vendor NCR (Default)
Clear After Read	USB Only - Indicates whether the device memory should be cleared after a read.	False (Read-Only)

## Keylock Data Capture Configuration Entries

Data capture for the Keylock is controlled through settings stored in the NCRDatacap.conf configuration file. The following example shows how to enable maximum data capture for all of the Keylock components.

```
[NCROposSO.Keylock]
```

```
LevelMask = 0xFFFFFFFF
```

```
TraceMask = 0xFFFFFFFF
```

```
UseDefault = F
```

```
[NCRIOChannelKeylockModel]
```

```
LevelMask = 0xFFFFFFFF
```

```
TraceMask = 0xFFFFFFFF
```

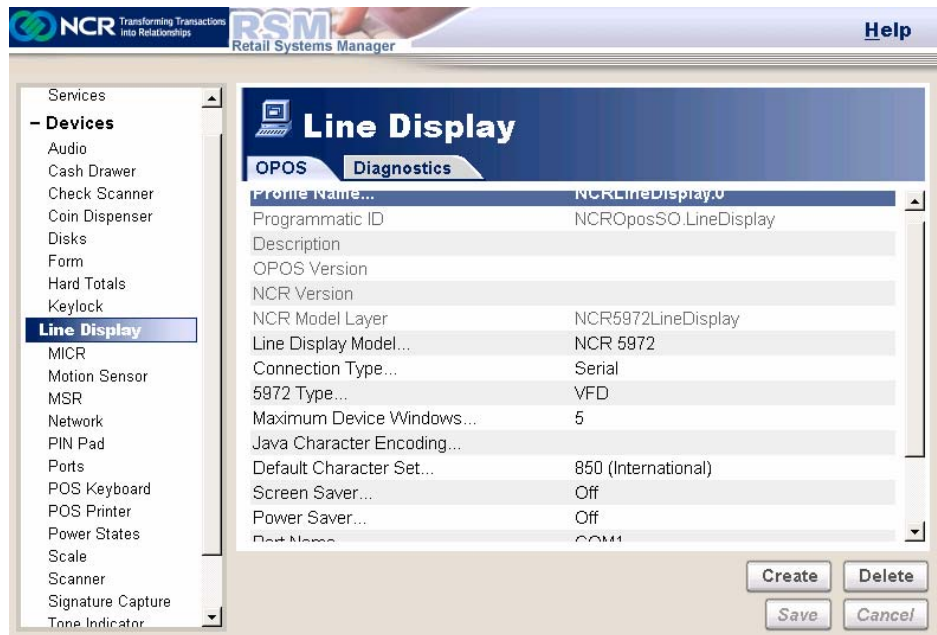
```
UseDefault = F
```

# Line Display

## NCR5972LineDisplay

The Line Display supports the NCR 5972 , LCD, VFD, and VFD Occular LCD Line Display.

## Line Display Configuration Entries



Parameter	Description	Valid Values
Profile Name	The name of a specific configuration.	NCRLineDisplay.0 (Any name you wish)
Programmatic ID	This is the Service Object name for the Line Display	NCR OposSO.LineDisplay
Line Display Model		NCR5972LineDisplay

Connection Type	Serial is the only option currently available.	Serial
5972 Type	The type of 5972 LCD device	VFD Ocular VFD LCD
Maximum Device Windows	The maximum number of logical windows that this device can support at once.	1 to 10 5 (Default)
Java Character Encoding	The Line Display operates using character sets, not Unicode. As a convenience for the application, the Line Display supports this parameter to automatically map Unicode strings into a particular character set using the specified Java Character Encoding type string. It is still the responsibility of the application to select the correct character set programmatically. The valid values for this parameter can be found in the Sun Java online documentation at URL. <a href="http://java.sun.com/products/jdk/1.2/docs/guide/internet/encoding.doc.html">http://java.sun.com/products/jdk/1.2/docs/guide/internet/encoding.doc.html</a> . If this parameter is omitted, or its value is an empty string, the application is assumed to be providing strings with character values between 0000 and 00FF where each character is already the correct value for the currently selected character set.	
Default Character Set	The character set to initialize during initial device enable.	<b>VFD</b> 850 - International (Default) 101 - Katakana 866 - Cyrillic 102 - 116, external ROM character sets <b>LCD</b> 850 - International <b>Ocular LCD</b> 858 - International
Screen Saver Style	The style of screen saver to enable after a predetermined interval of inactivity.	Off (Default) Screen Blanking Walking Text
Power Saver	Turn the low power state option after inactivity, On or Off.	Off (Default) On
PortName	The COM Port where the line display is attached.	COM1 - COM10 COM1 (Default for VFD and LCD) COM5 (Default for Ocular LCD)

## NCR5973LineDisplay

The Line Display supports the NCR 5973 International Line Display.

### Line Display Configuration Entries

Parameter	Description	Valid Values
Profile Name	The name of a specific configuration.	NCRLineDisplay.0 (Any name you wish)
Programmatic ID Line Display Model	This is the Service Object name for the Line Display	NCROposSO.LineDisplay NCR5973LineDisplay
Connection Type	Serial is the only option currently available.	Serial
Maximum Device Windows	The maximum number of logical windows that this device can support at once.	1 to 10 10 (Default)
Java Character Encoding	The Line Display operates using character sets, not Unicode. As a convenience for the application, the Line Display supports this parameter to automatically map Unicode strings into a particular character set using the specified Java Character Encoding type string. It is still the responsibility of the application to select the correct character set programmatically. The valid values for this parameter can be found in the Sun Java online documentation at URL. <a href="http://java.sun.com/products/jdk/1.2/docs/guide/internet/encoding.doc.html">http://java.sun.com/products/jdk/1.2/docs/guide/internet/encoding.doc.html</a> . If this parameter is omitted, or its value is an empty string, the application is assumed to be providing strings with character values between 0000 and 00FF where each character is already the correct value for the currently selected character set.	
Default Character Set	The character set to initialize during initial device enable.	850 – International English 932 – Japanese (Default) 950 – Traditional Chinese 936 – Simplified Chinese 949 - Korean
PortName	Serial only - The COM Port where the line display is	COM1 - COM10



attached.

COM1 (Default)

## NCR59752x20LineDisplay

The Line Display supports the NCR 5975-1xxx 2x20 Line Display connected either by Serial or USB.

## Line Display Configuration Entries

Parameter	Description	Valid Values
Profile Name	The name of a specific configuration.	NCRLineDisplay.0 (Any name you wish)
Programmatic ID Line Display Model	This is the Service Object name for the Line Display	NCROposSO.LineDisplay NCR59752x20LineDisplay
Connection Type	Serial or USB.	Serial USB
Maximum Device Windows	The maximum number of logical windows that this device can support at once.	1 to 10 5 (Default)
Java Character Encoding	The Line Display operates using character sets, not Unicode. As a convenience for the application, the Line Display supports this parameter to automatically map Unicode strings into a particular character set using the specified Java Character Encoding type string. It is still the responsibility of the application to select the correct character set programmatically. The valid values for this parameter can be found in the Sun Java online documentation at URL. <a href="http://java.sun.com/products/jdk/1.2/docs/guide/internet/encoding.doc.html">http://java.sun.com/products/jdk/1.2/docs/guide/internet/encoding.doc.html</a> . If this parameter is omitted, or its value is an empty string, the application is assumed to be providing strings with character values between 0000 and 00FF where each character is already the correct value for the currently selected character set.	
Default Character	The character set to initialize during initial device	850 - International

Set	enable.	(Default) 101 - Katakana 866 - Cyrillic 102 - 116, external ROM character sets
Screen Saver Style	The style of screen saver to enable after a predetermined interval of inactivity.	Off (Default) Blank Walk
Power Saver	Turn the low power state option after inactivity, On or Off.	Off (Default) On
PortName	Serial only - The COM Port where the line display is attached.	COM1 - COM10 COM1 (Default)

## NCR5975IntLineDisplay

The Line Display supports the NCR 5975-2xxx Grphical (International) Line Display connected either by Serial or USB.

## Line Display Configuration Entries

Parameter	Description	Valid Values
Profile Name	The name of a specific configuration.	NCRLineDisplay.0 (Any name you wish)
Programmatic ID Line Display Model	This is the Service Object name for the Line Display	NCROposSO.LineDisplay NCR5975IntLineDisplay
Connection Type	Serial or USB.	Serial USB
Maximum Device Windows	The maximum number of logical windows that this device can support at once.	1 to 10 10 (Default)
Java Character Encoding	The Line Display operates using character sets, not Unicode. As a convenience for the application, the Line Display supports this parameter to automatically map Unicode strings into a particular character set using the specified Java Character Encoding type string. It is still the responsibility of the application to select	

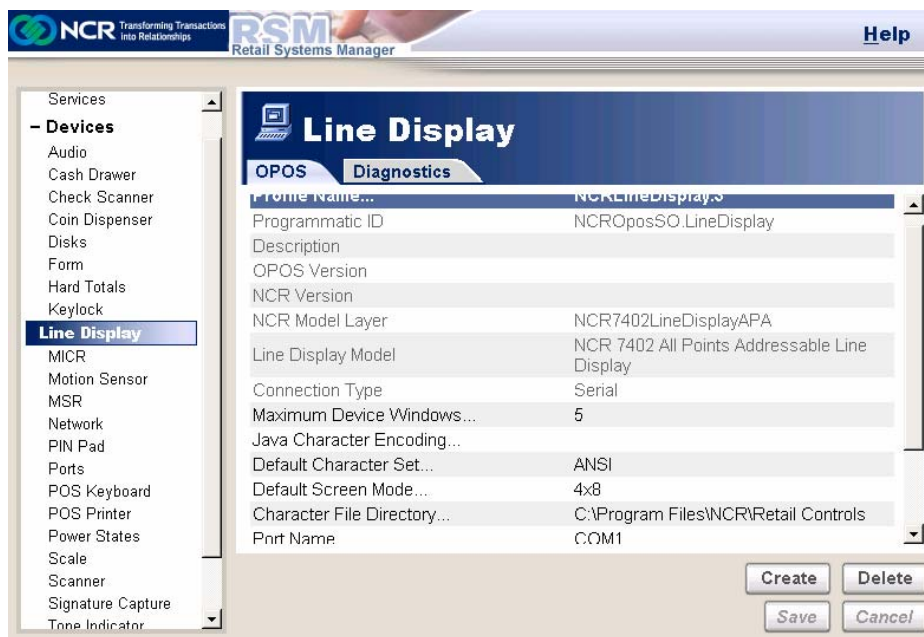
the correct character set programmatically. The valid values for this parameter can be found in the Sun Java online documentation at URL.  
<http://java.sun.com/products/jdk/1.2/docs/guide/internet/encoding.doc.html>. If this parameter is omitted, or its value is an empty string, the application is assumed to be providing strings with character values between 0000 and 00FF where each character is already the correct value for the currently selected character set.

Default Character Set	The character set to initialize during initial device enable.	850 – International English 932 – Japanese (Default) 950 – Traditional Chinese 936 – Simplified Chinese 949 - Korean
PortName	Serial only - The COM Port where the line display is attached.	COM1 - COM10 COM1 (Default)

## NCR7402LineDisplayAPA

The Line Display supports the APA Line Display on the NCR 7402.

## Line Display Configuration Entries



Parameter	Description	Valid Values
Profile Name	The name of a specific configuration.	NCRLineDisplay.0 (Any name you wish)
Programmatic ID Line Display Model	This is the Service Object name for the Line Display	NCROposSO.LineDisplay NCR7402LineDisplayAPA
Maximum Device Windows	The maximum number of logical windows that this device can support at once.	1 to 10 5 (Default)
Java Character Encoding	The Line Display operates using character sets, not Unicode. As a convenience for the application, the Line Display supports this parameter to automatically map Unicode strings into a particular character set using the specified Java Character Encoding type string. It is still the responsibility of the application to select the correct character set programmatically. The valid values for this parameter can be found in the Sun Java online documentation at URL. <a href="http://java.sun.com/products/jdk/1.2/docs/guide/inter/net/encoding.doc.html">http://java.sun.com/products/jdk/1.2/docs/guide/inter/net/encoding.doc.html</a> . If this parameter is omitted, or its value is an empty string, the application is	

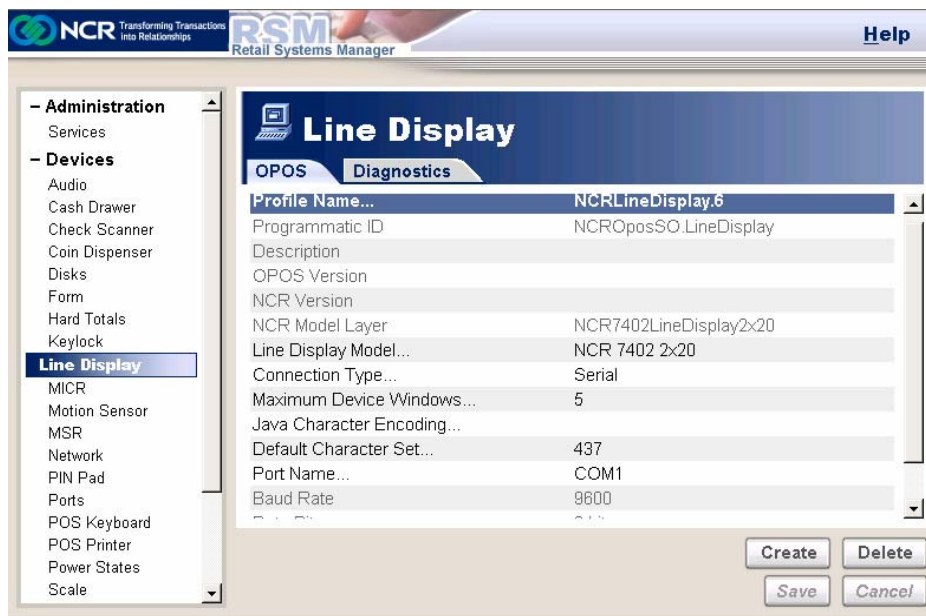
assumed to be providing strings with character values between 0000 and 00FF where each character is already the correct value for the currently selected character set.

Default Character Set	The character set to initialize during initial device enable.	ANSI (Default) Unicode
Default Screen Mode	The Row and Column resolution.	4x8 (Default) 4x16 8x16
Character File Directory	Where character renderings are located	C:\Program Files\NCR\Retail Controls
PortName	The COM Port where the line display is attached.	COM1 - COM10

## NCR7402LineDisplay2x20

The Line Display supports the 2x20 Line Display on the NCR 7402.

## Line Display Configuration Entries



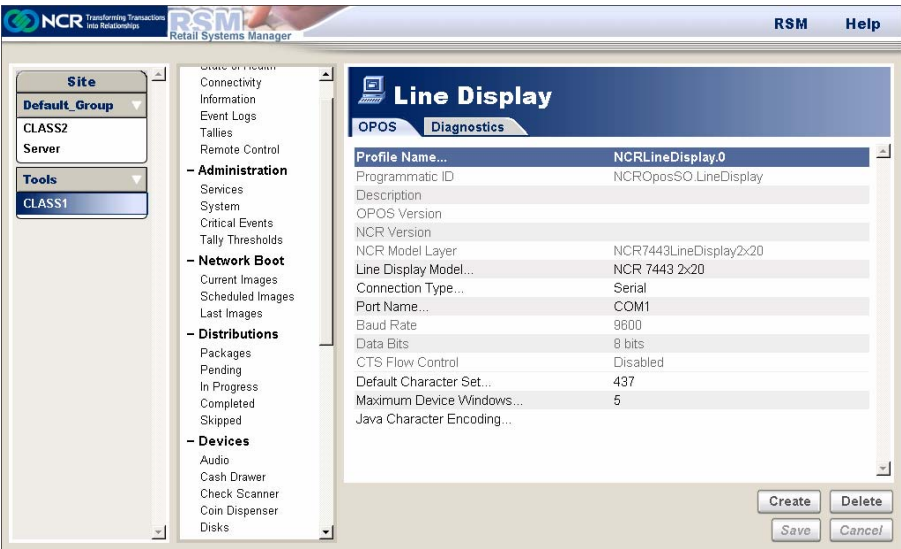
Parameter	Description	Valid Values
Profile Name	The name of a specific configuration.	NCRLineDisplay.0 (Any name you wish)
Programmatic ID	This is the Service Object name for the Line Display	NCROposSO.LineDisplay
Line Display Model		NCR7402LineDisplay2x20
Connection Type	Serial is the only option currently available.	0
Maximum Device Windows	The maximum number of logical windows that this device can support at once.	Serial
Java Character Encoding	The Line Display operates using character sets, not Unicode. As a convenience for the application, the Line Display supports this parameter to automatically map Unicode strings into a particular character set using the specified Java Character Encoding type string. It is still the responsibility of the application to select the correct character set programmatically. The valid values for this parameter can be found in the Sun Java online documentation at URL. <a href="http://java.sun.com/products/jdk/1.2/docs/guide/internet/encoding.doc.html">http://java.sun.com/products/jdk/1.2/docs/guide/internet/encoding.doc.html</a> . If this parameter is omitted, or its value is an empty string, the application is assumed to	1 to 10 5 (Default)

	be providing strings with character values between 0000 and 00FF where each character is already the correct value for the currently selected character set.	
Default Character Set	The character set to initialize during initial device enable.	437, 858, 866, 101 (Default 437)
PortName	The COM Port where the line display is attached.	COM1 - COM10 (Default COM3)

NCR7443/7446LineDisplay2x20

The Line Display supports the 2x20 Line Display on the NCR 7402.

Line Display Configuration Entries



Parameter	Description	Valid Values
Profile Name	The name of a specific configuration.	NCRLineDisplay.0 (Any name you wish)
Programmatic ID	This is the Service Object name for the Line Display	NCR OposSO.LineDisplay

Line Display Model		NCR744xLineDisplay2x2 0
Connection Type	Serial is the only option currently available.	Serial
Maximum Device Windows	The maximum number of logical windows that this device can support at once.	1 to 10 5 (Default)
Java Character Encoding	The Line Display operates using character sets, not Unicode. As a convenience for the application, the Line Display supports this parameter to automatically map Unicode strings into a particular character set using the specified Java Character Encoding type string. It is still the responsibility of the application to select the correct character set programmatically. The valid values for this parameter can be found in the Sun Java online documentation at URL. <a href="http://java.sun.com/products/jdk/1.2/docs/guide/internet/encoding.doc.html">http://java.sun.com/products/jdk/1.2/docs/guide/internet/encoding.doc.html</a> . If this parameter is omitted, or its value is an empty string, the application is assumed to be providing strings with character values between 0000 and 00FF where each character is already the correct value for the currently selected character set.	
Default Character Set	The character set to initialize during initial device enable.	437
PortName	The COM Port where the line display is attached.	COM1 - COM10 (Default COM3)

## LineDisplay Data Capture Configuration Entries

Data capture for the LineDisplay is controlled through settings stored in the NCRDatacap.conf configuration file. The following example shows how to enable maximum data capture for all of the LineDisplay components.

```
[NCROposSO.LineDisplay]
```

```
LevelMask = 0xFFFFFFFF
```

```
TraceMask = 0xFFFFFFFF
```

```
UseDefault = F
```

```
[NCR5972LineDisplay]
```



LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F

[NCR5973LineDisplay]

LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F

[NCR59752x20LineDisplay]

LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F

[NCR5975IntLineDisplay]

LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F

[NCR7402LineDisplayAPA]

LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F

[NCR7402LineDisplay2x20]

LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F

[NCR744xLineDisplay2x20]

LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F

## MICR

The MICR reader supports the MICR connected to one of the supported NCR Printers.

### DirectIO – MICR Parsing

The MICR\_ADD\_EXCEPTION method adds a special case MICR parsing template string to the service object's MICR parsing template list.

MICR data consists of digits, spaces, and four special MICR symbols: Transit, On-Us, Amount, Dash. In the following discussion, and in the DirectIO command NCRDIO\_MICR\_ADD\_EXCEPTION, these characters are used to describe MICR format data. Using this DirectIO Command with NCRDIO\_MICR\_ADD\_EXCEPTION has the effect of adding these special MICR template strings to the persistent storage in addition to adding them to the parsing template list temporarily stored in memory.

t o a - space	MICR substitution symbols for Transit, On-Us, Amount, Dash, and Space.
X	Digit, space, or dash.
E	EPC: Digit.
T	Transit number: Digit or dash. (For US checks, only digits are allowed.)
A	Account number: Digit, space, or dash. (Account number begins and ends with a digit.)
S	Serial/sequence number: Digit.
\$	Amount: Digit.
B	Series of blanks (spaces) and/or dashes. Optional unless a repeat count is given.
*	Trailer: Series of any digits, dashes, and spaces followed by an optional amount field. (=

"X[0+]a\$[10]aB" if amount present and "X[0+]" if not).

An optional repeat count may follow any character except \*:

[count]	Specifies exactly "count" characters.
[min-max]	Specifies between "min" and "max" characters.
[min+]	Specifies at least "min" characters.

The general MICR data format is:

oX[1+]o EtT[9]t X[1+]oX[0+] a\$[10]a

The fields are:

oX[1+]o	Auxiliary on-us (optional). Bounded on left and right with On-Us characters. Often used for serial number, in which case 'X' are digits.
E	EPC character (optional). Immediately precedes leftmost Transit character.
tT[9]t	Transit number (required). Bounded on left and right with Transit characters.
X[1+]oX[0+]	On-us (optional). (More than one On-Us symbol can be included.
a\$[10]a	Amount (optional). Bounded on left and right with Amount characters.

The transit number and amount can always be found and properly parsed. The placement of the serial and account numbers, however, varies between banks. In addition, the Auxiliary On-Us and the On-Us fields may contain other bank-specific data.

To handle the majority of cases, the following default handling of serial and account numbers is performed:

Case (1): Serial number in Auxiliary On-Us field. Account number in On-Us field from first digit to the On-Us symbol, not including spaces and dashes before the On-Us.

"BoS[1+]oBEtT[9]tBA[1+]Bo\*"

- Serial number is the number S[1+]
- Account number is the number A[1+].

Example:

"o9876o t123456789t 12-345-6789 o 6666 a0000054321a"

Transit: "123456789"

Amount: "0000054321"

Account: "12-345-6789"

Serial: "9876"

EPC: ""

\*\* If the RemoveNonDigits string contains an 'A', then Account is "123456789".

Case (2): Account number in On-Us field from first digit to the On-Us symbol, not including spaces and dashes before the On-Us. Serial number in On-Us field after the On-Us symbol. Must be three or more digits.

"BEtT[9]tBo[0-1]A[1+]BoBS[3+]\*"

- Account number is the number A[1+], if present.
- Serial number is the number S[3+]

Example:

"5t123456789t 12 345-67 o 6666"

Transit: "123456789"

Amount: ""

Account: "12 345-67"

Serial: "6666"

EPC: "5"

\*\* If the RemoveNonDigits string contains an 'A', then Account is "1234567".

Case (3): Serial number in On-Us field from first digit to next non-digit; must be between three and five digits. Account number in On-Us field from first digit after the serial number to the On-Us symbol, not including spaces and dashes before the On-Us.

"BEtT[9]tBS[3-5]B[1+]A[1+]Bo\*"

- Serial number is the number S[3-5]
- Account number is the number A[1+].

Example:

"t12345-789t 555 12 345-67 o 66"

Transit: "12345-789"

Amount: ""

Account: "12 345-67"

Serial: "555"

EPC: ""

\*\* If the RemoveNonDigits string contains a 'T', then Transit is "12345789".

\*\* If the RemoveNonDigits string contains an 'A', then Account is "1234567".

Example:

"t123456789t 555 12 345-67 o 66 a0000054321a"

Transit: "123456789"

Amount: "0000054321"

Account: "12 345-67"

Serial: "555"

EPC: ""

\*\* If the RemoveNonDigits string contains an 'A', then Account is "1234567".

Case (4): Account number in On-Us field from first digit to the On-Us symbol, not including spaces and dashes before the On-Us.

"BEtT[9]tBA[1+]Bo\*"

- No serial number.
- Account number is the number A[1+].

Example:

"t123456789t 55 12 345-67 o 66"

Transit: "123456789"

Amount: ""

Account: "55 12 345-67"

Serial: ""

EPC: ""

\*\* If the RemoveNonDigits string contains an 'A', then Account is "551234567".

Case (5): Serial number in On-Us field from first digit to the On-Us symbol,

must be three or more digits. Account number in second On-Us field from the first digit after the On-Us symbol that ended the serial number to the On-Us symbol, not including spaces and dashes before the On-Us.

"BtT[9]tBS[3+]BoBA[1+]o\*",

- Serial number is the number S[3+].
- Account number is the number A[1+].

Example:

"t12345-789t 555 o 12 345-67 o 66"

Transit: "12345-789"

Amount: ""

Account: "12 345-67"

Serial: 555

EPC: ""

\*\* If the RemoveNonDigits string contains a 'T',  
then

Transit is "12345789".

\*\* If the RemoveNonDigits string contains an 'A',  
then Account is "1234567".

If the default handling is not correct for some banks, then exception strings can be added for these cases. The exception strings shall be formatted as in these examples.

Example exception string "t061000052tBA[10-12]o\*":

This string requires that the EPC and Serial Number be absent.

If Micr Data is: "t061000052t 123 456 789o"

Transit: "061000052"

Amount: ""

Account: "123 456 789"

Serial: ""

EPC: ""

\*\* If the RemoveNonDigits string contains an 'A', then  
Account is "123456789".

(Without the exception, then default case (3) would have returned "123" as the serial number and "456 789" (\*\* or 456789) as the account number.)

If MICR Data is: "t061000052t

123456789012o3333a0000054321a"

Transit: "061000052"

Amount: "0000054321"

Account: "123456789012"

Serial: ""

EPC: ""

(Note that the "3333" between the On-Us and Amount symbols is ignored. Without the exception, then default case (2) would have returned "3333" as the serial number.)

Example exception string "t052000113tBS[3]BA[7-10]o\*":

This string requires that the EPC be absent.

If Micr Data is: "t052000113t 123 456 789o"

Transit: "052000113"

Amount: ""

Account: "456 789"

Serial: "123"

EPC: ""

\*\* If the RemoveNonDigits string contains an 'A', then Account is "456789".

Example exception string "tTTT00011TtBS[2]A[10]S[3]o\*":

This string requires that the EPC be absent.

If MICR Data is: "t99-000119t 123 456 7890987o

a9876543210a"

Transit: "99-000119"

Amount: "9876543210"

Account: "3 456 7890"

Serial: "12987"

EPC: ""

\*\* If the RemoveNonDigits string contains a 'T', then Transit is "99000119".

**\*\*** If the RemoveNonDigits string contains an 'A', then Account is "34567890".

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

**Parameter Description**

---

*command*    NCRDIO\_MICR\_ADD\_EXCEPTION

*data*        Not Used

*object*      String to add to the exception list (See command description)

**Errors**      An exception may be thrown when this method is invoked.  
Some possible values of the exception's *ErrorCode* property are:

JPOS\_E\_ILLEGAL - The string contains invalid characters.



## DirectIO - Set Remove Non Digits

Sets space and dash removal from account and transit numbers. The following strings are valid:

- "A": Remove spaces and dashes from within **AccountNumber**. Some banks use these for readability.
- "T": Remove spaces and dashes from within **TransitNumber**. Some countries may use these.
- "AT": Remove spaces and dashes from within **AccountNumber** and **TransitNumber**.

"": Do not remove spaces and dashes from either **AccountNumber** or **TransitNumber**.

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

<b>Parameter Description</b>	
<i>command</i>	NCRDIO_MICR_SET_REMOVE_NON_DIGITS
<i>data</i>	Not Used
<i>object</i>	List of characters to remove
"A"	Remove the account character
"T"	Remove the transit character
"AT"	Remove both characters
""	Do not remove any characters

**Errors**      An exception may be thrown when this method is invoked.

Some possible values of the exception's *ErrorCode* property are:

JPOS\_E\_ILLEGAL - The string contains invalid characters.

## Clear Exception Table

Clears, the MICR Exception Table in temporary system memory, the configuration data, or both.

**Syntax**     **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

**Parameter Description**

---

*command*     NCRDIO\_MICR\_CLEAR\_EXCEPTIONS

*data*             0 = Clear all exception strings  
                    1 = Clear exception strings entered via DirectIO  
                    2 = Clear exception strings entered via  
                    configuration data

*object*          Not Used

**Errors**         An exception may be thrown when this method is invoked.  
  
Some possible values of the exception's *ErrorCode* property are:

JPOS\_E\_ILLEGAL     Unsupported data value entered, the function failed.

## MICR - Magnetic Ink Character Recognition Configuration Entries



Parameter	Description	Valid Values
Profile Name	The name of a specific configuration.	NCRMICR.0 (Any name you wish)
Programmatic ID	This is the Service Object name for the MICR	NCROposSO.MICR
NCR Model Layer		NCR71xxPrinterCDMICRModel
Uses	The MICR reader is integrated in a retail printer. Enter the profile name of the associated POS Printer	POSPrinter\NCRPOSPrinter.0
Exception String 0	The ExceptionString entries are used to add exception patterns for MICR check parsing to handle special formatted checks that are not covered by the default parsing rules.	Refer to the DirectIO "MICR Parsing" for valid strings
Exception String 1	The ExceptionString entries are used to add exception patterns for MICR check parsing to handle special formatted checks that are not covered by the default parsing rules.	Refer to the DirectIO "MICR Parsing" for valid strings
Remove Non-Digits	RemoveNonDigits is used to remove digits	Don't remove digits from MICR data

	from parsed MICR data. Digits may be removed from transit numbers, account numbers, both, or neither.	Remove from transit numbers Remove from account numbers Remove from transit and account numbers
Return Error Data	In some cases, a failed MICR read may return some MICR data. ReturnErrorData determines if we should parse and return any data read as part of a failed MICR read. If enabled, the error data is returned in MICR error events.	Return partial data Don't return partial

## MICR Data Capture Configuration Entries

Data capture for the MICR is controlled through settings stored in the NCRDatacap.conf configuration file. The following example shows how to enable maximum data capture for all of the MICR components.

```
[NCR0posSO.MICR]
```

```
LevelMask = 0xFFFFFFFF
```

```
TraceMask = 0xFFFFFFFF
```

```
UseDefault = F
```

```
[NCR71xxMICRModel]
```

```
LevelMask = 0xFFFFFFFF
```

```
TraceMask = 0xFFFFFFFF
```

```
UseDefault = F
```

```
[NCR71xxPrinterCDMICRModel]
```

```
LevelMask = 0xFFFFFFFF
```

```
TraceMask = 0xFFFFFFFF
```

```
UseDefault = F
```

# MotionSensor

The Motion Sensor supports the motion sensors integrated into the NCR 7401 and 7402 terminals.

## MotionSensor - Motion Sensor Configuration Entries



Parameter	Description	Valid Values
Profile Name	The name of a specific configuration.	NCRMotionSensor.0 (Any name you wish)
Programmatic ID	This is the Service Object name for the Motion Sensor	NCROposSO.MotionSensor
NCR Model Layer		NCRSystemMonitorEventModel

## Motion Sensor Data Capture Configuration Entries

Data capture for the Motion Sensor is controlled through settings stored in the NCRDatacap.conf configuration file. The following example shows how to enable maximum data capture for all of the Motion Sensor components.

```
[NCROposSO.MotionSensor]
```

```
LevelMask = 0xFFFFFFFF
```

```
TraceMask = 0xFFFFFFFF
```

```
UseDefault = F
```

```
[NCRSystemMonitorEventModel]
```

```
LevelMask = 0xFFFFFFFF
```

```
TraceMask = 0xFFFFFFFF
```

```
UseDefault = F
```

# MSR

The MSR reader supports the MSR located on the NCR 5932 USB Keyboard or the NCR 5952 USB Dynakey.

## MSR - Magnetic Stripe Reader Configuration Entries

Parameter	Value
Profile Name...	NCRMSR0
Programmatic ID	NCROposSO.MSR
Description	
OPOS Version	
NCR Version	
NCR Model Layer	NCRIOChannelMSRModel
Read Head Type...	ISO
Connection Type...	USB
Device Usage Page...	0x8E
Device Usage...	1
Number of Reports...	1
Product...	5932 USB Keyboard
Vendor...	NCR
After Read	Clear

Parameter	Description	Valid Values
Profile Name	The name of a specific configuration.	NCRMSR.0 (Any name you wish)
Programmatic ID	This is the Service Object name for the MICR	NCROposSO.MSR
NCR Model Layer		NCRIOChannelMSRModel
Read Head Type	Type of MSR Hardware Reader. Different hardware is required for each format. A type of "ISO" indicates that ISO and JIS Type I cards can be read. A type of "JIS" indicates that JIS Types I and II cards can be read.	"ISO" (Default) "JIS"
ConnectionType	How the MSR is connected to the terminal.	USB (Default)

Device Usage Page	USB Configuration - Indicates the usage page for this MSR Device.	0x8E (Read-Only)
Device Usage	USB Configuration - Indicates the usage for this MSR Device.	1 (Read-Only)
Number of Reports	USB Configuration - Indicates the number of reports for this MSR Device.	1 (Read-Only)
Product	The Product where the USB MSR device resides.	Any HID USB MSR (Default) 5932 USB Keyboard 5952-6xxx/8xxx USB Dynakey 5952-9xxx USB Dynakey (ISO) 5952-9xxx USB Dynakey (JIS)
Vendor	The manufacturer of the MSR device.	Any Vendor NCR (Default)
ClearAfterRead	Should the device memory be cleared after a read.	True (Read-Only)

## MSR Data Capture Configuration Entries

Data capture for the MSR is controlled through settings stored in the NCRDatacap.conf configuration file. The following example shows how to enable maximum data capture for all of the MSR components.

[NCROposSO.MSR]

LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F

[NCRIOChannelMSRModel]

LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F

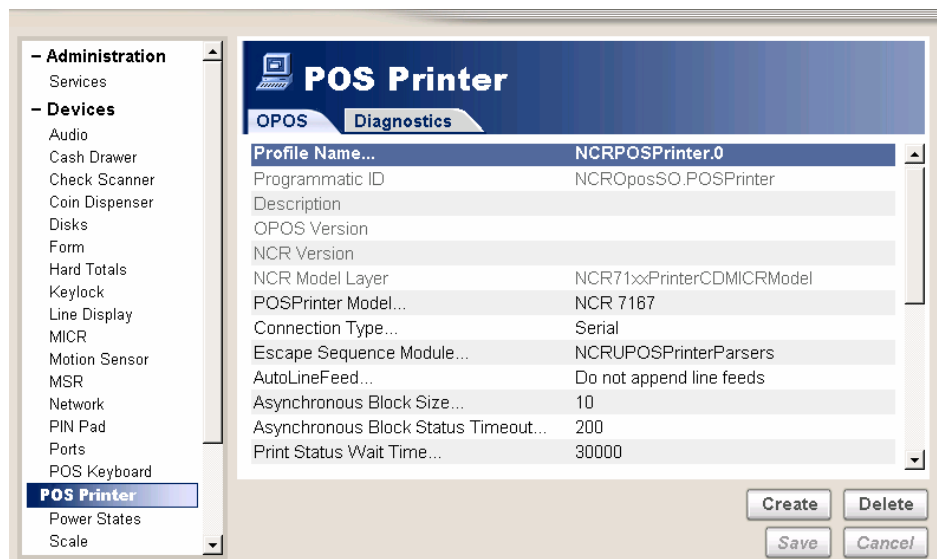


# POS Printer

The POS Printer supports the following NCR printers.

- 7156
- 7158
- 7167
- 7197
- K580
- K590
- 7342-F306

The Printer Model currently interfaces to the printer over a serial port, and it functions on any terminal with a standard serial port.



## DirectIO - Raw Output

This function is used to send data directly to the printer without having the Control manipulate that data in any way. This command can be used to access printer specific functions that are not provided by the Control.

**Note:** Do not use this function to alter any of the printer's physical printing characteristics (such as character pitch, lines per inch, margins or other such metrics). The Control knows nothing about the information that has been sent to the printer. If this command is used to alter any of the printer's physical printing characteristics, the Control may not be able to perform further requests correctly.

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

---

### Parameter Description

---

*command*    NCRDIO\_PTR\_RAW\_OUTPUT

*data*[0]      Printer Station identifier

*object*       Data being sent to the printer. The data must be of type java.lang.String.

**Errors**        An exception may be thrown when this method is invoked.

Some possible values of the exception's *ErrorCode* property are:

JPOS\_E\_ILLEGAL - The printer station is invalid.

## DirectIO - Set Barcode Width

Sets a new width for printing barcodes. The legal values for this option vary by printer model and should be determined from the printer's manual. This function returns the barcode width in the Data field and can therefore be used to determine what the current value is by sending in an illegal value and ignoring the error returned.

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### **Parameter Description**

---

*Command*    NCRDIO\_PTR\_SET\_BARCODE\_WIDTH

*data*            Horizontal size of the barcode, in dots

*object*          Not Used

## DirectIO - Set Bitmap Type

Sets a new type for bitmap print processing.

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### **Parameter Description**

---

*command*    NCRDIO\_PTR\_SET\_BITMAP\_TYPE

*data*[0]      bitmap type

NCRDIO\_BITMAP\_TYPE\_DEFAULT  
NCRDIO\_BITMAP\_TYPE\_HIGHQ  
NCRDIO\_BITMAP\_TYPE\_LOWQ  
NCRDIO\_BITMAP\_TYPE\_HIGHQ\_DL  
NCRDIO\_BITMAP\_TYPE\_LOWQ\_DL

*object*      Not Used

**Errors**      An exception may be thrown when this method is invoked.

Some possible values of the exception's *ErrorCode* property are:

JPOS\_E\_ILLEGAL - The bitmap type constant was invalid.

## DirectIO - Sound Audible Tone

Sounds a tone from the printer a specified number of times. The repeat count value can only range between 1 and 20. If a repeat count value less than one is sent, OPOS\_E\_ILLEGAL is returned.

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### **Parameter Description**

---

*command*    NCRDIO\_PTR\_SOUND\_AUDIBLE\_TONE

*data*[0]      Repeat count

*object*       Not Used

**Errors**        An exception may be thrown when this method is invoked.

Some possible values of the exception's *ErrorCode* property are:

JPOS\_E\_ILLEGAL - The printer cannot sound an audible tone or a repeat count of less than 1 or greater than 20 was selected.

## DirectIO - Set Minimum Number of Characters Per Line

Sets the minimum number of characters per line that are used when choosing a station's print mode. The mode (generally either standard or compressed printing) that is selected will be sufficient to print a line whose width is the greater of this value and the most recently set `XxxLineChars` property.

For example, the application needs to print on the 7167 slip in compressed mode, so that 40 characters of print data can fit on a small form. If `SlpLineChars` is set to 40, the service object normally selects standard mode, since up to 45 characters can be printed on a slip line in standard mode. By using this `DirectIO` to set the minimum number of characters per line to 55, the service object selects compressed mode for any `SlpLineChars` value, since compressed mode is required to print 55 characters on a line.

**Syntax**      **`void directIO (int command, int[] data, Object object) throws JposException;`**

---

### Parameter Description

---

<i>command</i>	NCRDIO_PTR_SET_MIN_LINE_CHARS_MODE
<i>data[0]</i>	Upper word ( <i>data[0]</i> & 0xFFFF0000): Station Lower word ( <i>data[0]</i> & 0x0000FFFF); Minimum number of characters  Upon return, <i>data[0]</i> contains the previous minimum number of characters per line.
<i>object</i>	Not Used

**Errors**      An exception may be thrown when this method is invoked.  
  
Some possible values of the exception's *ErrorCode* property are:

JPOS\_E\_ILLEGAL if the printer station cannot support the request number of characters or the station is invalid (in this case the extended result code is set to (NCR\_EPTR\_STATION)).

## DirectIO - Get the Current Printer Status

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### **Parameter Description**

---

*command*    NCRDIO\_PTR\_GET\_PRINTER\_STATUS

*data[0]*     The status is stored in data[0].

*object*      Not Used

**Errors**      An exception may be thrown when this method is invoked.  
  
Some possible values of the exception's *ErrorCode* property are:

If successful, the following table provides a detailed description of the data returned. It is described in Bit Mask form.

<b>Defined Constant</b>	<b>Description</b>
PRTSTAT_FLAG	Always On
PTRSTAT_ALL	All statuses
PTRSTAT_STATE	All state flags
PTRSTAT_PRINTER	All printer statuses
PTRSTAT_PRINTER_FAULT	All printer faults/failures
PTRSTAT_DRAWER	All drawer statuses
<b>Printer States</b>	
PTRSTAT_OFF_STATE	Printer is off
PTRSTAT_ERROR_STATE	Printer is in error: Not accepting requests
<b>OPOS Printer Sensor values.</b>	
PTRSTAT_COVER_OPEN	Cover or door open
PTRSTAT_JRN_EMPTY	Journal out
PTRSTAT_JRN_NEAREND	Journal low



Defined Constant	Description
PTRSTAT_REC_EMPTY	Receipt out
PTRSTAT_REC_LOW	Receipt low
PTRSTAT_SLP_EMPTY	Slip leading edge sensor: no paper
PTRSTAT_SLP_LOW	Slip trailing edge sensor: Low on paper
<b>Printer Slip sensor values.</b>	
PTRSTAT_SLP_TOF	Slip top of form sensor on
PTRSTAT_SLP_BOF	Slip bottom of form sensor on
<b>Printer states only available when real-time commands are supported.</b>	
PTRSTAT_SLP_SELECTED	Slip selected
PTRSTAT_SLP_WAITING	Printer waiting for insertion
<b>Printer failure conditions. Support varies by printer.</b>	
PTRSTAT_AUTORECOVER_FAULT	Auto-recoverable fault, such as temperature fault
PTRSTAT_MOTOR_FAULT	Motor fault
PTRSTAT_KNIFE_FAULT	Knife fault
PTRSTAT_FAILURE	Unrecoverable error
<b>Drawer states.</b>	
PTRSTAT_DWR1_OPEN	Drawer 1 open
PTRSTAT_DWR2_OPEN	Drawer 2 open

## DirectIO – Get Kiosk Specific Sensor Status

Returns the value of one of the kiosk specific sensors.

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### **Parameter Description**

---

*command*    NCRDIO\_PTR\_SENSOR\_STATUS

*data*[0]      The sensor to check

PTR\_Cabinet\_Door\_Switch  
PTR\_Presenter\_Paper\_Sensor  
PTR\_Cabinet\_Paper\_Chute  
PTR\_Presenter\_Orientation

On return, *data*[0] holds the sensor value.  
This will be one of:

PTR\_CDS\_Door\_Open  
PTR\_CDS\_Door\_Closed  
PTR\_PPS\_Paper\_Present  
PTR\_PPS\_No\_Paper  
PTR\_CPC\_Paper\_Present  
PTR\_CPC\_No\_Paper  
PTR\_PO\_Normal  
PTR\_PO\_Rotated\_90

*object*      Not Used

**Errors**      An exception may be thrown when this method is invoked.

Some possible values of the exception's *ErrorCode* property are:

JPOS\_E\_ILLEGAL - The sensor constant was invalid.

## DirectIO – Set Autoeject Timeout

Sets the autoeject timeout for the K580 or K590 kiosk printer. This time-out is the delay between the time a receipt is presented, and when it is auto ejected onto the floor.

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### **Parameter Description**

---

*command*    NCRDIO\_PTR\_AUTOEJECT\_TIMEOUT

*data*[0]        The new timeout, in seconds. This value may be either 20, 40, 60, or 80. Use the value JPOS\_FOREVER to indicate that the receipt should not be automatically ejected.

*object*        Not Used

**Errors**        An exception may be thrown when this method is invoked.

Some possible values of the exception's *ErrorCode* property are:

JPOS\_E\_ILLEGAL – The attached printer does not support receipt ejection, or the autoeject value was invalid.

## DirectIO – Eject Receipt

Ejects the receipt from the K580 or K590 kiosk printer.

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### **Parameter Description**

---

*command*    NCRDIO\_PTR\_EJECT\_RECEIPT

*data*[0]        Not used

*object*        Not Used

**Errors**        An exception may be thrown when this method is invoked.

Some possible values of the exception's *ErrorCode* property are:

JPOS\_E\_ILLEGAL – The attached printer does not support receipt ejection.

## DirectIO – Get Kiosk Specific Status

Returns the kiosk specific status bits.

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

**Parameter Description**

---

*command*    NCRDIO\_PTR\_GET\_KIOSK\_STATUS

*data*[0]      On return, *data*[0] holds the kiosk specific status bits. This will be a combination of the following values:

PTR\_CDS\_Door\_Open    Cabinet door open  
PTR\_CDS\_Door\_Closed   Cabinet door closed

PTR\_PPS\_Paper\_Present   Presenter paper present

PTR\_PPS\_No\_Paper      Presenter paper out

PTR\_CPC\_Paper\_Present   Cabinet paper chute: Paper present

PTR\_CPC\_No\_Paper      Cabinet paper chute: No paper

PTR\_PO\_Rotated\_90      Presenter rotated

PTR\_PO\_Normal          Presenter not rotated (normal)

*object*          Not Used

**Errors**          An exception may be thrown when this method is invoked.

## DirectIO – Change Paper Low Reporting

Changes the paper low reporting for the K590 kiosk printer.

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### **Parameter Description**

---

*command*    NCRDIO\_PTR\_SET\_PAPER\_LOW

*data*[0]      The new paper low reporting value. This will be one of:

0 - Paper is low if 50 feet or less remains

1 - Paper is low if 10 feet or less remains

2 - Disable paper low reporting

*object*      Not Used

**Errors**      An exception may be thrown when this method is invoked.

Some possible values of the exception's *ErrorCode* property are:

JPOS\_E\_ILLEGAL - The paper low value was invalid, or the printer does not allow the paper low setting to be changed from software.

## DirectIO – Set Black Mark Configuration

Changes the black mark configuration for the K590 kiosk printer. The black mark setting determines how paper cuts are performed relative to black marks on the back of the printer paper.

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### **Parameter Description**

---

*command*    NCRDIO\_PTR\_SET\_BLACK\_MARK

*data*[0]      The new black mark setting. This will be one of:

0 - Document Mode (Ignore black marks when cutting paper)

1 - Normal Mode (Cut at first black mark beyond printing)

2 - Advance Mode (Cut at 2nd black mark beyond printing)

*object*      Not Used

**Errors**      An exception may be thrown when this method is invoked.

Some possible values of the exception's *ErrorCode* property are:

JPOS\_E\_ILLEGAL - The black mark value was invalid, or the printer does not allow the black mark setting to be changed from software.

## DirectIO – Set Minimum Receipt Length

Changes the minimum receipt length for the K590 or F306 kiosk printer.

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

---

### Parameter Description

---

*command*    NCRDIO\_PTR\_SET\_MINIMUM\_REC\_LEN

*data*[0]      The new minimum receipt length.

For the K590, this value is one of:

- |   |       |
|---|-------|
| 1 | 90mm  |
| 2 | 111mm |
| 3 | 150mm |
| 4 | 225mm |
| 5 | 300mm |
| 6 | 375mm |
| 7 | 450mm |
| 8 | 525mm |
| 9 | 600mm |

For the F306, this value is one of:

- |   |          |
|---|----------|
| 0 | None     |
| 1 | 6 inches |

*object*      Not Used



**Errors**      An exception may be thrown when this method is invoked.

Some possible values of the exception's *ErrorCode* property are:

JPOS\_E\_ILLEGAL - The minimum receipt length value was invalid, or the printer does not allow the minimum receipt length to be changed from software.

## DirectIO – Set Receipt Retain Length

Changes the receipt retain length for the K590 kiosk printer.

**Syntax**     **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

---

### Parameter Description

---

*command*    NCRDIO\_PTR\_SET\_REC\_RETAIN\_LEN

*data*[0]        The new receipt retain length. This value is one of:

- |   |      |
|---|------|
| 1 | 25mm |
| 2 | 50mm |
| 3 | 75mm |

*object*        Not Used

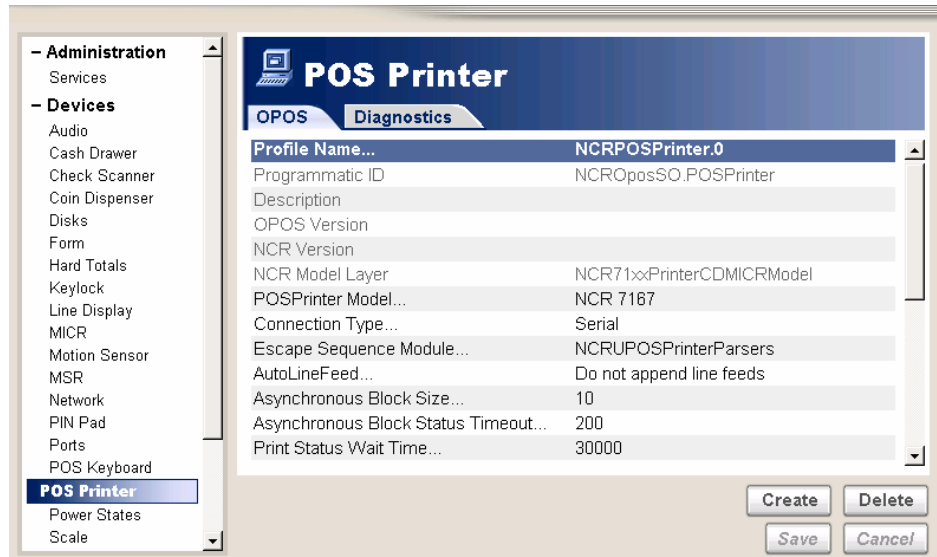
**Errors**        An exception may be thrown when this method is invoked.

Some possible values of the exception's *ErrorCode* property are:

JPOS\_E\_ILLEGAL - The receipt retain length value was invalid, or the printer does not allow the receipt retain length to be changed from software.



## POS Printer Configuration Entries



Parameter	Description	Valid Values
Profile Name	The name of a specific configuration.	NCRPOSPrinter.0 (Any name you wish)
Programmatic ID	This is the Service Object name for the POS Printer	NCROposSO.POSPrinter
NCR Model Layer		NCR71xxPrinterCDMICRModel or NCRK5xxPrinterHydraModel
POSPrinter Model	The model number of the attached printer	7156 7158 7167 7197 7401-K580 7401-K59x 7342-F306

Parameter	Description	Valid Values
Connection Type	Defines how the printer is connected to the terminal.	Serial (Read-Only)
Escape Sequence Module	The POS printer makes use of an escape sequence module to translate from API specific escape sequences to printer specific sequences. The default parser module, UPOSParsers, supports the OPOS/JavaPOS escape sequences.	NCRUPOSPrinterParsers
AutoLineFeed	AutoLineFeed determines if the printer should append a line feed to the last line of a print request, if a line feed is not already present.	Append a line feed if not already present Do Not append line feeds (Default)
Asynchronous Block Size	<p>Asynchronous print requests are grouped into logical units called asynchronous blocks. The Asynchronous Block Size determines the number of requests in each block.</p> <p>The status of the printer is checked at the end of each asynchronous block, rather than after each request. Thus, larger block sizes will reduce the number of status requests issued, and thereby improve performance. However, if a printer failure occurs, you do not know which (if any) requests in the block were printed successfully, and which were not. In this regard, a smaller block size may permit more precise error handling on the part of the application. .</p>	1 or more (Default 10)
Asynchronous Block Status Timeout	<p>The Asynchronous Block Status Timeout value is the maximum time (in milliseconds) to wait before terminating an asynchronous block. This is useful if the application stops sending print requests before an asynchronous block is full; the block will eventually time out, and the status of the requests in the block will be determined and reported to the application.</p> <p>A higher timeout value will usually permit more requests to be combined into each asynchronous block, thereby improving performance. If the timeout value is too large, and the application is not printing lines frequently, then there are longer delays before the status of each request is determined.</p>	50-10000 ms (Default 200 ms)

Parameter	Description	Valid Values
Print Status Wait Time	This is the maximum time to wait for a buffered status response.	5000 - 30000 ms 30000 (Default)
Heart Beat Poll Time	The Heartbeat Poll Time determines how often heartbeat requests are sent to the printer. Heartbeat requests are used to verify that the printer is still responsive. A smaller poll time results in faster detection when the printer goes offline or becomes unresponsive. This comes at the cost of performance, as time spent processing heartbeat requests is time that could have been spent on other print requests.	500 - 10000 ms 1000 (Default)
Java Character Encoding	The POS Printer operates using character sets, not Unicode. As a convenience for the application, the POS Printer supports this parameter to automatically map Unicode strings into a particular character set using the specified Java Character Encoding type string. It is still the responsibility of the application to select the correct character set programmatically. The valid values for this parameter can be found in the Sun Java online documentation at URL: <a href="http://java.sun.com/products/jdk/1.2/docs/guide/internet/encoding.doc.html">http://java.sun.com/products/jdk/1.2/docs/guide/internet/encoding.doc.html</a> . If this parameter is omitted, or its value is an empty string, the application is assumed to be providing strings with character values between 0000 and 00FF where each character is already the correct value for the currently selected character set.	
Port Name	Serial Only. The serial port that is connected to the printer.	COM1 - COM10 COM1 (Default)
Baud Rate	Serial Only. Specifies the speed for serial communications	9600 bps 19200 bps 38400 bps 57600 bps 115200 bps (Default)
Data Bits	Serial Only. The number of bits in the bytes transmitted and received	5 bits 6 bits 7 bits 8 bits (Default)
Parity	Serial Only. Specifies the parity scheme for	None (Default)

Parameter	Description	Valid Values
	serial communications	Even Odd
Stop Bits	Serial Only. The number of stop bits to use for serial communications.	1 bit (Default) 2 bits
EjectSlipFeed (NCR 7167 Only)	EjectSlipFeed determines the number of lines to feed the slip as part of slip removal processing. Negative values indicate a reverse feed, and positive values indicate a forward feed. If this value is left blank, a printer specific default value is used.	-100 - 100 -25 (Default)
SlipInSettleTime (NCR 7167 Only)	Slip insertion will not succeed until you have a slip inserted status that is stable for this many milliseconds. If this value is left blank, a printer specific default value is used.	0 - 10000 ms 500 ms (Default)
SlipOutSettleTime (NCR 7167 Only)	Slip removal will not succeed until we have a slip out status that is stable for this many milliseconds. If this value is left blank, a printer specific default value is used.	0 - 10000 ms 500 ms (Default)
SlipToTopOfForm (NCR 7167 Only)	Indicates if the printer should reposition the slip to the top of form after a slip is inserted	Reposition the slip after insertion (Default) Do not reposition the slip after insertion
ColorPaper	For the 7167 and the 7197 printers, this determines the type of Thermal Color paper being used. It defines the number of colors supported by the paper.	Thermal station contains monochrome paper (Default) Thermal station contains two color paper.
PrintIntegrityMode	Enables or disables print integrity mode. If disabled, receipt printing performance is improved, but there is a chance that some requests are reported as successful even though they actually failed.	Print Integrity mode disabled (Default) Print integrity mode enabled
PrintIntegrityMode	Enables or disables print integrity mode. If disabled, receipt printing performance is improved, but there is a chance that some requests are reported as successful even though they actually failed.	Print Integrity mode disabled (Default) Print integrity mode enabled

## Kiosk Specific Configuration Entries

The following entries apply only to the kiosk printers, i.e. the K580, K590, and 7342-F306. The kiosk printers also use the common entries presented in the table above.

Parameter	Description	Valid Values
Bitmap Load Eject Delay (K580, K590, F306)	Image downloads may cause problems if paper is in the presenter. When downloading an image, first delay for the specified time, eject the receipt, and then download the image.	0 – 15000ms, Default is 3000ms
Auto Eject Delay (K580, K590)	Delay in seconds between time the receipt is cut and when the receipt is automatically ejected.	None (Default) 20 40 60 80
Eject Adjustment Time (K580, K590)	When a print request spans multiple tickets, the default timeout for completion of a status request may not be long enough. For each ticket required, we add the eject adjustment time and the knife cut adjustment time to the default timeout for a status request.	0 – 30000ms Default is 10000ms
Eject On Enable (K580, K590)	If set to true, the printer will eject paper in the presenter at enable time.	Eject on Enable Do Not Eject on Enable (Default)
Ignore Paper In Presenter Errors (K580, K590, F306)	If True, ignore Paper-In-Presenter errors	Ignore Errors (Default for F306) Do Not Ignore Errors (Default for K580, K590)
Knife Cut Adjustment Time (K580, K590)	When a print request spans multiple tickets, the default timeout for completion of a status request may not be long enough. For each ticket required, we add the eject adjustment time and the knife cut adjustment time to the default timeout for a status request.	0 – 30000ms Default is 6000ms
Black Mark Option (K590, F306)	Determines where receipt is cut in relation to black marks on the back of the paper.	Document mode - Printer ignores black marks on paper. This is the default.



Parameter	Description	Valid Values
		Normal mode – Paper is cut at the first black mark on the paper after printing completes
		Advance mode – Paper is cut at the second black mark on the paper after printing completes
Pre-Eject Delay (K580, K590, F306)	Time to delay before attempting an eject to allow presentation of the receipt.	0 – 15000ms Default is 2000ms for K580 Default is 1500ms for K590 Default is 2000ms for F306
Post-Eject Delay (K580, K590, F306)	Time to delay after eject to let the receipt clear the presenter.	3000 – 8000ms Default is 5000ms for K580 Default is 1500ms for K590 Default is 1500ms for F306
Line Size (K590)	Selects the width of the paper in use for this printer.	80mm (Default) 82.5mm 114mm
Minimum Receipt Length (K590, F306)	Minimum amount of receipt paper to present to the user.	K590 values 90mm 111mm 150mm 225mm 300mm 375mm 450mm 525mm 600mm  F306 values None 6 inches
Paper Low Option (K590)	Determines when the printer will report a paper low status.	Report paper low when 50 feet or less paper remains (Default)  Report paper low when 10 feet or less paper remains  Disable paper low reporting

Parameter	Description	Valid Values
Receipt Retain Length (K590)	Determines the portion of each receipt to hold inside the presenter.	25mm 50mm 75mm
Use Soft Options (K590)	If True, the Receipt Retain Length and Minimum Receipt Length registry values will be saved to the printer.	Use Soft Options Do Not Use Soft Options (Default)
Black Mark Distance (F306)	Specifies the distance between black marks on the printer paper.	111mm 127mm 140mm 152mm

## POS Printer Data Capture Configuration Entries

Data capture for the POS Printer is controlled through settings stored in the NCRDatacap.conf configuration file. The following example shows how to enable maximum data capture for all of the POS Printer components.

```
[NCROposSO.POSPrinter]
```

```
LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F
```

```
[NCR71xxPOSPrinterModel]
```

```
LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F
```

```
[NCR71xxPrinterCDMICRModel]
```

```
LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F
```

[NCRK5xxPOSPrinterModel]

LevelMask = 0xFFFFFFFF

TraceMask = 0xFFFFFFFF

UseDefault = F

[NCRK5xxPrinterHydraModel]

LevelMask = 0xFFFFFFFF

TraceMask = 0xFFFFFFFF

UseDefault = F

## Two-Sided Thermal Printing

This section defines the OPOS implementation for the 2-sided thermal (2ST) printer. It defines the NCR proprietary enhancements to the OPOS standard to allow an application to fully control the printing of both sides of the thermal receipt station. This functionality currently supports the 7168-xxxx Two Station Printer.

Two methods of printing on the back side of the receipt paper:

- Proprietary enhancements to the `transactionPrint` method to allow full control on both sides.
- Proprietary escape sequences to control various two-sided thermal printing.

The two-sided thermal print station has four modes of operation:

1. Single Sided Mode
  - The printer only prints on the front side of the paper.
2. Double Sided Mode with Single Side Command (Split Receipt)
  - Normal print operations are buffered at the printer, executed and split between front and back side when a knife cut is encountered.

3. Double Sided Mode with Double Side Command (Full Control)
  - The side desired to print on is selected, and the data to print are sent to it. After the data for both sides are defined, a begin print, or knife cut command is issued to print the two-sided receipt.
4. Double Sided Mode with Predefined Data (Static Reverse Image)
  - Data to print is sent to the printer. A predefined image is selected from previously-defined images. When a print command or knife cut is received, the data is printed on the front side, and the predefined image selected is printed on the back.

**Note:** Refer to *OPOS / JavaPOS Mode Functional Definition* below for a detailed functional explanation on each mode.

### OPOS / JavaPOS Configuration for Selection of Printer-Defined Modes

All four modes of the 2-Sided thermal printing are supported by the OPOS / JavaPOS implementation. The commands and features that are available through OPOS vary with each mode. The selection of the mode to be used is defined by a configuration option, although it can be set dynamically by the use of a proprietary escape sequence.

The printer is initialized to the mode specified in the configuration at **DeviceEnable** time. The application mode can change the mode dynamically during operation, but it reverts back to the configured option if the device is disabled, and later re-enabled.

The following are the configuration options that are available to control the commands and features provided by OPOS / JavaPOS:

Configuration Option	Possible Values
TST2SideMode	0 – Single Sided (Default)
	1 – Double Sided with Single Side Command
	2 – Double Sided with Double Side Command
	3 – Double Sided with Predefined Data
TSTOrientation	0 – Front: Normal; Back: Normal

Configuration Option	Possible Values
	1 – Front: Normal; Back: Upside Down 2 – Front: Upside Down; Back: Normal 3 – Front: Upside Down; Back: Upside Down
TSTPrintSides	0 – Normal (Default) : Front side faces toward front of the printer 1 – Swapped : Front side faces away from the front of the printer
TSTMinRecLen	0 – None (Default), Range: 2-10 Inches
TSTPreDefinedTextFront	0 – Disabled (Default) 1 – Enabled
TSTFrontLine1Text	"" (Default)
TSTFrontLine1Attributes	0 (Default) See <i>Attribute Table</i> below for other values
TSTFrontLine2Text	"" (Default)
TSTFrontLine2Attributes	0 (Default) See <i>Attribute Table</i> below for other values
TSTPreDefinedTextBack	0 – Disabled (Default) 1 – Enabled
TSTBackLine1Text	"" (Default)
TSTBackLine1Attributes	0 (Default) See <i>Attribute Table</i> below for other values
TSTBackLine2Text	"" (Default)
TSTBackLine2Attributes	0 (Default) See <i>Attribute Table</i> below for other values
TSTReprintOnError	0 – Disabled (Default) 1 – Enabled
TSTPreDefReprint	0 – Disabled (Default)

Configuration Option	Possible Values
	1 – Enabled
TSTReprintLine1	"" (Default)
TSTReprintAttrib1	0 (Default) See <i>Attribute Table</i> below for other values
TSTReprintLine2	"" (Default)
TSTReprintAttrib2	0 (Default) See <i>Attribute Table</i> below for other values
TSTEndTransactionAuto Knife	This setting is only applicable if TST2SideMode value is 1 (Double Sided with Single Side Command) or 3 (Double Sided with Predefined Data).  1 – Enabled (Default) : OPOS Service Object issues a knife cut when the TransactionPrint method is called with a control parameter of PTR_TP_NORMAL.  0 – Disabled : No cut is performed and nothing is printed when TransactionPrint method is called with a control parameter of PTR_TP_NORMAL.

**Note:** If predefined text is defined either as **NULL** or Text Data and predefined modes are enabled, **DeviceEnable** may take up to 20 seconds due to loading or clearing of these strings in memory. To enable predefined modes but not load strings, use **-1** in the text field.

#### *Attribute Table for Predefined Text Attribute Values*

Add text attribute values (Hex) together to obtain final value of desired text attribute option.

Predefined Text Attribute Value	Text Attribute
0x1	Double High
0x4	Double Wide
0x10	Emphasize Mode
0x40	1 Dot Underline

Predefined Text Attribute Value	Text Attribute
0x100	Inverse Video
0x400	Italic
0x1000	2 Dot Underline
0x4000	Alternate Color

The following configuration options are not written to the registry by default. They only exist if the user has populated them.

Configuration Option	Possible Values
TSTPreDefinedImage1	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage2	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage3	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage4	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage5	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage6	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage7	Directory name and path of file containing Image Definition commands. "" (Default)

Configuration Option	Possible Values
TSTPreDefinedImage8	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage9	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage10	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage11	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage12	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage13	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage14	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage15	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage16	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage17	Directory name and path of file containing Image Definition commands. "" (Default)



Configuration Option	Possible Values
TSTPreDefinedImage18	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage19	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage20	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage21	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage22	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage23	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage24	Directory name and path of file containing Image Definition commands. "" (Default)
TSTPreDefinedImage25	Directory name and path of file containing Image Definition commands. "" (Default)

**Note:** These files must have the following format:

<u>Macro Select</u>	<u>Define Macro</u>	<u>Print Commands</u>	<u>End Macro Define</u>	<u>Data for Bitmaps</u>
1F 67 xx	1D 3A	various	1D 3A	various

## Escape Sequences to Control the Two-Sided Functionality within a Mode

The following escape sequences are used to control the two-sided functionality through OPOS / JavaPOS. These escape sequences take advantage of the OPOS / JavaPOS “Pass through embedded data” escape sequence. All of these escape sequences must be wrapped by this “Pass through embedded data `ESC|#E`” to be recognized.

The values set by these escape sequences are only valid for the current instance of the OPOS / JavaPOS service. If the device is set to disabled, then the configured settings are restored on the next enable.

The functionality provided by these escape sequences depends on the current `tsT2sideMode` configuration setting. For some settings, they may be ignored and have no effect. All numeric parameters are ASCII characters for numeric values.

Escape Sequence	Functionality	Possible Values
ESC [2ST; 1; 1; n !v	Sets the front side print orientation	n = 0 : No rotation of printing = 1 : 180-degree rotation (upside down)
ESC [2ST; 1; 2; n !v	Sets the back side print orientation	n = 0 : No rotation of printing = 1 : 180-degree rotation (upside down)
ESC [2ST; 1; 3; n !v	Switches Front / Back printing	n = 0 : Normal (front side faces toward front of the printer) = 1 : Swapped (front side faces away from the front of the printer)
ESC [2ST; 1; 4; n !v	Enables / disables predefined Reprint on Error message	n = 0 : Disable = 1 : Enable
ESC [2ST; 2; 1; n !v	Enables / disables predefined Bottom	n = 0 : Disable

Escape Sequence	Functionality	Possible Values
	Front message	= 1 : Enable
ESC [2ST; 2; 2; n !v	Enables / disables predefined Top Back message	n = 0 : Disable = 1 : Enable
ESC [2ST; 2; 3; x; y; z !v	Sets attributes for a predefined Bottom Front / Top Back / Reprint message	x = 0 : Front = 1 : Back = 2 : Reprint  y = 1 : First Line = 2 : Second Line  z = 0x1 : Double High 0x4 : Double Wide 0x10 : Emphasize Mode 0x40 : 1 Dot Underline Mode 0x100 : Inverse Video 0x400 : Italic 0x1000 : 2 Dot Underline Mode 0x4000 : Alternate Color  Adding all attribute values together would equal to 0x5555.  <b>Note:</b> Parameter <b>z</b> is coded as ASCII characters, but they represent a Hex-based number.
ESC [2ST; 2; 4; x; y !v [text]	Sets the string for a predefined Bottom Front / Top Back /	x = 0 : Front = 1 : Back

Escape Sequence	Functionality	Possible Values
	Reprint message	<p>= 2 : Reprint</p> <p>y = 1 : First Line</p> <p>= 2 : Second Line</p> <p>text = ASCII character to define pre-defined lines.</p>
ESC [2ST; 3; n !v [directory/filename]	Activates predefined back side image or loads a specified image.	<p>n = 1-25 : Corresponds to the <b>PreDefinedImageX</b> options. If the <b>PreDefinedImageX</b> option for n does not exist, it is ignored.</p> <p>= 0 : Corresponds to the directory and filename of the predefined image file.</p>
ESC [2ST; 4; n !v [text]	Defines variable print data. It allows a dynamic definition of predefined variables embedded in static images.	<p>n = 1-50 : Corresponds to the variable to define</p> <p>text = ASCII string to assign to the variable. It only allows a maximum of 57 characters. Characters beyond 57 are ignored.</p>

The escape sequences defined above that set the 2ST modes and change the pre-defined text values must be sent in their own Transaction print. They should be sent either before any Receipt is printed or between Receipt Transaction prints. They should not be sent as part of a Receipt. This is because they do not take effect until the next Transaction print.

The only exception is the ESC [2ST; 4; n !v [text] which defines the dynamic data for pre-defined variables in reverse side images. This escape sequence is intended to define text to be printed as part of the current Receipt.

## Special Escape Sequences to Switch between Modes

The following escape sequence is used to dynamically change the way OPOS handles two-sided printing. Because this sequence is changing the mode of operation it is only valid under certain condition. It will set the mode in the same manner as if the `DeviceEnable` had been performed with this mode configured in the `TST2SideMode` option.

Escape Sequence	Functionality	Possible Values
ESC [2ST; 5; n !v	Sets the printer's mode of operation	<p>n = 0 : Single Sided Mode</p> <p>This parameter is only valid immediately after a knife cut, before any other print orientation is sent to the printer. It is ignored if the printer is already processing data.</p> <p>= 1 : Double Sided Mode with Single Side Command</p> <p>If in mode 0, this is valid only if there are no outstanding print operations. If in a two-sided mode, the same restrictions as listed in mode 0 apply.</p> <p>= 2 : Double Sided Mode with Double Side Command</p> <p>If in mode 0, this is valid only if there are no outstanding operations. If in a two-sided mode, the same restrictions in mode 0 apply.</p> <p>= 3 : Double Sided Mode with</p>

Escape Sequence	Functionality	Possible Values
		Predefined Data
		If in mode 0, this is valid only if there are no outstanding operations. If in a two-sided mode, the same restrictions in mode 0 apply.

### Select Print Side Sequence

This special sequence is only valid when mode 2 (Double Sided Mode with Double Side Command) is selected. If any other mode is selected, it is ignored.

This escape sequence is used within a `transactionPrint` to designate which side of the thermal paper, subsequent print operations are performed.

Escape Sequence	Functionality	Possible Values
ESC [2ST; 6; n !v	Selects the side to print data on	n = 0 : Send data to front side of paper  = 1 : Send data to back side of paper

Once this command is issued, all print operations continue to be sent to that side of the paper until the sequence is sent again to change it, or until a `transactionPrint(PTR_TP_NORMAL)` is issued.

After transaction printing has completed, the current side is always reset to the front side of the paper.

## OPOS / JavaPOS Mode Functional Definition

### *Single Sided Mode (TST2SideMode = 0)*

This is the default mode of operation. When this setting is selected, the OPOS / JavaPOS Service sets the printer to this mode at **DeviceEnable** time. None of the other two-sided configuration options is processed. Any two-sided NCR proprietary escape sequences are ignored, except for the escape sequence that switches modes. In this mode, the printer functions the same as all previous thermal receipt printers.

### *Two-Sided Modes (TST2SideMode = 1, 2, 3)*

Once a two-sided mode is entered, the application can **ONLY** print using the OPOS / JavaPOS **transactionPrint** method. The reason for this restriction is due to error-handling. Because the printer has to buffer all data before printing, there is no way to determine errors on a line level. Due to this buffering, errors can only be recovered at a transaction level. Therefore we enforce the use of transaction printing only while in a two-sided mode.

While a two-sided mode is active, any print operation attempted outside of a transaction print results in an **OPOS\_E\_ILLEGAL** error.

The only exception is a **PrintNormal** containing only the escape sequence to switch to another two-sided mode, or to switch to single-sided mode.

Two-Sided Modes	Functionality
TST2SideMode = 1 (Double Sided Mode with Single Side Command)	<p>When this mode of operation is selected, the printer buffers all print commands sent to it until a paper cut command is issued. At that time, it splits the receipt and prints it between the front and back.</p> <p>All of the two-sided escape sequences and settings are valid <b>except</b> for those that deal with a predefined back side image, or selecting a specific print side.</p>
TST2SideMode = 2 (Double Sided Mode with	This mode of operation is the one that gives the application the most flexibility in two-sided

## Two-Sided Modes

## Double Side Command)

## Functionality

printing. It allows the application to define and print anything they want on both sides of the paper dynamically.

See *Select Print Side Sequence* for special ESC sequence only available in this mode.

## TST2SideMode = 3

(Double Sided Mode with  
Predefined Data)

As with all of the **TST2SideMode** settings, this mode is initiated at **DeviceEnable** time, or with the select mode escape sequence. At enable time, the Service Object loads the predefined image file from the configuration setting of **PreDefinedImageX** and enables the two-sided mode. If this mode is selected but the **TSTPreDefinedImage1** is not populated, any existing predefined image in the printer is cleared. All of the two-sided escape sequences are valid for this mode of operation **except** for the Select Print Side sequence.

The **ESC [2ST; 3; n !v [directory / filename]** activates the predefined back side image and can be used to change the image that is being used for the back side of the paper. If this sequence is sent with **n = 0** but no **[directory / filename]** parameter, it is assumed that the user wants to erase the existing predefined image and clears it.

The above sequence is ONLY VALID after a knife cut, and before any other print operation is sent to the printer. If any other print operation is sent to the printer before this sequence, then it is ignored.



# Scale

The Scale supports many NCR Scanner/Scale devices. See the supported device table for the Scale models supported.

NCR USB connection type is for use with USB serial emulation driver. The “Uses” parameter must be set with the scanner profile whose connection type must be SERIAL.

USB connection type is a HID USB connection and is supported for NCR 7883, 7872 and 7876 Release 2 scanner/scales.

## DirectIO - Scale Status

Syntax     **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

<b>Parameter Description</b>	
<i>command</i>	NCRDIO_SCAL_STATUS (601).
<i>data</i>	Not Used.
<i>object</i>	Returns the status string direct from the scale. You must know the scale interface to decode the string.

## DirectIO - Scale Read ROM

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### **Parameter Description**

---

*command*    NCRDIO\_SCAL\_READROM (602).

*data*            Address to read from.

*object*        Returns 30 bytes of ROM data.

## DirectIO - Scale ROM Version

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### **Parameter Description**

---

*command*    NCRDIO\_SCAL\_ROM\_VERSION (603).

*data*        Not Used.

*object*      Returns the ROM version number.

## DirectIO - Scale Live Weight

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### **Parameter Description**

---

*command*    NCRDIO\_SCAL\_LIVE\_WEIGHT (604).

*data*        Returns the weight prodivde by the low level "monitor" scale firmware command.

*object*      Not Used.

**Errors**      An exception may be thrown when this method is invoked.  
  
Some possible values of the exception's *ErrorCode* property are:

<b>Value</b>	<b>Meaning</b>
JPOS_E_FAILURE	The device cannot perform the requested procedure, even though the device is connected to the system, powered on, and on-line.
JPOS_E_TIMEOUT	A stable non-zero weight was not available before <i>Timeout</i> milliseconds elapsed.
JPOS_E_EXTENDED	<b>ResultCodeExtended =</b> JPOS_ESCAL_OVERWEIGHT: The weight was over <b>MaximumWeight</b> . NCR_ESCAL_UNSTABLE: The scale reading is not stable. NCR_ESCAL_ZEROWEIGHT: The scale is not registering a weight. NCR_ESCAL_UNDERZERO: The scale is reading less than zero weight.



## DirectIO - Scale Direct Access

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### **Parameter Description**

---

*command*    NCRDIO\_SCAL\_DIRECT (605).

*data*        False (0), no data is returned.

*object*      String to send to the scale, including the suffix and BCC character (if needed) or the string data returned from the scale.

## Method Updates

### ReadWeight Method *Updated in UPOS Release 1.9*

Syntax     **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

Parameter	Description
<i>command</i>	<b>ReadWeight</b>
<i>data</i>	If <b>AsyncMode</b> is FALSE, Points to the number where the weight is returned; else must be zero..
<i>object</i>	The number of milliseconds to wait for a settled weight before failing the method. If zero, the method attempts to read the scale weight, then returns the appropriate status immediately. If JPOS_FOREVER (-1), the method waits as long as needed until a weight is successfully read or an error occurs.

Remarks    Call to read a weight from the scale.

#### **Release 1.0 – 1.2**

The weighing process is performed synchronously and the method will return after finishing the weighing process.  
The weight is returned at *pWeightData*,

#### **Release 1.3 and later**

If **AsyncMode** is FALSE, then **ReadWeight** operates synchronously, as with earlier releases.

If **AsyncMode** is TRUE, the weighing process is performed asynchronously. The method will initiate a read, then return immediately. If the method returns a success status, the weighing process is started and a **DataEvent** containing the weight in its *Status* parameter indicates its completion.

The weight has an assumed decimal place located after the “thousands” digit position. For example, an actual value of 12345 represents 12.345, and an actual value of 5 represents 0.005.

**Errors** An exception may be thrown when this method is invoked.

Some possible values of the exception's *ErrorCode* property are:

Value	Meaning
E_TIMEOUT	A stable non-zero weight was not available before <i>timeout</i> milliseconds elapsed (only if <b>AsyncMode</b> is false).
E_EXTENDED	<i>ErrorCodeExtended</i> = ESCAL_OVERWEIGHT: The weight was over <b>MaximumWeight</b> . This can only be returned if <b>AsyncMode</b> is false.

The following standard extended error codes have been **added in Release 1.9** as possible values of the exception's *ErrorCode* property: E\_EXTENDED  
*ErrorCodeExtended* =  
ESCAL\_UNDER\_ZERO: The scale is reporting a weight that is less than zero due to a calibration error. The scale should be recalibrated.

This can only be returned if **AsyncMode** is false. E\_EXTENDED  
*ErrorCodeExtended* = ESCAL\_SAME\_WEIGHT: The scale is reporting that the item/weight on the scale



is identical to the previously reported Item/weight; i.e., the item has not been removed from the scale. This can only be returned if **AsyncMode** is false and the scale hardware directly supports this capability.

### zeroScale Method *Updated in UPOS Release 1.9*

**Syntax**     **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

Parameter	Description
-----------	-------------

<i>command</i>	<b>zeroScale</b>
----------------	------------------

**Remarks**     If **CapZeroScale** is true, sets the current scale weight to zero. It may be used for initial calibration, or to account for tare weight for the scale.

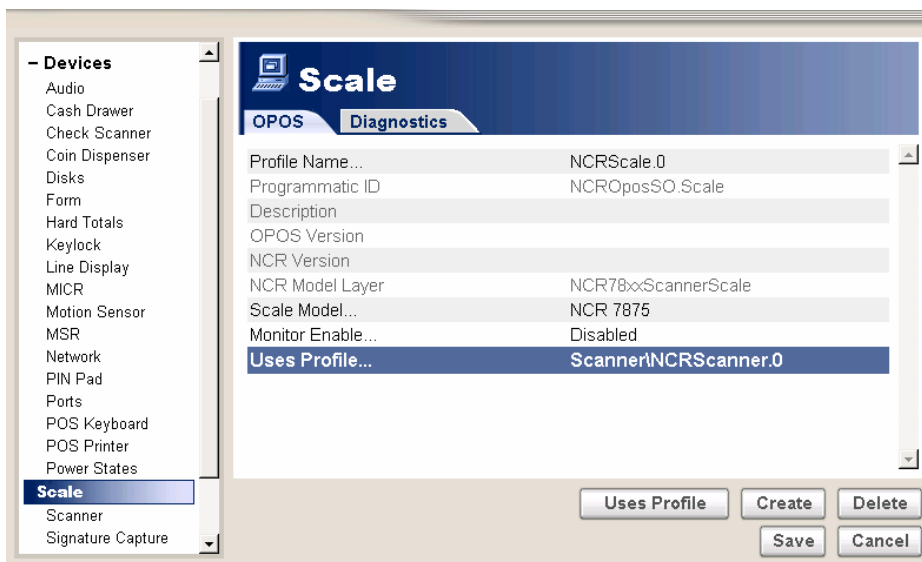
**Errors**        An exception may be thrown when this method is invoked.  
  
Some possible values of the exception's *ErrorCode* property are:

Value	Meaning
-------	---------

E_ILLEGAL	<b>CapZeroScale</b> is false.
-----------	-------------------------------

E_TIMEOUT	<b>CapZeroScale</b> is true but the command failed in executing the operation with the device.
-----------	--

## Scale Configuration Entries



Parameter	Description	Valid Values
Profile Name	The name of a specific configuration.	NCRScale.0 (Any name you wish)
Programmatic ID	This is the Service Object name for the Scale	NCROposSO.Scale
NCR Model Layer		NCR78xxScannerScale
ScaleModel	Model number of the device	NCR 7872 NCR 7875 (Default) NCR 7876 NCR 7883
MonitorEnable	Enables the ReadWeight method to return a continuous weight when a stable weight is available. Disable returns only a NEW weight.	Disabled (Default) Enabled - Return continuous weight. Application is responsible for assuring a stable legal weight.
Metric (USB only)	Selects the scale metric (kg) weight interface when enabled. Selects the English (lbs) weight interface when disabled. Must match the scale setting.	Disabled (Default) Pounds mode Enabled – Kilogram mode. “FiveDigit” weight must be enabled when in Kilogram mode.
FiveDigit (USB only)	Selects the number of digits returned from a weight request. When disabled, 4 digits of weight is returned with a accuracy to the	Disabled (Default) 4 Digit Enabled – 5 Digit “FiveDigit” must be enabled when

hundredths. When Enabled the Weight returned shall be 5 digits with an accuracy of thousandths.

“Metric” is enabled. 4 Digit is not support in Kilogram mode.

Uses (Serial only)

For Serial and NCR USB connections the Scale uses the parameters of the Scanner service object in addition to the parameters described here. Therefore, the Scanner parameter entries (using the NCROposS).Scanner Programmatic ID) must be set up first.

For a USB connection the “Uses” parameter is not needed and shall be empty.

The profile name of the Scanner for Serial and NCR USB.

None (blank) for USB.

## Scale Data Capture Configuration Entries

Data capture for the Motion Sensor is controlled through settings stored in the NCRDatacap.conf configuration file. The following example shows how to enable maximum data capture for all of the Scale components.

### [NCROposSO.Scale]

LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F

### [NCRScale]

LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F

### [NCRScannerHydra]

LevelMask = 0xFFFFFFFF  
TraceMask = 0xFFFFFFFF  
UseDefault = F



# Scanner

The Scanner supports many NCR Scanner devices. See the supported device table for the Scanner models supported.

DirectIO - Scanner Direct Access and DirectIO - Scanner ROM Version only is supported on models NCR 7837-xxxx, NCR 7838, 3800, 4600, 5600 and 5620.

No DirectIO is supported for model NCR 7837.

NCR USB connection type is for use with USB serial emulation driver.

USB connection type is a HID USB connection and is supported for NCR 7883 and 7892 Release 2 scanners and 7872 and 7876 Release 2 scanner/scales.

## DirectIO - Scanner NOT-ON-FILE

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

---

### Parameter Description

*command*    NCRDIO\_SCAN\_NOT\_ON\_FILE (508)

*data*        Not Used.

*object*      Not Used.

## DirectIO - Scanner Tone Control

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### **Parameter Description**

---

*command*    NCRDIO\_SCAN\_TONE (501)

*data*        SCAN\_TONE\_BEEP (1001)  
              SCAN\_TONE\_ENABLE (1002)  
              SCAN\_TONE\_DISABLE (1003)

*object*      Not Used.

## DirectIO - Scanner Reset

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### **Parameter Description**

---

*command*    NCRDIO\_SCAN\_RESET (502)

*data*        Not Used.

*object*      Not Used.

## DirectIO - Scanner Status

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### **Parameter Description**

---

*command*    NCRDIO\_SCAN\_STATUS (503)

*data*        Not Used.

*object*      Status string (direct from scanner). You must know the scanner interface to decode the string.



## DirectIO - Scanner Read ROM

Syntax     **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### **Parameter Description**

---

*command*    NCRDIO\_SCAN\_READROM (504)

*data*        Address to read from.

*object*      Returns 30 bytes of ROM data.

## DirectIO - Scanner ROM Version

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### **Parameter Description**

---

*command*    NCRDIO\_SCAN\_ROM\_VERSION (505)

*data*        Not Used.

*object*      Returns the ROM version ID.

# DirectIO - Scanner Pacesetter Options

Syntax     **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

**Parameter Description**

---

*command*    NCRDIO\_SCAN\_PACESETTER (506)

*data*               Requested sub-command  
                 PACESETTER\_READ\_GOOD (1101)  
                 PACESETTER\_READ\_NO\_READS (1102)  
                 PACESETTER\_READ\_OVERPRINT (1103)  
                 PACESETTER\_READ\_UNDERPRINT (1104)  
                 PACESETTER\_READ\_MISS\_MARGIN (1105)  
                 PACESETTER\_RESET (1106)  
                 PACESETTER\_ENABLE\_MODE\_3 (1107)  
                 PACESETTER\_DISABLE\_MODE\_3 (1108)

*object*        Not Used.

If the *command* is successful, and a READ sub-command (1101-1105) was issued, the tally value is placed in *data*.

**NOTE:** The Pacesetter command is only supported on the NCR 7875, scanner.

## DirectIO - Scanner Direct Access

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### **Parameter Description**

---

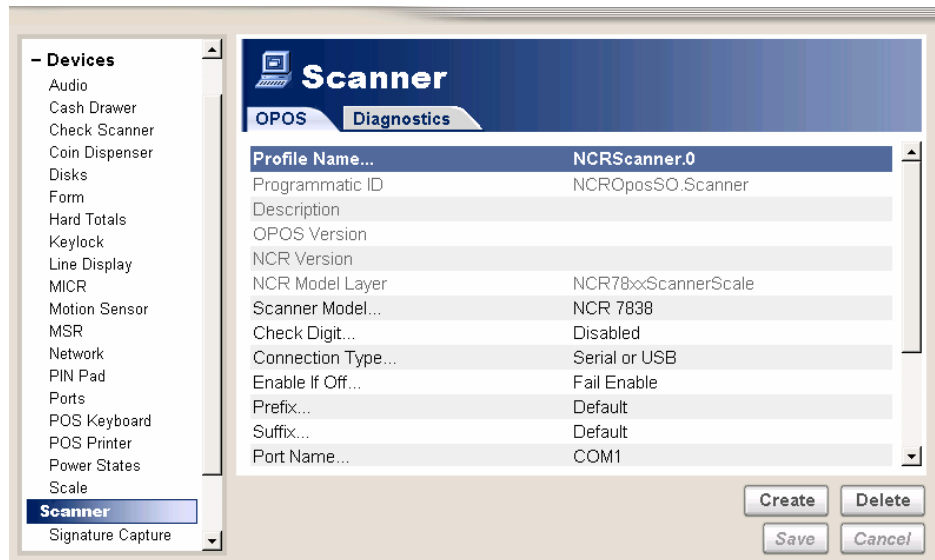
*command*    NCRDIO\_SCAN\_DIRECT (507)

*data*        False (0), no data is returned.

*object*      String to send to the scanner, including the suffix and BCC character (if needed) or the string data returned from the scanner.

*Note: For models NCR 7837-xxxx and 7838, the string should be the command string without the Menu command prefix and the command suffix. Both are added by the service object..*

## Scanner Configuration Entries



Parameters	Description	Valid Values
Profile Name	The name of a specific configuration.	NCRScanner.0 (Any name you wish)
Programmatic ID	This is the Service Object name for the Scale	NCROposSO.Scanner
NCR Model Layer		NCR78xxScannerScale
Scanner Model	Model number of the Scanner connected to the terminal.	NCR 7832 NCR 7837 NCR 7837-1300 NCR 7837-3000 NCR 7838 NCR 7838-SS NCR 7872 NCR 7875 NCR 7876 NCR 7880 NCR 7882 NCR 7883 NCR 7892 (Default)
	NCR 7838-SS indicates 7838 Self Service scanner for FastLane and Kiosk applications.	3800 4600 5600

Parameters	Description	Valid Values
Connection Type	Describes how the scanner is connected to the terminal. For NCR USB connection type is SERIAL. For HID USB connection type is USB.	5620 Other Serial (Default) USB Wedge
EnableIfOff	EnableIfOff value of 1 causes the model to succeed DeviceEnable when the device is powered off. This is for models that support power reporting and automatically initialize and enable the device when power is applied.	Fail Enable (Default) Succeed Enable
Prefix	Must match the device setting. .Prefix, prepends a byte or 2 to the device message. Prefix is not included in BCC calculation.	None (Default)
Suffix	Must match the device setting. Suffix prepends a byte or 2 to the device message. Suffix is included in the BCC calculation.	0x03 (Default)
InitDevice	Determines when device initialization occurs. Device initialization occurs at Device Enable but for better performance, initialization shall occur only at the FIRST Device Enable after either Open or Claim.  Set to a value of 1 shall initialize device at the first device enable after Open.  The default behavior shall initialize the device on the first device enable after Claim.  The default value is zero.	0x00 (Default)
<b>NCR 7872, 7875, 7876, 7883 Common Properties</b>		
MessageFormat	MessageFormat applies to scanner/scale devices only.. ScannerScale devices default to a message format of ScannerScale. Re-programming a scannerscale device to scanner-only mode will default to a message format of Scanner Only.	Scanner Only (Default 7832, 7837 7837-xxxx, 7838, 7883, 7892) ScannerScale (Default 7872, 7875, 7876)
Bcc	Block Check Characters(BCC) - Exclusive OR of all bytes in a message	Disabled (Default 7832, 7837, 7838, 7883, 7892)

Parameters	Description	Valid Values
	except the prefix byte. Default is Enabled for scanner/scale devices. Default is Disabled for scanner/scale devices re-programmed to Scanner Only.	Enabled (Default 7872, 7875, 7876)
RemoveSpaces	Applicable only when “CheckDigit” is “Enabled”. Enables removal of spaces within a barcode. “Remove” sets “RemoveSpaces” config file parameter to 1.	Do not remove (Default) Remove
CheckDigit	Applicable only to barcodes that support a check digit. Enables calculation of a check digit for barcodes received from the scanner without a check digit. “Enabled” sets “AutoCD” config file parameter to 1.	Disabled (Default) Enabled

Parameters	Description	Valid Values
<b>HID USB Parameters for NCR 7872, 7876, 7883 and Scanner-Only NCR Scanner NCR 7892</b>		
VendorID	Vendor ID of the HID device.	0x404 (Default)
ProductID	Product ID of the HID device.	0x200 (Default)
DevUsagePage	Device Usage Page of the HID device.	0xFF45 (Default)
DevUsage	Device Usage of the HID device.	0x4A00 (Default)
Reports	Number of reports of the HID device.	2 (Default)
ClearAfterRead	Clear buffer after read.	true (Default)
Config	Configuration 9 bytes in hex, 2 hex digits per byte.  Example: 0108004B3B02000015	"" (Default)
LabelID	Enable use of NCR specific symbology IDs. Enable this parameter if an application decodes the symbology ID characters within the scan data. Enabling uses the NCR symbology IDs in the scan data. If UPOS property ScanDataType is used, leave this parameter disabled.	Disabled (Default) Enabled
RSS	Enable RSS barcodes to be read.  Enable RSS/14.  Enable RSS Expanded.  Enable both.	Disabled (Default)



Parameters	Description	Valid Values
<b>Serial Port Parameters for NCR 7872, 7875, 7876, 7880, 7883 and Scanner-Only NCR Scanners, NCR 7882, NCR 7892</b>		

Port Name	Serial port name connected to the device. USB devices using IO Network drivers must assign the port used by the driver	COM1 - COM10 COM1 (Default)
Baud Rate	The communication speed for the Scanner. The setting must match the device setting	9600 bps (Default) 4800 bps
Data Bits	The number of bits in the bytes transmitted and received	5 bits 6 bits 7 bits (Default) 8 bits
Parity	Serial Only. Specifies the parity scheme for serial communications	None Even Odd (Default)
Stop Bits	Serial Only. The number of stop bits to use for serial communications.	1 bit (Default) 2 bits
CTS Flow Control	Permit serial flow control through the Clear to Send (CTS) signal.	Disabled Enabled (Default)
DSR Flow Control	Permit serial flow control through the Data Set Ready (DSR) signal.	Disabled Enabled (Default)

**NCR 7880,  
7882 and 7892 Common Properties**

Bcc	Block Check Characters(BCC) - Exclusive OR of all bytes in a message except the prefix byte.	Disabled (Disabled) Enabled
Proqgramming Sequence	For models that support programming sequences. The string is sent to the device verbatim during initialization.	

Parameters	Description	Valid Values
<b>NCR 7837 Common Properties</b>		
Port Name	Serial port name connected to the device. USB devices using IO Network drivers must assign the port used by the driver	COM1 - COM10, COM1 (Default)
Serial Port Properties for Input-Only (7837) Scanners		
NCR Version	NCR Release File Version of the Service Object.	ASCII Text
Baud Rate	The communication speed for the Scanner. The setting must match the device setting	9600 bps (Default) 4800 bps
Data Bits	The number of bits in the bytes transmitted and received	5 bits 6 bits 7 bits (Default) 8 bits
Parity	Serial Only. Specifies the parity scheme for serial communications	None Even (Default) Odd
Stop Bits	Serial Only. The number of stop bits to use for serial communications.	1 bit (Default) 2 bits
CTS Flow Control (7837 Only)	Permit serial flow control through the Clear to Send (CTS) signal.	Disabled (Default) Enabled
DSR Flow Control (7837 Only)	Permit serial flow control through the Data Set Ready (DSR) signal.	Disabled (Default) Enabled

Parameters	Description	Valid Values
<b>NCR 7837-xxxx, 7838, 2357-xxxx Common Properties</b>		
Port Name	Serial port name connected to the device. USB devices using IO Network drivers must assign the port used by the driver	COM1 – COM10, COM1 (Default)
NCR Version	NCR Release File Version of the Service Object.	ASCII Text
Baud Rate	The communication speed for the Scanner. The setting must match the device setting	38400 bps (Default) 115200 bps (Default 7838-SS)
Data Bits	The number of bits in the bytes transmitted and received	5 bits 6 bits 7 bits 8 bits(Default)
Parity	Serial Only. Specifies the parity scheme for serial communications	None(Default) Even Odd
Stop Bits	Serial Only. The number of stop bits to use for serial communications.	1 bit (Default) 2 bits
CTS Flow Control (7837 Only)	Permit serial flow control through the Clear to Send (CTS) signal.	Disabled (Default) Enabled
DSR Flow Control (7837 Only)	Permit serial flow control through the Data Set Ready (DSR) signal.	Disabled (Default) Enabled
AimerLED	Scanner Aimer LEDs brightness control.	100 (Default) 0, 50%, 100%
LightLED	Illumination LEDs control. Sets LEDs either ON or OFF while reading.	?? (Default), Disabled, Enabled Disabled = OFF Value of “??” means DO NOT MODIFY. (*See Note)
AllSymbologies	Send command to enable all supported barcode symbologies.	Disabled (Default), Enabled
ReadTimeout	Manual mode timeout from enable to when the LEDs go off. Value 0 sets the timeout to infinite.	?? (Default) (*See Note) Range: 0-300000 (ms) Value of “??” means DO NOT MODIFY.
TriggerMode	Trigger mode for starting a read Manual or Presentation mode.  Manual requires scanner trigger press.  Presentation detects barcode in scanner view.	MANUAL (Default) MANUAL - trigger mode 0 PRESENTATION - trigger mode 3

Parameters	Description	Valid Values
AuxLED	Auxiliary LEDs present in 7838-SS only.	Default (“??”)(*See Note) 0 = Off 1 = On 2 = Follow Trigger 3 = Follow LightLED (Default 7838-SS)
ScannerScaleACK	Use to decrease time to enable/disable by not waiting for device ACK to command. Default is wait for all ACKs.	0 (Default) Value 1 – Ignore disable ACK. Value 2 – Ignore enable ACK Value 3 – Ignore enable and disable ACK.

**\*Note: For 7837 and 7838 scanners, value “??” indicates this parameter will not be configured and the setting in the scanner shall be used. This improves performance by not initializing parameters not configured.**

#### Other Common Properties

Port Name	Serial port name connected to the device. USB devices using IO Network drivers must assign the port used by the driver	COM1 - COM10, COM1 (Default)
Serial Port Properties for Input-Only (Other) Scanners		
NCR Version	NCR Release File Version of the Service Object.	ASCII Text
Baud Rate	The communication speed for the Scanner. The setting must match the device setting	9600 bps (Default) 4800 bps
Data Bits	The number of bits in the bytes transmitted and received	5 bits 6 bits 7 bits 8 bits (Default)
Parity	Serial Only. Specifies the parity scheme for serial communications	None (Default) Even Odd
Stop Bits	Serial Only. The number of stop bits to use for serial communications.	1 bit (Default) 2 bits
CTS Flow Control (7837 Only)	Permit serial flow control through the Clear to Send (CTS) signal.	Disabled (Default) Enabled
DSR Flow Control (7837 Only)	Permit serial flow control through the Data Set Ready (DSR) signal.	Disabled (Default) Enabled

#### Serial Port Properties for NCR 7832 Scanners

Parameters	Description	Valid Values
Baud Rate	The communication speed for the Scanner. The setting must match the device setting	9600 bps (Default) 4800 bps
Data Bits	The number of bits in the bytes transmitted and received	5 bits 6 bits 7 bits 8 bits (Default)
Parity	Serial Only. Specifies the parity scheme for serial communications	None (Default) Even Odd
Stop Bits	Serial Only. The number of stop bits to use for serial communications.	1 bit 2 bits (Default)

#### Wedge Properties

Connection Type	Indicates the subcomponent on the Wedge bus to use.	Serial (Default)
-----------------	---	------------------

#### Default values based on Model setting:

Model	Baud	Parity	Byte Size	Stop Bits	Prefix	Suffix	CTS *	DSR *	DTR *	RTS *
7832	9600	None	8	2	None	0x0D	--	--	--	--
7837	9600	Even	7	1	None	0x0D	--	--	--	--
7837-1300	38400	None	8	1	Symbology ID	0x0D	--	--	--	--
7837-3000	38400	None	8	1	Symbology ID	0x0D	--	--	--	--
7838, 3800, 4600, 5600, 5620	38400	None	8	1	Symbology ID	0x0D	--	--	--	--
7838-SS	115200	None	8	1	Symbology ID	0x0D				
7872	9600	Odd	7	1	None	0x03	On	On	--	--
7875	9600	Odd	7	1	None	0x03	On	On	--	--
7876	9600	Odd	7	1	None	0x03	On	On		
7880	9600	Odd	7	1	None	0x03	--	--		
7882	9600	Odd	7	1	None	0x03	--	--	--	--

7883	9600	Odd	7	1	None	0x03	On	On		
7892	9600	Odd	7	1	None	0x03	--	--	--	--
Other	9600	None	8	1	None	None	Off	Off	On	On

- \*Signifies that these parameter are not configurable. These settings are provided for informational purposes only, and this information is targeted for those individuals using the "other" model setting. For the NCR family of scanners, these settings are known by the Scanner / Scale Service

## Scanner Data Capture Configuration Entries

Data capture for the Scanner is controlled through settings stored in the NCRDatacap.conf configuration file. The following example shows how to enable maximum data capture for all of the Scanner components.

```
[NCROposSO.Scanner]
```

```
LevelMask = 0xFFFFFFFF
```

```
TraceMask = 0xFFFFFFFF
```

```
UseDefault = F
```

```
[NCRScanner]
```

```
LevelMask = 0xFFFFFFFF
```

```
TraceMask = 0xFFFFFFFF
```

```
UseDefault = F
```

```
[NCRScannerHydra]
```

```
LevelMask = 0xFFFFFFFF
```

```
TraceMask = 0xFFFFFFFF
```

```
UseDefault = F
```

## Scanner Frequently Asked Questions

### 7837-3000 and 7838 Scanner setup for UPOS

The 7838 User's Guide is missing the second page of the Programming Chart (TAR 295385). The programming chart is needed for program sequences in setting the scanner. The 7837 User's Guide has the programming charts.

#### **[Suffix]:**

The default UPOS profile has suffix set to CR (0x0D) for all communication with the scanner. To set the scanner to this suffix see page 4-3 of the User's Guide.

#### **[Prefix]:**

The default UPOS profile has a prefix of none. The scanner default is none.

#### **[Barcode Symbology ID]:**

NCR UPOS will provide barcode decoding of the scan data type of the scanner Code ID symbology IDs when the UPOS property DecodeData is set TRUE.

The scanner User's Guide Page 4-3 starts the description.

To program for use with UPOS DecodeData property = TRUE:

1. Scan Clear All Prefixes barcode
2. Scan Add Prefix, scan the character barcode in appendix for the following "995C80", Save.

To program for use with UPOS DecodeData property = FALSE and set for NCR symbology ID:

1. Scan Clear All Prefixes barcode
2. Scan Add Prefix, follow the instructions for adding a specific prefix to a specific symbology, Save.

#### **[PDF417 Barcodes]:**

PDF417 barcodes are supported by the scanner and NCR UPOS. There could be suffix configuration issues depending on whether non-printable characters appear in the PDF417 tag. If non-printable characters are present, configuration of the scanner suffix must not match any of the character combinations. NCR UPOS does support 2 character suffix combinations.

#### **[RS-232]:**

Scan the configuration barcode on page 2-9 to select RS-232 terminal interface. This also adds a CR-LF suffix. You must match the suffix with the

UPOS profile settings. The default UPOS profile provided by NCR has a CR (0x0D) as the suffix. To set the suffix to CR only see the note above.

**[NCR USB]:**

To connect the scanner via USB, a virtual COM port driver must be installed. This driver allows the scanner to communicate via USB but provides a virtual serial COM port for NCR UPOS to connect internally to the USB port. The links below are for downloading the required driver.

To configure the scanner scan the configuration barcode at the top of page 1-11 of the scanner User's Guide for "USB COM Port Emulation". When you plug in the scanner to the USB port, the OS will request a driver for the device. Point the OS to the location of the USB COM port emulation driver INF file to have the OS install the driver. When the install is finished a COM port will have been added to the terminal devices. Update the NCR UPOS profile to this COM port.

[http://www.ncr.com/en/support/support\\_drivers\\_patches.asp](http://www.ncr.com/en/support/support_drivers_patches.asp)

Example driver file 35354004.zip, 05/17/2004, 100 KB (USB Serial Emulator Driver for Win98 1.10.000, Win 2000 & XP 1.10.003)

### Money Center Integrated Self Service Scanner

The Integrated Self Service Scanner 497-0438286 appears in RSM as NCR 7838 Self Service scanner.

### 7401/7402 Kiosk Integrated Scanner

The Kiosk Integrated Scanner model in RSM should be NCR 7892.

- Example NCD7492 scanner model: NCR 7892-0908

### HID USB Scanner Configuration

Only Release 2 (SA) scanners and scanner/scales support HID USB connection type to NCR UPOS.

No Release 1 scanners support HID USB. Release 2 scanners a model number for the scanner with a 2 in the second digit of the model version number.



Example Release 2 scanner model number: 7872 - 1298 has a model version 1298 with a '2' in the second position indicating Release 2.

#### [Scanner Configuration Program Sequences]:

The scanner must be configured to use HID USB as the communications setting. Program Mode (PM) followed by the hex digits shown and followed by Save and Reset (S&R).

#### **PM 1 0 D S&R**

The scanner must be configured to use HID USB ASCII data format.

#### **PM 3 2 F S&R**

#### [Scanner Configuration Bytes]:

The Configure Scanner bytes can be set in the scanner profile “Config” entry. The table below shows the meaning for the configuration bytes NCR specific restrictions.

See the descriptions below noting the *NCR Specific* restrictions.

**Byte 0 NCR Specific:** All bits supported except for Codabar and UPC D1-D5. The scanner will take either value of 1 or 2 for bits 0-1 and just turn UPC on.

<b>BIT(s)</b>	<b>Definition</b>
0-1	0 = Invalid 1 = UPC A/UPC E,EAN 8/13 enable 2 = UPC A/UPC E,EAN 8/13, UPC D1..D5 enable 3 = Invalid
2-4	0 = Disable 2 and 5 digit periodical add-on 1 = Enable +2 add-ons optional 2 = Enable +5 add-ons optional 3 = Invalid 4 = Enable +2 and +5 optional 5, 6, 7 = Invalid
5	Code 39 enable
6	ITF enable

7	Reserved for Codabar
---	----------------------

**Byte 1 NCR Specific:** Code 93 or the ability to enable/disable scanner programming via barcodes shall not be supported. All other bits are supported.

<b>BIT(s)</b>	<b>Byte 1 - Decode and Programming</b>
0	Reserved for Code 93
1	Code 128
2	UPC A to EAN-13 expansion
3	UPC E to EAN 13 expansion
4	UPC E to UPC A expansion
5	Verify 4 digit price check
6	Enable scanner programming via barcodes
7	Verify 5 digit price check

**Byte 2 NCR Specific:** All bits are supported.

<b>BIT(s)</b>	<b>Byte 2 - Interleaved 2 of 5, length 1</b>	
0-5	ITF length #1	Even value 4-32 inclusive
6	Not defined, must be zero	
7	Configure two ITF lengths	

**Byte 3 NCR Specific:** All bits are supported.

<b>BIT(s)</b>	<b>Byte 3 - Beeper Control</b>
0	Enable good read beep
1-2	Beeper duration 0-4 valid Short -- Long
3-5	Beeper volume 0-3 low -- high 4-7 invalid
6-7	Beeper frequency 0-4 valid low -- high

**Byte 4 NCR Specific:** All bits are supported.

<b>BIT(s)</b>	<b>Byte 4 - Interleaved 2 of 5, length 2</b>
0-5	Interleaved 2 of 5 length #2
6	Not defined, must be zero

7	Not defined, must be zero
---	---------------------------

**Byte 5 AND Byte 6 NCR Specific:** All bits are NOT supported for minimum scans per read.

**Byte 7 NCR Specific:** Laser and motor are tied together in hardware. They have to be the same values. The values for motor/laser timeouts and the double read timeout (TBI) shall be supported, the values in this config command have been altered to match the times valid for the NCR scanner. In other words, you could set the TBI (double read) timeout to be 700 ms, but we make it 750 because that's what our H/W can do.

The Scanner does NOT support EAN/JAN-13 Two Label Decoding with this command.

BIT(s)	Byte 7 - Timeouts
0-2	Motor Timeout 0 = 60 minutes 1 = 5 minutes 2 = 10 minutes 3 = 15 minutes 4 = 30 minutes 5 = 60 minutes 6 = invalid 7 = invalid
3-4	Laser Timeout 0 = 15 minutes 1 = 5 minutes 2 = 10 minutes 3 = 15 minutes
5-6	Double Read Timeout 0 = 500 ms 1 = 700 ms 2 = 900 ms 3 = invalid
7	Enable EAN/JAN-13 Two Label Decoding

**Byte 8 NCR Specific:** All bits are NOT supported.

BIT(s)	Byte 8 - Miscellaneous Controls
0-1	LED Good Read Duration 0 = 0.50 seconds 1 = 0.75 seconds 2 = 1.00 seconds 3 = invalid
2-3	Reserved for Universal Scans per Read
4	Label Buffering 0 = buffer one label 1 = buffer two labels
5	Enable/Disable switch controlled beep volume adjustment 0 = disabled 1 = enabled
6	Reserved for Enable Laser ON/OFF switch
7	Two Additional configuration bytes follow

## NCR 2357-xxxx( 3800, 4600, 5600 and 5620) Scanner setup for UPOS

The User's Guide for these models can be found on the NCR website using the NCR 2357 model number. The User's Guide has the programming charts needed for changing scanner setup as noted below.

NCR 2357 – 1000	3800 scanner
NCR 2357 – 1300	5600 scanner
NCR 2357 – 2000	4600 scanner
NCR 2357 – 3000	5620 scanner

### **[Suffix]:**

The default UPOS profile has suffix set to CR (0x0D) for all communication with the scanner. To set the scanner to this suffix see page 4-3 of the User's Guide.

### **[Prefix]:**

The default UPOS profile has a prefix of none. The scanner default is none.

### **[Barcode Symbology ID]:**

NCR UPOS will provide barcode decoding of the scan data type of the scanner Code ID symbology IDs when the UPOS property DecodeData is set TRUE.

The scanner User's Guide Page 4-3 starts the description.

To program for use with UPOS DecodeData property = TRUE:

1. Scan Clear All Prefixes barcode
2. Scan Add Prefix, scan the character barcode in appendix for the following "995C80", Save.

To program for use with UPOS DecodeData property = FALSE and set for NCR symbology ID:

1. Scan Clear All Prefixes barcode
2. Scan Add Prefix, follow the instructions for adding a specific prefix to a specific symbology, Save.

### **[PDF417 Barcodes] (4600 only):**

PDF417 barcodes are supported by the scanner and NCR UPOS. There could be suffix configuration issues depending on whether non-printable characters appear in the PDF417 tag. If non-printable characters are present, configuration of the scanner suffix must not match any of the character combinations. NCR UPOS does support 2 character suffix combinations.

**[RS-232]:**

Scan the configuration barcode on page 2-9 to select RS-232 terminal interface. This also adds a CR-LF suffix. You must match the suffix with the UPOS profile settings. The default UPOS profile provided by NCR has a CR (0x0D) as the suffix. To set the suffix to CR only see the note above.

**[NCR USB]:**

To connect the scanner via USB, a virtual COM port driver must be installed. This driver allows the scanner to communicate via USB but provides a virtual serial COM port for NCR UPOS to connect internally to the USB port. The links below are for downloading the required driver.

To configure the scanner scan the configuration barcode in the middle of page 1-3 of the scanner User's Guide for "USB COM Port Emulation". When you plug in the scanner to the USB port, the OS will request a driver for the device. Point the OS to the location of the USB COM port emulation driver INF file to have the OS install the driver. When the install is finished a COM port will have been added to the terminal devices. Update the NCR UPOS profile to this COM port.

Link for User's Guide:

<http://www.info.ncr.com/Retail/eRetail-BrowseBy.cfm?pl=&PID=&title=&release=&pl=Retail%20-%20Scanners>

Link for the USB Serial Emulation driver:

[http://www.ncr.com/support/support\\_drivers\\_patches.asp?Class=retail\\_RealScan](http://www.ncr.com/support/support_drivers_patches.asp?Class=retail_RealScan)

# ToneIndicator

The Tone Indicator supports the following devices:

- NCR 5932 USB Keyboard
- NCR 5952 USB Dynakey

## DirectIO – Set Keyclick Tone

Sets the tone played each time a key is pressed on the keyboard. This command is always handled synchronously.

**Syntax**      **void directIO (int *command*, int[] *data*, Object *object*) throws JposException;**

### Parameter Description

---

*command*    NCRDIO\_SET\_KEYCLICK\_TONE

*data[0]*     Not used

*object*       A comma separated string giving the pitch, duration, and volume of the keyclick tone, in that order.

*The pitch value must be 0 (silent) or a value between 28 and 42192 hz.*

*The duration value must be between 0 and 1024 ms.*

*Volume is expressed as a percentage of maximum volume, with 0 being silent and 100 being maximum volume.*

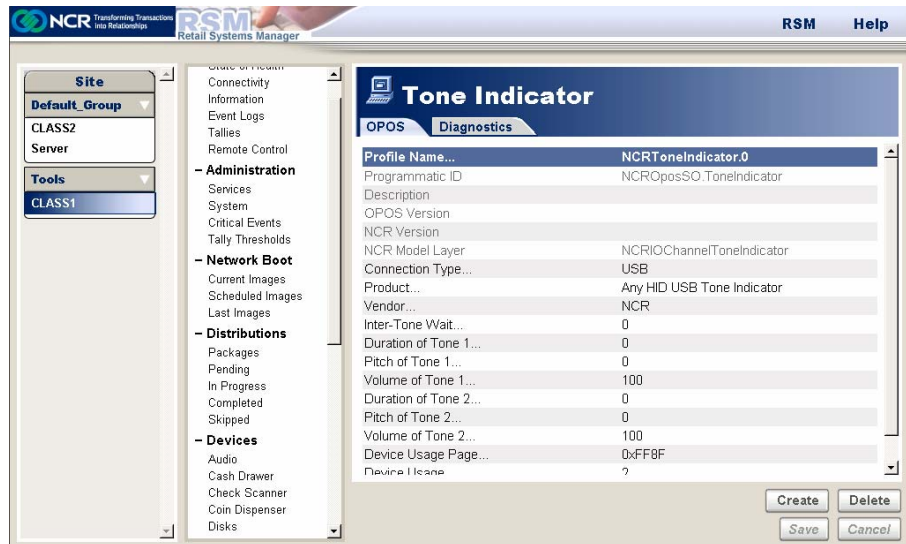
*Use the string "0,0,0" to disable keyclicks.*

**Errors**        An exception may be thrown when this method is invoked.

Some possible values of the exception's *ErrorCode* property are:

JPOS\_E\_ILLEGAL - The pitch, duration, or volume was missing from the string

## ToneIndicator - Tone Indicator Configuration Entries



Parameters	Description	Valid Values
Profile Name	The name of a specific configuration.	NCRToneIndicator.0 (Any name you wish)
Programmatic ID	This is the Service Object name for the Scale	NCROposSO.ToneIndicator
NCR Model Layer		NCRIOChannelToneIndicator
Connection Type	Indicates the means by which the Tone Indicator is connected to the terminal	USB
Product	The Product on which the USB Tone Indicator resides	Any HID USB Tone Indicator 5932 USB Keyboard 5952-6xxx/8xxx USB Dynakey 5952-9xxx USB Dynakey
Vendor	The manufacturer of the Tone Indicator device	NCR, Any Vendor
Inter-Tone Wait	The number of milliseconds between	0 - 60,000 ms



Parameters	Description	Valid Values
	tones 1 and 2.	
Duration of Tone 1	The duration of the first tone in milliseconds	0 - 1024 ms
Pitch of Tone 1	The pitch of the first tone in Hertz.	0 = Silent, 28 - 42192 hz
Volume of Tone 1	The volume of the first tone as a percent of the maximum tone.	Percent of full volume
Duration of Tone 2	The duration of the second tone in milliseconds	0 - 1024 ms
Pitch of Tone 2	The pitch of the second tone in Hertz.	0 = Silent, 28 - 42192 hz
Volume of Tone 2	The volume of the second tone as a percent of the maximum tone.	Percent of full volume
Device Usage Page	USB Configuration - Indicates the usage page for this Tone Indicator device.	0xFF8F (Read-Only)
Device Usage	USB Configuration - Indicates the usage for this Tone Indicator device.	2 (Read-Only)
Number of Reports	USB Configuration - Indicates the number of reports for the Tone Indicator device.	2 (Read-Only)

## Tone Indicator Data Capture Configuration Entries

Data capture for the Tone Indicator is controlled through settings stored in the NCRDatacap.conf configuration file. The following example shows how to enable maximum data capture for all of the Tone Indicator components.

```
[NCROposSO.ToneIndicator]
```

```
LevelMask = 0xFFFFFFFF
TraceMask = 0xFFFFFFFF
UseDefault = F
```

```
[NCRIOChannelToneIndicatorModel]
```

```
LevelMask = 0xFFFFFFFF
TraceMask = 0xFFFFFFFF
UseDefault = F
```



