

Toshiba Global Commerce Solutions
Point-of-Sale Subsystem

TOSHIBA
Leading Innovation >>>

UnifiedPOS User's Guide, Keyboards, and Code Pages

Version 22 Release 1

Toshiba Global Commerce Solutions
Point-of-Sale Subsystem

TOSHIBA
Leading Innovation >>>

UnifiedPOS User's Guide, Keyboards, and Code Pages

Version 22 Release 1

Note

Before using this information and the product it supports, be sure to read Safety Information–Read This First, Warranty Information, Uninterruptible Power Supply Information, and the information under Notices.

June 2015

This edition applies to Version 1.14 of the UnifiedPOS subsystem and to all subsequent releases and modifications until otherwise indicated in new editions.

This edition replaces GC30-4078-21.

If you send information to Toshiba Global Commerce Solutions (Toshiba), you grant Toshiba a nonexclusive right to use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

© Copyright Toshiba Global Commerce Solutions, Inc. 2015.

Contents

Safety	ix	Configuring devices	25
About this guide	xi	Testing connectivity	27
Publications accessibility	xi	Removing devices	30
Who should read this guide	xi	Deploying OPOS device configurations	30
UnifiedPOS publications	xi	Toshiba SurePoint Display (4820) touch support	31
Where to find more information	xi		
Notice statements	xi		
Part 1. User's guide	1		
Chapter 1. Introduction	3		
System requirements	3	JavaPOS problems	33
Hardware environment	3	Tracing for Linux	33
Software environment	4	Tracing for Javax.usb for Linux	34
Chapter 2. Installing UnifiedPOS	5	Tracing for Windows	35
Installation for Windows	5	OPOS problems	36
Interactive installation	5	Tracing	36
Silent installation	6	Getting help	37
Uninstallation	7		
Upgrade	8		
Installation for SUSE Linux Enterprise	8		
Resources for SUSE Linux Enterprise 11 SP3	9		
Installation package contents	10		
Package dependencies	10		
JavaPOS installation	11		
JavaPOS components	11		
javax.usb components	11		
JVM	11		
Installing Toshiba point of sale kernel mode drivers	12		
Systems Management installation	14		
Systems Management support for Toshiba POS peripherals	14		
Installing on SLE 11 SP3	14		
Validating Systems Management	15		
Uninstallation of UPOS Systems Management	15		
Chapter 3. Configuring devices	17		
JavaPOS configuration	17		
Locating sample jpos.xml files	17		
Creating the jpos.xml file	17		
Changing the logical name of JposEntry	18		
Device configuration	18		
POS Control Center utility	21		
Toshiba SurePoint Display (4820) touch support	21		
USB device access	21	Additional JavaPOS information	55
Microsoft Windows	21	4610 printer-attached cash drawer	55
Linux	21	Loading the cash drawer driver for Linux	56
Network configuration tool	22	JavaPOS DirectIO calls	56
OPOS configuration	22	Cash Drawer get serial number command	56
Network device communication	22	OPOS configuration	57
Using the OPOS Configuration utility	23	Additional OPOS information	57
OEM cash drawer support	57	USB cash drawer support	57

OPOS DirectIO calls	58
Cash Drawer get serial number command	58
Chapter 7. Check scanner	59
Check scanner supported devices	59
Supported properties and methods	59
JavaPOS DirectIO calls.	62
Scanner calibration command	62
Print scanned image command	63
Set check side command	63
Get check side command	64
Scanner image quality command	64
Additional OPOS information	64
Multiple devices.	64
Chapter 8. Fiscal printer	65
Fiscal printer supported devices	65
Supported properties and methods	66
Supported properties and methods (fiscal printer: Italy)	71
Supported properties and methods (fiscal printer: Greece)	74
JavaPOS configuration.	80
Monitor detection	80
JavaPOS DirectIO calls.	80
Fiscal information command.	81
Fiscal read command	81
Fiscal write command	81
Fiscal notify command	82
JavaPOS DirectIO events	82
Fiscal error event	82
Fiscal status event	83
Fiscal data available event	83
Fiscal raw status event	83
Fiscal IPL end status event	83
Additional JavaPOS information	84
Device sensing	84
OPOS DirectIO calls	85
Fiscal information command.	85
Fiscal read command	85
Fiscal write command	86
Fiscal notify command	86
OPOS DirectIO events	86
Fiscal error event	86
Fiscal status event	87
Fiscal data available event	88
Fiscal raw status event	88
Fiscal IPL end status event	88
Additional OPOS information	88
Suspended fiscal transaction.	88
Display device sensing	89
OPOS getData method (fiscal printer: Italy).	91
OPOS getData method (fiscal printer: Greece)	93
OPOS getTotalizer method (fiscal printer: Greece)	96
Chapter 9. Hard totals	99
Hard totals supported devices	99
Supported properties and methods	99
OPOS configuration	101

Chapter 10. Keylock	103
Supported devices.	103
Two-position (devices 1, 2, 3, 4, 5, 6, 20, 21, 22)	104
Two-position (device 7)	104
Three-position keylock (devices 8, 12)	105
Four-position (devices 9, 10, 11, 14, 15, 16, 17, 19)	106
Four-position (devices 20, 21, 22)	106
Five-position (device 13)	106
Six-position (device 18)	107
Electronic keylock (iButton)	107
Supported properties and methods	108
JavaPOS configuration	110
PositionCount	110
OPOS configuration	110
Additional OPOS information	110
Two-position keylock	110
Position count	111
Chapter 11. Line display	113
Supported devices.	113
Supported properties and methods	113
Supported properties and methods (line display attached fiscal printer: Italy)	119
JavaPOS configuration	121
CharacterSetASCIIBehavior.	121
CursorState	121
lineDisplayId	121
NumberOfRows	122
JavaPOS DirectIO calls	122
Select cursor type command	122
Select number of rows command	122
Select rotate text 180 command	122
Additional JavaPOS information	123
Descriptors	123
Capability definitions.	123
Property definitions	123
Character sets supported in JavaPOS	124
USB APA font download	125
OPOS configuration	126
OPOS DirectIO calls	126
Set number of APA rows	126
Additional OPOS information	127
USB APA font download	127
Character sets supported in OPOS	128
Chapter 12. Magnetic Ink Character Recognition	129
Supported devices.	129
Supported properties and methods	129
Magnetic Ink Character Recognition data	131
JavaPOS configuration	131
MinMicrSignalLevel	131
stripAccountDashes	131
stripTransitDashes	132
switchTransitDashToSpace	132
Additional JavaPOS information	132
MICR exception tables	132
OPOS configuration	135
OPOS DirectIO calls	136

Add new MICR exception	136
Additional OPOS information	136
MICR exception file	136
Chapter 13. Motion sensor	139
Motion sensor supported devices	139
Supported properties and methods	139
Additional JavaPOS information	140
Chapter 14. Magnetic stripe reader	143
Supported devices	143
Supported properties and methods	144
JavaPOS configuration	147
enableOnlineWatcher	147
msrID	147
JavaPOS DirectIO calls	147
MSR write data command	147
Additional JavaPOS information	148
SureOne MSR	148
OPOS configuration	148
Additional OPOS information	149
SureOne/SurePOS 100 MSR limitations	149
Chapter 15. POS keyboard	151
POS keyboard supported devices	151
Supported properties and methods	152
CANPOS keyboard	153
Installing USB system-attached keyboard in Windows XP	154
JavaPOS configuration	154
Click	154
FatFingerTimeout	154
ExtendedKeyMapping	155
KbdScanning	155
KeyTranslationFile	155
Typematic	155
TypematicDelay	156
TypematicRate	156
JavaPOS DirectIO calls	157
Get keyboard light status command	157
Set keyboard light on command	158
Set keyboard light off command	158
Additional JavaPOS information	159
Firmware Update	159
Configuration file update	159
Keyboard definition file	160
USB System attached keyboard on Linux	161
OPOS configuration	162
OPOS DirectIO calls	163
Get keyboard light status	163
Set keyboard light on	163
Set keyboard light off	163
Additional OPOS information	163
Firmware Update	163
Configuration file update	164
Configuring the Alphanumeric Point of Sale keyboard	164
SureOne/SurePOS 100 keyboard	165
Mapping POS keys	165
Chapter 16. POS printer	171
Supported devices	171
Supported properties and methods	171
JavaPOS configuration	188
Arabic code page support	188
AddLineFeed	189
Bitmap quality	189
Color support	190
Default8LPI	190
DefaultLargeFont	190
DocAdvance	190
DoubleHighAndDoubleWideIsQuad	190
Enable or Disable Partial Cut Detection	191
Enable or Disable Paper Motion Sensing	191
Enable or Disable Low Paper Detection	191
Enable or Disable Low Paper LED	192
PDF417AspectHeight	192
PDF417AspectWidth	192
PDF417ECCLevel	192
PDF417Truncation	192
OptimizedBaudRate	193
QRCodeEncodingMode	193
QRCodeErrorCorrectionLevel	194
QRCodeECIValue	194
QR Barcode Usage	195
ProportionalFontFixedWidth	196
RecLineCharsFont	196
TranslateCharacter	197
SetCompatibilityMode	197
SlpLineSpacing	197
SureOneVersion	197
JavaPOS DirectIO calls	198
Download DBCS Font ID Command	198
Download Non-Proportional Font ID Command	199
Download Prop Font ID Command	200
FontInfo Object	201
Get Font Info ID Command	201
Additional JavaPOS information	201
Calibration for low paper sense	201
4610 printer firmware update	202
4610 printer font download	204
Using POSPrinter's MICR device	204
Handling invalid characters	204
Escape sequence handling	205
Line wrapping on the slip station using rotatePrint method(PTR_RP_LEFT90)	206
Bar Code GS1 Databar information	206
Bitmap printing	208
Enable DBCS character compression on SLIP (Impact) station	208
Check flipping support	209
PageMode support	209
Paper saving mode	214
Concurrent MICR/full-image scan	215
Character sets supported by POSPrinter devices for JavaPOS	215
OPOS configuration	218
Arabic code page support	219
Unsupported settings	219
Supported settings	223
OPOS DirectIO calls	229

Miscellaneous	229
Flash memory	230
Check scanning support	233
Page Mode printing	238
OPOS DirectIO events	240
NVRAM data	240
NVRAM error	240
OPOS extended result codes	241
Additional OPOS information	242
4610 printer firmware update	242
4610 printer font download.	244
4689 printer font download.	247
Model 4 printer font download	248
Barcode printing	250
QR code printing	250
Rotated printing	252
DBCS support	253
Color printing	253
User-defined fonts.	253
Proportional font support	253
Device sharing	254
Code 128 A/B/C support	254
SureOne/SurePOS 100 DBCS printer limitations	254
Improving printer performance	255
Character sets supported by POSPrinter devices for OPOS.	255
Chapter 17. Scanner (bar code reader) 259	
Supported devices.	259
Supported properties and methods	259
JavaPOS configuration	261
enableLaserOnOffSwitch	264
enableProgrammingViaBarcodes	264
enableSwitchControlledVolumeAdjust	264
setBarCodes1	264
setBarCodes2	265
setBarCodes3	265
setBarCodes4	265
setBeepDuration.	266
setBeepFrequency	266
setBeeperVolume	266
setCheckModulo	267
setCode128ScansPerRead	267
setCode39ScansPerRead	267
setDecodeAlgorithm	267
setDoubleReadTimeOut	268
setDTouchMode	268
setEAN13ScansPerRead	268
setEAN8ScansPerRead	269
setEnable_2_DigitSupplements	269
setEnable_5_DigitSupplements	269
setEnableCodabar	269
setEnableCode128	270
setEnableCode128Supplements	270
setEnableCODE39	270
setEnableCode39CheckDigit	270
setEnableCode93	270
setEnableEAN_JAN_TwoLabelDecoding	270
setEnableCompositeCC_A	271
setEnableCompositeCC_B	271
setEnableCompositeCC_C	271
setEnableGoodReadBeep	271
setEnableInterleaved2of5	271
setEnableITFCheckDigit	272
setEnablePDF417	272
setEnableMicroPDF417	272
setEnableRSS14 (deprecated)	272
setEnableGS1Databar.	272
setEnableRSS_Expanded (deprecated)	273
setEnableGS1Databar_Expanded	273
setEnableStandard2of5	273
setEnableUCC_EAN128	273
setEnableUPC_A_CheckDigit	273
setEnableUPC_A_To_EAN13Expansion	273
setEnableUPC_E_CheckDigit	274
setEnableUPC_E_To_EAN13Expansion	274
setEnableUPC_E_To_UPC_AExpansion	274
setEnableUPCAE_EANJAN813	274
setEnableUPCD1D5	274
setEnableVerificationUPC_A_EAN13_fiveDigit	275
setEnableVerificationUPC_A_EAN13_fourDigit	275
setITFLength1	275
setITFLength2	275
setITFLengths	276
setITFLengthSpecifiedTwo	276
setITFScansPerRead	276
setLaserTimeOut	276
setLED_GoodRead_Duration	277
setMotorTimeOut	277
setSecurityLevelForInStore	278
setSTFLength1	278
setSTFLength2	278
setSTFLengths	278
setStoreScansPerRead.	279
setSupplements	279
setSupplementsSecurityLevel	279
setTransmitScannerPrefixAndSuffix	279
setUPCAScansPerRead	280
setUPCDScansPerRead	280
setUPCEScansPerRead	280
Additional JavaPOS information	280
Toshiba 4697 Point of Sale Scanner Configuration	280
USB OEM Scanner and scale configuration	281
Known scanner issues	283
OPOS configuration	284
Scanner programming	284
Scanner Programming (AnyPlace Kiosk)	286
Additional OPOS information	289
Scanner detection	289
Known scanner issues	289
Chapter 18. Scale. 291	
Supported devices.	291
Supported properties and methods	291
JavaPOS configuration	293
CenterOfZero	294
DisplayRequired	294
EnforceZeroReturn	294
FiveDigitWeight (USB OEM only)	294
OperationMode	295
VibrationSensitivity	295

WeighMode	295
Additional JavaPOS information	295
Known scale issues	295
OPOS configuration	296
Chapter 19. Tone indicator	297
Supported devices.	297
Supported properties and methods	297
OPOS configuration	299
Additional JavaPOS configuration	300
Enable or disable ELO tone interface	300
JavaPOS ELO tone interface configuration.	300
Part 3. Keyboards and code pages	301
Chapter 20. POS keyboard layouts and scan codes	303
Understanding scan codes	303
Checkout Keyboards Layout (50-key and Modular 67 Key)	303
50-Key Modifiable Layout Keyboard	303
50-Key Modifiable Layout Keyboard and Operator Display	303
Retail Point of Sale Keyboard Layout	304
Retail Point of Sale Keyboard with Card Reader and Display	304
Point of Sale Keyboard VI Layout	304
Modular 67-Key Keyboard Layout	305
Modular 67-Key Keyboard with Display Layout	305
Checkout Keyboards PS/2 or USB (System Keyboard) scan code set.	306
Checkout keyboards RS-485/USB scan code set	308
Modifiable-layout keyboard with card reader layout (133-key)	311
Modifiable layout keyboard with card reader RS-485/USB scan code set	312
Alphanumeric Point of Sale keyboards	316
PC Point of Sale keyboard with card reader (ANKPOS)	316
Retail Alphanumeric Point of Sale Keyboard with card reader (NANPOS)	316
Modular Alphanumeric POS Keyboard	317
Modular Compact Alphanumeric Point of Sale Keyboard.	317
RS-485/USB scan code set for Retail Alphanumeric Point of Sale Keyboard (NANPOS), PC Point of Sale Keyboard (ANKPOS), USB scan code set for Modular Alphanumeric Point of Sale Keyboard	318
USB scan code set for the Modular Compact Alphanumeric Point of Sale Keyboard	323
Retail Alphanumeric Point of Sale keyboard with card reader (PS/2 or USB system attached)	327
Retail Alphanumeric Point of Sale with card reader or PC Point of Sale keyboard (ANKPOS) layout keyboard	327
Retail Alphanumeric Point of Sale with card reader (NANPOS) layout keyboard	327
Modular Alphanumeric Point of Sale Keyboard	328
Compact Alphanumeric Point of Sale (CANPOS) layout keyboard	329
Modular Compact Alphanumeric Point of Sale Keyboard Layout	329
PS/2 or USB (System Attached) scan code set for Retail Alphanumeric Point of Sale Keyboard (NANPOS), Modular Alphanumeric Point of Sale Keyboard, PC Point of Sale Keyboard (ANKPOS)	330
PS/2 scan code set for the Compact Alphanumeric Point of Sale Keyboard (CANPOS), PS/2 / USB (System Attached) Modular Compact Alphanumeric Point of Sale Keyboard.	337
Point of Sale Keyboard V	344
Keyboard V scan codes	344
PLU Keyboard and Display III	344
4674 Point of Sale Keyboard (built-in)	345
4674 POS Keyboard (built-in) scan codes	345
4685 keyboards.	346
4685 Point of Sale Keyboard Model K01	346
4685 Keyboard K02 Ultra 7 (four-position keylock)	347
4685 Keyboard Model K02 Ultra 7 with MSR/E (four or six position keylock)	348
4685 Keyboard Model KC1	349
4685 Keyboard Model K03	350
4820 Toshiba SurePoint Solution Keypad	350
4820 Toshiba SurePoint Solution Keypad (system-attached)	352
SureOne model 4614/4615 and SurePOS 100 model 4613 keyboard (built-in)	354
SureOne model 4614/4615 and SurePOS 100 model 4613 keyboard (built-in) scan codes.	355
Alphanumeric Point-of-Sale (NANPOS) country-dependent keyboards	361
Canadian French keyboard layout	361
French keyboard layout	361
German keyboard layout	362
Italian keyboard layout	362
Spanish keyboard layout	362
U.K. English keyboard layout	363
U.S. English keyboard layout	363
Retail Alphanumeric Point of Sale country dependent keyboards.	363
Brazil-Portuguese keyboard layout	364
Canadian French keyboard layout	364
Danish keyboard layout	364
French keyboard layout	365
German keyboard layout	365
Italian keyboard layout	366
Norwegian keyboard layout	366
Spanish keyboard layout	367
Swedish and Finnish keyboard layout	367
U.K. English keyboard layout	368
U.S. English keyboard layout	368
Compact Alphanumeric Point-of-Sale (CANPOS) keyboard	369
CANPOS keyboards by country	369
CANPOS country-dependent keyboards	371

Chapter 21. Character sets for terminal printers and displays	375
Code Page 101	378
Code Page 437	379
Code Page 775	380
Code Page 808	381
Code Page 819	382
Code Page 848	383
Code Page 850	384
Code Page 852	385
Code Page 855	386
Code Page 857	387
Code Page 858	388
Code Page 860	389
Code Page 861	390
Code Page 862	391
Code Page 863	392
Code Page 864	393
Code Page 865	394
Code Page 866	395
Code Page 867	396
Code Page 869	397
Code Page 872	398
Code Page 874	399
Code Page 1116.	400
Code Page 1117.	401
Code Page 1118.	402
Code Page 1250	403
Code Page 1251	404
Code Page 1252	405
Code Page 1253	406
Code Page 1254	407
Code Page 1255	408
Code Page 1256	409
Code Page 1257	410
Code Page 932	411
Code Page 936	412
Code Page 949	413
Code Page 950	414
Code Page 951	415
Code Page 1381	428
Part 4. Appendixes	429
Appendix A. JavaPOS support for UnifiedPOS device statistics properties	431
Common properties	431
Cash drawer	431
Check scanner	433
Fiscal printer	434
Hard totals	436
Keylock	437
Line display	438
MICR	440
Motion sensor	441
MSR	441
POS keyboard	444
POS printer	446
Scale	449
Scanner	449
Tone indicator	451
Appendix B. Java Web Start configuration to access the POS control center	453
Install JavaPOS SDK	453
Create a KeyStore	453
Create a self-signed certificate	454
Create jpos.xml	454
Create the resource JAR	454
Sign the all JARs file with the test certificate	455
Run JNLP file	455
Appendix C. Additional information for SUSE Linux Enterprise (SLE)	457
USB Alphanumeric POS Keyboard does not receive scan codes	457
Known GUI issues	457
Power management	457
SLE 11 and PM Utils package	458
Appendix D. JavaPOS device configuration for Linux	459
Serial ports configuration	459
Serial port enumeration utility	459
Toshiba systems and COM port assignments reference	459
Serial port access	460
TCxWave USB to RS-232 adapter	461
USB device access	461
SLE 11 without Service Pack and newer SLE 11 versions	461
PS/2 keyboard configuration	462
USB system attached POS keyboard configuration	462
Touch device configuration	463
Notices	465
Glossary	467
Index	483

Safety

Before installing this product, read Safety Information—Read This First.

قبل تركيب هذا المنتج، يجب قراءة الملاحظات الأمنية

Antes de instalar este producto, leia as Informações de Segurança.

在安装本产品之前，请仔细阅读 **Safety Information**
(安全信息)。

安裝本產品之前，請先閱讀「安全資訊」。

Prije instalacije ovog produkta obavezno pročitajte Sigurnosne Upute.

Před instalací tohoto produktu si přečtěte příručku bezpečnostních instrukcí.

Læs sikkerhedsforskrifterne, før du installerer dette produkt.

Lees voordat u dit product installeert eerst de veiligheidsvoorschriften.

Ennen kuin asennat tämän tuotteen, lue turvaohjeet kohdasta Safety Information.

Avant d'installer ce produit, lisez les consignes de sécurité.

Vor der Installation dieses Produkts die Sicherheitshinweise lesen.

Πριν εγκαταστήσετε το προϊόν αυτό, διαβάστε τις πληροφορίες ασφάλειας (safety information).

לפני שתתקינו מוצר זה, קראו את הוראות הבטיחות.

A termék telepítése előtt olvassa el a Biztonsági előírásokat!

Prima di installare questo prodotto, leggere le Informazioni sulla Sicurezza.

製品の設置の前に、安全情報を読みください。

본 제품을 설치하기 전에 안전 정보를 읽으십시오.

Пред да се инсталира овој продукт, прочитајте информацијата за безбедност.

Les sikkerhetsinformasjonen (Safety Information) før du installerer dette produktet.

Przed zainstalowaniem tego produktu, należy zapoznać się z książką "Informacje dotyczące bezpieczeństwa" (Safety Information).

Antes de instalar este produto, leia as Informações sobre Segurança.

Перед установкой продукта прочтите инструкции по технике безопасности.

Pred inštaláciou tohto zariadenia si pečítaje Bezpečnostné predpisy.

Pred namestitvijo tega proizvoda preberite Varnostne informacije.

Antes de instalar este producto, lea la información de seguridad.

Läs säkerhetsinformationen innan du installerar den här produkten.

About this guide

This guide contains reference information for Unified Point of Service (UnifiedPOS) implementations. This guide also includes information about keyboards and code pages.

Publications accessibility

The softcopy version of this guide and other related publications is accessibility enabled.

Who should read this guide

This guide is intended for use by point-of-sale application developers who need to access Toshiba point-of-sale hardware using UnifiedPOS.

UnifiedPOS publications

Unified Point of Service (UnifiedPOS) Retail Peripheral Architecture publications can be found at www.nrf-arts.org/UnifiedPOS/default.htm.

Where to find more information

Current versions of Toshiba Global Commerce Solutions documentation are available on the Toshiba Global Commerce Solutions support website at <http://www.toshibagcs.com/support>.

1. On the right side of the web page under popular links, select **Publications**.
2. Click on the publication related to your product.

Notice statements

Notices in this guide are defined as follows:

Notes These notices provide important tips, guidance, or advice.

Important

These notices provide information or advice that might help you avoid inconvenient or problem situations.

Attention

These notices indicate potential damage to programs, devices, or data. An attention notice is placed just before the instruction or situation in which damage could occur.

CAUTION

These statements indicate situations that can be potentially hazardous to you. A caution statement is placed just before the description of a potentially hazardous procedure step or situation.

DANGER

These statements indicate situations that can be potentially lethal or extremely hazardous to you. A danger statement is placed just before the description of a potentially lethal or extremely hazardous procedure step or situation.

Part 1. User's guide

Chapter 1. Introduction

Unified Point of Service (UnifiedPOS) is an architectural specification developed by the Association for Retail Technology Standards (ARTS), of which Toshiba is a member. UnifiedPOS provides a standard application programming interface (API) specification for point-of-service devices that are used in the retail environment. The specification standard is independent of a specific operating system and language, and defines a set of retail device behaviors that are sufficient to support a range of point-of-service solutions.

The Toshiba UnifiedPOS implementation of the ARTS UnifiedPOS architectural specification includes JavaPOS and Object Linking and Embedding (OLE) for Retail Point of Sale (OPOS):

- **JavaPOS** is an implementation of the UnifiedPOS specification written in Java® for the support of Java applications.
- **OPOS** is an implementation of the UnifiedPOS specification for Microsoft® Windows® systems. The drivers are implemented using COM, and are delivered as ActiveX controls.

System requirements

This section contains information on the hardware, software, disk space, and memory requirements for the UnifiedPOS 1.14 Subsystem.

Hardware environment

The following table lists the point-of-sale terminals that are supported by the UnifiedPOS implementation.

Table 1. Supported POS terminals

Terminal	Models
4674 POS Terminal (Japan only)	All
4694 POS Terminal	2x5, 2x6, 2x7, 3x7
	OPOS only: 0x1, 0x4, Sx1, Sx4, 1x4, 2x4
SurePOS 100 Series	All
SurePOS 300 Series	All
SurePOS 500/600 Series	All
SurePOS 700 Series	All
Kiosk	All
SureOne® POS system	All
TCx Wave	All
TCx 300 Series	All
TCx 700 Series	All

Attention: Hot-plugging a powered peripheral device can cause damage to the device and/or the terminal and is not recommended or supported. This includes, but is not limited to, RS-485, USB, PS/2, and powered EIA-232 devices.

Software environment

The UnifiedPOS system requires the following software environment:

Table 2. Software environment for UnifiedPOS

Operating system	One of the following: <ul style="list-style-type: none">• Microsoft® POSReady• Microsoft Windows® XP Service Pack 3• Microsoft Windows Embedded Point of Service (WEPOS)• Microsoft Windows 7 Ultimate Service Pack 1• SUSE Linux Enterprise Desktop 11 (SLED 11) Service Pack 1. This is for JavaPOS only.• SUSE Linux Enterprise Server 11 (SLES 11) Service Pack 1. This is for JavaPOS only.• SUSE Linux Enterprise Point of Service 11 (SLEPOS 11) Service Pack 1. This is for JavaPOS only.
Java Virtual Machine (JVM)	IBM Java Runtime environment (JRE) or software development kit (SDK), version 1.6.0 SR16 FP4. To determine JVM version, at the command prompt enter: <code>java -fullversion</code> .
Control Objects	OPOS common control objects are required to run the Toshiba OLE for Retail Subsystem. They can be obtained from www.monroecs.com/oposccos.htm or the UnifiedPOS installer.

Chapter 2. Installing UnifiedPOS

This chapter contains information on the UnifiedPOS installation for Windows and IBM Retail Environment for SuSE Linux (IRES).

Installation for Windows

There are two options for installing the UnifiedPOS for Retail POS Suite: interactive installation and silent installation.

Interactive installation

1. Run the Setup.exe file and follow the directions on each panel. The prompt to install Microsoft .Net Framework might appear.
2. In the Features dialog (Figure 1), choose one of the following features:
 - JavaPOS Device Support
 - OPOS Device Support

The system unit information is displayed to the user if the system is recognized.

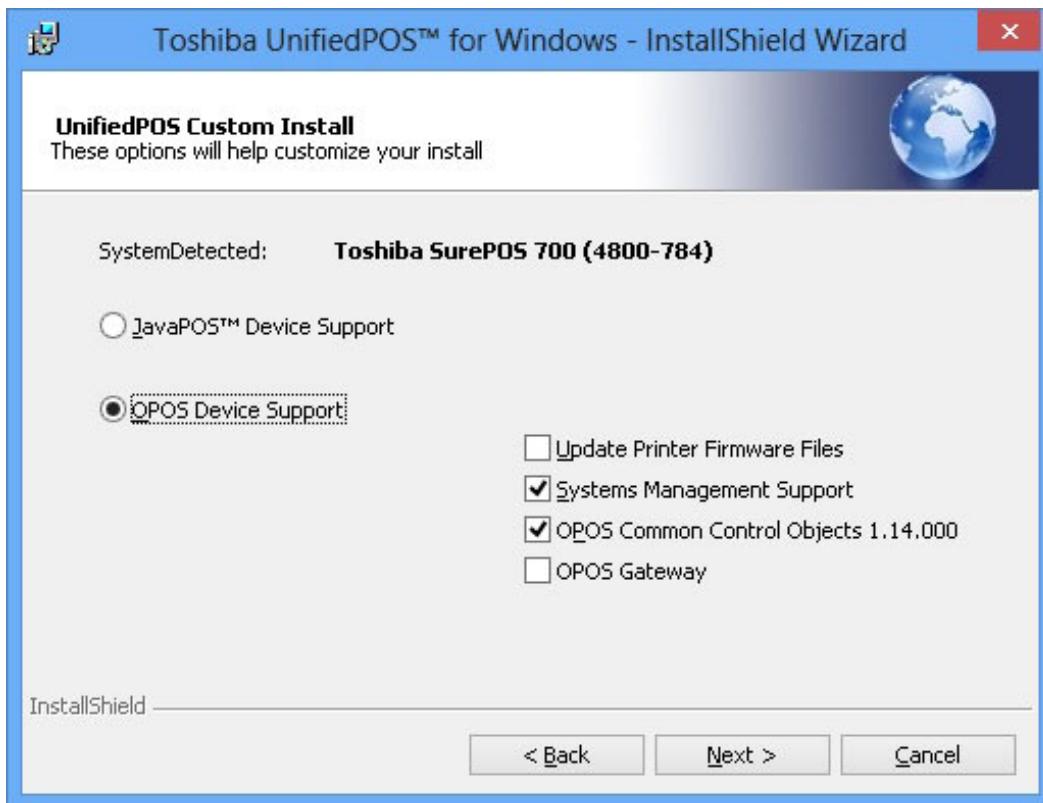


Figure 1. Features dialog

3. After the installation is complete, restart the system so that the configuration changes take effect.

Note:

1. If you select **OPOS Common Control Objects** during OPOS installation, the installation will override any existing OPOS Common Control Objects on the system.
2. It is your responsibility to re-register the OPOS Common Control Objects if UnifiedPOS is uninstalled.

Silent installation

UnifiedPOS can be installed and updated silently (unattended) using a response file. The response file C:\UpoSetup.iss is created during the initial installation of the package on a POS system. This response file can then be used to perform an unattended install or update on other systems.

Generate a response file

To generate a response file, the product must first be installed on a system with certain flags. Run setup.exe with the options described below to generate a response file without actually installing the product.

From a command prompt, enter:

```
setup.exe /v"ONLYINI=1"
```

Follow the instructions on the screen to install the OPOS or JavaPOS product. At the end of the installation, this process will generate a response file in C:\UpoSetup.iss.

Unattended installation

- Copy the setup.exe file to the target system.
- Copy the UpoSetup.iss response file to the C:\ root directory on the target system.
- Execute **setup.exe** on the client with additional flags described below.

When installing silently on Windows Vista, the user will see both the User Access Control and logo signing messages. To hide these messages, the user must disable both the UAC and the logo signing messages.

Unattended installation with reboot

To perform an unattended installation or update without a user interface, use the following command:

```
setup.exe /s /v/qn
```

To perform an unattended installation or update with a basic user interface showing progress, errors, and completion, use the following command:

```
setup.exe /s /v/qb
```

To enable logging for the installation without reboot, use the following command:

```
setup.exe /s /v"/qb /l*vx C:\msilog.txt"
setup.exe /s /v"/qn /l*vx C:\msilog.txt"
```

Unattended installation without reboot

To prevent the automatic reboot following installation or update, use one of the following commands:

```
setup.exe /s /v"/qn REBOOT=R"
setup.exe /s /v"/qn REBOOT=R /l*vx C:\msilog.txt"
```

Note:

1. There is no space between **/v"** and **/qn** or between **/v"** and **/qb**.
2. The commands **/qn** and **/qb** are interchangeable in all the examples shown
3. The double quotes are required as shown.
4. The C:\msilog.txt file contains the install log.

Unattended installation flags

In addition to the default flags available, the following parameters can also be set:

USEINI

Specify the location of the .iss file to use during silent install. If this parameter is not specified, the default value of [WindowsVolume]UposSetup.iss is used.

SAVEINI

Specify where to save the .iss file during installation. If this parameter is not specified, the default value of [WindowsVolume]UposSetup.iss is used.

REMOVEINI

Setting this value to 1 removes the .iss file from the target machine. Specify the USEINI parameter to remove a file saved in a location other than the default location.

ONLYINI

Setting this parameter to 1 runs the UI installation and generates a .iss file without actually installing on that system. The installation exits immediately after clicking "Install" in the "ReadyToInstall" dialog.

Examples of these commands:

```
setup.exe /s /v"/qn USEINI=C:\UposSetup.iss"  
setup.exe /v"SAVEINI=C:\UposSetup.iss"  
setup.exe /x /vREMOVEINI=1  
setup.exe /vONLYINI=1
```

Uninstallation

Interactively uninstall UnifiedPOS from Windows by using the **Add/Remove Programs** option in the Windows Control Panel.

To perform an unattended uninstall, use one of the following commands:

```
setup.exe /s /x /vqn  
setup.exe /x /s /vqb
```

Note: There is no space between **/v** and **qn**.

Windows 7

If you are logged in as the default "Administrator" account in Windows 7, then you may not be able to remove the UnifiedPOS for Windows product through the Control Panel Programs and Features method. To resolve this problem, do one of the following two solutions:

- Create a second user account with administrator privileges. Use this account to uninstall the product.
- Double-click **setup.exe** and click **Remove** to uninstall the product.

4840 systems

Before installing UnifiedPOS, the following is displayed in Device Manager under "Multifunction adapters":

Toshiba 4840-xx3 PCI Multi-I/O Controller

No driver files are required or loaded for this device.

After installing JPOS, the following is displayed in Device Manager under "Multifunction adapters":

Toshiba 4840 System Port

This replaces the 4840-xx3 PCI port listed above.

The driver for the 4840 System Port is:

c:\windows\system32\drivers\aiptn.sys

After uninstalling UnifiedPOS and the 4840 System Port, the hardware manager displays the following upon reboot: Toshiba 4840-xx3 PCI Multi-I/O Controller. The device manager then prompts you to reload the drivers for this device.

Upgrade

Beginning with the UnifiedPOS 1.13.3 release, the installation process can perform an upgrade over a previously installed UnifiedPOS product on the system. The installation process uninstalls the previous version and then installs the new version automatically.

Note: Depending on the system/device configurations, the operating system may prompt for two reboots to fully load the device drivers for the inputs/outputs. This is true specifically for system-attached PS2/USB keyboards.

Upgrade for interactive installation

When the UnifiedPOS installation detects a previous version of the product, the installation process displays an upgrade window:

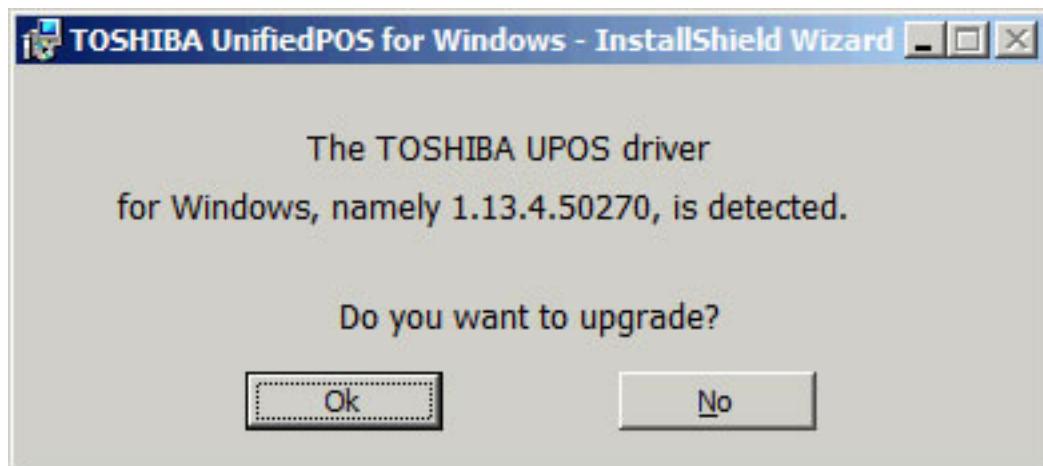


Figure 2. Upgrade Window

1. Click **Yes** to continue the installation process as if it is a new installation. See “Interactive installation” on page 5.
2. Click **No** to abort the installation without affecting the previously installed product on the system. In this case, you can manually uninstall the previous product and reinstall the new product.

Upgrade for silent installation

To perform a silent upgrade, create a response file for the new product as described in “Silent installation” on page 6.

The response file must match the product version that is being installed or the installation will be aborted.

Installation for SUSE Linux Enterprise

This section provides installation instructions for the Toshiba JavaPOS on Linux distributions. The Toshiba JavaPOS is tested and supported on the SUSE Linux Enterprise Family:

- SLED 11 SP3, SLES 11 SP3, SLEPOS 11 SP3

Note: The instructions provided in this document are based on the SUSE Linux file structure. The Toshiba JavaPOS installation can be easily adapted to other Linux distributions.

For support related to other Linux distributions, please contact your Toshiba representative or visit the Toshiba Global Commerce Solutions support website at following link to submit a Techline question under the "Contact Us" section: www.toshibacommerce.com/support

Resources for SUSE Linux Enterprise 11 SP3

The following table provides the resources for setting up the SUSE Linux Enterprise family.

Note: You must install the Toshiba POS kernel mode drivers. They are required for Toshiba JavaPOS support.

Table 3. Resources for SUSE Linux Enterprise 11 SP3 (SLE11 SP3)

Name	Description or Resource Link
Kernel version of SLED/SLES11 SP3 and SLEPOS11 SP3 tested and supported by Toshiba JavaPOS	<ul style="list-style-type: none"> 3.0.70-0.9-default (SLEPOS11-SP3 via Kiwi image creator) 3.0.70-0.9-pae (SLED/SLES11-SP3)
Toshiba POS Kernel Mode Drivers required for JavaPOS support	<p>http://drivers.suse.com/Toshiba/tgcs/pos/sle11sp3/i586/</p> <p>You only need to install the following RPM that contains Toshiba POS kernel mode drivers required by JavaPOS. The OS kernel will determine exactly which RPM to install.</p> <p>For Kernel Version: 3.0.<version>-default</p> <ul style="list-style-type: none"> <code>toshiba-pos-suse12-kmp-default-8.0.0_<version>.i586.rpm</code> <code>toshiba-sleep-prot-suse12-kmp-default-<version>.i586.rpm</code> <p>For Kernel Version: 3.0.<version>-pae</p> <ul style="list-style-type: none"> <code>toshiba-pos-suse12-kmp-pae-8.0.0_<version>.i586.rpm</code> <code>toshiba-sleep-prot-suse12-kmp-pae-<version>.i586.rpm</code>
SLEPOS 11 SP3 Quick Installation Guide	http://www-01.ibm.com/support/docview.wss?rs=220&uid=pos1R1004407
Creating Image with JavaPOS (Pre-requisite: SLEPOS11-SP3 Quick Installation Guide)	http://www-01.ibm.com/support/docview.wss?rs=0&uid=pos1R1004460
Toshiba POS Linux Configuration Guide	<p>http://www-01.ibm.com/support/docview.wss?rs=219&uid=pos1R7000429</p> <ul style="list-style-type: none"> Describes Toshiba POS Systems supported on SLE 11 SP3 Provides installation and configuration information
SLEPOS 11 SP3 Tools and Utilities	http://www-01.ibm.com/support/docview.wss?rs=220&uid=pos1R1004421
Toshiba Systems Management for POS peripherals	Refer to the document included in this package: Toshiba UnifiedPOS Management Services.pdf
Toshiba UnifiedPOS Programming Reference, Keyboards, and Code pages	<p>http://www.toshibacommerce.com</p> <ol style="list-style-type: none"> Click Support and select Publications. Click the Hardware tab. Click the Toshiba UPOS (JavaPOS/OPOS) link under Point of Sale Driver Subsystems.

Installation package contents

The main installation package can be downloaded from the Toshiba Global Commerce Solutions website: www.toshibacommerce.com.

The following is a list of RPMs that are necessary to install the Toshiba JavaPOS product.

Depending on the JVM architecture used, <arch> = i386

JavaPOS

- toshiba-javapos-<version>-<build>.i386.rpm
- toshibaposs -gcc43<version>-<build>.i386.rpm

JVM

- ibm-java-i386-jre-<version>.i386.rpm
- ibm-java-i386-javacomm-<version>.i386.rpm

OpenJDK is supported for 64-bit only.

JavaUsb

- javax-usb-1.0.2-1.i386.rpm
- javax-usb-ri-1.0.2-1.i386.rpm
- javax-usb-ri-linux-1.0.3-1.i386.rpm

Systems Management for POS Peripherals

- toshiba_upos_sblim-cmpi-upos-server-sled-<version>.i586.rpm
- posIBM_XML4C-<version>.i586.rpm
- Toshiba UnifiedPOS Management Services.pdf

Toshiba JavaPOS for Linux Installation Instructions.pdf

Toshiba POS kernel mode driver source for other Linux distributions

- pos_kernel_drivers_other_linux / toshibaposs-kernel-<version>.rpm

Package dependencies

During the operating system installation, you must select the following RPMs which will install necessary libraries required for the JavaPOS product:

SLE 11 SP3

IBM JVM 1.6 requires:

From Development C/C++ compiler and tools, select and install libstdc++.so.5 library from libstdc++33 RPM.

Toshiba JavaPOS Supported Environment on SLE 11 SP3

Table 4. Toshiba JavaPOS Supported Environment on SLE 11 SP3

Component	Version
Kernel Version	3.0.70-0.9-default 3.0.70-0.9-pae Available on SLED11- SP3/SLEPOS11- SP3
IBM JVM	1.6 – SR16 FP4
System Management of Toshiba POS Peripherals	Small Footprint CIM Broker (SFCB) sblim-sfcb-1.3.11-0.19.1.i586.rpm

Table 4. Toshiba JavaPOS Supported Environment on SLE 11 SP3 (continued)

Component	Version
POS Systems supported / Configuration	Refer to the Toshiba POS Linux Configuration Guide: http://www-01.ibm.com/support/docview.wss?rs=219&uid=pos1R7000429

JavaPOS installation

The instructions in this section assume the following:

- The user has root privileges
- % represents a console command prompt

The JavaPOS driver installation includes several components which must be installed separately. You must be root to install the files and perform many of the steps contained in the following sections.

JavaPOS components

The JavaPOS RPMs are required to support Toshiba JavaPOS and RS-485 devices.

To install:

```
% rpm -ivh toshibaposs-gcc43<version>-<build>.<arch>.rpm  
% rpm -ivh toshiba-javapos-<version>-<build level>.<arch>.rpm
```

To uninstall:

```
% rpm -e toshibaposs-gcc43  
% rpm -e toshiba-javapos
```

javax.usb components

The javax.usb RPMs are required to support Toshiba USB peripheral devices.

To install:

```
% rpm -ivh javax-usb-<version>.i386.rpm  
% rpm -ivh javax-usb-ri-<version>.i386.rpm  
% rpm -ivh javax-usb-ri-linux<version>.i386.rpm
```

To uninstall:

```
% rpm -e javax-usb  
% rpm -e javax-usb-ri  
% rpm -e javax-usb-ri-linux
```

JVM

The JVM component includes two RPMs, one for the JVM itself and the other for RS-232 device support. Install both JVM RPMs.

To install:

```
% rpm -ivh ibm-java-i386-jre-<version>.i386.rpm  
% rpm -ivh ibm-java-i386-javacom-<version>.i386.rpm
```

To uninstall:

```
% rpm -e ibm-java-i386-jre-<version>
```

```
% rpm -e ibm-java-i386-javacomm-<version>
```

Installing the Java Communications API

The Java Communications API is installed in the /opt/ibm/java-<arch>-60/ directory structure.

1. Copy the javacomm files into the correct directories in your SDK.
 - a. Copy lib/libLinuxSerialParallel.so to your jre/lib/<arch>/ directory, where <arch> is the architecture of your platform.
 - b. Copy jar/comm.jar to your jre/lib/ext/ directory.
 - c. Copy lib/javax.comm.properties to your jre/lib/ directory.
2. Customize the javax.comm.properties and add extra serial ports. By default, only two serial ports are enabled. For example, to configure serial port 3:
 - a. Edit javax.comm.properties
 - b. Add the line serpath2 = /dev/ttys2

Setting up symbolic links and paths

If the operating system provides a default JVM or you have previously installed another JVM, then symbolic links and the paths must be set manually to point to IBM JVM:

1. Remove old Java symbolic links and re-link them to JVM:
 - a. rm /usr/bin/java
 - b. ln -s /opt/ibm/java-<arch>-60/jre/bin/java /usr/bin/java
2. Set up the path, as necessary. This can be added to your .profile.
 - a. export PATH=/opt/ibm/java-<arch>-60/jre/bin:\$PATH
3. Check the IBM JVM version. The output should be similar to the following; if not, check symbolic links or paths:

```
java -version
```

```
java version "1.6.0"
```

```
Java(TM) SE Runtime Environment (build pxi3260sr9fp1-20110208_03(SR9 FP1))
```

Installing Toshiba point of sale kernel mode drivers

The Toshiba JavaPOS requires several kernel mode drivers which must be installed on target systems.

For SLE 11 SP3, SUSE distributes Toshiba point of sale (POS) kernel mode drivers in the form of binaries as well as source RPMs. See “Installing Toshiba POS kernel mode drivers for SLE 11 SP3 Linux” for more information.

For non-SLE distributions, the Toshiba POS kernel mode drivers must be compiled and installed on the target OS. The driver source RPM is included in this package. See “Installing Toshiba POS kernel mode drivers for other Linux distributions” on page 13 for more information.

Installing Toshiba POS kernel mode drivers for SLE 11 SP3 Linux

For SLE 11 SP3, the Toshiba POS kernel mode drivers can be obtained and installed directly from SUSE’s website:

1. Navigate to: <http://drivers.suse.com/Toshiba/tgcs/pos/sle11sp3/i586/>.
2. Run uname -a to determine the kernel of your system (**default** or **pae**).
 - If the kernel version is “3.0.<version>-**default**” then download:
toshiba-pos-suse11-kmp-default-10.0.0_3.0.70_0.9-35.i586.rpm or higher version

toshiba-sleep-prot-suse11-kmp-default-1.0_3.0.13_0.27-0.i586.rpm or higher version

- If the kernel version is “3.0<version>-pae” then download:

toshiba--poss-suse11-kmp-pae-10.0.0_3.0.70_0.9-35.i586.rpm or higher version

toshiba-sleep-prot-suse11-kmp-pae-1.0_3.0.13_0.27-0.i586.rpm or higher version

3. Install the RPM (this example uses default):

rpm - ivh toshiba-poss-suse11-kmp-default-8.0.0_3.0.13_0.9-12.1.i586.rpm

4. Toshiba POS kernel mode drivers location:

/lib/modules/3.0.version>/updates/Toshiba/pos/aipbcd.ko

/lib/modules/3.0.version>/updates/Toshiba/pos/aipdcs.ko

/lib/modules/3.0.version>/updates/Toshiba/pos/aipikbps.ko

/lib/modules/3.0.version>/updates/Toshiba/pos/aipmtn.ko

/lib/modules/3.0.version>/updates/Toshiba/pos/aipsocdkl.ko

/lib/modules/3.0.version>/updates/Toshiba/pos/aipsokbps.ko

/lib/modules/3.0.version>/updates/sleep-prot.ko

Installing Toshiba POS kernel mode drivers for other Linux distributions

For other Linux distributions, the Toshiba POS kernel mode drivers must be compiled on the specific kernel version.

The source code for the Toshiba POS kernel mode driver is included in:

toshibaposs-kernel-<version>-<build>.〈arch〉.rpm

This RPM is located under the pos_kernel_drivers_other-linux directory within the JavaPOS driver package.

Install kernel source (prerequisite)

To compile Toshiba drivers successfully, the kernel source code must be installed first. The kernel source is available on the installation CD or from the location where you obtained the kernel. If the kernel source does not exist, the Toshiba drivers will not compile successfully.

Extract Toshiba POS driver source

% rpm -i toshibaposs-kernel-<version><build>.〈arch〉.rpm

This extracts the driver source files into two separate directories:

- /usr/src/<kernel-version>//kernel-modules/toshiba/dcs/
- /usr/src/<kernel-version>//kernel-modules/toshiba/kbd/

Build and install Toshiba POS drivers

Note: Before proceeding with building Toshiba drivers, ensure that you have installed Linux kernel sources.

1. Build and install drivers in the dcs directory:

- % cd /usr/src/<kernel-version>//kernel-modules/toshiba/dcs
- % make (to compile drivers)
- % make install (to install drivers)
- % depmod -ae (this must be done to satisfy module dependency in modules.def file)

2. Build and install drivers in the kbd directory:

- % cd /usr/src/<kernel-version>//kernel-modules/toshiba/kbd

- % make (to compile drivers)
- % make install (to install drivers)
- % depmod -ae (this must be done to satisfy module dependency in modules.def file)

Toshiba driver installation location

The drivers are installed in the following locations:

/lib/modules/<kernel-version>/kernel/drivers/char/dcs
 /lib/modules/<kernel-version>/kernel/drivers/input/keyboard

Table 5. Driver details

Driver Name	Description
aipdcs.ko	Core driver for RS-485 devices, NVRAM, PCI Cash Drawer
aipbcd.ko	Cash Drawer driver for SP300
aipmtn.ko	Motion sensor driver for AnyPlace Kiosk
aipikbps.ko	PS/2 keyboard driver for Toshiba POS Keyboard
aipsocdkl.ko	SurePOS 100/SureOne: Cash Drawer and Keylock driver
aipsops.ko	SurePOS100/SureOne: keyboard driver

Systems Management installation

Systems Management support for Toshiba POS peripherals

The Toshiba JavaPOS supports Systems Management capability for Toshiba POS peripherals. Refer to the following documents and information included in the package:

- Toshiba UnifiedPOS Management Services.pdf
- Minimum version of SFCB RPM is 1.3.13-3.1.2
- RMA information can be found at this location: <http://www-01.ibm.com/support/docview.wss?rs=219&uid=pos1R4000158>

Installing on SLE 11 SP3

The Toshiba UnifiedPOS support for Systems Management is supported through the Small Footprint CIM Broker (SFCB). This RPM is available on the SUSE installation DVD or from the SUSE repository:
<http://ftp5.gwdg.de/pub/opensuse/discontinued/distribution/12.1/repo/oss/suse/i586/sblim-sfcb-1.3.13-3.1.2.i586.rpm>.

Note: You must be a root user to install and configure the Systems Management components.

1. Install posIBM_XML4C-5.7.1-1.i586.rpm.

```
rpm -ivh posIBM_XML4C-5.7.1-1.i586.rpm
```
2. Install SFCB RPM from the SUSE installation DVD or SUSE repository.
 - a.

```
rpm -ivh sblim-sfcb- 1.3.13-3.1.2.i586.rpm
```
 - b. Edit /etc/sfc/sfc.cfg to allow http communication and authentication. For example, edit the following two properties (refer to <http://sblim.sourceforge.net/wiki/index.php/Sfc> for more details):


```
enableHttp: true
doBasicAuth: false
```
 - c. Start SFCB with /etc/init.d/sfc start

Note: Known issue: These instructions do not work with SFCB 1.3.11-0.19.1 version and before due to an issue with the SFCB server. This problem was fixed in newer versions.

3. Install `toshiba_upos_sblim-cmpi-upos-server-sled-<version>. <arch>.rpm`:
`rpm -ivh toshiba_upos_sblim-cmpi-upos-server-sled-<version>. <arch>.rpm`
4. Install the `wbemcli` utility from the SLE 11 SP3 DVD.
`rpm -ivh sblim-wbemcli-1.6.1-1.1.84.i586.rpm`

Note: This utility is also available for download from: <http://sblim.sourceforge.net/wiki/index.php/Wbemcli>

Validating Systems Management

To view the Systems Management properties for a device, the device must be opened and claimed.

To open, claim, or enable a device in JavaPOS, start the POS Control Center utility:

1. Open a terminal window.
2. Issue: `POSControlCenter`.
3. To configure devices, click **AutoDetect** and save `jpos.xml` to the default location.
4. Select an online device and click the **Systems Management** tab.
5. Click **Start Statistics Test**. This displays the Systems Management properties of the device and will keep the device in an open/claim/enable state.
6. Get Systems Management properties for a given device by issuing the `wbemcli` command. For example, for a POSPrinter device:

```
wbemcli ei http://localhost:5988/root/cimv2:UPOS_POSPrinter
```

Uninstallation of UPOS Systems Management

1. Remove “`toshiba_upos_sblim-cmpi-upos-server-sled-<version>.i586.rpm`”
`rpm -e toshiba_upos_sblim-cmpi-upos-server-sled-<version>-<build>`
2. Remove SFCB RPM.
`rpm -e sblim-sfcb`
3. Remove `posIBM_XML4C` RPM.
`rpm -e posIBM_XML4C`

Chapter 3. Configuring devices

This chapter contains information on the configuration process for JavaPOS devices, OPOS devices, and USB access.

JavaPOS configuration

JavaPOS devices are configured using the `jpos.xml` file. This section details how to create a `jpos.xml` file and provides information necessary to configure certain devices.

Locating sample `jpos.xml` files

During installation, a sample XML file is installed on your system. The sample file is named `jposSample.xml`.

This file can be found in the following folders:

- Linux: `/opt/ibm/javapos/config`
- Windows: `C:\pos\ibmjpos\docs`

The sample XML files contain JPOS entries for all JavaPOS-supported devices.

Creating the `jpos.xml` file

The `jpos.xml` file can be created using one of the following two methods.

Using a text editor

To create a `jpos.xml` file using a text editor, follow these steps:

1. Create an empty `jpos.xml` file.
2. Copy the header information from the sample XML file into the new `jpos.xml` file.
3. Copy only those `JposEntries` that correspond to devices on the target system from the sample XML file into the `jpos.xml` file.
4. Change the Logical Name in each `JposEntry` so that it matches the Logical Name for that device in the JavaPOS application.
5. Save the `jpos.xml` to the `pos\ibmjpos\config` directory in Windows, or to the `/opt/ibm/javapos/etc` directory in Linux. Update the CLASSPATH if the `jpos.xml` file is saved to another location.
6. In the new `jpos.xml` file, keep only those `JposEntries` that correspond to devices on the target system.

Using the JavaPOS Configuration Editor

Create a `jpos.xml` file using the JavaPOS Configuration Editor by following these steps:

1. Copy the sample XML file to the `jpos.xml` file.
2. Change the directory to the location of the `jpos.xml` file.
3. Open the JavaPOS Configuration Editor by typing:
`java jpos.config.simple.editor.JposEntryEditor`

Note: The JavaPOS Configuration Editor can also be accessed from **Programs > Toshiba JavaPOS > JavaPOS Configuration Editor**.

4. Change the Logical Name in each `JposEntry` to match the Logical Name for that device in the JavaPOS application.
5. In the new `jpos.xml` file, keep only those `JposEntries` that correspond to devices on the target system.

Changing the logical name of JposEntry

Follow these steps to change the logical name of JposEntry:

1. Select a device entry.
2. Click **Edit** and select **Copy**.
3. Enter a new Logical Name in the JposEntry dialog box. Click **OK**.
4. Click **Yes** to delete the old entry.
5. For each device that is not on the target machine, perform the following steps:
 - a. Click the JposEntry for that device.
 - b. Click **Edit** and select **Delete**.
6. Click **File** and select **Save JposEntryRegistry As...** to save the changes to `jpos.xml`.

Device configuration

Devices are configured based on the entries in the `jpos.xml` file. Because JposEntries consume resources, it is recommended that `jpos.xml` contain only those JposEntries that correspond to devices on the target system. For example, if you are working with only EIA-232 devices, the `jpos.xml` should not include entries for RS-485 or USB devices.

The logical name used for a given device during open must match the `LogicalName` specified in the JposEntry in the `jpos.xml` file for that device.

The entries in the `jpos.xml` file are organized by device type.

Network device communication

The network devices for JavaPOS need to be configured in two places using this order:

1. In the network configuration file.
2. In the `jpos.xml` file.

The `DeviceAddress` or `DeviceName` used for a given device must match the `'com.ibm.posj.bus.network.hostName'` property specified in the JposEntry in the `jpos.xml` file for that device.

Device Support: 4610 2xR/1NR Ethernet models only, when available.

The JavaPOS network devices can be configured using the following information:

1. To add a network POSPrinter to the drivers configuration, copy the `ethernetPrinterSample.efg` file to the configuration folder:
 - Linux:
 - From: `/opt/ibm/javapos/config`
 - To: `/usr/share/pos/config/`
 - Windows:
 - From: `C:\POS\doc`
 - To: `C:\POS\Config`
2. Edit the file to add the `DeviceAddress` or `DeviceName` property of your network device.

For example:

- To configure one printer with device address:

```
[4610Printer]  
DeviceAddress: 169.254.100.10
```

- To configure one printer with device name:

```
[4610Printer]  
DeviceName: devicename.domain.com
```

- To configure two printers with device address and device name:
[4610Printer]
DeviceAddress: 169.254.100.10
[4610Printer]
DeviceName: devicename.domain.com
- Optional:** Rename the `ethernetPrinterSample.efg` file to your preferred name, such as `"posprinter.efg"`. The file should contain the `.efg` extension.
 - Configure your `jpos.xml`. See “Creating the `jpos.xml` file” on page 17.
 - To broadcast this configuration to other systems, transfer the `posprinter.efg` and `jpos.xml` files to clients.

EIA-232 device communication

Entries for the EIA-232 devices in the `jpos.xml` file include the EIA-232 communication configuration properties. Acceptable values for each EIA-232 communication property are documented in Table 6.

Table 6. EIA-232 device communication properties

JCL property name	Type	Possible values (typical values)	Default Value
portName	String	Valid port name such as: "COM1"	"" (empty string)
baudRate	String	"2400", "4800", "7200", "9600", "14400", "19200", "38400", "57600", "115200"	"9600"
dataBits	String	"4", "5", "6", "7", "8"	"8"
parity	String	"Even", "Odd", "None", "Mark", "Space"	"None"
stopBits	String	"1", "1.5", "2"	"1"
flowControl	String	"Xon/Xoff", "Hardware", "None"	"None"

Note:

- The value for the `portName` depends on the POS system to which the device is attached. All the ports start with the letters "COM"; a number is appended to identify the port, for example: "COM2".
- In Linux, the designations COM1 and COM2 are still used instead of the standard Linux terminology.

Device-specific EIA-232 configuration

Possible values for the device communication properties for specific EIA-232 devices are shown in Table 7 on page 20. The `dataBits`, `parity` and `stopBits` properties are omitted from the table and from `jpos.xml` entries.

Note:

- Values different than those described here will cause the device to fail.
- In the following table, Not required means that the user does not have to type these values into the XML.
- For embedded devices in the 4610 Printer such as magnetic ink character recognition (MICR) and CheckScanner, the `jpos.xml` must include the JposEntry for the 4610 Printer and the embedded devices.

Table 7. Specific values for EIA-232 devices in JavaPOS

Device	portName	baudRate	flowControl
CashDrawer 4610-A CashDrawer 4610-B	Port where the printer is attached "COM1" (default)	"9600" "19200" "115200" (Must match printer configuration)	Only "Xon/Xoff" or "Hardware" (Must match printer configuration)
CashDrawer SurePOS 500/600-A CashDrawer SurePOS 500/600-B	"COM4" (default)	Not required	Not required
CheckScanner 4610-TI8/TI9	Port where the printer is attached Default: "COM1"	"9600" "19200" (Must match printer configuration)	Only "Xon/Xoff" or "Hardware"
Fiscal Printer 4610-Kx3 Fiscal Printer 4610-Kx4 Fiscal Printer 4610-Kx5/Kx6 Fiscal Printer 4610-Gx3 Fiscal Printer 4610-Gx4 Fiscal Printer 4610-Gx5	Port where the printer is attached Default: "COM1"	"9600" "19200" "28800" (Windows only)	Only "None"
Fiscal Printer 4610-Sx6	Port where the printer is attached Default: "COM1"	"115200"	Only "None"
Line Display APA	Default: "COM4"	"9600"	"Hardware"
Line Display SureOne/SurePOS 100/300/500/600 VFD Line Display SureOne/SurePOS 100/300/500/600 LED	Default: "COM4"	Not required	Not required
MICR 4610 TI2/4/8/9 2CR	Port where the printer is attached Default: "COM1"	"9600" "19200" (Must match printer configuration)	Only "Xon/Xoff" or "Hardware"
MSR 4820/AnyPlace Kiosk/SurePOS 500/600 ISO MSR 4820/AnyPlace Kiosk/SurePOS 500/600 JUCC	Default: "COM3"	"19200"	Not required
POSPrinter 4610 T11/2/3/4/5/8/9 TM/F 6/7 2xR/1NR	Port where the printer is attached Default: "COM1"	"9600" "19200" (Must match printer configuration)	Only "Xon/Xoff" or "Hardware"

Table 7. Specific values for EIA-232 devices in JavaPOS (continued)

Device	portName	baudRate	flowControl
POSPrinter SureOne	Only "COM3"	"9600"	Only "Xon/Xoff"
Toshiba Anyplace Kiosk Line Scanner	Default: "COM4"	"19200"	Not required
Toshiba Anyplace Kiosk Omni Scanner	Default: "COM4"	"9600"	Not required
Tone Indicator 4610 TM/TF6/7 1NR	Port where the printer is attached Default: "COM1"	"9600" "19200" (Must match printer configuration)	Only "Xon/Xoff" or "Hardware"

POS Control Center utility

The POS Control Center displays information for devices found on the system and from device entries from the `jpos.xml` file. For multiple, identical devices such as cash drawers and displays, the `deviceNumber` property becomes important. For a given device, the device number displayed in the Device Information section of the POS Control Center must match the `deviceNumber` in the `JposEntry`.

To access the Toshiba POS Control Center:

- In Windows, click **Programs > Toshiba JavaPOS > POS Control Center**.
- In Linux, enter `POSControlCenter`.
- From a command prompt in either operating system, enter:
`java com.ibm.jpos.tools.sdicc.ControlCenterApp`

Toshiba SurePoint Display (4820) touch support

Touch support is not provided for the following:

1. Toshiba SurePoint Display (4820) RS-485 device in Windows Vista
2. IBM Retail Environment for SuSE Linux Version 2

USB device access

Microsoft Windows

Toshiba peripherals are USB 1.1 compliant, which can cause communications problems when connected to a USB 2.0 controller. It is strongly recommended that USB 2.0 be disabled in the terminal BIOS settings.

Important: The exception to this is the Toshiba 4610 2NR/2CR printer, which is a 2.0 compliant device and does not require that USB 2.0 be disabled.

Linux

The Toshiba JavaPOS drivers use the `javax.usb` subsystem to access the USB POS devices connected to the system. The `javax.usb` subsystem on Linux uses the USB filesystem, `usbfs`, to access the USB devices directly. The permissions on this filesystem default to access by the root user only. This security model prevents unauthorized users from accessing USB devices connected to the system. However, if the JavaPOS application (for example, JVM) is running as a non-root user, the `usbfs` permissions must be changed to allow non-root access.

To allow non-root access:

- The mount parameters for `usbfs` must be changed. Complete one of these two methods:

- Edit the `/etc/fstab` file to modify the mount parameters for `usbfs` (fourth field).
- Manually remount `usbfs`, specifying the required parameters on the `mount` command.

The most secure way to accomplish non-root access to the USB device nodes is to change the ownership of the USB device nodes to the userid which is running the JavaPOS application. To do this:

- Mount `usbfs` with the parameter, `devuid=n`, where *n* is the numeric ID of the JavaPOS application's user.
 - For example, if the JavaPOS application is running as user `javaposapp`, and `javaposapp` has a User ID (uid) of 1000, then specify `devuid=1000` when mounting `usbfs`. The user running the JavaPOS application must be considered a privileged user because this user can directly access all USB devices.
 - Users who do not need to run the JavaPOS application or the JavaPOS drivers should use a different login.

A less secure way to accomplish non-root access to the USBdevice nodes is to create a special group for the users who run JavaPOS applications. Two parameters must be used in mounting `usbfs`: `devgid=n`, where *n* is the numeric group id for the JavaPOS application group, and `devmode=0664`, which changes the permissions of the USB device nodes to read/write by owner and group. The group given permission to access the USB devices should be considered a privileged group and handled the same as a privileged user. Only those users that need to run JavaPOS applications and drivers should be made part of that group.

See the Linux manual pages for `useradd`, `usermod`, `groupadd`, and `groupmod` for information about creating and modifying users and groups.

Network configuration tool

The discovery feature of the network device configuration dialog in JPOS Control Center requires the network to be configured in the following way to work correctly:

- The network configuration (switch/router, etc.) should allow UDP broadcast.
- Each terminal should have any potential firewall software configured to listen for UDP broadcasts, and to allow access to UDP port 9100.
- The printer, and each terminal that is used for network discovery, needs the same class of IP address and the same network subnet mask.

OPOS configuration

Toshiba OPOS drivers must be configured to access the point-of-sale devices attached to the system. OPOS provides defaults for all resources associated with supported devices. This section contains information on how the application can configure a device to use a value different from the assigned default and how to specify some resource values.

To locate the Configuration Utility:

- Select **Programs > Toshiba UnifiedPOS for Windows > OLE for Retail Point of Sale > Configuration Utility.**

Network device communication

To configure network devices for OPOS, the network configuration file must be properly configured and set up at the correct location.

Device Support: 4610 2xR/1NR Ethernet models only, when available.

The OPOS network devices are configured in the following way:

1. To add a network POSPrinter to the driver configuration, copy the ethernetPrinterSample.efg file to the configuration folder.

From: C:\POS\doc
To: C:\POS\Config

2. Edit the file and add the DeviceAddress or DeviceName property of your network device.

For example:

To configure one printer with device address:

[4610Printer]
DeviceAddress: 169.254.100.10

To configure one printer with device name:

[4610Printer]
DeviceName: devicename.domain.com

To configure two printers with device address and device name:

[4610Printer]
DeviceAddress: 169.254.100.10
[4610Printer]

DeviceName: devicename.domain.com

Optional: Rename the ethernetPrinterSample.efg to your preferred name, such as "posprinter.efg". The file should contain the .efg extension.

Note: After the file is renamed, you must restart the service or reboot the computer in order for the .efg to take effect.

Using the OPOS Configuration utility

The OPOS configuration window has a separate tab for each of the four POS Terminal types supported by the Toshiba OPOS package (see Figure 3 on page 24).

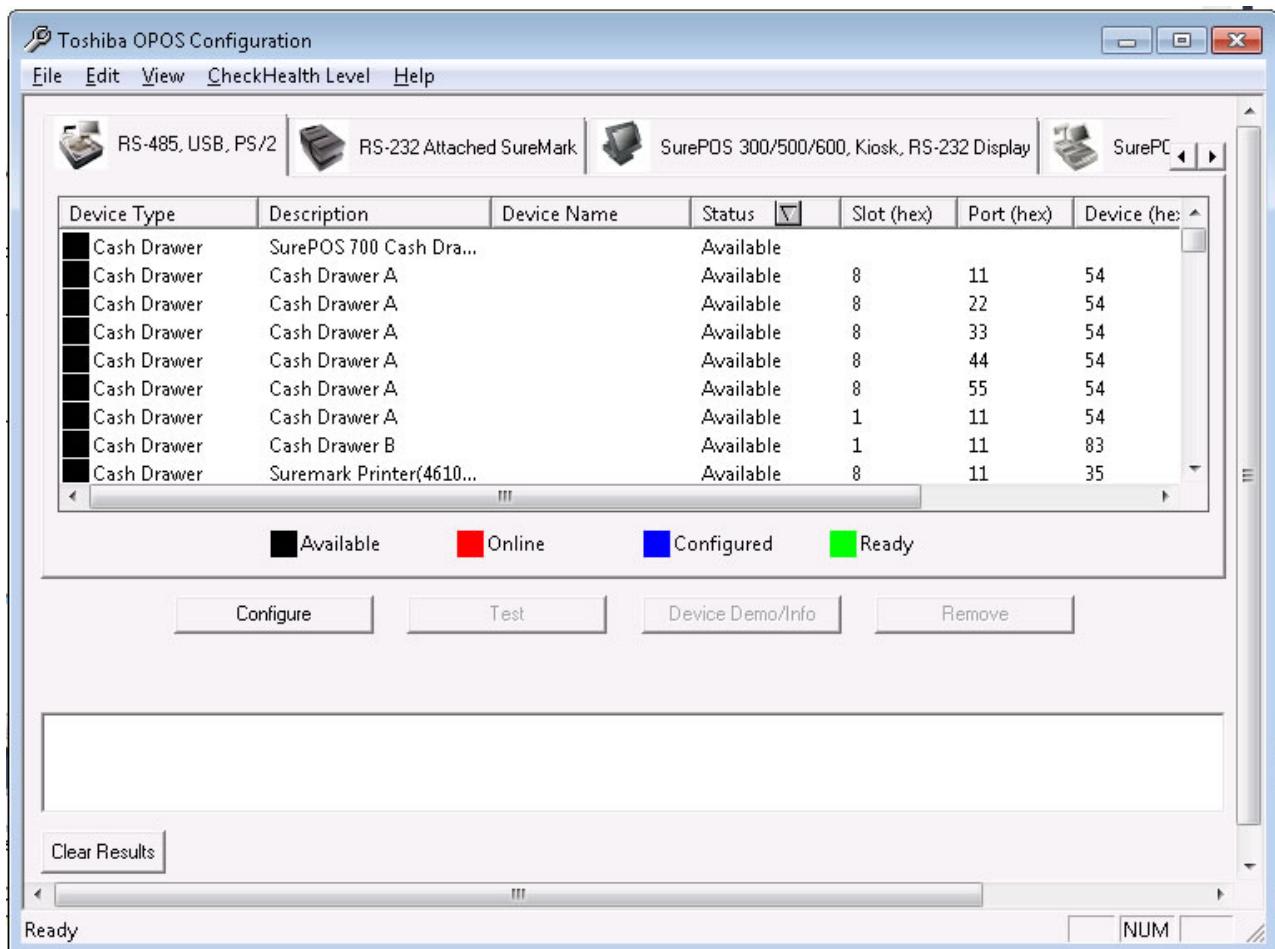


Figure 3. OPOS Configuration utility window

Each tab contains a list of devices that are supported on the POS Terminal. Each device entry contains descriptive information and the current status of the device. The status shows what level of configuration is completed using a status description. Table 8 explains the meaning of each status type.

Table 8. Device status types

Status	Color	Description
Available	Black	The device is supported on the selected POS Terminal. It can only be used to add a new configuration. Its settings will not be modified. No device name is given.
Online	Red	The device is supported on the selected POS Terminal, and is detected to be online and available on the current system. It can only be used to add a new configuration. Its settings will not be modified. No device name is given.
Configured	Blue	The device is supported on the selected POS Terminal and has a configuration entry in the registry. It can be modified, tested for connectivity, or removed.
Ready	Green	The device is supported on the selected POS Terminal and is ready to use. It is online and has a configuration entry in the registry. It can be modified, tested for connectivity, or removed.

RS-485, PS/2, and USB devices are automatically detected and listed as Online. The system tone and HardTotals are always listed as Online. Other devices can be listed as Online after testing.

Navigation

To find specific device entries, a column can be sorted by clicking its heading. The column is sorted in ascending order first; click again to sort in descending order. The current sort order is indicated by the up or down arrow in the column heading.

Note: The order is based on ASCII values and only one column can be sorted at a time. For example, a baud rate of 19200 is listed before 9600 when the column is sorted in ascending order.

Configuring devices

To configure a device, perform the following steps:

1. Select an entry in the device list.
2. Click the **Configure** button to display the configuration dialog for the selected device. The dialog differs depending on the POS Terminal type and the device type. Figure 4 on page 26 shows the configuration dialog for an RS-485 or USB device; Figure 5 on page 27 shows the configuration dialog for an EIA-232 device.
3. Enter a name for the device.
4. For USB, RS-485, and System devices, select the correct slot and port for the device (see Figure 4 on page 26). The slot can have the following values:

Slot	Description
0	System devices (keyboard, NVRAM)
1	Most RS-485 devices
2	Some RS-485 devices (4674 devices)
8	USB devices

The port can have the following values:

Port (hex)	Description
0	System devices (keyboard)
11	Most RS-485 devices
22	RS-485 devices (46x3 Model 2)

Note: The device number cannot be modified. That number is specific to the type of device, and changing it could change the device type.



Figure 4. Device properties dialog (RS-485/USB)

For EIA-232, select the correct Com Port, Baud Rate, and Flow Control for the device (See Figure 5 on page 27).

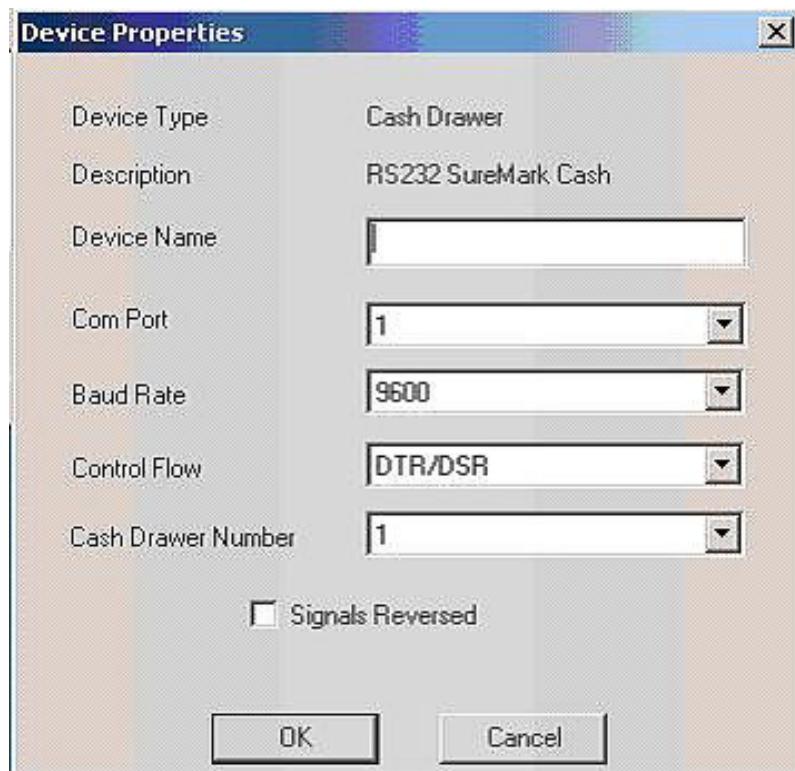


Figure 5. Device properties dialog (EIA-232)

5. Click **OK**. The entry is added to the device configuration list.

Note:

1. Devices that are not attached can still be configured.
2. For RS-485 and USB devices, Slot and Port should not have to be changed.
3. RS-232 devices will not show as Online in the OPOS configuration tool until they are configured.
4. For more information about Slot, Port, and Device numbers, see the *Point of Sale Subsystem: Programming Reference and User's Guide*.

Testing connectivity

Selecting a Configured or Ready device enables the **Test**, **Remove**, and **Configure** buttons.

Clicking on the **Test** button tests the connectivity and configuration of the device. While the device does not need to be attached to configure or remove a configuration entry, it must be attached for a successful test. The **Test** button uses OPOS to open, claim when necessary, and enable the device. It then performs a CheckHealth method. The CheckHealth level is set from the menu.

If all of these steps are completed successfully, then the test is considered successful. If the device configuration entry is not yet marked as Ready, it is updated as such.

The list box at the bottom of the window is updated with the test results (see Figure 6 on page 28).

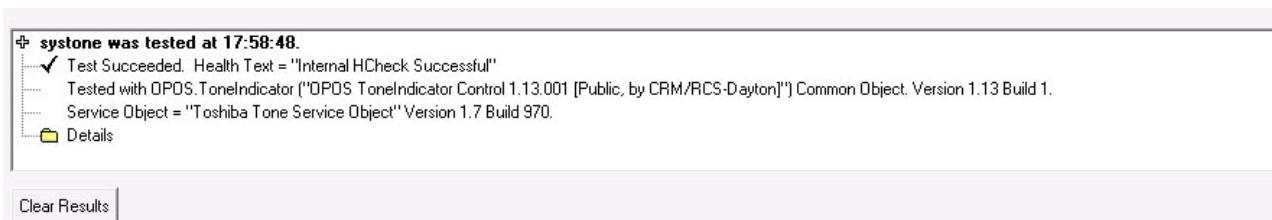


Figure 6. Results of connectivity test

By selecting a configured device, additional tests can be performed. Additional device information can also be obtained by clicking the **Device Demo/Info** button. For example, a sample receipt can be printed by using the following steps:

1. Click the **Device Demo/Info** button while selecting a configured POS Printer.

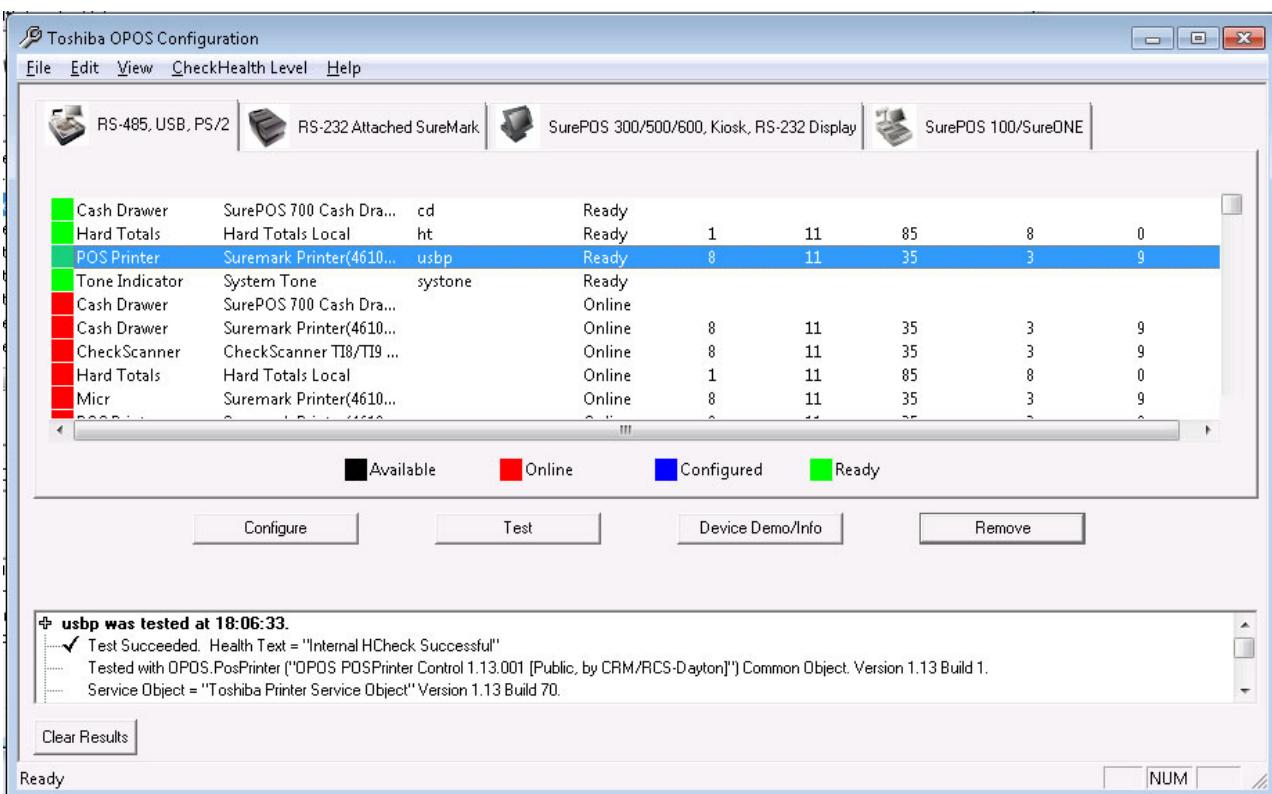


Figure 7. Device Demo/Info Button

2. Click the **Device Demo Tab**.

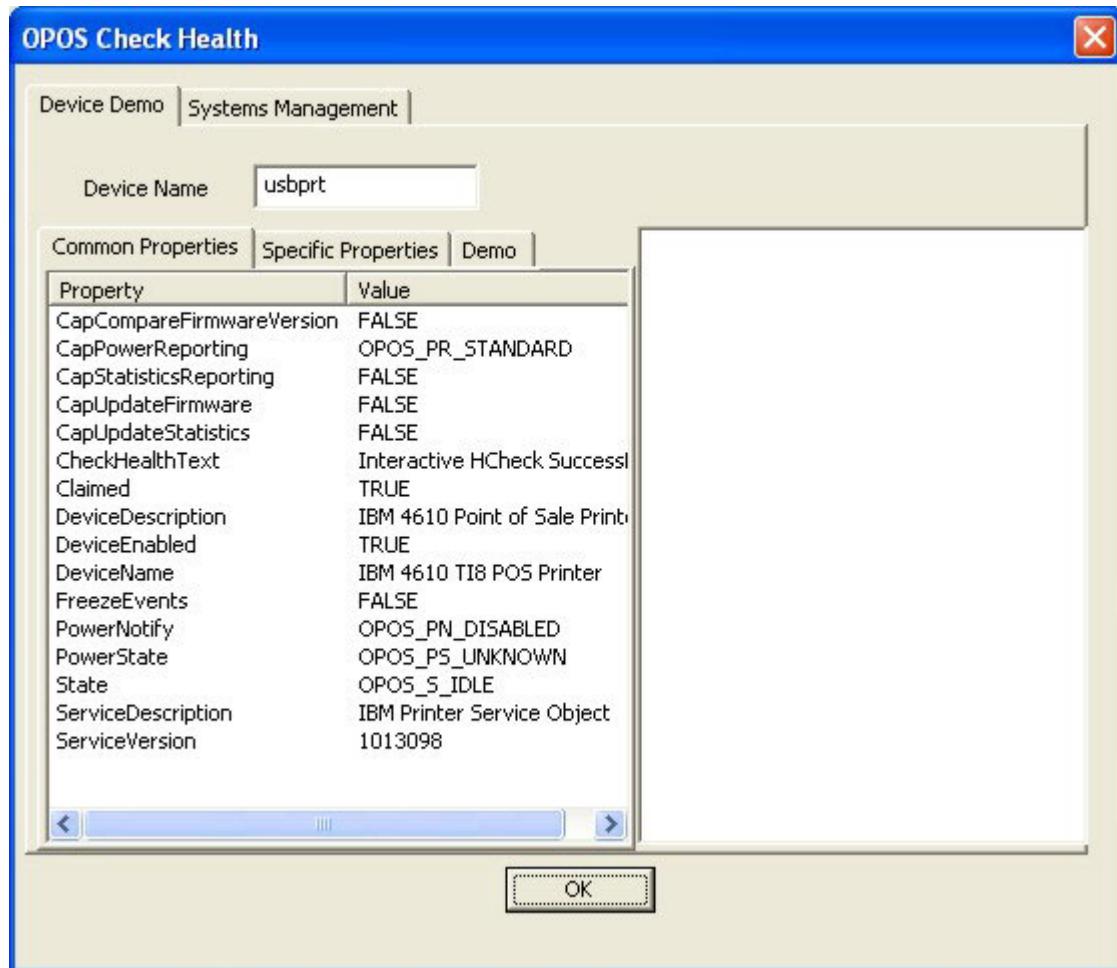


Figure 8. Demo Tab

3. Click the Print Test Receipt button. This prints a sample receipt.

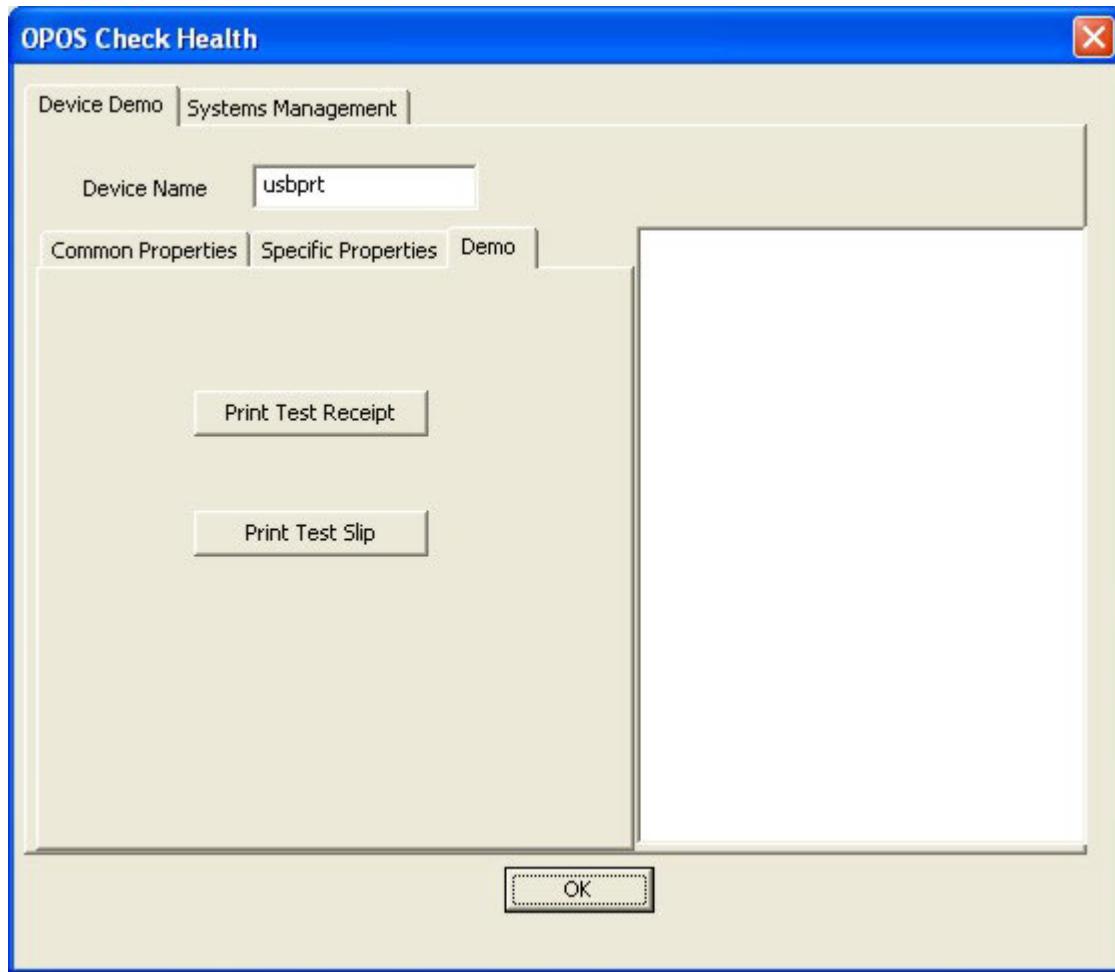


Figure 9. Print Test Receipt Button

Removing devices

It is possible to delete entries during configuration. The results are displayed in the list box at the bottom.

Deploying OPOS device configurations

The OPOS Configuration utility is used to import and export configuration information to and from an OPOS Device Registry (ODR) file. These options are available on the **File** menu.

To export configuration information, perform the following steps:

1. On each tab, select the devices that are to be exported. Use the **Shift** and **Ctrl** keys to select more than one. (See Figure 10 on page 31.)

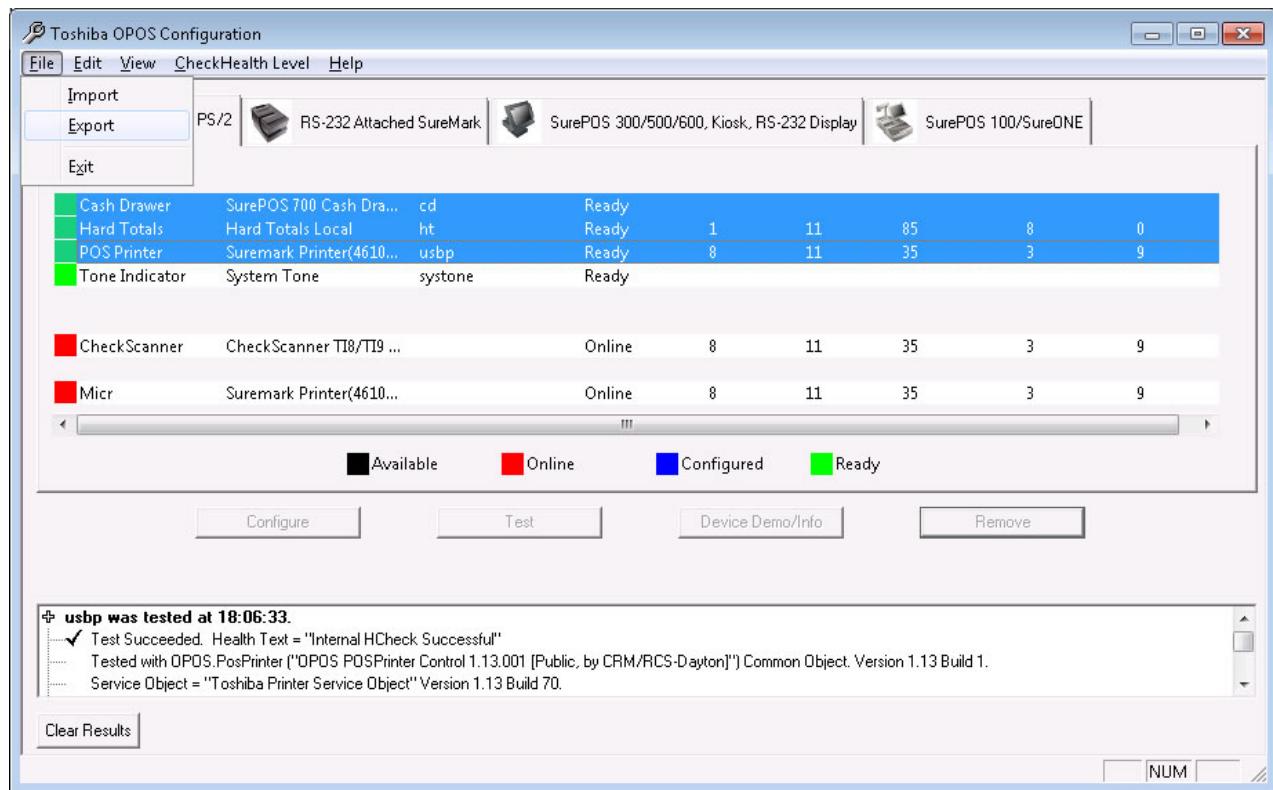


Figure 10. Selecting devices to export

- Select **File > Export**. The configuration information is exported.

Only configured entries with device names are exported. This enables the creation of files for different configurations, or for deploying new configurations to stores.

To import configuration information, perform the following steps:

- Select **File > Import**. A file dialog window is displayed.
- Select an ODR file and click **Open**. The configuration information is imported and the device listing and registry information are updated.

Toshiba SurePoint Display (4820) touch support

Touch support is not provided for the Toshiba SurePoint Display (4820) RS-485 device on Windows Vista.

Chapter 4. Problem determination

This chapter contains information on the logging files that record device events. These events allow you to determine and resolve problems.

JavaPOS problems

Tracing for Linux

For Toshiba JavaPOS device drivers, the following two methods are used to gather trace information:

- Using the `aiptrace.cfg`:
 1. Create the `aiptrace.cfg` file at `/usr/share/pos/config/` folder
 2. Rename the `aiptrace.cfg.allEnabled` to `aiptrace.cfg`
 3. For minimum tracing functionality, the following properties should be set:
 Enable: On
 StartupMode: Enabled
 <Tracer name>: On
 4. Output log file default location is `/var/log/ibmposstrc.log`
- Using the TraceConfigTool:
 1. Using a command line prompt, type in `TraceConfigTool`. The Trace Configuration window appears:

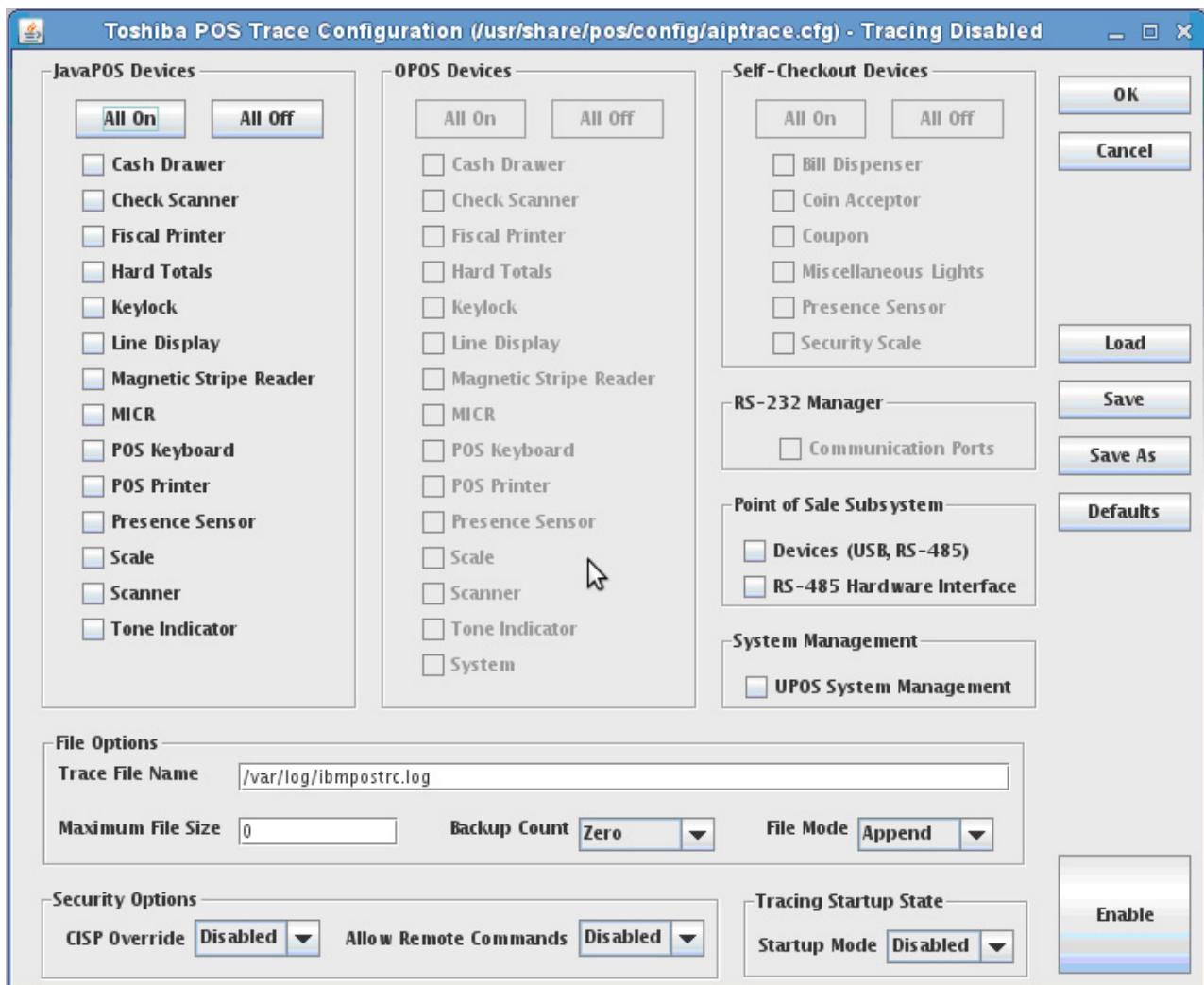


Figure 11. Trace Configuration

2. Choose the items requested for tracing.
3. When trace configuration is complete, click **Enable** and then click **Save**.

Tracing for Javax.usb for Linux

The Linux implementation of javax.usb has JNI tracing only. To change the settings for JNI tracing, edit the `javaxusb.properties` file. The available settings are:

- `#com.ibm.jusb.os.linux.LinuxUsbServices.JNI.tracing = true`
A value of false disables tracing entirely; a value of true enables some amount of tracing.
- `#com.ibm.jusb.os.linux.LinuxUsbServices.JNI.trace_output = 2`

This defines where the JNI tracing output is sent. The default is stderr file mode. Note that if append mode is used, the trace file will grow larger every time javax.usb is run; its size should be managed. In file mode, the file size is not managed or limited by javax.usb.

- 1 - stdout
- 2 - stderr
- 3 - file (truncate mode)
- 4 - file (append mode)
- `#com.ibm.jusb.os.linux.LinuxUsbServices.JNI.trace_filename =`

This is the filename to trace to, if trace_output is set to file tracing. There is no default. It must be set if trace_output is set to file tracing.

- `#com.ibm.jusb.os.linux.LinuxUsbServices.JNI.trace_level = 0`

This sets the tracing level. Higher levels mean more tracing. This level applies to ALL tracers. See each tracer for their levels.

- `#com.ibm.jusb.os.linux.LinuxUsbServices.JNI.trace_default = true`

This enables or disables default tracing. This is the most used tracer. The trace_data must also be enabled. The levels for this tracer are:

- 0 - CRITICAL
- 1 - ERROR
- 2 - INFO
- 3 - FUNCTION
- 4 - DEBUG
- 5 - OTHER

- `#com.ibm.jusb.os.linux.LinuxUsbServices.JNI.trace_hotplug = true`

This enables or disables hotplug tracing. The trace_data must also be enabled. These are the levels for this tracer:

- 0 - CRITICAL
- 1 - ERROR
- 2 - CHANGE
- 3 - DEVICE
- 4 - OTHER

- `#com.ibm.jusb.os.linux.LinuxUsbServices.JNI.trace_xfer = true`

This enables or disables xfer tracing. The trace_data must also be enabled. These are the levels for this tracer:

- 0 - CRITICAL
- 1 - ERROR
- 2 - REQUEST
- 3 - METADATA
- 4 - DATA
- 5 - OTHER

-

`#com.ibm.jusb.os.linux.LinuxUsbServices.JNI.trace_urb = false`

This enables or disables URB tracing. The trace_data must also be enabled. These are the levels for this tracer:

- 2 - METADATA
- 3 - DATA

Note: DATA-level tracing will generate a LOT of output.

The default settings are tracing enabled at level 0. The level should be increased to the desired level. Also, the output can be changed from stdout to a file or stderr. Finally, the default is not to trace actual URB data due to the large amount of trace information involved. If URB data should be traced, the trace_urb must be changed to "true" (and uncommented).

Tracing for Windows

For Toshiba JavaPOS device drivers, the following steps are used to gather trace information.

1. Using a command line prompt, type in AIPTRCCFG. The Trace Configuration application appears:

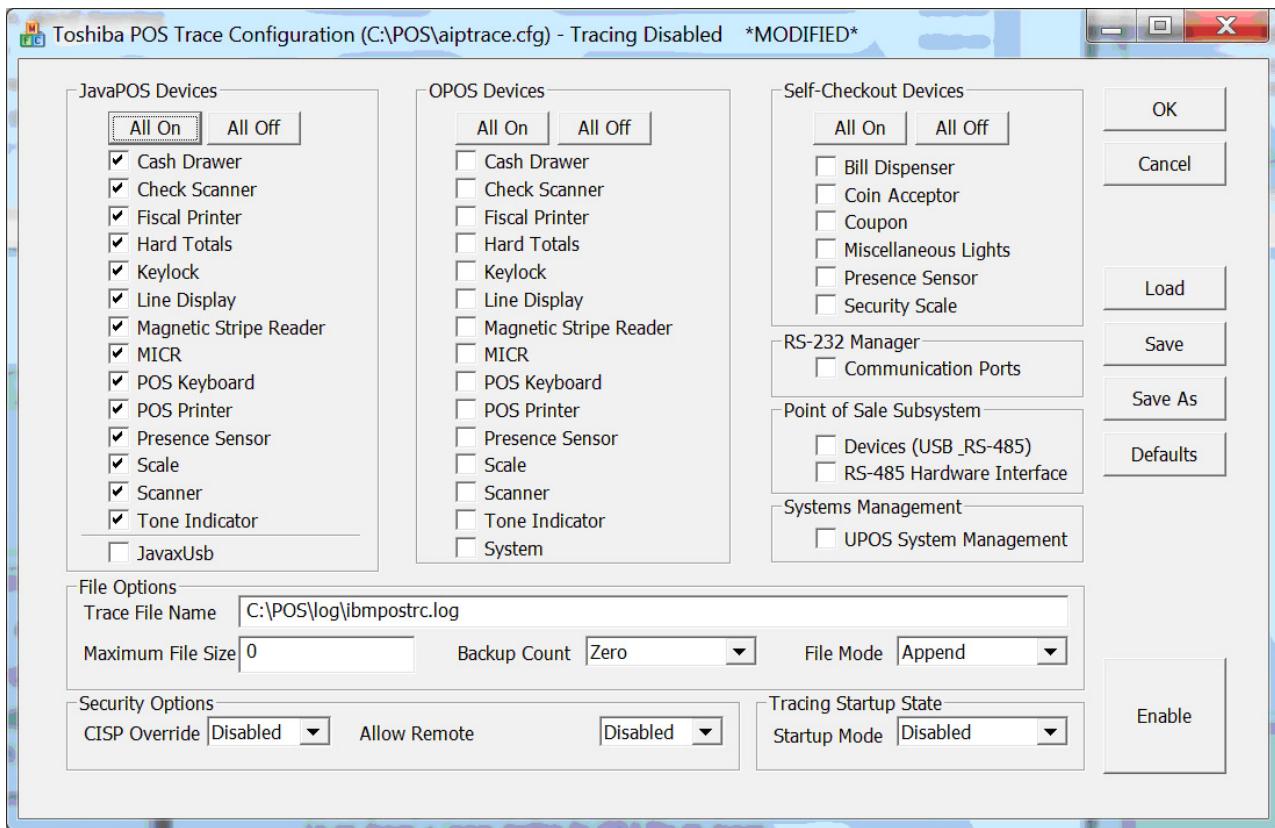


Figure 12. Trace Configuration for Windows

2. Choose the items requested for tracing.
3. When trace configuration is complete, click **Save** and then click **Enable**.

OPOS problems

Tracing

For Toshiba OPOS device drivers, the following steps are used to gather trace information.

1. Using a command line prompt, type in AIPTRCCFG. The Trace Configuration application window appears.

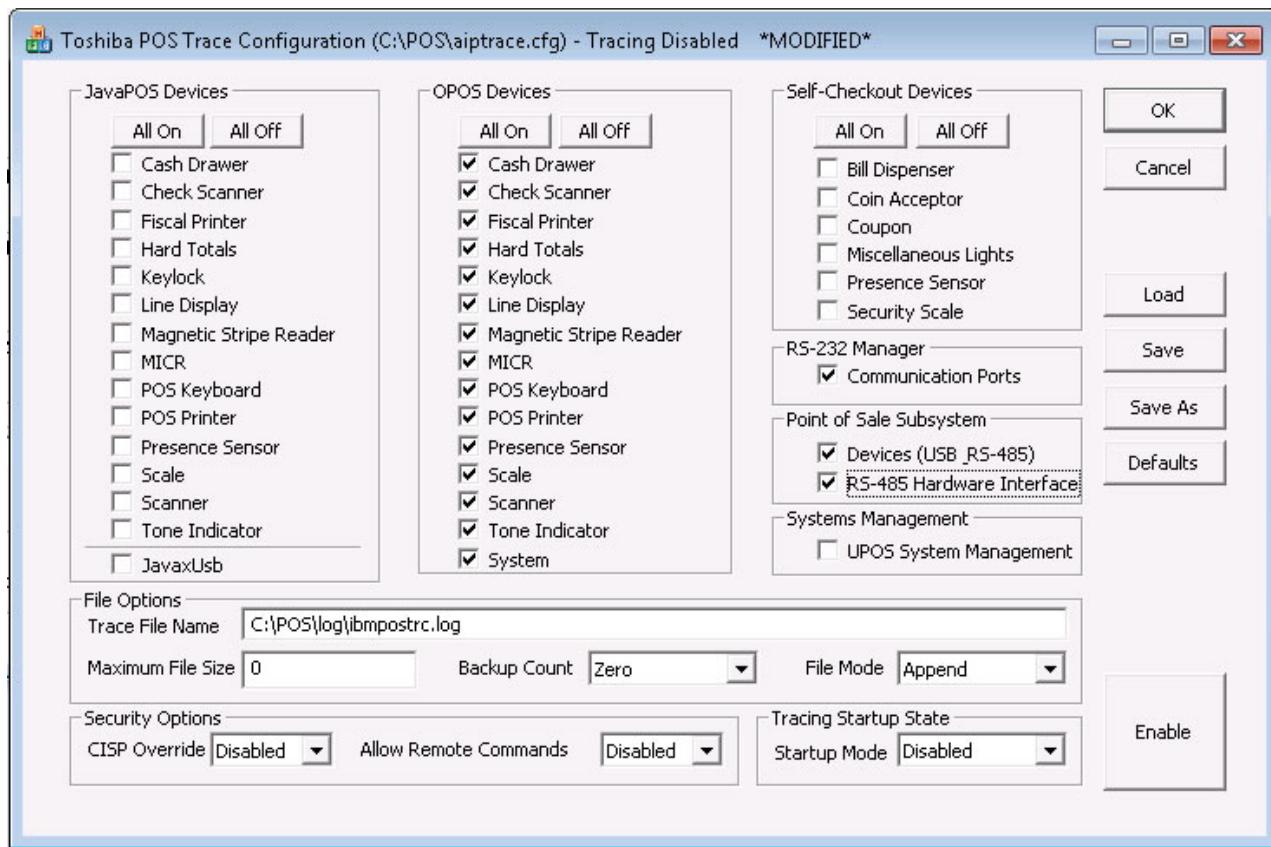


Figure 13. POS tracing facility

2. Choose the items requested for tracing.

Note:

- a. When diagnosing printer problems, it is normally recommended to enable tracing for all logical printer devices (cash drawer, check scanner, MICR, POS Printer, Tone Indicator).
- b. When diagnosing OPOS problems, remember to check the Communications Ports checkbox (for EIA-232 devices) or Devices checkbox (for USB and RS-485 devices).
- c. The Maximum File Size is measured in bytes; 0 implies no maximum.

3. When trace configuration is complete, click **Save** and then click **Enable**.

Getting help

Support website

The Toshiba Global Commerce Solutions website contains the latest version of the Toshiba OPOS software as well as fixes to known problems, hints, and tips for using the software: www.toshibacommerce.com/support.

The first stop for help should be the Toshiba Global Commerce Solutions *Knowledgebase*. It is filled with the latest tips, hints, and FAQs on our product lines. It is indexed and searchable by using keywords such as **OPOS** and **printer** to find all related articles. The Knowledgebase search window is found on the Support Overview page: www.toshibacommerce.com/support.

The next step is to check for updated driver modules. Go to the registration page for the POS drivers you are using and look for the **module update** link. Click it and all changed modules, along with information about where to use them, will be listed.

If you have general pre-sale or usage questions about drivers that are not answered in the publications and you are a Toshiba Business Partner, you can submit questions to the Partnerline team from our Knowledgebase page.

Reporting problems

To report problems, visit the Toshiba Global Commerce Solutions support website:
www.toshibacommerce.com/support

Part 2. Programming reference

Chapter 5. Common settings and configuration options

This chapter contains information on the implementation and support of UnifiedPOS. For each device type, there is a list of the specific devices that are supported and a list of the UnifiedPOS properties, methods, and events that are supported. For some devices, there is additional information specific to the JavaPOS and OPOS implementations, including additional configuration options as well as DirectIO documentation.

The following properties are used only in OPOS:

- BinaryConversion
- OpenResult
- ResultCode
- ResultCodeExtended

Some properties have different names in OPOS. Table 9 lists the UnifiedPOS and OPOS names for these properties.

Table 9. Alternate property names used in OPOS

UnifiedPOS name	OPOS name
DeviceControlDescription	ControlObjectDescription
DeviceControlVersion	ControlObjectVersion
DeviceServiceDescription	ServiceObjectDescription
DeviceServiceVersion	ServiceObjectVersion
PhysicalDeviceDescription	DeviceDescription
PhysicalDeviceName	DeviceName
claim	ClaimDevice
release	ReleaseDevice

JavaPOS

Configuration options

baudRate

```
<prop name="baudRate" type="String" value="9600"/>
```

Determines the baud rate to use for the EIA-232 device. The value to use depends on the device. Some hardware devices cannot have their flow control value changed.

Refer to the hardware manual for the device to determine the correct value.

deviceBus

```
<prop name="deviceBus" type="String" value="RS232"/>
```

The deviceBus property determines how the device is physically attached to the POS terminal.

It can be one of the following values:

- Proprietary

- RS232
- RS485
- HID

flowControl

```
<prop name="flowControl" type="String" value="Xon/Xoff"/>
```

Determines the flow control used for EIA-232 devices. The value to use depends on the device. Some hardware devices cannot have their flow control value changed.

It can be one of the following values:

- Xon/Xoff
- None
- Hardware

impClass

```
<prop name="impClass" type="String" value="com.ibm.jpos.services.sdi.ToneIndicatorServiceImp"/>
```

The impClass specifies the name of the Java class that is responsible for implementing the UnifiedPOS specification for a particular device. It uses the Java class specified in the abstractionClass entry to communicate with the physical hardware.

The abstractionClass and impClass are a matched pair of configuration entries that together determine how an application using JavaPOS communicates with a specific device.

Current XML entries should have an impClass entry that contains one of the following values:

```
com.ibm.jpos.services.sdi.CashDrawerServiceImp
com.ibm.jpos.services.sdi.CashDrawer4610ServiceImp
com.ibm.jpos.services.sdi.CheckScannerServiceImp
com.ibm.jpos.services.sdi.FiscalPrinterServiceImp
com.ibm.jpos.services.sdi.HardTotalsServiceImp
com.ibm.jpos.services.poss.IBMHardTotalsImp
com.ibm.jpos.services.sdi.KeylockServiceImp
com.ibm.jpos.services.sdi.LineDisplayAPAServiceImp
com.ibm.jpos.services.sdi.LineDisplayVFDServiceImp
com.ibm.jpos.services.poss.LineDisplayAnop0Imp
com.ibm.jpos.services.poss.LineDisplayShopperImp
com.ibm.jpos.services.sdi.LineDisplayServiceImp
com.ibm.jpos.services.sdi.MICRServiceImp
com.ibm.jpos.services.poss.IBMMICRImp
com.ibm.jpos.services.sdi.MotionSensorServiceImp
com.ibm.jpos.services.sdi.MSRServiceImp
com.ibm.jpos.services.sdi.POSKeyboardServiceImp
com.ibm.jpos.services.sdi.IBM4610PrinterServiceImp
com.ibm.jpos.services.sdi.IBMSureonePrinterServiceImp
com.ibm.jpos.services.sdi.IBM4689PrinterServiceImp
com.ibm.jpos.services.poss.POSSPrinter4Imp
com.ibm.jpos.services.sdi.ScaleServiceImp
com.ibm.jpos.services.sdi.ScannerServiceImp
com.ibm.jpos.services.sdi.ToneIndicatorServiceImp
```

Note: You should not need to manually change this value.

portName

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

Determines the port to use for EIA-232 devices.

Valid port names are determined by how many ports the POS terminal physically has.

Port names are the same format on Windows and Linux (COMx).

ProprietaryBusSubType

```
<prop name="com.ibm.posj.bus.ProprietaryBusSubType" type="String" value="Embedded"/>
```

For system devices (those having a deviceBus value of Proprietary), this determines whether the device is connected via PS/2 (keylock, tone indicator, MSR, POS Keyboard) or is physically built into the terminal (hard totals, motion sensor, keylock, cash drawer).

It can be one of the following values:

- Embedded
- PosKbd

sioDeviceNumber

```
<prop name="com.ibm.posj.bus.rs485.sioDeviceNumber" type="String" value="0x54"/>
```

A unique number that indicates the type of device.

Since RS-485 devices are detected automatically by the driver, it should not be necessary to manually change this value.

sioPortNumber

```
<prop name="com.ibm.posj.bus.rs485.sioPortNumber" type="String" value="0x11"/>
```

Determines the logical port number for RS-485 peripheral devices.

Valid values are:

- 0x11: RS-485 devices attached to the primary controller
- 0x22: RS-485 devices attached to the secondary controller

Since RS-485 devices are detected automatically by the driver, it should not be necessary to manually change this value.

sioSlotNumber

```
<prop name="com.ibm.posj.bus.rs485.sioSlotNumber" type="String" value="0x01"/>
```

Determines the logical slot number for RS-485 peripheral devices.

- 1: RS-485 devices attached via the ISA or PCI bus
- 5: RS-485 devices attached via the micro-channel bus

Since RS-485 devices are detected automatically by the driver, it should not be necessary to manually change this value.

USB firmware update files for JPOS

All Toshiba USB peripheral devices contain firmware to control the USB bus, in addition to the firmware that controls the device itself.

Depending on the device, the firmware for the USB can be packaged in a number of different ways:

1. Separate from the main device firmware (IE, 4610 Tx3/4/5/6/7)
2. Contains the USB bus firmware embedded in the main device firmware (IE 4610 Tx8/9)
3. Contains the main device firmware embedded with the USB bus firmware (IE, USB 2x20 display)

The JavaPOS driver automatically updates the firmware, if necessary, when the terminal first boots. It will not update the firmware if a device is hot-plugged while the terminal is running.

New firmware files may be downloaded from the Toshiba Global Commerce Solutions support website and placed into the relevant folder on the terminal.

Table 10. JavaPOS Firmware Update

Device	Bus	File Name	Copy files to:
Cash Drawer	USB	aip455*.dat	C:\pos\ibmjpos\res\flash\usb
Modular Keyboards	USB	aip460x3.dat	C:\pos\usb
Modular Keyboards	PS/2	aip460x3.dat	C:\pos\firmware
2x20 Line Displays	USB	aip452*.dat	C:\pos\ibmjpos\res\flash\usb
Printer	RS485	aip*.hex	C:\pos\firmware
Printer	RS232	aip*.hex	C:\pos\ibmjpos\res\flash\rs485
Printer	USB	aip*.hex	C:\pos\ibmjpos\res\flash\rs485

EIA-232 PosPrinter device detection

When a logical device is in use on an EIA-232 attached printer, the `jpos.xml` file must contain an entry for that device (for example, MICR) and an entry for the printer itself.

If the `jpos.xml` file does not contain an entry for the printer, then you will be unable to use the logical device.

OPOS

Accessing multiple devices

One physical device, such as a SureMark printer, can contain multiple logical OPOS devices (for example, a printer, tone indicator, MICR, and cash drawer). If multiple logical devices are accessed in one physical device at the same time, as many POS applications do, then you must access all of the logical devices from the same executable program. Different executable programs cannot be used to control each single logical device.

Configuration options

OPOS configuration settings are stored in the registry per the UnifiedPOS specification.

The settings in Table 11 on page 45 apply to PS/2, RS-485 and USB devices. Used in combination, they uniquely identify a peripheral device.

Because the devices are detected automatically by the driver, it should not be necessary to change any of the values.

The settings in Table 12 on page 45 apply to EIA-232 devices. Devices attached via EIA-232 cannot be automatically detected by the driver and will have to be configured manually.

Table 11. Service Object settings for non EIA-232 devices

Keyword	Type	Description
slotNumber	String	Logical slot (0-8) to which the device is connected. Typical values are: 0 - System devices 1 - RS-485 devices attached via the ISA or PCI bus 5 - RS-485 devices attached via the micro-channel bus 8 - USB devices
portNumber	String	Logical port to which the device is connected. Typical values are: 0 - System devices X'11' - RS-485 devices attached to the primary controller X'22' - RS-485 devices attached to the secondary controller
deviceNumber	String	Unique number that indicates the type of device. For valid values, refer to <i>Toshiba Point of Sale Subsystem Programming Reference and User's Guide</i> .

Table 12. Service Object settings for EIA-232 devices

Keyword	Type	Description
ComPort	String	Serial port that the device is attached to. Valid values are determined by the operating system.
BaudRate	String	Baud rate used for serial port. Valid values are determined by the specific peripheral device. Typical values are 9600, 19200 and 115200.
ControlFlow	String	Control flow for serial device. Valid values are DTR/DSR or XON/XOFF.

USB firmware update files for OPOS

All Toshiba USB peripheral devices contain firmware to control the USB bus, in addition to the firmware that controls the device itself.

Depending on the device, the firmware for the USB can be packaged in a number of different ways:

1. Separate from the main device firmware (IE, 4610 Tx3/4/5/6/7)
2. Contains the USB bus firmware embedded in the main device firmware (IE 4610 Tx8/9)
3. Contains the main device firmware embedded with the USB bus firmware (IE, USB 2x20 display)

If necessary, the OPOS driver will automatically update the firmware when the terminal first boots. It will not update the firmware if a device is hot-plugged while the terminal is running.

New firmware files may be downloaded from the Toshiba website and placed into the \POS\USB folder on the terminal. The status of the USB firmware can be checked by viewing \POS\LOG\AIPFLASH.LOG

Table 13. OPOS Firmware Update

Device	Bus	File Name	Copy files to:
Cash Drawer	USB	aip455*.dat	C:\pos\usb
Modular Keyboards	USB	aip460x3.dat	C:\pos\usb
Modular Keyboards	PS/2	aip460x3.dat	C:\pos\firmware
2x20 Line Displays	USB	aip452*.dat	C:\pos\usb
Printer	RS485	aip*.hex	C:\pos\firmware

Table 13. OPOS Firmware Update (continued)

Device	Bus	File Name	Copy files to:
Printer	RS232	aip*.hex	C:\pos\firmware
Printer	USB	aip*.hex	C:\pos\firmware

Modifying USB, RS-485, and PS/2 device behavior

For USB, RS-485, and PS/2 devices, additional device configuration can be done through the resource file: aipsys.res. Use of this file is documented in *Point of Sale Subsystem: Programming Reference and User's Guide (SC30-3560)*. For general information about using aipsys.res, see Chapter 3, *Customizing the Toshiba Point Of Sale Subsystem*. For device-specific resources that can be specified in aipsys.res, see Chapter 21, *Resource Sets*.

Note:

1. Resource names and values are case sensitive.
2. Resource names appear in the documentation as PosNxxxXxxx. However, when specifying resources names in the aipsys.res file, the PosN prefix must be removed.
3. Defined macros for resource values are documented as PosXXXX_XXXX. However, when using these macros in the aipsys.res file, the Pos prefix must be removed.
4. Changes to the aipsys.res file will not take effect until the system is rebooted.
5. In Windows, aipsys.res is located in the default install directory C:\POS.
6. On Linux systems, aipsys.res is located in the /etc directory.

Ignoring USB devices

In some cases, it might be necessary to configure OPOS to ignore certain USB devices. This can be accomplished by creating a String value, specified below:

Location: HKLM\SOFTWARE\Toshiba\Point of Sale Subsystem

Table 14. Ignoring USB devices

Keyword	Type	Description
DisableNonTbmUSB	String	This property provides a mechanism for disabling all non-Toshiba USB devices from Toshiba OPOS drivers. True: All OEM USB devices, such as Symbol scanner/scale devices are ignored. False (default): The USB OEM devices are supported through Toshiba OPOS drivers.

Table 14. Ignoring USB devices (continued)

Keyword	Type	Description
DisableUSB	String	<p>This property provides a mechanism for disabling USB devices from Toshiba OPOS drivers.</p> <p>True: Toshiba OPOS drivers will NOT claim all USB devices supported by Toshiba OPOS.</p> <p>False (default): Toshiba OPOS drivers will claim all USB devices supported by Toshiba OPOS.</p> <p>To selectively disable devices from being claimed by Toshiba OPOS drivers, you can specify a list of VendorID:ProductID pairs separated by ' ' (space).</p> <p>For example: DisableUSB="04b3:4604 04b3:4535 05e0:0200" will disable the Toshiba NANPOS USB Keyboard, the Toshiba 4610 Printer, and a Symbol Scanner.</p>

Network device connectivity

The Ethernet connection for network devices maybe disrupted or broken due to network issues or power interruption. While the driver attempts to reconnect to network devices when the connection is recovered, successful reconnection is not guaranteed. If the device is claimed by another system during reconnection, the device will be released automatically at the current system. Network devices have a keep-alive time of one minute before breaking the connection with the system.

By default, the OPOS driver does not monitor the connectivity of network devices. To enable the offline status monitoring of network devices, the following registry needs to be set:

Table 15. Ethernet printer offline monitoring timer

Keyword	Type	Description
EthernetPrinterWatchdogTimeout	DWORD	<p>This property sets the offline status monitoring timer (milliseconds) to verify network printer is still available.</p> <p>Values:</p> <p>0 (Default) Disables the monitoring of the printer.</p> <p>non-zero Milliseconds to sleep between checks and the milliseconds to check for a response from the printer.</p> <p>Note: Wait time for notification is potentially up to twice the indicated value. If no property is found, the default value is used. If the property is present but it has an invalid value, then the default value is used. It is recommended to set the value to something larger than 60,000 milliseconds to minimize the network traffic.</p>

USB system attached keyboard

The Toshiba alphanumeric keyboard requires a device-specific kernel mode driver (`aipkbpos.sys`). On certain systems, such as the Toshiba 4810-32x/33x with Microsoft Windows Embedded Point of Service OS, a second reboot is required to properly install the kernel mode driver.

Example of a typical error: "The file 'aipkbpos.sys' on Toshiba POS HID Keyboard Install Disk is needed."

Common Extended Result Codes

In certain situations the OPOS driver will return extended result codes that originate from the underlying Point of Sale Subsystem driver, which controls access to USB and RS-485 devices.

Resolution of these errors is out of the control of the application programmer, so they are listed here for informational purposes only. Consult your Toshiba support representative for more information if you receive one of the errors listed below.

Table 16. Errors

Error Number	Description
301	The Toshiba Point of Sale Subsystem received an operating system error.
302	The application has not performed a successful PosInitialize() subroutine call. An application must successfully issue the PosInitialize() subroutine call before issuing any other Toshiba Point of Sale Subsystem subroutine calls.
303	The device descriptor parameter is not a valid device descriptor. Only device descriptors returned by PosOpen() can be used.
304	The application has already successfully processed the PosInitialize() subroutine.
305	The Toshiba Point of Sale Subsystem has not been able to acquire the memory it requires to operate.
306	The Toshiba Point of Sale Subsystem has detected an error with one of the devices, or other hardware attached to the system.
307	A specific device could not be determined from the information given to the PosOpen() subroutine.
308	The value for the PosNqueueHandle resource is not valid.
309	The Toshiba Point of Sale Subsystem limit for device descriptors for each process has already been reached. The Toshiba Point of Sale Subsystem cannot start any more devices for this process until one or more devices are closed.
311	The application has requested a function that is not supported for the device descriptor. See the Related Information section of the individual device chapters for the valid subroutine calls for the device.
312	This error can occur when issuing a PosRead() subroutine call. The application specified a length value that was too small for the data being read. When this error occurs, data is not put into the application's buffer.
313	The application issued a POS_SYS_ACQUIRE_DEVICE PosIOCtl() request in order to acquire device connection, but another device descriptor has already acquired the device connection. The acquired device connection must be released before this POS_SYS_ACQUIRE_DEVICE PosIOCtl() request can be successful.
314	The application is trying to acquire a device connection that it has already acquired.
315	The application issued a subroutine call that requires an acquired device connection. The application must first successfully issue the POS_SYS_ACQUIRE_DEVICE PosIOCtl() request.
316	The request parameter of the PosIOCtl() subroutine is not valid for the device associated with the device descriptor.
317	The device is not currently connected to the system unit. The device may have gone offline since the application initialized the device.

Table 16. Errors (continued)

Error Number	Description
318	The application specified an incorrect value for nbytes when it issued a PosWrite() subroutine call.
319	The values of the class parameter and the PosNdeviceNumber resource given to the PosOpen() subroutine were both valid individually, but not valid in combination with each other.
320	A device was closed when data was available. The data is discarded. The data is not logged.
321	The Toshiba Point of Sale Subsystem has discovered an internal logic error.
325	The parameter is a negative number. It must be 0 (zero), or a positive number.
326	An incorrect value was specified for the PosNslotNumber resource.
327	The PosNslotNumber resource specified a slot containing an unsupported adapter.
328	An incorrect value was specified for the PosNportNumber resource.
329	The timeout specified by the PosNreadTimeout resource has expired while waiting for input from the Toshiba Point of Sale Subsystem input queue. Data did not become available for reading from the input queue before the timeout expired.
330	The length of the name parameter was 0 (zero) or was too long.
331	The class parameter given to the PosOpen() subroutine is not valid.
332	A system call was interrupted, or could not be completed for one of the following reasons: 1. The application called the PosRead() subroutine for the Toshiba Point of Sale Subsystem input queue (device descriptor zero) and was waiting for input to become available (the value of the PosNreadTimeout resource was non-zero). While the thread was waiting, an exception occurred, such as the user pressing Ctrl-Break. 2. The application called the PosRead() subroutine for the Toshiba Point of Sale Subsystem input queue (device descriptor zero) and was waiting for input to become available (the value of the PosNreadTimeout resource was non-zero). While the thread was waiting, another thread in the same process called the PosRead(), or PosIOCtl() subroutines for the input queue. In this case, the input queue was already busy and could not process the request.
334	The Toshiba Point of Sale Subsystem detected an address that is not valid. This error is set if the buffer, the control block, or the string pointed to is not valid, or if the process does not have access to the entire length of the control block, or buffer. The entire length of structures of known sizes are validated before being used. If the structure is an output parameter, then the structure is verified to be writeable. If the structure is of an unknown size, only the first byte of the structure is validated.
335	The application issued a PosRead() subroutine call when the target device is in a locked state and there is no data available to be returned.
336	The file parameter given to the PosInitialize() subroutine is not valid. The problem might be that the length of the file parameter is 0 (zero), or is too long.
337	The Toshiba Point of Sale Subsystem for Windows is not running.

Chapter 6. Cash drawer

Cash drawer supported devices

Table 17. Cash drawer supported devices

Device	Connectivity
1. Toshiba Cash Drawer	USB
2. SurePOS 500/600 family built-in cash drawer	EIA-232
3. SurePOS 700 family built-in cash drawer	Integrated
4. 4694 family built-in cash drawer	RS-485
5. 4610 printer built-in cash drawer	EIA-232, RS-485, USB, Network
6. SurePOS 300 family	Integrated
7. SureOne family built-in cash drawer (OPOS)	Integrated
8. TCx 700	USB

Supported properties and methods

Table 18. Cash drawer common properties

Property	JavaPOS	OPOS
AutoDisable		Not supported
BinaryConversion	Not supported	
CapCompareFirmwareVersion		False
CapPowerReporting	PR_STANDARD	All support STANDARD except SurePOS 500/600 family built-in cash drawer, which supports ADVANCED
CapStatisticsReporting	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CapUpdateFirmware		False
CapUpdateStatistics	False (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CheckHealthText	All	Not supported
Claimed		All
DataCount		Not supported
DataEventEnabled		Not supported
DeviceControlDescription		All
DeviceControlVersion		All
DeviceServiceDescription		All
DeviceServiceVersion		All
FreezeEvents		All
OpenResult	Not supported	All
OutputID		Not supported

Table 18. Cash drawer common properties (continued)

Property	JavaPOS	OPOS
PhysicalDeviceDescription		All
PhysicalDeviceName		All
PowerNotify		All
PowerState		All
State		All

Table 19. Cash drawer specific properties

Property	JavaPOS	OPOS
CapStatus		All
CapStatusMultiDrawerDetect		All except 4610 printer built-in cash drawer
DrawerOpened		All

Table 20. Cash drawer common methods

Method	JavaPOS	OPOS
compareFirmwareVersion		Not supported
checkHealth	All	All except SurePOS 500/600 family built-in cash drawer
claim		All
clearInput		Not supported
clearOutput	All	Not supported
close		All
directIO		Not supported
open		All
release		All
resetStatistics		Not supported
retrieveStatistics	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	Not supported
updateFirmware		Not supported
updateStatistics		Not supported

Table 21. Cash drawer specific methods

Method	JavaPOS	OPOS
openDrawer		All
waitForDrawerClose		All

Table 22. Cash drawer events

Event	JavaPOS	OPOS
DirectIOEvent		Not supported
IBM_JPOS_SUE_CASH_OPEN_TIME_OUT	1,8	Not supported
StatusUpdateEvent		All

Supported properties and methods (cash drawer attached fiscal printer: Italy)

Table 23. Common properties (cash drawer attached fiscal printer: Italy)

Property	JavaPOS	OPOS
AutoDisable		Not supported
CapCompareFirmwareVersion		False
CapPowerReporting		PR_NONE
CapStatisticsReporting		False
CapUpdateFirmware		False
CapUpdateStatistics		False
CheckHealthText		String
Claimed		True/False
DataCount		Not supported
DatEventEnabled		Not supported
DeviceEnabled		True/False
DeviceControlDescription		String
DeviceControlVersion		int
DeviceServiceDescription		String
DeviceServiceVersion		Supported
FreezeEvents		False
OutputID		E_ILLEGAL
PhysicalDeviceDescription		String
PhysicalDeviceName		String
PowerNotify		E_ILLEGAL
PowerState		PS_UNKNOWN
State		int

Table 24. Specific properties (cash drawer attached fiscal printer: Italy)

Property	JavaPOS	OPOS
CapStatus		False
CapStatusMultiDrawerDetect		False
DrawerOpened		False

Table 25. Common methods (cash drawer attached fiscal printer: Italy)

Method	JavaPOS	OPOS
checkHealth		Supported
claim		Supported
clearInput		Not supported
clearOutput		E_ILLEGAL
close		Supported

Table 25. Common methods (cash drawer attached fiscal printer: Italy) (continued)

Method	JavaPOS	OPOS
compareFirmwareVersion		E_ILLEGAL
directIO		Supported
open		Supported
release		Supported
resetStatistics		E_ILLEGAL
retrieveStatistics		E_ILLEGAL
updateFirmware		E_ILLEGAL
updateStatistics		E_ILLEGAL

Table 26. Cash drawer attached fiscal printer (Italy) specific methods

Method	JavaPOS	OPOS
openDrawer		Supported
waitForDrawerClose		E_ILLEGAL

JavaPOS configuration

CashDrawer.OpenDrawerRetries

```
<prop name="com.ibm.jpos.sdi.config.CashDrawer.OpenDrawerRetries" type="String" value="0"/>
```

This property is used to configure the number of retries to attempt before a fail with the openDrawer method.

Default value is 0.

Note:

1. If no property is found in the entry, the default value is used.
2. If the property is present but it has an invalid or negative value, then the default value is used.
3. The value is read only at open time.

CashDrawer.OpenTimeOut

This property is used to configure the Open timeout event of the cash drawer. When the cash drawer had been opened but was not closed during the timeout value, the driver will fire a com.ibm.jpos.DirectIO.IBM_JPOS_SUE_CASH_OPEN_TIME_OUT Status Update Event.

The value represents the interval of time (in seconds); the range of the property is from 0 to 255. If the value is 0, the timeout is disabled. If the value is out of range, it is set to the default (255).

- Values accepted: 0 to 255 (Default 255)
- Device Support: 1

Note:

1. If no property is found in the entry, the default value is used.
2. If the property is present but contains an invalid value, then the default value is used.
3. The value is read only at open time.

signalsReversed

```
<Prop name="signalsReversed" type="Boolean" value="false"/>
```

Reverses the wiring polarity of the cash drawer when the cash drawer is attached to a 4610 printer. Some OEM drawers are wired in reverse to the standard drawer, so will require this value to be changed to TRUE.

- Valid values are TRUE and FALSE.
- Default is FALSE.

deviceNumber

```
<prop name="com.ibm.posj.bus.deviceNumber" type="String" value="0"/>
```

Used to properly identify the cash drawers present. The values 0-3 are reserved for the cash drawer attached to the system unit. For example, 0 means the cash drawer is attached to port 3A on the system, and 1 means the cash drawer is attached to port 3B on the system. Cash drawers attached to RS-485 and USB 4610 printers have a deviceNumber starting with 4.

Pulse Width

This property is used to configure the Pulse Width during openDrawer call.

For System built-in Cash Drawers:

- Refer to the posj.properties file and property:
`com.ibm.posj.CashDrawer.PulseWidth`
- Values accepted:
 - 0-100 ms Cash Drawer Pulse (Default)
 - 1-200 ms Cash Drawer Pulse

For 4610 printer built-in cash drawer:

- Refer to the posj.properties file and properties:
`com.ibm.posj.CashDrawer.4610.PulseWidthOn`
- Values accepted:
 - From 0 to 255 - Cash Drawer Pulse x 2 milliseconds
 - (Default ON value = 100)

Additional JavaPOS information

4610 printer-attached cash drawer

JavaPOS supports regular Toshiba cash drawers and OEM cash drawers attached to the Toshiba 4610 Printer via EIA-232, RS-485, and USB.

For a Toshiba cash drawer, set the signalsReversed property to *false*. For example:

```
<Prop name="signalsReversed" type="Boolean" value="false"/>
```

For some OEM cash drawers, the signalsReversed property will need to be set to *true*. For example:

```
<Prop name="signalsReversed" type="Boolean" value="true"/>
```

Note: In the `jpos.xml.<os>Sample`, the signalsReversed property is in the JposEntries with the following logicalName values:

- "CashDrawer RS232 4610-A"
- "CashDrawer RS232 4610-B"

- "CashDrawer RS485 4610-A"
- "CashDrawer RS485 4610-B"
- "CashDarwer USB 4610-A"
- "CashDrawer USB 4610-B"

Loading the cash drawer driver for Linux

To load the correct cash drawer device driver in Linux, the /opt/ibm/javapos/etc/machine.conf file must be modified. The format of machine.conf is:

```
<keyword> <number of entries>
<machine type><model number>
```

"<keyword>" represents the device for which a driver is needed and "<number of entries>" indicates the number of "<machine type><model number>" pairs that follow. If the system on which JavaPOS is running matches one of the specified "<machine type><model number>" pairs, the device driver for that "<machine type><model number>" is loaded.

For the cash drawer, machine.conf might have the following entries:

```
CD 1
4810321
```

JavaPOS DirectIO calls

The following DirectIO commands are supported by the Cash Drawer device. The syntax is as follows:

```
Syntax directIO ( command: int32, inout data: int32, inout obj: object ):
    void { raises-exception, use after open-claim-enable }
```

To access DirectIO constants: import com.ibm.jpos.services.DirectIO.

Cash Drawer get serial number command

Get the serial number of the Cash Drawer device.

Supported devices:

- USB Cash Drawer
- AnyPlace Hub Cash Drawer
- 4610 printer built-in Cash Drawer

Table 27. DEVICE_INFORMATION_CMD information

Parameter	Type	Value
Command	int32	DirectIO.DEVICE_INFORMATION_CMD
Data	int32	Any value. A null value is accepted.
Obj	String	com.ibm.jpos.services.DeviceInformation.java

Remarks

For the DeviceInformation object:

getSerialNumber() returns the serial number in a String object. An empty string is returned for unsupported devices.

OPOS configuration

Table 28. Service Object settings for cash drawer

Keyword	Type	Description
CashDrawerNumber	String	Cash drawer number. Valid values are 1 (default) and 2.
PulseWidthOnTime	String	Pulse on time (0-512) for firing the cash drawer. The default value is 100. Note: This setting cannot be modified with the Configuration tool.
PulseWidthOffTime	String	Pulse off time (0-512) for firing the cash drawer. The default value is 100. Note: This setting cannot be modified with the Configuration tool.
SignalsReversed	REG_SZ	Reverses the wiring polarity of the cash drawer. Valid values: True Reverse polarity (other value) Standard polarity (default)
CDFireRetries	REG_DWORD	Determines the number of CD fire retries. Valid values: 0 No CD fire retries (default) 1 - 3 # of retries for CD fire
DisableCDStatus	REG_SZ	Set this registry to disable the cash drawer status checking after OpenDrawer and StatusUpdateEvent firing. Valid values: True Disable Other values (default) Enable
CheckOpenStatus	REG_SZ	Valid values: True Check the cash drawer open status and only fire the drawer open command when it's closed. Other values (default) Fire the drawer open command regardless of the cash drawer open status.

Additional OPOS information

OEM cash drawer support

Some OEM cash drawers are wired the opposite of Toshiba cash drawers. Use the Signals Reversed check box in the configuration utility to indicate that your cash drawer is one of these OEM cash drawers.

OPOS support for the USB SureMark printer assumes the Toshiba cash drawer wiring. The OPOS support for EIA-232 SureMark printers prior to release 1.7.1 assumes the OEM wiring. Starting with release 1.7.1, OPOS support for the EIA-232 SureMark Printer cash drawer has been changed to match the behavior of the USB Printer cash drawer. Therefore, it may be necessary to reconfigure EIA-232 SureMark printer cash drawers by modifying the Signals Reversed check box.

USB cash drawer support

When one or more USB cash drawers are attached, they are enumerated in the OPOS Configuration tool in the following way:

Table 29. USB cash drawer support

Cash Drawer on Port	Enumeration in OPOS Config tool
3A on AnyPlace POS Hub	slot = 0x08, port = 0x11, device = 0x54
1st Powered USB Port	slot = 0x08, port = 0x22, device = 0x54
2nd Powered USB Port	slot = 0x08, port = 0x33, device = 0x54

For repeatable port assignments, keep the cash drawer on the same powered USB port(s) during hardware configuration. This becomes crucial when attaching more than one cash drawer.

Hot-plug support:

Officially, hot-plugging is not recommended and is not supported for Toshiba Retail IOs. Currently, there is no means of associating a specific device to a specific slot, port, or device definition.

The supported means of changing USB cash drawers is to power down the machine and perform the device swapping while the machine is powered down.

OPOS DirectIO calls

The directIO commands in this section are supported for the cash drawer device.

The following is the correct syntax: **LONG DirectIO (LONGCommand, LONG*pData, BSTR*pString);**

To access constants for a cash drawer attached to a POS Printer, include IBMPOSPtr.h. For a cash drawer attached to a Fiscal Printer, include IBMFPT.H.

Note: For all operations, use BinaryConversion rather than OPOS_BC_NONE so that data is not prematurely truncated.

Cash Drawer get serial number command

Get the serial number of the Cash Drawer device or the main device where the cash drawer is attached to.

Supported devices:

- USB Cash Drawer
- AnyPlace Hub Cash Drawer
- 4610 printer built-in Cash Drawer
- Fiscal printer built-in Cash Drawer

Constant definition: #define DIRECTIO_DEVICE_INFORMATION_CMD 0x30

Table 30. Parameters for DIRECTIO_DEVICE_INFORMATION_CMD subcommand

Parameter	Type	Value
Command	LONG	DIRECTIO_DEVICE_INFORMATION_CMD
pData	LONG*	Not used
pString	BSTR*	A BSTR to be filled with data.

Remarks

pString returns the serial number in a BSTR object. An empty string is returned for unsupported devices.

Chapter 7. Check scanner

Check scanner supported devices

Table 31. Check scanner supported devices

Device	Connectivity
1. 4610 TI8 CheckScanner	EIA-232, RS-485, USB
2. 4610 TI9 CheckScanner	EIA-232, RS-485, USB

Supported properties and methods

Table 32. Check scanner common properties

Property	JavaPOS	OPOS
AutoDisable	All	Supported
CapCompareFirmwareVersion		False
CapPowerReporting	PR_STANDARD	Supported
CapStatisticsReporting	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CapUpdateFirmware		False
CapUpdateStatistics	False (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CheckHealthText	All	Supported
Claimed	All	Supported
CompareFirmwareVersion		Not supported
ItemCount	All	Supported
DataEventEnabled	All	Supported
DeviceControlDescription	All	Supported
DeviceControlVersion	All	Supported
DeviceEnabled	All	Supported
DeviceServiceDescription	All	Supported
DeviceServiceVersion	All	Supported
FreezeEvents	All	Supported
OutputID	All	Supported
PowerNotify	All	Supported
PowerState	All	Supported
PhysicalDeviceDescription	All	Supported
PhysicalDeviceName	All	Supported
State	All	Supported

Table 33. Check scanner specific properties

Property	JavaPOS	OPOS
CapAutoContrast	False	
CapContrast	True	
CapAutoGenerateFileID	False	Not supported
CapAutoGenerateImageData	False	Not supported
CapAutoSize	True	Supported
CapColor	CheckScannerConst.CHK_CCL_GRAYSCALE	Supported
CapConcurrentMICR	True	Supported
CapDefineCropArea	True	Supported
CapImageFormat	<ul style="list-style-type: none"> • CheckScannerConst.CHK_CIF_TIFF • CheckScannerConst.CHK_CIF_BMP • CheckScannerConst.CHK_CIF_JPEG • CheckScannerConst.CHK_CIP_NATIVE 	Supported: <ul style="list-style-type: none"> • CHK_CIF_TIFF • CHK_CIF_BMP • CHK_CIF_JPEG • CHK_CIP_NATIVE
CapImageTagFormat	True	Supported
CapMICRDevice	True	Supported
CapStoreImageFiles	True	Supported
CapValidationDevice	False	Not supported
Color	CheckScannerConst.CHK_CL_GRAYSCALE	CHK_CCL_GRAYSCALE only
ConcurrentMICR	True	Supported
Contrast	0 - 100, default = 50	50
CropAreaCount	0	Supported
DocumentHeight	Set by the driver after document scanned Initialized value: 8000	Supported
DocumentWidth	Set by the driver after document scanned Initialized value: 4000	Supported
FileID	""	Not supported
FileIndex	0	Supported
ImageData	NULL	Supported
ImageFormat	Supported: CheckScannerConst.CHK_CIF_NATIVE CheckScannerConst.CHK_CIF_TIFF CheckScannerConst.CHK_CIF_BMP CheckScannerConst.CHK_CIF_JPEG	Supported: <ul style="list-style-type: none"> • CHK_IF_TIFF • CHK_IF_JPEG • CHK_IF_BMP • CHK_IF_NATIVE
ImageMemoryStatus	Current state of checkscanner memory	Supported
ImageTagData	Max 32 characters allowed	Supported
MapMode	Only CheckScannerConst.CHK_MM_ENGLISH supported	Supported: CHK_MM_ENGLISH only
MaxCropAreas	20	Supported
Quality	Only 200 dpi supported	TI8: 200, TI9: 200 (default) or 100

Table 33. Check scanner specific properties (continued)

Property	JavaPOS			OPOS	
QualityList	1 - set to "100" 2 - set to "100, 200"			TI8: "200", TI9: "100,200"	
RemainingImagesEstimate (Estimates when memory is empty.)	Format:	TI8/TI9	TI9	TI8/TI9	TI/9
	JposConst.JPOS_CH_INTERNAL	200 DPI	100 DPI	200 DPI	100 DPI
	JposConst.JPOS_CH_EXTERNAL	108	216	108	216
	JposConst.JPOS_CH_INTERACTIVE	1	2	1	2
	JposConst.JPOS_CH_JPEG	20	40	20	40
CheckScannerConst.CHK_CIF_NATIVE	JposConst.JPOS_CH_NATIVE	1	2	1	2

Table 34. Check scanner common methods

Method	JavaPOS	OPOS
checkHealth	Supported: JposConst.JPOS_CH_INTERNAL JposConst.JPOS_CH_EXTERNAL Throws jposException(JPOS_E_ILEGAL) JposConst.JPOS_CH_INTERACTIVE	All Supported: OPOS_CH_INTERNAL OPOS_CH_EXTERNAL OPOS_CH_INTERACTIVE
claim	All	Supported
clearInput	All	Supported
clearInputProperties	All	Supported
clearOutput	All	Supported
close	All	Supported
directIO	All	Supported
open	All	Supported
release	All	Supported
resetStatistics	Not supported	
retrieveStatistics	True (see Appendix A, “JavaPOS support for UnifiedPOS device statistics properties,” on page 431)	Not supported
updateFirmware	Not supported	
updateStatistics	Not supported	

Table 35. Check scanner specific methods

Method	JavaPOS	OPOS
beginInsertion	All	Supported
beginRemoval	All	Supported
clearImage	All (Only CheckScannerConst.CHK_CLR_ALL supported.)	Supported: CHK_CLR_ALL only
defineCropArea	All	Supported

Table 35. Check scanner specific methods (continued)

Method	JavaPOS	OPOS
endInsertion	All	Supported
endRemoval	All	Supported
retrieveImage	Supported: CheckScannerConst.CHK_CROP_ AREA_ENTIRE_IMAGE. All other values are supported starting from EC level 0x68	Supported: CHK_CROP _AREA_ENTIRE _IMAGE All other values are supported starting from EC level 0x68
retrieveMemory	Supported: CheckScannerConst.CHK_LOCATE_ BY_FILEINDEX. All other values throw JposException(JPOS_E_ILLEGAL)	Supported: CHK_LOCATE_BY_FILEINDEX CHK_LOCATE_BY_IMAGETAGDATA. Not supported: CHK_LOCATE_BY_FILEID returns OPOS_E_ILLEGAL.
storeImage		All

Table 36. Check scanner events

Event	JavaPOS	OPOS
DataEvent	All	Supported
DirectIOEvent	All	Supported
ErrorEvent	All	Supported
StatusUpdateEvent	All	Supported

JavaPOS DirectIO calls

The DirectIO commands in this section are supported for the check scanner device. The syntax is as follows:

```
directIO ( command: int32, inout data: int32 , inout obj: object):
void { raises-exception }
```

To access DirectIO constants, import com.ibm.jpos.services.DirectIO.

Scanner calibration command

Calibrates the check scanner in Toshiba 4610 Printer Models TI8 and TI9.

Table 37. CHK_DIO_SCANNER_CALIBRATION_CMD information

Parameter	Type	Value
Command	Int32	DirectIO.CHK_DIO_SCANNER_CALIBRATION_CMD
Data	Int32	Any value. A null value is accepted.
Obj	Object	Any value. A null value is accepted.

Note: A white document must be inserted in the DI station. The printer scans the document two or three times to update the printer's calibration data, then DirectIO is paired with the beginInsertion method for controlling check insertion.

Print scanned image command

Prints the specified scanned image to the thermal station.

Table 38. *CHK_DIO_PRINT_SCANNED_IMAGE_CMD* information

Parameter	Type	Value
Command	Int32	DirectIO.CHK_DIO_PRINT_SCANNED_IMAGE_CMD
Data[0]	Int32	DirectIO.CHK_DIO_MEMORY_IMG prints the most recently scanned image. DirectIO.CHK_LOCATE_BY_FILEINDEX prints the image file using the FileIndex property.
Data[1]	Int32	The numeric identifier for the defined crop area. If the values is CHK_CROP_AREA_ENTIRE_IMAGE then the entire area of the scanned image is printed.
Data[2]	Int32	Percentage to scale image in x direction. (A value of 100 or 0 results in no scaling.)
Data[3]	Int32	Percentage to scale image in y direction. (A value of 100 or 0 results in no scaling.)
Data[4]	Int32	DirectIO.CHK_DIO_ROTATE_90 rotates the image counter clockwise 90%. DirectIO.CHK_DIO_NO_ROTATE does not rotate the image.
Obj	Object	Any value. A null value is accepted.

Note:

Printing stored images causes the printer to pause momentarily while the printer formats the data to be printed. This time varies depending on the amount of formatting required.

If cx and cy defined at the crop area extend the printer area further than the boundaries of the image, the values are truncated to the image boundary. If the size of the print area is greater than the thermal print head is capable of printing (either from image size or scaling), the image is truncated to the width of the print head. Currently, the only scaling options are 100% (no scaling) and 200% (double).

Printing resident images in the work area memory with CheckScannerConst.CHK_IF_JPEG format is not supported.

Set check side command

Set the side of the check to scan. This command assumes that the check is inserted face down.

Table 39. *CHK_DIO_SET_CHECK_SIDE_CMD* information

Parameter	Type	Value
Command	Int32	DirectIO.CHK_DIO_SET_CHECK_SIDE_CMD
Data[0]	Int32	<ul style="list-style-type: none">• DirectIO.CHK_DIO_SIDE1 selects the front of the check.• DirectIO.CHK_DIO_SIDE2 selects the back of the check.• DirectIO.CHK_DIO_OPPOSITE selects the opposite face of the check from the current side selected.
Obj	Object	Any value. A null value is accepted.

Note: This allows a check scan to occur on both sides of the document. If a document is not inserted, an error is returned. If data[0] is different from the value returned by

DirectIO.CHK_DIO_GET_CHECK_SIDE_CMD or data[0] is DirectIO.CHK_DIO_SIDE_OPPOSITE, the side of the document is changed and the document is fed. If data[0] is the same as the value returned by DirectIO.CHK_DIO_GET_CHECK_SIDE_CMD, nothing occurs and the method returns.

Get check side command

Return the current setting for the side of the check to be scanned. This command assumes the check is inserted face down. After calling this DirectIO, data[0] is populated with one of the values in the following table:

Table 40. *CHK_DIO_GET_CHECK_SIDE_CMD* information

Parameter	Type	Value
Command	Int32	DirectIO.CHK_DIO_GET_CHECK_SIDE_CMD
Data	Int32	<ul style="list-style-type: none">• DirectIO.CHK_DIO_SIDE_UNKNOWN (Indicates no check is inserted.)• DirectIO.CHK_DIO_SIDE1 (Indicates the front of the check is selected.)• DirectIO.CHK_DIO_SIDE2 (Indicates the back of the check is selected.)
Obj	Object	Any value. A null value is accepted.

Note: This property value can be changed using the DirectIO.CHK_DIO_SET_CHECK_SIDE_CMD method. When a check is inserted, the value returned is DirectIO.CHK_DIO_SIDE1.

Scanner image quality command

This command checks the quality of the image scanned on Model TI9 printers. The check scanner automatically checks the image quality of each check scanned against a set of internal quality attributes. After calling this DirectIO, data[0] is populated with a value of 0 or 1.

- 0 indicates image quality is acceptable.
- 1 indicates image quality may not be acceptable.

Table 41. *CHK_DIO_SCANNER_IMAGE_QUALITY_CMD* information

Parameter	Type	Value
Command	Int32	DirectIO.CHK_DIO_SCANNER_IMAGE_QUALITY_CMD
Data	Int32	An array with at least one element.
Obj	Object	Any value. A null value is accepted.

Note: This DirectIO is supported only for the 4610 TI9 model.

Additional OPOS information

Multiple devices

A POS machine cannot have both RS-485 and USB check scanners operating at the same time.

Chapter 8. Fiscal printer

Fiscal printer operations are supported through directIO() calls. The directIO() functions do not perform any verification of commands sent to the fiscal printer or of data returned from the printer. The functions provide a simple passthrough operation, allowing an application to access the fiscal functions on the printer.

Starting with version 1.13.1, the Toshiba JavaPOS products include full UnifiedPOS service drivers for Greece and Italy Fiscal Printers. Refer to the extended documentation of each country for details:

- Toshiba JavaPOS FiscalPrinter Greece
- Toshiba JavaPOS FiscalPrinter LineDisplay CashDrawer Italy

File locations:

Windows: <install dir>\doc

Linux: /opt/ibm/javapos/config

Note: For Linux, it may be necessary to disable **Auto-configuration only** and enable **XML at JCL** configuration.

Fiscal printer supported devices

USB fiscal printers are not supported on Linux.

Table 42. Fiscal printer supported devices

Toshiba SureMark Printer Fiscal Printer Device	Connectivity	Comments
1. Models Kx3	EIA-232	KD3: Hungary, Italy, Czech Republic, Romania
2. Models Kx4	EIA-232	KJ4: Brazil KN4: Brazil KC4: Chile
3. Models Kx5	EIA-232	KC5: Chile KD5: Hungary, Italy, Czech Republic, Romania
4. Models Kx3	RS-485	KR3: Hungary, Italy, Czech Republic, Greece, Turkey
5. Models Kx5	RS-485	KR5: Hungary, Italy, Czech republic, Greece, Turkey, Chile
6. Models Kx5	USB (no Linux support)	KB5: Turkey KH5: Greece
7. Models Gx3	EIA-232	
8. Models Gx3	RS-485	GR3: Argentina, Greece, Italy, Turkey
9. Models Gx3	USB	GB3: Greece, Turkey GE3: Argentina
10. Models Gx4	EIA-232	GD4: Venezuela
11. Models Gx5	EIA-232	GD5: Turkey
12. Models Gx5	RS-485	GF5: Greece, Italy, Turkey
13. Models Gx5	USB	GB5: Greece, Turkey
14. Models Sx6	EIA-232	SJ6: Brazil
15. Models Sx6	USB	SJ6: Brazil
16. Models 3xA	RS-485	3FA: Argentina

Table 42. Fiscal printer supported devices (continued)

Toshiba SureMark Printer Fiscal Printer Device	Connectivity	Comments
17. Models Kx4	RS-485	KS4: Chile
18. Models Kx3	USB	KB3: Turkey
19. Gx4	RS-485	GR4: Venezuela
20. 3Bx	RS-485	3BS: Brazil 3BM: Brazil
21. 3F	RS-485	Brazil, Bulgaria, Greece, Hungary, Italy, Latvia, Mexico, Romania, Russia, Slovakia, Turkey, Uruguay, Venezuela
22. 1NF	EIA-232 / USB	Chile, Greece, Hungary, Italy, Turkey, Venezuela
23. Models Kx4	USB	KJ4: Brazil
24. 2xF	EIA-232 / USB	2CF: Dominican Republic, Panamá, Venezuela 2NF: Greece, Italy, Turkey
Country-independent models (GF3/GF5, GG3/GG5, KF3/KF5, KG3/KG5) are not listed.		

Supported properties and methods

Table 43. Fiscal printer common properties

Property	JavaPOS	OPOS
AutoDisable		Not supported
CapCompareFirmwareVersion		False
CapPowerReporting	4-6, 8, 9, 12-13, 15-21, 22/USB, 23, 24/USB: PR_STANDARD	4-6, 8-9, 12-13, 15 - 21 : PR_STANDARD
CapStatisticsReporting	True	False
CapUpdateFirmware		False
CapUpdateStatistics	False	False
CheckHealthText	All	4-6, 8-9, 12-13, 15 - 21
Claimed	All	4-6, 8-9, 12-13, 15 - 21
DataCount		Not supported
DataEventEnabled		Not supported
DeviceControlDescription	All	4-6, 8-9, 12-13, 15 - 21
DeviceControlVersion	All	4-6, 8-9, 12-13, 15 - 21
DeviceServiceDescription	All	4-6, 8-9, 12-13, 15 - 21
DeviceServiceVersion	All	4-6, 8-9, 12-13, 15 - 21
FreezeEvents	All	4-6, 8-9, 12-13, 15 - 21
OutputID	All	4-6, 8-9, 12-13, 15 - 21
PhysicalDeviceDescription	All	4-6, 8-9, 12-13, 15 - 21
PhysicalDeviceName	All	4-6, 8-9, 12-13, 15 - 21
PowerNotify	PN_DISABLED	4-6, 8-9, 12-13, 15 - 21 : PN_DISABLED
PowerState	PS_UNKNOWN	4-6, 8-9, 12-13, 15 - 21 : PS_UNKNOWN

Table 43. Fiscal printer common properties (continued)

Property	JavaPOS	OPOS
State	All	4-6, 8-9, 12-13, 15 - 21

Table 44. Fiscal printer specific properties

Property	JavaPOS	OPOS
ActualCurrency	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
AdditionalHeader	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
AdditionalTrailer	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
AmountDecimalPlaces	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
AsyncMode	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapAdditionalHeader	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapAdditionalLines	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapAdditionalTrailer	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapAmountAdjustment	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapAmountNotPaid	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapChangeDue	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapCheckTotal	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapCoverSensor	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapDoubleWidth	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapDuplicateReceipt	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapEmptyReceiptIsVoidable	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapFiscalReceiptStation	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapFiscalReceiptType	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapFixedOutput	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapHasVatTable	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapIndependentHeader	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapItemList	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapJrnEmptySensor	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapJrnNearEndSensor	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapJrnPresent	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapMultiContractor	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapNonFiscalMode	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapOnlyVoidLastItem	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapOrderAdjustmentFirst	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapPackageAdjustment	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapPercentAdjustment	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapPositiveAdjustment	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapPostPreLine	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapPowerLossReport	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapPredefinedPaymentLines	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapReceiptNotPaid	All - false	4-6, 8-9, 12-13, 15 - 21 : false

Table 44. Fiscal printer specific properties (continued)

Property	JavaPOS	OPOS
CapRecEmptySensor	All - true	Not supported
CapRecNearEndSensor	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapRecPresent	All - true	Not supported
CapRemainingFiscalMemory	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapReservedWord	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapSetCurrency	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapSetHeader	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapSetPOSID	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapSetStoreFiscalID	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapSetTrailer	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapSetVatTable	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapSlpEmptySensor	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 17, 18, 20, 21, 23, 24	4-6, 8-9, 12-13, 15 - 21 : false
CapSlpFiscalDocument	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapSlpFullSlip	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapSlpNearEndSensor	All - false	Not supported
CapSlpPresent	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 17, 18, 20, 21, 23, 24	Not supported
CapSlpValidation	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapSubAmountAdjustment	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapSubPercentAdjustment	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapSubtotal	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapTotalizerType	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapTrainingMode	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapValidateJournal	All - false	4-6, 8-9, 12-13, 15 - 21 : false
CapXReport	All - false	4-6, 8-9, 12-13, 15 - 21 : false
ChangeDue	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
CheckTotal	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
ContractorId	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
CountryCode	All	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
CoverOpen	All - true or false	4-6, 8-9, 12-13, 15 - 21 : true or false
DateType	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
DayOpened	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
DescriptionLength	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
DuplicateReceipt	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
ErrorLevel	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
ErrorOutID	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
ErrorState	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
ErrorStation	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
ErrorString	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL

Table 44. Fiscal printer specific properties (continued)

Property	JavaPOS	OPOS
FiscalReceiptStation	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
FiscalReceiptType	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
FlagWhenIdle	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
JrnEmpty	7, 8, 9, 10, 11, 12, 13, 20, 21	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
JrnNearEnd	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
MessageLength	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
MessageType	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
NumHeaderLines	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
NumTrailerLines	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
NumVatRates	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
PostLine	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
PredefinedPaymentLines	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
PreLine	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
PrinterState	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
QuantityDecimalPlaces	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
QuantityLength	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
RecEmpty	All - true or false	4-6, 8-9, 12-13, 15 - 21 : true or false
RecNearEnd	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
RemainingFiscalMemory	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
ReservedWord	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
SlpEmpty	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 17, 18, 20, 21, 23	4-6, 8-9, 12-13, 15 - 21 : true or false
SlpNearEnd	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
SlipSelection	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
TotalizerType	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
TrainingModeActive	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL

Table 45. Fiscal printer common methods

Method	JavaPOS	OPOS
checkHealth	All	4-6, 8-9, 12-13, 15 - 21
claim	All	4-6, 8-9, 12-13, 15 - 21
clearInput		Not supported
clearOutput		Not supported
close	All	4-6, 8-9, 12-13, 15 - 21
compareFirmwareVersion	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
directIO	All	4-6, 8-9, 12-13, 15 - 21
open	All	4-6, 8-9, 12-13, 15 - 21
release	All	4-6, 8-9, 12-13, 15 - 21
resetStatistics	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL

Table 45. Fiscal printer common methods (continued)

Method	JavaPOS	OPOS
retrieveStatistics	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	E_ILLEGAL
updateFirmware	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL
updateStatistics	E_ILLEGAL	4-6, 8-9, 12-13, 15 - 21 : E_ILLEGAL

Table 46. Fiscal printer specific methods

Method	JavaPOS	OPOS
None supported	The communication to Fiscal Printer is only through directIO.	4-6, 8-9, 12-13, 15 - 21 : The communication to Fiscal Printer is only through directIO.

Table 47. Fiscal printer (DirectIO) events

Event	JavaPOS	OPOS
DirectIOEvent	All - yes	4-6, 8-9, 12-13, 15 - 21
ErrorEvent		Not supported
OutputCompleteEvent		Not supported
StatusUpdateEvent		Not supported
FPTR_SUE_COVER_OPEN	ALL	
FPTR_SUE_COVER_OK	ALL	
FPTR_SUE_JRN_EMPTY		Not supported
FPTR_SUE_JRN_NEAREMPTY		Not supported
FPTR_SUE_JRN_PAPEROK		Not supported
FPTR_SUE_REC_EMPTY		Not supported
FPTR_SUE_REC_NEAREMPTY		Not supported
FPTR_SUE_REC_PAPEROK		Not supported
FPTR_SUE_SLP_EMPTY		Not supported
FPTR_SUE_SLP_NEAREMPTY		Not supported
FPTR_SUE_SLP_PAPEROK		Not supported
FPTR_SUE_IDLE		Not supported
FPTR_SUE_JRN_COVER_OPEN	7, 8, 9, 10, 11, 12, 13, 20, 21	
FPTR_SUE_JRN_COVER_OK	7, 8, 9, 10, 11, 12, 13, 20, 21	
FPTR_SUE_REC_COVER_OPEN	ALL	
FPTR_SUE_REC_COVER_OK	ALL	
FPTR_SUE_SLP_COVER_OPEN	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 17, 18, 20, 21, 23, 24	
FPTR_SUE_SLP_COVER_OK	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 17, 18, 20, 21, 23, 24	

Supported properties and methods (fiscal printer: Italy)

Table 48. Fiscal printer (Italy) specific properties

Property	JavaPOS	OPOS
AutoDisable		Not supported
BinaryConversion		Not supported
CapCompareFirmwareVersion		False
CapPowerReporting		PR_STANDARD
CapStatisticsReporting	True	False
CapUpdateFirmware		False
CapUpdateStatistics	True	False
CheckHealthText		String
Claimed		True/False
DataCount		Not supported
DatEventEnabled		Not supported
DeviceEnabled		True/False
DeviceControlDescription		String
DeviceControlVersion		int
DeviceServiceDescription		String
DeviceServiceVersion		Supported
FreezeEvents		True/False
OutputID		int
PhysicalDeviceDescription		String
PhysicalDeviceName		String
PowerNotify		int
PowerState		E_ILLEGAL
State		int

Table 49. Fiscal printer (Italy) common methods

Method	JavaPOS	OPOS
checkHealth		Supported
claim		Supported
clearInput		Not supported
clearOutput		Not supported
close		Supported
compareFirmwareVersion		E_ILLEGAL
directIO		Supported
open		Supported
release		Supported
resetStatistics		E_ILLEGAL
retrieveStatistics		Supported
updateFirmware		E_ILLEGAL

Table 49. Fiscal printer (Italy) common methods (continued)

Method	JavaPOS	OPOS
updateStatistics		E_ILLEGAL

Table 50. Fiscal printer (Italy) specific methods

Method	JavaPOS	OPOS
setCurrency		E_ILLEGAL
setDate		Supported
setHeaderLine		Supported
setPOSID		Supported
setStoreFiscalID		E_ILLEGAL
setTrailerLine		E_ILLEGAL
setVatTable		E_ILLEGAL
setVatValue		E_ILLEGAL
beginFiscalReceipt		Supported
endFiscalReceipt		Supported
printDuplicateReceipt		E_ILLEGAL
printRecCash		E_ILLEGAL
printRecItem		Supported
printRecItemAdjustment		Supported
printRecItemAdjustmentVoid		Supported
printRecItemFuel		E_ILLEGAL
printRecItemFuelVoid		E_ILLEGAL
printRecItemRefund		Supported
printRecItemRefundVoid		E_ILLEGAL
printRecItemVoid		Supported
printRecMessage		Supported
printRecNotPaid		Supported
printRecPackageAdjustment		E_ILLEGAL
printRecPackageAdjustVoid		E_ILLEGAL
printRecRefund		Supported
printRecRefundVoid		E_ILLEGAL
printRecSubtotal		Supported
printRecSubtotalAdjustment		Supported
printRecSubtotalAdjustVoid		E_ILLEGAL
printRecTaxID		E_ILLEGAL
printRecTotal		Supported
printRecVoid		Supported
printRecVoidItem		Supported
beginFiscalDocument		Supported
endFiscalDocument		Supported
printFiscalDocumentLine		Supported

Table 50. Fiscal printer (Italy) specific methods (continued)

Method	JavaPOS	OPOS
beginItemList		E_ILLEGAL
endItemList		E_ILLEGAL
verifyItem		E_ILLEGAL
printPeriodicTotalsReport		Supported
printPowerLossReport		E_ILLEGAL
printReport		Supported
printXReport		E_ILLEGAL
printZReport		Supported
beginInsertion	KD3, KR3, GR3: Supported KR5, GF5, 3F: E_ILLEGAL KD5, 1NF, 2NF: Invalid command	KD3, KR3, GR3: Supported KD5, KR5, GF5, 3F, 1NF, 2NF: E_ILLEGAL
beginRemoval		Supported
endInsertion		E_ILLEGAL
endRemoval		E_ILLEGAL
beginFixedOutput		E_ILLEGAL
beginNonFiscal		Supported
beginTrainig		Supported
endFixedOutput		E_ILLEGAL
endNonFiscal		Supported
endTraining		Supported
printFixedOutput		E_ILLEGAL
printNormal		Supported
getData		Supported
getDate		Supported
getTotalizer		E_ILLEGAL
getVatEntry		E_ILLEGAL
clearError		E_ILLEGAL
resetPrinter		Supported

Table 51. Fiscal printer (Italy) events

Event	JavaPOS	OPOS
DirectIOEvent	All: yes	4-6, 8-9, 12-13, 15-21
ErrorEvent		Not supported
OutputCompleteEvent		Not supported
StatusUpdateEvent		Not supported
FPTR_SUE_COVER_OPEN	ALL	
FPTR_SUE_COVER_OK	ALL	
FPTR_SUE_JRN_EMPTY		Not supported
FPTR_SUE_JRN_NEAREMPTY		Not supported

Table 51. Fiscal printer (Italy) events (continued)

Event	JavaPOS	OPOS
FPTR_SUE_JRN_PAPEROK		Not supported
FPTR_SUE_REC_EMPTY		Not supported
FPTR_SUE_REC_NEAREMPTY		Not supported
FPTR_SUE_REC_PAPEROK		Not supported
FPTR_SUE_SLP_EMPTY		Not supported
FPTR_SUE_SLP_NEAREMPTY		Not supported
FPTR_SUE_SLP_PAPEROK		Not supported
FPTR_SUE_IDLE		Not supported
FPTR_SUE_JRN_COVER_OPEN	7, 8, 9, 10, 11, 12, 13, 20, 21	
FPTR_SUE_JRN_COVER_OK	7, 8, 9, 10, 11, 12, 13, 20, 21	
FPTR_SUE_REC_COVER_OPEN	ALL	
FPTR_SUE_REC_COVER_OK	ALL	
FPTR_SUE_SLP_COVER_OPEN	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 17, 18, 20, 21, 23, 24	
FPTR_SUE_SLP_COVER_OK	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 17, 18, 20, 21, 23, 24	

Supported properties and methods (fiscal printer: Greece)

Table 52. Fiscal printer (Greece) common properties

Property	JavaPOS	OPOS
AutoDisable		Not supported
CapCompareFirmwareVersion		False
CapPowerReporting		PR_STANDARD
CapStatisticsReporting		True
CapUpdateFirmware		False
CapUpdateStatistics		False
CheckHealthText		String
Claimed		True/False
DataCount		Not supported
DatEventEnabled		Not supported
DeviceEnabled		True/False
DeviceControlDescription		String
DeviceControlVersion		int
DeviceServiceDescription		String
DeviceServiceVersion		int
FreezeEvents		True/False
OutputID		E_ILLEGAL
PhysicalDeviceDescription		String
PhysicalDeviceName		String
PowerNotify		int

Table 52. Fiscal printer (Greece) common properties (continued)

Property	JavaPOS	OPOS
PowerState		int
State		int

Table 53. Fiscal printer (Greece) specific properties

Property	JavaPOS	OPOS
ActualCurrency		E_ILLEGAL
AdditionalHeader		E_ILLEGAL
AdditionalTrailer		E_ILLEGAL
AmountDecimalPlaces		2 (int)
AsyncMode		False
CapAdditionalHeader		False
CapAdditionalLines		False
CapAdditionalTrailer		False
CapAmountAdjustment		True
CapAmountNotPaid		False
CapChangeDue		False
CapCheckTotal		True
CapCoverSensor		True
CapDoubleWidth		True
CapDuplicateReceipt		False
CapEmptyReceiptsVoidable		True
CapFiscalReceiptStation		True
CapFiscalReceiptType		True
CapFixedOutput		False
CapHasVatTable		True
CapIndependentHeader		False
CapItemList		False
CapJrnEmptySensor		False
CapJrnNearEndSensor		True
CapJrnPresent		False
CapMultiContractor		False
CapNonFiscalMode		True
CapOnlyVoidLastItem		False
CapOrderAdjustmentFirst		False
CapPackageAdjustment		False
CapPercentAdjustment		False
CapPositiveAdjustment		True
CapPositiveSubtotalAdjustment		True
CapPostPreLine		False
CapPowerLossReport		False

Table 53. Fiscal printer (Greece) specific properties (continued)

Property	JavaPOS	OPOS
CapPredefinedPaymentLines		False
CapReceiptNotPaid		False
CapRecEmptySensor		True
CapRecNearEndSensor		False
CapRecPresent		True
CapRemainingFiscalMemory		False
CapReservedWord		False
CapSetCurrency		False
CapSetHeader		True
CapSetPOSID		True
CapSetStoreFiscalID		False
CapSetTrailer		False
CapSetVatTable		True
CapSlpEmptySensor		True
CapSlpFiscalDocument		False
CapSlpFullSlip		True
CapSlpNearEndSensor		False
CapSlpPresent		True
CapSlpValidation		False
CapSubAmountAdjustment		True
CapSubPercentAdjustment		False
CapSubtotal		True
CapTotalizerType		True
CapTrainingMode		False
CapValidateJournal		False
CapXReport		True
ChangeDue	E_ILLEGAL	
CheckTotal		True
ContractorId	E_ILLEGAL	
CountryCode	FPTR_CC_GREECE	
CoverOpen		True/False
DateType	FPTR_DT_RTC/FPTR_DT_START	
DayOpened		True/False
DescriptionLength	38 (int)	
DuplicateReceipt	E_ILLEGAL	
ErrorLevel	EL_NONE	
ErrorOutID	E_ILLEGAL	
ErrorState	E_ILLEGAL	
ErrorStation	E_ILLEGAL	
ErrorResponse	E_ILLEGAL	

Table 53. Fiscal printer (Greece) specific properties (continued)

Property	JavaPOS	OPOS
FiscalReceiptStation		FPTR_RS_RECEIPT
FiscalReceiptType		FPTR_RT_SALES/ FPTR_RT_CASH_IN/ FPTR_CASH_OUT
FlagWhenIdle		E_ILLEGAL
JrnEmpty		True/False
JrnNearEnd		False
MessageLength		0 (int)
MessageType		E_ILLEGAL
NumHeaderLines		6 (int)
NumTrailerLines		0 (int)
NumVatRates		5 (int)
PostLine		E_ILLEGAL
PredefinedPaymentLines		E_ILLEGAL
PreLine		E_ILLEGAL
PrinterState		FPTR_PS_MONITOR/ FPTR_PS_FISCAL_RECEIPT/ FPTR_PSFISCAL_RECEIPT_TOTAL/ FPTR_PSFISCAL_RECEIPT_ENDING/ FPTR_PS_NONFISCAL
QuantityDecimalPlaces		3 (int)
QuantityLength		7 (int)
RecEmpty		True/False
RecNearEnd		False
RemainingFiscalMemory		E_ILLEGAL
ReservedWord		E_ILLEGAL
SlpEmpty		True/False
SlpNearEnd		False
SlipSelection		FPTR_SS_FULL_LENGTH
TotalizerType		FPTR_TT_DAY / FPTR_TT_RECEIPT / FPTR_TT_GRAND
TrainingModeActive		False

Table 54. Fiscal printer (Greece) common methods

Method	JavaPOS	OPOS
checkHealth		Supported
claim		Supported
clearInput		Not supported
clearOutput		E_ILLEGAL
close		Supported
compareFirmwareVersion		E_ILLEGAL

Table 54. Fiscal printer (Greece) common methods (continued)

Method	JavaPOS	OPOS
directIO		Supported
open		Supported
release		Supported
resetStatistics		E_ILLEGAL
retrieveStatistics		E_ILLEGAL
updateFirmware		E_ILLEGAL
updateStatistics		E_ILLEGAL

Table 55. Fiscal printer (Greece) specific methods

Method	JavaPOS	OPOS
setCurrency		E_ILLEGAL
setDate		E_ILLEGAL
setHeaderLine		Supported
setPOSID		Supported
setStoreFiscalID		E_ILLEGAL
setTrailerLine		E_ILLEGAL
setVatTable		Supported
setVatValue		E_ILLEGAL
beginFiscalReceipt		Supported
endFiscalReceipt		Supported
printDuplicateReceipt		E_ILLEGAL
printRecCash		Supported
printRecItem		Supported
printRecItemAdjustment		Supported
printRecItemAdjustmentVoid		Supported
printRecItemFuel		E_ILLEGAL
printRecItemFuelVoid		E_ILLEGAL
printRecItemRefund		Supported
printRecItemRefundVoid		Supported
printRecItemVoid		Supported
printRecMessage		E_ILLEGAL
printRecNotPaid		E_ILLEGAL
printRecPackageAdjustment		E_ILLEGAL
printRecPackageAdjustVoid		E_ILLEGAL
printRecRefund		Supported
printRecRefundVoid		Supported
printRecSubtotal		Supported
printRecSubtotalAdjustment		Supported
printRecSubtotalAdjustVoid		Supported
printRecTaxID		E_ILLEGAL

Table 55. Fiscal printer (Greece) specific methods (continued)

Method	JavaPOS	OPOS
printRecTotal		Supported
printRecVoid		Supported
printRecVoidItem		Supported
beginFiscalDocument		E_ILLEGAL
endFiscalDocument		E_ILLEGAL
printFiscalDocumentLine		E_ILLEGAL
beginItemList		E_ILLEGAL
endItemList		E_ILLEGAL
verifyItem		E_ILLEGAL
printPeriodicTotalsReport		Supported
printPowerLossReport		E_ILLEGAL
printReport		Supported
printXReport		Supported
printZReport		Supported
beginInsertion		Supported
beginRemoval		Supported
endInsertion		E_ILLEGAL
endRemoval		E_ILLEGAL
beginFixedOutput		E_ILLEGAL
beginNonFiscal		Supported
beginTrainig		E_ILLEGAL
endFixedOutput		E_ILLEGAL
endNonFiscal		Supported
endTraining		E_ILLEGAL
printFixedOutput		E_ILLEGAL
printNormal		Supported
getData		Supported
getDate		Supported
getTotalizer		Supported
getVatEntry		supported
clearError		E_ILLEGAL
resetPrinter		Supported

Table 56. Fiscal printer (Greece) events

Event	JavaPOS	OPOS
DirectIOEvent	All: yes	4-6, 8-9, 12-13, 15-21
ErrorEvent		Not supported
OutputCompleteEvent		Not supported
StatusUpdateEvent		Not supported
FPTR_SUE_COVER_OPEN	ALL	

Table 56. Fiscal printer (Greece) events (continued)

Event	JavaPOS	OPOS
FPTR_SUE_COVER_OK	ALL	
FPTR_SUE_JRN_EMPTY		Not supported
FPTR_SUE_JRN_NEAREMPTY		Not supported
FPTR_SUE_JRN_PAPEROK		Not supported
FPTR_SUE_REC_EMPTY		Not supported
FPTR_SUE_REC_NEAREMPTY		Not supported
FPTR_SUE_REC_PAPEROK		Not supported
FPTR_SUE_SLP_EMPTY		Not supported
FPTR_SUE_SLP_NEAREMPTY		Not supported
FPTR_SUE_SLP_PAPEROK		Not supported
FPTR_SUE_IDLE		Not supported
FPTR_SUE_JRN_COVER_OPEN	7, 8, 9, 10, 11, 12, 13, 20, 21	
FPTR_SUE_JRN_COVER_OK	7, 8, 9, 10, 11, 12, 13, 20, 21	
FPTR_SUE_REC_COVER_OPEN	ALL	
FPTR_SUE_REC_COVER_OK	ALL	
FPTR_SUE_SLP_COVER_OPEN	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 17, 18, 20, 21, 23, 24	
FPTR_SUE_SLP_COVER_OK	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 17, 18, 20, 21, 23, 24	

JavaPOS configuration

Monitor detection

Detection of a monitor is available in the JavaPOS drivers; it can be sensed by the VGA or Data port.

Monitors supported:

- USB 4820 SurePoint
- EIA232 4820 SurePoint with Elo touch screen technology
- RS485 4820 SurePoint
- EIA232 SurePOS 500 Family
- EIA232 Anyplace Kiosk Family
- USB 6149 Touch Screen

To change the configuration settings, refer to the posj.properties file and the following properties:

- com.ibm.posj.FiscalPrinter.MonitorDetection
- com.ibm.posj.FiscalPrinter.EnableDisplaySensing

Note: For EIA232 4820 SurePoint, Windows relies on the touch driver provided by EloTouch Inc. Tested versions include 4.8.7b and 5.3.

JavaPOS DirectIO calls

The directIO commands in this section are supported for the fiscal printer device. The following is the correct syntax:

```
directIO ( command: int32, inout data: int32 , inout obj: object):
void { raises-exception }
```

To access FiscalPrinter constants, import: com.ibm.jpos.services.*

Fiscal information command

This obtains information about the fiscal device. Valid values are returned only after the first FISCAL_IPL_END_STATUS event.

Table 57. Parameters for FISCAL_INFORMATION subcommand

Parameter	Type	Value
Command	Int32	DirectIO.FISCAL_ID
Data	Int32	DirectIO.FISCAL_INFORMATION
Obj	Object	FiscalInformation Object

Note: For the FiscalInformation object:

- getCountry() returns the Country Code value that is specified in the Fiscal Printer Hardware Supplements.
- getVersion() returns the microcode version (EC level).
- getFiscalPowerInterrupted() returns true if the printer was turned off in the middle of a command; otherwise, it returns false. For more information, refer to the Fiscal Printer Hardware Supplements available on the Web at www.toshibacommerce.com/support.

Fiscal read command

This subcommand reads data from the fiscal device; it should be issued after receiving a FISCAL_DATA_AVAIL directIOEvent from the fiscal device.

Table 58. Parameters for FISCAL_READ subcommand

Parameter	Type	Value
Command	Int32	com.ibm.jpos.services.DirectIO.FISCAL_ID
Data	Int32	com.ibm.jpos.services.DirectIO.FISCAL_READ
Obj	Object	A byte [] to be filled with data.

Note: If a FISCAL_DATA_AVAIL event is received and the application has not read the previous fiscal data, the old data is replaced by the new incoming data. The next FISCAL_READ command reads the new data.

Fiscal write command

This writes data to the fiscal device.

Table 59. Parameters for FISCAL_WRITE subcommand

Parameter	Type	Value
Command	Int32	DirectIO.FISCAL_ID
Data	Int32	DirectIO.FISCAL_WRITE
Obj	Object	A byte [] with the data to send. The byte [] (object parameter) must contain the exact data to send. This data is a Fiscal Command without the prefix {X'1B',X'66'}.

Note: DirectIO is a synchronous method and returns successfully when the data is written to the device. However, command execution is not complete until a FISCAL_STATUS or FISCAL_ERROR event is received. If another command is submitted before the reception of these events, the second command is ignored. This means that there is no buffering of FISCAL_WRITE subcommands.

Fiscal notify command

This sets fiscal notification On or Off. If fiscal notification is On, every fiscal command returns a status event to the application.

Table 60. Parameters for FISCAL_NOTIFY subcommand

Parameter	Type	Value
Command	Int32	DirectIO.FISCAL_ID
Data	Int32	DirectIO.FISCAL_NOTIFY
Obj	Object	An Integer object with one of the following values: DirectIO.FISCAL_NOTIFY_ON or DirectIO.FISCAL_NOTIFY_OFF

JavaPOS DirectIO events

Fiscal error event

Indicates that a fiscal error has occurred. The DirectIOEvent parameters are shown in Table 61.

Table 61. Parameters for the FISCAL_ERROR event

Parameter	Type	Value
Event Number	Int32	DirectIO.FISCAL_ERROR
Data	Int32	Cause of the fiscal error event (Fiscal Printer Command Return Code).
Obj	Object	Integer object that indicates the error codes as in the POS Subsystem (see Table 62). The Integer value is the result of an OR operation of the current error codes (printer status).

Table 62. Error codes for FISCAL_ERROR event

Error code	Value
PosSTATUS_COVER_OPEN	X'0001'
PosSTATUS_TRANSPORT_ERROR	X'0002'
PosSTATUS_SJ_PAPER_ERROR	X'0004'
PosSTATUS_DOCUMENT_AT_FRONT	X'0008'
PosSTATUS_DOCUMENT_AT_TOP	X'0010'
PosSTATUS_DOCUMENT_READY	X'0020'
PosSTATUS_HEAD_PARKED	X'0040'
PosSTATUS_INSERTED_FORWARD	X'0080'
PosSTATUS_ERROR_PENDING	X'0100'
PosSTATUS_DL_FRONT_LOAD_ERROR	X'0200'
PosSTATUS_DL_TOP_LOAD_ERROR	X'0400'
PosSTATUS_PRINTER_ONLINE	X'1000'
PosSTATUS_MICR_INSTALLED	X'2000'

Table 62. Error codes for FISCAL_ERROR event (continued)

Error code	Value
PosSTATUS_CR_PAPER_LOW	X'4000'
PosSTATUS_SJ_PAPER_LOW	X'8000'

Fiscal status event

Indicates that the last fiscal command completed successfully. Fiscal notification must be On to receive these events. The DirectIOEvent parameters are shown in Table 63.

Table 63. Parameters for the FISCAL_STATUS event

Parameter	Type	Value
Event Number	Int32	DirectIO.FISCAL_STATUS
Data	Int32	Always 0
Obj	Object	Integer object that indicates the last command executed. For example, for the command {X'FF', X'10', '0', '0', '0', '0', '0', '0', '0', '0', '0', 'F'}, the object returns the value X'FF' (the first byte indicating the command). For more information, see the Fiscal Printer Hardware Supplements are available on the Web at www.toshibacommerce.com .

Fiscal data available event

Indicates that data is available to be read from the fiscal device. The DirectIOEvent parameters are shown in Table 64.

Table 64. Parameters for the FISCAL_DATA_AVAIL event

Parameter	Type	Value
Command	Int32	DirectIO.FISCAL_DATA_AVAIL
Data	Int32	Always 0
Obj	Object	Integer object that indicates the size in bytes of the data to be read.

Fiscal raw status event

RAW_STATUS event is not supported for Fiscal Printer DirectIO in OPOS.

Fiscal IPL end status event

Indicates that the initial program load (IPL) process is complete and that the application can start sending commands to the fiscal printer.

The directIO parameters are shown in Table 65.

Table 65. Parameters for the FISCAL_IPL_END_STATUS event

Parameter	Type	Value
Event Number	Int32	DirectIO.FISCAL_IPL_END_STATUS
Data	Int32	Always 0
Obj	Object	Always null

Note:

When the Fiscal Printer is reset or powered on, an IPL occurs. During IPL, the printer is not operational and rejects fiscal commands. In some situations, the IPL process can take more than one minute.

If the application tries to write a command before it receives a FISCAL_IPL_END_STATUS, the driver ignores the command.

Additional JavaPOS information

Device sensing

In certain fiscal countries, the fiscal printer must know that one or more devices (such as displays) are active before completing certain fiscal functions, usually related to totals accumulation (item sale commands, voucher total commands, etc.). If the required devices are not active, the fiscal printer will reject the appropriate commands with a failure return code to the application.

Countries requiring device sense:

- Italy, firmware v16 or higher
- Greece, firmware v09 or higher

Device sensing is disabled by default. To change the configuration settings, refer to the posj.properties file and the following property: - com.ibm.posj.FiscalPrinter.EnableDisplaySensing.

By default, Toshiba JavaPOS drivers provide the existence notification of the external displays. The supported combinations include the following:

1. EIA232 Fiscal Printer Models (4610-1NF only)

The EIA232 Fiscal Printer devices can sense external devices by Toshiba JavaPOS

Multiple external devices sensed:

- RS485 Line Display
- USB Line Display
- USB 4820 SurePoint device–Italy only
- RS485 4820 SurePoint device–Italy only
- USB 6149 Touch Screen–Italy only

And only one of:

- RS232 4820 SurePoint touch device–Italy only
- SurePOS 500 Touch screen–Italy only
- AnyPlace Kiosk Touch Screen–Italy only
- TCxWave Touch Screen–Italy only
- TCxFight Touch Screen–Italy only

2. RS485 Fiscal Printer Models

Italy only: The RS485 Fiscal Printer devices can sense external devices by Toshiba JavaPOS.

External devices sensed:

- RS485 Line Displays

And only one external devices of:

- USB Line Display
- RS232 4820 SurePoint touch device
- USB 4820 SurePoint device
- RS485 4820 SurePoint device
- USB 6149 Touch Screen

3. USB Fiscal Printer Models

The USB Fiscal Printer devices can sense external devices by Toshiba JavaPOS.

Multiple external devices sensed:

- RS485 Line Display
- USB Line Display
- USB 4820 SurePoint device—Italy only
- RS485 4820 SurePoint device—Italy only
- USB 6149 Touch Screen—Italy only

And only one of:

- RS232 4820 SurePoint touch device – Italy only
- SurePOS 500 Touch screen – Italy only
- AnyPlace Kiosk Touch Screen – Italy only
- TCxWave Touch Screen – Italy only
- TCxFlight Touch Screen – Italy only

Refer to the “Monitor detection” on page 80 section for configuration details and supported hardware.

OPOS DirectIO calls

The directIO commands in this section are supported for the fiscal printer device. The following is the correct syntax:

```
LONG DirectIO ( LONGCommand, LONG*pData, BSTR*pString);
```

To access FiscalPrinter constants, include IBMFPTR.H.

Note: For all operations, use BinaryConversion rather than OPOS_BC_NONE so that data is not prematurely truncated.

Fiscal information command

Obtains information about the fiscal device. Valid values are returned only after the first FISCAL_IPL_END_STATUS event.

Table 66. Parameters for

Parameter	Type	Value
Command	LONG	FISCAL_ID
pData	LONG*	FISCAL_INFORMATION
pString	BSTR*	FiscalInformation

Note: The fiscal information is put into the output string as a three-field comma-delimited string, encoded according to the current binary conversion mode. The ordering of the three fields in the comma-delimited string is: Country, Power interrupted, Version.

Fiscal read command

This subcommand reads data from the fiscal device and should be issued after receiving a FISCAL_DATA_AVAIL directIOEvent from the fiscal device.

Table 67. Parameters for FISCAL_READ subcommand

Parameter	Type	Value
Command	LONG	FISCAL_ID

Table 67. Parameters for FISCAL_READ subcommand (continued)

Parameter	Type	Value
pData	LONG*	FISCAL_READ
pString	BSTR*	A BSTR to be filled with data.

Note: If a FISCAL_DATA_AVAIL event is received and the application has not read the previous fiscal data, the old data is replaced by the new incoming data. The next FISCAL_READ command reads the new data.

Fiscal write command

This subcommand writes data to the fiscal device.

Table 68. Parameters for FISCAL_WRITE subcommand

Parameter	Type	Value
Command	LONG	FISCAL_ID
pData	LONG*	FISCAL_WRITE
pString	BSTR*	A BSTR with the data to send. The BSTR must contain the exact data to send. This data is a Fiscal Command without the prefix {X'1B',X'66'}.

Note: DirectIO is a synchronous method and returns successfully when the data is written to the device. However, command execution is not complete until a FISCAL_STATUS or FISCAL_ERROR event is received. If another command is submitted before the reception of these events, the second command is ignored. This means that there is no buffering of FISCAL_WRITE subcommands.

Fiscal notify command

Set fiscal notification On or Off. If fiscal notification is On, every fiscal command returns a status event to the application.

Table 69. Parameters for FISCAL_NOTIFY subcommand

Parameter	Type	Value
Command	LONG	FISCAL_ID
pData	LONG*	FISCAL_NOTIFY
pString	BSTR*	The first byte in the string has one of the following values: FISCAL_NOTIFY_ON or FISCAL_NOTIFY_OFF

OPOS DirectIO events

Fiscal error event

Indicates that a fiscal error has occurred. The DirectIOEvent parameters are shown in Table 70.

Table 70. Parameters for the FISCAL_ERROR event

Parameter	Type	Value
Event Number	LONG	FISCAL_ERROR

Table 70. Parameters for the FISCAL_ERROR event (continued)

Parameter	Type	Value
pData	LONG*	Cause of the fiscal error event (Fiscal Printer Command Return Code).
pString	BSTR*	Indicates the error codes as in the POS Subsystem (see Table 62 on page 82. The value is a string of the hex values of an OR operation of the current error codes (printer status).

Table 71. Error codes for FISCAL_ERROR event

Error code	Value
PosSTATUS_COVER_OPEN	X'0001'
PosSTATUS_TRANSPORT_ERROR	X'0002'
PosSTATUS_SJ_PAPER_ERROR	X'0004'
PosSTATUS_DOCUMENT_AT_FRONT	X'0008'
PosSTATUS_DOCUMENT_AT_TOP	X'0010'
PosSTATUS_DOCUMENT_READY	X'0020'
PosSTATUS_HEAD_PARKED	X'0040'
PosSTATUS_INSERTED_FORWARD	X'0080'
PosSTATUS_ERROR_PENDING	X'0100'
PosSTATUS_DI_FRONT_LOAD_ERROR	X'0200'
PosSTATUS_DI_TOP_LOAD_ERROR	X'0400'
PosSTATUS_PRINTER_ONLINE	X'1000'
PosSTATUS_MICR_INSTALLED	X'2000'
PosSTATUS_CR_PAPER_LOW	X'4000'
PosSTATUS_SJ_PAPER_LOW	X'8000'

Fiscal status event

Indicates that the last fiscal command completed successfully. Fiscal notification must be On to receive these events. The DirectIOEvent parameters are shown in Table 72.

Table 72. Parameters for the FISCAL_STATUS event

Parameter	Type	Value
Event Number	LONG	FISCAL_STATUS
pData	LONG*	Always 0
pString	BSTR*	BSTR that indicates the last command executed. For example, for the command {X'FF', X'10', '0', '0', '0', '0', '0', '0', '0', '0', '0', 'F'}, the object returns the value X'FF' (the first byte indicating the command). For more information, see the Fiscal Printer Hardware Supplements available on the Toshiba Global Commerce Solutions website: www.toshibacommerce.com .

Fiscal data available event

Indicates that data is available to be read from the fiscal device. The DirectIOEvent parameters are shown in Table 73.

Table 73. Parameters for the FISCAL_DATA_AVAIL event

Parameter	Type	Value
Event Number	LONG	FISCAL_DATA_AVAIL
pData	LONG*	Always 0
pString	BSTR*	BSTR that indicates the size in bytes of the data to be read.

Fiscal raw status event

Indicates status has been received. The directIO parameters are shown in Table 74.

Table 74. Parameters for the FISCAL_RAW_STATUS event

Parameter	Type	Value
Event Number	LONG	FISCAL_RAW_STATUS
pData	LONG*	Always 0
pString	BSTR*	Containing the raw status data.

Note: The object parameter is filled with bytes reported by the Fiscal Printer Device. The processing of these bytes must be done according to the Fiscal Printer Hardware Supplements available on the Toshiba Global Commerce Solutions website: www.toshibacommerce.com.

Fiscal IPL end status event

Indicates that the initial program load (IPL) process is complete and that the application can start sending commands to the fiscal printer.

The directIO parameters are shown in Table 75.

Table 75. Parameters for the FISCAL_IPL_END_STATUS event

Parameter	Type	Value
Event Number	LONG	FISCAL_END_STATUS
pData	LONG*	Always 0
pString	BSTR*	Always null

Note: When the fiscal printer is reset or powered on, an IPL occurs. During IPL, the printer is not operational and rejects fiscal commands. In some situations, the IPL process can take more than one minute. If the application tries to write a command before it receives a FISCAL_IPL_END_STATUS, the driver ignores the command.

Additional OPOS information

Suspended fiscal transaction

OPOS provides a mechanism to allow the NonFiscal Lines to be printed even after a printRecSubtotal() is called. This is done by buffering the subTotal call until one of the following methods is processed.

NonFiscal methods that will hold buffer of subTotal:

```
printRecMessage()  
printNormal()  
getData() <used for barcodePrint>
```

Buffered subTotal is fired by the following printRecXXX methods:

```
printRecItem  
printRecItemAdjustment  
printRecItemAdjustmentVoid  
printRecItemRefund  
printRecItemVoid  
printRecNotPaid  
printRecRefund  
printRecSubtotalAdjustment  
printRecTotal  
printRecVoid  
printRecVoidItem  
resetPrinter
```

The function will only be effective when the following registry is set:

Name BufferedSubTotal

Type REG_SZ

Value true/TRUE: Enable the buffering of NonFiscal Lines after printRecSubtotal() is called default, other values: Disable the function, it will behave as per UPOS specs

Location

HKEY_LOCAL_MACHINE\SOFTWARE\OLEforRetail\ServiceOPOS\FiscalPrinter\LogicalName

Display device sensing

In certain fiscal countries, the fiscal printer must know that one or more devices (such as displays) are active before completing certain fiscal functions. These functions are usually related to totals accumulation, such as item sale commands and voucher total commands. If the required devices are not active, the fiscal printer will reject the appropriate commands with a failure return code to the application.

Countries requiring device sense:

- Italy, firmware v16 or higher
- Greece, firmware v09 or higher

Device sensing is enabled by default for both Italy and Greece Fiscal Printers configured in OPOSConfig. No support is provided for DirectIO printers configured in OPOSConfig.

Italian fiscal regulation requires at least two displays and Greek fiscal regulation requires at least one display to be connected to either the system or printer to perform a transaction. Table 76 on page 90 contains information on the display devices supported by this functionality.

Note: The fiscal printers noted here have the following limitations:

- **GR3/GR5/KR3/KR5 Models (RS-485, Italy):** Only one external display device connected to the POS system can be detected.

- KD3/KD5 Models (RS-232, Italy):** Two displays must be sensed, and both displays must be attached either to the POS or to the printer. No mix is allowed.

Table 76. Supported display devices

Connectivity	Display device	Notes
EIA232	SurePOS 500 2x20 displays	<p>The following registry* must be set:</p> <p>Name: IntegratedDisplay Type: REG_SZ Value: True; treat SurePOS 500 2x20 displays as Integrated Display for Fiscal Printer display sensing (Italy)</p> <p>This setting is for SurePOS 500 system-attached 2x20 displays only.</p>
USB	<ul style="list-style-type: none"> Line Display 2x20 one-sided/two-sided, APA 6149 Touch Screen <p>Display devices can also connected to AnyPlace POS Hub</p>	A two-sided line display is considered two display devices.
RS485	Line Display 2x20 one-sided/two-sided, APA	A two-sided line display is considered two display devices.
System Integrated Display	<ul style="list-style-type: none"> AnyPlace Kiosk Family TCxWave Touch Screen TCxFlight Touch Screen 	
Data port (RS232)	4820 SurePoint with Elo touch screen technology (RS232)	<p>The following registry^ can be set:</p> <p>Name: EloMonitorPollTime Type: REG_DWORD Value: Contains the interval (in milliseconds) between the polling of the Elo Monitor Default/Not present: 2000</p> <p>When a 4820 SurePoint display device is connected to the system via data port (RS232), do not configure the VGA/DisplayPort any further.</p>
Data port (USB)	4820 SurePoint (USB)	<p>When a 4820 SurePoint display device is connected to the system via data port (USB), it is automatically sensed as one of the valid displays. Do not configure the VGA/DisplayPort any further.</p> <p>Generic display devices connected via USB can still be detected by setting the vendorId and productId of the device in the following registry^:</p> <p>Name: MonitorDevices Type: REG_SZ Value: vendorID<1>,productId<1>;vendorID<2>,productId<2>;...vendorID<n>,productId<n>;</p> <p>The vendorID and productId are separated by a comma and each device is separated by a semicolon. There is no whitespace between each. Put hex value prefixed by '0x'.</p>

Table 76. Supported display devices (continued)

Connectivity	Display device	Notes
VGA/DisplayPort	General VGA/Display monitors	<p>Note: For systems running Microsoft Windows 7 and above only.</p> <p>The following registry^ must be set:</p> <p>Name: DisplaySenseMonitor Type: REG_SZ Value: Contains the Hardware ID of the monitor to be detected. The Hardware ID can be found in the properties of the monitor in Device Manager. For example, if the Hardware ID of the monitor is "MONITOR\TOS1234", "TOS1234" should be set in the "DisplaySenseMonitor" registry.</p> <p>The polling interval can be set from the following registry^:</p> <p>Name: VideoCableMonitorPollTime Type: REG_DWORD Value: Contains the interval (in milliseconds) between the polling of the video cable monitor Default/Not present: 1000</p>

*The registry path is HKEY_LOCAL_MACHINE\SOFTWARE\OLEforRetail\ServiceOPOS\FiscalPrinter\LogicalName.

^The registry path is HKEY_LOCAL_MACHINE\SOFTWARE\OLEforRetail\ServiceOPOS\FiscalPrinter.

Note:

- **GB3/GB5/KB3/KB5 (USB, Greece):** Display sensing is not required for USB Greece fiscal printer.
- **GR3/GR5/KR3/KR5 (RS485, Greece):** Display sensing with two line displays (or one two-sided line display) is required. The line displays must be RS485 VFD or LCD displays and must be attached to the POS system, *not* to the printer. Do not attach an external line display with other buses (USB, RS232) to the system as well; only RS485 line displays can be attached to the system.

OPOS getData method (fiscal printer: Italy)

Table 77. Parameter support status and value for fiscal printer getData method (Italy)

Description	dataItem	optArgs	Data	Supported?
	<u>Identification data</u>			
Get the Fiscal Printer's firmware release number.	FPTR_GD_FIRMWARE	Not used	Value is set by the method	Y
Get the Fiscal Printer's fiscal ID.	FPTR_GD_PRINTER_ID	Not used	Value is set by the method	Y
	<u>Totals</u>			
Get the current receipt total.	FPTR_GD_CURRENT_TOTAL	Not used	Value is set by the method	Y
Get the daily total.	FPTR_GD_DAILY_TOTAL	Not used	Value is set by the method	Y
Get the Fiscal Printer's grand total.	FPTR_GD_GRAND_TOTAL			N
Get the total number of voided receipts.	FPTR_GD_MID_VOID	Not used	Value is set by the method	Y
Get the current total of not paid receipts.	FPTR_GD_NOT_PAID	Not used	Value is set by the method	Y
Get the number of fiscal receipts printed.	FPTR_GD_RECEIPT_NUMBER	Not used	Value is set by the method	Y
Get the current total of refunds.	FPTR_GD_REFUND	Not used	Value is set by the method	Y
Get the current total of voided refunds.	FPTR_GD_REFUND_VOID			N

Table 77. Parameter support status and value for fiscal printer getData method (Italy) (continued)

Description	dataItem	optArgs	Data	Supported?
Fiscal memory counts				
Get the grand number of configuration blocks.	FPTR_GD_NUMB_CONFIG_BLOCK			N
Get the grand number of currency blocks.	FPTR_GD_NUMB_CURRENCY_BLOCK			N
Get the grand number of header blocks.	FPTR_GD_NUMB_HDR_BLOCK			N
Get the grand number of reset blocks.	FPTR_GD_NUMB_RESET_BLOCK			N
Get the grand number of VAT blocks.	FPTR_GD_NUMB_VAT_BLOCK			N
Counter				
Get the number of daily fiscal documents.	FPTR_GD_FISCAL_DOC			N
Get the number of daily voided fiscal documents.	FPTR_GD_FISCAL_DOC_VOID			N
Get the number of daily fiscal sales receipts.	FPTR_GD_FISCAL_REC	Not used	Value is set by the method	Y
Get the number of daily voided fiscal sales receipts.	FPTR_GD_FISCAL_REC_VOID	Not used	Value is set by the method	Y
Get the number of daily non fiscal documents.	FPTR_GD_NONFISCAL_DOC	Not used	Value is set by the method	Y
Get the number of daily voided non fiscal documents.	FPTR_GD_NONFISCAL_DOC_VOID			N
Get the number of daily non fiscal receipts.	FPTR_GD_NONFISCAL_REC	Not used	Value is set by the method	Y
Get the Fiscal Printer's restart count	FPTR_GD_RESTART	Not used	Value is set by the method	Y
Get the number of daily simplified invoices.	FPTR_GD_SIMP_INVOICE			N
Get the Z report number.	FPTR_GD_Z_REPORT	Not used	Value is set by the method	Y
Fixed fiscal printer text				
Get the payment description used in the printRecTotal method, defined by the given identifier in the optArgs argument. Valid only if the CapPredefinedPaymentLines property is true.	FPTR_GD_TENDER	FPTR_PDL_CASH FPTR_PDL_CHEQUE FPTR_PDL_CHITTY FPTR_PDL_COUPON FPTR_PDL_CURRENCY FPTR_PDL_DRIVEN_OFF FPTR_PDL_EFT_IMPRINTER FPTR_PDL_EFT_TERMINAL FPTR_PDL_TERMINAL_IMPRINTER FPTR_PDL_FREE_GIFT FPTR_PDL_GIRO FPTR_PDL_HOME FPTR_PDL_IMPRINTER_WITH_ISSUER FPTR_PDL_LOCAL_ACCOUNT FPTR_PDL_LOCAL_ACCOUNT_CARD FPTR_PDL_PAY_CARD FPTR_PDL_PAY_CARD_MANUAL FPTR_PDL_PREPAY FPTR_PDL_PUMP_TEST FPTR_PDL_SHORT_CREDIT FPTR_PDL_STAFF FPTR_PDL_VOUCHER		N
Linecounter				

Table 77. Parameter support status and value for fiscal printer *getData* method (Italy) (continued)

OPOS getData method (fiscal printer: Greece)

Table 78. Parameter support status and value for fiscal printer getData method (Greece)

Description	dataitem	optArgs	Data	Supported?
	Identification data			

Table 78. Parameter support status and value for fiscal printer getData method (Greece) (continued)

Description	dataItem	optArgs	Data	Supported?
Get the Fiscal Printer's firmware release number.	FPTR_GD_FIRMWARE	Not used	Value is set by the method	Y
Get the Fiscal Printer's fiscal ID.	FPTR_GD_PRINTER_ID	Not used	Value is set by the method	Y
<hr/>				
Totals				
Get the current receipt total.	FPTR_GD_CURRENT_TOTAL	Not used	Value is set by the method	Y
Value is set by the method	FPTR_GD_DAILY_TOTAL	Not used	Value is set by the method	Y
Get the Fiscal Printer's grand total.	FPTR_GD_GRAND_TOTAL	Not used	Value is set by the method	Y
Get the total number of voided receipts.	FPTR_GD_MID_VOID	Not used	Value is set by the method	Y
Get the current total of not paid receipts.	FPTR_GD_NOT_PAID			N
Get the number of fiscal receipts printed.	FPTR_GD_RECEIPT_NUMBER	Not used	Value is set by the method	Y
Get the current total of refunds.	FPTR_GD_REFUND	Not used	Value is set by the method	Y
Get the current total of voided refunds.	FPTR_GD_REFUND_VOID	Not used	Value is set by the method	Y
<hr/>				
Fiscal memory counts				
Get the grand number of configuration blocks.	FPTR_GD_NUMB_CONFIG_BLOCK			N
Get the grand number of currency blocks.	FPTR_GD_NUMB_CURRENCY_BLOCK			N
Get the grand number of header blocks.	FPTR_GD_NUMB_HDR_BLOCK			N
Get the grand number of reset blocks.	FPTR_GD_NUMB_RESET_BLOCK			N
Get the grand number of VAT blocks.	FPTR_GD_NUMB_VAT_BLOCK			N
<hr/>				
Counter				
Get the number of daily fiscal documents.	FPTR_GD_FISCAL_DOC			N
Get the number of daily voided fiscal documents.	FPTR_GD_FISCAL_DOC_VOID			N
Get the number of daily fiscal sales receipts.	FPTR_GD_FISCAL_REC	Not used	Value is set by the method	Y
Get the number of daily voided fiscal sales receipts.	FPTR_GD_FISCAL_REC_VOID	Not used	Value is set by the method	Y
Get the number of daily non fiscal documents.	FPTR_GD_NONFISCAL_DOC			N
Get the number of daily voided non fiscal documents.	FPTR_GD_NONFISCAL_DOC_VOID			N
Get the number of daily non fiscal receipts.	FPTR_GD_NONFISCAL_REC			N
Get the Fiscal Printer's restart count	FPTR_GD_RESTART	Not used	Value is set by the method	Y
Get the number of daily simplified invoices.	FPTR_GD_SIMP_INVOICE			N
Get the Z report number.	FPTR_GD_Z_REPORT	Not used	Value is set by the method	Y
<hr/>				
Fixed fiscal printer text				

Table 78. Parameter support status and value for fiscal printer getData method (Greece) (continued)

Description	dataItem	optArgs	Data	Supported?
Get the payment description used in the printRecTotal method, defined by the given identifier in the optArgs argument. Valid only if the CapPredefinedPaymentLines property is true.	FPTR_GD_TENDER	FPTR_PDL_CASH FPTR_PDL_CHEQUE FPTR_PDL_CHITTY FPTR_PDL_COUPON FPTR_PDL_CURRENCY FPTR_PDL_DRIVEN_OFF FPTR_PDL_EFT_IMPRINTER FPTR_PDL_EFT_TERMINAL FPTR_PDL_TERMINAL_IMPRINTER FPTR_PDL_FREE_GIFT FPTR_PDL_GIRO FPTR_PDL_HOME FPTR_PDL_IMPRINTER_WITH_ISSUER FPTR_PDL_LOCAL_ACCOUNT FPTR_PDL_LOCAL_ACCOUNT_CARD FPTR_PDL_PAY_CARD FPTR_PDL_PAY_CARD_MANUAL FPTR_PDL_PREPAY FPTR_PDL_PUMP_TEST FPTR_PDL_SHORT_CREDIT FPTR_PDL_STAFF FPTR_PDL_VOUCHER		N
Get the number of printed lines, defined by the given identifier in the optArgs argument. If the CapMultiContractor property is true, line counters depend on the contractor defined by the ContractorId property.	FPTR_GD_LINECOUNT	FPTR_LC_ITEM FPTR_LC_ITEM_VOID FPTR_LC_DISCOUNT FPTR_LC_DISCOUNT_VOID FPTR_LC_SURCHARGE FPTR_LC_SURCHARGE_VOID FPTR_LC_REFUND FPTR_LC_REFUND_VOID FPTR_LC_SUBTOTAL_DISCOUNT FPTR_LC_SUBTOTAL_DISCOUNT_VOID FPTR_LC_SUBTOTAL_SURCHARGE FPTR_LC_SUBTOTAL_SURCHARGE_VOID FPTR_LC_COMMENT FPTR_LC_SUBTOTAL FPTR_LC_TOTAL		N
Get the maximum number of characters that may be passed as a description parameter for a specific method, defined by the ContractorId property.	FPTR_GD_DESCRIPTION_LENGTH	FPTR_DL_ITEM FPTR_DL_ITEM_ADJUSTMENT FPTR_DL_ITEM_FUEL FPTR_DL_ITEM_FUEL_VOID FPTR_DL_NOT_PAID FPTR_DL_PACKAGE_ADJUSTMENT FPTR_DL_REFUND FPTR_DL_REFUND_VOID FPTR_DL_SUBTOTAL_ADJUSTMENT FPTR_DL_TOTAL FPTR_DL_VOID FPTR_DL_VOID_ITEM	Value is set by the method	Y
Cut receipt	FPTR_GD_CUT_RECEIPT	Not used	Not used	Y
Set manual cut	FPTR_GD_MANUAL_CUT	0: set m_bManualCut to false 1: set m_bManualCut to true	Not used	Y

Table 78. Parameter support status and value for fiscal printer getData method (Greece) (continued)

Description	dataItem	optArgs	Data	Supported?
Get credit note number	FPTR_GD_CREDIT_NOTE_NUM	Not used	Value is set by the method	Y
Get credit note total	FPTR_GD_CREDIT_NOTE_TOT	Not used	Value is set by the method	Y
Get discounts	FPTR_GD_DISCOUNTS	Not used	Value is set by the method	Y

OPOS getTotalizer method (fiscal printer: Greece)

Table 79. Parameter support status and value for fiscal printer getTotalizer method (Greece)

optArgs		vatID	Data	Supported?
	TotalizerType			
FPTR_GT_GROSS	FPTR_TT_RECEIPT	1-5	Value is set by the method	Y
	FPTR_TT_DAY	1-5	Value is set by the method	Y
	FPTR_TT_GRAND	Not used	Value is set by the method	Y
FPTR_GT_NET	FPTR_TT_RECEIPT	Not used	Value is set by the method	Y
	FPTR_TT_DAY	Not used	Value is set by the method	Y
	FPTR_TT_GRAND	Not used	Value is set by the method	Y
FPTR_GT_DISCOUNT	FPTR_TT_RECEIPT	Not used	Value is set by the method	Y
	FPTR_TT_DAY	Not used	Value is set by the method	Y
	FPTR_TT_GRAND			N
FPTR_GT_DISCOUNT_VOID	FPTR_TT_RECEIPT	Not used	Value is set by the method	Y
	FPTR_TT_DAY	Not used	Value is set by the method	Y
	FPTR_TT_GRAND			N
FPTR_GT_ITEM	FPTR_TT_RECEIPT	Not used	Value is set by the method	Y
	FPTR_TT_DAY	Not used	Value is set by the method	Y
	FPTR_TT_GRAND			N
FPTR_GT_ITEM_VOID	FPTR_TT_RECEIPT	Not used	Value is set by the method	Y
	FPTR_TT_DAY	Not used	Value is set by the method	Y
	FPTR_TT_GRAND			N
FPTR_GT_NOT_PAID	FPTR_TT_RECEIPT			N
	FPTR_TT_DAY			N
	FPTR_TT_GRAND			N
FPTR_GT_REFUND	FPTR_TT_RECEIPT	Not used	Value is set by the method	Y
	FPTR_TT_DAY	Not used	Value is set by the method	Y
	FPTR_TT_GRAND			N

Table 79. Parameter support status and value for fiscal printer *getTotalizer* method (Greece) (continued)

optArgs		vatID	Data	Supported?
FPTR_GT_REFUND_VOID	FPTR_TT_RECEIPT	Not used	Value is set by the method	Y
	FPTR_TT_DAY	Not used	Value is set by the method	Y
	FPTR_TT_GRAND			N
FPTR_GT_SUBTOTAL_DISCOUNT	FPTR_TT_RECEIPT	Not used	Value is set by the method	Y
	FPTR_TT_DAY	Not used	Value is set by the method	Y
	FPTR_TT_GRAND			N
FPTR_GT_SUBTOTAL_DISCOUNT_VOID	FPTR_TT_RECEIPT	Not used	Value is set by the method	Y
	FPTR_TT_DAY	Not used	Value is set by the method	Y
	FPTR_TT_GRAND			N
FPTR_GT_SUBTOTAL_SURCHARGES	FPTR_TT_RECEIPT	Not used	Value is set by the method	Y
	FPTR_TT_DAY	Not used	Value is set by the method	Y
	FPTR_TT_GRAND			N
FPTR_GT_SUBTOTAL_SURCHARGES_VOID	FPTR_TT_RECEIPT	Not used	Value is set by the method	Y
	FPTR_TT_DAY	Not used	Value is set by the method	Y
	FPTR_TT_GRAND			N
FPTR_GT_SURCHARGE	FPTR_TT_RECEIPT	Not used	Value is set by the method	Y
	FPTR_TT_DAY	Not used	Value is set by the method	Y
	FPTR_TT_GRAND			N
FPTR_GT_SURCHARGE_VOID	FPTR_TT_RECEIPT	Not used	Value is set by the method	Y
	FPTR_TT_DAY	Not used	Value is set by the method	Y
	FPTR_TT_GRAND			N
FPTR_GT_VAT	FPTR_TT_RECEIPT	Not used	Value is set by the method	Y
	FPTR_TT_DAY	Not used	Value is set by the method	Y
	FPTR_TT_GRAND	Not used	Value is set by the method	Y
FPTR_GT_VAT_CATEGORY	FPTR_TT_RECEIPT	1-5	Value is set by the method	Y
	FPTR_TT_DAY			N
	FPTR_TT_GRAND	1-5	Value is set by the method	Y

Chapter 9. Hard totals

Hard totals supported devices

Table 80. Hard totals supported devices

Device	Connectivity
1. 4694 NVRAM	Integrated
2. SurePOS 700 Series NVRAM	Integrated
3. SureOne built-in NVRAM (excluding A04/A05 models)	Integrated
4. SurePOS 300 4810-35x	Integrated
5. TCx 300 Series 4810-36x/37x	Integrated
6. TCx 700 Series NVRAM	Integrated

Supported properties and methods

Table 81. Hard totals common properties

Property	JavaPOS	OPOS
AutoDisable		Not supported
BinaryConversion	Not supported	All
CapCompareFirmwareVersion		False
CapPowerReporting	PR_NONE	All support STANDARD
CapStatisticsReporting	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CapUpdateFirmware		False
CapUpdateStatistics	False (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CheckHealthText		All
Claimed		All
DataCount		Not supported
DeviceEventEnabled	Not supported	All
DeviceControlDescription		All
DeviceControlVersion		All
DeviceEnabled		All
DeviceServiceDescription		All
DeviceServiceVersion		All
FreezeEvents		All
OpenResult		All
OutputID	Not supported	All
PowerNotify	PN_DISABLED	All
PowerState	PS_UNKNOWN	All

Table 81. Hard totals common properties (continued)

Property	JavaPOS	OPOS
PhysicalDeviceDescription		All
PhysicalDeviceName		All
ResultCode		All
resultCodeExtended		All
State		All

Table 82. Hard totals specific properties

Property	JavaPOS	OPOS
CapErrorDetection		Not supported
CapSingleFile		All
CapTransactions		Not supported
FreeData		All
NumberOfFiles		Maximum of 1 file
TotalsSize		All
TransactionInProgress		Always false

Table 83. Hard totals common methods

Method	JavaPOS	OPOS
checkHealth		All
claim		All
clearInput		Not supported
clearOutput		Not supported
close		All
compareFirmwareVersion		Not supported
directIO		Not supported
open		All
release		All
resetStatistics		Not supported
retrieveStatistics	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	Not supported
updateFirmware		Not supported
updateStatistics		Not supported

Table 84. Hard totals specific methods

Method	JavaPOS	OPOS
beginTrans		Not supported
claimFile		All
commitTrans		Not supported
create		All
delete		All

Table 84. Hard totals specific methods (continued)

Method	JavaPOS	OPOS
find		All
findByIndex		All
releaseFile		All
read		All
recalculateValidationData		Not supported
rename		Not supported
rollback		Not supported
setAll		All
validateData		Not supported
write		All

Table 85. Hard totals events

Event	JavaPOS	OPOS
DirectIOEvent		Not supported
StatusUpdateEvent		All

OPOS configuration

Note: These settings cannot be modified with the Configuration tool.

Table 86. Service object settings for hard totals

Keyword	Type	Description
UseHarddiskFile	String	Enables use of file on hard drive for Hard Totals. Valid values are: True Use hard disk False Do not use hard disk (default)
HarddiskFileName	String	Path name of file to use for Hard Totals storage on hard disk.
HarddiskFileSize	String	Size (in bytes) of file used for Hard Totals Storage on hard disk.

Chapter 10. Keylock

Supported devices

Table 87. Keylock supported devices

Device		Connectivity
Two-position	1. Retail alphanumeric POS keyboard with card reader	PS/2, RS-485, USB
Two-position	2. Retail alphanumeric POS keyboard with MSR and pointing device	PS/2
Two-position	3. Retail POS keyboard	RS-485, USB
Two-position	4. Retail POS keyboard with card reader	RS-485, USB
Two-position	5. Retail POS keyboard with card reader and display	RS-485, USB
Two-position	6. Modifiable layout keyboard with card reader	RS-485, USB
Two-position	7. SurePoint 4820 integrated keylock	RS-485, USB
Three-position	8. SureOne integrated keyboard	PS/2
Four-position	9. POS keyboard V	RS-485, USB
Four-position	10. PC POS keyboard or PC Point of Sale keyboard	PS/2, RS-485, USB
Four-position	11. Retail POS keyboard VI	RS-485, USB
Three-position	12. SurePOS 100 integrated keyboard	PS/2
Five-position	13. 4674 Integrated keyboard	RS-485
Four-position	14. 4685-KC1 (OPOS only)	RS-485
Four-position	15. 4685-K01 (OPOS only)	RS-485
Four-position	16. 4685-K02 (Ultra7) keyboard with card reader	RS-485, USB
Four-position	17. 4685-K02 with MSR/Encoder and 4 position keylock	USB
Six-position	18. 4685-K02 with MSR/Encoder and 6 position keylock	USB
Four-position	19. 4685-K03	RS-485, USB
Two-position/ Four-position	20. Modular alphanumeric keyboard	PS/2, USB
Two-position/ Four-position	21. Modular Compact alphanumeric keyboard	PS/2, USB
Two-position/ Four-position	22. Modular 67 Key keyboard	PS/2, USB
Two-position	23. Modular 67 Key keyboard with Display and Keylock USB	PS/2, USB
Electronic	24. 6140-A3R Electronic Keylock	USB

The JavaPOS driver returns predetermined UnifiedPOS values for each keylock position. The OPOS driver generally allows the user to configure each key position to a different meaning.

For a two-position keylock, the OPOS driver can either return a predetermined UnifiedPOS value or the raw hardware value for each keylock position, which the application can then interpret as necessary.

For a three-position keylock, the OPOS driver returns predetermined UnifiedPOS values for each keylock position.

For four-, five-, and six-position keylocks, the OPOS driver allows you to map the raw hardware value to any UnifiedPOS value. This means that the driver can associate a physical key position with whatever UnifiedPOS value required.

For keylocks that contain more than the three positions that the UnifiedPOS standard provides, the drivers report those higher positions as a unique value. Table 88 describes the three standard positions in the UnifiedPOS standard.

Table 88. UnifiedPOS keylock standards

Position	Value returned
LOCK_KP_LOCK	1
LOCK_KP_NORM	2
LOCK_KP_SUPR	3

Two-position (devices 1, 2, 3, 4, 5, 6, 20, 21, 22)

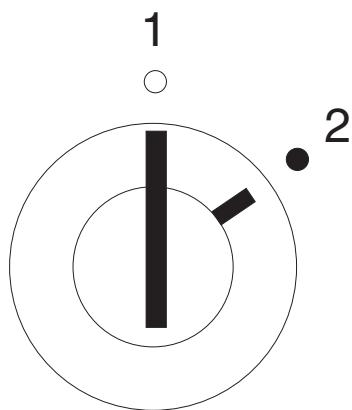


Figure 14. Two-position keylock

Position	JPOS value	OPOS mapped value	OPOS raw value
1	LOCK_KP_NORM	LOCK_KP_NORM	2
2	LOCK_KP_SUPR	LOCK_KP_SUPR	1

Two-position (device 7)

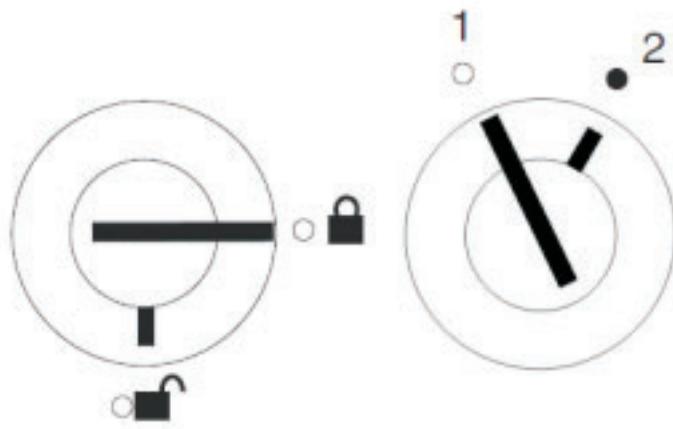


Figure 15. 4820 Two-position keylock

Position	JPOS value	OPOS mapped value	OPOS raw value
Lock (1)	LOCK_KP_NORM	LOCK_KP_NORM	2
Unlock (2)	LOCK_KP_SUPR	LOCK_KP_SUPR	1

Three-position keylock (devices 8, 12)

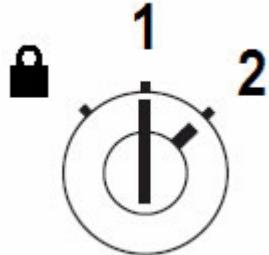


Figure 16. SureOne three-position keylock

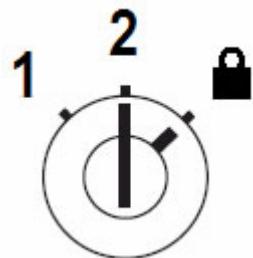


Figure 17. SurePOS 100 three-position keylock

Position	JPOS value	OPOS mapped value
LOCK	LOCK_KP_LOCK	LOCK_KP_LOCK

Position	JPOS value	OPOS mapped value
1	LOCK_KP_NORM	LOCK_KP_NORM
2	LOCK_KP_SUPR	LOCK_KP_SUPR

Four-position (devices 9, 10, 11, 14, 15, 16, 17, 19)

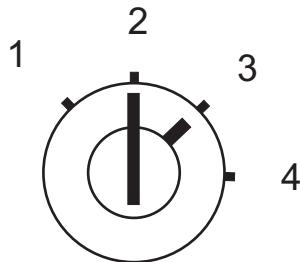


Figure 18. Four-position keylock

Position	JPOS value	OPOS mapped value	OPOS raw value
1	4	Configured by user	4
2	LOCK_KP_LOCK		3
3	LOCK_KP_NORM		2
4	LOCK_KP_SUPR		1

Four-position (devices 20, 21, 22)

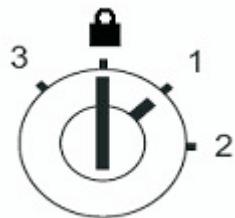


Figure 19. Four-position keylock

Position	JPOS value	OPOS mapped value	OPOS raw value
3	4	Configured by user	4
Lock	LOCK_KP_LOCK		3
1	LOCK_KP_NORM		2
2	LOCK_KP_SUPR		1

Five-position (device 13)

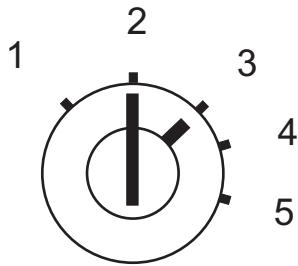


Figure 20. Five-position keylock

Position	JPOS value	OPOS mapped value	OPOS raw value
1	5	Configured by user	5
2	LOCK_KP_LOCK		3
3	LOCK_KP_NORM		2
4	LOCK_KP_SUPR		1
5	4		4

Six-position (device 18)

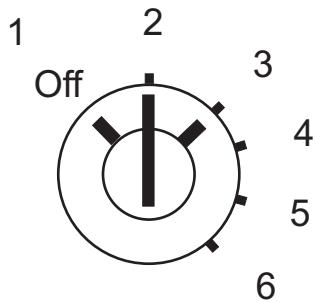


Figure 21. Six-position keylock

Position	JPOS value	OPOS mapped value	OPOS raw value
1	LOCK_KP_LOCK	Configured by user	3
2	LOCK_KP_NORM		2
3	5		5
4	6		6
5	LOCK_KP_SUPR		1
6	4		4

Electronic keylock (iButton)

The iButton is supported through the Keylock Device Category of the UnifiedPOS Standards. It behaves just like keylock, except the application reads keylock data from ElectronicKeyValue instead of the KeyPosition property.

The Electronic Keylock supports the following properties:

CapKeylockType:

Set to LOCK_KT_ELECTRONIC

ElectronicKeyValue:

Contains 8 bytes of hexadecimal data of the iButton

- When iButton is connected: XX XX XX XX XX XX XX XX
- When iButton is disconnected: 00 00 00 00 00 00 00 00

KeyPosition:

This property has no meaning for Electronic Keylock

Methods associated:

`waitForKeylockChange()`

StatusUpdateEvent:

`LOCK_KP_ELECTRONIC` event is fired when iButton is connected and disconnected

Configuration:

- OPOS: Use the OPOS Configuration Tool to configure an iButton device under the Keylock category.
- JavaPOS: Use the POS Control Center to generate new JposEntry (`jpos.xml`) for iButton.

Supported properties and methods

Table 89. Keylock common properties

Property	JavaPOS	OPOS
AutoDisable	Not supported	
BinaryConversion	Not supported	All
CapCompareFirmwareVersion		False
CapPowerReporting	PR_NONE for EIA-232, PS/2 and embedded PR_STANDARD for all other keylock devices	PR_STANDARD
CapStatisticsReporting	True (see Appendix A, “JavaPOS support for UnifiedPOS device statistics properties,” on page 431)	False
CapUpdateFirmware		False
CapUpdateStatistics	False (see Appendix A, “JavaPOS support for UnifiedPOS device statistics properties,” on page 431)	False
CheckHealthText		All
Claimed		All
DataCount		Not supported
DataEventEnabled		Not supported
DeviceControlDescription		All
DeviceControlVersion		All
DeviceEnabled		All
DeviceServiceDescription		All
DeviceServiceVersion		All
FreezeEvents		All

Table 89. Keylock common properties (continued)

Property	JavaPOS	OPOS
OpenResult		Not supported
OutputID		Not supported
PowerNotify		All
PowerState		All
PhysicalDeviceDescription		All
PhysicalDeviceName		All
State		All

Table 90. Keylock specific properties

Property	JavaPOS	OPOS
CapKeylockType		All
ElectronicKeyValue		24
KeyPosition		All
PositionCount		All

Table 91. Keylock common methods

Method	JavaPOS	OPOS
checkHealth		All
claim		Always shareable
clearInput		Not supported
clearOutput		Not supported
close		All
compareFirmwareVersion		Not supported
directIO		Not supported
open		All
release		Always shareable
resetStatistics		Not supported
retrieveStatistics	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	Not supported
updateFirmware		Not supported
updateStatistics		Not supported

Table 92. Keylock specific methods

Method	JavaPOS	OPOS
waitForKeylockChange		All

Table 93. Keylock events

Event	JavaPOS	OPOS
DirectIOEvent		Not supported

Table 93. Keylock events (continued)

Event	JavaPOS	OPOS
StatusUpdateEvent		All

JavaPOS configuration

PositionCount

```
<prop name="com.ibm.jpos.sdi.config.Keylock.PositionCount" type="String" value="4"/>
```

For some Toshiba keylocks, the number of keylock positions cannot be reliably determined:

- RS-485 4685-K03: four positions
- RS-485 4685-K02 Keyboard with MSR Encoder: six positions

In these situations, use the PositionCount property to define the number of keylock positions.

Values accepted

Any value on Integer range

Device support

All

OPOS configuration

Table 94. Service Object settings for keylock

Keyword	Type	Description
SecondKeyPosition	String	When present, the additional position on the Japanese ANPOS keyboard is mapped to a LOCK_KP_LOCK.
PositionCount	DWORD	4685-K02 only. Number of keylock positions (1-6).
PositionMapFrom	Binary	4685-K02 only. Keylock position code mapping (from).
PositionMapTo	Binary	4685-K02 only. Keylock position code mapping (to).
LetLastKeepAcquired	String	Allows the last interface to access the physical POS keyboard interface to keep the device acquired instead of passing it on to the next device that attempts to acquire it. Note: This setting is an <i>unsupported and untested</i> option used to modify Service Object behavior only in very limited cases. Use it only when directed to do so by Toshiba Support.
OnlineTimeout	String	Time in milliseconds to wait for device to come online. Note: This setting cannot be modified with the Configuration tool.

Additional OPOS information

Two-position keylock

The keylocks for the NANPOS, 50-Key, 133-Key, and 4820 Point of Sale Keyboards all have two positions and report them as LOCK_KP_NORM and either LOCK_KP_SUPR or LOCK_KP_LOCK.

In OPOS, two-position keylocks can report the key positions as LOCK_KP_NORM and either LOCK_KP_SUPR or LOCK_KP_LOCK. In the OPOS Configuration Utility:

- Select the **Toshiba Raw Values** radio button during configuration to cause the manager position on the keyboard to return LOCK_KP_LOCK.
- Select the **Map to OPOS Values** radio button to cause the manager position to return LOCK_KP_SUPR.

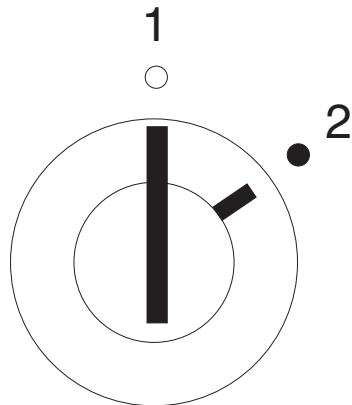


Figure 22. Two-position keylock

Key position	Description	Key position value
1	Operator	LOCK_KP_NORM
2	Manager	LOCK_KP_SUPR LOCK_KP_LOCK (OPOS only)

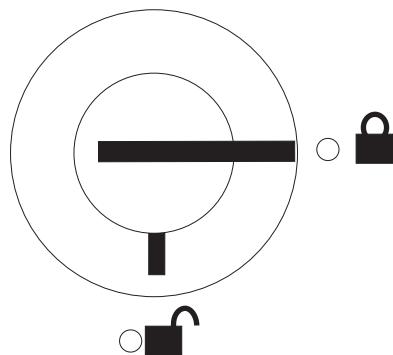


Figure 23. 4820 Two-position keylock

Key position	Description	Key position value
Lock	Operator	LOCK_KP_NORM
Unlock	Manager	LOCK_KP_SUPR LOCK_KP_LOCK (OPOS only)

Position count

For some Toshiba keylocks, the number of keylock positions cannot be reliably determined by OPOS:

- RS-485 4685-K03: four positions

- RS-485 4685-K02 Keyboard with MSR Encoder: six positions

For this reason, a mechanism is provided to define the mapping of the native keylock positions to OPOS values. During configuration of the keylock device, the number of positions and the mapping of the positions can be set. Use the three-six position keyboard keylock for keylocks with three to six positions.

In the OPOS Configuration Utility, use the Keylock entry labeled "Three-Six position keyboard keylock" or "Three-six position keyboard keylock - USB" for keylocks with three to six positions.

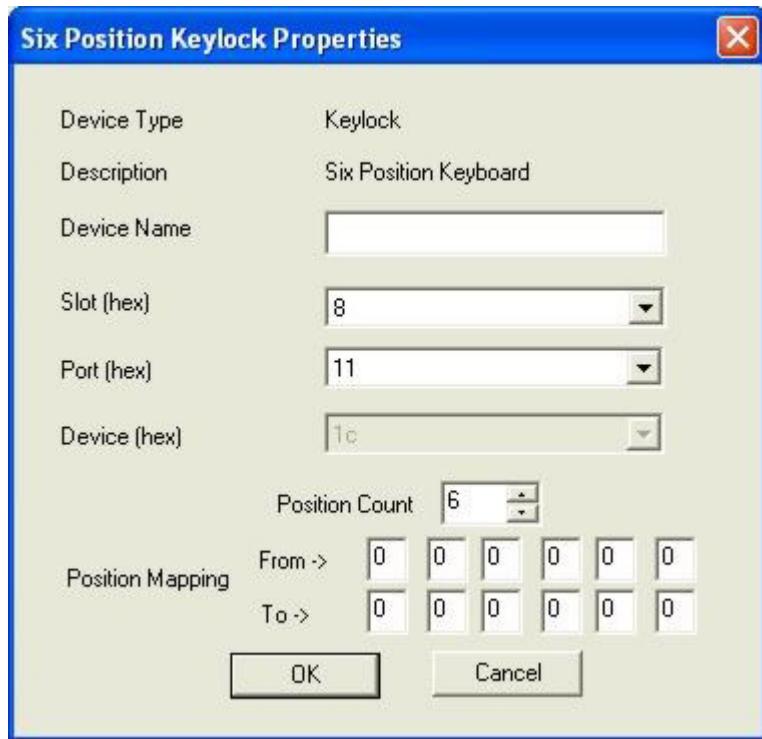


Figure 24. Three-six position keylock properties

Chapter 11. Line display

Supported devices

Table 95. Line display supported devices

Device		Connectivity
SBCS	1. 40-character VFD II/SureOne	EIA-232
	2. 40-character VFD II	USB, RS-485
	3. Two-sided VFD II	USB, RS-485
	4. 40-character VFD II (Japan)/4674	RS-485
	5. Character Graphic (C/G) APA	USB, RS-485
	6. 40-character LCD	USB, RS-485
	7. 50-key keyboard LCD	USB, RS-485
DBCS	8. Character Graphic (C/G) APA	EIA-232
	9. Character Graphic (C/G) APA	USB, RS-485
	10. PLU keyboard APA	USB, RS-485
SBCS	11. LED display 1x11	EIA-232
SBCS	12. Modular 67-Key Keyboard w/Card Reader and Display	USB
SBCS	13. TCxWave 6140 Integrated 40-character LCD	USB
SBCS	14. TCx 2x20 Single-sided LCD	USB
SBCS	15. TCx 2x20 Double-sided LCD	USB
DBCS	16. TCx APA Single-sided	USB

Note: On the SurePOS 500/600, the line display is preconfigured for COM4. Other values can be selected through the BIOS, but this is not recommended.

Supported properties and methods

Table 96. Line display common properties

Property	JavaPOS	OPOS
AutoDisable		Not supported
BinaryConversion	Not supported	All
CapCompareFirmwareVersion		False
CapPowerReporting	PR_NONE for 1,8,11 PR_STANDARD for all other LineDisplay	PR_NONE for 11 PR_STANDARD otherwise
CapStatisticsReporting	True (see Appendix A, “JavaPOS support for UnifiedPOS device statistics properties,” on page 431)	False
CapUpdateFirmware		False
CapUpdateStatistics	False (see Appendix A, “JavaPOS support for UnifiedPOS device statistics properties,” on page 431)	False

Table 96. Line display common properties (continued)

Property	JavaPOS	OPOS
CheckHealthText		All
Claimed		All
DataCount		Not supported
DataEventEnabled		Not supported
DeviceControlDescription		All
DeviceControlVersion		All
DeviceEnabled		All
DeviceServiceDescription		All
DeviceServiceVersion		All
FreezeEvents		All
OpenResult	Not supported	All
OutputID	Not supported	Not supported
PowerNotify	PN_DISABLED	OPOS_PN_DISABLED for 11, All for others
PowerState		All
PhysicalDeviceDescription		All
PhysicalDeviceName		All
ResultCode	Not supported	All
ResultCodeExtended	Not supported	All
State		All

Table 97. Line display specific properties

Property	JavaPOS	OPOS
BlinkRate		Not supported
CapBlink	Not supported except 10 (set CB_BLINKALL)	Not supported except 8 (set CB_BLINKALL)
CapBitmap		Not supported
CapBlinkRate		Not supported
CapBrightness		Not supported except 1 and 10 (Supported on EIA-232 integrated and external line displays)
CapCharacterSet	See “Character sets supported in JavaPOS” on page 124 and “Character sets supported in OPOS” on page 128	All (values based on HW)
CapCursorType	See Table 101 on page 118.	Not supported
CapCustomGlyph	1, 2, 3, 6, 7, 8, 12, 13, 14, 15	Not supported

Table 97. Line display specific properties (continued)

Property	JavaPOS	OPOS
CapDescriptors	True for: 2, 3, 4, 5, 9, 10, 16 USB 2x20 VFD External LineDisplay USB 2x20 VFD Integrated LineDisplay RS-485 2x20 VFD External LineDisplay USB Character/Graphic APA LineDisplay RS-485 APA LineDisplay False for the remaining of the LineDisplay devices	All: set true 1, 10, 11, 12, 13: set false
CapHMarquee		Not supported
CapICharWait		Not supported
CapMapCharacterSet	Not supported	1
CapReadBack		Not supported
CapReverse		Not supported
CapScreenMode	8	Not supported
CapVMarquee		Not supported
CharacterSet		All (values based on HW)
CharacterSetList		See “Character sets supported in JavaPOS” on page 124
CurrentWindow		All
Columns		All
CursorColumn		All
CursorRow		All
CursorType	See Table 101 on page 118. Not supported directly as property, but supported using DirectIO method for EIA-232 external LineDisplay (All-points addressable [APA]).	Not supported
CursorUpdate		All

Table 97. Line display specific properties (continued)

Property	JavaPOS	OPOS
CustomGlyphList	<p>1 "0015-001A,001C-001E" 2, 3, 6, 7, 12, 13, 14, 15 0000-0007 8 ASCII returns: "0020-00FF" English(437) returns: "0001- 0007,0009,000B,000C,000E- 001A,001C-00FF" Japanese(932) returns: "8140-81FF,8240-84FF,8840- 88FF,8940-9FFF,E040- E0FF,E140-EAFF" Korean(1361) returns: "A1A0-A1FF,A2A0- ACFF,B0A0-B0FF,B1A0- C8FF,CAA0-CAFF,CBA0- FDFF" Simplified Chinese(936) returns: "A1A0-A1FF,A2A0- A9FF,B0A0-B0FF,B1A0-F7FF" Traditional Chinese(950) returns: "A1A0-A1FF,A240- C6FF,C940-C9FF,CA40-F9FF"</p>	Not supported
CustomDSGlyphList		Not supported
CustomSSGlyphList		Not supported
DeviceBrightness	Only 1	0 or 100% only except on 1 and 10 (0 to 100%)
DeviceColumns		All
DeviceDescriptors	2,3,4,5,9,10,16 '20	All
DeviceRows		All
DeviceWindows	Not supported	All
GlyphHeight	<p>1, 2, 3 7 6, 7, 12, 13, 14, 15 8 8 16 (2x20), 7 (4x20), 7 (5x20) 9 SBCS: 16 (2x20), DBCS: 16 (2x20)</p>	Not supported

Table 97. Line display specific properties (continued)

Property	JavaPOS	OPOS
GlyphWidth	1, 2, 3, 6, 7, 12, 13, 14, 15 5 8 16 (2x20), 8(2x20), 5 (4x20), 5 (5x20) 9 SBCS: 8 (2x20), DBCS: 16 (2x20)	Not supported
InterCharacterWait		Not supported
MapCharacterSet	Not supported	1
MarqueeFormat		Not supported
MarqueeRepeatWait		Not supported
MarqueeType		Not supported
MarqueeUnitWait		Not supported
MaximumX		Not supported
MaximumY		Not supported
Rows		All
ScreenMode	All (Except 8) 0 8 0, 1, 2	Not supported
ScreenModeList	All (Except 8) "2x20" 8 "2x20,4x20,5x20"	Not supported

Table 98. Line display common methods

Method	JavaPOS	OPOS
checkHealth		All
claim		All
clearInput		Not supported
clearOutput		Not supported
close		All
compareFirmwareVersion		Not supported
directIO	5, 8, 9, 10 command 1-ScreenMode, pData Value = (0-2x20, 1-3x32)	8 Command 1 = ScreenMode, pData Value = (0-2x20, 1-3x32) 10 Command 0 = ScreenMode, pDataValue = (2-2x20, 4-4x20, 5-5x20)
open	9 command 0 = ScreenMode pData Value = (2-2x20,4-4x20,5-5x20)	All
release		All
resetStatistics		Not supported

Table 98. Line display common methods (continued)

Method	JavaPOS	OPOS
retrieveStatistics	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	Not supported
updateFirmware		Not supported
updateStatistics		Not supported

Table 99. Line display specific methods

Method	JavaPOS	OPOS
clearDescriptors	Supported for USB-VFD LineDisplay	All except 1, 10, 12, and 13
clearText		All
createWindow	Not supported	All
defineGlyph	1, 2, 3, 6, 7, 8, 12, 13, 14, 15 Note: glyph parameter is a byte array with length as follows: 1, 2, 3 7 6, 7, 12, 13, 14, 15 8 8 32 (2x20), 16 (2x20), 7 (4x20), 7 (5x20)	Not supported
destroyWindow	Not supported	All
displayBitmap		Not supported
displayText		All
displayTextAt		All
readCharacterAtCursor		Not supported
refreshWindow	Not supported	All
scrollText		All
setBitmap		Not supported
setDescriptor	Supported only for USB-VFD LineDisplay	All except 1, 10, 12, and 13

Table 100. Line display events

Event	JavaPOS	OPOS
DirectIOEvent		All
StatusUpdateEvent		All

Table 101. CapCursor Type and Cursor Type Properties

Cursor type	2, 3, 4, 5, 6 (RS-485), 7 (RS-485), 9, 10, 11, 14, 15, 16 (USB)	6 (USB), 7 (USB), 12, 13	1	8
capCursor Type	DISP_CCT_NONE	DISP_CCT_UNDERLINE +DISP_CCT_FIXED	DISP_CCT_UNDERLINE	DISP_CCT_UNDERLINE ¹
cursorType	DISP_CT_NONE	DISP_CT_UNDERLINE	DISP_CT_UNDERLINE, DISP_CT_NONE	DISP_CT_UNDERLINE ¹ DISP_CT_NONE

¹The 2x20 screen mode does not have a cursor.

Supported properties and methods (line display attached fiscal printer: Italy)

Table 102. Common properties (line display attached fiscal printer: Italy)

Property	JavaPOS	OPOS
AutoDisable		Not supported
CapCompareFirmwareVersion		False
CapPowerReporting		PR_NONE
CapStatisticsReporting	False	Not supported
CapUpdateFirmware		False
CapUpdateStatistics		False
CheckHealthText		String
Claimed		True/False
DataCount		Not supported
DatEventEnabled		Not supported
DeviceEnabled		True/False
DeviceControlDescription		String
DeviceControlVersion		int
DeviceServiceDescription		String
DeviceServiceVersion		Supported
FreezeEvents		False
OutputID		E_ILLEGAL
PhysicalDeviceDescription		String
PhysicalDeviceName		String
PowerNotify		PN_DISABLED
PowerState		PS_UNKNOWN
State		int

Table 103. Specific properties (line display attached fiscal printer: Italy)

Property	JavaPOS	OPOS
CapBlink		DISP_CB_NOBLINK
CapBitmap		False
CapBlinkRate		False
CapBrightness		False
CapCharacterSet		DISP_CCS_ASCII
CapCursorType		DISP_CCT_NONE
CapCustomGlyph		False
CapDescriptors		False
CapHMarquee		False
CapICharWait		False
CapMapCharacterSet		False
CapReadBack		DISP_CRB_NONE

Table 103. Specific properties (line display attached fiscal printer: Italy) (continued)

Property	JavaPOS	OPOS
CapReverse		DISP_CR_NONE
CapScreenMode		False
CapVMarquee		False
BlinkRate		E_ILLEGAL
CharacterSet		DISP_CS_ASCII
CharacterSetList		"998" (String)
Columns		20 (int)
CurrentWindow		E_ILLEGAL
CursorColumn		0-19 (int)
CursorPosition		0-1 (int)
CursorType		DISP_CT_NONE
CursorUpdate		False
CustomGlyphList		E_ILLEGAL
DeviceBrightness		E_ILLEGAL
DeviceColumns		20 (int)
DeviceDescriptors		0 (int)
DeviceRows		2 (int)
DeviceWindows		0 (int)
GlyphHeigh		E_ILLEGAL
GlyphWidth		E_ILLEGAL
InterCharacterWait		E_ILLEGAL
MapCharacterSet		False
MarqueeFormat		E_ILLEGAL
MarqueeRepeatWait		E_ILLEGAL
MarqueeType		DISP_MT_NONE
MarqueeUnitWait		E_ILLEGAL
MaximumX		0 (int)
Rows		2 (int)
ScreenMode		0 (int)
ScreenModeList		"2x20" (String)

Table 104. Common methods (line display attached fiscal printer: Italy)

Method	JavaPOS	OPOS
checkHealth		Supported
claim		Supported
clearInput		Not supported
clearOutput		E_ILLEGAL
close		Supported
compareFirmwareVersion		E_ILLEGAL
directIO		Supported

Table 104. Common methods (line display attached fiscal printer: Italy) (continued)

Method	JavaPOS	OPOS
open		Supported
release		Supported
resetStatistics		E_ILLEGAL
retrieveStatistics		E_ILLEGAL
updateFirmware		E_ILLEGAL
updateStatistics		E_ILLEGAL

Table 105. Specific methods (line display attached fiscal printer: Italy)

Method	JavaPOS	OPOS
clearText		Supported
displayText		Supported
displayTextAt		Supported
scrollText		E_ILLEGAL
clearDescriptors		E_ILLEGAL
setDescriptor		E_ILLEGAL
createWindow		E_ILLEGAL
destroyWindow		E_ILLEGAL
refreshWindow		E_ILLEGAL
defineGlyph		E_ILLEGAL
readCharacterAtCursor		E_ILLEGAL
displayBitmap		E_ILLEGAL
setBitmap		E_ILLEGAL

JavaPOS configuration

CharacterSetASCIIBehavior

The behavior of Character Set ASCII (998) is undefined for characters beyond 127 (0x7F). To make these characters behave as some other character set in the CharacterSetList, specify it using this property. To keep these characters working as in previous JPOS releases, specify 998. If the property does not exist, the default is 998.

```
=<com.ibm.jpos.services.sdi.config.LineDisplay.CharacterSetASCIIBehavior type="String" value="998"/>
```

CursorPosition

```
<prop name="com.ibm.jpos.services.sdi.config.LineDisplay.CursorState" type="String" value="ON"/>
```

Initializes the EIA-232 LineDisplay VFD cursor state to On or Off. This is done only at initialization time.

Valid values: ON or OFF

lineDisplayId

```
<prop name="com.ibm.posj.bus.rs232.lineDisplayId" type="String" value="APA"/>
```

For EIA-232 displays, this property is used to distinguish between All-Points Addressable (APA) and Vacuum Fluorescent Displays (VFD).

Valid values: APA, VFD, or LED

NumberOfRows

```
<prop name="com.ibm.jpos.services.sdi.config.LineDisplay.NumberOfRows" type="String" value="2"/>
```

Initializes the EIA-232 APA line display to a selected number of rows.

Valid values: 2, 4, or 5

JavaPOS DirectIO calls

The DirectIO commands in this section are supported for the LineDisplay. The following is the correct syntax:

```
Syntax directIO (command: int32, inout data: int32, inout obj: object) :  
    void {raises-exception}
```

To access line display constants, import: com.ibm.jpos.services.DirectIO.

Select cursor type command

Set display cursor type. This command applies to EIA-232 displays.

Table 106. Parameters for DirectIO.SELECT_APACHE_CURSOR_TYPE

Parameter	Type	Value
Command	Int32	DirectIO.SELECT_APACHE_CURSOR_TYPE
Data [0]	Int32	4 = DISP_CT_UNDERLINE, 1 = DISP_CT_NONE
Obj	Object	Always null

Select number of rows command

Set number of display rows. This command applies to EIA-232 APA (all points addressable) character graphics displays.

This command can also be used with ScreenMode property.

Table 107. Parameters for DirectIO.SELECT_APACHE_NUMBER_OF_ROWS

Parameter	Type	Value
Command	Int32	DirectIO.SELECT_APACHE_NUMBER_OF_ROWS
Data [0]	Int32	2 = 2 rows by 20 columns, 4 = 4 rows by 20 columns, 5 = 5 rows by 20 columns
Obj	Object	Always null

Select rotate text 180 command

Set display test mode. This command applies to USB APA Vacuum Fluorescent Display (VFD).

Table 108. Parameters for the DirectIO.SELECT_ROTATE_TEXT_180

Parameter	Type	Value
Command	Int32	DirectIO.SELECT_ROTATE_TEXT_180

Table 108. Parameters for the DirectIO.SELECT_ROTATE_TEXT_180 (continued)

Parameter	Type	Value
Data [0]	Int32	0 = display text in normal mode, 1 = display text mode 180 degrees rotated
Obj	Object	Always null

Errors A UposExceptionException may be thrown when this method is invoked. Some possible values of the exception ErrorCode properties are:

Table 109. ErrorCode property for DirectIO.SELECT_ROTATE_TEXT_180

Value	Possible causes
E_ILLEGAL	<ul style="list-style-type: none">cmd name /value is not known or is not valid.Some of the parameter values are not correct.

Additional JavaPOS information

Descriptors

Most models of Toshiba vacuum fluorescent and character graphics displays have the ability to display descriptors. The following displays have 20 descriptors:

- USB 2x20 VFD External LineDisplay
- USB 2x20 VFD Integrated LineDisplay
- RS-485 2x20 VFD External LineDisplay
- All descriptors are on one row, below the bottom line of text

The following displays have 24 descriptors:

- USB Character/Graphic APA LineDisplay
- RS-485 APA LineDisplay
- There are two rows of descriptors, 12 above the top row of text and 12 below the bottom row of text

In prior versions of JavaPOS, the descriptors were numbered incrementally from one, with position one being the top leftmost position. In JavaPOS 1.9.6, the descriptors have been renumbered to be consistent with OPOS. The descriptors are still numbered incrementally from one, but the starting position is now the lowest rightmost position.

Capability definitions

DISP_CCT_NONE

Cursor is not displayable.

DISP_CCT_UNDERLINE

Cursor is displayable as an underline. Cursor can be turned on and off.

DISP_CCT_UNDERLINE+DISP_CCT_FIXED

Cursor is displayable as an underline. Cursor is always displayed.

Property definitions

DISP_CT_NONE

Cursor is not displayed.

DISP_CT_UNDERLINE

Cursor is displayed as an underline.

Note: Another value received in the device will cause a JPOS_E_ILLEGAL at setCursorType()

Character sets supported in JavaPOS

Table 110 shows the country character sets supported by the line displays in JavaPOS. The US/European character set contains at least the upper case characters from many of the previous code pages (excluding Cyrillic). These are duplicated in the code pages for those countries.

Table 110. Character sets supported by line display in JavaPOS

Code page	LineDisplay device											
	1 EIA-232	2,3,4 RS485	2,3 USB	5,9,10 RS485	5,9,10 USB	6,7 RS485	6,7 USB	8 EIA232	11 EIA232	12, 13, 14, 15 USB	16 USB	
Single-byte character set												
101 (Mini-Alphanumeric)									✓			
437 U.S. English	✓	✓	✓		✓			✓		✓	✓	
775										✓	✓	
808 (Cyrillic-Russia)	✓	✓								✓	✓	
			Note 1 on page 125									
819 (Alphanumeric and special)				✓								
				Note 1 on page 125								
848										✓	✓	
852 (Central Europe or Hungary)	✓	✓	✓		✓					✓	✓	
				Note 1 on page 125								
855 (Cyrillic)	✓	✓	✓		✓					✓	✓	
				Note 1 on page 125								
857 (Turkey)	✓	✓	✓		✓					✓	✓	
				Note 1 on page 125								
858 (Multilingual)	✓	✓	✓		✓					✓	✓	
				Note 1 on page 125								
860										✓	✓	
861										✓	✓	
862 (Israel)	✓	✓	✓		✓					✓	✓	
				Note 1 on page 125								
863 (Canada-French)	✓	✓	✓		✓					✓	✓	
				Note 1 on page 125								
864 (Arabic)	✓	✓	✓		✓					✓	✓	
				Note 1 on page 125								
865 (Nordic)	✓	✓	✓		✓					✓	✓	
				Note 1 on page 125								
866 (Russia)		✓	✓		✓							
				Note 1 on page 125								
867										✓	✓	

Table 110. Character sets supported by line display in JavaPOS (continued)

Code page	LineDisplay device										
	1 EIA-232	2,3,4 RS485	2,3 USB	5,9,10 RS485	5,9,10 USB	6,7 RS485	6,7 USB	8 EIA232	11 EIA232	12, 13, 14, 15 USB	16 USB
869 (Greek)	✓	✓	✓		✓					✓	✓
		Note 1									
872					✓					✓	✓
897 (Katakana)	✓	✓	✓		✓					✓	✓
998 US/European (based on CP437)	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓
1250										✓	✓
1251										✓	✓
1252						✓				✓	✓
1253										✓	✓
1254										✓	✓
1255										✓	✓
1256										✓	✓
1257										✓	✓
Double-byte character set											
926 (Hangeul and Alphanumeric)				✓							
932 (Japanese)				✓	✓			✓			✓
				Note 2	Note 3						Note 3
936 (Simplified Chinese)					✓			✓			✓
					Note3						Note 3
949 (Korean Wansung)				✓	✓						✓
				Note 2	Note 3						Note 3
950 (Traditional Chinese)				✓	✓			✓			✓
				Note 2	Note 3						Note 3
1361 (Korean Johab)								✓			

Note:

- Only available if EC Level >= 16 (0x10)
- Only available if corresponding font is in device ROM
- Only available if corresponding font was downloaded

USB APA font download

The following information is necessary to download fonts to the USB Character/Graphics Display:

- The command line utility AipFntVfd, provided to download fonts to the Line Display.
- Font files for the USB Character/Graphics Display:
 - Windows file location: <install dir>\fonts
 - Linux file location: /usr/share/pos/fonts
- Only one font file can be applied at a time.

Table 111. Font File Summary

Font File Name	Font File Description
jpnm16.bin	932 (Japanese)
korm16.bin	949 (Korean Wansung)
sbccs0001.bin	437 U.S. English

Table 111. Font File Summary (continued)

Font File Name	Font File Description
prcmz16.bin	936 (Simplified Chinese)
rocbg16b.bin	950 (Traditional Chinese)

Syntax for the utility:

```
java com.tgcs.jpos.util.font.AipFntVfd [filename] [logicalname]
```

Where:

filename: The fully qualified pathname of the font file.

logicalname: The name specified in the JposEntry in the jpos.xml for that device

Example:

```
java com.tgcs.jpos.util.font.AipFntVfd /usr/share/pos/fonts/jpnm16.bin LineDisplay1  
java com.tgcs.jpos.util.font.AipFntVfd C:\POS\fonts\jpnm16.bin LineDisplay1
```

OPOS configuration

Note: These settings cannot be modified with the Configuration tool.

Table 112. Service Object settings for RS-485/USB line display

Keyword	Type	Description
DefineCharacter	Key	Key values for user-defined characters.
DefineCharacter\<X>	String	Hexadecimal value of user-defined character to be downloaded. Eight bytes of data for character definition. Each data byte is represented by 2 ASCII characters (e.g., the value 0x69969960 is the string 0609090609090600). For more information on the character format, refer to <i>Toshiba Point of Sale Subsystem Programming Reference and User's Guide</i> (SC30-3560).

Table 113. Service object settings for EIA-232 line display

Keyword	Type	Description
Type	String	Line display type. Valid values are: LINE Single-byte display APA Double-byte display
Mode	String	Screen display mode. Valid values are 2x20 (default), 4x20, and 5x20.

OPOS DirectIO calls

Set number of APA rows

Sets the number of rows on EIA-232 USB and RS-485 all points addressable (APA) line displays.

Table 114. Set number of APA rows

Parameter	Type	Description
Command	Int32	0

Table 114. Set number of APA rows (continued)

Parameter	Type	Description
Data	Long*	<p>Pointer to a byte which determines the line display size.</p> <p>Valid values are:</p> <ul style="list-style-type: none"> 2 (2x20) 3 (3x32) 4 (4x20) 5 (5x20)
Obj	BSTR*	Ignored

Additional OPOS information

USB APA font download

To download fonts to the USB Character/Graphics Display, you will need the following:

- The display font download program, AIPFNVD.EXE, which comes with the Toshiba Point of Sale Subsystem for Windows.
- Font files for the USB Character/Graphics Display, which are found in the Toshiba Point of Sale Subsystem for Windows FONTS directory.

Font File Name	Font File Description
jpnm16.bin	Japan, Mincho style
korm16.bin	Korea, Mincho style
sbc0001.bin	SBCS fonts, updated version
Prcmz16.bin	PRC Simplified Chinese
Rocbg16b.bin	Taiwan Traditional Chinese

Font files for the USB Character/Graphics Display must be downloaded manually using the AIPFNVD.EXE program.

AIPFNVD -Sslot -Pport {-Ndevice-number} -Fx:\dir\font-file {-Q}

-Sslot The slot number (in decimal) of the display for which the download is intended.

-Pport The port number (in decimal) of the display for which the download is intended.

-Ndevice-number

An indication of the device number of the display for which the download is intended. Valid values are:

- A : device number 2A (default)
- B : device number 2B
- C : device number 2C
- D : device number 2D

-Fx:\dir\font-file

Fully qualified path name for the font file to be downloaded to the display.

-Q Quiet mode. No progress information is displayed

Character sets supported in OPOS

Table 115 shows the country character sets supported by the line displays in OPOS. The US/European character set contains at least the upper case characters from many of the previous code pages (excluding Cyrillic). These are duplicated in the code pages for those countries.

Table 115. Character sets supported by line display in OPOS

Code page	LineDisplay device										
	1 EIA-232	2,3,4 RS485	2,3 USB	5,9,10 RS485	5,9,10 USB	6,7 RS485	6,7 USB	8 EIA232	11 EIA232	12, 13, 14, 15	16 USB
Single-byte character set											
101 (Mini-Alphanumeric)											
437 U.S. English	✓					✓	✓			✓	✓
775											✓
808 (Cyrillic-Russia)											✓
819 (Alphanumeric and special)											
848											✓
852 (Central Europe or Hungary)	✓	✓	✓		✓	✓	✓			✓	✓
855 (Cyrillic)	✓	✓	✓		✓	✓	✓			✓	✓
857 (Turkey)	✓	✓	✓		✓	✓	✓			✓	✓
858 (Multilingual)	✓	✓	✓		✓	✓	✓				✓
860											✓
861											✓
862 (Israel)	✓	✓	✓		✓	✓	✓			✓	✓
863 (Canada-French)	✓	✓	✓		✓	✓	✓			✓	✓
864 (Arabic)	✓	✓	✓		✓	✓	✓			✓	✓
865 (Nordic)	✓	✓	✓		✓	✓	✓			✓	✓
866 (Russia)		✓	✓		✓	✓	✓			✓	
867											✓
869 (Greek)	✓	✓	✓		✓	✓	✓			✓	✓
872											✓
897 (Katakana)	✓	✓	✓		✓						✓
998 US/European (based on CP437)	✓	✓	✓		✓	✓	✓	✓		✓	✓
999			✓			✓	✓				
1250											✓
1251											✓
1252			✓			✓	✓			✓	✓
1253											✓
1254											✓
1255											✓
1256											✓
1257											✓
Double-byte character set											
926 (Hangeul and Alphanumeric)											
932 (Japanese)					✓			✓			✓
936 (Simplified Chinese)											✓
949 (Korean Wansung)					✓						✓
950 (Traditional Chinese)											✓
1361 (Korean Johab)											

Chapter 12. Magnetic Ink Character Recognition

Supported devices

Table 116. Magnetic Ink Character Recognition (MICR) supported devices

Device	Connectivity
4610 Printer built-in MICR	EIA-232, RS-485, USB, Network
Model 4 printer (OPOS only)	RS-485

Supported properties and methods

Table 117. Magnetic Ink Character Recognition (MICR) common properties

Property	JavaPOS	OPOS
AutoDisable	Not supported	All
BinaryConversion	Not supported	All
CapCompareFirmwareVersion		False
CapPowerReporting	PR_STANDARD	All support PR_STANDARD
CapStatisticsReporting	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CapUpdateFirmware		False
CapUpdateStatistics	False (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CheckHealthText		All
Claimed		All
DataCount		All
DataEventEnabled		All
DeviceControlDescription		All
DeviceControlVersion		All
DeviceEnabled		All
DeviceServiceDescription		All
DeviceServiceVersion		All
FreezeEvents		All
OpenResult	Not supported	All
OutputID		Not supported
PowerNotify		All
PowerState		All
PhysicalDeviceDescription		All
PhysicalDeviceName		All
ResultCode	Not supported	All
ResultCodeExtended	Not supported	All

Table 117. Magnetic Ink Character Recognition (MICR) common properties (continued)

Property	JavaPOS	OPOS
State		All

Table 118. Magnetic Ink Character Recognition (MICR) specific properties

Property	JavaPOS	OPOS
AccountNumber		All
Amount		All
BankNumber	All	Not supported
CapValidationDevice		All
CheckType		All - cannot determine without exception processing
CountryCode		All - cannot determine without exception processing
EPC		All
RawData		All
ErrorReportingType		All
SerialNumber		All
TransitNumber		All

Table 119. Magnetic Ink Character Recognition (MICR) common methods

Method	JavaPOS	OPOS
checkHealth		All
claim		All
clearInput		All
clearInputProperties		Supported
clearOutput		Not supported
close		All
compareFirmwareVersion		Not supported
directIO		Not supported
open		All
release		All
resetStatistics		Not supported
retrieveStatistics	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	Not supported
updateFirmware		Not supported
updateStatistics		Not supported

Table 120. Magnetic Ink Character Recognition (MICR) specific methods

Method	JavaPOS	OPOS
beginInsertion		All
beginRemoval		All
endInsertion		All
endRemoval		All

Table 121. Magnetic Ink Character Recognition (MICR) events

Event	JavaPOS	OPOS
DataEvent		All
DirectIOEvent		Not supported
ErrorEvent		All
StatusUpdateEvent		All

Magnetic Ink Character Recognition data

The format of the data read from Magnetic Ink Character Recognition (MICR) looks like the following:

Transit Symbol	Transit Field 9-digit fixed	Transit Symbol	On-Us Field 19-character max	Amount Symbol	Amount Field 10-digit	Amount Symbol
----------------	--------------------------------	----------------	---------------------------------	---------------	--------------------------	---------------

Where:

Transit Field

Nine-digit number bracketed between the two Transit symbols. Bank Number is digits 5 through 8 of the Transit field.

On-Us Field

According to the X9.13 standard, section 8.4.1, the On-Us field "usually contains the account number and *may* also contain a serial number and/or a transaction code." In addition, section 8.4.2 also states that "*no recommendation is made* as to how the On-Us field is to be structured." This fact allows the individual financial institution the widest possible latitude in designing the field to suit its own internal system requirement. However, in no case shall the On-Us field contain more than 19 characters.

Amount Field

Ten-digit number bracketed between the two Amount symbols.

Auxiliary On-Us field

If it exists, it is to the left of the Routing Number (Transit field). It is also stated that "when a serial number or other data is printed in this field, it must be preceded by, and followed by, an On-Us symbol." The field length is variable up to nine characters.

Note: Only On-Us and Auxiliary On-Us fields need interpretation rules.

JavaPOS configuration

MinMicrSignalLevel

```
<prop name="com.ibm.jpos.sdi.config.MICR.MinMicrSignalLevel" type="String" value="50"/>
```

Defines a minimum signal level accepted for a successful MICR read. If a MICR reads below the minimum signal level, an error event is fired.

Valid values are 0 to 100. Default is 20.

stripAccountDashes

```
prop name="com.ibm.jpos.sdi.config.MICR.stripAccountDashes" type="String" value="false"/
```

Determines whether to remove any dashes from the account number. When true, a value of "1234-56-7" would become "1234567".

Valid values are TRUE or FALSE.

stripTransitDashes

```
<prop name="com.ibm.jpos.sdi.config.MICR.stripTransitDashes" type="String" value="false"/>
```

Determines whether to remove any dashes from the transit number. When true, a value of "3579-12-" would become "357912".

Valid values are TRUE or FALSE.

switchTransitDashToSpace

```
<prop name="com.ibm.jpos.sdi.config.MICR.switchTransitDashToSpace" type="String" value="false"/>
```

Determines whether to change any dashes in the transit number to spaces. When true, a value of "12-4-66-7" would become "12 4 66 7".

Valid values are TRUE or FALSE.

Additional JavaPOS information

MICR exception tables

An exception table contains Transit Field/On-Us Field interpretation rule pairs. This enables you to manually code the On-Us field interpretation rule based on the Transit Field because each bank is likely to have a different Transit Field. For this purpose, three formats for Exception Table are supported. The first format (format 1) is an XML description of how the data is to be extracted from the check, and should be written in a separate file. Formats two (2) and three (3) can be put directly into the MICR JposEntry.

Format 1

Supported since JavaPOS .7.5, this is an XML representation of the On-Us and Auxiliary On-Us fields stored in an independent file. This file is identified in a property in the JposEntry for the MICR device:

```
<prop name="com.ibm.jpos.sdi.config.MICR.exceptionTableFile"
      Type="String"
      Value="
```

If this property is omitted, JavaPOS searches the classpath for a file named `ibmmicreexception.xml`.

This file can also contain entries for legacy formats (PositionSkip and Symbol formats). An entry for legacy format must be a child of 'MICRExceptionTable' and is just like the prop elements described for formats 2 and 3.

For example:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE JposEntries PUBLIC "-//IBMJavaPOS//DTD//EN"
  "/com/ibm/jpos/res/ibmmicreexception.dtd">

<MICRExceptionTable FormatToUse="XML">
  <ExceptionEntry TransitField="123456789" CheckType="Business">
    <OnUsFieldParsingRule>
      <SkipCharacterLength value="1" />
      <AccountNumberLength value="5" />
      <SkipCharacterLength value="1" />
      <SerialNumberLength value="5" />
    </OnUsFieldParsingRule>
  </ExceptionEntry>
</MICRExceptionTable>
```

```

</OnUsFieldParsingRule>
<AuxOnUsFieldParsingRule>
  <SerialNumberLength value="5" />
</AuxOnUsFieldParsingRule>
</ExceptionEntry>
<prop name="com.ibm.jpos.sdi.config.MICR.exceptionTable0"
      type="String"
      value="P123456789AAAAAAAAAXSSSSSSS"/>
<prop name="com.ibm.jpos.sdi.config.MICR.exceptionTable1"
      type="String"
      value="B778899001D154R"/>
</MICRExceptionTable>

```

A sample file called `ibmmicrexception.xml.Sample` is included with the installation. This can be useful as a starting point for configuring your own exception tables. The location for `ibmmicrexception.xml.Sample` is:

- For Windows: `C:\POS\IBMJPOS\Docs`
- For Linux: `/opt/ibm/javapos/docs`

The following considerations must be applied:

- There can be as many `ExceptionEntry` and `prop` elements as needed. `ExceptionEntry` elements must come before any `prop` element.
- Values for transit number (`TransitField` in `ExceptionEntry` and characters 2-10 in the case of `prop` elements) must be unique.
- Valid values for the attribute `FormatToUse` are XML and Legacy. The default is XML. This indicates which format is to be read first. Since any repeated value for transit number is dropped, this switches priority from one format to the other.
- Formats 2 and 3 (legacy formats) are read in order of the property names, first the ones in the new `ibmmicrexception.xml` and then the ones in the `JposEntry`. All listed entries must have consecutive names inside each file, and counts should start at 0.
- Order of elements within `OnUsFieldParsingRule` is strict. The order shown above is the expected order. `OnUsFieldParsingRule` is required.
- `SerialNumberLength` and `AccountNumberLength` can be in either order, and `SkipCharacterLength` can be at the beginning or between `SerialNumberLength` and `AccountNumberLength`.
- Exactly one occurrence of `AccountNumberLength` is required.
- `SerialNumberLength` and `SkipCharacterLength` are optional.
- All `value` attributes are required. `TransitField` attribute from `ExceptionEntry` element is also required.
- `CheckType` attribute from `ExceptionEntry` is optional. If specified, it must be either *Personal* or *Business*. The default is *Personal*.
- The serial number may appear in the Auxiliary On-Us field; `AuxOnUsFieldParsingRule` holds a `SerialNumberLength` element if this is the case. `AuxOnUsFieldParsingRule` is optional.
- `SerialNumberLength` should appear in one field only. If it appears in both, the value from `AuxOnUsFieldParsingRule` is ignored.

Format 2

Supported since JavaPOS 1.7.2, the On-Us field interpretation rule is as follows:

- | | |
|---|--|
| A | account number character (can be a dash) |
| S | serial number character (can be a dash) |
| X | ignored (no parsing) |

A typical MICR exception table looks like this:

```

<prop name="com.ibm.jpos.sdi.config.MICR.exceptionTable0"
      type="String" value="P123456789AAAAAAAAAXSSSSSSS"/>
<prop name="com.ibm.jpos.sdi.config.MICR.exceptionTable1"
      type="String" value="B123456709AAAAAAAAXXXSSSSSSS"/>

```

where the value attribute is parsed as shown in Table 122.

Table 122. MICR exception table fields for format 2

Characters	Content
1	Check type: B = Business check, P = Personal check
2 to 10	Transit field.
11 to end	On-Us Field. The A,X,S field is not required to be 19 characters long. You can choose the number of A, X, or S characters to encode the rule. This rule is applicable to the first valid digit immediately after the Transit Field. The On-Us symbol can be part of the On-Us Field, and the parser is expected to handle this condition.

Format 3

Supported since JavaPOS 1.5.1, exception configuration properties are provided for MICR processing to handle MICR data that do not conform to some of the standard formats. Default exception tables are set up for each MICR entry in the JPOS.xml file.

Use the following format:

```
<prop name="exceptionTable0" type="string" value= "B778899001D154R"/>
```

where each byte represents the following:

P/B (byte 1)

Type of check: either *Business* or *Personal*. This option is not used for parsing but instead is used to set the check type property, which is usually set to UNKNOWN.

TRANSIT (byte 2-10)

A 9-digit bank transit number that is key for determining if the exception table entry is applied to the MICR data.

SPC (byte 11)

SPC options:

- If SPC is set to *D*, all spaces are removed from the primary On-Us field for processing
- If SPC is not set to *D*, the On-Us field remains unchanged.

Field (byte 12)

Not used. The driver always use the first field in the primary On-Us field.

SKIP (byte 13)

The account starts in this index and includes the remaining characters to the right in the primary On-Us field. (The index count starts at 0.)

SIZE (byte 14)

Not used because the driver identifies the serial number.

ACC (byte 15)

Not used.

Use the following format:

```
<prop name="exceptionTable0" type="string" value= "B778899001D154R"/>
```

where each byte represents the following:

P/B (byte 1)

Type of check: either *Business* or *Personal*. This option is not used for parsing but instead is used to set the check type property, which is usually set to UNKNOWN.

TRANSIT (byte 2-10)

A 9-digit bank transit number that is key for determining if the exception table entry is applied to the MICR data.

SPC (byte 11)

SPC options:

- If SPC is set to *D*, all spaces are removed from the primary On-Us field for processing
- If SPC is not set to *D*, the On-Us field remains unchanged.

Field (byte 12)

Not used. The driver always use the first field in the primary On-Us field.

SKIP (byte 13)

The account starts in this index and includes the remaining characters to the right in the primary On-Us field. (The index count starts at 0.)

SIZE (byte 14)

Not used because the driver identifies the serial number.

ACC (byte 15)

Not used.

The following table shows sample output for Format 3:

Table 123. Format 3 Exception Table (sample data)

Raw Data	t123456780t 0123456789o 0671	
Primary On-Us	0123456789o 0671	
Field #1	0123456789	
Exception Table entry	Account	Serial
P123456780D111R	123456789	0671
P123456780D165R	6789	0671
P123456780D295R	9	0671
P123456780D422R	23456789	0671

OPOS configuration

Table 124. Service object settings for MICR

Keyword	Type	Description
ExceptionFile	String	Fully-qualified filename of MICR exception file, which contains exception processing information. The default value is: C:\POS\BIN\PARSE.DAT.
MinMicrSignalLevel	String	Minimum signal string (0-100) for successful MICR read. The default value is 20 (full string).
SlipRegTimeout	String	Time (in milliseconds) to wait for check to be registered. Note: This setting cannot be modified with the Configuration tool.
SlipRegMinDelay	String	Time (in milliseconds) to delay before sending document register command. Note: This setting cannot be modified with the Configuration tool.

OPOS DirectIO calls

Add new MICR exception

Adds the specified MICR exception string into the current MICR exception file.

Table 125. Add new MICR exception

Parameter	Type	Description
Command	Int32	101
Data	Long*	Ignored
Obj	BSTR*	Pointer to a string containing the MICR exception string. See "Additional OPOS information" for more information on the format of the exception string.

Additional OPOS information

MICR exception file

Exception configuration properties for MICR processing enable you to handle MICR data that do not conform to some of the standard formats. Default exception tables are set up for each MICR entry in the <INSTALL_DIR>\bin\parse.dat file.

Use the following format to create exception table entries:

```
#P/B TRANSIT SPC FIELD SKIP SZ ACC  
P 123456789 D 1 5 5 R
```

Each parameter must be separated for any number of spaces, and each byte represents the following:

P/B (byte 1)

The type of check: either *Business* or *Personal*. This option is not used for parsing; instead, it is used to set the check type property, which is usually set to "UNKNOWN."

TRANSIT (byte 2-10)

9-digit bank transit number, which is used as the key to determine if the exception table entry is applied to the MICR data.

SPC (byte 11)

If SPC is set to *D*, all spaces are removed from the primary On-Us field for processing; otherwise, the On-Us field remains unchanged.

Field (byte 12)

Indicates which subfield in the primary On-Us field is used to get the account number and the serial number.

SKIP (byte 13)

The account starts in this index and includes the remaining characters to the right in the primary On-Us field. (The index count starts at 0.)

SIZE (byte 14)

The length of the serial number.

ACC (byte 15)

The alignment of the account number in the subfield:

- If it is "L," then the account number aligns on the left of the subfield.
- If it is "R," then the account number aligns on the right of the subfield.

The following table shows sample output:

Table 126. Exception configuration properties for MICR processing: Example 1

Raw Data	t123456780t 0123456789o 0671		
Primary On-Ups	0123456789o 0671		
Field #1	0123456789		
Field #2	0671		
Exception Table entry	Account		Serial
P123456780D 1 1 1 R	9		8
P123456780D 1 6 5 R	456789		0123
P123456780D 2 1 5 R	1		067
P123456780D 4 1 5 R	1		067
P123456780D 2 2 2 L			06
P123456780D 1 1 1 L	01234567		8
P123456780D 1 3 5 L	01		23456

The following image illustrates how the information is parsed to retrieve the Account and Serial numbers for the table entry "P123456780D 1 6 5 R":

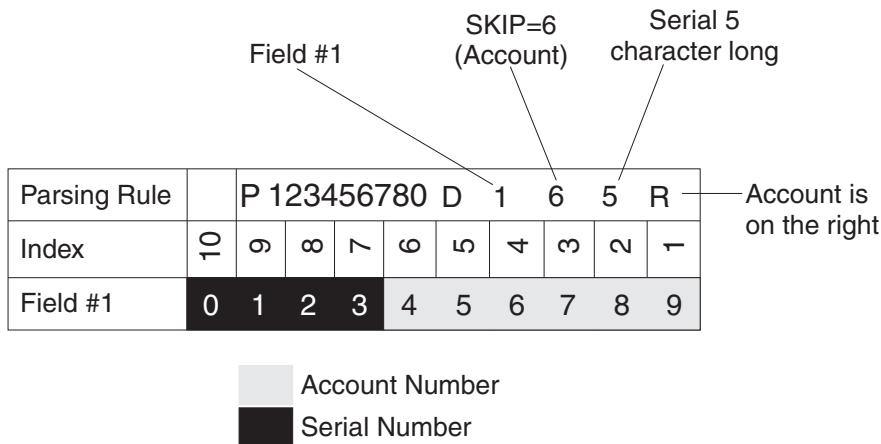


Figure 25. Data parsing: Example 1

The following image illustrates how the information is parsed to retrieve the Account and Serial numbers for the table entry "P123456780D 1 3 5 L":

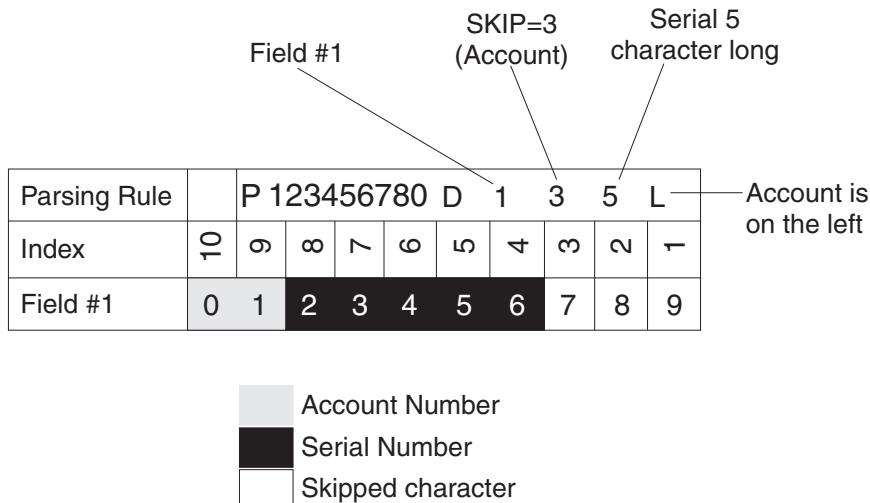


Figure 26. Data parsing: Example 2

Table 127. Exception configuration properties for MICR processing: Example 2

Raw Data	087654321o t123456780t 0123456789o	
Primary On-Us	0123456789o	
Field #1	0123456789	
Exception Table entry	Account	Serial
P123456780D 1 6 5 R	456789	0123
P123456780D 4 1 5 R	9	45678
P123456780D 2 2 2 L	012345	67
P123456780D 1 3 5 L	01	23456

Chapter 13. Motion sensor

Motion sensor supported devices

Table 128. Motion sensor supported devices

Device	Connectivity
1. SurePOS 500/600	Integrated
2. Kiosk	Integrated

Supported properties and methods

Table 129. Motion sensor common properties

Property	JavaPOS	OPOS
BinaryConversion	Not supported	All
CapCompareFirmwareVersion		False
CapPowerReporting	PR_NONE	PR_NONE
CapStatisticsReporting	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CapUpdateFirmware		False
CapUpdateStatistics	False (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CheckHealthText		All
Claimed		All
DeviceControlDescription		All
DeviceControlVersion		All
DeviceEnabled		All
DeviceServiceDescription		All
DeviceServiceVersion		All
FreezeEvents		All
OpenResult	Not supported	All
PhysicalDeviceDescription		All
PhysicalDeviceName		All
PowerNotify		All
PowerState		All
resultCode	Not supported	All
resultCodeExtended	Not supported	All
State		All

Table 130. Motion sensor specific properties

Property	JavaPOS	OPOS
Motion		All
Timeout		All

Table 131. Motion sensor common methods

Method	JavaPOS	OPOS
checkHealth		All
claim		Always shareable
close		All
compareFirmwareVersion		Not supported
directIO		Not supported
open		All
release		Always shareable
resetStatistics		Not supported
retrieveStatistics	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	Not supported
updateFirmware		Not supported
updateStatistics		Not supported

Table 132. Motion sensor specific methods

Method	JavaPOS	OPOS
waitForMotion		All

Table 133. Motion sensor events

Event	JavaPOS	OPOS
DirectIOEvent		Not supported
StatusUpdateEvent		All

Additional JavaPOS information

This section contains information on loading the correct motion sensor driver in Linux for your hardware.

Add the following lines in /opt/ibm/javapos/etc/machine.conf:

```
<keyword> <number of entries>
<machine type><model number>
```

"<keyword>" represents the device for which a driver is needed and "<number of entries>" indicates the number of "<machine type><model number>" pairs that follow. If the system on which JavaPOS is running matches one of the specified "<machine type><model number>" pairs, the device driver for that "<machine type><model number>" is loaded.

For the motion sensor, machine.conf might have the following entries:

```
Motion 2
4838132
4838135
```

In this case the <number of entries> value is **2**.

Chapter 14. Magnetic stripe reader

Supported devices

Table 134. Magnetic stripe reader (MSR) supported devices

Device		Connectivity
ISO (3-track)	1. Retail alphanumeric POS keyboard w/Card Reader	PS/2, RS-485, USB
ISO (3-track)	2. Retail alphanumeric POS keyboard with MSR and pointing device	PS/2
ISO (3-track)	3. Retail POS keyboard with card reader	RS-485, USB
ISO (3-track)	4. Retail POS keyboard with card reader and display	RS-485, USB
ISO (3-track)	5. Modifiable layout keyboard with card reader	RS-485, USB
ISO (3-track)	6. 32-key programmable keypad with ISO MSR	EIA-232, RS-485, USB
JUCC	7. 32-key programmable keypad with JUCC MSR	EIA-232, RS-485, USB
ISO (3-track)	8. SurePoint 4820 integrated ISO MSR	EIA-232, RS-485, USB
JUCC	9. SurePoint 4820 integrated JUCC MSR	EIA-232, RS-485, USB
ISO (3-track)	10. SurePOS 500/600 integrated ISO MSR	EIA-232
JUCC	11. SurePOS 500/600 integrated JUCC MSR	EIA-232
ISO (3-Track)	12. Anyplace Kiosk integrated ISO MSR	EIA-232
JUCC	13. Anyplace Kiosk integrated JUCC MSR	EIA-232
ISO (3-Track)	14. SureOne Integrated ISO MSR	PS/2
ISO (3-Track)	15. . Compact Alphanumeric POS Keyboard (requires manual firmware update for support)	PS/2
JUCC	16. POS keyboard V	RS-485, USB
JUCC	17. PC POS Keyboard or PC Point of Sale Keyboard	PS/2, RS-485, USB
JUCC	18. Retail POS keyboard VI	RS-485, USB
ISO (3-Track)	19. SurePOS 100 integrated ISO MSR	PS/2
JUCC	20. 4674 Integrated JUCC MSR	RS-485
JUCC	21. 4685-KC1 (OPOS only)	RS-485
JUCC	22. 4685-K01 (OPOS only)	RS-485
JUCC	23. 4685-K02 (Ultra7) keyboard with card reader	RS-485, USB
JUCC	24. 4685-K02 with MSR/Encoder and 4 position keylock	RS-485
JUCC	25. 4685-K02 with MSR/Encoder and 6 position keylock	RS-485
JUCC	26. 4685-K03	RS-485, USB
ISO (3-Track)	27. Modular alphanumeric keyboard	PS/2, USB
JUCC	28. Modular alphanumeric keyboard	PS/2, USB
ISO (3-Track)	29. Modular Compact Alphanumeric keyboard	PS/2, USB
JUCC	30. Modular Compact Alphanumeric keyboard	PS/2, USB
ISO (3-Track)	31. Modular 67 Key Keyboard	PS/2, USB
JUCC	32. Modular 67 Key Keyboard	PS/2, USB

Table 134. Magnetic stripe reader (MSR) supported devices (continued)

Device		Connectivity
ISO (3-Track)	33. Modular 67 Key Keyboard with Card Reader and Display USB	PS/2, USB
ISO (3-Track)	34. TCxWave 6140 Integrated ISO MSR	USB

Supported properties and methods

Table 135. MSR common properties

Property	JavaPOS	OPOS
AutoDisable	Not supported	All
BinaryConversion	Not supported	All
CapCompareFirmwareVersion		False
CapPowerReporting	PR_NONE for EIA-232 and PS/2 PR_STANDARD for all other MSR devices	PR_STANDARD
CapStatisticsReporting	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CapUpdateFirmware		False
CapUpdateStatistics	False (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CheckHealthText		All
Claimed		All
DataCount		All
DataEventEnabled		All
DeviceControlDescription		All
DeviceControlVersion		All
DeviceEnabled		All
DeviceServiceDescription		All
DeviceServiceVersion		All
FreezeEvents		All
OpenResult	Not supported	All
OutputID		Not supported
PowerNotify		All
PowerState		All
PhysicalDeviceDescription		All
PhysicalDeviceName		All All
ResultCode	Not supported	All
ResultCodeExtended	Not supported	All
State		All

Table 136. MSR specific properties

Property	JavaPOS	OPOS
AccountNumber		All
AdditionalSecurityInformation		Not supported
CapCardAuthentication		Not supported
CapDataEncryption		Not supported
CapDeviceAuthentication		Not supported
CapISO	All except JUCC 4840 MSR	All except JUCC 4840 MSR
CapJISOne	All	All except 4840 MSR, SureOne Built-In, and 4820 MSR
CapJISTwo	All	All except 4840 MSR and SureOne Built-In
CapTrackDataMasking		Not supported
CapTransmitSentinels	False	All
CapWritableTracks	24, 25	Not supported
CardAuthenticationData		Not supported
CardAuthenticationDataLength		Not supported
CardPropertyList	All	Not supported
CardType	All	Not supported
CardTypeList	All	Not supported
DataEncryptionAlgorithm		Not supported
DecodeData		All
DeviceAuthenticated		Not Supported
DeviceAuthenticationProtocol		Not Supported
EncodingMaxLength	24, 25	Not supported
ErrorReportingType		All
ExpirationDate		All
FirstName		All
MiddleInitial		All
ParseDecodeData		All
ServiceCode		All
Suffix		All
Surname		All
Title		All
Track1Data		All
Track1DiscretionaryData		All
Track1EncryptedData		Not supported
Track1EncryptedDataLength		Not supported
Track2Data		All
Track2DiscretionaryData		All
Track2EncryptedData		Not supported
Track2EncryptedDataLength		Not supported

Table 136. MSR specific properties (continued)

Property	JavaPOS	OPOS
Track3Data		All
Track3EncryptedData		Not supported
Track3EncryptedDataLength		Not supported
Track4Data		All JUCC
Track4EncryptedData		Not supported
Track4EncryptedDataLength		Not supported
TracksToRead		All
TracksToWrite	24, 25	Not supported
TransmitSentinels		Not supported
WriteCardType		Not supported

Table 137. MSR common methods

Method	JavaPOS	OPOS
checkHealth		All
claim		All
clearInput		All
ClearInputProperties	All	Not Supported
clearOutput		Not supported
close		All
compareFirmwareVersion		Not supported
directIO	RS-485 – 4685-K02 (Ultra 7)	Not supported
open		All
release		All
resetStatistics		Not supported
retrieveStatistics	True (see Appendix A, “JavaPOS support for UnifiedPOS device statistics properties,” on page 431)	Not supported
updateFirmware		Not supported
updateStatistics		Not supported

Table 138. MSR specific methods

Method	JavaPOS	OPOS
authenticateDevice		Not supported
deauthenticateDevice		Not supported
retrieveCardProperty	All	Not supported
retrieveDeviceAuthenticationData		Not supported
updateKey		Not supported
writeTracks	24, 25	Not supported

Table 139. MSR events

Event	JavaPOS	OPOS
DataEvent		All
DirectIOEvent		Not supported
ErrorEvent		All
StatusUpdateEvent		All

JavaPOS configuration

enableOnlineWatcher

```
<prop name="com.ibm.posj.bus.rs232.enableOnlineWatcher" type="Boolean" value="true"/>
```

Monitors the device for being online or offline and performs appropriate actions. To configure the poll time for events, modify the com.ibm.posj.bus.rs232.onLineWatcherPollTime.MSR property in the posj.properties file.

Values accepted

- TRUE - Default value
- FALSE

Device Support

- EIA-232 : 12, 13

msrID

```
<prop name="com.ibm.posj.bus.rs232.msrId" type="String" value="ISO"/>
```

Determines whether the EIA-32 MSR hardware is a 3-track ISO unit or a double-sided JUCC unit.

Valid values accepted:

- ISO (default value)
- JUCC

Device support:

- EIA-232: 6, 7, 8, 9, 10, 11, 12, 13

JavaPOS DirectIO calls

The following DirectIO commands are supported for the MSR. The syntax is as follows:

```
Syntax directIO (command: int32, inout data: int32, inout obj: object) :  
void {raises-exception, use after open-claim-enable}
```

MSR write data command

Note: This command is used to write data to the JIS magnetic stripe of payment cards conforming to the JIS specification.

See Table 140 on page 148 for a reference of the parameters to call DirectIO.MSR_WRITE_DATA_CMD.

Table 140. Parameters for DirectIO.MSR_WRITE_DATA_CMD

Parameter	Type	Value
Command	Int32	DirectIO.MSR_WRITE_DATA_CMD
Data [0]	Int32	Write wait timeout value in milliseconds. The wait timeout can be set to JposConst.JPOS_FOREVER and the command has no timeout to return. The maximum wait timeout is 25,000 milliseconds (25 seconds).
Obj	Object	String representation of the data to be written. The maximum length for the object is 69 characters. The first 69 characters will be written to the magnetic stripe and additional characters will be ignored.

Note: A card with a JIS II track must be swiped through the MSR during the directIO call with MSR_WRITE_DATA_CMD so data can be stored. Otherwise an exception is thrown.

Device Support:

- 24, 25

Errors

A UposException Exception may be thrown when this method is invoked. Some possible values of the exception's ErrorCode property are noted in Table 141:

Table 141. UposExceptionException error code property

Value	Possible causes
E_ILLEGAL	One of the following errors has occurred: <ul style="list-style-type: none"> • The device is not write enabled. • Some of the parameter values are not correct. • Command sent to not capable write MSR
E_TIMEOUT	Timeout expired and card was not swiped.

Additional JavaPOS information

SureOne MSR

SureOne firmware discards any invalid MSR reading and sends only valid track data. Track errors are not reported.

If track 2 is empty and track 3 contains data, the track 3 data is displayed as track 2 and no data appears in track 3.

OPOS configuration

Table 142. Service object settings for EIA-232 MSR

Keyword	Type	Description
MsrType	REG_SZ	Type of MSR reader attached. Valid values are: ISO Standard three-track reader JUCC JIS-I and II reader

Table 142. Service object settings for EIA-232 MSR (continued)

Keyword	Type	Description
ReadTotalTimeout	REG_SZ	Set the total read timeout of a single MSR swipe. Default: 1500 ms
ReadIntervalTimeout	REG_SZ	Set the read interval timeout between tracks of a single MSR swipe. Default: 190 ms Note: Due to error checking, it takes more time to read MSR cards which are not in good condition. To allow the reading of wider conditions of MSR cards, you may adjust the value up to 400 ms.

Additional OPOS information

SureOne/SurePOS 100 MSR limitations

The MSR on the SureOne is part of the Keyboard. The raw data is in ASCII format.

When a credit card is swiped with the MSR DataEventEnabled set to *false*, the incoming data is treated as keyboard data. The only way to get MSR data events is to set DataEventEnabled to *true*. In order to queue MSR data, both DataEventEnabled and FreezeEvents must be set to *true*.

Chapter 15. POS keyboard

POS keyboard supported devices

Table 143. POS keyboard supported devices

Device	Connectivity
1. Retail alphanumeric POS keyboard with card reader Also known as: ANPOS or NANPOS	PS/2, RS-485, USB
2. Retail alphanumeric POS keyboard with MSR and pointing device Also known as: ANPOS with pointer	PS/2
3. Retail POS keyboard Also known as: 50-Key keyboard	RS-485, USB
4. Retail POS Keyboard with card reader Also known as: 50-Key/MSR	RS-485, USB
5. Retail POS Keyboard with card reader and display Also known as: 50-Key/Display	RS-485, USB,
6. Modifiable Layout Keyboard with card reader Also known as: 133 Key or Matrix	RS-485, USB
7. 32-Key programmable keypad with MSR Also known as: 4820 Keyboard	RS-485, USB, PS/2
8. SureOne Integrated Keyboard	PS/2
9. Compact Alphanumeric POS Keyboard Also known as: CANPOS	PS/2
10. POS Keyboard V	RS-485, USB
11. PC POS Keyboard or PC Point of Sale keyboard Also known as: ANKPOS	PS/2, RS-485, USB
12. PLU Keyboard/Display III	RS-485, USB
13. Retail POS Keyboard VI Also known as: 50-Key keyboard with JUCC MSR	RS-485, USB
14. SurePOS 100 Integrated keyboard	PS/2
15. 4674 Integrated keyboard	RS-485
16. 4685-KCI (OPOS only)	RS-485
17. 4685-K01 (OPOS only)	RS-485
18. 4685-K02 (Ultra 7) keyboard with card reader	RS-485, USB
19. 4685-K02 with MSR/Encoder and 4 position keylock	USB
20. 4685-K02 with MSR/Encoder and 6 position keylock	USB
21. 4685-K03	RS-485, USB
22. Modular Alphanumeric keyboard	PS/2, USB

Table 143. POS keyboard supported devices (continued)

Device	Connectivity
23. Modular Compact Alphanumeric Keyboard	PS/2, USB
24. Modular 67-Key Keyboard	USB
25. Modular 67-Key Keyboard with Display USB	USB

Supported properties and methods

Table 144. POS keyboard common properties

Property	JavaPOS	OPOS
AutoDisable	Not supported	All
BinaryConversion	Not supported	All
CapCompareFirmwareVersion		False
CapPowerReporting	PR_STANDARD	PR_NONE for PS/2 connectivity, PR_STANDARD for other connectivities.
CapStatisticsReporting	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CapUpdateFirmware		False
CapUpdateStatistics	False (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CheckHealthText		All
Claimed		All
DataCount		All
DataEventEnabled		All
DeviceControlDescription		All
DeviceControlVersion		All
DeviceEnabled		All
DeviceServiceDescription		All
DeviceServiceVersion		All
FreezeEvents		All
OpenResult	Not supported	All
OutputID		Not supported
PowerNotify		All
PowerState		All
PhysicalDeviceDescription		All
PhysicalDeviceName		All
resultCode	Not supported	All
resultCodeExtended	Not supported	All
State		All

Table 145. POS keyboard specific properties

Property	JavaPOS	OPOS
CapKeyUp		All
EventTypes		All
POSKeyData		All
POSKeyEventType		All

Table 146. POS keyboard common methods

Method	JavaPOS	OPOS
checkHealth		All
claim		All
clearInput		All
clearOutput		All
close		All
compareFirmwareVersion		Not supported
directIO	All	All except 8, 14
open		All
release		All
resetStatistics		Not supported
retrieveStatistics	True (see Appendix A, “JavaPOS support for UnifiedPOS device statistics properties,” on page 431)	Not supported
updateFirmware		Not supported
updateStatistics		Not supported

Table 147. POS keyboard events

Event	JavaPOS	OPOS
DataEvent		All
DirectIOEvent	All	Not supported
ErrorEvent		All
StatusUpdateEvent		All

CANPOS keyboard

The CANPOS keyboard firmware version must be at least 1.2.3 for the CANPOS automatic download utility to work. The firmware can be automatically updated to newer versions in UnifiedPOS. If required, the update typically takes three to five minutes, depending on the operating system.

The keyboard function keys can also be updated automatically, but you must create your own keyboard configuration file. To create a configuration file, a separate utility is available from the Toshiba Global Commerce Solutions support website. To download and use this utility, perform the following steps:

1. Go to www.toshibacommerce.com/support.
2. Click **Other Systems and Devices**.
3. Click **Firmware and Utilities for CANPOS**. All utilities are packaged in a single zip file for supporting different operating systems. Based on your preferred system, you can download the appropriate utility.

4. Unzip the package and choose the appropriate utility to generate the configuration file.
5. Rename or save your file as aipcnpos.pcf and place it in folder \pos\config for Windows systems, or /usr/share/pos for Linux systems.
6. Reboot your POS system and the configuration file is loaded automatically.

To confirm the success of the firmware and configuration download, check aipanpos.log under \pos\log in Windows, or /var/log in Linux POS systems.

Installing USB system-attached keyboard in Windows XP

The proper device driver is automatically installed for the USB system keyboard in Windows XP if you select **Alphanumeric Point of Sale Keyboard** and indicate that it is attached to a USB port during installation. After reboot, a Windows message is displayed indicating that you are installing the POS USB Keyboard. Click **Continue Anyway** to install Toshiba's driver.

Note: If you are installing the keyboard for the first time, the system must be rebooted again before the new driver can be used.

JavaPOS configuration

Click

```
<prop name="com.ibm.jpos.sdi.config.POSKeyboard.Click" type="Byte" value="0x00"/>
```

JavaPOS provides a user-configurable mechanism to modify the loudness of the keyboard click. This is accomplished by specifying the following in the keyboard's JposEntry.

Valid values for the Click property (default is 0x00):

- 0x00 (off)
- 0x01 (soft)
- 0x02 (loud)

Device Support*:

- 1, 2, 3, 4, 5, 6, 7, 11, 13, 19, 20, 21, 22, 23, 24
- USB Only: 10, 12, 18, 21

*USB not supported when configured as system keyboard.

FatFingerTimeout

```
<prop name="com.ibm.jpos.sdi.config.POSKeyboard.FatFingerTimeout" type="Byte" value="0x03"/>
```

A fat-finger situation occurs when an operator attempts to press a single key, but misses the center of the key and presses both the intended key and an adjacent key. The time interval between subsequent key strokes is measured by the keyboard's processor. If the two keys were pressed within a certain interval, a fat-finger situation is assumed and a status indicating this occurrence accompanies the scan codes to the terminal.

Device Support:

- 1 (RS-485, USB)*, 3, 4, 5, 6, 7 (RS-485, USB), 11 (RS-485, USB)*, 13, 19, 20, 24
- USB Only: 10, 12, 18, 21, 22*, 23*

*USB not supported when configured as system keyboard.

Valid values are 0 through 4. The default is 3:

- 0 = 0 milliseconds (fat-finger detection disabled)

1 = 10 milliseconds
2 = 20 milliseconds
3 = 30 milliseconds
4 = 40 milliseconds

ExtendedKeyMapping

```
<prop name="com.ibm.jpos.sdi.config.POSKeyboard.ExtendedKeyMapping" type="Boolean" value="true"/>
```

The extended scan code property is used to differentiate between two keys having the same hardware scan code. A typical extended scan code is represented by two scan codes, for example: X'e0' X'23'.

When the ExtendedKeyMapping property is *true*, the first scan code of the extended key is replaced by X'01'; the second scan code is not altered. For example, the extended key X'e0' X'23' is returned to the application as X'123'.

When the ExtendedKeyMapping property is *false*, the extended keys are not supported. Only a single scan code is returned to the application (X'0023').

Valid values are TRUE and FALSE. The default is TRUE.

Device Support:

- PS2: 1, 2, 8, 9, 11, 14, 22, 23
- USB when configured as system keyboard: 1, 11, 22, 23

KbdScanning

```
<prop name="com.ibm.jpos.sdi.config.POSKeyboard.KbdScanning" type="Boolean" value="true"/>
```

This property controls whether to enable or disable keyboard scanning. When scanning is enabled, the keyboard generates make and break scan codes for each key press.

Device Support:

- 1, 2, 3, 4, 5, 6, 7, 9, 10, 11, 12, 13, 15, 18, 19, 20, 21, 22, 23, 24

Valid values are TRUE and FALSE. The default is TRUE.

KeyTranslationFile

```
<prop name="KeyTranslationFile" type="String" value="\keys.dat"/>
```

The KeyTranslationFile property contains the full path name of the file containing key translation information. A sample file called keys.dat is included with the installation.

The location of the sample file depends on the operating system:

- Linux: /opt/ibm/javapos/docs
- Windows: C:\POS\IBMJPOS\Docs

Typematic

```
<prop name="com.ibm.jpos.sdi.config.POSKeyboard.Typematic" type="Boolean" value="true"/>
```

This property controls the typematic function of the keyboard.

Most of the keys on all keyboards are optionally typematic keys. The typematic function is enabled or disabled for the entire keyboard with this option.

If the typematic function is disabled and a key is pressed, then the make key code is generated only once—no matter how long that key is held down. If the key is released, the break code is sent.

If the typematic function is enabled and a key is pressed, the make key code is generated. If the key remains pressed for a period longer than the amount of time defined by typematic delay in the `jpos.xml`, another make key code is generated. As long as the key remains pressed, make key codes are generated at the rate defined by typematic rate until the key is released, at which time the break code is sent.

Valid values are TRUE and FALSE. The default is TRUE.

Device Support:

- 1*, 2, 3, 4, 5, 6, 7, 11*, 13, 19, 20, 21
- USB Only: 10, 12, 18, 21, 22*, 23*, 24

*USB on Linux not supported when configured as system keyboard.

TypeMaticDelay

```
<prop name="com.ibm.jpos.sdi.config.POSKeyboard.TypeMaticDelay" type="Byte" value="0x01"/>
```

Device Support:

- 1 (RS-485, USB)*, 3, 4, 5, 6, 7 (RS-485, USB), 11 (RS-485, USB)*, 13, 19, 20, 22*, 23*
- USB Only: 10, 12, 18, 21

*USB not supported when configured as system keyboard.

The typematic delay determines how long the key must be pressed before the typematic function is enabled.

Valid values for the TypeMaticDelay property are listed below; the default is 0x01:

0x00 = Delay 250 milliseconds
0x01 = Delay 500 milliseconds
0x02 = Delay 750 milliseconds
0x03 = Delay 1000 milliseconds

TypeMaticRate

The typematic rate determines how many make codes are generated per second (+/-20%).

```
<prop name="com.ibm.jpos.sdi.config.POSKeyboard.TypeMaticRate" type="Byte" value="0x14"/>
```

Device Support:

- 1 (RS-485, USB)*, 3, 4, 5, 6, 7 (RS-485, USB), 11 (RS-485, USB)*, 13, 19, 20, 22*, 23*
- USB Only: 10, 12, 18, 21

*USB not supported when configured as system keyboard

Valid values for the TypeMaticRate property are listed below. The default is 0x14.

0x00 = Rate 2.0
0x01 = Rate 2.1
0x02 = Rate 2.3
0x03 = Rate 2.5
0x09 = Rate 4.3

```

0x0a = Rate 4.6
0x0b = Rate 5.0
0x0c = Rate 5.5
0x0d = Rate 6.0
0x0e = Rate 6.7
0x0f = Rate 7.5
0x10 = Rate 8.0
0x11 = Rate 8.6
0x12 = Rate 9.2
0x13 = Rate 10.0
0x14 = Rate 10.9
0x15 = Rate 12.0
0x16 = Rate 13.3
0x17 = Rate 15.0
0x18 = Rate 16.0
0x19 = Rate 17.1
0x1a = Rate 18.5
0x1b = Rate 20.0
0x1c = Rate 21.8
0x1d = Rate 24.0
0x1e = Rate 26.7
0x1f = Rate 30.0

```

JavaPOS DirectIO calls

```
Syntax directIO ( command: int32, inout data: int32, inout obj: object ): void {
    raises-exception, use after open-claim-enable}
```

To access DirectIO constants, import com.ibm.jpos.services.DirectIO.

Get keyboard light status command

This command returns the indicators status in data[0].

Table 148. DirectIO.GET_KBD_LIGHT_STATUS_CMD

Parameter	Type	Description
Command	Int32	DirectIO.GET_KBD_LIGHT_STATUS_CMD
Data	Int32	An int array with at least one element
Obj	Object	Any value. A null value is accepted

Remarks

Returns the indicator status on data[0] reference parameter. The bit values in the int parameter (data[0]) are:

- '1' = LED is to be turned on
- '0' = LED is to be turned off

Table 149. DirectIO.GET_KBD_LIGHT_STATUS_CMD bit representation

Parameter	Bit representation of the data element	
7-31 bits	Reserved - must	= 0
6	SCROLL-LOCK	= DirectIO.KBD_SCROLL_LOCK_LIGHT
5	CAPS-LOCK	= DirectIO.KBD_CAPS_LOCK_LIGHT
4	NUM-LOCK	= DirectIO.KBD_NUM_LOCK_LIGHT

Table 149. DirectIO.GET_KBD_STATUS_CMD bit representation (continued)

Parameter	Bit representation of the data element	
3	"no label" or READY	= DirectIO.KBD_NO_LABEL_LIGHT
2	MSG PEND or SYS MSG	= DirectIO.KBD_MESSAGE_PENDING_LIGHT
1	OFFLINE	= DirectIO.KBD_OFFLINE_LIGHT
0	WAIT	= DirectIO.KBD_WAIT_LIGHT

Set keyboard light on command

This command illuminates the indicator specified on the data[0] parameter.

Table 150. DirectIO.TURN_KBD_LIGHT_ON_CMD bit representation

Parameter	Type	Description
Command	Int32	DirectIO.TURN_KBD_LIGHT_ON_CMD
Data[0]	Int32	Data can be set to any of the following options: DirectIO.KBD_WAIT_LIGHT DirectIO.KBD_OFFLINE_LIGHT DirectIO.KBD_MESSAGE_PENDING_LIGHT DirectIO.KBD_NO_LABEL_LIGHT DirectIO.KBD_NUM_LOCK_LIGHT DirectIO.KBD_CAPS_LOCK_LIGHT DirectIO.KBD_SCROLL_LOCK_LIGHT
Obj	Object	Any value. A null value is accepted.

Set keyboard light off command

This command turns off the indicator specified on the data[0] parameter.

Table 151. DirectIO.TURN_KBD_LIGHT_OFF_CMD

Parameter	Type	Description
Command	Int32	DirectIO.TURN_KBD_LIGHT_OFF_CMD
Data[0]	Int32	Data can be set to any of the following options: DirectIO.KBD_WAIT_LIGHT DirectIO.KBD_OFFLINE_LIGHT DirectIO.KBD_MESSAGE_PENDING_LIGHT DirectIO.KBD_NO_LABEL_LIGHT DirectIO.KBD_NUM_LOCK_LIGHT DirectIO.KBD_CAPS_LOCK_LIGHT DirectIO.KBD_SCROLL_LOCK_LIGHT
Obj	Object	Any value. A null value is accepted.

A UposExceptionException might be thrown when this method is invoked.

Additional JavaPOS information

Firmware Update

The JavaPOS provides a mechanism to update the firmware for Modular keyboards for USB and PS/2 connectivity. The driver release contains the most up-to-date firmware files for these keyboards and the firmware will be automatically updated when system boots. The table below describes Modular keyboard type, firmware file, and location of the firmware file on the system. To update the firmware, copy the new firmware file at the location specified in the table below and reboot the system.

Table 152. Modular Keyboards (Device 22, 23, 24, 25)

Modular Keyboard Type	Firmware File	Location on Windows	Location on Linux
67-Key without MSR	aip46013.dat	c:\pos\usb	/opt(ibm/javapos/flash/usb
67-Key with MSR	aip46063.dat	c:\pos\usb	/opt(ibm/javapos/flash/usb
Alphanumeric	aip46043.dat	c:\pos\usb	/usr/share/pos/usb
Compact Alphanumeric	aip46093.dat	c:\pos\usb	/usr/share/pos/usb
67-Key with Display USB	aip46023.dat	c:\pos\usb	/usb/share/pos/usb

Table 153. Compact Alphanumeric Keyboard (Device 9)

Keyboard Type	Firmware File	Location on Windows	Location on Linux
Compact Alphanumeric Keyboard	aipcnpos.dat	c:\pos\firmware	/usr/share/pos/firmware

Configuration file update

The JavaPOS provides a mechanism to update the configuration file for Modular keyboards for USB and PS/2 connectivity. This is a two step process.

To avoid conflict with JavaPOS drivers, it is recommended to install this utility on a separate system which does not contain JavaPOS drivers.

1. Download Configuration Utility and install it.
 - Go to www.toshibacommerce.com.
 - Click **POS Keyboards** under Peripherals.
 - Click **Downloads** under Toshiba Modular POS Keyboards.
 - Click **Modular POS Keyboard & 4820 Keypad/MSR/Keylock Firmware & Configuration Utility**.
 - Install utility.
2. Attach the Modular keyboard to the system and using the utility, create a configuration file. Depending on keyboard type, the configuration will assign the correct configuration file name. However, make sure that it complies to the table below.
3. Copy the configuration file to the location specified in the table below.
4. Reboot the system.

Table 154. Modular Keyboards (Device 22, 23, 24)

Modular Keyboard Type	Configuration File	Location on Windows	Location on Linux
67-Key without MSR	aip46013.cfg	c:\pos\config	/usr/share/pos/config
67-Key with MSR	aip46063.cfg	c:\pos\config	/usr/share/pos/config

Table 154. Modular Keyboards (Device 22, 23, 24) (continued)

Modular Keyboard Type	Configuration File	Location on Windows	Location on Linux
Alphanumeric	aip46043.cfg	c:\pos\config	/usr/share/pos/config
Compact Alphanumeric	aip46093.cfg	c:\pos\config	/usr/share/pos/config

Keyboard definition file

The keyboard definition file is used in conjunction with the KeyTranslationFile property. The file allows the user to map physical keys into virtual key codes. It also allows the user to configure double keys.

Virtual key codes

Each line of the table consists of an integer virtual keyboard code followed by white space (any combination of spaces and tabs) followed by the input sequence consisting of any number of actual keyboard codes. Individual codes in the input sequence are separated by white spaces.

For example, the line for up-arrow might be coded like this:

301 01 26 30

This coding indicates that when the sequence of values 01, 26, 30 comes in, it is recognized as virtual keyboard code 301. (The characters corresponding to 01, 26, and 30 are Esc, [, and A.)

It is also possible to generate several virtual scan codes at once. This is achieved by grouping all the virtual scan codes to be generated within square brackets. The following example shows an entry indicating the drivers to generate two virtual scan codes: 302, followed by 303 when the value 141 is scanned.

[302 303] 141

The actual keyboard codes needed to generate multiple virtual scan codes can be a sequence, as in the first example:

[312 320 322] 141 29

In this case the virtual scan code sequence 312-320-322 is generated when the scan code sequence is complete. This means that the single 141 code does not trigger any scan code.

The character-encoding rules are:

1. Any single character, except space, tab, and # stands for itself. The example sequence can be coded as:

301 01 [A 2

2. A sequence of two or more digits is interpreted as an integer value, as in the example. If an integer starts with a zero, it is interpreted as an octal number.

The example above can be coded as:

301 01 032 036

If an integer starts with a 0x, it is interpreted as a hex number.

3. An integer (decimal, hex, or octal) can have a leading plus sign.

This is essential for coding decimal integers less than ten, because a single digit alone is interpreted as an ASCII character under Rule 1. A # character could be coded as +4 but not as a 4 only.

4. A # starts a comment. It and the rest of the line are skipped.

If a line consists of nothing but a comment, or is empty, it is skipped.

5. The character's space, tab, and # cannot be coded literally. They can be coded as integers 57, 15, and +4 (Windows scan codes).

6. Input characters that are their own virtual keyboard codes (such as carriage return, A, Z, =, ...) do not need to be coded in the table. When the Java Keyboards Mapping function is reading characters and is about to begin a new sequence, a character that does not match any sequence is returned as is.
7. When the finite-state machine is in the middle of some candidate sequences, a non matching character signals the end of sequence. All previous characters from the beginning of sequence to the non matching character are returned as is.

The finite-state machine then resets to accept a new sequence. For example, assume that 99 is the scan code for a, and 100 is the scan code for b. The entries in the keys.dat file are:

```
400 99 b
```

If you press the **a** key, and then press the **b** key, two data events are generated:

- Press the **a** key (scan code 99). The **a** key does not return scan codes until scan code completes the sequence.
- Press the **b** key. The **b** key does return 2 data events: scancode 99 (for a) and scancode 100 (for b); the rule is not reached.

8. Virtual keycode value range from 0000 to 0xFFFF and keyboard code value range from 00 to 0xFF. If any of the values exceed its range, the line is skipped.

When making the keyboard definition table, only scan codes may be used for the system-attached keyboard. Characters cannot stand for themselves, as stated in Rule 1.

Double key support

The double key definition format is as follows:
 0xFF <expected scan code> <scancode for 1st key>
 <scancode for 2nd key> <filter time in ms>

The <filter time in ms> is optional.

Its default value is zero, and the recommended filter time is 60 milliseconds

For example:

```
0xFF 0X1C 0x71 0x1C 50
0xFF 0x7E 0x7E 0x5F 0x32
```

where:

0x1C is the scancode for key switch 109
 0x71 is the scancode for key switch 108
 0x7E is the scancode for key switch 106
 0x5F is the scancode for key switch 107

Whenever keys 108 and 109 are pressed within 50 milliseconds of each other, the scancode returned to the application is 0x1C.

Whenever keys 106 and 107 are pressed within 50 milliseconds of each other, the scan code returned to the application is 0x7E.

The filter time specified for both double key definitions in the example above is 50 milliseconds

USB System attached keyboard on Linux

When there is a USB System keyboard, you can not have a PS/2 keyboard plugged into the system. It might cause the USB Keyboard to malfunction.

OPOS configuration

Table 155. Settings for POS keyboard

Keyword	Type	Description
NumlockOn	String	Initial state of Num Lock light. Overrides the Numlock setting. Valid values are: True Illuminated (other value) Dark (default)
NumlockEnabled	String	Whether the Num Lock key is enabled. Valid values are: True Enabled (other value) Disabled (default)
ScrollockOn	String	Initial state of Scroll Lock light. Valid values are: True Illuminated (other value) Dark (default)
ScrollockEnabled	String	Whether the Scroll Lock key is enabled. Valid values are: True Enabled (other value) Disabled (default)
Typematic	String	Whether the typematic function is enabled. Valid values are: True Enabled (other value) Disabled (default)

Note: The settings above apply to the physical keyboard. Since one physical keyboard can contain multiple logical devices (MSR, keylock, Line display, POS keyboard), these settings are not stored with the normal service object settings, but are stored in a different location in the registry (HKLM\SOFTWARE\OLEforRetail\ServiceInfo).

Table 156. Service Object settings for POS keyboard

Keyword	Type	Description
MapPosKeys	String	Filename of Key Map file. The default value is C:\POS\BIN\KBDKMAP.DAT.
MapKeyboardToOS	String	Denotes whether key events from RS-485 keyboard are converted to Windows key events. Valid values are: True Enable conversion (other value) Disable conversion Note: This setting cannot be modified with the Configuration tool.
OnlineTimeout	String	Time (in milliseconds) to wait for device to come online. Note: This setting cannot be modified with the Configuration tool.

OPOS DirectIO calls

Get keyboard light status

Returns the current state of the keyboard LEDs

Table 157. Get keyboard light status

Parameter	Type	Description
Command	Int32	200
Data	Long*	Pointer to a byte. On return the byte contains a mask of the current keyboard LED state. See Table 158 for more details.
Obj	BSTR*	Ignored

Table 158. Byte information

Bit	Value	LED
0	1	Wait LED
1	2	Offline LED
2	4	Message Pending LED
3	8	Unlabeled LED

Set keyboard light on

This turns on the POS LEDs on the keyboard. This does not affect the 3 standard keyboard LEDs (Caps Lock, Scroll Lock or Num Lock).

Table 159. Set keyboard light on

Parameter	Type	Description
Command	Int32	201
Data	Long*	Pointer to a byte containing a mask of LEDs to enable
Obj	BSTR*	Ignored

Set keyboard light off

This turns off the POS LEDs on the keyboard. This does not affect the 3 standard keyboard LEDs (Caps Lock, Scroll Lock or Num Lock).

Table 160. Set keyboard light off

Parameter	Type	Description
Command	Int32	202
Data	Long*	Pointer to a byte containing a mask of LEDs to disable
Obj	BSTR*	Ignored

Additional OPOS information

Firmware Update

The OPOS provides a mechanism to update the firmware for Modular keyboards for USB and PS/2 connectivity. The driver release contains the most up-to-date firmware files for these keyboards and the firmware will be automatically updated when system boots. The table below describes Modular keyboard

type, firmware file, and location of the firmware file on the system. To update the firmware, copy the new firmware file at the location specified in the table below and reboot the system.

Table 161. OPOS Modular Keyboards (Device 22, 23, 24)

Modular Keyboard Type	Firmware File	Location on Windows
67-Key without MSR	aip46013.dat	c:\pos\usb
67-Key with MSR	aip46063.dat	c:\pos\usb
Alphanumeric	aip46043.dat	c:\pos\usb
Compact Alphanumeric	aip46093.dat	c:\pos\usb

Table 162. Compact Alphanumeric Keyboard (Device 9)

Keyboard Type	Firmware File	Location on Windows
Compact Alphanumeric Keyboard	aipcnpos.dat	c:\pos\firmware

Configuration file update

The OPOS provides a mechanism to update the configuration file for Modular keyboards for USB and PS/2 connectivity. This is a two step process.

To avoid conflict with OPOS drivers, it is recommended to install this utility on a separate system which does not contain OPOS drivers.

1. Download Configuration Utility and install it.
 - Go to www.toshibacommerce.com.
 - Click **POS Keyboards** under Peripherals.
 - Click **Downloads** under Toshiba Modular POS Keyboards.
 - Click **Modular POS Keyboard & 4820 Keypad/MSR/Keylock Firmware & Configuration Utility**.
 - Install utility.
2. Attach the Modular keyboard to the system. Using the utility, create a configuration file. Depending on the keyboard type, the configuration will assign the correct configuration file name. Verify that this name complies with the table below.
3. Copy the configuration file to the location specified in the table below.
4. Reboot the system.

Table 163. Modular Keyboards (Device 22, 23, 24)

Modular Keyboard Type	Configuration File	Location on Windows
67-Key without MSR	aip46013.cfg	c:\pos\config
67-Key with MSR	aip46063.cfg	c:\pos\config
Alphanumeric	aip46043.cfg	c:\pos\config
Compact Alphanumeric	aip46093.cfg	c:\pos\config

Configuring the Alphanumeric Point of Sale keyboard

If you chose to install your Toshiba Alphanumeric Point-of-Sale (ANPOS) or Compact Alphanumeric Point-of-Sale (CANPOS) keyboard as the system keyboard, you should use the ANPOS utility program to:

1. Define double keys. These key switches can be doubled: 77, 78, 82, 87, 88, 90, 94, 95, 99, 100, 105 to 109, 112 to 123, 125 to 128.
2. Override the keyboard default values for the following settings:
 - Numeric keypad zero (key 94, 99)

- Key click
- POS LEDs initial setting

The ANPOS utility runs automatically at boot time. It uses the sample resource file, `aipsys.res`.

- On Windows, `aipsys.res` is located in the default install directory, `C:\POS`.
- On Linux systems, `aipsys.res` is located in the `/etc` directory.

Note:

1. In the `aipsys.res` file, the keyboard resource names must be qualified with the application name, `aipanpos`, and the device name, `system`. For example:

```
aipanpos.system.keyboardClick: SOFT
aipanpos.system.doubleKey01: 77,82
aipanpos.system.doubleKey02: 90,95
```

See “Modifying USB, RS-485, and PS/2 device behavior” on page 46 for more information on using the resource file.

2. The ANPOS utility records any errors in the file, `aipanpos.log`.
 - On Windows systems, `aipanpos.log` is created in the default log directory, `C:\POS\LOG`.
 - On Linux systems, `aipanpos.log` is created in the `/var/log` directory.
 3. For a system keyboard, the ANPOS utility is used only for:
 - Double key definition
 - LED setting (initial)
 - Key click setting
- The POS LEDs and the key click setting are not supported for the USB System Keyboard.
4. Changes to the `aipsys.res` file will not take effect until the system is rebooted.

SureOne/SurePOS 100 keyboard

The keyboard on the SureOne is a standard PS/2 keyboard. Toshiba OPOS does not implement the UnifiedPOS POS keyboard specification for this keyboard.

Mapping POS keys

To configure the keyboard, perform the following steps:

1. Select a POS keyboard entry and click **Configure**. The keyboard configuration dialog is displayed (see Figure 27 on page 166).

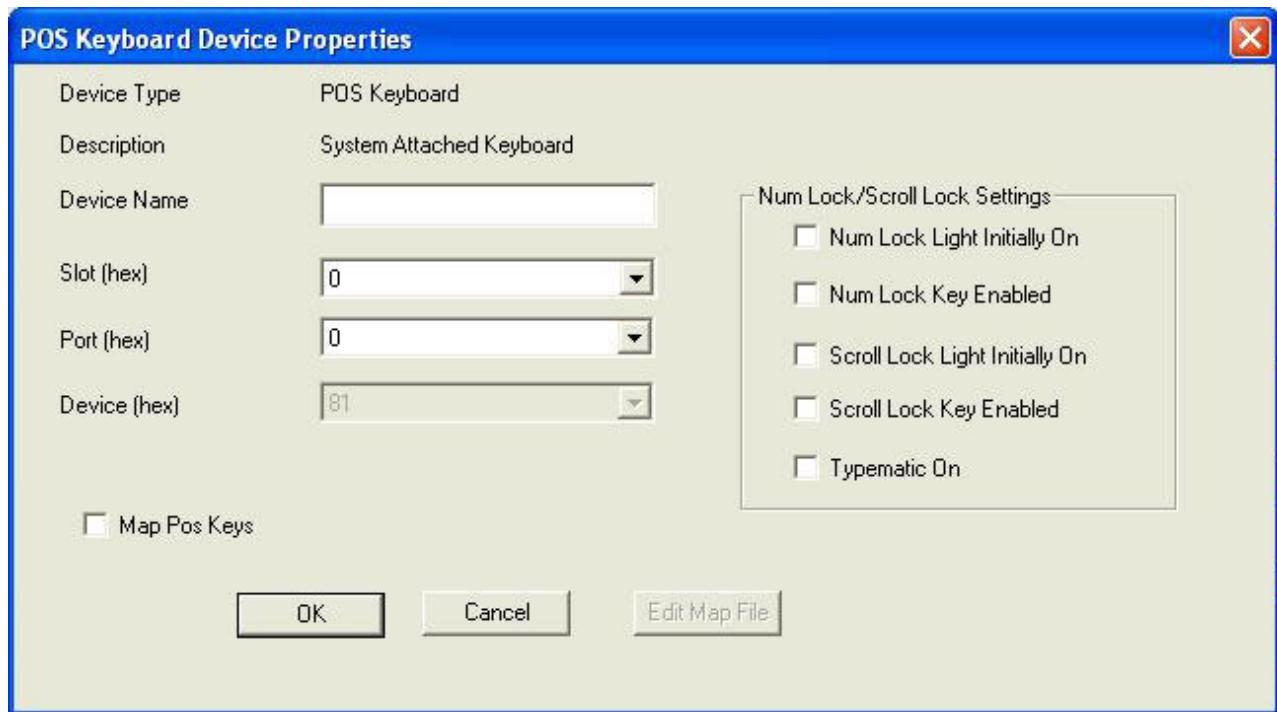


Figure 27. Keyboard properties dialog

Note: There are 2 entries for System Attached Keyboard. For a PS/2 system keyboard, use the entry with Slot value 0; for a USB system keyboard, use the entry with Slot value 8.

2. Select or clear the **Num Lock** and **Scroll Lock** to specify the initial settings you want for those keys.
3. To configure POS keyboard mapping, select **Map Pos Keys** and click the **Edit Map File** button. A filename field is displayed.
4. Enter the name of the POS keyboard mapping file. If the file exists, it is read and the values are displayed in the POS Keyboard Mapping window (see Figure 28 on page 167).

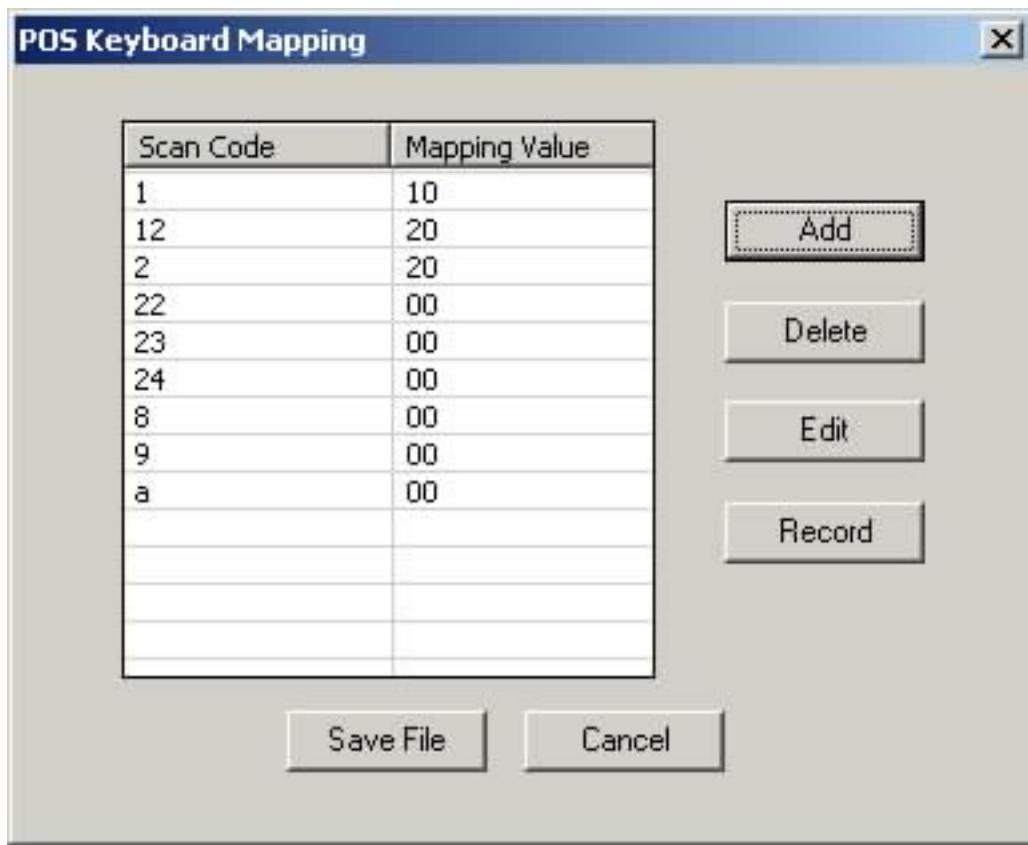


Figure 28. Sample POS keyboard mapping dialog

5. Edit the mapping data.

- To create an entry, click the **Add** button. A new entry is inserted at the top of the list. Enter the scan code and mapping value, then click **Save**.

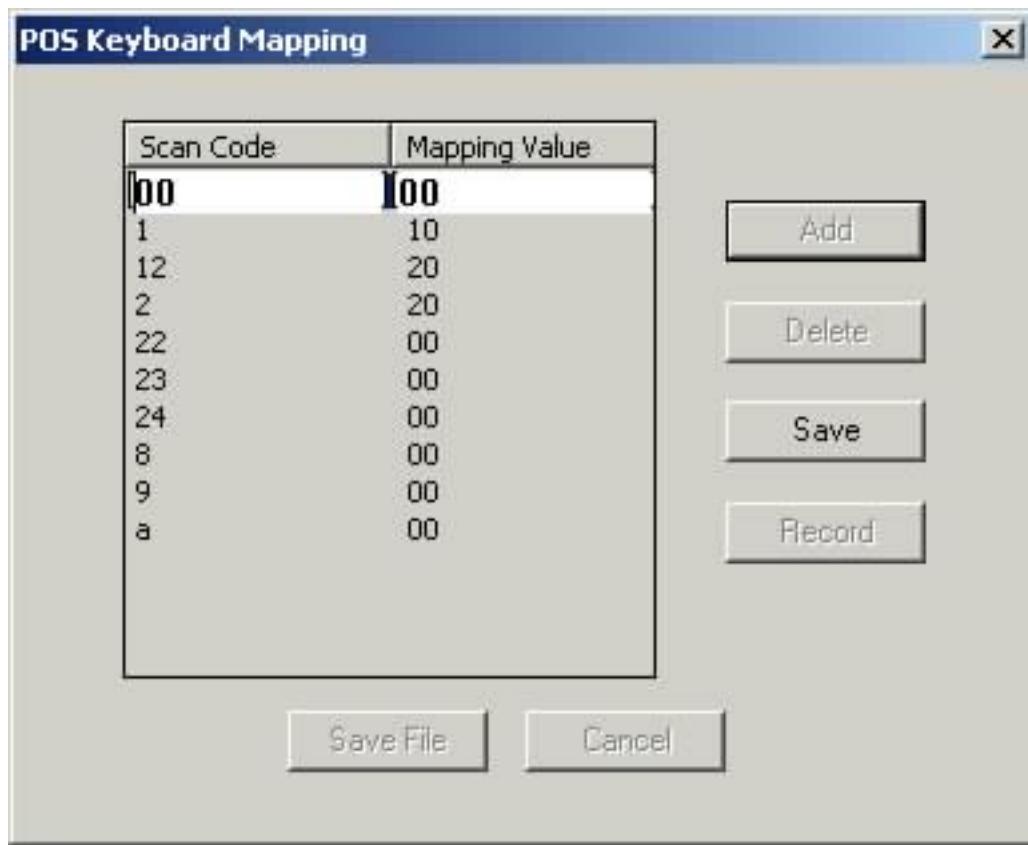


Figure 29. Example of adding a keyboard mapping entry

Note: Non-unique entries are allowed. If multiple entries with the same name exist, only the first entry is used.

- To modify an existing entry, select it and click the **Edit** button. Edit the values and click **Save**.

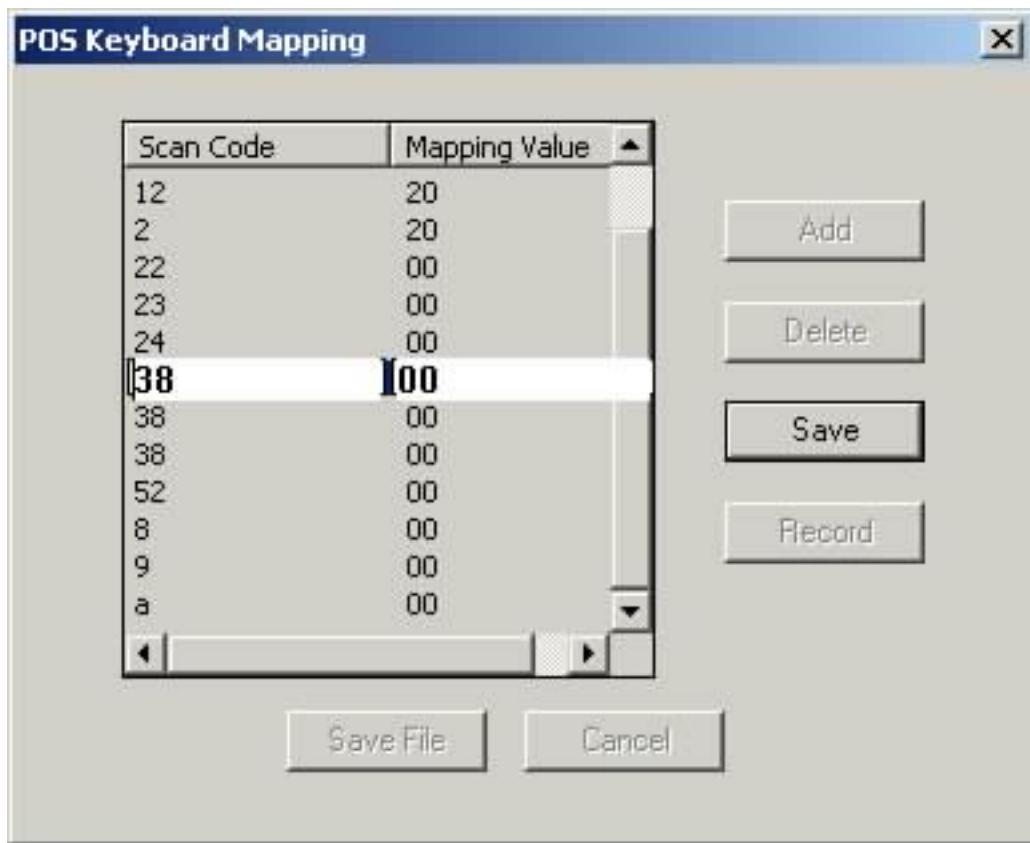


Figure 30. Editing a keyboard mapping entry

- To remove an entry, select it and click **Delete**.
 - To record the scan code of a key, click **Record** and press the key you want to record. An entry is inserted in the list with the scan code of the key you pressed. The dialog continues recording keys until you click the **Stop** button. Use the **Edit** menu to assign mapping values to the new entries.
6. When you have finished editing the keyboard mapping information, click **Save File**. The keyboard mapping file is saved.

Chapter 16. POS printer

Supported devices

Table 164. POS printer supported devices

Device	Connectivity	Comments
1. 4610 printer family	EIA-232	Includes Tx1, Tx2, Tx3, Tx4, Tx5, Tx6, Tx7, Tx8, Tx9 models
2. 4610 printer family	RS-485	Includes Tx1, Tx2, Tx3, Tx4, Tx5, Tx6, Tx7, Tx8, Tx9 models
3. 4610 printer family	USB	Includes Tx1, Tx2, Tx3, Tx4, Tx5, Tx6, Tx7, Tx8, Tx9 models
4. Model 3/4 printer family	RS-485	Includes 4, 4R, 4A
5. Model 3/4 printer family	USB	Printer Protocol Converter is used, include 4, 4R, 4A
6. 4689 printer family (impact)	RS-485	OPOS Only; Includes 001, 002
7. 4689 printer family (thermal)	RS-485	OPOS Only; Includes 301, 3G1, 3M1
8. 4689 printer family (thermal)	RS-485, USB	TD5
9. 4674 built-in printer	RS-485	TD5 integrated
10. SureOne printer (single-head impact)	EIA-232	OPOS only
11. SureOne and SurePOS 100 printer (thermal)	EIA-232	
12. SureOne printer (double-head impact)	EIA-232	OPOS only
13. SureOne printer (A04/A05 impact) and SurePOS 100 printer (impact)	EIA-232	
14. 4610 2xR/1NR	EIA-232	
15. 4610 2xR/1NR	RS-485	
16. 4610 2xR/1NR	USB	
17. 4610 2xR/1NR	Network	

Supported properties and methods

Table 165. POS printer common properties

Property	JavaPOS	OPOS
AutoDisable		Not supported
BinaryConversion	Not supported	All
CapCompareFirmwareVersion	False	1 true only for SST/TI3/TI4/TI5/TI8/TI9 14 - true
CapPowerReporting	PR_STANDARD	PR_STANDARD

Table 165. POS printer common properties (continued)

Property	JavaPOS	OPOS
CapStatisticsReporting	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CapUpdateFirmware	False	1 true only for SST/TI3/TI4/TI5/TI8/TI9 14 - true
CapUpdateStatistics	False (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CheckHealthText		All
Claimed		All
DataCount		Not supported
DeviceControlDescription		All
DeviceControlVersion		All
DeviceEnabled		All
DeviceServiceDescription		All
DeviceServiceVersion		All
FreezeEvents		All
OpenResult	Not supported	All
OutputID		All
PowerNotify		All
PowerState		All
PhysicalDeviceDescription		All
PhysicalDeviceName		All
resultCode	Not supported	All
resultCodeExtended	Not supported	All
State		All

Table 166. POS printer specific properties

Property	JavaPOS	OPOS
AsyncMode		All
CapCharacterSet		All - Values based on HW
CapConcurrentJrnRec	4, 5, 6, 7, 8, 9	4, 5, 6, 7, 8
CapConcurrentJrnSlp		Not supported
CapConcurrentPageMode		Not supported
CapConcurrentRecSlp	1, 2, 3 (TI1-5, TI8-9), 14, 15, 16 (2xR)	1, 2, 3 (TI1-5, TI8-9), 14, 15, 16, 17 (2xR)
CapCoverSensor	All	All except SureOne printers
CapMapCharacterSet	Not supported	1, 2, 3, 4, 5, 7, 8, 10, 11, 12, 13, 14, 15, 16, 17
CapRecPageMode	1, 2, 3, 14, 15, 16, 17	1, 2, 3 - true only for TI4/TI8/TI9, TI5/TI9 14, 15, 16, 17 - true

Table 166. POS printer specific properties (continued)

Property	JavaPOS	OPOS
CapSlpPageMode		Not supported
CapTransaction		All
CapJrnPresent	4, 5, 6, 7, 8, 9	4, 5, 6, 7, 8
CapJrn2Color		Not supported
CapJrnBold	4, 5, 7, 8, 9	4, 5
CapJrnCartridgeSensor		Not supported
CapJrnColor		Not supported
CapJrnDhigh	4, 5, 7, 8, 9	4, 5, 7, 8
CapJrnDwide	4, 5, 7, 8, 9	4, 5, 7, 8
CapJrnDwideDhigh	4, 5, 7, 8, 9	4, 5, 7, 8
CapJrnEmptySensor	4, 5, 6, 7, 8, 9	4, 5, 6, 7, 8
CapJrnItalic		Not supported
CapJrnNearEndSensor	7, 8, 9	7, 8
CapJrnUnderline	7, 8, 9	7, 8
CapRecPresent	All	All
CapRec2Color		1, 2, 3 (TI3-4,Tx6-9), 14, 15, 16, 17 EC > 0x33
CapRecBarCode	1, 2, 3, 7, 8, 9, 11, 14, 15, 16, 17	1, 2, 3, 7, 8, 11, 14, 15, 16, 17
CapRecBitmap		All
CapRecBold	1, 2, 3, 4, 5, 7, 8, 9, 11, 13, 14, 15, 16, 17	1, 2, 3, 4, 5, 10, 11, 13, 14, 15, 16, 17
CapRecCartridgeSensor		Not supported
CapRecColor		1, 2, 3 (TI3-4,Tx6-9), 14, 15, 16, 17 EC > 0x33
CapRecDhigh	1, 2, 3, 7, 8, 9, 11, 13, 14, 15, 16, 17 4, 5 DH forces DW reverts to normal Rotation	1, 2, 3, 7, 8, 11, 13, 14, 15, 16, 17 4, 5 DH forces DW 10 reverts to normal rotation in 180 mode
CapRecDwide	1, 2, 3, 4, 5, 7, 8, 9, 11, 13, 14, 15, 16, 17	1, 2, 3, 4, 5, 7, 8, 10, 11, 12, 13, 14, 15, 16, 17
CapRecDwideDhigh	1, 2, 3, 4, 5, 7, 8, 9, 11, 13, 14, 15, 16, 17	1, 2, 3, 4, 5, 7, 8, 11, 13, 14, 15, 16, 17
CapRecEmptySensor	1, 2, 3, 6, 7, 8, 9, 11 (Only SurePOS 100), 13 (Only SurePOS 100), 14, 15, 16, 17	1,2,3 (2CR, 2NR, 1NR models only) 6, 14, 15, 16, 17
CapRecItalic		Not supported
CapRecLeft90		1, 2, 3, 14, 15, 16, 17
CapRecMarkFeed		Not supported
CapRecNearEndSensor		7, 8, 14, 15, 16, 17 (2xR)
CapRecPapercut	All	All except 10, 12, and 13
CapRecRight90	1, 2, 3, 8, 9, 14, 15, 16, 17	1, 2, 3, 8, 14, 15, 16, 17

Table 166. POS printer specific properties (continued)

Property	JavaPOS	OPOS
CapRecRotate180	1, 2, 3, 11, 14, 15, 16, 17 EC > 0x33	1, 2, 3 EC > 0x33 and 10, 11, 12, 13, 14, 15, 16, 17
CapRecRuledLine		Not Supported
CapRecStamp		1, 2, 3, 11, 13, 14, 15, 16, 17- uses stored bitmap 1 6 - physical stamp 7, 8, 9 - downloaded stamp
CapRecUnderline	1, 2, 3, 7, 8, 9, 11, 13, 14, 15, 16, 17	1, 2, 3, 7, 8, 10, 11, 12, 13, 14, 15, 16, 17
CapSlpPresent	1, 2, 3 (TI1-5, TI8-9), 4, 5, 14, 15, 16, 17 (2xR)	1, 2, 3, 4, 5, 14, 15, 16, 17 (2xR)
CapSlpFullslip		4, 5
CapSlp2Color		Not supported
CapSlpBarCode		1, 2, 3, 14, 15, 16, 17 (2xR) EC > 0x1D
CapSlpBitmap	1, 2, 3 (TI1-5, TI8-9), 4, 5, 14, 15, 16, 17 (2xR)	1, 2, 3, 4, 5, 6, 14, 15, 16, 17 (2xR)
CapSlpBold	1, 2, 3 (TI1-5, TI8-9), 4, 5, 14, 15, 16, 17 (2xR)	1, 2, 3, 4, 5, 14, 15, 16, 17 (2xR)
CapSlpBothSidesPrint		1, 2, 3, (TI1-4, TI8-9), 14, 15, 16, 17 (2CR)
CapSlpCartridgeSensor		Not supported
CapSlpColor		Not supported
CapSlpDhigh		1, 2, 3 4, 5 DH forces DW 14, 15, 16, 17 (2xR)
CapSlpDwide		1, 2, 3, 4, 5, 14, 15, 16, 17 (2xR)
CapSlpDwideDhigh		1, 2, 3, 4, 5, 14, 15, 16, 17 (2xR)
CapSlpEmptySensor		1, 2, 3, 4, 5, 14, 15, 16, 17 (2xR)
CapSlpItalic		Not supported
CapSlpLeft90	1, 2, 3, 14, 15, 16, 17 (2xR)	1, 2, 3 (except TI5 and Tx7), 14,15,16, 17 (2xR - SBCS Only)
CapSlpNearEndSensor	1, 2, 3, 4, 5, 14, 15, 16, 17 (2xR)	1, 2, 3, 4, 5, 6, 14, 15, 16, 17 (2xR)
CapSlpRight90		Not supported
CapSlpRotate180		Not supported
CapSlpRuledLine		Not supported
CapSlpUnderline		Not supported
CartridgeNotify		Not supported
CharacterSet		All
CharacterSetList	See "Character sets supported by POSPrinter devices for JavaPOS" on page 215	See "Character sets supported by POSPrinter devices for OPOS" on page 255

Table 166. POS printer specific properties (continued)

Property	JavaPOS	OPOS
CoverOpen	1, 2, 3, 4, 5, 11, 14, 15, 16, 17 7, 8, 9 - when printer not idle	1, 2, 3, 4, 5, 14, 15, 16, 17 7, 8 - when printer not idle
ErrorLevel		All
ErrorStation		All
ErrorString		All
FlagWhenIdle		All
FontTypefaceList	Not supported	1, 2, 3 (TI3-5, Tx6-9) EC > 0x33 14, 15, 16, 17 can list "Fixed, Proportional" based on downloaded UD Fonts, null string otherwise
JrnCartridgeState		Not supported
JrnCurrentCartridge		Not supported
JrnEmpty	4, 5, 6, 7, 8, 9	4, 5, 6, 7, 8
JrnLetterQuality		Not supported
JrnLineChars	1, 2, 3, 14, 15, 16, 17 - 0 4, 5 - 38 6 - 25 7, 8, 9 - 32	1, 2, 3, 10, 11, 12, 13, 14, 15, 16, 17 - 0 4, 5 - 38 6 - 25 7, 8 - 32
JrnLineCharsList	1, 2, 3, 14, 15, 16, 17 - "" (empty) 4, 5 - "30,38" 6 - "25,30" 7, - "32,42" 8, 9 - "32"	1, 2, 3, 10, 11, 12, 13, 14, 15, 16, 17 - "" (empty) 4, 5 - "30,38" 6 - "25,30" 7, 8- "32,42"
JrnLineHeight	1, 2, 3, 14, 15, 16, 17 - 0 4, 5, 6 - 9 7, 8, 9 - 24	1, 2, 3, 10, 11, 12, 13, 14, 15, 16, 17 - 0 4, 5, 6 - 9 7, 8- 24
JrnLineSpacing	1, 2, 3, 14, 15, 16, 17 - 0 4, 5, 6 - 12 7, - 27 8, 9 - 30	1, 2, 3, 10, 11, 12, 13 - 0 4, 5, 6 - 12 7, 8- 27
JrnLineWidth	1, 2, 3, 14, 15, 16, 17 - 0 4, 5 - 380 6 - 300 7 - 420 8, 9 - 432	1, 2, 3, 10, 11, 12, 13, 14, 15, 16, 17 - 0 4, 5- 380 6 - 300 7, 8 - 420
JrnNearEnd	7, 8, 9	7, 8
MapCharacterSet	Not supported	1, 2, 3, 14, 15, 16, 17

Table 166. POS printer specific properties (continued)

Property	JavaPOS	OPOS
MapMode		All
PageModeArea	4610: When PageModeStation is set "0,0,576,1250" (for TI8/9/2xR/1NR) or "0,0,576,800"; else zero	1,2,3 (TI4/TI8/TI9, TI5/TI9 EC > 0x33) 14, 15, 16, 17 - When PageModeStation is set "0,0,576,1250" (for TI8/9) or "0,0,576,800"; else zero
PageModeDescriptor	4610: PTR_PM_BITMAP PTR_PM_BARCODE PTR_PM_BM_ROTATE PTR_PM_BC_ROTATE PTR_PM_OPAQUE When PageModeStation is set; else zero.	1,2,3 (TI4/TI8/TI9, TI5/TI9 EC > 0x33) 14, 15, 16, 17 PTR_PM_BITMAP PTR_PM_BARCODE PTR_PM_BM_ROTATE PTR_PM_BC_ROTATE PTR_PM_OPAQUE When PageModeStation is set; else zero.
PageModeHorizontalPosition	4610: Zero before setting PageModeStation	1, 2, 3 (TI4/TI8/TI9, TI5/TI9 EC > 0x33) 14, 15, 16, 17 - Zero before setting PageModeStation
PageModePrintArea	4610: When PageModeStation is set, max is PageArea value, else zero	OPOS: 1, 2, 3 (TI4/TI8/TI9, TI5/TI9 EC > 0x33) 14, 15, 16, 17 - When PageModeStation is set, max is PageArea value, else zero
PageModePrintDirection	4610: Initialized to jpos.POSPrinterConst.PTR_PD_LEFT_TO_RIGHT when the device is first enabled	1, 2, 3 (TI4/TI8/TI9, TI5/TI9 EC > 0x33) 14, 15, 16, 17 - Initialized to PTR_PD_LEFT_TO_RIGHT when the device is first enabled
PageModeStation	4610: jpos.POSPrinterConst.PTR_S_RECEIPT	1, 2, 3 (TI4/TI8/TI9, TI5/TI9 EC > 0x33) 14, 15, 16, 17 - PTR_S_RECEIPT
PageModeVerticalPosition	4610: Zero before setting PageModeStation	1, 2, 3 (TI4/TI8/TI9, TI5/TI9 EC > 0x33) 14, 15, 16, 17 - Zero before setting PageModeStation
RecBarcodeRotationList	1, 2, 3, 14, 15, 16, 17 (TI3-5, Tx6-9) EC > 0x33 can list "0,180". Otherwise, 1, 2, 3 list "0". 8, 9 - "0" 14, 15, 16, 17 "0,180"	1, 2, 3 (TI3-5, Tx6-9) EC > 0x33 can list "0,180". Otherwise, 1, 2, 3 list "0".
RecBitmapRotationList		"0"
RecCartridgeState		Not supported
RecCurrentCartridge		Not supported
RecEmpty	1, 2, 3, 6, 7, 8, 9, 14, 15, 16, 17	1, 2, 3, 6, 14, 15, 16, 17
RecLetterQuality		1, 2, 3 (TI3-5, Tx6-9), 14, 15, 16, 17 EC > 0x33

Table 166. POS printer specific properties (continued)

Property	JavaPOS	OPOS
RecLineChars	1, 2, 3 - (TI5,Tx7) - 38, (all other) 44 4, 5, 6, 7 - 38 11 - 48 13 – 40 with narrow paper 28 8, 9 - 32 14, 15, 16, 17 (DBCS) - 38 14, 15, 16, 17 (SBCS) - 44	1, 2, 3 (TI3-5, Tx6-9 EC > 0x33) - 44 1, 2, 3 (TI5,Tx7) - 38 (all other) - 44 4, 5, 6, 7, 8 - 38 10, 11, 12, 13 - 40 14, 15, 16, 17 (DBCS) - 38 14, 15, 16, 17 (SBCS) - 44
RecLineCharsList	1, 2, 3 - EC > 0x33 (TI5,Tx7) - "28,30,32,33,36,38,41,44,48" (TI4, Tx6, TI8, TI9, 2xR/1NR -SBCS) - "32,34,36,38,41,44,48,52,57,64,72" 1, 2, 3 (All other), - "34,44,48" 4, 5 - "30,38" 6 - "25,30" 7 - "32,42" 8, 9, "32" 11- "36, 38, 41, 48" 13 – “22, 33, 40” with narrow paper "15, 23, 28"	1, 2, 3 - EC > 0x33 (TI5,Tx7) - "28,30,32,33,36,38,41,44,48" (TI4, Tx6, TI8, TI9, 2xR/1NR -SBCS) - "32,34,36,38,41,44,48,52,57,64,72" 1, 2, 3 (All other), - "34,44,48" (older models only support "30, 38") 6 - "25, 30" 7, 8 - "32, 42" 10, 13 - "22, 33, 40" 11 - "36, 38, 44, 48" 12 - "33, 40" 14,15,16, 17 (DBCS) - "28,30,32,33,36,38,41,44,48" 14,15,16, 17 (SBCS) - "32,34,36,38,41,44,48,52,57,64,72"
RecLineHeight	1, 2, 3 (TI5,Tx7) - 24, (All other) - 20 4, 5, 6 - 9 7, 8, 9 - 24 11 - 24 13 - 9 14, 15, 16, 17 (DBCS) - 24 14, 15, 16, 17 (SBCS) - 20	1, 2, 3 (TI5,Tx7) - 24, (All other) - 20 4, 5, 6 - 9 7, 8, 9 - 24 11 - 24 13 - 9 14, 15, 16, 17 (DBCS) - 24 14, 15, 16, 17 (SBCS) - 20

Table 166. POS printer specific properties (continued)

Property	JavaPOS	OPOS
RecLineSpacing	1, 2, 3 - 34 4, 5, 6 - 12 7 - 27 8, 9 - 30 11 - 32 13 - 11 14, 15, 16, 17 - 34	1, 2, 3 - 34 4, 5, 6 - 12 7, 8 - 27 10, 12, 13 - 9 11 - 32 14, 15, 16, 17 - 34
RecLinesToPaperCut	1, 2, 3 - 5 4, 5, 7 - 6 8, 9 - 3 6 - 16, 17 11, 13 - 5 14, 15, 16, 17 - 5	1 - 7 2, 3 - 12 4, 5, 7, 8 - 6 6 - 16 10, 11, 12, 13 - 4 14, 15, 16, 17 - 6
RecLineWidth	1, 2, 3 (TI1-5, Tx6-9) - 576 1, 2, 3 Tx6-9 Narrow Paper - 400 4, 5 - 380 6 - 300 7 - 420 8, 9 - 320 11 - 576 13 - 200, with narrow paper 140 14, 15, 16 - 576 14, 15, 16, 17 (1NR) narrow paper- 400	1, 2, 3 (TI1-5, Tx6-9) - 576 1, 2, 3 Tx6-9 narrow paper - 400 4, 5 - 380 6 - 300 7, 8 - 420 10, 12, 13 - 280 11 - 576 14, 15, 16, 17 - 576 14, 15, 16, 17 (1NR) narrow paper- 400
RecNearEnd	1, 2, 3, 7, 8, 14, 15, 16, 17	14, 15, 16, 17
RecSidewaysMaxLines	1, 2, 3 TI1-5, Tx6-9 - 17 1, 2, 3 Tx6-9 Narrow Paper - 12 7, 8, 9 - 14 14, 15, 16 - 17 14, 15, 16, 17 (1NR) narrow paper- 12	1, 2, 3 TI1-2-25, T13-5,Tx9 -17 1, 2, 3 Tx6-7 Narrow Paper - 11 7, 8 - 19 14, 15, 16, 17 - 17 14, 15, 16, 17 (1NR) narrow paper- 12

Table 166. POS printer specific properties (continued)

Property	JavaPOS	OPOS
RecSidewaysMaxChars	1, 2, 3 - 61, TI8/9 – 95 7, 8, 9 – 256 14, 15, 16, 17 – 95	1, 2, 3 TI1-2 - 256, TI3-5 - 53, Tx6-Tx7 - 61, Tx8-Tx9 - 96 7, 8 - 53 14, 15, 16, 17 - 96
RotateSpecial	_NORMAL: 1, 2, 3, 8, 9, 11, 14, 15, 16 _LEFT90: -- _RIGHT90: -- 8, 9 _ROTATE180: 1, 2, 3 (TI3-5, Tx6-9) EC > 0x33, 14, 15, 16, 17	OPOS: _NORMAL: 1, 2, 3, 8, 9, 14, 15, 16, 17 _LEFT90: -- _RIGHT90: -- _ROTATE180: 1, 2, 3 (TI3-5, Tx6-9) EC > 0x33, 14, 15, 16, 17
SlpBarcodeRotationList	"0"	1, 2, 3, 14, 15, 16, 17 - (TI3-5, Tx6-9) EC > 1D has "0"
SlpBitmapRotationList		"0"
SlpCartridgeState		Not supported
SlpCurrentCartridge		Not supported
SlpEmpty		1, 2, 3, 4, 5, 14, 15, 16, 17 (2xR)
SlpLetterQuality	1, 2, 3 (TI3-5, Tx6-9), EC > 0x33 14, 15, 16, 17	1, 2, 3 (TI3-4, Tx6-9, TI8, TI9) EC > 0x33 (not supported on TI5) 14, 15, 16, 17
SlpLineChars	1, 2, 3 - (TI5)- 24, (all other) 47 4, 5 - 86 14, 15, 16, 17 (2xR-DBCS) - 24 14, 15, 16, 17 (2xR-SBCS) - 47	1, 2, 3- (TI5) -24 (all other) - 47 4, 5 - 86 6 - 58 14, 15, 16, 17 (2xR-DBCS) - 24 14, 15, 16, 17 (2xR-SBCS) - 47
SlpLineCharsList	1, 2, 3 - (TI5) - "19, 23, 24, 26, 27, 29", (all other) "30, 37, 38, 42, 47, 52" 4, 5 - "30, 38, 42, 68, 86, 94" — older models only support 30, 38, 68, 86 14, 15, 16, 17 (2xR-DBCS) - "19, 20, 21, 22, 23, 24, 26, 27, 29 14, 15, 16, 17 (2xR-DBCS) FW >= 0x0D.00 - "29, 31, 33, 36, 39, 43, 47, 52, 59" Note 1 14, 15, 16, 17 (2xR-SBCS) - "30, 37, 38, 42, 47, 52"	1 - (TI5,Tx7) - "19, 24" (all other)"37, 47, 52" 2, 3 - (TI5, Tx7) - "23, 24, 26, 27, 29" (all other)"30, 37, 38, 42, 47, 52" 4, 5 - "30, 38, 42, 68, 86, 94" -- older models support only 30, 38, 68, 86 6 - "58, 70" 14 (2xR-DBCS) - "19, 24", (with FW >= 0x0D.00, on DI station and codepage set to 1381), "30, 32, 34, 36, 38, 42, 44, 46, 48, 52, 54, 58" 15, 16, 17 (2xR-DBCS) - "23, 24, 26, 27, 29", (with FW >= 0x0D.00, on DI station and codepage set to 1381), "30, 32, 34, 36, 38, 42, 44, 46, 48, 52, 54, 58" 14 (2xR-SBCS) - "23, 24, 26, 27, 29" 15, 16, 17 (2xR-SBCS) - "30, 37, 38, 42, 47, 52"

Table 166. POS printer specific properties (continued)

Property	JavaPOS	OPOS
SlpLineHeight	1, 2, 3, 14, 15, 16, 17 (2xR) - 7 4, 5 - 9	1, 14 - 4 2, 3, 15, 16, 17 - 7 4, 5, 6 - 9
SlpLinesNearEndToEnd		All - values based on HW
SlpLineSpacing	1, 2, 3, 14, 15, 16, 17 (2xR) - 9 4, 5 - 12	1, 14 - 9 2, 3, 15, 16, 17 - 8 4, 5, 6 - 12
SlpLineWidth	1, 2, 3, 14, 15, 16, 17 (2xR) - 474 4, 5 - 880	1, 14 - 474 2, 3, 15, 16, 17 - 470 4, 5 - 880 6 - 300
SlpMaxLines		All - values based on HW
SlpNearEnd	1, 2, 3, 4, 5, 14, 15, 16, 17 (2xR)	1, 2, 3, 4, 5, 6, 14, 15, 16, 17 (2xR)
SlpPrintSide		1, 2, 3 (TI1–5, TI8–9), 14, 15, 16, 17 (2xR)
SlpSidewaysMaxLines	1, 2, 3, 14, 15, 16, 17 (2xR) - 19	1, 14 - 18 2, 3, 15, 16, 17 - 19
SlpSidewaysMaxChars	1, 2, 3, 14, 15, 16, 17 (2xR) - 148	1, 2, 3, 14, 15, 16, 17 (2xR) - 147

Note:

1. The property `com.ibm.jpos.sdi.config.POSPrinter.EnableSlipDBCSCharCompression` must be enabled in order to show the mentioned List; refer to the “Additional JavaPOS information” on page 123 section for property details.

Table 167. POS printer common methods

Method	JavaPOS	OPOS
checkHealth	INTERNAL = ALL EXTERNAL = ALL INTERACTIVE = Not supported	All
claim		All
clearInput		Not supported
clearOutput		All
close		All
compareFirmwareVersion	Not supported	1 true only for SST/TI3/TI4/TI5/TI8/TI9/2CR/2NR/1NR

Table 167. POS printer common methods (continued)

Method	JavaPOS	OPOS
directIO	For 1, 2, 3 only: X'01' Flip check 8, 9 - X'02' PRINT_DOWNLOADED_BITMAP_ID X'09' DOWNLOAD_NON_PROP_FONT_ID X'23' ERASE_FLASH_MEMORY_SECTOR_ID X'20' SET_STAMP_ID X'25' DOWNLOAD_DBCS_FONT_ID	For 1, 2, 3 only: X'01' Flip check X'10' Write flash memory X'11' Read flash memory X'12' Query flash size X'13' Query maximum records X'14' Set record length X'15' Erase flash memory X'16' Get record length 101 Clear page mode print area 102 Set/get page mode print 103 Set/get horizontal print position (page mode) 104 Set/get vertical print position (page mode) 105 Get page area 106 Set/get page mode station 107 Set/get page mode print area 108 Set/get print direction 201 Set/get MICR read with scan 202 Set/get compression format 203 Scan document 204 Store document 205 Retrieve by index 206 Retrieve by tagname 207 Erase image storage
open	All	
release	All	
resetStatistics	Not supported	
retrieveStatistics	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	Supported: 4610 and SureOne
updateFirmware	Not supported	1 true only for SST/TI3/TI4/TI5/TI8/TI9 14 - true
updateStatistics	Not supported	

Table 168. POS printer specific methods

Method	JavaPOS	OPOS
beginInsertion	1, 2, 3 (except SST), 4, 5 14,15,16, 17 (2xR)	1, 2, 3 (except SST), 4, 5, 6 14, 15, 16, 17 (2xR)

Table 168. POS printer specific methods (continued)

Method	JavaPOS	OPOS
beginRemoval	1, 2, 3 (except SST), 4, 5 14,15,16, 17 (2xR)	1, 2, 3 (except SST), 4, 5, 6 14, 15, 16, 17 (2xR)
changePrintSide	1, 2, 3 except SST and TI5 14,15,16, 17 (2CR)	1, 2, 3 (except SST, TI5), 4, 5, 6 14, 15, 16, 17 (2xR)
clearPrintArea	4610: supported	1, 2, 3 (TI3-4, Tx6-9, TI8, TI9) EC > 0x33 14, 15, 16, 17
cutPaper	1,2,3,7,8,9,14,15,16, 17	All except 11
drawRuledLine	Not supported	
endInsertion	1, 2, 3 (except SST), 4, 5 14,15,16, 17 (2xR)	1, 2, 3 (except SST), 4, 5, 6 14, 15, 16, 17 (2xR)
endRemoval	1, 2, 3 (except SST), 4, 5 14,15,16, 17 (2xR)	1, 2, 3 (except SST), 4, 5, 6 14, 15, 16, 17 (2xR)
markFeed		Not supported
pageModePrint	4610: Control is one of the following: PTR_PM_PAGE_MODE, PTR_PM_PRINT_SAVE, PTR_PM_NORMAL, PTR_PM_CANCEL	
printBarcode	1, 2, 3, 7, 8, 9, 11, 14, 15, 16, 17 (receipt) 1, 2, 3 (TI3-5,Tx6-9) EC>1D (slip), 14, 15, 16, 17	1, 2, 3, 11, 14, 15, 16, 17 (receipt) 1, 2, 3 (TI3-5, Tx6-9) EC>1D (slip) 14, 15, 16, 17
printBitmap	All 8, 9 (receipt)	All
printImmediate		All
printMemoryBitmap		All
printNormal		All
printTwoNormal	All except 11, 13	1, 2, 3, 4, 5, 6, 7, 8, 9, 14, 15, 16, 17
rotatePrint	_NORMAL: All _LEFT90: 1, 2, 3, 14, 15, 16, 17 _RIGHT90: 1, 2, 3, 7, 8, 9, 14, 15, 16, 17(receipt) _ROTATE180: 1, 2, 3, (TI3-5, Tx6-9) EC > 0x (receipt), 11, 13, 14, 15, 16, 17 (receipt)	_NORMAL: All _LEFT90: 1, 2, 3, 14, 15, 16, 17 _RIGHT90: 1, 2, 3, 7, 8, 14, 15, 16, 17 (receipt) _ROTATE180: 1, 2, 3 (TI3-5, Tx6-9) EC > 0x33 2CR, 2NR, 1NR(receipt), 10, 11, 12, 13, 14, 15, 16, 17 (receipt)
setBitmap	1,2,3,4,5,7,8,9,11,13,14,15,16, 17	All
setLogo		All
transactionPrint		All
validateData		All

Table 169. POS printer events

Event	JavaPOS	OPOS
DirectIOEvent	All	1, 2, 3, 14, 15, 16, 17
ErrorEvent		All
OutputCompleteEvent		All
StatusUpdateEvent:		

Table 169. POS printer events (continued)

Event	JavaPOS	OPOS
PTR_SUE_COVER_OK	All	All
PTR_SUE_COVER_OPEN	All	All
PTR_SUE_JRN_COVER_OK	4, 5	Not supported
PTR_SUE_JRN_COVER_OPEN	4, 5	Not supported
PTR_SUE_REC_COVER_OK	1, 2, 3, 4, 5, 11, 14, 15, 16, 17	1, 2, 3, 14, 15, 16, 17
PTR_SUE_REC_COVER_OPEN	1, 2, 3, 4, 5, 11, 14, 15, 16, 17	1, 2, 3, 14, 15, 16, 17
PTR_SUE_SLP_COVER_OK	1, 2, 3 except SST14, 15, 16, 17 except 1NR	1, 2, 3 except SST14, 15, 16, 17 except 1NR
PTR_SUE_SLP_COVER_OPEN	1, 2, 3 except SST14, 15, 16, 17 except 1NR	1, 2, 3 except SST14, 15, 16, 17 except 1NR
PTR_SUE_JRN_EMPTY	4, 5, 8, 9	4, 5, 8, 9
PTR_SUE_JRN_NEAREMPTY	8, 9	8, 9
PTR_SUE_JRN_PAPEROK	4, 5, 8, 9	4, 5, 8, 9
PTR_SUE_REC_EMPTY	8, 9, 14, 15, 16, 17	8, 9, 14, 15, 16, 17
PTR_SUE_REC_NEAREMPTY	8, 9, 14, 15, 16, 17	8, 9, 14, 15, 16, 17
PTR_SUE_REC_PAPEROK	8, 9, 14, 15, 16, 17	8, 9, 14, 15, 16, 17
PTR_SUE_SLP_EMPTY	1, 2, 3 (except SST), 4, 5, 14, 15, 16, 17 (except 1NR)	1, 2, 3 (except SST), 4, 5, 14, 15, 16, 17 (except 1NR)
PTR_SUE_SLP_NEAREMPTY	1, 2, 3, 4, 5 (except SST), 14, 15, 16, 17 (except 1NR)	1, 2, 3 (except SST), 4, 5, 14, 15, 16, 17 (except 1NR)
PTR_SUE_SLP_PAPEROK	1, 2, 3, 4, 5 (except SST), 14, 15, 16, 17 (except 1NR)	1, 2, 3 (except SST), 4, 5, 14, 15, 16, 17 (except 1NR)
PTR_SUE_IDLE	All	All
PTR_SUE_JRN_CARTRIDGE_EMPTY	Not Supported	Not Supported
PTR_SUE_JRN_HEAD_CLEANING	Not Supported	Not Supported
PTR_SUE_JRN_CARTRIDGE_NEAREMPTY	Not Supported	Not Supported
PTR_SUE_JRN_CARTRIDGE_OK	Not Supported	Not Supported
PTR_SUE_REC_CARTRIDGE_EMPTY	Not Supported	Not Supported
PTR_SUE_REC_HEAD_CLEANING	Not Supported	Not Supported
PTR_SUE_REC_CARTRIDGE_NEAREMPTY	Not Supported	Not Supported
PTR_SUE_REC_CARTRIDGE_OK	Not Supported	Not Supported
PTR_SUE_SLP_CARTRIDGE_EMPTY	Not Supported	Not Supported
PTR_SUE_SLP_HEAD_CLEANING	Not Supported	Not Supported
PTR_SUE_SLP_CARTRIDGE_NEAREMPTY	Not Supported	Not Supported
PTR_SUE_SLP_CARTRIDGE_OK	Not Supported	Not Supported
OPOS_CFY_FIRMWARE_OLDER	Not Supported	1 true only for SST/TI3/TI4/TI5/ TI8/TI9, 14

Table 169. POS printer events (continued)

Event	JavaPOS	OPOS
OPOS_CFY_FIRMWARE_SAME	Not Supported	1 true only for SST/TI3/TI4/TI5/TI8/TI9, 14
OPOS_CFY_FIRMWARE_NEWER	Not Supported	1 true only for SST/TI3/TI4/TI5/TI8/TI9, 14
OPOS_CFY_FIRMWARE_DIFFERENT	Not Supported	1 true only for SST/TI3/TI4/TI5/TI8/TI9, 14
OPOS_CFY_FIRMWARE_UNKNOWN	Not Supported	1 true only for SST/TI3/TI4/TI5/TI8/TI9, 14
OPOS_SUE_UF_PROGRESS	Not Supported	1 true only for SST/TI3/TI4/TI5/TI8/TI9, 14
OPOS_SUE_UF_COMPLETE	Not Supported	1 true only for SST/TI3/TI4/TI5/TI8/TI9, 14
OPOS_SUE_UF_COMPLETE_DEV_NOT_RESTORED	Not Supported	1 true only for SST/TI3/TI4/TI5/TI8/TI9, 14
OPOS_SUE_UF_FAILED_DEV_OK	Not Supported	1 true only for SST/TI3/TI4/TI5/TI8/TI9, 14
OPOS_SUE_UF_FAILED_DEV_UNRECOVERABLE	Not Supported	1 true only for SST/TI3/TI4/TI5/TI8/TI9, 14
OPOS_SUE_UF_FAILED_DEV_NEEDS_FIRMWARE	Not Supported	1 true only for SST/TI3/TI4/TI5/TI8/TI9, 14
OPOS_SUE_UF_FAILED_DEV_UNKNOWN	Not Supported	1 true only for SST/TI3/TI4/TI5/TI8/TI9, 14
IBM_JPOS_SUE_PTR_REC_UNEXPECTED_COVER_OPEN (1000):Notify the Receipt cover was opened without an error condition present.	14,15,16,17	Not Supported
IBM_JPOS_SUE_PTR_SLP_UNEXPECTED_COVER_OPEN (10001):Notify the Slip cover was opened without an error condition present.	14,15,16,17	All
IBM_JPOS_SUE_PTR_MAIN_LOGIC_CARD_FAILURE (10002): Notify the Main Logic Card of the printer has a failure.	14,15,16,17	Not Supported
IBM_JPOS_SUE_PTR_INTERFACE_LOGIC_CARD_FAILURE (10003): Notify the Interface Card of the printer has a failure.	14,15,16,17	Not Supported
IBM_JPOS_SUE_PTR_REC_PRINT_HEAD_FAILURE (10004): Notify the Receipt Print Head has a failure.	14,15,16,17	Not Supported
IBM_JPOS_SUE_PTR_SLP_PRINT_HEAD_FAILURE (10005): Notify the Slip Print Head has a failure.	14,15,16,17	Not Supported
IBM_JPOS_SUE_PTR_PAPER_MOTION_SENSOR_FAILURE (10006):Notify the Paper Motion Sensor has a failure.	14,15,16,17	Not Supported

Table 169. POS printer events (continued)

Event	JavaPOS	OPOS
IBM_JPOS_SUE_PTR_REC_CRITICALLY_LOW_PAPER (10007): Notify the paper in the Receipt station is almost out.	14,15,16,17	Not Supported
IBM_JPOS_SUE_PTR_REC_PRINT_HEAD_OVERHEAT (10008): Notify the Receipt Print Head has been overheated.	1,2,3 – (Tx8-Tx9) 14,15,16,17	Not Supported
IBM_JPOS_SUE_PTR_REC_PRINT_HEAD_OK (10009): Notify the Receipt Print Head is ready.	1,2,3 – (Tx8-Tx9) 14,15,16,17	Not Supported
IBM_JPOS_SUE_PTR_REC_CUTTER_JAM (10010): Notify the Receipt station fired a Cutter Jam Event.	14,15,16,17	Not Supported
IBM_JPOS_SUE_PTR_REC_PARTIAL_CUT (10011): Notify the Receipt station fired a Partial Cut Event.	14,15,16,17	Not Supported

Table 170. POS printer BarCode symbologies supported

BarCode Symbology	JavaPOS	OPOS
PTR_BCS_AZTEC		Not supported
PTR_BCS_DATAMATRIX		Not supported
PTR_BCS_UPCA		1, 2, 3, 8, 9, 11, 14, 15, 16,17
PTR_BCS_UPCA_S		Not supported
PTR_BCS_UPCE		1, 2, 3, 8, 9, 11, 14, 15, 16,17
PTR_BCS_UPCE_S		Not supported
PTR_BCS_UPCD1		Not supported
PTR_BCS_UPCD2		Not supported
PTR_BCS_UPCD3		Not supported
PTR_BCS_UPCD4		Not supported
PTR_BCS_UPCD5		Not supported
PTR_BCS_UPDF417		Not supported
PTR_BCS_QRCODE		14, 15, 16,17
PTR_BCS_QRCODE		14, 15, 16, 17 with firmware level ≥ 0E.00. Receipt only.
PTR_BCS_UQRCODE		Not supported
PTR_BCS_EAN8		Not supported
PTR_BCS_JAN8		1, 2, 3, 8, 9, 11, 14, 15, 16,17
PTR_BCS_EAN8_S		Not supported
PTR_BCS_EAN13		1, 2, 3, 8, 9, 11, 14, 15, 16,17
PTR_BCS_JAN13		Not supported
PTR_BCS_EAN13_S		Not supported
PTR_BCS_EAN128		Not supported
PTR_BCS_TF		Not supported
PTR_BCS_ITF		1, 2, 3, 8, 9, 11, 14, 15, 16,17
PTR_BCS_Codabar		1, 2, 3, 8, 9, 11, 14, 15, 16,17
PTR_BCS_Code39		1, 2, 3, 8, 9, 11, 14, 15, 16,17
PTR_BCS_Code93		1, 2, 3, 8, 9, 11, 14, 15, 16,17

Table 170. POS printer BarCode symbologies supported (continued)

BarCode Symbology	JavaPOS	OPOS
PTR_BCS_Code128		1, 2, 3, 8, 9, 11, 14, 15, 16, 17
PTR_BCS_OCRA		Not supported
PTR_BCS_OCRB		Not supported
PTR_BCS_Code128_Parsed		1,2,3,14,15,16, 17
PTR_BCS_GS1DATABAR		14,15,16,17 EC >= 0F (Receipt Station Only)
PTR_BCS_GS1DATABAR_S		14,15,16,17 EC >= 0F (Receipt Station Only)
PTR_BCS_GS1DATABAR_E		14,15,16,17 EC >= 0F (Receipt Station Only)
PTR_BCS_GS1DATABAR_E_S		14,15,16,17 EC >= 0F (Receipt Station Only)
PTR_BCS_PDF417		1,2,3,14,15,16,17 (Receipt only)
PTR_BCS_MAXICODE		Not supported
PTR_BCS_OTHER		Not supported

In the following table, the specific models and stations to which escape sequences can be sent are listed in parentheses () after the type identifier. If none are specified, assume that all models and stations can accept the escape sequence. An application can also determine if an escape sequence can be sent to a particular printer station by calling the validateData method.

Table 171. POS printer escape sequences

Operation	Escape sequence	JavaPOS	OPOS
Alternate color	ESC rC	1, 2, 3 (TI3-5, Tx6-9) EC > 0x33 with Configured setting and correct paper (receipt)	1, 2, 3 (TI3-5, Tx6-9) EC > 0x33, 14, 15, 16, 17 with Configured setting and correct paper
Alternate color (Custom)	ESC #rC	1, 2, 3 (TI3-5, Tx6-9) EC > 0x33 with Configured setting and correct paper (receipt)	1, 2, 3 (TI3-5, Tx6-9) EC > 0x33, 14, 15, 16, 17 with Configured setting and correct paper
Bold	ESC (!)bC	Same as CapXxxBold (For Slp it must be used at the beginning of the data and applies to the entire printed line.)	Same as CapXxxBold
Center	ESC cA		All
Double high	ESC 3C		Same as CapXxxDhigh
Double high & wide	ESC 4C		Same as CapXxxDwideDhigh
Double wide	ESC 2C		Same as CapXxxDwide
Feed and paper cut	ESC #fP	All (receipt), except 13	All
Feed, paper cut, and stamp	ESC #sP	<ul style="list-style-type: none"> • 1, 2, 3, 6, 11 (receipt) Prints downloaded receipt bitmap 1 • 7, 8, 9 (receipt) Prints the stamp stored with the DirectIO SET_STAMP_ID 	1, 2, 3, 6, 7, 8, 14, 15, 16, 17
Feed lines	ESC #lF		All
Feed reverse	ESC #rF	4, 5, 8, 11, 13 - Cannot mix with feed forward (receipt)	4, 5 - Cannot mix with feed forward

Table 171. POS printer escape sequences (continued)

Operation	Escape sequence	JavaPOS	OPOS
Feed units	ESC #uF	All	
Fire stamp	ESC sL	<ul style="list-style-type: none"> 1, 2, 3, 11, 13 - Prints downloaded receipt bitmap 1 8, 9 - Prints the stamp stored with the DirectIO SET_STAMP_ID 	1, 2, 3, 14, 15, 16, 17 - Prints downloaded receipt bitmap 1 6, 7, 8 - Supported Stamp, but not as a single command
Font typeface selection	ESC #fT	Not supported	1, 2, 3 (TI3-5, Tx6-9) EC > 0x33, 14, 15, 16, 17 with downloaded proportional font
Integrated escape	ESC #E	Not supported	Not supported
Italic	ESC iC	Not supported	
Left Justify	ESC lA	All	
Normal	ESC N	All	
Paper cut	ESC #P	All except 10, 12, 13	
Print bitmap	ESC #B	<ul style="list-style-type: none"> 1, 2, 3, 4, 5, 11, 13 7, 8, 9 (receipt) 	1, 2, 3, 4, 5, 7, 8, 10, 11, 12, 13, 14, 15, 16, 17
Print bottom logo	ESC bL	All	
Print in-line barcode	ESC #R	Same as CapXxxBarcode	Same as CapXxxBarcode
Print in-line ruled line	ESC *#dL	Not Supported	
Print top logo	ESC tL	All	
Reverse video	ESC (!)rvC	<ul style="list-style-type: none"> 1, 2, 3 (receipt) 7, 8, 9 	1, 2, 3, 8, 11, 14, 15, 16, 17 (11 reverts to normal rotation in 180 mode)
RGB color	ESC #fC	Not supported	
Right justify	ESC rA	All	
Scale horizontally	ESC #hC	1, 2, 3, 4, 5, 6, 7, 8, 9. Up to 2. Same as CapXxxDwide 1, 2, 3 (TI3-5, Tx6-9) EC > 0x33 Up to 8 (receipt) 11: Up to 6 (receipt) 13: Up to 2	1, 2, 3, 4, 5, 6, 7, 8, 11. Up to 2. Same as CapXxxDwide 1, 2, 3 (TI3-5, Tx6-9) EC > 0x33, 14, 15, 16, 17, Up to 8
Scale vertically	ESC #vC	1, 2, 3, 4, 5, 6. Up to 2 is for slip; up to 8 is for receipt. Same as CapXxxDhigh 1, 2, 3 (TI3-5, Tx6-9) EC > 0x33 Up to 8 (receipt) 8, 9. Up to 2 11: Up to 6 (receipt) 13: Up to 2	1, 2, 3, 4, 5, 6, 7, 8, 11. Up to 2 is for slip; up to 8 is for receipt. Same as CapXxxDhigh 1, 2, 3 (TI3-5, Tx6-9) EC > 0x33, 2CR, 2NR, 1NR Up to 8

Table 171. POS printer escape sequences (continued)

Operation	Escape sequence	JavaPOS	OPOS
Shading	ESC #sC	7	7, 8
Single high and wide	ESC 1C		All
Strike Through	ESC (!)#stC	14,15,16,17 (receipt only)	
Subscript	ESC tbC		Not supported
Superscript	ESC tpC		Not supported
Underline	ESC (!)#uC		Same as CapXxxUnderline

JavaPOS configuration

Arabic code page support

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.EnableArabic" type="Boolean" value="true"/>
```

Values accepted:

- True
- False (default)

When true, the JavaPOS drivers provide support for printing Arabic characters by performing the following actions automatically:

1. Set the POSPrinter.CharacterSet property:
 - “864” for 4610-2xR/1xR printers
 - “101” for the 4610-TI-3/4/5/8/9 TM/F 6/7
2. Perform word shaping for the Arabic characters.
3. Set intercharacter spacing to zero for Arabic characters.
4. Reverse the resulting text after word shaping.

Code page support:

4610 2xR/1NR

These models natively support Arabic code page 864 and 1256.

4610 TI-3/4/5/8/9 TM 6/7

The Arabic printing on these models is supported through the User Defined Arabic font file. Therefore, you must first download the user defined font file “4610cp864Thermal_a.fon” and “4610cp864Impact.fnt” to the location 101.

POS Applications

It is assumed that Windows POS applications use code page 1256 for generating Arabic characters.

Device support:

- 4610 2xR/1NR
- 4610 TI-3/4/5/8/9 TM/F 6/7

Supported since UPOS 1.13.6, with backward compatibility to 1.13.4.

Font Download for 4610-TI-3/4/5/8/9 TM/F 6/7:

- By default, those printers do not have the code page 864; use the following command line instructions to download the font:

1. Use aipfnt46s JavaPOS utility to download the fonts (dual station printers):
 - a. java com.ibm.jpos.util.font.AipFnt46s -f <filePath>/4610cp864Thermal_a.fon 1 <PrinterLogicalName>
 - b. java com.ibm.jpos.util.font.AipFnt46s -f <filePath>/4610cp864Impact.fnt 5 <PrinterLogicalName>
2. Use JavaPOS to download the fonts (single station printers):
 - a. java com.ibm.jpos.util.font.AipFnt46s -f <filePath>/4610cp864Thermal_a.fon 1 <PrinterLogicalName>

The font file is found in the following locations:

- **Windows:** [INSTALL DIR]\fonts
- **Linux:** /usr/share/pos/fonts

AddLineFeed

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.AddLineFeed" type="Boolean" value="True"/>
```

This property determines whether to add a line feed to the end of any text sent in a printNormal, printImmediate, or printTwoNormal command when it does not contain a complete line.

A complete line is defined as:

- A line finishing with the line feed character, for example:
"Complete Line\n"
- A line finishing with an escape sequence that produces a line feed, for example:
"Complete Line ESC|101F"
- A line that completely fills the characters per line for the station in use, for example when RecLineChars=20 and the print text is:
"Line of 20 chars7890"

Valid values are TRUE or FALSE. Default is TRUE.

Bitmap quality

UnifiedPOS standard has defined the RecLetterQuality and SlpLetterQuality properties to control the bitmap and text quality. Prior to Toshiba UnifiedPOS release 1.9.2, drivers supported High Quality bitmap printing only.

For Toshiba UnifiedPOS 1.9.2 and later the driver follows the UnifiedPOS specification. When the LetterQuality property is false, the bitmap will be printed in low quality/high speed mode. When the property true, the bitmap will be printed in high quality mode.

If you want to force bitmaps to always print in high quality mode, you must add the following property in the POSPrinter's JposEntry:

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.BitmapLetterQuality" type="Boolean" value="true"/>
```

Values accepted:

True Use high quality mode always.

False (default)

Use UnifiedPOS RecLetterQuality/SlpLetterQuality setting to control the bitmap quality.

Device support:

Supported since:

UnifiedPOS 1.9.5

Color support

Two-color printing is supported on the 4610 Models TI3, TI4, TI5, TI6, TI8, and TI9 printers that have a microcode EC level of 33 or greater. To enable color printing the JavaPOS configuration entry for the printer must be modified to have the following property:

```
<prop name="colorMode" type="Integer" value="2"/>
```

Any value other than 2 defaults to single color. Color mode should be configured only if color printing is used, because enabling the feature slows down the printer, whether or not color printing is performed.

Default8LPI

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.Default8LPI" type="String" value="False"/>
```

This property selects the font and the spacing needed to allow the user to print 8 lines per inch in the paper.

Valid values are TRUE and FALSE. Default is FALSE.

DefaultLargeFont

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.DefaultLargeFont" type="String" value="False"/>
```

This property selects the font with the largest size available on the printer, typically 12x24 dots on the receipt station and 7x9 dots on slip station.

Valid values are TRUE and FALSE. Default is FALSE.

DocAdvance

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.DocAdvance" type="String" value="50"/>
```

This property selects the number of motor steps to advance the paper on a Document Insert, from the top Document sensor to the first print position.

Valid values are 1 to 255. The default is 50.

Device support: 4610 TI-3/4/5/8/9.

DoubleHighAndDoubleWideIsQuad

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.DoubleHighAndDoubleWideIsQuad" type="String" value="OFF"/>
```

This property enables the conversion of the double wide (1b|2C) and double high (1b|3C) escape sequences to work as double wide and high (1b|4C).

Only supported on 4610 Tx5/Tx7 using a DBCS character set.

Valid values are ON and OFF. Default is OFF.

Enable or Disable Partial Cut Detection

This property enables or disables de partial cut detection on all 1NR printers and only on 2XR printers defined as capable of partial cut detection.

Valid values are TRUE and FALSE. Default is FALSE.

Device support: 4610 2XR/1NR

1. To Enable Partial Cut Detection, locate the aip46mct.cfg file located in the following directory:

Linux: /usr/share/pos/config/aipmct.cfg

Windows: c:\pos\config\ain46mct.cfg

2. Locate the partialCutDetection property and set the value of the property to true.
partialCutDetection=true
3. To disable Partial Cut Detection, set the partialCutDetection property to false.
partialCutDetection=false
4. To broadcast this value to other printers, transfer aip46mct.cfg file to clients to the location specified above and reboot the clients.

Enable or Disable Paper Motion Sensing

Partial cut is supported in all 1xR and in 2xR which have a required hardware change. This function is supported starting from firmware 0D_01.

Valid values are TRUE and FALSE. Default is FALSE.

1. To Enable Paper motion sensing, locate the aip46mct.cfg file located in the following directory:

Linux: /usr/share/pos/config/aip46mct.cfg

Windows: c:\pos\config\ain46mct.cfg

2. Locate the paperMotionDetection property and set the value of the property to true.
paperMotionDetection = true
3. To disable Paper motion sensing, set the paperMotionDetection property to false.
paperMotionDetection = false
4. To broadcast this value to other printers, transfer aip46mct.cfg file to clients to the location specified above and reboot the clients.

Enable or Disable Low Paper Detection

Low Paper Detection can be disabled in all 1xR and 2xR printers.

This function is supported starting from firmware 0C_01.

Valid values are TRUE and False. Default is False.

1. To disable Low Paper Detection, locate the aip46mct.cfg file located in the following directory:

Linux: /usr/share/pos/config/aip46mct.cfg

Windows: c:\pos\config\ain46mct.cfg

2. Locate the lowPaperDetectionDisabled property and set the value of the property to true.
lowPaperDetectionDisabled = true
3. To enable Low Paper Detection, set the lowPaperDetectionDisabled property to false.
lowPaperDetectionDisabled = false
4. To broadcast this value to other printers, transfer aip46mct.cfg file to clients to the location specified above and reboot the clients.
5. The Low paper detection configuration will require a system reboot for the printer to recognize it.

Enable or Disable Low Paper LED

The Low Paper LED indicator can be disabled in all 1xR and 2xR printers.

This function is supported starting from firmware 0C_01.

Valid values are TRUE and False. Default is False.

1. To disable the Low Paper LED indicator, locate the aip46mct.cfg file located in the following directory:
Linux: /usr/share/pos/config/aip46mct.cfg
Windows: c:\pos\config\ain46mct.cfg
2. Locate the lowPaperLEDDisabled property and set the value of the property to true.
lowPaperLEDDisabled = true
3. To enable the Low Paper LED indicator, set the lowPaperLEDDisabled property to false.
lowPaperLEDDisabled = false
4. To broadcast this value to other printers, transfer aip46mct.cfg file to clients to the location specified above and reboot the clients.
5. The Low Paper LED configuration will require a system reboot for the printer to recognize it.

PDF417AspectHeight

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.PDF417AspectHeight" type="String" value="1"/>
```

This property selects the value for the aspect ratio height of the PDF417 barcode.

Valid values are 1 to 9. Default is 1.

PDF417AspectWidth

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.PDF417AspectWidth" type="String" value="2"/>
```

This property selects the value for the aspect ratio width of the PDF417 barcode.

Valid values are 1 to 9. Default is 2.

PDF417ECCLevel

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.PDF417ECCLevel" type="String" value="15"/>
```

A security level is appended to the printed barcode. This property enables scanners to read the bar code even if it has been torn, written on, or otherwise damaged.

You can select a level from 0 to 8 to apply error correction. At level 0, a damaged PDF417 cannot be read, but the damage can be detected. At levels 1 through 8, a PDF417 symbol can still be read, even when damaged. As the error correction level increases, more damage can occur to the symbol and still be read. For values of 9 to 400, the ECC level is assume as a percentage of the code words in the barcode.

Valid values are 0 to 400. Default is 15.

PDF417Truncation

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.PDF417Truncation" type="String" value="OFF"/>
```

This property enables the truncation of the PDF417 barcode. The truncated PDF generates a single width bar symbol to replace the right row indicator and stop pattern. When truncation is enabled, decode performance is slightly degraded in order to allow more data to fit in the image width.

Valid values are ON or OFF. Default is OFF.

OptimizedBaudRate

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.4610optimizedBaudRate" type="Boolean" value="true"/>
```

Values accepted:

- False
- True (default)

Device support:

4610 2xR/1NR

Since: UPOS 1.13.7

Description:

Enable\Disable the optimized baud rate and flow control for EIA232 POSPrinters.

When the property is set to true, the baud rate is set to 115,200 baud rate and flow control to XON/XOFF.

When the property is set to false, the driver uses the values defined on the POSPrinter.

Note:

QRCodeEncodingMode

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.QRCodeEncodingMode" type="String" value="0"/>
```

This property selects the value for the encoding mode for the QR Barcode.

Valid values are 0 to 4. Default is 0.

- **0 Byte** - Encodes lower case letters, text double-byte characters, binary data and 8 bit values.
- **1 Alpha-Numeric Mode** - Alphanumeric mode encodes data from a set of 45 characters.*

They are:

- 10 numeric digits (0 - 9) (byte values 30 HEX to 39 HEX)
- 26 alphabetic characters (A - Z) (byte values 41 HEX to 5A HEX)
- 9 symbols SP, \$, %, *, +, -, ., /, : (byte values 20 HEX, 24 HEX, 25 HEX, 2A HEX, 2B HEX, 2D HEX, 2E HEX, 2F HEX, 3A HEX respectively)
- **2 Numeric Mode** - This mode encodes data from the decimal digit set (0 - 9) (byte values 30 HEX to 39 HEX).*
- **3 Kanji Mode** - This mode encodes Kanji characters.

Note:

1. The amount of data needs to be an even number.
 2. The supported range of Kanji characters are from 8140 HEX to 9FFC HEX and from E040 HEX to EBBF HEX.*
- **4 Extended Channel Interpretation (ECI) Mode** - This mode allows the encoding of character sets different than the default. See QRCodeECIValue property.

- **5 Mixing Mode** - This mode allows the printer to encode data using different modes. Currently works with numeric, alpha numeric, byte, and Kanji characters. In this mode, the printer will decide the type of characters and encode them accordingly.

QRCodeErrorCorrectionLevel

Note: *On this mode, the command will be rejected if a byte data is outside the supported range.

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.QRCodeErrorLevel" type="String" value="0"/>
```

This property selects the value for the error correction level for the QR Barcode.

Valid values are 0 to 3. Default is 0.

- **0 L** - 7% recovery
- **1 M** - 15% recovery
- **2 Q** - 25% recovery
- **3 H** - 30% recovery

The maximum number of characters to be encoded is:

Table 172. Characters to be encoded

Error Correction	Numeric	Alpha Numeric	Byte	Kanji
L (7%)	1000	1000	919	500
M (15%)	1000	1000	701	432
Q (25%)	1000	721	499	308
H (30%)	945	566	393	242

QRCodeECIValue

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.QRCodeECIValue" type="String" value="0"/>
```

This property is read only when the EncodingMode for QR Barcode is set as 04.

Valid values are 0 to 26, Default is 0.

Table 173. Supported CharactersSets as defined on the Extended Channel Interpretations standard:

ECI Value	CharacterSet
0	cp437
1	cp819
2	cp437 with ECI rules
3	cp819
4	cp912
5	cp913
6	cp914
7	cp915
8	cp1089
9	cp813

Table 173. Supported CharactersSets as defined on the Extended Channel Interpretations standard: (continued)

ECI Value	CharacterSet
10	cp916
11	cp920
12	iso8859_10
13	iso8859_11
14	Reserved on ECI standard
15	cp921
16	iso8859_14
17	cp923
18	iso8859_16
19	Reserved on ECI standard
20	cp932
21	cp1250
22	cp1251
23	cp1252
24	cp1256
25	UTF16BE
26	UTF-8

Note: ECI Values 14 and 19 are reserved by the ECI standard and are not accepted as a valid parameter. For detailed information on each code, refer to Chapter 21, “Character sets for terminal printers and displays,” on page 375.

QR Barcode Usage

From the UnifiedPOS Standard, the structure of a printBarcode Method will accept only String objects as a valid data parameter.

For JavaPOS, this means that those String objects must be encoded specifically for the Encoding Mode used so that the driver will decode them with the proper character set.

Depending on the Encoding Mode, the way to create the barcode uses a specific character set. Here are some examples on how to create this String object for each mode:

Byte Mode, Alpha-Numeric, Numeric and Mixed Mode:

The String object must be created from a valid byte array, this byte array should contain only valid values depending on the selected mode.

Kanji Mode:

The String object can be constructed using either the Unicode representation or following the 932 Code Page values (refer to Chapter 21, “Character sets for terminal printers and displays,” on page 375). The used characters must be inside the range specified for Kanji Mode.

ECI Mode:

The String object must be created using the correspondent Character set for the selected ECI Value as established by the ECI standards. Refer to the ECI Value/Characterset table in “QRCodeECIValue” on page 194 property.

Example code:

```

POSPrinter posPrinter = new POSPrinter();
posPrinter.open(args[0]);
//Sending kanji characters through the use of an array
try{
    //Data array which contains the value to send the array to be printed
    String kanjiStr = "\u250f\u2501\u2501\u2501\u2501\u2513";

    //Variable which will contains the encoded characters in kanji mode
    String data = new String(array,"Shift_JIS");
    posPrinter.claim(JposConst.JPOS_FOREVER);
    posPrinter.setDeviceEnabled(true);
    posPrinter.printBarCode(POSPrinterConst.PTR_S_RECEIPT,
                           data,
                           POSPrinterConst.PTR_BCS_QRCODE,
                           100,
                           (int)(posPrinter.getRecLineWidth()*.75),
                           POSPrinterConst.PTR_BC_CENTER,
                           POSPrinterConst.PTR_BC_TEXT_BELOW);
} catch (UnsupportedEncodingException e) {
    e.printStackTrace();
}
finally
{
    posPrinter.close();
}

```

ProportionalFontFixedWidth

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.ProportionalFontFixedWidth" type="String" value="20"/>
```

This property aligns proportional font characters on a fixed width. The value is expressed in printer dots. When this property is used, the proportional characters are treated as fixed characters.

Valid values are 8 to 32. Default is 20.

RecLineCharsFont

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.RecLineCharsFont" type="String" value="<letter>"/>
```

Values accepted:

- "A": To select font A (10x20 dots), (RecLineCharList="32,33,36,38,41,44,48,52,57")
- "B": To select font B (12x24 dots), (RecLineCharList="28,30,32,33,36,38,41,44,48")
- "C": To select font C (8x16 dots), (RecLineCharList="36,38,41,44,48,52,57,64,72")
- "TallA": To select font Tall A (10x24 dots), (RecLineCharList="32,33,36,38,41,44,48,52,57")

Default Value: "32, 34, 36, 38, 41, 44, 48, 52, 57, 64, 72" with mixed fonts

Device support: 4610 TI-3/4/5/6/7/8/9/2xR/1NR

Description: This property allows the user to select a specific resident Font so that the driver can provide a specific RecLineCharsList with values that are only valid for the specified Font.

Note: The property is read at OPEN time. If the property is not added into the XML, no changes to the default values will be performed. If the value provided in the property is invalid, it will be ignored and no changes will be performed. Font Tall A is only supported for 2xR/1NR printers with firmware Level 16.xx.

TranslateCharacter

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.TranslateCharacter.<AsciiHexCharNumber>" type="String" value="<NewHexCharacter>"/>
```

This property replaces the value of the *<AsciiHexCharNumber>* parameter with the value of the *<NewHexCharacter>* parameter. In the text that prints, all the characters that match the AsciiHexCharNumber are replaced by the new character.

Values accepted:

- *<AsciiHexCharNumber>*: 0x00 to 0xFF in Hexadecimal representation. This value represents the character number in the ASCII table to be replaced.
- *<NewHexCharacter>*: The character to be used in place of the ASCII Character.

Examples:

- Replace the character 29 0x1D (also known as group separator) with the character G 0x47:

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.TranslateCharacter.0x1D" type="String" value="0x47"/>
```

- Replace the character K 0x4B with the character N 0x4E;

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.TranslateCharacter.0x4B" type="String" value="0x4E"/>
```

SetCompatibilityMode

```
com.ibm.posj.POSPrinter.SetCompatibilityMode = false
```

This property sets the printer to work in Compatibility Mode (TI3/4 Emulation) when true or in Native Mode when false. This property is only available in the posj.properties file.

Valid values: True(Compatibility), False(Native) - Default

Device supported: 4610-2xR/1NR

SlpLineSpacing

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.SlpLineSpacing" type="String" value="3"/>
```

This property defines the spacing between lines in the SLIP station. Valid values are 1 to 255. The default is 9.

SureOneVersion

```
<prop name="SureOneVersion" type="String" value="Standard"/>
```

SureOneVersion defines the character set to be used by the service implementation of the SureOne printer thermal station. No default value is set, so you must indicate a selection.

To determine which character set is currently installed in the printer, perform an offline test as follows:

1. Turn off the system.
2. Press and hold the paper feed button.

3. Turn on the system and wait for a beep.
4. Release the paper feed button.

Possible values for this property are shown in the following table.

Table 174. SureOne Version property values

Possible Value	Character set
"Standard"	character set 998 (equal to character set 437)
"ChineseTrad"	character set 950
"ChineseSimp"	character set 1381
"Korean"	character set 1361
"Japanese"	character set 932

JavaPOS DirectIO calls

There are several DirectIO commands for downloading fonts to the PosPrinter. The general syntax for DirectIO commands is:

```
Syntax directIO ( command: int32, inout data: int32, inout obj: object ): void {
    raises-exception }
```

The device must be claimed and enabled before invoking the directIO() method.

To access POS printer constants, import: com.ibm.jpos.services.DirectIO.

Download DBCS Font ID Command

Download double-byte character sets (DBCS) and fonts to a double-byte printer.

Table 175. DirectIO.DOWNLOAD_DBCS_FONT_ID

Parameter	Type	Description
Command	Int32	com.ibm.jpos.services.DirectIO.DOWNLOAD_DBCS_FONT_ID
Data	Int32	Not used
Object	Object	com.ibm.jpos.services.sdiibmprinter.DBCSFontInfo. Class that holds the filename of the DBCS font to download.

Remarks

None Defined

Errors A UposException might be thrown when this method is invoked. Some possible values of the exception ErrorCode property are listed in Table 176:

Table 176. UposException error codes

Value	Meaning
E_ILLEGAL	One of the following errors has occurred: <ul style="list-style-type: none"> • This printer does not support DBCS font downloads. • This printer station does not allow DBCS fonts. • Font file type is incorrect.
E_NOEXIST	Font file was not found.
E_FAILURE	Error reading from font file.

Download Non-Proportional Font ID Command

Download non-proportional fonts to the Toshiba 4610 and 4689 printers.

Table 177. DirectIO.DOWNLOAD_NON_PROP_FONT_ID

Parameter	Type	Description
Command	Int32	com.ibm.jpos.services.DirectIO.DOWNLOAD_NON_PROP_FONT_ID
Data	Int32	Not used
Object	Object	com.ibm.jpos.services.sdiibmprinter. DirectIOfontInfo

Remarks for Toshiba 4689 POS Printer

- The *station* field of DirectIOfontInfo is ignored; the downloaded font applies to both the receipt and journal station.
- The *codepage* field of DirectIOfontInfo is ignored.
- For the Toshiba 4689 POSPrinter, non-proportional font characters for receipt/journal stations must be either width 8 and height 12 or width 12 and height 24.

Remarks for Toshiba 4610 POS Printer

- Valid values for the station field of DirectIOfontInfo are:
 - PTR_S_RECEIPT: Customer Receipt (CR) station
 - PTR_S_SLIP: slip (Document Insert or DI) station
- Valid values for the codepage field of DirectIOfontInfo are:
 - 1 – 4: one of the four codepages for the customer receipt (CR) station
 - 5 – 6: one of the two codepages for slip (DI) station
- Non-proportional font characters for the receipt station must have a width between 8 and 16 inclusive and a height between 16 and 32 inclusive; those for the slip station should have a width and height between 7 and 16 inclusive.

After a successful font download, use the characterSet property to retrieve and set the desired code pages for font printing. Table 180 on page 200 shows the usage of the different user-defined character sets.

Errors A UposException might be thrown when this method is invoked. Some possible values of the exception ErrorCode property are listed in Table 178:

Table 178. UposException error codes

Value	Meaning
E_ILLEGAL	One of the following errors has occurred: <ul style="list-style-type: none">• This printer does not support non-proportional fonts downloading.• This printer station does not allow non-proportional fonts.• Incorrect font number.• Incorrect font file type.• Character size not valid.
E_EXIST	Memory sector already has a font, erase the memory sector before downloading a font.
E_NOEXIST	Font file was not found.

Download Prop Font ID Command

Download a proportional font to the Toshiba 4610 printer.

Table 179. DirectIO.DOWNLOAD_PROP_FONT_ID

Parameter	Type	Description
Command	Int32	com.ibm.jpos.services.DirectIO.DOWNLOAD_PROP_FONT_ID
Data	Int32	Not used
Obj	Object	com.ibm.jpos.services.sdiibmprinter. DirectIOfontInfo.

Remarks

Proportional fonts are supported only for the Receipt station. Valid values for the codepage field of DirectIOfontInfo are:

- 1: User-defined (UD) Code Page 1
- 3: UD Code Page 3

Note:

1. Proportional font characters should have width and height between 8 and 32 inclusive.
2. When using proportional fonts, there is not a way to accurately wrap lines since each character may have a different width. You should manage the Line width using the RecLineChars property.

After a successful font download, use the characterSet property to retrieve and set the desired code pages for font printing. Table 180 shows the usage of the user-defined character sets.

Table 180. User-defined (UD) character sets

Character set	Font attribute	CR station	DI station
101	Fixed	UD Code Page 1	UD CodePage 1
	Proportional	UD Code Page 1	N\A
102	Fixed	UD Code Page 2	N\A
	Proportional	N/A	N\A
103	Fixed	UD Code Page 3	UD CodePage 2
	Proportional	UD Code Page 3	N\A
104	Fixed	UD Code Page 4	N\A
	Proportional	N\A	N\A

Errors A UposException might be thrown when this method is invoked. Some possible values of the exception ErrorCode property are:

Table 181. UposException error codes

Value	Meaning
E_ILLEGAL	One of the following errors has occurred: <ul style="list-style-type: none">• This printer does not support proportional fonts downloading.• This printer station does not allow proportional fonts.• Incorrect font number.• Incorrect font file type.• Character size not valid.
E_EXIST	Memory sector already has a font, erase the memory sector before downloading a font.
E_NOEXIST	Font file was not found.

FontInfo Object

The DirectIOFontInfo object is used as input or output to the DirectIO functions for downloading fonts to Toshiba 4610 and 4689 printers. The DirectIOFontInfo object has the structure shown in Table 182.

Table 182. DirectIOFontInfo object structure

Field Name	Type	Description
station	int	Print station identifier: <ul style="list-style-type: none">• POSPrinterConst.PTR_S_RECEIPT• POSPrinterConst.PTR_S_SLIP
codepage	int	Font number. This can have a value of 1 – 6.
filename	string	Font file name. This must be a fully qualified path.

Specific usage of the DirectIOFontInfo object will be documented for each DirectIO function

Get Font Info ID Command

Get Font information. Font information will be returned in the com.ibm.jpos.services.sdiibmprinter.DirectIOFontInfo object.

Table 183. DirectIO.GET_FONT_INFO_ID

Parameter	Type	Description
Command	Int32	com.ibm.jpos.services.DirectIO.GET_FONT_INFO_ID
Data[1]	Int32	com.ibm.jpos.services.DirectIO.GET_FONT_INFO
Obj	Object	com.ibm.jpos.services.sdiibmprinter.PrintGetDirectIOFontInfo

Remarks

None Defined

Errors None Defined

Additional JavaPOS information

The device service formats the entire print line internally. This includes left, center, and right alignments. It also inserts line breaks as necessary for long lines. This implementation approach does not allow a single print line to be built up from multiple printNormal() invocations.

The *LineHeight properties are not modifiable.

It is not possible to mix bold and normal printing on the Slip station of the 4610 printer. The line prints either all normal or all bold, depending on which escape sequence is used first.

Calibration for low paper sense

Low paper sensing on the 4610 printer models 2xR and 1NR is based upon the paper thickness and core size. The user can modify the low paper and critically low sensing settings. These settings determine the amount of paper that remains on the roll and at which time the low-paper and critically low-paper status is sent. The default setting for low paper status is 5 meters and critically low status is 1 meter remaining on the roll.

The user must calibrate the printer to accurately recognize the thickness of the paper supply. The user only needs to calibrate one printer for a specific paper supply. They can then broadcast that calibration constant to all printers that will use the same paper. For details, refer to User's Guide for SureMark Printers Models 2xR/1NR.

1. Find MCT values for low-paper sense by running an offline test on the printer:
 - a. Add a new paper roll to be calibrated.
 - b. Power off the printer.
 - c. Hold down the **CR** button and turn on the printer.
 - d. Release the **CR** button when CR light starts to flashing. A menu is printed.
 - e. Select option 5, **More selections**.
 - f. Select option 1, **MCT listing**.
 - g. Record the values for MCTs 0x26, 0x27, 0x28.
2. Transfer the MCT values to the `aip46mct.cfg` file located in the following directory:
 - Linux: `/usr/share/pos/config/aipmct.cfg`
 - Windows: `c:\pos\config\ainp46mct.cfg`
3. To broadcast this value to other printers, transfer the `aip46mct.cfg` file to clients in the location specified above and reboot the clients.

4610 printer firmware update

The Toshiba JavaPOS driver has the ability to automatically update the printer firmware if the printer is attached to the terminal when the terminal boots. It does not have this ability if the printer is attached to the terminal after the terminal has booted and the operating system has loaded.

In addition, printer firmware can be update manually at any time.

Updated printer firmware update files can be downloaded from the Toshiba website at: www.toshibacommerce.com.

1. Go to: www.toshibacommerce.com.
 - a. Under Peripherals, click **Toshiba SureMark Printer**.
 - b. Under Downloads, select the appropriate link for your printer model.
 - c. From the Search Results displayed, select **Toshiba 4610 SureMark Printer, Microcode Files for Firmware Update**.
 - d. Download the ZIP file for OPOS/JavaPOS. This file contains multiple firmware files; choose the correct one for your printer type.
2. Place the firmware file for your printer type into the correct folder.

Table 184. Firmware update information

Operating system	Location
Windows	\POS\IBMJPOS\res\flash\rs485
Linux	/opt/ibm/javapos/flash/rs485

Note: The folder `rs485` is used by EIA232, RS485, and USB POSPrinter connectivity. For automatic update of the RS485 printer in Windows, see the folder structure in the Automatic Update section.

3. Make sure the printer is attached and restart.

Each model of 4610 printer has a different firmware file associated with it.

Table 185. Toshiba 4610 printer firmware update

Printer Model	Firmware update filename
TI1, TI2	aip46mc.hex
TI3, TI4, TG3, TG4, TF6, TM7	aip46mch.hex
TI5, TM7, TF7	aip46mcd.hex
TI8	aip46ti8.hex
TI9	aip46ti8.hex
2CR, 2NR, 1NR	aip46v4.hex

Note: If the firmware file you select does not match the type of printer you have attached, the program will not update the printer.

Automatic update

When a terminal starts, JavaPOS will check the firmware version and update all devices (including printers) that are defined in the JPOS.XML file if necessary.

In Windows, the firmware update process is controlled by a windows service named:

- Toshiba JavaPOS Flash Utility

For automatic update of an RS485 printer in Windows, the desired firmware files should be placed in \pos\firmware. If they are not placed in this location, the automatic update will not be executed for this printer.

Since the Toshiba JavaPOS Flash Utility service controls the updating of firmware in all devices, not just printers, it is recommended that if you do not want the driver to automatically update the printer firmware then you should delete the firmware update files from the \pos\ibmjpos\flash\rs485 folder.

On Linux, the firmware update process is controlled by a shell script, /etc/init.d/flashdev.

Since the batch controls the updating of firmware in all devices, not just printers, it is recommended that if you do not want the driver to automatically update the printer firmware then you should delete the firmware update files from the /opt/ibm/javapos/flash/rs485 folder.

To confirm the success of the firmware, check the ibmflash.log file under \pos\log in Windows, or /var/log in Linux POS systems.

Manual update

Use this command line utility to manually update the firmware on an Toshiba 4610 printer. The syntax for the utility is:

```
java com.ibm.jpos.util.flash.AipFlash46s [param] [flash_filename]
```

where:

param:

-e: Update the printer microcode

flash_filename: The fully qualified path for the firmware file.

On Linux for example:

```
java com.ibm.jpos.util.flash.AipFlash46s -e /opt/ibm/javapos/flash/aipmch46.hex
```

4610 printer font download

A command utility is also provided to download fonts to the printer. The syntax for the utility is:

```
java com.ibm.jpos.util.font.AipFnt46s [param] [filename] [codepage] [logicalname]
```

where:

param:

- -f: fixed font file
- -p: proportional font file
- -d: dbcs font file (valid only for double-byte printers)

filename: The fully qualified pathname of the font file.

codepage:

- 1: CR (receipt) station Code Page 1
- 2: CR (receipt) station Code Page 2
- 3: CR (receipt) station Code Page 3
- 4: CR (receipt) station Code Page 4
- 5: DI (slip) station Code Page 1
- 6: DI (slip) station Code Page 2
- 0: DBCS font download

For example:

```
java com.ibm.jpos.util.font.AipFnt46s -p /opt/ibm/javapos/fonts/FontFile.fTH 3 POSPrinter4610
```

Using POSPrinter's MICR device

When using a printer with an integrated MICR, it is possible for a multi-threaded application to attempt to talk to both devices at the same time. Unpredictable results can occur. Such applications should provide a locking mechanism to ensure that only one thread at a time talks to these devices.

Handling invalid characters

The first 31 ASCII characters are non-printable. The POSPrinter handles these as strings and prints the hexadecimal value of the character in the string. Table 186 shows the values that the POSPrinter prints for these values.

Table 186. Handling invalid characters

Hex Value	Character	Also known as	JavaPOS print
0x00	NUL	Null	00
0x01	SOH	Start of Heading	01
0x02	STX	Start of Text	02
0x03	ETX	End of Text	03
0x04	EOT	End of Transmission	04
0x05	ENQ	Enquiry	05
0x06	ACK	Acknowledge	06

Table 186. Handling invalid characters (continued)

Hex Value	Character	Also known as	JavaPOS print
0x07	BEL	Bell	07
0x08	BS	Backspace	08
0x09	Tab	Horizontal Tab	*
0x0A	LF	NL line feed, New Line	*
0x0B	VT	Vertical Tab	0B
0x0C	FF	NP form feed, New Page	0C
0x0D	CR	Carriage Return	*
0x0E	SO	Shift Out	0E
0x0F	SI	Shift In	0F
0x10	DLE	Data Link Escape	10
0x11	DC1	Device Control 1	11
0x12	DC2	Device Control 2	12
0x13	DC3	Device Control 3	13
0x14	DC4	Device Control 4	14
0x15	NAK	Negative Acknowledge	15
0x16	SYN	Synchronous Idle	16
0x17	ETB	End of Transmission Block	17
0x18	CAN	Cancel	18
0x19	EM	End of Medium	19
0x1A	SUB	Substitute	1A
0x1B	ESC	Escape	*
0x1C	FS	File Separator	1C
0x1D	GS	Group Separator	1D
0x1E	RS	Record Separator	1E
0x1F	US	Unit Separator	1F

*These values are processed by the POSPrinter.

Escape sequence handling

- The Fire Stamp escape sequence is supported on the 4610 printers. JavaPOS uses the bitmap downloaded as bitmap #1 for the stamp.
- Handling of invalid, unrecognized, or illegal value escape sequences is performed as follows:
 - Sequences that do not start with esc| are passed to the printer unmodified. In this case, validateData() returns JPOS_E_FAILURE.
 - Sequences that start with esc| but are not valid JavaPOS sequences are passed through unmodified. In this case validateData() returns JPOS_E_FAILURE.
 - Sequences that specify a function not supported by the printer are ignored. Note that they are *not* passed to the printer. The validateData() would return JPOS_E_FAILURE.
 - Sequences that include a number that is considered invalid (such as less than 0, or in the case where the number represents a percentage, greater than 100), the device service chooses a number to use.

This is an interpretation of the validateData method in the case when JPOS_E_ILLEGAL is returned. The specification states that the service can select valid alternatives. In this case validateData() would return JPOS_E_ILLEGAL.

Line wrapping on the slip station using rotatePrint method(PTR_RP_LEFT90)

The hardware cannot determine the width of the slip station paper when print is rotated 90 degrees left; it is not possible to perform automatic wrapping in this case. Applications should not assume the printer service provides automatic wrapping before reaching the edge of the paper while in this mode. Set the SlpLineChars value property to a value that causes the print lines to wrap correctly for your document length.

Where printing is rotated 90 degrees on the slip station, only the following escape sequences are supported:

- Top and bottom log
- Double wide
- Left, center, and right alignment

Note: The 4610 hardware is not able to print characters from a font with a size of X x 16 dots on the slip station (document station) while in rotate mode. Examples are larger SBCS impact fonts and the DBCS impact fonts currently supported (9x16 dots).

Bar Code GS1 Databar information

Syntax **printBarcode** (station: int32, data: string, symbology: int32, height: int32, width: int32, alignment: int32, textPosition: int32): void { raises-exception, use after open-claim-enable }

The following table shows accepted argument values.

Table 187. Accepted Argument Values

Parameter	Accepted Value
station	POSPrinterConst.PTR_S_RECEIPT
data	Character string to be bar coded, See remarks for details (Below)
symbology	POSPrinterConst. PTR_BCS_GS1DATABAR POSPrinterConst. PTR_BCS_GS1DATABAR_S POSPrinterConst. PTR_BCS_GS1DATABAR_E POSPrinterConst. PTR_BCS_GS1DATABAR_E_S
height	Ignored
width	Ignored
alignment	POSPrinterConst.PTR_BC_LEFT POSPrinterConst.PTR_BC_CENTER POSPrinterConst.PTR_BC_RIGHT
text position	POSPrinterConst.PTR_BC_TEXT_NONE POSPrinterConst.PTR_BC_TEXT_ABOVE POSPrinterConst.PTR_BC_TEXT_BELOW

GS1 DataBar Omnidirectional and GS1 DataBar Stacked Omnidirectional

Character Set:

- Must begin with an Application Identifier (01).
- Data is 13 or 14 digits. If 13 digits are sent, check digit is calculated by the firm-ware and embedded into the barcode. If 14 digits are sent, firmware will ignore the check digit.

Example:

```
posPrinter.printBarcode( PTR_S_RECEIPT,  
"(01)12345678901234", PTR_BCS_GS1DATABAR,  
0, 0,PTR_BC_CENTER, PTR_BC_TEXT_ABOVE)
```

Invalid Input:

- Data does not start with Application Identifier (01)
- Alpha numeric values.
- Less than 13 digits or more than 14 digits.

GS1 DataBar Expanded and GS1 DataBar Expanded Stacked

Character Set:

- ISO/IEC 646: (0-9) (A-Za-z) ! " % & ' () * + , - . / : ; < = > ? _ (SPACE).
- Max number of digits depends on the AI and the order of the AI. See ISO/IEC specification

The data needs to be formatted following these rules:

- Must begin with an Application Identifier.
- Any Application Identifier needs to be enclosed in parentheses.
- There are different encode modes. The modes are enumerated from the most compressed (1) to the least compressed size (7). In addition, certain AI can only be used in specific modes:
 - Mode 1: (01) & (310x)
 - Mode 2: (01) & (320x)
 - Mode 3: (01), (11), (13), (15), (310x) & (320x)
 - Mode 4: (01) & (392x)
 - Mode 5: (01) & (393x)
 - Mode 6: (01) & any AI
 - Mode 7: any AI
- Mode 1, 2, 3, 4 and 5 need an indicator digit of 9 after (01) and a check digit at the end of Global Trade Item Number (GTIN).
- Mode 6 does not need an indicator, after (01). It does need a check digit.
- Mode 7 accepts any other AI

Example for Mode 1:

```
printBarcode( PTR_S_RECEIPT,  
"(01)90012345678908(3103)001750",  
PTR_BCS_GS1DATABAR,  
1, 1,PTR_BC_CENTER, PTR_BC_TEXT_ABOVE)
```

Firmware will append group separator after the variable length AIs. Depending on the scanner, the group separator most likely will be transmitted as 0x1D on scanner output data.

Invalid input:

Not beginning with a (
AI is not enclose in (and)
AI is not in AI list
Data length is not correct.

Example (15-digit GTIN & 6-digit weight of 1.750kg)
Mode 1 ? (01)90012345678908(3103)001750

Mode 6 ? (01)00012345678908(3103)001750
(mode 1 will produce a smaller printed barcode on the paper because it is more compressed)

Example (15-digit GTIN & 6-digit weight of 1.280lb)
Mode 2 ? (01)90012345678908(3103)001280
Mode 6 ? (01)00012345678908(3103)001280
(mode 2 will produce a smaller printed barcode on the paper because it is more compressed)

Example (serial number)
Mode 7 (21)123456789

Note:

1. The validateData(int station, String data) method will NOT validate the contents of the GS1 Databar inline data parameter. It will still check the values for the station, alignment and text position parameters and throw an exception if any of those fail.
2. Whenever invalid data is sent in an inline with printNormal, printImmediate or printTwoNormal, the barcode will NOT be printed and it will throw an exception in response of the invalid barcode data.

Bitmap printing

To use setBitmap and printBitmap in a non-graphics environment on Linux, the following parameter must be passed to Java:

```
java -Djava.awt.headless=true
```

- The methods printBitmap and setBitmap support JPEG and GIF file formats. Uncompressed Windows bitmaps (.BMP) up to 8 bits per pixel are also supported.
- If the bitmap width exceeds the RecLineWidth or SlpLineWidth, an exception is thrown with ErrorCodeExtended = EPTR_TOOBIG.

Table 188. Bitmap printing

Type of printer	Station	Method	Property	Max height	Size
4610 TI 3/4/5	receipt	printBitmap()	RecLineWidth	None	None
4610 TI 3/4/5	receipt	setBitmap()	RecLineWidth	2040	w*h/8<8K
4610 TI 3/4/5	slip	printBitmap()	SlpLineWidth	None	None
4610 TI 3/4/5	slip	setBitmap()	SlpLineWidth	40	w*h/8<8K

Note: printBitmap() has no height maximum because the bitmap is divided into horizontal slices. The height is physically limited on the slip station to the length of the paper.

Enable DBCS character compression on SLIP (Impact) station

```
<prop name = "com.ibm.jpos.sdi.config.POSPrinter.EnableSlipDBCSCharCompression"  
type= "Boolean" value= "true"/>
```

Values accepted:

False (default)

DBCS Chars will be printed with full width (32 dots).

True DBCS Chars will be printed with half width (16 dots).

Device support:

Only 4610 2xR with Firmware version >= 0x0D.00

Since:

1.13.1

EnableSlipDBCSCharCompression property enable/disable the mechanism to print more DBCS chars per line on the SLIP station.

When the property is false the driver will behave as in previous JavaPOS driver version, printing each DBCS char twice the width of a SBCS char.

When the property is true, the JavaPOS driver will print the half of the number of chars specified in the SlpLineChars property. The DBCS chars will have the same width as the SBCS chars.

In order to modify the behavior the property should be added into the JposEntry.

Note:

1. The compression is only enabled for fonts of 16 x 16 dots.
2. If the property is not added or it has an invalid value, the default value will be used (false).

Check flipping support

While the drivers still support the DirectIO.FLIP_CHECK_ID, the preferred method to flip the check is to use the POSPrinter method:

```
changePrintSide( )
```

PageMode support

JavaPOS supports Toshiba PageMode on Receipt Station only. When in PageMode, use of transactionPrint and rotatePrint are prohibited and will cause an exception to be thrown. When horizontalPosition or verticalPosition is set in PageMode, if the value set is not supported by the printer, the UnifiedPOS driver adjusts the value to the closest supported value.

PageMode properties (default)

- AfterOpen (no station):
 - PageModeArea = ""
 - PageModePrintArea = ""
 - PageModeHorizontalPosition = 0
 - PageModeVerticalPosition = 0
 - PageModePrintDirection = 0
 - PageModeStation = 0
- After setting a valid station:
 - PageModeArea = "576,800" ("576,1250" for Tx8/Tx9)
 - PageModePrintArea = "0,0,0,0"
 - PageModePrintDirection = PTR_PD_LEFT_TO_RIGHT
- After entering PageMode:
 - PageModePrintArea = "0,0,576,800" ("0,0,576,1250" for Tx8/Tx9).
- Setting the PrintDirection: Property may change PageModeHorizontalPosition and PageModeVerticalPostion.
- If the Vertical or Horizontal Position property values are not supported, then they will be set to the closest supported value. If the values are outside of the area described by PageModePrintArea, the driver sets these values to the size of the area described by PageModePrintArea.

Note: The vertical and horizontal position properties must be set before sending any print command. If you set the vertical position near the top edge, then text, bitmaps, and barcodes do not display because they are beyond the PageMode area. Set the vertical position to be greater than the vertical dimension of the image to print.

Several examples below illustrate the use of PageMode printing.

Code:

```
printer.setPageModeStation(POSPrinterConst.PTR_S_RECEIPT);
printer.setPageModeArea("0,0,576,1250");
printer.pageModePrint(jpos.POSPrinterConst.PTR_PM_PAGE_MODE);
printer.printBitmap(POSPrinterConst.PTR_S_RECEIPT,"logo.bmp",
    POSPrinterConst.PTR_BM_ASIS;
    POSPrinterConst.PTR_PM_CENTER);
printer.pageModePrint(jpos.POSPrinterConst.PTR_PM_NORMAL);
```



Figure 31. Partial image, no position setting

In the next example the printing is correct, with position properties set.

Example:

This example illustrates how to print a bitmap and a barcode correctly.

Code:

```
printer.setPageModeStation(POSPrinterConst.PTR_S_RECEIPT);
printer.setPageModePrintArea("0,0,576,400");
printer.pageModePrint(POSPrinterConst.PTR_PM_PAGE_MODE);
printer.setPageModeHorizontalPosition(0);
printer.setPageModeVerticalPosition(150);
printer.printBitmap( POSPrinterConst.PTR_S_RECEIPT,"logo.bmp",
    POSPrinterConst.PTR_BM_CENTER;
    POSPrinterConst.PTR_BM_ASIS);
printer.setPageModeHorizontalPosition(0);
printer.setPageModeVerticalPosition(300);
printer.printBarCode( POSPrinterConst.PTR_S_RECEIPT,"01234567",
    108,
    POSPrinterConst.PTR_BCS_CODE39,
    100, 200,
    POSPrinterConst.PTR_BC_CENTER;
    POSPrinterConst.PTR_BM_CENTER);
printer.pageModePrint(jpos.POSPrinterConst.PTR_BC_TEXT_BELOW;
```



Figure 32. Setting position properties for printing

Code:

```
printer.setPageModeStation(POSPrinterConst.PTR_S_RECEIPT);
printer.setPageModePrintArea("0,0,576,400");
printer.pageModePrint(POSPrinterConst.PTR_PM_PAGE_MODE);
printer.setPageModeHorizontalPosition(0);
printer.setPageModeVerticalPosition(150);
printer.printBitmap( POSPrinterConst.PTR_S_RECEIPT,"logo.bmp",
    POSPrinterConst.PTR_BM_CENTER;
    POSPrinterConst.PTR_BM_ASIS);
printer.pageModePrint(jposPOSPrinterConst.PTR_PM_NORMAL);
```

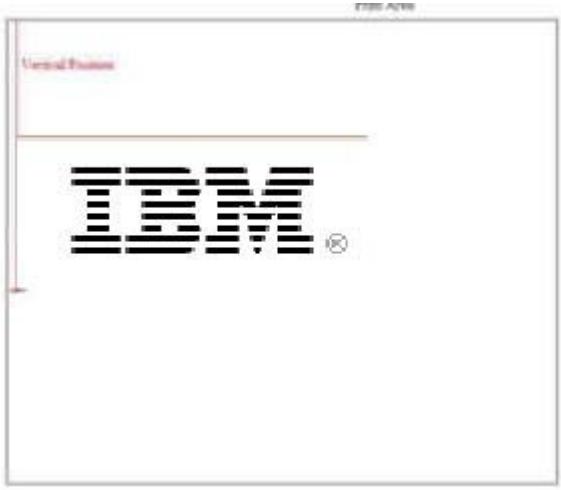


Figure 33. Setting image position for printing

Code:

```
printer.setPageModeStation(POSPrinterConst.PTR_S_RECEIPT);
printer.setPageModePrintArea("0,0,576,400");
printer.pageModePrint(POSPrinterConst.PTR_PM_PAGE_MODE);
printer.printNormalText(POSPrinterConst.PTR_S_RECEIPT,"Text");
printer.printNormalText(POSPrinterConst.PTR_S_RECEIPT,"TextSecondLine");
printer.pageModePrint(POSPrinterConst.PTR_PM_NORMAL);
```

Example:

Not specifying the PageMode Vertical Position/PageMode Horizontal Position results in partial image printing.

Example:

This is another example to illustrate how to print a bitmap.

Example:

This example illustrates printing two lines of code in PageMode.

Printing text:

Begin in the next line.

Text
TextSecondLine

Figure 34. Printing text

When an image follows printed text, the image appears larger than when the image is printed alone, because the image begins in the next line of the text.

Code:

```
printer.setPageModeStation(POSPrinterConst.PTR_S_RECEIPT);
printer.setPageModePrintArea("0,0,576,400");
printer.pageModePrint(POSPrinterConst.PTR_PM_PAGE_MODE);
printer.printNormal(POSPrinterConst.PTR_S_RECEIPT,"TestingPrintText\n");
printer.printBitmap(POSPrinterConst.PTR_S_RECEIPT,"logo.bmp";
    POSPrinterConst.PTR_BM_CENTER;
    POSPrinterConst.PTR_BM_ASIS);
printer.pageModePrint(POSPrinterConst.PTR_PM_NORMAL);
```



Figure 35. Image follows text

Example:

When you print an image or barcode, the following text, image, or barcode prints next to the first.

Code:

```
printer.setPageModeStation(POSPrinterConst.PTR_S_RECEIPT);
printer.setPageModePrintArea("0,0,576,1250");
printer.pageModePrint(jpos.POSPrinterConst.PTR_PM_PAGE_MODE);
printer.printBarcode( POSPrinterConst.PRT_S_RECEIPT,"01234567", 108, 100,
    200, -1, -13);
printer.printBitmap( POSPrinterConst.PTR_S_RECEIPT,"logo.bmp",
    POSPrinterConst.PTR_BM_CENTER;
    POSPrinterConst.PTR_BM_ASIS);
printer.pageModePrint(POSPrinterConst.PTR_PM_NORMAL);
```

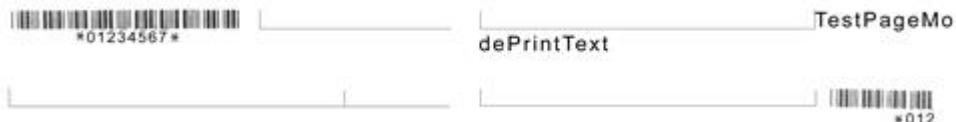


Figure 36. Second image beside first image

SetPrintArea:

If an empty string buffer is sent as a parameter to PageModePrintArea and a get is performed, the current PrintArea is returned. If non-numeric values are used, or the wrong number of values are entered, an exception is thrown. The driver adjusts values to the printer supported values as shown below:

```
if x < 0 then x = 0
if y < 0 then y = 0
if x > MaxWidth - MinWidth then
    x = MaxWidth - MinWidth
if y > MaxHeight - MinHeight
    then y = MaxHeight - MinHeight
if width < MinWidth then width = MinWidth
if height < MinHeight then height = MinHeight
if (width + x) > MaxWidth then width = MaxWidth - x
if (height + y) > MaxHeight then height = MaxHeight - y
```

The values for MinHeight, MaxHeight, MinWidth, and MaxWidth are as follows:

- MinHeight: 6
- MaxHeight: 800 (1250 for TI8/9)
- MinWidthPosition: 15
- MaxWidthPosition: 576

Examples:

- "0,-1,570,600" is adjusted to: "0,0, 570,600"
- "-1,0,800,5000" is adjusted to: "0,0,576,800"

for TI8/9, 0,0,576,1250

- "-1,6,200,8000" is adjusted to: "0,6,200,794" for TI8/9, 0,6,200,1244
- "" gets the current value of PageModePrintArea

Function of the ErrorEvent with response of ER_CLEAR

When the application responds to an ErrorEvent with ER_CLEAR, it clears all outputs that have not been completed. For example:

1. open/ claim/ enable/ async=true
2. printNormal1
3. ErrorEvent on printNormal2
4. printNormal3
5. printNormal4
6. Respond ErrorEvent with ER_CLEAR
7. printNormal5

All the printNormal events up to, and including printNormal4, are cleared in step 6.

Paper saving mode

Paper saving mode is implemented for 1xR/2xR printers and allows you to configure the driver to compress certain aspects of the receipt and thus save paper. Configuration is implemented using 3 different options: Dots Between Lines, Blank Lines Reduction and Barcode Height Reduction. These 3 properties can be separately customized in the relevant JPOS.XML entry.

Dots Between Lines

This setting allows you to explicitly set the number of dots between lines of text. When this property is disabled, the number of dots are calculated by the drivers. When enabled, depending on the option, it will override the calculated values to either be 2 dots or 4 dots between lines. Two dots represents the highest compression; four dots represent a medium compression setting.

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.DotsBetweenLines" type="String" value="0"/>
```

For DotBetweenLines property the valid values are:

- 0 -> Application Settings
- 1 -> 2 Dots Between lines of text
- 2 -> 4 Dots Between lines of text

Blank Line Reduction

This setting will reduce the size of any section composed of only line feeds. When this property is disabled, the size of the line feed behaves as it normally would. When enabled, depending on the option, it will override the default size compressing it either to a 50% or a 75% of the default value.

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.BlankLineReduction" type="String" value="0"/>
```

For BlankLineReduction property the valid values are:

- 0 -> No reduction
- 1 -> 75% Reduction of blank space
- 2 -> 50% reduction of blank space

Barcode Height Reduction

This setting will reduce the vertical size of any barcode. When this property is disabled, the barcode keeps its original size. When enabled, depending on the option, it will override the printer settings and reduce the barcode height by either 25%, 50% or 75% of the original size.

```
<prop name="com.ibm.jpos.sdi.config.POSPrinter.BarcodeHeightReduction" type="String" value="0"/>
```

For BarcodeHeightReduction property, the valid values are:

- 0 -> No reduction
- 1 -> 75% Barcode height reduction
- 2 -> 50% Barcode height reduction
- 3 -> 25% Barcode height reduction

Concurrent MICR/full-image scan

The 4610 Model TI9 performs a concurrent MICR/full-image scan; however, when using a Model TI8 you must do separate passes.

Rec/SlpLineChars behavior

If the parameter for setRec/SlpLineChars method is negative, the method will throw a JPOS_E_ILLEGAL exception, if value is $0 \leq x \leq 10$ the Rec/SlpLineChars property will be set to 10.

Character sets supported by POSPrinter devices for JavaPOS

Table 189. Single-byte character sets supported by POSPrinter devices for JavaPOS

Code Page - SBCS	Device															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
437	✓	✓	✓	✓	✓			✓	✓		✓		✓	✓	✓	✓
775													✓	✓	✓	✓
808																
819	✓	✓	✓										✓	✓	✓	✓
848													✓	✓	✓	✓
850	✓	✓	✓	✓	✓					✓			✓	✓	✓	✓
													Note 1 on page 217			
852	✓	✓	✓							✓			✓	✓	✓	✓
													Note 1 on page 217			
855	✓	✓	✓							✓			✓	✓	✓	✓
													Note 1 on page 217			
857	✓	✓	✓							✓			✓	✓	✓	✓
													Note 1 on page 217			

Table 189. Single-byte character sets supported by POSPrinter devices for JavaPOS (continued)

Code Page - SBCS	Device														
	1	2	3	4	5	6	7	8	9	10	11	12			
858	✓	✓	✓				✓	✓			✓ Note 1 on page 217	✓	✓	✓	✓
860	✓	✓	✓								✓ Note 1 on page 217	✓	✓	✓	
861	✓	✓	✓								✓ Note 1 on page 217	✓	✓	✓	
862	✓	✓	✓						✓		✓ Note 1 on page 217	✓	✓	✓	
863	✓	✓	✓								✓ Note 1 on page 217	✓	✓	✓	
864	✓	✓	✓						✓		✓ Note 1 on page 217	✓	✓	✓	
865	✓	✓	✓								✓ Note 1 on page 217	✓	✓	✓	
866	✓	✓	✓						✓		✓ Note 1 on page 217	✓	✓	✓	
867												✓	✓	✓	
869	✓	✓	✓								✓ Note 1 on page 217	✓	✓	✓	
872												✓	✓	✓	

Table 189. Single-byte character sets supported by POSPrinter devices for JavaPOS (continued)

Code Page - SBCS	Device														
874											✓		✓	Note 1	
897											✓		✓	Note 1	
998	✓	✓	✓	✓	✓			✓	✓		✓		✓	✓	✓
999	✓	✓	✓										✓	✓	✓
1116	✓	✓	✓										✓	✓	✓
1117	✓	✓	✓										✓	✓	✓
1118	✓	✓	✓										✓	✓	✓
1250	✓	✓	✓										✓	✓	✓
1251	✓	✓	✓										✓	✓	✓
1252	✓	✓	✓										✓	✓	✓
1253	✓	✓	✓										✓	✓	✓
1254	✓	✓	✓										✓	✓	✓
1255													✓	✓	✓
1256													✓	✓	✓
1257	✓	✓	✓										✓	✓	✓
Note:															
1. Standard model only															

Table 190. Double-byte character sets supported by POSPrinter devices for JavaPOS

Code Page - DBCS	Device														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
926															
932	✓ Note 6 on page 218	✓ Note 6 on page 218	✓ Note 6 on page 218							✓ Note 1 on page 218		✓ Note 1 on page 218	✓ Note 5 on page 218	✓ Note 5 on page 218	✓ Note 5 on page 218
936															
949	✓ Note 6 on page 218	✓ Note 6 on page 218	✓ Note 6 on page 218									✓ Note 5 on page 218	✓ Note 5 on page 218	✓ Note 5 on page 218	

Table 190. Double-byte character sets supported by POSPrinter devices for JavaPOS (continued)

Code Page - DBCS	Device													
	✓ Note 6	✓ Note 6	✓ Note 6						✓ Note 2		✓ Note 2	✓ Note 5	✓ Note 5	✓ Note 5
950	✓ Note 6	✓ Note 6	✓ Note 6						✓ Note 2		✓ Note 2	✓ Note 5	✓ Note 5	✓ Note 5
951											✓ Note 5	✓ Note 5	✓ Note 5	✓ Note 5
1361									✓ Note 3		✓ Note 3	✓ Note 5	✓ Note 5	✓ Note 5
1381	✓ Note 6	✓ Note 6	✓ Note 6						✓ Note 4		✓ Note 4	✓ Note 5	✓ Note 5	✓ Note 5
Only on:														
1. Japanese version														
2. Traditional Chinese version														
3. Korean version														
4. Simplified Chinese version														
5. Available after Font Download														
6. Available after Font Download on TI5/Tx7 models														

OPOS configuration

Table 191. Service object settings for SureOne devices (POS printer)

Keyword	Type	Description
ThermalPrinter	String	Whether printer is impact or thermal. Valid values are: 0 Impact (default) 1 Thermal Note: Applies only to SureOne/SurePOS 100.
NarrowPaper	String	Whether paper is normal or narrow width. Valid values are: 0 Normal (default) 1 Narrow Note: Applies only to SureOne/SurePOS 100.
BitmapResolution	String	Bitmap resolution (for thermal printers only). Valid values are: 0 Low resolution (default) 9 High resolution Note: Applies only to SureOne/SurePOS 100.
PrinterModel	String	Specifies the model of printer in use. Valid values are: 0 Single-head impact (default) 1 Double-head impact 2 Thermal 3 A04/A05 impact Note: Applies only to SureOne/SurePOS 100.

Arabic code page support

OPOS provides support for printing Arabic characters by performing the following actions when the codepage 864 or 1256 is selected:

1. Performs Arabic character word shaping (character transformation depending on the position of the character in a word).
2. Removes the intercharacter spacing between Arabic characters.

The driver also allows default setting of the code page (CharacterSet property) at device open via the following registry:

HKEY_LOCAL_MACHINE\SOFTWARE\OLEforRetail\ServiceOPOS\PosPrinter

Keyword	Type	Description
DefaultCodePage	REG_DWORD	Setting of CharacterSet property automatically from this registry entry when the device is opened.

The driver assumes that the input characters have been reversed by the application and the beginning character is at the right end of the word. The driver can also perform the Arabic phrase reversing when the following registry is set:

HKEY_LOCAL_MACHINE\SOFTWARE\Toshiba\Point Of Sale Subsystem

Keyword	Type	Description
ReverseArabicPhrases	REG_SZ	TRUE or true: Enable the reversing of Arabic phrases. All other values or if not exist: function is disabled.

Arabic phrase reversing rules:

- An Arabic phrase is a text portion made up of only Arabic text and single spaces.
- The beginning of an Arabic phrase is defined by the parser encountering an Arabic character.
- The ending of an Arabic phrase is reached when any of the following conditions occur:
 - The next character is a non-Arabic character (except a single space).
 - The next two characters are spaces.
 - The next character is a line feed or carriage return.
- Escape sequences will break an Arabic phrase.

Supported device and code page support:

- 4610 2xR/1NR:
 - These models natively support Arabic code page 864 and 1256.
- 4610 TI-3/4 TM 6:
 - The following font file must be downloaded to the printer first for the Arabic code page support: [INSTALL DIR]\fonts\4610cp864a.fon
 - The font can be downloaded with [INSTALL DIR]\bin\ainfnt46.exe tool with the following command line: ainfnt46 -s8 -p11 -f[INSTALL DIR]\fonts\4610cp864a.fon -c1

Unsupported settings

The following settings are *unsupported and untested* options used to modify Service Object behavior only in very limited cases. Use them only when directed to do so by Toshiba Support.

Note: These settings cannot be modified with the Configuration tool. Some settings may only be valid for RS-485/USB devices (POS printer).

Table 192. Service Object settings for POS printer device, unsupported

Keyword	Type	Description
AsyncBufferDelay	String	Number of milliseconds Async Thread will wait for Asynchronous request to be added to the queue before processing begins. The default value is 100.
AsyncBlockChase	String	Whether the printer driver waits for confirmation that the Asynchronous Block of data has been printed before continuing with print commands. Valid values are: ON Wait (default) OFF Do not wait
AsyncCarriageReturn	String	Whether a carriage return character is added after each print command in Async Mode. Valid values are: ON Add carriage return OFF Do not add carriage return (default)
SyncPrintWithoutWait	String	Whether the printer driver waits for confirmation that the synchronous data has been printed before continuing with print commands. Valid values are: False Wait (default) True Do not wait
ChineseDBCS	String	4610 only. Enables the use of bitmap printing to support DBCS characters. The character map is downloaded to the subsystem and converted to bitmap at print time. (No longer supported due to the availability of models TI5 and Tx7.) Valid values are: ON Enabled OFF Disabled (default)
StripDBLineFeeds	String	Whether carriage-return and line-feed characters are stripped out when in Chinese Double-Byte mode. Valid only when the ChineseDBCS setting is ON. Valid values are: ON Strip CR and LF OFF Do not strip (default)
OnlineTimeout	String	Time in milliseconds to wait for device to come online. Only applies to RS485/USB printers.
BarcodeScaleHeightCodabar	String	Value to scale height of Codabar barcodes. Only applies to RS485/USB printers.
BarcodeScaleHeightCode128	String	Value to scale height of Code128 barcodes. Only applies to RS485/USB printers.
BarcodeScaleHeightITF	String	Value to scale height of ITF barcodes. Only applies to RS485/USB printers.
BarcodeScaleHeightJAN13	String	Value to scale height of JAN-13 barcodes. Only applies to RS485/USB printers.
BarcodeScaleHeightJAN8	String	Value to scale height of JAN-8 barcodes. Only applies to RS485/USB printers.
BarcodeScaleHeightUPCA	String	Value to scale height of UPC-A barcodes. Only applies to RS485/USB printers.

Table 192. Service Object settings for POS printer device, unsupported (continued)

Keyword	Type	Description
BarcodeScaleHeightUPCE	String	Value to scale height of UPC-E barcodes. Only applies to RS485/USB printers.
BarcodeScaleWidthCodabar	String	Value to scale width of Codabar barcodes. Only applies to RS485/USB printers.
BarcodeScaleWidthCode128	String	Value to scale width of Code128 barcodes. Only applies to RS485/USB printers.
BarcodeScaleWidthCode39	String	Value to scale width of Code39 barcodes. Only applies to RS485/USB printers.
BarcodeScaleWidthITF	String	Value to scale width of ITF barcodes. Only applies to RS485/USB printers.
BarcodeScaleWidthJAN13	String	Value to scale width of JAN-13 barcodes. Only applies to RS485/USB printers.
BarcodeScaleWidthJAN8	String	Value to scale width of JAN-8 barcodes. Only applies to RS485/USB printers.
BarcodeScaleWidthUPCA	String	Value to scale width of UPC-A barcodes. Only applies to RS485/USB printers.
BarcodeScaleWidthUPCE	String	Value to scale width of UPC-E barcodes. Only applies to RS485/USB printers.
Default40CPL	String	Set the receipt to default to 40 characters per line. Does so by setting InterCharSpacingReceipt12Cpi to 2 dots. Valid values are TRUE and FALSE. Default is FALSE. Only applies to RS485/USB printers.
DocAdvance	String	Sets the number of motor steps from the top Document sensor to the first print position. Valid values are 1 to 256. Default is 50. Only applies to RS485/USB printers.
InterCharSpacingDocument12Cpi	String	Sets the number of printer dots between characters when the Slip station is set to 12 characters per inch. Only applies to RS485/USB printers.
InterCharSpacingDocument15Cpi	String	Sets the number of printer dots between characters when the Slip station is set to 15 characters per inch. Only applies to RS485/USB printers.
InterCharSpacingDocument17Cpi	String	Sets the number of printer dots between characters when the Slip station is set to 17 characters per inch. Only applies to RS485/USB printers.
InterCharSpacingDocument20Cpi	String	Sets the number of printer dots between characters when the Slip station is set to 20 characters per inch. Only applies to RS485/USB printers.
InterCharSpacingReceipt12Cpi	String	Sets the number of printer dots between characters when the Receipt station is set to 12 characters per inch. Only applies to RS485/USB printers.
InterCharSpacingReceipt15Cpi	String	Sets the number of printer dots between characters when the Receipt station is set to 15 characters per inch. Only applies to RS485/USB printers.
InterCharSpacingReceipt17Cpi	String	Sets the number of printer dots between characters when the Receipt station is set to 17 characters per inch. Only applies to RS485/USB printers.
InterCharSpacingReceipt20Cpi	String	Sets the number of printer dots between characters when the Receipt station is set to 20 characters per inch. Only applies to RS485/USB printers.

Table 192. Service Object settings for POS printer device, unsupported (continued)

Keyword	Type	Description
LegacyDIMode	String	In prior releases, the OPOS_SUE_PTR_PAPEROK was fired once the document was registered, not merely detected by the front sensor. This setting allows customers to maintain this behavior. Valid values are TRUE and FALSE. Default is FALSE. TRUE means the OPOS_SUE_PTR_PAPEROK event is fired when the paper is detected at the front sensor. FALSE means the OPOS_SUE_PTR_PAPEROK event is fired when the paper is registered. Only applies to RS485/USB printers.
LegacyRS232Mode	String	In prior releases, the OPOS_SUE_PTR_NEAREND was fired when the document was detected only by the Ready and Front Sensor. This behavior was specific to the RS232 printer. Valid values are TRUE and FALSE. Default is FALSE. TRUE means the OPOS_SUE_PTR_NEAREND event is fired when the document is positioned in the front. FALSE means that OPOS_SUE_PTR_NEAREND event is not fired when the document is positioned in the front. Only applies to RS485/USB printers.
RecLineSpacing	String	Sets the default RecLineSpacing. Default is 34 (dots). Only applies to RS485/USB printers.
TransactionHold	String	Determines whether to hold/release the buffer internal to the printer during a TransactionPrint to the receipt. Valid values are TRUE and FALSE. Default is FALSE. Only applies to RS485/USB printers.
BitmapResolution	String	Indicates bitmap resolution on SurePOS 100/SureOne thermal printers. Valid values are 0 (low) and 9 (high) only. Default is 9.
SetCompatibilityMode	String	This property allows the 2xR, 1NR printer to remain in compatibility mode even when driver is updated. Valid values: True, False (default). Note: <ol style="list-style-type: none">1. If no property is found, the default value is used.2. If the property is present but it has an invalid value, then the default value is used.
ReceiptInterCharacterSpacingUDFx DocumentInterCharacterSpacingUDFx	DWORD	This property sets the Inter Character Spacing of User Defined Font (UDF) with the Character Set between 101 to 104. ReceiptInterCharacterSpacingUDFx denotes the setting for Receipt (CR) station; DocumentInterCharacterSpacingUDFx denotes the setting for Document (DI) station; where x is a value from 1 to 4 which translates for the setting of 12, 15, 17, 20 characters per inch (CPI) respectively. Valid values: 0 to 7 in decimal representation, where 0 denotes No Inter Character Spacing. Note: This setting cannot be modified with the Configuration tool. If no property is found or if the property is present but the value is not within the valid range, no setting will take place.

Table 192. Service Object settings for POS printer device, unsupported (continued)

Keyword	Type	Description
IncreaseRecLeftMargin	String	<p>Increase the left margin of the Receipt Station. Valid with RecLineChars up to 42 characters.</p> <p>Valid Values: 0 to 9. If the property is present but it has a value that is not in the range of values, it is matched to the closest parameter.</p> <p>Only applies to RS232 printers.</p>
MapToPrintNormal	String	<p>Redirect the PrintImmediate function call to PrintNormal during asynchronous mode.</p> <p>Valid Values:</p> <p>True: Enabled</p> <p>Other, Empty: Disabled (Default)</p> <p>Only applicable to USB 1xR/2xR printers.</p>

Supported settings

Note: Not all of these settings can be modified with the Configuration tool. Some settings may only be valid for EIA-232 attached SureMark devices (POS printer).

Table 193. Service Object settings for SureMark devices (POS printer), supported

Keyword	Type	Description
BitmapSpeedQuality	String	<p>Speeds up bitmap downloading. Valid values are:</p> <p>False (default)</p> <p>True</p>
BitmapLetterQuality	String	<p>Enhances the bitmap letter quality (line thickness). Valid values are:</p> <p>False (default)</p> <p>True</p> <p>Please note that if this setting is present, it takes precedence over RecLetterQuality (UPOS POS Printer property).</p>

Table 193. Service Object settings for SureMark devices (POS printer), supported (continued)

Keyword	Type	Description
PDF417ECCLevel	String	<p>This property adds on the barcode a printed security level to enable scanners to read the bar code even if it has been torn, written on, or damaged in other ways. Error correction is specified by selecting a level from 0 to 8. At level 0, a damaged PDF417 cannot be read, but the damage can be detected. At levels 1 through 8, a PDF417 symbol can still be read, even when damaged. As the error correction level increases, more damage can occur to the symbol and still be read. For values of 9 to 400, the ECC level is assumed as a percentage of the code words in the barcode.</p> <p>Valid values: 0 to 400 in decimal representation.</p> <p>Note:</p> <ol style="list-style-type: none"> 1. This setting cannot be modified with the Configuration tool. 2. If no property is found, the default value is used: 15. 3. If the property is present but the value is not a number, the default value is used. 4. If the property is present but has a value not in the range of values, it is matched to the closest parameter.
PDF417AspectHeight	String	<p>This property selects the value for the aspect ratio height of the PDF417 barcode.</p> <p>Valid values: 1 to 9 in decimal representation.</p> <p>Note:</p> <ol style="list-style-type: none"> 1. This setting cannot be modified with the Configuration tool. 2. If no property is found, the default value is used: 1. 3. If the property is present but the value is not a number, the default value is used. 4. If the property is present but has a value not in the range of values, it is matched to the closest parameter.
PDF417AspectWidth	String	<p>This property selects the value for the aspect ratio width of the PDF417 barcode.</p> <p>Valid values: 1 to 9 in decimal representation.</p> <p>Note:</p> <ol style="list-style-type: none"> 1. This setting cannot be modified with the Configuration tool. 2. If no property is found, the default value is used: 2. 3. If the property is present but the value is not a number, the default value is used. 4. If the property is present but has a value not in the range of values, it is matched to the closest parameter.

Table 193. Service Object settings for SureMark devices (POS printer), supported (continued)

Keyword	Type	Description
PDF417Truncation	String	<p>This property enables the truncation of the PDF417 barcode. Truncated PDF generates a symbol with the right row indicator and stop pattern replaced by a single width bar. There is a slight degradation in decode performance if truncation is enabled, which allows more data to fit in the image width.</p> <p>Valid values:</p> <ul style="list-style-type: none"> ON (Enables truncation) <p>Other value (Disables truncation, default)</p> <p>Note:</p> <ol style="list-style-type: none"> 1. This setting cannot be modified with the Configuration tool. 2. If no property is found, the default value is used.
TranslateCharacter	Key	<p>Key values for translating one character to another. This is a key folder in the Registry in which entries that can replace the characters are stored. See TranslateCharacter\<X> for details about the entries that can be added inside this key.</p> <p>Note: This setting cannot be modified with the configuration tool.</p>
TranslateCharacter\<X>	String	<p>The one-byte hexadecimal value of the character to translate. For example, when you translate char "E" (0x45) to char "e" (0x65), the entry name is 45 and the entry value is 65. Write <i>only</i> the number in the entry.</p> <p>Note: This setting cannot be modified with the Configuration tool.</p>
CapRec2Color	String	<p>When color thermal paper is loaded in the SureMark printer, the valid values are:</p> <ul style="list-style-type: none"> N Color paper not loaded (default) Y Color paper loaded
ProportionalFontFixedWidth	String	<p>This property aligns proportional font characters on a fixed width. The value is expressed in printer dots. If this property is used the proportional characters are treated as fixed characters.</p> <p>Valid values: 8 to 32 in decimal representation.</p> <p>Note:</p> <ol style="list-style-type: none"> 1. This setting cannot be modified with the Configuration tool. 2. If no property is found, the default value is used: 20. 3. If the property is present, but the value is not a number, the default value is used. 4. If the property is present but has a value not in the range of values, it is matched to the closest parameter.
PrinterModel	String	<p>4689 only. Specifies the model of 4689 printer in use. Valid values are:</p> <ul style="list-style-type: none"> • 4689-TD5 • 4689-3M1 • 4689-3G1 • 4689-TD5(integrated into 4674) • 4689-TG1(integrated into 4674)

Table 193. Service Object settings for SureMark devices (POS printer), supported (continued)

Keyword	Type	Description
PersistantBitmaps	String	<p>Enables storage of downloaded bitmap in the registry. Valid values are:</p> <p>False Disabled (default). Service Object must download the bitmap each time the driver is opened.</p> <p>True Enabled. Downloaded bitmap is stored in the registry.</p> <p>Note: This setting cannot be modified with the Configuration tool.</p>
ErrorOnCoverOpen	String	<p>Fired Error Event when cover is opened.</p> <p>Valid values are:</p> <p>False Do not fire event (default)</p> <p>True Fire event</p> <p>Note: This setting cannot be modified with the Configuration tool.</p>
LegacyRS-232Mode	String	<p>Processes same as prior to release 1.7.1. Valid values are:</p> <p>OFF Default</p> <p>ON</p> <p>Note: This setting cannot be modified with the Configuration tool.</p>
LegacyDIMode	String	<p>Processes DI same as prior to release 1.7.1. Valid values are:</p> <p>OFF Default</p> <p>ON</p> <p>Note: This setting cannot be modified with the Configuration tool.</p>
OnlineTimeout	String	<p>Time (in milliseconds) to wait for device to come online.</p> <p>Note: This setting cannot be modified with the Configuration tool.</p>
OEMEmulation	String	<p>Emulate OEM printer.</p> <p>Valid values:</p> <p>True</p> <p>False (default)</p>
DefaultLargeFont	String	<p>This property selects the font with the largest size available on the printer, typically: 12 dots x 24 dots on Receipt and 7 half dots x 9 dots on SLIP. The selection is made when the device is OPENed.</p> <p>Valid values:</p> <p>True</p> <p>False (default)</p> <p>Note:</p> <ol style="list-style-type: none"> 1. If no property is found, the default value is used: False. 2. If the property is present but has an invalid value, then the default value is used.

Table 193. Service Object settings for SureMark devices (POS printer), supported (continued)

Keyword	Type	Description
DefaultTallFontA	String	<p>This property replaces the default font with the 10 dots x 24 dots font. The selection is made when the device is OPENed.</p> <p>Valid values:</p> <ul style="list-style-type: none"> True False (default) <p>Note:</p> <ol style="list-style-type: none"> 1. If no property is found, the default value is used: False. 2. If the property is present but has an invalid value, then the default value is used. 3. This property is only valid for 4610 2xR/1NR (EIA-232, RS-485, USB, Network) (SBCS).
Default8LPI	String	<p>This property selects the spacing needed to allow the user to print 8 lines per inch on the paper.</p> <p>Valid values:</p> <ul style="list-style-type: none"> True False (default) <p>Note:</p> <ol style="list-style-type: none"> 1. If no property is found, the default value is used. 2. If the property is present but has an invalid value, then the default value is used.
DocAdvance	String	<p>This property selects the number of motor steps to advance the paper on a Document Insert, from the top Document sensor to the first print position. Valid values: 1 to 255 in decimal representation.</p> <p>Note:</p> <ol style="list-style-type: none"> 1. This setting cannot be modified with the Configuration tool. 2. If no property is found, the default value is used: 50 (0x32). 3. If the property is present but the value is not a number, the default value is used. 4. If the property is present but has a value not in the range of values, it is matched to the closest parameter. 5. Applies only to EIA-232 4610 printer.
SetCompatibilityMode	String	<p>This property allows the 2xR, 1NR printer to remain in compatibility mode even when driver is updated. Valid values are True, False (default)</p> <p>Note:</p> <ol style="list-style-type: none"> 1. If no property is found, the default value is used 2. If the property is present but has an invalid value, then the default value is used.

Table 193. Service Object settings for SureMark devices (POS printer), supported (continued)

Keyword	Type	Description
PaperSaveMode	DWORD	<p>This property allows setting the paper save mode for 2xR, 1NR printer with EC level 0F and above. Please do not change this value manually. Paper Save Mode should only be configured from OPOS Configuration tool. PaperSaveMode is only supported in thermal station and is not supported in "Page mode."</p> <p>PaperSaveMode consists of 3 different options:</p> <ul style="list-style-type: none"> • Dots Between Lines, found on page 214 • Blank Line Reduction, found on page 214 • Barcode Height Reduction, found on page 214
DefaultCodePage	DWORD	The value stored in this property (if it exists) will be read during 2xR, 1NR printer device open and is used to initialize the CharacterSet property.
DefaultRecLineChars	DWORD	The value stored in this property (if it exists) will be read during TxN, 2xR, 1NR printer device open and is used to initialize the RecLineChars property.
Barcode128Mixed	String	<p>When the value is True, it converts the printBarcode symbology from PTR_BCS_Code_128 to PTR_BCS_Code_128_Parsed when the barcode data contains non numeric data and or odd length numeric data.</p> <p>It will also automatically add start sentinel A before the non numeric and single digit numeric data and sentinel C before the even-length numeric data when PTR_BCS_Code_128_Parsed symbology is selected and sentinel character is not detected in barcode data.</p> <p>Valid values:</p> <ul style="list-style-type: none"> True False (default)
NoSetBitmap	String	<p>This property allows the printing of bitmaps that are stored in printer memory without issuing SetBitmap each time the printer is opened.</p> <p>Valid values:</p> <ul style="list-style-type: none"> True False (default)
RecLineCharsFont	String	<p>This property allows the user to select a specific resident font so that the driver can provide a specific RecLineCharsList with values that are only valid for the specified font.</p> <p>A Select font A, RecLineCharList="32,33,36,38,41,44,48,52,57" B Select font B, RecLineCharList="28,30,32,34,36,38,41,44,48" C Select font C, RecLineCharList="36,38,41,44,48,52,57,64,72"</p> <p>Note:</p> <ol style="list-style-type: none"> 1. If the registry is not set, the default RecLineCharList = "32,34,36,38,41,44,48,52,57,64,72" is used. 2. To select Tall font A, set the registry to "A" and set DefaultTallFontA to "TRUE".

OPOS DirectIO calls

Miscellaneous

Flip Check

This flips the document currently in the DI station. It is supported on all printers that contain a check flipper.

Table 194. Flip check

Parameter	Type	Description
Command	Int32	0
Data	Long*	Ignored
Obj	BSTR*	Ignored

Attention: This OPOS DirectIO call should not be used. Applications should use the changePrintSide method of the printer object instead.

Print Downloaded Bitmap

This prints a bitmap previously stored in the printer. Not supported on EIA-232 printers.

Table 195. Print downloaded bitmap

Parameter	Type	Description
Command	Int32	0
Data	Long*	Pointer to a byte containing the bitmap number
Obj	BSTR*	Ignored

Attention: This OPOS DirectIO call should not be used. Applications should use the SetBitmap method and the ESC | #B escape sequence of the printer object instead.

Print Downloaded Message

This prints a message previously stored in the printer. It is not supported on EIA-232 printers.

Table 196. Print downloaded message

Parameter	Type	Description
Command	Int32	0
Data	Long*	Pointer to a byte containing the message number
Obj	BSTR*	Ignored

Read MICR Data

This reads the MICR data from the document currently in the DI station. It is only supported on printers connected via USB and RS-485.

Table 197. Read MICR data

Parameter	Type	Description
Command	Int32	4
Data	Long*	Ignored
Obj	BSTR*	Ignored

Attention: This OPOS DirectIO call should not be used. Applications should use the MICR service object instead.

Reset Printer Object

This forces the service object to reset itself to a known state. It is only supported on printers connected via USB and RS-485.

Table 198. Reset printer object

Parameter	Type	Description
Command	Int32	501
Data	Long*	Ignored
Obj	BSTR*	Ignored

Check for Errors

This forces the service object to determine if the printer is in an error condition. It will cause the ResultCode and ResultCodeExtended properties to be reset and is only supported on printers connected via USB and RS-485.

Table 199. Check for errors

Parameter	Type	Description
Command	Int32	502
Data	Long*	Ignored
Obj	BSTR*	Ignored

Set Double Strike Mode

This sets double-strike mode. It is only supported on Model 4 printers.

Table 200. Set double strike mode

Parameter	Type	Description
Command	Int32	20
Data	Long*	Pointer to a byte which determines the double-strike status Valid values are: 0 (disable) 1 (enable)
Obj	BSTR*	Ignored

Flash memory

All OPOS access to the Flash memory in the SureMark printer is through DirectIO() methods. The following DirectIO commands are added to the SureMark OPOS Service Object for the RS-485, USB and EIA-232 interfaces.

Table 201. Flash memory

Parameter	Type	Description
Command	Int32	FLASH_MEMORY
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	String

Remarks:

Write a record to flash memory. The format of the data to be written is r1.r2.r3.r4.n1.n2.data, where r1,r2, r3 and r4 are 32-bit numbers, in little-endian format, indicating the record number and n1 and n2 are 16-bit numbers, in little-endian format, indicating the number of data bytes to be written.

When AsyncMode is set to true, the data is queued to the printer. Any error associated with this write causes a DirectIoEvent. If the write is successful, an OutputCompleteEvent is fired.

If the record number is beyond the maximum records supported, pData is set to DIRECTIO_FLASH_ERROR_REASON_OUT_OF_RANGE (0x2F).

If the record number is beyond the record length or is longer than the set record length or the maximum supported, pData is set to DIRECTIO_FLASH_ERROR_REASON_TOO_LONG (0x2D).

When AsyncMode is false, this command waits until the memory is written to the printer (or an error condition occurs) before returning control to the application. If there is an error, ResultCodeExtended is set with either of the two values, or with the POSSWIN internal error.

Read flash memory

Table 202. Read flash memory

Parameter	Type	Description
Command	Long	FLASH_MEMORY (0x11)
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	String

Remarks:

Read the flash memory record number. The format of the data to be written is r1.r2.r3.r4.n1.n2.data, where r1, r2, r3 and r4 are 32-bit numbers, in little-endian format, indicating the record number.

When AsyncMode is set to true, the data is returned to the application by a DirectIoEvent with the EventNumber set to DIRECTIO_FLASH_DATA (0xFD). Any error associated with this read causes a DirectIoEvent with the EventNumber set to DIRECTIO_FLASH_ERROR (0xFE).

If the record number is beyond the maximum records supported, pData is set to DIRECTIO_FLASH_ERROR_REASON_OUT_OF_RANGE (0x2F).

If there is no response to the request within 5 seconds, pData is set to DIRECTIO_FLASH_ERROR_REASON_CMD_TIMEOUT (0x2E).

If the record number is beyond the record length or is longer than the set record length or the maximum supported, pData is set to DIRECTIO_FLASH_ERROR_REASON_TOO_LONG (0x2D).

When AsyncMode is false, this command waits until the memory is read from the printer (or an error condition) before returning the data to the application. The record is returned in the pString field of the DirectIO call. If there is an error, ResultCodeExtended is set with any one of these three values, or with the POSSWIN internal error.

Query flash size

Table 203. Query flash size

Parameter	Type	Description
Command	Long	QUERY_FLASH_SIZE (0x12)

Table 203. Query flash size (continued)

Parameter	Type	Description
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	n/a

Returns the size of memory.

The value is returned in the pData field of the DirectIO call.

Query max records

Table 204. Query max records

Parameter	Type	Description
Command	Long	QUERY_MAXIMUM_RECORDS (0x13)
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	n/a

Returns the maximum number of records. This number is calculated by dividing the maximum printer memory by the application requested memory size.

The value is returned in the pData field of the DirectIO call.

Set record length

Table 205. Set record length

Parameter	Type	Description
Command	Long	SET_RECORD_LENGTH (0x14)
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	n/a

Specifies the number of bytes for each record written to the flash memory. The flash memory should be erased after changing the record size. The OPOS drivers do not automatically erase the memory.

Erase flash memory

Table 206. Erase flash memory

Parameter	Type	Description
Command	Long	ERASE_FLASH_MEMORY (0x15)
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	n/a

Erases all data stored in the flash memory on the 4610 printer.

Get record length

Table 207. Get record length

Parameter	Type	Description
Command	Long	GET_RECORD_LENGTH (0x16)
Data	Long*	Refer to the 'Remarks' section below

Table 207. Get record length (continued)

Parameter	Type	Description
Obj	BSTR*	n/a

Retrieves the number of bytes for each record written to the flash memory.

The value is zero if it has not yet been set after an erase.

Check scanning support

DirectIO commands are available to support check scanning on the SureMark™ TI8/9 POS Printers. The supported commands for DirectIO check scanning are in the sections below.

Micro read with scan

Table 208. Micro read with scan

Parameter	Type	Description
Command	Long	CS_MICR_READ_WITH_SCAN (201)
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	n/a

Remarks:

This command is used to configure the printer to perform a MICR read in the same pass with a check scan. Set Data to CS_ENABLE_READ(1) to enable read with scan, CS_DISABLE_READ(0) to disable. To request the current value set Data to GET_VALUE(-1), which will return the current value in Data. The default is 0 (disabled). A return code of OPOS_E_ILLEGAL is returned if the value is not valid. A return code of OPOS_E_FAILURE is returned if the SureMark Printer not equipped with a document scanner.

Compression format

Table 209. Compression format

Parameter	Type	Description
Command	Long	CS_COMPRESSION_FORMAT (202)
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	n/a

Remarks:

This command is used to set the compression format when an image is stored or retrieved. Format of an image once stored does not change. To request the current value, set Data to GET_VALUE(-1); the current value will be returned in Data. The default is CS_CF_TIFF_COMP(0). A return code of OPOS_E_ILLEGAL is returned if the value is not valid. A return code of OPOS_E_FAILURE is returned if the SureMark Printer is not equipped with a document scanner. The possible values for Data are:

CS_CF_TIFF_COMP(0)

TIFF-CCIT-Group 4 compression

CS_CF_JPEG(1)

JPEG compression

CS_CF_BMP(2)

BMP (uncompressed)

CS_CF_NONE (3)

No compression (gray scale)

CS_CF_TIFF(4)

TIFF file, no compression (gray scale)

Scan document

Table 210. Scan document

Parameter	Type	Description
Command	Long	CS_SCAN_DOCUMENT (203)
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	String

Remarks:

This command is used to start the scan of the document present in the document insert station. A return code of OPOS_E_FAILURE is returned if the SureMark Printer is not equipped with document scanner or if scan is not successful. String parameter contains the width and length of the scanned document separated by a comma if the scan is successful.

Store document

Table 211. Store document

Parameter	Type	Description
Command	Long	CS_STORE_DOCUMENT (204)
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	String

Remarks:

This command is used to store all or part of the last document scanned. The document image will be stored in the format specified by compression format. The value in Data sets the area or areas to store. Upon return, Data will contain the first index of the storage area in the printer and String will contain the tag data string and the area dimensions to store. A return code of OPOS_E_FAILURE is returned if the SureMark Printer is not equipped with document scanner or if store is not successful. Possible Data values:

CS_STORE_ENTIRE(0)

store entire image

CS_STORE_PERSONAL(1)

store personal check template

CS_STORE_BUSINESS(2)

store business check template

CS_STORE_UD(3)

store using user defined values

For String, see “Storage string remarks” on page 235 for additional information.

Retrieve by index

Table 212. Retrieve by index

Parameter	Type	Description
Command	Long	CS_RETRIEVE_BY_INDEX (205)
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	String

Remarks:

This command retrieves the image stored at the location specified in Data and stores the image in String. If Data is zero, then the last scanned image is retrieved. The String data format is set based on BinaryConversion. All data starts with a header. A return code of OPOS_E_FAILURE is returned if the SureMark Printer is not equipped with a document scanner, or if the retrieve is not successful.

Retrieve by tagname

Table 213. Retrieve by tagname

Parameter	Type	Description
Command	Long	CS_RETRIEVE_BY_TAGNAME (206)
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	String

Remarks:

This command retrieves the image stored with the tagData value specified in String and stores it in String. The String data format is set based on OPOS-specific property. A return code of OPOS_E_ILLEGAL is returned if the string is empty. A return code of OPOS_E_FAILURE is returned if the SureMark Printer is not equipped with document scanner or if the retrieve is not successful. Image is marked as read after retrieval.

Erase storage

Table 214. Erase storage

Parameter	Type	Description
Command	Long	CS_ERASE_STORAGE (207)
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	n/a

Remarks:

This command erases all stored images from the printer storage. A return code of OPOS_E_FAILURE is returned if the SureMark Printer is not equipped with document scanner or if erase store is not successful.

Storage string remarks

The String parameter for the CS_STORE_DOCUMENT command contains the area or areas to be stored from the last scanned image, in the format:

$x0, y0, dx0, dy0, x1, y1, dx1, dy1, x2, y2, dx2, dy2, x3, y3, dx3, dy3, tagdata$

where:

$x0, y0$ 2 bytes each, top-left corner of area to be stored

dx0, dy0

2 bytes each, offset in the x and y direction respectively, to be stored

x1, x2, x3, y1, y2, y3,dx1,dx2,dx3,dy1,dy2,dy3

2 bytes each, corresponding offset in the x and y direction to define size of sub-block to store

tagdata

ASCII string null-terminated character, maximum 100 characters

String is interpreted based on the OPOS-specific property. All offsets are based upon the UnifiedPOS MapMode. If MapMode is set to PTR_MM_DOTS, 0.01 inches is used.

When storing an image, you can elect to store the entire image within the defined area, or just selected blocks within the defined area. Saving selected blocks reduces storage use, but keeps related data together. When storing multiple blocks of data, each block gets a unique storage location. A byte in the header message indicates that the block is part of a group. The first half-byte indicates which block of the group it is, the second half-byte indicates the number of blocks in the group. If the byte returned is X'14', then the block is the first of a group of four. All blocks contain header information, but only the first block in a group contains tag data. The storage method defines how the image should be stored.

CS_STORE_ENTIRE(0): All data defined within x0, y0, dx0, dy0 is stored; x1, x2, x3, y1, y2, y3, dx1, dx2, dx3, and dy1, dy2, dy3 should not be included.

CS_STORE_PERSONAL(1): A predefined template, based on a personal check is used to store the image. The area defined is the entire personal check, however only the name/address and the MICR blocks are saved; x0, x1, x2, x3, y0, y1, y2, y3, dx0, dx1, dx2, dx3, and dy0, dy1, dy2, dy3 should not be included.

CS_STORE_BUSINESS(2): A predefined template, based on a business check is used to store the image. The area defined is the entire business check, however only the name/address, and MICR blocks are saved; x0, x1, x2, x3, y0, y1, y2, y3, dx0, dx1, dx2, dx3, and dy0, dy1, dy2, dy3 should not be included.

CS_STORE_UD(3): This method allows the user to define sub-blocks of data to store. All command data (x0, x1, x2, x3, y0, y1, y2, y3, dx0, dx1, dx2, dx3, and dy0, dy1, dy2, dy3) must be sent. If fewer than four blocks of data are to be saved, the value of -1 must be sent for unused parameters.

The top-left corner of the scanned document is used as the point of reference (see Figure 37 on page 237). The xn command parameters are horizontal offsets from the left side of the scanned document. The yn parameters are vertical offsets from the top of the scanned document. The dxn and dyn parameters define the width and height, respectively, of the defined area or block.

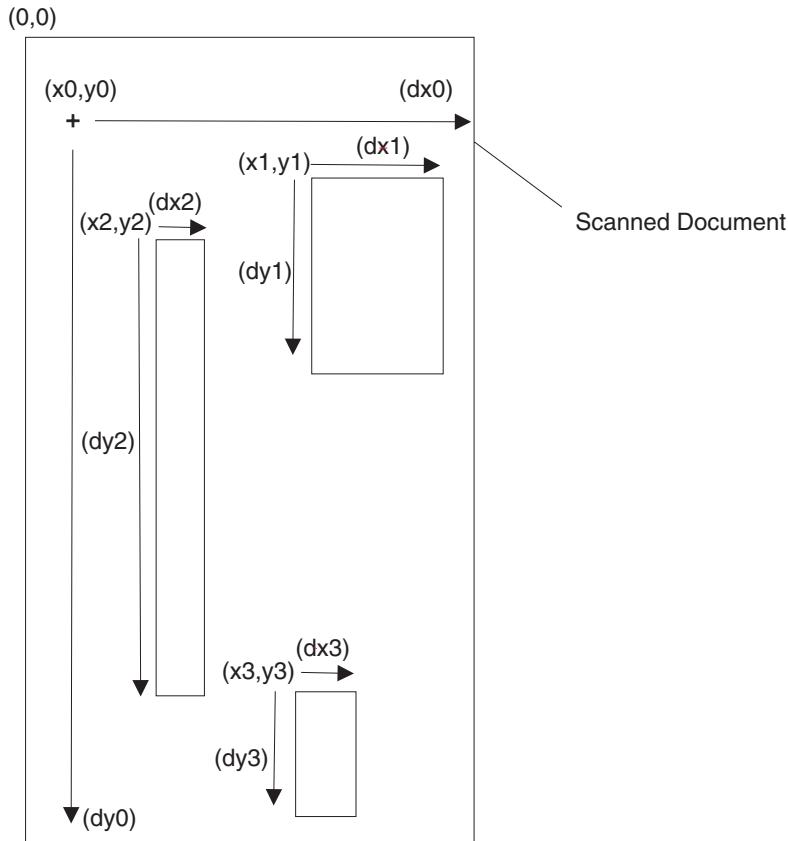


Figure 37. Scanned image organization

Retrieve storage header

Each image retrieved starts with header information in the following format:

- Status: 1 byte
 - Bit 7, Image read; 1 = image has been read, 0 = image has not been read
 - Bits 6 - 0; Reserved
- Compression: 1 byte
 - Bits 7 - 3; Reserved
 - Bits 2 - 0; Algorithm used to compress image. See Compression Format.
- Size1: 4 bytes
 - Number of bytes in image, not including tag data.
- Size2: 4 bytes (x_1, y_1)
 - Width of image (x_1) (in hundredths of inch increments), 2 bytes
 - Height of image (y_1) (in hundredths of inch increments), 2 bytes
- Block position: 1 byte
 - Lower four bits indicate the number of blocks that were saved for the image.
 - Upper four bits represent the current block number.
- Future: 1 byte
 - Reserved
- Tag data:
 - ASCII data supplied by the application, terminated by a null string.

Page Mode printing

DirectIO commands are available to support Page Mode printing on the SureMark TI3, TI4, TI8, and TI9 POS Printer. The following table shows the Page Mode supported commands for DirectIO.

You must use the Page Mode commands PM_HORIZONTAL_POSITION and PM_VERTICAL_POSITION to align Bar Codes. The printBarcode method alignment parameters are ignored for bar codes printed in Page Mode.

DirectIO commands are also used to set/get properties and capabilities. All commands are indirectly get commands, for example if the HORIZONTALPOSITION is set to a value that is not supported, the data parameter is set to a value that is supported and the hardware is set to that value.

Clear print area

Table 215. Clear print area

Parameter	Type	Description
Command	Long	PM_CLEAR_PRINT_AREA (101)
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	n/a

Remarks:

Clears the print area defined in the current PrintArea.

Page mode print

Table 216. Page mode print

Parameter	Type	Description
Command	Long	PM_PAGE_MODE_PRINT (102)
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	n/a

Remarks:

Data can be one of the following values:

PM_PM_PAGEMODE (1)

Enters Page Mode

PM_PM_PRINTSAVE (2)

Prints the current page and stays in page mode.

PM_PM_NORMAL (3)

Prints the current page and exits page mode.

PM_PM_CANCEL (4)

Clears the page area and exits page mode.

When the page is printed, the area printed is always from the top of the page area to the end of the print area specified. The top cannot be cropped by using the print area.

Horizontal and vertical position

Table 217. Horizontal and vertical position

Parameter	Type	Description
Command	Long	PM_HORIZONTAL_POSITION (103) PM_VERTICAL_POSITION (104)
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	n/a

Remarks:

Sets or returns the position for the horizontal/vertical print for the currently selected PageModeStation. If the value sent in position 0 is GETVALUE(-1) the current position is returned. Anything other than GETVALUE attempts to set the position; the value that is set is returned in position 0. Horizontal and vertical position is set based on the print direction. Horizontal position is parallel to the print direction. Vertical position is perpendicular to the print direction.

Page area

Table 218. Page area

Parameter	Type	Description
Command	Long	PM_PAGE_AREA (105)
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	String

Remarks:

Returns the Page area of the printer in the object for the currently selected PageModeStation. The format is "x,y", where x is the width (horizontal) and y is the length (vertical).

Page mode station

Table 219. Page mode station

Parameter	Type	Description
Command	Long	PM_PAGE_MODE_STATION (106)
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	n/a

Remarks:

Sets or returns the current PageMode station using position 0 of the data int array. If the value sent in position 0 is GETVALUE(-1) the current station is returned. Anything other than GETVALUE attempts to set the station; the value that is set is returned in position 0.

Print area

Table 220. Print area

Parameter	Type	Description
Command	Long	PM_PRINT_AREA (107)
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	String

Remarks:

Sets or returns the desired Print Area. If an empty StringBuffer is sent then a "get only" is emulated. Anything else attempts a set for the currently selected PageModeStation. The format is:

x,y,dx,dy

where:

- x is the horizontal start position.
- y is the vertical start position.
- dx is the horizontal offset.
- dy is the vertical offset.

Print direction

Table 221. Print direction

Parameter	Type	Description
Command	Long	PM_PRINT_DIRECTION (108)
Data	Long*	Refer to the 'Remarks' section below
Obj	BSTR*	n/a

Remarks:

OPOS DirectIO events

NVRAM data

This is received by the application in response to a READ_FLASH_MEMORY DirectIO call.

Table 222. NVRAM data

Parameter	Type	Description
Event Number	Int32	253
Data	Long*	0
Obj	BSTR*	BSTR containing the NVRAM data

NVRAM error

This is received by the application in response to a WRITE_FLASH_MEMORY or READ_FLASH_MEMORY DirectIO call.

Table 223. NVRAM error

Parameter	Type	Description
Event Number	Int32	254

Table 223. NVRAM error (continued)

Parameter	Type	Description
Data	Long*	<p>One of the following values:</p> <ul style="list-style-type: none"> 0x2A Data received is shorter than the record length. 0x2B NVRAM is full. 0x2C Unable to write record to NVRAM. 0x2D The record number is greater than the maximum number of records supported, or the record size is greater than the maximum record length. 0x2E The command took more than 5 seconds to complete. 0x2F The record number is greater than the maximum number of records supported.
Obj	BSTR*	Empty BSTR

OPOS extended result codes

Table 224. Extended result codes

ExtendedResultCode	Name	Description
501	IBM_EPTR_IMAGE_SCAN_FAILED	Image failed to be scanned by the printer.
502	IBM_EPTR_IMAGE_RETRIEVE_FAILED	An error occurred while reading check image data from printer
503	IBM_EPTR_HOME_ERROR	<p>The DI station print head is not in the home position.</p> <p>If opening/closing the DI station cover does not resolve the issue, call Toshiba and place a hardware service call.</p>
504	IBM_EPTR_PAPER_LOAD_ERROR	A required document was not in the slip station when data was to be printed.
505	IBM_EPTR_PAPER_ERROR	A paper feed error occurred .
506	IBM_EPTR_UNRECOVERABLE_ERROR	<p>Possible at startup if a firmware update corrupted the printer and the printer is running out of the boot sector.</p> <p>Should not happen on TI8/9 which does not update the firmware until after all the update data has been received.</p>
507	IBM_EPTR_COMMAND_REJECT	<p>The printer has rejected a command sent to it.</p> <p>The only time this is likely to occur is if the printer reset in the middle of sending logo data to the printer. When the printer is recovered and the first command is a continue data command containing the rest of the logo data, it will reject the data because it has not been setup to accept the starting logo data.</p>
508	IBM_EPTR_COMMUNICATION_ERROR	Low-level RS-485 communication protocol error.

Table 224. Extended result codes (continued)

ExtendedResultCode	Name	Description
509	IBM_EPTR_NOT_RESPONDING	An error occurred communicating with the printer.
510	IBM_EPTR_RESCAN_REQUIRED	An error occurred scanning the document, and it must be scanned again. Should only occur on TI9 printers, and only if the firmware level is less than ec69.
511	IBM_EPTR_PRINTER_RESET	The printer has reset itself. This is normally caused by the OS re-enumerating the USB bus and causing the printer to power cycle.
512	IBM_EPTR_NO_IMAGE_AVAILABLE	Check image failed to be read using RetriveByIndex or RetrieveByTagName DirectIO commands.
513	IBM_EPTR_CUTTER_JAM	Printer cutter jam occurred.

Additional OPOS information

4610 printer firmware update

The Toshiba OPOS driver has the ability to automatically update the printer firmware if the printer is attached to the terminal when the terminal boots. It does not have this ability if the printer is attached to the terminal after the terminal has booted and the operating system has loaded.

Printer firmware can be updated manually at any time.

Updated printer firmware update files are downloaded from the Toshiba website at www.toshibacommerce.com.

1. Go to www.toshibacommerce.com.
 - a. Under Peripherals, click **Toshiba SureMark Printer**.
 - b. Under Downloads, select the appropriate link for your printer model.
 - c. From the Search Results displayed, select **Toshiba 4610 SureMark Printer, Microcode Files for Firmware Update**.
 - d. Download the ZIP file for OPOS/JavaPOS. This file contains multiple firmware files; choose the correct one from the following for your printer type.
2. Place the firmware file for your printer type into the \pos\firmware folder.
3. Make sure the printer is attached and restart.

Each 4610 printer model has a different firmware file associated with it.

Table 225. Toshiba 4610 printer firmware update

Printer Model	Firmware update filename
TI1, TI2	aip46mc.hex
TI3, TI4, TG3, TG4, TF6, TM7	aip46mch.hex
TI5, TM7, TF7	aip46mcd.hex
TI8	aip46ti8.hex
TI9	aip46ti8.hex
2CR, 2NR	aip46v4.hex

If the firmware file you select does not match the type of printer you have attached, the program will not update the printer.

Manual update (EIA-232)

Printer firmware in an EIA-232 attached 4610 printer can be updated manually by using \\pos\\bin\\rs232firmwareupdate.exe. The status of the firmware update is logged to \\POS\\LOG\\rs232flash.log.

This is a GUI application that allows you to select a previously-configured printer to update, and which firmware file to use.

Manual update (RS485/USB)

Printer firmware in RS485 and USB attached 4610 printers can be updated manually by using \\pos\\bin\\aipfld46.exe.

This is a Windows command-line program that must be run from a Windows shell prompt. Use the following syntax to run the AIPFLD46 program:

```
aipfld46 -Sslot -Pport -Ffirmwarefilename [-O]
```

Table 226. Parameter setting for AIPFLD46

Parameter	Description
-S slot	The slot number of the 4610 for which the firmware download is intended. RS485 is slot number 1, USB is slot number 8.
-P port	The port number of the 4610 for which the firmware download is intended. The port number for RS485 and USB is 17.
-F firmwarefilename	The name of the file containing the firmware data for the printer. A fully qualified path name must be specified when this program is run manually.
-O	By default you can only upgrade firmware in a printer; you can force the application to downgrade printer firmware by using this option.

A log file named aipfld46.log is created in the \\pos\\log folder when the firmware download program is executed.

There is no progress indicator on screen, but you can determine the update progress by looking at the log file.

Automatic update (EIA-232)

Because EIA-232 printers can not be automatically detected by the driver, the firmware update program will try to apply all of the printer firmware files to all of the currently configured EIA-232 4610 printers.

The automatic firmware update process is started by a Windows service, "Toshiba OPOS RS232 Firmware Update".

If you do not want the firmware update process to run when the terminal boots, you may either disable the service or delete the firmware update files from the \\pos\\firmware folder

Automatic update (RS485/USB)

The automatic firmware update process for RS485 and USB printers is exactly the same as the manual process, except that the aipfld46 program is run automatically by the driver if it detects that the firmware in the update file is newer than the firmware in the printer.

If you do not want the firmware update process to run when the terminal boots, you should delete the firmware update files from the \pos\firmware folder.

4610 printer font download

To download DBCS fonts to the RS485/USB Toshiba SureMark Point of Sale Printer Model TI5, TM7, and TF7,1xR and 2xR you will need the following:

- The 4610 SureMark Point of Sale Printer font download program, AIPFNT46.EXE, which is installed in \pos\bin directory.
- DBCS Font files located in \pos\fonts directory

File Name	File Description
chnm1381.fon	Chinese Simplified (PRC) font - Code Page 1381
chnm950b.fon	Traditional Chinese (BIG-5) with Bold SBCS impact (Taiwan) fonts - Code
jpngs932.fon	Japanese Gothic Style font - Code Page 932
Jpnms932.fon	Japanese Mincho Style font - Code Page 932
Korms949.fon	Korean font - Code Page 949

Downloading DBCS fonts to the printer

Font files are downloaded manually using the AIPFNT46.EXE program:

```
AIPFNT46 -Sslot -Pport -Fx:\dir\font-file {-Q}
```

Argument	Description
-Sslot	The slot number (in decimal) of the printer for which the download is intended.
-Pport	The port number (in decimal) of the printer for which the download is intended
-Fx:\dir\font-file	Fully qualified path name for the font file to be downloaded to the printer.
-Q :	Quiet mode. No progress information is displayed

To download SBCS fonts to the RS485/USB Toshiba SureMark Point of Sale Printer Model TI3/4/6/8/9 1xR and 2xR you will need the following:

- The 4610 SureMark Point of Sale Printer font download program, AIPFNT46.EXE, which is installed in \pos\bin directory.
- For Windows The 4610 SureMark Point of Sale Printer font conversion program, AIPFNTCT.EXE, which is installed in \pos\bin directory.

Downloading DBCS fonts to the printer (EIA-232)

Font files are downloaded manually using the RS232FontUpdater program. The application can be run either as CLI or GUI (administrator privileges are required when running the application).

Run as CLI:

```
RS232FontUpdater {-p<port> -b<baud rate> -h<control flow>} {-n<printer name>} -f<font file path>
```

Note: Specify either the combination of port, baud rate and control flow of the printer, or the logical name configured to the printer.

Table 227. Details

Argument	Description
-p <port>	The port number (1-8) of the printer for which the download is intended.
-b <baud rate>	The baud rate of the printer for which the download is intended.
-h <control flow>	The control flow (1-3) of the printer for which the download is intended.
-n <printer name>	The logical name configured in OPOS config for the printer which the download is intended.
-f 	Fully qualified path name for the font file to be downloaded to the printer.

Run as **GUI**:

The printer font for an EIA-232 attached 4610 printer can be updated manually by using \pos\bin\RS232FontUpdater.exe. This is a GUI application that allows you to select a previously-configured printer to update, and which font file to use.

Converting printer font files

In Windows operating systems, manually convert the font files using the AIPFNTCT.EXE GUI program.

Downloading SBCS fonts to the printer

Font files are downloaded manually using the AIPFNT46.EXE program:

AIPFNT46 -Sslot -Pport -Fx:\dir\font-file {-Q} -Cx

Argument	Description
-Sslot	The slot number (in decimal) of the printer for which the download is intended.
-Pport	The port number (in decimal) of the printer for which the download is intended.
-Fx:\dir\font-file	Fully qualified path name for the font file to be downloaded to the printer.
-Q :	Quiet mode. No progress information is displayed.
-Cx	User-defined code page to be used for the proportional/user-defined fonts; valid values are: 1. One of the two proportional font locations; default, code page 1; CR station for user-defined. 2. CR station for user-defined. 3. One of the two proportional font locations; code page 3; CR station for user-defined. 4. CR station for user-defined. 5. DI station for user-defined. 6. DI station for user-defined.

Note:

1. Proportional fonts require two memory locations. For example, a proportional font in location one also fills location two.
2. Proportional font only applies to sectors 1 and 3. Fixed font can apply to all 6 sectors.
3. All fonts are erased when code page 1 is specified. Specify code page 1 first and then subsequent code pages.
4. The logfile (aipfnt46.log) contains the results of the last font download. This file is in the /pos/log directory.

5. You can get some SBCS Toshiba-defined font files from the following web site:
www.toshibacommerce.com.

Downloading SBCS fonts to the printer (EIA-232)

Font files are downloaded manually using the RS232FontUpdater program. The application can be run either as CLI or GUI (administrator privileges are required when running the application).

Run as **CLI**:

```
RS232FontUpdater {-p<port> -b<baud rate> -h<control flow>} {-n<printer name>} -f<font file path> -c<sector>
```

Note: Specify either the combination of port, baud rate and control flow of the printer, or the logical name configured to the printer.

Table 228. Details

Argument	Description
-p <port>	The port number (1-8) of the printer for which the download is intended.
-b <baud rate>	The baud rate of the printer for which the download is intended.
-h <control flow>	The control flow (1-3) of the printer for which the download is intended.
-n <printer name>	The logical name configured in OPOS config for the printer which the download is intended.
-f 	Fully qualified path name for the font file to be downloaded to the printer.
-c <sector>	User-defined code page to be used for the proportional/user-defined fonts. Valid values: <ol style="list-style-type: none"> 1. One of the two proportional font locations; default, code page 1; CR station for user-defined 2. CR station for user-defined 3. One of the two proportional font locations; code page 3; CR station for user-defined 4. CR station for user-defined 5. DI station for user-defined 6. DI station for user-defined

Note:

1. Proportional fonts require two memory locations. For example, a proportional font in location one also fills location two.
2. Proportional font only applies to sectors 1 and 3. Fixed font can apply to all 6 sectors.
3. All fonts are erased when code page 1 is specified. Specify code page 1 first and then subsequent code pages.
4. The logfile (aipfnt46.log) contains the results of the last font download. This file is in the /pos/log directory.
5. You can get some SBCS Toshiba-defined font files from the following web site:
www.toshibacommerce.com.

Run as **GUI**:

The printer font for an EIA-232 attached 4610 printer can be updated manually by using \pos\bin\RS232FontUpdater.exe. This is a GUI application that allows you to select a previously-configured printer to update, the sector to download the font, and which font file to use.

4689 printer font download

To download fonts to the Toshiba 4689 SurePOS Receipt Journal Printer, you will need the following:

- The 4689 Printer font download program, AIPFNTRJ.EXE, which comes with the Toshiba Point of Sale Subsystem for Windows.
 - The 4689 Printer font conversion program, 4689CNVT.EXE, and Toshiba 4689 Printer font files, which can be downloaded from the Toshiba Global Commerce Solutions support site:
www.toshibacommerce.com.
1. Select **Support** under Retail Store Solutions.
 2. Select **Other Systems and Adapters** under SurePOS Peripherals.
 3. Select **4689** under POS Printers.
 4. Select **Font Support Files** to download the font files.

The download contains the following:

File Name	File Description
4689JPNM.EXE	Japan, Mincho style font (diskette image)
4689JPNG.EXE	Korea, Mincho style font (diskette image)
4689JPNU.EXE	Japan, user-defined fonts (diskette image)
CONVERT.EXE	Contains: <ul style="list-style-type: none">• 4689CNVT.EXE : conversion program• jpngs932.cfg: Japan, gothic style + user-defined fonts font configuration file• jpnms932.cfg: Japan, mincho style + user-defined fonts font configuration file• readme.txt: the directions given here
readme.txt	The information presented in this section.

Converting Printer Font Files

Convert the downloaded files using the 4689CNVT.EXE program.

4689CNVT configuration-file

configuration-file

One of the configuration files contained in CONVERT.EXE

Note:

1. The font files must be in the same directory as the conversion program.
2. The output font file will have the same name as the supplied configuration file, but will have a file extension of .FON.

Downloading Fonts to the Printer

Font files for the Toshiba 4689 SurePOS Receipt Journal Printer must be downloaded manually using the AIPFNTRJ.EXE program:

AIPFNTRJ -Sslot -Pport -Fx:\dir\font-file {-Q}

-Sslot :

The slot number (in decimal) of the printer for which the download is intended.

-Pport :

The port number (in decimal) of the printer for which the download is intended.

-Fx:\dir\font-file :

Fully qualified path name for the font file to be downloaded to the printer.

-Q : Quiet mode. No progress information is displayed.

Model 4 printer font download

When the Toshiba JavaPOS system recognizes a Model 4A printer that it has not used before, it will automatically download one of the character sets described below to the printer:

File Description

M4A00850.FON

The single-byte characters are the same as those in the Model 4 printer. The characters in this font file are used for the following code pages:

- 437
- 850
- 852
- 857
- 860
- 861
- 862
- 863
- 864
- 865

M4A00932.FON

The double-byte characters for Japan. This character set is a subset of the characters in code page 932. For the supported characters, see the *Toshiba JavaPOS system: Installation, Keyboards, and Code Pages* publication.

M4A00949.FON

The double-byte characters for Korea. This character set is a subset of the characters in code page 949. For the supported characters, see the *Toshiba JavaPOS system: Installation, Keyboards, and Code Pages* publication.

Manually Downloading Characters

If you want to use another character set at a later time, or if you want to define your own characters and download them to the Model 4A printer, you can do so manually. The program to download a new character set to the Model 4A printer is AIPM4ALD.EXE. This program is in the \BIN directory where the Toshiba JavaPOS system is installed.

Use the following syntax to run the AIPM4ALD program:

```
AIPM4ALD -Sslot -Pport -Fx:\dir\font-file [ -Q ]
```

The parameters for the AIPM4ALD program are:

Parameter

Description

-S slot The slot number (in decimal) of the Toshiba Model 4A printer to which the new character set is to be downloaded.

-P port The port number (in decimal) of the Toshiba Model 4A printer to which the new character set is to be downloaded.

-F x:\dir\font-file

The name of the file containing the characters to be downloaded to the Toshiba Model 4A printer. A fully-qualified path name must be specified when this program is run manually.

-Q Quiet Mode. Do not display progress information

- V Verify Characters. Verify each character by printing it on the CR station after all the characters have been downloaded.

Font File Format

There are three types of records that are allowed in a Model 4A font file. These are:

- Comment Record
- Keyword Record
- Character Definition Record

A *comment record* can be either a blank line, or a line with an exclamation point (!) as the first non-blank character.

A *keyword record* is a line with one of the keywords defined below followed by one or more blank characters and ending with a value for the keyword.

A *character definition record* is a line of hexadecimal ASCII characters. Each pair of characters defines one byte of the character definition. The first two pairs of characters identify the character being defined and the remaining characters define the character.

Keywords:

To define characters to the Toshiba JavaPOS system for use with the Model 4A printer, the following keywords and their corresponding values must be defined in the font file along with the definition of each character. If any of these keywords are omitted, unpredictable results can occur when your application is using the printer.

Keyword

Description

CODEPAGE

The code page this font file defines.

HEIGHT

Defines the double-byte character height. The valid values are 9 or 16.

MODE

Defines this code page to be a single-byte, or a double-byte code page. A value of 0 (zero) indicates this is a single-byte code page and a value of 1 (one) indicates this is a double-byte code page.

SPACE

The number of dot columns to be added to each double-byte character. The valid values are 0 (zero) through 16.

WIDTH

Defines the double-byte character width. The valid values are 10 through 16.

Character Definition Record:

Each *character definition record* defines one character for the Model 4A printer. Each character on the line must be a hexadecimal ASCII character. Each pair of characters defines one byte of the character definition. The first two pairs of characters identify the character being defined and the remaining characters define the character.

Character Identifier:

The first two pairs of characters that identify the character being defined must be between 0x0000 and 0x00FF, or between 0x8000 and 0xFFFF. The characters in the range 0x0000 through 0x00FF define the single-byte characters and there must be 44 hexadecimal ASCII characters (22 bytes). The characters in the range 0x8000 through 0xFFFF define the double-byte characters and the number of hexadecimal ASCII characters depends upon the width of the character being defined. The number of characters in the line

can be calculated by multiplying the character width, as defined by the WIDTH keyword, by 4 and adding 4. The additional four characters are for the character identifier.

Note: If you define double-byte characters in your font file, you must define the character 0x8000 to be all zeros. This is because the Toshiba JavaPOS system uses that character for padding double-byte character print lines when positioning the print head over characters already printed.

Character Definition:

Each dot column on the Model 4A printer has nine print wires for single-byte characters and between nine to sixteen print wires for double-byte characters.

The definition of each character follows the character identifier on each line. Each pair of hexadecimal ASCII values defines one byte of the character. For the Model 4A printer, each pair of bytes (four characters) defines which print wires are turned on for each dot column to be printed. Within each pair, the first two hexadecimal ASCII characters (first byte) defines which of the lower eight print wires will be turned on and the second two hexadecimal ASCII characters (second byte) defines which of the upper eight print wires will be turned on. Within each byte, the least significant bit represents the top print wire.

For single-byte characters, the least significant bit of the first byte in the pair of dot column bytes represents print wire number nine. For single-byte characters, this is the only bit used in the first byte. For double-byte characters, the number of significant bits is determined by the LENGTH keyword. If the LENGTH keyword is set to 9, the double-byte character definition is the same as the single-byte character. If the LENGTH keyword is set to 16, then all eight bits of the first byte is used to define the print wires 9 through 16.

Character Definition Restrictions:

The following should be considered when defining characters for the Model 4A printer:

- When defining characters, the same print wire should not be turned on in consecutive dot columns.
- For single-byte characters, the spacing between characters is included in the character definition.
- For double-byte characters, the spacing between characters can be either included in the character definition, or defined by the SPACE keyword.

Barcode printing

The width parameter of the printBarcode method is not a definite value. Since the ratio of the width of thick and thin lines must be fixed, the width parameter is used to calculate a percentage of the total line width of the station, such as RecLineWidth. The printer hardware accepts values of 2 - 4. Therefore, if width is less than 34% of XxxLineWidth, then 2 is used, 34% to 66% sends a 3 and greater than 66% sends a 4.

The height parameter for the Receipt Station accepts 1 - 255 dot rows. On the slip station, it height parameter is converted to head passes. The acceptable number of head passes is 3 to 5, or 27 to 45 dots in map mode, when SlpLineSpacing equals 9 dots.

QR code printing

Creating registry entries

QR code printing requires these 3 REG_DWORD registry values to be present, under
HKEY_LOCAL_MACHINE\SOFTWARE\OLEforRetail\ServiceOPOS\PosPrinter\ThePrinterName:

Table 229. Creating registry entries

Registry value name	Description																																				
	Mode value	Mode name	Mode description																																		
QRCodeEncodingMode Accepted values: 0 to 5	0	Byte	Encodes lower case letters, text double-byte characters, binary data and 8 bit values.																																		
	1	Alpha-Numeric	Alphanumeric mode encodes data from this set of 45 characters: 10 numeric digits: Characters: 0 - 9 Byte values: 0x30 to 0x39 26 alphabetic characters: Characters: A - Z Byte values: 0x41 to 0x5A 9 symbols: Characters: SPACE \$ % * + - . / : Byte values: 0x20, 0x24, 0x25, 0x2A, 0x2B, 0x2D, 0x2E, 0x2F, 0x3A																																		
	2	Numeric	This mode encodes data from the decimal digit set: Characters: 0 - 9 Byte values: 0x30 to 0x39																																		
	3	Kanji	This mode encodes Kanji characters, subject to these constraints: 1. The number of data bytes to be encoded must be an even number. 2. The characters must be from the range: 0x8140 to 0xFFC, 0xE040 to 0xEBBF. If the constraints are not met, the command will be rejected.																																		
	4	Extended Channel Interpretation (ECI)	This mode allows the encoding of character sets different than the default.																																		
	5	Mixing	In this mode, the printer tries to decide whether to encode the data in Numeric, Alpha-Numeric, Byte or Kanji mode.																																		
The encoding mode, together with the error correction level determine the maximum number of characters that can be encoded. Please refer to the description of QRCodeErrorCorrectionLevel for details.																																					
QRCodeErrorCorrectionLevel Accepted values: 0 to 3	This value specifies the error correction level for the QR Barcode, with 0 being the lowest quality and 3 being the highest. Together with the encoding mode, the error correction level is also a factor in determining the maximum number of characters that can be encoded. In general, the lower the error correction level, the more characters can be encoded:																																				
	<table border="1"> <thead> <tr> <th>Level</th> <th>Recovery %</th> <th colspan="4">Maximum number of characters that can be encoded per mode</th></tr> </thead> <tbody> <tr> <td>0</td> <td>7</td> <td>Numeric</td> <td>Alpha-Numeric</td> <td>Byte</td> <td>Kanji</td> </tr> <tr> <td>1</td> <td>15</td> <td>1000</td> <td>1000</td> <td>919</td> <td>500</td> </tr> <tr> <td>2</td> <td>25</td> <td>1000</td> <td>1000</td> <td>701</td> <td>432</td> </tr> <tr> <td rowspan="2">3</td> <td rowspan="2">30</td> <td>1000</td> <td>721</td> <td>499</td> <td>308</td> </tr> <tr> <td>945</td> <td>566</td> <td>393</td> <td>242</td> </tr> </tbody> </table>				Level	Recovery %	Maximum number of characters that can be encoded per mode				0	7	Numeric	Alpha-Numeric	Byte	Kanji	1	15	1000	1000	919	500	2	25	1000	1000	701	432	3	30	1000	721	499	308	945	566	393
Level	Recovery %	Maximum number of characters that can be encoded per mode																																			
0	7	Numeric	Alpha-Numeric	Byte	Kanji																																
1	15	1000	1000	919	500																																
2	25	1000	1000	701	432																																
3	30	1000	721	499	308																																
		945	566	393	242																																

Table 229. Creating registry entries (continued)

Registry value name	Description																																																								
QRCodeECIValue Accepted values: 0 to 26, excluding 14, 19, 21 - 25	<p>This value is used only in Extended Channel Interpretation (ECI) encoding mode (Mode value = 4).</p> <table> <thead> <tr> <th>Value</th> <th>Encoding</th> </tr> </thead> <tbody> <tr><td>0</td><td>CP437</td></tr> <tr><td>1</td><td>ISO8859_1</td></tr> <tr><td>2</td><td>CP437</td></tr> <tr><td>3</td><td>ISO8859_1</td></tr> <tr><td>4</td><td>ISO8859_2</td></tr> <tr><td>5</td><td>ISO8859_3</td></tr> <tr><td>6</td><td>ISO8859_4</td></tr> <tr><td>7</td><td>ISO8859_5</td></tr> <tr><td>8</td><td>ISO8859_6</td></tr> <tr><td>9</td><td>ISO8859_7</td></tr> <tr><td>10</td><td>ISO8859_8</td></tr> <tr><td>11</td><td>ISO8859_9</td></tr> <tr><td>12</td><td>ISO8859_10</td></tr> <tr><td>13</td><td>ISO8859_11</td></tr> <tr><td>14</td><td>Reserved</td></tr> <tr><td>15</td><td>ISO8859_13</td></tr> <tr><td>16</td><td>ISO8859_14</td></tr> <tr><td>17</td><td>ISO8859_15</td></tr> <tr><td>18</td><td>ISO8859_16</td></tr> <tr><td>19</td><td>Reserved</td></tr> <tr><td>20</td><td>SJIS</td></tr> <tr><td>21</td><td>Reserved</td></tr> <tr><td>22</td><td>Reserved</td></tr> <tr><td>23</td><td>Reserved</td></tr> <tr><td>24</td><td>Reserved</td></tr> <tr><td>25</td><td>Reserved</td></tr> <tr><td>26</td><td>UTF-8</td></tr> </tbody> </table>	Value	Encoding	0	CP437	1	ISO8859_1	2	CP437	3	ISO8859_1	4	ISO8859_2	5	ISO8859_3	6	ISO8859_4	7	ISO8859_5	8	ISO8859_6	9	ISO8859_7	10	ISO8859_8	11	ISO8859_9	12	ISO8859_10	13	ISO8859_11	14	Reserved	15	ISO8859_13	16	ISO8859_14	17	ISO8859_15	18	ISO8859_16	19	Reserved	20	SJIS	21	Reserved	22	Reserved	23	Reserved	24	Reserved	25	Reserved	26	UTF-8
Value	Encoding																																																								
0	CP437																																																								
1	ISO8859_1																																																								
2	CP437																																																								
3	ISO8859_1																																																								
4	ISO8859_2																																																								
5	ISO8859_3																																																								
6	ISO8859_4																																																								
7	ISO8859_5																																																								
8	ISO8859_6																																																								
9	ISO8859_7																																																								
10	ISO8859_8																																																								
11	ISO8859_9																																																								
12	ISO8859_10																																																								
13	ISO8859_11																																																								
14	Reserved																																																								
15	ISO8859_13																																																								
16	ISO8859_14																																																								
17	ISO8859_15																																																								
18	ISO8859_16																																																								
19	Reserved																																																								
20	SJIS																																																								
21	Reserved																																																								
22	Reserved																																																								
23	Reserved																																																								
24	Reserved																																																								
25	Reserved																																																								
26	UTF-8																																																								

Issuing the printBarcode command

After creating the registry entries, QR codes can be printed to the receipt station on supported printers using the UnifiedPOS method for printing barcodes:

printBarcode (station: int32, data: string, symbology: int32, height: int32, width: int32, alignment: int32, textPosition: int32)

Table 230. Accepted argument values

Parameter	Accepted values
station	Receipt. QR code printing is supported on the Receipt station only .
data	The input string must already be in the character encoding that matches the encoding mode.
symbology	PTR_BCS_QRCODE (Value = 204).
height	<p>Height is used to scale the QR code.</p> <p>Valid values: 0<= height <= 5</p> <p>Other values: Auto-scaling where 0 is the minimum scale size, which translates to 3 dots per pixel.</p> <p>Supported from OPOS1.14.0 and printer FW 15.81 onward.</p>
width	Unused. Printer automatically derives QR code dimensions.
alignment	As per UnifiedPOS specification.
textPosition	Unused.

Rotated printing

On the slip, only the alignment settings are ignored. Bold and Double High attributes are ignored by the hardware.

DBCS support

Downloading fonts to SureMark printers that support DBCS is described in *Point of Sale Subsystem: Programming Reference and User's Guide, Appendix F*.

Color printing

In order to use color printing capability:

- The firmware level must be 33 or greater.
- Supported color thermal paper must be used.
- The CapRec2Color option must be set at configuration time.

User-defined fonts

The SureMark printers support up to four user-defined character sets. For EIA-232 printers, the user-defined character sets are identified as character sets 102 through 105; for RS-485 and USB printers, the user-defined character sets are identified as 101 through 104. These character set identifiers appear in the CharacterSetList property.

Note: For EIA-232 printers, character set 101 is the printer's generic code page. For RS-485 and USB, the printer's generic code page is not supported.

To define a single proportional font, two of the printer's user-defined character sets are required. For EIA-232, a user-defined proportional font can be either character set 102 or 104; for RS-485 and USB, a user-defined proportional font can be either character set 101 or 103.

Only two user-defined fonts are supported on the impact station. When using a printer that has an impact station:

1. The odd-numbered user-defined character sets on the receipt station map to user-defined character set 1 on the slip station; even numbered character sets on the receipt station map to slip user-defined character set 2.
2. If the application uses user-defined character sets with a printer that has both receipt and slip stations, then both receipt and slip character sets should be downloaded to the printer.

Proportional font support

The printer drivers default to fixed width. In order to switch between fixed and proportional fonts, the Font Typeface is used. To illustrate this difference, the FontTypefaceList displays values "Fixed, Proportional" only when the CharacterSet has been set to a Proportional User Defined Font (101 or 103 for RS-485/USB, or 102 and 104 for EIA-232). This is only valid for User Defined Proportional Fonts. The resident fonts are fixed width.

The OPOS Specification assumes that all characters are fixed width. Proportional font printing is supported, providing the following:

1. If a proportional font is active and the current font typeface is set to proportional, all properties such as RecLineChars, RecLineHeight and RecLineSpacing are set to zero and RecCharList is set to a null string. The downloaded font determines the properties of the printed line.
Based on this assumption, we do not wrap lines in proportional mode when the number of characters on a line is greater than RecLineChars. Instead, it is up to the application developer to send a line feed when the print line is complete or the printer feeds when it has reached the end of the line.
2. Text alignment is based on the fact that a fixed number of characters fit on a line. This is not possible with proportional fonts. Therefore, when the printer is in proportional mode, instead of formatting the line within the SO, we pass on the Alignment Escape Sequences within the printer to let the printer format the alignment. This allows the printer to center text or split left and right aligned text at the hardware.

Finally, if the font typeface is set to Fixed when a proportional font is used, the printer prints the characters a fixed distance apart, and all of the Line properties are valid. The actual fixed width is set in the registry entry for the printer, using the keyword "ProportionalFontFixedWidth". The range of values the printer supports is 8 to 32. If this value is not specified, the control uses a width of half of the height and then adjust it for a best fit within the valid range.

Device sharing

Certain models of SureMark printers are equipped with tone devices and may be attached with cash drawer. While the OPOS model states that tone indicators and cash drawers are shareable devices, this device cannot be shared from separate applications. Trying to enable this device from two applications will cause the enable to fail for the second application. If using the tone indicator or cash drawer from two applications, it is recommended that each application claim the device before using it.

Code 128 A/B/C support

The Code 128 Bar Code Symbology has three code sets and also includes some special characters that indicate a change in code set, a function, or a shift from Code Set A to B, or vice versa. Table 231 lists the characters for each code set.

Table 231. Code 128 character sets

Code set	Character set
Code A	X'00'-X'5F', FNC1, FNC2, FNC3, FNC4, SHIFT, CODE B, CODE C
Code B	X'20'-X'7f', FNC1, FNC2, FNC3, FNC4, SHIFT, CODEA, CODE C
Code C	X'00'-X'63' for decimal values 00-99, FNC1, CODE A, CODE B

Characters are mapped from ASCII to the corresponding value for the selected code set. In Code Sets A and B, this is a one-to-one mapping. In Code Set C, each two digits is converted to a single value. A sentinel character, the left curly bracket ({)followed by a particular character, is used to indicate a special character. A starting code set is required at the start of the data. The symbology value to use is 123.

Table 232 lists the character pairs for encoding the special characters:

Table 232. Code 128 special character encoding

Special character	Character pair for encoding
SHIFT	{S
CODE A	{A
CODE B	{B
CODE C	{C
FNC1	{1
FNC2	{2
FNC3	{3
FNC4	{4
{	{}{

SureOne/SurePOS 100 DBCS printer limitations

For a DBCS printer, the printer code page is assumed to be the same as the locale of Windows.

Improving printer performance

When `AsyncMode` is *false*, Toshiba's OPOS Printer drivers verify that each print line is printed on the paper in synchronous mode, not just that it has been sent to the printer. This approach provides the application with an accurate completion status, but as a result, the drivers appear to be slower than the rated printer throughput. To improve printer driver throughput, do the following:

- Set `AsyncMode` to *true* so that groups of printer lines are sent out to the printer as they are received by the driver.
- Use Transaction Mode.
- Group lines in single print command.
- Format a full line.

Character sets supported by POSPrinter devices for OPOS

Table 233. Single-byte character sets supported by POSPrinter devices for OPOS

Code Page - SBCS	Device															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
437	✓	✓	✓	✓	✓			✓	✓		✓		✓	✓	✓	✓
775													✓	✓	✓	✓
808																
819	✓	✓	✓										✓	✓	✓	✓
848													✓	✓	✓	✓
850	✓	✓	✓	✓	✓						✓		✓	✓	✓	✓
													Note 1 on page 217			
852	✓	✓	✓								✓		✓	✓	✓	✓
													Note 1 on page 217			
855	✓	✓	✓								✓		✓	✓	✓	✓
													Note 1 on page 217			
857	✓	✓	✓								✓		✓	✓	✓	✓
													Note 1 on page 217			
858	✓	✓	✓					✓	✓				✓	✓	✓	✓
													Note 1 on page 217			

Table 233. Single-byte character sets supported by POSPrinter devices for OPOS (continued)

Code Page - SBCS	Device														
	1	2	3	4	5	6	7	8	9	10	11	12			
860	✓	✓	✓								✓ Note 1 on page 217	✓	✓	✓	✓
861	✓	✓	✓								✓ Note 1 on page 217	✓	✓	✓	
862	✓	✓	✓							✓	✓ Note 1 on page 217	✓	✓	✓	
863	✓	✓	✓								✓ Note 1 on page 217	✓	✓	✓	
864	✓	✓	✓							✓	✓ Note 1 on page 217	✓	✓	✓	
865	✓	✓	✓								✓ Note 1 on page 217	✓	✓	✓	
866	✓	✓	✓							✓	✓ Note 1 on page 217	✓	✓	✓	
867												✓	✓	✓	
869	✓	✓	✓								✓ Note 1 on page 217	✓	✓	✓	
872												✓	✓	✓	
874										✓	✓ Note 1 on page 217				

Table 233. Single-byte character sets supported by POSPrinter devices for OPOS (continued)

Code Page - SBCS	Device														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
897											✓			✓	
998	✓	✓	✓	✓	✓			✓	✓		✓		✓	✓	✓
999	✓	✓	✓										✓	✓	✓
1116	✓	✓	✓										✓	✓	✓
1117	✓	✓	✓										✓	✓	✓
1118	✓	✓	✓										✓	✓	✓
1250	✓	✓	✓										✓	✓	✓
1251	✓	✓	✓										✓	✓	✓
1252	✓	✓	✓										✓	✓	✓
1253	✓	✓	✓										✓	✓	✓
1254	✓	✓	✓										✓	✓	✓
1255													✓	✓	✓
1256													✓	✓	✓
1257	✓	✓	✓										✓	✓	✓

Note:

1. Standard model only

Table 234. Double-byte character sets supported by POSPrinter devices for OPOS

Code Page - DBCS	Device														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
926															
932	✓ Note 5 on page 258	✓ Note 5 on page 258	✓ Note 5 on page 258			✓	✓	✓	✓		✓ Note 1 on page 258	✓ Note 6 on page 258	✓ Note 1 on page 258	✓ Note 5 on page 258	✓ Note 5 on page 258
936															
949	✓ Note 5 on page 258	✓ Note 5 on page 258	✓ Note 5 on page 258			✓	✓	✓			✓ Note 6 on page 258		✓ Note 5 on page 258	✓ Note 5 on page 258	✓ Note 5 on page 258

Table 234. Double-byte character sets supported by POSPrinter devices for OPOS (continued)

Code Page - DBCS	Device													
	✓ Note 5	✓ Note 5	✓ Note 5						✓ Note 2	✓ Note 6	✓ Note 2	✓ Note 5	✓ Note 5	✓ Note 5
950	✓ Note 5	✓ Note 5	✓ Note 5						✓ Note 2	✓ Note 6	✓ Note 2	✓ Note 5	✓ Note 5	✓ Note 5
951											✓ Note 5	✓ Note 5	✓ Note 5	
1361									✓ Note 3		✓ Note 3			
1381	✓ Note 5	✓ Note 5	✓ Note 5						✓ Note 4	✓ Note 6	✓ Note 4	✓ Note 5	✓ Note 5	✓ Note 5

Only on:

1. Japanese version.
2. Traditional Chinese version.
3. Korean version.
4. Simplified Chinese version.
5. Available after Font Download.
6. Defaults to a list of SBCS codepages. There is no way to query the hardware for a specific model, so the default language settings of the operating system are used to determine the DBCS codepage.

Chapter 17. Scanner (bar code reader)

Supported devices

Table 235. Scanner supported devices

Device	Connectivity
1. Toshiba 1520 Compatible Hand-held Scanner	RS-485
2. Toshiba 4687 Compatible Scanner	RS-485 USB (using Protocol Converter)
3. Toshiba 4696 Compatible Scanner	RS-485 USB (using Protocol Converter)
4. Toshiba 4697 Compatible Scanner	RS-485 USB (using Protocol Converter)
5. Toshiba 4698 Compatible Scanner	RS-485 USB (using Protocol Converter)
6. Toshiba HHBCR Compatible	RS-485
7. Toshiba HHBCR2 Compatible	RS-485
8. OEM Scanner	USB (must conform to <i>Toshiba USB OEM Point-of-Sale Device Interface Specification V1.0 or later</i>)
9. Toshiba 4685 Scanner	RS-485, USB
10. Kiosk scanner	Integrated

Supported properties and methods

Table 236. Scanner common properties

Property	JavaPOS	OPOS
AutoDisable		All
BinaryConversion	Not supported	All
CapCompareFirmwareVersion		False
CapPowerReporting	PR_STANDARD	All support PR_STANDARD
CapStatisticsReporting	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CapUpdateFirmware		False
CapUpdateStatistics	False (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CheckHealthText		All
Claimed		All
DataCount		All
DataEventEnabled		All
DeviceControlDescription		All

Table 236. Scanner common properties (continued)

Property	JavaPOS	OPOS
DeviceControlVersion		All
DeviceEnabled		All
DeviceServiceDescription		All
DeviceServiceVersion		All
FreezeEvents		All
OpenResult	Not supported	All
OutputID		Not supported
PowerNotify		All
PowerState		All
PhysicalDeviceDescription		All
PhysicalDeviceName		All
ResultCode	Not supported	All
ResultCodeExtended	Not supported	All
State		All

Table 237. Scanner specific properties

Property	JavaPOS	OPOS
DecodeData		All
ScanData		All
ScanDataLabel		All
ScanDataType		All

Table 238. Scanner common methods

Method	JavaPOS	OPOS
checkHealth		All
claim		All
clearInput		All
clearOutput		All
close		All
compareFirmwareVersion		Not supported
directIO		Not supported
open		All
release		All
resetStatistics		Not supported
retrieveStatistics	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	Not supported
updateFirmware		Not supported
updateStatistics		Not supported

Table 239. Scanner events

Event	JavaPOS	OPOS
DataEvent		All
DirectIOEvent		Not supported
ErrorEvent		All
StatusUpdateEvent		All

JavaPOS configuration

This section contains device-specific notes for the EIA-232, RS-485 and USB scanners. Scanner configuration is done through the jpos.xml file. An example of a typical configuration property is:

```
<prop name="setEnableCODE39" type="Boolean" value="true"/>
```

Note:

1. Each scanner's hardware capabilities determine which symbologies it can decode. Enabling a symbology property does not guarantee that the attached scanner can decode that symbology. See the Web site of the scanner vendor for more information about the supported symbologies and hardware capabilities.
2. When RS-485 scanners are attached to the system using a USB protocol converter, the supported symbologies are unchanged.
3. To read supplemental data on a 4685 scanner, at least one of the labels must be disabled. For example, setXXX label property to false in JposEntry for this scanner. The only way to enable all supported labels is to set the scanner to "test mode". In test mode, the 4685 reads all the supported labels, except for supplementals. To read supplementals, at least one label must be disabled.
4. If the device JposEntry in the jpos.xml does not contain a specific property, that property will be initialized to the default value specified in this document. Otherwise the value specified for that property in the JposEntry will be used.

Table 240 lists the supported symbologies for Toshiba scanners.

Table 240. Supported symbologies for Toshiba scanners

Symbology	1520	4500	4501	4685	4696	4697	4698	USB OEM	AnyPlace Kiosk Line	AnyPlace Kiosk Omni
UPC-A	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
UPC-E	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
UPC D1					✓	✓	✓	✓		
UPC D2					✓	✓	✓	✓		
UPC D3			✓	✓	✓	✓	✓	✓		
UPC D4					✓	✓	✓	✓		
UPC D5					✓	✓	✓	✓		
EAN/JAN 8		✓	✓	✓	✓	✓	✓	✓	✓	✓
EAN/JAN 13		✓	✓	✓	✓	✓	✓	✓	✓	✓
ITF	✓	✓	✓	✓		✓	✓	✓	✓	✓
CODABAR			✓	✓		✓		✓	✓	✓

Table 240. Supported symbologies for Toshiba scanners (continued)

Symbology	1520	4500	4501	4685	4696	4697	4698	USB OEM	AnyPlace Kiosk Line	AnyPlace Kiosk Omni
CODE 39	✓	✓	✓	✓		✓	✓	✓ Note 1	✓	✓
CODE 93			✓	✓		✓		✓	✓	✓
CODE 128			✓	✓		✓	✓	✓	✓	✓
Standard 2 of 5									✓	✓
UCC/EAN128								✓	✓	✓
GS1 DataBar Omnidirectional								✓	✓	✓
GS1 DataBar Expanded								✓	✓	✓
PDF-417								✓		✓
Maxicode								✓		
OCR-A OCR-B								✓		
Datamatrix								✓		
QR Code								✓		
Aztec Code								✓		
GS1 DataMatrix								✓		
GS1 QR Code								✓		
ISBT 128								✓		

Note:

1. Code 39 includes Code 39 with check character verified and transmitted (Code 39_CK) and Code 39 with Mod 32 check character (Code 32).

Table 241 lists the configurable parameters for scanners and indicates which parameters are supported by each scanner model.

Table 241. Scanner configurable parameters

Parameter	1520	4500	4501	4685	4696	4697	4698	USB OEM	AnyPlace Kiosk Line	AnyPlace Kiosk Omni
"enableLaserOnOffSwitch" on page 264								✓		
"enableProgrammingViaBarcodes" on page 264					✓	✓	✓	✓		✓
"enableSwitchControlledVolumeAdjust" on page 264					✓	✓	✓	✓		
"setBarCodes1" on page 264	✓	✓	✓	✓	✓	✓	✓	✓		
"setBarCodes2" on page 265	✓			✓		✓	✓	✓		
"setBarCodes3" on page 265	✓						✓	✓		
"setBarCodes4" on page 265	✓						✓	✓		
"setBeeperDuration" on page 266					✓	✓	✓	✓	✓	
"setBeeperFrequency" on page 266					✓	✓		✓	✓	✓
"setBeeperVolume" on page 266					✓	✓	✓	✓		
"setCheckModulo" on page 267	✓	✓	✓							
"setCode128ScansPerRead" on page 267							✓			
"setCode39ScansPerRead" on page 267							✓			
"setDecodeAlgorithm" on page 267					✓	✓	✓			
"setDoubleReadTimeOut" on page 268					✓	✓	✓	✓		

Table 241. Scanner configurable parameters (continued)

Parameter	1520	4500	4501	4685	4696	4697	4698	USB OEM	AnyPlace Kiosk Line	AnyPlace Kiosk Omni
"setDTouchMode" on page 268		✓	✓	✓						
"setEAN13ScansPerRead" on page 268					✓	✓	✓			
"setEAN8ScansPerRead" on page 269					✓	✓	✓			
"setEnabled_2_DigitSupplements" on page 269					✓			✓	✓	✓
"setEnabled_5_DigitSupplements" on page 269					✓			✓	✓	✓
"setEnabledCodabar" on page 269			✓	✓		✓		✓	✓	✓
"setEnabledCode128" on page 270			✓	✓		✓	✓	✓	✓	✓
"setEnabledCode128Supplements" on page 270								✓		
"setEnabledCODE39" on page 270	✓	✓	✓	✓		✓	✓	✓	✓	✓
"setEnabledCode39CheckDigit" on page 270								✓		
"setEnabledCode93" on page 270			✓	✓		✓		✓	✓	✓
"setEnabledEAN_JAN_TwoLabelDecoding" on page 270						✓	✓	✓		
"setEnabledGoodReadBeep" on page 271	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
"setEnabledInterleaved2of5" on page 271	✓	✓	✓	✓		✓	✓	✓	✓	✓
"setEnabledITFCheckDigit" on page 272								✓		
"setEnabledPDF417" on page 272										✓
"setEnabledRSS14 (deprecated)" on page 272									✓	✓
"setEnabledRSS_Expanded (deprecated)" on page 273									✓	✓
"setEnabledStandard2of5" on page 273									✓	✓
"setEnabledUCC_EAN128" on page 273								✓	✓	✓
"setEnabledUPC_A_CheckDigit" on page 273								✓	✓	✓
"setEnabledUPC_A_To_EAN13Expansion" on page 273						✓	✓	✓	✓	✓
"setEnabledUPC_E_CheckDigit" on page 274								✓	✓	✓
"setEnabledUPC_E_To_EAN13Expansion" on page 274						✓	✓	✓	✓	
"setEnabledUPC_E_To_UPC_AExpansion" on page 274								✓	✓	✓
"setEnabledUPCAE_EANJAN813" on page 274	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
"setEnabledUPCD1D5" on page 274			✓		✓	✓	✓	✓		
"setEnabledVerificationUPC_A_EAN13_fiveDigit" on page 275					✓	✓	✓	✓		
"setEnabledVerificationUPC_A_EAN13_fourDigit" on page 275					✓	✓	✓	✓		
"setITFLength1" on page 275	✓					✓	✓	✓	✓	✓
"setITFLength2" on page 275						✓	✓	✓	✓	✓
"setITFLengths" on page 276							✓	✓	✓	✓
"setITFLengthSpecifiedTwo" on page 276						✓	✓	✓		
"setITFScansPerRead" on page 276							✓			
"setLaserTimeOut" on page 276					✓	✓	✓	✓		
"setLED_GoodRead_Duration" on page 277								✓		
"setMotorTimeOut" on page 277					✓	✓	✓	✓		
"setSecurityLevelForInStore" on page 278								✓		
"setSTFLength1" on page 278									✓	✓
"setSTFLength2" on page 278									✓	✓
"setSTFLengths" on page 278									✓	✓
"setStoreScansPerRead" on page 279					✓	✓	✓			
"setSupplements" on page 279								✓		✓
"setSupplementsSecurityLevel" on page 279									✓	✓
"setUPCAScansPerRead" on page 280					✓	✓	✓			
"setUPCDScansPerRead" on page 280					✓	✓	✓			

Table 241. Scanner configurable parameters (continued)

Parameter	1520	4500	4501	4685	4696	4697	4698	USB OEM	AnyPlace Kiosk Line	AnyPlace Kiosk Omni
"setUPCEScansPerRead" on page 280					✓	✓	✓			

enableLaserOnOffSwitch

Property Type: boolean

Default: false

Models Supported: USB OEM

This property controls whether the laser power switch on the scanner unit is enabled or disabled. When a scanner is on, its motor is running and its laser is active. When a scanner is off, its motor is not running and its laser is inactive. When the laser power switch is enabled, it can be used to turn the scanner off. When the laser power switch is disabled, it cannot be used to turn the scanner off. The laser power switch can always be used to turn the scanner on.

enableProgrammingViaBarcodes

```
<prop name="enableProgrammingViaBarcodes" type="Boolean" value="false" />
```

This property controls whether or not the scanner can be programmed using the manufacturer-supplied programming bar codes.

Supported on: 4696, 4697, 4698, AnyPlace Kiosk Line scanner, AnyPlace Kiosk Omni scanner.

enableSwitchControlledVolumeAdjust

Property Type: boolean

Default: false

Models Supported: 4696, 4697, 4698 and USB OEM

This property enables and disables switch-controlled beep volume adjustment.

setBarCodes1

Property Type: byte

Default: 00

Models Supported: 1520, 4500, 4501, 4685, 4696, 4697, 4698, USB OEM

This property is one of several properties which determine what combination of bar code types a scanner recognizes. When configured in a specific mode, a scanner recognizes and returns all bar code types that are associated with that mode. Table 242 lists the values that can be given for the setBarCodes1 property for both RS-485 and USB-attached scanners.

Table 242. setBarCodes1 values

Resource Value	Bar Code Types
0 - GROUP_NONE	None
1 - GROUP_UPC_EAN_ITF	UPC-A, UPC-E, EAN/JAN-8, EAN/JAN-13, Interleaved 2 of 5 (ITF)

Table 242. *setBarCodes1* values (continued)

Resource Value	Bar Code Types
2 - GROUP_UPCAED	UPC-A, UPC-E, EAN/JAN-8, EAN/JAN-13UPC-D1, UPC-D2, UPC-D3, UPC-D4, UPC-D5
3 - GROUP_UPC_EAN_CODE128	UPC-A, UPC-E, EAN/JAN-8, EAN/JAN-13UPC-A, UPC-E, EAN/JAN-8, EAN/JAN-13, Code128
4 - GROUP_UPC_EAN_CODE93	UPC-A, UPC-E, EAN/JAN-8, EAN/JAN-13UPC-A, UPC-E, EAN/JAN-8, EAN/JAN-13, Code93
5 - GROUP_UPC_EAN_CODE39	UPC-A, UPC-E, EAN/JAN-8, EAN/JAN-13UPC-A, UPC-E, EAN/JAN-8, EAN/JAN-13, Code39
6 - GROUP_UPC_EAN_CODABAR	UPC-A, UPC-E, EAN/JAN-8, EAN/JAN-13UPC-A, UPC-E, EAN/JAN-8, EAN/JAN-13, Codabar
7 - GROUP_UPC_EAN_2_5_CODABAR	UPC-A, UPC-E, EAN/JAN-8, EAN/JAN-13UPC-A, UPC-A+2/+5, UPC-E, UPC-E+2/+5, EAN/JAN-8, EAN/JAN-8+2/+5, EAN/JAN-13, EAN/JAN-13+2/+5, Codabar (See Note.)

Note: Different scanners handle supplementals (+2, +5) differently. When supplementals are enabled, some scanners will only read UPC-A, UPC-E, EAN/JAN-8 and EAN/JAN-13 barcodes with a supplemental; other scanners will read UPC-A, UPC-E, EAN/JAN-8 and EAN/JAN-13 barcodes with and without a supplemental.

setBarCodes2

Property Type: byte

Default: 0

Scanner models supported: 1520, 4685, 4697, 4698, USB OEM

This property is one of several properties which determine what combination of bar code types a scanner recognizes. When configured in a specific mode, a scanner recognizes and returns all bar code types that are associated with that mode.

The values for the setBarCodes2 property are the same as those for setBarCodes1 property.

setBarCodes3

Property Type: byte

Default: 0

Models Supported: 1520, 4698, USB OEM

This property is one of several properties which determine what combination of bar code types a scanner recognizes. When configured in a specific mode, a scanner recognizes and returns all bar code types that are associated with that mode.

The values for the setBarCodes3 property are the same as those for setBarCodes1 property.

setBarCodes4

Property Type: byte

Default: 0

Models Supported: 1520, 4698, USB OEM

This property is one of several properties which determine what combination of bar code types a scanner recognizes. When configured in a specific mode, a scanner recognizes and returns all bar code types that are associated with that mode.

The values for the setBarCodes4 property are the same as those for setBarCodes1 property.

setBeeperDuration

Property Type: byte

Default: 0

Models Supported: USB OEM , 4696, 4697, 4698, AnyPlace Kiosk Line Scanner

This property specifies the duration of the tone that the beeper makes upon a successful read. This property can have the following values:

- 0 SHORT: Use the shortest time value for the beeper duration.
- 1 MEDLONG: Use a medium time value for the beeper duration.
- 2 LONG: Use a longtime value for the beeper duration.
- 3 LONGEST: Use the longest time value for the beeper duration.

setBeeperFrequency

Property Type: byte

Default: 0

Models Supported: 4696, 4697, USB OEM, AnyPlace Kiosk Line Scanner, AnyPlace Kiosk Omni Scanner

This property specifies the frequency of the tone that the beeper makes upon a successful read.

- 0 Lowest: The beep of the scanner has a very low frequency.
- 1 Low: The beep of the scanner has a low frequency.
- 2 High: The beep of the scanner has a high frequency.
- 3 Highest: The beep of the scanner has a very high frequency.

setBeeperVolume

Property Type: byte

Default: 0

Models Supported: 4696, 4697, 4698, USB OEM

This property specifies the volume of the tone that beeper makes. This property can have the following values:

- 0 Lowest: The volume of the beep is set at its lowest level.
- 1 Low: The volume of the beep is set at a low level.
- 2 Medium: The volume of the beep is set at a medium level.
- 3 High: The volume of the beep is set at a high level.

setCheckModulo

Property Type: boolean

Default: false

Models Supported: 4500, 4501, 4685

This property controls whether or not the scanner checks the modulo byte of a bar code to ensure that it is correct before returning the bar code to an application.

setCode128ScansPerRead

Property Type: byte

Default: 1

Models Supported: 4698

This property controls the minimum number of scans performed for Code 128 labels. This is the number of scans for a single pass of an item over the scanner window.

The value of this property has a range from 0 (zero) to 4. Any value outside of this range is not valid. A value of 0 (zero) indicates that the default value should be used.

setCode39ScansPerRead

Property Type: byte

Default: 1

Models Supported: 4698

This property controls the minimum number of scans performed for Code 39 labels. This is the number of scans for a single pass of an item over the scanner window.

The value of this property has a range from 0 (zero) to 4. Any value outside of this range is not valid. A value of 0 (zero) indicates that the default value should be used.

setDecodeAlgorithm

Property Type: byte

Default: 0

Models Supported: 4696, 4697, 4698

This property controls the use of decode algorithms. These decode algorithms use a complex set of tests to assemble bar code data from damaged or truncated labels. These techniques also give faster read performance on good bar codes. This property can have the following values:

0 Label assembly

1 marginless

Note: Label assembly can be selected only if Label assembly was enabled during factory configuration, if Label assembly was not enabled during factory configuration, the selection shall be considered valid but the parameter shall be forced to marginless.

setDoubleReadTimeOut

Property Type: byte

Default: 0

Models Supported: 4696, 4697, 4698, USB OEM

This property controls the length of the double-read timeout. Most scanners decode and recognize a bar code label several times as the bar code is passed through the scanning region. To prevent a scanner from returning data from the same bar code several times, scanners are programmed with a double-read timeout. The double-read timeout is the length of time that the scanner waits before returning the same bar code data twice.

This property can have the following values for 4696, 4697 and 4698:

- 0 Use 500 ms for the double-read timeout
- 70 Use 700 ms for the double-read timeout
- 90 Use 900 ms for the double-read timeout

This property can have the following values for USB:

- X'00' Use the shortest time value for the double-read timeout
- X'20' Use a moderate time value for the double-read timeout
- X'40' Use the longest time value for the double-read timeout

setDTouchMode

Property Type: boolean

Default: false

Models Supported: 4500, 4501, 4685

This property controls the state of the double-touch mode of the scanner. Double-touch mode allows scanners such as the hand-held bar code reader to read bar codes that are larger than the reading head. In double-touch mode, when the read head of the scanner is placed over the first half of the label, the scanner emits a repetitive beeping noise (if the beeper is enabled) to indicate that the data was read. When the read head is placed over the second half of the label, the data is sent to the system unit. Putting a scanner in double-touch mode does not prevent it from reading a bar code in a single touch. Only bar codes such as UPC-A, EAN-13, and UPC-D3 can be read using double-touch mode. For UPC-D3 bar codes, double-touch mode is automatically enabled by the scanner and cannot be switched off.

setEAN13ScansPerRead

Property Type: byte

Default: 1

Models Supported: 4696, 4697, 4698

This property controls the minimum number of scans performed for EAN-13 labels. This is the number of scans for a single pass of an item over the scanner window. The value of this resource has a range from one to four. Any value outside this range is not valid. A value of one indicates that the number of scans per read is not specified by the application. In this case, the default value is used.

This property can have the following values:

- 1 One scan
- 2 Two scans
- 3 Three scans
- 4 Four scans

setEAN8ScansPerRead

Property Type: byte

Default: 2

Models Supported: 4696, 4697, 4698

This property controls the minimum number of scans performed for EAN-8 labels.

This resource can have the following values:

- 1 One scan
- 2 Two scans
- 3 Three scans
- 4 Four scans

setEnabled_2_DigitSupplements

Property Type: boolean

Default: false

Models Supported: 4685, 4698, USB OEM, AnyPlace Kiosk Line scanner

This property controls whether the scanner recognizes 2-digit supplements for UPC, UPE, EAN/JAN8, and EAN/JAN13. When enabled, supplements are optional for all these symbologies.

setEnabled_5_DigitSupplements

Property Type: boolean

Default: false

Models Supported: , 4685, 4698, USB OEM, AnyPlace Kiosk Line scanner

This property controls whether the scanner recognizes 5-digit supplements for UPC-A, UPC-E, EAN/JAN8, and EAN/JAN13. When enabled, supplements are optional for all these symbologies.

setEnabledCodabar

Property Type: boolean

Default: false

Models Supported: 4501, 4685, 4697, USB OEM, AnyPlace Kiosk Line scanner, AnyPlace Kiosk Omni Scanner

Enable or disable the CodaBar barcodes.

setEnableCode128

Property Type: boolean

Default: false

Models Supported: 4501, 4685, 4697, 4698, USB OEM, AnyPlace Kiosk Line Scanner, AnyPlace Kiosk Omni Scanner

Enable or disable the Code128 barcodes.

setEnableCode128Supplements

Property Type: boolean

Default: false

Models Supported: USB OEM

Enable or disable supplementals for code128 barcodes.

setEnableCODE39

Property Type: boolean

Default: false

Models Supported: 1520, 4500, 4501, 4685, 4697, 4698, USB OEM, AnyPlace Kiosk Line Scanner, AnyPlace Kiosk Omni Scanner

Enable or disable the Code39 barcodes.

setEnableCode39CheckDigit

Property Type: boolean

Default: false

Models Supported: USB OEM

Enable or disable if check digits should be transmitted for code39 barcodes.

setEnableCode93

Property Type: boolean

Default: false

Models Supported: 4501, 4685, 4697, USB OEM

Enable or disable the Code93 barcodes.

setEnableEAN_JAN_TwoLabelDecoding

Property Type: boolean

Default: false

Models Supported: 4697, 4698, USB OEM

Enable or disable the EAN/JAN Two-label decoding.

setEnableCompositeCC_A

Property Type: boolean

Default: false

Models Supported: AnyPlace Kiosk Omni Scanner.

Enable or disable decoding of CC_A (or CC_B or CC_C) composite barcodes. As composite barcodes are 2D barcodes that appear on top of a 1D barcode (such as UPC-A, EAN13, EAN128, and so on), note that the corresponding 1D barcode must be enabled too. Enabling this or any other 2D barcode disables the omnidirectional feature because 2D barcodes cannot be scanned in omnidirectional mode.

setEnableCompositeCC_B

Property Type: boolean

Default: false

Models Supported: AnyPlace Kiosk Omni Scanner.

Enable or disable decoding of CC_A (or CC_B or CC_C) composite barcodes. As composite barcodes are 2D barcodes that appear on top of a 1D barcode (such as UPC-A, EAN13, EAN128, and so on), note that the corresponding 1D barcode must be enabled too. Enabling this or any other 2D barcode disables the omnidirectional feature because 2D barcodes cannot be scanned in omnidirectional mode.

setEnableCompositeCC_C

Property Type: boolean

Default: false

Models Supported: AnyPlace Kiosk Omni Scanner.

Enable or disable decoding of CC_A (or CC_B or CC_C) composite barcodes. As composite barcodes are 2D barcodes that appear on top of a 1D barcode (such as UPC-A, EAN13, EAN128, and so on), note that the corresponding 1D barcode must be enabled too. Enabling this or any other 2D barcode disables the omnidirectional feature because 2D barcodes cannot be scanned in omnidirectional mode.

setEnableGoodReadBeep

Property Type: boolean

Default: false

Models Supported: All

Enable or disable GoodReadBeep.

setEnableInterleaved2of5

Property Type: boolean

Default: false

Models Supported: 1520, 4500, 4501, 4685, 4697, 4698, USB OEM, AnyPlace Kiosk Line Scanner, AnyPlace Kiosk Omni Scanner

Enable or disable the Interleaved 2-of-5 (ITF) barcodes.

setEnableITFCheckDigit

Property Type: boolean

Default: false

Models Supported: USB OEM

Enable or disable if check digits should be transmitted for Interleaved 2-of-5 (ITF) barcodes.

setEnablePDF417

Property Type: boolean

Default: false

Models Supported: AnyPlace Kiosk Omni Scanner

Enable or disable decoding of PDF417 barcodes. Enabling this or any other 2D barcode disables the omnidirectional feature because 2D barcodes cannot be scanned in omnidirectional mode.

setEnableMicroPDF417

Property Type: boolean

Default: false

Models Supported: AnyPlace Kiosk Omni Scanner

Enable or disable decoding of Micro PDF417 barcodes. Enabling this or any other 2D barcode disables the omnidirectional feature because 2D barcodes cannot be scanned in omnidirectional mode.

setEnableRSS14 (deprecated)

Property Type: boolean

Default: false

Models Supported: AnyPlace Kiosk Line Scanner, AnyPlace Kiosk Omni Scanner

Enables or disables the Reduced Space Symbology RSS14 barcodes.

setEnableGS1Databar

Property Type: boolean

Default: false

Models Supported: AnyPlace Kiosk Line Scanner, AnyPlace Kiosk Omni Scanner

Enables or disables the GS1 Databar barcodes.

setEnableRSS_Expanded (deprecated)

Property Type: boolean

Default: false

Models Supported: AnyPlace Kiosk Line Scanner, AnyPlace Kiosk Omni Scanner

Enables or disables the RSS Expanded barcodes.

setEnableGS1Databar_Expanded

Property Type: boolean

Default: false

Models Supported: AnyPlace Kiosk Line Scanner, AnyPlace Kiosk Omni Scanner

Enables or disables the GS1 Databar Expanded barcodes.

setEnableStandard2of5

Property Type: boolean

Default: false

Models Supported: AnyPlace Kiosk Line scanner, AnyPlace Kiosk Omni scanner

Enable or disable the Standard 2-of-5 (STF) barcodes.

setEnableUCC_EAN128

Property Type: boolean

Default: false

Models Supported: USB OEM, AnyPlace Kiosk Line Scanner, AnyPlace Kiosk Omni Scanner

Enable or disable the UCC/EAN128 barcodes.

setEnableUPC_A_CheckDigit

Property Type: boolean

Default: false

Models Supported: USB OEM, AnyPlace Kiosk Line scanner, AnyPlace Kiosk Omni scanner

Enable or disable if check digits should be transmitted for UPC-A barcodes.

setEnableUPC_A_To_EAN13Expansion

Property Type: boolean

Default: false

Models Supported: 4696, 4697, 4698, USB OEM, AnyPlace Kiosk Line scanner

This property controls the report format for UPC-A labels. UPC-A and EAN-13 are part of the same numbering system. It is possible to have the scanner report all of these codes in a uniform format. UPC-A is a 12-digit subset of EAN-13. The scanner can add a leading 0 (zero) to the UPC-A number, yielding its EAN equivalent.

setEnableUPC_E_CheckDigit

Property Type: boolean

Default: false

Models Supported: USB OEM, AnyPlace Kiosk Line scanner, AnyPlace Kiosk Omni scanner

Enable or disable if check digits are transmitted for UPC-E barcodes.

setEnableUPC_E_To_EAN13Expansion

Property Type: boolean

Default: false

Models Supported: 4696, 4697, 4698, USB OEM

This property controls the report format for UPC-E labels. UPC-E and EAN-13 are part of the same numbering system. It is possible to have the scanner report all of these codes in a uniform format. UPC-E is a short form of a UPC-A number, UPC-A is a 12-digit subset of EAN-13. The scanner can add a leading 0 (zero) to the UPC-E number, yielding its EAN equivalent.

setEnableUPC_E_To_UPC_AExpansion

Property Type: boolean

Default: false

Models Supported: 4698, USB OEM, AnyPlace Kiosk Omni Scanner

This controls the report format for UPC-E labels. UPC-A, UPC-E, and EAN-13 are all part of the same numbering system. It is possible to have the scanner report all of these codes in a uniform format. The scanner can expand UPC-E data to its UPC-A format.

setEnableUPCAE_EANJAN813

Property Type: boolean

Default: false

Models Supported: All

Enable or disable the UPC-A, UPC-E, EAN/JAN 8 and EAN/JAN 13 barcodes.

setEnableUPCD1D5

Property Type: boolean

Default: false

Models Supported: 4501, 4696, 4697, 4698, USB OEM

Enable or disable the UPC D1 to UPC D5 barcodes.

setEnableVerificationUPC_A_EAN13_fiveDigit

Property Type: boolean

Default: false

Models Supported: 4696, 4697, 4698, USB OEM

This controls whether the scanner verifies the 5 digit price check character. UPC and EAN specifications allow for a price check character to be included in the digits encoded on in-store random weight items. This property is mutually exclusive with UPC-A, EAN-13 four Digit Verification.

setEnableVerificationUPC_A_EAN13_fourDigit

Property Type: boolean

Default: false

Models Supported: 4696, 4697, 4698, USB OEM

This property controls whether the scanner verifies the 4 digit price check character. UPC and EAN specifications allow for a price check character to be included in the digits encoded on in-store random weight items. This property is mutually exclusive with UPC-A, EAN-13 five Digit Verification.

setITFLength1

Property Type: byte

Default: 0

Models Supported: 1520, 4697, 4698, USB OEM. AnyPlace Kiosk Line scanner, AnyPlace Kiosk Omni scanner

For the scanners that support Interleaved 2-of-5 bar codes, this property can be used to specify one valid length for Interleaved 2-of-5 bar codes. This value indicates the exact length of the Interleaved 2-of-5 bar codes that the scanner reads. If an Interleaved 2-of-5 bar code is not of the correct length, then the bar code is not read by the scanner. The value of the setITFLength1 property must be an even number from 4 to 32. (For the Toshiba 1520-A02, the value must be an even number from 4 to 30.) This value specifies the exact length of the bar code.

If the scanner is not configured to read Interleaved 2-of-5 bar codes, the value of this property is ignored. This property is provided because scanners are prone to errors when reading Interleaved 2-of-5 labels. Because these labels are of variable length, it is possible for a scanner to read only part of a label, but process it as though it had read the complete label. If an application is only expecting Interleaved 2-of-5 labels of a certain length, this property ensures that the scanner does not read any partial labels.

setITFLength2

Property Type: byte

Default: 0

Models Supported: 1520, 4697, 4698, USB OEM, AnyPlace Kiosk Line scanner, AnyPlace Kiosk Omni scanner

For scanners that support Interleaved 2-of-5 bar codes, this property can be used to indicate a second valid length for Interleaved 2-of-5 bar codes. When reading Interleaved 2-of-5 bar codes, these scanners can read only bar codes of one or two specific lengths. The setITFLength1 and setITFLength2 properties are used to specify the Interleaved 2-of-5 lengths to be recognized and read by these scanners. The value of the setITFLength2 property must be an even number from 4 to 32. This value specifies the exact length of the bar code. A value of 0 (zero) indicates that only Interleaved 2-of-5 bar codes of the length specified in the iTFLength1 resource are read by the scanner. If the scanner is not configured to read Interleaved 2-of-5 bar codes, the value of this property is ignored.

setITFLengths

Property Type: boolean

Default: false

Models Supported: USB OEM, AnyPlace Kiosk Line scanner, AnyPlace Kiosk Omni scanner

Valid values:

true ITF lengths represent a range.

false ITF lengths are discrete.

Note: For the AnyPlace Kiosk Line scanner, *true* indicates that ITF lengths specifies a minimum value only (iTFLength1).

setITFLengthSpecifiedTwo

Property Type: boolean

Default: false

Models Supported: 4697, 4698, USB OEM

This property specifies if two ITF lengths are specified.

setITFScansPerRead

Property Type: byte

Default: 1

Models Supported: 4698

This property controls the minimum number of scans performed for Interleaved 2-of-5 labels. This is the number of scans for a single pass of an item over the scanner window.

The value of this property has a range from 0 (zero) to 4. Any value outside of this range is not valid. A value of 0 (zero) indicates that the default value should be used.

setLaserTimeOut

Property Type: byte

Default: 0

Models Supported: 4696, 4697, 4698, USB OEM

This property specifies the length of the period of inactivity that causes a laser scanner to turn off its laser. This property can have the following values for 4696, 4697 and 4698:

- 0 The laser turns off after 15 minutes of inactivity
- 5 The laser turns off after 5 minutes of inactivity
- 10 The laser turns off after 10 minutes of inactivity
- 15 The laser turns off after 15 minutes of inactivity

This property can have the following values for USB OEM:

- X'00' The laser always stays on
- X'08' The laser turns off after 5 minutes of inactivity
- X'10' The laser turns off after 10 minutes of inactivity
- X'18' The laser turns off after 15 minutes of inactivity

Note: For motor-driven laser scanners, the motor is always on if the laser is on.

setLED_GoodRead_Duration

Property Type: byte

Default: 0

Models Supported: USB OEM

This property specifies the LED good read duration. This property can have the following values:

- X'00' Use a short time value for LED good read duration
- X'01' Use a medium time value for LED good read duration
- X'02' Use a long time value for LED good read duration
- X'03' Use the longest time value for LED good read duration

setMotorTimeOut

Property Type: byte

Default: 0

Models Supported: 4696, 4697, 4698, USB OEM

This property specifies the length of the period of inactivity that causes a motorized laser scanner to turn off its motor. This property can have the following values for 4696, 4697 and 4698:

- 0 The motor turns off after 60 minutes of inactivity
- 5 The motor turns off after 5 minutes of inactivity
- 10 The motor turns off after 10 minutes of inactivity
- 15 The motor turns off after 15 minutes of inactivity
- 30 The motor turns off after 30 minutes of inactivity
- 60 The motor turns off after 60 minutes of inactivity

This property can have the following values for USB OEM:

- X'00' The motor is always on
- X'01' The motor turns off after 5 minutes of inactivity
- X'02' The motor turns off after 10 minutes of inactivity
- X'03' The motor turns off after 15 minutes of inactivity
- X'04' The motor turns off after 30 minutes of inactivity
- X'05' The motor turns off after 60 minutes of inactivity

setSecurityLevelForInStore

Property Type: byte

Default: 0

Models Supported: USB OEM

This property specify the security/integrity level for In-Store labels. This property can have the following values:

- X'00' Use a low security/integrity level.
- X'01' Use a medium security/integrity level.
- X'02' Use a high security/integrity level.
- X'03' Use the highest security/integrity level.

setSTFLength1

Property Type: byte

Default: 0

Models Supported: AnyPlace Kiosk Line scanner, AnyPlace Kiosk Omni scanner

This property allows you to set the first Standard 2 of 5 length to be accepted if STFLengths = false (discrete) or the minimum length to be accepted if STFLengths = true (range)

setSTFLength2

Property Type: byte

Default: 0

Models Supported: Line scanner, Omni scanner

This property allows you to set the second Standard 2 of 5 length to be accepted if STFLengths = false (discrete) If STFLengths = true (range) STFLength2 is the maximum value to be accepted for the Omni scanner. For the Line scanner, this value is meaningless.

setSTFLenghts

Property Type: boolean

Default: false

Models Supported: AnyPlace Kiosk Line scanner, AnyPlace Kiosk Omni scanner

Valid values:

- true (range)
 - For Line scanners, STF Length1 represent a minimum length to be accepted. STF Length2 is ignored.
 - For Omni scanners, STF Length1/2 represent a minimum/maximum length to be accepted.
- false (discrete) STF Length1 and Length2 are the only lengths to be accepted

Note: To accept any length STF, this property must be set to *true* and STFLength1 to 0.

setStoreScansPerRead

Property Type: byte

Default: 1

Models Supported: 4696, 4697, 4698

This property controls the minimum number of scans performed for in-store labels. This is the number of scans for a single pass of an item over the scanner window.

The value of this property has a range from 0 (zero) to 4. Any value outside of this range is not valid. A value of 0 (zero) indicates that the number of scans per read is not specified by the application.

setSupplements

Property Type: boolean

Default: false

Models Supported: 4698, AnyPlace Kiosk Omni scanner

This property controls whether the scanner recognizes 2-digit supplementals and 5 digit supplementals for UPCA, UPCE, EAN/JAN8, AND EAN/JAN13. When enabled, supplementals are optional for all these symbologies.

setSupplementsSecurityLevel

Property Type: byte

Default: 10

Models Supported: AnyPlace Kiosk Line scanner, AnyPlace Kiosk Omni scanner

This property specify the security level for UPC-A, UPC-E, EAN/JAN-8, and EAN/JAN-13 supplementals. Making this value higher reduces the chance of "short readings" but increases the difficulty of reading low-quality barcodes. This property can have values in the range 0 - 100.

setTransmitScannerPrefixAndSuffix

Property Type: boolean

Default: false

Models Supported: AnyPlace Kiosk Line scanner, AnyPlace Kiosk Omni scanner.

Scanner data must be transmitted without any prefixes or suffixes. If they are required, set this property to true. Note that there is no property that allows to set them to a particular char sequence (such as suffix = CR), so they must be set using barcode programming

setUPCAsScansPerRead

Property Type: byte

Default: 1

Models Supported: 4696, 4697, 4698

This property controls the minimum number of scans performed for UPC-A labels. This is the number of scans for a single pass of an item over the scanner window. The value of this property has a range from 0 (zero) to 4. Any value outside of this range is not valid. A value of 0 (zero) indicates that the number of scans per read is not specified by the application.

setUpCDScansPerRead

Property Type: byte

Default: 1

Models Supported: 4696, 4697, 4698

This property controls the minimum number of scans performed for UPC-D labels. This is the number of scans for a single pass of an item over the scanner window.

The value of this property has a range from 0 (zero) to 4. Any value outside of this range is not valid. A value of 0 (zero) indicates that the number of scans per read is not specified by the application.

setUpCEScansPerRead

Property Type: byte

Default: 2

Models Supported: 4696, 4697, 4698

This property controls the minimum number of scans performed for UPC-E labels. This is the number of scans for a single pass of an item over the scanner window.

The value of this property has a range from 0 (zero) to 4. Any value outside of this range is not valid. A value of 0 (zero) indicates that the number of scans per read is not specified by the application. This property can have the following values:

- 1 One scan
- 2 Two scans
- 3 Three scans
- 4 Four scans

Additional JavaPOS information

Toshiba 4697 Point of Sale Scanner Configuration

The Toshiba 4697 Point of Sale Scanner Model 1 follows the following rules in the symbology configuration:

1. No more than 2 of the supported symbology options may be included in the JPOS.XML entry for the scanner.

2. Only one of these two symbology options is allowed to be an industrial symbology. Choosing more than one industrial symbology option is considered invalid.
3. One of the non-industrial symbology must be included.

Non-industrial symbologies are considered to be:

- UPC-A/E, EAN/JAN-8/13
- UPC-A/E, EAN/JAN-8/13, UPC-D1..D5

Industrial symbologies are considered to be:

- ITF (Interleaved 2 of 5)
- CODABAR
- CODE 39
- CODE 93
- CODE 128

Example 1 - To enable one symbology with UPC-A/E, EAN/JAN-8/13, UPC-D1..D5 add following properties to the JposEntry:

```
<prop name="setEnableUPCAE_EANJAN813" type="Boolean" value="true"/>
<prop name="setEnableUPCD1D5" type="Boolean" value="true"/>
```

Example 2 - To enable two symbologies with UPC-A/E, EAN/JAN-8-13 and ITF add following properties to the JposEntry:

```
<prop name="setEnableUPCAE_EANJAN813" type="Boolean" value="true"/>
<prop name="setEnableInterleaved20f5" type="Boolean" value="true"/>
<prop name="setITFLength1" type="Byte" value="12"/>
```

USB OEM Scanner and scale configuration

The Toshiba JavaPOS supports new USB scanner and scale devices that comply with Toshiba or USB OEM Point-of-Sale Device interface specifications. Support for these devices required modification to configuration files. These configuration files currently include entries for known USB OEM Scanner devices. for a new device, the configuration files must be modified to include the VendorID and ProductID of that device.

For the Windows environment, two configuration files must be modified: javaxusb.inf and posj.properties.

For Linux, only posj.properties must be changed to include the VendorID and ProductID values.

Install javaxusb driver for OEM scanner/scale (Windows Only)

The Toshiba JavaPOS drivers automatically install javaxusb driver for selected USB OEM scanner and scale devices which comply to Toshiba USB OEM Point of Sale Device Interface Specifications. The OEM devices can be added or removed from javaxusb driver. An "aipdfix" utility is provided to add/remove devices from javaxusb driver.

Syntax of the utility:

Usage: aipdfix [-i <inf-file> | -u <inf-file>] [-q] [-v]

where:

- i <inf-file> Inf file to install
- u <inf-file> Inf file to uninstall
- q No installation messages
- v Verbose messages
- f force (hidden)

Note: Requires either "-i" or "-u" option

1. Copy driver file:
 - a. Copy c:\pos\drivers\javaxusb\javaxusb.sys to c:\pos\drivers\oem

Note: The inf file and javaxusb.sys driver files must be in the same directory.
2. To add new USB OEM device entry
 - a. Determine VID and PID for the USB OEM device, using an utility such as UsbView
 - b. Add a new device entry to javaxusb_oem.inf file
 - 1) Edit c:\pos\drivers\oem\javaxusb_oem.inf file
 - 2) Under the Manufacturer section, add a new entry for the OEM device
 - copy the template line "JavaxUsb <Description> Scanner"=JAVAXUSB_COMPOSITE, USB\VID_0000&PID_1111
 - remove ';' comment symbol at the beginning of line
 - update the required fields in the line:
 - change <Description> to say, for example "Magellan 8500XT"
 - change 0000 to VID of OEM Device
 - change 1111 to PID of OEM device
 3. Install javaxusb-oem driver
 - a. From command prompt, enter c:\pos\bin\apdidx -i c:\pos\drivers\oem\javaxusb_oem.inf -f
 4. Add USB OEM device entry to posj.properties file to associate it with Toshiba JavaPOS drivers.
 - a. Edit c:\pos\ibmjpos\config\posj.properties file
 - b. Under "OEM Scanner" section, copy the last line and modify it to include the VID/PID.

Note: Increment the XX number "knownPosDevice.XX" from previous number.
Example: com.ibm.posj.bus.hid(javaxusb.factory.knownPosDevices.89 = 0xXXXX,0xYYYY where
0xXXX = VID, 0xYYYY = PID
 5. Reboot the system

Uninstall javaxusb driver for an existing USB OEM scanner/scale (Windows only)

1. First, uninstall the existing javaxusb driver on the system. From command prompt, enter apdidx -u c:\pos\drivers\javaxusb\javaxusb.inf -f
2. Reboot system to ensure the associated driver inf files are removed by Operating System
3. Remove usb device entry from javaxusb.inf file
 - a. Edit c:\pos\drivers\javaxusb\javaxusb.inf file
 - b. Under "OEM Scanner" section, find and delete the JavaxUsb entry of interest.
Example: To disassociate scanner with VID=0x05E0 and PID = 0x1300, delete the following entry and save the file.
"JavaxUsb Symbol LS2280" =JAVAXUSB_COMPOSITE, USB\VID_05E0&PID_1300
4. Install javaxusb driver for the modified javaxusb.in file
 - a. From command prompt, enter apdidx -i c:\pos\drivers\javaxusb\javaxusb.inf -f
5. Remove OEM device entry from posj.properties file to disassociate the device from Toshiba JavaPOS drivers
 - a. Edit c:\pos\ibmjpos\config\posj.properties file
 - b. Under "OEM Scanner" section, find and delete the javaxusb entry associated with the device of interest. The entries are based on VID/PID pair.
Example: Delete the following entry for the device removed in step 3 above.
com.ibm.posj.bus.hid(javaxusb.factory.knownPosDevices.80 = 0x05e0,0x1300
6. Reboot the system

Debugging:

In case of uninstall/install errors, please use -v option to get more details.

Example: aipdifax -u c:\pos\drivers\javaxusb\javaxusb.inf -v -f

List of USB OEM Devices currently associated with javaxusb driver

Table 243. OEM Scanners

OEM Scanners	
"JavaxUsb PSC Magellan 8500 Scanner"	=JAVAXUSB_COMPOSITE, USB\VID_05f9&PID_0101
"JavaxUsb PSC Magellan Scanner"	=JAVAXUSB_COMPOSITE, USB\VID_05f9&PID_0110
"JavaxUsb Symbol Scanner"	=JAVAXUSB_COMPOSITE, USB\VID_05e0&PID_0101
"JavaxUsb Symbol LS2104"	=JAVAXUSB_COMPOSITE, USB\VID_05E0&PID_0200
"JavaxUsb Symbol M2004-I000"	=JAVAXUSB_COMPOSITE, USB\VID_05E0&PID_0300
"JavaxUsb Symbol LS4008I"	=JAVAXUSB_COMPOSITE, USB\VID_05E0&PID_038A
"JavaxUsb Symbol LS2280"	=JAVAXUSB_COMPOSITE, USB\VID_05E0&PID_1300
"JavaxUsb XAC Protocol Converter"	=JAVAXUSB_COMPOSITE, USB\VID_0886&PID_9900
"JavaxUsb IBM Scanner"	=JAVAXUSB_COMPOSITE, USB\VID_0886&PID_9901
"JavaxUsb IBM Scanner"	=JAVAXUSB_COMPOSITE, USB\VID_0886&PID_9902
"JavaxUsb Magellan 9500 Scanner"	=JAVAXUSB_COMPOSITE, USB\VID_05f9&PID_0180
"JavaxUsb Magellan 8502 Scanner"	=JAVAXUSB_COMPOSITE, USB\VID_05f9&PID_1101
"JavaxUsb MetroLogic Scanner"	=JAVAXUSB_COMPOSITE, USB\VID_0C2E&PID_0009
"JavaxUsb PSC 1000i Scanner"	=JAVAXUSB_COMPOSITE, USB\VID_05f9&PID_1601
"JavaxUsb HPP IT5600 Scanner"	=JAVAXUSB_COMPOSITE, USB\VID_0536&PID_0163

Update posj.properties configuration file

For both Windows and Linux environments, the posj.properties file must be updated to include VendorID and ProductID for the USB OEM scanner.

1. Obtain VendorID and ProductID for your device using a USB utility, such as UsbView.
2. Edit the posj.properties file, located at:
 - Windows: C:\pos\ibmjpos\config
 - Linux: /opt/ibm/javapos/etc
3. In the OEM Scanner section, copy the last entry and paste it to the end of that section.
4. Modify the VendorID and the ProductID to match your device.
5. Increment the numeric digit of the entry.

For example, if your new scanner has Vendor ID 0x1234 and Product ID 0x5678, and the last entry was com.ibm.posj.bus.hid.javaxusb.factory.knownPosDevice.72, then add the following entry:

```
com.ibm.posj.bus.hid.javaxusb.factory.knownPosDevice.73 = 0x1234,0x5678
```

Known scanner issues

This section contains information on known issues with specific models.

PSC 384 Magellan - USB

SetEnableCodabar and SetEnableCode93 configuration parameters must not be enabled. The scanner does not work correctly if these are enabled.

Toshiba 4698

- RS-485 through USB protocol converter
- SetEnableCodabar, SetEnableCode93, and SetEnableUCC_EAN128 configuration parameters must not be enabled. The scanner does not work correctly if these are enabled.
- Configuration must be correct for the scanner type attached, even though it looks like a USB scanner.

OPOS configuration

Scanner programming

RS-485 and USB scanner configuration can be modified programmatically using the resource file, AIPSY.S.RES, as mentioned earlier in “Modifying USB, RS-485, and PS/2 device behavior” on page 46. This allows the scanner configuration to be changed without having to physically reprogram every scanner by hand using the manufacturer’s programming barcodes.

The resource file will probably not support every possible configuration option that a particular scanner supports. The resource file support is based on older scanner hardware specifications and most scanner manufacturers have added features beyond those specifications. You may still have to use the manufacturer’s programming barcodes to enable/disable some options for your scanners.

There are many configuration options or resources that you can set in the resource file for the scanner. They are all documented in Chapter 21 "Resource Sets" of *Point of Sale Subsystem: Programming Reference and User’s Guide* (SC30-3560). The resources that are typically set for a scanner are:

```
*barCodes1  
*barCodes2  
*barCodes3  
*barCodes4  
*iTFLength1  
*iTFLength2  
*iTFLengthType
```

The barCodesX resources determine what types of barcodes the scanner will read. The iTFLengthXXX resources apply specifically to Interleaved 2 of 5 (ITF) barcodes and determine what subset of ITF barcodes the scanner will read.

Scanners are identified by the driver as being one of two types, hand-held or flatbed (table top). Each of these types is further divided into different subtypes based upon which Toshiba scanner interface is being emulated. For each Toshiba scanner interface, only certain resources can be used and for each resource, only certain values can be specified.

For example, an RS485 scanner configured for port 17B will report itself to the terminal (and have different configuration options) than the same physical scanner configured for port 5B.

If the resource file does not contain any scanner configuration information, the driver still sends a default configuration command to the scanner. See Table 244 and Table 245 on page 285 for the default configuration for each scanner subtype.

Table 244. Handheld scanners

Subtype	Description	Subset of valid resources	Default configuration
1	Toshiba Hand Held Bar Code Reader model 1 (HHBCR1)	*barCode1	UPC-A, UPC-E, EAN-8, EAN-13, JAN-8, JAN-13
2	Toshiba Hand Held Bar Code Reader model 2 (HHBCR2)	*barCode1	UPC-A, UPC-E, EAN-8, EAN-13, JAN-8, JAN-13

Table 244. Handheld scanners (continued)

Subtype	Description	Subset of valid resources	Default configuration
3	Toshiba Model 1520	*barCode1 *barCode2 *barCode3 *barCode4 *iTFLength1	UPC-A, UPC-E, UPC-D, UPC-8, UPC-13
4	USB scanner	*barCode1 *barCode2 *barCode3 *barCode4 *iTFLength1 *iTFLength2 *iTFLengthType	UPC-A, UPC-E, UPC-D, EAN-8, EAN-13, JAN-8, JAN-13

Table 245. Flatbed scanners

Subtype	Description	Subset of valid resources	Default configuration
1	Toshiba 4687	Cannot be programmed via the resource file	UPC-A, UPC-E, UPC-D, EAN-8, EAN-13, JAN-8, JAN-13
2	Toshiba 4686 model 1 & 2	*barCode1 *barCode2 *iTFLength1 *iTFLength2	UPC-A, UPC-E, UPC-D, EAN-8, EAN-13, JAN-8, JAN-13
3	Toshiba 4686 model 3 & 4	*barCode1 *barCode2 *iTFLength1 *iTFLength2	UPC-A, UPC-E, UPC-D, EAN-8, EAN-13, JAN-8, JAN-13
4	Toshiba 4696	*barCode1	UPC-A, UPC-E, UPC-D, EAN-8, EAN-13, JAN-8, JAN-13
5	Toshiba 4697	*barCode1 *barCode2 *iTFLength1 *iTFLength2	UPC-A, UPC-E, UPC-D, EAN-8, EAN-13, JAN-8, JAN-13
6	Toshiba 4698	*barCode1 *barCode2 *barCode3 *barCode4 *iTFLength1 *iTFLength2	UPC-A, UPC-E, UPC-D, EAN-8, EAN-13, JAN-8, JAN-13
7	USB scanner	*barCode1 *barCode2 *barCode3 *barCode4 *iTFLength1 *iTFLength2 *iTFLengthType	UPC-A, UPC-E, UPC-D, EAN-8, EAN-13, JAN-8, JAN-13

Following are several scanner configuration examples:

- One Toshiba 1520-compatible hand-held scanner is attached and it must be able to read Code 39 and UPC barcodes. The resource file should contain the following:

```
*barCodes1: LGROUP_UPC_EAN
*barCodes2: LGROUP_CODE_39
```

- One Toshiba USB OEM POS Device Interface-compliant scanner is attached and it needs to be able to read Code 39, Code 128, UPC, EAN and ITF barcodes (4 to 32 digits (inclusive) in length). The resource file should contain the following:

```
*barCodes1: LGROUP_UPC_EAN_D1_T0_D5
*barCodes2: LGROUP_CODE_39
*barCodes3: LGROUP_INT_2_OF_5
*itfLength1: 4
*itfLength2: 32
*itfLengthType: 1
```

- One Toshiba USB OEM POS Device Interface-compliant scanner is attached and it needs to be able to read Code 39, Code 128, UPC, EAN and ITF barcodes (20 and 24 digits in length only). The resource file should contain the following:

```
*barCodes1: LGROUP_UPC_EAN_D1_T0_D5
*barCodes2: LGROUP_CODE_39
*barCodes3: LGROUP_INT_2_OF_5
*itfLength1: 20
*itfLength2: 24
*itfLengthType: 0
```

- One Toshiba HHBCR2-compatible hand-held scanner is attached and it needs to be able to read ITF barcodes. The resource file should contain the following:

```
*barCodes1: LGROUP_INT_2_OF_5
```

- One Toshiba HHBCR2-compatible hand-held scanner is attached and it needs to be able to read UPC, EAN and ITF barcodes. The resource file should contain the following:

```
*barCodes1: LGROUP_UPC_EAN_ITF
```

- One Toshiba HHBCR2-compatible hand-held scanner is attached and it needs to be able to read Code 128, UPC, EAN and ITF barcodes. The resource file should contain the following:

```
*barCodes1: 0
```

Note: For Toshiba HHBCR2-compatible scanners, 0 (zero) specifies that the scanner should be placed in **test** mode which allows it to read all barcode types that it supports.

- One RS485 Toshiba HHBCR2-compatible hand-held scanner (OPOS name, **HH485**) and one Toshiba USB OEM POS Device Interface-compliant flatbed scanner (OPOS name, **FBUSB**) are attached; both scanners must be able to read Code 39, Code 128, UPC, and EAN barcodes. The resource file should contain the following:

```
*HH485.barCodes1: 0
```

```
*FBUSB.barCodes1: LGROUP_UPC_EAN
*FBUSB.barCodes1: LGROUP_CODE_39_CODE_128
```

- One Toshiba 1520-compatible hand-held scanner is attached and it needs to be able to read Code 39 and 20-digit ITF barcodes. The resource file should contain the following:

```
*barCodes1: LGROUP_UPC_EAN
*barCodes2: LGROUP_INT_2_OF_5
*itfLength1: 20
```

Scanner Programming (AnyPlace Kiosk)

The scanners on AnyPlace Kiosk are programmed from the OPOS Configuration Utility.

From the OPOS Configuration utility, click **SurePOS 300/500/600** and open the **Kiosk** tab. Select **AnyPlace Kiosk scanner** and click **Configure**.

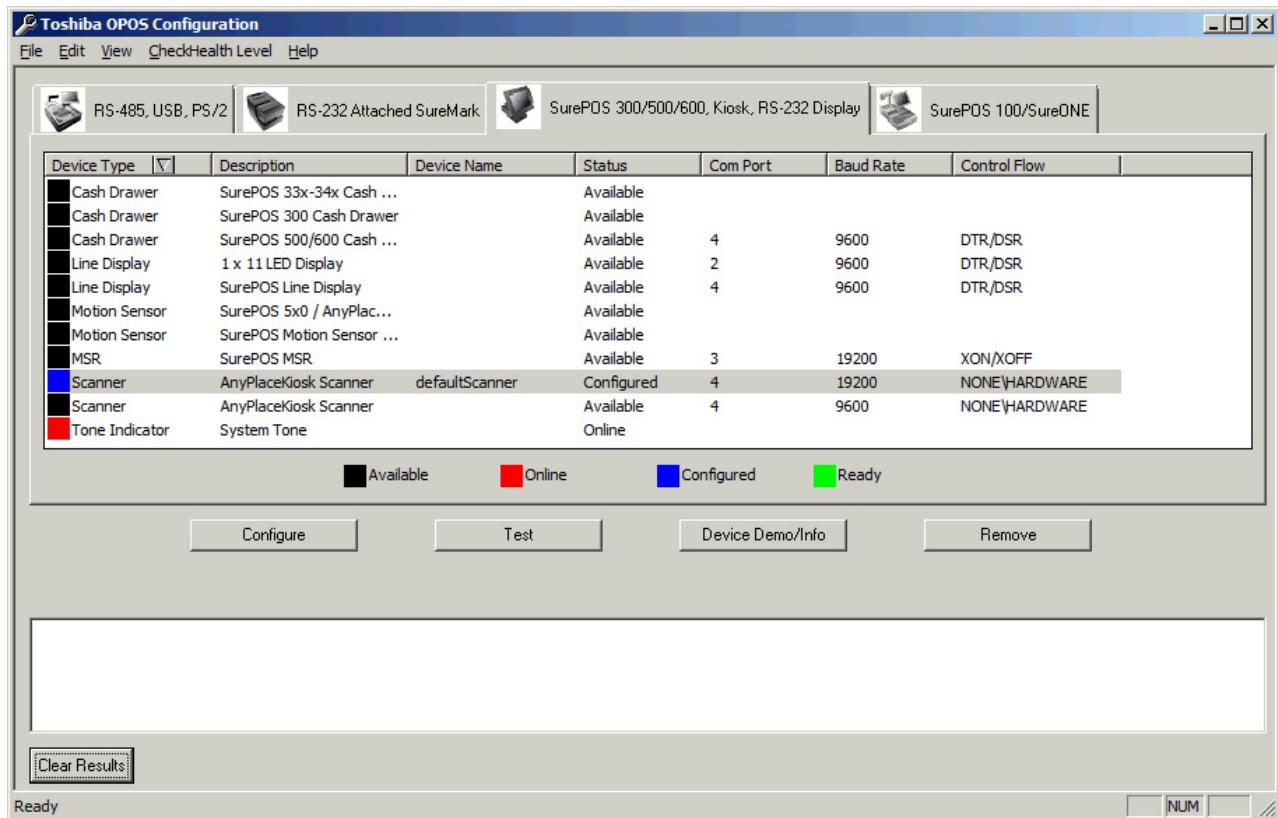


Figure 38. AnyPlace Kiosk Scanner

Enter Device Name and select Scanner Type. You can change the value of a property by clicking on Value and by selecting TRUE or FALSE as shown below.

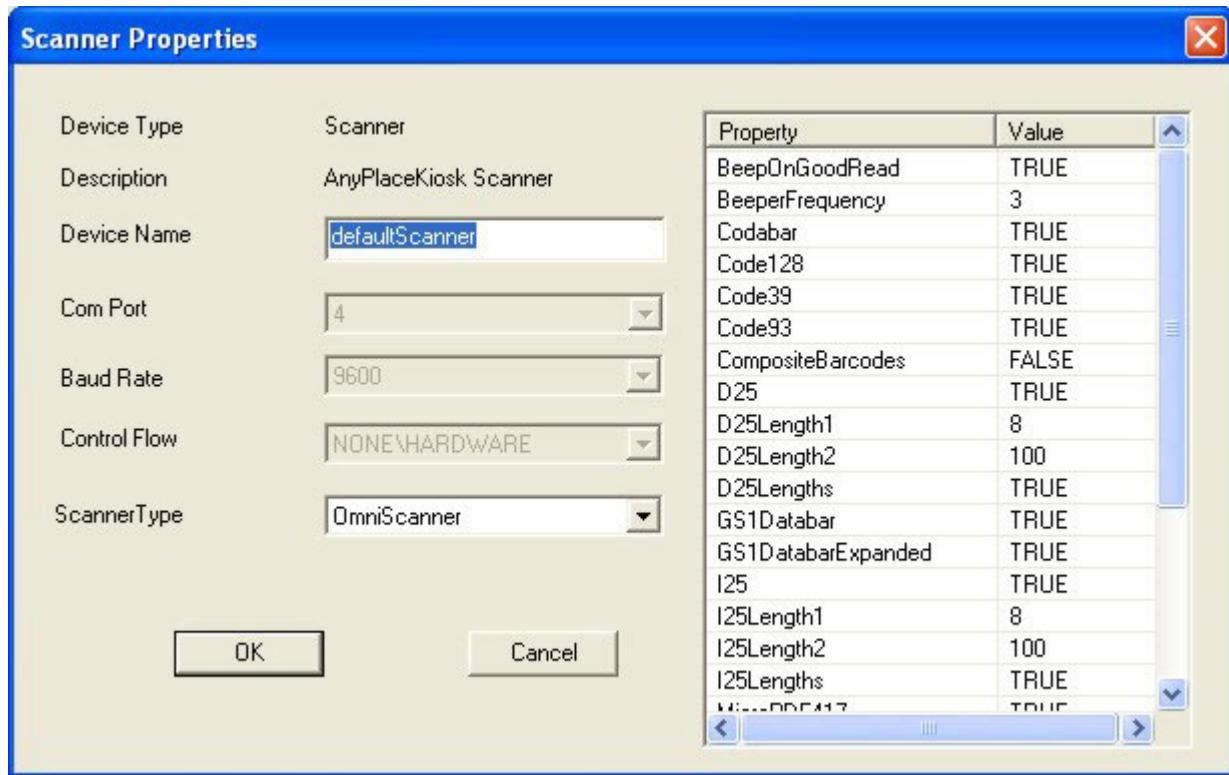


Figure 39. Scanner Properties

Refer to the table for available properties for AnyPlace scanners.

Table 246. AnyPlace Scanner Properties

Scanner Type	Properties
Anyplace Kiosk - Line Scanner	<p>Set to TRUE (default)</p> <p>BepOnGoodRead, Codabar, Code39, Code93, Code128, D25, D25Lengths, GS1Databar, GS1DatabarExpanded, I25, I25Lengths, MicroPDF417, Transmit_UPCA_CheckDigit, Transmit_UPCE_CheckDigit, UCC/EAN128, UPC/EAN Supplemental, UPC-A, UPC-E, EAN-8, EAN-13, JAN-8, JAN-13</p> <p>Set to FALSE (default)</p> <p>PDF417, CompositeBarcodes, ProgrammingViaBarcode</p>
Anyplace Kiosk - Omni Scanner	<p>Set to TRUE (default)</p> <p>BepOnGoodRead, Codabar, Code128, Code39, Code93, D25, D25Lengths, GS1Databar, GS1DatabarExpanded, I25, I25Lengths, MicroPDF417, Transmit_UPCA_CheckDigit, Transmit_UPCE_CheckDigit, UCC/EAN128, UPC/EAN Supplemental, UPC-A, UPC-E, EAN-8, EAN-13, JAN-8, JAN-13</p> <p>Set to FALSE (default)</p> <p>PDF 417, CompositeBarcodes ProgrammingViaBarcode</p>

Additional OPOS information

Scanner detection

The Toshiba OPOS driver will control all RS-485 scanners, and can optionally control USB scanners.

RS-485 scanners are detected automatically by the driver, and do not need any special programming in order to be detected. The only programming that might be required is to alter the model of legacy Toshiba scanner that the physical scanner is emulating, since altering the model of scanner will change what configuration options you can use to program the scanner through the driver.

USB scanners will be detected automatically if the scanner has been programmed as an Toshiba-compatible device. If it has not been programmed as such, you must use the scanner vendors own scanner driver.

Some scanner vendors have implemented their driver with the scanner programmed as an Toshiba-compatible device. In this situation, both the vendors own driver and the Toshiba driver will send programming commands to the device, since both drivers recognize the scanner as a device that they should control.

To work around this problem refer to "Ignoring USB devices" on page 46.

Known scanner issues

This section contains information on known issues with specific models.

Motorola DS9808

There are two issues with scanner support of the 2D barcode symbologies:

- Four types of 2D symbologies can be scanned and return correct encoded information, but are unable to return correct symbology types:
 - Data Matrix
 - GS1 Data Matrix
 - QR Code
 - Aztech Code
- Six types of 2D symbologies are unable to scan:
 - MicroPDF
 - Maxicode
 - OCR-A
 - OCR-B
 - GS1 QR Code
 - Code 49

Chapter 18. Scale

Supported devices

Table 247. Scale supported devices

Device	Connectivity	Comments
1. Toshiba 4687 Compatible Scanner/scale	RS-485, USB	
2. Toshiba 4696 Compatible Scanner/scale	RS-485, USB	
3. Toshiba 4698 Compatible Scanner/scale	RS-485, USB	
4. OEM Scale	USB	Must conform to <i>Toshiba USB OEM Point-of-Sale Device Interface Specification V1.0 or later.</i>

Supported properties and methods

Table 248. Scale common properties

Property	JavaPOS	OPOS
AutoDisable		Not supported
BinaryConversion	Not supported	All
CapCompareFirmwareVersion		False
CapPowerReporting	PR_STANDARD	All support PR_STANDARD
CapStatisticsReporting	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CapUpdateFirmware		False
CapUpdateStatistics	False (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CheckHealthText		All
Claimed		All
DataCount		All
DataEventEnabled		All
DeviceControlDescription		All
DeviceControlVersion		All
DeviceEnabled		All
DeviceServiceDescription		All
DeviceServiceVersion		All
FreezeEvents		All
OpenResult	Not supported	All
OutputID		Not supported
PowerNotify		All

Table 248. Scale common properties (continued)

Property	JavaPOS	OPOS
PowerState		All
PhysicalDeviceDescription		All
PhysicalDeviceName		All
ResultCode	Not supported	All
ResultCodeExtended	Not supported	All
State		All

Table 249. Scale specific properties

Property	JavaPOS	OPOS
AsyncMode		All
CapDisplay		All
CapDisplayText		Not supported
CapFreezeValue		False
CapPriceCalculating		Not supported
CapReadLiveWeightWithTare		False
CapSetPriceCalculationMode		False
CapSetUnitPriceWithWeighUnit		False
CapSpecialTare		False
CapStatusUpdate		False
CapTarePriority		False
CapTareWeight		Not supported
CapZeroScale	4	All except 1
MaxDisplayTextChars		Not supported - 0
MaximumWeight		All
MinimumWeight		Not supported
SalesPrice		Not supported
ScaleLiveWeight		Not supported
StatusNotify		Not supported
TareWeight		Not supported
UnitPrice		Not supported
WeightUnit		All

Table 250. Scale common methods

Method	JavaPOS	OPOS
checkHealth		All
claim		All
clearInput		All
clearOutput		All
close		All
compareFirmwareVersion		Not supported

Table 250. Scale common methods (continued)

Method	JavaPOS	OPOS
directIO		Not supported
open		All
release		All
resetStatistics		Not supported
retrieveStatistics	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	Not supported
updateFirmware		Not supported
updateStatistics		Not supported

Table 251. Scale specific methods

Method	JavaPOS	OPOS
displayText		Not supported
doPriceCalculating		Not supported
freezeValue		Not supported
readLiveWeightWithTare		Not supported
readWeight		All
setPriceCalculationMode		Not supported
setSpecialTare		Not supported
setTarePriority		Not supported
setUnitPriceWithWeightUnit		Not supported
zeroScale	4	All except 1

Table 252. Scale events

Event	JavaPOS	OPOS
DataEvent		All
DirectIOEvent	Not supported	Not supported
ErrorEvent		All
StatusUpdateEvent		All

JavaPOS configuration

Configuration options depend on the hardware capabilities. Table 253 details the configuration options supported by the scale models.

Table 253. Scale configuration options

Configuration option	Toshiba 4687	Toshiba 4698	USB OEM
centerOfZero		✓	✓
displayRequired		✓	✓
enforceZeroReturn		✓	✓
fiveDigitWeight			✓
operationMode		✓	✓

Table 253. Scale configuration options (continued)

Configuration option	Toshiba 4687	Toshiba 4698	USB OEM
vibrationSensitivity		✓	✓
weighMode		✓	✓

CenterOfZero

```
<prop name="com.ibm.jpos.sdi.config.Scale.centerOfZero" type="Boolean" value="false"/>
```

Indicate center-of-zero with an LED.

Valid values are TRUE and FALSE. Default is FALSE.

DisplayRequired

```
<prop name="com.ibm.jpos.sdi.config.Scale.displayRequired" type="Boolean" value="false"/>
```

This property specifies whether a remote scale display is required. It is your responsibility to ensure that a system operating without a remote display meets the applicable weights and measures regulations.

If the value of this resource is false, but there is a remote display attached to the scale device, the display might not be used.

If the value of this resource is true but there is no remote display attached to the scale device, the scale might be unusable.

Valid values are TRUE and FALSE. Default is FALSE.

EnforceZeroReturn

```
<prop name="com.ibm.jpos.sdi.config.Scale.enforceZeroReturn" type="Boolean" value="false"/>
```

This property specifies whether zero protection is required. With zero protection enabled, the scale does not answer weight requests if a negative weight value is indicated on the display prior to placing the item for weighing on the scale, or an item is left on the scale for four minutes.

If either of these conditions exists and the value of this property is true, no weight data can be transmitted to the host until the scale returns to a zero weight (either by resetting the scale or removing the item).

Valid values are TRUE and FALSE. Default is FALSE

FiveDigitWeight (USB OEM only)

```
<prop name="com.ibm.jpos.sdi.config.Scale.fiveDigitWeight" type="Boolean" value="true"/>
```

This property specifies whether to return 5 digits (units of thousandths of pounds) or 4 digits (units of hundredths of pounds) for a non-metric weight.

Valid values are TRUE and FALSE. Default is TRUE.

OperationMode

```
<prop name="com.ibm.jpos.sdi.config.Scale.operationMode" type="String" value="US"/>
```

This property specifies the regulations that the point-of-sale system must conform to. The weight and measures requirements in various countries differ, and require minor operational differences.

Valid values are US and UK. Default is US.

- US** The scale conforms to regulations specified by the United States NIST Handbook 44 and the Canadian Department of Consumer and Corporate Affairs, Weights and Measurements Act, Specifications SGM-1.
- UK** The scale conforms to regulations specified by the Non-automatic Weighing Instruments (EEC Requirements) Regulations 1992 (based on OIML R 76-1).

VibrationSensitivity

```
<prop name="com.ibm.jpos.sdi.config.Scale.vibrationSensitivity" type="String" value="0"/>
```

This property controls the scale's sensitivity to vibration. External vibrations can affect the stability of the scale. There is a programmable vibration filter that allows you to reduce the scale's sensitivity to vibration. Reducing the scale's sensitivity to vibration increases the scale settling time slightly, so the higher vibration filter settings should be selected only when testing reveals a stability problem in the checkstand.

Valid values are 0 (normal sensitivity) to 3 (Ultra low sensitivity). Default is 0.

Default Value: 0

WeighMode

```
<prop name="com.ibm.jpos.sdi.config.Scale.weighMode" type="String" value="0"/>
```

This property determines whether the weight is returned in pounds (Avoirdupois or English system) or in kilograms (Metric system).

Valid values are 0 and 1. Default is 0.

- 0** Weight is given in pounds. For USB OEM compliant scales the fiveDigitWeight property specifies the number of digits to return for an English weight (pounds). For 4698/4696 scales, four digits represent the weight of the item in hundredths.
- 1** Weight is given in kilograms. All scales return five digits which represent the weight of the item in thousandths of kilograms.

Additional JavaPOS information

Known scale issues

This section contains information on known issues with specific models.

PSC 384 Magellan - USB

This model does not support the fiveDigitWeight configuration property. This property must be set to *false*; otherwise the Scale configuration fails. For more information, see "FiveDigitWeight (USB OEM only)" on page 294.

OPOS configuration

Table 254. Service Object settings for scale

Keyword	Type	Description
weightMode	String	Unit of weight used by the scale. Valid values are: 0 U.S. pound (default) 1 Kilogram
RemoteDisplayAttached	String	Whether a Pole Display is attached. Valid values are: True Display attached (other value) No display (default)
ZeroScale	String	Whether zeroing the scale is supported. Valid values are: 0 Not supported (default) 1 Supported
NCRCompatible	String	Determines how zero weight is handled. Valid values are: False Working according to UnifiedPOS specification True When stable weight of zero, return OPOS_SUCCESS

Chapter 19. Tone indicator

Supported devices

Table 255. Tone indicator supported devices

Device	Connectivity
1. Retail alphanumeric POS keyboard w/Card Reader	PS/2, RS-485, USB
2. Retail alphanumeric POS Keyboard with MSR and pointing device	PS/2
3. Retail POS keyboard	RS-485, USB
4. Retail POS keyboard with Card Reader	RS-485, USB
5. Retail POS keyboard w/Card Reader and Display	RS-485, USB
6. Modifiable Layout Keyboard w/Card Reader	RS-485, USB
7. SurePoint 4820 integrated Tone Indicator	RS-485, USB, PS/2
8. SureOne Integrated Keyboard	PS/2
9. POS keyboard V	RS-485, USB
10. PC POS Keyboard or PC Point of Sale Keyboard	PS/2, RS-485, USB
11. PLU keyboard/Display III	RS-485, USB
12. Retail POS keyboard VI	RS-485, USB
13. SurePOS 100 Integrated keyboard	PS/2
14. 4674 Integrated Keyboard	RS-485
15. 4685-KC1 (OPOS only)	RS-485
16. 4685-K01 (OPOS only)	RS-485
17. 4685-K02 (Ultra7) keyboard with card reader	RS-485, USB
18. 4685-K02 with MSR/Encoder and 4 position Keylock	USB
19. 4685-K02 with MSR/Encoder and 6 position Keylock	USB
20. 4685-K03	RS-485, USB
21. 4610 Tx6/Tx7/1NR Printer built-in tone indicator	EIA-232, RS-485, USB, Network
22. Internal speaker (OPOS only)	Integrated
23. Modular Alphanumeric Keyboard	PS/2, USB
24. Modular Compact Alphanumeric Keyboard	PS/2, USB
25. Modular 67 Key Keyboard	PS/2, USB
26. Modular 67 Key Keyboard with Display USB	PS/2, USB
27. 6141 Tone Indicator	USB

Supported properties and methods

Table 256. Tone indicator common properties

Property	JavaPOS	OPOS
AutoDisable		Not supported
BinaryConversion	Not supported	All

Table 256. Tone indicator common properties (continued)

Property	JavaPOS	OPOS
CapCompareFirmwareVersion		False
CapPowerReporting	PR_STANDARD	All support PR_STANDARD
CapStatisticsReporting	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CapUpdateFirmware		False
CapUpdateStatistics	False (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	False
CheckHealthText		All
Claimed		All
ClearInput		Not supported
DataCount		Not supported
DataEventEnabled	All	Not supported
DeviceControlDescription		All
DeviceControlVersion		All
DeviceEnabled		All
DeviceServiceDescription		All
DeviceServiceVersion		All
FreezeEvents		All
OpenResult	Not supported	All
OutputID		All
PowerNotify		All
PowerState		All
PhysicalDeviceDescription		All
PhysicalDeviceName		All
resultCode	Not supported	All
resultCodeExtended	Not supported	All
State		All

Table 257. Tone indicator specific properties

Property	JavaPOS	OPOS
AsyncMode		All
CapPitch		All
CapMelody		Not Supported
CapVolume		All
InterToneWait		All
MelodyType		Not Supported
MelodyVolume		Not Supported
Tone1Duration		All
Tone1Pitch		All

Table 257. Tone indicator specific properties (continued)

Property	JavaPOS	OPOS
Tone1Volume		All
Tone2Duration		All
Tone2Pitch		All
Tone2Volume		All

Table 258. Tone indicator common methods

Method	JavaPOS	OPOS
checkHealth		All
claim		All
clearOutput		All
close		All
compareFirmwareVersion		Not supported
directIO		Not supported
open		All
release		All
resetStatistics		Not supported
retrieveStatistics	True (see Appendix A, "JavaPOS support for UnifiedPOS device statistics properties," on page 431)	Not supported
updateFirmware		Not supported
updateStatistics		Not supported

Table 259. Tone indicator specific methods

Method	JavaPOS	OPOS
sound		All
soundImmediate		All

Table 260. Tone indicator events

Event	JavaPOS	OPOS
DirectIOEvent		Not supported
ErrorEvent		All
OutputCompleteEvent		All
StatusUpdateEvent		All

OPOS configuration

Table 261. Service Object settings for tone indicator

Keyword	Type	Description
OnlineTimeout	String	Time (in milliseconds) to wait for device to come online. Note: This setting cannot be modified with the Configuration tool.

Additional JavaPOS configuration

Enable or disable ELO tone interface

SurePoint 4820 devices with ELO touch screen display have a Tone Indicator interface, but by default it is disabled. A command line utility is provided to enable or disable the Tone Indicator. The syntax for the utility is:

```
java com.tgcs.posj.util.SurePointEloUtility [param]
```

Where:

- Enable: enables the Tone Interface
- Disable: disable the Tone Interface
- No param: a message that shows the status of the Tone Interface

Devices:

- USB SurePoint 4820-21x/51x with ELO Touch Screen
- USB SurePoint 4820-2Lx/5Lx with ELO Touch Screen

OS supported:

- Linux

JavaPOS ELO tone interface configuration

For Linux environments, the posj.properties file must be updated to include the Vendor ID and Product ID for the SurePoint 4820 display:

1. Edit the posj.properties file, located at:
 - Linux: /opt/ibm/javapos/etc
2. Add the following entry to the end of the OEM Scanner section:
 - com.ibm.posj.bus.hid.javaxusb.factory.knownPosDevices.95 = 0x04e7,0x0030

Part 3. Keyboards and code pages

Chapter 20. POS keyboard layouts and scan codes

This chapter contains keyboard layouts and scan codes for Toshiba Point-of-Sale keyboards.

Note: The numeric keypad is shaded in all illustrations.

Understanding scan codes

When a key is pressed on any keyboard (sometimes referred to as, a KEY_DOWN event), the keyboard device driver receives a code that is called a *make scan code*. There is a different code for each key. When a key is released (on some keyboards) sometimes referred to as a KEY_UP event, the keyboard device driver receives a code that is called a *break scan code*. These codes are translated into ASCII character codes by the operating system.

The following scancode tables do not contain values for the OPOS driver, because the OPOS driver only sends a data event to the application if each individual key has been mapped in the configuration program. If a key has been mapped by the user the value that will be sent to the application will be the user-defined mapped value, not the hardware scancode.

For keys that generate different scan codes based on a modifier key (IE, S1 and S2) the scan code shown in the table below does not include the make scan code or break scan code for the modifier key, only the scan code for that key.

Checkout Keyboards Layout (50-key and Modular 67 Key)

This section contains illustrations of the layouts for the following keyboards:

- Retail Point of Sale Keyboard (50-key)
- Retail Point of Sale Keyboard with card reader (50-key)
- Retail Point of Sale Keyboard with card reader and display (50-key)
- Point of Sale Keyboard VI
- Modular 67 Key Keyboard

50-Key Modifiable Layout Keyboard

Figure 40 shows the key-switch numbers. The numeric keypad (key switches 18 to 29) is shown in the shaded area of the illustration.

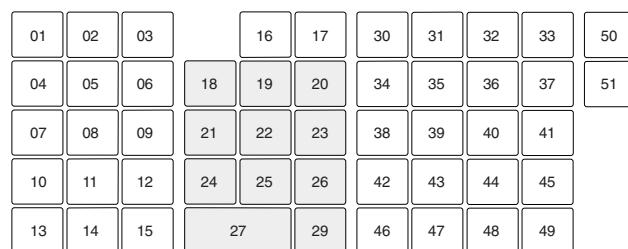


Figure 40. Layout for 50-Key Modifiable Keyboard

50-Key Modifiable Layout Keyboard and Operator Display

Figure 41 on page 304 shows the key-switch numbers. The numeric keypad (key switches 18 to 29) is shown in the shaded area of the illustration.

01	02	03		16	17	30	31	32	33	
04	05	06		18	19	20	34	35	36	37
07	08	09		21	22	23	38	39	40	41
10	11	12		24	25	26	42	43	44	45
13	14	15		27	28	29	46	47	48	49
									50	
										51

Figure 41. Layout for 50-Key Modifiable Keyboard and Operator Display

Retail Point of Sale Keyboard Layout

Figure 42 shows the key-switch numbers for the Retail Point of Sale Keyboard, both with and without the card reader. The numeric keypad (key switches 18 to 29) are shown in the shaded area of the illustration.

50	01	02	03		16	17	30	31	32	33	
	04	05	06		18	19	20	34	35	36	37
	07	08	09		21	22	23	38	39	40	41
	10	11	12		24	25	26	42	43	44	45
	13	14	15		27	28	29	46	47	48	49

Figure 42. Layout for Retail Point of Sale Keyboard

Retail Point of Sale Keyboard with Card Reader and Display

Figure 43 shows the key-switch numbers for the Retail Point of Sale Keyboard with Card Reader and Display. The numeric keypad (key switches 18 to 29) is shown in the shaded area of the illustration.

01	02	03		16	17	30	31	32	33	
04	05	06		18	19	20	34	35	36	37
07	08	09		21	22	23	38	39	40	41
10	11	12		24	25	26	42	43	44	45
50	13	14	15	27	28	29	46	47	48	49

Figure 43. Layout for Retail Point of Sale Keyboard with Card Reader and Display

Point of Sale Keyboard VI Layout

Figure 44 on page 305 shows the key-switch numbers for the Point of Sale Keyboard VI. The numeric keypad (key switches 18 to 29) is shown in the shaded area of the illustration.

50	01	02	03	16	17	30	31	32	33	
04	05	06		18	19	20	34	35	36	37
07	08	09		21	22	23	38	39	40	41
10	11	12		24	25	26	42	43	44	45
13	14	15		27	28	29	46	47	48	49

Figure 44. Layout for Point of Sale Keyboard VI

Modular 67-Key Keyboard Layout

Figure 45 shows the key-switch numbers for the Modular 67-Key Keyboard. The numeric keypad (key switches 18 to 29) is shown in the shaded area of the illustration.

55	56	57	58	59	60	61	62	63	64	65	66	67	S1	S2
50	01	02	03		16	17	30	31	32	33				
51	04	05	06		18	19	20	34	35	36	37			
52	07	08	09		21	22	23	38	39	40	41			
53	10	11	12		24	25	26	42	43	44	45			
54	13	14	15		27		29	46	47	48	49			

Figure 45. Layout for Modular 67 Key Keyboard

Modular 67-Key Keyboard with Display Layout

Figure 46 on page 306 shows the key-switch numbers for the Retail Point of Sale 67-Key Keyboard with Display. The numeric keypad (key switches 18 to 29) is shown in the shaded area of the illustration.

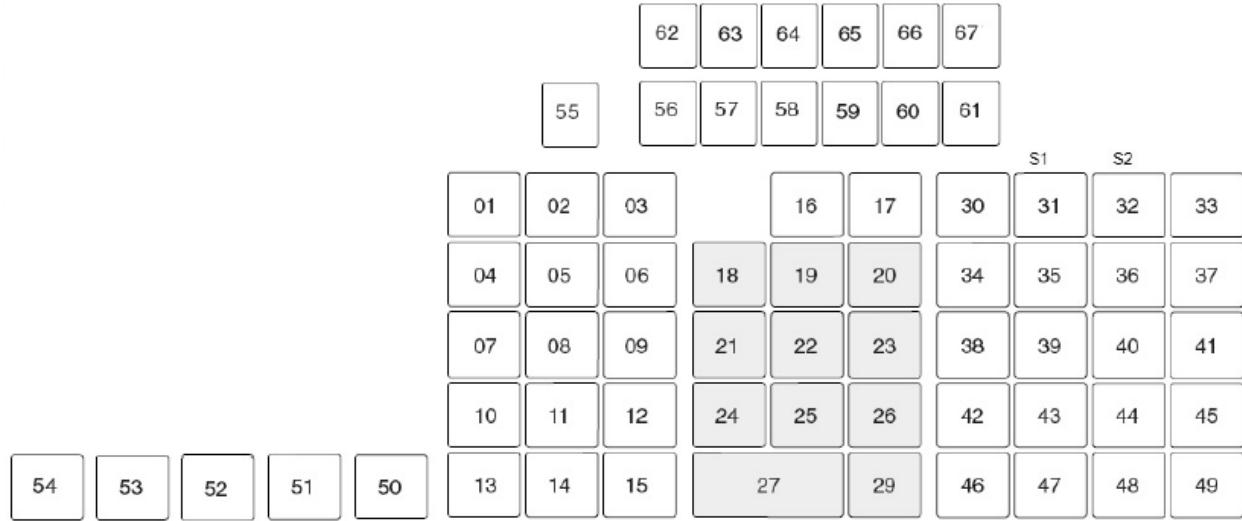


Figure 46. Layout for Retail Point of Sale 67-Key Keyboard with Display

Checkout Keyboards PS/2 or USB (System Keyboard) scan code set

In Linux, Modular 67-Key Keyboard scan codes may not match those provided in the table below until a configuration file is downloaded to the device. Follow the steps listed below to download the configuration file:

- Copy the aip46013.cfg.winCompatibility file from /opt/ibm/javapos/config to /usr/share/pos/config/aip46013.cfg (if the keyboard does not have an MSR).
- Copy the aip46063.cfg.winCompatibility file from /opt/ibm/javapos/config to /usr/share/pos/config/aip46063.cfg (if the keyboard has an MSR).
- Reboot the system.

Table 262 shows the key scan codes for the Modular 67 Key Keyboard.

Table 262. PS/2 or USB (System Keyboard) scan codes for Modular 67 Key Keyboard

Key switch number	Hardware make code	Hardware break code	JavaPOS – POSKeyData	Comments
1	X'3b'	X'bb'	X'3b'	
2	X'3c'	X'bc'	X'3c'	
3	X'3d'	X'bd'	X'3d'	
4	X'3e'	X'be'	X'3e'	
5	X'3f'	X'bf'	X'3f'	
6	X'40'	X'c0'	X'40'	
7	X'41'	X'c1'	X'41'	
8	X'42'	X'c2'	X'42'	
9	X'43'	X'c3'	X'43'	
10	X'44'	X'c4'	X'44'	
11	X'57'	X'd7'	X'57'	
12	X'58'	X'd8'	X'58'	

Table 262. PS/2 or USB (System Keyboard) scan codes for Modular 67 Key Keyboard (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS – POSKeyData	Comments
13	X'0f'	X'8f'	X'0f'	
14	X'e0' X'5b'	X'e0' X'db'	X'01' X'5b'	
15	X'38'	X'b8'	X'38'	
16	X'46'	X'c6'	X'46'	
17	X'45'	X'c5'	X'45'	
18	X'47'	X'c7'	X'47'	
19	X'48'	X'c8'	X'48'	
20	X'49'	X'c9'	X'49'	
21	X'4b'	X'cb'	X'4b'	
22	X'4c'	X'cc'	X'4c'	
23	X'4d'	X'cd'	X'4d'	
24	X'4f'	X'cf'	X'4f'	
25	X'50'	X'd0'	X'50'	
26	X'51'	X'd1'	X'51'	
27	X'64'	X'e4'	X'64'	
28	X'52'	X'd2'	X'52'	
29	X'53'	X'd2'	X'53'	
30	X'29'	X'a9'	X'29'	
31	X'e0' X'2a'X'e0'X'37'	X'e0' X'b7'X'e0'X'aa'	X'01' X'37'	On Windows, key release only
32	X'e1' X'1d' X'45' X'e1' X'9d'X'c5'		X'01' X'45'	Key press only
33	X'0e'	X'8e'	X'0e'	
34	X'0c'	X'8c'	X'0c'	
35	X'0d'	X'8d'	X'0d'	
36	X'28'	X'a8'	X'28'	
37	X'2b'	X'ab'	X'2b'	
38	X'33'	X'b3'	X'33'	
39	X'34'	X'b4'	X'34'	
40	X'35'	X'b35'	X'35'	
41	X'27'	X'a7'	X'27'	
42	X'65'	X'e5'	X'65'	
43	X'66'	X'e6'	X'66'	
44	X'67'	X'e7'	X'67'	
45	X'68'	X'e8'	X'68'	
46	X'1d'	X'9d'	X'1d'	
47	X'39'	X'b9'	X'39'	
48	X'2a'	X'aa'	X'2a'	
49	X'e0' X'1c'	X'e0' X'9c'	X'01' X'1c'	
50	X'01'	X'81'	X'01'	

Table 262. PS/2 or USB (System Keyboard) scan codes for Modular 67 Key Keyboard (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
51	X'1e'	X'9e'	X'1e'	
52	X'30'	X'b0'	X'30'	
53	X'2e'	X'ae'	X'2e'	
54	X'20'	X'a0'	X'20'	
55	X'12'	X'92'	X'12'	
56	X'21'	X'a1'	X'21'	
57	X'22'	X'a2'	X'22'	
58	X'23'	X'a3'	X'23'	
59	X'17'	X'97'	X'17'	
60	X'24'	X'a4'	X'24'	
61	X'25'	X'a5'	X'25'	
62	X'26'	X'a6'	X'26'	
63	X'32'	X'b2'	X'32'	
64	X'31'	X'b1'	X'31'	
65	X'18'	X'98'	X'18'	
66	X'19'	X'99'	X'19'	
67	X'10'	X'90'	X'10'	

Checkout keyboards RS-485/USB scan code set

Table 263 shows the key scan codes

- 50-Key Modifiable Layout Keyboard
- 50-Key Modifiable Layout Keyboard and Operator Display
- Retail Point of Sale Keyboard (50-Key)
- Retail Point of Sale Keyboard with card reader (50-key)
- Retail Point of Sale Keyboard with card reader and display (50-key)
- Point of Sale Keyboard VI
- Modular 67 Key Keyboard

The hardware scan code set for the 50-Key Modifiable Layout Keyboard, and the 50-Key Modifiable Layout Keyboard and Operator Display is different from the hardware scan code set for the retail point-of-sale keyboards. In order to allow the application to work with either keyboard more easily, the hardware scan code set for the 50-Key Modifiable Layout Keyboard and the 50-Key Modifiable Layout Keyboard and Operator Display is translated to the scan code set for the retail point-of-sale keyboards.

Note: The 50-Key Modifiable Layout Keyboard and 50-Key Modifiable Layout Keyboard and Operator Display only generate make scan codes when a key is pressed. In order for your application to work with either the old or the new keyboards, it must use the make scan codes from the checkout style keyboards and discard any break scan codes it receives.

Table 263. Checkout keyboards RS-485/USB scan code set

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
1	X'4b'	X'f0' X'4b'	X'4b'	

Table 263. Checkout keyboards RS-485/USB scan code set (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
2	X'3b'	X'f0' X'3b'	X'3b'	
3	X'6b'	X'f0' X'6b'	X'6b'	
4	X'4c'	X'f0' X'4c'	X'4c'	
5	X'3c'	X'f0' X'3c'	X'3c'	
6	X'6c'	X'f0' X'6c'	X'6c'	
7	X'4f'	X'f0' X'4f'	X'4f'	
8	X'3f'	X'f0' X'3f'	X'3f'	
9	X'6f'	X'f0' X'6f'	X'6f'	
10	X'4e'	X'f0' X'4e'	X'4e'	
11	X'3e'	X'f0' X'3e'	X'3e'	
12	X'6e'	X'f0' X'6e'	X'6e'	
13	X'4d'	X'f0' X'4d'	X'4d'	
14	X'3d'	X'f0' X'3d'	X'3d'	
15	X'6d'	X'f0' X'6d'	X'6d'	
16	X'7b'	X'f0' X'7b'	X'7b'	
17	X'1b'	X'f0' X'1b'	X'1b'	
18	X'7c '	X'f0' X'7c'	X'7c'	
19	X'0c'	X'f0' X'0c'	X'0c'	
20	X'1c'	X'f0' X'1c'	X'1c'	
21	X'7f'	X'f0' X'7f'	X'7f'	
22	X'0f'	X'f0' X'0f'	X'0f'	
23	X'1f'	X'f0' X'1f'	X'1f'	
24	X'7e'	X'f0' X'7e'	X'7e'	
25	X'0e'	X'f0' X'0e'	X'0e'	
26	X'1e'	X'f0' X'1e'	X'1e'	
27	X'7d'	X'f0' X'7d'	X'7d'	
28	X'0d'	X'f0' X'0d'	X'0d'	
29	X'1d'	X'f0' X'1d'	X'1d'	
30	X'8b'	X'f0' X'8b'	X'8b'	
31	X'af'	X'f0' X'af'	X'af'	
32	X'bf'	X'f0' X'bf'	X'bf'	
33	X'bb'	X'f0' X'bb'	X'bb'	
34	X'8c'	X'f0' X'8c'	X'8c'	
35	X'9c'	X'f0' X'9c'	X'9c'	
36	X'ac'	X'f0' X'ac'	X'ac'	
37	X'bc'	X'f0' X'bc'	X'bc'	
38	X'8f'	X'f0' X'8f'	X'8f'	
39	X'9f'	X'f0' X'9f'	X'9f'	
40	X'9b'	X'f0' X'9b'	X'9b'	

Table 263. Checkout keyboards RS-485/USB scan code set (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
41	X'ab'	X'f0' X'ab'	X'ab'	
42	X'8e'	X'f0' X'8e'	X'8e'	
43	X'9e'	X'f0' X'9e'	X'9e'	
44	X'ae'	X'f0' X'ae'	X'ae'	
45	X'be'	X'f0' X'be'	X'be'	
46	X'8d'	X'f0' X'8d'	X'8d'	
47	X'9d'	X'f0' X'9d'	X'9d'	
48	X'ad'	X'f0' X'ad'	X'ad'	
49	X'bd'	X'f0' X'bd'	X'bd'	
50	X'50'	X'f0' X'50'	X'50'	
51	X'5A'	X'F0'X'5A'	X'5A'	67-Key
52	X'2B'	X'F0'X'2B'	X'2B'	67-Key
53	X'61'	X'F0'X'61'	X'61'	67-Key
54	X'62'	X'F0'X'62'	X'62'	67-Key
55	X'21'	X'F0'X'21'	X'21'	67-Key
56	X'52'	X'F0'X'52'	X'52'	67-Key
57	X'53'	X'F0'X'53'	X'53'	67-Key
58	X'32'	X'F0'X'32'	X'32'	67-Key
59	X'23'	X'F0'X'23'	X'23'	67-Key
60	X'2A'	X'F0'X'2A'	X'2A'	67-Key
61	X'28'	X'F0'X'28'	X'28'	67-Key
62	X'08'	X'F0'X'08'	X'08'	67-Key
63	X'56'	X'F0'X'56'	X'56'	67-Key
64	X'07'	X'F0'X'07'	X'07'	67-Key
65	X'1A'	X'F0'X'1A'	X'1A'	67-Key
66	X'63'	X'F0'X'63'	X'63'	67-Key
67	X'64'	X'F0'X'64'	X'64'	67-Key
	+ Ctrl Case Pressed with 2 position Keylock			
31	X'f0' X'50' X'00' X'50'		X'00'	Key press only
32	X'f0' X'50' X'01' X'50'		X'01'	Key press only
	+ 4 Keylock position set to position #3			
31	X'f0' X'50' X'00' X'50'		X'00'	Key press only
32	X'f0' X'50' X'01' X'50'		X'01'	Key press only

Note: 67-Key

These keys are present only on the Modular 67 Key Keyboard.

Table 264 explains the scan codes received for the S1 and S2 function keys on the retail point-of-sale keyboards.

The S1 and S2 function keys send a series of scan codes on the retail point-of-sale keyboards. These function keys generate a break scan code for the Ctrl key (scan code of 0x50 with PosKC_KEYUP flag set), a make scan code for the S1 or S2 key and then a make scan code for the Ctrl key (scan code of 0x50 with PosKC_KEYUP flag reset).

The S1 and S2 function keys can be accessed in 2 different forms:

1. By pressing the Ctrl key. With the Ctrl key pressed, key 31 represents the S1 function and key 32 represents the S2 function. This applies for the 50 key POSKeyboards and the 67 key POSKeyboards with 2 position Keylock.
2. Moving the keylock to positions 3. This applies to 67 key POSKeyboards with 4 position Keylock.

No break scan codes are sent for the 0x00 or 0x01 scan codes.

Table 264. RS-485/USB scan codes for the Retail Point of Sale Keyboards

Key switch number	Key type	Ctrl + scan code
31	S1	X'00'
32	S2	X'01'

Modifiable-layout keyboard with card reader layout (133-key)

Figure 47 shows the key-switch numbers for the modifiable layout keyboard with card reader. The three possible locations for the numeric keypad are shown in the shaded area of the illustration. The default location for the numeric keypad is the right-most shaded area.

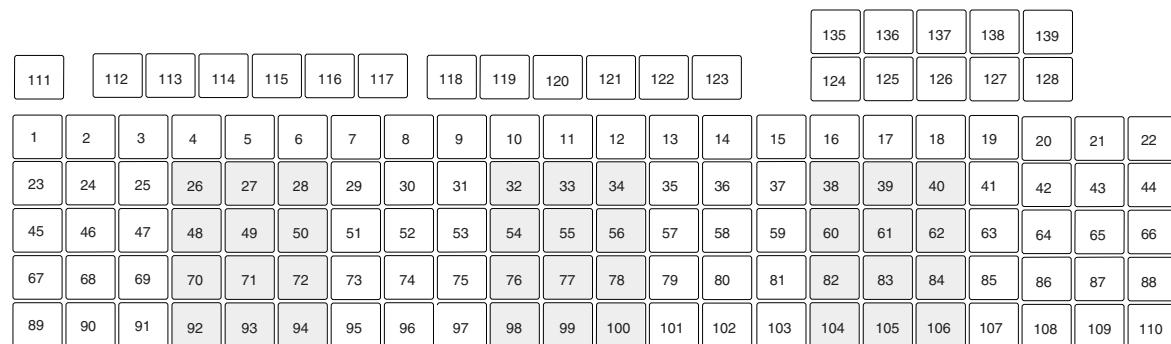


Figure 47. Modifiable layout keyboard with card reader

Modifiable layout keyboard with card reader RS-485/USB scan code set

Table 265 shows the key scan codes for the modifiable layout keyboard with card reader.

Table 265. Modifiable layout keyboard RS-485/USB scan code set

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
1	X'b4'	X'f0' X'b4'	X'b4'	
2	X'a4'	X'f0' X'a4'	X'a4'	
3	X'34'	X'f0' X'34'	X'34'	
4	X'44'	X'f0' X'44'	X'44'	
5	X'94'	X'f0' X'94'	X'94'	
6	X'84'	X'f0' X'84'	X'84'	
7	X'14'	X'f0' X'14'	X'14'	
8	X'04'	X'f0' X'04'	X'04'	
9	X'74'	X'f0' X'74'	X'74'	
10	X'64'	X'f0' X'64'	X'64'	
11	X'54'	X'f0' X'54'	X'54'	
12	X'5b'	X'f0' X'5b'	X'5b'	
13	X'6b'	X'f0' X'6b'	X'6b'	
14	X'7b'	X'f0' X'7b'	X'7b'	
15	X'0b'	X'f0' X'0b'	X'0b'	
16	X'1b'	X'f0' X'1b'	X'1b'	
17	X'8b'	X'f0' X'8b'	X'8b'	
18	X'9b'	X'f0' X'9b'	X'9b'	
19	X'3b'	X'f0' X'3b'	X'3b'	
20	X'4b'	X'f0' X'4b'	X'4b'	
21	X'ab'	X'f0' X'ab'	X'ab'	
22	X'bb'	X'f0' X'bb'	X'bb'	
23	X'b5'	X'f0' X'b5'	X'b5'	
24	X'a5'	X'f0' X'a5'	X'a5'	
25	X'35'	X'f0' X'35'	X'35'	
26	X'45'	X'f0' X'45'	X'45'	
27	X'95'	X'f0' X'95'	X'95'	
28	X'85'	X'f0' X'85'	X'85'	
29	X'15'	X'f0' X'15'	X'15'	
30	X'05'	X'f0' X'05'	X'05'	
31	X'75'	X'f0' X'75'	X'75'	
32	X'65'	X'f0' X'65'	X'65'	
33	X'55'	X'f0' X'55'	X'55'	
34	X'5a'	X'f0' X'5a'	X'5a'	
35	X'6a'	X'f0' X'6a'	X'6a'	
36	X'7a'	X'f0' X'7a'	X'7a'	

Table 265. Modifiable layout keyboard RS-485/USB scan code set (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
37	X'0a'	X'f0' X'0a'	X'0a'	
38	X'1a'	X'f0' X'1a'	X'1a'	
39	X'8a'	X'f0' X'8a'	X'8a'	
40	X'9a'	X'f0' X'9a'	X'9a'	
41	X'3a'	X'f0' X'3a'	X'3a'	
42	X'4a'	X'f0' X'4a'	X'4a'	
43	X'aa'	X'f0' X'aa'	X'aa'	
44	X'ba'	X'f0' X'ba'	X'ba'	
45	X'b6'	X'f0' X'b6'	X'b6'	
46	X'a6'	X'f0' X'a6'	X'a6'	
47	X'36'	X'f0' X'36'	X'36'	
48	X'46'	X'f0' X'46'	X'46'	
49	X'96'	X'f0' X'96'	X'96'	
50	X'86'	X'f0' X'86'	X'86'	
51	X'16'	X'f0' X'16'	X'16'	
52	X'06'	X'f0' X'06'	X'06'	
53	X'76'	X'f0' X'76'	X'76'	
54	X'66'	X'f0' X'66'	X'66'	
55	X'56'	X'f0' X'56'	X'56'	
56	X'59'	X'f0' X'59'	X'59'	
57	X'69'	X'f0' X'69'	X'69'	
58	X'79'	X'f0' X'79'	X'79'	
59	X'09'	X'f0' X'09'	X'09'	
60	X'19'	X'f0' X'19'	X'19'	
61	X'89'	X'f0' X'89'	X'89'	
62	X'99'	X'f0' X'99'	X'99'	
63	X'39'	X'f0' X'39'	X'39'	
64	X'49'	X'f0' X'49'	X'49'	
65	X'a9'	X'f0' X'a9'	X'a9'	
66	X'b9'	X'f0' X'b9'	X'b9'	
67	X'b3'	X'f0' X'b3'	X'b3'	
68	X'a3'	X'f0' X'a3'	X'a3'	
69	X'33'	X'f0' X'33'	X'33'	
70	X'43'	X'f0' X'43'	X'43'	
71	X'93'	X'f0' X'93'	X'93'	
72	X'83'	X'f0' X'83'	X'83'	
73	X'13'	X'f0' X'13'	X'13'	
74	X'03'	X'f0' X'03'	X'03'	
75	X'73'	X'f0' X'73'	X'73'	

Table 265. Modifiable layout keyboard RS-485/USB scan code set (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
76	X'63'	X'f0' X'63'	X'63'	
77	X'53'	X'f0' X'53'	X'53'	
78	X'5c'	X'f0' X'5c'	X'5c'	
79	X'6c'	X'f0' X'6c'	X'6c'	
80	X'7c'	X'f0' X'7c'	X'7c'	
81	X'0c'	X'f0' X'0c'	X'0c'	
82	X'1c'	X'f0' X'1c'	X'1c'	
83	X'8c'	X'f0' X'8c'	X'8c'	
84	X'9c'	X'f0' X'9c'	X'9c'	
85	X'3c'	X'f0' X'3c'	X'3c'	
86	X'4c'	X'f0' X'4c'	X'4c'	
87	X'ac'	X'f0' X'ac'	X'ac'	
88	X'bc'	X'f0' X'bc'	X'bc'	
89	X'b2'	X'f0' X'b2'	X'b2'	
90	X'a2'	X'f0' X'a2'	X'a2'	
91	X'32'	X'f0' X'32'	X'32'	
92	X'42'	X'f0' X'42'	X'42'	
93	X'92'	X'f0' X'92'	X'92'	
94	X'82'	X'f0' X'82'	X'82'	
95	X'12'	X'f0' X'12'	X'12'	
96	X'02'	X'f0' X'02'	X'02'	
97	X'72'	X'f0' X'72'	X'72'	
98	X'62'	X'f0' X'62'	X'62'	
99	X'52'	X'f0' X'52'	X'52'	
100	X'5d'	X'f0' X'5d'	X'5d'	
101	X'6d'	X'f0' X'6d'	X'6d'	
102	X'7d'	X'f0' X'7d'	X'7d'	
103	X'0d'	X'f0' X'0d'	X'0d'	
104	X'1d'	X'f0' X'1d'	X'1d'	
105	X'8d'	X'f0' X'8d'	X'8d'	
106	X'9d'	X'f0' X'9d'	X'9d'	
107	X'3d'	X'f0' X'3d'	X'3d'	
108	X'4d'	X'f0' X'4d'	X'4d'	
109	X'ad'	X'f0' X'ad'	X'ad'	
110	X'bd'	X'f0' X'bd'	X'bd'	
111	X'20'	X'f0' X'20'	X'20'	
112	X'b1'	X'f0' X'b1'	X'b1'	
113	X'a1'	X'f0' X'a1'	X'a1'	
114	X'31'	X'f0' X'31'	X'31'	

Table 265. Modifiable layout keyboard RS-485/USB scan code set (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
115	X'41'	X'f0' X'41'	X'41'	
116	X'91'	X'f0' X'91'	X'91'	
117	X'81'	X'f0' X'81'	X'81'	
118	X'88'	X'f0' X'88'	X'88'	
119	X'18'	X'f0' X'18'	X'18'	
120	X'58'	X'f0' X'58'	X'58'	
121	X'68'	X'f0' X'68'	X'68'	
122	X'78'	X'f0' X'78'	X'78'	
123	X'08'	X'f0' X'08'	X'08'	
124	X'1e'	X'f0' X'1e'	X'1e'	
125	X'8e'	X'f0' X'8e'	X'8e'	
126	X'9e'	X'f0' X'9e'	X'9e'	
127	X'ae'	X'f0' X'ae'	X'ae'	
128	X'be'	X'f0' X'be'	X'be'	
135	X'1f'	X'f0' X'1f'	X'1f'	
136	X'8f'	X'f0' X'8f'	X'8f'	
137	X'9f'	X'f0' X'9f'	X'9f'	
138	X'af'	X'f0' X'af'	X'af'	
139	X'bf'	X'f0' X'bf'	X'bf'	
+ Ctrl Case Pressed				
124 (S2)	X'f0' X'20' X'01' X'20'		X'01'	Key press only
135 (S1)	X'f0' X'20' X'00' X'20'		X'00'	Key press only

Table 266 explains the scan codes received for the S1 and S2 function keys on the Modifiable Layout Keyboard with Card Reader.

The S1 and S2 function keys send a series of scan codes on the Modifiable Layout Keyboard with Card Reader. These function keys generate a break scan code for the Ctrl key (scan code of 0x20 with PosKC_KEYUP flag set), a make scan code for the S1 or S2 key, and then a make scan code for the Ctrl key (scan code of 0x20 with the PosKC_KEYUP flag reset). The S1 and S2 function keys can only be accessed by pressing the Ctrl key. With the Ctrl key pressed, key 135 represents the S1 function, and key 124 represents the S2 function.

No break scan codes are sent for the 0x00 or 0x01 scan codes.

Table 266. RS-485 Scan Codes – Modifiable Layout Keyboard with Card Reader

Key switch number	Key type	Ctrl + scan code
124	S2	X'01'
135	S1	X'00'

Alphanumeric Point of Sale keyboards

This section contains illustrations of the layouts for the following keyboards:

- PC Point of Sale Keyboard with Card Reader (ANKPOS)
- Retail Alphanumeric POSKeyboard with Card Reader (NANPOS)
- Modular Alphanumeric POS Keyboard
- Modular Compact Alphanumeric POS Keyboard

PC Point of Sale keyboard with card reader (ANKPOS)

Figure 48 shows the layout and assigned key-switch numbers for the PC Point of Sale Keyboard. The numeric keypad location is shown in the shaded area of the illustration.

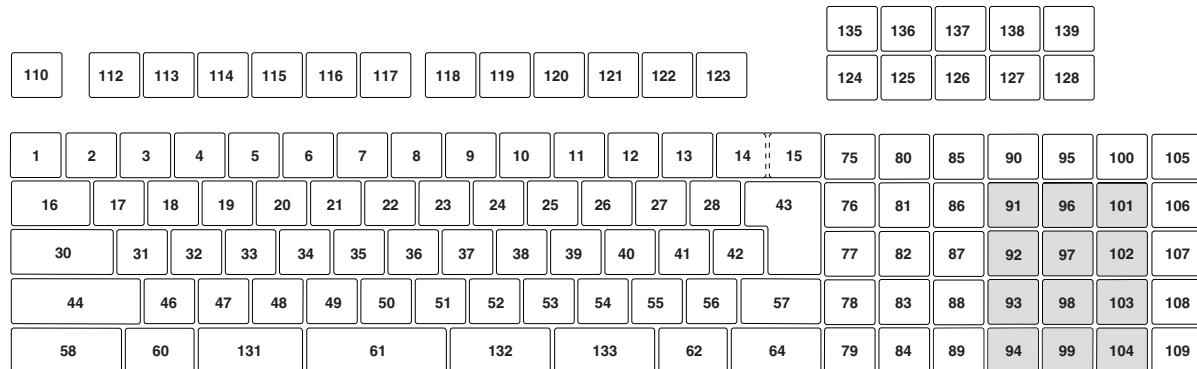


Figure 48. PC Point of Sale keyboard layout

Retail Alphanumeric Point of Sale Keyboard with card reader (NANPOS)

Figure 49 shows the layout and assigned key-switch numbers for the Retail Alphanumeric Point of Sale Keyboard with card reader.

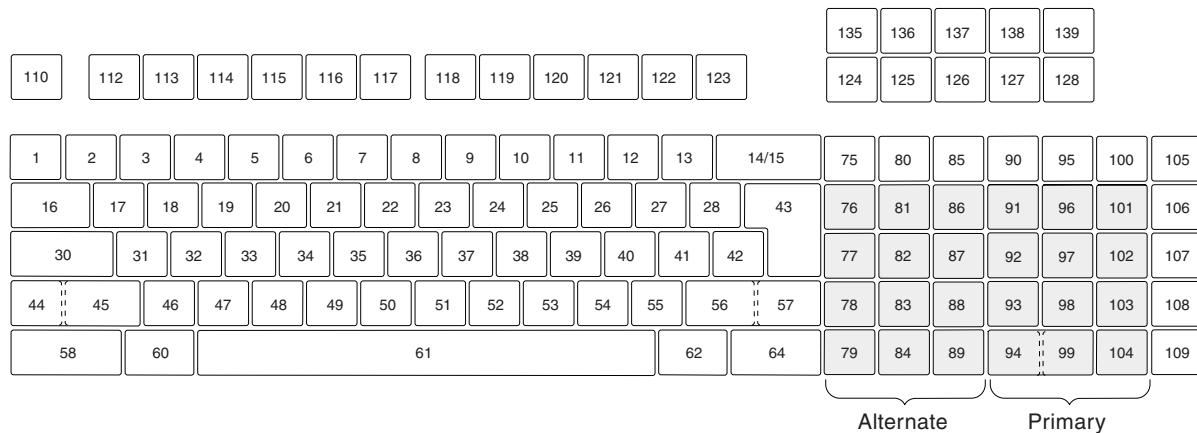


Figure 49. Layout and Assigned Switch Numbers

Note:

1. The two possible locations for the numeric keypad are shown in the shaded area of the illustration. The default location for the numeric keypad is the rightmost shaded area.

- Key 45 appears as a single key only on non-U.S. keyboards. On U.S. keyboards, key 44 also covers key 45 (key 44, 45 is a double key).
- Keys 94 and 99 can have a single, horizontal double-wide key covering both keys, or they can be split into two individual keys.
- This keyboard is similar to the 101-enhanced keyboard and the 102-enhanced keyboard. The following keys are on the Retail Alphanumeric Point of Sale Keyboard with Card Reader, but not on the 101-enhanced keyboard or the 102-enhanced keyboard. These keys are referred to as the *point-of-sale-unique keys*.

77	106
78	107
82	108
87	124
88	125
90	126
95	127
99	128
100	135
105	

See Table 267 on page 318 for the scan codes associated with these keys.

Modular Alphanumeric POS Keyboard

Figure 50 shows the layout and assigned key-switch numbers for the Modular Alphanumeric POS Keyboard.

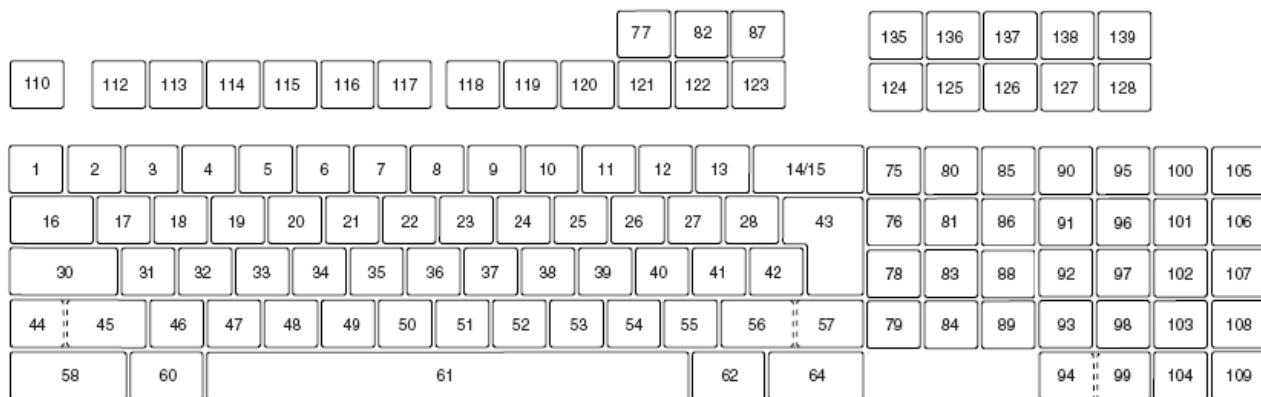


Figure 50. Layout and Assigned Switch Numbers for Modular Alphanumeric POS Keyboard

See Table 267 on page 318 for the scan codes associated with these keys.

Modular Compact Alphanumeric Point of Sale Keyboard

Figure 51 on page 318 shows the layout and assigned key-switch numbers for the Modular Compact Alphanumeric POS Keyboard.

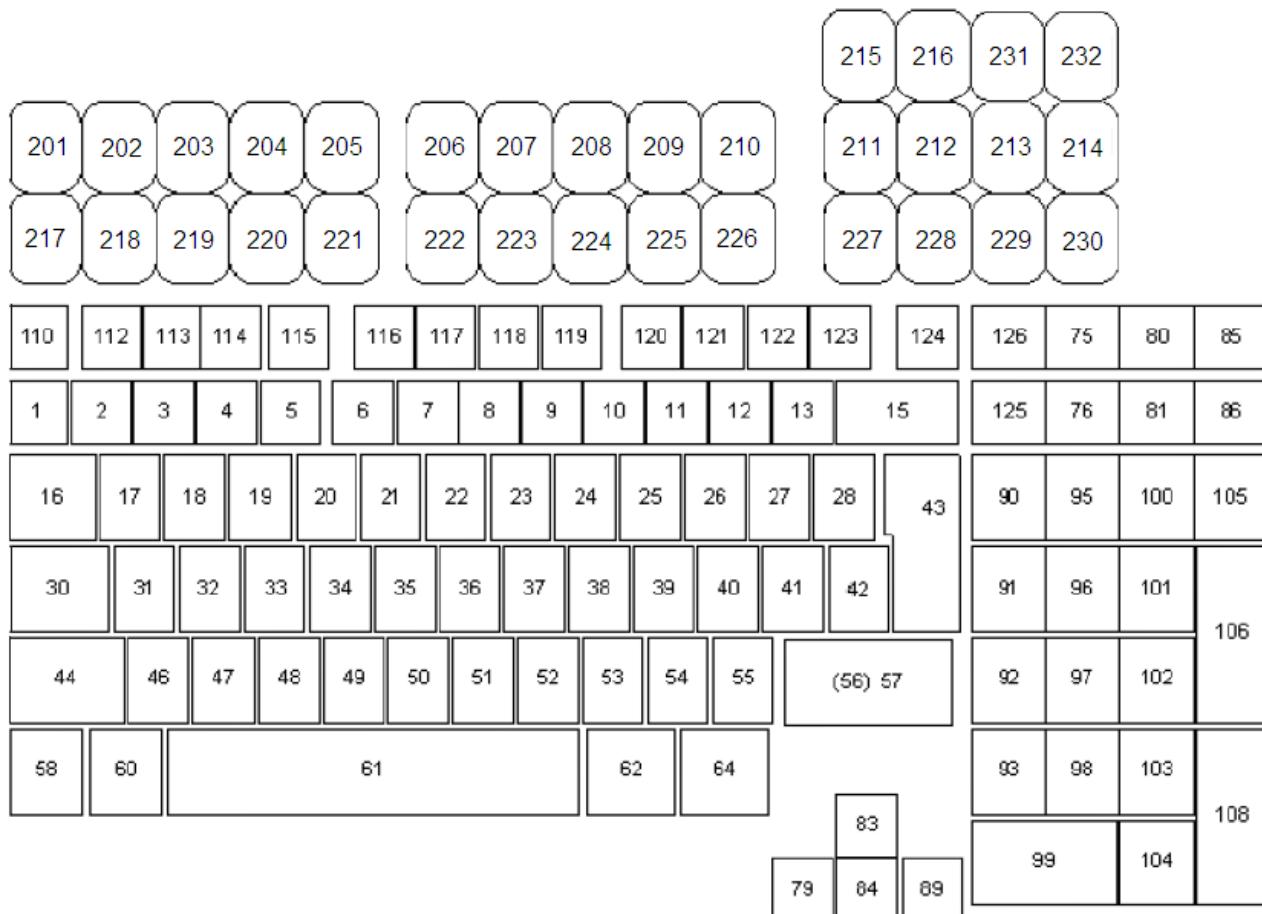


Figure 51. Layout and Assigned Switch Numbers for the Modular Compact Alphanumeric POS Keyboard

See Table 270 on page 323 for the scan codes associated with these keys.

RS-485/USB scan code set for Retail Alphanumeric Point of Sale Keyboard (NANPOS), PC Point of Sale Keyboard (ANKPOS), USB scan code set for Modular Alphanumeric Point of Sale Keyboard

Table 267 relates the keyboard key-switch number to the scan codes received when the keyboard is attached to the RS-485 or USB port.

Table 267. RS-485/USB scan code set

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
1	X'51'	X'f0' X'51'	X'51'	
2	X'11'	X'f0' X'11'	X'11'	
3	X'12'	X'f0' X'12'	X'12'	
4	X'13'	X'f0' X'13'	X'13'	
5	X'14'	X'f0' X'14'	X'14'	
6	X'54'	X'f0' X'54'	X'54'	
7	X'55'	X'f0' X'55'	X'55'	
8	X'15'	X'f0' X'15'	X'15'	

Table 267. RS-485/USB scan code set (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
9	X'18'	X'f0' X'18'	X'18'	
10	X'16'	X'f0' X'16'	X'16'	
11	X'17'	X'f0' X'17'	X'17'	
12	X'57'	X'f0' X'57'	X'57'	
13	X'58'	X'f0' X'58'	X'58'	
14	X'6A'	X'f0' X'6A'	X'6A'	ANKPOS (see Notes)
15	X'7A'	X'f0' X'7A'	X'7A'	
16	X'71'	X'f0' X'71'	X'71'	
17	X'61'	X'f0' X'61'	X'61'	
18	X'62'	X'f0' X'62'	X'62'	
19	X'63'	X'f0' X'63'	X'63'	
20	X'64'	X'f0' X'64'	X'64'	
21	X'74'	X'f0' X'74'	X'74'	
22	X'75'	X'f0' X'75'	X'75'	
23	X'65'	X'f0' X'65'	X'65'	
24	X'68'	X'f0' X'68'	X'68'	
25	X'66'	X'f0' X'66'	X'66'	
26	X'67'	X'f0' X'67'	X'67'	
27	X'77'	X'f0' X'77'	X'77'	
28	X'78'	X'f0' X'78'	X'78'	
30	X'72'	X'f0' X'72'	X'72'	
31	X'81'	X'f0' X'81'	X'81'	
32	X'82'	X'f0' X'82'	X'82'	
33	X'83'	X'f0' X'83'	X'83'	
34	X'84'	X'f0' X'84'	X'84'	
35	X'24'	X'f0' X'24'	X'24'	
36	X'25'	X'f0' X'25'	X'25'	
37	X'85'	X'f0' X'85'	X'85'	
38	X'88'	X'f0' X'88'	X'88'	
39	X'86'	X'f0' X'86'	X'86'	
40	X'87'	X'f0' X'87'	X'87'	
41	X'27'	X'f0' X'27'	X'27'	
42	X'47'	X'f0' X'47'	X'47'	
43	X'4a'	X'f0' X'4a'	X'4a'	
44	X'79'	X'f0' X'79'	X'79'	
45	X'22'	X'f0' X'22'	X'22'	World Trade key (see Notes)
46	X'41'	X'f0' X'41'	X'41'	
47	X'42'	X'f0' X'42'	X'42'	

Table 267. RS-485/USB scan code set (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
48	X'43'	X'f0' X'43'	X'43'	
49	X'44'	X'f0' X'44'	X'44'	
50	X'34'	X'f0' X'34'	X'34'	
51	X'35'	X'f0' X'35'	X'35'	
52	X'45'	X'f0' X'45'	X'45'	
53	X'48'	X'f0' X'48'	X'48'	
54	X'46'	X'f0' X'46'	X'46'	
55	X'37'	X'f0' X'37'	X'37'	
56	X'38'	X'f0' X'38'	X'38'	ANKPOS (see Notes)
57	X'49'	X'f0' X'49'	X'49'	
58	X'50'	X'f0' X'50'	X'50'	
60	X'2d'	X'f0' X'2d'	X'2d'	
61	X'3a'	X'f0' X'3a'	X'3a'	
62	X'3d'	X'f0' X'3d'	X'3d'	
64	X'40'	X'f0' X'40'	X'40'	
75	X'5c'	X'f0' X'5c'	X'5c'	
76	X'5b'	X'f0' X'5b'	X'5b'	
77	X'5a'	X'f0' X'5a'	X'5a'	
78	X'2b'	X'f0' X'2b'	X'2b'	
79	X'3e'	X'f0' X'3e'	X'3e'	
80	X'4b'	X'f0' X'4b'	X'4b'	
81	X'1e'	X'f0' X'1e'	X'1e'	
82	X'7b'	X'f0' X'7b'	X'7b'	
83	X'2e'	X'f0' X'2e'	X'2e'	
84	X'3b'	X'f0' X'3b'	X'3b'	
85	X'8f'	X'f0' X'8f'	X'8f'	
86	X'1f'	X'f0' X'1f'	X'1f'	
87	X'7e'	X'f0' X'7e'	X'7e'	
88	X'7f'	X'f0' X'7f'	X'7f'	
89	X'3c'	X'f0' X'3c'	X'3c'	
90	X'9b'	X'f0' X'9b'	X'9b'	
91	X'6b'	X'f0' X'6b'	X'6b'	
92	X'0b'	X'f0' X'0b'	X'0b'	
93	X'8b'	X'f0' X'8b'	X'8b'	
94	X'bb'	X'f0' X'bb'	X'bb'	
95	X'9c'	X'f0' X'9c'	X'9c'	
96	X'6c'	X'f0' X'6c'	X'6c'	
97	X'0c'	X'f0' X'0c'	X'0c'	
98	X'8c'	X'f0' X'8c'	X'8c'	

Table 267. RS-485/USB scan code set (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
99	X'bc'	X'f0' X'bc'	X'bc'	
100	X'ae'	X'f0' X'ae'	X'ae'	
101	X'6f'	X'f0' X'6f'	X'6f'	
102	X'0f'	X'f0' X'0f'	X'0f'	
103	X'5f'	X'f0' X'5f'	X'5f'	
104	X'4f'	X'f0' X'4f'	X'4f'	
105	X'3f'	X'f0' X'3f'	X'3f'	
106	X'6e'	X'f0' X'6e'	X'6e'	
107	X'0e'	X'f0' X'0e'	X'0e'	
108	X'8e'	X'f0' X'8e'	X'8e'	
109	X'be'	X'f0' X'be'	X'be'	
110	X'21'	X'f0' X'21'	X'21'	
112	X'52'	X'f0' X'52'	X'52'	
113	X'53'	X'f0' X'53'	X'53'	
114	X'32'	X'f0' X'32'	X'32'	
115	X'23'	X'f0' X'23'	X'23'	
116	X'2a'	X'f0' X'2a'	X'2a'	
117	X'28'	X'f0' X'28'	X'28'	
118	X'08'	X'f0' X'08'	X'08'	
119	X'56'	X'f0' X'56'	X'56'	
120	X'07'	X'f0' X'07'	X'07'	
121	X'1a'	X'f0' X'1a'	X'1a'	
122	X'1b'	X'f0' X'1b'	X'1b'	
123	X'1c'	X'f0' X'1c'	X'1c'	
124	X'1d'	X'f0' X'1d'	X'1d'	
125	X'6d'	X'f0' X'6d'	X'6d'	
126	X'4e'	X'f0' X'4e'	X'4e'	
127	X'4c'	X'f0' X'4c'	X'4c'	
128	X'9e'	X'f0' X'9e'	X'9e'	
131	X'31'	X'f0' X'31'	X'31'	ANKPOS (see Notes)
132	X'26'	X'f0' X'26'	X'26'	ANKPOS (see Notes)
133	X'36'	X'f0' X'36'	X'36'	ANKPOS (see Notes)
135	X'8d'	X'f0' X'8d'	X'8d'	
136	X'ac'	X'f0' X'ac'	X'ac'	
137	X'ab'	X'f0' X'ab'	X'ab'	
138	X'af'	X'f0' X'af'	X'af'	
139	X'bf'	X'f0' X'bf'	X'bf'	
ANKPOS Keylock in system position				

Table 267. RS-485/USB scan code set (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
	+ Left Ctrl Case Pressed			
124 (S2)	X'f0' X'50' X'01' X'50'		X'01'	Key press only (ANKPOS keylock in system position)
135 (S1)	X'f0' X'50' X'00' X'50'		X'00'	Key press only (ANKPOS keylock in system position)
ANKPOS Keylock in operator/manager position				
	+Right Ctrl Case Pressed			
124 (S2)	X'f0' X'40' X'01' X'40'		X'01'	Key press only (ANKPOS keylock in system position)
135 (S1)	X'f0' X'40' X'00' X'40'		X'00'	Key press only (ANKPOS keylock in system position)
	+ Both Ctrl Case Pressed			
124 (S2)	X'f0' X'40' X'f0' X'50' X'01' X'50'		X'01'	Key press only (ANKPOS keylock in system position)
135 (S1)	X'f0' X'40' X'f0' X'50' X'00' X'50'		X'00'	Key press only (ANKPOS keylock in system position)
	+ Left Ctrl Case Pressed			
124 (S2)	X'f0' X'50' X'1d' X'50'		X'1d'	ANKPOS keylock in operator/manager position
135 (S1)	X'f0' X'50' X'8d' X'50'		X'8d'	ANKPOS keylock in operator/manager position
	+Right Ctrl Case Pressed			
124 (S2)	X'f0' X'40' X'1d' X'40'		X'1d'	ANKPOS keylock in operator/manager position
135 (S1)	X'f0' X'40' X'8d' X'40'		X'8d'	ANKPOS keylock in operator/manager position
	+ Both Ctrl Case Pressed			
124 (S2)	X'f0' X'40' X'f0' X'50' X'1d' X'50'		X'1d'	ANKPOS keylock in operator/manager position
135 (S1)	X'f0' X'40' X'f0' X'50' X'8d' X'50'		X'8d'	ANKPOS keylock in operator/manager position

Notes:

ANKPOS

PC Point of Sale Keyboard unique keys. These keys are present only on the PC Point of Sale (ANKPOS keyboard).

World Trade key

This key is present on all non-U.S. versions of the NANPOS keyboard, but not on the ANKPOS keyboard.

Table 268 explains the scan codes received for the S1 and S2 function keys on the Retail Alphanumeric Point of Sale Keyboard with Card Reader.

The S1 and S2 function keys send a series of scan codes on the Retail Alphanumeric Point of Sale Keyboard with Card Reader. These function keys generate a break scan code for the Ctrl key pressed (scan code 0x50 or 0x40 with PosKC_KEYUP flag set), a make scan code for the S1 or S2 key and then a make scan code for the Ctrl key pressed (scan code of 0x50 or 0x40 with PosKC_KEYUP flag reset). The S1 and S2 function keys can only be accessed by pressing the Ctrl key. With the Ctrl key pressed, key 135 represents the S1 function, and key 124 represents the S2 function.

No break scan codes are sent for the 0x00 or 0x01 scan codes.

Table 268. Point of Sale Scan Codes for Retail Alphanumeric Point of Sale Keyboard with Card Reader

Key switch number	Key type	Ctrl + scan code
124	S2	X'01'
135	S1	X'00'

For the PC Point of Sale Keyboard, Table 269 shows the scan codes sent by the S1 and S2 function keys, depending upon the Keylock Positions.

Table 269. Serial I/O Scan Codes– PC Point of Sale Keyboard (ANKPOS) Keyboard

Key switch number	Key type	Ctrl + scan code (keylock in system position)	Ctrl + scan code (keylock in operator/manager position)
124	S2	X'01'	X'1D'
135	S1	X'00'	X'8D'

USB scan code set for the Modular Compact Alphanumeric Point of Sale Keyboard

Table 270. USB scan codes for Modular Compact Alphanumeric POS Keyboard

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
1	X'51'	X'f0' X'51'	X'51'	
2	X'11'	X'f0' X'11'	X'11'	
3	X'12'	X'f0' X'12'	X'12'	
4	X'13'	X'f0' X'13'	X'13'	
5	X'14'	X'f0' X'14'	X'14'	
6	X'54'	X'f0' X'54'	X'54'	
7	X'55'	X'f0' X'55'	X'55'	
8	X'15'	X'f0' X'15'	X'15'	
9	X'18'	X'f0' X'18'	X'18'	
10	X'16'	X'f0' X'16'	X'16'	
11	X'17'	X'f0' X'17'	X'17'	
12	X'57'	X'f0' X'57'	X'57'	
13	X'58'	X'f0' X'58'	X'58'	
15	X'7a'	X'f0' X'7a'	X'7a'	

Table 270. USB scan codes for Modular Compact Alphanumeric POS Keyboard (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
16	X'71'	X'f0' X'71'	X'71'	
17	X'61'	X'f0' X'61'	X'61'	
18	X'62'	X'f0' X'62'	X'62'	
19	X'63'	X'f0' X'63'	X'63'	
20	X'64'	X'f0' X'64'	X'64'	
21	X'74'	X'f0' X'74'	X'74'	
22	X'75'	X'f0' X'75'	X'75'	
23	X'65'	X'f0' X'65'	X'65'	
24	X'68'	X'f0' X'68'	X'68'	
25	X'66'	X'f0' X'66'	X'66'	
26	X'67'	X'f0' X'67'	X'67'	
27	X'77'	X'f0' X'77'	X'77'	
28	X'78'	X'f0' X'78'	X'78'	
30	X'72'	X'f0' X'72'	X'72'	
31	X'81'	X'f0' X'81'	X'81'	
32	X'82'	X'f0' X'82'	X'82'	
33	X'83'	X'f0' X'83'	X'83'	
34	X'84'	X'f0' X'84'	X'84'	
35	X'24'	X'f0' X'24'	X'24'	
36	X'25'	X'f0' X'25'	X'25'	
37	X'85'	X'f0' X'85'	X'85'	
38	X'88'	X'f0' X'88'	X'88'	
39	X'86'	X'f0' X'86'	X'86'	
40	X'87'	X'f0' X'87'	X'87'	
41	X'27'	X'f0' X'27'	X'27'	
42	X'47'	X'f0' X'47'	X'47'	
43	X'4a'	X'f0' X'4a'	X'4a'	
44	X'79'	X'f0' X'79'	X'79'	
46	X'41'	X'f0' X'41'	X'41'	
47	X'42'	X'f0' X'42'	X'42'	
48	X'43'	X'f0' X'43'	X'43'	
49	X'44'	X'f0' X'44'	X'44'	
50	X'34'	X'f0' X'34'	X'34'	
51	X'35'	X'f0' X'35'	X'35'	
52	X'45'	X'f0' X'45'	X'45'	
53	X'48'	X'f0' X'48'	X'48'	
54	X'46'	X'f0' X'46'	X'46'	
55	X'37'	X'f0' X'37'	X'37'	
57	X'49'	X'f0' X'49'	X'49'	

Table 270. USB scan codes for Modular Compact Alphanumeric POS Keyboard (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
58	X'50'	X'f0' X'50'	X'50'	
60	X'2d'	X'f0' X'2d'	X'2d'	
61	X'3a'	X'f0' X'3a'	X'3a'	
62	X'3d'	X'f0' X'3d'	X'3d'	
64	X'40'	X'f0' X'40'	X'40'	
75	X'5c'	X'f0' X'5c'	X'5c'	
76	X'5b'	X'f0' X'5b'	X'5b'	
79	X'3e'	X'f0' X'3e'	X'3e'	
80	X'4b'	X'f0' X'4b'	X'4b'	
81	X'1e'	X'f0' X'1e'	X'1e'	
83	X'2e'	X'f0' X'2e'	X'2e'	
84	X'3b'	X'f0' X'3b'	X'3b'	
85	X'8f'	X'f0' X'8f'	X'8f'	
86	X'1f'	X'f0' X'1f'	X'1f'	
89	X'3c'	X'f0' X'3c'	X'3c'	
90	X'bf'	X'f0' X'bf'	X'bf'	
91	X'6b'	X'f0' X'6b'	X'6b'	
92	X'0b'	X'f0' X'0b'	X'0b'	
93	X'8b'	X'f0' X'8b'	X'8b'	
95	X'4f'	X'f0' X'4f'	X'4f'	
96	X'6c'	X'f0' X'6c'	X'6c'	
97	X'0c'	X'f0' X'0c'	X'0c'	
98	X'8c'	X'f0' X'8c'	X'8c'	
99	X'bb'	X'f0' X'bb'	X'bb'	
100	X'ae'	X'f0' X'ae'	X'ae'	
101	X'6f'	X'f0' X'6f'	X'6f'	
102	X'0f'	X'f0' X'0f'	X'0f'	
103	X'5f'	X'f0' X'5f '	X'5f'	
104	X'9c'	X'f0' X'9c'	X'9c'	
105	X'3f'	X'f0' X'3f'	X'3f'	
106	X'6e'	X'f0' X'6e'	X'6e'	
108	X'be'	X'f0' X'be'	X'be'	
110	X'21'	X'f0' X'21'	X'21'	
112	X'52'	X'f0' X'52'	X'52'	
113	X'53'	X'f0' X'53'	X'53'	
114	X'32'	X'f0' X'32'	X'32'	
115	X'23'	X'f0' X'23'	X'23'	
116	X'2a'	X'f0' X'2a'	X'2a'	
117	X'28'	X'f0' X'28'	X'28'	

Table 270. USB scan codes for Modular Compact Alphanumeric POS Keyboard (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
118	X'08'	X'f0' X'08'	X'08'	
119	X'56'	X'f0' X'56'	X'56'	
120	X'07'	X'f0' X'07'	X'07'	
121	X'1a'	X'f0' X'1a'	X'1a'	
122	X'1b'	X'f0' X'1b'	X'1b'	
123	X'1c'	X'f0' X'1c'	X'1c'	
124	X'ac'	X'f0' X'ac'	X'ac'	
125	X'ab'	X'f0' X'ab'	X'ab'	
126	X'af'	X'f0' X'af'	X'af'	
P01	X'4c'	X'f0' X'4c'	X'4c'	
P02	X'9e'	X'f0' X'9e'	X'9e'	
P03	X'4b'	X'f0' X'4b'	X'4b'	
P04	X'3b'	X'f0' X'3b'	X'3b'	
P05	X'6b'	X'f0' X'6f'	X'6b'	
P06	X'4c'	X'f0' X'4c'	X'4c'	
P07	X'3c'	X'f0' X'3c'	X'3c'	
P08	X'6c'	X'f0' X'6c'	X'6c'	
P09	X'4f'	X'f0' X'4f'	X'4f'	
P10	X'3f'	X'f0' X'3f'	X'3f'	
P11	X'1d'	X'f0' X'1d'	X'1d'	
P12	X'6f'	X'f0' X'6f'	X'6f'	
P13	X'4e'	X'f0' X'4e'	X'4e'	
P14	X'3e'	X'f0' X'3e'	X'3e'	
P15	X'8d'	X'f0' X'8d'	X'8d'	
P16	X'6e'	X'f0' X'6e'	X'6e'	
P17	X'5a'	X'f0' X'5a'	X'5a'	
P18	X'2b'	X'f0' X'2b'	X'2b'	
P19	X'7b'	X'f0' X'7b'	X'7b'	
P20	X'7e'	X'f0' X'7e'	X'7e'	
P21	X'7f'	X'f0' X'7f'	X'7f'	
P22	X'7c'	X'f0' X'7c'	X'7c'	
P23	X'7d'	X'f0' X'7d'	X'7d'	
P24	X'0d'	X'f0' X'0d'	X'0d'	
P25	X'9b'	X'f0' X'9b'	X'9b'	
P26	X'bc'	X'f0' X'bc'	X'bc'	
P27	X'0e'	X'f0' X'0e'	X'0e'	
P28	X'8e'	X'f0' X'8e'	X'8e'	
P29	X'6d'	X'f0' X'6d'	X'6d'	
P30	X'4e'	X'f0' X'4e'	X'4e'	

Table 270. USB scan codes for Modular Compact Alphanumeric POS Keyboard (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
P31	X'4d'	X'f0' X'4d'	X'4d'	
P32	X'3d'	X'f0' X'3d'	X'3d'	

Retail Alphanumeric Point of Sale keyboard with card reader (PS/2 or USB system attached)

This section contains illustrations of the layouts for the following keyboards:

- Retail Alphanumeric POSKeyboard with card reader (ANKPOS)
- Retail Alphanumeric POSKeyboard with card reader (NANPOS)
- Compact Alphanumeric POSKeyboard (CANPOS)
- Modular Alphanumeric POS Keyboard
- Modular Compact Alphanumeric POS Keyboard

Retail Alphanumeric Point of Sale with card reader or PC Point of Sale keyboard (ANKPOS) layout keyboard

Figure 52 shows the layout and assigned key-switch numbers for the PC Point of Sale keyboard. The numeric keypad location is shown in the shaded area of the illustration.

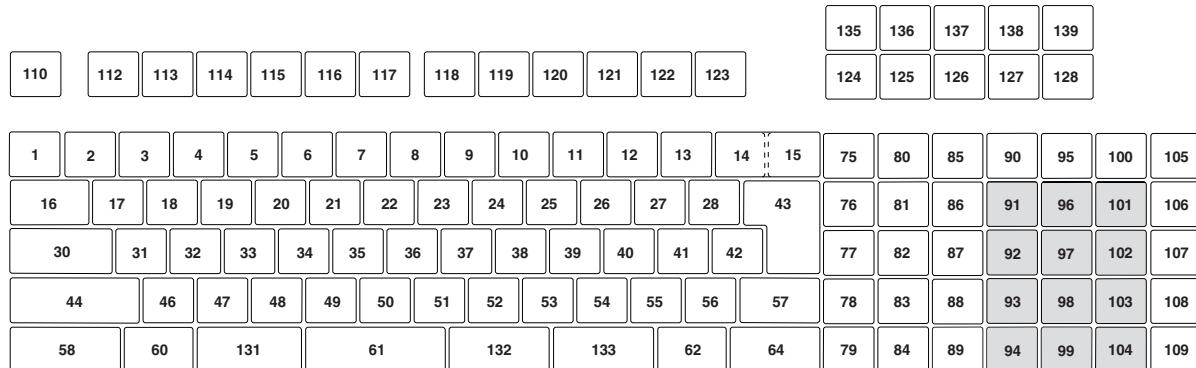


Figure 52. PC Point of Sale keyboard layout

Retail Alphanumeric Point of Sale with card reader (NANPOS) layout keyboard

Figure 53 on page 328 shows the layout and assigned key-switch numbers for the Retail Alphanumeric Point of Sale Keyboard with Card Reader.

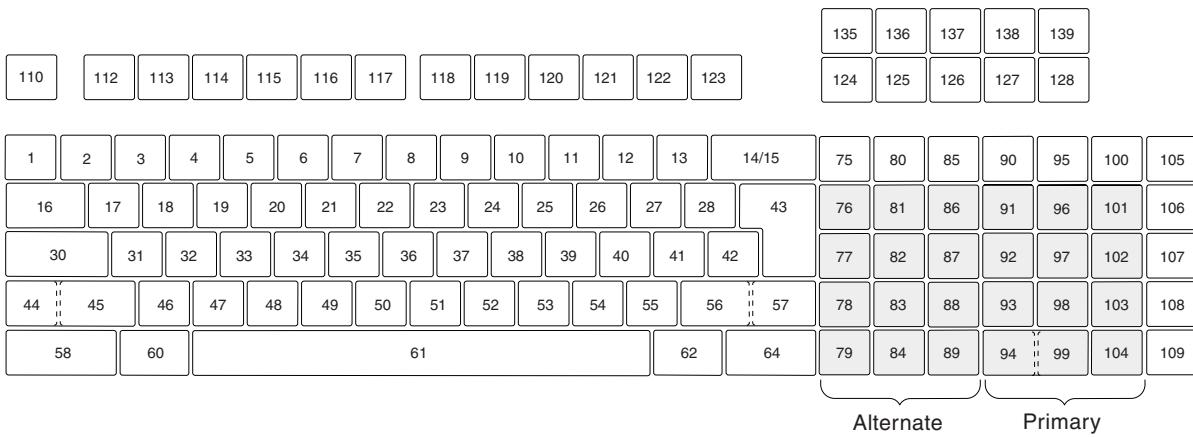


Figure 53. Layout and assigned switch numbers

Note:

1. The two possible locations for the numeric keypad are shown in the shaded area of the illustration. The default location for the numeric keypad is the right-most shaded area.
2. Key 45 appears as a single key only on non-U.S. keyboards. On U.S. keyboards, key 44 also covers key 45 (key 44, 45 is a double key).
3. Keys 94 and 99 can have a single, horizontal double-wide key covering both keys, or they can be split into two individual keys.
4. This keyboard is similar to the 101-enhanced keyboard and the 102-enhanced keyboard. The following keys are on the Retail Alphanumeric Point of Sale Keyboard with Card Reader, but not on the 101-enhanced keyboard or the 102-enhanced keyboard. These keys are referred to as the *point-of-sale-unique keys*.

77	106
78	107
82	108
87	124
88	125
90	126
95	127
99	128
100	135
105	

See Table 271 on page 330 for the scan codes associated with these keys.

Modular Alphanumeric Point of Sale Keyboard

Figure 54 on page 329 shows the layout and assigned key-switch numbers for the Modular Alphanumeric Point of Sale Keyboard.

110	112	113	114	115	116	117	118	119	120	77	82	87	135	136	137	138	139
										121	122	123	124	125	126	127	128
1	2	3	4	5	6	7	8	9	10	11	12	13	14/15	75	80	85	90
16	17	18	19	20	21	22	23	24	25	26	27	28	43	76	81	86	91
30	31	32	33	34	35	36	37	38	39	40	41	42		78	83	88	92
44	45	46	47	48	49	50	51	52	53	54	55	56	57	79	84	89	93
58	60					61				62	64			94	99	104	109

Figure 54. Layout and Assigned Switch Numbers for Modular ANPOS Keyboard

See Table 271 on page 330 for the scan codes associated with these keys.

Compact Alphanumeric Point of Sale (CANPOS) layout keyboard

Figure 55 shows the layout and assigned key-switch numbers for the Compact Alphanumeric Point of Sale keyboard with card reader.

201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216		
217	218	219	220	221	222	223	224	225	226	227	228	229	230	231	232		
110	112	113	114	115	116	117	118	119	120	121	122	123	124	126	75	80	85
1	2	3	4	5	6	7	8	9	10	11	12	13	15	125	76	81	86
16	17	18	19	20	21	22	23	24	25	26	27	28	43	90	95	100	105
30	31	32	33	34	35	36	37	38	39	40	41	29		91	96	101	106
44	46	47	48	49	50	51	52	53	54	55	(56)	57		92	97	102	108
58	60					61			62	64				93	98	103	108
									79	83	88			99	104		
									80	85	90						

Figure 55. Layout and assigned switch numbers for CANPOS keyboard

Note: See Table 272 on page 337 for scan codes related to this keyboard.

Modular Compact Alphanumeric Point of Sale Keyboard Layout

Figure 56 on page 330 shows the layout and assigned key-switch numbers for the Modular Compact Alphanumeric Point of Sale Keyboard.

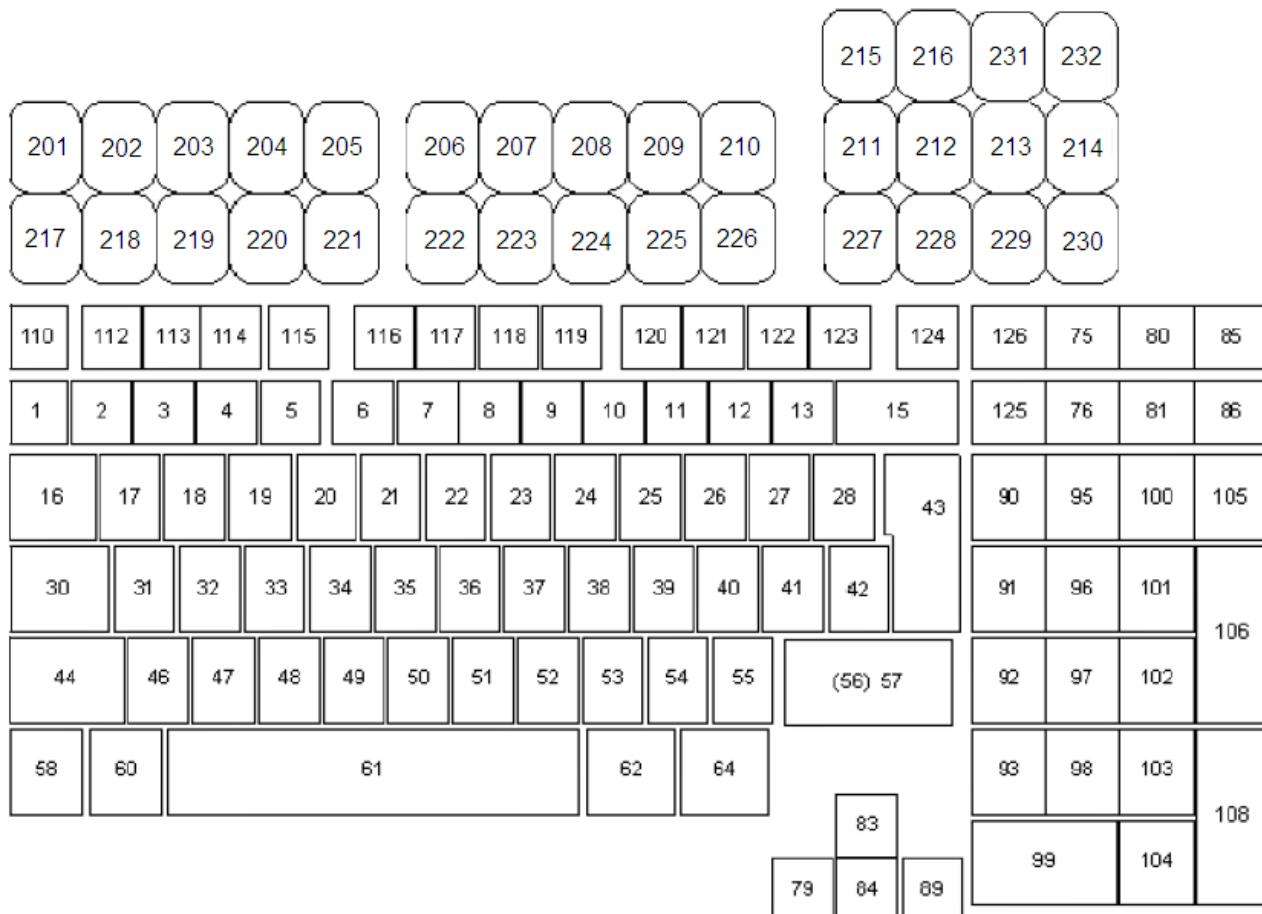


Figure 56. Modular CANPOS Keyboard Layout and Assigned Switch Numbers

Note: See Table 272 on page 337 for scan codes related to this keyboard.

PS/2 or USB (System Attached) scan code set for Retail Alphanumeric Point of Sale Keyboard (NANPOS), Modular Alphanumeric Point of Sale Keyboard, PC Point of Sale Keyboard (ANKPOS)

In Linux, Modular ANPOS Keyboard scan codes may not match those provided in the table below until a configuration file is downloaded to the device. Follow the steps listed below to download the configuration file:

- Copy the aip46043.cfg.winCompatibility file from /opt/ibm/javapos/config to /usr/share/pos/config/aip46043.cfg .
- Reboot the system.

Table 271 relates the keyboard key-switch number to the scan codes received when the keyboard is attached to the system keyboard port.

Table 271. PS/2 or USB (system attached) scan codes

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
1	X'29'	X'A9'	X'29'	
2	X'02'	X'82'	X'02'	
3	X'03'	X'83'	X'03'	

Table 271. PS/2 or USB (system attached) scan codes (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
4	X'04'	X'84'	X'04'	
5	X'05'	X'85'	X'05'	
6	X'06'	X'86'	X'06'	
7	X'07'	X'87'	X'07'	
8	X'08'	X'88'	X'08'	
9	X'09'	X'89'	X'09'	
10	X'0a'	X'8a'	X'0a'	
11	X'0b'	X'8b'	X'0b'	
12	X'0c'	X'8c'	X'0c'	
13	X'0d'	X'8d'	X'0d'	
14	X'7d'	X'fd'	X'7d'	ANKPOS only
15	X'0e'	X'8e'	X'0e'	
16	X'0f'	X'8f'	X'0f'	
17	X'10'	X'90'	X'10'	
18	X'11'	X'91'	X'11'	
19	X'12'	X'92'	X'12'	
20	X'13'	X'93'	X'13'	
21	X'14'	X'94'	X'14'	
22	X'15'	X'95'	X'15'	
23	X'16'	X'96'	X'16'	
24	X'17'	X'97'	X'17'	
25	X'18'	X'98'	X'18'	
26	X'19'	X'99'	X'19'	
27	X'1a'	X'9a'	X'1a'	
28	X'1b'	X'9b'	X'1b'	
30	X'3a'	X'ba'	X'3a'	
31	X'1e'	X'9e'	X'1e'	
32	X'1f'	X'9f'	X'1f'	
33	X'20'	X'a0'	X'20'	
34	X'21'	X'a1'	X'21'	
35	X'22'	X'a2'	X'22'	
36	X'23'	X'a3'	X'23'	
37	X'24'	X'a4'	X'24'	
38	X'25'	X'a5'	X'25'	
39	X'26'	X'a6'	X'26'	
40	X'27'	X'a7'	X'27'	
41	X'28'	X'a8'	X'28'	
42	X'2b'	X'ab'	X'2b'	
43	X'1c'	X'9c'	X'1c'	

Table 271. PS/2 or USB (system attached) scan codes (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
44	X'2a'	X'aa'	X'2a'	
45	X'56'	X'd6'	X'56'	World Trade key
46	X'2c'	X'ac'	X'2c'	
47	X'2d'	X'ad'	X'2d'	
48	X'2e'	X'ae'	X'2e'	
49	X'2f'	X'af'	X'2f'	
50	X'30'	X'b0'	X'30'	
51	X'31'	X'b1'	X'31'	
52	X'32'	X'b2'	X'32'	
53	X'33'	X'b3'	X'33'	
54	X'34'	X'b4'	X'34'	
55	X'35'	X'b5'	X'35'	
56	X'73'	X'f3'	X'73'	ANKPOS only
57	X'36'	X'b6'	X'36'	
58	X'1d'	X'9d'	X'1d'	
60	X'38'	X'b8'	X'38'	
61	X'39'	X'b9'	X'39'	
62	X'e0' X'38'	X'e0' X'b8'	X'01' X'38'	
64	X'e0' X'1d'	X'e0' X'9d'	X'01' X'1d'	
75	X'e0' X'52'	X'e0' X'd2'	X'01' X'52'	
76	X'e0' X'53'	X'e0' X'd3'	X'01' X'53'	
77	X'6a'	X'ea'	X'6a'	
78	X'6b'	X'eb'	X'6b'	
79	X'e0' X'4b'	X'e0' X'cb'	X'01' X'4b'	
80	X'e0' X'47'	X'e0' X'c7'	X'01' X'47'	
81	X'e0' X'4f'	X'e0' X'cf'	X'01' X'4f'	
82	X'6c'	X'ec'	X'6c'	
83	X'e0' X'48'	X'e0' X'c8'	X'01' X'48'	
84	X'e0' X'50'	X'e0' X'd0'	X'01' X'50'	
85	X'e0' X'49'	X'e0' X'c9'	X'01' X'49'	
86	X'e0' X'51'	X'e0' X'd1'	X'01' X'51'	
87	X'6d'	X'ed'	X'6d'	
88	X'6e'	X'ee'	X'6e'	
89	X'e0' X'4d'	X'e0' X'cd'	X'01' X'4d'	
90	X'6f'	X'ef'	X'6f'	
91	X'47'	X'c7'	X'47'	
92	X'4b'	X'cb'	X'4b'	
93	X'4f'	X'cf'	X'4f'	
94	X'52'	X'd2'	X'52'	

Table 271. PS/2 or USB (system attached) scan codes (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
95	X'78'	X'f8'	X'78'	
96	X'48'	X'c8'	X'48'	
97	X'4c'	X'cc'	X'4c'	
98	X'50'	X'd0'	X'50'	
99	X'77'	X'f7'	X'52'	
100	X'65'	X'e5'	X'65'	
101	X'49'	X'c9'	X'49'	
102	X'4d'	X'cd'	X'4d'	
103	X'51'	X'd1'	X'51'	
104	X'53'	X'd3'	X'53'	
105	X'7a'	X'fa'	X'7a'	
106	X'7e'	X'fe'	X'7e'	
107	X'5f'	X'df'	X'5f'	When using a USB system keyboard with the OPOS driver the make code will be reported as X'66', not X'5f'.
108	X'71'	X'f1'	X'71'	
109	X'e0' X'1c'	X'e0' X'9c'	X'01' X'1c'	
110	X'01'	X'81'	X'01'	
112	X'3b'	X'bb'	X'3b'	
113	X'3c'	X'bc'	X'3c'	
114	X'3d'	X'bd'	X'3d'	
115	X'3e'	X'be'	X'3e'	
116	X'3f'	X'bf'	X'3f'	
117	X'40'	X'c0'	X'40'	
118	X'41'	X'c1'	X'41'	
119	X'42'	X'c2'	X'42'	
120	X'43'	X'c3'	X'43'	
121	X'44'	X'c4'	X'44'	
122	X'57'	X'd7'	X'57'	
123	X'58'	X'd8'	X'58'	
124	X'63'	X'e3'	X'63'	
125	X'74'	X'f4'	X'74'	
126	X'75'	X'f5'	X'75'	
127	X'76'	X'f6'	X'76'	
128	X'59'	X'd9'	X'59'	
131	X'7b'	X'fb'	X'7b'	ANKPOS only
132	X'79'	X'f9'	X'79'	ANKPOS only
133	X'70'	X'f0'	X'70'	ANKPOS only

Table 271. PS/2 or USB (system attached) scan codes (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
135	X'72'	X'f2'	X'72'	
136	X'e0' X'2a' X'e0' X'37'	X'e0' X'b7' X'e0' X'aa'	X'01' X'37'	On Windows, key release only
137	X'46'	X'c6'	X'46'	
138	X'e1' X'1d' X'45' X'e1' X'9d' X'c5'		X'01' X'45'	Key press only
139	X'45'	X'c5'	X'45'	
	+ Alt Case			
136	X'54'	X'd4'	X'54'	On Windows, key release only
	+ Ctrl Case Pressed			
136	X'e0' X'37'	X'e0' X'b7'	X'01' X'37'	On Windows, key release only
138	X'e0' X'46' X'e0' X'c6'		none	
	+ Left Ctrl Case Pressed			
124	X'9d' X'1c' X'1d'		X'02' X'1c'	Key press only
135	X'9d' X'01' X'1d'		X'02' X'01'	Key press only
	+ Right Ctrl Case Pressed			
124	X'e0' X'9d' X'1c' X'e0' X'1d'		X'02' X'1c'	Key press only
135	X'e0' X'9d' X'01' X'e0' X'1d'		X'02' X'01'	Key press only
	+ Both Ctrl Case Pressed			
124	X'e0' X'9d' X'9d' X'1c' X'e0' X'1d' X'e0' X'1d'		X'02' X'1c'	Key press only
135	X'e0' X'9d' X'9d' X'01' X'e0' X'1d' X'e0' X'1d'		X'02' X'01'	Key press only
	+ Shift Case Pressed			
136	X'e0' X'37'	X'e0' X'b7'	X'01' X'37'	On Windows, key release only
	+ Left Shift Case Pressed, NUMLOCK OFF			
75	X'e0' X'aa' X'e0' X'52'	X'e0' X'd2' X'e0' X'2a'	X'01' X'52'	
76	X'e0' X'aa' X'e0' X'53'	X'e0' X'd3' X'e0' X'2a'	X'01' X'53'	
79	X'e0' X'aa' X'e0' X'4b'	X'e0' X'cb' X'e0' X'2a'	X'01' X'4b'	

Table 271. PS/2 or USB (system attached) scan codes (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
80	X'e0' X'aa' X'e0' X'47'	X'e0' X'c7' X'e0' X'2a'	X'01' X'47'	
81	X'e0' X'aa' X'e0' X'4f'	X'e0' X'cf' X'e0' X'2a'	X'01' X'4f'	
83	X'e0' X'aa' X'e0' X'48'	X'e0' X'c8' X'e0' X'2a'	X'01' X'48'	
84	X'e0' X'aa' X'e0' X'50'	X'e0' X'd0' X'e0' X'2a'	X'01' X'50'	
85	X'e0' X'aa' X'e0' X'49'	X'e0' X'c9' X'e0' X'2a'	X'01' X'49'	
86	X'e0' X'aa' X'e0' X'51'	X'e0' X'd1' X'e0' X'2a'	X'01' X'51'	
89	X'e0' X'aa' X'e0' X'4d'	X'e0' X'cd' X'e0' X'2a'	X'01' X'4d'	
	+ Right Shift Case Pressed NUMLOCK OFF			
75	X'e0' X'b6' X'e0' X'52'	X'e0' X'd2' X'e0' X'36'	X'01' X'52'	
76	X'e0' X'b6' X'e0' X'53'	X'e0' X'd3' X'e0' X'36'	X'01' X'53'	
79	X'e0' X'b6' X'e0' X'4b'	X'e0' X'cb' X'e0' X'36'	X'01' X'4b'	
80	X'e0' X'b6' X'e0' X'47'	X'e0' X'c7' X'e0' X'36'	X'01' X'47'	
81	X'e0' X'b6' X'e0' X'4f'	X'e0' X'cf' X'e0' X'36'	X'01' X'4f '	
83	X'e0' X'b6' X'e0' X'48'	X'e0' X'c8' X'e0' X'36'	X'01' X'48'	
84	X'e0' X'b6' X'e0' X'50'	X'e0' X'd0' X'e0' X'36'	X'01' X'50'	
85	X'e0' X'b6' X'e0' X'49'	X'e0' X'c9' X'e0' X'36'	X'01' X'49'	
86	X'e0' X'b6' X'e0' X'51'	X'e0' X'd1' X'e0' X'36'	X'01' X'51'	
89	X'e0' X'b6' X'e0' X'4d'	X'e0' X'cd' X'e0' X'36'	X'01' X'4d'	
	+ Both Shift Case Pressed NUMLOCK OFF			
75	X'e0' X'aa' X'b6' X'e0' X'52'	X'e0' X'd2' X'e0' X'2a' X'e0' X'36'	X'01' X'52'	
76	X'e0' X'aa' X'b6' X'e0' X'53'	X'e0' X'd3' X'e0' X'2a' X'e0' X'36'	X'01' X'53'	
79	X'e0' X'aa' X'b6' X'e0' X'4b'	X'e0' X'cb' X'e0' X'2a' X'e0' X'36'	X'01' X'4b'	

Table 271. PS/2 or USB (system attached) scan codes (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
81	X'e0' X'aa' X'b6' X'e0' X'4f'	X'e0' X'cf' X'e0' X'2a' X'e0' X'36'	X'01' X'4f'	
83	X'e0' X'aa' X'b6' X'e0' X'48'	X'e0' X'c8' X'e0' X'2a' X'e0' X'36'	X'01' X'48'	
84	X'e0' X'aa' X'b6' X'e0' X'50'	X'e0' X'd0' X'e0' X'2a' X'e0' X'36'	X'01' X'50'	
85	X'e0' X'aa' X'b6' X'e0' X'49'	X'e0' X'c9' X'e0' X'2a' X'e0' X'36'	X'01' X'49'	
86	X'e0' X'aa' X'b6' X'e0' X'51'	X'e0' X'd1' X'e0' X'2a' X'e0' X'36'	X'01' X'51'	
89	X'e0' X'aa' X'b6' X'e0' X'4d'	X'e0' X'cd' X'e0' X'2a' X'e0' X'36'	X'01' X'4d'	
	+ Shift + NUM LOCK ON			
75	X'e0' X'52'	X'e0' X'd2'	X'01' X'52'	
76	X'e0' X'53'	X'e0' X'd3'	X'01' X'53'	
79	X'e0' X'4b'	X'e0' X'cb'	X'01' X'4b'	
80	X'e0' X'47'	X'e0' X'c7'	X'01' X'47'	
81	X'e0' X'4f'	X'e0' X'cf'	X'01' X'4f'	
83	X'e0' X'48'	X'e0' X'c8'	X'01' X'48'	
84	X'e0' X'50'	X'e0' X'd0'	X'01' X'50'	
85	X'e0' X'49'	X'e0' X'c9'	X'01' X'49'	
86	X'e0' X'51'	X'e0' X'd1'	X'01' X'51'	
89	X'e0' X'4d'	X'e0' X'cd'	X'01' X'4d'	
	+ NUM LOCK ON			
75	X'e0' X'2a' X'e0' X'52'	X'e0' X'd2' X'e0' X'aa'	X'01' X'52'	
76	X'e0' X'2a' X'e0' X'53'	X'e0' X'd3' X'e0' X'aa'	X'01' X'53'	
79	X'e0' X'2a' X'e0' X'4b'	X'e0' X'cb' X'e0' X'aa'	X'01' X'4b'	
80	X'e0' X'2a' X'e0' X'47'	X'e0' X'c7' X'e0' X'aa'	X'01' X'47'	
81	X'e0' X'2a' X'e0' X'4f'	X'e0' X'cf' X'e0' X'aa'	X'01' X'4f'	
83	X'e0' X'2a' X'e0' X'48'	X'e0' X'c8' X'e0' X'aa'	X'01' X'48'	
84	X'e0' X'2a' X'e0' X'50'	X'e0' X'd0' X'e0' X'aa'	X'01' X'50'	
85	X'e0' X'2a' X'e0' X'49'	X'e0' X'c9' X'e0' X'aa'	X'01' X'49'	
86	X'e0' X'2a' X'e0' X'51'	X'e0' X'd1' X'e0' X'aa'	X'01' X'51'	

Table 271. PS/2 or USB (system attached) scan codes (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
89	X'e0' X'2a' X'e0' X'4d'	X'e0' X'cd' X'e0' X'aa'	X'01' X'4d'	

Notes:

ANKPOS

PC Point of Sale Keyboard unique keys. These keys are present only on the PC Point of Sale (ANKPOS keyboard).

World Trade key

This key is present on all non-U.S. versions of the NANPOS keyboard, but not on the ANKPOS keyboard.

Double keys

Key switch numbers 94 and 99 are defined as double keys.

PS/2 scan code set for the Compact Alphanumeric Point of Sale Keyboard (CANPOS), PS/2 / USB (System Attached) Modular Compact Alphanumeric Point of Sale Keyboard

Table 272 relates the keyboard key-switch number to the scan codes received when the keyboard is attached to the system keyboard port.

Table 272. PS/2 or USB scan codes for CANPOS keyboard

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
1	X'29'	X'A9'	X'29'	
2	X'02'	X'82'	X'02'	
3	X'03'	X'83'	X'03'	
4	X'04'	X'84'	X'04'	
5	X'05'	X'85'	X'05'	
6	X'06'	X'86'	X'06'	
7	X'07'	X'87'	X'07'	
8	X'08'	X'88'	X'08'	
9	X'09'	X'89'	X'09'	
10	X'0a'	X'8a'	X'0a'	
11	X'0b'	X'8b'	X'0b'	
12	X'0c'	X'8c'	X'0c'	
13	X'0d'	X'8d'	X'0d'	
15	X'0e'	X'8e'	X'0e'	
16	X'0f'	X'8f'	X'0f'	
17	X'10'	X'90'	X'10'	
18	X'11'	X'91'	X'11'	
19	X'12'	X'92'	X'12'	
20	X'13'	X'93'	X'13'	
21	X'14'	X'94'	X'14'	

Table 272. PS/2 or USB scan codes for CANPOS keyboard (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
22	X'15'	X'95'	X'15'	
23	X'16'	X'96'	X'16'	
24	X'17'	X'97'	X'17'	
25	X'18'	X'98'	X'18'	
26	X'19'	X'99'	X'19'	
27	X'1a'	X'9a'	X'1a'	
28	X'1b'	X'9b'	X'1b'	
29	X'2b'	X'ab'	X'2b'	
30	X'3a'	X'ba'	X'3a'	
31	X'1e'	X'9e'	X'1e'	
32	X'1f'	X'9f'	X'1f'	
33	X'20'	X'a0'	X'20'	
34	X'21'	X'a1'	X'21'	
35	X'22'	X'a2'	X'22'	
36	X'23'	X'a3'	X'23'	
37	X'24'	X'a4'	X'24'	
38	X'25'	X'a5'	X'25'	
39	X'26'	X'a6'	X'26'	
40	X'27'	X'a7'	X'27'	
41	X'28'	X'a8'	X'28'	
42	X'2b'	X'ab'	X'2b'	
43	X'1c'	X'9c'	X'1c'	
44	X'2a'	X'aa'	X'2a'	
45	X'56'	X'd6'	X'56'	
46	X'2c'	X'ac'	X'2c'	
47	X'2d'	X'ad'	X'2d'	
48	X'2e'	X'ae'	X'2e'	
49	X'2f'	X'af'	X'2f'	
50	X'30'	X'b0'	X'30'	
51	X'31'	X'b1'	X'31'	
52	X'32'	X'b2'	X'32'	
53	X'33'	X'b3'	X'33'	
54	X'34'	X'b4'	X'34'	
55	X'35'	X'b5'	X'35'	
56	X'73'	X'f3'	X'73'	
57	X'36'	X'b6'	X'36'	
58	X'1d'	X'9d'	X'1d'	
60	X'38'	X'b8'	X'38'	
61	X'39'	X'b9'	X'39'	

Table 272. PS/2 or USB scan codes for CANPOS keyboard (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
62	X'e0' X'38'	X'e0' X'b8'	X'01' X'38'	
64	X'e0' X'1d'	X'e0' X'9d'	X'01' X'1d'	
75	X'e0' X'52'	X'e0' X'd2'	X'01' X'52'	
76	X'e0' X'53'	X'e0' X'd3'	X'01' X'53'	
79	X'e0' X'4b'	X'e0' X'cb'	X'01' X'4b'	
80	X'e0' X'47'	X'e0' X'c7'	X'01' X'47'	
81	X'e0' X'4f'	X'e0' X'cf'	X'01' X'4f'	
83	X'e0' X'48'	X'e0' X'c8'	X'01' X'48'	
84	X'e0' X'50'	X'e0' X'd0'	X'01' X'50'	
85	X'e0' X'49'	X'e0' X'c9'	X'01' X'49'	
86	X'e0' X'51'	X'e0' X'd1'	X'01' X'51'	
89	X'e0' X'4d'	X'e0' X'cd'	X'01' X'4d'	
90	X'45'	X'c5'	X'45'	
91	X'47'	X'c7'	X'47'	
92	X'4b'	X'cb'	X'4b'	
93	X'4f'	X'cf'	X'4f'	
95	X'e0' X'35'	X'e0' X'b5'	X'01' X'35'	See 343.
96	X'48'	X'c8'	X'48'	
97	X'4c'	X'cc'	X'4c'	
98	X'50'	X'd0'	X'50'	
99	X'52'	X'd2'	X'52'	
100	X'37'	X'b7'	X'37'	See 343.
101	X'49'	X'c9'	X'49'	
102	X'4d'	X'cd'	X'4d'	
103	X'51'	X'd1'	X'51'	
104	X'53'	X'd3'	X'53'	See 343.
105	X'4a'	X'ca'	X'4a'	See 343.
106	X'4e'	X'ce'	X'4e'	See 343.
107	X'7e'	X'fe'	X'7e'	
108	X'e0' X'1c'	X'e0' X'9c'	X'01' X'1c'	
109	X'78'	X'f8'	X'78'	
110	X'01'	X'81'	X'01'	
112	X'3b'	X'bb'	X'3b'	
113	X'3c'	X'bc'	X'3c'	
114	X'3d'	X'bd'	X'3d'	
115	X'3e'	X'be'	X'3e'	
116	X'3f'	X'bf'	X'3f'	
117	X'40'	X'c0'	X'40'	
118	X'41'	X'c1'	X'41'	

Table 272. PS/2 or USB scan codes for CANPOS keyboard (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
119	X'42'	X'c2'	X'42'	
120	X'43'	X'c3'	X'43'	
121	X'44'	X'c4'	X'44'	
122	X'57'	X'd7'	X'57'	
123	X'58'	X'd8'	X'58'	
124	X'e0' X'2a' X'e0' X'37'	X'e0' X'b7' X'e0' X'aa'	X'01' X'37'	On Windows, key release only
125	X'46'	X'c6'	X'46'	
126	X'e1' X'1d' X'45' X'e1' X'9d' X'c5'		X'01' X'45'	Key press only
	+ Alt Case			
124	X'54'	X'd4'	X'54'	On Windows, key release only
	+ Left or Right Ctrl Pressed			
126	X'e0' X'46' X'e0' X'c6'		none	On Windows, key release only
	+ Left Ctrl or Shift Case Pressed			
124	X'e0' X'37'	X'e0' X'b7'	X'01' X'37'	On Windows, key release only
	+ Right Ctrl or Shift Case Pressed			
124	X'e0' X'37'	X'e0' X'b7'	X'01' X'37'	On Windows, key release only
	+ Both Ctrl or Shift Case Pressed			
124	X'e0' X'37'	X'e0' X'b7'	X'01' X'37'	On Windows, key release only
	+ Left Shift Case Pressed			
75	X'e0' X'aa' X'e0' X'52'	X'e0' X'd2' X'e0' X'2a'	X'01' X'52'	
76	X'e0' X'aa' X'e0' X'53'	X'e0' X'd3' X'e0' X'2a'	X'01' X'53'	
79	X'e0' X'aa' X'e0' X'4b'	X'e0' X'cb' X'e0' X'2a'	X'01' X'4b'	
80	X'e0' X'aa' X'e0' X'47'	X'e0' X'c7' X'e0' X'2a'	X'01' X'47'	
81	X'e0' X'aa' X'e0' X'4f'	X'e0' X'cf' X'e0' X'2a'	X'01' X'4f'	
83	X'e0' X'aa' X'e0' X'48'	X'e0' X'c8' X'e0' X'2a'	X'01' X'48'	
84	X'e0' X'aa' X'e0' X'50'	X'e0' X'd0' X'e0' X'2a'	X'01' X'50'	

Table 272. PS/2 or USB scan codes for CANPOS keyboard (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
85	X'e0' X'aa' X'e0' X'49'	X'e0' X'c9' X'e0' X'2a'	X'01' X'49'	
86	X'e0' X'aa' X'e0' X'51'	X'e0' X'd1' X'e0' X'2a'	X'01' X'51'	
89	X'e0' X'aa' X'e0' X'4d'	X'e0' X'cd' X'e0' X'2a'	X'01' X'4d'	
124	X'e0' X'37'	X'e0' X'b7'	X'01' X'37'	On Windows, key release only
	+ Right Shift Case Pressed			
75	X'e0' X'b6' X'e0' X'52'	X'e0' X'd2' X'e0' X'36'	X'01' X'52'	
76	X'e0' X'b6' X'e0' X'53'	X'e0' X'd3' X'e0' X'36'	X'01' X'53'	
79	X'e0' X'b6' X'e0' X'4b'	X'e0' X'cb' X'e0' X'36'	X'01' X'4b'	
80	X'e0' X'b6' X'e0' X'47'	X'e0' X'c7' X'e0' X'36'	X'01' X'47'	
81	X'e0' X'b6' X'e0' X'4f'	X'e0' X'cf' X'e0' X'36'	X'01' X'4f'	
83	X'e0' X'b6' X'e0' X'48'	X'e0' X'c8' X'e0' X'36'	X'01' X'48'	
84	X'e0' X'b6' X'e0' X'50'	X'e0' X'd0' X'e0' X'36'	X'01' X'50'	
85	X'e0' X'b6' X'e0' X'49'	X'e0' X'c9' X'e0' X'36'	X'01' X'49'	
86	X'e0' X'b6' X'e0' X'51'	X'e0' X'd1' X'e0' X'36'	X'01' X'51'	
89	X'e0' X'b6' X'e0' X'4d'	X'e0' X'cd' X'e0' X'36'	X'01' X'4d'	
124	X'e0' X'37'	X'e0' X'b7'	X'01' X'37'	On Windows, key release only
	+ Both Shift Case Pressed			
75	X'e0' X'aa' X'b6' X'e0' X'52'	X'e0' X'd2' X'e0' X'2a' X'e0' X'36'	X'01' X'52'	
76	X'e0' X'aa' X'b6' X'e0' X'53'	X'e0' X'd3' X'e0' X'2a' X'e0' X'36'	X'01' X'53'	
79	X'e0' X'aa' X'b6' X'e0' X'4b'	X'e0' X'cb' X'e0' X'2a' X'e0' X'36'	X'01' X'4b'	
81	X'e0' X'aa' X'b6' X'e0' X'4f'	X'e0' X'cf' X'e0' X'2a' X'e0' X'36'	X'01' X'4f'	
83	X'e0' X'aa' X'b6' X'e0' X'48'	X'e0' X'c8' X'e0' X'2a' X'e0' X'36'	X'01' X'48'	
84	X'e0' X'aa' X'b6' X'e0' X'50'	X'e0' X'd0' X'e0' X'2a' X'e0' X'36'	X'01' X'50'	

Table 272. PS/2 or USB scan codes for CANPOS keyboard (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
85	X'e0' X'aa' X'b6' X'e0' X'49'	X'e0' X'c9' X'e0' X'2a' X'e0' X'36'	X'01' X'49'	
86	X'e0' X'aa' X'b6' X'e0' X'51'	X'e0' X'd1' X'e0' X'2a' X'e0' X'36'	X'01' X'51'	
89	X'e0' X'aa' X'b6' X'e0' X'4d'	X'e0' X'cd' X'e0' X'2a' X'e0' X'36'	X'01' X'4d'	
124	X'e0' X'37'	X'e0' X'b7'	X'01' X'37'	On Windows, key release only
	+ Shift + NUM LOCK ON			
75	X'e0' X'52'	X'e0' X'd2'	X'01' X'52'	
76	X'e0' X'53'	X'e0' X'd3'	X'01' X'53'	
79	X'e0' X'4b'	X'e0' X'cb'	X'01' X'4b'	
80	X'e0' X'47'	X'e0' X'c7'	X'01' X'47'	
81	X'e0' X'4f'	X'e0' X'cf'	X'01' X'4f'	
83	X'e0' X'48'	X'e0' X'c8'	X'01' X'48'	
84	X'e0' X'50'	X'e0' X'd0'	X'01' X'50'	
85	X'e0' X'49'	X'e0' X'c9'	X'01' X'49'	
86	X'e0' X'51'	X'e0' X'd1'	X'01' X'51'	
89	X'e0' X'4d'	X'e0' X'cd'	X'01' X'4d'	
	+ NUM LOCK ON			
75	X'e0' X'2a' X'e0' X'52'	X'e0' X'd2' X'e0' X'aa'	X'01' X'52'	
76	X'e0' X'2a' X'e0' X'53'	X'e0' X'd3' X'e0' X'aa'	X'01' X'53'	
79	X'e0' X'2a' X'e0' X'4b'	X'e0' X'cb' X'e0' X'aa'	X'01' X'4b'	
80	X'e0' X'2a' X'e0' X'47'	X'e0' X'c7' X'e0' X'aa'	X'01' X'47'	
81	X'e0' X'2a' X'e0' X'4f'	X'e0' X'cf' X'e0' X'aa'	X'01' X'4f'	
83	X'e0' X'2a' X'e0' X'48'	X'e0' X'c8' X'e0' X'aa'	X'01' X'48 '	
84	X'e0' X'2a' X'e0' X'50'	X'e0' X'd0' X'e0' X'aa'	X'01' X'50'	
85	X'e0' X'2a' X'e0' X'49'	X'e0' X'c9' X'e0' X'aa'	X'01' X'49'	
86	X'e0' X'2a' X'e0' X'51'	X'e0' X'd1' X'e0' X'aa'	X'01' X'51'	
89	X'e0' X'2a' X'e0' X'4d'	X'e0' X'cd' X'e0' X'aa'	X'01' X'4d'	
	Programmable Keys 201 - 232			
201	X'e0' X'0b'	X'e0' X'8b'	X'01' X'0b'	

Table 272. PS/2 or USB scan codes for CANPOS keyboard (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
202	X'e0' X'02'	X'e0' X'82'	X'01' X'02'	
203	X'e0' X'03'	X'e0' X'83'	X'01' X'03'	
204	X'e0' X'04'	X'e0' X'84'	X'01' X'04'	
205	X'e0' X'05'	X'e0' X'85'	X'01' X'05'	
206	X'e0' X'06'	X'e0' X'86'	X'01' X'06'	
207	X'e0' X'07'	X'e0' X'87'	X'01' X'07'	
208	X'e0' X'08'	X'e0' X'88'	X'01' X'08'	
209	X'e0' X'09'	X'e0' X'89'	X'01' X'09'	
210	X'e0' X'0a'	X'e0' X'8a'	X'01' X'0a'	
211	X'e0' X'1e'	X'e0' X'9e'	X'01' X'1e'	
212	X'e0' X'30'	X'e0' X'b0'	X'01' X'30'	
213	X'e0' X'2e'	X'e0' X'ae'	X'01' X'2e'	
214	X'e0' X'20'	X'e0' X'a0'	X'01' X'20'	
215	X'e0' X'12'	X'e0' X'92'	X'01' X'12'	
216	X'e0' X'21'	X'e0' X'a1'	X'01' X'21'	
217	X'e0' X'22'	X'e0' X'a2'	X'01' X'22'	
218	X'e0' X'23'	X'e0' X'a3'	X'01' X'23'	
219	X'e0' X'17'	X'e0' X'97'	X'01' X'17'	
220	X'e0' X'24'	X'e0' X'a4'	X'01' X'24'	
221	X'e0' X'25'	X'e0' X'a5'	X'01' X'25'	
222	X'e0' X'26'	X'e0' X'a6'	X'01' X'26'	
223	X'e0' X'32'	X'e0' X'b2'	X'01' X'32'	
224	X'e0' X'31'	X'e0' X'b1'	X'01' X'31'	
225	X'e0' X'18'	X'e0' X'98'	X'01' X'18'	
226	X'e0' X'19'	X'e0' X'99'	X'01' X'19'	
227	X'e0' X'10'	X'e0' X'90'	X'01' X'10'	
228	X'e0' X'13'	X'e0' X'93'	X'01' X'13'	
229	X'e0' X'1f'	X'e0' X'9f'	X'01' X'1f'	
230	X'e0' X'14'	X'e0' X'94'	X'01' X'14'	
231	X'e0' X'16'	X'e0' X'96'	X'01' X'16'	
232	X'e0' X'2f'	X'e0' X'af'	X'01' X'2f'	

Note: Linux: When connected as USB System, the Modular CANPOS Keyboard will return the following JavaPOS POS key data for the following keys:

Key 95 - X'53'
 Key 100 - X'65'
 Key 104 - X'78"
 Key 105 - X'7a"
 Key 106 - X'7e'

Point of Sale Keyboard V

Figure 57 shows the Keyboard V layout.

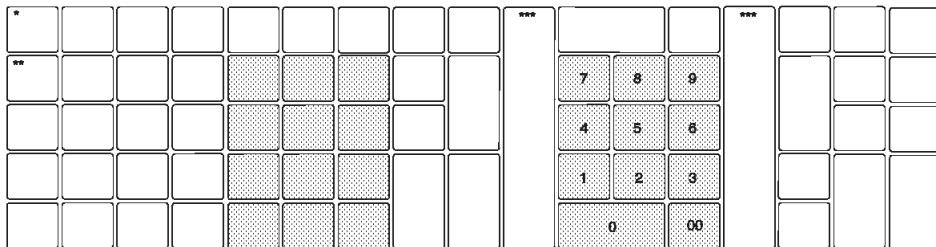


Figure 57. Point of Sale Keyboard V layout

Note:

1. The single asterisk (*) in the top-left portion of the keyboard indicates the S1 key.
2. The double asterisk (**) in the top-left portion of the keyboard indicates the S2 key.
3. The triple asterisk (***)) indicates that a 1x5 dummy cap covers those key switches.
4. The two possible locations for the numeric keypad are shown in the shaded area of the illustration. The default location for the numeric keypad is the right-most shaded area.

Keyboard V scan codes

Figure 58 shows the key scan codes for the Keyboard-V.

4C {00}	4F	4E	4D	49	48	47	4B	4A	(46)	45	{44}	43	(42)	41	40	50
3C {01}	3F	3E	3D	39	38	37	3B	(3A)	(36)	35	34	33	(32)	{31}	30	51
2C	2F	2E	2D	29	28	27	2B	2A	(26)	25	24	23	(22)	21	20	52
1C	1F	1E	1D	19	18	17	(1B)	(1A)	(16)	15	14	13	(12)	11	{10}	{53}
0C	0F	0E	0D	09	08	07	0B	0A	(06)	05	{04}	03	(02)	81	80	54

Figure 58. Keyboard-V scan code set

Note:

1. '()' indicates that these scan codes can be generated when the layout is changed.
2. '{ }' indicates these scan codes will be generated only when the keylock is in the **system** position.
3. Each double key produces only one scan code (the key scan code without parentheses in the illustration), unlike the single-byte character set (SBCS) keyboards.
4. Return only make scan codes.

PLU Keyboard and Display III

Figure 59 on page 345 shows the PLU Keyboard and Display-III layout and scan codes.

The two possible locations for the numeric keypad are shaded in Figure 59 on page 345. The default location for the numeric keypad is the right-most shaded area.

7F	7E	7D	79	78	77	7C	7B	7A	76	75	74	73	72	71	70
6F	6E	6D	69	68	67	6C	6B	6A	66	65	64	63	62	61	60
5F	5E	5D	59	58	57	5C	5B	5A	56	55	54	53	52	51	50
4F	4E	4D	49	48	47	4C	4B	4A	46	45	(44)	43	42	41	40
3F	3E	3D	39	38	37	3C	3B	3A	(36)	35	34	33	(32)	31	30
2F	2E	2D	29	28	27	2C	2B	2A	26	25	24	23	22	21	20
1F	1E	1D	19	18	17	1C	1B	(1A)	(16)	15	14	13	12	(11)	(10)
0F	0E	0D	09	08	07	0C	0B	0A	06	05	(04)	03	02	01	00

Figure 59. PLU Keyboard and Display-III layout

Note:

1. "()" indicates that these scan codes can be generated when the layout is changed.
2. Each double key produces only one scan code (the key scan code without parentheses in the illustration), unlike the single-byte character set keyboards.
3. Return only make scan codes.
4. Only double keys are shown in the figure.

4674 Point of Sale Keyboard (built-in)

Figure 60 shows the layout of the keyboard.

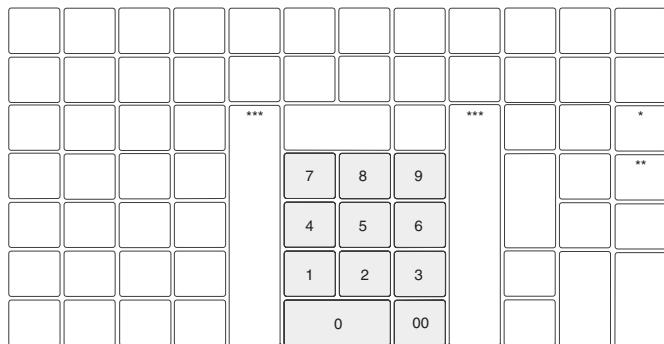


Figure 60. 4674 Point of Sale Keyboard (built-in)

Note:

1. A single asterisk (*) indicates the S1 key.
2. Double asterisks (**) indicate the S2 key.
3. Triple asterisks (***) indicates that a 1x5 dummy cap covers those key switches.

4674 POS Keyboard (built-in) scan codes

Figure 61 on page 346 provides the scan codes.

6Ah	69h	68h	67h	66h	65h	64h	63h	62h	6Fh	6Eh	6Dh
5Ah	59h	58h	57h	56h	55h	54h	53h	52h	5Fh	5Eh	5Dh
4Ah	49h	48h	47h	46h	45h	44h	43h	42h	4Fh	4Eh	4Dh (00)
3Ah	39h	38h	37h	36h	35h	34h	33h	32h	3Fh	3Eh	3Dh (01)
2Ah	29h	28h	27h	26h	25h	24h	23h	22h	2Fh	2Eh	2Dh
1Ah	19h	18h	17h	16h	15h	14h	13h	12h	1Fh	1Eh	1Dh
0Ah	09h	08h	07h	06h	05h	04h	03h	02h	0Fh	0Eh	0Dh

Figure 61. Scan codes for 4674 built-in keyboard

xxh Indicates scan codes that are generated in the default keyboard configuration

() Indicates scan codes that are generated only when the keylock is in the **System** position.

Note: Return only make scan codes.

4685 keyboards

This section describes all of the 4685 keyboards.

All 4685 keyboards return only make scan codes.

4685 Point of Sale Keyboard Model K01

Figure 62 shows the layout of the 4685 Keyboard Model K01.

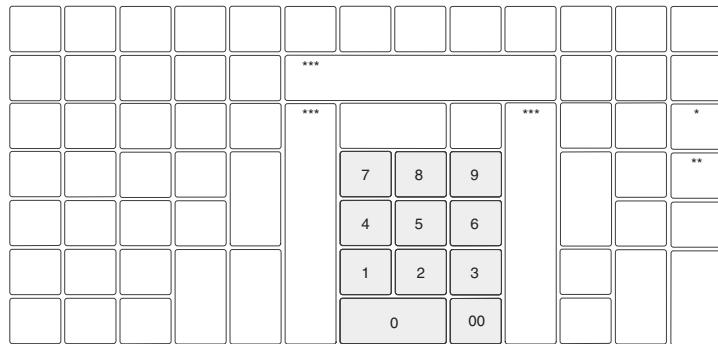


Figure 62. 4685 Point of Sale Keyboard Model K01 layout

Note:

1. The numeric keypad location is shaded in the illustration.
2. The single asterisk (*) in the middle-right portion of the keyboard indicates the S1 key.
3. The double asterisk (**) in the middle-right portion of the keyboard indicates the S2 key.
4. The triple asterisk (***)) indicates that a 1x5 dummy cap covers those key switches.

4685 Keyboard Model K01 scan code set

Figure 63 on page 347 shows the key scan codes for the 4685 Keyboard Model K01.

6B	6A	69	68	67	66	65	64	63	62	6F	6E	6D
5B	5A	59	58	57	*** 56	55	54	53	52	5F	5E	5D
4B	4A	49	48	47	*** 46	45	44	43	*** 42	4F	4E	* 4D (00)
3B	3A	39	38	37	36	35	34	33	32	3F	3E	** 3D (01)
2B	2A	29	28	27	26	25	24	23	22	2F	2E	2D
1B	1A	19	18	17	16	15	14	13	12	1F	1E	1D
0B	0A	09	08	07	06	05	04	03	02	0F	0E	0D

Figure 63. 4685 Keyboard Model K01 scan code set

Note:

1. "()" indicates these scan codes will be generated only when the keylock is in the "system" position.
2. Each double key produces only one scan code.
3. The single asterisk (*) in the middle-right portion of the keyboard indicates the S1 key.
4. The double asterisk (**) in the middle-right portion of the keyboard indicates the S2 key.
5. The triple asterisk (***)) indicates that a 1×5 dummy cap covers those key switches.

4685 Keyboard K02 Ultra 7 (four-position keylock)

Figure 64 shows the layout of the keyboard.

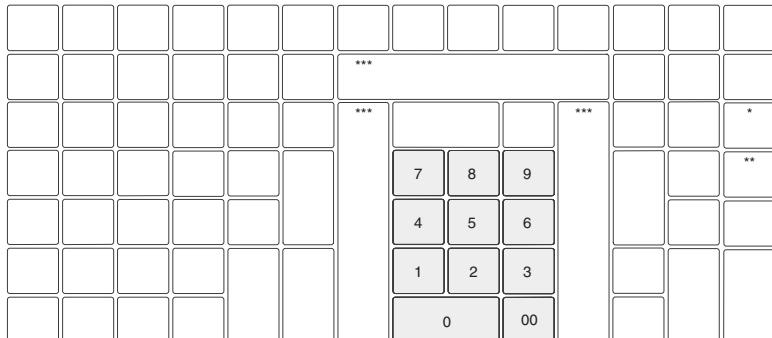


Figure 64. 4685 Keyboard K02 Ultra 7 with 4-position keylock

Note:

1. A single asterisk (*) indicates the S1 key.
2. Double asterisks (**) indicate the S2 key.
3. Triple asterisks (***)) indicate dummy keys.

4685 Keyboard K02 Ultra 7 scan codes

Figure 65 on page 348 shows the scan codes for the 4685 Keyboard K02 Ultra 7.

6Ch	6Bh	6Ah	69h	68h	67h	66h	65h	64h	63h	62h	6Fh	6Eh	6Dh
5Ch	5Bh	5Ah	59h	58h	57h	56h	55h	54h	53h	52h	5Fh	5Eh	5Dh
4Ch	4Bh	4Ah	49h	48h	47h	46h	45h	44h	43h	42h	4Fh	4Eh	4Dh (00)
3Ch	3Bh	3Ah	39h	38h	<37h>	36h	35h	34h	33h	32h	3Fh	3Eh	3Dh (01)
2Ch	2Bh	2Ah	29h	28h	27h	26h	25h	24h	23h	22h	2Fh	2Eh	2Dh
1Ch	1Bh	1Ah	19h	<18h>	<17h>	16h	15h	14h	13h	12h	1Fh	1Eh	1Dh
0Ch	0Bh	0Ah	09h	08h	07h	06h	05h	04h	03h	02h	0Fh	0Eh	0Dh

Figure 65. 4685 Keyboard K02 Ultra 7 scan codes

Note:

1. < > indicates these are dummy keys at default
2. () indicates scan codes that are generated only when the keylock is in the *System* position.

4685 Keyboard Model K02 Ultra 7 with MSR/E (four or six position keylock)

Figure 66 shows the layout of the keyboard.

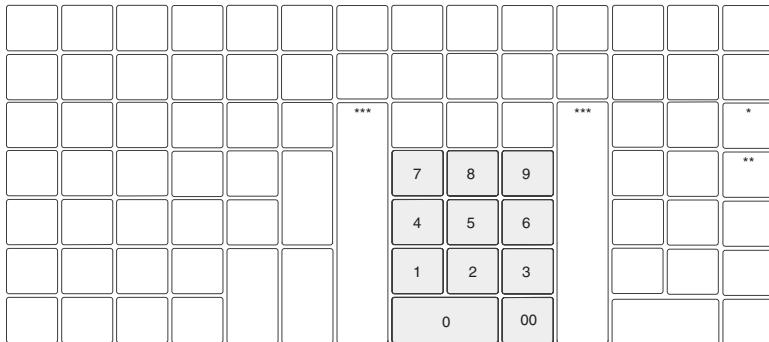


Figure 66. 4685 Keyboard K02 Ultra 7 with MSR/E (4 or 6 position keylock)

Note:

1. A single asterisk (*) indicates the S1 key.
2. Double asterisks (**) indicate the S2 key.
3. Triple asterisks (***)) indicate dummy keys.

4685 Keyboard K02 Ultra 7 with MSR/E scan codes

Figure 67 on page 349 provides the scan codes.

6Ch	6Bh	6Ah	69h	68h	67h	66h	65h	64h	63h	62h	6Fh	6Eh	6Dh
5Ch	5Bh	5Ah	59h	58h	57h	56h	55h	54h	53h	52h	5Fh	5Eh	5Dh
4Ch	4Bh	4Ah	49h	48h	47h	46h	45h	44h	43h	42h	4Fh	4Eh	4Dh (00)
3Ch	3Bh	3Ah	39h	38h	37h	36h	35h	34h	33h	32h	3Fh	3Eh	3Dh (01)
2Ch	2Bh	2Ah	29h	28h	27h	26h	25h	24h	23h	22h	2Fh	2Eh	2Dh
1Ch	1Bh	1Ah	19h	18h	17h	16h	15h	14h	13h	12h	1Fh	1Eh	1Dh
0Ch	0Bh	0Ah	09h	08h	07h	06h	05h	04h	03h	02h	0Fh	0Eh	0Dh

Figure 67. 4685 Keyboard K02 Ultra 7 with MSR/E scan codes

xxh Indicates scan codes that are generated in the default keyboard configuration

() Indicates scan codes that are generated only when the keylock is in the **System** position.

4685 Keyboard Model KC1

Figure 68 shows the keyboard layout.

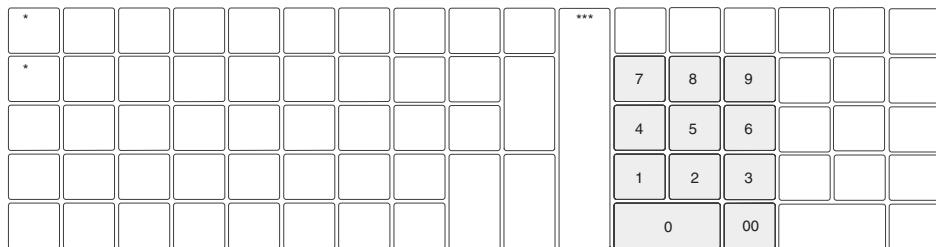


Figure 68. 4685 Model KC1

Note:

1. A single asterisk (*) indicates the S1 key.
2. Double asterisks (**) indicate the S2 key.
3. Triple asterisks (***) indicate dummy keys.

4685 Keyboard Model KC1 scan codes

Figure 69 provides the scan codes.

4Ch (00)	4Fh	4Eh	4Dh	49h	48h	47h	4Bh	4Ah	46h	45h	44h	43h	42h	41h	40h	50h
3Ch (00)	3Fh	3Eh	3Dh	39h	38h	37h	3Bh	3Ah	36h	35h	34h	33h	32h	31h	30h	51h
2Ch	2Fh	2Eh	2Dh	29h	28h	27h	2Bh	2Ah	26h	25h	24h	23h	22h	21h	20h	52h
1Ch	1Fh	1Eh	1Dh	19h	18h	17h	1Bh	1Ah	16h	15h	14h	13h	12h	11h	10h	53h
0Ch	0Fh	0Eh	0Dh	09h	08h	07h	0Bh	0Ah	06h	05h	04h	03h	02h	81h	80h	54h

Figure 69. Scan codes for Model 4685-KC1

xxh Indicates scan codes that are generated in the default keyboard configuration

() Indicates scan codes that are generated only when the keylock is in the **System** position.

4685 Keyboard Model K03

Figure 70 shows the layout of the keyboard.

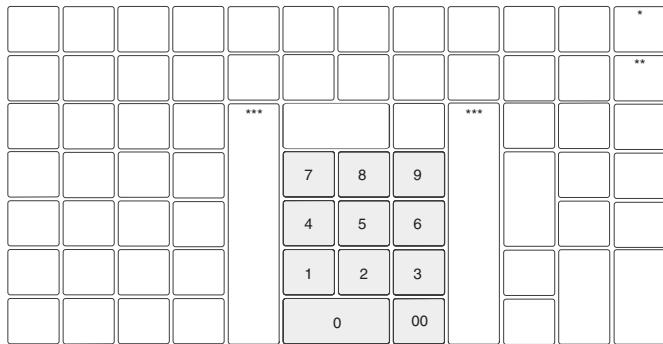


Figure 70. 4685 Keyboard Model K03 layout

Note:

1. A single asterisk (*) indicates the S1 key.
2. Double asterisks (**) indicate the S2 key.
3. Triple asterisks (***) indicate dummy keys.

4685 Keyboard Model K03 scan codes

Figure 71 provides the scan codes.

0x6A	0x69	0x68	0x67	0x66	0x65	0x64	0x63	0x62	0x6F	0x6E	0x6D (0x00)
0x5A	0x59	0x58	0x57	0x56	0x55	0x54	0x53	0x52	0x5F	0x5E	0x5D (0x01)
0x4A	0x49	0x48	0x47	<0x46>	0x45	<0x44>	0x43	<0x42>	0x4F	0x4E	0x4D
0x3A	0x39	0x38	0x37	<0x36>	0x35	0x34	0x33	<0x32>	<0x3F>	0x3E	0x3D
0x2A	0x29	0x28	0x27	<0x26>	0x25	0x24	0x23	<0x22>	0x2F	0x2E	0x2D
0x1A	0x19	0x18	0x17	<0x16>	0x15	0x14	0x13	<0x12>	<0x1F>	<0x1E>	<0x1D>
0x0A	0x09	0x08	0x07	<0x06>	0x05	0x04	0x03	<0x02>	0x0F	0x0E	0x0D

Figure 71. 4685 Keyboard Model K03 scan codes

Note:

1. <> indicates these are dummy keys at default.
2. () indicates scan codes that are generated only when the keylock is in the *System* position.
3. RS-485 connectivity returns only make scan codes.

4820 Toshiba SurePoint Solution Keypad

Figure 72 on page 351 shows the layout of the key-switch numbers for the 4820 Keypad.

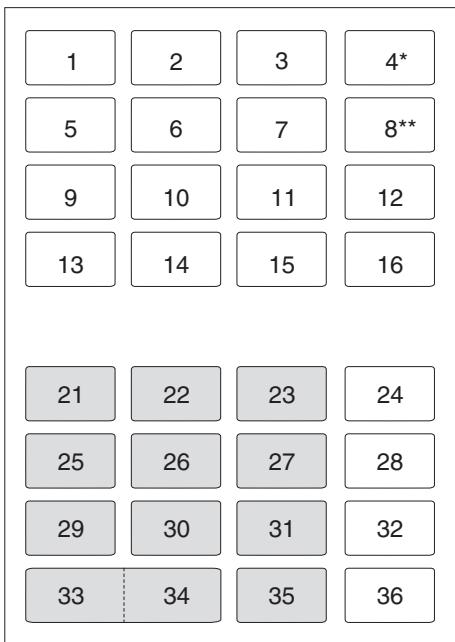


Figure 72. 4820 Keypad key-switch layout

Note:

1. A single asterisk (*) indicates the S1 key.
2. Double asterisks (**) indicate the S2 key.

Table 273 relates the keyboard key-switch number to the scan codes received when the keyboard is attached to the RS-485 or USB port.

Table 273. 4820 RS-485/USB Toshiba SurePoint Solution keypad scan codes

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
1	X'4b'	X'f0' X'4b'	X'4b'	
2	X'3b'	X'f0' X'3b'	X'3b'	
3	X'6b'	X'f0' X'6b'	X'6b'	
4	X'8b'	X'f0' X'8b'	X'8b'	This is the S1 key
5	X'4c'	X'f0' X'4c'	X'4c'	
6	X'3c'	X'f0' X'3c'	X'3c'	
7	X'6c'	X'f0' X'6c'	X'6c'	
8	X'8c'	X'f0' X'8c'	X'8c'	This is the S2 key.
9	X'4f'	X'f0' X'4f'	X'4f'	
10	X'3f'	X'f0' X'3f'	X'3f'	
11	X'6f'	X'f0' X'6f'	X'6f'	
12	X'8f'	X'f0' X'8f'	X'8f'	
13	X'4e'	X'f0' X'4e'	X'4e'	
14	X'3e'	X'f0' X'3e'	X'3e'	
15	X'6e'	X'f0' X'6e'	X'6e'	
16	X'8e'	X'f0' X'8e'	X'8e'	

Table 273. 4820 RS-485/USB Toshiba SurePoint Solution keypad scan codes (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
21	X'7c'	X'f0' X'7c'	X'7c'	
22	X'0c'	X'f0' X'0c'	X'0c'	
23	X'1c'	X'f0' X'1c'	X'1c'	
24	X'50'	X'f0' X'50'	X'50'	
25	X'7f'	X'f0' X'7f'	X'7f'	
26	X'0f'	X'f0' X'0f'	X'0f'	
27	X'1f'	X'f0' X'1f'	X'1f'	
28	X'9f'	X'f0' X'9f'	X'9f'	
29	X'7e'	X'f0' X'7e'	X'7e'	
30	X'0e'	X'f0' X'0e'	X'0e'	
31	X'1e'	X'f0' X'1e'	X'1e'	
32	X'9e'	X'f0' X'9e'	X'9e'	
33	X'7d'	X'f0' X'7d'	X'7d'	
34	X'0d'	X'f0' X'0d'	X'0d'	
35	X'1d'	X'f0' X'1d'	X'1d'	
36	X'9d'	X'f0' X'9d'	X'9d'	
+ Ctrl Case Pressed				
4	X'f0' X'50' X'00' X'50'		X'00'	Key press only
8	X'f0' X'50' X'01' X'50'		X'01'	Key press only

4820 Toshiba SurePoint Solution Keypad (system-attached)

Figure 73 on page 353 shows the layout of the key-switch numbers for the 4820 Keypad.

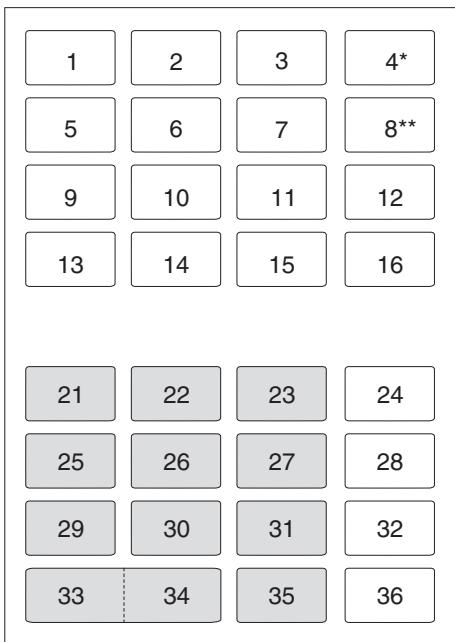


Figure 73. 4820 Keypad (system-attached) key-switch layout

Table 274 relates the keyboard key-switch number to the scan codes received when the keyboard is attached to the system keyboard port.

Table 274. 4820 Toshiba SurePoint Solution Keypad (system-attached) scan codes

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData
1	X'3b'	X'bb'	X'3b'
2	X'3c'	X'bc'	X'3c'
3	X'3d'	X'bd'	X'3d'
4	X'3e'	X'be'	X'3e'
5	X'3f'	X'bf'	X'3f'
6	X'40'	X'c0'	X'40'
7	X'41'	X'c1'	X'41'
8	X'42'	X'c2'	X'42'
9	X'43'	X'c3'	X'43'
10	X'44'	X'c4'	X'44'
11	X'57'	X'd7'	X'57'
12	X'58'	X'd8'	X'58'
13	X'01'	X'81'	X'01'
14	X'4e'	X'ce'	X'4e'
15	X'4a'	X'ca'	X'4a'
16	X'0e'	X'8e'	X'0e'
21	X'47'	X'c7'	X'47'
22	X'48'	X'c8'	X'48'
23	X'49'	X'c9'	X'49'
24	X'1d'	X'9d'	X'1d'

Table 274. 4820 Toshiba SurePoint Solution Keypad (system-attached) scan codes (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData
25	X'4b'	X'cb'	X'4b'
26	X'4c'	X'cc'	X'4c'
27	X'4d'	X'cd'	X'4d'
28	X'38'	X'b8'	X'38'
29	X'4f'	X'cf'	X'4f'
30	X'50'	X'd0'	X'50'
31	X'51'	X'd1'	X'51'
32	X'39'	X'b9'	X'39'
33	X'52'	X'd2'	X'52'
34	X'e0' X'35'	X'e0' X'b5'	X'01' X'35'
35	X'37'	X'b7'	X'37'
36	X'e0' X'1c'	X'e0' X'9c'	X'01' X'1c'

SureOne model 4614/4615 and SurePOS 100 model 4613 keyboard (built-in)

SureOne supports only the 101-key layout.

SurePOS 100 supports 101-, 102-, and 103-key layouts.

Note:

- When the keyboard is configured to be in POS mode, the application program must use standard PC Keyboard interfaces to intercept the keystrokes (scan codes) and perform the appropriate translation and detection.
- JavaPOS does not support POS mode.

Figure 74 on page 355 shows the layout of the key-switch numbers for the SureOne/SurePOS 100 keyboards.

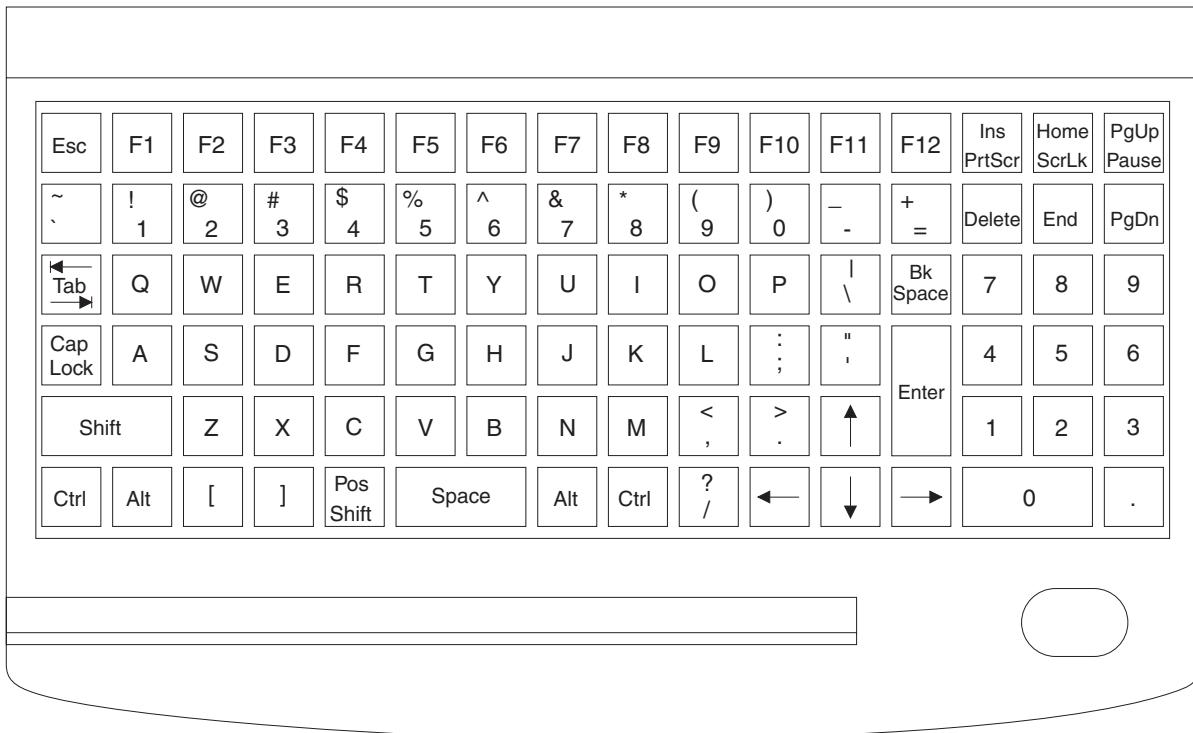


Figure 74. SureOne/SurePOS 100 keyboard (built-in) keyboard layout

SureOne model 4614/4615 and SurePOS 100 model 4613 keyboard (built-in) scan codes

Figure 75 on page 356 provides the key switch numbers for the SureOne/SurePOS 100 101-key keyboard.

Key Switch Number																
110	112	113	114	115	116	117	118	119	120	121	122	123	75	80	85	
1	2	3	4	5	6	7	8	9	10	11	12	13	76	81	86	
16	17	18	19	20	21	22	23	24	25	26	29	15	91	96	101	
30	31	32	33	34	35	36	37	38	39	40	41	43	92	97	102	
44		46	47	48	49	50	51	52	53	54	83		93	98	103	
58	60	27	28	(*1)	61		62	64	55	79	84	89	99	94	104	

Figure 75. SureOne/SurePOS 100 101-key keyboard switches

Figure 76 on page 357 provides the key switch numbers for the SurePOS 100 102-key keyboard.

Key Switch Number

110	112	113	114	115	116	117	118	119	120	121	122	123	75	80	85
1	2	3	4	5	6	7	8	9	10	11	12	13	76	81	86
16	17	18	19	20	21	22	23	24	25	26	42	15	91	96	101
30	31	32	33	34	35	36	37	38	39	40	41	43	92	97	102
44	45	46	47	48	49	50	51	52	53	54	83		93	98	103
58	60	27	28	(*1)		61	62	64	55	79	84	89	99	94	104

Figure 76. SurePOS 100 102-key keyboard switches

Figure 77 on page 358 provides the key switch numbers for the SurePOS 100 103-key keyboard.

Key Switch Number

110	112	113	114	115	116	117	118	119	120	121	122	123	75	80	85
1	2	3	4	5	6	7	8	9	10	11	12	13	76	81	86
16	17	18	19	20	21	22	23	24	25	26	42	15	91	96	101
30	31	32	33	34	35	36	37	38	39	40	41	43	92	97	102
44	45	46	47	48	49	50	51	52	53	54	83		93	98	103
58	60	27	28	(*1)	203	61	62	64	55	79	84	89	99	94	104

Figure 77. SurePOS 100 103-key keyboard switches

Table 275 shows the key scan codes for the SureOne and SurePOS 100 keyboards.

Table 275. SureOne/SurePOS 100 keyboard scan codes

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
1	0x29	0xA9	0x29	
2	0x02	0x82	0x02	
3	0x03	0x83	0x03	
4	0x04	0x84	0x04	
5	0x05	0x85	0x05	
6	0x06	0x86	0x06	
7	0x07	0x87	0x07	
8	0x08	0x88	0x08	
9	0x09	0x89	0x09	
10	0x0a	0x8a	0x0a	
11	0x0b	0x8b	0x0b	
12	0x0c	0x8c	0x0c	
13	0x0d	0x8d	0x0d	
15	0x0e	0x8e	0x0e	
16	0x0f	0x8f	0x0f	

Table 275. SureOne/SurePOS 100 keyboard scan codes (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
17	0x10	0x90	0x10	
18	0x11	0x91	0x11	
19	0x12	0x92	0x12	
20	0x13	0x93	0x13	
21	0x14	0x94	0x14	
22	0x15	0x95	0x15	
23	0x16	0x96	0x16	
24	0x17	0x97	0x17	
25	0x18	0x98	0x18	
26	0x19	0x99	0x19	
27	0x1a	0x9a	0x1a	
28	0x1b	0x9b	0x1b	
29	0x2b	0xab	0x2b	
30	0x3a	0xba	0x3a	
31	0x1e	0x9e	0x1e	
32	0x1f	0x9f	0x1f	
33	0x20	0xa0	0x20	
34	0x21	0xa1	0x21	
35	0x22	0xa2	0x22	
36	0x23	0xa3	0x23	
37	0x24	0xa4	0x24	
38	0x25	0xa5	0x25	
39	0x26	0xa6	0x26	
40	0x27	0xa7	0x27	
41	0x28	0xa8	0x28	
43	0x1c	0x9c	0x1c	
44	0x2a	0xaa	0x2a	
46	0x2c	0xac	0x2c	
47	0x2d	0xad	0x2d	
48	0x2e	0xae	0x2e	
49	0x2f	0xaf	0x2f	
50	0x30	0xb0	0x30	
51	0x31	0xb1	0x31	
52	0x32	0xb2	0x32	
53	0x33	0xb3	0x33	
54	0x34	0xb4	0x34	
55	0x35	0xb5	0x35	
58	0x1d	0x9d	0x1d	
60	0x38	0xb8	0x38	

Table 275. SureOne/SurePOS 100 keyboard scan codes (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
61	0x39	0xb9	0x39	
62	0xe0 0x38	0xe0 0xb8	0x01 0x38	
64	0xe0 0x1d	0xe0 0x9d	0x01 0x1d	
75	0xe0 0x52	0xe0 0xd2	0x01 0x52	
76	0xe0 0x53	0xe0 0xd3	0x 010x53	
79	0xe0 0x4b	0xe0 0xcb	0x01 0x4b	
80	0xe0 0x47	0xe0 0xc7	0x01 0x47	
81	0xe0 0x4f	0xe0 0xcf	0x01 0x4f	
83	0xe0 0x48	0xe0 0xc8	0x01 0x48	
84	0xe0 0x50	0xe0 0xd0	0x01 0x50	
85	0xe0 0x49	0xe0 0xc9	0x01 0x49	
86	0xe0 0x51	0xe0 0xd1	0x01 0x51	
89	0xe0 0x4d	0xe0 0xcd	0x01 0x4d	
91	0x47	0xc7	0x47	
92	0x4b	0xcb	0x4b	
93	0x4f	0xcf	0x4f	
94	0x52	0xd2	0x52	Double "00" disabled in BIOS
94	0x52 0x52	0xd2	0x52	SureOne: Double'00' enabled in BIOS
94	0x0b 0x8b 0x0b 0x8b		0x0b(down) 0x0b(up) 0x0b(down) 0x0b(up)	SurePOS 100: Double'00' enabled in BIOS
96	0x48	0xc8	0x48	
97	0x4c	0xcc	0x4c	
98	0x50	0xd0	0x50	
99	0x52	0xd2	0x52	
101	0x49	0xc9	0x49	
102	0x4d	0xcd	0x4d	
103	0x51	0xd1	0x51	
104	0x53	0xd3	0x53	
110	0x01	0x81	0x01	
112	0x3b	0xbb	0x3b	
113	0x3c	0xbc	0x3c	
114	0x3d	0xbd	0x3d	
115	0x3e	0xbe	0x3e	
116	0x3f	0xbf	0x3f	
117	0x40	0xc0	0x40	
118	0x41	0xc1	0x41	
119	0x42	0xc2	0x42	

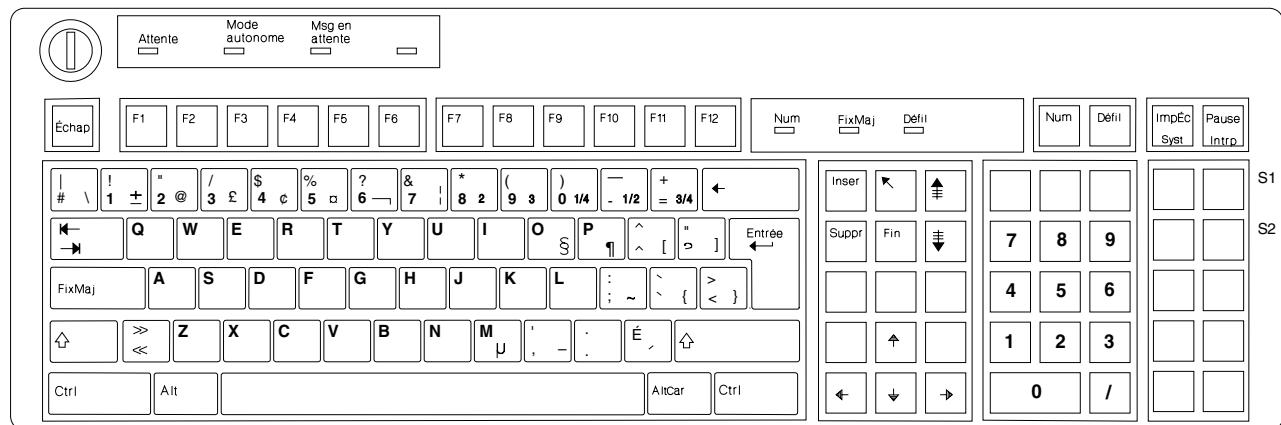
Table 275. SureOne/SurePOS 100 keyboard scan codes (continued)

Key switch number	Hardware make code	Hardware break code	JavaPOS - POSKeyData	Comments
120	0x43	0xc3	0x43	
121	0x44	0xc4	0x44	
122	0x57	0xd7	0x57	
123	0x58	0xd8	0x58	
Pos Shift + Ins		0xe0 0xb7 0xe0 0xaa	0x0137	Key release only
Pos Shift + Home	0x46	0xc6	0x46	
Pos Shift + PgUp	0xe1 0x1d 0x45 0xe1 0x9d 0xc5		0x0145	Key press only
Pos Shift + Esc	0xe1 0x1d 0x45 0xe1 0x9d 0xc5		0x0145	Key press only (SurePOS 100 only)
	+ Ins = (136-PrtScr)			
(*1) Pos Shift		0xe0 0xb7 0xe0 0xaa	0x0137	Key release only
	+ Home = (137-ScrLk)			
(*1)	0x46	0xc6	0x46	
	+ PgUp = (138-pause)			
(*1)	0xe1 0x1d 0x45 0xe1 0x9d 0xc5		0x0145	Key press only

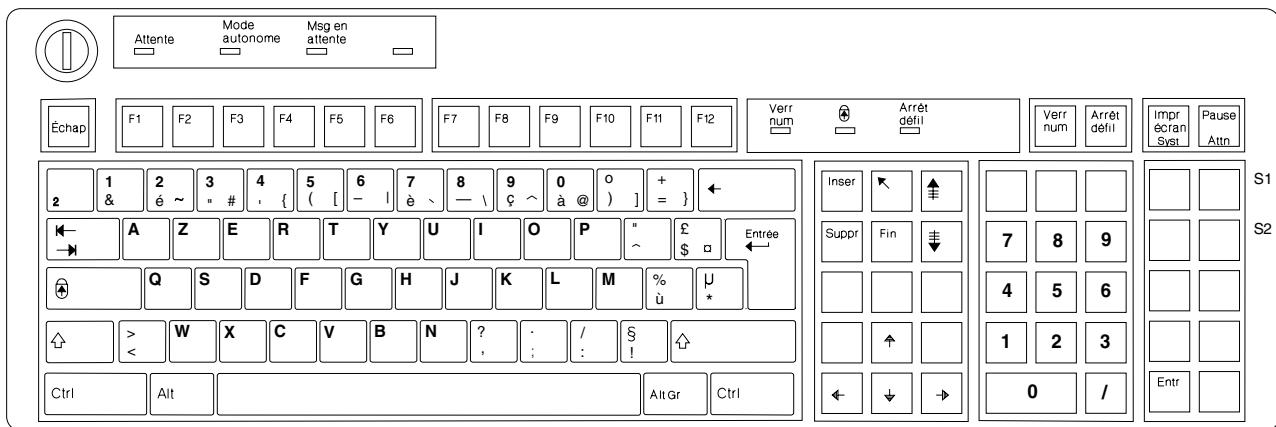
Alphanumeric Point-of-Sale (NANPOS) country-dependent keyboards

The following pages illustrate the keyboards for all the supported NANPOS keyboards.

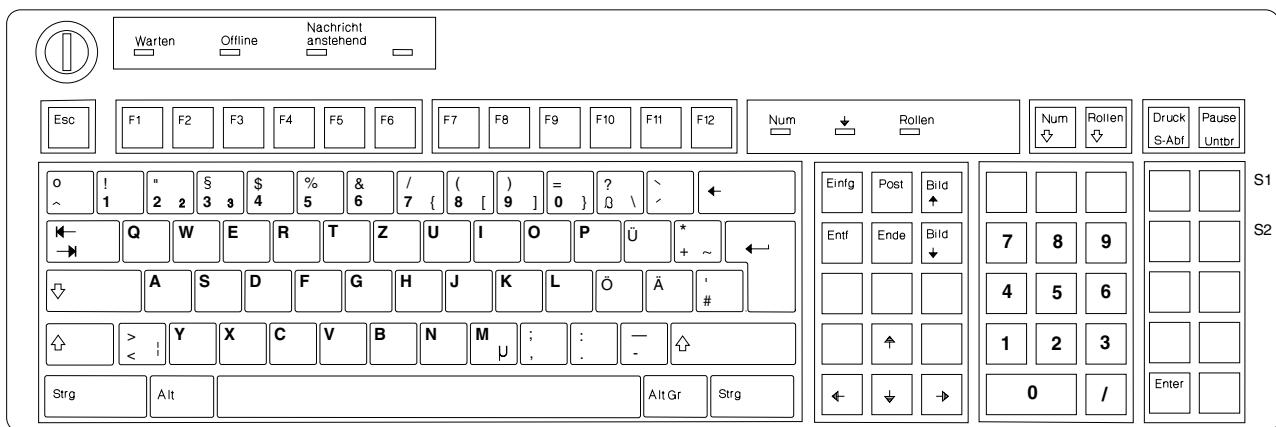
Canadian French keyboard layout



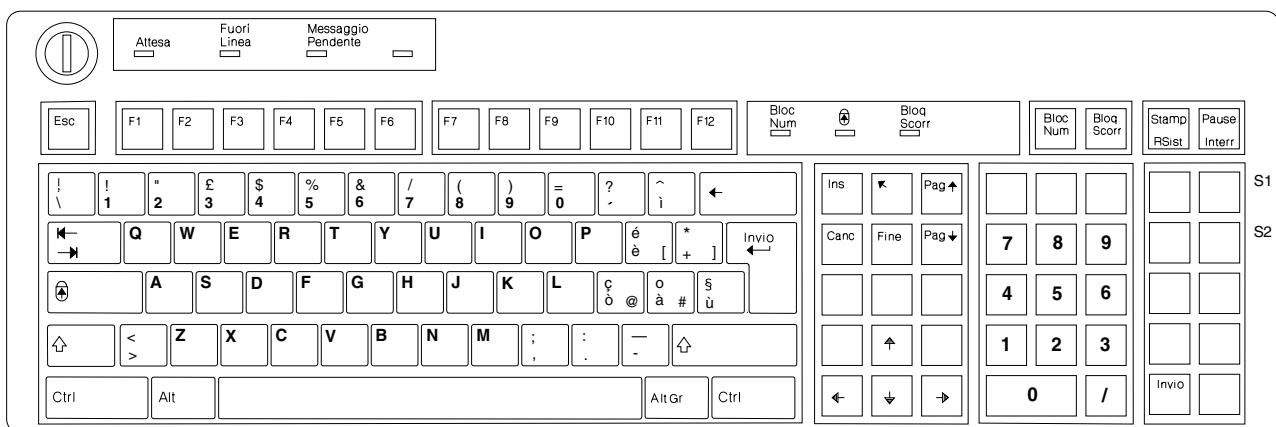
French keyboard layout



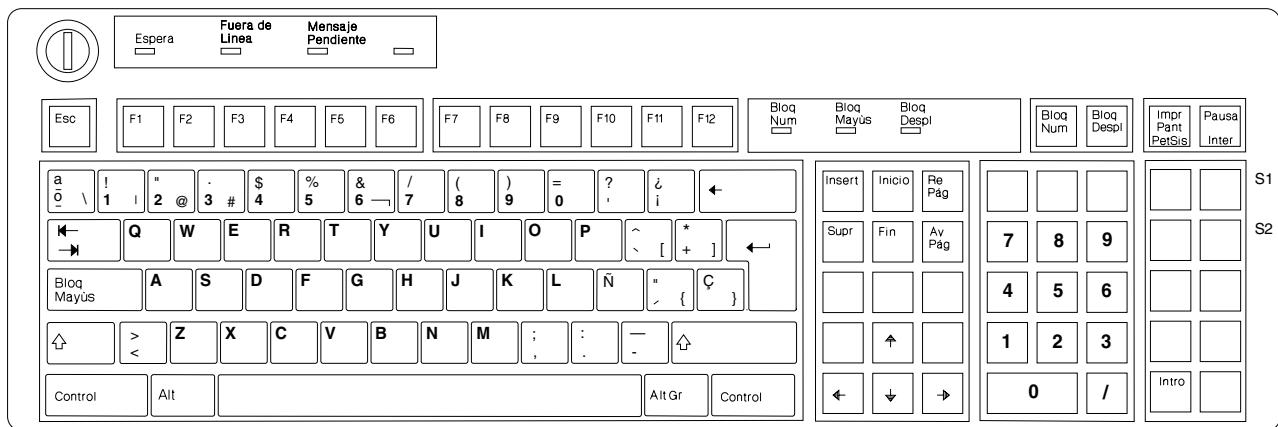
German keyboard layout



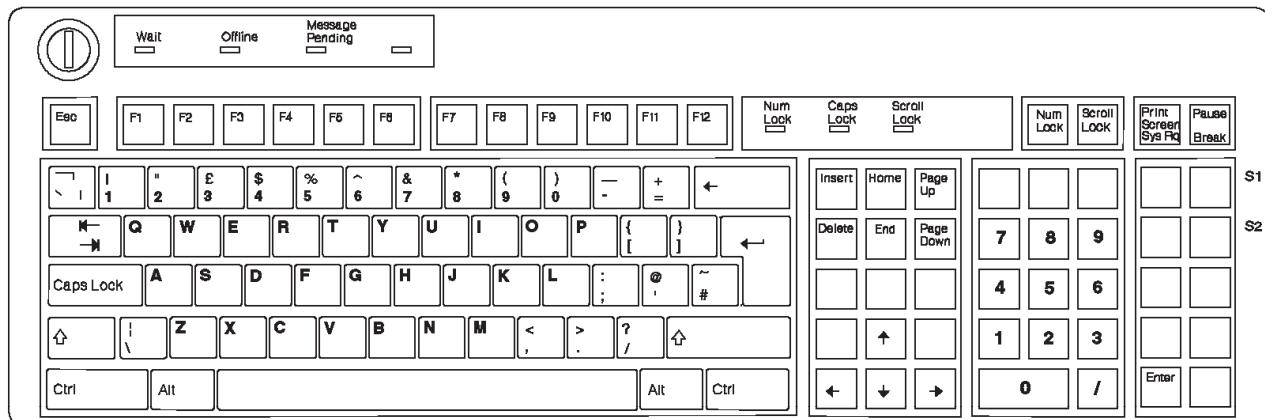
Italian keyboard layout



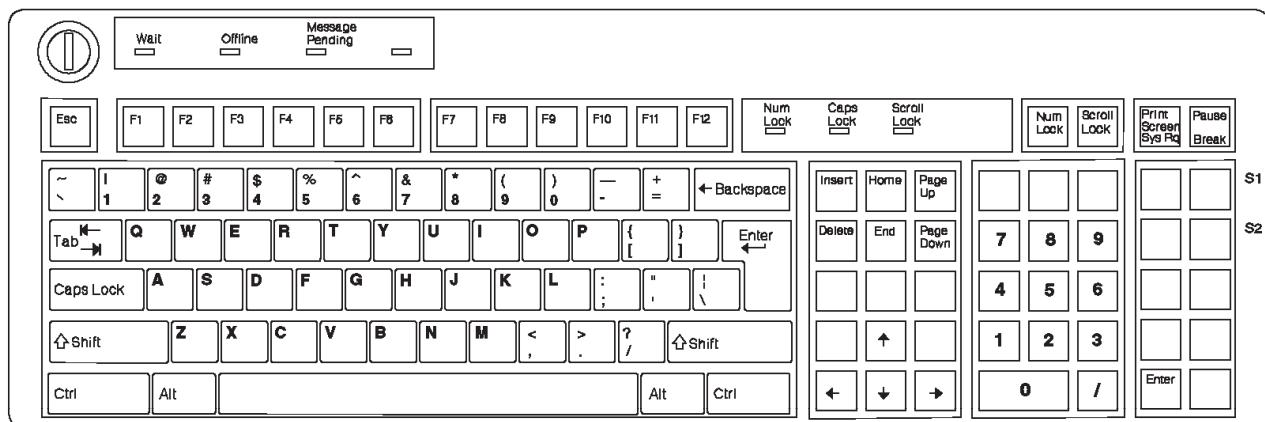
Spanish keyboard layout



U.K. English keyboard layout



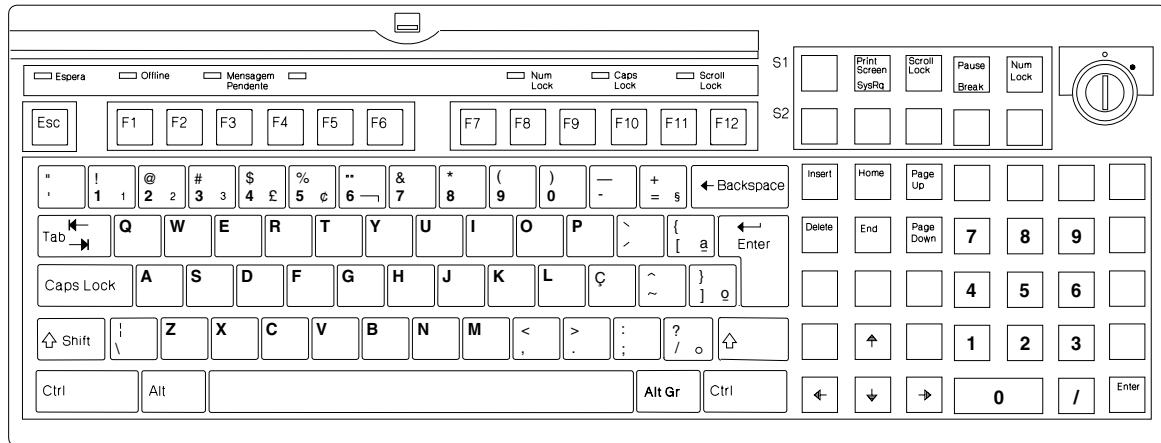
U.S. English keyboard layout



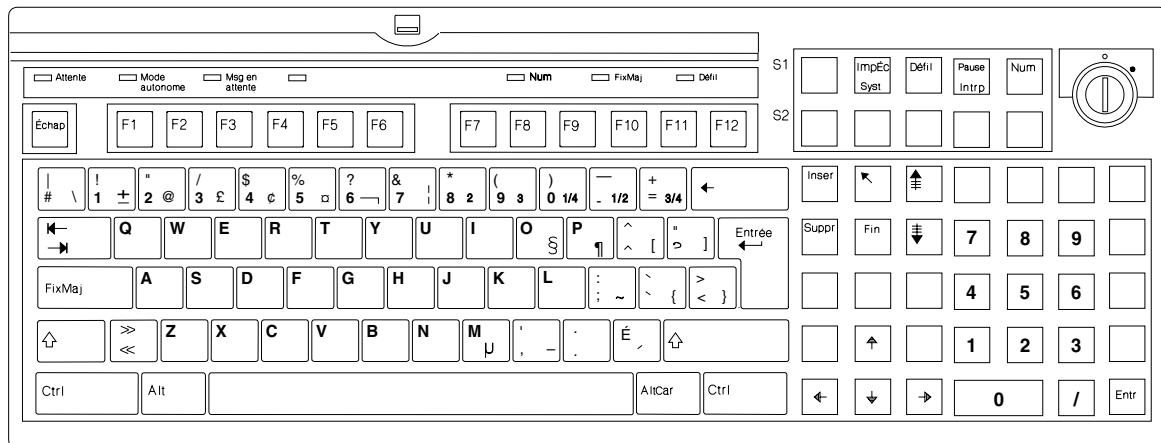
Retail Alphanumeric Point of Sale country dependent keyboards

The following pages illustrate the keyboards for all the supported Retail Alphanumeric Point of Sale Keyboards.

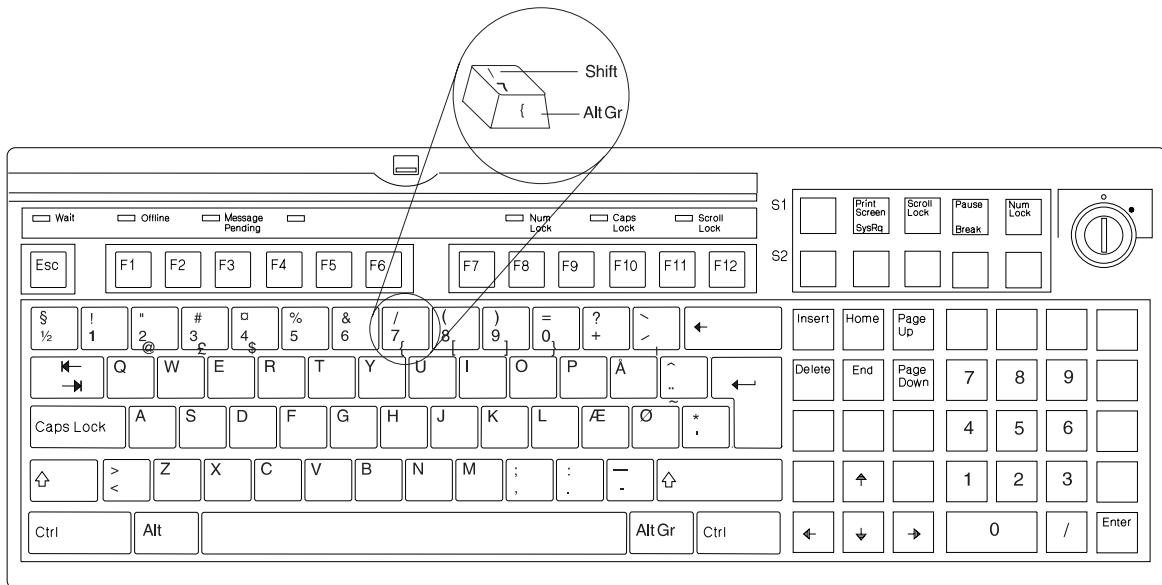
Brazil-Portuguese keyboard layout



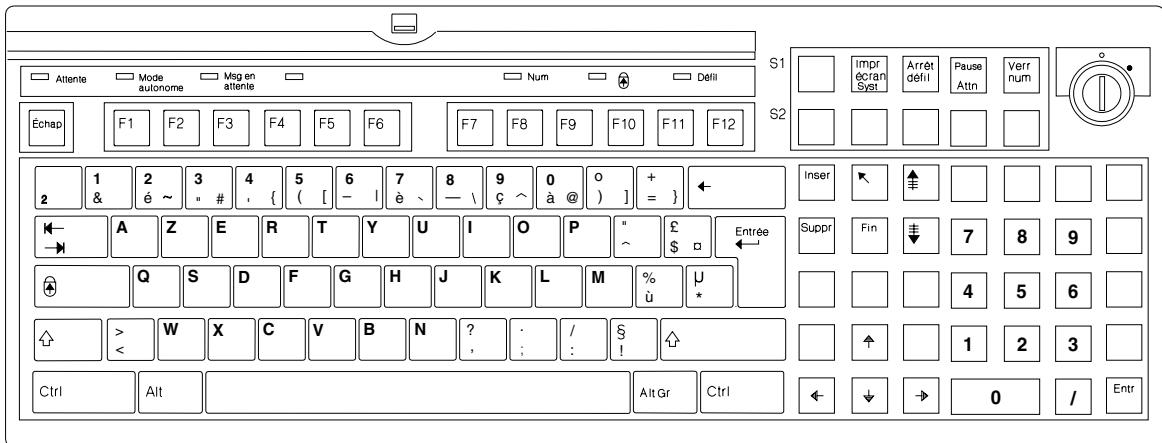
Canadian French keyboard layout



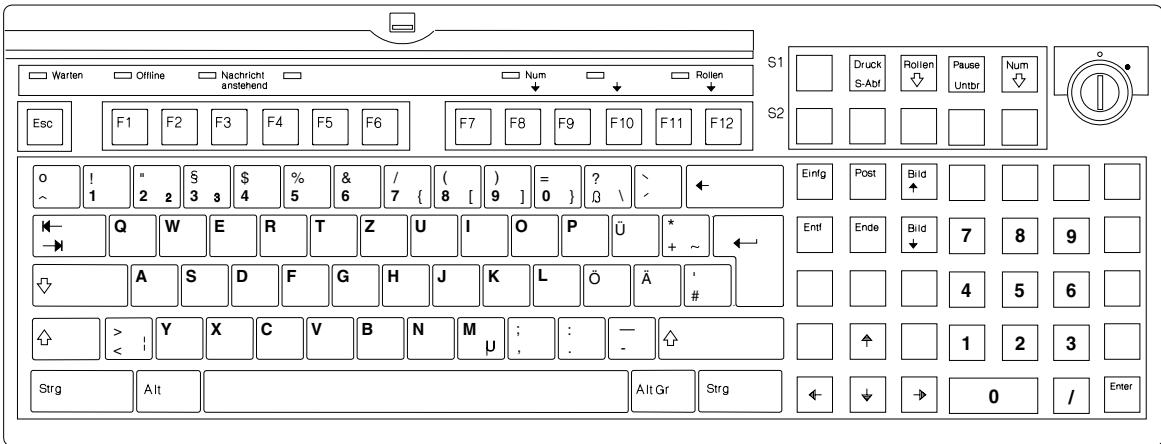
Danish keyboard layout



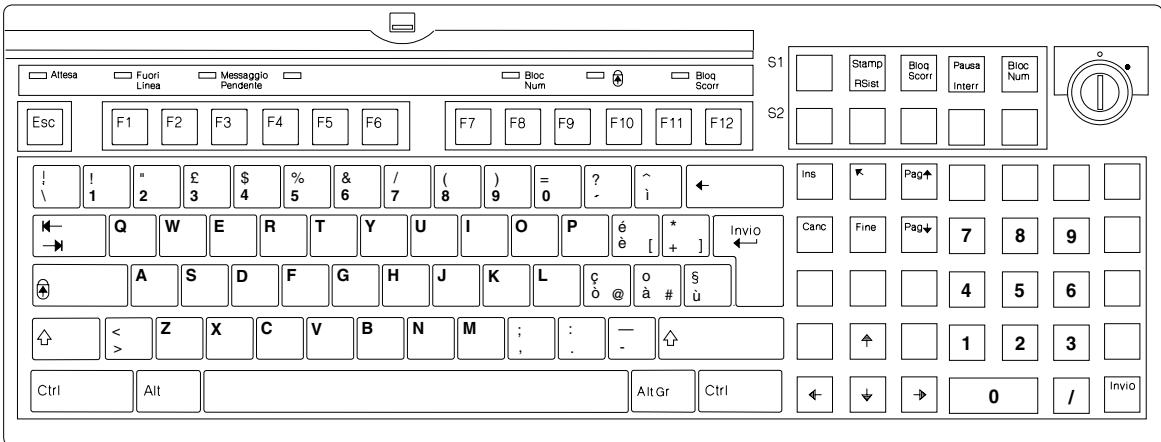
French keyboard layout



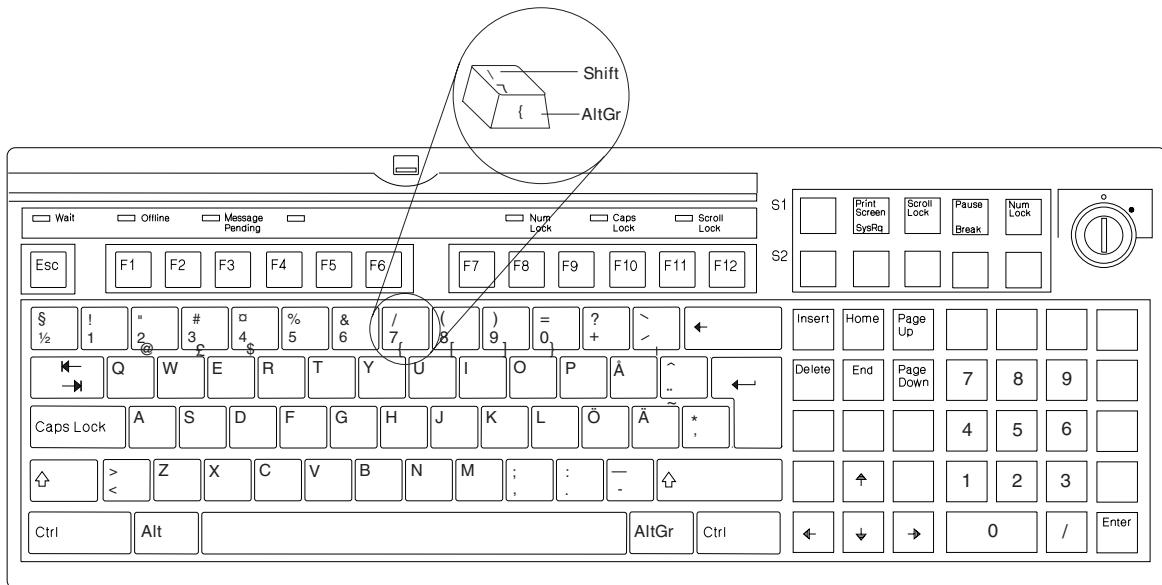
German keyboard layout



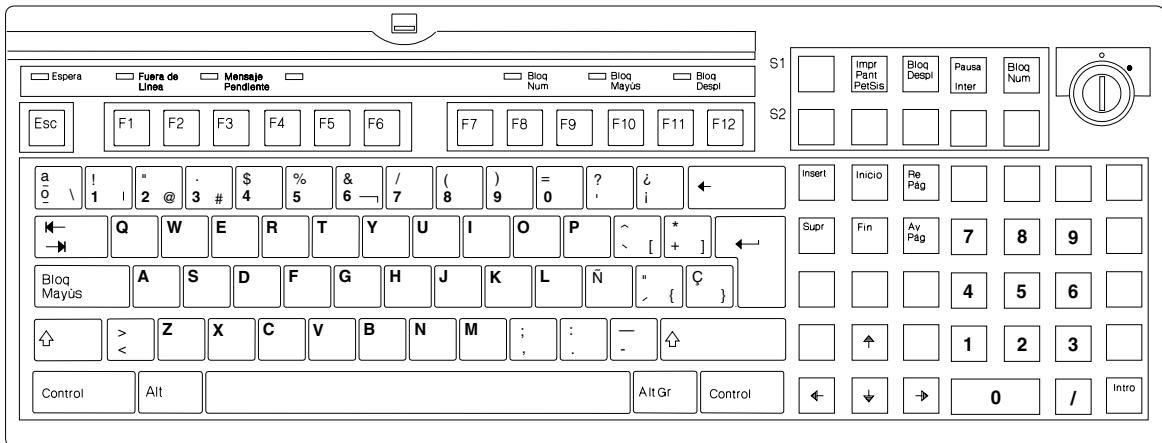
Italian keyboard layout



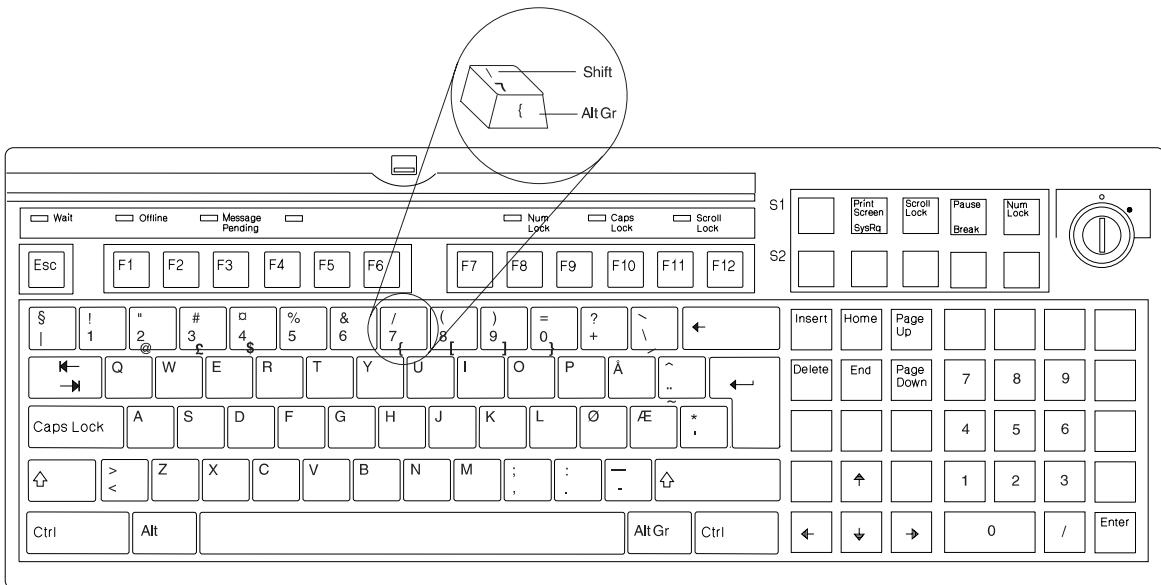
Norwegian keyboard layout



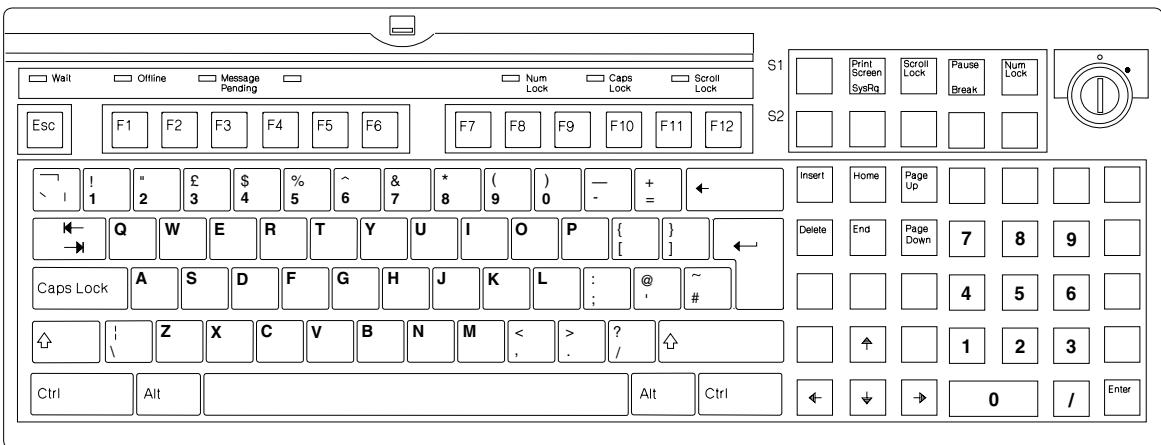
Spanish keyboard layout



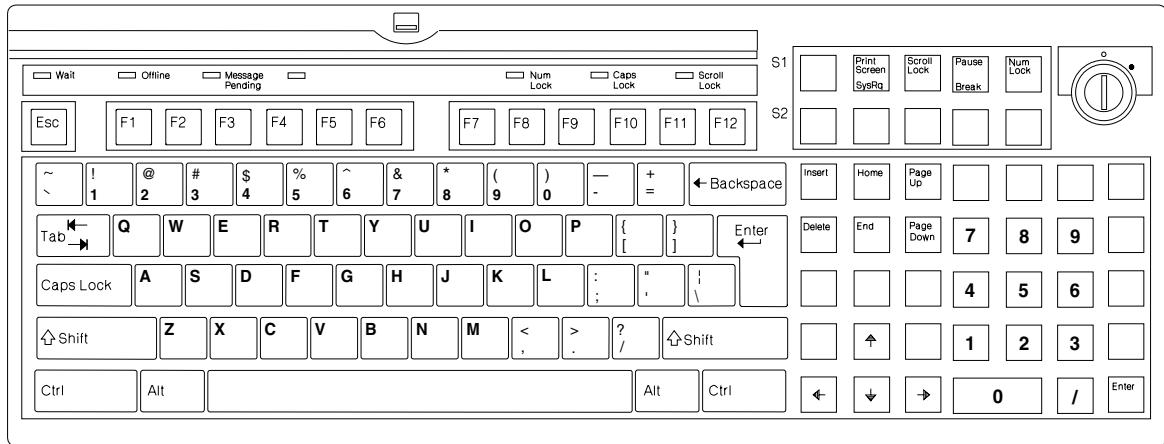
Swedish and Finnish keyboard layout



U.K. English keyboard layout



U.S. English keyboard layout



Compact Alphanumeric Point-of-Sale (CANPOS) keyboard

This section describes the layout and assigned key-switch numbers for the CANPOS keyboards

CANPOS keyboards by country

This section describes the keyboard layout for the Compact Alphanumeric Point of Sale (CANPOS) keyboard.

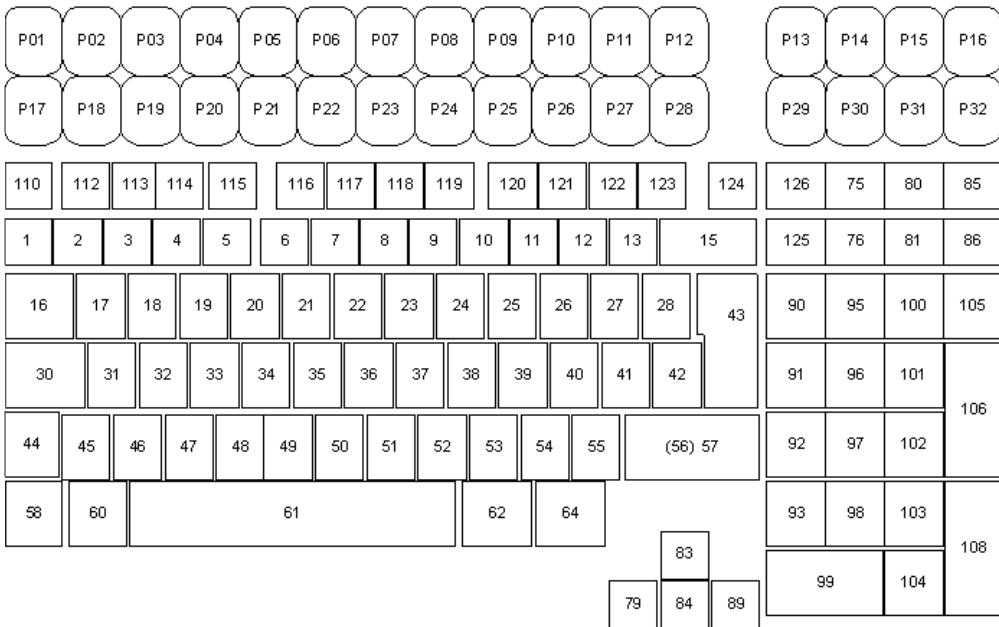
Scan code keyboard assignments

This section shows the CANPOS scan code keyboard assignment by country.

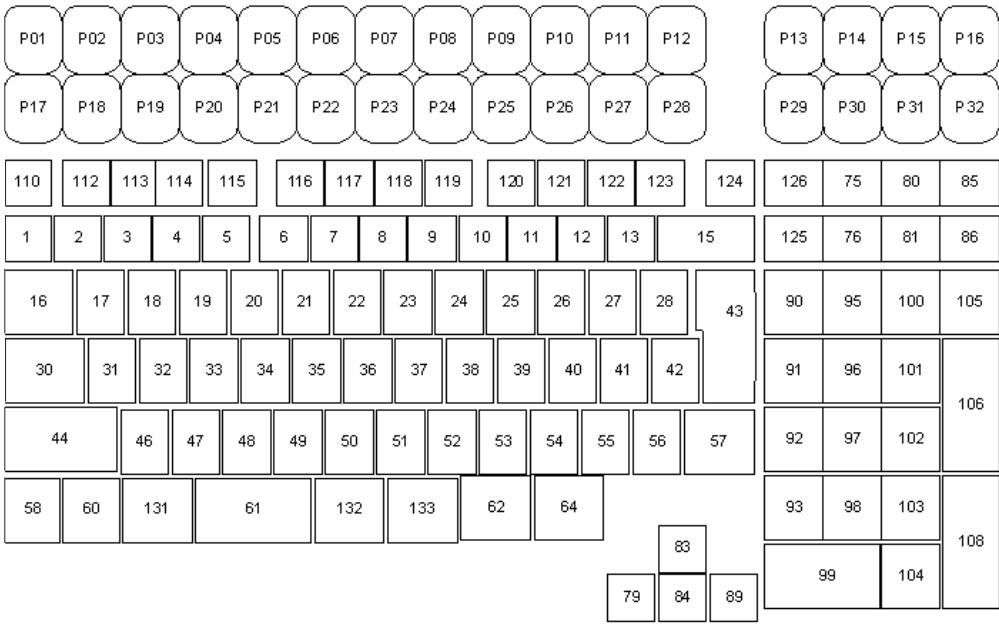
Brazilian keyboard:

P01	P02	P03	P04	P05	P06	P07	P08	P09	P10	P11	P12	P13	P14	P15	P16	
P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31	P32	
110	112	113	114	115	116	117	118	119	120	121	122	123	124	126	75	
1	2	3	4	5	6	7	8	9	10	11	12	13	15	125	80	
16	17	18	19	20	21	22	23	24	25	26	27	28	43	90	105	
30	31	32	33	34	35	36	37	38	39	40	41	42		91	96	
44	45	46	47	48	49	50	51	52	53	54	55	56	57	92	102	
58	60	61					62	64	83	79	84	89	93	98	103	108
														99	104	

European keyboard:



Japanese keyboard:



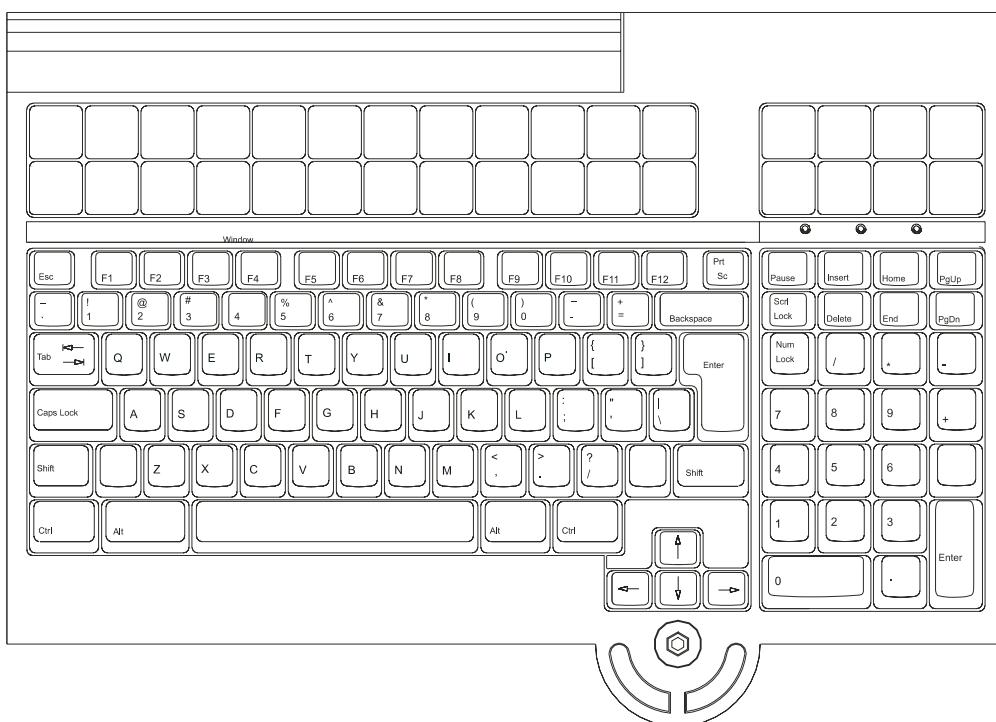
U.S. English keyboard:



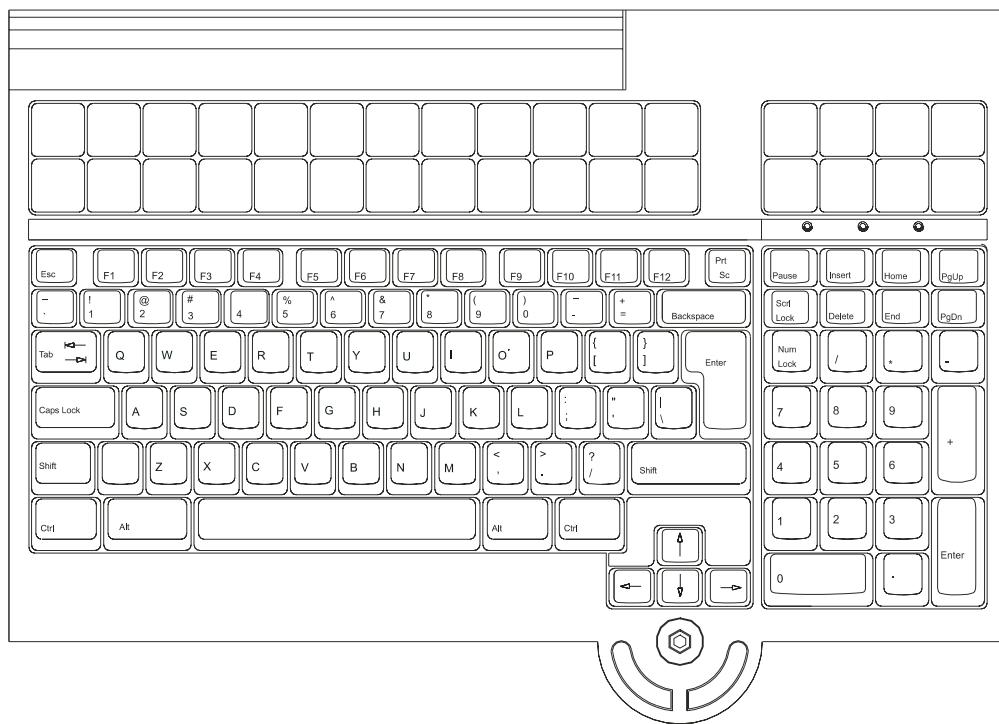
CANPOS country-dependent keyboards

The following pages illustrate the keyboards for all the supported CANPOS keyboards.

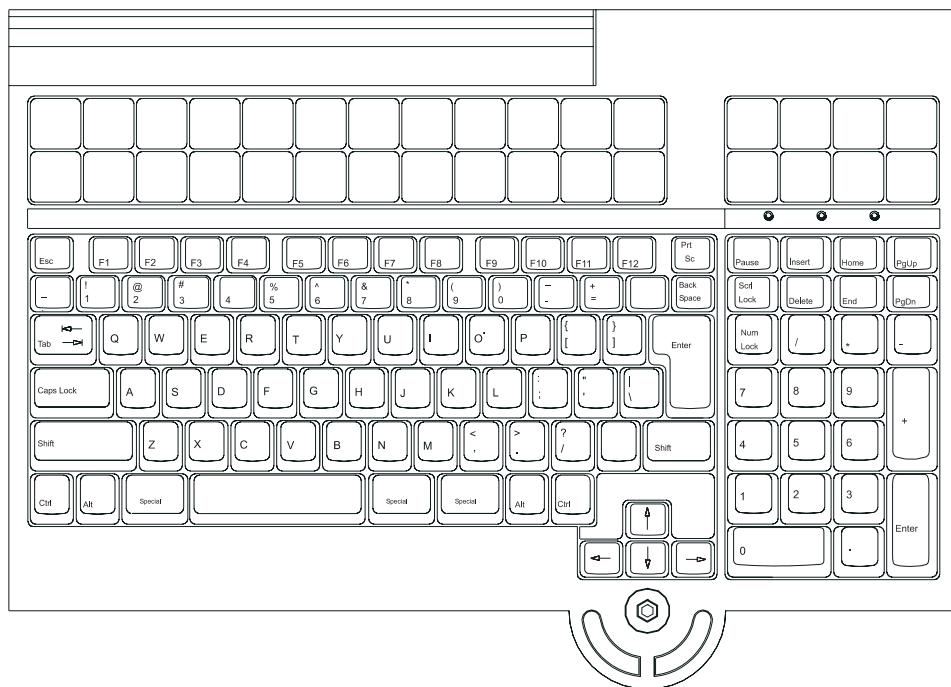
Brazilian keyboard



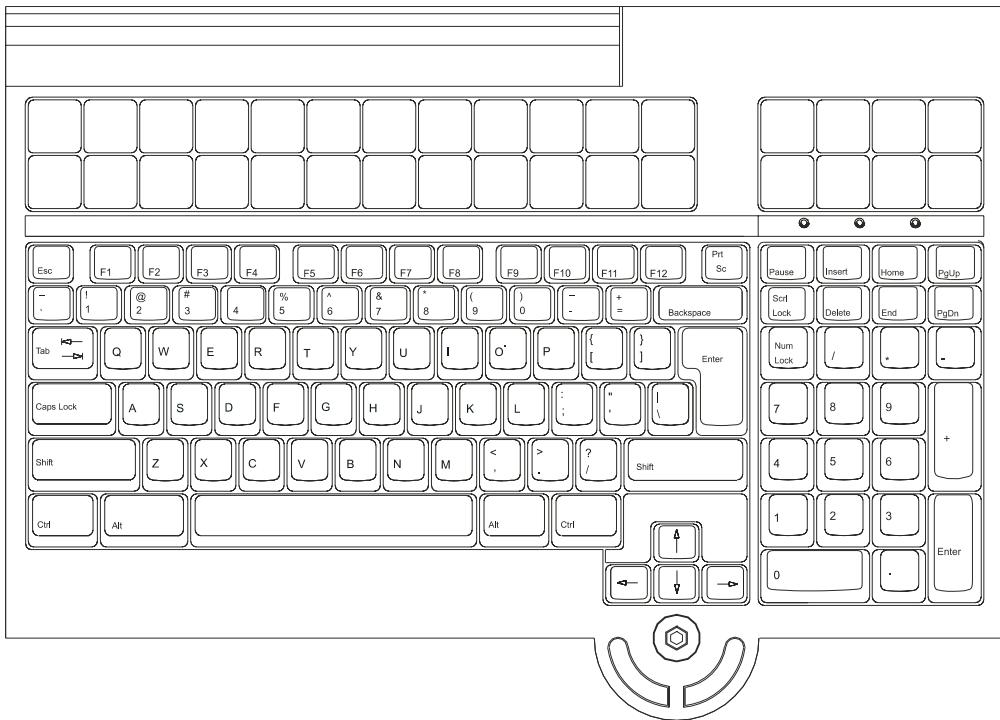
European keyboard



Japanese keyboard



U.S. English keyboard



Chapter 21. Character sets for terminal printers and displays

This chapter contains the character sets used by Toshiba UnifiedPOS drivers, including a link reference to the code page definition.

Codepage definitions can be found on the following Toshiba and Microsoft websites:

www.toshibacommerce.com

www.microsoft.com/en-us/goglobal/bb964653.aspx

Code page implementations may differ across hardware; line displays and printers may not implement characters in exactly the same way.

Table 276. Single-byte character set:

Code Page	Name	Web Reference
101	OEM 7-Segment Alphanumeric font for Line Display	None, see codepage chart below
437	U.S. English	www-01.ibm.com/software/globalization/cp/cp00437.html www.microsoft.com/en-us/goglobal/cc305156.aspx
775	MS-DOS Baltic Rim	www-01.ibm.com/software/globalization/cp/cp00775.html msdn.microsoft.com/en-us/goglobal/cc305159.aspx
808	Cyrillic (with Euro)	www-01.ibm.com/software/globalization/cp/cp00808.html
813	ISO/IEC 8859-7 Latin/Greek alphabet	www-01.ibm.com/software/globalization/ccsid/ccsid813.html
819	ISO/ANSI Multilingual	www-01.ibm.com/software/globalization/cp/cp00819.html
848	Ukrainian (with Euro)	www-01.ibm.com/software/globalization/cp/cp00848.html
850	Personal Computer - Multilingual	www-01.ibm.com/software/globalization/cp/cp00850.html www.microsoft.com/en-us/goglobal/cc305160.aspx
852	Latin-2 - PC	www-01.ibm.com/software/globalization/cp/cp00852.html www.microsoft.com/en-us/goglobal/cc305161.aspx
855	Cyrillic - PC	www-01.ibm.com/software/globalization/cp/cp00855.html www.microsoft.com/en-us/goglobal/cc305162.aspx
857	Latin #5, Turkey – PC	www-01.ibm.com/software/globalization/cp/cp00857.html www.microsoft.com/en-us/goglobal/cc305163.aspx
858	Personal Computer - Multilingual (with Euro)	www-01.ibm.com/software/globalization/cp/cp00858.html www.microsoft.com/en-us/goglobal/cc305164.aspx
860	Portugal - Personal Computer	www-01.ibm.com/software/globalization/cp/cp00860.html
861	Iceland - Personal Computer	www-01.ibm.com/software/globalization/cp/cp00861.html
862	Israel - Personal Computer	www-01.ibm.com/software/globalization/cp/cp00862.html www.microsoft.com/en-us/goglobal/cc305165.aspx
863	Canadian French - Personal Computer	www-01.ibm.com/software/globalization/cp/cp00863.html

Table 276. Single-byte character set: (continued)

Code Page	Name	Web Reference
864	Arabic - Personal Computer	www-01.ibm.com/software/globalization/cp/cp00864.html
865	Nordic - Personal Computer	www-01.ibm.com/software/globalization/cp/cp00865.html
866	PC Data, Cyrillic, Russian	www-01.ibm.com/software/globalization/cp/cp00866.html www.microsoft.com/en-us/goglobal/cc305166.aspx
867	Israel - Personal Computer	www-01.ibm.com/software/globalization/cp/cp00867.html
869	Greece - Personal Computer	www-01.ibm.com/software/globalization/cp/cp00869.html
872	Cyrillic - PC with euro	www-01.ibm.com/software/globalization/cp/cp00872.html
874	Thai with Low Tone Marks & Ancient Chars - PC	www-01.ibm.com/software/globalization/cp/cp00874.html
912	ISO/IEC 8859-2 Latin alphabet No. 2	www-01.ibm.com/software/globalization/ccsid/ccsid912.html
913	ISO/IEC 8859-3 Latin alphabet No. 3	www-01.ibm.com/software/globalization/ccsid/ccsid913.html
914	ISO/IEC 8859-4 Latin alphabet No. 4	www-01.ibm.com/software/globalization/ccsid/ccsid914.html
915	ISO/IEC 8859-5 Latin/Cyrillic alphabet	www-01.ibm.com/software/globalization/ccsid/ccsid915.html
916	ISO/IEC 8859-8 Latin/Hebrew alphabet	www-01.ibm.com/software/globalization/ccsid/ccsid916.html
920	ISO/IEC 8859-9 Latin alphabet No. 5	www-01.ibm.com/software/globalization/ccsid/ccsid920.html
921	ISO/IEC 8859-13 Latin alphabet No. 7 (Baltic Rim)	www-01.ibm.com/software/globalization/ccsid/ccsid921.html
998	The ASCII character set, supporting the ASCII characters	Defined by the UnifiedPOS Specification, www.nrf-arts.org/
999	The ANSI character set	Defined by the UnifiedPOS Specification, www.nrf-arts.org/
1116	Estonia – Personal Computer	www-03.ibm.com/systems/resources/systems_i_software_globalization_pdf_cp01116z.pdf
1117	Latvia – Personal Computer	www-03.ibm.com/systems/resources/systems_i_software_globalization_pdf_cp01117z.pdf
1118	Lithuania– Personal Computer	www-03.ibm.com/systems/resources/servers_eserver_iseries_software_globalization_pdf_cp01118z.pdf
1089	ISO/IEC 8859-6 Latin/Arabic alphabet	www-01.ibm.com/software/globalization/ccsid/ccsid1089.html
1250	Microsoft Windows - Central Europe	www.microsoft.com/en-us/goglobal/cc305143.aspx
1251	Microsoft Windows - Cyrillic	www.microsoft.com/en-us/goglobal/cc305144.aspx
1252	Microsoft Windows - Latin 1	www.microsoft.com/en-us/goglobal/cc305145.aspx
1253	Microsoft Windows - Greek	www.microsoft.com/en-us/goglobal/cc305146.aspx
1254	Microsoft Windows - Turkish	www.microsoft.com/en-us/goglobal/cc305147.aspx
1255	Microsoft Windows - Hebrew	www.microsoft.com/en-us/goglobal/cc305148.aspx
1256	Microsoft Windows - Arabic	www.microsoft.com/en-us/goglobal/cc305149.aspx
1257	Microsoft Windows - Baltic	www.microsoft.com/en-us/goglobal/cc305150.aspx

Table 277. Double-byte character set:

Code Page	Name	Web Reference
926	Korean Hangul	www-01.ibm.com/software/globalization/cp/cp00926.html
932	Japanese Shift-JIS (see Note 2)	www.microsoft.com/en-us/goglobal/cc305152.aspx
936	Simplified Chinese	www.microsoft.com/en-us/goglobal/cc305153.aspx
949	Korean Hangul (see Note 2)	www.microsoft.com/en-us/goglobal/cc305154.aspx
950	Traditional Chinese Big5 (see Note 2)	www.microsoft.com/en-us/goglobal/cc305155.aspx
951	Hong Kong Supplementary Character Set - 2004 (see Note 2)	
1361	Korean Johab	
1381	Simplified Chinese GB2312 (see Note 2, and Note 3)	

Note:

1. See the “Character sets supported” section in each device chapter for the list of supported code pages by device type.
2. Code pages 932, 949, 951 and 1381 are double-byte code pages that are made up of one single-byte character set (SBCS) codepage and one double-byte character set (DBCS) codepage.

Table 278. Codepages

Codepage	Members
932	Toshiba SBCS codepage 897 www-01.ibm.com/software/globalization/cp/cp00897.html Toshiba DBCS codepage 301 www-01.ibm.com/software/globalization/cp/cp00301.html
949	Toshiba SBCS codepage 1088 www-01.ibm.com/software/globalization/cp/cp01088.html Toshiba DBCS codepage 951 www-01.ibm.com/software/globalization/cp/cp00951.html
950	Toshiba SBCS codepage 1114 www-01.ibm.com/software/globalization/cp/cp01114.html Toshiba SBCS codepage 947 www-01.ibm.com/software/globalization/cp/cp00947.html
951	Microsoft DBCS codepage HKSCS (CP951) www.microsoft.com/hk/hkscs/code
1381	Toshiba SBCS codepage 1115 www-01.ibm.com/software/globalization/cp/cp01115.html Toshiba DBCS codepage 1380 www-01.ibm.com/software/globalization/cp/cp01380.html

3. Code pages 950 and 1381 are supported, but are too large to document in this book.

Code Page 101

LED 1X11 LINE DISPLAY CHARACTER SET AND EXPECTED OUTPUT																
	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
2 0 H				-												
															-	
3 0 H	0	1	2	3	4	5	6	7	8	9				=		
	[_]	 _	[_]	[_]	[_]	[_]	[_]	 _	[_]	[_]				-		
4 0 H	A	B	C	D	E	F	G	H	I	J	L		N	O		
	[_]	 _	[_]	[_]	[_]	[_]	[_]	[_]	 _	 _	 _		[_]	[_]	[_]	[_]
5 0 H	P	Q	R	S	T	U				Y						
	[_]	[_]	[_]	[_]	 _	 _			[_]							-
6 0 H	a	b	c	d	e	f	g	h	i	j	l		n	o		
	[_]	 _	[_]	[_]	[_]	[_]	[_]	[_]	 _	 _	 _		[_]	[_]	[_]	[_]
7 0 H	p	q	r	s	t	u			y							
	[_]	[_]	[_]	[_]	 _	 _			[_]							

Figure 78. Code page 101 chart

Code Page 437

HEX DIGITS 1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0		► (SP) SM590000	0 ND100000 SP010000	@ SM050000 ND010000	P LP020000 LA020000	` SD130000 LQ020000	p LP010000 LA010000	Ç LC420000 LU170000	É LE120000 LA510000	á LA110000 LO110000	█ SF140000 SF080000	□ SF020000 SF460000	■ SF470000 LS610000	α GA010000 SA480000	=	
-1	☺ SS000000	◀ SM630000	! SP020000	1 ND010000 LA020000	Q LQ020000 LB020000	a LA010000 LR010000	q LQ010000 LE110000	ü LU170000 LA520000	æ LA150000 LO150000	í LI110000 LO190000	█ SF150000 SF060000	□ SF070000 SF480000	■ SF470000 GG020000	β LS610000 SA530000	±	
-2	☻ SS010000	↑ SM760000	" SP040000	2 ND020000 LB020000	R LR020000 LB010000	b LR010000 LE110000	r LE100000 LA520000	é LE110000 LO110000	Æ LA150000 LO150000	ó LO110000 LU110000	█ SF160000 SF080000	□ SF060000 SF490000	■ SF480000 GP010000	Γ GG020000 SA520000	≥	
-3	♥ SS020000	!! SP330000	# SM010000	3 ND030000 LC020000	S LC020000 LS020000	c LC010000 LS010000	s LS010000 LA150000	â LA150000 LO150000	ô LO150000 LU110000	ú LU110000 SF110000	█ SF080000 SF490000	□ SF060000 SF490000	■ SF480000 GP010000	π GP010000 SA520000	≤	
-4	♦ SS030000	¶ SM250000	\$ SC030000	4 ND040000 LD020000	D LD020000 LT020000	T LD010000 LT010000	d LT010000 LA170000	ä LA170000 LO170000	ö LO170000 LN190000	ñ LN190000 SF090000	█ SF090000 SF100000	□ SF060000 SF500000	■ SF500000 GS020000	Σ GS020000 SS260000	ƒ	
-5	♣ SS040000	§ SM240000	% SM020000	5 ND050000 LE020000	E LE020000 LU020000	U LE010000 LU010000	e LE010000 LU010000	ù LU010000 LA130000	ò LO130000 LN200000	Ñ LN200000 SF190000	█ SF190000 SF050000	□ SF050000 SF510000	■ SF510000 GS010000	σ GS010000 SS270000	ј	
-6	♠ SS050000	- SM700000	& SM030000	6 ND060000 LF020000	F LF020000 LV020000	V LF010000 LV010000	f LV010000 LA270000	å LA270000 LU150000	û LU150000 SM210000	ä SM210000 SF200000	█ SF200000 SF360000	□ SF360000 SF520000	■ SF520000 GM010000	μ GM010000 SA060000	÷	
-7	● SM570000	↓ SM770000	' SP050000	7 ND070000 LG020000	G LG020000 LW020000	W LG010000 LW010000	g LW010000 LC410000	w LC410000 LU130000	ç LU130000 SM200000	à SM200000 SF210000	█ SF210000 SF370000	□ SF370000 SF530000	■ SF530000 GT010000	τ GT010000 SA700000	≈	
-8	▀ SM570001	↑ SM320000	(SP060000	8 ND080000 LH020000	H LH020000 LX020000	X LH010000 LX010000	h LX010000 LE150000	ê LE150000 LY170000	ÿ LY170000 SP160000	█ SF220000 SF380000	□ SF380000 SF540000	■ SF540000 GF020000	Φ GF020000 SM190000	°		
-9	○ SM750000	↓ SM330000) SP070000	9 ND090000 LI020000	I LI020000 LY020000	Y LI010000 LY010000	i LY010000 LE170000	ë LE170000 LO180000	ö LO180000 SM680000	█ SF230000 SF390000	□ SF390000 SF040000	■ SF040000 GT620000	Θ GT620000 SA790000	•		
-A	◎ SM750002	→ SM310000	* SM040000	: SP130000	J LJ020000 LZ020000	Z LJ010000 LZ010000	j LZ010000 LE130000	z LE130000 LU180000	è LU180000 SM660000	█ SF240000 SF400000	□ SF400000 SF010000	■ SF010000 GO320000	Ω GO320000 SD630000	•		
-B	♂ SM280000	← SM300000	+ SA010000	; SP140000	K LK020000 SM060000	[LK010000 SM110000	k SM110000 LI170000	{ LI170000 SC040000	í SC040000 NF010000	½ NF010000 SF250000	█ SF250000 SF410000	□ SF410000 SF610000	■ SF610000 GD010000	δ GD010000 SA800000	√	
-C	♀ SM290000	ₚ SA420000	, SP080000	< SA030000	L LL020000 SM070000	\ LL010000 SM130000	1 SM130000 LI150000	LI150000 SC020000	î SC020000 NF040000	¼ NF040000 SF260000	█ SF260000 SF420000	□ SF420000 SF570000	■ SF570000 SA450000	∞ SA450000 LN011000	n	
-D	♪ SM930000	↔ SM780000	- SP100000	= SA040000	M LM020000 SM080000] LM010000 SM140000	m LM010000 LI130000	{ LI130000 SC050000	ì SC050000 SP030000	¥ SP030000 SF270000	█ SF270000 SF430000	□ SF430000 SF580000	■ SF580000 GF010001	ϕ GF010001 ND021000	²	
-E	♪ SM910000	▲ SM600000	. SP110000	> SA050000	N LN020000 SD150000	^ LN010000 SD190000	n LN010000 LA180000	~ SD190000 SC060000	Ä SC060000 SP170000	Pts SP170000 SF280000	█ SF280000 SF440000	□ SF440000 SF590000	■ SF590000 GE010000	ε GE010000 SM470000		
-F	☀ SM690000	▼ SV040000	/ SP120000	? SP150000	O LO020000 SP090000	_ LO010000 SM790000	o LO010000 LA280000	Å LA280000 SC070000	f SC070000 SP180000	█ SP180000 SF030000	□ SF030000 SF450000	■ SF450000 SF600000	□ SF600000 SA380000	∩ SA380000 (RSP)		

Code Page 00437

Code Page 775

HEX DIGITS 1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0 SM590000	▶ (SP) SP010000	0 ND100000	@ SM050000	P LP020000	` SD130000	p LP010000	Ć LC120000	É LE120000	Ā LA320000	■ SF140000	Ł SF020000	ą LA430000	Ó LO120000	(SHY) SP320000		
-1 SS000000	☺ SM630000	◀ SP020000	! ND010000	1 LA020000	Q LQ020000	a LA010000	q LQ010000	ü LU170000	æ LA510000	Í LI320000	■ SF150000	Ł SF070000	č LC210000	ß LS610000	± SA020000	
-2 SS010000	☻ SM760000	↕ SP040000	" ND020000	2 LB020000	R LR020000	b LB010000	r LR010000	é LE110000	Æ LA520000	ó LO110000	■ SF160000	Ł SF060000	ę LE430000	ó LO320000	“ SP210000	
-3 SS020000	♥ SP330000	!! SM010000	# ND030000	3 LC020000	S LS020000	c LC010000	s LS010000	ā LA310000	ō LO310000	Ż LZ300000	■ SF110000	Ł SF080000	è LE290000	Ń LN120000	¾ NF050000	
-4 SS030000	♦ SM250000	¶ SC030000	\$ ND040000	4 LD020000	T LT020000	d LD010000	t LT010000	ä LA170000	ö LO170000	ż LZ290000	■ SF090000	Ł SF100000	ł LI430000	õ LO190000	¶ SM250000	
-5 SS040000	♣ SM240000	§ SM020000	% ND050000	5 LE020000	E LU020000	e LE010000	u LU010000	ğ LG410000	G LG420000	ż LZ110000	Ł LA440000	Ł SF050000	š LS210000	ő LO200000	§ SM240000	
-6 SS050000	♠ SM700000	- SM030000	& ND060000	6 LF020000	V LV020000	f LF010000	v LV010000	å LA270000	¢ SC040000	” SP220000	Č LC220000	Ų LU440000	ų LU430000	μ SM170000	÷ SA060000	
-7 SM570000	● SM770000	↕ SP050000	' ND070000	7 LG020000	W LW020000	g LG010000	w LW010000	ć LC110000	ś LS120000	ſ SM650000	I LE440000	Ū LU320000	ū LU310000	ń LN110000	„ SP230000	
-8 SM570001	↑ SM320000	(SP060000	8 ND080000	H LH020000	X LX020000	h LH010000	x LX010000	ł LL610000	ś LS110000	© SM520000	Ę LE300000	Ł SF380000	ž LZ210000	ķ LK420000	º SM190000	
-9 SM750000	○ SM330000	↓ SP070000) ND090000	I LI020000	Y LY020000	i LI010000	y LY010000	ē LE310000	ö LO180000	® SM530000	Ł SF230000	Ł SF390000	ķ LK410000	*	SA790000	
-A SM750002	→ SM310000	* SM040000	: SP130000	J LJ020000	Z LZ020000	j LJ010000	z LZ010000	ř LR420000	Ü LU180000	SM660000	Ł SF240000	Ł SF400000	Ł SF010000	ł LL420000	· SD630000	
-B SM280000	♂ SM300000	← SA010000	+ SP140000	K LK020000	[SM060000	k LK010000	{ SM110000	ř LR410000	ø LO610000	NF010000	½ SF250000	½ SF410000	½ SF610000	↓ LL410000	1 ND011000	
-C SM290000	♀ SA420000	, SP080000	< SA030000	L LL020000	\ SM070000	l LL010000	SM130000	í LI310000	£ SC020000	¼ NF040000	Ł SF260000	Ł SF420000	Ł SF570000	ń LN410000	3 ND031000	
-D SM930000	♪ SM780000	↔ SP100000	- SA040000	M LM020000] SM080000	m LM010000	} SM140000	ż LZ120000	Ø LO620000	Ł LL620000	Ł LI440000	Ł SF30000	Ł SF580000	Ę LE320000	2 ND021000	
-E SM910000	♪ SM600000	. SP110000	> SA050000	N LN020000	^ SD150000	n LN010000	~ SD190000	ää LA180000	× SA070000	« SP170000	» LS220000	Ł SF440000	Ł SF590000	ń LN420000	■ SM470000	
-F SM690000	☀ SV040000	/ SP120000	? SP150000	O LO020000	- SP090000	o LO010000	⌂ SM790000	Å LA280000	¤ SC010000	» SP180000	Ł SF030000	Ł LZ220000	Ł SF600000	,	(RSP) SP200000	

Figure 79. Code page 775 chart

Code Page 808

HEX DIGITS	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-	
1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-	
-0	► (SP) SM590000	0 SP010000	@ ND100000	P SM050000	' LP020000	р SD130000	р LP010000	А KA020000	Р KR020000	а KA010000	■ SF140000	Ч SF020000	Е SF460000	р KR010000	Ё KE180000		
-1	! SS000000	! SM630000	1 SP020000	1 ND010000	А LA020000	Q LQ020000	а LA010000	q LQ010000	Б KB020000	С KS020000	б KB010000	■ SF150000	Ч SF070000	Е SF470000	с KS010000	ё KE170000	
-2	! SS010000	! SM760000	" SP040000	" ND020000	В LB020000	Р LR020000	в LB010000	г LR010000	В KV020000	Т KT020000	в KV010000	■ SF160000	Ч SF060000	Е SF480000	т KT010000	€ KE160000	
-3	! SS020000	!! SP330000	# SM010000	# ND030000	С LC020000	С LS020000	с LC010000	с LS010000	Г KG020000	У KU020000	г KG010000	■ SF110000	Ч SF080000	Е SF490000	у KU010000	€ KE150000	
-4	♦ SS030000	♦ SM250000	\$ SC030000	\$ ND040000	Д LD020000	Т LT020000	д LD010000	т LT010000	Д KD020000	Ф KF020000	д KD010000	■ SF090000	Ч SF100000	Е SF500000	ф KF010000	ї KI180000	
-5	♣ SS040000	♣ SM240000	% SM020000	% ND050000	Е LE020000	У LU020000	е LE010000	у LU010000	Е KE020000	Х KH020000	е KE010000	■ SF190000	Ч SF050000	Е SF510000	х KH010000	ї KI170000	
-6	♠ SS050000	— SM700000	& SM030000	6 ND060000	F LF020000	V LV020000	f LF010000	v LV010000	Ж KZ220000	Ц KC020000	ж KZ210000	■ SF200000	Ч SF360000	Е SF520000	ц KC010000	Ӧ KU240000	
-7	● SM570000	! SM770000	' SP050000	' ND070000	G LG020000	W LW020000	g LG010000	w LW010000	З KZ020000	Ч KC220000	з KZ010000	■ SF210000	Ч SF370000	Е SF530000	ч KC210000	Ӧ KU230000	
-8	! SM570001	! SM320000	(SP060000	(ND080000	Н LH020000	Х LX020000	н LH010000	х LX010000	И KI020000	Ш KS220000	и KI010000	■ SF220000	Ч SF380000	Е SF540000	ш KS210000	Ӧ SM190000	
-9	○ SM750000	↓ SM330000) SP070000) ND090000	I LI020000	Y LY020000	i LI010000	y LY010000	Й KJ120000	Щ KS160000	й KJ110000	■ SF230000	Ч SF390000	Е SF400000	щ KS150000	*	SA790000
-A	♂ SM750002	→ SM310000	* SM040000	: SP130000	J LJ020000	Z LZ020000	j LJ010000	z LZ010000	К KK020000	҃ KU220000	к KK010000	■ SF240000	Ч SF400000	Е SF010000	҃ KU210000	· SD630000	
-B	♂ SM280000	← SM300000	+	;	K LK020000	[SM060000	k LK010000	{ SM110000	Л KL020000	Ы KY020000	л KL010000	■ SF250000	Ч SF410000	Е SF610000	ы KY010000	√ SA800000	
-C	♀ SM290000	↔ SA420000	< SP080000	< SA030000	L LL020000	\ SM070000	l LL010000	і SM130000	М KM020000	Ь KX120000	м KM010000	■ SF260000	Ч SF420000	Е SF570000	ь KX110000	№ SM000000	
-D	♪ SM930000	↔ SM780000	- SP100000	= SA040000	М LM020000] SM080000	m LM010000	{ SM140000	Н KN020000	Э KE140000	н KN010000	■ SF270000	Ч SF430000	Е SF580000	э KE130000	€ SC200000	
-E	♪ SM910000	↔ SM600000	. SP110000	> SA050000	N LN020000	^ SD150000	n LN010000	~ SD190000	О KO020000	Ю KU160000	о KO010000	■ SF280000	Ч SF440000	Е SF590000	ю KU150000	■ SM470000	
-F	! SM690000	▼ SV040000	/ SP120000	? SP150000	O LO020000	— SP090000	o LO010000	□ SM790000	П KP020000	Я KA160000	п KP010000	■ SF030000	Ч SF450000	Е SF600000	я KA150000	(RSP) SP300000	

Figure 80. Code page 808 chart

Code Page 819

HEX DIGITS 1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0			(SP) SP010000	0 ND100000	@ SM050000	P LP020000	' SD130000	p LP010000			(RSP) SP300000	° SM190000	À LA140000	Ð LD620000	à LA130000	ð LD630000
-1			! SP020000	1 ND010000	A LA020000	Q LQ020000	a LA010000	q LQ010000			i SP030000	± SA020000	Á LA120000	Ñ LN200000	á LA110000	ñ LN190000
-2			" SP040000	2 ND020000	B LB020000	R LR020000	b LB010000	r LR010000			¢ SC040000	² ND021000	Â LA160000	Ò LO140000	â LA150000	ò LO130000
-3			# SM010000	3 ND030000	C LC020000	S LS020000	c LC010000	s LS010000			£ SC020000	³ ND031000	Ã LA200000	Ó LO120000	ã LA190000	ó LO110000
-4			\$ SC030000	4 ND040000	D LD020000	T LT020000	d LD010000	t LT010000			¤ SC010000	' SD110000	Ä LA180000	Ö LO160000	ä LA170000	ö LO150000
-5			% SM020000	5 ND050000	E LE020000	U LU020000	e LE010000	u LU010000			¥ SC050000	µ SM170000	Å LA280000	Ø LO200000	å LA270000	ø LO190000
-6			& SM030000	6 ND060000	F LF020000	V LV020000	f LF010000	v LV010000			 SM650000	¶ SM250000	Æ LA520000	Ö LO180000	æ LA510000	ö LO170000
-7			' SP050000	7 ND070000	G LG020000	W LW020000	g LG010000	w LW010000			§ SM240000	· SD630000	Ç LC420000	× SA070000	ç LC410000	÷ SA060000
-8			(SP060000	8 ND080000	H LH020000	X LX020000	h LH010000	x LX010000			“ SD170000	„ SD410000	È LE140000	Ø LO620000	è LE130000	ø LO610000
-9) SP070000	9 ND090000	I LI020000	Y LY020000	i LI010000	y LY010000			© SM520000	¹ ND011000	É LE120000	Ù LU140000	é LE110000	ù LU130000
-A			*	:	J SP130000	Z LJ020000	j LJ010000	z LZ010000			¤ SM210000	¤ SM200000	Ê LE160000	Ú LU120000	ê LE150000	ú LU110000
-B			+	;	K LK020000	[SM060000	k LK010000	{ SM110000			« SP170000	» SP180000	Ë LE180000	Û LU160000	ë LE170000	û LU150000
-C			,	< SP080000	L SA030000	\ LL020000	l SM070000	 LL010000			¬ SM660000	¼ NF040000	Ì LI140000	Ü LU180000	ì LI130000	ü LU170000
-D			- SP100000	= SA040000	M LM020000] SM080000	m LM010000	} SM140000			(SHY) SP320000	½ NF010000	Í LI120000	Ý LY120000	í LI110000	ý LY110000
-E			.	> SP110000	N SA050000	^ LN020000	n SD150000	~ LN010000			® SM530000	¾ NF050000	Î LI160000	Þ LT640000	î LI150000	þ LT630000
-F			/ SP120000	?	O LO020000	— SP090000	o LO010000				– SM150000	ł SP160000	Ï LI180000	ß LS610000	ï LI170000	ÿ LY170000

Figure 81. Code page 819 chart

Code Page 848

HEX DIGITS	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0	► (SP) SM590000	0 SP010000	@ ND100000	P SM050000	' LP020000	р SD130000	р LP010000	А KA020000	Р KR020000	а KA010000	■ SF140000	Ч SF020000	Е SF460000	р KR010000	Ё KE180000	
-1	! SS000000	! SM630000	1 SP020000	1 ND010000	А LA020000	Q LQ020000	а LA010000	q LQ010000	Б KB020000	С KS020000	б KB010000	■ SF150000	Ч SF070000	Е SF470000	с KS010000	ё KE170000
-2	! SS010000	! SM760000	" SP040000	" ND020000	В LB020000	Р LR020000	в LB010000	г LR010000	В KV020000	Т KT020000	в KV010000	■ SF160000	Ч SF060000	Е SF480000	т KT010000	Г KG300000
-3	! SS020000	!! SP330000	# SM010000	# ND030000	С LC020000	С LS020000	с LC010000	с LS010000	Г KG020000	У KU020000	г KG010000	■ SF110000	Ч SF080000	Е SF490000	у KU010000	ѓ KG290000
-4	♦ SS030000	♦ SM250000	\$ SC030000	\$ ND040000	Д LD020000	Т LT020000	д LD010000	т LT010000	Д KD020000	Ф KF020000	д KD010000	■ SF090000	Ч SF100000	Е SF500000	ф KF010000	€ KE160000
-5	♣ SS040000	♣ SM240000	% SM020000	% ND050000	Е LE020000	У LU020000	е LE010000	у LU010000	Е KE020000	Х KH020000	е KE010000	■ SF190000	Ч SF050000	Е SF510000	х KH010000	€ KE150000
-6	♠ SS050000	— SM700000	& SM030000	6 ND060000	F LF020000	V LV020000	f LF010000	v LV010000	Ж KZ220000	Ц KC020000	ж KZ210000	■ SF200000	Ч SF360000	Е SF520000	ц KC010000	і KI120000
-7	● SM570000	↓ SM770000	' SP050000	7 ND070000	G LG020000	W LW020000	g LG010000	w LW010000	З KZ020000	Ч KC220000	з KZ010000	■ SF210000	Ч SF370000	Е SF530000	ч KC210000	і KI110000
-8	□ SM570001	↑ SM320000	(SP060000	8 ND080000	Н LH020000	Х LX020000	н LH010000	х LX010000	И KI020000	Ш KS220000	и KI010000	■ SF220000	Ч SF380000	Е SF540000	ш KS210000	ї KI180000
-9	○ SM750000	↓ SM330000) SP070000	9 ND090000	I LI020000	Y LY020000	i LI010000	y LY010000	Й KJ120000	Щ KS160000	й KJ110000	■ SF230000	Ч SF390000	Е SF400000	щ KS150000	ї KI170000
-A	♂ SM750002	→ SM310000	* SM040000	: SP130000	J LJ020000	Z LZ020000	j LJ010000	z LZ010000	К KK020000	҃ KU220000	к KK010000	■ SF240000	Ч SF400000	Е SF010000	ъ KU210000	÷ SA060000
-B	♂ SM280000	← SM300000	+SA010000	; SP140000	K LK020000	[SM060000	k LK010000	{ SM110000	Л KL020000	Ы KY020000	л KL010000	■ SF250000	Ч SF410000	Е SF610000	ы KY010000	± SA020000
-C	♀ SM290000	↔ SA420000	< SP080000	< SA030000	L LL020000	\ SM070000	l LL010000	1 SM130000	М KM020000	Ь KX120000	м KM010000	■ SF260000	Ч SF420000	Е SF570000	ь KX110000	№ SM000000
-D	♪ SM930000	↔ SM780000	- SP100000	= SA040000	M LM020000] SM080000	m LM010000	{ SM140000	Н KN020000	Э KE140000	н KN010000	■ SF270000	Ч SF430000	Е SF580000	э KE130000	€ SC200000
-E	♪ SM910000	↔ SM600000	. SP110000	> SA050000	N LN020000	^ SD150000	n LN010000	~ SD190000	О KO020000	Ю KU160000	о KO010000	■ SF280000	Ч SF440000	Е SF590000	ю KU150000	■ SM470000
-F	☀ SM690000	▼ SV040000	/ SP120000	? SP150000	O LO020000	— SP090000	o LO010000	□ SM790000	П KP020000	Я KA160000	п KP010000	■ SF030000	Ч SF450000	Е SF600000	я KA150000	(RSP) SP300000

Figure 82. Code page 848 chart

Code Page 850

HEX DIGITS 1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0		► (SP) SM590000	0 SP010000 ND100000 SM050000	@ LP020000 SD130000 LP010000	P LC420000 LA110000 LE120000	` LC420000 LA110000 SF140000	p LC420000 LA110000 SF020000	Ç SF140000 SF020000 LD630000	É SF140000 SF020000 LO120000	á SF140000 SF020000 (SHY) SP320000						
-1	☺ SS000000	◀ SM630000	! SP020000 ND010000 LA020000	1 LQ020000 LA010000 LQ010000	A Q LU170000 LA510000 LI110000	a Q LU170000 LA510000 SF150000	q LU170000 LA510000 SF070000	ü LU170000 LA510000 LD620000	æ LU170000 LA510000 SF160000	í LU170000 LA510000 SF060000		ð LD620000 LS610000	Ó LS610000 SA020000			
-2	☻ SS010000	↑ SM760000	" SP040000 ND020000 LB020000	2 LR020000 LB010000 LR010000	B R LE110000 LA520000 LO110000	b R LE110000 LA520000 SF160000	r R LE110000 LA520000 SF060000	é LE110000 LA520000 LE160000	Æ LE110000 LA520000 LE160000	ó LE110000 LA520000 LO160000		Ê LE160000 LO160000	Ô LO160000 SM100000			
-3	♥ SS020000	!! SP330000	# SM010000 ND030000 LC020000	3 LS020000 LC010000 LS010000	C S LA150000 LC150000 LA150000	c S LA150000 LC150000 LO150000	s LA150000 LC150000 LU110000	â LA150000 LC150000 SF110000	ô LA150000 LC150000 SF080000	ú LU110000 SF110000 LE180000		Ë LE180000 LO140000	Ò LO140000 NF050000			
-4	♦ SS030000	¶ SM250000	\$ SC030000 ND040000 LD020000	4 LT020000 LD010000 LT010000	D T LA170000 LO170000 LN190000	d T LA170000 LO170000 SF090000	t T LA170000 LO170000 SF100000	ä LA170000 LO170000 LE140000	ö LN190000 SF090000 LE140000	ñ LN190000 SF100000 LO190000		È LE140000 LO190000	õ LO190000 SM250000			
-5	♣ SS040000	§ SM240000	% SM020000 ND050000 LE020000	5 LU020000 LE010000 LU010000	E U LA130000 LC130000 LA130000	e U LA130000 LC130000 LO130000	u LA130000 LC130000 LN200000	à LA130000 LC130000 LA120000	ò LN200000 LA120000 SF050000	ñ LN200000 LA120000 LI160000		Á LI160000 LO200000	Õ LO200000 SM240000			
-6	♠ SS050000	- SM700000	& SM030000 ND060000 LF020000	6 LV020000 LF010000 LV010000	F V LA270000 LU150000 SM210000	f V LA270000 LU150000 LA160000	v V LA270000 LU150000 LA190000	å LU150000 LA160000 LI120000	û LU150000 LA160000 LI190000	ä LU150000 LA160000 LI190000		â LI190000 LI120000	í SM170000 SA060000			
-7	● SM570000	↓ SM770000	' SP050000 ND070000 LG020000	7 LW020000 LG010000 LW010000	G W LG410000 LG010000 LC410000	g W LG410000 LG010000 LU130000	w LG410000 LG010000 SM200000	ç LU130000 SM200000 LA140000	ù LU130000 SM200000 LA140000	ö LU130000 SM200000 LA200000		À LA200000 LI160000	Ã LI160000 LTG30000			
-8	▀ SM570001	↑ SM320000	(SP060000 ND080000 LH020000	8 LX020000 LH010000 LX010000	H X LH150000 LY170000 SP160000	h X LH150000 LY170000 LY170000	x LH150000 LY170000 SP160000	ê LY170000 SP160000 SM520000	ÿ LY170000 SP160000 SF380000	ö LY170000 SP160000 SF380000		© LY170000 SF380000 LI180000	Ï LI180000 LT640000			
-9	○ SM750000	↓ SM330000) SP070000 ND090000 LJ020000	9 LY020000 LJ010000 LY010000	I Y LJ130000 LM010000 LY010000	i Y LJ130000 LM010000 LE170000	y LJ130000 LM010000 LO180000	ë LE170000 LO180000 SM530000	® LO180000 SM530000 SF230000	ö LO180000 SM530000 SF390000		® LO180000 SF230000 SF400000	Ú LU120000 SD170000			
-A	○ SM750002	→ SM310000	* SM040000 SP130000 LJ020000	: J LZ020000 LJ010000 LZ010000	Z j LZ130000 LJ010000 LE130000	z j LZ130000 LJ010000 LU180000	è LZ130000 LJ010000 SM660000	Ü LU180000 SM660000 SF240000	¬ LZ130000 LJ010000 SF400000	¬ LZ130000 LJ010000 SF010000		À SF010000 LU160000	Â SF010000 SD630000			
-B	♂ SM280000	← SM300000	+ SA010000 SP140000 LK020000	;	K [SM060000 LK010000 SM110000	k [SM060000 LK010000 LI170000	{ SM110000 LI170000 LO610000	ï SM110000 LI170000 NF010000	ø SM110000 LI170000 SF250000	½ SM110000 LI170000 SF410000		Ù SF410000 SF610000	Û SF610000 LU140000			
-C	♀ SM290000	L SA420000	, SP080000 SA030000 LL020000	< L SM070000 LL010000 SM130000	‐ \ SM070000 LL010000 LI150000	‐ \ SM070000 LL010000 SC020000	‐ \ SM130000 LI150000 NF040000	ƒ LI150000 SC020000 SF260000	¼ LI150000 SC020000 SF420000	¼ LI150000 SC020000 SF570000		ÿ LY110000 SF570000	ý LY110000 ND031000			
-D	♪ SM930000	↔ SM780000	- SP100000 SA040000 LM020000	= M SM080000 LM010000 SM140000	M] SM080000 LM010000 LI130000	m] SM080000 LM010000 LO620000	} SM140000 LI130000 SP030000	ì SM140000 LI130000 SC040000	Ø LO620000 SP030000 SF430000	ø LO620000 SP030000 SM650000		í LO620000 SF430000 LY120000	Ý LY120000 ND021000			
-E	♪ SM910000	▲ SM600000	. SP110000 SA050000 LN020000	> N SD150000 LN010000 SD190000	N ^ LN010000 SD150000 LA180000	n ^ LN010000 SD150000 SA070000	~ SD190000 LA180000 SP170000	Ä SA070000 LA180000 SC050000	× SP170000 LA180000 SF440000	« SP170000 LA180000 SF030000		¥ SF440000 SF030000	í LI140000 SC010000	- SM150000 SF600000		
-F	☀ SM690000	▼ SV040000	/ SP120000 SP150000 LO020000	? O SP090000 LO010000 SM790000	O _ LO010000 SM790000 LA280000	o _ LO010000 SM790000 SC070000	◊ SM790000 LA280000 SP180000	Å LA280000 SC070000 SF030000	f SP180000 LA280000 SC010000	» SP180000 LA280000 SF030000		’ SF030000 SC010000 (RSP) SP300000				

Code Page 00850

Code Page 852

HEX DIGITS 1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0 SSM590000	► (SP) SM590000	0 ND100000 SP010000	@ SM050000 ND010000	P LP020000 LA020000	' SD130000 LQ020000	p LP010000 LA010000	Ç LC420000 LQ010000	É LE120000 LU170000	á LA110000 LL120000	SF140000 LI110000	■ SF020000 SF150000	□ SF020000 SF070000	đ LD610000 LS610000	Ó LO120000 SD620000	(SHY) SP320000 SD250000	
-1 SS000000	☺ SM630000	! SP020000 ND010000	1 LA020000 SP040000	A LQ020000 ND020000	Q LA010000 LB020000	a LR020000 LB010000	q LR010000 LE110000	ü LE110000 LL110000	Í LO110000 LO150000	í SF160000 SF060000	■ SF020000 SF110000	□ SF020000 SF080000	Đ LD620000 LO160000	ß LS610000 LN120000	" SD430000 SD210000	
-2 SS010000	☻ SM760000	↑ SP040000 ND020000	" LB020000 SP040000	2 LR020000 ND020000	B LR010000 LB010000	R LR010000 LE110000	b LE110000 LL110000	é LL110000 LO110000	í LO110000 LO150000	ó LO110000 LU110000	■ SF160000 SF060000	□ SF020000 SF110000	Đ LD220000 LO160000	Ô LN120000 LN120000	Ö SD430000 SD210000	
-3 SS020000	♥ SP330000	!! SM010000 ND030000	# SM010000 LC020000	3 LC020000 LS020000	C LC010000 LS010000	S LC010000 LA150000	c LS010000 LA150000	â LA150000 LO150000	ô LO150000 LU110000	ú LU110000 LA170000	■ SF110000 SF080000	□ SF020000 SF110000	Ë LE180000 LE180000	Ñ LN120000 LN120000	ˇ SD210000 SD230000	
-4 SS030000	♦ SM250000	¶ SC030000 ND040000	\$ ND040000 LD020000	4 LD020000 LT020000	D LT020000 LD010000	T LD010000 LT010000	d LT010000 LA170000	ä LA170000 LO170000	ö LA170000 LA440000	À LA440000 SF090000	■ SF090000 SF100000	□ SF020000 SF100000	đ LD210000 LN110000	ń LN110000 LN110000	^ SD230000 SD230000	
-5 SS040000	♣ SM240000	§ SM020000 ND050000	% ND050000 LE020000	5 LE020000 LU020000	E LU020000 LE010000	U LU010000 LU010000	e LU010000 LU270000	ü LU270000 LL220000	Ł LL220000 LA430000	ą LA430000 LA120000	Á LA120000 SF050000	■ SF050000 LN220000	Ñ LN220000 LN220000	њ LN220000 LN220000	§ SM240000 SM240000	
-6 SS050000	♠ SM700000	- SM030000 ND060000	& ND060000 LF020000	6 LF020000 LV020000	F LV020000 LF010000	V LF010000 LV010000	f LV010000 LC110000	v LC110000 LL210000	ć LL210000 LZ220000	Ĩ LZ220000 LA160000	Ž LA160000 LA240000	À LA240000 LJ120000	Í LS220000 LS220000	Š LS220000 SA060000	÷ SA060000	
-7 SM570000	● SM770000	↑ SP050000 ND070000	' LG020000 LG010000	7 LW020000 LG010000	G LW010000 LG010000	W LW010000 LC410000	g LW010000 LS120000	ć LC410000 LZ210000	Ś LZ210000 LE220000	ž LE220000 LA230000	Ę LA230000 LI160000	ä LI160000 LS210000	î LS210000 SD410000	, LS210000 SD410000		
-8 SM570001	▀ SM320000	↑ SP060000 ND080000	(LH020000 LH010000	8 LX020000 LH010000	H LX010000 LH010000	X LX010000 LL610000	h LX010000 LS110000	ł LL610000 LE440000	ś LE440000 LS420000	Ę LS420000 SF380000	Ž SF380000 SF400000	Ł SF400000 SF230000	Ŕ LE210000 LU120000	Ŗ LR120000 SD170000	° SM190000	
-9 SM750000	○ SM330000	↓ SP070000 ND090000) LI020000 LY020000	9 LI020000 LY020000	I LY020000 LI010000	Y LY020000 LY010000	i LY010000 LE170000	ë LE170000 LO180000	ö LO180000 LE430000	Ę LE430000 SF230000	Ő SF230000 SF390000	Ő SF390000 SF040000	Ú LU120000 LU120000	“ SD170000	” SD170000	
-A SM750002	○ SM310000	→ SM040000	* SP130000 LJ020000	: LJ020000 LZ020000	J LJ020000 LZ020000	Z LJ010000 LZ010000	j LJ010000 LO260000	z LO260000 LU180000	Ó LU180000 SF240000	Ü SF240000 SF400000	Ő SF400000 SF010000	Ü SF010000 LR110000	ŕ LR110000 SD290000	ŕ SD290000		
-B SM280000	♂ SM300000	← SA010000 SP140000	+ SA010000 LK020000	; K SM060000 LK010000	[SM060000 SM110000	k SM110000 LO250000	{ SM110000 LT220000	ő LT220000 LZ110000	Ł LZ110000 SF250000	Ž SF250000 SF410000	Ź SF410000 SF610000	Ű SF610000 LU260000	ű LU260000 LU250000	ű LU250000		
-C SM290000	♀ SA420000	, SP080000 SA030000	< SA030000 LL020000	L SM070000 SM070000	\ LL010000 SM130000	1 SM130000 LI150000	LI150000 LT210000	î LT210000 LC220000	Ł LC220000 SF260000	Č SF260000 SF420000	Ŗ SF420000 SF570000	Ý LY110000 LY110000	Ŕ LR220000 LR220000	Ŕ LR220000		
-D SM930000	♪ SM780000	↔ SP100000 SA040000	- SA040000 LM020000	= LM020000 SM080000	M LM010000 SM140000	m LM010000 LZ120000	} SM140000 LL620000	Ž LZ120000 LS410000	Ł LL620000 LZ300000	Ş SF430000 SF430000	Ž SF430000 SF440000	Ŕ LT420000 LU280000	Ŕ LY120000 LT410000	Ŕ LR210000 SM470000		
-E SM910000	♪ SM600000	. SP110000 SA050000	> LN020000 SD150000	N LN020000 LN010000	^ SD150000 SD190000	n LN010000 SD190000	~ SD190000 LA180000	Ä SA070000 SA070000	× SP170000 SP170000	« LZ290000 SF440000	ž LZ290000 SF440000	Ü SF440000 SF600000	ő LU280000 LT410000	ő LT410000 SM470000		
-F SM690000	☀ SV040000	/ SP120000 SP150000	? LO020000 SP090000	O LO010000 LO010000	— SM790000 SM790000	o SM790000 LC120000	◊ LC120000 LC210000	ć LC210000 SP180000	» SP180000 SF030000	„ SF030000 SC010000	„ SF030000 SF600000	’ SF600000 SD110000	’ SD110000 (RSP)	’ SD110000 SP300000		

Code Page 00852

Code Page 855

HEX DIGITS	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-	
1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-	
-0	▶ (SP) SM590000	SP010000	0 ND100000	@ SM050000	P LP020000	ˇ SD130000	p LP010000	჊ KD610000	ъ KL410000	ა KA010000	გ SF140000	ლ SF020000	լ KL010000	յ KA160000	я (SHY) SP320000		
-1	😊 SS000000	SM630000	! ND010000	1 LA020000	Q LQ020000	a LA010000	q LQ010000	҃ KD620000	҂ KL420000	ѧ KA020000	ց SF150000	լ SF070000	լ KL020000	ր KR010000	ы KY010000		
-2	😢 SS010000	SM760000	” SP040000	2 ND020000	R LB020000	b LB010000	r LR010000	՚ KG110000	՚ KN110000	բ KB010000	ց SF160000	լ SF060000	լ KM010000	ր KR020000	ы KY020000		
-3	♥ SS020000	SP330000	!! SM010000	# ND030000	C LC020000	S LS020000	c LC010000	s LS010000	՚ KG120000	՚ KN120000	բ KB020000	ց SF110000	լ SF080000	լ KM020000	չ KS010000	զ KZ010000	
-4	♦ SS030000	SM250000	¶ SC030000	\$ ND040000	D LD020000	T LT020000	d LD010000	t LT010000	՚ KE170000	՚ KC110000	պ KC010000	ց SF090000	լ SF100000	լ KN010000	չ KS020000	զ KZ020000	
-5	♣ SS040000	SM240000	% SM020000	5 ND050000	E LEO20000	U LU020000	e LEO10000	u LU010000	՚ KE180000	՚ KC120000	՛ KC020000	չ KH010000	լ SF050000	լ KN020000	տ KT010000	ի KS210000	
-6	♠ SS050000	SM700000	& SM030000	6 ND060000	F LF020000	V LV020000	f LF010000	v LV010000	՚ KE150000	՚ KK110000	դ KD010000	չ KH020000	կ KK010000	օ KO010000	տ KT020000	ի KS220000	
-7	● SM570000	SM770000	' SP050000	7 ND070000	G LG020000	W LW020000	g LG010000	w LW010000	՚ KE160000	՚ KK120000	դ KD020000	չ KI010000	կ KK020000	օ KO020000	յ KU010000	զ KE130000	
-8	↑ SM570001	SM320000	(SP060000	8 ND080000	H LH020000	X LX020000	һ LH010000	ҳ LX010000	՚ KZ150000	՚ KU230000	ե KE010000	լ KI020000	լ SF380000	լ KP010000	ւ KU020000	զ KE140000	
-9	○ SM750000	SM330000) SP070000	9 ND090000	I LI020000	Y LY020000	i LI010000	y LY010000	՚ KZ160000	՚ KU240000	ե KE020000	լ SF390000	լ SF040000	լ KZ210000	շ KS150000		
-A	⦿ SM750002	SM310000	* SM040000	:	J LJ020000	Z LZ020000	j LJ010000	z LZ010000	՚ KI110000	՚ KG210000	Փ KF010000	լ SF240000	լ SF400000	լ SF010000	ժ KZ220000	՛ KS160000	
-B	♂ SM280000	SM300000	← SA010000	+	;	K LK020000	[SM060000	k LK010000	{ SM110000	I KI120000	Ա KG220000	Փ KF020000	լ SF250000	լ SF410000	մ SF610000	վ KV010000	չ KC210000
-C	♀ SM290000	SA420000	, SP080000	՝ SA030000	L LL020000	\ SM070000	l LL010000	՚ SM130000	՚ KI170000	յ KU150000	յ KG010000	լ SF260000	լ SF420000	մ SF570000	վ KV020000	չ KC220000	
-D	♪ SM930000	SM780000	- SP100000	= SA040000	M LM020000	m SM080000] LM010000	}	՚ SM140000	՚ KI180000	յ KU160000	յ KG020000	լ SF430000	լ KP020000	յ KX110000	չ SM240000	
-E	♪ SM910000	SM600000	. SP110000	> SA050000	N LN020000	^ SD150000	n LN010000	~ SD190000	՚ KJ010000	՚ KU210000	յ SP170000	յ KJ120000	լ SF440000	լ KA150000	յ KX120000	մ SM470000	
-F	☀ SM690000	SV040000	/ SP120000	? SP150000	O LO020000	— SP090000	o LO010000	՞ SM790000	՚ KJ020000	՚ KU220000	՝ SP180000	՝ SF030000	մ SC010000	մ SF600000	մ SM000000	՞ (RSP) SP300000	

Code Page 857

HEX DIGITS 1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0 SSM590000	► (SP) SM590000	0 ND100000 SP010000	@ SM050000 ND010000	P LP020000 LA020000	' SD130000 LQ020000	p LP010000 LA010000	Ç LC420000 LU170000	É LE120000 LA510000	á LA110000 LO110000	SF140000 LI110000	■ SF020000 SF150000	□ SF020000 SF070000	◊ SM200000 LS210000	º LO120000 LE120000	Ó SP320000 SA200000	(SHY)
-1 SS000000	☺ SM630000	! SP020000 ND010000	1 LA020000 SP040000	A LQ020000 ND020000	Q LA010000 LB020000	a LR020000 LB010000	q LR010000 LE110000	ü LE100000 LA520000	æ LA150000 LO150000	í LO110000 LU110000	■ SF160000 SF060000	□ SF160000 SF080000	◊ SM210000 LS610000	ª SM200000 LE180000	ß SA200000 LO140000	± NF050000
-2 SS010000	☻ SM760000	" SP040000 ND020000	2 LB020000 SP040000	B LR020000 ND020000	R LB010000 LR020000	b LR010000 LE110000	r LE100000 LA520000	é LE110000 LO110000	Æ LO100000 SF110000	ó LO150000 SF110000	■ SF160000 SF060000	□ SF160000 SF080000	◊ LE160000 LO160000	Ê LE160000 LO160000	Ô LO160000 NF050000	
-3 SS020000	♥ SP330000	!! SM010000 ND030000	# SM010000 ND030000	3 LC020000 LC010000	S LS020000 LS010000	c LC010000 LS010000	s LS010000 LA150000	â LA150000 LO150000	ô LO150000 LU110000	ú LU110000 SF110000	■ SF160000 SF080000	□ SF160000 SF080000	◊ LE180000 LE180000	Ë LE180000 LO140000	Ò LO140000 NF050000	¾
-4 SS030000	♦ SM250000	¶ SC030000 ND040000	\$ ND040000 LD020000	4 LD020000 LT020000	D LT020000 LD010000	T LD010000 LT010000	d LT010000 LA170000	ä LA170000 LO170000	ö LO170000 LN190000	ñ LN190000 SF090000	■ SF100000 SF090000	□ SF100000 SF090000	◊ LE140000 LE140000	È LO190000 LO190000	õ SM250000 SM250000	
-5 SS040000	♣ SM240000	§ SM020000 ND050000	% ND050000 LE020000	5 LE020000 LU020000	E LU020000 LE010000	U LU010000 LU010000	e LU010000 LA130000	à LA130000 LO130000	ò LO130000 LN200000	Ñ LN200000 LA120000	Á LA120000 SF050000	◊ LO200000 LA190000	Õ LO200000 LA190000	§ SM240000 SM240000		
-6 SS050000	♠ SM700000	- SM030000 ND060000	& ND060000 LF020000	6 LF020000 LV020000	F LV020000 LF010000	V LF010000 LV010000	f LV010000 LA270000	å LA270000 LU150000	û LU150000 LG240000	Ğ LG240000 LA160000	Â LA160000 LA160000	˜ LA190000 LA190000	Í LI120000 LI120000	µ SM170000 SA060000	÷ SA060000 SA060000	
-7 SM570000	● SM770000	↑ SP050000 ND070000	' ND070000 LG020000	7 LG020000 LG010000	G LW020000 LG010000	W LW010000 LG010000	g LW010000 LC410000	w LC410000 LU130000	ç LU130000 LG230000	ù LG230000 LA140000	ÿ LA140000 LA200000	À LA200000 LI160000	Ã LI160000 LI160000	Î SD410000 SD410000		
-8 SM570001	▀ SM320000	↑ SP060000 ND080000	(ND080000 LH020000	8 LH020000 LX020000	H LX020000 LH010000	X LH010000 LX010000	h LX010000 LE150000	ê LE150000 LI300000	í LI300000 SP160000	® SP160000 SM520000	© SF380000 SF380000	▀ LI180000 LI180000	× SA070000 SM190000	° SM190000 SM190000		
-9 SM750000	○ SM330000	↓ SP070000 ND090000) ND090000 LI020000	9 LI020000 LY020000	I LY020000 LI010000	Y LY010000 LY010000	i LY010000 LE170000	ë LE170000 LO180000	ö LO180000 SM530000	® SM530000 SF230000	▀ SF390000 SF390000	▀ SF040000 SF040000	Ú LU120000 LU120000	„ SD170000 SD170000		
-A SM750002	○ SM310000	→ SM040000	* SP130000 LJ020000	: LJ020000 LZ020000	J LZ020000 LJ010000	Z LJ010000 LZ010000	j LZ010000 LE130000	z LE130000 LU180000	è LU180000 SM660000	Ü SM660000 SF240000	▀ SF400000 SF400000	▀ SF010000 SF010000	Û LU160000 LU160000	• SD630000 SD630000		
-B SM280000	♂ SM300000	← SA010000 SP140000	+ SA010000 LK020000	; K SM060000 LK010000	[SM060000 SM110000	k LK010000 SM110000	{ SM110000 LI170000	í LI170000 LO610000	ø LO610000 NF010000	½ NF010000 SF250000	▀ SF410000 SF410000	▀ SF610000 SF610000	Ù LU140000 LU140000	¹ ND011000 ND011000		
-C SM290000	♀ SA420000	, SP080000 SA030000	< SA030000 LL020000	L LL020000 SM070000	\ LL010000 SM130000	1 LL010000 LI150000	SM130000 SC020000	î LI150000 NF040000	£ NF040000 SF260000	¼ SF260000 SF420000	▀ SF570000 SF570000	▀ LI130000 LI130000	ì ND031000 ND031000	³ ND031000 ND031000		
-D SM930000	♪ SM780000	↔ SP100000 SA040000	- SA040000 LM020000	= LM020000 SM080000	M LM010000 SM140000] LM010000 SM140000	m SM140000 LI610000	} LI610000 LO620000	Ø SP030000 SC040000	í SP030000 SF430000	¢ SF430000 SF430000	▀ SM650000 SM650000	ÿ LY170000 LY170000	² ND021000 ND021000		
-E SM910000	♪ SM600000	. SP110000 SA050000	> SA050000 LN020000	N SD150000 SD150000	^ LN010000 SD190000	n LN010000 SD190000	~ SD190000 LA180000	Ä LA180000 LS420000	§ SP170000 SC050000	« SP170000 SF440000	¥ SF440000 SF440000	▀ LI140000 SF600000	- SM150000 SD110000	▀ SM150000 (RSP) SP300000	■ SM470000 SP300000	
-F SM690000	☀ SV040000	▼ SP120000 SP150000	/ SP150000 LO020000	? SP090000 LO010000	○ SM790000 SM790000	_ LO010000 LA280000	o LO010000 LS410000	◊ LA280000 SP180000	Å SP180000 SF030000	» SF030000 SC010000	▀ SF600000 SF600000	▀ SF600000 SF600000	▀ SD110000 SD110000	’ SP300000 SP300000		

Code Page 00857

Code Page 858

HEX DIGITS 1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0		► (SP) SM590000 SP010000	0 ND100000 SM050000 LP020000 SD130000 LP010000 LC420000 LE120000 LA110000 SF140000 SF020000	@ P ` p Ç É á										ð LD630000 LO120000	Ó (SHY) SP320000	
-1	☺ SS000000 SM630000 SP020000 ND010000 LA020000 LQ020000 LA010000 LQ010000 LU170000 LAS10000 LI110000 SF150000 SF070000	! 1 A Q a q ü æ í												Ð LD620000 LS610000 SA020000	Þ	
-2	☻ SS010000 SM760000 SP040000 ND020000 LB020000 LR020000 LB010000 LR010000 LE110000 LA520000 LO110000 SF160000 SF060000	↔ " 2 B R b r é Æ ó												Ê LE160000 LO160000 SM100000	Ô	
-3	♥ SS020000 SP330000 SM010000 ND030000 LC020000 LS020000 LC010000 LS010000 LA150000 LO150000 LU110000 SF110000 SF080000	!! # 3 C S c s â ô ú												Ë LE180000 LO140000 NF050000	Ò ¾	
-4	♦ SS030000 SM250000 SC030000 ND040000 LD020000 LT020000 LD010000 LT010000 LA170000 LO170000 LN190000 SF090000 SF100000	¶ \$ 4 D T d t ä ö ñ												È LE140000 LO190000 SM250000	õ ¶	
-5	♣ SS040000 SM240000 SM020000 ND050000 LE020000 LU020000 LE010000 LU010000 LA130000 LO130000 LN200000 LA120000 SF050000	§ % 5 E U e u à ò Ñ Á												€ L1610000 LO200000 SM240000	Ñ §	
-6	♠ SS050000 SM700000 SM030000 ND060000 LF020000 LV020000 LF010000 LV010000 LA270000 LU150000 SM210000 LA160000 LA190000 LI120000 SM170000 SA060000	- & 6 F V f v å û à												í L1120000 SM170000	÷	
-7	● SM570000 SM770000 SP050000 ND070000 LG020000 LW020000 LG010000 LW010000 LC410000 LU130000 SM200000 LA140000 LA200000 LI160000 LT630000 SD410000	↓ ' 7 G W g w ç ù ò												À LA200000 LI160000 LT630000 SD410000	Ã ï þ	
-8	● SM570001 SM320000 SP060000 ND080000 LH020000 LX020000 LH010000 LX010000 LE150000 LY170000 SP160000 SM520000 SF380000	↑ (8 H X h x ê ÿ ï												í LII80000 LT640000 SM190000	º	
-9	○ SM750000 SM330000 SP070000 ND090000 LJ020000 LY020000 LJ010000 LY010000 LE170000 LO180000 SM530000 SF230000 SF390000	↓) 9 I Y i y ë ö ®												ú LU120000 SD170000	ú ..	
-A	○ SM750002 SM310000 SM040000 SP130000 LJ020000 LZ020000 LJ010000 LZ010000 LE130000 LU180000 SM660000 SF240000 SF400000	→ * : J Z j z è Ü ÷												û LU160000 SD630000	•	
-B	♂ SM280000 SM300000 SA010000 SP140000 LK020000 SM060000 LK010000 SM110000 LI170000 LO610000 NF010000 SF250000 SF410000	← + ; K [k { ï ø ½												ù LU140000 ND011000	û 1	
-C	♀ SM290000 SA420000 SP080000 SA030000 LL020000 SM070000 LL010000 SM130000 LI150000 SC020000 NF040000 SF260000 SF420000	L , < L \ 1 î £ ¼												ý LY110000 ND031000	³	
-D	♪ SM930000 SM780000 SP100000 SA040000 LM020000 SM080000 LM010000 SM140000 LI130000 LO620000 SP030000 SC040000 SF430000	↔ - = M] m } ï Ø i ¢												ý LY120000 ND021000	²	
-E	♪ SM910000 SM600000 SP110000 SA050000 LN020000 SD150000 LN010000 SD190000 LA180000 SA070000 SP170000 SC050000 SF440000	→ . > N ^ n ~ Ä × « ¥												í LI140000 SM150000 SM470000	í - ■	
-F	☀ SM690000 SV040000 SP120000 SP150000 LO020000 SP090000 LO010000 SM790000 LA280000 SC070000 SP180000 SF030000 SC010000	↓ / ? O — o Å f »												’ SF600000 SD110000 SP300000	(RSP)	

Code Page 00858

Code Page 860

HEX DIGITS 1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0		► (SP)	0	@	P	'	p	Ç	É	á	[grid]	[square]	[square]	[square]	α	≡
-1	☺ SS000000	◀ SM630000	! SP020000	1 ND010000	A LA020000	Q LQ020000	a LA010000	q LQ010000	ü LU170000	À LA140000	í LI110000	[grid] SF150000	[square] SF070000	[square] SF470000	β LS610000	± SA020000
-2	☻ SS010000	↑ SM760000	" SP040000	2 ND020000	B LB020000	R LR020000	b LB010000	r LR010000	é LE110000	È LE140000	ó LO110000	[grid] SF160000	[square] SF060000	[square] SF480000	Γ GG020000	≥ SA530000
-3	♥ SS020000	!! SP330000	# SM010000	3 ND030000	C LC020000	S LC010000	c LS010000	s LS010000	â LA150000	ô LO150000	ú LU110000	[grid] SF110000	[square] SF080000	[square] SF490000	π GP010000	≤ SA520000
-4	♦ SS030000	¶ SM250000	\$ SC030000	4 ND040000	D LD020000	T LT020000	d LD010000	t LT010000	ã LA190000	õ LO190000	ñ LN190000	[grid] SF090000	[square] SF100000	[square] SF500000	Σ GS020000	ƒ SS260000
-5	♣ SS040000	§ SM240000	% SM020000	5 ND050000	E LE020000	U LU020000	e LE010000	u LU010000	à LA130000	ò LO130000	ñ LN200000	[grid] SF190000	[square] SF050000	[square] SF510000	σ GS010000	ϳ SS270000
-6	♠ SS050000	- SM700000	& SM030000	6 ND060000	F LF020000	V LV020000	f LF010000	v LV010000	Á LA120000	Ú LU120000	¤ SM210000	[grid] SF200000	[square] SF360000	[square] SF520000	μ GM010000	÷ SA060000
-7	● SM570000	↓ SM770000	' SP050000	7 ND070000	G LG020000	W LW020000	g LG010000	w LW010000	ç LC410000	ù LU130000	¤ SM200000	[grid] SF210000	[square] SF370000	[square] SF530000	τ GT010000	≈ SA700000
-8	▀ SM570001	↑ SM320000	(SP060000	8 ND080000	H LH020000	X LX020000	h LH010000	x LX010000	ê LE150000	Ì LJ140000	¿ SP160000	[grid] SF220000	[square] SF380000	[square] SF540000	Φ GF020000	° SM190000
-9	○ SM750000	↓ SM330000) SP070000	9 ND090000	I LI020000	Y LY020000	i LI010000	y LY010000	Ê LE160000	Ò LO200000	Ó LO140000	[grid] SF230000	[square] SF390000	[square] SF040000	Θ GT620000	• SA790000
-A	◎ SM750002	→ SM310000	* SM040000	: SP130000	J LJ020000	Z LZ020000	j LJ010000	z LZ010000	è LE130000	Ü LU180000	SM660000	[grid] SF240000	[square] SF400000	[square] SF010000	Ω GO320000	• SD630000
-B	♂ SM280000	← SM300000	+ SA010000	; SP140000	K LK020000	[SM060000	k LK010000	{ SM110000	í LI120000	¢ SC040000	½ NF010000	[grid] SF250000	[square] SF410000	[square] SF610000	δ GD010000	√ SA800000
-C	♀ SM290000	∟ SA420000	, SP080000	< SA030000	L LL020000	\ SM070000	LL010000	1 SM130000	Ô LO160000	£ SC020000	¼ NF040000	[grid] SF260000	[square] SF420000	[square] SF570000	∞ SA450000	▫ LN011000
-D	♪ SM930000	↔ SM780000	- SP100000	= SA040000	M LM020000] SM080000	m LM010000	{ SM140000	ì LI130000	Ù LU140000	í SP030000	[grid] SF270000	[square] SF430000	[square] SF580000	ϕ GF010001	² ND021000
-E	♪ SM910000	▲ SM600000	. SP110000	> SA050000	N LN020000	^ SD150000	n LN010000	~ SD190000	Ã LA200000	Pts SC060000	« SP170000	[grid] SF280000	[square] SF440000	[square] SF590000	ε GE010000	■ SM470000
-F	☀ SM690000	▼ SV040000	/ SP120000	? SP150000	O LO020000	_ SP090000	o LO010000	◊ SM790000	Â LA160000	Ó LO120000	» SP180000	[grid] SF030000	[square] SF450000	[square] SF600000	∩ SA380000	(RSP) SP300000

Code Page 00860

Code Page 861

HEX DIGITS 1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-		
-0		► SMS90000	(SP) SP010000	0 ND100000	@ SM050000	P LP020000	` SD130000	p LP010000	Ç LC420000	É LE120000	á LA110000	[grid] SF140000	[grid] SF020000	[grid] SF460000	α GA010000	≡ SA480000		
-1	☺ SS000000	◀ SM630000	! SP020000	1 ND010000	A LA020000	Q LQ020000	a LA010000	q LQ010000	ü LU170000	æ LA510000	í LI110000	[grid] SF150000	[grid] SF070000	[grid] SF470000	ß LS610000	± SA020000		
-2	☻ SS010000	↑ SM760000	" SP040000	2 ND020000	B LB020000	R LR020000	b LB010000	r LR010000	é LE110000	Æ LA520000	ó LO110000	[grid] SF160000	[grid] SF060000	[grid] SF480000	Γ GG020000	≥ SA530000		
-3	♥ SS020000	!! SP330000	# SM010000	3 ND030000	C LC020000	S LS020000	c LC010000	s LS010000	â LA150000	ô LO150000	ú LU110000	[grid] SF110000	[grid] SF080000	[grid] SF490000	π GP010000	≤ SA520000		
-4	♦ SS030000	¶ SM250000	\$ SC030000	4 ND040000	D LD020000	T LT020000	d LD010000	t LT010000	ä LA170000	ö LO170000	Á LA120000	[grid] SF090000	[grid] SF100000	[grid] SF500000	Σ GS020000	ƒ SS260000		
-5	♣ SS040000	§ SM240000	% SM020000	5 ND050000	E LE020000	U LU020000	e LE010000	u LU010000	à LA130000	þ LT630000	Í LI120000	[grid] SF190000	[grid] SF050000	[grid] SF510000	σ GS010000	J SS270000		
-6	♠ SS050000	- SM700000	& SM030000	6 ND060000	F LF020000	V LV020000	f LF010000	v LV010000	å LA270000	û LU150000	Ó LO120000	[grid] SF200000	[grid] SF360000	[grid] SF520000	μ GM010000	÷ SA060000		
-7	● SM570000	↑ SM770000	' SP050000	7 ND070000	G LG020000	W LW020000	g LG010000	w LW010000	ç LC410000	Ý LY120000	Ú LU120000	[grid] SF210000	[grid] SF370000	[grid] SF330000	τ GT010000	≈ SA700000		
-8	█ SM570001	↑ SM320000	(SP060000	8 ND080000	H LH020000	X LX020000	h LH010000	x LX010000	ê LE150000	ý LY110000	ķ SP160000	[grid] SF220000	[grid] SF380000	[grid] SF540000	Φ GF020000	° SM190000		
-9	○ SM750000	↓ SM330000) SP070000	9 ND090000	I LI020000	Y LY020000	i LI010000	y LY010000	ë LE170000	ö LO180000	™ SM680000	[grid] SF230000	[grid] SF390000	[grid] SF040000	Θ GT620000	• SA790000		
-A	○ SM750002	→ SM310000	*	:	J SM040000	Z SP130000	j LJ020000	z LJ010000	è LZ010000	Ü LE130000	¬ LU180000	[grid] SF240000	[grid] SP400000	[grid] SF010000	Ω GO320000	· SD630000		
-B	♂ SM280000	← SM300000	+	;	K SA010000	[SP140000	k LK020000	{ SM060000	Đ LK010000	ø SM110000	½ LD620000	[grid] LO610000	[grid] NF010000	[grid] SF250000	[grid] SF410000	[grid] SF610000	δ GD010000	✓ SA800000
-C	♀ SM290000	L SA420000	,	<	L SA030000	\ LL020000	l SM070000	 LL010000	ð SM130000	£ LD630000	¼ SC020000	[grid] NF040000	[grid] SF260000	[grid] SF420000	[grid] SF570000	∞ SA450000	n LN011000	
-D	♪ SM930000	↔ SM780000	-	=	M SA040000] LM020000	m SM080000	{ LM010000	Þ SM140000	Ø LT640000	i LO620000	[grid] SP030000	[grid] SF270000	[grid] SF430000	[grid] SF580000	ϕ GF010001	² ND021000	
-E	♪ SM910000	▲ SM600000	.	>	N SA050000	^ LN020000	n SD150000	~ LN010000	Ä SD190000	Þts LA180000	« SC060000	[grid] SP170000	[grid] SF280000	[grid] SF440000	[grid] SF590000	ε GE010000	■ SM470000	
-F	☀ SM690000	▼ SV040000	/	?	O SP120000	— SP150000	o LO020000	◊ SP090000	Å LO010000	f SM790000	» LA280000	[grid] SC070000	[grid] SP180000	[grid] SF030000	[grid] SF450000	□ SF600000	∩ SA380000	(RSP) SP300000

Code Page 00861

Code Page 862

HEX DIGITS	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-	
1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-	
-0	► (SP) SM590000	0 SP010000	@ ND100000	P SM050000	' LP020000	p SD130000	ñ LP010000	HX330000 HN010000	í LA110000	á SF140000	□ SF020000	□ SF460000	α GA010000	≡ SA480000			
-1	☺ SS000000	! SM630000	1 SP020000	A ND010000	Q LA020000	a LQ020000	q LA010000	ב LQ010000	ס HB010000	׮ HS010000	װ LI110000	ױ SF150000	ײ SF070000	״ SF470000	׵ LS610000	׶ SA020000	
-2	☻ SS010000	↑ SM760000	" SP040000	2 ND020000	B LB020000	R LR020000	b LB010000	r LR010000	׷ HG010000	׸ HX350000	׹ LO110000	׻ SF160000	׼ SF060000	׽ SF480000	׾ GG020000	׿ SA530000	
-3	♥ SS020000	!! SP330000	# SM010000	3 ND030000	C LC020000	S LS020000	c LC010000	s LS010000	׷ HD010000	׸ HP610000	׹ LU110000	׻ SF110000	׼ SF080000	׽ SF490000	׿ GP010000	׿ SA520000	
-4	♦ SS030000	¶ SM250000	\$ SC030000	4 ND040000	D LD020000	T LT020000	d LD010000	t LT010000	׷ HH010000	׸ HP010000	׹ LN190000	׻ SF090000	׼ SF100000	׽ SF500000	׿ GS020000	׿ SS260000	
-5	♣ SS040000	§ SM240000	% SM020000	5 ND050000	E LE020000	U LU020000	e LE010000	u LU010000	׷ HW010000	׸ HS610000	׹ LN200000	׻ SF190000	׼ SF050000	׽ SF510000	׿ GS010000	׿ SS270000	
-6	♠ SS050000	- SM700000	& SM030000	6 ND060000	F LF020000	V LV020000	f LF010000	v LV010000	׷ HZ010000	׸ HS450000	׹ SM210000	׻ SF200000	׼ SF360000	׽ SF520000	׿ GM010000	׿ SA060000	
-7	● SM570000	↓ SM770000	' SP050000	7 ND070000	G LG020000	W LW020000	g LG010000	w LW010000	׷ HH450000	׸ HQ010000	׹ SM200000	׻ SF210000	׼ SF370000	׽ SF530000	׿ GT010000	׿ SA700000	
-8	█ SM570001	↑ SM320000	(SP060000	8 ND080000	H LH020000	X LX020000	h LH010000	x LX010000	׷ HT450000	׸ HR010000	׹ SP160000	׻ SF220000	׼ SF380000	׽ SF540000	׿ GF020000	׿ SM190000	
-9	○ SM750000	↓ SM330000) SP070000	9 ND090000	I LI020000	Y LY020000	i LI010000	y LY010000	׷ HY010000	׸ HS210000	׹ SM680000	׻ SF230000	׼ SF390000	׽ SF040000	׿ GT620000	׿ SA790000	
-A	◎ SM750002	→ SM310000	* SM040000	:	J SP130000	Z LJ020000	j LJ010000	z LZ010000	׷ HK610000	׸ HT010000	׹ SM660000	׻ SF240000	׼ SF400000	׽ SF010000	׿ GO320000	׿ SD630000	
-B	♂ SM280000	← SM300000	+	;	K SA010000	[LK020000	k SM060000	{ LK010000	׷ SM110000	׸ HK010000	׹ SC040000	׻ NF010000	׼ SF250000	׽ SF410000	׿ SF610000	׿ GD010000	׿ SA800000
-C	♀ SM290000	„ SA420000	, SP080000	< SA030000	L LL020000	\ SM070000	1 LL010000	 SM130000	׷ HL010000	׸ SC020000	׹ NF040000	׻ SF260000	׼ SF420000	׽ SF570000	׿ SA450000	׿ LN011000	
-D	♪ SM930000	↔ SM780000	- SP100000	= SA040000	M LM020000] SM080000	m LM010000	{ SM140000	׷ HM610000	׸ SC050000	׹ SP030000	׻ SF270000	׼ SF430000	׽ SF580000	׿ GF010001	׿ ND021000	
-E	♪ SM910000	▲ SM600000	.	> SP110000	N SA050000	^ LN020000	n SD150000	׷ LN010000	׸ SD190000	׹ HM010000	׻ SC060000	׼ SP170000	׽ SF280000	׿ SF440000	׿ SF590000	׿ GE010000	׿ SM470000
-F	☀ SM690000	▼ SV040000	/	?	O SP150000	— LO020000	o SP090000	׷ LO010000	׸ SM790000	׹ HN610000	׻ SC070000	׼ SP180000	׽ SF030000	׿ SF450000	׿ SF600000	׿ SA380000	׿ (RSP) SP300000

Code Page 00862

Code Page 863

HEX DIGITS 1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0		► (SP) SM590000	0 ND100000 SP010000	@ SM050000 LP020000	P SD130000 LA010000	` LP010000 LC420000	p LE120000 LE140000	Ç SM1650000 SD110000	É SF140000 SF070000	I SF020000 SF470000	□ SF460000 LS610000	α GA010000 SA480000	≡			
-1	☺ SS000000 SM630000	◀ SP020000 ND010000 LA020000	! ND010000 LA010000	A LQ020000 LR010000	Q LA010000 LB010000	a LQ010000 LR010000	q LU170000 LE110000	ü LE160000 LE180000	È LE140000 LA150000	’ SD110000 LU110000	▀ SF150000 SF160000	▀ SF070000 SF060000	▀ SF470000 SF480000	▀ LS610000 GG020000	± SA020000 SA530000	
-2	☻ SS010000 SM760000	↑ SP040000 ND020000 LB020000	” ND020000 LR020000	2 LR020000 LB010000	B LR020000 LB010000	r LB010000 LR010000	é LE110000 LE160000	Ê LO110000 LO150000	ó LO110000 LU110000	▀ SF160000 SF110000	▀ SF060000 SF110000	▀ SF480000 SF490000	▀ GG020000 GP010000	≥ SA520000 SA260000		
-3	♥ SS020000 SP330000	!! SM010000 ND030000 LC020000	# SM010000 ND030000 LC020000	3 S LS020000 LC010000 LS010000	C LC010000 LS010000	s LA150000 LA160000	â LA150000 LA160000	ô LA150000 LE180000	ú LU110000 SD170000	▀ SF110000 SF090000	▀ SF080000 SF090000	▀ SF490000 SF100000	π GP010000 GS020000	≤ SA260000 SS260000		
-4	♦ SS030000 SM250000	¶ SC030000 ND040000 LD020000	\$ ND040000 LD020000	4 D LT020000 LD010000 LT010000	T LT020000 LD010000	d LT010000 LA160000	t LA160000 LE180000	À LE180000 SD170000	Ë SD170000 SD410000	” SF090000 SF190000	” SF100000 SF050000	” SF500000 SF100000	Σ GS020000 GS010000	ƒ SS260000 SS270000		
-5	♣ SS040000 SM240000	§ SM020000 ND050000 LE020000	% SM020000 ND050000 LE020000	5 E LU020000 LE010000 LU010000	E LU020000 LE010000	e LU010000 LC410000	u LU130000 LU130000	à LU130000 LI180000	İ LI180000 SD410000	▀ SF190000 SF190000	▀ SF050000 SF050000	▀ SF100000 SF100000	σ GS010000 GS010000	J SS270000 SS270000		
-6	♠ SS050000 SM700000	- SM030000 ND060000 LF020000	& SM030000 ND060000 LF020000	6 F LV020000 LF010000 LV010000	F LV020000 LF010000	f LV010000 SM250000	v LV010000 LU150000	¶ LU150000 LU150000	û LU150000 ND031000	▀ SF200000 SF200000	▀ SF360000 SF360000	▀ SF520000 SF520000	μ GM010000 GM010000	÷ SA060000 SA060000		
-7	● SM570000 SM770000	↓ SP050000 ND070000 LG020000	’ ND070000 LG020000	7 G LW020000 LG010000 LW010000	G LW020000 LG010000	w LG010000 LC410000	g LW010000 LU130000	ç LU130000 SM150000	ù LU130000 SF210000	▀ SF370000 SF370000	▀ SF530000 SF530000	▀ GT010000 GT010000	τ SA700000 SA700000			
-8	▀ SM570001 SM320000	↑ SP060000 ND080000 LH020000	(ND080000 LH020000	8 H LX020000 LH010000 LX010000	H LX020000 LH010000	x LX010000 LE150000	h LX010000 LE150000	ê LE150000 SC010000	Ѡ LI160000 LI160000	▀ SF220000 SF220000	▀ SF380000 SF380000	▀ SF540000 SF540000	Φ GF020000 GF020000	° SM190000 SM190000		
-9	○ SM750000 SM330000	↓ SP070000 ND090000 LJ020000) ND090000 LJ020000	9 I LY020000 LI010000 LY010000	I LY020000 LI010000	y LY010000 LE170000	i LY010000 LO160000	ë LE170000 LO160000	Ӯ SM680000 SM680000	▀ SF230000 SF230000	▀ SF390000 SF390000	▀ SF040000 SF040000	Θ GT620000 GT620000	• SA790000 SA790000		
-A	○ SM750002 SM310000	→ SM040000 SP130000 LJ020000	* SM040000 SP130000 LJ020000	: J LZ020000 LJ010000 LZ010000	Z LZ020000 LJ010000	j LZ010000 LE130000	z LZ010000 LU180000	è LE130000 SM660000	Ü LU180000 SF240000	▀ SF400000 SF400000	▀ SF010000 SF010000	▀ GO320000 GO320000	Ω SD630000 SD630000			
-B	♂ SM280000 SM300000	← SA010000 SP140000 LK020000	+ SA010000 SP140000 LK020000	;	K SM060000 LK010000 SM110000	[LK010000 LK010000	k SM110000 LI170000	{ LI170000 SC040000	í NF010000 NF010000	¢ SF250000 SF250000	▀ SF410000 SF410000	▀ SF610000 SF610000	δ GD010000 GD010000	√ SA800000 SA800000		
-C	♀ SM290000 SA420000	, SP080000 SA030000 LL020000	< SA030000 LL020000 SM070000	L SM070000 LL010000 SM130000	\ SM070000 LL010000	1 SM130000 LI150000	LI150000 SC020000	î NF040000 NF040000	£ NF050000 NF050000	▀ SF260000 SF260000	▀ SF420000 SF420000	▀ SF570000 SF570000	∞ SA450000 SA450000	n LN011000 LN011000		
-D	♪ SM930000 SM780000	↔ SP100000 SA040000 LM020000	- SA040000 LM020000	= M SM080000 LM010000 SM140000] SM080000 LM010000	m SM140000 SM100000	} SM140000 SM100000	= SM100000 LU140000	Ӯ LU140000 NF050000	▀ SF270000 SF270000	▀ SF430000 SF430000	▀ SF580000 SF580000	ϕ GF010001 GF010001	2 ND021000 ND021000		
-E	♪ SM910000 SM600000	▲ SP110000 SA050000 LN020000	. SA050000 LN020000	> N SD150000 LN010000 SD190000	^ LN010000 LN010000	n SD150000 LA140000	~ SD190000 LU160000	À LA140000 LU160000	Ӯ LU160000 SP170000	▀ SF280000 SF280000	▀ SF440000 SF440000	▀ SF590000 SF590000	ε GE010000 GE010000	▀ SM470000 SM470000		
-F	☀ SM690000 SV040000	▼ SP120000 SP150000 LO020000	/ SP120000 SP090000	? O LO010000 LO010000 SM790000	- LO010000 LO010000	o SM240000 SM240000	⌂ SM790000 SC070000	§ SP180000 SF030000	f SF030000 SF450000	▀ SF450000 SF600000	▀ SF600000 SF600000	▀ SA380000 SA380000	∩ (RSP) SP300000 SP300000			

Code Page 00863

Code Page 864

HEX DIGITS 1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0	▶ (SP) SM590000	0 (SP) SP010000	@ (SP) ND100000	P (SP) SM050000	‘ (SP) LP020000	پ (SP) SD130000	پ (SP) LP010000	۰ (SP) SM190000	ۢ (SP) GB010000	ۢ (SP) SP300000	ۣ (SP) ND100001	؋ (SP) SC040000	، (SP) AD470000	؍ (SP) SM860000	؎ (SP) AX100004	
-1	☺ (SP) SS000000	! (SP) SM630000	1 (SP) SP020000	A (SP) ND010000	Q (SP) LA020000	ا (SP) LQ020000	q (SP) LA010000	ۧ (SP) LQ010000	ۧ (SP) SD630000	ۧ (SP) SA450000	ۧ (SP) SP320000	ۧ (SP) ND010001	ۧ (SP) AX300000	ۧ (SP) AR010000	ۧ (SP) AF010003	ۧ (SP) AX100000
-2	♪ (SP) SM930000	↑ (SP) SM760000	“ (SP) SP040000	2 (SP) ND020000	B (SP) LB020000	R (SP) LR020000	b (SP) LB010000	r (SP) LR010000	۪ (SP) SA790000	۪ (SP) GF010001	۪ (SP) AA210002	۪ (SP) ND020001	۪ (SP) AA210000	۪ (SP) AZ010000	۪ (SP) AQ010003	۪ (SP) AN010000
-3	♪ (SP) SM910000	!! (SP) SP330000	# (SP) SM010000	3 (SP) ND030000	C (SP) LC020000	S (SP) LS020000	c (SP) LC010000	s (SP) LS010000	√ (SP) SA800000	± (SP) SA020000	£ (SP) SC020000	ۢ (SP) ND030001	ۢ (SP) AA310000	ۢ (SP) AS010003	ۢ (SP) AK010003	ۢ (SP) AH010000
-4	☀ (SP) SM690000	¶ (SP) SM250000	\$ (SP) SC030000	4 (SP) ND040000	D (SP) LD020000	T (SP) LT020000	d (SP) LD010000	t (SP) LT010000	۪ (SP) SF150000	۪ (SP) NF010000	۪ (SP) SC010000	۪ (SP) ND040001	۪ (SP) AW310000	۪ (SP) AS230003	۪ (SP) AL010003	۪ (SP) AH010004
-5	≡ (SP) SF430000	§ (SP) SM240000	% (SP) SM020007	5 (SP) ND050000	E (SP) LE020000	U (SP) LU020000	e (SP) LE010000	u (SP) LU010000	۪ (SP) SF100000	۪ (SP) NF040000	۪ (SP) AA310002	۪ (SP) ND050001	۪ (SP) AC470002	۪ (SP) AS450003	۪ (SP) AM010003	۪ (SP) AA020002
-6	□□□ (SP) SF240000	— (SP) SM700000	& (SP) SM030000	6 (SP) ND060000	F (SP) LF020000	V (SP) LV020000	f (SP) LF010000	v (SP) LV010000	۪ (SP) SF110000	۪ (SP) SA700000	۪ (SP) ND060001	۪ (SP) AY310000	۪ (SP) AD450003	۪ (SP) AN010003	۪ (SP) AY010002	
-7	☰ (SP) SF400000	↑ (SP) SM770000	‘ (SP) SP050000	7 (SP) ND070000	G (SP) LG020000	W (SP) LW020000	g (SP) LG010000	w (SP) LW010000	۪ (SP) SF050000	۪ (SP) SP170000	۪ (SP) SC200000	۪ (SP) ND070001	۪ (SP) AA010000	۪ (SP) AT450000	۪ (SP) AH010003	۪ (SP) AG310004
-8	☰ (SP) SF230000	↑ (SP) SM320000	((SP) SP060000	8 (SP) ND080000	H (SP) LH020000	X (SP) LX020000	h (SP) LH010000	x (SP) LX010000	۪ (SP) SF090000	۪ (SP) SP180000	۪ (SP) AA010002	۪ (SP) ND080001	۪ (SP) AB010003	۪ (SP) AZ450000	۪ (SP) AW010000	۪ (SP) AQ010000
-9	☰ (SP) SF410000	↓ (SP) SM330000) (SP) SP070000	9 (SP) ND090000	I (SP) LI020000	Y (SP) LY020000	i (SP) LI010000	y (SP) LY010000	۪ (SP) SF060000	۪ (SP) AL320000	۪ (SP) AB010000	۪ (SP) ND090001	۪ (SP) AT020000	۪ (SP) AC470003	۪ (SP) AA020000	۪ (SP) AL220000
-A	☰ (SP) SF420000	→ (SP) SM310000	* (SP) SM040007	:	J (SP) LJ020000	Z (SP) LZ020000	j (SP) LJ010000	z (SP) LZ010000	۪ (SP) SF080000	۪ (SP) AL320003	۪ (SP) AT010000	۪ (SP) AF010000	۪ (SP) AT010003	۪ (SP) AG310003	۪ (SP) AY010003	۪ (SP) AL220003
-B	☰ (SP) SF400000	← (SP) SM300000	+ (SP) SA010000	;	K (SP) LK020000	[(SP) SM060000	k (SP) LK010000	{ (SP) SM110000	۪ (SP) SF070000	۪ (SP) AT470000	۪ (SP) SP140007	۪ (SP) AT470003	۪ (SP) SM650000	۪ (SP) AD450000	۪ (SP) AL010000	
-C	☰ (SP) SF250000	، (SP) SA420000	< (SP) SP080000	L (SP) SA030000	\ (SP) LL020000	۠ (SP) SM070000	۠ (SP) LL010000	۠ (SP) SM130000	۪ (SP) SF030000	۪ (SP) SP080007	۪ (SP) AS010000	۪ (SP) AG230003	۪ (SP) SM660000	۪ (SP) AC470004	۪ (SP) AK010000	
-D	☰ (SP) SF390000	↔ (SP) SM780000	- (SP) SP100000	= (SP) SA040000	M (SP) LM020000] (SP) SM080000	m (SP) LM010000	} (SP) SM140000	۪ (SP) SF010000	۪ (SP) AL020000	۪ (SP) AG230000	۪ (SP) AS230000	۪ (SP) AH450003	۪ (SP) SA060000	۪ (SP) AG310002	۪ (SP) AY010000
-E	☰ (SP) SF380000	▲ (SP) SM600000	. (SP) SP110000	> (SP) SA050000	N (SP) LN020000	^ (SP) SD150000	n (SP) LN010000	~ (SP) SD190000	۪ (SP) SF020000	۪ (SP) AL020003	۪ (SP) AH450000	۪ (SP) AS450000	۪ (SP) AH470003	۪ (SP) SA070000	۪ (SP) AG310000	۪ (SP) SM470000
-F	☰ (SP) SF260000	▼ (SP) SV040000	/ (SP) SP120000	? (SP) SP150000	O (SP) LO020000	_ (SP) SP090000	o (SP) LO010000	□ (SP) SM790000	۪ (SP) SF040000	۪ (SP) SM870000	۪ (SP) AH470000	۪ (SP) SP150007	۪ (SP) AD010000	۪ (SP) AC470000	۪ (SP) AM010000	

Code Page 865

HEX DIGITS 1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0		► (SP) SM590000	0 ND100000 SP010000	@ SM050000 LP020000	P SD130000 LA010000	` LP010000 LC420000	p LE120000 LA110000	Ç SF140000 SF020000	É SF020000 SF460000	á SF140000 GA010000						
-1	☺ SS000000	◀ SM630000	! ND010000 SP020000	1 LA020000 LQ020000	A Q LA010000 LQ010000	a LQ010000 LU170000	q LU170000 LA510000	ü LA510000 LI110000	æ SF150000 SF070000	í SF150000 SF470000				ß LS610000 SA020000		±
-2	☻ SS010000	↑ SM760000	" ND020000 SP040000	2 LB020000 LR020000	B R LB010000 LR010000	b LB010000 LE110000	r LR010000 LA520000	é LE110000 LO110000	Æ SF160000 SF060000	ó SF160000 SF480000				Γ GG020000 SA530000		≥
-3	♥ SS020000	!! SP330000	# SM010000 ND030000	3 LC020000 LS020000	C S LC010000 LS010000	c LS010000 LA150000	s LA150000 LO150000	â LO150000 LU110000	ô SF110000 SF080000	ú SF110000 SF490000				π GP010000 SA520000		≤
-4	♦ SS030000	¶ SM250000	\$ SC030000 ND040000	4 LD020000 LT020000	D T LD010000 LT010000	d LD010000 LA170000	t LT010000 LO170000	ä LO170000 LN190000	ö SF090000 SF100000	ñ SF090000 SF500000				Σ GS020000 SS260000		ƒ
-5	♣ SS040000	§ SM240000	% SM020000 ND050000	5 LE020000 LU020000	E U LE010000 LU010000	e LU010000 LA130000	u LU010000 LO130000	à LA130000 LO130000	ò SF190000 SF050000	ñ SF190000 SF510000				σ GS010000 SS270000		J
-6	♠ SS050000	- SM700000	& SM030000 ND060000	6 LF020000 LV020000	F V LF010000 LV010000	f LV010000 LA270000	v LF010000 LU150000	å LU150000 SM210000	û SF200000 SF360000	ä SF200000 SF520000				μ GM010000 SA060000		÷
-7	● SM570000	↓ SM770000	' SP050000 ND070000	7 LG020000 LW020000	G W LG010000 LW010000	g LG010000 LC410000	w LG010000 LU130000	ç LU130000 SM200000	ù SF210000 SF370000	ø SF210000 SF530000				τ GT010000 SA700000		≈
-8	▀ SM570001	↑ SM320000	(SP060000 ND080000	8 LH020000 LX020000	H X LH010000 LX010000	h LH010000 LE150000	x LX010000 LY170000	ê LE150000 SP160000	ÿ SF220000 SF380000	ি SF220000 SF540000				Φ GF020000 SM190000		°
-9	○ SM750000	↓ SM330000) SP070000 ND090000	9 LJ020000 LY020000	I Y LJ010000 LY010000	i LJ010000 LY010000	y LY010000 LE170000	ë LE170000 LO180000	Ö SF230000 SF390000	— SF230000 SF400000				Θ GT620000 SA790000		•
-A	○ SM750002	→ SM310000	* SM040000 SP130000	: LJ020000 LK020000	J Z LJ010000 LK010000	j LZ020000 LK010000	z LJ010000 SM110000	è LE130000 LU180000	Ü SF240000 SF400000	— SF240000 SF010000				Ω GO320000 SD630000		·
-B	♂ SM280000	← SM300000	+ SA010000 SP140000	;	K LK020000 SM060000	[LK010000 SM060000	{} LK010000 SM110000	í LI170000 LO610000	ø NF010000 NF020000	½ SF250000 SF410000				δ SF610000 GD010000		√ SA800000
-C	♀ SM290000	L SA420000 SP080000	,	< SA030000 LL020000	L SM070000 SM130000	\ LL010000 LI150000	1 SM130000 LI150000	LI150000 SC020000	£ NF040000 NF050000	¼ SF260000 SF420000				∞ SF570000 SA450000		n LN011000
-D	♪ SM930000	↔ SM780000 SP100000	- SA040000 LM020000	= M SM080000 SM140000	M LM010000 SM140000	m LM010000 LI130000	} LM010000 LI130000	ì LI130000 LO620000	Ø SP030000 SP040000	í SF270000 SF430000				ϕ SF580000 GF010001		2 ND021000
-E	♪ SM910000	▲ SP110000 SA050000	> LN020000 SD150000	N ^ LN010000 SD150000	n LN010000 SD190000	~ LN010000 LA180000	Ä SC060000 SC070000	Pts SC060000 SC070000	« SP170000 SF280000	» SF280000 SF440000				ε SF590000 GE010000		■ SM470000
-F	☀ SM690000	▼ SP120000 SP150000	/ LO020000 LO010000	? O SP090000 SM790000	O LO010000 LA280000	◊ LO010000 SC010000	Å SC010000 SF030000	f SC010000 SF030000	¤ SF030000 SF450000	□ SF030000 SF600000				∩ SA380000 SP300000		(RSP)

Code Page 00865

Code Page 866

HEX DIGITS 1ST ↗ 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0 SM590000 SP010000 ND100000 SM050000 LP020000 SD130000 LP010000 KA020000 KR020000 KA010000 SF140000 SF020000 SF460000 KR010000 KE180000	► (SP) 0 @ P ` p A P a															Ё
-1 SS000000 SM630000 SP020000 ND010000 LA020000 LQ020000 LA010000 LQ010000 KB020000 KS020000 KB010000 SF150000 SF070000 SF470000 KS010000 KE170000	☺ ← ! 1 A Q a q Б С б															с ё
-2 SS010000 SM760000 SP040000 ND020000 LB020000 LR020000 LB010000 LR010000 KV020000 KT020000 KV010000 SF160000 SF060000 SF480000 KT010000 KE160000	☻ ⇧ " 2 B R b r В Т в															т €
-3 SS020000 SP330000 SM010000 ND030000 LC020000 LS020000 LC010000 LS010000 KG020000 KU020000 KG010000 SF110000 SF080000 SF490000 KU010000 KE150000	♥ !! # 3 C S c s Г У г															у €
-4 SS030000 SM250000 SC030000 ND040000 LD020000 LT020000 LD010000 LT010000 KD020000 KF020000 KD010000 SF090000 SF100000 SF500000 KF010000 KI180000	♦ ¶ \$ 4 D T d t Д Ф д															Ф І
-5 SS040000 SM240000 SM020000 ND050000 LE020000 LU020000 LE010000 LU010000 KE020000 KH020000 KE010000 SF190000 SF050000 SF510000 KH010000 KI170000	♣ § % 5 E U e u Е Х е															х ї
-6 SS050000 SM700000 SM030000 ND060000 LF020000 LV020000 LF010000 LV010000 KZ220000 KC020000 KZ210000 SF200000 SF360000 SF520000 KC010000 KU240000	♠ - & 6 F V f v Ж Ц ж															Ц ў
-7 SM570000 SM770000 SP050000 ND070000 LG020000 LW020000 LG010000 LW010000 KZ020000 KC220000 KZ010000 SF210000 SF370000 SF530000 KC210000 KU230000	● ⇧ ' 7 G W g w З Ч з															Ч ў
-8 SM570001 SM320000 SP060000 ND080000 LH020000 LX020000 LH010000 LX010000 KI020000 KS220000 KI010000 SF220000 SF380000 SF540000 KS210000 SM190000	█ ↑ (8 H X h x И Ш и															о
-9 SM750000 SM330000 SP070000 ND090000 LI020000 LY020000 LI010000 LY010000 KJ120000 KS160000 KJ110000 SF230000 SF390000 SF400000 KS150000 SA790000	○ ↓) 9 I Y i y Й Щ й															•
-A SM750002 SM310000 SM040000 SP130000 LJ020000 LZ020000 LJ010000 LZ010000 KK020000 KU220000 KK010000 SF240000 SF400000 SF010000 KU210000 SD630000	█ → * : J Z j z К Ъ к															ъ •
-B SM280000 SM300000 SA010000 SP140000 LK020000 SM060000 LK010000 SM110000 KL020000 KY020000 KL010000 SF250000 SF410000 SF610000 KY010000 SA800000	♂ ← + ; K [k { Л Ы л															ы ✓
-C SM290000 SA420000 SP080000 SA030000 LL020000 SM070000 LL010000 SM130000 KM020000 KX120000 KM010000 SF260000 SF420000 SF570000 KX110000 SM000000	♀ „ , < L \ 1 М Ъ м															ъ №
-D SM300000 SM780000 SP100000 SA040000 LM020000 SM080000 LM010000 SM140000 KN020000 KE140000 KN010000 SF270000 SF430000 SF580000 KE130000 SC010000	♪ ↔ - = М] m } Н Э н															э ☺
-E SM910000 SM600000 SP110000 SA050000 LN020000 SD150000 LN010000 SD190000 KO020000 KU160000 KO010000 SF280000 SF440000 SF590000 KU150000 SM470000	♪ ▲ . > N ^ n ~ О Ю о															ю ■
-F SM690000 SV040000 SP120000 SP150000 LO020000 SP090000 LO010000 SM790000 KP020000 KA160000 KP010000 SF030000 SF450000 SF600000 KA150000 SP300000	☀ ▼ / ? O — o □ П я п															я (RSP)

Code Page 867

HEX DIGITS 1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0	◀ (SP) SM590000	0 SP010000	@ ND100000	P SM050000	‘ LP020000	p SD130000	‘ LP010000	׮ HX330000	װ HN010000	ױ (LRM) SP5500Z0	ײ SF140000	׳ SF020000	״ SF460000	׵ GA010000	׶ SA480000	
-1	😊 SS000000	◀ ! SM630000	1 ND010000	A LA020000	Q LO020000	a LA010000	q LQ010000	׮ HB010000	װ HS010000	ױ (LRL) SP5600Z0	ײ SF150000	׳ SF070000	״ SF470000	׵ LS610000	׶ SA020000	
-2	😊 SS010000	↕ “ SM760000	2 ND020000	B LB020000	R LR020000	b LB010000	r LR010000	׮ HG010000	װ HX350000	ױ (LRE) SP5700Z0	ײ SF160000	׳ SF060000	״ SF480000	׵ GG020000	׶ SA530000	
-3	♥ SS020000	!! # SP330000	3 ND030000	C LC020000	S LS020000	c LC010000	s LS010000	׮ HD010000	װ HP610000	ױ (RLE) SP5800Z0	ײ SF110000	׳ SF080000	״ SF490000	׵ GP010000	׶ SA520000	
-4	♦ SS030000	¶ \$ SM250000	4 ND040000	D LD020000	T LT020000	d LD010000	t LT010000	׮ HH010000	װ HP010000	ױ (RLO) SP6000Z0	ײ SF090000	׳ SF100000	״ SF500000	׵ GS020000	׶ SS260000	
-5	♣ SS040000	§ % SM240000	5 ND050000	E LE020000	U LU020000	e LE010000	u LU010000	׮ HW010000	װ HS610000	ױ (RLO) SP6100Z0	ײ SF190000	׳ SF050000	״ SF510000	׵ GS010000	׶ SS270000	
-6	♠ SS050000	— & SM700000	6 ND060000	F LF020000	V LV020000	f LF010000	v LV010000	׮ HZ010000	װ HS450000	ױ (PDF) SP5900Z0	ײ SF200000	׳ SF360000	״ SF520000	׵ GM010000	׶ SA060000	
-7	● SM570000	↕ ‘ SM770000	7 ND070000	G LG020000	W LW020000	g LG010000	w LW010000	׮ HH450000	װ HQ010000	ױ (RLO) SF210000	ײ SF370000	׳ SF530000	״ GT010000	׵ SA700000		
-8	█ SM570001	↑ (SM320000	8 ND080000	H LH020000	X LX020000	h LH010000	x LX010000	׮ HT450000	װ HR010000	ױ (RLO) SF220000	ײ SF380000	׳ SF540000	״ GF020000	׵ SM190000		
-9	○ SM750000	↓) SM330000	9 ND090000	I LI020000	Y LY020000	i LI010000	y LY010000	׮ HY010000	װ HS210000	ױ (RLO) SM680000	ײ SF230000	׳ SF390000	״ SF040000	׵ GT620000	׶ SA790000	
-A	⦿ SM750002	→ * SM310000	: SM040000	J SP130000	Z LJ020000	j LJ010000	z LZ010000	׮ HK610000	װ HT010000	ױ (RLO) SM660000	ײ SF240000	׳ SF400000	״ SF010000	׵ GO320000	׶ SD630000	
-B	♂ SM280000	← + SM300000	; SA010000	K SP140000	[LK020000	k LK010000	{ SM060000	׮ SM110000	װ HK010000	ױ (RLO) SC040000	ײ NF010000	׳ SF250000	״ SF410000	׵ SF610000	׶ GD010000	׶ SA800000
-C	♀ SM290000	L , SA420000	< L SP080000	A030000	\ L LL020000	׮ SM070000	װ LL010000	׮ SM130000	װ HL010000	ױ (RLO) SC020000	ײ NF040000	׳ SF260000	״ SF420000	׵ SF570000	׶ SA450000	׶ LN011000
-D	♪ SM930000	↔ - SM780000	= M SP100000	A040000] m LM020000	׮ SM080000	װ LM010000	׮ SM140000	װ HM610000	ױ (RLO) SC050000	ײ SC200000	׳ SF270000	״ SF430000	׵ SF580000	׶ GF010001	׶ ND021000
-E	♪ SM910000	▲ . SM600000	> N SP110000	A050000	^ n LN020000	׮ SD150000	װ LN010000	׮ SD190000	װ HM010000	ױ (RLO) SP170000	ײ SP180000	׳ SF280000	״ SF440000	׵ SF590000	׶ GE010000	׶ SM470000
-F	☀ SM690000	▼ / SV040000	? O SP120000	A050000	— o LO020000	׮ SP090000	װ LO010000	׮ SM790000	װ HN610000	ױ (RLO) SC170000	ײ SP180000	׳ SF030000	״ SF450000	׵ SF600000	׶ SA380000	׶ SP300000

Figure 83. Code page 867 chart

Code Page 869

HEX DIGITS 1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0		►	(SP)	0	@	P	'	p		‘I	‘i	[grid]	[grid]	T	ζ	(SHY)
-1	☺	◀	!	1	A	Q	a	q		‘I	‘i	[grid]	[grid]	Y	η	±
-2	☻	↑	"	2	B	R	b	r		‘O	ó	[grid]	[grid]	Φ	θ	v
-3	♥	!!	#	3	C	S	c	s			ú	[grid]	[grid]	X	ι	φ
-4	♦	¶	\$	4	D	T	d	t			A	[grid]	[grid]	Ψ	κ	χ
-5	♣	§	%	5	E	U	e	u		‘Y	B	K	[grid]	Ω	λ	§
-6	♠	—	&	6	F	V	f	v	‘A	‘Y	Γ	Λ	Π	α	μ	ψ
-7	•	↓	'	7	G	W	g	w	€	©	Δ	M	P	β	v	'
-8	█	↑	(8	H	X	h	x	·	‘Ω	E	N	[grid]	γ	ξ	°
-9	○	↓)	9	I	Y	i	y	¬	²	Z	[grid]	[grid]	[grid]	o	..
-A	○	→	*	:	J	Z	j	z		³	H	[grid]	[grid]	[grid]	π	ω
-B	♂	←	+	;	K	[k	{	‘	á	½	[grid]	[grid]	[grid]	ρ	ü
-C	♀	„	,	<	L	\	l		,	£	Θ	[grid]	[grid]	[grid]	σ	ő
-D	♪	↔	-	=	M]	m	}	‘E	‘ɛ	I	Ξ	[grid]	δ	ς	ø
-E	♪	▲	.	>	N	^	n	~	—	‘í	«	O	[grid]	ε	τ	■
-F	☼	▼	/	?	O	_	o	◊	‘H	‘i	»	[grid]	Σ	[grid]	'	(RSP)

Code Page 00869

Code Page 872

HEX DIGITS 1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0 SM590000	▶ (SP) SP010000	0 ND100000	@ SM050000	P LP020000	` SD130000	p LP010000	Ђ KD610000	љ KL410000	а KA010000	⠠ SF140000	⠇ SF020000					
-1 SS000000	😊 SM630000	! SP020000	1 ND010000	А LA020000	Q LQ020000	а LA010000	҆ LQ010000	Ҋ KD620000	Љ KL420000	А KA020000	⠠ SF150000	⠇ SF070000				
-2 SS010000	☻ SM760000	↑ SP040000	" ND020000	Б LB020000	R LR020000	б LB010000	г LR010000	Ҋ KG110000	њ KN110000	б KB010000	⠠ SF160000	⠇ SF060000				
-3 SS020000	♥ SP330000	!! SM010000	# ND030000	С LC020000	S LS020000	с LC010000	ѕ LS010000	Ѓ KG120000	Њ KN120000	Б KB020000	⠠ SF110000	⠇ SF080000	⠇ KM020000	⠇ KS010000	⠇ KZ010000	⠇ KZ010000
-4 SS030000	♦ SM250000	¶ SC030000	4 ND040000	Д LD020000	Т LT020000	д LD010000	т LT010000	ě KE170000	Ћ KC110000	Ц KC010000	⠠ SF090000	⠇ SF100000				
-5 SS040000	♣ SM240000	§ SM020000	% ND050000	Е LE020000	У LU020000	е LE010000	у LU010000	Ё KE180000	Ћ KC120000	Ц KC020000	⠠ SF050000	⠇ KN020000	⠇ KT010000	⠇ KS210000	⠇ KS210000	⠇ KS210000
-6 SS050000	♠ SM700000	— SM030000	& ND060000	6 LF020000	V LV020000	f LF010000	v LV010000	€ KE150000	ќ KK110000	đ KD010000	⠠ KK010000					
-7 SM570000	● SM770000	↑ SP050000	' ND070000	G LG020000	W LW020000	g LG010000	w LW010000	€ KE160000	Ќ KK120000	đ KD020000	⠠ KI010000	⠠ KK020000	⠠ KO020000	⠠ KU010000	⠠ KE130000	⠠ KE130000
-8 SM570001	█ SM320000	↑ SP060000	(ND080000	H LH020000	X LX020000	h LH010000	x LX010000	ѕ KZ150000	ѹ KU230000	е KE010000	⠠ KI020000	⠠ SF380000	⠠ KP010000	⠠ KU020000	⠠ KE140000	⠠ KE140000
-9 SM750000	○ SM330000	↓ SP070000) ND090000	I LI020000	Y LY020000	i LI010000	y LY010000	Ѕ KZ160000	Ӯ KU240000	Е KE020000	⠠ SF230000	⠠ SF390000	⠠ SF040000	⠠ KZ210000	⠠ KS150000	⠠ KS150000
-A SM750002	⦿ SM310000	→ SM040000	* SP130000	: LJ020000	Z LZ020000	j LJ010000	z LZ010000	и KI110000	џ KG210000	Ѡ KF010000	⠠ SF240000	⠠ SF010000	⠠ SF010000	⠠ KZ220000	⠠ KS160000	⠠ KS160000
-B SM280000	♂ SM300000	← SA010000	+ SP140000	;	K LK020000	[SM060000	k LK010000	{ SM110000	I KI120000	Џ KG220000	⠠ SF020000	⠠ SF250000	⠠ SF410000	⠠ SF610000	⠠ KV010000	⠠ KC210000
-C SM290000	♀ SA420000	, SP080000	SA030000	L L020000	＼ SM070000	l LL010000	SM130000	ї KI170000	ю KU150000	ѓ KG010000	⠠ SF260000	⠠ SF420000	⠠ SF570000	⠠ KV020000	⠠ KC220000	⠠ KC220000
-D SM930000	♪ SM780000	↔ SP100000	SA040000	= LM020000] SM080000	m LM010000	} SM140000	Ӯ KI180000	Ѡ KU160000	Ѓ KG020000	⠠ KJ110000	⠠ SF430000	⠠ KP020000	⠠ KX110000	⠠ SM240000	⠠ SM240000
-E SM910000	♪ SM600000	. SP110000	SA050000	> LN020000	^ SD150000	n LN010000	~ SD190000	ј KJ010000	Ѡ KU210000	« SP170000	⠠ KJ120000	⠠ SF440000	⠠ KA150000	⠠ KX120000	⠠ SM470000	⠠ SM470000
-F SM690000	☀ SV040000	/ SP120000	? SP150000	— LO020000	o SP090000	⠇ LO010000	⠠ SM790000	Ј KJ020000	Ђ KU220000	» SP180000	⠠ SF030000	⠠ SC200000	⠠ SF600000	⠠ SM000000	⠠ (RSP)	⠠ SM000000

Figure 84. Code page 872 chart

Code Page 874

HEX DIGITS	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0			(SP)	0	@	P	'	p			'	ଶ୍ରୀ	କୀ	ଙ୍କ	ତ୍ରୀ	ଓ
-1			!	1	A	Q	a	q			ଗୀ	ମୀ	ମୁ	ବୀ	ଟୀ	ରୀ
-2			"	2	B	R	b	r			ଅୟ	ଯୀ	ଗୀ	ରୀ	କୀ	କୁ
-3			#	3	C	S	c	s			ଞ୍ଚ	ଣ୍ଣ	ର୍ମ	ନ୍ଦୀ	ନ୍ଦୀ	ନ୍ଦୀ
-4			\$	4	D	T	d	t			କ୍ରୀ	ତ୍ରୀ	ଟୀ	ବୀ	କୀ	କୁ
-5			%	5	E	U	e	u			କ୍ରୀ	ତ୍ରୀ	ଲୀ	ପୀ	ନ୍ଦୀ	ନ୍ଦୀ
-6			&	6	F	V	f	v			ଖ୍ରୀ	ଣ୍ଣ	କୀ	ବୀ	ନ୍ଦୀ	ନ୍ଦୀ
-7			'	7	G	W	g	w			ଙ୍ଗ	ଗୀ	ରୀ	କୀ	କୁ	କୁ
-8			(8	H	X	h	x			ଜୀ	କ୍ରୀ	ଶୀ	ବୀ	ନ୍ଦୀ	ନ୍ଦୀ
-9)	9	I	Y	i	y			ଇୟ	ନ୍ଦୀ	ଶୀ	ବୀ	ନ୍ଦୀ	ନ୍ଦୀ
-A			*	:	J	Z	j	z			ଅୟ	ପୀ	ସୀ	ବୀ	ନ୍ଦୀ	ନ୍ଦୀ
-B			+	;	K	[k	{			ଅୟ	ପୀ	ହୀ	ବୀ	କୀ	କୁ
-C			,	<	L	\	l				ଅୟ	ପୀ	ପୀ	ବୀ	ନ୍ଦୀ	ନ୍ଦୀ
-D			-	=	M]	m	}			ଅୟ	ପୀ	ପୀ	ବୀ	ନ୍ଦୀ	ନ୍ଦୀ
-E			.	>	N	^	n	~			ଅୟ	ପୀ	ପୀ	ବୀ	ନ୍ଦୀ	ନ୍ଦୀ
-F			/	?	O	_	o				ଅୟ	ପୀ	ପୀ	ବୀ	ନ୍ଦୀ	ନ୍ଦୀ

Figure 85. Code page 874 chart

Code Page 1116

HEX DIGITS 1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-		
-0		► (SP) SM590000	0 SP010000	@ ND100000	P SM050000	' LP020000	p SD130000	lc LP010000	ç LC420000	é LE120000	á LA110000	[grid] SF140000	[grid] SF020000	š LS210000	ó LO120000	(SHY) SP320000		
-1	☺ SS000000	◀ SM630000	! SP020000	1 ND010000	A LA020000	Q LQ020000	a LA010000	q LQ010000	ü LU170000	æ LA510000	í LI110000	[grid] SF150000	[grid] SF070000	š LS220000	ß LS610000	± SA020000		
-2	☻ SS010000	↓ SM760000	" SP040000	2 ND020000	B LB020000	R LR020000	b LB010000	r LR010000	é LE110000	æ LA520000	ó LO110000	[grid] SF160000	[grid] SF060000	ê LE160000	ô LO160000	= SM100000		
-3	♥ SS020000	!! SP330000	# SM010000	3 ND030000	C LC020000	S LS020000	c LC010000	s LS010000	â LA150000	ô LO150000	ú LU110000	[grid] SF110000	[grid] SF080000	ë LE180000	ò LO140000	¾ NF050000		
-4	♦ SS030000	¶ SM250000	\$ SC030000	4 ND040000	D LD020000	T LT020000	d LD010000	t LT010000	ä LA170000	ö LO170000	ñ LN190000	[grid] SF090000	[grid] SF100000	è LE140000	õ LO190000	¶ SM250000		
-5	♣ SS040000	§ SM240000	% SM020000	5 ND050000	E LE020000	U LU020000	e LE010000	u LU010000	à LA130000	ò LO130000	ñ LN200000	á LA120000	[grid] SF050000	í LI610000	õ LO200000	§ SM240000		
-6	♠ SS050000	- SM700000	& SM030000	6 ND060000	F LF020000	V LV020000	f LF010000	v LV010000	å LA270000	û LU150000	¤ SM210000	â LA160000	ã LA190000	í LI120000	µ SM170000	÷ SA060000		
-7	● SM570000	↓ SM770000	' SP050000	7 ND070000	G LG020000	W LW020000	g LG010000	w LW010000	ç LC410000	ù LU130000	º SM200000	â LA140000	ã LA200000	í LI160000	ž LZ210000	* SD410000		
-8	█ SM570001	↑ SM320000	(SP060000	8 ND080000	H LH020000	X LX020000	h LH010000	x LX010000	ê LE150000	ÿ LY170000	¸ SP160000	circled C SM520000	[grid] SF380000	ï LI180000	ž LZ220000	º SM190000		
-9	○ SM750000	↓ SM330000) SP070000	9 ND090000	I LI020000	Y LY020000	i LI010000	y LY010000	ë LE170000	ö LO180000	® SM530000	[grid] SF230000	[grid] SF390000	ú SF040000	º LU120000	º SD170000		
-A	○ SM750002	→ SM310000	*	: SM040000	J SP130000	Z LJ020000	j LJ010000	z LZ010000	è LE130000	Ü LU180000	¬ SM660000	[grid] SF240000	[grid] SF400000	û SF010000	º LU160000	º SD630000		
-B	♂ SM280000	← SM300000	+	; SA010000	K SP140000	[LK020000	k SM060000	{ LK010000	í SM110000	ø LI170000	½ LO610000	[grid] SF250000	[grid] SF410000	û SF610000	ú LU140000	º ND011000		
-C	♀ SM290000	L SA420000	,	< SP080000	L SA030000	\ LL020000	l SM070000	 LL010000	í SM130000	î LI150000	£ SC020000	¼ NF040000	[grid] SF260000	[grid] SF420000	ý SF570000	º LY110000	º ND031000	
-D	♪ SM930000	↔ SM780000	-	= SP100000	M SA040000] LM020000	m SM080000	{ LM010000	ì SM140000	ø LI130000	Ø LO620000	i SP030000	[grid] SF040000	[grid] SF430000	ý SM650000	º LY120000	º ND021000	
-E	♪ SM910000	▲ SM600000	.	> SP110000	N SA050000	^ LN020000	n SD150000	~ LN010000	ää SD190000	ä LA180000	× SA070000	« SP170000	[grid] SC050000	[grid] SF440000	í LI140000	º SM150000	º SM470000	
-F	☀ SM690000	▼ SP040000	/	?	O SP120000	— SP150000	o LO020000	◊ SP090000	å LO010000	ƒ SM790000	» LA280000	[grid] SC070000	[grid] SP180000	[grid] SF030000	[grid] SC010000	º SF600000	º SD110000	(RSP) SP300000

Figure 86. Code page 1116 chart

Code Page 1117

HEX DIGITS	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-	
1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-	
-0	▶ (SP) SM590000	0 ND100000 SP010000	@ SM050000 LP020000	P SD130000 LP010000	` LC120000 LC100000	Ć LE120000 LU170000	É LA320000 LZ300000	Ā SF140000 LI320000	܍ SF020000 SF150000	܍ SF460000 SF070000	܍ SF470000 SF060000	܍ SF480000 SF160000	܍ SF490000 SF080000	܍ SF500000 SF100000	܍ SF510000 SF090000	܍ SF520000 SF100000	܍ SP320000 LO120000
-1	😊 SS000000	! SM630000 SP020000	1 ND010000 LA020000	Q LQ020000 LA010000	a LQ010000 LU170000	q LU170000 LZ300000	ü LZ300000 LI110000	ż LO110000 LU310000	í LO110000 LU430000	܍ SF150000 SF110000	܍ SF070000 SF060000	܍ SF470000 SF480000	܍ SF490000 SF080000	܍ SF500000 SF100000	܍ SF510000 SF090000	܍ SA020000 LS610000	
-2	😊 SS010000	↔ SM760000 SP040000	" ND020000 LB020000	R LR020000 LB010000	b LR010000 LE110000	r LE110000 LZ290000	é LZ290000 LO110000	ż LO110000 LU430000	ó LU430000 SF110000	܍ SF160000 SF100000	܍ SF060000 SF080000	܍ SF480000 SF490000	܍ SF490000 SF080000	܍ SF500000 SF100000	܍ LN120000 LA510000		
-3	♥ SS020000	!! SP330000 SM010000	# ND030000 LC020000	S LS020000 LC010000	c LS010000 LA310000	s LA310000 LO310000	ā LO310000 LU430000	ó LU430000 SF110000	܍ SF170000 SF130000	܍ SF090000 SF050000	܍ SF100000 SF050000	܍ SF490000 SF080000	܍ SF500000 SF100000	܍ LN120000 LA520000			
-4	♦ SS030000	¶ SM250000 SC030000	\$ ND040000 LD020000	T LT020000 LD010000	d LT010000 LA170000	t LA170000 LO170000	ä LO170000 LA440000	ö LA440000 SF090000	܍ SF180000 SF140000	܍ SF090000 SF100000	܍ SF100000 SF050000	܍ SF490000 SF080000	܍ SF500000 SF100000	܍ SM250000 LO190000			
-5	♣ SS040000	§ SM240000 SM020000	% ND050000 LE020000	U LU020000 LE010000	e LU010000 LG410000	u LU010000 LG420000	܍ G LG410000 LG420000	܍ G LG420000 LA430000	܍ SF190000 SF150000	܍ SF190000 SF050000	܍ SF050000 SF510000	܍ SF510000 SF100000	܍ LO200000 SC010000				
-6	♠ SS050000	- SM700000 SM030000	& ND060000 LF020000	F LF020000 LV020000	v LV010000 LA270000	å LA270000 LU310000	܍ á LU310000 LZ220000	܍ á LZ220000 SF200000	܍ SF200000 SF360000	܍ SF360000 SF520000	܍ SF520000 LS220000	܍ SA060000 SA060000					
-7	● SM570000	↔ SM770000 SP050000	' ND070000 LG020000	G LW020000 LG010000	w LW010000 LC110000	܍ w LC110000 LS120000	܍ é LS120000 LZ210000	܍ é LZ210000 SF210000	܍ SF210000 SF370000	܍ SF370000 SF530000	܍ SF530000 LS210000	܍ LO610000 LO610000					
-8	▣ SM570001	↑ SM320000 SP060000	(ND080000 LH020000	H LX020000 LH010000	x LX010000 LL610000	܍ x LL610000 LS110000	܍ š LS110000 LE440000	܍ š LE440000 SF220000	܍ SF220000 SF380000	܍ SF380000 SF540000	܍ SF540000 LK420000	܍ SM190000 SM190000					
-9	○ SM750000	↓ SM330000 SP070000) ND090000 LI020000	I LY020000 LI010000	y LY010000 LY310000	܍ y LY310000 LO180000	܍ ö LO180000 LE430000	܍ ö LE430000 SF230000	܍ SF230000 SF390000	܍ SF390000 SF400000	܍ SF400000 LK410000	܍ LO620000 LO620000					
-A	Ⓐ SM750002	→ SM310000 SM040000	* SP130000 LJ020000	J LJ020000 LJ010000	܍ z LJ010000 LE290000	܍ z LE290000 LU180000	܍ é LU180000 LE300000	܍ é LE300000 SF240000	܍ SF240000 SF400000	܍ SF400000 SF410000	܍ SF410000 LU320000	܍ SD630000 SD630000					
-B	♂ SM280000	← SM300000 SA010000	+ SP140000 LK020000	K SM060000 LK010000	܍ k SM110000 LI430000	܍ k LI430000 LN110000	܍ í LN110000 LZ110000	܍ í LZ110000 SF250000	܍ SF250000 SF410000	܍ SF410000 SF610000	܍ SF610000 LU440000	܍ LR410000 LR410000					
-C	♀ SM290000	L SA420000 SP080000	, SA030000 LL020000	L SM070000 LL010000	\ SM130000 LI310000	܍ 1 SM130000 LL420000	܍ 1 LL420000 LC220000	܍ Č LC220000 SF260000	܍ SF260000 SF420000	܍ SF420000 SF570000	܍ SF570000 LL410000	܍ LR420000 LR420000					
-D	♪ SM930000	↔ SM780000 SP100000	- SA040000 LM020000	M SM080000 LM010000	܍ m SM140000 LZ120000	܍ m LZ120000 LL620000	܍ ž LL620000 LI440000	܍ ž LI440000 SF270000	܍ SF270000 SF430000	܍ SF430000 SF580000	܍ SF580000 LE320000	܍ SP230000 SP230000					
-E	♪ SM910000	↑ SM600000 SP110000	. SA050000 LN020000	N SD150000 LN010000	܍ ^ SD190000 LA180000	܍ ~ LA180000 SA070000	܍ ä SA070000 SP170000	܍ ä SP170000 SF280000	܍ SF280000 SF440000	܍ SF440000 SF590000	܍ SF590000 LN420000	܍ SP210000 SP210000					
-F	☀ SM690000	▼ SV040000 SP120000	/ SP150000 LO020000	O SP090000 LO010000	܍ - LA280000 LC210000	܍ o LA280000 SP180000	܍ č SP180000 SF030000	܍ č SF030000 SF450000	܍ SF450000 SF600000	܍ SF600000 LN410000	܍ p LN410000 (RSP)	܍ SP300000 SP300000					

Figure 87. Code page 1117 chart

Code Page 1118

HEX DIGITS 1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-		
-0	► (SP) SM590000	0 SP010000	@ ND100000	P SM050000	' LP020000	p SD130000	ç LP010000	LC420000	É LE120000	á LA110000	[grid] SF140000	[grid] SF020000	ä LA430000	α GA010000	≡ SA480000			
-1	☺ SS000000	◀ SM630000	! SP020000	1 ND010000	A LA020000	Q LQ020000	a LA010000	q LQ010000	ü LU170000	æ LA510000	í LI110000	[grid] SF150000	[grid] SF070000	č LC210000	β GB010000	± SA020000		
-2	☻ SS010000	↓ SM760000	" SP040000	2 ND020000	B LB020000	R LR020000	b LB010000	r LR010000	é LE110000	Æ LA520000	ó LO110000	[grid] SF160000	[grid] SF060000	ε LE430000	Γ GG020000	≥ SA530000		
-3	♥ SS020000	!! SP330000	# SM010000	3 ND030000	C LC020000	S LS020000	c LC010000	s LS010000	â LA150000	ô LO150000	ú LU110000	[grid] SF110000	[grid] SF080000	è LE290000	π GP010000	≤ SA520000		
-4	♦ SS030000	¶ SM250000	\$ SC030000	4 ND040000	D LD020000	T LT020000	d LD010000	t LT010000	ä LA170000	ö LO170000	ñ LN190000	[grid] SF090000	[grid] SF100000	í LI430000	Σ GS020000	„ SP230000		
-5	♣ SS040000	§ SM240000	% SM020000	5 ND050000	E LE020000	U LU020000	e LE010000	u LU010000	à LA130000	ò LO130000	Ñ LN200000	[grid] LA440000	[grid] SF050000	š LS210000	σ GS010000	“ SP210000		
-6	♠ SS050000	- SM700000	& SM030000	6 ND060000	F LF020000	V LV020000	f LF010000	v LV010000	å LA270000	û LU150000	ä SM210000	[grid] LC220000	[grid] LU440000	ψ LU430000	μ SM170000	÷ SA060000		
-7	● SM570000	↓ SM770000	' SP050000	7 ND070000	G LG020000	W LW020000	g LG010000	w LW010000	ç LC410000	ù LU130000	º SM200000	[grid] LE440000	[grid] LU320000	ū LU310000	τ GT010000	≈ SA700000		
-8	█ SM570001	↑ SM320000	(SP060000	8 ND080000	H LH020000	X LX020000	h LH010000	x LX010000	ê LE150000	ÿ LY170000	ķ SP160000	[grid] LE300000	[grid] SF380000	ž LZ210000	Φ GF020000	° SM190000		
-9	○ SM750000	↓ SM330000) SP070000	9 ND090000	I LI020000	Y LY020000	i LI010000	y LY010000	ë LE170000	ö LO180000	ń SM680000	[grid] SF230000	[grid] SF390000	θ SF040000	Θ GT620000	• SA790000		
-A	○ SM750002	→ SM310000	*	: SM040000	J SP130000	Z LJ020000	j LJ010000	z LZ010000	è LE130000	Ü LU180000	¬ SM660000	[grid] SF240000	[grid] SF400000	Ω SF010000	Ω GO320000	· SD290000		
-B	♂ SM280000	← SM300000	+	; SA010000	K SP140000	[LK020000	k SM060000	{ LK010000	í SM110000	¢ LI170000	½ SC040000	[grid] NF010000	[grid] SF250000	[grid] SF410000	δ SF610000	✓ GD010000	SA800000	
-C	♀ SM290000	L SA420000	,	< SP080000	L SA030000	\ LL020000	l SM070000	 LL010000	í SM130000	£ LI150000	¼ SC020000	[grid] NF040000	[grid] SF260000	[grid] SF420000	∞ SF570000	n SA450000	LN011000	
-D	♪ SM930000	↔ SM780000	-	= SP100000	M SA040000] LM020000	m SM080000	}	ì LM010000	¥ SM140000	í LI130000	í SC050000	í SP030000	[grid] LI440000	[grid] SF430000	φ SF580000	2 GF010000	ND021000
-E	♪ SM910000	▲ SM600000	.	> SP110000	N SA050000	~ LN020000	^ SD150000	n LN010000	~ SD190000	Ä LA180000	Þs SC060000	« SP170000	ſ LS220000	[grid] SF440000	[grid] SF590000	ε GE010000	■ SM470000	
-F	☀ SM690000	▼ SP040000	/	? SP120000	O SP150000	— LO020000	o SP090000	◊ LO010000	Å SM790000	f LA280000	» SC070000	[grid] SP180000	[grid] SF030000	ž LZ220000	[grid] SF600000	□ SA140000	(RSP) SP300000	

Figure 88. Code page 1118 chart

Code Page 1250

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	<u>NUL</u> 0000	<u>STX</u> 0001	<u>SOT</u> 0002	<u>ETX</u> 0003	<u>EOT</u> 0004	<u>ENQ</u> 0005	<u>ACK</u> 0006	<u>BEL</u> 0007	<u>BS</u> 0008	<u>HT</u> 0009	<u>LF</u> 000A	<u>VT</u> 000B	<u>FF</u> 000C	<u>CR</u> 000D	<u>SO</u> 000E	<u>SI</u> 000F
10	<u>DLE</u> 0010	<u>DC1</u> 0011	<u>DC2</u> 0012	<u>DC3</u> 0013	<u>DC4</u> 0014	<u>NAK</u> 0015	<u>SYN</u> 0016	<u>ETB</u> 0017	<u>CAN</u> 0018	<u>EM</u> 0019	<u>SUB</u> 001A	<u>ESC</u> 001B	<u>FS</u> 001C	<u>GS</u> 001D	<u>RS</u> 001E	<u>US</u> 001F
20	<u>SP</u> 0020	!	"	#	\$	%	&	*	()	*	+	,	-	.	/
30	0 0030	1 0031	2 0032	3 0033	4 0034	5 0035	6 0036	7 0037	8 0038	9 0039	:	;	<	=	>	?
40	@ 0040	A 0041	B 0042	C 0043	D 0044	E 0045	F 0046	G 0047	H 0048	I 0049	J 004A	K 004B	L 004C	M 004D	N 004E	O 004F
50	P 0050	Q 0051	R 0052	S 0053	T 0054	U 0055	V 0056	W 0057	X 0058	Y 0059	Z 005A	[005B	\ 005C] 005D	^ 005E	— 005F
60	~ 0060	a 0061	b 0062	c 0063	d 0064	e 0065	f 0066	g 0067	h 0068	i 0069	j 006A	k 006B	l 006C	m 006D	n 006E	o 006F
70	p 0070	q 0071	r 0072	s 0073	t 0074	u 0075	v 0076	w 0077	x 0078	y 0079	z 007A	{ 007B	 007C	} 007D	~ 007E	<u>DEL</u> 007F
80	€ 20AC	„ 201A	„ 201E	„ 2026	„ 2020	„ 2021	„ 2020	„ 2021	„ 2020	„ 2021	„ 2020	„ 2021	„ 2020	„ 2021	„ 2020	„ 2021
90	„ 2018	„ 2019	„ 201C	„ 201D	„ 2022	„ 2013	„ 2014	„ 2013	„ 2014	„ 2013	„ 2014	„ 2013	„ 2014	„ 2013	„ 2014	„ 2013
A0	<u>NBSP</u> 00A0	„ 02C7	„ 02D8	„ 0141	„ 00A4	„ 0104	„ 00A6	„ 00A7	„ 00A8	„ 00A9	„ 015E	„ 00AB	„ 00AC	„ 00AD	„ 00AE	„ 017B
B0	„ 00B0	„ 00B1	„ 02DB	„ 0142	„ 00B4	„ 00B5	„ 00B6	„ 00B7	„ 00B8	„ 0105	„ 015F	„ 00BB	„ 013D	„ 02DD	„ 013E	„ 017C
C0	„ 0154	„ 00C1	„ 00C2	„ 0102	„ 00C4	„ 0139	„ 0106	„ 00C7	„ 010C	„ 00C9	„ 0118	„ 00CB	„ 011A	„ 00CD	„ 00CE	„ 010E
D0	„ 0110	„ 0143	„ 0147	„ 00D3	„ 00D4	„ 0150	„ 00D6	„ 00D7	„ 0158	„ 016E	„ 00DA	„ 0170	„ 00DC	„ 00DD	„ 0162	„ 00DF
E0	„ 0155	„ 00E1	„ 00E2	„ 0103	„ 00E4	„ 013A	„ 0107	„ 00E7	„ 010D	„ 00E9	„ 0119	„ 00EB	„ 011B	„ 00ED	„ 00EE	„ 010F
F0	„ 0111	„ 0144	„ 0148	„ 00F3	„ 00F4	„ 0151	„ 00F6	„ 00F7	„ 0159	„ 016F	„ 00FA	„ 0171	„ 00FC	„ 00FD	„ 0163	„ 02D9

Figure 89. Code page 1250 chart

Code Page 1251

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	<u>NUL</u> 0000	<u>STX</u> 0001	<u>SOT</u> 0002	<u>ETX</u> 0003	<u>EOT</u> 0004	<u>ENQ</u> 0005	<u>ACK</u> 0006	<u>BEL</u> 0007	<u>BS</u> 0008	<u>HT</u> 0009	<u>LF</u> 000A	<u>VT</u> 000B	<u>FF</u> 000C	<u>CR</u> 000D	<u>SO</u> 000E	<u>SI</u> 000F
10	<u>DLE</u> 0010	<u>DC1</u> 0011	<u>DC2</u> 0012	<u>DC3</u> 0013	<u>DC4</u> 0014	<u>NAK</u> 0015	<u>SYN</u> 0016	<u>ETB</u> 0017	<u>CAN</u> 0018	<u>EM</u> 0019	<u>SUB</u> 001A	<u>ESC</u> 001B	<u>FS</u> 001C	<u>GS</u> 001D	<u>RS</u> 001E	<u>US</u> 001F
20	<u>SP</u> 0020	!	"	#	\$	%	&	*	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	<u>~</u>
60	'	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	()	{	}	<u>DEL</u> 007F
80	Ђ	Ѓ	Ѓ	„	„	„	„	†	‡	€	‰	Љ	<	Њ	Ќ	Ћ
90	Ђ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ
A0	<u>NBSP</u> 00A0	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ
B0	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ
C0	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ
D0	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ
E0	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ
F0	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ	Ѡ

Figure 90. Code page 1251 chart

Code Page 1252

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	<u>NUL</u> 0000	<u>STX</u> 0001	<u>SOT</u> 0002	<u>ETX</u> 0003	<u>EOT</u> 0004	<u>ENQ</u> 0005	<u>ACK</u> 0006	<u>BEL</u> 0007	<u>BS</u> 0008	<u>HT</u> 0009	<u>LF</u> 000A	<u>VT</u> 000B	<u>FF</u> 000C	<u>CR</u> 000D	<u>SO</u> 000E	<u>SI</u> 000F
10	<u>DLE</u> 0010	<u>DC1</u> 0011	<u>DC2</u> 0012	<u>DC3</u> 0013	<u>DC4</u> 0014	<u>NAK</u> 0015	<u>SYN</u> 0016	<u>ETB</u> 0017	<u>CAN</u> 0018	<u>EM</u> 0019	<u>SUB</u> 001A	<u>ESC</u> 001B	<u>FS</u> 001C	<u>GS</u> 001D	<u>RS</u> 001E	<u>US</u> 001F
20	<u>SP</u> 0020	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0 0030	1 0031	2 0032	3 0033	4 0034	5 0035	6 0036	7 0037	8 0038	9 0039	:	;	<	=	>	?
40	<u>Ø</u> 0040	A 0041	B 0042	C 0043	D 0044	E 0045	F 0046	G 0047	H 0048	I 0049	J 004A	K 004B	L 004C	M 004D	N 004E	O 004F
50	P 0050	Q 0051	R 0052	S 0053	T 0054	U 0055	V 0056	W 0057	X 0058	Y 0059	Z 005A	[005B	\] 005C	^ 005D	~ 005E	<u>Ø</u> 005F
60	<u>^</u> 0060	a 0061	b 0062	c 0063	d 0064	e 0065	f 0066	g 0067	h 0068	i 0069	j 006A	k 006B	l 006C	m 006D	n 006E	o 006F
70	P 0070	q 0071	r 0072	s 0073	t 0074	u 0075	v 0076	w 0077	x 0078	y 0079	z 007A	{ 007B	 007C	} 007D	<u>~</u> 007E	<u>DEL</u> 007F
80	<u>€</u> 20AC	<u>ƒ</u> 201A	<u>„</u> 0192	<u>„</u> 201E	<u>„</u> 2026	<u>„</u> 2020	<u>†</u> 2021	<u>‡</u> 02C6	<u>„</u> 2030	<u>„</u> 0160	<u>„</u> 2039	<u>„</u> 0152	<u>„</u> 017D	<u>„</u> 017D	<u>„</u> 017D	<u>„</u> 017D
90	<u>„</u> 2018	<u>„</u> 2019	<u>„</u> 201C	<u>„</u> 201D	<u>„</u> 2022	<u>„</u> 2013	<u>„</u> 2014	<u>„</u> 02DC	<u>„</u> 2122	<u>„</u> 0161	<u>„</u> 203A	<u>„</u> 0153	<u>„</u> 017E	<u>„</u> 017E	<u>„</u> 017E	
A0	<u>NBSP</u> 00A0	í 00A1	¢ 00A2	£ 00A3	¥ 00A4	₩ 00A5	₩ 00A6	₩ 00A7	₩ 00A8	₩ 00A9	₩ 00AA	₩ 00AB	₩ 00AC	₩ 00AD	₩ 00AE	₩ 00AF
B0	<u>°</u> 00B0	± 00B1	² 00B2	³ 00B3	⁴ 00B4	µ 00B5	¶ 00B6	· 00B7	, 00B8	· 00B9	· 00BA	» 00BB	» 00BC	» 00BD	» 00BE	» 00BF
C0	À 00C0	Á 00C1	Ã 00C2	Ä 00C3	Å 00C4	Æ 00C5	Ç 00C6	È 00C7	É 00C8	Ê 00C9	Ë 00CA	Ì 00CB	Í 00CC	Ï 00CD	Ï 00CE	Ï 00CF
D0	Ð 00D0	Ñ 00D1	Ò 00D2	Ó 00D3	Ô 00D4	Õ 00D5	× 00D6	× 00D7	Ø 00D8	Ù 00D9	Ú 00DA	Û 00DB	Û 00DC	Û 00DD	Û 00DE	Û 00DF
E0	à 00E0	á 00E1	ã 00E2	ä 00E3	ë 00E4	å 00E5	æ 00E6	ç 00E7	è 00E8	é 00E9	ë 00EA	ë 00EB	ì 00EC	í 00ED	í 00EE	í 00EF
F0	õ 00F0	ñ 00F1	ò 00F2	ó 00F3	ô 00F4	õ 00F5	ö 00F6	÷ 00F7	ø 00F8	ù 00F9	ú 00FA	û 00FB	û 00FC	û 00FD	û 00FE	û 00FF

Figure 91. Code page 1252 chart

Code Page 1253

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	<u>NUL</u> 0000	<u>STX</u> 0001	<u>SOT</u> 0002	<u>ETX</u> 0003	<u>EOT</u> 0004	<u>ENQ</u> 0005	<u>ACK</u> 0006	<u>BEL</u> 0007	<u>BS</u> 0008	<u>HT</u> 0009	<u>LF</u> 000A	<u>VT</u> 000B	<u>FF</u> 000C	<u>CR</u> 000D	<u>SO</u> 000E	<u>SI</u> 000F
10	<u>DLE</u> 0010	<u>DC1</u> 0011	<u>DC2</u> 0012	<u>DC3</u> 0013	<u>DC4</u> 0014	<u>NAK</u> 0015	<u>SYN</u> 0016	<u>ETB</u> 0017	<u>CAN</u> 0018	<u>EM</u> 0019	<u>SUB</u> 001A	<u>ESC</u> 001B	<u>FS</u> 001C	<u>GS</u> 001D	<u>RS</u> 001E	<u>US</u> 001F
20	<u>SP</u> 0020	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	~	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	()	{	}	<u>DEL</u> 007F
80	€ 20AC	⋮ 201A	⋮ 0192	⋮ 201E	⋮ 2026	⋮ 2020	⋮ 2021	⋮ 2030	⋮ 2039	⋮ 203A	⋮ 2122	⋮ 203B	⋮ 203C	⋮ 203D	⋮ 203E	⋮ 203F
90	⋮ 2018	⋮ 2019	⋮ 201C	⋮ 201D	⋮ 2022	⋮ 2013	⋮ 2014	⋮ 203B	⋮ 203C	⋮ 203D	⋮ 203E	⋮ 203F	⋮ 203A	⋮ 203B	⋮ 203C	⋮ 203D
A0	<u>NBSP</u> 00A0	⋮ 0385	⋮ 0386	⋮ 00A3	⋮ 00A4	⋮ 00A5	⋮ 00A6	⋮ 00A7	⋮ 00A8	⋮ 00A9	⋮ 00AB	⋮ 00AC	⋮ 00AD	⋮ 00AE	⋮ 2015	⋮ 2016
B0	⋮ 00B0	⋮ 00B1	⋮ 00B2	⋮ 00B3	⋮ 0384	⋮ 00B5	⋮ 00B6	⋮ 00B7	⋮ 0388	⋮ 0389	⋮ 038A	⋮ 00BB	⋮ 038C	⋮ 00BD	⋮ 038E	⋮ 038F
C0	⋮ 0390	⋮ 0391	⋮ 0392	⋮ 0393	⋮ 0394	⋮ 0395	⋮ 0396	⋮ 0397	⋮ 0398	⋮ 0399	⋮ 039A	⋮ 039B	⋮ 039C	⋮ 039D	⋮ 039E	⋮ 039F
D0	⋮ 03A0	⋮ 03A1	⋮ 03A2	⋮ 03A3	⋮ 03A4	⋮ 03A5	⋮ 03A6	⋮ 03A7	⋮ 03A8	⋮ 03A9	⋮ 03AA	⋮ 03AB	⋮ 03AC	⋮ 03AD	⋮ 03AE	⋮ 03AF
E0	⋮ 03B0	⋮ 03B1	⋮ 03B2	⋮ 03B3	⋮ 03B4	⋮ 03B5	⋮ 03B6	⋮ 03B7	⋮ 03B8	⋮ 03B9	⋮ 03BA	⋮ 03BB	⋮ 03BC	⋮ 03BD	⋮ 03BE	⋮ 03BF
F0	⋮ 03C0	⋮ 03C1	⋮ 03C2	⋮ 03C3	⋮ 03C4	⋮ 03C5	⋮ 03C6	⋮ 03C7	⋮ 03C8	⋮ 03C9	⋮ 03CA	⋮ 03CB	⋮ 03CC	⋮ 03CD	⋮ 03CE	⋮ 03CF

Figure 92. Code page 1253 chart

Code Page 1254

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	<u>NUL</u> 0000	<u>STX</u> 0001	<u>SOT</u> 0002	<u>ETX</u> 0003	<u>EOT</u> 0004	<u>ENQ</u> 0005	<u>ACK</u> 0006	<u>BEL</u> 0007	<u>BS</u> 0008	<u>HT</u> 0009	<u>LF</u> 000A	<u>VT</u> 000B	<u>FF</u> 000C	<u>CR</u> 000D	<u>SO</u> 000E	<u>SI</u> 000F
10	<u>DLE</u> 0010	<u>DC1</u> 0011	<u>DC2</u> 0012	<u>DC3</u> 0013	<u>DC4</u> 0014	<u>NAK</u> 0015	<u>SYN</u> 0016	<u>ETB</u> 0017	<u>CAN</u> 0018	<u>EM</u> 0019	<u>SUB</u> 001A	<u>ESC</u> 001B	<u>FS</u> 001C	<u>GS</u> 001D	<u>RS</u> 001E	<u>US</u> 001F
20	<u>SP</u> 0020	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0 0030	1 0031	2 0032	3 0033	4 0034	5 0035	6 0036	7 0037	8 0038	9 0039	:	;	<	=	>	?
40	<u>Ø</u> 0040	A 0041	B 0042	C 0043	D 0044	E 0045	F 0046	G 0047	H 0048	I 0049	J 004A	K 004B	L 004C	M 004D	N 004E	O 004F
50	P 0050	Q 0051	R 0052	S 0053	T 0054	U 0055	V 0056	W 0057	X 0058	Y 0059	Z 005A	[005B	\] 005C	^ 005D	~ 005E	<u>Ø</u> 005F
60	~ 0060	a 0061	b 0062	c 0063	d 0064	e 0065	f 0066	g 0067	h 0068	i 0069	j 006A	k 006B	l 006C	m 006D	n 006E	o 006F
70	P 0070	q 0071	r 0072	s 0073	t 0074	u 0075	v 0076	w 0077	x 0078	y 0079	z 007A	{ 007B	 007C	} 007D	~ 007E	<u>DEL</u> 007F
80	<u>€</u> 20AC	ƒ 201A	f 0192	" 201E	" 2026	" 2020	† 2021	‡ 2021	^ 2020	‰ 2030	„ 0160	„ 2039	Œ 0152	„ 2039	„ 0152	„ 2039
90	„ 2018	„ 2019	„ 201C	„ 201D	„ 2022	„ 2013	„ 2014	„ 02DC	„ 2122	„ 0161	„ 203A	„ 0153	„ 203A	„ 0153	„ 203A	„ 0178
A0	<u>NBSP</u> 00A0	ı 00A1	¢ 00A2	£ 00A3	¥ 00A4	₩ 00A5	₩ 00A6	₩ 00A7	₩ 00A8	₩ 00A9	₩ 00AA	₩ 00AB	₩ 00AC	₩ 00AD	₩ 00AE	₩ 00AF
B0	° 00B0	± 00B1	² 00B2	³ 00B3	⁴ 00B4	µ 00B5	¶ 00B6	· 00B7	, 00B8	· 00B9	· 00BA	» 00BB	· 00BC	· 00BD	· 00BE	· 00BF
C0	À 00C0	Á 00C1	Ã 00C2	Ä 00C3	Å 00C4	Æ 00C5	Ç 00C6	È 00C7	É 00C8	Ê 00C9	Ë 00CA	Ì 00CB	Í 00CC	Î 00CD	Ï 00CE	Ï 00CF
D0	Ğ 011E	Ñ 00D1	Ò 00D2	Ó 00D3	Ô 00D4	Õ 00D5	Ö 00D6	× 00D7	Ø 00D8	Ù 00D9	Ú 00DA	Û 00DB	Ü 00DC	Ü 0130	Ü 015E	Ü 00DF
E0	à 00E0	á 00E1	ã 00E2	ä 00E3	ë 00E4	å 00E5	æ 00E6	ç 00E7	è 00E8	é 00E9	ë 00EA	ë 00EB	ì 00EC	í 00ED	í 00EE	í 00EF
F0	ÿ 011F	ñ 00F1	ò 00F2	ó 00F3	ô 00F4	õ 00F5	ö 00F6	÷ 00F7	ø 00F8	ù 00F9	ú 00FA	ü 00FB	ü 00FC	ü 0131	ü 015F	ÿ 00FF

Figure 93. Code page 1254 chart

Code Page 1255

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	<u>NUL</u> 0000	<u>STX</u> 0001	<u>SOT</u> 0002	<u>ETX</u> 0003	<u>EOT</u> 0004	<u>ENQ</u> 0005	<u>ACK</u> 0006	<u>BEL</u> 0007	<u>BS</u> 0008	<u>HT</u> 0009	<u>LF</u> 000A	<u>VT</u> 000B	<u>FF</u> 000C	<u>CR</u> 000D	<u>SO</u> 000E	<u>SI</u> 000F
10	<u>DLE</u> 0010	<u>DC1</u> 0011	<u>DC2</u> 0012	<u>DC3</u> 0013	<u>DC4</u> 0014	<u>NAK</u> 0015	<u>SYN</u> 0016	<u>ETB</u> 0017	<u>CAN</u> 0018	<u>EM</u> 0019	<u>SUB</u> 001A	<u>ESC</u> 001B	<u>FS</u> 001C	<u>GS</u> 001D	<u>RS</u> 001E	<u>US</u> 001F
20	<u>SP</u> 0020	!	"	#	\$	%	&	*	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	<u>~</u>
60	~	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	()	{	}	<u>DEL</u> 007F
80	€ 20AC	⋮ 201A	⋮ 0192	⋮ 201E	⋮ 2026	⋮ 2020	⋮ 2021	⋮ 02C6	⋮ 2030	⋮ 2039	⋮ 203A	⋮ 203B	⋮ 203C	⋮ 203D	⋮ 203E	⋮ 203F
90	⋮ 2018	⋮ 2019	⋮ 201C	⋮ 201D	⋮ 2022	⋮ 2013	⋮ 2014	⋮ 02DC	⋮ 2122	⋮ 203A	⋮ 203B	⋮ 203C	⋮ 203D	⋮ 203E	⋮ 203F	⋮ 203G
A0	<u>NBSP</u> 00A0	⋮ 00A1	⋮ 00A2	⋮ 00A3	⋮ 20AA	⋮ 00A5	⋮ 00A6	⋮ 00A7	⋮ 00A8	⋮ 00A9	⋮ 00D7	⋮ 00AB	⋮ 00AC	⋮ 00AD	⋮ 00AE	⋮ 00AF
B0	⋮ 00B0	⋮ 00B1	⋮ 00B2	⋮ 00B3	⋮ 00B4	⋮ 00B5	⋮ 00B6	⋮ 00B7	⋮ 00B8	⋮ 00B9	⋮ 00F7	⋮ 00BB	⋮ 00BC	⋮ 00BD	⋮ 00BE	⋮ 00BF
C0	⋮ 05B0	⋮ 05B1	⋮ 05B2	⋮ 05B3	⋮ 05B4	⋮ 05B5	⋮ 05B6	⋮ 05B7	⋮ 05B8	⋮ 05B9	⋮ 05BB	⋮ 05BC	⋮ 05BD	⋮ 05BE	⋮ 05BF	⋮ 05BG
D0	⋮ 05C0	⋮ 05C1	⋮ 05C2	⋮ 05C3	⋮ 05F0	⋮ 05F1	⋮ 05F2	⋮ 05F3	⋮ 05F4	⋮ 05F5	⋮ 05F6	⋮ 05F7	⋮ 05F8	⋮ 05F9	⋮ 05FA	⋮ 05FB
E0	⋮ 05D0	⋮ 05D1	⋮ 05D2	⋮ 05D3	⋮ 05D4	⋮ 05D5	⋮ 05D6	⋮ 05D7	⋮ 05D8	⋮ 05D9	⋮ 05DA	⋮ 05DB	⋮ 05DC	⋮ 05DD	⋮ 05DE	⋮ 05DF
F0	⋮ 05E0	⋮ 05E1	⋮ 05E2	⋮ 05E3	⋮ 05E4	⋮ 05E5	⋮ 05E6	⋮ 05E7	⋮ 05E8	⋮ 05E9	⋮ 05EA	⋮ 05EB	⋮ 05EC	⋮ 05ED	⋮ 05EF	⋮ 05FG

Figure 94. Code page 1255 chart

Code Page 1256

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	<u>NUL</u> 0000	<u>STX</u> 0001	<u>SOT</u> 0002	<u>ETX</u> 0003	<u>EOT</u> 0004	<u>ENQ</u> 0005	<u>ACK</u> 0006	<u>BEL</u> 0007	<u>BS</u> 0008	<u>HT</u> 0009	<u>LF</u> 000A	<u>VT</u> 000B	<u>FF</u> 000C	<u>CR</u> 000D	<u>SO</u> 000E	<u>SI</u> 000F
10	<u>DLE</u> 0010	<u>DC1</u> 0011	<u>DC2</u> 0012	<u>DC3</u> 0013	<u>DC4</u> 0014	<u>NAK</u> 0015	<u>SYN</u> 0016	<u>ETB</u> 0017	<u>CAN</u> 0018	<u>EM</u> 0019	<u>SUB</u> 001A	<u>ESC</u> 001B	<u>FS</u> 001C	<u>GS</u> 001D	<u>RS</u> 001E	<u>US</u> 001F
20	<u>SP</u> 0020	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0 0030	1 0031	2 0032	3 0033	4 0034	5 0035	6 0036	7 0037	8 0038	9 0039	:	;	<	=	>	?
40	<u>Ø</u> 0040	A 0041	B 0042	C 0043	D 0044	E 0045	F 0046	G 0047	H 0048	I 0049	J 004A	K 004B	L 004C	M 004D	N 004E	O 004F
50	P 0050	Q 0051	R 0052	S 0053	T 0054	U 0055	V 0056	W 0057	X 0058	Y 0059	Z 005A	[005B	\] 005C	^ 005D	^ 005E	^ 005F
60	~ 0060	a 0061	b 0062	c 0063	d 0064	e 0065	f 0066	g 0067	h 0068	i 0069	j 006A	k 006B	l 006C	m 006D	n 006E	o 006F
70	P 0070	q 0071	r 0072	s 0073	t 0074	u 0075	v 0076	w 0077	x 0078	y 0079	z 007A	{ 007B	 007C	} 007D	~ 007E	<u>DEL</u> 007F
80	€ 20AC	߲ 067E	߳ 201A	ߴ 0192	ߵ 201E	߶ 2026	߷ 2020	߸ 2021	߹ 02C6	߻ 2030	߻ 0679	߻ 2039	߻ 0152	߻ 0686	߻ 0698	߻ 0688
90	݂ 06AF	݁ 2018	݂ 2019	݂ 201C	݂ 201D	݂ 2022	݂ 2013	݂ 2014	݂ 06A9	݂ 2122	݂ 0691	݂ 203A	݂ 0153	݂ 200C	݂ 200D	݂ 06BA
A0	<u>NBSP</u> 00A0	݂ 060C	݂ 00A2	݂ 00A3	݂ 00A4	݂ 00A5	݂ 00A6	݂ 00A7	݂ 00A8	݂ 00A9	݂ 06BE	݂ 00AB	݂ 00AC	݂ 00AD	݂ 00AE	݂ 00AF
B0	܂ 00B0	܂ 00B1	܂ 00B2	܂ 00B3	܂ 00B4	܂ 00B5	܂ 00B6	܂ 00B7	܂ 00B8	܂ 00B9	܂ 061B	܂ 00BB	܂ 00BC	܂ 00BD	܂ 00BE	܂ 061F
C0	܂ 06C1	܂ 0621	܂ 0622	܂ 0623	܂ 0624	܂ 0625	܂ 0626	܂ 0627	܂ 0628	܂ 0629	܂ 062A	܂ 062B	܂ 062C	܂ 062D	܂ 062E	܂ 062F
D0	܂ 0630	܂ 0631	܂ 0632	܂ 0633	܂ 0634	܂ 0635	܂ 0636	܂ 00D7	܂ 0637	܂ 0638	܂ 0639	܂ 063A	܂ 0640	܂ 0641	܂ 0642	܂ 0643
E0	܂ 00E0	܂ 0644	܂ 00E2	܂ 0645	܂ 0646	܂ 0647	܂ 0648	܂ 00E7	܂ 00E8	܂ 00E9	܂ 00EA	܂ 00EB	܂ 0643	܂ 064A	܂ 00EE	܂ 00EF
F0	܂ 064B	܂ 064C	܂ 064D	܂ 064E	܂ 00F4	܂ 064F	܂ 0650	܂ 00F7	܂ 0651	܂ 00F9	܂ 0652	܂ 00FB	܂ 00FC	܂ LTR 200E	܂ RTL 200F	܂ 06D2

Figure 95. Code page 1256 chart

Code Page 1257

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	<u>NUL</u> 0000	<u>STX</u> 0001	<u>SOT</u> 0002	<u>ETX</u> 0003	<u>EOT</u> 0004	<u>ENQ</u> 0005	<u>ACK</u> 0006	<u>BEL</u> 0007	<u>BS</u> 0008	<u>HT</u> 0009	<u>LF</u> 000A	<u>VT</u> 000B	<u>FF</u> 000C	<u>CR</u> 000D	<u>SO</u> 000E	<u>SI</u> 000F
10	<u>DLE</u> 0010	<u>DC1</u> 0011	<u>DC2</u> 0012	<u>DC3</u> 0013	<u>DC4</u> 0014	<u>NAK</u> 0015	<u>SYN</u> 0016	<u>ETB</u> 0017	<u>CAN</u> 0018	<u>EM</u> 0019	<u>SUB</u> 001A	<u>ESC</u> 001B	<u>FS</u> 001C	<u>GS</u> 001D	<u>RS</u> 001E	<u>US</u> 001F
20	<u>SP</u> 0020	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	~	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	()	{	}	<u>DEL</u> 007F
80	€ 20AC	⋮ 201A	⋮ 201E	⋮ 201D	⋮ 2022	⋮ 2013	⋮ 2014	⋮ 2012	⋮ 2011	⋮ 2030	⋮ 2039	⋮ 2038	⋮ 2037	⋮ 2036	⋮ 2035	⋮ 2034
90	⋮ 2018	⋮ 2019	⋮ 201C	⋮ 201D	⋮ 2022	⋮ 2013	⋮ 2014	⋮ 2012	⋮ 2122	⋮ 203A	⋮ 2039	⋮ 2038	⋮ 2037	⋮ 2036	⋮ 2035	⋮ 2034
A0	<u>NBSP</u> 00A0	◊ 00A2	£ 00A3	* 00A4	⋮ 00A5	 00A6	⋮ 00A7	⋮ 00D8	∅ 00A9	® 0156	® 00AB	« 00AC	¬ 00AD	® 00AE	® 00C6	
B0	° 00B0	± 00B1	z 00B2	ؒ 00B3	ؑ 00B4	ؔ 00B5	ؕ 00B6	ؖ 00B7	ؘ 00F8	ؙ 00B9	ؚ 0157	ؗ 00BB	ؘ 00BC	ؘ 00BD	ؘ 00BE	ؘ 00E6
C0	À _t 0104	Ā _t 012E	Ā _À 0100	Ā _Ā 0106	Ā _Ā 00C4	Ā _Ā 00C5	Ā _Ā 0118	Ā _Ā 0112	Ā _Ā 010C	Ā _Ā 00C9	Ā _Ā 0179	Ā _Ā 0116	Ā _Ā 0122	Ā _Ā 0136	Ā _Ā 012A	Ā _Ā 013B
D0	Š ₀₁₆₀	Ń ₀₁₄₃	Ń _Ń 0145	Ó _{00D3}	Ó _Ó 014C	Ó _Ó 00D5	Ó _Ó 00D6	Ó _Ó 00D7	Ó _Ó 0172	Ó _Ó 0141	Ó _Ó 015A	Ó _Ó 016A	Ó _Ó 00DC	Ó _Ó 017B	Ó _Ó 017D	Ó _Ó 00DF
E0	ä ₀₁₀₅	í _{012F}	ã ₀₁₀₁	ć ₀₁₀₇	ä _ä 00E4	å _ä 00E5	é _ä 0119	ë _ä 0113	č _ä 010D	é _ä 00E9	ž _ä 017A	é _ä 0117	ǵ _ä 0123	ķ _ä 0137	í _ä 012B	ł _ä 013C
F0	š ₀₁₆₁	ń ₀₁₄₄	ń _ń 0146	ó _ń 00F3	ó _ó 014D	ó _ó 00F5	ó _ó 00F6	ó _ó 00F7	ú _ó 0173	ú _ó 0142	ú _ó 015B	ú _ó 016B	ú _ó 00FC	ż _ó 017C	ż _ó 017E	ż _ó 02D9

Figure 96. Code page 1257 chart

Code Page 932

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	<u>NUL</u> 0000	<u>STX</u> 0001	<u>SOT</u> 0002	<u>ETX</u> 0003	<u>EOT</u> 0004	<u>ENQ</u> 0005	<u>ACK</u> 0006	<u>BEL</u> 0007	<u>BS</u> 0008	<u>HT</u> 0009	<u>LF</u> 000A	<u>VT</u> 000B	<u>FF</u> 000C	<u>CR</u> 000D	<u>SO</u> 000E	<u>SI</u> 000F
10	<u>DLE</u> 0010	<u>DC1</u> 0011	<u>DC2</u> 0012	<u>DC3</u> 0013	<u>DC4</u> 0014	<u>NAK</u> 0015	<u>SYN</u> 0016	<u>ETB</u> 0017	<u>CAN</u> 0018	<u>EM</u> 0019	<u>SUB</u> 001A	<u>ESC</u> 001B	<u>FS</u> 001C	<u>GS</u> 001D	<u>RS</u> 001E	<u>US</u> 001F
20	<u>SP</u> 0020	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0 0030	1 0031	2 0032	3 0033	4 0034	5 0035	6 0036	7 0037	8 0038	9 0039	:	;	<	=	>	?
40	@ 0040	A 0041	B 0042	C 0043	D 0044	E 0045	F 0046	G 0047	H 0048	I 0049	J 004A	K 004B	L 004C	M 004D	N 004E	O 004F
50	P 0050	Q 0051	R 0052	S 0053	T 0054	U 0055	V 0056	W 0057	X 0058	Y 0059	Z 005A	[005B] 005C	¥ 005D	^ 005E	— 005F
60	~ 0060	a 0061	b 0062	c 0063	d 0064	e 0065	f 0066	g 0067	h 0068	i 0069	j 006A	k 006B	l 006C	m 006D	n 006E	o 006F
70	p 0070	q 0071	r 0072	s 0073	t 0074	u 0075	v 0076	w 0077	x 0078	y 0079	z 007A	{ 007B	 007C	} 007D	~ 007E	DEL 007F
80 81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F	
90	90	91	92	93	94	95	96	97	98	99	9A	9B	9C	9D	9E	9F
A0	。 FF61	「 FF62	」 FF63	、 FF64	・ FF65	ヲ FF66	ア FF67	イ FF68	ウ FF69	エ FF6A	オ FF6B	ヤ FF6C	ュ FF6D	ヨ FF6E	ヅ FF6F	
B0	一 FF70	ア FF71	イ FF72	ウ FF73	工 FF74	才 FF75	カ FF76	キ FF77	ク FF78	ケ FF79	コ FF7A	サ FF7B	シ FF7C	ス FF7D	セ FF7E	ゞ FF7F
C0	タ FF80	チ FF81	ツ FF82	テ FF83	ト FF84	ナ FF85	ニ FF86	ヌ FF87	ネ FF88	ノ FF89	ハ FF8A	ヒ FF8B	フ FF8C	ヘ FF8D	ホ FF8E	ゞ FF8F
D0	ミ FF90	ム FF91	メ FF92	モ FF93	ヤ FF94	ユ FF95	ヨ FF96	ラ FF97	リ FF98	ル FF99	レ FF9A	ロ FF9B	ワ FF9C	ン FF9D	ヽ FF9E	ヾ FF9F
E0	<u>E0</u>	<u>E1</u>	<u>E2</u>	<u>E3</u>	<u>E4</u>	<u>E5</u>	<u>E6</u>	<u>E7</u>	<u>E8</u>	<u>E9</u>	<u>EA</u>	<u>EB</u>	<u>EC</u>	<u>ED</u>	<u>EE</u>	<u>EF</u>
F0	<u>F0</u>	<u>F1</u>	<u>F2</u>	<u>F3</u>	<u>F4</u>	<u>F5</u>	<u>F6</u>	<u>F7</u>	<u>F8</u>	<u>F9</u>	<u>FA</u>	<u>FB</u>	<u>FC</u>

Figure 97. Code page 932 chart

Code Page 936

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	<u>NUL</u> 0000	<u>STX</u> 0001	<u>SOT</u> 0002	<u>ETX</u> 0003	<u>EOT</u> 0004	<u>ENQ</u> 0005	<u>ACK</u> 0006	<u>BEL</u> 0007	<u>BS</u> 0008	<u>HT</u> 0009	<u>LF</u> 000A	<u>VT</u> 000B	<u>FF</u> 000C	<u>CR</u> 000D	<u>SO</u> 000E	<u>SI</u> 000F
10	<u>DLE</u> 0010	<u>DC1</u> 0011	<u>DC2</u> 0012	<u>DC3</u> 0013	<u>DC4</u> 0014	<u>NAK</u> 0015	<u>SYN</u> 0016	<u>ETB</u> 0017	<u>CAN</u> 0018	<u>EM</u> 0019	<u>SUB</u> 001A	<u>ESC</u> 001B	<u>FS</u> 001C	<u>GS</u> 001D	<u>RS</u> 001E	<u>US</u> 001F
20	<u>SP</u> 0020	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0 0030	1 0031	2 0032	3 0033	4 0034	5 0035	6 0036	7 0037	8 0038	9 0039	:	;	<	=	>	?
40	@ 0040	A 0041	B 0042	C 0043	D 0044	E 0045	F 0046	G 0047	H 0048	I 0049	J 004A	K 004B	L 004C	M 004D	N 004E	O 004F
50	P 0050	Q 0051	R 0052	S 0053	T 0054	U 0055	V 0056	W 0057	X 0058	Y 0059	Z 005A	[005B	\ 005C] 005D	^ 005E	_ 005F
60	~ 0060	a 0061	b 0062	c 0063	d 0064	e 0065	f 0066	g 0067	h 0068	i 0069	j 006A	k 006B	l 006C	m 006D	n 006E	o 006F
70	p 0070	q 0071	r 0072	s 0073	t 0074	u 0075	v 0076	w 0077	x 0078	y 0079	z 007A	{ 007B	 007C	} 007D	~ 007E	<u>DEL</u> 007F
80	€ 20AC	<u>81</u>	<u>82</u>	<u>83</u>	<u>84</u>	<u>85</u>	<u>86</u>	<u>87</u>	<u>88</u>	<u>89</u>	<u>8A</u>	<u>8B</u>	<u>8C</u>	<u>8D</u>	<u>8E</u>	<u>8F</u>
90	<u>90</u>	<u>91</u>	<u>92</u>	<u>93</u>	<u>94</u>	<u>95</u>	<u>96</u>	<u>97</u>	<u>98</u>	<u>99</u>	<u>9A</u>	<u>9B</u>	<u>9C</u>	<u>9D</u>	<u>9E</u>	<u>9F</u>
A0	<u>A0</u>	<u>A1</u>	<u>A2</u>	<u>A3</u>	<u>A4</u>	<u>A5</u>	<u>A6</u>	<u>A7</u>	<u>A8</u>	<u>A9</u>	<u>AA</u>	<u>AB</u>	<u>AC</u>	<u>AD</u>	<u>AE</u>	<u>AF</u>
B0	<u>BO</u>	<u>B1</u>	<u>B2</u>	<u>B3</u>	<u>B4</u>	<u>B5</u>	<u>B6</u>	<u>B7</u>	<u>B8</u>	<u>B9</u>	<u>BA</u>	<u>BB</u>	<u>BC</u>	<u>BD</u>	<u>BE</u>	<u>BF</u>
C0	<u>CO</u>	<u>C1</u>	<u>C2</u>	<u>C3</u>	<u>C4</u>	<u>C5</u>	<u>C6</u>	<u>C7</u>	<u>C8</u>	<u>C9</u>	<u>CA</u>	<u>CB</u>	<u>CC</u>	<u>CD</u>	<u>CE</u>	<u>CF</u>
D0	<u>DO</u>	<u>D1</u>	<u>D2</u>	<u>D3</u>	<u>D4</u>	<u>D5</u>	<u>D6</u>	<u>D7</u>	<u>D8</u>	<u>D9</u>	<u>DA</u>	<u>DB</u>	<u>DC</u>	<u>DD</u>	<u>DE</u>	<u>DF</u>
E0	<u>EO</u>	<u>E1</u>	<u>E2</u>	<u>E3</u>	<u>E4</u>	<u>E5</u>	<u>E6</u>	<u>E7</u>	<u>E8</u>	<u>E9</u>	<u>EA</u>	<u>EB</u>	<u>EC</u>	<u>ED</u>	<u>EE</u>	<u>EF</u>
F0	<u>FO</u>	<u>F1</u>	<u>F2</u>	<u>F3</u>	<u>F4</u>	<u>F5</u>	<u>F6</u>	<u>F7</u>	<u>F8</u>	<u>F9</u>	<u>FA</u>	<u>FB</u>	<u>FC</u>	<u>FD</u>	<u>FE</u>	

Figure 98. Code page 936 chart

Code Page 949

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	<u>NUL</u> 0000	<u>STX</u> 0001	<u>SOT</u> 0002	<u>ETX</u> 0003	<u>EOT</u> 0004	<u>ENQ</u> 0005	<u>ACK</u> 0006	<u>BEL</u> 0007	<u>BS</u> 0008	<u>HT</u> 0009	<u>LF</u> 000A	<u>VT</u> 000B	<u>FF</u> 000C	<u>CR</u> 000D	<u>SO</u> 000E	<u>SI</u> 000F
10	<u>DLE</u> 0010	<u>DC1</u> 0011	<u>DC2</u> 0012	<u>DC3</u> 0013	<u>DC4</u> 0014	<u>NAK</u> 0015	<u>SYN</u> 0016	<u>ETB</u> 0017	<u>CAN</u> 0018	<u>EM</u> 0019	<u>SUB</u> 001A	<u>ESC</u> 001B	<u>FS</u> 001C	<u>GS</u> 001D	<u>RS</u> 001E	<u>US</u> 001F
20	<u>SP</u> 0020	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0 0030	1 0031	2 0032	3 0033	4 0034	5 0035	6 0036	7 0037	8 0038	9 0039	:	;	<	=	>	?
40	@ 0040	A 0041	B 0042	C 0043	D 0044	E 0045	F 0046	G 0047	H 0048	I 0049	J 004A	K 004B	L 004C	M 004D	N 004E	O 004F
50	P 0050	Q 0051	R 0052	S 0053	T 0054	U 0055	V 0056	W 0057	X 0058	Y 0059	Z 005A	[005B] 005C	^ 005D	- 005E	005F
60	~ 0060	a 0061	b 0062	c 0063	d 0064	e 0065	f 0066	g 0067	h 0068	i 0069	j 006A	k 006B	l 006C	m 006D	n 006E	o 006F
70	p 0070	q 0071	r 0072	s 0073	t 0074	u 0075	v 0076	w 0077	x 0078	y 0079	z 007A	{ 007B	 007C	} 007D	~ 007E	<u>DEL</u> 007F
80 80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F
90	90	91	92	93	94	95	96	97	98	99	9A	9B	9C	9D	9E	9F
A0	<u>AO</u>	<u>A1</u>	<u>A2</u>	<u>A3</u>	<u>A4</u>	<u>A5</u>	<u>A6</u>	<u>A7</u>	<u>A8</u>	<u>A9</u>	<u>AA</u>	<u>AB</u>	<u>AC</u>	<u>AD</u>	<u>AE</u>	<u>AF</u>
B0	<u>BO</u>	<u>B1</u>	<u>B2</u>	<u>B3</u>	<u>B4</u>	<u>B5</u>	<u>B6</u>	<u>B7</u>	<u>B8</u>	<u>B9</u>	<u>BA</u>	<u>BB</u>	<u>BC</u>	<u>BD</u>	<u>BE</u>	<u>BF</u>
C0	<u>CO</u>	<u>C1</u>	<u>C2</u>	<u>C3</u>	<u>C4</u>	<u>C5</u>	<u>C6</u>	<u>C7</u>	<u>C8</u>	<u>C9</u>	<u>CA</u>	<u>CB</u>	<u>CC</u>	<u>CD</u>	<u>CE</u>	<u>CF</u>
D0	<u>DO</u>	<u>D1</u>	<u>D2</u>	<u>D3</u>	<u>D4</u>	<u>D5</u>	<u>D6</u>	<u>D7</u>	<u>D8</u>	<u>D9</u>	<u>DA</u>	<u>DB</u>	<u>DC</u>	<u>DD</u>	<u>DE</u>	<u>DF</u>
E0	<u>EO</u>	<u>E1</u>	<u>E2</u>	<u>E3</u>	<u>E4</u>	<u>E5</u>	<u>E6</u>	<u>E7</u>	<u>E8</u>	<u>E9</u>	<u>EA</u>	<u>EB</u>	<u>EC</u>	<u>ED</u>	<u>EE</u>	<u>EF</u>
F0	<u>FO</u>	<u>F1</u>	<u>F2</u>	<u>F3</u>	<u>F4</u>	<u>F5</u>	<u>F6</u>	<u>F7</u>	<u>F8</u>	<u>F9</u>	<u>FA</u>	<u>FB</u>	<u>FC</u>	<u>FD</u>	<u>FE</u> FO

Figure 99. Code page 949 chart

Code Page 950

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	<u>NUL</u> 0000	<u>STX</u> 0001	<u>SOT</u> 0002	<u>ETX</u> 0003	<u>EOT</u> 0004	<u>ENQ</u> 0005	<u>ACK</u> 0006	<u>BEL</u> 0007	<u>BS</u> 0008	<u>HT</u> 0009	<u>LF</u> 000A	<u>VT</u> 000B	<u>FF</u> 000C	<u>CR</u> 000D	<u>SO</u> 000E	<u>SI</u> 000F
10	<u>DLE</u> 0010	<u>DC1</u> 0011	<u>DC2</u> 0012	<u>DC3</u> 0013	<u>DC4</u> 0014	<u>NAK</u> 0015	<u>SYN</u> 0016	<u>ETB</u> 0017	<u>CAN</u> 0018	<u>EM</u> 0019	<u>SUB</u> 001A	<u>ESC</u> 001B	<u>FS</u> 001C	<u>GS</u> 001D	<u>RS</u> 001E	<u>US</u> 001F
20	<u>SP</u> 0020	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0 0030	1 0031	2 0032	3 0033	4 0034	5 0035	6 0036	7 0037	8 0038	9 0039	:	;	< 003A	= 003B	> 003C	? 003D
40	@ 0040	À 0041	À 0042	À 0043	À 0044	È 0045	È 0046	È 0047	È 0048	È 0049	Ì 004A	Ì 004B	Ò 004C	Ò 004D	Ò 004E	Ó 004F
50	P 0050	Q 0051	R 0052	S 0053	T 0054	U 0055	V 0056	W 0057	X 0058	Y 0059	Z 005A	[005B	\ 005C] 005D	^ 005E	— 005F
60	~ 0060	a 0061	b 0062	c 0063	d 0064	e 0065	f 0066	g 0067	h 0068	i 0069	j 006A	k 006B	l 006C	m 006D	n 006E	o 006F
70	þ 0070	þ 0071	þ 0072	þ 0073	þ 0074	þ 0075	þ 0076	þ 0077	þ 0078	þ 0079	þ 007A	{ 007B	 007C	{ 007D	~ 007E	<u>DEL</u> 007F
80		<u>81</u>	<u>82</u>	<u>83</u>	<u>84</u>	<u>85</u>	<u>86</u>	<u>87</u>	<u>88</u>	<u>89</u>	<u>8A</u>	<u>8B</u>	<u>8C</u>	<u>8D</u>	<u>8E</u>	<u>8F</u>
90	<u>90</u>	<u>91</u>	<u>92</u>	<u>93</u>	<u>94</u>	<u>95</u>	<u>96</u>	<u>97</u>	<u>98</u>	<u>99</u>	<u>9A</u>	<u>9B</u>	<u>9C</u>	<u>9D</u>	<u>9E</u>	<u>9F</u>
A0	<u>A0</u>	<u>A1</u>	<u>A2</u>	<u>A3</u>	<u>A4</u>	<u>A5</u>	<u>A6</u>	<u>A7</u>	<u>A8</u>	<u>A9</u>	<u>AA</u>	<u>AB</u>	<u>AC</u>	<u>AD</u>	<u>AE</u>	<u>AF</u>
B0	<u>BO</u>	<u>B1</u>	<u>B2</u>	<u>B3</u>	<u>B4</u>	<u>B5</u>	<u>B6</u>	<u>B7</u>	<u>B8</u>	<u>B9</u>	<u>BA</u>	<u>BB</u>	<u>BC</u>	<u>BD</u>	<u>BE</u>	<u>BF</u>
C0	<u>CO</u>	<u>C1</u>	<u>C2</u>	<u>C3</u>	<u>C4</u>	<u>C5</u>	<u>C6</u>	<u>C7</u>	<u>C8</u>	<u>C9</u>	<u>CA</u>	<u>CB</u>	<u>CC</u>	<u>CD</u>	<u>CE</u>	<u>CF</u>
D0	<u>DO</u>	<u>D1</u>	<u>D2</u>	<u>D3</u>	<u>D4</u>	<u>D5</u>	<u>D6</u>	<u>D7</u>	<u>D8</u>	<u>D9</u>	<u>DA</u>	<u>DB</u>	<u>DC</u>	<u>DD</u>	<u>DE</u>	<u>DF</u>
E0	<u>EO</u>	<u>E1</u>	<u>E2</u>	<u>E3</u>	<u>E4</u>	<u>E5</u>	<u>E6</u>	<u>E7</u>	<u>E8</u>	<u>E9</u>	<u>EA</u>	<u>EB</u>	<u>EC</u>	<u>ED</u>	<u>EE</u>	<u>EF</u>
F0	<u>FO</u>	<u>F1</u>	<u>F2</u>	<u>F3</u>	<u>F4</u>	<u>F5</u>	<u>F6</u>	<u>F7</u>	<u>F8</u>	<u>F9</u>	<u>FA</u>	<u>FB</u>	<u>FC</u>	<u>FD</u>	<u>FE</u>	

Figure 100. Code page 950 chart

Code Page 951

A1A1	A1A2	A1A3	A1A4	A1A5	A1A6	A1A7	A1A8	A1A9	A1AA	A1AB	A1AC	A1AD	A1AE	A1AF	A1BO
„	„	„	„	„	„	„	„	„	„	„	„	„	„	„	„
A1B1	A1B2	A1B3	A1B4	A1B5	A1B6	A1B7	A1B8	A1B9	A1BA	A1BB	A1BC	A1BD	A1BE	A1BF	A1CO
„	„	„	„	„	„	„	„	„	„	„	„	„	„	„	„
A1C1	A1C2	A1C3	A1C4	A1C5	A1C6	A1C7	A1C8	A1C9	A1CA	A1CB	A1CC	A1CD	A1D7	A1D8	A1D9
„	„	„	„	„	„	„	„	„	„	„	„	„	„	„	„
A1DA	A1DB	A1DC	A1DD	A1DE	A1DF	A1E0	A1E1	A1E2	A1E3	A1E4	A1E5	A1E6	A1E7	A1E8	A1E9
„	„	„	„	„	„	„	„	„	„	„	„	„	„	„	„
A1EA	A1EB	A3A1	A3A2	A3A3	A3A4	A3A5	A3A6	A3A7	A3A8	A3A9	A3AA	A3AB	A3AC	A3AD	A3AE
„	„	„	„	„	„	„	„	„	„	„	„	„	„	„	„
A3AF	A3B0	A3B1	A3B2	A3B3	A3B4	A3B5	A3B6	A3B7	A3B8	A3B9	A3BA	A3BB	A3BC	A3BD	A3BE
„	„	„	„	„	„	„	„	„	„	„	„	„	„	„	„
A3BF	A3C0	A3C1	A3C2	A3C3	A3C4	A3C5	A3C6	A3C7	A3C8	A3C9	A3CA	A3CB	A3CC	A3CD	A3CE
„	„	„	„	„	„	„	„	„	„	„	„	„	„	„	„
A3CF	A3D0	A3D1	A3D2	A3D3	A3D4	A3D5	A3D6	A3D7	A3D8	A3D9	A3DA	A3DB	A3DC	A3DD	A3DE
„	„	„	„	„	„	„	„	„	„	„	„	„	„	„	„
A3DF	A3E0	A3E1	A3E2	A3E3	A3E4	A3E5	A3E6	A3E7	A3E8	A3E9	A3EA	A3EB	A3EC	A3ED	A3EE
„	„	„	„	„	„	„	„	„	„	„	„	„	„	„	„
A3EF	A3F0	A3F1	A3F2	A3F3	A3F4	A3F5	A3F6	A3F7	A3F8	A3F9	A3FA	A3FB	A3FC	A3FD	A3FE
„	„	„	„	„	„	„	„	„	„	„	„	„	„	„	„
A4A1	A4A2	A4A3	A4A4	A4A5	A4A6	A4A7	A4A8	A4A9	A4AA	A4AB	A4AC	A4AD	A4AE	A4AF	A4B0
„	„	„	„	„	„	„	„	„	„	„	„	„	„	„	„
A4B1	A4B2	A4B3	A4B4	A4B5	A4B6	A4B7	A4B8	A4B9	A4BA	A4BB	A4BC	A4BD	A4BE	A4BF	A4C0
„	„	„	„	„	„	„	„	„	„	„	„	„	„	„	„
A4C1	A4C2	A4C3	A4C4	A4C5	A4C6	A4C7	A4C8	A4C9	A4CA	A4CB	A4CC	A4CD	A4CE	A4CF	A4D0
„	„	„	„	„	„	„	„	„	„	„	„	„	„	„	„
A4D1	A4D2	A4D3	AAA1	AAA2	AAA3	AAA4	AAA5	AAA6	AAA7	AAA8	AAA9	AAAA	AAAB	AAAC	AAAD
„	„	„	„	„	„	„	„	„	„	„	„	„	„	„	„
AAAEE	AAAFF	AAB0	AAB1	AAB2	AAB3	AAB4	AAB5	AAB6	AAB7	AAB8	AAB9	AABA	AABB	AABC	AABD
„	„	„	„	„	„	„	„	„	„	„	„	„	„	„	„
AABE	AABF	AAC0	AAC1	AAC2	AAC3	AAC4	AAC5	AAC6	AAC7	AAC8	AAC9	AACA	AACB	AACC	AACD
„	„	„	„	„	„	„	„	„	„	„	„	„	„	„	„

Figure 101. Special Symbols Alphanumeric/Numerics/Jamo/Hiragana/Hanja (1 of 3)



Figure 102. Special Symbols Alphanumeric/Numerics/Jamo/Hiragana/Hanja (2 of 3)

D9CD	D9ED	D9FE	DAA6	D9FD	DAAA	DAB1	DAE2	DAE3	DAF5	DAF7	DBA1	DBB0	DBC3	DBD5	DBDD
DBE3	DBF6	DCA8	DCAC	DCC3	DCD7	DDBB	DDC2	DDD5	DDD7	DDE1	DEA8	DEAA	DEBC	DEC0	DEC5
DECC	DED8	DEE4	DEE7	DFA7	DFA9	DFB2	DFBE	DFC2	DFE6	EOB4	EOBB	EOE2	EOF7	E1AA	E1B3
E1B4	E1B6	E1BC	E1EA	E1F4	E2A2	E2A5	E2A6	E2A9	E2ED	E2F7	E3A7	E3AF	E3B7	E3BC	E3C1
E3CB	E3D2	E3E1	E3E6	E3E9	E3F7	E4A8	E4CC	E4D0	E4E7	E4FE	E5B3	E5D5	E5EB	E5F6	E6B6
E7BD	E7E8	E7E9	E7ED	E8A6	E8C7	E8D9	E8DD	E8E2	E9C4	E9D3	AAAA	EAAB	EAAC	EAC5	EAF3
EBBF	EBDE	EBFD	ECA3	ECA4	ECCC	ECD1	ECDA	ECE3	ECE9	ECED	ECFD	ECAD	EDAE	EDBA	EDCO
EDC2	EDD1	EDDE	EDE5	EEA2	EEA4	EEA7	EEDD	EEE9	EEF1	EFB1	EFB7	EFC1	EFC3	EFD2	EFE1
EFF1	EFF4	F0A4	F0FA	F0FB	F1A7	F1AC	F1BB	F1CE	F1DE	F1E9	F2A4	F2A5	F2A8	F2AD	F3AC
F4A1	F4B6	F4F0	F4F8	F5C5	F5D5	F5E6	F5F0	F5F3	F6A2	F6B7	F6D2	F6E2	F7CF	F7D7	F7E5
F7FC	F8A1	F8A2	F8C1	F8CD	F8F6	F8F9	F8FA	F9A1	F9BB	F9BE	F9C3	F9D3	F9DC	F9EA	FABC
FADE	FAFO	FBA1	FBBC	FBDC	FBFD	FCB5	FCC0	FCDE	FCE5	FDAD	FDBA	FDCC			
現協形呼号火操活回会後訓休															

Figure 103. Special Symbols Alphanumeric/Numerics/Jamo/Hiragana/Hanja (3 of 3)

BOA1	BOA2	BOA3	BOA4	BOA5	BOA6	BOA7	BOA8	BOA9	BOAA	BOAB	BOAC	BOAD	BOAE	BOAF	BOBO
가	각	간	감	갈	감	감	감	감	갓	갓	강	갓	갓	갓	감
BOB1	BOB2	BOB3	BOB4	BOB5	BOB6	BOB7	BOB8	BOB9	BOBA	BOBB	BOBC	BOBD	BOBE	BOBF	BOCO
갓	꽈	개	꽕	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔
BOC1	BOC2	BOC3	BOC4	BOC5	BOC6	BOC7	BOC8	BOC9	BOCA	BOCB	BOCC	BOCD	BOCE	BOCF	BODO
꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔
BOD1	BOD2	BOD3	BOD4	BOD5	BOD6	BOD7	BOD8	BOD9	BODA	BODB	BODC	BODD	BODE	BODF	BOEO
꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔
BOE1	BOE2	BOE3	BOE4	BOE5	BOE6	BOE7	BOE8	BOE9	BOEA	BOEB	BOEC	BOED	BOEE	BOEF	BOFO
꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔
BOF1	BOF2	BOF3	BOF4	BOF5	BOF6	BOF7	BOF8	BOF9	BOFA	BOFB	BOFC	BOFD	BOFE	B1A1	B1A2
꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔
B1A3	B1A4	B1A5	B1A6	B1A7	B1A8	B1A9	B1AA	B1AB	B1AC	B1AD	B1AE	B1AF	B1B0	B1B1	B1B2
갓	꽝	꽕	꽌	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔
B1B3	B1B4	B1B5	B1B6	B1B7	B1B8	B1B9	B1BA	B1BB	B1BC	B1BD	B1BE	B1BF	B1C0	B1C1	B1C2
긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐
B1C3	B1C4	B1C5	B1C6	B1C7	B1C8	B1C9	B1CA	B1CB	B1CC	B1CD	B1CE	B1CF	B1DO	B1D1	B1D2
긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐
B1D3	B1D4	B1D5	B1D6	B1D7	B1D8	B1D9	B1DA	B1DB	B1DC	B1DD	B1DE	B1DF	B1EO	B1E1	B1E2
긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐
B1E3	B1E4	B1E5	B1E6	B1E7	B1E8	B1E9	B1EA	B1EB	B1EC	B1ED	B1EE	B1EF	B1FO	B1F1	B1F2
긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐	긐
B1F3	B1F4	B1F5	B1F6	B1F7	B1F8	B1F9	B1FA	B1FB	B1FC	B1FD	B1FE	B2A1	B2A2	B2A3	B2A4
꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔
B2A5	B2A6	B2A7	B2A8	B2A9	B2AA	B2AB	B2AC	B2AD	B2AE	B2AF	B2B0	B2B1	B2B2	B2B3	B2B4
꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔
B2B5	B2B6	B2B7	B2B8	B2B9	B2BA	B2BB	B2BC	B2BD	B2BE	B2BF	B2CO	B2C1	B2C2	B2C3	B2C4
꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔
B2C5	B2C6	B2C7	B2C8	B2C9	B2CA	B2CB	B2CC	B2CD	B2CE	B2CF	B2DO	B2D1	B2D2	B2D3	B2D4
꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔
B2D5	B2D6	B2D7	B2D8	B2D9	B2DA	B2DB	B2DC	B2DD	B2DE	B2DF	B2EO	B2E1	B2E2	B2E3	B2E4
꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔	꺔

Figure 104. Hangeul (1 of 10)

B2E5	B2E6	B2E7	B2E8	B2E9	B2EA	B2EB	B2EC	B2ED	B2EE	B2EF	B2FO	B2F1	B2F2	B2F3	B2F4
깂	깂	깂	깂	깂	깂	깂	깂	깂	깂	깂	깂	깂	깂	깂	깂
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
B3A7	B3A8	B3A9	B3AA	B3AB	B3AC	B3AD	B3AE	B3AF	B3B0	B3B1	B3B2	B3B3	B3B4	B3B5	B3B6
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
B3C7	B3C8	B3C9	B3CA	B3CB	B3CC	B3CD	B3CE	B3CF	B3D0	B3D1	B3D2	B3D3	B3D4	B3D5	B3D6
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
B3D7	B3D8	B3D9	B3DA	B3DB	B3DC	B3DD	B3DE	B3DF	B3EO	B3E1	B3E2	B3E3	B3E4	B3E5	B3E6
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
B3E7	B3E8	B3E9	B3EA	B3EB	B3EC	B3ED	B3EE	B3EF	B3FO	B3F1	B3F2	B3F3	B3F4	B3F5	B3F6
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
B3F7	B3F8	B3F9	B3FA	B3FB	B3FC	B3FD	B3FE	B4A1	B4A2	B4A3	B4A4	B4A5	B4A6	B4A7	B4A8
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
B4A9	B4AA	B4AB	B4AC	B4AD	B4AE	B4AF	B4BO	B4B1	B4B2	B4B3	B4B4	B4B5	B4B6	B4B7	B4B8
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
B4B9	B4BA	B4BB	B4BC	B4BD	B4BE	B4BF	B4CO	B4C1	B4C2	B4C3	B4C4	B4C5	B4C6	B4C7	B4C8
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
B4C9	B4CA	B4CB	B4CC	B4CD	B4CE	B4CF	B4DO	B4D1	B4D2	B4D3	B4D4	B4D5	B4D6	B4D7	B4D8
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
B4D9	B4DA	B4DB	B4DC	B4DD	B4DE	B4DF	B4EO	B4E1	B4E2	B4E3	B4E4	B4E5	B4E6	B4E7	B4E8
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
B4E9	B4EA	B4EB	B4EC	B4ED	B4EE	B4EF	B4FO	B4F1	B4F2	B4F3	B4F4	B4F5	B4F6	B4F7	B4F8
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
B4F9	B4FA	B4FB	B4FC	B4FD	B4FE	B5A1	B5A2	B5A3	B5A4	B5A5	B5A6	B5A7	B5A8	B5A9	B5AA
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
B5AB	B5AC	B5AD	B5AE	B5AF	B5B0	B5B1	B5B2	B5B3	B5B4	B5B5	B5B6	B5B7	B5B8	B5B9	B5BA
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃
B5BB	B5BC	B5BD	B5BE	B5BF	B5C0	B5C1	B5C2	B5C3	B5C4	B5C5	B5C6	B5C7	B5C8	B5C9	B5CA
깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃	깃

Figure 105. Hangeul (2 of 10)

B5CB	B5CC	B5CD	B5CE	B5CF	B5D0	B5D1	B5D2	B5D3	B5D4	B5D5	B5D6	B5D7	B5D8	B5D9	B5DA
₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩
B5DB	B5DC	B5DD	B5DE	B5DF	B5E0	B5E1	B5E2	B5E3	B5E4	B5E5	B5E6	B5E7	B5E8	B5E9	B5EA
₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩
B5EB	B5EC	B5ED	B5EE	B5EF	B5F0	B5F1	B5F2	B5F3	B5F4	B5F5	B5F6	B5F7	B5F8	B5F9	B5FA
₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩
B5FB	B5FC	B5FD	B5FE	B6A1	B6A2	B6A3	B6A4	B6A5	B6A6	B6A7	B6A8	B6A9	B6AA	B6AB	B6AC
₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩
B6AD	B6AE	B6AF	B6B0	B6B1	B6B2	B6B3	B6B4	B6B5	B6B6	B6B7	B6B8	B6B9	B6BA	B6BB	B6BC
₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩
B6BD	B6BE	B6BF	B6C0	B6C1	B6C2	B6C3	B6C4	B6C5	B6C6	B6C7	B6C8	B6C9	B6CA	B6CB	B6CC
₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩
B6CD	B6CE	B6CF	B6D0	B6D1	B6D2	B6D3	B6D4	B6D5	B6D6	B6D7	B6D8	B6D9	B6DA	B6DB	B6DC
₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩
B6DD	B6DE	B6DF	B6E0	B6E1	B6E2	B6E3	B6E4	B6E5	B6E6	B6E7	B6E8	B6E9	B6EA	B6EB	B6EC
₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩
B6ED	B6EE	B6EF	B6FO	B6F1	B6F2	B6F3	B6F4	B6F5	B6F6	B6F7	B6F8	B6F9	B6FA	B6FB	B6FC
₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩
B6FD	B6FE	B7A1	B7A2	B7A3	B7A4	B7A5	B7A6	B7A7	B7A8	B7A9	B7AA	B7AB	B7AC	B7AD	B7AE
₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩
B7AF	B7B0	B7B1	B7B2	B7B3	B7B4	B7B5	B7B6	B7B7	B7B8	B7B9	B7BA	B7BB	B7BC	B7BD	B7BE
₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩
B7BF	B7C0	B7C1	B7C2	B7C3	B7C4	B7C5	B7C6	B7C7	B7C8	B7C9	B7CA	B7CB	B7CC	B7CD	B7CE
₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩
B7CF	B7D0	B7D1	B7D2	B7D3	B7D4	B7D5	B7D6	B7D7	B7D8	B7D9	B7DA	B7DB	B7DC	B7DD	B7DE
₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩
B7DF	B7EO	B7E1	B7E2	B7E3	B7E4	B7E5	B7E6	B7E7	B7E8	B7E9	B7EA	B7EB	B7EC	B7ED	B7EE
₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩
B7EF	B7FO	B7F1	B7F2	B7F3	B7F4	B7F5	B7F6	B7F7	B7F8	B7F9	B7FA	B7FB	B7FC	B7FD	B7FE
₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩
B8A1	B8A2	B8A3	B8A4	B8A5	B8A6	B8A7	B8A8	B8A9	B8AA	B8AB	B8AC	B8AD	B8AE	B8AF	B8B0
₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩

Figure 106. Hangeul (3 of 10)

B8B1	B8B2	B8B3	B8B4	B8B5	B8B6	B8B7	B8B8	B8B9	B8BA	B8BB	B8BC	B8BD	B8BE	B8BF	B8C0
리	림	립	릿	링	마	막	만	만	만	말	말	망	망	맙	맙
B8C1	B8C2	B8C3	B8C4	B8C5	B8C6	B8C7	B8C8	B8C9	B8CA	B8CB	B8CC	B8CD	B8CE	B8CF	B8D0
마	막	만	만	매	막	만	매	만	매	만	매	망	매	맙	마
B8D1	B8D2	B8D3	B8D4	B8D5	B8D6	B8D7	B8D8	B8D9	B8DA	B8DB	B8DC	B8DD	B8DE	B8DF	B8E0
말	마	미	먼	먼	금	금	금	금	금	금	금	금	금	맥	먼
B8E1	B8E2	B8E3	B8E4	B8E5	B8E6	B8E7	B8E8	B8E9	B8EA	B8EB	B8EC	B8ED	B8EE	B8EF	B8F0
금	에	에	에	에	에	에	에	에	에	에	에	에	에	에	모
B8F1	B8F2	B8F3	B8F4	B8F5	B8F6	B8F7	B8F8	B8F9	B8FA	B8FB	B8FC	B8FD	B8FE	B9A1	B9A2
목	몫	몬	몰	몰	몸	몸	못	몽	마	완	왔	왕	미	원	월
B9A3	B9A4	B9A5	B9A6	B9A7	B9A8	B9A9	B9AA	B9AB	B9AC	B9AD	B9AE	B9AF	B9B0	B9B1	B9B2
월	윗	윙	묘	문	문	문	묘	묘	무	무	무	무	문	물	물
B9B3	B9B4	B9B5	B9B6	B9B7	B9B8	B9B9	B9BA	B9BB	B9BC	B9BD	B9BE	B9BF	B9CO	B9C1	B9C2
물	물	물	물	물	물	물	한	한	한	한	한	한	한	한	류
B9C3	B9C4	B9C5	B9C6	B9C7	B9C8	B9C9	B9CA	B9CB	B9CC	B9CD	B9CE	B9CF	B9DO	B9D1	B9D2
류	유	유	유	으	으	으	음	음	미	미	미	민	민	밀	밀
B9D3	B9D4	B9D5	B9D6	B9D7	B9D8	B9D9	B9DA	B9DB	B9DC	B9DD	B9DE	B9DF	B9EO	B9E1	B9E2
밀	밋	밍	미	미	민	바	바	바	바	바	바	바	바	黠	黠
B9E3	B9E4	B9E5	B9E6	B9E7	B9E8	B9E9	B9EA	B9EB	B9EC	B9ED	B9EE	B9EF	B9FO	B9F1	B9F2
黠	밥	밥	밥	밥	밥	밥	밥	밥	밥	밥	밥	밥	밥	黠	黠
B9F3	B9F4	B9F5	B9F6	B9F7	B9F8	B9F9	B9FA	B9FB	B9FC	B9FD	B9FE	BAA1	BAA2	BAA3	BAA4
黠	반	반	반	반	반	반	반	반	반	반	반	반	반	반	黠
BAA5	BAA6	BAA7	BAA8	BAA9	BAAA	BAAB	BAAC	BAAD	BAAE	BAAF	BABO	BAB1	BAB2	BAB3	BAB4
黠	반	반	반	반	반	반	반	반	반	반	반	반	반	반	黠
BAB5	BAB6	BAB7	BAB8	BAB9	BABA	BABB	BABC	BABD	BABE	BABF	BACO	BAC1	BAC2	BAC3	BAC4
黠	반	반	반	반	반	반	반	반	반	반	반	반	반	黠	黠
BAC5	BAC6	BAC7	BAC8	BAC9	BACA	BACB	BACC	BACD	BACE	BACF	BADO	BAD1	BAD2	BAD3	BAD4
黠	반	반	반	반	반	반	반	반	반	반	반	반	반	黠	黠
BAD5	BAD6	BAD7	BAD8	BAD9	BADA	BADB	BADC	BADD	BADE	BADF	BAEO	BAE1	BAE2	BAE3	BAE4
黠	반	반	반	반	반	반	반	반	반	반	반	반	반	黠	黠
BAE5	BAE6	BAE7	BAE8	BAE9	BAEA	BAEB	BAEC	BAED	BAEE	BAEF	BAFO	BAF1	BAF2	BAF3	BAF4
黠	반	반	반	반	반	반	반	반	반	반	반	반	반	黠	黠

Figure 107. Hangeul (4 of 10)

BAF5	BAF6	BAF7	BAF8	BAF9	BAFA	BAFB	BAFC	BAFD	BAFE	BBA1	BBA2	BBA3	BBA4	BBA5	BBA6
BBA7	BBA8	BBA9	BBAA	BBAB	BBAC	BBAD	BBAE	BBAF	BBBO	BBB1	BBB2	BBB3	BBB4	BBB5	BBB6
BBB7	BBB8	BBB9	BBBB	BBBB	BBBC	BBBD	BBBE	BBBF	BBCO	BBC1	BBC2	BBC3	BBC4	BBC5	BBC6
BBC7	BBC8	BBC9	BBCA	BBCB	BBCC	BBCD	BBCE	BBCF	BBDO	BBD1	BBD2	BBD3	BBD4	BBD5	BBD6
BBD7	BBD8	BBD9	BBDA	BBDB	BBDC	BBDD	BBDE	BBDF	BBEO	BBE1	BBE2	BBE3	BBE4	BBE5	BBE6
BBE7	BBE8	BBE9	BBEA	BEBB	BBEC	BBED	BBEE	BBEF	BBFO	BBF1	BBF2	BBF3	BBF4	BBF5	BBF6
BBF7	BBF8	BBF9	BBFA	BBFB	BBFC	BBFD	BBFE	BCA1	BCA2	BCA3	BCA4	BCA5	BCA6	BCA7	BCA8
BCA9	BCAA	BCAB	BCAC	BCAD	BCAE	BCAF	BCBO	BCB1	BCB2	BCB3	BCB4	BCB5	BCB6	BCB7	BCB8
BCB9	BCBA	BCBB	BCBC	BCBD	BCBE	BCBF	BCCO	BCC1	BCC2	BCC3	BCC4	BCC5	BCC6	BCC7	BCC8
BCC9	BCCA	BCCB	BCCC	BCCD	BCCE	BCCF	BCDO	BCD1	BCD2	BCD3	BCD4	BCD5	BCD6	BCD7	BCD8
BCD9	BCDA	BCDB	BCDC	BCDD	BCDE	BCDF	BCEO	BCE1	BCE2	BCE3	BCE4	BCE5	BCE6	BCE7	BCE8
BCE9	BCEA	BCEB	BCEC	BCED	BCEE	BCEF	BCFO	BCF1	BCF2	BCF3	BCF4	BCF5	BCF6	BCF7	BCF8
BCF9	BCFA	BCFB	BCFC	BCFD	BCFE	BDA1	BDA2	BDA3	BDA4	BDA5	BDA6	BDA7	BDA8	BDA9	BDA
BDAB	BDAC	BDAD	BDAE	BDAF	BDBO	BDB1	BDB2	BDB3	BDB4	BDB5	BDB6	BDB7	BDB8	BDB9	BDBA
BDBB	BDBC	BDBD	BDBE	BDBF	BDCO	BDC1	BDC2	BDC3	BDC4	BDC5	BDC6	BDC7	BDC8	BDC9	BDC
BDCB	BDCC	BDCD	BDCE	BDCF	BDDO	BDD1	BDD2	BDD3	BDD4	BDD5	BDD6	BDD7	BDD8	BDD9	BDDA

Figure 108. Hangeul (5 of 10)

BDBB	BDDC	BDDD	BDDE	BDDF	BDE0	BDE1	BDE2	BDE3	BDE4	BDE5	BDE6	BDE7	BDE8	BDE9	BDEA
◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀
BDEB	BDEC	BDED	BDEE	BDEF	BDF0	BDF1	BDF2	BDF3	BDF4	BDF5	BDF6	BDF7	BDF8	BDF9	BDFA
◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀
BDFB	BDFC	BDFD	BDFE	BEA1	BEA2	BEA3	BEA4	BEA5	BEA6	BEA7	BEA8	BEA9	BEAA	BEAB	BEAC
◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀
BEAD	BEAE	BEAF	BEBO	BEB1	BEB2	BEB3	BEB4	BEB5	BEB6	BEB7	BEB8	BEB9	BEBA	BEBB	BEBC
◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀
BEBD	BEBE	BEBF	BEC0	BEC1	BEC2	BEC3	BEC4	BEC5	BEC6	BEC7	BEC8	BEC9	BECA	BECB	BECC
◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀
BECD	BECE	BECF	BED0	BED1	BED2	BED3	BED4	BED5	BED6	BED7	BED8	BED9	BEDA	BEDB	BEDC
◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀
BEDD	BEDE	BEDF	BEO0	BEE1	BEE2	BEE3	BEE4	BEE5	BEE6	BEE7	BEE8	BEE9	BEAA	BEEB	BEEC
◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀
BEED	BEEE	BEEF	BEOF	BEF1	BEF2	BEF3	BEF4	BEF5	BEF6	BEF7	BEF8	BEF9	BEFA	BEFB	BEFC
◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀
BEFD	BEFE	BFA1	BFA2	BFA3	BFA4	BFA5	BFA6	BFA7	BFA8	BFA9	BFAB	BFAC	BFAD	BFAE	
◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	
BFAF	BFB0	BFB1	BFB2	BFB3	BFB4	BFB5	BFB6	BFB7	BFB8	BFB9	BFBA	BFBB	BFBC	BFBD	BFBE
◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀
BFBF	BFC0	BFC1	BFC2	BFC3	BFC4	BFC5	BFC6	BFC7	BFC8	BFC9	BFCA	BFCB	BFCC	BFCD	BFCE
◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀
BFCF	BFD0	BFD1	BFD2	BFD3	BFD4	BFD5	BFD6	BFD7	BFD8	BFD9	BFDA	BFDB	BFDC	BFDD	BFDE
◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀
BFDF	BFE0	BFE1	BFE2	BFE3	BFE4	BFE5	BFE6	BFE7	BFE8	BFE9	BFEA	BFEB	BFEC	BFED	BFEE
◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀
BFEF	BFF0	BFF1	BFF2	BFF3	BFF4	BFF5	BFF6	BFF7	BFF8	BFF9	BFFA	BFFB	BFFC	BFFD	BFFE
◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀
COA1	COA2	COA3	COA4	COA5	COA6	COA7	COA8	COA9	COAA	COAB	COAC	COAD	COAE	COAF	COBO
◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	
COB1	COB2	COB3	COB4	COB5	COB6	COB7	COB8	COB9	COBA	COBB	COBC	COBD	COBE	COBF	COCO
◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	◀◀◀	

Figure 109. Hangeul (6 of 10)

COC1	COC2	COC3	COC4	COC5	COC6	COC7	COC8	COC9	COCA	COCB	COCC	COCD	COCE	COCF	CODO
을	을	을	을	을	을	의	원	월	임	의	이	익	인	일	의
COD1	COD2	COD3	COD4	COD5	COD6	COD7	COD8	COD9	CODA	CODB	CODC	CODD	CODE	CODF	COEO
의	의	의	의	의	의	의	의	의	자	작	잔	잔	잔	잔	잔
COE1	COE2	COE3	COE4	COE5	COE6	COE7	COE8	COE9	COEA	COEB	COEC	COED	COEE	COEF	COFO
장	장	장	장	장	장	재	재	재	재	재	재	재	재	재	장
COF1	COF2	COF3	COF4	COF5	COF6	COF7	COF8	COF9	COFA	COFB	COFC	COFD	COFE	C1A1	C1A2
자	자	자	자	자	자	재	재	재	재	재	재	재	재	재	자
C1A3	C1A4	C1A5	C1A6	C1A7	C1A8	C1A9	C1AA	C1AB	C1AC	C1AD	C1AE	C1AF	C1B0	C1B1	C1B2
제	제	제	제	제	제	제	제	제	제	제	제	제	제	제	제
C1B3	C1B4	C1B5	C1B6	C1B7	C1B8	C1B9	C1BA	C1BB	C1BC	C1BD	C1BE	C1BF	C1C0	C1C1	C1C2
정	정	제	조	죽	존	줄	줄	줄	줄	줄	줄	줄	줄	줄	자
C1C3	C1C4	C1C5	C1C6	C1C7	C1C8	C1C9	C1CA	C1CB	C1CC	C1CD	C1CE	C1CF	C1D0	C1D1	C1D2
작	작	작	작	왕	왜	왔	왔	왔	왔	왔	왔	왔	왔	왔	교
C1D3	C1D4	C1D5	C1D6	C1D7	C1D8	C1D9	C1DA	C1DB	C1DC	C1DD	C1DE	C1DF	C1E0	C1E1	C1E2
표	표	표	표	표	표	표	표	표	표	표	표	표	표	표	표
C1E3	C1E4	C1E5	C1E6	C1E7	C1E8	C1E9	C1EA	C1EB	C1EC	C1ED	C1EE	C1EF	C1FO	C1F1	C1F2
직	직	직	직	직	직	직	직	직	직	직	직	직	직	직	직
C1F3	C1F4	C1F5	C1F6	C1F7	C1F8	C1F9	C1FA	C1FB	C1FC	C1FD	C1FE	C2A1	C2A2	C2A3	C2A4
표	표	표	표	표	표	표	표	표	표	표	표	표	표	표	표
C2A5	C2A6	C2A7	C2A8	C2A9	C2AA	C2AB	C2AC	C2AD	C2AE	C2AF	C2BO	C2B1	C2B2	C2B3	C2B4
파	파	파	파	파	파	파	파	파	파	파	파	파	파	파	파
C2B5	C2B6	C2B7	C2B8	C2B9	C2BA	C2BB	C2BC	C2BD	C2BE	C2BF	C2CO	C2C1	C2C2	C2C3	C2C4
짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜
C2C5	C2C6	C2C7	C2C8	C2C9	C2CA	C2CB	C2CC	C2CD	C2CE	C2CF	C2DO	C2D1	C2D2	C2D3	C2D4
짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜
C2D5	C2D6	C2D7	C2D8	C2D9	C2DA	C2DB	C2DC	C2DD	C2DE	C2DF	C2EO	C2E1	C2E2	C2E3	C2E4
짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜
C2E5	C2E6	C2E7	C2E8	C2E9	C2EA	C2EB	C2EC	C2ED	C2EE	C2EF	C2FO	C2F1	C2F2	C2F3	C2F4
짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜
C2F5	C2F6	C2F7	C2F8	C2F9	C2FA	C2FB	C2FC	C2FD	C2FE	C3A1	C3A2	C3A3	C3A4	C3A5	C3A6
짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜	짜

Figure 110. Hangeul (7 of 10)

C3A7	C3A8	C3A9	C3AA	C3AB	C3AC	C3AD	C3AE	C3AF	C3B0	C3B1	C3B2	C3B3	C3B4	C3B5	C3B6
ퟤ	ퟥ	ퟦ	ퟧ	ퟨ	ퟩ	ퟪ	ퟫ	ퟥ	ퟦ	ퟧ	ퟨ	ퟩ	ퟪ	ퟫ	ퟥ
C3B7	C3B8	C3B9	C3BA	C3BB	C3BC	C3BD	C3BE	C3BF	C3C0	C3C1	C3C2	C3C3	C3C4	C3C5	C3C6
ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ
C3C7	C3C8	C3C9	C3CA	C3CB	C3CC	C3CD	C3CE	C3CF	C3D0	C3D1	C3D2	C3D3	C3D4	C3D5	C3D6
ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ
C3D7	C3D8	C3D9	C3DA	C3DB	C3DC	C3DD	C3DE	C3DF	C3E0	C3E1	C3E2	C3E3	C3E4	C3E5	C3E6
ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ
C3E7	C3E8	C3E9	C3EA	C3EB	C3EC	C3ED	C3EE	C3EF	C3F0	C3F1	C3F2	C3F3	C3F4	C3F5	C3F6
ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ
C3F7	C3F8	C3F9	C3FA	C3FB	C3FC	C3FD	C3FE	C4A1	C4A2	C4A3	C4A4	C4A5	C4A6	C4A7	C4A8
ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ
C4A9	C4AA	C4AB	C4AC	C4AD	C4AE	C4AF	C4B0	C4B1	C4B2	C4B3	C4B4	C4B5	C4B6	C4B7	C4B8
ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ
C4B9	C4BA	C4BB	C4BC	C4BD	C4BE	C4BF	C4CO	C4C1	C4C2	C4C3	C4C4	C4C5	C4C6	C4C7	C4C8
ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ
C4C9	C4CA	C4CB	C4CC	C4CD	C4CE	C4CF	C4D0	C4D1	C4D2	C4D3	C4D4	C4D5	C4D6	C4D7	C4D8
ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ
C4D9	C4DA	C4DB	C4DC	C4DD	C4DE	C4DF	C4EO	C4E1	C4E2	C4E3	C4E4	C4E5	C4E6	C4E7	C4E8
ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ
C4E9	C4EA	C4EB	C4EC	C4ED	C4EE	C4FF	C4FO	C4F1	C4F2	C4F3	C4F4	C4F5	C4F6	C4F7	C4F8
ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ
C4F9	C4FA	C4FB	C4FC	C4FD	C4FE	C5A1	C5A2	C5A3	C5A4	C5A5	C5A6	C5A7	C5A8	C5A9	C5AA
ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ
C5AB	C5AC	C5AD	C5AE	C5AF	C5B0	C5B1	C5B2	C5B3	C5B4	C5B5	C5B6	C5B7	C5B8	C5B9	C5BA
ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ
C5BB	C5BC	C5BD	C5BE	C5BF	C5C0	C5C1	C5C2	C5C3	C5C4	C5C5	C5C6	C5C7	C5C8	C5C9	C5CA
ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ
C5CB	C5CC	C5CD	C5CE	C5CF	C5D0	C5D1	C5D2	C5D3	C5D4	C5D5	C5D6	C5D7	C5D8	C5D9	C5DA
ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ
C5DB	C5DC	C5DD	C5DE	C5DF	C5E0	C5E1	C5E2	C5E3	C5E4	C5E5	C5E6	C5E7	C5E8	C5E9	C5EA
ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ	ퟧ

Figure 111. Hangeul (8 of 10)

C5EB	C5EC	C5ED	C5EE	C5EF	C5FO	C5F1	C5F2	C5F3	C5F4	C5F5	C5F6	C5F7	C5F8	C5F9	C5FA
ㅌ	ㅍ	ㅎ	ㅋ	ㅌ	ㅌ	ㅍ	ㅎ	ㅋ	ㅌ	ㅌ	ㅍ	ㅎ	ㅋ	ㅌ	ㅍ
C5FB	C5FC	C5FD	C5FE	C6A1	C6A2	C6A3	C6A4	C6A5	C6A6	C6A7	C6A8	C6A9	C6AA	C6AB	C6AC
ㅌ	ㅍ	ㅎ	ㅋ	ㅌ	ㅌ	ㅍ	ㅎ	ㅋ	ㅌ	ㅌ	ㅍ	ㅎ	ㅋ	ㅌ	ㅍ
C6AD	C6AE	C6AF	C6B0	C6B1	C6B2	C6B3	C6B4	C6B5	C6B6	C6B7	C6B8	C6B9	C6BA	C6BB	C6BC
ㅌ	ㅍ	ㅎ	ㅋ	ㅌ	ㅌ	ㅍ	ㅎ	ㅋ	ㅌ	ㅌ	ㅍ	ㅎ	ㅋ	ㅌ	ㅍ
C6BD	C6BE	C6BF	C6C0	C6C1	C6C2	C6C3	C6C4	C6C5	C6C6	C6C7	C6C8	C6C9	C6CA	C6CB	C6CC
ㅌ	ㅌ	ㅌ	ㅌ	ㅌ	ㅌ	ㅌ	ㅌ	ㅌ	ㅌ	ㅌ	ㅌ	ㅌ	ㅌ	ㅌ	ㅌ
C6CD	C6CE	C6CF	C6D0	C6D1	C6D2	C6D3	C6D4	C6D5	C6D6	C6D7	C6D8	C6D9	C6DA	C6DB	C6DC
파	파	파	파	파	파	파	파	파	파	파	파	파	파	파	파
C6DD	C6DE	C6DF	C6EO	C6E1	C6E2	C6E3	C6E4	C6E5	C6E6	C6E7	C6E8	C6E9	C6EA	C6EB	C6EC
뗙	뗙	뗙	뗙	뗙	뗙	뗙	뗙	뗙	뗙	뗙	뗙	뗙	뗙	뗙	뗙
C6ED	C6EE	C6EF	C6FO	C6F1	C6F2	C6F3	C6F4	C6F5	C6F6	C6F7	C6F8	C6F9	C6FA	C6FB	C6FC
뗙	뗙	뗙	뗙	뗙	뗙	뗙	뗙	뗙	뗙	뗙	뗙	뗙	뗙	뗙	뗙
C6FD	C6FE	C7A1	C7A2	C7A3	C7A4	C7A5	C7A6	C7A7	C7A8	C7A9	C7AA	C7AB	C7AC	C7AD	C7AE
뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔
C7AF	C7B0	C7B1	C7B2	C7B3	C7B4	C7B5	C7B6	C7B7	C7B8	C7B9	C7BA	C7BB	C7BC	C7BD	C7BE
뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔
C7BF	C7C0	C7C1	C7C2	C7C3	C7C4	C7C5	C7C6	C7C7	C7C8	C7C9	C7CA	C7CB	C7CC	C7CD	C7CE
뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔	뫔
C7CF	C7D0	C7D1	C7D2	C7D3	C7D4	C7D5	C7D6	C7D7	C7D8	C7D9	C7DA	C7DB	C7DC	C7DD	C7DE
하	하	한	한	한	한	한	한	한	한	한	해	한	한	한	한
C7DF	C7E0	C7E1	C7E2	C7E3	C7E4	C7E5	C7E6	C7E7	C7E8	C7E9	C7EA	C7EB	C7EC	C7ED	C7EE
하	하	하	하	하	하	하	하	하	하	하	한	한	한	한	한
C7EF	C7F0	C7F1	C7F2	C7F3	C7F4	C7F5	C7F6	C7F7	C7F8	C7F9	C7FA	C7FB	C7FC	C7FD	C7FE
한	한	한	한	한	한	한	한	한	한	한	한	한	한	한	한
C8A1	C8A2	C8A3	C8A4	C8A5	C8A6	C8A7	C8A8	C8A9	C8AA	C8AB	C8AC	C8AD	C8AE	C8AF	C8B0
한	한	한	한	한	한	한	한	한	한	한	한	한	한	한	한
C8B1	C8B2	C8B3	C8B4	C8B5	C8B6	C8B7	C8B8	C8B9	C8BA	C8BB	C8BC	C8BD	C8BE	C8BF	C8C0
한	한	한	한	한	한	한	한	한	한	한	한	한	한	한	한
C8C1	C8C2	C8C3	C8C4	C8C5	C8C6	C8C7	C8C8	C8C9	C8CA	C8CB	C8CC	C8CD	C8CE	C8CF	C8D0
한	한	한	한	한	한	한	한	한	한	한	한	한	한	한	한

Figure 112. Hangeul (9 of 10)

C8D1	C8D2	C8D3	C8D4	C8D5	C8D6	C8D7	C8D8	C8D9	C8DA	C8DB	C8DC	C8DD	C8DE	C8DF	C8E0
₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩
C8E1	C8E2	C8E3	C8E4	C8E5	C8E6	C8E7	C8E8	C8E9	C8EA	C8EB	C8EC	C8ED	C8EE	C8EF	C8FO
₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩
C8F1	C8F2	C8F3	C8F4	C8F5	C8F6	C8F7	C8F8	C8F9	C8FA	C8FB	C8FC	C8FD	C8FE		
₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩	₩		

Figure 113. Hangeul (10 of 10)

Code Page 1381

HEX DIGITS 1ST → 2ND ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0		(SP)	0	@	P	'	p	£								
-1		SF570000	!	1	A	Q	a	q	¬							
-2		SM760000	↑	"	2	B	R	b	r	¥						
-3		SP330000	!!	#	3	C	S	c	s	-						
-4		SF160000	\$	4	D	T	d	t	!							
-5		SF400000	%	5	E	U	e	u								
-6		SF410000	&	6	F	V	f	v								
-7		SF230000	'	7	G	W	g	w								
-8			(8	H	X	h	x								
-9		SF420000)	9	I	Y	i	y								
-A		SF140000	*	:	J	Z	j	z								
-B		SM720000	↖	+	;	K	[k	{							
-C		SM320000	↑	,	<	L	\	l								
-D		SF110000	-	=	M]	m	}								
-E		SM310000	→	.	>	N	^	n	~							
-F		SM300000	←	/	?	O	—	o								

Figure 114. Code page 1381 chart

Part 4. Appendixes

Appendix A. JavaPOS support for UnifiedPOS device statistics properties

Common properties

Table 279. Description of common properties

Common Properties	Description	Value
UnifiedPOSVersion	Version of the UnifiedPOS specification supported	UnifiedPOS Version (for example: 1.9.1)
DeviceCategory	Device category (for example: POSPrinter)	[Corresponding category]
ManufacturerName	Device manufacturer's name	Toshiba
ModelName	Device model name	See the section for the specific device.
SerialNumber	Device serial number	[Serial_Number]
ManufactureDate	Device manufacture date	See the section for the specific device.
FirmwareRevision	Device firmware revision	[Firmware_Version]
Interface	Device hardware interface (for example: USB)	RS232/RS485/USB/Proprietary
InstallationDate	Device installation date	Not supported
HoursPoweredCount	Number of hours powered On	Not supported
CommunicationErrorCount	Number of communication errors	Not supported

Note:

1. Interface property used the *Proprietary* value for PS/2 and Embedded devices.
2. SerialNumber property for USB devices may not match the device label, except for the 4610 POSPrinter.
3. The ManufactureDate format for the [Printer/MICR/CheckScanner/CashDrawer] is WWYY, where WW is a two-digit representation for the week of the year the printer was manufactured, and YY is the last two digits of the year it was manufactured.
4. EIA232 hardware interface use RS232 as a value.

Support for the following properties applies to all devices:

- CapStatisticsReporting: Yes
- CapUpdateStatistics: No

Cash drawer

Table 280. Cash drawer

System/device	Bus	Model name value	
		Port 3A	Port 3B
SurePOS 730/750	USB	4800-73X/75X - A	4800-73X/75X - B
POSPrinter 4610	USB / EIA232 / RS485, Network	4610 - A	4610 - B
SurePOS 300	Embedded	4800-72x/74x/78x, 4810-3xx	4800-72x/74x/78x, 4810-3xx

Table 280. Cash drawer (continued)

SurePOS 500/600/Kiosk	EIA232	500/600 - A	500/600 - B
SurePOS 72x/74x/78x	Embedded	4800-72x/74x/78x, 4810-3xx - A	4800-72x/74x/78x, 4810-3xx - B
SurePOS 4694	RS485	4694-2xx/3xx - A	4694-2xx/3xx - B
SurePOS 4674	Embedded	4674 - A	4674 - B
SureOne 4614/4615 SurePOS 100	Embedded	461x	
AnyPlace POS Hub	USB	AnyPlace Hub	
Toshiba Cash Drawer	USB	Toshiba Cash Drawer - A	Toshiba Cash Drawer - B
TCx 700	USB / Embedded	4900-7x6-A	4900-7x6-B

Table 281. Cash drawer common properties

Common Properties	SurePOS 730/750 USB	POS Printer 4610 EIA232/ RS485/ USB/ Network	SurePOS 300 Embedded	SurePOS 500/ 600/ Kiosk EIA232	SurePOS 72x/ 74x/ 78x Embedded	SurePOS 4694 - RS485	SurePOS 4674 Embedded	SureOne/ SurePOS 100 Embedded	AnyPlace POS Hub - USB	Toshiba Cash Drawer USB/ TCx 700
UnifiedPOSVersions	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
DeviceCategory	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ManufacturerName	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ModelName	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
SerialNumber	Y	Y	N	N	N	N	N	N	Y	Y
ManufactureDate	N	N	N	N	N	N	N	N	N	N
Mechanical Revision	N	N	N	N	N	N	N	N	N	N
FirmwareRevision	Y	Y	N	N	N	N	N	N	Y	Y
Interface	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
InstallationDate	N	N	N	N	N	N	N	N	N	N
HoursPowered Count	N	N	N	N	N	N	N	N	N	N
Communication ErrorCount	N	N	N	N	N	N	N	N	N	N

Table 282. Cash drawer specific properties

Specific Properties	SurePOS 730/750 USB	POS Printer 4610 EIA232/ RS485/ USB Network	SurePOS 300 Embedded	SurePOS 500/ 600/ Kiosk EIA232	SurePOS 72x/ 74x/ 78x Embedded	SurePOS 4694 - RS485	SurePOS 4674 Embedded	SureOne/ SurePOS 100 Embedded	Any-Place POS Hub - USB	Toshiba Cash Drawer USB/ TCx 700
DrawerGoodOpen Count	N	N	N	N	N	N	N	N	N	N
DrawerFailedOpen Count	N	N	N	N	N	N	N	N	N	Y (1)
IBM_DrawerOpen CloseCycleCount	N	N	N	N	N	N	N	N	N	Y (2)

Note: 730/750 USB devices use BCD level for Firmware Revision.

Note:

1. The DrawerFailedOpenCount will be reported in 10s (1 = 10), only after 3 concurrent open failures the device will count that as 1 open failure.
It could be reported two extra errors values:
 - a. FF: indicates an error condition into the device.
 - b. FE: indicates a counter overflow condition.
2. The Open/Close cycle count increments in 1 after 100 cycles have been completed (open then close).
It could be reported two extra errors values:
 - a. FFFF: indicates an error condition into the device.
 - b. FFFE: indicates a counter overflow condition.

Check scanner

Check Scanner is only available on Toshiba 4610 POSPrinter models T18 and T19.

Table 283. Check scanner

Device	Bus	Model name value
POSPrinter 4610-TI8/TI9	USB / EIA232 / RS485	4610

Table 284. Check scanner common properties

Common Properties	Description	POSPrinter 4610 -TI8/TI9 - EIA232/RS485/USB Supported:
UnifiedPOSVersion	Version of UnifiedPOS specification supported	Y
DeviceCategory	Device Category	Y
ManufacturerName	Device manufacturer's name	Y
ModelName	Device model name	Y
SerialNumber	Device serial number	Y

Table 284. Check scanner common properties (continued)

Common Properties	Description	POSPrinter 4610 -TI8/TI9 - EIA232/RS485/USB Supported:
ManufactureDate (WWYY format)	Device manufacture date	Y
MechanicalRevision	Device hardware revision	N
FirmwareRevision	Device firmware revision	Y
Interface	Device hardware interface	Y
InstallationDate	Device installation date	N
HoursPoweredOn	Number of hours powered On	N
CommunicationErrorCount	Number of communication errors	N

Table 285. Check scanner manufacturer properties

Manufacturer properties	Unit of measure	Description	POSPrinter 4610 -TI8/TI9 - EIA232/RS485/USB supported:
IBM_CheckScannedCount	"" - empty	Number of checks scanned	Y
IBM_ChecksFailedQualityCount	"" - empty	Number of scanned checks failed to meet internal image quality test	Y
IBM_CheckScannerBrightnessQuality	"" - empty	Check Scanner Brightness Quality	Y
IBM_CheckScannerContrastQuality	"" - empty	Check Scanner Contrast Quality	Y
IBM_CheckScannerFocusQuality	"" - empty	Check Scanner Focus Quality	Y

Note:

1. CheckScanner microcode level used for firmware revision
2. ManufactureDate is only available for new POSPrinters (after December, 2005). Older ones are not supported.
3. The CheckScanner should be cleaned or recalibrated if the following values are not met:
 - IBM_CheckScannerBrightnessQuality: 90 decimal
 - IBM_CheckScannerContrastQuality: 90 decimal
 - IBM_CheckScannerFocusQuality: 77 decimal
4. For 4689 USB and RS485 models, PaperCutCount increments its value by 10 after 10 cuts, not one by one.

Fiscal printer

Table 286. Fiscal printer

Device	Bus	Model name value
4610-GX3	EIA232/USB/RS485	Gx3
4610-GX4	EIA232/USB/RS485	Gx4
4610-GX5	EIA232/USB/RS485	Gx5

Table 286. Fiscal printer (continued)

Device	Bus	Model name value
4610-KX3	EIA232/USB/RS485	Kx3
4610-KX4	EIA232/USB/RS485	Kx4
4610-KX5	EIA232/USB/RS485	Kx5
4610-SJ6	EIA232/USB	SJ6
4610-3XX	RS485	3xx
4610-1NF	EIA232/USB	1NF
4679-3FA 4679-3BS 4679-3BM	RS485	3xx
4610-2CF 4610-2NF	EIA232/USB	2CF 2NF

Table 287. Fiscal printer common properties

Common Properties	USB	EIA232	RS485
UnifiedPOSVersion	Y	Y	Y
DeviceCategory	Y	Y	Y
ManufacturerName	Y	Y	Y
ModelName	Y	Y	Y
SerialNumber	Y	N	N
ManufactureDate	N	N	N
MechanicalRevision	N	N	N
FirmwareRevision	Y	N	N
Interface	Y	Y	Y
InstallationDate	N	N	N
HoursPoweredOn	N	N	N
CommunicationErrorCount	N	N	N

Table 288. Fiscal printer specific properties

Specific Properties	USB	EIA232	RS485
BarcodePrintedCount	N	N	N
FormInsertionCount	N	N	N
HomeErrorCount	N	N	N
JournalCharacterPrinted Count	N	N	N
JournalLinePrintedCount	N	N	N
MaximumTempReached Count	N	N	N
NVRAMWriteCount	N	N	N
PaperCutCount	N	N	N
FailedPaperCutCount	N	N	N
PrinterFaultCount	N	N	N
PrintSideChangeCount	N	N	N

Table 288. Fiscal printer specific properties (continued)

Specific Properties	USB	EIA232	RS485
FailedPrintSideChangeCount	N	N	N
ReceiptCharacterPrintedCount	N	N	N
ReceiptCoverOpenCount	N	N	N
ReceiptLineFeedCount	N	N	N
ReceiptLinePrintedCount	N	N	N
SlipCharacterPrintedCount	N	N	N
SlipCoverOpenCount	N	N	N
SlipLineFeedCount	N	N	N
SlipLinePrintedCount	N	N	N
Stamp Fired Count	N	N	N

Note: USB devices use BCD level for firmware revision.

Hard totals

Table 289. Hard Totals

Device	Bus	Model name value
NVRAM 730/750	USB	4800-73x/75x
NVRAM 72X/74X/78X	Embedded	4800-72x/74x/78x
4694	Embedded	4694
4674	Embedded	4674
SurePOS 300 4810-35x	Embedded	4810-35x
TCx 300 Series	Embedded	4810-36x/37x
TCx 700 Series	Embedded	4900-74x/78x

Table 290. Hard totals common properties

Common properties	Description	Support USB	Support embedded
UnifiedPOSVersion	Version of the UnifiedPOS specification supported	Y	Y
DeviceCategory	Device category	Y	Y
ManufacturerName	Device manufacturer's name	Y	Y
ModelName	Device model name	Y	Y
SerialNumber	Device serial number	Y	N
ManufactureDate	Device manufacture date	N	N
MechanicalRevision	Device hardware revision	N	N
FirmwareRevision	Device firmware revision	Y	N
Interface	Device hardware interface	Y	Y
InstallationDate	Device installation date	N	N

Table 290. Hard totals common properties (continued)

Common properties	Description	Support USB	Support embedded
HoursPoweredCount	Number of hours powered On	N	N
CommunicationErrorCount	Number of communication errors	N	N

Note: USB devices use BCD level for firmware revision.

Keylock

Table 291. Keylock

Device	Bus	Model name value
NANPOS Keylock	USB/RS485	NANPOS
NANPOS Keylock as System attached	USB/PS2	NANPOS
4820 Keylock (SurePoint)	USB/RS485/PS2	4820
50-key Keylock SBCS/DBCS	USB/RS485	50-key
ANKPOS - DBCS	USB/RS485	ANKPOS
ANKPOS - DBCS as System Keyboard	USB/PS2	ANKPOS
133-key or Matrix - SBCS	USB/RS485	133-key
Keyboard 4685 K02 Ultra VI	USB/RS485	4685
Keyboard 4685 K02 Ultra VI with MSR/E	RS485	4685
Keyboard V	USB/RS485	Kbd-V
SurePOS 4674 Keylock	RS485	4674
SureOne 4614 / 4615 Keylock SurePOS 100 – Keylock	Embedded	461x
Keyboard 4685-K03	USB	4685-K03
Keyboard 4685-K03	RS-485	4674/4685-K03
ANPOS Keylock	PS/2, USB	Modular ANPOS
Modular CANPOS Keylock	PS/2, USB	Modular CANPOS
Modular 67 Key Keylock	PS/2, USB	Modular 67 Key
Modular 67 Key Keyboard with Display Keylock	USB	Modular 67 Key

Table 292. Keylock common properties

Common properties	Description	USB supported	RS485 supported	PS2 supported	Embedded
UnifiedPOSVersions	Version of the UnifiedPOS specification supported	Y	Y	Y	Y
DeviceCategory	Device category	Y	Y	Y	Y
ManufacturerName	Device manufacturer's name	Y	Y	Y	Y
ModelName	Device model name	Y	Y	Y	Y

Table 292. Keylock common properties (continued)

Common properties	Description	USB supported	RS485 supported	PS2 supported	Embedded
SerialNumber	Device serial number	Y	N	N	N
ManufactureDate	Device manufacture date. In WWYY format.	Y ¹	N	N	N
MechanicalRevision	Device hardware revision	N	N	N	N
FirmwareRevision	Device firmware revision	Y	N	N	N
Interface	Device hardware interface	Y	Y	Y	Y
InstallationDate	Device installation date	N	N	N	N
HoursPoweredCount	Number of hours powered On	Y ¹	N	N	N
CommunicationErrorCount	Number of communication errors	N	N	N	N

¹ Supported by USB Modular ANPOS, Modular CANPOS, Modular 67 Key Keyboards and SurePoint EC Level >= 400

Table 293. Keylock manufacturer properties table

Manufacturer properties	Description	USB	RS485	PS2	Embedded
IBM_PowerCycleCount	Number of power on/off cycles	Y ¹	N	N	N
IBM_UnexpectedResetsCount	Number of unexpected resets	Y ¹	N	N	N

¹ Supported by USB Modular ANPOS, Modular CANPOS, Modular 67 Key Keyboards and SurePoint EC Level >= 400

Table 294. Keylock specific properties

Specific properties	Description	USB supported	RS485 supported	PS2 supported	Embedded
LockPositionChangeCount	Number of lock position changes	Y ¹	N	N	N

¹ Supported by USB Modular ANPOS, Modular CANPOS, Modular 67 Key Keyboards and SurePoint EC Level >= 400

Note: USB devices use BCD level for firmware revision.

Line display

Table 295. Line display

Device	Bus	Model name value
2X20 VFD 1-sided	USB/EIA232/RS485	2X20 VFD
2X20 VFD 2-sided	USB/RS485	2X20 VFD - A or 2X20 VFD - B
2X20 LCD	USB/RS485	2X20 LCD
APA Display	USB/EIA232/RS485	APA

Table 295. Line display (continued)

Device	Bus	Model name value
50-key Keyboard Display LCD	USB/RS485	2X20 LCD on 50-key
PLU Keyboard III with APA Display	USB/RS485	APA
67-key Keyboard Display LCD	USB	2X20 LCD on 67-key
TCxWave 6140 Integrated 2X20 LCD	USB	2X20 LCD
TCx 2x20 Single-sided LCD	USB	2X20 LCD
TCx 2x20 Double-sided LCD	USB	2X20 LCD - A or 2X20 LCD - B
TCx APA Single-sided	USB	APA

Table 296. Line Display common properties

Common properties	Description	USB supported	EIA232 supported	RS485 supported
UnifiedPOSVersion	Version of the UnifiedPOS specification supported	Y	Y	Y
DeviceCategory	Device category	Y	Y	Y
ManufacturerName	Device manufacturer's name	Y	Y	Y
ModelName	Device model name	Y	Y	Y
SerialNumber	Device serial number	Y	N	N
ManufactureDate	Device manufacture date	Y ¹	N	N
MechanicalRevision	Device hardware revision	N	N	N
FirmwareRevision	Device firmware revision	Y	N	N
Interface	Device hardware interface	Y	Y	Y
InstallationDate	Device installation date	N	N	N
HoursPoweredCount	Number of hours powered On	N	N	N
CommunicationErrorCount	Number of communication errors	N	N	N

¹Supported by TCxWave 6140

Table 297. Line Display specific properties

Specific properties	Description	USB supported	EIA232 supported	RS485 supported
OnlineTransitionCount	Number of online transitions (on after screen blanking)	N	N	N

Note: USB devices use BCD level for firmware revision.

Table 298. Line Display manufacturer properties

Manufacturer properties	Description	USB supported	EIA232 supported	RS485 supported
IBM_PowerCycleCount	Number of power on/off cycles	Y ¹	N	N
IBM_UnexpectedResetsCount	Number of unexpected resets	Y ¹	N	N

¹Supported by TCxWave 6140

MICR

Table 299. MICR

Device	Bus	Model name value
POSPrinter 4610-TI4/5/8/9/2CR	EIA232/RS485/USB/Network	4610

Table 300. MICR common properties

Common properties	Description	EIA232/RS485/USB supported
UnifiedPOSVersion	Version of the UnifiedPOS specification supported	Y
DeviceCategory	Device category	Y
ManufacturerName	Device manufacturer's name	Y
ModelName	Device model name	Y
SerialNumber	Device serial number	Y
ManufactureDate (WWYY format)	Device manufacture date	Y
MechanicalRevision	Device hardware revision	N
FirmwareRevision	Device firmware revision	Y
Interface	Device hardware interface	Y
InstallationDate	Device installation date	N
HoursPoweredCount	Number of hours powered On	N
CommunicationErrorCount	Number of communication errors	N

Table 301. MICR specific properties

Specific properties	Description	EIA232/RS485/USB supported
GoodReadCount	Number of successful reads	Y
FailedReadCount	Number of failed reads	Y
FailedDateParseCount	Number of failed data parses	N

Table 302. MICR manufacturer properties

Manufacturer Properties	Description	EIA232/RS485/USB supported
IBM_CurrentMICRNoiseValue	Value of the MICR Noise on the last read operation	Y (Only 2CR model)

Note:

1. POSPrinter microcode level used for Firmware Revision.
2. ManufacturerDate is only available for newer POSPrinters (after December 2005). Older ones are not supported.
3. DeviceCategory value is taken from the programmatic name.

Motion sensor

Table 303. Motion sensor

Device	Bus	Model name value
SurePOS 500/600/AnyPlace Kiosk	Embedded	Toshiba Generic Motion Sensor

Table 304. Motion sensor common properties

Common Properties	Description	Embedded Supported
UnifiedPOSVersion	Version of the UnifiedPOS specification supported	Y
DeviceCategory	Device category	Y
ManufacturerName	Device manufacturer's name	Y
ModelName	Device model name	Y
SerialNumber	Device serial number	N
ManufactureDate	Device manufacture date	N
MechanicalRevision	Device hardware revision	N
FirmwareRevision	Device firmware revision	N
Interface	Device hardware interface	Y
InstallationDate	Device installation date	N
HoursPoweredCount	Number of hours powerd On	N
CommunicationErrorCount	Number of communication errors	N

Table 305. Motion Sensor specific properties

Specific properties	Description	Supported
MotionEventCount	Number of motion occurrences	N

MSR

Table 306. MSR

Device	Bus	Model name value
ANPOS ISO MSR	USB	NANPOS - ISO
ANPOS as System attached ISO MSR	PS2/USB	NANPOS - ISO
4820 ISO MSR (SurePoint)	USB	4820 - ISO
4820 JUCC MSR (SurePoint)	USB	4820 - JUCC
4820/AnyPlace Kiosk/SurePOS 500/600 ISO MSR	EIA232	4820/500/600/AnyPlace Kiosk – ISO
4820/AnyPlace Kiosk/SurePOS 500/600 JUCC MSR	EIA232	4820/500/600/AnyPlace Kiosk – JUCC
50-key ISO MSR	USB	50-key - ISO
CANPOS ISO MSR	PS2	CANPOS - ISO
ANKPOS JUCC MSR	USB	ANKPOS - JUCC
ANKPOS as System keyboard JUCC MSR	PS2/USB	ANKPOS - JUCC

Table 306. MSR (continued)

Device	Bus	Model name value
133-key or Matrix - ISO MSR	USB	133-key ISO
Keyboard 4685 K02 Ultra VII - JUCC MSR 4 pos	RS485/USB	4685/Kbd V JUCC
Keyboard 4685 K02 Ultra VII - JUCC MSR/E 4 pos	RS485	4685 JUCC with Encoder
Keyboard 4685 K02 Ultra VII - JUCC MSR/E 6 pos	RS485	4685 JUCC with Encoder
Keyboard V JUCC MSR	RS485/USB	4685/Kbd V - JUCC
133-key/4820/50-key/NANPOS ISO MSR	RS485	Toshiba MSR - ISO
SurePOS 4674/4820/50-key/ANKPOS JUCC MSR	RS485	Toshiba MSR - JUCC
SureOne 4614/4615 ISO MSR SurePOS 100 ISO MSR	PS2	461x - ISO
50-key JUCC MSR	USB	50-key - JUCC
Keyboard 4685 K03 JUCC MSR	USB	4685-K03 - JUCC
Keyboard 4685 K03 JUCC MSR	RS485	4685 / Kbd V - JUCC
Modular ANPOS with ISO MSR	PS/2, USB	Modular ANPOS-ISO
Modular ANPOS with JUCC MSR	PS/2, USB	Modular ANPOS-JUCC
Modular CANPOS with ISO MSR	PS/2, USB	Modular CANPOS-ISO
Modular CANPOS with JUCC MSR	PS/2, USB	Modular CANPOS-JUCC
Modular 67 Key with ISO MSR	PS/2, USB	Modular 67 Key-ISO
Modular 67 Key with JUCC MSR	PS/2, USB	Modular 67 Key-JUCC
Modular 67-Key/Display ISO MSR	USB	Modular 67 Key-ISO
TCxWave 6140 Integrated MSR	USB	TCxWave 6140-ISO

Table 307. MSR common properties

Common properties	Description	USB supported	PS2 supported	EIA232 supported	RS485 supported
UnifiedPOSVersion	Version of the UnifiedPOS specification supported	Y	Y	Y	Y
DeviceCategory	Device category	Y	Y	Y	Y
ManufacturerName	Device manufacturer's name	Y	Y	Y	Y
ModelName	Device model name	Y	Y	Y	Y
SerialNumber	Device serial number	Y	N	N	N
ManufactureDate	Device manufacture date. In WWYY format.	Y ¹	N	N	N
MechanicalRevision	Device hardware revision	N	N	N	N
FirmwareRevision	Device firmware revision	Y	N	N	N
Interface	Device hardware interface	Y	Y	Y	Y
InstallationDate	Device installation date	N	N	N	N

Table 307. MSR common properties (continued)

Common properties	Description	USB supported	PS2 supported	EIA232 supported	RS485 supported
HoursPoweredCount	Number of hours powered On	Y ¹	N	N	N
CommunicationErrorCount	Number of communication errors	N	N	N	N

¹ Supported by USB Modular ANPOS, Modular CANPOS, Modular 67 Key Keyboards and SurePoint EC Level >= 400, TCxWave 6140

Table 308. MSR manufacturer properties table

Manufacturer properties	Description	USB	RS485	EIA232
IBM_PowerCycleCount	Number of power on/off cycles	Y ¹	N	N
IBM_UncorrectedErrorsCount	Number of unexpected resets	Y ¹	N	N

¹ Supported by USB Modular ANPOS, Modular CANPOS, Modular 67 Key Keyboards and SurePoint EC Level >= 400, TCxWave 6140

Table 309. MSR specific properties

Specific properties	Description	USB supported	PS2 supported	EIA232 supported	RS485 supported
GoodReadCount	Number of successful reads	Y ¹	N	N	N
FailedReadCount	Number of failed reads	Y ¹	N	N	N
UnreadableCardCount	Number of failed data parses	Y ¹	N	N	N
MissingStartSentinelTrack1Count	Number of Reads that Miss Start Sentinel	Y ¹	N	N	N
ParityLRCErrorTrack1Count	Number of LRC Error	Y ¹	N	N	N
MissingStartSentinelTrack2Count	Number of Reads that Miss Start Sentinel	Y ¹	N	N	N
ParityLRCErrorTrack2Count	Number of LRC Error	Y ¹	N	N	N
MissingStartSentinelTrack3Count	Number of Reads that Miss Start Sentinel	Y ¹	N	N	N
ParityLRCErrorTrack3Count	Number of LRC Error	Y ¹	N	N	N

Table 309. MSR specific properties (continued)

Specific properties	Description	USB supported	PS2 supported	EIA232 supported	RS485 supported
MissingStartSentinelTrack4Count	Number of Reads that Miss Start Stentinel	Y ¹	N	N	N
ParityLRCErrorTrack4Count	Number of LRC Error	Y ¹	N	N	N

¹ Supported by USB Modular ANPOS, Modular CANPOS, Modular 67 Key Keyboards and SurePoint EC Level >= 400

Note: USB devices use BCD level for firmware revision.

POS keyboard

Table 310. POS keyboard

Device	Bus	Model name value
NANPOS POSKeyboard - SBCS	RS485/USB	NANPOS
NANPOS POSKeyboard - System attached	PS2/USB	NANPOS
4820 POSKeyboard (SurePoint-keypad)	PS2/RS485/USB	Keypad
50-key POSKeyboard - SBCS	RS485/USB	50-key
CANPOS POSKeyboard - SBCS	PS2	CANPOS
ANKPOS - DBCS	RS485/USB	ANKPOS
ANKPOS - DBCS System Keyboard	PS2/USB	ANKPOS
133-key or Matrix - SBCS	RS485/USB	133-key
PLU Keyboard III with Display	RS485/USB	PLU
Keyboard 4685 K02 Ultra VII - 4 pos Keylock	RS485/USB	4685
Keyboard 4685 K02 Ultra VII - MSR/E 4 pos Keylock	RS485	4685
Keyboard 4685 K02 Ultra VII - MSR/E 6 pos Keylock	RS485	4685
Keyboard V	RS485/USB	Kbd V
SurePOS 4674 POSKeyboard	RS485	4674/4685-K03
SureOne 4614/4615 Keyboard SurePOS 100 – Keyboard	PS2	461x
50-key POSKeyboards JUCC MSR - DBCS	RS485/USB	50-key
Keyboard 4685 K03	USB	4685-K03
Keyboard 4685 K03	RS485	4674/4685-K03
Modular ANPOS Keyboard	PS/2, USB	Modular ANPOS
Modular CANPOS Keyboard	PS/2, USB	Modular CANPOS
Modular 67 Key Keyboard	USB	Modular 67 Key

Table 310. POS keyboard (continued)

Device	Bus	Model name value
Modular 67 Key Keyboard with Display	USB	Modular 67 Key

Table 311. POS Keyboard common properties

Common properties	Description	USB supported	PS2 supported	RS485 supported
UnifiedPOSVersion	Version of the UnifiedPOS specification supported	Y	Y	Y
DeviceCategory	Device category	Y	Y	Y
ManufacturerName	Device manufacturer's name	Y	Y	Y
ModelName	Device model name	Y	Y	Y
SerialNumber	Device serial number	Y	N	N
ManufactureDate	Device manufacture date. In WWYY format	Y ¹	N	N
MechanicalRevision	Device hardware revision	N	N	N
FirmwareRevision	Device firmware revision	Y	N	N
Interface	Device hardware interface	Y	Y	Y
InstallationDate	Device installation date	N	N	N
HoursPoweredCount	Number of hours powered On	Y ¹	N	N
CommunicationErrorCount	Number of communication errors	N	N	N

¹ Supported by USB Modular ANPOS, Modular CANPOS, Modular 67 Key Keyboards and SurePoint EC Level >= 400

Table 312. POS keyboard manufacturer properties table

Manufacturer properties	Description	USB	PS2	RS485
IBM_PowerCycleCount	Number of power on/off cycles	Y ¹	N	N
IBM_UnexpectedResetsCount	Number of unexpected resets	Y ¹	N	N
IBM_ConfigurationVersion	Device configuration version	Y ¹	N	N

¹ Supported by USB Modular ANPOS, Modular CANPOS, Modular 67 Key Keyboards and SurePoint EC Level >= 400

Table 313. POS Keyboard specific properties

Specific properties	Description	USB supported	PS2 supported	RS485 supported
KeyPressedCount	Number of keys pressed	Y ¹	N	N

¹ Supported by USB Modular ANPOS, Modular CANPOS, Modular 67 Key Keyboards and SurePoint EC Level >= 400

Note: USB devices use BCD level for firmware revision.

POS printer

Table 314. POS printer

Device	Bus	Model name value
4610 TI3 (thermal/impact)	EIA232/RS485/USB	4610-TI3/4
Device: 4610 2NR (thermal/impact) - SBCS	EIA232/RS485/USB	4610-2NR-SBCS
4610 2CR (2NR+flipper/MICR) - SBCS	EIA232/RS485/USB	4610-2CR-SBCS
4610 2NR (Thermal/impact) - DBCS	EIA232/RS485/USB	4610-2NR-DBCS
4610 2CR (2NR+flipper/MICR) - DBCS	EIA232/RS485/USB	4610-2CR-DBCS
4610 TI4 (TI3+flipper/MICR)	EIA232/RS485/USB	4610-TI3/4
4610 TI5 (DBCD TI3)	EIA232/RS485/USB	4610-TI5
4610 TI8	EIA232/RS485/USB	4610-TI8
4610 TI9	EIA232/RS485/USB	4610-TI9
4610 TI1/2 (thermal/impact)	EIA232/RS485	4610-TI1/2
4610 Tx6 (SS thermal)	EIA232/RS485/USB	4610-Tx6
4610 Tx7 (SS thermal DBCS)	EIA232/RS485/USB	4610-Tx7
4689 TI5 all except 001 and 002 (DBCS)	RS485/USB	4689
4674 Embedded Printer - same as 4689	RS485	4674
SureOne 4614/4615 Thermal SurePOS 100 – Thermal	EIA232	461x-Thermal
SureOne 4613/4614/4615 Impact SurePOS 100 – Impact	EIA232	461x-Impact

Table 315. POS printer common properties

Common properties	Description	4610 Printer USB/EIA232/ RS485 supported	4674/4689 RS485	4689 USB	SureONE Impact /Thermal
UnifiedPOSVersion	Version of the UnifiedPOS specification supported	Y	Y	Y	Y
DeviceCategory	Device category	Y	Y	Y	Y
ManufacturerName	Device manufacturer's name	Y	Y	Y	Y
ModelName	Device model name	Y	Y	Y	Y
SerialNumber	Device serial number	Y	N	Y	N
ManufactureDate (WWYY format)	Device manufacture date	Y	N	N	N
MechanicalRevision	Device hardware revision	N	N	N	N
FirmwareRevision	Device firmware revision	Y	Y	Y	N
Interface	Device hardware interface	Y	Y	Y	Y
InstallationDate	Device installation date	N	N	N	N

Table 315. POS printer common properties (continued)

Common properties	Description	4610 Printer USB/EIA232/ RS485 supported	4674/4689 RS485	4689 USB	SureONE Impact /Thermal
HoursPoweredCount	Number of hours powered On	N	N	N	N
CommunicationErrorCount	Number of communication errors	N	N	N	N

Table 316. POS printer specific properties

Specific properties	Description	4610 Printer USB/EIA232/ RS485 supported	4674/4689 RS485	4689 USB	SureONE Thermal
BarCodePrintedCount	Number of barcodes printed	Y	N	N	N
FormInsertionCount	Number of forms inserted into the document/slip station	Y	N	N	N
HomeErrorCount	Number of home errors	Y	N	N	N
JournalCharacterPrintedCount	Number of journal characters printed	N	N	N	N
JournalLinePrintedCount	Number of journal lines printed	N	N	Y	N
MaximumTempReachedCount	Number of times maximum temperature reached	Y	N	N	N
NVRAMWriteCount	Number or times NVRAM is written to	Y	N	N	N
PaperCutCount	Number of paper cuts	Y	Y	Y	N
FailedPaperCutCount	Number of failed paper cuts	Y (See note 6.)	N	N	N
PrinterFaultCount	Number of printer faults	N	N	N	N
PrintSideChangeCount	Number of print side changes (or check flips) performed	Y (See note 5.)	N	N	N
FailedPrintSideChangeCount	Number of print side change (of check flip) failures	Y (See note 5.)	N	N	N
ReceiptCharacterPrintedCount	Number of receipt characters printed	Y	N	N	N
ReceiptCoverOpenCount	Number of times receipt cover was opened	Y	N	N	N
ReceiptLineFeedCount	Number of receipt line feeds performed	Y	N	N	N
ReceiptLinePrintedCount	Number of receipt lines printed	N	Y	Y	N
SlipCharacterPrintedCount	Number of document/slip characters printed	Y	N	N	N

Table 316. POS printer specific properties (continued)

Specific properties	Description	4610 Printer USB/EIA232/ RS485 supported	4674/4689 RS485	4689 USB	SureONE Thermal
SlipCoverOpenCount	Number of times the document/slip station cover opened	Y	N	N	N
SlipLineFeedCount	Number of document/slip line feeds performed	Y	N	N	N
SlipLinePrintedCount	Number of document/slip lines printed	N	N	N	N
Stamp Fired Count	Number of stamps fired	N	N	N	N

Note:

1. POSPrinter microcode level used for firmware revision.
2. ManufacturerDate is only available for newer POSPrinters (after December 2005). Older ones are not supported.
3. For 4689 USB and RS485 models, PaperCutCount increments by 10 after 10 cuts.
4. In some cases a tag <Value/> may be retrieved, indicating that the requested statistic is not supported for the printer firmware, or it has never been initialized.
5. Only valid on 4610-TI4/8/9/2xR.
6. Only valid on Tx6/Tx7/2xR/1NR models.

Table 317. POS printer manufacturer properties

Manufacturer Properties		RS232/RS485/USB
Name	Description	Supported
IBM_ReceiptPaperRemaining	Number of mm of paper left in the printer	Yes, see note ²
IBM_UnexpectedSlipCoverOpenCount	Number of times the Slip cover was opened without an error condition	Yes, see note ¹
IBM_SlipFeedErrorCount	Number of Slip Feed errors	Yes, see note ¹
IBM_ReceiptPaperJamCount	Number of Receipt Paper Jams	Yes, see note ²
IBM_FlashWriteFailedCount	Number of failed flash writes	Yes, see note ²
IBM_UnexpectedRecCoverOpenCount	Number of times the Receipt cover was opened without an error condition	Yes, see note ²
IBM_ImpactHeadFailedCount	Number of Impact head failed coils	Yes, see note ¹
IBM_ThermalPrintHeadElementFailure	Number of the Thermal Head element that is failing (Can be more than 1)	Yes, see note ²

¹Only valid on 4610-2xR models

²Only valid on 4610-2xR/1NR models.

Scale

Table 318. Scale

Device	Bus	Model name value	Manufacturer
OEM PSC Magellan Scanner/Scale Model 384	USB	OEM - Scale	OEM
OEM PSC Magellan Scanner/Scale Model 8201	USB	OEM - Scale	OEM
Toshiba 469x Model 1	RS485	469x	OEM
Toshiba 469x Model 2	RS485	469x	OEM

Table 319. Scale common properties

Common properties	Description	USB supported	RS485 supported
UnifiedPOSVersion	Version of the UnifiedPOS specification supported	Y	Y
DeviceCategory	Device category	Y	Y
ManufacturerName	Device manufacturer's name	Y	Y
ModelName	Device model name	Y	Y
SerialNumber	Device serial number	Y	N
ManufactureDate	Device manufacture date	N	N
MechanicalRevision	Device hardware revision	N	N
FirmwareRevision	Device firmware revision	Y	N
Interface	Device hardware interface	Y	Y
InstallationDate	Device installation date	N	N
HoursPoweredCount	Number of hours powered On	N	N
CommunicationErrorCount	Number of communication errors	N	N

Table 320. Scale specific properties

Specific properties	Description	USB supported	RS485 supported
GoodWeightReadCount	Number of successful weight reads	N	N

Note: USB devices use BCD level for Firmware Revision.

Scanner

Table 321. Scanner

Device	Bus	Model name value	Manufacturer
OEM PSC Magellan Scanner/Scale - Model 384	USB	OEM-Scanner	OEM
OEM PSC Magellan Scanner/Scale - Model 8201	USB	OEM-Scanner	OEM
OEM Symbol Hand Held - Model LS4804	USB	OEM-Scanner	OEM
OEM Symbol Hand Held - Model LS2104	USB	OEM-Scanner	OEM

Table 321. Scanner (continued)

Device	Bus	Model name value	Manufacturer
OEM Symbol Hand Held - Model M2004	RS485	OEM-Scanner	OEM
Toshiba Hand Held BCR Model 1 (4500)	RS485	4500	OEM
Toshiba Hand Held BCR Model 2 (4501)	RS485	4501	OEM
Toshiba 1520 Hand Held Model A02	RS485	1520	OEM
Toshiba 4685 Hand Held Model 001/K001	RS485	4685	OEM
Toshiba 4696 Scanner/Scale Model 1	RS485	4696	OEM
Toshiba 4697 Model 1	RS485	4697	OEM
Toshiba 4698 Model 1/2	RS485	4698	OEM
AnyPlace Kiosk - 4838 Line	EIA232	Line	OEM
AnyPlace Kiosk - Omni	EIA232	Omni	OEM

Note: When RS485 Scanners do not have a JposEntry configuration properties the “OEM-Scanner” Model Name value will be in use (Except 1520 model).

Table 322. Scanner common properties

Common properties	Description	USB supported	RS485 supported	EIA232
UnifiedPOSVersion	Version of the UnifiedPOS specification supported	Y	Y	Y
DeviceCategory	Device category	Y	Y	Y
ManufacturerName	Device manufacturer's name	Y	Y	Y
ModelName	Device model name	Y	Y	Y
SerialNumber	Device serial number	Y	N	N
ManufactureDate	Device manufacture date	N	N	N
MechanicalRevision	Device hardware revision	N	N	N
FirmwareRevision	Device firmware revision	Y	N	N
Interface	Device hardware interface	Y	Y	Y
InstallationDate	Device installation date	N	N	N
HoursPoweredCount	Number of hours powered On	N	N	N
CommunicationErrorCount	Number of communication errors	N	N	N

Table 323. Scanner specific properties

Specific properties	Description	USB supported	RS485 supported	EIA232
GoodScanCount	Number of successful scans	N	N	N

Note:

1. USB devices use BCD level for firmware revision.
2. DeviceCategory value is taken from programmatic name.

Tone indicator

Table 324. Tone indicator

Device	Bus	Model name value
NANPOS Tone Indicator - SBCS	RS485/USB	NANPOS
NANPOS Tone Indicator as System Keyboard	PS2/USB	NANPOS
4820 Tone Indicator (SurePoint)	PS2/RS485/USB	4820
50-key Tone Indicator - SBCS	RS485/USB	50-key
ANKPOS - DBCS	RS485/USB	ANKPOS
ANKPOS - DBCS as System Keyboard	PS2/USB	ANKPOS
133-key or Matrix - SBCS	RS485/USB	133-key
PLU Keyboard III with Display	RS485/USB	PLU
Keyboard 4685 K02 Ultra VII - 4 pos Keylock	RS485/USB	4685
Keyboard 4685 K02 Ultra VII - MSR/E 4 pos	USB	4685
Keyboard 4685 K02 Ultra VII - MSR/E 6 pos	USB	4685
Keyboard V	RS485/USB	Kbd V
SurePOS 4674 Tone Indicator	RS485	4674/4685-K03
50-key POSKeyboard JUCC MSR - DBCS	USB	50-key
4610 Tx6/Tx7 Printer ToneIndicator	EIA232/RS485/USB	4610-Tx6/Tx7
Keyboard 4685 K03	USB	4685-K03
Keyboard 4685 K03	RS485	4674/4685-K03
Modular ANPOS Keyboard	PS/2, USB	Modular ANPOS
Modular CANPOS Keyboard	PS/2, USB	Modular CANPOS
Modular 67 Key Keyboard	PS/2, USB	Modular 67 Key
Modular 67 Key Keyboard with Display	USB	Modular 67 Key
6141 Tone Indicator	USB	6141

Table 325. Tone indicator common properties

Common properties	Description	4610 Printer built-in tone indicator	USB supported	PS2 supported	RS485 supported
UnifiedPOSVersion	Version of the UnifiedPOS specification supported	Y	Y	Y	Y
DeviceCategory	Device category	Y	Y	Y	Y

Table 325. Tone indicator common properties (continued)

Common properties	Description	4610 Printer built-in tone indicator	USB supported	PS2 supported	RS485 supported
ManufacturerName	Device manufacturer's name	Y	Y	Y	Y
ModelName	Device model name	Y	Y	Y	Y
SerialNumber	Device serial number	Y	Y	N	N
ManufactureDate	Device manufacture date. In WWYY format.	Y ¹	N	N	N
MechanicalRevision	Device hardware revision	N	N	N	N
FirmwareRevision	Device firmware revision	Y	Y	N	N
Interface	Device hardware interface	Y	Y	Y	Y
InstallationDate	Device installation date	N	N	N	N
HoursPoweredCount	Number of hours powered On	Y ¹	N	N	N
CommunicationErrorCount	Number of communication errors	N	N	N	N

¹ Supported by USB Modular ANPOS, Modular CANPOS, Modular 67 Key Keyboards and SurePoint EC Level >= 400

Table 326. Tone Indicator manufacturer properties table

Manufacturer properties	Description	4610 Printer built-in tone indicator	USB	PS2	Embedded
IBM_PowerCycleCount	Number of power on/off cycles	N	Y ¹	N	N
IBM_UnexpectedResetsCount	Number of unexpected resets	N	Y ¹	N	N

¹ Supported by USB Modular ANPOS, Modular CANPOS, Modular 67 Key Keyboards and SurePoint EC Level >= 400

Table 327. Tone indicator specific properties

Specific properties	Description	4610 Printer built-in tone indicator	USB supported	PS2 supported	RS485 supported
ToneSoundedCount	Number of tones played	N	Y ¹	N	N

¹ Supported by USB Modular ANPOS, Modular CANPOS, Modular 67 Key Keyboards and SurePoint EC Level >= 400

Note: USB devices use BCD level for firmware revision.

Appendix B. Java Web Start configuration to access the POS control center

Install JavaPOS SDK

Create a KeyStore

To create a store key:

1. Change to webstart directory

For Windows: `cd c:\POS\IBMJPOS\webstart`

Note: There is assumed the product is installed in the default directory. Linux: `cd /opt/ibm/javapos/webstart`

For Linux: `cd /opt/ibm/javapos/webstart`

Note: This and following steps is assumed they are executed with root user.

2. For Windows:

Execute `keytool.exe -genkey -keystore jposKeyStore -alias jposAlias`

For Linux:

Execute `keytool -genkey -keystore jposKeyStore -alias jposAlias`

Expected input:

Enter keystore password: "keyStorePassword"

What is your first and last name?

[Unknown]: "userName"

What is the name of your organizational unit?

[Unknown]: "yourOrganizationalUnit"

What is the name of your organization?

[Unknown]: "yourOrganization"

What is the name of your City or Locality?

[Unknown]: "yourCity"

What is the name of your State or Province?

[Unknown]: "yourState"

What is the two-letter country code for this unit?

[Unknown]: "us"

Is CN=userName, OU=yourOrganizationUnit, O=yourOrganization,
L=yourCity, ST=yourState, C=us correct?
type "yes" or "no") [no]: yes

Enter key password

for <jposAlias>: RETURN if same as keystore password):"aliasPassword"

Create a self-signed certificate

To create a self-signed certificate:

1. Execute this command, `keytool.exe -list -v -keystore jposKeyStore` .
2. Enter keystore password: `keyStorePassword`

Expected input:

```
Keystore type: jks
Keystore provider: IBMJCE
```

```
Your keystore contains 1 entry
```

```
Alias name: jposAlias
Creation date: Apr 9, 2008
Entry type: keyEntry
Certificate chain length: 1
Certificate[1]:
Owner: CN=userName, OU=yourOrganizationUnit,
O=yourOrganization, L=yourCity, ST=yourState, C=us
Issuer: CN=userName, OU= yourOrganizationUnit,
O= yourOrganization, L= yourCity, ST= yourState, C=us
Serial number: 47fcfb8e
Valid from: 4/9/08 2:23 PM until: 7/8/08 2:23 PM
Certificate fingerprints:
MD5: BB:A4:36:45:C4:0D:FD:A0:0D:66:12:4F:D0:B3:D8:EE
SHA1:
BE:D4:D4:7A:50:A6:A3:8B:F9:D6:38:2E:0E:BD:15:D1:9B:83:B7:24
```

Create jpos.xml

Create `jpos.xml` and save in the following directory:

For Windows: `cd c:\POS\IBMJPOS\config`

For Linux: `cd /opt/ibm/javapos/etc`

Create the resource JAR

To create the resource JAR:

1. Change directory
 - Windows: `cd c:\POS\IBMJPOS\config`
 - Linux: `cd /opt/ibm/javapos/etc`
2. Execute the appropriate command:
 - Windows: `jar -cvf c:\POS\IBMJPOS\webstart\ibmjpos_resources.jar *.*`
 - Linux: `jar -cvf /opt/ibm/javapos/webstart/ibmjpos_resources.jar *.*`
3. Change directory (only for Linux)
 - Windows**
No action is necessary
 - Linux** `cd /opt/javax-usb/etc`
4. Execute the appropriate command (Only for Linux)
 - Windows**
No action is necessary
 - Linux** `jar -uf /opt/ibm/javapos/webstart/ibmjpos_resources.jar javax.usb.properties`

Sign the all JARs file with the test certificate

To sign the all JARs file with the test certificate

1. Copy all the jars located in:

Windows: C:\POS\IBMJPOS\lib to C:\POS\IBMJPOS\webstart directory.

Linux: /opt/ibm/javapos/lib and /opt/javax-usb/lib to /opt/ibm/javapos/webstart directory.

Note: It is necessary to copy all the jar files for the JavaPOS device drivers to the webstart directory. Otherwise, once the existing jar files are signed, a Java application that is not launched through the webstart interface will not be able to use the JavaPOS device drivers.

2. Sign all jars located at webstart directory.

For example, to sign the jtux.jar use the following command: **jarsigner.exe -keystore jposKeyStore -storepass "keyStorePassword" -keypass "aliasPassword" ibmjavapos.jar jposAlias**

Run JNLP file

To run the JNLP file

1. Move to:

Windows: cd c:\POS\IBMJPOS\webstart

Linux: cd /opt/ibm/javapos/webstart

2. Run.

For Windows or Linux: **Javaws toshibaJavaPOS.jnlp**.

Note: When UnifiedPOS was installed without "System Management Support", edit toshibaJavaPOS.jnlp and comment out the following line: `<!jar href="jpos_sysmgmt.jar" download="eagerly" />`

Appendix C. Additional information for SUSE Linux Enterprise (SLE)

USB Alphanumeric POS Keyboard does not receive scan codes

Although this problem can be caused by multiple issues, the most common problem is that the /dev/input/event<number> does not have the correct permissions. Follow this procedure to validate that this is the problem:

1. List the USB devices and identify the event associated with your keyboard:
 - a. Run cat /proc/bus/input/devices and search for an entry for your POSKeyboard. It should look similar to the following:

```
I: Bus=0003 Vendor=04b3 Product=4604 Version=0100
N: Name="(c) Copyright IBM Corp. 2008 IBM Retail USB Alphanumeric POS Keyboard
P: Phys=usb-0000:00:1d.0-1.4/input0
S: Sysfs=/devices/pci0000:00/0000:00:1d.0/usb4/4-1/4-1.4/4-1.4:1.0/input/input4
U: Uniq=
H: Handlers=kbd event0
B: EV=120013
B: KEY=10000 7 ff9f207a c14057ff febeffdff fefffff ffffff ffffffe
B: MSC=10
B: LED=1f
```
 - b. The 'event0' is associated with the POS Keyboard.
2. Review the event node and verify that it has the correct permissions. It should have 666:
 - a. Run ls -la /dev/input/event?
3. Manual Fix: Add permissions to the event node.
 - a. Run chmod 666 /dev/input/event0
4. Automatic fix: Add a udev rule to set the permissions when the USB keyboard is hot-plugged.
 - a. Create a file named 83-ibmjavapos.rules at /etc/udev/rules.d
 - b. Add the following line to the file:

```
KERNEL=="event*", NAME=="input/%k", MODE="0666", GROUP="users"
```
 - c. Restart your system

Known GUI issues

On certain systems, such as SurePOS 300 (4810-34x), GUI applications may not display correctly. This is due to conflicts with IBM JVM 1.6 and Desktop Effects, which are enabled on SLE 11 SP2 by default.

To resolve this issue, disable Desktop Effects on SLE 11 SP3:

1. Go to **Computers** → **More Applications** → **Desktop Effects**.
2. Under Tools, uncheck **Enable Desktop Effects**.

Power management

JavaPOS drivers support the Suspend to RAM (S3) option for SUSE Linux Enterprise (SLE), also referred to as Stand-by or Sleep. This section details the ways in which a system should be suspended.

SLE 11 and PM Utils package

For SLE 11 (without service pack and newer), the JavaPOS driver requires the pm-utils package to be installed. The system can be suspended in the following ways:

- Log in as root to X-Windows. Click **Menu** → **Shutdown**, then click **Suspend**.
- Log in as root to X-Windows. Open x terminal and execute `pm-suspend`.

Note: X-Windows may not work on all systems.

Make sure that the system has been updated with the latest video drivers and BIOS version, as described in the POS Linux Operating System Configuration Guide: <http://www-01.ibm.com/support/docview.wss?rs=219&uid=pos1R7000429>

Important: The use of the `s2ram` command to suspend to RAM is not supported. JavaPOS drivers require `pm-suspend`.

Appendix D. JavaPOS device configuration for Linux

Serial ports configuration

Serial port enumeration utility

The Toshiba SurePOS 300, 500, and 700 systems may require the configuration of additional COM ports beyond COM1 and COM2. By default, /dev/ttyS0 (COM1) and /dev/ttyS1 (COM2) are configured by the operating system.

To configure the additional COM ports available on SurePOS 300, 500, and 700 systems, a setserial utility is provided. This utility configures additional COM ports correctly and remaps them starting with /dev/ttyS2 (COM3). The setserial configuration utility is available for download from the following link: <http://www-01.ibm.com/support/docview.wss?rs=220&uid=pos1R1004421>.

Version tested: Toshiba_setserial-1.0.0-3.i586.rpm or higher.

On SurePOS 500 systems, the integrated MSR and Touch devices are mapped to the following COM ports:

- TOUCH device: /dev/ttyS4 (COM 5)
- MSR device: /dev/ttyS5 (COM 6)

This means that the portName for the integrated RS-232 MSR in jpos.xml must be set to COM6, as shown:

```
<prop name="portName" type="String" value="COM6"/>
```

Toshiba systems and COM port assignments reference

Table 328 provides default mapping information for additional COM ports on Toshiba systems.

Table 328. Default mapping of additional COM ports

System/Model	COM Port Label	COM Port Assignment	JavaPOS Device Mapping (jpos.xml)
All POS Systems			
Common	A B	/dev/ttyS0 /dev/ttyS1	COM1 COM2
SurePOS 700			
4800-7x5 4800-7x4 4800-7x3 4800-7x2	C D	/dev/ttyS4 /dev/ttyS5	COM5 COM6
4800-7x1	C D	/dev/ttyS5 /dev/ttyS4	COM6 COM5
4900-7x6	C D	/dev/ttyS2 /dev/ttyS3	COM3 COM4

Table 328. Default mapping of additional COM ports (continued)

System/Model	COM Port Label	COM Port Assignment	JavaPOS Device Mapping (jpos.xml)
4800-7x5 (with EIA232 IO card)	E	/dev/ttyS6	COM7
4800-7x4 (with EIA232 IO card)	F	/dev/ttyS7	COM8
4800-7x3 (with EIA232 IO card)	H	/dev/ttyS3	COM4
4800-7x2 (with EIA232 IO card)			
4800-7x1 (with EIA232 IO card)			
4900-7x6 (with EIA232 IO card)	E	/dev/ttyS4	COM5
	F	/dev/ttyS5	COM6
	H	/dev/ttyS7	COM8
SurePOS 500			
4846-545/565	C	/dev/ttyS5	COM6
	D	/dev/ttyS3	COM4
4851-514	C	/dev/ttyS4	COM5
	D	/dev/ttyS3	COM4
4840-5x3	C	<TBD>	<TBD>
	D	/dev/ttyS5	COM6
SurePOS 300			
4810-34x (with EIA232 IO card)	C	/dev/ttyS7	COM8
4810-35x (with EIA232 IO card)	D	/dev/ttyS4	COM5
4810-36x (with EIA232 IO card)	E	/dev/ttyS5	COM6
4810-37x (with EIA232 IO card)	F	/dev/ttyS6	COM7
4810-33x	C	/dev/ttyS6	COM7
4810-32x	D	/dev/ttyS7	COM8
	E	/dev/ttyS4	COM5
	F	/dev/ttyS5	COM6
4810-34x (with USB IO card)	C	/dev/ttyS7	COM8
AnyPlace Kiosk			
4613-108	See Common	See Common	See Common
4613-118			
SurePOS 100			
4613-108	C	/dev/ttyS2	COM3
4613-118	D	/dev/ttyS3	COM4
	Printer	/dev/ttyS4	COM5

Serial port access

The operating system does not, by default, provide users with access to serial ports. You must provide access to the non-root users. You must also set the raw mode in the serial port so that the device will work properly:

Manual configuration

1. Add permissions to serial ports:
 - Run `chmod 666 /dev/ttyS?`
2. Set RAW mode (for the RS-232 port, execute the following):

- Run `/bin/stty -F /dev/ttyS0 raw`

Automatic configuration using udev a rule

1. Create a file with the name `83-toshibajavapos.rules` at `/etc/udev/rules.d`
2. Add the following lines to the file:


```
KERNEL=="ttyS*", MODE="0666", GROUP="users"
KERNEL=="ttyS*", RUN+="/bin/stty -F /dev/%k raw"
```
3. Restart your system.

TCxWave USB to RS-232 adapter

On TCxWave systems, the USB to RS-232 adapter uses the FTDI drivers to communicate with serial devices. The FTDI drivers are included in the operative system, but JavaPOS drivers require that the `/dev/ttyUSB0` port is mapped to `/dev/ttyS0(COM1)`.

To manually configure this port:

1. Remove the default serial port:
Run `rm /dev/ttyS0(COM1)`
2. Create a symbolic link:
Run `ln -sf /dev/ttyUSB0 /dev/ttyS0`
3. Add permissions to serial ports:
Run `chmod 666 /dev/ttyS?`
4. Set RAW mode:
Run `/bin/stty -F /dev/ttyS0 raw`

Important: This configuration must be done every time the system starts.

Note: The last digit of the mapping on the `ttyUSB` may change depending on the previous USB devices connected to the system.

USB device access

The operating system does not, by default, provide users with access to USB devices. The JavaPOS RPM automatically enables USB device user access on your system.

If the default automatic configuration is unsuccessful, try the steps for manual configuration.

SLE 11 without Service Pack and newer SLE 11 versions

Automatic configuration using udev a rule

1. Create a file with the name `83-ibmjavapos.rules` at `/etc/udev/rules.d`
2. Add the following lines to the file:


```
ENV{SUBSYSTEM}=="input", ENV{DEVNAME}=="input/event*", MODE="0660", GROUP="users"
ENV{SUBSYSTEM}=="usb", ENV{DEVTYPE}=="usb_device", MODE="0674", OWNER="root", GROUP="users"
BUS=="usb", SYSFS{idVendor}=="04b3", SYSFS{idProduct}=="*", MODE="0660", GROUP="users"
```
3. Restart your system.

Manual configuration

To access Toshiba USB devices via `javax.usb`, create a rule in the `udev` directory.

For example, to provide access to all USB devices:

1. Edit the /etc/udev/rules.d/55-libsane.rules rule file.

2. Directly after the LABEL="libsane_rules_begin line add the following:

```
ATTR{idVendor}=="*", ATTR{idProduct}=="*", MODE="0666", GROUP="users", ENV{libsane_matched}="yes"
```

PS/2 keyboard configuration

No additional configuration is required to enable PS/2 scan codes.

Background information

The following information is provided for debug purposes in case of issues with the Toshiba PS/2 attached POS Keyboard.

/etc/ps2kbd.conf file

The ps2kbd.conf file is installed by default in the /etc directory. This file allows the JavaPOS driver to enumerate the Toshiba PS/2 keyboard as a system attached keyboard.

Enabling raw scan codes

To support Toshiba PS/2 keyboard scan codes, the raw scan codes must be enabled. By default, the raw scan codes are disabled by the OS. The JavaPOS RPM will automatically enable raw scan codes on your system.

How to verify the status of atkbd.softraw

Run the command: cat /sys/bus/serio/drivers/atkbd/serio0/softraw. To get raw scan codes, the atkbd.softraw should be set to 0.

Automatic configuration updating grub

Edit /boot/grub/menu.lst and add the atkbd.softraw=0 argument to the kernel line. For example:

```
title SUSE Linux2 enterprise Desktop 11 - 2.6.<version>
root (hd0,6)
kernel /boot/vmlinuz-2.6.<version> root=/dev/disk/by-id/ata-WDC_WD800BB-23FRA0_WD-WCAJD1482394
-part7 resume=/dev/disk/by-id/ata-WDC_WD800BB-23FRA0_WD-WCAJD1482394-part6 splash=silent
showopts vga=0x34b atkbd.softraw=0
initrd /boot/initrd-3.0.<version>
```

Necessary updates for 1.13.2 package

If you are using Toshiba UnifiedPOS for Linux version 1.13.2, it is necessary to update the /etc/init.d/aipstart with the following. At line 96 in /etc/init.d/aipstart, change:

```
if [ ${KERNEL_MAJOR} -lt 16 ]; then
to
if [ ${KERNEL_VERSION} -eq 2 -a ${KERNEL_MAJOR} -lt 16 ]; then
```

USB system attached POS keyboard configuration

For the Toshiba USB POS alphanumeric keyboard to function as a system attached keyboard, the following configuration file must be present:

/opt/ibm/javapos/etc/usbkbd.conf

If the configuration file for the USB system keyboard is not present, the Toshiba Alphanumeric USB Keyboard will be enumerated in Point of Sale mode. This means that the scan codes will not be available to the normal system keyboard mechanism.

Touch device configuration

Please refer to the following documentation for configuration of Toshiba touch devices.

SLED/SLES/SLEPOS 11 SP3: <http://www-01.ibm.com/support/docview.wss?rs=219&uid=pos1R7000429>

Notices

This information was developed for products and services offered in the U.S.A.

Toshiba Global Commerce Solutions may not offer the products, services, or features discussed in this document in other countries. Consult your local Toshiba Global Commerce Solutions representative for information on the products and services currently available in your area. Any reference to a Toshiba Global Commerce Solutions product, program, or service is not intended to state or imply that only that Toshiba Global Commerce Solutions product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any Toshiba Global Commerce Solutions intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-Toshiba Global Commerce Solutions product, program, or service.

Toshiba Global Commerce Solutions may have patents or pending patent applications covering the subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

Toshiba Global Commerce Solutions
Attn: General Counsel
3039 E. Cornwallis Rd
RTP, NC 27709

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: TOSHIBA GLOBAL COMMERCE SOLUTIONS PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. Toshiba Global Commerce Solutions may make improvements and/or changes in the product(s) and/or program(s) described in this publication at any time without notice.

Toshiba Global Commerce Solutions may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any references in this information to non-Toshiba Global Commerce Solutions Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this Toshiba Global Commerce Solutions product and use of those Web sites is at your own risk.

Information concerning non-Toshiba Global Commerce Solutions products was obtained from the suppliers of those products, their published announcements or other publicly available sources. Toshiba Global Commerce Solutions has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-Toshiba Global Commerce Solutions products. Questions on the capabilities of non-Toshiba Global Commerce Solutions products should be addressed to the suppliers of those products.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

Glossary

This glossary includes terms and definitions from:

- *American National Standard Dictionary for Information Systems*, ANSI X3.172-1990, copyright 1990 by the American National Standards Institute (ANSI). Copies may be purchased from the American National Standards Institute, 11 West 42nd Street, New York, New York 10036. Definitions are identified by the symbol (A) after the definition.
- The *Information Technology Vocabulary*, developed by Subcommittee 1, Joint Technical Committee 1, of the International Organization for Standardization and the International Electrotechnical Commission (ISO/IEC JTC1/SC1). Definitions of published parts of this vocabulary are identified by the symbol (I) after the definition; definitions taken from draft international standards, committee drafts, and working papers being developed by ISO/IEC JTC1/SC1 are identified by the symbol (T) after the definition, indicating that final agreement has not yet been reached among the participating National Bodies of SC1.

A

active Able to communicate on the network. A token-ring network adapter is active if it is able to transmit and receive on the network. Operational. Pertaining to a node, or device that is connected or is available for connection to another node or device. Currently transmitting or receiving.

adapter

In the point-of-sale terminal, a circuit card that, with its associated software, enables the terminal to use a function or feature. In a LAN, within a communicating device, a circuit card that, with its associated software and/or microcode, enables the device to communicate over the network.

address

In data communication, the IEEE-assigned unique code, or the unique locally administered code assigned to each device, or workstation connected to a network. A character, group of characters, or a value that identifies a register, a

particular part of storage, a data source, or a data link. The value is represented by one or more characters. To refer to a device, or an item of data by its address. The location in the storage of a computer where data is stored.

address space

The complete range of addresses that is available to a programmer.

all points addressable (APA)

In computer graphics, pertaining to the ability to address and display or not display each picture element (pel) on a display surface.

alphanumeric

Pertaining to a character set containing letters, digits, and other characters, such as punctuation marks.

Alphanumeric point-of-sale keyboard (NANPOS keyboard)

This keyboard consists of a section of alphanumeric keys, a programmable set of point-of-sale keys, a numeric keypad, and system function keys.

American National Standard Code for Information Interchange (ASCII)

The standard code, using a coded character set consisting of 7-bit coded characters (8 bits including parity check), used for information interchange among data processing systems, data communication systems, and associated equipment. The ASCII set consists of control characters and graphics characters.

American National Standards Institute (ANSI)

An organization for the purpose of establishing voluntary industry standards.

NANPOS keyboard

Alphanumeric Point of Sale Keyboard.

ANSI American National Standards Institute.

APA all points addressable.

API Application program interface.

application program

A program written for or by a user that applies to the user's own work. A program written for or by a user that

	<p>applies to a particular application. A program used to connect and communicate with stations in a network, enabling users to perform application-oriented activities.</p>	
application program interface (API)	The formally defined programming language interface that is between a Toshiba system control program or a licensed program and the user of the program.	
array	An arrangement of elements in one or more dimensions.	
ASCII	American National Standard Code for Information Interchange.	
asynchronous	Pertaining to two or more processes that do not depend upon the occurrence of specific events such as timing signals. Without regular time relationship; unexpected or unpredictable with respect to the execution of program instructions.	
attach	To connect a device physically. To make a device a part of a network logically. Compare with <i>connect</i> .	
attaching device	Any device that is physically connected to a network and can communicate over the network.	
B		
backup	Pertaining to a system, device, file, or facility that can be used in the event of a malfunction or the loss of data.	
backup copy	A copy, usually of a program or of a library member, that is kept in case the original or the working copy is unintentionally altered or destroyed.	
bar code	A code representing characters by sets of parallel bars of varying thickness and separation that are read optically by transverse scanning.	
BCD	Binary-coded decimal notation.	
binary	Pertaining to a system of numbers to the base two; the binary digits are 0 and 1. Pertaining to a selection, choice, or condition that has two possible different values or states.	
	binary-coded decimal notation (BCD)	A binary-coded notation in which each of the decimal digits is represented by a binary numeral. For example, in binary-coded decimal notation that uses the weights 8, 4, 2, 1, the number "twenty three" is represented by 0010 0011. In the pure binary numeration system, its representation is 10111.
	bit	Either of the binary digits: a 0 or 1.
	bit map	A coded representation in which each bit or group of bits represents or corresponds to an item; for example, a configuration of bits in main storage in which each bit indicates whether a peripheral device or a storage block is available or in which each group of bits corresponds to one pixel of a display image.
	bits per second (bps)	The rate at which bits are transmitted per second.
	block size	The minimum size that frames are grouped into for retransmission. The number of data elements (such as bits, bytes, characters, or records) that are recorded or transmitted as a unit.
	break scan code	The hardware scan code received by the keyboard device driver when a key on the keyboard is physically pressed.
	bps	Bits per second.
	Bps	Bytes per second.
	buffer	A portion of storage used to hold input or output data temporarily. A routine or storage used to compensate for a difference in data rate or time of occurrence of events, when transferring data from one device to another.
	byte	A string that consists of a number of bits, treated as a unit, and representing a character. A binary character operated upon as a unit and usually shorter than a computer word. A string that consists of a particular number of bits, usually 8, that is treated as a unit, and that represents a character. A group of 8 adjacent binary digits that represent one extended binary-coded decimal interchange code (EBCDIC). See <i>n-bit byte</i> .

C		
C	A high-level programming language designed to optimize run time, size, and efficiency.	executable program (an object program). To translate a program written in a high-level programming language into a machine language program.
call	The action of bringing a function or subprogram into effect, usually by specifying the entry conditions and jumping to an entry point.	
card reader	See <i>magnetic stripe reader, (MSR)</i> .	
cash drawer	A drawer at a point-of-sale terminal that can be programmed to open automatically. See <i>till</i> .	
channel	A functional unit, controlled by a host computer, that handles the transfer of data between processor storage and local peripheral equipment. A path along which signals can be sent. The portion of a storage medium that is accessible to a given reading or writing station.	
clear	To delete data from a screen or from memory.	
code page	A particular assignment of hexadecimal identifiers to graphic characters.	
code point	A 1-byte code representing one of 256 potential characters.	
command	A request for performance of an operation or execution of a program. A character string from a source external to a system that represents a request for system action.	
compile	To translate all or part of a program expressed in a high-level language into a computer program expressed in an intermediate language, an assembly language, or a machine language. To prepare a machine language program from a computer program written in another programming language by making use of the overall logic structure of the program, or generating more than one computer instruction for each symbolic statement, or both, as well as performing the function of an assembler. To translate a source program into an	compiler A program that decodes instructions written as pseudo codes and produces a machine language program to be executed at a later time. Contrast with <i>interpretive routine</i> .
component		Any part of a network other than an attaching device, such as an IBM® 8228 Multistation Access Unit. Hardware or software that is part of a functional unit.
configuration		configuration The group of devices, options, and programs that make up a data processing system or network as defined by the nature, number, and chief characteristics of its functional units. More specifically, the term refer to a hardware configuration or a software configuration. See also <i>system configuration</i> .
configuration file		configuration file The collective set of definitions that describes a configuration.
connect		connect In a LAN, to physically join a cable from a station to an access unit or network connection point. Contrast with <i>attach</i> .
constant		constant String or numeric value that does not change throughout program execution.
control character		control character A character whose occurrence in a particular context initiates, modifies, or stops a control operation. A control character may be recorded for use in a subsequent action, and it may have a graphic representation in some circumstances.
CRC		CRC Cyclic redundancy check.
customize		customize To tailor a program or store system through option selection.
cyclic redundancy check (CRC)		cyclic redundancy check (CRC) Synonym for <i>frame check sequence (FCS)</i> .
D		
data		data A representation of facts, concepts, or

instructions in a formalized manner suitable for communication, interpretation, or processing by human or automatic means. Any representations such as characters or analog quantities to which meaning is or might be assigned.

data communication

Transfer of information between functional units by means of data transmission according to a protocol. The transmission, reception, and validation of data.

data file

A collection of related data records organized in a specific manner; for example, a payroll file (one record for each employee, showing such information as rate of pay and deductions) or an inventory file (one record for each inventory item, showing such information as cost, selling price, and number in stock.) See also *data set, file*.

data set

Logically related records treated as a single unit. See also *file*.

data terminal equipment (DTE)

That part of a data station that serves as a data source, data receiver, or both. Equipment that sends or receives data, or both.

data transmission

The conveying of data from one place for reception elsewhere by means of telecommunications.

data type

The mathematical properties and internal representation of data and functions.

DBCS Double-byte character set.

DCE Data circuit-terminating equipment.

default

Pertaining to an attribute, value, or option that is assumed when none is explicitly specified.

default value

The value the system supplies when the user does not specify a value.

device A mechanical, electrical, or electronic contrivance with a specific purpose. An

input/output unit such as a terminal, display, or printer. See also *attaching device*.

device connection

The connection between an application and a hardware device created by the Toshiba JavaPOS system when the application opens a device.

device descriptor

An identifier that represents a device to the Toshiba JavaPOS system application programming interface. This identifier is created by the Toshiba JavaPOS system when the application opens a device.

device driver

The code needed to attach and use a device on a computer or a network.

digital Pertaining to data in the form of digits. Contrast with *analog*. Pertaining to data consisting of numerical values or discrete units.

direct file

A file in which records are assigned specific record positions. No matter what order the records are put in a direct file, they always occupy the assigned position. A direct file is the same as a random file except that a direct file contains no delimiting characters, such as quotes enclosing string fields.

directory

A table of identifiers and references that correspond to items of data. An index that a control program uses to locate one or more blocks of data that are stored in separate areas of a data set in direct access storage.

disabled

Pertaining to a state of a processing unit that prevents the occurrence of certain types of interruptions. Pertaining to the state in which a transmission control unit or audio response unit cannot accept incoming calls on a line.

disk

A round, flat plate coated with a magnetic substance that is used to store computer data. See also *integrated disk, fixed disk*.

Disk Operating System (DOS)

An operating system for computer systems that use disks and diskettes for auxiliary storage of programs and data.

display

A visual presentation of data. A device that presents visual information to the point-of-sale terminal operator and to the customer, or to the display station operator.

distributed

Physically separate but connected by cables.

DLL See *dynamic link library*.

DOS Disk Operating System.

double-byte character set (DBCS)

A set of characters in which each character is represented by 2 bytes. Languages such as Japanese, Chinese, and Korean, which contain more symbols than can be represented by 256 code points, require double-byte character sets. Because each character requires 2 bytes, the typing, display, and printing of DBCS characters requires hardware and programs that support DBCS. Contrast with single-byte character set.

DRAM

Dynamic RAM. See *RAM*.

driver Software component that controls a device.

DTE Data terminal equipment.

dump To write at a particular instant the contents of storage, or part of storage, onto another data medium for the purpose of safeguarding or debugging the data. Data that has been dumped.

duplex

In data communication, pertaining to a simultaneous two-way independent transmission in both directions. Synonymous with *full-duplex*. contrast with *half-duplex*.

DVD-ROM

Digital-video-disk read-only memory.

dynamic link library (DLL)

In the Windows operating systems, the delayed connection of a library to a routine until load time or run time.

E

EAN European article number.

EIA Electronics Industries Association.

EIA-232

In data communications, a specification of the Electronic Industries Association (EIA) that defines the interface between data terminal equipment (DTE) and data circuit-terminating equipment (DCE) using serial binary data interchange. Formerly known as RS-232.

enabled

On a LAN, pertaining to an adapter or device that is active, operational, and able to receive frames from the network. Pertaining to a state of a processing unit that allows the occurrence of certain types of interruptions. Pertaining to the state in which a transmission control unit or an audio response unit can accept incoming calls on a line.

end-of-file

An internal label, immediately following the last record of a file, signaling the end of that file.

error message

A message that is issued because an error has been detected.

escape character

Code extension character used, in some cases, with one or more succeeding characters to indicate by some convention that the coded representation following the character or the group of characters are to be interpreted according to a different code or different character set.

European article number (EAN)

A number that is assigned to and encoded on an article of merchandise for scanning in some countries.

event

Processing unit containing price changes and item file updates. All records in an event share common characteristics such as type of change and event due date. An occurrence of significance to a task; for example, the completion of an asynchronous operation, such as an I/O operation.

exception

An abnormal condition such as an I/O error encountered in processing a data set or a file. See also *overflow exception* and *underflow exception*.

exit

To execute an instruction or statement within a portion of a program in order to

terminate the execution of that portion.
Note: Such portions of programs include loops, routines, subroutines, and modules.

expansion board

In a Toshiba Personal Computer, a panel containing microchips that a user can install in an expansion slot to add memory or special features. Synonymous with *expansion card*, *extender card*.

expansion card

Synonym for *expansion board*.

extender card

Synonym for *expansion board*.

F

fat-finger

When two keys are pressed faster than the value specified using the PosNfatFingerTimeOut resource. This could occur under any of the following conditions: 1) Two keys on the keyboard were pressed at the same time. 2) The operator is keying faster than 25 keys per second. 3) A double key is not defined to the keyboard device handler.

field On a data medium or a storage medium, a specified area used for a particular category of data; for example, a group of character positions used to enter or display wage rates on a panel.

FIFO First-in-first-out.

file A named set of records stored or processed as a unit. For example, an invoice form a record and the complete set of such records form a file. See also *data file* and *data set*.

file name

A name assigned or declared for a file. The name used by a program to identify a file.

first-in-first-out (FIFO)

A queuing technique in which the next item to be retrieved is the item that has been in the queue for the longest time.

fixed disk (drive)

In a personal computer system unit, a disk storage device that reads and writes on rigid magnetic disks. It is faster and has a larger storage capacity than a diskette and is permanently installed.

flag A character or indicator that signals the

occurrence of some condition, such as the setting of a switch, or the end of a word.

flash memory

A data-storage device that is programmable, erasable, and does not require continuous power. The chief benefit of flash memory over other programmable and erasable data storage devices is that it can be reprogrammed without being removed from the circuit board.

frame

The unit of transmission in some LANs, including the Toshiba Token-Ring Network and the Toshiba PC Network. It includes delimiters, control characters, information, and checking characters. On a token-ring network, a frame is created from a token when the token has data appended to it. On a token-bus network (Toshiba PC Network), all frames including the token frame contain a preamble, start delimiter, control address, optional data and checking characters, end delimiter, and are followed by a minimum silence period. A housing for machine elements. In synchronous data link control (SDLC), the vehicle for every command, every response, and all information that is transmitted using SDLC procedures. Each frame begins and ends with a flag.

frequency

The rate of signal oscillation, expressed in hertz (cycles per second).

full-duplex

Synonym for *duplex*.

function

A specific purpose of an entity, or its characteristic action. A subroutine that returns the value of a single variable. In data communications, a machine action such as a carriage return or line feed.

G

GCGID

See *Graphic Character Global Identifier*.

global

Pertaining to that which is defined in one subdivision of a computer program and used in at least one other subdivision of that computer program.

**Graphic Character Set Global Identifier
(GCGID)**

A 4- to 8-character identifier assigned to a registered graphic character in a Toshiba registry.

group A set of related records that have the same value for a particular field in all records. A collection of users who can share access authorities for protected resources. A list of names that are known together by a single name.

H**half-duplex**

In data communication, pertaining to transmission in only one direction at a time. Contrast with *duplex*.

hardware

Physical equipment as opposed to programs, procedures, rules, and associated documentation.

hertz (Hz)

A unit of frequency equal to one cycle per second. **Note:** In the United States, line frequency is 60Hz or a change in voltage polarity 120 times per second; in Europe, line frequency is 50Hz or a change in voltage polarity 100 times per second.

hexadecimal notation

Notation for the base-16 number system using the symbols 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, and F to represent values from 0 to 15 (decimal).

hot key

The key combination used to change from one session to another on a workstation.

hot plug

To connect a USB I/O device to the universal serial bus without powering the host system down.

hot unplug

To disconnect a USB I/O device from the universal serial bus without powering the host system down.

Hz See *hertz*.

I**identifier**

String of characters used to name elements of a program, such as variable names, reserved words, and user-defined function names.

inactive

Not operational. Pertaining to a node or device not connected or not available for connection to another node or device. In the Toshiba Token-Ring Network, pertaining to a station that is only repeating frames or tokens, or both.

information (I) frame

A frame in I format used for numbered information transfer. See also *supervisory frame*, *unnumbered frame*.

initialize

In a LAN, to prepare the adapter (and adapter support code, if used) for use by an application program.

initial program load (IPL)

The initialization procedure that causes an operating system to begin operation.

input device

Synonym for *input unit*.

input/output device

See *I/O device*.

input/output (I/O)

Pertaining to a device whose parts can perform an input process and an output process at the same time. Pertaining to a functional unit or channel involved in an input process, output process, or both, concurrently or not, and to the data involved in such a process.

input unit

A device in a data processing system that is used to enter data into the system. Synonymous with *input device*.

instance

An occurrence of a particular device or object. For example, two instances of the PosDisplay class name can be Shopper1 and Shopper2, where these instances refer to the same physical display.

integrated

Arranged together as one unit.

integrated disk

An integral part of the processor that is used for magnetically storing files, application programs, and diagnostics. Synonymous with *disk*.

interaction

A basic unit used to record system activity, consisting of the acceptance of a

line of terminal input, processing of the line, and a response, if any.

interface

A shared boundary between two functional units, defined by functional characteristics, common physical interconnection characteristics, signal characteristics, and other characteristics as appropriate. A shared boundary. An interface may be a hardware component to link two devices or a portion of storage or registers accessed by two or more computer programs. Hardware, software, or both, that links systems, programs, or devices.

interleave

To insert segments of one program into another program so that the two programs can, in effect, be executed at the same time.

International Organization for Standardization (ISO)

An organization of national standards bodies from various countries established to promote development of standards to facilitate international exchange of goods and services, and develop cooperation in intellectual, scientific, technological, and economic activity.

interpretive routine

A routine that decodes instructions written as pseudocodes and immediately executes the instructions. Contrast with *compile*.

I/O Input/output.

I/O device

Equipment for entering and receiving data from the system.

IPL Initial program load.

ISO International Organization for Standardization.

item One member of a group. In a store, one unit of a commodity, such as one box, one bag, or one can. Usually an item is the smallest unit of a commodity to be sold.

J

JIS Japanese Industrial Standard

JUCC Japanese Unified Cash Card

K

keyboard

A group of numeric keys, alphabetic keys, special character keys, or function keys used for entering information into the terminal and into the system.

kHz Kilohertz. See also *hertz*.

kilohertz (kHz)

A thousand hertz. See also *hertz*.

L

LED Light-emitting diode.

lift-off When a pointing device is removed from a touch-sensitive surface.

light-emitting diode (LED)

A semiconductor chip that gives off visible or infrared light when activated.

line On a terminal, one or more characters entered before a return to the first printing or display position.

link The combination of physical media, protocols, and programming that connects devices on a network. In computer programming, the part of a program, in some cases a single instruction or an address, that passes control and parameters between separate portions of the computer program. To interconnect items of data or portions of one or more computer programs.

listing A printout, usually prepared by a language translator, that lists the source code.

load In computer programming, to enter data into memory or working registers.

lock To disable a device, such as a scanner or MSR, so that it cannot receive input. See also *unlock*.

logging

The chronological recording of events occurring in a system or a subsystem for accounting or data collection purposes.

logical connection

In a network, devices that can communicate or work with one another because they share the same protocol. See also *physical connection*.

logon (n)

The procedure for starting up a point-of-sale terminal or store controller for normal sales operations by

sequentially entering the correct security number and transaction number.
Synonymous with *sign-on*.

log on (v)

To initiate a session. In SNA products, to initiate a session between an application program and a logical unit (LU).
Synonymous with *sign-on*.

loop A set of instructions that may be executed repeatedly while a certain condition prevails. See also *store loop*. A closed unidirectional signal path connecting input/output devices to a network.

M

macro An instruction that causes the execution of a predefined sequence of instructions in the same source language.

magnetic stripe

The magnetic material (similar to recording tape) on merchandise tickets, credit cards, and employee badges. Information is recorded on the stripe for later “reading” by the magnetic stripe reader (MSR) or magnetic wand reader attached to the point-of-sale terminal.

magnetic stripe reader (MSR)

A device that reads coded information from a magnetic stripe on a card, such as a credit card, as it passes through a slot in the reader.

make scan code

The hardware scan code received by the keyboard device driver when a key on the keyboard is physically pressed.

Mb Megabit.

MB Megabyte.

megabit (Mb)

A unit of measure for throughput. 1 megabit = 1,048,576 bits.

megabyte (MB)

A unit of measure for data. 1 megabyte = 1,048,576 bytes.

memory

Program-addressable storage from which instructions and other data can be loaded directly into registers for subsequent execution or processing.

message

An arbitrary amount of information

whose beginning and end are defined or implied. A group of characters and control bit sequences transferred as an entity. In telecommunication, a combination of characters and symbols transmitted from one point to another. A logical partition of the user device’s data stream to and from the adapter. See also *error message*, *operator message*.

microcode

One or more microinstructions. A code, representing the instructions of an instruction set, that is implemented in a part of storage that is not program-addressable. To design, write, and also test one or more microinstructions.

microprocessor

An integrated circuit that accepts coded instructions for execution. The instructions may be entered, integrated, or stored internally.

migration

Installation of a new version of a release of a program to replace an earlier version or release.

modem (Modulator/DEModulator)

A device that converts digital data from a computer to an analog signal that can be transmitted in a telecommunication line, and converts the analog signal received to data for the computer.

modulo check

A function designed to detect most common input errors by performing a calculation on values entered into a system by an operator or scanning device.

monitor

A functional unit that observes and records selected activities for analysis within a data processing system. Possible uses are to show significant departures from the norm, or to determine levels of utilization of particular functional units. Software or hardware that observes, supervises, controls, or verifies operations of a system.

MSR Magnetic stripe reader.

N

name An alphanumeric term that identifies a data set, statement, program, or cataloged procedure.

n-bit byte

A string that consists of n bits.

network

A configuration of data processing devices and software connected for information interchange. An arrangement of nodes and connecting branches. Connections are made between data stations.

noise A disturbance that affects a signal and that can distort the information carried by the signal. Random variations of one or more characteristics of any entity, such as voltage, current, or data. Loosely, any disturbance tending to interfere with normal operation of a device or system.

nonvolatile random access memory (NVRAM)

Random access memory that retains its contents after electrical power is shut off.

NVRAM

nonvolatile random access memory.

O

offline

Operation of a functional unit without the control of a computer or control unit.

online Operation of a functional unit that is under the continual control of a computer or control unit. The term also describes a user's access to a computer using a terminal.

open To make an adapter ready for use. A break in an electrical circuit. To make a file ready for use.

operating system

Software that controls the execution of programs. An operating system provides services such as resource allocation, scheduling, input/output control, and data management.

operation

A defined action, namely, the act of obtaining a result from one or more operands in accordance with a rule that completely specifies the result for any permissible combination of operands. A program step undertaken or executed by a computer. An action performed on one

or more data items, such as adding, multiplying, comparing, or moving.

operator

A symbol that represents the action being performed in a mathematical operation. A person who operates a machine.

operator message

A message from the operating system or a program telling the operator to perform a specific function or informing the operator of a specific condition within the system, such as an error condition.

option A specification in a statement, a selection from a menu, or a setting of a switch, that may be used to influence the execution of a program. A hardware or software function that may be selected or enabled as part of a configuration process. A piece of hardware (such as a network adapter) that can be installed in a device to modify or enhance device function.

output device

A device in a data processing system by which data can be received from the system. Synonymous with *output unit*.

output unit

Synonym for *output device*.

overflow exception

A condition caused by the result of an arithmetic operation having a magnitude that exceeds the largest possible number. See also *underflow exception*.

P

parameter

A name in a procedure that is used to refer to an argument passed to that procedure. A variable that is given a constant value for a specified application and that may denote the application. An item in a menu or for which the user specifies a value or for which the system provides a value when the menu is interpreted. Data passed between programs or procedures.

parity bit

A binary digit appended to a group of binary digits to make the sum of all the digits (including the appended binary digit) either always odd (odd parity) or always even (even parity).

parity (even)

A condition when the sum of all of the digits in an array of binary digits is even.

parity (odd)

A condition when the sum of all of the digits in an array of binary digits is odd.

physical connection

The ability of two connectors to mate and make electrical contact. In a network, devices that are physically connected can communicate only if they share the same protocol. See also *logical connection*.

PLD Power line disturbance.

PLU Price Look Up.

plug A connector for attaching wires from a device to a cable, such as a store loop. A plug is inserted into a receptacle or plug. To insert a connector into a receptacle or socket.

pointer

An identifier that indicates the location of an item of data in memory. A data element that indicates the location of another data element. A physical or symbolic identifier of a unique target.

point-of-sale terminal

A unit that provides point-of-sale transaction, data collection, credit authorization, price look-up, and other inquiry and data entry functions.

polling

Interrogation of devices for purposes such as to avoid contention, to determine operational status, or to determine readiness to send or receive data. In data communication, the process of inviting data stations to transmit, one at a time. The polling process usually involves the sequential interrogation of several data stations.

polling characters (address)

A set of characters specific to a terminal and the polling operation; response to these characters indicates to the computer whether the terminal has a message to enter.

port

An access point for data entry or exit. A connector on a device to which cables for other devices such as display stations and printers are attached. Synonymous with *socket*.

post To affix to a usual place. To provide items such as return code at the end of a command or function. To define an appendage routine. To note the occurrence of an event.

POST Power-On Self Test.

power line disturbance (PLD)

Interruption or reduction of electrical power.

Power-On Self Test (POST)

A series of diagnostic tests that are run automatically each time the computer's power is switched on.

problem determination

The process of determining the source of a problem as being a program component, a machine failure, a change in the environment, a common-carrier link, a user-supplied device, or a user error.

procedure

A set of related control statements that cause one or more programs to be performed. In a programming language, a block, with or without formal parameters, whose execution is invoked by means of a procedure call. A set of instructions that gives a service representative a step-by-step procedure for tracing a symptom to the cause of failure.

process

An instance of an executing application and the resources it is using.

processor

In a computer, a functional unit that interprets and executes instructions.

prompt

A character or word displayed by the operating system to indicate that it is ready to accept input.

Q

queue A line or list formed by items in a system waiting for service; for example, tasks to be performed or messages to be transmitted in a message routing system.

R

RAM Random access memory.

random access

An access mode in which specific logical

	records	records are obtained from or placed into a mass storage file in a nonsequential manner.
	random access memory (RAM)	A computer's or adapter's volatile storage area into which data may be entered and retrieved in a nonsequential manner.
read	To acquire or to interpret data from a storage device, from a data medium, or from another source.	
	read-only memory (ROM)	A computer's or adapter's storage area whose contents cannot be modified by the user except under special circumstances.
receive	To obtain and store information transmitted from a device.	
record	A collection of related items of data, treated as a unit; for example, in stock control, each invoice could constitute one record. A complete set of such records form a file.	
register	A storage area in a computer's memory where specific data is stored. Registers are used in the actual manipulation of data values during the execution of a program. A storage device having a specified storage capacity such as bit, byte, or computer word, and usually intended for a special purpose. In the Toshiba Store System, a term that refers to the point-of-sale terminal.	
remove	To take an attaching device off a network. To stop an adapter from participating in data passing on a network.	
resource	An element that affects the way devices behave.	
resource set	The set of resources associated with a device.	
response	The information the network control program sends to the access method, usually in answer to a request received from the access method. (Some responses, however, result from conditions occurring within the network control program, such as accumulation of error statistics.)	
	retry	In data communication, sending the current block of data a prescribed number of times or until it is entered correctly and accepted.
	return code	A value (usually hexadecimal) provided by an adapter or a program to indicate the result of an action, command, or operation. A code used to influence the execution of succeeding instructions. A value established by the programmer to be used to influence subsequent program action. This value can be printed as output or loaded in a register.
	ROM	Read-only memory.
	routine	Part of a program, or a sequence of instructions called by a program, that may have some general or frequent use.
	S	
	satellite	A computer that is under the control of another computer and performs subsidiary operations. An offline auxiliary computer.
	SBCS	Single-byte character set.
	scan	To pass an item over or through the scanner so that the encoded information is read. See also <i>wanding</i> .
	scan codes	When a key is pressed on any keyboard, the keyboard device driver receives a code that is called a <i>make scan code</i> . Each key has a different code. When a key is released on some keyboards, the keyboard device driver receives a code that is called a <i>break scan code</i> . These codes are translated into ASCII character codes using the code page that the application is using.
	scanner	A device that examines the bar code on merchandise tickets, credit cards, and employee badges and generates analog or digital signals corresponding to the bar code.
	scroll	To move all or part of the display image vertically or horizontally to display data that cannot be observed within a single display image. See also <i>page</i> (2).

segment

See *cable segment*, *LAN segment*, *ring segment*.

sequential file

A disk file in which records are read from or placed into the file according to the order they are processed.

session

A connection between two application programs that allows them to communicate. In SNA, a logical connection between two network addressable units that can be activated, tailored to provide various protocols, and deactivated as requested. The data transport connection resulting from a call or link between two devices. The period of time during which a user of a node can communicate with an interactive system, usually the elapsed time between log on and log off. In network architecture, an association of facilities necessary for establishing, maintaining, and releasing connections for communication between stations.

signal

A time-dependent value attached to a physical phenomenon for conveying data. A variation of a physical quantity, used to convey data.

sign-on

A procedure to be followed at a terminal or workstation to establish a link to a computer. To begin a session at a workstation.

single-byte character set (SBCS)

A character set in which each character is represented by a one-byte code. Contrast with *double-byte character set*.

SLRS SUSE LINUX Retail Solution.**socket** Synonym for *port* (2).**state transition**

The act of moving from one conversation state to another.

station

A point-of-sale terminal that consists of a processing unit, a keyboard, and a display. It can also have input/output devices, such as a printer, a magnetic stripe reader or cash drawers. A communication device attached to a network. The term used most often in

LANs is an *attaching device* or *workstation*. An input or output point of a system that uses telecommunication facilities; for example, one or more systems, computers, terminals, devices, and associated programs at a particular location that can send or receive data over a telecommunication line. See also *attaching device*, *workstation*.

subdirectory

Any level of file directory lower than the root directory within a hierarchical file system.

subroutine

Section of code that performs a specific task and is logically separate from the rest of the program.

subsystem

A secondary or subordinate system, or programming support, usually capable of operating independently of or asynchronously with a controlling system.

summary journal

A record of the terminal operational activity that is printed at the terminal.

switch On an adapter, a mechanism used to select a value for, enable, or disable a configurable option or feature.**system**

In data processing, a collection of people, machines, and methods organized to accomplish a set of specific functions. See also *data processing system* and *operating system*.

system configuration

A process that specifies the devices and programs that form a particular data processing system.

system unit

A part of a computer that contains the processing unit, and may contain devices such as disk and diskette drives. In a Toshiba Personal Computer, the unit that contains the processor circuitry, read-only memory (ROM), random access memory (RAM), and the I/O channel. It may have one or more disk or diskette drives. In a Toshiba 4683/4684 terminal, the part of the terminal that contains the processing unit, ROM, RAM, disk and diskette drives, and the I/O channel.

T

terminal

In data communication, a device, usually equipped with a keyboard and a display, capable of sending and receiving information over a communication channel.

thread A unit of execution within a process. It uses the resources of the process.

throughput

A measure of the amount of work performed by a computer system over a given period of time, for example, number of jobs per day. A measure of the amount of information transmitted over a network in a given period of time. For example, a network's data transfer rate is usually measured in bits per second.

till A tray in the cash drawer of the point-of-sale terminal, used to keep the different denominations of bills and coins separated and easily accessible.

Toshiba Disk Operating System (DOS)

A disk operating system based on MS-DOS.¹

touch-down

When contact is made with a touch-sensitive surface.

trace A record of the execution of a computer program. It exhibits the sequences in which the instructions were executed. A record of the frames and bytes transmitted on a network.

transaction

The process of recording item sales, processing refunds, recording coupons, handling voids, verifying checks before tendering, and arriving at the amount to be paid by or to a customer. The receiving of payment for merchandise or service is also included in a transaction. In an SNA network, an exchange between two programs that usually involves a specific set of initial input data that causes the execution of a specific task or job. Examples of transactions include the entry of a customer's deposit that results in the updating of the customer's balance,

and the transfer of a message to one or more destination points.

transition

See *state transition*.

transmission

The sending of data from one place for reception elsewhere.

transmit

To send information from one place for reception elsewhere.

typematic

A keyboard button that will continue to enter characters or repeat its function as long as the button is held down.

U

underflow exception

A condition caused by the result of an arithmetic operation having a magnitude less than the smallest possible nonzero number. See also *overflow exception*.

unlock

To enable a device, such as a scanner or MSR, so that it can read data. See also *lock*.

universal product code (UPC)

An encoded number that can be assigned to and printed on or attached to an article of merchandise for scanning.

Universal Serial Bus (USB)

A serial interface standard for telephony and multimedia connections to personal computers.

Universal Serial Bus (USB), powered

A powered-USB connector provides additional power from the host system. A powered-USB receptacle consists of two connectors stacked vertically inside the common housing. The upper connector contains four contacts that are used for powering the attached device.

UDC User defined character.

UPC Universal product code.

user Category of identification defined for file access protection. A person using a program or system.

user defined character (UDC)

User defined character.

1. MS-DOS is a trademark of the Microsoft Corporation.

user interface

Hardware, software, or both that allows a user to interact with and perform operations on a system, program, or device.

V**variable**

A named entity that is used to refer to data and to which values can be assigned. Its attributes remain constant, but it can refer to different values at different times. In computer programming, a character or group of characters that refers to a value and, in the execution of a computer program, corresponds to an address. A quantity that can assume any of a given set of values.

version

A separate Toshiba-licensed program, based on an existing Toshiba-licensed program, that usually has significant new code or new function.

W**wanding**

Passing the tip of the wand reader over information encoded on a merchandise ticket, credit card, or employee badge.

workstation

An I/O device that allows either transmission of data or the reception of data (or both) from a host system, as needed to perform a job: for example, a display station or printer. A configuration of I/O equipment at which an operator works. A terminal or microcomputer, usually one connected to a mainframe or network, at which a user can perform tasks.

Index

Numerics

4674 POS Keyboard (built-in) 345
4674 POS Keyboard (built-in) scan codes 345
4685 keyboards
 4685 keyboard K02 Ultra 7 347
 4685 keyboard K02 Ultra 7 scan codes 347
 4685 keyboard K02 Ultra 7 with MSR/E 348
 4685 keyboard K02 Ultra 7 with MSR/E scan codes 348
 4685 Keyboard Model K01 scan code set 346
 4685 keyboard Model KC1 349
 4685 keyboard Model KC1 scan codes 349
 4685 Point of Sale Keyboard Model K01 346
4685 Point of Sale Keyboard Model K01 layout 346
4820 Toshiba SurePoint Solution Keypad 350
4820 Toshiba SurePoint Solution Keypad (system-attached) 352

A

accessibility of publications xi
additional JavaPOS information 159, 201, 280, 295
additional OPOS information 163, 242
 scanner detection 289
AddLineFeed 189
Alphanumeric Point of Sale (NANPOS) Keyboard
 country dependent 361
 layout 369
Arabic code page support
 JavaPOS 188
 OPOS 219

B

bar code reader statistics 259
barcode printing 250
baudRate 41
bitmap print quality 189
bitmap printing 208
Brazil-Portuguese keyboard layout 364
break scan code 303

C

Canadian French keyboard layout
 Alphanumeric Point of Sale, country dependent 361
 Retail Alphanumeric Point of Sale, country dependent 364

CANPOS keyboard 153
CANPOS keyboard layout 329
capability definitions 123
cash drawer 51, 259, 291, 297
 JavaPOS configuration 54
 OEM support 57
 OPOS configuration 57
 supported properties and methods 51, 53
 USB support 57
Cash Drawer
 UnifiedPOS device statistics properties 431
cash drawer device driver 56
CenterOfZero 294
character sets and code pages 375
CharacterSetASCIIBehavior 121
check flipping support 209
check scanner 59
 calibration command 62
 get check side 64
 JavaPOS DirectIO calls 62
 scanner image quality 64
 set check side 63
 supported devices 59
 supported properties and methods 59
Check Scanner
 UnifiedPOS device statistics properties 433
check scanner statistics 59
check scanning support 233
checkout keyboards
 layouts 303
code pages
 1116 400
 1117 401
 1118 402
 1250 403
 1251 404
 1252 405
 1253 406
 1254 407
 1255 408
 1256 409
 1257 410
 1381 428
 437 379
 775 380
 808 381
 819 382
 848 383
 850 384
 852 385
 855 386
 857 387
 858 388
 860 389
 861 390
 862 391
 863 392
code pages (*continued*)
 864 393
 865 394
 866 395
 867 396
 869 397
 872 398
 874 399
 932 411
 936 412
 949 413, 415
 950 414
 951 415
code pages and character sets 375
color printing 253
color support 190
command format
 direct I/O, keyboard 157
common properties 41
common settings 41
configurable parameters, scanner 262
configuration utility, OPOS 23
configuring devices 17
configuring devices for OPOS 22
configuring the Alphanumeric Point of Sale keyboard 164
connectivity testing 27
Control Center utility 21
CursorState 121

D

Danish keyboard layout 364
DBCS printer limitations 254
DBCS support 253
Default8LPI 190
DefaultLargeFont 190
deployment 30
descriptors 123
device configuration 17
device sharing 254
deviceBus 41
deviceNumber 55
DirectIO.GET_FONT_INFO_ID 201
DisplayRequired 294
DocAdvance 190
Double key support 161
DoubleHighAndDoubleWideIsQuad 190

E

EIA-232 device communication 19
EIA-232 PosPrinter device detection 44
Enable/Disable Partial Cut Detection 191
enableLaserOnOffSwitch 264
enableOnlineWatcher 147
enableProgrammingViaBarcodes 264
enableSwitchControlledVolumeAdjust 264
EnforceZeroReturn 294

ER_CLEAR in the POSPrinter 214
escape sequence handling 205

F

Finnish keyboard layout 367
fiscal printer 65
 device sensing 84
 JavaPOS configuration 80
 JavaPOS DirectIO calls 80
 monitor detection 80
 supported devices 65
 supported properties and
 methods 66
Fiscal Printer
 UnifiedPOS device statistics
 properties 434
FISCAL_INFORMATION
 subcommand 81
FISCAL_Notify subcommand 82
FISCAL_READ subcommand 81

G

German keyboard layout
 Alphanumeric Point of Sale, country
 dependent 362
 Retail Alphanumeric Point of Sale,
 country dependent 365

H

hard totals 99
 OPOS configuration 101
 supported properties and
 methods 99
Hard Totals

I

iButton 107
impClass 42
installation
 interactive upgrade 8
 Linux 8
 silent upgrade 8
 unattended 6
 unattended flags 6
 upgrade 8

installation (*continued*)

 Windows 5

 Windows interactive 5

 Windows silent 6

installation package 10

Installing UnifiedPOS 5

installing USB System attached

 keyboard 154

interactive installation, Windows 5

Introduction 3

Italian keyboard layout

 Alphanumeric Point of Sale, country

 dependent 362

 Retail Alphanumeric Point of Sale,

 country dependent 366

J

JavaPOS

 FISCAL_DATA_AVAIL

 DirectIOEvent 83

 FISCAL_ERROR DirectIOEvent 82

 FISCAL_IPL_END_STATUS event 83

 FISCAL_RAW_STATUS event 83

 FISCAL_STATUS DirectIOEvent 83

 supported methods 41

 supported properties 41

JavaPOS configuration 17, 188, 261, 293

 cash drawer 54

 fiscal printer 80

 keylock 110

 line display 121

 MICR 131

JavaPOS configuration file update 159

JavaPOS DirectIO calls 122, 157, 198

JavaPOS Firmware Update 159

JavaPOS problems 33

JavaPOS support 431

 Cash Drawer 431

 Check Scanner 433

 Fiscal Printer 434

 Hard Totals 436

 Keylock 437

 Line Display 438

 MICR 440

 Motion Sensor 441

 MSR 441

 POSKeyboard 444

 POSPrinter 446

 Scale 449

 Scanner 449

 Tone Indicator 451

Javax.usb for Linux 34

jpos.xml file 17

K

kernel mode drivers 12

 SLE 11 SP3 12

keyboard configuration 165

Keyboard definition file 160

Keyboard V layout 344

keyboards

 4685 Point of Sale Keyboard Model

 K01, layout 346

keyboards (*continued*)

 50-Key Modifiable Layout Keyboard

 and Operator Display, layout 303

 50-Key Modifiable Layout Keyboard,
 layout 303

 Alphanumeric Point of Sale
 (CANPOS) Keyboard, layout 369

 Modifiable Layout Keyboard with
 Card Reader, layout 311

 PC Point of Sale Keyboard,
 layout 316

 Retail Alphanumeric Point of Sale
 country dependent keyboards 363

 Retail Alphanumeric Point of Sale
 Keyboard, 316

 Retail Point of Sale Keyboard VI,
 layout 304

 Retail Point of Sale Keyboard with
 Card Reader and Display,
 layout 304

 Retail Point of Sale Keyboard,
 layout 304

keylock 103

 five-position 106

 four-position 106

 iButton 107

JavaPOS configuration 110

OPOS configuration 110

PositionCount 110

six-position 107

 supported devices 103

 supported properties and
 methods 108

 three-position 105

 two-position 104

Keylock

 UnifiedPOS device statistics

 properties 437

Knowledgebase 37

Known Scanner Issues 283

L

layouts, point-of-sale keyboards 303

line display 113

 JavaPOS configuration 121

 OPOS configuration 126

 supported devices 113

 supported properties and
 methods 113, 119

Line Display

 UnifiedPOS device statistics

 properties 438

line wrapping 206

lineDisplayId 121

Linux installation 8

M

Magnetic Ink Character Recognition 129

magnetic stripe reader 143

 JavaPOS configuration 147

 supported devices 143

 supported properties and

 methods 144

make scan code 303

mapping POS keys 165
 MICR 129
 data 131
 exception file 136
 exceptions 136
 JavaPOS configuration 131
 supported devices 129
 supported properties and
 methods 129
 UnifiedPOS device statistics
 properties 440
 MICR exception tables 132
 Model 4A font file format 248
 modifying device behavior 46
 motion sensor 139
 Linux driver 140
 supported devices 139
 supported properties and
 methods 139
 Motion Sensor
 UnifiedPOS device statistics
 properties 441
 MSR 143
 additional JavaPOS information 148
 additional OPOS information 149
 JavaPOS DirectIO calls 147
 OPOS configuration 148
 UnifiedPOS device statistics
 properties 441
 msrID 147

N

Norwegian keyboard layout 366
 notices 465
 Toshiba 465
 NumberOfRows 122

O

OEM Scanner support 281
 OPOS 44
 configuration options 44
 configuration utility 23
 configuring devices 22
 FISCAL_INFORMATION
 subcommand 85
 FISCAL_NOTIFY subcommand 86
 FISCAL_READ subcommand 85
 FISCAL_WRITE subcommand 86
 network device communication 22
 OPOS configuration 111, 162, 218, 284,
 296, 299
 cash drawer 57
 hard totals 101
 keylock 110
 line display 126
 OPOS configuration file update 164
 OPOS Firmware Update 163
 OPOS problems 36
 OptimizedBaudRate 193

P

page mode printing 238
 page mode support 209

PDF417AspectHeight 192
 PDF417AspectWidth 192
 PDF417ECCLevel 192
 PDF417Truncation 192
 PLU Keyboard and Display III 344
 point-of-sale keyboard layouts 303
 portName 42
 POS Control Center utility 21
 POS keyboard 151
 click 154
 ExtendedKeyMapping 155
 FatFingerTimeout 154
 JavaPOS configuration 154
 KbdScanning 155
 KeyTranslationFile 155
 supported devices 151
 supported properties and
 methods 152
 Typematic 155
 TypematicDelay 156
 POS printer 171
 POS printer statistics 171
 POS terminals supported 3
 position count 111
 POSKeyboard
 UnifiedPOS device statistics
 properties 444
 POSPrinter
 UnifiedPOS device statistics
 properties 446
 printer performance, improving 255
 printer-attached cash drawer 55
 problem determination
 logging 33
 properties, methods, events
 check scanner 59
 POS printer 171
 Scale 291
 Scanner 259
 ToneIndicator 297
 property definitions 123
 proportional font support 253
 ProportionalFontFixedWidth 196
 ProprietaryBusSubType 43
 pulse width 55

R

remove devices 30
 requirements
 hardware 3
 POS terminals 3
 software 4
 requirements for system 3
 response file 6
 rotated printing 252

S

safety information ix
 Scale
 UnifiedPOS device statistics
 properties 449
 Scale statistics 291
 scan codes 303

Scanner
 UnifiedPOS device statistics

 properties 449
 Scanner configuration 261
 Scanner statistics 259
 Scanner support 281
 service object settings
 hard totals 101
 Service Object settings
 cash drawer 57
 POS keyboard, RS-485/USB 162
 POS printer, SureOne devices 218
 Scale, RS-485/USB 296
 Tone indicator (keyboard tone),
 RS-485/USB 299
 Unsupported settings,
 RS-485/USB 219
 setBarCodes1 264
 setBarCodes2 265
 setBarCodes3 265
 setBarCodes4 265
 setBeeperDuration 266
 setBeeperFrequency 266
 setBeeperVolume 266
 setCheckModulo 267
 setCode128ScansPerRead 267
 setCode39ScansPerRead 267
 SetCompatibilityMode 197
 setDecodeAlgorithm 267
 setDoubleReadTimeOut 268
 setDTouchMode 268
 setEAN13ScansPerRead 268
 setEAN8ScansPerRead 269
 setEnable_2_DigitSupplements 269
 setEnable_5_DigitSupplements 269
 setEnableCodabar 269
 setEnableCode128 270
 setEnableCode128Supplements 270
 setEnableCODE39 270
 setEnableCode39CheckDigit 270
 setEnableCode93 270
 setEnableCompositeCC_A 271
 setEnableCompositeCC_B 271
 setEnableCompositeCC_C 271
 setEnableEAN_JAN_TwoLabelDecoding 270
 setEnableGoodReadBeep 271
 setEnableGS1Databar_Expanded 273
 setEnableInterleaved2of5 271
 setEnableITFCheckDigit 272
 setEnableMicroPDF417 272
 setEnablePDF417 272
 setEnableRSS_Expanded 272, 273
 setEnableRSS14 272
 setEnableStandard2of5 273
 setEnableUCC_EAN128 273
 setEnableUPC_A_CheckDigit 273
 setEnableUPC_A_To_EAN13Expansion 273
 setEnableUPC_E_CheckDigit 274
 setEnableUPC_E_To_EAN13Expansion 274
 setEnableUPC_E_To_UPC_AExpansion 274
 setEnableUPCAE_EANJAN813 274
 setEnableUPCD1D5 274
 setEnableVerificationUPC_A_EAN13_fiveDigit 275
 setEnableVerificationUPC_A_EAN13_fourDigit 275
 setITFLength1 275
 setITFLength2 275
 setITFLengths 276

setITFLengthSpecifiedTwo 276
setITFScamsPerRead 276
setLaserTimeOut 276
setLED_GoodRead_Duration 277
setMotorTimeOut 277
setSecurityLevelForInStore 278
setSTFLength1 278
setSTFLength2 278
setSTFLengths 278
setStoreScansPerRead 279
setSupplements 279
setSupplementsSecurityLevel 279
setTransmitScannerPrefixAndSuffix 279
setUPCAsScansPerRead 280
setUPCDScamsPerRead 280
setUPCEScamsPerRead 280
signalsReversed 55
silent installation, Windows 6
SIO scan code sets
 50-Key Modifiable Layout
 Keyboard 308
 50-Key Modifiable Layout Keyboard and Operator Display 308
 Modifiable Layout Keyboard with Card Reader 312
 Point of Sale Keyboard VI 308
 Retail Alphanumeric Point of Sale Keyboard with Card Reader 318
 Retail Point of Sale Keyboard 308
 Retail Point of Sale Keyboard with Card Reader and Display 308
sioDeviceNumber 43
sioPortNumber 43
sioSlotNumber 43
SLE11 SP3 9
SlpLineSpacing 197
software requirements 4
Spanish keyboard layout
 Alphanumeric Point of Sale, country dependent 362
 Retail Alphanumeric Point of Sale, country dependent 367
statistics properties, UnifiedPOS device 431
storage string remarks 235
support website 37
supported devices 99, 171, 259, 291, 297
 cash drawer 51
supported POS terminals 3
supported properties and methods 171, 259, 291, 297
SureOne keyboard (built-in) 354
SureOne MSR 148
SureOne/SurePOS 100 keyboard 165
SureOneVersion 197
SUSE Linux Enterprise 11 SP3 9
Swedish keyboard layout 367
symbolic links 12
symbology, scanner 261
system requirements 3
Systems Management
 installation 14
 validating 15

T
testing connectivity 27

Tone Indicator
 UnifiedPOS device statistics
 properties 451
ToneIndicator statistics 297
Toshiba support 37
touch support 21
tracing
 EIA-232 and USB 33
tracing, OPOS 36
TranslateCharacter 197
two-position keylock
 NANPOS 110
TypematicRate 156

U
U.K. English keyboard layout
 Alphanumeric Point of Sale, country dependent 363
 Retail Alphanumeric Point of Sale, country dependent 368
U.S. English keyboard layout
 Alphanumeric Point of Sale, country dependent 363
 CANPOS, country dependent 370, 372
 Retail Alphanumeric Point of Sale, country dependent 368
UnifiedPOS device statistics
 properties 431
uninstallation 7
 4840 systems 7
 Windows 7 7
upgrade 8
USB
 checkout keyboard scan code set 308
 retail point-of-sale keyboard scan codes 311
USB device access 21
 Linux 21
 Microsoft Windows 21
USB devices, ignoring 46
USB firmware update files for JPOS 43
USB firmware update files for OPOS 45
USB system attached keyboard 48
USB System attached keyboard, Linux 161
user-defined fonts 253

V
VibrationSensitivity 295
Virtual key codes 160

W
Web site xi
WeighMode 295
Windows installation 5

TOSHIBA
Leading Innovation >>>

Printed in USA

GC30-4078-22

