# SII Print Class Library for iOS
# Application Programmer's Guide

Rev.13

[Products]

RP-F10 Series

Seiko Instruments Inc.

| | |
|---|---|
| Rev.01 | February 2019 |
| Rev.02 | July 2019 |
| Rev.03 | March 2020 |
| Rev.04 | June 2020 |
| Rev.05 | August 2020 |
| Rev.06 | November 2020 |
| Rev.07 | June 2021 |
| Rev.08 | March 2022 |
| Rev.09 | October 2022 |
| Rev.10 | December 2022 |
| Rev.11 | April 2023 |
| Rev.12 | February 2024 |
| Rev.13 | March 2024 |

Copyright © 2019-2024 Seiko Instruments Inc.
All rights reserved.

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

Company names or product names in the text may be trademarks or registered trademarks of each company.

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.

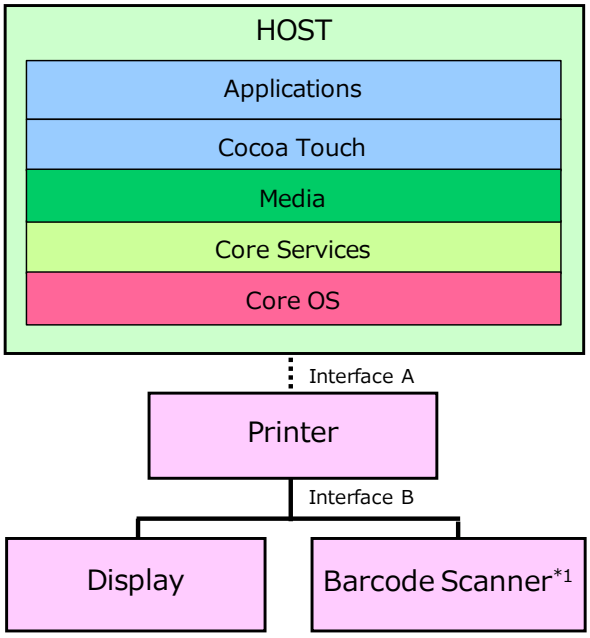
# INTRODUCTION

This manual describes "SII Print Class Library for iOS" (hereinafter referred to as "SII print class library") provided by Seiko Instruments Inc. (hereinafter referred to as "SII").

## Target Printers

The printers supported by SII print class library are listed below.

| Printer | Interface A | Display | Interface B |
|---|---|---|---|
| RP-F10 Series | Bluetooth | DSP-A01 Series | USB |
| | USB | | |
| | TCP/IP | | |




*1: See "RP-F10 SERIES THERMAL PRINTER USER'S GUIDE" for details about the combination of peripherals including the barcode scanner.

## Terms

The terms used in this manual are described below.

Printer

| Term | Description |
|---|---|
| Technical Reference for Printer | The following technical reference.<br>・RP-F10 SERIES THERMAL PRINTER TECHNICAL REFERENCE |
| Printer command | Command for controlling the printer described in "Technical Reference for Printer". |

Display

| Term | Description |
|---|---|
| Technical Reference for Display | The following technical reference.<br>・DSP-A01 SERIES CUSTOMER DISPLAY TECHNICAL REFERENCE |
| Display command | Command for controlling Display described in "Technical Reference for Display". |
| Slide | The image data of the screen size (Width 480 px × Height 272 px). Displays as a standby screen and as a backscreen superimposed on a template. |
| Template | The stylized form having elements that can set attributes such as drawing areas and mapping positions. The elements include text elements (text data), img elements (image data), barcode elements (barcode data), and qr elements (QR Code data).<br>To register templates, define a map ID each for an element to place.<br>Registered image data or text data is shown on the display by updating the screen after selecting a template and specifying its map ID.<br>The data in the template is required to be specified XML file format.<br>The maximum size of template data is width 480 px × height 272 px. |
| Map ID | An ID defined to an element which is holding positional information or modification information when a template is registered. |
| Macro | A function to register multiple APIs in order of execution, and execute automatically when an event occurs. |
| Event | An event which is defined by "Event notification" in Display commands. |

# Table of Contents


## Chapter 1 Product Overview 1-1

1.1 Functions Provided by SII Print Class Library ..... 1-1
1.2 SII Print Class Library Overview ..... 1-1
1.2.1 SII Print Class Library Configuration ..... 1-1
1.2.2 Functions Provided by Library ..... 1-2
1.2.3 Development of Application that Performs Bluetooth Communication with SII Printer ..... 1-2
1.2.4 Registered Data in Display at Shipping ..... 1-2

## Chapter 2 Product Specifications 2-1

2.1 Operating Environment ..... 2-1
2.1.1 Applicable iOS Devices ..... 2-1
2.1.2 Applicable iOS Versions ..... 2-2
2.2 Printer Settings ..... 2-3
2.3 Precaution ..... 2-4

## Chapter 3 How to Use library 3-1

3.1 iOS Application Development Environment ..... 3-1
3.2 Provided Files ..... 3-2
3.3 Build Library to Xcode Project ..... 3-3
3.3.1 Objective-C ..... 3-3
3.3.2 Swift ..... 3-7

## Chapter 4 Functions 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 ..... 4-9

(1) Method List ..... 4-9
① Common method to standard mode and page mode ..... 4-9
② Dedicated method for standard mode ..... 4-11
③ Dedicated method for page mode ..... 4-11
(2) Common property list to standard mode and page mode ..... 4-12
(3) Constant List ..... 4-13
① Printer model ..... 4-13
② Port type ..... 4-13
③ Printer response type ..... 4-13
④ Display response type ..... 4-14
⑤ International character set ..... 4-14
⑥ Codepage ..... 4-15
⑦ Barcode and PDF417 ..... 4-15
(4) Enumerated Constant List ..... 4-16
① Dithering (Dithering) ..... 4-16
② Batch processing selection (TransactionFunction) ..... 4-16
③ Bold print (CharacterBold) ..... 4-16
④ Underline (CharacterUnderline) ..... 4-16
⑤ Reverse print (CharacterReverse) ..... 4-17
⑥ Inversion print (CharacterInversion) ..... 4-17
⑦ Character font (CharacterFont) ..... 4-17
⑧ Character scale (CharacterScale) ..... 4-17
⑨ Alignment (PrintAlignment) ..... 4-18
⑩ Barcode symbol (BarcodeSymbol) ..... 4-18
⑪ Module size (ModuleSize) ..... 4-19
⑫ HRI character print position (HriPosition) ..... 4-21
⑬ N:W ratio (NwRatio) ..... 4-21
⑭ Error correction level (ErrorCorrection) ..... 4-22
⑮ PDF417 symbol (Pdf417Symbol) ..... 4-22
⑯ QR Code Model (QrModel) ..... 4-22
⑰ Data Matrix module (DataMatrixModule) ..... 4-23
⑱ MaxiCode Mode (MaxiCodeMode) ..... 4-24
⑲ Cutting method (CuttingMethod) ..... 4-24
⑳ Drawer number (DrawerNum) ..... 4-24
㉑ Pulse width (PulseWidth) ..... 4-25
㉒ Buzzer pattern (BuzzerPattern) ..... 4-25
㉓ Memory area (MemoryArea) ..... 4-25
㉔ Registered font (RegisterdFont) ..... 4-25
㉕ QR data mode (QrDataMode) ..... 4-26
㉖ QR quiet zone (QrQuietZone) ..... 4-26
㉗ Macro registration processing (MacroRegistrationFunction) ..... 4-26
㉘ Print direction (Direction) ..... 4-27
㉙ Line style (LineStyle) ..... 4-27
(5) Method Details ..... 4-28
① Common method to standard mode and page mode ..... 4-28
init Instance ..... 4-28
connect Start communicating with printer ..... 4-28
disconnect Stop communicating with printer ..... 4-29

openDrawer Open cash drawer ..... 4-29
buzzer Sound buzzer ..... 4-30
externalBuzzer Sound external buzzer ..... 4-30
getStatus Get printer status ..... 4-30
abort Abort waiting state of printer ..... 4-31
registerLogo Register logo ..... 4-32
unregisterLogo Delete registered logo ..... 4-32
registerStyleSheet
Register style sheet ..... 4-32
unregisterStyleSheet
Delete registered style sheet ..... 4-32
resetPrinter Reset printer ..... 4-33
getPrinterResponse
Get various responses from printer ..... 4-33
startDiscoveryPrinter
Start printer search (Bluetooth) ..... 4-34
startDiscoveryPrinter
Start printer search (TCP/IP) ..... 4-35
cancelDiscoveryPrinter
Cancel printer search ..... 4-35
getFoundPrinter
Get found printer information ..... 4-36
getVersion Get SDK version ..... 4-36
controlTransaction
Start/End batch processing ..... 4-36
defragment Optimize memory area ..... 4-38
initializeMemoryArea
Initialize memory area ..... 4-38
showTemplate Display template ..... 4-39
showSlide Display slide ..... 4-40
enterStandbyMode
Display standby ..... 4-40
executeMacro Execute macro ..... 4-41
turnOnScreen Turn on/off screen ..... 4-41
selectTemplate Select template ..... 4-41
setTemplateImageData
Set image data ..... 4-42
selectTemplateTextObject
Select text element ..... 4-43
setTemplateTextAlignment
Alignment of text data ..... 4-43
setTemplateTextLeftMargin
Set left margin of text data ..... 4-44
setTemplateTextLineSpacing
Set line spacing of text data ..... 4-44
setTemplateTextBold
Set bold character of text data ..... 4-45

setTemplateTextUnderline
Set underline of text data ..... 4-45
setTemplateTextSize
Set character size of text data ..... 4-46
setTemplateTextFont
Set character font of text data ..... 4-46
setTemplateTextRegisteredFont
Set registered font of text data ..... 4-47
setTemplateTextRightSpacing
Set right space of text data ..... 4-47
setTemplateTextColor
Set character color of text data ..... 4-48
setTemplateTextData
Input text data ..... 4-48
setTemplateBarcodeData
Input barcode data ..... 4-49
setTemplateQrCodeData
Input QR Code data ..... 4-50
registerTemplate
Register template ..... 4-51
unregisterTemplate
Delete template ..... 4-52
registerImageData
Register image data ..... 4-52
unregisterImageData
Delate image data ..... 4-53
registerSlideData
Register slide data ..... 4-54
unregisterSlideData
Delete slide data ..... 4-55
registerUserDefinedCharacter
Register user-defined character ..... 4-55
unregisterUserDefinedCharacter
Delete user-defined character ..... 4-56
registerOptionFont
Register optional font ..... 4-56
unregisterOptionFont
Delete optional font ..... 4-57
controlMacroRegistration
Start/End of macro registration ..... 4-57
getDisplayResponse
Get various response from Display ..... 4-59
② Dedicated method for standard mode ..... 4-61
sendText Send text data ..... 4-61
sendTextEx Send format specified text data ..... 4-61
printBarcode Print barcode ..... 4-62
printPDF417 Print PDF417 ..... 4-66
printQRcode Print QR Code ..... 4-67

printDataMatrix
Print Data Matrix....4-68
printMaxiCode Print MaxiCode....4-69
printGS1DataBarStacked
Print GS1 Databar Stacked....4-69
printGS1DataBarStackedOmnidirectional
Print GS1 Databar Stacked Omni-directional....4-70
printGS1DataBarExpandedStacked
Print GS1 Databar Expanded Stacked....4-70
printAztecCode
Print Aztec Code....4-71
cutPaper Cut paper....4-71
feedPosition Paper form feed....4-71
sendBinary Send binary data....4-71
sendDataFile Send specified file....4-72
printLogo Print logo....4-73
③ Dedicated method for page mode....4-74
enterPageMode Start page mode....4-75
exitPageMode End page mode....4-75
setPageModeArea
Specify print area of page mode....4-75
setPageModeDirection
Specify print direction of page mode....4-77
setPageModeLineSpacing
Specify line spacing of page mode....4-77
printPageMode
Print page mode....4-77
printPageModeText
Send text data of page mode....4-78
printPageModeTextEx
Send format specified text data of page mode....4-78
printPageModeBarcode
Print barcode of page mode....4-79
printPageModePDF417
Print PDF417 of page mode....4-83
printPageModeQRcode
Print QR Code of page mode....4-84
printPageModeDataMatrix
Print Data Matrix of page mode....4-85
printPageModeMaxiCode
Print MaxiCode of page mode....4-85
printPageModeGS1DataBarStacked
Print GS1 Databar Stacked of page mode....4-86
printPageModeGS1DataBarStackedOmnidirectional
Print GS1 Databar Stacked Omni-directional of page mode....4-87

printPageModeGS1DataBarExpandedStacked
Print GS1 Databar Expanded Stacked of page mode ..... 4-87
printPageModeAztecCode
Print Aztec Code of page mode ..... 4-88
sendPageModeBinary
Send binary data of page mode ..... 4-88
printPageModeImageFile
Draw Image file of page mode ..... 4-89
printPageModeRectangle
Draw rectangle image of page mode ..... 4-89
printPageModeLine
Print ruled line of page mode ..... 4-90
printPageModeLogo
Print logo of page mode ..... 4-92
(6) Common property detail to standard mode and page mode ..... 4-93
sendTimeout Get/Set send timeout period ..... 4-93
receiveTimeout
Get/Set receive timeout period ..... 4-93
internationalCharacter
Get/Set international character set ..... 4-93
codePage Get/Set codepage ..... 4-94
printerModel Get printer model ..... 4-94
portType Get connecting port type ..... 4-94
isConnect Verify connection state with printer ..... 4-94
socketKeepingTime
Get/Set socket keeping time ..... 4-95
delegate Register delegate ..... 4-95
4.3.2 SIIPrinterInfo Class ..... 4-96
(1) Method List ..... 4-96
(2) Property List ..... 4-96
(3) Method Details ..... 4-96
SIIPrinterInfo Constructor ..... 4-96
(4) Property Details ..... 4-97
name Get printer model name ..... 4-97
mac Get MAC address ..... 4-97
ip Get IP address ..... 4-97
4.3.3 SIIPrinterException Class ..... 4-98
(1) Method List ..... 4-98
(2) Property List ..... 4-98
(3) Constant List ..... 4-99
① Error code ..... 4-99
(4) Method Details ..... 4-101
SIIPrinterException
Constructor ..... 4-101
(5) Property Details ..... 4-101
errorCode Get error code ..... 4-101
errorMessage Get error message ..... 4-101

4.3.4 SIIPrinterManagerDelegate Protocol ....4-102
(1) Method List ....4-102
(2) Method Details ....4-102
didStatusChange
Notify printer status....4-102
didBarcodeScannerReadData
Receipt notify of barcode data....4-103
didBarcodeScannerChangedOnline ....
Connection notify of barcode scanner....4-103
didBarcodeScannerChangedOffline ....
Disconnection notify of barcode scanner ....4-103
4.3.5 SIISmartLabelManager Class ....4-104

Chapter 5 Sample Program 5-1

5.1 Screen Layout....5-1
5.2 Precaution....5-2

Appendix A Character Set A-1

A-1 Codepage Table (Character Code Table) .... A-1
A-2 International Character Set.... A-11

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
B.1.6 printGS1DataBarStacked, printPageModeGS1DataBarStacked.... B-12
B.1.7 printGS1DataBarStackedOmnidirectional,
printPageModeGS1DataBarStackedOmnidirectional .... B-13
B.1.8 printGS1DataBarExpandedStacked,
printPageModeGS1DataBarExpandedStacked .... B-14

Appendix C Open Source Software License

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 SII print class library.

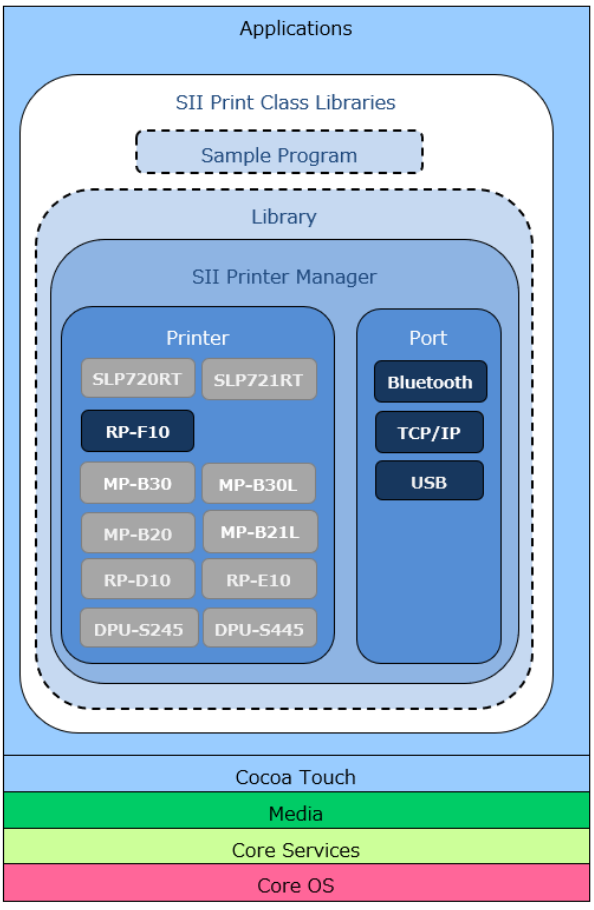
## 1.1 Functions Provided by SII Print Class Library

The SII print class library including the library and the sample program provides the functions to use SII printer RP-F10 Series (hereinafter referred to as "printer") in iOS applications.
Moreover, the SII print class library provides the library sample program in Xcode project.

## 1.2 SII Print Class Library Overview

### 1.2.1 SII Print Class Library Configuration

The library and the sample program in the SII print class library are indicated with dashed lines in the figure below.



### 1.2.2 Functions Provided by Library

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

The library provides the following functions:

- Connecting to / disconnecting from a printer
- Sending data to a printer (print data and/or printer commands*1)
- Printing barcode and 2-dimensional barcode
- Sending a data file to a printer (print data and/or printer commands*1)
- Cutting paper
- Getting the printer status
- Aborting the waiting state of a printer
- Getting various responses from a printer
- Bulk registration of print commands
- Registering a printer status call back function
- Searching the printer by TCP/IP
- Printer hardware reset
- Drawer operation control
- Buzzer beeping control
- Screen display control
- Registering a barcode scanner call back function
- Outputting a log file

*1: Commands that read the response from the printer are not supported.
In order to read responses from the printer, use **`getStatus`** or **`getPrinterResponse`**.

**(NOTE)** **・ RP-F10 does not support the APIs relating to label printing function.**
**・ See "RP-F10 SERIES Thermal Printer USER'S GUIDE" for details of the recommended barcode scanner and the barcode scanner setting.**

### 1.2.3 Development of Application that Performs Bluetooth Communication or USB Communication with SII Printer

To register the application that communicates with the printer through Bluetooth or USB to App Store, the pre-application to Apple from SII is necessary. For details, please contact SII.

### 1.2.4 Registered Data in Display at Shipping

Registered data in Display at the shipping, such as templates, may be added or changed without prior notice for the quality improvement.
A template which is specified appropriate encode is required to use depending on language settings or character codes to specify. See SII's Website for details about the data to be registered at the shipping.
https://www.sii-ps.com/dspa01/

# Chapter 2
# Product Specifications

This chapter describes the product specifications of the library.

## 2.1 Operating Environment

### 2.1.1 Applicable iOS Devices

Applicable iOS devices for the library are shown in the following list.

(1) RP-F10-x27J1-5 (USB Type-C + Bluetooth + USB host model)

iPhone models

- iPhone 11
- iPhone 11 Pro
- iPhone 11 Pro Max
- iPhone XR
- iPhone XS
- iPhone XS Max
- iPhone X
- iPhone 8
- iPhone 8 Plus
- iPhone 7
- iPhone 7 Plus
- iPhone SE
- iPhone 6s
- iPhone 6s Plus

iPad models

- iPad (7th generation)
- iPad (6th generation)
- 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 Air (3rd generation)
- iPad mini (5th generation)
- iPad mini 4

iPod models

- iPod touch (7th generation)
- iPod touch (6th generation)

(2) RP-F10-x27J1-4 (Bluetooth + USB host model)

iPhone models

- iPhone XR
- iPhone XS
- iPhone XS Max
- iPhone X
- iPhone 8
- iPhone 8 Plus
- iPhone 7
- iPhone 7 Plus
- iPhone SE
- iPhone 6s
- iPhone 6s Plus

iPad models

- iPad Pro 11-inch
- iPad Pro 12.9-inch (3rd generation)
- iPad (6th generation)
- 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

iPod models

- iPod touch (6th generation)

### 2.1.2 Applicable iOS Versions

Applicable iOS versions for the library are shown in the following list.

- iOS 15 to 15.7.8
- iPadOS 15 to 15.7.8
- iOS 16 to 16.7.1
- iPadOS 16 to 16.7.1
- iOS 17 to 17.4
- iPadOS 17 to 17.4

## 2.2 Printer Settings

Set the memory switches of the printer to [Value] in the following table when using the library.
The memory switch of the printer can be changed in the iOS app "SII RP Utility" on the App Store.

See "RP-F10 SERIES THERMAL PRINTER USER'S GUIDE" for details about the memory switches and the factory default settings.

- For Bluetooth connection

| MS | Function | Value |
|---|---|---|
| 5-1 | Automatic Status Response Selection<br>(Auto Status Back) | 0: Enable |
| 5-2 | Initialized Response Selection<br>(Init. Response) | 0: Enable |
| 5-3 | Data Discard Selection When Error Occurs<br>(Error Through) | 0: Enable |
| 5-4 | Data Discard Selection When Output Buffer Full Occurs<br>(Response Data Discarding) | 1: Disable |
| 13-3 | Realtime Command Selection<br>(Realtime Command) | 1: Enable |
| 38-1 | Scanner Automatic Status Response Selection*1<br>(Scanner Auto Status Back) | 0: Enable*2 |
| 39-1 | iOS Auto Connection<br>(Auto Connection) | 1: Disable<br>0: Enable*3 |

*1: The firmware of the printer to support the barcode scanner is Ver.1.10 or later.
*2: Select "Enable" when using the barcode scanner.
See "RP-F10 SERIES THERMAL PRINTER USER'S GUIDE" for details about the combination of peripherals.
*3: Select "Enable" when using **`resetPrinter`**.

- For USB connection

| MS | Function | Value |
|---|---|---|
| 5-1 | Automatic Status Response Selection<br>(Auto Status Back) | 0: Enable |
| 5-2 | Initialized Response Selection<br>(Init. Response) | 0: Enable |
| 5-3 | Data Discard Selection When Error Occurs<br>(Error Through) | 0: Enable |
| 5-4 | Data Discard Selection When Output Buffer Full Occurs<br>(Response Data Discarding) | 1: Disable |
| 13-3 | Realtime Command Selection<br>(Realtime Command) | 1: Enable |
| 38-1 | Scanner Automatic Status Response Selection*1<br>(Scanner Auto Status Back) | 0: Enable*2 |

*1: The firmware of the printer to support the barcode scanner is Ver.1.10 or later.
*2: Select "Enable" when using the barcode scanner.
See "RP-F10 SERIES THERMAL PRINTER USER'S GUIDE" for details about the combination of peripherals.

- For TCP/IP connection

| MS | Function | Value |
|---|---|---|
| 5-1 | Automatic Status Response Selection<br>(Auto Status Back) | 0: Enable |
| 5-2 | Initialized Response Selection<br>(Init. Response) | 0: Enable |
| 5-3 | Data Discard Selection When Error Occurs<br>(Error Through) | 0: Enable |
| 5-4 | Data Discard Selection When Output Buffer Full Occurs<br>(Response Data Discarding) | 1: Disable |
| 13-3 | Realtime Command Selection<br>(Realtime Command) | 1: Enable |
| 38-1 | Scanner Automatic Status Response Selection*1<br>(Scanner Auto Status Back) | 0: Enable*2 |

*1: The firmware of the printer to support the barcode scanner is Ver.1.10 or later.
*2: Select "Enable" when using the barcode scanner.
See "RP-F10 SERIES THERMAL PRINTER USER'S GUIDE" for details about the combination of peripherals.

## 2.3 Precaution

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

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

This library does not support a concurrent connection from multiple apps to one printer when multiple apps are started simultaneously by the Multitasking function for the iPad with iPadOS.

# Chapter 3
# How to Use Library

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

## 3.1 iOS Application Development Environment

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

- Xcode 12.0 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 SII print class library is as follows.

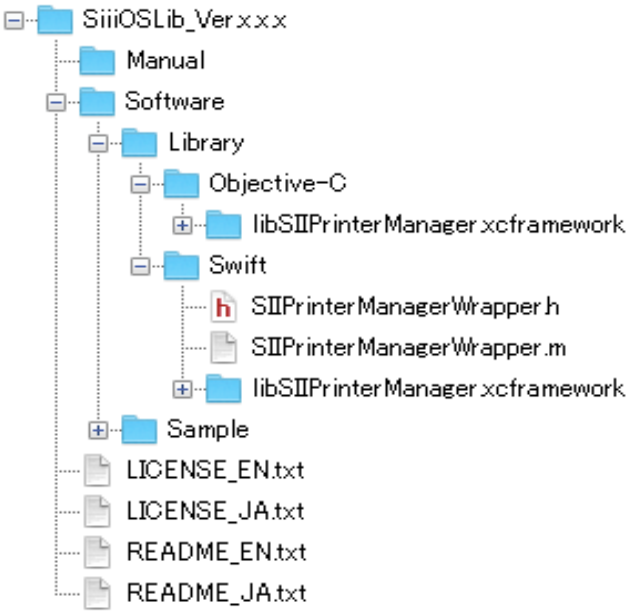



**Figure 3-1**

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 SII print class library as an example, this section describes by development language how to build the library into the project.

See "Chapter 5 Sample Program" for the sample program included in the SII print class library.

| 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.**

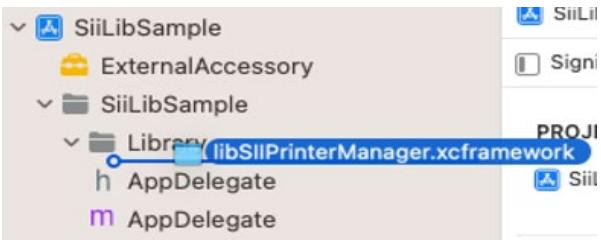
- **libSIIPrinterManager.a**
- **SIIPrinterEnum.h**
- **SIIPrinterException.h**
- **SIIPrinterManager.h**
- **SIISmartLabelManager.h**

### 3.3.1 Objective-C

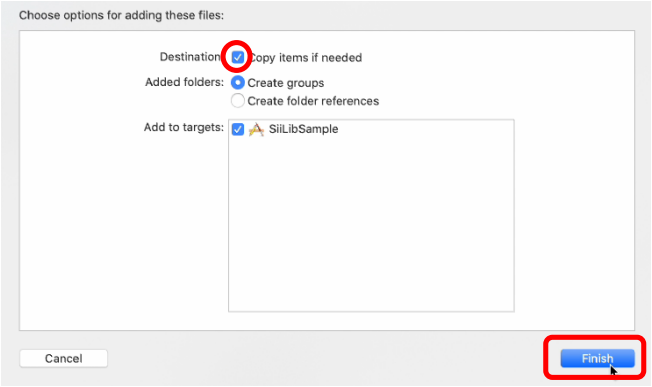
(1) Open the Xcode project.

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

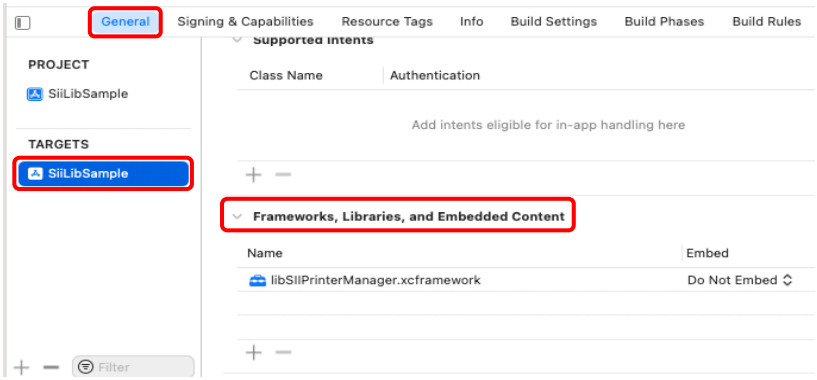
- libSIIPrinterManager.xcframework



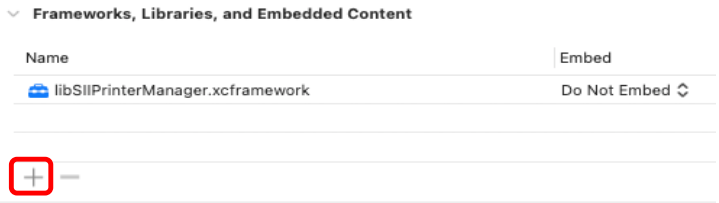

(3) Check the box [Copy items if needed], and click the [Finish] button.



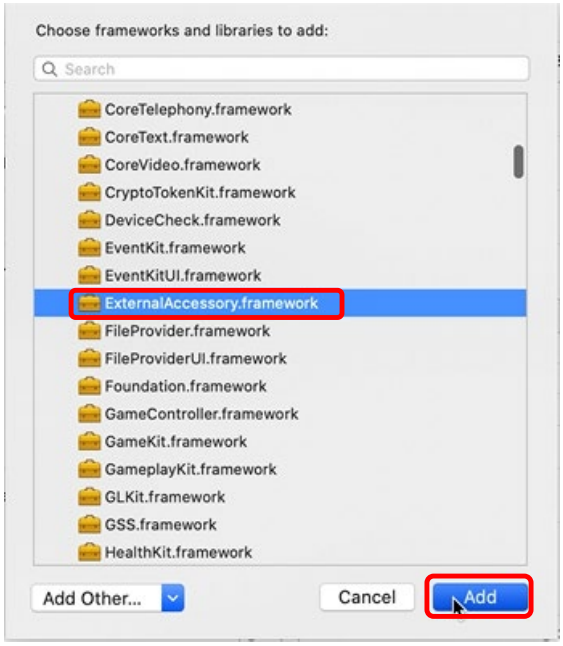
(4) Build the ExternalAccessory.framework into the project.
Select the target project in the [TARGETS], and open the [General] - [Frameworks, Libraries and Embedded Content].

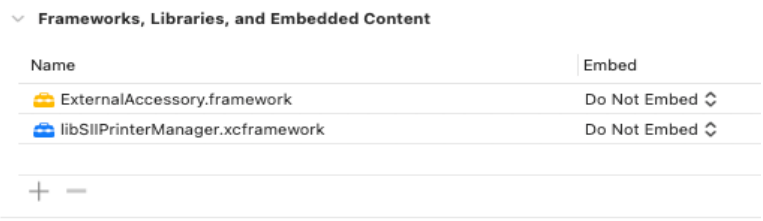



(5) Click the [+] button opened the [Frameworks, Libraries and Embedded Content].

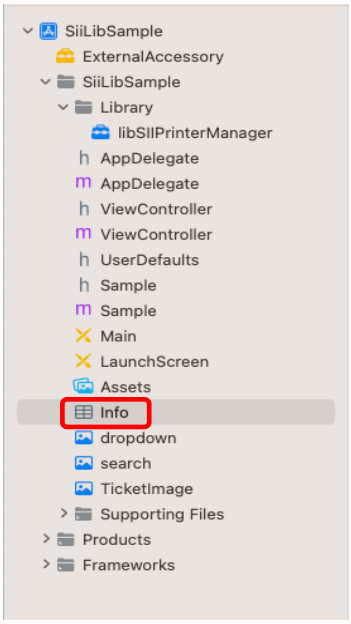



(6) Select the ExternalAccessory.framework from the list and click the [Add] button.

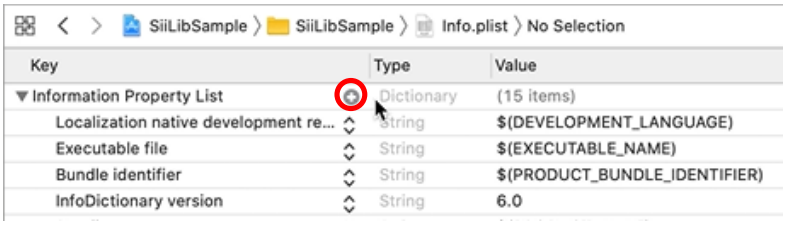


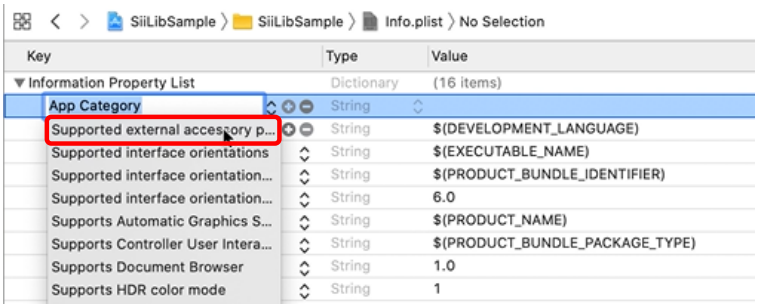
(7) Set the protocol name to use in the ExternalAccessory.framework. Select property list (.plist) in the [Project Navigator].



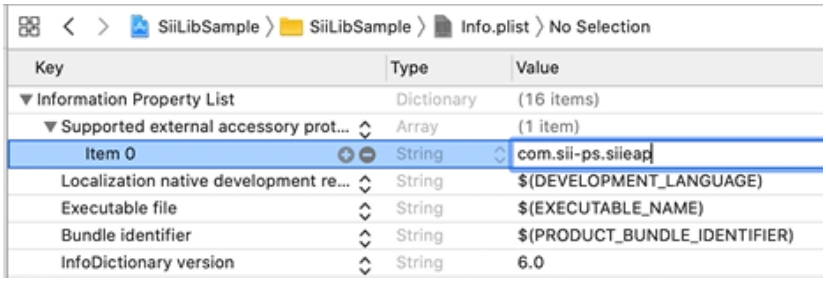

(8) Select the [Information Property List] - ⊕.




(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>

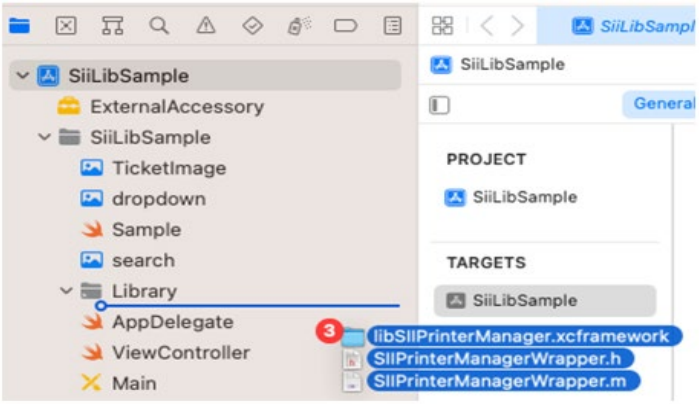
By completing these procedures, the library function becomes available.

### 3.3.2 Swift

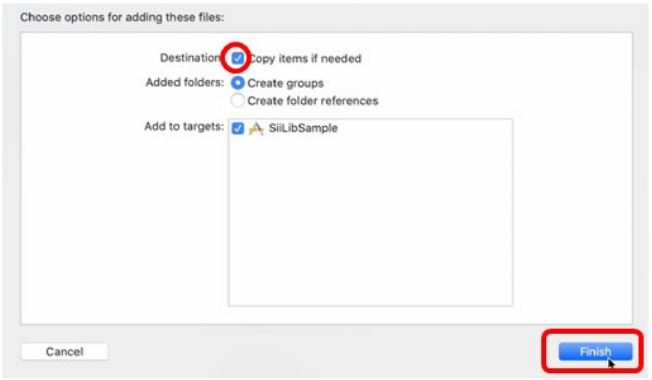
(1) Open the Xcode project.

(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

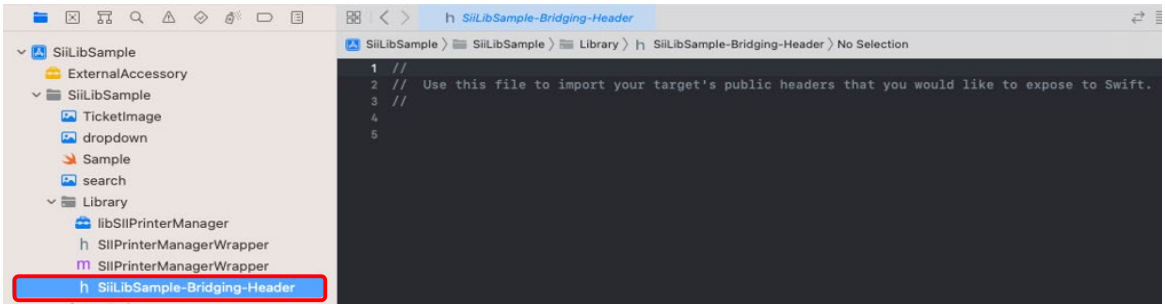



(3) Check the box [Copy items if needed], click the [Finish] button.




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

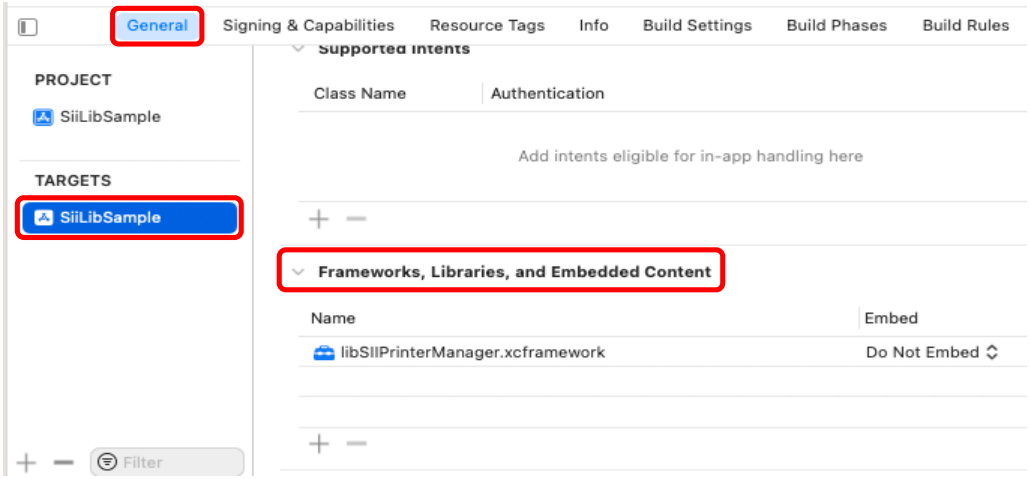
(5) Select the created xxxxxxxx-Bridging-Header.h.



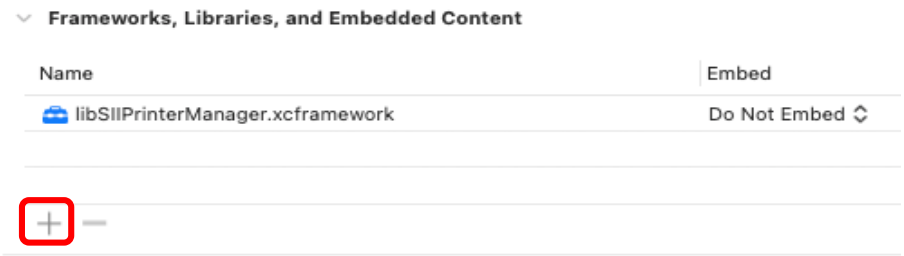

(6) Import the SIIPrinterManager.h and the SIIPrinterManagerWrapper.h into the xxxxxxxx-Bridging-Header.h.

```
//
//  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"

```

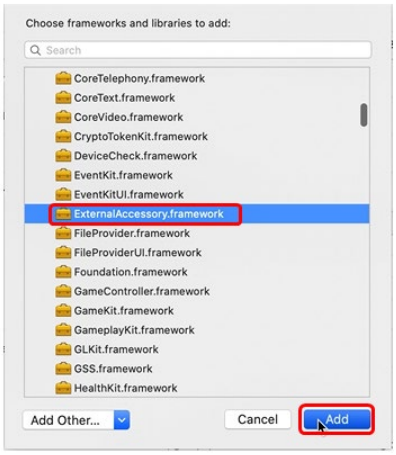
(7) Build ExternalAccessory.framework.
Select the target project in the [TARGETS], and open the [General] - [Frameworks, Libraries and Embedded Content].

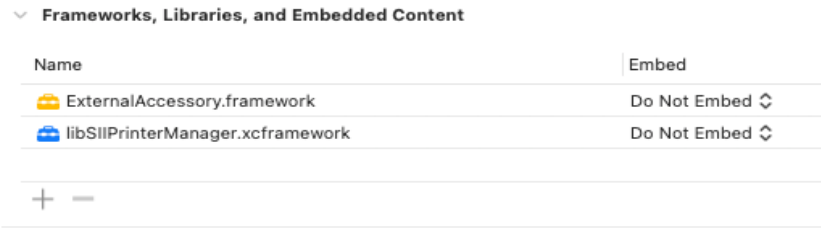



(8) Click the [+] button opened the [Frameworks, Libraries and Embedded Content].

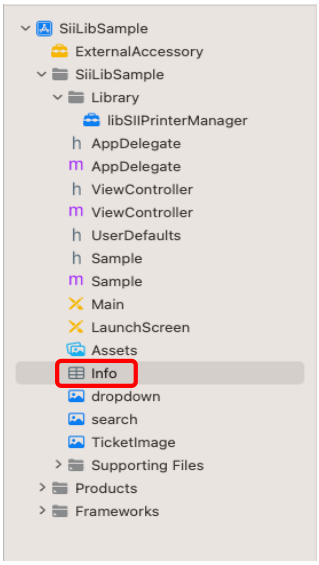


(9) Select the ExternalAccessory.framework from the list and click the [Add] button.

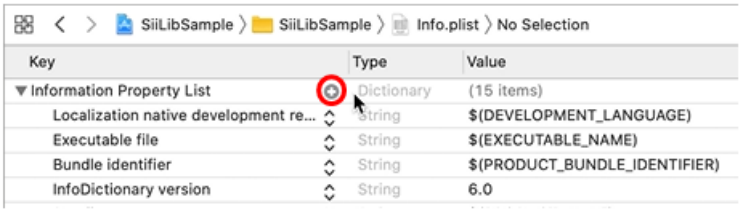


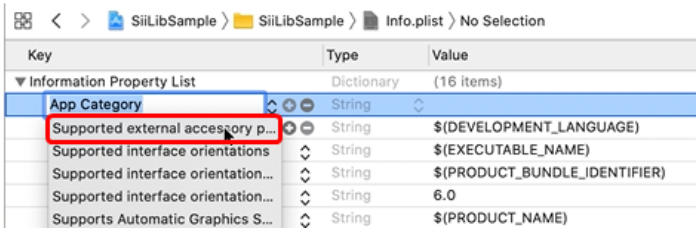

(10) Set the protocol name to use in the ExternalAccessory.framework. Select the property list (.plist) in the [Project Navigator].



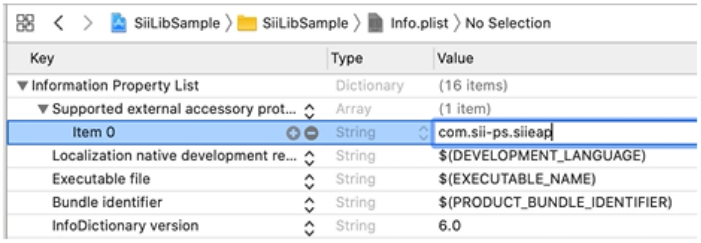

(11) Select the [Information Property List] - ⊕.



(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
# Functions of Library

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

## 4.1 Standard Mode and Page Mode

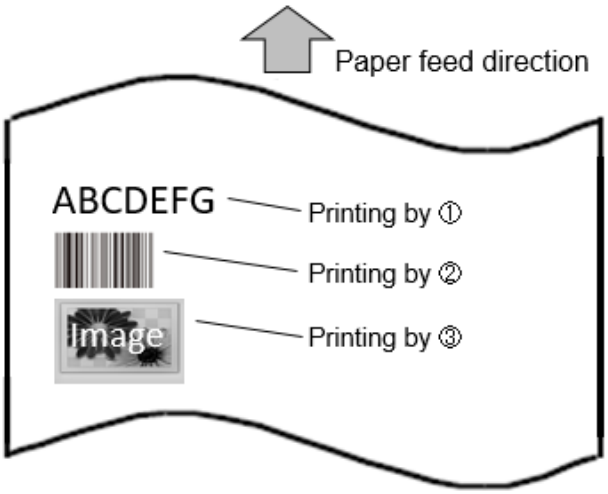
### 4.1.1 Basic Operation

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

(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)



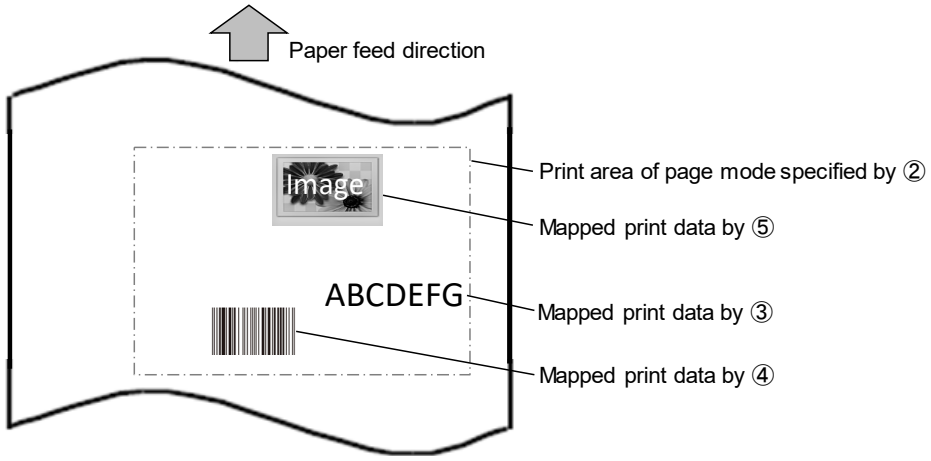

Standard mode suits the printing with an unfixed length such as a receipt.

(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.

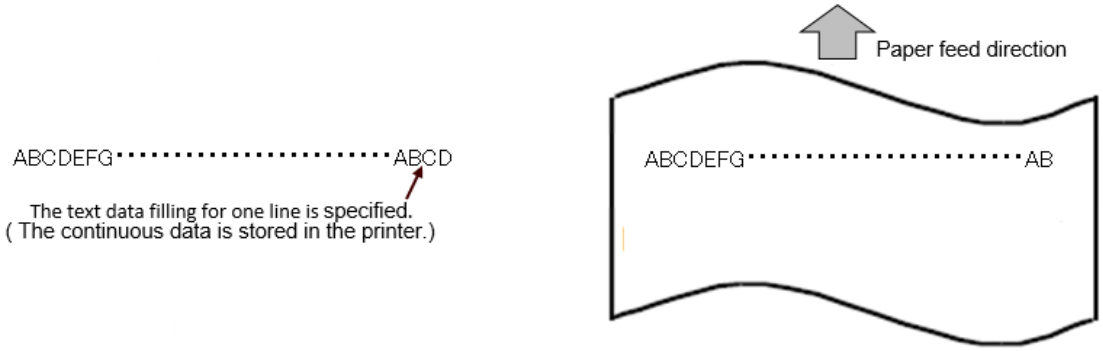
### 4.1.2 Text Data Printing in Standard Mode

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.

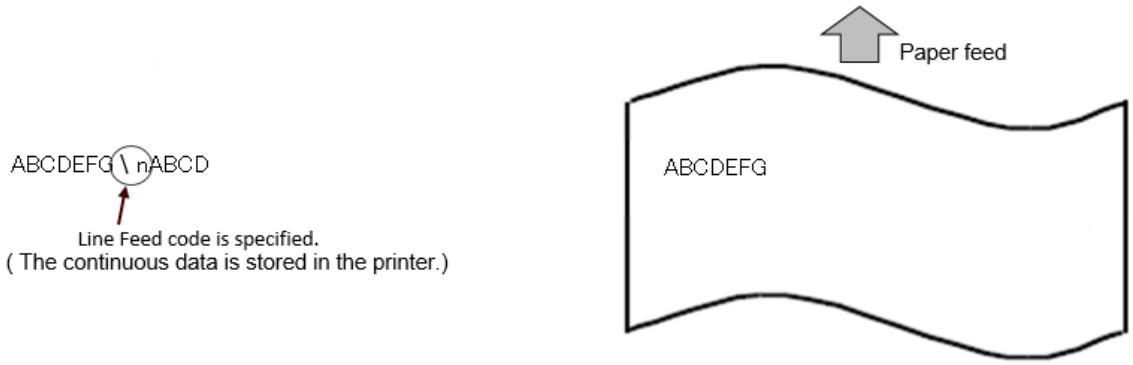
- The text data filling for one line is specified.
- Line Feed code is specified.

- **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**

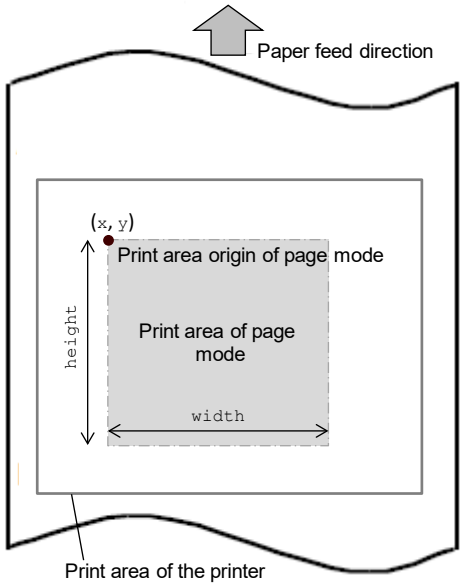
### 4.1.3 Mapping Position of Print Data in Page Mode

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

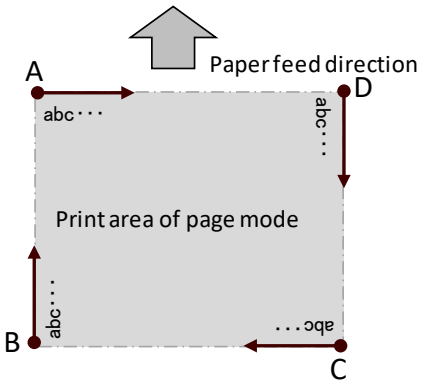
The print area of page mode is specified against the print area of the printer by the print area origin, and the width and the height of page mode. The view of the print area is shown in the following figures.

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




(2) Print direction

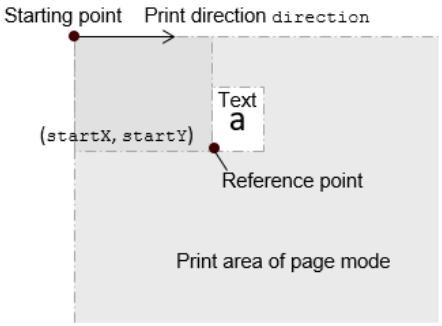
Specify the print direction at setting the print area of page mode.
The starting point is changed depending on specifying the print direction for each direction.
The relation between the print direction and the starting point is shown in the figure below.

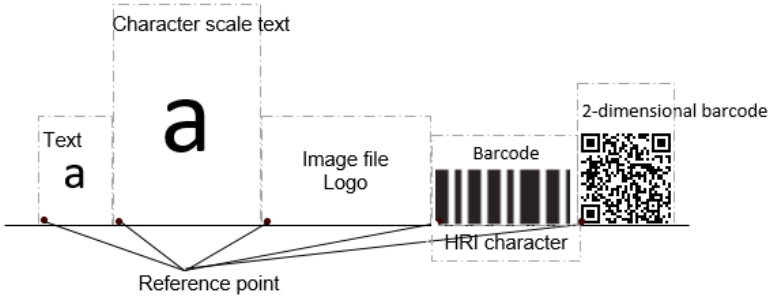



・Starting point: Upper left (A on the figure), Print direction: Left to Right
・Starting point: Left below (B on the figure), Print direction: Below to Upper
・Starting point: Right below (C on the figure), Print direction: Right to Left
・Starting point: Upper right (D on the figure), Print direction: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

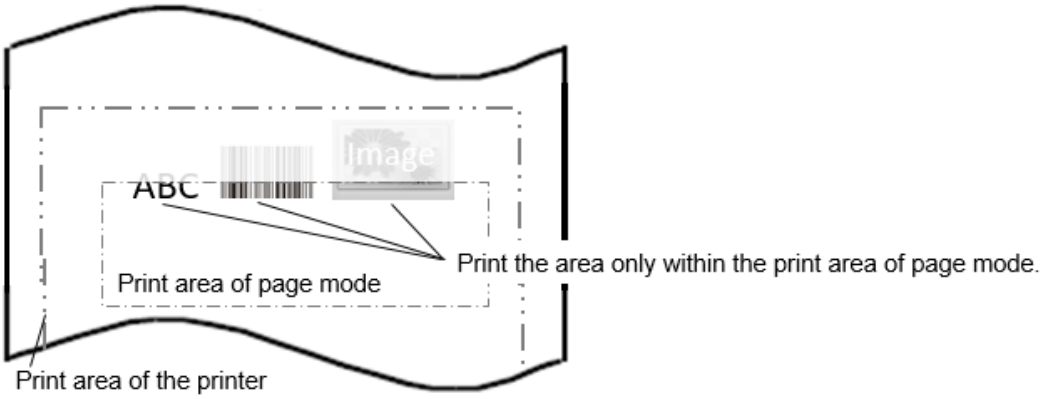
### 4.1.4 Print Data Process at Out of Print Area of Page Mode

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

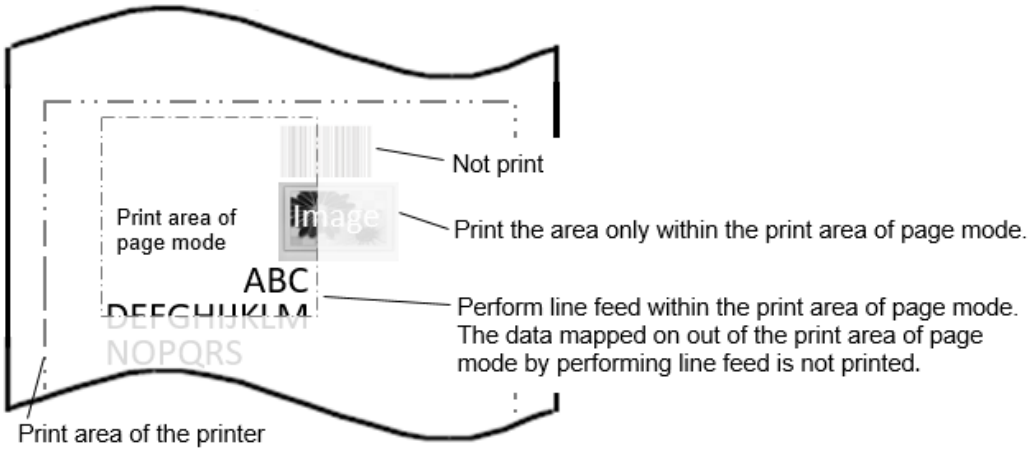
**Type of Print Data**

| **Text** | Barcode,<br>2-dimensional Barcode | Image File, Logo,<br>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 specific directory 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.<br>1 : Records an error log when PrinterException occurs.<br>2 : Records API execution history. |
| LOGSIZEMAX | Log file maximum size | 1MB : Log file maximum size is 1 MB<br>5MB : Log file maximum size is 5 MB<br>10MB : Log file maximum size is 10 MB<br>50MB : Log file maximum size is 50 MB |
| LOGOUTPUT | Console output enabled/disabled | 0 : Console output is disabled<br>1 : Console output is enabled |

### 4.2.3 Log File

Log files are saved as local files in the Android 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 library includes the following classes and protocol.

| Name | Description | Supported*1 |
|---|---|---|
| **`SIIPrinterManager`** | Provides the APIs used for communication with the printer and for printing.<br>See "**4.3.1 `SIIPrinterManager` Class**" for details. | ✓ |
| **`SIIPrinterInfo`** | Stores the printer information found by **`startDiscoveryPrinter`**.<br>See "**4.3.2 `SIIPrinterInfo` Class**" for details. | ✓ |
| **`SIIPrinterException`** | Exception class that is thrown at API call.<br>See "**4.3.3 `SIIPrinterException` Class**" for details. | ✓ |
| **`SIIPrinterManagerDelegate`** | Provides the API to get notice from the printer.<br>See "**4.3.4 `SIIPrinterManagerDelegate` Protocol**" for details. | ✓ |
| **`SIISmartLabelManager`** | Provides the API to specify label files or replace data. | − |

*1: ✓: Supported, - : Not supported in RP-F10

**(NOTE) RP-F10 does not support the APIs relating to label printing function.**

### 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.

| Method | Description |
|---|---|
| Common method to standard mode and page mode | The valid methods in standard mode and page mode. See "4.3.1(1)① 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)② Dedicated method for standard mode" for the methods. |
| Dedicated method for page mode | The valid methods in page mode. See "4.3.1(1)③ Dedicated method for page mode" for the methods. |

① 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)① Common method to standard mode and page mode" for details of the common methods.

| Name | Description | Supported *1 |
|---|---|---|
| **`init`** | Instance | ✓ |
| **`connect`** | Start communicating with printer | ✓ |
| **`disconnect`** | Stop communicating with printer | ✓ |
| **`openDrawer`** | Open cash drawer | ✓ |
| **`buzzer`** | Sound buzzer | - |
| **`externalBuzzer`** | Sound external buzzer | ✓ |
| **`getStatus`** | Get printer status | ✓ |
| **`abort`** | Abort waiting state of printer | ✓ |
| **`registerLogo`** | Register logo | ✓ |
| **`unregisterLogo`** | Delete registered logo | ✓ |
| **`registerStyleSheet`** | Register style sheet | - |
| **`unregisterStyleSheet`** | Delete registered style sheet | - |
| **`resetPrinter`** | Reset printer | ✓ |
| **`getPrinterResponse`** | Get various responses from printer | ✓ |
| **`startDiscoveryPrinter`** | Start printer search (Bluetooth) | ✓ |
| **`startDiscoveryPrinter`** | Start printer search (TCP/IP) | ✓ |
| **`cancelDiscoveryPrinter`** | Cancel printer search | ✓ |
| **`getFoundPrinter`** | Get found printer information | ✓ |
| **`getVersion`** | Get SDK version | ✓ |
| **`controlTransaction`** | Start/End batch processing | ✓ |
| **`defragment`** | Optimize memory area | ✓ |

| Name | Description | Supported *1 |
|---|---|---|
| **initializeMemoryArea** | Initialize memory area | ✓ |
| **showTemplate** | Display template | ✓ |
| **showSlide** | Display slide | ✓ |
| **enterStandbyMode** | Display standby | ✓ |
| **executeMacro** | Execute macro | ✓ |
| **turnOnScreen** | Turn on/off screen | ✓ |
| **selectTemplate** | Select template | ✓ |
| **setTemplateImageData** | Set image data | ✓ |
| **selectTemplateTextObject** | Select text element | ✓ |
| **setTemplateTextAlignment** | Alignment of text data | ✓ |
| **setTemplateTextLeftMargin** | Set left margin of text data | ✓ |
| **setTemplateTextLineSpacing** | Set line spacing of text data | ✓ |
| **setTemplateTextBold** | Set bold character of text data | ✓ |
| **setTemplateTextUnderline** | Set underline of text data | ✓ |
| **setTemplateTextSize** | Set character size of text data | ✓ |
| **setTemplateTextFont** | Set character font of text data | ✓ |
| **setTemplateTextRegisteredFont** | Set registered font of text data | ✓ |
| **setTemplateTextRightSpacing** | Set right space of text data | ✓ |
| **setTemplateTextColor** | Set character color of text data | ✓ |
| **setTemplateTextData** | Input text data | ✓ |
| **setTemplateBarcodeData** | Input barcode data | ✓ |
| **setTemplateQrCodeData** | Input QR Code data | ✓ |
| **registerTemplate** | Register template | ✓ |
| **unregisterTemplate** | Delete template | ✓ |
| **registerImageData** | Register image data | ✓ |
| **unregisterImageData** | Delete image data | ✓ |
| **registerSlideData** | Register slide data | ✓ |
| **unregisterSlideData** | Delete slide data | ✓ |
| **registerUserDefinedCharacter** | Register user-defined character | ✓ |
| **unregisterUserDefinedCharacter** | Delete user-defined character | ✓ |
| **registerOptionFont** | Register optional font | ✓ |
| **unregisterOptionFont** | Delete optional font | ✓ |
| **controlMacroRegistration** | Start/End macro registration | ✓ |
| **getDisplayResponse** | Get various responses from Display | ✓ |

*1: ✓: Supported, - : Not supported in RP-F10

② 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)② Dedicated method for standard mode" for details of the specified methods.

| Name | Description | Supported *1 |
|---|---|---|
| **sendText** | Send text data | ✓ |
| **sendTextEx** | Send format specified text data | ✓ |
| **printBarcode** | Print barcode | ✓ |
| **printPDF417** | Print PDF417 | ✓ |
| **printQRcode** | Print QR Code | ✓ |
| **printDataMatrix** | Print Data Matrix | ✓ |
| **printMaxiCode** | Print MaxiCode | ✓ |
| **printGS1DataBarStacked** | Print GS1 Databar Stacked | ✓ |
| **printGS1DataBarStackedOmnidirectional** | Print GS1 Databar Stacked Omni-directional | ✓ |
| **printGS1DataBarExpandedStacked** | Print GS1 Databar Expanded Stacked | ✓ |
| **printAztecCode** | Print Aztec Code | − |
| **cutPaper** | Cut paper | ✓ |
| **feedPosition** | Paper form feed | − |
| **sendBinary** | Send binary data | ✓ |
| **sendDataFile** | Send specified file | ✓ |
| **printLogo** | Print logo | ✓ |

*1: ✓: Supported, -: Not supported in RP-F10

③ 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)③ Dedicated method for page mode" for details of the specified methods.

| Name | Description | Supported *1 |
|---|---|---|
| **enterPageMode** | Start page mode | ✓ |
| **exitPageMode** | End page mode | ✓ |
| **setPageModeArea** | Specify print area of page mode | ✓ |
| **setPageModeDirection** | Specify print direction of page mode | ✓ |
| **setPageModeLineSpacing** | Specify line spacing of page mode | ✓ |
| **printPageMode** | Print page mode | ✓ |
| **printPageModeText** | Send text data of page mode | ✓ |
| **printPageModeTextEx** | Send format specified text data of page mode | ✓ |
| **printPageModeBarcode** | Print barcode of page mode | ✓ |
| **printPageModePDF417** | Print PDF417 of page mode | ✓ |
| **printPageModeQRcode** | Print QR Code of page mode | ✓ |
| **printPageModeDataMatrix** | Print Data Matrix of page mode | ✓ |
| **printPageModeMaxiCode** | Print MaxiCode of page mode | ✓ |

| Name | Description | Supported *1 |
|---|---|---|
| **printPageModeGS1DataBarStacked** | Print GS1 Databar Stacked of page mode | ✓ |
| **printPageModeGS1DataBarStackedOmnidirectional** | Print GS1 Databar Stacked Omni-directional of page mode | ✓ |
| **printPageModeGS1DataBarExpandedStacked** | Print GS1 Databar Expanded Stacked of page mode | ✓ |
| **printPageModeAztecCode** | Print Aztec Code of page mode | − |
| **sendPageModeBinary** | Send binary data of page mode | ✓ |
| **printPageModeImageFile** | Draw image file of page mode | ✓ |
| **printPageModeRectangle** | Draw rectangle image of page mode | ✓ |
| **printPageModeLine** | Print ruled line of page mode | ✓ |
| **printPageModeLogo** | Print logo of page mode | ✓ |

*1: ✓: Supported, -: Not supported in RP-F10

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

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

| Name | Access | Description | Supported*1 |
|---|---|---|---|
| **sendTimeout** | R/W | Get/Set send timeout period | ✓ |
| **receiveTimeout** | R/W | Get/Set receive timeout period | ✓ |
| **internationalCharacter** | R/W | Get/Set international character set | ✓ |
| **codePage** | R/W | Get/Set codepage | ✓ |
| **printerModel** | R | Get printer model | ✓ |
| **portType** | R | Get connecting port type | ✓ |
| **isConnect** | R | Verify connection state with printer | ✓ |
| **socketKeepingTime** | R/W | Get/Set socket keeping time | ✓ |
| **delegate** | R/W | Register delegate | ✓ |

*1: ✓: Supported, - : Not supported in RP-F10

(3) Constant List

① Printer model

Constants used for starting communication with the printer and getting the printer model are shown in the following table.

| Constant Name | Description | Value |
|---|---|---|
| **SII_PM_PRINTER_MODEL_RP_F10** | RP-F10 | 301 |

② Port type

Constants used for starting communication with the printer and getting the connection port type are shown in the following table.

| Constant Name | Description | Value |
|---|---|---|
| **SII_PM_PRINTER_PORT_TYPE_BLUETOOTH** | Bluetooth | 0 |
| **SII_PM_PRINTER_PORT_TYPE_USB** | USB | 1 |
| **SII_PM_PRINTER_PORT_TYPE_TCP** | TCP/IP | 2 |

③ Printer response type

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

| Constant Name | Description | Value |
|---|---|---|
| **SII_PM_PRINTER_RESPONSE_REQUEST** | Execution response request | 0 |
| **SII_PM_PRINTER_RESPONSE_USER_AREA** | Send remaining capacity of user area | 1 |
| **SII_PM_PRINTER_RESPONSE_ARRANGE_USER_AREA** | Send remaining capacity of user area after defragment | 2 |
| **SII_PM_PRINTER_RESPONSE_NV_GRAPHICS** | Send NV graphics memory capacity | 3 |
| **SII_PM_PRINTER_RESPONSE_KEY_CODE** | Send key code list of defined NV graphics | 4 |
| **SII_PM_PRINTER_RESPONSE_FIRMWARE_VERSION** | Send firmware version | 6 |

④ Display response type

Constants used for getting various responses from Display are shown in the following table.

| Constant Name | Description | Value |
|---|---|---|
| **SII_PM_PRINTER_RESPONSE_REQUEST** | Execution response request | 0 |
| **SII_PM_PRINTER_RESPONSE_USER_AREA** | Send remaining capacity of user area | 1 |
| **SII_PM_DISPLAY_RESPONSE_TEMPLATE_ID_LIST** | Send template ID | 2 |
| **SII_PM_DISPLAY_RESPONSE_IMAGE_ID_LIST** | Send image ID | 3 |
| **SII_PM_DISPLAY_RESPONSE_SLIDE_ID_LIST** | Send slide ID | 4 |
| **SII_PM_DISPLAY_RESPONSE_TEMPLATE_LABEL** | Send template name | 5 |
| **SII_PM_DISPLAY_RESPONSE_IMAGE_LABEL** | Send image name | 6 |
| **SII_PM_DISPLAY_RESPONSE_SLIDE_LABEL** | Send slide name | 7 |

⑤ International character set

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

| Constant Name | Description | Value |
|---|---|---|
| **SII_PM_COUNTRY_USA** | USA | 0 |
| **SII_PM_COUNTRY_FRANCE** | France | 1 |
| **SII_PM_COUNTRY_GERMANY** | Germany | 2 |
| **SII_PM_COUNTRY_ENGLAND** | United Kingdom | 3 |
| **SII_PM_COUNTRY_DENMARK_1** | Denmark I | 4 |
| **SII_PM_COUNTRY_SWEDEN** | Sweden | 5 |
| **SII_PM_COUNTRY_ITALY** | Italy | 6 |
| **SII_PM_COUNTRY_SPAIN** | Spain I | 7 |
| **SII_PM_COUNTRY_JAPAN** | Japan | 8 |
| **SII_PM_COUNTRY_NORWAY** | Norway | 9 |
| **SII_PM_COUNTRY_DENMARK_2** | Denmark II | 10 |
| **SII_PM_COUNTRY_SPAIN_2** | Spain II | 11 |
| **SII_PM_COUNTRY_LATIN_AMERICA** | Latin America | 12 |
| **SII_PM_COUNTRY_ARABIA** | Arabia | 17 |

⑥ Codepage

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

| Constant Name | Description | Value |
|---|---|---|
| **SII_PM_CODE_PAGE_437** | USA, Standard Europe (Code Page437) | 0 |
| **SII_PM_CODE_PAGE_KATAKANA** | Katakana | 1 |
| **SII_PM_CODE_PAGE_850** | Multilingual (Code Page850) | 2 |
| **SII_PM_CODE_PAGE_860** | Portuguese (Code Page860) | 3 |
| **SII_PM_CODE_PAGE_863** | Canadian-French (Code page863) | 4 |
| **SII_PM_CODE_PAGE_865** | Nordic (Code Page865) | 5 |
| **SII_PM_CODE_PAGE_857***1 | Turkish (Code Page857) | 13 |
| **SII_PM_CODE_PAGE_737** | Greek (Code Page737) | 14 |
| **SII_PM_CODE_PAGE_1252** | Latin (Code Page1252) | 16 |
| **SII_PM_CODE_PAGE_866** | Russian (Code Page866) | 17 |
| **SII_PM_CODE_PAGE_852** | Eastern Europe (CodePage 852) | 18 |
| **SII_PM_CODE_PAGE_858** | Euro (Code Page858) | 19 |
| **SII_PM_CODE_PAGE_855** | Cyrillic (Code Page855) | 34 |
| **SII_PM_CODE_PAGE_864***1*2 | Arabic (Code Page864) | 37 |
| **SII_PM_CODE_PAGE_1250** | Central European (Code Page1250) | 45 |
| **SII_PM_CODE_PAGE_1251** | Cyrillic (Code Page1251) | 46 |
| **SII_PM_CODE_PAGE_1253***3 | Greek (Code Page1253) | 47 |
| **SII_PM_CODE_PAGE_1254** | Turkish (Code Page1254) | 48 |

*1: 20ACh of the Unicode cannot be printed or displayed.
*2: Font B cannot be printed or displayed.
*3: 00AAh of the Unicode cannot be printed or displayed.

⑦ Barcode and PDF417

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

| Constant Name | Description | Value |
|---|---|---|
| **SII_PM_BARCODE_HEIGHT_DEFAULT** | Default value of barcode height | 162 |
| **SII_PM_PDF417_MODULE_HEIGHT_DEFAULT** | Default value of PDF417 height | 10 |
| **SII_PM_PDF417_ROW_AUTO** | Automatic selection of the number of rows | 0 |
| **SII_PM_PDF417_COLUMN_AUTO** | Automatic selection of the number of columns | 0 |

(4) Enumerated Constant List

① Dithering (`Dithering`)

Constants of enumerated type used for dithering are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_DITHERING_DISABLE** | Dithering is disabled |
| **SII_PM_DITHERING_ERRORDIFFUSION** | Dithering is enabled |

② Batch processing selection (`TransactionFunction`)

Constants of enumerated type used for batch processing selection are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_TRANSACTION_CLEAR** | Cancel batch processing |
| **SII_PM_TRANSACTION_START** | Start batch processing |
| **SII_PM_TRANSACTION_PRINT** | Finish batch printing and batch processing |

③ Bold print (`CharacterBold`)

Constants of enumerated type used for bold character are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_BOLD_CANCEL** | Cancel bold print |
| **SII_PM_BOLD** | Specify bold print |

④ Underline (`CharacterUnderline`)

Constants of enumerated type used for underline are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_UNDERLINE_CANCEL** | Cancel underline print |
| **SII_PM_UNDERLINE_1** | Specify 1-dot width underline print |
| **SII_PM_UNDERLINE_2***1 | Specify 2-dot width underline print |

*1: Supported only the printer.

⑤ Reverse print (`CharacterReverse`)

Constants of enumerated type used for reverse print are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_REVERSE_CANCEL** | Cancel reverse print |
| **SII_PM_REVERSE** | Specify reverse print |

⑥ Inversion print (`CharacterInversion`)

Constants of enumerated type 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.

| Constant Name | Description |
|---|---|
| **SII_PM_INVERSION_CANCEL** | Cancel inversion print |
| **SII_PM_INVERSION** | Specify inversion print |

⑦ Character font (`CharacterFont`)

Constants of enumerated type used for character fonts are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_FONT_A** | Font A (24 × 12) |
| **SII_PM_FONT_B** | Font B (16 × 8) |

⑧ Character scale (`CharacterScale`)

Constants of enumerated type used for character scale are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_VARTICAL_1_HORIZONTAL_1** | Height × 1 and width × 1 |
| **SII_PM_VARTICAL_1_HORIZONTAL_2** | Height × 1 and width × 2 |
| **SII_PM_VARTICAL_1_HORIZONTAL_3** | Height × 1 and width × 3 |
| **SII_PM_VARTICAL_1_HORIZONTAL_4** | Height × 1 and width × 4 |
| **SII_PM_VARTICAL_2_HORIZONTAL_1** | Height × 2 and width × 1 |
| **SII_PM_VARTICAL_2_HORIZONTAL_2** | Height × 2 and width × 2 |
| **SII_PM_VARTICAL_2_HORIZONTAL_3** | Height × 2 and width × 3 |
| **SII_PM_VARTICAL_2_HORIZONTAL_4** | Height × 2 and width × 4 |
| **SII_PM_VARTICAL_2_HORIZONTAL_6**[*1] | Height × 2 and width × 6 |
| **SII_PM_VARTICAL_3_HORIZONTAL_1** | Height × 3 and width × 1 |
| **SII_PM_VARTICAL_3_HORIZONTAL_2** | Height × 3 and width × 2 |

| Constant Name | Description |
|---|---|
| **SII_PM_VARTICAL_3_HORIZONTAL_3** | Height × 3 and width × 3 |
| **SII_PM_VARTICAL_3_HORIZONTAL_4** | Height × 3 and width × 4 |
| **SII_PM_VARTICAL_4_HORIZONTAL_1** | Height × 4 and width × 1 |
| **SII_PM_VARTICAL_4_HORIZONTAL_2** | Height × 4 and width × 2 |
| **SII_PM_VARTICAL_4_HORIZONTAL_3** | Height × 4 and width × 3 |
| **SII_PM_VARTICAL_4_HORIZONTAL_4** | Height × 4 and width × 4 |
| **SII_PM_VARTICAL_4_HORIZONTAL_6**[*1] | Height × 4 and width × 6 |
| **SII_PM_VARTICAL_4_HORIZONTAL_8**[*1] | Height × 4 and width × 8 |
| **SII_PM_VARTICAL_6_HORIZONTAL_2**[*1] | Height × 6 and width × 2 |
| **SII_PM_VARTICAL_6_HORIZONTAL_4**[*1] | Height × 6 and width × 4 |
| **SII_PM_VARTICAL_6_HORIZONTAL_6**[*1] | Height × 6 and width × 6 |
| **SII_PM_VARTICAL_6_HORIZONTAL_8**[*1] | Height × 6 and width × 8 |
| **SII_PM_VARTICAL_8_HORIZONTAL_4**[*1] | Height × 8 and width × 4 |
| **SII_PM_VARTICAL_8_HORIZONTAL_6**[*1] | Height × 8 and width × 6 |
| **SII_PM_VARTICAL_8_HORIZONTAL_8**[*1] | Height × 8 and width × 8 |

*1: Supported only the printer.

⑨ Alignment (`PrintAlignment`)

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

| Constant Name | Description |
|---|---|
| **SII_PM_ALIGNMENT_LEFT** | Aligned left |
| **SII_PM_ALIGNMENT_CENTER** | Centered |
| **SII_PM_ALIGNMENT_RIGHT** | Aligned right |

⑩ Barcode symbol (`BarcodeSymbol`)

Constants of enumerated type used for barcode symbols are shown in the following table.

| Constant Name | Description | Syntax[*1] |
|---|---|---|
| **SII_PM_BARCODE_UPC_A** | UPC-A | (a) |
| **SII_PM_BARCODE_UPC_E** | UPC-E | (a) |
| **SII_PM_BARCODE_EAN13** | EAN13 | (a) |
| **SII_PM_BARCODE_JAN13** | JAN13 | (a) |
| **SII_PM_BARCODE_EAN8** | EAN8 | (a) |
| **SII_PM_BARCODE_JAN8** | JAN8 | (a) |

| Constant Name | Description | Syntax*1 |
|---|---|---|
| **SII_PM_BARCODE_CODE39** | CODE39 | (a), (b) |
| **SII_PM_BARCODE_CODE93** | CODE93 | (c) |
| **SII_PM_BARCODE_CODE128** | CODE128 | (c) |
| **SII_PM_BARCODE_ITF** | ITF | (a), (b) |
| **SII_PM_BARCODE_CODABAR** | CODABAR | (a), (b) |
| **SII_PM_BARCODE_EAN13_ADDON** | EAN13 add-on | (a) |
| **SII_PM_BARCODE_JAN13_ADDON** | JAN13 add-on | (a) |
| **SII_PM_BARCODE_GS1_OMNI_DIRECTIONAL** | GS1 Databar Omni-directional | (a) |
| **SII_PM_BARCODE_GS1_TRUNCATED** | GS1 Databar Truncated | (a) |
| **SII_PM_BARCODE_GS1_LIMITED** | GS1 Databar Limited | (a) |
| **SII_PM_BARCODE_GS1_EXPANDED** | GS1 Databar Expanded | (a) |

*1: See **printBarcode** or **printPageModeBarcode** for details of syntax.

⑪ Module size (`ModuleSize`)

Constants of enumerated type used for width, nominal fine element width, and module size of barcode are shown in the following table.

| Constant Name | Description | Method to Use |
|---|---|---|
| **SII_PM_BARCODE_MODULE_WIDTH_2** | Fine element 2 dots<br>Module width 0.250 mm | ● **printBarcode**<br>● **printPageModeBarcode** |
| **SII_PM_BARCODE_MODULE_WIDTH_3** | Fine element 3 dots<br>Module width 0.375 mm | |
| **SII_PM_BARCODE_MODULE_WIDTH_4** | Fine element 4 dots<br>Module width 0.500 mm | |
| **SII_PM_BARCODE_MODULE_WIDTH_5** | Fine element 5 dots<br>Module width 0.625 mm | |
| **SII_PM_BARCODE_MODULE_WIDTH_6** | Fine element 6 dots<br>Module width 0.750 mm | |
| **SII_PM_PDF417_MODULE_WIDTH_2** | Nominal fine element width 2 dots | ● **printPDF417**<br>● **printPageModePDF417** |
| **SII_PM_PDF417_MODULE_WIDTH_3** | Nominal fine element width 3 dots | |
| **SII_PM_PDF417_MODULE_WIDTH_4** | Nominal fine element width 4 dots | |
| **SII_PM_PDF417_MODULE_WIDTH_5** | Nominal fine element width 5 dots | |
| **SII_PM_PDF417_MODULE_WIDTH_6** | Nominal fine element width 6 dots | |
| **SII_PM_PDF417_MODULE_WIDTH_7** | Nominal fine element width 7 dots | |
| **SII_PM_PDF417_MODULE_WIDTH_8** | Nominal fine element width 8 dots | |

| Constant Name | Description | Method to Use |
|---|---|---|
| **SII_PM_QR_MODULE_SIZE_2** | 2 dots | ● **printQRcode**<br>● **setTemplateQrCodeData**<br>● **printPageModeQRcode** |
| **SII_PM_QR_MODULE_SIZE_3** | 3 dots | |
| **SII_PM_QR_MODULE_SIZE_4** | 4 dots | |
| **SII_PM_QR_MODULE_SIZE_5** | 5 dots | |
| **SII_PM_QR_MODULE_SIZE_6** | 6 dots | |
| **SII_PM_QR_MODULE_SIZE_7** | 7 dots | |
| **SII_PM_QR_MODULE_SIZE_8** | 8 dots | |
| **SII_PM_QR_MODULE_SIZE_9** | 9 dots | |
| **SII_PM_QR_MODULE_SIZE_10** | 10 dots | |
| **SII_PM_QR_MODULE_SIZE_11** | 11 dots | |
| **SII_PM_QR_MODULE_SIZE_12** | 12 dots | |
| **SII_PM_QR_MODULE_SIZE_13** | 13 dots | |
| **SII_PM_QR_MODULE_SIZE_14** | 14 dots | |
| **SII_PM_QR_MODULE_SIZE_15** | 15 dots | |
| **SII_PM_QR_MODULE_SIZE_16** | 16 dots | |
| **SII_PM_DATAMATRIX_MODULE_SIZE_2** | 2 dots | ● **printDataMatrix**<br>● **printPageModeDataMatrix** |
| **SII_PM_DATAMATRIX_MODULE_SIZE_3** | 3 dots | |
| **SII_PM_DATAMATRIX_MODULE_SIZE_4** | 4 dots | |
| **SII_PM_DATAMATRIX_MODULE_SIZE_5** | 5 dots | |
| **SII_PM_DATAMATRIX_MODULE_SIZE_6** | 6 dots | |
| **SII_PM_DATAMATRIX_MODULE_SIZE_7** | 7 dots | |
| **SII_PM_DATAMATRIX_MODULE_SIZE_8** | 8 dots | |
| **SII_PM_DATAMATRIX_MODULE_SIZE_9** | 9 dots | |
| **SII_PM_DATAMATRIX_MODULE_SIZE_10** | 10 dots | |
| **SII_PM_DATAMATRIX_MODULE_SIZE_11** | 11 dots | |
| **SII_PM_DATAMATRIX_MODULE_SIZE_12** | 12 dots | |
| **SII_PM_DATAMATRIX_MODULE_SIZE_13** | 13 dots | |
| **SII_PM_DATAMATRIX_MODULE_SIZE_14** | 14 dots | |
| **SII_PM_DATAMATRIX_MODULE_SIZE_15** | 15 dots | |
| **SII_PM_DATAMATRIX_MODULE_SIZE_16** | 16 dots | |

| Constant Name | Description | Method to Use |
|---|---|---|
| **SII_PM_GS1DATABAR_MODULE_SIZE_2** | 2 dots | ● **printGS1DataBarStacked**<br>● **printGS1DataBarStackedOmnidirectional**<br>● **printGS1DataBarExpandedStacked**<br>● **printPageModeGS1DataBarStacked**<br>● **printPageModeGS1DataBarStackedOmnidirectional**<br>● **printPageModeGS1DataBarExpandedStacked** |
| **SII_PM_GS1DATABAR_MODULE_SIZE_3** | 3 dots | |
| **SII_PM_GS1DATABAR_MODULE_SIZE_4** | 4 dots | |
| **SII_PM_GS1DATABAR_MODULE_SIZE_5** | 5 dots | |
| **SII_PM_GS1DATABAR_MODULE_SIZE_6** | 6 dots | |
| **SII_PM_GS1DATABAR_MODULE_SIZE_7** | 7 dots | |
| **SII_PM_GS1DATABAR_MODULE_SIZE_8** | 8 dots | |
| **SII_PM_GS1DATABAR_MODULE_SIZE_9** | 9 dots | |
| **SII_PM_GS1DATABAR_MODULE_SIZE_10** | 10 dots | |
| **SII_PM_GS1DATABAR_MODULE_SIZE_11** | 11 dots | |
| **SII_PM_GS1DATABAR_MODULE_SIZE_12** | 12 dots | |
| **SII_PM_GS1DATABAR_MODULE_SIZE_13** | 13 dots | |
| **SII_PM_GS1DATABAR_MODULE_SIZE_14** | 14 dots | |
| **SII_PM_GS1DATABAR_MODULE_SIZE_15** | 15 dots | |
| **SII_PM_GS1DATABAR_MODULE_SIZE_16** | 16 dots | |

⑫ HRI character print position (`HriPosition`)

Constants of enumerated type used for HRI character print position are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_HRI_NONE** | Not printed |
| **SII_PM_HRI_POSITION_ABOVE** | Above barcode |
| **SII_PM_HRI_POSITION_BELOW** | Below barcode |
| **SII_PM_HRI_POSITION_ABOVE_BELOW** | Above and below barcode (both) |

⑬ N:W ratio (`NwRatio`)

Constants of enumerated type used for N:W ratio are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_NWRATIO_1TO2** | 1:2 |
| **SII_PM_NWRATIO_1TO2_5** | 1:2.5 |
| **SII_PM_NWRATIO_1TO3** | 1:3 |

⑭ Error correction level (`ErrorCorrection`)

Constants of enumerated type used for error correction level are shown in the following table.

| Constant Name | Description | Method to Use |
|---|---|---|
| **SII_PM_PDF417_ERROR_CORRECTION_0** | Error correction level 0 | ● **printPDF417**<br>● **printPageModePDF417** |
| **SII_PM_PDF417_ERROR_CORRECTION_1** | Error correction level 1 | |
| **SII_PM_PDF417_ERROR_CORRECTION_2** | Error correction level 2 | |
| **SII_PM_PDF417_ERROR_CORRECTION_3** | Error correction level 3 | |
| **SII_PM_PDF417_ERROR_CORRECTION_4** | Error correction level 4 | |
| **SII_PM_PDF417_ERROR_CORRECTION_5** | Error correction level 5 | |
| **SII_PM_PDF417_ERROR_CORRECTION_6** | Error correction level 6 | |
| **SII_PM_PDF417_ERROR_CORRECTION_7** | Error correction level 7 | |
| **SII_PM_PDF417_ERROR_CORRECTION_8** | Error correction level 8 | |
| **SII_PM_QR_ERROR_CORRECTION_L** | Error correction level L | ● **printQRcode**<br>● **setTemplateQrCodeData**<br>● **printPageModeQRcode** |
| **SII_PM_QR_ERROR_CORRECTION_M** | Error correction level M | |
| **SII_PM_QR_ERROR_CORRECTION_H** | Error correction level H | |
| **SII_PM_QR_ERROR_CORRECTION_Q** | Error correction level Q | |

⑮ PDF417 symbol (`Pdf417Symbol`)

Constants of enumerated type used for PDF417 symbols are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_PDF417_STANDARD** | PDF417 |
| **SII_PM_PDF417_COMPACT** | Compact PDF417 |

⑯ QR Code Model (`QrModel`)

Constants of enumerated type used for QR Code Model are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_QR_MODEL_2** | QR Code Model 2 |

⑰ Data Matrix module (`DataMatrixModule`)

Constants of enumerated type used for Data Matrix module are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_DATA_MATRIX_AUTO** | Number of modules: Automatic |
| **SII_PM_DATA_MATRIX_10_10** | Number of modules: 10 × 10 |
| **SII_PM_DATA_MATRIX_12_12** | Number of modules: 12 × 12 |
| **SII_PM_DATA_MATRIX_14_14** | Number of modules: 14 × 14 |
| **SII_PM_DATA_MATRIX_16_16** | Number of modules: 16 × 16 |
| **SII_PM_DATA_MATRIX_18_18** | Number of modules: 18 × 18 |
| **SII_PM_DATA_MATRIX_20_20** | Number of modules: 20 × 20 |
| **SII_PM_DATA_MATRIX_22_22** | Number of modules: 22 × 22 |
| **SII_PM_DATA_MATRIX_24_24** | Number of modules: 24 × 24 |
| **SII_PM_DATA_MATRIX_26_26** | Number of modules: 26 × 26 |
| **SII_PM_DATA_MATRIX_32_32** | Number of modules: 32 × 32 |
| **SII_PM_DATA_MATRIX_36_36** | Number of modules: 36 × 36 |
| **SII_PM_DATA_MATRIX_40_40** | Number of modules: 40 × 40 |
| **SII_PM_DATA_MATRIX_44_44** | Number of modules: 44 × 44 |
| **SII_PM_DATA_MATRIX_48_48** | Number of modules: 48 × 48 |
| **SII_PM_DATA_MATRIX_52_52** | Number of modules: 52 × 52 |
| **SII_PM_DATA_MATRIX_64_64** | Number of modules: 64 × 64 |
| **SII_PM_DATA_MATRIX_72_72** | Number of modules: 72 × 72 |
| **SII_PM_DATA_MATRIX_80_80** | Number of modules: 80 × 80 |
| **SII_PM_DATA_MATRIX_88_88** | Number of modules: 88 × 88 |
| **SII_PM_DATA_MATRIX_96_96** | Number of modules: 96 × 96 |
| **SII_PM_DATA_MATRIX_104_104** | Number of modules: 104 × 104 |
| **SII_PM_DATA_MATRIX_120_120** | Number of modules: 120 × 120 |
| **SII_PM_DATA_MATRIX_132_132** | Number of modules: 132 × 132 |
| **SII_PM_DATA_MATRIX_144_144** | Number of modules: 144 × 144 |
| **SII_PM_DATA_MATRIX_8_18** | Number of modules: 8 × 18 |
| **SII_PM_DATA_MATRIX_8_32** | Number of modules: 8 × 32 |
| **SII_PM_DATA_MATRIX_12_26** | Number of modules: 12 × 26 |
| **SII_PM_DATA_MATRIX_12_36** | Number of modules: 12 × 36 |
| **SII_PM_DATA_MATRIX_16_36** | Number of modules: 16 × 36 |
| **SII_PM_DATA_MATRIX_16_48** | Number of modules: 16 × 48 |

⑱ MaxiCode Mode (`MaxiCodeMode`)

Constants of enumerated type used for MaxiCode Mode are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_MAXI_CODE_2** | Mode2 |
| **SII_PM_MAXI_CODE_3** | Mode3 |
| **SII_PM_MAXI_CODE_4** | Mode4 |
| **SII_PM_MAXI_CODE_5** | Mode5 |

⑲ Cutting method (`CuttingMethod`)

Constants of enumerated type used for the cutting method are shown in the following table.

| Constant Name | Description | |
|---|---|---|
| | **Paper Feed to Cut Position** | **Cutting Method** |
| **SII_PM_CUT_FULL** | Enabled | Full cut |
| **SII_PM_CUT_FULL_NO_FEED** | Disabled | |
| **SII_PM_CUT_PARTIAL** | Enabled | Partial cut |
| **SII_PM_CUT_PARTIAL_NO_FEED** | Disabled | |
| **SII_PM_CUT_NONE**[*1] | Disabled | No cut |

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

⑳ Drawer number (`DrawerNum`)

Constants of enumerated type used for the drawer number are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_DRAWER_1** | Drawer 1 |
| **SII_PM_DRAWER_2** | Drawer 2 |

㉑ Pulse width (`PulseWidth`)

Constants of enumerated type used for the pulse width are shown in the following table. As for the drawer control time, follow the specifications of your drawer.

| Constant Name | Description |
|---|---|
| **SII_PM_ON_OFF_TIME_100** | ON/OFF time 100 milliseconds |
| **SII_PM_ON_OFF_TIME_200** | ON/OFF time 200 milliseconds |
| **SII_PM_ON_OFF_TIME_300** | ON/OFF time 300 milliseconds |
| **SII_PM_ON_OFF_TIME_400** | ON/OFF time 400 milliseconds |
| **SII_PM_ON_OFF_TIME_500** | ON/OFF time 500 milliseconds |
| **SII_PM_ON_OFF_TIME_600** | ON/OFF time 600 milliseconds |
| **SII_PM_ON_OFF_TIME_700** | ON/OFF time 700 milliseconds |
| **SII_PM_ON_OFF_TIME_800** | ON/OFF time 800 milliseconds |

㉒ Buzzer pattern (`BuzzerPattern`)

Constants of enumerated type used for the buzzer pattern of the external buzzer are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_BUZZER_PATTERN_1** | Pattern 1 |
| **SII_PM_BUZZER_PATTERN_2** | Pattern 2 |
| **SII_PM_BUZZER_PATTERN_3** | Pattern 3 |
| **SII_PM_BUZZER_PATTERN_4** | Pattern 4 |

㉓ Memory area (`MemoryArea`)

Constants of enumerated type used for operating memory area are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_MEMORY_DISPLAY_USERMEMORY** | User area |

㉔ Registered font (`RegisterdFont`)

Constants of enumerated type used for registered font are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_FONT_STANDARD** | Standard font |
| **SII_PM_FONT_OPTION** | Optional font |

㉕ QR data mode (`QrDataMode`)

Constants of enumerated type used for QR data mode are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_QRDATAMODE_NUMERIC** | Numeric mode |
| **SII_PM_QRDATAMODE_ALPHANUMERIC** | Alphanumeric mode |
| **SII_PM_QRDATAMODE_8BITBYTE** | 8-bit byte mode |
| **SII_PM_QRDATAMODE_KANJI** | Kanji mode |
| **SII_PM_QRDATAMODE_MIXTURE** | Mixed mode |

㉖ QR quiet zone (`QrQuietZone`)

Constants of enumerated type used for QR quiet zone are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_QRQUIETZONE_EXIST** | Set QR quiet zone on |
| **SII_PM_QRQUIETZONE_NONE** | Set QR quiet zone off |

㉗ Macro registration processing (`MacroRegistrationFunction`)

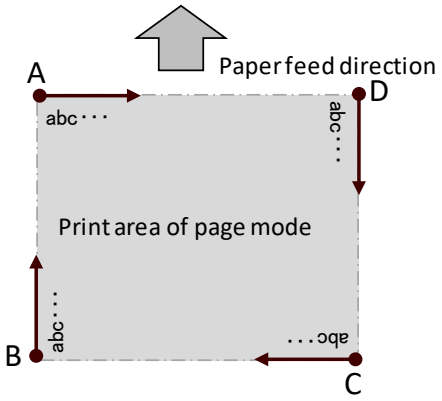
Constants of enumerated type used for macro registration processing are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_MACRO_REGISTRATION_CLEAR** | Cancel macro registration processing |
| **SII_PM_MACRO_REGISTRATION_START** | Start macro registration processing |
| **SII_PM_MACRO_REGISTRATION_REGIST** | Finish macro registration and macro registration processing |

㉘ Print direction (`Direction`)

Constants of enumerated type used for print direction in page mode are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_DIRECTION_LEFT_TO_RIGHT** | Starting point: Upper left (A on the figure),<br>Print direction: Left to Right |
| **SII_PM_DIRECTION_BOTTOM_TO_TOP** | Starting point: Left below (B on the figure),<br>Print direction: Below to Upper |
| **SII_PM_DIRECTION_RIGHT_TO_LEFT** | Starting point: Right below (C on the figure),<br>Print direction: Right to Left |
| **SII_PM_DIRECTION_TOP_TO_BOTTOM** | Starting point: Upper right (D on the figure),<br>Print direction: Upper to Below |




㉙ Line style (`LineStyle`)

Constants of enumerated type used for line style in page mode are shown in the following table.

| Constant Name | Description |
|---|---|
| **SII_PM_LINESTYLE_THIN** | Thin solid line (2 dots) |
| **SII_PM_LINESTYLE_MEDIUM** | Medium solid line (4 dots) |
| **SII_PM_LINESTYLE_THICK** | Thick solid line (8 dots) |

(5) Method Details

① 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`** is executed.

## `init` Instance

Syntax
```
- (id) init;
```

Description This method initializes the instance of **`SIIPrinterManager`** class.

Return value When succeeded, the initialized instance of **`SIIPrinterManager`** class is returned. When failed, nil is returned.

Example of use
```
SIIPrinterManager *printerManager =
 [[SIIPrinterManager alloc] init];
```

## `connect` Start communicating with printer

Starts communicating with the printer.

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

Parameter

| | |
|---|---|
| `printerModel` | Printer model constant.<br>See "4.3.1(3)① Printer model" for available constants. |
| `address` | Depends on the setting of `portType`.<br>・For **SII_PM_PRINTER_PORT_TYPE_BLUETOOTH**:<br>Specify the Bluetooth device name (Bluetooth Accessory).<br>Example: "RP-F10"<br>・For **SII_PM_PRINTER_PORT_TYPE_USB**<br>Specify the printer name<br>Example: "RP-F10"<br>・For **SII_PM_PRINTER_PORT_TYPE_TCP**:<br>Specify the IP address of the printer.<br>Example: "192.168.0.190" |
| `portType` | Port type<br>See "4.3.1(3)② Port type" for available constants. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

Description Call this method before using other class methods.
In order to make this library work properly, this method may change the printer settings when connecting.

For Bluetooth connection:
Communication with a printer paired with iOS device starts through Bluetooth connection. Connection is made to the paired Bluetooth device (Bluetooth accessory) specified by `address`.

For USB connection:
Communication with a printer connected with the iOS device through the USB cable starts through the USB connection.

For TCP/IP connection:
Communication with a printer connected to the same network as the iOS device starts through TCP/IP connection. Connection is made to the IP address specified by `address`. TCP port 9100 and 26100 are used for communication.

- **Creating/discarding of socket in TCP/IP connection of the library**

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

  Based on the completion 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 printer establishes communication with that host, so the reconnection may fail.

**Note** This method does not support a concurrent connection from multiple apps to one printer.

## `disconnect` Stop communicating with printer

Stops communicating with the printer.

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

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

**Note** It is recommended to get execution response by **SII_PM_PRINTER_RESPONSE_REQUEST** of **`getPrinterResponse`** before executing this method. If not, the following problems may occur:

- The communication is disconnected before the print data sending from iOS device to the printer is completed, and a part of the data may be lost.
- In Bluetooth connection, when either **`disconnect`** or **`connect`** is executed while the printer is in the buffer full state*1, the communication between iOS devise and the printer may be disconnected.

  *1: The state of buffer full means that the buffer of the printer is filled with print data. The size to be in buffer full state is approximately 4K bytes.

If you do not execute **`getPrinterResponse`** in your program, please fully evaluate your program to confirm no problems arise.

## `openDrawer` Open cash drawer

Opens the specified cash drawer.

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

Parameter

| | |
|---|---|
| `drawerNum` | Drawer number<br>See "4.3.1(4)⑳ Drawer number (`DrawerNum`)" for available constants. |
| `onOffTime` | Pulse width<br>See "4.3.1(4)㉑ Pulse width (`PulseWidth`)" for available constants. |

Error **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

## buzzer Sound buzzer

This method is not supported. When this method is executed, **SIIPrinterException** is thrown.

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

## externalBuzzer Sound external buzzer

Sounds the external buzzer.

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

Parameter

| | |
|---|---|
| `buzzerPattern` | Buzzer pattern<br>See "4.3.1(4)㉒ Buzzer pattern (`BuzzerPattern`)" for available constants.<br>The external buzzer sound stops under one of the following conditions:<br>・ Sounding for the number of times set by `buzzerCount`<br>・ Opening the cover<br>・ Executing the printer command "Stop External Buzzer" |
| `buzzerCount` | Buzzer sound count (times)<br>The external buzzer sounds for the number of times set by `buzzerCount`.<br>The valid range is 1 to 255. |

Error **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

## getStatus Get printer status

Gets the latest printer status.

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

Parameter

| | |
|---|---|
| `buf` | Status retrieved from the printer |

Error **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description The status retrieved from the printer is stored in an NSInteger array.

The printer status is shown below.
When the connection failed, the printer status is shown in 0x80000000.

| Bit | Function | Value | |
|---|---|---|---|
| | | 0 | 1 |
| 0 | Voltage error | No error | Error |
| 1 | Hardware error | No error | Error |
| 2 | Head temperature error | No error | Error |
| 3 | Reserved | No error | Error |
| 4 | Out-of-paper error | No error | Error |
| 5 | Reserved | Fixed | - |
| 6 | Reserved | Fixed | - |
| 7 | Cover open error | No error | Error |
| 8 | FEED Switch status | OFF | ON |
| 9 | Reserved | Fixed | - |
| 10 | Paper feed status | Stop | Operating |
| 11 | Return-waiting status | Not waiting | Waiting |
| 12 | Reserved | Fixed | - |
| 13 | Reserved | - | Fixed |
| 14 | Reserved | - | Fixed |
| 15 | Drawer switch input status | Low | High |
| 16 | FLASH memory rewriting | Not rewriting | Rewriting |
| 17 | Peripheral device selection | Printer | Others |
| 18 | Reserved | Fixed | - |
| 19 | Reserved | - | Fixed |
| 20 to 31 | Reserved | - | Fixed |

## `abort` Abort waiting state of printer

Aborts the waiting state of the printer.

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

Error
**`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

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

## registerLogo — Register logo

Registers image file to the NV graphics memory in the printer as a logo.
The method of syntax (a), dithering can be specified.
The method of syntax (b), dithering is fixed to be disabled.

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 be registered as a logo<br>Supported image file extensions are .bmp, .jpg, .jpeg, and .png. Colored image is converted to monochrome image by binarization and registered. |
| `logoId` | ID of the logo to be registered (key code)<br>Specify the ID of the logo to be registered as a two-character string.<br>Valid characters are ASCII character codes from 20h (space) to 7Eh (tilde) such as alphanumeric ('0' to '9', 'A' to 'Z', 'a' to 'z'). |
| `dithering` | Dithering<br>See "4.3.1(4)① Dithering (`Dithering`)" for available constants. |

Error

**`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

## unregisterLogo — Delete registered logo

Deletes the registered logo.

Syntax

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

Parameter

| | |
|---|---|
| `logoId` | ID of the logo to be deleted (key code)<br>Specify the ID of the registered logo as a character string. |

Error

**`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

## registerStyleSheet — Register style sheet

This method is not supported. When this method is executed, **`SIIPrinterException`** is thrown.

Syntax

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

## unregisterStyleSheet — Delete registered style sheet

This method is not supported. When this method is executed, **`SIIPrinterException`** is thrown.

Syntax

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

## resetPrinter — Reset printer

Performs a hardware reset of the printer.

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

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description
For Bluetooth connection:
The printer hardware reset is performed by the printer command "Printer Reset". When using this method, enable iOS Auto Connection in the iOS app "SII RP Utility" on the App Store. When it is disabled, this method fails to reconnect after reset and **SIIPrinterException** is thrown.
This method takes about 10 seconds to complete reconnection with the printer after performing the reset. Use this method after setting a sufficient receive timeout period.

For USB connection:
The hardware reset is not supported by USB communication.
When this method is executed in USB communication, the iOS device may not recognize the printer.
When reconnection with the printer fails after executing this method, disconnect the USB cable and then reconnect.

For TCP/IP connection:
The reset is performed to the connected printer by our proprietary command (reset request) to TCP port 26100.

The connection with the printer is retained even after this method is executed.

## getPrinterResponse — Get various responses from printer

Gets response data from the printer.

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

Parameter

| | | |
|---|---|---|
| `responseId` | Response type constant<br>See "4.3.1(3)③ Printer response type" for available constants. |
| `param` | Command parameter<br>The value to be specified varies with the response type constant.<br>See the following table for description of the value to be specified. |
| `response` | Buffer for storing the retrieved response data<br>The buffer type varies with the response type constant.<br>See the following table for the buffer type. |

**Response Type Constant**

| Parameter | Description |
|---|---|
| **SII_PM_PRINTER_RESPONSE_REQUEST** (Execution response request) | |
| `param` | Specify 0 to 15 (00h to 0Fh) in NSData type. |
| `response` | Specify an NSInteger array of length 1.<br>When the response is retrieved successfully, the response code of the execution response request is stored with 128 to 143 (80h to 8Fh). |

| Response Type Constant | | |
|---|---|---|
| | **Parameter** | **Description** |
| **SII_PM_PRINTER_RESPONSE_USER_AREA** (Send remaining capacity of user area) | | |
| | `param` | Specify nil. |
| | `response` | Specify an NSInteger array of length 1.<br>When the response is retrieved successfully, the remaining capacity of the user area is stored as a numerical value in bytes. |
| **SII_PM_PRINTER_RESPONSE_ARRANGE_USER_AREA** (Send remaining capacity of user area after defragment) | | |
| | `param` | Specify nil. |
| | `response` | Specify an NSInteger array of length 1.<br>When the response is retrieved successfully, the remaining capacity of the user area after defragment is stored as a numerical value in bytes. |
| **SII_PM_PRINTER_RESPONSE_NV_GRAPHICS** (Send NV graphics memory capacity) | | |
| | `param` | Specify nil. |
| | `response` | Specify an NSInteger array of length 1.<br>When the response is retrieved successfully, the NV graphics memory capacity is stored as a numerical value in bytes. |
| **SII_PM_PRINTER_RESPONSE_KEY_CODE** (Send key code list of defined NV graphics) | | |
| | `param` | Specify nil. |
| | `Response` | Specify an NSMutableArray array.<br>When the response is retrieved successfully, the key code of NV graphics is stored as a string array. |
| **SII_PM_PRINTER_RESPONSE_FIRMWARE_VERSION** (Send firmware version) | | |
| | `param` | Specify nil. |
| | `Response` | Specify an NSMutableArray array.<br>When the response is retrieved successfully, the firmware version is stored as a string array. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

## `startDiscoveryPrinter` Start printer search (Bluetooth)

Searches for the Bluetooth device (Bluetooth accessory).

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

Parameter

| | |
|---|---|
| `predicate` | Specify nil. |
| `completion` | Completion event of **`EABluetoothAccessoryPickerCompletion`**<br>Specify ^(`NSError *error`) to receive the completion event of **`EABluetoothAccessoryPickerCompletion`**. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.

Description This method searches for the Bluetooth device (Bluetooth accessory). This method is calling `showBluetoothAccessoryPickerWithNameFilter` of `EAAccessoryManager` internally. Pairing with the Bluetooth device can be performed in the window displayed at execution of this method.

Example when specifying ^(`NSError *error`)
(Statement of `EABluetoothAccessoryPickerCompletion`)

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

## startDiscoveryPrinter — Start printer search (TCP/IP)

Searches for SII printer connecting to the same network.

Syntax

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

Parameter

| | |
|---|---|
| `retryCount` | Retry count (times)<br>Sends the local broadcast packet the number of times set by `retryCount`.<br>The valid range is 1 to 5.<br>When the value is specified less than 1, the number is set to 1.<br>When the value is specified more than 5, the number is set to 5. |
| `timeout` | Search timeout period (millisecond: ms)<br>Sets the timeout period per search. Each time the local broadcast packet is sent, this method waits for a response from the printer until the period specified by `timeout` elapses.<br>The valid range is 3000 to 60000.<br>When the value is specified less than 3000, the period is set to 3000 ms.<br>When the value is specified more than 60000, the period is set to 60000 ms. |
| `completion` | Printer search completion event<br>Notifies the block set by `completion` as an event. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.

Description This method searches for SII printers. The printer information of the found printer can be retrieved by **`getFoundPrinter`**.

The definition of `SIIDiscoveryPrinterCompletion` is as follows:
`typedef void(^SIIDiscoveryPrinterCompletion)(NSArray *printerList);`

## cancelDiscoveryPrinter — Cancel printer search

Cancels **`startDiscoveryPrinter`** (TCP/IP) under execution.

Syntax

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

Description This method is available only when `portType` of **`connect`** is **SII_PM_PRINTER_PORT_TYPE_TCP**.

The cancellation of the search is notified as an event to the block set to `completion` of **`startDiscoveryPrinter`**.

## getFoundPrinter — Get found printer information

Returns the printer information found by **startDiscoveryPrinter** (TCP/IP) in NSArray type.

Syntax
```
- (NSArray *)getFoundPrinter;
```

Description
This method is available only when portType of **connect** is **SII_PM_PRINTER_PORT_TYPE_TCP**.

See "**4.3.2 SIIPrinterInfo Class**" for details of the printer information.

Return value
NSArray type printer information

## getVersion — Get SDK version

Gets the SDK version as a character string.

Syntax
```
- (NSString *)getVersion;
```

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

Description
This method can get the SDK version regardless of whether **isConnect** is YES or NO.

## controlTransaction — Start/End batch processing

Starts or ends batch processing.

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

Parameter
`control` Batch processing selection
See "4.3.1(4)② Batch processing selection (TransactionFunction)" for available constants.

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description
The procedure of batch processing is as follows:
(1) Starts batch processing.
Specify **SII_PM_TRANSACTION_START.**
(2) Executes 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) Finishes 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**

The batch processing target methods are as follows:

- **sendText**
- **sendTextEx**
- **printBarcode**
- **printPDF417**
- **printQRcode**
- **printDataMatrix**
- **printMaxiCode**
- **printGS1DataBarStacked**
- **printGS1DataBarStackedOmnidirectional**
- **printGS1DataBarExpandedStacked**
- **cutPaper**
- **openDrawer**
- **externalBuzzer**
- **sendBinary**
- **sendDataFile**
- **printLogo**[*1]
- **enterPageMode**
- **exitPageMode**
- **setPageModeArea**
- **setPageModeDirection**
- **setPageModeLineSpacing**
- **printPageMode**
- **printPageModeText**
- **printPageModeTextEx**
- **printPageModeBarcode**
- **printPageModePDF417**
- **printPageModeQRcode**
- **printPageModeDataMatrix**
- **printPageModeMaxiCode**
- **printPageModeGS1DataBarStacked**
- **printPageModeGS1DataBarStackedOmnidirectional**
- **printPageModeGS1DataBarExpandedStacked**
- **sendPageModeBinary**
- **printPageModeImageFile**
- **printPageModeRectangle**
- **printPageModeLine**
- **printPageModeLogo**[*1]

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

## defragment — Optimize memory area

Optimizes the memory area.

Syntax
```
- (void) defragment:(MemoryArea)memoryArea;
```

Parameter

| | |
|---|---|
| memoryArea | Memory area<br>See "4.3.1(4)㉓ Memory area (MemoryArea)" for available constants. |

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description
It may take several minutes for optimization.
Do not turn the printer power off during optimization.
Display is changed to Standby mode when this method is executed. A selecting template is deselected.

This method is ignored when Display is not connected to the printer.

## initializeMemoryArea — Initialize memory area

Initializes the memory area.

Syntax
```
- (void)initializeMemoryArea:(MemoryArea)memoryArea;
```

Parameter

| | |
|---|---|
| memoryArea | Memory area<br>See "4.3.1(4)㉓ Memory area (MemoryArea)" for available constants. |

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description
It may take several minutes for initialization.
Do not turn the printer power off during the initialization.
Display is changed to Standby mode when this method is executed. A selecting template is deselected.

This method is ignored when Display is not connected to the printer.

**Note**
Registered data in following methods is deleted when the memory area is initialized after specifying **SII_PM_MEMORY_DISPLAY_USERMEMORY**.

- **registerTemplate**
- **registerImageData**
- **controlMacroRegistration**
- **registerUserDefinedCharacter**
- **registerOptionFont**

**In addition, part of data which is registered at the shipping to use for the system is deleted either. Therefore Display becomes impossible to change to Guide mode when an error occurs in the printer.**
The used memory can be reused after executing **defragment**.

## showTemplate — Display template

Displays the template on the screen.

Syntax

```
- (void)showTemplate:(NSInteger)time_ms;
```

Parameter

| | |
|---|---|
| time_ms | Display time (ms: millisecond)<br>Specify display time on the screen with time_ms (ms)<br>The valid range is 0 to 25500.<br>When the value exceeds 0 and less than 100 is specified, the time is set to 100 ms.<br>When the value exceeding 25500 is specified, the time is set to 25500 ms.<br><br>For macro registration:<br>When 0 is specified with time_ms, the template is shown continuously.<br>When other than 0 is specified with time_ms, a next template is shown after the display time is elapsed.<br>For other than macro registration:<br>When 0 is specified with time_ms, the template is shown continuously.<br>When other than 0 is specified with time_ms, the template returns to a previous template after the display time is elapsed. In case of the previous screen has been updated with the display time other than 0, the screen is traced back to the template which was updated with the display time 0. |

Error

**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description

Updates the screen, and displays data being specified with the following methods.

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

This method is ignored when Display is not connected to the printer.

## showSlide — Display slide

Displays the slide on the screen.

Syntax

```
- (void)showslide:(NSInteger)slideID;
                  time_ms:(NSInteger)time_ms;
```

Parameter

| | |
|---|---|
| `slideID` | Slide ID<br>Specify the ID of the slide data which is registered at **`registerSlideData`**.<br>The valid range is 0 to 91.<br>This method is ignored when slide data is not registered in the specified ID. |
| `time_ms` | Display time (ms: millisecond)<br>Specify display time on the screen with `time_ms` (ms)<br>The valid range is 0 to 25500.<br>When the value exceeds 0 and less than 100 is specified, the display time is set to 100 ms.<br>When the value exceeds 25500 is specified, the display time is set to 25500 ms.<br><br>For macro registration:<br>When 0 is specified with `time_ms`, the slide is shown continuously.<br>When other than 0 is specified with `time_ms`, a next slide is shown after the display time is elapsed.<br>For other than macro registration:<br>When 0 is specified with `time_ms`, the slide is shown continuously.<br>When other than 0 is specified with `time_ms`, the slide returns to a previous slide after the display time is elapsed. In case of the previous screen has been updated with the display time other than 0, the screen is traced back to the slide which was updated with the display time 0. |

Error

**`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

Description

A selecting template is deselected.

This method is ignored when Display is not connected to the printer.

## enterStandbyMode — Display standby

Changes Display to Standby mode.

Syntax

```
- (void) enterStandbyMode;
```

Error

**`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

Description

This method is ignored during Standby mode or Guide mode is being displayed.
A selecting template is deselected.

This method is ignored when Display is not connected to the printer.

## executeMacro — Execute macro

Executes the macro.

Syntax

```
- (void)executeMacro:(NSInteger)macroID
                repeatCount:(NSInteger)repeatCount;
```

Parameter

| | |
|---|---|
| macroID | Macro ID<br>Specify the macro ID which is registered at **controlMacroRegistration**.<br>The valid range is 0 to 127.<br>This method is ignored when the macro is not registered in the specified ID. |
| repeatCount | The number of execution times<br>Specify times to execute the macro.<br>The valid range is 0 to 255.<br>Continues the repeating when 0 is specified. |

Error

**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description

A selecting template is deselected.

This method is ignored when Display is not connected to the printer.

## turnOnScreen — Turn on/off screen

Sets the screen backlight on/off.

Syntax

```
- (void)turnOnScreen:(BOOL)isOn;
```

Parameter

| | |
|---|---|
| isOn | Screen status<br>Specify the screen status from following.<br>Yes: backlight on<br>No: backlight off |

Error

**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description

This method is ignored when Display is not connected to the printer.

## selectTemplate — Select template

Selects the template to show on Display.
The method of syntax (a) selects slide data to be used for the template or the template background.
The method of syntax (b) selects a template.

Syntax

(a)
```
- (void)selectTemplate:(NSInteger)templateID
                slideID:(NSInteger)slideID;
```

(b)
```
- (void)selectTemplate:(NSInteger)templateID;
```

Parameter

| | |
|---|---|
| `templateID` | Template ID<br>Specify the ID of template to select.<br>The valid range is 0 to 127.<br>This method is ignored when the template is not registered in the specified ID. |
| `slideID` | Slide ID<br>Specify the ID of slide data to use for the background of the template.<br>The valid range is 0 to 91.<br>This method is ignored when slide data is not registered in the specified ID. |

Error

**`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

Description

The data on the template is cleared when this method is executed.
The selected template is displayed to the screen when **`showTemplate`** is executed.

The selecting template is deselected when **`showSlide`**, **`enterStandbyMode`**, or **`executeMacro`** is executed.
The selecting template is deselected when the specified display time is executed at **`showTemplate`**.

Use following templates depends on the values of **`codePage`** when characters other than 20h to 7Eh of ASCII character are input with **`setTemplateTextData`**.

・When **`codePage`** is **`CODE_PAGE_KATAKANA`**:
Use the template which encoding specifying is Shift_JIS.
・When **`codePage`** is other than **`SII_PM_CODE_PAGE_KATAKANA:`**
Use the template which encoding specifying is ISO-2022-JP.

This method is ignored when Display is not connected to the printer.

## `setTemplateImageData` Set image data

Sets image data to show on a selecting template.

Syntax

```
- (void)setTemplateImageData:(NSInteger)mapID
                     imageID:(NSInteger)imageID;
```

Parameter

| | |
|---|---|
| `mapID` | Map ID<br>The valid range is 0 to 63.<br>This method is ignored when a specified map ID is not defined in the template. |
| `imageID` | Image ID<br>Specify the ID of image data which was registered at **`registerImageData`**.<br>The valid range is 0 to 63.<br>This method is ignored when image data is not registered in the specified ID. |

Error

**`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

Description

After specifying the map ID of the selecting template with this method, specify the image ID to map.
The specified image data is displayed to the screen when **`showTemplate`** is executed.

This method setting is cleared under the following conditions.
・When **selectTemplate** is executed.
・When other than 0 is specified at time_ms of **showTemplate**, and the specified display time has elapsed.
・When **showtTemplate** registered in **executeMacro** is executed.

This method is ignored when a template is not selected.
This method is ignored when Display is not connected to the printer.

## selectTemplateTextObject — Select text element

Selects the text element to start editing.

Syntax
```
- (void)selectTemplateTextObject:(NSInteger)mapID;
```

Parameter

| | |
|---|---|
| mapID | Map ID<br>The valid range is 0 to 63.<br>This method is ignored when a specified map ID is not defined in the template. |

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description
After specifying a map ID of the selecting template with this method, start to edit characters.

When a scroll is set with the text element of the specified map ID and this method is executed after **showTemplate**, the scroll is executed.
When a scroll is not set with the text element of the set map ID and this method is executed after **showTemplate**, input text data is discarded.

This method setting is cleared under the following conditions.
・When **selectTemplate** is executed.
・When **showtTemplate** is executed.

This method is ignored when a template is not selected.
This method is ignored when Display is not connected to the printer.

## setTemplateTextAlignment — Alignment of text data

Sets alignment to text data shown on Display.

Syntax
```
- (void)setTemplateTextAlignment:(PrintAlignment)alignment;
```

Parameter

| | |
|---|---|
| alignment | Alignment<br>See "4.3.1(4)⑨ Alignment (PrintAlignment)" for available constants. |

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description
Alignment is valid only as following cases.
・Text data is not entered in the specified map ID at **selectTemplateTextObject**.
・Text data entered in the map ID which is specified at **selectTemplateTextObject** is registered right after a line feed.

Specify the map ID at **selectTemplateTextObject** before executing this method,
Input text data at **setTemplateTextDat** after executing this method,
The input text data is displayed to the screen when **showTemplate** is executed.

This method setting is cleared under the following conditions.
・When **selectTemplate** is executed.
・When other than 0 is specified at time_ms of **showTemplate**, and the specified display time has elapsed.
・When **showtTemplate** registered in **executeMacro** is executed.

This method is ignored when Display is not connected to the printer.

## setTemplateTextLeftMargin — Set left margin of text data

Sets left margin to text data shown on Display.

Syntax
```
- (void)setTemplateTextLeftMargin:(NSInteger)margin;
```

Parameter

| | |
|---|---|
| margin | Left margin (pixel: px)<br>The valid range is 0 to 479. |

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description
The left margin setting is valid only as following conditions.
・Text data is not entered in a specified map ID at **selectTemplateTextObject**.
・Text data entered in a map ID which is specified at **selectTemplateTextObject** is registered right after a line feed.

Specify the map ID at **selectTemplateTextObject** before executing this method.
Input text data at **selectTemplateTextData** after executing this method.
The input text data is displayed to the screen when **showTemplate** is executed.

This method setting is cleared under the following conditions.
・When **selectTemplate** is executed.
・When other than 0 is specified at time_ms of **showTemplate**, and the specified display time has elapsed.
・When **showtTemplate** registered in **executeMacro** is executed.

This method is ignored when Display is not connected to the printer.

## setTemplateTextLineSpacing — Set line spacing of text data

Sets line spacing to text data shown on Display.

Syntax
```
- (void)setTemplateTextLineSpacing:(NSInteger)spacing;
```

Parameter

| | |
|---|---|
| spacing | Line spacing (pixel: px)<br>The valid range is 0 to 255. |

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description
The line spacing setting is valid only as following cases.
・Text data is not entered in a specified map ID at **selectTemplateTextObject**.
・Text data entered in a map ID which is specified at **selectTemplateTextObject** is registered right after a line feed.

When a scroll is set to a text element of the specified mapID, the line spacing to text data is not reflected.

Specify the map ID at **selectTemplateTextObject** before executing this method.
Input text data at **selectTemplateTextData** after executing this method.
The input text data is displayed to the screen when **showTemplate** is executed.

This method setting is cleared under the following conditions.
・When **selectTemplate** is executed.
・When other than 0 is specified at time_ms of **showTemplate**, and the specified display time has elapsed.
・When **showtTemplate** registered in **executeMacro** is executed.

This method is ignored when Display is not connected to the printer.

## setTemplateTextBold — Set bold character of text data

Sets bold characters to text data shown on Display.

Syntax
```
- (void)setTemplateTextBold:(CharacterBold)bold;
```

Parameter

| | |
|---|---|
| bold | Bold character<br>See "4.3.1(4)③ Bold print (CharacterBold)" for available constants. |

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description
From text data after this method is executed, the bold characters are applied.
The bold character can be set one by one.

Specify the map ID at **selectTemplateTextObject** before executing this method.
Input text data at **selectTemplateTextData** after executing this method.
The input text data is displayed to the screen when **showTemplate** is executed.

This method setting is cleared under the following conditions.
・When **selectTemplate** is executed.
・When other than 0 is specified at time_ms of **showTemplate**, and the specified display time has elapsed.
・When **showtTemplate** registered in **executeMacro** is executed.

This method is ignored when Display is not connected to the printer.

## setTemplateTextUnderline — Set underline of text data

Sets underline to text data shown on Display.

Syntax
```
- (void)setTemplateTextUnderline:(CharacterUnderline)underline;
```

Parameter

| | |
|---|---|
| underline | Underline<br>See "4.3.1(4)④ Underline (CharacterUnderline)" for available constants. |

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description
From text data after this method is executed, the underlines are applied.
The underline can be set one by one.

Specify the map ID at **selectTemplateTextObject** before executing this method.
Input text data at **selectTemplateTextData** after executing this method.
The input text data is displayed to the screen when **showTemplate** is executed.

This method setting is cleared under the following conditions.
・When **selectTemplate** is executed.
・When other than 0 is specified at time_ms of **showTemplate**, and the specified display time has elapsed.
・When **showtTemplate** registered in **executeMacro** is executed.

This method is ignored when Display is not connected to the printer.

## setTemplateTextSize — Set character size of text data

Sets character size to text data shown on Display.

Syntax
```
- (void)setTemplateTextSize:(CharacterScale)scale;
```

Parameter

| | |
|---|---|
| scale | Character scale<br>See "4.3.1(4)⑧ Character scale (CharacterScale)" for available constants. |

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description
From text data after this method is executed, the character sizes are applied.
The character size can be set one by one.

Specify the map ID at **selectTemplateTextObject** before executing this method.
Input text data at **selectTemplateTextData** after executing this method.
The input text data is displayed to the screen when **showTemplate** is executed.

This method setting is cleared under the following conditions.
・When **selectTemplate** is executed.
・When other than 0 is specified at time_ms of **showTemplate**, and the specified display time has elapsed.
・When **showtTemplate** registered in **executeMacro** is executed.

This method is ignored when Display is not connected to the printer.

## setTemplateTextFont — Set character font of text data

Sets a character font to text data shown on Display.

Syntax
```
- (void)setTemplateTextFont:(CharacterFont)font;
```

Parameter

| | |
|---|---|
| font | Character font<br>See "4.3.1(4)⑦ Character font (CharacterFont)" for available constants. |

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description
From text data after this method is executed, the character fonts are applied.
The character font can be set one by one.

Specify the map ID at **selectTemplateTextObject** before executing this method.
Input text data at **selectTemplateTextData** after executing this method.
The input text data is displayed to the screen when **showTemplate** is executed.

This method setting is cleared under the following conditions.
・When **selectTemplate** is executed.
・When other than 0 is specified at time_ms of **showTemplate**, and the specified display time has elapsed.
・When **showtTemplate** registered in **executeMacro** is executed.

This method is ignored when Display is not connected to the printer.

## setTemplateTextRegisteredFont — Set registered font of text data

Sets the registered font used for text data shown on Display.

Syntax
```
- (void)setTemplateTextRegisteredFont:(RegisteredFont)font;
```

Parameter

| | |
|---|---|
| font | Registered font<br>See "4.3.1(4)㉔ Registered font (RegisterdFont)" for available constants.<br>This registered font is ignored when the optional font is not registered. |

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description
From text data after this method is executing, the registered fonts are applied.
The registered fonts can be set one by one.

Input text data with **setTemplateTextData** after executing this method.
The input text data is displayed to the screen when **showTemplate** is executed.

This method setting is cleared under the following conditions.
・When **selectTemplate** is executed.
・When other than 0 is specified at time_ms of **showTemplate**, and the specified display time has elapsed.
・When **showtTemplate** registered in **executeMacro** is executed.

This method is ignored when Display is not connected to the printer.

## setTemplateTextRightSpacing — Set right space of text data

Sets the amount of right space to text data shown on Display.

Syntax
```
- (void)setTemplateTextRightSpacing:(NSInteger)spacing;
```

Parameter

| | |
|---|---|
| spacing | The amount of character space (pixel: px)<br>The valid range is 0 to 255. |

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description
From text data after this method is executed, the amount of character right space is applied.
The amount of right space can be set one by one.

Specify the map ID at **selectTemplateTextObject** before executing this method.
Input text data at **selectTemplateTextData** after executing this method.
The input text data is displayed to the screen when **showTemplate** is executed.

This method setting is cleared under the following conditions.
・When **selectTemplate** is executed.
・When other than 0 is specified at time_ms of **showTemplate**, and the specified display time has elapsed.
・When **showtTemplate** registered in **executeMacro** is executed.

This method is ignored when Display is not connected to the printer.

## setTemplateTextColor — Set character color of text data

Sets the character color used for text data shown on Display.

Syntax
```
- (void)setTemplateTextColor:(NSInteger)color;
```

Parameter

| | |
|---|---|
| color | Character color<br>The valid range is 0 to 0xffffff. |

Error **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description This method can set character colors to text data.
The character colors can be set in RGB24 bit color. The set color is displayed in the color converted to 16 bit RGB555.

From text data after this method is executed, the character colors are applied.
The character color can be set one by one.

Specify the map ID at **selectTemplateTextObject** before executing this method.
Input text data at **selectTemplateTextData** after executing this method.
The input text data is displayed to the screen when **showTemplate** is executed.

This method setting is cleared under the following conditions.
・When **selectTemplate** is executed.
・When other than 0 is specified at time_ms of **showTemplate**, and the specified display time has elapsed.
・When **showtTemplate** registered in **executeMacro** is executed.

This method is ignored when Display is not connected to the printer.

## setTemplateTextData — Input text data

Inputs text data to show on Display.

Syntax
```
- (void)setTemplateTextData:(NSString *)text
```

Parameter

| | |
|---|---|
| text | Text data to show on Display<br>Data size which is able to be specified it once is 1 to 1020 bytes. |

Error **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description This method encodes input text data into text data which is enable to show on the basis of settings at **internationalCharacter** or **codePage**, and displays.

After specifying a map ID of the selecting template with **selectTemplateTextObject**, input text data with this method.
This method is ignored when a map ID is not specified in **selectTemplateTextObjec.**
The input text data is displayed to the screen display when **showTemplate** is executed.

The input text data is cleared under the following conditions.
・When **selectTemplate** is executed.
・When other than 0 is specified at time_ms of **showTemplate**, and the specified display time has elapsed.
・When **showtTemplate** registered in **executeMacro** is executed.

This method is ignored when Display is not connected to the printer.

## setTemplateBarcodeData — Input barcode data

Inputs a map ID of the barcode element on the selecting template, and inputs barcode data.
The method of syntax (a) inputs data with character strings to display barcode.
The method of syntax (b) inputs data with byte arrays to display barcode.

Syntax

(a) –
```
(void)setTemplateBarcodeData:(NSInteger)mapID
                        text:(NSString *)text;
```

(b) –
```
(void)setTemplateBarcodeData:(NSInteger)mapID
                        data:(NSData *)data;
```

Parameter

| | |
|---|---|
| mapID | Map ID<br>The valid range is 0 to 7.<br>This method is ignored when the specified map ID is not defined in the selecting template. |
| Text | Barcode data<br>The characters that can be input are ASCII characters 20h (space) to 7Eh (tilde) such as alphanumeric characters ('0' to '9', 'A' to 'Z', 'a' to 'z').<br>The available number of characters is 1 to 150.<br>Barcode data which is not complying with barcode specification is ignored. |
| data | Barcode data<br>The value that can be input is 00h to 7Fh.<br>The available number of data is 1 to 150.<br>Barcode data which is not comply with barcode specification is ignored. |

Error

**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description

After specifying a map ID of the selecting template with this method, input barcode data.
The input barcode data is displayed to the screen when **showTemplate** is executed.

The input barcode data is cleared on following conditions.
・When **selectTemplate** is executed.
・When other than 0 is specified at time_ms of **showTemplate**, and the specified display time has elapsed.
・When **showtTemplate** registered in **executeMacro** is executed.

This method is ignored when a template is not selected.
This method is ignored when Display is not connected to the printer.

## setTemplateQrCodeData — Input QR Code data

Specifies a map ID of a qr element on a selecting template, and inputs QR Code data.
The method of syntax (a) inputs QR Code data with character strings.
The method of syntax (b) inputs QR Code data with character strings. The setting of selecting template is reflected to `moduleSize`, `errorCorrection`, `mode`, and `qrQuietZone`.

The method of syntax (c) inputs QR Code data with byte array.
The method of syntax (e) inputs QR Code data with byte array. The setting of selecting template is reflected to `moduleSize`, `errorCorrection`, `mode`, and `qrQuietZone`.

Syntax

(a) –
```
(void)setTemplateBarcodeData:(NSInteger)mapID
                    moduleSize:(ModuleSize)moduleSize
                    errorCorrection:(ErrorCorrection)errorCorrection
                    mode:(QrDataMode)mode
                    qrQuietZone:(QrQuietZone)qrQuietZone
                    text:(String *)text;
```

(b) –
```
(void)setTemplateQRCodeData:(NSInteger)mapID
                    text:(String *)text;
```

(c) –
```
(void)setTemplateQRCodeData:(NSInteger)mapID
                    moduleSize:(ModuleSize)moduleSize
                    errorCorrection:(ErrorCorrection)errorCorrection
                    mode:(QrDataMode)mode
                    qrQuietZone:(QrQuietZone)qrQuietZone
                    data:(NSData *)data;
```

(d) –
```
(void)setTemplateQRCodeData:(NSInteger)mapID
                    data:(NSData *)data;
```

Parameter

| | |
|---|---|
| `mapID` | Map ID<br>The valid range is 0 to 7.<br>This method is ignored when the specified map ID is not defined in the selecting template. |
| `moduleSize` | Module size<br>See "4.3.1(4)⑪ Module size (`ModuleSize`)" for available constants. |
| `errorCorrection` | Error correction level<br>See "4.3.1(4)⑭ Error correction level (`ErrorCorrection`)" for available constants. |
| `mode` | Data mode<br>See "4.3.1(4)㉕ QR data mode (`QrDataMode`)" for available constants. |
| `qrQuietZone` | Quiet zone<br>See "4.3.1(4)㉖ QR quiet zone (`QrQuietZone`)" for available constants. |

| | | |
|---|---|---|
| | `text` | QR Code data<br>The characters that can be input are as follows.<br>・ASCII character codes from 20h (space) to 7Eh (tilde) alphanumeric ('0' to '9', 'A' to 'Z', 'a' to 'z').<br>・8 bits Latin / Katakana based on JIS X 0201<br>・Shift_JIS code based on JIS X 0208<br>The available data size is 1 to 3909 bytes.<br>QR Code data which is not complied with QR Code specification is ignored. |
| | `data` | QR Code data<br>QR Code data shown on Display.<br>The values that can be input is 00h to FFh.<br>The available number of data is 1 to 3909.<br>QR Code data which is not complied with QR Code specification is ignored. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

Description After specifying a map ID of the selecting template with this method, input QR Code data.
The input QR Code data is displayed to the screen when **`showTemplate`** is executed.

The input QR Code data is cleared on following conditions.
・When **`selectTemplate`** is executed.
・When other than 0 is specified at `time_ms` of **`showTemplate`**, and the specified display time has elapsed.
・When **`showtTemplate`** registered in **`executeMacro`** is executed.

This method is ignored when a template is not registered.
This method is ignored when Display is not connected to the printer.

## `registerTemplate` Register template

Registers a template in Display.

Syntax
```
- (void)registerTemplate:(NSInteger)templateID
                   label:(NSString *)label
                fileName:(NSStirng *)fileName;
```

| | | |
|---|---|---|
| Parameter | `templateID` | Template ID<br>Specify the ID of template to register.<br>The valid range is 0 to 127.<br>Do not specify the template ID of 127 because it is being used for the system. |
| | `label` | Template name<br>A name for identification can be specified to the template to be registered.<br>The characters that can be specified are ASCII code characters 20h (space) to 7Eh (tilde) such as alphanumeric characters ('0' to '9', 'A' to 'Z', 'a' to 'z').<br>Do not use Unicode 00A5h ('¥').<br>The available number of characters is 0 to 32.<br>This `label` is optional. Specify nil when the template name is not registered.<br>The specified template name can be retrieved with **`getDisplayResponse`**. |

| | | |
|---|---|---|
| | `fileName` | File path of the template data to register in Display<br>Supported file extension is .xml.<br>The maximum data size that can be registered is 8192 bytes.<br>See "Technical Reference for Display" for details on registration of the template data. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

Description Display is changed to Standby mode when this method is executed. A selecting template is deselected.

This method is ignored when Display is not connected to the printer.

**Note** **Registered data at the shipping may be added or changed without prior notice for quality improvement.**

## `unregisterTemplate` Delete template

Deletes the registered template in Display.

Syntax
```
- (void)unregisterTemplate:(NSInteger)templateID;
```

| Parameter | | |
|---|---|---|
| | `templateID` | Template ID<br>Specify the ID of template to delete.<br>The valid range is 0 to 127.<br>This method is ignored when a template is not registered in the specified ID. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

Description Display is changed to Standby mode when this method is executed. A selecting template is deselected.

Used memory is not released even the template is deleted. The used memory can be reused after executing **`defragment`**.

This method is ignored when Display is not connected to the printer.

## `registerImageData` Register image data

Registers image data in Display.

Syntax
```
- (void)registerImageData:(NSInteger)imageID
                    label:(NSString *)label
                 fileName:(NSStirng *)fileName;
```

| Parameter | | |
|---|---|---|
| | `imageID` | Image ID<br>The valid range is 0 to 63.<br>Do not select image IDs 49 to 63 because they are being used for the system. |

| | | |
|---|---|---|
| | `label` | Image name<br>A name for identification can be specified to image data to be registered.<br>The characters that can be specified are ASCII code characters 20h (space) to 7Eh (tilde) such as alphanumeric characters ('0' to '9', 'A' to 'Z', 'a' to 'z').<br>Do not use Unicode 00A5h ('¥').<br>The available number of characters is 0 to 32.<br>This `label` is optional. Specify nil when the template name is not registered.<br>The specified image name can be retrieved at **`getDisplayResponse`**. |
| | `fileName` | File path<br>Specify the file name of image data to register.<br>Supported file extensions are .jpg, .jpeg, and .png.<br>However, even the supported extensions may not be registered depending on the format.<br>The maximum file size that can be registered is 786432 bytes.<br>The maximum data size that can be registered is 480 horizontal × 272 vertical pixels(px). |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

Description Display is changed to Standby mode when this method is executed. A selecting template is deselected.

This method is ignored when Display is not connected to the printer.

**Note** **Registered data at the shipping may be added or changed without prior notice for quality improvement.**

## `unregisterImageData` Delate image data

Deletes registered image data in Display.

Syntax
```
- (void)unregisterImageData:(NSInteger)imageID;
```

| | | |
|---|---|---|
| Parameter | `imageID` | Image ID<br>The valid range is 0 to 63.<br>This method is ignored when image data is not registered in the specified ID. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

Description Display is changed to Standby mode when this method is executed. A selecting template is deselected.

Used memory is not released even image data is deleted. The used memory can be reused after executing **`defragment`**.

This method is ignored when Display is not connected to the printer.

## registerSlideData — Register slide data

Registers slide data in Display.

Syntax

```
- (void)registerSlideData:(NSInteger)imageID
                    label:(NSString *)label
                 fileName:(NSStirng *)fileName;
```

Parameter

| | |
|---|---|
| `slideID` | Slide ID<br>The valid range is 0 to 91.<br>Do not specify slide IDs of 80 to 90 because they are being used for the system. |
| `label` | Slide name<br>A name for identification can be specified to slide data to be registered.<br>The characters that can be specified are ASCII code characters 20h (space) to 7Eh (tilde) such as alphanumeric characters ('0' to '9', 'A' to 'Z', 'a' to 'z').<br>Do not use Unicode 00A5h ('¥').<br>The available number of characters is 0 to 32.<br>This `label` is optional. Specify nil when the template name is not registered.<br>The specified slide name can be retrieved by **`getDisplayResponse`**. |
| `fileName` | File path<br>Specify the file name of slide data to register.<br>Supported file name extensions are .jpg, jpeg, and .png.<br>However, even the supported extensions may not be registered depending on the format.<br>The maximum file size that can be registered is 786432 bytes.<br>The maximum data size that can be registered is 480 horizontal × 272 vertical pixels (px). |

Error

**`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

Description

Display is changed to Standby mode when this method is executed. A selecting template is deselected.

Execute **`showSlide`** to show registered slide data.
Specify the slide ID at **`showSlide`** to use a registered slide data as a backscreen of the template.

This method is ignored when Display is not connected to the printer.

**Note**

**Registered data at the shipping may be added or changed without prior notice for quality improvement.**

## unregisterSlideData — Delete slide data

Deletes registered slide data in Display.

Syntax
```
- (void)unregisterSlideData:(NSInteger)slideID;
```

Parameter

| | |
|---|---|
| `slideID` | Slide ID<br>The valid range is 0 to 91.<br>This method is ignored when slide data is not registered in a specified ID. |

Error
**`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

Description
Display is changed to Standby mode when this method is executed. A selecting template is deselected.
This method is ignored when Display is not connected to the printer.

## registerUserDefinedCharacter — Register user-defined character

Registers user-defined characters in Display.

Syntax
```
- (void) registerUserDefinedCharacter:(NSString *)fileName;
```

Parameter

| | |
|---|---|
| `fileName` | File path<br>Specify the file name of the user-defined characters to register.<br>Supported file extension is .bin.<br>See "Register User-Defined Character" of the display command in "Technical Reference for Display" for details on the user-defined character data. |

Error
**`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

Description
Display is changed to Standby mode when this method is executed. A selecting template is deselected.

Use a template which encoding specifying is Shift_JIS for displaying user-defined characters.
Specify **SII_PM_CODE_PAGE_KATAKANA** for **`codePage`** before user-defined characters are displayed.
Specify character codes that can be specified for `text` of **`SetTemplateTextData`** before user-defined characters are displayed.
The character codes that can be specified are 0xE000 to 0xE05D

This method is ignored when Display is not connected to the printer.

## unregisterUserDefinedCharacter — Delete user-defined character

Deletes registered user-defined characters in Display.

Syntax
```
- (void) unregisterUserDefinedCharacter;
```

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description
Display is changed to Standby mode when this method is executed. A selecting template is deselected.

All registered user-defined characters are deleted.
Used memory is not released even the user-defined characters are deleted. The used memory can be reused after executing **defragment**.

This method is ignored when Display is not connected to the printer.

## registerOptionFont — Register optional font

Registers optional fonts in Display.

Syntax
```
- (void)registerOptionFont:(NSInteger)startCode
                   endCode:(NSInteger)endCode
                     width:(NSInteger)width
                    height:(NSInteger)height
                  fileName:(NSString *)fileName;
```

Parameter

| | |
|---|---|
| `startCode` | Character code for registration starting<br>The valid range is 20h to FFh of ASCII character code. |
| `endCode` | Character code for registration finishing<br>The valid range is 20h to FFh of ASCII character code. |
| `width` | Character width (pixel: px)<br>The valid range is 1 to 255. |
| `height` | Character height (pixel: px)<br>The valid range is 1 to 255. |
| `fileName` | File path<br>Specify the file name of the optional font to register.<br>Supported file name extension is .bin.<br>See "Register Optional Font" of the display command in "Technical Reference for Display" for details on optional font data. |

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description
Display is changed to Standby mode when this method is executed. A selecting template is deselected.

When this method is executed with optional font registered status, the registered optional fonts are deleted, and a new memory area is allocated to register optional fonts.
Used memory is not released even the registered optional fonts are deleted. The used memory can be reused after executing **defragment**.

This method is ignored when Display is not connected to the printer.

## unregisterOptionFont — Delete optional font

Deletes registered optional fonts in Display.

Syntax
```
- (void)unregisterOptionFont;
```

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description
Display is changed to Standby mode when this method is executed. A selecting template is deselected.

All registered optional fonts are deleted.
Used memory is not released even the optional fonts are deleted. The used memory can be reused after executing **defragment**.

This method is ignored when Display is not connected to the printer.

## controlMacroRegistration — Start/End of macro registration

Specifies start or end of macro registration to use in Display.

Syntax
```
- (void) controlMacroRegistration:(NSInteger)macroID
                          control:(MacroRegistrationFunction)control;
```

Parameter

| | |
|---|---|
| `macroID` | Macro ID<br>The valid range is -1 to 127.<br>Do not select macro IDs of 120 to 126 because they are being used for the system. |
| `control` | Macro registration processing<br>See "4.3.1(4)㉗ Macro registration processing (`MacroRegistrationFunction`)" for available constants. |

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description
The procedures of the macro registration processing are as follows:
(1) Starts macro registration processing.
Specify -1 for `macroID`.
Specify **SII_PM_MACRO_REGISTRATION_START** for `control`.

(2) Executes methods.
Starts buffering of transmit data when methods are targeted in macro registration processing.
Transmit data of a macro registration processing target method which is executed during the buffering is not sent to the printer, buffered in macro data buffer. The maximum transmit data size to be able to buffer is 1024 bytes.
When the buffered transmit data exceeds the maximum size, the macro registration processing target method at the point of exceeding is to be error.
When the error occurs, data under the registration is discarded and canceled the macro mode.
Regarding transmit data which is held, finish the macro registration processing by procedure (3).
When a method is out of the macro registration processing target, it is executed immediately without buffering the transmit data.

Methods for the macro registration processing target are shown below.

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

(3) Finishes macro registration processing.
Specify the macro ID (0 to 127) to register at `macroID`.
When **SII_PM_MACRO_REGISTRATION_REGIST** is specified at `control`, buffered transmit data is sent to the printer. The buffered transmit data is held even after transmitting to the printer. Display is changed to Standby mode when this method is executed.
A selecting template is being deselected.

The holding transmit data is discarded by following processes.

- Specify **SII_PM_MACRO_REGISTRATION_CLEAR**
- Specify **SII_PM_MACRO_REGISTRATION_START**
- Execute **disconnect**

The registered macro can be executed at **executeMacro.**

A process to delete the registered macro is as follow.
Specify **SII_PM_MACRO_REGISTRATION_START** at `control` and specify -1 for `macroID` to call this method.
Specify **SII_PM_MACRO_REGISTRATION_START** at `control` and specify the macro ID to delete, and then call this method.
Display is changed to Standby mode when this method is executed. A selecting template is being deselected.

This method is ignored when Display is not connected to the printer.

**Note** **Registered data at the shipping may be added or changed without prior notice for quality improvement.**

## getDisplayResponse — Get various response from Display

Gets response data from Display.

Syntax

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

Parameter

| | |
|---|---|
| responseId | Display response type constant<br>See "4.3.1(3)④ Display response type" for available constants. |
| param | Command parameter<br>A value to be specified varies depending on Display response type constant.<br>See the following table for description of the value to be specified. |
| response | Buffer for storing the retrieved response data<br>A buffer type varies with Display response type constant.<br>See the following table for the buffer type. |

| Response Type Constant | | |
|---|---|---|
| | **Parameter** | **Description** |
| **SII_PM_DISPLAY_RESPONSE_REQUEST** (Execution response request) | | |
| | param | Specify 0 to 15 (00h to 0Fh) in NSData type. |
| | response | Specify an NSInteger array of length 1.<br>When the response is retrieved successfully, the response code of the execution response request is stored with 64 to 79 (40h to 4Fh). |
| **SII_PM_DISPLAY_RESPONSE_USER_AREA** (Send remaining capacity of user area) | | |
| | param | Specify nil. |
| | response | Specify an NSInteger array of length 1.<br>When the response is retrieved successfully, the remaining capacity of the user area is stored as a numerical value in bytes. |
| **SII_PM_DISPLAY_RESPONSE_TEMPLATE_ID_LIST** (Send template ID) | | |
| | param | Specify nil. |
| | response | Specify an NSMutableArray.<br>When the response is retrieved successfully, the registered template ID is stored as an NSInteger array. |
| **SII_PM_DISPLAY_RESPONSE_IMAGE_ID_LIST** (Send image ID) | | |
| | param | Specify nil. |
| | response | Specify an NSMutableArray.<br>When the response is retrieved successfully, the registered image ID is stored as an NSInteger array. |

| Response Type Constant | | |
|---|---|---|
| | **Parameter** | **Description** |
| **SII_PM_DISPLAY_RESPONSE_TEMPLATE_LABEL** (Send slide ID) | | |
| | `param` | Specify nil. |
| | `response` | Specify an NSMutableArray.<br>When the response is retrieved successfully, the registered slide ID is stored as an NSInteger array. |
| **SII_PM_DISPLAY_RESPONSE_SLIDE_LABEL** (Send template name) | | |
| | `param` | Specify 0 to 127 (00h to 7Fh) in NSData type. |
| | `response` | Specify an NSMutableArray.<br>When the response is retrieved successfully, the template name specified at template registration is stored as a character string. |
| **SII_PM_DISPLAY_RESPONSE_IMAGE_LABEL** (Send image name) | | |
| | `param` | Specify 0 to 63 (00h to 3Fh) in NSData type. |
| | `response` | Specify an NSMutableArray.<br>When the response is retrieved successfully, the image name specified at image data registration is stored as a character string. |
| **SII_PM_DISPLAY_RESPONSE_SLIDE_LABEL** (Send slide name) | | |
| | `param` | Specify 0 to 91 (00h to 5bh) in NSData type. |
| | `response` | Specify an NSMutableArray.<br>When the response is retrieved successfully, the slide name specified at slide data registration is stored as a character string. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

② 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

Sends text data.

Syntax

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

Parameter

| | |
|---|---|
| text | Text data to send to the printer<br>Data size that can be specified at one time is 16 KB (16384 bytes). |

Error

**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description

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

This method does not add a line feed code at the end of the text data. In order to print to the end, add a line feed code to the end of the text data.

## sendTextEx — Send format specified text data

Sends format specified text data to the printer.
The method of syntax (a) can specify bold character, underline, reverse print, font, character scale and alignment to text data.
The method of syntax (b) can specify bold character, underline, font and character scale to text data.
The method of syntax (c) can specify bold character, underline, inversion print, reverse print, font, character scale, and alignment to text data.

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
                       font:(CharacterFont)font
                       scale:(CharacterScale)scale;

(c)- (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 the printer<br>Data size that can be specified at 1 time is 16 KB (16384 bytes). |

| | | |
|---|---|---|
| | `bold` | Bold character<br>See "4.3.1(4)③ Bold print (`CharacterBold`)" for available constants. |
| | `underline` | Underline<br>See "4.3.1(4)④ Underline (`CharacterUnderline`)" for available constants. |
| | `reverse` | Reverse print<br>See "4.3.1(4)⑤ Reverse print (`CharacterReverse`)" for available constants. |
| | `inversion` | Inversion print<br>See "4.3.1(4)⑥ Inversion print (`CharacterInversion`)" for available constants. |
| | `font` | Character font<br>See "4.3.1(4)⑦ Character font (`CharacterFont`)" for available constants. |
| | `scale` | Character scale<br>See "4.3.1(4)⑧ Character scale (`CharacterScale`)" for available constants. |
| | `alignment` | Alignment<br>See "4.3.1(4)⑨ Alignment (`PrintAlignment`)" for available constants. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

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

This method does not add a line feed code at the end of the text data. In order to print to the end, add a line feed code to the end of the text data.

## `printBarcode` Print barcode

Prints the barcode.
The method of syntax (a) specifies the barcode data by character string.
The method of syntax (b) specifies the barcode data by character string, and specifies the alignment and N:W ratio of the barcode.
The method of syntax (c) specifies the barcode data by byte array and specifies the alignment of the barcode.
The method of syntax (d) is not supported.

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
See "4.3.1(4)⑩ Barcode symbol (`BarcodeSymbol`)" for available constants and corresponding 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<br>JAN13 | 12 to 13 characters ' | '0' to '9' | |
| EAN8<br>JAN8 | 7 to 8 characters | '0' to '9' | |
| CODE39 | 1 to 150 characters | '0' to '9'<br>'A' to 'Z'<br>' ', '$', '%', '+', '-', '.', '/' | 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.<br>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'<br>'$', '+', '-', '.', '/', ':' | It is needed to specify one of 'A' to 'D' at the beginning and end. |
| EAN13 add-on<br>JAN13 add-on | Add-on 2: 14 to 15 characters<br>Add-on 5: 17 to 18 characters | '0' to '9' | |
| Customer Bar Code_JP | - | - | Not supported. |
| GS1 Databar Omni-directional | 13 characters | '0' to '9' | Check digit is automatically added. |
| GS1 Databar Truncated | 13 characters | '0' to '9' | Check digit is automatically added. |
| GS1 Databar Limited | 13 characters | '0' to '9' | Check digit is automatically added. |
| GS1 Databar Expanded | 2 to 255 characters | ' ' to ""<br>'%' to '?'<br>'A' to 'Z'<br>'_'<br>'a' to 'z'<br>'{' | |

`moduleSize` Barcode width
See "4.3.1(4)⑪ Module size (`ModuleSize`)" for available constants.

`moduleHeight` Barcode height (dot)

- When `barcodeSymbol` is below, the valid range is 1 to 255.
  **SII_PM_BARCODE_UPC_A**
  **SII_PM_BARCODE_UPC_E**
  **SII_PM_BARCODE_EAN13**
  **SII_PM_BARCODE_JAN13**
  **SII_PM_BARCODE_EAN8**
  **SII_PM_BARCODE_JAN8**
  **SII_PM_BARCODE_CODE39**
  **SII_PM_BARCODE_CODE93**
  **SII_PM_BARCODE_CODE128**
  **SII_PM_BARCODE_ITF**
  **SII_PM_BARCODE_CODABAR**
  **SII_PM_BARCODE_EAN13_ADDON**
  **SII_PM_BARCODE_JAN13_ADDON**

- When `barcodeSymbol` is below, the valid range varies depending on `barcodeSymbol` and `moduleSize`.

| `barcodeSymbol` | `moduleSize` | **Valid Range** |
|---|---|---|
| **SII_PM_BARCODE_GS1_OMNI_DIRECTIONAL** | | |
| | **SII_PM_BARCODE_MODULE_WIDTH_2** | 66 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | 99 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | 132 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_5** | 165 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | 198 to 255 |
| **SII_PM_BARCODE_GS1_TRUNCATED** | | |
| | **SII_PM_BARCODE_MODULE_WIDTH_2** | 26 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | 39 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | 52 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_5** | 65 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | 78 to 255 |
| **SII_PM_BARCODE_GS1_LIMITED** | | |
| | **SII_PM_BARCODE_MODULE_WIDTH_2** | 20 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | 30 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | 40 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_5** | 50 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | 60 to 255 |
| **SII_PM_BARCODE_GS1_EXPANDED** | | |
| | **SII_PM_BARCODE_MODULE_WIDTH_2** | 68 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | 102 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | 136 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_5** | 170 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | 204 to 255 |

`hriPosition` HRI character print position
See "4.3.1(4)⑫ HRI character print position (`HriPosition`)" for available constants.

`hriFont` HRI character font
See "4.3.1(4)⑦ Character font (`CharacterFont`)" for available constants.

`alignment` Alignment
See "4.3.1(4)⑨ Alignment (`PrintAlignment`)" for available constants.

`nwRatio` N:W ratio
See "4.3.1(4)⑬ N:W ratio (`NwRatio`)" for available constants.
Depending on the specified `nwRatio` and `moduleSize`, the width of the wide element 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) |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

**Note** The quiet zone is not secured. Set the quiet zone in accordance with the standard of the barcode symbol.

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

## `printPDF417` Print PDF417

Prints PDF417.
The method of syntax (a) specifies the PDF417 symbol.
The method of syntax (b) is fixed to standard PDF417.

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 to the printer |
| | `errorCorrection` | Error correction level<br>See "4.3.1(4)⑭ Error correction level (`ErrorCorrection`)" for available constants. |
| | `row` | Number of rows (row)<br>The valid range is 0, 3 to 90.<br>When 0 is specified, the number of rows is automatically set. |
| | `column` | Number of columns in data area<br>The valid range is 0 to 30.<br>When 0 is specified, the number of columns in the data area is automatically set. |
| | `moduleSize` | Nominal fine element width<br>See "4.3.1(4)⑪ Module size (`ModuleSize`)" for available constants. |
| | `moduleHeight` | Module height (dot)<br>The valid range is 2 to 127.<br>When the module height is set smaller, some barcode scanners may not read it. Set 3 or more for normal use. |
| | `alignment` | Alignment<br>See "4.3.1(4)⑨ Alignment (`PrintAlignment`)" for available constants. |
| | `pdf417Symbol` | PDF417 symbol<br>See "4.3.1(4)⑮ PDF417 symbol (`Pdf417Symbol`)" for available constants. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

**Note** The quiet zone is not secured. Set the quiet zone in accordance with the standard of the barcode symbol.

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

## `printQRcode` Print QR Code

Prints QR Code.
The method of syntax (a) specifies QR Code Model.
The method of syntax (b) is fixed to QR Code Model 2.

Syntax

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

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

| | | |
|---|---|---|
| Parameter | `text` | Barcode data to send to the printer<br>The version is automatically set depending on the number of data bytes set with `text` in either syntax (a) and (b). |

| | | |
|---|---|---|
| | `errorCorrection` | Error correction level<br>See "4.3.1(4)⑭ Error correction level (`ErrorCorrection`)" for available constants. |
| | `moduleSize` | Module size<br>See "4.3.1(4)⑪ Module size (`ModuleSize`)" for available constants. |
| | `alignment` | Alignment<br>See "4.3.1(4)⑨ Alignment (`PrintAlignment`)" for available constants. |
| | `model` | QR Code Model<br>See "4.3.1(4)⑯ QR Code Model (`QrModel`)" for available constants. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

**Note** The quiet zone is not secured. Set the quiet zone in accordance with the standard of the barcode symbol.

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

## `printDataMatrix` Print Data Matrix

Prints Data Matrix.

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

Parameter

| | |
|---|---|
| `text` | Barcode data to send to the printer |
| `dataMatrixModule` | Number of Data Matrix modules<br>See "4.3.1(4)⑰ Data Matrix module (`DataMatrixModule`)" for available constants. |
| `moduleSize` | Module size<br>See "4.3.1(4)⑪ Module size (`ModuleSize`)" for available constants. |
| `alignment` | Alignment<br>See "4.3.1(4)⑨ Alignment (`PrintAlignment`)" for available constants. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

**Note** The quiet zone is not secured. Set the quiet zone in accordance with the standard of the barcode symbol.

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

## printMaxiCode — Print MaxiCode

Prints MaxiCode.

Syntax

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

Parameter

| | |
|---|---|
| `text` | Barcode data to send to the printer<br>• When `maxiCodeMode` is **SII_PM_MAXI_CODE_2**: Add service class (3 digits), country code (3 digits), and postal code (9 digits) to the beginning of the data.<br>• When `maxiCodeMode` is **SII_PM_MAXI_CODE_3**: Add service class (3 digits), country code (3 digits), and postal code (6 digits) to the beginning of the data. |
| `maxiCodeMode` | MaxiCode Mode<br>See "4.3.1(4)⑱ MaxiCode Mode (`MaxiCodeMode`)" for available constants. |
| `alignment` | Alignment<br>See "4.3.1(4)⑨ Alignment (`PrintAlignment`)" for available constants. |

Error

**`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

**Note**

The quiet zone is not secured. Set the quiet zone in accordance with the standard of the barcode symbol.

Reference

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

## printGS1DataBarStacked — Print GS1 Databar Stacked

Prints GS1 Databar Stacked.

Syntax

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

Parameter

| | |
|---|---|
| `text` | Barcode data to send to the printer<br>Enter 13 characters from '0' to '9'. The leading '01' is automatically added by the printer. The check digit is automatically calculated by the printer. |
| `moduleSize` | Module size<br>See "4.3.1(4)⑪ Module size (`ModuleSize`)" for available constants. |
| `alignment` | Alignment<br>See "4.3.1(4)⑨ Alignment (`PrintAlignment`)" for available constants. |

Error

**`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

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

## printGS1DataBarStackedOmnidirectional Print GS1 Databar Stacked Omni-directional

Prints GS1 Databar Stacked Omni-directional.

Syntax

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

Parameter

| | |
|---|---|
| `text` | Barcode data to send to the printer<br>Enter 13 characters from '0' to '9'. The leading '01' is automatically added by the printer. The check digit is automatically calculated by the printer. |
| `moduleHeight` | Barcode module height (number of modules)<br>The valid range is 33 to 255. |
| `moduleSize` | Module size<br>See "4.3.1(4)⑪ Module size (`ModuleSize`)" for available constants. |
| `alignment` | Alignment<br>See "4.3.1(4)⑨ Alignment (`PrintAlignment`)" for available constants. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

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

## printGS1DataBarExpandedStacked Print GS1 Databar Expanded Stacked

Prints GS1 Databar Expanded Stacked.

Syntax

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

Parameter

| | |
|---|---|
| `text` | Barcode data to send to the printer<br>Enter any number of characters using the following:<br>' ', '!', '"', '%', '&', ''', '(', ')', '*', '+', ',', '-', '.', '/', ':', ';', '<', '=', '>', '?', '_', '0' to '9', 'A' to 'Z', 'a' to 'z'<br>Enter '{1' for FNC1. Be sure to input the check digit because it is not automatically calculated by the printer. |
| `column` | Number of columns<br>Specify the number of columns in 1 line.<br>An even number from 2 to 20 is valid. |
| `moduleSize` | Module size<br>See "4.3.1(4)⑪ Module size (`ModuleSize`)" for available constants. |

| | | |
|---|---|---|
| | `alignment` | Alignment<br>See "4.3.1(4)⑨ Alignment (`PrintAlignment`)" for available constants. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

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

## `printAztecCode` Print Aztec Code

This method is not supported. When this method is executed, **`SIIPrinterException`** is thrown.

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

## `cutPaper` Cut paper

Selects enabled/disabled of the paper feed to the cut position and cuts the paper.

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

| | | |
|---|---|---|
| Parameter | `cuttingMethod` | Cutting method<br>See "4.3.1(4)⑲ Cutting method (`CuttingMethod`)" for available constants. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

## `feedPosition` Paper form feed

This method is not supported. When this method is executed, **`SIIPrinterException`** is thrown.

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

## `sendBinary` Send binary data

Sends binary data to the printer.

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

| | | |
|---|---|---|
| Parameter | `data` | Binary data to send to the printer<br>Data size that can be specified at one time is 256 KB (262144 bytes). |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

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

By sending printer command as binary data with this method, printer functions not supported in the library become available. However, this method does not support commands to get the response from the printer.

## sendDataFile — Send specified file

Sends file data.
The method of syntax (a), dithering can be specified.
The method of syntax (b), dithering is fixed to be disabled.

Syntax

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

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

Parameter

`fileName` File path of the data to send to the printer
The file size that can be specified is maximum 1 MB (1048576 bytes).
The file extensions that can be sent and the file transmission are described below.

- .bmp, .jpg, .jpeg, .png
  Data is sent to the printer as image file. Colored image file is converted to monochrome image by binarization and sent to the printer. Printing is performed in batch after mapping the image file on the memory of the printer.
- .txt
  Data is sent to the printer as text data. Text data format supports UTF-8. This method encodes the text data to printable text data based on **`internationalCharacter`** and **`codePage`**, and then sends it to the printer.
  This method does not add a line feed code at the end of the text data. In order to print to the end, add a line feed code at the end of the text data.
- .bin, .dat
  Data is sent to the printer as binary data without conversion.

`alignment` Alignment
It is valid when the extension of the file specified by `fileName` is .bmp, .jpg, .jpeg, .png, or .txt.
See "4.3.1(4)⑨ Alignment (`PrintAlignment`)" for available constants.

`dithering` Dithering
It is valid when the extension of the file specified by `fileName` is .bmp, .jpg, .jpeg, or .png.
See "4.3.1(4)① Dithering (`Dithering`)" for available constants.

Error

**`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

## printLogo — Print logo

Prints the registered logo.

Syntax

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

Parameter

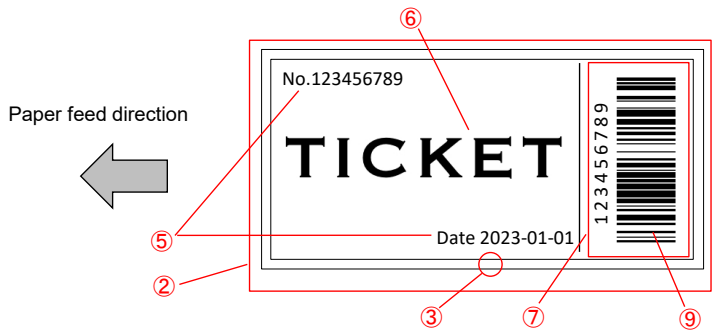
| | |
|---|---|
| `logoId` | ID of the logo to be printed (key code)<br>Specify the ID of the registered logo as a character string. |
| `alignment` | Alignment<br>See "4.3.1(4)⑨ Alignment (`PrintAlignment`)" for available constants. |

Error

**`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

③ 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

Starts page mode.

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

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

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

## exitPageMode — End page mode

Ends page mode and changes the mode to standard mode.

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

Error
**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

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

## setPageModeArea — Specify print area of page mode

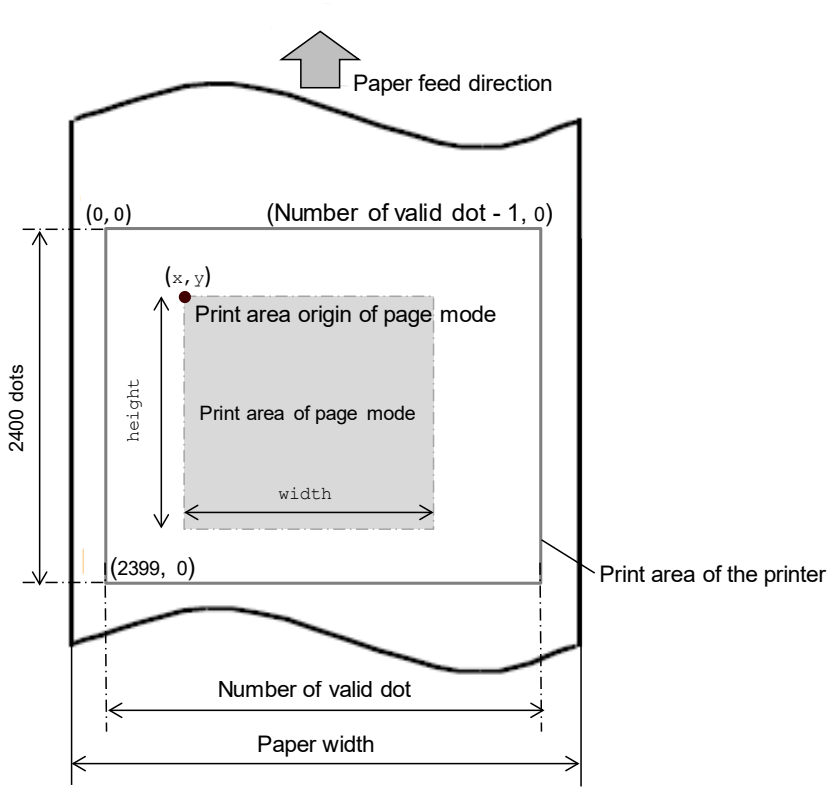
Specifies print area of page mode.

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<br>0 represents the left edge on the print area of the printer. |
| `y` | The vertical origin (dot) of the print area of page mode<br>The valid range is 0 to 2399.<br>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<br>The valid range is 1 to (2400−`y`). |

The valid range of `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 |
| | 512 | 512 | 0 to 511 | 1 to 512 |
| 58 mm | 432 | 432 | 0 to 431 | 1 to 432 |
| | 360 | 360 | 0 to 359 | 1 to 360 |

The number of valid dots differs depending on the memory switch setting.
See "RP-F10 SERIES THERMAL PRINTER USER'S GUIDE" for details of memory switch and the setting at shipping.

Error

**`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

Description

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

The print area of page mode can be specified when page mode is started by **`enterPageMode`** 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 `x` = 0, `y` = 0, `width` = number of a valid dot, `height` = 2400 after executing **`enterPageMode`**.

## setPageModeDirection — Specify print direction of page mode

Specifies print direction of page mode.

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

Parameter

| | |
|---|---|
| `direction` | Print direction<br>See "4.3.1(4)㉘ Print direction (`Direction`)" for available constants. |

Error
**`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

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

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

## setPageModeLineSpacing — Specify line spacing of page mode

Specifies line spacing of page mode.

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

Parameter

| | |
|---|---|
| `lineSpacing` | Line spacing (dot) of page mode<br>The valid range is 0 to 255. |

Error
**`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

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

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

## printPageMode — Print page mode

Prints the print data kept in page data buffer.

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

Parameter

| | |
|---|---|
| `cuttingMethod` | Cutting method<br>See "4.3.1(4)⑲ Cutting method (`CuttingMethod`)" for available constants. |

Error
**`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

Description
The print data is kept after printing. The print data is discarded at the timing of the following:

- Execute **`enterPageMode`**
- Execute **`disconnect`**
- Execute **`exitPageMode`**

## printPageModeText — Send text data of page mode

Maps the text data on the print area of page mode.

Syntax

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

Parameter

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

Error

**`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

Description

This method encodes the specified text data to printable text data based on **`internationalCharacter`** and **`codePage`**.

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

## printPageModeTextEx — Send format specified text data of page mode

Maps the format specified text data on the print area of page mode.

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<br>The valid range is 0 to 2399. |
| `startY` | The vertical reference point (dot) from the starting point<br>The valid range is 0 to 2399. |
| `text` | Text data<br>Data size that can be specified at 1 time is 16 KB (16384 bytes). |
| `bold` | Bold character<br>See "4.3.1(4)③ Bold print (`CharacterBold`)" for available constants. |
| `underline` | Underline<br>See "4.3.1(4)④ Underline (`CharacterUnderline`)" for available constants. |

| | | |
|---|---|---|
| | `reverse` | Reverse print<br>See "4.3.1(4)⑤ Reverse print (`CharacterReverse`)" for available constants. |
| | `font` | Font<br>See "4.3.1(4)⑦ Character font (`CharacterFont`)" for available constants. |
| | `scale` | Character scale<br>See "4.3.1(4)⑧ Character scale (`CharacterScale`)" for available constants. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

Description This method encodes the specified text data to printable text data based on **`internationalCharacter`** and **`codePage`**.

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

## `printPageModeBarcode` Print barcode of page mode

Maps the barcode on the print area of page mode.
The method of syntax (a) specifies the barcode data by character string.
The method of syntax (b) specifies the barcode data by character string and specifies N:W ratio of the barcode.
The method of syntax (c) specifies the barcode data by the array of bytes.
The method of syntax (d) is not supported.

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`
`barcodeSymbol:(BarcodeSymbol)barcodeSymbol`
`text:(NSString *)text`
`moduleSize:(ModuleSize)moduleSize;`

Parameter

| | |
|---|---|
| `startX` | The horizontal reference point (dot) from the starting point<br>The valid range is 0 to 2399. |
| `startY` | The vertical reference point (dot) from the starting point<br>The valid range is 0 to 2399. |
| `barcodeSymbol` | BarcodeSymbol<br>See "4.3.1(4)⑩ Barcode symbol (`BarcodeSymbol`) for available constants and correspondent syntax. |
| `text(data)` | Barcode data to send to the printer<br>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<br>JAN13 | 12 to 13 characters ' | '0' to '9' | |
| EAN8<br>JAN8 | 7 to 8 characters | '0' to '9' | |
| CODE39 | 1 to 150 characters | '0' to '9'<br>'A' to 'Z'<br>' ', '$,', '%', '+', '-', '.', '/' | Start code and stop code ('*') are automatically added. |
| CODE93 | 1 to 150 bytes | (0x00 to 0x2E) | Input data with 0x2F or more at the end. |
| CODE128 | 2 to 150 bytes | (0x00 to 0x66) | When inputting the start code (0x67 to 0x69) of the CODE128 code set.<br>Input data with 0x67 or more at the end. |
| | | (0x00 to 0x7F) | When starting with a CODE128 special code start code ("{A", "{B", "{C"). |
| ITF | 2 to 150 characters (However, an even number) | '0' to '9' | |
| CODABAR | 1 to 150 characters | '0' to '9'<br>'$', '+', '-', '.', '/', ':' | It is needed to specify one of 'A' to 'D' at the beginning and end. |
| EAN13 add-on<br>JAN13 add-on | Add-on 2: 14 to 15 characters<br>Add-on 5: 17 to 18 characters | '0' to '9' | |
| Customer Bar Code_JP | - | - | Not supported. |

| Barcode | Number of Data | Inputtable Data Character String (Data) | Remarks |
|---|---|---|---|
| GS1 Databar Omni-directional | 13 characters | '0' to '9' | Check digit is automatically added. |
| GS1 Databar Truncated | 13 characters | '0' to '9' | Check digit is automatically added. |
| GS1 Databar Limited | 13 characters | '0' to '9' | Check digit is automatically added. |
| GS1 Databar Expanded | 2 to 255 characters | ' ' to ""<br>'%' to '?'<br>'A' to 'Z'<br>'_'<br>'a' to 'z'<br>'{' | |

`moduleSize` Barcode width
See "4.3.1(4)⑪ Module size (`ModuleSize`)" for available constants.

`moduleHeight` Barcode height (dot)

- When `barcodeSymbol` is set to the following, the valid range is 1 to 255.
  **SII_PM_BARCODE_SYMBOL_UPC_A**
  **SII_PM_BARCODE_SYMBOL_UPC_E**
  **SII_PM_BARCODE_SYMBOL_EAN13**
  **SII_PM_BARCODE_SYMBOL_JAN13**
  **SII_PM_BARCODE_SYMBOL_EAN8**
  **SII_PM_BARCODE_SYMBOL_JAN8**
  **SII_PM_BARCODE_SYMBOL_CODE39**
  **SII_PM_BARCODE_SYMBOL_CODE93**
  **SII_PM_BARCODE_SYMBOL_CODE128**
  **SII_PM_BARCODE_SYMBOL_ITF**
  **SII_PM_BARCODE_SYMBOL_CODABAR**
  **SII_PM_BARCODE_SYMBOL_EAN13_ADDON**
  **SII_PM_BARCODE_SYMBOL_JAN13_ADDON**

- When `barcodeSymbol` is set to the following, the valid range is different by `barcodeSymbol` and `moduleSize`.

| `barcodeSymbol` | | |
|---|---|---|
| | `moduleSize` | **Valid Range** |
| **SII_PM_BARCODE_SYMBOL_GS1_OMNI_DIRECTIONAL** | | |
| | **SII_PM_BARCODE_MODULE_WIDTH_2** | 66 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | 99 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | 132 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_5** | 165 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | 198 to 255 |

| barcodeSymbol | | |
|---|---|---|
| | moduleSize | **Valid Range** |
| **SII_PM_BARCODE_SYMBOL_GS1_TRUNCATED** | | |
| | **SII_PM_BARCODE_MODULE_WIDTH_2** | 26 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | 39 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | 52 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_5** | 65 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | 78 to 255 |
| **SII_PM_SYMBOL_GS1_LIMITED** | | |
| | **SII_PM_BARCODE_MODULE_WIDTH_2** | 20 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | 30 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | 40 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_5** | 50 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | 60 to 255 |
| **SII_PM_BARCODE_SYMBOL_GS1_EXPANDED** | | |
| | **SII_PM_BARCODE_MODULE_WIDTH_2** | 68 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | 102 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | 136 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_5** | 170 to 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | 204 to 255 |

`hriPosition` HRI character print position
See "4.3.1(4)⑫ HRI character print position (`HriPosition`)" for available constants.

`hriFont` HRI character font
See "4.3.1(4)⑦ Character font (`CharacterFont`)" for available constants.

`nwRatio` N:W ratio
See "4.3.1(4)⑬ N:W ratio (`NwRatio`)" for available constants. Depending on specified `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) |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

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

**Note** Map the print data of the barcode not to overlap the other print data.
The quiet zone is not secured. Set the quiet zone in accordance with the standard of the barcode symbol.

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

## `printPageModePDF417` Print PDF417 of page mode

Maps PDF417 on the print area of page mode.
The method of syntax (a) specifies PDF417 symbol.
The method of syntax (b) is fixed to standard PDF417.

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<br>The valid range is 0 to 2399. |
| `startY` | The vertical reference point (dot) from the starting point<br>The valid range is 0 to 2399. |
| `text` | Barcode data |
| `errorCorrection` | Error correction level<br>See "4.3.1(4)⑭ Error correction level (`ErrorCorrection`)" for available constants. |
| `row` | The number of rows (row)<br>The valid range is 0, 3 to 90.<br>When 0 is specified, the number of rows is automatically set. |
| `column` | The number of columns in data area<br>The valid range is 0 to 30.<br>When 0 is specified, the number of columns in the data area is automatically set. |
| `moduleSize` | Nominal fine element width<br>See "4.3.1(4)⑪ Module size (`ModuleSize`)" for available constants. |

| | | |
|---|---|---|
| | `moduleHeight` | Module height (dot)<br>The valid range is 2 to 127.<br>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<br>See "4.3.1(4)⑮ PDF417 symbol (`Pdf417Symbol`)" for available constants. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

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

**Note** Map the print data of the barcode not to overlap the other print data.
The quiet zone is not secured. Set the quiet zone in accordance with the standard of the barcode symbol.

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

## `printPageModeQRcode` Print QR Code of page mode

Maps QR Code on the print area of page mode.
The method of syntax (a) specifies QR Code Model.
The method of syntax (b) is fixed to QR Code Model 2.

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<br>The valid range is 0 to 2399. |
| `startY` | The vertical reference point (dot) from the starting point<br>The valid range is 0 to 2399. |
| `text` | Barcode data<br>The version for either syntax (a) or (b) is automatically set depending on the number of data specified on `text`. |
| `errorCorrection` | Error correction level<br>See "4.3.1(4)⑭ Error correction level (`ErrorCorrection`)" for available constants. |
| `moduleSize` | Module size<br>See "4.3.1(4)⑪ Module size (`ModuleSize`)" for available constants. |
| `model` | QR Code Model<br>See "4.3.1(4)⑯ QR Code Model (`QrModel`)" for available constants. |

Error **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

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

**Note** Map the print data of the barcode not to overlap the other print data.
The quiet zone is not secured. Set the quiet zone in accordance with the standard of the barcode symbol.

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

## printPageModeDataMatrix — Print Data Matrix of page mode

Maps Data Matrix on the print area of page mode.

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<br>The valid range is 0 to 2399. |
| `startY` | The vertical reference point (dot) from the starting point<br>The valid range is 0 to 2399. |
| `text` | Barcode data |
| `dataMatrixModule` | The number of Data Matrix modules<br>See "4.3.1(4)⑰ Data Matrix module (`DataMatrixModule`)" for available constants. |
| `moduleSize` | Module size<br>See "4.3.1(4)⑪ Module size (`ModuleSize`)" for available constants. |

Error **SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

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

**Note** Map the print data of the barcode not to overlap the other print data.
The quiet zone is not secured. Set the quiet zone in accordance with the standard of the barcode symbol.

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

## printPageModeMaxiCode — Print MaxiCode of page mode

Maps MaxiCode on the print area of page mode.

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<br>The valid range is 0 to 2399. |
| `startY` | The vertical reference point (dot) from the starting point<br>The valid range is 0 to 2399. |
| `text` | Barcode data<br>• When **`maxiCodeMode`** is **SII_PM_MAXI_CODE_2**<br>Add the service class (3 digits), the country code (3 digits), and the postal code (9 digits) to the beginning of the data.<br>• When **`maxiCodeMode`** is **SII_PM_MAXI_CODE_3**<br>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<br>See "4.3.1(4)⑱ MaxiCode Mode (`MaxiCodeMode`)" for available constants. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

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

**Note** Map the print data of the barcode not to overlap the other print data.
The quiet zone is not secured. Set the quiet zone in accordance with the standard of the barcode symbol.

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

## `printPageModeGS1DataBarStacked` Print GS1 Databar Stacked of page mode

Maps GS1 Databar Stacked on the print area of page mode.

Syntax

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

Parameter

| | |
|---|---|
| `startX` | The horizontal reference point (dot) from the starting point<br>The valid range is 0 to 2399. |
| `startY` | The vertical reference point (dot) from the starting point<br>The valid range is 0 to 2399. |
| `text` | Barcode data<br>Enter 13 characters from '0' to '9'. The leading '01' is automatically added by the printer. The check digit is automatically calculated by the printer. |
| `moduleSize` | Module size<br>See "4.3.1(4)⑪ Module size (`ModuleSize`)" for available constants. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

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

**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.

## printPageModeGS1DataBarStackedOmnidirectional
## Print GS1 Databar Stacked Omni-directional of page mode

Maps GS1 Databar Stacked Omni-directional on the print area of page mode.

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

Parameter

| | |
|---|---|
| `startX` | The horizontal reference point (dot) from the starting point<br>The valid range is 0 to 2399. |
| `startY` | The vertical reference point (dot) from the starting point<br>The valid range is 0 to 2399. |
| `text` | Barcode data<br>Enter 13 characters from '0' to '9'. The leading '01' is automatically added by the printer. The check digit is automatically calculated by the printer. |
| `moduleHeight` | Barcode module height (the number of the modules)<br>The valid range is 33 to 255. |
| `moduleSize` | Module size<br>See "4.3.1(4)⑪ Module size (`ModuleSize`)" for available constants. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

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

**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.

## printPageModeGS1DataBarExpandedStacked
## Print GS1 Databar Expanded Stacked of page mode

Maps GS1 Databar Expanded Stacked on the print area of page mode.

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

Parameter

| | |
|---|---|
| `startX` | The horizontal reference point (dot) from the starting point<br>The valid range is 0 to 2399. |

| | | |
|---|---|---|
| | `startY` | The vertical reference point (dot) from the starting point<br>The valid range is 0 to 2399. |
| | `text` | Barcode data<br>Enter any number of characters using the following:<br>' ', '!', '"', '%', '&', '', '(', ')', '*', '+', ',', '-', '.', '/', ':', ';', '<', '=', '>', '?', '_', '0' to '9', 'A' to 'Z', 'a' to 'z'.<br>Enter '{1' to FNC1. Be sure to input the check digit because it is not automatically calculated by the printer. |
| | `column` | The number of columns<br>Specifies the number of the columns in 1 line.<br>The valid range is the even number from 2 to 20. |
| | `moduleSize` | Module size<br>See "4.3.1(4)⑪ Module size (`ModuleSize`)" for available constants. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

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

**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.

## `printPageModeAztecCode` Print Aztec Code of page mode

This method is not supported. When this method is executed, **`SIIPrinterException`** is thrown.

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

## `sendPageModeBinary` Send binary data of page mode

Maps binary data on the print area of page mode.

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

| | | |
|---|---|---|
| Parameter | `binary` | Binary data<br>Data size that can be specified at 1 time is 16 KB (16384 bytes). |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

Description Start page mode by **`enterPageMode`** 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.

**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

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

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<br>The valid range is 0 to 2399. |
| `startY` | The vertical reference point (dot) from the starting point<br>The valid range is 0 to 2399. |
| `fileName` | File path of the data<br>The maximum file size that can be specified is 1 MB (1048576 bytes).<br>The image files that can be sent are .bmp, .jpg, .jpeg, .png<br>Colored image file is converted to monochrome image by binarization and registered. |
| `dithering` | Dithering<br>See "4.3.1(4)① Dithering (`Dithering`)" for available constants. |

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

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

## printPageModeRectangle — Draw rectangle image of page mode

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

Syntax

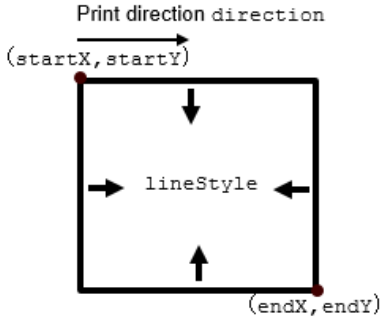
```
- (void) printPageModeRectangle:(NSInteger)startX
                         startY:(NSInteger)startY
                           endX:(NSInteger)endX
                           endY:(NSInteger)endY
                      lineStyle:(LineStyle)lineStyle;
```

| | |
|---|---|
| `startX` | The horizontal reference point (dot) from the starting point<br>The valid range is 0 to 2399. |
| `startY` | The vertical reference point (dot) from the starting point<br>The valid range is 0 to 2399. |
| `endX` | The horizontal reference point (dot) from the ending point<br>The valid range is 0 to 2399. |
| `endY` | The vertical reference point (dot) from the ending point<br>The valid range is 0 to 2399. |
| `lineStyle` | Line style<br>See "4.3.1(4)㉙ Line style (`LineStyle`)" for available constants. |

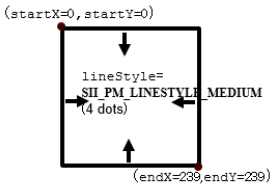
Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

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

The rectangle is mapped to `direction` of **`setPageModeDirection`** 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<br>`startY` 0<br>`endX` 239<br>`endY` 239<br>`lineStyle` **SII_PM_LINESTYLE_MEDIUM** |



## `printPageModeLine` Print ruled line of page mode

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

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<br>The valid range is 0 to 2399. |
| `startY` | The vertical reference point (dot) from the starting point<br>The valid range is 0 to 2399. |
| `endX` | The horizontal reference point (dot) from the ending point<br>The valid range is 0 to 2399. |
| `endY` | The vertical reference point (dot) from the ending point<br>The valid range is 0 to 2399. |

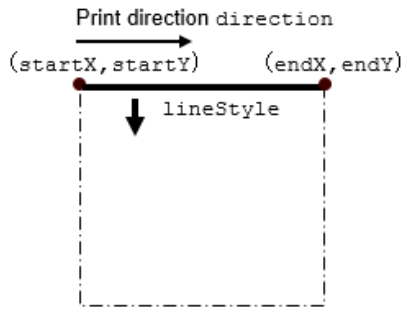
`lineStyle` Line style
See "4.3.1(4)㉙ Line style (`LineStyle`)" for available constants.

Error **`SIIPrinterException`** is thrown when an error occurs while this method is being called.
See "**4.3.3 `SIIPrinterException` Class**" for details on the error.

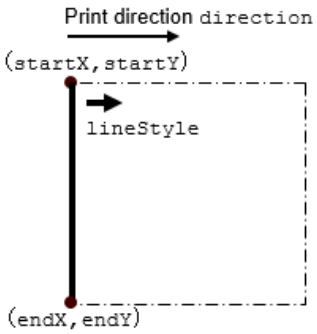
Description Start page mode by **`enterPageMode`** before executing this method.

A diagonal stroke cannot be drawn by this method.

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

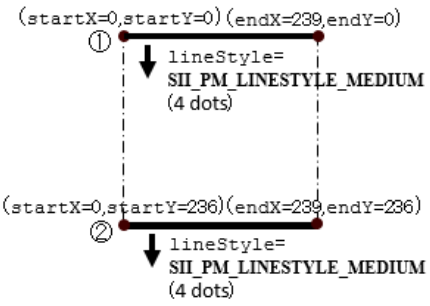



**Mapping direction of horizontal ruled line**

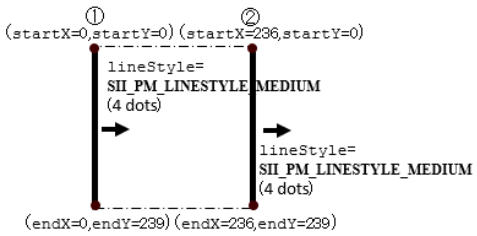



**Mapping direction of vertical ruled line**

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<br>startX 0<br>startY 0<br>endX 0<br>endY 239<br>lineStyle **SII_PM_LINESTYLE_MEDIUM**<br><br>②<br>startX 236<br>startY 0<br>endX 236<br>endY 239<br>lineStyle **SII_PM_LINESTYLE_MEDIUM** |

## printPageModeLogo — Print logo of page mode

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

Syntax

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

Parameter

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

Error

**SIIPrinterException** is thrown when an error occurs while this method is being called.
See "**4.3.3 SIIPrinterException Class**" for details on the error.

Description

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

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

## sendTimeout — Get/Set send timeout period

Gets or sets the timeout period in sending data.

Syntax `@property NSInteger **sendTimeout;**`

Valid range 100 to 300000 (millisecond: ms)
When a value is specified less than 100, the period is set to 100 ms.
When the value is specified more than 300000, 300000 ms is set.

Default 10000

Description This method can get or set the timeout period regardless of whether **isConnect** is YES or NO.
The set timeout period becomes effective at the next data sending.

## receiveTimeout — Get/Set receive timeout period

Gets or sets the timeout period in receiving data.

Syntax `@property NSInteger **receiveTimeout;**`

Valid range 100 to 300000 (millisecond: ms)
When a value is specified less than 100, the period is set to 100 ms.
When the value is specified more than 300000, 300000 ms is set.

Default 10000

Description This method can get or set the timeout period regardless of whether **isConnect** is YES or NO.
The set timeout period becomes effective at the next data receiving.

## internationalCharacter — Get/Set international character set

Gets or sets the value of international character set.

Syntax `@property NSInteger **internationalCharacter;**`

Description See "4.3.1(3)⑤ International character set" for configurable constants. When an invalid value is specified, it is ignored.
When this property is not set, the international character set is as follows depending on the language setting of iOS device.

When the language setting of iOS device is Japanese: **SII_PM_COUNTRY_JAPAN**
When the language setting of iOS device is other than Japanese:
**SII_PM_COUNTRY_USA**

When text data is sent by **sendText**, **sendTextEx**, **sendDataFile**, **printPageModeText**, or **printPageModeTextEx**, the print result of the following character codes varies. See "Appendix A Character Set" for details of the characters to be printed.

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

## codePage — Get/Set codepage

Gets or sets the value of codepage.

Syntax
```
@property NSInteger codePage;
```

Description
See "4.3.1(3)⑥ Codepage" for configurable constants. When an invalid value is specified, it is ignored.
When this property is not set, the codepage is as follows depending on the language setting of iOS device.

When the language setting of iOS device is Japanese:
**SII_PM_CODE_PAGE_KATAKANA**
When the language setting of iOS device is other than Japanese:
**SII_PM_CODE_PAGE_1252**

The encorder used for sending the text data by **sendText**, **sendTextEx**, **sendDataFile**, **printPageModeText**, or **printPageModeTextEx** is changed.
See "Appendix A Character Set" for characters to be printed.

## printerModel — Get printer model

Gets the value of the connecting printer model.

Syntax
```
@property(readonly) NSInteger printerModel;
```

Default
-1

Return value
See "4.3.1(3)① Printer model" for available constants.
When **isConnect** is NO, -1 is returned.

## portType — Get connecting port type

Gets the value of the port type used for connection with the printer.

Syntax
```
@property(readonly) NSInteger portType;
```

Default
-1

Return value
See "4.3.1(3)② Port type" for available constants.
When **isConnect** is NO, -1 is returned.

## isConnect — Verify connection state with printer

Verifies connection state with the printer.

Syntax
```
@property(readonly) BOOL isConnect;
```

Return value
YES Connected to the printer
NO Not connected to the printer

Description
This property retains the **connect** state as a BOOL value.
When **connect** succeeds, this property is YES. After **connect**, when **disconnect** succeeds, this property becomes NO.

## socketKeepingTime — Get/Set socket keeping time

Gets or sets the socket keeping time.

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

Valid range 60000 to 300000 (millisecond: ms)
When a value is specified less than 60000, time is set to 60000 ms.
When a value is specified more than 300000, time is set to 300000 ms.

Default 300000

Description This method can get or set the socket keeping time regardless of whether **isConnect** is YES or NO.

For the socket keeping time, specify a time equal to Receive Timeout of the printer to be connected. The setting of Receive Timeout can be changed in "SII RP Utility" with the iOS app on the App Store.

The set socket keeping time becomes effective at the next **connect** execution.

## delegate — Register delegate

Registers a delegate object that receives notifications from the printer.

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

Description 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 nil is specified for this property, the notifications of the printer status and barcode data are stopped.

### 4.3.2 `SIIPrinterInfo` Class

This class stores the printer information found by printer searching method. It gets the printer model name, MAC address, and IP address from the found printer information.

(1) Method List

Methods provided by **`SIIPrinterInfo`** class are shown in the following table.

| Name | Description |
|---|---|
| **`SIIPrinterInfo`** | Constructor of the printer information class |

(2) Property List

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

| Name | Access | Description |
|---|---|---|
| **`name`** | R | Get printer model name |
| **`mac`** | R | Get MAC address |
| **`ip`** | R | Get IP address |

(3) Method Details

`SIIPrinterInfo` Constructor

Syntax **`SIIPrinterInfo`**

Description This method stores the printer information found by **`startDiscoveryPrinter`**.

(4) Property Details

## name — Get printer model name

Syntax `@property NSString **name;`

Description This property gets the printer model name from the printer information found by **startDiscoveryPrinter**.

## mac — Get MAC address

Syntax `@property NSString **mac;`

Description This property gets the MAC address from the printer information found by **startDiscoveryPrinter**.

## ip — Get IP address

Syntax `@property NSString **ip;`

Description This property gets the IP address from the printer information found by **startDiscoveryPrinter**.

### 4.3.3 `SIIPrinterException` Class

(1) Method List

Methods provided by **`SIIPrinterException`** class are shown in the following table.

| Name | Description |
|---|---|
| **`SIIPrinterException`** | Constructor |

(2) Property List

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

| Name | Access | Description |
|---|---|---|
| **`errorCode`** | R | Get error code |
| **`errorMessage`** | R | Get error message |

(3) Constant List

① Error code

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

| Constant Name | Description | Value |
|---|---|---|
| **SII_PM_ERROR_ACCESS_DENIED** | Failed to get the handle.[*1] | -1 |
| | An unavailable port was specified. | |
| | An unsupported method was specified. | |
| **SII_PM_ERROR_SHARING_VIOLATION** | An already opened port was specified. | -11 |
| **SII_PM_ERROR_PORT_NOT_OPENED** | The port is not open. | -12 |
| **SII_PM_ERROR_DEVICE_NOT_CONNECTED** | There is a problem with the Bluetooth connection between the iOS device and the printer. | -21 |
| **SII_PM_ERROR_OFFLINE** | Disconnected state or the printer is offline. | -22 |
| **SII_PM_ERROR_EXTERNAL_DEVICE_NOT_CONNECTED** | Display is not connected. | -23 |
| **SII_PM_ERROR_DEVICE_INITIALIZE_FAILED** | Failed to change the printer settings. Data sending to the printer is not completed within the send timeout period, or data receiving from the printer is not completed within the receive timeout period. | -31 |
| **SII_PM_ERROR_DATA_SIZE_ZERO** | 0-byte data was specified. | -101 |
| **SII_PM_ERROR_OVER_MAX_DATA_SIZE** | Maximum data size is exceeded. | -102 |
| **SII_PM_ERROR_DATA_SIZE_INVALID** | Data size is invalid. | -103 |
| **SII_PM_ERROR_ENCODE_FAILED** | An error occurred in encoding text data.[*1] | -111 |
| **SII_PM_ERROR_TIMEOUT** | Send timeout occurred. | -201 |
| | Receive timeout occurred. | |
| **SII_PM_ERROR_FILE_NOT_FOUND** | The specified file is not found. | -301 |
| **SII_PM_ERROR_FILE_USED** | The specified file is in use by another process. | -302 |
| **SII_PM_ERROR_FILE_INVALID** | The specified file is invalid. | -303 |
| **SII_PM_ERROR_LOW_MEMORY** | Memory shortage occurred when loading image file. | -311 |
| **SII_PM_ERROR_OVER_MAX_IMAGE** | Either or both of width and height of image file exceeds the number of printable maximum dots. | -312 |
| **SII_PM_ERROR_LOGO_NOT_DEFINED** | The logo is not registered. | -313 |
| **SII_PM_ERROR_LOW_USER_AREA** | Remaining user area is insufficient. | -401 |
| **SII_PM_ERROR_LOW_EXTERNAL_RAM** | Remaining RAM capacity is insufficient. | -402 |

| Constant Name | Description | Value |
|---|---|---|
| **SII_PM_ ERROR_NOT_REGISTERD** | The template is not registered.<br>The image data is not registered.<br>The slide data is not registered.<br>The optional font is not registered.<br>The user-defined character is not registered. | -403 |
| **SII_PM_ERROR_NOT_UNREGISTERD** | The template is not deleted.<br>The image data is not deleted.<br>The slide data is not deleted.<br>The optional font is not deleted.<br>The user-defined character is not deleted. | -404 |
| **SII_PM_ERROR_INVALID_NO** | The specified value for the logo ID is invalid. | -501 |
| **SII_PM_ERROR_INVALID_DATA** | The specified data is invalid. | -503 |
| **SII_PM_ERROR_PAGE_MODE_SPECIFIED** | Page mode is specified. | -511 |
| **SII_PM_ERROR_PAGE_MODE_NOT_SPECIFIE** | Page mode is not specified. | -512 |
| **SII_PM_ERROR_INVALID_PARAM** | The specified parameter is invalid. | -9999 |

*1: Abnormal processing might have occurred.

(4) Method Details

## SIIPrinterException — Constructor

This is the exception class that is thrown when API of **SIIPrinterManager** class is called.

Syntax **SIIPrinterException**

(5) Property Details

## errorCode — Get error code

Gets the error code of the thrown exception.

Syntax `@property NSInteger` **errorCode;**

Return value See "4.3.3(3) Constant List".

## errorMessage — Get error message

Gets the error message of the thrown exception.

Syntax `@property NSString` ***errorMessage;**

Description A character string that supplements the contents of **errorCode** can be retrieved.

### 4.3.4 `SIIPrinterManagerDelegate` Protocol

(1) Method List

Methods provided by **`SIIPrinterManagerDelegate`** protocol are shown in the following table.

| Name | Description |
|---|---|
| **`didStatusChange`** | Notify printer status |
| **`didBarcodeScannerReadData`** | Receipt notify of barcode data |
| **`didBarcodeScannerChangedOnline`** | Connection notify of barcode scanner |
| **`didBarcodeScannerChangedOffline`** | Disconnection notify of barcode scanner |

(2) Method Details

`didStatusChange` Notify printer status

Notifies changes in printer status.

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`** is executed.
・When the printer status is changed.

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

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

When communication with the printer is disconnected, this method notifies 0x80000000. After disconnection from the printer, the library attempts to resume communication with the printer until **`disconnect`** is executed. When communication with the printer becomes possible, this method notifies the latest printer status.
See **`getStatus`** for description of the printer status.

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

## didBarcodeScannerReadData — Receipt notify of barcode data

Notifies the receipt of the barcode data.

Syntax
```
- (void) didBarcodeScannerReadData
            :(SIIPrinterManager *)printerManagerdata:(NSData *)data;
```

Parameter

| | |
|---|---|
| printerManager | Calling SIIPrinterManager object |
| data | Received barcode data |

Description

This method is called when the barcode scanner connected to the printer scans a barcode.

Receipt of barcode data is notified when **isConnect** is YES and the printer is connecting.

Do not execute the APIs of SIIPrinterManager within this method.

## didBarcodeScannerChangedOnline — Connection notify of barcode scanner

Notifies the connection of the barcode scanner.

Syntax
```
- (void) didBarcodeScannerChangedOnline
                        :(SIIPrinterManager *)printerManager;
```

Parameter

| | |
|---|---|
| printerManager | Calling SIIPrinterManager object |

Description

This method is called at the following timing.
- When the barcode scanner is in connected state with the printer and **connect** is executed.
- When the barcode scanner is connected to the printer.

The connection of the barcode scanner is notified when **isConnect** is YES and the printer is connecting.

Do not execute the APIs of SIIPrinterManager within this method.

## didBarcodeScannerChangedOffline — Disconnection notify of barcode scanner

Notifies the disconnection of the barcode scanner.

Syntax
```
- (void) didBarcodeScannerChangedOffline
                                :(SIIPrinterManager *)printerManager;
```

Parameter

| | |
|---|---|
| printerManager | Calling SIIPrinterManager object |

Description

This method is called at the following timing.
- When the barcode scanner is in unconnected state with the printer and **connect** is executed.
- When the barcode scanner is disconnected from the printer.

The disconnection of the barcode scanner is notified when **isConnect** is YES.

Do not execute the APIs of SIIPrinterManager within this method.

### 4.3.5 `SIISmartLabelManager` Class

**`SIISmartLabelManager`** class provides the function to covert the label file (*.sl) created using Smart Label Creator into the printable data from the printer.

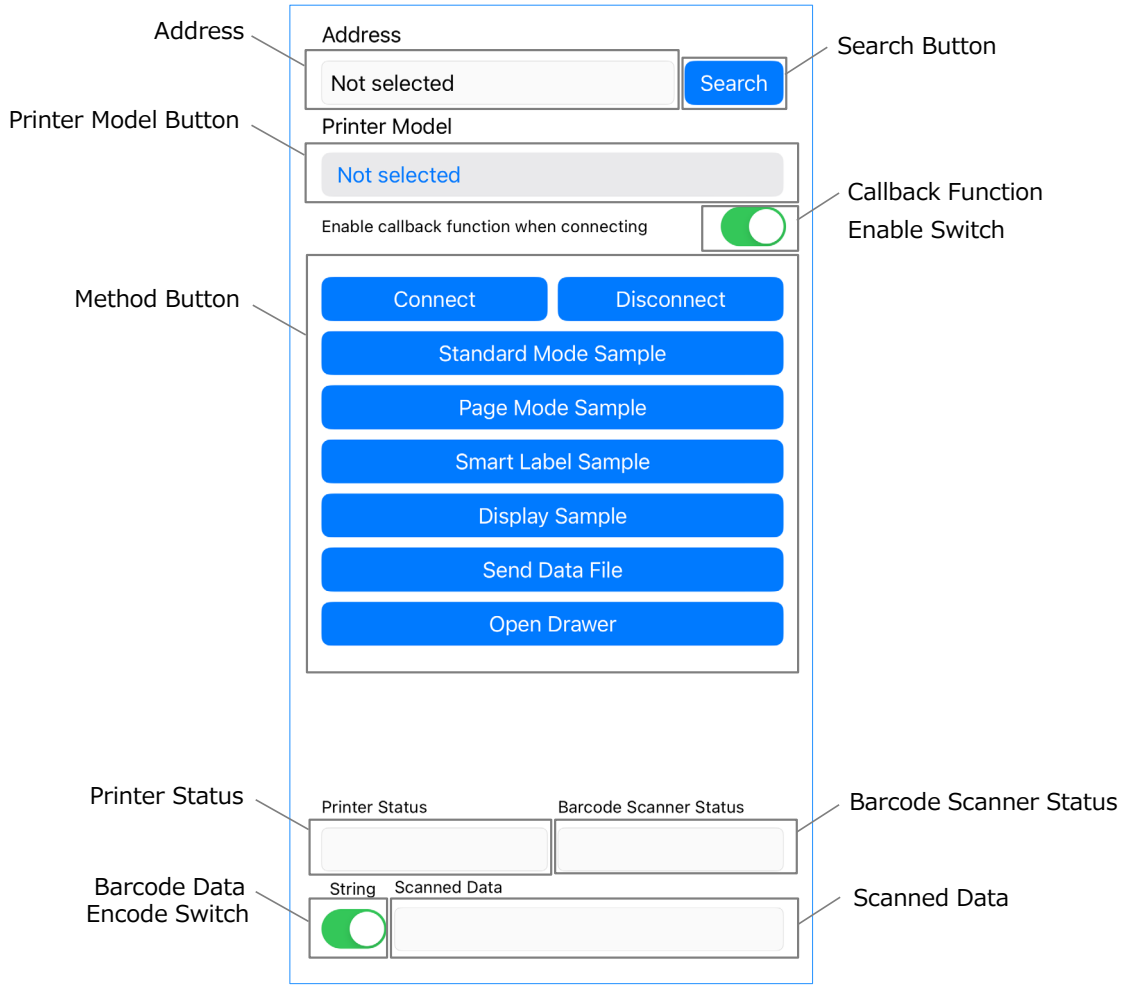
Do not use this class because it is not supported.

# Chapter 5
# Sample Program

This chapter describes the sample program provided by SII print class library.

## 5.1 Screen Layout

SII print class library includes SiiLibSample, a sample program in Xcode project format. This section describes the screen of SiiLibSample.



| Item | Description |
|---|---|
| Address | Displays the information about the selected printer. |
| Printer Model Button | Specifies the printer model.<br>When tapping [Printer Model Button], a list of printer models is displayed.<br>By selecting from the list, the printer model can be entered.<br>When the printer is selected from the printer search screen, the printer model is automatically displayed. |
| Search Button | Starts searching for printers.<br>Transits to the printer search view. A list of the searched printers is displayed. The printer is selected by tapping the searched printer and returns to the main view. |
| Callback Function Enable Switch | Select whether to enable the callback function when connecting to the printer.<br>On : Starts the callback function when connecting.<br>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.<br>When [Callback Function Enable Switch] is On, the latest status is displayed. |
| Barcode Scanner Status | Displays the connection status of the barcode scanner.<br>Online: The barcode scanner is connected.<br>Offline: The barcode scanner is unconnected. |
| Barcode Data Encode Switch | Selects the conversion of barcode data read by the barcode scanner.<br>On: Encodes the scanned binary value into the character strings and displays.<br>Off: Displays the scanned binary value as it is. |
| Scanned Data | Displays the barcode data scanned through the barcode scanner. |

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

## 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.

# 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. 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.

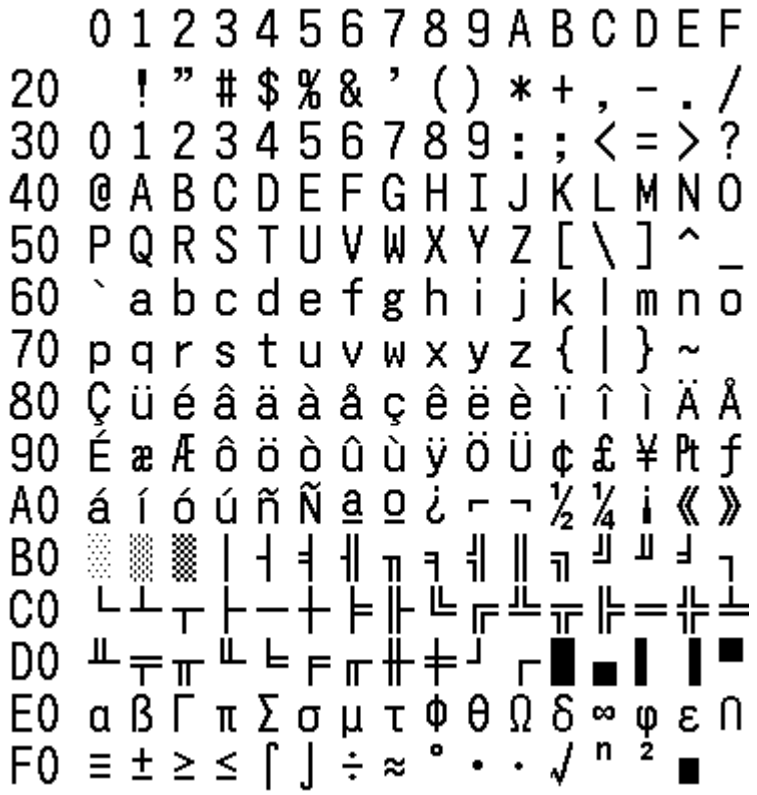



**Figure A-1 SII_PM_CODE_PAGE_437 (USA, Standard Europe)**

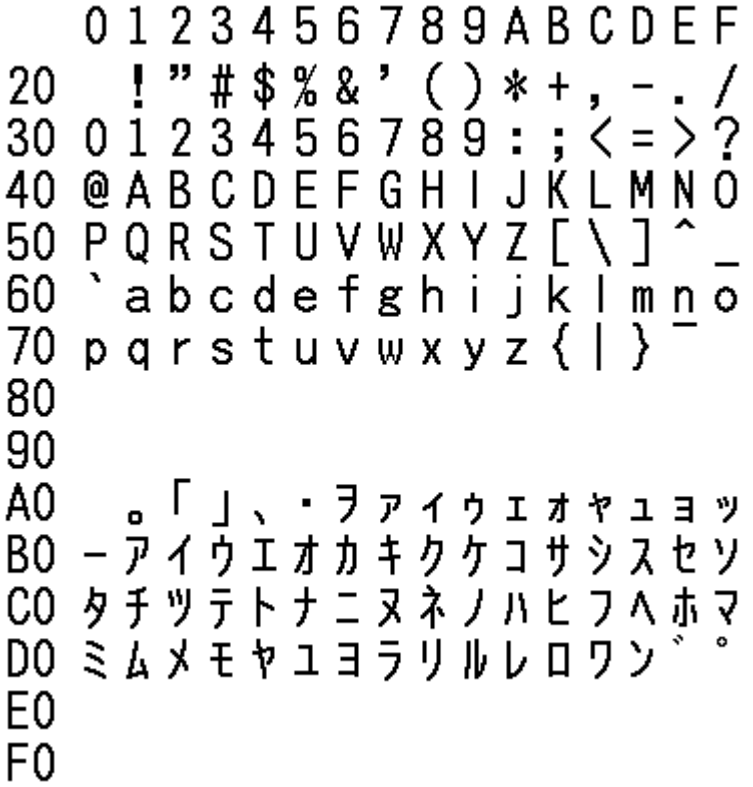



**Figure A-2 SII_PM_CODE_PAGE_KATAKANA**

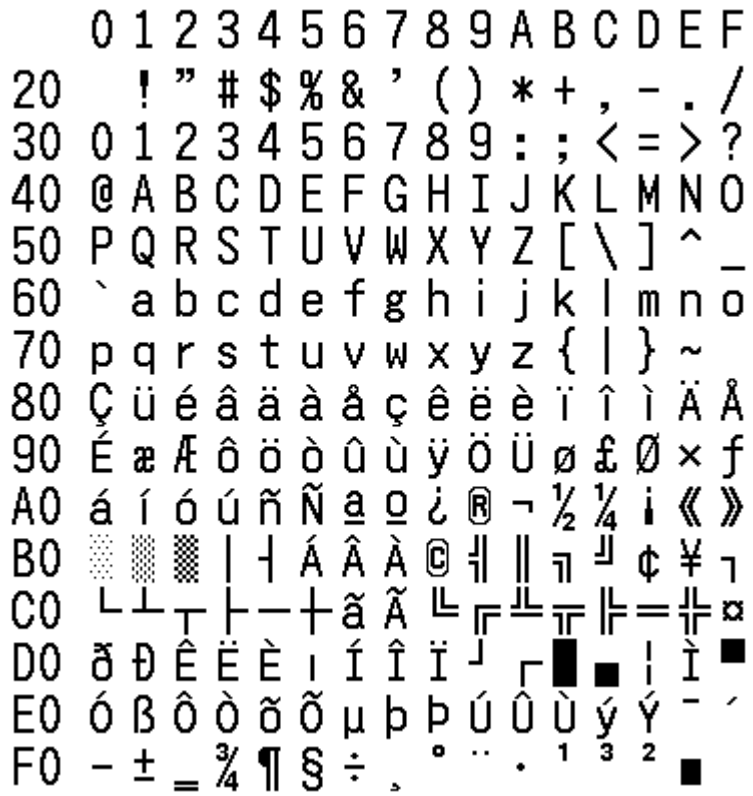



**Figure A-3 SII_PM_CODE_PAGE_850 (Multilingual)**

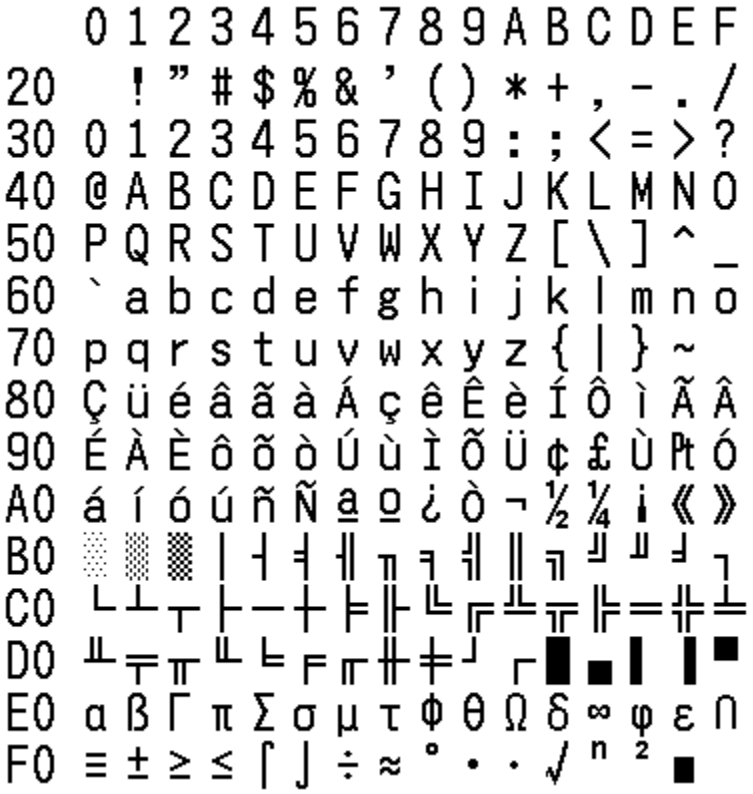



**Figure A-4 SII_PM_CODE_PAGE_860 (Portuguese)**

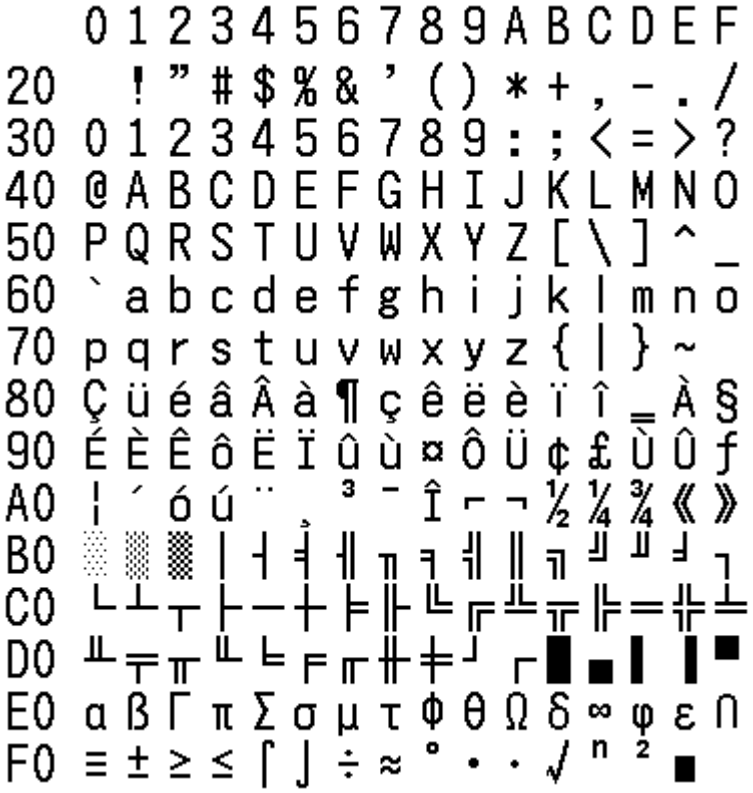



**Figure A-5 SII_PM_CODE_PAGE_863 (Canadian-French)**

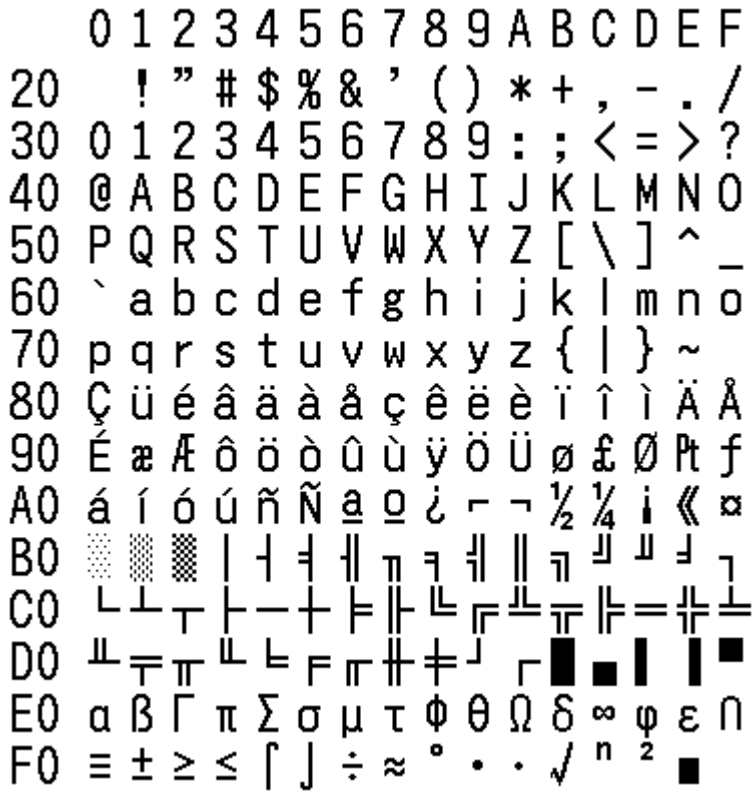



**Figure A-6 SII_PM_CODE_PAGE_865 (Nordic)**

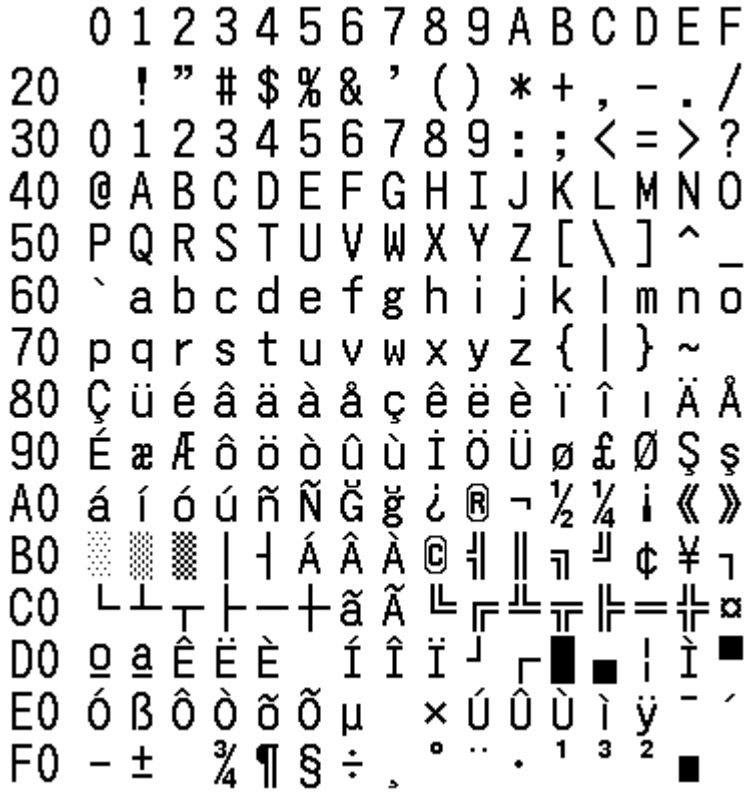



**Figure A-7 SII_PM_CODE_PAGE_857 (Turkish)**

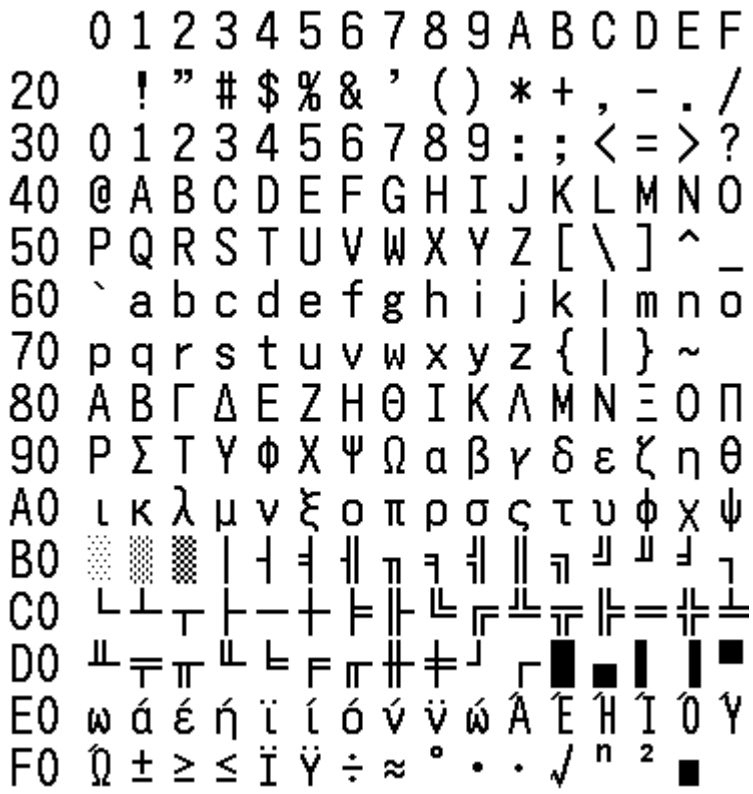



**Figure A-8 SII_PM_CODE_PAGE_737 (Greek)**

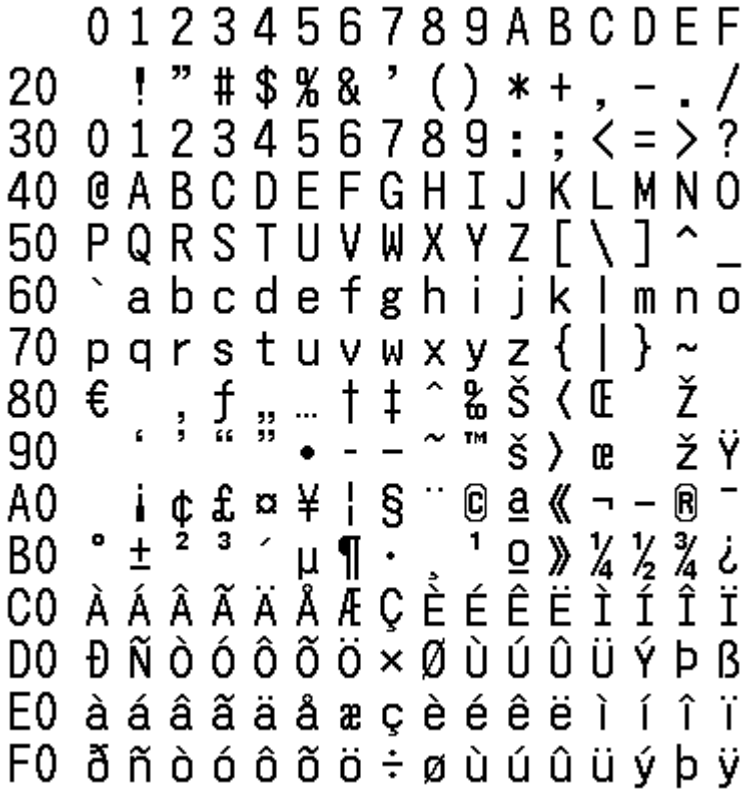



**Figure A-9 SII_PM_CODE_PAGE_1252 (Latin)**

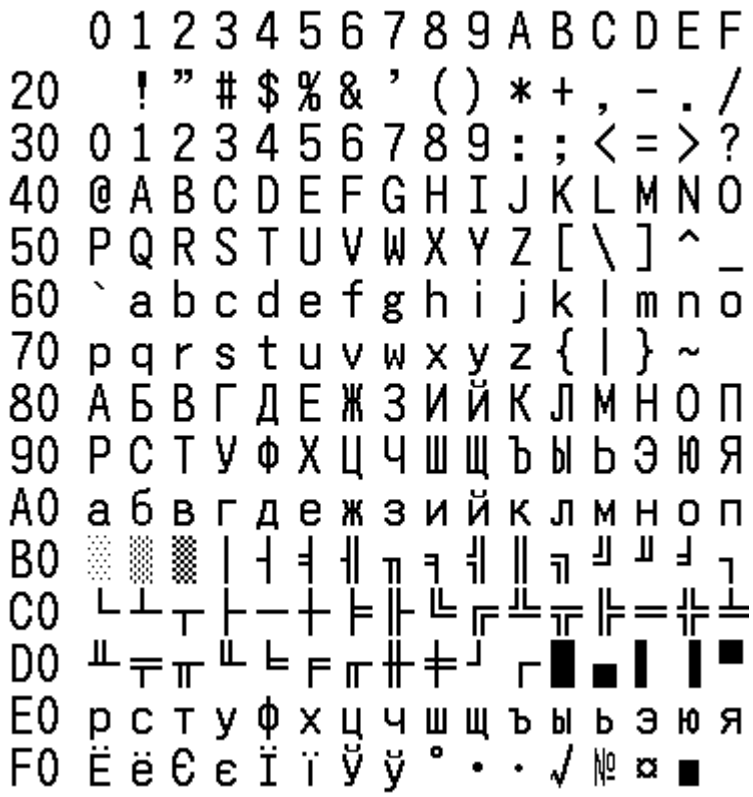



**Figure A-10 SII_PM_CODE_PAGE_866 (Russian)**

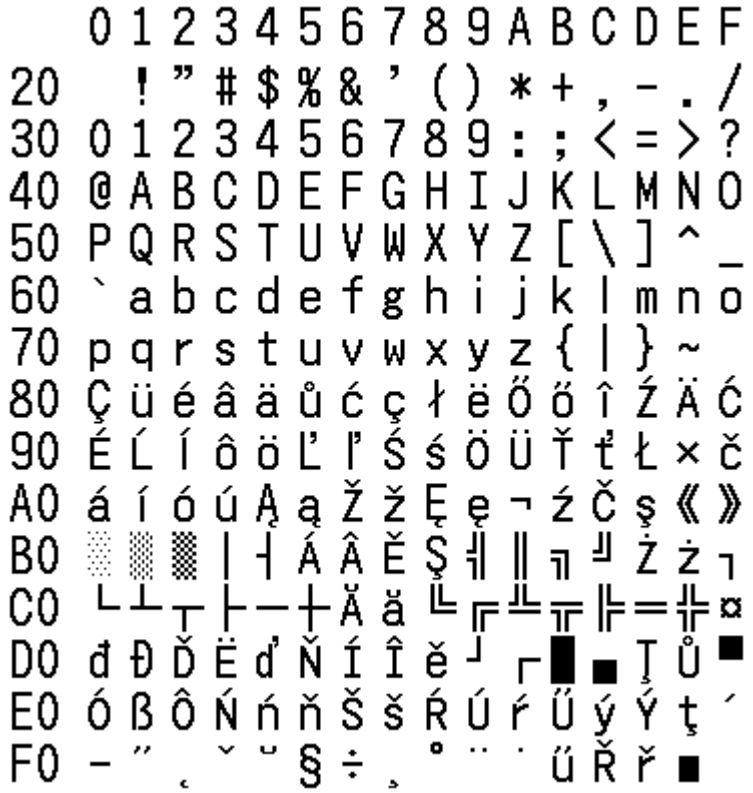



**Figure A-11 SII_PM_CODE_PAGE_852 (Eastern Europe)**

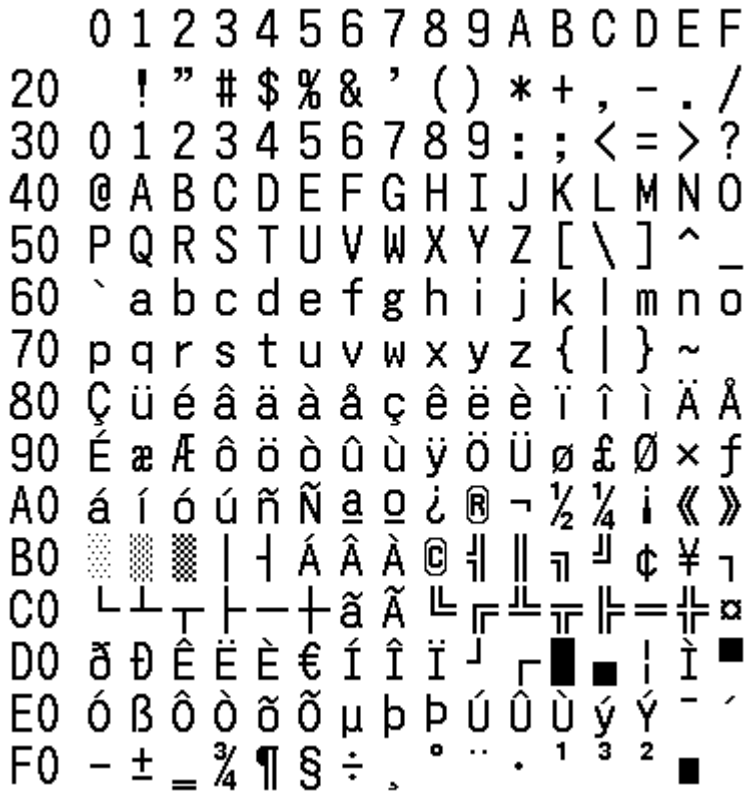



**Figure A-12 SII_PM_CODE_PAGE_858 (Euro)**

**Figure A-13 SII_PM_CODE_PAGE_855 (Cyrillic)**

**Figure A-14 SII_PM_CODE_PAGE_864 (Arabic)**

**Figure A-15 SII_PM_CODE_PAGE_1250 (Central European)**

**Figure A-16 SII_PM_CODE_PAGE_1251 (Cyrillic)**

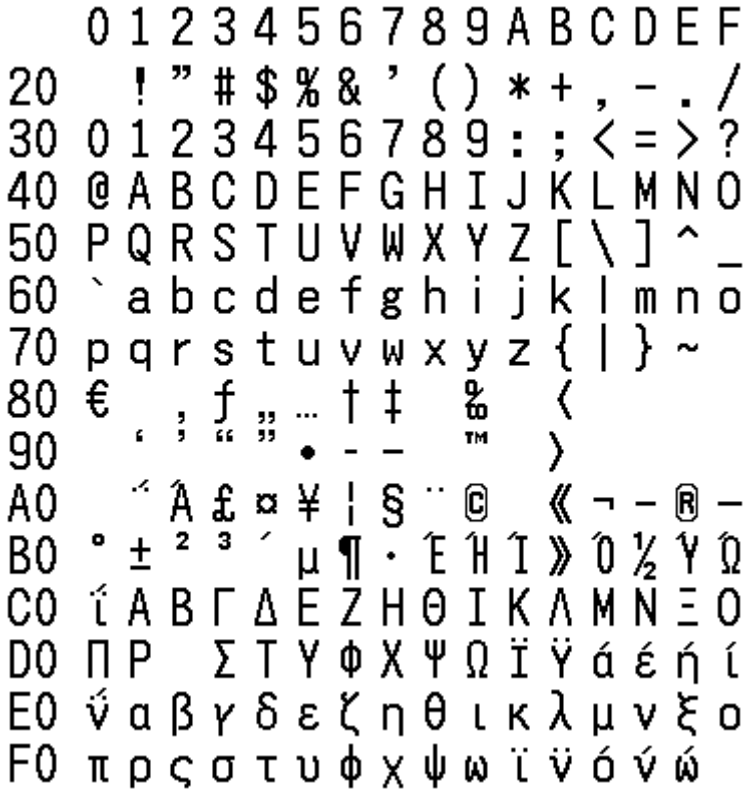



**Figure A-17 SII_PM_CODE_PAGE_1253 (Greek)**

**Figure A-18 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-19 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** | 190 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | 285 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | 380 |
| | **SII_PM_BARCODE_MODULE_WIDTH_5** | 475 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | 570 |
| **SII_PM_BARCODE_UPC_E** | **SII_PM_BARCODE_MODULE_WIDTH_2** | 102 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | 153 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | 204 |
| | **SII_PM_BARCODE_MODULE_WIDTH_5** | 255 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | 306 |
| **SII_PM_BARCODE_EAN13** | **SII_PM_BARCODE_MODULE_WIDTH_2** | 190 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | 285 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | 380 |
| | **SII_PM_BARCODE_MODULE_WIDTH_5** | 475 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | 570 |
| **SII_PM_BARCODE_JAN13** | **SII_PM_BARCODE_MODULE_WIDTH_2** | 190 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | 285 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | 380 |
| | **SII_PM_BARCODE_MODULE_WIDTH_5** | 475 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | 570 |
| **SII_PM_BARCODE_EAN8** | **SII_PM_BARCODE_MODULE_WIDTH_2** | 134 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | 201 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | 268 |
| | **SII_PM_BARCODE_MODULE_WIDTH_5** | 335 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | 402 |
| **SII_PM_BARCODE_JAN8** | **SII_PM_BARCODE_MODULE_WIDTH_2** | 134 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | 201 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | 268 |
| | **SII_PM_BARCODE_MODULE_WIDTH_5** | 335 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | 402 |
| **SII_PM_BARCODE_CODE93** | **SII_PM_BARCODE_MODULE_WIDTH_2** | 18 × number of barcode data + 56 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | 27 × number of barcode data + 84 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | 36 × number of barcode data + 112 |
| | **SII_PM_BARCODE_MODULE_WIDTH_5** | 45 × number of barcode data + 140 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | 54 × number of barcode data + 168 |
| **SII_PM_BARCODE_CODE128** | **SII_PM_BARCODE_MODULE_WIDTH_2** | 22 × number of barcode data + 26 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | 33 × number of barcode data + 39 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | 44 × number of barcode data + 52 |

| barcodeSymbol | moduleSize | **Width of Barcode Image** |
|---|---|---|
| **SII_PM_BARCODE_CODE128** | **SII_PM_BARCODE_MODULE_WIDTH_5** | 55 × number of barcode data + 65 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | 66 × number of barcode data + 78 |
| **SII_PM_BARCODE_GS1_OMNI_DIRECTIONAL** | **SII_PM_BARCODE_MODULE_WIDTH_2** | 192 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | 288 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | 384 |
| | **SII_PM_BARCODE_MODULE_WIDTH_5** | 480 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | 576 |
| **SII_PM_BARCODE_GS1_TRUNCATED** | **SII_PM_BARCODE_MODULE_WIDTH_2** | 192 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | 288 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | 384 |
| | **SII_PM_BARCODE_MODULE_WIDTH_5** | 480 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | 576 |
| **SII_PM_BARCODE_GS1_LIMITED** | **SII_PM_BARCODE_MODULE_WIDTH_2** | 158 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | 237 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | 316 |
| | **SII_PM_BARCODE_MODULE_WIDTH_5** | 395 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | 474 |
| **SII_PM_BARCODE_GS1_EXPANDED**[*1] | **SII_PM_BARCODE_MODULE_WIDTH_2** | number of barcode module × 2 |
| | **SII_PM_BARCODE_MODULE_WIDTH_3** | number of barcode module × 3 |
| | **SII_PM_BARCODE_MODULE_WIDTH_4** | number of barcode module × 4 |
| | **SII_PM_BARCODE_MODULE_WIDTH_5** | number of barcode module × 5 |
| | **SII_PM_BARCODE_MODULE_WIDTH_6** | number of barcode module × 6 |

*1: The number of barcode module is determined by the barcode data to be specified.

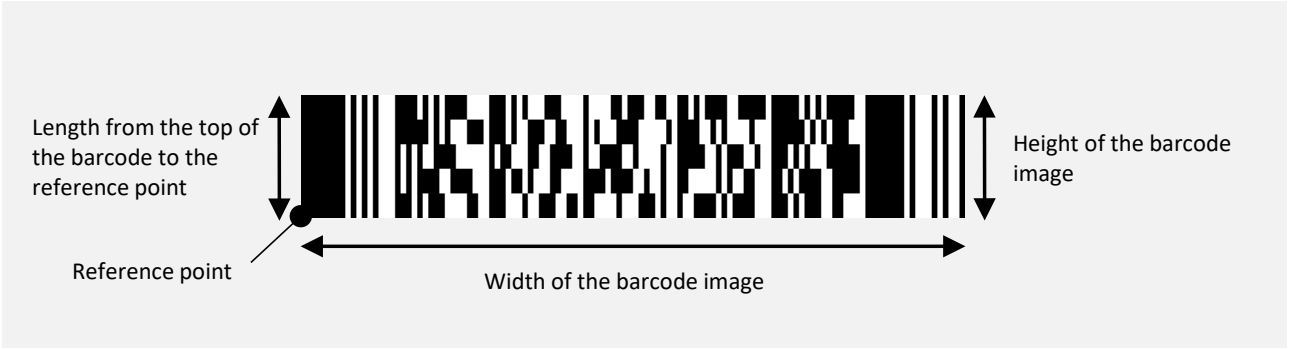
| barcodeSymbol | nwRatio | moduleSize | **Width of Barcode Image** |
|---|---|---|---|
| **SII_PM_BARCODE_CODE39** | **SII_PM_NWRATIO_1TO2** | **SII_PM_BARCODE_MODULE_WIDTH_2** | 26 × number of barcode data + 50 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_3** | 39 × number of barcode data + 75 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_4** | 52 × number of barcode data + 100 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_5** | 65 × number of barcode data + 125 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_6** | 78 × number of barcode data + 150 |
| | **SII_PM_NWRATIO_1TO2_5** | **SII_PM_BARCODE_MODULE_WIDTH_2** | 29 × number of barcode data + 56 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_3** | 45 × number of barcode data + 87 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_4** | 58 × number of barcode data + 112 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_5** | 74 × number of barcode data + 143 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_6** | 87 × number of barcode data + 168 |
| | **SII_PM_NWRATIO_1TO3** | **SII_PM_BARCODE_MODULE_WIDTH_2** | 32 × number of barcode data + 62 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_3** | 48 × number of barcode data + 93 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_4** | 64 × number of barcode data + 124 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_5** | 80 × number of barcode data + 155 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_6** | 96 × number of barcode data + 186 |
| **SII_PM_BARCODE_ITF** | **SII_PM_NWRATIO_1TO2** | **SII_PM_BARCODE_MODULE_WIDTH_2** | 14 × number of barcode data + 16 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_3** | 21 × number of barcode data + 24 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_4** | 28 × number of barcode data + 32 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_5** | 35 × number of barcode data + 40 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_6** | 42 × number of barcode data + 48 |
| | **SII_PM_NWRATIO_1TO2_5** | **SII_PM_BARCODE_MODULE_WIDTH_2** | 16 × number of barcode data + 17 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_3** | 25 × number of barcode data + 26 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_4** | 32 × number of barcode data + 34 |

| barcodeSymbol | nwRatio | moduleSize | Width of Barcode Image |
|---|---|---|---|
| **SII_PM_BARCODE_ITF** | **SII_PM_NWRATIO_1TO2_5** | **SII_PM_BARCODE_MODULE_WIDTH_5** | 41 × number of barcode data + 43 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_6** | 48 × number of barcode data + 51 |
| | **SII_PM_NWRATIO_1TO3** | **SII_PM_BARCODE_MODULE_WIDTH_2** | 18 × number of barcode data + 18 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_3** | 27 × number of barcode data + 27 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_4** | 36 × number of barcode data + 36 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_5** | 45 × number of barcode data + 45 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_6** | 54 × number of barcode data + 54 |
| **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) - 2 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_3** | 30 × number of data + 3 × (2 + number of wide data) - 3 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_4** | 40 × number of data + 4 × (2 + number of wide data) - 4 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_5** | 50 × number of data + 5 × (2 + number of wide data) - 5 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_6** | 60 × number of data + 6 × (2 + number of wide data) - 6 |
| | **SII_PM_NWRATIO_1TO2_5** | **SII_PM_BARCODE_MODULE_WIDTH_2** | 22 × number of data + 3 × (2 + number of wide data) - 2 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_3** | 34 × number of data + 5 × (2 + number of wide data) - 3 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_4** | 44 × number of data + 6 × (2 + number of wide data) - 4 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_5** | 56 × number of data + 8 × (2 + number of wide data) - 5 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_6** | 66 × number of data + 9 × (2 + number of wide data) - 6 |
| | **SII_PM_NWRATIO_1TO3** | **SII_PM_BARCODE_MODULE_WIDTH_2** | 24 × number of data + 4 × (2 + number of wide data) - 2 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_3** | 36 × number of data + 6 × (2 + number of wide data) - 3 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_4** | 48 × number of data + 8 × (2 + number of wide data) - 4 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_5** | 60 × number of data + 10 × (2 + number of wide data) - 5 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_6** | 72 × number of data + 12 × (2 + number of wide data) - 6 |

*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** | 244 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_3** | 366 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_4** | 488 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_5** | 610 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_6** | 732 |
| | 17 or 18 | **SII_PM_BARCODE_MODULE_WIDTH_2** | 298 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_3** | 447 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_4** | 596 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_5** | 745 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_6** | 894 |
| **SII_PM_BARCODE_JAN13_ADDON** | 14 or 15 | **SII_PM_BARCODE_MODULE_WIDTH_2** | 244 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_3** | 366 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_4** | 488 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_5** | 610 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_6** | 732 |
| | 17 or 18 | **SII_PM_BARCODE_MODULE_WIDTH_2** | 298 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_3** | 447 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_4** | 596 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_5** | 745 |
| | | **SII_PM_BARCODE_MODULE_WIDTH_6** | 894 |

### B.1.2 `printPDF417, printPageModePDF417`




(1) Height of the barcode image

Height of the barcode image[*1] = `moduleHeight` × `row`[*2]

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

(2) Width of the barcode image

When `pdf417Symbol` is **SII_PM_PDF417_STANDARD**:

Width of the barcode image = (17 × `column`[*1] + 69) × module size value

*1: `column`≠0

When `pdf417Symbol` is **SII_PM_PDF417_COMPACT**:

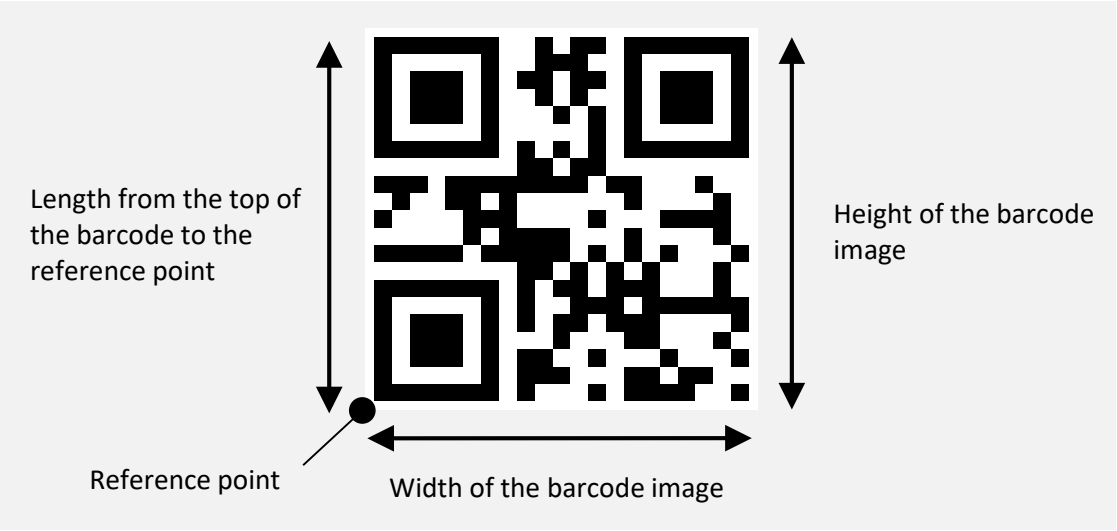
Width of the barcode image = (17 × `column`[*1] + 35) × module size value

*1: `column`≠0

**Module Size Value**

| `moduleSize` | **Module Size Value** |
|---|---|
| **SII_PM_PDF417_MODULE_WIDTH_2** | 2 |
| **SII_PM_PDF417_MODULE_WIDTH_3** | 3 |
| **SII_PM_PDF417_MODULE_WIDTH_4** | 4 |
| **SII_PM_PDF417_MODULE_WIDTH_5** | 5 |
| **SII_PM_PDF417_MODULE_WIDTH_6** | 6 |
| **SII_PM_PDF417_MODULE_WIDTH_7** | 7 |
| **SII_PM_PDF417_MODULE_WIDTH_8** | 8 |

### B.1.3 `printQRCode, printPageModeQRCode`




(1) Height and width of the barcode image

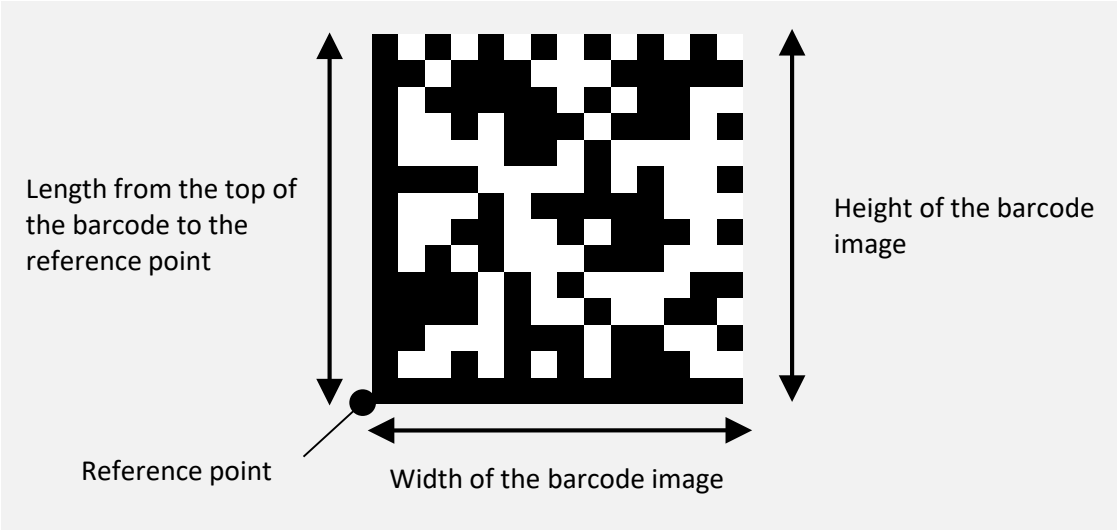
Height[*1] and width of the barcode image = (4 × version[*2] + 17) × module size value

*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 |
| **SII_PM_QR_MODULE_SIZE_12** | 12 |
| **SII_PM_QR_MODULE_SIZE_13** | 13 |
| **SII_PM_QR_MODULE_SIZE_14** | 14 |
| **SII_PM_QR_MODULE_SIZE_15** | 15 |
| **SII_PM_QR_MODULE_SIZE_16** | 16 |

### B.1.4 `printDataMatrix, printPageModeDataMatrix`




(1) Height and width of the barcode image

Height of the barcode image = number of vertical module × module size value

Width of the barcode image = number of horizontal module × module size value

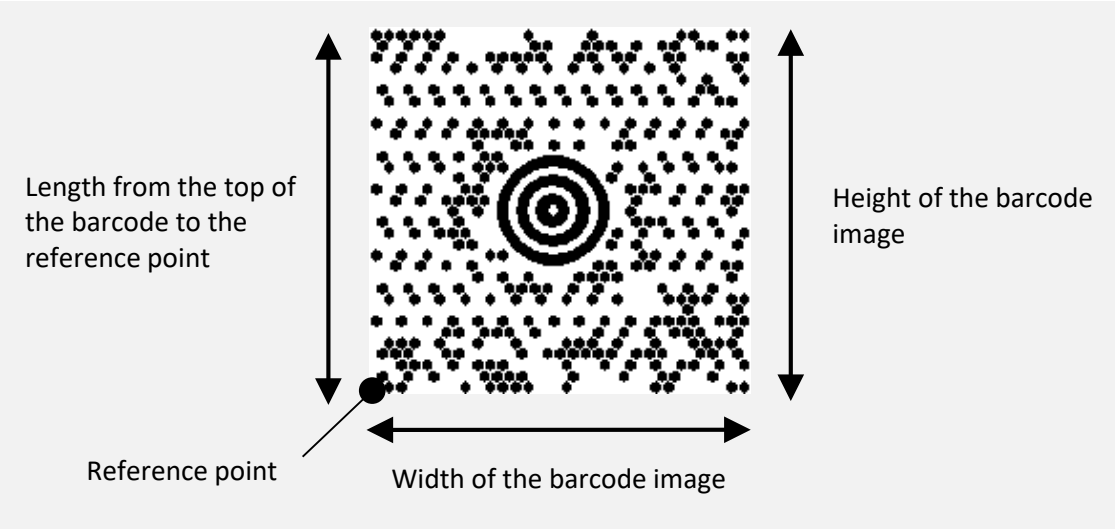
| dataMatrixModule | Number of Vertical Module | Number of Horizontal Module |
|---|---|---|
| **SII_PM_DATA_MATRIX_10_10** | 10 | 10 |
| **SII_PM_DATA_MATRIX_12_12** | 12 | 12 |
| **SII_PM_DATA_MATRIX_14_14** | 14 | 14 |
| **SII_PM_DATA_MATRIX_16_16** | 16 | 16 |
| **SII_PM_DATA_MATRIX_18_18** | 18 | 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 |
| **SII_PM_DATA_MATRIX_120_120** | 120 | 120 |

| dataMatrixModule | Number of Vertical Module | Number of Horizontal Module |
|---|---|---|
| **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 |
| **SII_PM_DATAMATRIX_MODULE_SIZE_12** | 12 |
| **SII_PM_DATAMATRIX_MODULE_SIZE_13** | 13 |
| **SII_PM_DATAMATRIX_MODULE_SIZE_14** | 14 |
| **SII_PM_DATAMATRIX_MODULE_SIZE_15** | 15 |
| **SII_PM_DATAMATRIX_MODULE_SIZE_16** | 16 |

### B.1.5 `printMaxicode, printPageModeMaxicode`




(1) Height of the barcode image

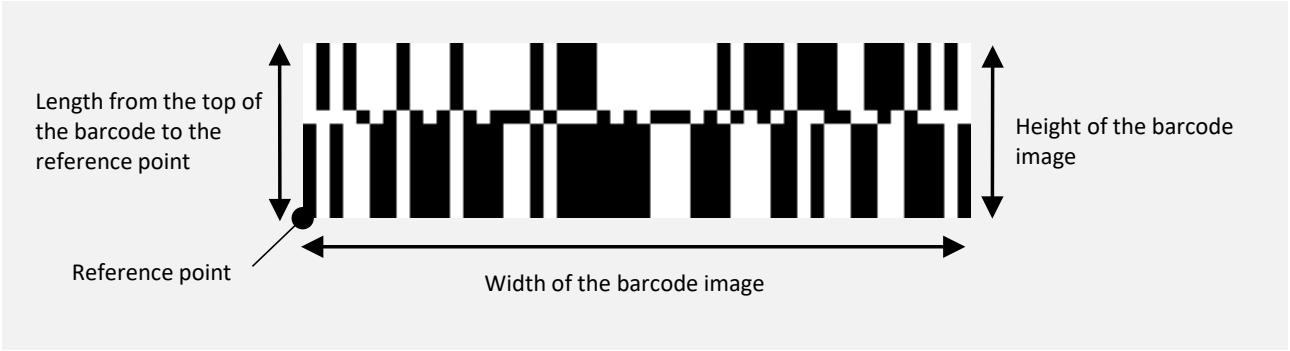
Height of the barcode image*1 = 200

*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 = 210

### B.1.6 `printGS1DataBarStacked, printPageModeGS1DataBarStacked`




(1) Height and width of the barcode image

Height of the barcode image[*1] = 13 × module size value

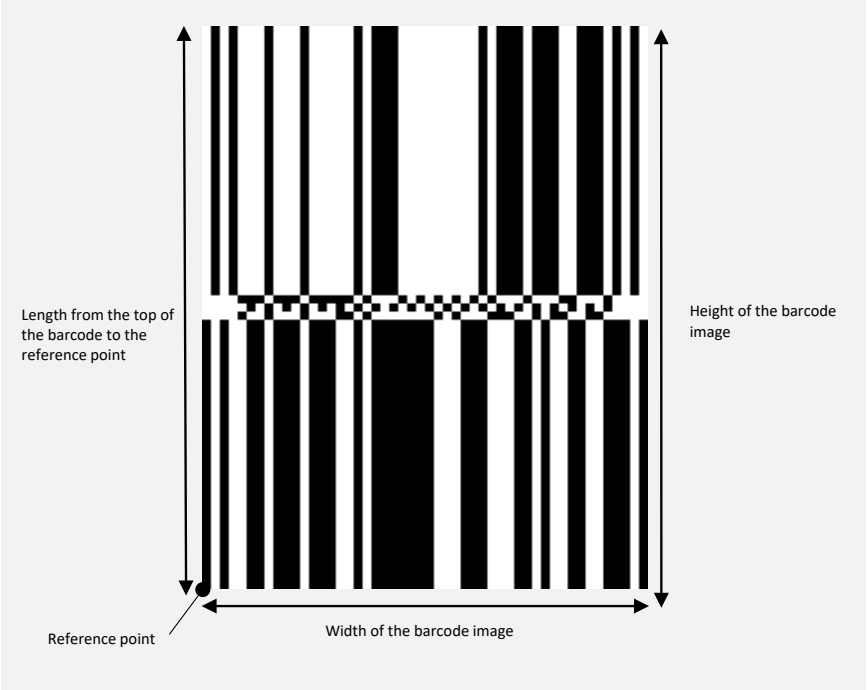
*1: Height of the barcode image = Length from the top of the barcode to the reference point

Width of the barcode image = 50 × module size value

**Module Size Value**

| moduleSize | Module Size Value |
|---|---|
| **SII_PM_GS1DATABAR_MODULE_SIZE_2** | 2 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_3** | 3 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_4** | 4 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_5** | 5 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_6** | 6 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_7** | 7 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_8** | 8 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_9** | 9 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_10** | 10 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_11** | 11 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_12** | 12 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_13** | 13 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_14** | 14 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_15** | 15 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_16** | 16 |

### B.1.7 `printGS1DataBarStackedOmnidirectional`, `printPageModeGS1DataBarStackedOmnidirectional`




(1) Height and width of the barcode image

Height of the barcode image[*1] = (`moduleHeight` × 2 + 3) × module size value

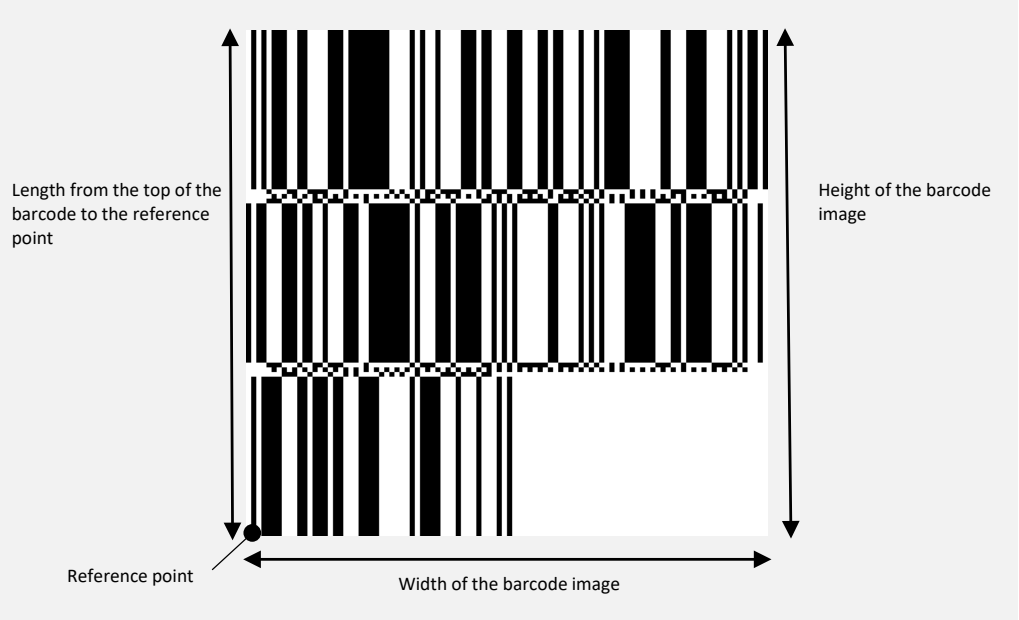
*1: Height of the barcode image = Length from the top of the barcode to the reference point

Width of the barcode image = 50 × module size value

**Module Size Value**

| `moduleSize` | **Module Size Value** |
|---|---|
| **SII_PM_GS1DATABAR_MODULE_SIZE_2** | 2 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_3** | 3 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_4** | 4 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_5** | 5 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_6** | 6 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_7** | 7 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_8** | 8 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_9** | 9 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_10** | 10 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_11** | 11 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_12** | 12 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_13** | 13 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_14** | 14 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_15** | 15 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_16** | 16 |

### B.1.8 `printGS1DataBarExpandedStacked`, `printPageModeGS1DataBarExpandedStacked`




(1) Height and width of the barcode image

Height of the barcode image[*1] = ((34 + 3) × number of row[*2] + 34) × module size value

*1: Height of the barcode image = Length from the top of the barcode to the reference point
*2: The number of row is determined by the barcode data.

Width of the barcode image = (4 + 49 × `column` / 2) × module size value

**Module Size Value**

| `moduleSize` | **Module Size Value** |
|---|---|
| **SII_PM_GS1DATABAR_MODULE_SIZE_2** | 2 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_3** | 3 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_4** | 4 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_5** | 5 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_6** | 6 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_7** | 7 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_8** | 8 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_9** | 9 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_10** | 10 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_11** | 11 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_12** | 12 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_13** | 13 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_14** | 14 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_15** | 15 |
| **SII_PM_GS1DATABAR_MODULE_SIZE_16** | 16 |

# 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.

Seiko Instruments Inc.
1-8, Nakase, Mihama-ku, Chiba-shi,
Chiba 261-8507, Japan
Print System Division
Telephone:+81-43-211-1106
Facsimile:+81-43-211-8037

Seiko Instruments USA Inc.
Thermal Printer Div.
21221 S. Western Avenue, Suite 250, Torrance, CA 90501, USA
Telephone:+1-310-517-7778  Facsimile:+1-310-517-7779

Seiko Instruments GmbH
Siemensstrasse 9, D-63263 Neu-Isenburg, Germany
Telephone:+49-6102-297-0  Facsimile:+49-6102-297-222
info@seiko-instruments.de

Seiko Instruments Trading (H.K.) Ltd.
7/F, Ying Tung Industrial Building, 802 Lai Chi Kok Road, Kowloon, Hong Kong
Telephone:+852-2494-5111  Facsimile:+852-2424-0901

(Specifications are subject to change without notice.)