Epson ePOS SDK for JavaScript ユーザーズマニュアル

機能

使い方

API リファレンス

機器仕様

サンプルプログラム

活用ガイド

付録

M00099718
Rev.S

ご注意

- 本書の内容の一部または全部を無断で転載、複写、複製、改ざんすることは固くお断りします。
- 本書の内容については、予告なしに変更することがあります。最新の情報はお問い合わせください。
- 本書の内容については、万全を期して作成いたしましたが、万一ご不審な点や誤り、記載もれなど、お気づきの点がありましたらご連絡ください。
- 運用した結果の影響については、上項にかかわらず責任を負いかねますのでご了承ください。
- 本製品がお客様により不適切に使用されたり、本書の内容に従わずに取り扱われたり、またはエプソンおよびエプソン指定の者以外の第三者により修理・変更されたことなどに起因して生じた損害などにつきましては、責任を負いかねますのでご了承ください。
- エプソン純正品およびエプソン品質認定品以外のオプションまたは消耗品を装着してトラブルが発生した場合には、責任を負いかねますのでご了承ください。

商標について

EPSON、EXCEED YOUR VISION、ESC/POS は、セイコーエプソン株式会社の登録商標です。

Windows®、Internet Explorer® は、米国 Microsoft Corporation の米国およびその他の国における商標または登録商標です。

Safari ™ は、米国および他の国々で登録された Apple Inc. の商標です。

Android ™、Google Chrome ™ は、Google LLC の商標です。

Mozilla®、Firefox® は、米国 Mozilla Foundation の米国およびその他の国における商標または登録商標です。

IOS® は、Cisco の米国およびその他の国における商標または登録商標です。

銀聯、UnionPay は、CHINA UNIONPAY CO., Ltd. の登録商標です。

楽天 Edy は、楽天 Edy 株式会社の登録商標です。

iD は株式会社 NTT ドコモの登録商標です。

nanaco は、株式会社セブン・カードサービスの登録商標です。

QUICPay は、株式会社ジェーシービーの登録商標です。

Suica は、東日本旅客鉄道株式会社の登録商標です。

WAON は、イオン株式会社の登録商標です。

ＰｉＴａＰａ は、株式会社スルッと ＫＡＮＳＡＩ の登録商標です。

QR コードは株式会社デンソーウェーブの登録商標です。

その他の製品名および会社名は、各社の商標または登録商標です。

ESC/POS® コマンドシステム

エプソンは、独自の POS プリンターコマンドシステム、ESC/POS により、業界のイニシアチブをとってきました。ESC/POS は特許取得済みのものを含む数多くの独自のコマンドを持ち、高い拡張性で多才な POS システムの構築を実現します。ほとんどのエプソン POS プリンターとディスプレイに互換性を持つ他、この独自の制御システムにはフレキシビリティーもあるため、将来アップグレードが行いやすくなります。その機能と利便性は世界中で評価されています。

© Seiko Epson Corporation 2016-2021.

安全のために

記号の意味

本書では以下の記号が使われています。それぞれの記号の意味をよく理解してから製品を取り扱ってください。

⚠	ご使用上、必ずお守りいただきたいことを記載しています。この表示を無視して誤った取り扱いをすると、製品の故障や動作不良の原因になる可能性があります。
✎	補足説明や知っておいていただきたいことを記載しています。

使用制限

本製品を航空機・列車・船舶・自動車などの運行に直接関わる装置・防災防犯装置・各種安全装置など機能・精度などにおいて高い信頼性・安全性が必要とされる用途に使用される場合は、これらのシステム全体の信頼性および安全維持のためにフェールセーフ設計や冗長設計の措置を講じるなど、システム全体の安全設計にご配慮いただいた上で弊社製品をご使用いただくようお願いいたします。

本製品は、航空宇宙機器、幹線通信機器、原子力制御機器、医療機器など、きわめて高い信頼性・安全性が必要とされる用途への使用を意図しておりませんので、これらの用途には本製品の適合性をお客様において十分ご確認の上、ご判断ください。

本書について

本書の目的

Epson ePOS SDK for JavaScript を利用したアプリケーション開発に必要な情報を提供します。

本書の構成

本書は次のように構成されています。

第 1 章　機能

第 2 章　使い方

第 3 章　API リファレンス

第 4 章　機器仕様

第 5 章　サンプルプログラム

第 6 章　活用ガイド

付録　オープンソースソフトウェアライセンス

もくじ

■安全のために 3
記号の意味 3
■使用制限 3
■本書について 3
本書の目的 3
本書の構成 3
■もくじ 4

機能 12

■Epson ePOS SDK for JavaScript でのアプリケーション開発 12
TM プリンターモデル 13
カスタマーディスプレイモデル 14
TM プリンター +DM-D+ バーコードスキャナーモデル 15
POS ターミナルモデル 16
■アプリケーションソフトの動作環境 17
対象プリンター 17
プリンターごとの提供機能 18
サポート周辺機器 19
ePOS-Print 設定 21
■ダウンロードコンテンツ 23
■制限事項 24

使い方 25

■プリンターとの接続 25
■Epson ePOS SDK for JavaScript の組み込み方法 25
■プログラミングガイド 25
プログラミングフロー 25
設定用の命令バッファーの有効範囲 32
エラーコード 32

API リファレンス 33

■ePOS SDK API 一覧 33
ePOSDevice オブジェクト 33
デバイスオブジェクト共通 34
OtherPeripheral オブジェクト 34
CashChanger オブジェクト 34
CAT オブジェクト 36
Display オブジェクト 36
Keyboard オブジェクト 39
POSKeyboard オブジェクト 39

MSR オブジェクト 39
Printer オブジェクト 39
Scanner オブジェクト 43
SimpleSerial オブジェクト 43
DeviceHubTerminal オブジェクト 44
CommBoxManager オブジェクト 44
CommBox オブジェクト 44
ePOSDeviceConfiguration オブジェクト 44

■ ePOSDevice オブジェクト 45

Constructor 45
connect メソッド 46
disconnect メソッド 48
isConnected メソッド 49
createDevice メソッド 50
deleteDevice メソッド 53
getAdmin メソッド 54
getLocation メソッド 55
sendOfscXml メソッド 56
getCommBoxManager メソッド 57
onreconnecting イベント 58
onreconnect イベント 59
ondisconnect イベント 60

■ デバイスオブジェクト共通 61

callEvent メソッド 61

■ OtherPeripheral オブジェクト 62

send メソッド 62
onreceive イベント 63

■ CashChanger オブジェクト（デバイス制御プログラム） 64

readCashCounts メソッド 64
beginDeposit メソッド 65
pauseDeposit メソッド 66
restartDeposit メソッド 67
endDeposit メソッド 68
dispenseChange メソッド 69
dispenseCash メソッド 70
sendCommand メソッド 71
oncashcount イベント 72
ondeposit イベント 73
ondispense イベント 75
oncommandreply イベント 76
ondirectio イベント 77
onstatusupdate イベント 78

■ CashChanger オブジェクト（デバイス制御スクリプト） 80

setConfig メソッド 80
readCashCounts メソッド 81
beginDeposit メソッド 82
pauseDeposit メソッド 83
restartDeposit メソッド 84
endDeposit メソッド 85
dispenseChange メソッド 86
dispenseCash メソッド 87
collectCash メソッド 88
openDrawer メソッド 89
sendCommand メソッド 90

onconfigchange イベント 91
oncashcount イベント 92
ondeposit イベント 94
ondispense イベント 96
oncollect イベント 97
oncommandreply イベント 98
onstatuschange イベント 99

■CAT オブジェクト 101

timeout プロパティー 101
trainingMode プロパティー 102
authorizeSales メソッド 103
authorizeVoid メソッド 104
authorizeRefund メソッド 105
authorizeCompletion メソッド 106
accessDailyLog メソッド 107
sendCommand メソッド 109
checkConnection メソッド 110
clearOutput メソッド 111
onauthorizesales イベント 112
onauthorizevoid イベント 114
onauthorizerefund イベント 116
onauthorizecompletion イベント 118
onaccessdailylog イベント 120
oncheckConnection イベント 122
onclearOutput イベント 123
oncommandreply イベント 124
ondirectio イベント 126
onstatusupdate イベント 127

■Display オブジェクト 128

createWindow メソッド 128
destroyWindow メソッド 130
setCurrentWindow メソッド 131
clearWindow メソッド 132
addCreateScreen メソッド 133
addCreateScreenCustom メソッド 135
createTextArea メソッド 137
destroyTextArea メソッド 139
setCurrentTextArea メソッド 140
clearTextArea メソッド 141
setCursorPosition メソッド 142
moveCursorPosition メソッド 143
setCursorType メソッド 144
addText メソッド 145
addReverseText メソッド 148
addMarquee メソッド 149
setBlink メソッド 151
setBrightness メソッド 152
addBackgroundColor メソッド 153
addStartSlideShow メソッド 155
addStopSlideShow メソッド 156
addDownloadImage メソッド 157
addRegisterDownloadImage メソッド 159
addNVImage メソッド 161
addClearImage メソッド 163
addSymbol メソッド 164
addClearSymbol メソッド 167

showClock メソッド 168
addCommand メソッド 169
send メソッド 170
reset メソッド 171
onreceive イベント 172

■ Keyboard オブジェクト 174

setPrefix メソッド 174
onkeypress イベント 175
onstring イベント 176
setMSRPrefix メソッド 177
ondata イベント 178

■ POSKeyboard オブジェクト 179

onkeypress イベント 179

■ MSR オブジェクト 180

ondata イベント 180

■ Printer オブジェクト 181

addTextAlign メソッド 181
addTextLineSpace メソッド 182
addTextRotate メソッド 183
addText メソッド 184
addTextLang メソッド 186
addTextFont メソッド 188
addTextSmooth メソッド 189
addTextDouble メソッド 190
addTextSize メソッド 191
addTextStyle メソッド 192
addTextPosition メソッド 194
addTextVPosition メソッド 195
addFeedUnit メソッド 196
addFeedLine メソッド 197
addFeedPosition メソッド 198
addFeed メソッド 199
addImage メソッド 200
addLogo メソッド 202
addBarcode メソッド 203
addSymbol メソッド 207
addHLine メソッド 212
addVLineBegin メソッド 214
addVLineEnd メソッド 215
addPageBegin メソッド 216
addPageEnd メソッド 217
addPageArea メソッド 218
addPageDirection メソッド 220
addPagePosition メソッド 221
addPageLine メソッド 223
addPageRectangle メソッド 225
addRotateBegin メソッド 227
addRotateEnd メソッド 229
addCut メソッド 230
addPulse メソッド 231
addSound メソッド 233
addLayout メソッド 235
recover メソッド 239
addRecovery メソッド 240

reset メソッド 241
addReset メソッド 242
addCommand メソッド 243
send メソッド 244
print メソッド 245
getPrintJobStatus メソッド 247
startMonitor メソッド 248
stopMonitor メソッド 249
halftone プロパティー 250
brightness プロパティー 251
force プロパティー 252
timeout プロパティー 253
interval プロパティー 254
drawerOpenLevel プロパティー 255
message プロパティー 256
onreceive イベント 257
onstatuschange イベント 260
onbatterystatuschange イベント 261
ononline イベント 262
onoffline イベント 263
onpoweroff イベント 264
oncoverok イベント 265
oncoveropen イベント 266
onpaperok イベント 267
onpapernearend イベント 268
onpaperend イベント 269
ondrawerclosed イベント 270
ondraweropen イベント 271
onbatteryok イベント 272
onbatterylow イベント 273

■ Scanner オブジェクト 274

ondata イベント 274

■ SimpleSerial オブジェクト 275

sendCommand 275
oncommandreply イベント 276

■ DeviceHubTerminal オブジェクト 277

shutdown メソッド 277
restart メソッド 279

■ CommBoxManager オブジェクト 281

openCommBox メソッド 281
closeCommBox メソッド 283

■ CommBox オブジェクト 284

getCommHistory メソッド 284
send メソッド 286
onreceive イベント 287

■ ePOSDeviceConfiguration オブジェクト 288

Constructor 288
getRegisterdDevices メソッド 289

■ Error Code 一覧 291

callback パラメーターで取得するエラーコードと対処方法 291
onreceive イベントで取得するエラーコードと対処方法 293

■ Key Code 一覧 297

機器仕様 299

■ サポートオブジェクト一覧 299
■ サポート API 一覧 301
ePOSDevice オブジェクト 301
Display オブジェクト 304
Printer オブジェクト 310
■ プリンター別サポート情報 320
TM-m10 320
TM-m30 325
TM-m30II 330
TM-m30II-H 335
TM-m30II-S 340
TM-m30II-SL 345
TM-T20II 350
TM-T20III 354
TM-T88VI 359
TM-T88VII 364
TM-L100 369
TM-T70II 375
TM-T88V 379
TM-T90II 382
TM-T90KP (UB-R04) 386
TM-T90KP (UB-E04) 389
TM-P20 392
TM-P60II (Peeler) 398
TM-P80 404
TM-T70-i 410
TM-T88V-i 413
TM-T70II-DT 418
TM-T70II-DT2 423
TM-T88V-DT 428
TM-T88VI-DT2 433
TM-T70 438
TM-T88IV 441
TM-T90 444
TM-L90 447
■ ファームウェアバージョンによる使用制限 450
TM-i シリーズ /TM-DT シリーズ 450
TM-m30 455
TM-m30II-H/TM-m30II-S/TM-m30II-SL 455
TM-T88VI 456
■ ePOS-Print 設定 457

サンプルプログラム 458

■ 機能 458
POS Terminal 458
Entry Terminal 459
Receipt Designer 459
Printer Sample 460

OtherPeripheral Sample....460
CashChanger Sample (Device Control Program)....461
CashChanger Sample (Device Control Script)....461
CAT Sample....462
Customer Display Sample....462
Keyboard Sample....463
POSKeyboard Sample....463
MSR Sample....464
Barcode Scanner Sample....464

■ 使用環境....465

■ サンプルプログラムの起動手順....465

プリンターの環境設定....465
サンプルプログラムの起動....470

■ サンプルプログラムの使い方....471

POS Terminal Sample....471
Entry Terminal....477
Receipt Designer....480
Printer Sample....486
Customer Display Sample....487
Keyboard Sample....489
MSR Sample....490
CashChanger Sample (Device Control Script)....491
CashChanger Sample (Device Control Program)....493
Barcode Scanner Sample....495
CAT Sample....496
POSKeyboard Sample....499
OtherPeripheral Sample....500

活用ガイド....501

■ 常時監視するには....501

■ ネットワークを自動で再接続するには....502

■ アプリケーション間でデータを送受信するには....504

コミュニケーションボックスを使用したアプリケーション例....504
コミュニケーションボックスの仕組み....505
コミュニケーションボックスの仕様....505
コミュニケーションボックスの使用したデータ処理例....506
基本的なプログラミングシーケンス....507

■ 迂回印刷するには....509

■ Web ページを頻繁に更新 / 遷移するアプリケーションを開発するには....511

ePOS-Print オプションを有効にするには....511
制限事項....511

■ 自動つり銭機を制御するには....512

デバイス制御プログラム....512
デバイス制御スクリプト....513

■ 信用決済端末を制御するには....514

付録 515

■オープンソースソフトウェアライセンス 515

機能

Epson ePOS SDK for JavaScript は、Web アプリケーションソフト開発用の SDK です。Epson ePOS SDK for JavaScript を使用すると、シンプルなコードでプリンターを制御できます。また、一部の TM プリンターでは、本体に接続したカスタマーディスプレイやバーコードスキャナーなどの周辺機器、ネットワーク接続した子機プリンターを制御できます。さらに、スプーラー機能やコミュニケーションボックスなどの独自機能を使用したアプリケーションソフトも開発できます。

本書ではプリンターを以下のように総称します。

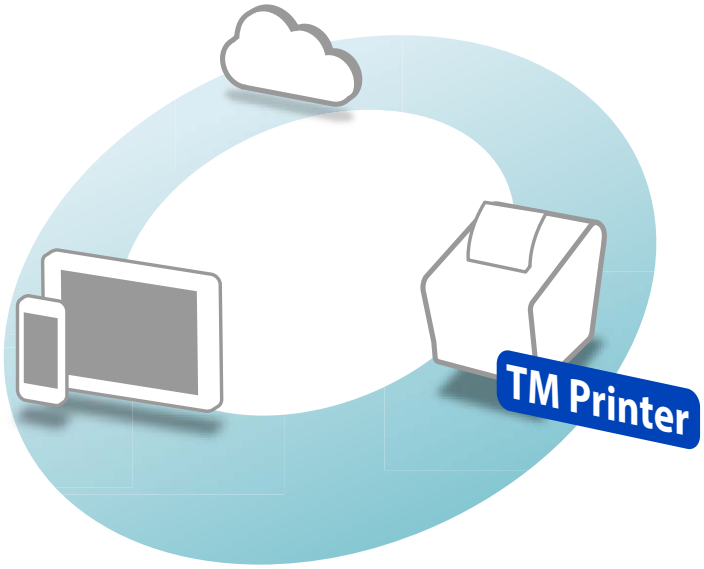
総称	説明
TM プリンター	以下のレシートプリンターの総称 • 単機能モデル • モバイルモデル • TM-i シリーズ • TM-DT シリーズ
子機プリンター	ネットワーク接続した TM プリンターで、スマートデバイスから親機となる TM プリンターを経由して制御する TM プリンターの総称。

Epson ePOS SDK for JavaScript でのアプリケーション開発

Epson ePOS SDK for JavaScript で開発できるアプリケーションソフトのシステム構成は、以下の通りです。システムの構築方法や使用制限などの詳細は、各プリンターの詳細取扱説明書を参照してください。

TM プリンターモデル

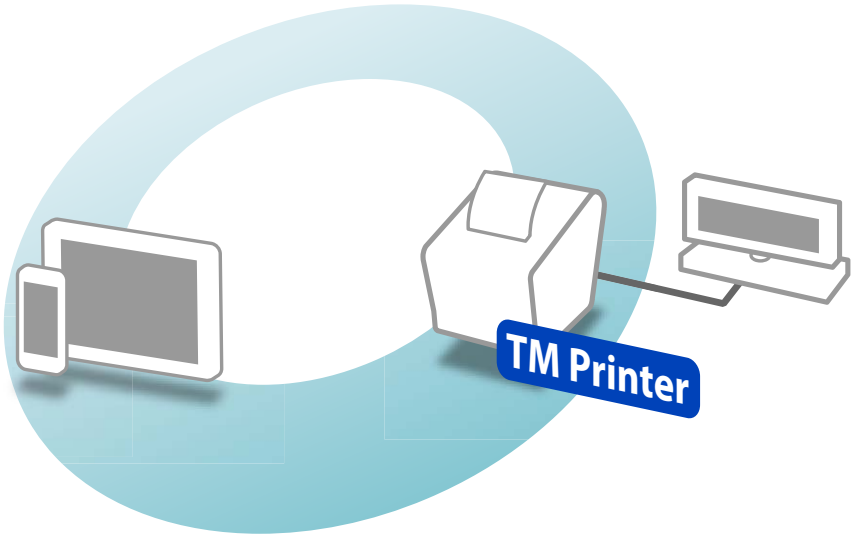
スマートデバイスと TM プリンターをネットワーク（有線 LAN、無線 LAN）で接続するシステムです。
スマートデバイス上の HTML5 と互換性のある Web ブラウザーから、TM プリンターを制御することができます。



- EpsonNet Config を用いて ePOS-Print 設定を「有効」にする必要があります。
  詳細は、ePOS-Print 設定を参照してください。
- connect メソッド実行時に、options パラメーターの eposprint プロパティーを有効にしてください。

カスタマーディスプレイモデル

スマートデバイスから TM プリンターとカスタマーディスプレイを制御するシステムです。
TM プリンターにカスタマーディスプレイを接続し、スマートデバイスと TM プリンターを有線 LAN/ 無線 LAN のネットワーク通信で接続して使用します。



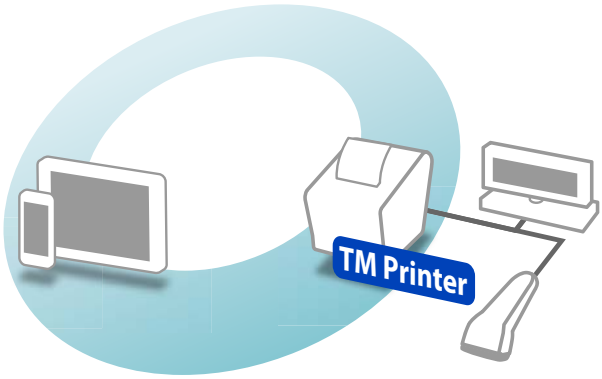
- connect メソッド実行時に、options パラメーターの eposprint プロパティーを有効にしてください。
- DM-D30 を DM-D70 に置き換えて使用する場合、DM-D70 の ePOS-Device 設定を「DM-D30」に変更する必要があります。
  DM-D70 の ePOS-Device 設定を変更するには、DM-D70 Utility を使用してください。

TM プリンター+DM-D+ バーコードスキャナーモデル

スマートデバイスから TM プリンターと DM-D（カスタマーディスプレイ）、バーコードスキャナーを制御するシステムです。TM プリンターに DM-D とバーコードスキャナーを接続し、スマートデバイスと TM プリンターを有線 LAN/ 無線 LAN のネットワーク通信で接続して使用します。

TM プリンター +DM-D+ バーコードスキャナーモデルは、以下の TM プリンターと DM-D の組み合わせで構成できます。

	DM-D30	DM-D70
TM-m30	✔	-
TM-m30II	✔	-
TM-m30II-H	✔	✔
TM-m30II-S	✔	✔
TM-m30II-SL	✔	✔
TM-T88VI	✔	-
TM-T88VII	✔	✔
TM-L100	✔	✔



- DM-D30 を使用する場合、バーコードスキャナーを DM-D30 に接続して使用できます。
- DM-D やバーコードスキャナーの接続方法は、各 TM プリンターの詳細取扱説明書を参照してください。
- TM プリンターに DM-D を接続せず、バーコードスキャナーのみを接続して使用することもできます。
- connect メソッド実行時に、options パラメーターの eposprint プロパティーを無効にしてください。

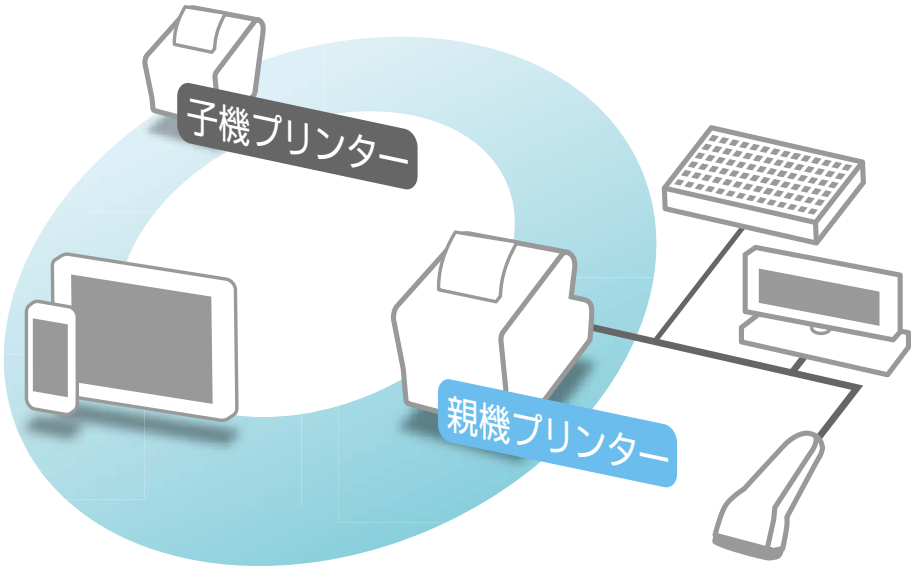
TM プリンターの ePOS-Device 設定を「有効」にする必要があり、各機種専用のユーティリティソフトウェアで設定できます。

TM プリンター	ソフトウェア
TM-m30	EpsonNet Config（Web バージョン）
TM-T88VI	TM-T88VI Utility
TM-T88VII	TM-T88VII Utility
TM-L100	TM-L100 Utility

以下の TM プリンターは、初期設定で ePOS-Device 設定が有効になっています。
TM-m30II / TM-m30II-H / TM-m30II-S / TM-m30II-SL

POS ターミナルモデル

スマートデバイスと POS ターミナルとなる TM プリンター（親機）を有線 LAN/ 無線 LAN のネットワーク通信で接続し、親機に子機プリンターやカスタマーディスプレイなど、複数の周辺機器を接続して運用するシステムです。



POS ターミナルモデルはエプソン製の周辺機器の他に、エプソンが独自に開発したデバイス制御プログラムやデバイス制御スクリプトから、様々な周辺機器を制御できます。

POS ターミナルとなる TM プリンターによって搭載するソフトウェアやそのバージョンが異なるため、制御できる周辺機器に違いがあります。

制御できる周辺機器については、各プリンターの詳細取扱説明書を参照してください。

デバイス制御プログラムは、TM-DT シリーズ（TM-DT ソフトウェア Ver.4.0 以降）で実装されています。

デバイス制御プログラムについては、TM-DT シリーズ周辺機器制御ガイドを参照してください。

Epson ePOS SDK for JavaScript で開発したアプリケーションを POS ターミナルモデルで利用するには、POS ターミナルとなる TM プリンターの ePOS-Device Service を「有効」に設定する必要があります。

> TM-i シリーズと TM-DT シリーズの ePOS-Device Service は、初期設定で「有効」に設定されています。

アプリケーションソフトの動作環境

Epson ePOS SDK for JavaScript を使用して開発するアプリケーションソフトの動作環境は、以下の通りです。
対象となる Web ブラウザーの最新情報は、Epson ePOS SDK for JavaScript パッケージの README.jp.txt を参照してください。

対象プリンター

アプリケーションソフトから制御できるプリンターは、以下の通りです。

> 単機能モデルとモバイルモデルは、ePOS-Print 設定を有効にする必要があります。
> 詳細は、ePOS-Print 設定を参照してください。

単機能モデル

❏ 以下の Wi-Fi/Ethernet モデル

- TM-m10
- TM-m30
- TM-m30II
- TM-m30II-H
- TM-m30II-S
- TM-m30II-SL
- TM-T20II
- TM-T20III
- TM-T88VI
- TM-T88VII
- TM-L100

❏ UB-E04 もしくは UB-R04 を搭載した以下のプリンター

> プリンターに搭載されているインターフェイスは、ステータスシート印刷から確認できます。
> 各プリンターの詳細取扱説明書を参照してください。

- TM-T70II
- TM-T88V
- TM-T90II
- TM-T90KP

モバイルモデル

❏ 以下の Wi-Fi/Ethernet モデル

- TM-P20
- TM-P60II
- TM-P80

TM-i シリーズ

❏ TM-T70-i（TM-i ファームウェア Ver.4.0 以降）

❏ TM-T88V-i（TM-i ファームウェア Ver.4.0 以降）

TM-DT シリーズ

❏ TM-T70II-DT

❏ TM-T70II-DT2

❏ TM-T88V-DT

❏ TM-T88VI-DT2

プリンターごとの提供機能

Epson ePOS SDK for JavaScript が提供する機能は、プリンターによって使用できる機能が異なります。
以下の一覧表を参照してください。

	SSL/TLS 通信 *1	内蔵ブザーの鳴動制御	外付けオプションブザーの鳴動制御	子機プリンターの制御	スプーラー / 迂回印刷	コミュニケーションボックス	周辺機器の制御 *2
TM-m10	✔	-	✔	-	-	-	-
TM-m30	✔	-	✔	-	-	-	✔
TM-m30II	✔	-	✔	-	-	-	✔
TM-m30II-H	✔	-	✔	✔	✔	-	✔
TM-m30II-S	✔	-	✔	✔	✔	-	✔
TM-m30II-SL	✔	-	✔	✔	✔	-	✔
TM-T88VI	✔*3	✔*4	✔	✔	✔	-	✔
TM-T88VII	✔*3	✔*4	✔	✔	✔	-	✔
TM-L100	✔	✔*4	✔	-	-	-	✔
TM-T20II	✔	-	✔	-	-	-	-
TM-T20III	✔	-	✔	-	-	-	-
TM-T70II	-	✔*4	✔	-	-	-	-
TM-T88V	-	✔*4	✔	-	-	-	-
TM-T90II	-	-	✔	-	-	-	-
TM-T90KP	-	-	✔	-	-	-	-
TM-P20	-	✔	-	-	-	-	-
TM-P60II	-	✔	-	-	-	-	-
TM-P80	-	✔	-	-	-	-	-
TM-T70-i	✔	-	-	✔	✔	✔	✔
TM-T88V-i	✔	-	✔	✔	✔	✔	✔
TM-T70II-DT	✔	-	✔	✔	✔	✔	✔
TM-T70II-DT2	✔	-	✔	✔	✔	✔	✔
TM-T88V-DT	✔	-	✔	✔	✔	✔	✔
TM-T88VI-DT2	✔	-	✔	✔	✔	✔	✔

*1:　connect メソッド使用時に通信方式として "8043" を指定します。
*2:　使用できる周辺機器は、プリンターによって異なります。各プリンターの詳細取扱説明書を参照してください。
*3:　TM プリンター +DM-D+ バーコードスキャナーモデルのシステム構成では使用できません。
*4:　ドロアーキックコマンドでの鳴動制御に対応しています。

サポート周辺機器

Epson ePOS SDK for JavaScript では、以下の周辺機器制御をサポートしています。

- 使用できる周辺機器は、プリンターによって異なります。
  プリンターごとの提供機能、または各プリンターの詳細取扱説明書を参照してください。
- TM-DT シリーズのプリンターは、デバイス制御プログラムを開発することで、ここで紹介する以外の周辺機器も制御できます。
  開発方法については、TM-DT シリーズ周辺機器制御ガイドを参照してください。

子機プリンター

以下の TM プリンターの Wi-Fi モデル

- ❏ TM-P20
- ❏ TM-P60II
- ❏ TM-P80

Wi-Fi/Ethernet インターフェイスを搭載した以下の TM プリンター

- ❏ TM-T20II
- ❏ TM-T20III
- ❏ TM-T20X
- ❏ TM-T70
- ❏ TM-T70II
- ❏ TM-T88IV
- ❏ TM-T88V
- ❏ TM-T88VI
- ❏ TM-T90
- ❏ TM-T90II
- ❏ TM-T90KP
- ❏ TM-L90

以下の TM-i シリーズ /TM-DT シリーズ

- ❏ TM-T70-i（TM-i ファームウェア Ver.4.0 以降）
- ❏ TM-T70II-DT（TM-DT ソフトウェア Ver.4.0 以降）
- ❏ TM-T88V-i（TM-i ファームウェア Ver.4.0 以降）
- ❏ TM-T88V-DT（TM-DT ソフトウェア Ver.4.0 以降）

カスタマーディスプレイ

- ❏ DM-D30
- ❏ DM-D70
- ❏ DM-D110

- DM-D70 を使用する場合、接続するプリンターによって使用可能な表示モード（標準モード、桁数固定モード）が異なります。
  詳細は、DM-D70 詳細取扱説明書を参照してください。
- 以下のシステム構成で DM-D30 を DM-D70 に置き換える場合は、DM-D70 を桁行固定モードで使用してください。
  ＊ カスタマーディスプレイモデル（SSL/TSL 通信）
  ＊ TM プリンター +DM-D+ バーコードスキャナーモデル
  ＊ POS ターミナルモデル

デバイス制御プログラムで制御する周辺機器

TM-DT シリーズ（TM-DT ソフトウェア Ver.4.0 以降）が持つデバイス制御プログラムから、OPOS 仕様準拠の周辺機器を制御できます。

以下のカテゴリーで、OPOS CCO 1.14.001 と組み合わせて動作するドライバーを持つ周辺機器を制御できます。

❏ 自動つり銭機

❏ 信用照会端末

❏ MSR

❏ POS キーボード

❏ バーコードスキャナー

デバイス制御スクリプトで制御する周辺機器

TM-i シリーズ /TM-DT シリーズが持つデバイス制御スクリプトから、以下の周辺機器を制御できます。

❏ キー入力デバイス

- MSR（日立オムロン V3TU-FK）
- キーボード（標準 HID）
- バーコードスキャナー（標準 HID）

❏ シリアル通信デバイス

- 自動つり銭機（グローリー RT-200/RAD-200）
- シリアル通信デバイス
- シリアル通信デバイス同等の制御が可能な USB デバイス

シリアル通信デバイス同等の制御が可能な USB デバイスを使用する場合、専用のドライバーをインストールする必要があります。
ドライバーの仕様によっては、使用できない場合があります。

ePOS-Print 設定

Epson ePOS SDK for JavaScript を使用して開発したアプリケーションソフトで単機能モデルとモバイルモデルの TM プリンターを制御するには、TM プリンターの ePOS-Print 設定を有効にする必要があります。機種によって設定方法が異なります。

> 以下の TM プリンターは、初期設定で ePOS-Print 設定が有効になっています。
> TM-m10 / TM-m30 / TM-m30II / TM-m30II-H / TM-m30II-S / TM-m30II-SL / TM-T88VI / TM-T88VII / UB-E04 搭載モデル

TM-m30II/TM-m30II-H/TM-m30II-S/TM-m30II-SL/TM-T20III

各機種専用のユーティリティで設定できます。

製品	ソフトウェア
TM-m30II	TM-m30II Utility, EpsonNet Config（Web バージョン）
TM-m30II-H	TM-m30II Utility, EpsonNet Config（Web バージョン）
TM-m30II-S	TM-m30II Utility, EpsonNet Config（Web バージョン）
TM-m30II-SL	TM-m30II Utility, EpsonNet Config（Web バージョン）
TM-T20III	TM-T20III Utility
TM-L100	TM-L100 Utility

詳細は、各製品の詳細取扱説明書を参照してください。

単機能モデル / モバイルモデル

EpsonNet Config（Web バージョン）を使って設定します。

> ネットワーク経由で設定します。USB 接続では設定できません。

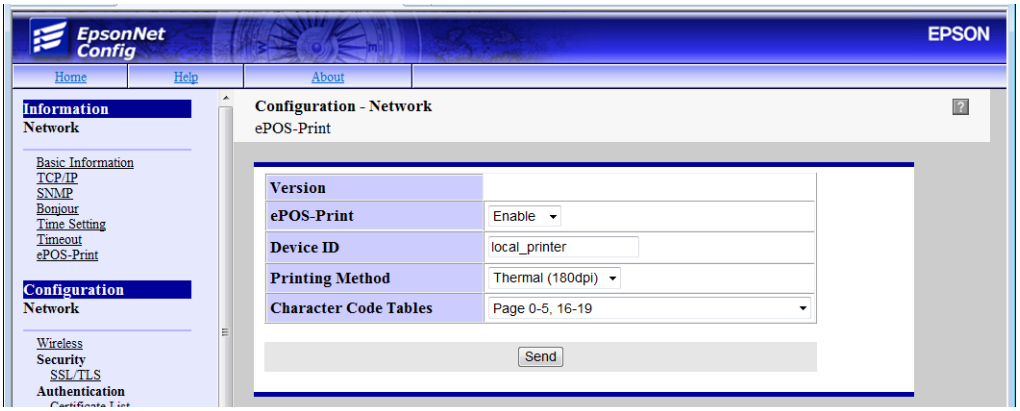
1 **TM プリンターをネットワークに接続します。**

2 **Web ブラウザーを起動し、以下の URL にアクセスします。**

http://[プリンターの IP アドレス]/

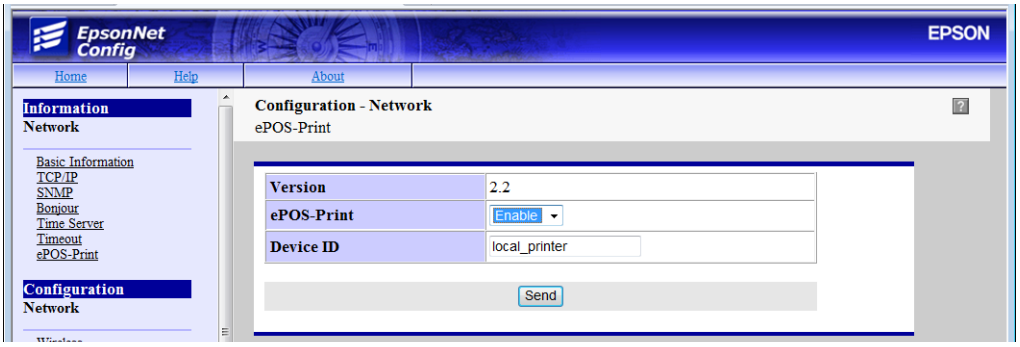
3 [ePOS-Print] をクリックし、以下項目を設定します。

❏ UB-E04 もしくは UB-R04 を搭載した TM プリンター



項目	設定
ePOS-Print	Enable（初期値：Disable）
Device ID	システムの設定に合わせる。（初期値：local_printer）
Printing Method	TM プリンターに合わせて設定する。（ePOS-Print 設定参照）
Character Code Table	TM プリンターに合わせて設定する。（ePOS-Print 設定参照）

❏ モバイルモデルの場合



項目	設定
ePOS-Print	Enable（初期値：Disable）
Device ID	システムの設定に合わせる。（初期値：local_printer）

4 [Send] をクリックします。

5 TM プリンターを再起動します。

ダウンロードコンテンツ

Epson ePOS SDK for JavaScript パッケージには、以下のファイルが含まれています。

ファイル名	説明
epos-x.x.x.js	機能実行用のライブラリーファイルです。
OPOS_CCOs_1.14.001.msi	OPOS CCO インストーラーパッケージです。
ePOS_SDK_Sample_JavaScript.zip	サンプルプログラムファイルです。
DeviceControlScript_Sample.zip	デバイス制御スクリプト用のサンプルプログラムファイルです。
DeviceControlProgram_Sample.zip	デバイス制御プログラム用のサンプルプログラムファイルです。
README.jp.txt	日本語版 README ファイルです。
README.en.txt	英語版 README ファイルです。
EULA.jp.txt	日本語版 SOFTWARE LICENSE AGREEMENT を記載しています。
EULA.en.txt	英語版 SOFTWARE LICENSE AGREEMENT を記載しています。
ePOS_SDK_JavaScript_um_ja_revx.pdf	日本語版ユーザーズマニュアルです。
ePOS_SDK_JavaScript_um_en_revx.pdf	英語版ユーザーズマニュアルです。
TM-DT_Peripherals_ja_revx.pdf	日本語版 TM-DT シリーズ周辺機器制御ガイドです。TM-DT シリーズで周辺機器を制御する方法を説明しています。
TM-DT_Peripherals_en_revx.pdf	英語版 TM-DT シリーズ周辺機器制御ガイドです。
JSON_Specification.pdf	海外向けの JSON の仕様書です。GermanyFiscalElement object で使用する JSON の仕様を説明しています。
ePOS_SDK_JavaScript_Migration_Guide_ja_revx.pdf	日本語版マイグレーションガイドです。ePOS-Print SDK、ePOS-Device SDK からの移行方法を説明しています。
ePOS_SDK_JavaScript_Migration_Guide_en_revx.pdf	英語版マイグレーションガイドです。

制限事項

- ❏ バーコードスキャナーのサフィックス（接尾語）の設定は CR（改行コード）にしてください。これ以外の設定ではデータを取得できません。
- ❏ 2 次元バーコードスキャナーを使用する場合、日本語などのマルチバイト文字は正しく取得できません。ただし、下記プリンターと 2 次元バーコードスキャナーを組み合わせて使用し、QR コードを読み取る場合は、UTF-8 でエンコードしたマルチバイト文字は正しく取得できます。
  - TM-m30
  - TM-m30II
  - TM-m30II-H
  - TM-m30II-S
  - TM-m30II-SL
  - TM-T88VI
  - TM-T88VII
  - TM-L100
- ❏ 2 次元バーコードのデータに ASCII 制御コード (0x00 ～ 0x1F) が含まれる場合、制御コードは取得できません。
- ❏ JavaScript ライブラリーの prototype.js は、Epson ePOS SDK for JavaScript のライブラリー (epos-x.x.x.js) と競合するため、一緒に使用できません。

使い方

Epson ePOS SDK for JavaScript の組み込みと、Epson ePOS SDK for JavaScript を使用した基本的なプログラミング方法を説明します。

プリンターとの接続

構築するシステム構成に合わせて、スマートデバイスとプリンターを接続します。
詳細は、各プリンターの詳細取扱説明書を参照してください。

Epson ePOS SDK for JavaScript の組み込み方法

1. epos-x.x.x.js を Web サーバーに配置します。
2. HTML の <script> タグで Web ページにスクリプトを組み込みます。

組み込み例

```
<script type="text/javascript" src="epos-2.0.0.js"></script>
```

プログラミングガイド

Epson ePOS SDK for JavaScript を使用した基本的なプログラミング方法を説明します。

プログラミングフロー

開発するアプリケーションソフトで制御するプリンターや周辺機器別に、プログラミングフローを説明します。

- プリンターを制御する
- カスタマーディスプレイを制御する
- キーボード、バーコードスキャナーを制御する

プリンターを制御する

TM プリンターや子機プリンターを制御し、レシート印刷するプログラミング方法を説明します。

1. ePOSDevice オブジェクトの作成（デバイスと接続 / 通信）

2. Printer オブジェクトを取得（プリンターの選択）

3. 印刷データの作成（データのバッファリング）

4. 印刷データの送信（印刷 / 切断）

ePOSDevice オブジェクトの作成(デバイスと接続 / 通信)

ePOSDevice オブジェクトを作成し、初期化します。

```
var ePosDev = new epson.ePOSDevice();

function connect() {
   var ipAddress = '192.168.192.168';
   var port = '8008';

   ePosDev.connect(ipAddress, port, callback_connect);
}
```

Printer オブジェクトを取得(プリンターの選択)

Printer オブジェクトを取得し、制御するプリンターを選択します。

```
function callback_connect(resultConnect){
   var deviceId = 'local_printer';
   var options = {'crypto' : false, 'buffer' : false};

   if ((resultConnect == 'OK') || (resultConnect == 'SSL_CONNECT_OK')) {
      //Printer オブジェクトを取得する
      ePosDev.createDevice(deviceId, ePosDev.DEVICE_TYPE_PRINTER, options,
callback_createDevice);
   }
   else {
      // エラーメッセージ表示
   }
}

var printer = null;

function callback_createDevice(deviceObj, errorCode){
   if (deviceObj === null) {
      // Printer オブジェクト取得失敗、エラーメッセージ表示
      return;
   }
   printer = deviceObj;
```

```
    // 印刷完了イベントを登録
    printer.onreceive = function(response){
        if (response.success) {
            // 印刷成功メッセージ表示
        }
        else {
            // エラーメッセージ表示
        }
    };
}
```

印刷データの作成(データのバッファリング)

Printer オブジェクトの addXXX 系の API を使用して、印刷データを作成します。

以下の例では、中央揃えで “Hello World” と印刷するデータを作成しています。

```
function createData(){
    printer.addTextAlign(printer.ALIGN_CENTER);
    printer.addText('Hello World\n');
}
```

印刷データの送信(印刷 / 切断)

印刷処理を行い、印刷結果を取得します。

```
function send(){
    if (ePosDev.isConnected) {
        printer.send();
    }
}
```

エラー発生時、作成した印刷データは命令バッファーをクリアするまで消えません。
エラー処理終了後などに、その印刷データを使って再印刷できます。

Printer オブジェクトを破棄し、プリンターと切断します。

```
//Printer オブジェクトを破棄する
ePosDev.deleteDevice(printer, callback_deleteDevice);

function callback_deleteDevice(errorCode){
    // デバイスと切断
    ePosDev.disconnect();
}
```

カスタマーディスプレイを制御する

カスタマーディスプレイを制御し、ディスプレイにテキストを表示するプログラミング方法を説明します。

1. ePOSDevice オブジェクトの作成（デバイスと接続 / 通信）

↓

2. Display オブジェクトを取得（デバイスの選択）

↓

3. 表示データの作成（データのバッファリング）

↓

4. 表示データの送信（表示 / 切断）

ePOSDevice オブジェクトの作成(デバイスと接続 / 通信)

ePOSDevice オブジェクトを作成し、初期化します。

```
var ePosDev = new epson.ePOSDevice();

function connect() {
   var ipAddress = '192.168.192.168';
   var port = '8008';

   ePosDev.connect(ipAddress, port, callback_connect);
}
```

Display オブジェクトを取得(デバイスの選択)

Display オブジェクトを取得し、制御するカスタマーディスプレイを選択します。

```
function callback_connect(resultConnect){
   var deviceId = 'local_display';
   var options = {'crypto' : false, 'buffer' : false};

   if ((resultConnect == 'OK') || (resultConnect == 'SSL_CONNECT_OK')) {
      //Display オブジェクトを取得する
      ePosDev.createDevice(deviceId, ePosDev.DEVICE_TYPE_DISPLAY, options,
callback_createDevice);
   }
   else {
      // エラーメッセージ表示
   }
}

var display = null;

function callback_createDevice(deviceObj, errorCode){
   if (deviceObj === null) {
      // Display オブジェクト取得失敗、エラーメッセージ表示
      return;
   }
   display = deviceObj;
```

```
    // ディスプレイ表示完了イベントを登録
    display.onreceive = function(response){
        if (response.success) {
            // ディスプレイ表示成功メッセージ表示
        }
        else {
            // エラーメッセージ表示
        }
    };
}
```

表示データの作成(データのバッファリング)

Display オブジェクトの addXXX 系の API を使用して、表示データを作成します。

以下の例では、“Hello World” と表示するデータを作成しています。

```
function createData(){
    diaplay.addText('Hello World');
}
```

表示データの送信(表示 / 切断)

ディスプレイ表示処理を行い、表示結果を取得します。

```
function send(){
    if (ePosDev.isConnected) {
        display.send();
    }
}
```

Display オブジェクトを破棄し、カスタマーディスプレイと切断します。

```
//Display オブジェクトを破棄する
ePosDev.deleteDevice(display, callback_deleteDevice);

function callback_deleteDevice(errorCode){
    // デバイスと切断
    ePosDev.disconnect();
}
```

キーボード、バーコードスキャナーを制御する

キーボードやバーコードスキャナーを制御し、入力データを受信するプログラミング方法を説明します。

以下の説明では、キーボードを制御する場合のプログラミングフローを説明します。

1. ePOSDevice オブジェクトの作成（デバイスと接続 / 通信）

↓

2. Keyboard オブジェクトを取得（デバイスの選択）

↓

3. デバイスとの切断

ePOSDevice オブジェクトの作成（デバイスと接続 / 通信）

ePOSDevice オブジェクトを作成し、初期化します。

```
var ePosDev = new epson.ePOSDevice();

function connect() {
   var ipAddress = '192.168.192.168';
   var port = '8008';

   ePosDev.connect(ipAddress, port, callback_connect);
}
```

Keyboard オブジェクトを取得（デバイスの選択）

Keyboard オブジェクトを取得してキーボードと接続し、入力データを受信するイベントを登録します。

```
function callback_connect(resultConnect){
   var deviceId = 'local_keyboard';
   var options = {'crypto' : false, 'buffer' : false};

   if ((resultConnect == 'OK') || (resultConnect == 'SSL_CONNECT_OK')) {
      //Keyboard オブジェクトを取得する
      ePosDev.createDevice(deviceId, ePosDev.DEVICE_TYPE_KEYBOARD, options,
callback_createDevice);
   }
   else {
      // エラーメッセージ表示
   }
}

var keyboard = null;

function callback_createDevice(deviceObj, errorCode){
   if (deviceObj === null) {
      // Keyboard オブジェクト取得失敗、エラーメッセージ表示
      return;
   }
   keyboard = deviceObj;
```

```
    // キー押下イベントを登録
    keyboard.onkeypress = function(response){
        if (response.keycode !== 0) {
            // 受信メッセージ表示
        }
    };
}
```

デバイスとの切断

入力処理完了後、Keyboard オブジェクトを破棄して通信を切断し、キーボードからの入力データ受信を終了します。

```
//Keyboard オブジェクトを破棄する
ePosDev.deleteDevice(keyboard, callback_deleteDevice);

function callback_deleteDevice(errorCode){
    // デバイスと切断
    ePosDev.disconnect();
}
```

設定用の命令バッファーの有効範囲

設定用に使用される Printer オブジェクトの addXXX の有効範囲は、addXXX 設定後、send メソッドが実行されるまで有効です。

設定した値は、send メソッドの実行ごとに初期化されます。以下を参考にして下さい。

```
printer.addText('Hello World\n');
printer.addTextFont(Printer.FONT_B);
printer.addText('Hello World\n');
printer.addText('Hello World\n');
printer.send();
printer.addText('Hello World\n');
printer.send();
```

赤字：フォント A
緑字：フォント B

エラーコード

Epson ePOS SDK for JavaScript には、callback パラメーターで取得するエラーコードと、onreceive イベントで取得するエラーコードがあります。それぞれのエラーコードに対応する要因と対処方法は、以下を参照してください。

- callback パラメーターで取得するエラーコードと対処方法
- onreceive イベントで取得するエラーコードと対処方法

API リファレンス

本章では、ePOS SDK for JavaScript で用意されている API について説明しています。

ePOS SDK API一覧

ePOS SDK for JavaScript には、以下のオブジェクトが用意されています。

✎	• プリンターや周辺機器によって、使用可能な API やパラメーターが異なります。 サポートオブジェクト一覧を参照してください。 • 各オブジェクトの持つ API とプリンターごとの対応一覧は、サポート API 一覧を参照してください。

- ❏ ePOSDevice オブジェクト (window.epson.ePOSDevice)
- ❏ デバイスオブジェクト共通
- ❏ OtherPeripheral オブジェクト
- ❏ CashChanger オブジェクト
- ❏ CAT オブジェクト
- ❏ Display オブジェクト
- ❏ Keyboard オブジェクト
- ❏ POSKeyboard オブジェクト
- ❏ MSR オブジェクト
- ❏ Printer オブジェクト
- ❏ Scanner オブジェクト
- ❏ SimpleSerial オブジェクト
- ❏ DeviceHubTerminal オブジェクト
- ❏ CommBoxManager オブジェクト
- ❏ CommBox オブジェクト
- ❏ ePOSDeviceConfiguration オブジェクト

ePOSDevice オブジェクト

アプリケーションとプリンターを接続します。

API		説明
初期化	Constructor	ePOSDevice オブジェクトの初期化
通信経路	connect メソッド	通信経路の確立
	disconnect メソッド	通信経路の切り離し
	isConnected メソッド	通信経路の確立状態を取得
デバイス	createDevice メソッド	デバイスオブジェクトの取得
	deleteDevice メソッド	デバイスオブジェクトの破棄

API		説明
管理情報	getAdmin メソッド	管理者情報の取得
	getLocation メソッド	設置場所情報の取得
OFSC	sendOfscXml メソッド	OFSC-Print 用 XML データの送信
アプリケーション間通信	getCommBoxManager メソッド	CommBoxManager オブジェクトを取得
再接続開始	onreconnecting イベント	再接続処理開始イベント
再接続終了	onreconnect イベント	再接続終了イベント
切断	ondisconnect イベント	ネットワーク切断イベント

デバイスオブジェクト共通

開発したデバイス制御スクリプト経由で、周辺機器を制御します。

API		説明
イベント呼び出し	callEvent メソッド	対応するデバイス制御スクリプトの任意イベントの呼び出し（SimpleSerial オブジェクトはサポートしていません）

OtherPeripheral オブジェクト

開発したデバイス制御プログラム経由で、周辺機器を制御します。

API		説明
送信	send メソッド	制御命令の送信
結果受信	onreceive イベント	制御結果受信イベント

CashChanger オブジェクト

自動つり銭機を用いた出入金処理を制御します。

デバイス制御プログラムから制御する場合と、デバイス制御スクリプトから制御する場合で、使用できる API が異なります。

デバイス制御プログラム

API		説明
計数	readCashCounts メソッド	デバイス内現金の金額を取得
入金	beginDeposit メソッド	入金処理を開始
	pauseDeposit メソッド	入金処理を一時停止
	restartDeposit メソッド	入金処理を再開
	endDeposit メソッド	入金処理を終了

API		説明
出金	dispenseChange メソッド	金額を指定して出金
	dispenseCash メソッド	金種を指定して出金
コマンド送信	sendCommand メソッド	OPOS の DirectIO メソッドを送信
結果受信	oncashcount イベント	計数の結果通知イベント
	ondeposit イベント	入金金額の通知イベント
	ondispense イベント	出金動作の通知イベント
	oncommandreply イベント	OPOS の DirectIO メソッドの送信結果通知イベント
	ondirectio イベント	OPOS ドライバーの DirectIOEvent
	onstatusupdate イベント	OPOS ドライバーの StatusUpdateEvent

デバイス制御スクリプト

API		説明
機器設定	setConfig メソッド	デバイスの設定を変更
計数	readCashCounts メソッド	デバイス内現金の金額を取得
入金	beginDeposit メソッド	入金処理を開始
	pauseDeposit メソッド	入金処理を一時停止
	restartDeposit メソッド	入金処理を再開
	endDeposit メソッド	入金処理を終了
出金	dispenseChange メソッド	金額を指定して出金
	dispenseCash メソッド	金種を指定して出金
回収	collectCash メソッド	デバイス内の現金を回収
ドロアー	openDrawer メソッド	キャッシュドロアーを開く
コマンド送信	sendCommand メソッド	任意コマンドを送信
結果受信	onconfigchange イベント	デバイスの設定変更の結果通知イベント
	oncashcount イベント	計数の結果通知イベント
	ondeposit イベント	入金金額の通知イベント
	ondispense イベント	出金動作の通知イベント
	oncollect イベント	回収動作の通知イベント
	oncommandreply イベント	任意コマンドの結果通知イベント
	onstatuschange イベント	状態変化イベント

CAT オブジェクト

信用照会端末での電子決済処理を制御します。

API		説明
タイムアウト	timeout プロパティー	決済メディア読み取りタイムアウト
動作モード	trainingMode プロパティー	動作モードの切り替え
売上	authorizeSales メソッド	売上処理を実行
取消	authorizeVoid メソッド	取消処理を実行
返品	authorizeRefund メソッド	返品処理を実行
承認後売上	authorizeCompletion メソッド	承認後売上処理を実行
日計	accessDailyLog メソッド	日計処理を実行
コマンド送信	sendCommand メソッド	OPOS の DirectIO メソッドを送信
通信確認	checkConnection メソッド	OPOS と信用照会端末の接続状態を確認
キャンセル	clearOutput メソッド	処理を中止
結果受信	onauthorizesales イベント	売上処理の結果通知イベント
	onauthorizevoid イベント	取消処理の結果通知イベント
	onauthorizerefund イベント	返品処理の結果通知イベント
	onauthorizecompletion イベント	承認後売上処理の結果通知イベント
	onaccessdailylog イベント	日計処理の結果通知イベント
	oncheckConnection イベント	接続確認の結果通知イベント
	onclearOutput イベント	処理中止の結果通知イベント
	oncommandreply イベント	OPOS の DirectIO メソッドの送信結果通知
	ondirectio イベント	OPOS ドライバーの DirectIOEvent
	onstatusupdate イベント	OPOS ドライバーの StatusUpdateEvent

Display オブジェクト

カスタマーディスプレイへの文字表示を制御します。

API		説明
ウィンドウ	createWindow メソッド	表示領域の定義
	destroyWindow メソッド	表示領域の設定の破棄
	setCurrentWindow メソッド	表示領域の切り替え
	clearWindow メソッド	現在の表示領域を消去
	addCreateScreen メソッド	ベースウィンドウの定義
	addCreateScreenCustom メソッド	ベースウィンドウの定義
	createTextArea メソッド	表示領域の定義

API		説明
ウィンドウ	destroyTextArea メソッド	表示領域の設定の破棄
	setCurrentTextArea メソッド	表示領域の切り替え
	clearTextArea メソッド	現在の表示領域を消去
カーソル	setCursorPosition メソッド	カーソル位置の移動
	moveCursorPosition メソッド	表示領域内でカーソル位置を移動
	setCursorType メソッド	カーソル表示の変更
テキスト表示	addText メソッド	テキストの表示
	addReverseText メソッド	反転テキストの表示
	addMarquee メソッド	マーキー表示
表示属性	setBlink メソッド	表示の点滅
	setBrightness メソッド	表示輝度の変更
	addBackgroundColor メソッド	背景色の設定
グラフィック	addStartSlideShow メソッド	スライドショーの開始
	addStopSlideShow メソッド	スライドショーの終了
	addDownloadImage メソッド	イメージ表示
	addRegisterDownloadImage メソッド	イメージ登録
	addNVImage メソッド	NV イメージ表示
	addClearImage メソッド	イメージ消去
バーコード	addSymbol メソッド	2 次元シンボル表示
	addClearSymbol メソッド	2 次元シンボル消去
時計	showClock メソッド	時計の表示
コマンド送信	addCommand メソッド	任意コマンドの実行
送信	send メソッド	制御命令の送信
初期化	reset メソッド	リセット
結果受信	onreceive イベント	制御結果受信イベント

使用するカスタマーディスプレイやその表示モードによって、使用できる API やパラメーターが異なります。
以下の表に各カスタマーディスプレイと表示モードで使用できる API を示します。

API	パラメーター	DM-D30	DM-D70（標準モード）	DM-D70（桁行固定モード）	DM-D110
createWindow メソッド		✔	-	✔	✔
destroyWindow メソッド		✔	-	✔	✔
setCurrentWindow メソッド		✔	-	✔	✔
clearWindow メソッド		✔	-	✔	✔
addCreateScreen メソッド		-	✔	✔	-

API	パラメーター	DM-D30	DM-D70（標準モード）	DM-D70（桁行固定モード）	DM-D110
addCreateScreenCustom メソッド		-	✔	✔	-
createTextArea メソッド		-	✔	-	-
destroyTextArea メソッド		-	✔	-	-
setCurrentTextArea メソッド		-	✔	-	-
clearTextArea メソッド		-	✔	-	-
setCursorPosition メソッド		✔	✔	✔	✔
moveCursorPosition メソッド		✔	✔	✔	✔
setCursorType メソッド		✔	-	-	✔
addText メソッド	data	✔	✔	✔	✔
	lang	✔*	✔	✔	✔*
	x	✔	✔	✔	✔
	y	✔	✔	✔	✔
	r	-	✔	-	-
	g	-	✔	-	-
	b	-	✔	-	-
addReverseText メソッド		✔	-	-	✔
addMarquee メソッド		✔	✔	✔	✔
setBlink メソッド		✔	-	-	✔
setBrightness メソッド		✔	✔	✔	✔
addBackgroundColor メソッド		-	✔	-	-
addStartSlideShow メソッド		-	✔	-	-
addStopSlideShow メソッド		-	✔	-	-
addDownloadImage メソッド		-	✔	-	-
addRegisterDownloadImage メソッド		-	✔	-	-
addNVImage メソッド		-	✔	-	-
addClearImage メソッド		-	✔	-	-
addSymbol メソッド		-	✔	-	-
addClearSymbol メソッド		-	✔	-	-
showClock メソッド		-	-	-	✔
addCommand メソッド		✔	✔	✔	✔
send メソッド		✔	✔	✔	✔
reset メソッド		✔	✔	✔	✔
onreceive イベント		✔	✔	✔	✔

*: 設定値 "mul" は使用できません。

Keyboard オブジェクト

キーボードによる文字入力を制御します。

API		説明
文字列設定	setPrefix メソッド	連続した文字列として扱う条件を設定
キー検出	onkeypress イベント	キー押下検出イベント
文字列検出	onstring イベント	文字列検出イベント
キーコード設定	setMSRPrefix メソッド	MSR データ等を制御する条件を設定
データ検出	ondata イベント	カード情報受信イベント

POSKeyboard オブジェクト

POS キーボードによる文字入力を制御します。

API		説明
キー検出	onkeypress イベント	キー押下検出イベント

MSR オブジェクト

MSR によるカード読み取りを制御します。

API		説明
データ検出	ondata イベント	カードデータ検出イベント

Printer オブジェクト

プリンターへの印刷を制御します。

API		説明
テキスト	addTextAlign メソッド	位置揃え設定を命令バッファーに追加
	addTextLineSpace メソッド	改行量設定を命令バッファーに追加
	addTextRotate メソッド	倒立印字設定を命令バッファーに追加
	addText メソッド	文字印字を命令バッファーに追加
	addTextLang メソッド	言語設定を命令バッファーに追加
	addTextFont メソッド	文字フォント設定を命令バッファーに追加
	addTextSmooth メソッド	文字スムージング設定を命令バッファーに追加
	addTextDouble メソッド	文字倍角設定を命令バッファーに追加
	addTextSize メソッド	文字倍率設定を命令バッファーに追加
	addTextStyle メソッド	文字装飾設定を命令バッファーに追加

API		説明
テキスト	addTextPosition メソッド	文字印字位置設定を命令バッファーに追加
	addTextVPosition メソッド	縦方向の印字開始位置設定を命令バッファーに追加
紙送り	addFeedUnit メソッド	ドット単位の紙送りを命令バッファーに追加
	addFeedLine メソッド	行単位の紙送りを命令バッファーに追加
	addFeedPosition メソッド	レシート紙、ラベル紙の制御を命令バッファーに追加
	addFeed メソッド	改行を命令バッファーに追加
グラフィック	addImage メソッド	ラスターイメージ印字を命令バッファーに追加
	addLogo メソッド	NV ロゴ印字を命令バッファーに追加
バーコード	addBarcode メソッド	バーコード印字を命令バッファーに追加
	addSymbol メソッド	2 次元シンボル印字を命令バッファーに追加
罫線	addHLine メソッド	横罫線印字を命令バッファーに追加
	addVLineBegin メソッド	縦罫線開始を命令バッファーに追加
	addVLineEnd メソッド	縦罫線終了を命令バッファーに追加
ページモード	addPageBegin メソッド	ページモード開始を命令バッファーに追加
	addPageEnd メソッド	ページモード終了を命令バッファーに追加
	addPageArea メソッド	ページモード印字領域設定を命令バッファーに追加
	addPageDirection メソッド	ページモード印字方向設定を命令バッファーに追加
	addPagePosition メソッド	ページモード印字位置設定を命令バッファーに追加
	addPageLine メソッド	ページモード直線描画を命令バッファーに追加
	addPageRectangle メソッド	ページモード四角形描画を命令
まとめ印刷（180 度回転）	addRotateBegin メソッド	まとめ印刷（180 度回転）モード開始を命令バッファーに追加
	addRotateEnd メソッド	まとめ印刷（180 度回転）モード終了を命令バッファーに追加
カット	addCut メソッド	用紙カットを命令バッファーに追加
ドロアー	addPulse メソッド	ドロアーキックを命令バッファーに追加
ブザー	addSound メソッド	ブザー鳴動を命令バッファーに追加
レイアウト	addLayout メソッド	用紙レイアウトの設定を命令バッファーに追加
リカバリー	recover メソッド	復帰可能エラーから復帰
	addRecovery メソッド	エラーからの復帰タグを追加する

API		説明
リセット	reset メソッド	プリンターをリセット
	addReset メソッド	プリンターのリセットタグを追加する
コマンド送信	addCommand メソッド	コマンドを命令バッファーに追加
送信	send メソッド	• 印刷ドキュメントの送信 • ジョブ ID を指定して送信
	print メソッド	• HTML5 Canvas の印刷 • ジョブ ID を指定して印刷
印刷ジョブ	getPrintJobStatus メソッド	印刷ジョブの状態を取得
状態監視	startMonitor メソッド	ステータスイベントの有効化
	stopMonitor メソッド	ステータスイベントの無効化
イメージ	halftone プロパティー	ラスターイメージのハーフトーン処理方法
	brightness プロパティー	ラスターイメージの明るさ補正値
強制送信	force プロパティー	強制送信モード
タイムアウト	timeout プロパティー	送信タイムアウト時間
監視間隔	interval プロパティー	プリンターステータスの更新間隔
ドロアーオープン	drawerOpenLevel プロパティー	ドロアーの信号線状態
命令バッファー	message プロパティー	命令バッファーの直接操作
結果受信	onreceive イベント	• 応答ドキュメント受信イベント • ジョブ ID 指定の受信イベント
	onstatuschange イベント	ステータス変更イベント
	onbatterystatuschange イベント	バッテリーステータス変更イベント
	ononline イベント	オンラインイベント
	onoffline イベント	オフラインイベント
	onpoweroff イベント	無応答イベント
	oncoverok イベント	カバークローズイベント
	oncoveropen イベント	カバーオープンイベント
	onpaperok イベント	用紙ありイベント
	onpaperend イベント	用紙エンドイベント
	onpapernearend イベント	用紙ニアエンドイベント
	ondrawerclosed イベント	ドロアークローズイベント
	ondraweropen イベント	ドロアーオープンイベント
	onbatteryok イベント	バッテリー残量ありイベント
	onbatterylow イベント	バッテリー残量なしイベント

印字モードには、スタンダードモードとページモード、まとめ印刷（180 度回転）モードがあります。

- スタンダードモード

  1 行単位で印字する印字モードです。文字サイズ、画像、バーコードなどの高さに合わせて改行量が調整されるため、印字量によって用紙の長さが変化するレシート印刷に適しています。

- ページモード

  印字領域を設定して印字データ（文字 / 画像 / バーコード等）を展開し、一括印字する印字モードです。

  ページモードでの処理を行う際は、addPageBegin メソッドでページモード処理を開始し、addPageEnd メソッドでページモード処理を終了してください。

- まとめ印刷（180 度回転）モード

  印字データ（文字 / 画像 / バーコード等）の範囲を設定し、設定した範囲の印字データをまとめて 180 度回転させて印刷する印字モードです。

  まとめ印刷（180 度回転）モードでの処理を行う際は、addRotateBegin メソッドでまとめ印刷（180 度回転）処理を開始し、addRotateEnd メソッドでまとめ印刷（180 度回転）処理を終了してください。

以下の表に各モードで使用できる API を示します。

API	スタンダードモード	ページモード	まとめ印刷（180 度回転）モード
addTextAlign メソッド	✔	-	✔
addTextLineSpace メソッド	✔	✔	✔
addTextRotate メソッド	✔	-	✔
addText メソッド	✔	✔	✔
addTextLang メソッド	✔	✔	✔
addTextFont メソッド	✔	✔	✔
addTextSmooth メソッド	✔	✔	✔
addTextDouble メソッド	✔	✔	✔
addTextSize メソッド	✔	✔	✔
addTextStyle メソッド	✔	✔	✔
addTextPosition メソッド	✔	✔	✔
addTextVPosition メソッド	-	✔	✔
addFeedUnit メソッド	✔	✔	✔
addFeedLine メソッド	✔	✔	✔
addFeedPosition メソッド	✔	-	✔
addFeed メソッド	✔	✔	✔
addImage メソッド	✔	✔	✔
addLogo メソッド	✔	✔	✔
addBarcode メソッド	✔	✔	✔
addSymbol メソッド	✔	✔	✔
addHLine メソッド	✔	-	✔
addVLineBegin メソッド	✔	-	✔
addVLineEnd メソッド	✔	-	✔

API	スタンダードモード	ページモード	まとめ印刷（180 度回転）モード
addPageBegin メソッド	✔	-	-
addPageEnd メソッド	-	✔	-
addPageArea メソッド	-	✔	-
addPageDirection メソッド	-	✔	-
addPagePosition メソッド	-	✔	-
addPageLine メソッド	-	✔	-
addPageRectangle メソッド	-	✔	-
addRotateBegin メソッド	✔	-	-
addRotateEnd メソッド	-	-	✔
addCut メソッド	✔	-	-
addPulse メソッド	✔	-	-
addSound メソッド	✔	-	-
addLayout メソッド	✔	-	✔
recover メソッド	✔	-	-
addRecovery メソッド	✔	-	-
reset メソッド	✔	-	-
addReset メソッド	✔	-	-
addCommand メソッド	✔	✔	✔
send メソッド	✔	✔	-
print メソッド	✔	✔	-
getPrintJobStatus メソッド	✔	✔	✔
startMonitor メソッド	✔	✔	✔
stopMonitor メソッド	✔	✔	✔

Scanner オブジェクト

バーコードスキャナーによるバーコード読み取りを制御します。

API		説明
データ検出	ondata イベント	バーコードデータ検出イベント

SimpleSerial オブジェクト

プリンターとデバイスのシリアル通信を制御します。

API		説明
文字列設定	sendCommand	任意コマンドの送信

API		説明
応答検出	oncommandreply イベント	任意コマンドの送信結果通知イベント

DeviceHubTerminal オブジェクト

TM-DT シリーズを制御します。

API		説明
シャットダウン	shutdown メソッド	プリンターをシャットダウン
再起動	restart メソッド	プリンターを再起動

CommBoxManager オブジェクト

コミュニケーションボックスのオープン / クローズを制御します。

API		説明
アプリケーション間通信	openCommBox メソッド	コミュニケーションボックスをオープン
	closeCommBox メソッド	コミュニケーションボックスをクローズ

CommBox オブジェクト

コミュニケーションボックスとアプリケーション間のデータ送受信を制御します。

API		説明
送信履歴	getCommHistory メソッド	データの送信履歴を取得
送信	send メソッド	コミュニケーションボックスにデータを送信
受信	onreceive イベント	コミュニケーションボックスのデータ受信を通知

ePOSDeviceConfiguration オブジェクト

プリンターに登録されているデバイスの情報を取得します。

API		説明
コンストラクター	Constructor	ePOSDeviceConfiguration オブジェクトを作成
状態取得	getRegisterdDevices メソッド	デバイスの利用可能状況を取得

ePOSDeviceオブジェクト

Constructor

ePOSDevice オブジェクトを新しく作成し、初期化します。

構文

```
ePOSDevice();
```

戻り値

戻り値	オブジェクト種類
ePOSDevice オブジェクト	ePOSDevice

connect メソッド

プリンターとの通信を開始します。

構文

- ❏ connect(ipAddress, port, callback);
- ❏ connect(ipAddress, port, callback, options);

パラメーター

ipAddress

設定値	説明
ipAddress	プリンターの IP アドレスを指定

port

通信方式を指定します。

設定値	説明
8008	HTTP 通信
8043	SSL/TLS 通信

callback

実行結果を受け取るコールバック関数を指定します。
第 1 パラメーターに結果を示す以下の文字列が渡されます。

文字列	説明
"OK"	ePOS-Print オプション有効時、もしくは ePOS-Print オプション無効時に HTTP 通信で接続が成功した
"SSL_CONNECT_OK"	ePOS-Print オプション無効時に SSL/TLS 通信で接続が成功した
"ERROR_TIMEOUT"	タイムアウトが発生した
"ERROR_PARAMETER"	パラメーターエラーが発生した

options

オブジェクトリテラル（文字列型）でプロパティーを指定します。

プロパティー	説明	true/false
eposprint	ePOS-Print オプションを有効にするかどうかを指定します。	true：有効にする false：無効にする

TM プリンターモデルとカスタマーディスプレイモデルのシステム構成では、ePOS-Print オプションを有効にしてください。

TM プリンター +DM-D+ バーコードスキャナーモデルと POS ターミナルモデルのシステム構成では、ePOS-Print オプションを無効にしてください。ePOS-Print オプションを有効にすると、子機プリンターとカスタマーディスプレイ以外の周辺機器が制御できなくなります。

例

```
connect("192.168.192.168", "8008", callback, {"eposprint" : true});
```

補足説明

- options を指定する構文の機能については、6 章「Web ページを頻繁に更新 / 遷移するアプリケーションを開発するには」を参照してください。
- 本 API は TM プリンター（親機）と周辺機器の接続状態に関わらず成功します。
- 連続して処理する場合や、複数の端末から制御しない場合は、デバイスと接続したままでの使用を推奨します。
- options パラメーターを省略すると、「false：無効」が指定されます。
- ePOS-Print オプションを無効にして接続に失敗した場合、ePOS-Print オプションを有効にして接続します。
  この接続に成功すると、port パラメーターで指定した通信方式に関わらず、callback パラメーターには "OK" が渡されます。
- portパラメーターに8043、optionsパラメーターにtrueを指定してeposprintオプションを有効にした場合、callback パラメーターには "OK" が返ります。
- options パラメーターに true を指定して eposprint オプションを有効にした場合、ePOSDevice オブジェクトで使用できるメソッドは以下の通りです。

メソッド	使用可否
Constructor	✔
connect メソッド	✔
disconnect メソッド	✔
isConnected メソッド	✔
createDevice メソッド	✔
deleteDevice メソッド	✔
getAdmin メソッド	-
getLocation メソッド	-
sendOfscXml メソッド	-
getCommBoxManager メソッド	-
onreconnecting イベント	-
onreconnect イベント	-
ondisconnect イベント	✔

disconnect メソッド

プリンターとの通信を終了します。

構文

```
disconnect();
```

補足説明

- 通信状態が「接続済み」のときに本 API を実行すると、ondisconnect イベントが発生します。
- 連続して処理を行う場合や、複数の端末から制御しない場合は、デバイスと接続したままでの使用を推奨します。

isConnected メソッド

connect メソッドによる通信状態を取得します。

構文

```
isConnected();
```

戻り値

戻り値	説明
true	接続済み
false	未接続

補足説明

- 端末がスリープした場合、戻り値に正しい値を返さない場合があります。
- ondisconnect イベントが発生すると、連動して false（未接続）になります。

createDevice メソッド

デバイスのインターフェイスとなるデバイスオブジェクトを取得します。取得したデバイスオブジェクトは、コールバック関数に渡されます。

構文

- ❏ createDevice(deviceId, deviceType, options, callback);
- ❏ createDevice(deviceId, deviceType, crypto, callback);

パラメーター

deviceId

設定値	説明
deviceId	デバイス ID を文字列で指定

deviceType

設定値	デバイスの種類
DEVICE_TYPE_CASH_CHANGER	自動つり銭機
DEVICE_TYPE_DISPLAY	カスタマーディスプレイ
DEVICE_TYPE_KEYBOARD	キーボード
DEVICE_TYPE_MSR	MSR
DEVICE_TYPE_PRINTER	プリンター
DEVICE_TYPE_SCANNER	バーコードスキャナー
DEVICE_TYPE_SIMPLE_SERIAL	シリアル通信デバイス
DEVICE_TYPE_DT	TM-DT シリーズ
DEVICE_TYPE_CAT	信用照会端末
DEVICE_TYPE_POSKEYBOARD	POS キーボード
DEVICE_TYPE_OTHER_PERIPHERAL	その他の周辺機器

options

オブジェクトリテラル（文字列型）でプロパティーを指定します。

プロパティー	説明	true/false
crypto (Boolean)	通信データを暗号化するかどうかを指定します。	true：暗号化する false：暗号化しない
buffer (Boolean)	デバイスとの再接続処理中に発生したデータを、再接続後に送信するかどうかを指定します。	true：送信する false：送信しない

crypto

プロパティー	説明	true/false
crypto (Boolean)	通信データを暗号化するかどうかを指定します。	true：暗号化する false：暗号化しない

callback

実行結果を受け取るコールバック関数を指定します。

第 1 パラメーターにデバイスオブジェクト、第 2 パラメーターには、結果を示す以下の文字列が渡されます。

文字列	説明
"OK"	デバイスオブジェクトの取得に成功した
"DEVICE_NOT_FOUND"	デバイスが見つからない
"DEVICE_IN_USE"	デバイスが使用中
"DEVICE_OPEN_ERROR"	デバイスのオープンに失敗した
"DEVICE_TYPE_INVALID"	デバイスの種類が違う
"PARAM_ERROR"	パラメーターエラーが発生した
"SYSTEM_ERROR"	その他のエラーが発生した

補足説明

- ❏ createDevice メソッドの実行に成功すると、プリンターを除くデバイスは排他的にロックされます。
- ❏ deviceType に DEVICE_TYPE_DISPLAY を指定する場合は、deviceId に local_display を指定します。
- ❏ deviceType に DEVICE_TYPE_DT を指定する場合は、deviceId に local_dt を指定します。
- ❏ デバイスオブジェクトが取得できなかった場合、第 1 パラメーターにデバイスオブジェクトは渡されず null が渡されます。
- ❏ 排他的にロックされたデバイスを指定して createDevice メソッドを呼び出すと、DEVICE_IN_USE が返ります。
- ❏ ondisconnect イベント発生時の復帰処理の際、createDevice メソッドを実行するタイミングによって DEVICE_IN_USE が発生することがあります。この場合、DEVICE_IN_USE 以外が取得できるまで、繰り返し createDevice メソッドを実行してください。
- ❏ createDevice メソッド実行後、同じデバイスに再度 createDevice メソッドを実行すると、処理のタイミングによって DEVICE_OPEN_ERROR が発生し、デバイスとの接続が切れることがあります。もう一度 createDevice メソッドを実行し、接続し直してください。
- ❏ 本 API は TM プリンター（親機）と周辺機器の接続状態に関わらず成功します。
- ❏ TM-m30からDM-D30/DM-D70を制御する場合とTM-T88VIからDM-D110/DM-D30/DM-D70を制御する場合、DM-D110/DM-D30/DM-D70 は排他的にロックされません。他のデバイスからこの DM-D110/DM-D30/DM-D70 を指定して createDevice メソッドを呼び出した際、DEVICE_IN_USE を返しません。
- ❏ 通信データの暗号化機能は、TM-i シリーズと TM-DT シリーズで使用できます。
- ❏ TM プリンター +DM-D+ バーコードスキャナーモデルのシステム構成では、connect メソッド実行時、options パラメーターの eposprint プロパティーを有効にしないでください。
- ❏ TM プリンター +DM-D+ バーコードスキャナーモデルのシステム構成では、deviceType に対し、以下の deviceId を指定してください。

deviceType	deviceId
DEVICE_TYPE_PRINTER	local_printer
DEVICE_TYPE_DISPLAY	local_display
DEVICE_TYPE_SCANNER	local_scanner

deleteDevice メソッド

createDevice メソッドで取得したデバイスオブジェクトを破棄します。破棄したデバイスオブジェクトが制御していたデバイスは解放され、他のアプリケーションから使用可能になります。

構文

```
deleteDevice(deviceObject, callback);
```

パラメーター

deviceObject

設定値	説明
deviceObject	破棄するデバイスオブジェクトを指定

callback

実行結果を受け取るコールバック関数を指定します。

第 1 パラメーターに結果を示す以下の文字列が渡されます。

文字列	説明
"OK"	デバイスのクローズに成功した
"DEVICE_NOT_OPEN"	デバイスがオープンされていない
"DEVICE_CLOSE_ERROR"	デバイスのクローズに失敗した
"SYSTEM_ERROR"	その他のエラーが発生した

getAdmin メソッド

TM-DT ソフトウェア /TM-i ファームウェアに設定された管理者名を取得します。

構文

```
getAdmin();
```

戻り値

戻り値	オブジェクト種類
管理者名の文字列	String

補足説明

管理者名は EPSON TMNet WebConfig で設定します。

getLocation メソッド

TM-DT ソフトウェア /TM-i ファームウェアに設定された設置場所情報を取得します。

構文

```
getLocation();
```

戻り値

戻り値	オブジェクト種類
設置場所情報の文字列	String

補足説明

設置場所情報は EPSON TMNet WebConfig で設定します。

sendOfscXml メソッド

OFSC-Print サービスに印刷データ（XML データ）を送信します。

構文

```
sendOfscXml(xml, timeout, crypto, callback);
```

パラメーター

xml

設定値	説明
xml	OFSC-Print 用 XML データを指定

timeout

設定値	説明
timeout	印刷時のタイムアウト時間を msec 単位で指定

crypto

設定値	説明
true	通信データを暗号化する
false	通信データを暗号化しない

callback

実行結果を受け取るコールバック関数を指定します。

設定値	説明
callback	実行結果 XML データ

補足説明

印刷データ（XML データ）の詳細は、OFSC-Print ユーザーズマニュアルを参照してください。

getCommBoxManager メソッド

アプリケーション間通信を行う CommBoxManager オブジェクトを取得します。

構文

```
getCommBoxManager();
```

戻り値

戻り値	オブジェクト種類
CommBoxManager オブジェクト	CommBoxManager

onreconnecting イベント

ネットワークの再接続処理が開始されたことを受け取ります。onreconnecting イベントが発生した際に、再接続が開始された旨を示すメッセージを表示するなどの処理を行ってください。

構文

```
Function()
```

補足説明

onreconnecting イベント発生時は何も呼び出さないでください。
onreconnect イベントもしくは ondisconnect イベントが発生するまで待ってください。

onreconnect イベント

再接続処理が成功したことを受け取ります。onreconnect イベントが発生した際に、再接続処理中に表示していたメッセージを消すなどの処理を行ってください。

構文

```
Function()
```

ondisconnect イベント

接続処理に失敗したことを受け取ります。ondisconnect イベントは、以下の場合に発生します。

- disconnect メソッドの実行時
- onreconnecting イベント発生後の接続処理失敗時

ondisconnect イベントが発生した際に、再度接続処理行うなど、ネットワーク切断対策の処理を行ってください。

構文

```
Function()
```

デバイスオブジェクト共通

callEvent メソッド

デバイス制御スクリプト側の任意イベントを指定し、プロパティー名と値のセットからなる JSON データを渡します。
詳細は、TM-DT シリーズ周辺機器制御ガイドを参照してください。

構文

```
callEvent(eventName, data);
```

パラメーター

eventName

設定値	説明
String	デバイス制御スクリプト側のイベント名を指定

data

設定値	説明
Object	イベントに渡すデータを指定

OtherPeripheral オブジェクト

send メソッド

周辺機器へデータを送信します。デバイス制御プログラム側の任意関数を指定し、プロパティー名と値のセットからなる JSON データを渡します。
詳細は、TM-DT シリーズ周辺機器制御ガイドを参照してください。

構文

```
send(methodName, data);
```

パラメーター

methodName

設定値	説明
String	実行する関数名を指定

data

設定値	説明
JSON object	関数へ渡すパラメーターを指定

data オブジェクトのプロパティー

プロパティー	説明	有効な値	オブジェクト種類
" パラメーター名 "	パラメーターに渡す任意データを指定	a ～ z、A ～ Z、0 ～ 9	String

onreceive イベント

周辺機器からの受信データを通知します。デバイス制御プログラム側で指定した任意のイベント名のプロパティーにコールバック関数を指定することで、パラメーターの値を受信します。

構文

```
Function(eventName, data);
```

パラメーター

eventName

設定値	説明
String	イベント名を指定

data

設定値	説明
JSON object	受信したパラメーター

data オブジェクトのプロパティー

プロパティー	説明	有効な値	オブジェクト種類
" パラメーター名 "	受信する任意データを指定	a ～ z、A ～ Z、0 ～ 9	String

CashChanger オブジェクト（デバイス制御プログラム）

readCashCounts メソッド

デバイス内の金種、枚数情報を取得します。readCashCounts メソッドの実行結果は、oncashcount イベントのコールバック関数に渡されます。

構文

```
readCashCounts();
```

beginDeposit メソッド

入金処理を開始します。beginDeposit メソッドの実行結果は、ondeposit イベントのコールバック関数に渡されます。

構文

```
beginDeposit();
```

pauseDeposit メソッド

入金処理を一時停止します。pauseDeposit メソッドの実行結果は、ondeposit イベントのコールバック関数に渡されます。

構文

```
pauseDeposit();
```

restartDeposit メソッド

入金処理を再開します。restartDeposit メソッドの実行結果は、ondeposit イベントのコールバック関数に渡されます。

構文

```
restartDeposit();
```

endDeposit メソッド

入金処理を終了します。endDeposit メソッドの実行結果は、ondeposit イベントのコールバック関数に渡されます。

構文

```
endDeposit(cmd);
```

パラメーター

cmd

終了動作を指定します。

設定値	説明
DEPOSIT_CHANGE	つり銭あり（収納部への収納を保留する）
DEPOSIT_NOCHANGE	つり銭なし（収納部に収納する）
DEPOSIT_REPAY	預かり金返却

dispenseChange メソッド

指定金額で出金します。dispenseChange メソッドの実行結果は、ondispense イベントのコールバック関数に渡されます。

構文

```
dispenseChange(cash);
```

パラメーター

cash

設定値	説明	オブジェクト種類
cash	出金する金額を指定	String

補足説明

cash の値は、0 ～ 99999 の範囲で指定します。

dispenseCash メソッド

紙幣 / 硬貨の枚数を指定して出金します。dispenseCash メソッドの実行結果は、ondispense イベントのコールバック関数に渡されます。

構文

```
dispenseCash(data);
```

パラメーター

data

プロパティー	紙幣 / 硬貨の種類	オブジェクト種類
jpy1	1 円硬貨	String
jpy5	5 円硬貨	String
jpy10	10 円硬貨	String
jpy50	50 円硬貨	String
jpy100	100 円硬貨	String
jpy500	500 円硬貨	String
jpy1000	1,000 円紙幣	String
jpy2000	2,000 円紙幣	String
jpy5000	5,000 円紙幣	String
jpy10000	10,000 円紙幣	String

補足説明

data の値は、0 ～ 99 の範囲で指定します。

sendCommand メソッド

OPOS の DirectIO メソッドを送信します。sendCommand メソッドの実行結果は、oncommandreply イベントのコールバック関数に渡されます。

構文

```
sendCommand(command)
```

パラメーター

パラメーター：command

オブジェクト種類：Object

プロパティー	説明	オブジェクト種類
command	DirectIO に渡すコマンド番号	Number
data	DirectIO に渡す数値データ	Number
string	DirectIO に渡す文字列データ	String

oncashcount イベント

readCashCounts メソッドが取得した情報を受け取ります。

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	取得実行結果	String
jpy1	1 円硬貨の枚数	String
jpy5	5 円硬貨の枚数	String
jpy10	10 円硬貨の枚数	String
jpy50	50 円硬貨の枚数	String
jpy100	100 円硬貨の枚数	String
jpy500	500 円硬貨の枚数	String
jpy1000	1,000 円紙幣の枚数	String
jpy2000	2,000 円紙幣の枚数	String
jpy5000	5,000 円紙幣の枚数	String
jpy10000	10,000 円紙幣の枚数	String

status

ステータス	説明
"SUCCESS"	取得成功
"DISCREPANCY"	金額不一致の可能性あり
"DEVICE_ERROR"	デバイスエラー

ondeposit イベント

beginDeposit メソッドと restartDeposit メソッド実行後、自動つり銭機へ入金される度に ondeposit イベントが入金金額を通知します。（使用する OPOS ドライバーが対応している場合のみ）

また、以下のメソッドの実行結果を受け取ります。

- beginDeposit メソッド
- pauseDeposit メソッド
- restartDeposit メソッド
- endDeposit メソッド

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	入金処理実行結果	String
amount	入金金額	String
jpy1	1 円硬貨の枚数	String
jpy5	5 円硬貨の枚数	String
jpy10	10 円硬貨の枚数	String
jpy50	50 円硬貨の枚数	String
jpy100	100 円硬貨の枚数	String
jpy500	500 円硬貨の枚数	String
jpy1000	1,000 円紙幣の枚数	String
jpy2000	2,000 円紙幣の枚数	String
jpy5000	5,000 円紙幣の枚数	String
jpy10000	10,000 円紙幣の枚数	String

status

ステータス	説明
"BUSY"	他コマンド要求による動作中
"PAUSE"	入金一時停止状態
"END"	入金停止状態

ステータス	説明
"COMMAND_ERROR"	送信コマンドエラー
"DEVICE_ERROR"	デバイスエラー
"SYSTEM_ERROR"	システムエラー
"OPOS_CODE_XX"	デバイスベンダー定義のエラー

補足説明

- beginDeposit メソッドと restartDeposit メソッドが成功した場合、status に "BUSY" が返ります。
- pauseDeposit メソッドが成功した場合、status に "PAUSE" が返ります。
- endDeposit メソッドが成功した場合、status に "END" が返ります。

ondispense イベント

以下のメソッドの実行結果を受け取ります。

- dispenseChange メソッド
- dispenseCash メソッド

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	出金処理実行結果	String

status

ステータス	説明
"SUCCESS"	出金の成功
"BUSY"	装置動作中のため出金不可
"COMMAND_ERROR"	送信コマンドエラー
"DEVICE_ERROR"	デバイスエラー
"SYSTEM_ERROR"	システムエラー
"ILLEGAL_PARAMETER_ERROR"	パラメーター不正エラー
"OPOS_CODE_XX"	デバイスベンダー定義のエラー

oncommandreply イベント

sendCommand メソッドの実行結果を受け取ります。

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	DirectIO の戻り値	String
command	DirectIO のコマンド番号	Number
data	DirectIO の数値データ	Number
string	DirectIO の文字列データ	String

status

ステータス	説明
"SUCCESS"	送信成功
"BUSY"	動作中のため実行できない
"SYSTEM_ERROR"	システムエラー
"DEVICE_ERROR"	デバイスエラー
"OPOS_CODE_XX"	デバイスベンダー定義のエラー

ondirectio イベント

OPOS ドライバーの DirectIOEvent を受け取ります。

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
eventnumber	DirectIOEvent のコマンド番号	Number
data	DirectIOEvent の数値データ	Number
string	DirectIOEvent の文字列データ	String

onstatusupdate イベント

OPOS ドライバーの StatusUpdateEvent を受け取ります。
StatusUpdateEvent は、自動つり銭機の収納庫ステータスや機器の状態が変化した際に発生します。
詳細は、OPOS ドライバーのマニュアルを参照してください。

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data
オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	StatusUpdateEvent の数値データ	Number

status

ステータス	説明
SUE_POWER_ONLINE	オンライン状態
SUE_POWER_OFF	電源が切れているか、プリンターと接続されていない
SUE_POWER_OFFLINE	オフライン状態
SUE_POWER_OFF_OFFLINE	• 電源が切れているか、プリンターと接続されていない • オフライン状態
SUE_STATUS_EMPTY	空状態の収納庫あり
SUE_STATUS_NEAREMPTY	空に近い状態の収納庫あり
SUE_STATUS_EMPTYOK	現金が補充され、SUE_STATUS_EMPTY、SUE_STATUS_NEAREMPTY ステータスが解除された
SUE_STATUS_FULL	満杯状態の収納庫あり
SUE_STATUS_NEARFULL	満杯に近い状態の収納庫あり
SUE_STATUS_FULLOK	現金が回収され、SUE_STATUS_FULL、SUE_STATUS_NEARFULL ステータスが解除された
SUE_STATUS_JAM	機器障害が発生
SUE_STATUS_JAMOK	機器障害から復帰

補足説明

onstatusupdate イベントで定義していないデバイスベンダー固有の StatusUpdateEvent 値については、OPOS ドライバーのマニュアルを参照してください。

CashChanger オブジェクト（デバイス制御スクリプト）

setConfig メソッド

自動つり銭機の設定を変更します。setConfig メソッドの実行結果は、onconfigchange イベントのコールバック関数に渡されます。

構文

```
setConfig(config, data);
```

パラメーター

config

設定を変更する項目を指定します。

設定値	説明
CONFIG_COUNT_MODE	計数モード
CONFIG_LEFT_CASH	残置指定

data

config で指定した項目のプロパティーを指定します。

設定項目	プロパティー	説明	値
計数モード	mode	計数モードを設定	MODE_MANUAL_INPUT ：手入力モード MODE_AUTO_COUNT ：自動計数モード
残置指定	coins	硬貨の残置金額を 10 円単位の文字列で指定	有効な値：0 ～ 99999 ＜例：10,000 円の場合＞ "1000"
	bills	紙幣の残置金額を千円単位の文字列で指定	有効な値：0 ～ 999 ＜例：50,000 円の場合＞ "50"

補足説明

bils と coins で有効な値の範囲外を指定すると、onconfigchange イベントの status に "ILLEGAL_PARAMETER_ERROR" が返ります。

readCashCounts メソッド

デバイス内の金種、枚数情報を取得します。readCashCounts メソッドの実行結果は、oncashcount イベントと onstatuschange イベントのコールバック関数に渡されます。

構文

```
readCashCounts();
```

beginDeposit メソッド

入金処理を開始します。beginDeposit メソッドの実行結果は、ondeposit イベントと onstatuschange イベントのコールバック関数に渡されます。

構文

```
beginDeposit();
```

pauseDeposit メソッド

入金処理を一時停止します。pauseDeposit メソッドの実行結果は、ondeposit イベントと onstatuschange イベントのコールバック関数に渡されます。

構文

```
pauseDeposit();
```

restartDeposit メソッド

入金処理を再開します。restartDeposit メソッドの実行結果は、ondeposit イベントと onstatuschange イベントのコールバック関数に渡されます。

構文

```
restartDeposit();
```

endDeposit メソッド

入金処理を終了します。endDeposit メソッドの実行結果は、ondeposit イベントと onstatuschange イベントのコールバック関数に渡されます。

構文

```
endDeposit(cmd);
```

パラメーター

cmd

終了動作を指定します。

設定値	説明
DEPOSIT_CHANGE	つり銭あり（収納部への収納を保留する）
DEPOSIT_NOCHANGE	つり銭なし（収納部に収納する）
DEPOSIT_REPAY	預かり金返却

補足説明

DEPOSIT_REPAY の実行結果は、ondispense イベントのコールバックメソッドに通知されます。

dispenseChange メソッド

指定金額で出金します。dispenseChange メソッドの実行結果は、ondispense イベントと onstatuschange イベントのコールバック関数に渡されます。

構文

```
dispenseChange(cash);
```

パラメーター

cash

設定値	説明	オブジェクト種類
cash	出金する金額を指定	String

補足説明

- cash の値は、0 ～ 99999 の範囲で指定します。範囲外の値を指定すると、ondispense イベントの status に "ILLEGAL_PARAMETER_ERROR" が返ります。
- ondispense イベントの status に "DEVICE_ERROR" が返る場合、自動つり銭機側の金額指定が 6 桁に設定されている可能性があります。金額指定を 5 桁に設定して使用してください。

dispenseCash メソッド

紙幣 / 硬貨の枚数を指定して出金します。dispenseCash メソッドの実行結果は、ondispense イベントと onstatuschange イベントのコールバック関数に渡されます。

構文

```
dispenseCash(data);
```

パラメーター

data

プロパティー	紙幣 / 硬貨の種類	オブジェクト種類
jpy1	1 円硬貨	String
jpy5	5 円硬貨	String
jpy10	10 円硬貨	String
jpy50	50 円硬貨	String
jpy100	100 円硬貨	String
jpy500	500 円硬貨	String
jpy1000	1,000 円紙幣	String
jpy2000	2,000 円紙幣	String
jpy5000	5,000 円紙幣	String
jpy10000	10,000 円紙幣	String

補足説明

data の値は、0 ～ 99 の範囲で指定します。

範囲外の値を指定すると、ondispense イベントの status に "ILLEGAL_PARAMETER_ERROR" が返ります。

collectCash メソッド

デバイス内の現金を回収します。collectCash メソッドの実行結果は、oncollect イベントと oncommandreply イベント、onstatuschange イベントのコールバック関数に渡されます。

構文

```
collectCash(type);
```

パラメーター

type

設定値	説明
COLLECT_ALL_CASH	全額回収
COLLECT_PART_OF_CASH	残置回収

openDrawer メソッド

キャッシュドロアー接続時に、キャッシュドロアーを開きます。openDrawer メソッドの実行結果は、onstatuschange イベントのコールバック関数に渡されます。

構文

```
openDrawer();
```

sendCommand メソッド

任意コマンドを送信します。sendCommand メソッドの実行結果は、oncommandreply イベントのコールバック関数に渡されます。

構文

```
sendCommand(command);
```

パラメーター

command

設定値	説明
command	コマンドを文字列で指定

onconfigchange イベント

setConfig メソッドによる設定変更の結果を受け取ります。

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	設定変更の実行結果	String

status

ステータス	説明
"SUCCESS"	取得成功
"BUSY"	動作中のため設定変更できない
"CASH_IN_TRAY_ERROR"	現金抜き取り待ち状態
"COMMAND_ERROR"	送信コマンドエラー
"DEVICE_ERROR"	デバイスエラー
"SYSTEM_ERROR"	システムエラー
"ILLEGAL_PARAMETER_ERROR"	パラメーター不正エラー
"ERROR_NOT_SUPPORTED"	デバイス制御プログラムで制御する自動つり銭機に処理を実行した

oncashcount イベント

readCashCounts メソッドが取得した情報を受け取ります。

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	取得実行結果	String
jpy1	1 円硬貨の枚数	String
jpy5	5 円硬貨の枚数	String
jpy10	10 円硬貨の枚数	String
jpy50	50 円硬貨の枚数	String
jpy100	100 円硬貨の枚数	String
jpy500	500 円硬貨の枚数	String
jpy1000	1,000 円紙幣の枚数	String
jpy2000	2,000 円紙幣の枚数	String
jpy5000	5,000 円紙幣の枚数	String
jpy10000	10,000 円紙幣の枚数	String
jpy1000cassette	回収部の 1,000 円紙幣の枚数	String
jpy2000cassette	回収部の 2,000 円紙幣の枚数	String
jpy5000cassette	回収部の 5,000 円紙幣の枚数	String
jpy10000cassette	回収部の 10,000 円紙幣の枚数	String

status

ステータス	説明
"SUCCESS"	取得成功
"BUSY"	動作中のため取得できない
"DISCREPANCY"	金額不一致の可能性あり
"COMMAND_ERROR"	送信コマンドエラー
"DEVICE_ERROR"	デバイスエラー

ステータス	説明
"SYSTEM_ERROR"	システムエラー

ondeposit イベント

beginDeposit メソッドと restartDeposit メソッド実行後、自動つり銭機へ入金される度に ondeposit イベントが入金金額を通知します。また、以下のメソッドの実行結果を受け取ります。

- beginDeposit メソッド
- pauseDeposit メソッド
- restartDeposit メソッド
- endDeposit メソッド

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	入金処理実行結果	String
amount	入金金額	String
jpy1	1 円硬貨の枚数	String
jpy5	5 円硬貨の枚数	String
jpy10	10 円硬貨の枚数	String
jpy50	50 円硬貨の枚数	String
jpy100	100 円硬貨の枚数	String
jpy500	500 円硬貨の枚数	String
jpy1000	1,000 円紙幣の枚数	String
jpy2000	2,000 円紙幣の枚数	String
jpy5000	5,000 円紙幣の枚数	String
jpy10000	10,000 円紙幣の枚数	String

status

ステータス	説明
"BUSY"	他コマンド要求による動作中
"PAUSE"	入金一時停止状態
"END"	入金停止状態
"CASH_IN_TRAY_ERROR"	現金抜き取り待ち状態

ステータス	説明
"REJECT_UNIT_ERROR"	リジェクトユニット容量オーバーエラー
"COMMAND_ERROR"	送信コマンドエラー
"DEVICE_ERROR"	デバイスエラー
"SYSTEM_ERROR"	システムエラー

補足説明

- beginDeposit メソッドと restartDeposit メソッドが成功した場合、status に "BUSY" が返ります。
- pauseDeposit メソッドが成功した場合、status に "PAUSE" が返ります。
- endDeposit メソッドが成功した場合、status に "END" が返ります。

ondispense イベント

以下のメソッドの実行結果を受け取ります。

- dispenseChange メソッド
- dispenseCash メソッド
- endDeposit メソッド（預かり金返却）

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	出金処理実行結果	String

status

ステータス	説明
"SUCCESS"	出金の成功
"BUSY"	装置動作中のため出金不可
"SHORTAGE_ERROR"	現金不足による失敗
"CASH_IN_TRAY_ERROR"	現金抜き取り待ち状態
"COMMAND_ERROR"	送信コマンドエラー
"DEVICE_ERROR"	デバイスエラー
"SYSTEM_ERROR"	システムエラー
"ILLEGAL_PARAMETER_ERROR"	パラメーター不正エラー

oncollect イベント

collectCash メソッドの実行結果を受け取ります。

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	collectCash メソッド実行結果	String

status

ステータス	説明
"SUCCESS"	出金の成功
"BUSY"	装置動作中のため出金不可
"SHORTAGE"	出金に成功したが、装置内現金が空、もしくは空に近い状態
"CASH_IN_TRAY_ERROR"	現金抜き取り待ち状態
"COMMAND_ERROR"	送信コマンドエラー
"DEVICE_ERROR"	デバイスエラー
"SYSTEM_ERROR"	システムエラー
"ERROR_NOT_SUPPORTED"	デバイス制御プログラムで制御する自動つり銭機に処理を実行した

oncommandreply イベント

sendCommand メソッドの実行結果を受け取ります。

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
data	応答データ文字列	String
status	コマンド送信結果	String

status

ステータス	説明
"SUCCESS"	送信成功
"BUSY"	動作中のため実行できない
"ILLEGAL_PARAMETER_ERROR"	パラメーター不正エラー

onstatuschange イベント

以下のメソッド実行時に収納庫ステータスの変化を受け取ります。

- readCashCounts メソッド
- beginDeposit メソッド
- pauseDeposit メソッド
- restartDeposit メソッド
- endDeposit メソッド
- dispenseChange メソッド
- dispenseCash メソッド
- collectCash メソッド
- openDrawer メソッド
- sendCommand メソッド

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	入金処理実行結果	String
st1	1 円収納庫ステータス	String
st5	5 円収納庫ステータス	String
st10	10 円収納庫ステータス	String
st50	50 円収納庫ステータス	String
st100	100 円収納庫ステータス	String
st500	500 円収納庫ステータス	String
st1000	1,000 円収納庫ステータス	String
st2000	2,000 円収納庫ステータス	String
st5000	5,000 円収納庫ステータス	String
st10000	10,000 円収納庫ステータス	String

status

ステータス	説明
"OK"	正常
"DISCREPANCY"	金額不一致の可能性あり

st1 ～ st10000

ステータス	説明
"EMPTY"	空
"NEAREMPTY"	空に近い状態
"OK"	適量
"NEARFULL"	満杯に近い状態
"FULL"	満杯

CAT オブジェクト

timeout プロパティー

以下のメソッドに、決済メディア読み取りまでのタイムアウト時間を指定します。
時間は msec 単位で指定します。

- authorizeSales メソッド
- authorizeVoid メソッド
- authorizeRefund メソッド
- authorizeCompletion メソッド
- accessDailyLog メソッド
- checkConnection メソッド

オブジェクト種類

Number

設定値	説明
1 ～ 999000	タイムアウト時間を msec 単位で指定
-1 or 0	タイムアウトなし *

*: タイムアウトを設定しない場合、使用する信用照会端末の仕様に合わせて "-1" か "0" を指定してください。

補足説明

timeout プロパティーが有効かどうかは、信用照会端末と制御ドライバーの仕様に依存します。

trainingMode プロパティー

以下のメソッドの動作モードを指定します。

- authorizeSales メソッド
- authorizeVoid メソッド
- authorizeRefund メソッド
- authorizeCompletion メソッド
- accessDailyLog メソッド
- sendCommand メソッド

オブジェクト種類

Boolean

設定値	説明
true	練習モード
false	通常モード（初期値）

authorizeSales メソッド

売上を実行します。authorizeSales メソッドの実行結果は、onauthorizesales イベントで通知されます。

構文

```
authorizeSales(data);
```

パラメーター

パラメーター：data

オブジェクト種類：Object

プロパティー	説明	オブジェクト種類
service	決済メディアの種類	String
totalAmount	売上金額合計（税込）： １～99999999	Number
amount	売上金額：１～99999999	Number
tax	税金など、その他の金額： ０～9999999	Number
sequence	信用照会端末が管理する４桁のシーケンス番号	Number
additionalSecurityInformation	売上の実行命令に追加するセキュリティ情報	String

service

設定値	説明
"credit"	クレジット
"debit"	デビット
"unionpay"	銀聯
"edy"	楽天 Edy
"id"	iD
"nanaco"	nanaco
"quicpay"	QUICPay
"suica"	交通系 IC
"waon"	WAON
"nfcpayment"	NFC Pay
"point"	ポイント
"pitapa"	ＰｉＴａＰａ
"fisc"	台湾金融カード
"qr"	QR コード決済

authorizeVoid メソッド

取消を実行します。authorizeVoid メソッドの実行結果は、onauthorizevoid イベントで通知されます。

構文

```
authorizeVoid(data);
```

パラメーター

パラメーター：data

オブジェクト種類：Object

プロパティー	説明	オブジェクト種類
service	決済メディアの種類	String
totalAmount	売上金額合計（税込）： １～99999999	Number
amount	売上金額：１～99999999	Number
tax	税金など、その他の金額： ０～9999999	Number
sequence	信用照会端末が管理する４桁のシーケンス番号	Number
additionalSecurityInformation	取消の実行命令に追加するセキュリティ情報	String

service

設定値	説明
"credit"	クレジット
"debit"	デビット
"unionpay"	銀聯
"id"	iD
"quicpay"	QUICPay
"suica"	交通系 IC
"waon"	WAON
"nfcpayment"	NFC Pay
"point"	ポイント
"pitapa"	ＰｉＴａＰａ
"fisc"	台湾金融カード
"qr"	QR コード決済

authorizeRefund メソッド

返品を実行します。authorizeRefund メソッドの実行結果は、onauthorizerefund イベントで通知されます。

構文

```
authorizeRefund(data);
```

パラメーター

パラメーター：data

オブジェクト種類：Object

プロパティー	説明	オブジェクト種類
service	決済メディアの種類	String
totalAmount	売上金額合計（税込）： １～99999999	Number
amount	売上金額： １～99999999	Number
tax	税金など、その他の金額： ０～9999999	Number
sequence	信用照会端末が管理する４桁のシーケンス番号	Number
additionalSecurityInformation	返品の実行命令に追加するセキュリティ情報	String

service

設定値	説明
"credit"	クレジット
"debit"	デビット
"unionpay"	銀聯
"id"	iD
"waon"	WAON
"nfcpayment"	NFC Pay
"pitapa"	ＰｉＴａＰａ

authorizeCompletion メソッド

承認後売上を実行します。authorizeCompletion メソッドの実行結果は、onauthorizecompletion イベントで通知されます。

構文

```
authorizeCompletion(data);
```

パラメーター

パラメーター：data

オブジェクト種類：Object

プロパティー	説明	オブジェクト種類
service	決済メディアの種類	String
totalAmount	売上金額合計（税込）： １～99999999	Number
amount	売上金額：１～99999999	Number
tax	税金など、その他の金額： ０～9999999	Number
sequence	信用照会端末が管理する４桁のシーケンス番号	Number
additionalSecurityInformation	承認後売上の実行命令に追加するセキュリティ情報	String

service

設定値	説明
"credit"	クレジット
"unionpay"	銀聯

accessDailyLog メソッド

日計を実行します。accessDailyLog メソッドの実行結果は、onaccessdailylog イベントで通知されます。

構文

```
accessDailyLog(data);
```

パラメーター

パラメーター：data

オブジェクト種類：Object

プロパティー	説明	オブジェクト種類
service	決済メディアの種類	String
sequence	信用照会端末が管理する４桁のシーケンス番号	Number
dailyLogType	実行する処理の種類	String
additionalSecurityInformation	日計処理の実行命令に追加するセキュリティ情報	String

service

設定値	説明
"credit"	クレジット
"debit"	デビット
"unionpay"	銀聯
"edy"	楽天 Edy
"id"	iD
"nanaco"	nanaco
"quicpay"	QUICPay
"suica"	交通系 IC
"waon"	WAON
"nfcpayment"	NFC Pay
"point"	ポイント
"pitapa"	ＰｉＴａＰａ
"fisc"	台湾金融カード
"qr"	QR コード決済

dailyLogType

設定値	説明
"reporting"	中間計
"settlement"	日計

sendCommand メソッド

OPOS の DirectIO メソッドを送信します。sendCommand の実行結果は、oncommandreply イベントで通知されます。

構文

```
sendCommand(data)
```

パラメーター

パラメーター：data

オブジェクト種類：Object

プロパティー	説明	オブジェクト種類
service	決済メディアの種類	String
command	DirectIO に渡すコマンド番号	Number
data	DirectIO に渡す数値データ	Number
string	DirectIO に渡す文字列データ	String
additionalSecurityInformation	OPOS の DirectIO メソッドに追加するセキュリティ情報	String

service

設定値	説明
"credit"	クレジット
"debit"	デビット
"unionpay"	銀聯
"edy"	楽天 Edy
"id"	iD
"nanaco"	nanaco
"quicpay"	QUICPay
"suica"	交通系 IC
"waon"	WAON
"nfcpayment"	NFC Pay
"point"	ポイント
"pitapa"	ＰｉＴａＰａ
"fisc"	台湾金融カード
"common"	端末共通
"qr"	QR コード決済

checkConnection メソッド

信用照会端末と OPOS ドライバーの通信状態を確認します。
checkConnection の実行結果は、oncheckConnection イベントで通知されます。

構文

```
checkConnection(data);
```

パラメーター

パラメーター：data
オブジェクト種類：Object

プロパティー	説明	オブジェクト種類
additionalSecurityInformation	OPOS ドライバーの通信状態確認命令に追加するセキュリティ情報	String

clearOutput メソッド

信用照会端末に送信した処理を中止します。
clearOutput メソッドの実行結果は、onclearOutput イベントで通知されます。

構文

```
clearOutput();
```

補足説明

中止された処理（例：authorizeSales メソッドなど）には、処理に失敗した結果が返ります。

onauthorizesales イベント

売上の実行結果を取得します。

構文

```
Function(data);
```

パラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	処理の実行結果	String
sequence	信用照会端末が管理する 4 桁のシーケンス番号	Number
service	authorizeSales メソッドの service パラメーターで指定した文字列	String
accountNumber	会員番号またはカード番号の下 4 桁	String
settledAmount	売上合計金額（税込）	Number
slipNumber	伝票番号 5 桁	String
kid	KID3 桁	String
approvalCode	承認番号 7 桁	String
transactionNumber	処理通番 6 桁	String
paymentCondition	支払い方法	String
voidSlipNumber	取消 / 返品対象の伝票番号 5 桁	String
balance	残高	Number
transactionType	取引種別	String
additionalSecurityInformation	authorizeSales メソッドで指定したセキュリティ情報	String

status

ステータス	説明
"SUCCESS"	処理が成功した
"BUSY"	動作中のため実行できない
"EXCEEDING_LIMIT" *	残高不足による失敗
"DISAGREEMENT" *	取引不一致（金額、IC 取扱通番不一致）

ステータス	説明
"ERROR_PARAMETER"	関数パラメーターの誤り
"DEVICE_ERROR"	デバイスエラー
"SYSTEM_ERROR"	システムエラー
"CENTER_ERROR" *	センター処理でエラーが発生した
"INVALID_CARD" *	無効なカード
"EX_TIMEOUT"	決済メディア読み取り待ちタイムアウト
"DEVICE_RESET" *	端末で操作をキャンセルした
"OPOS_CODE_XX"	デバイスベンダー定義のエラーコード

*:TM-DT ソフトウェア Ver.4.10 以降では発生しない。

paymentCondition

設定値	説明
"lump_sum"	一括払い
"bonus_1"	ボーナス払い パターン 1
"bonus_2"	ボーナス払い パターン 2
"bonus_3"	ボーナス払い パターン 3
"bonus_4"	ボーナス払い パターン 4
"bonus_5"	ボーナス払い パターン 5
"installment_1"	分割 パターン 1
"installment_2"	分割 パターン 2
"installment_3"	分割 パターン 3
"revolving"	リボ払い
"combination_1"	ボーナス併用 パターン 1
"combination_2"	ボーナス併用 パターン 2
"combination_3"	ボーナス併用 パターン 3
"combination_4"	ボーナス併用 パターン 4
"debit"	決済メディアの種類がデビット
"electronic_money"	決済メディアの種類が電子マネー
"other"	その他

補足説明

status が "SUCCESS" 以外の場合は、sequence と service だけに値が返ります。

onauthorizevoid イベント

取消の実行結果を取得します。

構文

```
Function(data);
```

パラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	処理の実行結果	String
sequence	信用照会端末が管理する 4 桁のシーケンス番号	Number
service	authorizeVoid メソッドの service パラメーターで指定した文字列	String
accountNumber	会員番号またはカード番号の下 4 桁	String
settledAmount	売上合計金額（税込）	Number
slipNumber	伝票番号 5 桁	String
kid	KID3 桁	String
approvalCode	承認番号 7 桁	String
transactionNumber	処理通番 6 桁	String
paymentCondition	支払い方法	String
voidSlipNumber	取消 / 返品対象の伝票番号 5 桁	String
balance	残高	Number
transactionType	取引種別	String
additionalSecurityInformation	authorizeVoid メソッドで指定したセキュリティ情報	String

status

ステータス	説明
"SUCCESS"	処理が成功した
"BUSY"	動作中のため実行できない
"EXCEEDING_LIMIT" *	残高不足による失敗
"DISAGREEMENT" *	取引不一致（金額、IC 取扱通番不一致）

ステータス	説明
"ERROR_PARAMETER"	関数パラメーターの誤り
"DEVICE_ERROR"	デバイスエラー
"SYSTEM_ERROR"	システムエラー
"CENTER_ERROR" *	センター処理でエラーが発生した
"INVALID_CARD" *	無効なカード
"EX_TIMEOUT"	決済メディア読み取り待ちタイムアウト
"DEVICE_RESET" *	端末で操作をキャンセルした
"OPOS_CODE_XX"	デバイスベンダー定義のエラーコード

*:TM-DT ソフトウェア Ver.4.10 以降では発生しない。

paymentCondition

設定値	説明
"lump_sum"	一括払い
"bonus_1"	ボーナス払い パターン 1
"bonus_2"	ボーナス払い パターン 2
"bonus_3"	ボーナス払い パターン 3
"bonus_4"	ボーナス払い パターン 4
"bonus_5"	ボーナス払い パターン 5
"installment_1"	分割 パターン 1
"installment_2"	分割 パターン 2
"installment_3"	分割 パターン 3
"revolving"	リボ払い
"combination_1"	ボーナス併用 パターン 1
"combination_2"	ボーナス併用 パターン 2
"combination_3"	ボーナス併用 パターン 3
"combination_4"	ボーナス併用 パターン 4
"debit"	決済メディアの種類がデビット
"electronic_money"	決済メディアの種類が電子マネー
"other"	その他

補足説明

status が "SUCCESS" 以外の場合は、sequence と service だけに値が返ります。

onauthorizerefund イベント

返品の実行結果を取得します。

構文

```
Function(data);
```

パラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	処理の実行結果	String
sequence	信用照会端末が管理する４桁のシーケンス番号	Number
service	authorizeRefund メソッドの service パラメーターで指定した文字列	String
accountNumber	会員番号またはカード番号の下４桁	String
settledAmount	売上合計金額（税込）	Number
slipNumber	伝票番号５桁	String
kid	KID3 桁	String
approvalCode	承認番号７桁	String
transactionNumber	処理通番６桁	String
paymentCondition	支払い方法	String
voidSlipNumber	取消 / 返品対象の伝票番号５桁	String
balance	残高	Number
transactionType	取引種別	String
additionalSecurityInformation	authorizeRefund メソッドで指定したセキュリティ情報	String

status

ステータス	説明
"SUCCESS"	処理が成功した
"BUSY"	動作中のため実行できない
"EXCEEDING_LIMIT" *	残高不足による失敗

ステータス	説明
"DISAGREEMENT" *	取引不一致（金額、IC 取扱通番不一致）
"ERROR_PARAMETER"	関数パラメーターの誤り
"DEVICE_ERROR"	デバイスエラー
"SYSTEM_ERROR"	システムエラー
"CENTER_ERROR" *	センター処理でエラーが発生した
"INVALID_CARD" *	無効なカード
"EX_TIMEOUT"	決済メディア読み取り待ちタイムアウト
"DEVICE_RESET" *	端末で操作をキャンセルした
"OPOS_CODE_XX"	デバイスベンダー定義のエラーコード

*:TM-DT ソフトウェア Ver.4.10 以降では発生しない。

paymentCondition

設定値	説明
"lump_sum"	一括払い
"bonus_1"	ボーナス払い パターン 1
"bonus_2"	ボーナス払い パターン 2
"bonus_3"	ボーナス払い パターン 3
"bonus_4"	ボーナス払い パターン 4
"bonus_5"	ボーナス払い パターン 5
"installment_1"	分割 パターン 1
"installment_2"	分割 パターン 2
"installment_3"	分割 パターン 3
"revolving"	リボ払い
"combination_1"	ボーナス併用 パターン 1
"combination_2"	ボーナス併用 パターン 2
"combination_3"	ボーナス併用 パターン 3
"combination_4"	ボーナス併用 パターン 4
"debit"	決済メディアの種類がデビット
"electronic_money"	決済メディアの種類が電子マネー
"other"	その他

補足説明

status が "SUCCESS" 以外の場合は、sequence と service だけに値が返ります。

onauthorizecompletion イベント

承認後売上の実行結果を取得します。

構文

```
Function(data);
```

パラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	処理の実行結果	String
sequence	信用照会端末が管理する４桁のシーケンス番号	Number
service	authorizeCompletion メソッドの service パラメーターで指定した文字列	String
accountNumber	会員番号またはカード番号の下４桁	String
settledAmount	売上合計金額（税込）	Number
slipNumber	伝票番号５桁	String
kid	KID3 桁	String
approvalCode	承認番号７桁	String
transactionNumber	処理通番６桁	String
paymentCondition	支払い方法	String
voidSlipNumber	取消 / 返品対象の伝票番号５桁	String
balance	残高	Number
transactionType	取引種別	String
additionalSecurityInformation	authorizeCompletion メソッドで指定したセキュリティ情報	String

status

ステータス	説明
"SUCCESS"	処理が成功した
"BUSY"	動作中のため実行できない
"EXCEEDING_LIMIT" *	残高不足による失敗

ステータス	説明
"DISAGREEMENT" *	取引不一致（金額、IC 取扱通番不一致）
"ERROR_PARAMETER"	関数パラメーターの誤り
"DEVICE_ERROR"	デバイスエラー
"SYSTEM_ERROR"	システムエラー
"CENTER_ERROR" *	センター処理でエラーが発生した
"INVALID_CARD" *	無効なカード
"EX_TIMEOUT"	決済メディア読み取り待ちタイムアウト
"DEVICE_RESET" *	端末で操作をキャンセルした
"OPOS_CODE_XX"	デバイスベンダー定義のエラーコード

*:TM-DT ソフトウェア Ver.4.10 以降では発生しない。

paymentCondition

設定値	説明
"lump_sum"	一括払い
"bonus_1"	ボーナス払い パターン 1
"bonus_2"	ボーナス払い パターン 2
"bonus_3"	ボーナス払い パターン 3
"bonus_4"	ボーナス払い パターン 4
"bonus_5"	ボーナス払い パターン 5
"installment_1"	分割 パターン 1
"installment_2"	分割 パターン 2
"installment_3"	分割 パターン 3
"revolving"	リボ払い
"combination_1"	ボーナス併用 パターン 1
"combination_2"	ボーナス併用 パターン 2
"combination_3"	ボーナス併用 パターン 3
"combination_4"	ボーナス併用 パターン 4
"debit"	決済メディアの種類がデビット
"electronic_money"	決済メディアの種類が電子マネー
"other"	その他

補足説明

status が "SUCCESS" 以外の場合は、sequence と service だけに値が返ります。

onaccessdailylog イベント

日計の実行結果を取得します。

構文

```
Function(data);
```

パラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	処理の実行結果	String
sequence	信用照会端末が管理する 4 桁のシーケンス番号	Number
service	accessDailyLog メソッドの service パラメーターで指定した文字列	String
dailyLog	日計データ	Array

status

ステータス	説明
"SUCCESS"	処理が成功した
"BUSY"	動作中のため実行できない
"ERROR_PARAMETER"	関数パラメーターの誤り
"DEVICE_ERROR"	デバイスエラー
"SYSTEM_ERROR"	システムエラー
"CENTER_ERROR" *	センター処理でエラーが発生した
"OPOS_CODE_XX"	デバイスベンダー定義のエラーコード

*:TM-DT ソフトウェア Ver.4.10 以降では発生しない。

dailyLog[index]

プロパティー	説明	オブジェクト種類
kid	KID3 桁文字列	String
salesCount	売上データ件数 0 ～ 9999999999	Number
salesAmount	売上データ金額 0 ～ 9999999999	Number

プロパティー	説明	オブジェクト種類
voidCount	取消、返品データ件数 0 ～ 9999999999	Number
voidAmount	取消、返品データ金額 0 ～ 9999999999	Number

oncheckConnection イベント

checkConnection メソッドの実行結果を受け取ります。

構文

```
Function(data);
```

パラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	処理の実行結果	String
additionalSecurityInformation	checkConnection メソッドで指定したセキュリティ情報	String

status

ステータス	説明
"SUCCESS"	処理が成功した
"BUSY"	動作中のため実行できない
"ERROR_PARAMETER"	関数パラメーターの誤り
"EX_TIMEOUT"	タイムアウト
"DEVICE_ERROR"	デバイスエラー
"SYSTEM_ERROR"	システムエラー
"OPOS_CODE_XX"	デバイスベンダー定義のエラー

onclearOutput イベント

clearOutput メソッドの実行結果を受け取ります。

構文

```
Function(data);
```

パラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	処理の実行結果	String
code	OPOS ドライバーのエラーコード	String

status

ステータス	説明
"SUCCESS"	処理が成功した
"COMMAND_ERROR"	中止できる処理が実行されていない
"ABORT_FAILURE"	処理に失敗した
"BUSY"	動作中のため実行できない
"ERROR_PARAMETER"	関数パラメーターの誤り
"EX_TIMEOUT"	タイムアウト
"DEVICE_ERROR"	デバイスエラー
"SYSTEM_ERROR"	システムエラー
"OPOS_CODE_XX"	デバイスベンダー定義のエラー

code

code が "ABORT_FAILURE" の場合、処理に失敗した理由を示す OPOS ドライバーのエラーコードが入ります。

code が "ABORT_FAILURE" 以外の場合、""（空文字）が入ります。

oncommandreply イベント

sendCommand メソッドの実行結果を受け取ります。

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	DirectIO の戻り値	String
sequence	信用照会端末が管理する 4 桁のシーケンス番号	Number
service	sendCommand メソッドの service パラメーターで指定した決済メディアの種類	String
command	DirectIO のコマンド番号	Number
data	DirectIO の数値データ	Number
string	DirectIO の文字列データ	String
accountNumber	会員番号またはカード番号の下 4 桁	String
settledAmount	売上合計金額 (税込)	Number
slipNumber	伝票番号 5 桁	String
transactionNumber	処理通番 6 桁	String
paymentCondition	支払い方法	String
balance	残高	Number
additionalSecurityInformation	拡張結果情報	String

status

ステータス	説明
"SUCCESS"	送信成功
"BUSY"	動作中のため実行できない
"ERROR_PARAMETER"	関数パラメーターの誤り
"EX_TIMEOUT"	タイムアウト
"DEVICE_ERROR"	デバイスエラー

ステータス	説明
"SYSTEM_ERROR"	システムエラー
"OPOS_CODE_XX"	デバイスベンダー定義のエラー

paymentCondition

支払い方法	説明
"lump_sum"	一括払い
"bonus_1"	ボーナス払い パターン１
"bonus_2"	ボーナス払い パターン２
"bonus_3"	ボーナス払い パターン３
"installment_1"	分割 パターン１
"installment_2"	分割 パターン２
"rebolving"	リボ払い
"combination_1"	ボーナス併用 パターン１
"combination_2"	ボーナス併用 パターン２
"debit"	決済メディアの種類がデビット
"electronic_money"	決済メディアの種類が電子マネー
"other"	その他

ondirectio イベント

OPOS ドライバーの DirectIOEvent を受け取ります。

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
eventnumber	DirectIOEvent のコマンド番号	Number
data	DirectIOEvent の数値データ	Number
string	DirectIOEvent の文字列データ	String

onstatusupdate イベント

OPOS ドライバーの StatusUpdateEvent を受け取ります。

StatusUpdateEvent は、信用照会端末の状態が変化した際に発生します。詳細は、OPOS ドライバーのマニュアルを参照してください。

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	StatusUpdateEvent の戻り値	Number

status

ステータス	説明
SUE_LOGSTATUS_OK	取引ログは充分な空き容量があります。
SUE_LOGSTATUS_NEARFULL	取引ログはフルに近い状態です。
SUE_LOGSTATUS_FULL	取引ログは一杯です。
SUE_POWER_ONLINE	デバイスは電源オンでかつレディ状態です。
SUE_POWER_OFF_OFFLINE	デバイスは電源オフ、またはオフライン状態です。

Display オブジェクト

createWindow メソッド

ウィンドウ設定を命令バッファーに追加します。
ウィンドウは最大で 4 つまで設定可能です。

構文

```
createWindow(number, x, y, width, height, scrollMode);
```

パラメーター

number

設定値	説明
1 ～ 4	ウィンドウ番号を指定

x

設定値	説明
1 ～ 44	x 座標原点位置を指定

y

設定値	説明
1 ～ 19	y 座標原点位置を指定

width

設定値	説明
1 ～ 44	ウィンドウの幅を指定

height

設定値	説明
1 ～ 19	ウィンドウの高さを指定

scrollMode

設定値	説明
SCROLL_OVERWRITE	• 表示位置が上段右端の時に文字表示すると、表示位置を下段左端に移動する。 • 表示位置が下段右端の時に文字表示すると、表示位置を上段左端に移動する。 • 表示位置の移動先に表示されている文字は、上書きされる。
SCROLL_VERTICAL	• 表示位置が上段右端の時に文字表示すると、表示位置を下段左端に移動する。 • 表示位置が下段右端の時に文字表示すると、下段の表示文字を上段へ移動させて下段の表示文字を消去し、表示位置を下段左端へ移動する。
SCROLL_HORIZONTAL	• 表示位置が右端の時に文字表示すると、行全体を 1 文字分左へ移動させ、右端に新しく文字を表示する。 • 改行は行わない。

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

補足説明

追加でウィンドウを設定する際は、設定済みのウィンドウとカスタマーディスプレイ上の設定範囲が重ならないように注意してください。

destroyWindow メソッド

createWindow メソッドで設定した任意のウィンドウ番号を指定し、指定したウィンドウの破棄設定を命令バッファーに追加します。

構文

```
destroyWindow(number);
```

パラメーター

number

設定値	説明
1 ～ 4	ウィンドウ番号を指定

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

setCurrentWindow メソッド

カレントウィンドウの変更設定を命令バッファーに追加します。
指定ウィンドウをカレントウィンドウとし、カーソルをカレントウィンドウの原点位置に移動させます。

構文

```
setCurrentWindow(number);
```

パラメーター

number

設定値	説明
1 ～ 4	ウィンドウ番号を指定

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

clearWindow メソッド

カレントウィンドウ内の表示消去を命令バッファーに追加します。

構文

```
clearWindow();
```

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

addCreateScreen メソッド

ベースウィンドウの設定を命令バッファーに追加します。
本 API は、カスタマーディスプレイが持つベースウィンドウのレイアウト設定を呼び出します。

構文

```
addCreateScreen(mode);
```

パラメーター

mode

ベースウィンドウの表示領域と、画面構成（テキスト / イメージ）の組み合わせを指定します。

設定値	設置方法	領域（桁 × 行）	画面構成
LAYOUT_MODE_1	横置き	20 x 2	【上下分割】 上下均等に分割。 上：テキスト、下：イメージ
LAYOUT_MODE_2	横置き	20 x 2	【上下分割】 上下均等に分割。 上：イメージ、下：テキスト
LAYOUT_MODE_3	横置き	20 x 2	【中央表示】 テキストのみを中央に表示。 分割しない。
LAYOUT_MODE_4	横置き	32 x 4	【上下分割】 上下均等に分割。 上：テキスト、下：イメージ
LAYOUT_MODE_5	横置き	32 x 4	【上下分割】 上下均等に分割。 上：イメージ、下：テキスト
LAYOUT_MODE_6	横置き	32 x 4	【中央表示】 テキストのみを中央に表示。 分割しない。
LAYOUT_MODE_7	横置き	42 x 8	【上下分割】 上下均等に分割。 上：テキスト、下：イメージ
LAYOUT_MODE_8	横置き	42 x 8	【上下分割】 上下均等に分割。 上：イメージ、下：テキスト
LAYOUT_MODE_9	横置き	42 x 8	【中央表示】 テキストのみを中央に表示。 分割しない。

設定値	設置方法	領域（桁 × 行）	画面構成
LAYOUT_MODE_10	横置き	32 x 3	【上下分割】 上下均等に分割。 上：テキスト、下：イメージ
LAYOUT_MODE_11	横置き	32 x 3	【上下分割】 上下均等に分割。 上：イメージ、下：テキスト
LAYOUT_MODE_12	横置き	32 x 3	【中央表示】 テキストのみを中央に表示。 分割しない。
LAYOUT_MODE_13	横置き	32 x 2	【上下分割】 上下均等に分割。 上：テキスト、下：イメージ
LAYOUT_MODE_14	横置き	32 x 2	【上下分割】 上下均等に分割。 上：イメージ、下：テキスト
LAYOUT_MODE_15	横置き	32 x 2	【中央表示】 テキストのみを中央に表示。 分割しない。

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

例外

戻り値	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

本 API を実行すると、カスタマーディスプレイに表示しているものは全て消去されます。

addCreateScreenCustom メソッド

表示モードの設定を命令バッファーに追加します。
本 API は、カスタマーディスプレイが持つベースウィンドウの画面構成に幅と高さを指定できます。

構文

```
addCreateScreenCustom(mode, column, row);
```

パラメーター

mode

ベースウィンドウの画面構成（テキスト / イメージ）を指定します。

設定値	設置方法	画面構成
LANDSCAPE_LAYOUT_MODE_1	横置き	【フルスクリーン】 全面にテキスト / イメージを表示。 分割しない。
LANDSCAPE_LAYOUT_MODE_2	横置き	【左右分割】 左右均等に分割。 左：イメージ、右：テキスト
LANDSCAPE_LAYOUT_MODE_3	横置き	【左右分割】 左右均等に分割。 左：テキスト、右：イメージ
LANDSCAPE_LAYOUT_MODE_4	横置き	【上下分割】 上下均等に分割。 上：テキスト、下：イメージ
LANDSCAPE_LAYOUT_MODE_5	横置き	【上下分割】 上下均等に分割。 上：イメージ、下：テキスト
PORTRAIT_LAYOUT_MODE_1	縦置き	【フルスクリーン】 全面にテキスト / イメージを表示。 分割しない。
PORTRAIT_LAYOUT_MODE_2	縦置き	【上下分割】 上下均等に分割。 上：テキスト、下：イメージ
PORTRAIT_LAYOUT_MODE_3	縦置き	【上下分割】 上下均等に分割。 上：イメージ、下：テキスト
PORTRAIT_LAYOUT_MODE_4	縦置き	【上下分割 1:2】 上下 1：2 に分割。 上：テキスト、下：イメージ

設定値	設置方法	画面構成
PORTRAIT_LAYOUT_MODE_5	縦置き	【上下分割 2:1】 上下 2：1 に分割。 上：イメージ、下：テキスト
PORTRAIT_LAYOUT_MODE_6	縦置き	【上下分割 2:1】 上下 2：1 に分割。 上：テキスト、下：イメージ
PORTRAIT_LAYOUT_MODE_7	縦置き	【上下分割 1:2】 上下 1：2 に分割。 上：イメージ、下：テキスト

column

ベースウィンドウ表示領域の幅（桁数）を指定します。

設定値	説明
1 ～ 44 の整数値	横置き設置時の有効指定範囲
1 ～ 22 の整数値	縦置き設置時の有効指定範囲

row

ベースウィンドウ表示領域の高さ（行数）を指定します。

設定値	説明
1 ～ 13 の整数値	横置き設置時の有効指定範囲
1 ～ 19 の整数値	縦置き設置時の有効指定範囲

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

例外

戻り値	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- 本 API を実行すると、カスタマーディスプレイに表示しているものは全て消去されます。
- mode に LANDSCAPE_LAYOUT_MODE を指定したとき、row に 14 ～ 19 を指定すると例外が発生します。
- mode に PORTRAIT_LAYOUT_MODE を指定したとき、column に 23 ～ 44 を指定すると例外が発生します。
- column に 45 以上、row に 20 以上の値を指定すると無視されます。

createTextArea メソッド

ウィンドウ設定を命令バッファーに追加します。
ウィンドウは最大で 4 つまで設定可能です。

構文

```
createTextArea(number, x, y, width, height, scrollMode);
```

パラメーター

number

設定値	説明
1 ～ 4 の整数値	ウィンドウ番号を指定

x

設定値	説明
1 ～ 44 の整数値	ウィンドウ原点のカスタマーディスプレイ上の x 座標を指定

y

設定値	説明
1 ～ 19 の整数値	ウィンドウ原点のカスタマーディスプレイ上の y 座標を指定

width

設定値	説明
1 ～ 44 の整数値	ウィンドウの幅を指定

height

設定値	説明
1 ～ 19 の整数値	ウィンドウの高さを指定

scrollMode

設定値	説明
SCROLL_OVERWRITE	• 表示位置が上段右端の時に文字表示すると、表示位置を下段左端に移動する。 • 表示位置が下段右端の時に文字表示すると、表示位置を上段左端に移動する。 • 表示位置の移動先に表示されている文字は、上書きされる。

設定値	説明
SCROLL_VERTICAL	• 表示位置が上段右端の時に文字表示すると、表示位置を下段左端に移動する。 • 表示位置が下段右端の時に文字表示すると、下段の表示文字を上段へ移動させて下段の表示文字を消去し、表示位置を下段左端へ移動する。
SCROLL_HORIZONTAL	• 表示位置が右端の時に文字表示すると、行全体を 1 文字分左へ移動させ、右端に新しく文字を表示する。 • 改行は行わない。

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

例外

戻り値	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

追加でウィンドウを設定する際は、設定済みのウィンドウとカスタマーディスプレイ上の設定範囲が重ならないように注意してください。

destroyTextArea メソッド

定義済みウィンドウの破棄設定を命令バッファーに追加します。

構文

```
destroyTextArea(number);
```

パラメーター

number

設定値	説明
1 ～ 4 の整数値	破棄するウィンドウ番号を指定

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

例外

戻り値	オブジェクト種類
Parameter " ... " is invalid	Error

setCurrentTextArea メソッド

カレントウィンドウの変更を命令バッファーに追加します。
指定ウィンドウをカレントウィンドウとし、カーソルをカレントウィンドウの原点位置に移動させます。

構文

```
setCurrentTextArea(number);
```

パラメーター

number

設定値	説明
１～４の整数値	移動先のウィンドウ番号を指定

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

例外

戻り値	オブジェクト種類
Parameter " ... " is invalid	Error

clearTextArea メソッド

カレントウィンドウの表示消去を命令バッファーに追加します。

構文

```
clearTextArea;
```

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

setCursorPosition メソッド

座標を指定し、カーソル位置設定を命令バッファーに追加します。

構文

```
setCursorPosition(x, y);
```

パラメーター

x

設定値	説明
1 ～ 44	x 座標を指定

y

設定値	説明
1 ～ 19	y 座標を指定

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

補足説明

- DM-D70（標準モード）では、setCurrentTextArea メソッドで指定したカレントウィンドウ内で、左上を原点としたカーソル位置を設定します。
  setCurrentTextArea メソッドでウィンドウを指定していない場合、ベースウィンドウ内で左上を原点としたカーソル位置を設定します。
- DM-D70（桁行固定モード）では、ベースウィンドウ内で左上を原点としたカーソル位置を設定します。
- 使用するカスタマーディスプレイやその表示モード、設置方法によって x, y パラメーターに指定できる値が異なります。

	x	y
DM-D30	1 ～ 20	1 ～ 2
DM-D110		
DM-D70（桁行固定モード）		
DM-D70（標準モード：横置き）	1 ～ 44	1 ～ 13
DM-D70（標準モード：縦置き）	1 ～ 22	1 ～ 19

moveCursorPosition メソッド

カレントウィンドウ内でのカーソル移動先を指定し、カーソルの位置設定を命令バッファーに追加します。

構文

```
moveCursorPosition(position);
```

パラメーター

position

設定値	説明
MOVE_TOP_LEFT	上段左端へ移動
MOVE_TOP_RIGHT	上段右端へ移動
MOVE_BOTTOM_LEFT	下段左端へ移動
MOVE_BOTTOM_RIGHT	下段右端へ移動

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

補足説明

- DM-D70（標準モード）では、setCurrentTextArea メソッドで指定したカレントウィンドウの原点から、指定された位置へカーソルを移動します。
  setCurrentTextArea メソッドでウィンドウを指定していない場合、ベースウィンドウの原点からカーソルを移動します。
- DM-D70（桁行固定モード）では、ベースウィンドウの原点から指定された位置へカーソルを移動します。

setCursorType メソッド

カーソルの表示設定を命令バッファーに追加します。

構文

```
setCursorType(type);
```

パラメーター

type

設定値	説明
CURSOR_NONE	表示なし
CURSOR_UNDERLINE	アンダーライン表示

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

addText メソッド

カスタマーディスプレイに表示する文字の表示設定を命令バッファーに追加します。

構文

- ❏ addText(data);
- ❏ addText(data, lang);
- ❏ addText(data, x, y);
- ❏ addText(data, x, y, lang);
- ❏ addText(data, x, y, lang, r, g, b);

パラメーター

data

設定値	説明
data	表示文字を文字列で指定

lang

設定値	説明
"en"（初期値）	英語
"ja"	日本語（カナ）
"mul"	複数言語（UTF-8）
"undefined"	使用しない。

x

設定値	説明
1 ～ 44	表示位置の x 座標を指定
"undefined"	使用しない。

y

設定値	説明
1 ～ 19	表示位置の y 座標を指定
"undefined"	使用しない。

r

設定値	説明
0 ～ 255 の整数値	表示する文字列の赤色値を指定
"undefined"	使用しない。

g

設定値	説明
0 ～ 255 の整数値	表示する文字列の緑色値を指定
"undefined"	使用しない。

b

設定値	説明
0 ～ 255 の整数値	表示する文字列の青色値を指定
"undefined"	使用しない。

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

例外

戻り値	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- 表示位置の座標指定を省略すると、addText メソッド実行時のカーソル位置座標が指定されます。
- 使用するカスタマーディスプレイやその表示モード、設置方法によって x, y パラメーターに指定できる値が異なります。

	x	y
DM-D30	1 ～ 20	1 ～ 2
DM-D110		
DM-D70（桁行固定モード）		
DM-D70（標準モード：横置き）	1 ～ 44	1 ～ 13
DM-D70（標準モード：縦置き）	1 ～ 22	1 ～ 19

- x, y パラメーター両方に設定値を指定してください。
  片方しか指定しなかった場合、例外は発生しませんが何も表示されません。
- r, g, b パラメーターは、RGB565 形式で指定してください。

- 表示する文字列の色を変えるときは、r, g, b 全てのパラメーターに設定値を指定してください。
- r, g, b 全てのパラメーターに設定値を指定すると、x, y パラメーターで指定した位置座標から文字色が変わります。
- lang パラメーターに "mul" を指定すると、文字色はカスタマーディスプレイが持つデフォルト設定に戻ります。
  lang パラメーターに "mul" を指定して r, g, b パラメーターを指定すると、lang の後に r, g, b が処理されるため、r, g, b で指定した文字色が適用されます。

addReverseText メソッド

カスタマーディスプレイに表示する白黒反転文字の表示設定を命令バッファーに追加します。

構文

- ❏ addReverseText(data);
- ❏ addReverseText(data, lang);
- ❏ addReverseText(data, x, y);
- ❏ addReverseText(data, x, y, lang);

パラメーター

data

設定値	説明
data	表示文字を文字列で指定

lang

設定値	説明
"en"（初期値）	英語
"ja"	日本語（ｶﾅ）

x

設定値	説明
1 ～ 20	表示位置の x 座標を指定

y

設定値	説明
1 ～ 2	表示位置の y 座標を指定

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

補足説明

表示位置の座標指定を省略すると、addReverseText メソッド実行時のカーソル位置座標が指定されます。

addMarquee メソッド

マーキー表示設定を命令バッファーに追加します。

マーキー表示設定は、createWindow メソッドで指定したウィンドウのスクロールモードの影響を受けず、1 行の文字列を水平スクロールモードで表示します。

構文

```
addMarquee(data, format, unitWait, repeatWait, repeatCount,
lang);
```

パラメーター

data

設定値	説明
data	表示文字を文字列で指定

format

設定値	説明
MARQUEE_WALK	ウィンドウ右端から文字を表示
MARQUEE_PLACE	ウィンドウ左端から文字を表示

unitWait

設定値	説明
0 ～ 2000	1 文字ごとの表示間隔を msec 単位で指定

repeatWait

設定値	説明
100 ～ 2000	文字の繰り返し間隔を msec 単位で指定

repeatCount

設定値	説明
0	無制限に繰り返す
1 ～ 127	文字の繰り返し回数を指定

lang

設定値	説明
"en"（初期値）	英語
"ja"	日本語（カナ）

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

補足説明

- DM-D30/DM-D70 の場合、repeatCount には 0 しか指定できません。
- DM-D30/DM-D70 を制御する場合、搭載されている LCD の特性により、文字表示切り替え間隔が短いとマーキー表示が乱れるおそれがあります。
  unitWait と repeatWait には、以下の値を推奨します。
  DM-D30：600 msec 以上
  DM-D70：100 msec 以上

setBlink メソッド

カスタマーディスプレイの点滅設定を命令バッファーに追加します。

構文

```
setBlink(interval);
```

パラメーター

interval

設定値	説明
0 ～ 12700	点滅時間を msec 単位で指定

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

補足説明

- 0 を指定すると点灯状態になります。
- 設定値は 50 msec 単位で切り上げられます。
  例：1 ⇒ 50、51 ⇒ 100、101 ⇒ 150

setBrightness メソッド

カスタマーディスプレイの輝度設定を命令バッファーに追加します。

構文

```
setBrightness(brightness);
```

パラメーター

brightness

設定値	説明
BRIGHTNESS_20	輝度値を 20%に設定
BRIGHTNESS_40	輝度値を 40%に設定
BRIGHTNESS_60	輝度値を 60%に設定
BRIGHTNESS_100	輝度値を 100%に設定

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

addBackgroundColor メソッド

背景色の指定を命令バッファーに追加します。
テキストエリアで指定した行の、先頭から末尾までの背景色を変更できます。

構文

```
addBackgroundColor(row, r, g, b);
```

パラメーター

row

設定値	説明
1 ～ 19 の整数値	指定行の背景色を変更
EVEN_ROWS	偶数行全ての背景色を変更
ODD_ROWS	奇数行全ての背景色を変更
ALL_ROWS	全ての行の背景色を変更

r

設定値	説明
0 ～ 255 の整数値	背景色の赤色値を指定

g

設定値	説明
0 ～ 255 の整数値	背景色の緑色値を指定

b

設定値	説明
0 ～ 255 の整数値	背景色の青色値を指定

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

例外

戻り値	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- 背景色は、テキストエリアにのみ適用できます。
- r, g, b パラメーターは、RGB565 形式で指定してください。
- r, g, b 全てのパラメーターに設定値を指定してください。
- 複数の行に背景色を適用したいときは、本 API を複数回実行してください。
- 複数のウィンドウで個別に背景色を指定したいときは、本 API を実行する前に setCurrentTextArea メソッドでウィンドウを定義してください。
  事前にウィンドウを定義しておくと、ウィンドウ単位で個別に背景色を指定できます。
  背景色を変えたい位置でウィンドウを分けることで、行の途中で背景色を変えることができます。

addStartSlideShow メソッド

スライドショーの開始を命令バッファーに追加します。

構文

```
addStartSlideShow(interval);
```

パラメーター

interval

設定値	説明
200 ～ 51000 の整数値	スライドショーの切り替え時間を msec 単位で指定

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

例外

戻り値	オブジェクト種類
Parameter " ... " is invalid	Error

addStopSlideShow メソッド

スライドショーの停止を命令バッファーに追加します。

構文

```
addStopSlideShow();
```

戻り値

戻り値	説明
Display オブジェクト	Display

addDownloadImage メソッド

addRegisterDownloadImage メソッドで登録したイメージの表示を命令バッファーに追加します。

構文

```
addDownloadImage (key1, key2, dotX, dotY, width, height);
```

パラメーター

key1

設定値	説明
0 ～ 255 の整数値	addRegisterDownloadImage メソッドで指定したキーコード 1 を指定

key2

設定値	説明
0 ～ 255 の整数値	addRegisterDownloadImage メソッドで指定したキーコード 2 を指定

dotX

カスタマーディスプレイの左上を原点とした、イメージ表示位置の x 座標を指定します。

設定値	説明
0 ～ 799 の整数値	横置き設置時の有効指定範囲
0 ～ 479 の整数値	縦置き設置時の有効指定範囲

dotY

カスタマーディスプレイの左上を原点とした、イメージ表示位置の y 座標を指定します。

設定値	説明
0 ～ 479 の整数値	横置き設置時の有効指定範囲
0 ～ 799 の整数値	縦置き設置時の有効指定範囲

width

設定値	説明
0 ～ 1440 の整数値	イメージの横幅表示サイズ（ドット単位）を指定

height

設定値	説明
0 ～ 1440 の整数値	イメージの高さ表示サイズ（ドット単位）を指定

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

例外

戻り値	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- addCreateScreenCustom メソッドでレイアウトを指定しているときに使用できます。
- 元イメージのサイズで表示するには、width, height パラメーター両方に 0 を指定してください。
- 表示領域に収まらない場合、イメージは表示されません。
- 本 API で指定したイメージは、API 実行時に表示中のテキストやイメージより前面に表示されます。
- 透過イメージを利用するときは、png 形式のデータを使用してください。

addRegisterDownloadImage メソッド

HTML5 Canvas で描画したイメージデータの登録を命令バッファーに追加します。
指定したイメージデータをカスタマーディスプレイのダウンロードグラフィックス領域に登録し、addDownloadImage メソッドで呼び出すことができます。
HTML5 Canvas の RGBA フルカラー画像の指定範囲を、イメージデータに変換します。
画像の 1 ピクセルがカスタマーディスプレイの 1 ドットに相当し、透明色が含まれている場合は、画像の背景を白とみなします。

構文

```
addRegisterDownloadImage(data, key1, key2);
```

パラメーター

context

設定値	説明
context	HTML5 Canvas の 2D コンテキストを指定

key1

設定値	説明
48 ～ 57、65 ～ 90、97 ～ 122 の整数値	キーコード 1 を指定

key2

設定値	説明
48 ～ 57、65 ～ 90、97 ～ 122 の整数値	キーコード 2 を指定

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

例外

戻り値	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- 本 API は他の API と同時に使用できません。
  本 API 実行後は、コールバックが返るまで他の API を実行しないでください。

- 本 API は 1 回の実行に対して 1 回の send メソッドを実行し、1 枚ずつイメージ登録してください。
- 本 API 実行後の send メソッド実行結果で、onreceive イベントに成功が返らなかった場合、プリンターを再起動してください。
  システム環境によっては、復旧処理に時間がかかる可能性があります。
- カスタマーディスプレイの電源が切れると、登録したイメージは消去されます。
- 登録するイメージのデータサイズが大きくなるにつれて、処理時間が長くなります。（最大 320 秒）
- 登録するイメージのデータサイズが大きく、通信環境が悪い場合、本 API で登録できない可能性があります。
  本 API で登録できなかった場合は、専用の Utility ソフトウェアを使用してください。
  専用の Utility ソフトウェアについては、各カスタマーディスプレイの詳細取扱説明書を参照してください。
- 指定したキーコードにイメージが登録されていた場合、上書きして登録します。
- YCbCr422 か YCbCr420 のピクセルフォーマットで表現されたイメージデータが使用できます。
- イメージサイズの上限は 1.5MB ですが、100KB 程度のイメージサイズにしてください。
- 登録できるイメージの合計データサイズは 1.5MB です。
  本 API は、イメージの形式やピクセル数をカスタマーディスプレイが処理できるように変換してから登録します。変換処理によりイメージサイズが増えることがあるので、登録するイメージの合計サイズは余裕を持たせてください。なお、ダウンロードグラフィックス領域の空き容量不足により登録できなかった場合でも、エラー検知できません。
- 登録するイメージの幅と高さ（ピクセル）は、偶数にしてください。
- 透過イメージを利用するときは、png 形式のデータを使用してください。
- カスタマーディスプレイのイメージエリアサイズに合わせたイメージを使用してください。
  イメージエリアサイズより大きいイメージを使用すると、イメージエリアサイズでイメージがトリミングされます。
- 異なるドメインからダウンロードした画像が含まれた HTML5 Canvas は表示できません。
  JavaScript の同一出身ポリシーにより、セキュリティーエラーが発生します。

addNVImage メソッド

カスタマーディスプレイの NV グラフィックス領域に登録されたイメージの表示を命令バッファーに追加します。

構文

```
addNVImage (key1, key2, dotX, dotY, width, height);
```

パラメーター

key1

設定値	説明
0 ～ 255 の整数値	NV グラフィックスのキーコード 1 を指定

key2

設定値	説明
0 ～ 255 の整数値	NV グラフィックスのキーコード 2 を指定

dotX

カスタマーディスプレイの左上を原点とした、イメージ表示位置の x 座標を指定します。

設定値	説明
0 ～ 799 の整数値	横置き設置時の有効指定範囲
0 ～ 479 の整数値	縦置き設置時の有効指定範囲

dotY

カスタマーディスプレイの左上を原点とした、イメージ表示位置の y 座標を指定します。

設定値	説明
0 ～ 479 の整数値	横置き設置時の有効指定範囲
0 ～ 799 の整数値	縦置き設置時の有効指定範囲

width

設定値	説明
0 ～ 1440 の整数値	イメージの横幅表示サイズ（ドット単位）を指定

height

設定値	説明
0 ～ 1440 の整数値	イメージの高さ表示サイズ（ドット単位）を指定

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

例外

戻り値	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- addCreateScreenCustom メソッドでレイアウトを指定しているときに使用できます。
- 元イメージのサイズで表示するには、width, height パラメーター両方に 0 を指定してください。
- 表示領域に収まらない場合、イメージは表示されません。
- 本 API で指定したイメージは、API 実行時に表示中のテキストやイメージより前面に表示されます。
- 透過イメージを利用するときは、png 形式のデータを使用してください。
- カスタマーディスプレイの NV グラフィックス領域へのイメージ登録は、専用の Utility ソフトウェアを使用します。
  詳細は、各カスタマーディスプレイの詳細取扱説明書を参照してください。

addClearImage メソッド

表示中のイメージ消去を命令バッファーに追加します。

構文

```
addClearImage();
```

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

補足説明

本 API ではイメージエリアに表示したイメージを消去します。

テキストエリアに表示したイメージを消去するには、reset メソッドでカスタマーディスプレイを初期化してください。

addSymbol メソッド

2 次元シンボルの表示を命令バッファーに追加します。

構文

```
addSymbol(data, type, level, width, height, dotX, dotY,
quietZone);
```

パラメーター

data

2 次元シンボルデータを文字列で指定します。

type で指定する 2 次元シンボルの規格に従った文字列を指定してください。

種類	説明
QR Code Model 1	文字列をシフト JIS に変換後、エスケープシーケンスの処理を行い、データの種類を以下の中から選択してエンコードします。 • 数字：0 ～ 9 • 英数字：0 ～ 9, A ～ Z, スペース , $, %, *, +, -, ., /, : • 漢字：シフト JIS で表現できる漢字 • 8 ビットバイトデータ：0x00 ～ 0xff
QR Code Model 2	

文字列で表現できないバイナリーデータを指定する場合、以下のエスケープシーケンスで指定します。

文字列	説明
\xnn	コントロールコード（nn は 16 進数で指定）
\\	バックスラッシュ

type

設定値	説明
SYMBOL_QRCODE_MODEL_1	QR Code Model 1 を指定
SYMBOL_QRCODE_MODEL_2	QR Code Model 2 を指定

level

設定値	説明
LEVEL_L	エラー訂正レベル L を指定
LEVEL_M	エラー訂正レベル M を指定
LEVEL_Q	エラー訂正レベル Q を指定
LEVEL_H	エラー訂正レベル H を指定
LEVEL_DEFAULT(LEVEL_M)	既定値 (エラー訂正レベル M) を指定

width

設定値	説明
1 ～ 255 の整数値	モジュールの幅（ドット単位）を指定 QR Code の有効指定範囲 3 ～ 16 の整数値（初期値：3）

height

設定値	説明
1 ～ 255 の整数値	モジュールの高さ（ドット単位）を指定 QR Code の有効指定範囲なし。 QR Code は正方形のため、高さの指定は無視されます。

dotX

2 次元シンボルを表示する x 座標の原点（ドット単位）を指定します。

設定値	説明
0 ～ 799 の整数値	横置き設置時の有効指定範囲
0 ～ 479 の整数値	縦置き設置時の有効指定範囲

dotY

2 次元シンボルを表示する y 座標の原点（ドット単位）を指定します。

設定値	説明
0 ～ 479 の整数値	横置き設置時の有効指定範囲
0 ～ 799 の整数値	縦置き設置時の有効指定範囲

quietZone

設定値	説明
true	クワイエットゾーン付与を有効に指定
false	クワイエットゾーン付与を無効に指定

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

例外

戻り値	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- addSymbol メソッドの指定は「行の先頭」で行ってください。
- addCreateScreenCustom メソッドでレイアウトを指定しているときに使用できます。
- テキストやイメージの表示と重なった場合、2 次元シンボルが最前面に表示されます。
- data で指定した 2 次元シンボルデータが、type で指定する 2 次元シンボルの種類に従っていない場合、2 次元シンボルは表示されません。
- クワイエットゾーン付与を有効にすると、2 次元シンボルの上下左右に width で指定した値の 4 倍ドット白余白（クワイエットゾーン）が付きます。
- クワイエットゾーン付与を有効にする場合、dotX, dotY にはクワイエットゾーン左上を原点とする値を指定してください。

addClearSymbol メソッド

表示中の 2 次元シンボル消去を命令バッファーに追加します。

構文

```
addClearSymbol();
```

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

補足説明

表示中の 2 次元シンボルがイメージに重なっていた場合、重なっていた部分のイメージは 2 次元シンボルと共に消去されます。

showClock メソッド

時刻の表示を命令バッファーに追加します。

時刻表示を実行すると、ウィンドウ内に表示中の文字はすべて消去されます。また、時刻表示中に他の命令を送信すると、時刻表示を終了します。

構文

```
showClock();
```

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

addCommand メソッド

ESC/POS コマンドを命令バッファーに追加します。

構文

```
addCommand(data);
```

パラメーター

data

設定値	説明
data	ESC/POS コマンドを文字列で指定

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

補足説明

- ESC/POS コマンドの詳細は以下の URL を参照してください。
  www.epson-biz.com/pos/reference_ja/
- Epson ePOS SDK は本 API で送信するコマンドをチェックしません。
  Epson ePOS SDK の動作を妨げるようなコマンドを送信した場合、他の API の動作やステータスの値が不正になることがあります。
  ESC/POS コマンドやカスタマーディスプレイの仕様を十分に理解した上で使用してください。

send メソッド

命令バッファー内のデータをプリンターへ送信します。send メソッドの実行結果は、onreceive イベントのコールバック関数に渡されます。

構文

```
send();
```

reset メソッド

カスタマーディスプレイの初期化設定を命令バッファーに追加します。
reset メソッドの実行後は、以下の状態になります。

- 表示文字や登録したウィンドウは全て破棄されます。
- カーソルの設定は初期状態に戻り、カーソルの位置はカスタマーディスプレイ原点へ移動します。
- 表示点滅と表示輝度の設定が初期状態に戻ります。

構文

```
reset();
```

戻り値

戻り値	オブジェクト種類
Display オブジェクト	Display

onreceive イベント

send メソッドで送信した命令の実行結果を受け取ります。

構文

```
Function(response);
```

コールバック関数のパラメーター

パラメーター：response

オブジェクト種類：Object

response オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
success	命令の実行結果	Boolean
code	エラーコード	String

success

値	説明
true	処理に成功
false	処理に失敗

code

値	説明
'EDSP_NOT_FOUND'	デバイスが見つからない
'EDSP_NOT_OPEN'	デバイスのオープンに失敗した
'EDSP_INVALID_WINDOW'	未登録のウィンドウが指定された
'EDSP_UNRECOVERABLE'	接続中のプリンターで復帰不可能エラーが発生した。
'EDSP_CUTTER'	接続中のプリンターでオートカッターエラーが発生した。
'EDSP_MECHANICAL'	接続中のプリンターでメカニカルエラーが発生した。
'EDSP_COVER_OPEN'	接続中のプリンターのカバーが開いている。
'EDSP_REC_EMPTY'	接続中のプリンターの用紙なし。
'EDSP_AUTOMATICAL'	接続中のプリンターで自動復帰エラーが発生した。
'EDSP_SCHEMA_ERROR'	XML 構造が間違っている
'EX_BADPORT'	デバイスとの内部通信エラーが発生した

値	説明
'EX_TIMEOUT'	デバイスとの通信でタイムアウトエラーが発生した
'EX_INVALID_VALUE'	パラメーター不正を検出した
'TooManyRequests'	ディスプレイ表示データ送信数が許容量を超えた

補足説明

- 通信異常が発生した場合、onreceive イベントを通知しません。
  切断イベントを通知できるように、onreconnecting イベント、onreconnect イベント、ondisconnect イベントを実装してください。
- 通信異常が発生して復旧しないと判断したときは、ondisconnect イベントを通知します。
  この場合は onreceive イベントを通知しないため、ondisconnect イベントでディスプレイ表示の失敗を検出してください。

Keyboard オブジェクト

setPrefix メソッド

キーボード入力を文字列としてまとめて受け取る際、文字列の始まりと判断するためのキーコードを設定します。

構文

```
setPrefix(data);
```

パラメーター

data

設定値	説明
data	キーコードの配列を指定

補足説明

指定するキーコードは、Key Code 一覧を参照してください。

onkeypress イベント

キー押下イベントを受け取ります。

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
keycode	キーコード	Number
ascii	キーコードに対応する文字	String

補足説明

キーコードに対応する文字がない場合、ascii には undefined がセットされます。

onstring イベント

setPrefix メソッドで指定したキーコードのいずれかで始まり、Enter キーを検出するまでの間に検出した入力文字を文字列として受け取ります。

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
input	検出した文字列	String
prefix	文字列の開始と判断したキーコード	Number

setMSRPrefix メソッド

MSR 付きプログラマブルキーボードの使用を開始するためのキーコードを指定します。拡張メソッドである setMSRPrefix メソッドの呼び出しには、コールバックイベントを使用してください。

構文

```
callEvent(setMSRPrefix, data);
```

パラメーター

setMSRPrefix

設定値	説明
setMSRPrefix	固定

data

設定値	説明
keycode	カード情報の受信開始を判断するためのキーコードを指定

戻り値

戻り値	説明	オブジェクト種類
sq	処理のシーケンス番号	String

補足説明

指定するキーコードは、Key Code 一覧を参照してください。

ondata イベント

MSR 付きプログラマブルキーボードのカード読み取り情報を受け取ります。

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
track1	トラック 1 のデータ全体	String
track2	トラック 2 のデータ全体	String
track4	トラック 4（JIS2 トラック）のデータ全体	String
account_number	アカウント番号	String
expiration_date	有効期限（"YYMM" 形式）	String
surname	姓	String
first_name	名	String
middle_initial	ミドルネーム / イニシャル	String
title	タイトル	String
service_code	サービスコード	String
track1_dd	トラック 1 の任意データ	String
track2_dd	トラック 2 の任意データ	String

POSKeyboardオブジェクト

onkeypress イベント

POS キー押下イベントを受け取ります。

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
poskeycode	POS キーに設定されたキーコード	Number

MSR オブジェクト

ondata イベント

MSR デバイスのカード読み取り情報を受け取ります。

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
track1	トラック 1 のデータ全体	String
track2	トラック 2 のデータ全体	String
track4	トラック 4（JIS2 トラック）のデータ全体	String
account_number	アカウント番号	String
expiration_date	有効期限（"YYMM" 形式）	String
surname	姓	String
first_name	名	String
middle_initial	ミドルネーム / イニシャル	String
title	タイトル	String
service_code	サービスコード	String
track1_dd	トラック 1 の任意データ	String
track2_dd	トラック 2 の任意データ	String

Printer オブジェクト

addTextAlign メソッド

印字位置設定を命令バッファーに追加します。

構文

```
addTextAlign(align);
```

パラメーター

align

定数	説明
ALIGN_LEFT （初期値）	左揃え
ALIGN_CENTER	中央揃え
ALIGN_RIGHT	右揃え

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- addTextAlign メソッドの指定は「行の先頭」で行ってください。先頭以外で使用した場合、無効になります。
- ページモードでは機能しません。
- addTextAlignメソッドの設定は、ラスターイメージ/NVロゴ/バーコード/2次元シンボルにも適用されます。
- ページモードでの横方向印字の設定は、addTextPosition メソッドを使用してください。

addTextLineSpace メソッド

改行量設定を命令バッファーに追加します。

構文

```
addTextLineSpace(linespc);
```

パラメーター

linespc

設定値	説明
0 ～ 255 の整数値	改行量をドット単位で指定

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

1 行の改行量を印字文字より小さい値に設定すると、印字のために設定量より多く紙送りする場合があります。

addTextRotate メソッド

倒立印字設定を命令バッファーに追加します。

構文

```
addTextRotate(rotate);
```

パラメーター

rotate

設定値	説明
true	倒立印字を設定
false（初期値）	倒立印字設定を解除

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- addTextRotate メソッドの指定は「行の先頭」で行ってください。先頭以外で使用した場合、無効になります。
- ページモードでは機能しません。
- addTextRotate メソッドの設定は、バーコード /2 次元シンボルにも適用されます。
- ページモードでの倒立印字の設定は、addPageDirection メソッドを使用してください。

addText メソッド

印字する文字を命令バッファーに追加します。

構文

```
addText(data);
```

パラメーター

data

設定値	説明
String	印字する文字を文字列で指定

水平タブと改行、バックスラッシュは、以下のエスケープシーケンスを使用してください。

文字列	説明
\t	水平タブ (HT)
\h	改行 (LF)
\\	バックスラッシュ

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- addText メソッドによる文字の印字後に、文字以外を印字するには、改行または紙送りを実行してください。
- ページモードでは、文字のベースラインドットを基準に現在の印字位置へ印字文字を展開します。
- addTextLang メソッドで日本語を指定し、以下の文字を印刷すると正常に印刷されない可能性があります。印刷できない場合は、字形が似た文字を指定してください。

印字されない可能性がある文字		字形が似た文字	
～	U+301C	～	U+FF5E
‖	U+2016	‖	U+2225
−	U+2212	−	U+FF0D

印字されない可能性がある文字		字形が似た文字	
¢	U+00A2	￠	U+FFE0
£	U+00A3	￡	U+FFE1
¬	U+00AC	￢	U+FFE2
—	U+2014	―	U+2015

addTextLang メソッド

言語設定を命令バッファーに追加します。
addText メソッドで指定した文字列の言語を設定します。

構文

```
addTextLang(lang);
```

パラメーター

lang

設定値	説明
en（初期値）	英語（ANK 仕様）
de	ドイツ語（ANK 仕様）
fr	フランス語（ANK 仕様）
it	イタリア語（ANK 仕様）
es	スペイン語（ANK 仕様）
ja	日本語 （国際文字セットも日本語に変更されます。）
ja-jp	
ko	韓国語 （国際文字セットも韓国語に変更されます。）
ko-kr	
zh-hans	簡体語 （国際文字セットも中国語に変更されます。）
zh-cn	
zh-hant	繁体語
zh-tw	
th	タイ語（南アジアモデルのみ）
mul	複数言語（UTF-8）
上記以外の言語コード	英語（ANK 仕様）

指定した言語によって、一部の文字は以下のように印字されます。

言語	Characters $ (U+0024)	Characters \ (U+005C)
日本語	$	¥
韓国語	$	₩
簡体語	¥	\
繁体語	$	\

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- 本メソッドは印刷データごとに、印刷データの先頭で指定してください。
- 印字可能言語は、使用するプリンターの文字仕様に依存します。各プリンターの詳細取扱説明書を参照してください。

addTextFont メソッド

文字のフォント設定を命令バッファーに追加します。

構文

```
addTextFont(font);
```

パラメーター

font

定数	説明
FONT_A （初期値）	フォント A
FONT_B	フォント B
FONT_C	フォント C
FONT_D	フォント D
FONT_E	フォント E
FONT_SPECIAL_A	特殊フォント A
FONT_SPECIAL_B	特殊フォント B

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

addTextSmooth メソッド

スムージング設定を命令バッファーに追加します。

構文

```
addTextSmooth(smooth);
```

パラメーター

smooth

設定値	説明
true	スムージングを設定
false （初期値）	スムージング設定を解除

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

addTextDouble メソッド

文字の倍角設定を命令バッファーに追加します。

構文

```
addTextDouble(dw, dh);
```

パラメーター

dw

設定値	説明
true	横倍角を設定
false（初期値）	横倍角設定を解除
undefined	設定を変更しない

dh

設定値	説明
true	縦倍角を設定
false（初期値）	縦倍角設定を解除
undefined	設定を変更しない

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

4 倍角文字は、dw と dh に true を指定します。

addTextSize メソッド

文字の倍率設定を命令バッファーに追加します。

構文

```
addTextSize(width, height);
```

パラメーター

width

設定値	説明
１～８の整数値（初期値：1）	横方向の倍率を指定
undefined	設定を変更しない

height

設定値	説明
１～８の整数値（初期値：1）	縦方向の倍率を指定
undefined	設定を変更しない

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

addTextStyle メソッド

文字の装飾設定を命令バッファーに追加します。

構文

```
addTextStyle(reverse, ul, em, color);
```

パラメーター

reverse

設定値	説明
true	白黒反転文字を設定
false（初期値）	白黒反転文字設定を解除
undefined	設定を変更しない

ul

設定値	説明
true	アンダーラインを設定
false（初期値）	アンダーライン設定を解除
undefined	設定を変更しない

em

設定値	説明
true	太字を設定
false（初期値）	太字設定を解除
undefined	設定を変更しない

color

設定値	説明
COLOR_NONE	非印字
COLOR_1 （初期値）	第１色
COLOR_2	第２色
COLOR_3	第３色
COLOR_4	第４色
undefined	設定を変更しない

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

addTextPosition メソッド

横方向の印字開始位置設定を命令バッファーに追加します。

構文

```
addTextPosition(x);
```

パラメーター

x

初期値：0

設定値	説明
0 ～ 65535 の整数値	横方向の印字開始位置をドット単位で指定

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- 本 API により、プリンターは『行の先頭』ではない状態になります。
  x に 0 を指定した場合でも『行の先頭』ではない状態になります。
- 本 API 実行後、addTextAlign メソッド、addTextRotate メソッド は使用できません。
- 本 API の設定は、バーコード /2 次元シンボル / ラスターイメージ /NV ロゴにも適用されます。
- 本 API の設定は、1 行（もしくは 1 つのバーコード、2 次元シンボル、ラスターイメージ、NV ロゴ）ごとに有効です。複数行に設定したい場合は、行ごとに指定してください。

addTextVPosition メソッド

縦方向の印字開始位置設定を命令バッファーに追加します。

構文

```
addTextVPosition(y);
```

パラメーター

y

初期値：21

設定値	説明
0 ～ 65535 の整数値	縦方向の印字開始位置をドット単位で指定

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

スタンダードモードでは機能しません。

addFeedUnit メソッド

ドット単位の紙送り量設定を命令バッファーに追加します。

構文

```
addFeedUnit(unit);
```

パラメーター

unit

設定値	説明
0 ～ 255 の整数値	紙送り量をドット単位で指定

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

addFeedLine メソッド

行単位の紙送り量設定を命令バッファーに追加します。

構文

```
addFeedLine(line);
```

パラメーター

line

設定値	説明
0 ～ 255 の整数値	紙送り量を行単位で指定

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

addFeedPosition メソッド

レシート紙とラベル紙の紙送り位置設定を命令バッファーに追加します。

構文

```
addFeedPosition(pos);
```

パラメーター

pos

設定値	説明
FEED_PEELING	剥離位置まで紙送り
FEED_CUTTING	カット位置まで紙送り
FEED_CURRENT_TOF	現在ラベルの頭出し位置まで紙送り
FEED_NEXT_TOF	次ラベルの頭出し位置まで紙送り

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

ページモードでは機能しません。

addFeed メソッド

改行を命令バッファーに追加します。

構文

```
addFeed();
```

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

addImage メソッド

HTML5 Canvas で描画したイメージデータの印字を命令バッファーに追加します。

HTML5 Canvas の RGBA フルカラー画像の指定範囲を、addImage メソッドのカラーモード設定と halftone プロパティー、brightness プロパティーの設定に従って、ラスターイメージデータに変換します。画像の 1 ピクセルがプリンターの 1 ドットに相当し、透明色が含まれている場合は、画像の背景を白とみなします。

構文

```
addImage(context, x, y, width, height, color, mode);
```

パラメーター

context

設定値	説明
context	HTML5 Canvas の 2D コンテキストを指定

x

設定値	説明
0 ～ 65535 の整数値	印字範囲の横方向開始位置を指定

y

設定値	説明
0 ～ 65535 の整数値	印字範囲の縦方向開始位置を指定

width

設定値	説明
0 ～ 65535 の整数値	印字範囲の幅を指定

height

設定値	説明
0 ～ 65535 の整数値	印字範囲の高さを指定

color

設定値	説明
COLOR_NONE	非印字
COLOR_1（初期値）	第 1 色
COLOR_2	第 2 色
COLOR_3	第 3 色

設定値	説明
COLOR_4	第 4 色
undefined	設定を変更しない

mode

設定値	説明
MODE_MONO	モノクロ（2 階調）
MODE_GRAY16	多階調（16 階調）
undefined	モノクロ（2 階調）

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- 異なるドメインからダウンロードした画像が含まれた HTML5 Canvas は印字できません。
  JavaScript の同一出身ポリシーにより、セキュリティーエラーが発生します。
- 多階調印字はスタンダードモードでは使用できますが、ページモードでは使用できません。
- イメージデータを高速で印字するためには、addTextAlign メソッドのパラメーター定数を ALIGN_LEFT に指定し、addImage メソッドの width パラメーター値をプリンターの用紙幅を超えない 8 の倍数に指定してください。
- ページモードでは、イメージデータの左下ドットを基準として現在の印字位置にイメージデータを展開するため、印字位置は移動しません。

addLogo メソッド

プリンターの NV メモリーに登録されている NV ロゴ印字を命令バッファーに追加します。
NV ロゴは、あらかじめプリンターに登録しておく必要があります。NV ロゴの登録方法は各プリンターの詳細取扱説明書を参照してください。

構文

```
addLogo(key1, key2);
```

パラメーター

key1

設定値	説明
0 ～ 255 の整数値	NV ロゴのキーコード 1 を指定

key2

設定値	説明
0 ～ 255 の整数値	NV ロゴのキーコード 2 を指定

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- 多階調印字はスタンダードモードでは使用できますが、ページモードでは使用できません。
- ページモードでは、NV ロゴの左下ドットを基準として、現在の印字位置に NV ロゴを展開します。
- 本 API で指定する NV ロゴは、事前にバッファリングした addTextStyle メソッドの色設定で印刷されます。

addBarcode メソッド

バーコード印字を命令バッファーに追加します。

構文

```
addBarcode(data, type, hri, font, width, height);
```

パラメーター

data

文字列	説明
String	バーコードデータを文字列で指定

文字列で表現できないバイナリーデータは、以下のエスケープシーケンスで指定します。

文字列	説明
\xnn	コントロールコード（nn は 16 進数で指定）
\\	バックスラッシュ

type

定数	バーコードの種類
BARCODE_UPC_A	UPC-A
BARCODE_UPC_E	UPC-E
BARCODE_EAN13	EAN13
BARCODE_JAN13	JAN13
BARCODE_EAN8	EAN8
BARCODE_JAN8	JAN8
BARCODE_CODE39	CODE39
BARCODE_ITF	ITF
BARCODE_CODABAR	CODABAR
BARCODE_CODE93	CODE93
BARCODE_CODE128	CODE128
BARCODE_CODE128_AUTO	CODE128 auto
BARCODE_GS1_128	GS1-128
BARCODE_GS1_DATABAR_OMNIDIRECTIONAL	GS1 DataBar Omnidirectional
BARCODE_GS1_DATABAR_TRUNCATED	GS1 DataBar Truncated
BARCODE_GS1_DATABAR_LIMITED	GS1 DataBar Limited

定数	バーコードの種類
BARCODE_GS1_DATABAR_EXPANDED	GS1 DataBar Expanded

バーコードの種類	説明
UPC-A	11 桁の数字を指定した場合、チェックデジットを自動で付加します。 12 桁の数字を指定した場合、12 桁目をチェックデジットとして処理しますが、チェックデジットの検算は行いません。
UPC-E	最初の桁に 0 を指定してください。 2 ～ 6 桁目にメーカーコードを指定してください。 7 ～ 11 桁目にアイテムコードを右詰めで指定してください。アイテムコードの桁数はメーカーコードにより異なります。使用しない桁は 0 を指定してください。 11 桁の数字を指定した場合、チェックデジットを自動で付加します。 12 桁の数字を指定した場合、12 桁目をチェックデジットとして処理しますが、チェックデジットの検算は行いません。
EAN13 JAN13	12 桁の数字を指定した場合、チェックデジットを自動で付加します。 13 桁の数字を指定した場合、13 桁目をチェックデジットとして処理しますが、チェックデジットの検算は行いません。
EAN8 JAN8	7 桁の数字を指定した場合、チェックデジットを自動で付加します。 8 桁の数字を指定した場合、8 桁目をチェックデジットとして処理しますが、チェックデジットの検算は行いません。
CODE39	先頭の文字が * の場合、この文字をスタートキャラクターとして処理します。それ以外の場合、スタートキャラクターを自動で付加します。
ITF	スタートコードおよびストップコードを自動で付加します。 チェックデジットの付加および検算は行いません。
CODABAR	スタートキャラクター (A ～ D, a ～ d) を指定してください。 ストップキャラクター (A ～ D, a ～ d) を指定してください。 チェックデジットの付加および検算は行いません。
CODE93	スタートキャラクターおよびストップキャラクターを自動で付加します。 チェックデジットを計算して自動で付加します。
CODE128	スタートキャラクター (CODE A, CODE B, CODE C) を指定してください。 ストップキャラクターを自動で付加します。 チェックデジットを計算して自動で付加します。 以下の文字をエンコードするには、文字 { で始まる 2 文字を指定してください。 • FNC1：{1 • FNC2：{2 • FNC3：{3 • FNC4：{4 • CODE A：{A • CODE B：{B • CODE C：{C • SHIFT：{S • {：{{ CODE C を指定する場合、バーコードデータはエスケープシーケンスのコントロールコードとして指定してください。

バーコードの種類	説明
CODE128 auto	スタートキャラクタ ー、チェックデジット、ストップキャラクターを自動で付加します。 また、Code128 シンボルの全体幅が最小となるように与えられた文字列を自動的に変換します。この機能ではコードセットを指定する必要はなく、シンボルにしたいデータを入力するだけでバーコードが印刷されます。 例) 数字を入力する場合："123" など 　アルファベットを入力する場合："ABCabc" など 0 ～ 255 のデータ（制御コードおよび文字コード）を指定できます。 自動で全体幅が変化するため、バーコードが印刷領域に収まり、問題なく印刷されることを確認して使用してください。 FNC1 などを使用する場合は、CODE128 auto ではなく CODE128 を使用してください。
GS1-128	スタートキャラクター、FNC1、チェックデジット、ストップキャラクターを自動で付加します。 アプリケーション識別子 (AI) とそれに続くデータのチェックデジットを計算して自動で付加するには、チェックデジットの位置に文字 * を指定します。 アプリケーション識別子 (AI) を括弧で囲むことができます。括弧は HRI の印字文字として使用し、データとしてエンコードしません。 アプリケーション識別子 (AI) とデータの間に空白を挿入することができます。空白は HRI の印字文字として使用し、データとしてエンコードしません。 以下の文字をエンコードするには、文字 { で始まる 2 文字を指定してください。 • FNC1：{1 • FNC3：{3 • (：{(•)：{) • *：{* • {：{{
GS1 DataBar Omnidirectional	アプリケーション識別子 (AI) とチェックデジットを除く 13 桁の商品識別番号 (GTIN) を指定してください。
GS1 DataBar Truncated	
GS1 DataBar Limited	
GS1 DataBar Expanded	アプリケーション識別子 (AI) を括弧で囲むことができます。括弧は HRI の印字文字として使用し、データとしてエンコードしません。 以下の文字をエンコードするには、文字 { で始まる 2 文字を指定してください。 • FNC1：{1 • (：{(•)：{)

hri

定数	説明
HRI_NONE（初期値）	印字しない
HRI_ABOVE	バーコードの上
HRI_BELOW	バーコードの下

定数	説明
HRI_BOTH	バーコードの上と下の両方

font

定数	説明
FONT_A（初期値）	フォント A
FONT_B	フォント B
FONT_C	フォント C
FONT_D	フォント D
FONT_E	フォント E

width

初期値：3

設定値	説明
2 ～ 6 の整数値	1 モジュールの幅をドット単位で指定

height

初期値：162

設定値	説明
1 ～ 255 の整数値	バーコードの高さをドット単位で指定

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- addBarcode メソッドの指定は「行の先頭」で行ってください。
- ページモードでは、バーコードの左下ドット（HRI を除く）を基準として、現在の印字位置に展開します。
- type で指定するバーコードの種類に従った文字列を指定してください。data で指定した文字列と type で指定するバーコードの種類が一致しないとバーコードは印字されません。
- バーコードの種類で CODE128 auto を指定できるのは、TM-m30II/TM-m30II-H/TM-m30II-S/TM-m30II-SL/TM-T88VII/TM-L100 です。

addSymbol メソッド

2 次元シンボル印字を命令バッファーに追加します。

構文

```
addSymbol(data, type, level, width, height, size);
```

パラメーター

data

文字列	説明
String	2 次元シンボルデータを文字列で指定

文字列で表現できないバイナリーデータは、以下のエスケープシーケンスで指定します。

文字列	説明
\xnn	コントロールコード（nn は 16 進数で指定）
\\	バックスラッシュ

type

定数	2 次元シンボルの種類
SYMBOL_PDF417_STANDARD	Standard PDF417
SYMBOL_PDF417_TRUNCATED	Truncated PDF417
SYMBOL_QRCODE_MODEL_1	QR Code Model 1
SYMBOL_QRCODE_MODEL_2	QR Code Model 2
SYMBOL_QRCODE_MICRO	Micro QR Code
SYMBOL_MAXICODE_MODE_2	MaxiCode Mode 2
SYMBOL_MAXICODE_MODE_3	MaxiCode Mode 3
SYMBOL_MAXICODE_MODE_4	MaxiCode Mode 4
SYMBOL_MAXICODE_MODE_5	MaxiCode Mode 5
SYMBOL_MAXICODE_MODE_6	MaxiCode Mode 6
SYMBOL_GS1_DATABAR_STACKED	GS1 DataBar Stacked
SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL	GS1 DataBar Stacked Omnidirectional
SYMBOL_GS1_DATABAR_EXPANDED_STACKED	GS1 DataBar Expanded Stacked
SYMBOL_AZTECCODE_FULLRANGE	Aztec Code Full-Range モード
SYMBOL_AZTECCODE_COMPACT	Aztec Code Compact モード
SYMBOL_DATAMATRIX_SQUARE	DataMatrix ECC200 正方形

定数	2 次元シンボルの種類
SYMBOL_DATAMATRIX_RECTANGLE_8	DataMatrix ECC200 長方形 行数 8
SYMBOL_DATAMATRIX_RECTANGLE_12	DataMatrix ECC200 長方形 行数 12
SYMBOL_DATAMATRIX_RECTANGLE_16	DataMatrix ECC200 長方形 行数 16

2 次元シンボルの種類	説明
Standard PDF417	文字列を UTF-8 に変換後、エスケープシーケンスの処理を行い、エンコードします。 データ領域の最大コードワード数は 928 個、1 段あたりのデータ領域の最大コードワード数は 30 個、最大段数は 90 段です。
Truncated PDF417	
QR Code Model 1	文字列をシフト JIS に変換後、エスケープシーケンスの処理を行い、データの種類を以下の中から選択してエンコードします。 • 数字：0 ～ 9 • 英数字：0 ～ 9, A ～ Z, スペース , $, %, *, +, -, ., /, : • 漢字：シフト JIS 値 • 8 ビットバイトデータ：0x00 ～ 0xff
QR Code Model 2	
Micro QR Code2	
MaxiCode Mode 2	文字列を UTF-8 に変換後、エスケープシーケンスの処理を行い、エンコードします。 モード 2 およびモード 3 は、最初のデータが [)>\x1e01\x1dyy（yy は 2 桁の数字）の場合、これをメッセージヘッダーとして処理し、次のデータからプライマリメッセージとして処理します。それ以外の場合、最初のデータからプライマリメッセージとして処理します。 モード 2 の場合、以下の形式でプライマリメッセージを指定してください。 • 郵便コード（1 ～ 9 桁の数字）GS:(\x1d) • ISO 国名コード（1 ～ 3 桁の数字）GS:(\x1d) • サービスクラスコード（1 ～ 3 桁の数字） モード 3 の場合、以下の形式でプライマリメッセージを指定してください。 • 郵便コード(1 ～ 6 個のコードセット A で変換可能なデータ)GS(\x1d) ISO • 国名コード（1 ～ 3 桁の数字）GS (\x1d) • サービスクラスコード（1 ～ 3 桁の数字）
MaxiCode Mode 3	
MaxiCode Mode 4	
MaxiCode Mode 5	
MaxiCode Mode 6	
GS1 DataBar Stacked	文字列を UTF-8 に変換後、エスケープシーケンスの処理を行い、エンコードします。 アプリケーション識別子 (AI) とチェックデジットを除く 13 桁の商品識別番号 (GTIN) を指定してください。
GS1 DataBar Stacked Omnidirectional	
GS1 DataBar Expanded Stacked	文字列を UTF-8 に変換後、エスケープシーケンスの処理を行い、エンコードします。 アプリケーション識別子 (AI) を括弧で囲むことができます。括弧は HRI の印字文字として使用し、データとしてエンコードしません。 以下の文字をエンコードするには、文字 { で始まる 2 文字を指定してください。 • FNC1：{1 • (：{(•)：{)

2 次元シンボルの種類	説明
Aztec Code	文字列を UTF-8 に変換後、エスケープシーケンスの処理を行い、エンコードします。
DataMatrix	文字列を UTF-8 に変換後、エスケープシーケンスの処理を行い、エンコードします。

level

❏ PDF417

定数	説明
LEVEL_0	エラー訂正レベル 0
LEVEL_1	エラー訂正レベル 1
LEVEL_2	エラー訂正レベル 2
LEVEL_3	エラー訂正レベル 3
LEVEL_4	エラー訂正レベル 4
LEVEL_5	エラー訂正レベル 5
LEVEL_6	エラー訂正レベル 6
LEVEL_7	エラー訂正レベル 7
LEVEL_8	エラー訂正レベル 8
LEVEL_DEFAULT	既定値（エラー訂正レベル 1）

❏ QR Code

定数	説明
LEVEL_L	エラー訂正レベル L
LEVEL_M	エラー訂正レベル M
LEVEL_Q	エラー訂正レベル Q
LEVEL_H	エラー訂正レベル H
LEVEL_DEFAULT	既定値（エラー訂正レベル M）

❏ Aztec Code

定数	説明
5 ～ 95 の整数	エラー訂正レベル
LEVEL_DEFAULT	既定値（エラー訂正レベル 23）

width

2 次元シンボルの種類	有効な値	初期値
PDF417	2 ～ 8	3
QR Code	3 ～ 16	3
MaxiCode	未使用	

2 次元シンボルの種類	有効な値	初期値
2 次元 GS1 DataBar	2 ～ 8	2
Aztec Code	2 ～ 16	3
DataMatrix	2 ～ 16	3

height

2 次元シンボルの種類	有効な値	初期値
PDF417	2 ～ 8 （width に対する倍率）	3
QR Code	未使用	
MaxiCode		
2 次元 GS1 DataBar		
Aztec Code		
DataMatrix		

size

2 次元シンボルの種類		初期値	説明
PDF417		0（自動）	1 段あたりのコードワード数を指定
QR Code		未使用	
MaxiCode		未使用	
2 次元 GS1 DataBar	Expanded Stacked	0（自動）	2 次元シンボルの最大幅を指定（106 以上）
	その他	未使用	
Aztec Code		未使用	
DataMatrix		未使用	

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- addSymbol メソッドの指定は「行の先頭」で行ってください。

- ページモードでは、2 次元シンボルの左上ドットを基準として、現在の印字位置に展開します。
- type で指定する 2 次元シンボルの種類に従った文字列を指定してください。data で指定した文字列と type で指定する 2 次元シンボルの種類が一致しないと 2 次元シンボルは印字されません。
- Micro QR Code は、LEVEL_H をサポートしていません。
- type で指定した 2 次元シンボルの種類に合わせて、定数を指定してください。
- data にて MaxiCode、2 次元 GS1 DataBar を指定した場合は、LEVEL_DEFAULT を指定してください。

addHLine メソッド

横罫線を描画し、横罫線印字を命令バッファーに追加します。

構文

```
addHLine(x1, x2, style);
```

パラメーター

x1

設定値	説明
0 ～ 65535 の整数値	横罫線の描画開始位置をドット単位で指定

x2

設定値	説明
0 ～ 65535 の整数値	横罫線の描画終了位置をドット単位で指定

style

定数	説明
LINE_THIN	実線：細
LINE_MEDIUM	実線：中太
LINE_THICK	実線：太
LINE_THIN_DOUBLE	二重線：細
LINE_MEDIUM_DOUBLE	二重線：中太
LINE_THICK_DOUBLE	二重線：太
undefined	実線：細

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- ページモードでは機能しません。
- ページモードでの横罫線は、addPageLine メソッドを使用してください。

addVLineBegin メソッド

縦罫線の描画開始を命令バッファーに追加します。

構文

```
addVLineBegin(x, style);
```

パラメーター

x

設定値	説明
0 ～ 65535 の整数値	縦罫線の描画開始位置をドット単位で指定

style

定数	説明
LINE_THIN	実線：細
LINE_MEDIUM	実線：中太
LINE_THICK	実線：太
LINE_THIN_DOUBLE	二重線：細
LINE_MEDIUM_DOUBLE	二重線：中太
LINE_THICK_DOUBLE	二重線：太
undefined	実線：細

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- ページモードでは機能しません。
- ページモードでの横罫線は、addPageLine メソッドを使用してください。
- addVLineEnd メソッドで終了を指定するまで縦罫線の描画を行います。addVLineBegin メソッドは、addVLineEnd メソッドと組み合わせて使用してください。

addVLineEnd メソッド

縦罫線の描画終了を命令バッファーに追加します。

構文

```
addVLineEnd(x, style);
```

パラメーター

x

設定値	説明
0 ～ 65535 の整数値	縦罫線の描画終了位置をドット単位で指定

style

定数	説明
LINE_THIN	実線：細
LINE_MEDIUM	実線：中太
LINE_THICK	実線：太
LINE_THIN_DOUBLE	二重線：細
LINE_MEDIUM_DOUBLE	二重線：中太
LINE_THICK_DOUBLE	二重線：太
undefined	実線：細

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- ページモードでは機能しません。
- ページモードでの縦罫線は、addPageLine メソッドを使用してください。
- addPageBegin メソッドと組み合わせて使用してください。

addPageBegin メソッド

ページモードの開始を命令バッファーに追加します。
addPageBegin メソッドを実行すると、ページモードでの処理を開始します。

構文

```
addPageBegin();
```

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

補足説明

- ページモードでは、多階調印字をサポートしていません。
- addPageEnd メソッドと組み合わせて使用してください。

addPageEnd メソッド

ページモードの終了を命令バッファーに追加します。
addPageEnd メソッドを実行すると、ページモードでの処理を終了します。

構文

```
addPageEnd();
```

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

補足説明

addPageBegin メソッドと組み合わせて使用してください。

addPageArea メソッド

ページモード印字領域を命令バッファーに追加します。

addPageArea メソッドでページモード印字領域（座標）を指定し、addText メソッドなどの印字データを指定します。

構文

```
addPageArea(x, y, width, height);
```

パラメーター

x

0 は印字可能領域の左端です。

設定値	説明
0 ～ 65535 の整数値	横方向の原点をドット単位で指定

y

0 は紙送り前の位置です。

設定値	説明
0 ～ 65535 の整数値	縦方向の原点をドット単位で指定

width

設定値	説明
0 ～ 65535 の整数値	印字領域の幅をドット単位で指定

height

設定値	説明
0 ～ 65535 の整数値	印字領域の高さをドット単位で指定

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- 印字する内容に合わせて、印字領域を指定してください。印字領域からはみ出した印字データは、印字されません。
- スタンダードモードでは機能しません。
- 印字領域の幅と高さは、印字方向の設定に合わせて指定してください。印字領域の幅と高さが印字方向の設定と一致しないと、印字領域からはみ出した印字データは印字されません。印字方向の設定は、addPageDirection メソッドで指定します。

addPageDirection メソッド

ページモード印字方向設定を命令バッファーに追加します。

構文

```
addPageDirection(dir);
```

パラメーター

dir

定数	説明
DIRECTION_LEFT_TO_RIGHT（初期値）	回転しない（左上を始点に右方向へ印字）
DIRECTION_BOTTOM_TO_TOP	反時計回り 90 度回転 （左下を始点に上方向へ印字）
DIRECTION_RIGHT_TO_LEFT	180 度回転（右下を始点に左方向へ印字）
DIRECTION_TOP_TO_BOTTOM	時計回り 90 度回転 （右上を始点に下方向へ印字）

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

スタンダードモードでは機能しません。

addPagePosition メソッド

addPageArea メソッドで指定した印字領域内での印字開始位置（座標）を指定し、ページモード印字領域内の印字位置設定を命令バッファーに追加します。

構文

```
addPagePosition(x, y);
```

パラメーター

x

初期値：0

設定値	説明
0 ～ 65535 の整数値	横方向の印字位置をドット単位で指定

y

初期値：21

設定値	説明
0 ～ 65535 の整数値	縦方向の印字位置をドット単位で指定

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- スタンダードモードでは機能しません。
- 印字内容に合わせて、印字開始位置（座標）を指定してください。

印字内容	指定方法
文字列	最初の文字のベースライン左端を指定します。 標準の大きさで左詰めで印字する場合は省略可能です。高さが 2 倍の文字を印刷する場合は、y を 42 以上に指定します。

印字内容	指定方法
バーコード	シンボルの左下を指定します。y にバーコードの高さを指定してください。
グラフィック / ロゴ	グラフィックデータの左下を指定します。y にグラフィックデータの高さを指定します。
2 次元シンボル	シンボルの左上を指定します。左上から印字する場合は、省略可能です。

addPageLine メソッド

ページモードの直線描画設定を命令バッファーに追加します。

構文

```
addPageLine(x1, y1, x2, y2, style);
```

パラメーター

x1

設定値	説明
0 ～ 65535 の整数値	横方向の描画開始位置をドット単位で指定

y1

設定値	説明
0 ～ 65535 の整数値	縦方向の描画開始位置をドット単位で指定

x2

設定値	説明
0 ～ 65535 の整数値	横方向の描画終了位置をドット単位で指定

y2

設定値	説明
0 ～ 65535 の整数値	縦方向の描画終了位置をドット単位で指定

style

定数	説明
LINE_THIN	実線：細
LINE_MEDIUM	実線：中太
LINE_THICK	実線：太
LINE_THIN_DOUBLE	二重線：細
LINE_MEDIUM_DOUBLE	二重線：中太
LINE_THICK_DOUBLE	二重線：太
undefined	実線：細

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- スタンダードモードでは機能しません。
- 斜線は描画できません。描画する直線が縦罫線 / 横罫線になるようにパラメーターを指定してください。

addPageRectangle メソッド

ページモードの四角形描画設定を命令バッファーに追加します。

構文

```
addPageRectangle(x1, y1, x2, y2, style);
```

パラメーター

x1

設定値	説明
0 ～ 65535 の整数値	横方向の描画開始位置をドット単位で指定

y1

設定値	説明
0 ～ 65535 の整数値	縦方向の描画開始位置をドット単位で指定

x2

設定値	説明
0 ～ 65535 の整数値	横方向の描画終了位置をドット単位で指定

y2

設定値	説明
0 ～ 65535 の整数値	縦方向の描画終了位置をドット単位で指定

style

定数	説明
LINE_THIN	実線：細
LINE_MEDIUM	実線：中太
LINE_THICK	実線：太
LINE_THIN_DOUBLE	二重線：細
LINE_MEDIUM_DOUBLE	二重線：中太
LINE_THICK_DOUBLE	二重線：太
undefined	実線：細

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

スタンダードモードでは機能しません。

addRotateBegin メソッド

まとめ印刷（180 度回転）モードの開始を命令バッファーに追加します。

構文

```
addRotateBegin();
```

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

補足説明

- ページモードでは機能しません。
- まとめて 180 度回転印刷する add 系 API(例 addText メソッド)を、本 API と addRotateEnd メソッドで囲んで使用してください。
- 本 API と addRotateEnd メソッドの間で、以下の API を使用しないでください。
  - ・addPageBegin メソッド
  - ・addPageEnd メソッド
  - ・addCut メソッド
  - ・addPulse メソッド
  - ・addSound メソッド
- 一度のまとめ印刷（180 度回転）でプリンターが処理できるデータ容量は以下の通りです。
  - ・文字列：80 行
  - ・グラフィック：2400 ドット
- まとめ印刷（180 度回転）モードで add 系 API（例 addText メソッド）を使用する際、使用上注意が必要な API があります。

API	注意事項
addTextSize メソッド	• 本APIより前にaddTextSizeメソッドで縦方向を3倍角以上に設定していると、本 API 以降は 2 倍角になります。 • 本APIより後にaddTextSizeメソッドで縦方向を3倍角以上に設定すると、縦方向の指定は無視されて 1 倍角になります。
addFeedUnit メソッド	addTextLineSpace メソッドで指定された改行量によって、紙送り行数が変化します。addTextLineSpace メソッドで改行量を 30 に指定すると、addFeedUnit メソッドの最大値 255 で 8 行紙送りできます。

API	注意事項
addImage メソッド	• 印刷データの縦のサイズの合計が 2400 ドットを超える場合、印刷が意図通りにならないことがあります。例えば、2000 ドットのデータ A がプリンターバッファーに溜まっているいるところへ 500 ドットのデータ B を送信すると、データ A を印刷した後にデータ B をプリンターバッファーに溜めます。 • 多階調（16 階調）の最大サイズは縦 600 ドットまでです。 多階調（16 階調）の場合、モノクロ（2 階調）と比べて 4 倍のデータ容量を必要とします。 • 1 つのイメージデータで縦のサイズが一度に処理できるデータ容量を超えると、印刷されません。 モノクロ（2 階調）最大値：2400 ドット 多階調（16 階調）最大値：600 ドット

addRotateEnd メソッド

まとめ印刷（180 度回転）モードの終了を命令バッファーに追加します。

構文

```
addRotateEnd();
```

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

補足説明

- ページモードでは機能しません。
- まとめて 180 度回転印刷する add 系 API（例 addText メソッド）を、本 API と addRotateBegin メソッドで囲んで使用してください。

addCut メソッド

用紙カットを命令バッファーに追加します。

構文

```
addCut(type);
```

パラメーター

type

設定値	説明
CUT_NO_FEED	フィードなしカット（紙送りせずにカット）
CUT_FEED	フィードカット（紙送り後カット）
CUT_RESERVE	カット予約（後に続く印字を実行後、カット位置でカット）
undefined	フィードカット（紙送り後カット）

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- ページモードでは機能しません。
- 本 API は「行の先頭」で使用してください。先頭以外で使用した場合、無効になります。
- カット予約（CUT_RESERVE）に続く印刷データが指定されていないとき、プリンターは予約したカット位置まで紙を送ってからカットを実行します。
- プリンターによっては、カット予約（CUT_RESERVE）に続く印刷データを約 2 秒間待ってから紙送りの動作を開始します。
- カット予約（CUT_RESERVE）を使用するときは、1 枚のレシートの長さを 20mm 以上に設定してください。

addPulse メソッド

ドロアーキック設定を命令バッファーに追加します。

構文

```
addPulse(drawer, time);
```

パラメーター

drawer

設定値	説明
DRAWER_1	ドロアーキックコネクター 2 番ピン
DRAWER_2	ドロアーキックコネクター 5 番ピン
undefined	ドロアーキックコネクター 2 番ピン

time

設定値	説明
PULSE_100	100 msec の信号
PULSE_200	200 msec の信号
PULSE_300	300 msec の信号
PULSE_400	400 msec の信号
PULSE_500	500 msec の信号
undefined	100 msec の信号

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- ページモードでは機能しません。
- ドロアー機能は、ブザー機能と同時に使用できません。

❏ 短時間でドロアーオープンを繰り返さないでください。ドロアーに負荷がかかり、破損するおそれがあります。

addSound メソッド

ブザーの鳴動設定を命令バッファーに追加します。

構文

```
addSound(pattern, repeat, cycle);
```

パラメーター

pattern

設定値	説明
PATTERN_NONE	停止
PATTERN_A	パターン A（外付けオプションブザー）
PATTERN_B	パターン B（外付けオプションブザー）
PATTERN_C	パターン C（外付けオプションブザー）
PATTERN_D	パターン D（外付けオプションブザー）
PATTERN_E	パターン E（外付けオプションブザー）
PATTERN_ERROR	エラー鳴動パターン（外付けオプションブザー）
PATTERN_PAPER_END	用紙なし鳴動パターン（外付けオプションブザー）
PATERN_0	パターン 0（内蔵ブザー）
PATERN_1	パターン 1（内蔵ブザー）
PATERN_2	パターン 2（内蔵ブザー）
PATERN_3	パターン 3（内蔵ブザー）
PATERN_4	パターン 4（内蔵ブザー）
PATERN_5	パターン 5（内蔵ブザー）
PATERN_6	パターン 6（内蔵ブザー）
PATERN_7	パターン 7（内蔵ブザー）
PATERN_8	パターン 8（内蔵ブザー）
PATERN_9	パターン 9（内蔵ブザー）
PATERN_10	パターン 10（内蔵ブザー）
undefined	パターン A（外付けオプションブザー）

repeat

設定値	説明
0	無限
1 ～ 255	1 ～ 255 回

設定値	説明
undefined	1 回

cycle

設定値	説明
1000 ～ 25500	ブザーの鳴動周期を msec 単位で指定

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- ページモードでは機能しません。
- ブザー機能は、ドロアー機能と同時に使用できません。
- cycle は、pattern にて PATTERN_0 ～ PATTERN_10 を指定した場合のみ有効です。
- pattern に PATERN_0 を指定した場合、cycle に指定できる設定値は 1000 ～ 6000 の範囲です。

addLayout メソッド

用紙レイアウト設定を命令バッファーに追加します。

構文

```
addLayout(type, width, height, margin_top,
margin_bottom, offset_cut, offset_label);
```

パラメーター

type

設定値	説明
LAYOUT_RECEIPT	レシート紙
LAYOUT_RECEIPT_BM	レシート紙（ブラックマーク付）
LAYOUT_LABEL	ラベル紙
LAYOUT_LABEL_BM	ラベル紙（ブラックマーク付）

width

設定値	説明
290 ～ 800	用紙の幅を 0.1 mm 単位で指定

height

用紙の種類	有効な値	説明
レシート紙	0	指定不要
レシート紙（ブラックマーク付）	• 0（自動） • 284 ～ 1550（手動）	ブラックマーク上端位置から次のブラックマーク上端位置までの距離を 0.1 mm 単位で指定
ラベル紙		ラベル上端位置から次のラベル上端位置までの距離を 0.1 mm 単位で指定
ラベル紙（ブラックマーク付）		ブラックマーク下端位置から次のブラックマーク下端位置までの距離を 0.1 mm 単位で指定

margin_top

用紙の種類	有効な値	説明
レシート紙	0	指定不要
レシート紙（ブラックマーク付）	-150 ～ 1500	ブラックマーク上端位置から頭出し位置までの距離を 0.1 mm 単位で指定
ラベル紙	0 ～ 1500	ラベル上端位置から頭出し位置までの距離を 0.1 mm 単位で指定

用紙の種類	有効な値	説明
ラベル紙（ブラックマーク付）	-15 ～ 1500	ブラックマーク下端位置から頭出し位置までの距離を 0.1 mm 単位で指定

margin_bottom

用紙の種類	有効な値	説明
レシート紙	0	指定不要
レシート紙（ブラックマーク付）	0	指定不要
ラベル紙	-15 ～ 0	ラベル下端位置から印字領域下端までの距離を 0.1 mm 単位で指定 （正数方向は余白が拡がり、負数方向は余白が縮む）
ラベル紙（ブラックマーク付）	-15 ～ 15	ブラックマーク上端位置から印字領域下端までの距離を 0.1 mm 単位で指定 （正数方向は余白が拡がり、負数方向は余白が縮む）

offset_cut

用紙の種類	有効な値	説明
レシート紙	0	指定不要
レシート紙（ブラックマーク付）	-290 ～ 50	ブラックマーク上端位置からカット位置までの距離を 0.1 mm 単位で指定
ラベル紙	0 ～ 50	ラベル下端位置からカット位置までの距離を 0.1 mm 単位で指定
ラベル紙（ブラックマーク付）	0 ～ 50	ブラックマーク上端位置からカット位置までの距離を 0.1 mm 単位で指定

offset_label

用紙の種類	有効な値	説明
レシート紙	0	指定不要
レシート紙（ブラックマーク付）	0	指定不要
ラベル紙	0	指定不要
ラベル紙（ブラックマーク付）	0 ～ 15	ブラックマーク上端位置からラベル下端位置までの距離を 0.1 mm 単位で指定

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

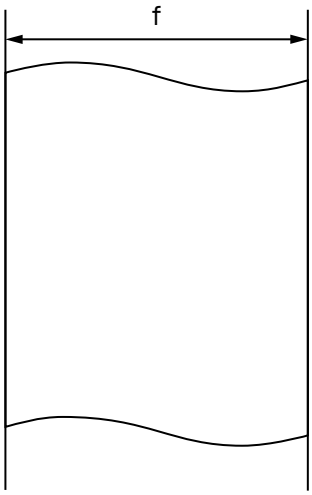
例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

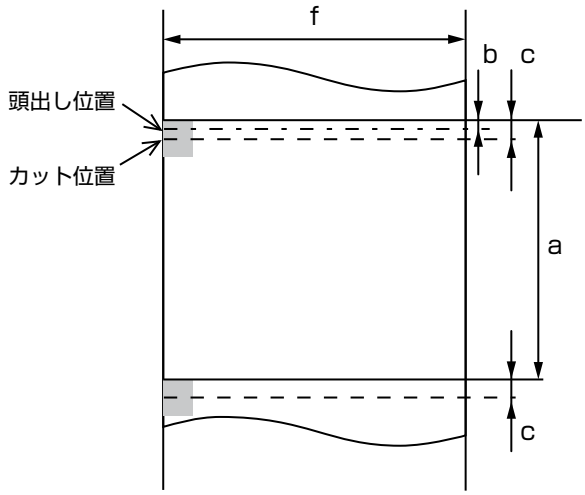
詳細説明

用紙ごと指定可能なパラメーターの位置は、以下を参照してください。

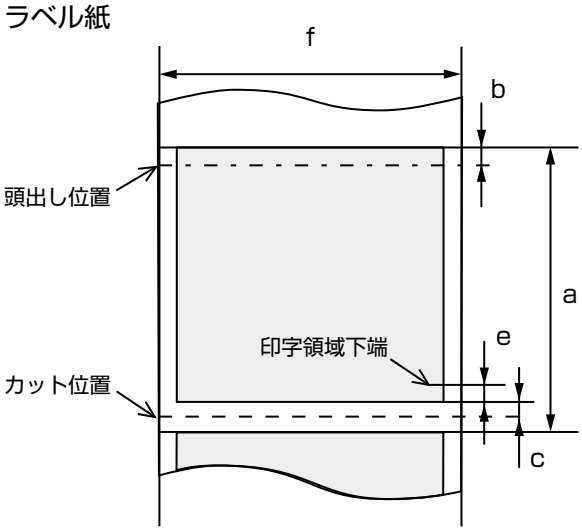
レシート紙

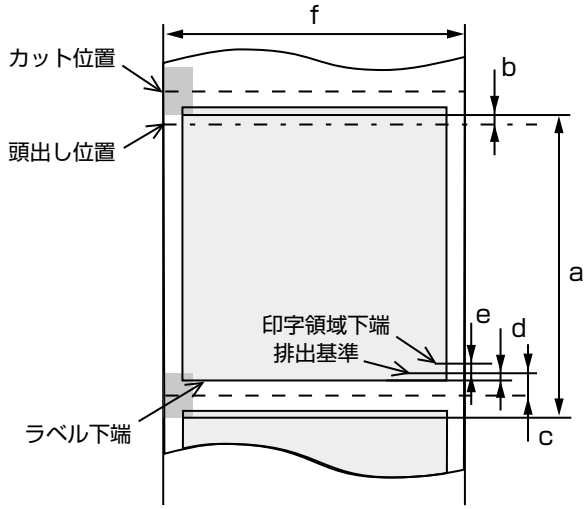


＜ブラックマーク付＞



ラベル紙





記号	パラメーター
f	width
a	height
b	margin_top
e	margin_bottom
c	offset_cut
d	offset_label

補足説明

ページモードでは機能しません。

recover メソッド

プリンター本体を復帰可能エラー状態 (例：オートカッターエラー) から復帰させます。

復帰可能エラーの詳細は、各プリンターの詳細取扱説明書を参照してください。

プリンター本体に復帰可能エラーが発生した時、プリンター本体のエラー要因を除去しただけでは、プリンター本体はエラー状態から復帰しません。recover メソッドを実行することで、プリンター本体はエラー状態から復帰し、印字可能な状態に戻ります。

構文

```
recover();
```

addRecovery メソッド

プリンター本体のエラー状態からの復帰命令を命令バッファーに追加します。

プリンター本体に復帰可能エラーが発生した時、addRecovery メソッドを実行することで、プリンター本体はエラー状態から復帰し、印字可能な状態に戻ります。

addRecovery メソッドを実行することによって得られる結果は、recover メソッドと同じです。
プリンター本体をエラー状態から復帰させる際は、recover メソッドの使用を推奨します。

構文

```
addRecovery();
```

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- 強制送信モード（force プロパティー）を有効にして使用してください。
- 復帰可能エラーから復帰すると、プリンター本体のバッファーはリセットされます。

reset メソッド

プリンター本体をリセットします。

プリンター本体は初期状態に戻り、プリンターバッファーに残っている印字データや、印刷設定などのプリンター本体に記憶されないデータは、全て失われます。

構文

```
reset();
```

addReset メソッド

プリンター本体のリセット命令を命令バッファーに追加します。

プリンター本体は初期状態に戻り、プリンターバッファーに残っている印字データや、印刷設定などのプリンター本体に記憶されないデータは、全て失われます。

addReset メソッドを実行することによって得られる結果は、reset メソッドと同じです。
addReset メソッドを命令バッファーに追加してプリンターのリセットを行うと、命令バッファーに含まれている他の印字命令は実行されません。プリンター本体のリセットには、reset メソッドの使用を推奨します。

構文

```
addReset();
```

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

addCommand メソッド

ESC/POS コマンドを命令バッファーに追加します。

構文

```
addCommand(data);
```

パラメーター

data

設定値	説明
String	ESC/POS コマンドを文字列で指定

戻り値

戻り値	オブジェクト種類
Printer オブジェクト	Printer

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- ESC/POS コマンドの詳細は以下の URL を参照してください。
  www.epson-biz.com/pos/reference_ja/
- Epson ePOS SDK は本 API で送信するコマンドをチェックしません。
  Epson ePOS SDK の動作を妨げるようなコマンドを送信した場合、他の API の動作やステータスの値が不正になることがあります。
  ESC/POS コマンドやレシートプリンターの仕様を十分に理解した上で使用してください。

send メソッド

命令バッファーのデータをプリンターへ送信します。
印刷ジョブ ID を指定できます。

構文

- ❏ send();
- ❏ send(printjobid);

パラメーター

printjobid

設定値	説明
String	1 ～ 30 文字の英数字、アンダースコア、ハイフン、ピリオドを使用して指定

サンプルプログラム

印刷ジョブ ID を指定して送信する

```
var printjobid = 'ABC123';

printer.addText('Hello, World!\n');
printer.addCut();
printer.onreceive = function (res) { alert(res.printjobid); };
printer.onerror = function (err) { alert(err.status); };
printer.send(printjobid);
```

補足説明

- ❏ send メソッドの実行結果は、onreceive イベントに渡されます。
- ❏ 強制送信モード（force プロパティー）を有効にして send メソッドを実行すると、データ送信後に強制送信モードを無効にします。
- ❏ 印刷ジョブ ID の指定機能は、TM-i シリーズ /TM-DT シリーズで使用できます。

print メソッド

HTML5 Canvas で描画したイメージデータを印字します。

印刷ジョブ ID を指定できます。

HTML5 Canvas の RGBA フルカラー画像の指定範囲を、halftone プロパティーと brightness プロパティーの設定に従って、ラスターイメージデータに変換します。画像の 1 ピクセルがプリンターの 1 ドットに相当します。透明色が含まれている場合は、画像の背景を白とみなします。

構文

- ❏ print(canvas, cut, mode);
- ❏ print(canvas, cut, mode, printjobid);

パラメーター

canvas

設定値	説明
canvas	HTML5 Canvas オブジェクトを指定

cut

設定値	説明
true	オートカットする
false	オートカットしない
undefined	オートカットしない

mode

設定値	説明
MODE_MONO	モノクロ（2 階調）
MODE_GRAY16	多階調（16 階調）
undefined	モノクロ（2 階調）

printjobid

設定値	説明
String	1 ～ 30 文字の英数字、アンダースコア、ハイフン、ピリオドを使用して指定

例外

例外	オブジェクト種類
Parameter " ... " is invalid	Error

補足説明

- 異なるドメインからダウンロードした画像が含まれた HTML5 Canvas は印字できません。JavaScript の同一出身ポリシーにより、セキュリティーエラーが発生します。
- 多階調印字はスタンダードモードでは使用できますが、ページモードでは使用できません。
- 印刷ジョブ ID の指定機能は、TM-i シリーズ /TM-DT シリーズで使用できます。

getPrintJobStatus メソッド

指定した印刷ジョブ ID の状態を取得します。
getPrintJobStatus メソッドで取得した印刷ジョブ ID の状態は、onreceive イベントへ渡されます。

構文

```
getPrintJobStatus(printjobid);
```

パラメーター

printjobid

設定値	説明
String	1 ～ 30 文字の英数字、アンダースコア、ハイフン、ピリオドを使用して指定

補足説明

同じ印刷ジョブ ID で複数の印字処理を実行した場合、最新の印刷ジョブ ID の状態を取得します。

startMonitor メソッド

Printer オブジェクトによるプリンターステータスの取得、更新を有効にします。

startMonitor メソッドを実行すると、Printer オブジェクトは interval プロパティーで指定した間隔でプリンターステータスを取得、更新し、以下のイベントに取得、更新したプリンターステータスを渡します。

- onstatuschange イベント
- onbatterystatuschange イベント
- ononline イベント
- onoffline イベント
- onpoweroff イベント
- oncoverok イベント
- oncoveropen イベント
- onpaperok イベント
- onpapernearend イベント
- onpaperend イベント
- ondrawerclosed イベント
- ondraweropen イベント
- onbatteryok イベント
- onbatterylow イベント

構文

```
startMonitor();
```

stopMonitor メソッド

startMonitor メソッドで有効にした Printer オブジェクトによるプリンターステータスの取得、更新を無効にします。

構文

```
stopMonitor();
```

halftone プロパティー

ハーフトーンの処理方法を指定します。
addImage メソッドと print メソッドのモノクロ（2 階調）印字に適用されます。

オブジェクト種類

String

定数	処理方法	説明
HALFTONE_DITHER（初期値）	ディザ	グラフィックのみの印刷
HALFTONE_ERROR_DIFFUSION	誤差拡散	文字とグラフィックが混在する印刷
HALFTONE_THRESHOLD	しきい値	文字のみの印刷

brightness プロパティー

明るさ補正値を指定します。
addImage メソッドと print メソッドの印字に適用されます。

オブジェクト種類

Number

初期値：1.0

設定値	説明
0.1 ～ 10.0	明るさ補正値を指定

force プロパティー

強制送信モードを指定します。
強制送信モードを有効にすると、プリンター本体へ強制的に印字命令を送信します。

オブジェクト種類

Boolean

設定値	説明
true	有効
false（初期値）	無効

補足説明

- 強制送信モードは、プリンター本体がオフラインの状態で使用してください。
- プリンター本体がオンラインの状態で使用した場合、onreceive イベントの code にエラーコード 'EX_BADPORT' が渡されます。
- 強制送信モードで有効な機能は、以下の通りです。
  ドロアーキック（addPulse メソッド）
  ブザーの停止（addSound メソッド）
  復帰可能エラーからの復帰（addRecovery メソッド）
  リセット（addReset メソッド）
  リアルタイムコマンドの送信（addCommand メソッド）

timeout プロパティー

send メソッドで送信するメッセージのタイムアウト時間を指定します。
機種の仕様や通信インターフェイス、送信データサイズに合わせてタイムアウト時間を調整します。

オブジェクト種類

Number

初期値：10000

設定値	説明
0 ～ 300000	タイムアウト時間を msec 単位で指定

interval プロパティー

startMonitor メソッドで有効にしたプリンターステータスの取得、更新間隔を指定します。

オブジェクト種類

Number

初期値：3000
不正な値を指定した場合は、初期値が指定されます。

設定値	説明
1000 ～ 6000	プリンターステータスの更新間隔を msec 単位で指定

drawerOpenLevel プロパティー

ondraweropen イベントと ondrawerclosed イベントを発生させるドロアーの信号線状態を指定します。
使用するドロアーの仕様に合わせて変更してください。

オブジェクト種類

Integer

設定値	信号線状態	説明
DRAWER_OPEN_LEVEL_LOW（初期値）	HIGH → LOW	ondraweropen イベント発生
	LOW → HIGH	ondrawerclosed イベント発生
DRAWER_OPEN_LEVEL_HIGH	HIGH → LOW	ondrawerclosed イベント発生
	LOW → HIGH	ondraweropen イベント発生

message プロパティー

Printer オブジェクトの API で追加したコマンドを、コードで表示します。
本プロパティーからコードを直接操作し、コマンドの確認や追加、削除ができます。

オブジェクト種類

String

onreceive イベント

印刷結果を受け取ります。

構文

Function (response)

コールバック関数のパラメーター

パラメーター：response

オブジェクト種類：Object

response オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
success	印字結果	Boolean
code	エラーコード	String
status	ステータス	Number
battery	バッテリーステータス	Number
printjobid	印刷ジョブ ID	String

success

値	説明
true	• 印字成功 • 処理に成功（スプーラー機能有効時）
false	• 印字失敗 • 処理に失敗（スプーラー機能有効時）

code

値	説明
'EPTR_AUTOMATICAL'	自動復帰エラー発生
'EPTR_BATTERY_LOW'	バッテリー残量なし
'EPTR_COVER_OPEN'	カバーオープンエラー発生
'EPTR_CUTTER'	オートカッターエラー発生
'EPTR_MECHANICAL'	メカニカルエラー発生
'EPTR_REC_EMPTY'	ロール紙エンド
'EPTR_UNRECOVERABLE'	復帰不可能エラー発生
'SchemaError'	印刷要求 XML の構文に誤りがある
'DeviceNotFound'	デバイス ID で指定したプリンターが存在しない

値	説明
'PrintSystemError'	印刷システムにエラーが発生
'EX_BADPORT'	通信ポートに異常を検出
'EX_TIMEOUT'	印刷タイムアウト発生
'EX_SPOOLER'	印刷キューに空き容量がない
'JobNotFound'	指定のジョブ ID が存在しない
'Printing'	印刷中
'TooManyRequests'	印刷データ送信数が許容量を超えた
'RequestEntityTooLarge'	印刷ジョブのデータサイズが、プリンターの許容量を超えている
'JobSpooling'	スプール中
'ERROR_WAIT_EJECT'	用紙除去待ち中

status

定数との AND 演算を行ってください。

定数 (status)	説明
ASB_NO_RESPONSE	プリンター無応答
ASB_PRINT_SUCCESS	印刷終了 / スプール完了
ASB_DRAWER_KICK	ドロアーキックコネクター 3 番ピンの状態＝ "H"
ASB_BATTERY_OFFLINE	バッテリー残量によるオフライン状態
ASB_OFF_LINE	オフライン状態
ASB_COVER_OPEN	カバーオープン
ASB_PAPER_FEED	紙送りスイッチによる紙送り中
ASB_PANEL_SWITCH	パネルスイッチ操作中
ASB_MECHANICAL_ERR	メカニカルエラー発生
ASB_AUTOCUTTER_ERR	オートカッターエラー発生
ASB_UNRECOVER_ERR	復帰不可能エラー発生
ASB_AUTORECOVER_ERR	自動復帰エラー発生
ASB_RECEIPT_NEAR_END	ロール紙ニアエンド
ASB_RECEIPT_END	ロール紙エンド
ASB_BUZZER	ブザー鳴動中（対応機種のみ）
ASB_WAIT_REMOVE_LABEL	ラベルの剥離待ち中
ASB_NO_LABEL	剥離待ちラベルなし
ASB_SPOOLER_IS_STOPPED	スプーラーの停止

battery

❏ AC アダプター接続状態

定数 (battery)	説明
0x30XX	接続されている
0x31XX	接続されていない

❏ バッテリー残量

定数 (battery)	説明
0xXX36	残量 6
0xXX35	残量 5
0xXX34	残量 4
0xXX33	残量 3
0xXX32	残量 2
0xXX31	残量 1（ニアエンド）
0xXX30	残量 0（リアルエンド）

printjobid

値	説明
printjobid	印刷ジョブ ID

補足説明

❏ バッテリー非搭載機種の場合、バッテリー残量の定数は 0xXX30 がセットされます。

❏ 印刷ジョブ ID が無い場合、printjobid には “” 空文字列がセットされます。

❏ 通信異常が発生した場合、onreceive イベントを通知しません。
切断イベントを通知できるように、onreconnecting イベント、onreconnect イベント、ondisconnect イベントを実装してください。

❏ 通信異常が発生して復旧しないと判断したときは、ondisconnect イベントを通知します。
この場合は onreceive イベントを通知しないため、ondisconnect イベントで印刷の失敗を検出してください。

onstatuschange イベント

Printer オブジェクトが取得したプリンターステータスを受け取ります。

構文

```
Function (status)
```

コールバック関数のパラメーター

パラメーター：status

オブジェクト種類：Number

onbatterystatuschange イベント

Printer オブジェクトが取得したバッテリーステータスを受け取ります。

構文

```
Function (battery)
```

コールバック関数のパラメーター

パラメーター：battery

オブジェクト種類：Number

ononline イベント

Printer オブジェクトが取得したオンラインステータスを受け取ります。

構文

```
Function ()
```

onoffline イベント

Printer オブジェクトが取得したオフラインステータスを受け取ります。

構文

```
Function ()
```

onpoweroff イベント

Printer オブジェクトが取得した無応答ステータスを受け取ります。

構文

```
Function ()
```

oncoverok イベント

Printer オブジェクトが取得したカバークローズステータスを受け取ります。

構文

```
Function ()
```

oncoveropen イベント

Printer オブジェクトが取得したカバーオープンステータスを受け取ります。

構文

```
Function ()
```

onpaperok イベント

Printer オブジェクトが取得した用紙ありステータスを受け取ります。

構文

```
Function ()
```

onpapernearend イベント

Printer オブジェクトが取得した用紙ニアエンドステータスを受け取ります。

構文

```
Function ()
```

onpaperend イベント

Printer オブジェクトが取得した用紙なしステータスを受け取ります。

構文

```
Function ()
```

ondrawerclosed イベント

Printer オブジェクトが取得したドロアークローズステータスを受け取ります。

構文

```
Function ()
```

補足説明

ondrawerclosed イベントを発生させるドロアーの信号線状態は、drawerOpenLevel プロパティーで指定します。

ondraweropen イベント

Printer オブジェクトが取得したドロアーオープンステータスを受け取ります。

構文

```
Function ()
```

補足説明

ondraweropen イベントを発生させるドロアーの信号線状態は、drawerOpenLevel プロパティーで指定します。

onbatteryok イベント

Printer オブジェクトが取得したバッテリー残量ありステータスを受け取ります。

構文

```
Function ()
```

onbatterylow イベント

Printer オブジェクトが取得したバッテリー残量なしステータスを受け取ります。

構文

```
Function ()
```

Scanner オブジェクト

ondata イベント

バーコードスキャナーがバーコードから読み取った情報を受け取ります。

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
input	検出された文字列	String

SimpleSerial オブジェクト

sendCommand

任意コマンドを送信します。sendCommand 実行によるデバイスからの応答データは、oncommandreply イベントのコールバック関数に渡されます。

構文

```
sendCommand(data);
```

パラメーター

data

設定値	説明
data	コマンドを文字列で指定

oncommandreply イベント

sendCommand 実行によるデバイスからの応答データを受け取ります。

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
data	応答データの文字列	String

補足説明

- デバイスからの応答データが分割されて通知されることがあるため、必ず oncommandreply イベントで全ての応答データを受信したかどうかを確認してください。
  応答データが分割されて通知された場合、oncommandreply イベントが複数回発生します。後続データを待って処理を進めてください。
- 通信異常が発生した場合、通信切断を通知しません。
  切断イベントを通知できるように、onreconnecting イベント、onreconnect イベント、ondisconnect イベントを実装してください。

DeviceHubTerminal オブジェクト

shutdown メソッド

プリンターの電源をシャットダウンします。

構文

```
shutdown(password, callback);
```

パラメーター

password

設定値	説明
password	EPSON TMNet WebConfig に設定されているパスワードを文字列で指定

callback

設定値	説明
callback	実行結果を受け取るコールバック関数を指定

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	シャットダウンの実行結果	String

status

値	説明
"SUCCESS"	成功
"AUTHENTICATION_ERROR"	パスワード認証エラー
"ACCESS_DENIED"	シャットダウン設定が「許可しない」設定の状態で、シャットダウンを実行した
"SYSTEM_ERROR"	システムエラー

補足説明

- EPSON TMNet WebConfig を使用して、プリンターのシャットダウン設定を「許可する」に設定する必要があります。詳細は、各プリンターの詳細取扱説明書を参照してください。
- EPSON TMNet WebConfig のシャットダウン設定にて、パスワード認証設定を「認証する」に設定した場合、password に指定した文字列でパスワード認証を行います。

restart メソッド

プリンターを再起動します。

構文

```
restart(password, callback);
```

パラメーター

password

設定値	説明
password	EPSON TMNet WebConfig に設定されているパスワードを文字列で指定

callback

設定値	説明
callback	実行結果を受け取るコールバック関数を指定

コールバック関数のパラメーター

パラメーター：data
オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
status	再起動の実行結果	String

status

値	説明
"SUCCESS"	成功
"AUTHENTICATION_ERROR"	パスワード認証エラー
"ACCESS_DENIED"	シャットダウン設定が「許可しない」設定の状態で、再起動を実行した
"SYSTEM_ERROR"	システムエラー

戻り値

戻り値	説明	オブジェクト種類
sq	処理のシーケンス番号	String

補足説明

- EPSON TMNet WebConfig を使用して、プリンターのシャットダウン設定を「許可する」に設定する必要があります。詳細は、各プリンターの詳細取扱説明書を参照してください。
- EPSON TMNet WebConfig のシャットダウン設定にて、パスワード認証設定を「認証する」に設定した場合、password に指定した文字列でパスワード認証を行います。

CommBoxManager オブジェクト

openCommBox メソッド

コミュニケーションボックスをオープンします。

TM-DT ソフトウェア /TM-i ファームウェアはコミュニケーションボックスを作成し、アプリケーションをコミュニケーションボックスのメンバーに追加します。

構文

```
openCommBox(boxID, option, callback);
```

パラメーター

boxID

設定値	説明
boxID	オープンするボックス ID を文字列で指定

option

プロパティー	説明	オブジェクト種類
memberID	コミュニケーションボックス内で、自身を識別するためのメンバー ID を指定	String

callback

オープン処理結果を受け取るコールバック関数を指定します。

第 1 パラメーターに commBox オブジェクト、第 2 パラメーターには、結果を示す以下の文字列が渡されます。

第 3 パラメーターには戻り値と同じシーケンス番号が渡されます。

文字列	説明
"OK"	コミュニケーションボックスのオープンに成功
"ALREADY_OPENED"	すでにコミュニケーションボックスがオープンしている
"MEMBERID_ALREADY_USED"	指定したメンバー ID はすでに使用されている
"BOX_COUNT_OVER"	コミュニケーションボックスを作成可能な最大数を超えている
"BOX_CLIENT_OVER"	コミュニケーションボックスに所属可能な最大数を超えている
"PARAMETER_ERROR"	パラメーターエラーが発生した
"SYSTEM_ERROR"	その他のエラーが発生した

戻り値

戻り値	説明	オブジェクト種類
sq	処理のシーケンス番号	String

補足説明

openCommBox メソッドの処理に失敗すると、第 1 パラメーターには null が渡されます。

closeCommBox メソッド

コミュニケーションボックスをクローズします。

TM-DT ソフトウェア /TM-i ファームウェアはアプリケーションをコミュニケーションボックスのメンバーから削除し、メンバーのいなくなったコミュニケーションボックスは破棄されます。

構文

```
closeCommBox(commBoxObj, callback);
```

パラメーター

commBoxObj

設定値	説明
commBoxObj	クローズする commBox オブジェクトを指定

callback

クローズ処理結果を受け取るコールバック関数を指定します。

第 1 パラメーターに結果を示す以下の文字列が渡され、第 2 パラメーターには戻り値と同じシーケンス番号が渡されます。

文字列	説明
"OK"	コミュニケーションボックスのクローズに成功
"NOT_OPENED"	コミュニケーションボックスがオープンしていない
"PARAMETER_ERROR"	パラメーターエラーが発生した
"SYSTEM_ERROR"	その他のエラーが発生した

戻り値

戻り値	説明	オブジェクト種類
sq	処理のシーケンス番号	String

CommBox オブジェクト

getCommHistory メソッド

コミュニケーションボックスの送信履歴を取得します。

構文

- ❏ getCommHistory(callback);
- ❏ getCommHistory(option, callback);

パラメーター

option

設定値	説明
true	コミュニケーションボックス内全ての送信履歴を取得
false	自身のメンバー ID 宛に送信されたデータの送信履歴のみを取得

callback

コミュニケーションボックスの送信履歴を受け取るコールバック関数を指定します。

第 1 パラメーターに結果を示す以下の文字列が渡されます。

文字列	説明
"OK"	送信履歴の取得に成功
"NOT_OPENED"	コミュニケーションボックスが開いていない
"SYSTEM_ERROR"	その他のエラーが発生した

第 2 パラメーターに以下の historyList が渡されます。

historyList.[index]	説明	オブジェクト種類
senderId	送信元の memberID	String
receiverId	送信先の memberID	String
message	受信したメッセージ	Hash

第 3 パラメーターには戻り値と同じシーケンス番号が渡されます。

戻り値

戻り値	説明	オブジェクト種類
sq	処理のシーケンス番号	String

補足説明

option を省略すると、false が指定されます。

send メソッド

コミュニケーションボックスにデータを送信します。

構文

```
send(message, memberId, callback);
```

パラメーター

message

設定値	説明
message	送信するデータを指定

memberId

設定値	説明
memberId	送信先のメンバー ID を指定

callback

send メソッドの実行結果を受け取るコールバック関数を指定します。

第 1 パラメーターに結果を示す以下の文字列が渡されます。

文字列	説明
"OK"	メッセージの送信に成功
"NOT_OPENED"	コミュニケーションボックスが開いていない
"MEMBER_NOT_FOUND"	指定したメンバーが見つからない
"SYSTEM_ERROR"	その他のエラーが発生した

第 2 パラメーターにデータの送信を試みたクライアント数が渡され、第 3 パラメーターには戻り値と同じシーケンス番号が渡されます。

戻り値

戻り値	説明	オブジェクト種類
sq	処理のシーケンス番号	String

補足説明

memberId に null もしくは ""（空文字）を指定すると、送信者を除くコミュニケーションボックス内全てのメンバーに送信します。

onreceive イベント

コミュニケーションボックスが受信した内容をアプリケーションに通知します。

構文

```
Function(data);
```

コールバック関数のパラメーター

パラメーター：data

オブジェクト種類：Object

data オブジェクトのプロパティー

プロパティー	説明	オブジェクト種類
senderId	送信元のメンバー ID	String
receiverId	送信先のメンバー ID	String
message	受信データ	String

補足説明

- openCommBox メソッド実行時にメンバー ID の指定がない場合、senderId には ""（空文字）がセットされます。
- コミュニケーションボックス内全てのメンバーに送信した場合、receiverId には ""（空文字）がセットされます。
- 通信異常が発生した場合、onreceive イベントを通知しません。
  切断イベントを通知できるように、onreconnecting イベント、onreconnect イベント、ondisconnect イベントを実装してください。
- 通信異常が発生して復旧しないと判断したときは、ondisconnect イベントを通知します。
  この場合は onreceive イベントを通知しないため、ondisconnect イベントでデータ送信の失敗を検出してください。

ePOSDeviceConfiguration オブジェクト

Constructor

ePOSDeviceConfiguration オブジェクトを作成します。

構文

```
ePOSDeviceConfiguration(address);
```

パラメーター

address

設定値	説明
address	接続先プリンターの IP アドレスを指定

getRegisterdDevices メソッド

プリンターに登録されているデバイスの利用可能状況を取得し、アプリケーションに通知します。

構文

```
getRegisterdDevices(deviceGroup, callback);
```

パラメーター

deviceGroup

設定値	説明
DEVICE_GROUP_ALL	全てのデバイスを取得

callback

getRegisterdDevices メソッドの実行結果を受け取るコールバック関数を指定します。

第 1 パラメーターに ePOSDeviceConfiguration オブジェクトが渡され、第 2 パラメーターには結果を示す以下の値が渡されます。

プロパティー	説明
deviceId	デバイス ID
deviceType	デバイスの種類
status	デバイスの状態

- deviceType

値	説明
type_cash_changer	自動つり銭機
type_display	カスタマーディスプレイ
type_keyboard	キーボード
type_msr	MSR
type_printer	プリンター
type_scanner	バーコードスキャナー
type_simple_serial	シリアル通信デバイス

- status

値	説明
online	利用可能
offline	利用不可能
unknown	状態不明

補足説明

deviceGroup パラメーターの指定は、DEVICE_GROUP_ALL 固定です。

Error Code 一覧

callback パラメーターで取得するエラーコードと対処方法

エラーコード	要因	対処方法
PARAM_ERROR	不正なパラメーターが渡された。 ＜例＞ サポート範囲外の数値が指定された。	パラメーターの指定値を確認してください。
ERROR_PARAMETER		
PARAMETER_ERROR		
ERROR_TIMEOUT	指定時間内にデバイスと通信できなかった。	タイムアウト時間を確認してください。 タイムアウト時間は、印刷所要時間以上に設定してください。
DEVICE_NOT_FOUND	指定対象を発見できなかった。 ＜例＞ 指定した接続先のプリンターが存在しない。	deviceId や deviceType が間違っていないか、確認してください。
DEVICE_IN_USE	指定されたデバイスが使用中。 ＜例＞ 他のアプリケーションソフトで使用されている。	他のアプリケーションソフトからの使用を終了してください。
DEVICE_TYPE_INVALID	• デバイスのオブジェクトが違う（Printer、Display）。 • サポートしていない接続方法が指定されています。	deviceId で指定したデバイスのタイプが deviceType と一致しません。 deviceId および deviceType パラメーターの指定を見直してください。
DEVICE_OPEN_ERROR	デバイスの接続に失敗した。	デバイスとの接続状態を確認してください。
DEVICE_NOT_OPEN	• デバイスと接続されいない。 • createDevice メソッドが呼ばれていない状態で、デバイスを制御しようとした。	• 通信状態を確認し、デバイスと接続してください。 • createDevice メソッドが呼ばれているか確認してください。
DEVICE_CLOSE_ERROR	デバイスの切断に失敗した。	デバイスとの通信状態を確認してください。
ALREADY_OPENED	既にコミュニケーションボックスがオープンしている。	コミュニケーションボックスとの通信を終了してください。
MEMBERID_ALREADY_USED	指定したメンバー ID は既に使用されている。	異なるメンバー ID を指定してください。
MEMBER_NOT_FOUND	指定対象を発見できなかった。 ＜例＞ コミュニケーションボックスで、指定したデータ送信先のメンバーが存在しない。	指定したメンバー ID と、送信先デバイスの通信状態を確認してください。
BOX_COUNT_OVER	コミュニケーションボックスを作成可能な最大数を超えている。	不要なコミュニケーションボックスを削除してください。

エラーコード	要因	対処方法
BOX_CLIENT_OVER	コミュニケーションボックスに所属可能な最大数を超えている。	コミュニケーションボックスを使用していないメンバーを除名してください。
NOT_OPENED	コミュニケーションボックスがオープンしていない。	コミュニケーションボックスをオープンしてください。
SYSTEM_ERROR	その他のエラーが発生した。	実行環境に問題がないか確認してください。
OPOS_CODE_XX	デバイスベンダー定義のエラーが発生した。	OPOS ドライバーのマニュアルで、ResultCodeExtended プロパティーを参照してください。

onreceive イベントで取得するエラーコードと対処方法

エラーコード

❏ Printer オブジェクト

エラーコード	要因	対処方法
EPTR_AUTOMATICAL	自動復帰エラーが発生した。	プリンター復帰後に処理を実行してください。
EPTR_BATTERY_LOW	バッテリー残量がなくなった。	バッテリーを交換するか、AC アダプターを接続してください。
EPTR_COVER_OPEN	カバーが開いている。	プリンターのカバーを閉じてください。
EPTR_CUTTER	オートカッターエラーが発生した。	エラー原因を取り除き、プリンターの電源を入れ直してください。
EPTR_MECHANICAL	メカニカルエラーが発生した。	エラー原因を取り除き、プリンターの電源を入れ直してください。
EPTR_REC_EMPTY	用紙なし。	用紙を補充してください。
EPTR_UNRECOVERABLE	復帰不可能エラーが発生した。	プリンターの電源を入れ直してください。 再発する場合は、購入先かサービスセンターに連絡してください。
SchemaError	印刷要求 XML の構文に誤りがある	印刷要求 XML の構文を確認してください。
DeviceNotFound	• 接続形式や IP アドレスなどの指定が間違っている。 • 指定したデバイスが接続されていない。 • 指定したデバイスがプリンターに登録されていない。	• 接続形式や IP アドレスなどの指定値を確認してください。 • デバイスの接続状況を確認してください。 • 該当デバイスのプリンターへの登録状態を確認してください。
PrintSystemError	TM-i ファームウェア、TM-DT ソフトウェアに異常が発生した。	TM-i シリーズ /TM-DT シリーズの電源を入れ直し、OS を再起動してください。
EX_BADPORT	オンライン状態で強制送信処理を実行した。	オフライン状態で強制送信処理を実行してください。
EX_TIMEOUT	処理の実行時間が、指定したタイムアウト時間を超過した。	タイムアウト時間を調整してください。
EX_SPOOLER	スプーラー容量を超えて、印刷データが送信された。	プリンターとの通信が切れていないか、接続状態を確認してください。
JobNotFound	存在しない印刷ジョブ ID が指定された。	印刷ジョブ ID の指定先を確認してください。
Printing	印刷中の印刷ジョブ ID を指定して、getPrintJobStatus メソッドが実行された。	印刷完了後に処理を実行してください。

エラーコード	要因	対処方法
TooManyRequests	プリンターファームウェアの許容量を超えて、印刷データが送信された。	プリンターの処理がある程度進むまで時間を空けてから、再度データを送信してください。
RequestEntityTooLarge	プリンターファームウェアの許容量を超えるデータサイズの印刷ジョブが送信された。	印刷ジョブの内容を確認し、データサイズを下げて再度送信してください。
JobSpooling	指定した ID の印刷ジョブは、スプーラーで待機中。	印刷完了後に処理を実行してください。

❏ Display オブジェクト

エラーコード	要因	対処方法
EDSP_NOT_FOUND	• 接続形式や IP アドレスなどの指定が間違っている。 • 指定したデバイスが接続されていない。	• 接続形式や IP アドレスなどの指定値を確認してください。 • デバイスの接続状況を確認してください。
EDSP_NOT_OPEN	デバイスが破損している可能性がある。	プリンターを再起動し、再度処理を行ってください。 再発する場合は、購入先またはサービスサポートまで問い合わせてください。
EDSP_INVALID_WINDOW	未登録のウィンドウが指定された。	指定するウィンドウ番号を確認してください。
EDSP_UNRECOVERABLE	接続中のプリンターで復帰不可能エラーが発生した。	プリンターの電源を入れ直してください。 再発する場合は、購入先かサービスセンターに連絡してください。
EDSP_CUTTER	接続中のプリンターでオートカッターエラーが発生した。	エラー原因を取り除き、プリンターの電源を入れ直してください。
EDSP_MECHANICAL	接続中のプリンターでメカニカルエラーが発生した。	エラー原因を取り除き、プリンターの電源を入れ直してください。
EDSP_COVER_OPEN	接続中のプリンターのカバーが開いている。	プリンターのカバーを閉じてください。
EDSP_REC_EMPTY	接続中のプリンターの用紙なし。	用紙を補充してください。
EDSP_AUTOMATICAL	接続中のプリンターで自動復帰エラーが発生した。	プリンター復帰後に処理を実行してください。
EX_BADPORT	EPSON TMNet WebConfig で、ディスプレイの使用が無効に設定されている。	EPSON TMNet WebConfig の設定を確認してください。
EX_TIMEOUT	処理の実行時間が、指定したタイムアウト時間を超過した。	タイムアウト時間を調整してください。
EX_INVALID_VALUE	不正なパラメーターが渡された。	パラメーターの指定値を確認してください。

エラーコード	要因	対処方法
TooManyRequests	プリンターファームウェアの許容量を超えて、ディスプレイ表示データが送信された。	プリンターの処理がある程度進むまで時間を空けてから、再度データを送信してください。

ステータス

❏ プリンターステータス

定数 (status)	要因	対処方法
ASB_NO_RESPONSE	• プリンターの電源が入っていない。 • プリンターと接続されていない。	プリンターの電源や通信状態を確認してください。
ASB_BATTERY_OFFLINE	バッテリー残量少のため、オフライン状態。	バッテリーを交換するか、AC アダプターに接続してください。
ASB_OFF_LINE	オフライン状態。	カバーオープンや用紙切れなど、オフラインとなる要因を取り除いてください。
ASB_COVER_OPEN	カバーが開いている。	プリンターのカバーを閉じてください。
ASB_PAPER_FEED	紙送りボタンによる紙送り中。	紙送り停止後に処理を実行してください。
ASB_PANEL_SWITCH	パネルスイッチ操作中。	操作完了後に処理を実行してください。
ASB_MECHANICAL_ERR	メカニカルエラーが発生した。	エラー原因を取り除き、プリンターの電源を入れ直してください。
ASB_AUTOCUTTER_ERR	オートカッターエラーが発生した。	エラー原因を取り除き、プリンターの電源を入れ直してください。
ASB_UNRECOVER_ERR	復帰不可能エラーが発生した。	プリンターの電源を入れ直してください。 再発する場合は、購入先かサービスセンターに連絡してください。
ASB_AUTORECOVER_ERR	自動復帰エラーが発生した。	プリンター復帰後に処理を実行してください。
ASB_RECEIPT_NEAR_END	用紙が残り少ない。	用紙を交換してください。
ASB_RECEIPT_END	用紙なし。	用紙を補充してください。
ASB_BUZZER	ブザー鳴動中。	各プリンターのブザー鳴動条件を確認し、ブザー鳴動原因を取り除いてから処理を実行してください。
ASB_WAIT_REMOVE_LABEL	ラベルの剥離待ち中。	印刷済みのラベルを剥離紙から取り外してから処理を実行してください。
ASB_NO_LABEL	剥離待ちラベルなし。	印刷完了後に処理を実行してください。

定数 (status)	要因	対処方法
ASB_SPOOLER_IS_STOPPED	スプーラー容量を超えて、印刷データが送信された。	プリンターとの通信が切れていないか、接続状態を確認してください。

❏ バッテリーステータス

定数 (battery)	要因	対処方法
0x31XX	AC アダプターが接続されていない。	AC アダプターの接続状態を確認してください。
0xXX31	バッテリー残量少。	バッテリーを充電するか、AC アダプターに接続してください。
0xXX30	バッテリー残量なし。	バッテリーを充電するか、AC アダプターに接続してください。

Key Code一覧

定数	コード	定数	コード
VK_BACK	0x08	VK_L	0x4C
VK_TAB	0x09	VK_M	0x4D
VK_RETURN	0x0D	VK_N	0x4E
VK_SHIFT	0x10	VK_O	0x4F
VK_CONTROL	0x11	VK_P	0x50
VK_MENU	0x12	VK_Q	0x51
VK_CAPITAL	0x14	VK_R	0x52
VK_ESCAPE	0x1B	VK_S	0x53
VK_CONVERT	0x1C	VK_T	0x54
VK_NONCONVERT	0x1D	VK_U	0x55
VK_SPACE	0x20	VK_V	0x56
VK_PRIOR	0x21	VK_W	0x57
VK_NEXT	0x22	VK_X	0x58
VK_END	0x23	VK_Y	0x59
VK_HOME	0x24	VK_Z	0x5A
VK_LEFT	0x25	VK_MULTIPLY	0x6A
VK_UP	0x26	VK_ADD	0x6B
VK_RIGHT	0x27	VK_SUBTRACT	0x6D
VK_DOWN	0x28	VK_F1	0x70
VK_INSERT	0x2D	VK_F2	0x71
VK_DELETE	0x2E	VK_F3	0x72
VK_0	0x30	VK_F4	0x73
VK_1	0x31	VK_F5	0x74
VK_2	0x32	VK_F6	0x75
VK_3	0x33	VK_F7	0x76
VK_4	0x34	VK_F8	0x77
VK_5	0x35	VK_F9	0x78
VK_6	0x36	VK_F10	0x79
VK_7	0x37	VK_F11	0x7A
VK_8	0x38	VK_F12	0x7B
VK_9	0x39	VK_OEM_1	0xBA
VK_A	0x41	VK_OEM_PLUS	0xBB
VK_B	0x42	VK_OEM_COMMA	0xBC

定数	コード	定数	コード
VK_C	0x43	VK_OEM_MINUS	0xBD
VK_D	0x44	VK_OEM_PERIOD	0xBE
VK_E	0x45	VK_OEM_2	0xBF
VK_F	0x46	VK_OEM_3	0xC0
VK_G	0x47	VK_OEM_4	0xDB
VK_H	0x48	VK_OEM_5	0xDC
VK_I	0x49	VK_OEM_6	0xDD
VK_J	0x4A	VK_OEM_7	0xDE
VK_K	0x4B	VK_OEM_ATTN	0xF0

機器仕様

プリンターや周辺機器などのデバイス側に依存する API 使用制限や、パラメーターの設定値についての情報を掲載します。

✎	機器ごとの詳細仕様は、各プリンターの詳細取扱説明書を参照してください。

サポートオブジェクト一覧

プリンターごとの対応オブジェクトは、以下の通りです。

	ePOSDevice オブジェクト[*1]	デバイスオブジェクト共通	OtherPeripheral オブジェクト	CashChanger オブジェクト	CAT オブジェクト	Display オブジェクト[*1]	Keyboard オブジェクト	POSKeyboard オブジェクト	MSR オブジェクト	Printer オブジェクト[*1]	Scanner オブジェクト	SimpleSerial オブジェクト	DeviceHubTerminal オブジェクト	CommBoxManager オブジェクト	CommBox オブジェクト	ePOSDeviceConfiguration オブジェクト
TM-m10	✔	-	-	-	-	-	-	-	-	✔	-	-	-	-	-	-
TM-m30	✔	-	-	-	-	✔	-	-	-	✔	✔[*3]	-	-	-	-	-
TM-m30II	✔	-	-	-	-	✔	-	-	-	✔	✔	-	-	-	-	-
TM-m30II-H	✔	-	-	-	-	✔	-	-	-	✔	✔	-	-	-	-	-
TM-m30II-S	✔	-	-	-	-	✔	-	-	-	✔	✔	-	-	-	-	-
TM-m30II-SL	✔	-	-	-	-	✔	-	-	-	✔	✔	-	-	-	-	-
TM-T88VI	✔	-	-	-	-	✔	-	-	-	✔	✔[*4]	-	-	-	-	-
TM-T88VII	✔	-	-	-	-	✔	-	-	-	✔	✔	-	-	-	-	-
TM-L100	✔	-	-	-	-	✔	-	-	-	✔	✔	-	-	-	-	-
TM-T20II	✔	-	-	-	-	-	-	-	-	✔	-	-	-	-	-	-
TM-T20III	✔	-	-	-	-	-	-	-	-	✔	-	-	-	-	-	-
TM-T70II	✔	-	-	-	-	-	-	-	-	✔	-	-	-	-	-	-
TM-T88V	✔	-	-	-	-	-	-	-	-	✔	-	-	-	-	-	-
TM-T90II	✔	-	-	-	-	-	-	-	-	✔	-	-	-	-	-	-
TM-T90KP	✔	-	-	-	-	-	-	-	-	✔	-	-	-	-	-	-
TM-P20	✔	-	-	-	-	-	-	-	-	✔	-	-	-	-	-	-

	ePOSDevice オブジェクト*1	デバイスオブジェクト共通	OtherPeripheral オブジェクト	CashChanger オブジェクト	CAT オブジェクト	Display オブジェクト*1	Keyboard オブジェクト	POSKeyboard オブジェクト	MSR オブジェクト	Printer オブジェクト*1	Scanner オブジェクト	SimpleSerial オブジェクト	DeviceHubTerminal オブジェクト	CommBoxManager オブジェクト	CommBox オブジェクト	ePOSDeviceConfiguration オブジェクト
TM-P60II	✔	-	-	-	-	-	-	-	-	✔	-	-	-	-	-	-
TM-P80	✔	-	-	-	-	-	-	-	-	✔	-	-	-	-	-	-
TM-T70-i	✔	-	-	✔*5	-	✔	✔	-	✔*5	✔	✔*5	✔	-	✔	✔	-
TM-T88V-i	✔	-	-	✔*5	-	✔	✔	-	✔*5	✔	✔*5	✔	-	✔	✔	-
TM-T70II-DT	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
TM-T70II-DT2	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
TM-T88V-DT	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
TM-T88VI-DT2	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
TM-T70*2	-	-	-	-	-	-	-	-	-	✔	-	-	-	-	-	-
TM-T88IV*2	-	-	-	-	-	-	-	-	-	✔	-	-	-	-	-	-
TM-T90*2	-	-	-	-	-	-	-	-	-	✔	-	-	-	-	-	-
TM-L90*2	-	-	-	-	-	-	-	-	-	✔	-	-	-	-	-	-

*1: プリンターごとに API の使用可否や、使用制限があります。サポート API 一覧を参照してください。

*2: 子機プリンターでの制御にのみ対応しています。

*3:TM-m30 ファームウェア Ver.1.40ESC/POS 以降で対応しています。

*4:TM-T88VI ファームウェア Ver.40.50ESC/POS 以降で対応しています。

*5: デバイス制御プログラムでの制御には対応していません。

サポート API 一覧

各プリンターのサポート API を一覧表で掲載します。

表内記号の意味は、以下の通りです。

- ✔：対応している。
- ✔*：対応しているが、指定できるパラメーター設定値に制限がある。
- -：対応していない。

「✔*」の制限については、プリンター別サポート情報を参照してください。

ePOSDevice オブジェクト

単機能モデル（1）

API	TM-m10	TM-m30	TM-m30II	TM-m30II-H	TM-m30II-S	TM-m30II-SL	TM-T20II	TM-T20III
Constructor	✔	✔	✔	✔	✔	✔	✔	✔
connect メソッド	✔	✔	✔	✔	✔	✔	✔	✔
disconnect メソッド	✔	✔	✔	✔	✔	✔	✔	✔
isConnected メソッド	-	-	-	-	-	-	-	-
createDevice メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
deleteDevice メソッド	✔	✔	✔	✔	✔	✔	✔	✔
getAdmin メソッド	-	-	-	-	-	-	-	-
getLocation メソッド	-	-	-	-	-	-	-	-
sendOfscXml メソッド	-	-	-	-	-	-	-	-
getCommBoxManager メソッド	-	-	-	-	-	-	-	-
onreconnecting イベント	-	-	✔	✔	✔	✔	-	-
onreconnect イベント	-	-	✔	✔	✔	✔	-	-
ondisconnect イベント	-	-	✔	✔	✔	✔	-	-

単機能モデル(2)

API	TM-T88VI	TM-T88VII	TM-L100	TM-T70II	TM-T88V	TM-T90II	TM-T90KP (UB-R04)	TM-T90KP (UB-E04)
Constructor	✔	✔	✔	✔	✔	✔	✔	✔
connect メソッド	✔	✔	✔	✔	✔	✔	✔	✔
disconnect メソッド	✔	✔	✔	✔	✔	✔	✔	✔
isConnected メソッド	-	-	-	-	-	-	-	-
createDevice メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
deleteDevice メソッド	✔	✔	✔	✔	✔	✔	✔	✔
getAdmin メソッド	-	-	-	-	-	-	-	-
getLocation メソッド	-	-	-	-	-	-	-	-
sendOfscXml メソッド	-	-	-	-	-	-	-	-
getCommBoxManager メソッド	-	-	-	-	-	-	-	-
onreconnecting イベント	-	-	-	-	-	-	-	-
onreconnect イベント	-	-	-	-	-	-	-	-
ondisconnect イベント	-	-	-	-	-	-	-	-

モバイルモデル /TM-i シリーズ /TM-DT シリーズ

API	TM-P20	TM-P60II (Peeler)	TM-P80	TM-T70-i	TM-T88V-i	TM-T70II-DT	TM-T70II-DT2	TM-T88V-DT	TM-T88VI-DT2
Constructor	✔	✔	✔	✔	✔	✔	✔	✔	✔
connect メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
disconnect メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
isConnected メソッド	-	-	-	✔	✔	✔	✔	✔	✔
createDevice メソッド	✔*	✔*	✔*	✔*	✔*	✔	✔	✔	✔
deleteDevice メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
getAdmin メソッド	-	-	-	✔	✔	✔	✔	✔	✔
getLocation メソッド	-	-	-	✔	✔	✔	✔	✔	✔

API	TM-P20	TM-P60II (Peeler)	TM-P80	TM-T70-i	TM-T88V-i	TM-T70II-DT	TM-T70II-DT2	TM-T88V-DT	TM-T88VI-DT2
sendOfscXml メソッド	-	-	-	✔	✔	✔	✔	✔	✔
getCommBoxManager メソッド	-	-	-	✔	✔	✔	✔	✔	✔
onreconnecting イベント	-	-	-	✔	✔	✔	✔	✔	✔
onreconnect イベント	-	-	-	✔	✔	✔	✔	✔	✔
ondisconnect イベント	-	-	-	✔	✔	✔	✔	✔	✔

子機プリンター

API	TM-T70	TM-T88IV	TM-T90	TM-L90
Constructor	-	-	-	-
connect メソッド	-	-	-	-
disconnect メソッド	-	-	-	-
isConnected メソッド	-	-	-	-
createDevice メソッド	-	-	-	-
deleteDevice メソッド	-	-	-	-
getAdmin メソッド	-	-	-	-
getLocation メソッド	-	-	-	-
sendOfscXml メソッド	-	-	-	-
getCommBoxManager メソッド	-	-	-	-
onreconnecting イベント	-	-	-	-
onreconnect イベント	-	-	-	-
ondisconnect イベント	-	-	-	-

Display オブジェクト

単機能モデル(1)

API	TM-m10	TM-m30	TM-m30II	TM-m30II-H	TM-m30II-S	TM-m30II-SL	TM-T20II	TM-T20III
createWindow メソッド	-	✔	✔	✔	✔	✔	-	-
destroyWindow メソッド	-	✔	✔	✔	✔	✔	-	-
setCurrentWindow メソッド	-	✔	✔	✔	✔	✔	-	-
clearWindow メソッド	-	✔	✔	✔	✔	✔	-	-
addCreateScreen メソッド	-	-	-	✔	✔	✔	-	-
addCreateScreenCustom メソッド	-	-	-	✔	✔	✔	-	-
createTextArea メソッド	-	-	-	✔	✔	✔	-	-
destroyTextArea メソッド	-	-	-	✔	✔	✔	-	-
setCurrentTextArea メソッド	-	-	-	✔	✔	✔	-	-
clearTextArea メソッド	-	-	-	✔	✔	✔	-	-
setCursorPosition メソッド	-	✔	✔	✔	✔	✔	-	-
moveCursorPosition メソッド	-	✔	✔	✔	✔	✔	-	-
setCursorType メソッド	-	✔	✔	✔	✔	✔	-	-
addText メソッド	-	✔*	✔*	✔	✔	✔	-	-
addReverseText メソッド	-	✔	✔	✔	✔	✔	-	-
addMarquee メソッド	-	✔	✔	✔	✔	✔	-	-
setBlink メソッド	-	✔	✔	✔	✔	✔	-	-
setBrightness メソッド	-	✔	✔	✔	✔	✔	-	-
addBackgroundColor メソッド	-	-	-	✔	✔	✔	-	-
addStartSlideShow メソッド	-	-	-	✔	✔	✔	-	-
addStopSlideShow メソッド	-	-	-	✔	✔	✔	-	-
addDownloadImage メソッド	-	-	-	✔	✔	✔	-	-
addRegisterDownloadImage メソッド	-	-	-	✔	✔	✔	-	-
addNVImage メソッド	-	-	-	✔	✔	✔	-	-

API	TM-m10	TM-m30	TM-m30II	TM-m30II-H	TM-m30II-S	TM-m30II-SL	TM-T20II	TM-T20III
addClearImage メソッド	-	-	-	✔	✔	✔	-	-
addSymbol メソッド	-	-	-	✔	✔	✔	-	-
addClearSymbol メソッド	-	-	-	✔	✔	✔	-	-
showClock メソッド	-	-	-	-	-	-	-	-
addCommand メソッド	-	✔	✔	✔	✔	✔	-	-
send メソッド	-	✔	✔	✔	✔	✔	-	-
reset メソッド	-	✔	✔	✔	✔	✔	-	-
onreceive イベント	-	✔	✔	✔	✔	✔	-	-

単機能モデル(2)

API	TM-T88VI	TM-T88VII	TM-L100	TM-T70II	TM-T88V	TM-T90II	TM-T90KP (UB-R04)	TM-T90KP (UB-E04)
createWindow メソッド	✔	✔	✔	-	-	-	-	-
destroyWindow メソッド	✔	✔	✔	-	-	-	-	-
setCurrentWindow メソッド	✔	✔	✔	-	-	-	-	-
clearWindow メソッド	✔	✔	✔	-	-	-	-	-
addCreateScreen メソッド	-	✔	✔	-	-	-	-	-
addCreateScreenCustom メソッド	-	✔	✔	-	-	-	-	-
createTextArea メソッド	-	✔	✔	-	-	-	-	-
destroyTextArea メソッド	-	✔	✔	-	-	-	-	-
setCurrentTextArea メソッド	-	✔	✔	-	-	-	-	-
clearTextArea メソッド	-	✔	✔	-	-	-	-	-
setCursorPosition メソッド	✔	✔	✔	-	-	-	-	-
moveCursorPosition メソッド	✔	✔	✔	-	-	-	-	-
setCursorType メソッド	✔	✔	✔	-	-	-	-	-
addText メソッド	✔*	✔	✔	-	-	-	-	-

API	TM-T88VI	TM-T88VII	TM-L100	TM-T70II	TM-T88V	TM-T90II	TM-T90KP (UB-R04)	TM-T90KP (UB-E04)
addReverseText メソッド	✔	✔	✔	-	-	-	-	-
addMarquee メソッド	✔	✔	✔	-	-	-	-	-
setBlink メソッド	✔	✔	✔	-	-	-	-	-
setBrightness メソッド	✔	✔	✔	-	-	-	-	-
addBackgroundColor メソッド	-	✔	✔	-	-	-	-	-
addStartSlideShow メソッド	-	✔	✔	-	-	-	-	-
addStopSlideShow メソッド	-	✔	✔	-	-	-	-	-
addDownloadImage メソッド	-	✔	✔	-	-	-	-	-
addRegisterDownloadImage メソッド	-	✔	✔	-	-	-	-	-
addNVImage メソッド	-	✔	✔	-	-	-	-	-
addClearImage メソッド	-	✔	✔	-	-	-	-	-
addSymbol メソッド	-	✔	✔	-	-	-	-	-
addClearSymbol メソッド	-	✔	✔	-	-	-	-	-
showClock メソッド	✔	✔	✔	-	-	-	-	-
addCommand メソッド	✔	✔	✔	-	-	-	-	-
send メソッド	✔	✔	✔	-	-	-	-	-
reset メソッド	✔	✔	✔	-	-	-	-	-
onreceive イベント	✔	✔	✔	-	-	-	-	-

モバイルモデル /TM-i シリーズ /TM-DT シリーズ

API	TM-P20	TM-P60II (Peeler)	TM-P80	TM-T70-i	TM-T88V-i	TM-T70II-DT	TM-T70II-DT2	TM-T88V-DT	TM-T88VI-DT2
createWindow メソッド	-	-	-	✔	✔	✔	✔	✔	✔
destroyWindow メソッド	-	-	-	✔	✔	✔	✔	✔	✔
setCurrentWindow メソッド	-	-	-	✔	✔	✔	✔	✔	✔

API	TM-P20	TM-P60II (Peeler)	TM-P80	TM-T70-i	TM-T88V-i	TM-T70II-DT	TM-T70II-DT2	TM-T88V-DT	TM-T88VI-DT2
clearWindow メソッド	-	-	-	✔	✔	✔	✔	✔	✔
addCreateScreen メソッド	-	-	-	-	-	-	✔	-	✔
addCreateScreenCustom メソッド	-	-	-	-	-	-	✔	-	✔
createTextArea メソッド	-	-	-	-	-	-	✔	-	✔
destroyTextArea メソッド	-	-	-	-	-	-	✔	-	✔
setCurrentTextArea メソッド	-	-	-	-	-	-	✔	-	✔
clearTextArea メソッド	-	-	-	-	-	-	✔	-	✔
setCursorPosition メソッド	-	-	-	✔	✔	✔	✔	✔	✔
moveCursorPosition メソッド	-	-	-	✔	✔	✔	✔	✔	✔
setCursorType メソッド	-	-	-	✔	✔	✔	✔	✔	✔
addText メソッド	-	-	-	✔*	✔*	✔*	✔*	✔*	✔*
addReverseText メソッド	-	-	-	✔	✔	✔	✔	✔	✔
addMarquee メソッド	-	-	-	✔	✔	✔	✔	✔	✔
setBlink メソッド	-	-	-	✔	✔	✔	✔	✔	✔
setBrightness メソッド	-	-	-	✔	✔	✔	✔	✔	✔
addBackgroundColor メソッド	-	-	-	-	-	-	✔	-	✔
addStartSlideShow メソッド	-	-	-	-	-	-	✔	-	✔
addStopSlideShow メソッド	-	-	-	-	-	-	✔	-	✔
addDownloadImage メソッド	-	-	-	-	-	-	✔	-	✔
addRegisterDownloadImage メソッド	-	-	-	-	-	-	✔	-	✔
addNVImage メソッド	-	-	-	-	-	-	✔	-	✔
addClearImage メソッド	-	-	-	-	-	-	✔	-	✔
addSymbol メソッド	-	-	-	-	-	-	✔	-	✔
addClearSymbol メソッド	-	-	-	-	-	-	✔	-	✔
showClock メソッド	-	-	-	✔	✔	✔	✔	✔	✔
addCommand メソッド	-	-	-	✔	✔	✔	✔	✔	✔

API	TM-P20	TM-P60II (Peeler)	TM-P80	TM-T70-i	TM-T88V-i	TM-T70II-DT	TM-T70II-DT2	TM-T88V-DT	TM-T88VI-DT2
send メソッド	-	-	-	✔	✔	✔	✔	✔	✔
reset メソッド	-	-	-	✔	✔	✔	✔	✔	✔
onreceive イベント	-	-	-	✔	✔	✔	✔	✔	✔

子機プリンター

API	TM-T70	TM-T88IV	TM-T90	TM-L90
createWindow メソッド	-	-	-	-
destroyWindow メソッド	-	-	-	-
setCurrentWindow メソッド	-	-	-	-
clearWindow メソッド	-	-	-	-
addCreateScreen メソッド	-	-	-	-
addCreateScreenCustom メソッド	-	-	-	-
createTextArea メソッド	-	-	-	-
destroyTextArea メソッド	-	-	-	-
setCurrentTextArea メソッド	-	-	-	-
clearTextArea メソッド	-	-	-	-
setCursorPosition メソッド	-	-	-	-
moveCursorPosition メソッド	-	-	-	-
setCursorType メソッド	-	-	-	-
addText メソッド	-	-	-	-
addReverseText メソッド	-	-	-	-
addMarquee メソッド	-	-	-	-
setBlink メソッド	-	-	-	-
setBrightness メソッド	-	-	-	-
addBackgroundColor メソッド	-	-	-	-
addStartSlideShow メソッド	-	-	-	-

API	TM-T70	TM-T88IV	TM-T90	TM-L90
addStopSlideShow メソッド	-	-	-	-
addDownloadImage メソッド	-	-	-	-
addRegisterDownloadImage メソッド	-	-	-	-
addNVImage メソッド	-	-	-	-
addClearImage メソッド	-	-	-	-
addSymbol メソッド	-	-	-	-
addClearSymbol メソッド	-	-	-	-
showClock メソッド	-	-	-	-
addCommand メソッド	-	-	-	-
send メソッド	-	-	-	-
reset メソッド	-	-	-	-
onreceive イベント	-	-	-	-

Printer オブジェクト

単機能モデル(1)

API	TM-m10	TM-m30	TM-m30II	TM-m30II-H	TM-m30II-S	TM-m30II-SL	TM-T20II	TM-T20III
addTextAlign メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addTextLineSpace メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addTextRotate メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addText メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addTextLang メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addTextFont メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addTextSmooth メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addTextDouble メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addTextSize メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addTextStyle メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addTextPosition メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addTextVPosition メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addFeedUnit メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addFeedLine メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addFeedPosition メソッド	-	-	-	-	-	-	-	-
addFeed メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addImage メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addLogo メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addBarcode メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addSymbol メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addHLine メソッド	-	-	-	-	-	-	-	-
addVLineBegin メソッド	-	-	-	-	-	-	-	-
addVLineEnd メソッド	-	-	-	-	-	-	-	-
addPageBegin メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addPageEnd メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addPageArea メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addPageDirection メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addPagePosition メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addPageLine メソッド	✔*	✔*	✔*	✔*	✔*	✔*	-	-

API	TM-m10	TM-m30	TM-m30II	TM-m30II-H	TM-m30II-S	TM-m30II-SL	TM-T20II	TM-T20III
addPageRectangle メソッド	✔*	✔*	✔*	✔*	✔*	✔*	-	-
addRotateBegin メソッド	-	-	✔	✔	✔	✔	-	-
addRotateEnd メソッド	-	-	✔	✔	✔	✔	-	-
addCut メソッド	✔*	✔	✔*	✔*	✔*	✔*	✔*	✔*
addPulse メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addSound メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addLayout メソッド	-	-	-	-	-	-	-	-
recover メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addRecovery メソッド	✔	✔	✔	✔	✔	✔	✔	✔
reset メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addReset メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addCommand メソッド	✔	✔	✔	✔	✔	✔	✔	✔
send メソッド	✔	✔	✔	✔	✔	✔	✔	✔
print メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
getPrintJobStatus メソッド	-	-	-	-	-	-	-	-
startMonitor メソッド	✔	✔	✔	✔	✔	✔	✔	✔
stopMonitor メソッド	✔	✔	✔	✔	✔	✔	✔	✔
halftone プロパティー	✔	✔	✔	✔	✔	✔	✔	✔
brightness プロパティー	✔	✔	✔	✔	✔	✔	✔	✔
force プロパティー	✔	✔	✔	✔	✔	✔	✔	✔
timeout プロパティー	✔	✔	✔	✔	✔	✔	✔	✔
interval プロパティー	✔	✔	✔	✔	✔	✔	✔	✔
drawerOpenLevel プロパティー	✔	✔	✔	✔	✔	✔	✔	✔
message プロパティー	✔	✔	✔	✔	✔	✔	✔	✔
onreceive イベント	✔	✔	✔	✔	✔	✔	✔	✔
onstatuschange イベント	✔	✔	✔	✔	✔	✔	✔	✔
onbatterystatuschange イベント	-	-	-	-	-	-	-	-
ononline イベント	✔	✔	✔	✔	✔	✔	✔	✔
onoffline イベント	✔	✔	✔	✔	✔	✔	✔	✔
onpoweroff イベント	✔	✔	✔	✔	✔	✔	✔	✔
oncoverok イベント	✔	✔	✔	✔	✔	✔	✔	✔

API	TM-m10	TM-m30	TM-m30II	TM-m30II-H	TM-m30II-S	TM-m30II-SL	TM-T20II	TM-T20III
oncoveropen イベント	✔	✔	✔	✔	✔	✔	✔	✔
onpaperok イベント	✔	✔	✔	✔	✔	✔	✔	✔
onpapernearend イベント	✔	✔	✔	✔	✔	✔	✔	✔
onpaperend イベント	✔	✔	✔	✔	✔	✔	✔	✔
ondrawerclosed イベント	✔	✔	✔	✔	✔	✔	✔	✔
ondraweropen イベント	✔	✔	✔	✔	✔	✔	✔	✔
onbatteryok イベント	-	-	-	-	-	-	-	-
onbatterylow イベント	-	-	-	-	-	-	-	-

単機能モデル(2)

API	TM-T88VI	TM-T88VII	TM-L100	TM-T70II	TM-T88V	TM-T90II	TM-T90KP (UB-R04)	TM-T90KP (UB-E04)
addTextAlign メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addTextLineSpace メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addTextRotate メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addText メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addTextLang メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addTextFont メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addTextSmooth メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addTextDouble メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addTextSize メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addTextStyle メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addTextPosition メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addTextVPosition メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addFeedUnit メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addFeedLine メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addFeedPosition メソッド	-	-	✔*	-	-	-	-	-
addFeed メソッド	✔	✔	✔	✔	✔	✔	✔	✔

API	TM-T88VI	TM-T88VII	TM-L100	TM-T70II	TM-T88V	TM-T90II	TM-T90KP (UB-R04)	TM-T90KP (UB-E04)
addImage メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addLogo メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addBarcode メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addSymbol メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addHLine メソッド	-	-	-	-	-	-	✔	✔
addVLineBegin メソッド	-	-	-	-	-	-	✔	✔
addVLineEnd メソッド	-	-	-	-	-	-	✔	✔
addPageBegin メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addPageEnd メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addPageArea メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addPageDirection メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addPagePosition メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addPageLine メソッド	✔*	✔*	✔*	-	-	-	✔	✔
addPageRectangle メソッド	✔*	✔*	✔*	-	-	-	✔	✔
addRotateBegin メソッド	-	✔	✔	-	-	-	-	-
addRotateEnd メソッド	-	✔	✔	-	-	-	-	-
addCut メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔	✔
addPulse メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addSound メソッド	✔*	✔*	✔*	✔*	✔*	-	✔*	✔*
addLayout メソッド	-	-	-	-	-	-	-	-
recover メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addRecovery メソッド	✔	✔	✔	✔	✔	✔	✔	✔
reset メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addReset メソッド	✔	✔	✔	✔	✔	✔	✔	✔
addCommand メソッド	✔	✔	✔	✔	✔	✔	✔	✔
send メソッド	✔	✔	✔	✔	✔	✔	✔	✔
print メソッド	✔	✔	✔	✔*	✔	✔	✔*	✔*
getPrintJobStatus メソッド	✔	✔	✔	-	-	-	-	-
startMonitor メソッド	✔	✔	✔	✔	✔	✔	-	-
stopMonitor メソッド	✔	✔	✔	✔	✔	✔	-	-

API	TM-T88VI	TM-T88VII	TM-L100	TM-T70II	TM-T88V	TM-T90II	TM-T90KP (UB-R04)	TM-T90KP (UB-E04)
halftone プロパティー	✔	✔	✔	✔	✔	✔	✔	✔
brightness プロパティー	✔	✔	✔	✔	✔	✔	✔	✔
force プロパティー	✔	✔	✔	✔	✔	✔	✔	✔
timeout プロパティー	✔	✔	✔	✔	✔	✔	✔	✔
interval プロパティー	✔	✔	✔	✔	✔	✔	✔	✔
drawerOpenLevel プロパティー	✔	✔	✔	✔	✔	✔	-	-
message プロパティー	✔	✔	✔	✔	✔	✔	✔	✔
onreceive イベント	✔	✔	✔	✔	✔	✔	✔	✔
onstatuschange イベント	✔	✔	✔	✔	✔	✔	✔	✔
onbatterystatuschange イベント	-	-	-	-	-	-	-	-
ononline イベント	✔	✔	✔	✔	✔	✔	✔	✔
onoffline イベント	✔	✔	✔	✔	✔	✔	✔	✔
onpoweroff イベント	✔	✔	✔	✔	✔	✔	✔	✔
oncoverok イベント	✔	✔	✔	✔	✔	✔	✔	✔
oncoveropen イベント	✔	✔	✔	✔	✔	✔	✔	✔
onpaperok イベント	✔	✔	✔	✔	✔	✔	✔	✔
onpapernearend イベント	✔	✔	✔	✔	✔	✔	✔	✔
onpaperend イベント	✔	✔	✔	✔	✔	✔	✔	✔
ondrawerclosed イベント	✔	✔	✔	✔	✔	✔	-	-
ondraweropen イベント	✔	✔	✔	✔	✔	✔	-	-
onbatteryok イベント	-	-	-	-	-	-	-	-
onbatterylow イベント	-	-	-	-	-	-	-	-

モバイルモデル /TM-i シリーズ /TM-DT シリーズ

API	TM-P20	TM-P60II (Peeler)	TM-P80	TM-T70-i	TM-T88V-i	TM-T70II-DT	TM-T70II-DT2	TM-T88V-DT	TM-T88VI-DT2
addTextAlign メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addTextLineSpace メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addTextRotate メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addText メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addTextLang メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addTextFont メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addTextSmooth メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addTextDouble メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addTextSize メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addTextStyle メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addTextPosition メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addTextVPosition メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addFeedUnit メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addFeedLine メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addFeedPosition メソッド	✔*	✔	✔*	-	-	-	-	-	-
addFeed メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addImage メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addLogo メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addBarcode メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addSymbol メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addHLine メソッド	-	-	-	-	-	-	-	-	-
addVLineBegin メソッド	-	-	-	-	-	-	-	-	-
addVLineEnd メソッド	-	-	-	-	-	-	-	-	-
addPageBegin メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addPageEnd メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addPageArea メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addPageDirection メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addPagePosition メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addPageLine メソッド	✔*	✔*	✔*	-	-	-	-	-	-

API	TM-P20	TM-P60II (Peeler)	TM-P80	TM-T70-i	TM-T88V-i	TM-T70II-DT	TM-T70II-DT2	TM-T88V-DT	TM-T88VI-DT2
addPageRectangle メソッド	✔*	✔*	✔*	-	-	-	-	-	-
addRotateBegin メソッド	-	-	-	-	-	-	-	-	-
addRotateEnd メソッド	-	-	-	-	-	-	-	-	-
addCut メソッド	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*	✔*
addPulse メソッド	-	-	-	✔	✔	✔	✔	✔	✔
addSound メソッド	✔*	✔*	✔*	-	✔*	✔*	✔*	✔*	✔*
addLayout メソッド	✔*	✔	✔*	-	-	-	-	-	-
recover メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addRecovery メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
reset メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addReset メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
addCommand メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
send メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
print メソッド	✔*	✔*	✔*	✔*	✔	✔	✔	✔	✔
getPrintJobStatus メソッド	-	-	-	✔	✔	✔	✔	✔	✔
startMonitor メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
stopMonitor メソッド	✔	✔	✔	✔	✔	✔	✔	✔	✔
halftone プロパティー	✔	✔	✔	✔	✔	✔	✔	✔	✔
brightness プロパティー	✔	✔	✔	✔	✔	✔	✔	✔	✔
force プロパティー	✔	✔	✔	✔	✔	✔	✔	✔	✔
timeout プロパティー	✔	✔	✔	✔	✔	✔	✔	✔	✔
interval プロパティー	✔	✔	✔	✔	✔	✔	✔	✔	✔
drawerOpenLevel プロパティー	-	-	-	✔	✔	✔	✔	✔	✔
message プロパティー	✔	✔	✔	✔	✔	✔	✔	✔	✔
onreceive イベント	✔	✔	✔	✔	✔	✔	✔	✔	✔
onstatuschange イベント	✔	✔	✔	✔	✔	✔	✔	✔	✔
onbatterystatuschange イベント	✔	✔	✔	-	-	-	-	-	-
ononline イベント	✔	✔	✔	✔	✔	✔	✔	✔	✔
onoffline イベント	✔	✔	✔	✔	✔	✔	✔	✔	✔

API	TM-P20	TM-P60II (Peeler)	TM-P80	TM-T70-i	TM-T88V-i	TM-T70II-DT	TM-T70II-DT2	TM-T88V-DT	TM-T88VI-DT2
onpoweroff イベント	✔	✔	✔	✔	✔	✔	✔	✔	✔
oncoverok イベント	✔	✔	✔	✔	✔	✔	✔	✔	✔
oncoveropen イベント	✔	✔	✔	✔	✔	✔	✔	✔	✔
onpaperok イベント	✔	✔	✔	✔	✔	✔	✔	✔	✔
onpapernearend イベント	✔	✔	✔	✔	✔	✔	✔	✔	✔
onpaperend イベント	✔	✔	✔	✔	✔	✔	✔	✔	✔
ondrawerclosed イベント	-	-	-	✔	✔	✔	✔	✔	✔
ondraweropen イベント	-	-	-	✔	✔	✔	✔	✔	✔
onbatteryok イベント	✔	✔	✔	-	-	-	-	-	-
onbatterylow イベント	✔	✔	✔	-	-	-	-	-	-

子機プリンター

API	TM-T70	TM-T88IV	TM-T90	TM-L90
addTextAlign メソッド	✔	✔	✔	✔
addTextLineSpace メソッド	✔	✔	✔	✔
addTextRotate メソッド	✔	✔	✔	✔
addText メソッド	✔	✔	✔	✔
addTextLang メソッド	✔*	✔*	✔*	✔*
addTextFont メソッド	✔*	✔*	✔*	✔*
addTextSmooth メソッド	✔	✔	✔	✔
addTextDouble メソッド	✔	✔	✔	✔
addTextSize メソッド	✔	✔	✔	✔
addTextStyle メソッド	✔*	✔*	✔*	✔*
addTextPosition メソッド	✔	✔	✔	✔
addTextVPosition メソッド	✔	✔	✔	✔
addFeedUnit メソッド	✔	✔	✔	✔
addFeedLine メソッド	✔	✔	✔	✔
addFeedPosition メソッド	-	-	-	✔
addFeed メソッド	✔	✔	✔	✔

API	TM-T70	TM-T88IV	TM-T90	TM-L90
addImage メソッド	✔*	✔*	✔*	✔*
addLogo メソッド	✔	✔	✔	✔
addBarcode メソッド	✔*	✔*	✔*	✔*
addSymbol メソッド	✔*	✔*	✔*	✔*
addHLine メソッド	-	-	-	-
addVLineBegin メソッド	-	-	-	-
addVLineEnd メソッド	-	-	-	-
addPageBegin メソッド	✔	✔	✔	✔
addPageEnd メソッド	✔	✔	✔	✔
addPageArea メソッド	✔	✔	✔	✔
addPageDirection メソッド	✔	✔	✔	✔
addPagePosition メソッド	✔	✔	✔	✔
addPageLine メソッド	-	-	-	-
addPageRectangle メソッド	-	-	-	-
addRotateBegin メソッド	-	-	-	-
addRotateEnd メソッド	-	-	-	-
addCut メソッド	✔*	✔*	✔*	✔*
addPulse メソッド	✔	✔	✔	✔
addSound メソッド	-	-	-	-
addLayout メソッド	-	-	-	-
recover メソッド	✔	✔	✔	✔
addRecovery メソッド	✔	✔	✔	✔
reset メソッド	✔	✔	✔	✔
addReset メソッド	✔	✔	✔	✔
addCommand メソッド	✔	✔	✔	✔
send メソッド	✔	✔	✔	✔
print メソッド	✔*	✔	✔	✔
getPrintJobStatus メソッド	-	-	-	-
startMonitor メソッド	✔	✔	✔	✔
stopMonitor メソッド	✔	✔	✔	✔
halftone プロパティー	✔	✔	✔	✔
brightness プロパティー	✔	✔	✔	✔
force プロパティー	✔	✔	✔	✔

API	TM-T70	TM-T88IV	TM-T90	TM-L90
timeout プロパティー	✔	✔	✔	✔
interval プロパティー	✔	✔	✔	✔
drawerOpenLevel プロパティー	✔	✔	✔	✔
message プロパティー	✔	✔	✔	✔
onreceive イベント	✔	✔	✔	✔
onstatuschange イベント	✔	✔	✔	✔
onbatterystatuschange イベント	-	-	-	-
ononline イベント	✔	✔	✔	✔
onoffline イベント	✔	✔	✔	✔
onpoweroff イベント	✔	✔	✔	✔
oncoverok イベント	✔	✔	✔	✔
oncoveropen イベント	✔	✔	✔	✔
onpaperok イベント	✔	✔	✔	✔
onpapernearend イベント	✔	✔	✔	✔
onpaperend イベント	✔	✔	✔	✔
ondrawerclosed イベント	✔	✔	✔	✔
ondraweropen イベント	✔	✔	✔	✔
onbatteryok イベント	-	-	-	-
onbatterylow イベント	-	-	-	-

プリンター別サポート情報

TM-m10

プリンター仕様

項目	仕様
初期改行量	30 ドット
ページモード初期領域	420 x 2400 ドット
ページモード最大領域	420 x 2400 ドット
Font A のベースライン	文字の上端から 21 ドット目
Font B のベースライン	文字の上端から 21 ドット目
Font C のベースライン	文字の上端から 16 ドット目

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_PRINTER
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B • FONT_C

API	パラメーター	指定可能な設定値
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B • FONT_C
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED • SYMBOL_AZTECCODE_FULLRANGE • SYMBOL_AZTECCODE_COMPACT • SYMBOL_DATAMATRIX_SQUARE • SYMBOL_DATAMATRIX_RECTANGLE_8 • SYMBOL_DATAMATRIX_RECTANGLE_12 • SYMBOL_DATAMATRIX_RECTANGLE_16
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • 5 ～ 95 の整数 • LEVEL_DEFAULT

API	パラメーター	指定可能な設定値
addPageLine メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addPageRectangle メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 1 ～ 255 • undefined
print メソッド	canvas	• canvas
	cut	• true • false • undefined
	mode	• MODE_MONO • undefined
	printjobid	• String

TM-m30

プリンター仕様

項目	58mm 仕様	80mm 仕様
初期改行量	30 ドット	
ページモード初期領域	420 x 2400 ドット	576 x 2400 ドット
ページモード最大領域	420 x 2400 ドット	576 x 2400 ドット
Font A のベースライン	文字の上端から 21 ドット目	
Font B のベースライン	文字の上端から 21 ドット目	
Font C のベースライン	文字の上端から 16 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_DISPLAY • DEVICE_TYPE_PRINTER • DEVICE_TYPE_SCANNER
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addText メソッド	data	• data
	lang	• "en"（初期値） • "ja" • "undefined"
	x	• 1 ～ 44 • "undefined"
	y	• 1 ～ 19 • "undefined"
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B • FONT_C

API	パラメーター	指定可能な設定値
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B • FONT_C
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED • SYMBOL_AZTECCODE_FULLRANGE • SYMBOL_AZTECCODE_COMPACT • SYMBOL_DATAMATRIX_SQUARE • SYMBOL_DATAMATRIX_RECTANGLE_8 • SYMBOL_DATAMATRIX_RECTANGLE_12 • SYMBOL_DATAMATRIX_RECTANGLE_16
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • 5 ～ 95 の整数 • LEVEL_DEFAULT

API	パラメーター	指定可能な設定値
addPageLine メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addPageRectangle メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 1 ～ 255 • undefined
print メソッド	canvas	• canvas
	cut	• true • false • undefined
	mode	• MODE_MONO • undefined
	printjobid	• String

TM-m30II

プリンター仕様

項目	58mm 仕様	80mm 仕様
初期改行量	30 ドット	
ページモード初期領域	420 x 4800 ドット	576 x 4800 ドット
ページモード最大領域	420 x 4800 ドット	576 x 4800 ドット
Font A のベースライン	文字の上端から 21 ドット目	
Font B のベースライン	文字の上端から 21 ドット目	
Font C のベースライン	文字の上端から 16 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_DISPLAY • DEVICE_TYPE_PRINTER • DEVICE_TYPE_SCANNER
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addText メソッド	data	• data
	lang	• "en"（初期値） • "ja" • "mul" • "undefined"
	x	• 1 ～ 44 • "undefined"
	y	• 1 ～ 19 • "undefined"
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp • mul

API	パラメーター	指定可能な設定値
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B • FONT_C
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_CODE128_AUTO • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B • FONT_C
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED • SYMBOL_AZTECCODE_FULLRANGE • SYMBOL_AZTECCODE_COMPACT • SYMBOL_DATAMATRIX_SQUARE • SYMBOL_DATAMATRIX_RECTANGLE_8 • SYMBOL_DATAMATRIX_RECTANGLE_12 • SYMBOL_DATAMATRIX_RECTANGLE_16
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • 5 ～ 95 の整数 • LEVEL_DEFAULT

API	パラメーター	指定可能な設定値
addPageLine メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addPageRectangle メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 1 ～ 255 • undefined
print メソッド	canvas	• canvas
	cut	• true • false • undefined
	mode	• MODE_MONO • undefined
	printjobid	• String

TM-m30II-H

プリンター仕様

項目	58mm 仕様	80mm 仕様
初期改行量	30 ドット	
ページモード初期領域	420 x 4800 ドット	576 x 4800 ドット
ページモード最大領域	420 x 4800 ドット	576 x 4800 ドット
Font A のベースライン	文字の上端から 21 ドット目	
Font B のベースライン	文字の上端から 21 ドット目	
Font C のベースライン	文字の上端から 16 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_DISPLAY • DEVICE_TYPE_PRINTER • DEVICE_TYPE_SCANNER
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp • mul
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B • FONT_C

API	パラメーター	指定可能な設定値
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_CODE128_AUTO • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B • FONT_C
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED • SYMBOL_AZTECCODE_FULLRANGE • SYMBOL_AZTECCODE_COMPACT • SYMBOL_DATAMATRIX_SQUARE • SYMBOL_DATAMATRIX_RECTANGLE_8 • SYMBOL_DATAMATRIX_RECTANGLE_12 • SYMBOL_DATAMATRIX_RECTANGLE_16
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • 5 ～ 95 の整数 • LEVEL_DEFAULT

API	パラメーター	指定可能な設定値
addPageLine メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addPageRectangle メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 1 ～ 255 • undefined
print メソッド	canvas	• canvas
	cut	• true • false • undefined
	mode	• MODE_MONO • undefined
	printjobid	• String

TM-m30II-S

プリンター仕様

項目	58mm 仕様	80mm 仕様
初期改行量	30 ドット	
ページモード初期領域	420 x 4800 ドット	576 x 4800 ドット
ページモード最大領域	420 x 4800 ドット	576 x 4800 ドット
Font A のベースライン	文字の上端から 21 ドット目	
Font B のベースライン	文字の上端から 21 ドット目	
Font C のベースライン	文字の上端から 16 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_DISPLAY • DEVICE_TYPE_PRINTER • DEVICE_TYPE_SCANNER
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp • mul
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B • FONT_C

API	パラメーター	指定可能な設定値
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_CODE128_AUTO • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B • FONT_C
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED • SYMBOL_AZTECCODE_FULLRANGE • SYMBOL_AZTECCODE_COMPACT • SYMBOL_DATAMATRIX_SQUARE • SYMBOL_DATAMATRIX_RECTANGLE_8 • SYMBOL_DATAMATRIX_RECTANGLE_12 • SYMBOL_DATAMATRIX_RECTANGLE_16
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • 5 ～ 95 の整数 • LEVEL_DEFAULT

API	パラメーター	指定可能な設定値
addPageLine メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addPageRectangle メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 1 ～ 255 • undefined
print メソッド	canvas	• canvas
	cut	• true • false • undefined
	mode	• MODE_MONO • undefined
	printjobid	• String

TM-m30II-SL

プリンター仕様

項目	58mm 仕様	80mm 仕様
初期改行量	30 ドット	
ページモード初期領域	420 x 4800 ドット	576 x 4800 ドット
ページモード最大領域	420 x 4800 ドット	576 x 4800 ドット
Font A のベースライン	文字の上端から 21 ドット目	
Font B のベースライン	文字の上端から 21 ドット目	
Font C のベースライン	文字の上端から 16 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_DISPLAY • DEVICE_TYPE_PRINTER • DEVICE_TYPE_SCANNER
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp • mul
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B • FONT_C

API	パラメーター	指定可能な設定値
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_CODE128_AUTO • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B • FONT_C
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED • SYMBOL_AZTECCODE_FULLRANGE • SYMBOL_AZTECCODE_COMPACT • SYMBOL_DATAMATRIX_SQUARE • SYMBOL_DATAMATRIX_RECTANGLE_8 • SYMBOL_DATAMATRIX_RECTANGLE_12 • SYMBOL_DATAMATRIX_RECTANGLE_16
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • 5 ～ 95 の整数 • LEVEL_DEFAULT

API	パラメーター	指定可能な設定値
addPageLine メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addPageRectangle メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 1 ～ 255 • undefined
print メソッド	canvas	• canvas
	cut	• true • false • undefined
	mode	• MODE_MONO • undefined
	printjobid	• String

TM-T20II

プリンター仕様

項目	58mm 仕様	80mm 仕様
初期改行量	30 ドット	
ページモード初期領域	420 x 831 ドット	576 x 831 ドット
ページモード最大領域	420 x 1662 ドット	576 x 1662 ドット
Font A のベースライン	文字の上端から 21 ドット目	
Font B のベースライン	文字の上端から 16 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_PRINTER
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined

API	パラメーター	指定可能な設定値
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • undefined
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • LEVEL_DEFAULT
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 1 ～ 255 • undefined

API	パラメーター	指定可能な設定値
print メソッド	canvas	• canvas
	cut	• true • false • undefined
	mode	• MODE_MONO • undefined
	printjobid	• String

TM-T20III

プリンター仕様

項目		58mm 仕様	80mm 仕様
初期改行量		30 ドット	
ページモード初期領域	通常モード	420 x 1662 ドット	576 x 1662 ドット
	42 桁モード	378 x 1662 ドット	546 x 1662 ドット
ページモード最大領域	通常モード	464 x 3324 ドット	639 x 3324 ドット
	42 桁モード	639 x 3324 ドット	
Font A のベースライン		文字の上端から 21 ドット目	
Font B のベースライン		文字の上端から 16 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_PRINTER
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B

API	パラメーター	指定可能な設定値
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED • SYMBOL_AZTECCODE_FULLRANGE • SYMBOL_AZTECCODE_COMPACT • SYMBOL_DATAMATRIX_SQUARE • SYMBOL_DATAMATRIX_RECTANGLE_8 • SYMBOL_DATAMATRIX_RECTANGLE_12 • SYMBOL_DATAMATRIX_RECTANGLE_16
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • 5 ～ 95 の整数 • LEVEL_DEFAULT
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined

API	パラメーター	指定可能な設定値
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 1 ～ 255 • undefined
print メソッド	canvas	• canvas
	cut	• true • false • undefined
	mode	• MODE_MONO • undefined
	printjobid	• String

TM-T88VI

プリンター仕様

項目		58mm 仕様	80mm 仕様
初期改行量		30 ドット	
ページモード初期領域		360 x 831 ドット	512 x 831 ドット
ページモード最大領域		360 x 1662 ドット	512 x 1662 ドット
Font A のベースライン		文字の上端から 21 ドット目	
Font B のベースライン	ANK	文字の上端から 16 ドット目	
	漢字	文字の上端から 15 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_DISPLAY • DEVICE_TYPE_PRINTER • DEVICE_TYPE_SCANNER
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addText メソッド	data	• data
	lang	• "en"（初期値） • "ja" • "undefined"
	x	• 1 ～ 44 • "undefined"
	y	• 1 ～ 19 • "undefined"
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B

API	パラメーター	指定可能な設定値
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • MODE_GRAY16 • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED • SYMBOL_AZTECCODE_FULLRANGE • SYMBOL_AZTECCODE_COMPACT • SYMBOL_DATAMATRIX_SQUARE • SYMBOL_DATAMATRIX_RECTANGLE_8 • SYMBOL_DATAMATRIX_RECTANGLE_12 • SYMBOL_DATAMATRIX_RECTANGLE_16
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • 5 ～ 95 の整数 • LEVEL_DEFAULT

API	パラメーター	指定可能な設定値
addPageLine メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addPageRectangle メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 1 ～ 255 • undefined

TM-T88VII

プリンター仕様

項目		58mm 仕様	80mm 仕様
初期改行量		30 ドット	
ページモード初期領域	標準モード	360 x 1662 ドット	512 x 1662 ドット
	48 桁モード	360 x 1662 ドット	480 x 1662 ドット
ページモード最大領域	標準モード	360 x 4800 ドット	512 x 4800 ドット
	48 桁モード	360 x 4800 ドット	480 x 4800 ドット
Font A のベースライン		文字の上端から 21 ドット目	
Font B のベースライン		文字の上端から 16 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_DISPLAY • DEVICE_TYPE_PRINTER • DEVICE_TYPE_SCANNER
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp • mul
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B

API	パラメーター	指定可能な設定値
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • MODE_GRAY16 • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_CODE128_AUTO • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED • SYMBOL_AZTECCODE_FULLRANGE • SYMBOL_AZTECCODE_COMPACT • SYMBOL_DATAMATRIX_SQUARE • SYMBOL_DATAMATRIX_RECTANGLE_8 • SYMBOL_DATAMATRIX_RECTANGLE_12 • SYMBOL_DATAMATRIX_RECTANGLE_16
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • 5 ～ 95 の整数 • LEVEL_DEFAULT

API	パラメーター	指定可能な設定値
addPageLine メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addPageRectangle メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 1 ～ 255 • undefined

TM-L100

プリンター仕様

項目	40mm 仕様	58mm 仕様	80mm 仕様（42 桁）	80mm 仕様（48 桁）
初期改行量	30 ドット			
ページモード初期領域	280 x 4800 ドット	400 x 4800 ドット	552 x 4800 ドット	576 x 4800 ドット
ページモード最大領域	280 x 4800 ドット	420 x 4800 ドット	576 x 4800 ドット	
Font A のベースライン	文字の上端から 21 ドット目			
Font B のベースライン	文字の上端から 21 ドット目			

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_DISPLAY • DEVICE_TYPE_PRINTER • DEVICE_TYPE_SCANNER
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp • mul
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B • FONT_SPECIAL_A • FONT_SPECIAL_B

API	パラメーター	指定可能な設定値
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addFeedPosition メソッド	pos	• FEED_CUTTING
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • MODE_GRAY16 • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_CODE128_AUTO • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED • SYMBOL_AZTECCODE_FULLRANGE • SYMBOL_AZTECCODE_COMPACT • SYMBOL_DATAMATRIX_SQUARE • SYMBOL_DATAMATRIX_RECTANGLE_8 • SYMBOL_DATAMATRIX_RECTANGLE_12 • SYMBOL_DATAMATRIX_RECTANGLE_16
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • 5 ～ 95 の整数 • LEVEL_DEFAULT

API	パラメーター	指定可能な設定値
addPageLine メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addPageRectangle メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 1 ～ 255 • undefined

参考情報

用紙除去待ち機能有効時の制限事項

プリンターが用紙除去待ち中（ERROR_WAIT_EJECT）になると、以下の制御が停止することがあります。

- ❏ 内蔵ブザーの鳴動制御
- ❏ 外付けオプションブザーの鳴動制御
- ❏ カスタマーディスプレイの制御
- ❏ バーコードスキャナーの制御

用紙を取り除き、用紙除去待ち中でなくなると復帰します。

内蔵ブザーと外付けオプションブザーは、プリンター本体のソフトウェア設定で用紙除去待ち時の鳴動設定をすることで、用紙除去待ち中に鳴動させることができます。

詳細は、詳細取扱説明書を参照してください。

TM-T70II

プリンター仕様

項目	58mm 仕様	80mm 仕様
初期改行量	30 ドット	
ページモード初期領域	416 x 1662 ドット	576 x 1662 ドット
ページモード最大領域	416 x 1662 ドット	576 x 1662 ドット
Font A のベースライン	文字の上端から 21 ドット目	
Font B のベースライン	文字の上端から 15 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_PRINTER
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined

API	パラメーター	指定可能な設定値
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • undefined
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • LEVEL_DEFAULT
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 1 ～ 255 • undefined

API	パラメーター	指定可能な設定値
print メソッド	canvas	• canvas
	cut	• true • false • undefined
	mode	• MODE_MONO • undefined
	printjobid	• String

TM-T88V

プリンター仕様

項目		58mm 仕様	80mm 仕様
初期改行量		30 ドット	
ページモード初期領域		360 x 831 ドット	512 x 831 ドット
ページモード最大領域		360 x 1662 ドット	512 x 1662 ドット
Font A のベースライン		文字の上端から 21 ドット目	
Font B のベースライン	ANK	文字の上端から 16 ドット目	
	漢字	文字の上端から 15 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_PRINTER
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined

API	パラメーター	指定可能な設定値
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • MODE_GRAY16 • undefined
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • LEVEL_DEFAULT
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 1 ～ 255 • undefined

TM-T90II

プリンター仕様

項目		58mm 仕様	80mm 仕様
初期改行量		30 ドット	
ページモード初期領域	モノクロ印字	420 x 738 ドット	576 x 738 ドット
	2 色印字	420 x 369 ドット	576 x 369 ドット
ページモード最大領域	モノクロ印字	420 x 1476 ドット	576 x 1476 ドット
	2 色印字	420 x 738 ドット	576 x 738 ドット
Font A のベースライン		文字の上端から 21 ドット目	
Font B のベースライン		文字の上端から 21 ドット目	
Font C のベースライン		文字の上端から 15 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_PRINTER
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B • FONT_C

API	パラメーター	指定可能な設定値
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • COLOR_2 • undefined
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • MODE_GRAY16 • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B • FONT_C
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • LEVEL_DEFAULT
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined

TM-T90KP (UB-R04)

プリンター仕様

項目	仕様
初期改行量	30 ドット
ページモード初期領域	576 x 738 ドット
ページモード最大領域	576 x 1476 ドット
Font A のベースライン	文字の上端から 21 ドット目
Font B のベースライン	文字の上端から 21 ドット目
Font C のベースライン	文字の上端から 15 ドット目

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_PRINTER
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B • FONT_C

API	パラメーター	指定可能な設定値
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B • FONT_C
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 1 ～ 255 • undefined
print メソッド	canvas	• canvas
	cut	• true • false • undefined
	mode	• MODE_MONO • undefined
	printjobid	• String

TM-T90KP (UB-E04)

プリンター仕様

項目	仕様
初期改行量	30 ドット
ページモード初期領域	576 x 738 ドット
ページモード最大領域	576 x 1476 ドット
Font A のベースライン	文字の上端から 21 ドット目
Font B のベースライン	文字の上端から 21 ドット目
Font C のベースライン	文字の上端から 15 ドット目

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_PRINTER
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addTextLang メソッド	lang	• ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B • FONT_C
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined

API	パラメーター	指定可能な設定値
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • undefined
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B • FONT_C
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 1 ～ 255 • undefined
print メソッド	canvas	• canvas
	cut	• true • false • undefined
	mode	• MODE_MONO • undefined
	printjobid	• String

TM-P20

プリンター仕様

項目	仕様
初期改行量	30 ドット
ページモード初期領域	384 x 2400 ドット
ページモード最大領域	384 x 2400 ドット
Font A のベースライン	文字の上端から 21 ドット目
Font B のベースライン	文字の上端から 21 ドット目
Font C のベースライン	文字の上端から 16 ドット目
Font D のベースライン	文字の上端から 21 ドット目
Font E のベースライン	文字の上端から 15 ドット目

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_PRINTER
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B • FONT_C • FONT_D • FONT_E

API	パラメーター	指定可能な設定値
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addFeedPosition メソッド	pos	• FEED_CUTTING • FEED_CURRENT_TOF • FEED_NEXT_TOF
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B • FONT_C • FONT_D • FONT_E
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED • SYMBOL_AZTECCODE_FULLRANGE • SYMBOL_AZTECCODE_COMPACT • SYMBOL_DATAMATRIX_SQUARE • SYMBOL_DATAMATRIX_RECTANGLE_8 • SYMBOL_DATAMATRIX_RECTANGLE_12 • SYMBOL_DATAMATRIX_RECTANGLE_16
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • 5 ～ 95 の整数 • LEVEL_DEFAULT

API	パラメーター	指定可能な設定値
addPageLine メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addPageRectangle メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined
addSound メソッド	pattern	• PATTERN_NONE • PATERN_1 • PATERN_2 • PATERN_3 • PATERN_4 • PATERN_5 • PATERN_6 • PATERN_7 • PATERN_8 • PATERN_9 • PATERN_10 • undefined
	repeat	• 1 ～ 255 • undefined
	cycle	• 1000 ～ 25500

API	パラメーター	指定可能な設定値
addLayout メソッド	type	• LAYOUT_RECEIPT • LAYOUT_RECEIPT_BM
	width	• 290 ～ 800
	height	• レシート紙 • レシート紙（ブラックマーク付）
	margin_top	• レシート紙 • レシート紙（ブラックマーク付）
	margin_bottom	• レシート紙 • レシート紙（ブラックマーク付）
	offset_cut	• レシート紙 • レシート紙（ブラックマーク付）
	offset_label	• レシート紙 • レシート紙（ブラックマーク付）
print メソッド	canvas	• canvas
	cut	• true • false • undefined
	mode	• MODE_MONO • undefined
	printjobid	• String

TM-P60II (Peeler)

プリンター仕様

項目	レシート仕様	ダイカットラベル仕様
初期改行量	30 ドット	
ページモード初期領域	432 x 1624 ドット	400 x 1624 ドット
ページモード最大領域	432 x 1624 ドット	400 x 1624 ドット
Font A のベースライン	文字の上端から 21 ドット目	
Font B のベースライン	文字の上端から 16 ドット目	
Font C のベースライン	文字の上端から 15 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_PRINTER
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B • FONT_C

API	パラメーター	指定可能な設定値
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B • FONT_C
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED • SYMBOL_AZTECCODE_FULLRANGE • SYMBOL_AZTECCODE_COMPACT • SYMBOL_DATAMATRIX_SQUARE • SYMBOL_DATAMATRIX_RECTANGLE_8 • SYMBOL_DATAMATRIX_RECTANGLE_12 • SYMBOL_DATAMATRIX_RECTANGLE_16
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • 5 ～ 95 の整数 • LEVEL_DEFAULT

API	パラメーター	指定可能な設定値
addPageLine メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addPageRectangle メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined
addSound メソッド	pattern	• PATTERN_NONE • PATERN_1 • PATERN_2 • PATERN_3 • PATERN_4 • PATERN_5 • PATERN_6 • PATERN_7 • PATERN_8 • PATERN_9 • PATERN_10 • undefined
	repeat	• 1 ～ 255 • undefined
	cycle	• 1000 ～ 25500

API	パラメーター	指定可能な設定値
print メソッド	canvas	• canvas
	cut	• true • false • undefined
	mode	• MODE_MONO • undefined
	printjobid	• String

TM-P80

プリンター仕様

項目	仕様
初期改行量	30 ドット
ページモード初期領域	576 x 1662 ドット
ページモード最大領域	576 x 1662 ドット
Font A のベースライン	文字の上端から 21 ドット目
Font B のベースライン	文字の上端から 21 ドット目
Font C のベースライン	文字の上端から 15 ドット目

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_PRINTER
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B • FONT_C

API	パラメーター	指定可能な設定値
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addFeedPosition メソッド	pos	• FEED_CUTTING • FEED_CURRENT_TOF • FEED_NEXT_TOF
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B • FONT_C
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED • SYMBOL_AZTECCODE_FULLRANGE • SYMBOL_AZTECCODE_COMPACT • SYMBOL_DATAMATRIX_SQUARE • SYMBOL_DATAMATRIX_RECTANGLE_8 • SYMBOL_DATAMATRIX_RECTANGLE_12 • SYMBOL_DATAMATRIX_RECTANGLE_16
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • 5 ～ 95 の整数 • LEVEL_DEFAULT

API	パラメーター	指定可能な設定値
addPageLine メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addPageRectangle メソッド	x1	• 0 ～ 65535 の整数値
	y1	• 0 ～ 65535 の整数値
	x2	• 0 ～ 65535 の整数値
	y2	• 0 ～ 65535 の整数値
	style	• LINE_THIN • LINE_MEDIUM • LINE_THICK • undefined
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined
addSound メソッド	pattern	• PATTERN_NONE • PATERN_1 • PATERN_2 • PATERN_3 • PATERN_4 • PATERN_5 • PATERN_6 • PATERN_7 • PATERN_8 • PATERN_9 • PATERN_10 • undefined
	repeat	• 1 ～ 255 • undefined
	cycle	• 1000 ～ 25500

API	パラメーター	指定可能な設定値
addLayout メソッド	type	• LAYOUT_RECEIPT • LAYOUT_RECEIPT_BM
	width	• 290 ～ 800
	height	• レシート紙 • レシート紙（ブラックマーク付）
	margin_top	• レシート紙 • レシート紙（ブラックマーク付）
	margin_bottom	• レシート紙 • レシート紙（ブラックマーク付）
	offset_cut	• レシート紙 • レシート紙（ブラックマーク付）
	offset_label	• レシート紙 • レシート紙（ブラックマーク付）
print メソッド	canvas	• canvas
	cut	• true • false • undefined
	mode	• MODE_MONO • undefined
	printjobid	• String

TM-T70-i

プリンター仕様

項目	58mm 仕様	80mm 仕様
初期改行量	30 ドット	
ページモード初期領域	416 x 1662 ドット	576 x 1662 ドット
ページモード最大領域	416 x 1662 ドット	576 x 1662 ドット
Font A のベースライン	文字の上端から 21 ドット目	
Font B のベースライン	文字の上端から 15 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_DISPLAY • DEVICE_TYPE_KEYBOARD • DEVICE_TYPE_PRINTER • DEVICE_TYPE_SCANNER • DEVICE_TYPE_SIMPLE_SERIAL
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addText メソッド	data	• data
	lang	• "en"（初期値） • "ja" • "undefined"
	x	• 1 ～ 44 • "undefined"
	y	• 1 ～ 19 • "undefined"
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B

API	パラメーター	指定可能な設定値
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • undefined
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2
	level	• LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • LEVEL_DEFAULT
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined
print メソッド	canvas	• canvas
	cut	• true • false • undefined
	mode	• MODE_MONO • undefined
	printjobid	• String

TM-T88V-i

プリンター仕様

項目		58mm 仕様	80mm 仕様
初期改行量		30 ドット	
ページモード初期領域		360 x 831 ドット	512 x 831 ドット
ページモード最大領域		360 x 1662 ドット	512 x 1662 ドット
Font A のベースライン		文字の上端から 21 ドット目	
Font B のベースライン	ANK	文字の上端から 16 ドット目	
	漢字	文字の上端から 15 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
createDevice メソッド	deviceId	• deviceId
	deviceType	• DEVICE_TYPE_DISPLAY • DEVICE_TYPE_KEYBOARD • DEVICE_TYPE_PRINTER • DEVICE_TYPE_SCANNER • DEVICE_TYPE_SIMPLE_SERIAL
	options	• crypto (Boolean) • buffer (Boolean)
	crypto	• crypto (Boolean)
	callback	-
addText メソッド	data	• data
	lang	• "en"（初期値） • "ja" • "undefined"
	x	• 1 ～ 44 • "undefined"
	y	• 1 ～ 19 • "undefined"
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp

API	パラメーター	指定可能な設定値
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • MODE_GRAY16 • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • LEVEL_DEFAULT
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined

API	パラメーター	指定可能な設定値
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 0 • 1 ～ 255 • undefined

TM-T70II-DT

プリンター仕様

項目	58mm 仕様	80mm 仕様
初期改行量	30 ドット	
ページモード初期領域	416 x 1662 ドット	576 x 1662 ドット
ページモード最大領域	416 x 1662 ドット	576 x 1662 ドット
Font A のベースライン	文字の上端から 21 ドット目	
Font B のベースライン	文字の上端から 15 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
addText メソッド	data	• data
	lang	• "en"（初期値） • "ja" • "undefined"
	x	• 1 ～ 44 • "undefined"
	y	• 1 ～ 19 • "undefined"
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B

API	パラメーター	指定可能な設定値
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • MODE_GRAY16 • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • LEVEL_DEFAULT
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined

API	パラメーター	指定可能な設定値
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 0 • 1 ～ 255 • undefined

TM-T70II-DT2

プリンター仕様

項目	58mm 仕様	80mm 仕様
初期改行量	30 ドット	
ページモード初期領域	416 x 1662 ドット	576 x 1662 ドット
ページモード最大領域	416 x 1662 ドット	576 x 1662 ドット
Font A のベースライン	文字の上端から 21 ドット目	
Font B のベースライン	文字の上端から 15 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
addText メソッド	data	• data
	lang	• "en"（初期値） • "ja" • "undefined"
	x	• 1 ～ 44 • "undefined"
	y	• 1 ～ 19 • "undefined"
	r	• 0 ～ 255 の整数値 • "undefined"
	g	• 0 ～ 255 の整数値 • "undefined"
	b	• 0 ～ 255 の整数値 • "undefined"
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B

API	パラメーター	指定可能な設定値
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • MODE_GRAY16 • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • LEVEL_DEFAULT
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined

API	パラメーター	指定可能な設定値
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 0 • 1 ～ 255 • undefined

TM-T88V-DT

プリンター仕様

項目		58mm 仕様	80mm 仕様
初期改行量		30 ドット	
ページモード初期領域		360 x 831 ドット	512 x 831 ドット
ページモード最大領域		360 x 1662 ドット	512 x 1662 ドット
Font A のベースライン		文字の上端から 21 ドット目	
Font B のベースライン	ANK	文字の上端から 16 ドット目	
	漢字	文字の上端から 15 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
addText メソッド	data	• data
	lang	• "en"（初期値） • "ja" • "undefined"
	x	• 1 ～ 44 • "undefined"
	y	• 1 ～ 19 • "undefined"
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B

API	パラメーター	指定可能な設定値
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • MODE_GRAY16 • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • LEVEL_DEFAULT
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined

API	パラメーター	指定可能な設定値
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 0 • 1 ～ 255 • undefined

TM-T88VI-DT2

プリンター仕様

項目		58mm 仕様	80mm 仕様
初期改行量		30 ドット	
ページモード初期領域		360 x 831 ドット	512 x 831 ドット
ページモード最大領域		360 x 1662 ドット	512 x 1662 ドット
Font A のベースライン		文字の上端から 21 ドット目	
Font B のベースライン	ANK	文字の上端から 16 ドット目	
	漢字	文字の上端から 15 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
addText メソッド	data	• data
	lang	• "en"（初期値） • "ja" • "undefined"
	x	• 1 ～ 44 • "undefined"
	y	• 1 ～ 19 • "undefined"
	r	• 0 ～ 255 の整数値 • "undefined"
	g	• 0 ～ 255 の整数値 • "undefined"
	b	• 0 ～ 255 の整数値 • "undefined"
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B

API	パラメーター	指定可能な設定値
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • MODE_GRAY16 • undefined

API	パラメーター	指定可能な設定値
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128 • BARCODE_GS1_128 • BARCODE_GS1_DATABAR_OMNIDIRECTIONAL • BARCODE_GS1_DATABAR_TRUNCATED • BARCODE_GS1_DATABAR_LIMITED • BARCODE_GS1_DATABAR_EXPANDED
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6 • SYMBOL_GS1_DATABAR_STACKED • SYMBOL_GS1_DATABAR_STACKED_OMNIDIRECTIONAL • SYMBOL_GS1_DATABAR_EXPANDED_STACKED • SYMBOL_AZTECCODE_FULLRANGE • SYMBOL_AZTECCODE_COMPACT • SYMBOL_DATAMATRIX_SQUARE • SYMBOL_DATAMATRIX_RECTANGLE_8 • SYMBOL_DATAMATRIX_RECTANGLE_12 • SYMBOL_DATAMATRIX_RECTANGLE_16
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • 5 ～ 95 の整数 • LEVEL_DEFAULT
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined

API	パラメーター	指定可能な設定値
addSound メソッド	pattern	• PATTERN_NONE • PATTERN_A • PATTERN_B • PATTERN_C • PATTERN_D • PATTERN_E • PATTERN_ERROR • PATTERN_PAPER_END • undefined
	repeat	• 1 ～ 255 • undefined

TM-T70

プリンター仕様

項目	58mm 仕様	80mm 仕様
初期改行量	30 ドット	
ページモード初期領域	416 x 1662 ドット	576 x 1662 ドット
ページモード最大領域	416 x 1662 ドット	576 x 1662 ドット
Font A のベースライン	文字の上端から 21 ドット目	
Font B のベースライン	文字の上端から 15 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • undefined

API	パラメーター	指定可能な設定値
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • undefined
	mode	• MODE_MONO • undefined
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値
addSymbol メソッド	data	• String
	type	• SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2
	level	• LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • LEVEL_DEFAULT
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined

API	パラメーター	指定可能な設定値
print メソッド	canvas	• canvas
	cut	• true • false • undefined
	mode	• MODE_MONO • undefined
	printjobid	• String

TM-T88IV

プリンター仕様

項目		58mm 仕様	80mm 仕様
初期改行量		30 ドット	
ページモード初期領域	モノクロ印字	360 x 831 ドット	512 x 831 ドット
	2 色印字	360 x 415 ドット	512 x 415 ドット
ページモード最大領域	モノクロ印字	360 x 1662 ドット	512 x 1662 ドット
	2 色印字	360 x 831 ドット	512 x 831 ドット
Font A のベースライン		文字の上端から 21 ドット目	
Font B のベースライン		文字の上端から 16 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • COLOR_2 • undefined

API	パラメーター	指定可能な設定値
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • COLOR_2 • undefined
	mode	• MODE_MONO • MODE_GRAY16 • undefined
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値
addSymbol メソッド	data	• String
	type	• SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2
	level	• LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • LEVEL_DEFAULT

API	パラメーター	指定可能な設定値
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined

TM-T90

プリンター仕様

項目		58mm 仕様	80mm 仕様
初期改行量		30 ドット	
ページモード初期領域	モノクロ印字	420 x 738 ドット	576 x 738 ドット
	2 色印字	420 x 369 ドット	576 x 369 ドット
ページモード最大領域	モノクロ印字	420 x 1476 ドット	576 x 1476 ドット
	2 色印字	420 x 738 ドット	576 x 738 ドット
Font A のベースライン		文字の上端から 21 ドット目	
Font B のベースライン		文字の上端から 21 ドット目	
Font C のベースライン		文字の上端から 15 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B • FONT_C
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • COLOR_2 • undefined

API	パラメーター	指定可能な設定値
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • COLOR_2 • undefined
	mode	• MODE_MONO • MODE_GRAY16 • undefined
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B • FONT_C
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_DEFAULT
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined

TM-L90

プリンター仕様

項目		レシート仕様	ダイカットラベル仕様
初期改行量		30 ドット	
ページモード初期領域	モノクロ印字	576 x 738 ドット	560 x 738 ドット
	2 色印字	576 x 369 ドット	560 x 369 ドット
ページモード最大領域	モノクロ印字	576 x 1476 ドット	560 x 1476 ドット
	2 色印字	576 x 738 ドット	560 x 738 ドット
Font A のベースライン		文字の上端から 21 ドット目	
Font B のベースライン		文字の上端から 21 ドット目	
Font C のベースライン		文字の上端から 15 ドット目	

指定できるパラメーター設定値に制限のある API

API	パラメーター	指定可能な設定値
addTextLang メソッド	lang	• en（初期値） • de • fr • it • es • ja • ja-jp
addTextFont メソッド	font	• FONT_A （初期値） • FONT_B • FONT_C
addTextStyle メソッド	reverse	• true • false（初期値） • undefined
	ul	• true • false（初期値） • undefined
	em	• true • false（初期値） • undefined
	color	• COLOR_1 （初期値） • COLOR_2 • undefined

API	パラメーター	指定可能な設定値
addImage メソッド	context	• context
	x	• 0 ～ 65535 の整数値
	y	• 0 ～ 65535 の整数値
	width	• 0 ～ 65535 の整数値
	height	• 0 ～ 65535 の整数値
	color	• COLOR_1（初期値） • COLOR_2 • undefined
	mode	• MODE_MONO • MODE_GRAY16 • undefined
addBarcode メソッド	data	• String
	type	• BARCODE_UPC_A • BARCODE_UPC_E • BARCODE_EAN13 • BARCODE_JAN13 • BARCODE_EAN8 • BARCODE_JAN8 • BARCODE_CODE39 • BARCODE_ITF • BARCODE_CODABAR • BARCODE_CODE93 • BARCODE_CODE128
	hri	• HRI_NONE（初期値） • HRI_ABOVE • HRI_BELOW • HRI_BOTH
	font	• FONT_A（初期値） • FONT_B • FONT_C
	width	• 2 ～ 6 の整数値
	height	• 1 ～ 255 の整数値

API	パラメーター	指定可能な設定値
addSymbol メソッド	data	• String
	type	• SYMBOL_PDF417_STANDARD • SYMBOL_PDF417_TRUNCATED • SYMBOL_QRCODE_MODEL_1 • SYMBOL_QRCODE_MODEL_2 • SYMBOL_MAXICODE_MODE_2 • SYMBOL_MAXICODE_MODE_3 • SYMBOL_MAXICODE_MODE_4 • SYMBOL_MAXICODE_MODE_5 • SYMBOL_MAXICODE_MODE_6
	level	• LEVEL_0 • LEVEL_1 • LEVEL_2 • LEVEL_3 • LEVEL_4 • LEVEL_5 • LEVEL_6 • LEVEL_7 • LEVEL_8 • LEVEL_L • LEVEL_M • LEVEL_Q • LEVEL_H • LEVEL_DEFAULT
addCut メソッド	type	• CUT_NO_FEED • CUT_FEED • undefined

ファームウェアバージョンによる使用制限

プリンターに搭載されているファームウェアやソフトウェアのバージョンによって、使用できる API やパラメーターなどに制限があります。

ファームウェアやソフトウェアのバージョン確認、バージョンアップ方法は、各プリンターの詳細取扱説明書を参照してください。

TM-i シリーズ /TM-DT シリーズ

❏ TM-DT ソフトウェア Ver.2.2、TM-i ファームウェア Ver.4.0 以降対応

クラス	API	パラメーター	値
ePOSDevice オブジェクト	createDevice メソッド	options	-
Printer オブジェクト	addTextVPosition メソッド	-	-
	addLayout メソッド	-	-
	addRecovery メソッド	-	-
	addReset メソッド	-	-
	force プロパティー	-	-
	onbatterystatuschange イベント	-	-
	onbatterystatuschange イベント	-	-
	onbatteryok イベント	-	-
	onbatterylow イベント	lang	zh-hans
		lang	zh-hant
	addSymbol メソッド	type	SYMBOL_AZTECCODE_FULLRANGE
			SYMBOL_AZTECCODE_COMPACT
			SYMBOL_DATAMATRIX_SQUARE
			SYMBOL_DATAMATRIX_RECTANGLE_8
			SYMBOL_DATAMATRIX_RECTANGLE_12
			SYMBOL_DATAMATRIX_RECTANGLE_16
		level	整数値（5 ～ 95）

クラス	API	パラメーター	値
Printer オブジェクト	addSound メソッド	pattern	PATERN_1
			PATERN_2
			PATERN_3
			PATERN_4
			PATERN_5
			PATERN_6
			PATERN_7
			PATERN_8
			PATERN_9
			PATERN_10
		cycle	-

❏ TM-DT ソフトウェア Ver.2.5、TM-i ファームウェア Ver.4.0 以降対応

クラス	API	パラメーター	値
Printer オブジェクト	recover メソッド	-	-
	reset メソッド	-	-

❏ TM-DT ソフトウェア Ver.2.5、TM-i ファームウェア Ver.4.1 以降対応

クラス	API	パラメーター	値
Printer オブジェクト	send メソッド	printjobid	-
	print メソッド	printjobid	-
	getPrintJobStatus メソッド	-	-
	onreceive イベント	printjobid	-
		code	EX_SPOOLER
			JobNotFound
			Printing
		status	ASB_SPOOLER_IS_STOPPED

❏ TM-DT ソフトウェア Ver.3.0、TM-i ファームウェア Ver.4.1 以降対応

クラス	API	パラメーター	値
Printer オブジェクト	drawerOpenLevel プロパティー	-	-
DeviceHubTerminal オブジェクト	restart メソッド	-	-
CommBox オブジェクト	getCommHistory メソッド	option	-
ePOSDeviceConfiguration オブジェクト	-	-	-

❏ TM-DT ソフトウェア Ver.4.0 以降対応

クラス	API	パラメーター	値
ePOSDevice オブジェクト	createDevice メソッド	deviceType	DEVICE_TYPE_CAT
			DEVICE_TYPE_POSKEYBOARD
			DEVICE_TYPE_OTHER_PERIPHERAL
OtherPeripheral オブジェクト	-	-	-
CashChanger オブジェクト（デバイス制御プログラム）	-	-	-
CashChanger オブジェクト（デバイス制御スクリプト）	dispenseChange メソッド	cash	-
	dispenseCash メソッド	data	-
	oncashcount イベント	data	jpy1000cassette
			jpy2000cassette
			jpy5000cassette
			jpy10000cassette
	ondeposit イベント *	-	-
CAT オブジェクト	-	-	-
POSKeyboard オブジェクト	-	-	-

* 自動つり銭機へ入金される度に入金金額を通知します。

❏ TM-DT ソフトウェア Ver.4.1 以降対応

クラス	API	パラメーター	値
CAT オブジェクト	sendCommand メソッド	-	-
CAT オブジェクト	oncommandreply イベント	-	-
CAT オブジェクト	onstatusupdate イベント	-	-
CAT オブジェクト	-	status	OPOS_CODE_XX

❏ TM-DT ソフトウェア Ver.5.10 以降対応

クラス	API	パラメーター	値
Display オブジェクト	addCreateScreen メソッド	-	-
	addCreateScreenCustom メソッド	-	-
	createTextArea メソッド	-	-
	destroyTextArea メソッド	-	-
	setCurrentTextArea メソッド	-	-
	clearTextArea メソッド	-	-

クラス	API	パラメーター	値
Display オブジェクト	addText メソッド	lang	"mul"
		r	-
		g	-
		b	-
	addBackgroundColor メソッド	-	-
	addStartSlideShow メソッド	-	-
	addStopSlideShow メソッド	-	-
	addDownloadImage メソッド	-	-
	addRegisterDownloadImage メソッド	-	-
	addNVImage メソッド	-	-
	addClearImage メソッド	-	-
	addSymbol メソッド	-	-
	addClearSymbol メソッド	-	-
CAT オブジェクト	authorizeSales メソッド authorizeVoid メソッド	service	"nfcpayment"
			"point"
			"pitapa"
			"fisc"
		amount	-
		tax	-
		additionalSecurityInformation	-
	authorizeCompletion メソッド	amount	-
		tax	-
		additionalSecurityInformation	-
	accessDailyLog メソッド	service	"nfcpayment"
			"point"
			"pitapa"
			"fisc"
		dailyLogType	-
	accessDailyLog メソッド	additionalSecurityInformation	-

クラス	API	パラメーター	値
CAT オブジェクト	sendCommand メソッド	additionalSecurityInformation	-
	checkConnection メソッド	-	-
	clearOutput メソッド	-	-
	onauthorizesales イベント onauthorizevoid イベント onauthorizecompletion イベント	data	transactionType
			additionalSecurityInformation
			paymentCondition • "bonus_4" • "bonus_5" • "installment_3" • "combination_3" • "combination_4"
	oncheckConnection イベント	-	-
	onclearOutput イベント	-	-
	ondirectio イベント	-	-

❏ TM-DT ソフトウェア Ver.5.30 以降対応

クラス	API	パラメーター	値
CAT オブジェクト	authorizeSales メソッド	service	"qr"
	authorizeVoid メソッド	service	"qr"
	authorizeRefund メソッド	service	"nfcpayment"
			"pitapa"
		amount	-
		tax	-
		additionalSecurityInformation	-
	authorizeCompletion メソッド	service	"unionpay"
	accessDailyLog メソッド	service	"qr"
	sendCommand メソッド	service	"nfcpayment"
			"pitapa"
			"qr"

クラス	API	パラメーター	値
CAT オブジェクト	onauthorizerefund イベント	data	transactionType
			additionalSecurityInformation
			paymentCondition • "bonus_4" • "bonus_5" • "installment_3" • "combination_3" • "combination_4"

TM-m30

❏ TM-m30 Firmware Ver.1.40 ESC/POS 以降で対応

クラス	API	パラメーター	値
Scanner オブジェクト	-	-	-

❏ TM-m30 Firmware Ver.1.44 ESC/POS 以降で対応

クラス	API	パラメーター	値
Printer オブジェクト	addCut メソッド	type	CUT_RESERVE

TM-m30II-H/TM-m30II-S/TM-m30II-SL

❏ Firmware Ver.3.06B ESC/POS 以降で対応

クラス	API	パラメーター	値
Display オブジェクト	addCreateScreen メソッド	-	-
	addCreateScreenCustom メソッド	-	-
	createTextArea メソッド	-	-
	destroyTextArea メソッド	-	-
	setCurrentTextArea メソッド	-	-

クラス	API	パラメーター	値
Display オブジェクト	clearTextArea メソッド	-	-
	addText メソッド	lang	"mul"
		r	-
		g	-
		b	-
	addBackgroundColor メソッド	-	-
	addStartSlideShow メソッド	-	-
	addStopSlideShow メソッド	-	-
	addDownloadImage メソッド	-	-
	addRegisterDownloadImage メソッド	-	-
	addNVImage メソッド	-	-
	addClearImage メソッド	-	-
	addSymbol メソッド	-	-
	addClearSymbol メソッド	-	-

TM-T88VI

❏ TM-T88VI Firmware Ver.40.50 ESC/POS 以降で対応

クラス	API	パラメーター	値
Scanner オブジェクト	-	-	-

ePOS-Print 設定

機種名	Printing Method	Character Code Tables
TM-m10	Thermal (203 dpi)	Page 0-5, 16-19, 20-21, 26,30-31, 11-15, 32-53
TM-m30	Thermal (203 dpi)	Page 0-5, 16-19, 20-21, 26,30-31, 11-15, 32-53
TM-T70II	Thermal (203 dpi)	Page 0-5, 16-19, 20-21, 26,30-31, 11-15, 32-53
TM-T88V	Thermal (180 dpi)	Page 0-5, 16-19, 20-21, 26,30-31, 11-15, 32-53
TM-T90II	Thermal (203 dpi)	Page 0-5, 16-19, 20-21, 26,30-31, 11-15, 32-53
TM-T90KP	Thermal (203 dpi)	Page 0-5, 16-19*
TM-L90	Thermal (203 dpi)	Page 0-5, 16-19

* UB-E04 搭載モデルでは、「Page 0-5, 16-19, 20-21, 26,30-31, 11-15, 32-53」が初期設定されています。

サンプルプログラム

Epson ePOS SDK for JavaScript が提供するサンプルプログラムは、Web アプリケーションソフト開発者向けの実装サンプルです。

機能

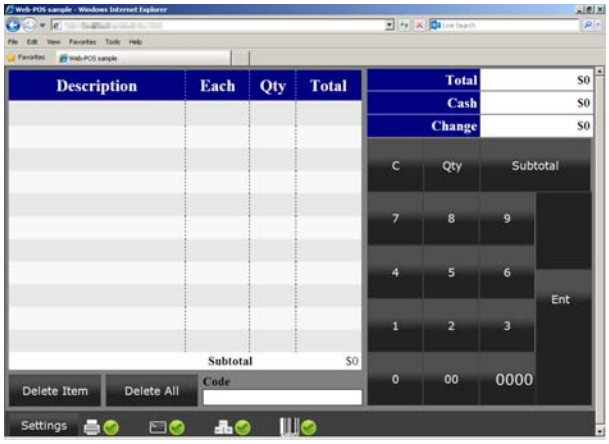
Epson ePOS SDK for JavaScript で提供しているサンプルプログラムには、以下の機能があります。

POS Terminal

POS システムのサンプルプログラムです。

タブレット端末に表示するコミュニケーションボックスの機能を使用することもできます。

> Internet Explorer で POS Terminal Sample を使用する場合は、Internet Explorer 9 以降を使用してください。



<POS Terminal Customer Display Sample>

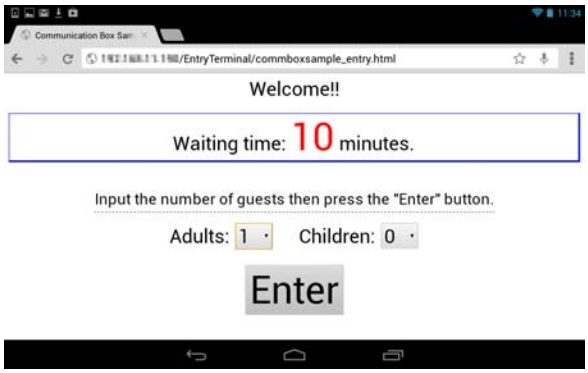
対応するシステム構成：

- POS ターミナルモデル

Entry Terminal

受付端末のサンプルプログラムです。
コミュニュニケーションボックスの双方向通信機能を使用して、タブレット等の Web ブラウザー間の通信ができます。

<Entry Sample>



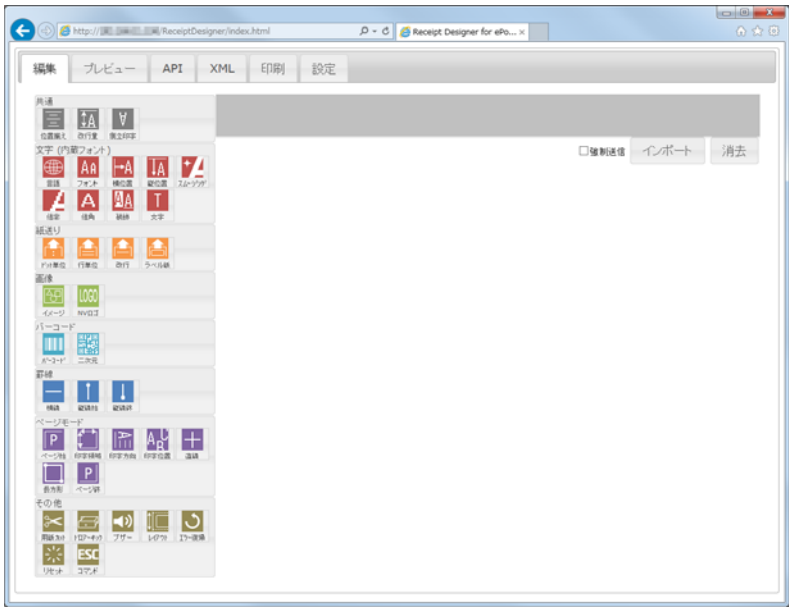
<Call Sample>

対応するシステム構成：

- POS ターミナルモデル

Receipt Designer

レシートプリンターの印刷データを簡単に生成し、印刷できます。



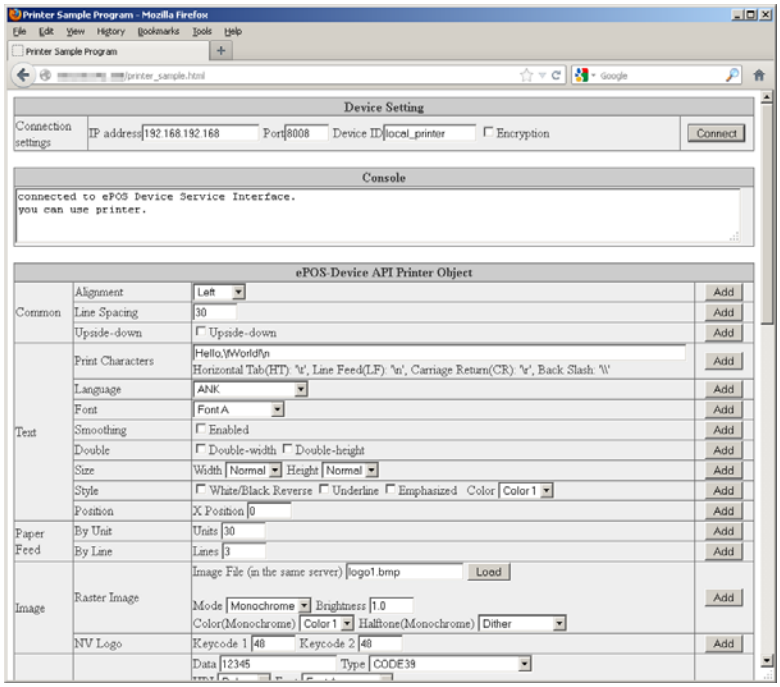
対応するシステム構成：

- TM プリンターモデル
- カスタマーディスプレイモデル
- TM プリンター +DM-D+ バーコードスキャナーモデル
- POS ターミナルモデル

Printer Sample

プリンターの印刷データを生成し、印刷できます。

レシートプリンターの場合は、Receipt Designer で生成することを推奨します。

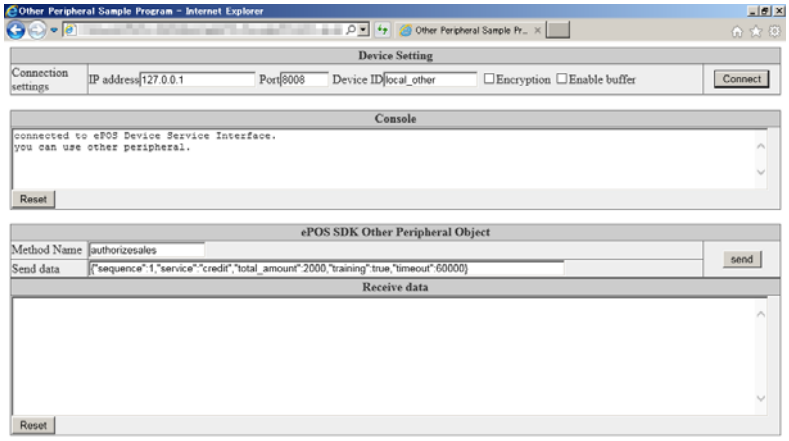


対応するシステム構成：

- TM プリンターモデル
- カスタマーディスプレイモデル
- TM プリンター +DM-D+ バーコードスキャナーモデル
- POS ターミナルモデル

OtherPeripheral Sample

デバイス制御プログラムのサンプルを使って、その他の周辺機器とのデータ送受信を確認できます。

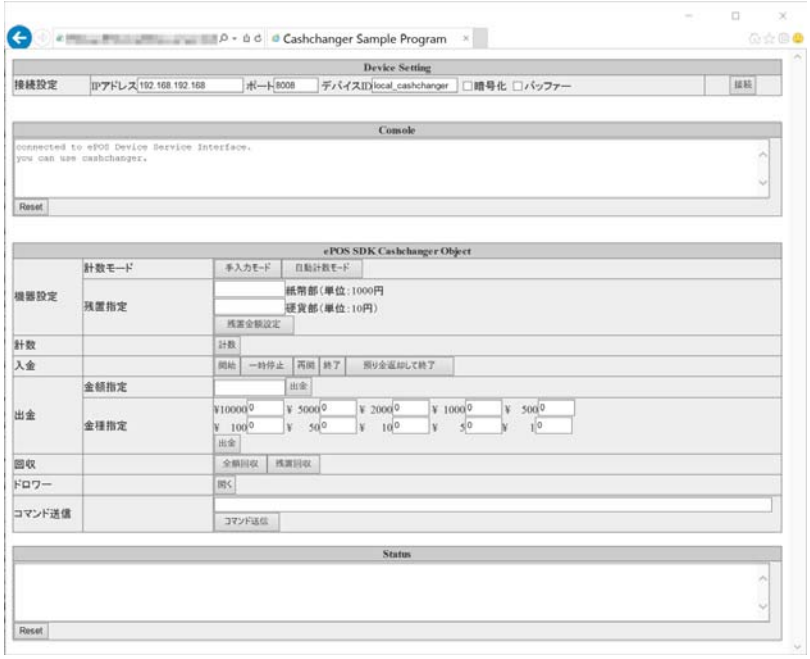


対応するシステム構成：

- POS ターミナルモデル

CashChanger Sample (Device Control Program)

デバイス制御プログラムから自動つり銭機を制御できます。

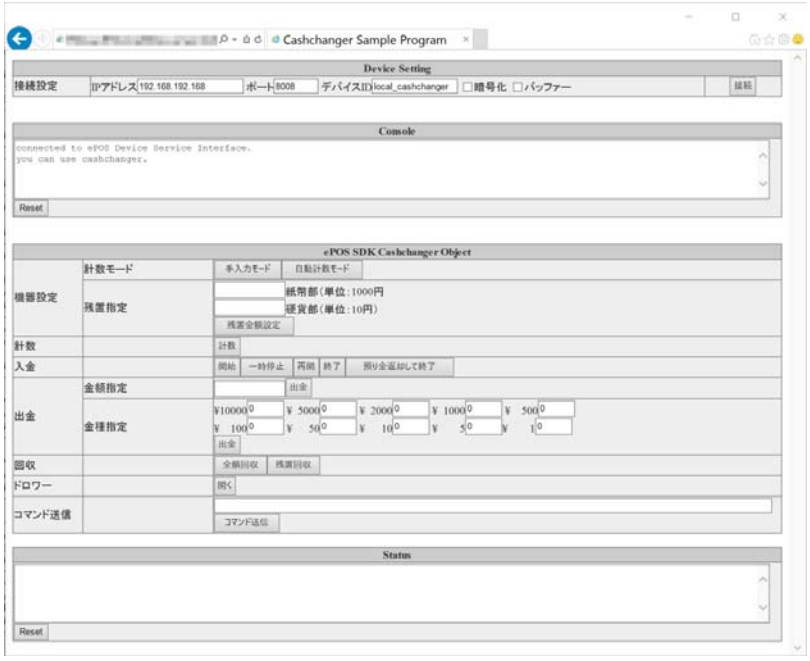


対応するシステム構成：

- POS ターミナルモデル

CashChanger Sample (Device Control Script)

デバイス制御スクリプトから自動つり銭機を制御できます。

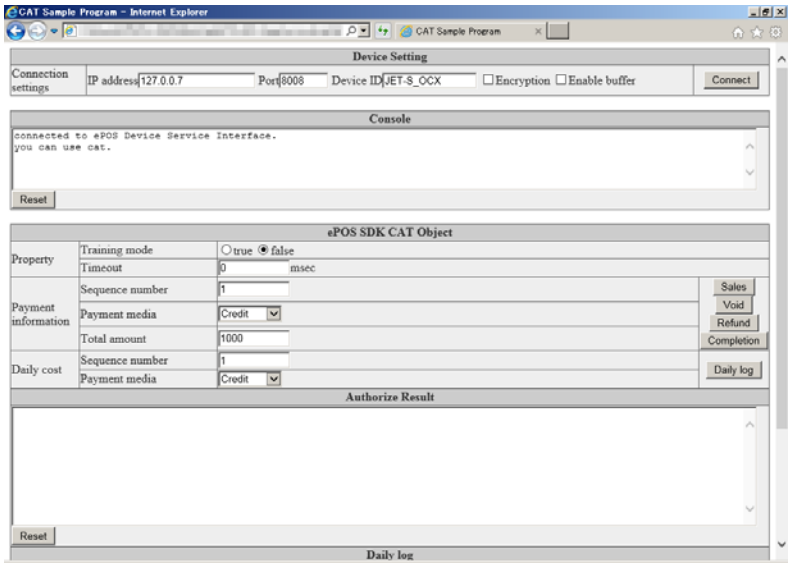


対応するシステム構成：

- POS ターミナルモデル

CAT Sample

信用照会端末を使って、決済処理と 1 日の集計を出すことができます。

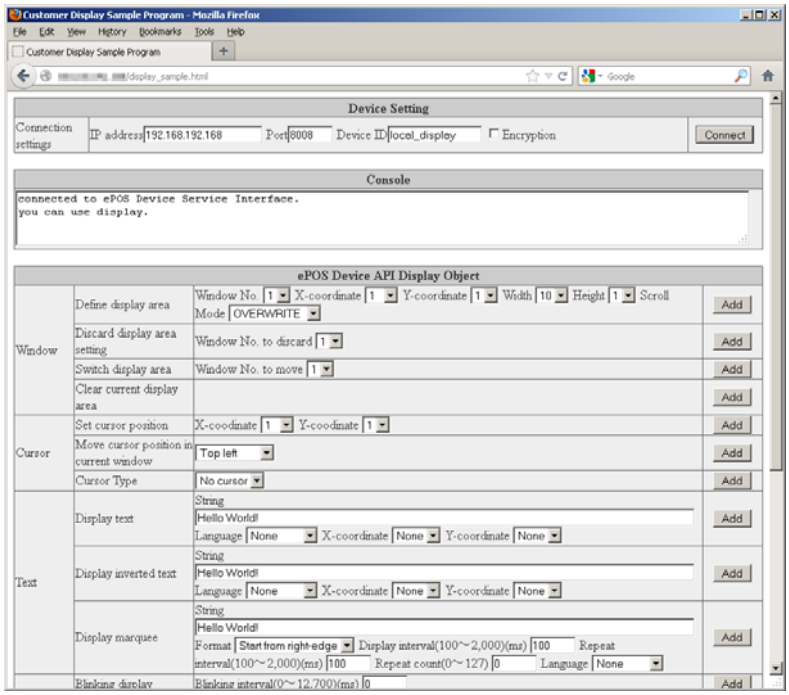


対応するシステム構成：

- POS ターミナルモデル

Customer Display Sample

カスタマーディスプレイの表示データを生成し、表示できます。

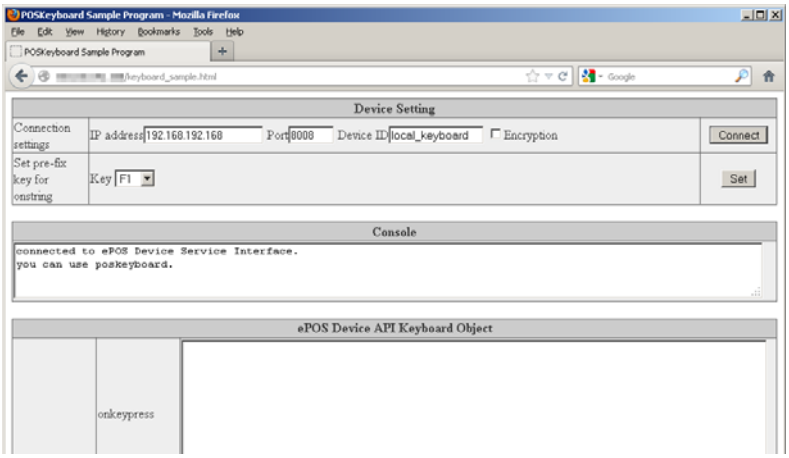


対応するシステム構成：

- カスタマーディスプレイモデル
- TM プリンター +DM-D+ バーコードスキャナーモデル
- POS ターミナルモデル

Keyboard Sample

キーボードから入力データを取得できます。

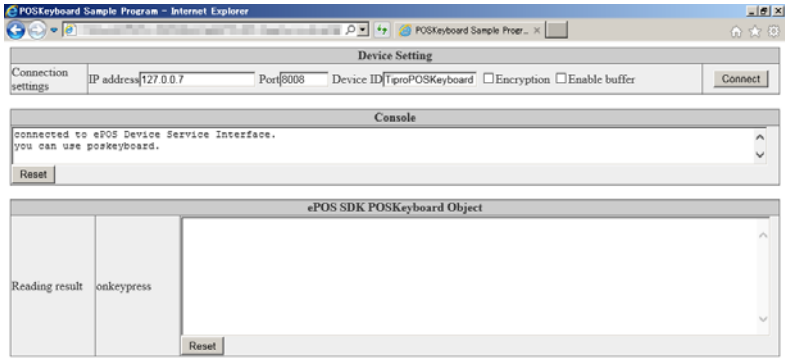


対応するシステム構成：

- POS ターミナルモデル

POSKeyboard Sample

POS キーボードから入力データを取得できます。

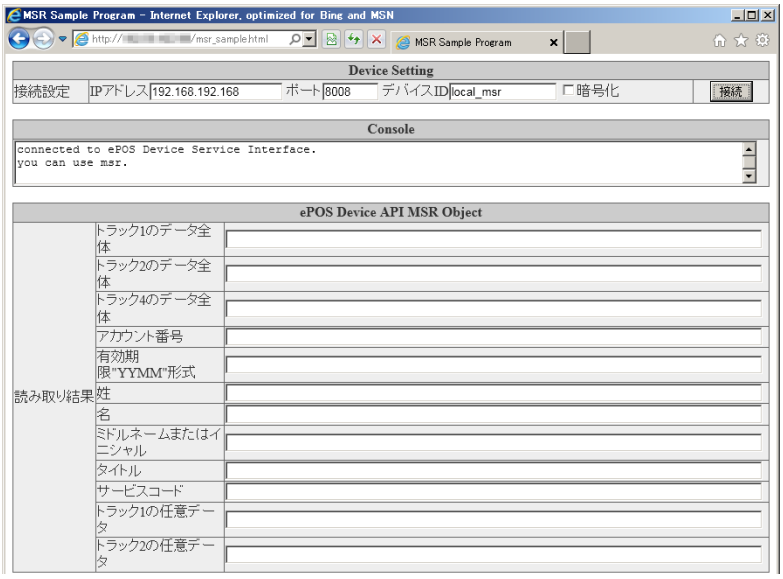


対応するシステム構成：

- POS ターミナルモデル

MSR Sample

MSR データを取得できます。

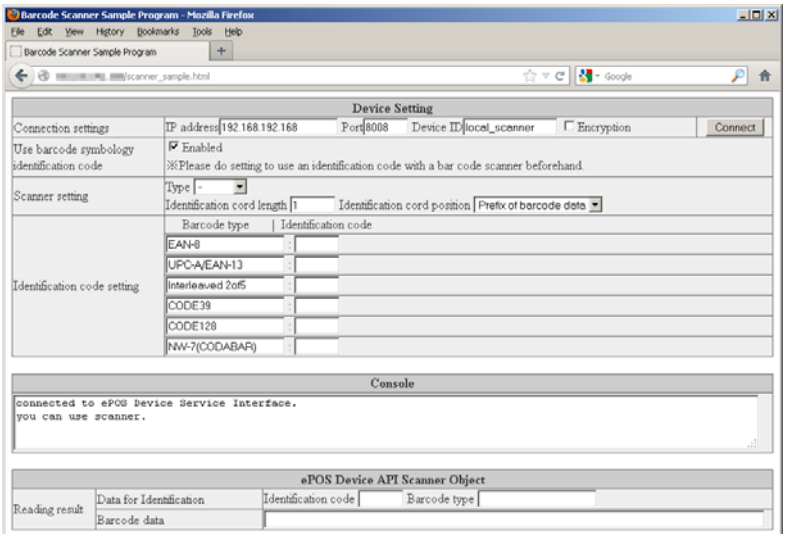


対応するシステム構成：

- POS ターミナルモデル

Barcode Scanner Sample

バーコードスキャナーで、バーコードデータを取得できます。



対応するシステム構成：

- TM プリンター +DM-D+ バーコードスキャナーモデル
- POS ターミナルモデル

使用環境

アプリケーションソフトの動作環境を参照してください。

サンプルプログラムの起動手順

Epson ePOS SDK for JavaScript の提供するサンプルプログラム起動手順を説明します。

1. プリンターの環境設定

2. サンプルプログラムの起動

プリンターの環境設定

プリンターによって環境の設定方法が異なります。

- TM-i シリーズ /TM-DT シリーズ
- 単機能モデル / モバイルモデル

TM-i シリーズ /TM-DT シリーズ

実行するサンプルプログラムによって、環境設定方法が異なります。

- Other Peripheral Sample 以外を実行する
- Other Peripheral Sample を実行する

Other Peripheral Sample 以外を実行する

EPSON TMNet WebConfig の操作詳細は、各プリンターの詳細取扱説明書を参照してください。

1 **使用するデバイスをプリンターに接続します。**

2 **プリンターにネットワークを設定します。**

3 **サンプルプログラム (ePOS_SDK_Sample_JavaScript.zip) をプリンターに登録します。**

EPSON TMNet WebConfig の「Web サービス設定」-「Web コンテンツ」-「更新設定」から手動更新機能を使用して登録します。

4 **デバイスをプリンターに設定します。**

EPSON TMNet WebConfig で、使用するデバイスごとに以下を設定してください。

- OPOS 仕様準拠の周辺機器
  EPSON TMNet WebConfig の「Web サービス設定」-「制御プログラム」-「デバイス登録」から設定します。

デバイス	設定項目	設定値
自動つり銭機	デバイス ID	論理デバイス名
	制御プログラム	OposCashChangerHandler.exe
信用照会端末	デバイス ID	論理デバイス名
	制御プログラム	OposCATHandler.exe
MSR	デバイス ID	論理デバイス名
	制御プログラム	OposMSRHandler.exe
POS キーボード	デバイス ID	論理デバイス名
	制御プログラム	OposPOSKeyboardHandler.exe
バーコードスキャナー	デバイス ID	論理デバイス名
	制御プログラム	OposScannerHandler.exe

❏ カスタマーディスプレイ、標準 HID デバイス、シリアル通信デバイス
EPSON TMNet WebConfig の「Web サービス設定」-「デバイス管理」-「デバイス登録」から設定します。

デバイス	設定項目	設定値
カスタマーディスプレイ	カスタマーディスプレイ	使用する
キーボード	デバイス ID	local_keyboard
	デバイス名	リストから該当するデバイスを選択します。
	制御スクリプト	Keyboard_Generic.js
MSR	デバイス ID	local_msr
	デバイス名	リストから該当するデバイスを選択します。
	制御スクリプト	MSR_V3TU_FK.js
バーコードスキャナー	デバイス ID	local_scanner
	デバイス名	リストから該当するデバイスを選択します。
	制御スクリプト	Scanner_Generic.js
自動つり銭機	デバイス ID	local_cashchanger
	デバイス名	リストから該当するデバイス、もしくは使用するポートを選択します。
	制御スクリプト	CashChanger_RT_200RAD_200.js

Other Peripheral Sample を実行する

EPSON TMNet WebConfig の操作詳細は、各プリンターの詳細取扱説明書を参照してください。

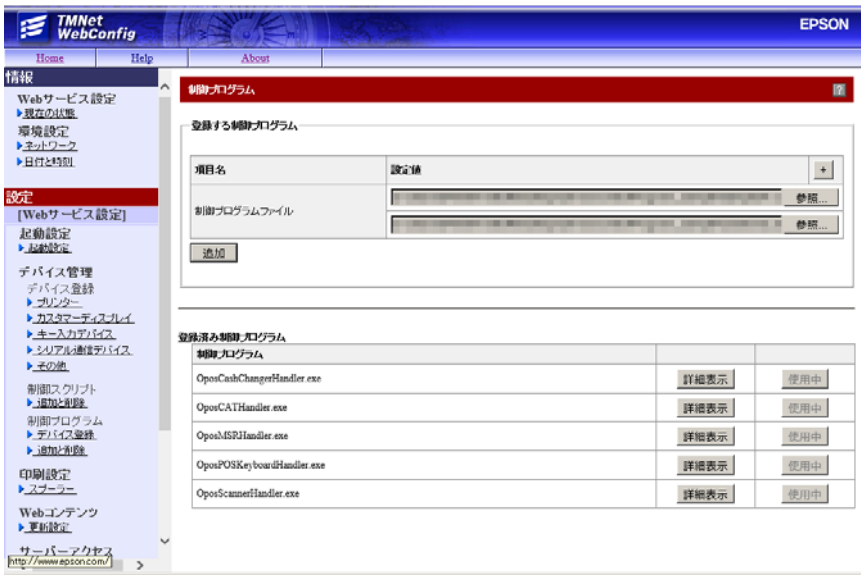
1 **プリンターにネットワークを設定します。**

2 **サンプルプログラム (ePOS_SDK_Sample_JavaScript.zip) をプリンターに登録します。**

EPSON TMNet WebConfig の「Web サービス設定」-「Web コンテンツ」-「更新設定」から手動更新機能を使用して登録します。

3 デバイス制御プログラムのサンプルプログラムをプリンターに登録します。

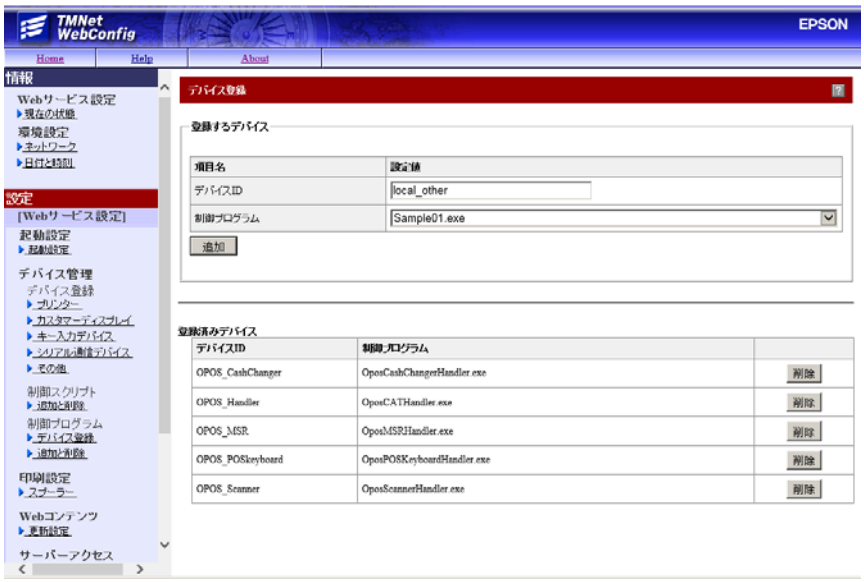
EPSON TMNet WebConfig の「Web サービス設定」-「制御プログラム」-「追加と削除」から制御プログラムファイルを選択して追加します。
デバイス制御プログラム用のサンプルプログラムファイルから、Sample01.exe と GGateway.dll を登録します。



4 デバイスを登録します。

EPSON TMNet WebConfig の「Web サービス設定」-「制御プログラム」-「デバイス登録」から登録します。

デバイス	設定項目	設定値
その他	デバイス ID	local_other
	制御プログラム	Sample01.exe



単機能モデル / モバイルモデル

各手順の詳細は、各プリンターの詳細取扱説明書を参照してください。

1 **サンプルプログラム (ePOS_SDK_Sample_JavaScript.zip) を展開し、Web サーバーの以下のフォルダーにコピーします。**

システムドライブ :\Inetpub\wwwroot（IIS で構築した Web サーバーの場合）

サンプルプログラムは管理者権限のユーザーでコピーしてください。

2 **ePOS-Print 設定を参照し、TM プリンターに ePOS-Print を設定します。**

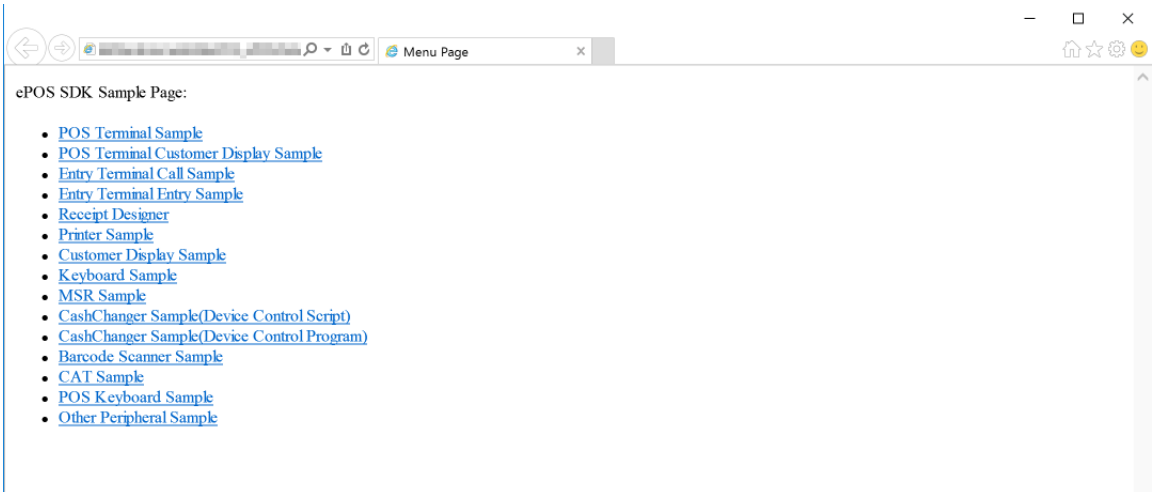
サンプルプログラムの起動

1 Web ブラウザーを起動し、以下の URL にアクセスします。

プリンター	URL
TM-DT シリーズ	http://[プリンターの IP アドレス]/index.ja.html
TM-i シリーズ	http://[プリンターの IP アドレス]/webapp/index.ja.html
単機能モデル / モバイルモデル	http://[サーバーの IP アドレス]/index.ja.html

2 Web ブラウザーでサンプルプログラムが表示されます。

表示されない場合は、プリンターのネットワーク接続状態を確認してください。



サンプルプログラムの使い方

- POS Terminal Sample
- Entry Terminal
- Receipt Designer
- Printer Sample
- Customer Display Sample
- Keyboard Sample
- MSR Sample
- CashChanger Sample (Device Control Script)
- CashChanger Sample (Device Control Program)
- Barcode Scanner Sample
- CAT Sample
- POSKeyboard Sample
- OtherPeripheral Sample

POS Terminal Sample

POS システムのような動作をさせることができます。

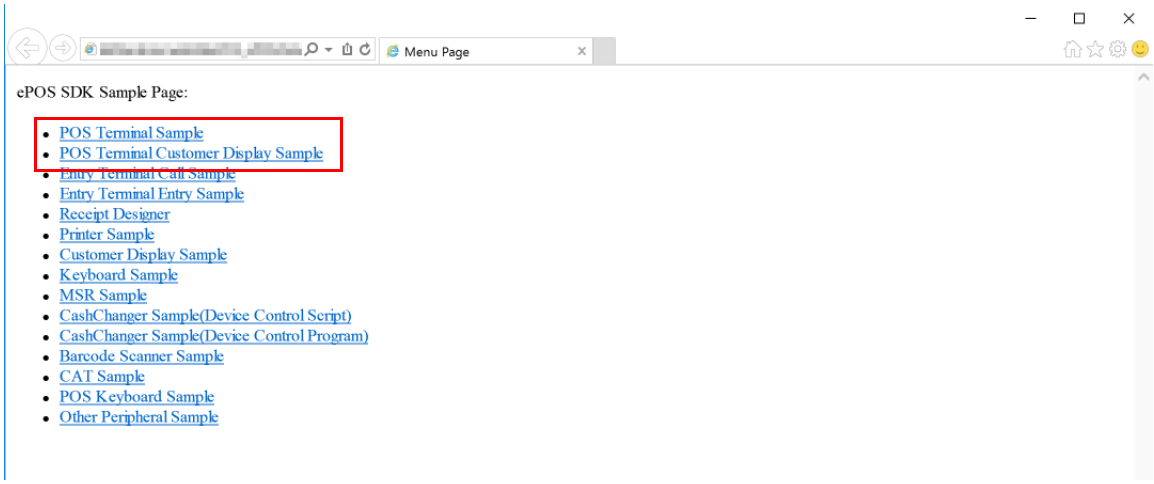
準備

サンプルプログラムを起動し、各デバイスを設定します。

以下のサンプルプログラムを、別々の Web ブラウザーで起動します。

- POS Terminal Customer Display Sample
- POS Terminal Sample

> 以降の手順に従い、[POS Terminal Customer Display Sample]-[POS Terminal Sample] の順番でサンプルプログラムと接続してください。
> 順序を誤ると、POS Terminal Customer Display Sample を正常に使用できません。



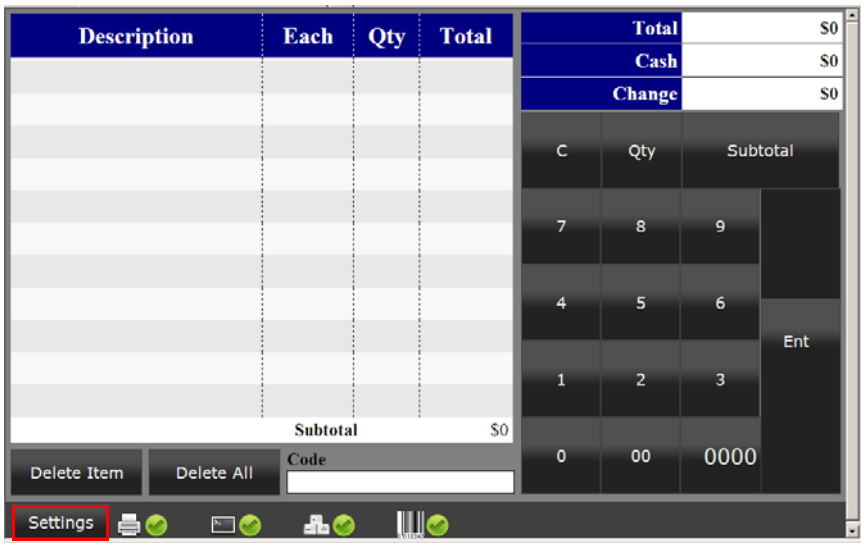
1 POS Terminal Customer Display Sample 画面を表示します。

2 "Connection Information" 画面が表示されます。
プリンターの IP アドレス / ポート番号を設定し、[Connect] ボタンを押します。

3 以下の画面が表示されます。

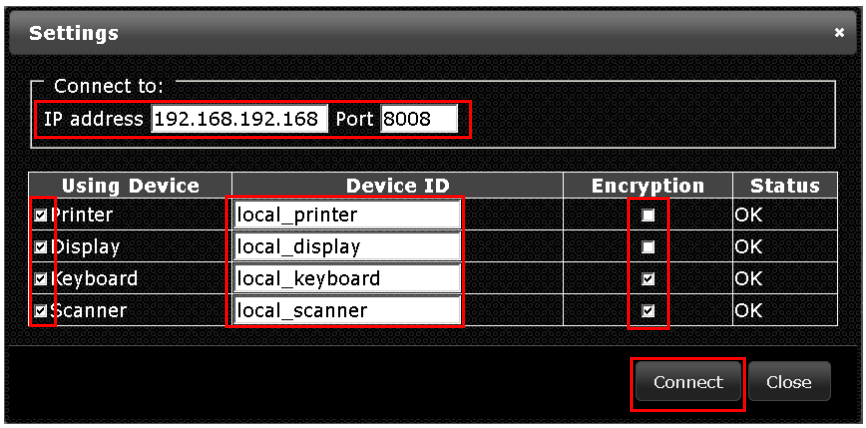
以上で、POS Terminal Customer Display Sample の設定は終了です。

4 "POS Terminal Sample" 画面を表示します。[Settings] ボタンを押します。



5 "Settings" 画面が表示されます。以下の設定をし、[Connect] ボタンを押します。ステータスが [OK] になることを確認後、画面を閉じます。

- ❏ [Connect to] を設定します。
- ❏ 使用するデバイスにチェックし、[Device ID] を入力します。



以上で、POS Terminal Sample の設定は終了です。

6 しばらくすると、POS Terminal Customer Display Sample 画面に、デジタルサイネージが表示されます。

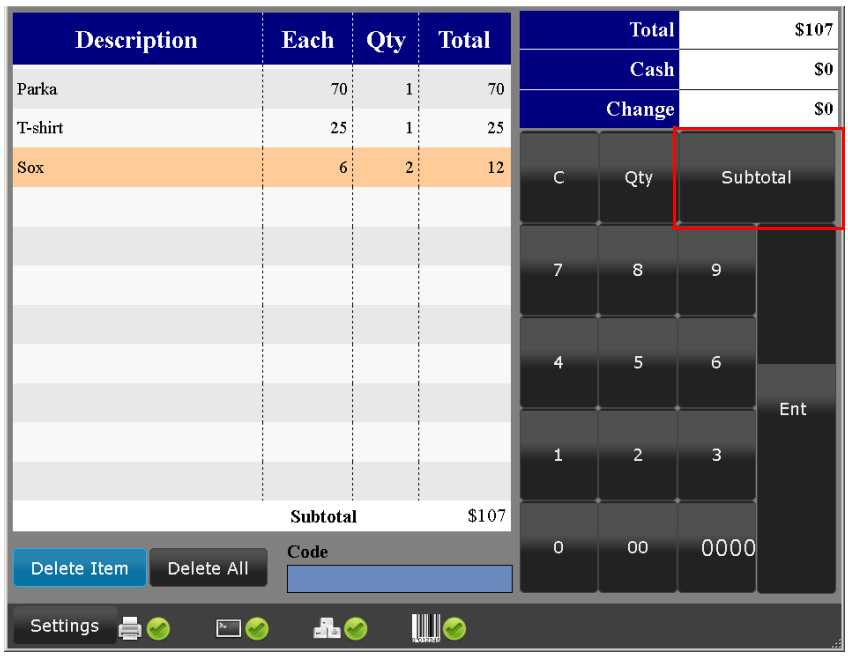
操作

商品の読み込みから、レシート発行までの手順を説明します。

> 商品のバーコードサンプルは、サンプルプログラムに PDF で付属されます。印刷してお使いください。

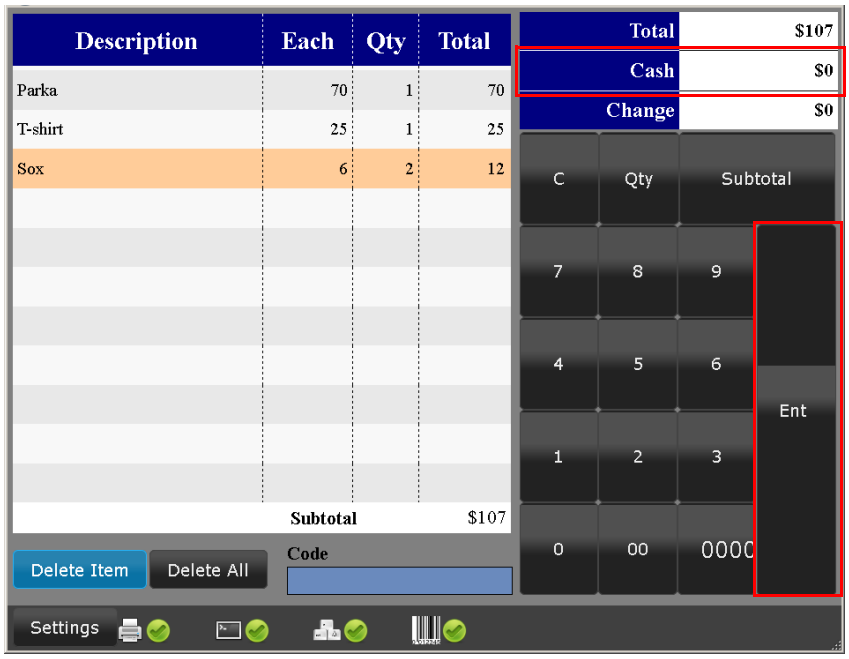
1 **バーコードスキャナーまたは、キーボードから商品入力します。**
商品入力後、[Subtotal] ボタンを押します。

[Delete Item] ボタンをクリックすると、選択中の商品が削除されます。また、[Delete All] ボタンを押すとすべての商品を削除します。
数量を変更する場合、商品入力後に [Qty] ボタンを押し、数字を入力します。数量を確定する時は [Ent] ボタンを押します。

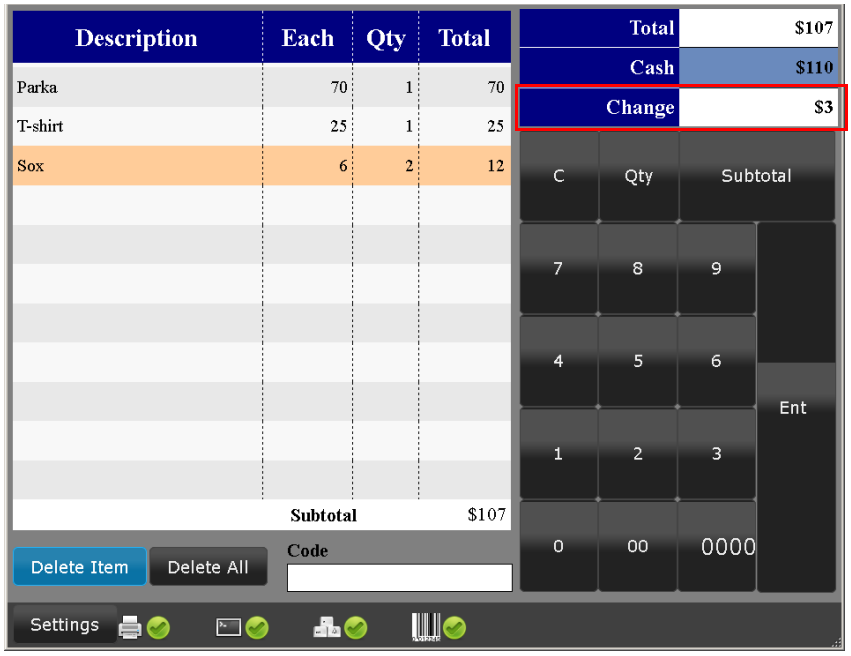


> POS Terminal Customer Display Sample の画面に、スキャンしたデータ、および [Total] が表示されます。

2 預かり金額を入力します。金額入力後、[Ent] ボタンを押します。



3 つり銭が表示され、プリンターからレシートが印刷されます。

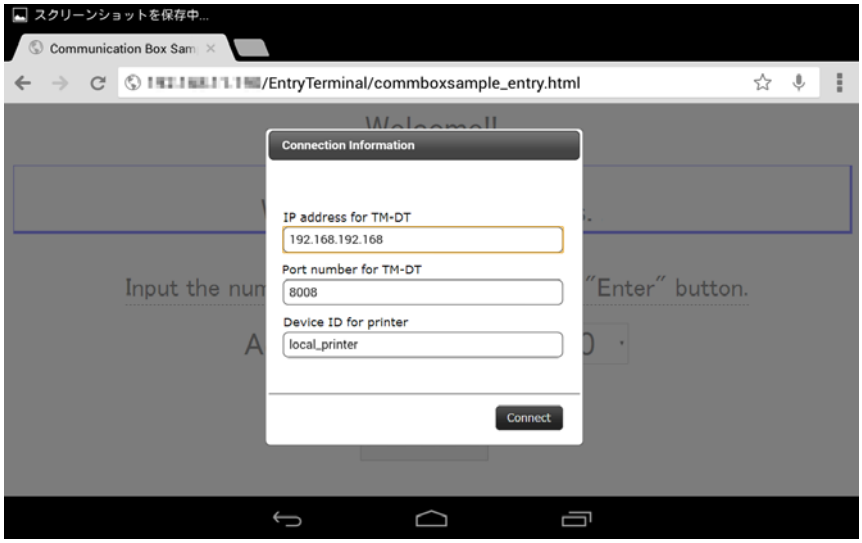


Entry Terminal

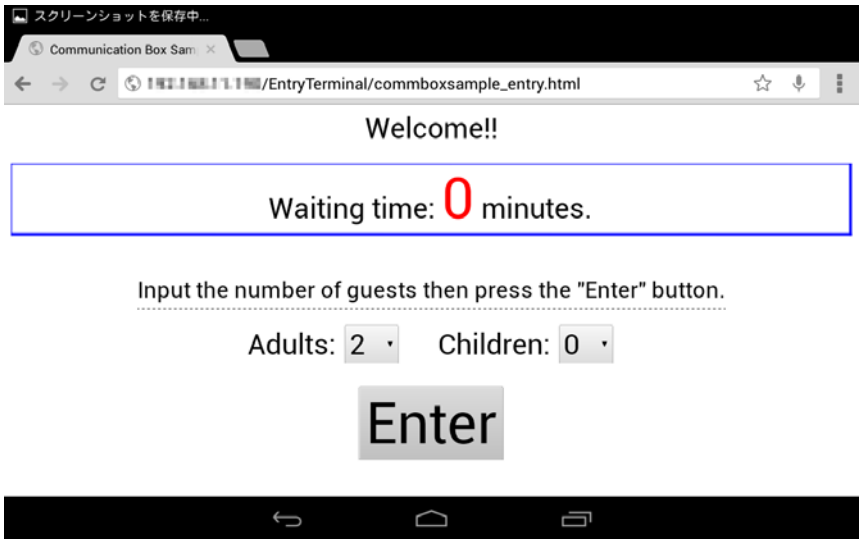
受付端末を想定したサンプルプログラムです。
コミュニニケーションボックスの機能を使用し、アプリケーション間でデータの通信を行います。

1 受付側のサンプルプログラムを起動します。[Entry Terminal Entry Sample] を選択します。

2 "Connection Information" 画面が表示されます。
プリンターの IP アドレス / ポート番号 / プリンターのデバイス ID を設定し、[Connect] ボタンを押します。

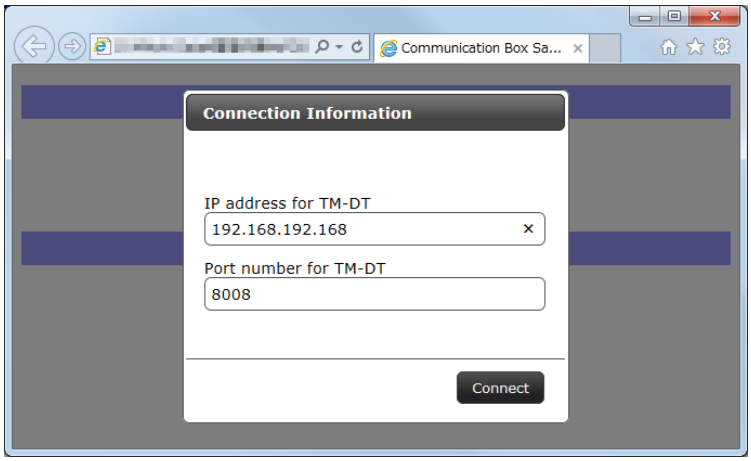


3 Entry Terminal Entry Sample 画面が表示されます。

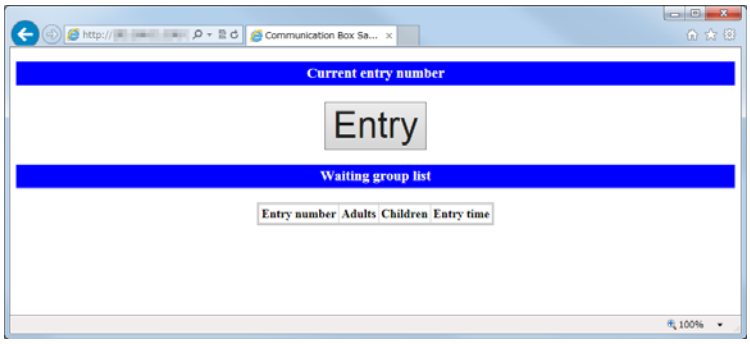


4 呼び出し側のサンプルプログラムを起動します。[Entry Terminal Call Sample] を選択します。

5 "Connection Information" 画面が表示されます。
プリンターの IP アドレス / ポート番号を設定し、[Connect] ボタンを押します。

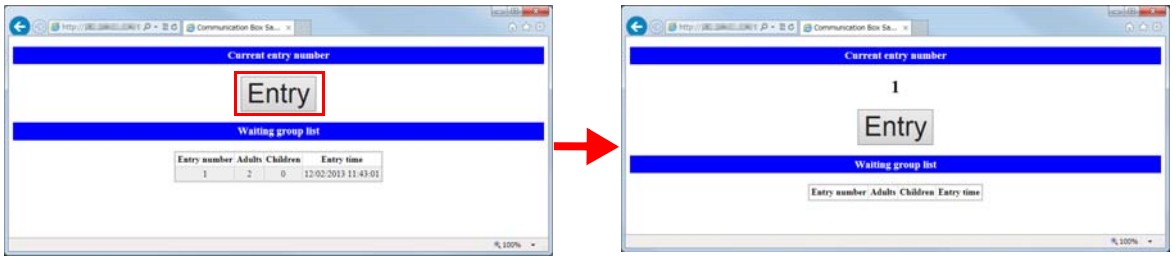


6 Entry Terminal Call Sample 画面が表示されます。

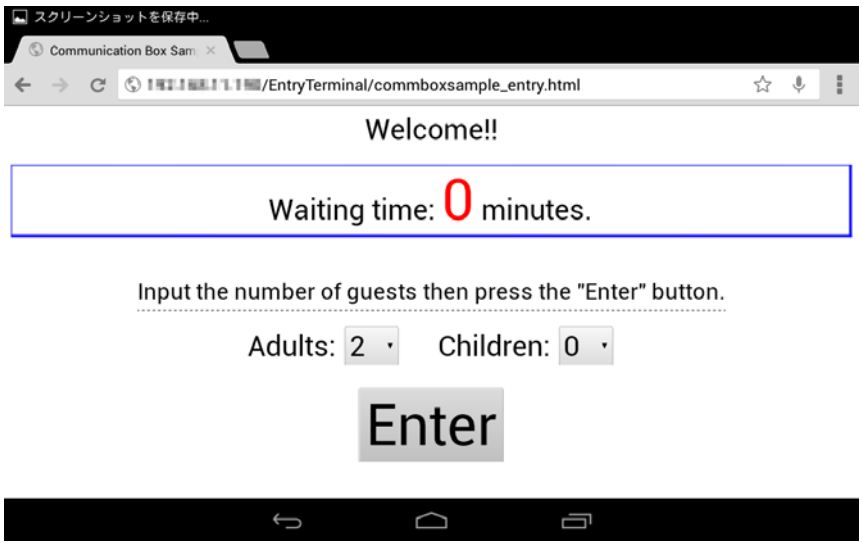


7 Entry Terminal Entry Sample 画面にしたがって操作します。
受付番号が、プリンターから印刷され、待ち時間が表示されます。

8 手順 7 の操作をすると、Entry Terminal Call Sample 画面に受付番号が表示され、[Entry] が有効になります。[Entry] を押します。



9 Entry Terminal Entry Sample 画面の待ち時間が更新されます。



Receipt Designer

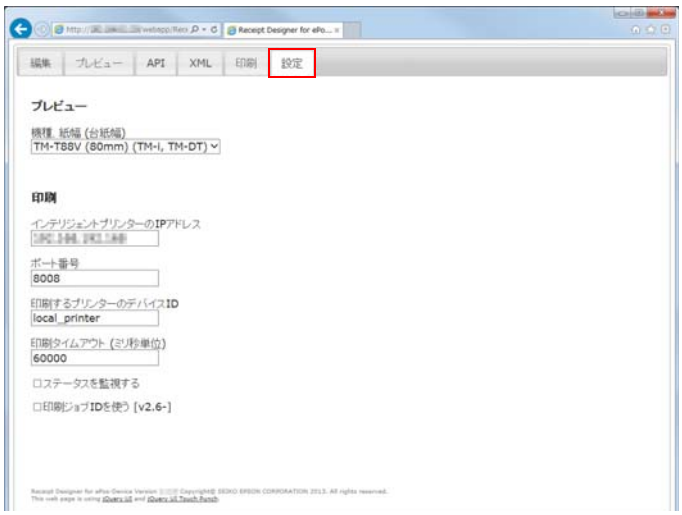
レシートプリンターの印刷データのサンプルコードを簡単に生成し、印刷できます。

- Google Chrome では、画像を含むプレビューを表示すると「SECURITY_ERR: DOM Exception 18」エラーが発生します。
- Windows Internet Explorer 9 では、印刷を実行すると「SCRIPT5: Access is denied.」エラーが発生します。

準備

1 サンプルプログラムを起動します。[Receipt Designer] を選択します。

2 "Receipt Designer for ePOS-Device" 画面が表示されます。[設定] タブを押します。



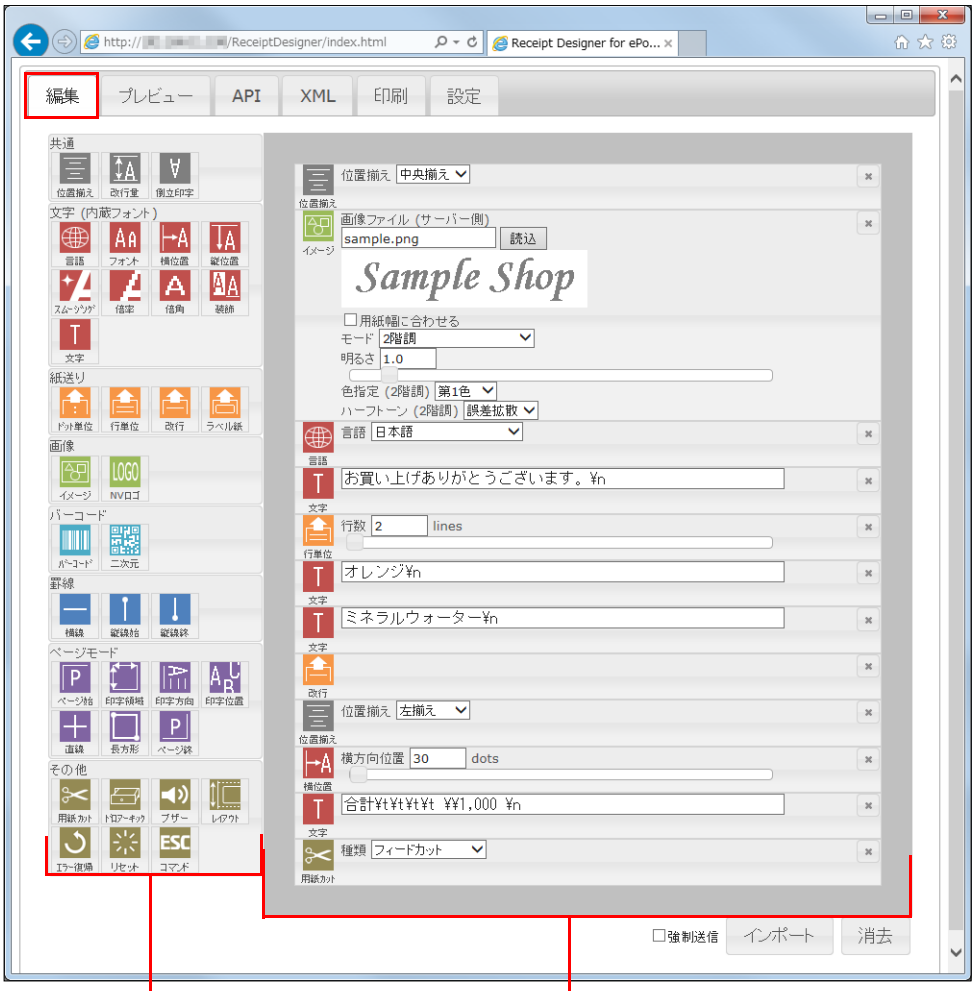
以下を設定します。

機能名	説明
機種、紙幅（台紙幅）	使用するプリンターの機種と紙幅を選択します。 機種の用紙幅にあわせてプレビュー画面が変化します。
インテリジェントプリンターの IP アドレス	プリンターの IP アドレスを指定します。 本項目は必ず指定してください。
ポート番号	TM-DT ソフトウェアのポート番号を指定します。 通常は、8008 を指定します。SSL/TLS 通信の場合、8043 を指定します。
印刷するプリンターのデバイス ID	プリンターのデバイス ID を指定します。 本項目は必ず指定してください。
印刷タイムアウト（msec 単位）	印刷のタイムアウト時間を msec 単位で指定します。 最大値は 60000（60 秒）です。
ステータスを監視する	チェックすると、プリンターのステータスを監視します。
印刷ジョブ ID を使う *	チェックすると、印刷ジョブ ID を付与して印刷します。

* TM-DT ソフトウェア Ver.3.0 以降、TM-i ファームウェア Ver.4.1 以降対応。

サンプルコードの作成

[編集] タブを選択すると、編集画面が表示されます。編集画面で Epson ePOS SDK for JavaScript のサンプルコードを作成します。



項目	説明
メニュー領域	使用できる機能が表示されます。 クリックで編集領域の最後に追加、ドラッグで編集領域の任意の位置に挿入できます。
編集領域	メニュー領域で選択された機能が表示されます。 要素の順序は、ドラッグによる入れ替えが可能です。要素右の × ボタンで要素を削除できます。
インポート	Receipt Designer は、ePOS-Device XML を使用して、過去に保存した XML データをインポートできます。詳細はインポートを参照してください。
消去	編集内容を消去します。
強制送信	強制送信モードの設定をします。

サンプルコードを以下の手順で作成します。

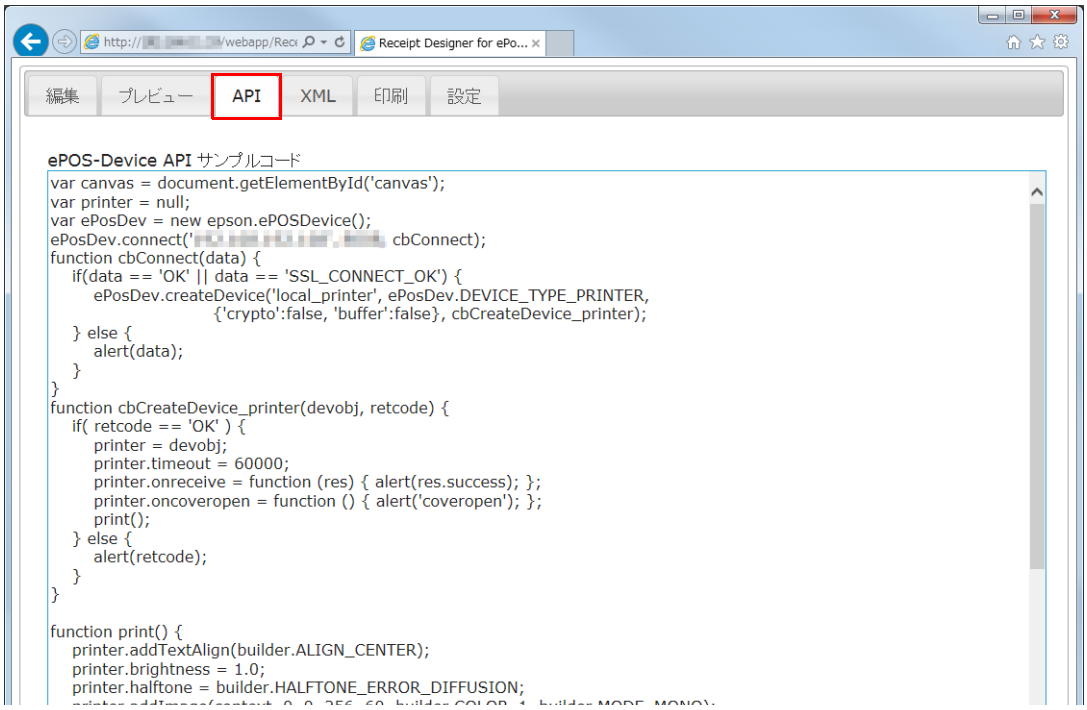
1 **メニュー領域のアイコンを押し、編集領域に要素を追加します。**
追加した機能は、ドラッグによる順番の入れ替えが可能です。

2 **追加した要素を設定します。**
例 :NV ロゴを追加した場合はキーコードを設定します。

3 **[プレビュー] タブを選択し、プレビューを確認します。**
プリンターが接続されている場合は、印刷して確認することもできます。
（詳細は印刷を参照してください。）

- ロゴ印字、バーコード印字、2D コード印字、ESC コマンド、ブザーの鳴動、ドロアーキック、用紙カットはアイコンで表示されます。
- プレビュー設定によって、レイアウトが変わる場合があります。
（詳細は、準備を参照してください）

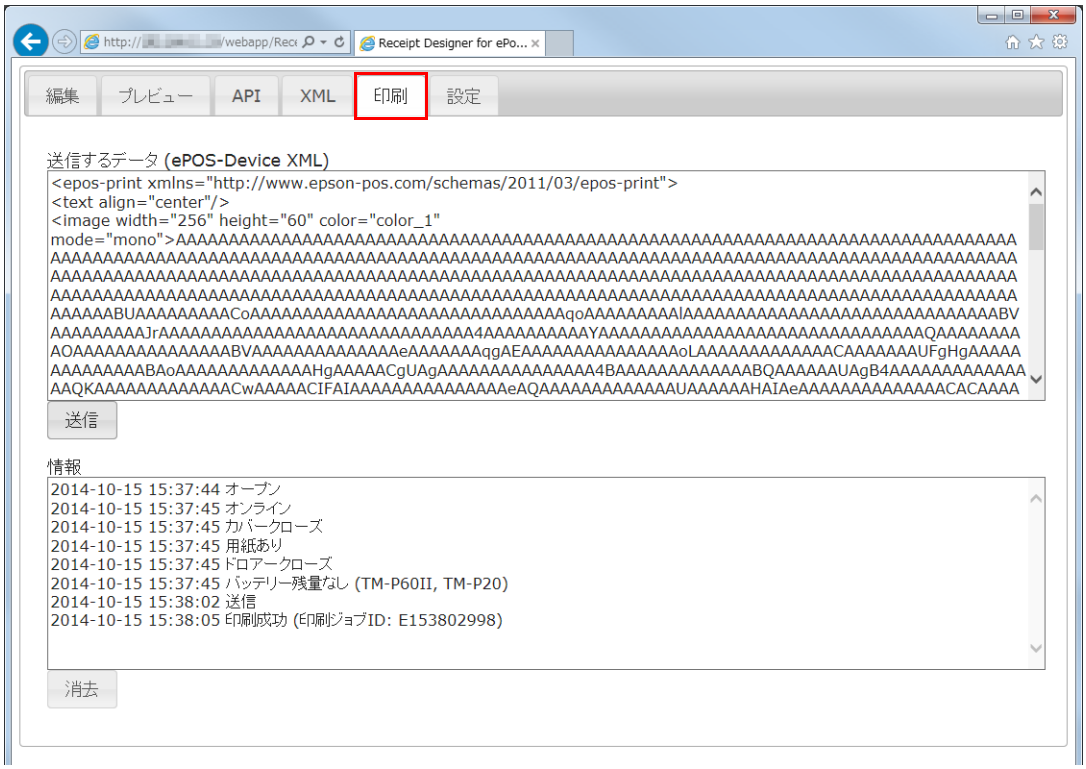
4 [API] タブを選択します。Epson ePOS SDK for JavaScript サンプルコードが表示されます。コピーしてお使いください。



> インポートで ePOS-Device XML の印刷ドキュメントを使用します。
> 必要な方は、XML タブを選択し ePOS-Device XML の印刷ドキュメントの内容をコピーして保存しておいてください。

印刷

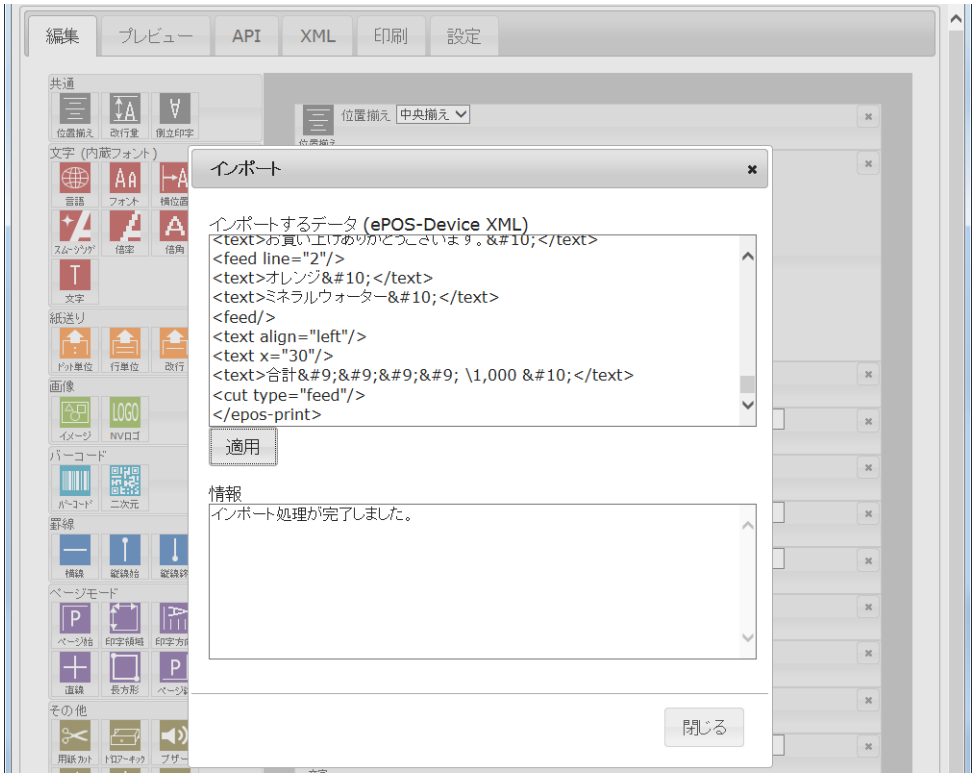
プリンター設定に基づいて、印刷ドキュメントをプリンターでテスト印字します。
（プリンター設定に関しては準備を参照してください）


http://webapp/Rec
Receipt Designer for ePo...
編集
プレビュー
API
XML
印刷
設定
送信するデータ (ePOS-Device XML)
<epos-print xmlns="http://www.epson-pos.com/schemas/2011/03/epos-print">
<text align="center"/>
<image width="256" height="60" color="color_1"
mode="mono">AAAA

インポート

Receipt Designer は、インポートして一度作成した ePOS-Device XML の印刷ドキュメントを再編集することができます。

> Epson ePOS SDK for JavaScript のソースコードを使用したインポートはできません。
> ePOS-Device XML の印字データを使用してインポートを行ってください。



項目	説明
インポートするデータ (ePOS-Device XML):	インポートする ePOS-Device XML の印刷ドキュメントの貼り付け、確認をします。
適用	ePOS-Device XML の印刷ドキュメントをインポートします。
情報	インポート情報が表示されます。
閉じる	インポートウィンドウを閉じます。

Receipt Designer は、以下の手順で ePOS-Device XML の印刷ドキュメントをインポートできます。

1 [編集] タブを選択し、[インポート] ボタンを押します。

2 [インポート] ウィンドウが表示されます。[インポートするデータ (ePOS-Device XML)] に ePOS-Device XML の印刷ドキュメントを貼り付けます。

3 [適用] ボタンを押します。

4 [確認] 画面が表示されます。[はい] ボタンを押します。

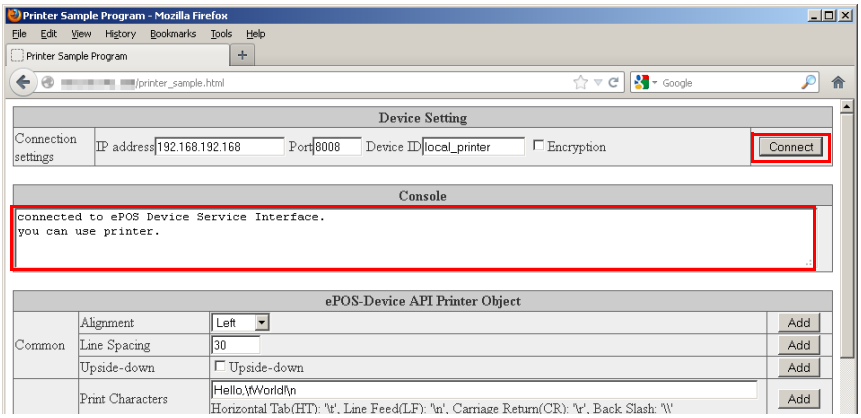
Printer Sample

プリンターを動作させたり、プリンターのサンプルコードを生成したりすることができます。

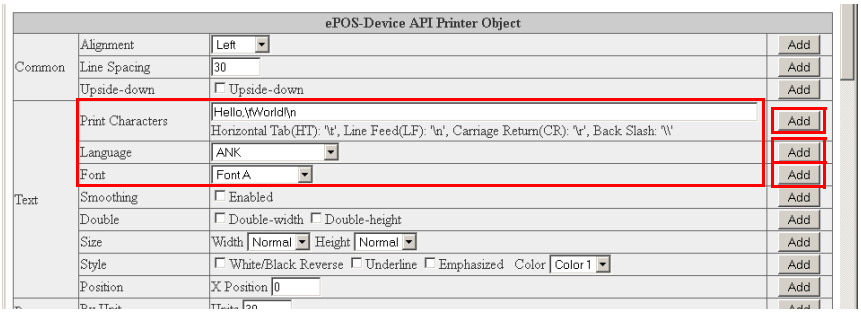
1 サンプルプログラムを起動します。[Printer Sample] を選択します。

2 "Printer Sample" 画面が表示されます。[Connect] ボタンを押します。
接続に成功すると、[Console] に以下のメッセージが表示されます。

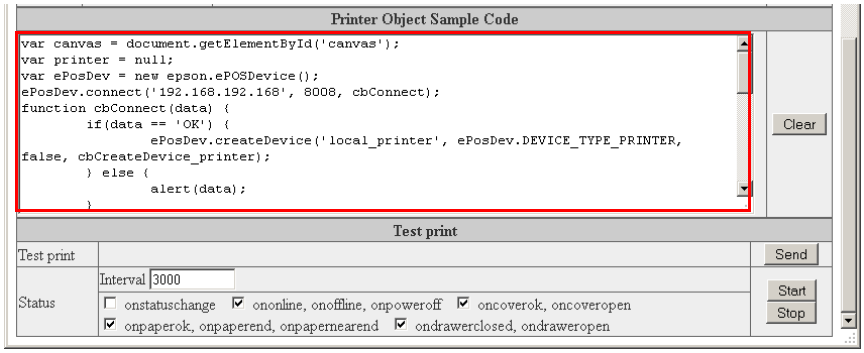
> [Device Setting] の設定を変更する必要はありません。ただし、IP アドレスなどを変更している場合、変更した値に合わせてください。



3 動作させたい機能を設定し、[Add] ボタンを押します。



4 [Printer Object Sample Code] に、手順 3 で追加した機能のソースが表示されます。
コピーして使用することができます。



5 [Send] ボタンをクリックすると、手順 3 で追加した機能に合わせてテスト印字されます。

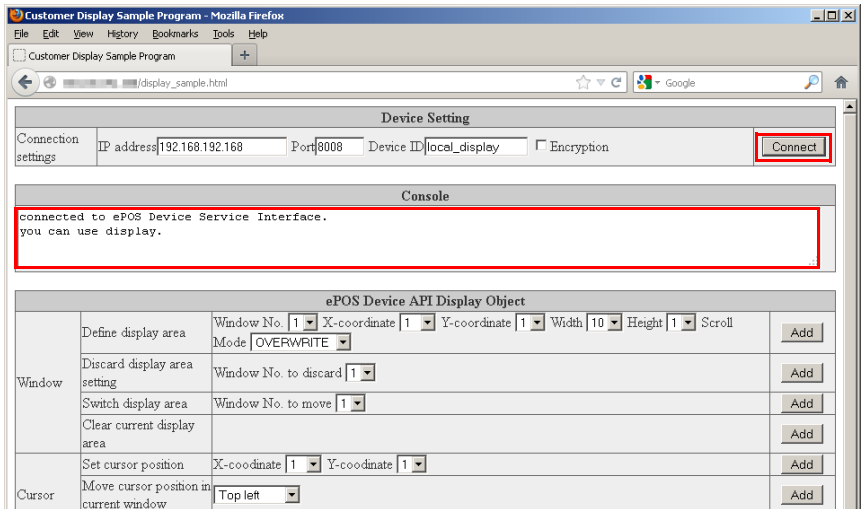
Customer Display Sample

カスタマーディスプレイに表示させるサンプルコードを生成させたり、生成させたコードを送信してカスタマーディスプレイを動作させたりできます。

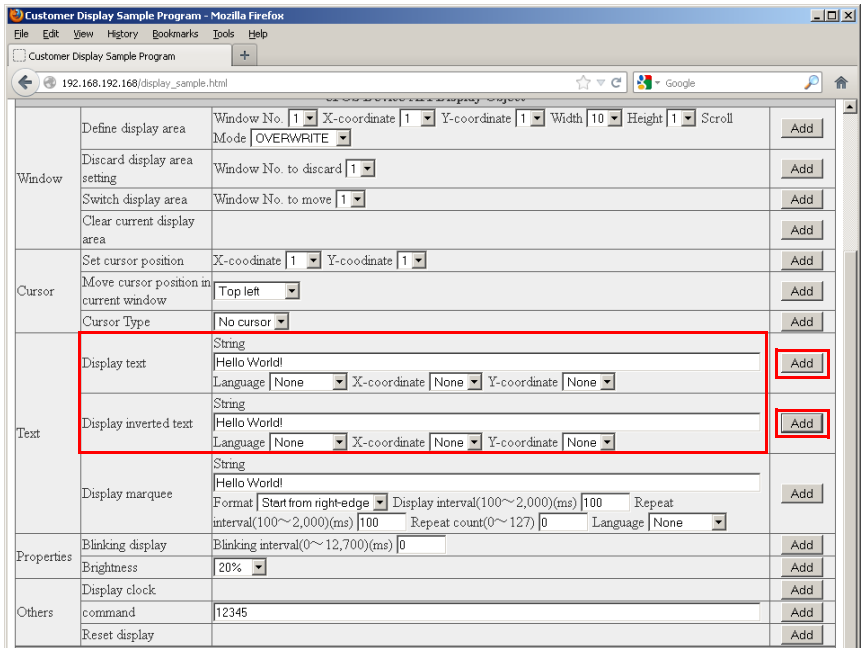
1 サンプルプログラムを起動します。[Customer Display Sample] を選択します。

2 "Customer Display Sample" 画面が表示されます。[Connect] ボタンを押します。接続に成功すると、[Console] に以下のメッセージが表示されます。また、カスタマーディスプレイの表示が変わります。

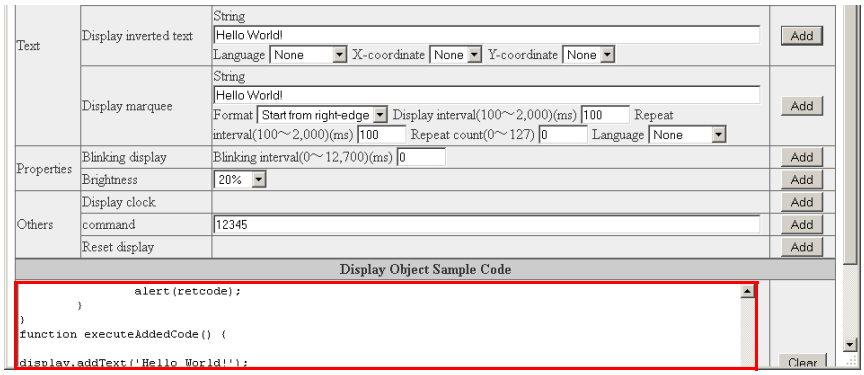
> [Device Setting] の設定を変更する必要はありません。ただし、IP アドレスなどを変更している場合、変更した値に合わせてください。



3 動作させたい機能を設定し、[Add] ボタンを押します。



4 [Display Object Sample Code] に、手順 3 で追加した機能のソースが表示されます。コピーして使用することができます。



5 [Send] ボタンをクリックすると、手順 3 で追加した機能に合わせてカスタマーディスプレイの表示が変わります。

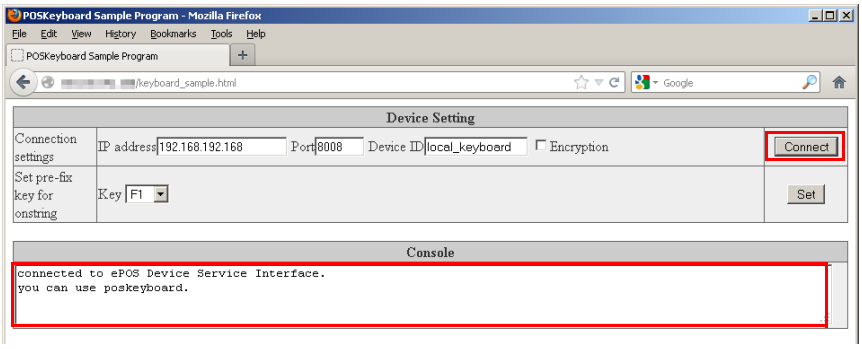
Keyboard Sample

キーボードからの入力を取得することができます。

1 **サンプルプログラムを起動します。[Keyboard Sample] を選択します。**

2 **"Keyboard Sample" 画面が表示されます。[Connect] ボタンを押します。**
接続に成功すると、[Console] に以下のメッセージが表示されます。

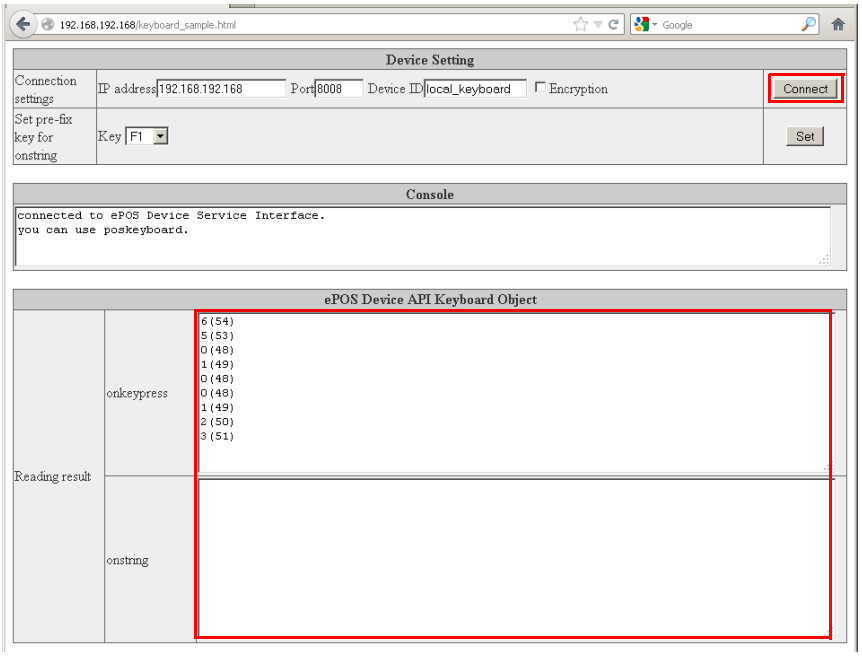
> [Device Setting] の設定を変更する必要はありません。ただし、IP アドレスなどを変更している場合、変更した値に合わせてください。



3 **文字列の始まりと判断するキーを選択し、[Set] ボタンを押します。**
あらかじめキーボード側の登録が必要です。
例：F1＋ 指定文字列 +Enter

4 **キーボードから入力すると、入力結果が [ePOS Device API Keyboard Object] に表示されます。**

文字列を直接入力：	onkeyress に表示されます。
キーに設定された文字列を入力：	文字列が指定されたキーを押すと、onstring に表示されます。



MSR Sample

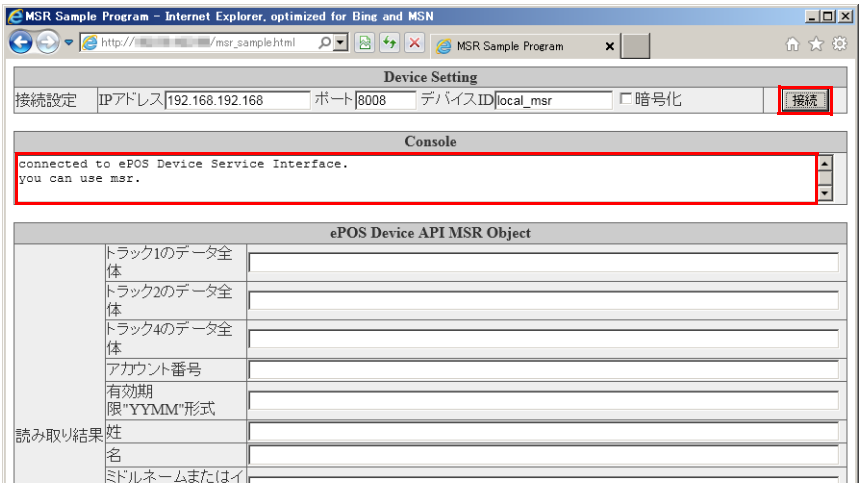
MSR からの入力を取得することができます。

> TM-DT シリーズのみ使用できます。

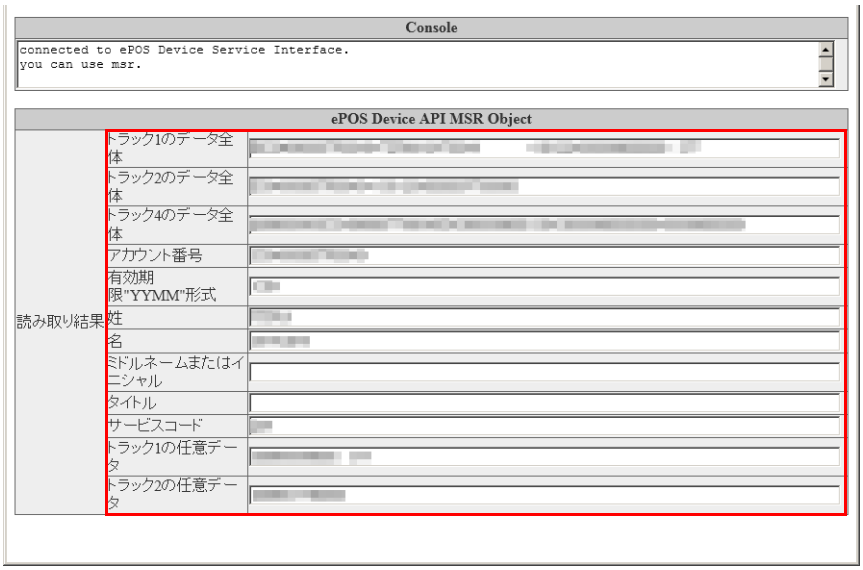
1 サンプルプログラムを起動します。[MSR Sample] を選択します。

2 "MSR Sample" 画面が表示されます。[接続] ボタンを押します。
接続に成功すると、[Console] に以下のメッセージが表示されます。

> - 標準HIDのMSRを使用する場合、[Device Setting] の設定を変更する必要はありません。ただし、IP アドレスなどを変更している場合は、変更した値に合わせてください。
> - OPOS仕様準拠のMSRを使用する場合は、デバイスIDにプリンターの環境設定で設定した論理デバイス名を入力してください。



3 カードを MSR に読み込ませます。読み込ませた情報が [ePOS Device API MSR Object] に表示されます。



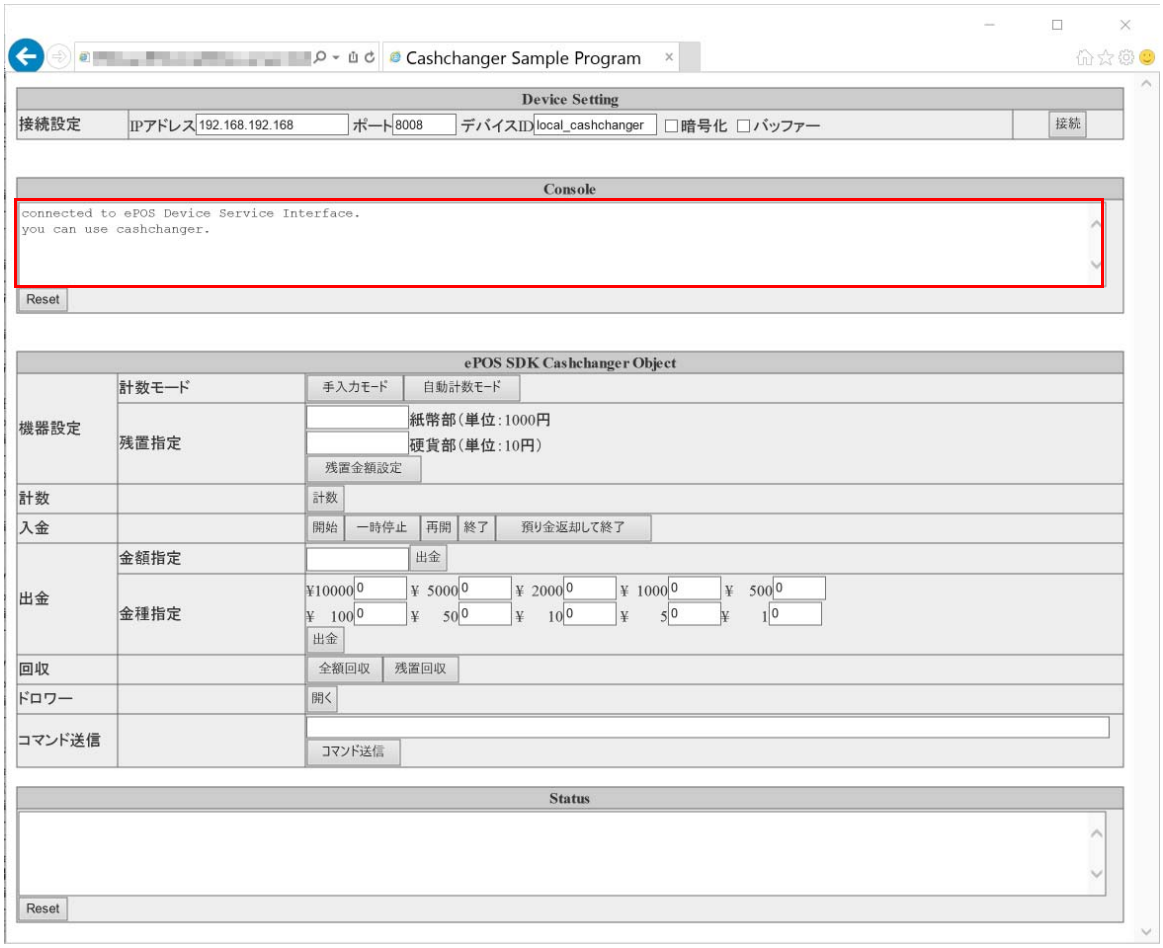
CashChanger Sample (Device Control Script)

デバイス制御スクリプトから自動つり銭機を制御できます。

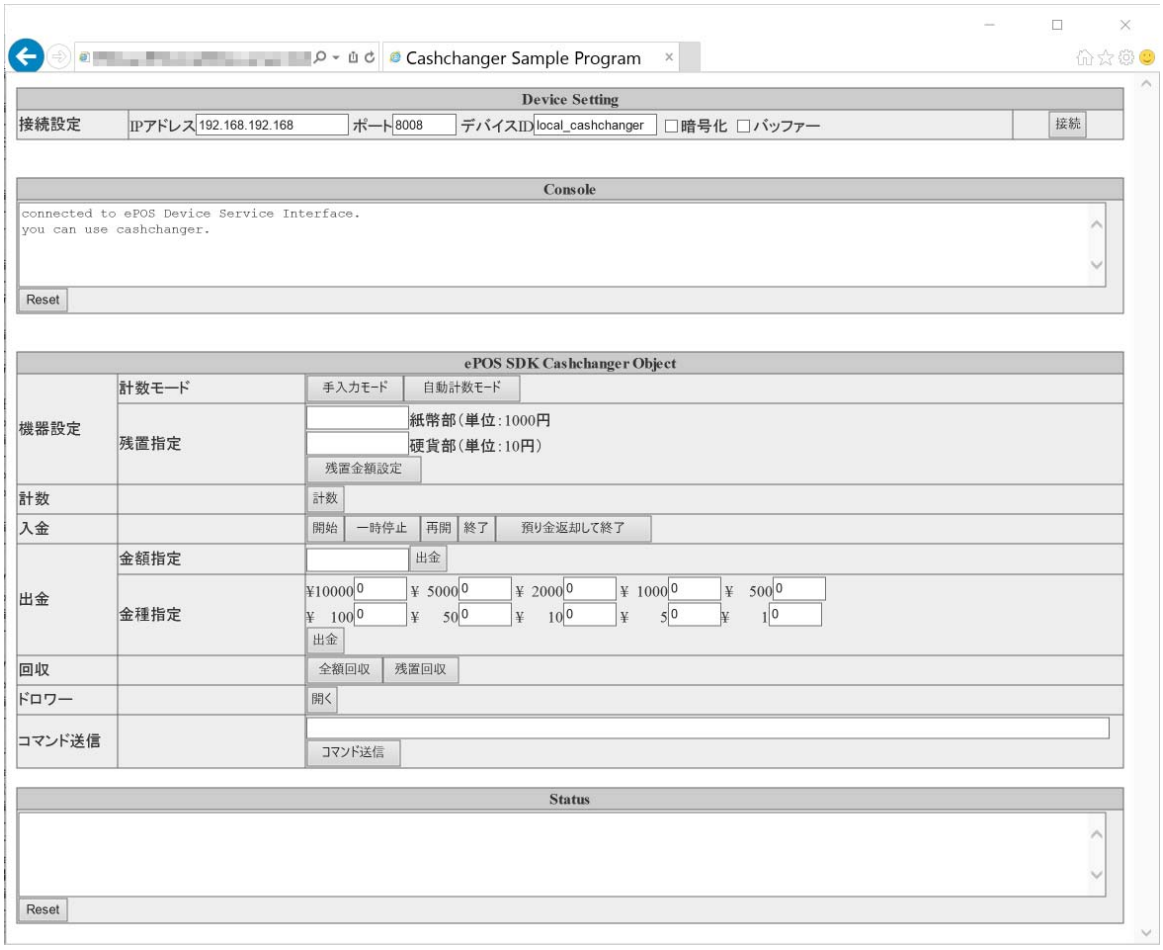
1 **サンプルプログラムを起動します。[CashChanger Sample (Device Control Script)] を選択します。**

2 **"CashChanger Sample" 画面が表示されます。[接続] ボタンを押します。接続に成功すると、[Console] に以下のメッセージが表示されます。**

> [Device Setting] の設定を変更する必要はありません。ただし、IP アドレスなどを変更している場合、変更した値に合わせてください。



3 使いたい機能を選択して、動作を設定します。



機能		説明
機器設定	計数モード	計数モードを設定します。
	残置指定	硬貨の残置金額を指定します。
計数		デバイス内現金の金額を取得します。
入金		入金処理を操作します。
出金	金額指定	金額を指定して出金します。
	金種指定	金種を指定して出金します。
回収		デバイス内の現金を回収します。
ドロワー		キャッシュドロアーを開きます。
コマンド送信		任意コマンドを送信します。

4 機能の実行結果が、Console に表示されます。

CashChanger Sample (Device Control Program)

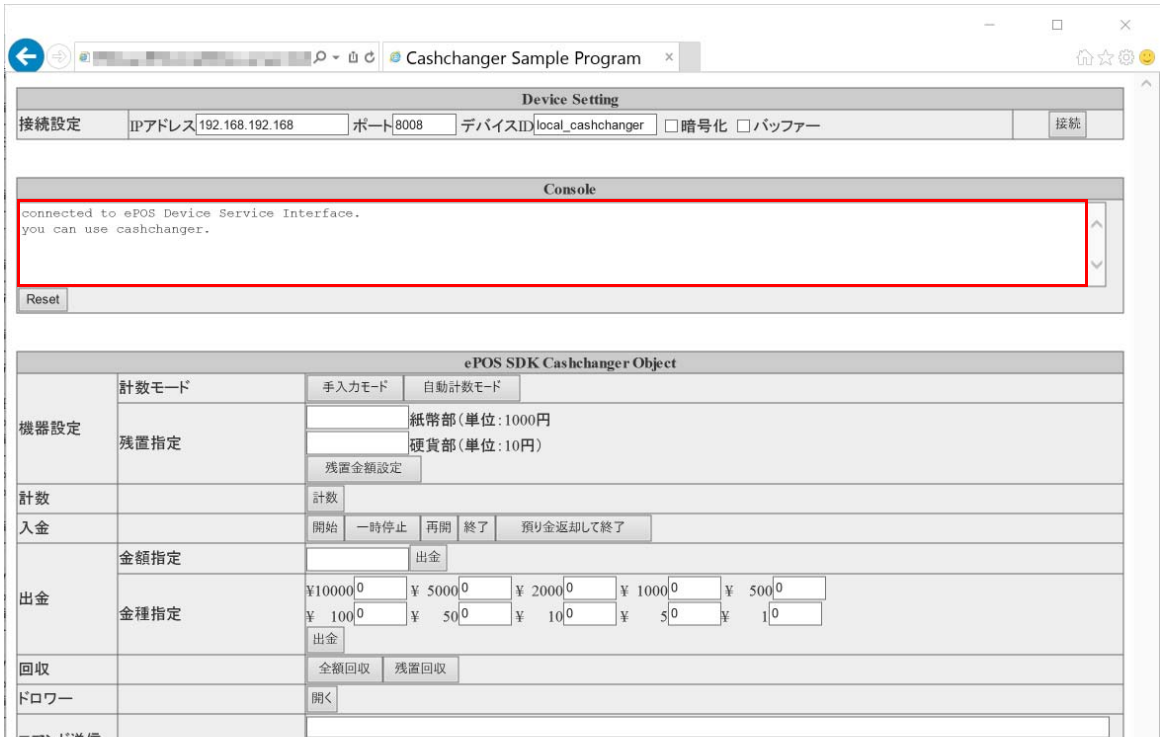
デバイス制御プログラムから自動つり銭機を制御できます。

1 サンプルプログラムを起動します。[CashChanger Sample (Device Control Program)] を選択します。

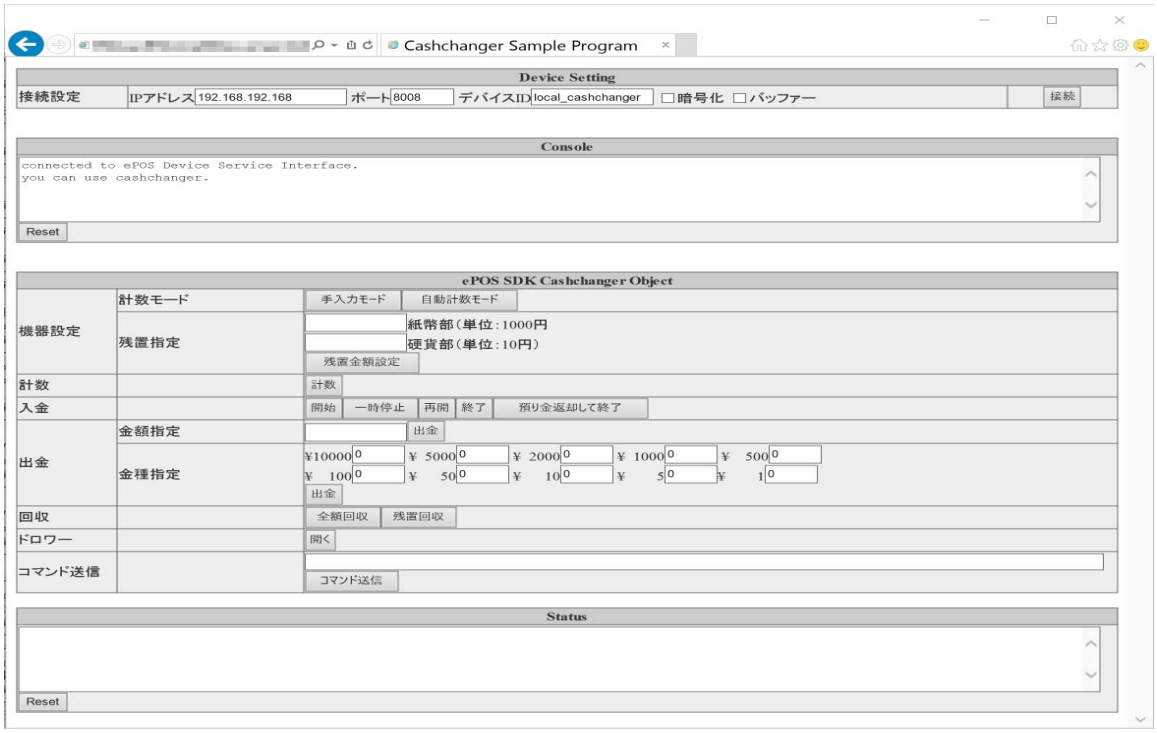
2 "CashChanger Sample" 画面が表示されます。デバイス ID にプリンターの環境設定で設定した論理デバイス名を入力し、[接続] ボタンを押します。

> IP アドレスなどを変更している場合は、変更した値に合わせてください。

3 接続に成功すると、[Console] に以下のメッセージが表示されます。



4 使いたい機能を選択して、動作を設定します。



機能		説明
計数		デバイス内現金の金額を取得します。
入金		入金処理を操作します。
出金	金額指定	金額を指定して出金します。
	金種指定	金種を指定して出金します。
コマンド送信		DirectIO に渡す値を指定します。

5 機能の実行結果が、Console に表示されます。

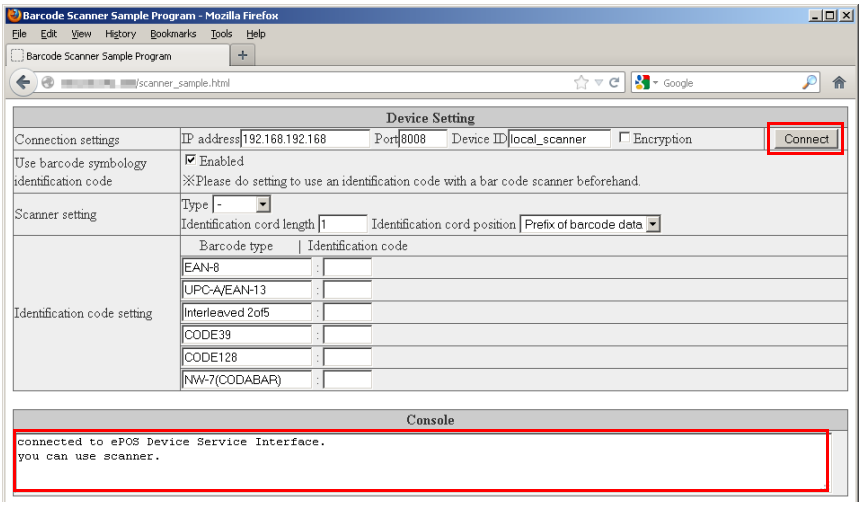
Barcode Scanner Sample

バーコードスキャナーからの入力を取得することができます。

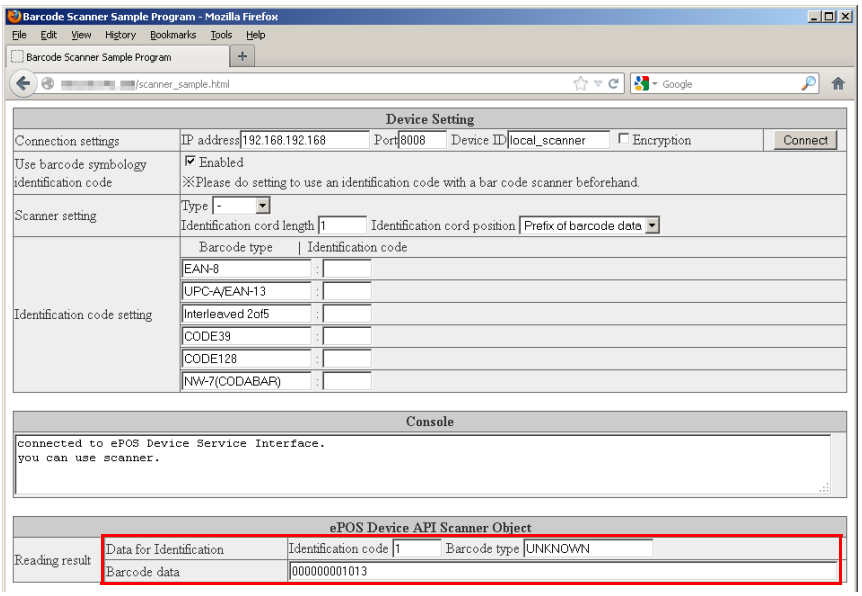
1 サンプルプログラムを起動します。[Barcode Scanner Sample] を選択します。

2 "Barcode Scanner Sample" 画面が表示されます。[Connect] ボタンを押します。接続に成功すると、[Console] に以下のメッセージが表示されます。

- 標準 HID のバーコードスキャナーを使用する場合、[Device Setting] の設定を変更する必要はありません。ただし、IP アドレスなどを変更している場合は、変更した値に合わせてください。
- OPOS仕様準拠のバーコードスキャナーを使用する場合は、デバイスIDにプリンターの環境設定で設定した論理デバイス名を入力してください。



3 バーコードを読み込ませます、読み込ませた情報が [ePOS Device API Scanner Object] に表示されます。



CAT Sample

信用照会端末を使って、決済処理と 1 日の集計を出すことができます。

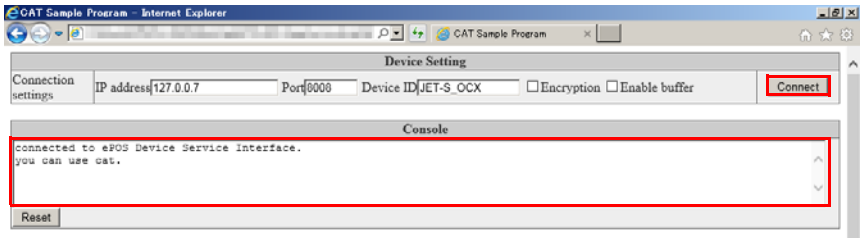
準備

1 サンプルプログラムを起動します。[CAT Sample] を選択します。

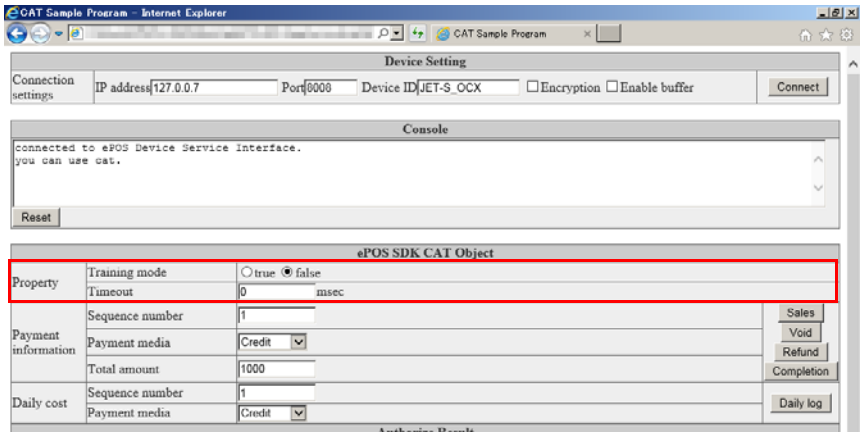
2 "CAT Sample" 画面が表示されます。デバイス ID にプリンターの環境設定で設定した論理デバイス名を入力し、[接続] ボタンを押します。

> IP アドレスなどを変更している場合は、変更した値に合わせてください。

3 接続に成功すると、[Console] に以下のメッセージが表示されます。

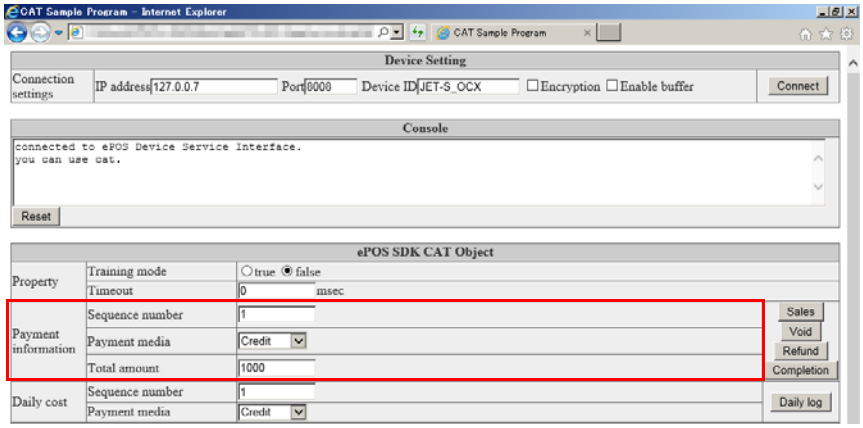


4 トレーニングモードとタイムアウト時間を指定します。



決済

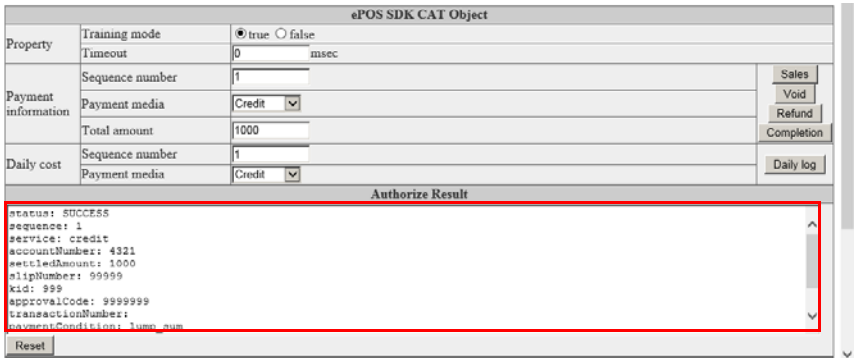
1 支払い情報を指定します。



2 実行する処理を選択すると、デバイス側は決済メディア待ち状態になります。

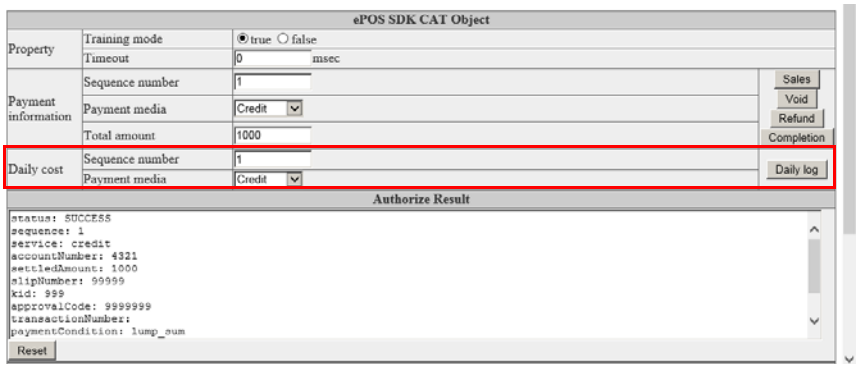
3 デバイスを操作し、決済処理を進めます。

4 決済処理を実行すると、［Authorize Result］に実行結果が表示されます。

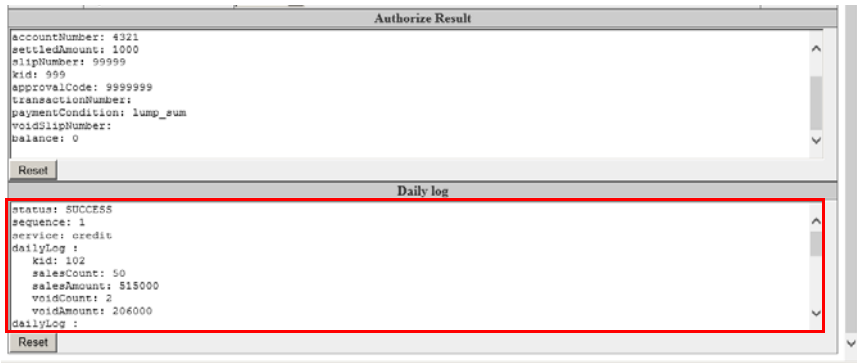


日計印刷

1 日計処理する決済メディア指定して［Daily log］ボタンを押すと、日計結果が印刷されます。



2 日計結果の印刷が終了すると、Daily log に処理の実行結果が表示されます。



POSKeyboard Sample

POS キーボードからの入力を取得することができます。

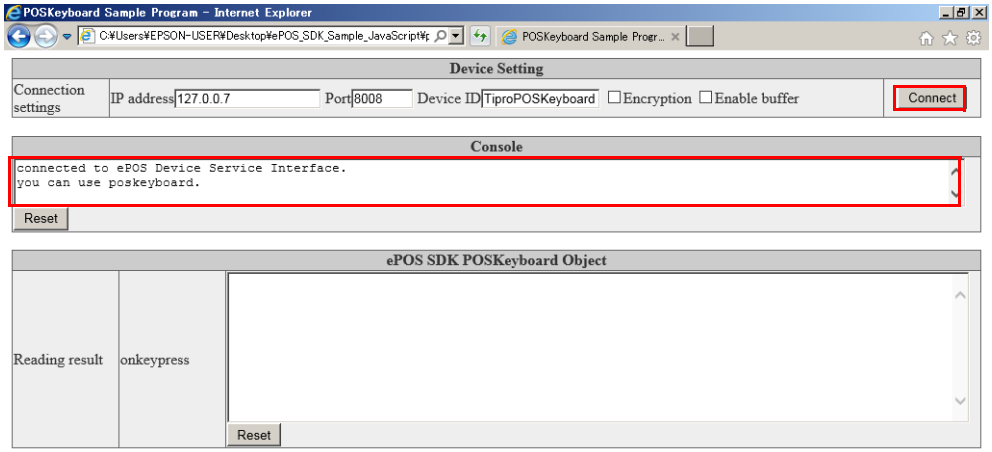
> あらかじめ POS キーボード側で、POS キーの登録が必要です。

1 サンプルプログラムを起動します。[POSKeyboard Sample] を選択します。

2 "POSKeyboard Sample" 画面が表示されます。デバイス ID にプリンターの環境設定で設定した論理デバイス名を入力し、[接続] ボタンを押します。

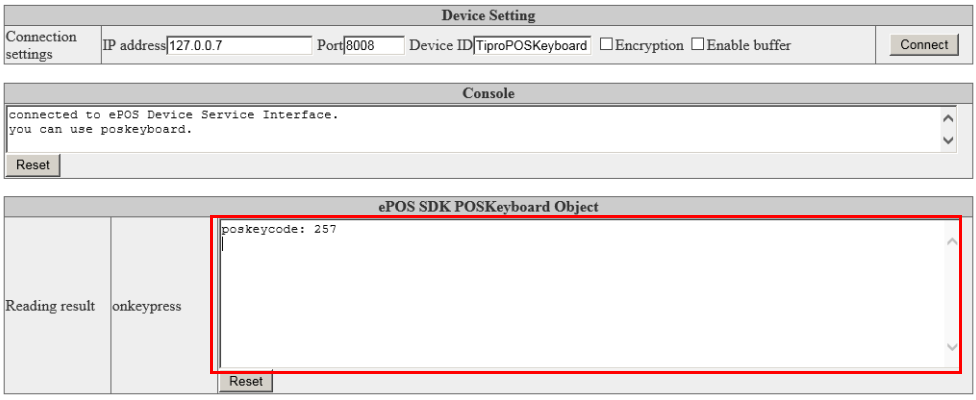
> IP アドレスなどを変更している場合は、変更した値に合わせてください。

3 接続に成功すると、[Console] に以下のメッセージが表示されます。



4 POS キーボードから入力すると、入力結果が [ePOS SDK POSKeyboard Object] に表示されます。

POS キーを入力：POS キーに設定されたキーコードが表示されます。



OtherPeripheral Sample

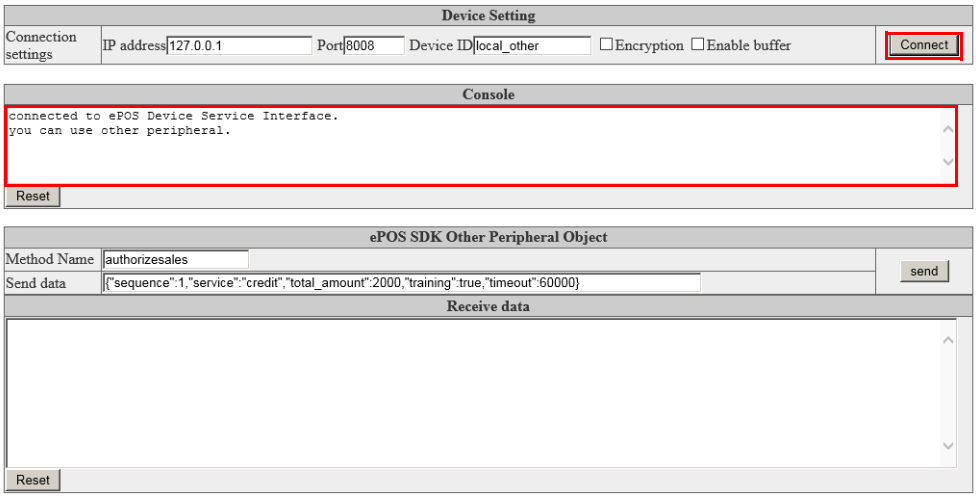
デバイス制御プログラムのサンプルプログラムを利用して、その他の周辺機器（デバイス制御プログラムで制御可能なデバイス）とのデータ送受信を確認できます。

> デバイス制御プログラムの追加と、デバイス登録が必要です。詳細は、Other Peripheral Sample を実行するを参照してください。

1 サンプルプログラムを起動します。[OtherPeripheral Sample] を選択します。

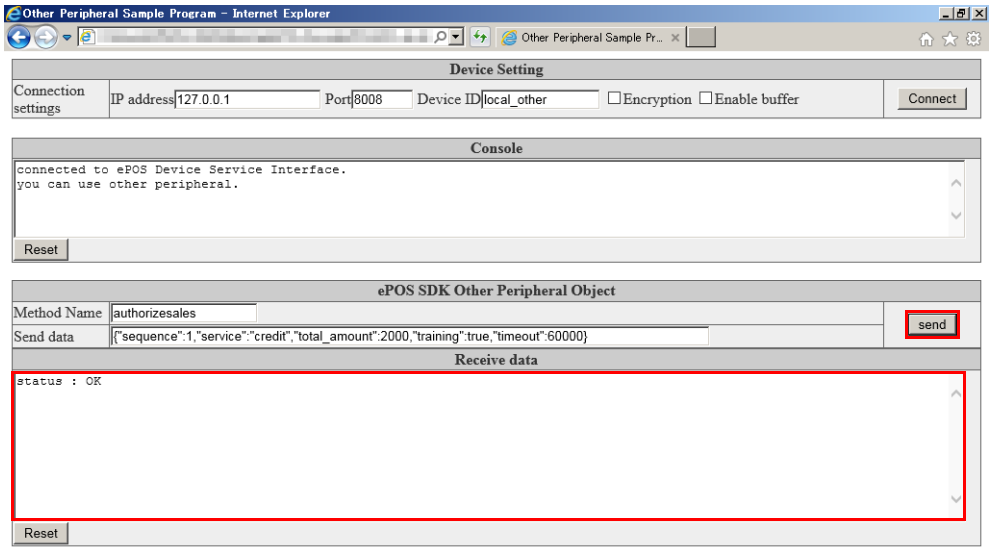
2 "OtherPeripheral Sample" 画面が表示されます。[Connect] ボタンを押します。接続に成功すると、[Console] に以下のメッセージが表示されます。

> [Device Setting] の設定を変更する必要はありません。ただし、IP アドレスなどを変更している場合、変更した値に合わせてください。



3 [send] ボタンを押して、デバイス制御プログラムにリクエストデータを送信します。

[Receive data] にデバイス制御プログラムからのレスポンスデータが表示されます。



活用ガイド

常時監視するには

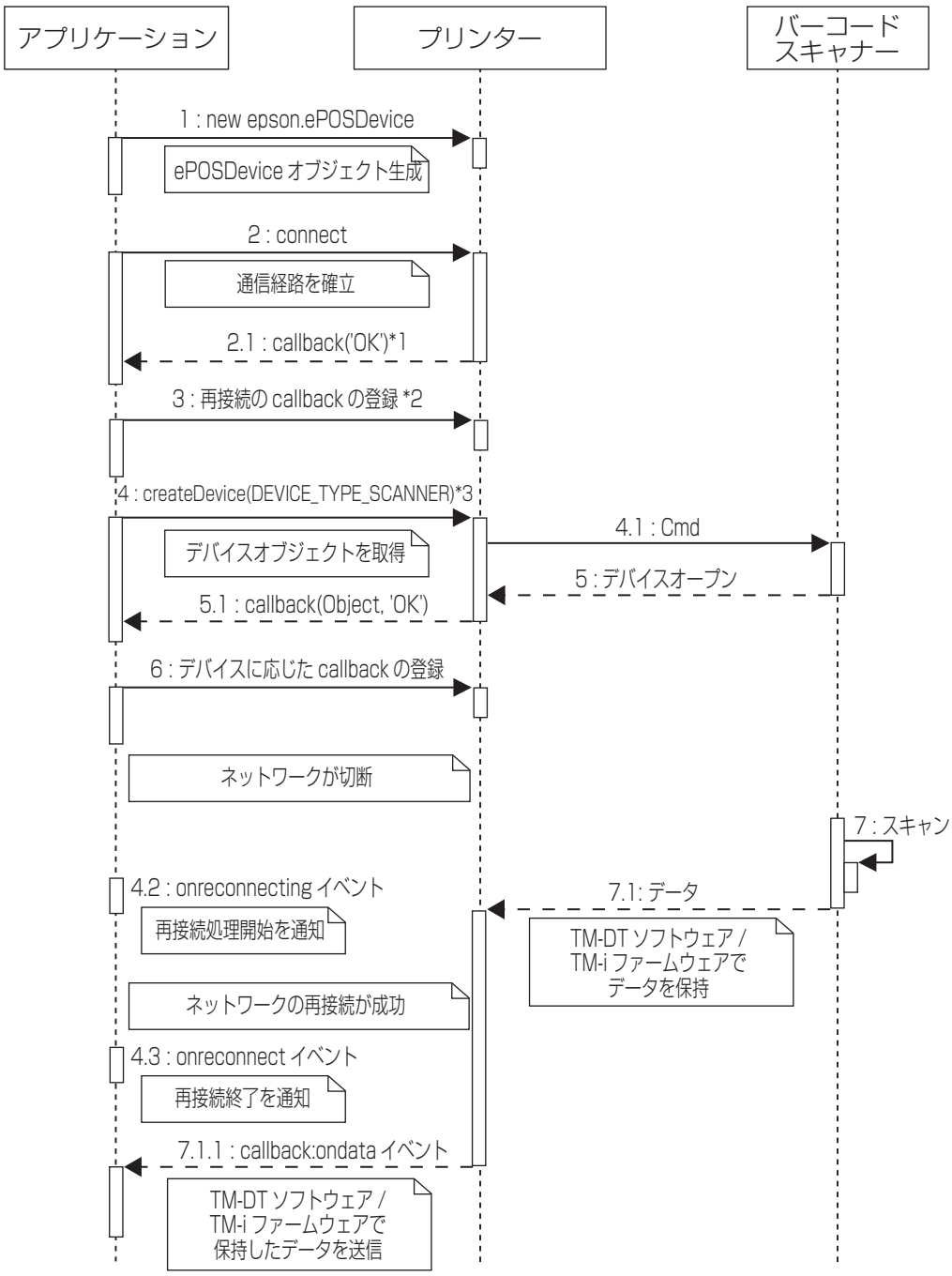
```
var ePosDev = new epson.ePOSDevice();
var printer = null;

function connect(){
   // デバイスと接続
   ePosDev.connect('192.168.192.168', '8008', callback_connect);
}

function callback_connect(resultConnect){
   if ((resultConnect == 'OK') || (resultConnect == 'SSL_CONNECT_OK')) {
      //Printer オブジェクトを取得する
      ePosDev.createDevice('local_printer', ePosDev.DEVICE_TYPE_PRINTER, {'crypto' : false,
'buffer' : false}, callback_createDevice);
   }
   else {
      // エラーメッセージ表示
   }
}

function callback_createDevice(deviceObj, retcode){
   printer = deviceObj;
   if( retcode == 'OK' ) {
      printer.timeout = 60000;
   // イベントを登録
      printer.onstatuschange = function (res) { alert(res.success); };
      printer.onbatterystatuschange = function (res) { alert(res.success); };
      print();
   } else {
      alert(retcode);
   }
}

function startMonitor(){
   // ステータス監視の開始
   printer.startMonitor();
}

// プリンターのカバーを開ける

function stopMonitor(){
   // ステータス監視の終了
   printer.stopMonitor();
}

function disconnect(){
   //Printer オブジェクトを破棄する
   ePosDev.deleteDevice(printer, callback_deleteDevice);
}

function callback_deleteDevice(errorCode){
   // デバイスと切断
   ePosDev.disconnect();
}
```

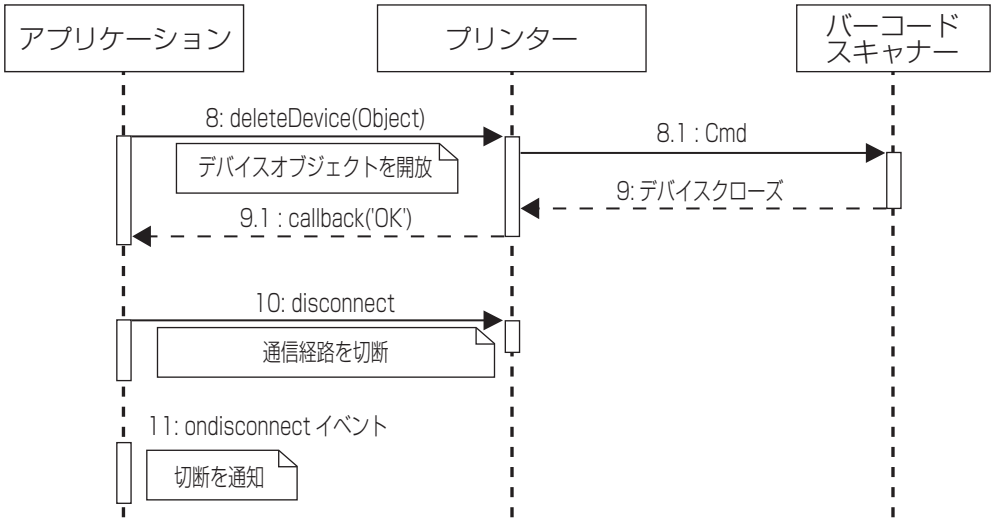
「対処メッセージ表示」に表示するメッセージは、onreceive イベントで取得するエラーコードと対処方法を参照してください。

ネットワークを自動で再接続するには

プリンターとのネットワーク通信が切断された場合、自動で再接続するシーケンスは、以下の通りです。

以下はバーコードスキャナー使用時を例にしています。





*1:"SSL_CONNECT_OK"（SSL/TLS 通信）

*2: 再接続処理開始イベント / 再接続終了イベント / 再接続失敗イベントが含まれます。

*3:options パラメーターを、ネットワークの再接続時にデータを再送する設定にします。

アプリケーション間でデータを送受信するには

TM-i シリーズ /TM-DT シリーズを使用すると、コミュニケーションボックス機能を使用して、簡単なプログラムでアプリケーション間でのデータの送受信が可能です。送受信できるデータはテキストデータ、イメージデータです。

コミュニケーションボックスを使用したアプリケーション例

スマートデバイスのカスタマーディスプレイ化

プリンターに接続したバーコードスキャナーでデータを読み込み、カスタマーディスプレイ化したスマートデバイスへ送信します。

データを受信したスマートデバイスは、イメージデータとともに受信データを表示します。

スマートデバイスを使った受付端末システム

お客様のスマートデバイスで入力した予約を店舗のスマートデバイスで受け付け、準備完了後に呼び出しメールをお客様のスマートデバイスへ送信します。

オーダーエントリーシステムとキッチンディスプレイ

店員がスマートデバイスを使って注文した料理をキッチンのスマートデバイス（キッチンディスプレイ）で管理し、調理完了後に配膳依頼を店員のスマートデバイスへ送信します。

デジタルサイネージ（電子看板）

店舗のメインスマートデバイスから、各コーナーのスマートデバイスへ適切な「お知らせ」を送信します。

周辺機器の使用権管理

TM-DT シリーズに接続した周辺機器を複数のスマートデバイスから使用する場合、周辺機器の使用が競合する可能性があります。コミュニケーションボックスを利用すると、この問題を解決できます。

例：

周辺機器の使用権を取得したスマートデバイスが、使用権の取得を他のスマートデバイスへ一斉送信します。

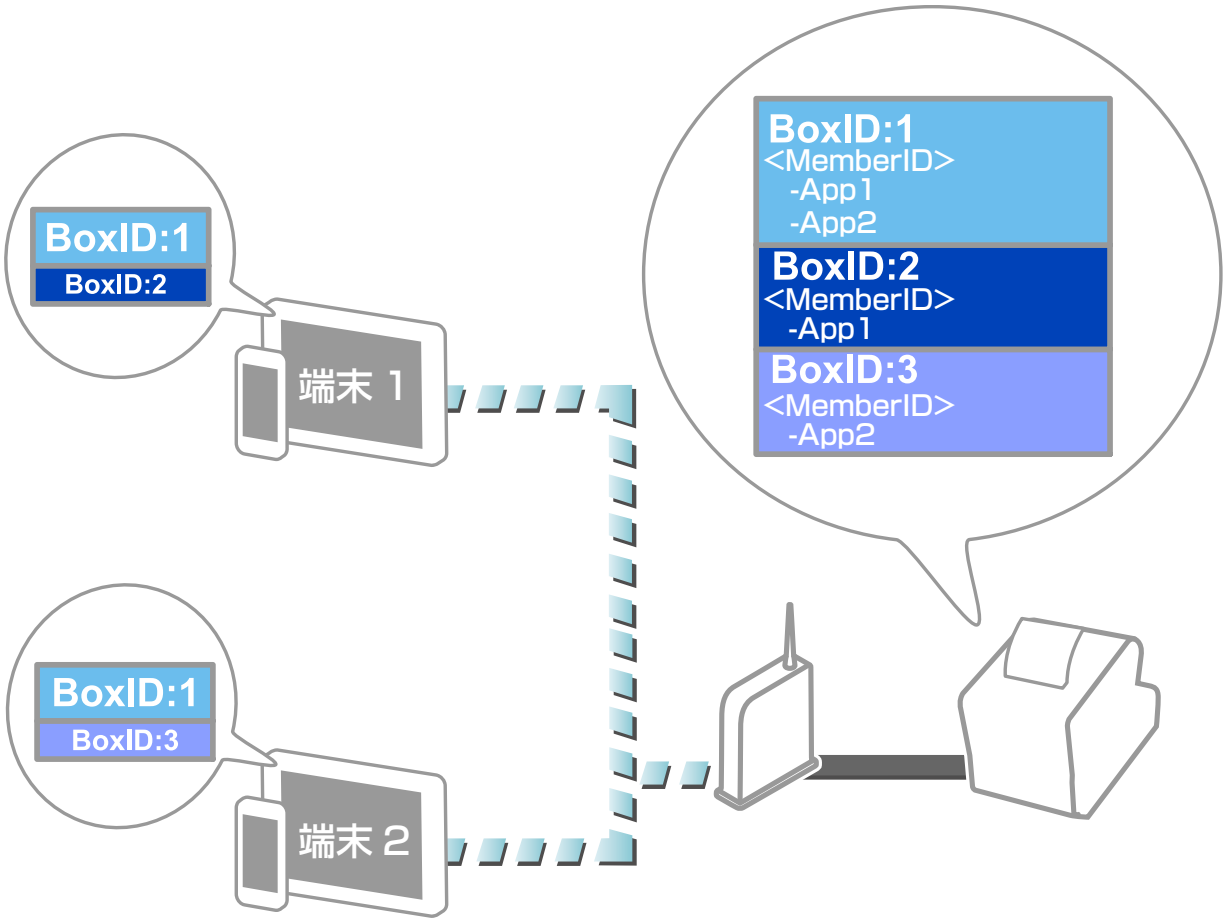
周辺機器を使い終わったら、使用権の破棄を他のスマートデバイスへ一斉送信することで、次のスマートデバイスが使用権を取得して周辺機器を使用できます。

コミュニケーションボックスの仕組み

コミュニケーションボックスはボックス ID で管理します。

コミュニケーションボックスに所属しているアプリケーション間でデータの送受信が可能です。

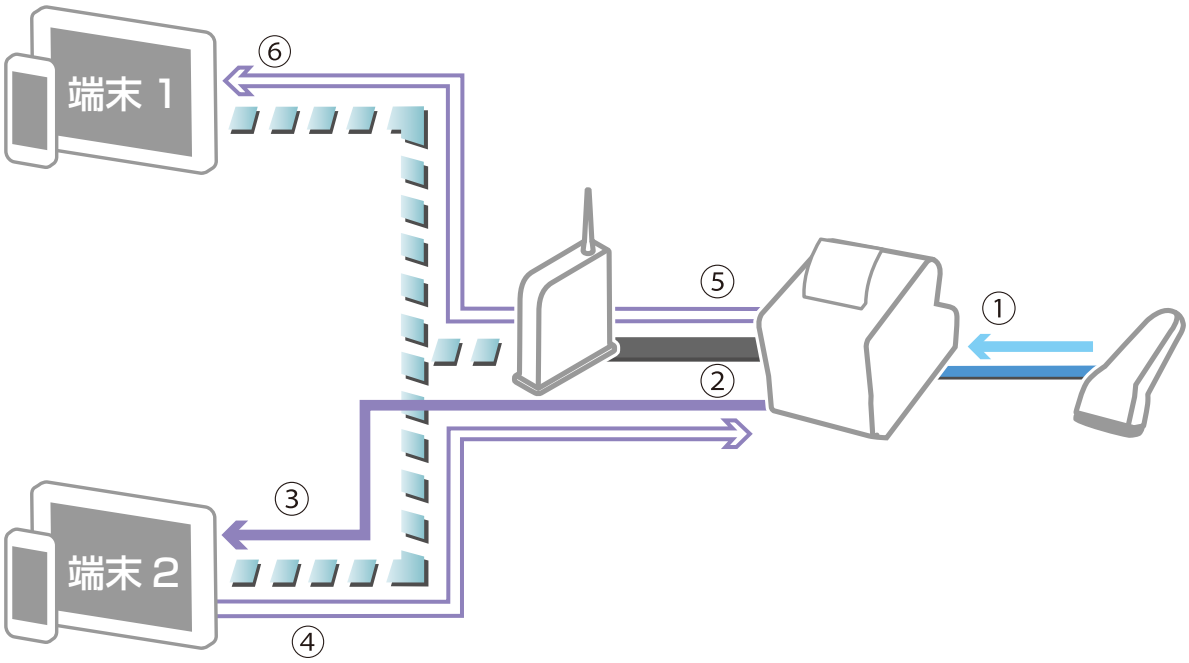
下図の場合、端末 1 と端末 2 のアプリケーションは、ボックス ID「BoxID:1」のコミュニケーションボックスを使って、アプリケーション間でデータの送受信ができます。



コミュニケーションボックスの仕様

機能	仕様
作成できるコミュニケーションボックスの最大数	20 個
1 つのコミュニケーションボックスに所属できるアプリケーションの最大数	20 個
1 つのコミュニケーションボックスが保持できる送信履歴の容量	10240 バイト
一度に送信できるデータサイズ	1024 バイト

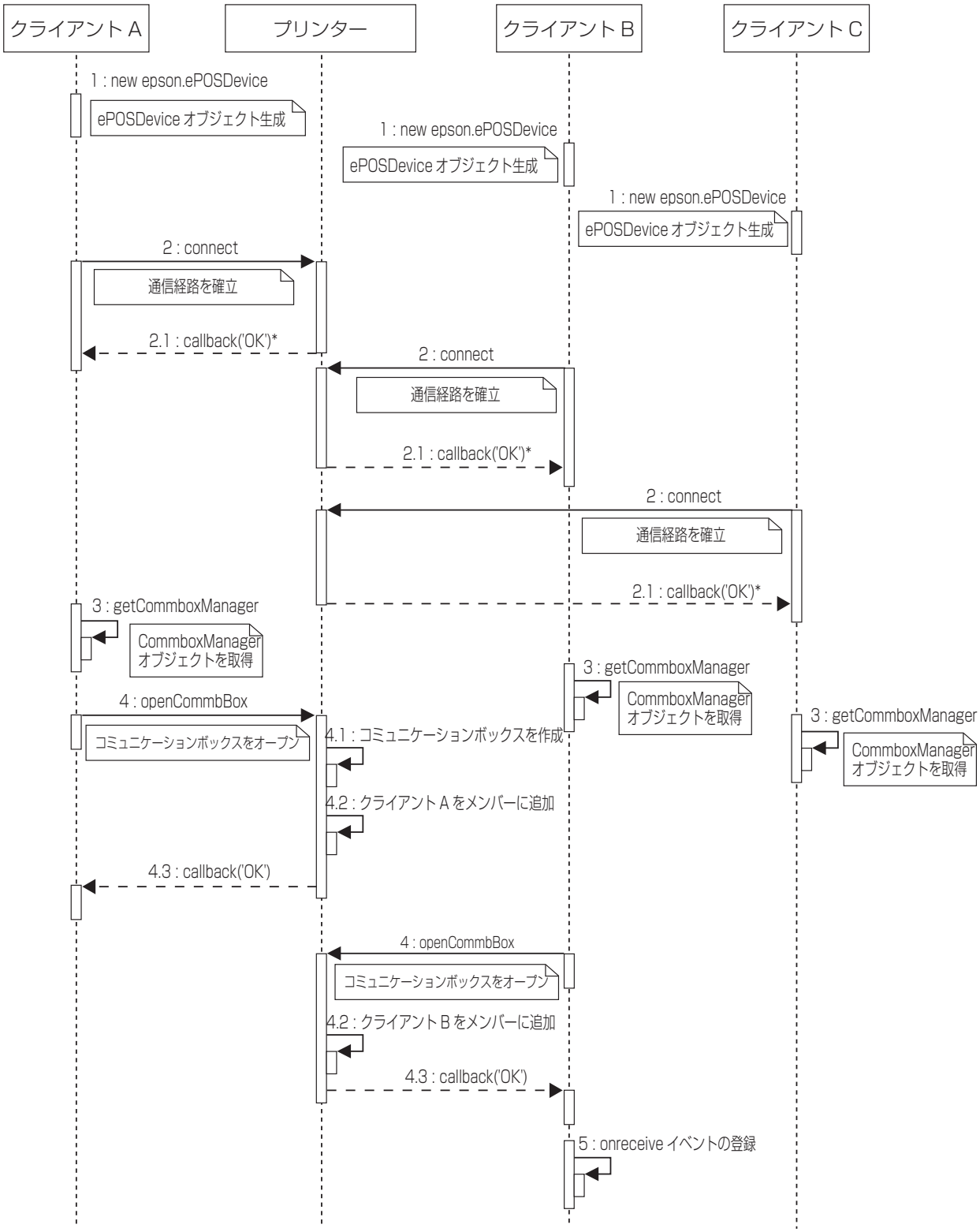
コミュニケーションボックスの使用したデータ処理例

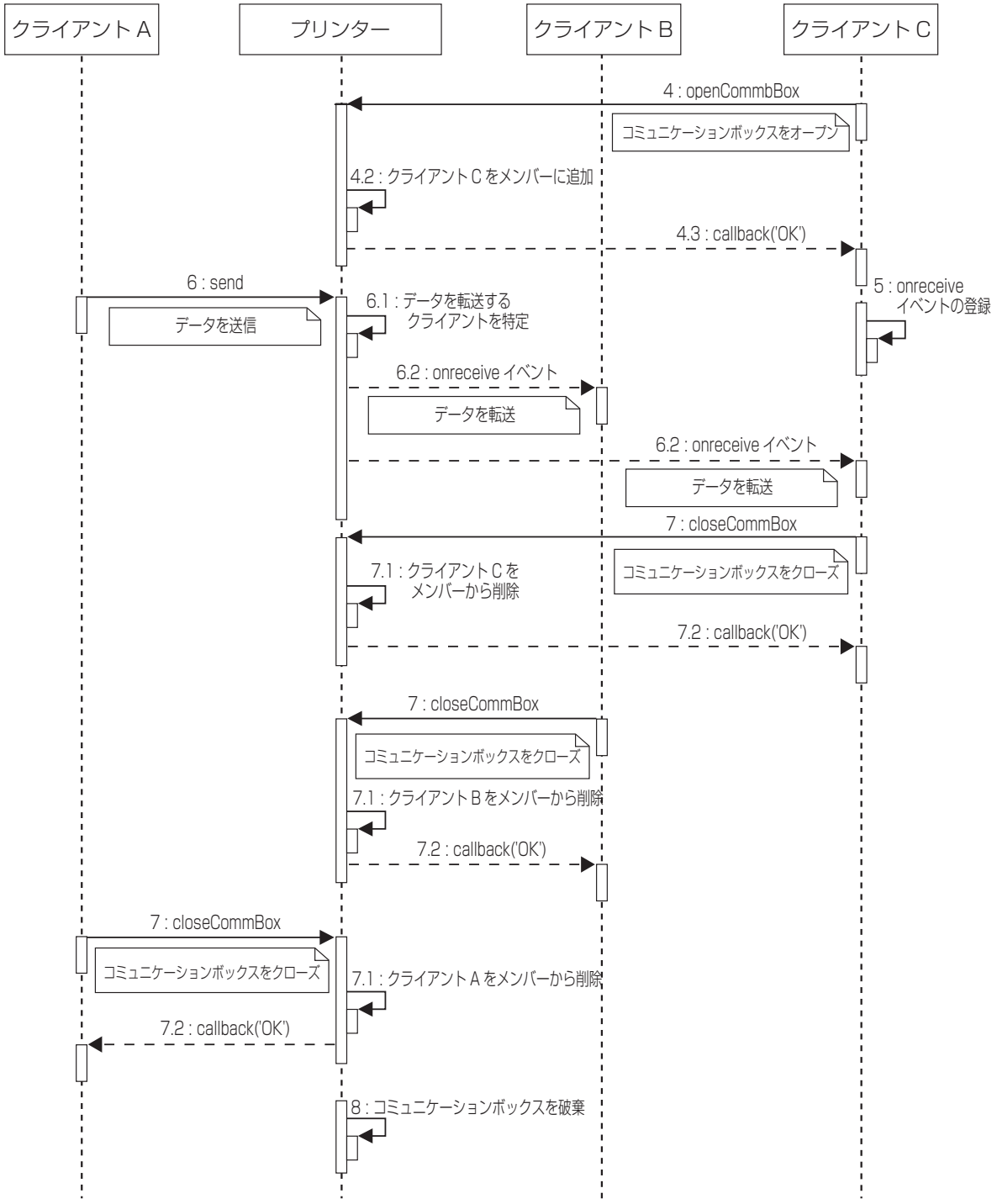


1. バーコードスキャナーが読み取ったバーコードデータを、プリンターが受信します。
2. プリンターは、バーコードデータを端末２に通知します。
3. 端末２はバーコードデータを取得し、POS データに変換します。
4. 端末２は、コミュニケーションボックスに POS データを送信します。
5. POS データが格納されたことを端末 1 に通知します。
6. 端末 1 は、コミュニケーションボックスに格納された POS データを取得します。

基本的なプログラミングシーケンス

シーケンス図中の「クライアント」は、アプリケーションを指しています。





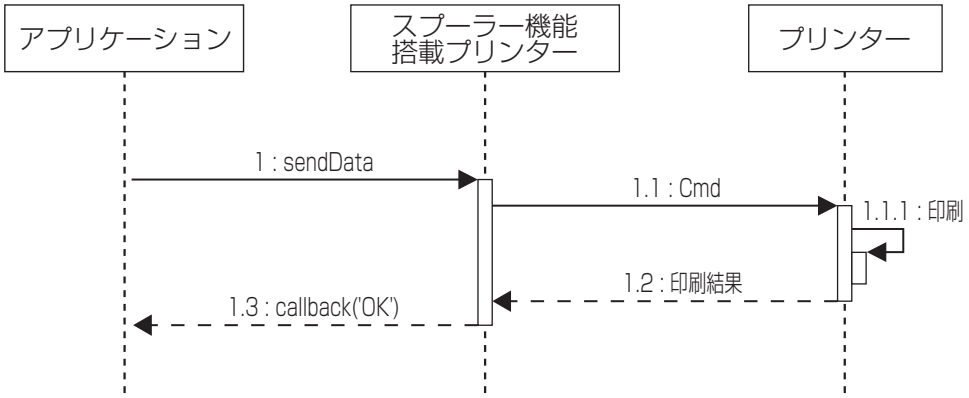
*:"SSL_CONNECT_OK"（SSL/TLS 通信）

迂回印刷するには

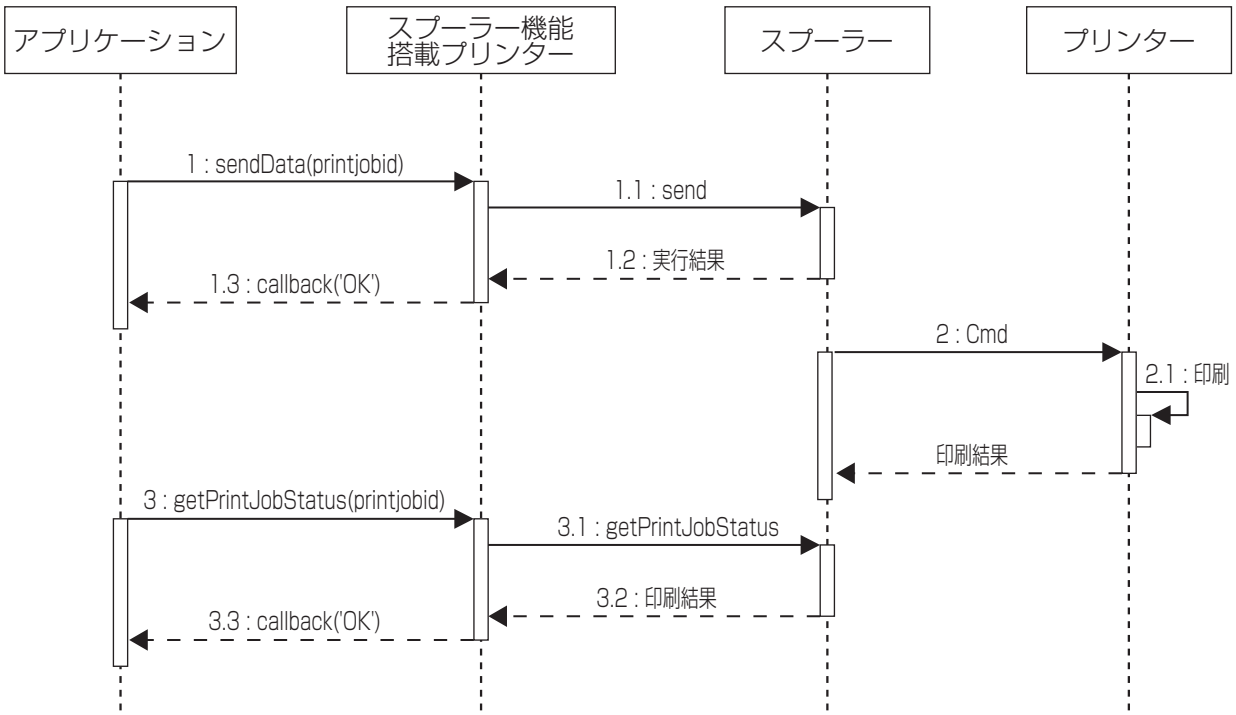
プリンターのスプーラー機能を使用します。

スプーラー機能を搭載したプリンターは、プリンターごとの提供機能を参照してください。

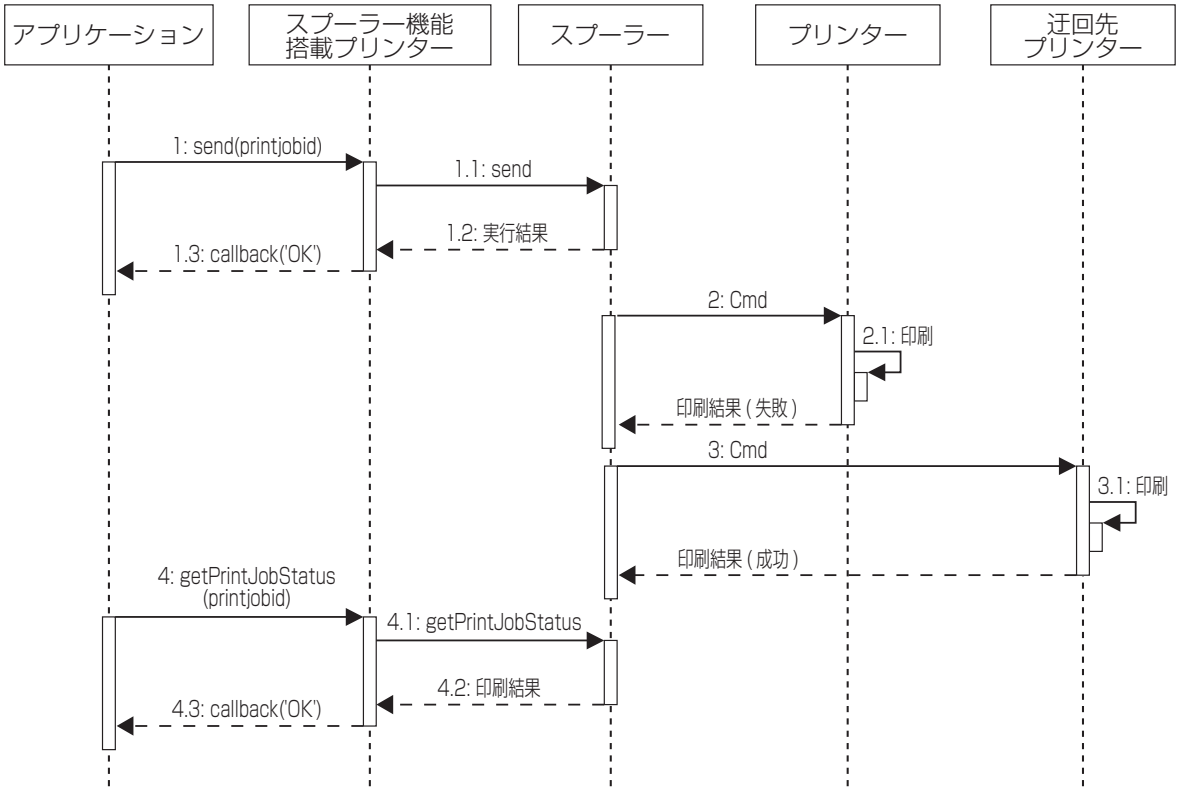
スプーラー機能を無効にした場合、アプリケーションからスプーラー機能搭載プリンターにリクエストを送信すると、印刷を直ちに実行し、印刷完了後にレスポンスをアプリケーションに返します。



スプーラー機能を有効にした場合、アプリケーションからスプーラー機能搭載プリンターにリクエストを送信すると、印刷データをキューに入れ、印刷完了を待たずにレスポンスをアプリケーションに返します。



この時、出力先のプリンターが印刷できない場合、スプーラー機能搭載プリンターは、アプリケーションにエラーを返しません。迂回印刷を有効にすると、代替プリンターで印刷でき、アプリケーションは、印刷結果も後で取得できます。以下のシーケンス図を参照してください。



❑ プログラミング例

スプーラー機能を有効にすると、印刷ジョブ ID を持つレスポンスを返します。

アプリケーションが印刷ジョブ ID を指定しない場合は、スプーラー機能搭載プリンターが印刷ジョブ ID を発行します。

印刷ジョブの状態は、レスポンスに含まれる印刷ジョブ ID で取得します。

```
var printjobid = '';

function sendJob() {
   printer.addText('Hello, World!\n');
   printer.addCut();
   printer.onreceive = function (res) {
      if (res.success) {
         printjobid = res.printjobid;
      }
   }
   printer.onerror = function (err) { alert(err.status); };
   printer.send();
}

function getJobStatus() {
   if (printjobid.length > 0) {
      printer.onreceive = function (res) { alert(res.success); };
      printer.onerror = function (err) { alert(err.status); };
      printer.getPrintJobStatus(printjobid);
   }
}
```

Webページを頻繁に更新/遷移するアプリケーションを開発するには

Epson ePOS SDK for JavaScript で開発したアプリケーションは、Web ページを更新 / 遷移する度に接続デバイスとの通信を切断し、再接続する処理が発生します。このため印字開始やディスプレイ表示が遅くなることがあります。
ePOS-Print オプションを有効にすることで、接続処理にかかる時間を短縮できます。

ePOS-Print オプションを有効にするには

ePOS-Print オプションは、connect メソッドの option パラメーターで設定します。

例）ePOS-Print オプションを有効にする。

```
connect("192.168.192.168", "8008", callback, {"eposprint" : true});
```

制限事項

- ❏ ePOS-Print オプションは、単機能モデルとモバイルモデルの制御に影響しません。
- ❏ ePOS-Print オプション有効下では、子機プリンターとカスタマーディスプレイ以外の周辺機器を制御できません。

> ✎ ePOS-Print オプション有効下で使用できるオブジェクトは、以下のとおりです。（参考：サポートオブジェクト一覧）
> - ePOSDevice オブジェクト
> - Display オブジェクト
> - Printer オブジェクト

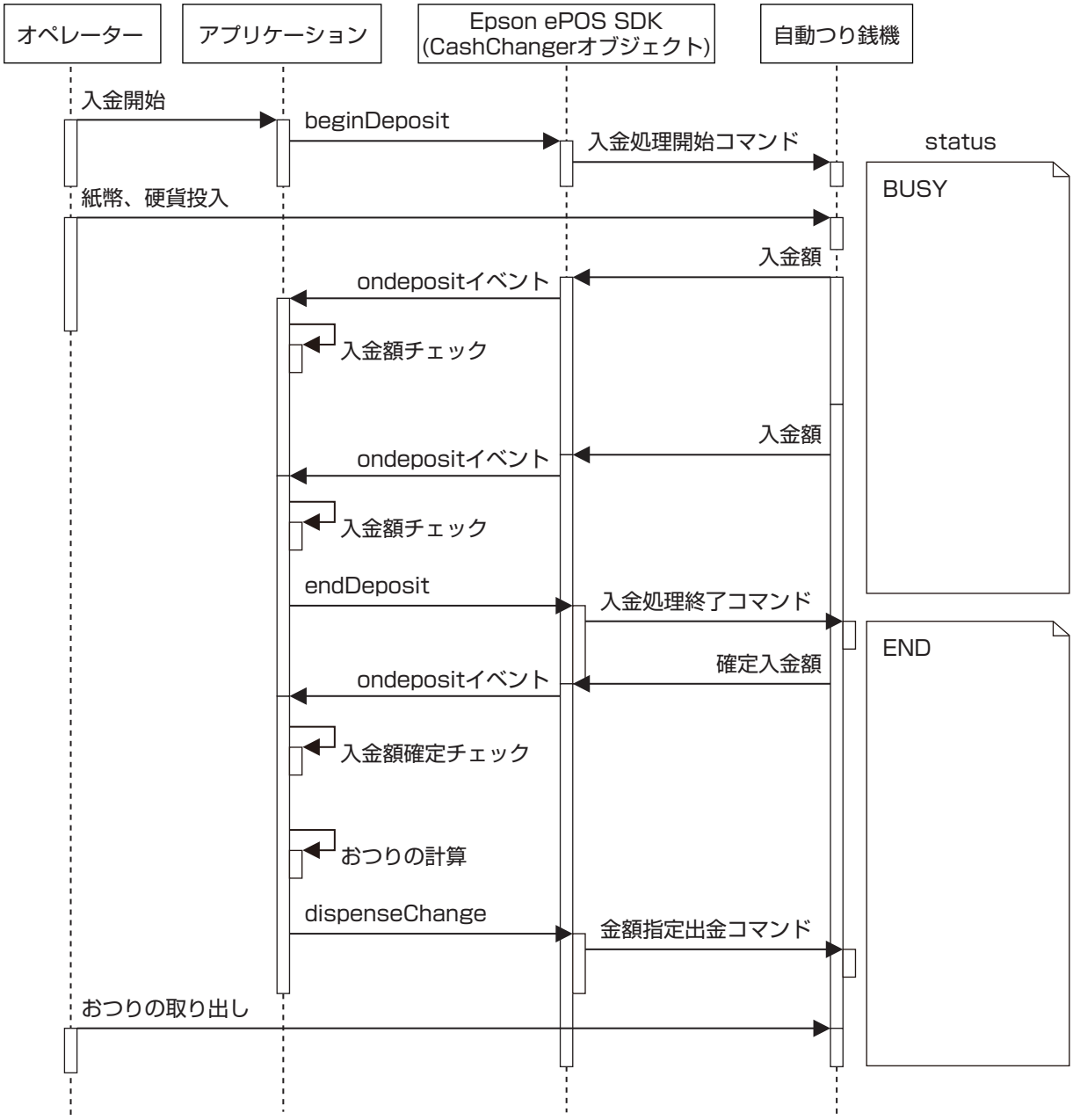
- ❏ TM プリンター +DM-D+ バーコードスキャナーモデルと POS ターミナルモデルのシステム構成では、ePOS-Print オプションを有効にしないでください。
  子機プリンターとカスタマーディスプレイ以外の周辺機器が制御できなくなるおそれがあります。

自動つり銭機を制御するには

CashChanger オブジェクトで自動つり銭機を制御します。
デバイス制御プログラムを使う場合と、デバイス制御スクリプトを使う場合で制御シーケンスが異なります。

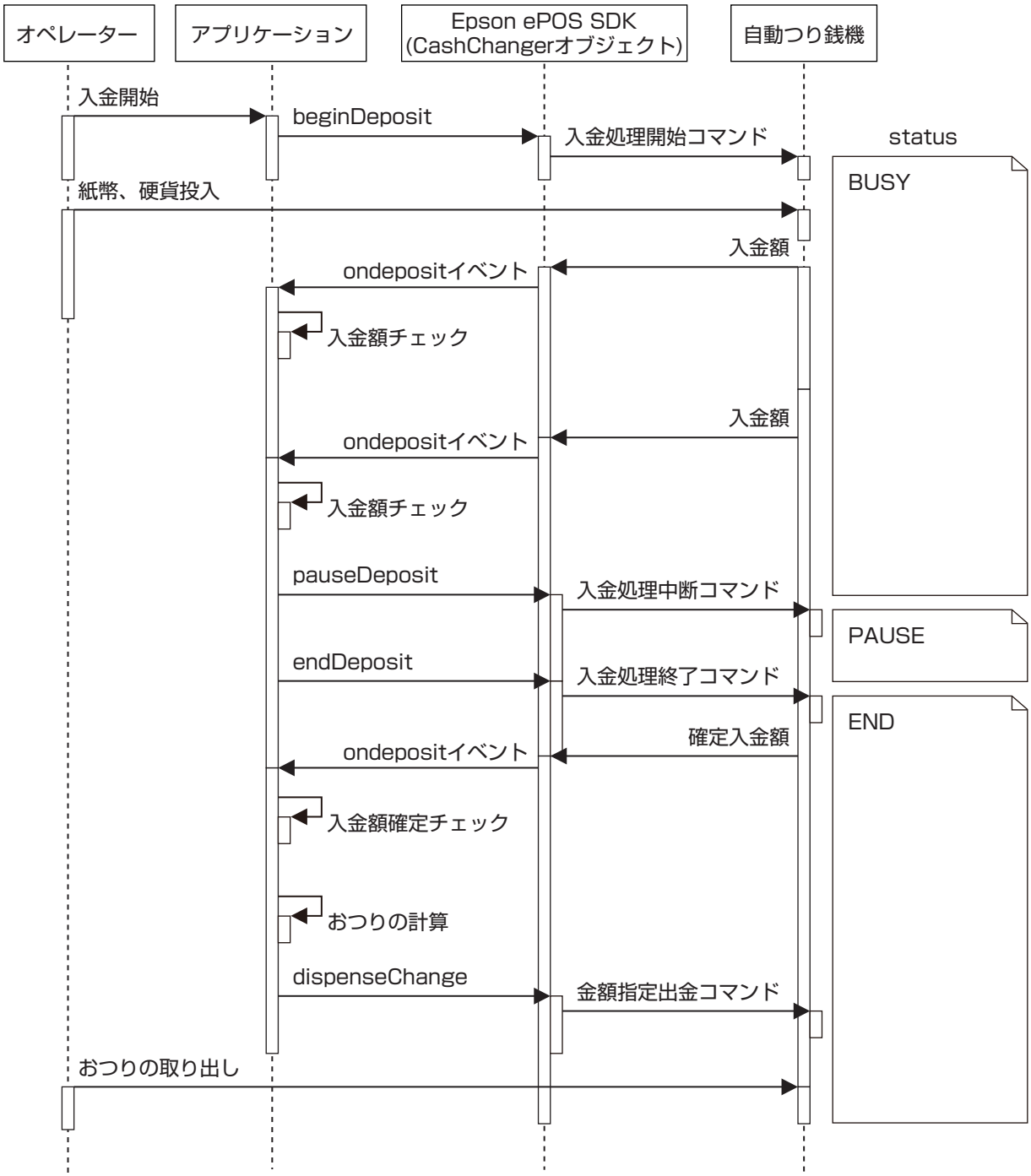
デバイス制御プログラム

- 紙幣や硬貨が投入されるたびに、入金額を通知する実行結果通知イベント（ondeposit イベント）が通知されます。
- 自動つり銭機内の処理によって、紙幣や硬貨の投入後に実行結果通知イベント（ondeposit イベント）が複数回通知されることがあります。
- 入金額のチェックは、入金額が精算金額を上回るまで繰り返してください。
- 入金額確定チェック時は、自動つり銭機の status が END であることを確認してください。



デバイス制御スクリプト

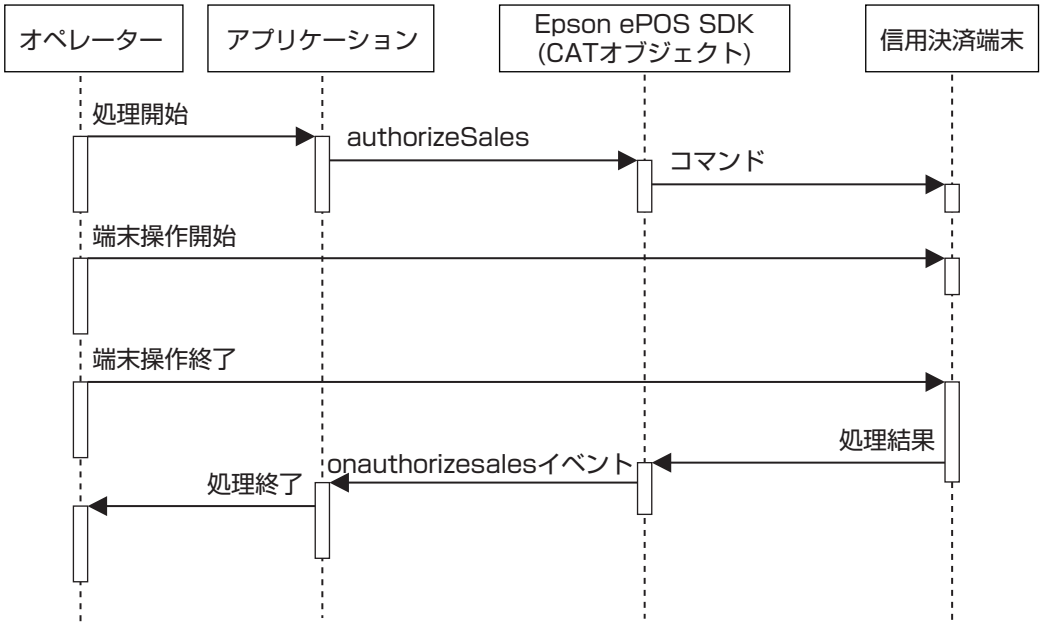
- 紙幣や硬貨が投入されるたびに、入金額を通知する ondeposit イベントが通知されます。
- 自動つり銭機内の処理によって、紙幣や硬貨の投入後に ondeposit イベントが複数回通知されることがあります。
- 入金額のチェックは、入金額が精算金額を上回るまで繰り返してください。
- 入金額確定チェック時は、自動つり銭機の status が END であることを確認してください。



信用決済端末を制御するには

以下に CAT オブジェクトで信用決済端末を制御するシーケンス図を示します。

以下のシーケンス図では、売上処理をする場合の制御シーケンスを例にしています。



付録

オープンソースソフトウェアライセンス

Epson ePOS SDK for JavaScript が提供するサンプルプログラムとライブラリーには、以下のオープンソースソフトウェアを使用しています。

MIT Licensed

```
/**
 * @license
 * zlib.js
 * JavaScript Zlib Library
 * https://github.com/imaya/zlib.js
 *
 * The MIT License
 *
 * Copyright (c) 2012 imaya
 *
 * Permission is hereby granted, free of charge, to any person obtaining a copy
 * of this software and associated documentation files (the "Software"), to deal
 * in the Software without restriction, including without limitation the rights
 * to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
 * copies of the Software, and to permit persons to whom the Software is
 * furnished to do so, subject to the following conditions:
 *
 * The above copyright notice and this permission notice shall be included in
 * all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 * IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
 * FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
 * AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
 * LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
 * OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
 * THE SOFTWARE.
 */
```

BSD Licensed

Copyright (c) 2008-2018 Brian Turek, 1998-2009 Paul Johnston & Contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.