DPU-S245, DPU-S445,
RP-D10, RP-E10
Print Class Library for iOS
Application Programmer's Guide

U00134990822

Seiko Instruments Inc.

U00134990800	January 2015
U00134990801	March 2015
U00134990802	August 2015
U00134990803	August 2015
U00134990804	February 2016
U00134990805	December 2016
U00134990806	April 2017
U00134990807	January 2018
U00134990808	February 2018
U00134990809	February 2019
U00134990810	July 2019
U00134990811	October 2019
U00134990812	March 2020
U00134990813	June 2020
U00134990814	September 2020
U00134990815	March 2022
U00134990816	October 2022
U00134990817	December 2022
U00134990818	April 2023
U00134990819	February 2024
U00134990820	March 2024
U00134990821	January 2025
U00134990822	June 2026

Copyright © 2015-2026 by Seiko Instruments Inc. All rights reserved.

App Store℠ is a service mark of Apple Inc.
iPad®, iPad Air®, iPad mini®, iPhone®, and iPod® are trademarks of Apple Inc., registered in the U.S. and other countries and regions.
The Bluetooth® word mark and logos are registered trademarks owned by Bluetooth SIG, Inc.
IOS is a trademark or registered trademark of Cisco in the U.S. and other countries and is used under license.

All other trademarks are the properties of their respective companies.

Seiko Instruments Inc. (hereinafter referred to as "SII") has prepared this manual for use by SII personnel, licensees, and customers. The information contained herein is the property of SII and shall not be reproduced in whole or in part without the prior written approval of SII.

SII reserves the right to make changes without notice to the specifications and materials contained herein and shall not be responsible for any damages (including consequential) caused by reliance on the materials presented, including but not limited to typographical, arithmetic, or listing errors.

Introduction

This manual describes the Print Class Library for iOS SDK (hereinafter referred to as "the SDK") for the printers provided by Seiko Instruments Inc. (hereinafter referred to as "SII").

Target Printers

This section lists the printers supported by the SDK.

	Description in This Manual	Interface	Printer
Mobile Printer	DPU-S245	Bluetooth	DPU-S245-01C-E
	DPU-S445	Bluetooth	DPU-S445-01C-E
POS Printer	RP-D10	Bluetooth	RP-D10-x27J2-B
		Ethernet	RP-D10-x27J1-E
	RP-E10	Bluetooth	RP-E10-x3FJ2-B
			RP-E11-x3FJ2-B
		Ethernet	RP-E10-x3FJ1-E
			RP-E11-x3FJ1-E

Use main firmware version 1.05 or later, LAN interface firmware version 1.13.01 or later for RP-D10-x27J1-E.

Use main firmware version 1.11 or later, LAN interface firmware version 1.13.01 or later for RP-E10-x3FJ1-E or RP-E11-x3FJ1-E.

RP-E10-x3FJ2-B and RP-E11-x3FJ2-B are the products destined only for Japan.

Table of Contents

Chapter 1 Product Overview 1-1

1.1 Function Provided by SDK 1-1
1.2 SII Print Class Library Overview 1-1
1.2.1 SII Print Class Library Configuration 1-1
1.2.2 Function Provided by Library 1-2
1.2.3 Development of Application that Performs Bluetooth Communication with SII Printer 1-2

Chapter 2 Product Specification 2-1

2.1 Operating Environment 2-1
2.1.1 Applicable iDevice Models 2-1
2.1.2 Applicable OS Versions 2-2
2.1.3 Operating Conditions 2-2
2.1.4 Precaution 2-3

Chapter 3 How to Use library 3-1

3.1 Development Environment for Android Application 3-1
3.2 Provided Files 3-2
3.3 Build Library into Xcode Project 3-3
3.3.1 Objective-C 3-3
3.3.2 Swift 3-7

Chapter 4 Function of Library 4-1

4.1 Standard Mode and Page Mode 4-1
4.1.1 Basic Operation 4-1
(1) Standard mode 4-1
(2) Page mode 4-2
4.1.2 Text Data Printing in Standard Mode 4-3
4.1.3 Mapping Position of Print Data in Page Mode 4-4
(1) Print area of page mode 4-4
(2) Print direction 4-4
(3) Reference point 4-5
4.1.4 Print Data Process at Out of Print Area of Page Mode 4-6
4.2 Log File Output Function 4-7
4.2.1 How to Set Log Output 4-7
4.2.2 Log Output Settings 4-7
4.2.3 Log File 4-7
4.3 API Reference 4-8
4.3.1 SIIPrinterManager Class 4-8
(1) Method List 4-8

(a) Common method to standard mode and page mode....4-9
(b) Dedicated method for standard mode....4-10
(c) Dedicated method for page mode....4-10
(2) Common property list to standard mode and page mode....4-11
(3) Constant List....4-12
(a) Printer model constant....4-12
(b) Connecting port type constant....4-12
(c) Response type constant....4-12
(d) International character setting constant....4-13
(e) Codepage constant....4-13
(f) Constant used for barcode and PDF417....4-14
(4) Constant List of Enumerated Type....4-15
(a) Drawer number (DrawerNum)....4-15
(b) Activation pulse width (PulseWidth)....4-15
(c) Dithering (Dithering)....4-15
(d) Batch processing selection (TransactionFunction)....4-16
(e) Bold print (CharacterBold)....4-16
(f) Underline (CharacterUnderline)....4-16
(g) Reverse print (CharacterReverse)....4-16
(h) Inversion print (CharacterInversion)....4-17
(i) Character font (CharacterFont)....4-17
(j) Character Scale (CharacterScale)....4-17
(k) Alignment (PrintAlignment)....4-18
(l) Barcode symbol (BarcodeSymbol)....4-18
(m) Module size (ModuleSize)....4-19
(n) HRI character print position (HriPosition)....4-20
(o) N:W ratio (NwRatio)....4-20
(p) Error correction level (ErrorCorrection)....4-21
(q) PDF417 symbol (Pdf417Symbol)....4-21
(r) QR code model (QrModel)....4-22
(s) Data Matrix Module (DataMatrixModule)....4-22
(t) MaxiCode Mode (MaxiCodeMode)....4-23
(u) Cutting method (CuttingMethod)....4-23
(v) Image rotation direction (Rotate)....4-24
(w) Image scaling (ImageScale)....4-24
(x) Print direction (Direction)....4-24
(y) Line style (LineStyle)....4-25
(5) Method Details....4-26
(a) Common method to standard mode and page mode....4-26
init Instance....4-26
connect Start communicating with a printer (Bluetooth)....4-26
connect Start communicating with a printer (TCP/IP)....4-27
disconnect Disconnect a printer....4-27
openDrawer Open cash drawer....4-28
buzzer Sound buzzer....4-28
externalBuzzer Sound external buzzer....4-29
getStatus Obtain printer status....4-29

abort Abort the waiting state of a printer 4-30
registerLogo Register logo (image) to a printer 4-31
unregisterLogo
Delete specified logo (image) on a printer 4-31
registerStyleSheet
Register style sheet to a printer 4-32
unregisterStyleSheet
Delete specified style sheet on a printer 4-32
resetPrinter Printer hardware reset 4-32
getPrinterResponse
Obtaining various responses from a printer 4-33
startDiscoveryPrinter
Start printer search (Bluetooth) 4-35
startDiscoveryPrinter
Start printer search (TCP/IP) 4-36
cancelDiscoveryPrinter
Cancel printer search 4-36
getFoundPrinter
Obtain searched printer information 4-37
getVersion Get SDK version 4-37
controlTransaction
Start/End batch processing 4-37
(b) Dedicated method for standard mode 4-39
sendText Send text data 4-39
sendTextEx Send format specified text data 4-39
sendTextEx Send format specified text data 4-40
printBarcode Print barcode 4-41
printPDF417 Print PDF417 4-44
printQRcode Print QR code 4-45
printDataMatrix
Print Data Matrix 4-46
printMaxiCode
Print MaxiCode 4-46
printGS1DataBarStacked
Print GS1 Databar Stacked 4-47
printGS1DataBarStackedOmnidirectional
Print GS1 Databar Stacked Omni-directional 4-47
printGS1DataBarExpandedStacked
Print GS1 Databar Expanded Stacked 4-47
printAztecCode
Print Aztec Code 4-47
cutPaper Cut paper 4-48
feedPosition Paper form feed 4-48
sendBinary Send binary data 4-48
sendDataFile Send specified file 4-48
sendImage Send image 4-50
printPDF Print PDF page 4-50
printLogo Print specified logo (image) on printer 4-52

printLogo Print specified logo (image) on printer 4-52
(c) Dedicated method for page mode 4-53
enterPageMode
Start page mode 4-54
exitPageMode End page mode 4-54
setPageModeArea
Specify print area of page mode 4-54
setPageModeDirection
Specify print direction of page mode 4-56
setPageModeLineSpacing
Specify line spacing of page mode 4-56
printPageMode
Print page mode 4-56
printPageModeText
Send text data of page mode 4-57
printPageModeTextEx
Send format specified text data of page mode 4-57
printPageModeBarcode
Print barcode of page mode 4-58
printPageModePDF417
Print PDF417 of page mode 4-60
printPageModeQRcode
Print QR Code of page mode 4-62
printPageModeDataMatrix
Print Data Matrix of page mode 4-63
printPageModeMaxiCode
Print MaxiCode of page mode 4-63
printPageModeGS1DataBarStacked
Print GS1 Databar Stacked of page mode 4-64
printPageModeGS1DataBarStackedOmnidirectional
Print GS1 Databar Stacked Omni-directional of page mode 4-64
printPageModeGS1DataBarExpandedStacked
Print GS1 Databar Expanded Stacked of page mode 4-65
printPageModeAztecCode
Print Aztec Code of page mode 4-65
sendPageModeBinary
Send binary data of page mode 4-65
printPageModeImageFile
Draw Image file of page mode 4-66
printPageModeRectangle
Draw rectangle image of page mode 4-66
printPageModeLine
Print ruled line of page mode 4-67
printPageModeLogo
Print logo of page mode 4-69
(6) Common property detail to standard mode and page mode 4-70
endTimeout Timeout period when sending data 4-70

receiveTimeout
Timeout period when receiving data....4-70
internationalCharacter
Set international character set....4-70
codePage Code page....4-71
printerModel Obtain printer model....4-71
portType Connecting port type....4-71
isConnect Verify connection state with a printer....4-71
socketKeepingTime
Socket keeping time....4-72
delegate Register delegate....4-72
4.3.2 SIIPrinterInfo Class....4-73
(1) Method List....4-73
(2) Property List....4-73
(3) Method Details....4-73
IIPrinterInfo Constructor....4-73
(4) Property Details....4-74
name Obtain printer model name....4-74
mac Obtain MAC address....4-74
ip Obtain IP address....4-74
4.3.3 SIIPrinterIdentifier Class....4-75
(1) Method List....4-75
(2) Property List....4-75
(3) Method Details....4-75
SIIPrinterIdentifier
Constructor....4-75
(4) Property Details....4-76
printerModelID
Get printer model ID....4-76
typeID Get type ID....4-76
romVersionID Get ROM version ID....4-76
mainFirmwareVersion
Get firmware version (main)....4-76
manufacturer Get manufacturer....4-76
modelName Get model name....4-77
serialNumber Get serial number....4-77
bootFirmwareVersion
Get firmware version (boot)....4-77
bootFirmwareChecksum
Get firmware checksum (boot)....4-77
mainFirmwareChecksum
Get firmware checksum (main)....4-77
firmwareChecksum
Get firmware checksum (main + boot)....4-77
4.3.4 SIIPrinterException Class....4-78
(1) Method List....4-78
(2) Property List....4-78
(3) Constant List....4-78

(4) Method Details....4-80
SIIPrinterException
Constructor....4-80
(5) Property Details....4-80
errorCode Obtain error codes....4-80
errorMessage Obtain error message....4-80
4.3.5 SIIPrinterManagerDelegate Protocol....4-81
(1) Method List....4-81
(2) Method Details....4-81
didStatusChange
Notify printer status....4-81
4.3.6 SIISmartLabelManager Class....4-82

Chapter 5 Sample Program 5-1

5.1 Screen Layout....5-1
5.1.1 Main Screen....5-1
5.1.2 [Settings] Screen....5-3
5.2 Precaution....5-3

Chapter 6 Disclaimer 6-1

Appendix A Character Set A-1

A.1 Codepage Table (Character Code Table)....A-1
A.2 International Character Set....A-10

Appendix B Barcode Size List B-1

B.1 Barcode Size List....B-1
B.1.1 printBarcode, printPageModeBarcode....B-1
B.1.2 printPDF417, printPageModePDF417....B-7
B.1.3 printQRCode, printPageModeQRCode....B-8
B.1.4 printDataMatrix, printPageModeDataMatrix....B-9
B.1.5 printMaxicode, printPageModeMaxicode....B-11

Appendix C Open Source Software License C-1

C.1 MIT License....C-1
C.2 Apache License 2.0....C-2

Chapter 1
Product Overview

This chapter describes the product overview of the SDK.

1.1 Function Provided by SDK

The SII print class library included in the SDK provides applications enabled by iOS device with the functions to use following SII printers: DPU-S245, DPU-S445 (hereinafter referred to as "Mobile printer" or "Mobile"), RP-D10, RP-E10 (hereinafter referred to as "POS printer" or "POS").

Moreover, the SDK includes Xcode projects for iOS device as a sample.

1.2 SII Print Class Library Overview

1.2.1 SII Print Class Library Configuration

The SII print class library (hereinafter referred to as "the library") and the sample programs included in the SDK are located in the section surrounded by dashed lines in the iOS configuration diagram (Figure 1-1). The library consists of two classes: the class which produces printer commands, and the class which controls communication port.

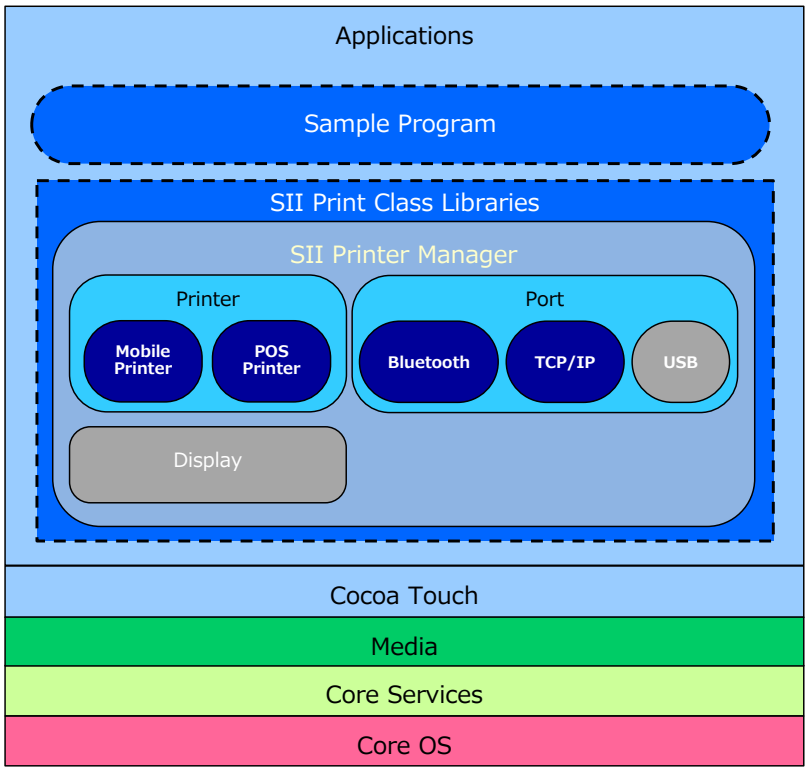


Figure 1-1

1.2.2 Function Provided by Library

By using the library, applications can easily transmit commands and data to a printer through communication port (Bluetooth or TCP/IP) on an iOS device. Also, applications can obtain printer status.

The library provides the following functions.

- Connection/disconnection to/from a printer
- Sending data to a printer (print data and/or commands[*1])
- Barcode print and 2-dimensional barcode print
- Sending a data file to a printer (print data and/or commands[*1])
- Cut paper
- Obtaining printer status
- Aborting the waiting state of a printer
- Obtaining various responses from a printer
- Bulk registration of print commands
- Registering a printer status call back function
- Printer search by TCP/IP
- Outputting a log file

*1: Commands that obtains the response from the printer are not available. In order to obtain responses from a printer, use "Obtaining printer status" or "Obtaining various responses from a printer".

(NOTE) ・ **Mobile printer and POS printer do not support the APIs relating to Display, the barcode scanner, or label printing function.**
・ **Mobile printer does not support the APIs relating to page mode.**

1.2.3 Development of Application that Performs Bluetooth Communication with SII Printer

When registering an application that performs Bluetooth communication with Mobile printer or POS printer to App Store, advance application from SII to Apple is necessary. For the details, contact SII.

Chapter 2
Product Specification

This chapter describes the product specification of the library.

2.1 Operating Environment

2.1.1 Applicable iDevice Models

Applicable iDevice Models for the library are shown below.

iPhone models

- iPhone X
- iPhone 8
- iPhone 8 Plus
- iPhone 7
- iPhone 7 Plus
- iPhone SE
- iPhone 6s
- iPhone 6s Plus
- iPhone 6
- iPhone 6 Plus

iPad models

- iPad Pro 12.9-inch (2nd generation)
- iPad Pro 10.5-inch
- iPad (5th generation)
- iPad Pro 9.7-inch
- iPad Pro 12.9-inch (1st generation)
- iPad mini 4
- iPad Air 2
- iPad mini 3

iPod models

- iPod touch (6th generation)

2.1.2 Applicable OS Versions

Applicable OS versions for the library are shown below.

- iOS 17 to 17.6.1
- iPadOS 17 to 17.6.1
- iOS 18 to 18.6.2
- iPadOS 18 to 18.6.2
- iOS 26 to 26.5
- iPadOS 26 to 26.5

2.1.3 Operating Conditions

This section describes the operating conditions for the library in Table 2-1, Table 2-2 and Table 2-3. Set the Function Setting/function selection to the values shown in each table before using the library. See the technical reference of each printer for details about Function Setting/function selection.

Table 2-1 Function Setting of the DPU-S245/DPU-S445 When Using Bluetooth Connection

SWDIP	Function	Value	Setting
2-1	Data Input Mode selection	1	Bluetooth/USB
2-2		1	
4-6	Busy Output When Error Occurs	0	Disable
4-8	Bluetooth Link Key Selection	0	Enable

Table 2-2 Function Setting of RP-D10/RP-E10 When Using TCP/IP Connection

MS	Function	Value	Setting
5-2	Initialized Response Selection	0	Enable
5-3	Data Discard Selection When Error Occurs	0	Enable

Table 2-3 Function Setting of RP-D10/RP-E10 When Using Bluetooth Connection

MS	Function	Value	Setting
5-2	Initialized Response Selection	0	Enable
5-3	Data Discard Selection When Error Occurs	0	Enable
39-1	iOS Automatic Connection Selection	0 / 1	0: Enable*1 1: Disable

*1: When use **`resetPrinter`**, select "Enable".

2.1.4 Precaution

This library is not thred-safe. When this library is used on multiple threads, abnormal termination may occur.

When using TCP/IP connection in this library, the communication port cannot be shared with printer drivers or other libraries.
When using TCP/IP connection in this library, wireless LAN access point connected to iOS device and POS printer need to be connected to the same network.

A concurrent connection from multiple apps to one printer is not supported when multiple apps are worked simultaneously by Multitasking on iPad with iPadOS.

Chapter 3
How to Use Library

This chapter describes development environment for iOS application and how to use the library.

3.1 Development Environment for iOS Application

In order to develop iOS applications, following tools are required.

- Xcode 15.0 or later

The description in and after this chapter is on the premise that the environment where each tool is available is prepared.

3.2 Provided Files

The file configuration of the SDK is as follows.

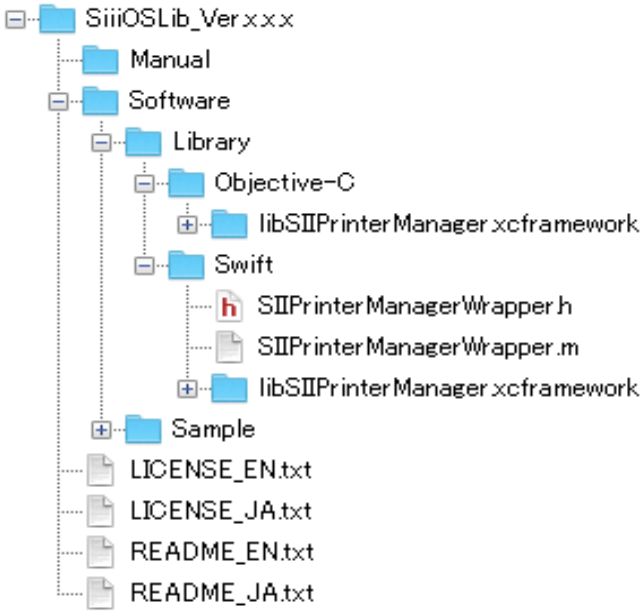


Figure 3-1

The file format of the library is XCFramework. The file name of the library is libSIIPrinterManager.xcframework.

3.3 Build Library into Xcode Project

Using the project of the sample program (SIILibSample) included in the SDK as an example, this section describes by development language how to build the library into the project.

See "Chapter 5 Sample Program" for sample programs included in the SDK.

Development Language	Description
Objective-C	See "3.3.1 Objective-C" for details to build the library as Objective-C.
Swift	See "3.3.2 Swift" for details to build the library as Swift.

(NOTE) If the following libraries provided SII Print Class Library for iOS Ver. 3.8.0 or earlier versions are included in the target project, delete them all.

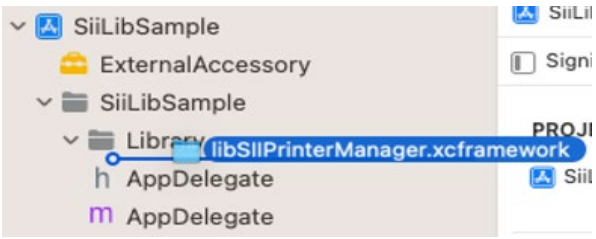
- **libSIIPrinterManager.a**
- **SIIPrinterEnum.h**
- **SIIPrinterException.h**
- **SIIPrinterManager.h**
- **SIISmartLabelManager.h**

3.3.1 Objective-C

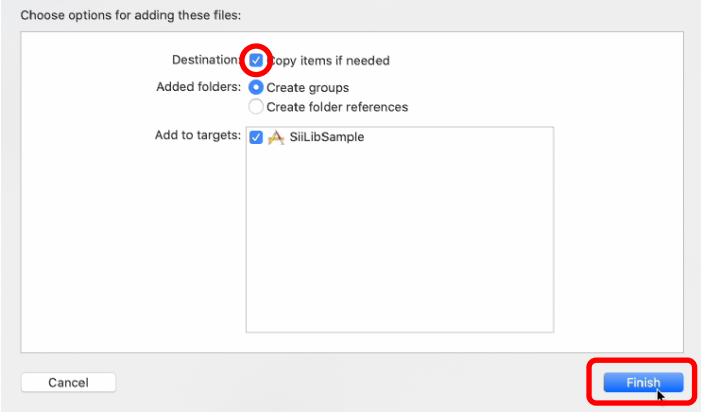
(1) Open the Xcode project.

(2) Drag the following files to any hierarchy in the target project in the [Project Navigator] of the navigation window.

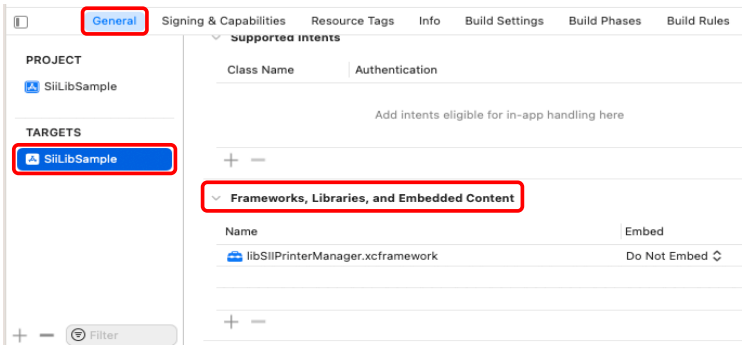
- libSIIPrinterManager.xcframework



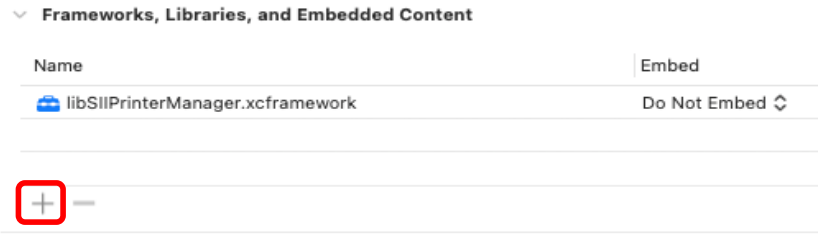
(3) Check the box [Copy items if needed], and click the [Finish] button.



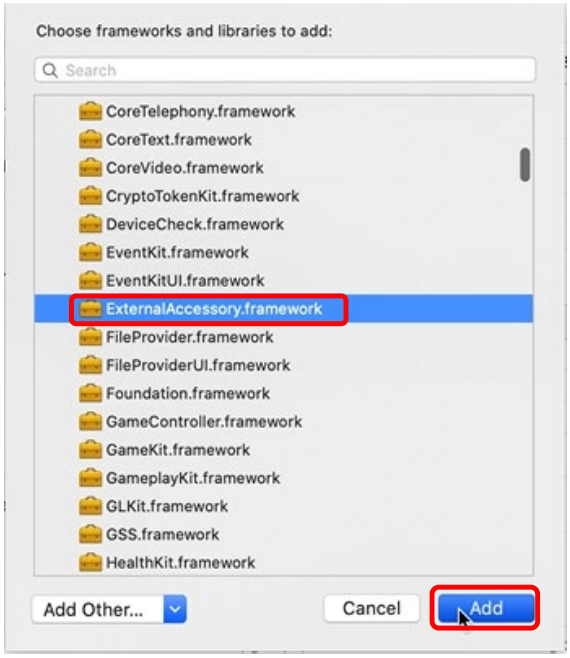
(4) Build the ExternalAccessory.framework into the project.
Select the target project in the [TARGETS], and open the [General] - [Frameworks, Libraries and Embedded Content].

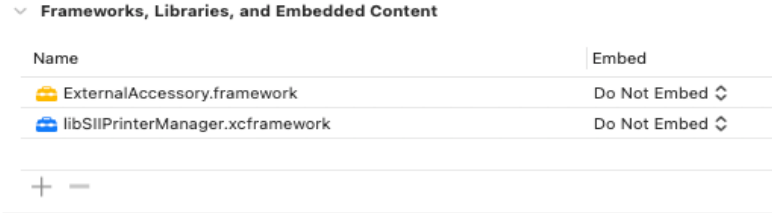


(5) Click the [+] button opened the [Frameworks, Libraries and Embedded Content].

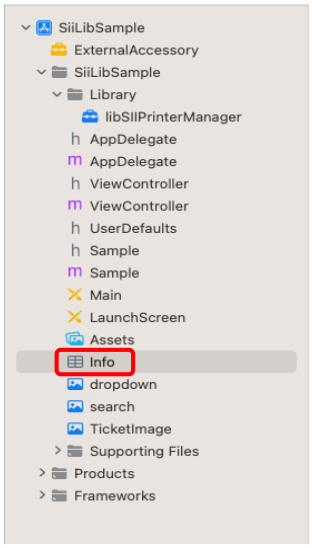


(6) Select the ExternalAccessory.framework from the list and click the [Add] button.

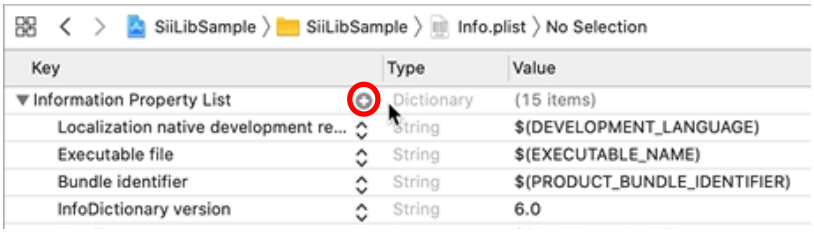




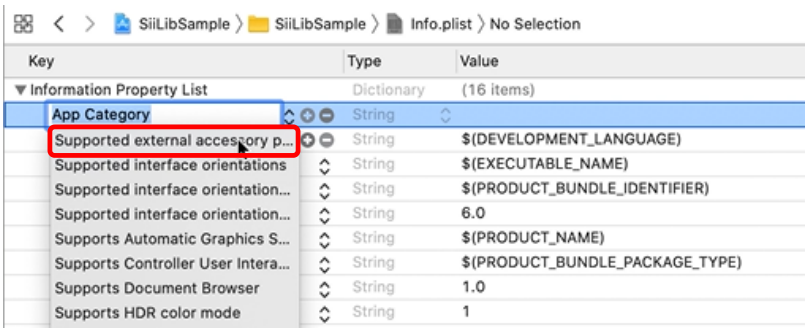
(7) Set the protocol name to use in the ExternalAccessory.framework. Select property list (.plist) in the [Project Navigator].



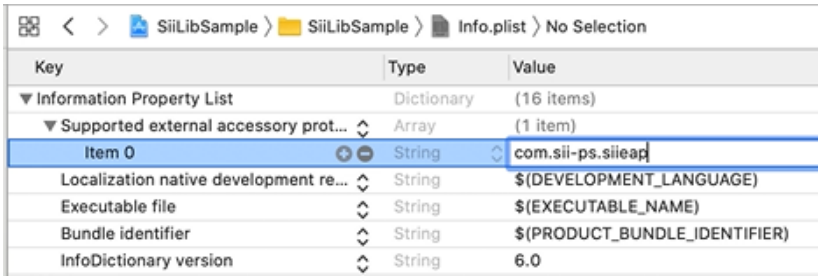
(8) Select the [Information Property List] - ⊕.



(9) Select the [Supported external accessory protocols] from the list.



(10) Open the added [Supported external accessory protocols].
The [Item 0] displayed in the opened [Supported external accessory protocols], enter com.sii-ps.siieap as the Value.



Use the following import statement when importing libraries.
#import <SIIPrinterManager/SIIPrinterManager.h>

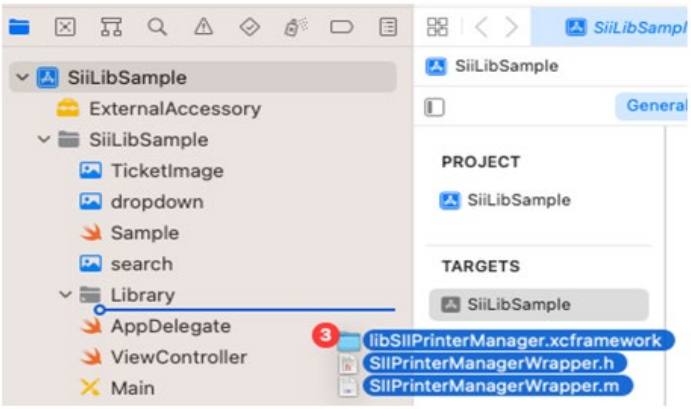
By completing these procedures, the library function becomes available.

3.3.2 Swift

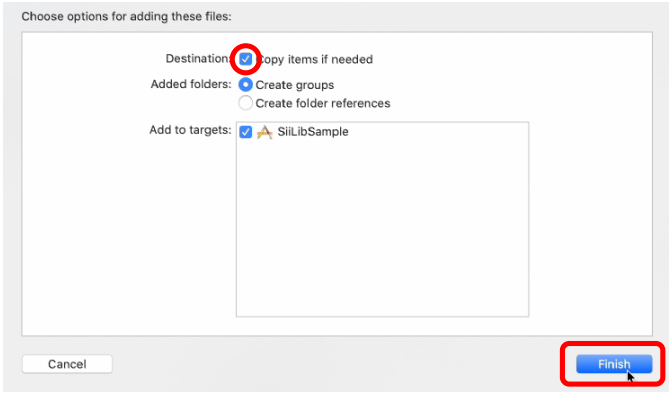
(1) Open the Xcode project.

(2) Drag the following files to any hierarchy in the target project in [Project Navigator] of the navigator window.

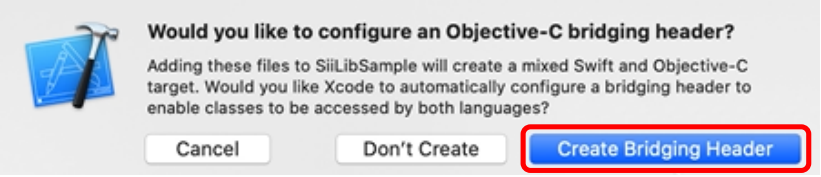
- libSIIPrinterManager.xcframework
- SIIPrinterManagerWrapper.h
- SIIPrinterManagerWrapper.m



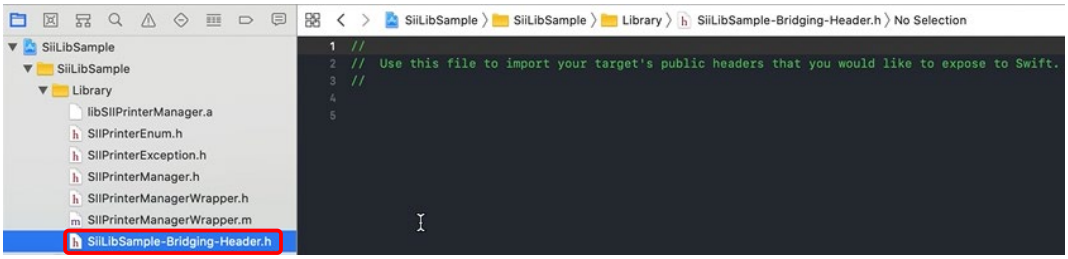
(3) Check the box [Copy items if needed], click the [Finish] button.



(4) The dialog is displayed. Select the [Create Bridging Header] button and create xxxxxxxx-Bridging-Header.h.



(5) Select the created xxxxxxxx-Bridging-Header.h.



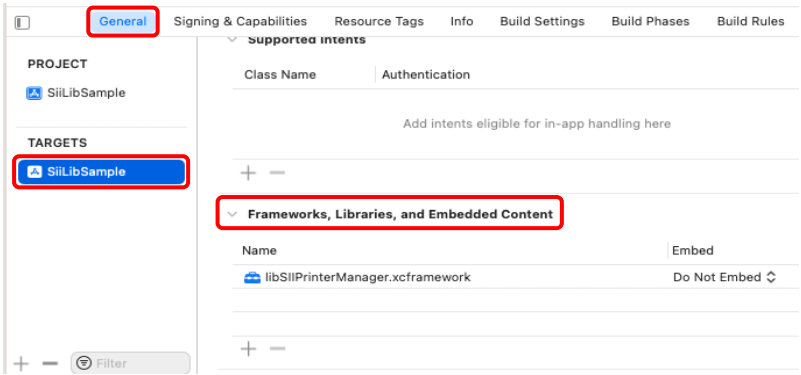
(6) Import the SIIPrinterManager.h and the SIIPrinterManagerWrapper.h into the xxxxxxxx-Bridging-Header.h.

```
//
//  Use this file to import your target's public headers that you would like to expose to Swift.
//

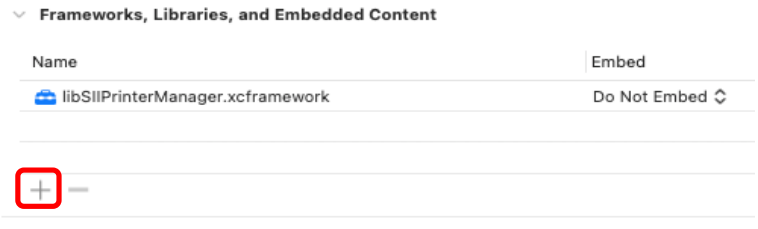
#import <SIIPrinterManager/SIIPrinterManager.h>
#import "SIIPrinterManagerWrapper.h"

```

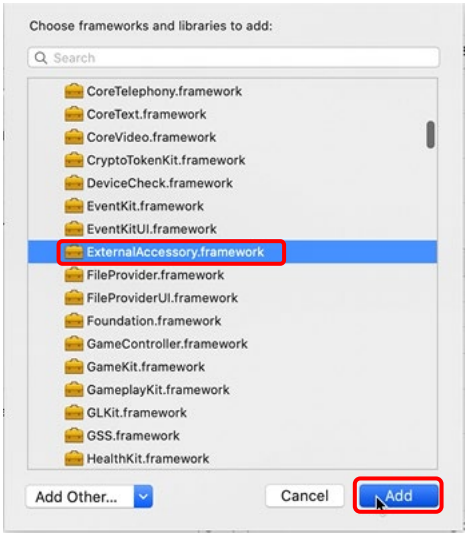
(7) Build the ExternalAccessory.framework.
Select the target project in the [TARGETS], and open the [General] - [Frameworks, Libraries and Embedded Content].

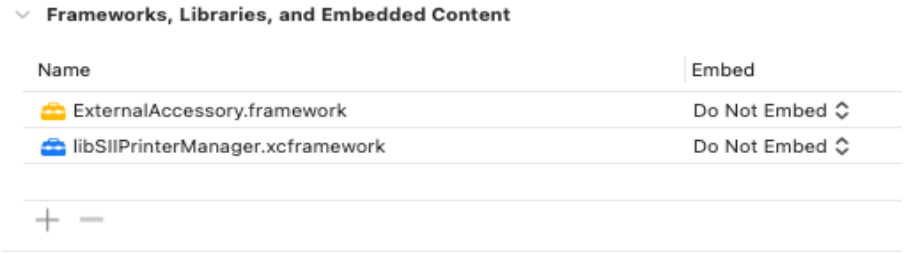


(8) Click the [+] button opened the [Frameworks, Libraries and Embedded Content].

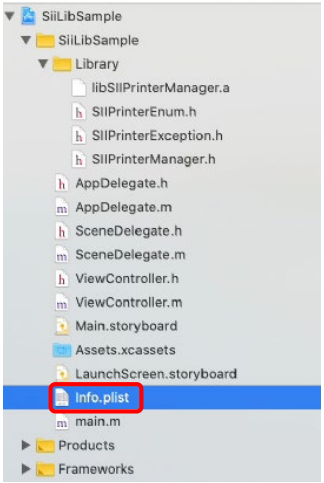


(9) Select the ExternalAccessory.framework from the list and click the [Add] button.

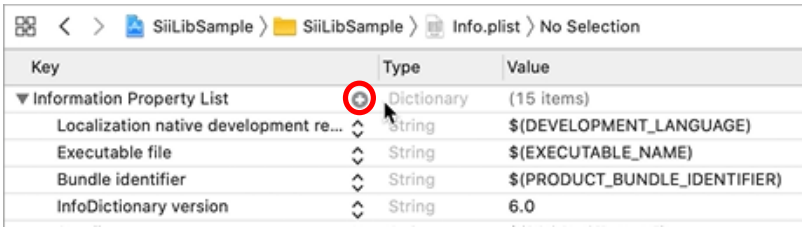




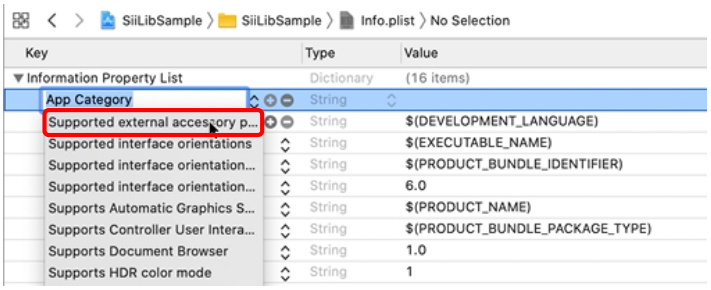
(10) Set the protocol name to use in the ExternalAccessory.framework. Select property list (.plist) in the [Project Navigator].



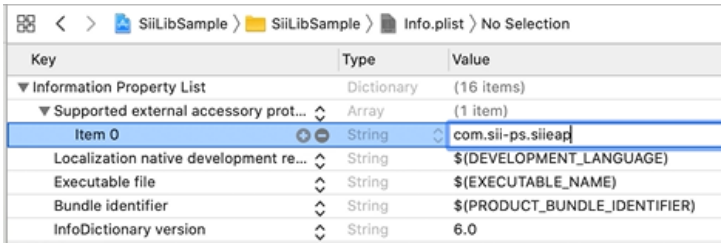
(11) Select the [Information Property List] - ⊕.



(12) Select the [Supported external accessory protocols] from the list.



(13) Open the added [Supported external accessory protocols].
The [Item 0] displayed in the opened [Supported external accessory protocols], enter com.sii-ps.siieap as the Value.



By completing these procedures, the library function becomes available.

Chapter 4
Function of Library

This chapter describes the APIs of each class implemented in the library.

4.1 Standard Mode and Page Mode

4.1.1 Basic Operation

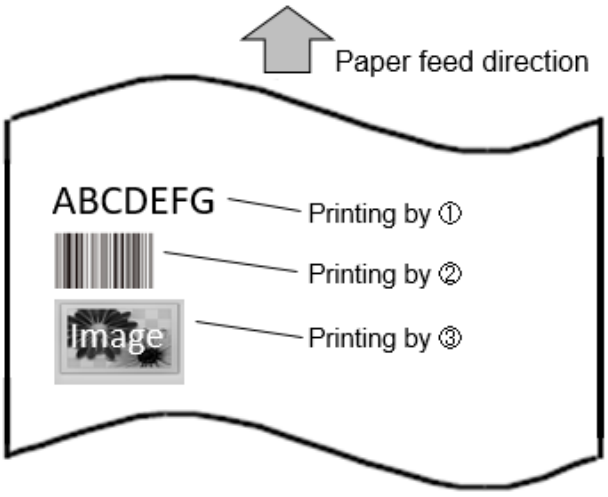
There are two printing modes "Standard mode" and "Page mode" in the library.
The "Standard mode" and "Page mode" are described below.

"Page mode" is supported only by POS printer.

(1) Standard mode

Standard mode is the mode to perform the printing in sequence.

Sample print command
① Send text data
② Print barcode
③ Send specified file (Specify an image file)



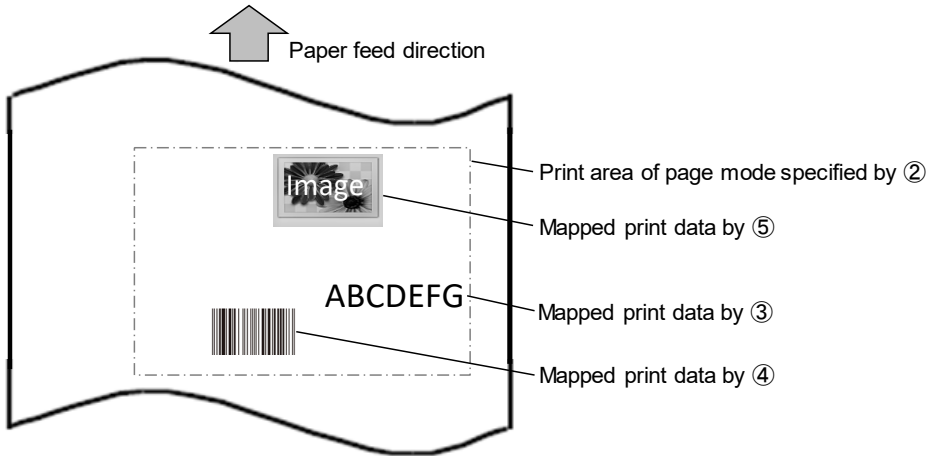
Standard mode suits the printing with an unfixed length such as a receipt.

(2) Page mode

Page mode is the mode to perform the printing on a per-page basis.

In page mode, the print area of page mode is allocated at first, and then print data is mapped on an arbitrary position of the print area.
The mapped print data is printed by the print method of page mode.

Sample print command
① Start page mode
② Specify print area of page mode
③ Send text data of page mode
④ Print barcode of page mode
⑤ Draw image file of page mode
⑥ Print page mode (print the data of ③④⑤ on the print area of ②)
⑦ End page mode



Page mode suits the printing for the followings.
・The printing with a fixed length.
・The printing with the coordinate determination of the character starting position or the ruled line
・printing position.

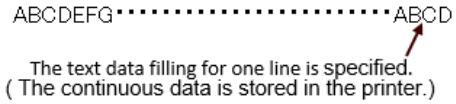
4.1.2 Text Data Printing in Standard Mode

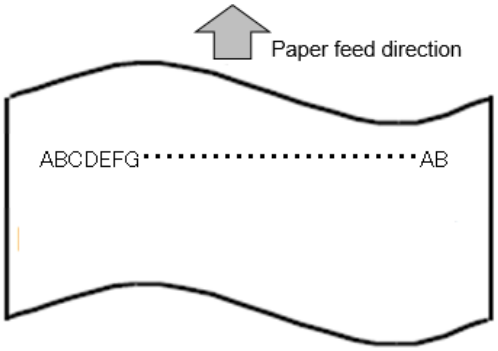
The text data in standard mode is printed each one line.

The text data is stored in the printer when the text data less than one line is specified.
The stored text data is printed by either the following conditions.

・The text data filling for one line is specified.
・Line Feed code is specified.

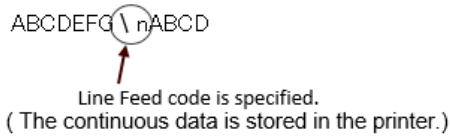
- **The print process when the text data filling for one line is specified.**

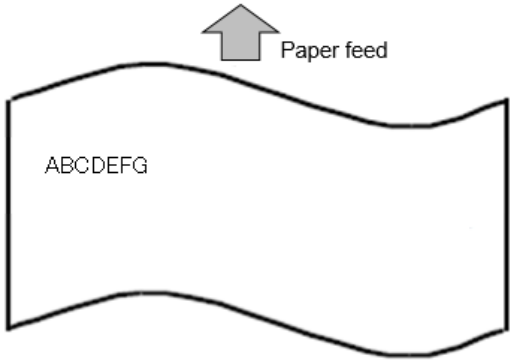




Input data **Print result**

- **The print process when Line Feed code is specified.**





Input data **Print result**

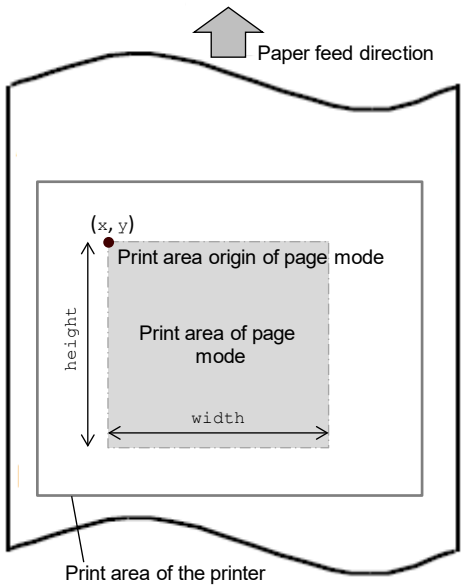
4.1.3 Mapping Position of Print Data in Page Mode

In page mode, the mapping position of print data is determined by print area, print direction, and reference point.
This section describes the print area, print direction, and reference point.

(1) Print area of page mode

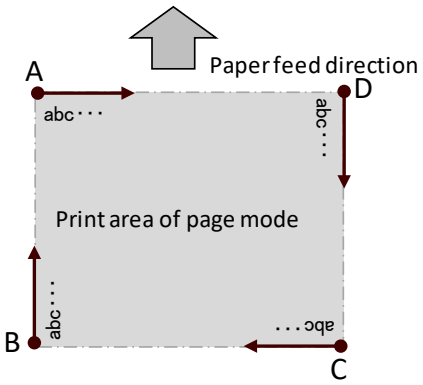
The print area of page mode is specified against the print area of the printer by the print area origin, and the width and the height of page mode. The view of the print area is shown in the following figures.

The print area of page mode can be specified more than one.



(2) Print direction

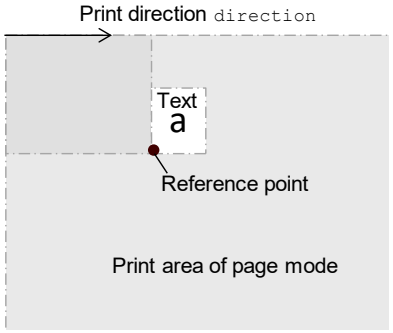
Specify the print direction at setting the print area of page mode.
The starting point is changed depending on specifying the print direction for each direction.
The relation between the print direction and the starting point is shown in the figure below.

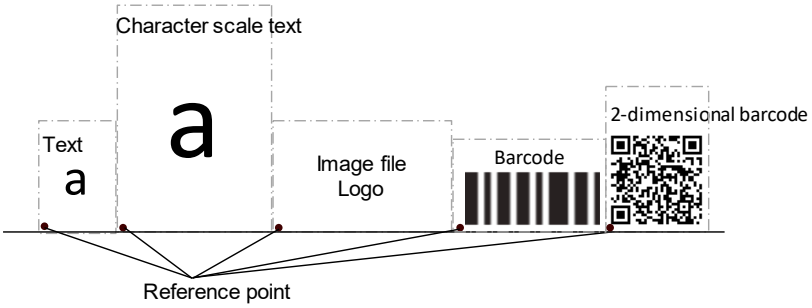


・Starting point: Upper left (A on the figure), Print direction: Left to Right
・Starting point: Left below (B on the figure), Print direction: Below to Upper
・Starting point: Right below (C on the figure), Print direction: Right to Left
・Starting point: Upper right (D on the figure), Print direction: Upper to Below

(3) Reference point

The relation between the reference point for mapping data and each print element (text, image file, logo, and barcode, etc.) is shown in the figures below.





(NOTE) The reference point cannot be specified out of the print area of page mode.

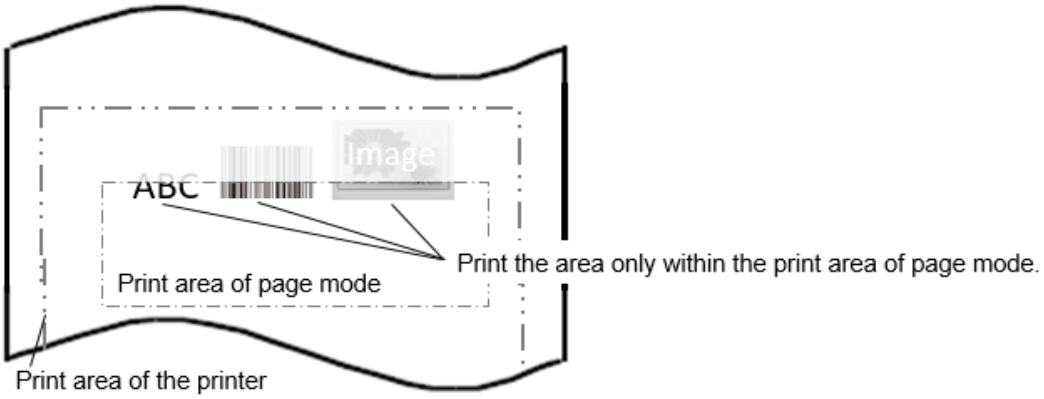
4.1.4 Print Data Process at Out of Print Area of Page Mode

This section describes the process when mapped data is to be mapped on out of the print area of page mode.

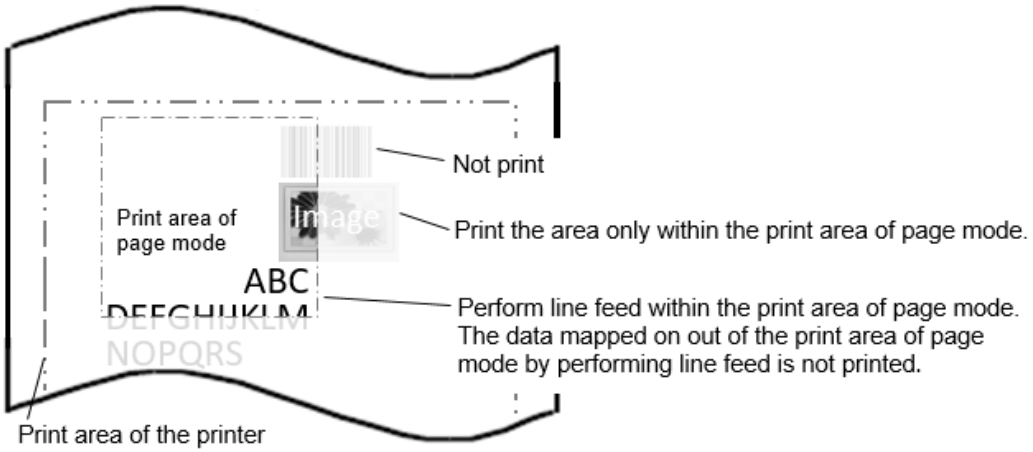
Type of Print Data

Text	Barcode, 2-dimensional Barcode	Image File, Logo, Rectangle, Ruled Line
ABC		

(1) The print data is mapped on the upper of the print area of page mode.



(2) The print data is mapped on the right of print area of page mode.



(NOTE) Read error or incorrect reading may occur when the part of mapped barcode data is on out of the print area of page mode.

4.2 Log File Output Function

The logs can be retrieved and the log files can be output using the library.

4.2.1 How to Set Log Output

Log output settings can be configured by adding the config.ini file with the following content to the Documents folder of the ios application that incorporates the library.

config.ini

```
LOGLEVEL=x
LOGSIZEMAX=xMB
LOGOUTPUT=x
```

Reference: See "4.2.2 Log Output Settings" for details on the settings for x.

4.2.2 Log Output Settings

Item	Description	Settings
LOGLEVEL	Log level	0 : Not record the log. 1 : Records an error log when **`SIIPrinterException`** is thrown. 2 : Records API execution history.
LOGSIZEMAX	Log file maximum size	1MB : Log file maximum size is 1 MB 5MB : Log file maximum size is 5 MB 10MB : Log file maximum size is 10 MB 50MB : Log file maximum size is 50 MB
LOGOUTPUT	Console output enabled/disabled	0 : Console output is disabled 1 : Console output is enabled

4.2.3 Log File

Log files are saved as local files in the iOS application that incorporates the library.

Log file name : PrinterManagerX.log (range of X is 0 to 4)

The 1st log file is created as PrinterManager0.log. If the log file maximum size is exceeded, changes the file name to PrinterManager1.log and creates a new PrinterManager0.log.

Up to 5 log files can be created.

4.3 API Reference

This section describes API in each class included in this library. This manual describes it as follows.

Class Name	Description
`SIIPrinterManager`	Class that provides the API used for communication with the printer and for printing. See "**4.3.1 `SIIPrinterManager` Class**".
`SIIPinterInfo`	Class that stores the printer information searched by printer searching method. See "**4.3.2 `SIIPrinterInfo` Class**".
`SIIPrinterIdentifier`	Class that stores the printer ID of the printer retrieved by **`getPrinterResponse`** method. See "**4.3.3 `SIIPrinterIdentifier` Class**" for details.
`SIIPrinterException`	Exception class that is thrown at API call. See "**4.3.4 `SIIPrinterException` Class**".
`SIIPrinterManagerDelegate`	API that gets notice from the printer. See "**4.3.5 `SIIPrinterManagerDelegate` Protocol**".
`SIISmartLabelManager`	Provides the API to specify label files or replace data.

(NOTE) **・ Mobile printer and POS printer do not support the APIs relating to Display, the barcode scanner, or label printing function.**
・ Mobile printer does not support the APIs relating to page mode.

4.3.1 `SIIPrinterManager` Class

(1) Method List

Methods provided by the **`SIIPrinterManager`** class are shown in the following table.
"Standard mode" or "Page mode" can be selected in the **`SIIPrinterManager`** class.
The available methods vary according to whether the target printer is the mobile printer or the POS printer.

Method	Description
Common method to standard mode and page mode	The valid methods in standard mode and page mode. See "4.3.1(1)(a) Common method to standard mode and page mode" for the methods.
Dedicated method for standard mode	The valid methods in standard mode. See "4.3.1(1)(b) Dedicated method for standard mode" for the methods.
Dedicated method for page mode	The valid methods in page mode. See "4.3.1(1)(c) Dedicated method for page mode" for the methods.

(a) Common method to standard mode and page mode

Methods provided by the common method to standard mode and page mode are shown in the following table. See "4.3.1(5)(a) Common method to standard mode and page mode" for details of the common methods.

Table 4-1 Common Method to Standard Mode and Page Mode in `SIIPrinterManager` Class

Method	Function Summary	Target	
		Mobile	POS
`init`	Instance	Supported	Supported
`connect`	Start communicating with a printer (Bluetooth)	Supported	Supported
`connect`	Start communicating with a printer (TCP/IP)	Not supported	Supported
`disconnect`	Disconnect a printer	Supported	Supported
`openDrawer`	Open cash drawer	Not supported	Supported
`buzzer`	Sound buzzer	Not supported	Supported
`externalBuzzer`	Sound external buzzer	Not supported	Not supported
`getStatus`	Obtain printer status	Supported	Supported
`abort`	Abort the waiting state of a printer	Supported	Supported
`registerLogo`	Register logo (image) to a printer	Supported*1	Supported*1
`unregisterLogo`	Delete specified logo (image) on a printer	Supported	Supported
`registerStyleSheet`	Register style sheet to a printer	Not supported	Supported
`unregisterStyleSheet`	Delete specified style sheet on a printer	Not supported	Supported
`resetPrinter`	Printer hardware reset	Not supported	Supported
`getPrinterResponse`	Obtain various responses from a printer	Supported*1	Supported*1
`startDiscoveryPrinter`	Start searching printer (Bluetooth)	Supported	Supported
`startDiscoveryPrinter`	Start searching printer (TCP/IP)	Not supported	Supported
`cancelDiscoveryPrinter`	Cancel printer search	Not supported	Supported
`getFoundPrinter`	Obtain searched printer information	Not supported	Supported
`getVersion`	Get SDK version	Supported	Supported
`controlTransaction`	Start/End batch processing	Supported	Supported

*1: Provided function varies depending on the target printer.

(b) Dedicated method for standard mode

Methods provided by the dedicated method for standard mode are shown in the following table.
See "4.3.1(5)(b) Dedicated method for standard mode" for details of the specified methods.

Table 4-2 Dedicated Method for Standard Mode in `SIIPrinterManager` Class

Method	Function Summary	Target	
		Mobile	POS
`sendText`	Send text data	Supported	Supported
`sendTextEx`	Send format specified text data	Supported	Not supported
`sendTextEx`	Send format specified text data	Not supported	Supported
`printBarcode`	Print barcode	Supported	Supported
`printPDF417`	Print PDF417	Supported	Supported
`printQRcode`	Print QR code.	Supported	Supported
`printDataMatrix`	Print Data Matrix	Supported	Supported
`printMaxiCode`	Print MaxiCode	Supported	Supported
`printGS1DataBarStacked`	Print GS1 Databar Stacked	Not supported	Not supported
`printGS1DataBarStackedOmnidirectional`	Print GS1Databar Stacked Omni-directional	Not supported	Not supported
`printGS1DataBarExpandedStacked`	Print GS1Databar Expanded Stacked	Not supported	Not supported
`printAztecCode`	Print AztecCode	Not supported	Not supported
`cutPaper`	Cut paper	Not supported	Supported
`feedPosition`	Paper form feed	Not supported	Not supported
`sendBinary`	Send binary data	Supported	Supported
`sendDataFile`	Send specified file	Supported*1	Supported*1
`sendImage`	Send image	Supported	Supported
`printPDF`	Print PDF page	Not supported	Supported
`printLogo`	Print specified logo on a printer	Supported	Not supported
`printLogo`	Print specified logo on a printer	Not supported	Supported

*1: Provided function varies depending on the target printer.

(c) Dedicated method for page mode

Methods provided by the dedicated method for page mode are shown in the following table.
See "4.3.1(5)(c) Dedicated method for page mode" for details of the specified methods.

Table 4-3 Dedicated Method for Page Mode in `SIIPrinterManager` Class

Method	Function Summary	Target	
		Mobile	POS
`enterPageMode`	Start page mode	Not supported	Supported
`exitPageMode`	End page mode	Not supported	Supported
`setPageModeArea`	Specify print area of page mode	Not supported	Supported
`setPageModeDirection`	Specify print direction of page mode	Not supported	Supported
`setPageModeLineSpacing`	Specify line spacing of page mode	Not supported	Supported

Method	Function Summary	Target	
		Mobile	POS
printPageMode	Print page mode	Not supported	Supported
printPageModeText	Send text data of page mode	Not supported	Supported
printPageModeTextEx	Send format specified text data of page mode	Not supported	Supported
printPageModeBarcode	Print barcode of page mode	Not supported	Supported
printPageModePDF417	Print PDF417 of page mode	Not supported	Supported
printPageModeQRcode	Print QR Code of page mode	Not supported	Supported
printPageModeDataMatrix	Print Data Matrix of page mode	Not supported	Supported
printPageModeGS1DataBarStacked	Print GS1 Databar Stacked of page mode	Not supported	Not supported
printPageModeGS1DataBarStackedOmnidirectional	Print GS1 Databar Stacked Omni-directional of page mode	Not supported	Not supported
printPageModeGS1DataBarExpandedStacked	Print GS1 Databar Expanded Stacked of page mode	Not supported	Not supported
printPageModeAztecCode	Print Aztec Code of page mode	Not supported	Not supported
printPageModeMaxiCode	Print MaxiCode of page mode	Not supported	Supported
sendPageModeBinary	Send binary data of page mode	Not supported	Supported
printPageModeImageFile	Draw image file of page mode	Not supported	Supported
printPageModeRectangle	Draw rectangle image of page mode	Not supported	Supported
printPageModeLine	Print ruled line of page mode	Not supported	Supported
printPageModeLogo	Print logo of page mode	Not supported	Supported

(2) Common property list to standard mode and page mode

Properties provided by the **SIIPrinterManager** class are shown in the following table.

Table 4-4 Property in SIIPrinterManager Class

Property	Access	Function Summary	Target	
			Mobile	POS
sendTimeout	R/W	Timeout period when sending data	Supported	Supported
receiveTimeout	R/W	Timeout period when receiving data	Supported	Supported
internationalCharacter	R/W	Set international character	Supported	Supported
codePage	R/W	Code page	Supported	Supported
printerModel	R	Obtain printer model	Supported	Supported
portType	R	Connecting port type	Supported	Supported
isConnect	R	Verify connection state with a printer	Supported	Supported
socketKeepingTime	R/W	Setting/Obtaining Socket keeping time	Not supported	Supported
delegate	R/W	Register delegate	Supported	Supported

(3) Constant List

(a) Printer model constant

Constants used for obtaining the printer model are shown in the following table.

Table 4-5 Printer Model Constant

Constant Name	Description	Value	Target	
			Mobile	POS
SII_PM_PRINTER_MODEL_DPU_S245	DPU-S245	284	Supported	Not supported
SII_PM_PRINTER_MODEL_DPU_S445	DPU-S445	281	Supported	Not supported
SII_PM_PRINTER_MODEL_RP_D10	RP-D10	295	Not supported	Supported
SII_PM_PRINTER_MODEL_RP_E10	RP-E10	291	Not supported	Supported

(b) Connecting port type constant

Constants used for obtaining the connecting port type are shown in the following table.

Table 4-6 Port Type Constant

Constant Name	Description	Value	Target	
			Mobile	POS
SII_PM_PRINTER_PORT_TYPE_ BLUETOOTH	Bluetooth	0	Supported	Supported
SII_PM_PRINTER_PORT_TYPE_TCP	TCP/IP	2	Not supported	Supported

(c) Response type constant

Constants used for obtaining various responses from the printer are shown in the following table.

Table 4-7 Response Type Constant

Constant Name	Description	Value	Target	
			Mobile	POS
SII_PM_PRINTER_RESPONSE_ REQUEST	Execution response request	0	Supported	Supported
SII_PM_PRINTER_RESPONSE_ USER_AREA	Send remaining capacity of user area	1	Supported	Supported
SII_PM_PRINTER_RESPONSE_ ARRANGE_USER_AREA	Send remaining capacity of user area after defragment	2	Not supported	Supported
SII_PM_PRINTER_RESPONSE_ NV_GRAPHICS	Send NV graphics memory capacity	3	Not supported	Supported
SII_PM_PRINTER_RESPONSE_ KEY_CODE	Send key code list of defined NV graphics	4	Not supported	Supported
SII_PM_PRINTER_RESPONSE_ BATTERY_STATUS	Battery voltage state	5	Supported	Not supported
SII_PM_PRINTER_RESPONSE_ FIRMWARE_VERSION	Send firmware version	6	Not supported	Supported
SII_PM_PRINTER_RESPONSE_ MAINTENANCE_COUNTER	Send maintenance counter	7	Not supported	Supported

Constant Name	Description	Value	Target	
			Mobile	POS
SII_PM_PRINTER_RESPONSE_ PRINTER_IDENTIFIER	Send printer ID	8	Not supported	Supported

(d) International character setting constant

Constants used for setting/obtaining the international character set are shown in the following table.

Table 4-8 International Character Setting Constant

Constant Name	Description	Value	Target	
			Mobile	POS
SII_PM_COUNTRY_USA	USA	0	Supported	Supported
SII_PM_COUNTRY_FRANCE	France	1	Supported	Supported
SII_PM_COUNTRY_GERMANY	Germany	2	Supported	Supported
SII_PM_COUNTRY_ENGLAND	United Kingdom	3	Supported	Supported
SII_PM_COUNTRY_DENMARK_1	Denmark I	4	Supported	Supported
SII_PM_COUNTRY_SWEDEN	Sweden	5	Supported	Supported
SII_PM_COUNTRY_ITALY	Italy	6	Supported	Supported
SII_PM_COUNTRY_SPAIN	Spain I	7	Supported	Supported
SII_PM_COUNTRY_JAPAN	Japan	8	Supported	Supported
SII_PM_COUNTRY_NORWAY	Norway	9	Supported	Supported
SII_PM_COUNTRY_DENMARK_2	Denmark II	10	Supported	Supported
SII_PM_COUNTRY_SPAIN_2	Spain II	11	Supported	Supported
SII_PM_COUNTRY_LATIN_AMERICA	Latin America	12	Supported	Supported
SII_PM_COUNTRY_ARABIA	Arabia	17	Not supported	Supported

(e) Codepage constant

Constants used for setting/obtaining codepage are shown in the following table.

Table 4-9 Codepage constant

Constant Name	Description	Value	Target	
			Mobile	POS
SII_PM_CODE_PAGE_437	USA, Standard Europe (Code Page437)	0	Not supported	Supported
SII_PM_CODE_PAGE_KATAKANA	Katakana	1	Supported	Supported
SII_PM_CODE_PAGE_850	Multilingual (Code Page850)	2	Not supported	Supported
SII_PM_CODE_PAGE_860	Portuguese (Code Page860)	3	Not supported	Supported
SII_PM_CODE_PAGE_863	Canadian-French (Code Page863)	4	Not supported	Supported
SII_PM_CODE_PAGE_865	Nordic (Code Page865)	5	Not supported	Supported
SII_PM_CODE_PAGE_1252	Latin (Code Page1252)	16	Supported	Supported
SII_PM_CODE_PAGE_852	Eastern Europe (Code Page852)	18	Not supported	Supported

Constant Name	Description	Value	Target	
			Mobile	POS
SII_PM_CODE_PAGE_858	Euro (Code Page858)	19	Not supported	Supported
SII_PM_CODE_PAGE_864[*1*2]	Arabic (Code Page864)	37	Not supported	Supported
SII_PM_CODE_PAGE_1250	Central European (Code Page1250)	45	Not supported	Supported
SII_PM_CODE_PAGE_1251	Cyrillic (Code Page1251)	46	Not supported	Supported
SII_PM_CODE_PAGE_1253[*3]	Greek (Code Page1253)	47	Not supported	Supported
SII_PM_CODE_PAGE_1254	Turkish (Code Page1254)	48	Not supported	Supported

Note: This library does not support other than the above code page.
*1: When **SII_PM_CODE_PAGE_864** is specified, 20ACh of the Unicode cannot be printed.
*2: When **SII_PM_CODE_PAGE_864** is specified, Font B cannot be printed.
*3: When **SII_PM_CODE_PAGE_1253** is specified, 00AAh of the Unicode cannot be printed.

(f) Constant used for barcode and PDF417

Constants used for printing barcode and PDF417 are shown in the following table.

Table 4-10 Constant for Barcode and PDF417

Constant Name	Description	Value	Target	
			Mobile	POS
SII_PM_BARCODE_HEIGHT_DEFAULT	Default value of barcode height	162	Supported	Supported
SII_PM_PDF417_ MODULE_HEIGHT_DEFAULT	Default value of PDF417 height	10	Supported	Supported
SII_PM_PDF417_ROW_AUTO	Automatic selection of the number of rows	0	Supported	Supported
SII_PM_PDF417_COLUMN_AUTO	Automatic selection of the number of columns	0	Supported	Supported

(4) Constant List of Enumerated Type

(a) Drawer number (**DrawerNum**)

Enumerated type constants used for drawer number are shown in the following table.

Table 4-11 Drawer Number (DrawerNum)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_DRAWER_1	Drawer 1	Not supported	Supported
SII_PM_DRAWER_2	Drawer 2	Not supported	Supported

(b) Activation pulse width (**PulseWidth**)

Enumerated type constants used for activation pulse width are shown in the following table.

Table 4-12 Activation Pulse Width (PulseWidth)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_ON_OFF_TIME_100	ON/OFF time 100 millisecond	Not supported	Supported
SII_PM_ON_OFF_TIME_200	ON/OFF time 200 millisecond	Not supported	Supported
SII_PM_ON_OFF_TIME_300	ON/OFF time 300 millisecond	Not supported	Supported
SII_PM_ON_OFF_TIME_400	ON/OFF time 400 millisecond	Not supported	Supported
SII_PM_ON_OFF_TIME_500	ON/OFF time 500 millisecond	Not supported	Supported
SII_PM_ON_OFF_TIME_600	ON/OFF time 600 millisecond	Not supported	Supported
SII_PM_ON_OFF_TIME_700	ON/OFF time 700 millisecond	Not supported	Supported
SII_PM_ON_OFF_TIME_800	ON/OFF time 800 millisecond	Not supported	Supported

(c) Dithering (**Dithering**)

Enumerated type constants used for dithering are shown in in the following table.

Table 4-13 Dithering (Dithering)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_DITHERING_DISABLE	Dithering is disable	Supported	Supported
SII_PM_DITHERING_ERRORDIFFUSION	Dithering is enable	Supported	Supported

(d) Batch processing selection (**TransactionFunction**)

Enumerated type constants used for batch processing selection are shown in the following table.

Table 4-14 Batch Processing Selection (TransactionFunction)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_TRANSACTION_CLEAR	Clear batch transaction	Supported	Supported
SII_PM_TRANSACTION_START	Start batch transaction	Supported	Supported
SII_PM_TRANSACTION_PRINT	Finish batch printing and batch processing	Supported	Supported

(e) Bold print (**CharacterBold**)

Enumerated type constants used for bold print are shown in the following table.

Table 4-15 Bold Print (CharacterBold)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_BOLD_CANCEL	Release bold print	Supported	Supported
SII_PM_BOLD	Specify bold print	Supported	Supported

(f) Underline (**CharacterUnderline**)

Enumerated type constants used for underline are shown in the following table.

Table 4-16 Underline (CharacterUnderline)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_UNDERLINE_CANCEL	Release underline print	Supported	Supported
SII_PM_UNDERLINE_1	Specify 1 dot width underline print	Supported	Supported
SII_PM_UNDERLINE_2	Specify 2 dots width underline print	Supported	Supported

(g) Reverse print (**CharacterReverse**)

Enumerated type constants used for reverse print are shown in the following table.

Table 4-17 Reverse Print (CharacterReverse)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_REVERSE_CANCEL	Release reverse print	Not Supported	Supported
SII_PM_REVERSE	Specify reverse print	Not Supported	Supported

(h) Inversion print (**CharacterInversion**)

Enumerated type constants used for inversion print are shown in the following table.
Inversion print cannot be added to the text data before inserting a new line feed.

Table 4-18 Character Inversion (CharacterInversion)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_INVERSION_CANCEL	Cancel inversion print	Supported	Supported
SII_PM_INVERSION	Specify inversion print	Supported	Supported

(i) Character font (**CharacterFont**)

Enumerated type constants used for Character font are shown in the following table.

Table 4-19 Character Font (CharacterFont)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_FONT_A	Font A (24 × 12)	Supported	Supported
SII_PM_FONT_B	Font B (16 × 8)	Supported	Supported

(j) Character Scale (**CharacterScale**)

Enumerated type constants used for character scale are shown in the following table.

Table 4-20 Character Scale (CharacterScale)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_VARTICAL_1_HORIZONTAL_1	Height × 1 and width × 1	Supported	Supported
SII_PM_VARTICAL_1_HORIZONTAL_2	Height × 1 and width × 2	Supported	Supported
SII_PM_VARTICAL_1_HORIZONTAL_3	Height × 1 and width × 3	Not supported	Supported
SII_PM_VARTICAL_1_HORIZONTAL_4	Height × 1 and width × 4	Not supported	Supported
SII_PM_VARTICAL_2_HORIZONTAL_1	Height × 2 and width × 1	Supported	Supported
SII_PM_VARTICAL_2_HORIZONTAL_2	Height × 2 and width × 2	Supported	Supported
SII_PM_VARTICAL_2_HORIZONTAL_3	Height × 2 and width × 3	Not supported	Supported
SII_PM_VARTICAL_2_HORIZONTAL_4	Height × 2 and width × 4	Not supported	Supported
SII_PM_VARTICAL_2_HORIZONTAL_6	Height × 2 and width × 6	Not supported	Supported
SII_PM_VARTICAL_3_HORIZONTAL_1	Height × 3 and width × 1	Not supported	Supported
SII_PM_VARTICAL_3_HORIZONTAL_2	Height × 3 and width × 2	Not supported	Supported
SII_PM_VARTICAL_3_HORIZONTAL_3	Height × 3 and width × 3	Not supported	Supported
SII_PM_VARTICAL_3_HORIZONTAL_4	Height × 3 and width × 4	Not supported	Supported
SII_PM_VARTICAL_4_HORIZONTAL_1	Height × 4 and width × 1	Not supported	Supported
SII_PM_VARTICAL_4_HORIZONTAL_2	Height × 4 and width × 2	Not supported	Supported
SII_PM_VARTICAL_4_HORIZONTAL_3	Height × 4 and width × 3	Not supported	Supported
SII_PM_VARTICAL_4_HORIZONTAL_4	Height × 4 and width × 4	Not supported	Supported
SII_PM_VARTICAL_4_HORIZONTAL_6	Height × 4 and width × 6	Not supported	Supported

Constant Name	Description	Target	
		Mobile	POS
SII_PM_VARTICAL_4_HORIZONTAL_8	Height × 4 and width × 8	Not supported	Supported
SII_PM_VARTICAL_6_HORIZONTAL_2	Height × 6 and width × 2	Not supported	Supported
SII_PM_VARTICAL_6_HORIZONTAL_4	Height × 6 and width × 4	Not supported	Supported
SII_PM_VARTICAL_6_HORIZONTAL_6	Height × 6 and width × 6	Not supported	Supported
SII_PM_VARTICAL_6_HORIZONTAL_8	Height × 6 and width × 8	Not supported	Supported
SII_PM_VARTICAL_8_HORIZONTAL_4	Height × 8 and width × 4	Not supported	Supported
SII_PM_VARTICAL_8_HORIZONTAL_6	Height × 8 and width × 6	Not supported	Supported
SII_PM_VARTICAL_8_HORIZONTAL_8	Height × 8 and width × 8	Not supported	Supported

(k) Alignment (**PrintAlignment**)

Enumerated type constants used for alignment are shown in the following table.
Alignment cannot be added to the text data before inserting a new line feed.

Table 4-21 Alignment (PrintAlignment)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_ALIGNMENT_LEFT	Align left	Supported	Supported
SII_PM_ALIGNMENT_CENTER	Centered	Supported	Supported
SII_PM_ALIGNMENT_RIGHT	Align right	Supported	Supported

(l) Barcode symbol (**BarcodeSymbol**)

Enumerated type constants used for barcode symbol are shown in the following table.

Table 4-22 Barcode Symbol (BarcodeSymbol)

Constant Name	Description	Type	Target	
			Mobile	POS
SII_PM_BARCODE_UPC_A	UPC-A	(a)	Supported	Supported
SII_PM_BARCODE_UPC_E	UPC-E	(a)	Supported	Supported
SII_PM_BARCODE_EAN13	EAN13	(a)	Supported	Supported
SII_PM_BARCODE_JAN13	JAN13	(a)	Supported	Supported
SII_PM_BARCODE_EAN8	EAN8	(a)	Supported	Supported
SII_PM_BARCODE_JAN8	JAN8	(a)	Supported	Supported
SII_PM_BARCODE_CODE39	CODE39	(a), (b)	Supported	Supported
SII_PM_BARCODE_CODE93	CODE93	(c)	Not supported	Supported
SII_PM_BARCODE_CODE128	CODE128	(c)	Supported	Supported
SII_PM_BARCODE_ITF	ITF	(a), (b)	Supported	Supported
SII_PM_BARCODE_CODABAR	CODABAR	(a), (b)	Supported	Supported
SII_PM_BARCODE_EAN13_ADDON	EAN13 add-on	(a)	Not supported	Supported
SII_PM_BARCODE_JAN13_ADDON	JAN13 add-on	(a)	Not supported	Supported

See **`printBarcode`** method or **`printPageModeBarcode`** method for the type.

(m) Module size (**ModuleSize**)

Enumerated type constants used for module size are shown in the following table.

Table 4-23 Module Size (ModuleSize)

Constant Name	Description	Using Method	Target	
			Mobile*1	POS
SII_PM_BARCODE_MODULE_WIDTH_2	Fine element 2 dots Module width 0.250 mm	**printBarcode** **printPageModeBarcode**	Supported	Supported
SII_PM_BARCODE_MODULE_WIDTH_3	Fine element 3 dots Module width 0.375 mm		Supported	Supported
SII_PM_BARCODE_MODULE_WIDTH_4	Fine element 4 dots Module width 0.500 mm		Supported	Supported
SII_PM_BARCODE_MODULE_WIDTH_5	Fine element 5 dots Module width 0.625 mm		Not supported	Supported
SII_PM_BARCODE_MODULE_WIDTH_6	Fine element 6 dots Module width 0.750 mm		Not supported	Supported
SII_PM_PDF417_MODULE_WIDTH_2	Nominal fine element width 2 dots	**printPDF417** **printPageModePDF417**	Supported	Supported
SII_PM_PDF417_MODULE_WIDTH_3	Nominal fine element width 3 dots		Supported	Supported
SII_PM_PDF417_MODULE_WIDTH_4	Nominal fine element width 4 dots		Supported	Supported
SII_PM_PDF417_MODULE_WIDTH_5	Nominal fine element width 5 dots		Supported	Not supported
SII_PM_PDF417_MODULE_WIDTH_6	Nominal fine element width 6 dots		Supported	Not supported
SII_PM_PDF417_MODULE_WIDTH_7	Nominal fine element width 7 dots		Supported	Not supported
SII_PM_PDF417_MODULE_WIDTH_8	Nominal fine element width 8 dots		Supported	Not supported
SII_PM_QR_MODULE_SIZE_2	2 dots	**printQRcode** **printPageModeQRcode**	Supported	Supported
SII_PM_QR_MODULE_SIZE_3	3 dots		Supported	Supported
SII_PM_QR_MODULE_SIZE_4	4 dots		Supported	Supported
SII_PM_QR_MODULE_SIZE_5	5 dots		Supported	Supported
SII_PM_QR_MODULE_SIZE_6	6 dots		Supported	Supported
SII_PM_QR_MODULE_SIZE_7	7 dots		Supported	Supported
SII_PM_QR_MODULE_SIZE_8	8 dots		Supported	Supported
SII_PM_QR_MODULE_SIZE_9	9 dots		Supported	Supported
SII_PM_QR_MODULE_SIZE_10	10 dots		Supported	Supported
SII_PM_QR_MODULE_SIZE_11	11 dots		Supported	Supported
SII_PM_DATAMATRIX_MODULE_SIZE_2	2 dots	**printDataMatrix** **printPageModeDataMatrix**	Supported	Supported
SII_PM_DATAMATRIX_MODULE_SIZE_3	3 dots		Supported	Supported
SII_PM_DATAMATRIX_MODULE_SIZE_4	4 dots		Supported	Supported

Constant Name	Description	Using Method	Target	
			Mobile*1	POS
SII_PM_DATAMATRIX_ MODULE_SIZE_5	5 dots	`printDataMatrix` `printPageModeDataMatrix`	Supported	Supported
SII_PM_DATAMATRIX_ MODULE_SIZE_6	6 dots		Supported	Supported
SII_PM_DATAMATRIX_ MODULE_SIZE_7	7 dots		Supported	Supported
SII_PM_DATAMATRIX_ MODULE_SIZE_8	8 dots		Supported	Supported
SII_PM_DATAMATRIX_ MODULE_SIZE_9	9 dots		Supported	Supported
SII_PM_DATAMATRIX_ MODULE_SIZE_10	10 dots		Supported	Supported
SII_PM_DATAMATRIX_ MODULE_SIZE_11	11 dots		Supported	Supported

*1: Mobile does not support the dedicated method for page mode.

(n) HRI character print position (**HriPosition**)

Enumerated type constants used for HRI character print position are shown in the following table.

Table 4-24 HRI Character Print Position (HriPosition)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_HRI_NONE	Do not print	Supported	Supported
SII_PM_HRI_POSITION_ABOVE	Above barcode	Supported	Supported
SII_PM_HRI_POSITION_BELOW	Below barcode	Supported	Supported
SII_PM_HRI_POSITION_ABOVE_ BELOW	Above and below barcode (both)	Supported	Supported

(o) N:W ratio (**NwRatio**)

Enumerated type constants used for N:W ratio are shown in the following table.

Table 4-25 N:W Ratio (NwRatio)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_BARCODE_WIDE_WIDTH_1	Wide width type 1	Supported	Not supported
SII_PM_BARCODE_WIDE_WIDTH_2	Wide width type 2	Supported	Not supported
SII_PM_BARCODE_WIDE_WIDTH_3	Wide width type 3	Supported	Not supported
SII_PM_BARCODE_WIDE_WIDTH_4	Wide width type 4	Supported	Not supported
SII_PM_NWRATIO_1TO2	1:2	Not supported	Supported
SII_PM_NWRATIO_1TO2_5	1:2.5	Not supported	Supported
SII_PM_NWRATIO_1TO3	1:3	Not supported	Supported

(p) Error correction level (**ErrorCorrection**)

Enumerated type constants used for error correction level are shown in the following table.

Table 4-26 Error Correction Level (ErrorCorrection)

Constant Name	Description	Using Method	Target	
			Mobile*1	POS
SII_PM_PDF417_ERROR_CORRECTION_0	Error correction level 0	`printPDF417` `printPageModePDF417`	Supported	Supported
SII_PM_PDF417_ERROR_CORRECTION_1	Error correction level 1		Supported	Supported
SII_PM_PDF417_ERROR_CORRECTION_2	Error correction level 2		Supported	Supported
SII_PM_PDF417_ERROR_CORRECTION_3	Error correction level 3		Supported	Supported
SII_PM_PDF417_ERROR_CORRECTION_4	Error correction level 4		Supported	Supported
SII_PM_PDF417_ERROR_CORRECTION_5	Error correction level 5		Supported	Supported
SII_PM_PDF417_ERROR_CORRECTION_6	Error correction level 6		Supported	Supported
SII_PM_PDF417_ERROR_CORRECTION_7	Error correction level 7		Supported	Supported
SII_PM_PDF417_ERROR_CORRECTION_8	Error correction level 8		Supported	Supported
SII_PM_QR_ERROR_CORRECTION_L	Error correction level L	`printQRcode` `printPageModeQRcode`	Supported	Supported
SII_PM_QR_ERROR_CORRECTION_M	Error correction level M		Supported	Supported
SII_PM_QR_ERROR_CORRECTION_H	Error correction level H		Supported	Supported
SII_PM_QR_ERROR_CORRECTION_Q	Error correction level Q		Supported	Supported

*1: Mobile does not support the dedicated method for page mode.

(q) PDF417 symbol (**Pdf417Symbol**)

Enumerated type constants used for PDF417 symbol are shown in the following table.

Table 4-27 PDF417 Symbol (Pdf417Symbol)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_PDF417_STANDARD	Standard PDF417	Supported	Supported
SII_PM_PDF417_COMPACT	Compact PDF417	Supported	Supported

(r) QR code model (**QrModel**)

Enumerated type constants used for QR code model are shown in the following table.

Table 4-28 QR Code Model (QrModel)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_QR_MODEL_1	QR code model 1	Supported	Supported
SII_PM_QR_MODEL_2	QR code model 2	Supported	Supported

(s) Data Matrix Module (**DataMatrixModule**)

Enumerated type constants used for Data Matrix module are shown in the following table.

Table 4-29 Data Matrix Module (DataMatrixModule)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_DATA_MATRIX_AUTO	Module numbers: Automatic	Supported	Supported
SII_PM_DATA_MATRIX_10_10	Module numbers: 10 × 10	Supported	Supported
SII_PM_DATA_MATRIX_12_12	Module numbers: 12 × 12	Supported	Supported
SII_PM_DATA_MATRIX_14_14	Module numbers: 14 × 14	Supported	Supported
SII_PM_DATA_MATRIX_16_16	Module numbers: 16 × 16	Supported	Supported
SII_PM_DATA_MATRIX_18_18	Module numbers: 18 × 18	Supported	Supported
SII_PM_DATA_MATRIX_20_20	Module numbers: 20 × 20	Supported	Supported
SII_PM_DATA_MATRIX_22_22	Module numbers: 22 × 22	Supported	Supported
SII_PM_DATA_MATRIX_24_24	Module numbers: 24 × 24	Supported	Supported
SII_PM_DATA_MATRIX_26_26	Module numbers: 26 × 26	Supported	Supported
SII_PM_DATA_MATRIX_32_32	Module numbers: 32 × 32	Supported	Supported
SII_PM_DATA_MATRIX_36_36	Module numbers: 36 × 36	Supported	Supported
SII_PM_DATA_MATRIX_40_40	Module numbers: 40 × 40	Supported	Supported
SII_PM_DATA_MATRIX_44_44	Module numbers: 44 × 44	Supported	Supported
SII_PM_DATA_MATRIX_48_48	Module numbers: 48 × 48	Supported	Supported
SII_PM_DATA_MATRIX_52_52	Module numbers: 52 × 52	Supported	Supported
SII_PM_DATA_MATRIX_64_64	Module numbers: 64 × 64	Supported	Supported
SII_PM_DATA_MATRIX_72_72	Module numbers: 72 × 72	Supported	Supported
SII_PM_DATA_MATRIX_80_80	Module numbers: 80 × 80	Supported	Supported
SII_PM_DATA_MATRIX_88_88	Module numbers: 88 × 88	Supported	Supported
SII_PM_DATA_MATRIX_96_96	Module numbers: 96 × 96	Supported	Supported
SII_PM_DATA_MATRIX_104_104	Module numbers: 104 × 104	Supported	Supported
SII_PM_DATA_MATRIX_120_120	Module numbers: 120 × 120	Supported	Supported
SII_PM_DATA_MATRIX_132_132	Module numbers: 132 × 132	Supported	Supported
SII_PM_DATA_MATRIX_144_144	Module numbers: 144 × 144	Supported	Supported

Constant Name	Description	Target	
		Mobile	POS
SII_PM_DATA_MATRIX_8_18	Module numbers: 8 × 18	Supported	Supported
SII_PM_DATA_MATRIX_8_32	Module numbers: 8 × 32	Supported	Supported
SII_PM_DATA_MATRIX_12_26	Module numbers: 12 × 26	Supported	Supported
SII_PM_DATA_MATRIX_12_36	Module numbers: 12 × 36	Supported	Supported
SII_PM_DATA_MATRIX_16_36	Module numbers: 16 × 36	Supported	Supported
SII_PM_DATA_MATRIX_16_48	Module numbers: 16 × 48	Supported	Supported

(t) MaxiCode Mode (**MaxiCodeMode**)

Enumerated type constants used for MaxiCode mode are shown in the following table.

Table 4-30 MaxiCode Mode (MaxiCodeMode)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_MAXI_CODE_2	Mode2	Supported	Supported
SII_PM_MAXI_CODE_3	Mode3	Supported	Supported
SII_PM_MAXI_CODE_4	Mode4	Supported	Supported
SII_PM_MAXI_CODE_5	Mode5	Supported	Supported

(u) Cutting method (**CuttingMethod**)

Enumerated type constants used for cutting method are shown in the following table.

Table 4-31 Cutting Method (CuttingMethod)

Constant Name	Description		Target	
	Paper Feed to Cut Position	Cutting Method	Mobile	POS
SII_PM_CUT_FULL	Enabled	Full cut	Not supported	Supported
SII_PM_CUT_FULL_NO_FEED	Disabled			
SII_PM_CUT_PARTIAL	Enabled	Partial cut	Not supported	Supported
SII_PM_CUT_PARTIAL_NO_FEED	Disabled			
SII_PM_CUT_NONE[*1]	Disabled	Supported	Not supported	Supported

*1: Supported only by **printPageMode** method.

(v) Image rotation direction (**Rotate**)

Enumerated type constants used for image rotation direction are shown in the following table.

Table 4-32 Image Rotation Direction (Rotate)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_ROTATE_NONE	No rotation	Not supported	Supported
SII_PM_ROTATE_180	Rotate 180 degrees	Not supported	Supported

(w) Image scaling (**ImageScale**)

Enumerated type constants used for image scaling are shown in the following table.

Table 4-33 Image Scaling (ImageScale)

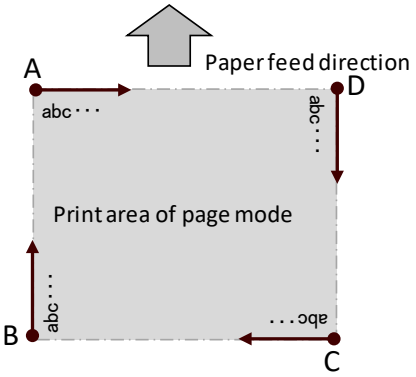
Constant Name	Description	Target	
		Mobile	POS
SII_PM_IMAGE_SCALE_NONE	No scaling	Not supported	Supported
SII_PM_IMAGE_SCALE_WIDTH_FIT	Scale to fit print width	Not supported	Supported

(x) Print direction (**Direction**)

Enumerated type constants used for print direction in page mode are shown in the following table.

Table 4-34 Print Direction (Direction)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_DIRECTION_LEFT_TO_RIGHT	Starting point: Upper left (A on the figure), Print direction: Left to Right	Not supported	Supported
SII_PM_DIRECTION_BOTTOM_TO_TOP	Starting point: Left below (B on the figure), Print direction: Below to Upper	Not supported	Supported
SII_PM_DIRECTION_RIGHT_TO_LEFT	Starting point: Right below (C on the figure), Print direction: Right to Left	Not supported	Supported
SII_PM_DIRECTION_TOP_TO_BOTTOM	Starting point: Upper right (D on the figure), Print direction: Upper to Below	Not supported	Supported



(y) Line style (**LineStyle**)

Enumerated type constants used for line style in page mode are shown in the following table.

Table 4-35 Line Style (LineStyle)

Constant Name	Description	Target	
		Mobile	POS
SII_PM_LINESTYLE_THIN	Thin solid line (2 dots)	Not supported	Supported
SII_PM_LINESTYLE_MEDIUM	Medium solid line (4 dots)	Not supported	Supported
SII_PM_LINESTYLE_THICK	Thick solid line (8 dots)	Not supported	Supported

(5) Method Detals

(a) Common method to standard mode and page mode

The following methods are valid in standard mode and page mode. Standard mode is set immediately after **`connect`** method is executed.

`init` — Instance

Target	Mobile printer / POS printer
Syntax	`- (id) init;`
Description	This method initializes the instance of **`SIIPrinterManager`** class.
Return	When succeeded, the initialized instance of **`SIIPrinterManager`** class is returned. When failed, nil is returned.
Using Example	`SIIPrinterManager *printerManager =[[SIIPrinterManager alloc] init];`

`connect` — Start communicating with a printer (Bluetooth)

Target	Mobile printer / POS printer
Syntax	`- (void) connect:(NSInteger)printerModel address:(NSString)address portType:(NSInteger)portType;`
Parameter	**`printerModel`** Printer model constant for Bluetooth connection **`address`** Bluetooth device name (Bluetooth accessory) Example: "DPU-S245" **`portType`** Port type Example: **SII_PM_PRINTER_PORT_TYPE_BLUETOOTH**
Description	This method starts communication between iOS device and paired printer through Bluetooth connection. Call this method before using other methods of this class. This method connects to the paired Bluetooth device (Bluetooth accessory) specified by **`address`** parameter. See Table 4-5 Printer Model Constant for available setting in **`printerModel`** parameter. For **`portType`** parameter, specify **SII_PM_PRINTER_PORT_TYPE_BLUETOOTH**. In order to operate this library properly, printer settings may be needed to change at the connection in this method. The Library disconnects Bluetooth connection with printer by executing **`disconnect`** method.
Error	**`SIIPrinterException`** is thrown when an error occurs while this method is to be called. See "**4.3.4 `SIIPrinterException` Class**" for details on the error.
Note	A concurrent connection from multiple apps to one printer is not supported.

connect — Start communicating with a printer (TCP/IP)

Target: POS printer

Syntax:
```
- (void) connect:(NSInteger)printerModel
                 address:(NSString)address
                 portType:(NSInteger)portType;
```

Parameter:

printerModel	Printer model constant for Ethernet connection
address	IP address Example: "192.168.0.1"
portType	Port type Example: **SII_PM_PRINTER_PORT_TYPE_TCP**

Description: This method is valid only for POS printer. This method starts communication between iOS device and printer connected to the same network through TCP/IP connection. Call this method before using other methods of this class.

This method connects to the IP address specified by **address** parameter. TCP port number 9100 is used for communication. Also, printer default settings are executed based on **printerModel** parameter which is specified at the connection. See Table 4-5 Printer Model Constant for available printer model constant in **printerModel** parameter.

For the portType parameter, specify **SII_PM_PRINTER_PORT_TYPE_TCP**.

In order to operate a printer properly, printer settings may be needed to change at the connection in this method.

<Introduction to creating/discarding a socket during connection with TCP/IP in the Library>

After **connect** method, the library retains the created socket until **disconnect** method executed. And connecting to the same printer from other applications is not possible until **disconnect** method executed.

Based on the completion time of data transmission to the printer, the socket is once discarded after elapsing the socket keeping time set by **socketKeepingTime**. Then the new socket is created immediately and used for the next connection.
If the printer is receiving a connection request from another host on the same network at the time of discarding the socket, the reconnection may fail because the printer establishes communication with the host.

Error: **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

Note: A concurrent connection from multiple apps to one printer is not supported.

disconnect — Disconnect a printer

Target: Mobile printer / POS printer

Syntax:
```
- (void) disconnect;
```

Description: This method disconnects communication with a printer.

Error: **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

Note Execute this method after transmitting the data.
If executing this method before data transmission is completed, the following problems may be occurred:

- A part of the data may be deleted.
- In Bluetooth connection, when **`connect`** method is executed soon after executing this method, the connection between iOS device and the printer becomes abnormal and may not transmit/receive the data.

To verify data transmission completion, it is recommended to obtain execution response by **SII_PM_PRINTER_RESPONSE_REQUEST** of **`getPrinterResponse`** method before executing this method.
When this method is executed without executing **`getPrinterResponse`** method in your program, evaluate your program to confirm no problems arise.

`openDrawer` Open cash drawer

Target POS printer

Syntax
```
- (void) openDrawer:(DrawerNum)drawerNum
                 onOffTime:(PulseWidth)onOffTime;
```

Parameter

`drawerNum`	Drawer number
`onOffTime`	pulse width

Description This method is valid only for POS printer. This method opens the specified cash drawer.

See Table 4-11 Drawer Number (**DrawerNum**) for available setting in **`drawerNum`** parameter.

See Table 4-12 Activation Pulse Width (**PulseWidth**) for available setting in **`onOffTime`** parameter.

Error **`SIIPrinterException`** is thrown when an error occurs while this method is to be called.
See "**4.3.4 `SIIPrinterException` Class**" for details on the error.

`buzzer` Sound buzzer

Target POS printer

Syntax
```
- (void) buzzer:(NSInteger)onTime
                 offTime:(NSInteger)offTime;
```

Parameter

`onTime`	Buzzer On time (millisecond)
`offTime`	Buzzer Off time (millisecond)

Description This method is valid only for POS printer. This method sounds the buzzer.

The valid range of **`onTime`** and **`offTime`** parameter is from 0 to 510.

Error **`SIIPrinterException`** is thrown when an error occurs while this method is to be called.
See "**4.3.4 `SIIPrinterException` Class**" for details on the error.

externalBuzzer — Sound external buzzer

Target: Mobile printer / POS printer

Syntax:
```
- (void) externalBuzzer:(BuzzerPattern)buzzerPattern
                 buzzerCount:(NSInteger)buzzerCount;
```

Description: This method is not supported. **SIIPrinterException** is thrown when this method is executed.

getStatus — Obtain printer status

Target: Mobile printer / POS printer

Syntax:
```
- (void) getStatus:(NSInteger[])buf;
```

Parameter: **buf** — Status obtained from a printer

Description: This method obtains printer status. Status obtained from a printer is stored to an NSInteger array. Contents of status vary between Mobile printer and POS printer.

When specified **printerModel** parameter while executing **connect** method is **SII_PM_PRINTER_MODEL_DPU_S245** or **SII_PM_PRINTER_MODEL_DPU_S445**, see the contents of Mobile printer status. When the **printerModel** parameter is **SII_PM_PRINTER_MODEL_RP_D10** or **SII_PM_PRINTER_MODEL_RP_E10**, see the POS printer status.

The Mobile printer status is shown in the following table.
When the connection failed, the Mobile printer status is shown in 0x00000000.

Table 4-36 Printer Status (Mobile Printer)

Bit	Function	Value	
		0	1
0	Out-of-paper error	OK	Error
1	Head up error	OK	Error
2	Vp voltage malfunction	OK	Error
3	Thermal head temperature error	OK	Error
4	Function Setting error	OK	Error
5	Battery voltage state	See table below	
6			
7	Reserved	-	Fixed
8 to 31	Reserved	Fixed	-

Bit 6	Bit 5	Battery Voltage State
0	0	8.0 V or higher
0	1	7.5 or higher to lower than 8.0 V
1	0	7.0 or higher to lower than 7.5 V
1	1	Lower than 7.0 V

The POS printer status is shown in the following table.
When the connection failed, the POS printer status is shown in 0x80000000.

Table 4-37 Printer Status (POS Printer)

Bit	Function	Value	
		0	1
0	VP voltage error	OK	Error
1	Hardware error	OK	Error
2	Head temperature error	OK	Error
3	Autocutter error	OK	Error
4	Out-of-paper error	OK	Error
5	Paper-near-end sensor error*1	OK	Error
6	Paper jam error while detecting mark*1	OK	Error
7	Cover open error	OK	Error
8	FEED Switch status	OFF	ON
9	Reserved	Fixed	-
10	Paper feed status	Stop	Operating
11	Return-waiting status	No	Yes
12	Reserved	Fixed	-
13	Reserved	-	Fixed
14	Reserved	-	Fixed
15	Drawer switch input status	Low	High
16	FLASH memory rewriting	No	Yes
17	Peripheral device selection	Printer	Other
18 to 31	Reserved	-	Fixed

*1: Supported only in RP-E10. In RP-D10, it is always OK (value: 0).

Error **`SIIPrinterException`** is thrown when an error occurs while this method is to be called.
See "**4.3.4 `SIIPrinterException` Class**" for details on the error.

`abort` Abort the waiting state of a printer

Target Mobile printer / POS printer

Syntax
```
- (void) abort;
```

Description When sending of image file by **`sendDataFile`** method is aborted, a printer does not accept other processes until specified image file is received completely. (Method or sent data are misinterpreted and recognized as part of the image file.)
To solve this situation, use this method to abort the waiting state of a printer.
Note that when executing this method, a part of unprocessed image file may be printed.

Error **`SIIPrinterException`** is thrown when an error occurs while this method is to be called.
See "**4.3.4 `SIIPrinterException` Class**" for details on the error.

registerLogo — Register logo (image) to a printer

Target: Mobile printer / POS printer

Syntax:

```
(a) - (void) registerLogo:(NSString *)fileName
                   logoId:(NSString *)logoId
                   dithering:(Dithering)dithering;

(b) - (void) registerLogo:(NSString *)fileName
                   logoId:(NSString *)logoId;
```

Parameter:

fileName	File path of image file to register as logo
logoId	Logo ID to register
dithering	Dithering

Description: This method registers image file specified by **fileName** parameter to a printer as a logo.

In syntax (a), dithering can be specified.
In syntax (b), dithering is fixed to be disabled.

File extension for image file supported by **fileName** parameter is .bmp, .jpg, .jpeg, or .png.

The valid range of **logoId** parameter varies depends on the target printer.

- For Mobile printer, the valid range is a character string indicating the value from 0 to 127.
- For POS printer, specify by 2 characters. The valid characters are ASCII character code from 20h (space) to 7Eh (tilde) such as alphanumeric ('0' to '9', 'A' to 'Z', 'a' to 'z').

See Table 4-13 Dithering (**Dithering**) for available setting in **dithering** parameter.

Error: **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

unregisterLogo — Delete specified logo (image) on a printer

Target: Mobile printer / POS printer

Syntax:

```
- (void) unregisterLogo:(NSString *)logoId;
```

Parameter:

logoId	Logo ID to delete

Description: This method deletes the logo (image) registered by **registerLogo** method (for Mobile printer). Specify registered logo ID in **logoId** parameter. The valid range of **logoId** parameter varies depends on the target printer.

For Mobile printer, the valid range is a character string indicating the value from 0 to 127.

For POS printer, specify by 2 characters. The valid characters are ASCII character code from 20h (blank) to 7Eh (tilde) such as alphanumeric ('0' to '9', 'A' to 'Z', 'a' to 'z').

Error: **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

registerStyleSheet — Register style sheet to a printer

Target: POS printer

Syntax:
```
- (void) registerSytleSheet:(NSString *)fileName
                     cssId:(NSInteger)cssId;
```

Parameter:

fileName	CSS file path to register as style sheet
cssId	Style sheet number to register

Description: This method is valid only for POS printer. This method registers CSS file specified by **fileName** parameter to a printer. Maximum number of registerable style sheet is 4 sheets.

Style sheet supported by **fileName** parameter is that style sheet language is written in CSS (cascading style sheets), and that file extension is .css.
Maximum number of style registerable to 1 CSS file is 64.
Also, the valid range of **cssId** parameter is from 1 to 4.

See "6.5.13 Tag Processing Mode" in "RP-D10 SERIES THERMAL PRINTER TECHNICAL REFERENCE" or "RP-E10 SERIES THERMAL PRINTER TECHNICAL REFERENCE" for more details about style sheet.

Error: **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

unregisterStyleSheet — Delete specified style sheet on a printer

Target: POS printer

Syntax:
```
- (void) unregisterStyleSheet:(NSInteger)cssId;
```

Parameter:

cssId	Style sheet number to delete

Description: This method is valid only for POS printer. This method deletes the style sheet registered by **registerStyleSheet** method.
Specify the registered style sheet number for **cssId** parameter.
The valid range of **cssId** parameter is from 1 to 4.

Error: **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

resetPrinter — Printer hardware reset

Target: POS printer

Syntax:
```
- (void) resetPrinter;
```

Description: This method is valid only for POS printer.

For TCP/IP connection:
This method resets a connected printer by using the SII original command (reset command) to TCP port number 26100.

For Bluetooth connection:
This method resets the printer hardware by using the printer command "Printer Reset". When using this method, set iOS Auto Connection in Function Setting of the printer to "Enabled". When iOS Auto Connection is set to "Disabled", this method fail to reconnect after reset and **SIIPrinterException** is thrown.
This method needs about 10 seconds until complete to reconnect after reset executing. When calling this method, set **receiveTimeout** property to enough time to complete this method.

After executing this method, connection to a printer is maintained.

Error **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

getPrinterResponse — Obtaining various responses from a printer

Target Mobile printer / POS printer

Syntax
```
- (void) getPrinterResponse:(NSInteger)responseId
                      param:(NSObject *)param
                   response:(void *)response;
```

Parameter

responseId	Response type constant
param	Response code of execution response request
response	Buffer that stores obtained response data (Buffer type varies depending on a response type constant.)

Description This method stores the response data specified by **responseId** parameter to the object specified by **response** parameter. See Table 4-7 Response Type Constant for available setting for **responseId** parameter. Contents of obtainable response from a printer vary between Mobile printer and POS printer.

When specified **printerModel** parameter while executing **connect** method is **SII_PM_PRINTER_MODEL_DPU_S245** or **SII_PM_PRINTER_MODEL_DPU_S445**, see the contents of obtained response for Mobile printer. When the **printerModel** parameter is **SII_PM_PRINTER_MODEL_RP_D10** or **SII_PM_PRINTER_MODEL_RP_E10**, see the contents of obtained response for POS printer.

Contents of obtained response for Mobile printer are shown in the following table.

Table 4-38 Contents of Obtained Response (Mobile Printer)

Constant Name	Description
SII_PM_PRINTER_RESPONSE_REQUEST	Obtains "Execution response request". Specify 0 to 15 (00h to 0Fh) of NSData to **param** parameter. Specify NSInteger array with the length of 1 to **response** parameter. When responses are obtained successfully, 80 to 95 (50h to 5Fh) is stored to the **response** parameter.
SII_PM_PRINTER_RESPONSE_USER_AREA	Obtains "Send remaining capacity of user area". Specify nil to **param** parameter. Specify NSInteger array with the length of 1 to **response** parameter. When responses are obtained successfully, remaining user area is stored to the **response** parameter in byte value.

Constant Name	Description
SII_PM_PRINTER_RESPONSE_BATTERY_STATUS	Obtains "Battery voltage state". Specify nil to **param** parameter. Specify NSInteger array with the length of 1 to **response** parameter. When responses are obtained successfully, the battery status value is stored to the **response** parameter. Meanings of battery status values are as follows.

Battery Status Value	Battery Voltage State
0	8.0 V or higher
1	7.5 or higher to lower than 8.0 V
2	7.0 or higher to lower than 7.5 V
3	Lower than 7.0 V

Contents of obtained response for POS printer are shown in the following table.

Table 4-39 Contents of Obtained Response (POS Printer)

Constant Name	Description
SII_PM_PRINTER_RESPONSE_REQUEST	Obtains "Execution response request". Specify 0 to 15 (00h to 0Fh) of NSData to **param** parameter. Specify NSInteger array with the length of 1 to **response** parameter. When responses are obtained successfully, 128 to 143 (80h to 8Fh) is stored to **response** parameter.
SII_PM_PRINTER_RESPONSE_USER_AREA	Obtains "Send remaining capacity of user area". Specify nil to **param** parameter. Specify NSInteger array with the length of 1 to **response** parameter. When responses are obtained successfully, remaining user area is stored to **response** in byte value.
SII_PM_PRINTER_RESPONSE_ARRANGE_USER_AREA	Obtains "Send remaining capacity of user area after defragment". Specify nil to **param** parameter. Specify NSInteger array with the length of 1 to **response** parameter. When responses are obtained successfully, remaining memory after user area defragment is stored to **response** parameter in byte value.
SII_PM_PRINTER_RESPONSE_NV_GRAPHICS	Obtains "Send NV graphics memory capacity". Specify nil to **param** parameter. Specify NSInteger array with the length of 1 to **response** parameter. When responses are obtained successfully, NV graphics memory capacity is stored to **response** in byte value.
SII_PM_PRINTER_RESPONSE_KEY_CODE	Obtains "Send key code list of defined NV graphics". Specify nil to **param** parameter. Specify NSMutableArray array to **response** parameter. When responses are obtained successfully, the key code of NV graphics is stored to **response** parameter in string array.

Constant Name	Description
SII_PM_PRINTER_RESPONSE_FIRMWARE_VERSION	Obtains "Firmware version". Specify nil to **param** parameter. Specify NSMutableArray array to **response** parameter. When responses are obtained successfully, the firmware version is stored to **response** parameter in string array.
SII_PM_PRINTER_RESPONSE_MAINTENANCE_COUNTER	Obtains "Send maintenance counter". Specify nil to **param** parameter. Specify NSMutableArray array to **response** parameter. When the response is retrieved successfully, the maintenance counter value is stored as an NSNumber object as shown below. response[0] : Number of paper feed lines (unit: 100 dot-line) response[1] : Number of thermal head activation times (unit: 100 dot-line) response[2] : Number of autocutter drive times response[3] : Drive time of printer (unit: minute) response[4] : Number of paper feed lines (integrated value) (unit: 100 dot-line) response[5] : Number of thermal head activation times (integrated value) (unit: 100 dot-line) response[6] : Number of autocutter drive times (integrated value) response[7] : Drive time of printer (integrated value) (unit: minute)
SII_PM_PRINTER_RESPONSE_PRINTER_IDENTIFIER	Obtains "Send printer ID". Specify nil to **param** parameter. Specify NSMutableArray array to **response** parameter. When the response is retrieved successfully, **SIIPrinterIdentifier** class object is stored in the first element of response. Example: response[0] = "(**SIIPrinterIdentifier** class object)" See "**4.3.3 SIIPrinterIdentifier Class**" for the **SIIPrinterIdentifier** class object.

Error **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

startDiscoveryPrinter — Start printer search (Bluetooth)

Target Mobile printer / POS printer

Syntax
```
- (void)startDiscoveryPrinter:(NSPredicate *)predicate
    completion:(EABluetoothAccessoryPickerCompletion)completion;
```

Parameter

predicate	Specify nil.
completion	Event completion of `EABluetoothAccessoryPickerCompletion`

Description This method searches Bluetooth device (Bluetooth accessory). This method calls `showBluetoothAccessoryPickerWithNameFilter` of `EAAccessoryManager` internally. When executing this method, paring with Bluetooth device is enable in the displayed window.

Specify ^ (`NSError *error`) to **completion** parameter in order to receive event completion of `EABluetoothAccessoryPickerCompletion`.

(cf.) Statement of `EABluetoothAccessoryPickerCompletion`

```
typedef void(^EABluetoothAccessoryPickerCompletion)
(NSError *error);
```

Error **`SIIPrinterException`** is thrown when an error occurs while this method is to be called.
See "**4.3.4 `SIIPrinterException` Class**" for details on the error.

`startDiscoveryPrinter` Start printer search (TCP/IP)

Target POS printer

Syntax

```
- (void)startDiscoveryPrinter:(NSInteger)retryCount
        timeout:(NSInteger)timeout
        completion:(SIIDiscoveryPrinterCompletion)completion;
```

Parameter

`retryCount`	Retry count (times)
`timeout`	Searching timeout period (millisecond: ms)
`completion`	Event completion of printer search

Description This method is valid only for POS printer. This method is available only when specified **`portType`** parameter is **SII_PM_PRINTER_PORT_TYPE_TCP** while executing **`connect`** method. This method sends local broad cast packet and searches for SII printer connecting to the same network. Completion or cancellation of searching is notified to a block specified in **`completion`** parameter as an event.

This method sends the local broad cast packet the number of times set in **`retryCount`** parameter. The valid range of **`retryCount`** parameter is 1 to 5. When the value is specified less than 1, the number is set to 1. When the value is specified more than 5, the number is set to 5.

This method waits for response from the printer until the search timeout period set by **`timeout`** parameter. The valid range of **`timeout`** parameter is 3000 to 60000. When the value is specified less than 3000, the period is set to 3000. When the value is specified more than 60000, the period is set to 60000. The searching printer information can be obtained by **`getFoundPrinter`** method.

The definition of `SIIDiscoveryPrinterCompletion` is as follows.

```
typedef void(^SIIDiscoveryPrinterCompletion)(NSArray
*printerList);
```

In order to use this method, LAN interface F/W version needs to be 1.13.01 or later.

Error **`SIIPrinterException`** is thrown when an error occurs while this method is to be called.
See "**4.3.4 `SIIPrinterException` Class**" for details on the error.

`cancelDiscoveryPrinter` Cancel printer search

Target POS printer

Syntax

```
- (void)cancelDiscoveryPrinter;
```

Description This method is valid only for POS printer. This method is available only when specified **`portType`** parameter is **SII_PM_PRINTER_PORT_TYPE_TCP** while executing **`connect`** method. This method cancels executing **`startDiscoveryPrinter`** method. Cancellation of searching is notified to a block specified in **`completion`** parameter as an event.

`getFoundPrinter` Obtain searched printer information

Target POS printer

Syntax `- (NSArray *)getFoundPrinter;`

Parameter This method is valid only for POS printer. This method is available only when specified **`portType`** parameter is **SII_PM_PRINTER_PORT_TYPE_TCP** while executing **`connect`** method. This method returns the printer information searched by printer search as `NSArray` type.
See "4.3.2 **`SIIPrinterInfo`** Class" for the details of printer information.

Return value Printer information of `NSArray` type

`getVersion` Get SDK version

Target Mobile printer / POS printer

Syntax `- (NSString *)getVersion;`

Description Gets the SDK version as a character string.
This property can be executed regardless of whether isConnect is YES or NO.

Return value SDK version character string (Example: When the SDK version is Ver.1.0.0, the return value is "1.0.0")

`controlTransaction` Start/End batch processing

Target Mobile printer / POS printer

Syntax `- (void) controlTransaction:(TransactionFunction) control;`

Parameter **`control`** Batch processing selection

Description The procedure of batch processing is as follows:
See Table 4-14 Batch Processing Selection (**TransactionFunction**) for the details about configurable values.
(1) Start batch processing.
Specify **SII_PM_TRANSACTION_START**.

(2) Execute the method.
In the case of the batch processing target method, buffering of transmission data is started.
The transmission data of the batch processing target method executed during buffering is buffered in the transmission buffer without being sent to the printer.
The maximum size of transmission data to be buffered is system dependent.
If the buffered transmission data exceeds the maximum size, the batch processing target method at the time of exceeding becomes an error. If an error occurs, the transmission data up to the error is retained.
As for the retained transmission data, finish the batch processing in step (3).
In the case of a method other than the batch processing target method, transmission data is immediately executed without being buffered.

(3) Finish batch processing.

When **SII_PM_TRANSACTION_PRINT** is specified, the buffered transmission data is sent to the printer. The buffered transmission data is retained even after sent to the printer.

The retained transmission data is discarded by any of the following:

・Specify **SII_PM_TRANSACTION_CLEAR**

・Specify **SII_PM_TRANSACTION_START**

・Execute **`disconnect`** method

The batch processing target methods are as follows:

- **`sendText`**
- **`sendTextEx`**
- **`printBarcode`**
- **`printPDF417`**
- **`printQRcode`**
- **`printDataMatrix`**
- **`printMaxiCode`**
- **`cutPaper`**[*1]
- **`openDrawer`**[*1]
- **`buzzer`**[*1]
- **`sendBinary`**
- **`sendDataFile`**
- **`sendImage`**
- **`printPDF`**[*1]
- **`printLogo`**[*2]
- **`enterPageMode`**[*1]
- **`exitPageMode`**[*1]
- **`setPageModeArea`**[*1]
- **`setPageModeDirection`**[*1]
- **`setPageModeLineSpacing`**[*1]
- **`printPageMode`**[*1]
- **`printPageModeText`**[*1]
- **`printPageModeTextEx`**[*1]
- **`printPageModeBarcode`**[*1]
- **`printPageModePDF417`**[*1]
- **`printPageModeQRcode`**[*1]
- **`printPageModeDataMatrix`**[*1]
- **`printPageModeMaxiCode`**[*1]
- **`sendPageModeBinary`**[*1]
- **`printPageModeImageFile`**[*1]
- **`printPageModeRectangle`**[*1]
- **`printPageModeLine`**[*1]
- **`printPageModeLogo`**[*1*2]

*1: This method is not supported by Mobile printer.

*2: The method under batch processing does not notify the error even when the registered logo does not exist.

Error

`SIIPrinterException` is thrown when an error occurs while this method is to be called.

See "**4.3.4 `SIIPrinterException` Class**" for details on the error.

(b) Dedicated method for standard mode

The following methods are valid in standard mode. **SIIPrinterException** is thrown when the dedicated method for standard mode are executed in page mode.

sendText — Send text data

Target: Mobile printer / POS printer

Syntax:
```
- (void) sendText:(NSString *)text;
```

Parameter:

text	Text data sent to a printer

Description: This method sends the text data specified by **text** parameter to a printer. Data size that can be specified at 1 time is 16KB (16384 bytes).

This method encodes the specified text data to printable text data based on **internationalCharacter** property and **codePage** property, and then sends it to the printer.

In this method, the line spacing code is not added to the end of text data.

Error: **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

sendTextEx — Send format specified text data

Target: Mobile printer

Syntax:
```
- (void) sendTextEx:(NSString *)text
                bold:(CharacterBold)bold
                underline:(CharacterUnderline)underline
                font:(CharacterFont)font
                scale:(CharacterScale)scale;
```

Parameter:

text	Text data to send to a printer
bold	Bold print
underline	Underline
font	Character font
scale	Character scale

Description: This method is valid only for Mobile printer. This method encodes the specified text data to printable text data based on **internationalCharacter** property and **codePage** property, and then sends it to the printer. Data size that can be specified at one time is 16KB (16384 bytes).

See Table 4-15 Bold Print (**CharacterBold**) for available setting in **bold** parameter.

See Table 4-16 Underline (**CharacterUnderline**) for available setting in **underline** parameter.

See Table 4-19 Character Font (**CharacterFont**) for available setting in **font** parameter.

See Table 4-20 Character Scale (**CharacterScale**) for available setting in **scale** parameter.

In this method, no line feed code is added at the end of text data.

Error: **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

sendTextEx — Send format specified text data

Target: POS printer

Syntax:

```
(a) - (void) sendTextEx:(NSString *)text
                        bold:(CharacterBold)bold
                        underline:(CharacterUnderline)underline
                        reverse:(CharacterReverse)reverse
                        font:(CharacterFont)font
                        scale:(CharacterScale)scale
                        alignment:(PrintAlignment)alignment;

(b) - (void) sendTextEx:(NSString *)text
                        bold:(CharacterBold)bold
                        underline:(CharacterUnderline)underline
                        reverse:(CharacterReverse)reverse
                        inversion:(CharacterInversion)inversion
                        font:(CharacterFont)font
                        scale:(CharacterScale)scale
                        alignment:(PrintAlignment)alignment;
```

Parameter:

text	Text data to send to a printer
bold	Bold print
underline	Underline
reverse	Reverse print
inversion	Inversion print
font	Character font
scale	Character scale
alignment	Alignment

Description: This method is valid only for POS printer. This method encodes the specified text data to printable text data based on **internationalCharacter** property and **codePage** property, and then sends it to the printer. Data size that can be specified at one time is 16KB (16384 bytes).

See Table 4-15 Bold Print (**CharacterBold**) for available setting in **bold** parameter.

See Table 4-16 Underline (**CharacterUnderline**) for available setting in **underline** parameter.

See Table 4-17 Reverse Print (**CharacterReverse**) for available setting in **reverse** parameter.

See Table 4-18 Character Inversion (**CharacterInversion**) for available setting in **inversion** parameter.

See Table 4-19 Character Font (**CharacterFont**) for available setting in **font** parameter.

See Table 4-20 Character Scale (**CharacterScale**) for available setting in **scale** parameter.

See Table 4-21 Alignment (**PrintAlignment**) for available setting in **alignment** parameter.

In this method, no line feed code is added at the end of text data.

Error: **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

printBarcode — Print barcode

Target: Mobile printer / POS printer

Syntax:

```
(a) - (void) printBarcode:(BarcodeSymbol)barcodeSymbol
                     text:(NSString *)text
                     moduleSize:(ModuleSize)moduleSize
                     moduleHeight:(NSInteger)moduleHeight
                     hriPosition:(HriPosition)hriPosition
                     hriFont:(CharacterFont)hriFont
                     alignment:(PrintAlignment)alignment;

(b) - (void) printBarcode:(BarcodeSymbol)barcodeSymbol
                     text:(NSString *)text
                     moduleSize:(ModuleSize)moduleSize
                     moduleHeight:(NSInteger)moduleHeight
                     hriPosition:(HriPosition)hriPosition
                     hriFont:(CharacterFont)hriFont
                     alignment:(PrintAlignment)alignment
                     nwRatio:(NwRatio)nwRatio;

(c) - (void) printBarcode:(BarcodeSymbol)barcodeSymbol
                     data:(NSData*)data
                     moduleSize:(ModuleSize)moduleSize
                     moduleHeight:(NSInteger)moduleHeight
                     hriPosition:(HriPosition)hriPosition
                     hriFont:(CharacterFont)hriFont
                     alignment:(PrintAlignment)alignment;

(d) - (void) printBarcode:(BarcodeSymbol)barcodeSymbol
                     text:(NSString *)text
                     moduleSize:(ModuleSize)moduleSize
                     alignment:(PrintAlignment)alignment;
```

Parameter:

barcodeSymbol	Barcode symbol
text (data)	Text data to send to a printer
moduleSize	Barcode width
moduleHeight	Barcode height (dot)
hriPosition	HRI character print position
hriFont	HRI character font
alignment	Alignment
nwRatio	N:W ratio

Description: This method executes barcode print.

For format (a), **nwRatio** parameter is fixed to wide width type 3 for Mobile printers and 1:2.5 for POS printers.
The format (d) is not supported.

See Table 4-22 Barcode Symbol (**BarcodeSymbol**) for available setting in **barcodeSymbol** parameter.

The available input conditions for barcode in **text (data)** parameter are as follows. See Table 4-40 for Mobile printers and Table 4-41 for POS printers.

Table 4-40 Barcode Input Conditions for Mobile Printer

Barcode	Number of Data	Inputtable Data Character String (Data)	Remarks
UPC-A	11 characters	'0' to '9'	
UPC-E	11 characters	'0' to '9'	
EAN13 JAN13	12 characters	'0' to '9'	
EAN8 JAN8	7 characters	'0' to '9'	
CODE39	1 to 254 characters	'0' to '9' 'A' to 'Z' ' ', '$', '%', '+', '-', '.', '/'	Start code and stop code ('*') are automatically added.
CODE128	2 to 255 characters	(0x00 to 0x66)	Input data with 0x67 or more at the end.
ITF	2 to 254 characters (However, an even number)	'0' to '9'	
CODABAR	1 to 254 characters	'0' to '9' '$', '+', '-', '.', '/', ':'	It is needed to specify one of 'A' to 'D' at the beginning and end.

Table 4-41 Barcode Input Conditions for POS Printer

Barcode	Number of Data	Inputtable Data Character String (Data)	Remarks
UPC-A	11 to 12 characters	'0' to '9'	
UPC-E	11 to 12 characters	'0' to '9'	
EAN13 JAN13	12 to 13 characters	'0' to '9'	
EAN8 JAN8	7 to 8 characters	'0' to '9'	
CODE39	1 to 150 characters	'0' to '9' 'A' to 'Z' ' ', '$', '%', '+', '-', '.', '/'	Start code and stop code ('*') are automatically added.
CODE93	1 to 150 bytes	(0x00 to 0x2E)	Input data with 0x2F or more at the end.
CODE128	2 to 150 bytes	(0x00 to 0x66)	When inputting the start code (0x67 to 0x69) of the CODE128 code set. Input data with 0x67 or more at the end.
		(0x00 to 0x7F)	When starting with a CODE128 special code start code ("{A", "{B", "{C").
ITF	2 to 150 characters (However, an even number)	'0' to '9'	

Barcode	Number of Data	Inputtable Data Character String (Data)	Remarks
CODABAR	1 to 150 characters	'0' to '9' '$', '+', '-', '.', '/', ':'	It is needed to specify one of 'A' to 'D' at the beginning and end.
EAN13 add-on JAN13 add-on	Add-on 2: 14 to 15 characters Add-on 5: 17 to 18 characters	'0' to '9'	

See Table 4-23 Module Size (**ModuleSize**) for available setting in **`moduleSize`** parameter.

The valid range of **`moduleHeight`** parameter is from 1 to 255.

See Table 4-24 HRI Character Print Position (**HriPosition**) for available setting in **`hriPosition`** parameter.

See Table 4-19 Character Font (**CharacterFont**) for available setting in **`hriFont`** parameter.

See Table 4-21 Alignment (**PrintAlignment**) for available setting in **`alignment`** parameter.

See Table 4-25 N:W Ratio (**NwRatio**) for available setting in **`nwRatio`** parameter. Depending on the relationship between **`nwRatio`** parameter and **`moduleSize`** parameter, the wide element width is set in the following tables. See Table 4-42 N:W Ratio for Mobile Printer for Mobile printer, and Table 4-43 N:W Ratio for POS Printer for POS printer.

Table 4-42 N:W Ratio for Mobile Printer

`moduleSize`	`nwRatio`			
	SII_PM_BARCODE_WIDE_WIDTH_1	**SII_PM_BARCODE_WIDE_WIDTH_2**	**SII_PM_BARCODE_WIDE_WIDTH_3**	**SII_PM_BARCODE_WIDE_WIDTH_4**
SII_PM_BARCODE_MODULE_WIDTH_2	0.625 mm (5 dots)	0.750 mm (6 dots)	0.750 mm (6 dots)	0.750 mm (6 dots)
SII_PM_BARCODE_MODULE_WIDTH_3	0.875 mm (7 dots)	1.000 mm (8 dots)	1.125 mm (9 dots)	1.125 mm (9 dots)
SII_PM_BARCODE_MODULE_WIDTH_4	1.125 mm (9 dots)	1.250 mm (10 dots)	1.375 mm (11 dots)	1.500 mm (12 dots)

Table 4-43 N:W Ratio for POS Printer

`moduleSize`	`nwRatio`		
	SII_PM_NWRATIO_1TO2	**SII_PM_NWRATIO_1TO2_5**	**SII_PM_NWRATIO_1TO3**
SII_PM_BARCODE_MODULE_WIDTH_2	0.500 mm (4 dots)	0.625 mm (5 dots)	0.750 mm (6 dots)
SII_PM_BARCODE_MODULE_WIDTH_3	0.750 mm (6 dots)	1.000 mm (8 dots)	1.125 mm (9 dots)
SII_PM_BARCODE_MODULE_WIDTH_4	1.000 mm (8 dots)	1.250 mm (10 dots)	1.500 mm (12 dots)

moduleSize	nwRatio		
	SII_PM_ NWRATIO_1TO2	SII_PM_ NWRATIO_1TO2_5	SII_PM_ NWRATIO_1TO3
SII_PM_BARCODE_MODULE_WIDTH_5	1.250 mm (10 dots)	1.625 mm (13 dots)	1.875 mm (15 dots)
SII_PM_BARCODE_MODULE_WIDTH_6	1.500 mm (12 dots)	1.875 mm (15 dots)	2.250 mm (18 dots)

Error **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

Reference See "Appendix B Barcode Size List" for details of the barcode size.

printPDF417 Print PDF417

Target Mobile printer / POS printer

Syntax
```
(a) - (void) printPDF417:(NSString *)text
                errorCorrection:(ErrorCorrection)errorCorrection
                row:(NSInteger)row
                column:(NSInteger)column
                moduleSize:(ModuleSize)moduleSize
                moduleHeight:(NSInteger)moduleHeight
                alignment:(PrintAlignment)alignment
                pdf417Symbol:(Pdf417Symbol)pdf417Symbol;

(b) - (void) printPDF417:(NSString *)text
                errorCorrection:(ErrorCorrection)errorCorrection
                row:(NSInteger)row
                column:(NSInteger)column
                moduleSize:(ModuleSize)moduleSize
                moduleHeight:(NSInteger)moduleHeight
                alignment:(PrintAlignment)alignment;
```

Parameter

text	Barcode data to send a printer
errorCorrection	Error correction level
row	The number of row
column	The number of columns in data area
moduleSize	Nominal Fine Element Width
moduleHeight	Module height (dot)
alignment	Alignment
pdf417Symbol	Symbol of PDF417

Description This method prints PDF417. **pdf417Symbol** parameter for syntax (b) is fixed to standard PDF417.

See Table 4-26 Error Correction Level (**ErrorCorrection**) for available setting in **errorCorrection** parameter.

The valid range of **row** parameter is from 0, 3 to 90. When 0 is specified, the number of row is automatically set.

The valid range of **column** parameter is from 0 to 30. When 0 is specified, the number of row in the data area is automatically set.

See Table 4-23 Module Size (**ModuleSize**) for available setting in **moduleSize** parameter.

The valid range of **moduleHeight** parameter is from 2 to 127. When the module height is set smaller, some barcode scanners may not read it. Set 3 or more for normal use.

See Table 4-21 Alignment (**PrintAlignment**) for available setting in **alignment** parameter.

See Table 4-27 PDF417 Symbol (**Pdf417Symbol**) for available setting in **pdf417Symbol** parameter.

Error **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

Reference See "Appendix B Barcode Size List" for details of the barcode size.

printQRcode — Print QR code

Target Mobile printer / POS printer

Syntax

```
(a) - (void) printQRcode:(NSString *)text
                errorCorrection:(ErrorCorrection)errorCorrection
                moduleSize:(ModuleSize)moduleSize
                alignment:(PrintAlignment)alignment;

(b) - (void) printQRcode:(NSString *)text
                errorCorrection:(ErrorCorrection)errorCorrection
                moduleSize:(ModuleSize)moduleSize
                alignment:(PrintAlignment)alignment
                model:(QrModel)model;
```

Parameter

text	Barcode data to send a printer
errorCorrection	Error correction level
moduleSize	Module Size
alignment	Alignment
model	QR code model

Description This method prints QR code. The type (a) is a QR code model 2 fixed.

Also the version for either type (a) or (b) is automatically set depends on the number of data specified on **text** parameter.

See Table 4-26 Error Correction Level (**ErrorCorrection**) for available setting in **errorCorrection** parameter.

See Table 4-23 Module Size (**ModuleSize**) for available setting in **moduleSize** parameter.

See Table 4-21 Alignment (**PrintAlignment**) for available setting in **alignment** parameter.

See Table 4-28 QR Code Model (**QrModel**) for available setting in the **model** parameter.

Error **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

Reference See "Appendix B Barcode Size List" for details of the barcode size.

printDataMatrix — Print Data Matrix

Target: Mobile printer / POS printer

Syntax:
```
- (void) printDataMatrix:(NSString *)text
                  dataMatrixModule:(DataMatrixModule)dataMatrixModule
                  moduleSize:(ModuleSize)moduleSize
                  alignment:(PrintAlignment)alignment;
```

Parameter:

text	Barcode data to send to the printer
dataMatrixModule	The number of the Data Matrix modules
moduleSize	Module Size
alignment	Alignment

Description: This method prints Data Matrix.

See Table 4-29 Data Matrix Module (**DataMatrixModule**) for available setting in **dataMatrixModule** parameter.

See Table 4-23 Module Size (**ModuleSize**) for available setting in **moduleSize** parameter.

See Table 4-21 Alignment (**PrintAlignment**) for available setting in **alignment** parameter.

Error: **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

Reference: See "Appendix B Barcode Size List" for details of the barcode size.

printMaxiCode — Print MaxiCode

Target: Mobile printer / POS printer

Syntax:
```
- (void) printMaxiCode:(NSString *)text
                  maxiCodeMode:(MaxiCodeMode)maxiCodeMode
                  alignment:(PrintAlignment)alignment;
```

Parameter:

text — Barcode data to send to the printer

- When `maxiCodeMode` parameter is **SII_PM_MAXI_CODE_2**:
  Add the service class (3 digits), the country code (3 digits), and the postal code (9 digits) in the beginning of data.
- When `maxiCodeMode` parameter is **SII_PM_MAXI_CODE_3**:
  Add the service class (3 digits), the country code (3 digits), and the postal code (6 digits) in the beginning of data.

maxiCodeMode	MaxiCode mode
alignment	Alignment

Description: This method prints MaxiCode.

See Table 4-30 MaxiCode Mode (**MaxiCodeMode**) for available setting in **maxiCodeMode** parameter.

See Table 4-21 Alignment (**PrintAlignment**) for available setting in **alignment** parameter.

Error **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

Reference See "Appendix B Barcode Size List" for details of the barcode size.

printGS1DataBarStacked — Print GS1 Databar Stacked

Target Mobile printer / POS printer

Syntax
```
- (void) printGS1DataBarStacked:(NSString *)text
                         moduleSize:(ModuleSize)moduleSize
                         alignment:(PrintAlignment)alignment;
```

Description This method is not supported. **SIIPrinterException** is thrown when this method is executed.

printGS1DataBarStackedOmnidirectional — Print GS1 Databar Stacked Omni-directional

Target Mobile printer / POS printer

Syntax
```
- (void) printGS1DataBarStackedOmnidirectional:(NSString *)text
                         moduleHeight:(NSInteger)moduleHeight
                         moduleSize:(ModuleSize)moduleSize
                         alignment:(PrintAlignment)alignment;
```

Description This method is not supported. **SIIPrinterException** is thrown when this method is executed.

printGS1DataBarExpandedStacked — Print GS1 Databar Expanded Stacked

Target Mobile printer / POS printer

Syntax
```
- (void) printGS1DataBarExpandedStacked:(NSString *)text
                         column(NSInteger)column
                         moduleSize(ModuleSize)moduleSize
                         alignment:(PrintAlignment)alignment;
```

Description This method is not supported. **SIIPrinterException** is thrown when this method is executed.

printAztecCode — Print Aztec Code

Target Mobile printer / POS printer

Syntax
```
- (void) printAztecCode:(NSString *)text
                         layer(NSInteger)layer
                         errorCorrection:(NSInteger)errorCorrection
                         moduleSize:(ModuleSize)moduleSize
                         aztecSymbol:(AztecSymbol)aztecSymbol
                         alignment:(PrintAlignment)alignment;
```

Description This method is not supported. **SIIPrinterException** is thrown when this method is executed.

cutPaper — Cut paper

Target: POS printer

Syntax:
```
- (void) cutPaper:(CuttingMethod)cuttingMethod;
```

Parameter: **cuttingMethod** — Cutting method

Description: This method selects enabled/disabled of the paper feed to the cut position and cuts the paper.

See Table 4-31 Cutting Method (**CuttingMethod**) for available setting in **cuttingMethod** parameter.

Error: **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

feedPosition — Paper form feed

Target: Mobile printer / POS printer

Syntax:
```
- (void) feedPosition:(FeedPosition)feedPosition;
```

Description: This method is not supported. **SIIPrinterException** is thrown when this method is executed.

sendBinary — Send binary data

Target: Mobile printer / POS printer

Syntax:
```
- (void) sendBinary:(NSData*)data;
```

Parameter: **data** — Binary data sent to a printer

Description: This method sends the binary data which is specified by **data** parameter to a printer. Data size that can be specified at a time is 3MB (3145728 bytes).

In this method, specified binary data is sent to a printer without conversion.

By sending printer command as binary data with this method, printer functions which are not supported in the library become available. However, this method does not support commands which obtain responses from a printer.

Error: **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

sendDataFile — Send specified file

Target: Mobile printer / POS printer

Syntax:
```
(a) - (void) sendDataFile:(NSString *)fileName
                alignment:(PrintAlignment)alignment
                dithering(Dithering)dithering;
```

```
(b) - (void) sendDataFile:(NSString *)fileName
                  alignment:(PrintAlignment)alignment;

(c) - (void) sendDataFile:(NSString *)fileName;
```

Parameter

fileName	File path of data to sent to the printer
alignment	Alignment
dithering	Dithering

Description

This method determines data format based on the file extension specified by **fileName** parameter, converts it to printer-enabled data format, and sends it to a printer. The maximum file size that can be specified is 16MB (16777216 bytes).

In syntax (a), alignment and dithering can be specified to the sending file.
In syntax (b), alignment of the sending file can be specified.
In syntax (c), the sending file can be specified.

The file extensions capable of sending by **fileName** parameter and sending the file are describes below.

- .bmp, .jpg, .jpeg, or .png:
  Send data to a printer as image file. When that image file is colored one, it is converted to monochrome image by binarization, and sent to a printer.
  Printing start timing of the image depends on the connecting method.
  For Bluetooth connection, printing is executed at one time after mapping the image file in memory of a printer.
  For TCP/IP connection, printing is executed immediately after a printer receives the image file.

- .txt:
  Send data to a printer as text data. Text data format supports UTF-8. Just like **sendText** method, encodes the data to printable text data based on **internationalCharacter** property and **codePage** property, and then sends it to the printer. Also, in this method, the line spacing code is not added to the end of text data.

- .bin or .dat:
  Data is sent to a printer as binary data without conversion.

- .htm or .html:
  Data is sent to a printer as html data without conversion. When **SII_PM_PRINTER_MODEL_DPU_S245** or **SII_PM_PRINTER_MODEL_DPU_S445** is specified by **printerModel** parameter of **connect** method, this method is not supported, so that **SIIPrinterException** is thrown.

alignment parameter is valid only when the file extension specified on **fileName** parameter is .bmp, .jpg, .jpeg, .png or .txt. See Table 4-21 Alignment (**PrintAlignment**) for available setting. When **SII_PM_PRINTER_MODEL_DPU_S245** or **SII_PM_PRINTER_MODEL_DPU_S445** is specified by **printerModel** parameter of **connect** method, the alignment is fixed to left.

dithering parameter is valid only when the file extension specified on **fileName** parameter is .bmp, .jpg, .jpeg, or .png. See Table 4-13 Dithering (**Dithering**) for available setting.

Error

SIIPrinterException is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

sendImage — Send image

Target: Mobile printer / POS printer

Syntax:

```
(a) - (void) sendImage:(UIImage *)image
                    alignment:(PrintAlignment)alignment
                    dithering(Dithering)dithering;

(b) - (void) sendImage:(UIImage *)image
                    alignment:(PrintAlignment)alignment;

(c) - (void) sendImage: (UIImage *)image;
```

Parameter

image	UIImage-type object to be sent to the printer
alignment	Alignment
dithering	Dithering

Description

This method prints the specified UIImage-type image.

In syntax (a), for mobile printer, image alignment is fixed to left alignment, and dithering can be specified.
For POS printer, image alignment and dithering can be specified.
In syntax (b), for mobile printer, image alignment is fixed to left alignment, and dithering is fixed to disable.
For POS printer, image alignment can be specified, and dithering is fixed to disabled.
In syntax (c), image alignment is fixed to left alignment, and dithering is fixed to disabled.

In **image** parameter, data is sent to the printer as image file. Colored image file is converted to monochrome image by binarization and registered. Printing is performed in batch after mapping the image file.

See Table 4-21 Alignment (**PrintAlignment**) for available setting in **alignment** parameter.

See Table 4-13 Dithering (**Dithering**) for available setting in **dithering** parameter.

Error

SIIPrinterException is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

printPDF — Print PDF page

Target: POS printer

Syntax:

```
- (void) printPDF:(NSString *)fileName
                  startIndex:(NSInteger)startIndex
                  endIndex:(NSInteger)endIndex
                  rotate:(Rotate)rotate
                  imageScale:(ImageScale)imageScale
                  bottomMargin:(NSInteger)bottomMargin
                  alignment:(PrintAlignment)alignment
                  dithering:(Dithering)dithering;
```

Parameter

fileName	Path of the PDF file
startIndex	Start number of the printed page
endIndex	End number of the printed page
rotate	Rotation direction of the image

	imageScale	Image scaling
	bottomMargin	Bottom margin (dot)
	dithering	Dithering
	alignment	Alignment

Description

Prints the specified page of the selected PDF file.
PDF is converted to printable format for the printer and is sent to the printer.

The file extension for supporting PDF is .pdf.
A maximum of 16 MB (16777216 bytes) of file size can be specified.
The color image data is converted to monochrome image by binarization.

The valid range of **startIndex** parameter is -1, 1 to the number of pages in the PDF file.
When -1 is specified for **startIndex** parameter, all pages are printed.
When -1 is specified for **startIndex** parameter, the value of **endIndex** parameter is ignored.
When 1 or more is specified for **startIndex** parameter, pages from the page number specified in **startIndex** parameter to the page number specified in **endIndex** parameter are printed.
When a value more than the value specified for **endIndex** parameter, is specified for **startIndex** parameter, an error occurs.
When an out-of-range value is specified for **startIndex** parameter or **endIndex** parameter, an error occurs.

The valid range of **endIndex** parameter is 1 to 2147483647.
When a value more than the number of pages in the PDF file is specified for **endIndex** parameter, pages from the page number specified in **startIndex** parameter to the last page of the PDF file are printed.

See Table 4-32 Image Rotation Direction (**Rotate**) for available setting in **rotate** parameter.

See Table 4-33 Image Scaling (**ImageScale**) available setting in **imageScale** parameter.
When **SII_PM_IMAGE_SCALE_WIDTH_FIT** is specified for **imageScale** parameter, the image width is converted to the print width of the printer while maintaining the aspect ratio.

The valid range of **bottomMargin** parameter is -1, 0 to 2400.
When -1 is specified for **bottomMargin** parameter, the image is created and printed while maintaining the bottom margin.
When a value between 0 and 2400 is specified for **bottomMargin** parameter, the bottom margin is changed to the specified size.
When a value between 0 and 2400 is specified for **bottomMargin** parameter, blank pages are not printed.

See Table 4-13 Dithering (**Dithering**) for available setting in **dithering** parameter.

See Table 4-21 Alignment (**PrintAlignment**) for available setting in **alignment** parameter.

When the memory switches MS 1-3 (Mark Mode Selection) of RP-E10 is set to "Enable", form feed of marked paper or label is performed after printing the PDF page.

Error

SIIPrinterException is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

Note

No guarantee of printing operation when more than 100 pages are printed at a time.

printLogo — Print specified logo (image) on printer

Target: Mobile printer

Syntax:
```
- (void) printLogo:(NSString *)logoId;
```

Parameter:

logoId	Logo ID to print

Description: This method is valid only for Mobile printer. This method print the logo (image) registered by **registerLogo** method. Specify registered logo ID in **logoId** parameter. The valid range of **logoId** parameter is a character string indicating the value from 0 to 127.

Error: **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

printLogo — Print specified logo (image) on printer

Target: POS printer

Syntax:
```
- (void) printLogo:(NSString *)logoId
                 alignment(PrintAlignment)alignment;
```

Parameter:

logoId	Logo ID to print
alignment	Alignment

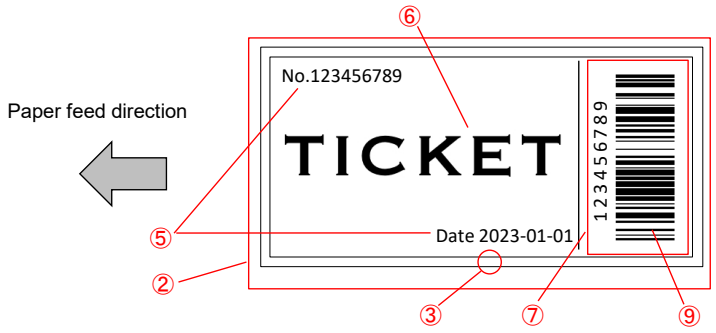
Description: This method is valid only for POS printer. This method print the logo (image) registered by **registerLogo** method. Specify registered logo ID in **logoId** parameter. The valid range of **logoId** parameter is 2 characters. The valid characters are ASCII character code from 20h (blank) to 7Eh (tilde) such as alphanumeric ('0' to '9', 'A' to 'Z', 'a' to 'z').

See Table 4-21 Alignment (**PrintAlignment**) for available setting in **alignment** parameter.

Error: **SIIPrinterException** is thrown when an error occurs while this method is to be called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

(c) Dedicated method for page mode

The following methods are dedicated methods to use page mode. An example for the print process in page mode is shown below.



① Start page mode

```
[printerManager enterPageMode];
```

② Specify print area of page mode

```
[printerManager setPageModeArea:0 y:0 width:355 height:576];
```

③ Specify a rectangle and a ruled line

```
[printerManager printPageModeRectangle:0 startY:0 endX:344 endY:575 lineStyle:SII_PM_LINESTYLE_THIN];
[printerManager printPageModeRectangle:7 startY:7 endX:336 endY:567 lineStyle:SII_PM_LINESTYLE_THIN];
[printerManager printPageModeLine:11 startY:404 endX:334 endY:404 lineStyle:SII_PM_LINESTYLE_THIN];
```

④ Specify print direction of page mode

```
[printerManager setPageModeDirection: SII_PM_DIRECTION_ TOP _TO_ BOTTOM];
```

⑤ Specify a character

```
[printerManager printPageModeText:21 startY:47 text:@"NO.123456789"];
[printerManager printPageModeText:212 startY:340 text:@"Date 2023-01-01"];
```

⑥ Specify an image file

```
[NSString *filePath = [[NSBundle mainBundle] pathForResource:@"TicketImage" ofType:@"jpg"];
[printerManager printPageModeImageFile:10 startY:222 fileName:filePath
dithering:SII_PM_DITHERING_DISABLE];
```

⑦ Specify print area of page mode

```
[printerManager setPageModeArea:0 y:404 width:345 height:163];
```

⑧ Specify print direction

```
[printerManager setPageModeDirection:SII_PM_DIRECTION_LEFT_TO_RIGHT];
```

⑨ Specify a barcode

```
[printerManager printPageModeBarcode:20 startY:132 barcodeSymbol:SII_PM_BARCODE_CODE128
data:[@"{B123456789" dataUsingEncoding:NSASCIIStringEncoding]
moduleSize:SII_PM_BARCODE_MODULE_WIDTH_2 moduleHeight:80
hriPosition:SII_PM_HRI_POSITION_ABOVE hriFont:SII_PM_FONT_A];
```

⑩ Print in page mode

```
[printerManager printPageMode:SII_PM_CUT_PARTIAL];
```

⑪ Ends page mode

```
[printerManager exitPageMode];
```

enterPageMode — Start page mode

Target: POS printer

Syntax:
```
- (void) enterPageMode;
```

Description: This method starts page mode. The dedicated method for page mode and common methods to standard mode and page mode can be used after this method execution.

Executing **exitPageMode** method discards the print data kept in the page data buffer and changes the mode to standard mode.
Executing **printPageMode** method prints the print data kept in the page data buffer.

Error: **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

exitPageMode — End page mode

Target: POS printer

Syntax:
```
- (void) exitPageMode;
```

Description: Discards the print data kept in the page data buffer and changes the mode to standard mode.

Error: **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

setPageModeArea — Specify print area of page mode

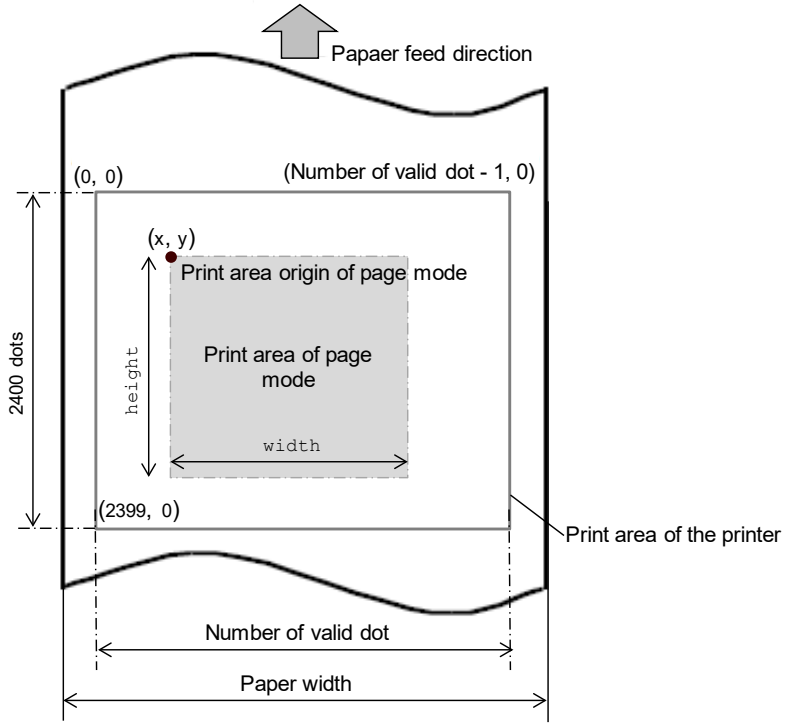
Target: POS printer

Syntax:
```
- (void) setPageModeArea:(NSInteger)x
                        y:(NSInteger)y
                    width:(NSInteger)width
                   height:(NSInteger)height;
```

Parameter:

x	The horizontal origin (dot) of the print area of page mode 0 represents the left edge on the print area of the printer.
y	The vertical origin (dot) of the print area of page mode The valid range is 0 to 2399. 0 represents the position where paper feed has not been performed.
width	The print area width (dot) of page mode
height	The print area height (dot) of page mode The valid range is 1 to (2400−**y**).

The valid range of parameter **x** and **width** is shown in the figure below.



Memory Switch Setting of Printer		Number of Valid Dot	setPageModeArea	
MS4-4 (Paper Width)	MS4-5 (Number of Effective Dots)		**x**	**width**
80 mm	576	576	0 to 575	1 to (576 – **x**)
	512	512	0 to 511	1 to (512 – **x**)
58 mm	432	432	0 to 431	1 to (432 – **x**)
	360	360	0 to 359	1 to (360 – **x**)

The number of valid dots differs depending on the memory switch.
See the technical reference of the target printer for details of memory switch and the setting at shipping.

Description Specifies print area of page mode.

Start page mode by **enterPageMode** method before executing this method.

The print area of page mode can be specified when page mode is started by **enterPageMode** method and then this method is executed after executing the dedicated method for page mode. The data that has been mapped is kept.
The data of the dedicated method for page mode is mapped to the print area of page mode added by this method after executing this method.

The print area of page mode is parameter **x** = 0, **y** = 0, **width** = number of a valid dot, **height** = 2400 after executing **enterPageMode** method.

Error **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

setPageModeDirection — Specify print direction of page mode

Target: POS printer

Syntax:
```
- (void) setPageModeDirection: (Direction)direction;
```

Parameter:

direction	Print direction See Table 4-34 Print Direction (**Direction**) for available setting.

Description: Specifies print direction of page mode.

Start page mode by **enterPageMode** method before executing this method.

The print direction is left to right after executing **enterPageMode** method.

Error: **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

setPageModeLineSpacing — Specify line spacing of page mode

Target: POS printer

Syntax:
```
- (void) setPageModeLineSpacing:(NSInteger)lineSpacing;
```

Parameter:

lineSpacing	Line spacing (dot) of page mode The valid range is 0 to 255.

Description: Specifies line spacing of page mode.

Start page mode by **enterPageMode** method before executing this method.

The line spacing is 34 dots after executing **enterPageMode** method.

Error: **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

printPageMode — Print page mode

Target: POS printer

Syntax:
```
- (void) printPageMode: (CuttingMethod)CuttingMethod;
```

Parameter:

cuttingMethod	Cutting method See Table 4-31 Cutting Method (**CuttingMethod**) for available setting.

Description: Prints the print data kept in page data buffer.
The print data is kept after printing. The print data is discarded at the timing of the followings:

- Execute **enterPageMode** method
- Execute **disconnect** method
- Execute **exitPageMode** method

Error: **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

printPageModeText — Send text data of page mode

Target: POS printer

Syntax:
```
- (void) printPageModeText:(NSInteger)startX
                    startY:(NSInteger)startY
                    text:(NSString *)text;
```

Parameter:

startX	The horizontal reference point (dot) from the starting point The valid range is 0 to 2399.
startY	The vertical reference point (dot) from the starting point The valid range is 0 to 2399.
text	Text data Data size that can be specified at 1 time is 16 KB (16384 bytes).

Description: Maps the text data on the print area of page mode.
This method encodes the specified text data to printable text data based on **internationalCharacter** property and **codePage** property.

Start page mode by **enterPageMode** method before executing this method.

Error: **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

printPageModeTextEx — Send format specified text data of page mode

Target: POS printer

Syntax:
```
- (void) printPageModeTextEx:(NSInteger)startX
                      startY:(NSInteger)startY
                      text:(NSString *)text
                      bold:(CharacterBold)bold
                      underline:(CharacterUnderline)underline
                      reverse:(CharacterReverse)reverse
                      font:(CharacterFont)font
                      scale:(CharacterScale)scale;
```

Parameter:

startX	The horizontal reference point (dot) from the starting point The valid range is 0 to 2399.
startY	The vertical reference point (dot) from the starting point The valid range is 0 to 2399.
text	Text data Data size that can be specified at 1 time is 16 KB (16384 bytes).
bold	Bold character See Table 4-15 Bold Print (**CharacterBold**) for available setting.
underline	Underline See Table 4-16 Underline (**CharacterUnderline**) for available setting.

	reverse	Reverse print See Table 4-17 Reverse Print (**CharacterReverse)** for available setting.
	font	Font See Table 4-19 Character Font (**CharacterFont**) for available setting.
	scale	Character scale See Table 4-20 Character Scale (**CharacterScale**) for available setting.

Description Maps the format specified text data on the print area of page mode.
This method encodes the specified text data to printable text data based on **internationalCharacter** property and **codePage** property.

Start page mode by **enterPageMode** method before executing this method.

Error **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

printPageModeBarcode — Print barcode of page mode

Target POS printer

Syntax (a)
```
- (void) printPageModeBarcode:(NSInteger)startX
                       startY:(NSInteger)startY
                barcodeSymbol:(BarcodeSymbol)barcodeSymbol
                         text:(NSString *)text
                   moduleSize:(ModuleSize)moduleSize
                 moduleHeight:(NSInteger)moduleHeight
                  hriPosition:(HriPosition)hriPosition
                      hriFont:(CharacterFont)hriFont;
```

(b)
```
- (void) printPageModeBarcode:(NSInteger)startX
                       startY:(NSInteger)startY
                barcodeSymbol:(BarcodeSymbol)barcodeSymbol
                         text:(NSString *)text
                   moduleSize:(ModuleSize)moduleSize
                 moduleHeight:(NSInteger)moduleHeight
                  hriPosition:(HriPosition)hriPosition
                      hriFont:(CharacterFont)hriFont
                      nwRatio:(NwRatio)nwRatio;
```

(c)
```
- (void) printPageModeBarcode:(NSInteger)startX
                       startY:(NSInteger)startY
                barcodeSymbol:(BarcodeSymbol)barcodeSymbol
                         data:(NSData*)data
                   moduleSize:(ModuleSize)moduleSize
                 moduleHeight:(NSInteger)moduleHeight
                  hriPosition:(HriPosition)hriPosition
                      hriFont:(CharacterFont)hriFont;
```

(d)
```
- (void) printPageModeBarcode:(NSInteger)startX
                       startY:(NSInteger)startY
                barcodeSymbol:(BarcodeSymbol)barcodeSymbol
                         text:NSString *)text
                   moduleSize:ModuleSize)moduleSize;
```

Parameter

startX The horizontal reference point (dot) from the starting point
The valid range is 0 to 2399.

startY The vertical reference point (dot) from the starting point
The valid range is 0 to 2399.

barcodeSymbol BarcodeSymbol
See Table 4-22 Barcode Symbol (**BarcodeSymbol**) for available setting and correspondent syntax.

text(data) Barcode data to send to the printer
The input conditions for barcode are as follows.

Barcode	Number of Data	Inputtable Data Character String (Data)	Remarks
UPC-A	11 to 12 characters	'0' to '9'	
UPC-E	11 to 12 characters	'0' to '9'	
EAN13 JAN13	12 to 13 characters	'0' to '9'	
EAN8 JAN8	7 to 8 characters	'0' to '9'	
CODE39	1 to 150 characters	'0' to '9' 'A' to 'Z' ' ', '$,', '%', '+', '-', '.', '/'	Start code and stop code ('*') are automatically added.
CODE93	1 to 150 bytes	(0x00 to 0x2E)	Input data with 0x2F or more at the end.
CODE128	2 to 150 bytes	(0x00 to 0x66)	When inputting the start code (0x67 to 0x69) of the CODE128 code set. Input data with 0x67 or more at the end.
		(0x00 to 0x7F)	When starting with a CODE128 special code start code ("{A", "{B", "{C").
ITF	2 to 150 characters (However, an even number)	'0' to '9'	
CODABAR	1 to 150 characters	'0' to '9' '$', '+', '-', '.', '/', ':'	It is needed to specify one of 'A' to 'D' at the beginning and end.
EAN13 add-on JAN13 add-on	Add-on 2: 14 to 15 characters Add-on 5: 17 to 18 characters	'0' to '9'	

moduleSize Barcode width
See Table 4-23 Module Size (**ModuleSize**) for available setting.

moduleHeight Barcode height (dot)
The valid range is 1 to 255.

hriPosition HRI character print position
See Table 4-24 HRI Character Print Position (**HriPosition**) for available setting.

hriFont HRI character font
See Table 4-19 Character Font (**CharacterFont**) for available setting.

nwRatio N:W ratio
See Table 4-25 N:W Ratio (**NwRatio**) for available setting.
Depending on specified parameter **nwRatio** and **moduleSize**, the wide element width is set as shown in the following table.

moduleSize	**nwRatio**		
	SII_PM_ NWRATIO_1TO2	**SII_PM_ NWRATIO_1TO2_5**	**SII_PM_ NWRATIO_1TO3**
SII_PM_BARCODE_MODULE_WIDTH_2	0.500 mm (4 dots)	0.625 mm (5 dots)	0.750 mm (6 dots)
SII_PM_BARCODE_MODULE_WIDTH_3	0.750 mm (6 dots)	1.000 mm (8 dots)	1.125 mm (9 dots)
SII_PM_BARCODE_MODULE_WIDTH_4	1.000 mm (8 dots)	1.250 mm (10 dots)	1.500 mm (12 dots)
SII_PM_BARCODE_MODULE_WIDTH_5	1.250 mm (10 dots)	1.625 mm (13 dots)	1.875 mm (15 dots)
SII_PM_BARCODE_MODULE_WIDTH_6	1.500 mm (12 dots)	1.875 mm (15 dots)	2.250 mm (18 dots)

Description Maps the barcode on the print area of page mode.
In syntax (a), barcode data can be specified as a character string.
In syntax (b), barcode data can be specified as a character string, and the N:W ratio can also be specified.
In syntax (c), barcode data can be specified as a byte array.
The syntax (d) is not supported.

Start page mode by **enterPageMode** method before executing this method.

Error **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

Note Map the print data of the barcode not to overlap the other print data.

Reference See "Appendix B Barcode Size List" for details of the barcode size.

printPageModePDF417 Print PDF417 of page mode

Target POS printer

Syntax (a) - `(void) printPageModePDF417:(NSInteger)startX`
`startY:(NSInteger)startY`
`text:(NSString *)text`
`errorCorrection:(ErrorCorrection)errorCorrection`
`row:(NSInteger)row`
`column:(NSInteger)column`
`moduleSize:(ModuleSize)moduleSize`
`moduleHeight:(NSInteger)moduleHeight`
`pdf417Symbol:(Pdf417Symbol)pdf417Symbol;`

(b) - (void) **printPageModePDF417:**(NSInteger)startX
startY:(NSInteger)startY
text:(NSString *)text
errorCorrection:(ErrorCorrection)errorCorrection
row:(NSInteger)row
column:(NSInteger)column
moduleSize:(ModuleSize)moduleSize
moduleHeight:(NSInteger)moduleHeight;

Parameter

startX	The horizontal reference point (dot) from the starting point The valid range is 0 to 2399.
startY	The vertical reference point (dot) from the starting point The valid range is 0 to 2399.
text	Barcode data
errorCorrection	Error correction level See Table 4-26 Error Correction Level (**ErrorCorrection**) for available setting.
row	The number of rows (row) The valid range is 0, 3 to 90. When 0 is specified, the number of rows is automatically set.
column	The number of columns in data area The valid range is 0 to 30. When 0 is specified, the number of columns in the data area is automatically set.
moduleSize	Nominal fine element width See Table 4-23 Module Size (**ModuleSize**) for available setting.
moduleHeight	Module height (dot) The valid range is 2 to 127. When the module height is set smaller, some barcode scanners may not read it. Set 3 or more for normal use.
pdf417Symbol	Symbol of PDF417 See Table 4-27 PDF417 Symbol (**Pdf417Symbol**) for available setting.

Description

Maps PDF417 on the print area of page mode.
In syntax (a), the PDF417 symbol can be specified as a.
In syntax (b), the PDF417 symbol is fixed to standard PDF417.

Start page mode by **enterPageMode** method before executing this method.

Error

SIIPrinterException is thrown when an error occurs while this method is being called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

Note

Map the print data of the barcode not to overlap the other print data.

Reference

See "Appendix B Barcode Size List" for details of the barcode size.

printPageModeQRcode — Print QR Code of page mode

Target: POS printer

Syntax:

(a) –
```
(void) printPageModeQRcode:(NSInteger)startX
                   startY:(NSInteger)startY
                   text:(NSString *)text
                   errorCorrection:(ErrorCorrection)errorCorrection
                   moduleSize:(ModuleSize)moduleSize
                   model:(QrModel)model;
```

(b) –
```
(void) printPageModeQRcode:(NSInteger)startX
                   startY:(NSInteger)startY
                   text:(NSString *)text
                   errorCorrection:(ErrorCorrection)errorCorrection
                   moduleSize:(ModuleSize)moduleSize;
```

Parameter:

startX	The horizontal reference point (dot) from the starting point The valid range is 0 to 2399.
startY	The vertical reference point (dot) from the starting point The valid range is 0 to 2399.
text	Barcode data The version for either syntax (a) or (b) is automatically set depending on the number of data specified on parameter **text**.
errorCorrection	Error correction level See Table 4-26 Error Correction Level (**ErrorCorrection**) for available setting.
moduleSize	Module size See Table 4-23 Module Size (**ModuleSize**) for available setting.
model	QR Code Model See Table 4-28 QR Code Model (**QrModel**) for available setting.

Description: Maps QR Code on the print area of page mode.
In syntax (a), QR Code Model can be specified.
In syntax (b), QR Code Model is fixed to QR Code Model 2.

Start page mode by **enterPageMode** method before executing this method.

Error: **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

Note: Map the print data of the barcode not to overlap the other print data.

Reference: See "Appendix B Barcode Size List" for details of the barcode size.

printPageModeDataMatrix — Print Data Matrix of page mode

Target: POS printer

Syntax:
```
- (void) printPageModeDataMatrix:(NSInteger)startX
                          startY:(NSInteger)startY
                          text:(NSString *)text
                          dataMatrixModule:(DataMatrixModule)dataMatrixModule
                          moduleSize:(ModuleSize)moduleSize;
```

Parameter:

startX	The horizontal reference point (dot) from the starting point The valid range is 0 to 2399.
startY	The vertical reference point (dot) from the starting point The valid range is 0 to 2399.
text	Barcode data
dataMatrixModule	The number of Data Matrix modules See Table 4-29 Data Matrix Module (**DataMatrixModule**) available setting.
moduleSize	Module size See Table 4-23 Module Size (**ModuleSize**) for available setting.

Description: Maps Data Matrix on the print area of page mode.

Start page mode by **enterPageMode** method before executing this method.

Error: **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

Note: Map the print data of the barcode not to overlap the other print data.

Reference: See "Appendix B Barcode Size List" for details of the barcode size.

printPageModeMaxiCode — Print MaxiCode of page mode

Target: POS printer

Syntax:
```
- (void) printPageModeMaxiCode:(NSInteger)startX
                          startY:(NSInteger)startY
                          text:(NSString *)text
                          maxiCodeMode:(MaxiCodeMode)maxiCodeMode;
```

Parameter:

startX	The horizontal reference point (dot) from the starting point The valid range is 0 to 2399.
startY	The vertical reference point (dot) from the starting point The valid range is 0 to 2399.
text	Barcode data • When parameter **maxiCodeMode** is **SII_PM_MAXI_CODE_2** Add the service class (3 digits), the country code (3 digits), and the postal code (9 digits) to the beginning of the data.

- When parameter **maxiCodeMode** is **SII_PM_MAXI_CODE_3**
  Add the service class (3 digits), the country code (3 digits), and the postal code (6 digits) to the beginning of the data.

maxiCodeMode	MaxiCode Mode See Table 4-30 MaxiCode Mode (**MaxiCodeMode**) available setting.

Description Maps MaxiCode on the print area of page mode.

Start page mode by **enterPageMode** method before executing this method.

Error **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

Note Map the print data of the barcode not to overlap the other print data.

Reference See "Appendix B Barcode Size List" for details of the barcode size.

printPageModeGS1DataBarStacked Print GS1 Databar Stacked of page mode

Target POS printer

Syntax
```
- (void) printPageModeGS1DataBarStacked:(NSInteger)startX
                                 startY:(NSInteger)startY
                                 text:(NSString *)text
                                 moduleSize:(ModuleSize)moduleSize;
```

Description This method is not supported. **SIIPrinterException** is thrown when this method is executed.

printPageModeGS1DataBarStackedOmnidirectional Print GS1 Databar Stacked Omni-directional of page mode

Target POS printer

Syntax
```
- (void) printPageModeGS1DataBarStackedOmnidirectional:
                                 (NSInteger)startX
                                 startY:(NSInteger)startY
                                 text:(NSString *)text
                                 moduleHeight:(NSInteger)moduleHeight
                                 moduleSize:(ModuleSize)moduleSize;
```

Description This method is not supported. **SIIPrinterException** is thrown when this method is executed.

printPageModeGS1DataBarExpandedStacked — Print GS1 Databar Expanded Stacked of page mode

Target POS printer

Syntax
```
- (void) printPageModeGS1DataBarExpandedStacked:(NSInteger)startX
                                        startY:(NSInteger)startY
                                        text:(NSString *)text
                                        column:(NSInteger)column
                                        moduleSize:(ModuleSize)moduleSize;
```

Description This method is not supported. **SIIPrinterException** is thrown when this method is executed.

printPageModeAztecCode — Print Aztec Code of page mode

Target POS printer

Syntax
```
- (void) printPageModeAztecCode:(NSInteger)startX
                                startY:(NSInteger)startY
                                text:(NSString *)text
                                layer:(NSInteger)layer
                                errorCorrection:(NSInteger)errorCorrection
                                moduleSize:(ModuleSize)moduleSize
                                aztecSymbol:(AztecSymbol)aztecSymbol;
```

Description This method is not supported. **SIIPrinterException** is thrown when this method is executed.

sendPageModeBinary — Send binary data of page mode

Target POS printer

Syntax
```
- (void) sendPageModeBinary:(NSData*)data;
```

Parameter

binary	Binary data Data size that can be specified at 1 time is 3 MB (3145728 bytes).

Description Maps binary data on the print area of page mode.

Start page mode by **enterPageMode** method before executing this method.

This method sends the specified binary data to the printer without conversion.

By sending printer commands as binary data with this method, printer functions which are not supported in the library become available.

Error **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

Note This method may execute unexpected performance depending on the data to send. Please ensure the performance with your actual device in advance.

printPageModeImageFile — Draw Image file of page mode

Target: POS printer

Syntax:
```
- (void) printPageModeImageFile:(NSInteger)startX
                         startY:(NSInteger)startY
                       fileName:(NSString *)fileName
                      dithering:(Dithering)dithering;
```

Parameter:

startX	The horizontal reference point (dot) from the starting point The valid range is 0 to 2399.
startY	The vertical reference point (dot) from the starting point The valid range is 0 to 2399.
fileName	File path of the data The maximum file size that can be specified is 16 MB (16777216 bytes). The image files that can be sent are .bmp, .jpg, .jpeg, .png. Colored image file is converted to monochrome image by binarization and registered.
dithering	Dithering See Table 4-13 Dithering (**Dithering**) for available setting.

Description: Maps the image file on the print area of page mode.

Start page mode by **enterPageMode** method before executing this method.

Error: **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

printPageModeRectangle — Draw rectangle image of page mode

Target: POS printer

Syntax:
```
- (void) printPageModeRectangle:(NSInteger)startX
                         startY:(NSInteger)startY
                           endX:(NSInteger)endX
                           endY:(NSInteger)endY
                      lineStyle:(LineStyle)lineStyle;
```

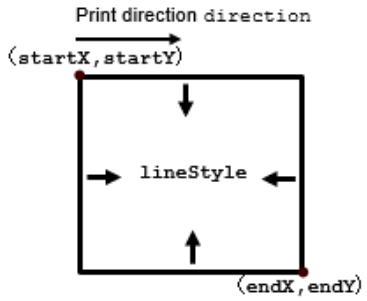
Parameter:

startX	The horizontal reference point (dot) from the starting point The valid range is 0 to 2399.
startY	The vertical reference point (dot) from the starting point The valid range is 0 to 2399.
endX	The horizontal reference point (dot) from the ending point The valid range is 0 to 2399.
endY	The vertical reference point (dot) from the ending point The valid range is 0 to 2399.
lineStyle	Line style See Table 4-35 Line Style (**LineStyle**) for available setting.

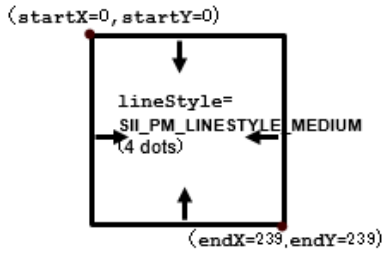
Description Maps the rectangle image on the print area of page mode.

Start page mode by **enterPageMode** method before executing this method.

The rectangle is mapped to the parameter **direction** of **setPageModeDirection** method as shown in the figure below.



The example of the parameter setting to the image is shown below.
Example: Draw a square with a medium solid line (4 dots) at 240 dots (30 mm) from the starting point.

Image	Parameter	
	**startX**	0
	startY	0
	endX	239
	endY	239
	lineStyle	**SII_PM_LINESTYLE_MEDIUM**

Error **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

printPageModeLine — Print ruled line of page mode

Target POS printer

Syntax
```
- (void) printPageModeLine:(NSInteger)startX
                    startY:(NSInteger)startY
                      endX:(NSInteger)endX
                      endY:(NSInteger)endY
                 lineStyle:(LineStyle)lineStyle;
```

Parameter

startX The horizontal reference point (dot) from the starting point
The valid range is 0 to 2399.

startY The vertical reference point (dot) from the starting point
The valid range is 0 to 2399.

endX The horizontal reference point (dot) from the ending point
The valid range is 0 to 2399.

endY The vertical reference point (dot) from the ending point
The valid range is 0 to 2399.

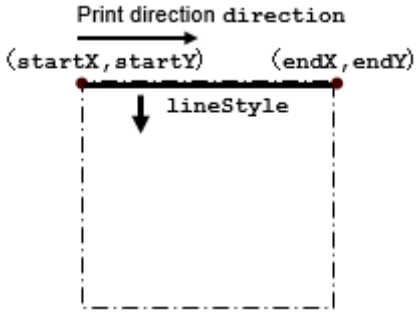
	lineStyle	Line style See Table 4-35 Line Style (**LineStyle**) for available setting.

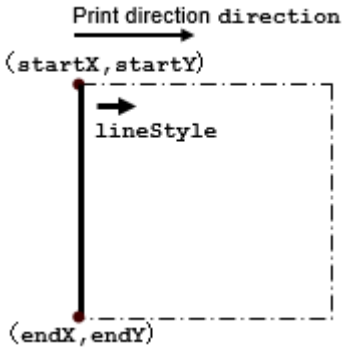
Description Maps the ruled line on the print area of page mode.

Start page mode by **enterPageMode** method before executing this method.

A diagonal stroke cannot be drawn by this method.

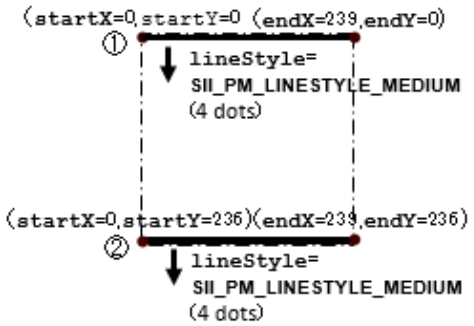
The ruled line is mapped to the parameter **direction** of **setPageModeDirection** method as shown in the figure below.



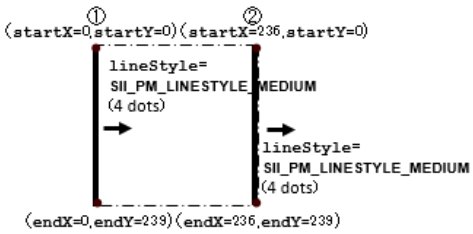


Mapping direction of horizontal ruled line **Mapping direction of vertical ruled line**

The setting example of the parameter to the image is shown below.
Example: Draw a horizontal ruled line of a square with a medium solid line (4 dots) at 240
dots (30 mm) from the starting point.

Image	Parameter
	① **startX** 0 **startY** 0 **endX** 239 **endY** 0 **lineStyle** **SII_PM_LINESTYLE_MEDIUM** ② **startX** 0 **startY** 236 **endX** 239 **endY** 236 **lineStyle** **SII_PM_LINESTYLE_MEDIUM**

Example: Draw a vertical ruled line of a square with a medium solid line (4 dots) at 240 dots (30 mm) from the starting point.

Image	Parameter
	① **startX** 0 **startY** 0 **endX** 0 **endY** 239 **lineStyle** **SII_PM_LINESTYLE_MEDIUM** ② **startX** 236 **startY** 0 **endX** 236 **endY** 239 **lineStyle** **SII_PM_LINESTYLE_MEDIUM**

Error **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

printPageModeLogo — Print logo of page mode

Target POS printer

Syntax
```
- (void) printPageModeLogo:(NSInteger)startX
                    startY:(NSInteger)startY
                    logoId:(NSString *)logoId;
```

Parameter

startX	The horizontal reference point (dot) from the starting point The valid range is 0 to 2399.
startY	The vertical reference point (dot) from the starting point The valid range is 0 to 2399.
logoId	Logo ID to print (key code) Specify the ID of the registered logo as a character string

Description Maps the registered logo on the print area of page mode.

Start page mode by **enterPageMode** method before executing this method.

Error **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.4 SIIPrinterException Class**" for details on the error.

(6) Common property detail to standard mode and page mode

endTimeout — Timeout period when sending data

Target: Mobile printer / POS printer

Syntax:
```
@property NSInteger sendTimeout;
```

Description: This property maintains the timeout period when data is sent. This property is configurable or obtainable whether a printer is connected or not. However, the configured timeout period is enabled at the next data sending. When the configured value is outside of the valid range, 100 is set when the set value is lower than 100, or 300000 is set when the set value is more than 300000.

Default	10000 msec (10 seconds)
Effective range	100 msec to 300000 msec (5 minutes)

receiveTimeout — Timeout period when receiving data

Target: Mobile printer / POS printer

Syntax:
```
@property NSInteger receiveTimeout;
```

Description: This property maintains the timeout period when data is received. This property is configurable or obtainable whether a printer is connected or not. When the configured value is outside of the valid range, 100 is set when the set value is lower than 100, or 300000 is set when the set value is more than 300000.

Default	10000 msec (10 seconds)
Effective range	100 msec to 300000 msec (5 minutes)

internationalCharacter — Set international character set

Target: Mobile printer / POS printer

Syntax:
```
@property NSInteger internationalCharacter;
```

Description: This property maintains the configuration of international character set. When text data is sent by **sendText** method, **sendTextEx** method, **sendDataFile** method, or **printPageModeText** method, or **printPageModeTextEx** method, the print result of the following character codes varies.

Character codes with the varying print result depending on the configuration of the international character:
0x23, 0x24, 0x40, 0x5B, 0x5C, 0x5D, 0x5E, 0x60, 0x7B, 0x7C, 0x7D, 0x7E

See Table 4-8 International Character Setting Constant for details about configurable values. When international character set is not configured, it is initialized to following state depending on a language setting of an iOS device. Also, when invalid value is specified, it is ignored.

When a language setting of an iOS device is Japanese:
SII_PM_COUNTRY_JAPAN

When a language setting of an iOS device is other languages than Japanese:
SII_PM_COUNTRY_USA

codePage — Code page

Target	Mobile printer / POS printer
Syntax	`@property NSInteger` **`codePage;`**
Description	This property maintains setting values of codepage. The encoder used for sending the text data by **`sendText`** method, **`sendTextEx`** method, **`sendDataFile`** method, p**`rintPageModeText`** method, or **`printPageModeTextEx`** method is changed. See Table 4-9 Codepage constant for the details about configurable values. When a codepage is not configured, it is initialized to the following state depending on a language setting of an iOS device. Also, when invalid value is specified, it will be ignored. When a language setting of an iOS device is Japanese: **SII_PM_CODE_PAGE_KATAKANA** When a language setting of an iOS device is other languages than Japanese: **SII_PM_CODE_PAGE_1252**

printerModel — Obtain printer model

Target	Mobile printer / POS printer
Syntax	`@property(readonly) NSInteger` **`printerModel;`**
Description	This property obtains a model value for the connected printer. When a printer is not connected, -1 is returned.
Return value	See Table 4-5 Printer Model Constant for details.
Default	-1

portType — Connecting port type

Target	Mobile printer / POS printer
Syntax	`@property(readonly) NSInteger` **`portType;`**
Description	This property obtains the port type of connecting printer in use during connection with a printer. When a printer is not connected, -1 is returned.
Return value	See Table 4-6 Port Type Constant for details.
Default	-1

isConnect — Verify connection state with a printer

Target	Mobile printer / POS printer
Syntax	`@property(readonly) BOOL` **`isConnect;`**
Description	This property maintains BOOL value with a printer. When the **`connect`** method succeeds, this property is YES. After **`connect`** method, when **`disconnect`** method succeeds, this property becomes NO.
Return value	Following values are returned depending on the connection state with a printer.

YES	Connected to a printer
NO	Disconnected to a printer

socketKeepingTime Socket keeping time

Target POS printer

Syntax `@property NSInteger` **`socketKeepingTime;`**

Description This method is valid only for POS printer. This property maintains the setting of socket keeping time during TCP/IP connection. For socket keeping time, set the same time as Network Printer Receive Timeout. The setting of Network Printer Receive Timeout can be changed in "SII Printer Utility" with the iOS app on the App Store.
This property is configurable and obtainable whether a printer is connected or not. However, socket keeping time is enabled at the next execution of **`connect`** method (TCP/IP). Moreover, when configured value is outside of the valid range, set 6000 when the set value is lower than 6000 and set 300000 when the set value is more than 300000.

Default	300000msec (5 minutes)
Effective range	60000 to 300000msec (5 minutes)

delegate Register delegate

Target Mobile printer / POS printer

Syntax `@property(weak, nonatmic) id<SIIPrinterManagerDelegate>` **`delegate;`**

Description Registers a delegate object that receives notifications from the printer.

Specify an object conforming to **`SIIPrinterManagerDelegate`** protocol.
When this property is executed with the delegate object registered, the already registered delegate object becomes disabled, and a new delegate object is registered.

When specifying nil for this property, the notification of the printer status is stopped.

4.3.2 `SIIPrinterInfo` Class

This class stores printer information searched by printer search method. This class obtains printer model, MAC address, and IP address from the searched printer information.

(1) Method List

This class stores the printer information searched by printer searching method. This class obtains printer model, MAC address, and IP address from the searched printer information.

The list of methods provided by **`SIIPrinterInfo`** class is shown in the following table.

Table 4-44 Method of `SIIPrinterInfo` Class

Method	Function Summary	Target	
		Mobile	POS
`SIIPrinterInfo`	Constructor for printer information class	Not supported	Supported

(2) Property List

The list of methods provided by the **`SIIPrinterInfo`** class is shown in the following table.

Table 4-45 Property of `SIIPrinterInfo` Class

Method	Access	Function Summary	Target	
			Mobile	POS
`name`	R	Obtain printer model name	Not supported	Supported
`mac`	R	Obtain MAC address	Not supported	Supported
`ip`	R	Obtain IP address	Not supported	Supported

(3) Method Details

`IIPrinterInfo` Constructor

Target POS printer

Syntax **`SIIPrinterInfo`**

Description This method is valid only for POS printer. This method is used for storing printer information searched by printer searching method.

(4) Property Details

name — Obtain printer model name

Target	POS printer
Syntax	`@property NSString *name;`
Description	This property is valid only for POS printer. This property obtains printer model name searched by printer searching method.

mac — Obtain MAC address

Target	POS printer
Syntax	`@property NSString *mac;`
Description	This property is valid only for POS printer. This property obtains MAC address searched by printer searching method.

ip — Obtain IP address

Target	POS printer
Syntax	`@property NSString *ip;`
Description	This property is valid only for POS printer. This property obtains IP address searched by printer searching method.

4.3.3 `SIIPrinterIdentifier` Class

This class stores the printer ID of the printer retrieved by **`getPrinterResponse`** method. This class is valid only for POS printer.

(1) Method List

The list of methods provided by **`PrinterIdentifier`** class is shown in the following table.

Table 4-46 Method of `PrinterIdentifier` Class

Method	Function Summary	Target	
		Mobile	POS
`SIIPrinterIdentifier`	Constructor for **`PrinterIdentifier`** class	Not supported	Supported

(2) Property List

The list of properties provided by **`PrinterIdentifier`** class are shown in the following table.

Table 4-47 Property of `PrinterIdentifier` Class

Method	Access	Function Summary	Target	
			Mobile	POS
`printerModelID`	R	Get printer model ID	Not supported	Supported
`typeID`	R	Get type ID	Not supported	Supported
`romVersionID`	R	Get ROM version ID	Not supported	Supported
`mainFirmwareVersion`	R	Get firmware version (main)	Not supported	Supported
`manufacturer`	R	Get manufacturer	Not supported	Supported
`modelName`	R	Get model name	Not supported	Supported
`serialNumber`	R	Get serial number	Not supported	Supported
`bootFirmwareVersion`	R	Get firmware version (boot)	Not supported	Supported
`bootFirmwareChecksum`	R	Get firmware checksum (boot)	Not supported	Supported
`mainFirmwareChecksum`	R	Get firmware checksum (main)	Not supported	Supported
`firmwareChecksum`	R	Get firmware checksum (main + boot)	Not supported	Supported

(3) Method Details

`SIIPrinterIdentifier` Constructor

Target POS printer

Syntax **`SIIPrinterIdentifier`**

Description This method stores the printer ID of the printer retrieved by **`getPrinterResponse`** method.

(4) Property Details

printerModelID — Get printer model ID

Target POS printer

Syntax `@property UInt8 **printerModelID;**`

Description This property gets the printer model ID from the printer ID of the printer retrieved by **`getPrinterResponse`** method.

typeID — Get type ID

Target POS printer

Syntax `@property UInt8 **typeID;**`

Description This property gets the type ID from the printer ID of the printer retrieved by **`getPrinterResponse`** method.

romVersionID — Get ROM version ID

Target POS printer

Syntax `@property UInt8 **romVersionID;**`

Description This property gets the ROM version ID from the printer ID of the printer retrieved by **`getPrinterResponse`** method.

mainFirmwareVersion — Get firmware version (main)

Target POS printer

Syntax `@property NSString *__mainFirmwareVersion;__`

Description This property gets the firmware version (main) as a string from the printer ID of the printer retrieved by **`getPrinterResponse`** method.

manufacturer — Get manufacturer

Target POS printer

Syntax `@property NSString *__manufacturer;__`

Description This property gets the manufacturer name as a string from the printer ID of the printer retrieved by **`getPrinterResponse`** method.

modelName — Get model name

Target POS printer

Syntax `@property NSString *modelName;`

Description This property gets the model name as a string from the printer ID of the printer retrieved by **getPrinterResponse** method.

serialNumber — Get serial number

Target POS printer

Syntax `@property NSString *serialNumber;`

Description This property gets the serial number as a string from the printer ID of the printer retrieved by **getPrinterResponse** method.

bootFirmwareVersion — Get firmware version (boot)

Target POS printer

Syntax `@property NSString *bootFirmwareVersion;`

Description This property gets the firmware version (boot) as a string from the printer ID of the printer retrieved by **getPrinterResponse** method.

bootFirmwareChecksum — Get firmware checksum (boot)

Target POS printer

Syntax `@property NSData *bootFirmwareChecksum;`

Description This property gets the firmware checksum (boot) from the printer ID of the printer retrieved by **getPrinterResponse** method.

mainFirmwareChecksum — Get firmware checksum (main)

Target POS printer

Syntax `@property NSData *mainFirmwareChecksum;`

Description This property gets the firmware checksum (main) from the printer ID of the printer retrieved by **getPrinterResponse** method.

firmwareChecksum — Get firmware checksum (main + boot)

Target POS printer

Syntax `@property NSData *firmwareChecksum;`

Description This property gets the firmware checksum (main + boot) from the printer ID of the printer retrieved by **getPrinterResponse** method.

4.3.4 `SIIPrinterException` Class

(1) Method List

The list of methods provided by **`SIIPrinterException`** class is shown in the following table.

Table 4-48 Method of `SIIPrinterException` Class

Method	Function Summary	Target	
		Mobile	POS
`SIIPrinterException`	Exception class thrown when API for **`SIIPrinterManager`** class is called.	Supported	Supported

(2) Property List

The list of property provided by the **`SIIPrinterException`** class is shown in the following table.

Table 4-49 Property of `SIIPrinterException` Class

Property	Access	Function Summary	Target	
			Mobile	POS
`errorCode`	R	Obtain error code	Supported	Supported
`errorMessage`	R	Obtain error message	Supported	Supported

(3) Constant List

Constants used for obtaining error codes are shown in the following table.

Table 4-50 Error Codes List

Constant Name	Description	Value	Target	
			Mobile	POS
SII_PM_ERROR_ ACCESS_DENIED	Failed to get the handle.[*1]	-1	Supported	Supported
	An unavailable port was specified.		Supported	Supported
SII_PM_ERROR_ SHARING_VIOLATION	An already opened port was specified.	-11	Supported	Supported
SII_PM_ERROR_ PORT_NOT_OPENED	The port is not open.	-12	Supported	Supported
SII_PM_ERROR_DEVICE_ NOT_CONNECTED	A printer with the specified Bluetooth device name does not exist.	-21	Supported	Supported
	A printer with the specified IP address does not exist.		Not supported	Supported
SII_PM_ERROR_OFFLINE	Disconnected state or the printer is offline.	-22	Supported	Supported
SII_PM_ERROR_DEVICE_ INITIALIZE_FAILED	Failed to change the printer settings. Data sending to printer is not completed within the send timeout period, or data receiving from the printer is not completed within the receive timeout period.	-31	Supported	Supported

Constant Name	Description	Value	Target	
			Mobile	POS
SII_PM_ERROR_ DATA_SIZE_ZERO	0-byte data was specified.	-101	Supported	Supported
SII_PM_ERROR_ OVER_MAX_DATA_SIZE	Maximum data size is exceeded.	-102	Supported	Supported
SII_PM_ERROR_ ENCODE_FAILED	An Error occurred in encoding text data.*1	-111	Supported	Supported
SII_PM_ERROR_ TIMEOUT	Send timeout occurred.	-201	Supported	Supported
	Receive timeout occurred.		Supported	Supported
SII_PM_ERROR_ FILE_NOT_FOUND	The specified file is not found.	-301	Supported	Supported
SII_PM_ERROR_ LOW_MEMORY	Memory shortage occurred when loading image file file.	-311	Supported	Supported
SII_PM_ERROR_ OVER_MAX_IMAGE	Either or both of width and height of image file exceeds the number of printable maximum dots.	-312	Supported	Supported
SII_PM_ERROR_ LOGO_NOT_DEFINED	The logo is not registered.	-313	Not supported	Supported
SII_PM_ERROR_ LOW_USER_AREA	Remaining user area is insufficient.	-401	Supported	Supported
SII_PM_ERROR_ LOW_EXTERNAL_RAM	Remaining RAM capacity is insufficient.	-402	Supported	Not supported
SII_PM_ERROR_ INVALID_NO	The specified value for the logo ID or style sheet number is invalid.	-501	Supported	Supported
SII_PM_ERROR_ OVER_STYLE_NUM	The number of style registered in the specified file exceeds rated value (64).	-502	Not supported	Supported
SII_PM_ERROR_ PAGE_MODE_SPECIFIED	Page mode is specified.	-511	Not supported	Supported
SII_PM_ERROR_ PAGE_MODE_NOT_SPECIFIED	Page mode is not specified.	-512	Not supported	Supported
SII_PM_ERROR_ INVALID_PARAM	The specified parameter is invalid.	-9999	Supported	Supported

*1; Abnormality processing might have happened.

(4) Method Details

SIIPrinterException Constructor

Target	Mobile printer / POS printer
Syntax	**SIIPrinterException**
Description	Exception class thrown when API for **SIIPrinterManager** class is called.

(5) Property Details

errorCode Obtain error codes

Target	Mobile printer / POS printer
Syntax	`@property NSInteger` **errorCode;**
Description	This method obtains exception error code thrown.
Return value	See Table 4-50 Error Codes List for details.

errorMessage Obtain error message

Target	Mobile printer / POS printer
Syntax	`@property NSString *`**errorMessage;**
Description	This method obtains exception error message thrown. Character string to complement **errorCode** property can be obtained.

4.3.5 `SIIPrinterManagerDelegate` Protocol

(1) Method List

The list of methods provided by **`SIIPrinterManagerDelegate`** protocol is shown in the following table.

Table 4-51 Method of `SIIPrinterManagerDelegate` Protocol

Method	Function Summary	Target	
		Mobile	POS
`didStatusChange`	Notify printer status.	Supported	Supported

(2) Method Details

`didStatusChange` Notify printer status

Target Mobile printer / POS printer

Syntax
```
- (void) didStatusChange:(SIIPrinterManager *)printerManager
                   status:(NSInteger)status;
```

Parameter

`printerManager` Calling `SIIPrinterManager` object

`status` Printer status

Description This method is called the latest status at the following timing.
・When **`connect`** method is executed.
・When the printer status is changed.

This method is called when **`isConnect`** property is YES.

The notification of the printer status is stopped by **`disconnect`** method.
The notification of the printer status is stopped by setting nil to **`delegate`** property.

Mobile printer:
After disconnection from the printer, this method notifies 0x00.

POS printer:
After disconnection from the printer, this method notifies 0x80000000.

The library attempts to resume communication with the printer until **`disconnect`** method is executed.
When communication with the printer becomes possible, this method notifies the latest printer status.
See **`getStatus`** method for description of the printer status.

Do not execute the APIs of **`SIIPrinterManager`** class within this method.

4.3.6 `SIISmartLabelManager` Class

`SIISmartLabelManager` class provides the function to covert the label file (*.sl, slex) created using SII Layout Editor into the printable data from the printer.

Do not use this class because it is not supported.

Chapter 5
Sample Program

This chapter describes the sample program provided by the SDK.

5.1 Screen Layout

The SDK includes SiiLibSample as the sample program with Xcode project format.
This section describes the screen of SiiLibSample.

5.1.1 Main Screen

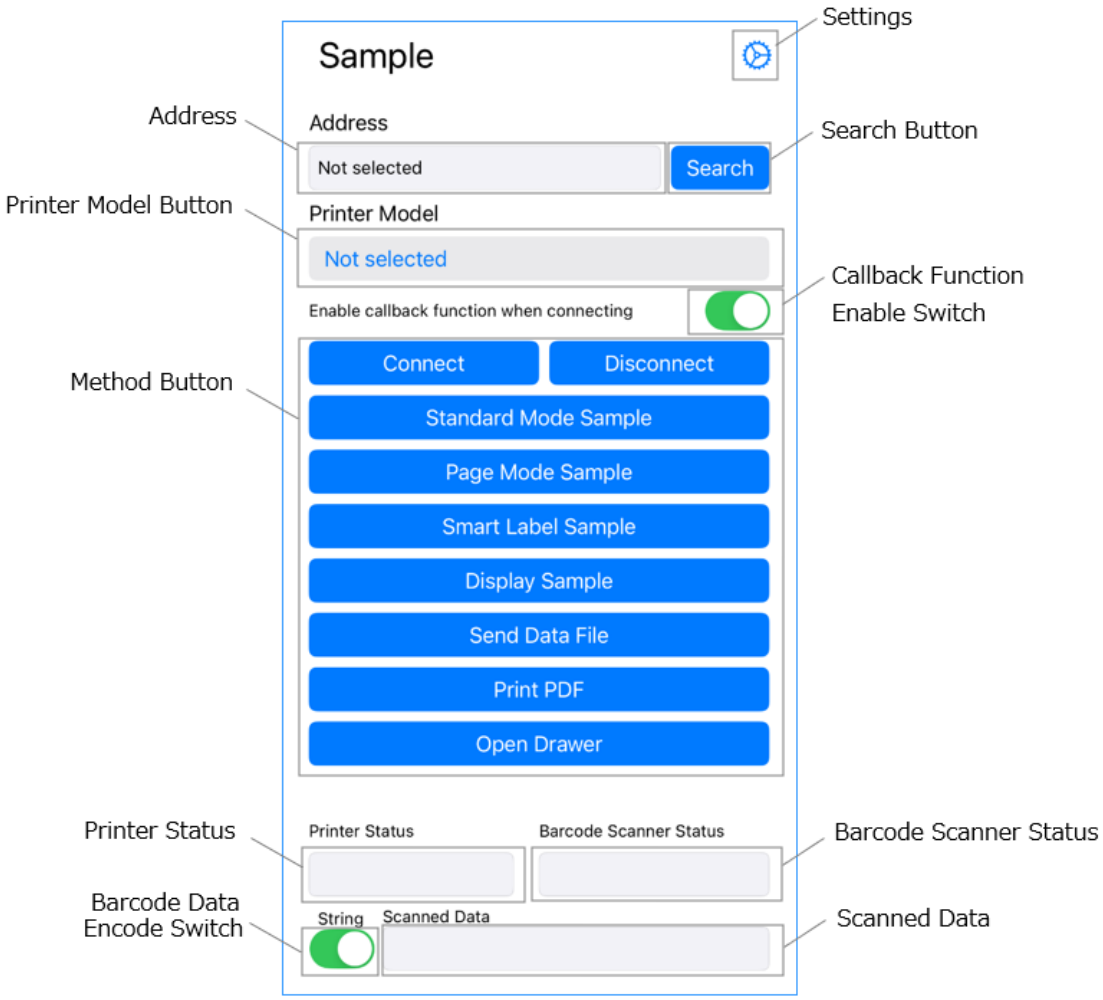


Figure 5-1 SiiLibSample

Table 5-1 Functions of Sample Program

Item	Description
Settings	Tapping the [Settings] button opens the function setting screen. In order to go back to the main screen, tap [close] on the top left of the screen.
Address	Displays the information about the selected printer.
Printer Model Button	Specifies the printer model. When tapping [Printer Model Button], a list of printer models is displayed. By selecting from the list, the printer model can be entered. When the printer is selected from the printer search screen, the printer model is automatically displayed.
Search Button	Starts searching for printers. Transits to the printer search screen. A list of the searched printers is displayed. The printer is selected by tapping the searched printer and returns to the main screen.
Callback Function Enable Switch	Select whether to enable the callback function when connecting to the printer. On : Starts the callback function when connecting. Off : The callback function does not respond.
Method Button*1	In addition to the method buttons for executing **`connect`** and **`disconnect`**, the sample by the combination of some methods can be printed and checked for the operation of peripheral devices.
Printer Status	Displays the printer status. When [Callback Function Enable Switch] is On, the latest status is displayed.
Barcode Scanner Status	Displays the connection status of the barcode scanner. Mobile printer and POS printer do not support the barcode scanner.
Barcode Data Encode Switch	Selects the conversion of barcode data read by the barcode scanner. Mobile printer and POS printer do not support the barcode scanner.
Scanned Data	Displays the barcode data read by the barcode scanner. Mobile printer and POS printer do not support the barcode scanner.

*1: Supported functions vary by model. Only supported functions can be operated.

5.1.2 [Settings] Screen

Various setting functions are displayed in [Settings].

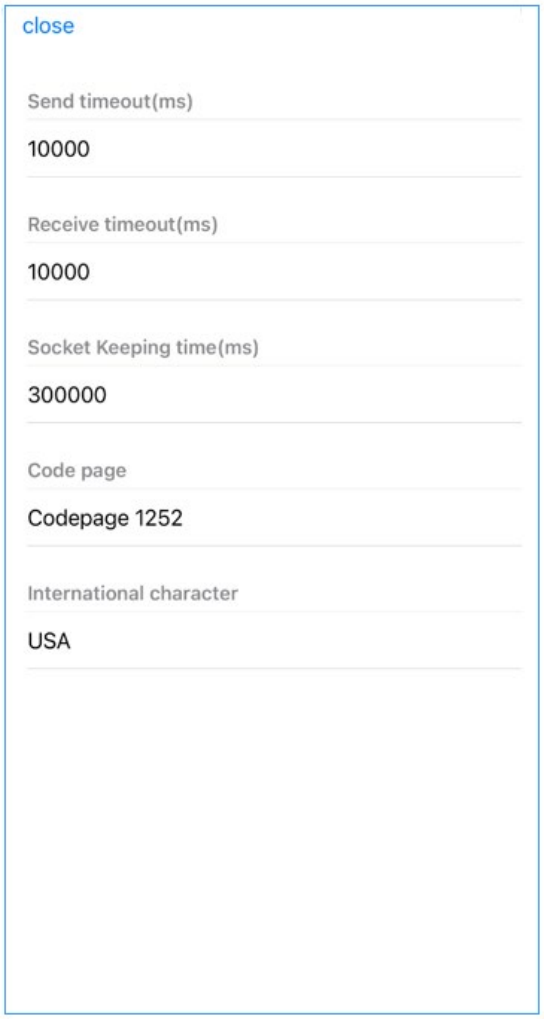


Figure 5-2 [Settings] Screen

5.2 Precaution

The sample program is subject to change without notice.

No guarantee of proper operation and support are provided for the sample program.

Chapter 6
Disclaimer

We closely monitor the development of this software in order to avoid problems. However, we are not responsible for any damages arising out of the use of this software.

Appendix A
Character Set

A.1 Codepage Table (Character Code Table)

The codepages when **SII_PM_COUNTRY_USA** is set for the international character set are shown below.
Available code pages vary depending on the printer.
Print results of the specific character codes or display results vary depending on the setting of the international character set.
See "A.2 International Character Set" for the specific character codes.

- **Mobile printer**

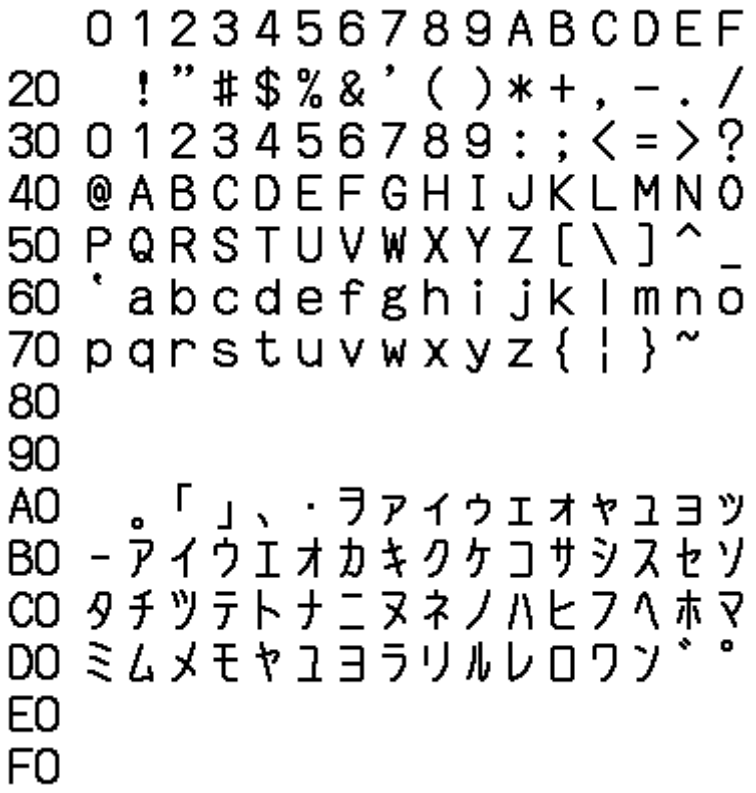


Figure A-1 SII_PM_CODE_PAGE_KATAKANA



Figure A-2 SII_PM_CODE_PAGE_1252 (Latin)

- **POS printer**

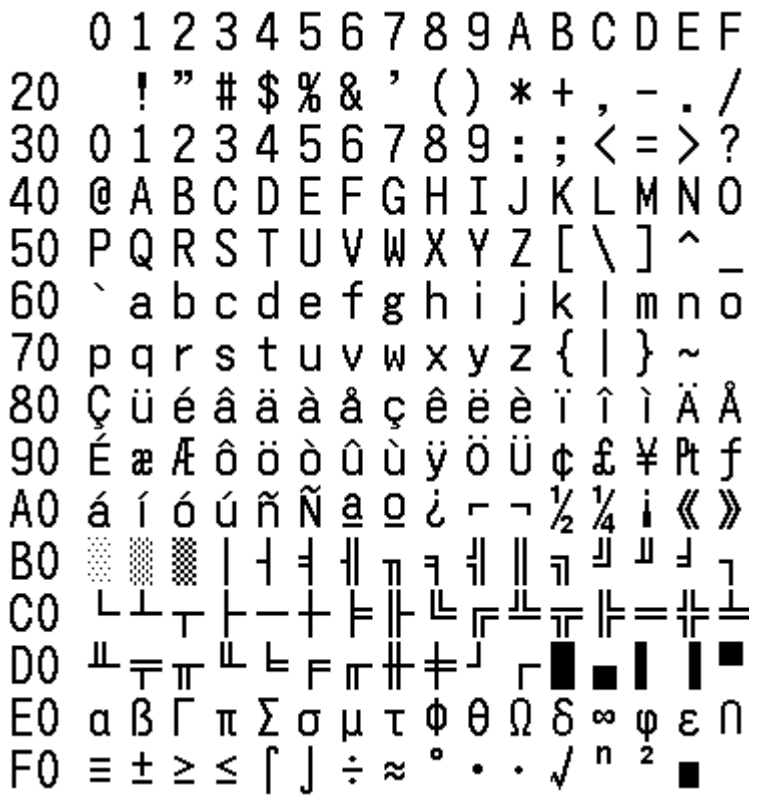


Figure A-3 SII_PM_CODE_PAGE_437 (USA, Standard Europe)

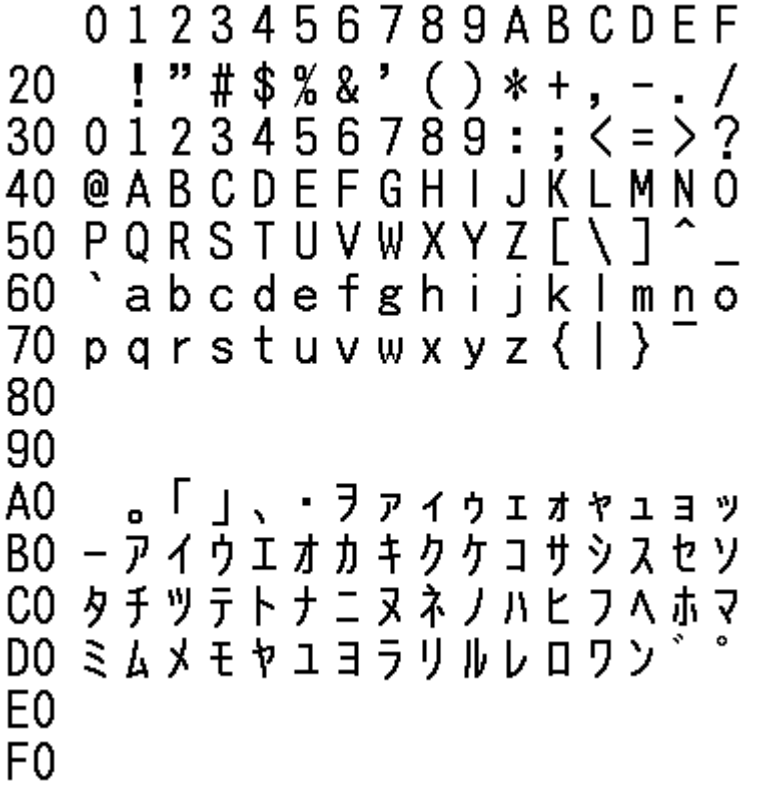


Figure A-4 SII_PM_CODE_PAGE_KATAKANA

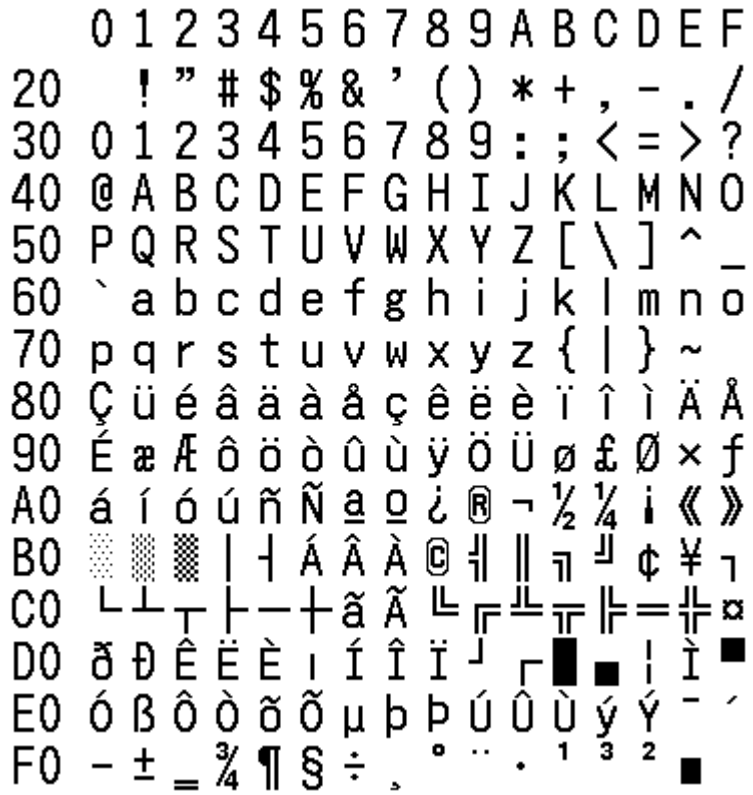


Figure A-5 SII_PM_CODE_PAGE_850 (Multilingual)

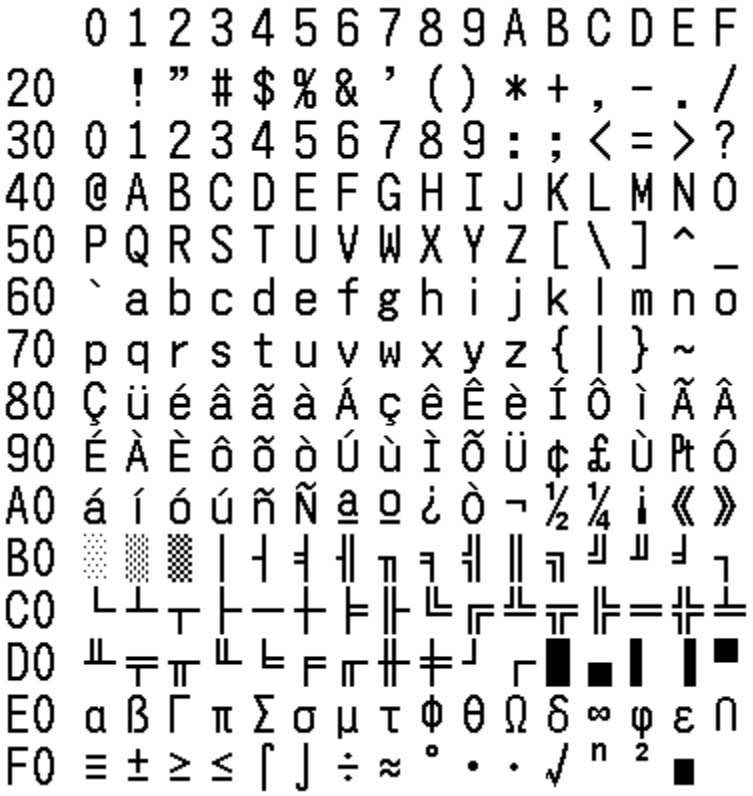


Figure A-6 SII_PM_CODE_PAGE_860 (Portuguese)

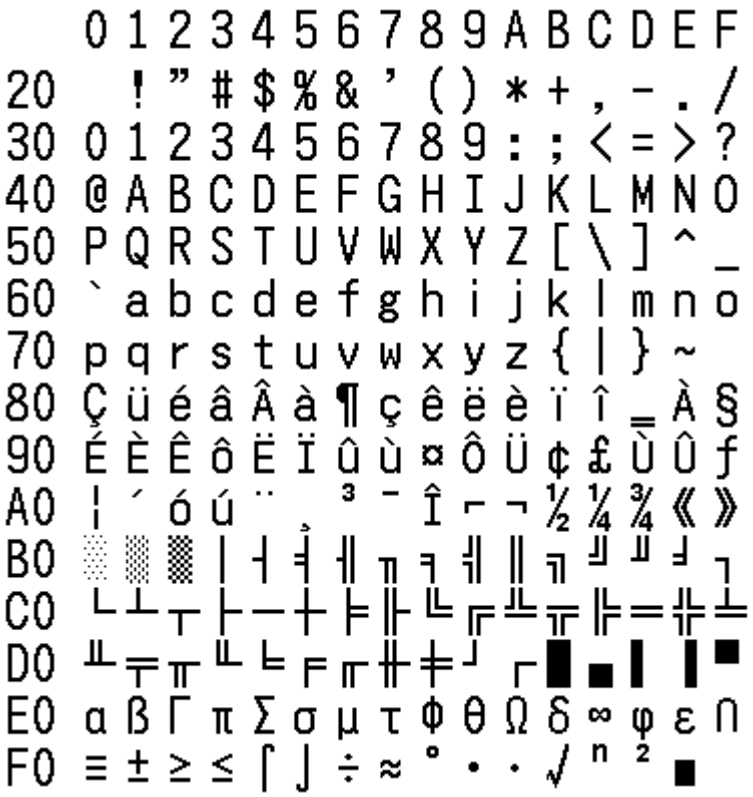


Figure A-7 SII_PM_CODE_PAGE_863 (Canadian-French)

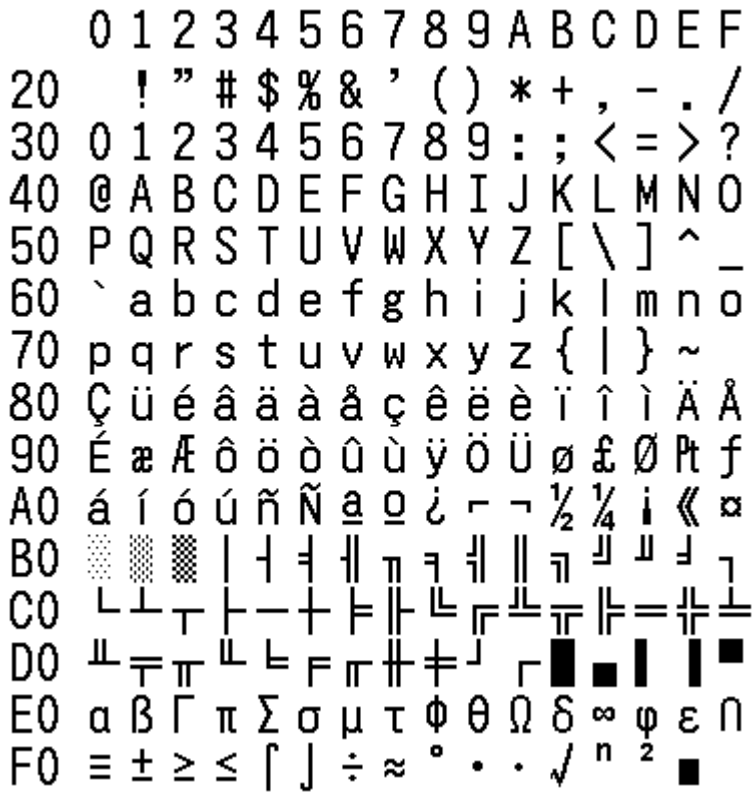


Figure A-8 SII_PM_CODE_PAGE_865 (Nordic)

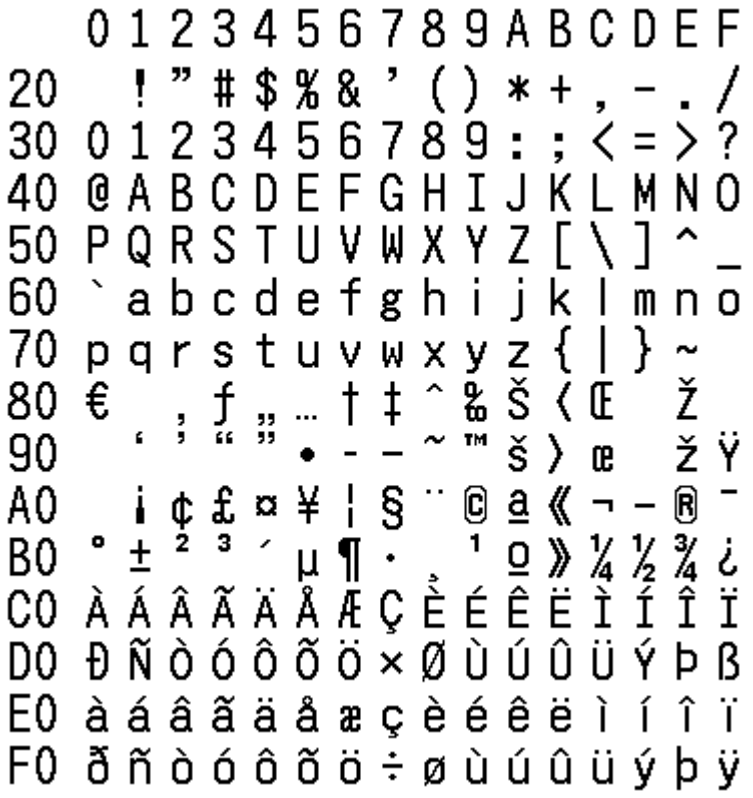


Figure A-9 SII_PM_CODE_PAGE_1252 (Latin)

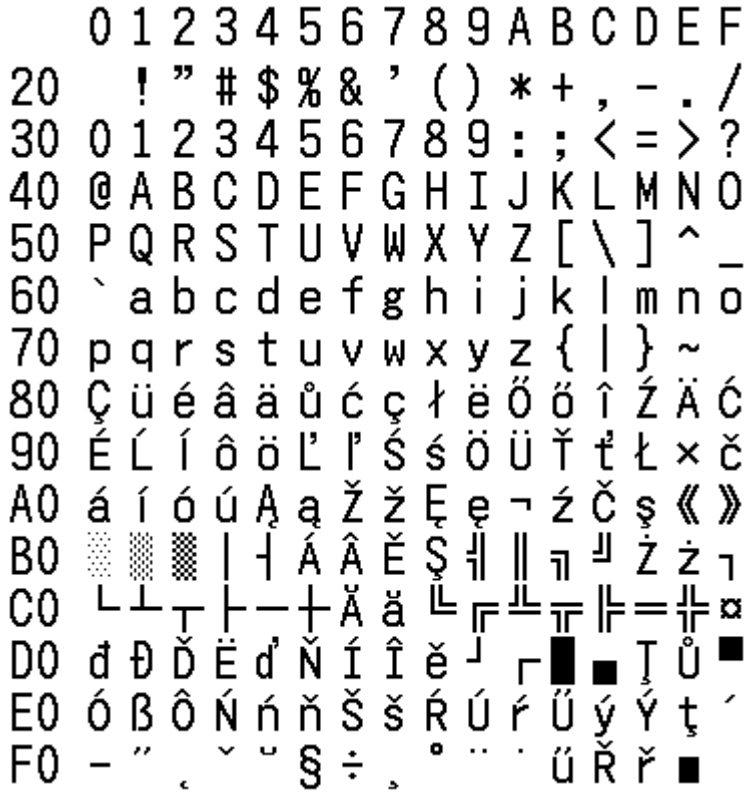


Figure A-10 SII_PM_CODE_PAGE_852 (Eastern Europe)

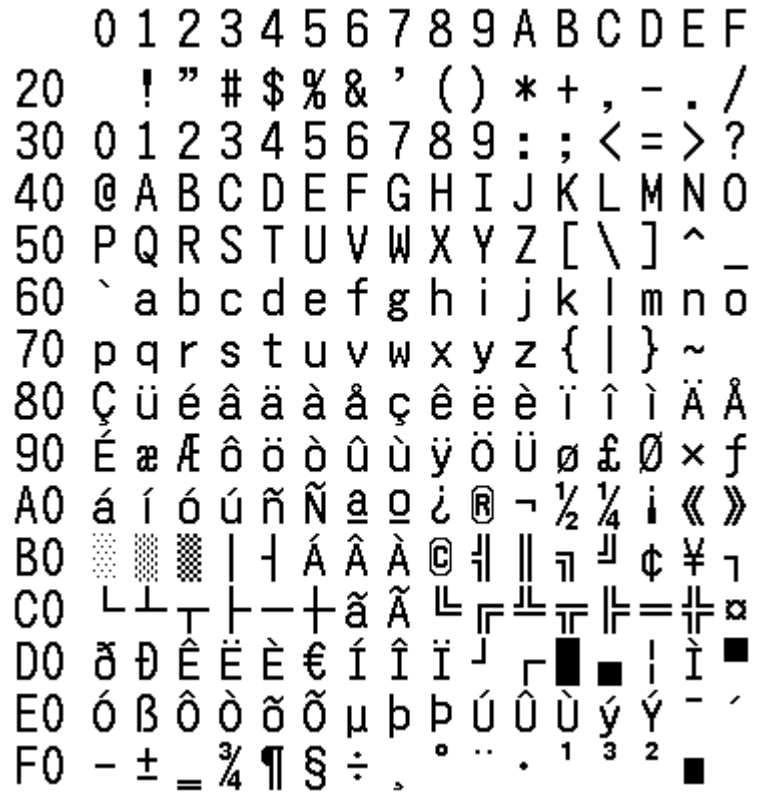


Figure A-11 SII_PM_CODE_PAGE_858 (Euro)



Figure A-12 SII_PM_CODE_PAGE_864 (Arabic)

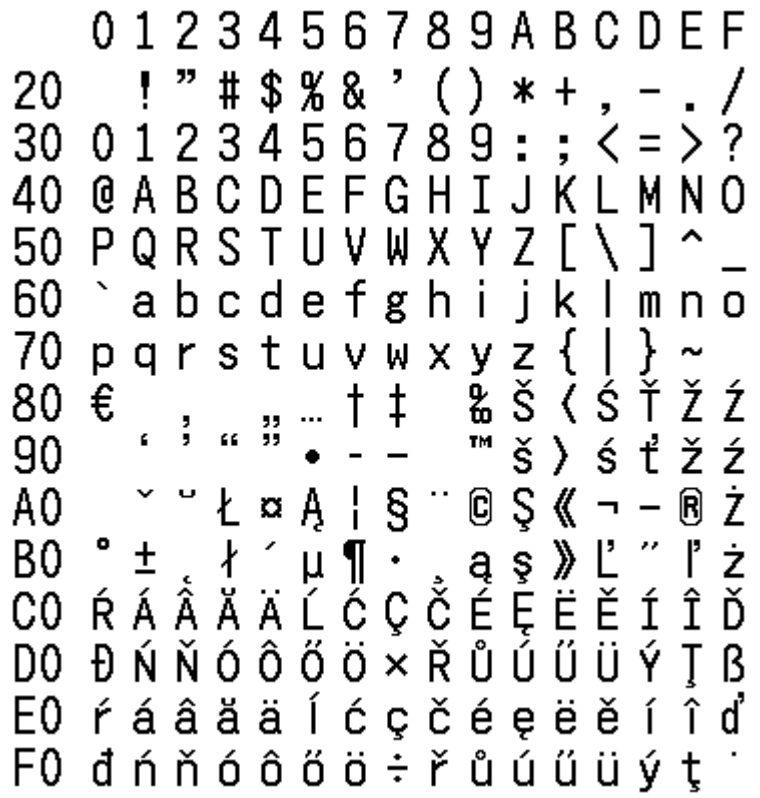


Figure A-13 SII_PM_CODE_PAGE_1250 (Central European)



Figure A-14 SII_PM_CODE_PAGE_1251 (Cyrillic)

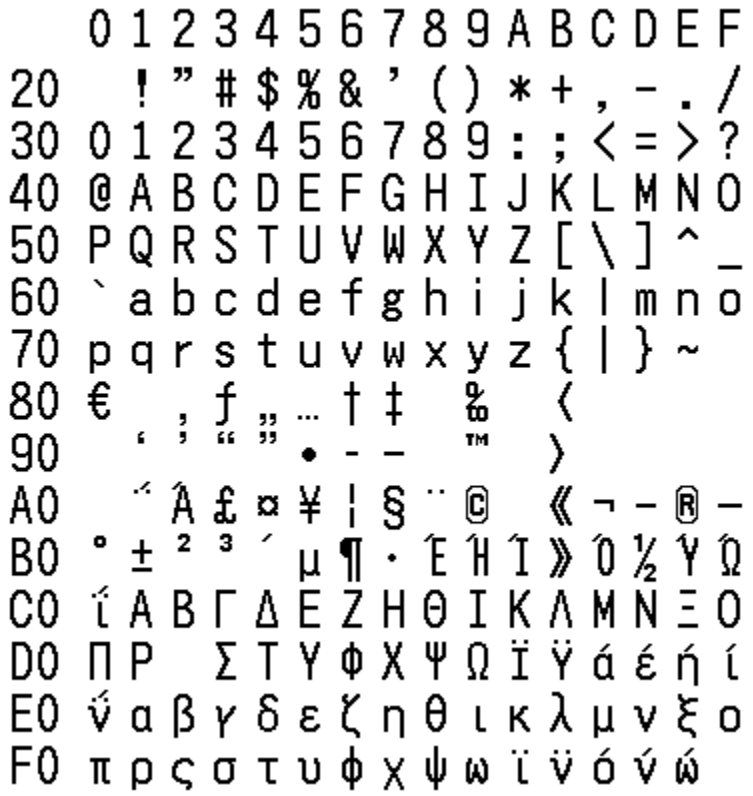


Figure A-15 SII_PM_CODE_PAGE_1253 (Greek)



Figure A-16 SII_PM_CODE_PAGE_1254 (Turkish)

A.2 International Character Set

Print results of the specific character codes or display results vary depending on the setting of the international character set.
The following table shows the specific character codes and their print results.

	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	#	¤	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü
COUNTRY_ITALY	#	$	@	°	\	é	^	ù	à	ò	è	ì
COUNTRY_SPAIN	₧	$	@	¡	Ñ	¿	^	`	¨	ñ	}	~
COUNTRY_JAPAN	#	$	@	[	¥	]	^	`	{	|	}	~
COUNTRY_NORWAY	#	¤	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	#	$	@	[	\	]	^	`	{	|	}	~

Figure A-17 International Character Set

Appendix B
Barcode Size List

B.1 Barcode Size List

B.1.1 `printBarcode, printPageModeBarcode`



(1) Height of the barcode image

hriFont	hriPosition	Length from Top of Barcode to Reference Point	Height of Barcode Image
SII_PM_FONT_A	SII_PM_HRI_NONE	moduleHeight	moduleHeight
	SII_PM_HRI_POSITION_ABOVE	moduleHeight + 32	moduleHeight + 32
	SII_PM_HRI_POSITION_BELOW	moduleHeight	moduleHeight + 32
	SII_PM_HRI_POSITION_ABOVE_BELOW	moduleHeight + 64	moduleHeight + 64
SII_PM_FONT_B	SII_PM_HRI_NONE	moduleHeight	moduleHeight
	SII_PM_HRI_POSITION_ABOVE	moduleHeight + 24	moduleHeight + 24
	SII_PM_HRI_POSITION_BELOW	moduleHeight	moduleHeight + 24
	SII_PM_HRI_POSITION_ABOVE_BELOW	moduleHeight + 48	moduleHeight + 48

(2) Width of the barcode image

barcodeSymbol	moduleSize	**Width of Barcode Image**
SII_PM_BARCODE_UPC_A	**SII_PM_BARCODE_MODULE_WIDTH_2**	226
	SII_PM_BARCODE_MODULE_WIDTH_3	339
	SII_PM_BARCODE_MODULE_WIDTH_4	452
	SII_PM_BARCODE_MODULE_WIDTH_5	565
	SII_PM_BARCODE_MODULE_WIDTH_6	678
SII_PM_BARCODE_UPC_E	**SII_PM_BARCODE_MODULE_WIDTH_2**	130
	SII_PM_BARCODE_MODULE_WIDTH_3	195
	SII_PM_BARCODE_MODULE_WIDTH_4	260
	SII_PM_BARCODE_MODULE_WIDTH_5	325
	SII_PM_BARCODE_MODULE_WIDTH_6	390
SII_PM_BARCODE_EAN13	**SII_PM_BARCODE_MODULE_WIDTH_2**	226
	SII_PM_BARCODE_MODULE_WIDTH_3	339
	SII_PM_BARCODE_MODULE_WIDTH_4	452
	SII_PM_BARCODE_MODULE_WIDTH_5	565
	SII_PM_BARCODE_MODULE_WIDTH_6	678
SII_PM_BARCODE_JAN13	**SII_PM_BARCODE_MODULE_WIDTH_2**	226
	SII_PM_BARCODE_MODULE_WIDTH_3	339
	SII_PM_BARCODE_MODULE_WIDTH_4	452
	SII_PM_BARCODE_MODULE_WIDTH_5	565
	SII_PM_BARCODE_MODULE_WIDTH_6	678
SII_PM_BARCODE_EAN8	**SII_PM_BARCODE_MODULE_WIDTH_2**	162
	SII_PM_BARCODE_MODULE_WIDTH_3	243
	SII_PM_BARCODE_MODULE_WIDTH_4	324
	SII_PM_BARCODE_MODULE_WIDTH_5	405
	SII_PM_BARCODE_MODULE_WIDTH_6	486
SII_PM_BARCODE_JAN8	**SII_PM_BARCODE_MODULE_WIDTH_2**	162
	SII_PM_BARCODE_MODULE_WIDTH_3	243
	SII_PM_BARCODE_MODULE_WIDTH_4	324
	SII_PM_BARCODE_MODULE_WIDTH_5	405
	SII_PM_BARCODE_MODULE_WIDTH_6	486
SII_PM_BARCODE_CODE93	**SII_PM_BARCODE_MODULE_WIDTH_2**	18 × number of barcode data + 96
	SII_PM_BARCODE_MODULE_WIDTH_3	27 × number of barcode data + 144
	SII_PM_BARCODE_MODULE_WIDTH_4	36 × number of barcode data + 192
	SII_PM_BARCODE_MODULE_WIDTH_5	45 × number of barcode data + 240
	SII_PM_BARCODE_MODULE_WIDTH_6	54 × number of barcode data + 288
SII_PM_BARCODE_CODE128	**SII_PM_BARCODE_MODULE_WIDTH_2**	22 × number of barcode data + 66
	SII_PM_BARCODE_MODULE_WIDTH_3	33 × number of barcode data + 99
	SII_PM_BARCODE_MODULE_WIDTH_4	44 × number of barcode data + 132

barcodeSymbol	moduleSize	Width of Barcode Image
SII_PM_BARCODE_CODE128	**SII_PM_BARCODE_MODULE_WIDTH_5**	55 × number of barcode data + 165
	SII_PM_BARCODE_MODULE_WIDTH_6	66 × number of barcode data + 198

barcodeSymbol	nwRatio	moduleSize	Width of Barcode Image
SII_PM_BARCODE_CODE39	**SII_PM_NWRATIO_1TO2**	**SII_PM_BARCODE_MODULE_WIDTH_2**	26 × number of barcode data + 90
		SII_PM_BARCODE_MODULE_WIDTH_3	39 × number of barcode data + 135
		SII_PM_BARCODE_MODULE_WIDTH_4	52 × number of barcode data + 180
		SII_PM_BARCODE_MODULE_WIDTH_5	65 × number of barcode data + 225
		SII_PM_BARCODE_MODULE_WIDTH_6	78 × number of barcode data + 270
	SII_PM_NWRATIO_1TO2_5	**SII_PM_BARCODE_MODULE_WIDTH_2**	29 × number of barcode data + 96
		SII_PM_BARCODE_MODULE_WIDTH_3	45 × number of barcode data + 147
		SII_PM_BARCODE_MODULE_WIDTH_4	58 × number of barcode data + 192
		SII_PM_BARCODE_MODULE_WIDTH_5	74 × number of barcode data + 243
		SII_PM_BARCODE_MODULE_WIDTH_6	87 × number of barcode data + 288
	SII_PM_NWRATIO_1TO3	**SII_PM_BARCODE_MODULE_WIDTH_2**	32 × number of barcode data + 102
		SII_PM_BARCODE_MODULE_WIDTH_3	48 × number of barcode data + 153
		SII_PM_BARCODE_MODULE_WIDTH_4	64 × number of barcode data + 204
		SII_PM_BARCODE_MODULE_WIDTH_5	80 × number of barcode data + 255
		SII_PM_BARCODE_MODULE_WIDTH_6	96 × number of barcode data + 306
SII_PM_BARCODE_ITF	**SII_PM_NWRATIO_1TO2**	**SII_PM_BARCODE_MODULE_WIDTH_2**	14 × number of barcode data + 56
		SII_PM_BARCODE_MODULE_WIDTH_3	21 × number of barcode data + 84
		SII_PM_BARCODE_MODULE_WIDTH_4	28 × number of barcode data + 112
		SII_PM_BARCODE_MODULE_WIDTH_5	35 × number of barcode data + 140
		SII_PM_BARCODE_MODULE_WIDTH_6	42 × number of barcode data + 168

barcodeSymbol	nwRatio	moduleSize	Width of Barcode Image
SII_PM_BARCODE_ITF	**SII_PM_NWRATIO_1TO2_5**	**SII_PM_BARCODE_MODULE_WIDTH_2**	16 × number of barcode data + 57
		SII_PM_BARCODE_MODULE_WIDTH_3	25 × number of barcode data + 86
		SII_PM_BARCODE_MODULE_WIDTH_4	32 × number of barcode data + 114
		SII_PM_BARCODE_MODULE_WIDTH_5	41 × number of barcode data + 143
		SII_PM_BARCODE_MODULE_WIDTH_6	48 × number of barcode data + 171
	SII_PM_NWRATIO_1TO3	**SII_PM_BARCODE_MODULE_WIDTH_2**	18 × number of barcode data + 58
		SII_PM_BARCODE_MODULE_WIDTH_3	27 × number of barcode data + 87
		SII_PM_BARCODE_MODULE_WIDTH_4	36 × number of barcode data + 116
		SII_PM_BARCODE_MODULE_WIDTH_5	45 × number of barcode data + 145
		SII_PM_BARCODE_MODULE_WIDTH_6	54 × number of barcode data + 174
SII_PM_BARCODE_CODABAR*1	**SII_PM_NWRATIO_1TO2**	**SII_PM_BARCODE_MODULE_WIDTH_2**	20 × number of data + 2 × (2 + number of wide data) + 38
		SII_PM_BARCODE_MODULE_WIDTH_3	30 × number of data + 3 × (2 + number of wide data) + 57
		SII_PM_BARCODE_MODULE_WIDTH_4	40 × number of data + 4 × (2 + number of wide data) + 76
		SII_PM_BARCODE_MODULE_WIDTH_5	50 × number of data + 5 × (2 + number of wide data) + 95
		SII_PM_BARCODE_MODULE_WIDTH_6	60 × number of data + 6 × (2 + number of wide data) + 114
	SII_PM_NWRATIO_1TO2_5	**SII_PM_BARCODE_MODULE_WIDTH_2**	22 × number of data + 3 × (2 + number of wide data) + 38
		SII_PM_BARCODE_MODULE_WIDTH_3	34 × number of data + 5 × (2 + number of wide data) + 57
		SII_PM_BARCODE_MODULE_WIDTH_4	44 × number of data + 6 × (2 + number of wide data) + 76
		SII_PM_BARCODE_MODULE_WIDTH_5	56 × number of data + 8 × (2 + number of wide data) + 95
		SII_PM_BARCODE_MODULE_WIDTH_6	66 × number of data + 9 × (2 + number of wide data) + 114

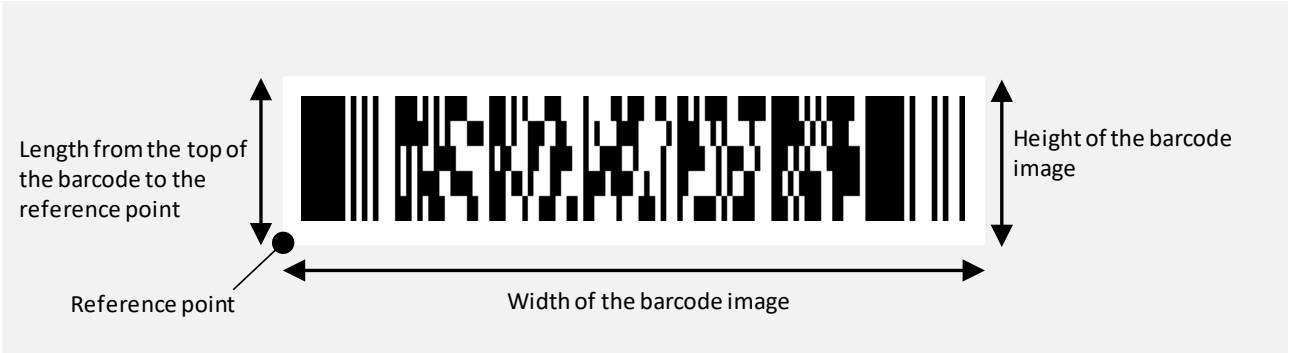
*1: The number of data is the number of all characters except for the start and stop characters.
The wide data is the number of ": / . +".

barcodeSymbol	nwRatio	moduleSize	Width of Barcode Image
SII_PM_BARCODE_CODABAR*1	**SII_PM_NWRATIO_1TO3**	**SII_PM_BARCODE_MODULE_WIDTH_2**	24 × number of data + 4 × (2 + number of wide data) + 38
		SII_PM_BARCODE_MODULE_WIDTH_3	36 × number of data + 6 × (2 + number of wide data) + 57
		SII_PM_BARCODE_MODULE_WIDTH_4	48 × number of data + 8 × (2 + number of wide data) + 76
		SII_PM_BARCODE_MODULE_WIDTH_5	60 × number of data + 10 × (2 + number of wide data) + 95
		SII_PM_BARCODE_MODULE_WIDTH_6	72 × number of data + 12 × (2 + number of wide data) + 114

*1: The number of data is the number of all characters except for the start and stop characters.
The wide data is the number of ": / . +".

barcodeSymbol	Number of Data	moduleSize	Width of Barcode Image
SII_PM_BARCODE_EAN13_ADDON	14 or 15	**SII_PM_BARCODE_MODULE_WIDTH_2**	276
		SII_PM_BARCODE_MODULE_WIDTH_3	414
		SII_PM_BARCODE_MODULE_WIDTH_4	552
		SII_PM_BARCODE_MODULE_WIDTH_5	690
		SII_PM_BARCODE_MODULE_WIDTH_6	828
	17 or 18	**SII_PM_BARCODE_MODULE_WIDTH_2**	330
		SII_PM_BARCODE_MODULE_WIDTH_3	495
		SII_PM_BARCODE_MODULE_WIDTH_4	660
		SII_PM_BARCODE_MODULE_WIDTH_5	825
		SII_PM_BARCODE_MODULE_WIDTH_6	990
SII_PM_BARCODE_JAN13_ADDON	14 or 15	**SII_PM_BARCODE_MODULE_WIDTH_2**	276
		SII_PM_BARCODE_MODULE_WIDTH_3	414
		SII_PM_BARCODE_MODULE_WIDTH_4	552
		SII_PM_BARCODE_MODULE_WIDTH_5	690
		SII_PM_BARCODE_MODULE_WIDTH_6	828
	17 or 18	**SII_PM_BARCODE_MODULE_WIDTH_2**	330
		SII_PM_BARCODE_MODULE_WIDTH_3	495
		SII_PM_BARCODE_MODULE_WIDTH_4	660
		SII_PM_BARCODE_MODULE_WIDTH_5	825
		SII_PM_BARCODE_MODULE_WIDTH_6	990

B.1.2 `printPDF417, printPageModePDF417`



(1) Height of the barcode image

`moduleSize`	**Height of Barcode Image*1**
SII_PM_PDF417_MODULE_WIDTH_2	`moduleHeight` × `row`*2 + 8
SII_PM_PDF417_MODULE_WIDTH_3	`moduleHeight` × `row`*2 + 12
SII_PM_PDF417_MODULE_WIDTH_4	`moduleHeight` × `row`*2 + 16

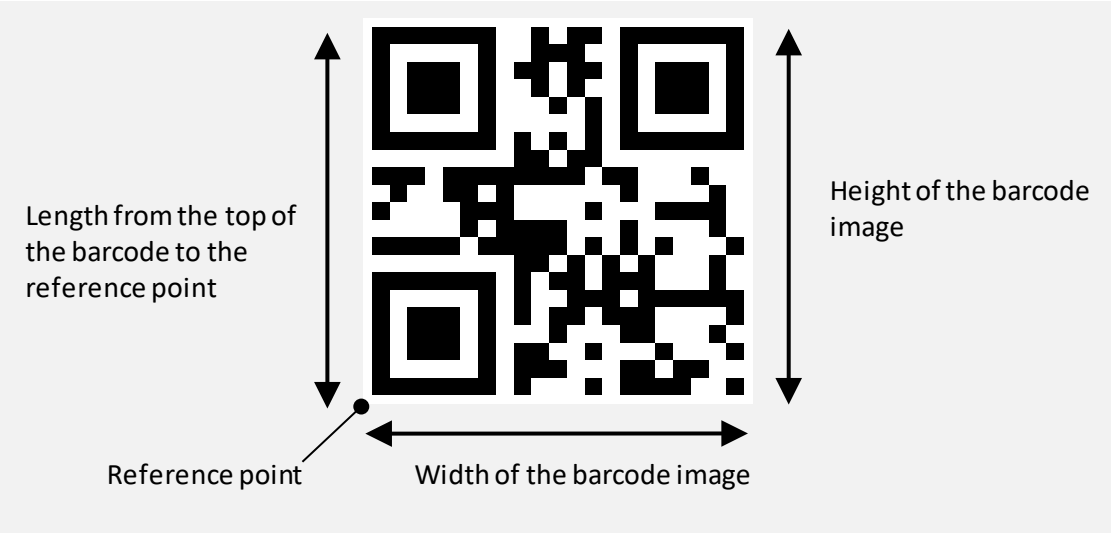
*1: Height of the barcode image = Length from the top of the barcode to the reference point
*2: Parameter `row`≠0

(2) Width of the barcode image

`pdf417Symbol`	`moduleSize`	**Width of Barcode Image**
SII_PM_PDF417_STANDARD	**SII_PM_PDF417_MODULE_WIDTH_2**	(17 × `column`*1 + 69) × 2 + 8
	SII_PM_PDF417_MODULE_WIDTH_3	(17 × `column`*1 + 69) × 3 + 12
	SII_PM_PDF417_MODULE_WIDTH_4	(17 × `column`*1 + 69) × 4 + 16
SII_PM_PDF417_COMPACT	**SII_PM_PDF417_MODULE_WIDTH_2**	(17 × `column`*1 + 69) × 2 + 8
	SII_PM_PDF417_MODULE_WIDTH_3	(17 × `column`*1 + 69) × 3 + 12
	SII_PM_PDF417_MODULE_WIDTH_4	(17 × `column`*1 + 69) × 4 + 16

*1: Parameter `column`≠0

B.1.3 `printQRCode, printPageModeQRCode`



(1) Height and width of the barcode image

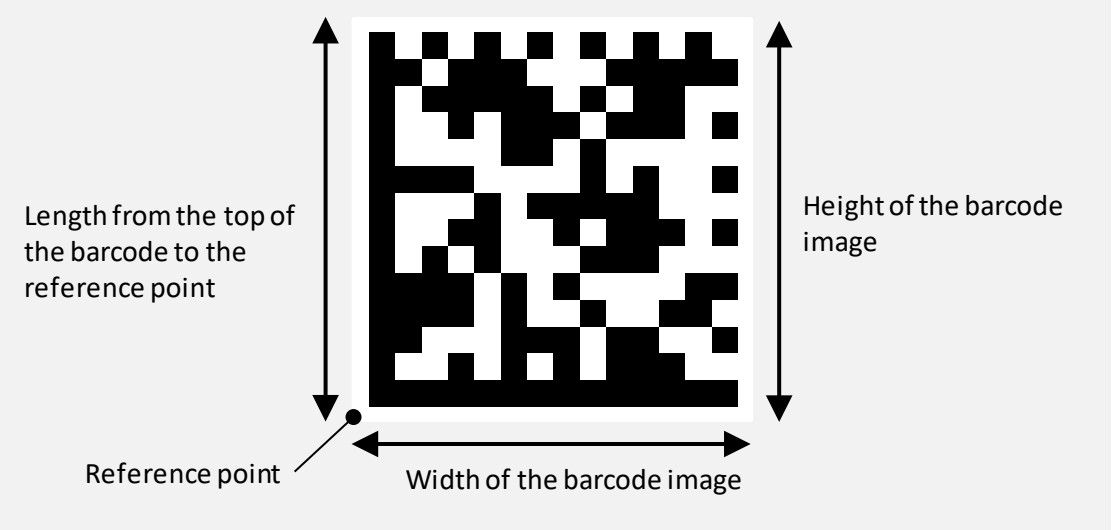
Height[*1] and width of the barcode image = (4 × version[*2] + 17 + 8) × module size value

*1: Height of the barcode image = Length from the top of the barcode to the reference point
*2: The version is determined by the content of the barcode data and the error correction level.

Module Size Value

`moduleSize`	**Module Size Value**
SII_PM_QR_MODULE_SIZE_2	2
SII_PM_QR_MODULE_SIZE_3	3
SII_PM_QR_MODULE_SIZE_4	4
SII_PM_QR_MODULE_SIZE_5	5
SII_PM_QR_MODULE_SIZE_6	6
SII_PM_QR_MODULE_SIZE_7	7
SII_PM_QR_MODULE_SIZE_8	8
SII_PM_QR_MODULE_SIZE_9	9
SII_PM_QR_MODULE_SIZE_10	10
SII_PM_QR_MODULE_SIZE_11	11

B.1.4 `printDataMatrix, printPageModeDataMatrix`



(1) Height and width of the barcode image

Height of the barcode image = (number of vertical module + 2) × module size value

Width of the barcode image = (number of horizontal module + 2) × module size value

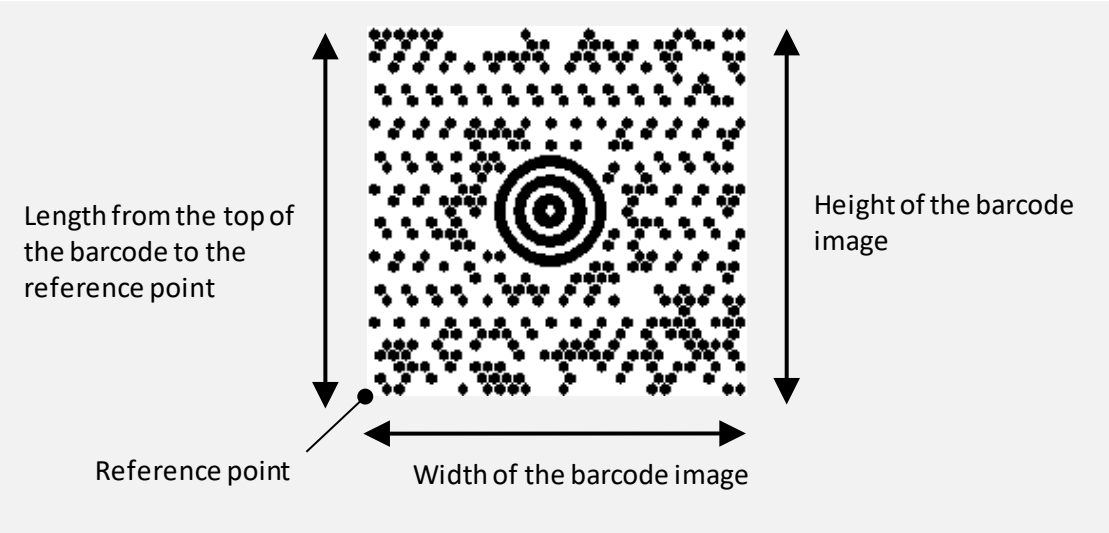
dataMatrixModule	Number of Vertical Module	Number of Horizontal Module
SII_PM_DATA_MATRIX_10_10	10	10
SII_PM_DATA_MATRIX_12_12	12	12
SII_PM_DATA_MATRIX_14_14	14	14
SII_PM_DATA_MATRIX_16_16	16	16
SII_PM_DATA_MATRIX_18_18	18	18
SII_PM_DATA_MATRIX_20_20	20	20
SII_PM_DATA_MATRIX_22_22	22	22
SII_PM_DATA_MATRIX_24_24	23	23
SII_PM_DATA_MATRIX_26_26	26	26
SII_PM_DATA_MATRIX_32_32	32	32
SII_PM_DATA_MATRIX_36_36	36	36
SII_PM_DATA_MATRIX_40_40	40	40
SII_PM_DATA_MATRIX_44_44	44	44
SII_PM_DATA_MATRIX_48_48	48	48
SII_PM_DATA_MATRIX_52_52	52	52
SII_PM_DATA_MATRIX_64_64	64	64
SII_PM_DATA_MATRIX_72_72	72	72
SII_PM_DATA_MATRIX_80_80	80	80
SII_PM_DATA_MATRIX_88_88	88	88
SII_PM_DATA_MATRIX_96_96	96	96
SII_PM_DATA_MATRIX_104_104	104	104

dataMatrixModule	Number of Vertical Module	Number of Horizontal Module
SII_PM_DATA_MATRIX_120_120	120	120
SII_PM_DATA_MATRIX_132_132	132	132
SII_PM_DATA_MATRIX_144_144	144	144
SII_PM_DATA_MATRIX_8_18	8	18
SII_PM_DATA_MATRIX_8_32	8	32
SII_PM_DATA_MATRIX_12_26	12	26
SII_PM_DATA_MATRIX_12_36	12	36
SII_PM_DATA_MATRIX_16_36	16	36
SII_PM_DATA_MATRIX_16_48	16	48

Module Size Value

moduleSize	Module Size Value
SII_PM_DATAMATRIX_MODULE_SIZE_2	2
SII_PM_DATAMATRIX_MODULE_SIZE_3	3
SII_PM_DATAMATRIX_MODULE_SIZE_4	4
SII_PM_DATAMATRIX_MODULE_SIZE_5	5
SII_PM_DATAMATRIX_MODULE_SIZE_6	6
SII_PM_DATAMATRIX_MODULE_SIZE_7	7
SII_PM_DATAMATRIX_MODULE_SIZE_8	8
SII_PM_DATAMATRIX_MODULE_SIZE_9	9
SII_PM_DATAMATRIX_MODULE_SIZE_10	10
SII_PM_DATAMATRIX_MODULE_SIZE_11	11

B.1.5 `printMaxicode, printPageModeMaxicode`



(1) Height of the barcode image

Height of the barcode image*1 = 215

*1: Height of the barcode image = Length from the top of the barcode to the reference point

(2) Width of the barcode image

Width of the barcode image = 225

Appendix C
Open Source Software License

This chapter describes the License of open source software used in the library.

C.1 MIT License

- **SSZipArchive**

  Copyright (c) 2010-2012 Sam Soffes

  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.

C.2 Apache License 2.0

- **zxingify-objc**

  Copyright 2012 ZXing authors

  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.