



SII Print Class Library for Android™ アプリケーションプログラマーズガイド

Rev.01

[対象製品]

DSP-A01シリーズ

セイコーインスツル株式会社

Rev.01

2019年 7月

©セイコーインスツル株式会社 2019

無断転載を禁じます。

Android™は、Google LLC.の商標です。

Bluetooth® は、Bluetooth SIG, Inc.の登録商標です。

OracleとJavaは、Oracle Corporation 及びその子会社、関連会社の米国及びその他の国における登録商標です。

文中の社名、商品名等は各社の商標または登録商標である場合があります。

本書の内容は、断りなく変更することがあります。

本書及び本書に記載された製品の利用によって発生した
損害及びその回復に要する費用に対し、当社は一切の責任を負いかねます。

本書を第三者に無断で頒布することを禁じます。

はじめに

本書は、セイコーインスツル株式会社(以降:SII)が提供する「SII Print Class Library for Android™」(以降:SIIプリントクラスライブラリ)について説明します。

対象製品

本書の対象製品について以下に記載します。

製品	インターフェイス	本書での記載
DSP-A01-1	USB	ディスプレイ

対応ドライバー

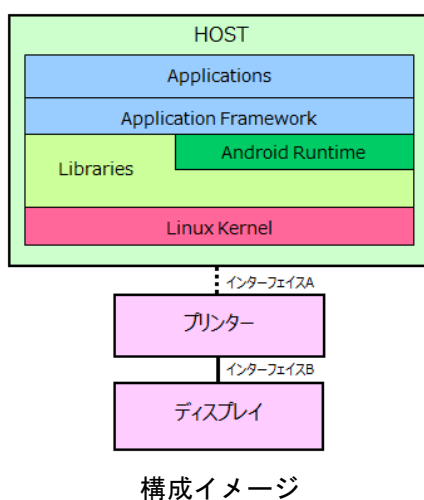
ディスプレイの利用にあたり、SII製ドライバーが必要です。

対応するSII製ドライバーについて以下に記載します。

ディスプレイの接続方法は2種類あります。

(1) ディスプレイを SII 製プリンターに接続して使用する場合（以降、“プリンター経由で使用”と記載）

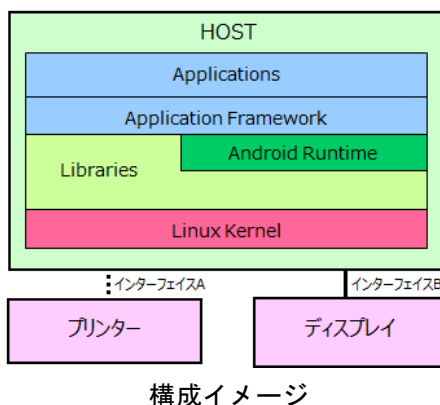
SII製ドライバー	インターフェイスA	プリンター	インターフェイスB
RP-F10/G10シリーズ用 「SII Printer Driver for Windows」	Bluetooth TCP/IP USB	RP-F10シリーズ	USB



(2) ディスプレイを単体で使用する場合（以降、“ディスプレイ単体で使用”と記載）

SII製ドライバー	インターフェイスA	プリンター*1	インターフェイスB
DSP-A01シリーズ用 「SII Printer Driver for Windows」 RP-F10/G10シリーズ用 「SII Printer Driver for Windows」	-	-	USB

*1: ディスプレイをサポートする SII 製プリンターは以下の通りです。
・ RP-F10 シリーズ



用語一覧

本書で使用する用語について以下の通りに定義します。

プリンター

用語	内容
プリンター用技術説明書	下記の技術説明書。 ・RP-F10シリーズ サーマルプリンタ技術説明書
プリンターコマンド	「プリンター用技術説明書」に記載されている、プリンターを制御するための命令。

ディスプレイ

用語	内容
ディスプレイ用技術説明書	下記の技術説明書。 ・DSP-A01シリーズ カスタマーディスプレイ技術説明書
ディスプレイコマンド	「ディスプレイ用技術説明書」に記載されている、ディスプレイを制御するための命令。
スライド	画面サイズ(横480px×縦272px)のイメージデータ。 スタンバイ画面としてディスプレイへの表示、及び背景画面としてテンプレートと重ね合わせてディスプレイに表示する。
テンプレート	描画領域や配置位置などの属性を設定できる要素を持つ定型フォーム。要素にはtext要素(テキストデータ)、img要素(イメージデータ)、barcode要素(バーコードデータ)、及びqr要素(QRコードデータ)がある。 テンプレートを登録する場合、配置する要素にそれぞれマップIDを定義する。 登録済みのイメージデータやテキストデータ等は、テンプレートを選択し、そのマップIDを指定し、画面更新することでディスプレイに表示ができる。 テンプレートのデータはXMLファイル形式で指定する必要がある。 登録できるテンプレートの最大サイズは横480px×縦272px。
マップID	テンプレートを登録する際に位置や修飾情報等を保持した要素に定義するID。
マクロ	複数のAPIを実行する順に登録し、イベントが発生した際に自動的に実行される機能。
イベント	ディスプレイコマンド「イベント通知」によって定義されたイベント。

目次

1章	製品概要	1-1
1.1	SIIプリントクラスライブラリの提供機能	1-1
1.2	SIIプリントクラスライブラリの概要	1-1
1.2.1	SIIプリントクラスライブラリの構成	1-1
1.2.2	ライブラリの提供機能	1-2
1.2.3	ディスプレイの工場出荷時の登録データ	1-2
2章	製品仕様	2-1
2.1	動作環境	2-1
3章	ライブラリの利用方法	3-1
3.1	Androidアプリケーションの開発環境	3-1
3.2	提供ファイル	3-2
3.3	Android Studioのプロジェクトへの組み込み	3-3
3.4	作成したAndroidアプリケーションのAndroidデバイスでの利用	3-5
4章	ライブラリの機能	4-1
4.1	APIリファレンス	4-1
4.1.1	PrinterManagerクラス	4-2
PrinterManager	コンストラクタ	4-10
connect	プリンターとの通信開始 (Bluetooth)	4-10
connect	デバイスとの通信開始 (USB)	4-11
connect	プリンターとの通信開始 (TCP/IP)	4-11
disconnect	デバイスとの通信切断	4-12
sendBinary	バイナリデータの送信	4-12
sendDataFile	指定ファイルの送信	4-13
getPrinterResponse	プリンターからの各種応答取得	4-13
startDiscoveryPrinter		
	プリンターの探索開始 (Bluetooth)	4-14
startDiscoveryPrinter		
	プリンターの探索開始 (USB)	4-15
startDiscoveryPrinter		
	プリンターの探索開始 (TCP/IP)	4-15
startDiscoveryDevice		
	デバイスの探索開始 (USB)	4-16
cancelDiscoveryPrinter		
	プリンターの探索中断	4-16
getFoundPrinter	発見されたプリンター情報リストの取得	4-16
getFoundDevice	発見されたデバイス情報リストの取得	4-17

getSendTimeout	送信タイムアウト時間の取得	4-17
setSendTimeout	送信タイムアウト時間の設定	4-17
getReceiveTimeout	受信タイムアウト時間の取得	4-17
setReceiveTimeout	受信タイムアウト時間の設定	4-18
getInternationalCharacter	国際文字セットの取得	4-18
setInternationalCharacter	国際文字セットの設定	4-18
getCodePage	コードページの取得	4-19
setCodePage	コードページの設定	4-19
getPrinterModel	デバイスモデルの取得	4-19
getPortType	接続中ポート種別の取得	4-19
isConnect	デバイスとの通信状態の確認	4-20
getSocketKeepingTime	ソケット維持時間の取得	4-20
setSocketKeepingTime	ソケット維持時間の設定	4-20
defragment	メモリ領域の最適化	4-21
initializeMemoryArea	メモリ領域の初期化	4-21
showTemplate	テンプレートの表示	4-22
showSlide	スライドの表示	4-23
enterStandbyMode	スタンバイの表示	4-24
executeMacro	マクロの実行	4-24
turnOnScreen	画面の表示オン・オフ	4-25
selectTemplate	テンプレートの選択	4-25
setTemplateImageData	イメージデータの設定	4-26
selectTemplateTextObject	text要素の選択	4-27
setTemplateTextAlignment	テキストデータの位置揃え	4-27
setTemplateTextLeftMargin	テキストデータの左マージンの設定	4-28
setTemplateTextLineSpacing	テキストデータの改行量の設定	4-28
setTemplateTextBold	テキストデータの強調文字の設定	4-29
setTemplateTextUnderline	テキストデータのアンダーラインの設定	4-30
setTemplateTextSize	テキストデータの文字サイズの設定	4-30
setTemplateTextFont	テキストデータの文字フォントの設定	4-31
setTemplateTextRegisteredFont	テキストデータの登録フォントの設定	4-31
setTemplateTextRightSpacing	テキストデータの右スペース量の設定	4-32
setTemplateTextColor	テキストデータの文字色の設定	4-33

setTemplateTextData	テキストデータの入力	4-33
setTemplateBarcodeData	バーコードデータの入力	4-34
setTemplateQrCodeData	QRコードデータの入力	4-35
registerTemplate	テンプレートの登録	4-36
unregisterTemplate	テンプレートの削除	4-37
registerImageData	イメージデータの登録	4-38
unregisterImageData	イメージデータの削除	4-39
registerSlideData	スライドデータの登録	4-39
unregisterSlideData	スライドデータの削除	4-40
registerUserDefinedCharacter	外字の登録	4-41
unregisterUserDefinedCharacter	外字の削除	4-41
registerOptionFont	オプションフォントの登録	4-42
unregisterOptionFont	オプションフォントの削除	4-43
controlMacroRegistration	マクロ登録の開始・終了	4-43
getDisplayResponse	ディスプレイからの各種応答取得	4-45
4.1.2 PrinterEventクラス		4-47
getEventType	終了イベントの取得	4-47
4.1.3 PrinterListenerインターフェイス		4-48
finishEvent	デバイス探索の終了イベント	4-48
4.1.4 PrinterInfoクラス		4-49
getDevicePath	デバイスパスの取得	4-49
4.1.5 PrinterExceptionクラス		4-50
PrinterException	コンストラクタ	4-51
getErrorCode	エラーコードの取得	4-51

5章 サンプルプログラム 5-1

5.1	インストール	5-1
5.2	画面構成	5-3
5.2.1	メイン画面	5-3
5.2.2	[SETTINGS]画面	5-4
5.3	注意事項	5-4

付録A 文字セット A-1

A.1	コードページ表(文字コードテーブル)	A-1
A.2	国際文字セット	A-11

1章 製品概要

本章では、SIIプリントクラスライブラリの製品概要について説明します。

1.1 SIIプリントクラスライブラリの提供機能

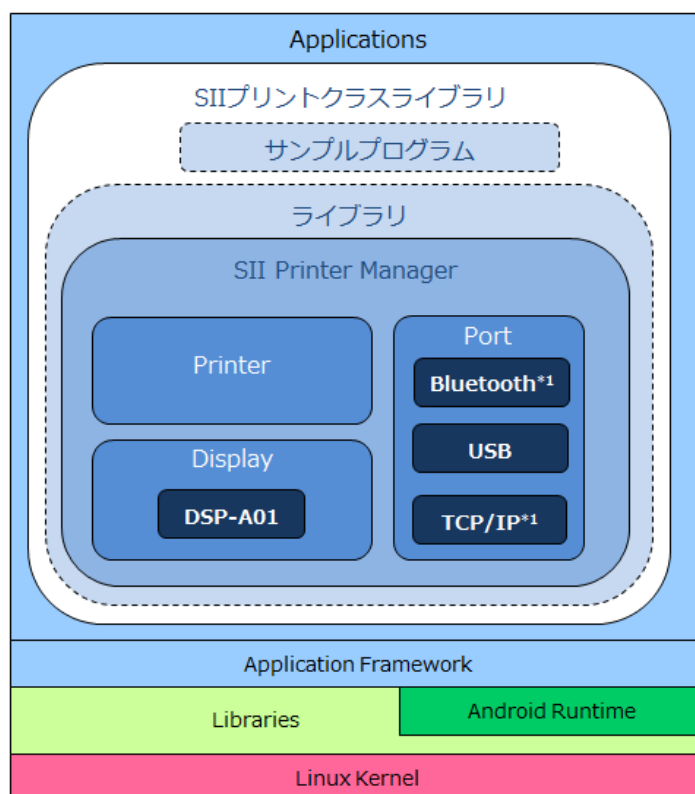
ライブラリとサンプルプログラムを含むSIIプリントクラスライブラリは、Androidアプリケーションに「はじめに 対象製品」に記載のディスプレイを利用するための機能を提供します。

また、SIIプリントクラスライブラリは、ライブラリのサンプルプログラムをAndroid Studioのプロジェクトで提供します。

1.2 SIIプリントクラスライブラリの概要

1.2.1 SIIプリントクラスライブラリの構成

SIIプリントクラスライブラリに含まれるライブラリとサンプルプログラムの範囲を、下図に破線で示します。



*1: プリンター経由で使用する場合のみ有効

1.2.2 ライブラリの提供機能

ライブラリを使用することにより、Androidアプリケーションは、Androidデバイスの通信ポート（Bluetooth^{*1}、USB、またはTCP/IP^{*1}）を通じて、デバイスに表示データやディスプレイコマンドを容易に送信することができます。また、ディスプレイの応答を取得することができます。

ライブラリでは、以下の機能を提供します。

- デバイスとの接続及び切断
- デバイスへのデータ送信（表示データやディスプレイコマンド^{*2}）
- デバイスへのデータファイル送信（表示データやディスプレイコマンド^{*2}）
- ディスプレイからの各種応答取得
- ディスプレイの表示制御

*1: プリンター経由で使用する場合のみ有効

*2: ディスプレイから応答を読み取るには、`getDisplayResponse` を使用してください。

1.2.3 ディスプレイの工場出荷時の登録データ

プリンターにディスプレイを接続し、テンプレートを使用する場合、ディスプレイに工場出荷時に登録されるデータは、品質向上のため予告無く追加・変更される場合があります。

また、指定する言語設定、及び文字コードによっては適切なエンコードが指定されているテンプレートを利用する必要があります。工場出荷時に登録されるデータの詳細については、弊社Webサイトを参照してください。

<http://www.sii-ps.com/dspa01/>

2章 製品仕様

本章では、ライブラリの製品仕様について説明します。

2.1 動作環境

ライブラリの動作環境を下記に示します。

ディスプレイ	モデル		DSP-A01
	通信インターフェイス		USB
Android デバイス	通信ポート		USB*1
	OS	Android 5.0 (API 21)	対応
		Android 5.1 (API 22)	
		Android 6.0 (API 23)	
		Android 7.0 (API 24)	
		Android 7.1 (API 25)	
		Android 8.0 (API 26)	
		Android 8.1 (API 27)	
		Android 9.0 (API 28)	
サポート言語		日本語/英語	

*1: AndroidデバイスがUSBホスト機能をサポートしている必要があります。

3章 ライブラリの利用方法

本章では、Androidアプリケーションの開発環境とライブラリの利用方法について説明します。

3.1 Androidアプリケーションの開発環境

Androidアプリケーションを開発するためには、以下のツールが必要です。詳細は各URLを参照してください。

- Android Studio
<https://developer.android.com/studio/index.html>
- Windows用USB ドライバ（Windows環境で開発する場合）
<https://developer.android.com/studio/run/oem-usb.html>

本書では、本章以降、各ツールを利用できる環境が整っていることを前提に説明します。

3.2 提供ファイル

Siiプリントクラスライブラリのファイル構成は、以下の通りです。

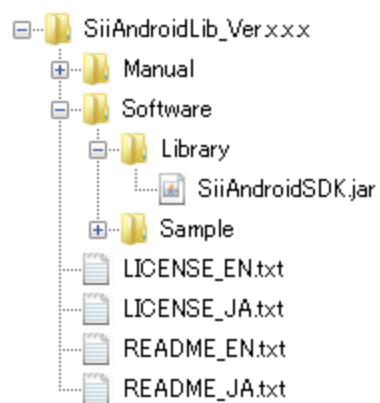


図 3-1

ライブラリは、JARファイル形式です。ライブラリのファイル名は、SiiAndroidSDK.jarです。

3.3 Android Studioのプロジェクトへの組み込み

SIIプリントクラスライブラリに含まれるサンプルプログラムのプロジェクトを例にして、ライブラリをAndroid Studioのプロジェクトに組み込む手順を説明します。

SIIプリントクラスライブラリに含まれるサンプルプログラムについては、「5章 サンプルプログラム」を参照してください。

- (1) Android StudioのAndroid Projectビューに表示されているモジュール(app)を選択した状態で右クリックし、[New] - [Directory]を選択してください。(図 3-2)
表示されるダイアログのフォルダ名に"libs"を入力して[OK]ボタンをクリックしてください。(図 3-3)

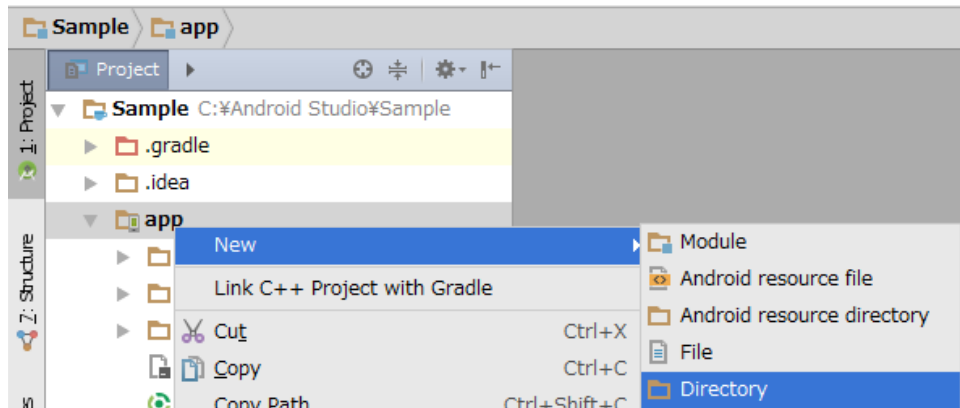


図 3-2

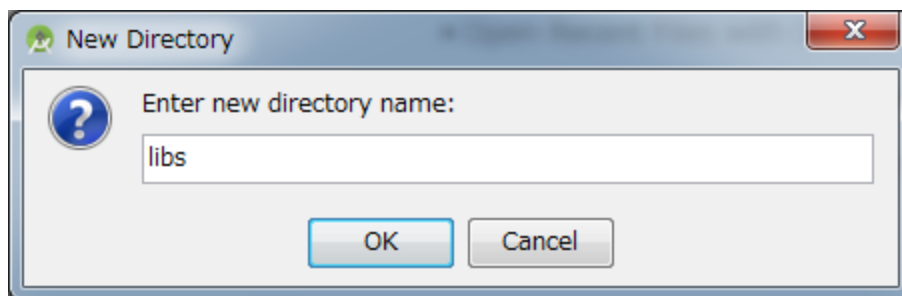


図 3-3

- (2) (1)で作成したフォルダ(¥Sample¥app¥libs)にライブラリファイル(SiiAndroidSDK.jar)をコピーしてください。

(3) ライブラリの追加が完了すると図 3-4の状態となります。

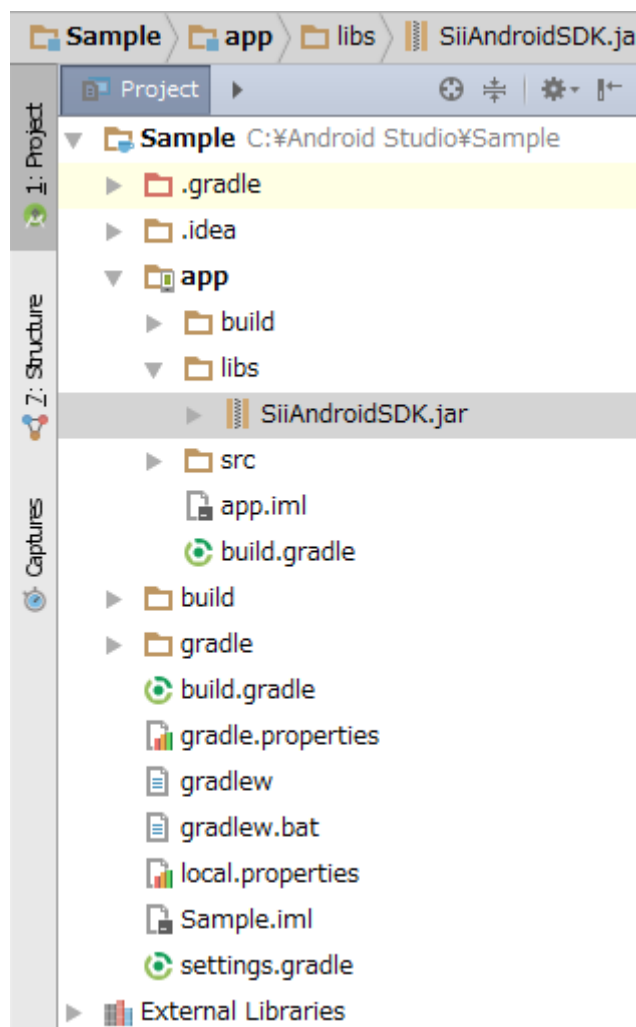


図 3-4

(4) メインとなるソースファイルの先頭に以下を追加してください。
(サンプルプログラムでは、MainActivity.javaの先頭に追加してください。)

```
import com.seikoinstruments.sdk.thermalprinter.PrinterEvent;  
import com.seikoinstruments.sdk.thermalprinter.PrinterException;  
import com.seikoinstruments.sdk.thermalprinter.PrinterInfo;  
import com.seikoinstruments.sdk.thermalprinter.PrinterListener;  
import com.seikoinstruments.sdk.thermalprinter.PrinterManager;
```

以上で、ライブラリの機能が利用可能になります。

3.4 作成したAndroidアプリケーションのAndroidデバイスでの利用

作成したAndroidアプリケーションをAndroidデバイス上で利用するためには、Androidデバイスで以下の設定を行ってください。

(注意) 本手順はAndroid 7.1のメニューを基に記載しています。ご利用のAndroidデバイスにより、メニュー内容が異なる場合があります。

- (1) [設定] - [セキュリティ] - [提供元不明のアプリ]をオンにしてください。(図 3-5)



図 3-5

- (2) [設定] - [開発者向けオプション] - [USBデバッグ]をオンにしてください。(図 3-6)

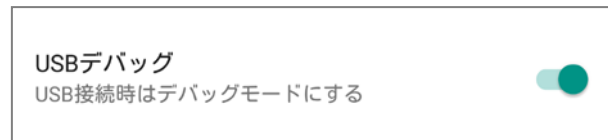


図 3-6

4章 ライブラリの機能

本章では、ライブラリに実装されている各クラスのAPIについて説明します。

4.1 APIリファレンス

ライブラリのパッケージは、`com.seikoinstruments.sdk.thermalprinter`です。
`com.seikoinstruments.sdk.thermalprinter`には、下記のクラスが含まれます。

クラス名	説明	サポート ^{*1}
PrinterManager	デバイスとの通信、表示を行うためのAPIを提供します。 「4.1.1 PrinterManagerクラス」を参照してください。	✓
PrinterEvent	startDiscoveryPrinter、またはstartDiscoveryDeviceが終了した際の終了イベントを取得するAPIを提供します。 「4.1.2 PrinterEventクラス」を参照してください。	✓
PrinterListener	startDiscoveryPrinter、またはstartDiscoveryDeviceが終了した際の終了イベントを取得するためのインターフェイスです。 「4.1.3 PrinterListenerインターフェイス」を参照してください。	✓
PrinterInfo	startDiscoveryPrinter、またはstartDiscoveryDeviceで発見されたデバイス情報を格納します。 「4.1.4 PrinterInfoクラス」を参照してください。	✓
PrinterException	API呼び出し時にスローされる例外クラスです。 「4.1.5 PrinterExceptionクラス」を参照してください。	✓

*1: ✓: サポート、-: DSP-A01は非サポート

4.1.1 PrinterManagerクラス

(1) メソッド一覧

PrinterManagerクラスで提供されるメソッドを下記に示します。

名前	説明	サポート*1
PrinterManager	コンストラクタ	✓
connect	プリンターとの通信開始(Bluetooth)	✓
connect	デバイスとの通信開始(USB)	✓
connect	プリンターとの通信開始(TCP/IP)	✓
disconnect	デバイスとの通信切断	✓
sendBinary	バイナリデータの送信	✓
sendDataFile	指定ファイルの送信	✓
getPrinterResponse	プリンターからの各種応答取得	✓
startDiscoveryPrinter	プリンターの探索開始(Bluetooth)	✓
startDiscoveryPrinter	プリンターの探索開始(USB)	✓
startDiscoveryPrinter	プリンターの探索開始(TCP/IP)	✓
startDiscoveryDevice	デバイスの探索開始(USB)	✓
cancelDiscoveryPrinter	プリンターの探索中断	-
getFoundPrinter	発見されたプリンター情報リストの取得	✓
getFoundDevice	発見されたデバイス情報リストの取得	✓
getSendTimeout	送信タイムアウト時間の取得	✓
setSendTimeout	送信タイムアウト時間の設定	✓
getReceiveTimeout	受信タイムアウト時間の取得	✓
setReceiveTimeout	受信タイムアウト時間の設定	✓
getInternationalCharacter	国際文字セットの取得	✓
setInternationalCharacter	国際文字セットの設定	✓
getCodePage	コードページの取得	✓
setCodePage	コードページの設定	✓
getPrinterModel	デバイスモデルの取得	✓
getPortType	接続ポートの種別取得	✓
isConnect	デバイスとの通信状態の確認	✓
getSocketKeepingTime	ソケット維持時間の取得	✓
setSocketKeepingTime	ソケット維持時間の設定	✓
defragment	メモリ領域の最適化	✓
initializeMemoryArea	メモリ領域の初期化	✓
showTemplate	テンプレートの表示	✓
showSlide	スライドの表示	✓
enterStandbyMode	スタンバイの表示	✓
executeMacro	マクロの実行	✓
turnOnScreen	画面の表示オン・オフ	✓

名前	説明	サポート ^{*1}
selectTemplate	テンプレートの選択	✓
setTemplateImageData	イメージデータの設定	✓
selectTemplateTextObject	text要素の選択	✓
setTemplateTextAlignment	テキストデータの位置揃え	✓
setTemplateTextLeftMargin	テキストデータの左マージンの設定	✓
setTemplateTextLineSpacing	テキストデータの改行量の設定	✓
setTemplateTextBold	テキストデータの強調文字の設定	✓
setTemplateTextUnderline	テキストデータのアンダーラインの設定	✓
setTemplateTextSize	テキストデータの文字サイズの設定	✓
setTemplateTextFont	テキストデータの文字フォントの設定	✓
setTemplateTextRegisteredFont	テキストデータの登録フォントの設定	✓
setTemplateTextRightSpacing	テキストデータの右スペース量の設定	✓
setTemplateTextColor	テキストデータの文字色の設定	✓
setTemplateTextData	テキストデータの入力	✓
setTemplateBarcodeData	バーコードデータの入力	✓
setTemplateQrCodeData	QRコードデータの入力	✓
registerTemplate	テンプレートの登録	✓
unregisterTemplate	テンプレートの削除	✓
registerImageData	イメージデータの登録	✓
unregisterImageData	イメージデータの削除	✓
registerSlideData	スライドデータの登録	✓
unregisterSlideData	スライドデータの削除	✓
registerUserDefinedCharacter	外字の登録	✓
unregisterUserDefinedCharacter	外字の削除	✓
registerOptionFont	オプションフォントの登録	✓
unregisterOptionFont	オプションフォントの削除	✓
controlMacroRegistration	マクロ登録の開始・終了	✓
getDisplayResponse	ディスプレイからの各種応答取得	✓

*1: ✓: サポート、-: DSP-A01は非サポート

(2) 定数一覧

① 国際文字セット

国際文字セットの設定/取得で利用する定数を下記に示します。

定数名	説明	値
COUNTRY_USA	アメリカ	0
COUNTRY_FRANCE	フランス	1
COUNTRY_GERMANY	ドイツ	2
COUNTRY_ENGLAND	イギリス	3
COUNTRY_DENMARK_1	デンマーク I	4
COUNTRY_SWEDEN	スウェーデン	5
COUNTRY_ITALY	イタリア	6
COUNTRY_SPAIN	スペイン I	7
COUNTRY_JAPAN	日本	8
COUNTRY_NORWAY	ノルウェー	9
COUNTRY_DENMARK_2	デンマーク II	10
COUNTRY_SPAIN_2	スペイン II	11
COUNTRY_LATIN_AMERICA	ラテンアメリカ	12
COUNTRY_ARABIA	アラビア	17

② コードページ

コードページの設定/取得で利用する定数を下記に示します。

定数名	説明	値
CODE_PAGE_437	USA, Standard Europe (Code Page437)	0
CODE_PAGE_KATAKANA	Katakana	1
CODE_PAGE_850	Multilingual (Code Page850)	2
CODE_PAGE_860	Portuguese (Code Page860)	3
CODE_PAGE_863	Canadian-French (Code Page863)	4
CODE_PAGE_865	Nordic (Code Page865)	5
CODE_PAGE_857 ^{*1}	Turkish (Code Page857)	13
CODE_PAGE_737	Greek (Code Page737)	14
CODE_PAGE_1252	Latin (Code Page1252)	16
CODE_PAGE_866	Russian (Code Page866)	17
CODE_PAGE_852	Eastern Europe (Code Page852)	18
CODE_PAGE_858	Euro (Code Page858)	19
CODE_PAGE_855	Cyrillic (Code Page855)	34
CODE_PAGE_864 ^{*1*2}	Arabic (Code Page864)	37
CODE_PAGE_1250	Central European (Code Page1250)	45
CODE_PAGE_1251	Cyrillic (Code Page1251)	46

定数名	説明	値
CODE_PAGE_1253 ^{*3}	Greek (Code Page1253)	47
CODE_PAGE_1254	Turkish (Code Page1254)	48

*1: Unicodeの20AChは表示できません。

*2: フォントBは表示できません。

*3 Unicodeの00AAhは表示できません。

③ デバイスモデル

デバイスとの通信開始と、デバイスモデル取得で利用する定数を下記に示します。

定数名	説明	値
PRINTER_MODEL_RP_FG10 ^{*1}	RP-F10	301
PRINTER_MODEL_DSP_A01	DSP-A01	303
PRINTER_MODEL_DEFAULT	デバイスモデルの初期値	284

*1: プリンター経由で使用している場合のみ有効

④ ポート種別

デバイスとの通信開始と、接続ポート種別取得で利用する定数を下記に示します。

定数名	説明	値
PRINTER_TYPE_BLUETOOTH ^{*1}	Bluetooth	0
PRINTER_TYPE_USB	USB	1
PRINTER_TYPE_TCP ^{*1}	TCP/IP	2

*1: プリンター経由で使用している場合のみ有効

⑤ プリンター応答種別

プリンターからの各種応答取得で利用する定数を下記に示します。

定数名	説明	値
PRINTER_RESPONSE_REQUEST	実行応答リクエスト	0
PRINTER_RESPONSE_USER_AREA	ユーザ領域の残り容量の送信	1
PRINTER_RESPONSE_ARRANGE_USER_AREA	ユーザ領域の整理後の残り容量の送信	2
PRINTER_RESPONSE_NV_GRAPHICS	NVグラフィックスメモリ容量の送信	3
PRINTER_RESPONSE_KEY_CODE	定義されているNVグラフィックスのキーコード一覧の送信	4

⑥ ディスプレイ応答種別

ディスプレイからの各種応答取得で利用する定数を下記に示します。

定数名	説明	値
DISPLAY_RESPONSE_REQUEST	実行応答リクエスト	0
DISPLAY_RESPONSE_USER_AREA	ユーザ領域の残り容量の送信	1
DISPLAY_RESPONSE_TEMPLATE_ID_LIST	テンプレートIDの送信	2
DISPLAY_RESPONSE_IMAGE_ID_LIST	イメージIDの送信	3
DISPLAY_RESPONSE_SLIDE_ID_LIST	スライドIDの送信	4
DISPLAY_RESPONSE_TEMPLATE_LABEL	テンプレート名の送信	5
DISPLAY_RESPONSE_IMAGE_LABEL	イメージ名の送信	6
DISPLAY_RESPONSE_SLIDE_LABEL	スライド名の送信	7

(3) 列挙型定数一覧

① 強調文字(CharacterBold)

強調文字で利用する列挙型定数を下記に示します。

定数名	説明
BOLD_CANCEL	強調文字を解除
BOLD	強調文字を指定

② アンダーライン(CharacterUnderline)

アンダーラインで利用する列挙型定数を下記に示します。

定数名	説明
UNDERLINE_CANCEL	アンダーラインを解除
UNDERLINE_1	1ドット幅アンダーラインを指定

③ 文字フォント(CharacterFont)

文字フォントで利用する列挙型定数を下記に示します。

定数名	説明
FONT_A	フォントA(24×12)
FONT_B	フォントB(16×8)

④ 文字倍率(CharacterScale)

文字倍率で利用する列挙型定数を下記に示します。

定数名	説明
VERTICAL_1_HORIZONTAL_1	縦1倍・横1倍
VERTICAL_1_HORIZONTAL_2	縦1倍・横2倍
VERTICAL_1_HORIZONTAL_3	縦1倍・横3倍
VERTICAL_1_HORIZONTAL_4	縦1倍・横4倍
VERTICAL_2_HORIZONTAL_1	縦2倍・横1倍
VERTICAL_2_HORIZONTAL_2	縦2倍・横2倍
VERTICAL_2_HORIZONTAL_3	縦2倍・横3倍
VERTICAL_2_HORIZONTAL_4	縦2倍・横4倍
VERTICAL_3_HORIZONTAL_1	縦3倍・横1倍
VERTICAL_3_HORIZONTAL_2	縦3倍・横2倍
VERTICAL_3_HORIZONTAL_3	縦3倍・横3倍
VERTICAL_3_HORIZONTAL_4	縦3倍・横4倍
VERTICAL_4_HORIZONTAL_1	縦4倍・横1倍
VERTICAL_4_HORIZONTAL_2	縦4倍・横2倍
VERTICAL_4_HORIZONTAL_3	縦4倍・横3倍
VERTICAL_4_HORIZONTAL_4	縦4倍・横4倍

⑤ 位置揃え(PrintAlignment)

位置揃えで利用する列挙型定数を下記に示します。

定数名	説明
ALIGNMENT_LEFT	左揃え
ALIGNMENT_CENTER	中央揃え
ALIGNMENT_RIGHT	右揃え

⑥ モジュールサイズ(ModuleSize)

QRコードのモジュールサイズで利用する列挙型定数を下記に示します。

定数名	説明	使用メソッド
QR_MODULE_SIZE_2	2ドット	setTemplateQrCodeData
QR_MODULE_SIZE_3	3ドット	
QR_MODULE_SIZE_4	4ドット	
QR_MODULE_SIZE_5	5ドット	
QR_MODULE_SIZE_6	6ドット	
QR_MODULE_SIZE_7	7ドット	
QR_MODULE_SIZE_8	8ドット	
QR_MODULE_SIZE_9	9ドット	
QR_MODULE_SIZE_10	10ドット	
QR_MODULE_SIZE_11	11ドット	

⑦ エラー訂正レベル(ErrorCorrection)

エラー訂正レベルで利用する列挙型定数を下記に示します。

定数名	説明	使用メソッド
QR_ERROR_CORRECTION_L	エラー訂正レベル L	setTemplateQrCodeData
QR_ERROR_CORRECTION_M	エラー訂正レベル M	
QR_ERROR_CORRECTION_H	エラー訂正レベル H	
QR_ERROR_CORRECTION_Q	エラー訂正レベル Q	

⑧ メモリ領域(MemoryArea)

メモリ領域の操作で利用する列挙型定数を下記に示します。

定数名	説明
MEMORY_DISPLAY_USERMEMORY	ユーザ領域

⑨ 登録フォント(RegisteredFont)

登録フォントで利用する列挙型定数を下記に示します。

定数名	説明
FONT_STANDARD	標準フォント
FONT_OPTION	オプションフォント

⑩ QRデータモード(QrDataMode)

QRデータモードで利用する列挙型定数を下記に示します。

定数名	説明
QRDATAMODE_NUMERIC	数字モード
QRDATAMODE_ALPHANUMERIC	英数字モード
QRDATAMODE_8BITBYTE	8ビットバイトモード
QRDATAMODE_KANJI	漢字モード
QRDATAMODE_MIXTURE	混在モード

⑪ QRクワイエットゾーン(QrQuietZone)

QRクワイエットゾーンで利用する列挙型定数を下記に示します。

定数名	説明
QRQUIETZONE_EXIST	QRクワイエットゾーンあり
QRQUIETZONE_NONE	QRクワイエットゾーンなし

⑫ マクロ登録処理(MacroRegistrationFunction)

マクロ登録処理で利用する列挙型定数を下記に示します。

定数名	説明
MACRO_REGISTRATION_CLEAR	マクロ登録処理の中止
MACRO_REGISTRATION_START	マクロ登録処理の開始
MACRO_REGISTRATION_REGIST	マクロ登録とマクロ登録処理の終了

PrinterManager**コンストラクタ**

`com.seikoinstruments.sdk.thermalprinter.PrinterManager`クラスのコンストラクタです。

構文 `public PrinterManager(Context context)`

パラメータ *context* 本メソッドを呼び出すアプリケーションのコンテキストを指定してください。
例: `MainActivity.this`

connect**プリンターとの通信開始(Bluetooth)**

Bluetooth接続でプリンターとの通信を開始します。

構文(a)のメソッドは、常にセキュアなモードでプリンターと通信を行います。

構文(b)のメソッドは、セキュアなモード、またはインセキュアなモードを指定してプリンターとの通信を行います。

構文 (a) `public void connect(int printerModel, String address) throws PrinterException`

(b) `public void connect(int printerModel, String address, boolean secure) throws PrinterException`

パラメータ *printerModel* Bluetooth接続対象のデバイスモデル定数
利用可能な定数は、「4.1.1(2)③ デバイスモデル」を参照してください。

address Bluetoothアドレス
例: `"00:11:22:AA:BB:CC"`

secure true セキュアなモードでプリンターと通信
false インセキュアなモードでプリンターと通信
通常はセキュアなモードでの通信を推奨します。

例外 **PrinterException**

本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーの内容は、「4.1.5 **PrinterException**クラス」を参照してください。

説明 本メソッドはプリンター経由で使用している場合のみ有効です。

本メソッドは、他の**PrinterManager**クラスのメソッドを使用する前に呼び出してください。

*address*で指定したBluetoothアドレスに対し、*printerModel*で指定したプリンターを接続します。
また、指定した*printerModel*を基に、接続時にプリンターの初期設定を行います。

USB接続でデバイスとの通信を開始します。

構文(a)のメソッドは、指定した機種 of デバイスと通信を行います。

構文(b)のメソッドは、指定した機種、及びパス of デバイスと通信を行います。

構文 (a) `public void connect(int printerModel) throws PrinterException`
 (b) `public void connect(int printerModel, String address) throws PrinterException`

パラメータ *printerModel* USB接続対象のデバイスモデル定数
 利用可能な定数は、「4.1.1(2)③ デバイスモデル」を参照してください。

address USBデバイスファイルのパス
 例: `"/dev/bus/usb/001/002"`

例外 **PrinterException**
 本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
 エラーの内容は、「4.1.5 **PrinterException**クラス」を参照してください。

説明 本メソッドは、他の**PrinterManager**クラスのメソッドを使用する前に呼び出してください。

printerModel、及び、*address*で指定したデバイスに接続します。
 また、指定した*printerModel*を基に、接続時にデバイスの初期設定を行います。

TCP/IP接続でプリンターとの通信を開始します。

構文 `public void connect(int printerModel, String address) throws PrinterException`

パラメータ *printerModel* イーサネット接続対象のデバイスモデル定数
 利用可能な定数は、「4.1.1(2)③ デバイスモデル」を参照してください。

address IPアドレス
 例: `"192.168.0.190"`

例外 **PrinterException**
 本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
 エラーの内容は、「4.1.5 **PrinterException**クラス」を参照してください。

説明 本メソッドはプリンター経由で使用している場合のみ有効です。

本メソッドは、他の**PrinterManager**クラスのメソッドを使用する前に呼び出してください。

Androidデバイスと同じネットワークに接続されたプリンターと、TCP/IP接続で通信を開始します。
*address*で指定されたIPアドレスに対して接続を行います。通信にはTCPポート9100番、及び26100番を使用します。また、指定した*printerModel*を基に、接続時にプリンターの初期設定を行います。

• TCP/IP 接続中のライブラリのソケットの作成・削除について

ライブラリは `connect` 後、`disconnect` するまで作成したソケットを維持します。また、`disconnect` するまで別のアプリケーションから同じプリンターに接続することはできません。

プリンターへのデータ送信完了時を基準とし、`socketKeepingTime` で設定するソケット維持時間経過後、ソケットをいったん削除しますが、すぐに再度ソケットを作成します。

disconnect

デバイスとの通信切断

デバイスとの通信を終了します。

構文 `public void disconnect() throws PrinterException`

例外 **PrinterException**

本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーの内容は、「4.1.5 **PrinterException**クラス」を参照してください。

注意 本メソッドの直前に、`getPrinterResponse`の**PRINTER_RESPONSE_REQUEST**(実行応答リクエスト)、または`getDisplayResponse`の**DISPLAY_RESPONSE_REQUEST**(実行応答リクエスト)により実行応答を取得することを推奨します。取得しない場合は、Androidデバイスのデータが全てデバイスに送信される前に通信が切断され、データの一部分が消失する場合があります。
`getPrinterResponse`、または`getDisplayResponse`を実行しない場合は、お客様のプログラムにおいて、問題が無いことを十分評価してからご使用ください。

sendBinary

バイナリデータの送信

バイナリデータをデバイスに送信します。

構文 `public void sendBinary(byte [] binary) throws PrinterException`

パラメータ *binary* デバイスに送信するバイナリデータ
一度に指定可能なデータサイズは16KB(16384バイト)です。

例外 **PrinterException**

本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。
デバイスとの通信状態の確認は、`isConnect`を参照してください。

説明 本メソッドは、指定されたバイナリデータを変換せずにデバイスに送信します。

本メソッドでディスプレイコマンドをバイナリデータとして送信することにより、ライブラリではサポートされていないディスプレイ機能を利用できます。ただし、本メソッドはディスプレイから応答を取得するコマンドには対応しておりません。

ファイルデータを送信します。

構文 `public void sendDataFile(String fileName) throws PrinterException`

パラメータ *fileName* デバイスに送信するデータファイル名
指定可能な最大ファイルサイズは、1MB(1048576バイト)です。
送信可能なファイル拡張子とファイルの送信について下記に説明します。

- .bin、.dat
バイナリデータとして変換せずにデバイスに送信します。

例外 **PrinterException**

本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。
デバイスとの通信状態の確認は、**isConnect**を参照してください。

プリンターからの応答データを取得します。

構文 `public void getPrinterResponse(int id, Object buf) throws PrinterException`

パラメータ *id* プリンター応答種別定数
利用可能な定数は、「4.1.1(2)⑤ プリンター応答種別」を参照してください。

buf 取得した応答データを格納するバッファー
*id*で指定した応答データを*buf*で指定されたオブジェクトに格納します。
プリンター応答種別定数によりバッファーの型が異なります。
バッファーの型は、下表を参照してください。

応答種別定数	
パラメータ	説明
PRINTER_RESPONSE_REQUEST (実行応答リクエスト)	
<i>buf</i>	長さ1のint型の配列を指定してください。 <i>buf[0]</i> には0～15(00h～0Fh)を指定してください。 応答取得成功時には、 <i>buf[0]</i> に実行応答リクエストの応答コードが128～143(80h～8Fh)で格納されます。
PRINTER_RESPONSE_USER_AREA (ユーザ領域の残り容量の送信)	
<i>buf</i>	長さ1のint型の配列を指定してください。 応答取得成功時には、ユーザ領域の残り容量(単位:バイト)が数値で格納されます。
PRINTER_RESPONSE_ARRANGE_USER_AREA (ユーザ領域の整理後の残り容量の送信)	
<i>buf</i>	長さ1のint型の配列を指定してください。 応答取得成功時には、ユーザ領域の整理後の残り容量(単位:バイト)が数値で格納されます。

応答種別定数	
パラメータ	説明
PRINTER_RESPONSE_NV_GRAPHICS (NVグラフィックスメモリ容量の送信)	
<i>buf</i>	長さ1のint型の配列を指定してください。 応答取得成功時には、NVグラフィックスメモリ容量(単位: バイト)が数値で格納されます。
PRINTER_RESPONSE_KEY_CODE (定義されているNVグラフィックスのキーコード一覧の送信)	
<i>buf</i>	ArrayList<String>型の配列を指定してください。 応答取得成功時には、NVグラフィックスのキーコードが文字列の配列で格納されます。 例: <i>buf.size()</i> = 3、 <i>buf[0]</i> = "22"、 <i>buf[1]</i> = "23"、 <i>buf[2]</i> = "24"など

例外 **PrinterException**
 本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
 エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。
 データ送信に失敗すると、プリンターとの通信を終了し、**PrinterException**をスローすることがあります。プリンターとの通信状態の確認は、**isConnect**を参照してください。

startDiscoveryPrinter

プリンターの探索開始(Bluetooth)

Bluetooth接続のプリンターを探索します。発見されたプリンターのプリンター情報は、**PrinterInfo**クラスに格納されます。

構文 `public void startDiscoveryPrinter(PrinterListener listener) throws PrinterException`

パラメータ *listener* **PrinterListener**のインスタンス
 本メソッドの終了や**cancelDiscoveryPrinter**による中断は*listener*に設定されたインスタンスを経由して、**finishEvent**により、終了イベントとしてユーザーアプリケーションに通知されます。

例外 **PrinterException**
 本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
 エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。

説明 本メソッドでは、SII製プリンター以外も発見される場合があります。
 また、すでにライブラリや他のアプリケーションでBluetooth接続が確立されているプリンターは発見されません。

本メソッドはアプリケーションのメインスレッドからは呼び出さないでください。

```
public void startDiscoveryPrinter(PrinterListener listener, int deviceType) throws PrinterException
```

deviceType ポート種別
PRINTER TYPE USBを指定してください。

説明 本メソッドは、SII製デバイスを探査します。発見したデバイスのデバイス情報は、後述する **PrinterInfo** クラスに格納されます。

探索の終了や中断は`listener`に設定されたインスタンスを経由して、`finishEvent`により、イベントとしてユーザアプリケーションに通知されます。

```
構文 public void startDiscoveryPrinter(PrinterListener listener, int retry, int timeout) throws PrinterException
```

<i>retry</i>	リトライ回数(回) <i>retry</i> で設定した回数だけローカルブロードキャストパケットを送信します。 有効範囲は、1～5です。 1未満の値を指定した場合は1回に設定されます。 5を超える値を指定した場合は5回に設定されます。
--------------	---

エラー 本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。

- 説明 本メソッドは、SII製プリンターを探索します。発見したプリンターのプリンター情報は、後述する **PrinterInfo**クラスに格納されます。
- 探索の終了や中断は`listener`に設定されたインスタンスを経由して、`finishEvent`により、イベントとしてユーザアプリケーションに通知されます。

startDiscoveryDevice

デバイスの探索開始(USB)

SII製デバイスを探索します。

構文 `public void startDiscoveryDevice(PrinterListener listener, int deviceType) throws PrinterException`

パラメータ `listener` **PrinterListener**のインスタンス
 本メソッドの終了や`cancelDiscoveryDevice`による中断は`listener`に設定されたインスタンスを経由して、`finishEvent`により、終了イベントとしてユーザアプリケーションに通知されます。

`deviceType` ポート種別
PRINTER_TYPE_USBを指定してください。

エラー 本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。

説明 本メソッドは、SII製デバイスを探索します。発見したデバイスのデバイス情報は、後述する **PrinterInfo**クラスに格納されます。

探索の終了や中断は`listener`に設定されたインスタンスを経由して、`finishEvent`により、イベントとしてユーザアプリケーションに通知されます。

cancelDiscoveryPrinter

プリンターの探索中断

本メソッドはサポートしていません。

構文 `public void cancelDiscoveryPrinter()`

getFoundPrinter

発見されたプリンター情報リストの取得

`startDiscoveryPrinter`で発見されたプリンターのプリンター情報を、格納先の**PrinterInfo**クラスから`ArrayList`で取得します。

構文 `public ArrayList<PrinterInfo> getFoundPrinter()`

戻り値 **PrinterInfo**クラスの`ArrayList`

発見されたデバイス情報リストの取得

startDiscoveryDeviceで発見されたデバイスのデバイス情報を、格納先の**PrinterInfo**クラスから**ArrayList**で取得します。

構文 `public ArrayList<PrinterInfo> getFoundDevice()`

戻り値 **PrinterInfo**クラスのArrayList

送信タイムアウト時間の取得

データ送信時のタイムアウト時間を取得します。

構文 `public int getSendTimeout()`

戻り値 送信タイムアウト時間(ミリ秒:ms)

説明 本メソッドはisConnectがtrue/falseに関わらず取得が可能です。

送信タイムアウト時間の設定

データ送信時のタイムアウト時間を設定します。

構文 `public void setSendTimeout(int sendTimeout)`

パラメータ	<i>sendTimeout</i>	送信タイムアウト時間(ミリ秒:ms) 有効範囲は、100～90000です。 有効範囲外の値を指定した場合は10000msに設定されます。
-------	--------------------	--

説明 本メソッドで送信タイムアウト時間の設定を行っていない場合の値は10000となります。

本メソッドはisConnectがtrue/falseに関わらず設定が可能です。

設定したタイムアウト時間が有効になるのは、次回データ送信時となります。

受信タイムアウト時間の取得

データ受信時のタイムアウト時間を取得します。

構文 `public int getReceiveTimeout()`

戻り値 受信タイムアウト時間(ミリ秒:ms)

説明 本メソッドはisConnectがtrue/falseに関わらず取得が可能です。

データ受信時のタイムアウト時間を設定します。

構文 `public void setReceiveTimeout(int receiveTimeout)`

パラメータ *receiveTimeout* 受信タイムアウト時間(ミリ秒:ms)
有効範囲は、100～90000です。
有効範囲外の値を指定した場合は10000msに設定されます。

説明 本メソッドで受信タイムアウト時間の設定を行っていない場合の値は10000となります。

本メソッドはisConnectがtrue/falseに関わらず設定が可能です。

設定したタイムアウト時間が有効になるのは、次回データ受信時となります。

国際文字セットの値を取得します。

構文 `public int getInternationalCharacter()`

戻り値 値については、「4.1.1(2)① 国際文字セット」を参照してください。

説明 sendDataFile、またはsetTemplateTextDataにてテキストデータを送信した場合は、以下に示す文字コードの表示結果が異なります。表示される文字に関しては、「付録A 文字セット」を参照してください。

国際文字セット設定により表示結果が異なる文字コード

0x23、0x24、0x40、0x5B、0x5C、0x5D、0x5E、0x60、0x7B、0x7C、0x7D、0x7E

国際文字セットの値を設定します。

構文 `public void setInternationalCharacter(int internationalCharacter)`

パラメータ *internationalCharacter* 国際文字セット定数
設定可能な値は、「4.1.1(2)① 国際文字セット」を参照してください。
無効な値を指定した場合は無視されます。

説明 本メソッドで国際文字セットの設定を行っていない場合は、Androidデバイスの言語設定により以下の国際文字セットとなります。

Androidデバイスの言語設定が日本の場合: COUNTRY_JAPAN

Androidデバイスの言語設定が日本以外の場合: COUNTRY_USA

コードページの値を取得します。

構文 `public int getCodePage()`

戻り値 値については、「4.1.1(2)② コードページ」を参照してください。

コードページの値を設定します。

構文 `public void setCodePage(int codePage)`

パラメータ *codePage* コードページ定数
設定可能な値は、「4.1.1(2)② コードページ」を参照してください。
無効な値を指定した場合は無視されます。

説明 本メソッドでコードページの設定を行っていない場合は、Androidデバイスの言語設定により以下のコードページとなります。

Androidデバイスの言語設定が日本の場合: `CODE_PAGE_KATAKANA`

Androidデバイスの言語設定が日本以外の場合: `CODE_PAGE_1252`

`sendDataFile`、または`setTemplateTextData`でテキストデータを送信した場合は、本プロパティの設定により利用するエンコードが変更されます。

通信中のデバイスモデルの値を取得します。

構文 `public int getPrinterModel()`

戻り値 値については、「4.1.1(2)③ デバイスモデル」を参照してください。
`isConnect`がfalseの場合は、`PRINTER_MODEL_DEFAULT`が返ります。

説明 デバイスが未接続でも、一度`connect`が成功している場合は、前回接続に成功したデバイスモデル値が返ります。

デバイスとの接続に利用しているポート種別の値を取得します。

構文 `public int getPortType()`

戻り値 値については、「4.1.1(2)④ ポート種別」を参照してください。
`isConnect`がfalseの場合は、`PRINTER_TYPE_BLUETOOTH`が返ります。

説明 デバイスが未接続でも、一度`connect`が成功している場合は、前回接続に成功したポート種別の値が返ります。

デバイスとの通信状態を確認します。

構文 `public boolean isConnect()`

戻り値	<code>true</code>	デバイスと通信中
	<code>false</code>	デバイスと未接続

説明 データ送信に失敗すると、デバイスとの通信を終了し、本メソッドは`false`になります。`false`の場合は、`connect`で再度通信を開始してください。

getSocketKeepingTime

ソケット維持時間の取得

ソケット維持時間を取得します。

構文 `public int getSocketKeepingTime()`

戻り値 ソケット維持時間(ミリ秒:ms)

説明 本メソッドはプリンター経由で使用している場合のみ有効です。
本メソッドは`isConnect`が`true/false`に関わらず設定が可能です。

setSocketKeepingTime

ソケット維持時間の設定

ソケット維持時間を設定します。

構文 `public void setSocketKeepingTime(int socketKeepingTime)`

有効範囲 `60000～300000`(ミリ秒:ms)
60000未満の値を指定した場合は60000msに設定されます。
300000を超える値を指定した場合は300000msに設定されます。

初期値 300000

説明 本メソッドはプリンター経由で使用している場合のみ有効です。
本メソッドは`isConnect`が`true/false`に関わらず設定または取得が可能です。

ソケット維持時間には、接続するプリンターの受信タイムアウト時間設定と等しい時間を指定してください。受信タイムアウト時間設定の詳細は「RP-F10シリーズ サーマルプリンタ技術説明書」に記載されている、プリンターコマンド「無線LAN通信の設定」を参照してください。

設定したソケット維持時間が有効になるのは、次回`connect`実行時となります。

メモリ領域を最適化します。

構文 `public void defragment(MemoryArea memoryArea) throws PrinterException`

パラメータ *memoryArea* メモリ領域
 利用可能な定数は、「4.1.1(3)⑧ メモリ領域(MemoryArea)」を参照してください。

例外 **PrinterException**

本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。
デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明 本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。
最適化には数分かかることがあります。
最適化中にデバイスの電源をオフにしないでください。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

メモリ領域を初期化します。

構文 `public void initializeMemoryArea(MemoryArea memoryArea) throws PrinterException`

パラメータ *memoryArea* メモリ領域
 利用可能な定数は、「4.1.1(3)⑧ メモリ領域(MemoryArea)」を参照してください。

例外 **PrinterException**

本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。
デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明 本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。
初期化には数分かかることがあります。
初期化中にデバイスの電源をオフにしないでください。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

注意

MEMORY_DISPLAY_USERMEMORYを指定して初期化すると、以下のメソッドで登録したデータは削除されます。

- registerTemplate
- registerImageData
- controlMacroRegistration
- registerUserDefinedCharacter
- registerOptionFont

工場出荷時に登録されるシステムで利用する一部のデータも削除されます。

それにより、プリンターでエラーなどが発生した際にガイド画面が表示できなくなります。

使用済みのメモリ領域を再度利用する場合は、defragmentをご利用ください。

showTemplate

テンプレートの表示

画面をテンプレート表示にします。

構文 `public void showTemplate(int time_ms) throws PrinterException`

パラメータ *time_ms*

表示時間(ms:ミリ秒)

画面に表示する時間を*time_ms* (ms)で指定してください。

有効範囲は0～25500です。

0を超える値、かつ100未満の値を指定した場合は100msに設定されます。

25500を超える値を指定した場合は25500msに設定されます。

マクロ登録の場合：

*time_ms*に0を指定した場合は、テンプレート表示を続けます。

*time_ms*に0以外を指定した場合は、表示時間経過後は次のテンプレートに移ります。

マクロ登録以外の場合：

*time_ms*に0を指定した場合は、テンプレート表示を続けます。

*time_ms*に0以外を指定した場合は、表示時間経過後は直前のテンプレートに戻ります。直前に表示時間に0以外を指定して画面更新を行っていた場合は、0を指定して画面更新を行ったテンプレートまで戻ります。

例外

PrinterException

本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。

エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。

データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明 画面を更新し、以下のメソッドで指定されたデータを表示します。

- `selectTemplate`
- `setTemplateImageData`
- `selectTemplateTextObject`
- `setTemplateTextAlignment`
- `setTemplateTextLeftMargin`
- `setTemplateTextLineSpacing`
- `setTemplateTextBold`
- `setTemplateTextUnderline`
- `setTemplateTextSize`
- `setTemplateTextFont`
- `setTemplateTextRightSpacing`
- `setTemplateTextColor`
- `setTemplateTextData`
- `setTemplateBarcodeData`
- `setTemplateQrCodeData`

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

showSlide

スライドの表示

画面をスライド表示にします。

構文 `public void showSlide(int slideID, int time_ms) throws PrinterException`

パラメータ *slideID*

スライドID

`registerSlideData`で登録したスライドデータのスライドIDを指定してください。
有効範囲は0～91です。
指定したスライドIDにスライドデータが登録されていない場合は無視されます。

time_ms

表示時間(ms:ミリ秒)

画面に表示する時間を`time_ms` (ms)で指定してください。

有効範囲は0～25500です。

0を超える値、かつ100未満の値を指定した場合は100msに設定されます。

25500を超える値を指定した場合は25500msに設定されます。

マクロ登録の場合:

`time_ms`に0を指定した場合は、スライド表示を続けます。

`time_ms`に0以外を指定した場合は、表示時間経過後は次のスライドに移ります。

マクロ登録以外の場合:

`time_ms`に0を指定した場合は、スライド表示を続けます。

`time_ms`に0以外を指定した場合は、表示時間経過後は直前のスライドに戻ります。直前に表示時間に0以外を指定して画面更新を行っていた場合は、0を指定して画面更新を行ったスライドまで戻ります。

例外	PrinterException 本メソッド呼び出し時にエラーが発生した場合は、 PrinterException をスローします。 エラーについての詳細は、「4.1.5 PrinterException クラス」を参照してください。 データ送信に失敗すると、デバイスとの通信を終了し、 PrinterException をスローすることがあります。 デバイスとの通信状態の確認は、 isConnect を参照してください。
説明	選択中のテンプレートは解除されます。 プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

enterStandbyMode

スタンバイの表示

ディスプレイをスタンバイモードにします。

構文	<code>public void enterStandbyMode() throws PrinterException</code>
例外	PrinterException 本メソッド呼び出し時にエラーが発生した場合は、 PrinterException をスローします。 エラーについての詳細は、「4.1.5 PrinterException クラス」を参照してください。 データ送信に失敗すると、デバイスとの通信を終了し、 PrinterException をスローすることがあります。 デバイスとの通信状態の確認は、 isConnect を参照してください。
説明	本メソッドはスタンバイモード中、及びガイドモード中は無視されます。 選択中のテンプレートは選択が解除されます。 プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

executeMacro

マクロの実行

マクロを実行します。

構文	<code>public void executeMacro(int macroID, int repeatCount) throws PrinterException</code>				
パラメータ	<table> <tr> <td><i>macroID</i></td><td> マクロID controlMacroRegistrationで登録したマクロのマクロIDを指定してください。 有効範囲は0～127です。 指定したマクロIDにマクロが登録されていない場合は無視されます。 </td></tr> <tr> <td><i>repeatCount</i></td><td> 実行回数 マクロを実行する回数を指定してください。 有効範囲は0～255です。 0を指定した場合は、繰り返しを継続します。 </td></tr> </table>	<i>macroID</i>	マクロID controlMacroRegistration で登録したマクロのマクロIDを指定してください。 有効範囲は0～127です。 指定したマクロIDにマクロが登録されていない場合は無視されます。	<i>repeatCount</i>	実行回数 マクロを実行する回数を指定してください。 有効範囲は0～255です。 0を指定した場合は、繰り返しを継続します。
<i>macroID</i>	マクロID controlMacroRegistration で登録したマクロのマクロIDを指定してください。 有効範囲は0～127です。 指定したマクロIDにマクロが登録されていない場合は無視されます。				
<i>repeatCount</i>	実行回数 マクロを実行する回数を指定してください。 有効範囲は0～255です。 0を指定した場合は、繰り返しを継続します。				
例外	PrinterException 本メソッド呼び出し時にエラーが発生した場合は、 PrinterException をスローします。 エラーについての詳細は、「4.1.5 PrinterException クラス」を参照してください。 データ送信に失敗すると、デバイスとの通信を終了し、 PrinterException をスローすることがあります。 デバイスとの通信状態の確認は、 isConnect を参照してください。				
説明	選択中のテンプレートは解除されます。				

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

turnOnScreen

画面の表示オン・オフ

画面のバックライトのオン・オフを設定します。

構文 `public void turnOnScreen(boolean isOn) throws PrinterException`

パラメータ *isOn* 画面状態
画面状態を以下から指定してください。
true バックライトオン
false バックライトオフ

例外 **PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。

説明 プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

selectTemplate

テンプレートの選択

ディスプレイで表示するテンプレートを選択します。

構文(a)のメソッドは、テンプレートとテンプレートの背景に使用するスライドデータを選択します。

構文(b)のメソッドは、テンプレートを選択します。

構文 (a) `public void selectTemplate(int templateID, int slideID) throws PrinterException`

(b) `public void selectTemplate(int templateID) throws PrinterException`

パラメータ *templateID* テンプレートID
選択するテンプレートのテンプレートIDを指定してください。
有効範囲は0～127です。
指定したテンプレートIDにテンプレートが登録されていない場合は無視されます。

slideID スライドID
テンプレートの背景に使用するスライドデータのスライドIDを指定してください。
有効範囲は0～91です。
指定したスライドIDにスライドデータが登録されていない場合は無視されます。

例外 **PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。
デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明 本メソッドを実行するとテンプレート上のデータはクリアされます。
選択したテンプレートは**showTemplate**を実行すると画面に表示されます。

選択中のテンプレートはshowSlide、enterStandbyMode、executeMacroで選択が解除されます。
選択中のテンプレートはshowTemplateで時間指定表示を実行すると選択が解除されます。

setTemplateTextDataで20h～7EhのASCII文字以外を入力する場合、codePageの値に応じて以下のテンプレートを使用してください。

codePageがCODE_PAGE_KATAKANAの場合:

エンコード指定がShift_JISのテンプレートを使用してください。

codePageがCODE_PAGE_KATAKANA以外:

エンコード指定がISO-2022-JPのテンプレートを使用してください。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateImageData

イメージデータの設定

選択中のテンプレートに表示するイメージデータを設定します。

構文 `public void setTemplateImageData(int mapID, int imageID) throws PrinterException`

パラメータ *mapID*

マップID

有効範囲は0～63です。

指定したマップIDがテンプレートに定義されていない場合は無視されます。

imageID

イメージID

registerImageDataで登録したイメージデータのイメージIDを指定してください。

有効範囲は0～63です。

指定したイメージIDにイメージデータが登録されていない場合は無視されます。

例外

PrinterException

本メソッド呼び出し時にエラーが発生した場合は、PrinterExceptionをスローします。

エラーについての詳細は、「4.1.5 PrinterExceptionクラス」を参照してください。

データ送信に失敗すると、デバイスとの通信を終了し、PrinterExceptionをスローすることがあります。デバイスとの通信状態の確認は、isConnectを参照してください。

説明

選択中のテンプレートのマップIDを本メソッドで指定してから、配置するイメージIDを指定してください。

指定したイメージデータはshowTemplateを実行すると画面に表示されます。

本メソッドの設定は以下の条件でクリアされます。

- ・selectTemplateを実行した場合
- ・showTemplateのtime_msで0以外を指定し、指定した表示時間を経過した場合
- ・executeMacroに登録されているshowTemplateを実行した場合

テンプレートが未選択の場合は無視されます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

編集を開始するtext要素を選択します。

構文 `public void selectTemplateTextObject(int mapID) throws PrinterException`

パラメータ *mapID* マップID
有効範囲は0～63です。
指定したマップIDがテンプレートに定義されていない場合は無視されます。

例外 **PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。
デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明 選択中のテンプレートのマップIDを本メソッドで指定してから、文字の編集を開始してください。

指定したマップIDのtext要素にスクロールを設定している場合は、**showTemplate**の後に本メソッドを実行した場合、スクロールを実行します。
指定したマップIDのtext要素にスクロールを設定していない場合は、**showTemplate**の後に本メソッドを実行した場合、入力済みのテキストデータを破棄します。

本メソッドの設定は以下の条件でクリアされます。

- ・**selectTemplate**を実行した場合
- ・**showTemplate**を実行した場合

テンプレートが未選択の場合は無視されます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

ディスプレイで表示するテキストデータに位置揃えを設定します。

構文 `public void setTemplateTextAlignment(PrintAlignment alignment) throws PrinterException`

パラメータ *alignment* 位置揃え
利用可能な定数は、「4.1.1(3)⑤ 位置揃え(PrintAlignment)」を参照してください。

例外 **PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。
デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明 位置揃えは以下の場合のみ有効です。

- ・**selectTemplateTextObject**で指定したマップIDにテキストデータが未入力時
- ・**selectTemplateTextObject**で指定したマップIDに入力したテキストデータが改行直後

例外	<p>PrinterException</p> <p>本メソッド呼び出し時にエラーが発生した場合は、PrinterExceptionをスローします。</p> <p>エラーについての詳細は、「4.1.5 PrinterExceptionクラス」を参照してください。</p> <p>データ送信に失敗すると、デバイスとの通信を終了し、PrinterExceptionをスローすることがあります。デバイスとの通信状態の確認は、isConnectを参照してください。</p>
説明	<p>改行量の設定は以下の場合のみ有効です。</p> <ul style="list-style-type: none"> •selectTemplateTextObjectで指定したマップIDにテキストデータが未入力時 •selectTemplateTextObjectで指定したマップIDに入力したテキストデータが改行直後 <p>指定したマップIDのtext要素にスクロールを設定している場合は、本メソッドの設定は反映されません。</p> <p>本メソッドを実行する前にselectTemplateTextObjectでマップIDを指定してください。</p> <p>本メソッドを実行した後にsetTemplateTextDataでテキストデータを入力してください。</p> <p>入力したテキストデータはshowTemplateを実行すると画面に表示されます。</p> <p>本メソッドの設定は以下の条件でクリアされます。</p> <ul style="list-style-type: none"> •selectTemplateを実行した場合 •showTemplateの<code>time_ms</code>で0以外を指定し、指定した表示時間を経過した場合 •executeMacroに登録されているshowTemplateを実行した場合 <p>プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。</p>

setTemplateTextBold

テキストデータの強調文字の設定

ディスプレイで表示するテキストデータに強調文字を設定します。

構文	<code>public void setTemplateTextBold(CharacterBold bold) throws PrinterException</code>
パラメータ	<p><code>bold</code> 強調文字</p> <p>利用可能な定数は、「4.1.1(3)① 強調文字(CharacterBold)」を参照してください。</p>
例外	<p>PrinterException</p> <p>本メソッド呼び出し時にエラーが発生した場合は、PrinterExceptionをスローします。</p> <p>エラーについての詳細は、「4.1.5 PrinterExceptionクラス」を参照してください。</p> <p>データ送信に失敗すると、デバイスとの通信を終了し、PrinterExceptionをスローすることがあります。デバイスとの通信状態の確認は、isConnectを参照してください。</p>
説明	<p>本メソッド実行後のテキストデータから強調文字は適用されます。1文字ずつ強調文字の設定ができます。</p> <p>本メソッドを実行する前にselectTemplateTextObjectでマップIDを指定してください。</p> <p>本メソッドを実行した後にsetTemplateTextDataでテキストデータを入力してください。</p> <p>入力したテキストデータはshowTemplateを実行すると画面に表示されます。</p> <p>本メソッドの設定は以下の条件でクリアされます。</p> <ul style="list-style-type: none"> •selectTemplateを実行した場合 •showTemplateの<code>time_ms</code>で0以外を指定し、指定した表示時間を経過した場合 •executeMacroに登録されているshowTemplateを実行した場合

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateTextUnderline

テキストデータのアンダーラインの設定

ディスプレイで表示するテキストデータにアンダーラインを設定します。

構文 `public void setTemplateTextUnderline(CharacterUnderline underline) throws PrinterException`

パラメータ *underline* アンダーライン
利用可能な定数は、「4.1.1(3)② アンダーライン(CharacterUnderline)」を参照してください。

例外 **PrinterException**

本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。
デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明 本メソッド実行後のテキストデータからアンダーラインは適用されます。1文字ずつアンダーラインの設定ができます。

本メソッドを実行する前に**selectTemplateTextObject**でマップIDを指定してください。
本メソッドを実行した後に**setTemplateTextData**でテキストデータを入力してください。
入力したテキストデータは**showTemplate**を実行すると画面に表示されます。

本メソッドの設定は以下の条件でクリアされます。

- ・**selectTemplate**を実行した場合
- ・**showTemplate**の*time_ms*で0以外を指定し、指定した表示時間を経過した場合
- ・**executeMacro**に登録されている**showTemplate**を実行した場合

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateTextSize

テキストデータの文字サイズの設定

ディスプレイで表示するテキストデータに文字サイズを設定します。

構文 `public void setTemplateTextSize(CharacterScale scale) throws PrinterException`

パラメータ *scale* 文字倍率
利用可能な定数は、「4.1.1(3)④ 文字倍率(CharacterScale)」を参照してください。

例外 **PrinterException**

本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。
デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明 本メソッド実行後のテキストデータから文字サイズは適用されます。1文字ずつ文字サイズの設定ができます。

本メソッドを実行する前にselectTemplateTextObjectでマップIDを指定してください。
本メソッドを実行した後にsetTemplateTextDataでテキストデータを入力してください。
入力したテキストデータはshowTemplateを実行すると画面に表示されます。

本メソッドの設定は以下の条件でクリアされます。

- selectTemplateを実行した場合
- showTemplateのtime_msで0以外を指定し、指定した表示時間を経過した場合
- executeMacroに登録されているshowTemplateを実行した場合

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateTextFont

テキストデータの文字フォントの設定

ディスプレイで表示するテキストデータに文字フォントを設定します。

構文 public void setTemplateTextFont(CharacterFont font) throws PrinterException

パラメータ font 文字フォント
 利用可能な定数は、「4.1.1(3)③ 文字フォント(CharacterFont)」を参照してください。

例外 PrinterException

本メソッド呼び出し時にエラーが発生した場合は、PrinterExceptionをスローします。
エラーについての詳細は、「4.1.5 PrinterExceptionクラス」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、PrinterExceptionをスローすることがあります。
デバイスとの通信状態の確認は、isConnectを参照してください。

説明 本メソッド実行後のテキストデータから文字フォントは適用されます。1文字ずつ文字フォントの設定ができます。

本メソッドを実行する前にselectTemplateTextObjectでマップIDを指定してください。
本メソッドを実行した後にsetTemplateTextDataでテキストデータを入力してください。
入力したテキストデータはshowTemplateを実行すると画面に表示されます。

本メソッドの設定は以下の条件でクリアされます。

- selectTemplateを実行した場合
- showTemplateのtime_msで0以外を指定し、指定した表示時間を経過した場合
- executeMacroに登録されているshowTemplateを実行した場合

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateTextRegisteredFont

テキストデータの登録フォントの設定

ディスプレイで表示するテキストデータで使用する登録フォントを設定します。

構文 public void setTemplateTextRegisteredFont(RegisteredFont font) throws PrinterException

パラメータ font 登録フォント
 利用可能な定数は、「4.1.1(3)⑨ 登録フォント(RegisteredFont)」を参照してください。
 オプションフォントが登録されていない場合は無視されます。

例外	<p>PrinterException</p> <p>本メソッド呼び出し時にエラーが発生した場合は、PrinterExceptionをスローします。 エラーについての詳細は、「4.1.5 PrinterExceptionクラス」を参照してください。 データ送信に失敗すると、デバイスとの通信を終了し、PrinterExceptionをスローすることがあります。 デバイスとの通信状態の確認は、isConnectを参照してください。</p>
説明	<p>本メソッド実行後のテキストデータから登録フォントは適用されます。1文字ずつ登録フォントの設定ができます。</p> <p>本メソッドを実行した後にsetTemplateTextDataでテキストデータを入力してください。 入力したテキストデータはshowTemplateを実行すると画面に表示されます。</p> <p>本メソッドの設定は以下の条件でクリアされます。</p> <ul style="list-style-type: none"> ・selectTemplateを実行した場合 ・showTemplateの<code>time_ms</code>で0以外を指定し、指定した表示時間を経過した場合 ・executeMacroに登録されているshowTemplateを実行した場合 <p>プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。</p>

setTemplateTextRightSpacing

テキストデータの右スペース量の設定

ディスプレイで表示するテキストデータに右スペース量を設定します。

構文 `public void setTemplateTextRightSpacing(int spacing) throws PrinterException`

パラメータ `spacing` 文字の右スペース量(ピクセル:px)
有効範囲は0~255です。

例外	<p>PrinterException</p> <p>本メソッド呼び出し時にエラーが発生した場合は、PrinterExceptionをスローします。 エラーについての詳細は、「4.1.5 PrinterExceptionクラス」を参照してください。 データ送信に失敗すると、デバイスとの通信を終了し、PrinterExceptionをスローすることがあります。 デバイスとの通信状態の確認は、isConnectを参照してください。</p>
説明	<p>本メソッド実行後のテキストデータから文字の右スペース量は適用されます。1文字ずつ右スペース量の設定ができます。</p> <p>本メソッドを実行する前にselectTemplateTextObjectでマップIDを指定してください。 本メソッドを実行した後にsetTemplateTextDataでテキストデータを入力してください。 入力したテキストデータはshowTemplateを実行すると画面に表示されます。</p> <p>本メソッドの設定は以下の条件でクリアされます。</p> <ul style="list-style-type: none"> ・selectTemplateを実行した場合 ・showTemplateの<code>time_ms</code>で0以外を指定し、指定した表示時間を経過した場合 ・executeMacroに登録されているshowTemplateを実行した場合 <p>プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。</p>

ディスプレイで表示するテキストデータで使用する文字色を設定します。

構文 `public void setTemplateTextColor(int color) throws PrinterException`

パラメータ *color* 文字色
有効範囲は0～0xffffffffです。

例外 **PrinterException**

本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明

本メソッドでテキストデータに文字色を設定できます。
文字色はRGB24bitカラーで設定できます。設定した文字色はRGB555の16bitカラーに減色して表示されます。

本メソッド実行後のテキストデータから文字色は適用されます。1文字ずつ文字色の設定ができます。

本メソッドを実行する前に**selectTemplateTextObject**でマップIDを指定してください。
本メソッドを実行した後に**setTemplateTextData**でテキストデータを入力してください。
入力したテキストデータは**showTemplate**を実行すると画面に表示されます。

本メソッドの設定は以下の条件でクリアされます。

- ・**selectTemplate**を実行した場合
- ・**showTemplate**の*time_ms*で0以外を指定し、指定した表示時間を経過した場合
- ・**executeMacro**に登録されている**showTemplate**を実行した場合

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

ディスプレイで表示するテキストデータを入力します。

構文 `public void setTemplateTextData(String text) throws PrinterException`

パラメータ *text* ディスプレイで表示するテキストデータ
一度に指定可能なデータサイズは1～1020バイトです。

例外 **PrinterException**

本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明

本メソッドは、入力されたテキストデータを、**internationalCharacter**、及び**codePage**の設定を基に表示可能なテキストデータにエンコードして画面に表示します。

本メソッドを実行する前に**selectTemplate**でテンプレートを選択してください。
 選択中のテンプレートのマップIDを**selectTemplateTextObject**で指定してから、本メソッドでテキストデータを入力してください。
selectTemplateTextObjectでマップIDが指定されていない場合は、無視されます。
 入力したテキストデータは**showTemplate**を実行すると画面に表示されます。

本メソッドで入力したテキストデータは以下の条件でクリアされます。

- ・**selectTemplate**を実行した場合
- ・**showTemplate**の`time_ms`で0以外を指定し、指定した表示時間を経過した場合
- ・**executeMacro**に登録されている**showTemplate**を実行した場合

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateBarcodeData

バーコードデータの入力

選択中のテンプレートのbarcode要素のマップIDを指定し、バーコードデータを入力します。

構文(a)のメソッドは、表示するバーコードにデータを文字列で入力します。

構文(b)のメソッドは、表示するバーコードにデータをバイト配列で入力します。

構文 (a) `public void setTemplateBarcodeData(int mapID, String text) throws PrinterException`
 (b) `public void setTemplateBarcodeData(int mapID, byte [] data) throws PrinterException`

パラメータ	<p><i>mapID</i> マップID 有効範囲は0～7です。 指定したマップIDが選択中のテンプレートに定義されていない場合は無視されます。</p> <p><i>text</i> バーコードデータ 入力可能な文字は、英数字('0'～'9'、'A'～'Z'、'a'～'z')などのASCII文字コード20h(スペース)～7Eh(チルダ)の文字です。 入力可能な文字数は1～150文字です。 バーコードの仕様に沿わないバーコードデータは無視されます。</p> <p><i>data</i> バーコードデータ 入力可能な値は00h～7Fhです。 入力可能なデータ数は1～150です。 バーコードの仕様に沿わないバーコードデータは無視されます。</p>
-------	--

例外 **PrinterException**
 本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
 エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。
 データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。
 デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明 選択中のテンプレートのマップIDを本メソッドで指定してから、バーコードデータを入力してください。
 入力したバーコードデータは**showTemplate**を実行すると画面に表示されます。

本メソッドで入力したバーコードデータは以下の条件でクリアされます。

- ・**selectTemplate**を実行した場合
- ・**showTemplate**の`time_ms`で0以外を指定し、指定した表示時間を経過した場合
- ・**executeMacro**に登録されている**showTemplate**を実行した場合

テンプレートが未選択の場合は無視されます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateQrCodeData

QRコードデータの入力

選択中のテンプレートのqr要素のマップIDを指定し、QRコードデータを入力します。

構文(a)のメソッドは、QRコードデータを文字列で入力します。

構文(b)のメソッドは、QRコードデータを文字列で入力します。`moduleSize`、`errorCorrection`、`mode`、及び`qrQuietZone`は選択中のテンプレートの設定が反映されます。

構文(c)のメソッドは、QRコードデータをバイト配列で入力します。

構文(d)のメソッドは、QRコードデータをバイト配列で入力します。`moduleSize`、`errorCorrection`、`mode`、及び`qrQuietZone`は選択中のテンプレートの設定が反映されます。

- 構文
- (a) `public void setTemplateQrCodeData(int mapID, ModuleSize moduleSize, ErrorCorrection errorCorrection, QrDataMode mode, QrQuietZone qrQuietZone, String text) throws PrinterException`
- (b) `public void setTemplateQrCodeData(int mapID, String text) throws PrinterException`
- (c) `public void setTemplateQrCodeData(int mapID, ModuleSize moduleSize, ErrorCorrection errorCorrection, QrDataMode mode, QrQuietZone qrQuietZone, byte [] data) throws PrinterException`
- (d) `public void setTemplateQrCodeData(int mapID, byte [] data) throws PrinterException`

パラメータ	<code>mapID</code>	マップID 有効範囲は0～7です。 指定したマップIDが選択中のテンプレートに定義されていない場合は無視されます。
	<code>moduleSize</code>	モジュールサイズ 利用可能な定数は、「4.1.1(3)⑥ モジュールサイズ(ModuleSize)」を参照してください。
	<code>errorCorrection</code>	エラー訂正レベル 利用可能な定数は、「4.1.1(3)⑦ エラー訂正レベル(ErrorCorrection)」を参照してください。

<i>mode</i>	<p>データモード</p> <p>利用可能な定数は、「4.1.1(3)⑩ QRデータモード(QrDataMode)」を参照してください。</p>
<i>qrQuietZone</i>	<p>クワイエットゾーン</p> <p>利用可能な定数は、「4.1.1(3)⑪ QRクワイエットゾーン(QrQuietZone)」を参照してください。</p>
<i>text</i>	<p>QRコードデータ</p> <p>入力可能な文字は以下の通りです。</p> <ul style="list-style-type: none"> ・英数字('0' ~ '9'、'A' ~ 'Z'、'a' ~ 'z')などのASCII文字コード 20h(スペース) ~ 7Eh(チルダ)の文字 ・JIS X 0201に基づく8ビットラテン/カナ文字 ・JIS X 0208に基づくシフトJISコード <p>入力可能なデータサイズは1~3909バイトです。</p> <p>QRコードの仕様に沿わないQRコードデータは無視されます。</p>
<i>data</i>	<p>QRコードデータ</p> <p>ディスプレイで表示するQRコードデータ</p> <p>入力可能な値は00h~FFhです。</p> <p>入力可能なデータ数は1~3909です。</p> <p>QRコードの仕様に沿わないQRコードデータは無視されます。</p>
例外	<p>PrinterException</p> <p>本メソッド呼び出し時にエラーが発生した場合は、PrinterExceptionをスローします。</p> <p>エラーについての詳細は、「4.1.5 PrinterExceptionクラス」を参照してください。</p> <p>データ送信に失敗すると、デバイスとの通信を終了し、PrinterExceptionをスローすることがあります。デバイスとの通信状態の確認は、isConnectを参照してください。</p>
説明	<p>選択中のテンプレートのマップIDを本メソッドで指定してから、QRコードデータを入力してください。</p> <p>入力したQRコードデータはshowTemplateを実行すると画面に表示されます。</p> <p>本メソッドで入力したQRコードデータは以下の条件でクリアされます。</p> <ul style="list-style-type: none"> ・selectTemplateを実行した場合 ・showTemplateの<code>time_ms</code>で0以外を指定し、指定した表示時間を経過した場合 ・executeMacroに登録されているshowTemplateを実行した場合 <p>テンプレートが未選択の場合は無視されます。</p> <p>プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。</p>

テンプレートを登録します。

```
構文 public void registerTemplate(int templateID,
                                String label,
                                String filePath) throws PrinterException
```

パラメータ	templateID	テンプレートID 登録するテンプレートのテンプレートIDを指定してください。 有効範囲は0～127です。 テンプレートID=127はシステムで利用していますので、指定しないでください。
-------	------------	---

label

テンプレート名

登録するテンプレートに識別用の名前を付けることができます。

指定可能な文字は、英数字('0' ~ '9'、'A' ~ 'Z'、'a' ~ 'z')などのASCII文字コード20h(スペース) ~ 7Eh(チルダ)の文字です。

ユニコードの00A5h('¥')は用いないでください。

指定可能な文字数は0 ~ 32文字です。

*label*は省略することも可能です。省略する場合はnullを指定してください。

指定したテンプレート名は**getDisplayResponse**で取得できます。

filePath ディスプレイに登録するテンプレートデータのファイルパス
サポートするファイルの拡張子は.xmlです。
登録可能な最大データサイズは8192バイトです。
登録するテンプレートデータの詳細については「ディスプレイ用技術説明書」
を参照してください。

例外 PrinterException
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.1.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。
デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明 本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

注意 工場出荷時に登録されるデータは、品質向上のため予告無く追加・変更される場合があります。

登録済みのテンプレートを削除します。

構文 `public void unregisterTemplate(int templateID) throws PrinterException`

パラメータ <i>templateID</i>	テンプレートID 削除するテンプレートのテンプレートIDを指定してください。 有効範囲は0～127です。 指定したテンプレートIDにテンプレートが登録されていない場合は無視されます。
-------------------------	--

例外 PrinterException
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.1.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。
デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明 本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

テンプレートを削除しても使用済みのメモリは解放されません。使用済みのメモリは、defragmentを実行すると再度利用できます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

イメージデータを登録します。

```
構文 public void registerImageData(int imageID,
                                   String label,
                                   String filePath) throws PrinterException
```

パラメータ	<i>imageID</i>	イメージID 有効範囲は0～63です。 イメージID=49～63はシステムで利用していますので、指定しないでください。
-------	----------------	---

label

イメージ名

登録するイメージデータに識別用の名前を付けることができます。

指定可能な文字は、英数字('0' ~ '9'、'A' ~ 'Z'、'a' ~ 'z')などのASCII文字コード20h(スペース) ~ 7Eh(チルダ)の文字です。

ユニコードの00A5h('¥')は用いないでください。

指定可能な文字数は0 ~ 32文字です。

*label*は省略することも可能です。省略する場合はnullを指定してください。

指定したイメージ名はgetDisplayResponseで取得できます。

filePath ファイルパス
登録するイメージデータのファイル名を指定してください。
サポートするファイルの拡張子は.jpg、.jpeg、及び.pngです。
ただし、サポートする拡張子であっても、フォーマットによっては登録できない場合があります。
指定可能な最大ファイルサイズは786432バイトです。
登録できる最大データサイズは横480×縦272ピクセル(px)です。

例外 **PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。
デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明 本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

注意 工場出荷時に登録されるデータは、品質向上のため予告無く追加・変更される場合があります。

unregisterImageData

イメージデータの削除

登録済みのイメージデータを削除します。

構文 `public void unregisterImageData(int imageID) throws PrinterException`

パラメータ *imageID* イメージID
有効範囲は0～63です。
指定したイメージIDにイメージデータが登録されていない場合は無視されます。

例外 **PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。
デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明 本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

イメージデータを削除しても使用済みのメモリは解放されません。使用済みのメモリは、**defragment**を実行すると再度利用できます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

スライドデータを登録します。

```
public void registerSlideData(int slideID,
                             String label,
                             String filePath) throws PrinterException
```

パラメータ	<i>slideID</i>	スライドID 有効範囲は0～91です。 スライドID=80～90はシステムで利用していますので、指定しないでください。
-------	----------------	---

label

スライド名

登録するスライドデータに識別用の名前を付けることができます。

指定可能な文字は、英数字('0' ~ '9'、'A' ~ 'Z'、'a' ~ 'z')などのASCII文字コード20h(スペース) ~ 7Eh(チルダ)の文字です。

ユニコードの00A5h('¥')は用いないでください。

指定可能な文字数は0 ~ 32文字です。

*label*は省略することも可能です。省略する場合はnullを指定してください。

指定したスライド名は`getDisplayResponse`で取得できます。

filePath ファイルパス

登録するスライドデータのファイル名を指定してください。

サポートするファイルの拡張子は.jpg、.jpeg、及び.pngです。

ただし、サポートする拡張子であっても、フォーマットによっては登録できない場合があります。

指定可能な最大ファイルサイズは786432バイトです。

登録できるデータサイズは横480×縦272ピクセル(px)固定です。

例外 PrinterException
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.1.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。
デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明 本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

登録したスライドデータを表示する場合はshowSlideを実行してください。
登録したスライドデータをテンプレートの背景として使用する場合はselectTemplateでスライドIDを指定してください。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

注意 工場出荷時に登録されるデータは、品質向上のため予告無く追加・変更される場合があります。

登録済みのスライドデータを削除します。

構文 `public void unregisterSlideData(int slideID) throws PrinterException`

パラメータ *slideID*

スライドID

有効範囲は0～91です。

指定したスライドIDにスライドデータが登録されていない場合は無視されます。

例外 **PrinterException**

本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。

エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。

データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明

本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

ディスプレイに外字を登録します。

構文 `public void registerUserDefinedCharacter(String filePath) throws PrinterException`

パラメータ *filePath*

ファイルパス

登録する外字のファイル名を指定してください。

サポートするファイルの拡張子は.binです。

外字データについては、「ディスプレイ用技術説明書」のディスプレイコマンド「外字の登録」を参照してください。

例外 **PrinterException**

本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。

エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。

データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明

本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

外字が登録済みの状態で本メソッドを実行した場合は、外字は上書きされます。

外字の表示にはエンコード指定がShift_JISのテンプレートを使用してください。

外字を表示する前に**setCodePage**の*codePage*に**CODE_PAGE_KATAKANA**を指定してください。

外字を表示する際は**setTemplateTextData**の*text*に指定可能な文字コードを指定してください。

指定可能な文字コードはE000h～E05Dhです。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

登録済みの外字を削除します。

構文 `public void unregisterUserDefinedCharacter() throws PrinterException`

例外 **PrinterException**

本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。
デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明 本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

登録されている外字は全て削除されます。
外字を削除しても使用済みのメモリは解放されません。使用済みのメモリは、**defragment**を実行すると再度利用できます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

ディスプレイにオプションフォントを登録します。

構文 `public void registerOptionFont(int startCode,
 int endCode,
 int width,
 int height,
 String filePath) throws PrinterException`

パラメータ	<i>startCode</i>	登録開始文字コード 有効範囲は20h~FFhのASCII文字コードです。
	<i>endCode</i>	登録終了文字コード 有効範囲は20h~FFhのASCII文字コードです。
	<i>width</i>	文字幅(ピクセル:px) 有効範囲は1~255です。
	<i>height</i>	文字高さ(ピクセル:px) 有効範囲は1~255です。
	<i>filePath</i>	ファイルパス 登録するオプションフォントのファイル名を指定してください。 サポートするファイルの拡張子は.binです。 オプションフォントデータについては、「ディスプレイ用技術説明書」のディスプレイコマンド「オプションフォントの登録」を参照してください。

例外	PrinterException 本メソッド呼び出し時にエラーが発生した場合は、 PrinterException をスローします。 エラーについての詳細は、「4.1.5 PrinterException クラス」を参照してください。 データ送信に失敗すると、デバイスとの通信を終了し、 PrinterException をスローすることがあります。 デバイスとの通信状態の確認は、 isConnect を参照してください。
説明	本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。 オプションフォントが登録済みの状態で本メソッドを実行した場合は、登録済みのオプションフォントは削除され、新たにメモリ領域を確保してオプションフォントを登録します。 登録済みのオプションフォントが削除されても使用済みのメモリは解放されません。使用済みのメモリは、 defragment を実行すると再度利用できます。 プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

unregisterOptionFont

オプションフォントの削除

登録済みのオプションフォントを削除します。

構文 `public void unregisterOptionFont() throws PrinterException`

例外	PrinterException 本メソッド呼び出し時にエラーが発生した場合は、 PrinterException をスローします。 エラーについての詳細は、「4.1.5 PrinterException クラス」を参照してください。 データ送信に失敗すると、デバイスとの通信を終了し、 PrinterException をスローすることがあります。 デバイスとの通信状態の確認は、 isConnect を参照してください。
説明	本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。 登録されているオプションフォントは全て削除されます。 オプションフォントを削除しても使用済みのメモリは解放されません。使用済みのメモリは、 defragment を実行すると再度利用できます。 プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

controlMacroRegistration

マクロ登録の開始・終了

マクロ登録の開始・終了を指定します。

構文 `public void controlMacroRegistration(int macroID,
MacroRegistrationFunction control) throws PrinterException`

パラメータ	macroID マクロID 有効範囲は-1～127です。 マクロID=120～126はシステムで利用していますので、指定しないでください。
-------	--

control

マクロ登録処理

利用可能な定数は、

「4.1.1(3)⑫ マクロ登録処理(MacroRegistrationFunction)」を参照してください。

例外

PrinterException

本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。

エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。

データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明

マクロ登録処理の手順は以下の通りです。

① マクロ登録処理を開始します。

*macroID*に-1を指定してください。

*control*に**MACRO_REGISTRATION_START**を指定してください。

② メソッドを実行します。

マクロ登録処理対象メソッドの場合、送信データのバッファリングを開始します。

バッファリング中に実行したマクロ登録処理対象メソッドの送信データは、デバイスへ送信せずにマクロデータバッファにバッファリングします。

バッファリングできる最大送信データのサイズは1024バイトです。

バッファリングされた送信データが最大サイズを超える場合、超えた時点のマクロ登録対象メソッドがエラーとなります。

エラーとなった場合は登録中のデータは破棄され、マクロモードが解除されます。

保持された送信データは、手順③でマクロ登録処理を終了してください。

マクロ登録処理対象以外のメソッドの場合は、送信データはバッファリングされずに即実行されます。

マクロ登録処理対象メソッドを下記に示します。

- **showTemplate**
- **showSlide**
- **selectTemplate**
- **setTemplateImageData**
- **selectTemplateTextObject**
- **setTemplateTextAlignment**
- **setTemplateTextLeftMargin**
- **setTemplateTextLineSpacing**
- **setTemplateTextBold**
- **setTemplateTextUnderline**
- **setTemplateTextSize**
- **setTemplateTextFont**
- **setTemplateTextRegisteredFont**
- **setTemplateTextRightSpacing**
- **setTemplateTextColor**
- **setTemplateTextData**
- **setTemplateBarcodeData**
- **setTemplateQrCodeData**

③ マクロ登録処理を終了します。

*macroID*に登録したいマクロID(0~127)を指定してください。

*control*にMACRO_REGISTRATION_REGISTを指定するとバッファリングされた送信データをデバイスへ送信します。バッファリングされた送信データはデバイス送信後も保持されます。本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

保持されている送信データは以下のいずれかにより破棄されます。

- ・MACRO_REGISTRATION_CLEARを指定する。
- ・MACRO_REGISTRATION_STARTを指定する。
- ・disconnectを実行する。

登録したマクロはexecuteMacroで実行できます。

登録済みのマクロを削除する手順は以下の通りです。

*control*にMACRO_REGISTRATION_STARTを指定し、*macroID*に-1を指定して本メソッドを呼び出してください。

*control*にMACRO_REGISTRATION_REGISTを指定し、*macroID*に削除するマクロIDを指定して本メソッドを呼び出してください。

本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

注意 工場出荷時に登録されるデータは、品質向上のため予告無く追加・変更される場合があります。

getDisplayResponse

ディスプレイからの各種応答取得

ディスプレイからの応答データを取得します。

構文 `public void getDisplayResponse(int id, int param, Object buf) throws PrinterException`

パラメータ	<i>id</i>	ディスプレイ応答種別定数 利用可能な定数は、「4.1.1(2)⑥ ディスプレイ応答種別」を参照してください。
	<i>param</i>	コマンドパラメータ ディスプレイ応答種別定数によりバッファの型が異なります。 指定する値の説明は、下表を参照してください。
	<i>buf</i>	取得した応答データを格納するバッファ <i>id</i> で指定した応答データを <i>buf</i> で指定されたオブジェクトに格納します。 ディスプレイ応答種別定数によりバッファの型が異なります。 バッファの型は、下表を参照してください。

応答種別定数	
パラメータ	説明
DISPLAY_RESPONSE_REQUEST (実行応答リクエスト)	
<i>param</i>	int型で0～15(00h～0Fh)を指定してください。
<i>buf</i>	長さ1のint型の配列を指定してください。 <i>buf[0]</i> には0～15(00h～0Fh)を指定してください。 応答取得成功時には、 <i>buf[0]</i> に実行応答リクエストの応答コードが64～79(40h～4Fh)で格納されます。
DISPLAY_RESPONSE_USER_AREA (ユーザ領域の残り容量の送信)	
<i>param</i>	int型で0を指定してください。
<i>buf</i>	長さ1のint型の配列を指定してください。 応答取得成功時には、ユーザ領域の残り容量(単位:バイト)が数値で格納されます。
DISPLAY_RESPONSE_TEMPLATE_ID_LIST (テンプレートIDの送信)	
<i>param</i>	int型で0を指定してください。
<i>buf</i>	ArrayList<Integer>型の配列を指定してください。 応答取得成功時には、登録済みのテンプレートIDがintの配列で格納されます。
DISPLAY_RESPONSE_IMAGE_ID_LIST (イメージIDの送信)	
<i>param</i>	int型で0を指定してください。
<i>buf</i>	ArrayList<Integer>型の配列を指定してください。 応答取得成功時には、登録済みのイメージIDがintの配列で格納されます。
DISPLAY_RESPONSE_SLIDE_ID_LIST (スライドIDの送信)	
<i>param</i>	int型で0を指定してください。
<i>buf</i>	ArrayList<Integer>型の配列を指定してください。 応答取得成功時には、登録済みのスライドIDがintの配列で格納されます。
DISPLAY_RESPONSE_TEMPLATE_LABEL (テンプレート名の送信)	
<i>param</i>	int型で0～127(00h～7Fh)を指定してください。
<i>buf</i>	ArrayList<String>型の配列を指定してください。 応答取得成功時には、テンプレートの登録時に指定したテンプレート名が文字列で格納されます。
DISPLAY_RESPONSE_IMAGE_LABEL (イメージ名の送信)	
<i>param</i>	int型で0～63(00h～3Fh)を指定してください。
<i>buf</i>	ArrayList<String>型の配列を指定してください。 応答取得成功時には、イメージデータの登録時に指定したイメージ名が文字列で格納されます。
DISPLAY_RESPONSE_SLIDE_LABEL (スライド名の送信)	
<i>param</i>	int型で0～91(00h～5bh)を指定してください。
<i>buf</i>	ArrayList<String>型の配列を指定してください。 応答取得成功時には、スライドデータの登録時に指定したスライド名が文字列で格納されます。

例外

PrinterException

本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。

エラーについての詳細は、「4.1.5 **PrinterException**クラス」を参照してください。

データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

4.1.2 PrinterEventクラス

PrinterEventクラスは、startDiscoveryPrinter、またはstartDiscoveryDeviceが終了した際に発生する終了イベントを取得するクラスです。

(1) メソッド一覧

PrinterEventクラスで提供されるメソッドを下記に示します。

名前	説明
getEventType	終了イベントの取得

(2) 終了イベント定数

終了イベントの取得で利用する定数を下記に示します。

定数名	説明	値
EVENT_FINISHED_DISCOVERY	startDiscoveryPrinter、またはstartDiscoveryDeviceの終了	1

(3) メソッド詳細

getEventType終了イベントの取得

startDiscoveryPrinter、またはstartDiscoveryDeviceが終了した際の終了イベントを取得します。

構文 `public int getEventType()`

戻り値 値については、「4.1.2(2) 終了イベント定数」を参照してください。

説明 終了イベントの判定条件を下記に示します。

- ・startDiscoveryPrinterが終了
- ・startDiscoveryDeviceが終了

デバイスが発見できなかった場合もEVENT_FINISHED_DISCOVERYが返ります。

4.1.3 PrinterListenerインターフェイス

PrinterListenerインターフェイスは、startDiscoveryPrinter、またはstartDiscoveryDeviceが終了した際の終了イベントを取得するためのインターフェイスです。

(1) メソッド一覧

PrinterListenerインターフェイスのメソッドを下記に示します。

名前	説明
finishEvent	デバイス探索の終了イベント

(2) メソッド詳細

finishEvent	デバイス探索の終了イベント
--------------------	----------------------

startDiscoveryPrinter、またはstartDiscoveryDeviceが終了した場合に呼び出される終了イベントです。

構文 public void **finishEvent(PrinterEvent event)**

パラメータ *event* 終了イベント
PrinterEventクラスで指定されます。

説明 インターフェイスのため実装はありません。
本メソッドをstartDiscoveryPrinter、またはstartDiscoveryDeviceの終了によって、終了イベントの通知を受け取るユーザアプリケーションに実装してください。PrinterEventクラスのgetEventTypeにより、終了イベントの種別を判定してください。

4.1.4 PrinterInfoクラス

PrinterInfoクラスは、**startDiscoveryPrinter**、または**startDiscoveryDevice**で発見されたデバイスのデバイス情報を格納するクラスです。

(1) メソッド一覧

ポート名を取得できます。**PrinterInfo**クラスで提供されるメソッドの一覧を下記に示します。

名前	説明
getDevicePath	デバイスパスの取得

(2) メソッド詳細

getDevicePath

デバイスパスの取得

startDiscoveryPrinter、または**startDiscoveryDevice**で発見されたデバイスのデバイス情報からUSBデバイスファイルのパスの文字列を取得します。

構文 `public String getDevicePath()`

戻り値 デバイスパス

4.1.5 PrinterExceptionクラス

(1) メソッド一覧

PrinterExceptionクラスで提供されるメソッドの一覧を下記に示します。

名前	説明
PrinterException	コンストラクタ
getErrorCode	エラーコードの取得

(2) 定数一覧

① エラーコード

エラーコードの取得で利用される定数を下記に示します。

定数名	説明	値
ERROR_ACCESS_DENIED	ハンドルの取得に失敗しました。 ^{*1}	-1
	利用できないポートが指定されました。	
	サポートされていないメソッドが指定されました。	
ERROR_SHARING_VIOLATION	すでにオープン済みのポートが指定されました。	-11
ERROR_PORT_NOT_OPENED	ポートがオープンされていません。	-12
ERROR_DEVICE_NOT_CONNECTED	AndroidデバイスとプリンターのBluetooth接続に問題があります。	-21
	AndroidデバイスとプリンターのUSB接続に問題があります。	
ERROR_OFFLINE	切断状態またはプリンターがオフラインです。	-22
ERROR_EXTERNAL_DEVICE_NOT_CONNECTED	ディスプレイが接続されていません。	-23
ERROR_DEVICE_INITIALIZE_FAILED	プリンターの設定変更に失敗しました。プリンターへのデータ送信が送信タイムアウト時間内に完了してないか、プリンターからのデータ受信が受信タイムアウト時間内に完了していません。	-31
ERROR_DATA_SIZE_ZERO	データサイズが0バイトのデータが指定されました。	-101
ERROR_OVER_MAX_DATA_SIZE	データサイズが最大値を超過しています。	-102
ERROR_DATA_SIZE_INVALID	既定のデータサイズではありません。	-103
ERROR_ENCODE_FAILED	テキストデータのエンコードでエラーが発生しました。 ^{*1}	-111
ERROR_TIMEOUT	送信タイムアウトが発生しました。	-201
	受信タイムアウトが発生しました。	
ERROR_FILE_NOT_FOUND	指定されたファイルが存在しません。	-301
ERROR_FILE_USED	指定されたファイルは別プロセスで使用中です。	-302
ERROR_FILE_INVALID	指定されたファイルは無効です。	-303

定数名	説明	値
ERROR_LOW_MEMORY	イメージデータファイルの読み込みでメモリ不足が発生しました。	-311
ERROR_OVER_MAX_IMAGE	イメージデータの幅と高さのいずれか、または両方が印字可能な最大ドット数を超過しています。	-312
ERROR_LOGO_NOT_DEFINED	ロゴが登録されていません。	-313
ERROR_LOW_USER_AREA	ユーザ領域の残り容量が不足しています。	-401
ERROR_LOW_EXTERNAL_RAM	RAMの残り容量が不足しています。	-402
ERROR_NOT_REGISTERD	テンプレートが登録されていません。 イメージデータが登録されていません。 スライドデータが登録されていません。 オプションフォントが登録されていません。 外字が登録されていません。	-403
ERROR_NOT_UNREGISTERD	テンプレートが削除されていません。 イメージデータが削除されていません。 スライドデータが削除されていません。 オプションフォントが削除されていません。 外字が削除されていません。	-404
ERROR_INVALID_NO	指定されたロゴIDの値は無効です。	-501
ERROR_INVALID_DATA	指定されたデータは無効です。	-503
ERROR_INVALID_PARAM	指定されたパラメータは無効です。	-9999

*1: 異常処理の可能性があります。

(3) メソッド詳細

PrinterException コンストラクタ

com.seikoinstruments.sdk.thermalprinter.PrinterExceptionクラスのコンストラクタです。

構文 `public PrinterException(int code, String message)`

getErrorCode エラーコードの取得

スローされた例外のエラーコードを取得します。

構文 `public int getErrorCode()`

戻り値 エラーコードの内容は、「4.1.5(2) 定数一覧」を参照してください。

5章 サンプルプログラム

本章では、SIIプリントクラスライブラリで提供するサンプルプログラムについて説明します。

SIIプリントクラスライブラリには、Android Studioのプロジェクト形式のサンプルプログラムが含まれています。

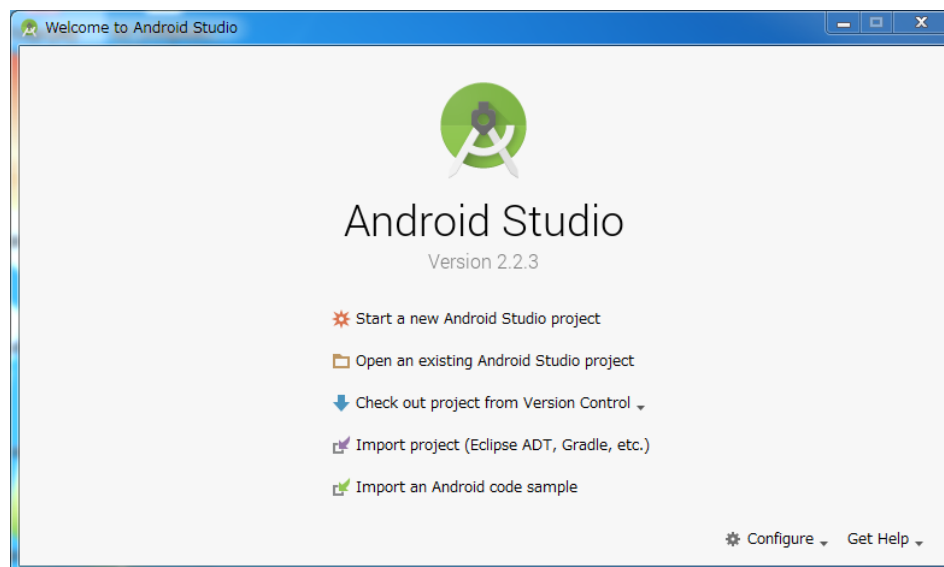
5.1 インストール

サンプルプログラムをインストールしてください。

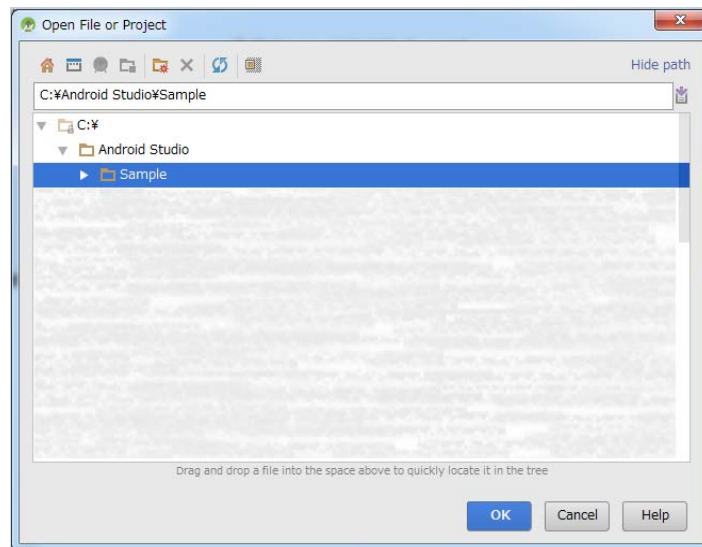
Androidアプリケーションの開発環境が整っていることを確認してください。必要な開発環境については、「3章 ライブラリの利用方法」を参照してください。

インストール手順は以下の通りです。

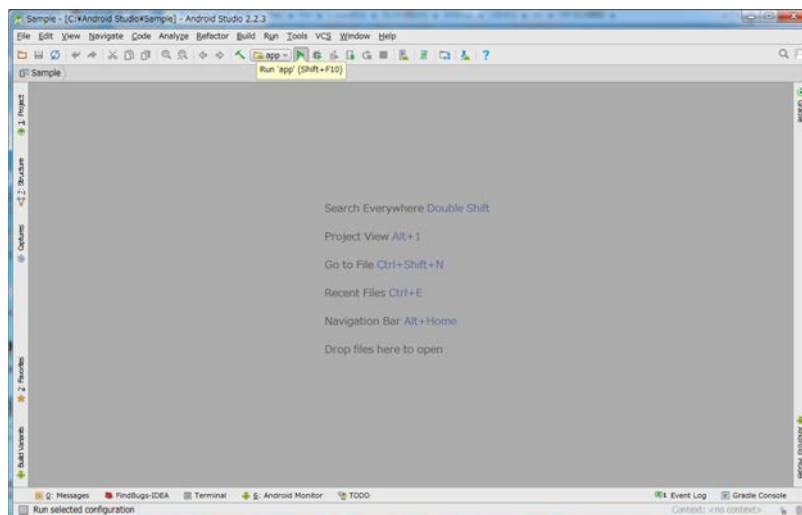
- (1) Sampleフォルダを任意の位置に配置してください。
- (2) Android Studioを起動し、「Open an existing Android Studio project」をクリックしてください。



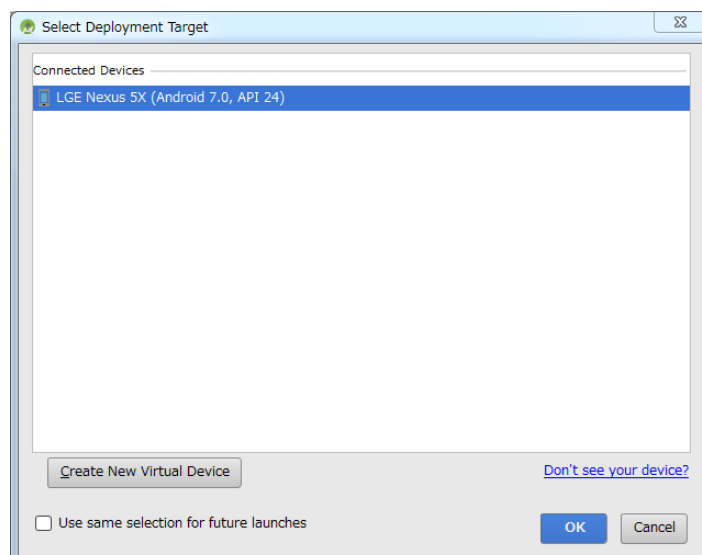
(3) 手順(1)で配置したフォルダを選択し、[OK]をクリックしてください。



(4) [Run 'app']をクリックしてください。



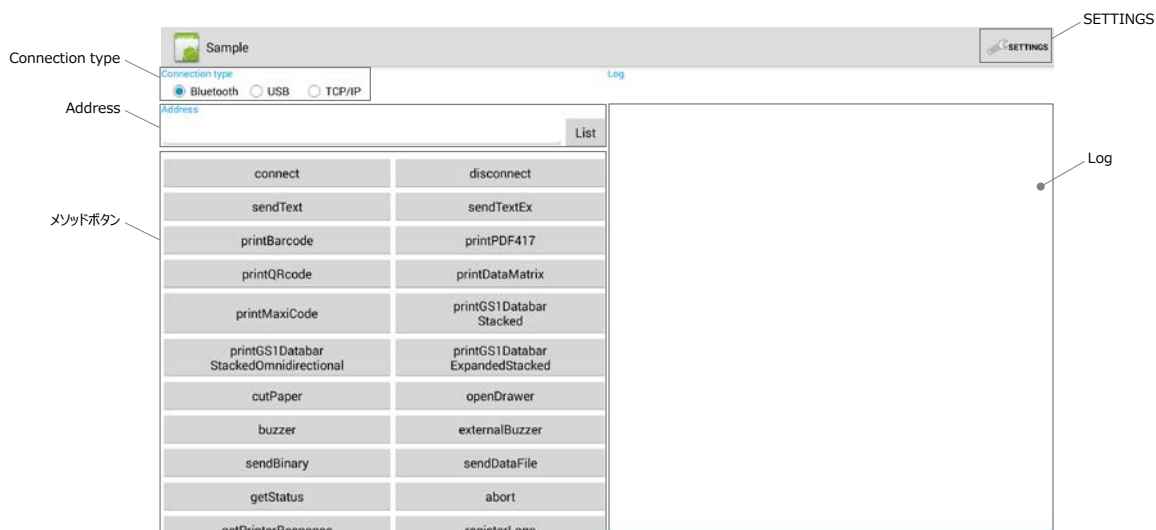
(5) デバイスを選択し、[OK]をクリックしてください。



5.2 画面構成

サンプルプログラムの画面について説明します。

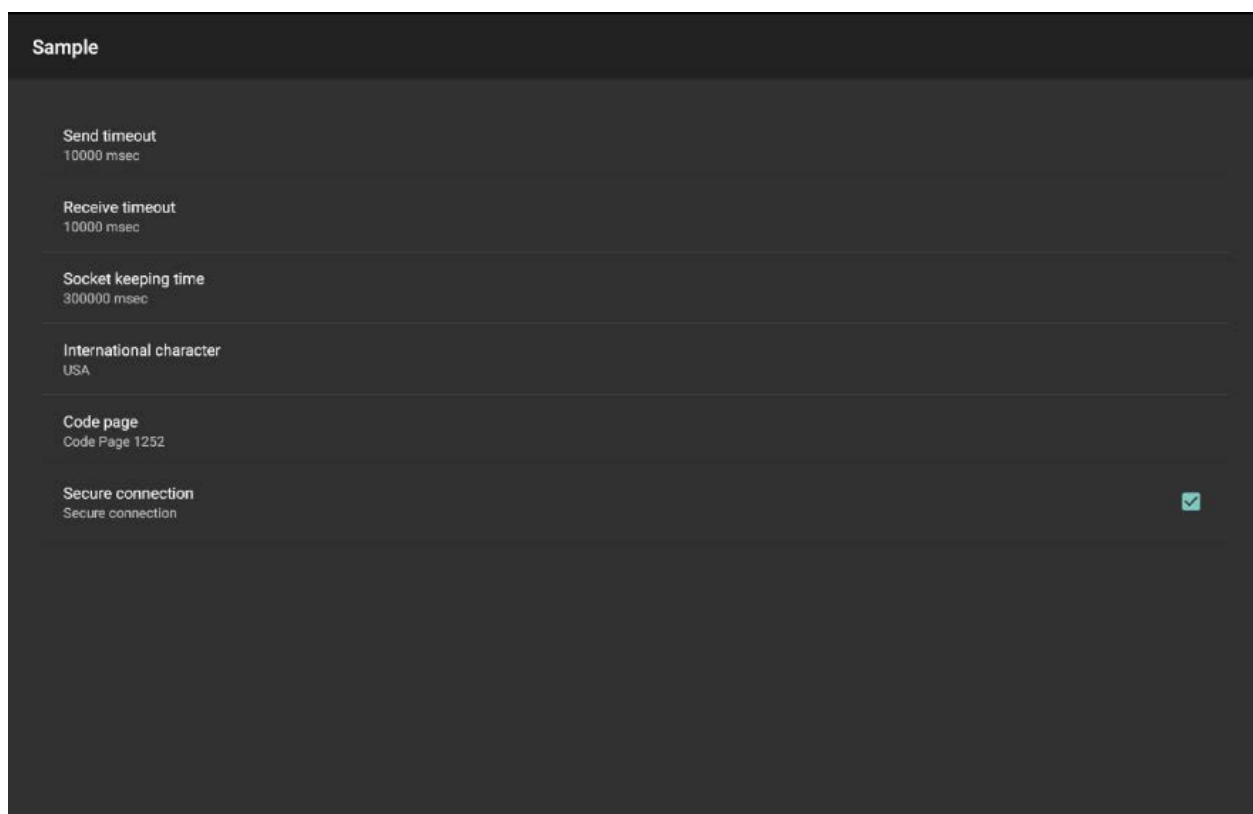
5.2.1 メイン画面



項目	説明
Connection type	デバイスへの接続種別を選択します。
Address	プリンター経由で使用する場合のみ有効です。 プリンターのアドレスを指定します。 手動入力の場合： Bluetoothで接続する場合は、Bluetoothアドレスを入力してください。 TCP/IPで接続する場合はMACアドレスを入力してください。 自動入力の場合： [List]ボタンをタップすると、発見されたプリンターのプリンター情報が一覧で表示されます。 表示された一覧の中からプリンターを選択すると、Bluetoothアドレスが自動入力されます。
メソッドボタン	各メソッドを実行できるボタンです。 スクロールすると、表示されないメソッド及びプロパティを見ることができます。 各メソッドの詳細については、「4章 ライブラリの機能」を参照してください。
SETTINGS	[SETTINGS]ボタンをタップすると、機能設定画面が開きます。 メイン画面に戻る場合は、画面上の◀をタップしてください。
Log	「メソッドボタン」の各メソッドを実行すると、メソッド実行ログが表示されます。

5.2.2 [SETTINGS]画面

[SETTINGS]では各種設定機能を表示します。



The screenshot displays a settings interface with a dark background. At the top, the word "Sample" is written in white. Below it, several settings are listed, each with a label and a value, separated by horizontal lines. The settings are: "Send timeout" with a value of "10000 msec", "Receive timeout" with a value of "10000 msec", "Socket keeping time" with a value of "300000 msec", "International character" with a value of "USA", "Code page" with a value of "Code Page 1252", and "Secure connection" with a value of "Secure connection" and a green checkmark icon to its right.

Setting	Value
Send timeout	10000 msec
Receive timeout	10000 msec
Socket keeping time	300000 msec
International character	USA
Code page	Code Page 1252
Secure connection	Secure connection <input checked="" type="checkbox"/>

5.3 注意事項

サンプルプログラムは予告なしに変更される場合があります。

サンプルプログラムに対しての動作保証及びサポートは行っておりません。

付録 A 文字セット

A.1 コードページ表(文字コードテーブル)

国際文字セットでCOUNTRY_USA設定時のコードページを下記に示します。
 設定する国際文字セットにより、特定の文字コードの表示結果が異なります。
 特定の文字コードについては、「A.2 国際文字セット」を参照してください。

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80	Ç	ü	é	â	ä	à	å	ç	ê	ë	è	ï	î	ì	Ä	Å
90	É	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	φ	£	¥	℔	ƒ
A0	á	í	ó	ú	ñ	Ñ	ä	ö	¿	¬	½	¼	¿	»	«	»
B0	☐	☐	☐		†	‡	§	¶	§	¶	§	¶	§	¶	§	¶
C0	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞
D0	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞
E0	α	β	Γ	π	Σ	σ	μ	τ	φ	θ	Ω	δ	∞	φ	ε	∩
F0	≡	±	≥	≤	∫	∫	÷	≈	°	•	•	√	n	2	■	■

図 A-1 CODE_PAGE_437 (USA, Standard Europe)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80																
90																
A0	。	「	」	、	・	ヲ	ア	イ	ウ	エ	オ	ヤ	ユ	ヨ	ッ	
B0	ー	ア	イ	ウ	エ	オ	カ	キ	ク	ケ	コ	サ	シ	ス	セ	ソ
C0	タ	チ	ツ	テ	ト	ナ	ニ	ヌ	ネ	ノ	ハ	ヒ	フ	ヘ	ホ	マ
D0	ミ	ム	メ	モ	ヤ	ユ	ヨ	ラ	リ	ル	レ	ロ	ワ	ン	ゝ	。
E0																
F0																

図 A-2 CODE_PAGE_KATAKANA

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80	Ç	ü	é	â	ä	à	â	ç	ê	ë	è	ï	î	ì	Ä	Å
90	É	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	ø	£	Ø	×	f
A0	á	í	ó	ú	ñ	Ñ	ä	ö	¿	®	¬	½	¼	¡	«	»
B0	☐	☐	☐			Á	Â	À	©	¶	¶	¶	¶	¶	¥	₱
C0	⌞	⌞	⌞	⌞	⌞	ã	Ã	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞
D0	ð	Đ	Ê	Ë	È	Í	Î	Ï	⌞	⌞	⌞	⌞	⌞	⌞	⌞	⌞
E0	Ó	β	Ô	Ò	Õ	μ	þ	þ	Ú	Û	Ü	Ý	Ý	Ý	Ý	Ý
F0	-	±	=	¾	¶	§	÷	,	°	…	.	¹	³	²	■	■

図 A-3 CODE_PAGE_850 (Multilingual)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	!	”	#	\$	%	&	'	()	*	+	,	-	.	/	
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80	Ç	ü	é	â	ã	à	Á	ç	ê	Ê	è	Í	Ô	ì	Ã	Â
90	É	À	È	ô	õ	ò	Ú	ù	Ì	Õ	Ü	¢	£	Ù	Þ	Ó
A0	á	í	ó	ú	ñ	Ñ	ä	ö	í	ò	¬	½	¼	¡	«	»
B0	☐	☐	☐													
C0	L	L	T		-	+	+	+	+	+	+	+	+	+	+	+
D0	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌
E0	α	β	Γ	π	Σ	σ	μ	τ	φ	θ	Ω	δ	∞	φ	ε	Π
F0	≡	±	≥	≤		J	÷	≈	°	•	•	√	n	2	■	

图 A-4 CODE_PAGE_860 (Portuguese)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	!	”	#	\$	%	&	'	()	*	+	,	-	.	/	
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80	Ç	ü	é	â	Â	à	¶	ç	ê	ë	è	ï	î	≡	À	§
90	É	È	Ê	ô	Ë	Ï	Ô	Ù	⌘	Ô	Ü	¢	£	Ù	Ô	f
A0		'	ó	ú	..	³	-	î	¬	¬	½	¼	¾	«	»	
B0	☐	☐	☐													
C0	L	L	T		-	+	+	+	+	+	+	+	+	+	+	+
D0	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌
E0	α	β	Γ	π	Σ	σ	μ	τ	φ	θ	Ω	δ	∞	φ	ε	Π
F0	≡	±	≥	≤		J	÷	≈	°	•	•	√	n	2	■	

图 A-5 CODE_PAGE_863 (Canadian-French)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80	Ç	ü	é	â	ä	à	å	ç	ê	ë	è	ï	î	ì	Ä	Å
90	É	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	ø	£	Ø	Pt	f
A0	á	í	ó	ú	ñ	Ñ	ä	ö	¿	¬	½	¼	í	«	»	
B0	☐	☐	☐													
C0	L	L	T													
D0	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌
E0	α	β	Γ	π	Σ	σ	μ	τ	φ	θ	Ω	δ	∞	φ	ε	Π
F0	≡	±	≥	≤		J	÷	≈	°	.	.	√	n	2	■	

☒ A-6 CODE_PAGE_865 (Nordic)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80	Ç	ü	é	â	ä	à	å	ç	ê	ë	è	ï	î	ì	Ä	Å
90	É	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	ø	£	Ø	Ş	ş
A0	á	í	ó	ú	ñ	Ñ	Ğ	ğ	¿	®	¬	½	¼	í	«	»
B0	☐	☐	☐													
C0	L	L	T													
D0	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌	⌌
E0	ó	β	ô	ò	õ	õ	μ		×	ú	û	ü	ì	ÿ	-	'
F0	-	±	¾	¶	§	÷	,		°	..	.	1	3	2	■	

☒ A-7 CODE_PAGE_857 (Turkish)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80	A	B	Γ	Δ	E	Z	H	Θ	I	K	Λ	M	N	Ξ	O	Π
90	P	Σ	T	Υ	Φ	X	Ψ	Ω	α	β	γ	δ	ε	ζ	η	θ
A0	ι	κ	λ	μ	ν	ξ	ο	π	ρ	σ	ς	τ	υ	φ	χ	ψ
B0	⌘	⌘	⌘		†	‡	§	¶	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
C0	L	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
D0	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
E0	ω	ά	έ	ή	ϊ	ί	ό	ύ	ϋ	ώ	À	É	Η	Ι	Ο	Υ
F0	Ω	±	≥	≤	İ	ÿ	÷	≈	°	•	•	√	n	2		

图 A-8 CODE_PAGE_737 (Greek)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80	€	‚	ƒ	„	…	†	‡	^	‰	Š	‹	Œ		Ž		
90		‘	’	“	”	•	-	-	~	™	š	›	œ	ž	ÿ	
A0		ı	¢	£	¤	¥	¦	§	¨	©	ª	«	¬	®	¯	
B0	°	±	²	³	´	µ	¶	·		¹	º	»	¼	½	¾	¿
C0	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D0	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
E0	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F0	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

图 A-9 CODE_PAGE_1252 (Latin)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80	А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П
90	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
A0	а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п
B0	␣	␣	␣													
C0	␣	␣	␣	␣	␣	␣	␣	␣	␣	␣	␣	␣	␣	␣	␣	␣
D0	␣	␣	␣	␣	␣	␣	␣	␣	␣	␣	␣	␣	␣	␣	␣	␣
E0	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я
F0	Ё	ё	Є	є	İ	ı	Ÿ	ÿ	°	•	•	√	№	α	■	

☒ A-10 CODE_PAGE_866 (Russian)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80	Ç	ü	é	â	ä	û	ç	ł	ë	ő	ö	î	ž	Ä	Ć	
90	É	Í	í	ô	ö	Ł	ł	Ś	ś	Ö	Ü	Ť	ť	Ł	×	č
A0	á	í	ó	ú	À	à	Ž	ž	Ę	ę	¬	ž	Č	š	«	»
B0	␣	␣	␣			Á	Â	Ě	Š					Ž	ž	ı
C0	␣	␣	␣	␣	␣	Ä	ä	Ł	ł	Ł	ł	Ł	ł	Ł	Ł	α
D0	đ	Đ	Ď	Ě	ď	Ň	í	î	ě	ı	ı	ı	ı	ı	ı	ı
E0	ó	ß	ô	ń	ň	š	š	ř	ú	ř	ú	ý	ý	ı	ı	ı
F0	-	"	˘	˘	˘	§	÷	°	°	°	°	ı	Ř	ř	ı	ı

☒ A-11 CODE_PAGE_852 (Eastern Europe)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80	Ç	ü	é	â	ä	à	â	ç	ê	ë	è	ï	î	ì	Ä	Å
90	É	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	ø	£	Ø	×	f
A0	á	í	ó	ú	ñ	Ñ	ä	ö	¿	®	¬	½	¼	¡	«	»
B0	☐	☐	☐			Á	Â	Ã	©			¶	¶	¢	¥	₱
C0	L	L	T	T	+	+	ã	Ã	ℓ	ℓ	ℓ	ℓ	ℓ	=	ℓ	α
D0	ð	Ð	Ê	Ë	È	€	Í	Î	Ï	Ј	Г	■	■	І	İ	■
E0	ó	β	ô	ò	õ	õ	μ	þ	þ	ú	û	ü	ý	ý	-	'
F0	-	±	=	¾	¶	§	÷	,	°	..	.	1	3	2		■

☒ A-12 CODE_PAGE_858 (Euro)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80	ђ	Ђ	ѓ	Ѓ	ё	Ё	є	Є	ѕ	Ѕ	і	І	ї	Ї	ј	Ј
90	љ	Љ	њ	Њ	ћ	Ћ	ќ	Ќ	џ	Џ	џ	џ	џ	џ	џ	џ
A0	а	А	б	Б	в	В	г	Г	д	Д	е	Е	ф	Ф	г	Г
B0	☐	☐	☐			x	X	и	И			¶	¶	й	Й	₱
C0	L	L	T	T	+	+	к	К	ℓ	ℓ	ℓ	ℓ	ℓ	=	ℓ	α
D0	л	Л	м	М	н	Н	о	О	п	П	Г	■	■	П	я	■
E0	Я	Р	Р	с	С	т	Т	у	У	ж	Ж	в	В	ь	ь	№
F0	-	ы	Ы	э	Э	ш	Ш	э	Э	щ	Щ	ч	Ч	§		■

☒ A-13 CODE_PAGE_855 (Cyrillic)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80	°	•	•	√	■	-		+	+	+	+	+	+	+	+	+
90	β	∞	φ	±	½	¼	≈	《	》	لَا	لَا	لَا	لَا	لَا	لَا	لَا
A0	-	ل	ل	ل	ل	ل	ل	ل	ل	ل	ل	ل	ل	ل	ل	ل
B0	•	١	٢	٣	٤	٥	٦	٧	٨	٩	ف	س	س	س	س	س
C0	¢	ء	آ	أ	ؤ	ع	ئ	ب	ة	ث	ج	ح	خ	د	ذ	ذ
D0	ذ	ز	ر	س	ش	ص	ض	ط	ظ	ع	غ	ف	ق	ك	م	م
E0	-	ف	ق	ك	م	ل	م	ل	م	ل	م	ل	م	ل	م	ل
F0	-	ن	ه	ه	ي	ي	ي	ي	ي	ي	ي	ي	ي	ي	ي	ي

图 A-14 CODE_PAGE_864 (Arabic)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80	€	;	;	;	;	;	;	;	;	;	;	;	;	;	;	;
90	€	;	;	;	;	;	;	;	;	;	;	;	;	;	;	;
A0	~	~	~	~	~	~	~	~	~	~	~	~	~	~	~	~
B0	°	±	±	±	±	±	±	±	±	±	±	±	±	±	±	±
C0	Á	Á	Á	Á	Á	Á	Á	Á	Á	Á	Á	Á	Á	Á	Á	Á
D0	Đ	Đ	Đ	Đ	Đ	Đ	Đ	Đ	Đ	Đ	Đ	Đ	Đ	Đ	Đ	Đ
E0	á	á	á	á	á	á	á	á	á	á	á	á	á	á	á	á
F0	đ	đ	đ	đ	đ	đ	đ	đ	đ	đ	đ	đ	đ	đ	đ	đ

图 A-15 CODE_PAGE_1250 (Central European)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80	ђ	ѓ	;	ѓ	„	...	†	‡	€	‰	Љ	<	Њ	ќ	ћ	џ
90	ђ	‘	;	“	”	•	-	-	™	љ	>	њ	ќ	ћ	џ	
A0	ѳ	ѵ	Ј	Ѡ	Г	І	Ѕ	Ё	©	Є	«	¬	-	®	İ	
B0	°	±	І	і	г	μ	¶	•	ё	№	є	»	ј	Ѕ	ѕ	ї
C0	А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П
D0	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
E0	а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п
F0	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я

☒ A-16 CODE_PAGE_1251 (Cyrillic)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80	€	‘	;	ƒ	„	...	†	‡	‰		<					
90		‘	;	“	”	•	-	-	™		>					
A0	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ
B0	°	±	²	³	´	μ	¶	•	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ
C0	ı	Α	Β	Γ	Δ	Ε	Ζ	Η	Θ	Ι	Κ	Λ	Μ	Ν	Ξ	Ο
D0	Π	Ρ		Σ	Τ	Υ	Φ	Χ	Ψ	Ω	İ	ÿ	ά	έ	ή	ί
E0	ˆ	α	β	γ	δ	ε	ζ	η	θ	ι	κ	λ	μ	ν	ξ	ο
F0	π	ρ	ς	σ	τ	υ	φ	χ	ψ	ω	ï	ÿ	ό	ύ	ώ	

☒ A-17 CODE_PAGE_1253 (Greek)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80	€	‘	’	“	”	•	-	-	~	™	š	<	Œ			
90		‘	’	“	”	•	-	-	~	™	š	>	œ			ÿ
A0	ı	¢	£	¤	¥	¦	§	¨	©	ª	«	¬	®	¯		
B0	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
C0	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D0	Ğ	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	İ	Ş	ß
E0	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F0	ğ	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ı	ş	ÿ

☒ A-18 CODE_PAGE_1254 (Turkish)

A.2 国際文字セット

設定する国際文字セットにより、特定の文字コードの表示結果が異なります。

特定の文字コード、及びその表示結果を下記に示します。

	23	24	40	5B	5C	5D	5E	60	7B	7C	7D	7E
COUNTRY_USA	#	\$	@	[\]	^	`	{		}	~
COUNTRY_FRANCE	#	\$	à	°	ç	§	^	`	é	ù	è	..
COUNTRY_GERMANY	#	\$	§	Ä	Ö	Ü	^	`	ä	ö	ü	ß
COUNTRY_ENGLAND	£	\$	@	[\]	^	`	{		}	~
COUNTRY_DENMARK_1	#	\$	@	Æ	Ø	Å	^	`	æ	ø	å	~
COUNTRY_SWEDEN	#	α	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü
COUNTRY_ITALY	#	\$	@	°	\	é	^	ù	à	ò	è	ì
COUNTRY_SPAIN	Pt	\$	@	¡	Ñ	¿	^	`	..	ñ	}	~
COUNTRY_JAPAN	#	\$	@	[¥]	^	`	{		}	~
COUNTRY_NORWAY	#	α	É	Æ	Ø	Å	Ü	é	æ	ø	å	ü
COUNTRY_DENMARK_2	#	\$	É	Æ	Ø	Å	Ü	é	æ	ø	å	ü
COUNTRY_SPAIN_2	#	\$	á	¡	Ñ	¿	é	`	í	ñ	ó	ú
COUNTRY_LATIN_AMERICA	#	\$	á	¡	Ñ	¿	é	ü	í	ñ	ó	ú
COUNTRY_ARABIA	#	\$	@	[\]	^	`	{		}	~

図 A-19 国際文字セット



セイコーインスツル株式会社

プリントシステム事業部

千葉県千葉市美浜区中瀬 1-8 〒261-8507

電話番号：043-211-1212（直通） ファクシミリ：043-211-8037

ホームページ：<http://www.sii-ps.com>

大阪営業所

大阪府大阪市北区豊崎 3-2-1 淀川 5 番館 〒531-0072

電話番号：06-7711-0858（直通） ファクシミリ：06-7711-0856

©本書の内容は、製品の改良に伴い、予告なしに変更することがあります。